SIEMENS

SIMATIC

STEP 7 Professional V13.0

Manual de sistema

Descripción del sistema STEP 7 y WinCC 1

Léame 2

Instalación 3

Migrar proyectos y programas 4

Primeros pasos 5

Introducción al TIA Portal 6

Editar proyectos 7

Editar dispositivos y redes 8

Programar el PLC 9

Visualización de procesos 10

Utilizar funciones tecnológicas 11

Uso del Team Engineering 12

Uso de las funciones Online y diagnóstico 13

Documentación del hardware 14

Impresión de la Ayuda en pantalla

02/2014

Notas jurídicas

Filosofía en la señalización de advertencias y peligros

Este manual contiene las informaciones necesarias para la seguridad personal así como para la prevención de daños materiales. Las informaciones para su seguridad personal están resaltadas con un triángulo de advertencia; las informaciones para evitar únicamente daños materiales no llevan dicho triángulo. De acuerdo al grado de peligro las consignas se representan, de mayor a menor peligro, como sigue.

⚠ PELIGRO
Significa que, si no se adoptan las medidas preventivas adecuadas **se producirá** la muerte, o bien lesiones corporales graves.

⚠ ADVERTENCIA
Significa que, si no se adoptan las medidas preventivas adecuadas **puede producirse** la muerte o bien lesiones corporales graves.

⚠ PRECAUCIÓN
Significa que si no se adoptan las medidas preventivas adecuadas, pueden producirse lesiones corporales.

ATENCIÓN
Significa que si no se adoptan las medidas preventivas adecuadas, pueden producirse daños materiales.

Si se dan varios niveles de peligro se usa siempre la consigna de seguridad más estricta en cada caso. Si en una consigna de seguridad con triángulo de advertencia se alarma de posibles daños personales, la misma consigna puede contener también una advertencia sobre posibles daños materiales.

Personal cualificado

El producto/sistema tratado en esta documentación sólo deberá ser manejado o manipulado por **personal cualificado** para la tarea encomendada y observando lo indicado en la documentación correspondiente a la misma, particularmente las consignas de seguridad y advertencias en ella incluidas. Debido a su formación y experiencia, el personal cualificado está en condiciones de reconocer riesgos resultantes del manejo o manipulación de dichos productos/sistemas y de evitar posibles peligros.

Uso previsto o de los productos de Siemens

Considere lo siguiente:

⚠ ADVERTENCIA
Los productos de Siemens sólo deberán usarse para los casos de aplicación previstos en el catálogo y la documentación técnica asociada. De usarse productos y componentes de terceros, éstos deberán haber sido recomendados u homologados por Siemens. El funcionamiento correcto y seguro de los productos exige que su transporte, almacenamiento, instalación, montaje, manejo y mantenimiento hayan sido realizados de forma correcta. Es preciso respetar las condiciones ambientales permitidas. También deberán seguirse las indicaciones y advertencias que figuran en la documentación asociada.

Marcas registradas

Todos los nombres marcados con ® son marcas registradas de Siemens AG. Los restantes nombres y designaciones contenidos en el presente documento pueden ser marcas registradas cuya utilización por terceros para sus propios fines puede violar los derechos de sus titulares.

Exención de responsabilidad

Hemos comprobado la concordancia del contenido de esta publicación con el hardware y el software descritos. Sin embargo, como es imposible excluir desviaciones, no podemos hacernos responsable de la plena concordancia. El contenido de esta publicación se revisa periódicamente; si es necesario, las posibles las correcciones se incluyen en la siguiente edición.

Siemens AG
Industry Sector
Postfach 48 48
90026 NÜRNBERG
ALEMANIA

℗ 02/2014 Sujeto a cambios sin previo aviso

Copyright © Siemens AG 2014.
Reservados todos los derechos

Índice

1 Descripción del sistema STEP 7 y WinCC....27

1.1 Escalado de STEP 7 y WinCC en el TIA Portal....27

1.2 Opciones para el sistema de ingeniería STEP 7....28

1.3 Opciones para sistemas de ingeniería y runtime WinCC....28

2 Léame....31

2.1 Indicaciones generales....31

2.2 STEP 7 Professional....35
2.2.1 Información de seguridad....35
2.2.2 Novedades....37
2.2.3 Indicaciones de manejo....37
2.2.4 Editar dispositivos y redes....40
2.2.4.1 Indicaciones generales relativas a dispositivos y redes....40
2.2.4.2 Uso de módulos en el S7-1200....40
2.2.4.3 Sustitución de los módulos de posicionamiento ET 200S....41
2.2.4.4 CP 343-2 en SIMATIC S7 Embedded Controller EC31-RTX....42
2.2.4.5 Indicaciones relativas a Online y diagnóstico....42
2.2.4.6 Componentes de red....43
2.2.5 Programación del PLC....46
2.2.5.1 Indicaciones generales relativas a la programación del PLC....46
2.2.5.2 Instrucciones....50
2.2.5.3 Comprobar el programa de usuario....52
2.2.6 Inter Project Engineering (IPE)....53
2.2.6.1 Indicaciones relativas a IPE....53
2.2.7 Funciones tecnológicas....54
2.2.7.1 Indicaciones sobre las funciones tecnológicas....54
2.2.7.2 Indicaciones sobre las funciones tecnológicas....55
2.2.7.3 Indicaciones sobre las funciones tecnológicas....55
2.2.7.4 SIMATIC STEP 7 Easy Motion Control....55
2.2.7.5 SIMATIC STEP 7 PID Professional....55

2.3 WinCC Basic....56
2.3.1 Indicaciones de seguridad....56
2.3.2 Novedades....58
2.3.3 Indicaciones de uso....58
2.3.4 Migración....60
2.3.5 Engineering System....64
2.3.5.1 Imágenes y objetos de imagen....64
2.3.5.2 Variables y conexiones....68
2.3.5.3 Sistema de avisos y visor de avisos....70
2.3.5.4 Funciones de sistema....71
2.3.5.5 Recetas....71
2.3.5.6 Administración de usuarios....71
2.3.5.7 Comunicación....72
2.3.6 Funciones generales....74

2.3.7 Compilar y cargar....74
2.3.8 Runtime....77
2.3.8.1 Indicaciones relativas al funcionamiento en runtime....77
2.3.8.2 Indicaciones relativas al funcionamiento de Panels en runtime....79
2.3.9 Paneles de operador....80
2.3.9.1 Indicaciones acerca de paneles de operador....80

3 Instalación....85

3.1 Indicaciones de instalación....85

3.2 Requisitos del sistema para la instalación....86
3.2.1 Indicaciones relativas a licencias....86
3.2.2 Indicaciones relativas a los requisitos del sistema....87
3.2.3 Requisitos del sistema STEP 7 Professional....88
3.2.3.1 Concesión de licencia de STEP 7....88
3.2.3.2 Uso de licencias y claves de licencia....90
3.2.3.3 STEP 7 Professional para PC-CP....92
3.2.3.4 Requisitos de software y hardware de STEP 7....93
3.2.4 Requisitos del sistema en WinCC Basic....96
3.2.4.1 Requisitos de software y hardware....96
3.2.4.2 Instalaciones paralelas....100
3.2.4.3 Licencias y power packs....101

3.3 Informe de instalación....106

3.4 Iniciar la instalación....107

3.5 Comprobar disponibilidad de actualizaciones y Support Packages y proceder a su instalación....109

3.6 Mostrar el software instalado....111

3.7 Modificar o actualizar productos instalados....112

3.8 Reparación de los productos instalados....114

3.9 Iniciar la desinstalación....115

3.10 Instalación y desinstalación de la herramienta de migración....117
3.10.1 Requisitos del sistema....117
3.10.2 Instalación de la herramienta de migración....117
3.10.3 Desinstalación de la herramienta de migración....118

4 Migrar proyectos y programas....119

4.1 Migrar proyectos en un proyecto del TIA Portal....119
4.1.1 Migración de proyectos con el TIA Portal....119
4.1.2 Preparación de proyectos con la herramienta de migración....121
4.1.2.1 Migración de proyectos con la herramienta de migración....121
4.1.2.2 Inicio de la herramienta de migración....122
4.1.2.3 Crear el archivo de migración....123
4.1.3 Migración de proyectos....124
4.1.4 Mostrar el desarrollo de la migración....125
4.1.5 Mostrar el informe de la migración....126
4.1.6 Migrar proyectos de STEP 7....126
4.1.6.1 Migración de proyectos STEP 7....126
4.1.6.2 Requisitos para la migración de proyectos STEP 7....127
4.1.6.3 Comprobar la capacidad de migración de los proyectos STEP 7....129

4.1.6.4 Comprobación de coherencia en el proyecto de origen antes de la migración....130
4.1.6.5 Eliminación de componentes hardware no soportados....130
4.1.6.6 Particularidades de la migración....131
4.1.7 Migrar proyectos de WinCC flexible (Basic)....146
4.1.7.1 Principios básicos (WinCC flexible)....146
4.1.7.2 Migrar los datos de ingeniería (WinCC flexible)....151
4.1.7.3 Migrar los datos runtime (WinCC flexible)....162
4.1.7.4 Migrar los proyectos integrados (WinCC flexible)....165
4.1.7.5 Referencia (WinCC flexible)....168
4.1.8 Migrar proyectos integrados....176
4.1.8.1 Migración de un proyecto integrado....176
4.1.8.2 Reajustar proyectos integrados....178
4.1.8.3 Convertir CPUs no especificadas en CPUs especificadas....180
4.1.8.4 Crear una conexión HMI integrada....181
4.1.8.5 Volver a enlazar variables HMI....183
4.1.8.6 Borrar una conexión no especificada....184

4.2 Migración de programas PLC a una CPU S7-1500....185
4.2.1 Principios básicos de la migración de programas PLC....185
4.2.2 Ejecución de la migración....186
4.2.3 Visualización de un archivo de registro....188
4.2.4 Particularidades en la migración de programas PLC....189
4.2.4.1 Indicaciones sobre la migración de programas PLC....189
4.2.4.2 Migración de instrucciones a S7-1500....195
4.2.4.3 Migración de programas KOP/FUP a S7-1500....218
4.2.4.4 Migración de programas AWL a S7-1500....222
4.2.4.5 Migración de programas SCL a S7-1500....235
4.2.4.6 Migración de programas GRAPH a S7-1500....240

4.3 Migración de S7-1200 a firmware V4 y posterior....244
4.3.1 Principios básicos de la migración a V4....244
4.3.2 Realizar una migración a V4....246
4.3.3 Particularidades tras la migración a V4....247

4.4 Recomendaciones de programación....250
4.4.1 Resumen de las nuevas funciones de la CPU S7-1500....250
4.4.2 Direccionamiento simbólico....253
4.4.3 Uso de temporizadores y contadores CEI....255
4.4.4 Uso flexible de la salida de habilitación ENO....256
4.4.5 Uso de instrucciones MOVE en AWL....258
4.4.6 Realizar un acceso de array con índice variable....260
4.4.7 Uso de tipos de datos PLC (UDT)....262
4.4.8 Consultar y eliminar errores en el código del programa....265
4.4.9 Uso de instrucciones VARIANT....269

5 Primeros pasos....277

5.1 Documentación en forma de Getting Started....277

6 Introducción al TIA Portal....279

6.1 Interfaz de usuario y manejo....279
6.1.1 Iniciar, configurar y cerrar el TIA Portal....279
6.1.1.1 Iniciar y cerrar el TIA Portal....279
6.1.1.2 Resumen de los ajustes del programa....279
6.1.1.3 Sinopsis de la configuración de los editores de scripts y de textos....281

6.1.1.4 Resumen de los ajustes de impresión....282
6.1.1.5 Modificar la configuración....283
6.1.2 Estructura de la interfaz de usuario....283
6.1.2.1 Vistas....283
6.1.2.2 Vista del portal....284
6.1.2.3 Vista del proyecto....285
6.1.2.4 Vista de librerías....288
6.1.2.5 Árbol del proyecto....288
6.1.2.6 Área de trabajo....291
6.1.2.7 Ventana de inspección....299
6.1.2.8 Task Cards....301
6.1.2.9 Proyectos de referencia....303
6.1.2.10 Vista detallada....305
6.1.2.11 Ventana sinóptica....306
6.1.2.12 Diseño de la interfaz de usuario....310
6.1.3 Manejo del teclado en el TIA Portal....314
6.1.3.1 Manejar el TIA Portal con el teclado....314
6.1.3.2 Mostrar una relación de todas las combinaciones de teclas....314
6.1.3.3 Funciones básicas del TIA Portal....314
6.1.3.4 Uso de funciones de proyecto....316
6.1.3.5 Organización de ventanas....316
6.1.3.6 Navegación por la interfaz del programa....317
6.1.3.7 Ajuste de editores....319
6.1.3.8 Editar objetos....320
6.1.3.9 Edición de textos....321
6.1.3.10 Editar tablas....322
6.1.3.11 Uso de funciones online....324
6.1.3.12 Uso del teclado de pantalla....324
6.1.4 Peculiaridades sujetas al sistema operativo....325
6.1.4.1 Influencia de los derechos de usuario....325
6.1.4.2 Ampliar los derechos de usuario....326

6.2 Ayuda del sistema de información....327
6.2.1 Generalidades acerca del sistema de información....327
6.2.2 Abrir la ayuda suministrada....330
6.2.3 Buscar palabras clave en la Ayuda....331
6.2.4 Buscar en todo el texto....331
6.2.5 Utilizar Favoritos....332
6.2.6 Imprimir temas de Ayuda....333
6.2.7 Configurar la visualización de tooltips y tooltips en cascada....334
6.2.8 Consignas de seguridad....334
6.2.9 Elaboración de documentación individual....336

7 Editar proyectos....339

7.1 Principios básicos de los proyectos....339

7.2 Uso de informes....339

7.3 Crear y gestionar proyectos....340
7.3.1 Crear un proyecto nuevo....340
7.3.2 Compatibilidad de proyectos....341
7.3.3 Abrir proyectos....342
7.3.4 Actualizar proyectos....343
7.3.5 Visualización de las propiedades del proyecto....346

7.3.6 Guardar proyectos....346
7.3.7 Cerrar proyectos....347
7.3.8 Borrar proyectos....348
7.3.9 Trabajar con proyectos multilingües....348
7.3.9.1 Principios básicos de los textos del proyecto....348
7.3.9.2 Seleccionar los idiomas del proyecto....351
7.3.9.3 Definir el idioma de edición....351
7.3.9.4 Traducción tabular de todos los textos del proyecto....352
7.3.9.5 Traducción de textos correspondientes a objetos individuales....353
7.3.9.6 Traducir textos con ayuda de textos de referencia....353
7.3.9.7 Importar y exportar textos del proyecto....354
7.3.9.8 Ejemplo de aplicación para proyectos en varios idiomas....356
7.3.10 Archivación y desarchivación de proyectos....357
7.3.10.1 Uso de ficheros de proyecto....357
7.3.10.2 Archivación de proyectos....358
7.3.10.3 Desarchivación de proyectos....359

7.4 Uso de proyectos de referencia....360
7.4.1 Principios básicos de los proyectos de referencia....360
7.4.2 Apertura y cierre de un proyecto de referencia....361
7.4.3 Comparación de proyectos de referencia....362

7.5 Edición de los datos de proyecto....363
7.5.1 Compilación y carga de los datos de proyecto....363
7.5.1.1 Compilar datos de proyecto....363
7.5.1.2 Cargar los datos de proyecto....364
7.5.2 Comparar datos de proyecto....373
7.5.2.1 Principios básicos de la comparación de datos de proyecto....373
7.5.2.2 Realizar una comparación online/offline....374
7.5.2.3 Realizar una comparación offline/offline....374
7.5.2.4 Utilizar el editor de comparación....375
7.5.3 Proteger los datos de proyecto....389
7.5.3.1 Protección de los datos de proyecto....389
7.5.3.2 Borrado de los derechos de acceso a dispositivos....389
7.5.4 Imprimir contenidos del proyecto....390
7.5.4.1 Imprimir documentación del proyecto....390
7.5.4.2 Imprimir tiras rotulables para módulos....409

7.6 Deshacer y rehacer acciones....413
7.6.1 Principios básicos para deshacer y rehacer acciones....413
7.6.2 Deshacer una acción....415
7.6.3 Rehacer una acción....416

7.7 Buscar y reemplazar en proyectos....417
7.7.1 Información acerca de la función de búsqueda....417
7.7.2 Buscar y reemplazar....418

7.8 Trabajar con listas de textos....420
7.8.1 Listas de textos....420
7.8.2 Crear listas de textos de usuario....422
7.8.3 Editar listas de textos de usuario....422
7.8.4 Editar listas de textos de sistema....423

7.9 Utilizar Memory Cards....424
7.9.1 Principios básicos de las Memory Cards....424

7.9.2 Agregar un lector de tarjetas personalizado....425
7.9.3 Acceder a Memory Cards....425
7.9.4 Mostrar las propiedades de Memory Cards....426

7.10 Utilizar librerías....427
7.10.1 Principios básicos de las librerías....427
7.10.2 Uso de la Task Card "Librerías"....429
7.10.2.1 Sinopsis de la Task Card "Librerías"....429
7.10.2.2 Utilizar la vista de elementos....431
7.10.3 Uso de la vista de librerías....432
7.10.3.1 Sinopsis de la vista de librerías....432
7.10.3.2 Apertura y cierre de la vista de librerías....434
7.10.4 Uso de la administración de librerías....435
7.10.4.1 Sinopsis de la administración de librerías....435
7.10.4.2 Abrir la administración de librerías....437
7.10.4.3 Filtrar la visualización de tipos....438
7.10.4.4 Visualizar instancias del proyecto....438
7.10.4.5 Visualizar referencias cruzadas de una instancia....439
7.10.4.6 Visualizar relaciones entre objetos de librerías....440
7.10.5 Utilizar librerías globales....440
7.10.5.1 Crear librería global....440
7.10.5.2 Compatibilidad de las librerías globales....441
7.10.5.3 Abrir una librería global....443
7.10.5.4 Actualizar librerías globales....444
7.10.5.5 Mostrar las propiedades de las librerías globales....445
7.10.5.6 Mostrar los informes de las librerías globales....446
7.10.5.7 Guardar una librería global....447
7.10.5.8 Cerrar una librería global....448
7.10.5.9 Borrar una librería global....448
7.10.5.10 Archivar y desactivar librerías globales....449
7.10.5.11 Utilizar librerías de empresa globales....452
7.10.6 Crear carpetas en una librería....454
7.10.7 Utilizar plantillas maestras....455
7.10.7.1 Principios básicos de las plantillas maestras....455
7.10.7.2 Agregar plantillas maestras....456
7.10.7.3 Filtrar plantillas maestras....457
7.10.7.4 Uso de plantillas maestras....457
7.10.8 Utilizar tipos y sus versiones....458
7.10.8.1 Principios básicos de los tipos....458
7.10.8.2 Estado de las versiones de tipo....460
7.10.8.3 Mostrar versión de tipo habilitada....461
7.10.8.4 Mostrar propiedades de un tipo o de una versión....462
7.10.8.5 Trabajar con tipos en la librería del proyecto....463
7.10.8.6 Trabajar con tipos en las librerías globales....474
7.10.8.7 Asignar una versión....477
7.10.9 Editar elementos de librería....479
7.10.10 Actualizar una librería con los contenidos de otra librería....481
7.10.11 Armonizar nombres y estructura de rutas....483
7.10.12 Limpiar librería....484
7.10.13 Comparación de elementos de librerías....485

7.11 Uso de referencias cruzadas....487
7.11.1 Uso de referencias cruzadas....487

7.12 Simulación de dispositivos....487
7.12.1 Simulación de dispositivos....487
7.12.2 Iniciar la simulación....488

8 Editar dispositivos y redes....489

8.1 Configurar dispositivos y redes....489
8.1.1 Editor de hardware y redes....489
8.1.1.1 Resumen del editor de hardware y redes....489
8.1.1.2 Vista de redes....491
8.1.1.3 Vista de dispositivos....493
8.1.1.4 Vista topológica....496
8.1.1.5 Resumen de los ajustes de la configuración hardware....497
8.1.1.6 Imprimir configuraciones de hardware y redes....498
8.1.1.7 Activar vista preliminar de salto de página para la impresión....500
8.1.1.8 Modificar las opciones de impresión....500
8.1.1.9 Ventana de inspección....501
8.1.1.10 Catálogo de hardware....503
8.1.1.11 Activar el soporte del producto....505
8.1.1.12 Mostrar el soporte del producto para componentes de hardware....506
8.1.1.13 Comandos de teclado: navegación en el editor....507
8.1.1.14 Comandos de teclado: editar objetos....508
8.1.2 Configurar dispositivos....509
8.1.2.1 Principios básicos....509
8.1.2.2 Configurar dispositivos individuales....518
8.1.2.3 Comparar dispositivos....533
8.1.3 Configurar redes....534
8.1.3.1 Conectar dispositivos en red....534
8.1.3.2 Comunicación vía conexiones....557
8.1.3.3 Mostrar y configurar topología....733
8.1.3.4 Industrial Ethernet Security....751
8.1.4 Crear configuraciones....896
8.1.4.1 Información importante sobre el servidor web....896
8.1.4.2 Información importante sobre los modos de operación en PROFIBUS DP....897
8.1.4.3 Configurar sistemas de automatización (S7-300, S7-400, S7-1500)....898
8.1.4.4 Configurar sistemas de automatización (S7-1200)....971
8.1.4.5 Configuraciones para acoplamientos punto a punto (S7-1500)....1009
8.1.4.6 Configuraciones para acoplamientos punto a punto (S7-300, S7-400)....1068
8.1.4.7 Configurar estaciones PC....1197
8.1.4.8 CM/CP S7-1200....1207
8.1.4.9 SCALANCE X, W y M....1223
8.1.4.10 Configuración de PROFIBUS DP....1386
8.1.4.11 Configurar PROFIBUS DP (S7-1200)....1463
8.1.4.12 Configuraciones para PROFINET IO....1486
8.1.4.13 Acoplamiento de bus con acoplador PN/PN....1607
8.1.4.14 Integración de herramientas externas....1608
8.1.4.15 Cargar la configuración....1610
8.1.5 Información adicional sobre configuraciones....1621
8.1.5.1 Funcionamiento de las CPUs S7-1200....1621
8.1.5.2 Funcionamiento de las CPU S7-300/400....1700
8.1.5.3 Funcionamiento de las CPUs S7-1500....1900
8.1.5.4 CPs, estaciones PC y componentes de red....1994
8.1.5.5 Sistemas de identificación....2088

8.1.5.6 Periferia descentralizada....2095
8.1.5.7 Configuración IPv6....2151
8.1.6 Mostrar avisos....2153
8.1.6.1 Sinopsis del visor de avisos....2153
8.1.6.2 Vista de fichero....2153
8.1.6.3 Estructura de los avisos en la vista de fichero....2154
8.1.6.4 Recibir avisos....2154
8.1.6.5 Exportar el fichero....2155
8.1.6.6 Vaciar el fichero....2155
8.1.6.7 Vista "Avisos activos"....2155
8.1.6.8 Estructura de los avisos en la vista "Avisos activos"....2155
8.1.6.9 Estado de los avisos....2156
8.1.6.10 Acusar avisos....2156
8.1.6.11 Ignorar avisos....2157
8.1.6.12 Manejo del teclado en el visor de avisos....2158

8.2 Diagnosticar dispositivos y redes....2158
8.2.1 Diagnosticar hardware....2158
8.2.1.1 Resumen del diagnóstico de hardware....2158
8.2.1.2 Mostrar valores actuales y no modificables de propiedades de módulos parametrizables. ...2169
8.2.1.3 Mostrar valores actuales de propiedades de módulos dinámicas....2176
8.2.1.4 Comprobar los fallos de un módulo....2180
8.2.1.5 Modificar las propiedades de un módulo o de la PG/el PC....2187
8.2.1.6 Diagnóstico en estado operativo STOP....2204
8.2.1.7 Accesos online en la vista online y de diagnóstico....2207
8.2.1.8 Comprobar fallos en subredes PROFIBUS DP....2210
8.2.2 Diagnosticar conexiones....2213
8.2.2.1 Vista general del diagnóstico de conexiones....2213
8.2.2.2 Ver el estado de conexión mediante iconos....2214
8.2.2.3 Diagnosticar conexiones detalladamente....2215
8.2.3 Diagnóstico del sistema con 'Report System Errors'....2218
8.2.3.1 Introducción al diagnóstico del sistema con "Report System Errors"....2218
8.2.3.2 Principios básicos del diagnóstico del sistema....2218
8.2.3.3 Módulos con posibilidades de diagnóstico limitadas....2219
8.2.3.4 Bloques de diagnóstico para notificar errores de sistema....2221
8.2.3.5 Propiedades de los bloques....2221
8.2.3.6 OBs de error soportados....2222
8.2.3.7 DB de estado de diagnóstico....2224
8.2.3.8 Visualización de los ajustes del diagnóstico del sistema....2227
8.2.3.9 Configuración....2228
8.2.4 Diagnóstico del sistema para PLCs S7-1500....2230
8.2.4.1 Introducción....2230
8.2.4.2 Principios básicos del diagnóstico del sistema....2230
8.2.4.3 Visualización de los ajustes del diagnóstico del sistema....2230
8.2.4.4 Configuración....2231
8.2.5 Utilizar opciones de diagnóstico adicionales....2231
8.2.5.1 Diagnosticar el hardware....2231
8.2.5.2 Diagnosticar conexiones....2248

9 Programar el PLC....2249

9.1 Cree el programa de usuario....2249
9.1.1 Principios básicos de la programación....2249
9.1.1.1 Sistema operativo y programa de usuario....2249

9.1.1.2 Bloques del programa de usuario....2250
9.1.1.3 Llamadas de bloque....2263
9.1.1.4 Utilizar y direccionar operandos....2280
9.1.1.5 Tipos de datos....2321
9.1.1.6 Control del flujo de programa....2539
9.1.2 Declarar variables PLC....2566
9.1.2.1 Sinopsis de las tablas de variables PLC....2566
9.1.2.2 Estructura de la tablas de variables PLC....2567
9.1.2.3 Reglas para variables PLC....2570
9.1.2.4 Crear y gestionar tablas de variables PLC....2573
9.1.2.5 Declarar variables PLC....2575
9.1.2.6 Agrupar variables PLC en estructuras para entradas y salidas....2579
9.1.2.7 Declarar constantes globales....2583
9.1.2.8 Editar propiedades....2585
9.1.2.9 Observar variables PLC....2588
9.1.2.10 Editar tablas de variables PLC....2588
9.1.3 Crear y gestionar bloques....2595
9.1.3.1 Crear bloques....2595
9.1.3.2 Definir propiedades de bloques....2607
9.1.3.3 Gestionar bloques....2614
9.1.4 Programar bloques....2619
9.1.4.1 Editor de programación....2619
9.1.4.2 Programar bloques lógicos....2642
9.1.4.3 Programar bloques de datos....2937
9.1.4.4 Programar tipos de datos PLC....2967
9.1.4.5 Utilizar archivos fuente externos....2976
9.1.5 Comparar programas de PLC....2982
9.1.5.1 Principios básicos para comparar programas de PLC....2982
9.1.5.2 Comparar bloques....2986
9.1.5.3 Comparar variables PLC....3009
9.1.5.4 Comparar tipos de datos PLC....3010
9.1.6 Compilar y cargar bloques....3012
9.1.6.1 Compilar bloques....3012
9.1.6.2 Cargar bloques....3018
9.1.6.3 Cargar bloques....3037
9.1.7 Proteger bloques....3045
9.1.7.1 Proteger bloques....3045
9.1.7.2 Establecer y quitar la protección contra copia de un bloque....3048
9.1.7.3 Configurar la protección de know-how para bloques....3049
9.1.7.4 Abrir bloques con protección de know-how....3050
9.1.7.5 Imprimir bloques con protección de know-how....3050
9.1.7.6 Cambiar la contraseña....3052
9.1.7.7 Quitar la protección de know-how para los bloques....3052

9.2 Mostrar información del programa....3053
9.2.1 Resumen de la información del programa disponible....3053
9.2.2 Visualizar el plano de ocupación....3054
9.2.2.1 Introducción al plano de ocupación....3054
9.2.2.2 Estructura del plano de ocupación....3055
9.2.2.3 Símbolos del plano de ocupación....3056
9.2.2.4 Visualizar el plano de ocupación....3057
9.2.2.5 Configurar las opciones de visualización del plano de ocupación....3058
9.2.2.6 Posibilidades de filtrado en el plano de ocupación....3059

9.2.2.7 Definir filtros para el plano de ocupación....3060
9.2.2.8 Filtrar el plano de ocupación....3061
9.2.2.9 Definir áreas de memoria remanentes para marcas....3062
9.2.2.10 Activar la indicación de la remanencia de marcas....3062
9.2.3 Mostrar la estructura de llamada....3063
9.2.3.1 Introducción a la estructura de llamadas....3063
9.2.3.2 Símbolos utilizados en la estructura de llamadas....3065
9.2.3.3 Composición de la estructura de llamadas....3066
9.2.3.4 Mostrar la estructura de llamadas....3067
9.2.3.5 Configurar las opciones de visualización de la estructura de llamadas....3068
9.2.3.6 Introducción a la comprobación de coherencia en la estructura de llamada....3069
9.2.3.7 Comprobar la coherencia de bloques en la estructura de llamada....3069
9.2.4 Mostrar la estructura de dependencias....3070
9.2.4.1 Introducción a la estructura de dependencias....3070
9.2.4.2 Estructura de dependencias....3071
9.2.4.3 Símbolos utilizados en la estructura de dependencias....3072
9.2.4.4 Mostrar la estructura de dependencias....3073
9.2.4.5 Configurar las opciones de visualización de la estructura de dependencias....3073
9.2.4.6 Introducción a la comprobación de coherencia en la estructura de dependencias....3074
9.2.4.7 Comprobar la coherencia de bloques en la estructura de dependencias....3075
9.2.5 Mostrar la carga de la memoria de la CPU....3076
9.2.5.1 Introducción a la carga de la memoria....3076
9.2.5.2 Estructura de la ficha "Carga de la memoria"....3078
9.2.5.3 Mostrar la carga de la memoria....3079
9.2.5.4 Seleccionar la memoria de carga máxima disponible....3080

9.3 Mostrar referencias cruzadas....3080
9.3.1 Información general sobre las referencias cruzadas....3080
9.3.2 Estructura de la lista de referencias cruzadas....3081
9.3.3 Visualizar la lista de referencias cruzadas....3083
9.3.4 Visualizar las referencias cruzadas en la ventana de inspección....3084

9.4 Comprobar el programa de usuario....3085
9.4.1 Principios básicos del test del programa de usuario....3085
9.4.2 Test con el estado del programa....3086
9.4.2.1 Introducción al test con el estado del programa....3086
9.4.2.2 Ajustar el entorno de llamada....3087
9.4.2.3 Test en modo de etapa individual....3089
9.4.2.4 Activar/desactivar el test con el estado del programa....3097
9.4.2.5 Editar bloques durante el test del programa....3099
9.4.2.6 Forzar variables en el estado del programa....3100
9.4.2.7 Cambiar formatos de visualización en el estado del programa....3100
9.4.2.8 Particularidades del test de programas GRAPH....3101
9.4.2.9 Ejemplos de visualización del estado del programa....3108
9.4.3 Test con la tabla de observación....3115
9.4.3.1 Introducción al test con la tabla de observación....3115
9.4.3.2 Estructura de la tabla de observación....3116
9.4.3.3 Modo básico/avanzado en la tabla de observación....3117
9.4.3.4 Símbolos y botones utilizados en la tabla de observación....3118
9.4.3.5 Crear y editar tablas de observación....3119
9.4.3.6 Introducir variables en la tabla de observación....3122
9.4.3.7 Observar variables en la tabla de observación....3130
9.4.3.8 Forzar variables en la tabla de observación....3135

9.4.4 Test con la tabla de forzado permanente....3143
9.4.4.1 Introducción al test con la tabla de forzado permanente....3143
9.4.4.2 Medidas de seguridad para el forzado permanente de variables....3144
9.4.4.3 Estructura de la tabla de forzado permanente....3145
9.4.4.4 Modo básico y modo avanzado en la tabla de forzado permanente....3146
9.4.4.5 Símbolos utilizados en la tabla de forzado permanente....3147
9.4.4.6 Abrir y procesar tabla de forzado permanente....3148
9.4.4.7 Introducir variables en la tabla de forzado permanente....3149
9.4.4.8 Observar variables en la tabla de forzado permanente....3157
9.4.4.9 Forzar permanentemente variables en la tabla de forzado permanente....3160
9.4.4.10 Finalizar forzado permanente de variables....3169

9.5 Configurar avisos....3171
9.5.1 Introducción a la configuración de avisos....3171
9.5.2 Asignación de números de aviso....3172
9.5.3 Componentes de un aviso....3172
9.5.4 Bloques de aviso disponibles....3173
9.5.5 Plantilla y avisos....3174
9.5.6 Parámetros formales, tipos de datos de avisos y bloques de aviso....3176
9.5.7 Estructura del editor de avisos....3177
9.5.8 Crear y editar avisos....3178
9.5.8.1 Crear avisos de programa....3178
9.5.8.2 Editar avisos de programa en el editor de avisos....3179
9.5.8.3 Editar avisos de programa en el editor de programación....3179
9.5.8.4 Borrar avisos de programa....3180
9.5.8.5 Crear y editar la plantilla....3180
9.5.8.6 Crear y editar DBs de instancia....3180
9.5.8.7 Crear avisos de diagnóstico de usuario....3181
9.5.8.8 Editar avisos de diagnóstico de usuario....3182
9.5.8.9 Borrar avisos de diagnóstico de usuario....3182
9.5.8.10 Insertar valores asociados en los avisos....3182
9.5.8.11 Estructura de los valores asociados....3183
9.5.8.12 Ejemplos de valores asociados....3184
9.5.8.13 Borrar valores asociados....3185
9.5.9 Crear y editar avisos....3186
9.5.9.1 Crear avisos de programa....3186
9.5.9.2 Editar avisos de programa en el editor de avisos....3186
9.5.9.3 Editar avisos de programa en el editor de programación....3187
9.5.9.4 Borrar avisos de programa....3188
9.5.9.5 Crear y editar la plantilla....3188
9.5.9.6 Crear y editar DBs de instancia....3188
9.5.9.7 Insertar valores asociados en los avisos....3189
9.5.9.8 Ajustes al introducir valores asociados....3190
9.5.9.9 Ejemplos de valores asociados....3191
9.5.9.10 Borrar valores asociados....3192
9.5.10 Textos y atributos....3192
9.5.10.1 Palabras clave para textos de aviso....3192
9.5.10.2 Introducir textos....3193
9.5.10.3 Inhibir textos....3193
9.5.10.4 Inhibir atributos....3194
9.5.11 Listas de textos para avisos....3195
9.5.11.1 Principios básicos de las listas de textos para avisos....3195
9.5.11.2 Editar listas de textos para avisos....3195

9.5.11.3 Integrar en avisos los textos de listas de textos....3196
9.5.11.4 Ejemplo de integración en avisos de los textos de las listas de textos....3196
9.5.12 Categorías....3197
9.5.12.1 Crear categorías....3197
9.5.12.2 Editar categorías....3198

9.6 Uso del Team Engineering....3198
9.6.1 Intercambio de datos con Inter Project Engineering (IPE)....3198
9.6.1.1 Principios básicos de la ingeniería interproyecto (IPE)....3198
9.6.1.2 Requisitos para la ingeniería interproyecto (IPE)....3200
9.6.1.3 Procedimientos al trabajar con ingeniería interproyecto (IPE)....3201
9.6.1.4 Creación de los datos del proxy del dispositivo....3204
9.6.1.5 Abrir datos del proxy del dispositivo....3205
9.6.1.6 Intercambio de datos de control mediante un archivo IPE....3205
9.6.1.7 Intercambio de datos de control mediante un archivo de proyecto....3206

9.7 Ejemplos de programación....3207
9.7.1 Ejemplos de programación (S7-1200, S7-1500)....3207
9.7.1.1 Ejemplos de programación KOP....3207
9.7.1.2 Ejemplos de programación FUP....3216
9.7.1.3 Ejemplos de programación AWL....3224
9.7.1.4 Ejemplos de programación SCL....3236
9.7.2 Ejemplos de programación (S7-300, S7-400)....3241
9.7.2.1 Ejemplos de programación KOP....3241
9.7.2.2 Ejemplos de programación FUP....3251
9.7.2.3 Ejemplos de programación AWL....3260
9.7.2.4 Ejemplos de programación SCL....3272

9.8 Referencias....3277
9.8.1 Referencias (S7-1200, S7-1500)....3277
9.8.1.1 Parámetros generales de las instrucciones....3277
9.8.1.2 Instrucciones básicas....3283
9.8.1.3 Instrucciones avanzadas....4690
9.8.1.4 Tecnología....5019
9.8.1.5 Comunicación....5244
9.8.2 Referencias (S7-300, S7-400)....5595
9.8.2.1 Parámetros generales de las instrucciones....5595
9.8.2.2 Instrucciones básicas....5602
9.8.2.3 Instrucciones avanzadas....6683
9.8.2.4 Tecnología....7078
9.8.2.5 Comunicación....7661

10 Visualización de procesos....8309

10.1 Crear imágenes....8309
10.1.1 Principios básicos....8309
10.1.1.1 Principios básicos de las imágenes 8309
10.1.1.2 Volumen de funciones de las imágenes en función del dispositivo....8311
10.1.1.3 Principios básicos....8312
10.1.1.4 Trabajar con imágenes....8315
10.1.1.5 Trabajar con plantillas....8318
10.1.1.6 Trabajar con estilos....8324
10.1.2 Trabajar con objetos....8327
10.1.2.1 Sinopsis de los objetos....8327
10.1.2.2 Posibilidades de edición de objetos....8330

10.1.2.3 Insertar un objeto....8331
10.1.2.4 Borrar un objeto....8333
10.1.2.5 Posicionar un objeto....8334
10.1.2.6 Modificar el tamaño de un objeto....8335
10.1.2.7 Seleccionar varios objetos....8337
10.1.2.8 Alinear objetos....8339
10.1.2.9 Desplazar un objeto hacia adelante o hacia atrás....8341
10.1.2.10 Mostrar objetos fuera del área de la imagen....8342
10.1.2.11 Girar un objeto....8342
10.1.2.12 Voltear un objeto....8344
10.1.2.13 Diseñar un objeto....8345
10.1.2.14 Diseñar patrón de relleno....8345
10.1.2.15 Formatear los gráficos en un objeto....8348
10.1.2.16 Formatear los textos dentro de un objeto....8349
10.1.2.17 Diseño de objetos basados en tablas....8350
10.1.2.18 Diseño de bordes....8351
10.1.2.19 Definición del gradiente de color....8353
10.1.2.20 Aplicación de estilos predefinidos....8354
10.1.2.21 Insertar varios objetos del mismo tipo (sellar)....8355
10.1.2.22 Modificar la posición y el tamaño de varios objetos....8357
10.1.2.23 Gráficos externos....8358
10.1.2.24 Administrar gráficos externos....8359
10.1.2.25 Agregar un gráfico externo a la colección de gráficos....8360
10.1.2.26 Trabajar con grupos de objetos....8363
10.1.2.27 Configuración del acceso con el teclado....8370
10.1.2.28 Ejemplos....8373
10.1.3 Trabajar con listas de textos y gráficos....8376
10.1.3.1 Trabajar con listas de textos....8376
10.1.3.2 Trabajar con listas de gráficos....8384
10.1.4 Dinamización de imágenes....8393
10.1.4.1 Principios básicos de la dinamización....8393
10.1.4.2 Dinamización en la ventana de inspección....8393
10.1.4.3 Dinamizar con animaciones....8395
10.1.4.4 Dinamizar con funciones de sistema....8404
10.1.5 Trabajar con teclas de función....8406
10.1.5.1 Trabajar con teclas de función....8406
10.1.5.2 Asignar teclas de función globalmente....8408
10.1.5.3 Asignar teclas de función localmente....8410
10.1.5.4 Asignar una función a una tecla de función....8411
10.1.5.5 Autorizar el uso de una tecla de función....8412
10.1.5.6 Asignar un gráfico a una tecla de función....8414
10.1.5.7 Configurar variables LED....8416
10.1.5.8 Ejemplo: Utilizar teclas de función para navegar por imágenes....8418
10.1.6 Trabajar con niveles....8419
10.1.6.1 Principios básicos para trabajar con niveles....8419
10.1.6.2 Desplazar objetos a niveles....8420
10.1.6.3 Configurar el nivel activo....8421
10.1.6.4 Mostrar y ocultar niveles....8422
10.1.6.5 Cambiar el nombre de los niveles....8423
10.1.7 Objetos de manejo y visualización....8425
10.1.7.1 Objetos dependientes de los paneles....8425
10.1.7.2 Objetos....8426

10.1.8 Configurar la navegación de imágenes....8456
10.1.8.1 Principios básicos de la navegación de imágenes....8456
10.1.8.2 Asignar un cambio de imagen a un botón....8457
10.1.8.3 Asignar un cambio de imagen a una tecla de función....8458

10.2 Trabajar con variables....8459
10.2.1 Principios básicos....8459
10.2.1.1 Principios básicos de las variables....8459
10.2.1.2 Resumen de las tablas de variables HMI....8460
10.2.1.3 Variables externas....8461
10.2.1.4 Direccionamiento de variables externas....8463
10.2.1.5 Variables internas....8466
10.2.2 Trabajar con variables....8467
10.2.2.1 Crear variables....8467
10.2.2.2 Edición de variables....8471
10.2.2.3 Configurar variables....8477
10.2.3 Trabajar con matrices....8496
10.2.3.1 Principios básicos de las matrices....8496
10.2.3.2 Crear una variable de matriz....8498
10.2.3.3 Ejemplos de matrices....8499
10.2.4 Trabajar con ciclos....8500
10.2.4.1 Principios básicos de los ciclos....8500
10.2.5 Representar variables....8501
10.2.5.1 Emitir valores de variables en imágenes....8501
10.2.5.2 Configurar el visor de curvas para los valores del controlador....8501

10.3 Trabajar con avisos....8503
10.3.1 Principios básicos....8503
10.3.1.1 El sistema de avisos de WinCC....8503
10.3.1.2 Procedimientos de aviso....8505
10.3.1.3 Estados de aviso 8506
10.3.1.4 Categorías....8508
10.3.1.5 Acuse....8509
10.3.1.6 Grupos de avisos....8511
10.3.1.7 Número de aviso....8512
10.3.2 Trabajar con avisos....8512
10.3.2.1 Componentes y propiedades de los avisos....8512
10.3.2.2 Configuración de avisos....8514
10.3.2.3 Configurar la salida de avisos....8527
10.3.2.4 Acuse de avisos....8532
10.3.3 Manejar avisos en runtime....8535
10.3.3.1 Avisos en runtime....8535
10.3.3.2 Visor de avisos simple, ventana de avisos simple en runtime 8537
10.3.3.3 Indicador de avisos en runtime....8539
10.3.3.4 Acusar avisos....8539
10.3.4 Referencia....8540
10.3.4.1 Funciones de sistema para avisos....8540
10.3.4.2 Avisos de sistema....8541
10.3.5 Configurar diagnóstico de sistema....8565
10.3.5.1 Principios básicos del diagnóstico de sistema....8565
10.3.5.2 Vistas en el diagnóstico de sistema....8566
10.3.5.3 Configurar el visor de diagnóstico del sistema....8568

10.4 Trabajar con recetas....8569

10.4.1 Principios básicos....8569
10.4.1.1 Definición y campos de aplicación....8569
10.4.1.2 Ejemplos de utilización de recetas....8571
10.4.1.3 Estructura de las recetas....8571
10.4.1.4 Visualización de las recetas....8573
10.4.1.5 Flujo de datos en las recetas....8573
10.4.1.6 Sincronizar registros de receta con el controlador....8574
10.4.2 Elementos y configuración básica....8576
10.4.2.1 Editor "Recetas"....8576
10.4.3 Visualización y edición de recetas en runtime....8579
10.4.3.1 Visor de recetas simple....8579
10.4.3.2 Comportamiento del visor de recetas en runtime....8580
10.4.4 Configurar recetas....8581
10.4.4.1 Procedimiento general de configuración....8581
10.4.4.2 Crear y editar recetas....8582
10.4.4.3 Configurar la visualización de la receta....8590
10.4.5 Manejar recetas en runtime....8593
10.4.5.1 Manejar el visor de recetas simple....8593
10.4.6 Ejemplo....8599
10.4.6.1 Ejemplo de creación de una receta....8599

10.5 Configurar una administración de usuarios....8601
10.5.1 Campo de aplicación de la administración de usuarios....8601
10.5.2 Estructura de la administración de usuarios....8601
10.5.3 Principios básicos....8603
10.5.3.1 Usuarios....8603
10.5.3.2 Área de trabajo de usuarios....8603
10.5.3.3 Grupos de usuarios....8604
10.5.3.4 Área de trabajo de grupos de usuarios....8605
10.5.3.5 Configuración de la administración de usuarios....8606
10.5.4 Estructurar la administración de usuarios....8608
10.5.4.1 Principios básicos de la administración de usuarios....8608
10.5.4.2 Administrar usuarios para runtime....8609
10.5.4.3 Administrar usuarios en runtime....8616
10.5.4.4 Configurar una protección de acceso....8625
10.5.5 Referencia....8626
10.5.5.1 Objetos con protección de acceso....8626
10.5.5.2 Autorizaciones y grupos de usuarios predefinidos....8626
10.5.6 Ejemplos....8627
10.5.6.1 Ejemplo: Configurar un botón con el cuadro de diálogo de inicio de sesión....8627
10.5.6.2 Ejemplo: Registrar el inicio y el cierre de sesión....8628
10.5.6.3 Ejemplo de una administración de usuarios....8629

10.6 Trabajar con funciones de sistema....8636
10.6.1 Principios básicos....8636
10.6.1.1 Funciones de sistema....8636
10.6.1.2 Utilizar funciones de sistema....8638
10.6.2 Trabajar con listas de funciones....8638
10.6.2.1 Principios básicos de la lista de funciones....8638
10.6.2.2 Propiedades de la lista de funciones....8639
10.6.2.3 Configurar la lista de funciones....8640
10.6.2.4 Editar la lista de funciones....8642
10.6.2.5 Procesar la lista de funciones en runtime....8643

10.6.3 Ejemplo....8643
10.6.3.1 Cambiar el modo de operación del panel de operador y visualizar el estado actual....8643
10.6.4 Referencia....8646
10.6.4.1 Lista de funciones....8646
10.6.4.2 Eventos....8711

10.7 Planificar tareas....8728
10.7.1 Campo de aplicación del planificador de tareas....8728
10.7.2 Trabajar con tareas y disparadores....8729
10.7.3 Principios básicos....8730
10.7.3.1 Área de trabajo del editor Planificador de tareas....8730
10.7.3.2 Lista de funciones....8731
10.7.3.3 Disparador....8732
10.7.3.4 Planificar una tarea con el disparo por eventos....8732
10.7.4 Ejemplos....8734
10.7.4.1 Ejemplo: Actualizar el usuario al cambiar de usuario....8734

10.8 Comunicación con controladores....8736
10.8.1 Principios básicos de la comunicación....8736
10.8.1.1 Comunicación entre dispositivos....8736
10.8.1.2 Dispositivos y redes en el sistema de automatización....8737
10.8.1.3 Intercambio de datos mediante variables....8742
10.8.1.4 Intercambio de datos mediante punteros de área....8743
10.8.1.5 Driver de comunicación....8743
10.8.2 Editores para la comunicación....8744
10.8.2.1 Editor "Dispositivos y redes"....8744
10.8.2.2 Vista de redes....8745
10.8.2.3 Datos de red....8748
10.8.2.4 Diagnóstico de conexiones online....8750
10.8.2.5 Vista de dispositivos....8752
10.8.2.6 Vista topológica....8754
10.8.2.7 Ventana de inspección....8756
10.8.2.8 Catálogo de hardware....8758
10.8.2.9 Información sobre los componentes de hardware....8760
10.8.3 Redes y conexiones....8762
10.8.3.1 Redes de comunicación SIMATIC....8762
10.8.3.2 Configurar redes y conexiones....8766
10.8.4 Intercambio de datos....8774
10.8.4.1 Intercambio de datos mediante variables....8774
10.8.4.2 Intercambio de datos mediante punteros de área....8781
10.8.5 Dependencia del panel....8788
10.8.5.1 Basic Panel....8788
10.8.6 Comunicación con SIMATIC S7 1500....8794
10.8.6.1 Comunicación con SIMATIC S7 1500....8794
10.8.6.2 Comunicación vía PROFINET....8795
10.8.6.3 Comunicación vía PROFIBUS....8808
10.8.6.4 Intercambio de datos....8817
10.8.6.5 Características de rendimiento de la comunicación....8836
10.8.6.6 Configurar conexiones en el editor "Conexiones"....8838
10.8.7 Comunicación con SIMATIC S7 1200....8843
10.8.7.1 Comunicación con SIMATIC S7 1200....8843
10.8.7.2 Comunicación vía PROFINET....8844
10.8.7.3 Comunicación vía PROFIBUS....8859

10.8.7.4 Intercambio de datos 8868
10.8.7.5 Características de rendimiento de la comunicación....8888
10.8.7.6 Crear conexiones en el editor "Conexiones"....8890
10.8.8 Comunicación con SIMATIC S7 300/400....8898
10.8.8.1 Comunicación con SIMATIC S7 300/400....8898
10.8.8.2 Comunicación vía PROFINET....8899
10.8.8.3 Comunicación vía PROFIBUS....8912
10.8.8.4 Comunicación vía MPI....8920
10.8.8.5 Intercambio de datos....8928
10.8.8.6 Características de rendimiento de la comunicación....8946
10.8.8.7 Crear conexiones en el editor "Conexiones"....8946
10.8.9 Comunicación con SIMATIC S7 200....8959
10.8.9.1 Comunicación con SIMATIC S7 200....8959
10.8.9.2 Crear una conexión con SIMATIC S7 200....8959
10.8.9.3 Parámetros de la conexión....8961
10.8.9.4 Intercambio de datos 8969
10.8.9.5 Características de rendimiento de la comunicación....8987
10.8.10 Comunicación con SIMATIC LOGO!....8988
10.8.10.1 Comunicación con SIMATIC LOGO!....8988
10.8.10.2 Crear una conexión con SIMATIC LOGO!....8988
10.8.10.3 Parámetros de la conexión....8990
10.8.10.4 Intercambio de datos....8994
10.8.10.5 Características de rendimiento de la comunicación....8998
10.8.11 Comunicación con otros controladores....8999
10.8.11.1 Comunicación con otros controladores....8999
10.8.11.2 Particularidades de la configuración....9000
10.8.11.3 Drivers de comunicación....9000
10.8.11.4 Intercambio de datos mediante punteros de área....9118

10.9 Utilizar funciones extendidas....9135
10.9.1 Conceptos básicos del asistente de paneles de operador....9135
10.9.2 Trabajar con librerías....9136
10.9.2.1 Principios básicos de las librerías....9136
10.9.2.2 Sinopsis de la vista de librerías....9137
10.9.2.3 Plantillas maestras y tipos....9139
10.9.2.4 Librerías en WinCC....9140
10.9.2.5 Administrar librerías....9142
10.9.2.6 Administrar objetos en una librería....9149
10.9.2.7 Utilizar tipos y sus versiones....9154
10.9.3 Utilizar datos de control de otros proyectos....9158
10.9.3.1 Principios básicos de la ingeniería interproyecto (IPE)....9158
10.9.3.2 Datos de control....9159
10.9.3.3 Inicializar proxy de dispositivos mediante archivo IPE....9159
10.9.3.4 Inicializar proxy de dispositivos mediante archivo de proyecto....9165
10.9.4 Importar y exportar datos de proyecto....9170
10.9.4.1 Importación y exportación de datos de proyecto....9170
10.9.4.2 Importar y exportar recetas....9171
10.9.4.3 Importar y exportar avisos....9175
10.9.4.4 Importar y exportar variables....9182
10.9.4.5 Importar y exportar listas de textos....9188
10.9.4.6 Exportar e importar textos del proyecto....9192
10.9.5 Utilizar referencias cruzadas....9195
10.9.5.1 Información general sobre referencias cruzadas....9195

10.9.5.2 Mostrar la lista de referencias cruzadas....9195
10.9.5.3 Estructura de la lista de referencias cruzadas....9196
10.9.5.4 Mostrar referencias cruzadas en la ventana de inspección....9197
10.9.5.5 Reasignar variables en imágenes....9198
10.9.6 Gestionar idiomas....9201
10.9.6.1 Conceptos lingüísticos en WinCC....9201
10.9.6.2 Configuración de idiomas en el sistema operativo....9202
10.9.6.3 Configuración en el sistema operativo para los idiomas asiáticos....9203
10.9.6.4 Ajustar idiomas del proyecto....9204
10.9.6.5 Crear un proyecto en varios idiomas....9207
10.9.6.6 Utilizar gráficos específicos del idioma....9215
10.9.6.7 Idiomas de runtime....9219
10.9.6.8 Ejemplo de un proyecto multilingüe....9226
10.9.7 Sustituir dispositivo....9229
10.9.7.1 Principios básicos....9229
10.9.7.2 Funciones dependientes del panel....9230
10.9.7.3 Adaptar imágenes al nuevo dispositivo....9233
10.9.8 Copiar entre paneles de operador y editores....9239
10.9.8.1 Principios básicos....9239
10.9.8.2 Copiar y pegar....9241
10.9.8.3 Copia entre distinas versiones de RT y ES....9245
10.9.9 Utilizar la compatibilidad de versiones de WinCC....9246
10.9.9.1 Principios básicos de la compatibilidad de versiones....9246
10.9.9.2 Edición de proyectos de una versión anterior de WinCC....9248
10.9.9.3 Actualización del proyecto....9249
10.9.9.4 Actualizar una librería global....9250
10.9.9.5 Cambio entre distintas versiones de dispositivos....9251
10.9.9.6 Cambio de versión del dispositivo....9252
10.9.10 Mostrar datos de una tarjeta de memoria....9253
10.9.10.1 Principios básicos....9253
10.9.10.2 Trabajo con copias de seguridad....9254

10.10 Compilar y cargar....9256
10.10.1 Compilar y cargar proyectos....9256
10.10.1.1 Resumen de cómo compilar y cargar proyectos....9256
10.10.1.2 Compilar un proyecto....9258
10.10.1.3 Cargar proyectos....9259
10.10.1.4 Inicio de runtime....9263
10.10.2 Simular proyectos....9264
10.10.2.1 Principios básicos de la simulación....9264
10.10.2.2 Simular un proyecto....9266
10.10.2.3 Trabajar con el simulador de variables....9268
10.10.2.4 Restricciones en la simulación....9270
10.10.3 Mantenimiento del panel de operador....9270
10.10.3.1 Resumen del mantenimiento del panel de operador (Basic Panels)....9270
10.10.3.2 ProSave....9271
10.10.3.3 Crear una copia de seguridad de los datos del panel de operador....9272
10.10.3.4 Guardar y restablecer los datos del panel de operador....9273
10.10.3.5 Actualizar el sistema operativo....9274
10.10.3.6 Actualizar el sistema operativo del panel de operador....9275
10.10.4 Referencia....9277
10.10.4.1 Mensajes de error al cargar proyectos....9277
10.10.4.2 Adaptar el proyecto a otro panel de operador....9278

10.10.4.3 Establecer una conexión con el panel de operador....9280

10.11 Manejar en runtime....9281
10.11.1 Principios básicos....9281
10.11.1.1 Visión de conjunto....9281
10.11.1.2 Variables en Runtime....9282
10.11.1.3 Funciones de sistema en runtime....9282
10.11.2 Poner en marcha proyectos....9283
10.11.2.1 Configuración del software runtime....9283
10.11.2.2 Cargar un proyecto....9284
10.11.2.3 Iniciar runtime en la estación de ingeniería....9285
10.11.2.4 Iniciar el runtime en el panel de operador....9286
10.11.2.5 Comprobar el proyecto....9287
10.11.2.6 Salir del proyecto....9288
10.11.2.7 Guardar y restaurar los datos del panel de operador....9288
10.11.3 Idiomas de runtime....9289
10.11.3.1 Idiomas de runtime....9289
10.11.3.2 Ajustar idioma de runtime....9290
10.11.3.3 Ajustar fuente para un idioma de runtime....9291
10.11.3.4 Configurar cambio de idioma....9292
10.11.3.5 Particularidades de los idiomas asiáticos y orientales en runtime....9293
10.11.4 Manejar el proyecto....9294
10.11.4.1 Principios básicos....9294
10.11.4.2 Manejar objetos....9301
10.11.4.3 Seguridad en el proyecto....9314
10.11.4.4 Manejar avisos....9322
10.11.4.5 Manejar recetas....9330

10.12 Características....9341
10.12.1 Engineering System....9341
10.12.2 Basic Panel....9342
10.12.3 Basic Panel de 2.ª generación....9346
10.12.4 Datos técnicos generales....9350
10.12.4.1 Caracteres utilizables....9350
10.12.4.2 Impresoras recomendadas....9350
10.12.4.3 Memoria necesaria para recetas en Basic Panels....9351

10.13 Cambio a WinCC en el TIA Portal....9352
10.13.1 Resumen de la migración a WinCC V12....9352
10.13.2 Librerías....9353
10.13.3 Imágenes y plantillas....9355

11 Utilizar funciones tecnológicas....9359

11.1 Contaje, medición y lectura de posición....9359
11.1.1 Contaje, medición y lectura de posición (S7-1500)....9359
11.1.1.1 Principios básicos de contaje, medición y lectura de posición....9359
11.1.1.2 Utilización de High_Speed_Counter....9408
11.1.1.3 Utilización del módulo tecnológico....9430
11.1.2 Contaje y medición (S7-300, S7-400)....9467
11.1.2.1 Contaje y medición....9467

11.2 Leva electrónica (S7-300, S7-400)....9738
11.2.1 Control por levas....9738
11.2.1.1 Conceptos básicos del control por levas....9738

11.2.1.2 Funciones de FM x52....9752
11.2.1.3 Uso del FM 352....9757
11.2.1.4 Uso del FM 452....9834

11.3 Regulación PID....9915
11.3.1 Principios básicos de la regulación....9915
11.3.1.1 Lazo de regulación y elementos finales de control....9915
11.3.1.2 Sistemas regulados....9916
11.3.1.3 Valores característicos del sistema regulado....9919
11.3.1.4 Regulador de impulsos....9921
11.3.1.5 Respuesta de regulación y a perturbaciones....9925
11.3.1.6 Respuesta de regulación con diferentes estructuras de realimentación....9926
11.3.1.7 Elección de la estructura del regulador con un sistema regulado definido....9934
11.3.1.8 Ajuste de los parámetros PID....9935
11.3.2 Configuración de un regulador por software....9935
11.3.2.1 Vista general de los reguladores de software....9935
11.3.2.2 Pasos para la configuración de un regulador por software....9937
11.3.2.3 Agregar objetos tecnológicos....9937
11.3.2.4 Configurar objetos tecnológicos....9938
11.3.2.5 Llamar la instrucción en el programa de usuario....9939
11.3.2.6 Cargar objetos tecnológicos en el dispositivo....9940
11.3.2.7 Poner en marcha el regulador de software....9942
11.3.2.8 Guardar los parámetros PID optimizados en el proyecto....9942
11.3.2.9 Comparar valores....9943
11.3.2.10 Vista de parámetros....9946
11.3.2.11 Mostrar el DB de instancia de un objeto tecnológico....9964
11.3.3 Regulación PID (S7-1200, S7-1500)....9965
11.3.3.1 Utilizar PID_Compact....9965
11.3.3.2 Utilizar PID_3Step....9997
11.3.4 Regulación PID (S7-300, S7-400, S7-1500)....10032
11.3.4.1 Utilizar funciones básicas PID....10032
11.3.5 Regulación PID (S7-300, S7-400)....10065
11.3.5.1 Utilizar control PID estándar....10065
11.3.5.2 Utilizar Modular PID Control....10167
11.3.5.3 Utilizar PID Self Tuner....10200
11.3.5.4 Fundamentos de los módulos de regulación....10216
11.3.5.5 Uso del FM 355....10255
11.3.5.6 Utilización del FM 355-2....10289
11.3.5.7 Utilizar el FM 455....10340
11.3.5.8 Utilizar ET 200S 2PULSE....10375

11.4 Control de movimiento....10431
11.4.1 Control de movimiento (S7-1200, S7-1500)....10431
11.4.1.1 Utilizar S7-1200 Motion Control....10431
11.4.1.2 Utilizar S7-1500 Motion Control....10640
11.4.2 Control de movimiento (S7-300, S7-400)....10855
11.4.2.1 Principios básicos para la lectura del recorrido....10855
11.4.2.2 Utilizar SM 338....10858
11.4.2.3 Utilizar ET 200S 1SSI....10868
11.4.2.4 Principios básicos del posicionamiento con módulos de función....10891
11.4.2.5 Uso del FM 351....10899
11.4.2.6 Uso del FM 451....10993
11.4.2.7 Uso del ET 200S 1PosU....11094

11.4.2.8 Utilizar ET 200S 1STEP 5V....11162
11.4.2.9 Uso del IM 174....11199
11.4.2.10 Uso de Easy Motion Control....11215

12 Uso del Team Engineering....11283

12.1 Poner en marcha proyectos en equipo....11283
12.1.1 Principios básicos de la puesta en marcha común....11283
12.1.2 Requisitos para la puesta en marcha común....11285
12.1.3 Procedimiento para la puesta en marcha en equipo....11287
12.1.4 Reglas para la puesta en marcha común....11290

12.2 Intercambio de datos con ingeniería interproyecto (Inter Project Engineering) (IPE)....11294
12.2.1 Principios básicos de la ingeniería interproyecto (IPE)....11294
12.2.2 Requisitos para la ingeniería interproyecto (IPE)....11297
12.2.3 Procedimientos al trabajar con ingeniería interproyecto (IPE)....11298

13 Uso de las funciones Online y diagnóstico....11301

13.1 Mostrar nodos accesibles....11301

13.2 Modificación online de la configuración del dispositivo....11302

13.3 Conectar dispositivos online....11303
13.3.1 Información general sobre el modo online....11303
13.3.2 Vista en el modo online....11305
13.3.3 Establecer y deshacer una conexión online....11306
13.3.4 Conexión online con varios dispositivos....11308
13.3.5 Deshacer conexiones online de varios dispositivos....11309

13.4 Crear una copia de seguridad del software y la configuración de hardware de un dispositivo 11309
13.4.1 Crear una copia de seguridad de un dispositivo....11309
13.4.2 Copia de seguridad de la configuración de un dispositivo....11310
13.4.3 Restablecer el software y la configuración de hardware de un dispositivo....11311

13.5 Configurar interfaces PG/PC....11312
13.5.1 Accesos online....11312
13.5.2 Principios básicos para parametrizar la interfaz PG/PC....11315
13.5.3 Mostrar u ocultar interfaces....11315
13.5.4 Mostrar y modificar las propiedades de una interfaz....11316
13.5.5 Agregar interfaces....11316
13.5.6 Parametrizar la interfaz Ethernet....11317
13.5.6.1 Parametrizar la interfaz Industrial Ethernet....11317
13.5.6.2 Visualizar los parámetros del sistema operativo....11318
13.5.6.3 Conectar la interfaz PG/PC con una subred....11318
13.5.6.4 Parametrizar la interfaz Ethernet....11319
13.5.6.5 Asignar una dirección IP temporal....11320
13.5.6.6 Gestionar direcciones IP temporales....11320
13.5.6.7 Restablecer la configuración TCP/IP....11321
13.5.7 Parametrizar la interfaz MPI y PROFIBUS....11321
13.5.7.1 Parametrizar la interfaz MPI y PROFIBUS....11321
13.5.7.2 Parametrizar la interfaz MPI o PROFIBUS automáticamente....11322
13.5.7.3 Parametrizar la interfaz MPI....11323
13.5.7.4 Parametrizar la interfaz PROFIBUS....11325
13.5.7.5 Resumen de los parámetros del bus para PROFIBUS....11327
13.5.7.6 Restablecer la configuración MPI o PROFIBUS....11329

13.6 Uso de la función Trace y de analizador lógico....11329
Prólogo....11329
13.6.1 Información de seguridad....11330
13.6.1 Descripción....11331
13.6.1.1 Hardware admitido....11331
13.6.1.2 Registro de medidas con la función Trace....11331
13.6.1.3 Configuración de Trace, registro y medición....11332
13.6.1.4 Almacenamiento de datos....11333
13.6.2 Interfaz de software....11334
13.6.2.1 Navegador de proyecto....11336
13.6.2.2 Área de trabajo....11336
13.6.2.3 Área específica de dispositivo....11345
13.6.3 Manejo....11345
13.6.3.1 Guía de inicio rápido....11345
13.6.3.2 Uso de la función Trace: vista general....11350
13.6.3.3 Llamada del editor de Trace....11351
13.6.3.4 Administración de Trace....11351
13.6.3.5 Tabla de señales....11355
13.6.3.6 Diagrama de curvas....11356
13.6.4 Dispositivos....11358
13.6.4.1 CPU S7-1200/1500....11358

13.7 Establecer una conexión remota con TeleService....11371
13.7.1 Principios básicos para el trabajo con TeleService....11371
13.7.1.1 Introducción a TeleService....11371
13.7.1.2 Funcionalidad de TeleService....11372
13.7.1.3 Listín telefónico en TeleService....11373
13.7.2 Trabajar con el listín telefónico....11374
13.7.2.1 Principios básicos para trabajar con el listín telefónico....11374
13.7.2.2 Configuración del listín telefónico....11374
13.7.2.3 Símbolos del listín telefónico....11375
13.7.2.4 Administrar el listín telefónico....11376
13.7.3 Conexión remota como conexión por línea telefónica....11382
13.7.3.1 Principios básicos para el establecimiento de una conexión por línea telefónica....11382
13.7.3.2 Redes telefónicas y módems....11383
13.7.3.3 Protección de acceso en las conexiones por línea telefónica....11386
13.7.3.4 TS Adapter MPI....11391
13.7.3.5 TS Adapter IE....11398
13.7.3.6 Establecimiento de una conexión por línea telefónica con una instalación remota....11404
13.7.4 Conexión remota como conexión VPN....11406
13.7.4.1 Principios básicos para el establecimiento de una conexión VPN....11406
13.7.4.2 Principios básicos de los certificados CA....11407
13.7.4.3 Instalar certificado CA para conexión VPN....11409
13.7.4.4 Borrar certificado CA para conexión VPN....11412
13.7.4.5 Establecimiento de una conexión VPN con una instalación remota....11412
13.7.4.6 TS Adapter IE Advanced....11414
13.7.5 Conexiones remotas controladas por CPU en TeleService....11418
13.7.5.1 Vista general de las conexiones remotas controladas por CPU....11418
13.7.5.2 Establecer conexiones desde y hacia instalaciones remotas (acoplamiento remoto PG-AS) 11419
13.7.5.3 Intercambio de datos entre instalaciones remotas (acoplamiento remoto AS-AS)....11421
13.7.5.4 Enviar un SMS desde una instalación....11423
13.7.5.5 Enviar un correo electrónico desde una instalación....11424

13.7.6 Indicaciones para la búsqueda de errores....11426
13.7.6.1 Indicaciones generales para la búsqueda de errores en caso de problemas con el módem 11426
13.7.6.2 Grabación de un archivo de informe para el módem....11427
13.7.6.3 No se establece una conexión por línea telefónica con el TS Adapter....11427
13.7.6.4 No se establece una conexión por línea telefónica desde el TS Adapter....11429
13.7.6.5 La conexión módem se interrumpe....11430
13.7.6.6 Lista de chequeo en caso de dificultades con el módem....11430
13.7.6.7 Mensajes de error del módem....11431
13.7.6.8 Posibles mensajes de error en las conexiones VPN....11432

14 Documentación del hardware....11433

14.1 Información general sobre la documentación del hardware....11433

14.2 PLC....11433
14.2.1 SIMATIC S7-1200....11433
14.2.1.1 CPU....11433
14.2.1.2 Signal Boards....11437
14.2.1.3 Módulos de comunicación....11438
14.2.1.4 Battery Boards....11439
14.2.1.5 Módulos de entradas digitales....11439
14.2.1.6 Módulos de salidas digitales....11440
14.2.1.7 Módulos de entradas y salidas digitales....11441
14.2.1.8 Módulos de entradas analógicas....11442
14.2.1.9 Módulos de salidas analógicas....11444
14.2.1.10 Módulos de entradas y salidas analógicas....11444
14.2.1.11 Módulos de comunicación....11445
14.2.1.12 Módulos tecnológicos....11447
14.2.2 SIMATIC S7-1500....11447
14.2.2.1 CPU....11447
14.2.2.2 Otros módulos para S7-1500....11449

14.3 Periferia descentralizada....11450
14.3.1 ET 200MP....11450
14.3.1.1 Módulos de interfaz....11450
14.3.1.2 Módulos de entradas digitales....11450
14.3.1.3 Módulos de salidas digitales....11451
14.3.1.4 Módulos de entradas y salidas digitales....11453
14.3.1.5 Módulos de entradas analógicas....11453
14.3.1.6 Módulos de salidas analógicas....11454
14.3.1.7 Módulos de entradas y salidas analógicas....11454
14.3.1.8 Módulos de comunicación....11455
14.3.1.9 Módulos de alimentación....11456
14.3.1.10 Módulos tecnológicos....11457
14.3.2 ET 200SP....11458
14.3.2.1 CPU....11458
14.3.2.2 Módulos de interfaz....11458
14.3.2.3 Módulos de entradas digitales....11459
14.3.2.4 Módulos de salidas digitales....11460
14.3.2.5 Módulos de entradas analógicas....11462
14.3.2.6 Módulos de salidas analógicas....11463
14.3.2.7 Módulos de comunicación....11464
14.3.2.8 Módulos de potencia....11464
14.3.2.9 Módulos especiales....11464

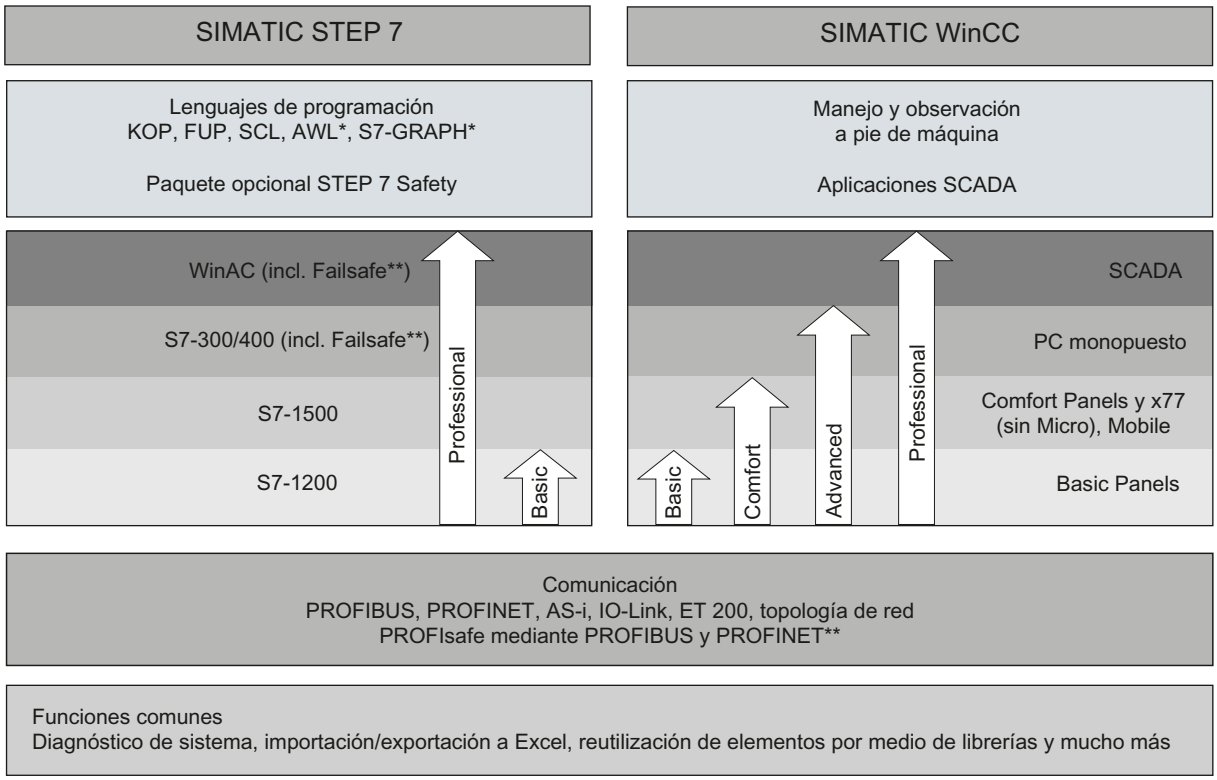
14.3.2.10 Módulos tecnológicos.....11465
Índice alfabético.....11467

Descripción del sistema STEP 7 y WinCC 1

1.1 Escalado de STEP 7 y WinCC en el TIA Portal

Volumen de prestaciones de los productos

El siguiente gráfico muestra el volumen de prestaciones de los distintos productos de STEP 7 y WinCC:



* Solo con STEP 7 Professional para S7-300/400/WinAC y S7-1500
** Con paquete opcional instalado "STEP 7 Safety Advanced"

STEP 7

STEP 7 (TIA Portal) es el software de ingeniería para configurar las familias de controladores S7-1200, S7-1500, S7-300/400 y WinAC. STEP 7 (TIA Portal) está disponible en dos ediciones distintas en función de las familias de controladores configurables:

- STEP 7 Basic para la configuración del S7-1200
- STEP 7 Professional para la configuración de S7-1200, S7-1500, S7-300/400 y WinAC

WinCC

WinCC (TIA Portal) es un software de ingeniería para configurar SIMATIC Panels, SIMATIC PC industriales y Standard PC con el software de visualización WinCC Runtime Advanced o el sistema SCADA WinCC Runtime Professional.

WinCC (TIA Portal) está disponible en cuatro ediciones distintas en función de los sistemas de operador configurables:

- WinCC Basic para la configuración de los Basic Panels
  WinCC Basic se incluye siempre en las ediciones STEP 7 Basic y STEP 7 Professional.
- WinCC Comfort para la configuración todos los Panels (incluidos los Comfort Panels, Mobile Panels)
- WinCC Advanced para la configuración de todos los Panels y PCs con el software de visualización WinCC Runtime Advanced
  WinCC Runtime Advanced es un software de visualización para sistemas monopuesto basados en PC. WinCC Runtime Advanced está disponible con licencias para PowerTags (variables con conexión al proceso) de 128, 512, 2k, 4k y 8k.
- WinCC Professional para configurar Panels así como PCs con WinCC Runtime Advanced o el sistema SCADA WinCC Runtime Professional WinCC Professional está disponible en las ediciones WinCC Professional para 512 y 4096 PowerTags, y como WinCC Professional para el máximo de PowerTags.
  WinCC Runtime Professional es un sistema SCADA para crear una configuración con sistemas monopuesto y sistemas multipuesto con clientes estándar o web. WinCC Runtime Professional está disponible con licencias para PowerTags (variables con conexión al proceso) de 128, 512, 2k, 4k, 8k y 64k.

WinCC (TIA Portal) permite configurar asimismo SINUMERIK PC con WinCC Runtime Advanced o bien WinCC Runtime Professional y paneles de operador con SINUMERIK HMI Pro sl RT o SINUMERIK Operate WinCC RT Basic.

1.2 Opciones para el sistema de ingeniería STEP 7

Otros productos para STEP 7

Para aplicaciones con requisitos de seguridad elevados, STEP 7 Professional puede complementarse con la opción STEP 7 Safety Advanced.

Si se utiliza la opción STEP 7 Safety Advanced, es posible configurar una periferia de seguridad y crear programas de seguridad para CPU F en KOP y FUP.

1.3 Opciones para sistemas de ingeniería y runtime WinCC

Los SIMATIC Panels, así como WinCC Runtime Advanced y WinCC Runtime Professional contienen todas las funciones esenciales para el manejo y visualización de máquinas e instalaciones. Para ampliar el ámbito de tareas, las funciones se pueden complementar en parte con opciones adicionales.

Opciones para Comfort Panels, Mobile Panels, Multi Panels

Para Comfort Panels, Mobile Panels y Multi Panels existen las siguientes posibilidades de ampliación:

- WinCC SmartServer (manejo remoto)
- WinCC Audit (Audit Trail y firma electrónica para aplicaciones reguladas)

Nota

En comparación con WinCC flexible 2008, en la funcionalidad básica se han incluido funciones de las opciones WinCC flexible /Sm@rtService, WinCC flexible /Sm@rtAccess, así como la opción WinCC flexible /OPC-Server.

Opciones para WinCC Runtime Advanced

Para WinCC Runtime Advanced existen las siguientes posibilidades de ampliación:

- WinCC SmartServer (manejo remoto)
- WinCC Recipes (sistema de recetas)
- WinCC Logging (archivación de valores de proceso y avisos)
- WinCC Audit (Audit Trail para aplicaciones reguladas)
- WinCC ControlDevelopment (ampliación mediante controles específicos del cliente)

Nota

En comparación con WinCC flexible 2008, en la funcionalidad básica se han incluido funciones de las opciones WinCC flexible /Sm@rtService, WinCC flexible /Sm@rtAccess, así como la opción WinCC flexible /OPC-Server.

Opciones para WinCC Runtime Professional

Para WinCC Runtime Professional existen las siguientes posibilidades de ampliación:

- WinCC Client (cliente estándar para crear sistemas multipuesto)
- WinCC Server (en WinCC Runtime se ha implementado la funcionalidad de servidor)
- WinCC Recipes (sistema de recetas, hasta ahora WinCC /UserArchives)
- WinCC WebNavigator (manejo y observación vía web)
- WinCC DataMonitor (visualización y evaluación de estados de proceso y datos históricos)
- WinCC ControlDevelopment (ampliación mediante controles específicos del cliente)

Nota

En comparación con WinCC V7, en la funcionalidad básica se han incluido funciones de las opciones WinCC /OPC-Server, WinCC /ConnectivityPack. Asimismo, la funcionalidad básica incluye Runtime API de WinCC /ODK.

Léame 2

2.1 Indicaciones generales

Las indicaciones del archivo Léame prevalecen sobre lo indicado en otros documentos.

Lea detenidamente las presentes indicaciones, puesto que contienen información importante acerca de la instalación y el uso. Lea estas indicaciones antes de realizar la instalación.

Indicaciones sobre el sistema de información

La siguiente función aparece descrita en el sistema de información, pero todavía no está disponible en el TIA Portal V13:

- Comparación de hardware de dispositivos

Funciones para el S7-1200 a partir de la versión de firmware V4

En el sistema de información del TIA Portal V13 se describen funciones para el S7-1200 a partir de la versión de firmware V4, que todavía no están disponibles en el software.

Los tipos de datos de 64 bits están disponibles para el S7-1200 a partir de la versión de firmware V4.

Documentación personalizada

En el DVD que viene con el producto encontrará en "Documents\UserDocumentation" unas instrucciones sobre la forma de crear y usar la documentación personalizada.

Visualización de interfaces mediante accesos online

Si esporádicamente y durante un tiempo breve no se mostraran las interfaces Ethernet para los accesos online, instale el hotfix KB2588507 (para Microsoft Windows) desde la página web de asistencia de Microsoft.

Enlace de Internet: http://support.microsoft.com/kb/2588507 (http://support.microsoft.com/kb/2588507/es)

Visualización de fuentes asiáticas en el TIA Portal

Si en caso de instalar el TIA Portal chino en otro sistema operativo asiático de Windows (p. ej. coreano) los textos no se visualizan correctamente en el TIA Portal, elija en el Control Panel de Windows inglés como "Idioma para programas no Unicode".

Instrucción "GET_NAME: Leer el nombre de un módulo"

Según la Ayuda en pantalla, el nombre que se lee depende del tipo del dispositivo IO.

Sin embargo, en el caso de un panel HMI no se devuelve el nombre de la estación, como se indica en la Ayuda, sino el nombre del módulo de interfaz.

Funcionamiento online en estado de reposo

Se recomienda no utilizar las opciones "Hibernar" y "Suspender" en el funcionamiento online, ya que es posible que ocurran problemas de comunicación. Dado el caso, adapte las opciones de ahorro de energía del equipo.

Instalación de nuevas versiones .Net o Service Packs .Net

- Cierre el TIA Portal antes de instalar una nueva versión .Net o un nuevo Service Pack de .Net en la programadora o PC.
- No abra el TIA Portal hasta no haber completado la instalación de la nueva versión .Net o del nuevo Service Pack de .Net.

Indicaciones de manejo

- Si un proyecto de la lista de últimos proyectos utilizados se encuentra en una unidad de red no conectada, pueden producirse retrasos al abrir el menú "Proyecto".
- Al insertar una CPU puede producirse un tiempo de espera prolongado si al mismo tiempo está abierto el editor de textos del proyecto. El tiempo de espera será especialmente largo si se trata de la primera CPU de un proyecto recién creado. Para garantizar un buen ritmo de trabajo, cierre el editor de textos del proyecto antes de insertar una CPU.
- En Windows 7 puede aparecer el mensaje "La aplicación no responde" al ejecutarse funciones que requieren cierto tiempo (p. ej. al cargar la CPU). En ese caso, espere a que la función finalice correctamente.
- Si ha instalado un ratón Microsoft con IntelliPoint, pueden solaparse los botones de la barra de herramientas. En ese caso, desinstale el software IntelliPoint de Microsoft.
- La activación de la opción "Virtual Desktop" con tarjetas gráficas NVIDIA puede causar problemas. En tal caso, desactive el "nView virtual desktop manager" del driver gráfico de la tarjeta NVIDIA.

Uso del TIA Portal desde el Escritorio remoto

En principio, es posible utilizar el TIA Portal a través de una conexión del Escritorio remoto. No obstante, evite interrumpir la conexión con el cliente del escritorio durante una configuración. En algún caso aislado, la interrupción puede causar el bloqueo de la interfaz del software.

Si observara un bloqueo de la interfaz, realice los siguientes pasos en el cliente del escritorio.

1. Abra el administrador de tareas de Windows y finalice el proceso "rdpclip.exe".
2. Introduzca "rdpclip.exe" en Símbolo del sistema para reiniciar el proceso.

Tenga en cuenta que con ello se perderá el contenido del portapapeles. A continuación es posible continuar la configuración de la forma habitual. Para mayor seguridad, es conveniente reiniciar el TIA Portal en cuanto haya oportunidad.

Información importante sobre las SIMATIC Memory Cards

Las SIMATIC Memory Cards han sido formateadas y preparadas por Siemens para ser utilizadas en módulos S7-1200 y S7-1500. El formato no puede sobrescribirse, ya que de lo contrario los módulos no aceptarán la tarjeta. Por ello no se permite el formateo con medios de Windows.

Comportamiento en caso de petición de forzado abierta permanente

Tenga en cuenta que una petición de forzado permanente todavía activa se mantiene incluso tras cargar un proyecto nuevo en la SIMATIC Memory Card. Por ello es necesario borrar la petición de forzado permanente antes de retirar una SIMATIC Memory Card de una CPU y de grabar otro proyecto en la tarjeta desde el PC. Si se utiliza una SIMATIC Memory Card con un contenido desconocido, es preciso formatear la SIMATIC Memory Card antes de volver a cargarla.

Direccionamiento de subred en CP 1613 y CP 1623

CP 1613 y CP 1623 son tarjetas de comunicación con microprocesador. Para ejecutar de forma segura las conexiones de comunicación, éstos se procesan en la tarjeta. La pila de protocolo se utiliza en el PC a efectos de diagnóstico (SNMP, DCP). Para permitir el acceso a las dos pilas de protocolo (es decir, firmware CP 1613/23 y acceso NDIS CP 1613/23) a los mismos interlocutores, se recomienda poner las dos pilas de una tarjeta en la misma subred.

Edición de la dirección IP de un dispositivo

A la hora de editar la dirección IP de un dispositivo no debe utilizarse el rango de direcciones de 192.168.x.241 a 192.168.x.250. En caso necesario, el sistema asigna este rango de direccione automáticamente a una programadora. Esto se aplica análogamente a todas las clases de redes en función de la máscara de subred.

Migración de proyectos con el TIA Portal

Después de migrar configuraciones hardware y bloques de programa de soluciones de automatización anteriores, compruebe primero el buen funcionamiento del proyecto migrando, antes de utilizarlo en modo productivo.

Trabajar con unidades de red sincronizadas automáticamente

Mediante la sincronización automática tras una interrupción de la red pueden guardarse datos del proyecto actuales (locales) como "backup" en la unidad de red por interacción del usuario y de este modo descargarse datos antiguos del proyecto de la unidad de red al abrir el proyecto. Por ello no es recomendable guardar proyectos del TIA Portal en unidades de red sincronizadas.

Si no obstante se trabaja en unidades de red sincronizadas, en caso de interrupciones en la red se puede continuar trabajando localmente. En estos casos es absolutamente importante comprobar que la aplicación TIA Portal esté cerrada al sincronizar los datos. La sincronización debe efectuarse de manera que los datos actuales (locales) del proyecto sustituyan los datos del proyecto que se encuentran en la unidad de red.

Entrada de decimales

Con determinados ajustes del idioma en Windows puede ocurrir que no se reconozca la entrada de valores con coma (la entrada de "1,23" produce un error). Utilice en su lugar la notación internacional ("1.23").

Protección de acceso para Memory Cards y lectores de tarjetas USB

Con la mejora de los mecanismos de seguridad en el acceso online y la ingeniería de las CPU S7-1500, el almacenamiento de datos se ha pasado a Memory Cards. Por ese motivo, al leer datos de proyecto de Memory Cards a las que se accede a través de un lector de tarjeta USB esta versión de STEP 7 no puede evaluar las contraseñas del nivel de protección configurado. El comportamiento modificado afecta a las Memory Cards para CPU de las series S7-1200/1500. Por ello hay que utilizar mecanismos de protección físicos para proteger los datos de proyecto críticos guardados en Memory Cards de estos dispositivos.

Nota

Esta limitación no está relacionada con el acceso online a dispositivos o con la protección de know how de bloques de programa.

Información relacionada con el TIA Portal en el Online Support

Relación de la información técnica más importante y soluciones para el TIA Portal en el Siemens Industry Online Support.

Enlace de Internet: TIA Portal en Siemens Industry Online Support (www.siemens.com/industry/onlinesupport/tiaportal)

Toda la información relacionada con el servicio técnico y soporte en el Siemens Industry Online Support:

Enlace de Internet: Servicio técnico y soporte en el Siemens Industry Online Support (http://support.automation.siemens.com)

Aquí también es posible inscribirse en el newsletter, que le mantiene siempre al día con información actual sobre sus productos.

Iniciar el TIA Portal

Al iniciar el TIA Portal, Windows intenta actualizar la Certificate Revocation List (CRL) de "windowsupdate.com".

Si no se dispone de acceso a Internet y hay varios servidores DNS, pueden darse rebases de tiempo que retrasen el inicio del TIA Portal.

Compatibilidad con V12 SP1

En el directorio de instalación, en ..\Portal V13\SampleProjects, se instala un proyecto V12 SP1 vacío con el nombre "TIA_Portal_Project_V12.0.1.3.ap12" para poder abrir el TIA Portal V13 en modo de compatibilidad con V12 SP1. Este proyecto debe copiarse en un directorio

local con acceso total antes de poderlo utilizar. Encontrará más información en la FAQ con ID 835846.

FAQs acerca del TIA Portal

Las FAQ del TIA Portal se encuentran en Preguntas frecuentes (http://support.automation.siemens.com/WW/view/es/28919804/133000).

2.2 STEP 7 Professional

2.2.1 Información de seguridad

Actualizaciones

Siemens suministra productos y soluciones con funciones de seguridad industrial que contribuyen al funcionamiento seguro de instalaciones, soluciones, máquinas, equipos y redes. Dichas funciones son un componente importante de un sistema global de seguridad industrial. En consideración de lo anterior, los productos y soluciones de Siemens son objeto de mejoras continuas. Por ello, le recomendamos que se informe periódicamente sobre las actualizaciones de nuestros productos.

Para el funcionamiento seguro de los productos y soluciones de Siemens, es preciso tomar medidas de protección adecuadas (como el concepto de protección de células) e integrar cada componente en un sistema de seguridad industrial integral que incorpore los últimos avances tecnológicos. También deben tenerse en cuenta los productos de otros fabricantes que se estén utilizando. Encontrará más información sobre seguridad industrial en

http://www.siemens.com/industrialsecurity. (http://www.industry.siemens.com/topics/global/en/industrial-security/Pages/Default.aspx)

Si desea mantenerse al día de las actualizaciones de nuestros productos, regístrese para recibir un boletín de noticias específico del producto que desee. Encontrará más información en

http://support.automation.siemens.com. (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=99&lang=es&referer=%2fWW%2f&func=cslib.csinfo2&siteid=csius&extranet=standard&viewreg=WW&groupid=4000002)

Ajustes de red

Las tablas siguientes muestran los ajustes de red de cada producto, necesarios para analizar la seguridad de red y la configuración de cortafuegos externos:

STEP 7 Professional					
Nombre	Número de puerto	Protocolo de transporte	Sentido	Función	Descripción
ALM	4410*	TCP	In-/Outbound	Servicio de licencias	Este servicio pone a disposición todas las funciones para licencias de software, y se utiliza tanto en el Automation License Manager como en todos los productos de software sujetos a licencias.
RFC 1006	102	TCP	Outbound	Comunicación S7	Comunicación con el controlador S7 a través de Ethernet / PROFINET con fines de programación y diagnóstico.
DCP	---	Ethernet	Outbound	PROFINET	El protocolo DCP (Discovery and basic Configuration Protocol) se utiliza en PROFINET y es la función básica para la localización y configuración de dispositivos PROFINET.
SNMP	161	UDP	Outbound	PROFINET	La función de cliente SNMP se utiliza en STEP 7 para leer información de estado de dispositivos PROFINET.
* Puerto estándar, que el usuario puede cambiar mediante configuración					

WinCC ES Basic (sin simulación)					
Nombre	Número de puerto	Protocolo de transporte	Sentido	Función	Descripción
ALM	4410*	TCP	In-/Outbound	Servicio de licencias	Este servicio pone a disposición todas las funciones para licencias de software, y se utiliza tanto en el Automation License Manager como en todos los productos de software sujetos a licencias.
HMI Load	1033	TCP	Outbound	HMI Load (RT Basic)	Este servicio se utiliza para transferir imágenes y datos de configuración a Basic Panels.
* Puerto estándar, que el usuario puede cambiar mediante configuración					

Simulation RT Basic					
Nombre	Número de puerto	Protocolo de transporte	Sentido	Función	Descripción
HMI Load	1033	TCP	Inbound	HMI Load (RT Basic)	Este servicio se utiliza para transferir imágenes y datos de configuración a Basic Panels.

Simulation RT Basic					
EtherNet/ IP	44818	TCP	Outbound	Ethernet/ canal IP	El protocolo Ethernet/IP se utiliza para conexiones con todos los PLC Allen Bradley.
	2222	UDP	Inbound	Ethernet/ canal IP	El protocolo Ethernet/IP se utiliza para conexiones con todos los PLC Allen Bradley.
Modbus TCP	502	TCP	Outbound	Canal Modbus TCP	El protocolo Modbus TCP se utiliza para conexiones con PLC Schneider.
RFC 1006	102	TCP	Outbound	Canal S7	Comunicación con el controlador S7 vía Ethernet/PROFINET
Mitsubishi MC	5002	TCP	Outbound	Canal Mitsubishi MC	El protocolo Mitsubishi se utiliza para conexiones con PLC Mitsubishi.

2.2.2 Novedades

Novedades en el TIA Portal

Para que disponga siempre de las últimas novedades, ahora en Siemens Industry Online Support hay disponible una página que le informa de las últimas novedades sobre los diferentes temas del TIA Portal.

Aquí encontrará una vista general de todas las novedades de STEP 7 Professional:

http://support.automation.siemens.com//WW/view/84047138 (http://support.automation.siemens.com/WW/view/es/84047138)

2.2.3 Indicaciones de manejo

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Funcionamiento online

El funcionamiento paralelo en modo online de STEP 7 V5.5 o anterior y STEP 7 Professional V13 no está autorizado.

Conexiones online simultáneas con una CPU S7-1200

No es posible establecer una conexión online con la misma CPU S7-1200 desde varias instancias del TIA Portal.

Uso paralelo

Si después de la instalación, el nombre o la clave de licencia del software llevase el indicativo "Combo", estará autorizado para utilizar los siguientes productos y versiones según el apartado 1.6 de las Condiciones generales (véase también el texto de la instalación):

- STEP 7 Professional 2010
- STEP 7 Professional 2006

Parametrizar y configurar módulos

Los módulos que pueden parametrizarse y configurarse con STEP 7 Professional V13 se indican en la dirección http://support.automation.siemens.com (http://support.automation.siemens.com/WW/view/es/28919804/133000).

Documentación de S7-PLCSIM

S7-PLCSIM se instala junto con STEP 7 Professional. La documentación se almacena de forma centralizada.

Los manuales se encuentran de forma estándar en: C:\Programme\Siemens\Documentación\Indicaciones de productos

Los archivos "Léame" se encuentran de forma estándar en: C:\Programme\Siemens\Documentación\Léame

Migración de PLC

Duante la migración del PLC no debe estar activado ningún perfil de instrucción.

Por ello, antes de la migración del PLC desactive el perfil de instrucción que use. A continuación puede reactivar dicho perfil.

Retirar/insertar la Memory Card

Después de retirar o insertar una Memory Card efectúe siempre un borrado total de la CPU para pasarla a un estado operacional.

Borrado de Data Logs en la Memory Card con CPU S7-1500

Si borra Data Logs (archivos csv creados con instrucciones de Data Log) de la Memory Card desde el explorador de archivos del PC, puede producirse un comportamiento no definido del Data Log. Con el método mencionado para borrar, es posible que los Data Logs no se borren por completo.

Solución: Para borrar los Data Logs de manera fiable, utilice la instrucción "DataLogDelete" en el programa de usuario o formatee la Memory Card con las funciones online en la CPU.

Borrado de WinAC RTX de un SIMATIC PC

Una Controller Application WinAC RTX con una interfaz CP Onboard asignada en un slot de interfaz (IF) no se puede borrar de un SIMATIC Box PC / Panel PC / Rack PC. Si se desea borrar la WinAC RTX habrá que eliminar previamente la interfaz CP Onboard del slot de la interfaz. Para ello es necesario seleccionar la interfaz CP Onboard en la vista de redes o dispositivos y activar otro botón de opción (p. ej. "Windows only" o "SIMATIC PC-Station") en la ventana de inspección bajo "CP Assignment" en lugar del botón de opción "SIMATIC WinAC".

Enchufar y desenchufar tarjetas o módulos Ethernet

En caso de desenchufar y volver a enchufar tarjetas o módulos Ethernet durante el funcionamiento, es necesario reiniciar el PC; de lo contrario la funcionalidad "Dispositivos accesibles" en STEP 7 o NCM PC no mostrará todos los dispositivos (estaciones). Mientras arranca el PC deben estar activadas las tarjetas o módulos Ethernet.

Comparar datos de proyecto

Las funciones de comparación (online/offline, offline/offline) no tienen en cuenta actualmente el hardware.

Cargar datos de proyecto con el TIA Portal V12 y V13 (S7-1200)

Si se han cargado los datos de proyecto de una CPU S7-1200 con el TIA Portal V13, ya no será posible acceder a dichos datos con el TIA Portal V12. Para poder hacerlo, primero debe restablecerse la configuración de fábrica de la CPU. A tal efecto lea la información contenida en la Ayuda en pantalla en "Restablecer la configuración de fábrica de la CPU".

Uso de datos de proyecto de módulos maestros IO-Link descentralizados del TIA Portal V12 en V13

Si en TIA Portal V12 utiliza módulos maestros IO-Link descentralizados que no son dispositivos GSD y que se han configurado con PCT, siga el siguiente procedimiento: para poder continuar utilizando los datos de proyecto del TIA Portal V12.0 en TIA Portal V13.0 hay que exportarlos a PCT antes de migrar el proyecto a una versión superior. Tras la migración hay que volver a importar los datos de proyecto mediante PCT.

Compatibilidad

La configuración de dispositivos y el programa de una CPU S7-1200 deben estar configurados siempre con la misma versión de STEP 7. Normalmente, el TIA Portal impide que surjan conflictos entre versiones mediante las indicaciones oportunas durante la carga en el dispositivo.

Sin embargo, en el caso de las CPU S7-1200 con versión de firmware V1.x no es posible realizar esta comprobación automática, por lo que dicha responsabilidad recae en el usuario.

2.2.4 Editar dispositivos y redes

2.2.4.1 Indicaciones generales relativas a dispositivos y redes

Contenido

Indicaciones generales relativas a dispositivos y redes.

Conexión PROFIBUS de la periferia a S7-300/S7-400/WinAC

En la conexión de periferia descentralizada del ET 200MP/SP vía PROFIBUS DP a S7-300/S7-400/WinAC Controller no pueden visualizarse sus datos I&M y con ello tampoco su diagnóstico.

Para soslayar estas limitaciones pueden conectarse ambos esclavos por medio de GSD.

Utilización del IM 155-6 DP HF con S7-300/400, ET 200S, ET 200pro o WinAC

El módulo de interfaz IM 155-6 DP HF, de la gama ET 200SP, con la referencia 6ES7 155-6BU00-0CN0 no debe usarse en PROFIBUS aguas abajo de una CPU de la gama S7-300/400, ET 200S, ET 200pro o WinAC. Esta configuración no se puede compilar.

S7-PCT IO-Link

La S7-Port Configuration Tool puede descargarse gratuitamente en el siguiente enlace.

http://support.automation.siemens.com/WW/view/37936752 (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=0&lang=es&referer=%2fWW%2f&func=cslib.csinfo&siteid=csius&groupid=4000002&groupid=4000002&extranet=standard&viewreg=WW&nodeid0=33102519&objaction=csopen)

2.2.4.2 Uso de módulos en el S7-1200

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Uso de módulos en el S7-1200

Los módulos indicados a continuación no son compatibles con el S7-1200:

Familia	Módulo	Referencia
FM S7-300	SM 338	6ES7 338-4BC01-0AB0
	FM 350-1	6ES7 350-1AH03-0AE0
	FM 350-2	6ES7 350-2AH00-0AE0, 6ES7 350-2AH01-0AE0

Familia	Módulo	Referencia
	FM 351	6ES7 351-1AH01-0AE0, 6ES7 351-1AH02-0AE0
	FM 352	6ES7 352-1AH02-0AE0
	FM 355 S	6ES7 355-1VH10-0AE0
	FM 355 C	6ES7 355-0VH10-0AE0
	FM 355-2 C	6ES7 355-2CH00-0AE0
	FM 355-2 S	6ES7 355-2SH00-0AE0
CP PtP S7-300	CP 340	6ES7 340-1AH02-0AE0, 6ES7 340-1BH02-0AE0, 6ES7 340-1CH02-0AE0
	CP 341	6ES7 341-1AH01-0AE0, 6ES7 341-1AH02-0AE0, 6ES7 341-1BH01-0AE0, 6ES7 341-1BH02-0AE0, 6ES7 341-1CH01-0AE0, 6ES7 341-1CH02-0AE0
Componente de red	Repetidor de diagnóstico	6ES7 972-0AB01-0XA0
ET 200S	1 Count 24V	6ES7 138-4DA04-0AB0
	1 Count 5V	6ES7 138-4DE02-0AB0
	1 Step 5V	6ES7 138-4DC00-0AB0, 6ES7 138-4DC01-0AB0
	2 Pulse	6ES7 138-4DD00-0AB0, 6ES7 138-4DD01-0AB0
	1 SI	6ES7 138-4DF01-0AB0
	1 SI Modbus	6ES7 138-4DF11-0AB0
	1 SSI	6ES7 138-4DB02-0AB0, 6ES7 138-4DB03-0AB0
	1 Pos Universal	6ES7 138-4DL00-0AB0
	SIWAREX	7MH4910-0AA01, 7MH4912-0AA01, 7MH4920-0AA01
ET 200M	SIWAREX	7MH4 900-2AA01, 7MH4 900-3AA01, 7MH4 950-1AA01, 7MH4 950-2AA01

Cargar en el PG/PC los comentarios de módulo del S7-1200

En las configuración centralizadas con S7-1200 no se cargan los comentarios de los módulos, submódulos y signalboards. En el cado de CP/CM solo se cargan los comentarios de la interfaz IE o DP, respectivamente. En las configuración descentralizadas con ET 200SP o ET 200MP, de los comentarios de los módulos de E/S solo se cargan los comentarios de los canales.

2.2.4.3 Sustitución de los módulos de posicionamiento ET 200S

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Sustitución de los módulos de posicionamiento ET 200S

La presente información es válida para los módulos de posicionamiento "1 Step 5V" (6ES7 138-4DC00-0AB0) de un proyecto creado con el TIA-Portal V11.0. Al sustituir estos módulos del TIA Portal V11.0 por módulos de una versión superior, los ajustes de los parámetros se restablecen con los valores predeterminados.

Este caso ocurre al seguir uno de los siguientes procedimientos:

- Sustitución del módulo de posicionamiento 6ES7 138-4DC00-0AB0 por su módulo sucesor 6ES7 138-4DC01-0AB0 mediante sustitución del dispositivo.
- Actualización de la versión del módulo con el botón correspondiente en las propiedades del dispositivo en la ventana de inspección.

2.2.4.4 CP 343-2 en SIMATIC S7 Embedded Controller EC31-RTX

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

CP 343-2 en SIMATIC S7 Embedded Controller EC31-RTX

Si bien el módulo AS-Interface CP 343-2 (referencia: 6GK7 343-2AH01) puede insertarse en un rack de ampliación del SIMATIC S7 Embedded Controller EC31-RTX (referencia: 6ES7 677-1DDxx-0BB0), el CP 343-2 no puede utilizarse con el EC31-RTX.

2.2.4.5 Indicaciones relativas a Online y diagnóstico

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Detección de hardware y posterior conexión online

Si se ejecuta el comando "Online > Detección de hardware" para una CPU sin especificar, no se carga la configuración online de la CPU. Si no se carga en la CPU la configuración obtenida mediante la detección de hardware, siempre habrá diferencia entre las configuraciones offline y online en la vista de dispositivos y redes. En la vista Online y diagnóstico puede verse que hay configuraciones diferentes, aunque sean idénticas las referencias de la CPU realmente existente y de la CPU offline.

2.2.4.6 Componentes de red

Componentes de red

CP 1242-7

Copiar un CP 1242-7 en otro proyecto

Si copia un CP 1242-7 de un proyecto a otro, los siguientes parámetros se cambiarán en el grupo de parámetros "Identificación del CP" de la estación de destino:

- Número de proyecto del CP
- Número de estación del CP

Cargar en dispositivo

La función "Cargar en dispositivo" solo puede realizarse en el CP 1242-7 mediante una conexión TeleService del siguiente modo:

1. Seleccione el CP en STEP 7.
2. Seleccione el menú "Online" > "Cargar en dispositivo".
3. Seleccione la interfaz TeleService en el cuadro de diálogo visualizado "Carga avanzada".
4. Cargue los datos del proyecto desde el cuadro de diálogo "Carga avanzada".

Cargar de dispositivo

La función "Cargar de dispositivo" no es soportada por el CP 1242-7.

CM/CP S7-1500

CM 1542-1: eliminar el almacenamiento remanente de la dirección IP en caso de doble direccionamiento

La dirección IP y el nombre de dispositivo del CM 1542-1 se guardan de forma remanente:

Cuando el CM detecta un doble direccionamiento p. ej. al arrancar en otra red, el CM no se incorpora a la red. El CM conmuta a RUN y no está accesible a través de la interfaz Ethernet.

Para incorporar el CM a la red, elimine la dirección IP guardada de forma remanente como se indica a continuación:

1. Retire la tarjeta de memoria de la CPU.
2. En caso de parada de la CPU, ajuste a 0.0.0.0 la dirección IP del CM sin configuración a través de DCP con la Primary Setup Tool (PST).
   Ha eliminado la dirección IP del CM guardada de forma remanente. El CM puede incorporarse a la red.
3. Vuelva a insertar la tarjeta de memoria de la CPU.

CP PC:

CP PC para VMware vSphere Hypervisor ESXi

Los siguientes procesadores de comunicaciones están liberados bajo VMware vSphere Hypervisor ESXi:

- CP 5622
- CP 5711
- Adaptador de red VMware E1000

Los siguientes procesadores de comunicaciones no están liberados bajo VMware vSphere Hypervisor ESXi:

- CP 5611
- CP 5612
- CP 5621
- CP 5512

Otros procesadores de comunicaciones como CP 1613, CP 1623, CP 5613, CP 5614, CP 5623, CP 5624 están liberados además mediante instalación del software de PC SIMATIC NET. Para más información al respecto consulte el archivo Léame del software de PC SIMATIC NET.

Módulos de seguridad

Migración de proyectos con CP Ethernet y funciones de seguridad informática activadas

En proyectos de STEP 7 V5.5 que incluyan CP Industrial Ethernet con funciones de seguridad informática activadas, los ajustes de seguridad informática se desactivarán en caso de migración a STEP 7 Professional.

En caso necesario, proceda del siguiente modo tras la migración:

1. Active la funcionalidad de seguridad.
2. Configure los ajustes de seguridad necesarios.

Nombres simbólicos en la configuración de funciones de seguridad

No se soporta el uso de nombres simbólicos en la configuración de las funciones de seguridad.

Migración de la lista de protección de acceso IP al activar funciones de seguridad

Una lista de protección de acceso IP activa se convierte a reglas de cortafuegos al activar las funciones de seguridad. Dichas reglas son visibles en el modo de cortafuegos avanzado, donde pueden adaptarse. El modo de cortafuegos avanzado se activa automáticamente.

Diagnóstico online de seguridad de CPs S7

El diagnóstico online de seguridad de un CP apto para seguridad solo es posible si la conexión online se establece directamente a través del CP. Si la conexión online de STEP 7 con la estación se estableció a través de la CPU, el diagnóstico online de seguridad del CP no será posible.

Para establecer una conexión online directamente a través del CP es necesario registrar la dirección IP del CP en el campo de entrada "Dirección del dispositivo" en la entrada "Accesos online". Solo entonces se podrá realizar el diagnóstico online de seguridad del CP.

Modo IKE

Durante la negociación de la clave en la fase 1 es preferible el modo IKE "Main". Por lo general, este modo ofrece más seguridad que el modo "Aggressive". Un motivo para el uso del ajuste "Aggressive" es el empleo de grupos VPN con diferentes claves "pre-shared", ya que con el ajuste "Main" se soporta una sola "pre-shared key".

El modo IKE "Aggressive" no debe usarse con certificados. Utilice únicamente "pre-shared keys" en el modo IKE "Aggressive".

El SOFTNET Security Client solo soporta el modo IKE "Main".

Un módulo de seguridad no puede estar simultáneamente en grupos VPN que utilicen modos IKE distintos.

Carga de configuraciones en una estación S7-300/S7-400 vía túnel VPN

Al cargar una configuración en una estación S7-300-/S7-400 a través de la interfaz Gigabit de un CP x43-1 Advanced, la ruta en la que tiene lugar la carga se guarda en el proyecto. Si el proyecto se carga a continuación a través de un túnel VPN establecido entre un módulo SCALANCE S y el CP x43-1 Advanced, la operación de carga fallará debido a que ha cambiado la ruta de carga.

Para realizar la operación de cardga a través del túnel VPN, proceda del siguiente modo:

1. Conéctese a través del botón "Establecer conexión online" con la interfaz Gigabit del CP x43-1 Advanced.
2. Deshaga la conexión online al CP x43-1 Advanced.
3. Cargue el proyecto en la estación a través de la interfaz Gigabit del CP x43-1 Advanced.

Establecimiento de túneles VPN mediante CP 1543-1

Para establecer conexiones de túnel VPN por medio del CP 1543-1 con el método de autenticación "Pre-shared key", el interlocutor de la comunicación VPN del CP 1543-1 deberá ser a su vez también un CP 1543-1. El establecimiento de conexiones de túnel VPN por el CP 1543-1 a los demás interlocutores de VPN solo es posible con el método de autenticación "Certificado".

Convertidor de medios

Cuando vaya a migrar un proyecto, deberá reconfigurar el convertidor de medios.

2.2.5 Programación del PLC

2.2.5.1 Indicaciones generales relativas a la programación del PLC

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Información sobre la seguridad de la red

Los accesos de comunicación entre TIA Portal y CPU, o bien entre HMI y CPU (excepto cuando los accesos HMI se utilizan a través de la "Comunicación GET/PUT) ofrecen funciones de seguridad integradas. Estas ofrecen una mayor protección frente a manipulaciones y una mayor protección de acceso. Para prevenir accesos no autorizados a través de redes a una CPU con accesos de comunicación estándar, como p.ej. "GET/PUT", "TSEND/TRCV", "Modbus", "FETCH/WRITE", se recomienda tomar además medidas de protección adecuadas (p. ej. concepto de protección de células).

Cambiar el número de los tipos de datos PLC

Las CPUs del módulo S7-1200 procesan tipos de datos PLC con números debido a las ventajas de rendimiento que ello aporta. Dichos números permanecen ocultos para el usuario, ya que el sistema procesa los números de forma autónoma. Eso significa que los conflictos entre dichos números se resuelven automáticamente. No obstante, si se utilizan bloques con protección de know how que a su vez usan tipos de datos PLC, en determinadas circunstancias puede resultar ventajoso establecer un esquema numérico propio (se recomienda > 5000), ya que la resolución automática de los conflictos numéricos podría requerir la solicitud de una contraseña (compilación necesaria).

Para renumerar el número predeterminado de un tipo de datos PLC, proceda del siguiente modo:

1. Abra la librería del proyecto en la Task Card "Librerías".
2. Arrastre el tipo de datos PLC compilable mediante Drag & Drop a la carpeta "Tipos".
   Se abre el cuadro de diálogo "Agregar tipo".
3. Introduzca las propiedades del tipo nuevo.
4. Confirme con "Aceptar".
5. Haga clic con el botón derecho del ratón en el tipo de datos PLC.
   Se abre el menú contextual.
6. Elija "Editar tipo" en el menú contextual.
7. Abra de nuevo la Task Card "Programación PLC".
   Después del nombre del tipo de datos PLC ahora se indica el estado "en prueba".
8. Haga clic con el botón derecho del ratón en el tipo de datos PLC.
   Se abre el menú contextual.
9. Elija "Propiedades" del menú contextual.

10. En la sección "General", cambie el número del tipo de datos PLC.
11. Elija "Habilitar versión" en el menú contextual de la librería.

El tipo de datos PLC ahora tiene el número nuevo y puede utilizarse. El número asignado se conserva aunque se anule la tipificación del tipo de datos PLC.

Generación de fuentes externas a partir de bloques

Al generar fuentes externas a partir de bloques, los cambios de valores predeterminados de tipos de datos PLC realizados directamente en la interfaz del bloque no se exportan a la fuente. Por ese motivo, al volver a importar las fuentes esos valores ya no están disponibles. En su lugar se vuelven a aplicar los valores predeterminados. Para impedir esta pérdida de datos en los valores predeterminados modificados, los cambios deben realizarse directamente en el tipo de datos PLC y no en la interfaz del bloque. De ese modo los cambios sí se exportan al generar fuentes externas.

Indexación indirecta de componentes ARRAY del tipo de datos Secuencia de bits en SCL

En el TIA Portal V13, en SCL existe la posibilidad de indicar una variable del tipo de datos BYTE, WORD, DWORD o LWORD como índice además de una variable del tipo de datos Entero para el direccionamiento de los componentes de un ARRAY en una CPU de las series S7-1200/1500, siempre que la verificación CEI no esté ajustada. Eso no es posible en una CPU de las series S7-300/400.

Eso provoca una incompatibilidad con respecto al TIA Portal V12 SP1, ya que en él también está permitida la indexación indirecta con el tipo de datos Secuencia de bits en una CPU de las series S7-300/400.

Declaración de parámetros internos de un bloque GRAPH como remanentes (S7-1500 < V1.5)

Si declara como remanentes parámetros internos de un bloque GRAPH, el control secuencial permanecerá en el estado actual y, después del cambio a "RUN", continuará con la ejecución en el punto en el que se interrumpió. Tenga en cuenta que al activar el nuevo parámetro "Parámetro interno remanente" en un bloque GRAPH no deben utilizarse supervisiones.

Conversión explícita de tipo de datos en SCL (S7-1200/1500)

A partir del TIA Portal V13, en la conversión explícita de datos en SCL de SINT/INT/DINT/LINT_TO_STRING o WSTRING la cadena de caracteres se representa con un carácter antepuesto y se transmite justificada a la izquierda.

Eso provoca una incompatibilidad con el TIA Portal V12 SP1, ya que aquí la cadena de caracteres se transfería justificada a la derecha en la conversión.

Funciones de la librería global "Long Functions" con PLCSIM S7-1500

Mediante el proceso de compilación, las funciones de la familia de CPU S7-1500 quedan vinculadas a la CPU para protegerla contra manipulaciones. Las funciones empleadas en la CPU de la librería global "Long Functions" que poseen este tipo de vínculo no se pueden cargar en una simulación. Sin embargo, existe la posibilidad de sustituir estos elementos por sus originales de la librería global. Estos elementos no están vinculados a la CPU.

Uso del OB de error de acceso a la periferia en una CPU S7-1500

Si en una CPU S7-1500 se utiliza el OB de error de acceso a la periferia y la casilla de verificación "Acceso optimizado al bloque" está activada en "Propiedades > Atributos" (ajuste predeterminado), para la variable "Area" rige la información de arranque: "Area" es la identificación del área en la que se ha producido el acceso erróneo y puede adoptar los valores siguientes:

- B#16#01: acceso directo a la entrada
- B#16#02: acceso directo a la salida
- B#16#81: acceso a la memoria imagen de proceso de las entradas
- B#16#82: acceso a la memoria imagen de proceso de las salidas

Cargar programas no coherentes en una CPU S7-1500

En el TIA Portal no es posible cargar programas no coherentes en una CPU S7-1500 sin realizar previamente una prueba de coherencia. Durante la carga se comprueban automáticamente todos los bloques del programa y, si hay incoherencias, se compilan nuevamente. No obstante, si la CPU contiene programas que fueron cargados con versiones anteriores de STEP 7, estos programas pueden presentar incoherencias.

En tal caso, tenga en cuenta lo siguiente:

Si carga un programa incoherente desde un dispositivo, no podrá cargar después el programa en una CPU S7-1500 sin modificarlo, ya que durante la carga se comprueba automáticamente la coherencia y se eliminan las incoherencias encontradas.

Cargar bloques de datos desde un dispositivo

Tenga en cuenta que ambas familias de CPU tratan los valores de las variables de distinto modo al cargar bloques desde un dispositivo.

- S7-1200/1500
  Las CPU de la serie S7-1200 guardan en la memoria de carga los valores de arranque con los que se cargó un bloque de datos en la CPU. Si carga un bloque de datos de una CPU S7-1200, estos valores de arranque de la memoria de carga se aplicarán nuevamente al bloque de datos offline. Aunque haya modificado valores durante el funcionamiento en la memoria de carga con la instrucción "WRIT_DBL", en el bloque de datos offline se aplicarán los valores que se cargaron originalmente en el dispositivo.
  Para aplicar los valores de observación actuales de la CPU, puede utilizarse la función "Instantánea de los valores de observación". Encontrará información detallada al respecto en el sistema de información, en el capítulo "Programar bloques de datos".
- S7-300/400
  Las CPU de la serie S7-300/400 también guardan en la memoria de carga los valores de arranque con los que se cargó un bloque de datos en la CPU. Sin embargo, los valores no se pueden releer de la memoria de carga. Al cargar un bloque de datos de un dispositivo, se cargan los valores de observación actuales de la CPU y se registran en la columna "Valores de arranque" del bloque de datos offline.

Comportamiento de los valores de arranque al "Cargar de dispositivo" (S7-1500)

Los valores de arranque modificados con la instrucción "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" se vuelven a perder al ejecutar la acción "Cargar de dispositivo".

Importación de bloques de organización de fuentes AWL/SCL externas

Las CPU de la serie S7-300/400 aceptan durante la importación bloques de organización de cualquier nombre. Si se importa un OB con un nombre estándar (por ejemplo, CYCL_EXC o CYC_INT5), se asigna automáticamente la clase de OB correcta. Si se importa un OB con un nombre desconocido, se crea un OB 0. Este número puede modificarse después de la importación para, de este modo, asignar la clase de OB deseada. Para modificar el número del OB, elija el comando "Propiedades" del menú contextual.

Al importar fuentes externas a una CPU de la serie S7-1200/1500 solo se aceptan bloques de organización con nombres estándar. Un OB con nombre desconocido deja de ser válido después de la importación y se identifica con un símbolo de "prohibido aparcar". Así pues, el nombre del bloque de organización debe modificarse con un editor de texto en la barra de título de la fuente externa antes de importarlo al TIA Portal.

Memoria imagen de proceso de salidas PTO/PWM

No utilice las salidas de PTO/PWM en la memoria imagen de proceso (p. ej. en accesos del programa de usuario, en funciones online o en HMI). La frecuencia de actualización de la memoria imagen de proceso es notablemente inferior a la velocidad de los cambios de señal. Por ello, la indicación en la memoria imagen de proceso no refleja el curso de la señal.

Desplazar bloques de sistema

No está permitido desplazar objetos de la carpeta "Bloques de sistema" a la carpeta "Bloques de programa".

Observar bloques en KOP y FUP

Si el inicio del circuito está fuera del área visible, es posible que el valor de entrada no pueda determinarse. En este caso el circuito se representa en gris.

Resetear los requisitos de registro para bloques GRAPH

Para el registro de errores y fallos mediante el parámetro de entrada "REG_EF", entre las familias de CPU existen las diferencias siguientes a la hora de resetear los requisitos de registro:

- S7-300/400
  Los requisitos de registro se mantienen hasta que se produce un error de interlock o supervisión en una etapa. En este momento se dispara el evento "R1", se ejecuta la acción vinculada y se resetean los requisitos de registro.
- S7-1500
  La solicitud de registro se inicializa siempre, independientemente de si se produce un error de interlock o de supervisión en el ciclo actual.

Evitar la utilización en librerías de tipos de datos PLC generados por el sistema

Al crear instancias, algunas instrucciones crean tipos de datos PLC propios que se guardan en la carpeta de proyecto "Tipos de datos PLC". Los tipos de datos PLC generados por el sistema no deben utilizarse en ninguna librería porque el sistema puede volver a crearlos en cualquier momento y, además, pueden propiciar un comportamiento indeseable del sistema.

Nota sobre la coherencia de datos en la comunicación S7 y Open User Communication

En la familia de módulos S7-1500 es posible modificar parámetros durante el funcionamiento en las instrucciones de la comunicación S7, así como en la Open User Communication. Los parámetros modificados se utilizan de inmediato aunque haya todavía una petición activa. En ambos casos pueden producirse incoherencias en los datos. Para evitar incoherencias en los datos, en ningún caso deben modificarse parámetros mientras se está ejecutando una petición.

Utilizar bloques de datos globales en asignaciones

No es posible asignar el contenido de un bloque de datos global a un bloque de datos con la misma estructura, p. ej. mediante una casilla de desplazamiento, si uno de los dos DB contiene una reserva de memoria.

2.2.5.2 Instrucciones

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Indicaciones sobre el sistema de información

La siguiente función aparece descrita en el sistema de información, pero todavía no está disponible en el TIA Portal V13:

- Instrucción "ASI_CTRL" para CPU del sistema S7-1200/1500

Instrucción "PE_WOL: Iniciar y finalizar vía WakeOnLan el modo de ahorro de energía" (S7-1500)

Esta instrucción solo soporta dispositivos con un número de dispositivo inferior al 256. Los dispositivos con un número superior a éste no son considerados.

Instrucción "TMAIL_C: Enviar e-mail"

Además de los valores descritos en la Ayuda en pantalla, el parámetro RET_VAL puede adoptar el siguiente valor:

Código de error (W#16#....)	Explicación
8555	Si se escriben varias direcciones, éstas deben separarse con ";".

Instrucciones "BLKMOV: Copiar área", "UBLKMOV: Copiar área sin interrupciones" y "FILL: Rellenar área"

Además de los valores descritos en la Ayuda en pantalla, el parámetro RET_VAL puede adoptar el siguiente valor:

Código de error (W#16#....)	Explicación
8352	Los tipos de datos WSTRING, WCHAR y BOOL no se soportan.

Instrucción "GET_IM_DATA: Leer datos de identificación y mantenimiento" (S7-1500)

La estructura "IM0_DATA" no está disponible como estructura de sistema en el TIA Portal V13 para leer los datos I&M0. En su lugar, puede utilizar la estructura de datos "ARRAY of BYTE" para depositar los datos.

Código de error 0x807F en el parámetro RET_VAL

Las CPU S7-1500 son capaces de ejecutar como máximo 10 instancias de la instrucción "Leer datos de identificación y mantenimiento". Si se excede este número, el parámetro de salida RET_VAL adopta el valor 0x807F y no se ejecuta la instrucción.

Solución:

- Evite más de 10 llamadas paralelas.
- Repita la llamada en otro momento ya que se trata de un error temporal.

Instrucción "Gen_UsrMsg: Generar avisos de diagnóstico de usuario" (S7-1500)

La instrucción "Gen_UsrMsg" descrita en la ayuda queda suprimida en V13.

Uso de instrucciones con parámetros del tipo VARIANT en bloques lógicos con diferentes tipos de acceso (S7-1500)

Los bloques lógicos (FB/FC) y bloques de datos (DB) se pueden crear con diferentes tipos de acceso ("Estándar" y "Optimizado"). En los bloques lógicos se puede llamar cualquier tipo de instrucciones. Algunas instrucciones (p. ej. "WRIT_DBL" y "READ_DBL") utilizan punteros del tipo VARIANT en los parámetros de entrada y salida para direccionar bloques de datos.

Asegúrese de no utilizar estas instrucciones en programas en los que los bloques lógicos se llamen unos a otros con diferente tipo de acceso, de modo que pudieran darse los siguientes casos:

- Una estructura de un bloque de datos estándar se transfiere directa o indirectamente a un bloque lógico optimizado que retransfiere esta estructura directa o indirectamente a uno de los bloques arriba mencionados.
- El caso inverso en el que una estructura de un bloque lógico optimizado se transfiera directa o indirectamente a un bloque de datos estándar que retransfiere esta estructura directa o indirectamente a uno de los bloques arriba mencionados.

Parámetros de salida de las instrucciones S7-1500

Si al procesar la instrucción se produce un error de acceso o un error de programación, el valor de la salida queda sin definir.

2.2.5.3 Comprobar el programa de usuario

Test con la tabla de observación

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Accesos múltiples a la misma CPU (S7-300/400/1200)

Desde una programadora o un PC sólo se puede acceder a una CPU con el TIA Portal abierto. Los accesos múltiples a la misma CPU no están permitidos y pueden provocar errores.

Cargar bloques de datos mientras se procesa una petición de forzado

Nota

Al cargar bloques de datos modificados mientras se procesa una petición de forzado pueden darse estados operativos imprevistos. La petición de forzado continúa forzando la dirección indicada, mientras que la asignación de direcciones puede haber cambiado en el bloque de datos. Finalice las peticiones de forzado que se estén procesando antes de proceder a cargar bloques de datos.

Función "Desbloquear salidas"

La función "Desbloquear salidas" no está disponible en el TIA Portal V13 para las CPU de la serie S7-1500.

Esta función solo puede ejecutarse en el TIA Portal V13 con una CPU S7-300, S7-400 o S7-1200.

Probar programas convertidos de STEP 7 V10.5/V11.0

Para observar y probar un programa convertido de STEP 7 V10.5 es necesario compilarlo y cargarlo primero con STEP 7 V13.0.

Test con la tabla de forzado permanente

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Forzado permanente de variables con acceso directo a la periferia

Si en el programa de usuario se utiliza un acceso directo a la periferia para una CPU S7-300, no está permitido el forzado permanente de esta dirección de periferia.

Ejemplo

Si en el programa de usuario se accede a la dirección "IB0:P", no podrán forzarse permanentemente las siguientes áreas de direcciones de la periferia: I0.0:P, IB0:P, IW0:P e ID0:P.

2.2.6 Inter Project Engineering (IPE)

2.2.6.1 Indicaciones relativas a IPE

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Campo de comentario del proxy del dispositivo

En contra de lo descrito en el sistema de información del TIA Portal V13, el campo de comentario de un proxy de dispositivo no puede editarse en la función "Inter Project Engineering (IPE)".

Funcionalidad limitada

En el TIA Portal V13 no se puede utilizar la funcionalidad "Vista general S7-GRAPH" y "Visor de código PLC" en combinación con el proxy del dispositivo IPE.

2.2.7 Funciones tecnológicas

2.2.7.1 Indicaciones sobre las funciones tecnológicas

Módulos tecnológicos S7-1500

Contrariamente a lo indicado en la Ayuda en pantalla, el editor de puesta en marcha para los módulos tecnológicos S7-1500 todavía no está disponible.

S7-1500 Motion Control - Configuración de un eje de posicionamiento desde un dispositivo HMI

Si una variable existe tanto en el eje de velocidad como en el eje de posicionamiento (p. ej. <TO>.Velocity), no podrá acceder a la variable del eje de posicionamiento desde un dispositivo HMI.

Copie la variable del eje de posicionamiento desde el programa de usuario a un bloque de datos que haya creado previamente. A la copia de la variable sí puede acceder desde un dispositivo HMI.

S7-1500 Motion Control: FAQ Soluciones para alarmas tecnológicas

El FAQ con las soluciones para alarmas tecnológicas se encuentra disponible en Internet (http://support.automation.siemens.com/WW/view/es/66958052).

S7-1500 Motion Control: Optimización

Después de iniciar un paso de prueba de la optimización y una vez transcurrido el tiempo de medición se frena el eje con la deceleración máxima configurada.

S7-1500 Motion Control: Modificar la configuración offline

Si modifica la configuración del eje en modo offline y la carga en la CPU en estado operativo RUN, los parámetros modificados se mostrarán, pero solo se harán efectivos tras reiniciar el objeto tecnológico.

Llame para ello MC_Reset con Restart = TRUE.

S7-1500 Motion Control y S7-1200 Motion Control

Existen diferencias entre S7-1500 Motion Control y S7-1200 Motion Control. Los modos de la instrucción de Motion Control MC_Home no son compatibles.

Es necesario adaptar el programa de usuario. Los programas que se transfieren de una familia de CPU a otra, deben reiniciarse siempre.

S_USSI a S7-1500

Tenga en cuenta lo siguiente si utiliza la instrucción S_USSI de la periferia descentralizada ET 200S 1SI en un S7-1500.

El parámetro ANZ indica el valor 0 también en caso de error.

Si la CPU es conmutada a STOP por la instrucción S_USSI, compruebe todos los errores posibles listados en el parámetro ANZ.

2.2.7.2 Indicaciones sobre las funciones tecnológicas

No hay indicaciones sobre las funciones tecnológicas.

2.2.7.3 Indicaciones sobre las funciones tecnológicas

Parámetrización FM x51 y FM x52

Contrariamente a lo indicado en la ayuda en pantalla, en caso de modificar el sistema de medida los valores ya introducidos de los parámetros no se convierten al nuevo sistema de medida.

Programación FM x50-1, FM x51 y FM x52

Los "valores iniciales" contenidos en las tablas de parámetros que figuran en la ayuda en pantalla de las instrucciones equivalen al "valor predeterminado" que se indica en la interfaz del programa de usuario.

Migración IM 174

Al migrar proyectos con IM 174, la parametrización del IM 174 se ajusta a los valores predeterminados. El IM 174 debe reparametrizarse en cualquier caso después de la migración.

2.2.7.4 SIMATIC STEP 7 Easy Motion Control

Indicaciones relativas a SIMATIC STEP 7 Easy Motion Control

No hay indicaciones relativas a SIMATIC STEP 7 Easy Motion Control.

2.2.7.5 SIMATIC STEP 7 PID Professional

Indicaciones relativas a SIMATIC STEP 7 PID Professional

No hay indicaciones relativas a SIMATIC STEP 7 PID Professional.

2.3 WinCC Basic

2.3.1 Indicaciones de seguridad

Información de seguridad

Siemens suministra productos y soluciones con funciones de seguridad industrial que contribuyen al funcionamiento seguro de instalaciones, soluciones, máquinas, equipos y redes. Dichas funciones son un componente importante de un sistema global de seguridad industrial. En consideración de lo anterior, los productos y soluciones de Siemens son objeto de mejoras continuas. Por ello, le recomendamos que se informe periódicamente sobre las actualizaciones de nuestros productos.

Para el funcionamiento seguro de los productos y soluciones de Siemens, es preciso tomar medidas de protección adecuadas (como el concepto de protección de células) e integrar cada componente en un sistema de seguridad industrial integral que incorpore los últimos avances tecnológicos. Asimismo hay que tener en cuenta los productos utilizados de otros fabricantes. Encontrará más información sobre seguridad industrial en

http://www.siemens.com/industrialsecurity (http://support.automation.siemens.com)

Si desea mantenerse al día de las actualizaciones de nuestros productos, regístrese para recibir un boletín de noticias específico del producto que desee. Encontrará más información en

http://support.automation.siemens.com (http://support.automation.siemens.com)

Contraseñas

En WinCC hay varias contraseñas preajustadas. Por motivos de seguridad es conveniente cambiar dichas contraseñas.

- Para el usuario "Administrador" está preajustada la contraseña "administrator".

Comunicación vía Ethernet

En la comunicación basada en Ethernet, el usuario final es el responsable de la seguridad de su red de datos, ya que en intrusiones selectivas que provocan una sobrecarga del equipo, no está garantizado el funcionamiento correcto de éste.

Finalización automática de runtime

Si en el panel de operador está activada la transferencia automática y se inicia una transferencia en el PC de configuración, en el panel de operador se cierra automáticamente el proyecto abierto. El panel de operador cambia entonces automáticamente al modo de operación "Transfer". El modo de transferencia puede provocar reacciones imprevistas en la instalación.

Desactive la transferencia automática tras la fase de puesta en servicio para que el panel de operador no pase por error al modo de transferencia. Para bloquear el acceso a los ajustes

de transferencia, y evitar así modificaciones no autorizadas, asigne una contraseña en el Control Panel.

Ajustes de red

Las tablas siguientes muestran los ajustes de red de cada producto, necesarios para analizar la seguridad de la red y la configuración de cortafuegos externos:

WinCC Basic (sin simulación)					
Nombre	**Número de puerto**	**Protocolo de transporte**	**Dirección**	**Función**	**Descripción**
ALM	4410*	TCP	Inbound, outbound	Servicio de licencias	Este servicio pone a disposición todas las funciones para licencias de software, y se utiliza tanto en el Automation License Manager como en todos los productos de software sujetos a licencias.
HMI Load	1033	TCP	Outbound	HMI Load (RT Basic)	Este servicio se utiliza para transferir imágenes y datos de configuración a Basic Panels.
* Puerto estándar, que el usuario puede cambiar mediante configuración					

WinCC Simulation para Basic Panels					
Nombre	**Número de puerto**	**Protocolo de transporte**	**Dirección**	**Función**	**Descripción**
HMI Load	1033	TCP	Inbound	HMI Load (RT Basic)	Este servicio se utiliza para transferir imágenes y datos de configuración a Basic Panels.
EtherNet/IP	44818	TCP	Outbound	Ethernet/canal IP	El protocolo Ethernet/IP se utiliza para conexiones con todos los PLC Allen Bradley.
	2222	UDP	Inbound	Ethernet/canal IP	El protocolo Ethernet/IP se utiliza para conexiones con todos los PLC Allen Bradley.
Modbus TCP	502	TCP	Outbound	Canal Modbus TCP	El protocolo Modbus TCP se utiliza para conexiones con PLC Schneider.
RFC 1006	102	TCP	Outbound	Canal S7	Comunicación con el controlador S7 vía Ethernet/PROFINET
Mitsubishi MC	5002	TCP	Outbound	Canal Mitsubishi MC	El protocolo Mitsubishi se utiliza para conexiones con PLC Mitsubishi.

Consulte también

http://www.siemens.com/industrialsecurity (http://www.industry.siemens.com/topics/global/en/industrial-security/Pages/Default.aspx)

2.3.2 Novedades

Novedades en el TIA Portal

En Siemens Industry Online Support puede informarse acerca de las novedades de las distintas áreas temáticas del TIA Portal.

Encontrará todas las novedades sobre WinCC aquí: Novedades (http://support.automation.siemens.com/WW/view/es/88360672)

2.3.3 Indicaciones de uso

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Copia de dispositivos HMI con conexiones HMI

En caso de copiar un dispositivo HMI con conexiones HMI a un controlador, la conexión HMI en el nuevo dispositivo HMI no se conectará automáticamente con un controlador existente que tenga el mismo nombre. Esto vale tanto para la copia en un mismo proyecto como para la copia entre distintos proyectos.

Para poder acceder con una variable HMI a la variable del controlador también en el nuevo dispositivo HMI, se recomienda completar la configuración de la conexión HMI inmediatamente después de copiarla. Proceda del siguiente modo:

1. Abra el editor "Dispositivos y redes".
2. Conecte el nuevo dispositivo HMI con la red deseada.
3. Abra la tabla de conexiones.
4. Seleccione la conexión HMI del nuevo dispositivo HMI.
5. Elija el controlador deseado en "Interlocutor".

Si entre que se copia el dispositivo HMI y se completa la conexión se compila el nuevo dispositivo HMI o se conectan otras variables PLC, en algunos casos se genera una conexión HMI adicional con el mismo controlador. Este es el caso especialmente cuando hay variables HMI con elementos de matrices de DB.

Cambio de dispositivo

Después de un cambio de dispositivo conviene comprobar el aspecto de las imágenes configuradas. Si cambia el tamaño de la pantalla, es posible que cambien también la posición y el aspecto de algunos objetos de imagen, como visores de recetas y avisos.

Cambio de dispositivo - Comunicación

Al cambiar de dispositivo, pueden aparecer mensajes de error del tipo "... no se soporta en la nueva configuración y por lo tanto será eliminado.". Estos mensajes hacen referencia a conexiones configuradas del dispositivo y son generados p. ej. debido a un número distinto de interfaces de los paneles de operador. Después de cambiar de dispositivo estas conexiones aparecen marcadas en rojo. Si desea seguir utilizando estas conexiones, deberá adaptar la configuración de la conexión. Proceda del siguiente modo:

1. Abra el editor "Dispositivos y redes".
2. Haga clic en "Conectar en red" en la barra de herramientas de la vista de redes.
3. Conecte la interfaz del panel de operador con la interfaz de la CPU.
4. Haga clic en la tabla "Conexiones" en el área de tablas de la vista de redes.
5. Seleccione la conexión marcada en rojo.
6. Configure la nueva interfaz en la ventana de inspección "Propiedades > General > Interfaz".

Indicación de la fecha y hora de cambio en la ventana general

Los tiempos de cambio indicados en la ventana general solo hacen referencia a los cambios realizados en el propio objeto. Los cambios realizados en objetos subordinados, p. ej. en los objetos de una imagen, no modifican la fecha y hora correspondiente al último cambio de la imagen.

Asistente de paneles de operador

Si se crea un panel con una pantalla a color con el asistente de paneles de operador, es posible que los gráficos de los botones de navegación se muestren en blanco y negro. Sin embargo, este error solo ocurre si el nuevo panel se ha creado con el nombre que tenía un panel de pantalla monocromática que ha sido borrado entre tanto.

Este error puede evitarse borrando los gráficos correspondientes de la colección de gráficos cada vez que se borre un panel del proyecto.

Objetos con referencias a objetos de la librería del proyecto

En WinCC flexible se podían utilizar dos métodos para copiar.

- Con la "Copia simple" se copia p. ej. una imagen WinCC flexible, incluido un campo ES. De una variable configurada en el campo ES solo se copia el nombre de objeto, ya que se trata de una referencia.
- Al "Copiar" se copian una imagen, el campo ES que contenga y la variable configurada en el campo ES junto con sus propiedades.

Asimismo, para guardar un objeto en una librería se podían utilizar ambos métodos. Con la migración se migran librerías del proyecto con los objetos que contienen, pudiéndose utilizar en WinCC.

Sin embargo, en WinCC ahora solo se dispone de un método para copiar. En lo que respecta a las variables, este método funciona como la "Copia simple" de WinCC flexible. En cuanto a gráficos, listas de gráficos y listas de textos, se comporta como la función "Copiar" de WinCC flexible.

Si en WinCC flexible se guardan en una librería objetos con referencias a variables, al utilizar estos objetos en WinCC hay que reconfigurar los objetos referenciados.

Configurar idiomas de proyecto de Asia oriental en un PC sin sistema operativo asiático

Si no tiene instalado un sistema operativo asiático en el PC y selecciona un idioma de Asia oriental en la configuración, en el editor "Configuración de Runtime > Idioma & fuente" se invalida la fuente estándar.

Para solucionar el problema, active la opción "Instalar archivos para los idiomas de Asia oriental" en "Configuración regional y de idioma" del panel de control.

Orden de instalación para Startdrive

Si instala Startdrive en un PC, siga este orden de instalación:

- Instale STEP 7 V13.0.
- Instale Startdrive.

Compatibilidad con V12

En el directorio de instalación, en ..\Portal V13\SampleProjects, se instala un proyecto V12.0.1.3 vacío con el nombre "TIA_Portal_Project_V12.0.1.3.ap12" para poder abrir el TIA Portal V13 en modo de compatibilidad con V12. Este proyecto debe copiarse en un directorio local con acceso total antes de poderlo utilizar. Encontrará más información en la FAQ con ID 66027369.

Explorador de archivos en un PC Windows 8 con pantalla táctil

En un PC Windows 8 con pantalla táctil, el diálogo del explorador de archivos solo puede manejarse con el ratón, el teclado o el teclado de pantalla (sin utilizar la función Touch). En un PC con pantalla táctil y Windows 8 es recomendable utilizar el diálogo del explorador de archivos del sistema operativo con un script.

2.3.4 Migración

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Idiomas del proyecto en WinCC

En WinCC V13 no se soportan todos los idiomas del proyecto que se podían configurar en WinCC flexible, como p. ej. el árabe. Si como resultado de la migración se obtiene un proyecto vacío, compruebe el idioma de edición utilizado. Los idiomas de proyecto no soportados no

pueden estar configurados como idioma de edición en el proyecto de origen. Proceda del siguiente modo:

1. Abra el proyecto con WinCC flexible.
2. Cambie el idioma de edición por inglés, por ejemplo.
3. Guarde el proyecto.
4. Reinicie la migración.

Migración de un proyecto integrado con objetos ProTool

Si se migra un proyecto de WinCC flexible integrado en STEP 7 y aparece el mensaje de error "Faltan paquetes opcionales PROTOOL en STEP 7", significa que tiene instalado WinCC flexible 2008 SP3. Además, el proyecto aún contiene objetos que se han configurado con ProTool. No abra el proyecto con WinCC flexible 2008 SP3. Para migrar el proyecto, proceda del siguiente modo:

1. Copie el proyecto en un equipo en el que esté instalado WinCC flexible 2008 SP2 y STEP 7.
2. Abra el proyecto en el SIMATIC Manager.
3. Borre todos los objetos ProTool del proyecto.
4. Ejecute el comando "Guardar como" del menú "Archivo".
5. Active la opción "Con reorganización" en el cuadro de diálogo "Guardar proyecto como".
6. Haga clic en Aceptar.
7. Copie el proyecto nuevamente en el equipo de origen.
8. Reinicie la migración.

Migración de un proyecto WinCC V7: línea de borde de rectángulos

Ha configurado un rectángulo en un proyecto WinCC V7 con los ajustes "Grosor de línea=1" y "Dibujar borde interior=sí".

Después, ha migrado el proyecto WinCC V7 a WinCC V13. Para que el rectángulo se represente correctamente, proceda del siguiente modo:

1. Abra la ventana de inspección del rectángulo.
2. Abra la lista de propiedades.
3. Desactive "Extender la línea del borde hacia dentro".

Indicador de progreso

Cuando el indicador de progreso muestra 100%, el software todavía tiene que realizar algunos trabajos residuales, como cerrar referencias, por ejemplo. Durante este tiempo, el software no reaccionará a entradas en la interfaz de usuario.

Abrir un proyecto creado con WinCC V11

Tras abrir un proyecto V11 con la versión de WinCC V13, ya no será posible abrir ese proyecto con una versión anterior.

Controles ActiveX de terceros

La migración admite controles ActiveX de terceros. No obstante, los controles deben estar registrados en el sistema operativo. Si un control ActiveX no está registrado, la migración se cancela.

Si guarda un proyecto con la herramienta de migración y realiza la propia migración en otro PC, los controles deben estar registrados también en ese PC.

Migración de proyectos integrados con visores de avisos

En un proyecto integrado, hay activado un visor de avisos con todas las categorías. Al migrar el proyecto pueden desactivarse las categorías.
Después de la migración del proyecto, compruebe los ajustes en el visor de avisos.
En caso necesario, active las categorías de aviso oportunas en la ventana de inspección del visor de avisos, en "Propiedades > General".

Migración de proyectos extensos de WinCC V7

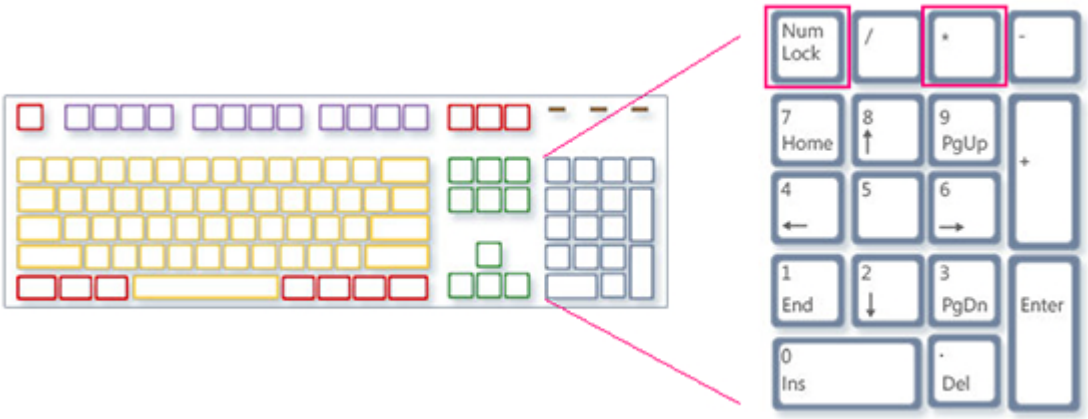
Para la migración de proyectos extensos de WinCC V7 se recomienda utilizar un sistema operativo de 64 bits.

Migrar proyectos de WinCC V7

En el TIA Portal V13 se pueden seguir utilizando proyectos de WinCC V7.2 tras una migración. Los proyectos de versiones anteriores de WinCC no se pueden migrar directamente a WinCC TIA Portal Version V13. Si desea seguir utilizando este tipo de proyectos en el TIA Portal V13, primero deberá migrar estos a WinCC V7.2 en el sistema clásico. Utilice para ello WinCC V7.2 con la última actualización.

Informe de la migración

A partir del TIA Portal V13.0 los avisos aparecen ordenados en una estructura de árbol en el informe de la migración. De este modo, todos los avisos que pertenecen a un determinado subsistema, se guardan en una misma carpeta. La consecuencia es que aumenta el número de carpetas en el árbol. Para expandir la carpeta actualmente seleccionada y todas sus subcarpetas de una vez, pulse la combinación de teclas <NUM+asterisco (*)>.



Restricciones en los datos de proyecto específicos del usuario

1. Las carpetas y archivos que haya creado el usuario en el directorio del proyecto WinCC V7.2 no se copiarán en el nuevo directorio de proyecto en la migración. Los scripts que accedan a este tipo de directorios y archivos deben adaptarse en consecuencia después de la migración.
2. Las funciones estándar en C de WinCC V7 no se migran. Si ha modificado las funciones estándar en C en el proyecto WinCC V7, deberá aplicar estas modificaciones manualmente al TIA Portal una vez realizada la migración.

Migración de textos en español (internacional) y español (tradicional)

Si un proyecto WinCC V7 incluye textos en español (tradicional), estos textos se migrarán a WinCC V13 como español (España).

Si un proyecto WinCC V7 incluye textos en español (internacional), estos textos se migrarán a WinCC V13 como español (España).

Si un proyecto WinCC V7 incluye textos tanto en español (internacional) como en español (tradicional), solo se migrarán a WinCC V13 como español (España) los textos en español (tradicional). Los textos en español (internacional) no se tendrán en cuenta.

Consulte también

Compatibilidad de objetos en la migración

2.3.5 Engineering System

2.3.5.1 Imágenes y objetos de imagen

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Objetos de visualización tras cambiar de panel de operador

Tras actualizar un equipo con una nueva versión de panel de operador, se recomienda comprobar las imágenes contenidas en el proyecto. A causa del nuevo aspecto y de las funciones de manejo mejoradas puede ocurrir, p. ej., que en los campos E/S simbólicos el texto aparezca recortado u oculto por elementos de manejo.

Copiar objetos de visualización entre dos proyectos o dos dispositivos

Así, p. ej., en el Proyecto_1 se configura una ventana de avisos en la Imagen general. Esta ventana de avisos se copia y se pega en la Imagen general del Proyecto_2.

Algunas de las categorías activadas no están activadas en la ventana de avisos después de pegar.

Este comportamiento se produce con los siguientes objetos de visualización:

- Ventana de avisos
- Indicador de avisos
- Visor de avisos

Representación de las referencias cruzadas en la ventana de inspección

Los objetos utilizados por un objeto de imagen aparecen en la ventana de inspección, en la ficha "Información > Referencia cruzada".

Hay una imagen abierta y un objeto seleccionado. Se utiliza una variable HMI en el objeto como variable de proceso.

En las referencias cruzadas aparece el objeto y la variable HMI interconectada. Además se listan todas las ubicaciones del objeto y de la variable HMI.

Si la variable HMI está interconectada con una variable PLC o una variable DB, también se muestran los lugares de uso de las variables PLC o DB interconectadas.

Denominación de los eventos en los avisos que aparecen en la ventana de inspección, en la ficha "Información".

En algunos avisos de la ventana de inspección, en la ficha "Información", se utilizan nombres de eventos que difieren del nombre en la ficha "Propiedades".

Nombre en la ventana de inspección, ficha "Propiedades"	Nombre en la ventana de inspección, ficha "Información"
Borrado	ClearScreen
Creado	GenerateScreen
Activar	Activate
Cambio	Change
Al abrir un cuadro de diálogo	ONMODALBEGIN
Al cerrar un cuadro de diálogo	ONMODALEND
Cambio de usuario	PASSWORD
Cambio de imagen	SCREEN
Desactivar	Deactivate
Pulsar	Press
Desaparecido	Going
Aparecido	Coming
Rebase por exceso del valor límite "Error superior"	AboveUpperLimit
Rebase por defecto del valor límite "Error inferior"	BelowLowerLimit
Hacer clic	Click
Loop-In-Alarm	LoopInAlarm
Soltar	Release
Desbordamiento del búfer de avisos	OVERFLOW
Acusar	Acknowledgement
Parada de runtime	Shutdown
Pulsar tecla	KeyDown
Soltar tecla	KeyUp
Conmutar ON	SwitchOn
Conmutar OFF	SwitchOff
Modificación de valor	Change value

Faceplates

Los faceplates no pueden rotarse ni voltearse.

Orden de tabulación en imágenes con faceplates

Si ha configurado una secuencia de tabulación en imágenes con faceplates en WinCC V12 o WinCC V12 SP1, es conveniente comprobar la secuencia de tabulación de dichas imágenes en WinCC V13. La secuencia de tabulación puede estar modificada tanto en la imagen como en el faceplate.

Prefijo de variable de una ventana de imagen en WinCC Runtime Professional

El prefijo de variable configurable para una ventana de imagen no es compatible con los objetos de la paleta "Controles".

Campo E/S con formato de visualización "Decimal" y formato de representación sin signo "s"

Se ha conectado una variable de proceso con un campo E/S. El formato de visualización del campo E/S es "Decimal".

Para el formato de representación se puede elegir un formato con o sin signo.

Si selecciona como "Formato de representación" un formato sin "s", p. ej., "999", esto tiene los siguientes efectos:

1. En runtime no se pueden establecer valores negativos mediante el campo E/S.
2. Si la variable adopta un valor negativo, el campo E/S crea un complemento a dos y se emite un valor positivo falseado.

Visor de curvas en Basic Panel

En los Basic Panels no se muestran los botones del visor de curvas. El visor de curvas se puede manejar con teclas de función. Asocie para ello las correspondientes funciones del sistema a las teclas de función del panel de operador.

Agrupación de objetos de imagen

Si se agrupan objetos de imagen en WinCC, pueden surgir problemas de rendimiento en WinCC si la profundidad de anidamiento es elevada.

Controles ActiveX y .NET

En runtime, los controles ActiveX y .NET siempre se sitúan en primer plano.

No se admite la configuración en niveles de controles ActiveX y .NET.

Uso de mapas de bits como iconos

En Windows 8 y Windows 8.1 no es posible usar como iconos mapas de bits de un tamaño de 48 x 48 píxeles y una intensidad de color de 32 bits.

Línea de borde de rectángulos

Ha configurado un rectángulo en un proyecto WinCC V7 con los ajustes "Grosor de línea=1" y "Dibujar borde interior=sí".

Después, ha migrado el proyecto WinCC V7 a WinCC V12. Para que el rectángulo se represente correctamente, proceda del siguiente modo:

1. Abra la ventana de inspección del rectángulo.
2. Abra la lista de propiedades.
3. Desactive "Extender la línea del borde hacia dentro".

Línea de borde del objeto "Campo E/S gráfico"

En WinCC V13, la dinamización de los colores de borde del objeto "Campo E/S gráfico" en modo "Dos estados" no tiene repercusiones en runtime.

Dinamización de instancias de un tipo de faceplate en un grupo

Utilice la instancia de un tipo de faceplate en un grupo de objetos. Las propiedades de la instancia también se muestran como propiedades del grupo. En runtime no se muestra cada dinamización con variables, scripts o animaciones del grupo.

Vista preliminar en la ventana de imagen

Utiliza diseños propios con sombras en objetos de imagen. Los objetos de imagen se visualizan en la ventana de imagen.
No obstante, la sombra de los objetos de imagen no se muestra en la vista preliminar de la ventana de imagen. Este comportamiento solo se da en el sistema de ingeniería. En runtime se visualiza correctamente.

Asignación de gráficos a teclas de función

A una tecla de función solo se le puede asignar un gráfico si el borde inferior de la ventana permanente no cubre el área del gráfico de la tecla de función.

Cambio de nombre de PLC en runtime

Si en WinCC Professional hay un PLC conectado con el visor de código PLC y el runtime está ejecutándose, cualquier modificación del nombre del PLC en tiempo de ejecución dará lugar a un error. No modifique en tiempo de ejecución el nombre del PLC, la dirección IP ni ninguna otra propiedad de la conexión HMI.

Panels y RT Advanced con la versión de panel V13: muchos objetos de imagen visualmente diferentes

El uso de objetos de imagen con muchas características visualmente diferentes (p. ej. un gran número de estilos diferentes) puede reducir el rendimiento de la interfaz de usuario en runtime y aumentar el consumo de memoria disponible. Evite, p. ej., la proliferación de redondeos de vértices distintos: 0 píxeles, 1 píxel, 2 píxeles, 3 píxeles, etc.

Si se utiliza un gran número de objetos "Indicador gauge" de distintos tamaños, puede producirse el mismo efecto. Evite los objetos "Indicador gauge" con alturas de 48 píxeles, 49 píxeles, 51 píxeles, etc. En su lugar, utilice p. ej. tamaños como: 50 píxeles, 70 píxeles, 100 píxeles.

Dinamización de objetos agrupados

En caso de existir agrupaciones anidadas a varios niveles (grupo en grupo, faceplate en grupo, grupo en faceplate, etc.), para la dinamización con funciones de sistema solo pueden utilizarse los eventos del último grupo exterior y de los últimos objetos interiores. Las funciones de sistema configuradas en eventos del grupo subordinado o del faceplate subordinado no se ejecutan.

Número de caracteres en campos de texto, listas y textos de aviso

El número de caracteres disponibles en el texto de un objeto de imagen es dinámico y depende del panel de operador y del formato de memoria. Al introducir datos de texto se consideran las instrucciones de control y el formato, y el número máximo de caracteres se reduce en consecuencia.

Transparencias en WinCC V13

Las imágenes transparentes pueden visualizarse sin pérdidas en runtime. Esto es válido para todos los Comfort Panels y WinCC Runtime Advanced con la versión 13.0.0.0.
Para usar la transparencia en un visor de gráficos o un campo E/S gráfico, la propiedad ""Patrón de relleno“ debe estar ajustada a "Transparente" y la propiedad "Utilizar color transparente“ debe estar desactivada.

Si la propiedad "Utilizar color transparente" está activada en un dispositivo con la versión 13.0.0.0, se perderá la transparencia de la imagen y las partes transparentes de la imagen se visualizarán de color negro. Después de actualizar proyectos existentes a la versión de dispositivo 13.0.0.0, se recomienda comprobar la propiedad "Utilizar color transparente“ en los casos de uso de imágenes transparentes. Si las imágenes no se visualizan correctamente, desactive esta propiedad.

2.3.5.2 Variables y conexiones

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Visualización de elementos de matriz borrados en las ubicaciones de variables HMI

En las ubicaciones de las variables HMI, p. ej. valor de proceso de campos E/S, se visualiza generalmente el nombre de la variable. Si se utiliza el elemento de una variable de matriz, se añade entre corchetes el índice del elemento al nombre de la variable.

Si una variable utilizada ya no existe en el proyecto, en la ubicación se sigue viendo el nombre de la variable. Para indicar que falta la variable, el campo aparece sobre fondo rojo. Si un elemento de matriz utilizado o la propia variable de matriz ya no existe, solo se ve entre corchetes el índice del elemento de la matriz. El nombre de la variable no se visualiza. El campo aparece sobre fondo rojo. En este caso ya no se puede identificar el nombre de la respectiva variable de matriz por medio de las ubicaciones.

Si se desconoce qué variable de matriz estaba conectada en la ubicación, puede ser necesario volver a conectar el elemento de matriz.

Si se ha creado una variable o una variable de matriz conforme a la referencia, la referencia marcada se cierra automáticamente.

Si una variable HMI está conectada con un elemento de matriz de una variable PLC y la variable PLC ya no existe en el proyecto, se produce el mismo comportamiento en el editor "Variables HMI".

Variables de matriz como entrada de la lista de variables multiplexadas

Las variables de matriz del tipo Char se pueden utilizar como variables del tipo String.

Sin embargo, no se soporta el uso de una variable de matriz del tipo Char como entrada de la lista de una variable multiplexada en el editor "Variables HMI".

Multiplexar variables en un Basic Panel

Si se multiplexa una variable con una variable externa en un Basic Panel, en el primer ciclo de lectura en runtime se leerá la dirección del PLC. El valor de la dirección leída tan solo estará disponible en el segundo ciclo de lectura.

Runtime Advanced y Panels: Importación de elementos de matrices y estructuras

Las variables de matrices y estructuras siempre se importan íntegramente, incluidos sus elementos. Los elementos de las variables de matrices y estructuras no se siguen rellenado durante la importación.

Si el nombre de una variable coincide con el nombre de un elemento de una matriz o estructura en el archivo de importación, se creará otra variable.

Ejemplo:

El archivo de importación contiene una variable de matriz llamada "Otto" con diez elementos de matriz. Por consiguiente, los elementos de la matriz tendrán el nombre Otto[1], Otto[2].

Si el archivo de importación contiene una variable de nombre "Otto[1]" no se rellenará el primer elemento de la variable de matriz sino que se creará una nueva variable en el sistema de ingeniería.

ID local de conexiones HMI

El valor para la "ID local" en las propiedades de una conexión HMI no se puede cambiar. La ID local es necesaria p. ej. para la comunicación mediante AR_SEND. Para poder utilizar la "ID local" en la comunicación proceda del siguiente modo:

1. Abra la vista de redes en el editor "Dispositivos y redes".
2. Haga clic en "Conexiones".
3. Seleccione un enlace S7.
4. En el menú contextual del PLC elija el comando "Añadir nueva conexión".
5. Haga clic en la interfaz.
6. Defina la "ID local (hex)".
7. Haga clic en "Agregar" y a continuación en "Cerrar".
8. Introduzca la dirección IP del dispositivo HMI para la nueva conexión creada en la ventana de inspección, en "Propiedades > General" en el área Interlocutor.
9. Configure las variables de datos sin formato necesarias para la comunicación en el dispositivo HMI.

Variables del tipo de datos DTL

Las variables que utilizan el tipo de datos "DTL" para elementos, únicamente pueden emplearse en modo de solo lectura.

Nombres de variables en faceplates

En los nombres de variables indicados en los tipos de faceplates no se admiten los caracteres "." y "@". No utilice estos caracteres especiales en los nombres de variables indicados en faceplates.

RT Professional: Variables con direccionamiento simbólico y tipo de datos "Char Array"

Las variables con direccionamiento simbólico y el tipo de datos "Char Array" no están habilitadas para la comunicación de RT Professional y SIMATIC S7-1200 V3.

Elementos de matriz en WinCC

Si ha conectado una variable HMI con una matriz de un bloque de datos de STEP 7 que no empieza con un límite inferior de 0, en WinCC se representan los elementos de matriz en el límite inferior de 0.

Para evitar tener que cambiar entre el índice STEP 7 y el índice WinCC al acceder a los distintos elementos de matriz, debe hacer que los límites inferiores de las matrices también empiecen en 0 en STEP 7.

2.3.5.3 Sistema de avisos y visor de avisos

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Representación de caracteres especiales en textos de aviso

En el sistema de ingeniería se utiliza un juego de caracteres para la configuración de textos de aviso. Este juego de caracteres ofrece la posibilidad de utilizar distintos caracteres especiales en los textos de aviso.

Para la representación en runtime se utilizan fuentes específicas del idioma, p. ej. MS PGothic, SimSun. Las fuentes utilizadas en runtime no soportan todos los caracteres especiales. Por ello, algunos caracteres no se visualizan en runtime.

Uso de variables multiplexadas en campos de salida en los textos de aviso

En el sistema de ingeniería también es posible utilizar variables multiplexadas en los campos de salida de los textos de aviso. Esto puede dar lugar a una representación incorrecta del aviso en runtime, ya que los Basic Panels no soportan el uso de variables multiplexadas.

Parámetros en avisos de usuario

A diferencia de lo que se dice en la Ayuda en pantalla, no es posible configurar parámetros de aviso para avisos de usuario.
La opción de menú "Propiedades > Propiedades > Parámetros de aviso" de la ventana de inspección no está disponible.

Variables en textos de aviso de RT Advanced

Las variables del tipo de datos WChar o WString no pueden visualizarse en el visor de avisos de RT Advanced.

2.3.5.4 Funciones de sistema

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Función de sistema "SimularVariable" en Basic Panels

Si se utiliza la función de sistema "SimularVariable" en combinación con un tiempo de ciclo corto en un Basic Panel, el panel de operador se puede sobrecargar.

2.3.5.5 Recetas

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Matrices en elementos de receta

Cuando se han configurado en elementos de receta una receta, una matriz y elementos de dicha matriz, se interrumpe la carga de registros con el siguiente mensaje de error: "290055: Importación de registros cancelada con errores"

Utilice para elementos de una misma receta solamente las matrices o bien solamente los elementos de matriz.

2.3.5.6 Administración de usuarios

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

2.3.5.7 Comunicación

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Interrupciones de la conexión en controladores Mitsubishi

Tras varias interrupciones de la conexión, puede ocurrir que queden ocupados todos los recursos de conexión en el controlador Mitsubishi y ya no sea posible establecer la conexión. Por ello se recomienda comprobar los recursos de conexión en el programa PLC del controlador y liberarlos nuevamente.

Número limitado de conexiones HMI posibles

Al compilar un dispositivo aparece un mensaje de error indicando que la configuración de la conexión HMI en el editor "Dispositivos y redes" no es válida. La causa puede ser que se haya excedido el número máximo de conexiones posibles del dispositivo HMI o del PLC.

Compruebe el número máximo de conexiones posibles. Consulte el manual de producto de los dispositivos utilizados.

Comunicación mediante routing con S7 300/400

La comunicación entre interlocutores de subredes diferentes es posible mediante routing con las siguientes conexiones: PROFINET, PROFIBUS, MPI.

Utilización de PROFINET IO en paneles de operador tipo Panel

Si utiliza PROFINET IO para conectar las teclas directas y LED de paneles de operador con el controlador, al efectuar la configuración en HW Config es posible definir un offset para el área de direcciones de las entradas y salidas.

Si utiliza una CPU de la serie 400 apta para PROFINET IO en combinación con uno de los paneles de operador indicados abajo, tenga en cuenta la siguiente restricción:

El offset para el inicio del área de direcciones de las entradas no puede ser mayor que el offset para el inicio del área de direcciones de las salidas.

Esta restricción rige para los siguientes paneles de operador:

- OP 177B
- OP 277
- Mobile Panel 177

Para configurar los parámetros de dirección abra en HW Config el controlador con la CPU de la serie 400. Seleccione en la ventana de la estación de HW Config el panel de operador conectado vía PROFINET IO. En el área inferior de la ventana de la estación aparece en la vista detallada una tabla con las propiedades del panel de operador. Seleccione en la tabla la fila que contiene las direcciones del panel de operador y abra las propiedades de objeto desde el menú contextual.

Active la ficha "Direcciones" en el cuadro de diálogo "Propiedades del objeto". Configure el offset para las entradas en "Entradas > Inicio". Configure el offset para las salidas en "Salidas > Inicio".

Rebase del rango de valores en Mitsubishi MC y Mitsubishi FX

Con ciertos tipos de datos, los drivers de comunicación Mitsubishi MC y Mitsubishi FX no comprueban si el valor de una variable de receta rebasa el rango de las variables del controlador. Los tipos de datos afectados son:

- 4 bit block
- 12 bit block
- 20 bit block
- 24 bit block
- 28 bit block

Puntero de área "Coordinación" en una conexión OPC

Por principio, en una conexión OPC se puede utilizar ocho veces el puntero de área "Coordinación". Si ha configurado una conexión OPC y crea automáticamente otra conexión OPC con "Agregar", el puntero de área "Coordinación" se mostrará una sola vez en la nueva conexión. En este caso es conveniente cambiar el driver de comunicación de la conexión. Si a continuación ajusta nuevamente OPC como driver de comunicación, el puntero de área "Coordinación" volverá a estar disponible ocho veces.

Cambio de configuración IP y nombre de un controlador en el Control Panel del panel de operador

En el panel de operador está abierto el Control Panel en el menú "Service and Commissoning > IP-Adaptation". Si desea cambiar la configuración IP o el nombre de un controlador, tenga en cuenta lo siguiente:

En el sistema de ingeniería ha activado previamente las siguientes opciones en la ventana de inspección del controlador, en "Propiedades > General > Interfaz PROFINET > Direcciones Ethernet":
- "Obtener dirección IP por otra vía" y
- "Obtener nombre de dispositivos PROFINET por otra vía".

Conexiones HMI en WinCC V13

Las conexiones HMI con controladores SIMATIC S7-1200 con una versión de firmware anterior a V2.0 no son posibles en WinCC V13.

Conexiones mediante PROFIBUS DP

Si se interrumpe y restablece una conexión mediante PROFIBUS DP entre un controlador y un panel de operador, el resto de conexiones PROFIBUS DP de la red de comunicación se interrumpirán y restablecerán esporádicamente.

Desconecte la tensión del dispositivo desconectado antes de volver a conectarlo.

"Establecer la (dirección) IP-Suite del PLC en el Control Panel" en SIMATIC S7-1200 V1

La función "Establecer la (dirección) IP-Suite del PLC en el Control Panel" no está habilitada en los controladores siguientes:

- SIMATIC S7-1200 V1

Cambio de una conexión

Al cambiar la conexión de un panel de operador a un controlador SIMATIC S7-300/400 por un controlador SIMATIC S7-1500 o SIMATIC S7-1200, puede interrumpirse la conexión.

Tenga en cuenta los siguientes ajustes en los controladores SIMATIC S7-1500 o SIMATIC S7-1200:

- Direccionamiento absoluto de las variables
- Debe estar seleccionada la opción "Desactivar comunicación PUT-GET"
- No debe estar configurado el nivel de protección "Protección completa"

Comunicación de datos sin formato en proyectos redundantes

Simatic.NET, Named Connections y diversos bloques de comunicación como, p. ej., BSEND/BRCV, solo pueden utilizarse de forma limitada en un equipo PC configurado de forma redundante porque los parámetros de conexión para la pareja redundante del servidor no pueden configurarse.

2.3.6 Funciones generales

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Ingeniería interproyecto: avisos del visor de diagnóstico del sistema

Si al crear datos de proxy de dispositivos no se activa "Avisos de PLC", los avisos de diagnóstico del sistema no se incluyen tampoco en los datos de proxy de dispositivos.

Después de inicializar los datos de proxy de dispositivos sin "Avisos de PLC", los avisos de diagnóstico del sistema del proxy de dispositivos no se muestran.

2.3.7 Compilar y cargar

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Compilar y cargar

Si al compilar se producen errores internos o advertencias, compile el proyecto completamente con el comando "Compilar > Software (compilar todo)" del menú contextual del panel de operador.

Antes de pasar al modo productivo con el proyecto, compílelo completamente con el comando "Compilar > Software (compilar todo)" del menú contextual del panel de operador.

Si en el proyecto se utilizan variables HMI que están vinculadas a variables del controlador, antes de compilar el panel de operador HMI compile todos los bloques modificados con el comando "Compilar > Software" del menú contextual.

Ajustes al actualizar el sistema operativo

Si selecciona desde WinCC el comando de menú "Online > Mantenimiento de paneles de operador > Actualizar sistema operativo", no podrá modificar ajustes tales como el tipo de interfaz PG/PC o la velocidad de transferencia. Se utilizan siempre los ajustes utilizados en la última carga.

Para modificar ajustes, abra el cuadro de diálogo "Carga avanzada" con "Online > Carga avanzada en dispositivo" y modifique los ajustes. Al pulsar el botón "Cargar" se guardan los ajustes modificados.

Opcionalmente se puede ejecutar con ProSave una actualización del sistema operativo con ajustes modificados. ProSave se inicia desde el menú Inicio de Windows "Siemens Automation > Options and Tools > HMI Tools > SIMATIC ProSave".

Instalación incorrecta de ProSave

Si al cargar los datos en un panel de destino o al realizar el mantenimiento del panel aparece un mensaje de error relativo a la instalación de ProSave, no se podrá eliminar el error con la función de reparación del programa de instalación. Desinstale ProSave desde el panel de control. Seguidamente, inicie el programa de instalación e instale nuevamente el componente "ProSave".

Comprobación de los parámetros de dirección

Al compilar un panel de operador en el árbol del proyecto con el comando "Compilar > Software" del menú contextual, no se comprueban los parámetros de dirección del panel de operador como, p. ej., la dirección IP. Para asegurarse de que también se comprueban los parámetros de dirección, hay que compilar el panel de operador en el editor "Dispositivos y redes" pulsando el botón "Compilar" de la barra de herramientas.

Mensaje de error al cargar datos en el PLC

Un Panel y un PLC están conectados y se comunican.

Si al cargar datos desde el Panel al PLC se accede a una variable, se visualiza un mensaje de error en el Panel.

Reacción retardada en el cuadro de diálogo "Carga avanzada"

Si en el cuadro de diálogo "Carga avanzada" los ajustes de "Tipo de interfaz PG/PC" e "Interfaz PG/PC" no coinciden con los ajustes del panel de operador, puede ocurrir que la aplicación no reaccione hasta transcurrido un minuto.

Carga avanzada en un S7-1200 y un Comfort Panel

Un PLC S7-1200 y un Comfort Panel se encuentran en la misma red física que la programadora o el PC. Ud. abre el cuadro de diálogo "Carga avanzada" para el Comfort Panel.

Si activa la opción "Mostrar todos los dispositivos accesibles" puede ocurrir que la aplicación ya no reaccione.

OP77A, OP73, TP177A: Carga de un proyecto

Al cargar un proyecto en un panel de operador puede ocurrir que aunque esté activada la opción "Remote Transfers" en el Panel no se finalice el Runtime automáticamente.

En este caso, pare el Runtime y ajuste el modo de transferencia manualmente en el panel de operador.

Cargar una aplicación SIMATIC HMI en una estación PC

En los casos siguientes puede aparecer un mensaje de error al cargar por primera vez una estación SIMATIC PC:

- en el proyecto hay una aplicación SIMATIC HMI configurada en una estación PC
  - WinCC Runtime Advanced
  - WinCC Runtime Professional
  - WinCC Standby
  - o WinCC Client
- la propiedad "S7RTM está instalado" está activada.

Antes de cargar por primera vez una estación SIMATIC PC, seleccione en el árbol del proyecto el dispositivo configurado HMI_RT (WinCC...). Abra el cuadro de diálogo "Carga avanzada" y seleccione los ajustes de interfaz y los parámetros. Pulse "Cargar".

Cargue a continuación la estación PC de la forma habitual.

Transferencia de proyectos vía USB

Si ha conectado más de un panel de operador en el PC de configuración vía USB, solo podrá transferir proyectos al último panel de operador conectado.

Abrir archivos de proyecto

Si se ejecuta el archivo "HmiIRTm.exe", se abre un cuadro de diálogo en el que se pregunta si se desea abrir un archivo de proyecto (.fwc).

Están disponibles las siguientes opciones:

- "Sí": se abre un cuadro de diálogo en el que se selecciona un archivo de proyecto (.fwc).
- "No": el cuadro de diálogo se cierra.

El simulador de variables no arranca en WinCC Professional

Si el simulador de variables no arranca en WinCC Professional, ello puede deberse a que el archivo fm20.dll todavía no está instalado en el sistema. El archivo fm20.dll no puede destribuirse y debe instalarse como componente de una aplicación de Microsoft: http://support.microsoft.com (http://support.microsoft.com/default.aspx?scid=kb;EN-US;224305)

Para instalar el archivo fm20.dll se dispone de las opciones siguientes:

- Instale una aplicación como Microsoft Office 97 en el sistema de destino. Durante la instalación se instalará también el archivo fm20.dll.
- Como alternativa también puede descargar e instalar el Microsoft ActiveX Control Pad. En este caso también se instalará el archivo fm20.dll. Para más información sobre el ActiveX Control Pad consulte el sitio web de Microsoft Developer Network: http://msdn.microsoft.com/en-us/library/ms968493.aspx (http://support.microsoft.com/default.aspx?scid=kb;EN-US;224305)

2.3.8 Runtime

2.3.8.1 Indicaciones relativas al funcionamiento en runtime

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Foco en runtime

Si en un proyecto V12 se ha configurado una combinación de color de foco y color de borde de bajo contraste, es posible que tras cambiar la versión del panel en runtime el foco no se distinga claramente. Modifique uno de los dos colores.

Comportamiento en caso de cambio de idioma - Formato del teclado de pantalla

El formato del teclado de pantalla no cambia al cambiar el idioma de Runtime.

Valores de variables que exceden la longitud máxima

Se introduce una cadena de caracteres en una variable String mediante un campo E/S. Si la cadena de caracteres excede la longitud configurada de la variable, se reducirá hasta la longitud configurada.

Textos de avisos vacíos

Runtime se ejecuta con un proyecto. El proyecto está guardado en una unidad de red.

En caso de interrupciones hacia la unidad de red puede ocurrir que Runtime intente cargar textos de avisos de la unidad de red.

Si la conexión está desconectada, la ventana o el visor de avisos permanecen vacíos.

Para evitar este comportamiento se debe copiar el proyecto en una unidad local antes de iniciar el proyecto en Runtime.

Carga completa en Service Mode

Si es necesaria una "carga completa" desde la estación de ingeniería a OS en Service Mode, se finaliza automáticamente el runtime y a continuación se vuelve a iniciar.

El proyecto deja de estar en Service Mode.

Si se interrumpe la alimentación eléctrica en este estado, ya no se inicia automáticamente el runtime en OS WinCC.

Solución:

1. Pase el proyecto a Service Mode manualmente después de haber realizado la "carga completa".
2. Finalice el proyecto manualmente
3. Active el Service Mode.
4. Vuelva a iniciar runtime mediante el icono Surrogate de la barra de tareas.

Reacción lenta del teclado de pantalla y SmartServer

Es posible que en Windows 7 y Windows 2008 Server los siguientes programas se inicien y respondan muy lentamente:

- Teclado de pantalla Microsoft OSK y HMI TouchInputPC
- SmartServer: Combinación de teclas <Ctrl+Alt+Supr> en el diálogo de inicio de sesión

El retraso es provocado por la comprobación de certificados revocados de Internet.

Solución:

En el DVD de producto encontrará bajo:
Support\Windows7\CRL_Check o CD_RT\ Support\Windows7\CRL_Check\
los siguientes archivos:

- DisableCRLCheck_LocalSystem.cmd
- DisableCRLCheck_CurrentUser.cmd

1. Ejecute el archivo "DisableCRLCheck_LocalSystem.cmd" con derechos de administrador. Llame el comando "Run as administrator" del menú contextual del archivo.
2. Reinicie el PC.

Si de esta manera no se resuelve el problema, proceda del siguiente modo:

1. Haga doble clic en el archivo y ejecute el archivo "DisableCRLCheck_CurrentUser.cmd" con derechos de usuario.
2. Reinicie el PC

Nota

La comprobación de certificados revocados se desactiva para los usuarios o el PC. Para restaurar el estado inicial ejecute los siguientes archivos:

- RestoreDefaults_LocalSystem.cmd
- RestoreDefaults_CurrentUser.cmd

Los archivos se encuentran en el siguiente directorio del DVD de producto:

- Support\Windows7\CRL_Check o CD_RT\Support\Windows7\CRL_Check\

Iniciar runtime

Desde el TIA-Portal V13 solo puede iniciarse WinCC Runtime V13. WinCC Runtime V11.02, V12 o V13 pueden simularse en el TIA-Portal V13.

2.3.8.2 Indicaciones relativas al funcionamiento de Panels en runtime

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Transferencia de licencia mediante S7USB

La transferencia de licencia a un panel mediante S7USB solo es posible desde WinCC.

Transferencia de licencias en un panel con sistemas operativos de 64 bits

Si en un sistema operativo de 64 bits no está disponible el comando de menú "Edición > Conectar sistema de destino > Conectar panel de operador" en el Automation License Manager, abra la línea de comando y ejecute el comando siguiente con derechos de administrador:

"%WINDIR%\system32\RegSvr32.exe" "%CommonProgramFiles%\siemens\AlmPanelPlugin\ALMPanelParam.dll"

Utilización de la rueda del ratón en runtime

La utilización de la rueda del ratón en runtime no se soporta en ningún panel.

Basic Panels, OP73, OP77A y TP177A: Visualización de textos en runtime

La fuente estándar seleccionada en el editor "Configuración de runtime > Idiomas & fuente" se aplica en la visualización de textos en runtime.

Si elige un tamaño o estilo de fuente inadecuado, puede ocurrir que los textos se representen cortados

Ocurre, p. ej., con los textos siguientes:

- Tooltips
- Texto de aviso largo
- Textos de cuadros de diálogo

Basic Panels: conexiones con S7-1200 y S7-1500 para Backup/Restore

En caso de utilizar la función "Backup/Restore", son posibles como máximo dos conexiones simultáneas de Basic Panels con los siguientes controladores.

- SIMATIC S7-1200
- SIMATIC S7-1500

Basic Panels: copia de seguridad en la tarjeta de memoria del PLC

Se crea el archivo de backup "A.psb" en la tarjeta de memoria del PLC. Al crear la copia de seguridad se produce un error, p. ej. una interrupción de la comunicación.
Con ello se genera un archivo defectuoso en la tarjeta de memoria del PLC. Los archivos de este tipo llevan el prefijo "~$".
Borre el archivo con el prefijo "~$" si quiere guardar una nueva copia de seguridad con el mismo nombre "A.psb".

Basic Panels: Panel Data Storage y S7-1500F

La función "Panel Data Storage" PDS no puede utilizarse en combinación con S7-1500F en Basic Panels si se utiliza la contraseña para el nivel de protección "Full access incl. fail-safe".

2.3.9 Paneles de operador

2.3.9.1 Indicaciones acerca de paneles de operador

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Si el PC pasa al estado Hibernate o Standby mientras se ejecuta la transferencia, cuando esta se interrumpa no estará definido el estado del Panel.

Manejo simultáneo de varias teclas

El manejo simultáneo de varias teclas puede provocar acciones no intencionadas:

- En caso de utilizar un equipo con teclado, no está permitido accionar más de dos teclas de función a la vez.
- En caso de utilizar un equipo táctil, un PC estándar o un Panel PC, no está permitido accionar más de una tecla de función o botón a la vez.

TS Adapter con interfaz Ethernet

Si un panel de operador está conectado a través de Ethernet y de un TS Adapter, no podrá restablecerse la configuración de fábrica.

Simulación de los Basic Panels

Para la salida de una variable externa utilice un campo de salida en un texto de aviso. En la simulación se mostrará siempre un "0" como contenido de este campo de salida.

Simulación con conexión real al controlador

El punto de acceso utilizado por la simulación es independiente de los ajustes en el Engineering System y únicamente puede modificarse con la herramienta "Ajustar interface PG/PC" en el Panel de control. Si la conexión con un controlador se deshace directamente después de arrancar la simulación y se genera el aviso 140001, debería verificar el punto de acceso utilizado por la simulación con "Ajustar interface PG/PC".

1. Haga doble clic en "Ajustar interface PG/PC" dentro del Panel de control. Se abre un cuadro de diálogo.
2. Seleccione en el área "Punto de acceso de la aplicación" "S7ONLINE" de manera estándar para HMI.
3. Seleccione la interface en el área "Parametrización de interface utilizada".
4. Finalice el cuadro de diálogo "Ajustar interface PG/PC" con Aceptar.

Carga de proyectos sin registros de receta

En un proyecto se utilizan recetas. El proyecto se transfiere a un Basic Panel pero sin los registros de receta.

Si ha modificado la estructura de la receta en el sistema de ingeniería y el dispositivo ya contenía registros de receta, pueden aparecer inconsistencias.

En tal caso, compruebe la consistencia de los registros. En el dispositivo no aparece ningún aviso en todos los casos en los que haya habido una modificación estructural.

Números en coma flotante en el MP 277, MP 377, TP 177B 4" y CP4

En los paneles de operador MP 277, MP 377, TP 177B 4" y CP4 los números en coma flotante solo se visualizan correctamente en el rango de 10^{-293} a 10^{+307}. Si el valor de la variable se encuentra fuera de este rango será visualizado como "0".

Mobile Panels V2

Los proyectos en los que se utilizan Mobile Panels V2 no pueden abrirse con WinCC V11 SP1. Esta medida afecta a proyectos con los siguientes dispositivos:

- Mobile Panel 277F IWLAN (RFID Tag)
- Mobile Panel 277F IWLAN V2
- Mobile Panel 277 IWLAN V2

Variable para la "ID de zona/de punto de acceso" de un Mobile Panel 277 IWLAN V2

La variable utilizada para la "ID de zona/de punto de acceso" debe tener el tipo de datos INT en los dispositivos Mobile Panel 277 IWLAN V2. Al migrar un proyecto procure adaptar el tipo de datos si es necesario.

Paneles de operador con el sistema operativo Windows CE 5.0 y superior

Debido a un cambio en los ajustes de seguridad en la comunicación entre cliente y servidor, la diferencia de hora entre un panel de operador (cliente) y un PC (servidor) no puede ser superior a 1 día. Si guarda una copia de seguridad p. ej. de datos de recetas del panel de operador en una unidad de red, asegúrese de que la hora del PC (servidor) y del panel de operador (cliente) esté ajustada correctamente.

Paneles de operador con elevada carga de comunicación

Si en un panel se han configurado muchas conexiones con controladores u otros paneles de operador, el diagnóstico S7 debe estar desactivado. De lo contrario, hay peligro de que el panel se sobrecargue.

Sustitución de dispositivos en el sistema de ingeniería

En el sistema de ingeniería se sustituye un dispositivo con teclado LED configurado por un dispositivo sin teclado LED. Después de la transferencia de los datos del proyecto al dispositivo no se inicia runtime.

Por eso, borre la configuración del teclado LED antes de realizar la sustitución del dispositivo.

Restricciones del panel de operador MP 377 15" Touch daylight readable

En WinCC V12 no se soportan las siguientes funciones para el panel de operador MP 377 15" Touch daylight readable:

- Opción: Sm@rtServer
- Función de sistema: DefinirYLeerBrillo
- Teclas directas

Actualización de Basic Panels a WinCC V13

Antes de actualizar Basic Panels de la versión V12 a la versión V13, transfiera a los dispositivos el archivo de imagen del V12 SP2 actualización 5 o superior.
En el cuadro de diálogo "SIMATIC ProSave [OS-Update]" elija el ajuste "Restablecer configuración de fábrica".

Con ello iniciará siempre una actualización operativa de la imagen.

Dispositivos afectados:

- KP300 Basic mono PN
- KP400 Basic color PN
- KTP400 Basic color PN

Cambio de conexión en el Control Panel para Basic Panels

Si se utiliza la función "Override projected connection information", hay que tener en cuenta la restricción siguiente:

En el Control Panel de un Basic Panel no es posible cambiar la conexión de un controlador sin nivel de protección a un controlador con el nivel de protección "Protección completa".

Mobile Panels 277F 8" IWLAN

Los Mobile Panels 277F 8" IWLAN V2 y 277F 8" IWLAN (RFID Tag) no pueden configurarse con WinCC V13.

Basic Panels de 2.ª generación

El Basic Panel KTP1200 Basic no se puede configurar actualmente con WinCC V13.

Los siguientes valores máximos son válidos para los Basic Panels de 2.ª generación.

Avisos

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de avisos de bit	1000	1000	1000	1000

Imágenes

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de objetos complejos por imagen[1)]	Esta información no es relevante para los Basic Panels de 2.ª generación.			
Número de visores de diagnóstico del sistema por imagen	5	5	5	5
Número de elementos de matriz por imagen [2)]	100	100	100	100

1) Son objetos complejos: barra, deslizador, librería de símbolos, reloj y todos los objetos del área Controles.

2) Los elementos de matriz contenidos en recetas también cuentan.

Ficheros

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de entradas por fichero (incl. todos los segmentos) [1]	10000	10000	10000	10000
Número de segmentos de fichero	400	400	400	400
Número de variables archivables por fichero	10	10	10	10
Disparador cíclico para archivar variables	1 s	1 s	1 s	1 s

Instalación 3

3.1 Indicaciones de instalación

Contenido

Indicaciones que no se han podido incluir en la Ayuda en pantalla e indicaciones importantes sobre las características del producto.

Instalación automatizada

Encontrará una descripción de la instalación automatizada en el directorio "Documents \Readme\<Directorio de idioma>" del DVD de producto.

Uso de versiones idénticas de los productos del TIA Portal

Los productos del TIA Portal STEP 7 (incl. PLCSIM), WinCC y Startdrive deben actualizarse a la misma versión (V13). Si las versiones difieren, ya no se podrá iniciar el TIA Portal. Con la desinstalación de un producto tampoco se obtiene una versión ejecutable. Antes de proceder a la instalación, asegúrese de que los Service Packs de estos productos están disponibles. Los Service Packs pueden descargarse gratuitamente de Siemens Industry Online Support (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=99&lang=es&referer=%2fWW%2f&func=cslib.csinfo2&siteid=csius&extranet=standard&viewreg=WW&groupid=4000002).

Directorio de destino de la instalación

No utilice caracteres UNICODE (p. ej. caracteres chinos) en la ruta de instalación.

Uso de programas antivirus

Durante la instalación es necesario poder acceder en lectura y escritura a los archivos ya instalados. Algunos programas antivirus lo impiden. Durante la instalación del TIA Portal se recomienda desactivar y volver a activar los antivirus.

3.2 Requisitos del sistema para la instalación

3.2.1 Indicaciones relativas a licencias

Puesta a disposición de licencias

Las licencias de los productos del TIA Portal generalmente se suministran en un soporte de datos de instalación y se transfieren automáticamente desde el Automation Licence Manager durante la instalación del TIA Portal.

Antes de desinstalar el TIA Portal es necesario transferir y realizar copia de seguridad de las licencias necesarias. Para ello, utilice el "Automation License Manager".

Preparación del Automation License Manager

El Automation License Manager se suministra en el medio de instalación y se transfiere automáticamente durante la instalación del producto.

Cuando se desinstala el TIA Portal, el Automation License Manager permanece instalado en el sistema.

Trabajar con el Automation License Manager

El Automation License Manager es un producto de Siemens AG y se utiliza para gestionar las claves de licencia (representantes técnicos de las licencias).

Los productos de software para cuyo funcionamiento se requieren claves de licencia (License Keys), como p. ej. el TIA-Portal, notifican automáticamente al Automation License Manager las claves de licencia requeridas. Si el Automation License Manager encuentra una clave de licencia válida para el software, este podrá utilizarse teniendo en cuenta las condiciones de uso asociadas a dicha clave de licencia.

Nota

Para más información sobre cómo administrar sus licencias con el Automation License Manager , consulte la documentación suministrada junto con el Automation License Manager .

Consulte también

Indicaciones relativas a los requisitos del sistema (Página 87)

Iniciar la instalación (Página 107)

Mostrar el software instalado (Página 111)

Modificar o actualizar productos instalados (Página 112)

Reparación de los productos instalados (Página 114)

Iniciar la desinstalación (Página 115)

Informe de instalación (Página 106)

3.2.2 Indicaciones relativas a los requisitos del sistema

Requisitos del sistema para los distintos productos

Los requisitos del sistema son diferentes, dependiendo de los productos que se desean instalar. Por tanto, tenga en cuenta los requisitos del sistema específicos de sus productos.

Si desea instalar varios productos, aplique los requisitos del sistema del producto que tenga las exigencias más elevadas.

Visualizar archivos PDF

Para leer los archivos PDF suministrados se requiere un lector de PDF compatible con el formato PDF 1.7, p. ej. Adobe (R) Reader versión 9.

Mostrar la Welcome Tour

Para iniciar la Welcome Tour del TIA Portal se requiere el Adobe (R) Flashplayer a partir de la versión 9.

Consulte también

Indicaciones relativas a licencias (Página 86)

Iniciar la instalación (Página 107)

Mostrar el software instalado (Página 111)

Modificar o actualizar productos instalados (Página 112)

Reparación de los productos instalados (Página 114)

Iniciar la desinstalación (Página 115)

3.2.3 Requisitos del sistema STEP 7 Professional

3.2.3.1 Concesión de licencia de STEP 7

Introducción

Para obtener la licencia de las siguientes ediciones de STEP 7 se necesita una License Key o clave de licencia:

- STEP 7 Basic
- STEP 7 Professional

La License Key o clave de licencia correspondiente para STEP 7 puede instalarse conjuntamente con la instalación o transferirse con el Automation License Manager posteriormente.

Licencias para STEP 7

Las siguientes licencias están disponibles con sus correspondientes claves de licencia:

- STEP 7 Basic
- STEP 7 Professional
- STEP 7 Professional Combo

Validez de las claves de licencia para versiones anteriores de STEP 7

Con una License Key válida para la versión V13.x de STEP 7 Professional y STEP 7 Professional Combo también se pueden utilizar otras versiones anteriores de STEP 7 sin restricciones. La siguiente tabla contiene información más detallada al respecto:

Edición	Licencia	Válido para
STEP 7 Basic V13.x	STEP 7 Basic	• STEP 7 Basic V13.x • STEP 7 Basic V12.x • STEP 7 Basic V11.x • STEP 7 Basic V10.5
STEP 7 Professional V13.x	STEP 7 Professional	• STEP 7 Professional V13.x • STEP 7 Professional V12.x • STEP 7 Professional V11.x • STEP 7 Basic V13.x • STEP 7 Basic V12.x • STEP 7 Basic V11.x • STEP 7 Basic V10.5
STEP 7 Professional V13.x	STEP 7 Professional Combo	• STEP 7 Professional V13.x • STEP 7 Professional V12.x • STEP 7 Professional V11.x • STEP 7 Basic V13.x • STEP 7 Basic V12.x • STEP 7 Basic V11.x • STEP 7 Basic V10.5 • STEP 7 V5.5 • STEP 7 V5.4 • STEP 7 Professional 2010 • STEP 7 Professional 2006

Inicio sin clave de licencia válida

Si inicia una edición de STEP 7 sin una License Key válida, el sistema le indicará que el software se está ejecutando en modo de demostración. En este caso puede activar una Trial License (licencia de prueba) una única vez. No obstante, esta solo tiene una validez limitada y caduca al cabo de 21 días.

Cuando caduca la Trial License pueden darse los casos siguientes:

- En el PC en cuestión no había existido nunca antes una licencia de STEP 7:
  - En STEP 7 ya no se pueden realizar más acciones sujetas a licencias.
- En el PC en cuestión ya había una licencia de STEP 7:
  - Cada 10 minutos y para todas las acciones sujetas a licencia aparece un aviso indicando que el software se está ejecutando en modo de demostración. Este aviso se debe confirmar.

Requerimiento de licencia para la simulación

Si inicia la simulación en STEP 7 con el comando de menú "Online > Simulación"; no necesitará ninguna otra licencia.

Si se cumplen los siguientes requisitos, para la simulación también necesitará las licencias correspondientes a la edición de STEP 7 que haya instalado:

- La estación de ingeniería está conectada a un PLC.
- La conexión con el PLC está configurada y activa.

Consulte también

Uso de licencias y claves de licencia (Página 90)

3.2.3.2 Uso de licencias y claves de licencia

Introducción

Para utilizar STEP 7 Basic y STEP 7 Professional necesitará sendas License Key válidas.

Instalación de claves de licencia

Al instalar STEP 7 Basic, la License Key necesaria se instala automáticamente. Al instalar STEP 7 Professional, una vez finalizada la instalación se le pide que transfiera la licencia desde el soporte de datos suministrado al PC.

Si desea instalar License Keys adicionales, deberá utilizar el Automation License Manager.

Al instalar una licencia, la clave de licencia en cuestión se borra de la ubicación correspondiente.

ATENCIÓN
Destrucción de claves de licencia al hacer una copia Una License Key no se puede copiar. El método de protección anticopia impide que se copien las claves de licencia. Al tratar de copiar una License Key, esta se destruye.

Desinstalación de claves de licencia

Las claves de licencia se desinstalan siempre con el Automation License Manager. Es necesario desinstalar una License Key en los siguientes casos:

- Si desea crear una copia de seguridad de los datos.
- Si ya no necesita la licencia.

También se puede utilizar una licencia válida en un PC o panel de operador diferente.

Copia de seguridad

Para crear una copia de seguridad de los datos del panel de operador, o bien al cambiar de equipo, desinstale las License Keys en el panel de operador. Para ello, abra el Automation License Manager y guarde la clave de licencia que desea desinstalar en una ubicación diferente.

ATENCIÓN
Destrucción de claves de licencia en PC Desinstale todas las claves de licencia en los casos indicados a continuación: • Antes de formatear el disco duro. • Antes de comprimir el disco duro. • Antes de restaurar el disco duro. • Antes de iniciar un programa de optimización que desplace bloques fijos. • Antes de instalar un sistema operativo nuevo. Lea la descripción del Automation License Manager ("Inicio > Siemens Automation > Documentación"). Tenga en cuenta todas las indicaciones y advertencias.

En los paneles de operador basados en PC, así como al utilizar el Automation License Manager en paneles no basados en PC, la ubicación de las claves de licencia es apta para varias licencias. Apto para varias licencias significa que es posible depositar varias licencias del mismo tipo en la misma ubicación. Al hacer la copia de seguridad, utilice una misma ubicación para todas las claves de licencia existentes en el panel de operador.

ATENCIÓN
Conserve siempre la ubicación original de las claves de licencia.

Licencia no válida tras cambiar la zona horaria

La licencia instalada dejará de funcionar en el caso siguiente.

- Si cambia la zona horaria en un PC como se indica a continuación:
  De una hora basada en la hora completa a una hora que no esté basada en la hora completa.
  Ejemplo: La zona horaria se cambia de GMT +3:00 a GMT +3:30 h.

Para evitar este comportamiento, desinstale la clave de licencia con la zona horaria que estaba ajustada al instalar la clave de licencia con el Automation License Manager.

Este comportamiento no afecta a la Trial License.

Licencia defectuosa

La licencia está defectuosa en los casos siguientes:

- Si ya no se puede acceder a la clave de licencia en su ubicación.
- Si la clave de licencia desaparece al transferirla a la unidad de destino.

La licencia defectuosa se puede reparar con ayuda del Automation License Manager. Para ello, utilice la función "Restaurar" o el "Asistente de restauración" del Automation License Manage. Para la restauración es necesario ponerse en contacto con el Customer Support.

Encontrará más información al respecto en Internet: http://support.automation.siemens.com (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=99&lang=es&referer=%2fWW%2f&func=cslib.csinfo2&siteid=csius&extranet=standard&viewreg=WW&groupid=4000002)

Consulte también

Concesión de licencia de STEP 7 (Página 88)

3.2.3.3 STEP 7 Professional para PC-CP

Funciones de STEP 7 Professional para PC-CP

STEP 7 Professional para PC-CP es una versión completa de STEP 7 Professional que se ha instalado sin licencia. En esta versión solo están disponibles las posibilidades de configuración y diagnóstico para los siguientes componentes:

- PC-CP y aplicaciones (servidor OPC) para estaciones PC
- Componentes de red y routers: SCALANCE X, SCALANCE W, SCALANCE M
- Funciones de seguridad: SCALANCE S, configuración de seguridad (SCT)
- Dispositivos PROFINET IO y esclavos PROFIBUS DP
- Lectores RF y MV

Las estaciones SIMATIC y los componentes WinAC solo están disponibles en la vista de red y de topología (no en la vista de dispositivos). Las funciones de programación no están disponibles.

Documentación

La información de sistema presente en la versión de STEP 7 Professional para PC-CP describe toda la funcionalidad de STEP 7 Professional. Por consiguiente, debe tenerse en cuenta la limitación a las funciones de configuración de los grupos de dispositivos citados anteriormente.

Reequipar a la funcionalidad de STEP 7 Professional

Con la instalación de STEP 7 Professional para PC-CP ya está disponible la funcionalidad completa de STEP 7 Professional en su estación de ingeniería, pero solo se puede utilizar con restricciones.

Por tanto, mediante una instalación de licencia a posteriori es posible en todo momento actualizar a la funcionalidad completa de STEP 7 Professional. No es necesaria una nueva instalación. Los proyectos creados previamente en STEP 7 Professional para PC-CP pueden seguir utilizándose sin restricciones tras la instalación de una licencia.

3.2.3.4 Requisitos de software y hardware de STEP 7

Requisitos mínimos de software y hardware

La tabla siguiente muestra los requisitos mínimos de software y hardware que deben cumplirse para la instalación del paquete de software "SIMATIC STEP 7 Professional":

Hardware/software	Requisitos
Procesador	Intel® Celeron® Dual Core 2,2 GHz (Ivy/Sandy Bridge)
RAM	4 GB
Memoria disponible en disco duro	5 GB
Sistemas operativos *	**Windows 7 (32 bits)** • Windows 7 Professional SP1 • Windows 7 Enterprise SP1 • Windows 7 Ultimate SP1 **Windows 7 (64 bits)** • Windows 7 Professional SP1 • Windows 7 Enterprise SP1 • Windows 7 Ultimate SP1 **Windows 8.1 (64 bits)** • Windows 8.1 Professional • Windows 8.1 Enterprise **Windows Server (64 bits)** • Windows Server 2008 R2 StdE SP1 (instalación completa) • Windows Server 2012 R2 StdE (instalación completa)
Resolución de pantalla	1024 x 768

* Encontrará más información sobre los sistemas operativos en la Ayuda de Microsoft Windows o en el sitio web de Microsoft.

Requisitos recomendados de software y hardware

La tabla siguiente muestra el software y hardware recomendado para el uso de STEP 7.

Hardware/software	Requisitos
Equipo	SIMATIC FIELD PG M4 PREMIUM o superior (o PC comparable)
Procesador	Intel® Core™ i5-3320M 3,3 GHz o superior
RAM	8 GB o más
Disco duro	300 GB SSD
Pantalla	Pantalla Wide Screen de 15,6" (1920 x 1080)
Sistemas operativos *	**Windows 7 (64 bits)** • Windows 7 Professional SP1 • Windows 7 Enterprise SP1 • Windows 7 Ultimate SP1 **Windows 8.1 (64 bits)** • Windows 8.1 • Windows 8.1 Professional • Windows 8.1 Enterprise **Windows Server (64 bits)** • Windows Server 2008 R2 StdE SP1 (instalación completa) • Windows Server 2012 R2 StdE (instalación completa)

* Encontrará más información sobre los sistemas operativos en la Ayuda de Microsoft Windows o en el sitio web de Microsoft.

Plataformas de virtualización compatibles

El paquete de software "SIMATIC STEP 7 Professional" puede instalarse en una máquina virtual. Para ello, utilice una de las siguientes plataformas de virtualización en la versión indicada o una superior:

- VMware vSphere Hypervisor (ESXi) 5.5
- VMware Workstation 10
- VMware Player 6.0
- Microsoft Windows Server 2012 R2 Hyper-V

Los siguientes sistemas operativos pueden servir de sistema operativo host para dichas plataformas de virtualización:

- Windows 7 Professional/Ultimate/Enterprise (64 bits)
- Windows Server 2008 R2 (64 bits)
- Windows Server 2012 R2 (64 bits)
- Windows 8.1 Professional/Enterprise (64 bits)

Dentro de la plataforma de virtualización seleccionada pueden utilizarse los siguientes sistemas operativos huésped para instalar "SIMATIC STEP 7 Professional":

- Windows 7 Professional/Ultimate/Enterprise (64 bits)
- Windows 8.1 Professional/Enterprise (64 bits)

Nota

- Para los sistemas operativos huésped valen los mismos requisitos de hardware que para los propios productos TIA.
- El operador de la instalación debe asegurarse de que los sistemas operativos huésped dispongan de suficientes recursos de sistema.
- Para el uso de servidores HyperV y ESXi se recomienda hardware certificado por los fabricantes.
- Si se utiliza Microsoft Hyper-V, no es posible mostrar los dispositivos accesibles.

Programas de seguridad compatibles

Los siguientes programas de seguridad son compatibles con "SIMATIC STEP 7 Professional":

- Programas antivirus:
  - Symantec Endpoint Protection 12.1
  - Trend Micro Office Scan Corporate Edition 10.6
  - McAfee VirusScan Enterprise 8.8
  - Kaspersky Anti-Virus 2014
  - Windows Defender (Windows 8.1 o superior)
- Software de encriptación:
  - Microsoft Bitlocker
- Intrusion Detection System basado en host:
  - McAfee Application Control 6.0

3.2.4 Requisitos del sistema en WinCC Basic

3.2.4.1 Requisitos de software y hardware

Introducción

Para la instalación se deben cumplir determinados requisitos en cuanto al sistema operativo y la configuración por software.

Nota

WinCC está autorizado básicamente para el funcionamiento en un dominio o grupo de trabajo.

Sin embargo, tenga en cuenta que las directivas de grupos y las limitaciones del dominio pueden impedir la instalación. En este caso elimine el equipo del dominio antes de la instalación de Microsoft Message Queuing, Microsoft SQL Server 2005 y WinCC. Inicie la sesión con derechos de administrador en el equipo en cuestión. Realice la instalación. Luego de instalar correctamente el software es posible reincorporar el equipo WinCC al dominio. Si las directivas de grupos y las limitaciones del dominio no impiden la instalación, no es necesario eliminar el equipo del dominio durante la instalación.

Tenga en cuenta que las directivas de grupos del dominio y las limitaciones del dominio pueden impedir el funcionamiento. Si no es posible eliminar estas limitaciones utilice el equipo WinCC en un grupo de trabajo.

Dado el caso, diríjase al administrador del dominio.

Requisitos para la instalación

La tabla siguiente muestra los requisitos mínimos de software y hardware que deben cumplirse para la instalación del paquete de software "SIMATIC WinCC Basic":

Hardware/software	Requisitos
Tipo de procesador	Intel® Celeron® Dual Core 2,2 GHz (Ivy/Sandy Bridge)
RAM	4 GB
Espacio libre en el disco duro	5 GB

Hardware/software	Requisitos
Sistemas operativos *	**Windows 7 (32 bits)** • Windows 7 Home Premium SP1 • Windows 7 Professional SP1 • Windows 7 Enterprise SP1 • Windows 7 Ultimate SP1 **Windows 7 (64 bits)** • Windows 7 Home Premium SP1 • Windows 7 Professional SP1 • Windows 7 Enterprise SP1 • Windows 7 Ultimate SP1 **Windows 8 (64 bits)** • Windows 8.1 • Windows 8.1 Professional • Windows 8.1 Enterprise **Windows Server (64 bits)** • Windows Server 2012 R2 Standard Edition
Resolución de pantalla	1024x768
Red	A partir de Ethernet 10 Mbits/s
Unidad óptica	DVD-ROM
Software	Microsoft .Net Framework 4.5

* Encontrará más información sobre los sistemas operativos en la Ayuda de Microsoft Windows o en el sitio web de Microsoft.

En caso de abrir simultáneamente varias instancias de WinCC en un PC de configuración, pueden ser necesarios requisitos de hardware más altos.

Nota

"Aero Glass Style" de Microsoft Windows 7

Para "Aero Glass Style" se requiere una tarjeta gráfica potente. Los requisitos son la compatibilidad con DirectX9 y una memoria gráfica extra de 128 MB.

El rendimiento de la arquitectura del sistema gráfico puede influir notablemente en el rendimiento de WinCC.

Hardware recomendado

La tabla siguiente muestra el hardware recomendado para el uso de SIMATIC WinCC.

Hardware	Requisitos
Equipo	SIMATIC FIELD PG M4 PREMIUM o superior (o PC comparable)
Procesador	Intel® Core™ i5-3320M 3,3 GHz o superior
RAM	8 GB o más
Disco duro	300 GB SSD

Hardware	Requisitos
Pantalla	Pantalla Wide Screen de 15,6" (1920 x 1080)
Unidad óptica	DL MULTISTANDARD DVD RW

Plataformas de virtualización compatibles

El paquete de software "SIMATIC WinCC Basic" puede instalarse en una máquina virtual. Para ello, utilice una de las siguientes plataformas de virtualización:

- VMware vSphere Hypervisor (ESXi) 5.5
- VMware Workstation 10
- VMware Player 6.0
- Microsoft Windows Server 2012 R2 Hyper-V

Los siguientes sistemas operativos pueden servir de sistema operativo Host para dichas plataformas de virtualización:

- Windows 7 Professional/Ultimate/Enterprise (64 bits)
- Windows 8.1 Professional/Enterprise (64 bits)
- Windows Server 2008 R2 (64 bits)
- Windows Server 2012 R2 (64 bits)

Dentro de la plataforma de virtualización seleccionada pueden utilizarse los siguientes sistemas operativos huésped para instalar "SIMATIC WinCC Basic":

- Windows 7 Professional/Ultimate/Enterprise (64 bits)
- Windows 8.1 Professional/Enterprise (64 bits)

Nota

- Para los sistemas operativos huésped valen los mismos requisitos de hardware que para los propios productos TIA.
- El operador de la instalación debe asegurarse de que los sistemas operativos huésped dispongan de suficientes recursos de sistema.
- Para el uso de servidores HyperV y ESXi se recomienda hardware certificado por los fabricantes.

Programas de seguridad compatibles

Los siguientes programas de seguridad son compatibles con "SIMATIC WinCC Basic":

- Programas antivirus:
  - Symantec Endpoint Protection 12.1
  - Trend Micro Office Scan Corporate Edition 10.6
  - McAfee VirusScan Enterprise 8.8
  - Kaspersky Anti-Virus 2014
  - Windows Defender (Windows 8.1 o superior)
- Software de encriptación:
  - Microsoft Bitlocker
- Intrusion Detection System basado en host:
  - McAfee Application Control 6.0

Instalar Microsoft .Net Framework

El software necesita .Net Framework 4.5, que se instala y activa automáticamente.

Se le pedirá instalar la versión de .Net Framework necesaria en caso de que esto no pueda realizarse con la instalación del paquete de software. Una vez instalado .Net Framework puede ser necesario reiniciar el equipo.

Ayuda en pantalla de Windows 7 / Windows Server 2008

Windows 7 y Windows Server 2008 ya no soportan de forma estándar todos los formatos de Ayuda en pantalla. En WinCC estos formatos de Ayuda en pantalla se utilizan en los casos siguientes:

- Llamada de la ayuda directa de WinCC
- Llamada del WinCC Information System desde los editores de WinCC o por medio de los enlaces de la ayuda directa

Para poder seguir llamando la ayuda directa de WinCC, durante la instalación de WinCC se instala el siguiente componente:

- Microsoft Help Engine

El WinCC Information System también se puede abrir en Windows 7 y Windows Server 2008 desde el menú Inicio de Windows o desde la carpeta de instalación.

Para llamar el WinCC Information System desde los editores de WinCC o por medio de los enlaces de la ayuda directa es preciso adaptar el sistema operativo. Encontrará más información al respecto en el apartado "Más información para usuarios avanzados" del artículo del Microsoft Support con la ID "917607": http://support.microsoft.com/kb/917607 (http://support.microsoft.com/kb/917607)

Ajustes de seguridad durante la instalación

Durante la instalación de WinCC V13 se modifican ajustes de seguridad en el sistema operativo.

Durante la instalación aparece una lista de los ajustes de seguridad afectados.

La modificación de los ajustes de seguridad tiene que confirmarse.

Si después de finalizada la instalación realiza modificaciones en el sistema operativo, los cambios en los ajustes de seguridad pueden verse modificados por la instalación del TIA Portal.

Los cambios en los ajustes de seguridad debidos a la instalación del TIA Portal pueden restaurarse:
"Inicio > Todos los programas > Siemens Automation > Security Controller > Restablecer Configuración".

Instancia SQL de WinCC V13

Si ha instalado un producto WinCC V11 y desea instalar WinCC V13, antes de la instalación deberá desinstalar la instancia WinCC del SQL Server 2005.

Con la instalación de WinCC V13 se instala una nueva instancia de WinCC SQL 2008.

Consulte también

Concesión de licencias del WinCC Engineering System (Página 101)

3.2.4.2 Instalaciones paralelas

Instalaciones paralelas en el TIA Portal V13

Si se realiza una instalación paralela no permitida de STEP 7 y WinCC, el TIA Portal no se podrá iniciar. Las siguientes instalaciones paralelas están permitidas en el TIA Portal:

- STEP 7 V13 y WinCC V13

Durante la instalación aparece un cuadro de diálogo indicando que la instalación paralela no es conforme. Las siguientes instalaciones paralelas están permitidas:

- WinCC V13 y RT Advanced V13
- WinCC V13 y RT Professional V13

El sistema de ingeniería y runtime deben tener siempre la misma versión tras una instalación.

Instalación paralela de WinCC V13 y otros productos SIMATIC HMI

No está permitida la instalación paralela de WinCC V13 con versiones de WinCC flexible anteriores a WinCC flexible 2008 SP1.

No está permitida la instalación paralela de WinCC V13 con versiones de WinCC anteriores a WinCC V7.0 SP2. No está permitida la instalación paralela de WinCC V13 con WinCC V7.0 SP2 o WinCC V7.0 SP3 para:

- WinCC V 13 Professional
- WinCC V 13 Runtime Professional

Uso paralelo

Si después de la instalación, el nombre o la clave de licencia del software llevasen el indicativo "Combo", estará autorizado para utilizar los siguientes productos y versiones según el apartado 1.6 de las Condiciones generales (vea también el texto de la instalación):

- Con la licencia "WinCC V13 Comfort Combo": WinCC flexible 2008 Standard
- Con la licencia "WinCC V13 Advanced Combo": WinCC flexible 2008 Advanced

3.2.4.3 Licencias y power packs

Concesión de licencias del WinCC Engineering System

Se requiere una clave de licencia para:

- WinCC Engineering System, p. ej. WinCC Professional
- Opciones del WinCC Engineering System

La clave de licencia de WinCC se puede instalar al mismo tiempo que se instala WinCC. Las licencias para las opciones de WinCC se deben transferir después de la instalación con el Automation License Manager.

Inicio sin clave de licencia válida

Si inicia WinCC sin una clave de licencia válida, el sistema le indicará que el software se está ejecutando en modo de demostración. En este caso puede activar una Trial License (licencia de prueba) una única vez. En las ediciones de ingeniería de WinCC Basic, Comfort/Advanced y Professional la Trial License caduca al cabo de 21 días.

Cuando caduque la Trial License pueden darse los casos siguientes:

- En el PC en cuestión no había existido nunca antes una licencia de WinCC.
  - En WinCC ya no se pueden realizar más acciones sujetas a licencias.
- En el PC en cuestión había existido una licencia de WinCC.
  - Cada 10 minutos y para todas las acciones sujetas a licencia aparece un aviso indicando que el software se está ejecutando en modo de demostración. Este aviso se debe confirmar.

Requerimiento de licencia para la simulación

Si inicia la simulación con el comando de menú "Online > Simulación > Con simulador de variables" en WinCC, no necesitará licencias para WinCC Runtime o para las opciones sujetas a licencia.

Si se cumplen los siguientes requisitos se necesitarán también para la simulación las licencias correspondientes para WinCC Runtime y para las opciones sujetas a licencia:

- La Engineering Station está conectada con un PLC.
- La conexión con el PLC está configurada y activa.

Inicie el simulador con el comando de menú "Online > Simulación > Iniciar".

Consulte también

Requisitos de software y hardware (Página 96)

Licenciar el software en los paneles de operador (Página 102)

Uso de las claves de licencia (Página 103)

Licenciar el software en los paneles de operador

Los paneles de operador no basados en PC se equipan siempre con "licencias de runtime". Para el funcionamiento en runtime no se requiere una clave de licencia.

Para una opción de un panel de operador no basado en PC puede ser necesaria una licencia. La clave de licencia de la licencia en cuestión autoriza a utilizar una opción en cada caso.

Clave de licencia

Para poder licenciar el software en paneles de operador no basados en PC se requiere el add-on "SIMATIC HMI License Manager Panel Plugin".

Este add-on se instala de forma estándar durante la instalación de WinCC. El License Manager Panel Plugin se abre en el Automation License Manager con el comando de menú "Edición > Conectar sistema de destino > Conectar panel de operador".

Si WinCC no está instalado, es necesario instalar ProSave 7.2 o superior.

Nota

Para más información sobre el uso de las licencias, consulte la Ayuda en pantalla del Automation License Manager.

Nota

Antes de licenciar el software, asegúrese de que en el panel de operador esté instalada la versión actual del sistema operativo. De lo contrario, actualice el sistema operativo utilizando ProSave.

Modo de demostración

Sin una licencia válida, las opciones de runtime están disponibles temporalmente sin limitaciones de funcionamiento. Cada 10 minutos aparece un aviso indicando que el software se está ejecutando en modo de demostración. Este aviso se debe confirmar.

Consulte también

Concesión de licencias del WinCC Engineering System (Página 101)

Uso de las claves de licencia

Introducción

En los siguientes casos se transfiere una clave de licencia al panel de operador:

- Para utilizar el WinCC Engineering System
- Para utilizar WinCC Runtime
- Para utilizar las opciones de WinCC Runtime en paneles de operador basados en PC
- Para utilizar las opciones en paneles de operador no basados en PC

En los siguientes casos se transfiere una clave de licencia desde el panel de operador:

- Si desea crear una copia de seguridad de los datos
- Si ya no necesita la licencia

Esa licencia se puede utilizar entonces en un PC o panel de operador diferente.

Cuando se transfiere una licencia a un panel de operador, la clave de licencia en cuestión se borra de la ubicación de las claves de licencia.

Nota

Una clave de licencia no se puede copiar. El método de protección anticopia impide que se copien las claves de licencia.

Copia de seguridad

Para crear una copia de seguridad de los datos del panel de operador, o bien al cambiar de equipo, transfiera las claves de licencia del panel de operador.

Las claves de licencia de un panel de operador se guardan con el Automation License Manager en la ubicación de las claves de licencia.

ATENCIÓN

Destrucción de claves de licencia en paneles de operador no basados en PC

Las operaciones de Backup/Restore destruyen las claves de licencia transferidas en los siguientes paneles de operador.

- Serie 270
- Serie 370

Antes de comenzar con la restauración, lleve a cabo los siguientes pasos:

- Compruebe con el Automation License Manager y ProSave, si el panel de operador tiene instaladas claves de licencia.
- Transfiera las claves de licencia existentes en el panel de operador a una ubicación. Después de la restauración vuelva a transferir las claves de licencia al panel de operador.

ATENCIÓN

Destrucción de claves de licencia en PC

Transfiera en primer lugar todas las claves de licencia a una ubicación en los siguientes casos:

- Antes de formatear el disco duro
- Antes de comprimir el disco duro
- Antes de restaurar el disco duro
- Antes de iniciar un programa de optimización que desplace bloques fijos
- Antes de instalar un sistema operativo nuevo

Lea la descripción del Automation License Manager ("Inicio > Siemens Automation > Documentación"). Tenga en cuenta todas las indicaciones y advertencias.

En los paneles de operador basados en PC, así como al utilizar el Automation License Manager en paneles no basados en PC, la ubicación de las claves de licencia es apta para varias licencias. Apto para varias licencias significa que es posible depositar varias licencias del mismo tipo en la misma ubicación. Al hacer la copia de seguridad, utilice una misma ubicación para todas las claves de licencia existentes en el panel de operador.

Nota

Conserve en todo caso la ubicación original de las claves de licencia.

Licencia no válida tras cambiar la zona horaria

La licencia transferida dejará de funcionar en el siguiente caso:

- Si cambia la zona horaria en un PC con WinCC como se indica a continuación:
  - De una hora basada en la hora completa a una hora que no esté basada en la hora completa.
    Ejemplo: La zona horaria se cambia de GMT +3:00 a GMT +3:30 h.

Para evitar este comportamiento, transfiera la clave de licencia desde el panel de operador con el mismo ajuste de zona horaria que en el momento de la transferencia de la clave de licencia.

Ejemplo:

Ha transferido al panel de operador la clave de licencia con un ajuste de la zona horaria basado en la hora completa. En este caso, transfiera la clave de licencia del panel de operador también con un ajuste de zona horaria basado en la hora completa.

Este comportamiento no afecta a la Trial License.

Licencia defectuosa

La licencia está defectuosa en los casos siguientes:

- Si ya no se puede acceder a la clave de licencia en su ubicación.
- Si la clave de licencia desaparece al transferirla a la unidad de destino.

Nota

Si se ajusta la fecha del sistema a un momento anterior en el tiempo, todas las licencias se volverán defectuosas.

La licencia defectuosa se puede reparar con ayuda del Automation License Manager. Utilice para ello la función "Restaurar", o bien el "Asistente para la restauración" del Automation License Manager. Para restaurar las licencias es preciso ponerse en contacto con el Customer Support. Encontrará más información al respecto en: http://support.automation.siemens.com (http://support.automation.siemens.com)

Nota

El software Runtime puede seguir funcionando correctamente aunque la licencia falte o esté defectuosa. En intervalos breves, el sistema emite avisos indicando que el software se está ejecutando en modo de demostración.

Nota

Si inicia el WinCC Engineering System sin una clave de licencia válida, el sistema le indicará que el software se está ejecutando en modo de demostración. Existe la posibilidad de activar una licencia de prueba una única vez. La licencia de prueba (Trial License) caduca al cabo de 21 días.

Cuando caduque la Trial License pueden darse los casos siguientes:

- En el PC en cuestión no había existido nunca antes una licencia de WinCC.
  WinCC ya no se podrá iniciar.
- En el PC en cuestión había existido una licencia de WinCC.
  WinCC se podrá iniciar. Cada 10 minutos aparece un aviso indicando que el software se está ejecutando en modo de demostración. Este aviso se debe confirmar.

Consulte también

Concesión de licencias del WinCC Engineering System (Página 101)

3.3 Informe de instalación

Función del informe de instalación

Durante los siguientes procesos de instalación se registran los progresos en un archivo:

- Instalar productos
- Modificar o actualizar productos ya instalados
- Reparar una instalación existente
- Desinstalar productos

En caso de producirse algún error durante el proceso de instalación o de emitirse una advertencia, los errores o advertencias se pueden evaluar con ayuda del archivo de registro. La evaluación puede realizarla Ud. mismo o bien, diríjase al Product Support.

Ubicación de los informes de instalación

El archivo de informe es el archivo más reciente con la extensión ".log" cuyo nombre empiece por "SIA".

La ubicación del archivo de registro está guardada en la variable de entorno "%autinstlog%". Esta variable se puede introducir en la barra de dirección del Windows Explorer para abrir la carpeta que contiene los archivos de registro. También existe la alternativa de navegar hasta el directorio correspondiente escribiendo "CD %autinstlog%" en la línea de comandos.

La ubicación depende del sistema operativo utilizado; por ejemplo, "C:\Documents and Settings\All Users\Application Data\Siemens\Automation\Logfiles\Setup" en la versión en inglés de Windows XP.

Setup_Report (archivo CAB)

Para poder entregar todos los archivos necesarios al Product Support, se crea adicionalmente un archivo comprimido en formato CAB que contiene el informe de instalación y todos los demás archivos requeridos. Este archivo comprimido se encuentra en "%autinstlog%\Reports \Setup_report.cab". Si necesita ayuda para la instalación, envíe el archivo CAB al Product Support. De este modo, el Product Support podrá determinar rápidamente si la instalación se ha ejecutado correctamente. Los archivos CAB generados en instalaciones anteriores quedan guardados en el directorio "Reports" con un identificativo de la fecha.

Consulte también

Indicaciones relativas a licencias (Página 86)

Iniciar la instalación (Página 107)

Comprobar disponibilidad de actualizaciones y Support Packages y proceder a su instalación (Página 109)

Mostrar el software instalado (Página 111)

Modificar o actualizar productos instalados (Página 112)

Reparación de los productos instalados (Página 114)

Iniciar la desinstalación (Página 115)

3.4 Iniciar la instalación

Introducción

Una aplicación setup instala automáticamente los paquetes de software. La aplicación se inicia al insertar el medio de instalación en la unidad de disco correspondiente.

Requisitos

- El hardware y software de la programadora o PC cumplen los requisitos del sistema.
- Usted dispone de derechos de administración para su equipo.
- Se han cerrado todas las aplicaciones que se estaban ejecutando.

Procedimiento

Para instalar los paquetes de software, proceda del siguiente modo:

1. Inserte el medio de instalación en la unidad de disco correspondiente.
   La aplicación Setup arranca automáticamente si el inicio automático no está desactivado en la programadora o PC.
2. Si el programa de instalación no se inicia automáticamente, hágalo manualmente haciendo doble clic en el archivo "Start.exe".
   Se abre el diálogo para seleccionar el idioma de instalación.

3. Seleccione el idioma en el que desea visualizar los diálogos del programa de instalación.
4. Para leer las instrucciones del producto y de la instalación, haga clic en el botón "Leer indicaciones" o "Indicaciones de instalación".
   Se abre el archivo de ayuda correspondiente con las indicaciones.
5. Tras haberlas leído, cierre el archivo de ayuda y haga clic en el botón "Siguiente".
   Se abre el diálogo para seleccionar los idiomas del producto.
6. Seleccione los idiomas para la interfaz del producto y haga clic en el botón "Siguiente".

   Nota

   El idioma base de instalación del producto es siempre el inglés.

   Se abre el diálogo para seleccionar la configuración de producto.
7. Seleccione los productos que desea instalar:
   – Si sólo desea una instalación mínima del programa, haga clic en el botón "Mínima".
   – Si quiere instalar los componentes típicos del programa, haga clic en el botón "Típica".
   – Si prefiere elegir Ud. mismo los componentes que se van a instalar, haga clic en el botón "Personalizada". En este último caso, active a continuación las casillas de verificación correspondientes a los productos que desea instalar.
8. Si desea crear un acceso directo en el escritorio, active la casilla de verificación "Crear acceso directo en el escritorio".
9. Haga clic en el botón de comando "Examinar" si desea cambiar el directorio de destino para la instalación. Tenga en cuenta que la longitud del nombre de la ruta de instalación no debe exceder 89 caracteres.
10. Haga clic en el botón "Siguiente".
    Se abre el diálogo correspondiente a las condiciones de licencia.
11. Para continuar la instalación, lea y acepte todos los acuerdos de licencia y haga clic en "Siguiente".
    En caso de que sea necesario modificar opciones de seguridad y permisos para la instalación del TIA Portal, se abrirá el diálogo para la configuración de seguridad.
12. Para continuar la instalación, acepte todos los cambios efectuados en las opciones de seguridad y permisos y haga clic en "Siguiente".
    En el siguiente diálogo aparecen listados los ajustes de instalación.
13. Compruebe los ajustes de instalación seleccionados. Si desea efectuar cambios, haga clic en el botón "Atrás" hasta acceder al ajuste que desea modificar. Una vez efectuados los cambios deseados, regrese a la vista general con "Siguiente".

14. Haga clic en el botón "Instalar".
    Se inicia la instalación.

    Nota

    Si no se encuentra ninguna clave de licencia (License Key) durante la instalación, existe la posibilidad de transferir ésta al PC. Si se omite la transferencia de licencias, este paso puede realizarse posteriormente con el Automation License Manager.

    Tras la instalación aparece un aviso indicando si la instalación se ha completado correctamente.

15. Es posible que deba reiniciar el equipo. En ese caso, active el botón de opción "Sí, deseo reiniciar mi equipo ahora". A continuación, haga clic en el botón "Reiniciar".
16. Si no es preciso volver a iniciar el equipo, haga clic en el botón "Finalizar".

Resultado

El TIA Portal, los productos y licencias que ha adquirido, así como el Automation License Manager se han instalado en el equipo.

Consulte también

Informe de instalación (Página 106)

Indicaciones relativas a los requisitos del sistema (Página 87)

Indicaciones relativas a licencias (Página 86)

Mostrar el software instalado (Página 111)

Modificar o actualizar productos instalados (Página 112)

Reparación de los productos instalados (Página 114)

Iniciar la desinstalación (Página 115)

3.5 Comprobar disponibilidad de actualizaciones y Support Packages y proceder a su instalación

En el TIA Portal tiene la posibilidad de comprobar si hay nuevas actualizaciones de software o nuevos Support Packages, p. ej. Hardware Support Packages (HSP). En caso afirmativo, puede instalar el software a continuación.

Nota

Se admiten actualizaciones y Support Packages a partir del TIA Portal V13.

Procedimiento

Para comprobar la disponibilidad de actualizaciones de software y Support Packages y proceder a su instalación, haga lo siguiente:

1. Inicie el proceso de instalación de una de las dos siguientes formas:
   - En el TIA Portal, haga clic en "Software instalado" en el menú "Ayuda".
     Se abre el diálogo "Software instalado".
   - Abra el programa "Siemens TIA Updater" mediante el acceso directo de su PC.
     Se abrirá el cuadro de diálogo "Siemens TIA Updater" y se mostrarán los paquetes de software disponibles.
2. Haga clic en "Comprobar actualizaciones". Si ya aparece el cuadro de diálogo "Siemens TIA Updater", este paso es opcional.
   Una vez iniciado el proceso de instalación en el TIA Portal, se abre el cuadro de diálogo "Siemens TIA Updater" y se muestran las actualizaciones disponibles.
3. Haga clic en "Descargar" en la línea de la actualización o del Support Package que desee instalar.
   Se descargará la actualización o el Support Package. Cuando finalice el proceso de descarga, se activará el correspondiente botón "Instalar".

   Nota

   Tenga en cuenta las indicaciones siguientes:
   1. Puede iniciar simultáneamente varios procesos de descarga.
   2. Se puede cerrar sesión o incluso apagar el PC mientras está en curso una descarga. En estos casos, el proceso de descarga continuará en segundo plano cuando vuelva a iniciar sesión.
4. Cierre el TIA Portal, en caso de que aún esté abierto.
5. En el cuadro de diálogo "Siemens TIA Updater", haga clic en el botón "Instalar" del paquete de software que desee instalar.
   Aparecerá el cuadro de diálogo de instalación.

   Nota

   Tenga en cuenta las indicaciones siguientes:
   1. No es posible instalar varias actualizaciones al mismo tiempo.
   2. No cierre sesión ni apague el PC durante una instalación. De este modo evitará versiones de software inconsistentes en su PC.
6. Haga clic en "Siguiente".
   Se instalará el producto seleccionado.

Procedimiento alternativo de instalación de Support Packages

Existe la posibilidad de seguir otro procedimiento para instalar un Support Package. Proceda para ello del siguiente modo:

1. En el TIA Portal, haga clic en "Support Packages" en el menú "Herramientas".
   Se abre el diálogo "Información detallada". En la tabla se listan todos los Support Packages del directorio que se definieron en la configuración como ubicación de los Support Packages.
2. Si desea instalar un Support Package que no aparece en la lista, existen las siguientes posibilidades:
   – Si el Support Package se encuentra ya en el equipo, puede hacerse clic en "Agregar del sistema de archivos" para incluirlo en la lista.
   – Si se desea agregar un Support Package de la página "Service & Support" de Internet, en primer lugar es preciso descargarlo haciendo clic en "Descargar de Internet". A continuación, lo puede agregar del sistema de archivos.
3. Seleccione el Support Package que desee instalar.
4. Haga clic en "Instalar".
5. Cierre el TIA Portal y reinícielo.

Consulte también

Informe de instalación (Página 106)

3.6 Mostrar el software instalado

En todo momento se puede obtener información sobre los productos de software que están instalados. También es posible visualizar más información sobre el software instalado.

Procedimiento

Para obtener una vista de los productos de software instalados, proceda del modo siguiente:

1. Haga clic en "Software instalado" del menú "Ayuda".
   Se abre el diálogo "Software instalado". En el diálogo se ven los productos de software instalados. Expanda las entradas para ver la versión instalada, respectivamente.
2. Si desea ver más información sobre el software de automatización que está instalado, haga clic en el enlace ubicado en el diálogo "Más información acerca del software instalado".
   Se abre el diálogo "Información detallada".
3. Elija en la navegación local el tema del que desee obtener información adicional.

Consulte también

Indicaciones relativas a los requisitos del sistema (Página 87)

Indicaciones relativas a licencias (Página 86)

Iniciar la instalación (Página 107)
Modificar o actualizar productos instalados (Página 112)
Reparación de los productos instalados (Página 114)
Iniciar la desinstalación (Página 115)
Informe de instalación (Página 106)

3.7 Modificar o actualizar productos instalados

A través del programa de instalación existe la posibilidad de modificar los productos instalados o bien actualizarlos a una nueva versión (update).

Requisitos

- El hardware y software de la programadora o PC cumplen los requisitos del sistema.
- Usted dispone de derechos de administración para su equipo.
- Se han cerrado todas las aplicaciones que se estaban ejecutando.

Procedimiento

Para modificar o actualizar los productos instalados, proceda del siguiente modo:

1. Inserte el medio de instalación en la unidad de disco correspondiente.
   La aplicación Setup arranca automáticamente si el inicio automático no está desactivado en la programadora o PC.
2. Si el programa de instalación no se inicia automáticamente, hágalo manualmente haciendo doble clic en el archivo "Start.exe".
   Se abre el diálogo para seleccionar el idioma de instalación.
3. Seleccione el idioma en el que desea visualizar los diálogos del programa de instalación
4. Para leer las instrucciones del producto y de la instalación, haga clic en el botón "Leer indicaciones" o "Indicaciones de instalación".
   Se abre el archivo de ayuda correspondiente con las indicaciones.
5. Tras haberlas leído, cierre el archivo de ayuda y haga clic en el botón "Siguiente".
   Se abre el diálogo para seleccionar la variante de instalación.
6. Active el botón de opción "Modificar/Actualizar" y haga clic en el botón "Siguiente".
   Se abre el diálogo para seleccionar los idiomas del producto.
7. Active la casilla de verificación correspondiente a los idiomas de producto que desea instalar. Los idiomas de producto ya instalados pueden desinstalarse desactivando las casillas de verificación correspondientes.

   Nota

   No obstante, debe tener en cuenta que el idioma de producto "inglés" no puede desinstalarse.

8. Haga clic en el botón "Siguiente".
   Se abre el diálogo para seleccionar la configuración de producto.
9. Active la casilla de verificación correspondiente a los componentes que desea instalar. Los componentes ya instalados pueden desinstalarse desactivando las casillas de verificación correspondientes.
10. Haga clic en el botón "Siguiente".

Nota

Observe que no es posible cambiar el directorio de destino porque la instalación existente está siendo modificada.

En caso de que sea necesario modificar opciones de seguridad y permisos para la instalación del TIA Portal, se abrirá el diálogo para la configuración de seguridad.

11. Para continuar la instalación, acepte todos los cambios efectuados en las opciones de seguridad y permisos y haga clic en "Siguiente".
    En el siguiente diálogo aparecen listados los ajustes de instalación.
12. Haga clic en el botón de comando "Modificar".
    Se inicia la instalación de los componentes adicionales.

Nota

Tras la instalación aparece un aviso indicando si la instalación existente se ha modificado correctamente.

13. Es posible que deba reiniciar el equipo. En ese caso, active el botón de opción "Sí, deseo reiniciar mi equipo ahora". A continuación, haga clic en el botón "Reiniciar".
14. Si no es preciso volver a iniciar el equipo, haga clic en el botón "Finalizar".

Resultado

En su equipo se ha modificado la instalación existente.

Consulte también

Indicaciones relativas a los requisitos del sistema (Página 87)

Indicaciones relativas a licencias (Página 86)

Iniciar la instalación (Página 107)

Mostrar el software instalado (Página 111)

Reparación de los productos instalados (Página 114)

Iniciar la desinstalación (Página 115)

Informe de instalación (Página 106)

3.8 Reparación de los productos instalados

En el programa de instalación existe la posibilidad de reparar los productos instalados mediante una reinstalación completa.

Requisitos

- El hardware y software de la programadora o PC cumplen los requisitos del sistema.
- Usted dispone de derechos de administración para su equipo.
- Se han cerrado todas las aplicaciones que se estaban ejecutando.

Procedimiento

Para reparar productos instalados, proceda del siguiente modo:

1. Inserte el medio de instalación en la unidad de disco correspondiente.
   La aplicación Setup arranca automáticamente si el inicio automático no está desactivado en la programadora o PC.
2. Si el programa de instalación no se inicia automáticamente, hágalo manualmente haciendo doble clic en el archivo "Start.exe".
   Se abre el diálogo para seleccionar el idioma de instalación.
3. Seleccione el idioma en el que desea visualizar los diálogos del programa de instalación
4. Para leer las instrucciones del producto y de la instalación, haga clic en el botón "Leer indicaciones" o "Indicaciones de instalación".
   Se abre el archivo de ayuda correspondiente con las indicaciones.
5. Tras haberlas leído, cierre el archivo de ayuda y haga clic en el botón "Siguiente".
   Se abre el diálogo para seleccionar la variante de instalación.
6. Active el botón de opción "Reparar" y haga clic en el botón "Siguiente".
   En el siguiente diálogo aparecen listados los ajustes de instalación.
7. Haga clic en el botón de comando "Reparar".
   Se inicia la reparación de la instalación existente.

   Nota

   Tras la instalación aparece un aviso indicando si los productos instalados se han reparado correctamente.

8. Es posible que deba reiniciar el equipo. En ese caso, active el botón de opción "Sí, deseo reiniciar mi equipo ahora". A continuación, haga clic en el botón "Reiniciar".
9. Si no es preciso volver a iniciar el equipo, haga clic en el botón "Finalizar".

Resultado

Los productos instalados se vuelven a instalar.

Consulte también

Indicaciones relativas a los requisitos del sistema (Página 87)

Indicaciones relativas a licencias (Página 86)

Iniciar la instalación (Página 107)

Mostrar el software instalado (Página 111)

Modificar o actualizar productos instalados (Página 112)

Iniciar la desinstalación (Página 115)

Informe de instalación (Página 106)

3.9 Iniciar la desinstalación

Introducción

Un programa de instalación desinstala automáticamente los paquetes de software. Una vez ejecutado el programa, le guiará paso a paso por todo el proceso de desinstalación.

Existen dos posibilidades para realizar la desinstalación:

- Desinstalación de los componentes seleccionados a través del controlador del sistema
- Desinstalación de un producto con ayuda del medio de instalación

---

Nota

El Automation License Manager no se desinstala automáticamente junto con los paquetes de software, ya que esta aplicación se utiliza para administrar las claves de licencia ("License Keys") de diversos productos de Siemens AG.

Desinstalación de los componentes seleccionados a través del controlador del sistema

Para desinstalar los paquetes de software concretos, proceda del siguiente modo:

1. Abra el Panel de control seleccionando "Inicio > Configuración > Panel de control".
2. Haga doble clic en la entrada "Agregar o quitar programas" del Panel de control.
   Se abre el diálogo "Agregar o quitar programas".
3. Seleccione el paquete de software que desea desinstalar en el diálogo "Agregar o quitar programas" y haga clic en el botón "Quitar".
   Se abre el diálogo para seleccionar el idioma de instalación.
4. Seleccione el idioma en el que desea visualizar los cuadros de diálogo del programa de instalación y haga clic en el botón "Siguiente".
   Se abre el diálogo que permite seleccionar los productos que deben desinstalarse.
5. Active la casilla de verificación para los productos que desea desinstalar y haga clic en el botón "Siguiente".
   En el siguiente diálogo aparecen listados los ajustes de instalación.

6. Verifique la lista de los productos que se desinstalarán. Si desea efectuar cambios, haga clic en el botón de comando "Atrás".
7. Haga clic en el botón de comando "Desinstalar".
   Se inicia la desinstalación.
8. Es posible que deba reiniciar el equipo. En ese caso, active el botón de opción "Sí, deseo reiniciar mi equipo ahora". A continuación, haga clic en el botón "Reiniciar".
9. Si no es preciso volver a iniciar el equipo, haga clic en el botón "Finalizar".

Desinstalación de un producto con ayuda del medio de instalación

Para desinstalar todos los paquetes de software, proceda del siguiente modo:

1. Inserte el medio de instalación en la unidad de disco correspondiente.
   La aplicación Setup arranca automáticamente si el inicio automático no está desactivado en la programadora o PC.
2. Si el programa de instalación no se inicia automáticamente, hágalo manualmente haciendo doble clic en el archivo "Start.exe".
   Se abre el diálogo para seleccionar el idioma de instalación.
3. Seleccione el idioma en el que desea visualizar los diálogos del programa de instalación.
4. Para leer las instrucciones del producto y de la instalación, haga clic en el botón "Leer información de producto" o "Leer indicaciones de instalación".
   Se abre el archivo de ayuda correspondiente con las indicaciones.
5. Tras haberlas leído, cierre el archivo de ayuda y haga clic en el botón "Siguiente".
   Se abre el diálogo para seleccionar la variante de instalación.
6. Active el botón de opción "Desinstalar" y haga clic en el botón "Siguiente".
   En el siguiente diálogo aparecen listados los ajustes de instalación.
7. Haga clic en el botón de comando "Desinstalar".
   Se inicia la desinstalación.
8. Es posible que deba reiniciar el equipo. En ese caso, active el botón de opción "Sí, deseo reiniciar mi equipo ahora". A continuación, haga clic en el botón "Reiniciar".
9. Si no es preciso volver a iniciar el equipo, haga clic en el botón "Finalizar".

Consulte también

Informe de instalación (Página 106)

Indicaciones relativas a los requisitos del sistema (Página 87)

Indicaciones relativas a licencias (Página 86)

Iniciar la instalación (Página 107)

Mostrar el software instalado (Página 111)

Modificar o actualizar productos instalados (Página 112)

Reparación de los productos instalados (Página 114)

3.10 Instalación y desinstalación de la herramienta de migración

3.10.1 Requisitos del sistema

Requisitos del sistema para la herramienta de migración

Para el uso de la herramienta de migración rigen los siguientes requisitos del sistema:

- Deben estar instalados todos los productos con los que se ha creado el proyecto de origen. Se soportan los siguientes productos:
  - WinCC flexible 2008 SP2 y SP3
  - WinCC V7.0 SP3
  - STEP 7 V5.5
  - Proyectos integrados de STEP 7 V5.5 y de los productos WinCC arriba mencionados
  - SINUMERIK
  - STARTER y Startdrives
  - SIMOTION SCOUT V4.4
    Para la migración de proyectos SIMOTION SCOUT V4.4 se necesita SCOUT Migration Tool PlugIn V4.4.
- Deben estar instalados todos los paquetes opcionales requeridos para editar el proyecto STEP 7. Esto incluye por ejemplo todos los HSPs para todos los dispositivos utilizados en el proyecto de origen.

3.10.2 Instalación de la herramienta de migración

Suministro de la herramienta de migración

La herramienta de migración se encuentra en el directorio "Support" del DVD de instalación del TIA Portal. También está disponible en el área del Service & Support del sitio web de Siemens para que pueda descargarla. Algunos productos requieren plug-ins adicionales para la herramienta de migración, p. ej. SIMATIC Failsafe o SIMOTION. Dichos plug-ins también pueden descargarse de la página del Service & Support o instalarse desde el DVD de instalación de los correspondientes productos.

Por lo general, la herramienta de migración se instala sin el TIA Portal. Dado que en el TIA Portal ya viene integrada una función de migración, no es necesario instalar adicionalmente la herramienta de migración.

Procedimiento

Para instalar la herramienta de migración, proceda del siguiente modo:

1. Descargue el archivo de instalación desde el área de Service & Support del sitio web de Siemens o utilice para la instalación el archivo que se encuentra en el directorio "Support" del DVD de instalación del TIA Portal.
2. Inicie el archivo de instalación.
   Se abre el programa de instalación de la herramienta de migración.
3. En el primer paso, seleccione el idioma que desea ver durante la instalación y haga clic en el botón "Siguiente".
   Se abre la página para seleccionar el idioma del software.
4. Puesto que la herramienta de migración únicamente se suministra en inglés, no se puede seleccionar ningún otro idioma para la instalación del software. Haga clic en "Siguiente" para acceder al siguiente paso.
   Se abre la página para seleccionar los componentes que desea instalar.
5. La herramienta de migración consta de un solo componente de software. Por dicho motivo, esta ya aparece marcada.
   Para crear un icono con el que iniciar la herramienta de migración desde el escritorio, active la casilla de verificación "Crear icono en el escritorio". A continuación, haga clic en el botón "Siguiente".
   Se carga la página para aceptar las condiciones de la licencia.
6. Haga clic en uno de los puntos de la lista de términos y condiciones de la licencia para leerlo. Si está de acuerdo con todos los términos y condiciones de la licencia, active la casilla de verificación "Acepto los términos del contrato de licencia". A continuación, haga clic en el botón "Siguiente".
   Se abre un listado de la instalación.
7. Haga clic en el botón "Instalar".
   La instalación se ejecuta con la configuración mostrada.

3.10.3 Desinstalación de la herramienta de migración

La herramienta de migración se puede desinstalar desde el Panel de control.

Procedimiento

Para desinstalar la herramienta de migración, proceda del siguiente modo:

1. Abra el Panel de control.
2. Haga doble clic en la entrada "Agregar o quitar programas" del Panel de control.
   Se abre el diálogo "Agregar o quitar programas".
3. Seleccione la entrada de la herramienta de migración en el cuadro de diálogo "Agregar o quitar programas" y haga clic en el botón "Quitar".
   Aparece una consulta de seguridad.
4. Confirme la consulta de seguridad haciendo clic en el botón de comando "Desinstalar".
   La herramienta de migración se desinstala.

Migrar proyectos y programas 4

4.1 Migrar proyectos en un proyecto del TIA Portal

4.1.1 Migración de proyectos con el TIA Portal

Migración de proyectos existentes

Es posible migrar proyectos de soluciones de automatización anteriores al TIA Portal. En cada migración se crea un proyecto nuevo para los datos migrados, con el que se puede seguir trabajando posteriormente. Primero se cierran aquellos proyectos TIA Portal que puedan estar abiertos.

La migración se muestra posteriormente en la tabla del historial de proyecto. Desde ahí es posible acceder al informe de migración que se crea automáticamente para la migración.

Productos soportados para la migración

En el capítulo "Vista general del sistema STEP 7 y WinCC" obtendrá información sobre qué productos están disponibles en el TIA Portal. Todos los productos allí indicados son básicamente soportados por el TIA Portal durante la migración.

Los requisitos que se deben cumplir dependen de los productos de origen utilizados y de los productos instalados actualmente. Encontrará más información sobre las posibilidades de migración de los productos en Siemens Industry Online Support y en la documentación de los productos de software, p. ej.

Consulte también: Escalado de STEP 7 y WinCC en el TIA Portal (Página 27)

Procedimiento durante la migración

El proceso de migración se compone básicamente de los pasos siguientes:

1. Preparación del proyecto de origen
   Si el software para editar el proyecto de origen no está completamente instalado en la programadora/el PC con el TIA Portal, o si el proyecto de origen es un proyecto integrado, debe convertir en primer lugar el proyecto de origen en un archivo de migración. Para ello instale la herramienta de migración en una PG/un PC donde esté instalado el software necesario para editar el proyecto de origen. Luego convierta el proyecto de origen con la herramienta de migración y copie el archivo en la programadora o en el PC que tiene instalado el TIA Portal. Este paso puede omitirse si el proyecto de origen y el software de origen correspondiente se encuentran en la misma programadora o PC que contiene el TIA Portal y si el proyecto de origen no es un proyecto integrado.
2. Ejecutar migración
   Ejecute la migración en el TIA Portal. Durante la migración, indique como fuente el archivo de migración que ha creado con la herramienta de migración o bien el proyecto de origen, si todo el software necesario está instalado.
3. Revisión del informe de migración
   Para cada migración se crea un informe. En él encontrará indicaciones sobre componentes del proyecto que se han modificado. Puede solicitar el informe en el árbol del proyecto, en "Datos comunes > Informes" o mediante el historial del proyecto. Una vez concluida la migración, el informe se muestra en el TIA Portal. Revise el informe cuando finalice la migración.
   Si la migración no ha concluido con éxito, en el directorio del proyecto se crea en "\Logs" un informe en forma de archivo XML. Dicho archivo se puede abrir con cualquier editor XML y consultar las causas del fallo de la migración.
4. Procesamiento posterior del proyecto migrado
   Debido a que las configuraciones del proyecto de origen no siempre son totalmente compatibles con el TIA Portal, puede ocurrir que las configuraciones que se apliquen en el proyecto migrado no sean idénticas. Por ello, deberá procesar de forma sistemática las indicaciones que contiene el informe de migración. Si no ha incluido la configuración hardware en la migración, debe convertir adicionalmente los dispositivos no especificados en hardware apropiado.

Inclusión de la configuración hardware en la migración

De forma estándar, durante la migración solo se tienen en cuenta los componentes de software del proyecto. Para cada uno de los dispositivos que contiene el proyecto de origen se genera un dispositivo no especificado en el proyecto migrado. La configuración hardware, la configuración de redes y las conexiones no se migran. Por ello, convierta después de la migración los dispositivos no especificados en un dispositivo adecuado y vuelva a establecer manualmente las configuraciones de red y conexiones existentes.

Si está seguro de que el hardware utilizado en el proyecto de origen posee un equivalente adecuado en el TIA Portal, puede incluir la configuración hardware en la migración. En este caso, se migra tanto la configuración hardware como también el software.

Consulte también

Mostrar el informe de la migración (Página 126)

Escalado de STEP 7 y WinCC en el TIA Portal (Página 27)

4.1.2 Preparación de proyectos con la herramienta de migración

4.1.2.1 Migración de proyectos con la herramienta de migración

Pasos previos a la migración

En muchos casos puede ocurrir que el proyecto que se desea migrar no se encuentre en la misma programadora o PC en que está instalada la versión actual del TIA Portal. Por lo tanto, para poder ejecutar la migración primero deberá convertir el proyecto de origen a un formato compatible. Lo mismo es válido para los proyectos integrados.

Después de crear el archivo de migración, cópielo a la programadora o el PC con la instalación actual del TIA Portal. En el TIA Portal, indique el archivo de migración como fuente para la migración y podrá crear un proyecto en el formato de archivo actual del TIA Portal.

Procedimiento al migrar proyectos con la herramienta de migración

Para preparar una migración con la herramienta de migración, son necesarios los siguientes pasos:

1. Instale la herramienta de migración en la programadora o PC con el proyecto de origen. Para ello, descargue del Siemens Industry Online Support el archivo de instalación o instale la herramienta de migración incluida en el DVD de instalación del TIA Portal.
2. Inicie la herramienta de migración y convierta con ella el proyecto de origen al formato de archivo de migración que tiene la extensión ".am13".
   Antes de dar este paso, asegúrese de que todo el software necesario para procesar el proyecto de origen esté instalado en la programadora o el PC. En estos componentes están incluidos todos los Service Packs requeridos, Support Packages del hardware, así como todo el software de ampliación que se requiere para procesar el proyecto de origen. Si algunos productos no están instalados, no se podrá realizar la migración o ésta se realizará de forma incompleta.
3. Copie el archivo de migración en un sistema de destino que disponga de una instalación actualizada del TIA Portal.
   Observe que en el sistema de destino deben estar instalados todos los programas requeridos para configurar todos los dispositivos contenidos en el archivo de migración.
4. Lleve a cabo la migración dentro del TIA Portal e indique como fuente el archivo de migración con la extensión ".am13".
5. Tras la migración, revise el informe de migración, lea sistemáticamente las indicaciones que contiene y retoque lo necesario en el proyecto recién creado. Después de la primera compilación de la configuración, deben tenerse en cuenta especialmente las indicaciones de la ventana de inspección.

Inclusión de la configuración hardware en la migración

De forma estándar, durante la migración solo se tienen en cuenta los componentes de software del proyecto. Para cada uno de los dispositivos que contiene el proyecto de origen se genera un dispositivo no especificado en el proyecto migrado. La configuración hardware, la configuración de redes y las conexiones no se migran. Por ello, convierta después de la migración los dispositivos no especificados en un dispositivo adecuado y vuelva a establecer manualmente las configuraciones de red y conexiones existentes.

Si está seguro de que el hardware utilizado en el proyecto de origen posee un equivalente adecuado en el TIA Portal, puede incluir la configuración hardware en la migración. En este caso, se migra tanto la configuración hardware como también el software.

Consulte también

Migración de proyectos con el TIA Portal (Página 119)

Migración de proyectos (Página 124)

Inicio de la herramienta de migración (Página 122)

Crear el archivo de migración (Página 123)

4.1.2.2 Inicio de la herramienta de migración

Iniciar la herramienta de migración

Durante la instalación, en la entrada "Siemens Automation > Migration Tool" del menú de inicio se crea de manera predeterminada un acceso directo "Migration to TIA Portal V13". Llame este enlace.

También es posible iniciar la herramienta de migración directamente desde el Windows Explorer. De forma estándar, la herramienta de migración se guarda durante la instalación en uno de los siguientes directorios:

- En un sistema operativo de 64 bits:
  C:\Program Files (x86)\Siemens\Automation\MIGTOOL_V13\Bin
- En un sistema operativo de 32 bits:
  C:\Program Files\Siemens\Automation\MIGTOOL_V13\Bin

Para iniciar la herramienta de migración, ejecute el archivo "Siemens.Automation.MigrationApplication.exe" en uno de los directorios.

Consulte también

Crear el archivo de migración (Página 123)

4.1.2.3 Crear el archivo de migración

A continuación se indica cómo convertir el proyecto de origen en un archivo de migración compatible con el TIA Portal con ayuda de la herramienta de migración. Tras la conversión, este archivo se transfiere al sistema de destino y se migra allí.

Se puede determinar si el archivo de migración debe contener todo el proyecto, incluida la configuración hardware completa y el software correspondiente, o si sólo se desea migrar el software.

Requisitos

- Para todas las configuraciones utilizadas en el proyecto de origen es preciso que esté instalado el software original correspondiente con una licencia válida.
- El proyecto de origen no dispone de protección de acceso.
- El proyecto de origen debe encontrarse en un estado coherente, ya que de lo contrario no queda garantizada la migración correcta.

Procedimiento

Para crear el archivo de migración, haga lo siguiente:

1. Elija en el campo "Storage Location (Path)" la ruta del archivo de origen para la migración.
2. Indique qué componentes del proyecto deben migrarse:
   - Active la casilla de verificación "Incluir datos de redes y HW durante la migración" para migrar no sólo el software, sino también los componentes de hardware completos y la configuración de redes del proyecto.
   - Active la casilla de verificación "Copiar datos runtime SCADA" si, además de los datos del sistema de ingeniería, desea migrar los datos runtime, tales como ficheros de avisos, ficheros de variables o ficheros de usuario.
3. Elija en el campo "Intermediate file" la ruta y el nombre del archivo de migración.
4. Haga clic en el botón de comando "Migrate".

Resultado:

Se crea un archivo de migración. Seguidamente copie este archivo en el sistema de destino y migre este archivo en el TIA Portal.

Consulte también

Migración de proyectos (Página 124)

Inicio de la herramienta de migración (Página 122)

Migración de proyectos con la herramienta de migración (Página 121)

4.1.3 Migración de proyectos

Requisitos

- Ya existe un archivo convertido al formato ".am13" o bien está instalado el software original adecuado y con una licencia válida para todas las configuraciones utilizadas en el proyecto de origen.
- El proyecto de origen no dispone de protección de acceso.
- El proyecto de origen debe encontrarse en un estado coherente, ya que de lo contrario no queda garantizada la migración correcta.

Tenga en cuenta las indicaciones adicionales sobre las condiciones básicas que aparecen en la ayuda de los productos instalados.

Nota

Estado de hibernación del sistema durante la migración

El sistema no debe ponerse en modo standby o de hibernación durante la migración. De lo contrario, se cancelará la migración.

Procedimiento

Para migrar un proyecto, proceda del siguiente modo:

1. Elija el comando "Migrar proyecto" del menú "Proyecto".
   Se abre el cuadro de diálogo "Migrar proyecto".
2. En el campo "Ruta de origen" indique la ruta y el nombre de archivo del proyecto que debe migrarse. Elija un proyecto que tenga el formato de migración ".am12" o bien el formato del proyecto de origen.
3. Active la casilla de verificación "Incluir configuración hardware" para incluir la configuración hardware en la migración.
   Si ha seleccionado un archivo de migración creado con la herramienta de migración, la casilla de verificación no será activable. En este caso, ya durante la conversión con la herramienta de migración debe indicar si desea incluir la configuración hardware en la migración.
4. Active la casilla de verificación "Copiar datos de WinCC Runtime Professional" si, además de los datos del sistema de ingeniería, desea migrar los datos runtime, tales como ficheros de avisos, ficheros de variable o ficheros de usuario.
   Si ha seleccionado un archivo de migración creado con la herramienta de migración, la casilla de verificación no será activable. En este caso, ya durante la conversión con la herramienta de migración debe indicar si desea incluir los datos runtime SCADA en la migración.
5. Elija un nombre para el nuevo proyecto en el campo "Nombre del proyecto".
6. En el campo "Ruta de destino" elija una ruta en la cual deba crearse el nuevo proyecto.
7. En el campo "Autor" introduzca su nombre o el nombre de otra persona responsable del proyecto.

8. De ser necesario, introduzca un comentario en el campo "Comentario".
9. Haga clic en "Migrar".

Resultado

El proyecto de origen se convierte y, tras finalizar este proceso, aparece un mensaje. A continuación, el proyecto recién creado se abre en la vista del proyecto y el informe de migración se abre en el TIA Portal.

Si la migración ha fallado, se crea un directorio del proyecto dentro del cual se crea un informe de migración en forma de archivo XML. En el mensaje final después de la migración encontrará un enlace a dicho archivo XML. Haga clic en el enlace para abrir el archivo XML. También encontrará el archivo XML en "\Logs" dentro del directorio del proyecto.

Consulte también

Reajustar proyectos integrados (Página 178)

Mostrar el informe de la migración (Página 126)

Uso de informes (Página 339)

Migración de proyectos con la herramienta de migración (Página 121)

Crear el archivo de migración (Página 123)

4.1.4 Mostrar el desarrollo de la migración

Si un proyecto se ha generado mediante migración, la migración se muestra en la tabla del historial del proyecto. A continuación, en la tabla puede abrirse el informe de migración. Además, se muestra la hora exacta de la migración.

Procedimiento

Para visualizar una tabla sinóptica de la migración, proceda del siguiente modo:

1. Seleccione el proyecto abierto en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual del proyecto.
   Se abre el diálogo con las propiedades del proyecto.
3. Seleccione el grupo "Historial del proyecto" en la navegación local.
   Se abre la tabla sinóptica.

Consulte también

Visualización de las propiedades del proyecto (Página 346)

4.1.5 Mostrar el informe de la migración

Para cada migración correcta se crea un informe. El informe contiene la siguiente información:

- Objetos migrados
- Modificaciones en objetos realizadas durante la migración
- Errores ocurridos durante la migración
- Si es necesario, un enlace a la ayuda para determinados eventos.
  En este caso, obtendrá más ayuda haciendo clic en el interrogante.

Procedimiento

Para visualizar el archivo de registro de la migración, proceda del siguiente modo:

1. Abra la carpeta "Datos comunes > Informes" en el árbol del proyecto.
2. Haga doble clic en el informe deseado de la lista.
   El contenido del informe se muestra en el área de trabajo.

Consulte también

Migración de proyectos con el TIA Portal (Página 119)

Uso de informes (Página 339)

4.1.6 Migrar proyectos de STEP 7

4.1.6.1 Migración de proyectos STEP 7

Introducción

Es posible migrar un proyecto desde SIMATIC STEP 7 V5.4 SP5 o desde SIMATIC STEP 7 V5.5 para utilizarlo en el TIA Portal. Si desea migrar proyectos de versiones STEP 7 antiguas, en primer lugar debe abrirlas, compilarlas y guardarlas en STEP 7 V5.4 SP5 o V5.5. También es posible migrar proyectos integrados con dispositivos de otros productos de software.

Volumen de migración

Por lo general se pueden migrar todas las configuraciones y objetos de SIMATIC STEP 7 V5.4 SP5 o V5.5 que sean también compatibles con la versión instalada del TIA Portal. Esto incluye, por ejemplo, los siguientes dispositivos y configuraciones:

- Dispositivos de las familias S7-300 y S7-400.
- Configuraciones de PROFIBUS con periferia descentralizada conectada. Esto también incluye esclavos basados en GSD, esclavos I e intercambio directo de datos.
- Configuraciones de PROFINET con periferia descentralizada. Esto también incluye dispositivos basados en GSD y dispositivos I.

- Configuraciones de red
- Conexiones
- Bloques creados en los lenguajes KOP, FUP o AWL, S7-SCL y S7-GRAPH.
- Variables PLC
- Tipos de datos definidos por el usuario (UDT)
- Avisos y categorías
- Alarmas
- Atributos definidos por el usuario (UDA)
- Librería de textos de usuario

Inclusión de la configuración hardware en la migración

De forma estándar, durante la migración sólo se tienen en cuenta los componentes de software del proyecto. Para cada uno de los dispositivos que contiene el proyecto de origen se genera un dispositivo no especificado en el proyecto migrado. La configuración hardware, la configuración de redes y las conexiones no se migran. Por ello, convierta después de la migración los dispositivos no especificados en un dispositivo adecuado y vuelva a establecer manualmente las configuraciones de red y conexiones existentes.

Si está seguro de que el hardware utilizado en el proyecto de origen posee un equivalente adecuado en el TIA Portal, puede incluir la configuración hardware en la migración. En este caso, se migra tanto la configuración hardware como también el software.

Consulte también

Notificar errores de sistema (Página 135)

Indicaciones generales sobre la migración de bloques de programa (Página 136)

4.1.6.2 Requisitos para la migración de proyectos STEP 7

Para poder llevar a cabo la migración, es necesario que se cumplan varios requisitos en lo que respecta al software instalado en la programadora o PC original y al proyecto de origen.

Requisitos que debe cumplir la programadora o PC

La programadora o PC debe cumplir los siguientes requisitos:

- Hay instalada una versión con licencia de STEP 7 V5.4 SP5 ó V5.5. En proyectos que solo contienen estaciones PC es suficiente con que esté instalado el software SIMATIC NET PC.
- Para todas las configuraciones utilizadas en el proyecto está instalado el software adicional correspondiente con licencia válida, p. ej., los paquetes opcionales.
- Todos los HSP relativos a módulos que no están incluidos en el catálogo de hardware deben estar instalados.

- Todos los archivos GSD utilizados en el proyecto deben cumplir las especificaciones.
- El usuario tiene que haber iniciado una sesión en el sistema operativo con derechos de administrador.
- Debe estar instalado uno de los siguientes productos de software:
  - TIA Portal con STEP 7 Professional V13 o superior
    Si la migración no se realiza con la última versión del TIA Portal en la programadora o PC original, la funcionalidad del proyecto en las versiones más actuales del TIA Portal estará limitada en un principio. Para poder utilizar toda la funcionalidad de las versiones más nuevas, antes es necesario actualizar el proyecto.
  - Herramienta de migración

Requisitos que debe cumplir el proyecto de origen

- El proyecto de origen no debe tener protección de acceso.
- El hardware y el software deben ser coherentes.
- La asignación de número de aviso está ajustada para toda la CPU.
- No contiene bloques protegidos con conflicto de sello de tiempo.
- Todos los programas y sus fuentes se pueden compilar sin errores.
- Todos los bloques llamados están disponibles en la carpeta de bloques.
- Los bloques no llamados, especialmente los bloques de datos de instancia, no pueden estar en la carpeta de bloques.
- Es aconsejable migrar bloques sin encriptar, puesto que los bloques encriptados con KNOW_HOW_PROTECT (de STEP 7 V5.x) o S7-Block Privacy (de STEP 7 V5.5) no se pueden leer ni modificar en el TIA Portal.

Requisitos que debe cumplir la PG/el PC con STEP 7 Professional

- Debe estar instalado uno de los siguientes productos de software:
  - STEP 7 Professional V13 o superior
  - STEP 7 Professional V13 o superior (sin licencia) si el proyecto migrado solo contiene componentes que se habían configurado con el software de PC SIMATIC NET.
- Para todas las configuraciones utilizadas en el proyecto tiene que estar instalado el software adicional correspondiente con licencia válida, p. ej., los paquetes opcionales.
- Deben estar instalados todos los HSPs relativos a módulos que no están incluidos en el catálogo de hardware y todos los archivos GSD utilizados en el proyecto.

4.1.6.3 Comprobar la capacidad de migración de los proyectos STEP 7

Antes de iniciar la migración, puede comprobar si se cumplen todos los requisitos necesarios para la migración.

Nota

Tenga en cuenta que, al efectuar los pasos siguientes, se restablecen los valores actuales del proyecto original. Observe también que al ejecutar la comprobación de coherencia de bloques en STEP7 V5.x se restablecen los valores predeterminados para los valores de arranque en los DB de instancia. Si utiliza funciones tecnológicas, este comportamiento no es deseado.

Procedimiento

Para verificar la capacidad de migración de un proyecto, proceda del siguiente modo:

1. Abra el proyecto original en SIMATIC STEP 7 V5.4 SP5 o V5.5. Para que la calibración sea correcta, solo deben estar instalados los paquetes opcionales y módulos disponibles en el TIA Portal, incluidos los HSP y los archivos GSD.
2. Abra las diferentes estaciones. Todos los componentes necesarios para la migración están disponibles si, al abrir las estaciones, no aparece ningún mensaje que indique que faltan componentes.
3. Lleve a cabo una comprobación de coherencia de bloque para cada contenedor de bloques existente en el proyecto.
   Encontrará más información sobre la comprobación de coherencia en el capítulo "Comprobación de coherencia en el proyecto de origen antes de la migración (Página 130)".
4. Compile todo el proyecto. Si no se muestra ningún error durante la compilación, significa que los programas se pueden migrar.
5. Compruebe si todos los módulos están incluidos en el catálogo de hardware del TIA Portal con idéntica referencia (MLFB) y versión de firmware.
6. Compruebe que todas las CPUs tengan activada la asignación de número de aviso para toda la CPU.
7. Compile el proyecto en NetPro. Aquí tampoco debe aparecer ningún error.

Consulte también

Comprobación de coherencia en el proyecto de origen antes de la migración (Página 130)

Entrada de FAQ sobre la comprobación de coherencia (http://support.automation.siemens.com/WW/view/es/5416540)

4.1.6.4 Comprobación de coherencia en el proyecto de origen antes de la migración

Proyecto de origen coherente

El requisito para que una migración de un proyecto SIMATIC STEP 7 V5.x al TIA Portal sea correcta es que el proyecto de origen sea coherente. Por este motivo, antes de la migración debe efectuar una comprobación de coherencia en el proyecto de origen. Encontrará las instrucciones para la comprobación de coherencia en SIMATIC STEP 7 V5.x en una entrada FAQ (http://support.automation.siemens.com/WW/view/es/5416540) en el área Service & Support o en la ayuda sobre SIMATIC STEP 7 V5.x.

Migración de valores de arranque a los DB de instancia para bloques tecnológicos

La compilación en la comprobación de coherencia de bloques da lugar a que las variables en los DB de instancia se restablezcan a los valores predeterminados. Los valores de arranque ajustados se pierden durante este proceso.

Si a pesar de todo se desea migrar los valores de arranque, es recomendable uno de los siguientes procedimientos:

- Para proyectos pequeños: Copia de seguridad de los DB de instancia fuera de la carpeta de bloques
  Haga una copia de seguridad de los DB de instancia antes de iniciar la comprobación de coherencia. Después de la comprobación de coherencia, vuelva a copiar los DB de instancia en la carpeta de bloques.
- Para proyectos grandes: Creación de una copia del proyecto
  Cree primero una copia del proyecto y lleve a cabo la verificación con la copia del proyecto. Los errores que pueda encontrar durante la comprobación de coherencia debe subsanarlos en el proyecto de origen.

A continuación, puede migrar el proyecto con los valores que haya configurado. Con ello se conserva también la configuración de funciones tecnológicas.

Consulte también

Principios básicos de los valores de arranque (Página 2948)

4.1.6.5 Eliminación de componentes hardware no soportados

Aunque en el proyecto de origen se hayan creado configuraciones para las que no se ha instalado todo el software apropiado, dicho proyecto se podrá migrar eliminando manualmente del mismo las configuraciones no soportadas. De manera alternativa se puede usar la función "Reorganización".

Procedimiento

Para depurar el proyecto de origen, proceda del siguiente modo:

1. Utilice una instalación de SIMATIC STEP 7 V5.4 o V5.5 que contenga solo los paquetes opcionales existentes en el TIA Portal y todos los módulos necesarios.
2. Guarde de nuevo el proyecto seleccionando la opción "Con reorganización". De este modo se eliminan del proyecto todas las configuraciones no soportadas.

Para los módulos que en el TIA Portal solo están disponibles en una versión más reciente o en una versión de firmware más nueva, utilice en la estación la función "Sustituir objeto" para sustituir un módulo antiguo por otro que se pueda migrar.

4.1.6.6 Particularidades de la migración

Nombres de dispositivo

Nombres de los dispositivos PROFINET IO

En STEP 7 V5.5 se utiliza el nombre de un dispositivo PROFINET IO como nombre del dispositivo IO. Después de la migración a un proyecto del TIA Portal, el nombre del dispositivo PROFINET IO del proyecto de origen se utiliza:

- Como nombre de dispositivo PROFINET
- Como nombre para el módulo de interfaz
- Como nombre de la estación
  El TIA Portal solo admite nombres de estación con un máximo de 24 caracteres. Por ello, si es necesario se acorta el nombre de la estación a un máximo de 24 caracteres. Si al acortar el nombre de la estación surgieran varias estaciones con el mismo nombre, los últimos caracteres del nombre de la estación se utilizarán para numerar las estaciones.

Nombres de módulos y racks

En SIMATIC STEP 7 V5.5, los nombres de módulos y racks (bastidores) no tienen que ser unívocos. En el TIA Portal, por el contrario, es imprescindible que éstos tengan una denominación unívoca. Por eso, en caso necesario se realiza un contaje incremental de los módulos y racks durante la migración, a fin de garantizar denominaciones unívocas. Una excepción son los dispositivos PROFINET IO y los CPs PROFINET En estos casos hay que prestar atención a no dar nombres repetidos.

Configuraciones de PROFINET IO

Necesidad de asignar nombres unívocos

Los nombres de los dispositivos PROFINET IO no pueden repetirse en una misma subred. Esto también vale para los nombres de CPs PROFINET. Por ello, antes de proceder a la migración asegúrese de que los nombres no aparecen repetidos en el proyecto de origen y

después de la migración compruebe que sean unívocos antes de proceder a compilar por primera vez.

Redundancia de medios en dominios Sync

Si existe un dominio Sync con una redundancia de medios activada en el proyecto de origen, se adoptará la configuración del dominio Sync en la migración. Si, por el contrario, existen varios dominios Sync con una redundancia de medios activada, éstos se eliminarán y el proyecto se migrará sin dominios Sync. En todo caso recibirá un mensaje sobre los dominios Sync que se han adoptado.

Para configurar la redundancia de medios en el proyecto de origen, utilice en cualquier caso SIMATIC STEP 7 V5.5.

Modo isócrono con PROFINET IO

En las configuraciones de PROFINET IO con modo isócrono hay que tener en cuenta lo siguiente:

- IRT de alto rendimiento
  Si se ha configurado IRT de alto rendimiento al planificar la topología del proyecto de origen, se modificará la opción a "RT" en la migración. Recibirá un mensaje que le informará sobre la modificación.
- Utilización del OB 61
  Cuando en la configuración original del proyecto STEP 7 está activado el modo isócrono para PROFINET IO con OB 61, la migración lo desactiva. No obstante, los dispositivos sí se transfieren. Las configuraciones de PROFINET IRT con la opción "Alta flexibilidad" que no utilizan el OB 61 se pueden migrar.

CBA con controladores IO

Si en el proyecto de origen se ha utilizado el servicio cíclico CBA (Component based Automation), éste será desactivado por la migración. Recibirá un mensaje que le informará sobre la modificación.

Utilización de dispositivos compartidos

Si en el proyecto de origen se han configurado Shared Devices, éstos se eliminarán en la migración. Recibirá un mensaje que le informará sobre la eliminación.

Áreas de transferencia

Si en el proyecto de origen se han configurado áreas de transferencia para el intercambio de datos entre el controlador IO y el I-Device, éstas se eliminarán en la migración. Se le informará mediante un mensaje cuando un área de transferencia sea eliminada por la migración.

Grupos de actualización para PROFINET IO

En caso de haberlos, elimine los grupos de actualización del proyecto STEP 7 antes de la migración. Si todavía hay grupos de actualización configurados, no se podrá realizar la migración.

I-Slaves e I-Devices

Sólo se pueden migrar configuraciones de esclavos I y dispositivos I conectadas y consistentes. Las configuraciones que no sean consistentes quedarán excluidas de la migración.

Sistemas PROFINET IO no conectados

Las estaciones con sistemas PROFINET IO no conectados se pueden migrar. Tras la migración, los dispositivos IO y los sistemas PROFINET IO aparecen desconectados en la vista de redes. Conecte estos dispositivos IO tras la migración con un controlador IO, o borre los dispositivos IO no necesarios.

Configuraciones de PROFIBUS

PROFIBUS en modo isócrono

Es posible migrar configuraciones de PROFIBUS que funcionen en modo isócrono. Ahora bien, después de la migración se aconseja recompilarlas para descartar posibles incoherencias. Observe también si hay ligeros cambios en los bloques SDB.

Sistemas maestro huérfanos

Las estaciones con sistemas maestro DP huérfanos pueden migrarse. Tras la migración, los esclavos DP del sistema maestro DP huérfano aparecen desconectados en la vista de redes. Conecte estos esclavos DP tras la migración con un maestro DP, o borre los esclavos DP no necesarios.

Dispositivos GSD y GSDML

Para los dispositivos configurados con archivos GSD PROFIBUS o GSD PROFINET, los archivos GSD correspondientes también deben estar instalados en el TIA Portal. Si los archivos GSD todavía no están instalados en el TIA Portal, se instalarán automáticamente durante la migración. Los archivos GSD deben cumplir las especificaciones (p. ej., GSDML-Specification for PROFINET IO, ver www.profinet.com (http://www.profinet.com)) para poder migrar correctamente el proyecto. Si los archivos GSD de un fabricante de dispositivos no cumplen la especificación, el proyecto no puede migrarse. El informe de migración ofrece avisos que indican qué archivos GSD no están normalizados.

Si los archivos GSD no cumplen las especificaciones, compruebe si el fabricante ha proporcionado una versión más reciente del archivo GSD. Instale la versión más reciente. Si no existe ninguna versión del archivo GSD que cumpla las especificaciones, póngase en contacto con el fabricante del dispositivo.

Conexiones

Las conexiones de los proyectos de SIMATIC STEP 7 normalmente se pueden migrar, siempre y cuando también sean compatibles con la versión instalada del TIA Portal.

Tipos de conexiones que se pueden migrar

Los tipos de conexiones que se pueden migrar son los siguientes:

- Conexión S7 unilateral
- Conexión S7 bilateral
- Conexión S7 de seguridad
- Conexión TCP
- Conexión ISO on TCP
- Conexión ISO
- Conexión UDP
- Conexión de correo electrónico
- Conexión FDL
- Conexión punto a punto

La migración no acepta las conexiones FMS con PROFIBUS. Adapte las conexiones o cambie el programa de usuario.

Multiproyectos

Multiproyectos en SIMATIC STEP 7

En SIMATIC STEP 7 existe la posibilidad de dividir los proyectos en lo que se denomina un multiproyecto; por ejemplo, cuando un proyecto es muy grande o cuando hay varias personas trabajando en él. Los subproyectos pueden contener referencias para todos los proyectos, p. ej. conexiones. Todos los subproyectos de un multiproyecto y sus respectivas librerías se encuentran en el mismo directorio.

Migración de partes de un proyecto

Es posible migrar los subproyectos de un multiproyecto. Para ello, seleccione uno de los subproyectos al hacer la migración. Esta toma todos los dispositivos y configuraciones incluidos en él.

Para migrar subproyectos, tenga en cuenta lo siguiente:

- Conexiones interproyecto
  Las conexiones entre subproyectos de multiproyectos se crean como conexión no especificada.
- Interconexión interproyecto
  Con la migración se pierden los datos sobre la interconexión de dispositivos más allá de los límites del proyecto, pero sí se conservan las conexiones entre dispositivos incluidos en el mismo subproyecto.

Agrupación de un multiproyecto para la migración

La migración de un multiproyecto entero con todos los subproyectos correspondientes sólo es posible si éstos se vuelven a juntar manualmente antes de la migración. Esto se hace, por ejemplo, copiando los distintos dispositivos y agregándolos a un único proyecto.

Avisos

Los avisos de PLC se migran con todos los parámetros e incluidas todas las categorías. También se migran los tipos de aviso e instancias de aviso y se almacenan bajo los avisos de PLC en el TIA Portal.

Requisitos

El requisito es que en el proyecto de origen esté seleccionada la asignación de los números de aviso para toda la CPU. Por este motivo, en caso necesario, antes de la migración active en el proyecto de origen la opción "Asignar siempre números de aviso unívocos para toda la CPU".

Restricciones

- Los avisos de símbolos no se pueden migrar.
- Se incorporan las palabras clave "$$AKZ$$", "$$BlockComment$$" y "$$Area$$", pero no se realiza ninguna sustitución en tiempo de ejecución.

Notificar errores de sistema

En principio, los avisos de error de sistema configurados con SIMATIC STEP 7 se pueden migrar.

Limitaciones

Al migrar la aplicación "Notificar errores de sistema", sólo se migran los ajustes realizados por el usuario en el proyecto SIMATIC STEP 7 (por ejemplo, configuraciones básicas, números de bloque, nombres de bloque, etc.).

Los siguientes elementos no se incluyen en la migración:

- Bloques SFM
- Los avisos SFM y los ajustes realizados para configurar los textos de aviso
- Librerías de textos del sistema

Los ajustes se pueden volver adaptar manualmente una vez realizada la migración. Los elementos que falten se generarán de nuevo al compilar por primera vez la configuración hardware.

Migración de bloques de programa

Indicaciones generales sobre la migración de bloques de programa

Por lo general se pueden migrar todos los bloques creados en KOP, FUP, SCL, AWL y GRAPH. No obstante, tenga en cuenta las indicaciones siguientes:

Programas con direccionamiento absoluto

En el TIA Portal se declaran automáticamente operandos simbólicos para todas las direcciones absolutas. Si en el programa STEP 7 que se va a migrar se ha estado trabajando con direccionamiento absoluto, al realizar la migración también se declararán operandos simbólicos para las direcciones absolutas. Esta declaración automática puede provocar conflictos en los tipos de datos, especialmente cuando está activada la verificación IEC en el TIA Portal. Así, por ejemplo, podría surgir un conflicto de tipo de datos cuando un valor de 32 bits es declarado como variable de tipo DWORD, pero luego en el programa se espera una variable con datos de tipo REAL.

En este caso no es suficiente desactivar la verificación IEC. En lugar de ello se debe corregir la declaración en la tabla de variables PLC.

Migración de bloques de comunicación

Durante la migración de instrucciones de comunicación de STEP 7 V5.x se sustituyen punteros del tipo de datos ANY por variables PLC dentro de la propia estación S7.

- Tras la migración, se muestra una advertencia en los parámetros afectados.
- Sustituya las variables PLC en los parámetros afectados por un puntero.
  En el direccionamiento absoluto de un bloque de datos con tipo de datos ANY, el bloque de datos ha debido crearse con acceso estándar, puesto que no es posible el direccionamiento simbólico con el tipo de datos ANY.
  Ejemplo: Utilice la instrucción P#M10.0 BYTE 4 en lugar de %MD10 para la entrada de valores en los parámetros correspondientes.

Nombres simbólicos para bloques y tipos de datos definidos por el usuario (UDT)

En el TIA Portal todo bloque o UDT tiene un número y un nombre. Por eso ya no es necesario declarar nombres simbólicos. Si el programa migrado contiene declaraciones de símbolos para bloques o UDT, después de la migración se utilizarán como nombres.

Bloques de librerías STEP 7

El TIA Portal no ofrece funciones y bloques de función estándar alojados en librerías, sino que se hallan disponibles a modo de instrucciones en la Task Card "Instrucciones". Las instrucciones están ordenadas por funciones y se pueden encontrar bajo su nombre simbólico.

Si el programa que se va a migrar contiene llamadas de funciones o bloques de función estándar, durante la migración se reemplazarán por las instrucciones equivalentes a la función o bloque de función estándar original.

En caso de que el bloque de la librería del TIA Portal ya no esté soportado, existen las siguientes alternativas:

- El bloque de la librería se sustituye por una instrucción compatible. En este caso, aparece un aviso indicando que debe compilar el programa después de la migración. Tenga en cuenta que el bloque invocante no puede llevar ninguna protección de know-how, ya que la llamada de bloque debe adaptarse durante la compilación.
- Si no existe ninguna instrucción compatible, se migra el bloque como bloque de usuario. Este aparece entonces como bloque con protección de know-how en la carpeta "Bloques de programa" y tiene el complemento del nombre "_LF" (legacy function).

Librerías de bloques personalizadas

Las librerías de bloques personalizadas ya no están disponibles en el TIA Portal en su forma conocida. No obstante, se pueden migrar librerías personalizadas insertándolas en un proyecto antes de su migración y, a continuación, migrando éste. Seguidamente, en el TIA Portal se pueden copiar desde el proyecto y utilizarse.

Instrucciones con otra representación

La representación de algunos comandos de KOP, FUP, AWL, SCL y GRAPH en el TIA Portal es distinta de la de antiguas versiones de STEP 7. Así, por ejemplo, las funciones matemáticas y los comparadores ya no son específicos de los datos. En su lugar hay una instrucción central aplicable a todos los tipos de datos. El comando "ADD_I", por ejemplo, ya no se admite y es sustituido por la instrucción "ADD".

También hay otros comandos que tienen una nueva representación en el TIA Portal; por ejemplo, comandos de flancos, operaciones lógicas con palabras, conversiones, temporizadores CEI, contadores CEI, etc.

Los comandos adoptan la nueva representación durante la migración.

Verificación IEC más estricta

En el TIA Portal se aplican reglas más estrictas para verificar la compatibilidad de los tipos de datos. Además, al verificar la sintaxis, ya se detectan aquellos errores que pueden provocar errores en tiempo de ejecución. Así, por ejemplo, en las funciones (FC) ya no se admite la

escritura en parámetros de entrada propios ni la lectura de parámetros de salida propios. Para evitar errores de sintaxis, tenga en cuenta las siguientes reglas:

- Utilice únicamente variables con tipos de datos compatibles en la transferencia de parámetros.
- Utilice únicamente variables con tipos de datos compatibles en las instrucciones de comparación o en instrucciones aritméticas.
- No se puede escribir en parámetros de entrada ni leer parámetros de salida.

Encontrará más información sobre las interfaces de bloques y la conversión de los tipos de datos con verificación CEI en "Consulte también".

Identificador de acceso a periferia ":P"

En el TIA Portal se utiliza el identificador de acceso ":P" para el direccionamiento directo de la periferia. No se admite, por ejemplo, la siguiente notación:

```
%PEW3
```

En su lugar se utiliza la siguiente:

```
%EW3:P //representación absoluta
MyTag:P //representación simbólica
```

Los accesos adoptan la nueva notación durante la migración. Ahora bien, no es posible los nombres simbólicos declarados para variables de periferia en el programa original. Por el contrario, se crean nuevas declaraciones. Encontrará más información sobre el acceso a la periferia en "Consulte también".

En las etiquetas de salto no se distingue entre mayúsculas y minúsculas.

En el TIA Portal no se distingue entre mayúsculas y minúsculas en las etiquetas de salto. Si el programa original contiene marcas de salto que sólo se distinguen por las mayúsculas y minúsculas, éstas se transforman en marcas de salto unívocas. El archivo de protocolo proporciona información sobre las marcas de salto modificadas.

El carácter especial alemán "ß" no se diferencia de "ss"

En el TIA Portal, el carácter especial alemán "ß" no se diferencia de "ss" en nombres de variables. Si el programa original contiene, p. ej., nombres de variables que solo se distinguen por estos dos caracteres, estos se transforman en nombres de variables unívocos.

Migración de bloques de datos globales

La representación y manejo de los valores en los bloques de datos varía en el TIA Portal y en SIMATIC STEP 7. En los bloques de datos globales que no han sido derivados de un objeto de orden superior, p. ej. de un UDT, las variables obtienen siempre el valor predeterminado del tipo de datos, p. ej. FALSE para el tipo de datos BOOL. Este valor predeterminado no se puede editar. En caso de que en el proyecto de origen se hayan asignado valores iniciales offline en la vista de declaración, éstos no se adoptarán en la migración. Si en el programa son necesarios valores predeterminados del usuario, utilice un bloque de datos global basado en un tipo de datos PLC. En un tipo de datos PLC se pueden asignar valores predeterminados del usuario para los valores de variables.

En "Consulte también" encontrará más información sobre los valores de datos en los bloques de datos.

Consulte también

Programar bloques de datos (Página 2937)

Direccionar la periferia (Página 2294)

Reglas generales de la declaración de la interfaz del bloque (Página 2645)

Tipos de datos (Página 2321)

Migración de programas KOP/FUP/AWL

Los bloques KOP/FUP y AWL pueden ejecutarse después de la migración sin necesidad de compilarlos o cargarlos de nuevo.

Protección de know-how

Los bloques que tienen protección de know-how antes de la migración siguen teniéndola después de migrarlos. Puesto que las fuentes no se migran, no se puede eliminar la protección de know-how. Por eso, los bloques no se pueden abrir ni editar. No obstante, existe la posibilidad de cargar los bloques en la CPU para ejecutarlos.

Para suprimir la protección de know-how de un bloque existen dos posibilidades:

- Antes de la migración, elimine de la fuente el atributo KNOW_HOW_PROTECT y seguidamente genere un bloque sin protección de know-how. A continuación migre este bloque.
- Importe la fuente al TIA Portal y, a partir de ésta, genere un bloque sin protección de know-how. Proteja seguidamente el bloque con una contraseña a través del comando de menú "Editar > Protección de know-how".

Conversión de KOP/FUP a AWL

El sistema exige que aquellos programas que fueron creados en KOP o en FUP y que contienen operaciones de salto a redes subsiguientes se representen en AWL después de la migración. Sin embargo, en las propiedades del bloque sigue apareciendo el lenguaje KOP o FUP.

Cambie el programa a AWL en las propiedades del bloque. Después, vuelva a cambiarlo a KOP o FUP.

Migración de programas GRAPH

Requisitos

El requisito para migrar bloques GRAPH es que el dispositivo original tenga instalado el paquete opcional "S7-GRAPH, versión 5.3 SP6" o superior.

Configuración de bloque específica de GRAPH

La configuración de bloque específica de GRAPH se ha reducido mucho en el TIA Portal. En la siguiente tabla se muestra cómo se transfieren las configuraciones de bloque originales al TIA Portal.

Configuración en el proyecto de origen		Configuración en el TIA Portal
Parámetros FB	Mínimo	Se migran los parámetros existentes
	Estándar	
	Máximo	
	Definible	
Descripción de la interfaz	Minimizado	Generar DB minimizado
	Campos de estructura	Siempre se crean estructuras individuales
	Estructuras individuales	
Capacidad de ejecución	FC70/71	FC 73 si está activada la opción "Generar DB minimizado"; de lo contrario, FC 72
	FC72	
	FC73	
Análisis de criterios en el DB		Configuración en el TIA Portal irrelevante
Saltar etapas		Saltar etapas
Acuse obligatorio en caso de error		Acuse obligatorio en los errores de supervisión
Sincronización		En el TIA Portal siempre activado
Procesamiento permanente de todos los interlocks en modo manual		Procesamiento permanente de todos los interlocks en modo manual
Bloqueo de la selección del modo de operación		Bloqueo de la selección del modo de operación
Comportamiento de conmutación seguro		En el TIA Portal siempre activado
Advertencias	Ninguna	En el TIA Portal siempre activado
	Todos	
Activar avisos		Activar avisos
Avisos con WR_USMSG		En el TIA Portal avisos siempre con ALARM_SQ/ALARM_S

Extensión del nombre de las etapas

Las extensiones del nombre de las etapas no se soportan en el TIA Portal. Durante la migración se convierten en texto específico de la etapa.

Compilación del bloque migrado

Después de la migración hay que compilar y cargar de nuevo los bloques GRAPH.

Durante la compilación pueden llegar a producirse los siguientes errores:

- Debido a la verificación IEC más estricta pueden surgir conflictos en los tipos de datos. En dicho caso, modifique la declaración.
- Puesto que en el TIA Portal ya no está disponible la opción "Campos de estructura" para la interfaz, sus elementos ya no se pueden direccionar, por ejemplo, con G7S[1].X. Por eso es necesario corregir el acceso al elemento, cambiándolo, por ejemplo, a STEP1.X.

Modificación de la interfaz del bloque GRAPH

Los cambios de la configuración pueden provocar cambios en la interfaz del bloque. Por eso es muy posible que haya que generar de nuevo el DB de instancia o actualizar las llamadas de bloque.

Migración de programas SCL

El requisito para migrar bloques SCL es que en el dispositivo original esté instalado el paquete opcional "S7-SCL, versión 5.3 SP5".

Después de la migración hay que compilar y cargar de nuevo los bloques SCL.

Procedimiento básico en la migración

La migración completa de bloques SCL sólo se lleva a cabo si se hallan presentes las fuentes correspondientes en el proyecto de origen.

La tabla siguiente muestra el procedimiento básico a seguir:

Presente en el proyecto de origen	Presente después de la migración
Bloques SCL con fuente	Bloques SCL editables
Bloques SCL con protección de know-how con fuente	Bloques SCL editables. Después de la migración, los bloques ya no tienen protección de know-how, pero en caso de necesidad se les puede volver a asignar dicha protección.
Bloques SCL sin fuente	Bloques SCL con protección de know-how
Bloques SCL con protección de know-how sin fuente	Bloques SCL con protección de know-how

Opciones específicas del compilador SCL

Solo se aplican como propiedad del bloque durante la migración al TIA Portal las opciones del compilador que están definidas directamente en una fuente SCL. En caso de que no esté definida ninguna opción del compilador en la fuente SCL de origen, se desactivarán las opciones en las propiedades del bloque migrado.

Las opciones del compilador que estén definidas como ajuste en el editor SCL o en un archivo de control de compilación SCL no se migrarán.

En la siguiente tabla se muestra una vista general de las opciones del compilador que se transfieren al TIA Portal como propiedad del bloque.

Opción del compilador en la fuente SCL	Propiedad del bloque en el TIA Portal
scl_monitorarraylimits	Comprobar límites de ARRAY
scl_createdebuginfo	Crear información avanzada de estado
scl_setokflag	Asignar ENO automáticamente

Nuevo mecanismo EN/ENO

SCL utiliza el mecanismo EN/ENO del TIA Portal. Por ello, durante la migración se sustituyen todos los usos del flag OK por ENO. Los puntos en los que se ha utilizado ENO en el programa original quedan marcados. Estos puntos se deben verificar tras la migración para adaptarlos al nuevo mecanismo.

Encontrará más información sobre el mecanismo EN/ENO del TIA Portal en "Consulte también".

Operador "DIV"

El operador "DIV" ya no está disponible en el TIA Portal. Durante la migración se convierten todos los usos de "DIV" en la notación "/" conforme con las normas.

ARRAYs anidados

En el TIA Portal no es posible anidar ARRAYs. No se admitiría, por ejemplo, la siguiente declaración:

```
ARRAY[1..5] OF ARRAY[0..3] OF INT
```

Durante la migración los ARRAYs anidados se convierten en ARRAYs multidimensionales. Después de la migración, el ejemplo anterior quedaría así:

```
ARRAY[1..5, 0..3] OF INT
```

Declaración de etiquetas (LABEL)

La declaración de etiquetas de salo no es posible en el TIA Portal. No se aceptaría, por ejemplo, la siguiente declaración del proyecto de origen:

```
LABEL
MARKE1, MARKE2, MARKE3 ;
END_LABEL
```

Pero las marcas de salto programadas se conservan en el código del programa y se pueden utilizar para instrucciones GOTO.

Declaración de constantes simbólicas en la interfaz del bloque

En el TIA Portal, las constantes simbólicas siempre tienen un tipo de datos. Si en el programa original hay constantes simbólicas declaradas sin tipo de datos, durante la migración se les asigna un tipo de datos apropiado.

Accesos indexados a la periferia

En el TIA Portal se utiliza la sintaxis con paréntesis para accesos indexados a la periferia. No se admite, por ejemplo, la siguiente notación:
`PEB[1]`

En su lugar se utiliza la siguiente:
`EB(1):P`

Los accesos adoptan la nueva notación durante la migración. Encontrará más información sobre el acceso a la periferia en "Consulte también".

Accesos indexados a la memoria

En el TIA Portal se utiliza la sintaxis con paréntesis para accesos indexados a la memoria. No se admite, por ejemplo, la siguiente notación:
`EB[2]`

En su lugar se utiliza la siguiente:
`EB(2)`

Los accesos adoptan la nueva notación durante la migración. Encontrará más información sobre el direccionamiento en "Consulte también".

Acceso indexado a bloques de datos

En el TIA Portal se utiliza la sintaxis con paréntesis para accesos indexados a los bloques de datos. No se admite, por ejemplo, la siguiente notación:
`%DB100.DW[5]`

En su lugar se utiliza la siguiente:
`%DB100.DW(5)`

Los accesos adoptan la nueva notación durante la migración. Encontrará más información sobre el direccionamiento en "Consulte también".

Acceso absoluto a bloques de datos

En un acceso absoluto se tiene que utilizar el identificador absoluto del bloque de datos; el acceso con el identificador simbólico no se admite en el TIA Portal.

No se admite, por ejemplo, la siguiente notación:
`DB100.DW3`

En su lugar se utiliza la siguiente:
`%DB100.DW3`

Durante la migración se añade el signo "%" cuando se detecta un acceso absoluto a bloques de datos. Encontrará más información sobre el direccionamiento en "Consulte también".

Funciones logarítmicas "EXPD" y "LOG"

Las funciones logarítmicas "EXPD" y "LOG" ya no están disponibles en el TIA Portal. Durante la migración, todos los usos de "EXPD" se convierten en la notación normalizada

"10**(<Expresión>)". Durante la migración, todos los usos de "LOG" se convierten en la notación normalizada "LN(<Expresión>)".

Instrucción matemática "MAX"

En el TIA Portal la instrucción "MAX" no puede procesar ningún tipo de datos temporales. Sin embargo, en lugar de eso puede utilizar comparadores.

El siguiente ejemplo muestra cómo puede determinar el "mayor" de dos tipos de datos temporales con la ayuda de comparadores:

SCL

```
#xs_tod := #xi_tod + #xi_time;
   // statt #xstod1 := MAX(IN1:=#xtod1, IN2:=#xstod1);
   IF #xs_tod < #xi_tod THEN
         #xs_tod := #xi_tod;
   END_IF;

   // statt #xsd1 := MAX(IN1:=#xd1, IN2:=#xsd1);
   IF #xi_date <#xs_date THEN
         #xs_date := #xi_date;
   END_IF;
#xs_date_time := #xi_date_time + #xi_time;
   // statt #xs_date_time := MAX(IN1:=#xi_date_time, IN2:=#xs_date_time);
   IF #xi_date_time <#xs_date_time THEN
          #xs_date_time := #xi_date_time;
   END_IF;
```

Números en coma flotante - Representación exponencial

La representación exponencial de los números en coma flotante sin punto decimal ya se no admite en el TIA Portal. No se admite, por ejemplo, la siguiente notación:
`3E10`

En su lugar se utiliza la siguiente:
`3.0E10`

Durante la migración se convierten todos los usos de "3E10" en la notación "3.0E10" conforme con las normas. Encontrará más información sobre números en coma flotante en "Consulte también".

Bloques de librerías STEP 7

Las funciones FC1 - FC40 de la librería de STEP 7 Classic "Funciones IEC estándar" ya no están disponibles en el TIA Portal. Durante la migración, las llamadas de las funciones estándar se convierten en llamadas de las instrucciones avanzadas correspondientes. Si la conversión no es posible, se emite un error de sintaxis.

Modifique en tal caso el bloque SCL después de la migración y utilice las instrucciones propias de SCL u operadores en lugar de funciones IEC estándar.

Ejemplos:

Utilice S5TIME_TO_TIME en lugar de S5TI_TIM (FC33)
```
"TAG_TIME" := S5TIME_TO_TIME ("TAG_S5TIME");
```

Utilice el operador de comparación <> en lugar de NE_DT (FC 9):
```
IF #D1 <> #D2 THEN "MyOutput" : = 10;
END_IF;
```

Utilice el operador de comparación = en lugar de EQ_STRNG (FC 10):
```
IF #My_String1 = #MyString2 THEN "MyOutput" : = 10;
END_IF;
```

Interrupción de strings con "$"

En STEP 7 Classic, en SCL los strings podían interrumpirse mediante caracteres especiales, p. ej. caracteres de control o caracteres no imprimibles. Esos caracteres se identificaban con las secuencias de interrupción "$>" y "$<".

Ejemplo:
```
#myString := 'Test1$>     $<Test1';
```

En el TIA Portal ya no existe esa posibilidad. Las interrupciones de strings se borran en la migración.

Puntero NULL

En el TIA Portal, para los punteros NULL ya no se utiliza la notación "NIL", sino "NULL". La notación se sustituye automáticamente en la migración.

Comentarios

Si es posible se incorporan los comentarios de la fuente migrada. Rigen las reglas siguientes:

- Los comentarios que en el bloque de origen están inmediatamente detrás de la definición del bloque se convierten en comentarios del bloque.
- Los comentarios que son líneas de comentario situadas tras una declaración de variable se incorporan en la columna "Comentario" de la interfaz del bloque.
- Los comentarios restantes no se incorporan.

Ejemplo:
```
// Comentario que no se incorpora.
FUNCTION_BLOCK FB57
// Comentario que se incorpora en las propiedades del bloque.
(*Comentario que se incorpora en las propiedades del bloque. *)
VAR_TEMP
    myVar1 : INT;// Comentario que se incorpora en la interfaz.
    myVar2 : INT;(* Comentario que no se incorpora porque es una
sección de comentario *)
END_VAR
// Comentario que no se incorpora.
```

```
BEGIN
    ;
END_FUNCTION_BLOCK
```

Consulte también

Declarar constantes globales (Página 2583)

Principios básicos de las constantes (Página 2288)

Mecanismo EN/ENO (Página 2539)

Direccionar la periferia (Página 2294)

Números en coma flotante (Página 2338)

Direccionar operandos (Página 2293)

4.1.7 Migrar proyectos de WinCC flexible (Basic)

4.1.7.1 Principios básicos (WinCC flexible)

Migración (WinCC flexible)

Introducción

En WinCC se pueden seguir utilizando proyectos de WinCC flexible. Se soporta la siguiente versión de WinCC flexible:

- WinCC flexible 2008 SP2

En los siguientes capítulos encontrará información sobre qué paneles de operador se soportan y qué condiciones son necesarias para efectuar una migración correctamente.

Los proyectos de ProTool y de versiones anteriores de WinCC flexible no se pueden migrar directamente a WinCC. Si desea seguir utilizando estos proyectos en WinCC, primero deberá migrarlos a una versión compatible de WinCC flexible.

Consulte también

Compatibilidad de objetos en la migración (WinCC flexible) (Página 152)

Migrar proyectos desde WinCC flexible (WinCC flexible) (Página 148)

Compilar y cargar un proyecto migrado (WinCC flexible) (Página 150)

Migración de datos runtime (WinCC flexible) (Página 162)

Migración de proyectos integrados (WinCC flexible) (Página 165)

Paneles de operador soportados (WinCC flexible) (Página 151)

Migración de los tipos de datos (WinCC flexible) (Página 168)

Principios básicos sobre la migración (WinCC flexible)

Introducción

Durante la migración, los datos de un proyecto de WinCC flexible se convierten al nuevo formato de datos de WinCC. No se comprueba si los datos del proyecto que debe migrarse son coherentes. Si al compilar un proyecto de origen se generan errores o advertencias, no se eliminan en la migración. Por lo tanto, el proyecto de origen no deberá contener errores y deberá poderse compilar sin errores antes de la migración. Para la migración, tenga en cuenta el volumen de un proyecto. Para la migración rigen las características funcionales de WinCC. Encontrará más información al respecto en la Ayuda en pantalla, en el capítulo "Visualizar procesos > Características > Engineering System".

Univocidad de los nombres de objeto

En WinCC flexible, las carpetas que contienen los objetos determinan la univocidad de los mismos. La univocidad de los elementos gráficos reunidos en grupos viene determinada por el nombre del grupo.

En WinCC, todo nombre de objeto de un panel de operador debe ser unívoco. En los elementos gráficos, el nombre que aparece en una imagen debe ser unívoco.

Durante la migración se comprueba que los nombres sean unívocos. Si según la nueva regulación un nombre no fuera unívoco, se cambia el nombre del objeto en cuestión. Un objeto cuyo nombre ha sido cambiado recibe el sufijo "#Mign", siendo "n" un número correlativo.

Ejemplo:

En WinCC flexible, las variables que están guardadas en carpetas distintas pueden tener el mismo nombre. En WinCC, el nombre de las variables debe ser unívoco en todo el panel de operador configurado. Por ello, en la migración se cambia el nombre de las variables de carpetas distintas que tienen el mismo nombre.

El nombre de las variables se cambia de acuerdo con el esquema siguiente:

Antes de la migración	Después de la migración
Folder_1/Tag_1	Folder_1/Tag_1
Folder_1/Tag_2	Folder_1/Tag_2
Folder_2/Tag_1	Folder_2/Tag_1#Mig1
Folder_2/Tag_2	Folder_2/Tag_2#Mig1
Folder_3/Tag_1	Folder_3/Tag_1#Mig2
Folder_3/Tag_2	Folder_3/Tag_2#Mig2

Objetos susceptibles de cambio

En caso necesario, se cambia el nombre de los siguientes objetos:

- Imágenes
- Objetos gráficos
- Recetas
- Variables

Cancelar la migración

La migración se cancela en los siguientes casos:

- cuando el proyecto que hay que migrar está abierto en el sistema de ingeniería o en runtime.
- cuando no hay suficiente memoria disponible en el disco duro para crear una copia de la migración del proyecto.
- cuando la migración no puede acceder a la base de datos del proyecto por problemas con el servidor SQL instalado.
- cuando la migración no puede acceder a la base de datos del proyecto porque no tiene la autorización del usuario.
- cuando en un proyecto integrado se selecciona el archivo "*.hmi" para la migración. En un proyecto integrado se debe seleccionar el archivo "*.s7" para la migración.
- cuando se ha creado el proyecto con una versión no soportada por la migración.

Guardar el proyecto en formato de migración

La migración de un proyecto de WinCC flexible no necesariamente debe realizarse íntegramente en el PC en el que está el proyecto. Es posible preparar la migración mientras se guarda el proyecto en formato de migración. Para guardar un proyecto de WinCC flexible en formato de migración, se dispone de la herramienta de migración. La herramienta de migración exporta los datos de ingeniería desde el proyecto de WinCC flexible y los guarda en el formato de migración "*.AM11".

Para la migración propiamente dicha, copie los datos en formato de migración a un PC en el que esté instalado el TIA Portal.

Encontrará más información sobre la herramienta de migración en el capítulo "Auto-Hotspot".

Migrar proyectos desde WinCC flexible (WinCC flexible)

Introducción

Al migrar un proyecto se cargan los datos de un proyecto de WinCC flexible a un proyecto nuevo de WinCC. Por consiguiente, para la migración se crea automáticamente un proyecto nuevo. No es posible migrar a un proyecto existente.

La migración se puede llamar tanto desde la vista del portal como desde la vista del proyecto.

El proyecto deberá migrarse únicamente en un TIA Portal recién abierto.

Encontrará información sobre la migración de un proyecto integrado en el capítulo Migración de proyectos integrados (WinCC flexible) (Página 165).

Si quiere guardar el proyecto únicamente en formato de migración, tiene a su disposición la herramienta de migración. Encontrará más información al respecto en Principios básicos sobre la migración (WinCC flexible).

Requisitos

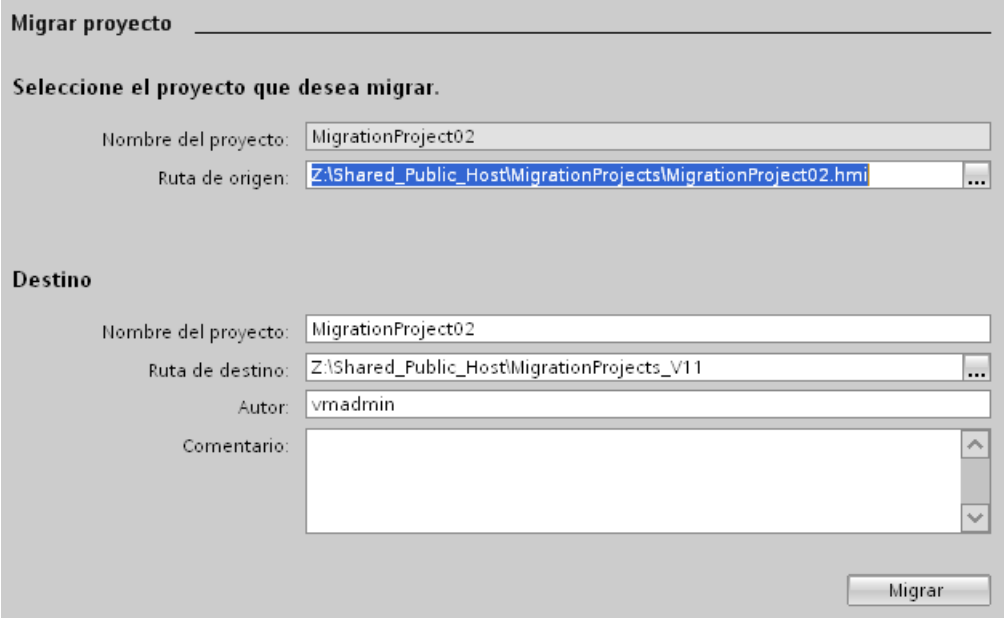
- Debe existir un proyecto de WinCC flexible.
- El proyecto no puede estar abierto en WinCC flexible.

Procedimiento

El proyecto se migra en la vista del portal del siguiente modo:

1. Seleccione la acción "Inicio > Migrar proyecto".

2. En el campo "Ruta de origen" navegue hasta el proyecto que desea migrar.



3. Seleccione el archivo de proyecto de WinCC flexible "*.hmi".

4. En caso necesario, modifique los datos del proyecto que debe crearse. Cambie, p. ej., el nombre o la ruta del proyecto. En el proyecto nuevo se crearán los datos que deben migrarse.
5. Haga clic en "Migrar".
   Se creará un proyecto nuevo y se iniciará la migración de los datos:
   – Se abre la vista del proyecto.
   – El progreso de la migración se indica en una ventana de migración.
   – En la ventana de inspección bajo "Información > General" se muestran informaciones, advertencias y errores correspondientes al proceso de migración.
   – Toda la información relacionada con la migración se guarda en un archivo de informe.
   – Una vez concluida la migración, el proyecto se guarda y se emite un aviso. El aviso contiene un enlace mediante el cual se puede abrir el archivo de informe.

Una vez concluida la migración, en el árbol del proyecto encontrará un dispositivo nuevo para cada panel de operador migrado. Estos dispositivos contienen los datos migrados, p. ej. imágenes, avisos y variables.

Abrir el informe de migración en otro momento

El informe de migración se guarda junto con el proyecto migrado. Puede ver el informe más adelante. El archivo de informe se abre del siguiente modo:

1. Abra la carpeta "Datos comunes > Informes" en el árbol del proyecto. Esta carpeta contiene los informes de todas las migraciones realizadas hasta el momento.
2. Haga doble clic en el informe de migración deseado.
   Se abrirá el informe.

Consulte también

Migración de proyectos integrados (WinCC flexible) (Página 165)

Compilar y cargar un proyecto migrado (WinCC flexible)

Compilar un proyecto migrado

Después de migrar correctamente un proyecto de WinCC flexible, debe volver a compilarse antes de cargarlo en el panel de operador. El proyecto sólo se compilará correctamente si antes de la migración era compilable y no contenía errores.

Si durante la compilación del proyecto migrado se producen errores, éstos se deben eliminar.

Una vez el proyecto haya sido compilado sin errores, cárguelo en el panel de operador.

Ajustes para cargar en el panel de operador

Durante la migración no se aplican los ajustes para cargar el proyecto en el panel de operador. Después de la migración es necesario volver a configurar los ajustes para cargar el proyecto.

Seleccione el panel de operador en el árbol del proyecto y seleccione el comando del menú contextual "Cargar en el equipo > Software (cargar por completo)".Se abre el cuadro de diálogo "Carga ampliada". Configure los ajustes necesarios para la interfaz. Haga clic en el botón "Cargar". El proyecto se vuelve a compilar y se abre el cuadro de diálogo "Cargar vista preliminar".

Expanda la entrada "Sobrescribir" y controle la configuración de las siguientes opciones:

- ¿Desea sobrescribir los datos de administración de usuarios existentes en este equipo?
- ¿Desea sobrescribir los datos de receta existentes en el panel de operador?

Configure las opciones según desee utilizarlas más adelante en el proyecto. Después cargue el proyecto en el panel de operador.

4.1.7.2 Migrar los datos de ingeniería (WinCC flexible)

Paneles de operador soportados (WinCC flexible)

Introducción

Al migrar proyectos de WinCC flexible hay que tener en cuenta que WinCC soporta los siguientes tipos de panel de operador:

- KTP400 Basic mono PN
- KTP400 Basic mono PN Portrait
- KTP600 Basic DP
- KTP600 Basic DP Portrait
- KTP600 Basic PN
- KTP600 Basic PN Portrait
- KTP600 Basic mono PN
- KTP600 Basic mono PN Portrait
- KTP1000 Basic DP
- KTP1000 Basic PN
- TP1500 Basic PN

En WinCC sólo se soportan las funcionalidades que ofrecen estos tipos de paneles.

Si el proyecto de WinCC flexible contiene un panel de operador que no es soportado por WinCC, el proceso de migración se cancelará. Para migrar el proyecto, en WinCC flexible debe asignarse al panel de operador un tipo de panel de operador que sea soportado por WinCC.

Un proyecto de WinCC flexible puede contener funcionalidades que no son soportadas por un Basic Panel, p. ej. por haber cambiado el tipo de panel. Las funcionalidades no soportadas no se migran.

Adaptaciones previas a la migración

Si en el proyecto a migrar se cambió el panel de operador por uno con un tamaño de imagen distinto, hay que volver a compilar el proyecto antes de migrarlo. En el proceso de compilación se adaptan las imágenes y los elementos gráficos a su tamaño.

Consulte también

Compatibilidad de objetos en la migración (WinCC flexible) (Página 152)

Migración (WinCC flexible) (Página 146)

Migración de categorías y grupos de avisos (WinCC flexible) (Página 156)

Migración de contenidos específicos del idioma (WinCC flexible) (Página 158)

Migración de librerías (WinCC flexible) (Página 161)

Migración de variables (WinCC flexible) (Página 155)

Cambios de valor en las propiedades de objeto originados por la migración (WinCC flexible) (Página 154)

Compatibilidad de objetos en la migración (WinCC flexible)

Introducción

Al migrar proyectos de WinCC flexible se migran también todos los datos de configuración que contiene un panel de operador soportado por WinCC. Básicamente se migran íntegramente todos los tipos de objetos y todas las funciones que están disponibles en el nuevo entorno del proyecto y que pueden reproducirse.

Algunos tipos de objetos globales no se migran, p. ej. los diccionarios y librerías globales.

Tipos de objetos soportados

La migración soporta los siguientes tipos de objeto:

- Animaciones
- Planificador de tareas
- Administración de usuarios
- Puntero de área
- Imágenes
- Plantilla
- Tipos de datos
- Listas de funciones
- Listas de gráficos
- Objetos de manejo y visualización
  La migración soporta todos los objetos de manejo y visualización que están disponibles en los paneles de operador soportados.

- Avisos
- Categorías
- Grupos de avisos
- Librería del proyecto
- Idiomas del proyecto
- Recetas
- Idiomas de runtime
- Fuentes de runtime
- Funciones de sistema
- Textos
- Listas de textos
- Variables
- Conexiones

Tipos de objetos no soportados

La migración no soporta los siguientes tipos de objetos y funcionalidades:

- Librerías globales
- Diccionarios
- Versiones de proyecto
- Informe de cambios

Migración de la plantilla de imagen

En WinCC se dispone de un concepto avanzado para trabajar con plantillas de imagen. En WinCC se dispone de una imagen general y varias plantillas para cada dispositivo. Al migrar una plantilla desde WinCC flexible, los objetos que incluye así como las propiedades configuradas en la plantilla se migran a diferentes plantillas de WinCC.

Los objetos siguientes se migran a la "imagen general" de WinCC:

- Ventana de avisos
- Indicador de avisos
- Teclas de función de paneles de operador con teclas de función

Los demás objetos y propiedades se migran a una plantilla de WinCC.

La conexión de los objetos y propiedades con la plantilla correspondiente es adaptada automáticamente por la migración.

Migración de funciones de sistema

En WinCC han cambiado los nombres de algunas funciones de sistema.

Las funciones de sistema cuyos nombres se han modificado se renombran.

Afecta a las siguientes funciones de sistema:

Nombre de la función en WinCC flexible	Nombre de la función en WinCC
IncreaseValue	IncreaseTag
DecreaseValue	DecreaseTag
SetValue	SetTag

Consulte también

Paneles de operador soportados (WinCC flexible) (Página 151)

Cambios de valor en las propiedades de objeto originados por la migración (WinCC flexible) (Página 154)

Cambios de valor en las propiedades de objeto originados por la migración (WinCC flexible)

Introducción

Debido a la unificación de propiedades de objeto de WinCC V7 y WinCC flexible, en el transcurso de la migración puede resultar necesario introducir cambios en algunas propiedades de objeto. La migración calcula los cambios de manera que la representación de los objetos tras la migración sea igual que antes de ella. Estos cambios durante la migración dan lugar a diferentes unidades de medida y valores para algunas propiedades de objeto en la configuración.

Migración de los ajustes de escritura de un objeto

En WinCC V7 y WinCC flexible, para especificar el tamaño de la fuente utilizada en un objeto se usa la unidad de medida "punto". En WinCC, para especificar el tamaño de las fuentes utilizadas en un objeto se usa la unidad de medida "píxel". En la migración se calcula el tamaño de fuente consecuentemente para que la representación de la fuente con un nivel de zoom del 100% tenga el mismo tamaño. A causa de las unidades de medida distintas, las cifras de los tamaños de las fuentes cambian tras la migración.

Ejemplo:

Fuente antes de la migración	Fuente tras la migración
Arial 10 puntos	Arial 13 píxeles
Arial 16 puntos	Arial 21 píxeles
Tahoma 10 puntos	Tahoma 13 píxeles
Tahoma 16 puntos	Tahoma 21 píxeles

Migración de márgenes de objeto

En WinCC flexible, durante la configuración de la representación, algunos objetos permiten introducir valores <0 y >127 para situar los márgenes de objeto. En WinCC, el rango de valores

para los márgenes de objeto está restringido a valores entre 0 y 127. La migración cambia los valores <0 al valor "0" y los valores >127 al valor "127".

Consulte también

Paneles de operador soportados (WinCC flexible) (Página 151)

Compatibilidad de objetos en la migración (WinCC flexible) (Página 152)

Migración de variables (WinCC flexible)

Introducción

Al migrar variables deben tenerse en cuenta ciertas particularidades. Deben diferenciarse los siguientes aspectos:

- Migración de los tipos de datos de variables
- Migración de variables internas
- Migración de variables externas
- Nombres de variables

Migración de los tipos de datos

WinCC dispone en parte de otros tipos de datos y utiliza otros nombres para los tipos de datos que WinCC flexible. Al migrar una variable con estas características, el tipo de datos de WinCC flexible se reproduce con el tipo de datos correspondiente en WinCC. Encontrará más información al respecto en el capítulo Migración de los tipos de datos (WinCC flexible) (Página 168).

Migración de variables

Las variables se migran siempre por completo. Únicamente los nombres de los tipos de datos y de las variables pueden cambiar con la migración.

Migración de los nombres de variables

En WinCC flexible, las variables que se guardan en carpetas distintas pueden tener el mismo nombre. En WinCC, el nombre de las variables debe ser unívoco en todo el panel de operador configurado. Por ello, en la migración se cambia el nombre de las variables de carpetas distintas que tienen el mismo nombre. Encontrará más información al respecto en el capítulo Principios básicos sobre la migración (WinCC flexible) (Página 147).

Consulte también

Principios básicos sobre la migración (WinCC flexible) (Página 147)

Migración de los tipos de datos (WinCC flexible) (Página 168)

Paneles de operador soportados (WinCC flexible) (Página 151)

Migración de categorías y grupos de avisos (WinCC flexible)

Cambio de nombre de las categorías

A diferencia de WinCC flexible, los nombres de las categorías predefinidas no dependen del idioma de interfaz actualmente utilizado. En la migración, los nombres de las categorías se asignan del modo siguiente:

WinCC flexible	WinCC
Errores	Alarms
Sistema	System
Advertencias	Events

Después de la migración es posible cambiar los nombres de las categorías a discreción.

Migración de grupos de avisos

En la migración sólo se migran los grupos de avisos realmente utilizados.

Los grupos de avisos con una ID de 1-31 se migran 1:1.

En WinCC, el sistema crea un grupo de avisos por categoría. A estos grupos de avisos creados por el sistema se les asignan IDs a partir del número 32 y se incrementan correlativamente. Las 4 categorías estándar que existen en todo proyecto de WinCC ocupan automáticamente las IDs 32-35 a través de sus grupos de avisos. Para cada categoría personalizada se crea un grupo de avisos adicional y, para ello, se le asigna una ID adicional. Por consiguiente, en los grupos de avisos con IDs > 31 puede cambiar la ID después de la migración. Asimismo cambia la asignación existente entre nombres de grupos de avisos e IDs.

Ejemplo:

En el ejemplo puede observarse la asignación de IDs en WinCC por la migración.

Grupos de avisos	ID en WinCC flexible	ID en WinCC	
Grupo de avisos 1-16	1-16	1-16	Reservadas para grupos de avisos de avisos de sistema
Grupo de avisos 17-31	17-31	17-31	Grupos de avisos definidos por el usuario
		32-35	Reservadas en WinCC para los grupos de avisos de las categorías predefinidas.
Grupo de avisos 32	32	36	Asignación de ID y grupo de avisos modificada en WinCC
Grupo de avisos 33	33	37	Asignación de ID y grupo de avisos modificada en WinCC

Tenga también en cuenta lo siguiente:

Al migrar grupos de avisos que en apariencia tienen el mismo nombre, la migración adapta los nombres. Esto ocurre cuando el nombre de un grupo contiene, p. ej., un espacio al final del nombre. La migración borra todos los espacios existentes al final del nombre. Si 2 grupos reciben el mismo nombre tras el proceso de borrado, la migración amplía el nombre de los siguientes grupos de avisos con el sufijo "#Mign", siendo "n" un número correlativo.

Ejemplo:

En WinCC flexible existen los siguientes grupos de avisos:

"AlarmGroup_18"

"AlarmGroup_18 " - el nombre del grupo contiene 1 espacio

"AlarmGroup_18 " - el nombre del grupo contiene 2 espacios

El grupo de avisos "AlarmGroup_18" es el grupo con el número más alto.

Resultado después de la migración:

"AlarmGroup_18"

"AlarmGroup_18#Mig1"

"AlarmGroup_18#Mig1.1"

Cambio de nombre de las categorías

A diferencia de WinCC flexible, los nombres de las categorías predefinidas no dependen del idioma de interfaz actualmente utilizado. En la migración, los nombres de las categorías se asignan del modo siguiente:

WinCC flexible	WinCC
Errores	Errors
Sistema	System
Advertencias	Warnings

Después de la migración es posible cambiar los nombres de las categorías a discreción.

Visualización de avisos ALARM_S y avisos SIMATIC SFM

En WinCC flexible es posible activar los discriminadores de avisos ALARM_S en proyectos integrados. La visualización de avisos SIMATIC SFM se activa en WinCC flexible mediante un ajuste separado. En WinCC no existe el ajuste separado para activar la visualización de avisos SIMATIC SFM. Al igual que la visualización de avisos ALARM_S, la visualización de avisos SIMATIC SFM se controla en WinCC exclusivamente activando el discriminador correspondiente.

Como consecuencia del cambio de concepto, puede cambiar la visualización de los avisos después de la migración.

Si en el proyecto de WinCC flexible están activados todos los discriminadores de avisos ALARM_S y está desactivada la visualización de avisos SIMATIC SFM, tras la migración se visualizarán avisos ALARM_S y avisos SIMATIC SFM.

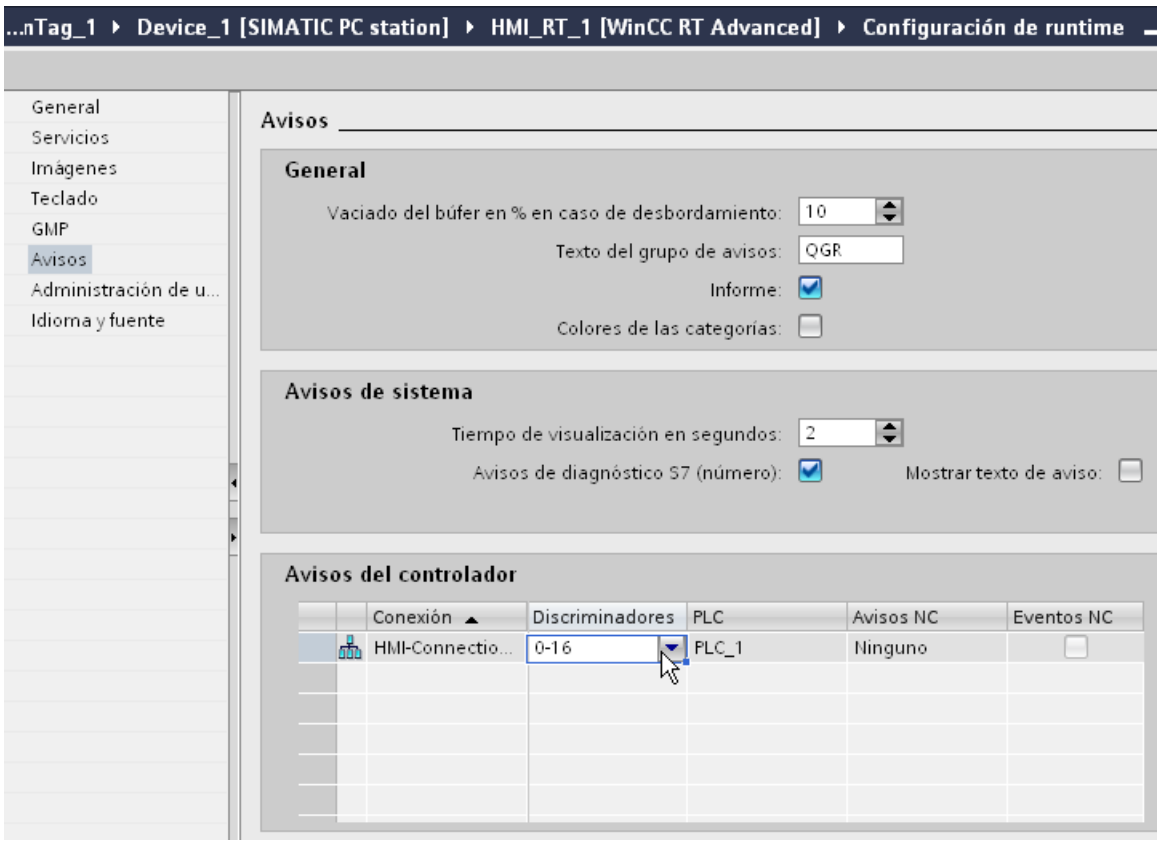
Para que después de la migración se visualicen únicamente avisos ALARM_S, después de la migración a STEP 7 es necesario asignar los avisos SIMATIC SFM a un discriminador no utilizado. Luego deberá desactivar este discriminador en WinCC.

Si en el proyecto de WinCC flexible están desactivados todos los discriminadores de avisos ALARM_S y está activada la visualización de avisos SIMATIC SFM, después de la migración no se visualizarán avisos ALARM_S ni avisos SIMATIC SFM.

Para que después de la migración se visualicen únicamente avisos SIMATIC SFM, después de la migración a STEP 7 es necesario asignar los avisos SIMATIC SFM a un discriminador no utilizado. Luego deberá activar este discriminador en WinCC.

El discriminador depende de los ajustes en STEP 7. En STEP 7 el ajuste estándar de los avisos SIMATIC SFM es el discriminador "0". Para activar la visualización en WinCC debe activarse el discriminador "0".

Los discriminadores se activan en WinCC en la configuración de runtime del panel de operador en cuestión en la categoría "Avisos".



Consulte también

Paneles de operador soportados (WinCC flexible) (Página 151)

Migración de contenidos específicos del idioma (WinCC flexible)

Introducción

WinCC ofrece las mismas posibilidades que WinCC flexible en lo relativo a la configuración en distintos idiomas. Al migrar un proyecto se migran también todos los idiomas soportados por WinCC.

Migración de contenidos específicos del idioma

Al migrar un proyecto se migran también los siguientes contenidos específicos del idioma:

- Idiomas del proyecto
- Textos del proyecto
- Fuentes para la representación en runtime
- Gráficos específicos del idioma

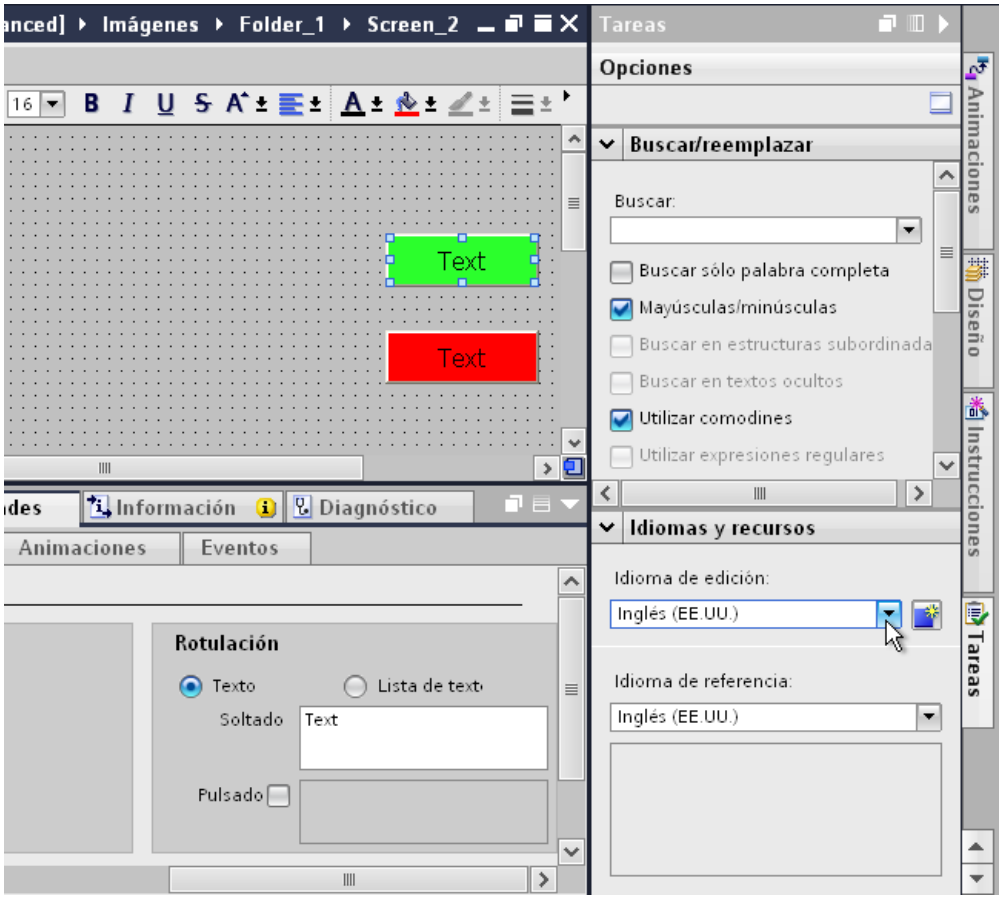
Al migrar contenidos específicos del idioma debe tenerse en cuenta lo siguiente:

- El sistema operativo del PC utilizado para la migración debe soportar los idiomas utilizados en el proyecto.
- En el PC utilizado para la migración deben estar instaladas las fuentes que se utilizan para la representación en runtime.
- La migración no soporta diccionarios.

Idioma de edición de proyectos integrados después de la migración

Al migrar un proyecto integrado, las partes del proyecto que se van a migrar de STEP 7 y WinCC flexible ya vienen con sus respectivos ajustes del idioma de edición. En WinCC ahora ya sólo existe un idioma de edición para todas las partes del proyecto. La migración activa para el proyecto migrado el idioma de edición que estaba activado en STEP 7 antes de la migración. Si este ajuste no coincide con el ajuste de WinCC flexible, los textos configurados ya no podrán verse en WinCC. En los lugares de uso de los textos o bien no aparece ningún texto o bien sólo la entrada "Text". Para poder visualizar los textos, es preciso cambiar el

idioma de edición. Haga clic en el margen derecho del TIA Portal en la Task Card "Tareas" y seleccione en el área "Idiomas y recursos" el idioma de edición correcto.



Idiomas no soportados

La migración de contenidos específicos del idioma depende de si WinCC soporta el idioma respectivo.

Si un proyecto sólo contiene idiomas no soportados por WinCC, este no podrá migrarse.

Si un proyecto contiene idiomas tanto soportados como no soportados, únicamente se migrarán los idiomas soportados. A los idiomas de edición y de referencia se les asigna un idioma soportado.

WinCC no soporta los siguientes idiomas:

- Árabe
- Hebreo
- Dhivehi
- Guyaratí
- Canarés
- Tamil

- Telugú
- Urdú
- Panyabí
- Persa
- Sirio

Consulte también

Paneles de operador soportados (WinCC flexible) (Página 151)

Migración de librerías (WinCC flexible)

Introducción

Al migrar librerías debe hacerse la siguiente distinción:

1. Migración de la librería del proyecto
2. Migración de una librería global

Migración de la librería del proyecto

La librería y los datos del proyecto se guardan conjuntamente en el archivo del proyecto. Por ello, una librería de proyecto migra con las mismas limitaciones que los datos de proyecto.

Migración de una librería global

La migración no soporta librerías globales. No obstante, sí se migran los objetos de librería utilizados en el proyecto. Los objetos de librería se copian en el proyecto al utilizarse y luego ya no tienen ninguna conexión con la librería.

Para poder migrar una librería global, los objetos que contiene deben copiarse o desplazarse a la librería del proyecto. Durante la migración se migran también los objetos. En WinCC desplace los objetos migrados a una nueva librería global, que deberá crearse previamente. Es posible copiar o desplazar tanto objetos individuales como categorías completas de la librería.

Consulte también

Paneles de operador soportados (WinCC flexible) (Página 151)

4.1.7.3 Migrar los datos runtime (WinCC flexible)

Migración de datos runtime (WinCC flexible)

Introducción

Al migrar un proyecto únicamente se migran los datos de configuración. La migración no tiene efecto en los datos runtime. Los datos runtime deben actualizarse después de la migración.

Los datos runtime están compuestos del siguiente modo:

- Proyecto runtime
  El proyecto runtime contiene los datos compilados del proyecto.
- Datos de recetas y la administración de usuarios
  Los datos de recetas y la administración de usuarios son datos que se pueden modificar en runtime.

Migración de los datos runtime

El proyecto runtime se actualiza compilando nuevamente el proyecto en WinCC y cargándolo luego en el panel de operador.

Si los datos de recetas y la administración de usuarios han sido modificados en runtime, debe hacerse una copia de seguridad desde el panel de operador antes de cargar el proyecto migrado. A continuación, cargue el proyecto migrado en el panel de operador. Finalmente, vuelva a cargar en el panel de operador la copia de seguridad de los datos de recetas y la administración de usuarios. Encontrará más información al respecto en el capítulo Auto-Hotspot.

Consulte también

Migración (WinCC flexible) (Página 146)

Hacer una copia de seguridad de los datos de receta y de la administración de usuarios (WinCC flexible) (Página 162)

Restaurar los datos de receta y la administración de usuarios (WinCC flexible) (Página 163)

Hacer una copia de seguridad de los datos de receta y de la administración de usuarios (WinCC flexible)

Introducción

Para poder seguir utilizando los datos de recetas y la administración de usuarios en un proyecto migrado, primero hay que realizar una copia de seguridad de los datos del panel de operador. A continuación, cargue los datos en el proyecto de WinCC migrado. Para crear una copia de seguridad de los datos utilice ProSave.

Requisitos

- El proyecto de WinCC flexible se está ejecutando en el panel de operador en runtime.
- El panel de operador está conectado con el PC en que está instalado ProSave.

Procedimiento

Para crear una copia de seguridad de los datos de recetas y de la administración de usuarios, proceda del siguiente modo:

1. Inicie ProSave.
2. Seleccione el tipo de panel de operador y los parámetros de conexión en la ficha "General".
3. Vaya a la ficha "Backup".
4. Seleccione la entrada "Recetas de la memoria del equipo" en el campo "Clase de datos". No elija "Backup completo" para que al restaurar los datos de recetas pueda seleccionarlos individualmente.
5. En el campo "Guardar como" navegue hasta la ubicación deseada y haga clic en el botón "Inicio de Backup".
   Se crea una copia de seguridad de los datos de recetas.
6. Seleccione la entrada "Administración de usuarios" en el campo "Clase de datos" y haga clic en el botón "Inicio de Backup".
   Se crea una copia de seguridad de la administración de usuarios.

Para obtener información detallada al respecto, consulte la ayuda en pantalla de ProSave.

Procedimiento alternativo

ProSave se instala junto con WinCC flexible de forma estándar. En el PC de configuración se dispone en WinCC flexible de toda la gama de funciones de ProSave con el comando "Proyecto > Transferencia".

Como alternativa se puede crear una copia de seguridad de los datos de recetas y de la administración de usuarios con el software ProSave, que está integrado en WinCC flexible. Inicie WinCC flexible y elija el comando de menú "Proyecto > Transferencia > Copia de seguridad". Haga una copia de seguridad de los datos de recetas y la administración de usuarios conforme a lo descrito en los pasos 4-6.

Consulte también

Migración de datos runtime (WinCC flexible) (Página 162)

Restaurar los datos de receta y la administración de usuarios (WinCC flexible) (Página 163)

Restaurar los datos de receta y la administración de usuarios (WinCC flexible)

Introducción

Para volver a utilizar la copia de seguridad de los datos de recetas y de la administración de usuarios después de la migración, primero se debe compilar el proyecto migrado y luego se

debe cargar en el panel de operador. A continuación, transfiera la copia de seguridad de los datos al panel de operador. Para restaurar los datos utilice ProSave.

Requisitos

- El proyecto migrado ha sido transferido al panel de operador y se está ejecutando en runtime.
- El panel de operador está conectado con el PC en que está instalado ProSave.

Procedimiento

Para cargar la copia de seguridad de los datos de recetas y de la administración de usuarios en el panel de operador, proceda del siguiente modo:

1. Inicie ProSave.
2. Seleccione el tipo de panel de operador y los parámetros de conexión en la ficha "General".
3. Vaya a la ficha "Restore".
4. En el campo "Abrir..." navegue hasta la ubicación de la copia de seguridad de los datos de recetas y seleccione el archivo.
5. Haga clic en el botón "Iniciar Restore".
   Los datos de recetas se transfieren al panel de operador.
6. Para restaurar la administración de usuarios repita los pasos 4-5.
   La administración de usuarios se transfiere al panel de operador.

Para obtener información detallada al respecto, consulte la Ayuda en pantalla de ProSave.

Procedimiento alternativo

ProSave se instala junto con WinCC de forma estándar. En el PC de configuración se dispone en WinCC de toda la gama de funciones de ProSave con el comando "Proyecto > Transferencia".

Como alternativa se pueden restaurar los datos de recetas y la administración de usuarios con el software ProSave, que está integrado en WinCC. Inicie WinCC y en la barra de menús elija el comando "Online > Mantenimiento de paneles de operador > Restaurar". Restaure los datos de recetas y la administración de usuarios conforme a o descrito en los pasos 4-6.

Consulte también

Migración de datos runtime (WinCC flexible) (Página 162)

Hacer una copia de seguridad de los datos de receta y de la administración de usuarios (WinCC flexible) (Página 162)

4.1.7.4 Migrar los proyectos integrados (WinCC flexible)

Migración de proyectos integrados (WinCC flexible)

Introducción

Los controladores y paneles de operador contenidos en un proyecto integrado en STEP 7 están conectados entre sí por configuración. Asimismo los datos de configuración de WinCC flexible y STEP 7 están interconectados. Al migrar un proyecto integrado también se migra todo el proyecto con las partes correspondientes a WinCC flexible y STEP 7. Las conexiones se conservan.

Nota

Antes de migrar un proyecto integrado, este proyecto debe compilarse y guardarse en WinCC flexible. Una compilación sin errores garantiza que los datos están sincronizados en WinCC flexible y en STEP 7.

Migración de un proyecto integrado

Al migrar un proyecto integrado, para la parte correspondiente a WinCC flexible rigen las mismas condiciones que para la migración de un proyecto de WinCC flexible no integrado. Los objetos y propiedades contenidas en la parte correspondiente a WinCC flexible deben ser soportados por WinCC, p. ej. el panel de operador o el driver de comunicación. En una conexión proyectada debe estar activada la propiedad "Online". Una conexión que tenga la propiedad "Online" desactivada no será migrada.

Aparte de las condiciones que rigen para la parte correspondiente a WinCC flexible, rigen otras condiciones para la parte correspondiente a STEP 7 del proyecto integrado. Los objetos y propiedades incluidos en la parte correspondiente a STEP 7 V5.4 SP5 o V5.5 deben ser soportados en STEP 7. Encontrará más información al respecto en la documentación de STEP 7.

Para poder migrar completamente y volver a procesar un proyecto integrado, en el PC utilizado para la migración deben estar instalados los siguientes componentes:

- STEP 7 V5.4 SP5 o STEP 7 V5.5
- WinCC flexible 2008 SP2 o WinCC flexible 2008 SP3
- STEP 7

Si quiere guardar el proyecto únicamente en formato de migración, tiene a su disposición la herramienta de migración. Encontrará más información al respecto en Principios básicos sobre la migración (WinCC flexible) (Página 147).

Un proyecto integrado siempre se migra completamente. Si únicamente desea migrar el proyecto de WinCC flexible allí contenido, antes de la migración deberá separarlo del proyecto de STEP 7. Para separar el proyecto de la integración, abra el proyecto en STEP 7 V5.4 SP5 o V5.5. Abra el proyecto de WinCC flexible en el SIMATIC Manager. El proyecto se abre con

WinCC flexible. Elija el comando "Proyecto > Copiar proyecto de STEP 7" en WinCC flexible. WinCC flexible guarda una copia no integrada del proyecto.

Consulte también

Principios básicos sobre la migración (WinCC flexible) (Página 147)

Migrar proyecto integrado

Introducción

Al migrar un proyecto integrado se migran tanto las partes correspondientes al proyecto de WinCC flexible como las correspondientes al proyecto de STEP 7. Por ello, para efectuar la migración debe elegirse el archivo de proyecto con la extensión "*.s7p". Durante la migración se copian los datos del proyecto existente y se migran a un nuevo proyecto. No es posible migrar a un proyecto existente.

La migración se puede llamar tanto desde la vista del portal como desde la vista del proyecto.

El proyecto deberá migrarse únicamente en un TIA Portal recién abierto.

Si quiere guardar el proyecto únicamente en formato de migración, tiene a su disposición la herramienta de migración. Encontrará más información en Principios básicos de la migración (WinCC flexible).

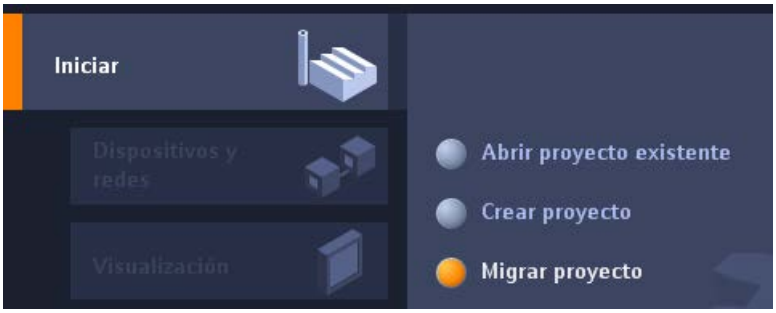
Requisitos

- STEP 7 V5.4 SP5 o STEP 7 V5.5 y todos los paquetes opcionales utilizados están instalados.
- STEP 7 y todos los paquetes opcionales utilizados están instalados.
- El TIA Portal está recién iniciado.
- No debe haber ningún proyecto abierto en WinCC.
- Se ha creado un proyecto integrado.
- El proyecto integrado no está abierto.

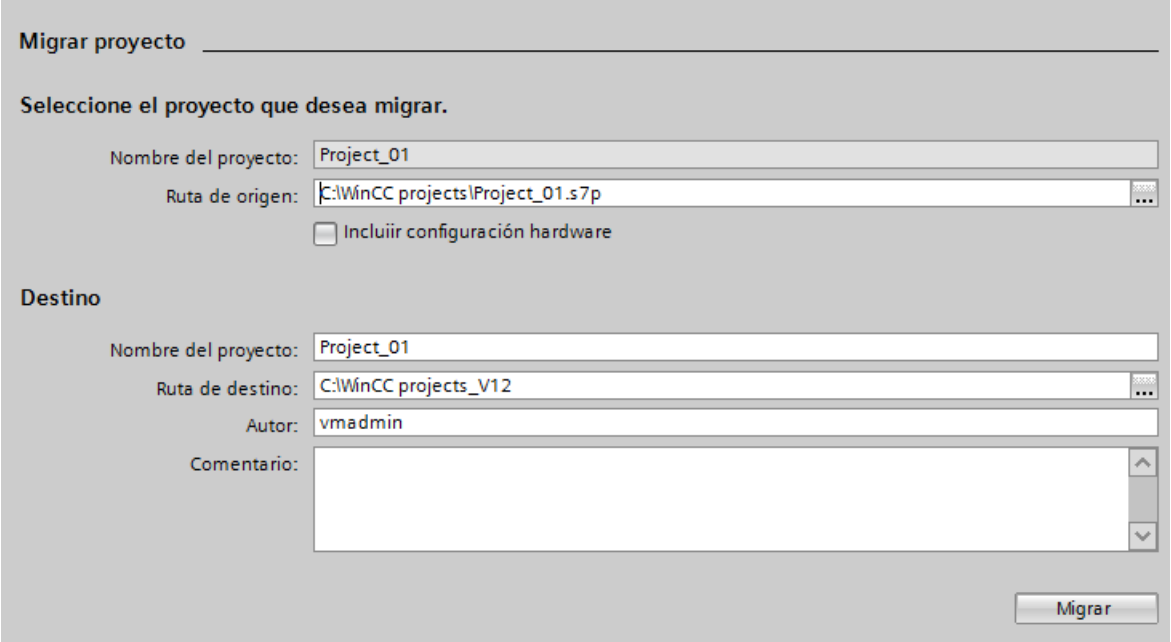
Procedimiento

Para migrar un proyecto integrado a la vista del portal proceda del siguiente modo:

1. Seleccione la acción "Inicio > Migrar proyecto".



2. En el campo "Ruta de origen" navegue hasta el proyecto que desea migrar.



3. Seleccione el archivo de proyecto "*.s7p".
4. En caso necesario, modifique los datos del proyecto que debe crearse. Cambie, p. ej., el nombre o la ruta del proyecto. En el proyecto nuevo se crearán los datos que deben migrarse.
5. Para migrar el proyecto con configuración hardware, active "Incluir la configuración hardware".

6. Haga clic en "Migrar".
   Se creará un proyecto nuevo y se iniciará la migración de los datos:
   - Se abre la vista del proyecto.
   - El progreso de la migración se indica en una ventana de migración.
   - En la ventana de inspección bajo "Información > General" se muestran informaciones, advertencias y errores correspondientes al proceso de migración.
   - Toda la información relacionada con la migración se guarda en un archivo de informe.
   - Una vez concluida la migración, aparece un aviso. El aviso contiene un enlace mediante el cual se puede abrir el archivo de informe.
7. Guarde el proyecto al finalizar la migración.

Una vez concluida la migración, en el árbol del proyecto encontrará un dispositivo nuevo para cada panel de operador migrado y para cada controlador migrado. Estos dispositivos contienen los datos migrados.

Abrir el informe de migración en otro momento

El informe de migración se guarda junto con el proyecto migrado. Puede ver el informe más adelante. El archivo de informe se abre del siguiente modo:

1. Abra "Datos comunes > Informes" en el árbol del proyecto.
2. Haga doble clic en el archivo de informe. Se abre el informe de migración.

Consulte también

Principios básicos sobre la migración (WinCC flexible) (Página 147)

4.1.7.5 Referencia (WinCC flexible)

Migración de los tipos de datos (WinCC flexible)

Introducción

Con el fin de armonizar los tipos de datos utilizados por controladores y sistemas HMI, se cambia el nombre de algunos tipos de datos de las variables HMI internas. Los nombres se otorgan conforme a la IEC. Dado que sólo cambian los nombres, en lo que respecta a las variables internas la configuración no sufre cambios.

La tabla siguiente describe la reproducción de los tipos de datos de WinCC flexible en los tipos de datos de WinCC.

Migración de los tipos de datos

Los tipos de datos internos se reproducen unos sobre otros durante la migración, del siguiente modo:

Tipos de datos internos de WinCC flexible	Tipos de datos internos de WinCC
Bool	Bool
Char	SInt
Byte	USInt
Int	Int
UInt	UInt
Long	DInt
ULong	UDInt
Float	Real
Double	LReal
String	WString
DateTime	DateTime

Migración de los tipos de datos externos

En las siguientes páginas se explica cómo se representan los tipos de datos de los drivers de comunicación disponibles.

Consulte también

Migración (WinCC flexible) (Página 146)

Migración de los tipos de datos de Allen-Bradley DF1 (WinCC flexible) (Página 170)

Migración de los tipos de datos de Allen-Bradley Ethernet IP (WinCC flexible) (Página 170)

Migración de los tipos de datos de Mitsubishi FX (WinCC flexible) (Página 171)

Migración de los tipos de datos de Modicon Modbus (WinCC flexible) (Página 171)

Migración de los tipos de datos de Modicon Modbus TCP/IP (WinCC flexible) (Página 172)

Migración de los tipos de datos de Omron Hostlink/Multilink (WinCC flexible) (Página 172)

Migración de los tipos de datos de SIMATIC S7 200 (WinCC flexible) (Página 173)

Migración de los tipos de datos de SIMATIC S7 300/400 (WinCC flexible) (Página 173)

Migración de los tipos de datos de Allen-Bradley DF1 (WinCC flexible)

Migración de los tipos de datos Allen-Bradley DF1

Los tipos de datos del driver de comunicación Allen-Bradley DF1 se reproducen de la siguiente manera durante la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
ASCII	ASCII
BCD4	UInt
BCD8	UDInt
Bit	Bool
Int	Int
Long	DInt
Real	Real
UInt	UInt
ULong	UDInt

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de Allen-Bradley Ethernet IP (WinCC flexible)

Migración de los tipos de datos Allen-Bradley Ethernet IP

Los tipos de datos del driver de comunicación Allen-Bradley Ethernet IP se reproducen de la siguiente manera durante la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
Bool	Bool
DInt	DInt
Int	Int
Real	Real
SInt	SInt
String	String
UDInt	UDInt
UInt	UInt
USInt	USInt

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de Mitsubishi FX (WinCC flexible)

Migración de los tipos de datos Mitsubishi FX

Los tipos de datos del driver de comunicación Mitsubishi FX se reproducen de la siguiente manera durante la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
12 Bit Block	12-Bit Block
16 Bit Block	16-Bit Block
20 Bit Block	20-Bit Block
24 Bit Block	24-Bit Block
28 Bit Block	28-Bit Block
32 Bit Block	32-Bit Block
4 Bit Block	4-Bit Block
8 Bit Block	8-Bit Block
Bit	Bool
Double	DWord
IEEE-Float	Real
String	String
Word	Word

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de Modicon Modbus (WinCC flexible)

Migración de los tipos de datos Modicon Modbus

WinCC no soporta el driver de comunicación Modicon Modbus ; será sustituido durante la migración por el driver Modicon Modbus RTU. Los tipos de datos del driver de comunicación Modicon Modbus se reproducen de la siguiente manera en la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
+/-Double	+/- Double
+/-Int	+/- Int
16 Bit Group	16 Bit Group
ASCII	ASCII
Bit	Bit
Double	Double
Float	Float
Int	Int

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de Modicon Modbus TCP/IP (WinCC flexible)

Migración de los tipos de datos Modicon Modbus TCP/IP

Los tipos de datos del driver de comunicación Modicon Modbus TCP/IP se reproducen de la siguiente manera durante la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
+/-Double	+/- Double
+/-Int	+/- Int
16 Bit Group	16 Bit Group
ASCII	ASCII
Bit	Bit
Double	Double
Float	Float
Int	Int

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de Omron Hostlink/Multilink (WinCC flexible)

Migración de los tipos de datos Omron Hostlink/Multilink

WinCC no soporta el driver de comunicación Omron Hostlink/Multilink ; será sustituido durante la migración por el controlador Omron Host Link . Los tipos de datos del driver de comunicación Omron Hostlink/Multilink se reproducen de la siguiente manera durante la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
+/-DEC	Int
+/-LDEC	DInt
ASCII	String
BIN	Bool
BYTE	Byte
DEC	UInt
IEEE	Real
LDEC	UDInt

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de SIMATIC S7 200 (WinCC flexible)

Migración de los tipos de datos SIMATIC S7 200

Los tipos de datos del driver de comunicación SIMATIC S7 200 se reproducen de la siguiente durante la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
Bool	Bool
Byte	Byte
Char	Char
DInt	DInt
DWord	DWord
Int	Int
Real	Real
StringChar	StringChar
Timer	Timer
Word	Word

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

Migración de los tipos de datos de SIMATIC S7 300/400 (WinCC flexible)

Migración de los tipos de datos SIMATIC S7 300/400

Los tipos de datos del driver de comunicación SIMATIC S7 300/400 se reproducen de la siguiente manera en la migración a WinCC:

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
Bool	Bool
Byte	Byte
Char	véase más abajo
Counter	véase más abajo
Date	Date
Date and Time	Date_And_Time
DInt	DInt
DWord	DWord
Int	Int

Tipo de datos en WinCC flexible	Tipo de datos en WinCC
Real	Real
String	String
StringChar	véase más abajo
Time	Time
Time of Day	Time_Of_Day
Timer	véase más abajo
Word	Word

Particularidades de algunos tipos de datos

Al migrar variables externas que contienen tipos de datos de un controlador SIMATIC S7-300/400, deben tenerse en cuenta ciertas particularidades.

Reproducción del tipo de datos S7 "Char"

Según la especificación, "Char" el tipo de datos S7 es un tipo de datos para representar caracteres. Dado que este tipo de datos ha sido utilizado frecuentemente para leer y escribir valores numéricos, éste se reproduce en WinCC en el tipo de datos S7 "Byte". Si este caso se da durante la migración, la ventana de resultados mostrará un aviso.

Si el tipo de datos S7 "Char" se utiliza para cifras y en la ubicación se han configurado números negativos, en caso de reproducción en el tipo de datos S7 "Byte" se produce un error. El tipo de datos S7 "Byte" no puede reproducir números negativos. Para eliminar el error se debe adaptar en consecuencia la configuración. Para el procesamiento de números positivos y negativos, utilice un tipo de datos con signo, p. ej., el tipo de datos "Int".

Si el tipo de datos S7 "Char" se utiliza de todas formas para reproducir caracteres, la configuración se debe modificar tras la migración. Para representar caracteres utilice el tipo de datos "String".

Al migrar un proyecto integrado, el tipo de datos "Char" también se migra en WinCC al tipo de datos "Byte". Si hay una variable PLC conectada, el tipo de datos "Char" permanece como "Char". La modificación del tipo de datos de la variable HMI no provoca la migración del direccionamiento simbólico de las variables afectadas. Tras la migración, las variables están conectadas entre sí mediante direcciones absolutas y siguen funcionando. Si desea restablecer el direccionamiento simbólico, la configuración se debe modificar en consecuencia tras la migración.

Reproducción de una matriz del tipo de datos S7 "Char"

Una matriz del tipo de datos S7 "Char" se reproduce en la migración en una matriz del tipo de datos "Byte".

Si una matriz del tipo de datos S7 "Char" se utiliza para cifras y en la ubicación se han configurado números negativos, en caso de reproducción en una matriz del tipo de datos S7 "Byte" se produce un error. El tipo de datos S7 "Byte" no puede reproducir números negativos. Para eliminar el error se debe adaptar en consecuencia la configuración. Para el procesamiento de números positivos y negativos, utilice un tipo de datos con signo, p. ej., el tipo de datos "Int".

Reproducción del tipo de datos S7 "Counter"

Una variable externa con el tipo de datos S7 "Counter" con dirección Counter se reproduce en el tipo de datos S7 "Counter". La dirección se mantiene.

Cuando una variable externa con el tipo de datos S7 "Counter" direcciona un bloque de datos o una dirección de marcas, se reproducirá en el tipo de datos S7 "Word". La dirección se mantiene. La migración establece la codificación en "SimaticBCDCounter".

El rango de valores del tipo de datos S7 "Counter" es de 0-999. Al transferir valores con el tipo de datos S7 "Word" , se puede rebasar por exceso el rango de valores en el lado del controlador. Asegúrese de que se respete el rango de valores.

Ejemplo:

WinCC flexible

Variable	Tipo de datos S7	Dirección	Comentario
Counter_Actual_Value	Counter	C10	Valor de contador codificado en BCD
Counter_Setpoint_Value	Counter	DB10.DBW200	Valor de contador codificado en BCD
Counter_Setpoint_Value#2	Counter	MW20	Valor de contador codificado en BCD

WinCC

Variable	Tipo de datos S7	Dirección	Codificación	Comentario
Counter_Actual_Value	Counter	%C10	<Standard>	Valor de contador codificado en BCD
Counter_Setpoint_Value	Word	%DB10.%DBW200	SimaticBCDCounter	Valor de contador codificado en BCD
Counter_Setpoint_Value#2	Word	%MW20	SimaticBCDCounter	Valor de contador codificado en BCD

Reproducción del tipo de datos "StringChar"

En WinCC no hay ningún tipo de datos correspondiente sobre el que se pueda reproducir el tipo de datos "StringChar" . La reproducción en WinCC depende de la propiedad "Longitud" del tipo de datos S7.

Una variable del tipo de datos "StringChar" con la propiedad "Longitud" > 1 se migra a una matriz del tipo de datos S7 "Char". La longitud de la matriz se corresponde con la longitud del tipo de datos "StringChar" configurado originariamente.

Si la propiedad "Longitud" = 1, el tipo de datos en WinCC se reproduce en una matriz del tipo de datos S7 "Char" con la longitud = 1. La expresión para una matriz con un elemento es "Array[0 ..0] of Char".

Reproducción del tipo de datos S7 "Timer"

Una variable externa con el tipo de datos S7 "Timer" con dirección Temporizador se reproduce en el tipo de datos S7 "Timer". La dirección se mantiene.

Cuando una variable externa con el tipo de datos S7 "Timer" direcciona un bloque de datos o una dirección de marcas, se reproducirá en el tipo de datos S7 "S5 Time". La dirección se mantiene.

Ejemplo:

WinCC flexible

Variable	Tipo de datos S7	Dirección	Comentario
Timer_Actual_Value	Timer	T10	Valor Timer codificado en BCD
Timer_Setpoint_Value	Timer	DB10.DBW200	Valor Timer codificado en BCD
Timer_Setpoint_Value#2	Timer	MW20	Valor Timer codificado en BCD

WinCC

Variable	Tipo de datos S7	Dirección	Comentario
Timer_Actual_Value	Timer	%T10	Valor Timer codificado en BCD
Timer_Setpoint_Value	S5Time	%DB10.%DBW200	Valor Timer codificado en BCD
Timer_Setpoint_Value#2	S5Time	%MW20	Valor Timer codificado en BCD

Consulte también

Migración de los tipos de datos (WinCC flexible) (Página 168)

4.1.8 Migrar proyectos integrados

4.1.8.1 Migración de un proyecto integrado

Introducción

En los proyectos integrados, se utilizan conjuntamente en un proyecto controladores SIMATIC y componentes WinCC. Al migrar un proyecto integrado, también se migra todo el proyecto con las partes correspondientes a WinCC y STEP 7. Las conexiones configuradas entre el controlador y la visualización se conservan.

Migración de un proyecto integrado

Al migrar un proyecto integrado, para la parte correspondiente a STEP 7 rigen las mismas condiciones que para la migración de un proyecto STEP 7 no integrado. Los objetos y propiedades que contiene la parte correspondiente a WinCC también deben ser soportados por WinCC (TIA Portal). Encontrará más información al respecto en la documentación de WinCC.

Recuerde también que el proyecto de partida debe compilarse antes de la migración.

Para poder migrar completamente un proyecto integrado, en la programadora/en el PC utilizado para la migración deben estar instalados los siguientes componentes:

- STEP 7 V5.4 SP5 o STEP 7 V5.5
- WinCC V7.0 SP3 o WinCC flexible 2008 SP2 y SP3

Para poder reeditar completamente un proyecto integrado, en el PC utilizado para tal finalidad deben estar instalados los siguientes componentes:

- STEP 7 Professional V12 SP1
- WinCC Basic, WinCC Comfort/Advanced o WinCC Professional, dependiendo de los componentes utilizados

Uso de la herramienta de migración

En las siguientes condiciones es necesario utilizar la herramienta de migración:

- El proyecto de migración no se encuentra en la misma programadora o el mismo PC que la instalación del TIA Portal.
- En el proyecto de partida hay dispositivos SCADA. Dichos dispositivos solo pueden migrarse con la herramienta de migración.
- WinCC Professional V12 y STEP 7 con WinCC V7.0 SP3 no pueden estar instalados en la misma programadora o el mismo PC. Por ello, los proyectos migrados con componentes WinCC V7.0 SP3 deben prepararse para la migración con la herramienta de migración.

Migración de la parte de un proyecto integrado correspondiente a STEP 7

Un proyecto integrado siempre se migra completamente. Partes individuales de este no pueden migrarse por separado. Únicamente se puede migrar el proyecto STEP 7 contenido por sí solo si primero se borran todas las estaciones HMI del SIMATIC Manager y, después, se vuelve a compilar el proyecto en NetPro.

De manera alternativa, el proyecto se puede abrir en una instalación de STEP 7 V5.4 SP5 ó V5.5 sin una instalación de WinCC. Guarde el proyecto de nuevo seleccionando la opción "Reorganización". Al guardar la copia, las partes de WinCC se borran automáticamente.

A continuación se migra el proyecto de STEP 7 sin el proyecto de WinCC.

Migración de un proyecto integrado con la configuración hardware

En proyectos integrados, los dispositivos HMI también se migran si la configuración hardware no se incluye en la migración. El componente de STEP 7 de la configuración hardware, incluida la configuración de red, conexiones y alarmas, solo se migra si la configuración hardware se

incluye en la migración. En caso contrario, para los dispositivos STEP 7 se crean módulos no especificados, que tras la migración deben volver a convertirse en un módulo adecuado.

Los módulos HMI que están insertados en una estación PC se convierten en una estación propia durante la migración. Si realiza la migración excluyendo la configuración hardware, el proyecto migrado contiene una estación PC SIMATIC no especificada y una estación PC SIMATIC con los dispositivos HMI. Las referencias a dispositivos HMI no se transfieren en la migración. Si la configuración hardware está incluida, el proyecto migrado contiene dos estaciones separadas, la estación HMI y la estación PC.

Ubicación de un proyecto de WinCC integrado

Cuando se migra un proyecto integrado, la parte correspondiente a HMI incluida en él tiene que hallarse en la misma programadora/en el mismo PC que la parte del proyecto correspondiente a STEP 7. Si la parte correspondiente a HMI se encuentra en otra programadora, solo se migra la parte correspondiente a STEP 7.

Objetos no soportados

La migración no soporta los siguientes componentes:

- Multiproyecto STEP 7
  No se puede migrar un multiproyecto STEP 7. La migración cancela la operación.
- Central Archive Server - CAS
  Cuando un proyecto integrado contiene un CAS, la migración se lleva a cabo, pero no se migran los datos del servidor CAS.

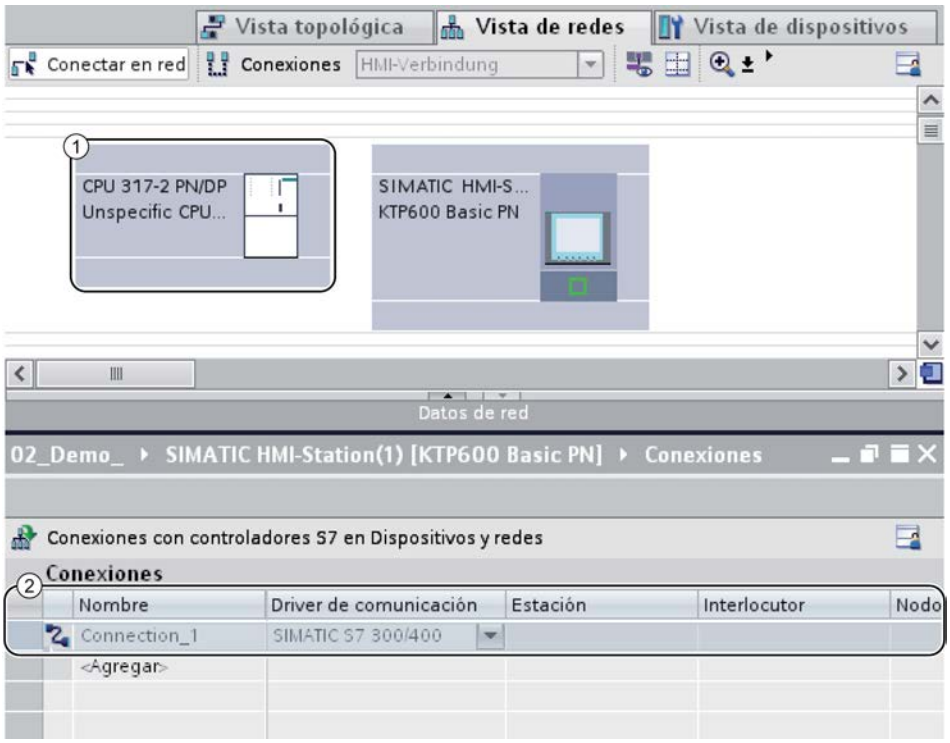
Consulte también

Reajustar proyectos integrados (Página 178)

4.1.8.2 Reajustar proyectos integrados

Si se ha migrado un proyecto integrado sin configuración de hardware, las CPUs no especificadas se utilizan en lugar de las CPUs del proyecto original. Como no puede haber una conexión entre una CPU no especificada y un dispositivo HMI, las conexiones del proyecto de origen también se aplican siempre sin especificar.

La siguiente imagen muestra el estado en un proyecto de ejemplo después de una migración sin configuración de hardware.



① La CPU 317-2 PN/DP original se ha sustituido durante la migración por una CPU no especificada.

② La conexión entre la CPU y el dispositivo HMI también es "no especificada" y debe sustituirse.

Procedimiento

Para seguir utilizando un proyecto integrado después de la migración, proceda del siguiente modo:

1. Convierta nuevamente los dispositivos no especificados en dispositivos apropiados.
2. Establezca la conexión HMI integrada entre el dispositivo HMI y el PLC.
3. Conecte todas las variables HMI con la conexión integrada recién creada.
4. Restablezca la conexión entre las variables HMI y las variables PLC.
5. Borre la conexión HMI no integrada.

En los siguientes capítulos se describen los diferentes pasos a partir de un proyecto de ejemplo.

Consulte también

Convertir CPUs no especificadas en CPUs especificadas (Página 180)

Crear una conexión HMI integrada (Página 181)

Volver a enlazar variables HMI (Página 183)

Borrar una conexión no especificada (Página 184)

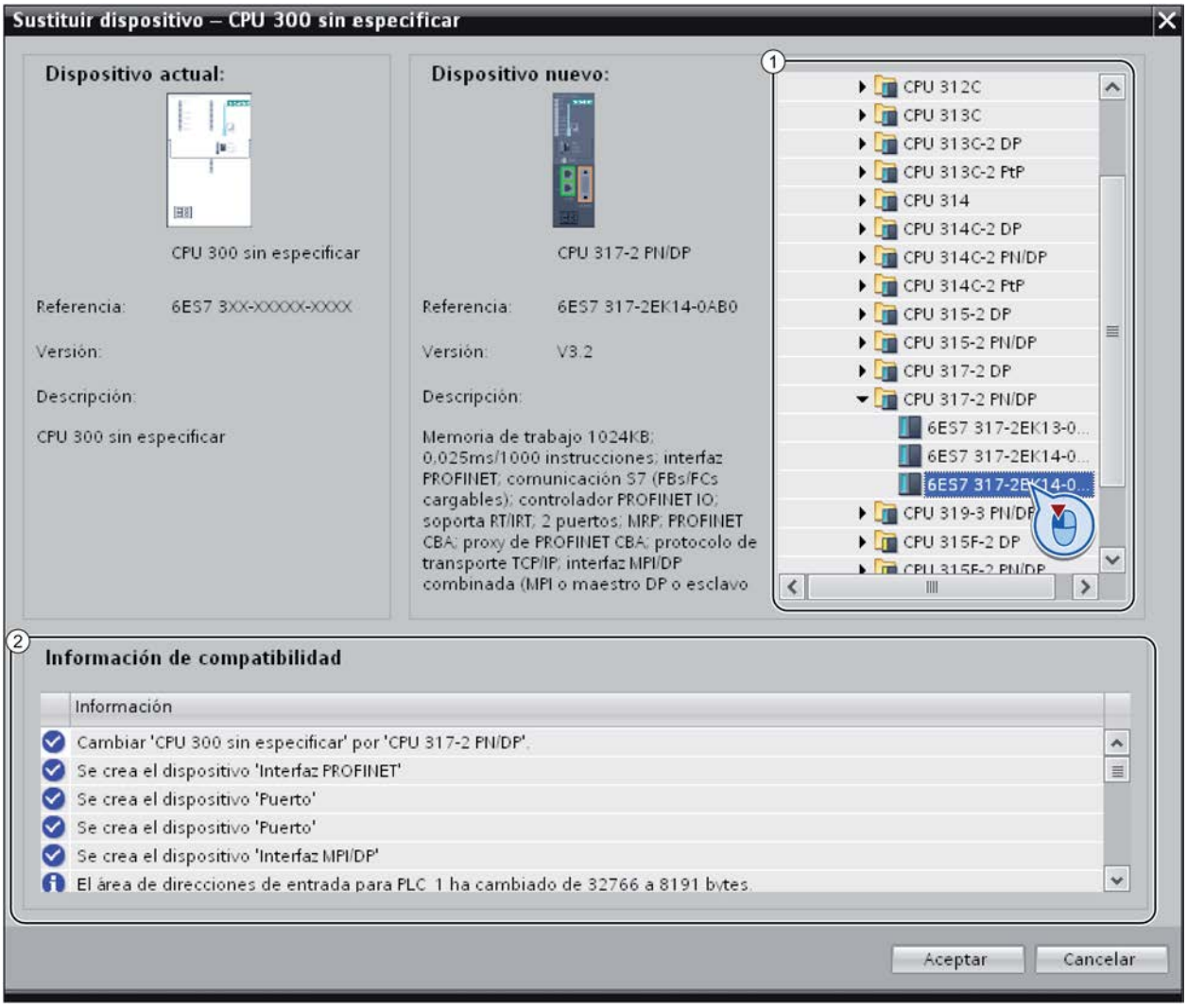
4.1.8.3 Convertir CPUs no especificadas en CPUs especificadas

El primer paso después de la migración sin configuración de hardware es la conversión de las CPUs no especificadas en CPUs especificadas. Las CPUs no especificadas deben entenderse como comodines para determinadas CPUs del catálogo de hardware, que aún no se conocen. Es posible definir parámetros generales y referenciar las CPUs ya en el programa de usuario. No obstante, el proyecto no será totalmente funcional mientras la CPU no especificada siga sin especificarse.

Especificar una CPU mediante la sustitución de módulos

Para especificar una CPU no especificada mediante la sustitución de módulos, proceda del siguiente modo:

1. Seleccione la CPU no especificada en la vista de redes o dispositivos.
2. En el menú contextual, elija el comando "Sustituir dispositivo".
   Se abre el cuadro de diálogo "Sustituir dispositivo".



3. En el elemento "Dispositivo nuevo" de la estructura de árbol seleccione el módulo que reemplazará a la CPU no especificada. (Rango 1)
   En "Información de compatibilidad" se pueden obtener instrucciones sobre el grado de compatibilidad que existe entre la CPU elegida y la configuración del proyecto de origen. (Rango 2)
4. Haga clic en "Aceptar".
5. Repita los pasos anteriores para todas las CPUs no especificadas.

Consulte también

Crear una conexión HMI integrada (Página 181)

4.1.8.4 Crear una conexión HMI integrada

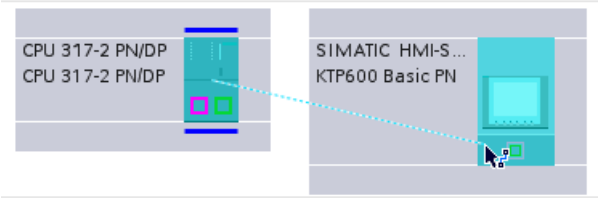
Una vez especificada la CPU no especificada, cree una conexión con el dispositivo HMI.

Procedimiento

Para crear una conexión gráficamente, proceda del siguiente modo:

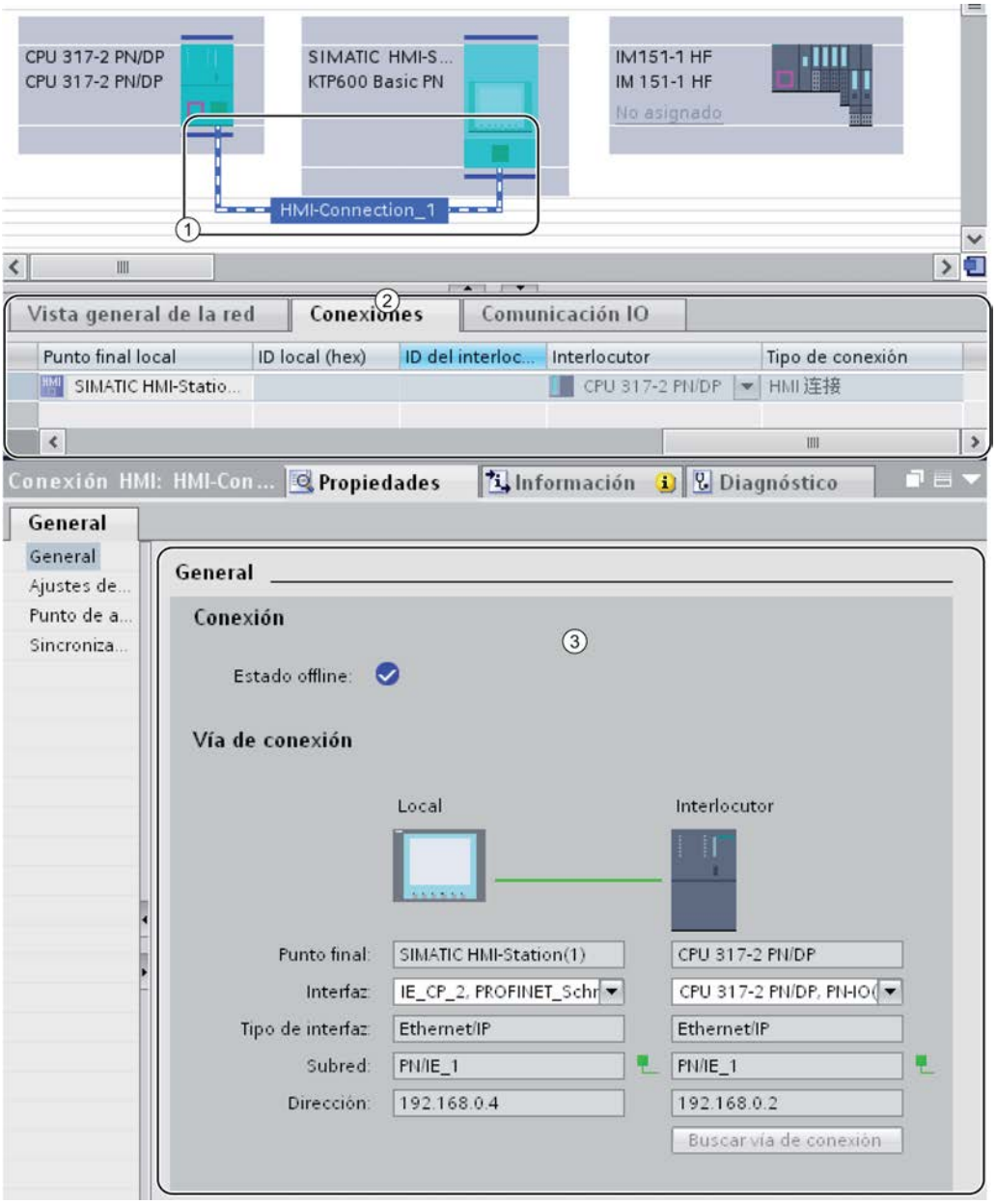
1. Haga clic en el símbolo "Conexiones" de la barra de herramientas. De esta forma se activa el modo de conexión.

2. Seleccione el tipo de conexión "Conexión HMI" en la lista desplegable adjunta.
   En la vista de redes se resaltan en color las CPUs y los dispositivos HMI que pueden utilizarse como conexión HMI.
3. A continuación es posible determinar automáticamente la vía de conexión, o seleccionar expresamente una vía de conexión a través de determinadas interfaces:
   - Determinar automáticamente la vía de conexión
     Seleccione la CPU de la que debe partir una conexión. Arrastre el ratón al componente de destino. Confirme el punto final de la conexión mediante un nuevo clic.
     Alternativa: Mientras mantiene presionada la tecla "Mayús", seleccione además el componentes de destino y elija con el botón derecho del ratón el comando "Agregar conexión".
   - Seleccionar la vía de conexión explícita de interfaz a interfaz
     En la interfaz de subred, haga clic en el dispositivo del que deba partir una conexión. A continuación, arrastre el puntero mientras mantiene pulsado el botón del ratón hasta la interfaz deseada del dispositivo de destino y, por último, suelte el botón del ratón.



Resultado

La imagen siguiente muestra el estado después de crear la conexión integrada:



① La conexión integrada HMI queda creada y se representa resaltada en la vista de redes.

② La conexión se muestra en la tabla de conexiones de los componentes.

③ La conexión puede editarse en las propiedades de conexión.

Consulte también

Volver a enlazar variables HMI (Página 183)

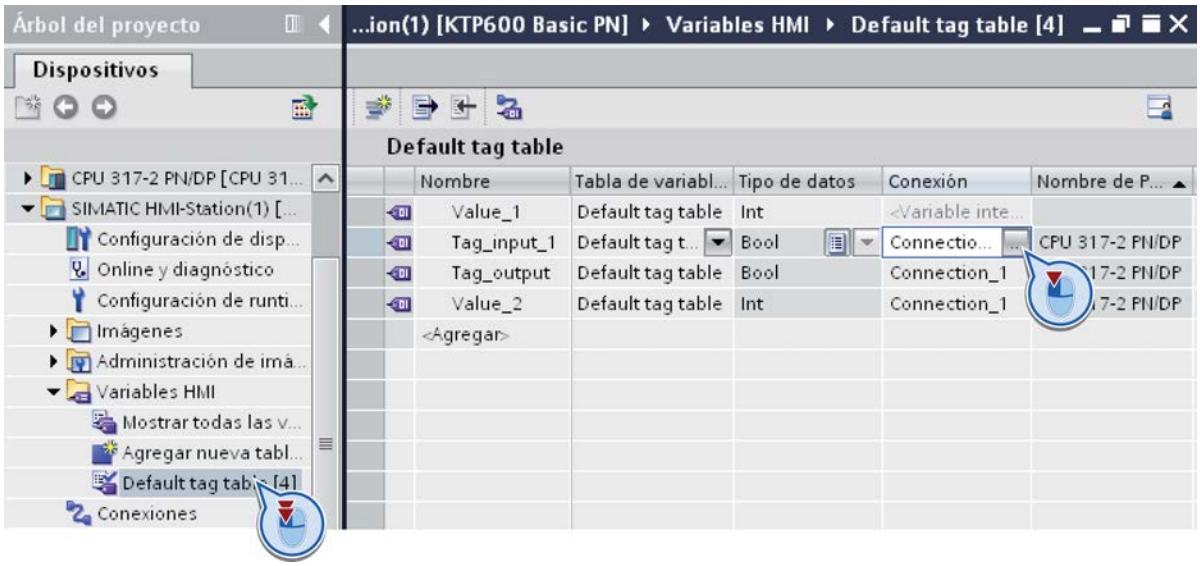
4.1.8.5 Volver a enlazar variables HMI

Una vez establecida una conexión HMI entre la CPU y el dispositivo HMI, es preciso asignar las variables HMI existentes a la nueva conexión. Realice los pasos siguientes para cada fila de la tabla de variables correspondiente.

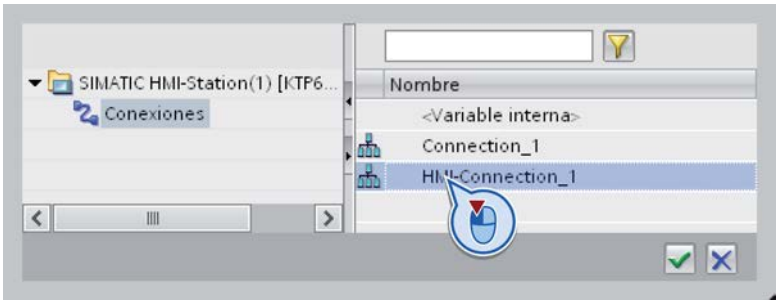
Procedimiento

Para enlazar nuevamente las variables HMI, proceda del siguiente modo:

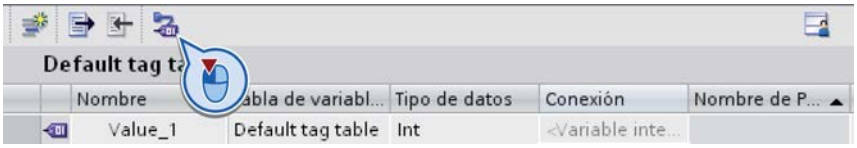
1. En el árbol del proyecto, desplácese a las variables HMI y haga doble clic en la tabla de variables correspondiente para visualizar las variables en el área de trabajo.
   Se abre la tabla de variables.



2. Haga clic en el botón "..." de la columna "Conexión".
   Se abre un cuadro de diálogo para seleccionar la conexión.
3. Seleccione la conexión HMI recién creada.



4. Haga clic en el botón "✓" para adoptar la conexión seleccionada.
5. En la barra de función, haga clic en el botón "Conectar nuevamente variable PLC".



Consulte también

Borrar una conexión no especificada (Página 184)

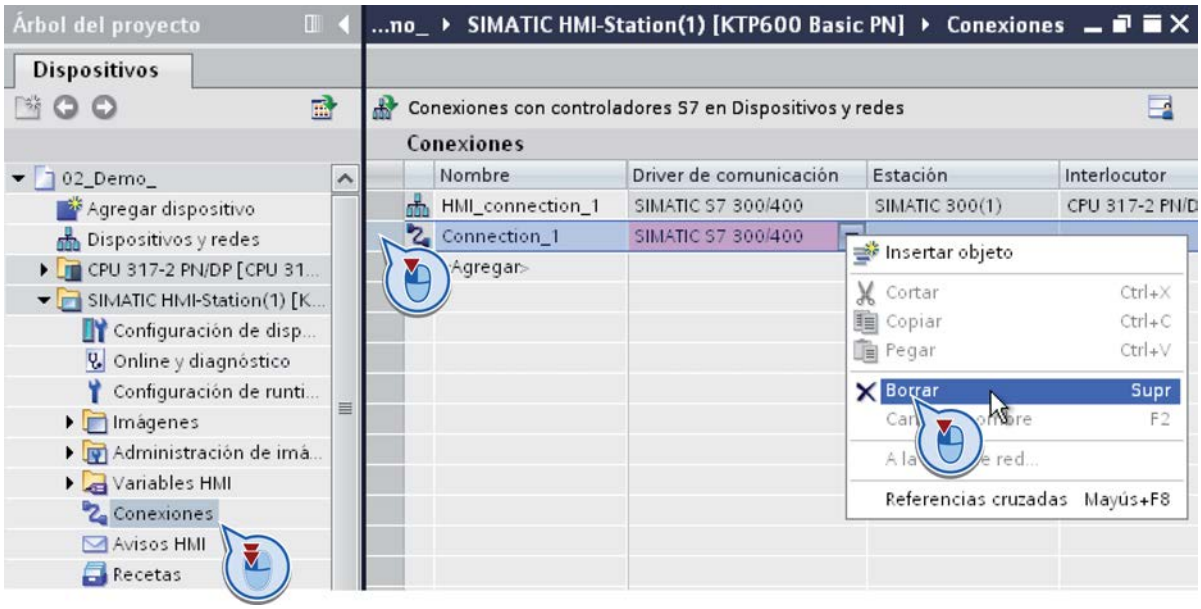
4.1.8.6 Borrar una conexión no especificada

Por último, es posible borrar las conexiones no especificadas que queden aún del proyecto de origen.

Procedimiento

Para borrar conexiones no especificadas, proceda del siguiente modo:

1. En el árbol del proyecto, abra el dispositivo HMI y haga doble clic en la entrada "Conexiones".
   Se abre la tabla de conexiones.



2. Seleccione la fila de la tabla con la conexión antigua.
3. En el menú contextual de la fila de conexión, elija el comando "Borrar".
4. Repita los pasos anteriores para todas las conexiones no especificadas del proyecto de origen.

4.2 Migración de programas PLC a una CPU S7-1500

4.2.1 Principios básicos de la migración de programas PLC

Introducción

La migración de programas PLC dentro del TIA Portal es posible desde una CPU de la serie S7-300/400 a una CPU de la serie S7-1500. La migración dentro del TIA Portal se denomina "Migración PLC". En esta migración se crea la CPU nueva en el proyecto y el programa PLC existente se copia en esta CPU nueva. La CPU antigua permanece invariable en el proyecto.

La ventaja entre migrar y copiar radica en que con la migración el programa se adapta automáticamente a la nueva familia de CPU dentro de las posibilidades. Las estructuras del programa que ya no se admiten se actualizan. Las instrucciones que no están disponibles en la CPU nueva se reemplazan por las instrucciones S7-1500 correspondientes.

Volumen de migración

En la migración PLC se copian todos los componentes del programa PLC en la CPU recién creada. Entre ellos se incluyen, por ejemplo, los siguientes objetos:

- Bloques de programa creados en un lenguaje soportado por S7-1500 y que no tienen protección de know-how.
- Tablas de variables PLC
- Tablas de observación y forzado permanente
- Tipos de datos PLC
- Objetos tecnológicos
- Grupos personalizados en el árbol del proyecto

La configuración hardware de la CPU de origen no se transfiere automáticamente a la CPU nueva de la serie S7-1500.

⚠ ADVERTENCIA

Evitar daños personales y materiales

En algunos casos se realizan cambios en el programa durante la migración PLC. Por este motivo es necesario, después de la migración, comprobar detalladamente el programa en un entorno de prueba antes de ponerlo en marcha.

Nota

Soporte adicional para la migración PLC

En Siemens Industry Online Support encontrará preguntas frecuentes actuales sobre el tema "Migración PLC" (http://support.automation.siemens.com/WW/view/es/64898867).

Si necesita soporte adicional para la migración PLC, diríjase al SIMATIC Customer Support.

Nota

Más información sobre el TIA Portal

Encontrará más información relacionada con el TIA Portal en TIA Portal en Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/es/65601780).

Allí también encontrará información sobre ofertas de formación. Para aquellos que estén familiarizados con el mundo SIMATIC S7 clásico y deseen cambiar a SIMATIC S7-1500 con el TIA Portal recomendamos el curso de formación "TIA-SYSUP - Curso de transición a SIMATIC S7-1500 para el sistema SIMATIC TIA Portal".

4.2.2 Ejecución de la migración

Requisitos

- Existe un programa de partida coherente.
- Se ha compilado una versión actual de todos los bloques. Esto también es válido para los bloques contenidos en el proyecto pero que no han sido llamados durante la ejecución del programa.
- El programa ha sido creado con TIA Portal V12 o superior, o bien actualizado a esta versión.

Preparar la migración

Para preparar el programa para la migración, proceda del siguiente modo:

1. En el árbol del proyecto, abra la CPU que contiene el programa de partida.
2. Abra la carpeta "Bloques de programa" y compruebe si contiene bloques con protección de know-how.
3. Elimine la protección de know-how de los bloques.

4. Abra la carpeta "Bloques de programa > Bloques de sistema" y compruebe si contiene bloques con la extensión "_LF" (legacy functions).
   Son bloques de la librería de STEP 7 que ya no son compatibles con el TIA Portal. En la migración PLC, estos bloques no se transfieren al S7-1500 porque tienen protección de know-how.
5. Si es posible, sustituya estos bloques por una instrucción de la Task Card "Instrucciones".

Procedimiento

Para migrar un programa PLC dentro del TIA Portal a S7-1500 proceda del siguiente modo:

1. En el árbol del proyecto, abra la CPU que contiene el programa de partida.
2. Abra la configuración de dispositivos.
3. En la vista de dispositivos o de redes marque la CPU que contiene el programa de partida.
4. Elija el comando "Migrar a S7-1500" del menú contextual.
   Se abre el cuadro de diálogo "Migrar a S7-1500".
5. Seleccione en el campo "Dispositivo nuevo" la CPU a la que desea migrar el programa y confirme con "Aceptar".
   Aparece una consigna de seguridad que le informa de que el programa se modificará durante la migración.
6. Confirme la consigna de seguridad.
   Aparece un cuadro de diálogo en el que puede indicar qué instrucciones sobre la comunicación punto a punto desea utilizar en el programa migrado.
7. Seleccione el juego de instrucciones que desee. Debe seleccionar una de las dos opciones aunque no utilice la comunicación punto a punto. En ese caso, la selección no tendrá ninguna repercusión en el programa.
   El programa PLC se migra. A continuación aparece un mensaje que informa de si se han producido errores durante la migración. El mensaje contiene asimismo un enlace al informe de migración.
8. Abra el informe de migración. Este contiene información detallada sobre la migración e informa sobre los cambios que deben realizarse en el programa para que este pueda ejecutarse en la nueva CPU.
9. Procese todas las indicaciones del informe de migración.
10. Compile el programa migrado.

Resultado

En el árbol del proyecto se ha creado una CPU nueva junto a la CPU original. Esta contiene el programa PLC migrado.

Para aprovechar toda la funcionalidad de las S7-1500, recomendamos activar el acceso optimizado a bloques para los bloques migrados. Encontrará más información al respecto en "Consulte también".

Nota

Tenga en cuenta que con la migración PLC solo se convierte el programa PLC. La configuración hardware de la CPU de origen no se transfiere automáticamente a una CPU de la serie S7-1500. Ejecute manualmente la configuración hardware de la CPU nueva después de la migración.

Consulte también

Uso de informes (Página 339)

Bloques con acceso optimizado (Página 2258)

4.2.3 Visualización de un archivo de registro

Para cada migración se crea un archivo de registro. El archivo de registro contiene la siguiente información:

- Modificaciones en objetos realizadas durante la migración.
- Indicaciones sobre las adaptaciones necesarias que se deben realizar en el programa.

Procedimiento

Para visualizar el archivo de registro de la migración, proceda del siguiente modo:

1. Abra la carpeta "Datos comunes > Informes" en el árbol del proyecto. Esta carpeta contiene los informes de todas las migraciones realizadas hasta el momento.
2. Haga doble clic en el informe de migración deseado.
   El informe se abre.
3. Los mensajes que hacen referencia a un punto determinado del programa están marcados con una flecha en la columna "Ir a". Haga doble clic en la flecha para saltar a la posición relevante del programa.
4. Los mensajes que poseen información más detallada están marcados con un signo de interrogación en la columna "?". Para obtener información más detallada sobre el mensaje, haga clic en el signo de interrogación.

Consulte también

Uso de informes (Página 339)

4.2.4 Particularidades en la migración de programas PLC

4.2.4.1 Indicaciones sobre la migración de programas PLC

Bloques de organización

En la migración de bloques de organización se deben tener en cuenta las reglas siguientes:

- El nombre y el número del bloque se adoptan sin cambios.
- La interfaz del bloque también se adopta sin cambios.
- Al OB se le asignará el evento adecuado al tipo de OB.
- Los parámetros del OB, p. ej. la prioridad, se adoptan sin cambios. Si en la CPU nueva hay otros parámetros asignables, estos tendrán los valores predeterminados.

Consulte también:

Principios básicos del tratamiento de errores (Página 2932)

Algunos bloques de organización (OB) no están soportados por el S7-1500 y por ello no pueden migrarse. La siguiente tabla proporciona una vista general de estos bloques de organización y contiene indicaciones acerca de cómo puede reproducir la funcionalidad en su programa:

OB no migrable	Indicaciones
OB60 (alarma de multiprocesamiento)	No se puede reemplazar el OB. La alarma de multiprocesamiento no está disponible en S7-1500.
OB65 (alarma de sincronismo tecnológica)	Compruebe si el OB del programa puede reemplazarse por el OB de servo MC o el OB de interpolador MC.
OB7x (error de redundancia)	No se puede reemplazar el OB. La redundancia no está disponible en S7-1500.
OB81 (fallo de la fuente de alimentación)	Compruebe si el OB del programa puede reemplazarse por el OB de alarma de diagnóstico.
OB84 (error de hardware en CPU)	Compruebe si el OB del programa puede reemplazarse por el OB de alarma de diagnóstico.
OB85 (error de ejecución del programa)	Compruebe si el OB del programa puede reemplazarse por el OB de extracción/inserción o por el OB de error de rack.
OB87 (error de comunicación)	Compruebe si el OB del programa puede reemplazarse por el OB de alarma de diagnóstico.
OB88 (interrupción del procesamiento)	Compruebe si el OB del programa puede reemplazarse por el OB de error de programación.
OB90 (ciclo de segundo plano)	No se puede reemplazar el OB. La ejecución de fondo no está disponible en S7-1500.
OB101 (rearranque)	No se puede reemplazar el OB. El rearranque no está disponible en S7-1500.
OB102 (arranque en frío)	Compruebe si el OB del programa puede reemplazarse por el OB de arranque.

Consulte también:

Auto-Hotspot

Auto-Hotspot

Palabra de estado

En el S7-1500, la palabra de estado no está soportada en su totalidad. Contiene menos información y solo puede evaluarse en AWL. La interpretación de la palabra de estado ya no está soportada en KOP y FUP. Los siguientes capítulos contienen indicaciones sobre la migración de accesos a la palabra de estado en los diferentes lenguajes de programación.

Consulte también:

Migración de programas KOP/FUP a S7-1500 (Página 218)

Migración de programas AWL a S7-1500 (Página 222)

Principios básicos de la palabra de estado (Página 2559)

Uso de parámetros de bloque

Para la utilización de los parámetros de bloque dentro del bloque rigen en S7-1500 las reglas siguientes:

- Los parámetros de entrada (Input) solo pueden leerse.
- Los parámetros de salida (Output) solo pueden escribirse.
- Los parámetros de entrada/salida (InOut) pueden leerse y escribirse.
- Los valores de función (Ret_Val) solo pueden escribirse.

Si el programa no cumple esta norma, se mostrará una advertencia durante la compilación. En ese caso convierta el parámetro de entrada o salida en cuestión en un parámetro de entrada/salida.

Para la asignación de parámetros a funciones (FC) rigen reglas adicionales.

Consulte también:

Asignación de parámetros a funciones (Página 2270)

Acceso a parámetros de bloque durante la ejecución del programa (Página 2273)

Parámetros de bloque como marcas de flanco en funciones (FC)

También al programar evaluaciones de flancos en funciones (FC) deben observarse las reglas anteriores sobre la transferencia de valores a parámetros:

Como marca de flanco se requiere un valor de datos que pueda leerse y escribirse y que se mantenga durante varios ciclos. Dado que los parámetros de entrada (Input) solo pueden leerse y los parámetros de salida (Output) solo pueden escribirse, no pueden utilizarse como marcas de flanco. Tampoco los datos locales temporales (Temp) pueden utilizarse como marcas de flanco, ya que solo están disponibles durante un ciclo.

Por ello, en las funciones (FC) solo pueden utilizarse parámetros de entrada/salida (InOut) como marcas de flanco. También es posible utilizar como marca de flanco un solo bit de un parámetro de entrada/salida, siempre que este se direccione a través de un acceso Slice.

El siguiente ejemplo muestra cómo utilizar un parámetro de entrada/salilda como marca de flanco.

AWL

```
FP #InOutFlagStore.x0
```

"InOutFlagStore" es un parámetro de entrada/salida del tipo de datos BYTE. La dirección "#InOutFlagStore.x0" direcciona la dirección de bit 0, que ejerce de marca de flanco.

Inicialización automática de parámetros de bloque en funciones (FC)

En el S7-300/400 es imprescindible preasignar un valor a los datos locales temporales (Temp) de una función. De lo contrario el programa podría funcionar con valores no definidos.

En el S7-1500 el peligro de procesar valores no definidos se reduce notablemente, ya que los siguientes parámetros se inicializan automáticamente en la llamada del bloque:

- A los datos temporales del tipo de datos STRING se les preasigna siempre información correcta sobre la longitud (máxima y real).
- En las funciones (FC), los datos temporales con tipo de datos simple se inicializan automáticamente con acceso optimizado. Se les asigna el valor predefinido para el tipo de datos especificado. Por ejemplo, el valor predefinido para BOOL es "false". A los elementos de los tipos de datos PLC se les preasigna el valor predeterminado especificado en la declaración del tipo de datos PLC (UDT). A los elementos ARRAY se les preasigna el valor "0", aunque se utilicen en un tipo de datos PLC.

En el S7-300/400, debía escribirse forzosamente un valor en los parámetros de salida (Output) de funciones, ya que de lo contrario podían producirse reacciones no deseadas en el programa. Cuando se utilizaban instrucciones de salto o instrucciones que dependían del RLO, era necesario comprobar esto en todos los circuitos posibles del programa.

En el S7-1500, el peligro que suponen los parámetros de salida no definidos se reduce notablemente, ya que los parámetros de salida del tipo de datos simple se inicializan automáticamente en las llamadas de bloque. A estos parámetros se les asigna previamente el valor predefinido para el tipo de datos especificado. Por ejemplo, el valor predefinido para BOOL es "false". No obstante, a los parámetros de salida estructurados no se les asigna ningún valor. En la llamada de bloque se transmiten como puntero, por lo que no pueden ser no definidos.

Consulte también:

Asignar valores a parámetros de funciones (Página 2270)

Bloques de datos de CPU

Los bloques creados por las instrucciones CREAT_DB o CREATE_DB en la CPU y que solo existen online no se migran.

Consulte también:

Migración de instrucciones de bloques de datos (Página 203)

Bloques de datos de CPU (Página 2256)

Bloques multiinstancia

En las CPU de la serie S7-1500 todos los bloques de función son aptos para multiinstancia. Por ello los FBs que estaban marcados en el programa de partida como no aptos para multiinstancia se convierten en bloques aptos para multiinstancia con la migración. Para cada FB convertido se incluye una indicación en el informe de migración. Si después de la migración se continúa utilizando el FB como una instancia individual, el programa funcionará sin cambios.

Consulte también:

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Bloques de librerías STEP 7

Algunos bloques de librerías STEP 7 ya no se soportan en el TIA Portal. Cuando no era posible la migración de uno de estos bloques al TIA Portal, estos bloques se adoptaban como bloques de usuario con protección de know-how en el TIA Portal y se proveían con la extensión "_LF" (legacy function). En la migración PLC dentro del TIA Portal estos bloques no se adoptan en S7-1500 ya que poseen una protección de know-how.

Si es posible, compruebe antes de la migración si los bloques con la extensión "_LF" pueden reemplazarse por una instrucción de la Task Card "Instrucciones".

Números en coma flotante no válidos

Existe una diferencia en el tratamiento de números en coma flotante no válidos en S7-1500 y S7-300/400. Ejemplo:

- S7-300/400: el resultado de la expresión "Número en coma flotante no válido <> 1.0" es FALSE.
- S7-1500: el resultado de la expresión "Número en coma flotante no válido <> 1.0" es TRUE.

Observe que las instrucciones en el programa pueden suministrar otros resultados debido a esta diferencia.

Consulte también:

Números en coma flotante (Página 2338)

Accesos absolutos a datos locales

En el S7-300/400 se podía acceder de forma absoluta a los datos locales de un bloque sin declarar un símbolo para la dirección.

Como el acceso absoluto en S7-1500 ya no se soporta, la migración lo procesa del siguiente modo:

- Si para un área dentro de los datos locales se declara una variable temporal en la interfaz del bloque, la dirección absoluta se convertirá en una dirección simbólica.
- Si no existe una variable temporal adecuada, la dirección absoluta se mantendrá. Sin embargo, dado que durante la migración pueden crearse otras variables locales, es posible que las direcciones se desplacen dentro de los datos locales. Por ello hay que comprobar si la dirección absoluta utilizada aún es correcta y si la fecha correcta está direccionada. Si es posible debe crearse una variable en el apartado "Temp" y utilizarse para el direccionamiento. Para direccionar áreas selectivamente dentro de variables declaradas, se pueden superponer variables existentes con AT o direccionar áreas individuales con la sintaxis .X, .B, .W o .D.

Consulte también:

Superponer variables con AT (Página 2302)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Los siguientes ejemplos muestran la migración de accesos absolutos en datos locales.

En la primera tabla se observa un segmento del programa antes de la migración:

AWL

```
L %LW20
L %LW3
+I
T %LW5
```

En la segunda tabla se observa el segmento del programa después de la migración:

AWL

```
L "MyTempVar1"
L "MyTempVar2"
+I
T %LW5
```

- Para "LW20" y "LW3" ya se habían declarado variables temporales. Estas se utilizarán en el programa después de la migración.
- Para "LW5" no se había declarado ninguna variable. Después de la migración se debe comprobar si el acceso aún es correcto.

Carga de modificaciones de software

Al cargar modificaciones de software, las CPU de la serie S7-1500 se comportan de forma distinta a las CPU de las series S7-300/400. Por ejemplo, ofrecen la posibilidad de cargar modificaciones en STOP y en RUN sin influir en los valores actuales de variables ya cargadas.

Consulte también:

Auto-Hotspot

Listas de estado del sistema

Las CPU de la serie S7-1500 no proporcionan listas de estado del sistema. No obstante, existe la posibilidad de leer algunas informaciones a través de las instrucciones "GET_DIAG", "Geo2Log", "DeviceStates" o "ModuleStates".
Encontrará más información sobre la migración de las listas de estado del sistema en: Migración de la instrucción RDSYSST (Página 205)

Diagnóstico del sistema con "Report System Errors"

Las CPU de la serie S7-1500 disponen de un diagnóstico de sistema integrado. Por este motivo, "Report System Errors" ya no se soporta. Si el programa contiene bloques para "Report System Errors", desactive el diagnóstico de sistema en el programa de partida antes de la migración y vuelva a compilar el programa. Al hacerlo se borrarán los bloques correspondientes.

El diagnóstico de sistema de las CPU S7-1500 está activado por defecto. No es necesario que realice otros ajustes aquí.
Para modificar los ajustes del diagnóstico de sistema, seleccione la CPU en la configuración de dispositivos. A continuación, en la ventana de inspección se muestran los ajustes del diagnóstico de sistema en "Propiedades > Diagnóstico de sistema". Los avisos del diagnóstico de sistema están activados y puede anularse la selección si no se necesitan.

Diagnóstico PROFINET/PROFIBUS

En la migración del diagnóstico PROFINET / PROFIBUS se aplican las siguientes reglas:

- "Notificar errores de sistema" se sustituye por el diagnóstico de sistema integrado.
- El FB 126 no se soporta y, por tanto, no puede migrarse.
- Para la evaluación aplicada pueden utilizarse las instrucciones "DeviceStates" y "ModulStates".

Nota

Soporte adicional

En Siemens Industry Online Support encontrará preguntas frecuentes actuales sobre el tema "Migración PLC" (http://support.automation.siemens.com/WW/view/es/64898867).

Si necesita soporte adicional para la migración PLC, diríjase al SIMATIC Customer Support.

Consulte también

Transferencia de parámetros en la llamada de un bloque (Página 2268)

4.2.4.2 Migración de instrucciones a S7-1500

Indicaciones sobre la migración de instrucciones

Migración automática de instrucciones

Durante la migración se adoptan en el nuevo programa las instrucciones utilizadas en el programa de partida. Las adaptaciones necesarias se llevan a cabo de forma automática en lo posible, sin cambiar la semántica del programa de partida. Si existen varias versiones de una instrucción, la migración utilizará siempre la última versión.

Las instrucciones que no están disponibles en S7-1500 se reemplazan, en lo posible, por una instrucción compatible o similar automáticamente.

Escenarios de migración

La tabla siguiente muestra los posibles escenarios de migración para las instrucciones:

Clase	Situación	Reacción del sistema	Adaptación manual del programa
1	La instrucción en las dos familias de CPU es idéntica.	La instrucción se migra.	No es necesaria una adaptación. El programa migrado no tiene cambios semánticos y se puede compilar.
2	En S7-1500 ya no existe la instrucción, pero existe una instrucción compatible más reciente.	La instrucción será reemplazada por la nueva. Es posible que se añadan más instrucciones que sean necesarias para mantener la semántica del programa de partida. El informe de migración notifica el reemplazo.	No es necesaria una adaptación. El programa migrado no tiene cambios semánticos y se puede compilar.

Clase	Situación	Reacción del sistema	Adaptación manual del programa
3	En S7-1500 ya no existe la instrucción, pero existe una instrucción similar.	La instrucción se reemplaza por la instrucción similar. El informe de migración notifica el reemplazo. Las posiciones que deben comprobarse están marcadas en el programa. Obtendrá indicaciones acerca de las adaptaciones necesarias como comentarios directamente en la parte del programa correspondiente o al compilar.	Debe comprobar y, dado el caso, adaptar el programa.
4	En S7-1500 ya no existe la instrucción y tampoco existe una instrucción similar.	La instrucción se representa en color rojo. No es posible compilar el programa.	Debe comprobar y, dado el caso, adaptar el programa.

Indicaciones sobre instrucciones de la clase de migración 3

Algunas instrucciones que estaban disponibles en S7-300/400 ya no se ofrecen en S7-1500, ya que en este caso hay métodos más sencillos o con mayor rendimiento para la solución de determinadas tareas. La migración PLC le ayuda en la migración al ser reemplazadas esas instrucciones por instrucciones nuevas adecuadas. Sin embargo, las instrucciones del caso 3 no son completamente migrables de forma automática, ya que se han añadido, por ejemplo, parámetros formales nuevos. Las posiciones que no son migrables automáticamente se marcan en el programa. Obtendrá indicaciones acerca de las adaptaciones necesarias como comentarios directamente en el lugar correspondiente del programa, en el informe de migración o al compilar.

Pueden ser necesarias las siguientes adaptaciones manuales:

- Si la nueva instrucción tiene parámetros formales adicionales, asigne parámetros actuales adecuados a esos parámetros.
- Si la nueva instrucción tiene nombres de parámetros modificados, cambie la asignación de parámetros.
- Si la migración no puede garantizar de forma unívoca que la transferencia de parámetros sea semánticamente idéntica con el programa de partida, los parámetros actuales que deben comprobarse se marcan. Compruebe la semántica de estas partes del programa después de la migración y corríjalas si es necesario.
- Si el programa original contiene identificadores HW, p. ej. en el parámetro "LADDR", estos también se marcan. Introduzca el nuevo identificador HW en el parámetro "LADDR".
- Si la nueva instrucción tiene códigos de error modificados, adapte el tratamiento de errores en el programa.

Encontrará algunos ejemplos de instrucciones de la clase de migración 3 en el capítulo "Migración de instrucciones de bloques de datos (Página 203)".

Indicaciones sobre instrucciones de la clase de migración 4

Algunas instrucciones ya no se ofrecen en S7-1500, ya que la funcionalidad ha cambiado desde el principio. Esas instrucciones no se reemplazan por la migración. Se representan en rojo en el programa y hacen que el programa no se pueda compilar. Obtendrá indicaciones acerca de las adaptaciones necesarias como comentarios directamente en el lugar correspondiente del programa, en el informe de migración o al compilar.

En los siguientes capítulos encontrará una relación de las instrucciones de la clase de migración 4 que no pueden migrarse e información sobre soluciones alternativas.

Nota

Soporte adicional

En Siemens Industry Online Support encontrará preguntas frecuentes actuales sobre el tema "Migración PLC" (http://support.automation.siemens.com/WW/view/es/64898867).

Si necesita soporte adicional para la migración PLC, diríjase al SIMATIC Customer Support.

Vista general de las instrucciones no migrables

Introducción

Algunas instrucciones de S7-300/400 no pueden migrarse a S7-1500. Esto ocurre, por ejemplo, cuando las instrucciones se utilizan solo en relación con módulos determinados o cuando la funcionalidad de la instrucción en S7-1500 se debe implementar de otra forma.

Indicaciones sobre el reemplazo de instrucciones no migrables (clase 4)

A continuación encontrará ejemplos de instrucciones no migrables e indicaciones sobre su reemplazo:

Grupo de instrucciones	Instrucción no migrable	Indicaciones
Otras instrucciones	SETP	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "SET_BF".
	RESETP	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RESET_BF".
	DRUM_X	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "DRUM".
	TONR_X	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "TONR".
	RESET	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RESET_BF".
	SET	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "SET_BF".

Grupo de instrucciones	Instrucción no migrable	Indicaciones
	WSR	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	SHRB	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	RESETI	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RESET_BF".
	SETI	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "SET_BF".
Control del tiempo de ejecución	COMPRESS	Borre la instrucción. En S7-1500 ya no es necesario comprimir la memoria de trabajo de la CPU o la memoria de carga.
	PROTECT	Configure el nivel de protección y la contraseña a través del panel frontal de la CPU S7-1500 o en las propiedades de módulo de la CPU.
	CiR	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
Fecha y hora	SET_CLKS	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "WR_SYS_T".
	LOC_TIME	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RD_SYS_T".
	BT_LT	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RD_SYS_T".
	LT_BT	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RD_SYS_T".
	S_LTINT	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "SET_TINTL".
	SET_SW	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "WR_SYS_T" o "RD_SYS_T".
	SET_SW_S	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "WR_SYS_T" o "RD_SYS_T".
	TIMESTMP	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "WR_SYS_T" o "RD_SYS_T".
	WS_RULES	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
Periferia descentralizada	D_PRAL	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	ASi_3422	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	SALRM	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
Parametrización del módulo	PARM_MOD	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
Alarmas	MP_ALM	La alarma de multiprocesamiento no está disponible en S7-1500.
	REPL_VAL	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.

Grupo de instrucciones	Instrucción no migrable	Indicaciones
Avisos	EN_MSG	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	DIS_MSG	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	ALARM_SQ	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	ALARM_S	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	ALARM_SC	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	WR_USMSG	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	READ_SI	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	DEL_SI	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	ALARM_DQ	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	ALARM_D	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	NOTIFY_8P	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	ALARM	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	ALARM_8	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	ALARM_8P	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	NOTIFY	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
	AR_SEND	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "Program_Alarm".
Diagnóstico	RDSYSST	La instrucción "RDSYSST: Leer lista de estado del sistema" ya no está disponible en S7-1500, ya que las CPU de dicha familia no elaboran listas de estado de sistema. No obstante, existe la posibilidad de leer algunas informaciones a través de las instrucciones "GET_DIAG", "Geo2Log", "DeviceStates" o "ModuleStates". Encontrará información adicional en el capítulo "Migración de la instrucción RDSYSST (Página 205)" y a través del SIMATIC Customer Support.
	OB_RT	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "RD_SYS_T".
Funciones de tabla	ATT	
	FIFO	
	TBL_FIND	
	LIFO	
	TBL	

Grupo de instrucciones	Instrucción no migrable	Indicaciones
	TBL_WRD	Las instrucciones no pueden reemplazarse. Si las instrucciones son necesarias en el programa, hay que implementarlas manualmente. Encontrará ayuda adicional en el capítulo "Migración de instrucciones de tabla (Página 208)" y a través del SIMATIC Customer Support.
	WRD_TBL	
	DEV	
	CDT	
	TBL_TBL	
	PACK	
iSlave	SET_ADDR	La instrucción no puede reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
PID Control	TUN_EC	Compruebe si las instrucciones del programa pueden reemplazarse por los objetos tecnológicos "PID_Compact" o "PID_3Step".
	TUN_ES	
Módulos de función	CNT_CTRL	Compruebe si las instrucciones del programa pueden reemplazarse por el objeto tecnológico "High_Speed_Counter" en combinación con los módulos de contador "S7-1500 TM Count 2x24V" o "TM PosInput 2". Como alternativa, compruebe si la función deseada puede realizarse con los módulos "ET 200S 1Count24V" o "1Count5V".
	DIAG_INF	
	CNT_CTL1	
	CNT_CTL2	
	CNT2WRPN	
	CNT2RDPN	
	CNT2_CTR	
	DIAG_RD	
	ABS_INIT	Compruebe si las funciones pueden ser relevadas por "S7-1500 Motion" o "ET 200S 1PosU".
	ABS_CTRL	
	ABS_DIAG	
	CAM_INIT	Compruebe si pueden leerse señales de encóder con módulos "S7-1500 TM Count 2x24V" o "TM PosInput 2" y utilizarse ambas DQ para reacciones rápidas. Dependiendo del valor del encóder se pueden conectar otras DQ de DQ estándar del programa de usuario.
	CAM_CTRL	
	CAM_DIAG	
	PID_FM	Compruebe si las instrucciones del programa pueden reemplazarse por los objetos tecnológicos "PID_Compact" o "PID_3Step" o funciones PID básicas.
	FUZ_355	
	FORCE355	
	READ_355	
	CH_DIAG	
	PID_PAR	
	CJ_T_PAR	
	FMT_PID	

Grupo de instrucciones	Instrucción no migrable	Indicaciones
	FMT_PAR	
	FMT_CJ_T	
	FMT_DS1	
	FMT_TUN	
	FMT_PV	
Funciones 300C	ANALOG_300C	Compruebe si las instrucciones del programa pueden reemplazarse por las instrucciones del grupo "Motion Control".
	DIGITAL_300C	
	COUNT_300C	Compruebe si las instrucciones del programa pueden reemplazarse por el objeto tecnológico "High_Speed_Counter" en combinación con módulos de contador.
	FREQUENC_300C	
	PULSE_300C	Compruebe si la instrucción del programa puede reemplazarse por la instrucción "CTRL_PWM".
	SEND_RK_300C	Las instrucciones de comunicación del S7-1500 no soportan RK 512. Si se requiere la funcionalidad RK, hay que implementarla manualmente.
	FETCH_RK_300C	
	SERVE_RK_300C	
Comunicación con iSlave/ iDevice	I_GET	Las instrucciones no pueden reemplazarse. Si necesita mayor asistencia, diríjase al SIMATIC Customer Support.
	I_PUT	
	I_ABORT	
Comunicación S7	PRINT	Compruebe si las instrucciones del programa pueden reemplazarse por las instrucciones de la comunicación S7 para S7-1500. Las instrucciones "CONTROL" y "C_CNTRL" podrían sustituirse, p. ej., por la instrucción "T_DIAG".
	START	
	STOP	
	RESUME	
	STATUS	
	USTATUS	
	CONTROL	
	C_CNTRL	
Open User Communication	TCON_PAR	Estas instrucciones no son necesarias para la Open User Communication en S7-1500. Los parámetros de conexión se ajustan en las propiedades del bloque.
	TADDR_PAR	
	TCP_conn_active	
	TCP_conn_passive	
	ISOonTCP_conn_active	
	SOonTCP_conn_passive	
	ISOonTCP_conn_CP_active	
	ISOonTCP_conn_CP_passive	
	UDP_local_open	

Grupo de instrucciones	Instrucción no migrable	Indicaciones
	UDP_rem_address_and_port	
Procesador de comunicaciones	Instrucciones del grupo CP SIMATIC NET Instrucciones de los grupos PTP CP 340, PTP CP341, PTP CP 440, PTP CP 441	Encontrará información detallada sobre estas instrucciones en el capítulo "Migración de instrucciones de comunicación (Página 211)".
PROFINET/CBA	PN_IN	S7-1500 no soporta la comunicación PROFINET/CBA. Utilice la comunicación S7 o la Open User Communication.
	PN_OUT	
	PN_DP	
Comunicación MPI	X_SEND	S7-1500 no soporta la comunicación MPI. Utilice la comunicación S7 o la Open User Communication.
	X_RCV	
	X_GET	
	X_PUT	
	X_ABORT	
Comunicación de datos globales	GD_SND	S7-1500 no soporta la comunicación de datos globales. Utilice la comunicación S7 o la Open User Communication.
	GD_RCV	
Comunicación PtP	P_PRINT	Los módulos de comunicaciones PtP no soportan la función de impresión.
	P_PRINT341	Los módulos de comunicaciones PtP no soportan la función de impresión.
	MODB_341	Estos bloques para CP 341 / CP 441 no están soportados por el S7-1500. Utilice en su lugar las instrucciones para MODBUS (RTU). • Modbus_Comm_Load • Modbus_Master • Modbus_Slave
	MODB_441	
	S_MODB	Estos blouqes para ET 200S 1SI no están soportados por el S7-1500.
	S_USST	
	S_USSR	
	S_USSI	
TeleService	PG_DIAL	S7-1500 no soporta la interfaz MPI como interfaz estándar. La instrucción ya no está disponible.
	AS_DIAL	S7-1500 no soporta la interfaz MPI como interfaz estándar. La instrucción ya no está disponible.
	SMS_SEND	S7-1500 no soporta la interfaz MPI como interfaz estándar. Compruebe si la instrucción del programa puede reemplazarse por la instrucción "TMAIL_C".

Grupo de instrucciones	Instrucción no migrable	Indicaciones
	AS_MAIL	S7-1500 no soporta la interfaz MPI como interfaz estándar. Compruebe si la instrucción del programa puede reemplazarse por la instrucción "TMAIL_C".
Librería de usuario	PNIO_DIAG	La instrucción "PNIO_DIAG" se pudo cargar para S7-300/400 como descarga en una librería de usuario y utilizar a efectos de diagnóstico. "PNIO_DIAG" ha sido sustituido en S7-1500 por el diagnóstico integrado y, por tanto, ya no es necesario. En S7-1500 el diagnóstico de sistema está activado por defecto. Puede usar el diagnóstico de sistema sin necesidad de realizar más ajustes. En caso necesario, puede editar los ajustes para el diagnóstico de sistema en la configuración de dispositivos de S7-1500 en el registro "Propiedades > Diagnóstico". Ahí también puede configurar las alarmas del diagnóstico de sistema. Las alarmas están activadas por defecto. En el panel HMI puede borrar, después de la migración, las imágenes de diagnóstico utilizadas en relación con "PNIO_DIAG" y las etiquetas correspondientes. En su lugar, utilice el control "Visor de diagnóstico del sistema" en sus imágenes de diagnóstico.

Nota

Soporte para las instrucciones no migrables

Si necesita soporte adicional con las instrucciones no migrables diríjase al SIMATIC Customer Support.

En Siemens Industry Online Support encontrará preguntas frecuentes actuales sobre el tema "Migración PLC" (http://support.automation.siemens.com/WW/view/es/64898867).

Migración de instrucciones de bloques de datos

Procedimiento en la migración de instrucciones de bloques de datos (clase de migración 3)

S7-1500 ofrece un juego de instrucciones modificado de la categoría "Bloques de datos". Esas instrucciones se reemplazan automáticamente durante la migración. A continuación, deberá realizar adaptaciones manuales.

La tabla siguiente muestra una vista general de la migración de las instrucciones de bloques de datos:

Instrucción en S7-300/400	Instrucción en S7-1500	Clase de migración
CREA_DBL	CREATE_DB	3
CREAT_DB	CREATE_DB	3
CREA_DB	CREATE_DB	3

Instrucción en S7-300/400	Instrucción en S7-1500	Clase de migración
DEL_DB	DELETE_DB	3
TEST_DB	ATTR_DB	3

ATENCIÓN
Ejecución asíncrona En S7-1500 las nuevas instrucciones trabajan de forma asíncrona, es decir, la ejecución se extiende a varios ciclos de programa.

Adaptaciones manuales tras la migración

En S7-1500 las instrucciones tienen parámetros formales adicionales que son necesarios para la ejecución asíncrona, por ejemplo. Estos parámetros adicionales deben abastecerse con parámetros actuales adecuados después de la migración.

El número de bloque del DB creado en S7-1500 debe estar comprendido entre 60000 y 60999. Por ello, adapte los parámetros "LOW_LIMIT" y "UP_LIMIT" al nuevo rango numérico.

Encontrará una descripción detallada de los parámetros en la Ayuda en pantalla de S7-1500.

Consulte también:

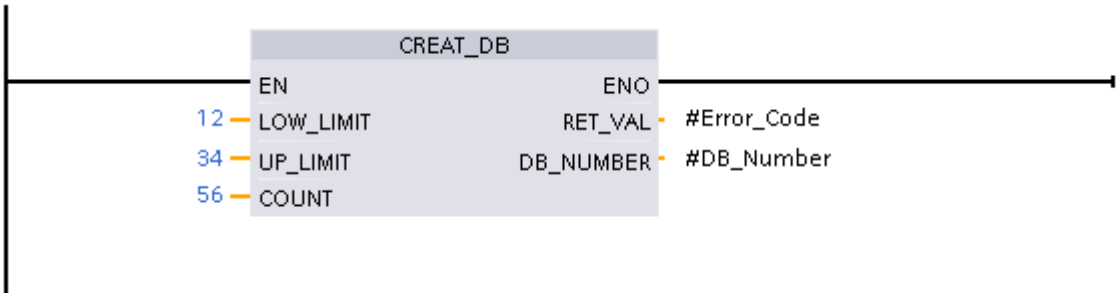
Funciones de bloques de datos (Página 4994)

Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277)

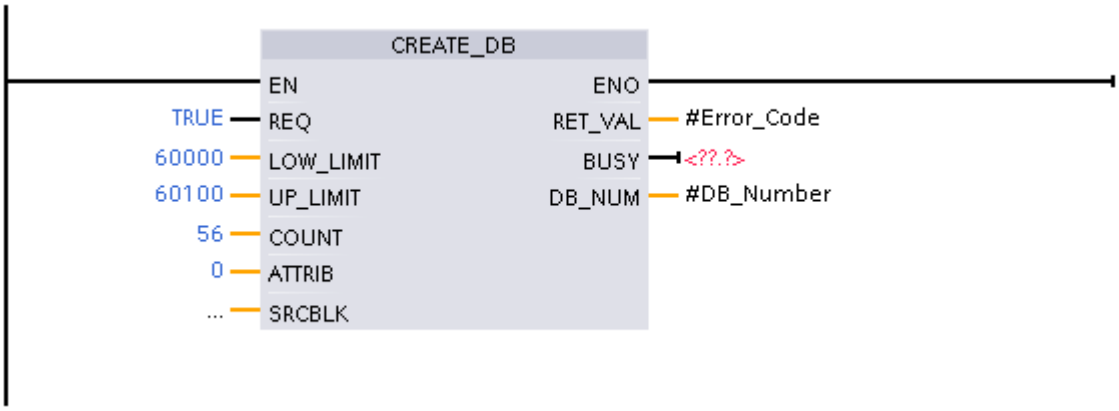
Ejemplo

El ejemplo siguiente muestra la migración de la instrucción "CREAT_DB" a la instrucción "CREATE_DB".

En la primera figura se indica la llamada de "CREAT_DB" antes de la migración:



En la segunda figura se indica la llamada de "CREATE_DB" tras la migración:



- El parámetro "REQ" no existía en "CREAT_DB" y se ha agregado con la migración. Recibe el valor predeterminado "TRUE".
- Los parámetros "LOW_LIMIT" y "UP_LIMIT" estaban fuera del rango numérico permitido de entre 60000 y 60999. Por ello, los valores "12" y "34" se han marcado como no válidos durante la migración. A continuación se han introducido manualmente los valores permitidos "60000" y "60100".
- El parámetro "ATTRIB" no existía en "CREAT_DB" y se ha agregado con la migración. Recibe el valor predeterminado "0". En este parámetro se especifican las propiedades del DB que se va a crear.
- El parámetro "SRCBLK" no existía en "CREAT_DB" y se ha agregado con la migración. Aquí es posible indicar el área de datos con cuyo contenido se describe el DB que se va a crear.
- El parámetro "BUSY" no existía en "CREAT_DB" y se ha agregado con la migración. El parámetro conduce la señal "1" hasta que el proceso de creación del bloque de datos haya concluido. Para evaluar la señal es necesario conectar el parámetro de salida con un operando.

Migración de la instrucción RDSYSST

Procedimiento en la migración de RDSYSST (clase de migración 4)

La instrucción "RDSYSST: Leer lista de estado del sistema" ya no está disponible en S7-1500, ya que las CPUs de dicha familia no elaboran listas de estado de sistema. No obstante, existe la posibilidad de leer algunas de las informaciones que usted ha obtenido hasta el momento a través de la lista de estado del sistema mediante una de las siguientes instrucciones:

- "RDREC (Página 4746)" (Instrucciones avanzadas > Periferia descentralizada)
- "GET_DIAG (Página 4956)" (Instrucciones avanzadas > Diagnóstico)
- "GEO2LOG (Página 5004)" (Instrucciones avanzadas > Diagnóstico)
- "DeviceStates (Página 4948)" (Instrucciones avanzadas > Diagnóstico)
- "ModuleStates (Página 4951)" (Instrucciones avanzadas > Diagnóstico)

- "RALRM (Página 4753)" (Instrucciones avanzadas > Periferia descentralizada)
- "LED (Página 4935)" (Instrucciones avanzadas > Diagnóstico)

Leer información de estado de sistema en S7-1500

La tabla siguiente muestra qué informaciones pueden leerse a través del estado de sistema en S7-1500.

En la columna "SZL-ID" encontrará las ID que ha leído hasta el momento con la instrucción RDSYSST. Las dos cifras finales son relevantes El comodín "xy" varía en función del módulo empleado.

La columna "Migración" indica si la información todavía está disponible en S7-1500 y cómo puede usted leerla.

Lista parcial	SZL-ID	Migración
ID de módulo	W#16#xy11	Esta información puede leerse con la instrucción "RDREC". Lea el registro 16#AFF0 en el parámetro "INDEX".
Características de la CPU	W#16#xy12	En S7-1500 esta información no está disponible actualmente.
Áreas de memoria del usuario	W#16#xy13	En S7-1500 esta información no está disponible actualmente.
Áreas del sistema	W#16#xy14	En S7-1500 esta información no está disponible actualmente.
Tipos de bloque	W#16#xy15	En S7-1500 esta información no está disponible actualmente.
Identificación de un componente	W#16#xy1C	Para módulos centrales y PROFINET IO esta información puede leerse con la instrucción "RDREC". Lea el registro 16#AFF0 en el parámetro "INDEX".
Estado de alarma	W#16#xy22	En S7-1500 esta información no está disponible.
Correspondencia entre memorias imagen parcial de proceso y OBs	W#16#xy25	En S7-1500 esta información no está disponible.
Comunicación: datos de estado	W#16#xy32	En S7-1500 esta información no está disponible.
Información de grupo CPU H	W#16#xy71	Irrelevante
Estado de los LEDs de módulo	W#16#xy74	Esta información puede leerse con la instrucción "LED". Lea el registro W#16#xy74 en el parámetro "LED".
Esclavos DP conectados en el sistema H	W#16#xy75	Irrelevante
Información del sistema maestro DP	W#16#xy90	En S7-1500 esta información no está disponible.
Información de estado del módulo	W#16#xy91	Esta información puede leerse con las instrucciones "GET_DIAG" o "ModuleStates".

Lista parcial	SZL-ID	Migración
Información de estado del rack/de la estación	W#16#xy92	Esta información puede leerse con las instrucciones "GET_DIAG" o "DeviceStates".
Información de estado del rack/de la estación	W#16#xy94	Esta información puede leerse con las instrucciones "GET_DIAG" o "DeviceStates".
Información avanzada del sistema maestro DP	W#16#xy95	En S7-1500 esta información no está disponible.
Información de estado del módulo con PROFINET IO y PROFIBUS DP	W#16#xy96	Esta información puede leerse con las instrucciones "GET_DIAG", "ModuleStates" o "DeviceStates".
Información del cambiador de herramientas (PROFINET IO)	W#16#xy9C	En S7-1500 esta información no está disponible.
Búfer de diagnóstico	W#16#xyA0	En S7-1500 esta información no está disponible.
Información de diagnóstico del módulo (DS 0)	W#16#00B1	Esta información puede leerse con las instrucciones "GET_DIAG" o "RDREC". Dentro del OB de alarma obtendrá información de diagnóstico también a través de "RALRM".
Información de diagnóstico del módulo (DS 1), dirección física	W#16#00B2	Esta información puede leerse con las instrucciones "RDREC" o "GEO2LOG". Dentro del OB de alarma obtendrá información de diagnóstico también a través de "RALRM".
Información de diagnóstico del módulo (DS 1), dirección lógica	W#16#00B3	Esta información puede leerse con las instrucciones "GET_DIAG" o "RDREC". Dentro del OB de alarma obtendrá información de diagnóstico también a través de "RALRM".
Datos de diagnóstico de un esclavo DP	W#16#00B4	Esta información puede leerse parcialmente con las instrucciones "ModuleStates" o "GET_DIAG". Dentro del OB de alarma obtendrá información de diagnóstico también a través de "RALRM". Algunas de las informaciones no están disponibles en S7-1500.

Nota

Soporte para la información que no puede leerse

Si necesita soporte adicional para la información que no puede leerse, diríjase al SIMATIC Customer Support.

En Siemens Industry Online Support encontrará preguntas frecuentes actuales sobre el tema "Migración PLC" (http://support.automation.siemens.com/WW/view/es/64898867).

Migración de instrucciones de tabla

Ejemplo de migración de "ATT: Introducir valor en tabla"

Procedimiento en la migración de instrucciones de tabla (clase de migración 4)

Las instrucciones del grupo "Instrucciones de tabla" ya no están disponibles en S7-1500 y no pueden sustituirse por la migración, ya que se basan en un direccionamiento absoluto. No obstante, en S7-1500 hay disponibles cómodas funciones que permiten implementar las instrucciones manualmente. A continuación se describe un ejemplo para la implementación de la instrucción "ATT: Introducir valor en tabla".

Implementación

El siguiente programa SCL muestra la implementación de la instrucción "ATT". La función del ejemplo se ha implementado como FC en SCL.

La tabla siguiente muestra la declaración de las variables utilizadas:

Operando		Declaración	Tipo de datos	Descripción
val		Input	INT	Valor introducido
tab		InOut	Tipo de datos PLC "Tabla10"	Tabla con 10 filas
	x	InOut	ARRAY [0..9] of INT	Cada elemento ARRAY representa una fila. En el ejemplo se utiliza un ARRAY of INT. También es posible crear ARRAYs con cualquier tipo de datos. Igualmente pueden emplearse ARRAY of STRUCT o un ARRAY de un tipo de datos PLC.
	lng	InOut	INT	Longitud del búfer. El valor se rige por el número de elementos ARRAY de "tab.x".
	first	InOut	INT	Primer valor escrito
	last	InOut	INT	Último valor escrito
h		Temp	INT	Variable auxiliar
Ret_Val			BOOL	Valor de función

La tabla siguiente muestra la FC "Attend":

La función escribe el valor del parámetro de entrada "#val" en la última fila de la tabla.

```
SCL
#h:=(#tab.last+1) MOD #tab.lng;    // Incrementar índice
IF #h <> #tab.first THEN           // Consultar si los índices se adelantan
   #tab.x[#tab.last] := #val;      // Escribir valor
   #tab.last := #h;               // Guardar índice del siguiente elemento libre
   #Attend:= false;
ELSE
```

```
SCL
  #Attend:= true;  // Error >> La tabla está llena, el valor no puede
introducirse
END_IF;
```

Ejemplo de migración de "FIFO: Emitir primer valor de la tabla"

Procedimiento en la migración de instrucciones de tabla (clase de migración 4)

Las instrucciones del grupo "Instrucciones de tabla" ya no están disponibles en S7-1500 y no pueden sustituirse por la migración, ya que se basan en un direccionamiento absoluto. No obstante, en S7-1500 hay disponibles cómodas funciones que permiten implementar las instrucciones manualmente. A continuación se describe un ejemplo para la implementación de la instrucción "FIFO: Emitir primer valor de la tabla".

Implementación

El siguiente programa SCL muestra la implementación de la función de tabla "FIFO". La función del ejemplo se ha implementado como FC en SCL.

La tabla siguiente muestra la declaración de las variables utilizadas:

Operando		Declaración	Tipo de datos	Descripción
val		Output	INT	Valor emitido
tab		InOut	Tipo de datos PLC "Tabla10"	Tabla con 10 filas
	x	InOut	ARRAY of INT [0..9]	Cada elemento ARRAY representa una fila. En el ejemplo se utiliza un ARRAY of INT. También es posible crear ARRAYs con cualquier tipo de datos. Igualmente pueden emplearse ARRAY of STRUCT o un ARRAY de un tipo de datos PLC.
	lng	InOut	INT	Longitud del búfer. El valor se rige por el número de elementos ARRAY de "tab.x".
	first	InOut	INT	Primer valor escrito
	last	InOut	INT	Último valor escrito
h		Temp	INT	Variable auxiliar
Ret_Val			BOOL	Valor de función

La tabla siguiente muestra la FC "Fifo":

La función devuelve el primer valor escrito en la tabla (first in, first out). El valor de función "true" indica que la tabla está vacía.

```
SCL
#h:=(#tab.first+1) MOD #tab.lng; // Incrementar índice
  IF #h <> #tab.last THEN        // Consultar si los índices se
adelantan
    #tab.first := #h;   ·        // Guardar índice del siguiente
valor
    #val := #tab.x[#h];      // Emitir el valor deseado
    #Fifo:= false;
  ELSE
    #Fifo:= true; // Error >> La tabla está vacía, no se emite
ningún valor
   #val := #tab.x[#tab.first]; // Volver a emitir el último valor
válido
  END_IF;
```

Ejemplo de migración de "LIFO: Emitir último valor de la tabla"

Procedimiento en la migración de instrucciones de tabla (clase de migración 4)

Las instrucciones del grupo "Instrucciones de tabla" ya no están disponibles en S7-1500 y no pueden sustituirse por la migración, ya que se basan en un direccionamiento absoluto. No obstante, en S7-1500 hay disponibles cómodas funciones que permiten implementar las instrucciones manualmente. A continuación se describe un ejemplo para la implementación de la instrucción "LIFO: Emitir último valor de la tabla".

Implementación

El siguiente programa SCL muestra la implementación de la función de tabla "LIFO". La función del ejemplo se ha implementado como FC en SCL.

La tabla siguiente muestra la declaración de las variables utilizadas:

Operando		Declaración	Tipo de datos	Descripción
val		Output	INT	Valor emitido
tab		InOut	Tipo de datos PLC "Tabla10"	Tabla con 10 filas
	x	InOut	ARRAY of INT [0..9]	Cada elemento ARRAY representa una fila. En el ejemplo se utiliza un ARRAY of INT. También es posible crear ARRAYs con cualquier tipo de datos. Igualmente pueden emplearse ARRAY of STRUCT o un ARRAY de un tipo de datos PLC.
	lng	InOut	INT	Longitud del búfer. El valor se rige por el número de elementos ARRAY de "tab.x".

Operando		Declaración	Tipo de datos	Descripción
	first	InOut	INT	Primer valor escrito
	last	InOut	INT	Último valor escrito
h		Temp	INT	Variable auxiliar
Ret_Val			BOOL	Valor de función

La tabla siguiente muestra la FC "Lifo":

La función devuelve el último valor escrito en la tabla (last in, first out). El valor de función "true" indica que la tabla está vacía.

SCL

```
#h:=(#tab.last-1) MOD #tab.lng;          // Decrementar índice
  IF #h<0 THEN #h:=#h + #tab.lng; END_IF; // Solo se permiten
índices positivos
   IF #h <> #tab.first THEN              // Consultar si los
índices se adelantan
    #tab.last := #h;                      // Guardar índice
del siguiente valor
    #val := #tab.x[#h];              // Emitir el valor deseado
    #Lifo:= false;
  ELSE
    #Lifo:= true;  // Error >> La tabla está vacía, no se
emite ningún valor
    #val := #tab.x[#tab.last];           // Volver a emitir
el último valor válido
  END_IF;
```

Migración de instrucciones de comunicación

SIMATIC NET CP/CM

Migración de bloques de programa

Introducción

S7-300/400 utiliza bloques de programa específicos para funciones de comunicación a través de PCs SIMATIC NET. S7-1500 ofrece un juego de instrucciones modificado para estas funciones de comunicación.

Las tablas siguientes proporcionan una vista general de la migración de los bloques de programa utilizados en S7-300/400.

Encontrará más información sobre la indicación de la columna "Clase" en el capítulo Indicaciones sobre la migración de instrucciones (Página 195)

Bloques de programa para Industrial Ethernet / PROFINET

Servicio de comunicación / rango de funciones	Tipo de instrucción	Librería		Instrucción en S7-1500	Clase
		CP 300	CP 400		
SEND / RECEIVE (servicios de comunicación abierta)	AG_SEND Migración AG_SEND/ AG_LSEND (Página 214)	x	x	T_SEND	3
	AG_LSEND Migración AG_SEND/ AG_LSEND (Página 214)	x	x	T_SEND	3
	AG_SEND (vía UDP) Migración AG_SEND/ AG_LSEND (Página 214)	x	x	TUSEND	3
	AG_SEND (e-mail) Migración AG_SEND/ AG_LSEND (Página 214)	x	x	TMAIL_C	3
	AG_RECV Migración AG_RECV/ AG_LRECV (Página 215)	x	x	T_RCV	3
	AG_LRECV Migración AG_RECV/ AG_LRECV (Página 215)	x	x	T_RCV	3
	AG_LOCK	x	x	-	4
	AG_UNLOCK	x	x	-	4
	AG_CNTRL	x	x	T_DIAG, T_RESET, TCON, TDISCON	3
Conexiones programadas	IP_CONFIG	x	x	T_CONFIG	3
FTP	FTP_CMD	x	x	FTP_CMD	3
PROFINET IO	PNIO_SEND	x		-	4 *)
	PNIO_RECV	x		-	4 *)
	PNIO_RW_REC	x		-	4 *)
	PNIO_ALARM	x		-	4 *)

*) Deben utilizarse otros mecanismos de acceso.

Bloques de programa para PROFIBUS

Servicio de comunicación / rango de funciones	Tipo de instrucción	Librería		Instrucción en S7-1500	Caso
		CP 300	CP 400		
SEND / RECEIVE (servicios de comunicación abierta)	AG_SEND Migración AG_SEND/ AG_LSEND (Página 214)	x	x	T_SEND	3
	AG_LSEND Migración AG_SEND/ AG_LSEND (Página 214)	-	x	T_SEND	3
	AG_RECV Migración AG_RECV/ AG_LRECV (Página 215)	x	x	T_RCV	3
	AG_LRECV Migración AG_RECV/ AG_LRECV (Página 215)	-	x	T_RCV	3
PROFIBUS DP	DP_SEND	x	-	-	4 *)
	DP_RECV	x	-	-	4 *)
	DP_DIAG	x	-	-	4 *)
	DP_CTRL	x	-	-	4 *)

*) Deben utilizarse otros mecanismos de acceso.

Parámetros formales adicionales

En S7-1500 las instrucciones tienen parámetros formales adicionales que son necesarios para la ejecución asíncrona, por ejemplo. Estos parámetros adicionales deben abastecerse con parámetros actuales adecuados después de la migración.

Encontrará una descripción detallada de los parámetros en la Ayuda en pantalla de S7-1500.

Consulte también:

Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277)

Ejemplo

En el capítulo Migración de instrucciones de bloques de datos (Página 203) encontrará un ejemplo básico para la representación de una instrucción migrada.

Migración AG_SEND/AG_LSEND

Migración de los parámetros de llamada - Comunicación vía ISO-Transport, ISO-on-TCP, TCP

Tabla 4-1 Migración AG_SEND/AG_LSEND

AG_SEND / AG_LSEND		T_SEND		Descripción / indicaciones
Parámetro	Tipo	Parámetro	Tipo	
ACT	BOOL	REQ	BOOL	
ID	INT	ID	CONN_OUC (WORD)	
LADDR	WORD	--	--	
SEND	ANY	DATA	VARIANT	
--	--	ADDR	VARIANT	
LEN	INT	LEN	UINT	
DONE	BOOL	DONE	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Pasos adicionales

- Configure el tipo de conexión correspondiente con parámetros de dirección relevantes.

Migración de los parámetros de llamada - Comunicación vía UDP

Tabla 4-2 Migración AG_SEND/AG_LSEND

AG_SEND / AG_LSEND		TUSEND		Descripción / indicaciones
Parámetro	Tipo	Parámetro	Tipo	
ACT	BOOL	REQ	BOOL	
ID	INT	ID	CONN_OUC (WORD)	
LADDR	WORD	--	--	
SEND	ANY	DATA	VARIANT	
--	--	ADDR	VARIANT	
LEN	INT	LEN	UINT	
DONE	BOOL	DONE	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Pasos adicionales

- Configure el tipo de conexión correspondiente con parámetros de dirección relevantes.

Migración de los parámetros de llamada - Comunicación vía E-MAIL

Tabla 4-3 Migración AG_SEND/AG_LSEND

AG_SEND / AG_LSEND		TMAIL_C		Descripción / indicaciones
Parámetro	Tipo	Parámetro	Tipo	
ACT	BOOL	REQ	BOOL	
ID	INT	--	--	
LADDR	WORD	--	--	
SEND	ANY	TO_S	STRING	
SEND	ANY	CC	STRING	
SEND	ANY	SUBJECT	STRING	
SEND	ANY	TEXT	STRING	
SEND	ANY	ATTACHMENT	VARIANT	
--	--	ATTACHMENT _NAME	STRING	
--	--	MAIL_ADDR_PARAM	VARIANT	
LEN	INT	--	--	
DONE	BOOL	DONE	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Pasos adicionales

- Configure el tipo de conexión correspondiente con parámetros de dirección relevantes.

Migración AG_RECV/AG_LRECV

Migración de los parámetros de llamada

Tabla 4-4 Migración AG_RECV/AG_LRECV

AG_RECV/AG_LRECV		TRCV/TURCV		Descripción / indicaciones
Parámetro	Tipo	Parámetro	Tipo	
		EN_R	BOOL	
ID	INT	ID	CONN_OUC (WORD)	
LADDR	WORD	--	--	

AG_RECV/AG_LRECV		TRCV/TURCV		Descripción / indicaciones
Parámetro	Tipo	Parámetro	Tipo	
--	--	LEN	UDINT	
RECV	ANY	DATA	VARIANT	
--	--	ADDR	VARIANT	
LEN	INT	LEN	UINT	
NDR	BOOL	NDR	BOOL	
--	--	BUSY	BOOL	
ERROR	BOOL	ERROR	BOOL	
STATUS	WORD	STATUS	WORD	

Pasos adicionales

- Configure el tipo de conexión correspondiente con parámetros de dirección relevantes.

Punto a punto CP/CM

Migración de bloques de programa PtP

Introducción

S7-300/400 utiliza bloques de programa específicos para funciones de comunicación a través de CP punto a punto.

S7-1500 ofrece un juego de instrucciones modificado para estas funciones de comunicación.

En la migración se le consultará si desea utilizar el juego de instrucciones modificado para los módulos de comunicación de S7-1500 o los bloques de programa para los procesadores de comunicaciones de S7-300/400.

La tabla siguiente ofrece una vista general de la migración de los bloques de programa utilizados en el S7-300/400 a instrucciones del S7-1500.

Encontrará más información sobre la indicación de la columna "Clase" en el capítulo Indicaciones sobre la migración de instrucciones (Página 195)

Bloque de programa para la comunicación punto a punto

Servicio de comunicación / rango de funciones	Bloque	Librería			Instrucción en S7-1500	Clase
		CP 300	CP 400	ET 200		
Bloques para CP 340	P_RCV	x	-	-	Receive_P2P	3 [1)]
	P_SEND	x	-	-	Send_P2P	3 [1)]
	P_RESET	x	-	-	Receive_Reset	3 [1)]

Servicio de comunicación / rango de funciones	Bloque	Librería			Instrucción en S7-1500	Clase
		CP 300	CP 400	ET 200		
	V24_STAT_340	x	-	-	Signal_Get	3 [1]
	V24_SET_340	x	-	-	Signal_Set	3 [1]
	P_PRINT	x	-	-	-	4
Bloques para CP 341	P_RCV_RK	x	-	-	Receive_P2P	3 [1]
	P_SND_RK	x	-	-	Send_P2P	3 [1]
	V24_STAT	x	-	-	Signal_Get	3 [1]
	V24_SET	x	-	-	Signal_Set	3 [1]
	P-PRT341	x	-	-	-	4
Bloques para CP 440	RECV_440	-	x	-	Receive_P2P	3 [1]
	SEND_440	-	x	-	Send_P2P	3 [1]
	RES_RECV	-	x	-	Receive_Reset	3 [1]
Bloques para CP 441	V24_STAT_441	-	x	-	Signal_Get	3 [1]
	V24_SET_441	-	x	-	Signal_Set	3 [1]
Bloques para Modbus	MODB_341	x	-	-	-	4
	MODB_441	-	x	-	-	4
Bloques para ET 200S 1SI	S_RCV	-	-	x	Receive_P2P	3 [1]
	S_SEND	-	-	x	Send_P2P	3 [1]
	S_VSTAT	-	-	x	Signal_Get	3 [1]
	S_VSET	-	-	x	Signal_Set	3 [1]
	S_XON	-	-	x	Port_Config	3 [1]
	S_RTS	-	-	x	Port_Config	3 [1]
	S_V24	-	-	x	Send_Config	3 [1]
	S_MODB	-	-	x	-	4
	S_USST	-	-	x	-	4
	S_USSR	-	-	x	-	4
	S_USSI	-	-	x	-	4
Bloques para CPU compactas S7 300	SEND_PTP	x	-	-	Send_P2P	3 [1]
	RCV_PTP	x	-	-	Receive_P2P	3 [1]
	RES_RCVB	x	-	-	Receive_Reset	3 [1]

[1] Encontrará información sobre la migración de esta instrucción en este enlace (Página 195).

Parámetros formales adicionales

En el S7-1500, las instrucciones tienen parámetros formales adicionales. Estos parámetros adicionales deben abastecerse con parámetros actuales adecuados después de la migración.

Encontrará una descripción detallada de los parámetros en Siemens Industry Online Support, en las preguntas frecuentes actuales sobre el tema "Migración de PLC" (http://support.automation.siemens.com/WW/view/es/67858106).

4.2.4.3 Migración de programas KOP/FUP a S7-1500

Indicaciones sobre la migración de programas KOP/FUP

---(SAVE): Cargar resultado lógico en registro BR

La instrucción no está disponible en S7-1500. La migración procesa la instrucción del siguiente modo:

- Si se ha utilizado la instrucción "---(SAVE)" en el último segmento de un bloque se reemplazará por la instrucción "---(RET)" con el parámetro "RLO".
- Si se ha utilizado la instrucción "---(SAVE)" en un segmento que no es el último del bloque no podrá reemplazarse. Modifique el programa manualmente. Por ejemplo, guarde el RLO en una variable PLC, una variable DB o una variable local y devuelva la variable al bloque invocante al final del bloque con la instrucción --(RET).

Temporizadores CEI y contadores CEI con conexión EN/ENO

Las instrucciones se han adaptado a la norma IEC 1131-3 para S7-1500. Ahora están enlazadas con el circuito a través de los parámetros "IN" y, por tanto, ya no disponen de los parámetros "EN" y "ENO". La migración notifica un error cuando el programa de partida contiene un temporizador o un contador CEI con una operación lógica previa que se analiza con el parámetro de entrada "EN".

Agregue una instrucción de salto entre la operación lógica previa y el temporizador o contador CEI para llamar el temporizador o contador CEI en función del RLO. Existen las siguientes instrucciones de salto:

- ---(JMP): Saltar si RLO = 1
- ---(JMPN): Saltar si RLO = 0

---(CALL) Llamar bloque sin parámetros

La instrucción no está disponible en S7-1500. La migración adapta el programa de tal forma que la llamada se realice a través de un cuadro.

Master Control Relais

El Master Control Relais no está disponible en S7-1500. La migración notifica un error. Modifique el programa manualmente. Por ejemplo, use las variables temporales para ejecutar instrucciones o segmentos de forma condicional.

Cambio de idioma dentro de un bloque

Los contenidos de los registros, de los acumuladores y de la palabra de estado solo están disponibles en segmentos AWL. Si un segmento KOP o FUP sigue a un segmento AWL, desde el segmento KOP o FUP no se podrá acceder a los contenidos de registros que antes se habían modificado en AWL. Sin embargo, en un posterior segmento AWL, los contenidos de

los registros vuelven a estar disponibles. El bit RLO es una excepción: al realizar un cambio de idioma se pone en "indefinido" y en los siguientes segmentos ya no estará disponible.

Consulte también: Transferencia de valores a través de registros en caso de cambio de lenguaje (Página 224)

Consulte también

--(RET): Retroceder (Página 3519)

RET: Retroceder (Página 3832)

Mecanismo EN/ENO (Página 2539)

Direcciones parcialmente especificadas en KOP/FUP

Introducción

El direccionamiento de variables DB sin indicar el nombre del DB se denomina direccionamiento parcialmente especificado. El direccionamiento parcialmente especificado no se soporta en S7-1500 para KOP/FUP. Las instrucciones "---(OPN)" y "---(OPNI)" utilizadas con frecuencia en relación con el direccionamiento parcialmente especificado tampoco se soportan en S7-1500.

Migración de direcciones parcialmente especificadas

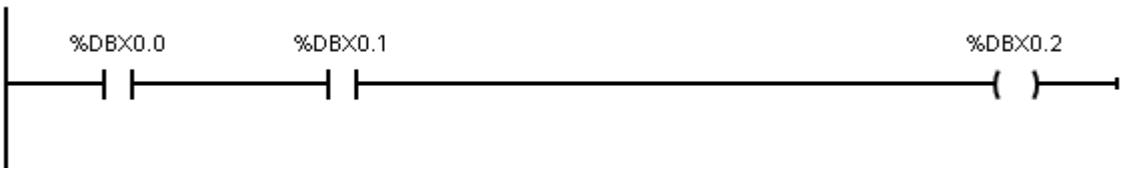
La migración procesa direcciones parcialmente especificadas, en lo posible, del siguiente modo:

- Si el bloque de datos se determina de forma unívoca, la migración reemplaza las direcciones parcialmente especificadas por direcciones totalmente especificadas indicando el DB.
- Si el DB se transfiere indirectamente a través del registro de bloques de datos, el bloque de datos no puede determinarse de forma unívoca. En ese caso la migración agrega un parámetro del tipo de datos "DB_Any" en la interfaz del bloque. A este parámetro se transfiere el nombre del bloque de datos. A continuación la migración reemplaza todas las direcciones parcialmente especificadas por direcciones con la siguiente sintaxis: `#<Nombre del parámetro>.%<dirección absoluta>.`

Ejemplo

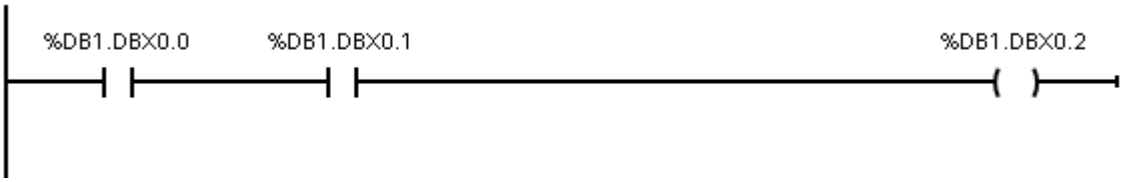
El siguiente ejemplo muestra la migración de un direccionamiento parcialmente especificado. En la primera figura se muestra el programa antes de la migración:

Segmento 1: Las variables se direccionan de forma parcialmente especificada.



La figura siguiente muestra el programa después de la migración:

Segmento 1: El DB en el que están las variables se puede determinar de forma unívoca. El programa se ha migrado automáticamente. Para las variables en "DB1" el direccionamiento es totalmente especificado.



Nota

Migración de parámetros de bloque parcialmente especificados

La migración convierte también parámetros de DB parcialmente especificados en parámetros totalmente especificados. Tenga en cuenta que esto puede modificar la manera en que se transfieren parámetros al bloque llamado: posiblemente, el bloque llamado no accederá ya directamente al parámetro actual a través de la dirección totalmente especificada, sino que trabajará con una copia que se transferirá en la llamada del bloque.
Por tanto, compruebe si la semántica del programa migrado coincide con la del programa de partida.
Encontrará más información en: Acceso a parámetros de bloque durante la ejecución del programa (Página 2273)

Si no se desea este ajuste, puede utilizarse, por ejemplo, un parámetro con tipo de datos estructurado en lugar de un parámetro de bloque simple.
Defina como parámetro formal el tipo de datos PLC (UDT) y transfiera una variable de este tipo o un DB derivado del tipo de datos PLC(UDT).

Consulte también

Direccionar operandos (Página 2293)

Acceso a los bits de estado en KOP/FUP

Introducción

Para alcanzar el máximo rendimiento posible del procesamiento de programas en CPUs de la serie S7-1500, se permite transferir valores entre los bloques únicamente a través de las interfaces del bloque, bloques de datos globales o variables PLC. En KOP y FUP no existe la posibilidad de transferir valores a través de la palabra de estado.

Migración de las instrucciones "--| |--: Consultar si el bit de estado es 1" / "--|/|--: Consultar si el bit de estado es 0"

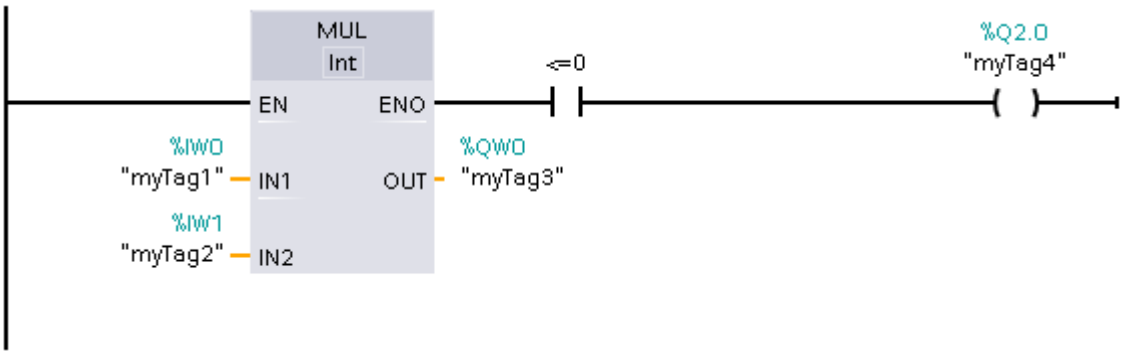
La instrucciones no están disponibles en S7-1500. La migración distingue los siguientes casos:

- Si una consulta del bit de estado se sitúa inmediatamente después de una instrucción matemática en el mismo segmento, entonces se empleará un comparador en su lugar.
- En cualquier otro caso, el programa dará error después de la migración. En ese caso es necesario modificar el programa. Por ejemplo, en lugar de la consulta del bit de estado OV utilice una consulta de la salida ENO. Si se enlaza la negación de varias consultas ENO con la operación lógica O, es posible reemplazar la consulta del bit de estado OS.

Ejemplo

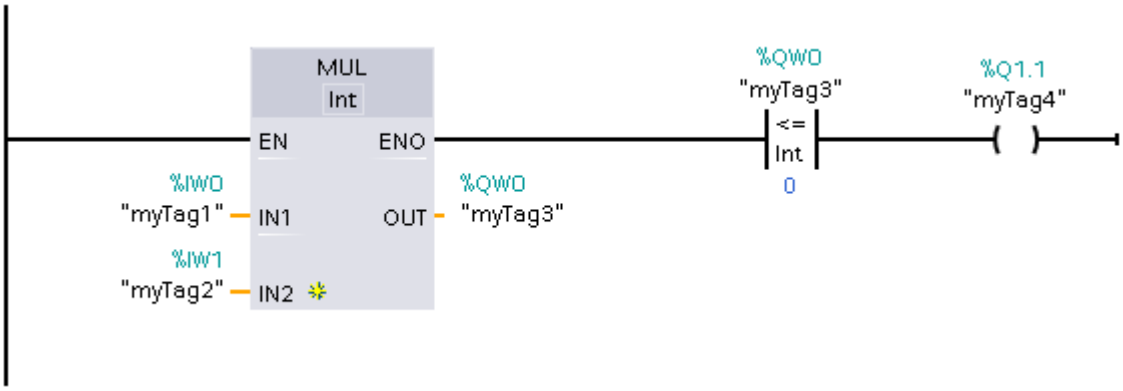
El siguiente ejemplo muestra la migración de una consulta del bit de estado. En la primera figura se muestra el programa antes de la migración.

Segmento 1: La instrucción "<=0" consulta el bit de estado "A1". Se indica si el resultado de la multiplicación es mayor o igual a cero.



La figura siguiente muestra el programa después de la migración.

Segmento 1: El programa se ha podido migrar automáticamente. La instrucción "<=0" se ha eliminado. En lugar de eso se consulta el evento de la multiplicación mediante una comparación a través de la variable "myTag3".



4.2.4.4 Migración de programas AWL a S7-1500

Indicaciones sobre la migración de programas AWL

Transferencia de valores a través de registros y de la palabra de estado

Para alcanzar el máximo rendimiento posible del procesamiento de programas en CPUs de la serie S7-1500, se permite transferir valores entre los bloques únicamente a través de las interfaces del bloque, bloques de datos globales o variables PLC.

En KOP y FUP no se pueden transferir valores a través de registros (p. ej. acumuladores AR1, AR2, DB, DI) o de la palabra de estado. En AWL la transferencia de valores es posible con algunas restricciones. Sin embargo, tenga en cuenta que la ejecución del programa se ralentiza al usar estas áreas para transferir valores entre distintos bloques.

Las siguientes reglas rigen en AWL:

- Los contenidos de los registros, de los acumuladores y de la palabra de estado solo están disponibles en segmentos AWL. Si un segmento KOP o FUP sigue a un segmento AWL, desde el segmento KOP o FUP no se podrá acceder a los contenidos de registro que antes se habían colocado en AWL. No obstante, en un segmento AWL posterior los contenidos de registro vuelven a estar disponibles.
  El bit RLO es una excepción: al realizar un cambio de idioma se pone en "indefinido" y en los siguientes segmentos ya no estará disponible.
- Los valores de los registros, acumuladores y de la palabra de estado no se transfieren a los bloques llamados. La única excepción son las instrucciones "CC" y "UC". Si utiliza "UC" o "CC", para transferir al bloque llamado parámetros a través de registros, de la palabra de estado o de acumuladores, active la opción "Alimentación de parámetros a través de registros" en las propiedades del bloque llamado. Tenga en cuenta que esta opción solo está disponible para bloques AWL con acceso estándar y que el bloque no puede tener parámetros formales. Si esta opción está activada, los contenidos de registro pueden transferirse entre bloques. También en este caso el bit RLO es una excepción: al salir del bloque se pone en "indefinido" y ya no estará disponible tras una llamada de bloque.
- Para transferir una información de error al bloque invocante, se puede utilizar el bit BR. La información de error primero debe guardarse en el bloque llamado con la instrucción "SAVE" en el bit BR. A continuación se puede leer el bit BR en el bloque invocante.
- Después de cada acceso a un bloque de datos indicando una dirección totalmente especificada (p. ej. `%DB10.DBW10`), el registro de bloques de datos DB se pone a "0". Un acceso posterior parcialmente especificado provoca un error durante la compilación.
- Si se direcciona simbólicamente un parámetro formal local desde la interfaz de bloque de un FB en S7-1500 (p. ej. con la instrucción L #myIn), siempre se accede al bloque de datos que se ha indicado como instancia en la llamada del bloque. Las instrucciones OPNDI, L AR2, +AR2, TDB, TAR cambian el contenido del registro DI o de direcciones, pero los registros ya no se evalúan en el direccionamiento de parámetros formales locales.

Encontrará ejemplos sobre la programación en los siguientes capítulos.

Master Control Relais

El Master Control Relais no está disponible en S7-1500. La migración notifica un error. Modifique el programa manualmente. Formule las condiciones, p. ej. en forma de parámetros de bloque, para ejecutar instrucciones o segmentos de forma condicional.

LEAVE y ENT

Las instrucciones "LEAVE" y "ENT" no se ofrecen en S7-1500 porque solo existen dos acumuladores. La migración notifica un error. Modifique el programa manualmente. Por ejemplo, use variables temporales para crear resultados provisionales.

Parámetros de bloque del tipo de parámetro "Block_DB"

El tipo de parámetro "Block_DB" no está disponible en S7-1500. La migración modifica estos parámetros y les asigna en su lugar el tipo de datos "DB_Any".

Sin embargo, en S7-1500 no es posible asignar instrucciones de librería durante la llamada de una instancia en forma de un parámetro del tipo "DB_Any".

El siguiente ejemplo muestra la llamada de un bloque de librería con una instancia variable en una CPU de la familia S7-300/400. Esta secuencia no puede migrarse a S7-1500.

AWL	Explicación
CALL GET, #myBlock_DB	// El bloque de librería "GET" se llama con el bloque de datos de instancia ubicado actualmente en el parámetro de bloque "myBlock_DB".
REQ := <Operando>	
ID := <Operando>	
NDR := <Operando>	
ERROR := <Operando>	

Consulte también

Bases de AWL (Página 2753)

Principios básicos de la palabra de estado (Página 2559)

Cadenas lógicas y saltos

Introducción

En S7-300/400 es posible colocar etiquetas en AWL en cualquier instrucción para saltar de forma controlada por el programa a la posición marcada. No obstante, si define la meta de salto dentro de una unidad de programa lógica en la que el RLO no tenga ningún estado definido, pueden aparecer resultados inesperados durante la ejecución del programa. La CPU puede pasar a modo STOP. Por eso, durante la elaboración del programa debe procurar utilizar etiquetas solo al principio de una secuencia de programa lógica.

Cuando los programas contienen saltos desde cadenas lógicas o a cadenas lógicas, en S7-1500 se produce un error de compilación. Así pueden evitarse errores en tiempo de ejecución durante la ejecución del programa.

El siguiente ejemplo muestra un salto a una cadena lógica. El RLO no tiene ningún estado definido en la etiqueta n_OK de la fila 6:

```
1        A     #Motor_Enabled
2        AN    #Motor_EmergencyStop
3        JC    n_OK
4        =     #Motor_Start
5        AN    #Motor_SpeedOK
6  n_OK: AN    #Motor_BrakesEnabled
7        =     #Motor_Stop
```

Migración de saltos a cadenas lógicas

La migración no cambia las marcas de su programa AWL. Sin embargo, al compilar se notifica un error cuando se reconocen etiquetas dentro de cadenas lógicas. Modifique el programa. Utilice las instrucciones SET o CLR para marcar el inicio de una nueva "unidad lógica" y pasar el RLO a un estado definido.

El siguiente ejemplo muestra cómo puede modificar el programa tras la migración utilizando "SET":

```
1        A     #Motor_Enabled
2        AN    #Motor_EmergencyStop
3        JC    n_OK
4        =     #Motor_Start
5        AN    #Motor_SpeedOK
6        AN    #Motor_BrakesEnabled
7        =     #Motor_Stop
8        JU    End
9  n_OK: SET
10       AN    #Motor_BrakesEnabled
11       =     #Motor_Stop
12 End:  NOP 0
```

Transferencia de valores a través de registros en caso de cambio de lenguaje

Introducción

Los contenidos de los registros, de los acumuladores y de la palabra de estado solo están disponibles en segmentos AWL. Si un segmento KOP o FUP sigue a un segmento AWL, desde el segmento KOP o FUP no se podrá acceder a los contenidos de registro que antes se habían

colocado en AWL. No obstante, en un segmento AWL posterior los contenidos de registro vuelven a estar disponibles.

El bit RLO es una excepción: al realizar un cambio de idioma se pone en "indefinido" y en los siguientes segmentos ya no estará disponible.

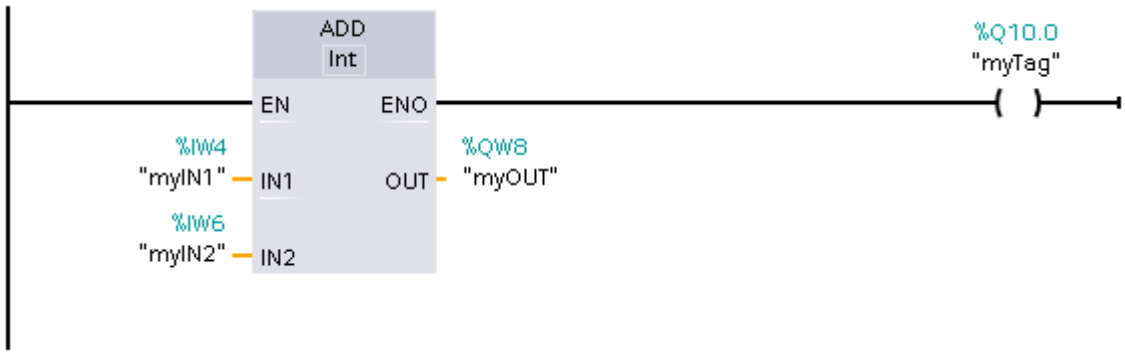
Migración de transferencias de valores a través de registros en caso de cambio de lenguaje

Si se accede a registros en un segmento KOP o FUP migrado, se notifica un error al compilar. Modifique el programa de tal forma que los registros solo se posicionen y lean en segmentos AWL.

Ejemplo

El siguiente ejemplo muestra la migración de un acceso a acumulador. En las dos primeras figuras se muestra el programa antes de la migración:

Segmento 1: El operando "myIN1" se carga en el acumulador 1. A continuación se carga "myIN2" al acumulador 1 y de esta forma se transfiere "myIN1" al acumulador 2. Ahora se suman los dos valores. El resultado se guarda en el acumulador 1 y desde allí se asigna al operando "myOUT".



Segmento 2: El operando "myIN3" se carga en el acumulador 1. El operando "myOUT" que aún existe en el acumulador 1 se transfiere con ello al acumulador 2, y puede agregarse inmediatamente al operando "myIN3". No es necesario cargar nuevamente el operando "myOUT" de forma explícita.

```
1    L    "myIN3"    %IW12
2    +I
3    T    "myResult"    %IW14
4
```

Tras la migración, el segmento 2 presenta errores porque el segmento KOP ya no escribe valores en los acumuladores. Es necesario adaptar el programa manualmente para cargar ambos valores en AWL de forma explícita en los acumuladores.

La figura siguiente muestra el segmento 2 tras solucionar el error.

Segmento 2: Ambos valores se cargan en AWL de forma explícita antes de sumarse.

```
1    L    "myIN3"      %IW12
2    L    "myOUT"      %QW8
3    +I
4    T    "myResult"   %QW14
5
```

Consulte también

Principios básicos de la palabra de estado (Página 2559)

Bases de AWL (Página 2753)

Transferencia de valores a través de registros o de la palabra de estado durante la llamada de bloque

Migración de transferencias de valores a través de registros o de la palabra de estado durante la llamada de bloque

En un cambio de bloque los valores de los registros, los acumuladores y la palabra de estado se ponen a "0" o reciben el estado "indefinido". Por ello no pueden transferirse a los bloques llamados.

La única excepción son las instrucciones "CC" y "UC". Si utiliza "UC" o "CC", para transferir al bloque llamado parámetros a través de registros, de la palabra de estado o de acumuladores, active la opción "Alimentación de parámetros a través de registros" en las propiedades del bloque llamado. Tenga en cuenta que esta opción solo está disponible para bloques AWL con acceso estándar y que el bloque no puede tener parámetros formales. Si esta opción está activada, los contenidos de registro pueden transferirse entre bloques. El bit RLO es una excepción: en una transición de bloque siempre adopta el estado "indefinido" y ya no estará disponible tras una llamada de bloque.

Para transferir una información de error al bloque invocante, se puede utilizar el bit BR. La información de error primero debe guardarse en el bloque llamado bit BR. Utilice para ello p. ej. las instrucciones "SAVE" o "SPBNB". A continuación se puede leer el bit BR en el bloque invocante.

Si tras una llamada de bloque se accede a contenidos de registros que se habían modificado en el bloque llamado, se notifica un error. Luego es necesario modificar el programa. Utilice por ejemplo variables en bloques de datos o variables PLC para devolver valores al bloque invocante.

Ejemplos

El ejemplo siguiente muestra cómo hay que cambiar el programa para transferir valores a un bloque invocante a través de registros.

En la primera tabla se observa el programa antes de la migración:

AWL	Explicación
CALL "MyFB", "MyFB_DB"	El RLO del bloque "MyFB" se asigna al operando "MyBit" después de su ejecución.

AWL	Explicación
= #MyBit	

En la segunda tabla se observa cómo hay que modificar el programa.

AWL	Explicación
CALL "MyFB", "MyFB_DB"	En el bloque llamado "MyFB" se escribe el RLO actual en un punto cualquiera con la instrucción "SAVE" en el bit BR.
A BR	El bit BR se lee en el bloque invocante.
= #MyBit	El valor del bit BR se asigna al operando "MyBit".

Consulte también

Principios básicos de la palabra de estado (Página 2559)

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Fundamentos del acceso a bloques (Página 2258)

Proteger bloques (Página 3045)

Bases de AWL (Página 2753)

Transferencia de valores a través de registros con las instrucciones CC y UC

Introducción

En S7-300/400 es posible programar llamadas de bloque mediante las instrucciones UC y CC. La transferencia de parámetros al bloque invocante no se lleva a cabo en la interfaz sino a través de registros, p. ej. AR1, AR2, DB, DI,, de los acumuladores o de la palabra de estado.

Estas llamadas ralentizan la ejecución del programa y por ello ya no son posibles en S7-1500 por ajuste predeterminado. En su lugar se utiliza la instrucción CALL en S7-1500. Sin embargo, con la instrucción CALL no es posible hacer llamadas de bloque indirectas. Si desea utilizar UC o CC para llamadas de bloque indirectas es necesario activar la opción "Alimentación de parámetros a través de registros" en las propiedades del bloque llamado. De ese modo podrá transmitir contenidos de registro entre diferentes bloques. El bit RLO es una excepción: en una transición de bloque adopta el estado "indefinido" y ya no estará disponible tras una llamada de bloque.

Tenga en cuenta que esta opción solo está disponible para bloques AWL con acceso estándar y que el bloque no puede tener parámetros formales. Si esta opción está activada, los contenidos de registro pueden transferirse entre bloques. También en este caso el bit RLO es una excepción: en una transición de bloque adopta el estado "indefinido" y ya no estará disponible tras una llamada de bloque.

Al crear elementos de librería con protección de know-how ya no es necesario utilizar llamadas de bloque indirectas. El TIA Portal le ayuda con un automatismo: Al insertar un bloque desde

una librería se resuelven automáticamente los posibles conflictos numéricos con bloques que ya se encuentran en el programa de usuario.

Migración de llamadas de bloque a través de "UC" o "CC"

La migración procesa las llamadas de bloque a través de "UC" o "CC" del siguiente modo:

- La instrucción "UC FC" con indicación de un número de bloque se reemplaza por la instrucción "CALL" .
- La instrucción "CC FC" con indicación de un número de bloque se reemplaza por la instrucción "CALL". Asimismo, se inserta una instrucción de salto que realiza la llamada condicional.
- Las instrucciones "UC FC" y "CC FC" con indicación indirecta del número de bloque permanecen invariables.
- Las instrucciones "UC FB" y "CC FB" con indicación directa o indirecta de un número de bloque permanecen invariables.

Ejemplo

El siguiente ejemplo muestra la migración de llamadas de bloque a través de "UC".

En la primera tabla se observa el programa antes de la migración:

AWL
UC FC 10
UC FC[#temp0]
UC FB 10
UC FB [#temp0]

En la siguiente tabla se observa el programa después de la migración:

AWL	Explicación
CALL FC 10	
UC FC[#temp0]	La opción "Alimentación de parámetros a través de registros" debe haberse activado en el bloque llamado.
UC FB 10	La opción "Alimentación de parámetros a través de registros" debe haberse activado en el bloque llamado.
UC FB [#temp0]	La opción "Alimentación de parámetros a través de registros" debe haberse activado en el bloque llamado.

Consulte también

Principios básicos de la palabra de estado (Página 2559)

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Bases de AWL (Página 2753)

Direcciones totalmente especificadas en AWL

Introducción

El direccionamiento de variables DB con indicación del nombre o el número de DB se denomina direccionamiento totalmente especificado. Después de cada acceso a un bloque de datos indicando una dirección totalmente especificada, el registro de bloques de datos se pone a "0". Si desea acceder nuevamente al registro del DB después de un acceso mediante una dirección totalmente especificada, primero deberá asignar un valor al registro del DB con la instrucción OPN DB.

Migración de direcciones totalmente especificadas

Si es necesario, la migración agrega la instrucción OPN después de un acceso mediante dirección totalmente especificada para volver a cargar el bloque de datos actual al registro de bloques de datos.

Ejemplo

El siguiente ejemplo muestra la migración de una dirección totalmente especificada.

En la primera tabla se observa el programa antes de la migración:

AWL	Explicación
OPN "MyDB"	"MyDB" se carga en el registro de bloques de datos.
L %DBW1	Los elementos de "MyDB" se direccionan con especificación parcial.
L "Global_DB".Data1	A continuación, se direccionan elementos de un DB global de forma totalmente especificada. Como consecuencia, se carga implícitamente el "Global_DB" en el registro de bloques de datos.
L "Global_DB".DBW2	
T DBW[AR1, P#0.0]	El siguiente acceso direcciona el "Global_DB" porque está contenido en el registro DB.

En la siguiente tabla se observa el programa después de la migración:

AWL	Explicación
OPN "MyDB"	

AWL	Explicación
L "MyDB".DBW1	El acceso parcialmente especificado se convierte en un acceso totalmente especificado.
L "Global_DB".Data1	
L "Global_DB".DBW2	Después del acceso mediante una dirección totalmente especificada se inicializa el registro del DB.
OPN "Global_DB"	El "Global_DB" debe cargarse de nuevo con OPN en el registro DB.
T DBW[AR1, P#0.0]	

Consulte también

Bases de AWL (Página 2753)

Principios básicos de la palabra de estado (Página 2559)

Direcciones parcialmente especificadas en AWL

Introducción

El direccionamiento de variables DB sin indicar el nombre o el número del DB se denomina direccionamiento parcialmente especificado. El direccionamiento parcialmente especificado accede a un valor definido en el bloque de datos guardado en ese momento en el registro del DB.

Para direcciones parcialmente especificadas existen las siguientes restricciones en S7-1500:

- En S7-1500 las direcciones con especificación parcial solo están permitidas si previamente se ha activado de forma explícita el registro de DB en el bloque actual. El registro de DB se activa p. ej. con la instrucción "OPN". Solo es posible direccionar con especificación parcial variables en bloques de datos con acceso estándar.
- En S7-1500 los registros de bloques de datos se ponen a "0" durante la llamada de bloque. Por ello no es posible abrir un bloque de datos en un bloque ni direccionar de forma parcialmente especificada elementos de datos del bloque de datos en un bloque subordinado. Active el registro de DB en el bloque actual antes de direccionar una variable DB con especificación parcial.
- El registro de bloque de datos DB también se pone a "0" en cada acceso totalmente especificado (p. ej. %DB10.DBW10). Tras un acceso con dirección totalmente especificada vuelva a activar el registro de DB antes de direccionar una variable DB con especificación parcial.
- Utilizando direcciones parcialmente especificadas se ralentiza el procesamiento del programa en runtime.

Migración de direcciones parcialmente especificadas

La migración procesa direcciones parcialmente especificadas en AWL del siguiente modo:

- La dirección parcialmente especificada se convierte, si es posible, en una dirección totalmente especificada.
- Si el bloque de datos se ha abierto en el bloque lógico invocante y el DB no puede determinarse unívocamente, la migración agrega un parámetro del tipo de datos "DB_Any" en la interfaz del bloque llamado. En ese parámetro se transfiere el nombre del bloque de datos. La migración agrega la instrucción "OPN" en el bloque llamado para abrir el bloque de datos.

Ejemplo

El siguiente ejemplo muestra la migración de direcciones parcialmente especificadas.

En la primera tabla se observa el programa antes de la migración:

AWL	Explicación
L DBW10 L DBW12 +I T DBW14	Un bloque de datos se ha abierto en el bloque invocante y se ha transferido al registro del DB. En el bloque actual se adoptan los valores "DBW10", "DBW12" y "DBW14" desde el bloque de datos ubicado actualmente en el registro del DB.

En la siguiente tabla se observa el programa después de la migración:

AWL	Explicación
OPN "PlcmigTempBlockDB" L DBW10 L DBW12 +I T DBW14	La migración agrega un parámetro del tipo de datos "DB_Any" en la interfaz del bloque llamado. En ese parámetro se transfiere el nombre del bloque de datos. La migración agrega la instrucción "OPN" en el bloque llamado para abrir el bloque de datos trasferido en la interfaz.

Nota

Migración de parámetros de bloque parcialmente especificados

La migración convierte también parámetros de DB parcialmente especificados en parámetros totalmente especificados. Tenga en cuenta que esto puede modificar la manera en que se transfieren parámetros al bloque llamado: Posiblemente, el bloque llamado ya no accede directamente a los parámetros actuales a través de la dirección totalmente especificada, sino que trabaja con una copia, que se transfiere en la llamada del bloque.
Por ello, compruebe si la semántica del programa migrado todavía coincide con la del programa de origen.
Encontrará más información en: Acceso a parámetros de bloque durante la ejecución del programa (Página 2273)

Si no se desea esta adaptación, puede utilizarse, por ejemplo, un parámetro con tipo de datos estructurado en lugar de un parámetro de bloque simple.
Defina como parámetro formal el tipo de datos PLC (UDT) y transfiera una variable de este tipo o un DB derivado del tipo de datos PLC(UDT).

Ejemplos:

```
CALL  "MyFC"
          InStruct :="DBofUDT"
```

oder

```
    CALL  "MyFC"
          InStruct := "DBArrayOfUDT".a[#i]
```

Consulte también

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Bases de AWL (Página 2753)

Principios básicos de la palabra de estado (Página 2559)

Accesos al DB instancia en AWL

Introducción

Las instrucciones "OPNDI" o "CDB" colocan un bloque de datos en el registro DI. En S7-300/400 será bloque de datos instancia el bloque que esté abierto allí. El direccionamiento simbólico posterior de un parámetro formal local de la interfaz de un FB (IN, OUT, InOut, Static)

ya no direcciona el bloque de datos que se indicó como bloque de datos de instancia en la llamada del bloque, sino el bloque de datos que se encuentra en el registro DI. Para direccionar después simbólicamente un parámetro formal local desde la interfaz del bloque, es necesario cargar el DB instancia en el registro DI.

Tras las instrucciones "L AR2", "+ AR" y "TAR", en el S7-300/400 tampoco es posible direccionar simbólicamente un parámetro formal desde la interfaz del bloque, ya que las instrucciones destruirían la base para el acceso al parámetro.

Este comportamiento se ha corregido en S7-1500: si se direcciona simbólicamente un parámetro formal local desde la interfaz de bloque en S7-1500 (p. ej. con la instrucción L #myIn), siempre se accede al bloque de datos que se ha indicado como instancia en la llamada del bloque. Las instrucciones OPNDI, L AR2, +AR2, TDB, TAR cambian el contenido del registro DI o de direcciones, pero los registros ya no se evalúan en el direccionamiento de parámetros formales locales.

Migración de accesos a variables locales en el DB instancia

La migración no modifica los accesos programados. Si en el programa original ha utilizado las instrucciones OPNDI, L AR2, +AR2, TDB, TAR,, etc., es posible que cambie la semántica del programa.
Para recuperar la semántica original, hay que cambiar el programa manualmente. A menudo ya no es necesario direccionar datos a través de registros. En lugar de ello aproveche la posibilidad de utilizar ARRAYs dentro del DB instancia e indexar los elementos ARRAY de forma indirecta.

Ejemplo 1

El siguiente ejemplo muestra la semántica modificada de la instrucción "OPNDI":

AWL	Explicación
L #MyIn1 L #MyIn2 +I T #MyOut3	Las instrucciones "L" y "T" direccionan variables locales declaradas en la interfaz del bloque. Los valores de las variables se encuentran en el DB instancia indicado en la llamada del bloque.
OPNDI "MyDB" L #MyIn1 L #MyIn2 +I T #MyOut3	El bloque de datos global "MyDB" se escribe en el registro DI. Antes de la migración, las instrucciones direccionan variables "L" y "T" declaradas previamente en "MyDB". Tras la migración, las instrucciones direccionan variables "L" y "T" declaradas previamente en la interfaz del bloque. El registro DI no se evalúa en S7-1500 para el acceso.

Ejemplo 2

El ejemplo siguiente muestra la semántica modificada de la instrucción "LAR2":

AWL	Explicación
L P#M23.0	
LAR2	Antes de la migración la asignación a AR2 destruye la base para accesos a parámetros.
L #MyIn1	Antes de la migración no es posible acceder a "MyIn1" , o dicho acceso provoca un error. Tras la migración el acceso al parámetro formal "MyIn1" se realiza correctamente.
L IW [AR2, P#1.0]	Acceso a %IW24.0
A [AR2, P#0.4]	Acceso a %M23.4

Ejemplo 3

El siguiente ejemplo muestra cómo se puede direccionar una variable DB indirectamente en S7-1500 sin utilizar los registros de direcciones:

AWL	Explicación
OPN "MyDB" L #index LAR1 L %DBW [AR1 , P#10.0]	Antes de la migración se utiliza el direccionamiento indirecto por registro interárea. El valor variable (#index) se carga en el registro de direcciones 1. En función de ese valor se carga una palabra de datos de "MyDB" en el acumulador 1.
L "MyDB".MyArray1[#index]	Tras la migración se pueden depositar los valores de datos de "MyDB" en un ARRAY. Los diferentes elementos ARRAY pueden indexarse de forma variable a través del parámetro de entrada "#index".

4.2.4.5 Migración de programas SCL a S7-1500

Indicaciones sobre la migración de programas SCL

Representación de bloques SCL después de la migración

Los bloques SCL creados con S7-300/400 se migran automáticamente a S7-1500 si es posible interpretar su semántica de forma unívoca. Las partes del programa que no se pueden interpretar de forma unívoca se marcan con signos de interrogación, que provocan errores de sintaxis después de la migración. Compruebe la semántica de estas partes del programa después de la migración y corríjalas si es necesario.

En algunos casos la migración modifica el código lógico. Luego la sección del código correspondiente del programa de partida se adopta como sección de comentario en el bloque migrado. Así es posible entender más fácilmente los cambios realizados en la migración.

Consulte también

Principios básicos de la migración de programas PLC (Página 185)

Indicaciones sobre la migración de programas PLC (Página 189)

Direccionamiento indirecto en SCL

Introducción

En S7-1500 se han unificado las posibilidades del direccionamiento indirecto para todos los lenguajes de programación. Por ello los direccionamientos indirectos, que eran posibles en S7-300/400, se convierten parcialmente en la migración.

La siguiente tabla proporciona una vista general de las posibilidades de conversión. La migración de las distintas construcciones del lenguaje se explica a continuación más detalladamente.

Direccionamiento indirecto	S7-300/400	S7-1500
Direccionamiento indirecto de un DB	Tipo de datos "BLOCK_DB"	Tipo de datos "DB_ANY"
Direccionamiento indirecto de variables DB	#block.%DBW3	Instrucciones "PEEK"/"POKE"
	WORD_TO_BLOCK_DB(#myWord).%DBW3	Instrucciones "PEEK"/"POKE"
	#block.DW(IDX := #myInt)	Instrucciones "PEEK"/"POKE"
	WORD_TO_BLOCK_DB(#myWord).DW(IDX:=#myInt)	Instrucciones "PEEK"/"POKE"
	%DB1.DW(IDX :=#myInt)	Instrucciones "PEEK"/"POKE"
Direccionamiento indirecto de la periferia	QB(IDX :=#myInt):P	Instrucciones "PEEK"/"POKE"
Direccionamiento indirecto de variables PLC	IX(IDX :=#myInt1,Bit:=#myInt2)	Instrucciones "PEEK"/"POKE"
	QB(IDX :=#myInt)	Instrucciones "PEEK"/"POKE"
	MW(IDX :=#myInt)	Instrucciones "PEEK"/"POKE"

Migración del tipo de datos "BLOCK_DB" a "DB_ANY"

El tipo de datos "BLOCK_DB" no está disponible en S7-1500. La migración hace que las variables de este tipo de datos se conviertan al tipo de datos "DB_ANY" . Las funciones de conversión "BLOCK_DB_TO_WORD" y "WORD_TO_BLOCK_DB" se migran a "UINT_TO_WORD(DB_ANY_TO_UINT)" y "UINT_TO_DB_ANY(WORD_TO_UINT)".

Los siguientes ejemplos muestran la migración del tipo de datos "BLOCK_DB".

En la primera tabla se observan tres segmentos del programa antes de la migración:

SCL

```
#myBlock  //Tipo de datos  BLOCK_DB;
//...
#myWord :=BLOCK_DB_TO_WORD(#myBlock);
//...
#myBlockDB := WORD_TO_BLOCK_DB(#myWord);
```

En la siguiente tabla se observan los tres segmentos del programa después de la migración:

SCL

```
#myBlock  //Datentyp DB_Any;
//...
#myWord := UINT_TO_WORD(DB_ANY_TO_UINT(#myBlock));
//...
#myDBANY:= UINT_TO_DB_ANY(WORD_TO_UINT(#myWord));
```

Migración del direccionamiento indirecto de variables DB

Las indicaciones indirectas de los bloques de datos o las variables DB se implementan en S7-1500 a través de las instrucciones "PEEK"/"POKE". La migración convierte automáticamente estos direccionamientos indirectos.

Los siguiente ejemplos muestran la migración del direccionamiento indirecto de variables DB.

En la primera tabla se observan algunos segmentos del programa antes de la migración:

SCL

```
#myWord := %DB1.DW(IDX :=#myInt);
//...
#myBool := %DB1.DX(IDX :=#myByteOffset,Bit:=#myBitOffset);
//...
%DB1.DW(IDX := #myInt) := 12;
//...
```

En la siguiente tabla se observan los segmentos del programa después de la migración:

SCL

```
#myWord := PEEK_WORD(area:=16#84, dbNumber:=1, byteOffset:=#myInt);
//...
#myBool:=PEEK_BOOL(area:=16#84, dbNumber:=1, byteOffset:=#myByteOffset, bitOffset:=#myBitOffset);
//...
POKE(area:=16#84,dbNumber:=1,byteOffset:=#myInt,value:=12);
//...
```

Migración del direccionamiento indirecto de variables PLC

Las indicaciones indirectas de variables PLC se implementan en S7-1500 a través de las instrucciones "PEEK"/"POKE". La migración convierte automáticamente estos direccionamientos indirectos.

Los siguiente ejemplos muestran la migración del direccionamiento indirecto de variables DB.

En la primera tabla se observa un segmento del programa antes de la migración:

SCL

```
#myWord := MW(IDX := #myInt);
```

En la siguiente tabla se observan los segmentos del programa después de la migración:

SCL

```
#myWord := PEEK(area:=16#83,dbNumber:=0,byteOffset:=#myInt);
```

Consulte también

Direccionar operandos de modo indirecto (Página 2305)

POKE: Escribir dirección de memoria (Página 4388)

POKE_BOOL: Escribir bit de memoria (Página 4390)

PEEK: Leer dirección de memoria (Página 4384)

PEEK_BOOL: Leer bit de memoria (Página 4386)

POKE_BLK: Escribir área de memoria (Página 4392)

Instrucciones en SCL

Migración de instrucciones

S7-1500 ofrece un juego de instrucciones ligeramente modificado. Algunas instrucciones se han innovado o reemplazado por instrucciones nuevas.

Durante la migración se adoptan en lo posible las instrucciones utilizadas en el programa. En caso de que la instrucción no esté disponible en S7-1500 se intentará reemplazarla por una

instrucción compatible o semejante. Es necesario comprobar nuevamente el programa migrado y, dado el caso, adaptarlo.

De esta manera pueden darse las siguientes modificaciones:

- La instrucción nueva tiene parámetros formales adicionales. Asigne los parámetros adicionales.
- Un parámetro formal de la instrucción nueva tiene otro tipo de datos. En ese caso la migración reemplaza automáticamente una instrucción para la conversión de los tipos de datos.

Si la migración no puede garantizar de forma unívoca que la asignación de parámetros sea semánticamente idéntica al programa de partida, se comentan los parámetros en cuestión. Compruebe la semántica de estas partes del programa después de la migración y corríjalas si es necesario.

Los siguientes ejemplos muestran la migración de la instrucción "AG_SEND" a la instrucción "TSEND".

En la primera tabla se observa la llamada de "AG_SEND" antes de la migración:

SCL

```
AG_SEND (ACT:=#myBool,
         ID:=#myInt1,
         LADDR:=#myWord,
         SEND:=#myAny,
         LEN:=#myInt2,
         DONE=>#myBool,
         ERROR=>#myBool,
         STATUS=>#myWord);
```

En la segunda tabla se observa la llamada de "TSEND" después de la migración:

SCL

```
"TSEND_DB"((* ToReplace: REQ:=#myBool *)
         (* ToReplace: ID:=#myInt1 *)
         LEN:=INT_TO_UDINT(#myInt2),
         DONE=>#myBool,
         ERROR:=#myBool,
         STATUS:=#myWord,
         DATA:=#myAny;
```

El parámetro actual de "REQ" ha sido adoptado por el parámetro "ACT" de la instrucción "AG_SEND". Está comentado porque es necesario comprobar el programa en este punto.

El parámetro actual de "ID" ha sido adaptado por "AG_SEND" y también se debe comprobar.

El parámetro "LEN" tenía en "AG_SEND" otro tipo de datos. La migración agrega una conversión automáticamente.

Migración de instrucciones a expresiones

En algunos casos la migración crea varias instrucciones a partir de una instrucción, por ejemplo, cuando un parámetro de salida tiene otro tipo de datos después de la migración y el parámetro formal se debe convertir. La instrucción de conversión se agrega después de la instrucción migrada Sin embargo, de esta forma no es posible utilizar la instrucción original como expresión. En ese caso la migración agrega una variable temporal en la interfaz del bloque migrado y le asigna el resultado de la instrucción. Esta variable temporal puede utilizarse después como expresión. En expresiones complejas no siempre es posible realizar una migración automática. Las expresiones que no se pueden interpretar de forma unívoca se marcan con signos de interrogación, que provocan errores de sintaxis después de la migración. Compruebe la semántica de estas partes del programa después de la migración y corríjalas si es necesario.

Los siguientes ejemplos muestran la migración de la instrucción "READ_RTM" a la instrucción "RTM".

En la primera tabla se observa la llamada de "READ_RTM" antes de la migración:

SCL

```
IF READ_RTM(NR:=#myByte, CQ=>#myBool, CV=>#myOutInt)= 1 THEN
...;
END_IF;
```

En la segunda tabla se observa la llamada de "RTM" después de la migración.

SCL

```
(* classic code: IF READ_RTM(NR:=#myByte, CQ=>#myBool, CV=>#myOutInt)= 1 THEN*)
#SCL_MIGRA_TEMP_INT_1:=RTM(NR:=#myByte,
                           MODE:=0,
                           PV:=#Migra_PV,
                           CQ:=#myBool,
                           CV:=#Migra_CV);
#myOutInt:=DINT_TO_INT(#Migra_CV);

IF #SCL_MIGRA_TEMP_INT_1= 1 THEN;
...
END_IF;
```

La instrucción del programa de partida se adopta como sección de comentario en el bloque migrado Así es posible entender más fácilmente los cambios realizados en la migración.

Como CV en "RTM" tiene el tipo de datos DINT se agrega una instrucción de conversión que convierta el parámetro actual INT a DINT. De esa manera ya no es posible utilizar el resultado de RTM como expresión en la instrucción IF.

En la interfaz del bloque migrado se agrega la variable temporal "SCL_MIGRA_TEMP_INT_1". El resultado de la instrucción RTM se asigna a la variable temporal.

En lugar de la instrucción "RTM" se utiliza la variable temporal como expresión en la instrucción IF.

4.2.4.6 Migración de programas GRAPH a S7-1500

Indicaciones sobre la migración de programas GRAPH

Introducción

Los programas GRAPH que han sido creados con S7-300/400 pueden ser migrados automáticamente a S7-1500. Mediante la migración PLC se optimizan los programas para la ejecución en una CPU de la serie S7-1500. El protocolo de migración notifica todos los cambios automáticos realizados en el programa durante la migración y le informa sobre cambios manuales que pueden ser necesarios y que usted deberá realizar a continuación.

Adaptación del programa GRAPH mediante la migración PLC

La siguiente lista muestra algunos ejemplos de adaptaciones automáticas que se realizan durante la migración PLC. Todas las adaptaciones se describen detalladamente en los capítulos siguientes.

- Adaptación de la interfaz del bloque
  Se modifica ligeramente la interfaz de los bloques GRAPH. Así es posible usar el acceso optimizado al bloque de S7-1500 para bloques GRAPH.
- Direccionamiento simbólico homogéneo
  SIMATIC S7-1500 destaca por una programación simbólica homogénea que hace que el programa tenga un mayor rendimiento y se reduzca el peligro de errores de acceso.
  Si el programa de partida utiliza operandos para los que no se había declarado ningún nombre simbólico, la migración adjudica, cuando es posible, un nombre simbólico para el operando.
- Modelo de memoria de DB de GRAPH
  Debido a los datos de rendimiento de S7-1500, ya no hay necesidad de utilizar DB de GRAPH minimizados. Por eso, GRAPH ya no ofrece modelos de ubicación diferentes. En todos los DB de GRAPH dispone del juego de parámetros completo y, por tanto, de toda la funcionalidad.
  Durante la migración se amplían los DB minimizados al juego de parámetros completo.

Nuevas funciones de S7-1500

Tras la migración de un programa GRAPH puede aprovechar por completo la funcionalidad de S7-1500. La siguiente lista muestra algunos ejemplos de las nuevas funciones:

- Acceso optimizado al bloque
  Con un acceso optimizado al bloque, los elementos de datos declarados se disponen automáticamente en el área de memoria disponible del bloque de manera que se aproveche al máximo la capacidad de éste. Los datos se estructuran y depositan de una manera óptima para la CPU utilizada. Esto se traduce en un mayor rendimiento de la CPU. Se excluyen los errores de acceso, p. ej. desde la HMI.
  El atributo "acceso optimizado al bloque" siempre está activado para bloques GRAPH en S7-1500 y no puede anularse la selección.
  Consulte también: Bloques con acceso optimizado (Página 2258)
- Nuevas instrucciones
  En la Task Card "Instrucciones" está disponible el juego de instrucciones completo de S7-1500. Se han desarrollado o innovado numerosas instrucciones. Así, por ejemplo, encontrará en la paleta "Instrucciones básicas"·de la carpeta "Control del programa" la instrucción "GET_ERROR" para programar un tratamiento de errores local en el bloque.
  Consulte también: GET_ERROR: Consultar error localmente (Página 4616)
- Funciones KOP/FUP
  En GRAPH para S7-300/400 solo existía la posibilidad de programar una operación lógica previa en la primera entrada booleana de un cuadro. En S7-1500 se puede hacer una operación lógica previa de todas las entradas booleanas.
  La salida de habilitación ENO puede activarse o desactivarse selectivamente para instrucciones individuales.
  Consulte también: Activar y desactivar el mecanismo EN/ENO (Página 2699)
- Nuevos tipos de datos
  S7-1500 ofrece un juego avanzado de tipos de datos y nuevas posibilidades para la conversión de tipos de datos.
  Consulte también: Vista general de los tipos de datos válidos (Página 2321)
- Constantes
  S7-1500 ofrece posibilidades adicionales para la notación de constantes que también le permiten asignar tipos de datos a las constantes.
  Consulte también: Auto-Hotspot

Modificaciones en el programa GRAPH

Introducción

La migración PLC adapta automáticamente en lo posible el programa GRAPH a la nueva familia de CPU. Se analizan tanto las acciones como los segmentos KOP/FUP que usted ha empleado para la programación de condiciones. Las estructuras del programa que ya no se admiten se actualizan. Las instrucciones que no están disponibles en la nueva CPU se reemplazan por las instrucciones correspondientes de S7-1500.

Direccionamiento simbólico

SIMATIC S7-1500 destaca por una programación simbólica homogénea que hace que el programa tenga un mayor rendimiento y se reduzca el peligro de errores de acceso.

Cuando el programa de partida utiliza operandos absolutos para los que no hay ningún nombre simbólico declarado, el operando absoluto se marca como no válido. Entonces debe declarar un nombre simbólico antes de poder compilar el programa migrado.

Si el programa de partida utiliza operandos absolutos para los que hay un nombre simbólico declarado, la migración emplea el nombre simbólico en lugar del operando absoluto. Los accesos a los distintos bits dentro de un operando simbólico declarado son reemplazados por accesos Slice.

El siguiente ejemplo muestra un acceso absoluto antes de la migración:

```
%DB1.DBx1.1
```

A continuación puede ver cómo reemplaza la migración el acceso absoluto. Como en "DataBlock1" se había declarado la variable "FirstDataByte" para el primer byte de datos, el acceso al primer bit dentro de esta variable puede reemplazarse por un acceso slice.

```
"DataBlock1".FirstDataByte.x1
```

Direccionamiento totalmente especificado

Los programas S7-300/400 utilizan a menudo direcciones parcialmente especificadas para acceder de forma indirecta a operandos. Solo se indica la dirección de una variable DB dentro de un bloque de datos. La información sobre el nombre de DB o el número de DB no está incluida en la dirección (p. ej. `%DBX0.2`). Durante el tiempo de ejecución, el programa accede al bloque de datos que está almacenado actualmente en el registro DB. S7-1500 no soporta el direccionamiento parcialmente especificado. Para direccionar variables DB de forma indirecta, utilice parámetros de bloque del tipo de datos "DB_ANY".

Si ha utilizado direcciones totalmente especificadas en programas GRAPH, estas se notificarán como error tras la migración. El bloque no puede compilarse. En ese caso, sustituya la dirección parcialmente especificada por una completamente especificada.

Elementos KOP/FUP en condiciones

Los segmentos KOP o FUP en transiciones, interlock, supervisión e instrucciones permanentes son migrados según las mismas reglas que los segmentos en tipos de bloque KOP o FUP.

Consulte también: Migración de programas KOP/FUP a S7-1500 (Página 218)

Acciones GRAPH

SIMATIC S7-1500 ofrece un juego de instrucciones ligeramente modificado. Algunas instrucciones se han innovado o reemplazado por instrucciones nuevas.

Si ha utilizado instrucciones en acciones GRAPH, la migración comprueba si S7-1500 sigue soportándolas. Las instrucciones se adoptan en lo posible. En caso de que la instrucción no

esté disponible en S7-1500 se intentará reemplazarla por una instrucción compatible o semejante.

Un ejemplo de ello son las instrucciones "RLDA: Rotar a la izquierda vía bit de estado A1" y "RRDA: Rotar a la derecha vía bit de estado A1". Son reemplazadas por las instrucciones "ROR: Rotar a la derecha" y "ROL: Rotar a la izquierda".

Si las instrucciones son reemplazadas por la migración, recibirá una indicación en el protocolo de migración. Es necesario comprobar nuevamente el programa migrado y, dado el caso, adaptarlo.

Consulte también: Migración de instrucciones a S7-1500 (Página 195)

Adaptación de la interfaz de bloques GRAPH

Cambio de la interfaz del bloque durante la migración

La interfaz de los bloques GRAPH se adapta durante la migración. Mediante las adaptaciones, puede usar el concepto de memoria optimizado de S7-1500 con GRAPH. El acceso optimizado al bloque siempre está activo para los bloques GRAPH en S7-1500 y no puede anularse la selección. Se realizan cambios en las siguientes áreas:

- Parámetros estáticos: "STATIC"
- Parámetros de salida: "OUTPUT"

Consulte también: Bloques con acceso optimizado (Página 2258)

Si ha utilizado en el programa de partida nombres de parámetros que están reservados en S7-1500, estos cambiarán de nombre durante la migración.

Parámetros estáticos

Durante la migración, los parámetros estáticos se adaptan como sigue:

- La interfaz contiene una estructura para cada paso y cada transición del FB de GRAPH en la sección "Static". En S7-300/400 se han utilizado los tipos de datos "GraphStep", "GraphStepMin", "GraphTransition" y "GraphTransitionMin". En S7-1500 se introducen los nuevos tipos de datos "G7_StepPlus" y "G7_TransitionPlus" para la reproducción de las estructuras de pasos y de transición.
  La migración se desarrolla del siguiente modo:
  - Los parámetros "SNO" y "TNO" se adoptan con sus valores predeterminados a partir de sus antiguas estructuras; el resto de valores predeterminados no se transfiere. Los programas GRAPH con DB minimizado no contienen los parámetros "SNO" y "TNO". Si migra un DB minimizado, la migración asigna automáticamente valores predeterminados que empiezan por "1" a esos parámetros.
  - Las ubicaciones de los parámetros modificados en el programa GRAPH no se adaptan automáticamente, sino que se marcan como errores. Debe adaptar manualmente las ubicaciones tras la migración.
- El resto de parámetros estáticos se transfiere a la estructura "RT_DATA". Los valores predeterminados se adoptan en lo posible y las ubicaciones se adaptan automáticamente en el programa.

Parámetros de salida

Como S7-1500 ya no soporta el análisis de criterios, se eliminan los siguientes parámetros de salida durante la migración:

- T_CRIT
- T_CRITOLD
- T_CRITFLT
- S_CRITLOC
- S_CRITLOCERR
- S_CRITSUP

Las ubicaciones del programa GRAPH no se adaptan automáticamente, sino que se marcan como errores. Debe adaptar manualmente las ubicaciones tras la migración.

Consulte también

Palabras clave (Página 2280)

4.3 Migración de S7-1200 a firmware V4 y posterior

4.3.1 Principios básicos de la migración a V4

Introducción

Si en el proyecto ha utilizado una CPU con la versión de firmware V3 y desea migrar a una CPU con una versión de firmware V4.0 o posterior, puede sustituir los dispositivos fácilmente.

El TIA Portal ofrece para ello la función "Sustituir dispositivo". Al sustituir el dispositivo el proyecto se conserva intacto. Los programas que se han creado con la versión de firmware V3 pueden continuar utilizándose.

Reglas

Para la sustitución de dispositivos se aplican las siguientes reglas fundamentales:

- La sustitución de dispositivos solo es posible si la CPU utilizada hasta el momento corresponde a la versión de firmware V3.0. Las CPUs S7-1200 con la versión de firmware V1.0 o V2.0 primero deben sustituirse por una CPU V3.0.
- La sustitución de una CPU V4 por una CPU V3 no es posible. Si desea continuar utilizando la CPU V3 utilizada hasta el momento, cree una copia de la CPU antes de sustituir el dispositivo.
- El programa no puede transferirse a una CPU nueva a través de una Memory Card. Utilice en lugar de eso la función "Sustituir dispositivo", que se describe en los capítulos siguientes.

HMI Panels

En la sustitución de dispositivos, los HMI Panels configurados se tratan de forma diferente en función de su versión de firmware y del tipo de comunicación.

La siguiente tabla muestra qué conexiones HMI se soportan en la migración:

Versión de firmware del panel	Comunicación PUT/GET	Migración a V4
V11 o anterior	No	La S7-1200 no soporta esta configuración. Actualice el firmware del HMI Panel a V12.0. Seguidamente, compile y cargue la configuración.
V11 o anterior	Sí	La S7-1200 soporta esta configuración. La conexión se establece automáticamente mientras el proyecto se compila y carga tras la sustitución del dispositivo.
V12 o superior	No	La S7-1200 soporta esta configuración. La conexión se establece automáticamente mientras el proyecto se compila y carga tras la sustitución del dispositivo.

Nota

HMI TP 177B 4"

El panel HMI TP 177B 4" con versión de firmware V11.0.2 no puede utilizarse con S7-1200 V4. En caso necesario, reemplace el panel por un dispositivo más nuevo.

Módulos de ampliación S7-1200

Si ya utiliza los siguientes módulos S7-1200 en su instalación de forma centralizada, deberá actualizar el firmware de dichos módulos para garantizar el funcionamiento con S7-1200 V4.

- Maestro ASi - CM 1243
- Maestro DP - CM 1243-5
- CP WAN - CP1243-1

Los módulos S7-1200 nuevos ya están equipados de fábrica con el firmware más actual.

Bloques protegidos

Los bloques que disponen de protección de know-how o contra copia no pueden migrarse a V4. Si el proyecto contiene bloques protegidos, antes de realizar el cambio hay que desactivar la protección.

Si se trata de bloques suministrados cuya contraseña desconoce, solicite al proveedor la contraseña o bien un bloque compatible con V4.

⚠ ADVERTENCIA

Evitar daños personales y materiales

En algunos casos se realizan cambios en el programa durante la sustitución de dispositivos. Por ese motivo, tras la sustitución de dispositivos debe comprobar detalladamente el programa en un entorno de prueba antes de ponerlo en marcha.

Nota

Soporte adicional

En Siemens Industry Online Support encontrará FAQs actuales sobre la migración a S7-1200 V4 (http://support.automation.siemens.com/WW/view/es/82140966).

Si necesita soporte adicional para la migración a S7-1200 V4, diríjase al SIMATIC Customer Support.

Consulte también

Realizar una migración a V4 (Página 246)

Particularidades tras la migración a V4 (Página 247)

4.3.2 Realizar una migración a V4

Requisitos

- El proyecto contiene una CPU con la versión de firmware V3.
- El proyecto no contiene bloques protegidos.

Procedimiento

Para sustituir una CPU, proceda del siguiente modo:

1. Seleccione la CPU V3 que desea sustituir.
2. En el menú contextual, elija el comando "Sustituir dispositivo".
   Se abre el cuadro de diálogo "Sustituir dispositivo".
3. En "Dispositivo nuevo" de la estructura de árbol seleccione la CPU V4 por la que desea cambiar la CPU V3 actual.
4. Haga clic en "Aceptar".
   La CPU existente se sustituye por la nueva.

5. Seleccione la CPU nueva y elija el comando "Compilar > Hardware y software (solo cambios)" del menú contextual.
   La configuración del dispositivo y el programa de usuario se vuelven a compilar.
6. Opcionalmente: en caso necesario, active la protección de know-how o contra copia para diferentes bloques del programa.
7. Seleccione la CPU nueva y elija el comando "Cargar en dispositivo > Hardware y software (solo cambios)" del menú contextual.
   La configuración del dispositivo y el programa de usuario se cargan en la CPU nueva.
   Con ello ha concluido la sustitución de dispositivos.

ADVERTENCIA

Evitar daños personales y materiales

En algunos casos se realizan cambios en el programa durante la sustitución de dispositivos. Por ese motivo, tras la sustitución de dispositivos debe comprobar detalladamente el programa en un entorno de prueba antes de ponerlo en marcha.

Nota

Soporte adicional

En Siemens Industry Online Support encontrará FAQs actuales sobre la migración a S7-1200 V4 (http://support.automation.siemens.com/WW/view/es/82140966).

Si necesita soporte adicional para la migración a S7-1200 V4, diríjase al SIMATIC Customer Support.

Consulte también

Principios básicos de la migración a V4 (Página 244)

Particularidades tras la migración a V4 (Página 247)

4.3.3 Particularidades tras la migración a V4

Cambios de función de la V4

S7-1200 V4 ofrece una gama de funciones notablemente mayor. Los principales cambios en las funciones que deben considerarse tras la migración de V3 a V4 se resumen a continuación.

Encontrará más información sobre la S7-1200 en el "Manual de sistema SIMATIC S7 Controlador programable S7-1200".

Consulte también:

TIA Portal en Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/es/65601780)

Bloques de organización

Con S7-1200 V4 se puede ajustar de forma selectiva la posibilidad de interrupción de cada bloque de organización utilizado. En la sustitución de dispositivos todos los bloques de organización se configuran de forma que no pueden interrumpirse para garantizar que el programa V3 permanece inalterado y funcional. También las prioridades de los OB se adoptan del programa V3 sin modificaciones. En caso necesario, tras la migración es posible cambiar los ajustes de prioridad y de posibilidad de interrupción.

En V4 el comportamiento de las alarmas de diagnóstico se ha modificado del siguiente modo:

En V3 la información de arranque contenía siempre datos sobre el módulo causante, incluido el número de canal. En V4 estos dos datos solo se indican con un evento de diagnóstico pendiente. Si no hay ningún evento de diagnóstico pendiente, p. ej. porque el fallo ya se ha solucionado, solo se indica el módulo causante.

Niveles de acceso

S7-1200 V4 ofrece un concepto ampliado de niveles de acceso. La tabla siguiente muestra cómo se reflejan los niveles de protección del firmware V3 en V4:

Nivel de protección V3	Nivel de acceso V4
Sin protección	Acceso completo (sin protección)
Protección contra escritura	Acceso de lectura
Protección contra escritura/lectura	Acceso HMI

Con el firmware V4 se ha incorporado el nivel de acceso "Sin acceso (protección completa)". En las versiones anteriores no existía.

Librerías de instrucciones

Tras la migración a S7-1200 V4 continúan disponibles las instrucciones de las librerías de la versión de firmware V3. De ese modo se garantiza que el programa pueda seguir utilizándose sin cambios. La S7-1200 V4 ofrece asimismo numerosas instrucciones nuevas que también son compatibles con las instrucciones de la S7-1500.

Encontrará más información sobre las librerías de instrucciones de la S7-1200 en el "Manual de sistema SIMATIC S7 Controlador programable S7-1200".

Consulte también:

TIA Portal en Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/es/65601780)

Motion Control

En la sustitución de dispositivos los objetos de Motion Control de las librerías de las versiones de firmware V1 y V2 se sustituyen por los objetos equivalentes de las librerías V3. Los objetos de las librerías V3 son compatibles, por lo que los programas pueden continuar utilizándose sin cambios.

Las librerías de la S7-1200 V4 ofrecen numerosas funciones nuevas para Motion Control, que también son compatibles con las funciones de la S7-1500. Si desea utilizar las librerías V4, selecciónelas tras la sustitución de dispositivos en la Task Card "Instrucciones".

Encontrará más información sobre las nuevas funciones de Motion Control en el "Manual de sistema SIMATIC S7 Controlador programable S7-1200".

Consulte también:

TIA Portal en Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/es/65601780)

Servidor web

Los siguientes ajustes sobre el funcionamiento a través de servidor web se transfieren de la CPU V3 a la CPU V4 con la sustitución de dispositivos:

- Activar servidor web en este módulo
- Permitir el acceso solo vía HTTPS

Para hacer funcionar la CPU V4 a través de un servidor web hay que configurar nodos de usuarios con derechos asignados a través de la administración de usuarios. Los usuarios estándar sin derechos adicionales solo tienen disponibles las páginas web estándar.

Nota

Soporte adicional

En Siemens Industry Online Support encontrará FAQs actuales sobre la migración a S7-1200 V4 (http://support.automation.siemens.com/WW/view/es/82140966).

Si necesita soporte adicional para la migración a S7-1200 V4, diríjase al SIMATIC Customer Support.

Comunicación mediante PUT/GET

Tras la sustitución de dispositivos está activada la comunicación mediante PUT/GET. Recuerde que los nuevos tipos de conexión integrados ofrecen un estándar de seguridad superior al de la comunicación PUT/GET. Si la comunicación PUT/GET no se utiliza, conviene desactivarla.

Consulte también

Principios básicos de la migración a V4 (Página 244)

Realizar una migración a V4 (Página 246)

4.4 Recomendaciones de programación

4.4.1 Resumen de las nuevas funciones de la CPU S7-1500

Mayor rendimiento

La familia de CPUs S7-1500 ofrece un rendimiento considerablemente mayor que las CPUs de las series S7-300/400. A la hora de realizar la programación con STEP 7 V5.x probablemente estaba acostumbrado a trabajar con determinadas formas de programación (p. ej. el direccionamiento absoluto) con el fin de conseguir un mayor rendimiento de la CPU y un código del programa más reducido.

Dichas formas de programación han quedado obsoletas debido al gran rendimiento que presenta la S7-1500.

En los apartados siguientes le presentamos algunas posibilidades de programación nuevas para la S7-1500.

Simbología homogénea

La S7-1500 ofrece la posibilidad de utilizar una simbología homogénea en todo el proyecto. La función de autocompletar ayuda al usuario en función del contexto a la hora de programar con símbolos dentro de los editores de programación. En la declaración se asigna a los elementos de datos (p. ej. dentro de un bloque de datos) un nombre simbólico, pero no una dirección fija dentro del bloque de datos. Esto permite aprovechar al máximo el gran rendimiento de la S7-1500 cuando se accede a dichos elementos de datos. Las direcciones absolutas de los operandos ya no tienen que conocerse, con lo que se evitan errores de acceso.

El código del programa se hace más comprensible gracias a los símbolos y se reduce la necesidad de introducir comentarios. Cuando se corrige la simbología se actualizan automáticamente todas las ubicaciones.

Encontrará un ejemplo del uso de la simbología homogénea en: Direccionamiento simbólico (Página 253)

Acceso optimizado al bloque

Con un acceso optimizado al bloque, los elementos de datos declarados se disponen automáticamente en el área de memoria disponible del bloque de manera que se aproveche al máximo la capacidad de este último. Los datos se estructuran y depositan de una manera óptima para la CPU utilizada. Del almacenamiento se encarga el sistema. En la declaración se asigna a los elementos de datos un nombre simbólico con el que se llama la variable dentro

del bloque. Esto se traduce en un mayor rendimiento de la CPU. Se excluyen los errores de acceso, p. ej. desde la HMI.

Accesos a bloques en la comparación
Estándar < > Optimizado

SIEMENS

	Acceso estándar al bloque (S7-1200/1500 compatible con S7-300/400)	Acceso optimizado al bloque (solo S7-1200/1500)
Gestión de datos	Las variables se pueden direccionar tanto de forma simbólica (con optimización de memoria) como absoluta (definida por el usuario).	El sistema se encarga de gestionar y optimizar el almacenamiento de datos. De ese modo se aprovecha al máximo la capacidad de memoria.
Rendimiento	El acceso a una CPU de las series S7-1200/1500 no siempre es el más rápido posible, ya que el almacenamiento de datos puede no ser efectivo debido al direccionamiento absoluto.	El acceso siempre es el más rápido posible, ya que el sistema optimiza el almacenamiento de datos y no se asignan direcciones fijas.
Frecuencia de errores	El direccionamiento absoluto (p. ej. desde HMI o con direccionamiento indirecto) puede provocar incoherencias si se cambia la dirección fija.	Se excluyen los errores de acceso, p. ej. en caso de direccionamiento indirecto o desde la HMI, ya que se accede simbólicamente.
Remanencia	Válido para todas las variables de un bloque de datos	Válido para variables individuales

Recomendación:
Para conseguir el mejor rendimiento posible se recomienda no mezclar los tipos de acceso a bloques dentro del programa.

Encontrará más información sobre bloques con acceso optimizado en "Consulte también".

Nuevos tipos de datos

Los nuevos tipos de datos LWORD, LINT, ULINT, LTIME, LTOD, LDT y Array (límite de 32 bits) ofrecen una precisión de cálculo considerablemente superior cuando se utilizan funciones matemáticas. Las posibilidades en cuanto a la conversión implícita y explícita de datos son mayores que las de las CPUs de las series S7-300/400.

Encontrará más información sobre los nuevos tipos de datos en "Consulte también".

Tipos de datos PLC

Los tipos de datos PLC (UDT) son estructuras de datos definidas por el usuario que pueden utilizarse varias veces en el programa. La estructura de un tipo de datos PLC consta de varios componentes, que pueden pertenecer a diferentes tipos de datos. El tipo de los componentes se define en la declaración del tipo de datos PLC.

El tipo de datos PLC puede utilizarse como tipo de datos básico para la definición de variables y como plantilla para crear bloques de datos globales. Si se realizan a posteriori cambios en el tipo de datos PLC, estos se aplican automáticamente a todas las ubicaciones.

También es posible acceder simbólicamente a diferentes elementos de un array dentro de un tipo de datos PLC.

Encontrará más información sobre los nuevos tipos de datos en "Consulte también".

Encontrará un ejemplo del uso de tipos de datos PLC en: Uso de tipos de datos PLC (UDT) (Página 262)

Instrucciones unitarias en todos los lenguajes de programación

Está disponible un juego unitario de instrucciones en todos los lenguajes de programación (KOP, FUP, AWL, SCL y GRAPH).

Accesos Slice

Los accesos Slice permiten direccionar áreas de forma selectiva dentro de variables declaradas. El acceso simbólico a un solo bit es posible hasta el nivel de variable. El bit en concreto se llama de forma absoluta.

Encontrará más información sobre los accesos Slice en "Consulte también".

Direccionamiento indirecto

El direccionamiento indirecto ofrece la posibilidad de direccionar operandos cuya dirección no se calcula hasta el momento de la ejecución. En todos los lenguajes de programación hay disponibles posibilidades generales, como el direccionamiento indirecto mediante POINTER, por ejemplo. En el lenguaje de programación SCL, además, es posible utilizar también las instrucciones PEEK y POKE.

Encontrará más información sobre el direccionamiento indirecto en "Consulte también".

Otras recomendaciones de programación

Encontrará más información sobre recomendaciones de programación y una guía de programación en el Siemens Industry Online Support en:

- FAQ: Recomendaciones de programación (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=0&lang=es&referer=%2fWW%2f&func=cslib.csinfo&siteid=csius&Datakey=47071380&extranet=standard&groupid=4000002&viewreg=WW&nodeid0=29156492&objaction=csopen)
- Contexto y descripciones del sistema en la Guía de programación (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=100&lang=es&referer=%2fWW%2f&func=cslib.cssearch&nodeid0=4000024&viewreg=WW&siteid=csius&extranet=standard&groupid=4000002&objaction=cssearch&content=adsearch%2Fadsearch%2Easpx)

Consulte también

Bloques con acceso optimizado (Página 2258)

Tipos de datos (Página 2321)

Tipos de datos PLC (Página 2367)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Direccionamiento indirecto (Página 2305)

4.4.2 Direccionamiento simbólico

Ventajas del direccionamiento simbólico

El uso homogéneo de símbolos claros en todo el proyecto hace que el código del programa sea más comprensible y fácil de leer.

Las ventajas resultantes son las siguientes:

- Ya no es necesario escribir comentarios amplios.
- El acceso a datos es más rápido.
- No se producen errores al acceder a datos.
- Ya no es necesario trabajar con direcciones absolutas.
- STEP 7 supervisa la asignación del símbolo al área de memoria, es decir, cuando cambia el nombre o la dirección de una variable se actualizan automáticamente todas las ubicaciones.

Programación en STEP 7 V5.x

En STEP 7 V5.x ya existía la posibilidad de escribir el programa de forma clara utilizando nombres expresivos para operandos y bloques. Para ello se asignaban los operandos simbólicos a las áreas de memoria y los bloques en la tabla de símbolos. Para que un cambio en la simbología repercutiera también en el código del programa del editor de programación, había que determinar mediante la propiedad "Operandos preferentes" si debía ser determinante el símbolo o el valor absoluto.

El direccionamiento simbólico permitía mejorar la claridad del programa. No obstante, en algunos casos, p. ej. al programar con tipos de datos definidos por el usuario (UDT), el direccionamiento simbólico podía afectar al rendimiento.

Para aumentar el rendimiento había que ignorar la simbología en UDT y direccionar de forma absoluta. Pero para ello era necesario conocer la ubicación de los datos. Los cambios en UDT no se aplicaban automáticamente. Por medio del direccionamiento absoluto también era posible acceder a partes de una variable y editarlas. Sin embargo, el inconveniente de un direccionamiento absoluto exclusivo era que el código del programa se volvía confuso a partir de cierto volumen y había que introducir comentarios adicionales para orientarse mejor.

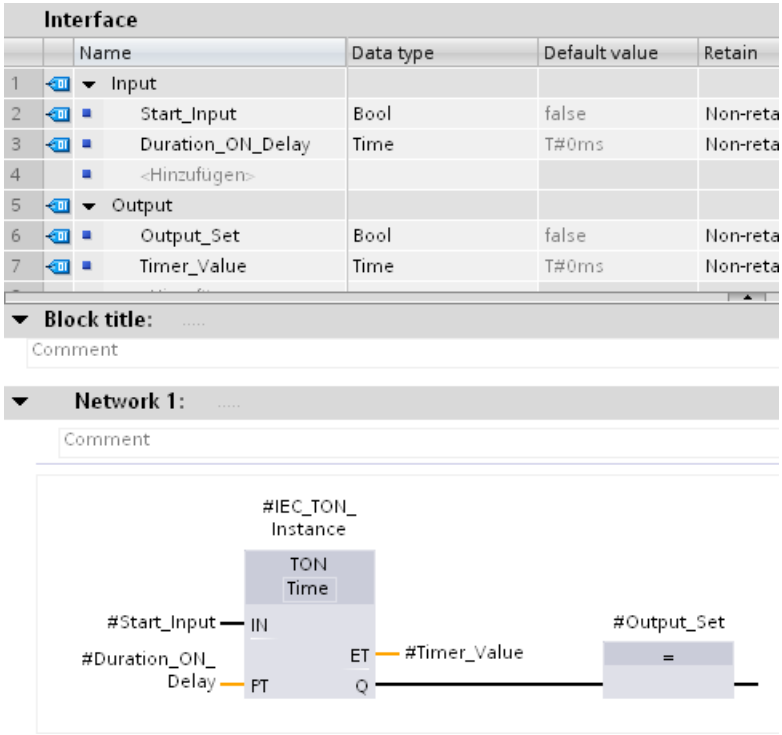
Procedimiento en STEP 7 TIA Portal

La CPU S7-1500 ofrece un rendimiento considerablemente mayor que las CPUs S7-300/400. Para poder aprovechar al máximo este gran rendimiento, recomendamos activar el acceso optimizado para todos los bloques y direccionarlo simbólicamente en el código del programa.

El editor de programación ayuda al usuario a trabajar con símbolos por medio de ayudas contextuales a las entradas, como la función de autocompletar. De ese modo es posible acceder cómodamente a variables o instrucciones existentes durante la programación.

Ejemplo de programación

El ejemplo siguiente muestra la forma de acceder simbólicamente a diferentes elementos:



Los nombres de variables definidos en la interfaz del bloque pueden utilizarse directamente en los parámetros de la instrucción TON sin conocer la dirección absoluta de las variables.

Consulte también

Ajustar el acceso a un bloque (Página 2260)

Fundamentos del acceso a bloques (Página 2258)

Utilizar la función automática de autocompletar (Página 2637)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Uso de instrucciones VARIANT (Página 269)

4.4.3 Uso de temporizadores y contadores CEI

Ventajas de los temporizadores y contadores CEI

Gracias al uso homogéneo de temporizadores y contadores CEI, el código del programa se vuelve más efectivo.

Las ventajas resultantes son las siguientes:

- Los bloques pueden llamarse varias veces con bloques de datos de instancia recién generados.
- Los contadores CEI disponen de un gran rango de contaje.
- Los temporizadores CEI presentan un mayor rendimiento y una mayor precisión de tiempo que los temporizadores S5.

Programación en STEP 7 V5.x

Los temporizadores y contadores S5 en STEP 7 V5.x se direccionaban de forma absoluta utilizando un número. Debido a la dependencia del número no era posible utilizar varias veces bloques de programa con temporizadores y contadores S5.

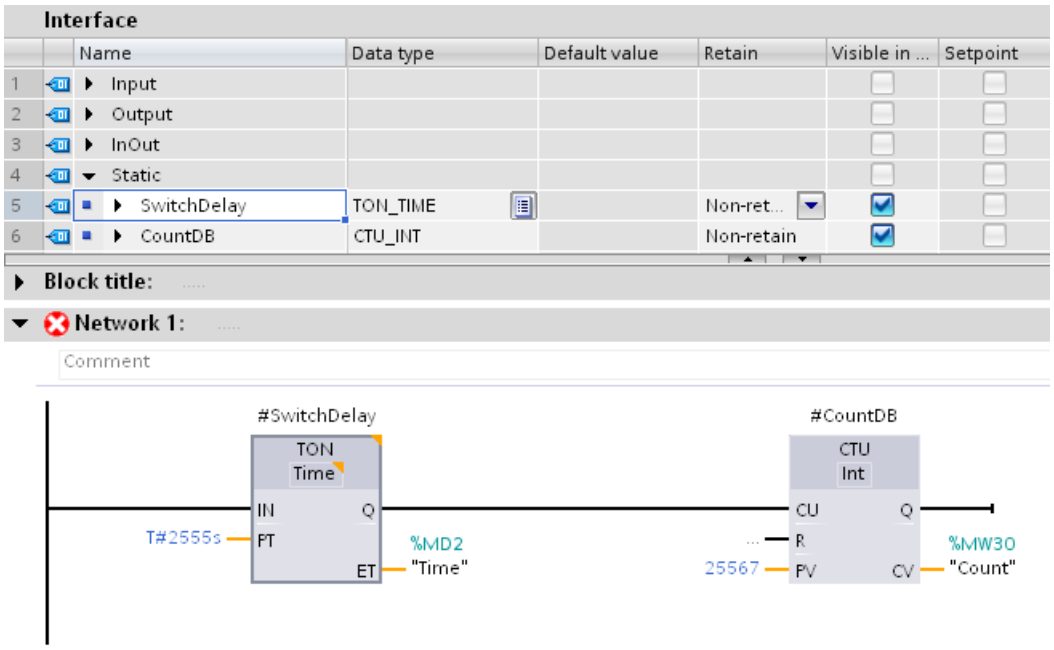
El rango de valores de un temporizador estaba limitado a 9990s como máximo y el de un contador a un valor límite máximo de 999.

Procedimiento en STEP 7 TIA Portal

Los temporizadores y contadores CEI deben declararse en el bloque de programa en el que se llamarán y utilizarán. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, o p. ej. TON_TIME y TON_LTIME, que también puede declararse como variable local en un bloque. El contador IEC es una estructura del tipo de datos IEC_SCOUNTER, IEC_USCOUNTER, etc.

Ejemplo de programación en el TIA Portal

El ejemplo siguiente muestra la forma de declarar un temporizador CEI y un contador CEI como variable local:



Los datos del temporizador CEI TON y del contador CEI CTU se depositan en la interfaz del bloque como variable local (multiinstancia).

Consulte también

Temporizadores (Página 2341)

Temporizadores CEI (Página 3306)

4.4.4 Uso flexible de la salida de habilitación ENO

Ventajas

En algunas instrucciones y llamadas de bloque existe la posibilidad de detectar errores en el tiempo de ejecución por medio del mecanismo EN/ENO e impedir que se interrumpa el programa. Así, por ejemplo, en las funciones matemáticas los desbordamientos se notifican mediante la salida de habilitación ENO.

En STEP 7 TIA Portal, la salida de habilitación ENO está desactivada por defecto para los lenguajes de programación KOP y FUP. En caso necesario, la salida de habilitación puede activarse y, de esta forma, es posible controlar de forma selectiva las instrucciones en las que deben evaluarse los errores.

Las ventajas resultantes son las siguientes:

- Si ENO está desactivada aumenta el rendimiento.
- Si ENO está activada los errores en tiempo de ejecución no provocan un STOP de la CPU.

Procedimiento en STEP 7 TIA Portal

Para activar el mecanismo EN/ENO de una instrucción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el programa sobre la instrucción para la que desea activar el mecanismo EN/ENO.
2. Elija el comando "Generar ENO" del menú contextual.
   Para esa instrucción se genera el valor ENO. Cualquier otra instrucción añadida se insertará con salida de habilitación.

En las tablas siguientes se listan las instrucciones en las que es posible desactivar la salida de habilitación ENO:

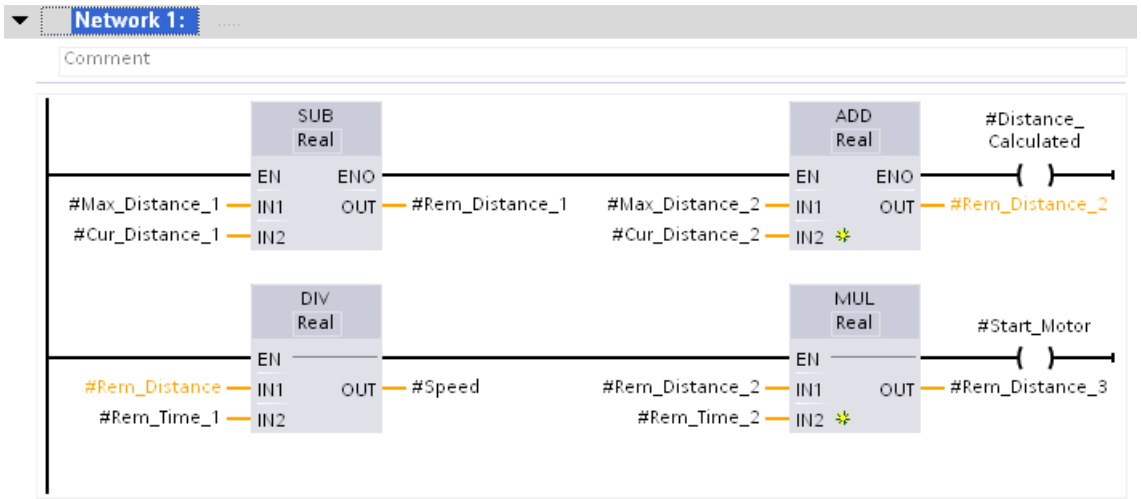
Instrucciones básicas	
Funciones matemáticas	ADD, SUB, MUL, DIV, MOD, INC, DEC, ABS, NEG, SQR, SQRT, LN, EXP, SIN, COS, TAN, ASIN, ACOS, ATAN, FRAC, EXPT, MIN, MAX, LIMIT, CALCULATE
Transferencia	MOVE, SWAP, MOVE_BLK, UMOVE_BLK, FILL_BLK, UFILL_BLK, MOVE_BLK_VARIANT
Conversión	CONVERT, ROUND, CEIL, TRUNC, FLOOR, NORM_X, SCALE_X
Operaciones lógicas con palabras	AND, OR, XOR, INV, DECO, ENCO
Desplazamiento y rotación	SHR, SHL, ROR, ROL

Instrucciones avanzadas	
String + Char	CONCAT, LEFT, RIGHT, MID, DELETE, INSERT, REPLACE, FIND, LEN, S_CONV
Fecha y hora	T_CONV

Encontrará más información sobre el mecanismo EN/ENO en los diferentes lenguajes de programación en "Consulte también".

Ejemplo de programación

El ejemplo siguiente muestra la forma de utilizar instrucciones con la salida de habilitación ENO activada y desactivada:



Si se ha activado la salida de habilitación ENO, como en la instrucción SUB, todas las instrucciones posteriores se crearán también con la salida de habilitación activada. En ese caso, si se produce un error de aritmética durante el procesamiento de la instrucción SUB, la instrucción ADD no se ejecutará.

En la segunda rama de la instrucción DIV la salida de habilitación ENO está desactivada. Si durante el procesamiento se produce un error en el tiempo de ejecución, la instrucción MUL se ejecutará de todas formas.

Consulte también

Activar y desactivar el mecanismo EN/ENO (Página 2699)

Principios básicos del mecanismo EN/ENO (Página 2539)

4.4.5 Uso de instrucciones MOVE en AWL

Posibilidades de uso

Ahora, en una CPU S7-1500 también es posible programar instrucciones MOVE en AWL.

Las ventajas resultantes son las siguientes:

- Es posible crear una estructura de programa más sencilla.
- El rendimiento de la CPU aumenta.

Programación en STEP 7 V5.x

En STEP 7 V5.x, para realizar las funciones MOVE se utilizaban las funciones de sistema "BLKMOV": Copiar área" y "UBLKMOV: Copiar áreas sin interrupciones".

Procedimiento en STEP 7 TIA Portal

En STEP 7 TIA Portal están disponibles las siguientes instrucciones MOVE nuevas:

- MOVE: Copiar valor
- MOVE_BLK: Copiar área
- MOVE_BLK_VARIANT: Copiar área
- UMOVE_BLK: Copiar área sin interrupciones

Encontrará más información sobre las nuevas instrucciones MOVE en "Consulte también".

Ejemplo de programación

El siguiente ejemplo muestra el uso de la instrucción "MOVE_BLK: Copiar área". Para ello se copia un área de array en otra:

Data_DB

	Name	Data type
1	Static	
2	Array_1	Array [0..10] of Int
3	Array_1[0]	Int
4	Array_1[1]	Int
5	Array_1[2]	Int
6	Array_1[3]	Int
7	Array_1[4]	Int
8	Array_1[5]	Int
9	Array_1[6]	Int
10	Array_1[7]	Int
11	Array_1[8]	Int
12	Array_1[9]	Int
13	Array_1[10]	Int
14	Array_2	Array [0..10] of Int
15	Array_2[0]	Int
16	Array_2[1]	Int
17	Array_2[2]	Int
18	Array_2[3]	Int
19	Array_2[4]	Int
20	Array_2[5]	Int
21	Array_2[6]	Int
22	Array_2[7]	Int
23	Array_2[8]	Int
24	Array_2[9]	Int
25	Array_2[10]	Int

```
Network 1:
Comment
1   CALL  MOVE_BLK
2        Int  UInt
3        IN    :="Data_DB".Array_1[0]
4        COUNT :=10
5        OUT   :="Data_DB".Array_2[0]
6
7
```

La instrucción MOVE_BLK permite copiar 10 elementos del "Array_1" del bloque de datos "Data_DB" en el "Array_2" del mismo bloque de datos.

Consulte también

MOVE: Copiar valor (Página 3940)

MOVE_BLK: Copiar área (Página 3951)

MOVE_BLK_VARIANT: Copiar área (Página 3953)

UMOVE_BLK: Copiar área sin interrupciones (Página 3958)

4.4.6 Realizar un acceso de array con índice variable

Ventajas del índice variable

Como índices para direccionar los componentes de un array no solo se pueden especificar constantes, sino también variables del tipo de datos Entero. Se admiten enteros con una longitud de hasta 32 bits. Si se usan variables, el índice se calcula en tiempo de ejecución. Así se puede utilizar un índice diferente para cada ejecución, p. ej. en bucles de programa. También es posible acceder a un array dentro de un tipo de datos PLC.

Las ventajas resultantes son las siguientes:

- No se requiere direccionamiento con registros de direcciones ni con punteros creados por el usuario, como p. ej. un puntero ANY.
- Más flexibilidad dentro del programa.
- El índice variable está disponible en todos los lenguajes de programación de STEP 7.
- Se utilizan los nombres existentes para los bloques de datos y las variables de array. De este modo se aumenta la legibilidad del código del programa.
- La dirección básica del array no tiene que conocerse.
- El código del programa es más fácil de crear.
- El compilador genera un código de programa optimizado.

Procedimiento en STEP 7 V5.x

En STEP 7 V5.x había que utilizar registros de direcciones para direccionar elementos de array de forma indexada mediante POINTER propios. El lenguaje de programación ya soportaba el direccionamiento indirecto con un índice variable.

Ejemplo de programación en STEP 7 V5.x

Para el siguiente ejemplo en AWL se requiere el bloque de datos "Data_classic". Para llamar un elemento del array "Quantities" deben utilizarse los comandos siguientes:

AWL	Explicación
OPN "Data_classic"	// Se llama el bloque de datos "Data_classic".
L #index	// El valor de la variable local #index se carga en el acumulador 1.
SLD 3	// Desplazar los bits 0 a 31 del acumulador 1 tres posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
LAR1	// Cargar el contenido del acumulador 1 en el registro de direcciones 1.
L DBW [AR1, P#10.0]	// Cargar en el acumulador 1 el elemento de array direccionado mediante #index.
	// P#10.0 = dirección básica del campo

De todas formas, con este procedimiento hay que tener en cuenta lo siguiente:

- El nombre del array no se utiliza. Esto reduce la legibilidad del código del programa y obliga a introducir comentarios.
- La dirección básica del array (P#10.0) tiene que conocerse para el direccionamiento.
- El direccionamiento indexado debe realizarse utilizando el registro de direcciones.

Ejemplo de programación en STEP 7 TIA Portal

Para indexar de forma indirecta el array "Quantities" que se ha declarado en el bloque de datos "Data_DB" se usa la siguiente sintaxis:

```
"Data_DB".Quantities["i"]          // Array unidimensional
"Data_DB".Quantities["i"].a        // Array of STRUCT unidimensional
"Data_DB".Quantities["i","j"]      // Array multidimensional
"Data_DB".Quantities["i","j"].a    // Array of STRUCT multidimensional
```

Parte	Descripción
Data_DB	Nombre del bloque de datos en el que se encuentra el array
Quantities	Variable del tipo de datos Array
i, j	Variables PLC del tipo de datos Entero que se emplean como punteros
a	Otras variables parciales de la estructura

El siguiente ejemplo muestra la indexación indirecta de un componente array en AWL.

Para el direccionamiento de un elemento array se requiere únicamente una línea del programa:

Direccionamiento en AWL	Explicación
`L "Data_DB".Quantities[#index]`	`// El valor del elemento array #index se carga directamente del bloque de datos en el acumulador 1.`

Para conseguir el mejor rendimiento posible deben observarse las indicaciones siguientes:

- Las variables que se utilizan como índice de array deben declararse como entero menor o igual a 32 bits.
- Los resultados intermedios y los índices de array deben crearse en el área temporal de datos locales.

Consulte también

Direccionar operandos de modo indirecto (Página 2305)

Matriz (Página 2354)

4.4.7 Uso de tipos de datos PLC (UDT)

Ventajas de los tipos de datos PLC

Los tipos de datos PLC (UDT) son estructuras de datos definidas por el usuario que pueden utilizarse varias veces en el programa. La estructura de un tipo de datos PLC consta de varios componentes, que pueden pertenecer a diferentes tipos de datos. El tipo de los componentes se define en la declaración del tipo de datos PLC. El uso de tipos de datos PLC permite aprovechar al máximo el gran rendimiento de la CPU S7-1500.

En muchos programas se requieren registros relacionados, que son procesados por diferentes puntos del programa. Se optimizan y cambian mientras se ejecuta el programa. Son, por ejemplo:

- registros para seguimiento de material
- un registro de parámetros para un ajuste del motor
- diferentes recetas

Las ventajas resultantes son las siguientes:

- Los elementos de un tipo de datos PLC también pueden direccionarse indirectamente, es decir, la dirección es variable y no se calcula hasta el momento de la ejecución.
- Las variables que se basan en un tipo de datos PLC heredan todas las propiedades del mismo.
- La modificación del tipo de datos PLC tiene como consecuencia una adaptación automática de todas las variables derivadas.

- Gracias al simbolismo homogéneo es más fácil leer el programa, pues los nombres de los diferentes elementos de un tipo de datos PLC se muestran en el programa.
- Mayor rendimiento, pues se genera un código de programa optimizado.
- Al llamar el bloque es posible transferir el tipo de datos PLC como estructura completa.
- Interfaz de llamada simplificada debido a la reducción del número de parámetros que deben rellenarse.

Procedimiento en STEP 7 V5.x

En STEP 7 V5.x ya existía la posibilidad de crear un registro como variable estructurada utilizando el tipo de datos STRUCT o un tipo de datos PLC (UDT). Sin embargo, el uso del direccionamiento simbólico perjudicaba el rendimiento.

La declaración en los bloques de datos solía realizarse como estructura anónima. En ese caso, los bloques propiamente dichos se programaban de modo que se transferían como parámetros actuales los valores de la estructura y los valores calculados se copiaban de nuevo en la estructura. Eso permitía transferir también el número del bloque de datos y direccionarlo de forma absoluta en el bloque. El número de parámetros que debían rellenarse solía ser muy grande. Los datos actuales se depositaban en los bloques de datos y los valores calculados se transferían a otros bloques. No obstante, a la hora de transferir las variables del bloque de datos ya no estaban disponibles los símbolos.

Procedimiento en STEP 7 TIA Portal

Si se trabaja en el programa con tipos de datos PLC, es posible asignarlos tanto a un parámetro formal como a uno actual. Con ello ya no hace falta declarar parámetros individuales. Si un bloque tiene un parámetro de entrada basado en un tipo de datos PLC, es necesario transferir como parámetro actual una variable que sea del mismo tipo de datos PLC.

Para utilizar un tipo de datos PLC, proceda del siguiente modo:

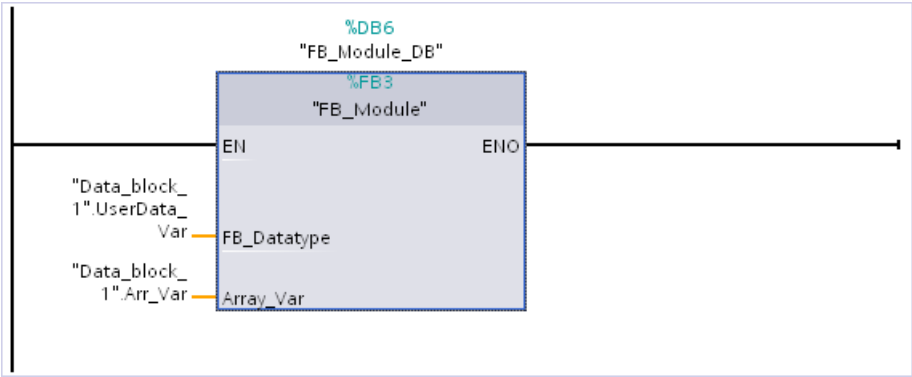
1. Declare los registros como tipo de datos PLC con un nombre apropiado, p. ej. "Seguimiento de material".
2. Utilice este tipo de datos para la declaración de un bloque de datos o una variable DB.
3. Defina el parámetro formal como variable VAR_IN_OUT/VAR_IN o VAR_OUT en la interfaz del bloque.
4. Asigne el tipo de datos PLC al parámetro formal en la columna "Data type".

Ejemplo de programación en el TIA Portal

El ejemplo siguiente muestra la llamada y parametrización de un bloque de función (FB) con dos parámetros formales. El tipo de datos de las variables DB (Array [1 .. 10] of BOOL) y el tipo de datos PLC (UDT) son idénticos a los tipos de datos de los parámetros FB.

Data_block_1

	Name	Data type	Start value
1	Static		
2	Arr_Var	Array [1..10] of Bool	
3	UserData_Var	"UserDataType"	
4	Var1	Bool	false
5	Var2	Bool	false
6	ByteVar1	Byte	16#0
7	WordVar1	Word	16#0
8	IntVar1	Int	0
9	RealVar1	Real	0.0



El tipo de datos PLC "UserDataType" se declara en la carpeta "Tipos de datos PLC" del árbol del proyecto. Seguidamente, se utiliza como tipo de datos para la variable "UserData_Var". Las dos variables "UserData_Var" y "Arr_Var" se declaran en el bloque de datos "Data_block_1" y se utilizan para suministrar valores a los parámetros del bloque "FB_Module".

Consulte también

Información importante sobre variables PLC estructuradas (Página 2579)

Crear variables PLC estructuradas (Página 2582)

Tipos de datos PLC (Página 2367)

Reglas generales para la transferencia de valores de parámetros (Página 2269)

4.4.8 Consultar y eliminar errores en el código del programa

Tratamiento de errores en el TIA Portal

Dentro del código del programa pueden darse varios tipos de errores en runtime. Puede tratarse p. ej. de errores de acceso o de un desbordamiento en el caso de operaciones matemáticas. Si no intercepta estos errores en runtime con una secuencia de código en el programa, en caso de error la CPU reaccionará del siguiente modo en función del modelo:

- CPU S7-1200:
  - La CPU permanece en estado operativo RUN y escribe una entrada en el búfer de diagnóstico.
- CPU S7-1500:
  - En caso de error de programación, la CPU cambia al estado operativo STOP y escribe una entrada en el búfer de diagnóstico.
  - En caso de error de acceso a la periferia, la CPU permanece en estado operativo RUN y escribe una entrada en el búfer de diagnóstico.

Para poder reaccionar debidamente a posibles errores, recomendamos integrar en el programa, además de un tratamiento de errores global (como p. ej. el uso de los bloques de organización "OB de error de programación" u "OB de error de acceso a la periferia"), un tratamiento de errores local (como p. ej. el uso del mecanismo EN/ENO, los parámetros RET_VAL, STATUS y ERROR o el uso de las instrucciones "GET_ERROR"/"GET_ERR_ID").

Nota

Lenguaje de programación AWL

En AWL se utiliza el bit BR de la palabra de estado en lugar del mecanismo EN/ENO.

El tratamiento de errores global actúa al final de cada ciclo del programa, mientras que el tratamiento de errores local permite intervenir justo después de producirse el error.

Nota

Bit BR y mecanismo EN/ENO como primer indicador

Un primer indicador de un error puede ser el bit BR de la palabra de estado o la salida de habilitación ENO. Si estos devuelven el estado lógico "0", hay un error en la ejecución de la instrucción. Con el estado lógico "1" no hay ningún error y no se requieren mayores análisis.

Posibilidades del tratamiento de errores local

Las opciones de tratamiento local de errores de programación y acceso disponibles son las siguientes:

Tipo de tratamiento de errores	Validez	Explicación
Mecanismo EN/ENO	S7-1200/1500	La salida de habilitación ENO permite detectar y tratar errores en runtime. La ejecución de las siguientes instrucciones varía en función del estado lógico de este parámetro. Con el uso del mecanismo EN/ENO se evitan interrupciones del programa. El estado del bloque se transmite en forma de variable booleana.
Parámetro de salida RET_VAL	S7-1200/1500	El parámetro RET_VAL como valor de retorno de funciones de sistema (SFC) permite visualizar códigos de error generales, que pueden referirse a cualquier instrucción, o bien específicos, que solo se refieren a la instrucción en cuestión. Puede devolverse como máximo una variable del tipo de datos INT o WORD. Este tipo de tratamiento de errores es adecuado, p. ej., para consultar errores de comunicación o un acceso erróneo a datos.
Parámetros de salida STATUS y ERROR	S7-1200/1500	Con los parámetros STATUS y ERROR como valores de retorno de bloques de función del sistema (SFB) se puede consultar información de error específica de bloque. La información de error se devuelve en una estructura predeterminada.
Instrucciones "GET_ERROR" y "GET_ERR_ID"	S7-1200/1500	Estas instrucciones permiten obtener una ID de error o información detallada de un error. A partir de la información de error se puede ver, p. ej. en el caso de un error de acceso, qué parámetros han causado el error de acceso. Para que las instrucciones puedan devolver la información necesaria del error, deben programarse en el programa de usuario para cada uno de los bloques de los cuales deben evaluarse posibles errores. Esta posibilidad es especialmente idónea para librerías propias con tratamiento de errores. Si trabaja con las instrucciones "GET_ERROR" o "GET_ERR_ID", no se llama ningún OB de error y no se produce ninguna entrada en el búfer de diagnóstico. Con este tipo de tratamiento de errores se interviene activamente en el desarrollo del programa en el momento en que se produce el error.
Bit BR de la palabra de estado	S7-1200/1500	El bit BR de la palabra de estado permite detectar si se ha producido un error al ejecutar la instrucción. (Bit BR = "0" => hay un error; bit BR = "1" => No hay ningún error)

⚠ ADVERTENCIA

Parámetro de salida RET_VAL

Si durante la ejecución de una instrucción que contenga el parámetro RET_VAL se producen errores mientras los parámetros de entrada reciben valores, en el parámetro RET_VAL se devuelve un código de error no válido y los parámetros de salida de la instrucción no pueden evaluarse.

Ejemplo

Las opciones de tratamiento de errores indicadas arriba pueden programarse tanto individual como conjuntamente. Para garantizar la detección de todos los tipos de error que pueden aparecer en el programa, recomendamos combinar las opciones de tratamiento de errores de la forma descrita en el ejemplo siguiente.

Para un análisis preciso de los errores, además del parámetro de salida RET_VAL también se pueden utilizar las instrucciones "GET_ERROR" o "GET_ERR_ID". Estas opciones proporcionan códigos de error, cuya explicación detallada puede consultarse en la descripción de cada instrucción.

No obstante, también hay tipos de error en los que el parámetro de salida RET_VAL no devuelve ningún código de error. Si se produce, p. ej., un error de acceso durante la lectura de un parámetro de entrada, las salidas de la instrucción ya no se escriben, ya que la ejecución de la instrucción se ha interrumpido. En tal caso recomendamos integrar en el programa ambas instrucciones "GET_ERROR" o "GET_ERR_ID", ya que en este caso también proporcionan información fiable.

⚠ ADVERTENCIA

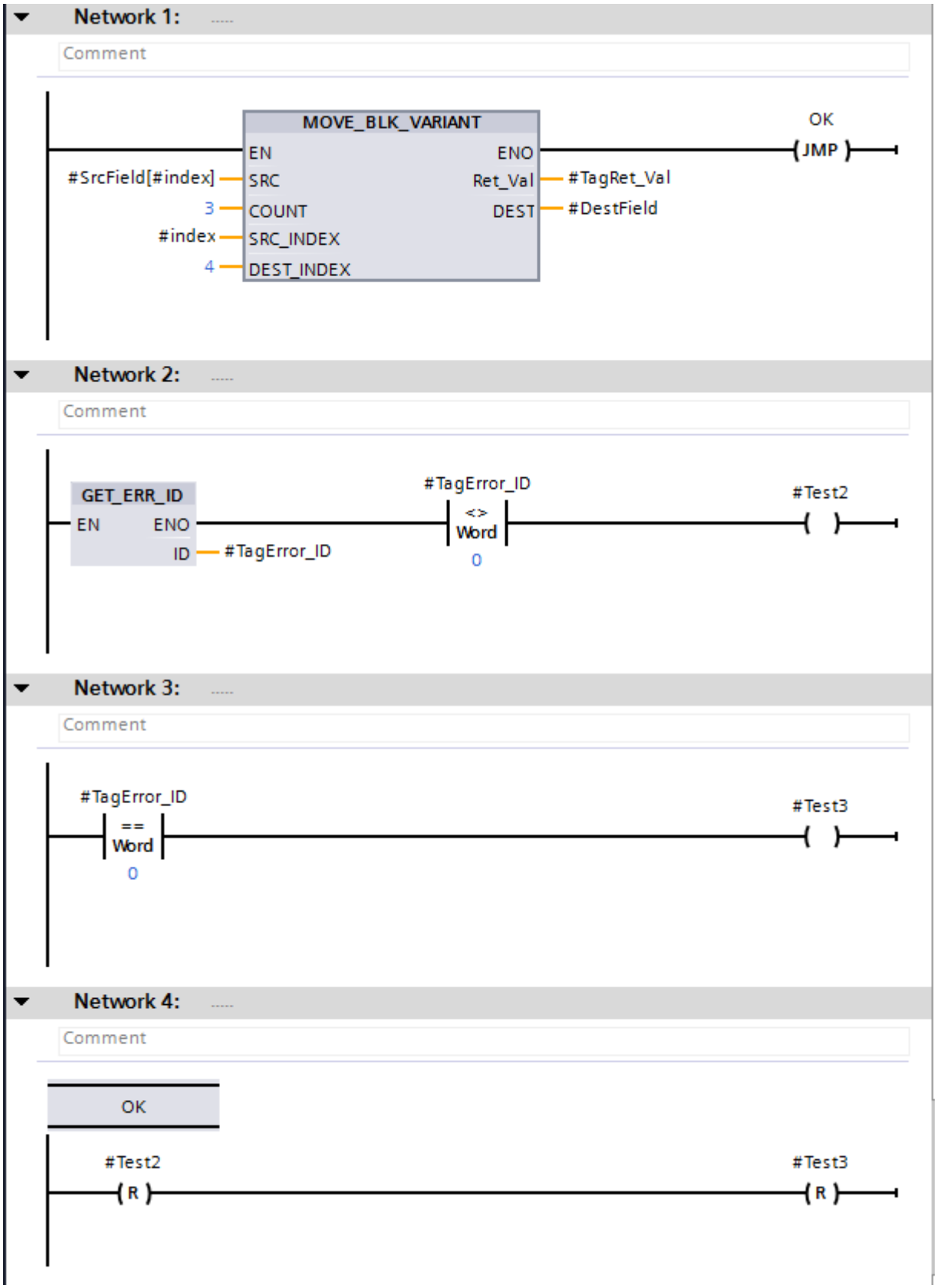
Error de acceso durante la lectura de un parámetro de entrada

El parámetro RET_VAL no proporciona ningún código de error válido y en el búfer de diagnóstico no se registra información detallada del error.

El siguiente ejemplo muestra cómo detectar un error de acceso durante la lectura de un parámetro de entrada.

Interface

	Name	Data type	Default value
1	Input		
2	SrcField	Array [1..100] of Real	
3	index	DInt	0
4	<Add new>		
5	Output		
6	TagOutput	Real	0.0
7	Test2	Bool	false
8	TagRet_Val	Int	0
9	DestField	Array [0..20] of Real	
10	Test3	Bool	false
11	<Add new>		
12	InOut		
13	Static		
14	Temp		
15	TagError_ID	Word	


Network 1:
Comment
MOVE_BLK_VARIANT
EN
ENO
#SrcField[#index]
SRC
Ret_Val
#TagRet_Val
3
COUNT
DEST
#DestField
#index
SRC_INDEX
4
DEST_INDEX
OK
JMP
Network 2:
Comment
GET_ERR_ID
EN
ENO
ID
#TagError_ID
#TagError_ID
<>
Word
0
#Test2
Network 3:
Comment
#TagError_ID
==
Word
0
#Test3
Network 4:
Comment
OK
#Test2
R
#Test3
R

En el segmento 1 se llama la instrucción "MOVE_BLK_VARIANT: Copiar área". En el parámetro SRC se accede al área de origen "SrcField" con un índice variable. Si durante la ejecución de la instrucción no se producen errores, la salida de habilitación ENO devuelve el estado lógico "1" y la ejecución del programa salta al segmento 4 y continúa allí.

Si durante la ejecución de la instrucción, p. ej. debido al índice variable, se produce un error de acceso, la instrucción "GET_ERR_ID: Consultar ID de error localmente" en el segmento 2 devuelve una ID de error. La comprobación de si el ID de error es "DIFERENTE" al valor "0" devuelve el resultado "Test2" = TRUE.

En tal caso, el operando #TagRet_Val del parámetro de salida RET_VAL no devuelve ningún código de error válido.

Consulte también

Tratar errores de ejecución del programa (Página 2932)

GET_ERROR: Consultar error localmente (Página 3527)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Principios básicos del mecanismo EN/ENO (Página 2539)

Parámetro STATUS (Página 4755)

4.4.9 Uso de instrucciones VARIANT

El tipo de datos VARIANT

Para crear bloques de función o funciones genéricos, se puede usar el tipo de datos VARIANT. Estos permiten interconectar los parámetros del bloque con variables de cualquier tipo durante la llamada. Los parámetros del tipo de datos VARIANT se pueden declarar en la sección VAR_IN, VAR_IN_OUT y VAR_TEMP.

Al llamar el bloque, además del puntero a la variable también se transfiere información del tipo de la variable. De este modo es posible procesar el código del bloque conforme al tipo y la variable transferida durante la ejecución.

Si, p. ej., un parámetro de bloque de una función tiene el tipo de datos VARIANT, se le puede transferir en un punto del programa una variable del tipo de datos Entero, y en otro punto una variable del tipo de datos PLC. En este caso, las instrucciones VARIANT permiten que la función reaccione sin errores.

Procedimiento en el S7-300/400

En las CPU de las series S7-300/400 se podía utilizar el tipo de datos ANY del siguiente modo:

- En SCL y AWL se podía transferir cualquier tipo de memoria en una llamada de bloque si en un parámetro del bloque se programaba un puntero del tipo de datos ANY.
  Sin embargo, de este modo se pierde información esencial sobre la estructura de la memoria, ya que p. ej. el puntero ANY no memoriza que señala a una variable del tipo de datos PLC. El puntero ANY lo considera ARRAY of BYTE.
- Se podían transferir parámetros del tipo de datos ANY a bloques de función de sistema (SFB) o funciones de sistema (SFC).
- En combinación con el tipo de datos ANY se podía utilizar la instrucción "BLKMOV: Copiar área" para copiar en el programa datos de cualquier tipo.

El tipo de datos ANY presenta las siguientes limitaciones:

- El puntero ANY debía crearse manualmente
- No podían transferirse datos con acceso a bloque optimizado.
- El puntero ANY no podía señalar a estructuras.

Procedimiento en el S7-1500

En el TIA Portal recomendamos utilizar el tipo de datos VARIANT en lugar del puntero ANY.

Ventajas del tipo de datos VARIANT:

- El tipo de datos VARIANT permite escribir programas que ejecuten una tipificación dinámica, es decir, una comprobación de tipo integrada durante la ejecución del programa. El tipo de datos VARIANT no pierde la información sobre el tipo de datos de la variable a la que señala, de modo que si el parámetro de bloque de una función del tipo VARIANT recibe un tipo de datos PLC, se puede comprobar dentro de la función y en tiempo de ejecución si VARIANT señala a "myUDTa" o "myUDTb".
- El puntero VARIANT también puede crearse simbólicamente. Ello mejora la legibilidad del código del programa. Encontrará más información sobre las ventajas del direccionamiento simbólico en: Direccionamiento simbólico (Página 253)
- A un parámetro del tipo VARIANT se puede transferir cualquier tipo de datos.
- El tipo de datos del parámetro actual al que señala VARIANT se determina en runtime. Así es posible programar en el código del programa reacciones distintas en función del tipo de datos del parámetro actual.

Encontrará más información sobre cómo trabajar con el tipo de datos VARIANT en el Siemens Industry Online Support en: Direccionamiento indirecto con el tipo de datos VARIANT (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=0&lang=es&referer=%2fWW%2f&func=cslib.csinfo&siteid=csius&groupid=4000002&extranet=standard&viewreg=WW&nodeid0=29156492&objaction=csopen)

En el TIA Portal se dispone, entre otras, de las siguientes instrucciones con el tipo de datos VARIANT como parámetro:

Instrucciones básicas		
Lenguaje de programación	**Instrucción**	**Descripción**
KOP/FUP - Comparación	EQ_Type	Comparar si el tipo de datos es IGUAL al tipo de datos de una variable
	NE_Type	Comparar si el tipo de datos es DIFERENTE al tipo de datos de una variable
	EQ_ElemType	Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable
	NE_ElemType	Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable
	IS_NULL	Comparar si es IGUAL A CERO
	NOT_NULL	Comparar si es DIFERENTE DE CERO
	IS_ARRAY	Consultar si es un ARRAY
SCL - Comparación	TypeOf	Consultar tipo de datos de una variable VARIANT
	TypeOfElements	Consultar tipo de datos de un elemento de una variable VARIANT
	IS_ARRAY	Consultar si es un ARRAY
KOP/FUP/SCL/GRAPH - Transferencia	VariantGet	Leer valor de una variable VARIANT
	VariantPut	Escribir valor en una variable VARIANT
	CountOfElements	Consultar número de elementos de ARRAY
KOP/FUPAWL/SCL/GRAPH - Transferencia	ReadFromArrayDB	Leer en un bloque de datos de array
	WriteToArrayDB	Escribir en un bloque de datos ARRAY
	ReadFromArrayDBL	Leer en un bloque de datos ARRAY de la memoria de carga
	WriteToArrayDBL	Escribir en un bloque de datos ARRAY de la memoria de carga
	BLKMOV	Copiar área
	UBLKMOV	Copiar área sin interrupciones
	FILL	Rellenar área
	Deserialize	Copiar tipo de datos de ARRAY of BYTE (Deserializar)
	Serialize	Copiar tipo de datos en ARRAY of BYTE (Serializar)
	MOVE_BLK_VARIANT	Copiar área
SCL - Transferencia	READ_LITTLE	Leer datos en formato Little-Endian
	WRITE_LITTLE	Escribir datos en formato Little-Endian
	READ_BIG	Leer datos en formato Big-Endian
	WRITE_BIG	Escribir datos en formato Big-Endian
KOP/FUP/AWL/SCL - Conversión	VARIANT_TO_DB_ANY	Convertir VARIANT en DB_ANY
	DB_ANY_TO_VARIANT	Convertir DB_ANY en VARIANT

Encontrará otras instrucciones con el tipo de datos VARIANT en la Ayuda en pantalla, en "Instrucciones avanzadas".

Ejemplo

El siguiente ejemplo muestra un ejemplo de una cola de espera (FIFO, First In First Out) que trabaja con elementos de cualquier tipo de datos en el lenguaje de programación SCL. Programado utilizando las instrucciones descritas arriba.

Declaración de la interfaz del bloque de función "FIFOQueue":

Declaración	Parámetro	Tipo de datos	Comentario
Input	request	BOOL	La instrucción se ejecuta cuando en el parámetro "request" se registra un flanco de señal ascendente.
	mode	BOOL	0 = se devuelve la primera entrada del búfer en anillo. 1 = se escribe una entrada en la última posición del búfer en anillo.
	initialValue	VARIANT	Valor con el que se inicializa el ARRAY del búfer en anillo.
Output	error	INT	Información de error
InOut	item	VARIANT	La entrada que se devuelve del búfer en anillo o que se escribe en él.
	buffer	VARIANT	Un ARRAY que se utiliza como búfer en anillo.
Static	edgeupm	BOOL	Marca de flancos en la que se almacena el RLO de la consulta anterior.
	firstItemIndex	INT	Índice de la entrada más antigua del búfer en anillo
	nextEmptyItemIndex	INT	Índice del siguiente elemento libre en el búfer en anillo
Temp	edgeup	BOOL	Resultado de la evaluación de flancos
	internalError	INT	Información de error
	newFirstItemIndex	INT	Índice variable
	newNextEmptyItemIndex	INT	Índice variable
	bufferSize	UDINT	Número de elementos ARRAY en el búfer en anillo

Código de programa del bloque de función "FIFOQueue":

```
(* Esta sección del código del programa solo se ejecuta una vez tras un flanco de señal
ascendente. Si no hay ningún cambio en el estado lógico del enlace, la ejecución del
programa del FB "FIFOQueue" se interrumpe. *)
#edgeup := #request & NOT #edgeupm;
#edgeupm := #request;
IF NOT (#edgeup) THEN
```

```
   RETURN;
END_IF;


// ------Validación de si todas las entradas de parámetro son válidas.----
(* Esta sección del código del programa comprueba si el búfer en anillo es un ARRAY. En caso
afirmativo, se lee el número de elementos del ARRAY. Si no se trata de un ARRAY, la
ejecución del programa se interrumpe en este punto y se devuelve el código de error "-10".
*)
IF NOT (IS_ARRAY(#buffer)) THEN
   #error := -10;
   RETURN;
ELSE
   #bufferSize := CountofElements(#buffer);
END_IF;

(* Esta sección del código del programa comprueba si el tipo de datos de los elementos ARRAY
coincide con el tipo de datos de la entrada (variable #item). Si los tipos de datos no
coinciden, la ejecución del programa se interrumpe en este punto y se devuelve el código de
error "-11". *)
IF NOT (TypeOf(#item) = TypeOfElements(#buffer)) THEN
   #error := -11;
   RETURN;
END_IF;

(* Esta sección del código del programa comprueba si el valor inicial del búfer en anillo
coincide con la entrada (variable #item). Si los tipos de datos no coinciden, la ejecución
del programa se interrumpe en este punto y se devuelve el código de error "-12". *)
IF NOT (TypeOf(#item) = TypeOf(#initialValue)) THEN
   #error := -12;
   RETURN;
END_IF;

(* Esta sección del código del programa comprueba si los índices variables se encuentran
dentro de los límites ARRAY. Si no es así, la ejecución del programa se interrumpe en este
punto y por cada índice se devuelven los códigos de error "-20" o "-21". *)
IF (#nextEmptyItemIndex >= #bufferSize) THEN
   #error := -20;
   RETURN;
END_IF;
IF (#firstItemIndex >= #bufferSize) THEN
   #error := -21;
   RETURN;
END_IF;
```

```
//-----------Ejecución del código del programa en función del parámetro Mode-------------
// La ejecución de las instrucciones depende del estado lógico del parámetro Mode.
IF #mode = 0 THEN

// Si el parámetro Mode presenta el estado lógico "0", se devuelve la primera entrada del
búfer en anillo transferido.
(* Esta sección del código del programa comprueba si el búfer en anillo está vacío. En caso
afirmativo, la ejecución del programa se interrumpe en este punto y se devuelve el código
de error "-40". *)
   IF (#firstItemIndex = -1) THEN
      #error := -40;
      RETURN;
   END_IF;

// Esta sección del código del programa devuelve la primera entrada del búfer en anillo.
   #internalError := MOVE_BLK_VARIANT(SRC := #buffer,
                                      COUNT := 1,
                                      SRC_INDEX := #firstItemIndex,
                                      DEST_INDEX := 0,
                                      DEST => #item);

   IF (#internalError = 0) THEN
(* Esta sección del código del programa comprueba si el búfer en anillo contiene elementos
de ARRAY. En caso afirmativo, se desplaza la primera entrada y el índice incrementa en 1.
*)
      #internalError := MOVE_BLK_VARIANT(SRC := #initialValue,
                                         COUNT := 1,
                                         SRC_INDEX := 0,
                                         DEST_INDEX := #firstItemIndex,
                                         DEST => #buffer);

// Esta sección del código del programa calcula el nuevo índice de la primera entrada.
      #newFirstItemIndex := #firstItemIndex +1;
      #newFirstItemIndex := #newFirstItemIndex MOD #bufferSize;

// Esta sección del código del programa comprueba si el búfer en anillo está vacío.
      IF (#nextEmptyItemIndex = #newFirstItemIndex) THEN
// Si el búfer en anillo está vacío, el índice se pone a 0.
         #firstItemIndex := -1;
         #nextEmptyItemIndex := 0;
      ELSE
// El índice de la primera entrada se modifica.
         #firstItemIndex := #newFirstItemIndex;
      END_IF;
   END_IF;
```

```
ELSE


// Si el parámetro Mode presenta el estado lógico "1", la entrada se escribe en el búfer en
anillo transferido.
(* Esta sección del código del programa comprueba si el búfer en anillo está lleno. En caso
afirmativo, la ejecución del programa se interrumpe en este punto y se devuelve el código
de error "-50". *)
   IF (#nextEmptyItemIndex = #firstItemIndex) THEN
      #error := -50;
      RETURN;
   END_IF;

// Esta sección del código del programa escribe la entrada en el búfer en anillo.
   #internalError := MOVE_BLK_VARIANT(SRC := #item,
                                      COUNT := 1,
                                      SRC_INDEX := 0,
                                      DEST_INDEX := #nextEmptyItemIndex,
                                      DEST => #buffer);

   IF (#internalError = 0) THEN
// Esta sección del código del programa incrementa el índice en 1 y calcula el nuevo índice
de entrada vacío.
      #newNextEmptyItemIndex := #nextEmptyItemIndex +1;
      #newNextEmptyItemIndex := #newNextEmptyItemIndex MOD #bufferSize;
      #nextEmptyItemIndex := #newNextEmptyItemIndex;

(* Esta sección del código del programa comprueba qué índice tiene la variable
"#firstItemIndex". Si el número = -1, el búfer en anillo se ha inicializado y la entrada se
escribe en el búfer en anillo. Por ello hay que asignar "0" a la variable. *)
      IF (#firstItemIndex = -1) THEN
         #firstItemIndex := 0;
      END_IF;
   END_IF;
END_IF;

//-------------------------Tratamiento local de errores----------------------------
(* Esta sección del código del programa comprueba si se ha producido un error local. En caso
afirmativo, el programa se interrumpe en este punto y se devuelve el código de error "-100".
*)
IF (#internalError > 0) THEN
   #error := -100;
   RETURN;
END_IF;
```

```
// Si no se ha producido ningún error durante la ejecución del programa, se devuelve el
código de error "0".
#error := 0;
```

Consulte también

VARIANT (Página 2364)

TypeOf: Consultar tipo de datos de una variable VARIANT (Página 4331)

TypeOfElements: Consultar tipo de datos de un elemento de una variable VARIANT (Página 4332)

IS_ARRAY: Consultar si es ARRAY (Página 4334)

MOVE_BLK_VARIANT: Copiar área (Página 4363)

Primeros pasos 5

5.1 Documentación en forma de Getting Started

Iniciación al TIA Portal

Para iniciarse en el TIA Portal se dispone de documentación en forma de Getting Started.

La documentación en forma de Getting Started son instrucciones que describen paso a paso cómo crear un proyecto en el TIA Portal y ofrecen la posibilidad de familiarizarse rápidamente con el volumen de prestaciones del TIA Portal.

Contenidos

Las documentaciones Getting Started describen cómo crear un proyecto homogéneo para STEP7 y WinCC y ampliarlo paso a paso en cada capítulo. Empezando con funciones básicas simples, irá utilizando funciones cada vez más complejas para crear el proyecto.

Además de las instrucciones paso a paso, la documentación en forma de Getting Started ofrece para cada nuevo tema breve información de fondo que explica la función utilizada con más detalle y aclara las interrelaciones.

Grupo objetivo

La documentación en forma de Getting Started está dirigida a principiantes, pero también es adecuada para nuevos usuarios familiarizados con las versiones anteriores de SIMATIC STEP 7 y WinCC

Descarga

Este tipo de documentación está disponible gratuitamente en formato de PDF en el portal del Service&Support (https://support.automation.siemens.com).

La documentación se puede descargar a través de los enlaces siguientes:

- STEP 7 Basic y WinCC Basic (V10.5) (http://support.automation.siemens.com/WW/view/es/40263542/0/es)
- STEP 7 Professional y WinCC Advanced (V11) (http://support.automation.siemens.com/WW/view/es/28919804/133300)
- STEP 7 Professional y WinCC Advanced (V12) (http://www.automation.siemens.com/salesmaterial-as/interactive-manuals/getting-started_simatic-s7-1500/_content/EN/content_en.html)

Introducción al TIA Portal 6

6.1 Interfaz de usuario y manejo

6.1.1 Iniciar, configurar y cerrar el TIA Portal

6.1.1.1 Iniciar y cerrar el TIA Portal

Iniciar el TIA Portal

Para iniciar el TIA Portal, proceda del siguiente modo:

1. En Windows, seleccione el comando "Inicio > Programas > Siemens Automation > TIA Portal V13".
   El TIA Portal se abre con los ajustes que se utilizaron la última vez.

Cerrar el TIA Portal

Para cerrar el TIA Portal, proceda del siguiente modo:

1. Elija el comando "Salir" del menú "Proyecto".
   Si el proyecto contiene cambios que aún no han sido guardados, aparecerá la pregunta de si desea guardarlos.
   - Seleccione "Sí" para guardar los cambios en el proyecto actual y cerrar el TIA Portal.
   - Seleccione "No" para cerrar el TIA Portal sin guardar los últimos cambios efectuados en el proyecto.
   - Elija "Cancelar" para cancelar el cierre. En ese caso, el TIA Portal permanece abierto.

6.1.1.2 Resumen de los ajustes del programa

Sinopsis

La tabla siguiente muestra los ajustes que se pueden efectuar en la aplicación:

Grupo	Ajuste	Descripción
Configuración general	Nombre de usuario	Nombre correspondiente al usuario. El nombre de usuario se guarda en las propiedades del proyecto al crear un proyecto.
	Idioma de la interfaz	Idioma utilizado en la interfaz del programa

Grupo	Ajuste	Descripción
	Nemónicos	Define los nemónicos para la programación: "Alemán" utiliza los nemónicos alemanes, p. ej. "E1.0" "Internacional" utiliza los nemónicos internacionales, p. ej. "I1.0" Las diferencias respecto a los nemónicos de los distintos comandos se explican en la descripción del lenguaje de programación en cuestión.
	Mostrar la lista de los últimos proyectos utilizados	Número de entradas de la lista de los últimos proyectos utilizados en el menú "Proyecto"
	Al iniciar, cargar el último proyecto abierto	El último proyecto abierto se abre automáticamente al iniciar el TIA Portal.
	Visualización completa de textos cortados	Los textos que aparecen cortados por ser demasiado largos se muestran en un tooltip.
	Mostrar tooltips (ayuda contextual disponible)	Los tooltips se visualizan y el usuario obtiene ayuda contextual. Si la función está desactivada, el tooltip se puede abrir con <F1>.
	Abrir automáticamente los tooltips en cascada	Transcurrido poco tiempo los tooltips se despliegan automáticamente y se visualiza ayuda adicional en cascada. Si la opción está desactivada, los tooltips deben desplegarse manualmente.
Restablecer a predeterminado	Todos los ajustes de la aplicación	Se deshacen todos los cambios realizados en el TIA Portal después de la instalación.
	Diseño de los editores	Restablece el estado de suministro de todo el diseño de la aplicación.
	Mostrar todas las ventanas de aviso	Todas las ventanas de aviso cuya aparición se suprimió manualmente se vuelven a visualizar.
Vista inicial	Última vista utilizada	Inicia el programa siempre en la última vista utilizada. Ésta puede ser la vista del portal o la vista del proyecto.
	Vista del portal	Inicia el TIA Portal siempre en la vista del portal, sin importar en qué vista se trabajó por última vez.
	Vista del proyecto	Inicia el TIA Portal siempre en la vista del proyecto, sin importar en qué vista se trabajó por última vez.
Vista de los objetos en la ventana general	Detalles	Si hay varias vistas disponibles, se inicia de forma predeterminada la vista detallada; por ejemplo, en la ventana general.
	Lista	Si hay varias vistas disponibles, se inicia de forma predeterminada la vista de lista; por ejemplo, en la ventana ventana.
	Iconos	Si hay varias vistas disponibles, se inicia de forma predeterminada la vista de iconos; por ejemplo, en la ventana general.
Opciones de almacenamiento	Usar última ubicación utilizada	Al guardar por primera vez un proyecto está predefinida la última ruta utilizada para guardar.
	Definir el ajuste estándar de la ubicación	Permite definir la ruta para guardar: • proyectos • librerías • archivo de configuración para librerías de empresa
Intercambio de datos	Ubicación para importación de datos	De forma estándar, los archivos importados se buscan en esta ruta de almacenamiento.

Grupo	Ajuste	Descripción
	Ubicación para la exportación de datos	Esta ruta de almacenamiento está predefinida para la exportación de datos.
	Ubicación para Support Packages	Después de cargar support packages, éstos se almacenan en la ruta de almacenamiento indicada y, después, pueden instalarse desde dicha ubicación.
	Ubicación para archivos de registro	Los archivos de registro se guardan en el lugar aquí indicado.

Consulte también

Iniciar y cerrar el TIA Portal (Página 279)

Restablecer diseño de interfaz de usuario (Página 313)

Modificar la configuración (Página 283)

Configurar la visualización de tooltips y tooltips en cascada (Página 334)

6.1.1.3 Sinopsis de la configuración de los editores de scripts y de textos

Sinopsis

La tabla siguiente muestra los ajustes que se pueden efectuar para los editores de scripts y de textos:

Grupo	Ajuste	Descripción
Fuente	Estilo y tamaño de fuente	Ajusta el estilo y el tamaño de fuente para el texto de los editores de texto.
Colores de fuente	Ajuste de colores	En las listas desplegables correspondientes se pueden elegir los colores para los distintos elementos de texto de los editores de texto. Se dispone de opciones de ajuste para los siguientes elementos de texto: • Texto • Palabras clave • Comentarios • Operandos • Scripts • Funciones estándar • Instrucciones/Funciones de sistema • Constantes string • Constantes numéricas • Constantes simbólicas • Variables • Modelos de objeto • Parámetro formal
	Restablecer a predeterminado	Restablece los ajustes de fábrica de todos los colores de fuente en los editores.

Grupo	Ajuste	Descripción
Tabuladores	Ancho de tabulación	Ajusta el ancho de los tabuladores.
	Utilizar tabuladores	Activa el uso de tabuladores.
	Utilizar espacios	Utiliza espacios en vez de tabuladores.
Sangría	Ninguna	No se sangran automáticamente las entradas de texto.
	Párrafo	Se sangra automáticamente la línea o el párrafo seleccionado.
	Smart	Se sangra automáticamente la línea o el párrafo seleccionado. Además se eliminan todos los espacios en blanco innecesarios.
Vista	Mostrar espacios en blanco	Muestra caracteres de control dentro de un texto.
AWL (lista de instrucciones)	Estilo y tamaño de fuente	Ajusta el estilo y el tamaño de fuente para el código de programa AWL.
SCL (Structured Control Language)	Mostrar números de línea	Muestra los números de línea en los programas SCL.

Consulte también

Modificar la configuración (Página 283)

6.1.1.4 Resumen de los ajustes de impresión

Sinopsis

La tabla siguiente muestra los ajustes que se pueden efectuar para la impresión:

Grupo	Ajuste	Descripción
General	Imprimir los datos de tabla siempre en parejas de valores	Las tablas no se imprimen en forma de tabla, sino en forma de lista. Junto a cada columna se enumeran los correspondientes valores. Active esta opción, por ejemplo, si desea imprimir una tabla demasiado grande para la superficie de impresión.
Configuración hardware	Vista gráfica activa	Los gráficos de la vista de redes y de dispositivos también se imprimen.
	Tabla activa	Al imprimir el editor se imprime también la tabla perteneciente al editor, si la hubiere.
Programación PLC	Factor de zoom	Define el tamaño en que deben imprimirse los bloques.
	con interfaz	Las interfaces de bloques también se imprimen.
	con comentarios	Los comentarios de bloques también se imprimen.
	con números de línea	En los lenguajes de programación textuales también se imprimen los números de línea del código de programa.
Movimiento y tecnología	Representación de diálogos/ gráficos	El contenido del editor se imprime en forma de gráfico, siempre que el editor soporte este formato.
	Tabla	Los parámetros de los objetos tecnológicos se imprimen en forma de tabla.
Imágenes HMI	Mostrar orden de tabulación	En la impresión se indica el orden en el que pueden seleccionarse los objetos en runtime con el tabulador.

Consulte también

Modificar la configuración (Página 283)

6.1.1.5 Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General" en la navegación local para modificar los ajustes descritos en los capítulos anteriores. O bien haga clic en una de las otras entradas de la navegación local para modificar los ajustes de los productos instalados.
3. Modifique la configuración.

Resultado

El cambio se aplica inmediatamente sin necesidad de guardarlo explícitamente.

Consulte también

Resumen de los ajustes del programa (Página 279)

Sinopsis de la configuración de los editores de scripts y de textos (Página 281)

Resumen de los ajustes de impresión (Página 282)

6.1.2 Estructura de la interfaz de usuario

6.1.2.1 Vistas

Vistas

Para el proyecto de automatización se dispone de tres vistas diferentes, a saber:

- La vista del portal es una vista orientada a las tareas del proyecto.
- La vista del proyecto es una vista de los componentes del proyecto con sus áreas de trabajo y editores.
- La vista de librerías (Página 288) muestra los elementos de la librería del proyecto y de las librerías globales abiertas.

Un enlace permite cambiar entre ambas vistas.

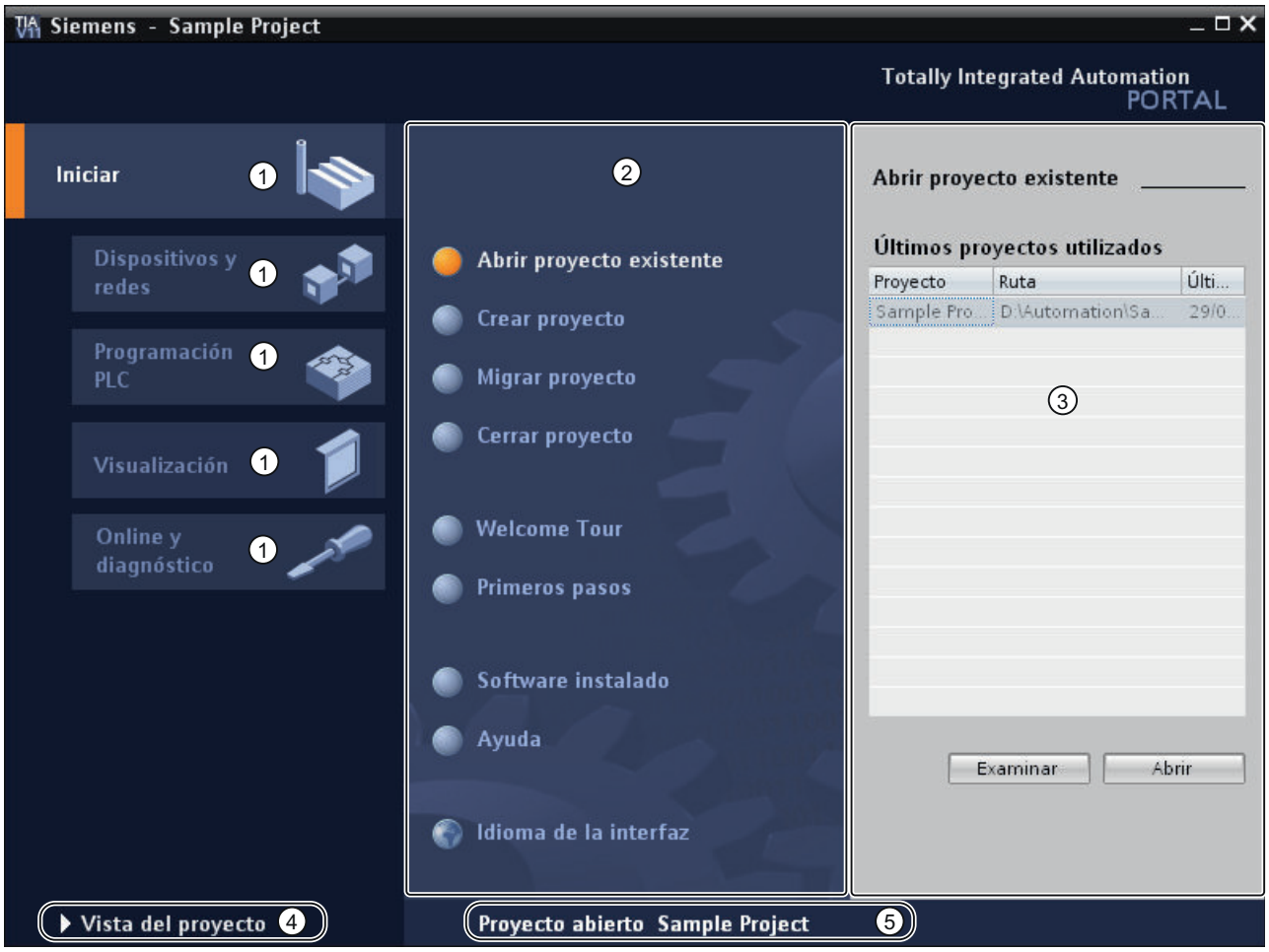
6.1.2.2 Vista del portal

Función de la vista del portal

La vista del portal ofrece una vista de las herramientas orientada a las tareas. Esta vista permite decidir rápidamente qué hacer y seleccionar una herramienta para la tarea en cuestión. En caso necesario se cambia automáticamente a la vista del proyecto (Página 285) para realizar la tarea seleccionada.

Estructura de la vista del portal

La figura siguiente muestra un ejemplo de los componentes de la vista del portal:



① Portales para las distintas tareas
② Acciones del portal seleccionado
③ Ventana de selección de la acción elegida
④ Cambiar a la vista del proyecto
⑤ Indicación del proyecto abierto actualmente

Portales

Los portales proveen las funciones básicas para los distintos campos de tareas. Los portales disponibles en la vista del portal dependen de los productos instalados.

Acciones del portal seleccionado

Aquí aparecen las acciones realizables en el portal en cuestión, que pueden variar en función del portal. El acceso contextual a la Ayuda es posible desde cualquier portal.

Ventana de selección de la acción elegida

La ventana de selección está disponible en todos los portales. El contenido de la ventana se adapta a la selección actual.

Cambiar a la vista del proyecto

El enlace "Vista del proyecto" permite cambiar a la vista del proyecto.

Indicación del proyecto abierto actualmente

Aquí se indica qué proyecto está abierto actualmente.

Consulte también

Árbol del proyecto (Página 288)

Principios básicos del área de trabajo (Página 291)

Ventana de inspección (Página 299)

Principios básicos de las Task Cards (Página 301)

Vista detallada (Página 305)

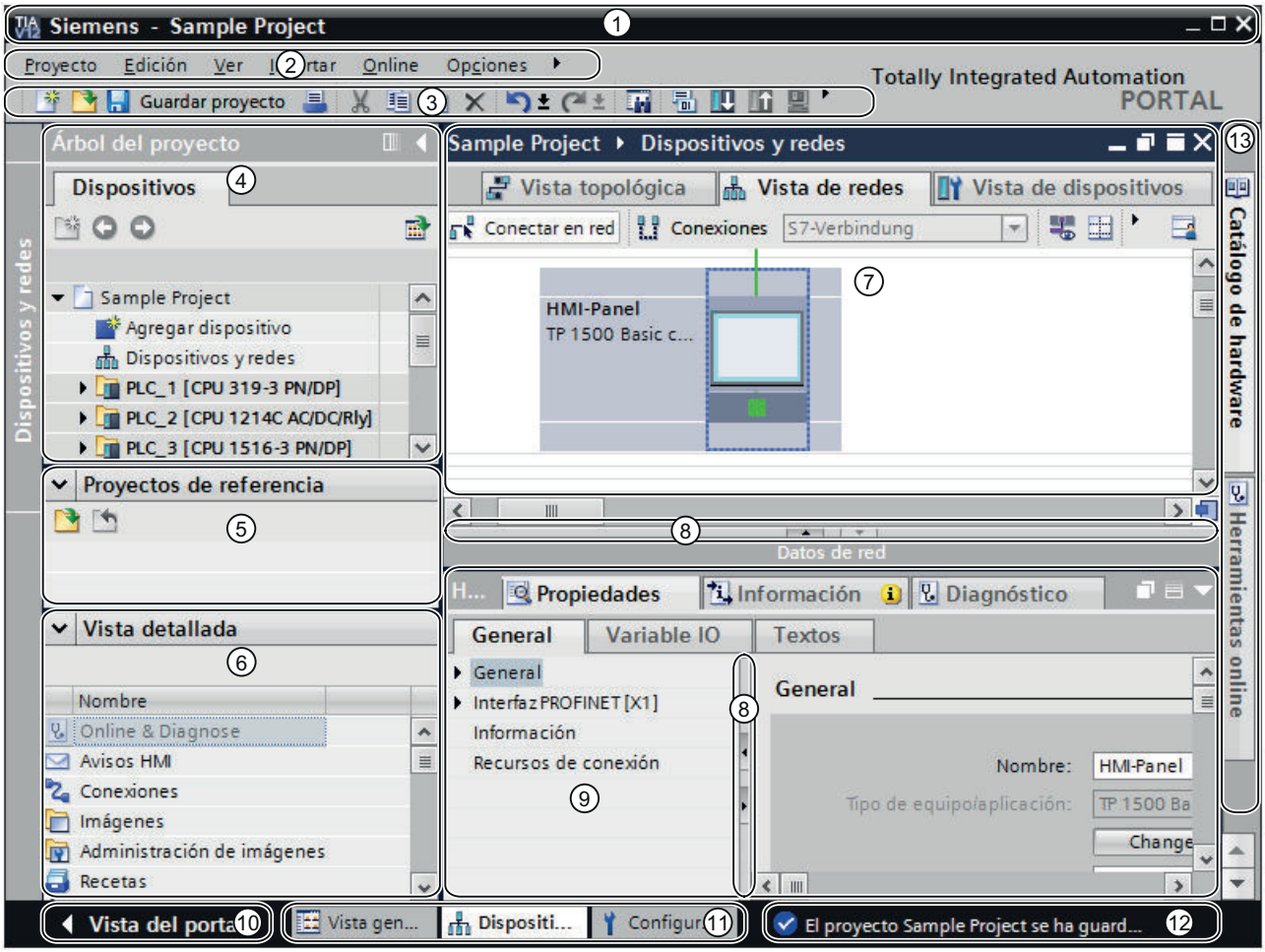
6.1.2.3 Vista del proyecto

Función de la vista del proyecto

La vista del proyecto ofrece una vista estructurada de todos los componentes del proyecto.

Estructura de la vista del proyecto

La figura siguiente muestra un ejemplo de los componentes de la vista del proyecto:



① Barra de título
② Barra de menús
③ Barra de herramientas
④ Árbol del proyecto (Página 288)
⑤ Proyectos de referencia (Página 303)
⑥ Vista detallada (Página 305)
⑦ Área de trabajo (Página 301)
⑧ Divisor de ventana
⑨ Ventana de inspección (Página 299)
⑩ Cambiar a la vista del portal (Página 284)
⑪ Barra de editores
⑫ Barra de estado con indicador de progreso
⑬ Task Cards (Página 301)

Barra de título

En la barra de título se visualiza el nombre del proyecto.

Barra de menús

En la barra de menús se encuentran todos los comandos necesarios para trabajar con el software.

Barra de herramientas

La barra de herramientas contiene botones de comando que ofrecen acceso directo a los comandos mas frecuentes. De esta manera es posible acceder más rápidamente a los comandos.

Divisor de ventana

Los divisores de ventanas dividen los diferentes elementos de la interfaz de usuario. Las flechas que aparecen sobre los divisores permiten mostrar y ocultar las partes colindantes de la interfaz.

Cambiar a la vista del portal

El enlace "Vista del portal" permite cambiar a la vista del portal.

Barra de editores

En esta barra se visualizan los editores abiertos. Si hay muchos editores abiertos, se representarán agrupados. La barra de editores permite cambiar rápidamente entre los elementos abiertos.

Barra de estado con indicador de progreso

En la barra de estado encontrará el indicador de progreso de los procesos que se están realizando actualmente en segundo plano. Ésta incluye una barra que muestra gráficamente el progreso del proceso en curso. Posicione brevemente el puntero del ratón sobre la barra de progreso y aparecerá un tooltip con información adicional sobre los procesos que se están realizando actualmente en segundo plano. Los procesos en segundo plano se pueden cancelar con el botón ubicado junto a la barra de progreso.

Si actualmente no se están ejecutando procesos en segundo plano, en la barra de estado se visualiza el último mensaje generado.

Consulte también

Principios básicos del área de trabajo (Página 291)

6.1.2.4 Vista de librerías

Función de la vista de librerías

La vista de librerías ofrece una vista general de los elementos de la librería del proyecto y de las librerías globales abiertas. En la Task Card "Librerías" se puede cambiar a la vista de librerías.

Consulte también: Sinopsis de la vista de librerías (Página 432)

6.1.2.5 Árbol del proyecto

Función del árbol del proyecto

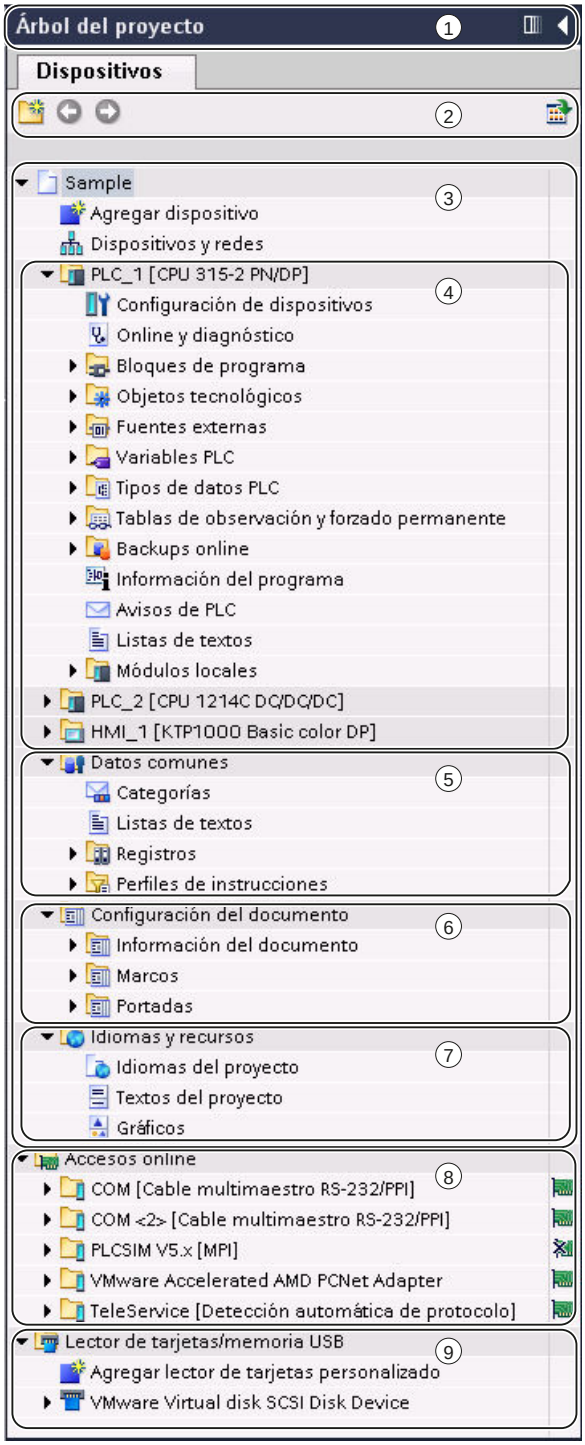
A través del árbol del proyecto es posible acceder a todos los componentes y datos del proyecto. En el árbol del proyecto pueden realizarse p. ej. las siguientes acciones:

- Agregar componentes adicionales
- Editar componentes existentes
- Visualizar y modificar las propiedades de los componentes existentes

Los objetos del árbol del proyecto se pueden seleccionar con el ratón o con el teclado introduciendo la letra inicial del objeto en cuestión. Si existen varios objetos que comienzan con la misma letra, se selecciona el siguiente objeto en orden descendente. El árbol del proyecto debe ser el elemento actual de la interfaz para poder seleccionar los objetos por su letra inicial.

Estructura del árbol del proyecto

La figura siguiente muestra un ejemplo de los componentes del árbol del proyecto:



① Barra de título
② Barra de herramientas

③ Proyecto
④ Dispositivos
⑤ Datos comunes
⑥ Configuración de la documentación
⑦ Idiomas y recursos
⑧ Accesos online
⑨ Card Reader/Memoria USB

Barra de título

La barra de título del árbol del proyecto incluye botones que permiten contraer el árbol de forma manual o automática. Tras contraerlo manualmente, el botón "Contraer" se posiciona en el margen izquierdo. La flecha hacia la izquierda se convierte en una flecha hacia la derecha, permitiendo expandir nuevamente el árbol del proyecto. El botón "Contraer automáticamente" permite cerrar el árbol del proyecto automáticamente cuando éste no se necesita.

Consulte también: Maximizar y minimizar el área de trabajo (Página 293)

Barra de herramientas

Mediante la barra de herramientas del árbol del proyecto pueden realizarse las siguientes acciones:

- Crear una nueva carpeta de usuario, p. ej. para agrupar bloques en la carpeta "Bloques de programa".
- Navegar hasta el origen de un enlace y viceversa
  Para los enlaces del árbol del proyecto se dispone de dos botones de comando que permiten navegar desde el enlace hasta el origen y viceversa.
- Mostrar una vista general del objeto en el área de trabajo
  Cuando se visualiza la vista general, se ocultan todos los objetos subordinados y las acciones de los elementos en el árbol del proyecto.

Proyecto

La carpeta "Proyecto" contiene todos los objetos y acciones relevantes para el proyecto, tales como:

- Dispositivos
- Idiomas y recursos
- Accesos online

Dispositivo

Todo dispositivo del proyecto tiene una carpeta propia con su nombre interno dentro del proyecto. En esta carpeta están estructurados todos los objetos y acciones que pertenecen al dispositivo.

Datos comunes

En esta carpeta se encuentran los datos que pueden utilizarse para varios dispositivos, tales como categorías, informes, scripts y listas de textos comunes.

Configuración de la documentación

En esta carpeta se define el formato de la impresión posterior de la documentación del proyecto.

Idiomas y recursos

En esta carpeta se definen los idiomas y textos del proyecto.

Accesos online

En esta carpeta se encuentran todas las interfaces de la PG/del PC, aunque no se utilicen para la comunicación con un módulo.

Card Reader/Memoria USB

En esta carpeta se gestionan todos los lectores de tarjetas conectados a la PG/al PC, además de otros medios de almacenamiento USB.

Consulte también

Vista del portal (Página 284)

Vista del proyecto (Página 285)

Principios básicos del área de trabajo (Página 291)

Ventana de inspección (Página 299)

Principios básicos de las Task Cards (Página 301)

Vista detallada (Página 305)

6.1.2.6 Área de trabajo

Principios básicos del área de trabajo

Función del área de trabajo

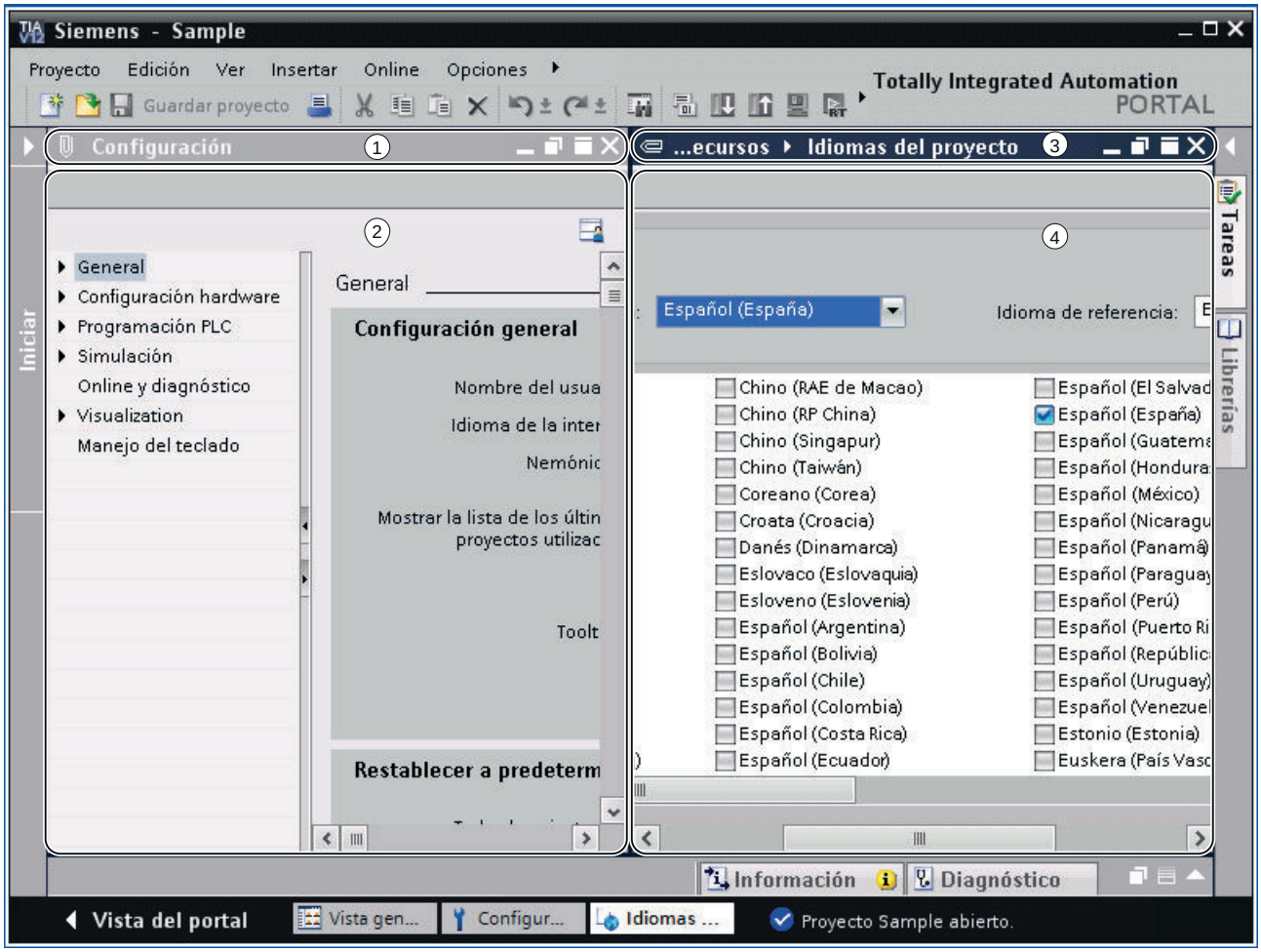
En el área de trabajo se visualizan los objetos que se abren para editarlos. Estos objetos son p. ej.:

- Editores y vistas
- Tablas

Es posible abrir varios objetos, pero en el área de trabajo generalmente sólo se visualiza uno. Todos los demás objetos se visualizan en forma de fichas en la barra de editores. Si desea ver simultáneamente dos objetos para determinadas tareas, puede dividir el área de trabajo en sentido vertical u horizontal, o desacoplar elementos del área de trabajo. Si no está abierto ningún objeto, el área de trabajo aparecerá vacía.

Estructura del área de trabajo

La figura siguiente muestra un ejemplo de un área de trabajo dividida verticalmente:



① Barra de título del editor de la izquierda
② Área de trabajo del editor de la izquierda
③ Barra de título del editor de la derecha
④ Área de trabajo del editor de la derecha

Consulte también

Maximizar y minimizar el área de trabajo (Página 293)

Dividir el área de trabajo (Página 295)

Desacoplar los elementos del área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Guardar el diseño de la interfaz de usuario (Página 310)

Maximizar y minimizar el área de trabajo

Existe la posibilidad de ajustar el área de trabajo para disponer del mayor espacio posible. Para ello se puede utilizar la siguiente función:

- Maximizar el área de trabajo
  Las Task Cards, el árbol del proyecto y la ventana de inspección se pueden cerrar con un clic. Con ello se amplía el tamaño del área de trabajo. El área de trabajo se puede volver a minimizar en todo momento para regresar a la vista anterior.
- Contraer automáticamente Task Cards, el árbol del proyecto y la ventana de inspección
  Para las Task Cards, el árbol del proyecto y la ventana de inspección se puede utilizar la opción "Contraer automáticamente". Esto hace que se contraigan estas ventanas si no se necesitan.

Maximizar y minimizar el área de trabajo

Para maximizar el área de trabajo, proceda del siguiente modo:

1. Abra un elemento, p. ej. un editor o una tabla.
   El elemento aparece en el área de trabajo.
2. Haga clic en el botón "Maximizar" de la barra de título del elemento.
   Las Task Cards, el árbol del proyecto y la ventana de inspección se contraen y el área de trabajo se representa en su tamaño máximo.

Para minimizar el área de trabajo, proceda del siguiente modo:

1. Haga clic en el botón "Incrustar" de la barra de título del elemento visualizado.
   Obtendrá nuevamente la vista que tenía antes de maximizar el área de trabajo. Es decir, si las Task Cards, el árbol del proyecto y la ventana de inspección estaban desplegadas, volverán a desplegarse.

Contraer automáticamente Task Cards, el árbol del proyecto y la ventana de inspección

Para contraer automáticamente las Task Cards, proceda del siguiente modo:

1. Haga clic en "Contraer automáticamente" en la barra de título de las Task Cards.
   Las Task Cards se contraen al hacer clic con el ratón en un área fuera de las Task Cards.
2. Para utilizar las Task Cards, haga clic en las Task Cards contraídas.
3. Las Task Cards se despliegan y se pueden volver a utilizar. La opción "Contraer automáticamente" permanece activada.

Para contraer automáticamente el árbol del proyecto, proceda del siguiente modo:

1. Haga clic en "Contraer automáticamente" en la barra de título del árbol del proyecto.
   El árbol del proyecto se contrae al hacer clic con el ratón en un área fuera del árbol del proyecto.
2. Para utilizar el árbol del proyecto, haga clic en el árbol del proyecto contraído.
   El árbol del proyecto se despliega y se puede volver a utilizar. La opción "Contraer automáticamente" permanece activada.

Para contraer automáticamente la ventana de inspección, proceda del siguiente modo:

1. Haga clic en "Contraer automáticamente" en la barra de título de la ventana de inspección.
   La ventana de inspección se contrae al hacer clic con el ratón en un área fuera de la ventana de inspección.
2. Para utilizar la ventana de inspección, haga clic en la ventana de inspección contraída.
   La ventana de inspección se despliega y se puede volver a utilizar. La opción "Contraer automáticamente" permanece activada.

Para desactivar la función "Contraer automáticamente", proceda del siguiente modo:

1. Haga clic en "Expandir permanentemente" en la ventana correspondiente.
   La opción "Contraer automáticamente" se desactiva y la ventana permanece abierta.

Consulte también

Principios básicos del área de trabajo (Página 291)

Dividir el área de trabajo (Página 295)

Desacoplar los elementos del área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Dividir el área de trabajo

El área de trabajo se puede dividir vertical u horizontalmente.

Procedimiento

Para dividir el área de trabajo vertical u horizontalmente, proceda del siguiente modo:

1. Elija el comando "Dividir el área de editores verticalmente" o "Dividir el área de editores horizontalmente" del menú "Ventana".
   El elemento en el que ha hecho clic y el siguiente elemento de la barra de editores se visualizan uno junto al otro, o bien uno debajo del otro.

Nota

Si en el área de trabajo no hay ningún elemento abierto, los comandos "Dividir el área de editores verticalmente" y "Dividir el área de editores horizontalmente" no estarán disponibles

Consulte también

Principios básicos del área de trabajo (Página 291)

Maximizar y minimizar el área de trabajo (Página 293)

Desacoplar los elementos del área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Desacoplar los elementos del área de trabajo

Los elementos siguientes se pueden desacoplar del área de trabajo y abrir en una ventana propia:

- Editores
- Tablas
- Ventana de configuración
- Task Cards
- Ventana de inspección

Los elementos desacoplados se pueden volver a incrustar en el área de trabajo en todo momento.

Nota

Propiedades de elementos en una ventana desacoplada

Las propiedades de elementos seleccionados en una ventana desacoplada solo se muestran en la ventana de inspección si esta también está desacoplada.

Desacoplar los elementos del área de trabajo

Para desacoplar elementos del área de trabajo, proceda del siguiente modo:

1. Haga clic en el botón "Soltar" de la barra de título del elemento.
   El elemento se desacopla del área de trabajo y se visualiza en una ventana propia. La ventana se puede posicionar entonces libremente. Si se minimiza la ventana, puede restablecerse desde la barra de editores.

Incrustar elementos en el área de trabajo

Para volver a incrustar elementos en el área de trabajo, proceda del siguiente modo:

1. Haga clic en el botón "Incrustar" de la barra de título del elemento.
   El elemento aparece nuevamente en el área de trabajo.

Consulte también

Principios básicos del área de trabajo (Página 291)

Maximizar y minimizar el área de trabajo (Página 293)

Dividir el área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Uso de elementos agrupados del área de trabajo

Si se abren más de cinco elementos del mismo tipo, p. ej. editores o tablas, éstos se agrupan en la barra de editores. Dichos grupos se pueden utilizar del siguiente modo:

- Mostrar elementos individuales de un grupo
- Mostrar todos los elementos de un grupo en ventanas individuales
- Incrustar todos los elementos visualizados de un grupo en el área de trabajo
- Minimizar todos los elementos visualizados
- Cerrar todos los elementos de un grupo

Mostrar elementos individuales de un grupo

Para mostrar elementos individuales de un grupo, proceda del modo siguiente:

1. En la barra de editores haga clic en el grupo en que se encuentra el elemento que desea visualizar.
   Se visualiza una lista con todos los elementos disponibles del grupo.
2. Haga clic en el elemento que desea visualizar.

Mostrar todos los elementos de un grupo en ventanas individuales

Para mostrar todos los elementos de un grupo en ventanas individuales, proceda del modo siguiente:

1. En la barra de editores haga clic con el botón derecho del ratón en el grupo cuyos elementos desea visualizar.
2. Elija el comando "Restablecer grupo" del menú contextual.
   Todos los elementos del grupo se visualizan en ventanas individuales que se superponen. Desplace las ventanas para ver los elementos individuales o seleccione un elemento por medio del grupo en la barra de editores.

Incrustar todos los elementos visualizados de un grupo en el área de trabajo

Para volver a incrustar en el área de trabajo todos los elementos de un grupo visualizados en ventanas individuales, proceda del modo siguiente:

1. En la barra de editores haga clic con el botón derecho del ratón en el grupo cuyos elementos desea incrustar.
2. Elija el comando "Acoplar grupo" del menú contextual.
   Todos los elementos del grupo se vuelven a incrustar en el área de trabajo.

Minimizar todos los elementos visualizados

Para minimizar todos los elementos de un grupo, proceda del siguiente modo:

1. En la barra de editores haga clic con el botón derecho del ratón en el grupo cuyos elementos desea minimizar.
2. Elija el comando "Minimizar grupo" del menú contextual.
   Todos los elementos del grupo se minimizan. Aunque se minimicen, los elementos continúan abiertos y pueden maximizarse rápidamente por medio del grupo.

Cerrar todos los elementos de un grupo

Para cerrar todos los elementos de un grupo, proceda del siguiente modo:

1. En la barra de editores haga clic con el botón derecho del ratón en el grupo cuyos elementos desea cerrar.
2. Elija el comando "Cerrar grupo" del menú contextual.
   Todos los elementos del grupo se cierran. El grupo desaparece.

Consulte también

Principios básicos del área de trabajo (Página 291)

Maximizar y minimizar el área de trabajo (Página 293)

Dividir el área de trabajo (Página 295)

Desacoplar los elementos del área de trabajo (Página 295)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Minimizar y maximizar los elementos del área de trabajo

Si es necesario, se pueden minimizar los elementos abiertos del área de trabajo (p. ej. los editores o tablas). Aunque se minimicen, continúan abiertos y pueden maximizarse rápidamente desde la barra de editores.

Minimizar los elementos del área de trabajo

Para minimizar elementos del área de trabajo, proceda del siguiente modo:

1. Haga clic en el botón "Minimizar" de la barra de título del elemento.
   El elemento se minimiza y siguedisponible a través de la barra de editores.

Para minimizar todos los elementos simultáneamente, proceda del siguiente modo:

1. Elija el comando "Minimizar todos" del menú "Ventana".

Maximizar los elementos del área de trabajo

Para volver a maximizar elementos del área de trabajo, proceda del siguiente modo:

1. Haga clic en el elemento deseado en la barra de editores.
   El elemento se maximiza y se visualiza en el área de trabajo.

Consulte también

Principios básicos del área de trabajo (Página 291)

Maximizar y minimizar el área de trabajo (Página 293)

Dividir el área de trabajo (Página 295)

Desacoplar los elementos del área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Cambiar entre los elementos del área de trabajo

En todo momento es posible cambiar entre los distintos elementos del área de trabajo.

Cambiar entre los elementos del área de trabajo

Para cambiar al editor anterior o siguiente, proceda del siguiente modo:

1. Elija el comando "Editor siguiente" o "Editor anterior" del menú "Ventana".
   Se abre el editor siguiente o anterior, respectivamente.

Consulte también

Principios básicos del área de trabajo (Página 291)

Maximizar y minimizar el área de trabajo (Página 293)

Dividir el área de trabajo (Página 295)

Desacoplar los elementos del área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Guardar la configuración de ventana de los editores y tablas (Página 312)

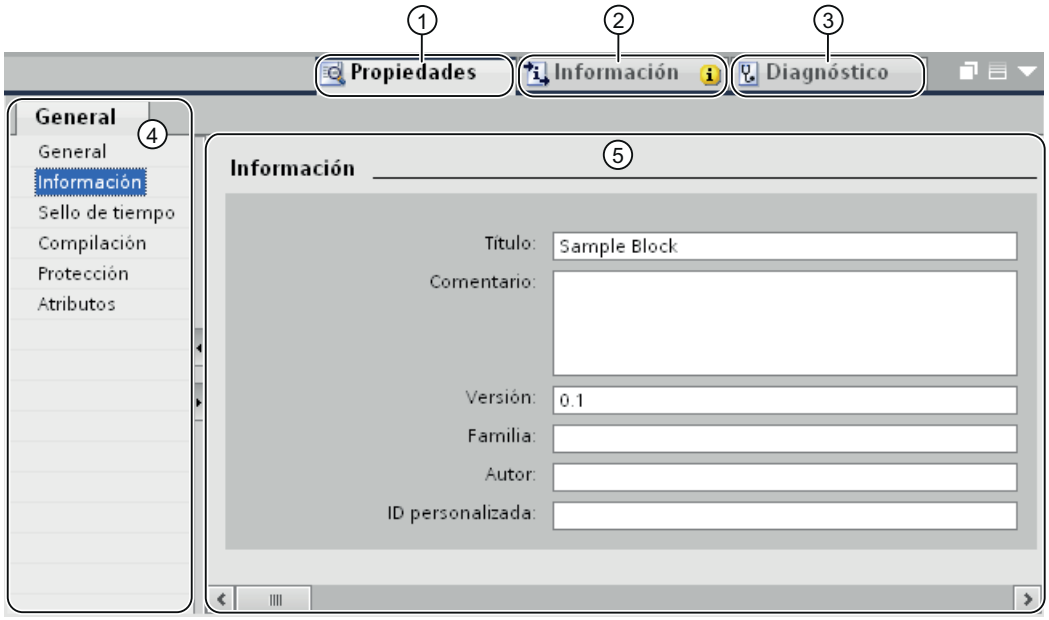
6.1.2.7 Ventana de inspección

Función de la ventana de inspección

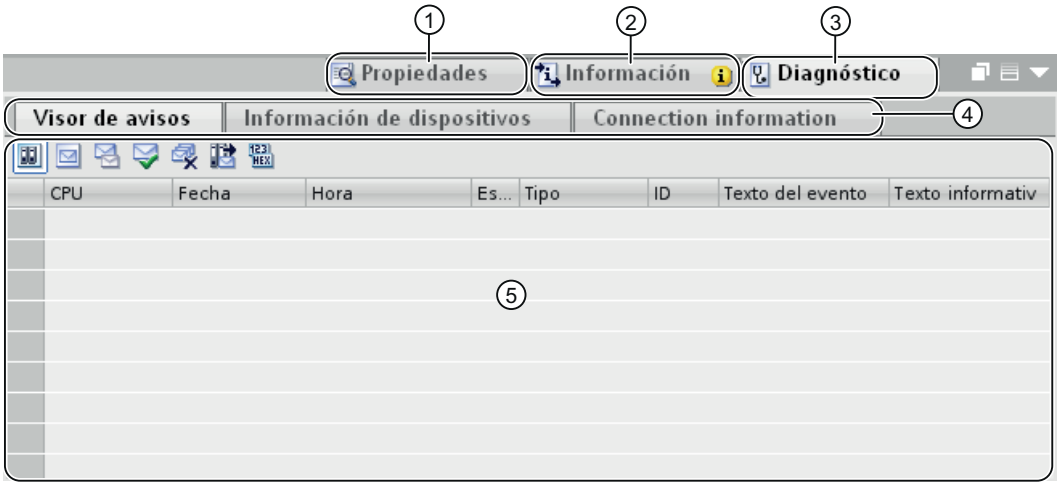
En la ventana de inspección se visualiza información adicional acerca de un objeto seleccionado o sobre las acciones realizadas.

Estructura de la ventana de inspección

Las figuras siguientes muestran los componentes de la ventana de inspección:



① Ficha "Propiedades"
② Ficha "Información"
③ Ficha "Diagnóstico"
④ Navegación local en la ficha "Propiedades"
⑤ Contenido de la ficha "Configuración"



① Ficha "Propiedades"
② Ficha "Información"
③ Ficha "Diagnóstico"
④ Navegación dentro de la ficha (solo disponible en las fichas "Información" y "Diagnóstico")
⑤ Contenido de la ficha "Diagnóstico"

Ficha "Propiedades"

En esta ficha se visualizan las propiedades del objeto seleccionado. Las propiedades editables pueden modificarse aquí.

Ficha "Información"

En esta ficha se visualiza información adicional acerca del objeto seleccionado y mensajes sobre las acciones realizadas (p. ej. compilar).

Ficha "Diagnóstico"

Esta ficha proporciona información sobre los eventos de diagnóstico del sistema, los eventos de aviso configurados y el diagnóstico de conexión.

Navegación en las fichas

Mediante la navegación local y las fichas subordinadas es posible visualizar la información deseada en las distintas fichas.

Consulte también

Árbol del proyecto (Página 288)

Principios básicos del área de trabajo (Página 291)

Vista del portal (Página 284)

Vista del proyecto (Página 285)

Principios básicos de las Task Cards (Página 301)

Vista detallada (Página 305)

6.1.2.8 Task Cards

Principios básicos de las Task Cards

Función de las Task Cards

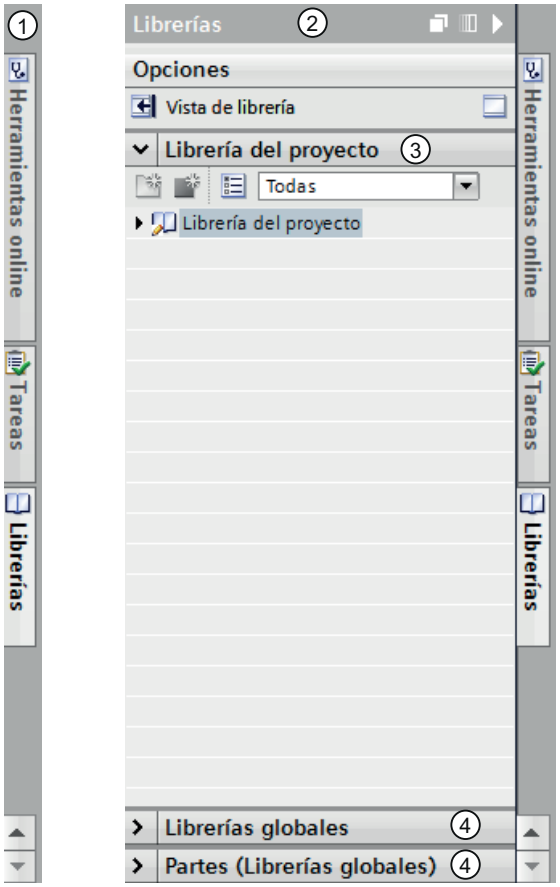
Dependiendo del objeto editado o seleccionado se dispone de Task Cards con las que pueden realizarse acciones adicionales, tales como:

- Seleccionar objetos de una librería o del catálogo de hardware
- Buscar y reemplazar objetos en el proyecto
- Arrastrar objetos predefinidos hasta el área de trabajo

Las Task Cards disponibles se encuentran en una barra en el borde derecho de la pantalla. Se pueden expandir y contraer en todo momento. Las Task Cards disponibles dependen de los productos instalados. Las Task Cards complejas están subdivididas en paletas que, a su vez, también se pueden expandir y contraer.

Estructura de las Task Cards

La figura siguiente muestra un ejemplo de la barra de Task Cards:



① Task Cards cerradas
② Task Card abierta
③ Paleta abierta de una Task Card
④ Paleta cerrada de una Task Card

Consulte también

Cambiar modo de paleta (Página 303)

Árbol del proyecto (Página 288)

Principios básicos del área de trabajo (Página 291)

Ventana de inspección (Página 299)

Vista del portal (Página 284)

Vista del proyecto (Página 285)

Vista detallada (Página 305)

Cambiar modo de paleta

Existen dos modos de paleta distintos:

- Modo monopaleta:
  Nunca hay más de una paleta abierta. Si abre otra paleta, la paleta que fue abierta previamente se cierra automáticamente.
- Modo multipaleta:
  Se pueden abrir simultáneamente varias paletas.

Procedimiento

Para cambiar el modo de paleta, proceda del siguiente modo:

1. Haga clic dentro de una Task Card por encima de las paletas en el botón "Cambiar modo de paleta".

Consulte también

Principios básicos de las Task Cards (Página 301)

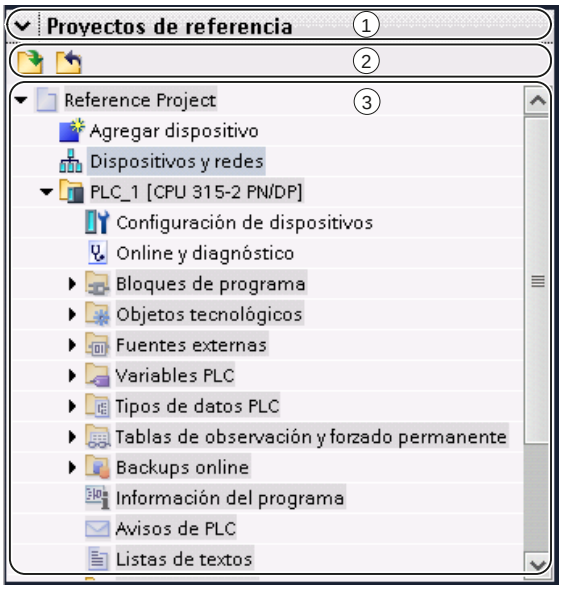
6.1.2.9 Proyectos de referencia

Función de proyectos de referencia

En la paleta "Proyectos de referencia" se pueden abrir proyectos adicionales al actual. Los proyectos de referencia se abren con protección contra escritura y no pueden editarse. No obstante, sí es posible arrastrar los objetos de un proyecto de referencia al proyecto actual y editarlos en él. También es posible comparar los objetos de un proyecto de referencia con los objetos del proyecto actual.

Estructura de la paleta "Proyectos de referencia"

La figura siguiente muestra la estructura de la paleta "Proyectos de referencia":



① Barra de título

② Barra de herramientas

③ Proyectos de referencia abiertos

Barra de título

En la barra de título de la paleta "Proyectos de referencia" hay una flecha que permite cerrarla. Tras cerrar la vista detallada, la flecha hacia abajo se convierte en una flecha hacia la derecha, que permite abrir nuevamente la paleta.

Barra de herramientas

En la barra de herramientas hay botones para abrir y cerrar proyectos de referencia.

Proyectos de referencia abiertos

Los proyectos de referencia abiertos se muestran protegidos contra escritura con los objetos y la estructura jerárquica correspondiente.

Consulte también

Principios básicos de los proyectos de referencia (Página 360)

Apertura y cierre de un proyecto de referencia (Página 361)

6.1.2.10 Vista detallada

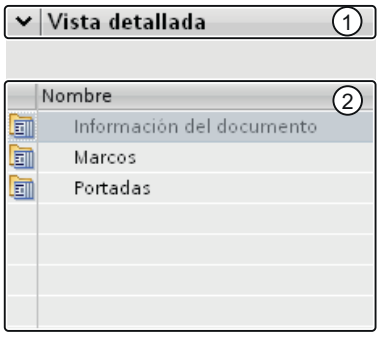
Función de la vista detallada

En la vista detallada se visualizan determinados contenidos de un objeto seleccionado, ya sea en la ventana sinóptica o en el árbol del proyecto. Los contenidos posibles son p. ej. listas de textos o variables.

En cambio, los contenidos de carpetas no se visualizan. Para visualizar el contenido de carpetas, utilice el árbol del proyecto o la ventana sinóptica.

Estructura de la vista detallada

La figura siguiente muestra un ejemplo de la vista detallada:



① Barra de título
② Contenidos del objeto seleccionado

Barra de título

En la barra de título de la vista detallada hay una flecha que permite cerrarla. Tras cerrar la vista detallada, la flecha hacia abajo se convierte en una flecha hacia la derecha, que permite abrir nuevamente esta vista.

Objetos

Los contenidos visualizados varían según el objeto seleccionado. Los contenidos de los objetos se pueden arrastrar mediante Drag & Drop directamente desde la vista detallada hasta la ubicación deseada.

Consulte también

Árbol del proyecto (Página 288)

Principios básicos del área de trabajo (Página 291)

Ventana de inspección (Página 299)

Principios básicos de las Task Cards (Página 301)

Vista del portal (Página 284)

Vista del proyecto (Página 285)

6.1.2.11 Ventana sinóptica

Ventana sinóptica

Funciones de la ventana general

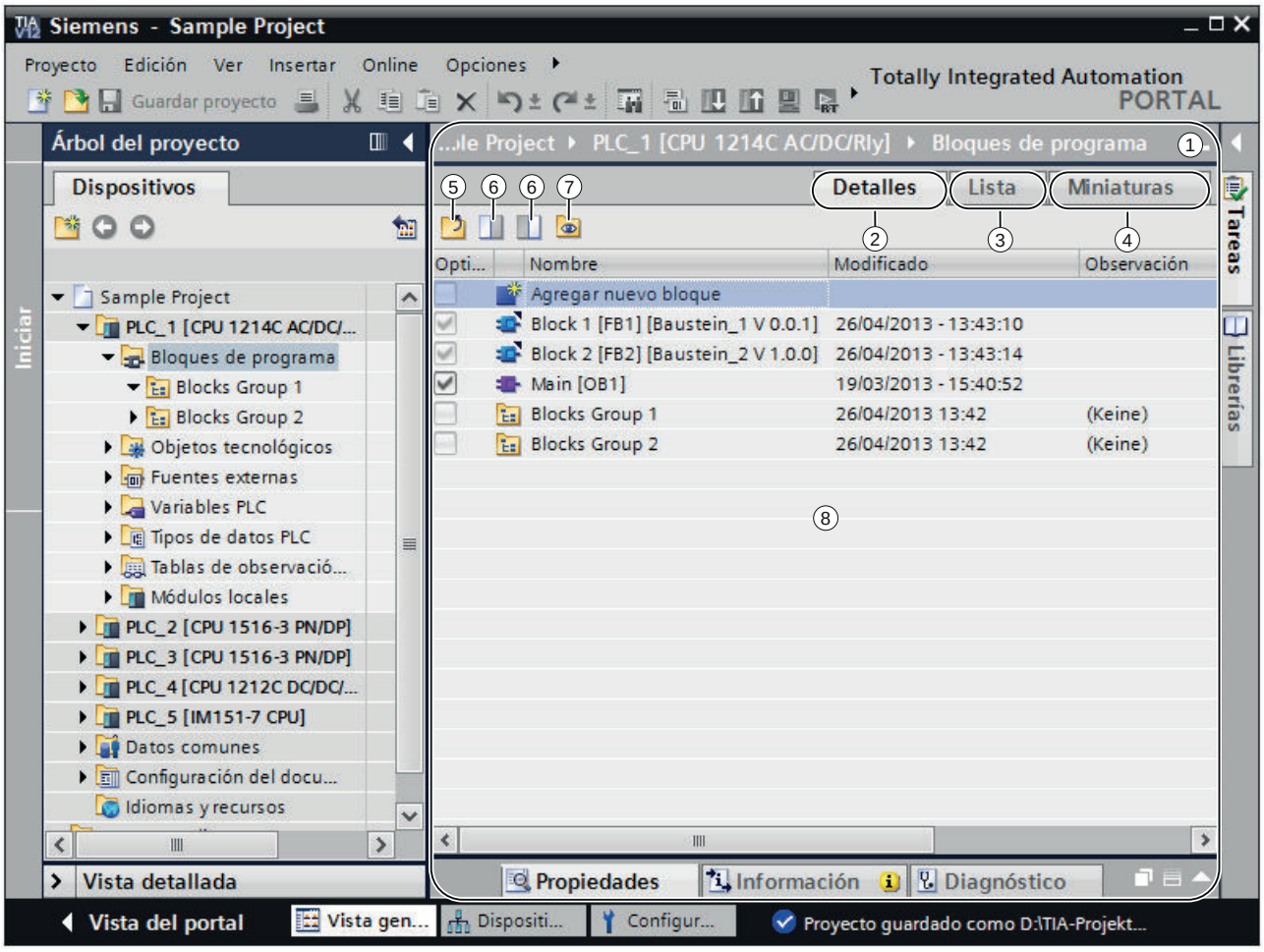
La ventana general complementa el árbol del proyecto. La ventana general muestra los contenidos de la carpeta seleccionada actualmente en el árbol del proyecto.

Además, en la ventana general pueden realizarse las siguientes acciones:

- Abrir objetos
- Visualizar y editar las propiedades de objetos en la ventana de inspección
- Cambiar el nombre de objetos
- Acceder a acciones específicas del objeto a través del menú contextual
- Contrastar objetos
- Ejecutar diferentes operaciones con objetos, p. ej. insertar objetos desde la librería mediante Drag & Drop, desplazar, copiar, pegar y borrar objetos

Estructura de la ventana general

La figura siguiente muestra los componentes de la ventana general:



① Ventana general

② Cambiar a la vista detallada

③ Cambiar a la vista de lista

④ Cambiar a la vista de iconos

⑤ Cambiar al nivel de orden superior

⑥ Dividir la ventana general en dos páginas. Se sincroniza la mitad derecha o izquierda de la ventana general. Si se vuelve a hacer clic, la división desaparece.

⑦ Se muestran todos los elementos incluidos en una carpeta seleccionada aunque se encuentren en grupos subordinados. Esta opción solo está disponible en la vista detallada.

⑧ Contenido del objeto que aparece seleccionado en el árbol del proyecto.

Formas de representación de la ventana general

Los contenidos de la ventana general pueden representarse como se indica a continuación:

- Vista detallada
  Los objetos se visualizan en una lista con información adicional (p. ej. la fecha de modificación).
- Vista de lista
  Los objetos se visualizan en una lista sencilla.
- Vista de iconos
  Los objetos se muestran en forma de iconos y ordenados por categorías.

Consulte también

Comparar objetos en la ventana general (Página 308)

Clasificar la vista detallada de la ventana general (Página 309)

Sinopsis de la vista de librerías (Página 432)

Comparar objetos en la ventana general

En la ventana sinóptica se pueden contrastar los contenidos de dos carpetas o dos objetos y compararlos. La ventana sinóptica se divide por la mitad y en el lado izquierdo y derecho se visualizan contenidos distintos, respectivamente.

También es posible desplazar objetos entre las ventanas divididas mediante Drag & Drop. Así por ejemplo es posible desplazar contenidos de una carpeta a la otra.

Procedimiento

Para dividir la ventana sinóptica por la mitad, o para volver a eliminar la división, proceda del siguiente modo:

1. En la barra de herramientas haga clic en el botón "El lado izquierdo se adapta a la selección" o "El lado derecho se adapta a la selección" para dividir la ventana sinóptica. De esta forma, el lado izquierdo o el lado derecho (según corresponda) de la ventana sinóptica se sincroniza con el contenido del objeto seleccionado en el árbol del proyecto.
2. Para que desaparezca la división, haga clic de nuevo en el botón seleccionado antes.

Consulte también

Ventana sinóptica (Página 306)

Clasificar la vista detallada de la ventana general

En la vista detallada de la ventana general se dispone de varias posibilidades de adaptar la representación:

- Insertar columnas adicionales
  Con el fin de aumentar la claridad, algunas columnas están ocultas de forma estándar. Las columnas ocultas pueden visualizarse en caso necesario. Las columnas disponibles dependen del objeto seleccionado.
- Mostrar el contenido de las carpetas en jerarquía plana
  El contenido de las carpetas puede visualizarse en una jerarquía plana. Todos los contenidos se muestran a la vez aunque se encuentren en grupos distintos.
- Clasificar las columnas de la tabla
  Es posible clasificar las columnas de la tabla en orden ascendente o descendente.

Mostrar u ocultar columnas

Para mostrar u ocultar columnas adicionales, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la barra de título de la tabla.
2. Elija el comando "Mostrar/ocultar" del menú contextual y seleccione las columnas que desea visualizar.

Mostrar el contenido de las carpetas en jerarquía plana

Para ver el contenido de una carpeta en jerarquía plana, proceda del siguiente modo:

1. Seleccione la carpeta deseada en el árbol del proyecto o en el navegador de librerías de la vista de librerías.
2. Haga clic en el botón "Mostrar elementos subordinados" de la barra de herramientas.
   Todos los elementos se muestran a la vez en la tabla aunque se encuentren en carpetas subordinadas.

Clasificar una tabla en orden ascendente o descendente

Para clasificar una tabla en orden ascendente o descendente a partir de una columna, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna para clasificarla en orden ascendente.
2. Vuelva a hacer clic en el encabezado de la misma columna para clasificarla en orden descendente.
3. Vuelva a hacer clic por tercera vez en el encabezado de la misma columna para anular la clasificación.

Consulte también

Ventana sinóptica (Página 306)

6.1.2.12 Diseño de la interfaz de usuario

Guardar el diseño de la interfaz de usuario

Posibilidades para asegurar el diseño de la interfaz de usuario

Si se adopta un cambio en la interfaz de usuario, éste también se conservará tras reiniciar el TIA Portal. Por ejemplo, el desplazamiento de una ventana o la adaptación del tamaño de un editor son cambios en el diseño de la interfaz de usuario.

Además del guardado automático del diseño de la interfaz de usuario, también existe la posibilidad de asegurar manualmente determinadas disposiciones:

- Guardar diseño de ventana
  Se pueden guardar manualmente las disposiciones de ventanas y editores del TIA-Portal y restaurarlas más adelante. Por medio de una combinación de teclas se puede acceder a cinco diseños de ventana.
  Esta función se puede utilizar, por ejemplo, si está trabajando con un PC portátil y necesita complementarlo con un monitor externo. Así, puede crear un diseño de ventana para su uso móvil en la pantalla del PC portátil y otro para trabajar en la oficina con un monitor externo.
- Guardado de la disposición dentro de los editores
  En algunos editores puede adaptarse la representación. Por ejemplo, se puede adaptar el ancho de las tablas o bien mostrar u ocultar las distintas columnas de una tabla.

Consulte también

Guardar diseño de ventana (Página 310)

Cargar diseño de ventana (Página 311)

Administrar diseños de ventana (Página 312)

Guardar la configuración de ventana de los editores y tablas (Página 312)

Restablecer diseño de interfaz de usuario (Página 313)

Principios básicos del área de trabajo (Página 291)

Guardar diseño de ventana

Puede guardar el diseño de ventana actual para volver a abrirlo más adelante de la misma forma.

Procedimiento

Para guardar un diseño de ventana, proceda del siguiente modo:

1. Disponga todas las ventanas en el modo en que desee guardarlas.
2. Elija el comando "Guardar diseño de ventana como" del menú "Ventana".
   Se abre el cuadro de diálogo "Guardar diseño de ventana".
3. Introduzca en el campo "Nombre" un nombre para el diseño de ventana.

4. Introduzca una descripción del diseño de ventana en el campo "Descripción", para poder identificar fácilmente el diseño de ventana más adelante.
5. Haga clic en el botón "Guardar".

Resultado

El nuevo diseño de ventana se guardará en última posición, detrás del diseño de ventana ya existente. Se puede acceder a los cinco primeros diseños de ventana mediante una combinación de teclas.

Consulte también

Guardar el diseño de la interfaz de usuario (Página 310)

Cargar diseño de ventana

Si ya ha guardado un diseño de ventana, puede cargarlo y adaptar así rápidamente su entorno de trabajo a las circunstancias respectivas. Los cinco primeros diseños de ventana pueden cargarse mediante el acceso rápido en el menú "Ventana" o de una combinación de teclas.

Si ha guardado un diseño de ventana y a continuación ha realizado cambios en la disposición de la ventana, puede rehacer el diseño de ventana guardado con anterioridad.

Cargar diseños de ventana 1 a 5 mediante un acceso rápido

Para cargar uno de los cinco primeros diseños de ventana guardados, proceda del siguiente modo:

1. Seleccione un diseño de ventana en el menú "Ventana" o seleccione la combinación de teclas <Alt+Shift+[1 ... 5]>.

Cargar otros diseños de ventana

Para seleccionar un diseño de ventana que no se encuentre entre los cinco primeros, proceda del modo siguiente:

1. Elija el comando "Otros diseños de ventana" del menú "Ventana".
   Se abre el cuadro de diálogo "Administrar diseños de ventana".
2. Seleccione el diseño de ventana deseado.
3. Haga clic en "Aceptar".

Rehacer diseño de ventana

Para volver a un diseño de ventana guardado, proceda del siguiente modo:

1. Seleccione en el menú "Ventana" el comando "Rehacer diseño de ventana" o seleccione la combinación de teclas <Alt+Shift+0>.

Consulte también

Guardar el diseño de la interfaz de usuario (Página 310)

Administrar diseños de ventana

Puede realizar las siguientes acciones con los diseños de ventana ya existentes:

- Modificar el orden de los diseños de ventana
  El orden de los diseños de ventana es decisivo, ya que se puede acceder directamente a los cinco primeros diseños de ventana a través del menú "Ventana" y una combinación de teclas.
- Seleccione un diseño de ventana
  Si un diseño de ventana no se encuentra entre los cinco primeros diseños de ventana, puede acceder a éste a través de la ventana de diálogo "Administrar diseños de ventana".
- Borrado de diseños de ventana

Procedimiento

Para administrar diseños de ventana existentes, proceda del siguiente modo:

1. Elija el comando "Administrar diseños de ventana" del menú "Ventana".
   Se abre el cuadro de diálogo "Administrar diseños de ventana".
2. Seleccione el diseño de ventana que desea modificar.
3. Haga clic en los iconos "adelante" o "atrás" para desplazar el diseño de ventana hacia delante o hacia atrás.
4. Haga clic en el icono "Borrar" para borrar el diseño de ventana seleccionado.
5. Haga clic en "Aceptar".
   Se activará el diseño de ventana seleccionado.

Consulte también

Guardar el diseño de la interfaz de usuario (Página 310)

Guardar la configuración de ventana de los editores y tablas

Los editores y tablas se pueden personalizar. En las tablas es posible, p. ej., ocultar columnas que no se necesiten. La vista personalizada se puede guardar a continuación.

Procedimiento

Para guardar la configuración de ventana de los editores y tablas del área de trabajo, proceda del siguiente modo:

1. Personalice el editor o la tabla.
2. Haga clic en el botón "Guardar configuración de ventana" en el editor o en la tabla.

Resultado

La configuración se guarda. A abrir nuevamente el editor o la tabla, se aplicará esta configuración.

Consulte también

Principios básicos del área de trabajo (Página 291)

Maximizar y minimizar el área de trabajo (Página 293)

Dividir el área de trabajo (Página 295)

Desacoplar los elementos del área de trabajo (Página 295)

Uso de elementos agrupados del área de trabajo (Página 296)

Minimizar y maximizar los elementos del área de trabajo (Página 298)

Cambiar entre los elementos del área de trabajo (Página 299)

Guardar el diseño de la interfaz de usuario (Página 310)

Restablecer diseño de interfaz de usuario

Todo cambio que se haga en el diseño de la interfaz de usuario se guarda. Por lo tanto, los cambios realizados vuelven a estar disponibles tras reiniciar el TIA Portal. Si por ejemplo se modifica la altura o el ancho de un editor de textos o se modifica la división de una tabla, los cambios se conservan de forma permanente y ya no es necesario repetir los ajustes realizados anteriormente.

No obstante, en algunos casos puede resultar conveniente restablecer los ajustes de diseño originales, p. ej. si otro usuario prefiere utilizar una división distinta de la interfaz de usuario.

Procedimiento

Para restablecer la configuración estándar de la interfaz de usuario, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General" en la navegación local.
3. En "Restablecer a predeterminado > Diseño de los editores" haga clic en el botón "Restablecer a predeterminado".

Resultado

Se restablecen los ajustes estándar de la interfaz de usuario.

Consulte también

Resumen de los ajustes del programa (Página 279)

Guardar el diseño de la interfaz de usuario (Página 310)

6.1.3 Manejo del teclado en el TIA Portal

6.1.3.1 Manejar el TIA Portal con el teclado

Existe la posibilidad de moverse por el TIA Portal utilizando el teclado, por ejemplo cuando no se dispone de ningún ratón. Además, es posible acceder a muchas funciones con una combinación de teclas. Encontrará una relación de todas las combinaciones de teclas en los ajustes del TIA Portal.

En los capítulos siguientes se explica cómo moverse por el TIA Portal utilizando el teclado, cómo editar objetos y adaptar el TIA Portal a las necesidades de cada uno.

Consulte también

Mostrar una relación de todas las combinaciones de teclas (Página 314)

6.1.3.2 Mostrar una relación de todas las combinaciones de teclas

Existe la posibilidad de visualizar una relación de todas las combinaciones de teclas.

Procedimiento

Para visualizar una relación de todas las combinaciones de teclas posibles, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Aparecen los ajustes del TIA Portal.
2. Abra la entrada "Manejo del teclado" en la navegación local.
   Aparece una relación de todas las combinaciones de teclas válidas para los productos instalados actualmente.

6.1.3.3 Funciones básicas del TIA Portal

A continuación se describe cómo manejar las funciones básicas del portal TIA únicamente con el teclado.

Manejo de las funciones básicas del TIA Portal con el teclado

La tabla siguiente muestra cómo acceder a las funciones básicas del TIA Portal con combinaciones de teclas:

Función	Combinación de teclas	Comando
Cambiar entre la vista del proyecto y la vista del portal	<Alt+F7>	
Abrir la Ayuda Para acceder a la ayuda del TIA Portal pulse <F1>.	<F1>	Ayuda > Mostrar ayuda
Cancelar la acción actual	<Esc>	
Buscar	<Ctrl+F>	
Reemplazar un objeto	<Ctrl+H>	
Buscar siguiente Si se ha iniciado una búsqueda, se puede saltar al siguiente punto de coincidencia con <F3>	<F3>	
Imprimir un objeto	<Ctrl+P>	Proyecto > Imprimir

Manejar menús

La tabla siguiente muestra cómo moverse por los menús con ayuda del teclado:

Función	Combinación de teclas
Iniciar el manejo de teclas dentro del menú Al menú se accede con la tecla <Alt>, y a continuación se puede navegar por él con las teclas de dirección. Para confirmar la elección de un comando de menú pulse <Intro>.	<Alt>
Saltar directamente a un menú determinado Se puede saltar directamente a un comando de menú determinado manteniendo pulsada la tecla <Alt>. En cada comando de menú hay una letra subrayada. Pulse la letra subrayada junto con <Alt>.	<Alt+letra subrayada en el respectivo menú>
Abrir el menú contextual de un objeto Con la tecla de menú contextual (en teclados compatibles con Microsoft Windows) se abre el menú contextual del objeto seleccionado. También es posible utilizar la combinación <Mayús +F10> en teclados no compatibles con Microsoft Windows. Utilice las teclas de dirección para navegar por el menú contextual y elija el comando de menú que desee con <Intro>.	<Tecla de menú contextual> Alternativa: <Mayús +F10>

Manejar elementos desplegables

La tabla siguiente muestra cómo manejar los elementos desplegables con el teclado:

Función	Combinación de teclas
Abrir una carpeta en un árbol Con <flecha a la derecha> se abre, por ejemplo, una carpeta en el árbol del proyecto.	<Flecha a la derecha>
Cerrar una carpeta en un árbol Con <flecha a la izquierda> se cierra, por ejemplo, una carpeta en el árbol del proyecto.	<Flecha a la izquierda>

Función	Combinación de teclas
Abrir una lista desplegable Las listas desplegables se abren con <F4> y a continuación se puede navegar por ellas con las teclas de dirección. Confirme la elección con <Intro>.	<F4>
Abrir la función de autocompletar	<Ctrl+barra espaciadora> <Ctrl+I>
Mostrar una selección de objetos	<Ctrl+J>

6.1.3.4 Uso de funciones de proyecto

Procesar el proyecto

Función	Combinación de teclas	Comando
Abrir un proyecto	<Ctrl+O>	Proyecto > Abrir
Cerrar un proyecto	<Ctrl+W>	Proyecto > Cerrar
Guardar un proyecto	<Ctrl+S>	Proyecto > Guardar
Guardar un proyecto con un nombre diferente	<Ctrl+Mayús+S>	Proyecto > Guardar como
Borrar un proyecto	<Ctrl+E>	Proyecto > Borrar proyecto
Imprimir un proyecto	<Ctrl+P>	Proyecto > Imprimir
Deshacer la última acción	<Ctrl+Z>	Edición > Deshacer
Rehacer la última acción	<Ctrl+Y>	Edición > Rehacer

Acceder a la Ayuda

Función	Combinación de teclas	Comando
Acceder a la Ayuda	<F1> o <Mayús+F1>	Ayuda > Mostrar ayuda

6.1.3.5 Organización de ventanas

A continuación se describe cómo abrir y cerrar ventanas del TIA Portal con el teclado o cómo trabajar con diseños de ventana guardados con ayuda del teclado.

Abrir y cerrar ventanas

La tabla siguiente muestra cómo abrir y cerrar ventanas con combinaciones de teclas:

Función	Combinación de teclas	Comando
Abrir/cerrar el árbol del proyecto	<Ctrl+1>	Ver > Árbol del proyecto
Abrir/cerrar la vista detallada	<Ctrl+4>	Ver > Vista detallada

Función	Combinación de teclas	Comando
Abrir/cerrar la vista general	<Ctrl+2>	Ver > Vista general
Abrir/cerrar una Task Card	<Ctrl+3>	Ver > Task Card
Abrir librerías	<Ctrl+Mayús+L>	
Abrir catálogo de hardware Si se encuentra en la vista de dispositivos o redes, se abre el catálogo de hardware.	<Ctrl+Mayús+C>	
Abrir/cerrar la ventana de inspección	<Ctrl+5>	Ver > Ventana de inspección
Abrir la ficha "Propiedades" de la ventana de inspección	<Ctrl+6>	
Abrir la ficha "Información" de la ventana de inspección	<Ctrl+7>	
Abrir la ficha "Diagnóstico" de la ventana de inspección	<Ctrl+8>	
Mostrar u ocultar proyectos de referencia	<Ctrl+9>	
Mostrar el teclado de pantalla Es posible visualizar un teclado de pantalla, p. ej. cuando se utilicen pantallas táctiles.	<Ctrl+Mayús+K>	
Cerrar todos los editores	<Ctrl+Mayús+F4>	Ventana > Cerrar todos

Utilizar diseños de ventana guardados

Se pueden guardar organizaciones de ventana concretas, y restablecerlas con posterioridad. La tabla siguiente muestra cómo acceder con combinaciones de teclas a diseños de ventana guardados:

Función	Combinación de teclas	Comando
Rehacer diseño de ventana activo Si se utiliza un diseño de ventana guardado y entretanto se han realizado cambios en la interfaz del programa, se puede restablecer el estado original del diseño de ventana activo con <Alt+Mayús+0>.	<Mayús+Alt+0>	Ventana > Rehacer diseño de ventana
Cargar diseño de ventana Con <Alt+Mayús+[número del diseño de ventana] se puede activar uno de los primeros cinco diseños de ventana guardados.	<Mayús+Alt+[número del diseño de ventana]>	Ventana > Diseño de ventana 1 a 5

6.1.3.6 Navegación por la interfaz del programa

El TIA Portal está dividido en diferentes áreas, como son distintas ventanas, barras de herramientas y editores. Para trabajar con el teclado dentro de un área de la interfaz, primero hay que activar el área en cuestión. A continuación se explica cómo activar las diferentes áreas de la interfaz con el teclado. Además se explica cómo moverse dentro de un área de la interfaz del TIA Portal utilizando el teclado.

Cambiar entre áreas de la interfaz y editores

La tabla siguiente muestra cómo cambiar entre las diferentes áreas de la interfaz del TIA Portal:

Función	Combinación de teclas
Cambiar entre las áreas de la interfaz en sentido horario La tecla <F6> permite cambiar entre las diferentes áreas de la interfaz del TIA Portal en sentido horario. El área activa en esos momentos tiene la barra de título azul. Si, por ejemplo, se encuentra en el árbol del proyecto y pulsa la tecla <F6>, pasará al editor que esté abierto actualmente. Si vuelve a pulsar <F6>, se activarán las Task Cards. Si, por el contrario, pulsa <Mayús+F6>, cambiará entre las áreas de trabajo en sentido antihorario.	<F6>
Cambiar entre las áreas de la interfaz en sentido antihorario Con <Mayús+F6> se cambia en sentido antihorario entre las áreas de la interfaz del TIA Portal.	<Mayús+F6>
Cambiar al siguiente editor abierto Con <Ctrl+Alt+flecha a la derecha> se cambia al siguiente editor abierto. Los editores abiertos pueden verse en la barra de editores.	<Ctrl+Alt+flecha a la derecha> Alternativa: <Ctrl+F6>
Cambiar al anterior editor abierto Con <Ctrl+Alt+flecha a la izquierda> se cambia al último editor abierto.	<Ctrl+Alt+flecha a la izquierda> Alternativa: <Ctrl+Mayús+F6>
Saltar a la parte superior del área de la interfaz Con <Mayús+Esc> se salta a la parte superior de la interfaz del programa. Si ha seleccionado, por ejemplo, un dispositivo en el árbol del proyecto y pulsa <Mayús+Esc>, se activará todo el árbol del proyecto.	<Mayús+Esc> Alternativa: <Alt+flecha hacia arriba>
Saltar a la parte inferior del área de la interfaz Con <Intro> se activa la parte inferior de la interfaz del programa. Si ha abierto p. ej. las propiedades de un dispositivo en la ventana de inspección para parametrizarlo, con <Intro> accederá a un nivel inferior en la interfaz del programa. A continuación se puede navegar al parámetro deseado con el tabulador.	<Intro> Alternativa: <Alt+flecha hacia abajo>

Navegar por áreas de la interfaz y editores

La tabla siguiente muestra cómo moverse por un área de la interfaz con el teclado:

Función	Combinación de teclas
Saltar al próximo elemento de un área de la interfaz La tecla de tabulación permite saltar de un elemento al siguiente dentro de un área de trabajo. Si, por ejemplo, ha abierto las propiedades de un dispositivo y desea saltar al próximo campo de entrada, pulse la tecla de tabulación. Al hacerlo se aplicarán los cambios que haya realizado en el campo de entrada actual.	<Tab>
Saltar al elemento anterior de un área de la interfaz Con <Mayús+Tab> se salta al elemento anterior dentro un área de trabajo, p. ej. a un campo de texto anterior. Al hacerlo se aplicarán los cambios que haya realizado en el campo de entrada actual.	<Mayús+Tab>
Cambiar a la siguiente ficha dentro un área de la interfaz Si un área de la interfaz está dividida en fichas, es posible cambiar entre las fichas con la combinación de teclas <Ctrl+Tab>. Así, por ejemplo, si se encuentra en la ficha "Propiedades" de la ventana de inspección y desea pasar a la ficha "Información", pulse la combinación de teclas <Ctrl+Tab>.	<Ctrl+Tab>

Función	Combinación de teclas
Cambiar a la ficha anterior Con <Ctrl+Mayús+Tab> se cambia a la última ficha abierta dentro del área de la interfaz.	<Ctrl+Mayús+Tab>
Saltar a la barra de herramientas de un editor Con la tecla <Alt+F10> se salta a la barra de herramientas de un editor. Si ha abierto p. ej. la vista preliminar y desea cambiar a la siguiente página de la impresión, pulse <Alt +F10>. A continuación, vaya con las teclas de dirección al botón correspondiente de la barra de herramientas y confirme la elección con <Intro>.	<Alt+F10>
Manejar las flechas de los divisores de ventanas para mostrar u ocultar elementos de la interfaz La tabla visualizada en el área de trabajo puede minimizarse y maximizarse. Vaya primero al área de trabajo y active una de las pequeñas flechas sobre la línea de separación situada encima de la tabla con ayuda del tabulador. Las flechas se activan en cuanto se muestran sobre fondo azul. A continuación pulse la barra espaciadora para minimizar o maximizar la tabla.	<Barra espaciadora>

6.1.3.7 Ajuste de editores

A continuación se explica cómo cambiar la organización de los editores con el teclado. Además se explica cómo elegir el tamaño de representación y la sección dentro de un editor gráfico.

Organización y ajuste de editores

La tabla siguiente muestra cómo organizar los editores abiertos vertical u horizontalmente y cómo cerrar el editor abierto:

Función	Combinación de teclas	Comando
Cerrar el editor activo	<Ctrl+F4>	
Dividir el área de editores verticalmente Si, por ejemplo, tiene abiertas la ventana general y la vista de redes y desea mostrarlas una al lado de otra, pulse la tecla <F12>.	<F12>	Ventana > Dividir el área del editor verticalmente
Dividir el área de editores horizontalmente Dos editores abiertos pueden ordenarse uno debajo del otro en el área de trabajo.	<Ctrl+F12>	Ventana > Dividir área del editor horizontalmente
Quitar la división de la ventana Si dos editores se han organizado horizontal o verticalmente en modo dividido en el área de trabajo, la división puede quitarse con <Alt+Mayús+F12>.	<Alt+Mayús+F12>	Ventana > Quitar división

Adaptar la representación en un editor

La tabla siguiente muestra cómo aumentar o reducir la representación dentro de editores gráficos y cómo mover la sección de un editor:

Función	Combinación de teclas
Aumentar progresivamente la representación en un editor Con <Ctrl> y la tecla <Más> del bloque numérico del teclado se aumenta la representación en el editor.	<Ctrl+más> Alternativa: <Ctrl+Flecha arriba>
Reducir progresivamente la representación en un editor Con <Ctrl> y la tecla <Menos> del bloque numérico del teclado se reduce la representación en el editor.	<Ctrl+Menos> Alternativa: <Ctrl+Flecha abajo>
Ajustar la vista 100% en el editor En un editor gráfico la vista actual se aumenta o reduce al 100% pulsando <Ctrl+0>.	<Ctrl+0>
Mover la sección del editor Manteniendo pulsada la barra espaciadora se puede mover con el ratón la sección visualizada de un editor.	<Barra espaciadora>

6.1.3.8 Editar objetos

Seleccionar objetos

La tabla siguiente muestra cómo seleccionar objetos individuales, como p. ej. dispositivos, en el árbol del proyecto:

Función	Combinación de teclas	Comando
Seleccionar un objeto ubicado a la izquierda o derecha, arriba o abajo	<Teclas de dirección>	
Saltar al primer objeto del área activa de la interfaz Se selecciona el primer objeto del área activa de la interfaz. En el caso del árbol del proyecto, por ejemplo, sería el primer nodo superior del proyecto.	<Inicio>	
Saltar al último objeto del área activa de la interfaz Se selecciona el último objeto dentro del área activa de la interfaz, p. ej. la última entrada del árbol del proyecto.	<Fin>	
Seleccionar todos los objetos de un área Se seleccionan todos los objetos del área de trabajo activa.	<Ctrl+A>	Edición > Seleccionar todo
Seleccionar varios objetos Para seleccionar varios objetos que no están situados directamente uno bajo el otro, primero hay que activar (recuadro gris alrededor de un objeto) el siguiente objeto que se desee seleccionar con <Ctrl+teclas de dirección>. De ese modo se mantiene la selección actual. A continuación pulse la barra espaciadora para seleccionar también el objeto activo. Repita el procedimiento hasta haber seleccionado todos los objetos deseados.	<Ctrl+teclas de dirección> + <barra espaciadora>	

Editar objetos

La tabla siguiente muestra una vista general de todas las combinaciones de teclas necesarias para editar objetos:

Función	Combinación de teclas	Comando
Insertar un objeto nuevo Dependiendo del contexto actual, se insertará un objeto nuevo. Si se encuentra, por ejemplo, en la vista de dispositivos, se abre el cuadro de diálogo "Agregar dispositivo", en el que se crea un dispositivo.	<Ctrl+N>	
Abrir un objeto	<Intro>	
Cambiar el nombre de un objeto	<F2>	Edición > Cambiar nombre
Copiar un objeto	<Ctrl+C> Alternativa: <Ctrl+Insert>	Edición > Copiar
Cortar un objeto	<Ctrl+X> Alternativa: <Mayús +Supr>	Edición > Cortar
Insertar un objeto	<Ctrl+V> Alternativa: <Mayús +Insert>	Edición > Pegar
Borrar un objeto	<Supr>	Edición > Borrar
Compilar un objeto	<Ctrl+B>	Edición > Compilar
Abrir las propiedades de un objeto Muchos objetos del TIA Portal disponen de propiedades editables. Pulse la combinación de teclas <Alt+Intro> para visualizar las propiedades de un objeto.	<Alt+Intro>	-

6.1.3.9 Edición de textos

A continuación se explica cómo utilizar las funciones de edición de texto solo con el teclado.

Editar textos

La tabla siguiente muestra funciones básicas para la edición de texto:

Función	Combinación de teclas
Cambiar entre los modos de inserción y sobrescritura	<Insert>
Finalizar el modo de edición	<Esc>
Borrar	<Supr>
Borrar caracteres	<Retroceso>
Confirmar entradas en un campo de entrada y salir del campo	<Intro>

Función	Combinación de teclas
Salto de línea en un campo de entrada de varias líneas En un campo de entrada de varias líneas hay que mantener pulsada la tecla <Mayús> para generar un salto de línea.	<Mayús+Intro>
Deshacer las entradas en un campo de entrada Si se encuentra en un campo de entrada y pulsa <Esc>, abandonará el campo de entrada sin que se apliquen los cambios realizados.	<Esc>

Navegación dentro de un área de texto

La tabla siguiente muestra cómo navegar dentro de un área de texto con el teclado:

Función	Combinación de teclas
Saltar al inicio de la línea	<Inicio>
Saltar al final de la línea	<Fin>
Saltar al inicio del texto	<Ctrl+Inicio>
Saltar al final del texto	<Ctrl+Fin>
Saltar a la página anterior	<RePág>
Saltar a la página siguiente	<AvPág>
Confirmar entradas en un campo de entrada y salir del campo	<Intro>
Salto de línea en un campo de entrada de varias líneas	<Mayús+Intro>
Deshacer las entradas en un campo de entrada Si se encuentra en un campo de entrada y pulsa <Esc>, abandonará el campo de entrada sin que se apliquen los cambios realizados.	<Esc>

Seleccionar texto

La tabla siguiente muestra cómo seleccionar texto con el teclado:

Función	Combinación de teclas
Ampliar selección a la palabra situada a izquierda o derecha Se selecciona el texto o la selección de texto hasta el final de la palabra. Si se encuentra al inicio o al final de una palabra, se selecciona la palabra anterior o posterior.	<Ctrl+Mayús+flecha a la izquierda o flecha a la derecha>
Ampliar selección hasta el inicio de la línea	<Mayús+Inicio>
Ampliar selección hasta el final de la línea	<Mayús+Fin>
Ampliar selección hasta el inicio del texto El texto se selecciona hasta el inicio o el final.	<Ctrl+Mayús+Inicio>
Ampliar selección hasta el final del texto El texto se selecciona hasta el inicio o el final.	<Ctrl+Mayús+Fin>

6.1.3.10 Editar tablas

A continuación se describe cómo navegar en tablas, editar campos y seleccionar partes de tablas con el teclado.

Comandos de teclado generales en tablas

La tabla siguiente muestra cómo editar tablas solo con el teclado:

Función	Combinación de teclas
Activar el modo de edición en una celda	<F2> o <Intro>
Confirmar una entrada y finalizar el modo de edición	<Intro>
Cancelar la edición y rechazar los cambios	<Esc>
Abrir la lista desplegable en una celda Con <F4> se abre la lista desplegable. A continuación se selecciona la entrada deseada con las teclas de dirección y la selección se confirma con <Intro>.	<F4>
Cerrar la lista desplegable en una celda y rechazar los cambios	<Esc>

Navegar en tablas

La tabla siguiente muestra cómo moverse por una tabla con el teclado:

Función	Combinación de teclas
Ir a la celda siguiente	<Teclas de dirección>
Ir a la siguiente celda editable hacia la derecha	<Tab>
Ir a la siguiente celda editable hacia la izquierda	<Mayús+Tab>
Retroceder una pantalla	<RePág>
Avanzar una pantalla	<AvPág>
Ir a la primera celda de la fila	<Inicio>
Ir a la última celda de la fila	<Fin>
Ir a la primera celda de la tabla	<Ctrl+Inicio>
Ir a la última celda de la tabla	<Ctrl+Fin>
Ir a la primera celda de la columna	<Ctrl+flecha hacia arriba>
Ir a la última celda de la columna	<Ctrl+flecha hacia abajo>

Seleccionar áreas en tablas

La tabla siguiente muestra cómo seleccionar áreas de una tabla con el teclado:

Función	Combinación de teclas
Seleccionar una columna	<Ctrl+barra espaciadora>
Seleccionar una fila	<Mayús+barra espaciadora>
Seleccionar todas las celdas	<Ctrl+A>
Ampliar la selección una celda	<Mayús+teclas con flecha>
Ampliar la selección una página hacia arriba	<Mayús+RePág>
Ampliar la selección una página hacia abajo	<Mayús+AvPág>
Ampliar la selección hasta la primera fila	<Ctrl+Mayús+flecha hacia arriba>
Ampliar la selección hasta la última fila	<Ctrl+Mayús+flecha hacia abajo>

Función	Combinación de teclas
Ampliar la selección hasta la primera celda de la fila	<Ctrl+Mayús+flecha a la izquierda>
Ampliar la selección hasta la última celda de la fila	<Ctrl+Mayús+flecha a la derecha>

6.1.3.11 Uso de funciones online

Control de funciones online con el teclado

La tabla siguiente muestra una vista general de las combinaciones de teclas con las que pueden utilizarse las funciones online del TIA Portal:

Función	Combinación de teclas	Comando
Establecer conexión online	<Ctrl+K>	Online > Establecer conexión online
Deshacer la conexión online	<Ctrl+M>	Online > Deshacer conexión online
Cargar los datos de proyecto en el dispositivo	<Ctrl+L>	Online > Cargar en dispositivo
Mostrar dispositivos accesibles Se abre un cuadro de diálogo en el que se muestran todos los dispositivos que están conectados con la interfaz PG/PC de la programadora o el PC.	<Ctrl+U>	Online > Mostrar dispositivos accesibles
Arrancar CPU Conmuta la CPU al estado operativo "RUN". Para ello es necesario que la CPU esté conectada online.	<Ctrl+Mayús+E>	Online > Arrancar CPU
Parar CPU Conmuta la CPU al estado operativo "STOP". Para ello es necesario que la CPU esté conectada online.	<Ctrl+Mayús+Q>	Online > Parar CPU
Iniciar simulación El hardware y el software del proyecto pueden probarse en un entorno online simulado sin necesidad de que los módulos estén realmente conectados online.	<Ctrl+Mayús+X>	Online > Simulación > Iniciar

6.1.3.12 Uso del teclado de pantalla

Introducción

Para el manejo del TIA Portal se dispone además del teclado de pantalla de Microsoft.

Mostrar el teclado de pantalla

Para mostrar el teclado de pantalla, proceda del siguiente modo:

1. Elija el comando "Teclado de pantalla" del menú "Ver".

Cerrar el teclado de pantalla

Para cerrar el teclado de pantalla, proceda del siguiente modo:

1. Elija el comando "Salir" del menú "Archivo" del teclado de pantalla.

6.1.4 Peculiaridades sujetas al sistema operativo

6.1.4.1 Influencia de los derechos de usuario

Restricciones debidas a derechos de usuario limitados

El software ofrece algunas funciones que permiten acceder directamente al hardware de la PG/del PC y, con ello, al sistema operativo instalado. Para poder aprovechar todas las funciones, el software debe trabajar estrechamente con el sistema operativo. Por tanto, se recomienda iniciar la sesión con suficientes derechos de usuario en el sistema operativo para lograr una interacción correcta.

Si se trabaja con derechos de usuario limitados, posiblemente no podrán utilizarse por completo algunas funciones que requieran una conexión online o que modifiquen la configuración de tarjetas de interfaz.

Reconocer funciones limitadas

Las funciones que requieren derechos especiales se reconocen por lo siguiente:

- Junto a la función figura el símbolo de un escudo.
  La función se puede manejar pero está reglamentada por el Control de cuentas de usuario (UAC).
- Un campo aparece atenuado y no se puede operar.
  Para poder editar este campo es indispensable tener derechos de administrador. Con algunos sistemas operativos es posible obtener rápidamente derechos de administrador introduciendo una contraseña de administrador.

Nota

El hecho de que un campo aparezca atenuado no está necesariamente ligado a la falta de derechos. Los tooltips en cascada proporcionan información adicional acerca de las condiciones de edición.

6.1.4.2 Ampliar los derechos de usuario

Suspender las restricciones debidas a derechos de usuario

Si no ha iniciado sesión con suficientes derechos de usuario en el sistema operativo, es posible que algunas funciones no se puedan ejecutar. Para suspender las restricciones existentes existen las siguientes posibilidades:

- Habilitar derechos avanzados mediante el Control de cuenta de usuarios (UAC) de Windows
- Iniciar sesión con derechos de administrador en el sistema operativo
- Utilizar derechos de administrador temporales

Obtener derechos avanzados mediante el Control de cuentas de usuario (UAC) de Windows

Para utilizar una función junto a la que aparezca el símbolo de escudo del Control de cuentas de usuario de Windows, proceda del siguiente modo:

1. Haga clic en el campo o en el botón de comando que tiene el símbolo de escudo.
   Aparece la consulta de seguridad del Control de cuentas de usuario de Windows.
2. Siga las instrucciones del Control de cuentas de usuario de Windows y, en caso necesario, introduzca la contraseña de administrador.

Entonces podrá utilizar la función una sola vez sin limitaciones.

Iniciar sesión con derechos de administrador en el sistema operativo

Para utilizar una función desactivada debido a la falta de derechos de usuario, proceda del siguiente modo:

1. Cierre el software.
2. Cierre la sesión en el sistema operativo.
3. Inicie sesión con derechos de administrador en el sistema operativo.
4. Reinicie el software.

Utilizar derechos de administrador temporales

Para adquirir derechos de administrador temporales, proceda del siguiente modo:

1. Haga clic en el botón de comando "Cambiar configuración". Este botón se encuentra en los diálogos que permiten asignar temporalmente derechos de administrador.
   Se abre un diálogo del sistema operativo para introducir una contraseña de administrador.
2. Introduzca la contraseña de administrador.

La configuración se puede modificar temporalmente. Repita el procedimiento si el cuadro se abre de nuevo.

Nota

No todos los sistemas operativos soportan esta función. Si el botón de comando "Cambiar configuración" no aparece o está atenuado, inicie sesión con derechos de administrador en el sistema operativo.

6.2 Ayuda del sistema de información

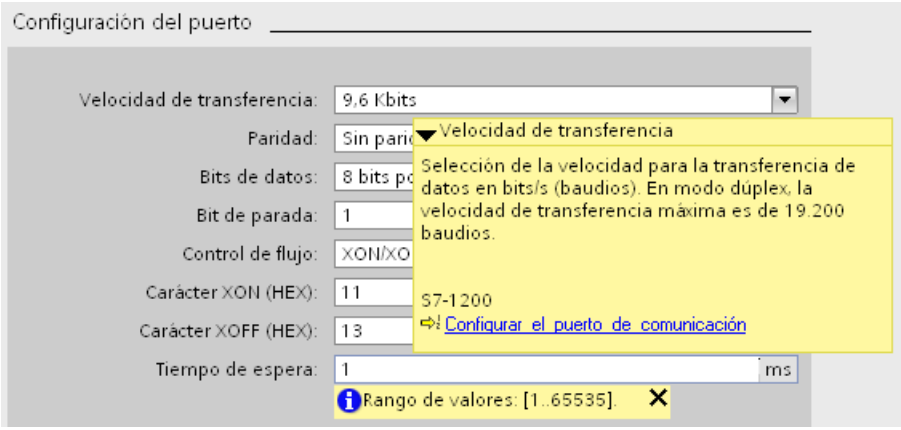
6.2.1 Generalidades acerca del sistema de información

Respuestas rápidas a sus preguntas

Para solucionar las tareas el TIA Portal incluye una ayuda completa, en la que se describen conceptos básicos, procedimientos y funciones. Otras ayudas disponibles al trabajar con el programa son:

- Roll-outs para entradas correctas en cuadros de diálogo
- Tooltips de información sobre los elementos de la interfaz de usuario, tales como campos de entrada, botones y símbolos. Algunos tooltips se complementan con textos en cascada que ofrecen información más detallada.
- Pulsando las teclas <F1> o <Mayús+F1> se obtiene ayuda acerca del contexto actual (p. ej. sobre comandos de menú).

La figura siguiente muestra un ejemplo de un tooltip con cascada (arriba) y de un roll-out (abajo):



Ayuda

En la ayuda suministrada se describen conceptos, procedimientos y funciones. Además, se incluyen ejemplos e información de referencia. La ayuda se abre en una ventana propia.

A la izquierda de la ventana de ayuda se muestra un área de navegación. El área de navegación se puede ocultar para conseguir más espacio en la pantalla. En el área de navegación dispone de las siguientes funciones:

- Índice de contenido
- Búsqueda en el índice
- Búsqueda de texto completo en toda la Ayuda
- Favoritos

Identificación de los temas de ayuda según el tipo de información

Los temas de ayuda se identifican con diferentes símbolos en función del tipo de información.

Símbolo	Tipo de información	Significado
	Instrucción	Describe el procedimiento paso a paso para realizar una determinada tarea.
	Ejemplo	Contiene un ejemplo de aplicación concreto que explica la tarea.
	Información general	Contiene información general necesaria para realizar una tarea.
	Referencia	Contiene amplia información de referencia para realizar consultas.

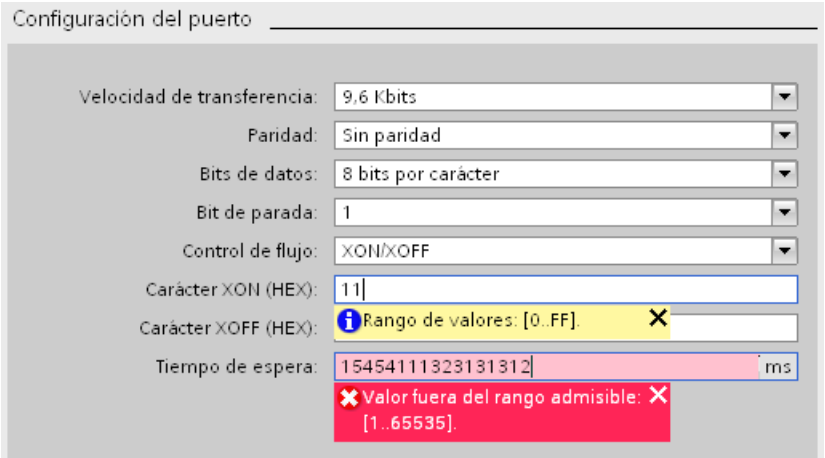
Identificación de los temas de ayuda según el sistema de destino

Dependiendo de los productos instalados, la ayuda puede contener capítulos que solo sean válidos para determinados dispositivos. Para reconocer este tipo de capítulos a primera vista, verá una nota entre paréntesis en la tabla de contenido. Los resultados en la búsqueda de texto completo y en el índice se identifican del mismo modo si solo son válidos para determinados dispositivos.

Roll-out

Algunos campos de entrada ofrecen una información desplegable ("roll-out"), que le ayuda a introducir parámetros y valores válidos. El roll-out proporciona información acerca de los rangos de valores y tipos de datos válidos para los campos de entrada.

La figura siguiente muestra un roll-out (amarillo) y un mensaje de error en forma de roll-out (rojo) que indica un valor no válido:



El roll-out se cierra en cuanto se sale del campo o se hace clic en la cruz de la esquina superior derecha.

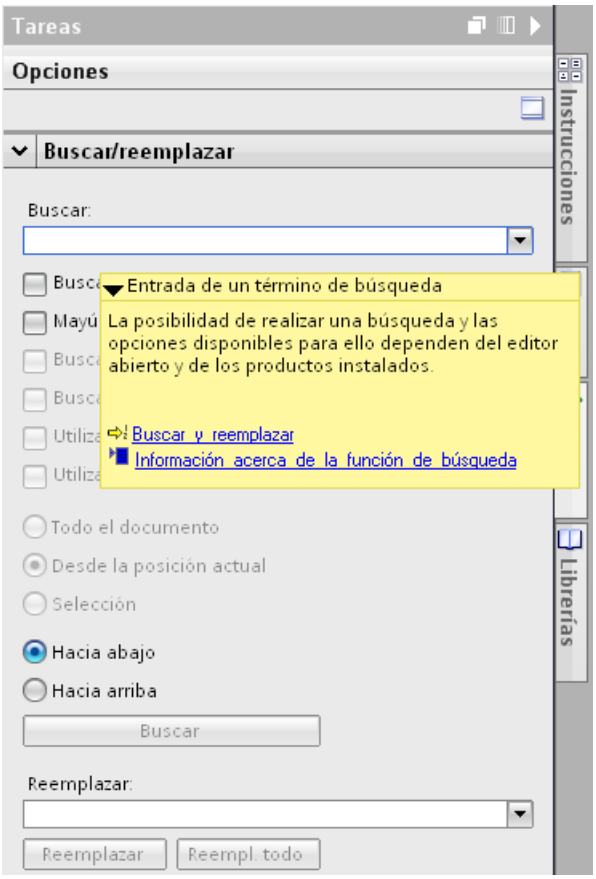
Tooltip

Cada elemento de la interfaz de usuario dispone de un tooltip que permite identificarlo fácilmente.

Los tooltips que tienen el símbolo de una flecha en el lado izquierdo contienen información adicional en cascada. Esta información se visualiza situando el puntero del ratón unos instantes sobre el tooltip, o bien haciendo clic en el símbolo de flecha. La visualización automática de tooltips en cascada se puede desactivar.

Si la ayuda contiene información más detallada, en la cascada aparecerá un enlace con el tema de ayuda correspondiente. Si hace clic en el enlace, se abrirá el tema de ayuda correspondiente.

La figura siguiente muestra un tooltip con la cascada abierta:



Consulte también

Configurar la visualización de tooltips y tooltips en cascada (Página 334)

6.2.2 Abrir la ayuda suministrada

La ayuda se abre desde el menú, mediante enlaces que aparecen en tooltips en cascada o con la tecla <F1>.

Procedimiento

Para abrir la ayuda suministrada, proceda del siguiente modo:

1. Elija el comando "Mostrar ayuda" del menú "Ayuda" o pulse <F1> para visualizar el tema de ayuda relacionado con el contexto actual.

O bien:

1. Haga clic en el enlace que aparece dentro de un tooltip en cascada para acceder a un tema más detallado de la ayuda.

6.2.3 Buscar palabras clave en la Ayuda

Buscar palabras clave en el texto de Ayuda

Para buscar palabras clave predefinidas en los temas de Ayuda, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar contenido" de la barra de herramientas de la Ayuda para mostrar la tabla de contenido.
   Se abre el contenido y se visualizan las fichas "Índice", "Búsqueda" y "Favoritos".
2. Abra la ficha "Índice".
3. Introduzca el término de búsqueda en el campo de entrada o selecciónelo de la lista de palabras clave.
4. Haga clic en "Mostrar".

6.2.4 Buscar en todo el texto

Buscar en todo el texto

Para buscar determinadas palabras en todo el texto de la Ayuda, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar contenido" de la barra de herramientas de la Ayuda para mostrar la tabla de contenido.
   Se abre el contenido y se visualizan las fichas "Índice", "Búsqueda" y "Favoritos".
2. Abra la ficha "Búsqueda".
3. Introduzca el término de búsqueda en el campo de entrada.
4. En caso necesario, la búsqueda se puede optimizar con criterios adicionales:
   - Seleccione "Buscar en anteriores" para iniciar una búsqueda adicional sólo en los últimos resultados de búsqueda.
   - Seleccione "Palabras similares" para buscar también aquellas palabras que se diferencien levemente del término de búsqueda.
   - Seleccione "Buscar sólo en los títulos" para obtener únicamente resultados que contengan el término de búsqueda en el título. El contenido de los temas de Ayuda no se tiene en cuenta en la búsqueda.

5. Haga clic en el botón de flecha situado a la derecha del campo de búsqueda para utilizar operadores booleanos. Operadores booleanos disponibles:
   - Combine dos o más términos de búsqueda con el operador "AND" para buscar sólo los temas de Ayuda cuyo texto contenga todas las palabras buscadas.
   - Combine dos o más términos de búsqueda con el operador "OR" para buscar sólo los temas de Ayuda cuyo texto contenga una o varias de las palabras buscadas.
   - Combine dos o más términos de búsqueda con el operador "NEAR" para buscar sólo los temas de Ayuda en los que las palabras buscadas estén próximas unas a otras (separadas por no más de ocho palabras).
   - Si introduce el operador "NOT" antes de una palabra, se excluirán de la búsqueda los temas de Ayuda que contengan esa palabra.
6. Haga clic en "Mostrar temas" para iniciar la búsqueda.
   Los resultados aparecen en una lista con indicación de título, posición y rango. En la columna "Posición" se indica el capítulo que contiene el tema de Ayuda encontrado. La clasificación por rango se efectúa conforme a la posición de los temas de Ayuda encontrados en el contenido y al número de resultados en los temas de Ayuda.

6.2.5 Utilizar Favoritos

Utilizar Favoritos

Los temas de Ayuda se pueden guardar como Favoritos. De esta manera se evita tener que buscar nuevamente un tema de Ayuda.

Guardar Favoritos:

Para guardar una página como Favorito, proceda del siguiente modo:

1. Abra el tema de Ayuda o el capítulo que desea guardar como Favorito.
2. Haga clic en el botón "Mostrar/ocultar contenido" de la barra de herramientas de la Ayuda para mostrar la tabla de contenido.
   Se abre el contenido y se visualizan las fichas "Índice", "Búsqueda" y "Favoritos".
3. Abra la ficha "Favoritos".
4. Haga clic en el botón "Agregar".
   El tema de Ayuda o el capítulo se guarda como Favorito y volverá a estar disponible cuando se vuelva a abrir la Ayuda.

Llamar Favoritos:

Para llamar una página de los Favoritos, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar contenido" de la barra de herramientas de la Ayuda para mostrar la tabla de contenido.
   Se abre el contenido y se visualizan las fichas "Índice", "Búsqueda" y "Favoritos".
2. Abra la ficha "Favoritos".

3. Seleccione el tema que desea abrir de la lista.
4. Haga clic en el botón "Mostrar".

Borrar Favoritos

Para borrar una entrada de los Favoritos, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar contenido" de la barra de herramientas de la Ayuda para mostrar la tabla de contenido.
   Se abre el contenido y se visualizan las fichas "Índice", "Búsqueda" y "Favoritos".
2. Abra la ficha "Favoritos".
3. Seleccione el tema que desea eliminar de la lista.
4. Haga clic en el botón "Quitar".

6.2.6 Imprimir temas de Ayuda

Imprimir información

Es posible imprimir todo el contenido de la Ayuda o algunos de sus temas.

Procedimiento

Para determinar qué temas desea imprimir, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar diálogo de impresión".
   El índice se abre en una ventana propia.
2. Active en el diálogo "Imprimir temas de ayuda" las casillas de verificación de las carpetas y temas de ayuda que desee imprimir.
3. Haga clic en el botón "Imprimir" para imprimir la información seleccionada.
   Se abre el cuadro de diálogo "Imprimir".
4. Seleccione la impresora que desea utilizar para imprimir los temas de Ayuda.
5. Haga clic en "Propiedades" si desea especificar más opciones de impresión.
6. Confirme las entradas realizadas con "Aceptar".
   Los temas de Ayuda se imprimen en la impresora seleccionada.

6.2.7 Configurar la visualización de tooltips y tooltips en cascada

Posibilidades de configuración para tooltips y tooltips en cascada

La visualización de tooltips y tooltips en cascada puede adaptarse a las necesidades de cada caso. Existen las siguientes posibilidades de ajuste:

- Mostrar u ocultar la visualización de textos cortados
  En ocasiones algún texto puede ser demasiado largo para un campo de texto. En ese caso, los textos solo se muestran completos en un tooltip al colocar el puntero del ratón sobre el campo de texto. Esta función puede activarse o desactivarse.
- Activar o desactivar tooltips
  Los tooltips proporcionan información detallada sobre un elemento de la interfaz. Además se activa la visualización de tootips en cascada. Por eso, si desactiva los tooltips tampoco se mostrarán las cascadas con ayuda contextual. No obstante, también es posible visualizar manualmente el tooltip del elemento de la interfaz activo pulsando <F1>.
- Activación y desactivación de la apertura automática de tooltips en cascada
  Tras un pequeño tiempo de permanencia con el puntero del ratón colocado sobre un tooltip, las cascadas existentes se mostrarán automáticamente. La visualización automática de los tooltips en cascada se puede activar y desactivar. Con la visualización automática desactivada, las cascadas deberán abrirse manualmente. Para ello haga clic en la flecha que aparece dentro del tooltip.

Procedimiento

Para configurar la visualización de tooltips y tooltips en cascada, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
2. Seleccione el grupo "General" en la navegación local.
3. Active o desactiva las diferentes casillas de verificación en el área "Tooltips" en función de sus necesidades. La casilla de verificación "Abrir cascada en los tooltips automáticamente" solo puede modificarse si se ha activado la visualización de los tooltips.

Consulte también

Generalidades acerca del sistema de información (Página 327)

6.2.8 Consignas de seguridad

Consignas de seguridad

En la Ayuda encontrará indicaciones para su seguridad personal así como para la prevención de daños materiales. Las indicaciones para su seguridad personal están resaltadas con un triángulo de advertencia; las informaciones para evitar únicamente daños materiales no llevan

dicho triángulo. De acuerdo al grado de peligro las consignas se representan, de mayor a menor peligro, como sigue.

⚠ PELIGRO
Significa que, si no se adoptan las medidas preventivas adecuadas se producirá la muerte, o bien lesiones corporales graves.

⚠ ADVERTENCIA
Significa que, si no se adoptan las medidas preventivas adecuadas puede producirse la muerte o bien lesiones corporales graves.

⚠ PRECAUCIÓN
Con triángulo de advertencia significa que, si no se adoptan las medidas preventivas adecuadas, pueden producirse lesiones corporales.

ATENCIÓN
Sin triángulo de advertencia significa que, si no se adoptan las medidas preventivas adecuadas, pueden producirse daños materiales.

Nota

Significa que puede producirse un resultado o estado no deseado si no se respeta la consigna de seguridad correspondiente.

Si se dan varios niveles de peligro se usa siempre la consigna de seguridad más severa en cada caso. Si en una consigna de seguridad con triángulo de advertencia se alarma de posibles daños personales, la misma consigna puede contener también una advertencia sobre posibles daños materiales.

Personal cualificado

El equipo/sistema correspondiente sólo deberá instalarse y operarse respetando las especificaciones de este documento. Sólo está autorizado a intervenir en este equipo el personal cualificado. En el sentido de esta documentación se trata de personas que disponen de los conocimientos técnicos necesarios para poner en funcionamiento, conectar a tierra y marcar los aparatos, sistemas y circuitos de acuerdo con las normas estándar de seguridad.

Uso conforme

Considere lo siguiente:

⚠ ADVERTENCIA
El equipo sólo se podrá utilizar para los casos de aplicación previstos en el catálogo y en la descripción técnica, y sólo asociado a los equipos y componentes de Siemens y de terceros que han sido recomendados y homologados por Siemens. El funcionamiento correcto y seguro del producto exige que su transporte, almacenamiento, instalación, montaje, manejo y mantenimiento hayan sido realizados de forma correcta.

Marcas registradas

Todos los nombres marcados con ® son marcas registradas de Siemens AG. Los restantes nombres y designaciones contenidos en el presente documento pueden ser marcas registradas cuya utilización por terceros para sus propios fines puede violar los derechos de sus titulares.

Exención de responsabilidad

Hemos comprobado la concordancia del contenido de esta publicación con el hardware y el software descritos. Sin embargo, como es imposible excluir excepciones, no podemos hacernos responsable de la plena concordancia. El contenido de esta publicación se revisa periódicamente; si es necesario, las posibles las correcciones se incluyen en la siguiente edición.

6.2.9 Elaboración de documentación individual

Documentación individualizada

En el Siemens Industry Online Support se puede elaborar documentación individualizada y personalizada. Para ello se dispone de todos los manuales e instrucciones de servicio configurables del Siemens Industry Online Support. De ellos pueden extraerse y combinarse las partes deseadas para elaborar una documentación personalizada en una librería. La documentación se organiza en la librería con ayuda de carpetas. Las carpetas se convierten posteriormente en los diferentes capítulos de la documentación personalizada.

Acceda aquí a su librería personal (https://www.automation.siemens.com/mdm/?guiLanguage=es).

Requisitos

- Los manuales o instrucciones de servicio utilizados deben ser configurables. Los manuales configurables se reconocen por el complemento "configurable" en el nombre.
- Para poder utilizar todas las funciones, debe estar registrado y haber iniciado sesión en el Siemens Industry Online Support.

Documentación en diferentes idiomas

En caso necesario, la documentación elaborada se puede cambiar a los idiomas inglés, francés, español, italiano y chino. De ese modo es posible, por ejemplo, reunir información relevante para un proyecto determinado y ponerla a disposición de compañeros que utilizan otra lengua.

Función de exportación dentro de la documentación

Dentro de la librería es posible exportar partes a diferentes formatos (PDF, XML, RTF) en cualquier punto.

Ayuda para la elaboración de la documentación

Encontrará más ayuda para la elaboración y el uso de documentación individualizada en el Siemens Industry Online Support (https://www.automation.siemens.com/mdm/_help/en/mdm_reference_manual_de-DE.htm).

Editar proyectos 7

7.1 Principios básicos de los proyectos

Introducción

Los proyectos sirven para almacenar de forma ordenada los datos y programas que se generan al crear una solución de automatización. Los datos contenidos en un proyecto comprenden básicamente:

- Datos de configuración del hardware instalado y datos de parametrización de los módulos
- Datos de configuración para la comunicación en redes
- Datos de configuración de los dispositivos
- Informes de eventos importantes en el ciclo vital del proyecto

Jerarquía del proyecto

Los datos se depositan en cada proyecto en forma de objetos. Los objetos se organizan en el proyecto en una estructura de árbol (jerarquía del proyecto).

La base de la jerarquía del proyecto la constituyen los dispositivos y estaciones, así como sus respectivos datos de configuración y programas.

Además, en el árbol del proyecto se representan p. ej. los datos comunes del proyecto y los accesos online.

Consulte también

Crear un proyecto nuevo (Página 340)

Abrir proyectos (Página 342)

Guardar proyectos (Página 346)

Borrar proyectos (Página 348)

Uso de informes (Página 339)

7.2 Uso de informes

En algunas operaciones realizadas dentro del TIA Portal se generan automáticamente y en segundo plano informes que documentan cambios en el proyecto. Los informes se crean automáticamente, por ejemplo durante la migración de proyectos y programas o la actualización de instancias de la librería.

Los informes se muestran en el árbol del proyecto, en la carpeta "Datos comunes". Se guardan con el proyecto en la carpeta del proyecto, por lo que pueden leerse independientemente de la PG/el PC utilizado en cuanto se haya abierto el proyecto.

Además de poder visualizarse en el TIA Portal, los informes también pueden imprimirse.

Abrir protocolos

Para abrir un informe, proceda del siguiente modo:

1. Abra la carpeta "Datos comunes > Informes" en el árbol del proyecto.
2. Haga doble clic en el informe deseado de la lista.
   El contenido del informe se muestra en el área de trabajo.

Borrado de un informe

Para borrar un informe, proceda del siguiente modo:

1. Seleccione el informe en el árbol del proyecto.
2. Pulse la tecla <Supr>.
   El informe seleccionado se borra del directorio del proyecto y se elimina del árbol del proyecto.

7.3 Crear y gestionar proyectos

7.3.1 Crear un proyecto nuevo

Procedimiento

Para crear un proyecto nuevo, proceda del siguiente modo:

1. Elija el comando "Nuevo" del menú "Proyecto".
   Se abre el diálogo "Crear proyecto nuevo".
2. Introduzca el nombre del proyecto y la ruta deseados o acepte los datos propuestos.
3. Haga clic en el botón "Crear".

Resultado

El proyecto se crea y se visualiza en el árbol del proyecto.

Consulte también

Principios básicos de los proyectos (Página 339)

Abrir proyectos (Página 342)

Guardar proyectos (Página 346)

Borrar proyectos (Página 348)

7.3.2 Compatibilidad de proyectos

El TIA Portal permite abrir proyectos que se crearon con una versión anterior del TIA Portal o con un volumen de instalación distinto. A continuación se explica lo que debe tenerse en cuenta en este caso.

Abrir proyectos de versiones anteriores del producto

La tabla siguiente muestra el comportamiento del TIA Portal cuando se abren proyectos de una versión anterior del producto:

Versión de producto del TIA Portal (extensión de archivo de la versión correspondiente)	Comportamiento al abrir el proyecto con TIA Portal V13
V10.5 (.ap10) V11.x (.ap11)	Al abrir el proyecto se actualiza automáticamente a la versión más reciente del producto tras confirmación por parte del usuario. El proyecto actualizado es una copia del original. El proyecto original se conserva sin cambios.
V12 (.ap12)	El proyecto se actualiza al abrirlo. El usuario puede elegir si desea actualizar el proyecto a la versión V12 SP1 o a la versión V13 del producto. En ambos casos, el proyecto actualizado es una copia del original. El proyecto original se conserva sin cambios. Si se actualiza el proyecto a la versión de producto V12 SP1, se abre en modo de compatibilidad.
V12 SP1 (.ap12)	Al abrir el proyecto el usuario puede elegir si desea abrirlo en modo de compatibilidad o bien actualizarlo a la última versión de producto. Si se elige actualizar el proyecto se crea una copia del original. El proyecto original se conserva sin cambios.

Modo de compatibilidad

Si se abre un proyecto en el formato de la versión V12 SP1 y no se actualiza, dicho proyecto estará en modo de compatibilidad. El volumen de funciones del TIA Portal queda limitado a las funciones de V12 SP1. El proyecto sigue siendo compatible con versiones anteriores y puede seguir abriéndose y editándose con el TIA Portal V12 SP1.

A los proyectos en modo de compatibilidad es posible agregarles componentes suministrados posteriormente para la versión V12 SP1 en el marco de un Hardware Support Package (HSP) El proyecto sigue siendo compatible con el TIA Portal V12 SP1. Para continuar trabajando con el proyecto en el TIA Portal V12 SP1 simplemente debe instalarse el Hardware Support Package.

Las librerías globales se crean siempre en el formato más actual y no son compatibles con versiones anteriores aunque el proyecto se abra en modo de compatibilidad.

Para utilizar el volumen de funciones completo de la versión actual es necesario actualizar el proyecto. La actualización convierte el proyecto al formato actual. A partir de ese momento podrá utilizarse el volumen de funciones completo de la versión actual.

Compatibilidad de la versión de producto actual con versiones anteriores

Los proyectos guardados con la versión actual del TIA Portal no son compatibles con versiones anteriores debido a la ampliación de funciones de la versión actual. Los proyectos guardados con el TIA Portal V13 solo pueden abrirse con el TIA Portal V13 o con versiones más recientes.

Abrir proyectos creados con productos adicionales

Existe la posibilidad de que el proyecto que debe abrirse contenga datos creados con un software opcional. Si estos componentes de software no están instalados pueden darse los casos siguientes:

- Faltan componentes de software que no son absolutamente indispensables:
  Se abre un cuadro de diálogo en el que aparecen listados los componentes de software que faltan. Después de abrir el proyecto se visualizan sus propiedades. En ese momento es posible instalar adicionalmente los componentes que faltan. Incluso si no instala adicionalmente los componentes que faltan, estarán disponibles todos los dispositivos que contiene el proyecto. No obstante, solo podrán editarse los dispositivos soportados por el software actualmente instalado.
  Los dispositivos que no se pueden editar porque falta el componente de software necesario, aparecen marcados en el árbol del proyecto con el siguiente símbolo:

- Faltan paquetes de software que son absolutamente indispensables:
  Se abre un cuadro de diálogo en el que aparecen listados los componentes de software que faltan. Los componentes de software que se necesitan forzosamente aparecen marcados. El proyecto solo se puede abrir si se instalan los componentes de software que faltan.

Consulte también

Abrir proyectos (Página 342)

Actualizar proyectos (Página 343)

Compatibilidad de las librerías globales (Página 441)

7.3.3 Abrir proyectos

En el TIA Portal es posible abrir todos los proyectos de la versión actual y de versiones anteriores.

Dado el caso, será necesario actualizar a un formato de proyecto más reciente algunos proyectos de versiones anteriores del TIA Portal. Al abrir el proyecto, se le requiere que lo actualice.

Los proyectos del TIA Portal se reconocen por la extensión ".ap[número de versión]" en el nombre de archivo. Los proyectos del TIA Portal V13 tienen la extensión ".ap13" en el nombre de archivo.

Procedimiento

Para abrir un proyecto existente, proceda del siguiente modo:

1. Elija el comando "Abrir" del menú "Proyecto".
   Aparece el cuadro de diálogo "Abrir proyecto", en el que se muestra una lista de los proyectos utilizados recientemente.
2. Seleccione un proyecto de la lista y haga clic en "Abrir".
3. Si el proyecto deseado no está disponible en la lista, haga clic en el botón "Examinar". Navegue hasta la carpeta de proyecto deseada y abra el archivo de proyecto.
   Los proyectos en el formato actual se abren en la vista del proyecto. Al seleccionar un proyecto de una versión anterior del TIA Portal se abre el cuadro de diálogo "Actualizar proyecto". Encontrará más información sobre la actualización del proyecto en el capítulo "Actualizar proyectos (Página 343)".

Consulte también

Compatibilidad de proyectos (Página 341)

Actualizar proyectos (Página 343)

Compilar datos de proyecto (Página 364)

Abrir una librería global (Página 443)

Principios básicos de los proyectos (Página 339)

Crear un proyecto nuevo (Página 340)

Guardar proyectos (Página 346)

Borrar proyectos (Página 348)

7.3.4 Actualizar proyectos

Los proyectos de versiones anteriores del TIA Portal pueden editarse con la versión actual del mismo. En función de la versión de producto con la que se haya creado el proyecto se puede elegir entre actualizar el proyecto o abrirlo en modo de compatibilidad. Para saber las posibilidades disponibles en función de la versión de producto del proyecto lea el capítulo "Compatibilidad de proyectos (Página 341)".

Al abrir proyectos de versiones anteriores del producto se le pedirá que actualice el proyecto. Los proyectos que ya se hayan abierto en modo de compatibilidad pueden actualizarse manualmente. Una vez actualizado el proyecto a la versión de producto actual, ya no podrá abrirse en versiones anteriores del TIA Portal. Sin embargo, el proyecto original se conserva. El proyecto actualizado se guarda como copia.

Actualizar proyectos de V11.x o versiones anteriores

Para actualizar un proyecto del TIA Portal V11.x o anterior, proceda del siguiente modo:

1. Abra el proyecto.
   Se abre el cuadro de diálogo "Actualizar proyecto".
2. Haga clic en "Aceptar".
   Se crea y actualiza una copia del proyecto. El nombre de dicha copia se complementa con "_V13". Se abre el proyecto.
3. Compile el hardware y software de todos los dispositivos del proyecto.

Actualizar proyectos de V12

Para actualizar un proyecto del TIA Portal V12, proceda del siguiente modo:

1. Abra el proyecto.
   Se abre el cuadro de diálogo "Actualizar proyecto".
2. Seleccione la versión de destino para la actualización:
   - Para actualizar el proyecto a la versión de producto V13 haga clic en "Sí".
   - Para actualizar el proyecto a la versión de producto V12 SP1 y trabajar en modo de compatibilidad haga clic en "No".

   Se crea y actualiza una copia del proyecto. El nombre de dicha copia se complementa con "_V13" o "V12SP1". Se abre el proyecto.
3. Compile el hardware y software de todos los dispositivos del proyecto.

Actualizar proyectos de V12 SP1 o utilizarlos en modo de compatibilidad

Para actualizar un proyecto del TIA Portal V12 SP1 o utilizarlo en modo de compatibilidad, proceda del siguiente modo:

1. Abra el proyecto.
   Se abre el cuadro de diálogo "Actualizar proyecto".
2. Elija lo que desea hacer con el proyecto:
   - Para actualizar el proyecto a la versión de producto V13 haga clic en "Sí".
   - Para utilizar el proyecto en el modo de compatibilidad haga clic en "No".
3. Opcionalmente: si ha actualizado el proyecto a la última versión del producto, compile el hardware y software de todos los dispositivos del proyecto.

Actualizar manualmente proyectos en modo de compatibilidad a V13

Para esta operación debe haber un proyecto abierto en modo de compatibilidad para el TIA Portal V12 SP1.

Para actualizar manualmente un proyecto en modo de compatibilidad, proceda del siguiente modo:

1. Elija el comando "Actualizar" del menú "Proyecto".
   Aparece una consulta de seguridad.
2. Confirme la consulta de seguridad con "Sí".
   Se crea y actualiza una copia del proyecto. El nombre de dicha copia se complementa con "_V13". Se abre el proyecto.
3. Compile el hardware y software de todos los dispositivos del proyecto.

Indicaciones

Tras actualizar un proyecto tenga en cuenta las indicaciones siguientes:

Nota

Actualización de librerías globales

Puesto que las librerías globales son independientes de los proyectos, no se actualizan automáticamente junto con el proyecto. Si desea seguir utilizando librerías globales de versiones anteriores del TIA Portal, deberá actualizarlas también. Encontrará más información sobre la actualización de librerías globales en el capítulo "Compatibilidad de las librerías globales (Página 441)".

Nota

Actualización de bloques con protección de know-how

El bloque solo se actualiza y puede cargarse después de abrirlo con la contraseña. Para actualizar también los bloques, después de actualizar el proyecto abra los bloques con protección de know-how. Si ha protegido con la misma contraseña muchos bloques con protección de know-how, es posible seleccionarlos y abrirlos todos de una vez.

Consulte también

Compatibilidad de proyectos (Página 341)

Abrir proyectos (Página 342)

Actualizar librerías globales (Página 444)

Compilar datos de proyecto (Página 364)

Compatibilidad de las librerías globales (Página 441)

7.3.5 Visualización de las propiedades del proyecto

Las propiedades de un proyecto se pueden visualizar. Las propiedades visualizadas son las siguientes:

- Metadatos del proyecto
  Aquí obtendrá por ejemplo información sobre la fecha de creación, el autor, la ruta, el tamaño del proyecto, indicaciones sobre los derechos de autor, los idiomas del proyecto utilizados, etc. Muchas propiedades se pueden modificar.
- Historial del proyecto
  El historial del proyecto contiene un resumen con los eventos más importantes del ciclo de vida de un proyecto. Aquí se puede ver, por ejemplo, con qué versión del TIA Portal se ha creado un proyecto y si entretanto se ha convertido a otra versión. Si un proyecto se ha creado, por ejemplo, mediante migración, también se muestra en la tabla del historial de proyecto con la fecha y la hora del momento de la migración. Si se ha creado un informe para un evento, además puede consultar el informe directamente.
- Support Packages en el proyecto
  Se visualiza una relación del software adicional que se requiere para poder editar todos los dispositivos del proyecto. Además se listan los archivos GSD instalados (archivos de descripción de dispositivos para otros dispositivos del catálogo de hardware).
- Productos de software en el proyecto
  Es posible visualizar un resumen de todos los productos de software instalados que son indispensables en el proyecto.

Procedimiento

Para ver las propiedades del proyecto, proceda del siguiente modo:

1. Seleccione el proyecto abierto en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual del proyecto.
   Se abrirá el cuadro de diálogo con las propiedades del proyecto.
3. Seleccione en la navegación local aquellas propiedades del proyecto que desee visualizar.

7.3.6 Guardar proyectos

Un proyecto se puede guardar en cualquier momento bien sea con el mismo nombre, o bien con un nombre diferente. Un proyecto también se puede guardar aunque aún contenga elementos erróneos.

Guardar un proyecto

Para guardar un proyecto, proceda del siguiente modo:

1. Elija el comando "Guardar" del menú "Proyecto".
   Todos los cambios efectuados en el proyecto se guardan con el nombre del proyecto actual. Si se edita un proyecto de una versión anterior del TIA Portal, la extensión de archivo del proyecto también se conserva y, además, sigue siendo posible editarlo en la versión anterior del TIA Portal.

Guardar un proyecto como

Para guardar el proyecto con un nombre diferente, proceda del siguiente modo:

1. Elija el comando "Guardar como" del menú "Proyecto".
   Se abre el diálogo "Guardar proyecto actual como".
2. En el campo "Guardar en", seleccione la carpeta de proyecto deseada.
3. Introduzca el nuevo nombre del proyecto en el campo "Nombre de archivo".
4. Confirme su entrada con "Guardar".
   El proyecto se guarda con el nuevo nombre y se abre.

Consulte también

Principios básicos de los proyectos (Página 339)

Crear un proyecto nuevo (Página 340)

Abrir proyectos (Página 342)

Borrar proyectos (Página 348)

Actualizar proyectos (Página 343)

Compatibilidad de proyectos (Página 341)

7.3.7 Cerrar proyectos

Procedimiento

Para cerrar un proyecto, proceda del siguiente modo:

1. Elija el comando "Cerrar" del menú "Proyecto".
   Si ha modificado el proyecto desde la última vez que fue guardado, aparecerá un mensaje.
2. Indique si desea guardar los cambios.

7.3.8 Borrar proyectos

Nota

Al borrar un proyecto se eliminan todos los datos de proyecto del medio de almacenamiento.

Requisitos

El proyecto que se desea borrar no está abierto.

Procedimiento

Para borrar un proyecto existente, proceda del siguiente modo:

1. Elija el comando "Borrar proyecto" del menú "Proyecto".
   Se abre el diálogo "Borrar proyecto" con la lista de los proyectos utilizados recientemente.
2. Seleccione un proyecto de la lista.
   Si el proyecto deseado no está disponible en la lista, haga clic en el botón "Examinar".
   Navegue hasta la carpeta de proyecto deseada y abra el archivo de proyecto.
3. Haga clic en el botón "Borrar".
4. Confirme la consulta de seguridad con "Sí" para borrar el proyecto definitivamente.

Resultado

Toda la carpeta de proyecto se borra del sistema de archivos.

Consulte también

Principios básicos de los proyectos (Página 339)

Crear un proyecto nuevo (Página 340)

Abrir proyectos (Página 342)

Guardar proyectos (Página 346)

7.3.9 Trabajar con proyectos multilingües

7.3.9.1 Principios básicos de los textos del proyecto

Textos en distintos idiomas en el proyecto

Al editar un proyecto, los textos se introducen por lo general en el idioma materno. Si el proyecto se entrega más adelante a otro operador que no domine este idioma, este necesitará una traducción de todos los textos relevantes en un idioma que comprenda. Por ello todos los

textos pueden ser traducidos. De esta manera se garantiza que toda persona que deba trabajar posteriormente con los textos del proyecto pueda visualizarlos en el idioma deseado.

Idioma del proyecto

Los idiomas del proyecto son todos los idiomas en los que un proyecto debe editarse posteriormente. Todos los textos pueden traducirse del idioma de edición a los distintos idiomas del proyecto. En el árbol del proyecto en "Idiomas y recursos > Idiomas del proyecto" se define qué idiomas han de estar disponibles en el proyecto.

Idioma de edición

Todo proyecto tiene un idioma de edición. Cuando se introducen textos, estos se crean siempre en el idioma de edición. Por esta razón tenga en cuenta que el idioma de edición configurado corresponda al idioma en el que se introducen los textos. De esta manera se evitan problemas al traducir los textos más adelante.

El idioma de edición no depende del idioma de la interfaz de usuario. Por ejemplo, se puede configurar inglés como idioma de la interfaz pero utilizar Italiano como idioma de edición. En este caso, los textos que se introduzcan se crearán en el idioma del proyecto "Italiano" aunque la interfaz de usuario del TIA Portal se visualice en inglés.

El idioma de edición se configura en el árbol del proyecto en "Idiomas y recursos" > Idiomas del proyecto > Idioma de edición".

Idioma de referencia

El idioma de referencia se utiliza como plantilla para la traducción. En la Task Card "Tareas > Idiomas y recursos" se muestra para cada campo de entrada el texto en el idioma de referencia. De ese modo es posible saber qué texto corresponde a qué campo de texto, aunque en el idioma de edición activo todavía no se haya introducido ningún texto.

Textos de usuario y textos del sistema

Para mejorar la comprensión se distingue entre textos de usuario y textos del sistema:

- Los textos de usuario son textos que han sido creados por el usuario.
- Los textos del sistema son textos que se crean automáticamente en el proyecto en función de la configuración.

Los textos del proyecto se gestionan en el árbol del proyecto en "Idiomas y recursos > Textos del proyecto".

Ejemplo de textos del proyecto en varios idiomas

Por ejemplo, los siguientes textos del proyecto pueden gestionarse en varios idiomas:

- Títulos y comentarios de bloque
- Títulos y comentarios de segmento
- Comentarios recogidos en tablas

- Textos de aviso
- Textos relevantes para el operador
- Listas de textos
- Rotulaciones de botones
- Nombres de visualización de recetas

Traducir textos

Para traducir textos existen los procedimientos siguientes.

- Traducción tabular de todos los textos utilizados en un proyecto
  Las traducciones en los distintos idiomas del proyecto se pueden introducir directamente en la tabla "Textos del proyecto". Estos se encuentran en el árbol del proyecto en "Idiomas y recursos > Textos del proyecto".
- Entrada de textos asignados a objetos determinados en la ventana de inspección
  En la ventana de inspección se pueden traducir textos asignados a los objetos seleccionados actualmente. En una tabla se muestran columnas para todos los idiomas del proyecto disponibles. En ellas se puede introducir la traducción del texto en cuestión.
- Traducir textos con ayuda de textos de referencia
  Si el número de textos no es muy elevado, se puede cambiar el idioma de edición. Todos los campos de texto se rellenan luego con los valores estándar, pudiéndose sobrescribir en el idioma actual. Para saber qué fue lo último que se introdujo en un campo, el texto se puede visualizar en el idioma de referencia. Para ello, utilice la Task Card "Tareas" y abra la paleta "Idiomas y recursos".
- Exportar textos y traducirlos externamente
  Grandes volúmenes de texto se pueden exportar a un archivo Office Open XML y traducir en cualquier programa de hoja de cálculo. A continuación, la lista traducida se debe importar nuevamente al TIA Portal.

Nota

Utilización de idiomas de proyecto asiáticos

Los idiomas de proyecto de Asia oriental solo se visualizan de forma correcta en Windows XP si la opción "Instalar archivos para los idiomas de Asia oriental" está activada en el panel de control bajo "Opciones regionales y de idioma > Idiomas".

Consulte también

Resumen de los ajustes del programa (Página 279)

Modificar la configuración (Página 283)

Ejemplo de aplicación para proyectos en varios idiomas (Página 356)

7.3.9.2 Seleccionar los idiomas del proyecto

Todos los textos de un proyecto pueden representarse en el mismo idioma que se ha seleccionado para la interfaz de usuario. Para ello es preciso que todos los textos del proyecto existan en el idioma correspondiente. El usuario puede definir los idiomas del proyecto disponibles.

Requisitos

- La vista del proyecto está abierta.
- Hay un proyecto abierto.

Procedimiento

Para seleccionar los idiomas del proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Idiomas del proyecto".
   En el área de trabajo aparece una lista de los idiomas que se pueden seleccionar.
3. Active los idiomas deseados.

Resultado

Todos los textos pueden visualizarse en los idiomas activados siempre y cuando ya exista una traducción en dichos idiomas.

7.3.9.3 Definir el idioma de edición

Todos los textos del proyecto se crean en el idioma de edición en el momento de introducirlos. Si cambia el idioma de edición, todos los textos que se introduzcan en adelante se guardarán en el nuevo idioma de edición.

Requisitos

- La vista del proyecto está abierta.
- Hay un proyecto abierto.

Procedimiento

Para cambiar el idioma de edición, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Idiomas del proyecto".
   En el área de trabajo se visualizan las opciones de ajuste para los idiomas del proyecto.
3. Seleccione el idioma de edición en "General > Idioma de edición".

7.3.9.4 Traducción tabular de todos los textos del proyecto

Todos los textos utilizados en el proyecto activo pueden visualizarse y editarse en una lista. Para una visualización más clara, los textos de usuario y los textos del sistema están separados en dos listas diferentes. Cada una de las listas contiene una columna para cada idioma del proyecto, en la que se puede introducir la traducción de los textos.

Requisitos

- La vista del proyecto está abierta.
- Se ha seleccionado por lo menos un idioma del proyecto adicional.

Procedimiento

Para traducir textos en la lista válida para todo el proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto".
   En el área de trabajo aparece una lista con los textos de usuario del proyecto.
3. Haga clic en "Textos del sistema" en caso de que se desee editar la lista con los textos del sistema en lugar de los textos de usuario.
4. Procure crear la lista de la forma más clara posible en caso de que el volumen de los textos sea demasiado grande.
   - Para agrupar textos idénticos y traducirlos de una sola vez, haga clic en el botón "Mostrar y ocultar grupo" de la barra de herramientas.
   - Para ocultar textos sin traducir, haga clic en el botón "Mostrar y ocultar textos vacíos" de la barra de herramientas.
   - Para limitar la visualización de los textos del proyecto a determinados dispositivos, seleccione en la lista desplegable los dispositivos cuyos textos del proyecto desee visualizar.
5. Introduzca la traducción de los textos del proyecto en la columna correspondiente.

7.3.9.5 Traducción de textos correspondientes a objetos individuales

Para editar los textos de objetos individuales resultaría demasiado complicado buscar en la tabla los textos correspondientes con todos los textos del proyecto. Por ello, en la ventana de inspección hay una tabla en la que solo se muestran los textos asignados a los objetos seleccionados en ese momento. En la tabla se pueden completar las traducciones que falten en determinados idiomas, o pueden modificarse textos ya existentes.

Requisitos

Para los textos que se traducen debe existir ya un texto como mínimo en un idioma del proyecto.

Procedimiento

Para editar los textos de los objetos seleccionados, proceda del siguiente modo:

1. Seleccione los objetos que desea editar.
2. Abra la ficha "Propiedades" de la ventana de inspección.
3. Abra la ficha "Textos" de la ventana de inspección.
   Se muestra una tabla con todos los textos correspondientes a todos los objetos seleccionados. Dicha tabla contiene una columna para cada uno de los idiomas de edición ajustados actualmente y el idioma de referencia, y columnas para los idiomas adicionales del proyecto.
4. Complete o modifique en la tabla las entradas de los diferentes idiomas del proyecto.

Consulte también

Ejemplo de aplicación para proyectos en varios idiomas (Página 356)

7.3.9.6 Traducir textos con ayuda de textos de referencia

Introducción

Tras cambiar el idioma de edición, todos los textos contenidos en los campos de entrada se visualizan en el nuevo idioma de edición. Si los textos no están traducidos todavía al idioma configurado, los campos de entrada aparecen vacíos o se rellenan con los valores estándar.

Si se introduce texto en un campo de entrada, el texto se guardará en el idioma de edición actual. Posteriormente, los textos de ese campo de entrada existirán en dos idiomas del proyecto, a saber: en el idioma de edición anterior y en el idioma de edición actual. De esta manera se pueden crear textos en varios idiomas del proyecto.

En un campo de entrada es posible visualizar las traducciones existentes en otros idiomas del proyecto. Éstas sirven de comparación para la entrada de textos en el idioma de edición actual y se denominan idioma de referencia.

Nota

La visualización de textos de referencia depende de los productos instalados y no todos los editores la soportan.

Requisitos

Para un campo de entrada existe por lo menos una traducción a un idioma del proyecto adicional.

Procedimiento

Para visualizar la traducción del contenido de un campo de entrada en un idioma de referencia, proceda del siguiente modo:

1. Seleccione la paleta "Idiomas y recursos" de la Task Card "Tareas".
2. Seleccione un idioma de referencia de la lista desplegable "Idioma de referencia".

Resultado

El idioma de referencia está predeterminado. Si hace clic en un campo de texto, en la Task Card "Tareas > Idiomas y recursos" se visualizarán las traducciones existentes en otros idiomas del proyecto.

7.3.9.7 Importar y exportar textos del proyecto

Es posible exportar textos del proyecto para traducirlos y luego importarlos nuevamente. Los datos se exportan a un archivo Office Open XML con la extensión ".xlsx". Este tipo de archivo se puede editar en Microsoft Excel o en otros programas de hojas de cálculo.

Los datos se pueden exportar de las maneras siguientes:

- Exportar textos del proyecto individuales
- Exportar simultáneamente todos los textos de usuario o textos del sistema
  En este caso la exportación se puede limitar adicionalmente por categorías.

Nota

Limitación de filas en Microsoft Excel

Recuerde que los programas de hojas de cálculo pueden tener una limitación del número de filas procesable. Microsoft Excel 2003 soporta, por ejemplo, 65536 filas como máximo. Las nuevas versiones de Microsoft Excel soportan muchas más filas.

Exportar textos del proyecto individuales

Para exportar textos del proyecto individuales, proceda del siguiente modo:

1. Abra la carpeta "Idiomas y recursos" del árbol del proyecto.
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto".
   Se abre el editor de textos del proyecto.
3. Dependiendo de qué textos desee exportar, elija en el editor la ficha "Textos de usuario" o "Textos del sistema".
4. Seleccione los textos del proyecto que desee exportar.
5. Haga clic en "Exportar textos del proyecto" en la barra de herramientas del editor.
   Se abrirá el cuadro de diálogo "Exportar".
6. Elija en la lista desplegable "Idioma de origen" el idioma a partir del cual desea efectuar la traducción.
7. Elija en la lista desplegable "Idioma de destino" el idioma al que desea traducir el texto. En la lista desplegable están disponibles los idiomas del proyecto definidos previamente. Si no está disponible el idioma deseado, defínalo primero en el editor de idiomas del proyecto.
8. Indique una ruta y un nombre para el archivo de exportación en el campo de entrada "Seleccionar archivo que exportar".
9. Haga clic en "Exportar".

Exportar todos los textos de usuario o del sistema

Para exportar todos los textos del proyecto, proceda del siguiente modo:

1. Elija el comando "Exportar textos del proyecto" del menú "Herramientas".
   Se abrirá el cuadro de diálogo "Exportar".
2. Elija en la lista desplegable "Idioma de origen" el idioma a partir del cual desea efectuar la traducción.
3. Elija en la lista desplegable "Idioma de destino" el idioma al que desea traducir el texto. En la lista desplegable están disponibles los idiomas del proyecto definidos previamente. Si no está disponible el idioma deseado, defínalo primero en el editor de idiomas del proyecto.
4. Active la casilla de verificación "Textos de usuario" en "Seleccionar contenidos" para exportar los textos de usuario. Seleccione "Textos del sistema" para exportar los textos del sistema. Active las dos casillas de verificación para exportar los textos de usuario y los textos del sistema.
5. Seleccione las categorías de texto deseadas para los textos de usuario o los textos del sistema en "Seleccionar contenidos".
6. Indique un nombre para el archivo de exportación en el campo de entrada "Archivo de exportación".
7. En el campo de entrada "Ruta" seleccione la ruta del sistema de archivos donde se debe guardar el archivo de exportación.
8. Haga clic en "Exportar".

Importar textos del proyecto

Para importar un archivo con textos del proyecto, proceda del siguiente modo:

1. Elija el comando "Importar textos del proyecto" del menú "Herramientas".
   Se abrirá el cuadro de diálogo "Importar".
2. Seleccione la ruta y el nombre del archivo de importación en el campo "Seleccionar archivo que importar".
3. Si ha realizado modificaciones en el idioma base del archivo de exportación y desea que las entradas correspondientes del proyecto se sobrescriban con estas modificaciones, active la casilla de verificación "Importar idioma base".
4. Haga clic en "Importar".

Consulte también

Ejemplo de aplicación para proyectos en varios idiomas (Página 356)

7.3.9.8 Ejemplo de aplicación para proyectos en varios idiomas

Introducción

Supongamos que se está trabajando en un equipo con colegas que hablan inglés, francés o alemán. Se ha creado un proyecto mediante el TIA Portal y se ha elaborado una configuración que ya está funcionando.

Para hacer que el proyecto sea comprensible para los demás compañeros del equipo se dotan todos los dispositivos con comentarios en los idiomas inglés y alemán. En primera instancia se introducen los comentarios en alemán. A continuación se encomienda a una agencia de traducción, por razones de tiempo y costes, traducir los textos al inglés en un programa de hoja de cálculo.

Además, es necesario redactar un comentario en francés acerca de un dispositivo determinado para que el colega francófono pueda seguir editando este dispositivo.

En el siguiente ejemplo se describe cómo se puede llegar a este punto a través del TIA Portal.

Traducir el proyecto al inglés

Para introducir los comentarios en alemán y luego mandarlos a traducir al inglés, proceda del siguiente modo:

1. Configure "Alemán" como el idioma de edición y rellene todos los campos de comentario con los respectivos textos en alemán.
   En el dispositivo seleccionado para el colega francófono introduzca en alemán "Unser neues Gerät", por ejemplo.
   De esta manera, todos los comentarios se han guardado en alemán.
2. Exporte todos los textos de usuario a un archivo Office Open XML con la extensión ".xlsx".

3. Mande traducir al inglés los textos de usuario que se encuentran en este archivo mediante cualquier programa de hoja de cálculo como, por ejemplo, Microsoft Excel.
4. Importe el archivo traducido al TIA Portal.
   Ahora todos los textos están disponibles en alemán e inglés.

Traducir un campo de comentario individual al francés

Para traducir el contenido de un campo de comentario individual al francés, proceda del siguiente modo:

1. Abra el campo de comentario del dispositivo que debe seguir siendo editado por el colega francófono.
2. Abra la paleta "Idiomas y recursos" de la Task Card "Tareas".
3. Configure "Francés" como idioma de edición en la paleta "Idiomas y recursos". Configure "Inglés" como idioma de referencia, por ejemplo.
   El campo de comentario está vacío ya que aún no existe una traducción en francés. Sin embargo, en la paleta "Idiomas y recursos" se visualiza la traducción al inglés "Our new device" como referencia.
4. Oriéntese según el texto de referencia en inglés e introduzca "Notre nouvel appareil" en el campo de comentario.
   Ahora el comentario sobre este dispositivo está en alemán, inglés y francés.

Consulte también

Principios básicos de los textos del proyecto (Página 348)

Importar y exportar textos del proyecto (Página 354)

Traducción de textos correspondientes a objetos individuales (Página 353)

7.3.10 Archivación y desarchivación de proyectos

7.3.10.1 Uso de ficheros de proyecto

Archivación y transmisión de proyectos

Si ha trabajado de forma prolongada con un proyecto, con instalaciones de hardware amplias pueden generarse grandes archivos. Por ello, es posible reducir su tamaño p. ej. para archivar un proyecto en un disco duro externo o enviarlo por correo electrónico, para lo que es importante que el tamaño del archivo sea reducido.

Posibilidades de reducción del tamaño del proyecto

Existen dos formas de reducir el tamaño del proyecto:

- Creación de un fichero de proyecto
  Los ficheros de proyecto del TIA-Portal son archivos comprimidos que contienen un proyecto completo, incluida toda la estructura de carpetas del proyecto. Antes de convertir el directorio del proyecto en un archivo comprimido se reducen todos los archivos a sus componentes más esenciales para reducir aún más el tamaño del proyecto. Por ello, los ficheros de proyecto son especialmente adecuados para el envío por correo electrónico. Los ficheros de proyecto tienen la extensión ".zap[número de versión del TIA Portal]". Los proyectos generados con el TIA Portal V13 tienen la extensión ".zap13".
  Para abrir un fichero de proyecto es necesario desarchivarlo. Al hacerlo, el archivo comprimido se descomprime a la estructura de directorio original del proyecto en una ubicación seleccionada por el usuario, con todos los archivos contenidos en el proyecto.
- Minimizar un proyecto
  Se puede renunciar a la compresión adicional en un archivo comprimido y crear en su lugar una copia del directorio del proyecto. Los archivos contenidos se reducen a los componentes esenciales del proyecto. Con ello se minimiza el espacio de memoria necesario. Se conservan todas las funciones del proyecto, y el proyecto se puede abrir de la forma habitual.
  Un proyecto minimizado es especialmente adecuado para la archivación, p. ej. en un soporte de datos externo.

Consulte también

Desarchivación de proyectos (Página 359)

Archivación de proyectos (Página 358)

7.3.10.2 Archivación de proyectos

El espacio necesario en memoria del proyecto abierto actualmente se puede reducir comprimiendo el proyecto en un archivo comprimido o reduciendo los archivos del proyecto a sus componentes esenciales. Ambas opciones se consiguen con la función de archivación del TIA Portal.

Nota

Para la archivación se utiliza la última versión guardada del proyecto. Por ello es necesario guardar el proyecto antes de utilizar la función de archivación. De ese modo, los últimos cambios también estarán contenidos en el proyecto archivado.

Procedimiento

Para archivar un proyecto, proceda del siguiente modo:

1. Elija el comando "Archivar" del menú "Proyecto".
   Se abre el cuadro de diálogo "Archivar el proyecto actual como...".
2. Seleccione el directorio en el que desea guardar el archivo comprimido o el nuevo directorio del proyecto.

3. Seleccione el tipo de archivo de la lista desplegable "Tipo de archivo":
   - Archivo de proyecto del TIA Portal si desea crear un archivo comprimido del proyecto.
   - Proyecto TIA Portal minimizado si solo desea crear una copia del directorio del proyecto con espacio en memoria minimizado.
4. Introduzca un nombre de archivo en el campo "Nombre de archivo" para crear un archivo comprimido. Si crea un directorio de proyecto minimizado, introduzca en su lugar el nombre del directorio nuevo en el campo "Nombre de archivo".
5. Haga clic en "Guardar".

Resultado

Si ha creado un fichero de proyecto, se genera un archivo comprimido con la extensión ".zap13". Los ficheros de proyectos en modo de compatibilidad para la versión de producto V12 SP1 tienen la extensión ".zap12" y son compatibles con el TIA Portal V12 SP1. El archivo comprimido contiene todo el directorio del proyecto. Los diferentes archivos del proyecto se han reducido además a sus componentes esenciales para ahorrar espacio en la memoria.

Si ha creado un proyecto TIA Portal minimizado, en la ubicación deseada solo se genera una copia del directorio original del proyecto. Los archivos contenidos en ella se han reducido a sus componentes esenciales para ahorrar espacio en la memoria.

Consulte también

Uso de ficheros de proyecto (Página 357)

Desarchivación de proyectos (Página 359)

7.3.10.3 Desarchivación de proyectos

Los ficheros de proyecto del TIA Portal se descomprimen con la función "Desarchivar". Al hacerlo se restaura la estructura del directorio del proyecto con todos los archivos del proyecto.

Requisitos

No puede haber ningún proyecto abierto.

Procedimiento

Para descomprimir un fichero de proyecto, proceda del siguiente modo:

1. Elija el comando "Desarchivar" del menú "Proyecto".
   Se abre el cuadro de diálogo "Desarchivar proyecto archivado".
2. Seleccione el fichero del proyecto.
3. Haga clic en "Abrir".
4. Se abre el cuadro de diálogo "Buscar carpeta".
5. Seleccione el directorio de destino en el que desea descomprimir el proyecto archivado.
6. Haga clic en "Aceptar".

Resultado

El proyecto se descomprime en el directorio seleccionado y se abre inmediatamente. Cuando se descomprime un fichero de proyecto que contiene un proyecto de una versión de producto anterior a V12 SP1 es posible que deba actualizarse el proyecto. Se recibirá automáticamente una petición al respecto en cuanto se abra el proyecto. Rigen las mismas reglas descritas en el capítulo "Compatibilidad de proyectos (Página 341)".

Consulte también

Uso de ficheros de proyecto (Página 357)

Abrir proyectos (Página 342)

Compatibilidad de proyectos (Página 341)

Actualizar proyectos (Página 343)

7.4 Uso de proyectos de referencia

7.4.1 Principios básicos de los proyectos de referencia

Introducción

Adicionalmente al proyecto actual el usuario puede abrir otros proyectos de referencia. Los proyectos de referencia pueden utilizarse del siguiente modo:

- Se pueden arrastrar objetos individuales de un proyecto de referencia al proyecto actual y editarlos a continuación.
- Se pueden abrir con protección de escritura determinados objetos de un proyecto de referencia, como p. ej. bloques lógicos. No obstante, esta función no está disponible para todos los elementos.
- En una comparación offline/offline se pueden comparar dispositivos del proyecto de referencia con dispositivos del proyecto actual.

Recuerde que los proyectos de referencia están protegidos contra escritura. Así, los objetos de un proyecto de referencia no se pueden modificar.

También es posible abrir como proyectos de referencia proyectos que se crearon con una versión anterior del TIA Portal o con un volumen de instalación distinto. En este caso se aplican las mismas reglas de compatibilidad que al abrir normalmente un proyecto de una versión anterior del TIA Portal.

Consulte también: Compatibilidad de proyectos (Página 341)

Consulte también

Comparación de proyectos de referencia (Página 362)

Apertura y cierre de un proyecto de referencia (Página 361)

Proyectos de referencia (Página 303)

7.4.2 Apertura y cierre de un proyecto de referencia

Apertura de un proyecto de referencia

Para abrir un proyecto de referencia, proceda del siguiente modo:

1. En la paleta "Proyectos de referencia" del árbol del proyecto haga clic en "Abrir proyecto de referencia" en la barra de herramientas.
   Se abre el cuadro de diálogo "Abrir proyecto de referencia".
2. Navegue hasta la carpeta de proyecto deseada y abra el archivo de proyecto. Los proyectos del TIA Portal V13.x tienen la extensión ".ap13". Los proyectos antiguos del TIA Portal tienen la extensión "ap[número de versión]".
3. Haga clic en "Abrir".
   El proyecto seleccionado se abre como proyecto de referencia con protección contra escritura.

Cierre de un proyecto de referencia

Para cerrar un proyecto de referencia, proceda del siguiente modo:

1. En la paleta "Proyectos de referencia" del árbol del proyecto seleccione el proyecto de referencia que desea cerrar.
2. Haga clic en el botón "Cerrar proyecto de referencia" de la barra de herramientas.
   El proyecto de referencia seleccionado se cierra.

Consulte también

Principios básicos de los proyectos de referencia (Página 360)

Comparación de proyectos de referencia (Página 362)

Proyectos de referencia (Página 303)

7.4.3 Comparación de proyectos de referencia

Introducción

Se pueden comparar dispositivos de proyectos de referencia con dispositivos del proyecto actual, con dispositivos del mismo proyecto de referencia o de uno distinto y con dispositivos de una librería.

Nota

Tenga en cuenta las indicaciones siguientes:

- No se puede definir ninguna acción para los objetos de comparación ya que los proyectos de referencia están protegidos contra escritura.
- Para los objetos de comparación se puede realizar una comparación detallada si el tipo del objeto de comparación permite una comparación detallada en general.
- Asimismo, al comparar proyectos de referencia se puede cambiar en todo momento entre la comparación automática y la manual.

Procedimiento

Para comparar los objetos de un proyecto de referencia con los datos de dispositivo del proyecto actual, proceda del siguiente modo:

1. En el árbol del proyecto seleccione el dispositivo cuyos datos desea comparar con los datos del proyecto de referencia y que permita una comparación offline/offline.
2. En el menú contextual seleccione el comando "Comparar > offline/offline".
   El editor de comparación se abre y el dispositivo seleccionado se muestra en el área izquierda.
3. Abra la paleta "Proyectos de referencia" en el árbol del proyecto.
4. Seleccione el dispositivo del proyecto de referencia que desea comparar con los datos del dispositivo del proyecto actual.
5. Arrastre el dispositivo del proyecto de referencia a la superficie de arrastre derecha del editor de comparación.
   Los símbolos del área de estado y acción indican el estado de los objetos. Al seleccionar un objeto se muestra una sinopsis de sus propiedades y del objeto correspondiente del dispositivo asignado en una comparativa de propiedades.
   En cualquier momento se pueden arrastrar otros dispositivos del proyecto actual, de una librería o de un proyecto de referencia a las áreas de arrastre e iniciar así una nueva comparación. No importa qué dispositivo se arrastre a qué área de arrastre.

Consulte también

Principios básicos de los proyectos de referencia (Página 360)

Proyectos de referencia (Página 303)

Apertura y cierre de un proyecto de referencia (Página 361)

Realizar una comparación offline/offline (Página 374)
Utilizar el editor de comparación (Página 375)

7.5 Edición de los datos de proyecto

7.5.1 Compilación y carga de los datos de proyecto

7.5.1.1 Compilar datos de proyecto

Información general acerca de la compilación de datos de proyecto

Compilación de los datos de proyecto

Al compilar los datos de proyecto, estos se convierten de manera que el dispositivo los pueda leer. Los datos de la configuración hardware y los datos de programa se pueden compilar por separado o conjuntamente. Los datos de proyecto pueden compilarse simultáneamente para uno o varios sistemas de destino.

Los siguientes datos de proyecto deben compilarse antes de la carga:

- Datos de proyecto de hardware, p. ej. datos de configuración de los dispositivos o redes y conexiones
- Datos de proyecto de software, p. ej. bloques de programa o sinópticos del proceso

Nota

Mientras se está compilando un dispositivo no es posible iniciar ningún otro proceso de compilación. Recuerde que un proceso de compilación no solo puede iniciarse manualmente por el operador, sino que también puede ser lanzado automáticamente desde elementos de manejo y visualización.

Alcance de la compilación

Para compilar los datos de proyecto se dispone de las siguientes opciones (en función del dispositivo utilizado):

- Hardware y software (solo cambios)
- Hardware (solo cambios)
- Hardware (compilar por completo)
- Software (solo cambios)

- Software (compilar todos los bloques)
- Software (inicializar reserva de memoria)

Consulte también

Compilar datos de proyecto (Página 364)

Compilar datos de proyecto

Este capítulo describe el proceso general de compilación de datos de proyecto en el árbol del proyecto. En la ayuda del producto encontrará cómo funciona la compilación de determinados objetos y las particularidades que deben tenerse en cuenta.

Procedimiento

Para compilar los datos de proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, seleccione los dispositivos cuyos datos del proyecto desea compilar.
2. Seleccione la opción deseada del menú contextual en el submenú "Compilar".

   Nota

   Tenga en cuenta que las opciones disponibles dependen del dispositivo seleccionado.

   Los datos de proyecto se compilarán. En la ventana de inspección puede verificar bajo "Información > Compilar" si la compilación se ha realizado correctamente.

Consulte también

Información general acerca de la compilación de datos de proyecto (Página 363)

7.5.1.2 Cargar los datos de proyecto

Información general acerca de la carga

Introducción

Para configurar el sistema de automatización es necesario cargar los datos de proyecto creados offline en los dispositivos conectados. Estos datos de proyecto se generan p. ej. al configurar el hardware, las redes y las conexiones, al programar el programa de usuario o al crear recetas.

Los datos de proyecto se cargan íntegramente la primera vez que se realiza la carga. En los procesos de carga posteriores se cargan únicamente los cambios.

Los datos del proyecto pueden cargarse en dispositivos y Memory Cards.

Nota

Mientras se está compilando un dispositivo no es posible iniciar ningún otro proceso de carga. Recuerde que un proceso de compilación no solo puede iniciarse manualmente por el operador, sino que también puede ser lanzado automáticamente desde elementos de manejo y visualización.

Opciones posibles para la carga

Dependiendo del objeto que desee cargar, existen las opciones siguientes:

- Hardware y software (solo cambios)
  Tanto la configuración hardware como software se cargan en el destino si hay diferencias entre las versiones online y offline.
- Configuración hardware
  Solo se carga en el destino la configuración hardware.
- Software (solo cambios)
  Solo se cargan en el destino los objetos que difieren online y offline.
- Cargar el programa de PLC en el dispositivo e inicializar
  Todos los bloques se cargan en el destino y todos los valores se inicializan a los valores de arranque. Recuerde que lo mismo ocurre con los valores remanentes.

Incluso es posible volver a cargar en el proyecto los datos de proyecto que ya se encuentran en un dispositivo. Para ello, se dispone de las siguientes posibilidades:

- Carga del dispositivo como estación nueva
  La configuración hardware del dispositivo y el software existente en el dispositivo se cargan en el proyecto.
  El dispositivo se carga en el proyecto con todos los datos relevantes.
- Cargar el software de un dispositivo
  Únicamente se cargan en una CPU del proyecto los bloques y parámetros del dispositivo.

En ambos casos, durante la carga se enlazan nuevamente todas las instancias de los tipos de librerías con la versión adecuada del tipo de librería del proyecto. Si no existe un tipo adecuado para la instancia cargada o no existe la versión correcta del tipo en la librería del proyecto, entonces se agregará el tipo o la versión a la librería del proyecto.

Carga con sincronización

En el contexto del Team Engineering (ingeniería de equipo), puede darse el caso de que varios usuarios con varios sistemas de ingeniería trabajen en un proyecto y accedan a una CPU S7-1500 en paralelo. Para garantizar la coherencia dentro de dicho proyecto es necesario sincronizar los datos modificados antes de cargarlos para que no se sobrescriban accidentalmente.

Así pues, si durante la carga se detectan diferencias entre la gestión de datos online y offline dentro del proyecto común y estas diferencias han sido causadas por otro sistema de ingeniería, en el proceso de carga se ofrece automáticamente una sincronización de los datos que van a cargarse.

En ese caso, en el cuadro de diálogo "Sincronización" se muestran los datos que deben sincronizarse con el estado actual (comparación online/offline) y las acciones posibles.

Caso de aplicación	Recomendación	Sincronización
Uno o varios bloques son más nuevos en la CPU (online) que en el sistema de ingeniería (offline).	Antes de la carga, estos bloques deberían cargarse primero de la CPU al sistema de ingeniería.	La sincronización automática es posible: Los bloques del sistema de ingeniería se actualizan antes de la carga.
Uno o varios bloques se crean de nuevo y solo existen en la CPU (online).	Antes de la carga, estos bloques deberían cargarse primero de la CPU al sistema de ingeniería.	La sincronización automática es posible: Los nuevos bloques se agregan al sistema de ingeniería antes de la carga.
Se han borrado uno o varios bloques en la CPU.	Estos bloques también deberían borrarse en el sistema de ingeniería antes de la carga.	La sincronización automática no es posible. Los bloques borrados en la CPU deberían borrarse manualmente en el proyecto offline del sistema de ingeniería.
Uno o varios bloques son diferentes en la CPU y en el sistema de ingeniería. Esto ocurre cuando otro usuario ha modificado y cargado en la CPU bloques en los que usted también ha realizado correcciones.	Estos bloques con modificaciones rivalizantes deben adaptarse manualmente. En este caso, decida qué cambios desea aplicar. Si deben conservarse los bloques de la CPU, deberá aceptar estos bloques antes de carga de la CPU a su sistema de ingeniería. Si deben aceptarse los bloques modificados por usted, puede continuar con la carga sin sincronización.	La sincronización automática no es posible: Los bloques afectados de la CPU o el sistema de ingeniería deben adaptarse manualmente. Esto implica que se sobrescribirá una de las versiones existentes del bloque (online u offline).
Se detectan diferencias en la configuración hardware de la CPU (online) y del sistema de ingeniería (offline).	Las diferencias en la configuración hardware deben resolverse manualmente. En este caso, decida qué configuración hardware desea aplicar. Si debe conservarse la configuración hardware existente en la CPU, debe aceptarla antes de cargarla en su sistema de ingeniería. Si debe aplicarse la configuración hardware modificada por usted, puede continuar con la carga sin sincronización.	La sincronización automática no es posible: La configuración hardware debe ajustarse de forma manual. Esto implica que se sobrescribirá una de las configuraciones hardware existentes (online u offline).

El comando "Forzar la carga en dispositivo" permite cargar bloques sin sincronización, si lo desea.

Consulte también

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar los datos de proyecto en una Memory Card (Página 368)

Cargar datos de proyecto de un dispositivo (Página 369)

Cargar los datos de proyecto en un dispositivo

Este capítulo describe el proceso general de carga de datos de proyecto en un dispositivo. En la ayuda del producto encontrará cómo funciona la carga de determinados objetos y las particularidades que deben tenerse en cuenta.

Requisitos

- Los datos de proyecto son coherentes.
- A través de un acceso online se puede acceder a todo dispositivo que debe cargarse.

Procedimiento

Para cargar los datos de proyecto en dispositivos seleccionados, proceda del siguiente modo:

1. Seleccione uno o varios dispositivos en el árbol del proyecto.
2. Haga clic con el botón derecho del ratón en un elemento seleccionado.
   Se abre el menú contextual.
3. Seleccione la opción deseada del menú contextual en el submenú "Cargar en dispositivo".

   Nota

   Tenga en cuenta que las opciones disponibles dependen del dispositivo seleccionado.

   Los datos de proyecto se compilan en caso necesario.

   – Si ya ha establecido una conexión online, aparecerá el cuadro de diálogo "Vista preliminar Carga". En este cuadro de diálogo se visualizan mensajes y se proponen las acciones necesarias para la carga.
   – Si hasta ahora no ha establecido ninguna conexión online, aparecerá el diálogo "Carga avanzada" y primero deberá seleccionar las interfaces a través de las que va a establecer la conexión online con el dispositivo. Existe la posibilidad de visualizar todos los dispositivos compatibles activando la opción correspondiente y haciendo clic en el comando "Iniciar búsqueda".
     Consulte también: Auto-Hotspot
4. Verifique los mensajes en el cuadro de diálogo "Vista preliminar Carga" y, dado el caso, active las acciones en la columna "Acción".

   ⚠ **ADVERTENCIA**

   Evitar daños personales y materiales

   La realización de las acciones propuestas con la instalación en marcha puede causar graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

   Antes de activar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

   El botón "Cargar" se activará tan pronto como sea posible la carga.

5. Haga clic en el botón "Cargar".
   La carga se realizará. Si es necesario realizar una sincronización, el sistema muestra automáticamente el cuadro de diálogo "Sincronización". En él se muestran mensajes y se proponen las acciones necesarias para la sincronización. Existe la posibilidad de ejecutar estas acciones o forzar la carga sin sincronización haciendo clic en "Forzar la carga en dispositivo". En caso de haber ejecutado las acciones propuestas, el sistema le preguntará si desea continuar con la carga. A continuación se abre el diálogo "Cargar resultados". Este diálogo permite comprobar si la carga se ha realizado correctamente y, en caso necesario, seleccionar acciones adicionales.
6. Haga clic en el botón "Finalizar".

Resultado

Los datos de proyecto seleccionados se han cargado en los dispositivos.

Consulte también

Información general acerca de la carga (Página 364)

Cargar los datos de proyecto en una Memory Card (Página 368)

Cargar datos de proyecto de un dispositivo (Página 369)

Cargar los datos de proyecto en una Memory Card

Los datos del proyecto se pueden cargar en una Memory Card mediante Drag & Drop. Para las CPU de la serie S7-300/400 también es posible cargar explícitamente el programa de usuario en una Memory Card insertada en la CPU.

Requisitos

Se muestra una Memory Card.

Consulte también: Acceder a Memory Cards (Página 425)

Cargar los datos del proyecto en una Memory Card con la función drag & drop

Para cargar datos del programa en una Memory Card, proceda del siguiente modo:

1. En el árbol del proyecto, arrastre los datos de proyecto que desea cargar hasta la Memory Card.
   Se abre el cuadro de diálogo "Vista preliminar Carga". En este cuadro de diálogo se visualizan mensajes y se proponen las acciones necesarias para la carga.
2. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar" se activará tan pronto como sea posible la carga.
3. Haga clic en el botón "Cargar".
   La carga se realizará. A continuación se abre el diálogo "Cargar resultados". Este diálogo permite comprobar si la carga se ha realizado correctamente y, en caso necesario, seleccionar acciones adicionales.
4. Haga clic en el botón "Finalizar".

Cargar el programa de usuario en una Memory Card de la CPU (solo S7-300/400)

Para cargar el programa de usuario en una Memory Card de una CPU de la serie S7-300/400, proceda del siguiente modo.

1. Seleccione una CPU de la serie S7-300/400 en el árbol del proyecto.
2. Elija el comando "Cargar programa de usuario en la Memory Card" del menú "Online". Se abre el cuadro de diálogo "Vista preliminar Carga". En este cuadro de diálogo se visualizan mensajes y se proponen las acciones necesarias para la carga.
3. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción". El botón "Cargar" se activará tan pronto como sea posible la carga.
4. Haga clic en el botón "Cargar". Se ejecuta la operación de carga y aparece el cuadro de diálogo "Resultados de la operación de carga". En él se muestran mensajes y se proponen las acciones necesarias para la sincronización.
5. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción".
6. Haga clic en el botón "Finalizar".

Consulte también

Información general acerca de la carga (Página 364)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar datos de proyecto de un dispositivo (Página 369)

Cargar datos de proyecto de un dispositivo

Este capítulo describe el proceso general de carga de datos de proyecto desde un dispositivo. Qué datos de proyecto se pueden cargar desde un dispositivo depende de los productos instalados.

Al cargar datos de proyecto desde un dispositivo al propio proyecto se dispone de las opciones siguientes:

- Carga del dispositivo como estación nueva
  Esta opción permite cargar los datos de proyecto existentes de un dispositivo en el proyecto propio como estación nueva.
- Cargar los datos de proyecto de un dispositivo
  Esta opción permite cargar los datos de proyecto de un dispositivo en una CPU existente en el proyecto propio. En la ayuda del producto puede consultar qué datos de proyecto se cargarán.

En ambos casos, durante la carga se enlazan nuevamente todas las instancias de los tipos de librerías con la versión adecuada del tipo en la librería del proyecto. Si, para una instancia cargada, no existe todavía un tipo adecuado o no existe la versión correcta del tipo en la librería del proyecto, se agregará el tipo o la versión a la librería del proyecto.

Requisitos

- Hay un proyecto abierto.
- La configuración hardware que debe cargarse y el software deben ser compatibles con el TIA Portal. La compatibilidad se debe garantizar en especial si los datos del dispositivo han sido creados con una versión anterior del programa o con otro software de configuración.

Carga del dispositivo como estación nueva

Para cargar el dispositivo completo en el proyecto, proceda del siguiente modo:

1. Seleccione el nombre del proyecto en el árbol del proyecto.
2. En el menú "Online" seleccione el comando "Carga del dispositivo como estación nueva (hardware y software)".
   Se abre el diálogo "Cargar dispositivo en PG/PC".
3. Seleccione en la lista desplegable "Tipo de interfaz PG/PC" el tipo de interfaz que desea utilizar para el proceso de carga.
4. Seleccione en la lista desplegable "Interface PG/PC" la interfaz que debe utilizarse.
5. Haga clic en el botón "Configurar interfaz" a la derecha de la lista desplegable "Interface PG/PC" para adaptar los ajustes de la interfaz seleccionada.
   Consulte también: Auto-Hotspot
6. Visualice todos los dispositivos compatibles activando la opción correspondiente y haciendo clic en el comando "Iniciar búsqueda". En la tabla de dispositivos accesibles seleccione el dispositivo desde el que desea cargar los datos de proyecto.
7. Haga clic en "Cargar".
   Dependiendo del equipo seleccionado aparece un diálogo en el que debe introducir más información, como por ejemplo, la posición en el rack.
   Los datos de proyecto del dispositivo se cargan en el proyecto. Este puede editarse offline y luego volverse a cargar en el dispositivo.

Cargar los datos de proyecto de un dispositivo

Para cargar en el proyecto únicamente los datos de proyecto de un dispositivo, proceda del siguiente modo:

1. Establezca una conexión online con el dispositivo del que desea cargar los datos de proyecto.
   Consulte también: Auto-Hotspot
2. Seleccione el dispositivo en el árbol del proyecto.
   En el menú "Online" se activa el comando "Cargar de dispositivo (software)".
3. Elija el comando "Cargar de dispositivo (software)" del menú "Online".
   Se abre el cuadro de diálogo "Vista preliminar para cargar de dispositivo".

4. Verifique los mensajes en el diálogo "Vista preliminar para cargar del dispositivo" y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activará tan pronto como sea posible la carga.
5. Haga clic en el botón "Cargar de dispositivo".
   La carga se realizará.

Consulte también

Información general acerca de la carga (Página 364)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar los datos de proyecto en una Memory Card (Página 368)

Cargar datos de proyecto desde una Memory Card

Al cargar datos de proyecto en el proyecto desde una Memory Card se dispone de las opciones siguientes:

- Cargar datos de proyecto de la Memory Card como estación nueva
  Esta opción permite cargar los datos de proyecto de una Memory Card en el proyecto propio como estación nueva.
- Cargar datos de proyecto de la Memory Card en un dispositivo existente
  Esta opción permite cargar los datos de proyecto de una Memory Card en un dispositivo existente en el proyecto. En la ayuda del producto puede consultar qué datos de proyecto se cargarán.

En ambos casos, durante la carga se enlazan nuevamente todas las instancias de los tipos de librerías con la versión adecuada del tipo en la librería del proyecto. Si, para una instancia cargada, no existe todavía un tipo adecuado o no existe la versión correcta del tipo en la librería del proyecto, se agregará el tipo o la versión a la librería del proyecto.

Requisitos

- Hay un proyecto abierto.
- Se muestra la Memory Card.
  Consulte también: Acceder a Memory Cards (Página 425)
- La configuración hardware que debe cargarse y el software deben ser compatibles con el TIA Portal. La compatibilidad se debe garantizar en especial si los datos de la Memory Card han sido creados con una versión anterior del programa o con otro software de configuración.

Cargar datos de proyecto como estación nueva

Para cargar los datos de proyecto de una Memory Card en el proyecto como estación nueva, proceda del siguiente modo:

1. En el árbol del proyecto, seleccione la Memory Card cuyos datos de proyecto desea cargar.
2. En el menú "Online" seleccione el comando "Carga del dispositivo como estación nueva (hardware y software)".

O bien:

1. En el árbol del proyecto, arrastre la carpeta de la Memory Card al proyecto.

O bien:

1. Haga clic en la Memory Card con el botón derecho del ratón.
2. En el menú contextual, elija el comando "Copiar".
3. Haga clic con el botón derecho del ratón en el proyecto.
4. Elija el comando "Pegar" del menú contextual.

Cargar los datos de proyecto en un dispositivo existente

Para cargar los datos de proyecto de una Memory Card en un dispositivo existente, proceda del siguiente modo:

1. En el árbol del proyecto, arrastre la carpeta de la Memory Card a un dispositivo del proyecto o bien copie la Memory Card y pegue los datos en un dispositivo.
   Se abre el cuadro de diálogo "Vista preliminar para cargar de dispositivo".
2. Verifique los mensajes en el diálogo "Vista preliminar para cargar del dispositivo" y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activará tan pronto como sea posible la carga.
3. Haga clic en el botón "Cargar de dispositivo".
   La carga se realizará.

7.5.2 Comparar datos de proyecto

7.5.2.1 Principios básicos de la comparación de datos de proyecto

Función

Se pueden comparar datos de proyecto del mismo tipo con el fin de establecer posibles diferencias. Fundamentalmente existen los siguientes tipos de comparación:

- Comparación online/offline
  Este tipo de comparación permite comparar el software de los objetos de un dispositivo con los objetos de un proyecto. Para ello debe establecerse una conexión online con el dispositivo.
- Comparación offline/offline
  Este tipo de comparación permite comparar tanto el software como el hardware. Con la comparación de software se comparan objetos procedentes de proyectos o librerías. La comparación de hardware está disponible para dispositivos del proyecto abierto actualmente o de proyectos de referencia. Tanto para la comparación de software como para la de hardware se puede elegir si la comparación se realiza automáticamente para todos los objetos o si se desea comparar objetos individuales manualmente.
- Comparación detallada
  Para algunos objetos, como p. ej. bloques, se puede realizar una comparación detallada más allá de las comparaciones online/offline y offline/offline. Los objetos comparados se abren uno junto al otro y las diferencias se marcan.

Al establecer una conexión online tiene lugar automáticamente una comparación online-offline simple. Durante este proceso, los objetos comparables del árbol del proyecto se marcan con símbolos que indican el resultado de la comparación.

La comparación normal online/offline y offline/offline se realiza en el editor de comparación. Al comparar el software, en el editor de comparación también pueden seleccionarse acciones para objetos no idénticos.

Nota

- No todos los objetos admiten cualquier tipo de comparación. Los datos de proyecto comparables en cada tipo de comparación dependen de los productos instalados.
- Compile el programa de usuario antes de iniciar una comparación o una comparación detallada. Este proceso debe repetirse después de cada cambio en el programa durante una comparación, antes de actualizar el resultado de la comparación. De esta manera se garantiza que la comparación muestre el estado actual.

Consulte también

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realización de una comparación detallada (Página 384)

7.5.2.2 Realizar una comparación online/offline

La comparación online/offline permite comparar objetos de un dispositivo con objetos de un proyecto.

Requisitos

El árbol del proyecto está abierto.

Procedimiento

Para realizar una comparación online/offline, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación online/offline.
2. Elija el comando "Comparar > offline/online" del menú contextual.
3. Si hasta ahora no se ha establecido ninguna conexión online para ese dispositivo, aparecerá el cuadro de diálogo "Establecer conexión online". En ese caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Conectar".
   La conexión online se establece y se abre el editor de comparación.

Resultado

Se visualizan todos los objetos existentes online y offline. Los símbolos del editor de comparación y del árbol del proyecto indican el estado de los objetos. En el editor de comparación se pueden definir acciones determinadas para los objetos en función del estado de los objetos.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realización de una comparación detallada (Página 384)

7.5.2.3 Realizar una comparación offline/offline

En la comparación offline/offline existe la posibilidad de comparar datos de proyecto de dos dispositivos. La comparación es posible tanto para software como para hardware. Con la comparación de software se comparan objetos procedentes de proyectos o librerías. La comparación de hardware está disponible para dispositivos del proyecto abierto actualmente o de proyectos de referencia. Se puede distinguir si la comparación se realiza automáticamente para todos los objetos o si se desea comparar objetos manualmente.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Requisitos

El árbol del proyecto está abierto.

Procedimiento

Para realizar una comparación offline/offline, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   Todos los objetos disponibles de los dispositivos seleccionados se muestran en función de los ajustes realizados en la ficha "Software" del editor de comparación y se lleva a cabo una comparación automática. Los símbolos del editor de comparación indican el estado de los objetos. Dependiendo del estado de los objetos se pueden definir determinadas acciones.
4. Si desea realizar una comparación manual, en el área de estado y acción haga clic en el botón para conmutar entre la comparación automática y manual. Seguidamente, seleccione los objetos que desea comparar.
   Se muestra la comparación de propiedades. Los símbolos indican el estado de los objetos. Dependiendo del estado de los objetos se pueden definir determinadas acciones.
5. Si desea realizar una comparación de hardware, abra la ficha "Hardware". En caso necesario, aquí también es posible realizar una comparación manual. Sin embargo, no pueden definirse acciones.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realización de una comparación detallada (Página 384)

7.5.2.4 Utilizar el editor de comparación

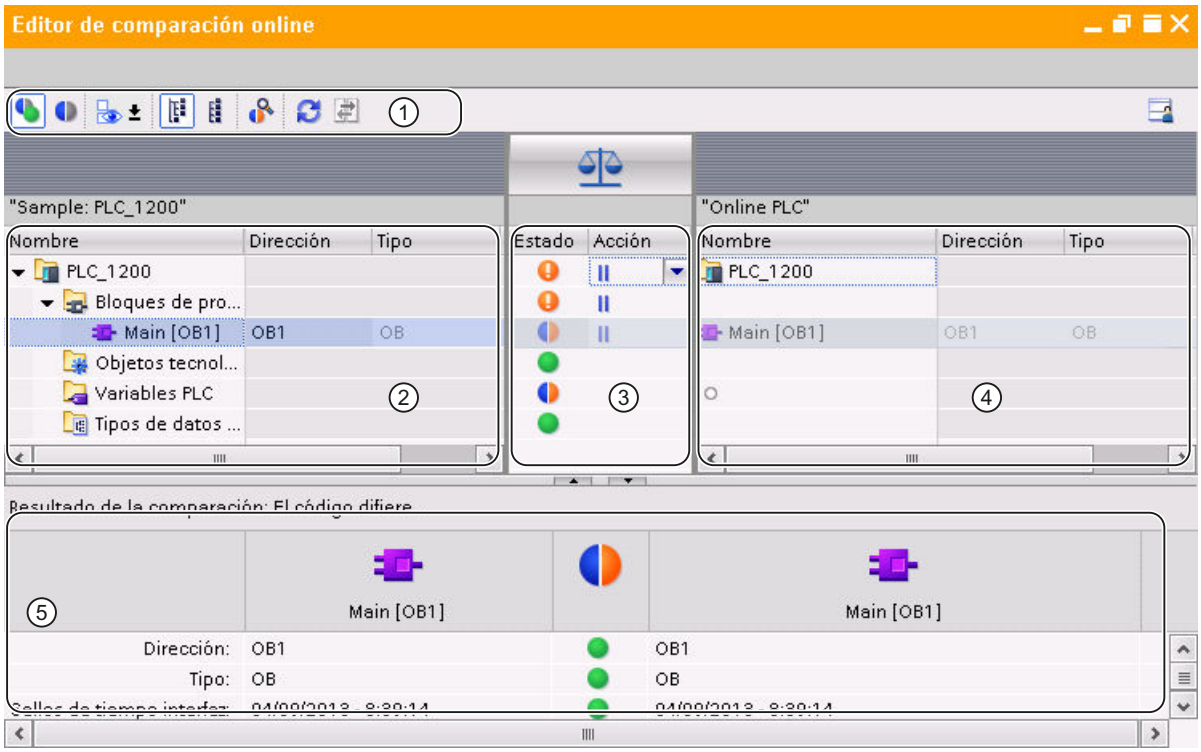
Resumen del editor de comparación

Función

El editor de comparación muestra los resultados de una comparación en una vista general en forma de tabla. El aspecto varía ligeramente en caso de comparación online/offline o comparación de hardware/software.

Estructura del editor de comparación

La siguiente figura muestra la estructura del editor de comparación en caso de comparación online/offline:



① Barra de herramientas del editor de comparación

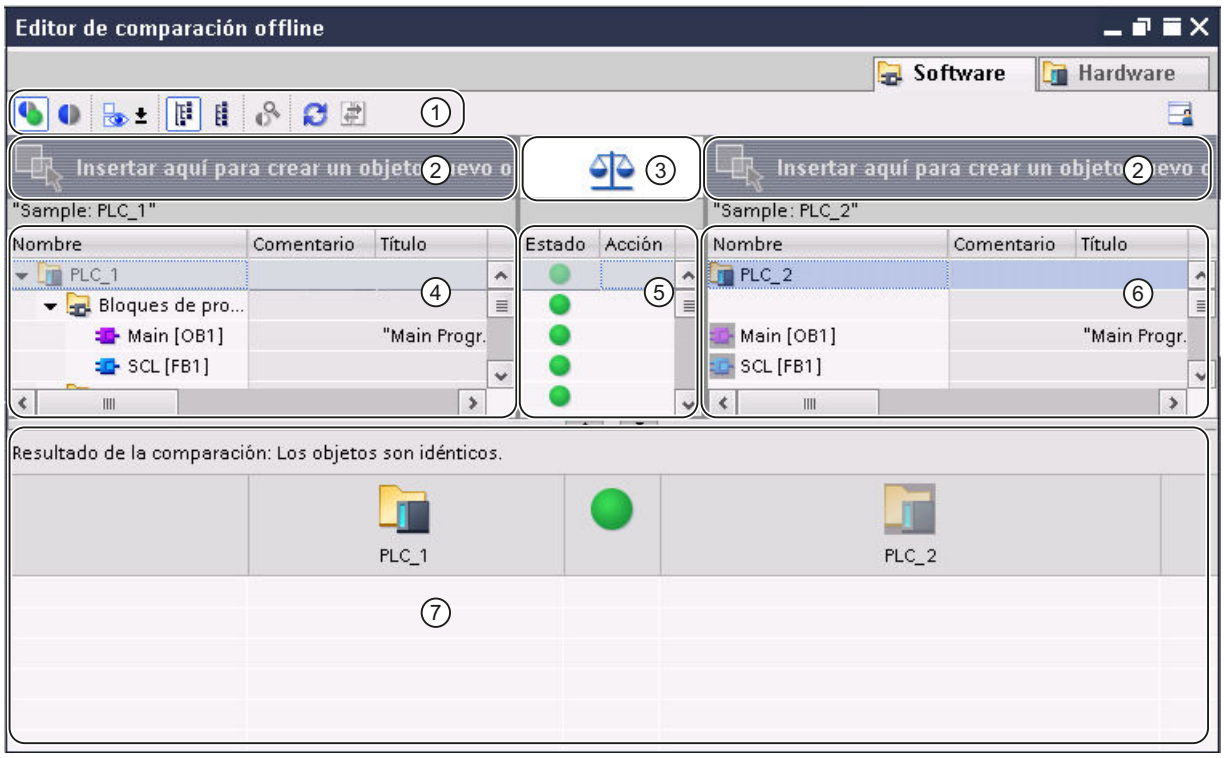
② Tabla de comparación izquierda

③ Área de estados y acciones

④ Tabla de comparación derecha

⑤ Comparación de propiedades

La siguiente imagen muestra la estructura del editor de comparación en caso de comparación offline/offline (software):



① Barra de herramientas del editor de comparación

② Superficies de arrastre

③ Botón para la conmutación entre las comparaciones automática y manual

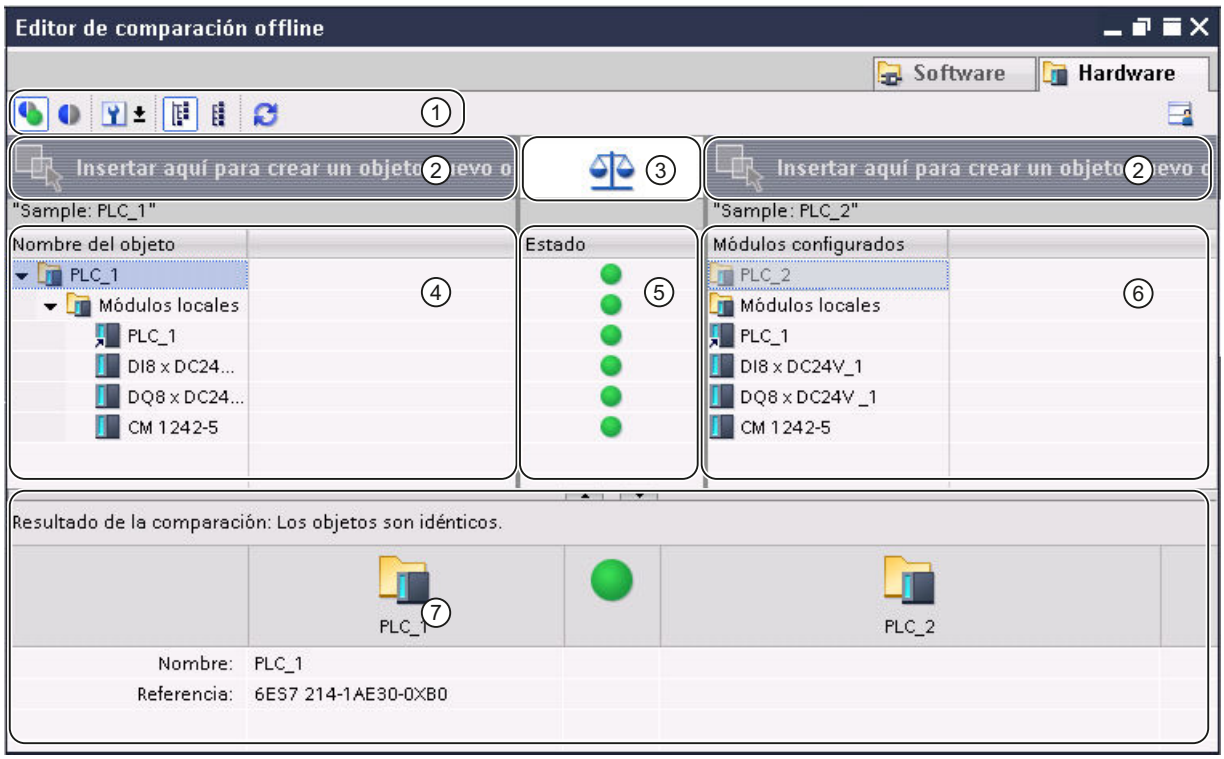
④ Tabla de comparación izquierda

⑤ Área de estado y acción

⑥ Tabla de comparación derecha

⑦ Comparación de propiedades

La siguiente imagen muestra la estructura del editor de comparación en caso de comparación offline/offline (hardware):



① Barra de herramientas del editor de comparación

② Superficies de arrastre

③ Botón para la conmutación entre las comparaciones automática y manual

④ Tabla de comparación izquierda

⑤ Área de estados

⑥ Tabla de comparación derecha

⑦ Comparación de propiedades

Barra de herramientas del editor de comparación

La barra de herramientas permite acceder a las siguientes funciones del editor de comparación:

- Mostrar solo los objetos diferentes
  A fin de mejorar la claridad, los objetos idénticos pueden ocultarse.
- Mostrar objetos idénticos y diferentes
  Si se desea visualizar la comparación completa, pueden mostrarse también los objetos idénticos.
- Alcance de la comparación (solo comparación online/offline y comparación offline/offline para software)
  Se puede definir qué objetos se deben comparar.
- Seleccionar criterio de asignación (solo comparación offline/offline para hardware)
  Se puede definir el criterio según el cual los módulos se asignarán unos a otros para la comparación.

- Modificar la vista
  Se puede elegir entre una vista jerárquica y una vista plana. En la vista jerárquica se representan los dispositivos en su estructura, mientras que en la vista plana los objetos aparecen listados sin ninguna estructura.
- Iniciar comparación detallada (solo comparación online/offline y comparación offline/offline para software)
  Es posible iniciar una comparación detallada de objetos para visualizar las distintas diferencias. Sin embargo, esta función no está disponible para todos los objetos.
- Actualizar los resultados de la comparación
  Esta función permite actualizar los resultados de la comparación después de haber modificado algún objeto.
- Sincronizar objetos no idénticos (solo comparación online/offline y comparación offline/offline para software)
  Los objetos no idénticos pueden sincronizarse mediante determinadas acciones.

Superficies de arrastre

En una comparación offline/offline, pueden arrastrarse dispositivos a las superficies de arrastre para compararlos. En una comparación de software, los dispositivos que se comparan pueden proceder del proyecto abierto, de proyectos de referencia, de la librería del proyecto o de librerías globales. Sin embargo, debe tener en cuenta que las librerías enteras solo pueden arrastrarse a la superficie de arrastre derecha. En una comparación de hardware, pueden compararse dispositivos del proyecto abierto o de proyectos de referencia.

Botón para conmutar entre las comparaciones automática y manual

En una comparación offline/offline puede conmutar entre comparación automática y manual. En la comparación automática, los objetos que se comparan se asignan unos a otros automáticamente; en la comparación manual es posible seleccionar los objetos que deben compararse.

Tablas de comparación

En las tablas de comparación se muestran los objetos de los dispositivos que se comparan entre sí.

En la tabla siguiente se explica el significado de las columnas de las tablas de comparación:

Columna	Significado
Nombre	Nombre del objeto de comparación
Comentario	Comentario sobre el objeto de comparación
Título	Título del objeto de comparación
Dirección	Dirección del objeto de comparación
Numeración	Tipo de numeración para el objeto de comparación
Tipo	Tipo del objeto de comparación
Lenguaje	Lenguaje de programación ajustado para el objeto de comparación.
Sello de tiempo Interfaz	Momento del último cambio de la interfaz del bloque

Columna	Significado
Sello de tiempo código	Momento del último cambio del código fuente
Autor	Nombre del autor del objeto de comparación
Versión	Versión del objeto de comparación
Familia	Nombre de la familia del objeto
Memoria de carga	Carga de la memoria de carga del objeto de comparación
Memoria de trabajo	Carga de la memoria de trabajo del objeto de comparación
Fecha de la última modificación	Momento de la última modificación
Firma	Firma del objeto de comparación (SIMATIC Safety)
Firma de interfaz	Firma de la interfaz de bloque del objeto de comparación (SIMATIC Safety)

No todas las columnas están visibles de forma estándar Sin embargo, como en todos los editores de tablas, las columnas se pueden mostrar u ocultar según sea necesario y clasificarse por columnas individuales. En la comparación de hardware, las tablas de comparación solo contienen la columna "Nombre".

Área de estado y acción

El área de estado y acción ofrece las siguientes posibilidades:

- Se pueden visualizar los resultados de una comparación automática. Los resultados se muestran con símbolos.
- En la comparación online/offline y en la comparación offline/offline para software, pueden definirse acciones para objetos no idénticos.

Símbolos de estado y acción

La siguiente tabla muestra los símbolos utilizados para los resultados de una comparación online/offline:

Símbolo	Significado
	La carpeta contiene los objetos cuyas versiones online y offline son diferentes
	El resultado de la comparación no se conoce
	Las versiones online y offline del objeto son idénticas
	Las versiones online y offline del objeto son diferentes
	El objeto solo existe offline
	El objeto solo existe online

La siguiente tabla muestra los símbolos utilizados para los resultados de una comparación offline/offline:

Símbolo	Significado
	Programa de partida
	Versión comparada
	La carpeta contiene objetos cuyas versiones comparadas son diferentes
	El resultado de la comparación offline/offline no se conoce
	Las versiones comparadas del objeto son idénticas
	Las versiones comparadas del objeto son diferentes
	El objeto solo existe en el programa de partida
	El objeto solo existe en la versión comparada
	Solo comparación de hardware: los objetos subordinados del contenedor son idénticos, pero hay diferencias entre los propios contenedores. Un contenedor puede ser un rack, p. ej.
	Solo comparación de hardware: los objetos subordinados del contenedor son diferentes. Además hay diferencias entre los contenedores. Un contenedor puede ser un rack, p. ej.

La siguiente tabla muestra los símbolos utilizados para las posibles acciones en una comparación de software:

Símbolo	Significado
	Ninguna acción
	Sobrescribir el objeto de la versión comparada con el objeto del programa de partida
	Sobrescribir el objeto del programa de partida con el objeto de la versión comparada
	Acciones diferentes para los objetos comparados contenidos en la carpeta

Comparación de propiedades

La comparación de propiedades compara las propiedades de los objetos seleccionados. El resultado se muestra con símbolos. En la comparación manual solo se comparan las propiedades, por lo que el área de estado y acción permanece vacía. En la comparación automática se puede utilizar la comparación de propiedades además de la comparación de las tablas de comparación.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Modificación de la vista (Página 388)

Mostrar y ocultar las columnas de la tabla (Página 382)

Filtrar la vista en el editor de comparación (Página 383)

Actualizar resultados de la comparación (Página 385)

Sincronizar objetos diferentes (Página 386)

Mostrar y ocultar las columnas de la tabla

En una comparación de software, las columnas de las tablas de comparación pueden mostrarse u ocultarse según sea necesario.

Procedimiento

Para mostrar u ocultar las columnas de una tabla, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna.
2. Elija el comando "Mostrar/ocultar" del menú contextual.
   Aparece una selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación correspondiente.
4. Para ocultar una columna, desactive la casilla de verificación correspondiente.

La modificación tiene efecto tanto en la tabla de comparación izquierda como en la derecha.

Resultado

Las columnas se muestran u ocultan tanto en la tabla de comparación izquierda como en la derecha.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Filtrar la vista en el editor de comparación (Página 383)

Realización de una comparación detallada (Página 384)

Actualizar resultados de la comparación (Página 385)

Sincronizar objetos diferentes (Página 386)

Modificación de la vista (Página 388)

Filtrar la vista en el editor de comparación

Para mejorar la claridad del editor de comparación se puede limitar la visualización mediante los siguientes mecanismos de filtro:

- Ocultar objetos comparados idénticos
  Es posible ocultar objetos comparados que sean idénticos en las versiones online/offline u offline/offline. Por otra parte, los objetos de este tipo se pueden volver a mostrar en cualquier momento si están ocultos.
- Objetos visualizados
  En la comparación online/offline o la comparación offline/offline para software, puede definirse para qué objetos deben mostrarse los resultados de la comparación.

Requisitos

El editor de comparación está abierto.

Ocultar objetos comparados idénticos

Para ocultar objetos idénticos, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar sólo los objetos que difieren" de la barra de herramientas.
   Se visualizan sólo los elementos que son diferentes online y offline.

Mostrar objetos comparados idénticos

Para volver a mostrar los objetos idénticos, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar objetos idénticos y diferentes" de la barra de herramientas.
   Se muestran todos los elementos existentes.

Seleccionar los objetos visualizados

Para seleccionar los objetos para los que se deben visualizar los resultados de la comparación, proceda del siguiente modo:

1. Realice una comparación online/offline u offline/offline para software.
2. En la barra de herramientas, haga clic en la flecha del botón "Mostrar otros filtros disponibles".
3. Seleccione el filtro que desee utilizar.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Modificación de la vista (Página 388)

Mostrar y ocultar las columnas de la tabla (Página 382)

Realización de una comparación detallada (Página 384)

Actualizar resultados de la comparación (Página 385)

Sincronizar objetos diferentes (Página 386)

Realización de una comparación detallada

Nota

No todos los objetos admiten una comparación detallada. Los datos de proyecto que se pueden comparar detalladamente dependen de los productos instalados. En la comparación de hardware no es posible comparar detalladamente los componentes de hardware.

Procedimiento

Para realizar una comparación detallada, proceda del siguiente modo:

1. Realice primero una comparación online/offline u offline/offline para software.
   El editor de comparación se abre.

   Nota

   Solo es posible realizar una comparación detallada con objetos que se encuentran tanto en la tabla de comparación izquierda como en la derecha.

2. Seleccione en el editor de comparación el objeto con el que desea realizar una comparación detallada.
3. Haga clic en el botón "Iniciar comparación detallada" de la barra de herramientas.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Mostrar y ocultar las columnas de la tabla (Página 382)

Modificación de la vista (Página 388)

Filtrar la vista en el editor de comparación (Página 383)

Actualizar resultados de la comparación (Página 385)

Sincronizar objetos diferentes (Página 386)

Actualizar resultados de la comparación

En cuanto se modifica un objeto, los resultados de la comparación dejan de ser válidos y deben actualizarse.

Nota

En la comparación online/offline puede ocurrir que el sistema actualice el editor de comparación automáticamente debido a modificaciones en el dispositivo, si algún objeto de la comparación se ve afectado por el cambio en cuestión. Esto puede tener las siguientes consecuencias:

- Posiblemente, algunas acciones definidas no sean válidas, p. ej. en caso de que el objeto ya no exista en el dispositivo. Los objetos con tales acciones no válidas se resaltan para que se pueda definir una nueva acción válida.
- Posiblemente se corrija la selección que se había establecido antes de la actualización automática.

Requisitos

El editor de comparación está abierto.

Procedimiento

Para actualizar los resultados de la comparación, proceda del siguiente modo:

1. Haga clic en el botón "Actualizar la vista" de la barra de herramientas.
   Los resultados de comparación se actualizarán.

   Nota

   Tenga en cuenta que el botón "Actualizar la vista" no está disponible mientras el editor de comparación carga o sincroniza el contenido.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Mostrar y ocultar las columnas de la tabla (Página 382)

Modificación de la vista (Página 388)

Filtrar la vista en el editor de comparación (Página 383)

Realización de una comparación detallada (Página 384)

Sincronizar objetos diferentes (Página 386)

Sincronizar objetos diferentes

Definir acciones

Tras haber realizado una comparación, se pueden definir en el editor de comparación qué acciones deben aplicarse a los objetos que no sean idénticos. Para los objetos idénticos no se puede seleccionar ninguna acción. Tenga en cuenta que en la comparación de hardware no es posible ejecutar acciones.

En una comparación online/offline solo están permitidas acciones de sincronización en un sentido, para conservar la coherencia del programa. Por ejemplo, es posible cargar varios bloques en un dispositivo o desde un dispositivo, pero no mezclar acciones de carga diferentes en una misma acción de sincronización. La primera acción que se ajusta en el editor de comparación determina el sentido de sincronización. Por ejemplo, si en un bloque se ajusta que el bloque offline debe cargarse en el dispositivo, mediante una sincronización solo es posible cargar los demás objetos en este mismo dispositivo. Para volver a cargar objetos desde el dispositivo, ajuste la opción "Ninguna acción", y seguidamente ya será posible ajustar la acción que sea necesaria. También es posible realizar una nueva comparación.

Nota

Tenga en cuenta las siguientes particularidades específicas de la CPU al definir acciones:

- S7-300/400:
  - Es posible definir acciones para la carpeta "Bloques de programa", para carpetas creadas por el usuario o para bloques individuales.
  - No es posible cargar bloques SCL y GRAPH desde el dispositivo al proyecto offline.
- S7-1200/1500:
  - Es posible definir acciones para la carpeta "Bloques de programa", para carpetas creadas por el usuario o para bloques individuales. Si se ha realizado una comparación online/offline y se elige como acción la carga en el dispositivo, se lleva a cabo una carga coherente. En cambio, si la acción consiste en cargar el objeto desde el dispositivo al proyecto, también es posible cargar bloques individuales.
  - No es posible cargar bloques SCL desde el dispositivo al proyecto offline.

Requisitos

El editor de comparación está abierto.

Procedimiento

Para seleccionar una acción para un objeto que no sea idéntico, proceda del siguiente modo:

1. En el área de estado y acción haga doble clic en la columna "Acción" en la celda del objeto para el que desea definir una acción.
   La celda se convierte en una lista desplegable.
2. Haga clic en la lista desplegable.

3. Seleccione la acción deseada.
   La acción definida para el objeto se ejecuta en la próxima sincronización.
   Si se ha modificado involuntariamente la selección previamente definida para la acción, la modificación se puede deshacer hasta la próxima sincronización.
4. Para restablecer la selección previamente definida para la acción, haga clic en el objeto o la carpeta con el botón derecho del ratón.
5. Elija el comando "Restablecer la última selección" del menú contextual.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Mostrar y ocultar las columnas de la tabla (Página 382)

Filtrar la vista en el editor de comparación (Página 383)

Actualizar resultados de la comparación (Página 385)

Sincronizar objetos (Página 387)

Sincronizar objetos

Al realizar una sincronización se ejecutan las acciones definidas para los objetos no idénticos. No obstante, tenga en cuenta que en una comparación online/offline, en una acción de sincronización solo es posible ejecutar acciones en un sentido.

Requisitos

- El editor de comparación está abierto.
- Las acciones deseadas están seleccionadas.

Procedimiento

Para sincronizar los objetos, proceda del siguiente modo:

1. Haga clic en el botón "Ejecutar acciones" de la barra de herramientas.

Resultado

Se ejecutan las acciones definidas para los objetos.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Mostrar y ocultar las columnas de la tabla (Página 382)

Filtrar la vista en el editor de comparación (Página 383)

Actualizar resultados de la comparación (Página 385)

Definir acciones (Página 386)

Modificación de la vista

Se puede elegir entre una vista jerárquica y una vista plana para la tabla de comparación izquierda. En la vista jerárquica se representan los dispositivos en su estructura, mientras que en la vista plana los objetos aparecen listados sin ninguna estructura. En la tabla de comparación derecha, los objetos siempre se muestran en una vista plana.

Ajustar la vista jerárquica

Para ajustar la vista jerárquica, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar jerárquicamente" de la barra de herramientas del editor de comparación.

Ajustar la vista plana

Para ajustar la vista plana, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar en vista plana" de la barra de herramientas del editor de comparación.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Resumen del editor de comparación (Página 375)

Mostrar y ocultar las columnas de la tabla (Página 382)

Filtrar la vista en el editor de comparación (Página 383)

Realización de una comparación detallada (Página 384)

Actualizar resultados de la comparación (Página 385)

Sincronizar objetos diferentes (Página 386)

7.5.3 Proteger los datos de proyecto

7.5.3.1 Protección de los datos de proyecto

Introducción

Los datos de proyecto pueden protegerse contra accesos no autorizados. A saber, p. ej.:

- Protección de acceso para dispositivos
- Protección de copia y visualización de objetos
- Restricciones para la impresión de objetos con protección de know-how

Si un objeto se equipa con protección de know-how, esta protección permanece vigente incluso después de insertarlo en una librería. Observe que no todos los mecanismos de protección están disponibles para todos los objetos. En la ayuda del producto encontrará la forma de proteger determinados objetos.

Borrar derechos de acceso a dispositivos

Si desea ejecutar una función que está protegida por contraseña con el nivel de protección del dispositivo, aparecerá un diálogo en el que se solicita la introducción de la contraseña. Si la contraseña es correcta, se podrá ejecutar la función deseada. El derecho de acceso al dispositivo permanece activado hasta cerrar el TIA Portal.

Si desea activar nuevamente la protección por contraseña estando abierto el TIA Portal, es posible borrar los derechos de acceso explícitamente para un dispositivo. Con ello, para poder ejecutar determinadas funciones en el dispositivo protegido será necesario introducir nuevamente la contraseña correcta. Al parametrizar el nivel de protección del dispositivo se define para qué funciones es necesario introducir una contraseña.

Consulte también

Imprimir los datos de proyecto (Página 407)

7.5.3.2 Borrado de los derechos de acceso a dispositivos

Requisitos

- Para el dispositivo se ha ajustado un nivel de protección.
- Una función protegida ha sido habilitada para el dispositivo mediante la entrada de una contraseña.

Procedimiento

Para borrar los derechos de acceso al dispositivo, proceda del siguiente modo:

1. En el árbol del proyecto, seleccione el dispositivo para cuyos derechos de acceso desea borrar.
2. Elija el comando "Borrar derechos de acceso" del menú "Online".

Resultado

Se borran los derechos de acceso y para todas las funciones del dispositivo que están protegidas por contraseña se abre a partir de dicho momento el diálogo de solicitud de entrada de contraseña. La función sólo se puede ejecutar introduciendo la contraseña correcta.

Si el dispositivo tiene establecida una conexión online, se desconecta la misma.

Consulte también

Protección de los datos de proyecto (Página 389)

7.5.4 Imprimir contenidos del proyecto

7.5.4.1 Imprimir documentación del proyecto

Función de documentación

Introducción

Una vez creado el proyecto, es posible imprimir su contenido de forma clara y comprensible. Es posible imprimir el proyecto completo o bien objetos determinados. Un impreso bien estructurado facilita tanto el procesamiento ulterior del proyecto como los trabajos de mantenimiento y reparación. El impreso también puede servir de presentación para el cliente o como documentación completa de la instalación.

El proyecto se puede preparar en forma de manuales normalizados e imprimir con un diseño unificado. El alcance de la impresión se puede limitar. Es posible imprimir el proyecto completo, objetos individuales con sus propiedades o bien un resumen compacto del proyecto. Además se puede imprimir el contenido de un editor que esté abierto.

Mejorar el impreso mediante marcos y portadas

Es posible personalizar el aspecto del impreso, p. ej. seleccionando el logotipo de la empresa o aplicando el diseño corporativo de la empresa en la documentación del proyecto. Pueden crearse tantas variantes de diseño como bordes y cubiertas. Los bordes y las cubiertas se guardan en la entrada "Configuración del documento" del árbol del proyecto y forman parte del proyecto. Dentro de los bordes y las cubiertas se pueden insertar comodines para

visualizar datos del documento previamente guardados. Éstos se rellenan durante la impresión automáticamente con los metadatos correspondientes.

Si no desea utilizar las opciones de diseño personalizado, hay disponibles marcos y portadas preconfeccionados. incluyendo también plantillas según el estándar ISO para la documentación técnica de productos.

Estructura modular de un impreso

Un impreso se compone por lo general de las partes siguientes:

- Portada (sólo en caso de imprimir desde el árbol del proyecto)
- Tabla de contenido (sólo en caso de imprimir desde el árbol del proyecto)
- Nombre y ruta de un objeto del árbol del proyecto
- Datos de objetos

La impresión de la portada o de la tabla de contenido se puede desactivar en el diálogo "Imprimir".

Consulte también

Crear marcos (Página 396)

Crear portada (Página 397)

Editar portadas y marcos (Página 398)

Introducir información del documento (Página 394)

Función de impresión para tiras rotulables de módulos (Página 409)

Impresión de contenidos del proyecto

Disponibilidad de la función de impresión

Pueden imprimirse los contenidos siguientes:

- Un proyecto completo en el árbol del proyecto
- Uno o varios objetos pertenecientes a un proyecto del árbol del proyecto
- Contenidos de un editor
- Tablas
- Librerías
- Vista de diagnóstico de la ventana de inspección

En las áreas siguientes no es posible imprimir:

- Vista del portal
- Vista detallada
- Ventana general

- Editor de comparación
- Todas las fichas de la ventana de impresión excepto la vista de diagnóstico
- Todas las Task Cards excepto las librerías
- La mayoría de diálogos
- Propiedades y dispositivos de la PG o el PC que no pertenecen al proyecto, como accesos online y lectores de tarjetas conectados.

Alcance de la impresión

Para poder imprimir, tiene que estar seleccionado al menos un elemento imprimible.

Si se imprime un objeto seleccionado, también se imprimirán todos los objetos subordinados. Si por ejemplo se selecciona un dispositivo en el árbol del proyecto, también se imprimirán todos los datos pertenecientes al dispositivo. Si se selecciona el proyecto completo para la impresión en el árbol del proyecto, se imprimirán todos los contenidos del proyecto excepto las vistas gráficas. Estas deben imprimirse por separado. Las entradas del árbol del proyecto que no pertenecen al proyecto no pueden imprimirse. Entre ellas se incluyen, por ejemplo, los accesos online y los lectores de tarjeta y memorias USB conectados.

Al imprimir el contenido de una tabla se imprimirán todas las filas de la tabla en las que esté marcada una celda. Para imprimir una o varias columnas de una tabla, deben estar marcadas las columnas deseadas. Si no están marcadas celdas o columnas individuales, se imprimirá la tabla completa.

Restricciones al imprimir

Por lo general pueden imprimirse todos los objetos que también se pueden visualizar en la interfaz de usuario. Y por extensión, los objetos a los que no se puede acceder, tampoco se pueden imprimir. Si no es posible realizar una impresión, puede ser por los motivos siguientes:

- No está disponible una licencia válida para la visualización de un objeto.
- No existe ninguna descripción de dispositivo relativa a un objeto.
- Falta por instalar un componente de software necesario para visualizar un objeto.

Consulte también

Imprimir los datos de proyecto (Página 407)

Modificación de los ajustes de impresión

Modificación de los ajustes de impresión

Es posible efectuar ajustes generales de impresión que permanezcan vigentes incluso tras cerrar y abrir nuevamente el TIA Portal. Algunos ajustes dependen de los productos instalados. No obstante, los ajustes siguientes se pueden efectuar siempre:

Imprimir siempre los datos de tabla en parejas de valores

Si esta función está activada, las tablas no se imprimen en forma tabular sino mediante parejas formadas por clave y valor.

Ejemplo:

Nombre del objeto	Propiedad 1	Propiedad 2
Objeto A	Valor A1	Valor A2
Objeto B	Valor B1	Valor B2

En este caso, el impreso tendría el siguiente aspecto:

Objeto A

Propiedad 1: Valor A1

Propiedad 2: Valor A2

Objeto B

Propiedad 1: Valor B1

Propiedad 2: Valor B2

Imprimir editores de máscaras

- Imprimir datos siempre en forma tabular
  Todos los parámetros de objetos tecnológicos se imprimen en forma de tabla.
- Imprimir gráficos de máscaras si es posible
  Si el editor utilizado soporta esta función, los contenidos del editor no se imprimen como tabla sino como gráfico completo tal y como el mismo aparece en pantalla.

Procedimiento

Para cambiar los ajustes de impresión, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General".
3. Active los ajustes predeterminados deseados en el área "Ajustes de impresión".
   Los cambios se aplican inmediatamente y permanecen vigentes para todos los proyectos incluso tras cerrar el TIA Portal.

Consulte también

Resumen de los ajustes de impresión (Página 282)

Definición del diseño de impresión

Definir el diseño de impresión

Si no desea utilizar las plantillas de impresión preconfeccionadas, puede definir libremente una portada o el diseño de las distintas páginas. Los diseños se guardarán siempre junto con el proyecto correspondiente.

Los diseños de la portada y las plantillas para el diseño de página se encuentran en el grupo "Información del documento" en el árbol del proyecto. Allí encontrará también, en la entrada "Información del documento", los metadatos del proyecto. Para realizar la impresión final es posible confeccionar una presentación individual a partir de las portadas guardadas, las plantillas para el diseño de página y los metadatos en el cuadro de diálogo "Imprimir".

Personalizar la portada

El diseño de la portada se puede personalizar. Se puede insertar un gráfico de fondo y dotar la página con comodines para texto. Durante la impresión, los comodines se rellenan automáticamente con datos provenientes de la información del documento.

Las portadas se encuentran en el grupo "Información del documento > Portadas" del árbol del proyecto.

Personalizar la página de contenidos

Las páginas correlativas de un impreso pueden contener los elementos siguientes:

- Marcos con contenido estático, p. ej. el logotipo de la empresa
- Comodines para texto, p. ej. el nombre del proyecto, el número de página o el instante de inicio de la impresión.
  Los valores de los distintos comodines se definen en la información del documento. Otros valores, como p. ej. el nombre del proyecto, están predefinidos y se insertan automáticamente durante la impresión.
- Pie de página
  El pie de página se visualiza siempre por debajo del área del contenido.
- Área para el contenido
  Es posible definir un área dentro del cual debe incrustarse el contenido de la impresión.

El diseño de las páginas de contenido se guarda en marcos. Los distintos marcos se encuentran en el grupo "Información del documento > Marcos" del árbol del proyecto.

Introducir información del documento

Para cada proyecto es posible introducir metadatos en la información del documento. En la información del documento se definen además un marco para la impresión y una portada. Si es necesario, se pueden crear varias informaciones del documento para poder cambiar rápidamente entre éstas y sus diferentes informaciones, marcos, portadas, tamaños de página y orientaciones durante la impresión. Esto también es muy útil, por ejemplo, si se desea imprimir en distintos idiomas y para cada idioma se han previsto diferentes informaciones del documento.

En el editor de documentación se pueden definir comodines en la portada o en el marco de las páginas correlativas. Estos comodines se pueden rellenar automáticamente con los metadatos provenientes de la información del documento durante la impresión.

Todas las informaciones del documento forman parte de la función de impresión y determinan el diseño y los contenidos de la impresión.

Procedimiento

Para agregar metadatos, proceda del siguiente modo:

1. Haga doble clic en "Agregar información del documento" bajo "Información del documento > Información del documento" del árbol del proyecto para crear una nueva información. Las nuevas informaciones del documento se crean y se abren inmediatamente.
2. Introduzca en el campo "Nombre" un nombre para el juego de datos.
3. Rellene los distintos campos con los metadatos del proyecto.

Gestionar portadas y marcos

Uso de portadas y marcos

Ventaja de las portadas

Anteponiendo una portada al impreso de la documentación de la instalación éste obtiene un aspecto profesional. La portada se puede diseñar libremente o también se puede recurrir a las portadas preconfeccionadas. Las portadas preconfeccionadas se pueden adaptar y utilizar nuevamente como plantilla.

Es posible guardar las portadas en librerías globales, de manera que las mismas quedan disponibles para todos los proyectos.

Las portadas están previstas siempre como página de impresión derecha.

Ventaja de los marcos

Las páginas correlativas de la documentación de la instalación se pueden incrustar en un marco que permanece constante. El marco puede contener comodines para metadatos del proyecto que están guardados en la información del documento. Asimismo, puede contener elementos gráficos de diseño libre.

Es posible crear marcos personalizados así como utilizar marcos preconfeccionados. Los marcos preconfeccionados se pueden adaptar y luego guardar nuevamente como marcos nuevos.

Al igual que las portadas, los marcos se pueden guardar en librerías globales, de manera que los mismos quedan disponibles para todos los proyectos.

Los marcos están previstos siempre como página de impresión derecha.

Portadas y plantillas en el árbol del proyecto

Las portadas y los marcos pertenecientes al proyecto se guardan en la entrada "Información del documento" del árbol del proyecto. Allí existen carpetas para marcos y portadas, respectivamente.

Las operaciones siguientes están disponibles en el árbol del proyecto y se pueden utilizar para portadas y marcos:

- Crear subcarpetas propias
- Copiar y pegar
- Pegar portadas y marcos de la librería del sistema "Plantillas de documentación".
- Copiar portadas y plantillas en una librería global

Portadas y plantillas en librerías

En la librería del sistema "Plantillas de documentación" están guardadas algunas portadas y plantillas que están disponibles en cualquier proyecto. Las portadas y plantillas se pueden arrastrar desde allí hasta el árbol del proyecto mediante Drag & Drop. En el árbol del proyecto finalmente se pueden adaptar las portadas y las plantillas al proyecto.

Las portadas y plantillas se pueden arrastrar desde el árbol del proyecto hasta una librería global. Luego están disponibles en cualquier proyecto.

Consulte también

Principios básicos de las librerías (Página 427)

Sinopsis de la Task Card "Librerías". (Página 429)

Personalizar portadas y marcos (Página 398)

Utilizar marcos y portadas prediseñados (Página 398)

Crear marcos

Para cada proyecto se pueden crear tantos marcos como se requieran. Los marcos se guardan en el grupo "Información del documento > Marcos" del árbol del proyecto. A cada información del documento se le puede asignar un marco. Si para la impresión se opta por una información del documento en particular, se utiliza el marco correspondiente.

Procedimiento

Para crear un marco nuevo, proceda del siguiente modo:

1. Haga doble clic en la entrada "Agregar marco" del grupo "Información del documento > Marcos" del árbol del proyecto.
   Se abre el diálogo "Crear marco".
2. Introduzca en el campo "Nombre" un nombre para el marco.

3. Seleccione el tamaño de papel de la lista desplegable "Tipo de papel".
4. Elija en la lista desplegable "Orientación" si se desea crear la página en formato vertical u horizontal.

Haga clic en el botón "Agregar".

Resultado

Se crea un marco nuevo. Luego el marco se abre automáticamente en el editor de documentación y puede editarse allí.

Consulte también

Editar portadas y marcos (Página 398)

Crear portada (Página 397)

Crear portada

Para cada proyecto se pueden crear tantas portadas como se requieran para la impresión. Las portadas se guardan en el grupo "Información del documento > Portadas" del árbol del proyecto. A cada información del documento se le puede asignar una portada. Si para la impresión se opta por una información del documento en particular, se utiliza la portada correspondiente.

Procedimiento

Para crear una portada nueva, proceda del siguiente modo:

1. Haga doble clic en la entrada "Agregar portada" del grupo "Información del documento > Portadas" del árbol del proyecto.
   Se abrirá el cuadro de diálogo "Agregar portada".
2. Introduzca en el campo "Nombre" un nombre para la portada.
3. Seleccione el tamaño de papel de la lista desplegable "Tipo de papel".
4. Elija en la lista desplegable "Orientación" si se desea crear la página en formato vertical u horizontal.

Haga clic en el botón "Agregar".

Resultado

Se crea una portada nueva. Luego la portada se abre automáticamente en el editor de documentación y puede editarse allí.

Consulte también

Editar portadas y marcos (Página 398)

Crear marcos (Página 396)

Utilizar marcos y portadas prediseñados

El TIA Portal incorpora ya algunos marcos y portadas que pueden modificarse a discreción.

Procedimiento

Para insertar y editar los marcos y portadas prediseñados, proceda del siguiente modo:

1. Abra la paleta "Librerías globales" en la Task Card "Librerías".
2. En la carpeta "Plantillas maestras" abra la carpeta "Cover Pages" para portadas o "Frames" para marcos.
3. Utilice la función de drag & drop para desplazar una portada o un marco desde una de las dos carpetas hasta una de las carpetas siguientes del árbol del proyecto:
   – Para marcos: "Información del documento > Marco"
   – Para portadas: "Información del documento > Portadas".

   El marco o la portada prediseñados pueden utilizarse ahora en el proyecto.
4. Haga doble clic en la nueva entrada del árbol del proyecto para editar el marco o la portada.

Consulte también

Uso de portadas y marcos (Página 395)

Editar portadas y marcos (Página 398)

Personalizar portadas y marcos

Editar portadas y marcos

El editor de documentación es un editor gráfico y sirve para diseñar marcos y portadas para la documentación de la instalación. En el editor de documentación puede colocar las imágenes o elementos de texto en los marcos y portadas. Los elementos de texto son estáticos o bien, durante la impresión, se rellenan automáticamente con los datos provenientes de la información del documento que haya seleccionado en el diálogo de impresión.

Procedimiento

Para editar una portada o un marco en el editor de documentación, proceda del siguiente modo:

1. Haga doble clic en la entrada correspondiente a una portada o un marco existente en el grupo "Información del documento > Marcos" o "Información del documento > Portada" del árbol del proyecto.
   Se abre el editor de documentación.
2. Diseñe la portada o el marco libremente según sus necesidades.
3. Cierre el editor de documentación.
   Los cambios efectuados en la portada o en el marco se aplican automáticamente.

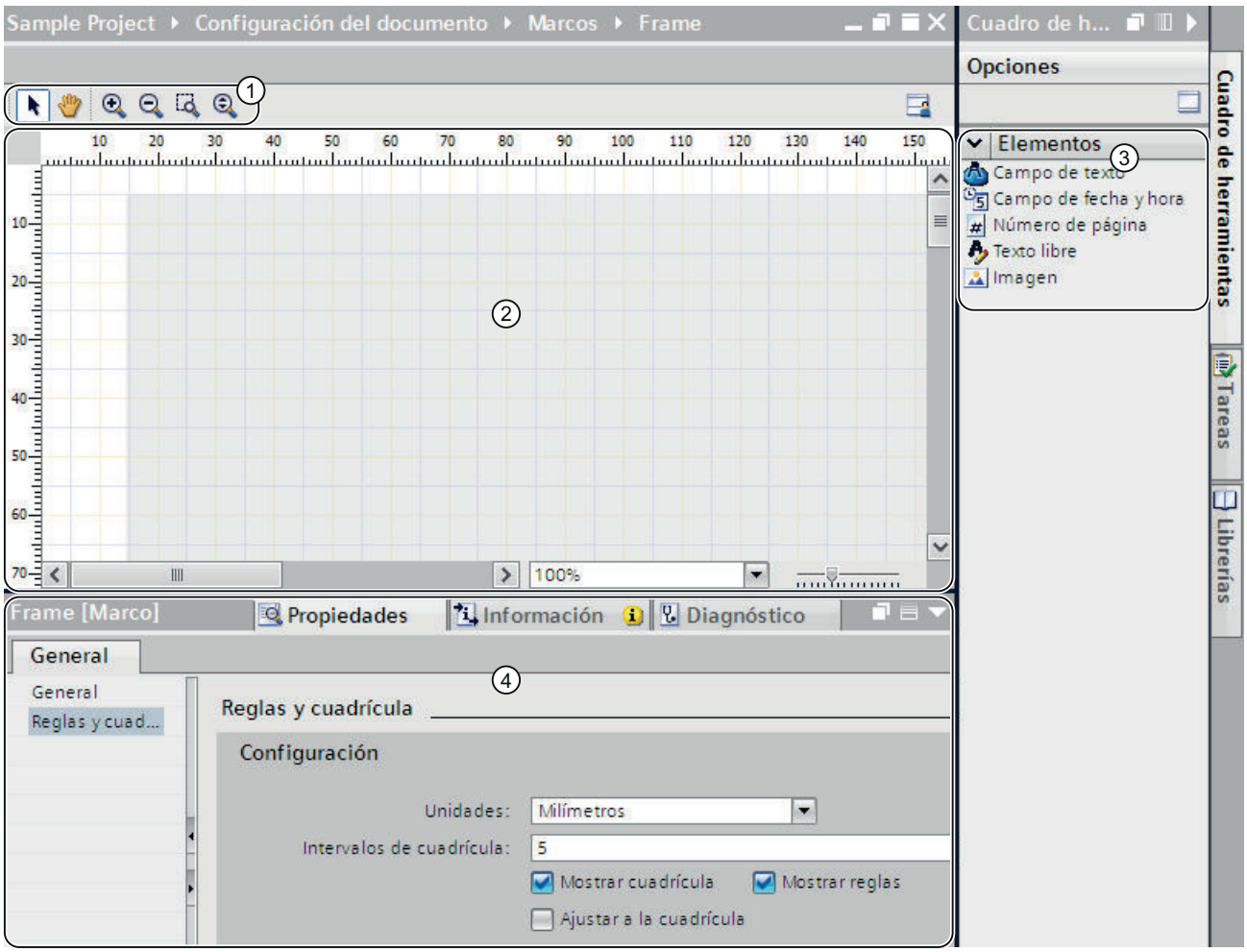
Consulte también

Crear portada (Página 397)
Crear marcos (Página 396)
Manejo general del editor de documentación (Página 399)

Manejo general del editor de documentación

Componentes del editor de documentación

La figura siguiente muestra una vista general de los componentes del editor de documentación:



① Barra de herramientas

En la barra de herramientas encontrará las herramientas siguientes (de izquierda a derecha):

- Herramienta de flecha
  Permite seleccionar un objeto.
- Herramienta de navegación
  Permite desplazar la sección de la página.
- Botón de acercar
  Amplía gradualmente la representación de la página.
- Botón de alejar
  Reduce gradualmente la representación de la página.
- Seleccionar factor de zoom
  Ajusta el tamaño de la página a un área de trabajo seleccionada.
- Zoom dinámico
  Ajusta el ancho de la página al área de trabajo.

② Área de trabajo

En el área de trabajo se puede diseñar la portada o el marco.

③ Task Card "Herramientas"

En la Task Card "Herramientas" están disponibles diferentes tipos de comodines que se pueden utilizar en la portada o en el marco. Los comodines se pueden posicionar en el área de trabajo mediante Drag & Drop.

④ Propiedades en la ventana de inspección

En la ficha "Propiedades" de la ventana de inspección se pueden visualizar y modificar las propiedades del objeto seleccionado actualmente. Por ejemplo se pueden modificar las propiedades de la página, formatear texto, definir exactamente la posición de los objetos en la página, etc.

Manejo del editor de documentación

El editor de documentación ofrece las funciones básicas siguientes:

- Función Drag & Drop
  El editor de documentación es un editor gráfico. Es decir, permite posicionar objetos libremente con el ratón. En el área de trabajo se visualiza una imagen de la página que representa exactamente el diseño posterior.
  Si desea seleccionar objetos en la página con el fin de desplazarlos o modificar las propiedades de los mismos, en la barra de herramientas debe estar activada la herramienta de flecha.
- Función de zoom
  Con la función de zoom se modifica el tamaño de representación de la página. Existen dos posibilidades para ajustar el tamaño de la página:
  - Con los botones de la barra de herramientas
    Active el símbolo de lupa "Ampliar factor de zoom" o "Reducir factor de zoom" en la barra de herramientas del editor de documentación. Luego haga clic en la página para ampliarla o reducirla gradualmente.
    Para ampliar un área determinada, seleccione la herramienta "Seleccionar factor de zoom" y, arrastrando el ratón, abra un marco alrededor del área que desea enfocar.
    Para ampliar o reducir el área de trabajo de modo continuo, utilice la función "Zoom dinámico". Para ampliar la representación de la página haga clic en un punto del área de trabajo y arrastre el ratón hacia arriba con la tecla del ratón pulsada. Para reducir la representación de la página, arrastre el ratón hacia abajo.
  - Con la barra de zoom
    Con la barra de zoom ubicada en el extremo inferior derecho del área de trabajo también se puede modificar el tamaño de representación. Introduzca un porcentaje o bien seleccione un porcentaje en la lista desplegable. También es posible modificar el tamaño de representación con el control deslizante.
- Navegación a través de la página
  Aparte de la función de desplazamiento existe la posibilidad de modificar la sección de la página con la herramienta de navegación. Para modificar la sección con la herramienta de navegación, active el símbolo de mano en la barra de herramientas. Luego haga clic con el ratón en la página y con la tecla del ratón pulsada desplace la página hasta la posición deseada.

Uso y ajuste de medios auxiliares de posicionamiento

Para posicionar elementos en la página se dispone de distintos medios auxiliares.

- Reglas
  En los bordes laterales del área de trabajo se encuentran unas reglas.
- Cuadrícula
  El fondo de página del área de trabajo está provisto de una cuadrícula.

Los medios auxiliares de posicionamiento se pueden visualizar/ocultar o ajustar en "Propiedades > Reglas y cuadrícula" de la ventana de inspección. Existen las posibilidades de ajuste siguientes:

- Unidades:
  Defina la unidad de medida para la cuadrícula y las reglas.
- Intervalos de cuadrícula:
  Defina la amplitud de la cuadrícula.
- Mostrar cuadrícula:
  Determine si la cuadrícula se debe visualizar u ocultar.
- Ajustar a la cuadrícula:
  Defina si los objetos deben ajustarse automáticamente a la cuadrícula. Si la opción está activada, las líneas de la cuadrícula adquieren una función "magnética".
- Mostrar reglas:
  Determine si deben mostrarse las reglas.

Consulte también

Editar portadas y marcos (Página 398)

Definir el área de impresión (Página 402)

Insertar comodines para metadatos (Página 403)

Definir el área de impresión

Dentro de un marco existe un área que está destinada para los contenidos de impresión relevantes. Los datos de proyecto se insertan luego en el marco y específicamente dentro del área definida, la cual no cambia. Es posible ajustar el tamaño del área de impresión.

Requisitos

Una marco debe estar abierto en el editor de documentación.

Procedimiento

Para definir un área para los contenidos de impresión, proceda del siguiente modo:

1. Haga clic en el área sombreada dentro de la representación de la página en el editor de documentación para seleccionar el área de los contenidos de impresión.
   Las propiedades del área de impresión se abren en la ventana de inspección.
2. Introduzca en la ventana de inspección la posición del área de impresión en el eje X y el eje Y.
3. Indique en la ventana de inspección el ancho y la altura en cm del área de impresión.

También es posible modificar el ancho y la posición del área de impresión en la representación gráfica de la página. Para ello, arrastre con el ratón los bordes del área de impresión hasta alcanzar el tamaño y la posición deseadas.

Consulte también

Crear marcos (Página 396)

Manejo general del editor de documentación (Página 399)

Insertar comodines para metadatos

Es posible insertar comodines en la portada y en un marco. Durante la impresión, los comodines se rellenan automáticamente con metadatos provenientes de la información del documento, siempre y cuando se trate de comodines para texto. También es posible añadir datos fijos, como por ejemplo, un texto libre o una imagen.

Todos los elementos están ordenados en niveles numerados. En caso de que se superpongan objetos, puede determinar en qué secuencia deben ser ordenados.

Tipos de comodines

Tipos de comodines disponibles:

- Campo de texto
  El campo de texto está disponible como comodín para un elemento de texto de una información de documento. En las propiedades del campo de texto puede ajustarse el texto de la información del documento que debe insertarse automáticamente en la impresión.
- Campo de fecha y hora
  En lugar del comodín, se añadirá una fecha y una hora durante la impresión. Puede corresponder, por ejemplo, a la fecha de creación o también a la hora de la última modificación del proyecto. Determine en las propiedades de la ventana de inspección de qué fecha o de qué hora se trata.
- Número de página
  Durante la impresión se insertará automáticamente el número de página correspondiente.
- Texto libre
  En las propiedades del campo de texto puede introducir texto de libre elección. El texto es estático y no se ve afectado por la información del documento seleccionada durante la impresión.
- Imagen
  Seleccione el archivo de imagen en las propiedades del comodín, en la ventana de inspección. Son posibles las imágenes en los formatos BMP, JPEG, PNG, EMF o GIF.

Requisitos

Una portada o un marco debe estar abierto en el editor de documentación.

Procedimiento

Para insertar comodines para metadatos en la portada o en un marco, proceda del siguiente modo:

1. Arrastre mediante Drag & Drop un campo desde la Task Card "Herramientas > Elementos" hasta el área de trabajo del editor de documentación.
   El comodín se inserta. En la ventana de inspección se muestran las propiedades del comodín y allí se pueden editar.
2. Seleccione el metadato que debe insertarse durante la impresión de la lista desplegable "Texto" en la ventana de inspección bajo "Propiedades > General > Campo de texto". O bien introduzca un texto libre alternativo o seleccione una imagen, en función del tipo de comodín del que se trate.
3. En "Propiedades > General > Posición y tamaño" de la ventana de inspección defina la posición del comodín en el eje X y en el eje Y e introduzca en cm el ancho y la altura del mismo. Determine la secuencia de los objetos en el campo "Nivel", en caso de que dichos objetos se superpongan. Cuanto más pequeño es el valor, más atrás se encuentra el objeto.
4. En "Propiedades > Vista" de la ventana de inspección seleccione el tipo de fuente y la orientación del texto, así como la alineación de la fuente. En caso de imágenes, no podrá realizar este ajuste.

Consulte también

Manejo general del editor de documentación (Página 399)

Mostrar la vista preliminar

Crear la vista preliminar

Crear la vista preliminar

Es posible crear una vista preliminar del impreso. Al igual que para la impresión real es posible seleccionar la información del documento. Esto permite ver ya en la vista preliminar el marco elegido y, dado el caso, la portada. Los ajustes se conservan para la impresión posterior.

Procedimiento

Para crear una vista preliminar y determinar el volumen de la posterior impresión, proceda del siguiente modo:

1. Elija el comando "Vista preliminar" del menú "Proyecto".
   Se abre el diálogo "Vista preliminar".
2. Seleccione el diseño del borde que desea utilizar para la impresión.
   - Seleccione la información del documento que se va a utilizar más adelante para la impresión en la lista desplegable "Información del documento".
   - Active la casilla de verificación "Imprimir portada" para imprimir la portada definida en la información del documento elegida.
   - Active la casilla de verificación "Imprimir tabla de contenido" para anteponer una tabla de contenido al impreso.

   Las casillas de verificación que activan la impresión de la portada y de la tabla de contenidos sólo se pueden seleccionar si se ha iniciado la impresión desde el árbol del proyecto.
3. Seleccione en "Imprimir objetos / área" lo que debe imprimirse. La selección sólo es posible si se ha iniciado la impresión en un editor que soporte esta función.
   - Elija "Todos" para imprimir todos los contenidos del editor.
   - Elija "Selección" para imprimir sólo los objetos seleccionados actualmente en el editor.
4. Seleccione el alcance de la impresión en "Propiedades".
   - Elija "Todos" para imprimir todos los datos de configuración de los objetos seleccionados.
   - Elija "Visible" para imprimir todas las informaciones visibles actualmente en pantalla en un editor. Esta opción sólo se puede elegir si se ha iniciado el trabajo de impresión en un editor que soporte esta función.
   - Elija "Compacto" si desea imprimir una versión corta de los datos de proyecto.
5. Haga clic en "Vista preliminar" para generar la vista preliminar.
   En el área de trabajo se crea una vista preliminar.

Nota

Tiempo de espera en documentos extensos

En proyectos especialmente extensos la elaboración de una vista preliminar puede durar varios minutos. En sistemas con capacidad suficiente se puede seguir trabajando simultáneamente de manera normal. En la barra de estado se muestra el progreso de la vista preliminar.

Consulte también

Manejo dentro de la vista preliminar (Página 406)

Manejo dentro de la vista preliminar

Funciones dentro de la vista preliminar

La vista preliminar muestra una imagen exacta del impreso posterior. Con los botones de la barra de herramientas se puede modificar la visualización de la vista preliminar. Están disponibles las funciones siguientes (de izquierda a derecha):

- Modo de navegación
  Permite desplazar la sección de la página.
  Para modificar la sección de página con la herramienta de navegación, active el símbolo de flecha. Luego haga clic con el ratón en la página y con la tecla del ratón pulsada desplace la página hasta la posición deseada.
- Función de zoom
  - "Acercar" y "Alejar"
    Amplía o reduce la representación de la página.
    Para ampliar o reducir gradualmente la representación, active el símbolo correspondiente. Luego haga clic en la página para ampliarla o reducirla gradualmente.
    Para ampliar un área determinada, active el símbolo "Seleccionar factor de zoom" y, arrastrando el ratón, abra un marco alrededor del área que desea enfocar.
    Active el símbolo "Acercar/Alejar dinámicamente" para ampliar la página dinámicamente. Luego, manteniendo pulsado el botón del ratón, arrastre el ratón hacia abajo por la página para reducir la representación. Por el contrario, arrástrelo hacia arriba para ampliar la representación.
  - Porcentaje en la lista desplegable
    Define el tamaño de representación de la página en porcentajes.
    Introduzca un porcentaje o bien seleccione un porcentaje en la lista desplegable. Como alternativa elija en la lista desplegable la opción "Ajustar a la página" para ajustar el tamaño de la página al área de trabajo. O bien elija "Ajustar al ancho" para ajustar el ancho de página al área de trabajo.
- "Adelante" y "Atrás"
  Toda modificación de la seccióndе página, del número de página o del tamaño de representación se guarda en un historial en segundo plano. Con el botón "Adelante" o "Atrás" se puede ir a la vista siguiente o regresar a la vista anterior, respectivamente.
- Navegar por las páginas
  - "Primera página"
    Salta a la primera página.
  - "Página anterior"
    Retrocede una página.
  - Campo de entrada "Número de página"
    Muestra la página actual. Para saltar directamente a una página, introduzca el número de página deseado.
  - "Página siguiente"
    Va a la siguiente página.
  - "Última página"
    Salta a la última página.

Consulte también

Crear la vista preliminar (Página 404)

Imprimir los datos de proyecto

Si desea imprimir datos de proyecto, existen dos posibilidades

- Impresión inmediata con ajustes estándar mediante el botón "Imprimir" de la barra de herramientas.
  El botón sólo se activa si se selecciona un objeto imprimible.
- Impresión con el comando de menú "Proyecto > Imprimir" con posibilidades de ajuste avanzadas.
  Es posible, por ejemplo, elegir una impresora diferente o determinados datos del documento o bien definir si se debe imprimir una portada y una tabla de contenido. Además se puede definir el alcance de la impresión o visualizar una vista preliminar antes de la impresión.

Requisitos

- Está configurada una impresora como mínimo.
- Los objetos a imprimir no están protegidos.
  Si hay objetos protegidos, el alcance de la impresión está restringido. Quite la protección de know how para imprimir los objetos por completo.

Imprimir los datos de proyecto

Para imprimir los datos del proyecto actual o bien el proyecto completo con posibilidades de ajuste avanzadas, proceda del siguiente modo:

1. Para imprimir todo el proyecto, selecciónelo completamente en el árbol del proyecto. Para imprimir elementos individuales pertenecientes a un proyecto, selecciónelos en el árbol del proyecto.
2. Elija el comando "Imprimir" del menú "Proyecto".
   Se abre el diálogo "Imprimir".
3. Seleccione la impresora en el campo "Nombre".
4. Haga clic en "Avanzado", para editar los ajustes de Windows de la impresora.

5. Seleccione el diseño del borde que desea utilizar para la impresión.
   - Seleccione la información del documento en la lista desplegable "Información del documento".
     El marco guardado en la información del documento se utiliza para la impresión. Todos los comodines dentro del marco seleccionado se rellenan con los metadatos de la información del documento seleccionada.
   - Active la casilla de verificación "Imprimir portada" para imprimir la portada guardada en la información del documento elegida.
   - Active la casilla de verificación "Imprimir tabla de contenido" para anteponer una tabla de contenido al impreso.

   Las casillas de verificación que activan la impresión de la portada y de la tabla de contenidos sólo se pueden seleccionar si se ha iniciado la impresión desde el árbol del proyecto.
6. Seleccione en "Imprimir objetos / área" lo que debe imprimirse. La selección sólo es posible si se ha iniciado la impresión en un editor que soporte esta función.
   - Elija "Todos" para imprimir todos los contenidos del editor.
   - Elija "Selección" para imprimir sólo los objetos seleccionados actualmente en el editor.
7. Seleccione el alcance de la impresión en "Propiedades".
   - Elija "Todos" para imprimir todos los datos de configuración de los objetos seleccionados.
   - Elija "Visible" para imprimir todas las informaciones visibles actualmente en pantalla en un editor. Esta opción sólo se puede elegir si se ha iniciado el trabajo de impresión en un editor.
   - Elija "Compacto" si desea imprimir una versión corta de los datos de proyecto.
8. Haga clic en "Vista preliminar" para generar una vista preliminar.
   En el área de trabajo se crea una vista preliminar.
9. Haga clic en "Imprimir" para iniciar la impresión.

Nota

Alcance del diálogo "Imprimir"

Las opciones que ofrece el cuadro de diálogo "Imprimir" varían en función de los elementos que se desean imprimir.

Resultado

Los datos de proyecto se preparan en segundo plano para la impresión y finalmente se imprimen en la impresora seleccionada. El progreso de la impresión se indica en la barra de estado. Mientras se prepara la impresión se puede seguir trabajando normalmente.

Los resultados de la impresión, así como posibles errores y advertencias aparecen listados una vez finalizada la impresión en "Información" de la ventana de inspección.

Cancelar el trabajo de impresión

Para cancelar un trabajo de impresión en curso, proceda del siguiente modo:

1. Haga clic en la cruz azul de la barra de estado, ubicada junto al indicador de progreso de la impresión.
   La impresión se cancelará en poco tiempo.

Consulte también

Protección de los datos de proyecto (Página 389)

Borrado de los derechos de acceso a dispositivos (Página 389)

Impresión de contenidos del proyecto (Página 391)

Personalizar portadas y marcos (Página 398)

7.5.4.2 Imprimir tiras rotulables para módulos

Función de impresión para tiras rotulables de módulos

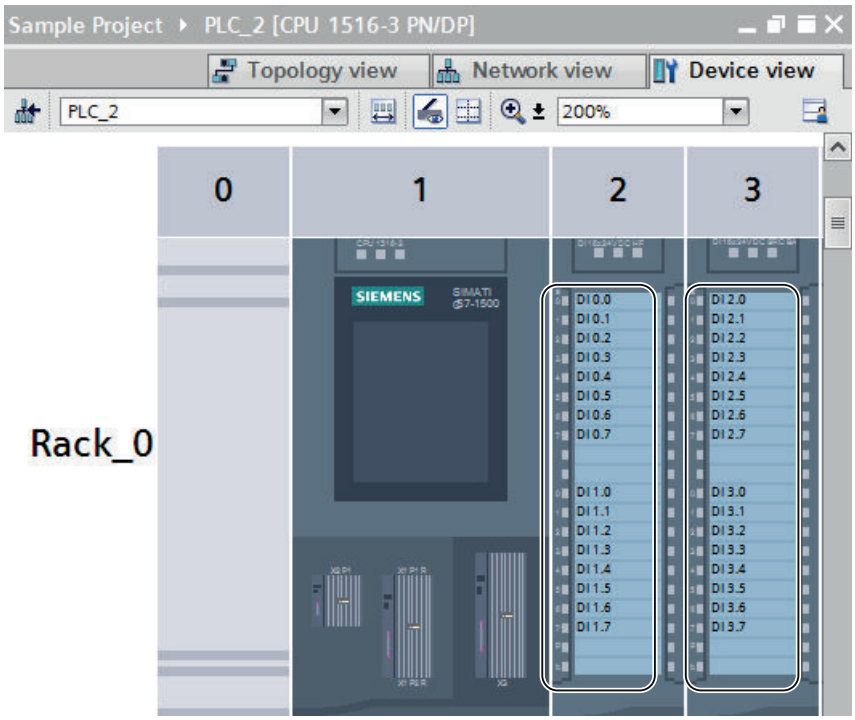
Impresión de tiras rotulables para módulos de hardware

El TIA Portal le ayuda a imprimir tiras rotulables para los módulos de su proyecto. Las tiras rotulables se adaptan perfectamente a los módulos empleados y permiten imprimir las informaciones siguientes:

- nombre simbólico de la entrada o la salida
- dirección absoluta de la entrada o la salida
- nombre simbólico más la dirección absoluta de la entrada o la salida. Puede ajustarse el orden que se desee.

En la vista de dispositivos se representan gráficamente los módulos. Si en la vista de dispositivos se ajusta el zoom a un mínimo de 200%, es posible ver los rótulos de cada uno de los módulos. El texto impreso en la tira rotulable se corresponde con la rotulación representada de la vista de dispositivos.

La figura siguiente muestra un ejemplo de dos módulos en la vista de dispositivos en los que puede verse la rotulación de las entradas y salidas:



Exportación y procesamiento posterior como archivo de Microsoft Word

Antes de imprimirse, es necesario exportar las tiras rotulables como archivo DOCX de Microsoft. Posteriormente las mismas pueden editarse con los programas de procesamiento de textos habituales, p. ej. con Microsoft Word 2010. Las tiras rotulables se representan en forma de tabla en el archivo DOCX. Es posible formatear a voluntad el texto o el fondo de cada una de las celdas.

Como ajuste predeterminado, la separación del texto dentro de la tabla se adapta para que los textos no se corten. Si no desea que el texto aparezca demasiado extendido o demasiado comprimido, modifique la separación del texto en las propiedades de la celda de la tabla.

Medios de impresión

Las tiras rotulables pueden imprimirse en pliegos prefabricados o en papel normal de formato DIN A4. En los pliegos prefabricados es posible desprender las tiras rotulables e insertarlas en las casillas de rotulación previstas en los módulos. Si realiza la impresión en papel normal, recorte las tiras rotulables. Pueden servirle de ayuda las marcas de corte que produce automáticamente la impresión.

Dado que los sistemas de alimentación de papel de las diversas impresoras presentan ligeras diferencias, en determinadas condiciones la impresión puede salir algo desplazada en el papel. Ahora bien, la impresión de las tiras rotulables sobre pliegos prefabricados debe tener una precisión milimétrica, puesto que en caso contrario el texto no coincidiría con el espacio disponible dentro del borde preperforado. Además, si la impresión no es precisa la rotulación de una entrada o salida no coincide plenamente con los indicadores de estado de canal del módulo. Por ello, en el TIA Portal es posible introducir un valor para el desplazamiento de la impresora con el fin de garantizar una impresión precisa. En el capítulo "Determinar el

desplazamiento del área de impresión (Página 412)" puede consultar cómo determinar el valor adecuado para el desplazamiento de la impresora.

Consulte también

Impresión de tiras rotulables (Página 411)

Determinar el desplazamiento del área de impresión (Página 412)

Función de documentación (Página 390)

Impresión de tiras rotulables

Es posible imprimir tiras rotulables para los módulos de un proyecto si está previsto que los módulos empleados estén rotulados. Primero hay que exportar las tiras rotulables a archivos DOCX de Microsoft Word. Para cada familia de módulos se crea un archivo DOCX propio (p. ej. para todos los módulos S7-1500 seleccionados). A continuación es posible diseñar a voluntad las tiras rotulables. La impresión se realiza finalmente desde el programa de tratamiento de textos.

Si no está prevista la impresión de tiras rotulables, el correspondiente comando de menú contextual estará desactivado. Si se han seleccionado varios módulos para la impresión de tiras rotulables y al menos uno de los módulos no soporta dicha impresión, en la ficha "Información > General" de la ventana de inspección se mostrará un aviso. En él se muestran todos los módulos que no son compatibles; la exportación del archivo de impresión para los módulos compatibles sigue su curso.

Requisitos

- Los módulos seleccionados deben soportar la impresión de tiras rotulables.
- Debe estar instalado un programa de tratamiento de textos compatible con archivos DOCX de Microsoft Word, p. ej. Microsoft Word 2010.
- Es necesario disponer de tiras rotulables prefabricadas para los módulos o papel normal de formato DIN A4.

Procedimiento

Para imprimir tiras rotulables para módulos de hardware, proceda del siguiente modo:

1. En el árbol del proyecto seleccione los módulos para los que desea imprimir tiras rotulables.
   - Se pueden seleccionar una o varias estaciones para imprimir tiras rotulables para todos los módulos insertados en ellas.
   - También es posible seleccionar los módulos deseados en la carpeta "Módulos locales" de las estaciones.
2. Haga clic con el botón derecho del ratón en uno de los dispositivos y elija el comando "Exportar tiras rotulables" del menú contextual.
   Se abrirá el cuadro de diálogo "Exportar tiras rotulables".

3. En el área "Contenido de la tira rotulable" seleccione los datos que deben imprimirse en la tira:
   - Elija "Nombre simbólico" para imprimir el nombre simbólico de la entrada o salida (corresponde al contenido de la columna "Nombre" de la tabla de variables E/S).
   - Elija "Dirección absoluta" para imprimir la dirección absoluta de la entrada o salida (corresponde al contenido de la columna "Dirección" de la tabla de variables E/S).
   - Elija "Dirección absoluta y simbólica" o "Dirección simbólica y absoluta" para imprimir ambas direcciones. La secuencia de impresión se corresponde con la secuencia señalada.
4. En el área "Clase de papel" seleccione el papel en el que desea realizar la impresión.
   - Elija "Impresión en pliego de rotulación SIEMENS" si desea realizar la impresión para los módulos en un pliego de rotulación prefabricado.
   - Elija "Impresión en papel normal" si desea imprimir en papel normal de formato DIN A4.
5. En el área "Desplazar área de impresión" elija los valores de corrección de su impresora para orientar correctamente el área de impresión. Este paso es necesario si se van a imprimir tiras rotulables prefabricadas.
   - En el campo "Desplazamiento vertical" introduzca un valor de corrección en milímetros. Un valor negativo desplaza el área de impresión hacia arriba. Un valor positivo desplaza el área de impresión hacia abajo.
   - En el campo "Desplazamiento horizontal" introduzca un valor de corrección en milímetros. Un valor negativo desplaza el área de impresión hacia la izquierda. Un valor positivo desplaza el área de impresión hacia la derecha.
6. En el campo "Ruta" elija una ruta en la que deban guardarse los archivos DOCX exportados.
7. Haga clic en el botón "Exportar" para iniciar la exportación a un archivo DOCX.
   Se crearán los archivos DOCX.
8. Abra los archivos DOCX con un programa habitual de tratamiento de textos, p. ej., con Microsoft Word 2010, y modifique si es necesario el diseño de las tiras rotulables.
9. Imprima las tiras rotulables desde el programa de tratamiento de textos. Para ello, utilice el papel que ha indicado en el cuadro de diálogo de exportación.
10. Si ha utilizado papel prefabricado, separe las tiras rotulables por los puntos preperforados previstos. Si ha utilizado papel normal DIN A4, recorte las tiras rotulables.

Consulte también

Determinar el desplazamiento del área de impresión (Página 412)

Determinar el desplazamiento del área de impresión

Si utiliza un pliego de rotulación prefabricado, la impresión debe ser muy precisa para que el texto coincida exactamente con las tiras rotulables preperforadas y concuerde después con los indicadores de estados de canal del módulo. Sin embargo, los sistemas de alimentación de papel varían ligeramente de una impresora a otra. Por ello, en el TIA Portal puede ser necesario introducir un valor de corrección adecuado para la impresora utilizada. En el archivo

DOCX exportado el área de impresión se desplaza para que la impresión se adecue a los pliegos de rotulación prefabricados.

Los ajustes para desplazar el área de impresión se guardan para el usuario de Windows en cuestión. Si se inicia sesión en Windows utilizando otro nombre de usuario será necesario volver a introducir los valores de corrección.

A continuación se explica cómo determinar el valor de corrección para la impresora.

Requisitos

- Es necesario disponer de un pliego de rotulación prefabricado.
- Debe disponer de la misma impresora que se utilizará después para la impresión. La impresora debe estar preparada para la impresión sobre papel normal de formato DIN A4.

Procedimiento

Para determinar el valor de corrección para la impresora, proceda del siguiente modo:

1. Imprima un pliego de rotulación en papel normal de formato DIN A4, tal como se describe en el capítulo "Impresión de tiras rotulables (Página 411)".
2. Compare la impresión sobre papel DIN A4 con el pliego de rotulación prefabricado.
3. Si el área de impresión está desplazada, debe emplear valores de corrección.
   - Mida con una regla el desplazamiento horizontal respecto al pliego de rotulación prefabricado. Dicho desplazamiento se introducirá posteriormente en el campo "Desplazamiento horizontal" del cuadro de diálogo de exportación para la impresión. Si el área de impresión se ha desplazado hacia la derecha, hay que introducir un valor de corrección negativo. Si el área de impresión se ha desplazado hacia la izquierda, hay que introducir un valor de corrección positivo.
   - Mida con una regla el desplazamiento vertical respecto al pliego de rotulación prefabricado. Dicho desplazamiento se introducirá posteriormente en el campo "Desplazamiento vertical" del cuadro de diálogo de exportación para la impresión. Si el área de impresión se ha desplazado hacia abajo, hay que introducir un valor de corrección negativo. Si el área de impresión se ha desplazado hacia arriba, hay que introducir un valor de corrección positivo.

7.6 Deshacer y rehacer acciones

7.6.1 Principios básicos para deshacer y rehacer acciones

Función

En todo momento es posible deshacer las acciones realizadas. Para ello, cada acción que se realiza se guarda en una pila de acciones. Al deshacer las acciones la pila se procesa de

arriba a abajo. Es decir, al deshacer una acción que está en la parte inferior de la pila, también se deshacen automáticamente todas las acciones que están por encima de la misma.

Las acciones deshechas se pueden rehacer tantas veces como se requiera hasta que se realice una nueva acción. Si se ejecuta una nueva acción, ya no será posible rehacer acciones deshechas.

Particularidades al deshacer acciones

Hay ciertas acciones que vacían la pila de acciones. Estas acciones e incluso las acciones realizadas antes de las mismas no se pueden deshacer. Las siguientes acciones vacían la pila de acciones:

- Guardar
- Administración de proyectos (crear un proyecto, así como abrir, cerrar y borrar un proyecto)
- Compilar
- Restablecer bloques
- Establecer conexión online
- Cargar
- Escribir en Memory Cards

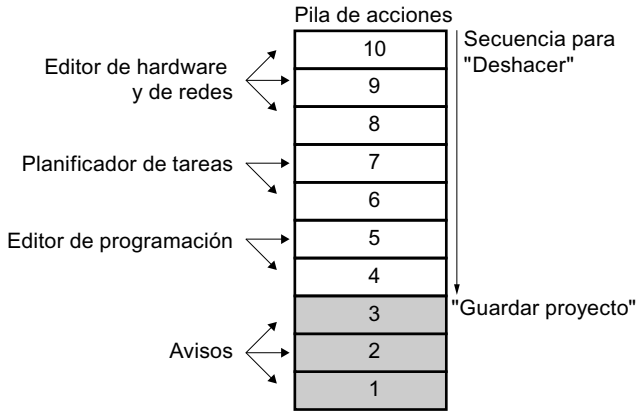
Visualización de la pila de acciones

Tan pronto como se realiza una acción que pueda deshacerse, se activa el botón "Deshacer" en la barra de herramientas. Este botón está dividido y con la flecha hacia abajo se puede abrir una lista desplegable. En esta lista desplegable aparecen todas las acciones de la pila de acciones que se pueden deshacer. En caso de haber ejecutado acciones en otro editor diferente al visualizado actualmente, también se visualizará el editor correspondiente como subtítulo. Esto permite reconocer siempre en qué lugar tiene efecto la función "Deshacer". Los subtítulos se eliminarán de la lista si no se pueden deshacer más acciones para el editor correspondiente.

Las acciones deshechas se insertan en la pila de acciones para ser repetidas. Aquí se pueden repetir las acciones deshechas. La visualización de las acciones que se pueden repetir es análoga a la visualización de las acciones que se pueden deshacer.

Ejemplo de la función para deshacer acciones

La figura siguiente muestra cómo se deshacen acciones en distintos editores y tablas:



En este ejemplo no es posible deshacer las acciones 1 a 3 debido a que se ha guardado el proyecto. Las acciones 4 a 10 se pueden deshacer en la dirección que indica la flecha. Esto significa que primero debe deshacerse la acción 10. Después de deshacer la acción 8 no es posible deshacer la acción 5. Primero deben deshacerse las acciones 7 y 6. La última acción que se puede deshacer es la acción cuatro. También es posible deshacer varias acciones en un solo paso, para ello se debe deshacer una acción que se encuentre en la parte inferior de la pila. Todas las acciones que están por encima de ésta se deshacen automáticamente.

El mismo principio rige a la hora de rehacer acciones.

Consulte también

Deshacer una acción (Página 415)

Rehacer una acción (Página 416)

7.6.2 Deshacer una acción

Para deshacer acciones se dispone de las siguientes posibilidades:

- Deshacer sólo la última acción
  Se deshace únicamente la última acción realizada.
- Deshacer tantas acciones como se requiera
  En un solo paso se deshacen varias acciones de la pila de acciones.

Deshacer sólo la última acción

Para deshacer la última acción realizada, proceda del siguiente modo:

1. Haga clic en el botón "Deshacer" de la barra de herramientas.
   - Si la acción no ha sido realizada en el editor visualizado actualmente, aparece una consulta de seguridad.
   - Si para deshacer la acción en cuestión es necesario abrir un editor que contiene un objeto protegido, será necesario introducir la contraseña correspondiente al objeto.
2. Dado el caso, confirme la consulta de seguridad con "Sí".
3. Dado el caso, introduzca la contraseña.
   Se visualizará el editor en que se realizó la acción y a continuación se deshace la misma.

Deshacer tantas acciones como se requiera

Para deshacer en un solo paso varias acciones de la pila de acciones, proceda del siguiente modo:

1. Haga clic en la flecha hacia abajo que se encuentra junto al botón "Deshacer" de la barra de herramientas.
   Se abrirá una lista desplegable que muestra todas las acciones que se pueden deshacer. Las acciones de otros editores se marcan con los nombres de los editores como título intermedio.
2. Haga clic en la acción que desea deshacer.
   La acción seleccionada y todas las acciones que se encuentran por encima de la misma se deshacen. Si para deshacer la acción en cuestión es necesario abrir un editor que contiene un objeto protegido, será necesario introducir la contraseña correspondiente al objeto.
3. Dado el caso, introduzca las contraseñas requeridas.
   Se visualizarán los editores en que se realizaron las acciones y a continuación se deshacen las mismas.

Consulte también

Principios básicos para deshacer y rehacer acciones (Página 413)

Rehacer una acción (Página 416)

7.6.3 Rehacer una acción

Existe la posibilidad de rehacer una acción deshecha, de manera que se obtenga el estado previo a la acción "Deshacer". Observe que esto sólo es posible antes de realizar una nueva acción. Para rehacer acciones se dispone de las siguientes posibilidades:

- Rehacer sólo la última acción deshecha
  Se rehace únicamente la última acción deshecha.
- Rehacer tantas acciones deshechas como se requiera
  En un solo paso se rehacen varias acciones deshechas de la pila de acciones.

Rehacer sólo la última acción deshecha

Para rehacer la última acción deshecha, proceda del siguiente modo:

1. Haga clic en el botón "Rehacer" de la barra de herramientas.
   - Si la acción no se rehace en el editor visualizado actualmente, aparece una consulta de seguridad.
   - Si para rehacer la acción en cuestión es necesario abrir un editor que contiene un objeto protegido, será necesario introducir la contraseña correspondiente al objeto.
2. Dado el caso, confirme la consulta de seguridad con "Sí".
3. Dado el caso, introduzca la contraseña.
   Se visualizará el editor en que se deshizo la acción y a continuación se rehace la misma.

Rehacer tantas acciones deshechas como se requiera

Para rehacer en un solo paso varias acciones deshechas de la pila de acciones, proceda del siguiente modo:

1. Haga clic en la flecha hacia abajo que se encuentra junto al botón "Rehacer" de la barra de herramientas.
   Se abrirá una lista desplegable que muestra todas las acciones deshechas. Las acciones de otros editores se marcan con los nombres de los editores como título intermedio.
2. Haga clic en la acción que desea rehacer.
   La acción seleccionada y todas las acciones que se encuentran por encima de la misma se rehacen. Si para rehacer la acción en cuestión es necesario abrir un editor que contiene un objeto protegido, será necesario introducir la contraseña correspondiente al objeto.
3. Dado el caso, introduzca las contraseñas requeridas.
   Se visualizarán los editores en que se deshicieron la acciones y a continuación se rehacen las mismas.

Consulte también

Principios básicos para deshacer y rehacer acciones (Página 413)

Deshacer una acción (Página 415)

7.7 Buscar y reemplazar en proyectos

7.7.1 Información acerca de la función de búsqueda

Buscar y reemplazar

Los editores ofrecen la posibilidad de buscar textos. La función de búsqueda permite encontrar en el editor abierto actualmente todos los textos que contienen el término buscado. Los resultados se seleccionan consecutivamente en el editor abierto.

Además, se dispone de las siguientes posibilidades:

- Optimización de la búsqueda mediante opciones adicionales
- Sustitución de textos encontrados

Las opciones adicionales disponibles y el tipo de textos que pueden buscarse dependen de los productos instalados y del editor abierto.

Consulte también

Buscar y reemplazar (Página 418)

7.7.2 Buscar y reemplazar

Uso de la función de búsqueda

La función "Buscar y reemplazar" permite buscar o sustituir textos en un editor.

Opciones adicionales para la búsqueda

La búsqueda se puede optimizar eligiendo entre las siguientes opciones adicionales:

- Buscar sólo palabra completa
  Se buscan sólo palabras completas, ignorándose las que contengan el término buscado como parte de la palabra.
- Mayúsculas/minúsculas
  Las mayúsculas y minúsculas se tienen en cuenta en la búsqueda.
- Buscar en estructuras subordinadas
  Se buscan también los textos contenidos en otro objeto.
- Buscar en textos ocultos
  En la búsqueda también se tienen en cuenta los textos asignados a un texto diferente, pero que están ocultos en ese momento.
- Utilizar comodines
  Introduzca un asterisco como comodín para un número cualquiera de caracteres. Ejemplo: Desea buscar todas las palabras que contengan la secuencia de caracteres "disp". Por tanto, introduzca "disp*" en el campo de búsqueda.
  Si desea omitir sólo un carácter, introduzca un signo de interrogación como comodín.
- Utilizar expresiones regulares (sólo para la búsqueda en scripts)
  Una expresión regular es una cadena de caracteres que sirve para describir cantidades y que se utiliza como filtro. De esta manera es posible realizar búsquedas complejas.

Las opciones adicionales disponibles dependen de los productos instalados y del editor abierto.

Iniciar la búsqueda

Para iniciar la función "Buscar y reemplazar", proceda del siguiente modo:

1. Elija el comando "Buscar y reemplazar" del menú "Edición" o abra la paleta "Buscar y reemplazar" de la Task Card "Tareas".
   Se abrirá la paleta "Buscar y reemplazar".
2. Introduzca un término en la lista desplegable "Buscar".
   Como alternativa, seleccione uno de los últimos términos buscados en la lista desplegable.
3. Active las opciones adicionales deseadas para la búsqueda.
4. Seleccione el punto de partida y el sentido de la búsqueda mediante los botones de opción.
   - Seleccione "Todo el documento" si desea examinar todo el editor, independientemente de la selección actual.
   - Seleccione "Desde la posición actual" si desea iniciar la búsqueda desde la selección actual.
   - Seleccione "Selección" si sólo desea buscar dentro de la selección actual.
   - Seleccione "Hacia abajo" para examinar el editor de arriba abajo o de izquierda a derecha.
   - Seleccione "Hacia arriba" para examinar el editor de abajo arriba o de derecha a izquierda.
5. Haga clic en "Buscar".
   El primer resultado se destaca en el editor.
6. Para ver el siguiente resultado, haga clic nuevamente en "Buscar".
   El siguiente resultado se destaca en el editor. Dado el caso, repita este paso hasta alcanzar el último resultado.

Reemplazar el término de búsqueda

Es posible reemplazar términos de búsqueda individuales, o bien sustituir automáticamente todos los resultados si el editor en cuestión soporta esta función. Para reemplazar un término de búsqueda, proceda del siguiente modo:

1. Introduzca un término en la lista desplegable "Buscar".
   Como alternativa, seleccione uno de los últimos términos buscados en la lista desplegable.
2. Active las opciones adicionales deseadas para la búsqueda.
3. Haga clic en el botón "Buscar" para iniciar la búsqueda del término introducido.
   El primer resultado se muestra en el editor.
4. Introduzca en la lista desplegable "Reemplazar" el texto por el que desea reemplazar el término buscado.
   Como alternativa, seleccione uno de los últimos textos introducidos en la lista desplegable.

5. Para reemplazar el resultado destacado por el texto introducido, haga clic en el botón "Reemplazar".
   El texto encontrado se sustituye y el siguiente resultado se destaca en el editor.
   Repita este paso hasta haber reemplazado todos los resultados deseados. Para saltar al siguiente resultado sin sustituir la palabra destacada, haga clic en el botón "Buscar" en vez de "Reemplazar".
6. Para sustituir automáticamente todos los resultados, haga clic en el botón "Reemplazar todos".

Consulte también

Información acerca de la función de búsqueda (Página 417)

7.8 Trabajar con listas de textos

7.8.1 Listas de textos

Introducción

Los textos a los que se hace referencia en los avisos pueden administrarse de forma centralizada. Todos los textos están almacenados en listas de textos. Cada lista de textos tiene un nombre unívoco con el que se accede a su contenido. Todo texto de una lista de textos tiene asignado un rango de valores. Cuando aparece un valor de un rango determinado, se accede al texto correspondiente.

Todos los textos pueden traducirse a todos los idiomas del proyecto. Para ello se dispone de dos posibilidades:

- Los textos traducidos se pueden introducir en una lista. La lista se encuentra en "Idiomas y recursos > Textos del proyecto" del árbol del proyecto.
- Todos los textos se pueden exportar a un archivo en formato Office Open XML para introducir luego la traducción en un programa de hojas de cálculo. Las traducciones se pueden volver a importar posteriormente.

La traducción de los textos a los demás idiomas del proyecto se llevan a cabo en el marco de los textos del proyecto. En el editor de listas de textos únicamente se asignan los distintos textos a una lista de textos determinada.

Todo dispositivo del proyecto puede tener listas de textos propias. Por tanto, en el árbol del proyecto están dispuestas debajo de los dispositivos. Adicionalmente hay listas de textos que rigen para todos los dispositivos. Éstas se encuentran en el árbol del proyecto en "Datos comunes > Listas de textos".

Listas de textos de usuario y listas de textos de sistema

Existen dos tipos de listas de textos:

- Listas de textos de usuario
  El usuario puede crear listas de textos personalizadas y rellenarlas con textos, es decir, definir por su propia cuenta los rangos de valores y los textos correspondientes. En las listas de textos personalizadas, el nombre de la lista de textos comienza de forma estándar con "USER". Sin embargo, es posible modificar este nombre como se desee.
- Listas de textos de sistema
  Las listas de textos de sistema son predeterminadas por el sistema. Todas estas listas contienen textos relacionados con los dispositivos. Se crean automáticamente tan pronto se inserte un dispositivo en el proyecto. En los avisos de sistema, el nombre de la lista de textos comienza con "SYSTEM". El nombre de la lista de textos y los rangos de valores que contiene no se pueden modificar. Sólo es posible editar los textos que están asignados a los distintos rangos de valores.

Listas de textos de usuario	Listas de textos de sistema
Una lista de textos de usuario se puede asignar sólo a un dispositivo.	Las listas de textos de sistema pueden asignarse tanto a un dispositivo como al proyecto entero.
Es posible crear listas de textos adicionales y borrar las listas de textos existentes.	No es posible crear listas de textos adicionales ni tampoco borrar las listas de textos existentes.
Es posible agregar y borrar rangos de valores en las listas de textos.	No es posible agregar ni borrar rangos de valores en las listas de textos.
Es posible determinar tanto los rangos de valores como los textos correspondientes.	Sólo es posible editar el texto correspondiente a un rango de valores.

Listas de textos específicas del dispositivo y listas de textos generales

Las listas de textos específicas del dispositivo afectan sólo a un dispositivo del proyecto, siendo válidas sólo para éste. Por tanto, en el árbol del proyecto están ordenadas debajo de un dispositivo. Las listas de textos específicas del dispositivo pueden ser listas definidas por el usuario o predeterminadas por el sistema.

Si las listas de textos de sistema valen para varios dispositivos o no pueden asignarse unívocamente a un dispositivo en particular, se guardan en el árbol del proyecto bajo "Datos comunes". Estas listas de textos están disponibles para todos los dispositivos. Las listas de textos generales siempre están predeterminadas por el sistema y sirven tan sólo para los avisos de diagnóstico del sistema. Por tanto, las listas de textos de usuario no se pueden almacenar en los "Datos comunes".

Consulte también

Importar y exportar textos del proyecto (Página 354)

7.8.2 Crear listas de textos de usuario

Crear listas de textos

Es posible crear listas de textos de usuario para los distintos dispositivos.

Requisitos

- La vista del proyecto está abierta.
- Hay un proyecto abierto.
- El proyecto contiene como mínimo un dispositivo.

Procedimiento

Para crear una lista de textos de usuario, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en la flecha situada a la izquierda de un dispositivo.
   Se visualizan los elementos subordinados del dispositivo.
2. Haga doble clic en "Listas de textos".
   Todas las listas de textos asignadas al dispositivo aparecen en una tabla en el área de trabajo.
3. Haga doble clic en la primera fila vacía de la tabla.
   Se crea una nueva lista de textos de usuario.
4. En la columna "Nombre", introduzca un nombre para la nueva lista de textos.
5. En la lista desplegable de la columna "Selección", indique si los rangos de valores deben visualizarse de forma decimal, binaria o en bits. Dependiendo del dispositivo también es posible realizar otras opciones.
6. Introduzca un comentario en la columna "Observación".
   Se ha creado una nueva lista de textos de usuario. A continuación es posible definir los rangos de valores e introducir textos.

7.8.3 Editar listas de textos de usuario

Editar listas de textos de usuario

En las listas de textos de usuario es posible definir rangos de valores e introducir los textos correspondientes. En el árbol del proyecto, las listas de textos de usuario se encuentran siempre debajo de un dispositivo.

Requisitos

- La vista del proyecto está abierta.
- Hay un proyecto abierto.
- El proyecto contiene como mínimo un dispositivo.

Complementar listas de textos de usuario con rangos de valores y textos

Para complementar listas de textos de usuario con rangos de valores y textos, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en la flecha situada a la izquierda de un dispositivo.
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Listas de textos".
   Todas las listas de textos asignadas al dispositivo aparecen en una tabla en el área de trabajo.
3. Seleccione una lista de textos en la tabla.
   En el área de trabajo se muestran los contenidos de la lista de textos seleccionada. Allí se pueden introducir rangos de valores y asignar textos individualmente a los distintos rangos.
4. Introduzca los rangos de valores deseados en las columnas "Rango de" y "Rango a". La entrada debe corresponderse con el formato numérico seleccionado para la tabla.
5. Introduzca para cada rango de valores un texto en la columna "Entrada".

7.8.4 Editar listas de textos de sistema

Editar listas de textos de sistema

En las listas de textos de sistema sólo es posible modificar los distintos textos asignados a un rango de valores.

En el árbol del proyecto, las listas de textos de sistema se encuentran bien sea debajo de un dispositivo, o bien bajo "Datos comunes".

Requisitos

- La vista del proyecto está abierta.
- Hay un proyecto abierto.
- El proyecto contiene como mínimo un dispositivo.

Modificar textos en listas de textos de sistema

Para editar textos asignados a un rango de valores en las listas de textos del sistema, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en la flecha situada a la izquierda de un dispositivo o en el elemento "Datos comunes".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Listas de textos".
   Todas las listas de textos asignadas al dispositivo o utilizadas en común aparecen en una tabla en el área de trabajo.
3. Seleccione una lista de textos en la tabla.
   En el área de trabajo se muestran los contenidos de las listas de textos seleccionadas. Allí puede complementar o editar los textos asignados a un rango de valores.
4. Introduzca para cada rango de valores un texto en la columna "Entrada".

7.9 Utilizar Memory Cards

7.9.1 Principios básicos de las Memory Cards

Introducción

Las Memory Cards son tarjetas de memoria insertables que pueden utilizarse para distintos fines y que están disponibles en distintas versiones. Dependiendo del tipo o bien de la familia de dispositivos, las Memory Cards se pueden utilizar, p. ej., para los fines siguientes:

- Como memoria de carga de una CPU
- Medio de almacenamiento para proyectos, backups de firmware y para cualquier otro archivo
- Medio de almacenamiento para realizar una actualización de firmware
- Medio de almacenamiento para el nombre de dispositivo PROFINET

En la documentación del dispositivo en cuestión encontrará información sobre las características técnicas y el manejo general de la Memory Card respectiva. La palabra clave "Memory Card" de la Ayuda en pantalla le proporcionará información sobre el uso de Memory Cards en el TIA Portal.

ATENCIÓN
No utilice Memory Cards para fines ajenos a SIMATIC y no formatee las mismas con dispositivos de otros fabricantes ni utilizando medios de Windows. Esto sobrescribiría la estructura interna de la Memory Card, la cual no se puede restablecer. De lo contrario, la Memory Card no se podrá volver a utilizar en dispositivos SIMATIC.

Consulte también

Agregar un lector de tarjetas personalizado (Página 425)
Acceder a Memory Cards (Página 425)
Mostrar las propiedades de Memory Cards (Página 426)

7.9.2 Agregar un lector de tarjetas personalizado

Introducción

Si el lector de tarjetas no se ha detectado automáticamente, puede agregarse manualmente.

Requisitos

La vista del proyecto está abierta.

Procedimiento

Para agregar un lector de tarjetas, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Elija el comando "Card Reader/Memoria USB > Agregar lector de tarjetas personalizado" del menú "Proyecto".
   Se abre el cuadro de diálogo "Agregar lector de tarjetas personalizado".
3. Seleccione la ruta del lector de tarjetas de la lista desplegable.
4. Confirme la entrada con "Aceptar".

Consulte también

Principios básicos de las Memory Cards (Página 424)
Acceder a Memory Cards (Página 425)
Mostrar las propiedades de Memory Cards (Página 426)

7.9.3 Acceder a Memory Cards

Requisitos

- Hay una Memory Card insertada en el lector de tarjetas.
- La vista del proyecto está abierta.

Procedimiento

Para acceder a Memory Cards, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Elija el comando "Card Reader/Memoria USB > Mostrar Card Reader/memoria USB" del menú "Proyecto".
   La carpeta "Card Reader/Memoria USB" aparece en el árbol del proyecto.
3. Abra la carpeta "Card Reader/Memoria USB".
   A continuación puede acceder a la Memory Card.

Nota

Si en la Memory Card hay guardados datos de un producto no instalado, las carpetas que contienen dichos datos se muestran en gris. Cuando se accede a una de estas carpetas se recibe un mensaje de error. Instale el producto correspondiente en caso necesario.

Consulte también

Principios básicos de las Memory Cards (Página 424)

Agregar un lector de tarjetas personalizado (Página 425)

Mostrar las propiedades de Memory Cards (Página 426)

7.9.4 Mostrar las propiedades de Memory Cards

Las propiedades de las Memory Cards utilizadas se pueden visualizar. Observe que dependiendo del dispositivo en cuestión deben utilizarse diferentes Memory Cards que tienen a su vez distintas propiedades.

Requisitos

- Hay una Memory Card insertada en el lector de tarjetas.
- La vista del proyecto está abierta.

Procedimiento

Para visualizar las propiedades de una Memory Card, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la Memory Card cuyas propiedades desea visualizar.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el diálogo "Memory Card <nombre de la Memory Card>". En él que se visualizan las propiedades.

Consulte también

Principios básicos de las Memory Cards (Página 424)
Agregar un lector de tarjetas personalizado (Página 425)
Acceder a Memory Cards (Página 425)

7.10 Utilizar librerías

7.10.1 Principios básicos de las librerías

Introducción

En las librerías se depositan objetos para su posterior reutilización. En cada proyecto existe una librería enlazada con este. Además de la libraría del proyecto es posible crear un número cualquiera de librerías globales que pueden utilizarse en varios proyectos. Las librerías son compatibles entre sí haciendo posible copiar y mover elementos entre las librerías. Por ejemplo, mediante las librerías se crean plantillas para bloques que, a continuación, se insertan en la librería del proyecto y luego se perfeccionan. Finalmente se copian los bloques de la librería del proyecto en una librería global. Las librerías globales están a disposición de las demás personas que trabajan en el proyecto. Dichas personas utilizan los bloques y, en caso necesario, los adaptan a sus exigencias personales.

Tanto la librería del proyecto como las librerías globales distinguen entre dos tipos de objetos diferentes:

- Plantillas maestras
  Casi todos los objetos pueden guardarse como plantillas maestras e insertarse después en el proyecto. Por ejemplo, para la documentación de la instalación es posible guardar como plantillas maestras dispositivos completos con sus contenidos o sus portadas.
- Tipos
  Son tipos adecuados los elementos necesarios para la ejecución de programas de usuario, por ejemplo, bloques, tipos de datos PLC, tipos de datos de usuario HMI o faceplates. Es posible crear versiones diferentes de los tipos, de modo que contribuyen a un perfeccionamiento profesional. Los proyectos en los que se utilizan tipos se actualizan tan pronto haya nuevas versiones de los tipos.

Librería del proyecto

Todo proyecto tiene una librería propia, la librería del proyecto. En ella se depositan los objetos que se desean reutilizar en el proyecto. La librería del proyecto se abre, guarda y cierra siempre junto con el proyecto actual.

Librerías globales

Además de la librería del proyecto se utilizan librerías globales que pueden utilizarse en varios proyectos. Existen tres variantes de librerías globales:

- Librerías del sistema
  Siemens suministra librerías globales para sus productos de software. Estas contienen funciones y bloques de función preprogramados que pueden utilizarse en el proyecto. No es posible modificar las librerías suministradas. Las librerías suministradas se cargan automáticamente de acuerdo con el proyecto. Si se trabaja en V12 SP1 en modo de compatibilidad se cargan las librerías adecuadas para la versión de producto V12 SP1. Para todos los demás proyectos se cargan las librerías suministradas para la última versión del TIA Portal.
- Librerías de empresa
  Las librerías de empresa son puestas a disposición de forma centralizada por su organización, p. ej. en una carpeta centralizada de una unidad de red. El TIA Portal administra las librerías de empresa de forma automática. En cuanto hay disponible una versión más reciente de una librería de empresa existente, se pide al usuario que actualice la librería de empresa correspondiente a la última versión.
- Librerías de usuario
  Las librerías de usuario globales no dependen de un proyecto determinado y es posible transmitirlas a otros usuarios. Un acceso conjunto a librerías de usuario globales también es posible, por ejemplo, en una unidad de red, si todos los usuarios abren la librería de usuario global en modo de solo lectura.
  Las librerías de usuario globales creadas por el usuario en versiones anteriores del TIA Portal pueden seguir utilizándose. Para seguir utilizando librerías de usuario globales de versiones anteriores del TIA Portal es necesario actualizarlas antes.

Comparación de objetos de librería

Existe la posibilidad de comparar los bloques y tipos de datos PLC con los objetos de un dispositivo. De esta forma es posible saber, por ejemplo, si determinados bloques o tipos de datos PLC se han utilizado en un proyecto y si se han modificado.

Consulte también

Sinopsis de la Task Card "Librerías". (Página 429)

Sinopsis de la vista de librerías (Página 432)

Sinopsis de la administración de librerías (Página 435)

Principios básicos de las plantillas maestras (Página 455)

Principios básicos de los tipos (Página 458)

7.10.2 Uso de la Task Card "Librerías"

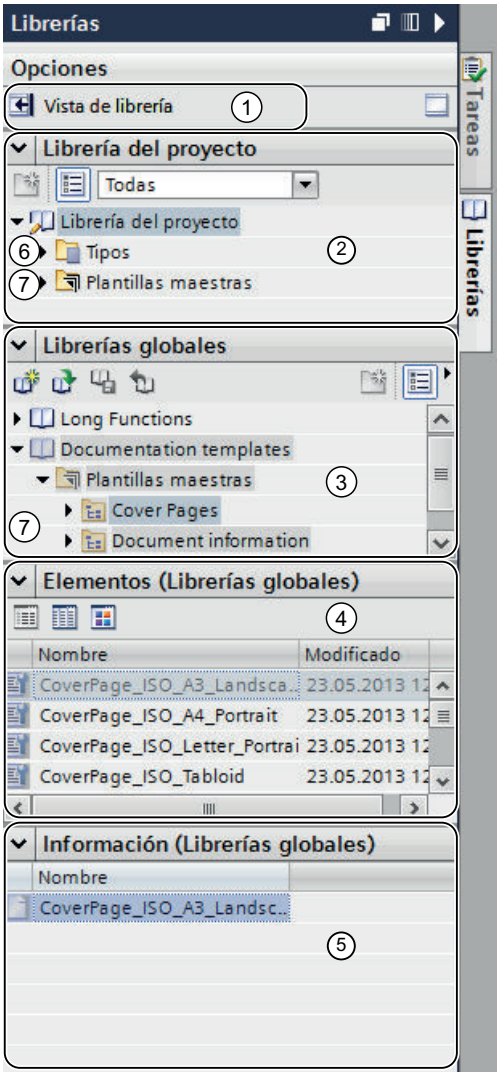
7.10.2.1 Sinopsis de la Task Card "Librerías".

Función de la Task Card "Librerías"

La Task Card "Librerías" permite utilizar eficientemente la librería del proyecto y las librerías globales.

Estructura de la Task Card "Librerías"

La Task Card "Librerías" consta de los siguientes componentes:



① Botón "Vista de librería"
② Paleta "Librería del proyecto"
③ Paleta "Librerías globales"
④ Paleta "Elementos"
⑤ Paleta "Información"
⑥ Carpeta "Tipos"
⑦ Carpeta "Plantillas maestras"

Botón "Vista de librería"

Con el botón "Vista de librería" se cambia a la vista de librería. La Task Card "Librerías" y el árbol del proyecto se ocultan.

Consulte también: Uso de la vista de librería (Página 432)

Paleta "Librería del proyecto"

En la paleta "Librería del proyecto" se depositan los objetos que deben utilizarse varias veces en el proyecto.

Paleta "Librerías globales"

En la paleta "Librerías globales" se gestionan librerías globales cuyos elementos pueden reutilizarse en varios proyectos.

En la paleta "Librerías globales" figuran también las librerías suministradas con los productos adquiridos. Estas contienen, p. ej., funciones y bloques de función preprogramados. No es posible modificar las librerías globales suministradas.

Paleta "Elementos"

En esta paleta se visualizan los contenidos de las carpetas en la librería. La paleta "Elementos" no se muestra de manera estándar. Si desea visualizar la paleta "Elementos", primero debe activarse. En la paleta "Elementos" hay tres formas de representación:

- Modo detallado
  En el modo detallado se muestran en forma de tabla las propiedades de las carpetas, las plantillas maestras y los tipos.
- Modo de lista
  En el modo de lista se muestran los contenidos de las carpetas.
- Modo de vista general
  En el modo de vista general se muestran los contenidos de las carpetas con símbolos grandes.

Consulte también: Utilizar la vista de elementos (Página 431)

Paleta "Información"

En la paleta "Información" se visualizan los contenidos de los elementos de librería. Además se muestran las distintas versiones de los tipos y la última fecha de modificación de la versión.

Carpeta "Tipos"

En los directorios "Tipos" se gestionan los tipos y las versiones de tipo de los objetos utilizados como instancias en el proyecto.

Consulte también: Utilización de tipos (Página 458)

Carpeta "Plantillas maestras"

En los directorios "Plantillas maestras" se gestionan las plantillas maestras de los objetos que pueden insertarse como copias en el proyecto.

Consulte también: Utilización de plantillas maestras (Página 455)

Consulte también

Principios básicos de las librerías (Página 427)

Comparación de elementos de librerías (Página 485)

7.10.2.2 Utilizar la vista de elementos

Introducción

La primera vez que se abre la Task Card "Librerías", las paletas "Librería del proyecto" y "Librerías globales" aparecen abiertas, mientras que la paleta "Información" aparece cerrada. La paleta "Elementos" se visualiza en caso necesario.

La vista de elementos muestra los elementos de la librería seleccionada. En la vista de elementos hay tres formas de representación:

- Detalles
  En el modo detallado se muestran en forma de tabla las propiedades de las carpetas, las plantillas maestras y los tipos.
- Lista
  En el modo de lista se muestran los contenidos de las carpetas.
- Sinopsis
  En el modo de vista general se muestran los contenidos de las carpetas con símbolos grandes.

La paleta "Información" muestra el contenido del elemento seleccionado de la librería. Por ejemplo, si se selecciona un tipo en la vista de elementos, se mostrarán las versiones del tipo en la paleta "Información".

Requisitos

Se visualiza la Task Card "Librerías".

Procedimiento

Para utilizar la vista de elementos, proceda del siguiente modo:

1. Haga clic en "Abrir o cerrar vista de elementos" en la paleta "Librería del proyecto" o en la paleta "Librerías globales".
2. Para cambiar la forma de representación de la vista detallada a la vista de lista o general, haga clic sobre el icono correspondiente de la barra de herramientas.

Consulte también

Principios básicos de las librerías (Página 427)

Sinopsis de la Task Card "Librerías". (Página 429)

Utilizar librerías globales (Página 440)

Comparación de elementos de librerías (Página 485)

7.10.3 Uso de la vista de librerías

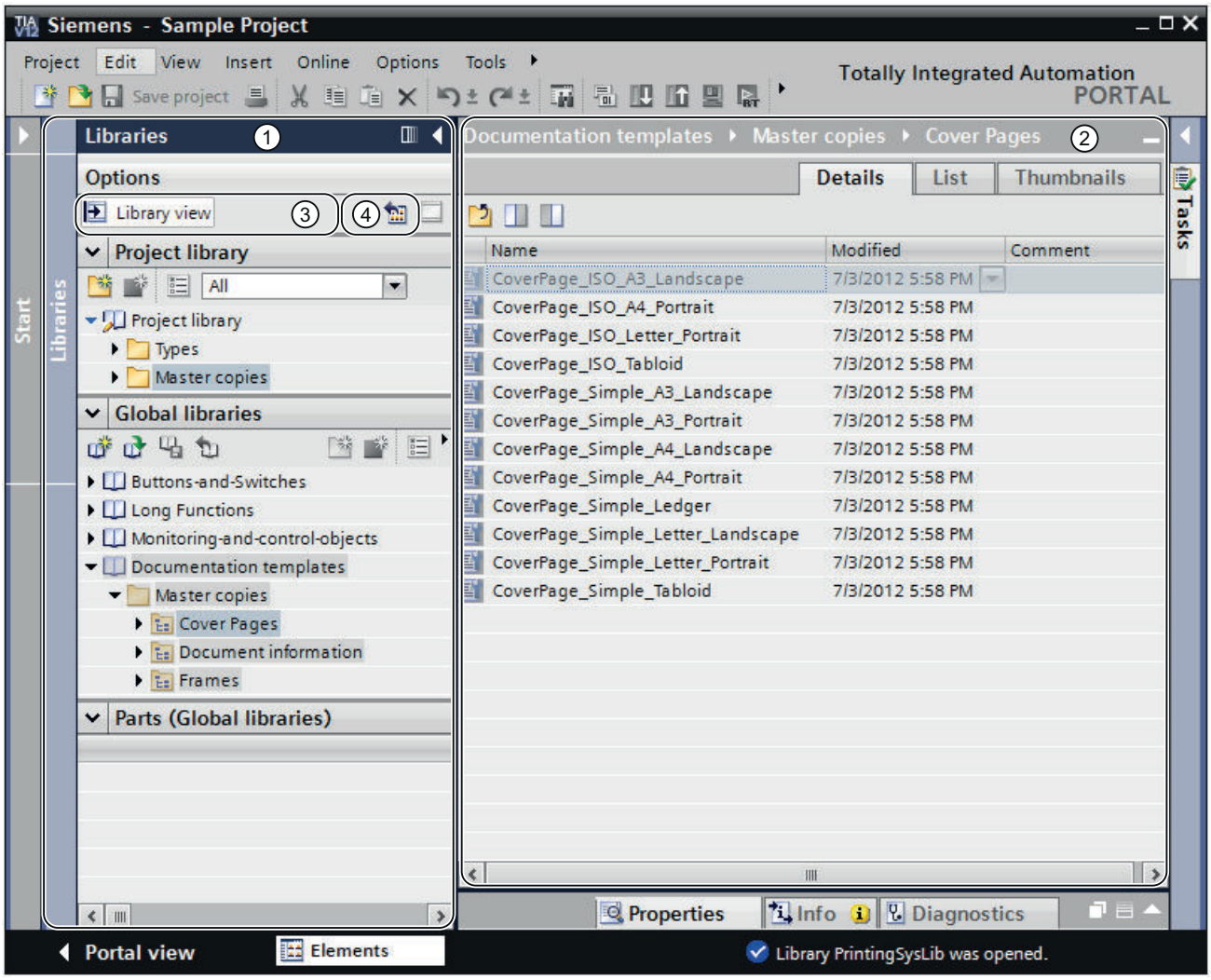
7.10.3.1 Sinopsis de la vista de librerías

Función de la vista de librería

La vista de librería unifica las funciones de la Task Card "Librerías" y las de la ventana general. En la vista de librería se visualizan los elementos de una librería en diferentes vistas y, además, propiedades adicionales de los distintos elementos en la vista detallada. Aparte de esto, en la vista de librería se editan y versionan los tipos.

Estructura de la vista de librería

La figura siguiente muestra los componentes de la vista de librería:



① Árbol de librerías

② Vista general de librerías

③ Botón "Vista de librería"

④ Botón "Abrir o cerrar vista general de librerías"

Árbol de librerías

Salvo pequeñas diferencias, el árbol de librerías es igual que la Task Card "Librerías". Al contrario que en la Task Card, no hay una paleta "Elementos", ya que los elementos se muestran en la vista general de librerías. Además, en el árbol de librerías se puede cerrar la vista de librería o abrir y cerrar la vista general de librerías.

Consulte también: Task Card "Librerías" (Página 429)

Vista general de librerías

La vista general de librerías se corresponde con la ventana general y muestra los elementos del objeto seleccionado en el árbol de librerías. Los elementos pueden visualizarse en tres vistas diferentes:

- Vista detallada
  Los objetos se visualizan en una lista con información adicional (p. ej. la fecha de modificación).
- Vista de lista
  Los objetos se visualizan en una lista sencilla.
- Vista de iconos
  Los objetos se muestran en forma de iconos y ordenados por categorías.

Además, en la vista general de librerías pueden realizarse las siguientes acciones:

- Cambiar el nombre de elementos
- Borrar elementos
- Copiar elementos
- Desplazar elementos
- Editar instancias de tipos
- Versionar tipos
- Solo WinCC: Editar faceplates y tipos de datos de usuario HMI

Consulte también: Ventana general (Página 306)

Consulte también

Principios básicos de las plantillas maestras (Página 455)

Principios básicos de los tipos (Página 458)

Apertura y cierre de la vista de librerías (Página 434)

Principios básicos de las librerías (Página 427)

Comparación de elementos de librerías (Página 485)

7.10.3.2 Apertura y cierre de la vista de librerías

La vista de librería se abre automáticamente en algunos casos, por ejemplo cuando se edita la instancia de test de un tipo o cuando se editan faceplates y tipos de datos de usuario HMI. La vista de librería también puede abrirse manualmente.

Apertura de la vista de librería

Para abrir la vista de librería manualmente, proceda del siguiente modo:

1. Abra la Task Card "Librerías".
2. En la Task Card "Librerías" haga clic en el botón "Abrir vista de librería".
   Se abre el árbol de librerías. La Task Card "Librerías" y el árbol del proyecto se cierran.
3. Si la vista general de librerías no se abre, haga clic en el botón "Abrir/cerrar vista general de librerías" en el árbol de librerías.
   Se abre la vista general de librerías.

Salir de la vista de librería

Para salir de la vista de librería, proceda del siguiente modo:

1. Haga clic en el botón "Cerrar vista de librería" del árbol de librerías.
   Se cierra el árbol de librerías. La Task Card "Librerías" y el árbol del proyecto se abren.

Consulte también

Sinopsis de la vista de librerías (Página 432)

Principios básicos de las librerías (Página 427)

Uso de la Task Card "Librerías" (Página 429)

Utilizar librerías globales (Página 440)

Comparación de elementos de librerías (Página 485)

7.10.4 Uso de la administración de librerías

7.10.4.1 Sinopsis de la administración de librerías

Función de la administración de librerías

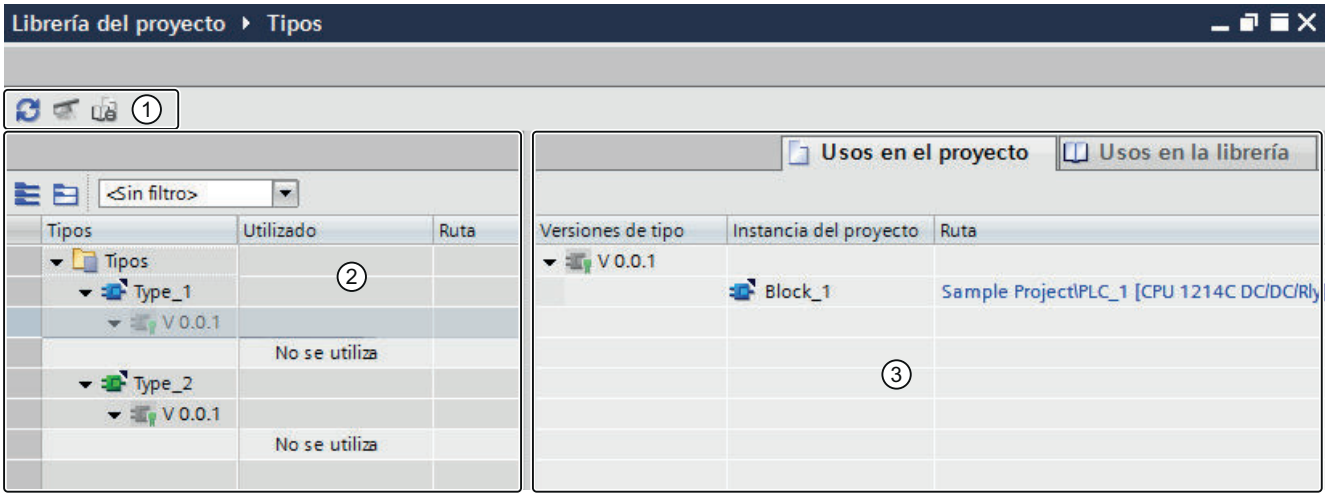
Las plantillas maestras y los tipos con dependencias de otros elementos de librerías están sujetos a algunas restricciones de funciones. Por ejemplo, no es posible borrarlos mientras existan dependencias. De este modo se impide que se inutilicen otros elementos de la librería. La administración de librerías permite identificar dependencias y proporciona una visión global del progreso del trabajo.

La administración de librerías ofrece las siguientes funciones:

- Visualización de las relaciones de los tipos y las plantillas maestras
  Si un tipo se referencia a otros tipos o a plantillas maestras, las relaciones se muestran en la administración de librerías. Asimismo se muestran los elementos de librerías a los que hace referencia un tipo o una plantilla maestra.
- Indicación de las ubicaciones de tipos en el proyecto
- Limitación de los tipos mostrados, con ayuda de filtros

Estructura de la administración de librerías

La figura siguiente muestra los componentes de la administración de librerías:



① Barra de herramientas de la administración de librerías
② Área "Tipos"
③ Área "Usos"

Barra de herramientas de la administración de librerías

La barra de herramientas de la administración de librerías permite realizar las siguientes acciones:

- Actualizar la vista
  Si se ha modificado el proyecto, actualice la vista de la administración de librerías.
- Limpiar librería
  Es posible limpiar la librería del proyecto y las librerías globales. Al limpiar una librería se borran todos los tipos y todas las versiones de tipos que no están enlazados con una instancia del proyecto.
- Armonizar proyecto
  Al armonizar un proyecto, los nombres y las estructuras de rutas de utilización de tipos en el proyecto se adaptan a los correspondientes nombres y estructuras de rutas de los tipos dentro de una librería.

Área "Tipos"

En el área "Tipos" se muestra el contenido de la carpeta que se ha seleccionado en la vista de librería. Con cada tipo se muestran también los tipos a los que hace referencia. Los botones de la barra de herramientas del área "Tipos" permiten abrir o cerrar todos los nodos de tipo. También es posible filtrar la vista con la lista desplegable "Filtro".

Área "Usos"

En el área "Usos" se muestra una relación de las ubicaciones de los tipos y plantillas maestras seleccionados. El área "Usos" se divide en dos fichas:

- Ficha "Usos en el proyecto"
  En la ficha "Usos en el proyecto" se muestran las instancias de versiones de tipo, así como su correspondiente ubicación en el proyecto. Si se selecciona una instancia, en la ventana de inspección pueden mostrarse las referencias cruzadas de la instancia en el proyecto.
- Ficha "Usos en la librería"
  En la ficha "Usos en la librería" se muestran todas las ubicaciones dentro de la librería en las que se utiliza un tipo o una plantilla maestra.

Consulte también

Abrir la administración de librerías (Página 437)

Filtrar la visualización de tipos (Página 438)

Visualizar referencias cruzadas de una instancia (Página 439)

Visualizar instancias del proyecto (Página 438)

Visualizar relaciones entre objetos de librerías (Página 440)

Principios básicos de las librerías (Página 427)

Principios básicos de las plantillas maestras (Página 455)

Principios básicos de los tipos (Página 458)

7.10.4.2 Abrir la administración de librerías

Procedimiento

Para abrir la administración de librerías, proceda del siguiente modo:

1. Abra la vista de librería.
2. Seleccione un tipo o una carpeta cualquiera que contenga tipos.
3. Elija el comando "Administración de librerías" del menú contextual.

Resultado

Se abre la administración de librerías y se muestran los tipos junto con sus versiones.

Consulte también

Sinopsis de la administración de librerías (Página 435)

7.10.4.3 Filtrar la visualización de tipos

Los filtros proporcionan una mejor visión de conjunto de los tipos existentes en librerías extensas. Los filtros ofrecen la posibilidad de limitar los tipos mostrados. Están disponibles los siguientes filtros:

- Visualización de todos los tipos que tienen una versión en el estado "en test" o "en proceso"
- Visualización de todos los tipos habilitados
- Visualización de todos los tipos que no tienen instancias en el proyecto
- Visualización de todos los tipos que tienen más de una versión

Requisitos

La administración de librerías está abierta.

Procedimiento

Para filtrar los tipos mostrados, haga lo siguiente:

1. Seleccione en el área "Tipos" la carpeta a cuyo contenido desea aplicar el filtro.
2. Seleccione el filtro deseado en la lista desplegable "Filtro".
   En el área "Tipos" se muestran los tipos que se ajustan a los criterios de filtro seleccionados.

Consulte también

Sinopsis de la administración de librerías (Página 435)

Estado de las versiones de tipo (Página 460)

Crear una versión de test de un tipo (Página 465)

Editar una versión de test de un tipo (Página 466)

Crear una versión en proceso de un tipo (Página 467)

7.10.4.4 Visualizar instancias del proyecto

En la administración de librerías pueden visualizarse las instancias de todas las versiones de un tipo o de una única versión de tipo. Puede ir directamente a cada instancia del proyecto.

Requisitos

La administración de librerías está abierta.

Procedimiento

Para visualizar las instancias de un tipo o de sus versiones, haga lo siguiente:

1. Seleccione un tipo o una de sus versiones en el área "Tipos".
2. Abra la ficha "Usos en el proyecto" del área "Usos".
   Se mostrarán las instancias del proyecto de cada versión de tipo. En la columna "Ruta" se muestra la ruta en la que se ubica la correspondiente instancia en el proyecto.
3. Opcionalmente: para ir directamente a la instancia correspondiente en el árbol del proyecto, haga clic en la ruta.
   Se ocultará la administración de librerías y se seleccionará la instancia en el árbol del proyecto.

Consulte también

Visualizar referencias cruzadas de una instancia (Página 439)

Uso de tipos (Página 464)

Sinopsis de la administración de librerías (Página 435)

Visualizar relaciones entre objetos de librerías (Página 440)

7.10.4.5 Visualizar referencias cruzadas de una instancia

Puede mostrar las referencias cruzadas de una instancia sin salir de la administración de librerías.

Requisitos

La administración de librerías está abierta.

Procedimiento

Para visualizar las referencias cruzadas de una instancia en el proyecto, haga lo siguiente:

1. Seleccione en el área "Tipos" la versión de tipo cuyas instancias desea mostrar.
2. Seleccione la instancia de la versión de tipo deseada en el área "Usos > Usos en el proyecto".
3. Abra la ficha "Información > Referencias cruzadas" en la ventana de inspección.
   Se mostrarán las referencias cruzadas de la instancia en el proyecto.

Consulte también

Uso de referencias cruzadas (Página 487)

Sinopsis de la administración de librerías (Página 435)

Visualizar instancias del proyecto (Página 438)

7.10.4.6 Visualizar relaciones entre objetos de librerías

En la administración de librerías pueden visualizarse las relaciones entre los diferentes objetos de librerías. Las referencias de las diferentes versiones de tipo a otros objetos de librería se muestran automáticamente en el área "Tipos". En el área "Usos" pueden visualizarse adicionalmente otros objetos de librerías en los que se hace referencia a la versión de tipo correspondiente.

Requisitos

La administración de librerías está abierta.

Procedimiento

Para visualizar qué otros objetos de librerías hacen referencia a una versión de tipo, haga lo siguiente:

1. Seleccione una carpeta, un tipo o una versión en el área "Tipos".
2. Abra la ficha "Usos en la librería" del área "Usos".
   En el área "Usos" se muestra qué otros objetos de librerías hacen referencia a las diferentes versiones de tipo.
3. Opcionalmente: para ir a un objeto de librería con relaciones de referencia, haga clic en la ruta correspondiente de la columna "Ruta".

Consulte también

Visualizar instancias del proyecto (Página 438)

Sinopsis de la administración de librerías (Página 435)

7.10.5 Utilizar librerías globales

7.10.5.1 Crear librería global

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Procedimiento

Para crear una librería global nueva, proceda del siguiente modo:

1. En la paleta "Librerías globales" de la barra de herramientas, haga clic en el icono "Crear librería global" o elija el comando "Librerías globales > Crear librería" del menú "Opciones". Se abre el cuadro de diálogo "Crear librería global".
2. Introduzca el nombre y la ubicación de la nueva librería global.
3. Confirme las entradas realizadas con "Crear".

Resultado

La nueva librería global se crea y se inserta en la paleta "Librerías globales". En la ubicación de la librería global se crea una carpeta con el nombre de la librería global en el sistema de archivos. El archivo de librería tiene la extensión ".al12".

Consulte también

Principios básicos de las librerías (Página 427)

Abrir una librería global (Página 443)

Mostrar las propiedades de las librerías globales (Página 445)

Guardar una librería global (Página 447)

Cerrar una librería global (Página 448)

Borrar una librería global (Página 448)

7.10.5.2 Compatibilidad de las librerías globales

El TIA Portal permite abrir librerías globales que se crearon con una versión anterior del TIA Portal. Sin embargo, si se desea utilizar objetos de las librerías globales en el proyecto, la versión de producto de las librerías globales debe coincidir con la del proyecto abierto. Por esta razón deberán actualizarse las librerías globales en caso necesario.

Abrir librerías globales de versiones anteriores del producto

La siguiente tabla muestra el comportamiento del TIA Portal cuando se abren librerías globales de una versión anterior del producto:

Versión de producto del TIA Portal (extensión de archivo de la librería global correspondiente)	Comportamiento al abrir la librería global
V10.5 (.al10) V11.x (.al11)	Al abrirla, la librería global se actualiza automáticamente a la versión más reciente de la librería tras confirmación por parte del usuario. La librería global actualizada es una copia de la librería original. La librería global original se conserva sin cambios.
V12 (.al12)	La librería global se actualiza al abrirla. Puede elegir actualizar la librería global a la versión V12 SP1 o a la V13. La librería actualizada es en ambos casos una copia de la librería global original. La librería global original se conserva sin cambios. Si actualiza la librería global a la versión V12 SP1, dicha librería se abrirá en modo de compatibilidad.
V12 SP1 (.al12)	Puede elegir abrir la librería global en modo de compatibilidad o actualizarla a la versión actual. Si actualiza la librería global, se creará una copia de la original. La librería global original se conserva sin cambios.

Modo de compatibilidad

Si se ha abierto un proyecto en modo de compatibilidad para la versión de producto V12 SP1 del TIA Portal, la librería global también debe estar en modo de compatibilidad. Pueden utilizarse objetos de la librería global en el proyecto abierto en modo de compatibilidad. Además pueden guardarse nuevos objetos en la librería global si los objetos proceden de un proyecto abierto en modo de compatibilidad. La librería global en modo de compatibilidad puede seguir abriéndose y editándose con el TIA Portal V12 SP1.

Compatibilidad de la versión de librería actual con versiones anteriores

Las librerías globales guardadas en el formato de librería del TIA Portal V13 no son compatibles con versiones anteriores debido a la ampliación de funciones de la versión actual. Las librerías globales en el formato de librería del TIA Portal V13 solo pueden utilizarse en combinación con proyectos del TIA Portal V13.

Consulte también

Compatibilidad de proyectos (Página 341)

Abrir una librería global (Página 443)

Actualizar librerías globales (Página 444)

Actualizar proyectos (Página 343)

7.10.5.3 Abrir una librería global

Las librerías globales se perfeccionan de forma centralizada y pueden utilizarse en varios proyectos. Es posible que varias personas abran al mismo tiempo una librería global desde una ubicación central, siempre y cuando todos los usuarios la abran en modo de solo lectura.

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Procedimiento

Para abrir una librería global, proceda del siguiente modo:

1. En la paleta "Librerías globales" de la barra de herramientas, haga clic en el icono "Abrir librería global" o elija el comando "Librerías globales > Abrir librería" del menú "Opciones". Se abre el cuadro de diálogo "Abrir librería global".
2. Seleccione la librería global que desea abrir. El archivo de librería se reconoce por la extensión ".al[número de versión]". Así pues, las librerías globales guardadas con el TIA Portal V13 tienen la extensión ".al13" en el nombre de archivo.
3. La protección contra escritura está activada para la librería. Si desea modificar la librería global, desactive la opción "Abrir con protección contra escritura".
4. Haga clic en "Abrir".
   Si la versión de la librería coincide con la versión del proyecto, la librería global se abre y se inserta en la paleta "Librerías globales". Si se selecciona una librería global de una versión anterior del TIA Portal, se abre el cuadro de diálogo "Actualizar una librería global". Para más información sobre la actualización de librerías globales, consulte el capítulo "Actualizar librerías globales (Página 444)".

Consulte también

Desarchivar librerías globales (Página 451)

Actualizar librerías globales (Página 444)

Abrir proyectos (Página 342)

Compatibilidad de las librerías globales (Página 441)

Principios básicos de las librerías (Página 427)

Crear librería global (Página 440)

Mostrar las propiedades de las librerías globales (Página 445)

Guardar una librería global (Página 447)

Cerrar una librería global (Página 448)

Borrar una librería global (Página 448)

7.10.5.4 Actualizar librerías globales

Si desea utilizar objetos de una librería global en un proyecto, la versión de la librería debe coincidir con la versión de producto del proyecto. Si se edita un proyecto en modo de compatibilidad V12 SP1, las librerías globales deben estar disponibles en la versión de librería del TIA Portal V12 SP1. Si se edita un proyecto de la versión de producto V13, las librerías globales deben estar disponibles en la versión de librería del TIA Portal V13.

En función de la versión de la librería, se dan las siguientes posibilidades:

- Actualización de la librería global a la versión de producto V13
  Puede actualizar cada librería global de una versión anterior a la versión actual del producto. Al abrir la librería global se muestra el requerimiento correspondiente. Las librerías globales que ya se hayan abierto en modo de compatibilidad pueden actualizarse manualmente a la versión de producto V13.
- Actualización de la librería global a la versión de producto V12 SP1 (modo de compatibilidad)
  Si se abre una librería global de la versión V12 del TIA Portal, existe la posibilidad adicional de actualizar la librería global a la versión de librería V12 SP1. Sin embargo, en este caso la librería global solo podrá utilizarse en combinación con proyectos del TIA Portal V12 SP1.

Actualización de librerías globales de V11.x o versiones anteriores

Para actualizar una librería global del TIA Portal V11.x o anterior, haga lo siguiente:

1. Abra la librería global.
   Se abrirá el cuadro de diálogo "Actualizar una librería global".
2. Haga clic en "Aceptar".
   Se creará y actualizará una copia de la librería global. Se añadirá "_V13" al nombre de la copia de la librería global. Se abre la librería global.

Actualización de librerías globales de V12

Para actualizar una librería global del TIA Portal V12, haga lo siguiente:

1. Abra la librería global.
   Se abrirá el cuadro de diálogo "Actualizar una librería global".
2. Seleccione la versión de destino para la actualización:
   - Para actualizar la librería global a la versión de producto V13, haga clic en "Sí".
   - Para actualizar la librería global a la versión de producto V12 SP1 y trabajar en modo de compatibilidad, haga clic en "No".

   Se creará y actualizará una copia de la librería global. Se añadirá "_V13" o "_V12SP1" al nombre de la copia de la librería global. Se abre la librería global.

Actualización de librerías de V12 SP1 o utilización en modo de compatibilidad

Para actualizar una librería global del TIA Portal V12 SP1 o utilizarla en modo de compatibilidad, haga lo siguiente:

1. Abra la librería global.
   Se abrirá el cuadro de diálogo "Actualizar una librería global".
2. Elija lo que desea hacer con la librería global:
   – Para actualizar la librería global a la versión de producto V13, haga clic en "Sí".
   – Para utilizar la librería global en modo de compatibilidad, haga clic en "No".

   Se creará y actualizará una copia de la librería global, o la librería se abrirá sin cambios. Se añadirá "_V13" al nombre de la copia de la librería global.

Actualización manual de librerías en modo de compatibilidad a V13

Para esta operación debe haber una librería global abierta en modo de compatibilidad para el TIA Portal V12 SP1.

Para actualizar manualmente librerías globales a la versión de librería V13, haga lo siguiente:

1. Haga clic con el botón derecho del ratón en la librería global que desea actualizar.
2. Elija el comando "Actualizar librería" del menú contextual.
   Se muestra el cuadro de diálogo "Actualizar".
3. Confirme haciendo clic en "Sí".
   Se creará y actualizará una copia de la librería global. Se añadirá "_V13" al nombre de la copia de la librería global. Se abrirá la librería global actualizada. Se cerrará la librería global original.

Consulte también

Abrir una librería global (Página 443)

Compatibilidad de las librerías globales (Página 441)

Actualizar proyectos (Página 343)

7.10.5.5 Mostrar las propiedades de las librerías globales

Las librerías globales tienen propiedades para describir detalladamente la librería respectiva. Las propiedades visualizadas son las siguientes:

- Información general sobre la librería
  Aquí obtendrá por ejemplo información sobre la fecha de creación, el autor, la ruta, el tamaño del archivo, indicaciones sobre los derechos de autor, etc. Muchos atributos se pueden modificar.
- Historial de la librería
  El historial de la librería contiene un resumen de las migraciones ejecutadas. Aquí también se puede llamar el archivo de registro de las migraciones. Además, el historial de la librería contiene información sobre las actualizaciones de la librería global.

- Support Packages en la librería
  Es posible visualizar un resumen del software que se requiere adicionalmente para poder editar todos los dispositivos del proyecto.
- Productos de software en la librería
  Es posible visualizar un resumen de todos los productos de software instalados que son indispensables en el proyecto.

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Procedimiento

Para visualizar las propiedades de una librería global, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la librería global cuyas propiedades desea visualizar.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo con las propiedades de la librería global.
3. Seleccione en la navegación local aquellas propiedades que desee visualizar.

Consulte también

Abrir una librería global (Página 443)

Principios básicos de las librerías (Página 427)

Crear librería global (Página 440)

Guardar una librería global (Página 447)

Cerrar una librería global (Página 448)

Borrar una librería global (Página 448)

7.10.5.6 Mostrar los informes de las librerías globales

Cuando se actualizan o limpian librerías globales o se asigna una versión común a varios tipos, se crean informes. Dichos informes incluyen todos los cambios realizados en la librería global. Los informes se guardan junto con la librería global y siempre están disponibles una vez que se abre la librería global.

Procedimiento

Para abrir los informes de una librería global, haga lo siguiente:

1. Abra la librería global en la Task Card "Librerías" o en la vista de librería.
2. Abra la carpeta subordinada "Datos comunes > Informes".
3. Haga doble clic en el informe deseado.
   El informe se abre en el área de trabajo.

Consulte también

Actualizar una librería con los contenidos de otra librería (Página 481)

7.10.5.7 Guardar una librería global

Después de modificar una librería global, guárdela. El comando "Guardar librería como" permite guardar una librería global con un nombre diferente.

Nota

Compatibilidad con versiones anteriores del TIA Portal

Tenga en cuenta que no es posible abrir las librerías globales en versiones anteriores del TIA Portal una vez se hayan guardado en la versión actual.

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Guardar modificaciones

Para guardar una librería global, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la librería global que desea guardar.
2. Elija el comando "Guardar librería" del menú contextual.

Guardar una librería global con un nombre diferente

Para guardar una librería global con un nombre diferente, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la librería global que desea guardar con otro nombre.
2. Elija el comando "Guardar librería como" del menú contextual.
   Se abre el diálogo "Guardar librería global como".
3. Seleccione la ubicación e introduzca el nombre de archivo.
4. Confirme las entradas realizadas con "Guardar".
   La librería se guarda en la ubicación indicada con el nuevo nombre elegido. La librería original se conserva.

Consulte también

Trabajar con ficheros de librerías globales (Página 449)

Archivar librerías globales (Página 450)

Principios básicos de las librerías (Página 427)

Crear librería global (Página 440)

Abrir una librería global (Página 443)

Mostrar las propiedades de las librerías globales (Página 445)

Cerrar una librería global (Página 448)

Borrar una librería global (Página 448)

7.10.5.8 Cerrar una librería global

Las librerías globales son independientes de los proyectos. Esto implica que no se cierran junto con el proyecto. Por tanto, las librerías globales deben cerrarse explícitamente.

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Procedimiento

Para cerrar una librería global abierta, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la librería global que desea cerrar.
2. Elija el comando "Cerrar librería" del menú contextual.
   Si se han realizado cambios en la librería global, elija si desea guardar los cambios o no.
   La librería global se cerrará.

Consulte también

Crear librería global (Página 440)

Abrir una librería global (Página 443)

Mostrar las propiedades de las librerías globales (Página 445)

Guardar una librería global (Página 447)

Principios básicos de las librerías (Página 427)

Borrar una librería global (Página 448)

7.10.5.9 Borrar una librería global

Si una librería global ya no es necesaria, es posible borrarla. Las librerías suministradas por Siemens no pueden borrarse.

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Procedimiento

Para borrar una librería global, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la librería global que desea borrar.
2. Elija el comando "Borrar" del menú contextual.
3. Confirme la consulta de seguridad con "Sí".

Resultado

Los librería global se eliminará de la paleta "Librerías globales". Todo el directorio de la librería global se borrará del sistema de archivos.

Consulte también

Principios básicos de las librerías (Página 427)

Crear librería global (Página 440)

Abrir una librería global (Página 443)

Mostrar las propiedades de las librerías globales (Página 445)

Guardar una librería global (Página 447)

Cerrar una librería global (Página 448)

7.10.5.10 Archivar y desactivar librerías globales

Trabajar con ficheros de librerías globales

Si desea hacer una copia de seguridad de librerías globales en un disco duro externo o si desea enviar librerías globales por correo electrónico, por ejemplo, reduzca el espacio necesario de memoria de la librería con la función de archivación.

Posibilidades de reducción del tamaño del proyecto

Existen dos formas de reducir el espacio de memoria necesario de librerías globales:

- Creación de un fichero comprimido de librerías globales
  Los ficheros de librerías globales son archivos comprimidos que contienen una librería global completa, incluida toda la estructura de carpetas de la librería. Antes de convertir el directorio con la librería global en un archivo comprimido se reducen todos los archivos a sus componentes más esenciales para reducir aún más el espacio de memoria necesario. Así, los ficheros comprimidos de librerías globales son especialmente adecuados para el envío por correo electrónico.
  Los ficheros comprimidos de librerías globales del TIA Portal V13 tienen la extensión ".zal13". Los ficheros del TIA Portal V12.x tienen la extensión ".zal12".
  Para abrir un fichero comprimido de una librería global es necesario desarchivar el fichero. El archivo comprimido se descomprime en la ubicación que haya seleccionado, con la estructura de carpetas completa y todos los archivos.
- Librerías globales minimizadas
  Se puede renunciar a la compresión adicional en un archivo comprimido y elaborar en su lugar una copia del directorio de la librería global. Los archivos contenidos se reducen a los componentes esenciales. Con ello se minimiza el espacio de memoria necesario. Se mantiene el volumen de funciones completo de la librería global, que se puede cargar de la manera habitual.
  Las librerías globales minimizadas son especialmente adecuadas para la archivación, p. ej. en un soporte de datos externo.

Consulte también

Archivar librerías globales (Página 450)

Desarchivar librerías globales (Página 451)

Archivar librerías globales

El espacio de memoria necesario para una librería global puede reducirse comprimiéndola en un archivo comprimido. También es posible reducir el espacio necesario en memoria guardando la librería global reducida a sus componentes esenciales. Ambas opciones se consiguen con la función de archivación para librerías globales. Cuando se archiva una librería, la versión original se conserva. Por tanto, las librerías en modo de compatibilidad para el TIA Portal V12 SP1 siguen siendo compatibles con el TIA Portal V12 SP1.

Requisitos

La librería global está cargada.

Procedimiento

Para archivar una librería global, proceda del siguiente modo:

1. Seleccione la librería global que desea archivar.
2. Haga clic en la librería global con el botón derecho del ratón y elija el comando "Archivar" del menú contextual.
   Se abre el cuadro de diálogo "Archivar librería global como...".
3. Seleccione el directorio en el que desea guardar el archivo comprimido o el nuevo directorio de la librería global.
   El directorio no puede estar en un directorio de proyecto ni dentro del directorio de una librería global.
4. Seleccione el tipo de archivo de la lista desplegable "Tipo de archivo":
   – Fichero Librerías globales si desea crear un archivo comprimido de la librería.
   – Librería global minimizada si solo desea crear una copia del directorio de la librería con espacio en memoria minimizado.
5. Introduzca un nombre de archivo en el campo "Nombre de archivo" para crear un archivo comprimido. Si crea una librería global minimizada, introduzca en su lugar, en el campo "Nombre de archivo", el nombre del directorio nuevo de librería.
6. Haga clic en "Guardar".

Resultado

Se genera un archivo comprimido con la extensión ".zal13" o ".zal12". La extensión del archivo depende de si la librería se ha archivado en modo de compatibilidad V12 SP1 o en formato de librería del TIA Portal V13.

El archivo contiene el directorio completo de la librería global. Los diferentes archivos de la librería global se han reducido además a sus componentes esenciales para ahorrar espacio en la memoria.

Si ha minimizado la librería global, solo se crea una copia del directorio original de la librería global en el lugar deseado. Para ahorrar espacio en la memoria, los archivos contenidos en ella se han reducido a sus componentes esenciales.

Consulte también

Trabajar con ficheros de librerías globales (Página 449)

Desarchivar librerías globales (Página 451)

Desarchivar librerías globales

Antes de poder utilizar una librería global archivada debe desarchivarla. Al hacerlo, la librería global se descomprime y a continuación se abre en el TIA Portal.

Procedimiento

Para descomprimir el fichero de una librería global, proceda del siguiente modo:

1. Elija el comando "Librerías globales > Desarchivar librería" del menú "Opciones". Se abre el cuadro de diálogo "Desarchivar la librería global archivada".
2. Seleccione el archivo comprimido.
3. Si desea cargar la librería global con protección contra escritura, active la casilla de verificación "Abrir con protección contra escritura".
4. Haga clic en "Abrir".
5. Se abre el cuadro de diálogo "Buscar carpeta".
6. Seleccione el directorio de destino en el que desea descomprimir la librería global archivada.
7. Haga clic en "Aceptar".

Resultado

La librería global se descomprime en el directorio seleccionado y se abre inmediatamente.

Si el archivo comprimido contiene una librería global del TIA Portal V12 SP1, dicha librería se abrirá en modo de compatibilidad. En caso necesario, actualice la librería manualmente a la versión del TIA Portal V13.

Consulte también

Trabajar con ficheros de librerías globales (Página 449)

Archivar librerías globales (Página 450)

Abrir una librería global (Página 443)

Compatibilidad de las librerías globales (Página 441)

Actualizar librerías globales (Página 444)

7.10.5.11 Utilizar librerías de empresa globales

Principios básicos de las librerías de empresa

Introducción

Las librerías de empresa son librerías globales proporcionadas por un administrador y asignadas al TIA Portal. El administrador puede en todo momento asignar nuevas librerías o modificar las existentes. Las librerías nuevas se cargan automáticamente en el TIA Portal tras confirmación por parte del usuario. En cuanto hay una versión nueva disponible de una librería de empresa, las librerías de empresa existentes se actualizan igualmente a la última versión, tras confirmación por parte del usuario.

Las librerías de empresa están en la paleta "Librerías globales" de la Task Card "Librerías", igual que las librerías globales normales.

Preparación de librerías de empresa

Las librerías de empresa pueden guardarse en cualquier directorio del disco duro del PC o bien en una unidad de red. Para asignar librerías de empresa al TIA Portal, utilice un archivo XML. En el archivo XML están guardados los directorios y nombres de las librerías de empresa asignadas. El archivo XML debe estar guardado en el directorio siguiente del disco duro del PC:

`C:\ProgramData\Siemens\Automation\Portal V13\CorporateSettings\`

El archivo XML debe tener el nombre "`CorporateSettings.xml`".

El archivo de configuración lo copia el propio usuario en el directorio correspondiente o bien se asigna al usuario desde la red empresarial. La configuración válida se aplica automáticamente al iniciar el TIA Portal. Mientras el TIA Portal esté abierto, se vigilará continuamente el directorio con los archivos de configuración. Si el archivo de configuración ha cambiado, recibirá una petición para aplicar la nueva configuración. Es posible rechazar dicha petición un máximo de dos veces. La siguiente petición aparece al cabo de tres horas en cada caso. A la tercera petición tendrá que aplicar la nueva configuración. El administrador del proyecto prepara un archivo de configuración modificado cuando se agregan o eliminan librerías de empresa, por ejemplo.

Posibilidades como administrador del proyecto

Puede asignar de forma automatizada el archivo de configuración o las librerías de empresa a los PC de los miembros del equipo, o hacerles llegar actualizaciones. Esta función no forma parte del TIA Portal y requiere una infraestructura de TI correspondiente en su empresa. Si desea administrar el archivo de configuración de forma centralizada, hable con el responsable de TI de su empresa.

Crear un archivo de configuración

Para que las librerías de empresa estén disponibles en el TIA Portal, utilice un archivo de configuración en formato XML. El archivo de configuración contiene los directorios y nombres de archivo de las librerías que deben cargarse. A continuación se explica cómo crear el archivo de configuración XML y dónde debe guardarse.

Procedimiento

Para poner a disposición un archivo de configuración para librerías de empresa, proceda del siguiente modo:

1. Cree un archivo de configuración XML con el contenido indicado a continuación. Utilice la codificación "UTF-8".
2. Guarde el archivo XML con el nombre "CorporateSettings.xml".
3. Guarde el archivo en el directorio siguiente del PC:
   C:\ProgramData\Siemens\Automation\Portal V13\CorporateSettings\

Contenido del archivo de configuración XML

El archivo de configuración XML debe tener el contenido siguiente:

XML

```
<?xml version="1.0" encoding="utf-8"?>
<Document>
 <Settings.Settings ID="0">
  <ObjectList>
   <Settings.General ID="1" AggregationName="General">
    <AttributeList>
     <CorporateLibraryPaths>
     <!-- Ejemplo de una entrada -->
     <Item>D:\CorporateLibraries\Corporate_Library_1.al13</Item>
     <!-- Introduzca aquí todas las demás librerías globales, si las hubiera. -->
     </CorporateLibraryPaths>
    </AttributeList>
   </Settings.General>
  </ObjectList>
 </Settings.Settings>
</Document>
```

Resultado

Una vez que se ha guardado el archivo de configuración XML en el directorio correspondiente, recibirá en el TIA Portal la petición de cargar la librería de empresa

7.10.6 Crear carpetas en una librería

Dentro de las librerías, los elementos de librería se guardan en las carpetas "Tipos" y "Plantillas maestras" dependiendo de qué tipo sean. Para agregar más plantillas maestras y tipos, se deben crear más carpetas debajo de "Tipos" y "Plantillas maestras".

Requisitos

- Se visualiza la Task Card "Librerías" o se abre la vista de librería.
- Si desea crear carpetas nuevas dentro de una librería global, esta debe estar abierta con derechos de escritura.

Procedimiento

Para crear una carpeta nueva, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una carpeta cualquiera de la librería.
2. Elija el comando "Agregar carpeta" del menú contextual.
   Se crea una carpeta nueva.
3. Introduzca un nombre para la carpeta.

Consulte también

Trabajar con tipos en la librería del proyecto (Página 463)

Filtrar plantillas maestras (Página 457)

7.10.7 Utilizar plantillas maestras

7.10.7.1 Principios básicos de las plantillas maestras

Las plantillas maestras se utilizan para crear de forma estándar elementos utilizados frecuentemente. Basándose en una plantilla maestra es posible crear cuantos elementos se necesiten e insertarlos en el proyecto. Los elementos heredan las propiedades de la plantilla maestra.

Las plantillas maestras se almacenan en la librería del proyecto o en una librería global. Las plantillas maestras de la librería del proyecto solo pueden utilizarse dentro del proyecto. Si se crea la plantilla maestra en una librería global, puede utilizarse en diferentes proyectos.

En la librería pueden crearse, por ejemplo, los siguientes elementos como plantillas maestras:

- Dispositivos con su configuración
- Tablas de variables
- Perfiles de instrucción
- Tablas de observación
- Elementos de los ajustes de la documentación, p. ej. portadas y marcos
- Bloques y grupos que contienen varios bloques
- Tipos de datos PLC y grupos que contienen varios tipos de datos PLC
- Listas de textos
- Categorías
- Objetos tecnológicos

En muchos casos los objetos que se agregan como plantillas maestras contienen más elementos. Así, por ejemplo, una CPU puede contener bloques. Si los elementos contenidos son utilizaciones de una versión de tipo, en la librería se crean automáticamente las versiones utilizadas de los tipos. A continuación, los elementos contenidos se utilizan como instancia y se enlazan con el tipo.

Consulte también

Agregar plantillas maestras (Página 456)

Uso de plantillas maestras (Página 457)

Principios básicos de los tipos (Página 458)

Filtrar plantillas maestras (Página 457)

7.10.7.2 Agregar plantillas maestras

Guarde los objetos que desee utilizar reiteradamente como plantilla maestra en la librería del proyecto o en una librería global. Al crear plantillas maestras están disponibles las siguientes posibilidades:

- Seleccione uno o más elementos y genere una plantilla maestra a partir de estos
- Seleccione varios elementos y genere una plantilla maestra de cada uno de ellos

Requisitos

- Se visualiza la Task Card "Librerías".
- Si se agrega un dispositivo como plantilla maestra, el dispositivo debe cumplir los requisitos siguientes:
  - El dispositivo está compilado y en estado coherente.
  - El dispositivo no contiene instancias de test de un tipo.
- Cuando se agrega la plantilla maestra a una librería global, esta está abierta con derechos de escritura.

Generar una plantilla maestra a partir de uno o más elementos

Para generar una plantilla maestra a partir de uno o más elementos, proceda del siguiente modo:

1. Abra la librería en la Task Card "Librerías".
2. Arrastre uno o más elementos mediante Drag & Drop a la carpeta "Plantillas maestras" o a cualquier subcarpeta de "Plantillas maestras".
   Alternativa: copie los elementos en el portapapeles y péguelos en el lugar deseado.
   El elemento se inserta en la librería como plantilla maestra única.

Generación de una plantiílla maestra por cada uno de los elementos seleccionados

Para generar una plantilla maestra por cada elemento a partir de varios elementos, proceda del siguiente modo:

1. Abra la librería en la Task Card "Librerías".
2. Copie en el portapapeles los elementos que desea crear como plantillas maestras.
3. En la librería, haga clic con el botón derecho del ratón en la carpeta "Plantillas maestras" o en cualquier subcarpeta.
4. Elija el comando "Insertar como plantillas maestras separadas" del menú contextual.
   Los elementos se insertan en la librería como plantillas maestras separadas.
   Automáticamente se generará un tipo a partir de cada objeto que pueda haber dentro.

Consulte también

Principios básicos de las plantillas maestras (Página 455)

Uso de plantillas maestras (Página 457)

Principios básicos de las librerías (Página 427)

Agregar tipos a la librería del proyecto (Página 463)

7.10.7.3 Filtrar plantillas maestras

Para mejorar la claridad en caso de haber un gran número de plantillas maestras es posible filtrar la visualización según la clase de plantilla maestra.

Requisitos

Se visualiza la Task Card "Librerías" o se abre la vista de librería.

Procedimiento

Para filtrar la vista, proceda del siguiente modo:

1. Abra la carpeta "Plantillas maestras" en la librería del proyecto o en una librería global.
2. Seleccione en la lista desplegable de la barra de herramientas la clase de objetos que desea visualizar en "Plantillas maestras".

Resultado

Solo se visualizará la clase de plantillas maestras seleccionadas. Para regresar a una vista no filtrada, ponga el filtro a "Todos".

Consulte también

Principios básicos de las librerías (Página 427)

Crear carpetas en una librería (Página 454)

Principios básicos de las plantillas maestras (Página 455)

Uso de plantillas maestras (Página 457)

Utilizar la vista de elementos (Página 431)

7.10.7.4 Uso de plantillas maestras

Las plantillas maestras se encuentran en la librería del proyecto o en una librería global.

Requisitos

Se visualiza la Task Card "Librerías".

Procedimiento

Para insertar una plantilla maestra en el proyecto, proceda del siguiente modo:

1. Abra la carpeta "Plantillas maestras" o cualquier otra subcarpeta de "Plantillas maestras" en una librería.
2. Arrastre una o más plantillas maestras mediante Drag & Drop hasta la ubicación deseada. Se inserta una copia de la plantilla maestra.

O:

1. Abra la vista de elementos.
2. Arrastre la plantilla maestra mediante Drag & Drop desde la paleta "Elementos" hasta la ubicación deseada.
   Se inserta una copia de la plantilla maestra.

Consulte también

Principios básicos de las plantillas maestras (Página 455)

Agregar plantillas maestras (Página 456)

Filtrar plantillas maestras (Página 457)

Principios básicos de las librerías (Página 427)

Utilizar la vista de elementos (Página 431)

7.10.8 Utilizar tipos y sus versiones

7.10.8.1 Principios básicos de los tipos

Utilización de tipos

Los tipos son elementos necesarios para la ejecución de programas de usuario. Los tipos pueden versionarse y perfeccionarse de forma centralizada.

Es posible depositar como tipo los elementos siguientes en la librería del proyecto o en la librería global:

- funciones (FC)
- bloques de función (FB)
- tipos de datos PLC
- tipos de datos de usuario
- faceplates
- imágenes
- funciones personalizadas

A partir de las versiones de los tipos es posible derivar cualquier cantidad de instancias en el proyecto. Después de ello, las instancias están vinculadas con la versión del tipo. Al utilizar tipos de una librería global, el tipo también se creará en la librería del proyecto. Si en la librería del proyecto ya existe el tipo, en caso necesario se completará con las versiones del tipo que falten. A partir de entonces la instancia solo estará enlazada con la versión de tipo respectiva en la librería del proyecto.

Los tipos y sus instancias se marcan con un triángulo negro. La figura siguiente muestra una instancia marcada con un triángulo negro y un bloque de programa común:

Principios básicos del versionamiento de tipos

Mediante el versionamiento de tipos existe la posibilidad de desarrollar tipos de forma centralizada y luego extender la versión más reciente a los proyectos en forma de actualización. Así, las correcciones de errores y las ampliaciones funcionales pueden integrarse fácilmente en los proyectos existentes. Si se ha creado una versión nueva de una librería global, los proyectos existentes se actualizan en un proceso automático. De esta manera se minimizan los errores y se reduce el trabajo necesario para mantener grandes soluciones de automatización con muchos proyectos individuales.

El versionamiento hace que el proceso de desarrollo de los distintos tipos sea más comprensible. Antes de habilitar una versión es posible probar en un entorno de test si los cambios realizados en un tipo se han integrado correctamente en el proyecto existente. Cuando se haya asegurado de que todo funciona perfectamente, habilite una versión para la utilización productiva. El historial de las distintas instancias puede verse en cualquier momento en el proyecto y es posible determinar de qué versión se deriva la instancia.

Además, el TIA Portal comprueba automáticamente si hay objetos relacionados con las distintas versiones de un tipo. Pueden ser objetos relacionados, por ejemplo, tipos de datos PLC referenciados en un bloque, u otros bloques. Al crearse un tipo o al copiarse entre librerías se tienen en cuenta todos los objetos relacionados. Para que no se produzcan incoherencias en el proyecto, antes de la habilitación se comprueba la coherencia de las versiones de los tipos.

Versiones de tipos

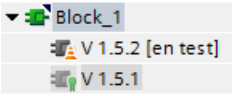
Cada tipo tiene asignadas versiones. El número de versión se visualiza en la Task Card "Librerías" y en la vista de librería junto con el tipo respectivo. Además, se muestra en el árbol del proyecto junto a las instancias de tipos. De esta forma es posible ver en qué versión se utiliza una instancia en el proyecto.

El número de versión consta de tres partes separadas por puntos. Las dos primeras partes se pueden determinar libremente. Ambas admiten también números de 1 a 999. La tercera parte es el número de versión interno. Este se incrementa automáticamente en uno tan pronto como se edita una instancia correspondiente a la versión. Si se habilita la versión, el número de versión interno se pone a 1.

Las versiones de tipos pueden tener tres estados:

- En proceso (faceplates y tipos de datos de usuario HMI)
- En test (todas las clases de tipos excepto los faceplates y tipos de datos de usuario HMI)
- Habilitado

La figura siguiente muestra un tipo con dos versiones. Un versión tiene el estado "en test" y la otra está habilitada:



Consulte también

Estado de las versiones de tipo (Página 460)

Principios básicos de las plantillas maestras (Página 455)

Agregar tipos a la librería del proyecto (Página 463)

Uso de tipos (Página 464)

7.10.8.2 Estado de las versiones de tipo

Las versiones de los tipos pueden tener tres estados diferentes. Los estados se reconocen por la instancia o en la librería.

Estado "en proceso"

Solo tienen el estado "en proceso" las versiones de faceplates y de tipos de datos de usuario HMI. Si una versión se está procesando aparece "en proceso" al lado de la versión en la librería.

Si se crea un tipo nuevo o una versión nueva de un tipo habilitado, el tipo se pondrá al estado "en proceso".

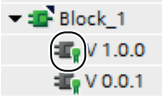
Los tipos que tienen el estado "en proceso" se editan en la vista de librería. No es necesario que exista relación con una instancia en el proyecto. La compatibilidad del tipo se comprueba mediante una prueba de coherencia al realizar la habilitación.

Estado "en test"

Todos los tipos excepto los faceplates y los tipos de datos de usuario HMI pueden tener el estado "en test". Si una versión se está probando aparece "en test" al lado de la instancia y en la librería. Una versión en test está enlazada con una instancia de test en el proyecto. De esta manera es posible probar los efectos de los cambios en un entorno de test incluyendo todas las funciones online antes de habilitar un tipo para su uso durante el funcionamiento.

Estado "Habilitado"

El estado "Habilitado" está disponible para todos los tipos sin importar la ubicación. Cuando se habilita una versión, el símbolo de la versión se identifica en la librería con un sello.



Las versiones habilitadas se pueden abrir en su instancia en modo de solo lectura. Si se desea editar una versión habilitada es necesario crear primero una versión nueva "en proceso" o en "en test".

Consulte también

Principios básicos de los tipos (Página 458)

Uso de tipos (Página 464)

Crear una versión de test de un tipo (Página 465)

Editar una versión de test de un tipo (Página 466)

Crear una versión en proceso de un tipo (Página 467)

Comprobar la coherencia de una versión (Página 468)

Rechazar versiones (Página 468)

Habilitar versiones (Página 469)

Asignar una versión (Página 477)

Actualizar un proyecto a las últimas versiones (Página 472)

Deshacer la conexión entre la instancia y el tipo (Página 473)

Filtrar la visualización de tipos (Página 438)

7.10.8.3 Mostrar versión de tipo habilitada

Si desea visualizar una versión habilitada pero no desea editarla, abra la instancia en modo de solo lectura. Todos los tipos con excepción de los faceplates y los tipos de datos de usuario HMI pueden abrirse directamente en la instancia. Los faceplates y los tipos de datos de usuario HMI solo pueden abrirse en la Task Card "Librerías" o en la vista de librería.

Requisitos

La versión habilitada tiene una instancia en el proyecto, excepto si el tipo es un faceplate o un tipo de datos de usuario HMI.

Abrir una versión de tipo en una instancia

Para abrir una versión habilitada de un tipo en modo de solo lectura a partir de una instancia, proceda del siguiente modo:

1. Seleccione la versión habilitada en la instancia del árbol del proyecto.
2. Haga clic en la instancia con el botón derecho del ratón y elija el comando "Abrir" del menú contextual.
   Se abre la instancia en modo de solo lectura.

Abrir una versión de tipo en la Task Card "Librerías" o en la vista de librería

Para abrir una versión habilitada de un tipo en la Task Card "Librerías" o en la vista de librería, proceda del siguiente modo:

1. Seleccione la versión.
2. Haga clic en la versión con el botón derecho del ratón y elija el comando "Abrir" del menú contextual.
   Si se trata de un faceplate o de un tipo de datos de usuario HMI, se abre directamente en la vista de librería. En este caso, no ejecute los demás pasos. Si se trata de otro tipo, se abre el cuadro de diálogo "Abrir tipo".
3. En la lista de instancias seleccione la instancia con la versión que desea visualizar.
4. Confirme con "Aceptar".
   Se abre la instancia en modo de solo lectura.

7.10.8.4 Mostrar propiedades de un tipo o de una versión

En las propiedades de un tipo o sus versiones es posible visualizar y, en su caso, modificar las propiedades siguientes:

- nombre del tipo
- número de versión (visible solo en una versión)
- fecha y hora de la última modificación de la versión (visible solo en una versión)
- autor de la versión (visible solo en una versión)
- comentario acerca del tipo o la versión del tipo
  El comentario de un tipo se puede modificar.
- librería original (visible solo en una versión)
  Se muestran el proyecto y la librería desde la que se ha generado la versión actual del tipo. Esta información es importante, por ejemplo, para encontrar el original del tipo después de haberlo copiado de otra librería.
- ID del tipo o de la versión del tipo
  Con esta ID es posible identificar de forma unívoca el tipo o su versión, por ejemplo si existen tipos o versiones con una denominación idéntica dentro de la librería del proyecto o la librería global. La ID no puede modificarse y se asigna automáticamente.

Procedimiento

Para mostrar las propiedades de un tipo o una versión e introducir un comentario, proceda del siguiente modo:

1. Seleccione un tipo o la versión de un tipo en la Task Card "Librerías" o en la vista de librería.
2. Haga clic en el tipo o en una de sus versiones con el botón derecho del ratón y elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo "Propiedades".
3. En caso necesario, introduzca un comentario sobre el tipo en el campo "Comentario" o modifique un comentario existente.

7.10.8.5 Trabajar con tipos en la librería del proyecto

Agregar tipos a la librería del proyecto

En la librería del proyecto es posible crear tipos a partir de distintos elementos para reutilizarlos en el proyecto. Pueden crearse los siguientes elementos, por ejemplo:

- bloques de programa
- faceplates
- tipos de datos PLC
- tipo de datos de usuario HMI

Si se agrega un elemento como tipo a la librería del proyecto y dicho elemento depende de otros elementos, entonces los elementos dependientes también se crearán como tipo automáticamente.

Después de agregar un tipo a la librería del proyecto, el tipo está enlazado con el elemento agregado del proyecto.

Requisitos

- Se visualiza la Task Card "Librerías".
- Los elementos que se desean agregar como tipo están compilados.
- Los elementos tienen un estado coherente.

Procedimiento

Para agregar un elemento ya existente como tipo a la librería del proyecto, proceda del siguiente modo:

1. Abra la librería del proyecto en la Task Card "Librerías".
2. Arrastre uno o varios elementos mediante Drag & Drop a la carpeta "Tipos" o a cualquier subcarpeta de "Tipos".
   Alternativa: copie los elementos del árbol de proyecto en el portapapeles y péguelos en la carpeta deseada de la librería del proyecto.
   Se abre el cuadro de diálogo "Crear tipo".

3. Introduzca las propiedades del tipo nuevo:
   - Introduzca un nombre para el tipo nuevo en el campo "Nombre del tipo".
   - Introduzca un número de versión para el tipo nuevo en el campo "Versión".
   - En el campo "Autor" introduzca el nombre del autor responsable del tipo.
   - Introduzca un comentario acerca del tipo en el campo "Comentario".
4. Confirme con "Aceptar".
   Se creará el tipo nuevo con una versión habilitada. La versión está enlazada con el elemento agregado.

Consulte también

Principios básicos de los tipos (Página 458)

Principios básicos de las librerías (Página 427)

Agregar plantillas maestras (Página 456)

Uso de tipos

Para utilizar la versión de un tipo debe crearse una instancia de la versión en una ubicación adecuada en el árbol del proyecto. También es posible colocar faceplates y tipos de datos de usuario HMI directamente en el editor. A continuación, la instancia se conecta con la versión de tipo respectivo. Si el tipo contiene elementos dependientes, tales como tipos de datos PLC referenciados en un bloque, estos también se crearán como instancia en la ubicación adecuada. Solo es posible asignar una versión del mismo tipo a un dispositivo.

En caso necesario pueden crearse instancias de varios tipos a la vez.

Requisitos

- Se visualiza la Task Card "Librerías".
- Las versiones deseadas están habilitadas.
- En el proyecto ya hay un dispositivo que soporta la clase de los tipos deseados.
- El dispositivo aún no tiene asignada otra instancia de los mismos tipos.

Procedimiento

Para generar instancias a partir de versiones de tipos, proceda del siguiente modo:

1. En la librería del proyecto navegue hasta las versiones de los tipos que desea utilizar.
2. Seleccione las versiones a partir de las cuales desea crear una instancia.
3. Arrastre las versiones de tipos mediante Drag & Drop desde la librería del proyecto hasta la ubicación.
   Alternativa: para utilizar automáticamente la versión más reciente, arrastre los tipos propiamente dichos mediante Drag & Drop desde la librería del proyecto hasta la ubicación.

Resultado

Se generan instancias a partir de los tipos y sus elementos dependientes y se insertan en la ubicación deseada. Las instancias están enlazadas a la correspondiente versión del tipo en la librería del proyecto.

Consulte también

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Uso de plantillas maestras (Página 457)

Crear una versión de test de un tipo

Antes de habilitar un tipo para la utilización productiva es necesario probar el tipo dentro de un proyecto y en una instalación de automatización. El test se lleva a cabo en un entorno de test determinado. Este entorno de test puede ser una CPU, por ejemplo.

Para el test es necesario crear una versión en el estado "en test". Es posible crear una versión en el estado "en test" para todas las clases de tipos, excepto para faceplates y tipos de datos de usuario HMI. En cambio, para faceplates y tipos de datos de usuario HMI es posible crear versiones en el estado "en proceso".

Existen dos posibilidades para crear una versión de test de un tipo y definir el entorno de test:

- En la Task Card "Librerías" o en la vista de librería
  La versión nueva se genera en el estado "en test" en la Task Card "Librerías" o en la vista de librería. La versión nueva se genera directamente a partir del tipo o de una versión determinada del tipo.
- En una instancia del árbol del proyecto
  También es posible crear la versión de test directamente en la instancia del árbol del proyecto. Puesto que la instancia siempre se utiliza en una versión determinada del proyecto, se genera una versión nueva del tipo a partir de la versión utilizada en la instancia.

También puede crear simultáneamente versiones de test de varios tipos.

Para una versión en el estado "en test" rigen las siguientes reglas:

- Únicamente es posible poner una sola versión de cada tipo en el estado "en test".
- Una versión en test solo puede estar enlazada con una sola instancia en el proyecto. Por ello, no es posible copiar una instancia en el portapapeles, duplicarla o crear otro tipo a partir de la instancia mientras esta tenga el estado "en test".

Requisitos

- Dentro del proyecto existe como mínimo una instancia del tipo en una versión cualquiera.
- Si desea crear una versión nueva a partir de una versión determinada del tipo, la instancia debe utilizarse en esta versión en el proyecto.

Procedimiento

Para crear una versión de test nueva de un tipo o la versión de un tipo, proceda del siguiente modo:

1. Seleccione el tipo, una versión del tipo o la instancia.
   Si crea la versión de test directamente en la instancia, es posible seleccionar varios elementos o carpetas utilizando la selección múltiple. Como mediante la instancia seleccionada ya está definido el entorno de test, es posible omitir los pasos 3 y 4.
2. Haga clic con el botón derecho del ratón en el elemento seleccionado y elija el comando "Editar tipo" del menú contextual.
   Si se ha iniciado la edición en la Task Card "Librerías" o en la vista de librería, se abrirá el cuadro de diálogo "Editar tipo". Si se ha iniciado la edición en la instancia en el árbol del proyecto, la instancia de test se abrirá inmediatamente para su edición en la vista de librería.
3. En la lista, seleccione una instancia del tipo en el proyecto.
   Si se ha iniciado la edición en el tipo, rige lo siguiente:
   – La ubicación en la que se utiliza la instancia (p. ej. la CPU) sirve para editar seguidamente el tipo como entorno de test.
   – Mediante la selección de la instancia de test se especifica también la versión que se editará.

   Para la edición de una versión determinada rige lo siguiente:
   Si el punto de partida es una versión determinada, solo es posible elegir en la lista instancias que se utilicen en la misma versión.
4. Confirme con "Aceptar".

Resultado

Se crea una versión nueva del tipo. La versión nueva tiene el estado "en test" y se identifica correspondientemente en la interfaz de usuario.

Consulte también

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Editar una versión de test de un tipo

Si sigue editando una versión en estado "en test", no se creará una versión nueva. La edición de la versión de test puede iniciarse en la instancia del árbol del proyecto, en la Task Card "Librerías" o en la vista de librería.

Nota

Borrar y cambiar el nombre de los parámetros de interfaz

Es posible agregar nuevos parámetros. No obstante, si cambia el nombre de parámetros ya existentes o los borra, estos dejan de recibir valores al llamar el bloque.

Procedimiento

Para editar la versión de test de un tipo, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la versión de test o en la instancia.
2. Elija el comando "Editar tipo" del menú contextual.
   La instancia de test se abre en la vista de librería y puede editarse.

Consulte también

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Comprobar la coherencia de una versión (Página 468)

Rechazar versiones (Página 468)

Habilitar versiones (Página 469)

Crear una versión en proceso de un tipo

Si se desea editar un tipo con faceplates o tipos de datos de usuario HMI, es necesario crear una versión nueva del tipo en estado "en proceso". La edición de la versión nueva se lleva a cabo en la vista de librería. Para comprobar la compatibilidad de los cambios se comprueba automáticamente la coherencia del tipo antes de la habilitación.

Requisitos

La librería del proyecto está abierta en la Task Card "Librerías" o en la vista de librería.

Procedimiento

Para crear una versión nueva de un tipo en proceso, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el tipo o en la versión del tipo.
2. Elija el comando "Editar tipo" del menú contextual.
   Se creará una versión nueva "en proceso" y se abrirá en la vista de librería para su edición.

Consulte también

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Comprobar la coherencia de una versión

Al editar una versión de tipo es posible que se produzca por error un estado incoherente. Para detectar a tiempo errores en el proceso de desarrollo, compruebe con frecuencia la coherencia. De todas formas, esta prueba de coherencia siempre se lleva a cabo automáticamente tan pronto se habilita una versión.

A continuación se explica cómo iniciar manualmente la prueba de coherencia de la versión de un tipo.

Requisitos

- La librería del proyecto está abierta en la Task Card "Librerías" o en la vista de librería.
- La versión se encuentra en estado "en proceso" o "en test".

Procedimiento

Para comprobar la coherencia de la versión de un tipo, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la versión cuya coherencia desea comprobar.
2. Elija el comando "Comprobar coherencia" del menú contextual.
   Se comprueba la coherencia. Se obtiene un aviso con el resultado de la prueba de coherencia.

Consulte también

Habilitar versiones (Página 469)

Rechazar versiones (Página 468)

Editar una versión de test de un tipo (Página 466)

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Rechazar versiones

Es posible rechazar las versiones de un tipo que tengan el estado "en test" o "en proceso" si ya no se necesitan. También existe la posibilidad de seleccionar varios tipos o carpetas y rechazar todas las versiones de test o procesamiento que haya dentro. Todos los usos de las versiones borradas se restablecen al último estado habilitado.

Requisitos

- La versión que desea descartar tiene el estado "en test" o "en proceso".
- Se encuentra en la vista de librería o está abierta la Task Card "Librerías".

Procedimiento

Para rechazar una versión, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la versión que desea descartar.
2. Elija el comando "Rechazar cambios y borrar versión" del menú contextual.
   La versión se borra.

Alternativa:

1. Mientras haya una versión abierta para su edición, haga clic en el botón "Rechazar cambios y borrar versión" de la barra de herramientas.
   La versión se borra.

Consulte también

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Comprobar la coherencia de una versión (Página 468)

Rechazar todas las versiones contenidas en una carpeta

Puede rechazar a la vez todas las versiones en el estado "en test" o "en proceso" contenidas en una carpeta. Todos los usos de la versión borrada se restablecen al último estado habilitado.

Requisitos

Se encuentra en la vista de librería o está abierta la Task Card "Librerías".

Rechazar la versión de un solo tipo

Para rechazar todas las versiones contenidas en una carpeta, haga lo siguiente:

1. Haga clic en la carpeta con el botón derecho del ratón.
2. Seleccione el comando "Rechazar todo" del menú contextual.
   Se borrarán todas las versiones "en test" o "en proceso".

Habilitar versiones

Una vez finalizada la edición de una versión de tipo, habilite la versión para la utilización productiva. Al realizar la habilitación se asigna un número a la versión. También es posible habilitar varias versiones a la vez utilizando la selección múltiple.

Requisitos

- La Task Card "Librerías" está abierta o se encuentra en la vista de librería.
- Las versiones que desea habilitar tienen el estado "en test" o "en proceso".
- Las versiones tienen un estado coherente.
  Se realiza una comprobación de coherencia en cuanto se inicia la habilitación. Si ocurren errores que impiden la habilitación durante la prueba de coherencia, aparecerá un aviso con recomendaciones para eliminar el error.

Procedimiento

Para habilitar versiones de tipos, proceda del siguiente modo:

1. Seleccione las versiones que desea habilitar.
2. Haga clic en la selección con el botón derecho del ratón.
3. Elija el comando "Habilitar versión" del menú contextual.
   Se abre el cuadro de diálogo "Habilitar versión de tipo".
4. Modifique las propiedades de la versión en caso necesario:
   - Introduzca un nombre para el tipo en el campo "Nombre". Si ha seleccionado varias versiones para la habilitación, el campo "Nombre" no puede modificarse.
   - En el campo "Versión" especifique un número de versión principal y uno intermedio para la versión que se va a habilitar. Si ha seleccionado varias versiones para la habilitación, el campo "Versión" no puede modificarse y para la habilitación se utiliza el último número de versión en cada caso.
   - En el campo "Autor" introduzca el autor de la versión que se va a habilitar.
   - En el campo "Comentario" introduzca un comentario acerca de la versión que se va a habilitar.
5. Opcionalmente: Active la casilla de verificación "Borrar de la librería las versiones de tipo no utilizadas" para borrar de la librería todas las versiones que no están enlazadas con una instancia del proyecto. Las versiones que dependen de otros tipos o plantillas maestras no se borran.
6. Confirme con "Aceptar".

Alternativa:

1. Mientras esté abierta una versión para editar, haga clic en el botón "Habilitar versión" de la barra de herramientas.
2. Continúe con los pasos 3 a 5 de la descripción que antecede.

Resultado

Las versiones seleccionadas se habilitan.

Las propiedades se aplican a los tipos en cuestión, las versiones que se van a habilitar y todas las versiones posteriores. Los cambios no afectan a las versiones ya habilitadas.

Dado el caso, todas las instancias con la misma versión de partida se actualizarán a la versión más reciente y se borrarán las versiones del tipo que no se utilizan.

Consulte también

Habilitar todas las versiones dentro de una carpeta (Página 471)

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Comprobar la coherencia de una versión (Página 468)

Asignar una versión (Página 477)

Agregar tipos a una librería global (Página 474)

Actualizar un proyecto a las últimas versiones (Página 472)

Habilitar todas las versiones dentro de una carpeta

Una vez finalizada la edición de todos los tipos de una carpeta es posible habilitar todas las versiones a la vez.

Requisitos

- Se encuentra en la Task Card "Librerías" o en la vista de librería.
- La carpeta contiene versiones en estado "en test" o "en proceso".
- Todas las versiones "en test" o "en proceso" tienen un estado coherente.
  Se realiza una comprobación de coherencia en cuanto se inicia la habilitación. Si ocurren errores que impiden la habilitación durante la comprobación de coherencia, aparecerá un aviso con recomendaciones para eliminar el error.

Procedimiento

Para habilitar todas las versiones de tipos de una carpeta, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la carpeta deseada.
2. Elija el comando "Habilitar todo" del menú contextual.
   Se abre el cuadro de diálogo "Habilitar versión de tipo".
3. Modifique las propiedades de la versión en caso necesario:
   - En el campo "Autor" introduzca el autor de las versiones que se van a habilitar.
   - En el campo "Comentario" introduzca un comentario acerca de las versiones que se van a habilitar.
4. Active la casilla de verificación "Borrar de la librería las versiones de tipo no utilizadas" para borrar de la librería todas las versiones que no están enlazadas con una instancia del proyecto. Las versiones que dependen de otros tipos o plantillas maestras no se borran.
5. Confirme con "Aceptar".

Resultado

Se habilitan todas las versiones de tipos "en test" o "en proceso" dentro de la carpeta seleccionada.

Las propiedades se aplican a las versiones que se van a habilitar y todas las versiones posteriores. Los cambios no afectan a las versiones ya habilitadas.

Dado el caso se borran las versiones de tipos que no se utilizan en el proyecto.

Consulte también

Habilitar versiones (Página 469)

Actualizar un proyecto a las últimas versiones

Después de actualizar varios tipos en la librería del proyecto, es necesario actualizar todas las instancias del proyecto a la versión más reciente de los tipos de la librería del proyecto. Si no desea aplicar los cambios a todo el proyecto, limite la actualización a solo algunos dispositivos del proyecto.

Los siguientes elementos pueden seleccionarse como origen de la actualización:

- toda la librería del proyecto
- distintas carpetas dentro de la librería del proyecto
- distintos tipos
  Está permitido seleccionar varios tipos.

Requisitos

Se encuentra en la Task Card "Librerías" o en la vista de librería.

Procedimiento

Para actualizar instancias en un proyecto con los contenidos de la librería del proyecto, proceda del siguiente modo:

1. Seleccione toda la librería del proyecto o algunos elementos de ella.
2. Haga clic en los elementos deseados con el botón derecho del ratón y elija el comando "Actualizar > Proyecto" del menú contextual.
   Se abre el cuadro de diálogo "Actualizar proyecto".
3. Seleccione todo el proyecto o algunos dispositivos para la actualización.

4. Elija las opciones para el proceso de actualización:
   – La casilla de verificación "Actualizar todas las instancias de los tipos seleccionados" siempre está activa para este proceso.
   – Active la casilla de verificación "Borrar de la librería del proyecto todas las versiones no utilizadas" para borrar de la librería del proyecto todas las versiones anteriores de los tipos actualizados.
5. Confirme con "Aceptar".
   Se realiza la actualización.

Resultado

En el proyecto se han efectuado las modificaciones siguientes:

- Dado el caso se han borrado de la librería del proyecto todas las versiones anteriores.
- Todas las instancias dentro de los dispositivos seleccionados se han actualizado a la versión más reciente del tipo enlazado.
- En el árbol del proyecto, bajo "Datos comunes", encontrará un informe del proceso de actualización.

Consulte también

Actualizar un proyecto a la versión de tipo más reciente (Página 476)

Uso de informes (Página 339)

Actualizar una librería con los contenidos de otra librería (Página 481)

Principios básicos de los tipos (Página 458)

Principios básicos de las librerías (Página 427)

Estado de las versiones de tipo (Página 460)

Deshacer la conexión entre la instancia y el tipo

Las instancias de los tipos siempre están enlazadas con la versión del tipo correspondiente y no pueden editarse como un objeto común. Al editar una instancia, se crea automáticamente una versión nueva del tipo en la librería y, de este modo, los cambios influyen en todo el proyecto.

Si se deshace la conexión de la instancia con su tipo, a continuación el objeto se edita como un objeto común en el árbol del proyecto.

Requisitos

La instancia no debe tener el estado "en test".

Procedimiento

Para deshacer la conexión entre las instancias y sus versiones de tipos, proceda del siguiente modo:

1. Seleccione una o varias instancias en el árbol del proyecto.
2. Haga clic en la selección con el botón derecho del ratón y elija el comando "Desconectar del tipo" del menú contextual.
3. Se deshace la conexión con las versiones de tipos correspondientes.

Consulte también

Principios básicos de los tipos (Página 458)

Principios básicos de las librerías (Página 427)

Estado de las versiones de tipo (Página 460)

7.10.8.6 Trabajar con tipos en las librerías globales

Agregar tipos a una librería global

Las librerías globales sirven de recurso central para trabajar en varios proyectos. Por ello, la edición directa de tipos está reservada a los tipos de la librería del proyecto. Para trabajar en los tipos es necesario utilizar la librería del proyecto. Una vez finalizada la edición de un tipo en la librería del proyecto, es posible agregar el tipo a la librería global. El proceso de agregar tipos desde la librería del proyecto es igual al proceso de copia habitual desde la librería del proyecto.

Requisitos

- La Task Card "Librerías" o la vista de librería está abierta.
- La librería global a la que desea agregar tipos está abierta con derechos de escritura.

Procedimiento

Para agregar tipos a una librería global, proceda del siguiente modo:

1. Abra la carpeta deseada de la librería global en la Task Card "Librerías" o en la vista de librería.
2. Arrastre uno o varios tipos mediante Drag & Drop desde la librería del proyecto hasta la carpeta "Tipos" o cualquier subcarpeta de la librería global.

Alternativa:

1. Copie los tipos deseados de la librería del proyecto en el portapapeles.
2. Abra la librería global deseada en la paleta "Librería global" de la Task Card "Librerías".

3. Haga clic con el botón derecho del ratón en la carpeta "Tipos" o en cualquier subcarpeta de "Tipos".
4. Elija el comando "Pegar" del menú contextual.

Resultado

Los tipos se insertan en la librería global. Los tipos dependientes como, por ejemplo, tipos de datos de usuario HMI o variables, también se copian en la librería global, siempre y cuando aún no existan en ella. Así se garantiza que todos los elementos necesarios para crear una instancia se encuentren en la librería global.

Si un tipo ya está presente en la librería global, el proceso descrito corresponde a una actualización de la librería global. En este caso se añadirán a la librería global las versiones actuales habilitadas del tipo.

Consulte también

Principios básicos de los tipos (Página 458)

Habilitar versiones (Página 469)

Asignar una versión (Página 477)

Actualizar un proyecto a la versión de tipo más reciente (Página 476)

Principios básicos de las librerías (Página 427)

Uso de tipos

Para utilizar la versión de un tipo desde la librería global debe crearse una instancia de la versión en una ubicación adecuada del árbol del proyecto. También es posible colocar faceplates y tipos de datos de usuario HMI directamente en el editor.

Las instancias de los tipos de una librería global no están enlazadas con el tipo de la librería global en el proyecto. Por el contrario, al insertar una instancia al proyecto se genera una copia del tipo y sus elementos dependientes en la librería del proyecto. Los elementos dependientes pueden ser tipos de datos PLC que se referencian en un bloque, por ejemplo. La copia del tipo y de los elementos dependientes en la librería del proyecto contiene en cada caso la versión que se enlaza con la instancia. Si el tipo o un elemento dependiente ya existe en la librería del proyecto, solo se complementará la versión faltante del tipo en la librería del proyecto.

La instancia está enlazada con la copia del tipo en la librería del proyecto. Solo es posible asignar un tipo a un dispositivo una vez, independientemente de la versión.

Requisitos

- Se visualiza la Task Card "Librerías".
- En el proyecto ya hay un dispositivo que soporta la clase del tipo deseado.
- El dispositivo aún no tiene asignada otra instancia del mismo tipo.

Procedimiento

Para generar una instancia a partir de la versión de un tipo y luego utilizarla, proceda del siguiente modo:

1. En la librería global navegue hasta la versión del tipo que desea utilizar.
2. Arrastre la versión deseada del tipo hasta la ubicación mediante Drag & Drop.
   Alternativa: para utilizar automáticamente la versión más reciente, arrastre el tipo propiamente dicho mediante Drag & Drop hasta la ubicación.

Resultado

Los elementos faltantes se complementarán en la librería del proyecto. Se genera una instancia a partir del tipo y sus elementos dependientes y se inserta en la ubicación deseada. Las instancias están enlazadas a la correspondiente versión del tipo en la librería del proyecto.

Consulte también

Principios básicos de los tipos (Página 458)

Actualizar un proyecto a la versión de tipo más reciente (Página 476)

Principios básicos de las librerías (Página 427)

Utilizar la vista de elementos (Página 431)

Actualizar un proyecto a la versión de tipo más reciente

Las librerías globales suelen editarse de forma centralizada especialmente en grandes empresas con muchos proyectos de automatización. Una vez finalizada una versión nueva, las librerías globales actualizadas se ponen a disposición de los distintos proyectos. Si se obtiene una versión más reciente de una librería global, se reemplazan las instancias obsoletas del proyecto por la versión más reciente. Si no desea aplicar los cambios a todo el proyecto, limite la actualización a solo algunos dispositivos del proyecto.

Durante la actualización del proyecto o de los distintos dispositivos también se actualiza la librería del proyecto con las versiones nuevas de los tipos en la librería global.

Los siguientes elementos pueden seleccionarse como origen de la actualización:

- una librería global
- diferentes carpetas dentro de una librería global
- distintos tipos
  Está permitido seleccionar varios tipos.

Requisitos

- Se encuentra en la Task Card "Librerías" o en la vista de librería.
- La librería global actualizada está abierta.

Procedimiento

Para actualizar instancias en un proyecto con los contenidos de una librería global, proceda del siguiente modo:

1. Seleccione la librería global actualizada o algunos elementos de ella.
2. Haga clic en la librería global o en los elementos deseados con el botón derecho del ratón y elija el comando "Actualizar > Proyecto" del menú contextual.
   Se abre el cuadro de diálogo "Actualizar proyecto".
3. Seleccione todo el proyecto o algunos dispositivos para la actualización.
4. Elija las opciones para el proceso de actualización:
   – La casilla de verificación "Actualizar todas las instancias de los tipos seleccionados" siempre está activa para este proceso.
   – Active la casilla de verificación "Borrar de la librería del proyecto todas las versiones no utilizadas" para borrar de la librería del proyecto todas las versiones anteriores de los tipos actualizados.
5. Confirme con "Aceptar".
   Se realiza la actualización.

Resultado

En el proyecto se han efectuado las modificaciones siguientes:

- La versión más reciente de los tipos seleccionados se encuentra en la librería del proyecto. En caso necesario se han borrado todas las versiones anteriores.
- Todas las instancias dentro de los dispositivos seleccionados se han actualizado a la versión más reciente del tipo enlazado.
- En el árbol del proyecto, bajo "Datos comunes", encontrará un informe del proceso de actualización.

Consulte también

Actualizar un proyecto a las últimas versiones (Página 472)

Uso de informes (Página 339)

Actualizar una librería con los contenidos de otra librería (Página 481)

Principios básicos de los tipos (Página 458)

Principios básicos de las librerías (Página 427)

Agregar tipos a una librería global (Página 474)

7.10.8.7 Asignar una versión

Una librería es más clara si los tipos con contenido relacionado tienen el mismo número de versión. El número de versión idéntico refleja el progreso. Tras concluir el trabajo con varios tipos relacionados, puede asignarles el mismo número de versión.

Para asignar una versión común a varios tipos están disponibles las siguientes posibilidades de selección:

- toda la librería del proyecto o una librería global completa
- una o varias carpetas de una librería
- uno o varios tipos

Requisitos

- La Task Card "Librerías" o la vista de librería está abierta.
- La selección no puede contener tipos en estado "en test" o "en proceso".

Procedimiento

Para asignar la misma versión a varios tipos, proceda del siguiente modo:

1. Seleccione los tipos a los que desea asignar una versión común.
2. Elija el comando "Asignar versión" del menú contextual.
   Se abre el cuadro de diálogo "Asignar versión".
3. Modifique las propiedades de la versión en caso necesario:
   – Defina el número de versión nuevo en el campo "Versión". El número de versión debe ser mayor que el número de versión más alto de todos los tipos seleccionados.
   – En el campo "Autor" introduzca el nombre del responsable de la versión que se va a habilitar.
   – En el campo "Comentario" introduzca un comentario acerca de la versión que se va a habilitar.
4. Si ha iniciado la operación desde la librería del proyecto, elija las opciones para la habilitación. Dentro de una librería global no pueden elegirse las opciones siguientes.
   – Active la casilla de verificación "Actualizar todas las instancias" para actualizar a la versión nueva todas las instancias del proyecto. Los tipos utilizados en plantillas maestras no se actualizarán.
   – Active la casilla de verificación "Borrar de la librería las versiones no utilizadas" para borrar de la librería del proyecto todas las versiones anteriores de los tipos. Las versiones solo se borran si no tienen asignadas instancias en el proyecto y no dependen de otros tipos.
5. Confirme con "Aceptar".

Resultado

Las versiones de tipo seleccionadas se modifican del siguiente modo:

- Se crea una versión nueva de todos los tipos seleccionados con el número de versión especificado.
- Las propiedades se aplican a todos los tipos seleccionados, a la versión nueva y a todas las versiones posteriores. Los cambios no afectan a las versiones anteriores. Si no se cambian las propiedades, estas adoptarán la última versión habilitada de cada tipo.

- Dado el caso, todas las instancias se actualizarán a la versión más reciente y se borrarán de la librería del proyecto las versiones de todos los tipos que no se utilizan.
- El número de versión interno de tipos dependientes se incrementa al siguiente número libre, siempre que los tipos dependientes no estuvieran incluidos en la selección. Si también se había seleccionado un tipo dependiente, se asignará el número de versión determinado por el usuario.

Se crea un informe de los cambios. Si se han versionado tipos en la librería del proyecto, encontrará el informe en "Datos comunes > Informes" en el árbol del proyecto. Si se han versionado tipos en una librería global, encontrará el informe en la carpeta "Datos comunes > Informes" de la librería global.

Consulte también

Principios básicos de los tipos (Página 458)

Estado de las versiones de tipo (Página 460)

Principios básicos de las librerías (Página 427)

Habilitar versiones (Página 469)

Agregar tipos a una librería global (Página 474)

Uso de informes (Página 339)

Mostrar los informes de las librerías globales (Página 446)

7.10.9 Editar elementos de librería

Dentro de la Task Card "Librerías" o en la vista de librería es posible cortar, copiar, pegar, desplazar, cambiar de nombre o borrar tipos, plantillas maestras y carpetas del modo habitual. Las librerías globales deben estar abiertas con derechos de escritura para realizar los procesos mencionados.

Nota

Ayuda de usuario para tipos y plantillas maestras

La ayuda de usuario no se ve afectada por las operaciones realizadas en la librería. Si se mueve una plantilla maestra o un tipo a otra ubicación, debe moverse también la ayuda de usuario correspondiente a la ubicación en cuestión en el sistema de archivos.

Encontrará más información sobre el uso de la ayuda de usuario en el capítulo "Convenciones para crear la ayuda de usuario".

Copiar tipos

Si se copia un tipo en el portapapeles rigen las reglas siguientes:

- Los tipos siempre se copian en el portapapeles con todas las versiones correspondientes. Sin embargo, solo se copian versiones que ya están habilitadas.
- Los tipos siempre se copian en el portapapeles con todos los elementos dependientes.
- Las plantillas maestras siempre se copian en el portapapeles junto con las versiones de tipos utilizadas en ellas.

Copiar y pegar versiones de tipos

Si se copian versiones de tipos y se pegan en otra librería, los tipos deben existir ya en la librería de destino.

Cortar elementos

Los elementos de librería cortados solo se pueden insertar en la misma librería. Las plantillas maestras solo se pueden insertar en la carpeta "Plantillas maestras" o en cualquier subcarpeta de la misma y los tipos solo se pueden insertar en la carpeta "Tipos" o en cualquier subcarpeta de la misma.

Pegar tipos

El proceso de pegar tipos en otra librería es igual al de una actualización de la librería de destino.

Si se ha pegado un tipo en otra librería, rigen las reglas siguientes:

- Un tipo siempre se pega con todas sus versiones.
- Si el tipo ya existe en la librería de destino, todas las versiones más actuales que las existentes se complementarán con en el tipo correspondiente en la librería de destino.
- Si en la librería de destino ya existe una versión en estado habilitado, no se insertará nuevamente la misma versión.
- Si la misma versión ya existe en la librería de destino en estado en test o edición, se reemplazará por la versión habilitada.
- Si un tipo necesita otros tipos, estos se insertan también en la ubicación correspondiente.

Pegar plantillas maestras

Al pegar plantillas maestras también se insertarán todas las versiones de tipos utilizadas allí. Si los tipos correspondientes ya existen en la librería, solo se complementarán las versiones faltantes de los diferentes tipos. Si aún no existe uno de los tipos utilizados, este se insertará en el nivel más alto de la librería. El tipo contiene la versión de tipo que se utilizaba en la plantilla maestra.

Desplazar elementos

Si desplaza un elemento de una librería a otra, el elemento no se desplaza realmente, sino que se copia. Rigen las mismas reglas expuestas en "Pegar tipos" y "Pegar plantillas maestras".

Borrar tipos y versiones de tipo

Al borrar tipos o versiones de tipo se debe tener en cuenta lo siguiente:

- Un tipo o una versión de tipo solo puede borrarse si no existen dependencias con otros tipos.
- Al borrar un tipo también se borran todas las versiones del tipo.
- Si se borran todas versiones de un tipo, también se borrará el tipo.
- Si se borra una versión que contiene instancias en el proyecto, también se borrarán las instancias del proyecto.
- Si se borra un tipo que está guardado al mismo tiempo como plantilla maestra, se borrará también la plantilla maestra.

Borrar instancias

Si se borra una instancia que tiene dependencias con otras instancias, se restablecerá durante la próxima compilación. La instancia se enlaza nuevamente con la versión de tipo original. Así se restablece otra vez la coherencia del proyecto.

Consulte también

Principios básicos de las librerías (Página 427)

Deshacer la conexión entre la instancia y el tipo (Página 473)

Actualizar una librería con los contenidos de otra librería (Página 481)

7.10.10 Actualizar una librería con los contenidos de otra librería

Una librería existente puede actualizarse con los contenidos de otra librería. Para actualizar librerías existen las siguientes posibilidades:

- actualización de una librería global con tipos de otra librería global o librería del proyecto
- actualización de la librería del proyecto con tipos de una librería global

Los siguientes elementos pueden seleccionarse como origen de la actualización:

- una librería total
- diferentes carpetas dentro de una librería
- distintos tipos
  Está permitido seleccionar varios tipos.

Durante la actualización se complementan tipos ya existentes con las versiones más recientes en la librería de destino. Los tipos que aún no existen en la librería de destino se copian en la misma con todas sus versiones.

Nota

Ayuda de usuario al copiar tipos

La ayuda de usuario no se copia en otra librería junto con un tipo. El propio usuario debe copiar la ayuda de usuario para tipos en el directorio adecuado.

Encontrará más información sobre el uso de la ayuda de usuario en el capítulo "Convenciones para crear la ayuda de usuario".

Requisitos

Si desea actualizar una librería global, tiene que estar abierta con derechos de escritura.

Procedimiento

Para actualizar una librería con contenidos de otra librería, proceda del siguiente modo:

1. Seleccione una librería o diversos elementos de esta como origen para la actualización.
2. Haga clic en el origen con el botón derecho del ratón y elija el comando "Actualizar > Librería" del menú contextual.
   Se abre el cuadro de diálogo "Actualizar librería".
3. Seleccione qué clase de librería desea actualizar:
   - Elija "Actualizar librería del proyecto" para actualizar la librería del proyecto con tipos de una librería global.
   - Elija "Actualizar librería global" si desea actualizar una librería global.
4. Opcionalmente: seleccione en la lista desplegable la librería global que desea actualizar si se trata de una librería global.
5. Seleccione las opciones para la actualización:
   - La opción "Actualizar todas las instancias" siempre está desactivada durante este proceso.
   - Active la casilla de verificación "Borrar de la librería del proyecto las versiones no utilizadas" si desea borrar de la librería del proyecto todas las versiones del tipo que no están enlazadas con una instancia del proyecto y que no tienen dependencias con otros tipos. No es posible seleccionar la opción durante la actualización de una librería global, ya que los tipos de una librería global nunca poseen una ubicación en el proyecto.
6. Confirme con "Aceptar".
   Se realiza la actualización.

Resultado

En la librería de destino se han realizado las modificaciones siguientes:

- Los tipos aún no existentes en la librería de destino se han complementado con todas sus versiones. Los tipos ya existentes en la librería de destino se han complementado con las versiones más recientes. Si en la librería de destino ya existía una versión actual de un tipo, la versión actual se ha copiado igualmente desde la librería de origen y se le ha asignado automáticamente un número de versión más reciente.
- En caso necesario, se han borrado todas las versiones de los tipos de la librería del proyecto si no se utilizan en ninguna instancia del proyecto.
- Se ha creado un informe para el proceso de actualización que contiene todas las modificaciones realizadas en la librería de destino.
  Si se ha actualizado la librería del proyecto, encontrará el informe bajo "Datos comunes > Informes" en el árbol del proyecto.
  Si se ha actualizado una librería global, encontrará el informe de la librería global en la carpeta "Datos comunes > Informes" del árbol del proyecto.

Consulte también

Uso de informes (Página 339)

Actualizar un proyecto a las últimas versiones (Página 472)

Actualizar un proyecto a la versión de tipo más reciente (Página 476)

Mostrar los informes de las librerías globales (Página 446)

Principios básicos de las librerías (Página 427)

7.10.11 Armonizar nombres y estructura de rutas

Es posible armonizar el proyecto con una librería. Esto permite corregir los puntos siguientes:

- Nombres de las instancias:
  Durante la fase de desarrollo de una librería pueden surgir instancias cuyos nombres se han corregido automáticamente con una extensión ("_1", "_2", etc.). La extensión se crea para evitar nombres duplicados en el proyecto. En la armonización, las instancias reciben de nuevo el nombre del tipo al que corresponden.
- Estructura de rutas:
  Al desarrollar en paralelo o al copiar instancias dependientes es posible que se pierda la estructura original de rutas. Este hecho perjudica la claridad del proyecto. En la armonización, la estructura de rutas dentro del proyecto se adapta a la estructura de rutas de la librería.

Procedimiento

Para armonizar los nombres y la estructura de rutas, proceda del siguiente modo:

1. Abra la administración de librerías.
2. Haga clic en "Armonizar proyecto" de la barra de herramientas.
   Se abre el cuadro de diálogo "Armonizar proyecto".
3. Seleccione el dispositivo con el que desea armonizar la librería.
4. Si desea restablecer la estructura de rutas, active la casilla de verificación "Armonizar rutas entre proyecto y librería".
5. Si desea corregir los nombres, active la casilla de verificación "Armonizar nombres entre proyecto y librería".
6. Confirme las entradas con "Aceptar".

Resultado

Los nombres y la estructura de rutas del proyecto se armonizan con la librería dependiendo de los ajustes realizados.

Los cambios realizados en el proyecto se registran en un informe. Encontrará el informe en "Datos comunes > Informes" del árbol del proyecto.

Consulte también

Principios básicos de las librerías (Página 427)

Sinopsis de la vista de librerías (Página 432)

Sinopsis de la administración de librerías (Página 435)

Uso de informes (Página 339)

7.10.12 Limpiar librería

Es posible limpiar la librería del proyecto o librerías globales para eliminar tipos o versiones que no están enlazados con ninguna instancia del proyecto. De este modo se aumenta la claridad dentro de las librerías y se reduce el tamaño de la librería.

Limpiar librería del proyecto

Para limpiar la librería del proyecto, proceda del siguiente modo:

1. Abra la administración de librerías.
2. Haga clic en "Limpiar librería" de la barra de herramientas.
   Se abre el cuadro de diálogo "Limpiar librería del proyecto".

3. Seleccione los tipos o versiones que se borrarán:
   – Para conservar la versión con el número más alto aunque esta no tenga ninguna instancia, seleccione la opción "Borrar versiones de tipo antiguas y conservar la versión de tipo más reciente".
   – Para borrar el tipo completo si no hay ninguna versión enlazada con una instancia, seleccione la opción "Borrar tipos completos".
4. Confirme la entrada realizada con "Aceptar".
   De la librería del proyecto se eliminarán las versiones de tipo no utilizadas o tipos completos, dependiendo de la elección efectuada.
   Los cambios realizados se registran en un informe. Encontrará el informe en "Datos comunes > Informes" del árbol del proyecto.

Limpiar librería global

Para limpiar una librería global, proceda del siguiente modo:

1. Abra la administración de librerías.
2. Haga clic en "Limpiar librería" de la barra de herramientas.
   Se abre el cuadro de diálogo "Limpiar librería global".
3. Haga clic en "Continuar".
   Se borran las versiones de tipo no utilizadas. Se mantiene siempre la versión más reciente de un tipo.
   Los cambios realizados se registran en un informe. Encontrará el informe en la carpeta "Datos comunes > Informes" de la librería global.

Consulte también

Principios básicos de las librerías (Página 427)

Sinopsis de la vista de librerías (Página 432)

Sinopsis de la administración de librerías (Página 435)

Uso de informes (Página 339)

Mostrar los informes de las librerías globales (Página 446)

7.10.13 Comparación de elementos de librerías

Introducción

Se pueden comparar dispositivos de librerías con dispositivos del proyecto actual, con dispositivos de la misma librería o de otras distintas o de proyectos de referencia. No obstante, recuerde que los proyectos de referencia están protegidos contra escritura. También es posible comparar instancias de un dispositivo con su versión de tipo en una librería. En la comparación con tipos no están disponibles todas las acciones Por ejemplo, no es posible que una versión de tipo anterior de la librería sobrescriba una instancia de una versión más reciente.

Al comparar elementos de librerías se puede cambiar en todo momento entre la comparación automática y la manual.

Procedimiento

Para comparar elementos de librerías con los datos de dispositivo de un proyecto, proceda del siguiente modo:

1. En el árbol del proyecto seleccione el dispositivo cuyos datos desea comparar con un elemento de librería y que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Abra la Task Card "Librerías".
4. Seleccione el elemento de librería que desea comparar con los datos del dispositivo.
5. Arrastre el elemento de librería a la superficie de arrastre derecha del editor de comparación.
   Los símbolos del área de estado y acción indican el estado de los objetos. Si se selecciona un objeto, sus propiedades y las del objeto correspondiente del dispositivo asignado se muestran sinópticamente en la comparación de propiedades.
   En todo momento es posible arrastrar a las superficies de arrastre otros dispositivos desde el proyecto actual, desde una librería o desde un proyecto de referencia, para iniciar una nueva comparación. No importa qué dispositivo se arrastre a qué área de arrastre.

Consulte también

Utilizar el editor de comparación (Página 375)

Realizar una comparación offline/offline (Página 374)

Uso de la vista de librerías (Página 432)

Principios básicos de las librerías (Página 427)

Sinopsis de la Task Card "Librerías". (Página 429)

Sinopsis de la vista de librerías (Página 432)

Utilizar la vista de elementos (Página 431)

Utilizar librerías globales (Página 440)

7.11 Uso de referencias cruzadas

7.11.1 Uso de referencias cruzadas

Introducción a las referencias cruzadas

La lista de referencias cruzadas ofrece una visión de conjunto de la utilización de objetos en el proyecto. En esta lista puede consultar qué objetos dependen unos de otros y dónde se encuentran los distintos objetos. Por tanto, las referencias cruzadas forman parte de la documentación del proyecto.

Además, permiten saltar directamente a la ubicación de un objeto.

Los objetos que se visualizan y que pueden localizarse en la lista de referencias cruzadas dependen de los productos instalados.

Consulte también

Visualizar referencias cruzadas de una instancia (Página 439)

7.12 Simulación de dispositivos

7.12.1 Simulación de dispositivos

Introducción

El TIA Portal permite ejecutar y probar el hardware y el software del proyecto en un entorno simulado. La simulación se ejecuta directamente en la programadora o PC. Por ello no se requiere hardware adicional.

El software de simulación ofrece una interfaz de usuario gráfica para observar y modificar la configuración. Ésta varía en función del dispositivo seleccionado actualmente.

Integración en el TIA Portal

El software de simulación está totalmente integrado en el TIA Portal pero sólo es soportado por determinados dispositivos. Por lo tanto, el botón para iniciar el software de simulación sólo está activado si el dispositivo seleccionado soporta la simulación.

El software de simulación requiere para algunos dispositivos una interfaz virtual propia para comunicarse con los dispositivos simulados. La interfaz virtual se encuentra en el árbol del proyecto en la rúbrica "Accesos online" junto a las interfaces físicas de la progamadora o PC.

Obtendrá ayuda adicional referente a la simulación al abrir el software a través de un enlace especial.

Consulte también

Iniciar la simulación (Página 488)

7.12.2 Iniciar la simulación

Algunos dispositivos se pueden simular con un software adicional. Por ello no es necesario disponer realmente de los dispositivos para poder comprobar el proyecto exhaustivamente.

Procedimiento

Para iniciar el software de simulación, proceda del siguiente modo:

1. Seleccione el dispositivo que desea simular, p. ej. en el árbol del proyecto.
2. Elija el comando "Simulación > Iniciar" del menú "Online".
   Se inicia el software de simulación.

Consulte también

Simulación de dispositivos (Página 487)

Editar dispositivos y redes 8

8.1 Configurar dispositivos y redes

8.1.1 Editor de hardware y redes

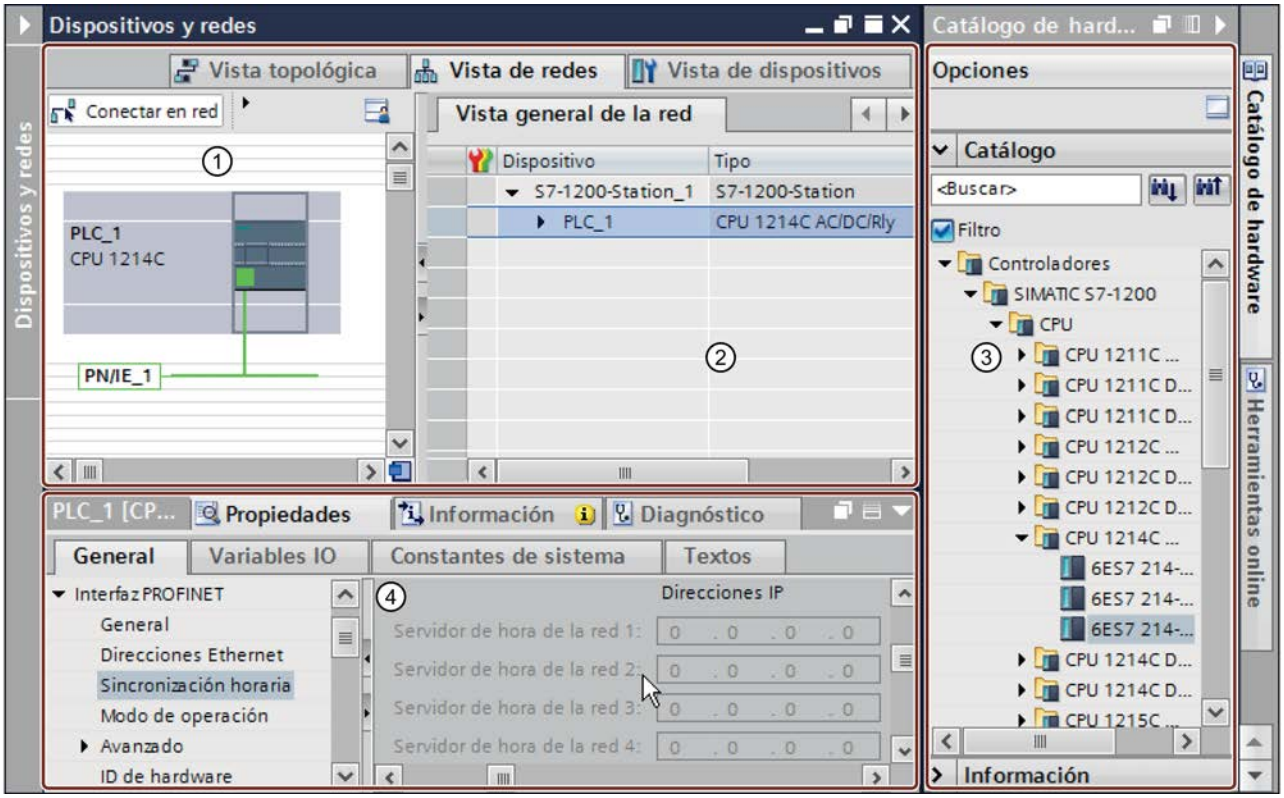
8.1.1.1 Resumen del editor de hardware y redes

Función del editor de hardware y redes

Al hacer doble clic en la entrada "Dispositivos y redes" del árbol del proyecto se abre el editor de hardware y redes. El editor de hardware y redes es el entorno de desarrollo integrado para configurar, parametrizar y conectar dispositivos y módulos en red. Ofrece todo el soporte necesario para realizar el proyecto de automatización.

Estructura del editor de hardware y redes

El editor de hardware y redes incluye los componentes siguientes:



① Vista de dispositivos (Página 493), Vista de redes (Página 491), Vista topológica (Página 496): Área gráfica

② Vista de dispositivos (Página 493), Vista de redes (Página 491), Vista topológica (Página 496): Área de tabla

③ Catálogo de hardware (Página 503)

④ Ventana de inspección (Página 501)

El editor de hardware y redes ofrece tres vistas distintas para su proyecto. Es posible cambiar entre estas tres vistas en cualquier momento, en función de si deben crearse y editarse dispositivos y módulos individuales, redes y configuraciones de dispositivos enteras o bien la estructura topológica del proyecto.

Desde el catálogo de hardware se arrastran los dispositivos y módulos necesarios para el sistema de automatización hasta la vista de dispositivos, redes o topología.

La ventana de inspección ofrece información sobre el objeto actualmente seleccionado. También permite modificar los ajustes del objeto seleccionado.

8.1.1.2 Vista de redes

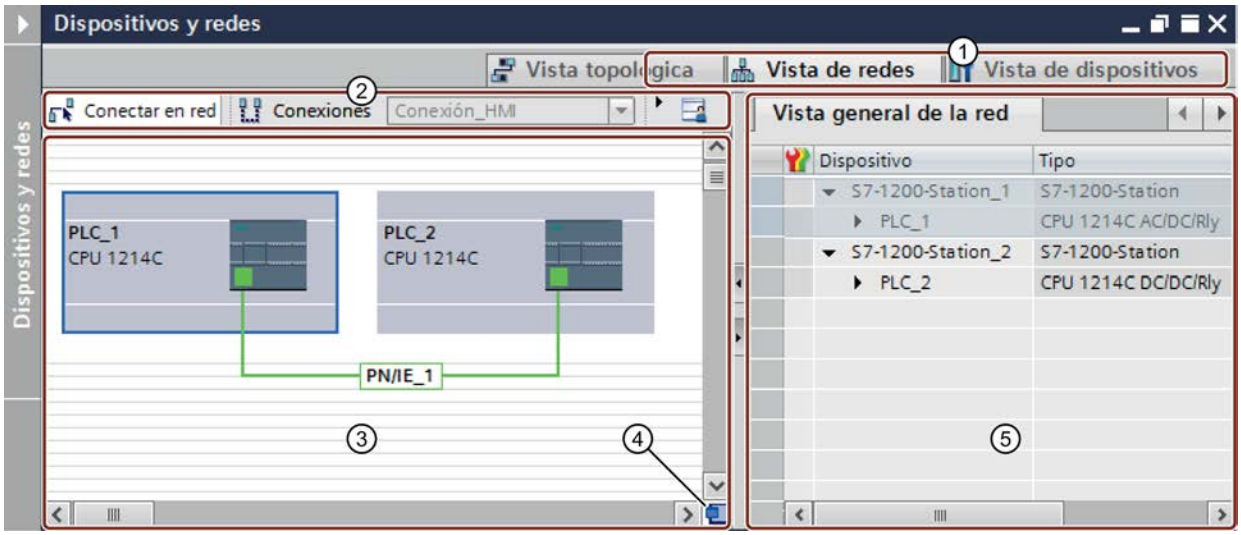
Introducción

La vista de redes es una de las tres áreas de trabajo del editor de hardware y de redes. Aquí se realizan las tareas siguientes:

- Configurar y parametrizar dispositivos
- Interconectar dispositivos

Estructura

La figura siguiente muestra los componentes de la vista de redes:



① Conmutador vista de redes/vista de dispositivos/vista topológica
② Barra de herramientas de la vista de redes
③ Área gráfica de la vista de redes
④ Navegación general
⑤ Área de tabla de la vista de redes

La división entre el área gráfica y de tabla de la vista de redes se modifica con el ratón. Para ello, haga clic entre el área gráfica y el área de tabla y, sin soltar el botón del ratón, desplace la línea divisoria a izquierda o derecha para modificar la división. El Speedy Splitter (las dos teclas de flecha pequeñas) permiten minimizar o maximizar el área de tabla con un solo clic, así como restablecer la última división seleccionada.

Barra de herramientas

La barra de herramientas ofrece las siguientes funciones:

Icono	Significado
	Modo de conexión en red de los dispositivos.
	Modo para crear conexiones. El tipo de conexión se ajusta desde la lista desplegable adjunta.
	Modo para crear relaciones.
	Visualización de direcciones de interfaces.
	Ajuste del nivel de zoom. Es posible seleccionar o introducir directamente el nivel de zoom desde la lista desplegable adjunta. El símbolo de zoom permite ampliar o reducir la imagen gradualmente o bien rodear con un borde un área que deba ser ampliada.
	Mostrar saltos de página Activa la vista preliminar del salto de página. Se visualizan líneas punteadas allí donde se insertará un salto de página al imprimir.
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Área gráfica

El área gráfica de la vista de redes muestra los dispositivos relacionados con la red, redes, conexiones y relaciones. Aquí se insertan dispositivos desde el catálogo de hardware, se conectan entre sí mediante las interfaces y se editan los ajustes de comunicación.

Navegación general

Haga clic con el ratón en la navegación general para obtener una vista general de los objetos creados en el área gráfica. Manteniendo pulsado el botón del ratón en la navegación general se navega rápidamente a los objetos deseados y se visualizan en el área gráfica.

Área de tabla

El área de tabla de la vista de redes contiene diversas tablas relacionadas con los dispositivos existentes, las conexiones y los ajustes de comunicación:

- Vista general de redes
- Conexiones
- Relaciones
- Comunicación E/S
- VPN

En el menú contextual de la barra de título de la tabla se puede adaptar la representación de la tabla.

Consulte también

Agregar un dispositivo a la configuración de hardware (Página 520)

Estructura de la interfaz de usuario (Página 283)

Mostrar el estado de diagnóstico y de comparación mediante iconos (Página 2160)

Conectar dispositivos en red en la vista de redes (Página 535)

Área de tabla de la vista general de redes (Página 538)

8.1.1.3 Vista de dispositivos

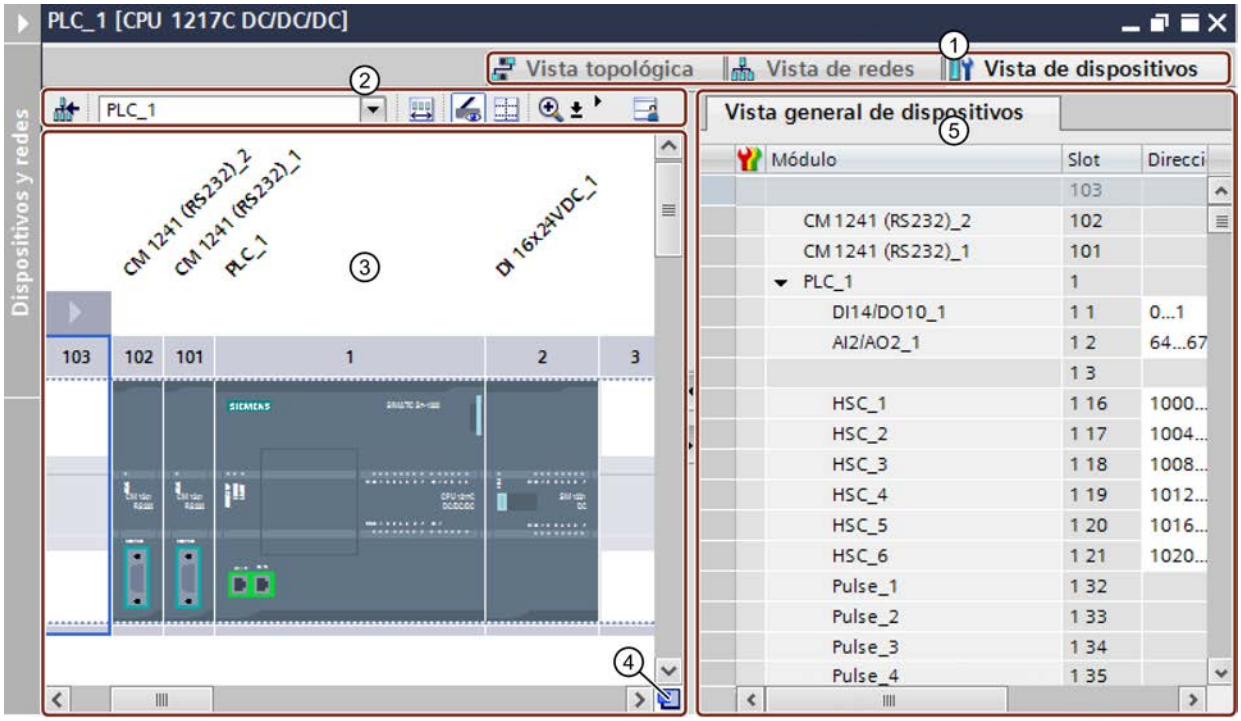
Introducción

La vista de dispositivos es una de las tres áreas de trabajo del editor de hardware y de redes. Aquí se realizan las tareas siguientes:

- Configurar y parametrizar dispositivos
- Configurar y parametrizar módulos

Estructura

La figura siguiente muestra los componentes de la vista de dispositivos:



① Conmutador vista de redes/vista de dispositivos/vista topológica
② Barra de herramientas de la vista de dispositivos
③ Área gráfica de la vista de dispositivos
④ Navegación general
⑤ Área de tabla de la vista de dispositivos

La división entre el área gráfica y de tabla de la vista de dispositivos se modifica con el ratón. Para ello, haga clic entre el área gráfica y el área de tabla y, sin soltar el botón del ratón, desplace la línea divisoria a izquierda o derecha para modificar la división. El Speedy Splitter (las dos teclas de flecha pequeñas) permiten minimizar o maximizar el área de tabla con un solo clic, así como restablecer la última división seleccionada.

Barra de herramientas

La barra de herramientas ofrece las siguientes funciones:

Icono	Significado
	Cambia a la vista de redes. Nota: en la vista de dispositivos se puede cambiar entre los dispositivos disponibles desde la lista desplegable adjunta.
	Visualización del área de módulos no enchufados
	Visualización de la rotulación de módulos.

Icono	Significado
	Ajuste del nivel de zoom. Es posible seleccionar o introducir directamente el nivel de zoom en la lista desplegable adjunta. El símbolo de zoom también permite aumentar o reducir la vista gradualmente o bien trazar un borde alrededor de un área que desea ampliarse. A partir de un nivel de zoom del 200 % es posible distinguir las denominaciones de direcciones de los canales E/S en los módulos de señales.
	Mostrar saltos de página Activa la vista preliminar del salto de página. Se visualizan líneas punteadas allí donde se insertará un salto de página al imprimir.
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Área gráfica

El área gráfica de la vista de dispositivos muestra los componentes de hardware y, en su caso, los módulos correspondientes que están asignados unos a otros a través de uno o varios racks. En los dispositivos con racks es posible insertar otros objetos de hardware desde el catálogo de hardware hasta los slots de los racks.

Navegación general

Haga clic con el ratón en la navegación general para obtener una vista general de los objetos creados en el área gráfica. Manteniendo pulsado el botón del ratón en la navegación general se navega rápidamente a los objetos deseados y se visualizan en el área gráfica.

Área de tabla

El área de tabla de la vista de dispositivos ofrece una vista general del hardware utilizado con los principales datos técnicos y organizativos.

En el menú contextual de la barra de título de la tabla se puede adaptar la representación de la tabla.

Consulte también

Trabajar con racks (Página 514)

Vista de redes (Página 491)

Área de módulos no enchufados (Página 516)

Insertar un módulo en un rack (Página 523)

Objetos en la vista de dispositivos (Página 515)

Estructura de la interfaz de usuario (Página 283)

Mostrar el estado de diagnóstico y de comparación mediante iconos (Página 2160)

8.1.1.4 Vista topológica

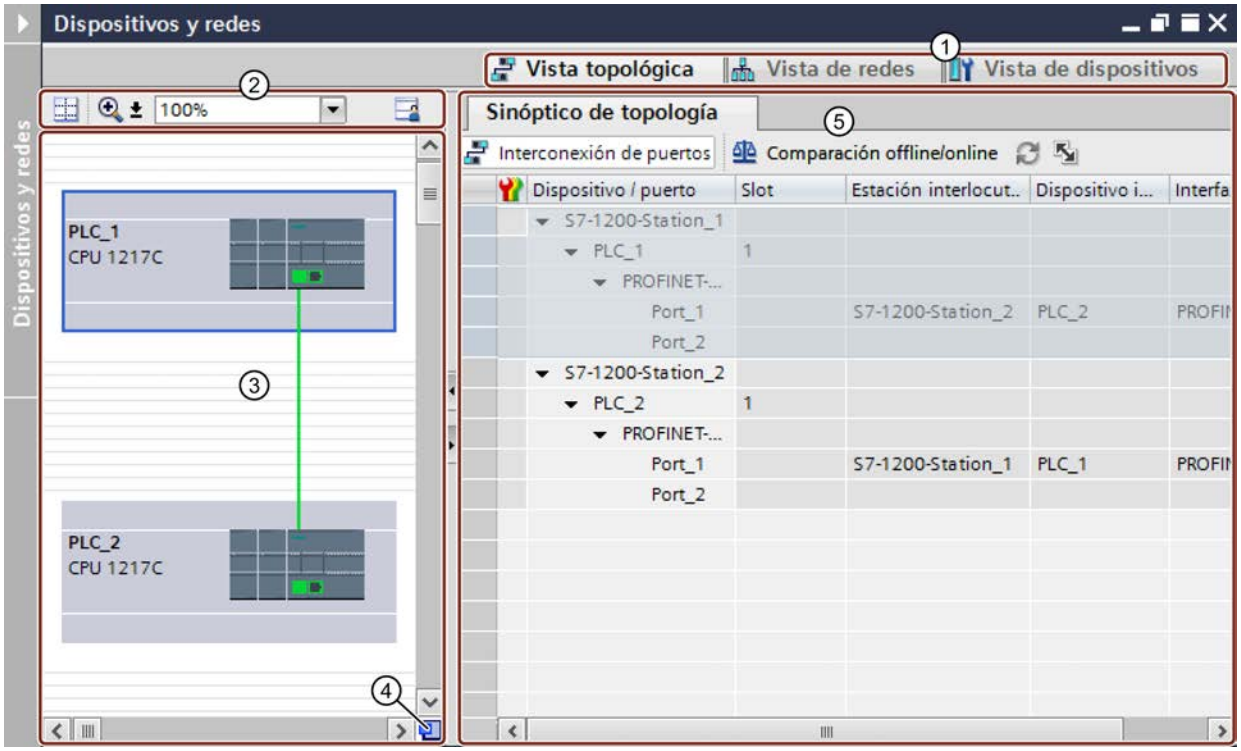
Introducción

La vista topológica es una de las tres áreas de trabajo del editor de hardware y redes. Aquí se realizan las tareas siguientes:

- Mostrar la topología de Ethernet
- Configurar la topología Ethernet
- Determinar y minimizar diferencias entre la topología prevista y la topología real

Estructura

La siguiente figura muestra una vista general de la vista topológica.



① Conmutador vista de dispositivos/vista de redes/vista topológica
② Barra de herramientas de la vista topológica
③ Área gráfica de la vista topológica
④ Navegación general
⑤ Área de tabla de la vista topológica

La división entre el área gráfica y de tabla de la vista topológica se modifica con el ratón. Para ello, haga clic entre el área gráfica y el área de tabla y, sin soltar el botón del ratón, desplace la línea divisoria a izquierda o derecha para modificar la división. El Speedy Splitter (las dos teclas de flecha pequeñas) permiten minimizar o maximizar el área de tabla con un solo clic, así como restablecer la última división seleccionada.

Barra de herramientas

La barra de herramientas ofrece las siguientes funciones:

Icono	Significado
	Ajuste del nivel de zoom. Es posible seleccionar o introducir directamente el nivel de zoom desde la lista desplegable adjunta. El símbolo de zoom permite ampliar o reducir la imagen gradualmente o bien rodear con un borde un área que deba ser ampliada.
	Mostrar saltos de página Activa la vista preliminar del salto de página. Se visualizan líneas punteadas allí donde se insertará un salto de página al imprimir.
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Área gráfica

En el área gráfica de la vista topológica se visualizan los módulos Ethernet con los correspondientes puertos e interconexiones de puertos. Aquí se pueden agregar otros objetos de hardware con interfaces Ethernet. Consulte: Agregar un dispositivo a la configuración de hardware (Página 520)

Navegación general

Haga clic con el ratón en la navegación general para obtener una vista general de los objetos creados en el área gráfica. Manteniendo pulsado el botón del ratón en la navegación general se navega rápidamente a los objetos deseados y se visualizan en el área gráfica.

Área de tabla

Aquí se muestran los módulos Ethernet y PROFINET cuyos puertos e interconexiones de puertos tienen formato de tabla. Esta tabla corresponde a la tabla de la vista general de la red en la vista de redes.

En el menú contextual de la barra de título de la tabla se puede adaptar la representación de la tabla.

Consulte también

Estructura de la interfaz de usuario (Página 283)

Mostrar el estado de diagnóstico y de comparación mediante iconos (Página 2160)

8.1.1.5 Resumen de los ajustes de la configuración hardware

En los ajustes del TIA Portal pueden modificarse algunas opciones de la configuración hardware.

Resumen

La siguiente tabla muestra un resumen de los ajustes de la configuración hardware:

Grupo	Ajuste	Descripción
Información del Product Support	Desactivado	Impide el acceso a Siemens Industry Online Support
	Vía Internet	Permite acceder vía Internet a la información de producto de los diferentes dispositivos del catálogo de hardware.
Vista general de la topología	Asignar temporalmente dirección IP	Si un dispositivo no posee una dirección IP válida, permite asignarle una con carácter temporal para facilitar la detección de la topología. En ausencia de una dirección IP válida, no es posible obtener información de la topología (LLDP) de un dispositivo.
	Mostrar advertencia si la opción está activada	Muestra una advertencia cuando se asigna una dirección IP temporal a un dispositivo para la detección de la topología.
Compilar	Descargar comentario del módulo	Al cargar la configuración hardware en el dispositivo, transfiere también los comentarios que puedan existir. Los comentarios estarán disponibles tras cargar el dispositivo en la programadora.

Consulte también

Modificar la configuración (Página 283)

8.1.1.6 Imprimir configuraciones de hardware y redes

Impresión de configuraciones de hardware y redes

En la documentación del proyecto es posible imprimir los elementos siguientes de la vista de hardware y redes:

- Vista gráfica de redes
- Tabla general de las redes
- Vista gráfica de dispositivos
- Tabla general de los dispositivos
- Los parámetros del objeto seleccionado actualmente en el editor

Impresión de contenidos de un editor

Si se inicia la impresión dentro de un editor abierto y no se selecciona ningún módulo, se imprimen siempre los contenidos del editor. Aquí se incluyen la representación gráfica del editor así como la tabla perteneciente al editor. Sin embargo, es posible adaptar la impresión. Es posible ajustar si debe imprimirse sólo la vista gráfica, sólo la tabla o ambas. Lea al respecto el capítulo "Modificar las opciones de impresión (Página 500)".

Si el gráfico es mayor que el formato de página seleccionado, la impresión continúa en la página siguiente. De esta manera no se pierden contenidos. Alternativamente, es posible

modificar el nivel de zoom de la representación gráfica para ajustar la impresión a una página. La impresión se realiza siempre en el nivel de zoom actualmente seleccionado.

Para controlar si todos los contenidos se ajustan a una página, utilice la vista preliminar o active la vista preliminar de salto de página. Con la vista preliminar de salto de página activada, en el editor gráfico se visualizan líneas punteadas en el lugar donde después habrá un salto de página.

Impresión de tablas muy grandes

Si una tabla es mayor que el área de impresión y no se puede imprimir completamente, los contenidos de la tabla no se imprimen en forma de tabla sino como una relación de valores y claves.

Ejemplo:

Nombre de objeto	Propiedad 1	Propiedad 2
Objeto A	Valor A1	Valor A2
Objeto B	Valor B1	Valor B2

En este caso, la impresión queda del siguiente modo:

Objeto A

Propiedad 1: Valor A1

Propiedad 2: Valor A2

Objeto B

Propiedad 1: Valor B1

Propiedad 2: Valor B2

También es posible ajustar este formato de impresión como estándar, de forma que las tablas se impriman siempre como una relación de claves y valores. Lea al respecto el capítulo "Modificación de los ajustes de impresión (Página 392)".

Impresión de parámetros de módulos

Los parámetros de los módulos seleccionados se imprimen junto con los valores actuales en forma de texto. Todos los parámetros de los respectivos módulos se imprimen igualmente. Así, por ejemplo, si se ha seleccionado una CPU, se imprimirán los parámetros de la Signal Board, si hay una insertada.

Se determina el alcance con que deben imprimirse los parámetros de módulos. En el cuadro de diálogo "Imprimir", seleccione si quiere imprimir todas las propiedades y los parámetros de un módulo, o si desea una impresión compacta. Si selecciona la forma compacta, se imprimen sólo las entradas incorporadas en el área "General" de las propiedades del módulo. Quedan excluidos de aquí los comentarios sobre el módulo así como el autor y la descripción del módulo. En el modo compacto se imprimen además, por ejemplo, los siguientes parámetros de un módulo:

- Indicaciones sobre el módulo
  Nombre, slot de módulo, resumen técnico, referencia, versión de firmware
- Nombre de la interfaz PROFINET
- Indicaciones sobre la subred
  Nombre de la subred, ID de la subred S7

Consulte también

Modificar las opciones de impresión (Página 500)

Función de documentación (Página 390)

Crear la vista preliminar (Página 404)

Imprimir los datos de proyecto (Página 407)

Activar vista preliminar de salto de página para la impresión (Página 500)

8.1.1.7 Activar vista preliminar de salto de página para la impresión

En los editores gráficos es posible activar una vista preliminar de salto de página para la impresión. Si la opción está activada, se visualizan líneas punteadas en los lugares del editor gráfico donde se inserta un salto de página en la impresión posterior.

Procedimiento

Para activar la vista preliminar de salto de página, proceda del siguiente modo:

1. Seleccione el área gráfica de la vista correspondiente.
2. Haga clic en el símbolo "Mostrar saltos de página" en la barra de herramientas del editor gráfico.
   Dentro del editor gráfico se visualizan líneas punteadas en el lugar donde después habrá un salto de página.
3. Para modificar el diseño del borde, elija el comando "Imprimir" del menú "Proyecto".
4. Para desactivar la vista preliminar de saltos de página, haga clic de nuevo en el símbolo "Mostrar saltos de página" en la barra de herramientas del editor gráfico.

8.1.1.8 Modificar las opciones de impresión

Modificar el alcance de la impresión

Si se imprime desde un editor, es posible determinar si deben imprimirse tanto gráficos como tablas o bien sólo uno de los dos. Por defecto se imprimen ambos.

Procedimiento

Para cambiar el alcance de la impresión, proceda del siguiente modo:

1. En el menú "Herramientas", elija el comando "Configuración".
2. Abra el grupo de parámetros "Configuración de impresión" en "General" de la navegación local.
3. Desplácese hasta el grupo "Configuración hardware".
4. Active o desactive la casilla de verificación "Vista gráfica activa" según si desea imprimir los gráficos de la vista de redes y dispositivos.
5. Active o desactive la casilla de verificación "Tabla activa" según si desea imprimir la tabla perteneciente a un editor.

Consulte también

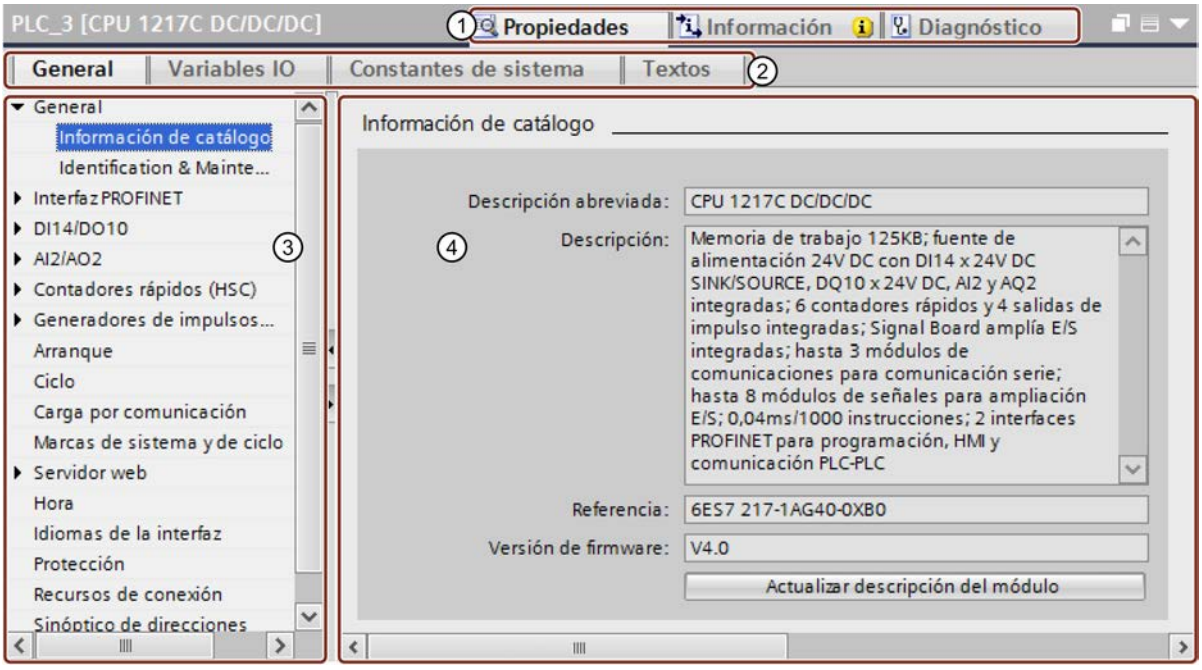
Imprimir configuraciones de hardware y redes (Página 498)

8.1.1.9 Ventana de inspección

Las propiedades y los parámetros visualizados del objeto seleccionado se editan dentro de la ventana de inspección.

Estructura

La ventana de inspección incluye los componentes siguientes:



① Conmutación entre diferentes áreas de información y trabajo

② Conmutación entre distintas fichas de las áreas de información y trabajo

③ Navegación entre distintas informaciones y parámetros

④ Visualización de las informaciones y parámetros seleccionados

Función

La información y los parámetros de la ventana de inspección se dividen en distintos tipos de información:

- Propiedades
- Información
- Diagnóstico

Para visualizar la información y los parámetros correspondientes haga clic en el área deseada. Para configurar un sistema de automatización, el área más importante es la de "Propiedades". Esta área se muestra por defecto y contiene diferentes fichas:

- General: Indica las propiedades y ajustes del dispositivo o módulo. Aquí pueden editarse los ajustes y parámetros. En la parte izquierda de la ventana de inspección se encuentra la navegación local, en la que la información y los parámetros aparecen clasificados en grupos. Si se hace clic en el símbolo de flecha situado a la izquierda del nombre de grupo se abrirá el grupo con los subgrupos existentes. Si se selecciona un grupo o subgrupo, la información y los parámetros correspondientes aparecerán en la parte derecha de la ventana de inspección, donde también pueden editarse.
- Variables IO: Indica las variables IO de los PLC. Es posible asignar nombres a las variables, asignar las variables mediante una lista desplegable a las tablas de variables definidas por el usuario y dotar las variables de comentarios. Las variables IO también aparecen en la tabla de variables PLC.
- Constantes de sistema: Indica las constantes requeridas por el sistema con los identificadores de hardware de los módulos. Las constantes de sistema también aparecen en la tabla de variables PLC.
- Textos: Indica el idioma de referencia y el origen de los textos del proyecto.

Consulte también

Editar las propiedades y los parámetros (Página 529)

Resumen del editor de hardware y redes (Página 489)

Traducción de textos correspondientes a objetos individuales (Página 353)

Sinopsis de las tablas de variables PLC (Página 2566)

Principios básicos de los textos del proyecto (Página 348)

Información importante sobre identificadores de hardware (Página 917)

Direccionamiento de módulos (Página 971)

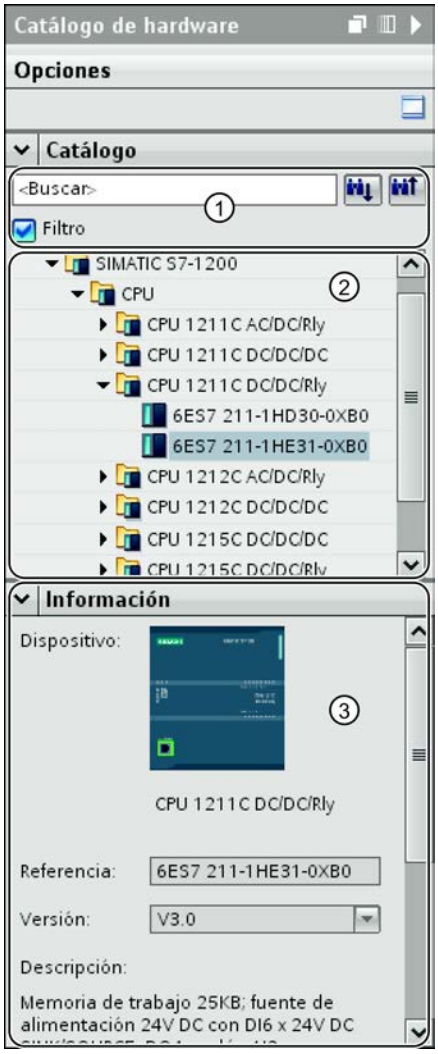
Direccionamiento de módulos (S7-1500) (Página 915)

8.1.1.10 Catálogo de hardware

La Task Card "Catálogo de hardware" permite acceder fácilmente a los componentes de hardware más diversos.

Estructura

La Task Card "Catálogo de hardware" consta de las paletas siguientes:



① Paleta "Catálogo", función de búsqueda y filtro

② Paleta "Catálogo", selección de componentes

③ Paleta "Información"

Función de búsqueda y filtro

Las funciones de búsqueda y filtro de la paleta "Catálogo" permiten buscar cómodamente determinados componentes de hardware. La función de filtro permite limitar la visualización de los componentes HW según determinados criterios. Por ejemplo, es posible limitar la visualización a objetos que se pueden colocar también en el contexto actual o que contienen determinadas funciones.

Los objetos utilizables en el contexto actual son, p. ej., objetos que pueden conectarse en red en la vista de redes, o bien los únicos módulos compatibles con el dispositivo en la vista de dispositivos.

Selección de componentes

La selección de componentes de la paleta "Catálogo" contiene los componentes de hardware instalados en una estructura de árbol. Desde allí se arrastran los dispositivos o módulos deseados al área de trabajo gráfica de la vista de dispositivos o redes.

Los componentes de hardware que están instalados pero que todavía no disponen de licencia se visualizan atenuados. Los componentes HW sin licencia no se pueden utilizar.

Los componentes de hardware que pertenecen a diversos grupos de componentes temáticos se presentan parcialmente como objeto enlazado. Si se hace clic en uno de estos componentes de hardware enlazados, se abre la estructura del catálogo donde se encuentran los respectivos componentes de hardware.

Información

En la paleta "Información" aparece información detallada del objeto seleccionado en el catálogo:

- Representación esquemática
- Nombre
- Referencia
- Número de versión
- Descripción

Consulte también

Examinar el catálogo de hardware (Página 512)

Resumen del editor de hardware y redes (Página 489)

8.1.1.11 Activar el soporte del producto

Para cada dispositivo del catálogo de hardware existe información relacionada que está almacenada en Siemens Industry Online Support. Dicha función está desactivada por defecto. A continuación se explica cómo activarla.

Requisitos

El TIA Portal dispone de acceso a Internet.

Procedimiento

Para activar el acceso a Siemens Industry Online Support, proceda del siguiente modo:

1. En el menú "Herramientas", elija el comando "Configuración".
2. Abra el grupo "Configuración de hardware" en la navegación local.
3. Active la casilla de verificación "Vía Internet".

Resultado

Dentro del catálogo de hardware se accede al soporte del producto, las preguntas frecuentes y los manuales desde el menú contextual de un módulo.

Consulte también

Mostrar el soporte del producto para componentes de hardware (Página 506)

8.1.1.12 Mostrar el soporte del producto para componentes de hardware

En el catálogo de hardware se accede directamente a la información almacenada en Siemens Industry Online Support para cada módulo. Se ofrecen los siguientes accesos directos a las páginas respectivas de Siemens Industry Online Support:

- Información de soporte para el producto
- Preguntas frecuentes
- Manuales

Requisitos

- Hay acceso a Internet.
- El acceso al soporte del producto está activado en la configuración del TIA Portal. Para saber cómo activar la función, consulte el capítulo "Activar soporte del producto (Página 505)".

Procedimiento

Para mostrar la información de Siemens Industry Online Support sobre un módulo concreto, proceda del siguiente modo:

1. Navegue hasta el módulo deseado en el catálogo de hardware.
2. Haga clic con el botón derecho del ratón en el módulo.
3. Elija una de las entradas siguientes del menú contextual:
   - Información de soporte para el producto
   - Preguntas frecuentes
   - Manuales

Resultado

Se abre el navegador predeterminado ajustado en el sistema operativo y se carga la página pertinente de Siemens Industry Online Support.

Consulte también

Activar el soporte del producto (Página 505)

8.1.1.13 Comandos de teclado: navegación en el editor

En la vista de redes y dispositivos es posible navegar con combinaciones de teclas entre los componentes del editor de hardware y redes.

Navegar entre elementos y funciones

Función	Combinación de teclas
Cambio al siguiente nivel de selección inferior La tecla <Intro> permite cambiar, por ejemplo, de un rack seleccionado al nivel de selección inferior de los dispositivos y módulos disponibles. Si hay un dispositivo seleccionado, con la tecla <Intro> se cambia al nivel de selección inferior de las interfaces visualizadas en el dispositivo.	<Intro>
Cambio al siguiente nivel de selección superior La tecla <Esc> permite cambiar, por ejemplo, de una interfaz seleccionada al nivel de selección superior de los dispositivos y módulos. Si hay un dispositivo seleccionado, con la tecla <Esc> se cambia al nivel de selección superior de los racks.	<Esc>
Navegación entre objetos en el nivel de selección actual Las teclas de flecha permiten navegar entre los objetos en el nivel de selección actual. Para cambiar el nivel de selección utilice las teclas <Intro> o <Esc>.	<Flecha hacia arriba> <Flecha hacia abajo> <Flecha a la derecha> <Flecha a la izquierda>
Cambia a la vista de dispositivos	<Ctrl+Mayús+D>
Cambia a la vista de redes	<Ctrl+Mayús+N>
Cambia a la vista topológica	<Ctrl+Mayús+T>
Cambiar entre elementos del editor Con la tecla <Tab> se cambia de un elemento del editor al siguiente. Con <Mayús +Tab> se cambia al elemento anterior. Así, por ejemplo, es posible cambiar entre la vista gráfica, Speedy Splitter, la vista de tabla o fichas subordinadas.	<Tab> <Mayús+Tab>
Cambiar entre fichas Con las teclas <Ctrl+Tab> se cambia de una ficha a la siguiente de la derecha. Con <Ctrl+Mayús+Tab> se cambia a la siguiente ficha de la izquierda. Así, por ejemplo, con las teclas es posible cambiar entre la vista de dispositivos, redes y topológica.	<Ctrl+Tab> <Ctrl+Mayús+Tab>

Abrir elementos y funciones

Función	Combinación de teclas
Abrir la vista online y de diagnóstico Con el dispositivo seleccionado, pulsando <Ctrl+D> se abre la vista online y de diagnóstico para el dispositivo seleccionado.	<Ctrl+D>
Abrir el cuadro de diálogo para cargar en dispositivo Con el dispositivo seleccionado, pulsando <Ctrl+L> se abre el cuadro de diálogo para la carga avanzada en el dispositivo seleccionado.	<Ctrl+L>
Agregar un dispositivo Con <Ctrl+N> se abre el cuadro de diálogo para agregar un dispositivo nuevo.	<Ctrl+N>
Abre la Task Card "Catálogo de hardware"	<Ctrl+Mayús+C>
Abre la Task Card "Herramientas online"	<Ctrl+Mayús+O>

Consulte también

Manejo del teclado en el TIA Portal (Página 314)

8.1.1.14 Comandos de teclado: editar objetos

El editor de hardware y redes permite realizar algunas funciones de la vista de redes y dispositivos utilizando combinaciones de teclado y ratón. El manejo del teclado en las tablas (Página 314) responde al comportamiento habitual. Aquí se explica el manejo del teclado para el área de trabajo gráfica de la vista de redes y dispositivos.

Manejo general del teclado

Función	Combinación de teclas
Ampliar la imagen al marco trazado Trace un marco en la vista gráfica para cambiar el tamaño de la vista en correspondencia.	<Ctrl+espacio> + botón del ratón pulsado
Desplazar la imagen Mueva el puntero del ratón para desplazar la vista.	<espacio> + botón del ratón pulsado
Cancelar la operación actual	<Esc>
Soltar el conector Pulsando <Esc> o haciendo doble clic se sale del modo de conexión al arrastrar una conexión.	<Esc> o doble clic
Zoom a la vista gráfica El aumento o la reducción depende del sentido de giro.	<Ctrl> + rodar rueda del ratón

Objetos seleccionados

Función	Combinación de teclas
Seleccionar objeto	Clic con el ratón
Cortar un objeto El objeto seleccionado se copia en el portapapeles y se borra de la vista gráfica.	<Ctrl+X>
Copiar objeto El objeto seleccionado se copia en el portapapeles.	<Ctrl+C>
Pegar objeto El objeto del portapapeles se pega en la selección.	<Ctrl+V>
Borrar objeto seleccionado	<Supr>
Seleccionar varios objetos 1 Es posible agregar varios objetos a los seleccionados haciendo clic con el ratón en cada uno de ellos. Otra posibilidad consiste en dibujar un marco alrededor de los objetos que desea seleccionar con <Mayús> + botón del ratón pulsado.	<Mayús> + clic con el ratón

Función	Combinación de teclas
Seleccionar varios objetos 2 Es posible agregar varios objetos a los seleccionados haciendo clic con el ratón en cada uno de ellos. Otra posibilidad consiste en dibujar un marco alrededor de los objetos que desea seleccionar con <Mayús> + botón del ratón pulsado. Manteniendo pulsada la tecla <Ctrl> y haciendo clic con el ratón se desmarcan objetos seleccionados.	<Ctrl> + clic con el ratón
Desplazar selección Manteniendo pulsado el botón del ratón es posible arrastrar dispositivos o módulos a slots permitidos de un rack.	Botón del ratón pulsado
Copiar selección Con <Ctrl> y manteniendo pulsado el botón del ratón es posible arrastrar dispositivos o módulos a slots permitidos de un rack. En este caso, los dispositivos o módulos se copian.	<Ctrl> + botón del ratón pulsado

8.1.2 Configurar dispositivos

8.1.2.1 Principios básicos

Introducción a la configuración de hardware

Para crear un sistema de automatización, los diferentes componentes de hardware deben configurarse, parametrizarse y conectarse entre sí. Las tareas correspondientes se realizan en la vista de dispositivos y redes.

Configurar

Se entiende por "configurar" la disposición, el ajuste y la interconexión de dispositivos y módulos dentro de la vista de dispositivos o redes. Los racks se representan mediante símbolos. Al igual que los racks "reales", permiten insertar un número determinado de módulos.

A cada módulo se le asigna automáticamente una dirección. Las direcciones se pueden modificar posteriormente.

Al arrancar el sistema de automatización, la CPU compara la configuración teórica creada por software con la configuración real de la instalación. Esto permite detectar y notificar inmediatamente los posibles errores.

Parametrizar

Se entiende por "parametrizar" el ajuste de las propiedades de los componentes utilizados. Se parametrizan los componentes de hardware y los ajustes para el intercambio de datos:

- Propiedades de módulos parametrizables
- Ajustes para el intercambio de datos entre componentes

Al arrancar, los parámetros se cargan en la CPU y se transfieren desde ella a los módulos correspondientes. Es muy fácil sustituir módulos, porque los parámetros creados se transfieren automáticamente al módulo nuevo al arrancar.

Adaptar el hardware a las necesidades del proyecto

Es necesario configurar el hardware al crear, ampliar o modificar un proyecto de automatización. Para ello se agregan componentes de hardware a la instalación, se conectan con los componentes existentes y se adaptan sus propiedades a las tareas.

Las propiedades de los sistemas de automatización y módulos están preajustadas de modo que en muchos casos no es necesario realizar ninguna parametrización. Sin embargo, la parametrización es necesaria en los siguientes casos:

- Cuando hay que modificar parámetros predeterminados de un módulo.
- Cuando hay que utilizar funciones especiales.
- Cuando hay que configurar conexiones.

Consulte también

Modificar las propiedades de los módulos (Página 1644)

Aplicar configuraciones existentes

Abrir proyectos existentes

Al abrir proyectos que ya existen, se verifica automáticamente si se ha instalado el software correspondiente para todos los módulos utilizados en el proyecto. Si se intenta abrir un proyecto con módulos que no se soportan en el actual alcance de suministro del TIA Portal, nada más abrirse el proyecto aparece una indicación de los componentes de software que faltan. El proyecto se puede abrir si los componentes de software no resultan imprescindibles para su inicio.

Comportamiento en caso de faltar componentes de software

Los proyectos que incluyen módulos no soportados por el alcance de suministro actual se comportan de la siguiente manera:

- Visualización de módulos en la interfaz de usuario
  - Los módulos no soportados se visualizan en el árbol del proyecto con todos sus objetos subordinados, pero los propios módulos no se pueden procesar en los editores o en la ventana de inspección. A ser posible, se utiliza un módulo sustitutivo que se adapte óptimamente al módulo original. Los módulos sustitutivos están marcados con un signo de exclamación.
  - La visualización de las propiedades en las tablas está limitada. Esto concierne sobre todo a la representación de parámetros de red, como p. ej. la dirección IP.
- Restricciones funcionales
  - Los módulos no soportados no se pueden imprimir ni compilar.
  - No es posible establecer una conexión online con los módulos. Por ello tampoco es posible realizar una descarga.
  - Para modificar el tipo de dispositivo, es necesario borrar primero el dispositivo e insertarlo de nuevo. La función "Modificar tipo de dispositivo" no se soporta.
  - Se permite copiar y pegar objetos subordinados, como p. ej. bloques, pero no copiar y pegar el propio dispositivo.
  - No se permite modificar la configuración de red dentro de la vista de redes en el caso de los módulos sustitutivos.
  - Las referencias cruzadas sí se visualizan. No obstante, las referencias cruzadas reproducen sólo el último estado guardado en el proyecto, porque no se puede realizar una sincronización online con el módulo original.

Consulte también

Abrir proyectos (Página 342)

Reglas generales para los slots

Introducción

En cada sistema de automatización y en cada módulo rigen unas reglas específicas para los slots.

Al seleccionar un módulo del catálogo de hardware en la vista de dispositivos, en el rack se marcan todos los slots posibles para el módulo seleccionado. Los módulos solo pueden arrastrarse a los slots marcados.

Si se inserta, desplaza o sustituye un módulo, también se aplican las reglas para los slots.

Coherencia

Algunas reglas para slots dependen de la configuración del entorno. Esto significa que, a veces, es posible enchufar módulos en el rack aunque esto dé lugar a incoherencias en ese momento. Al modificar la configuración, p. ej. en vistas a seleccionar otros módulos o a configurar los parámetros de los módulos, es posible restablecer la coherencia de la configuración.

En los casos en los que la inserción de un módulo provoca una incoherencia corregible se permite dicha inserción. Al compilar la configuración se comprueba la coherencia. Si se detectan incoherencias, se notifican en la ventana de inspección, en "Información". Dependiendo de cuáles sean los resultados de la comprobación de coherencia debe revisarse la configuración y restablecer la coherencia.

Reglas para la disposición de los módulos

Para los módulos insertados en los racks rigen las reglas siguientes:

- En un rack solo se insertan módulos.
- En un módulo solo se insertan módulos de interfaz.
- En un rack se permite utilizar solo módulos de la misma familia de productos o sistemas.

Además, para algunos módulos se aplican estas reglas especiales:

- Posibilidad de insertar solo en determinados slots
- Posibilidad de insertar en función de otros módulos, CPUs o ajustes
- Restricción del número de usos en un rack

Examinar el catálogo de hardware

Introducción

Para seleccionar los componentes de hardware deseados para una configuración, utilice la Task Card "Catálogo de hardware". Utilice el catálogo de hardware para seleccionar los componentes de hardware que pueden conectarse en red en la vista de redes y topológica así como para seleccionar los módulos deseados en la vista de dispositivos.

Filtro contextual

El número de componentes HW visualizados y que debe encontrar la búsqueda se puede limitar con la opción "Filtro" del catálogo de hardware.

Si se activa el filtro, sólo se visualizarán los componentes del catálogo de hardware que pueden seleccionarse en esos momentos. Si no se activa el filtro se mostrará todo el catálogo de hardware.

Al cambiar entre las distintas vistas, la vista de los objetos filtrados se adapta al contexto actual.

Posibilidades de búsqueda

Con la función de búsqueda se buscan entradas determinadas en el catálogo de hardware. Tenga en cuenta las reglas siguientes al introducir los términos de búsqueda:

- No se distingue entre mayúsculas y minúsculas.
- Los guiones y los espacios se ignoran.
- En la búsqueda se tienen en cuenta partes de un término buscado.
- Cuando hay varios términos de búsqueda hay que separarlos con un espacio.

La búsqueda se inicia hacia arriba o hacia abajo en el catálogo de hardware partiendo del objeto actualmente seleccionado.

Símbolo	Significado
	Búsqueda hacia abajo
	Búsqueda hacia arriba

Examinar el catálogo de hardware

Para examinar el catálogo de hardware, proceda del siguiente modo:

1. Haga clic con el ratón en el campo de entrada de la función de búsqueda
2. Introduzca un término de búsqueda. Se examinarán los elementos siguientes:
   - Nombre del dispositivo o del módulo
   - Referencia (MLFB)
   - Descripción en la paleta "Información"
3. Haga clic en el botón "Búsqueda descendente" o "Búsqueda ascendente".

Nota

Para saber el sentido de búsqueda correcto, observe el lugar que ha seleccionado en el catálogo de hardware. Para examinarlo todo, haga clic en el primer objeto del catálogo de hardware e inicie la búsqueda pulsando "Búsqueda descendente" después de haber introducido el término.

Como resultado aparece la primera coincidencia encontrada con el término de búsqueda. Si desea seguir buscando para encontrar más resultados, vaya pulsando el botón "Búsqueda descendente" o "Búsqueda ascendente".

Tenga en cuenta el filtro contextual del catálogo de hardware. Si está activado, la búsqueda dentro del catálogo HW se limita únicamente a los componentes visualizados que pueden insertarse.

Consulte también

Catálogo de hardware (Página 503)

Trabajar con racks

Introducción

Para asignar módulos a un dispositivo se necesita un rack, p. ej. un perfil soporte. Los módulos se fijan al rack y se conectan con la CPU, una fuente de alimentación u otros módulos a través del bus de fondo.

Crear un rack

Cuando se inserta un dispositivo en la vista de redes se crean automáticamente una estación y un rack adecuado al dispositivo seleccionado. En la vista de dispositivos aparece el rack con los slots disponibles. El número de slots disponibles depende del tipo de dispositivo utilizado.

Estructura de un rack

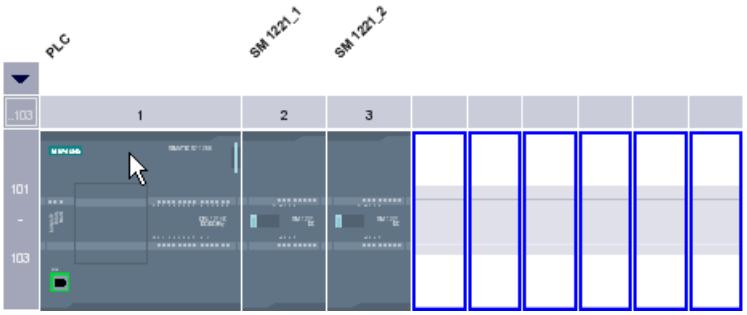
Un rack contiene siempre el dispositivo que se ha insertado en la vista de redes. El dispositivo está asignado de forma fija a un slot que, a su vez, depende del tipo de dispositivo. A la derecha y, dado el caso, también a la izquierda del dispositivo hay otros slots, encima de los cuales aparece el número correspondiente si tienen un módulo insertado.

Encima de los dispositivos y módulos insertados se indica el nombre abreviado correspondiente. Este nombre se activa y desactiva con el comando "Mostrar rotulación del módulo" del menú "Ver" de la barra de menús o con el botón correspondiente de la barra de herramientas de la vista de dispositivos (Página 493).

Símbolo	Significado
	Mostrar rotulación del módulo

Al seleccionar módulos en el catálogo de hardware, se marcan todos los slots que pueden alojar dichos módulos. De este modo se reconoce inmediatamente en qué slot puede insertarse el módulo seleccionado.

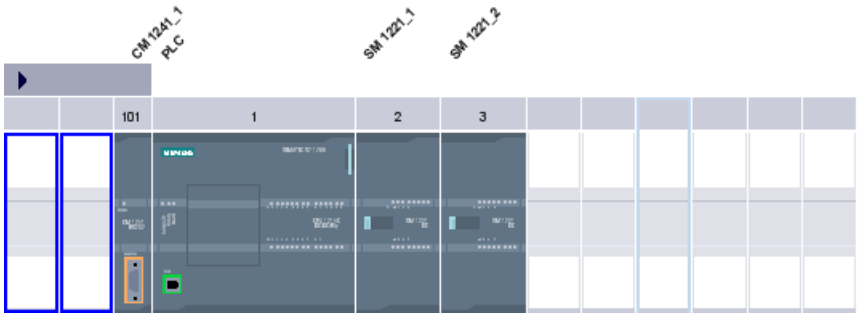
En la figura siguiente se ha seleccionado un módulo de señales del catálogo de hardware en un rack parcialmente ocupado de un S7-1200:



Puesto que los slots 101-103 están reservados para módulos de comunicación, sólo se muestran como disponibles los demás slots libres.

El grupo de slots anterior se abre y se cierra por medio de un símbolo de flecha situado encima del slot ampliable. Si está cerrado, se muestran el primer y último número de slot del grupo.

La figura siguiente muestra el grupo de slots abierto:



No es posible cerrar los grupos de slots con módulos ya insertados.

Selección múltiple de módulos y slots

Existen varias posibilidades para seleccionar varios módulos o slots:

- Selección de varios módulos o slots manteniendo pulsada la tecla <Mayús> o <Ctrl>.
- Clic con el ratón fuera del rack y manteniendo pulsado el botón del ratón: trazar un marco alrededor de los módulos o slots que deben seleccionarse.

Objetos en la vista de dispositivos

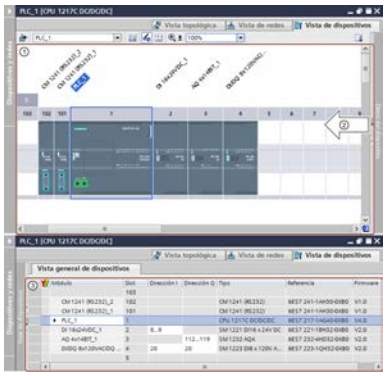
En la parte izquierda de la vista de dispositivos aparece una imagen gráfica del rack y de los dispositivos a él conectados. En la parte derecha de la vista de dispositivos se encuentra la vista general de dispositivos. La vista general de dispositivos es una tabla que contiene la información más importante relacionada con los módulos insertados en el rack. Ambas partes aparecen en una ventana. El tamaño de sendas partes se puede modificar con un separador. Mediante este separador se puede mostrar u ocultar una parte o la otra.

Estructura y contenido de la vista de dispositivos

En la vista gráfica de dispositivos se visualiza la configuración offline de los dispositivos en el rack. La representación equivale simbólicamente a la estructura en el rack real.

En la vista general de dispositivos se muestra la ocupación del rack en forma de tabla. Cada fila de la tabla contiene la información relacionada con la asignación de un slot.

La figura siguiente muestra la vista de dispositivos con la configuración de una CPU SIMATIC S7-1200.



① Vista gráfica con el rack equipado con CPU y los distintos módulos en los slots 1 a 4, así como 101 y 102.

② Con el separador puede modificar la división de la vista de dispositivos entre la parte izquierda (vista gráfica) y la parte derecha (vista de dispositivos). Haciendo clic en las flechas puede cambiar rápidamente la división de las partes.

③ Vista de dispositivos con vista de tabla de los slots del rack y de los componentes enchufados.

Cada fila de la vista general de dispositivos representa un slot. Las distintas columnas muestran la información más importante relacionada con cada slot:

Columna	Significado
Módulo	Denominación del módulo, editable a discreción
Slot	Número del slot
Dirección E	Dirección de entrada, editable
Dirección S	Dirección de salida, editable
Tipo	Denominación del módulo en el catálogo
Referencia	Referencia del módulo
Firmware	Versión de firmware del módulo
Comentario	Comentario opcional

Consulte también

Vista de dispositivos (Página 493)

Área de módulos no enchufados

En algunos casos, los módulos de una configuración de hardware no se asignan temporalmente a ningún slot. Estos módulos no enchufados se desplazan al área de módulos no enchufados, un área específica dentro de la vista de dispositivos.

Inclusión de módulos en el área

Al área de módulos no enchufados se desplazan automáticamente los módulos que, p. ej., debían asignarse a un dispositivo mediante una acción de copia pero el rack correspondiente no dispone de ningún slot libre compatible.

En determinadas circunstancias, los módulos se incluyen automáticamente en el área de módulos no enchufados:

- Un módulo se arrastra hasta un dispositivo en la vista de redes pero el rack no dispone de ningún slot libre compatible.
- Un módulo se desplaza o copia del rack, catálogo de hardware o árbol del proyecto directamente en el área de módulos no enchufados en la vista de dispositivos.

Los CPs y FMs que ocupan un recurso de red pueden desplazarse al área de módulos no enchufados, pero pierden sus recursos de red ya asignados.

También puede agregar módulos al área de módulos no enchufados, por ejemplo mediante arrastrar y soltar. Para ello el área debe estar abierta.

Uso del área de módulos no enchufados

El área de módulos no enchufados se abre con el botón correspondiente.

El área de módulos no enchufados está en la vista de dispositivos.

El área de módulos no enchufados se abre con el botón correspondiente de la barra de herramientas de la vista de dispositivos (Página 493).

Icono	Significado
	Abrir el área de módulos no enchufados

Nota

Para liberar slots, desplace módulos de la configuración al área e inserte los módulos deseados del área en los slots que han quedado libres.

De este modo se retiran de la configuración los módulos que ya están parametrizados sin necesidad de borrarlos.

Tratamiento de módulos en el área

Los módulos del área se rigen por las reglas siguientes:

- Los módulos están en la carpeta "Módulos locales" del dispositivo correspondiente en el árbol del proyecto.
- Los módulos conservan todos los ajustes y parámetros que se han realizado hasta el momento.

- Los módulos no se tienen en cuenta al realizar la carga en el sistema de destino, por lo que no se comprueba la coherencia de los módulos en el área de módulos no enchufados.
- En el menú contextual se puede, p. ej., copiar, cortar o eliminar los módulos.

8.1.2.2 Configurar dispositivos individuales

Seleccionar una CPU

Introducción

En la vista de redes se selecciona una CPU del catálogo de hardware y se crea junto con un rack. Los módulos deseados se arrastran del catálogo de hardware al dispositivo y se ordenan automáticamente en el rack.

Selección de los componentes en el catálogo de hardware

En el catálogo de hardware, todos los componentes se visualizan en forma de carpetas. Al abrir dichas carpetas, aparecerán las diferentes versiones de los componentes de hardware seleccionados junto con sus respectivas referencias.

Aquí se muestra a modo de ejemplo cómo crear una CPU con un rack en la vista de redes.

Requisitos

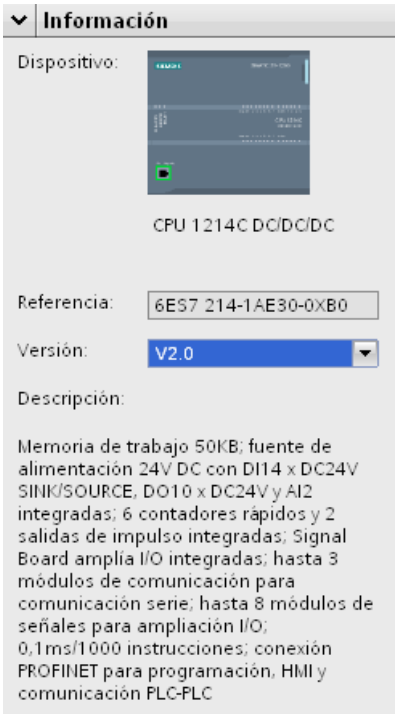
- El catálogo de hardware está abierto.
- Se encuentra en la vista de redes.

Procedimiento

Para seleccionar una CPU del catálogo de hardware, proceda del siguiente modo:

1. En el catálogo de hardware, navegue hasta la carpeta que contiene las CPUs deseadas.
2. Abra la carpeta con el tipo de CPU que busque. Aparecerán todas las referencias del tipo de CPU seleccionado.

3. Haga clic en la referencia de una CPU para obtener información acerca de la CPU seleccionada en la paleta "Información".



4. Cree la CPU y un rack. Para ello existen las posibilidades siguientes:
   - Arrastre la CPU desde el catálogo de hardware hasta la vista de redes con la función Drag & Drop.
   - Copie la CPU en la vista de redes con la función de copiar y pegar.
   - Haga doble clic en la entrada de la CPU del catálogo de hardware.

Consulte también

Examinar el catálogo de hardware (Página 512)

Agregar un dispositivo a la configuración de hardware (Página 520)

Insertar un módulo en un rack (Página 523)

Trabajar con racks (Página 514)

Crear una CPU sin especificar (Página 521)

Agregar un dispositivo a la configuración de hardware

Introducción

Para agregar un dispositivo conectable en red a la configuración de hardware existen las posibilidades siguientes en la vista de redes y topológica:

- Comando "Agregar nuevo dispositivo" en el árbol del proyecto
- Doble clic en el dispositivo en el catálogo de hardware
- Drag & Drop del catálogo de hardware a la vista de redes o topológica:
  - Entrada de texto de la paleta "Catálogo"
  - Gráfico preliminar de la paleta "Información"
- Comando "Insertar > Dispositivo" de la barra de menús de la vista de redes o topológica
- Menú contextual de un dispositivo en el catálogo de hardware para "Copiar" y "Pegar"

Junto con el dispositivo nuevo se crea un rack adecuado. El dispositivo seleccionado se inserta en el primer slot admisible del rack.

Independientemente del modo elegido, el dispositivo agregado será visible tanto en el árbol del proyecto como en la vista de redes o topológica del editor de hardware y redes.

Agregar un dispositivo desde el árbol del proyecto

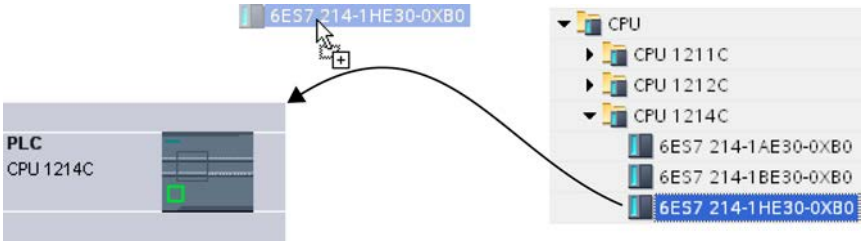
Para agregar un dispositivo a la configuración de hardware desde el árbol del proyecto, proceda del siguiente modo:

1. Haga clic en el comando "Agregar nuevo dispositivo" en el árbol del proyecto.
   Se abrirá el cuadro de diálogo "Agregar nuevo dispositivo".
2. Visualice el dispositivo deseado en la estructura de árbol:
   - Navegue hasta el dispositivo deseado en la estructura de árbol.
   - Introduzca un nombre en el campo de entrada.
3. Seleccione el dispositivo deseado en la estructura de árbol.
   En la parte derecha del cuadro de diálogo aparece información relacionada con el dispositivo seleccionado.
4. En caso necesario, ajuste la versión de firmware a través de la lista desplegable del cuadro de diálogo.
5. Seleccione la casilla de verificación "Abrir la vista de dispositivos" si desea pasar a la misma después de agregar el dispositivo.
   Allí podrá continuar inmediatamente con la configuración del dispositivo y la ocupación del rack.
6. Haga clic en "Aceptar" para agregar el dispositivo seleccionado.
   Se cerrará el cuadro de diálogo.

Agregar un dispositivo desde el catálogo de hardware

Para agregar un dispositivo a la configuración de hardware desde el catálogo de hardware, proceda del siguiente modo:

1. Abra la vista de redes o la vista topológica.
2. Abra el catálogo de hardware.
3. Navegue hasta el dispositivo deseado en el catálogo de hardware.
4. Seleccione el dispositivo deseado haciendo clic con el ratón.
5. En caso necesario, ajuste la versión de firmware a través de la lista desplegable del catálogo de hardware.
6. Arrastre el dispositivo a la vista de redes o topológica con Drag & Drop.



Ahora ha colocado el dispositivo en la vista de redes o topológica. El rectángulo mostrado (también "estación") simboliza el dispositivo insertado con su rack y, dado el caso, los módulos subordinados. Haciendo doble clic en el dispositivo o la estación se abre la vista de dispositivos, donde aparece el rack nuevo con el dispositivo insertado. En los pasos siguientes se configura el dispositivo en la vista de dispositivos y se asignan módulos al rack.

Consulte también

Vista de redes (Página 491)

Crear una CPU sin especificar (Página 521)

Vista topológica (Página 496)

Crear una CPU sin especificar

Introducción

Si todavía no se ha seleccionado ninguna CPU pero ya se empieza a programar o debe utilizarse un programa existente, existe la posibilidad de utilizar una CPU sin especificar. En las CPU sin especificar también es posible realizar algunos ajustes. Las posibilidades de ajuste se limitan a los parámetros que son comunes en todas las CPU de la misma familia.

Crear una CPU sin especificar en la vista de portal

Para crear una CPU sin especificar en la vista de portal, proceda del siguiente modo:

1. Haga clic en una de las posibilidades siguientes:
   – "Dispositivos y redes > Agregar nuevo dispositivo"
   – "Programación PLC" > botón "Dispositivo"
2. Seleccione para una familia de dispositivos una CPU sin especificar de la estructura de árbol del cuadro de diálogo "Agregar nuevo dispositivo".
3. Haga clic en "Agregar".

Se crea una CPU sin especificar y se abre su vista de dispositivos.

Otras posibilidades para crear CPUs sin especificar

En la vista del proyecto, las CPUs sin especificar se crean igual que las CPUs especificadas:

- En el árbol del proyecto con el botón "Agregar nuevo dispositivo"
- En la Task Card "Catálogo de hardware"

Estos métodos también permiten crear varias CPUs sin especificar.

Especificar CPUs sin especificar

Existen dos posibilidades para especificar CPUs sin especificar:

- Asigne una CPU existente a una CPU sin especificar mediante el cambio de módulo (Página 528) utilizando la función Drag & Drop desde el catálogo de hardware.
- Con la CPU sin especificar seleccionada, elija el comando de menú "Online > Detección de hardware" y asígnele una CPU determinada online. Para tal fin, adjudique una dirección IP con el botón "Agregar dirección para PG/PC".

Nota

Si desea pasar a online tras realizar la detección de hardware, deberá cargar primero la configuración detectada en su proyecto, pues de lo contrario se puede producir un error debido a configuraciones incoherentes. Para la CPU del proyecto y la CPU realmente existente, si bien las referencias son idénticas tras la detección de hardware, pero no lo son sus parámetros. Los parámetros de la CPU del proyecto tienen valores predeterminados, y los parámetros de la CPU realmente existente, los valores ajustados por el usuario.

Consulte también

Seleccionar una CPU (Página 518)

Agregar un dispositivo a la configuración de hardware (Página 520)

Insertar un módulo en un rack

Introducción

Después de haber agregado dispositivos del catálogo de hardware en la configuración desde la vista de redes, es posible asignar módulos a dichos dispositivos. Existen las posibilidades siguientes para insertar un módulo en un rack en la vista de dispositivos:

- Si hay un slot válido libre, hacer doble clic en un módulo del catálogo de hardware.
- Drag&Drop del catálogo de hardware a un slot válido libre en el área gráfica o de tabla:
  - Entrada de texto de la paleta "Catálogo"
  - Gráfico preliminar de la paleta "Información"
- "Copiar" del menú contextual de un módulo en el catálogo de hardware y "Pegar" del menú contextual en un slot válido libre en el área gráfica o de tabla.

Para acceder a la vista de dispositivos desde la vista de redes, haga doble clic en un dispositivo o estación en la vista de redes o utilice la ficha Vista de dispositivos. En la vista de dispositivos hay una representación del dispositivo seleccionado dentro de un rack. La representación gráfica del rack en el software se corresponde con la instalación real, es decir, hay tantos slots visibles como en la instalación real.

Nota

También es posible arrastrar un módulo hasta un rack en la vista de redes. Para ello, la función de filtro del catálogo de hardware debe estar desactivada. El módulo se insertará automáticamente en un slot permitido y libre. Si no hay ningún slot disponible, el módulo se desplazará al área de módulos no enchufados (Página 516).

Ocupación de un rack

Disponga los módulos en un rack siguiendo las reglas para slots correspondientes.

Después de insertar un módulo en un rack que ya tiene una CPU insertada se comprueban automáticamente las áreas de direcciones para que no se asignen por duplicado. De este modo, cada módulo tiene un área de direcciones válida una vez insertado. Para ello, los esclavos DP y dispositivos IO deben estar conectados en red con una CPU a través del correspondiente sistema maestro DP o sistema IO.

Requisitos

- Se encuentra en la vista de dispositivos.
- El catálogo de hardware está abierto.

Agregar un módulo desde el catálogo de hardware

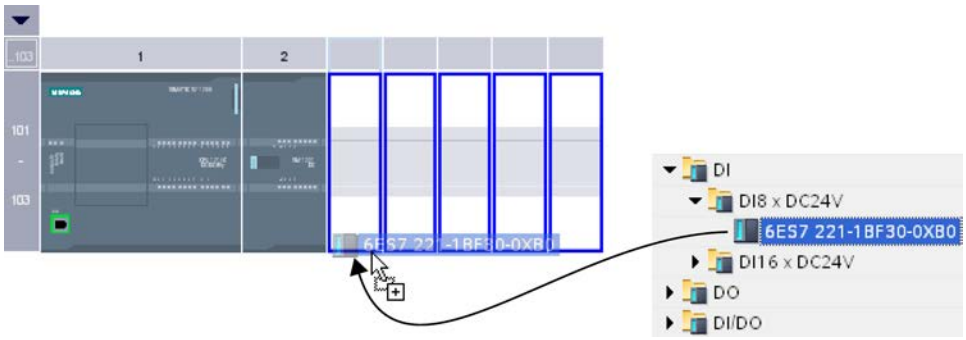
La inserción de un módulo del catálogo de hardware en un rack se muestra siguiendo el ejemplo de un módulo de señales digitales. Proceda del siguiente modo:

1. Navegue hasta el módulo deseado en el catálogo de hardware.

   Nota

   Si se activa la función de filtro del catálogo de hardware, sólo aparecerán los módulos compatibles con el tipo de dispositivo seleccionado.

2. Seleccione el módulo deseado.
3. En caso necesario, ajuste la versión de firmware a través de la lista desplegable del catálogo de hardware.
4. Arrastre el módulo de señales hasta un slot libre del rack con la función Drag & Drop.



Ahora se ha insertado el módulo de señales digitales en un slot del rack. Proceda del mismo modo con los demás módulos.

Encima de los módulos insertados se visualiza su nombre. La rotulación del módulo se activa y desactiva mediante "Ver > Mostrar rotulación del módulo" de la barra de menús.

Insertar módulo

Los módulos también pueden insertarse con Drag & Drop entre otros módulos ya insertados. Para ello, arrastre un módulo hasta que esté entre dos módulos insertados manteniendo pulsado el botón del ratón.

Aparece una marca de inserción. Al soltar el puntero del ratón, todos los módulos insertados a la derecha de la marca de inserción se desplazarán un slot a la derecha. Si sobrara un módulo, se desplazaría al área de módulos no enchufados. El módulo nuevo se insertará en el slot que ha quedado libre.

Consulte también

Vista de dispositivos (Página 493)

Área de módulos no enchufados (Página 516)

Reglas generales para los slots (Página 511)

Reglas para módulos (S7-300) (Página 898)

Reglas para módulos (S7-400) (Página 900)

Borrar un componente de hardware

Los componentes de hardware se borran de diferentes maneras. Los componentes de hardware borrados se eliminan del sistema y las direcciones ocupadas vuelven a estar disponibles.

Reglas

- Es posible borrar CPUs o módulos del rack y del área de módulos no enchufados.
- Si se borra un rack en la vista de dispositivos, los componentes de hardware insertados se desplazan al área de módulos no enchufados.

Procedimiento

Para borrar un componente de hardware, proceda del siguiente modo:

1. Seleccione el componente de hardware que desea borrar.
   - Vista de redes: en la vista gráfica o en la vista general de redes seleccione dispositivos o componentes de hardware relevantes para la red.
   - Vista de dispositivos: en la vista gráfica o en la vista general de dispositivos seleccione racks o módulos de los racks o del área de módulos no enchufados.
   - Vista topológica: en la vista gráfica o en la vista topológica general de redes dispositivos o componentes de hardware con interfaces Ethernet.
   - Árbol del proyecto: seleccione los dispositivos o componentes de hardware individuales de la estructura de árbol.
2. Elija "Borrar" del menú contextual o pulse <Supr>.
   Si el comando de menú "Borrar" está atenuado, significa que la selección contiene al menos un componente que no se puede borrar.

Los componentes de hardware seleccionados se borran.

Nota

Al borrar componentes de hardware es posible que el proyecto presente incoherencias, p. ej. violación de reglas para slots. Con la comprobación de coherencia se notificarán las incoherencias. Elimínelas realizando las acciones pertinentes, p. ej. restableciendo la observación de las reglas para slots.

Consulte también

Comandos de teclado: editar objetos (Página 508)

Copiar un componente de hardware

Los componentes de hardware se copian en la vista de dispositivos o redes. Los componentes de hardware copiados se guardan en el portapapeles y pueden copiarse en otro lugar desde este último. Las estaciones copiadas se insertan como nuevas en la vista de redes, los dispositivos y módulos copiados pueden insertarse en los racks existentes en la vista de redes y dispositivos.

Reglas

- Se pueden copiar tanto objetos individuales como varios objetos al mismo tiempo.
- Es posible copiar módulos insertados del rack y del área de módulos no enchufados.
- Sólo es posible copiar dispositivos y módulos en slots válidos y libres observando las reglas para slots.
- Los racks con una CPU insertada no pueden copiarse individualmente, sino sólo completos junto con todos los componentes de hardware insertados.

Procedimiento

Para copiar un componente de hardware, proceda del siguiente modo:

1. Seleccione el componente de hardware que desea copiar.
   - Vista de dispositivos: seleccione el módulo en un rack o en el área de módulos no enchufados.
   - Vista de redes: seleccione la estación o el componente de hardware relevante para la vista de redes.
   - Árbol del proyecto: seleccione la estación o el módulo.
2. Elija "Copiar" del menú contextual o pulse <Ctrl+C>.
   Si el comando de menú "Copiar" está atenuado, significa que la selección contiene al menos un componente que no se puede copiar.
3. Seleccione el lugar en el que debe insertarse el contenido del portapapeles.
   - Vista de dispositivos: seleccione un slot libre del rack o el área de módulos no enchufados.
   - Vista de redes: seleccione una estación para insertar dispositivos o módulos o mueva el puntero del ratón hasta un lugar libre de la vista de redes para insertar una estación copiada o un componente de hardware relevante para la vista de redes.
4. Elija "Pegar" del menú contextual o pulse <Ctrl+V>.
   Si el comando de menú "Pegar" está atenuado, significa que el portapapeles está vacío o que contiene al menos un componente que no se puede pegar en este lugar.

El objeto seleccionado se insertará en el lugar elegido.

Si se ha seleccionado una estación en la vista de redes para insertar un módulo, dicho módulo se insertará en el primer slot válido y libre. Si no hay ninguno, el objeto se insertará en el área de módulos no enchufados.

Nota

También es posible copiar un módulo de un dispositivo a otro:

Para ello, copie un módulo en el editor de hardware y redes, seleccione otro dispositivo en la vista de redes o en la lista desplegable de la vista de dispositivos e inserte el módulo.

En la vista de dispositivos, el objeto copiado puede insertarse directamente en un slot o en el área de módulos no enchufados. Si el objeto copiado se agrega a un dispositivo o estación en la vista de redes, se insertará en el primer slot libre.

Si no hay ningún slot disponible para el objeto, se insertará automáticamente en el área de módulos no enchufados (Página 516).

Nota

También es posible copiar cualquier componente de hardware directamente con <Ctrl> y Drag & Drop.

Consulte también

Comandos de teclado: editar objetos (Página 508)

Desplazar un componente de hardware

Los componentes de hardware se desplazan en la vista de dispositivos o redes.

Reglas

- Es posible desplazar dispositivos y módulos del rack y del área de módulos no enchufados observando las reglas para slots.
- Los CPs no se pueden desplazar en la vista de redes. El CP se inserta en un slot libre y válido del dispositivo de destino. Si no hay ningún slot libre, el CP que se va a insertar se desplaza al área de módulos no enchufados.
- Las CPUs y los módulos de cabecera esclavos pueden desplazarse entre los dispositivos de la vista de redes e incluso dentro del rack, según sea el tipo de CPU.

Nota

Los CPs desplazados se desconectan de su red, pero conservan los parámetros de red y la dirección. Si el CP se conecta de nuevo a la red pero su dirección se ha utilizado en otro lugar, es posible asignar una nueva dirección unívoca al CP desde un cuadro de diálogo.

Procedimiento

Para desplazar un componente de hardware, proceda del siguiente modo:

1. Seleccione el componente de hardware que desea desplazar.
   - Vista de dispositivos: Seleccione el módulo en un rack o en el área de módulos no enchufados.
   - Vista de redes: seleccione el componente de hardware relevante para la vista de redes.
2. Elija "Cortar" del menú contextual o pulse <Ctrl+X>.
   Si el comando de menú "Cortar" está atenuado, significa que la selección contiene al menos un componente que no se puede cortar.
3. Seleccione el lugar al que debe desplazarse el objeto cortado.
   - Vista de dispositivos: Seleccione un slot libre del rack o el área de módulos no enchufados.
   - Vista de redes: seleccione una estación para insertar dispositivos o módulos.
4. Elija "Pegar" del menú contextual o pulse <Ctrl+V>.
   Si el comando de menú "Pegar" está atenuado, significa que el portapapeles está vacío o que contiene al menos un componente que no se puede pegar en este lugar.

El componente de hardware seleccionado se desplazará a la ubicación de destino. Si el componente de hardware desplazado es un objeto conectado en red, se desacoplará de la red.

Nota

También es posible desplazar cualquier componente de hardware directamente con Drag & Drop.

Consulte también

Comandos de teclado: editar objetos (Página 508)

Sustituir un componente de hardware

Los componentes de hardware se pueden sustituir. Esto permite, por ejemplo, reemplazar CPUs no especificadas (Página 521) por CPUs disponibles del catálogo de hardware.

Reglas

Sólo está permitido sustituir componentes de hardware que soporten el cambio de módulo y sean compatibles entre sí.

Procedimiento

Para sustituir módulos, proceda del siguiente modo:

1. Seleccione el módulo que desea sustituir.
2. Abra el menú contextual:
   - Si la entrada "Sustituir dispositivo" está activada, se podrán sustituir los módulos.
   - Si la entrada "Sustituir dispositivo" está desactivada, no se podrán sustituir los módulos.
3. Haga clic en la entrada "Sustituir dispositivo" del menú contextual. Se abre el cuadro de diálogo "Sustituir dispositivo".
4. En "Dispositivo nuevo" de la estructura de árbol seleccione el módulo que reemplazará al módulo actual.
5. Haga clic en "Aceptar".

El módulo existente será reemplazado por el nuevo.

Otra posibilidad es arrastrar directamente un módulo desde el catálogo de hardware hasta el módulo que debe sustituirse con la función Drag & Drop. Un símbolo en el puntero del ratón indicará si es posible sustituir los dos módulos.

Editar las propiedades y los parámetros

Una vez insertados los componentes de hardware en el rack, existe la posibilidad de editar sus propiedades predeterminadas en la vista de redes o dispositivos, p. ej. parámetros o direcciones.

Requisitos

Se encuentra en la vista de dispositivos.

Nota

Las propiedades y los parámetros también se editan en la vista de redes. Desde la vista gráfica de redes se accede a la estación y a los componentes de hardware relevantes para la red. Desde el área de tabla de la vista de redes se accede a los módulos y componentes de hardware que no aparecen en la vista gráfica.

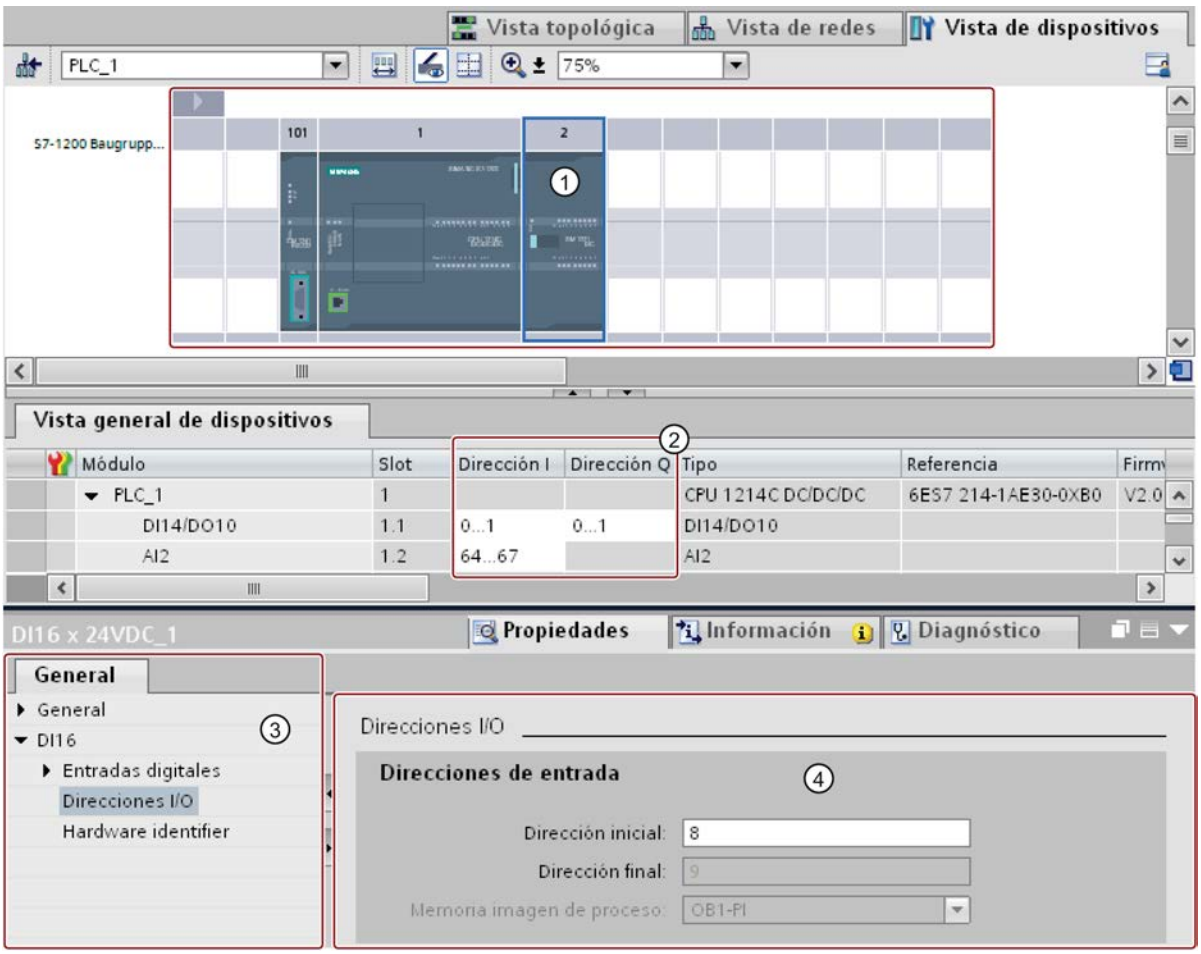
Procedimiento

Para modificar las propiedades y los parámetros de los componentes de hardware, proceda del siguiente modo:

1. En la representación gráfica, seleccione la CPU, el módulo, el rack o la interfaz que desee editar.
2. Edite los ajustes del objeto seleccionado:
   - En la representación de tabla se modifican las direcciones y los nombres, p. ej.
   - En "Propiedades" de la ventana de inspección existen las posibilidades de ajuste más variadas.

Tenga en cuenta que los módulos sólo pueden parametrizarse por completo cuando están asignados a una CPU. Por ello, los módulos de interfaz PROFIBUS o PROFINET deben conectarse en red primero con la CPU o con un módulo de comunicación que esté insertado centralmente, de modo que se forme un sistema maestro o un sistema IO. Sólo entonces será posible editar las direcciones de los componentes que estén insertados de forma descentralizada, por ejemplo.

Ejemplo de la modificación de ajustes



① Selección de un módulo
② Posibilidad de edición de direcciones en la vista general de dispositivos
③ Posibilidades de selección en la ventana de inspección
④ Posibilidad de edición de direcciones en la ventana de inspección

Consulte también

Ventana de inspección (Página 501)

Direcciones de entrada y salida en la vista de direcciones

Introducción

En la vista de direcciones podrán mostrarse, en forma de tabla, las direcciones de entrada y salida actualmente utilizadas. El sinóptico de direcciones se encuentra en "Propiedades" de la CPU, dentro de la ventana de inspección.

Estructura de la vista de direcciones

Podrá ajustar qué objetos deberán mostrarse en la vista de direcciones a través de distintas casillas opcionales:

- Entradas: Visualización de direcciones de entrada
- Salidas: Visualización de direcciones de salida
- Huecos en las direcciones: Visualización de áreas libres de dirección
- Slot: Visualización del número de slot

Normalmente, en la vista de direcciones se muestra la siguiente información:

Encabezado de la tabla	Significado
Tipo	Indica si el área de direcciones es de entrada o salida.
Direc. de	Dirección inicial del área de direcciones.
Direc. hasta	Dirección final del área de direcciones.
Módulo	Módulo desde el que se utilizará el área de direcciones.
MIPP	N.º de la memoria imagen parcial de proceso Indica si el OB cíclico está en la memoria imagen parcial de proceso.
OB	Bloque de organización asignado a la memoria imagen de proceso. Esta columna no está disponible en todas las CPUs.
DP	Número de sistema maestro. De esta forma puede determinarse qué esclavos están asignados a un maestro. El valor entre paréntesis indica la dirección PROFIBUS del componente de hardware.
PN	Número del sistema IO. El valor entre paréntesis representa el número de dispositivo del componente de hardware.
Rack	Número del rack donde se encuentra insertado el componente hardware.
Slot	Número del slot donde se encuentra insertado el componente hardware en el rack.

Consulte también

Definir las direcciones de entrada y salida (Página 972)

Actualizar versión del módulo

Explicación de términos

A continuación se explican con detalle los términos "versión de módulo" y "versión de firmware".

- Versión de módulo: es la versión del software de configuración de la que se deriva la descripción del módulo.
  Ej.: V11.0.0.0
- Versión de firmware: es la versión del firmware del módulo parametrizado offline
  Ej.: V2.0

Requisitos

- Ha creado una configuración de dispositivos.
- Posteriormente, ha instalado una actualización o un paquete opcional, por ejemplo. Debido a dicha instalación se ha actualizado la versión de un tipo de módulo como mínimo en el catálogo de hardware, siendo dicha versión incompatible con la anterior.
- Ha utilizado estos módulos en la configuración de dispositivos y desea aprovechar las propiedades modificadas o agregadas.

Procedimiento

Realice los pasos siguientes para cada uno de los tipos de módulo afectados.

1. Seleccione un módulo afectado en la vista de dispositivos.
2. Vaya a "Propiedades > General > Información de catálogo" en la ventana de inspección. Haga clic en el botón de comando "Actualizar versión del módulo".
3. En la ventana que aparece, especifique si desea actualizar solo la versión del módulo seleccionado o la de todos los módulos de este tipo en el proyecto actual.

Resultado

En el proyecto actual se reemplazan los módulos seleccionados por los mismos módulos con una versión actualizada.

¿En qué casos no es necesario actualizar la versión del módulo?

La actualización de la versión del módulo es innecesaria en los casos siguientes:

- No se quieren aprovechar las propiedades modificadas o agregadas de los módulos.
- Se abre un proyecto existente cuya versión del software de configuración es más actual que la versión con la que se ha creado el proyecto, y el sistema lleva a cabo una conversión automática del proyecto, p. ej., de TIA Portal V12 a V13. En este caso se adaptan automáticamente todas las versiones de módulo no actualizadas.

8.1.2.3 Comparar dispositivos

Principios básicos de la comparación de dispositivos

Funcionamiento

Existe la posibilidad de comparar los componentes de hardware de dos dispositivos con el fin de determinar posibles diferencias. Para ello puede realizar una comparación offline/offline. Los dispositivos que se compararán pueden proceder de un proyecto o de proyectos diferentes.

Es posible comparar tanto periferia centralizada como descentralizada. Los dispositivos que se compararán se pueden asignar automática o manualmente. La asignación automática de periferia centralizada se realiza a partir del número de slot. En el caso de la periferia descentralizada, la asignación automática puede realizarse conforme a los siguientes criterios:

- Asignación por dirección/ID: la asignación se realiza a través de las direcciones o las IDs de los dispositivos. Este criterio es adecuado para la comparación de dispositivos que se encuentran dentro de un proyecto.
- Asignación por nombre: la asignación se basa en los nombres de los dispositivos. Este criterio es adecuado para la comparación de dispositivos que se encuentran en proyectos diferentes.

El usuario puede decidir el tipo de asignación por sí mismo o dejar que lo haga el sistema. En este último caso, el sistema elije la asignación en función del contexto.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Resumen del editor de comparación (Página 375)

Realizar una comparación de dispositivos (Página 533)

Realizar una comparación de dispositivos

Procedimiento

Para realizar una comparación de dispositivos, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Abra la ficha "Hardware".

4. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha. Todos los objetos disponibles de los dispositivos seleccionados se muestran en función de los ajustes realizados en la ficha "Hardware" del editor de comparación y se lleva a cabo una comparación automática. Los símbolos del editor de comparación indican el estado de los objetos.
5. Para modificar el criterio de asignación, haga clic en la barra de herramientas en la flecha del botón "Mostrar criterios de asignación disponibles". Seleccione el criterio de asignación que desea utilizar.
6. Si desea realizar una comparación manual, haga clic sobre el área de estado en el botón para conmutar entre la comparación automática y manual. Seguidamente, seleccione los objetos que desea comparar.
   Se muestra la comparación de propiedades. Los símbolos indican el estado de los objetos.

Consulte también

Principios básicos de la comparación de dispositivos (Página 533)

Resumen del editor de comparación (Página 375)

8.1.3 Configurar redes

8.1.3.1 Conectar dispositivos en red

Comunicación y redes

Comunicación entre dispositivos

La base para cualquier tipo de comunicación es siempre una red previamente configurada. La configuración de una red crea los requisitos necesarios para la comunicación:

- Todos los dispositivos de una red disponen de direcciones unívocas
- Comunicación de los dispositivos con propiedades de transferencia coherentes

Configuración de una red

Para configurar redes son necesarios los siguientes pasos:

- Conectar los dispositivos a una subred
- Definir las propiedades y los parámetros de cada subred
- Definir las propiedades de los nodos para cada módulo conectado
- Cargar los datos de configuración en los dispositivos para proporcionar a las interfaces los ajustes resultantes de la configuración de red
- Documentar la configuración de red

Para la Open User Communication, la creación y configuración de una subred es soportada por la parametrización de la conexión.

Relación entre la configuración de redes y el proyecto

Dentro de un proyecto se gestionan las subredes junto con sus propiedades. Las propiedades resultan básicamente de los parámetros de red ajustables y del número y las propiedades de comunicación de los nodos conectados.

Los nodos que hay que conectar en red deben estar dentro del mismo proyecto.

Nombre e ID de la subred

Las subredes se identifican unívocamente dentro del proyecto con un nombre y una ID. La ID de subred está almacenada en todos los componentes con interfaces conectables en red. De este modo, es posible asignar componentes a una subred de forma unívoca incluso después de cargarlos en un proyecto.

Posibilidades de la conexión en red

En el proyecto pueden crearse y conectarse en red dispositivos con componentes aptos para la comunicación. Para conectar los dispositivos en red existen básicamente las posibilidades siguientes:

- Conectar entre sí las interfaces de los componentes aptos para la comunicación. De este modo se creará una subred nueva adecuada al tipo de interfaz.
- Conectar la interfaz de los dispositivos aptos para la comunicación con una subred nueva o ya existente.
- Crear una conexión de la Open User Communication. Al parametrizar la conexión de la Open User Communication se creará automáticamente una subred entre los interlocutores.
- Configurar conexiones mediante la configuración gráfica de conexiones; las conexiones en red que faltan se detectan y se crean de forma automática o con la ayuda de un cuadro de diálogo.

Debido a las diferentes tareas de los dispositivos o a la extensión de la instalación es posible que se requieran varias subredes. Dichas subredes se gestionan en un proyecto.

Conectar dispositivos en red en la vista de redes

Posibilidades

En la vista gráfica de redes se conserva la vista general de las subredes existentes en toda la instalación del proyecto. El área de tabla de la vista general de redes sirve de ayuda.

Para conectar en red la interfaz de un componente apto para la comunicación existen distintas posibilidades en la configuración, según sea la situación de partida. A continuación se describen estos procedimientos:

- Crear una subred individual
- Crear varias subredes al mismo tiempo

- Conectar dos dispositivos de destino mediante una subred nueva
- Conectar los dispositivos a una subred existente
- Seleccionar una subred existente de la lista
- Conexión automática en red durante la configuración de la conexión:
  Consulte también: Crear una conexión: sinopsis (Página 616)

Las posibles situaciones de partida son:

- Todavía no existe ninguna subred adecuada.
- La subred a la que debe conectarse el componente ya existe.

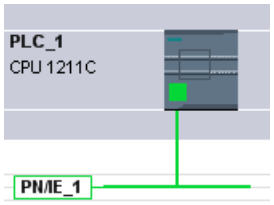
Procedimiento para crear una subred individual

Para crear una subred y conectarla a una interfaz, proceda del siguiente modo:

1. Seleccione la interfaz de una CPU / un CP.
2. Elija el comando "Crear subred" del menú contextual de la interfaz.

La interfaz seleccionada se conecta a una subred nueva. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

La figura siguiente muestra una interfaz con una línea de conexión que sale hacia una subred:



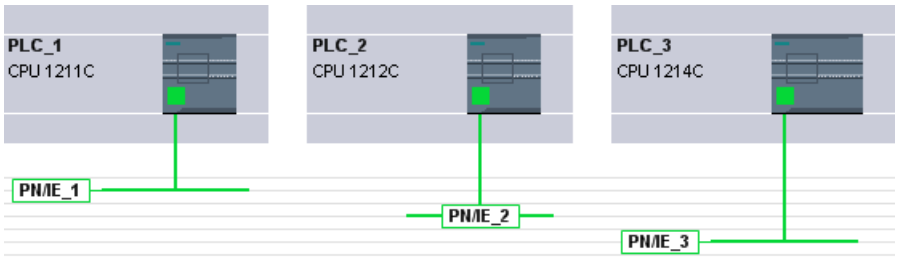
Procedimiento para crear varias subredes al mismo tiempo

Para crear varias subredes al mismo tiempo, proceda del siguiente modo:

1. Seleccione varias interfaces haciendo clic con el ratón y pulsando simultáneamente la tecla <Ctrl>.
2. Elija el comando "Crear subred" del menú contextual de la interfaz.

Cada interfaz seleccionada se conectará a una subred nueva. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

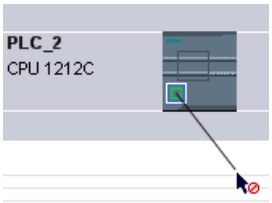
La figura siguiente muestra varias subredes que se han creado mediante una selección múltiple de las interfaces:



Procedimiento para conectar dos dispositivos de destino mediante una subred nueva

Para conectar una interfaz con otro dispositivo a través de una subred que todavía no existe, proceda del siguiente modo:

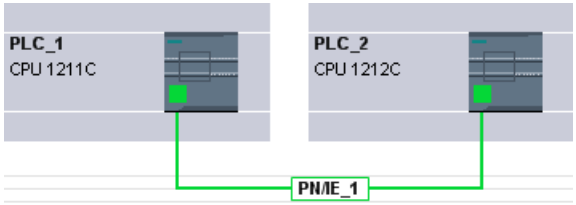
1. Coloque el cursor del ratón en la interfaz que debe conectarse de un componente apto para la comunicación.
2. Haga clic con el botón izquierdo del ratón y manténgalo pulsado.
3. Mueva el cursor del ratón.
   Ahora, el cursor indica el modo "Conectar a la red" con el símbolo correspondiente. Al mismo tiempo, el cursor del ratón muestra el símbolo de bloqueo, que desaparece cuando se pasa por encima de una posición de destino válida.



4. Arrastre el cursor del ratón hasta la interfaz del dispositivo de destino. El botón izquierdo del ratón puede permanecer pulsado o soltarse.
5. Ahora, suelte el botón izquierdo del ratón o vuelva a hacer clic con el mismo.

Se creará una subred nueva. Ahora, las interfaces están conectadas a través de la subred nueva. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

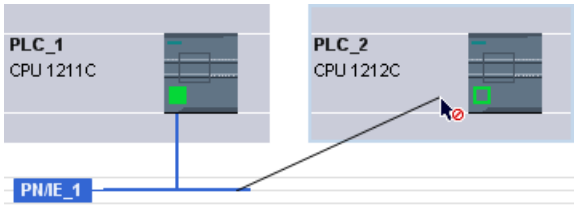
La figura siguiente muestra dos dispositivos conectados en red:



Procedimiento para conectar los dispositivos a una subred existente

Para conectar una interfaz a una subred ya existente, proceda del siguiente modo:

1. Coloque el cursor del ratón en la interfaz que debe conectarse de un componente apto para la comunicación o en la subred existente.
2. Haga clic con el botón izquierdo del ratón y manténgalo pulsado.
3. Mueva el cursor del ratón.
   Ahora, el cursor indica el modo "Conectar a la red" con el símbolo correspondiente. Al mismo tiempo, el cursor del ratón muestra el símbolo de bloqueo, que desaparece cuando se pasa por encima de una posición de destino válida.
4. Arrastre el cursor del ratón hasta la subred existente o hasta la interfaz que debe conectarse en red. El botón izquierdo del ratón puede permanecer pulsado o soltarse.



5. Ahora, suelte el botón izquierdo del ratón o vuelva a hacer clic con el mismo.

Resultado:

La interfaz y la subred seleccionada están ahora conectadas. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

Procedimiento para seleccionar una subred existente de la lista

Para conectar una interfaz a una subred ya creada, proceda del siguiente modo:

1. Seleccione la interfaz de una CPU.
2. Elija el comando "Asignar a la subred nueva" del menú contextual de la interfaz.
   Aparece una lista de selección con las subredes existentes.
3. Seleccione una subred de la lista.

La interfaz y la subred seleccionada están ahora conectadas. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

Área de tabla de la vista general de redes

Significado

El área de tabla de la vista general de redes complementa la vista gráfica de redes con las funciones siguientes:

- Ofrece información detallada sobre la estructura y parametrización de los dispositivos.
- Desde la columna "Subred" se conectan componentes aptos para la comunicación con subredes creadas.

Funciones básicas de las tablas

La vista general de redes soporta las siguientes funciones básicas para editar una tabla:

- Mostrar y ocultar las columnas de la tabla
  Nota: no es posible ocultar las columnas relevantes para la configuración.
- Optimizar el ancho de las columnas
- Ordenar la tabla
- Explicar el significado de una columna, fila o campo mediante tooltips

Conectar dispositivos en red en la vista de dispositivos

Conexión en red en la vista de dispositivos

La vista de dispositivos permite comprobar detalladamente y ajustar todos los parámetros de las interfaces y los componentes que corresponden a un dispositivo. Aquí también es posible asignar las interfaces a las subredes creadas en el proyecto.

Requisitos

- La subred a la que debe conectarse una interfaz ya está creada.
- Si la subred todavía no está creada, pase a la vista de redes y realice la conexión en red.

Procedimiento para conectar a una subred existente

Para conectar la interfaz a una subred ya creada, proceda del siguiente modo en la vista de dispositivos:

1. Seleccione todo el componente apto para la comunicación o bien la interfaz que debe conectarse en red.
   En la ventana de inspección aparecerán las propiedades de la interfaz o el componente seleccionado.
2. En la ventana de inspección, seleccione el grupo de parámetros correspondientes a la interfaz seleccionada, p. ej. el grupo de parámetros "Direcciones Ethernet" en una interfaz PROFINET.
3. Elija la subred que debe conectarse en la lista desplegable "Subred" de "Interfaz conectada en red con".

La interfaz y la subred seleccionada están ahora conectadas. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

Procedimiento para crear una subred nueva

Para crear una subred y conectarla a la interfaz, proceda del siguiente modo en la vista de dispositivos:

1. Seleccione todo el componente apto para la comunicación o bien la interfaz que debe conectarse en red.
   En la ventana de inspección aparecerán las propiedades de la interfaz o el componente seleccionado.
2. En la ventana de inspección, seleccione el grupo de parámetros correspondientes a la interfaz seleccionada, p. ej. el grupo de parámetros "Direcciones Ethernet" en una interfaz PROFINET.
3. Haga clic en el botón "Agregar nueva subred" en "Interfaz conectada en red con".

La interfaz se conectará a una subred nueva del tipo adecuado. Los parámetros de dirección de la interfaz se ajustan automáticamente de forma coherente.

Comprobar o modificar los parámetros de red o interfaz

Introducción

Para que los dispositivos conectados en red sean aptos para la comunicación, hay que configurar los parámetros siguientes:

- Parámetros de red
  Los parámetros de red identifican la red dentro de la configuración de la instalación, p. ej. mediante un nombre.
- Parámetros de interfaz
  Los parámetros de interfaz definen propiedades específicas de un componente apto para la comunicación. El ajuste coherente con los parámetros de red para las direcciones y las propiedades de transferencia se genera automáticamente.

Nota

Los parámetros de red e interfaz están configurados por defecto en la conexión en red de modo que la comunicación es posible en muchos casos sin tener que modificar parámetros.

Procedimiento para comprobar o modificar parámetros de red

Para comprobar o modificar parámetros de red, proceda del siguiente modo:

1. Vaya a la vista de redes.
2. Seleccione la subred en la vista de redes.
   En la ventana de inspección se muestran los parámetros de red en la ficha "Propiedades".
3. Compruebe y, dado el caso, modifique los parámetros de red en el grupo correspondiente.

Procedimiento para comprobar o modificar parámetros de interfaz

Los parámetros de interfaz se comprueban y modifican en la vista de redes y en la de dispositivos.

Para comprobar o modificar parámetros de interfaz, proceda del siguiente modo:

1. Vaya a la vista de redes o de dispositivos.
2. Seleccione la interfaz.
   En la ventana de inspección se muestran los parámetros de interfaz en la ficha "Propiedades".
3. Compruebe y, dado el caso, modifique los parámetros de interfaz en el grupo correspondiente.

Modificar conexiones en red

Introducción

Existe la posibilidad de eliminar la conexión de red de una interfaz o de asignarla a otra subred del mismo tipo.

Repercusiones

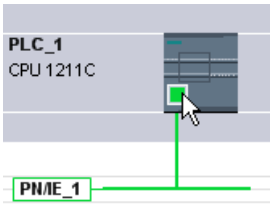
Según el modelo hay que distinguir entre:

- Eliminar la conexión de red de una interfaz
  Los parámetros configurados de la interfaz no se modificarán.
- Asignar la conexión de red a otra subred
  Si las direcciones de la subred asignada no son unívocas, es decir, ya existen, se convertirán automáticamente en unívocas.

Procedimiento para eliminar la conexión de red

Para eliminar la conexión de red de una interfaz, proceda del siguiente modo:

1. Seleccione la interfaz conectada en red.



2. Elija el comando "Deshacer la conexión con la subred" del menú contextual de la interfaz.

La conexión de red se borrará sin modificar las direcciones de la interfaz.

Las conexiones configuradas se conservan, aunque se marcan en rojo en la tabla de conexiones porque falta la conexión en red. Las conexiones especificadas siguen estando especificadas.

Consulte también

Conectar dispositivos en red en la vista de redes (Página 535)

Copiar, cortar o borrar subredes

Introducción

Es posible copiar subredes como objetos individuales, junto con dispositivos conectados en red o junto con otras redes.

De este modo se crean sin mayor esfuerzo, p. ej., configuraciones complejas que deben incluirse varias veces en el proyecto.

Repercusiones en la subred copiada

Las propiedades que deben ser asignadas de forma unívoca dentro del proyecto se reasignan en correspondencia a los objetos copiados durante el proceso de copia.

Para las subredes esto significa lo siguiente: la ID y el nombre de la subred se reasignan a la subred copiada.

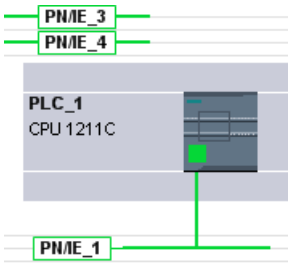
Las propiedades configuradas se aplican a la subred copiada.

Procedimiento para copiar una subred

Para copiar una o varias redes, proceda del siguiente modo:

1. Seleccione una o varias subredes.
2. En el menú contextual, elija el comando "Copiar".
3. En el menú contextual, elija el comando "Pegar".

Las subredes copiadas aparecen en el área superior de la vista de redes como subredes "huérfanas".



Procedimiento para copiar subredes con dispositivos conectados

Para copiar una o varias redes con dispositivos conectados, proceda del siguiente modo:

1. Seleccione una o varias subredes con los dispositivos conectados, p. ej. con la función de lazo.
2. En el menú contextual, elija el comando "Copiar".
3. En el menú contextual, elija el comando "Pegar".

Se crean copias completas de las subredes y los dispositivos conectados.

Las conexiones configuradas se aplican y se conservan en los dispositivos copiados. Las conexiones con dispositivos que no se han copiado se deshacen y quedan sin especificar.

Configuración de red MPI

Asignación de direcciones MPI

Para dispositivos con una interfaz MPI, todos los dispositivos de una subred deben tener una dirección distinta.

Las CPUs con dirección MPI se entregan con la dirección predeterminada MPI 2. Puesto que esta dirección sólo se puede utilizar una vez en la subred MPI, deberá modificarse la dirección predeterminada en las demás CPUs.

Para dispositivos con las referencias 6ES7 3xx-xxxxx-0AB0 rige lo siguiente:

Al planificar direcciones MPI para varias CPUs, deberán preverse "huecos en las direcciones MPI" para FMs y CPs con direcciones MPI propias para que las direcciones no se asignen por duplicado.

Los ajustes sólo deben cargarse a través de la red si todos los módulos de una subred poseen direcciones distintas y su estructura real coincide con la configuración de red creada.

Reglas para la asignación de direcciones MPI

- Las direcciones MPI deben asignarse en orden ascendente.
- La dirección MPI 0 debe reservarse para una PG.
- En una subred MPI pueden interconectarse hasta 126 dispositivos (direccionables), y hasta 8 dispositivos con una velocidad de transferencia de 19,2 kbits/s.
- Todas las direcciones MPI de una subred MPI deben ser distintas.

En los manuales de montaje de sistemas de automatización se encuentran más reglas sobre la configuración de una red.

Configuración de red PROFIBUS

Direcciones PROFIBUS

Reglas de la configuración de red

Todos los nodos de una subred deben tener direcciones PROFIBUS distintas.

Los ajustes sólo deben cargarse a través de la red si todos los módulos de una subred poseen direcciones distintas y su estructura real coincide con la configuración de red creada.

En la subred PROFIBUS pueden conectarse nodos que se comuniquen a través de conexiones configuradas o formen parte de un sistema maestro PROFIBUS DP.

Los siguientes capítulos ofrecen más información sobre la configuración de un sistema maestro DP.

Requisitos

La CPU 121xC es apta para PROFIBUS a partir de la versión de firmware 2.0.

Reglas para la asignación de direcciones PROFIBUS

- Las direcciones PROFIBUS deben asignarse en orden ascendente.
- La dirección PROFIBUS "0" debe reservarse para una programadora.
- Debe asignarse una dirección PROFIBUS unívoca en el rango de 0 a 126 para cada dispositivo en la red PROFIBUS o para cada maestro DP y cada esclavo DP en la red PROFIBUS.
- Existen módulos cuya dirección configurable más pequeña tiene que ser mayor que 1.
- Todas las direcciones PROFIBUS de una subred PROFIBUS deben ser distintas.

En los manuales de montaje de sistemas de automatización, p. ej. SIMATIC S7-1200, se encuentran más reglas sobre la configuración de una red.

Nota

Dirección PROFIBUS "0"

La dirección PROFIBUS "0" debe reservarse para una programadora que se conectará a la red PROFIBUS más adelante y por un tiempo limitado para fines de servicios.

Consulte también

Información importante sobre los parámetros de bus PROFIBUS (Página 545)

Información importante sobre los parámetros de bus PROFIBUS

Parámetros sintonizados entre sí

La subred PROFIBUS sólo funciona a la perfección si los parámetros del perfil de bus están sintonizados entre sí. Por esta razón, los valores predeterminados sólo deberían modificarse si se está familiarizado con la parametrización del perfil de bus para PROFIBUS.

Nota

En función del perfil de bus, los parámetros podrán ajustarse o no. Si los parámetros de bus no pueden ajustarse, aparecen atenuados. Siempre se visualizan los valores offline de los parámetros del bus, incluso si están conectados online con el sistema de destino.

Los parámetros visualizados sirven para toda la subred PROFIBUS y se explican brevemente a continuación.

Activar difusión cíclica de los parámetros de bus

Si, con la subred PROFIBUS desmarcada, está activada la casilla de verificación "Activar difusión cíclica de los parámetros de bus" en el apartado "Parámetros del bus" de la ventana de inspección, los módulos que soportan esta función enviarán cíclicamente los parámetros de bus durante el funcionamiento. De esta forma, por ejemplo, la PG puede conectarse sin problemas al PROFIBUS durante el funcionamiento.

Esta función debe desactivarse:

- En una subred PROFIBUS heterogénea (exactamente, en dispositivos de terceros conectados cuyo protocolo utilice el DSAP 63 para Multicast)
- en caso de funcionamiento equidistante (minimizar el ciclo de bus)

Parámetros de bus para el perfil de bus de las subredes PROFIBUS

Parámetros del bus	¿Ajustables?	Valores límite
Tslot_Init	Sí	Max. Tsdr + 15 <= Tslot_init <= 16.383 t_bit
Max. Tsdr	Sí	35 + 2*Tset + Tqui <= Max. Tsdr <= 1.023 t_bit
Min. Tsdr	Sí	11 t_bit <= Min. Tsdr <= MIN(255 t_bit, ... Max. Tsdr - 1, 34 + 2*Tset + Tqui)
Tset	Sí	1 t_bit <= Tset <= 494 t_bit
Tqui	Sí	0 t_bit <= Tqui <= MIN(31 t_bit, Min. Tsdr - 1)
Factor GAP	Sí	1 <= Factor GAP <= 100
Retry Limit	Sí	1 <= Retry Limit <= 15
Tslot	No	---
Tid2	No	Tid2 = Max. Tsdr
Trdy	No	Trdy = Min. Tsdr
Tid1	No	Tid1 = 35 + 2*Tset + Tqui
Ttr	Sí	256 t_bit <= Ttr <= 16.777.960 t_bit

Parámetros del bus	¿Ajustables?	Valores límite
Ttr típico	No	Este tiempo sólo tiene carácter informativo y no se transmite a los nodos.
Supervisión de respuesta		10 ms <= Supervisión de respuesta (watchdog) <= 650 s

Si se desea crear un perfil de bus definido por el usuario, recomendamos realizar los siguientes ajustes:

- Target-Rotation-Time mínimo (Ttr) = 5000 x HSA (dirección PROFIBUS más alta)
- Supervisión de respuesta mínima (Watchdog) = 6250 x HSA

Recalcular

Pulsando el botón "Recalcular" se recalculan los parámetros.

Consulte también

Direcciones PROFIBUS (Página 544)

Descripción de los parámetros de bus (Página 546)

Descripción de los parámetros de bus

Descripción detallada de los parámetros de bus PROFIBUS

Parámetros de bus	Significado
Tslot_Init	El tiempo de espera a recepción determina cuánto tiempo debe esperar como máximo el emisor hasta recibir una respuesta del interlocutor direccionado. Si en el grupo de parámetros "Configuración de línea" se configura la influencia de los componentes de la línea sobre los tiempos de ejecución de telegramas, deberán tenerse en cuenta también estas proporciones. La proporción se suma al Tslot_Init predefinido y el total se utiliza como Tslot.
Max. Tsdr	El tiempo máximo de procesamiento de protocolo determina el tiempo máximo al cabo del cual el nodo que responde debe haber respondido.
Min. Tsdr	El tiempo mínimo de procesamiento de protocolo determina el tiempo mínimo al cabo del cual el nodo que responde puede responder.
Tset	El tiempo de activación es el tiempo que puede transcurrir entre la recepción de un telegrama de datos y la reacción correspondiente en el nodo.
Tqui	El tiempo de desactivación del modulador es el tiempo que requiere un nodo emisor después del fin del telegrama para conmutar de Enviar a Recibir.
Factor GAP	El factor de actualización GAP determina al cabo de cuántas rotaciones del token puede incorporarse un nuevo nodo activo al anillo lógico.
Retry Limit	Con este parámetro se determina cuántos intentos (repeticiones del telegrama) pueden realizarse como máximo para acceder a un nodo.

Parámetros de bus	Significado
Tslot	El tiempo de espera a recepción (slot time) determina cuánto tiempo debe esperar como máximo el emisor hasta recibir una respuesta del interlocutor direccionado. Si en la ficha "Líneas" se configura la influencia de los componentes físicos del bus sobre los tiempos de ejecución de telegramas, deberán tenerse en cuenta también estas proporciones. La proporción se suma al Tslot_Init predefinido y el total se utiliza como Tslot.
Tid2	El tiempo de reposo 2 determina al cabo de cuánto tiempo como mínimo un nodo emisor puede enviar el siguiente telegrama tras el envío de un telegrama no acusado.
Trdy	El tiempo de disponibilidad determina al cabo de cuánto tiempo como mínimo un nodo emisor puede recibir un telegrama de respuesta.
Tid1	El tiempo de reposo 1 determina al cabo de cuánto tiempo como mínimo un nodo emisor puede enviar el siguiente telegrama tras la recepción de una respuesta.
Ttr	El tiempo consignado de rotación del token es el tiempo máximo disponible para una rotación del token. Durante ese tiempo, todos los nodos activos (maestros DP, etc.) reciben una vez el testigo (token). La diferencia entre el tiempo consignado de rotación del token y el tiempo real de parada del token de un nodo determina cuánto tiempo queda para que los otros nodos activos (PG, otros maestros DP, etc.) puedan enviar telegramas.
Ttr típico	El tiempo de ciclo de datos típico es el tiempo de reacción medio del bus cuando todos los esclavos configurados intercambian datos con el maestro DP. Ninguno de los esclavos notifica un diagnóstico y no se produce ningún tráfico de telegramas adicional con PGs u otros nodos activos, etc. en el bus.
Supervisión de respuesta	El tiempo de supervisión de respuesta sólo se requiere para los sistemas de bus PROFIBUS DP. Determina al cabo de cuánto tiempo como máximo el maestro DP debe activar un esclavo DP con un nuevo telegrama de datos. Si esto no ocurre, el esclavo DP presupone que el maestro DP ha fallado y resetea sus salidas a un estado seguro.

Consulte también

Información importante sobre los parámetros de bus PROFIBUS (Página 545)

Perfiles de bus en PROFIBUS

Introducción

Según sean los tipos de dispositivos conectados y los protocolos utilizados en PROFIBUS, hay disponibles diferentes perfiles. Los perfiles se distinguen por sus opciones de ajuste y por el cálculo de los parámetros de bus. Seguidamente, se describen los perfiles.

Dispositivos con distintos perfiles en la misma subred PROFIBUS

La subred PROFIBUS sólo funciona a la perfección si los parámetros de bus de todos los dispositivos tienen los mismos valores. Si, por ejemplo, en una subred se utilizan tanto servicios DP como FMS, deben ajustarse siempre los juegos de parámetros de bus "más lentos" para todos los dispositivos, es decir, también el perfil "Universal (DP/FMS)" para dispositivos DP.

Perfiles y velocidades de transferencia

Perfiles	Velocidades de transferencia soportadas en Kbits/s
DP	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Estándar	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Universal (DP-FMS)	9,6 19,2 93,75 187,5 500 1500
Definido por el usuario	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000

Significado de los perfiles

Perfil	Significado
DP	Seleccione el perfil de bus "DP" si en la subred PROFIBUS sólo hay conectados dispositivos que cumplen con las exigencias de la norma EN 50170 Volumen 2/3, Parte 8-2 PROFIBUS. El ajuste de los parámetros de bus se ha optimizado para estos dispositivos. Entre ellos, se incluyen los dispositivos con interfaces maestro DP y esclavo DP de SIMATIC S7, así como unidades periféricas descentralizadas de otros fabricantes.
Estándar	Con respecto al perfil "DP", el perfil "Estándar" ofrece además la posibilidad de tener en cuenta para el cálculo de los parámetros de bus dispositivos de otro proyecto o dispositivos no configurados aquí. Los parámetros de bus se calcularán entonces con un algoritmo sencillo y no optimizado.

Perfil	Significado
Universal (DP/FMS)	Seleccione el perfil de bus "Universal (DP/FMS)" si en la subred PROFIBUS hay dispositivos individuales que utilizan el servicio PROFIBUS FMS. Se trata, por ejemplo, de los siguientes dispositivos: • CP 343-5 (SIMATIC S7) • Dispositivos PROFIBUS FMS de otros fabricantes Al igual que en el perfil "Estándar", en este caso también existe la posibilidad de tener en cuenta dispositivos adicionales para el cálculo de los parámetros de bus.
Definido por el usuario	La subred PROFIBUS sólo funciona a la perfección si los parámetros del perfil de bus están sintonizados entre sí. Seleccione el perfil de bus "Definido por el usuario" cuando, para el funcionamiento de un dispositivo PROFIBUS, ninguno de los perfiles restantes "se ajuste" y deban adaptarse los parámetros de bus a la estructura especial. Para más información, puede consultarse la documentación del dispositivo PROFIBUS. Los valores predeterminados sólo deberían modificarse si se está familiarizado con la parametrización del perfil de bus para PROFIBUS. Con este perfil de bus tampoco pueden introducirse todas las combinaciones teóricamente ajustables. La norma PROFIBUS define algunos valores límite de parámetros en función de otros parámetros. Así, por ejemplo, no se permite que un respondedor pueda responder (Min Tsdr) antes de que el iniciador pueda recibir el telegrama (Trdy). Estos requisitos también se comprueban en el perfil "Definido por el usuario". Sugerencia: Los últimos parámetros de bus válidos en la subred PROFIBUS siempre se ajustan automáticamente como definidos por el usuario. Si, por ejemplo, el perfil de bus "DP" era válido para la subred, se ajustarán los parámetros de bus para "DP" en el perfil de bus "Definido por el usuario". A partir de este ajuste, pueden modificarse los parámetros. El recálculo de los tiempos de vigilancia no se realiza automáticamente para no poner en peligro la igualdad de los valores ajustados, p. ej. para la configuración de otras herramientas de configuración, sin que el usuario lo sepa. Los tiempos de vigilancia Ttr y la supervisión de respuesta pueden calcularse con los parámetros ajustados por el usuario: Para ello, haga clic en el botón "Recalcular".

Nota

En todos los perfiles PROFIBUS es posible tanto el funcionamiento monomaestro como el funcionamiento multimaestro.

Información importante sobre la configuración de línea PROFIBUS

Configuración de línea y parámetros de bus

La información sobre la configuración de línea puede tenerse en cuenta para el cálculo de los parámetros de bus. Para ello, hay que activar la casilla de verificación "Considerar configuración de línea" en las propiedades de la subred PROFIBUS.

La información restante depende del tipo de línea utilizada. Pueden ajustarse las siguientes opciones:

- Cable de cobre
- Cable de fibra óptica / anillo óptico

Configuración de línea PROFIBUS, anillo óptico

El cálculo depende de los tipos de OLM utilizados. La selección se efectúa activando la casilla de verificación (pueden activarse varias casillas y como mínimo deberá seleccionarse un tipo de OLM):

- OLM/P12
- OLM/G12
- OLM/G12-EEC
- OLM/G12-1300

Se realizan las siguientes adaptaciones de los parámetros de bus:

- Configuración de un nodo no existente

  Nota

  Las siguientes restricciones también rigen para los nodos pasivos (p. ej. esclavos DP) en los anillos ópticos:

  Pueden conectarse a la red PROFIBUS nodos con HSA-1 como máximo. En el caso de un HSA de 126, las direcciones 126 y 125 no pueden utilizarse. Así, puede haber como máximo 125 nodos en el bus (n.º 0 a 124).

  En el caso de un HSA menor o igual a 125, no pueden utilizarse las direcciones HSA y mayores. En cambio, la dirección HSA-1 puede utilizarse.

- Aumento del valor Retry a 3
- Ajuste del tiempo slot mínimo necesario para el funcionamiento del anillo

  Nota

  Para OLM/P12 se requieren valores de tiempo slot bajos; para OLM/G12 y OLM/G12-EEC, valores medios, y para OLM/G12-1300, valores altos. De ahí resulta un alto rendimiento en una extensión de red pequeña o un rendimiento de medio a bajo en una extensión de red de media a grande.

Carga de comunicación PROFIBUS

Carga de comunicación - considerar los nodos de red adicionales

Los parámetros de bus dependen del volumen de comunicación de los nodos de red activos. En este caso, existen diferencias entre la comunicación cíclica (DP) y la comunicación acíclica orientada a la conexión (comunicación S7, Send/Receive (FDL), FMS). A diferencia del DP, el número y la magnitud de las peticiones de comunicación (carga de comunicación) dependen

del programa de usuario. Por este motivo, la carga de comunicación no siempre puede determinarse automáticamente.

Para calcular los tiempos de bus, puede fijarse una configuración de red en el grupo de parámetros "Dispositivos de red adicionales" que sea diferente de la configuración de red configurada.

Considerar el perfil

La configuración de red puede determinarse para los perfiles "Estándar", "Universal (DP/FMS)" y "Definido por el usuario". En el perfil "DP", los parámetros del grupo de parámetros "Dispositivos de red adicionales" no pueden introducirse.

Cuantificación de la carga de comunicación

Para tener en cuenta la carga de comunicación, pueden realizarse los siguientes ajustes:

- información sobre el número de nodos de red no configurados;
- información sobre la carga de comunicación resultante de los programas de usuario para comunicación FDL o S7. Para ello, pueden seleccionarse los siguientes niveles:
  - Bajo
    Típico para DP, no hay ninguna comunicación de datos mayor a excepción de DP.
  - Medio
    Típico para el modo mixto de DP y otros servicios de comunicación (p. ej. comunicación S7), cuando DP tiene altas demandas de tiempo y en caso de volumen medio de comunicación acíclica.
  - Alto
    Para el modo mixto de DP y otros servicios de comunicación (p. ej. comunicación S7), cuando DP tiene bajas demandas de tiempo y en caso de volumen alto de comunicación acíclica.

Configuración de red de Industrial Ethernet

Reglas para la configuración de red

Las interfaces Ethernet de los dispositivos tienen una dirección IP predeterminada que puede modificarse.

Dirección IP

Los parámetros IP son visibles si el módulo apto para la comunicación soporta el protocolo TCP/IP. Esto suele suceder en todos los módulos Ethernet.

La dirección IP consiste en 4 números decimales comprendidos en un rango entre 0 y 255. Los decimales están separados entre sí por un punto.

Ejemplo: 140.80.0.2

La dirección IP se compone de:

- dirección de la subred IP
- dirección del dispositivo (también se denomina host o nodo de red)

Máscara de subred

La máscara de subred separa las dos direcciones. Determina qué parte de la dirección IP direcciona la red y qué parte direcciona el dispositivo.

Los bits activados de la máscara de subred determinan la parte de la dirección IP correspondiente a la red.

Ejemplo:

Máscara de subred: 255.255.0.0 = 11111111.11111111.00000000.00000000

En el ejemplo superior de dirección IP, la máscara de subred tiene el significado siguiente:

Los 2 primeros bytes de la dirección IP determinan la subred, es decir, 140.80. Los 2 últimos bytes direccionan el dispositivo, es decir, 0.2.

Por lo general rige lo siguiente:

- La dirección de la red resulta de la combinación lógica Y de la dirección IP y la máscara de subred.
- La dirección del dispositivo resulta de la combinación lógica Y-NO de la dirección IP y la máscara de subred.

Relación entre la dirección IP y la máscara de subred predeterminada

Existe un acuerdo respecto de la asignación entre áreas de direcciones IP y "máscaras de subred predeterminadas". El primer decimal de la dirección IP (desde la izquierda) determina la estructura de la máscara de subred predeterminada en cuanto al número de valores "1" (binarios), del siguiente modo:

Dirección IP (dec.)	Dirección IP (bin.)	Clase de dirección	Máscara de subred predeterminada
0 a 126	0xxxxxxx.xxxxxxxx....	A	255.0.0.0
de 128 a 191	10xxxxxx.xxxxxxxx...	B	255.255.0.0
de 192 a 223	110xxxxx.xxxxxxxx...	C	255.255.255.0

Nota

Rango de valores del primer decimal

En el primer decimal de la dirección IP también se acepta un valor entre 224 y 255 (clase de dirección D, etc.). Sin embargo, no es recomendable porque no se verifica la dirección de estos valores.

Enmascarar otras subredes

La máscara de subred permite seguir estructurando una subred asignada a las clases de dirección A, B o C y crear subredes "privadas" poniendo a "1" otras posiciones de orden inferior de la máscara de subred. Por cada bit puesto a "1" se duplica el número de redes "privadas" y se divide en dos el número de dispositivos que contiene. Hacia fuera, la red sigue pareciendo una red individual.

Ejemplo:

En una subred de la clase de dirección B (p. ej. dirección IP 129.80.xxx.xxx), la máscara de subred predeterminada se modifica del siguiente modo:

Máscaras	Decimal	Binario
Máscara de subred predeterminada	255.255.0.0	11111111.11111111.00000000.00000000
Máscara de subred	255.255.128.0	11111111.11111111.10000000.00000000

Resultado:

Todos los dispositivos con direcciones comprendidas entre 129.80.001.xxx y 129.80.127.xxx están en una subred IP y todos los dispositivos con direcciones comprendidas entre 129.80.128.xxx y 129.80.255.xxx están en otra subred IP.

Transición de red (router)

Las transiciones de red (router) sirven para conectar las subredes IP. Si un datagrama IP debe enviarse a otra red, primero hay que transmitirlo a un router. Para que esto sea posible, hay que introducir la dirección del router en cada dispositivo de la subred IP.

La dirección IP de un dispositivo de la subred y la dirección IP de la transición de red (router) solo pueden diferir en las cifras que son "0" en la máscara de subred.

Consulte también

Ajustes para la interconexión de dispositivos Ethernet (Página 553)

Ajustes para la interconexión de dispositivos Ethernet

A continuación se describe el comportamiento de STEP 7 en la interconexión de dispositivos PROFINET y los efectos de la interconexión de puertos en la vista de redes.

Relación entre la interconexión de puertos y la vista de redes

Vista topológica:

En la vista topológica se define la interconexión física de puertos Ethernet.

En ella se determina de manera concreta qué puerto Ethernet de un dispositivo debe conectarse a un determinado puerto Ethernet de otro dispositivo mediante un cable Ethernet (topología teórica).

Ejemplo:

Se establece que el puerto 1 de la interfaz PROFINET de la CPU se conectará con el puerto 2 de la interfaz PROFINET del dispositivo A a través de un cable Ethernet.

En los dispositivos con varias interfaces PROFINET se define también la interfaz.

Ejemplo:

Se establece que el puerto 1 de la interfaz PROFINET X2 de la CPU se conectará con el puerto 2 de la interfaz PROFINET del dispositivo A a través de un cable Ethernet.

La interconexión de los puertos Ethernet puede realizarse en modo gráfico o de tabla.

Vista de redes:

En la vista de redes se establece qué dispositivos se conectarán entre sí a través de una subred Ethernet. No se determina a través de qué puertos Ethernet se interconectarán los dispositivos (esto se lleva a cabo en la interconexión de puertos).

La interconexión de puertos tiene repercusiones sobre la vista de redes:

Cuando se interconectan puertos Ethernet de dispositivos en la vista topológica, STEP 7 une las interfaces PROFINET interconectadas de los dispositivos en la vista de redes con una subred Ethernet (línea verde).

El recorrido de la línea verde no refleja el tendido efectivo de los cables. El cableado concreto se define en la vista topológica.

Una subred Ethernet tiene un nombre y una ID de subred S7. Ambos valores pueden ajustarse en las propiedades de la subred.

¿Con qué subred Ethernet se conectan los dispositivos interconectados?

STEP 7 distingue los siguientes casos:

- La subred Ethernet está especificada (subred predeterminada).
- La subred Ethernet no está especificada (sin subred predeterminada).

La subred Ethernet está especificada (subred predeterminada)

En las propiedades de una subred Ethernet está activada la opción "Conectar dispositivos no interconectados en caso de interconexión de puertos con esta subred" (opción predeterminada).

Esta opción puede estar activada con una y solo una subred Ethernet.

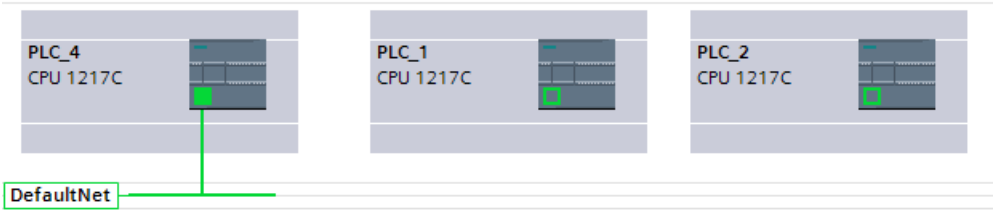
Active esta opción si es necesario, con la subred Ethernet que se deba prolongar al interconectar dispositivos no conectados en red. En lo sucesivo esta subred se denominará "subred predeterminada".

Reacción de STEP 7:

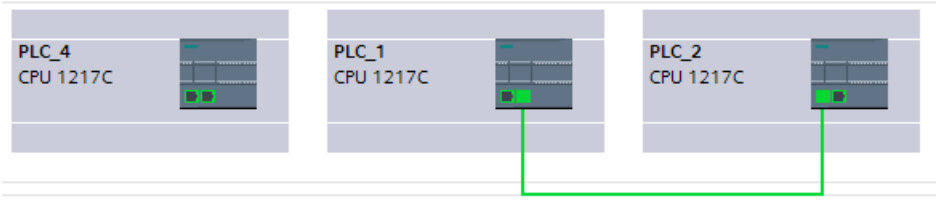
Cuando en la vista topológica se interconectan los puertos de dos dispositivos no conectados en red, STEP 7 conecta estos dispositivos a la subred predeterminada.

Ejemplo:

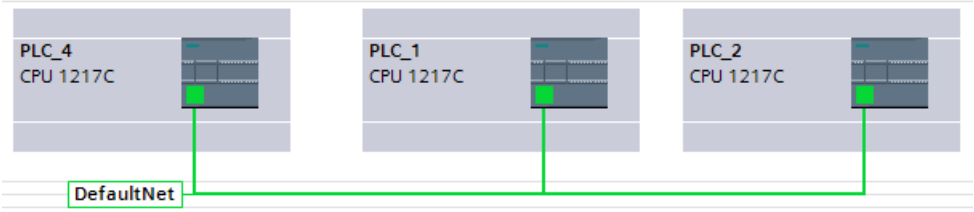
Paso 1: Crear una subred en la vista de redes en PLC_4, la opción "Conectar dispositivos no intercaonectados en caso de interconexión de puertos con esta subred" está **activada**.



Paso 2: Interconectar PLC_1 con PLC_2 (vista topológica).



Resultado: Todos los PLC están conectados a la misma subred predeterminada (vista de redes).



La subred Ethernet no está especificada (sin subred predeterminada).

Este caso se produce cuando se cumplen las siguientes condiciones:

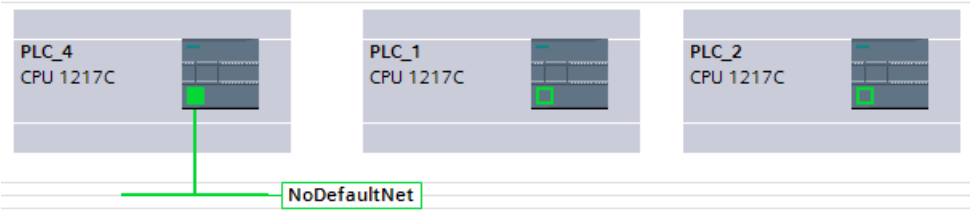
- Se ha agregado una subred a una interfaz PROFINET (esta interfaz se denominará en lo sucesivo "interfaz A").
- En las propiedades de esta subred está desactivada la opción "Conectar dispositivos no interconectados en caso de interconexión de puertos con esta subred" (sin subred predeterminada).
- Tampoco existe ninguna otra subred Ethernet en la que esta opción esté activada.

Reacción de STEP 7:

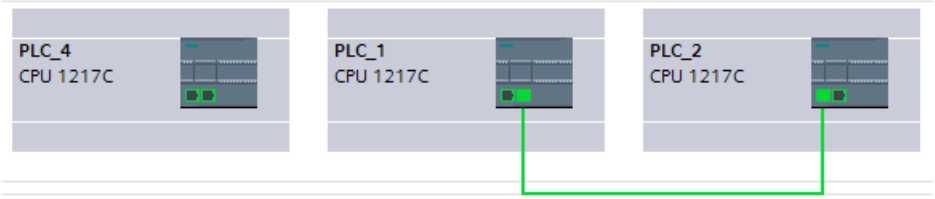
- La subred Ethernet de la interfaz A solo se prolonga si se interconecta un puerto de la interfaz A con un puerto de otro dispositivo.
- Al interconectar puertos de otros dispositivos no conectados en red, STEP 7 crea otra subred Ethernet.

Ejemplo:

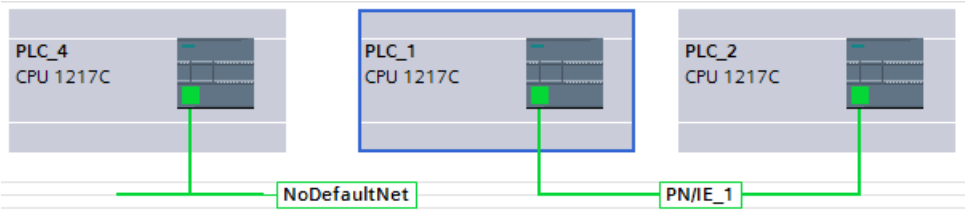
Paso 1: Crear una subred en la vista de redes en PLC_4, la opción "Conectar dispositivos no intercaonectados en caso de interconexión de puertos con esta subred" está **desactivada**.



Paso 2: Interconectar PLC_1 con PLC_2 (vista topológica).



Resultado: PLC_1 y PLC_2 están conectadas con una nueva subred (vista de redes).



Adaptación manual de las direcciones IP

Es posible que STEP 7 no adapte las direcciones IP de los dispositivos interconectados de manera que pueda compilarse la conexión en red resultante, p. ej. en el caso de dispositivos con varias interfaces PROFINET.

En estos casos es necesario adaptar manualmente las direcciones IP de los dispositivos.

Se aplican las reglas siguientes:

- Los dispositivos que deben comunicarse unos con otros sin router, no pueden pertenecer a subredes IP diferentes.
- En los dispositivos con varias interfaces PROFINET, las interfaces deben encontrarse en diferentes subredes IP.

Para cambiar la dirección IP de una interfaz PROFINET; ejecute los pasos siguientes:

1. Pase a la vista de redes (si no se encuentra ya en ella).
2. Haga clic con el botón izquierdo del ratón en el símbolo de la interfaz PROFINET que no deba incorporarse a la subred IP.
3. En las propiedades de la interfaz PROFINET, modifique el componente de subred de la dirección IP (en el apartado "Direcciones Ethernet").

Ejemplo: Cambio manual de la parte de subred IP en dispositivos con varias interfaces PROFIENT en una subred

La dirección IP es: "192.168.0.1".

La máscara de subred es "255.255.255.0".

Los tres primeros números "192.168.0" constituyen el componente de subred IP de la dirección IP "192.168.0.1".

Sustituya el componente de subred IP por "192.168.1", por ejemplo

Consulte también

Interconectar los puertos en la vista gráfica (Página 742)

Interconexión de los puertos en la vista de tabla (Página 744)

Configuración de red de Industrial Ethernet (Página 551)

Configuración de red de AS-Interface

Una AS-Interface se compone de un maestro AS-i y esclavos AS-i que están conectados entre sí a través de una subred AS-i.

Reglas para la configuración de red AS-i

Todos los nodos de una subred AS-i deben tener direcciones distintas.

Los ajustes sólo deben cargarse a través de la red si todos los módulos de una subred poseen direcciones distintas y la estructura real coincide con la configuración de red creada.

En una subred AS-i puede operarse un maestro AS-i y hasta 31 esclavos AS-i.

En el capítulo "AS-Interface" y en la documentación de los módulos maestros AS-i encontrará más información sobre la configuración de una AS-Interface con maestro AS-i y esclavos AS-i.

8.1.3.2 Comunicación vía conexiones

Trabajar con conexiones (S7-1200)

Conexión S7

Introducción a la configuración de conexiones

Definición

Una conexión define la asignación lógica de dos interlocutores con el fin de ejecutar servicios de comunicación. Una conexión define lo siguiente:

- Interlocutores que intervienen
- Tipo de conexión (p. ej. conexión S7)

- Propiedades especiales (p. ej. si una conexión permanece establecida o si se establece y deshace dinámicamente en el programa de usuario y si deben enviarse avisos sobre el estado operativo)
- Vía de conexión

Información importante para configurar conexiones

Al configurar conexiones se asigna un nombre de conexión local en una conexión S7 en calidad de identificación local unívoca.

En la vista de redes, junto a la ficha "Vista general de redes" aparece la ficha "Conexiones". Esta ficha incluye la tabla de conexiones. Una fila de dicha tabla representa una conexión configurada desde el punto de vista del interlocutor local con sus propiedades, p. ej. entre dos CPUs S7-1200.

Información importante sobre el consumo de recursos de conexión

Introducción

Cada conexión requiere recursos de los dispositivos implicados tanto para el punto final como para el punto de transición. El número de recursos de conexión es específico del dispositivo.

Si todos los recursos de conexión de un interlocutor están ocupados, no podrá establecerse ninguna conexión nueva. Esto se reconoce por el hecho de que la nueva conexión se resaltará en rojo en la tabla de conexiones. En este caso, la configuración es incoherente y no puede compilarse.

Conexiones S7

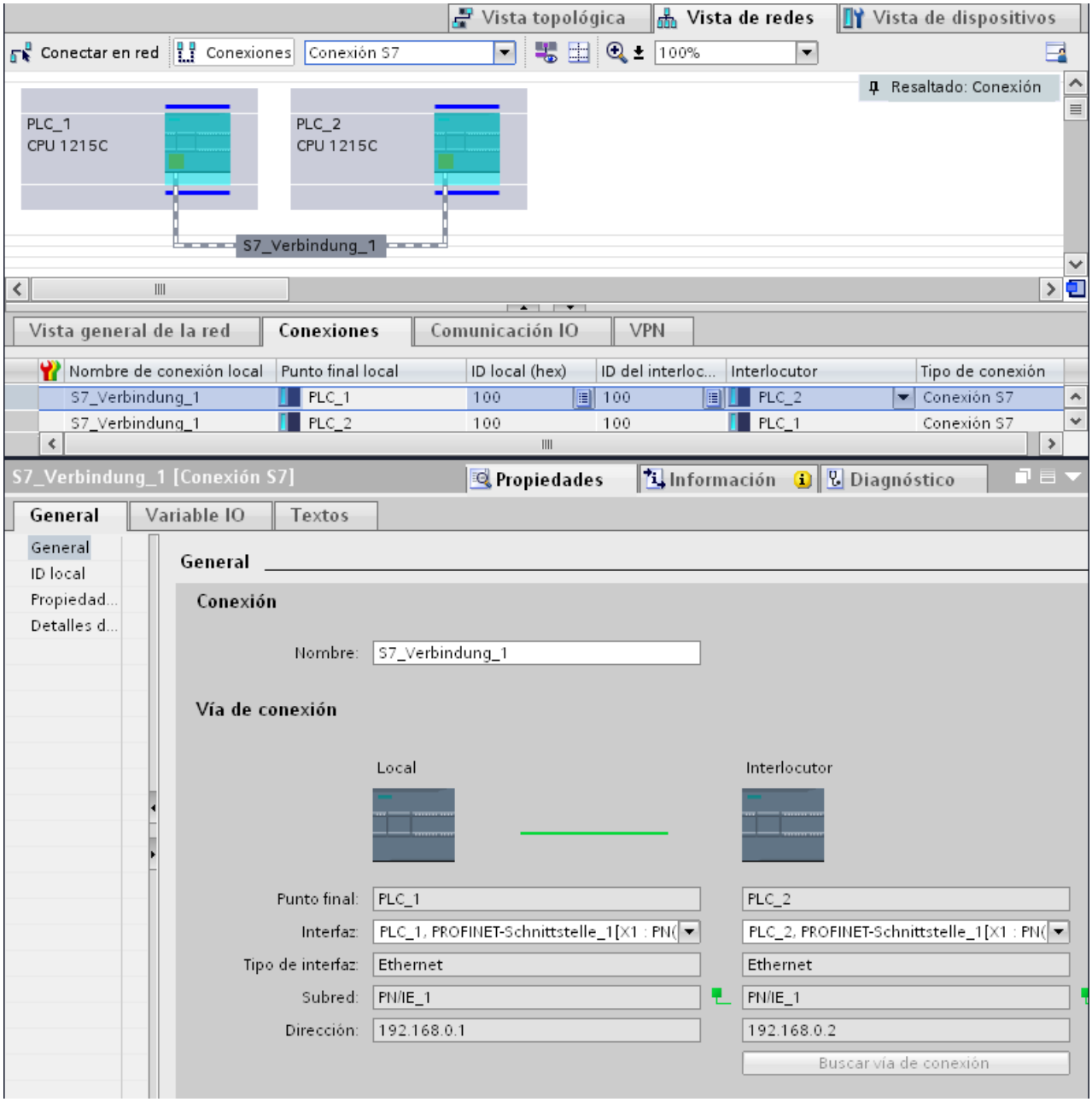
Cada conexión S7 establecida a través de la interfaz PN ocupa un recurso de conexión para el punto final en la CPU S7-1200. El interlocutor también requiere un recurso de conexión.

En "Propiedades > Recursos de conexión" de la ventana de inspección encontrará un resumen de los recursos de conexión disponibles y ocupados en la CPU S7-1200 seleccionada.

Vistas con información sobre las conexiones configuradas

Desde las vistas descritas a continuación se accede a toda la información y las funciones relacionadas con la configuración y verificación de conexiones.

- Visualización de conexiones en la vista de redes
- Tabla de conexiones
- Ficha "Propiedades" de una conexión en la ventana de inspección



Ventajas

La información representada en estas vistas siempre es actual en cuanto a las acciones de usuario inmediatas. Esto significa que:

- La tabla de conexiones muestra todas las conexiones creadas.
- Si se ha seleccionado una conexión en la tabla de conexiones, rige lo siguiente:
  - Si el modo de conexión es activo, la vía de conexión se resalta gráficamente en la vista de redes.
  - La ficha "Propiedades" de la ventana de inspección muestra los parámetros de dicha conexión.

La tabla de conexiones

La tabla de conexiones ofrece las funciones siguientes:

- Relación de todas las conexiones del proyecto
- Selección de una conexión y su visualización en la vista de redes (en caso de modo de conexión activo)
- Cambio de interlocutores
- Visualización de información de estado

Ficha "Propiedades" de una conexión en la ventana de inspección

El cuadro de diálogo de propiedades tiene el significado siguiente:

- Visualización de los parámetros de la conexión
- Visualización de la vía de conexión
- Especificación posterior de las conexiones por medio del botón "Buscar vía de conexión"

Crear una conexión nueva

Alternativas para crear una conexión

Existen las posibilidades siguientes para crear una conexión en la vista de redes:

- Configuración gráfica de la conexión
- Configuración de la conexión desde un cuadro de diálogo

Los diferentes pasos que deben seguirse se explican en los capítulos siguientes.

Requisitos y resultado

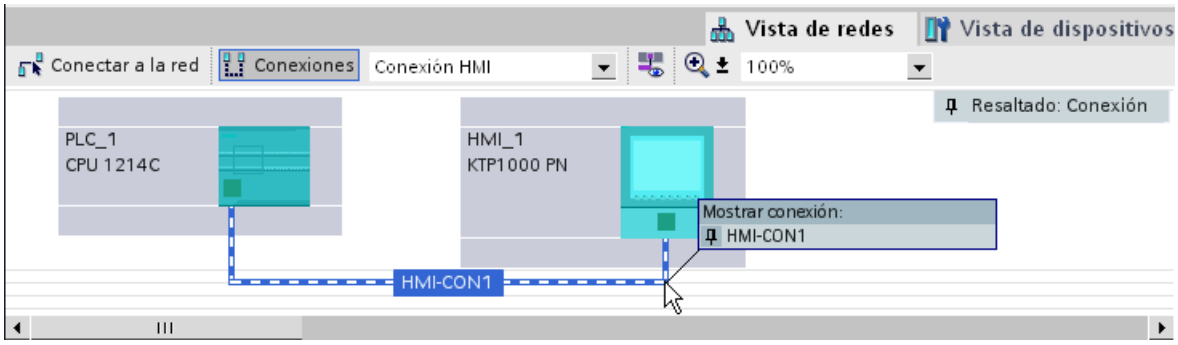
En la vista de redes se han insertado los dispositivos entre los que deben configurarse las conexiones.

Especificar una conexión

Si ambos interlocutores están conectados a la misma red para el tipo de conexión seleccionado, cree una conexión completamente especificada entre ambos interlocutores mediante la selección gráfica o desde un cuadro de diálogo.

Dicha conexión se registrará automáticamente en la tabla de conexiones de la CPU S7-1200. y se le asignará un nombre de conexión local.

La representación siguiente muestra una conexión configurada en un dispositivo conectado en red:



Crear una conexión nueva gráficamente

Configurar conexiones gráficamente

En el caso de la configuración gráfica de conexiones, la vía de conexión se determina automáticamente, siempre que haya interfaces y recursos. Seleccione los dispositivos que deben conectarse en la configuración actual.

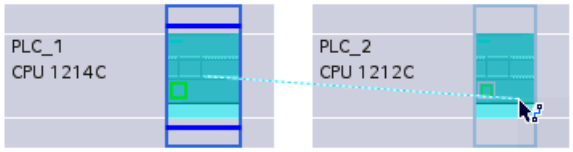
Determinar automáticamente la vía de conexión

Para crear una conexión gráficamente, proceda del siguiente modo:

1. Haga clic en el botón "Conexiones".

   De esta forma se activa el modo de conexión. Ahora puede seleccionar el tipo de conexión deseado. Esto se reconoce por el comportamiento siguiente:
   En la vista de redes están resaltados en color los dispositivos que pueden utilizarse en el proyecto para el tipo de conexión seleccionado.

2. Arrastre el cursor del ratón, manteniendo pulsado el botón del ratón, desde el dispositivo del que debe partir la conexión hasta el dispositivo en el que debe finalizar la conexión.

3. Suelte el botón del ratón en el dispositivo de destino para crear la conexión entre ambos.

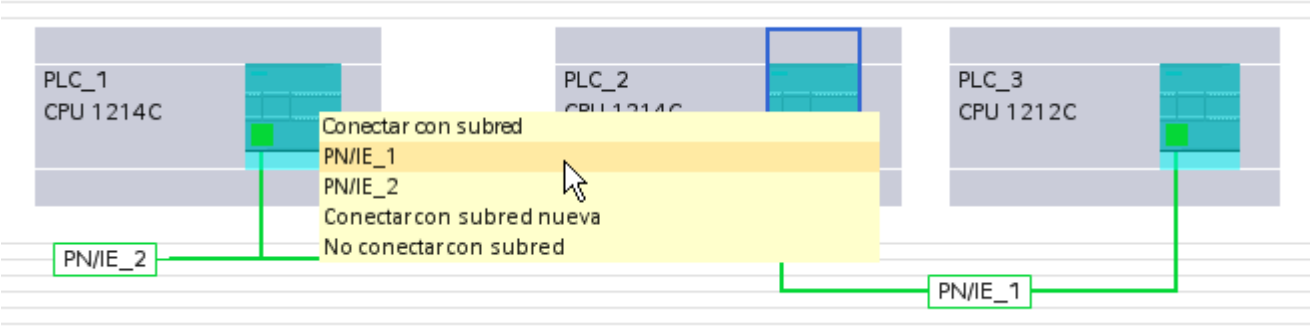
Resultado

- Se ha creado una conexión especificada.
- La vía de conexión se representa resaltada.
- La conexión está registrada en la tabla de conexiones.

Configurar una conexión cuando falta la asignación de red o ésta no es unívoca

Las conexiones en red que faltan se establecen en la medida de lo posible de forma automática al crear una conexión. En caso de no ser posible una asignación de red unívoca, se produce una consulta al finalizar la configuración de la conexión. En este caso se ofrecen las subredes existentes para su selección.

Ejemplo en la siguiente figura: al crear una conexión entre las estaciones PLC_1 y PLC_2 que aún no han sido conectadas en red se produce una consulta.



Crear una conexión nueva desde un cuadro de diálogo

Configurar conexiones desde un cuadro de diálogo

Determine el dispositivo local y su interlocutor.

Procedimiento

Para crear una conexión desde un cuadro de diálogo, proceda del siguiente modo:

1. En el menú contextual de un interlocutor para el que debe crearse una conexión, elija el comando "Crear conexión nueva".
   Se abre el cuadro de diálogo "Crear conexión nueva".
2. Seleccione el punto final del interlocutor.
   En el lado derecho del cuadro de diálogo aparecerá una posible vía de conexión apropiada para el punto final seleccionado, si existiera. Las vías incompletas, como en el caso de una CPU no especificada, se marcan con un signo de exclamación sobre fondo rojo.
3. Para aplicar la conexión configurada y configurar otras conexiones referentes a otros puntos finales, haga clic en "Agregar".
   Para salir del cuadro de diálogo pulse "Aceptar".

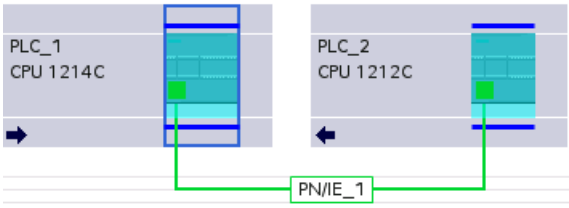
Trabajar en la vista de redes

Resaltar la vía de conexión y el interlocutor en la vista de redes

Para mostrar los interlocutores de todos o algunos tipos de conexión en la vista de redes, proceda del siguiente modo:

1. Haga clic en el botón "Conexiones".

2. En el menú contextual de la CPU S7 cuyo interlocutor desea visualizar en la vista de redes, elija el comando "Resaltar interlocutores".
3. En el menú que aparece a continuación, elija "Todos los interlocutores".
   El dispositivo local y las CPUs de los dispositivos de destino están seleccionados. El interlocutor local presenta una flecha hacia la derecha y los interlocutores remotos una flecha hacia la izquierda.
4. Abra una lista con los datos de los dispositivos de destino seleccionando la flecha del dispositivo local. Esta función complementaria es útil en las configuraciones de red complejas en las que no se ven todos los dispositivos.



Nota

Es posible visualizar uno de los interlocutores que no se ve en el área de visualización actual de la vista de redes. Haga clic en el interlocutor en la lista que aparece. Resultado: la visualización se desplaza hasta que aparece el interlocutor.

Trabajar con la tabla de conexiones

Funciones básicas de las tablas

La tabla de conexiones soporta las siguientes funciones básicas para editar una tabla:

- Modificar el ancho de las columnas
- Mostrar el significado de una columna, fila o campo a través de tooltips.

Modificar el ancho de las columnas

Para adaptar el ancho de una columna al contenido de modo que se lean los textos de todas las filas, proceda del siguiente modo:

1. Posicione el puntero del ratón en el encabezado de la tabla de conexiones, a la derecha de la columna que debe optimizarse, hasta que el puntero adopte la forma de dos líneas paralelas (como si quisiera modificar el ancho de la columna arrastrando el puntero del ratón).
2. Haga doble clic en este punto.

o

1. Abra el menú contextual en el encabezado de la tabla.
2. Haga clic en
   - "Optimizar ancho de columnas" o
   - "Optimizar ancho de todas las columnas".

Si las columnas son demasiado estrechas, se puede ver todo el contenido de los diferentes campos colocando el puntero del ratón brevemente sobre el campo en cuestión.

Mostrar/ocultar columnas

Mediante el menú contextual de los encabezados de la tabla de conexiones se controla la visualización de las diferentes columnas. Con el comando "Mostrar/ocultar columnas" del menú contextual aparece una vista general de las columnas disponibles. Con las casillas de verificación se controla la visibilidad de las columnas.

Si desea guardar la disposición, el ancho y la visibilidad de las columnas, haga clic en la función "Anotar disposición" en la parte superior derecha de la vista de redes.

Símbolo	Significado
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Navegar por la tabla de conexiones con las teclas de cursor

Las techas de cursor FLECHA ARRIBA y FLECHA ABAJO permiten seleccionar una conexión en la tabla de conexiones; la conexión queda seleccionada y se resalta en la vista de redes.

Modificar las propiedades de la conexión

Algunos de los parámetros visualizados en la tabla de conexiones pueden editarse directamente. Así, por ejemplo, el nombre de la conexión sólo puede modificarse en la tabla de conexiones.

Modificar el interlocutor

El interlocutor de una conexión se modifica del siguiente modo:

1. Seleccione la conexión.
2. Elija el nuevo interlocutor en la lista desplegable activada de la columna "Interlocutor".

Borrar conexiones

Las conexiones configuradas se borran en la vista de redes o en la tabla de conexiones.

En la vista de redes se puede borrar en cada caso una conexión resaltada. En la tabla de conexiones se borran una o varias conexiones.

Procedimiento

Para borrar una conexión, proceda del siguiente modo:

1. Seleccione la conexión que desea borrar:
   - En la vista de redes: seleccione la conexión que desea borrar.
   - En la tabla de conexiones: seleccione las filas de las conexiones que deben borrarse (posibilidad de selección múltiple).
2. Abra el menú contextual con el botón derecho del ratón.
3. Elija el comando "Borrar".

Resultado

La conexión seleccionada se borrará por completo.

Copiar conexiones

Introducción

Las conexiones no se copian individualmente, sino siempre en relación con el proyecto o dispositivo.

Es posible copiar:

- Proyectos enteros
- Uno o varios dispositivos dentro de un proyecto o en varios proyectos

Copiar un proyecto

Al copiar un proyecto se copiarán también todas las conexiones configuradas. No es necesario realizar ajustes en las conexiones copiadas porque siguen siendo coherentes.

Copiar dispositivos

Al copiar dispositivos para los que se han configurado conexiones también se copiarán las conexiones. Para completar la vía de conexión todavía hay que arrastrar la conexión en red.

Una CPU S7-1200 con un firmware V1.0 es simplemente un servidor de conexiones y no tiene ninguna configuración de conexiones propia. Por ello, al copiar una CPU S7-1200 con un firmware V1.0 no se copiarán conexiones.

Conexiones incoherentes - conexiones sin asignación

Cuando una conexión es incoherente, significa que la estructura de los datos de conexión está destruida o bien la conexión no funciona en relación con el proyecto.

No es posible compilar ni cargar conexiones incoherentes, como tampoco lo es su operación.

En la tabla de conexiones, las conexiones incoherentes se reconocen por el color rojo.

Causas posibles de conexiones incoherentes

- Borrado o modificaciones de la configuración de hardware
- Falta de conexiones en red entre las interfaces del proyecto necesarias para una conexión.
- Rebase por exceso de los recursos de conexión
- Conexiones con un interlocutor no especificado sin indicar la dirección del mismo.

Encontrará información detallada sobre la causa de la incoherencia en la ficha "Compilar" después de la compilación (Edición > Compilar).

Soluciones

Para asignar una vía de conexión cerrada a una vía de conexión abierta existente, amplíe la configuración del dispositivo de tal forma que las interfaces necesarias para el tipo de conexión se encuentren disponibles para ambos interlocutores. Con el botón "Buscar vía de conexión" de "Propiedades > General > Interfaz" de la ventana de inspección es posible establecer una conexión con un interlocutor existente.

Ajustes generales de la conexión S7

Parámetros de conexión generales

En el grupo de parámetros "General" de las propiedades de la conexión se muestran los parámetros generales que identifican el punto final local de la conexión.

Aquí existe la posibilidad de asignar la vía de conexión y especificar por completo a los interlocutores.

ID local

Aquí se muestra la ID local del módulo desde el que se observa la conexión (interlocutor local). La ID local se puede modificar. Esto es necesario cuando ya se han programado bloques de función de comunicación y debe utilizarse la ID local indicada allí para la conexión.

Propiedades de conexión especiales

Visualización de las propiedades de conexión (modificable en función de los componentes utilizables):

- Unilateral
  Unilateral significa que el interlocutor es el servidor de dicha conexión y que no puede enviar o recibir activamente.
- Iniciativa local
  En una conexión unilateral, p. ej. con una CPU S7-1200 con versión de firmware V1.0, sólo puede haber disponible un interlocutor para la iniciativa local. En una conexión bilateral existe la posibilidad de especificar qué interlocutor debe tomar la iniciativa.
- Enviar avisos de estado operativo
  Indica si el interlocutor local envía avisos de estado operativo al interlocutor.

Detalles de dirección

Visualización de los detalles de direcciones de la conexión S7. Si un interlocutor no está especificado, pueden modificarse los valores de rack y slot. Todos los demás valores se tomarán de la configuración actual y no podrán modificarse.

Detalles de direcciones de la conexión S7

Significado

En los detalles de dirección se muestran los puntos finales de la conexión, que pueden localizarse indicando el rack y el slot.

Cuando se establece una conexión, se asignan de forma fija recursos específicos de un módulo a dicha conexión. Para la asignación es imprescindible que el recursos de conexión pueda direccionarse. El TSAP (Transport Service Access Point) es prácticamente la dirección del recurso y se forma, entre otros, por medio del recurso de conexión o, en CPUs S7-1200 a partir del firmware V2.0, con la SIMATIC-ACC (SIMATIC Application Controlled Communication).

Estructura del TSAP para S7-1200

- Para CPU S7-1200 con firmware V2.0 o superior:
  "SIMATIC-ACC"<nnn><mm>
  nnn = ID local
  mm = valor libre
- Para CPU S7-1200 con firmware V1.0:
  <xx>.<yz>
  xx = número del recurso de conexión
  y = número de rack
  z = número de slot

Estructura TSAP en función del interlocutor

La estructura del TSAP en las CPUs S7-1200 depende del firmware correspondiente y del interlocutor remoto. Si se conecta una CPU S7-1200 con una CPU S7-300/400, la primera utiliza también la estructura TSAP incluyendo el recurso de conexión.

Consulte los ejemplos siguientes para TSAPs de distintas configuraciones de conexiones

- Conexión entre dos CPUs S7-1200 (ambas con firmware V2.0):
  - CPU S7-1200 "A" con firmware V2.0 e ID local 100:
    TSAP: SIMATIC-ACC10001
  - CPU S7-1200 "B" con firmware V2.0 e ID local 5AE:
    TSAP: SIMATIC-ACC5AE01
- Conexión entre dos CPUs S7-1200 (con firmware V2.0 y V1.0):
  - CPU S7-1200 con firmware V2.0 e ID local 1FF:
    TSAP: SIMATIC-ACC1FF01
  - CPU S7-1200 con firmware V1.0 (rack 0, slot 1, recurso de conexión 03):
    TSAP: 03.01
- Conexión entre una CPU S7-1200 (con firmware V2.0) y una CPU S7-300/400:
  - CPU S7-1200 con firmware V2.0 (rack 0, slot 1, recurso de conexión 12):
    TSAP: 12.01
  - CPU S7-300/400 (rack 0, slot 2, recurso de conexión 11):
    TSAP: 11.02

Conexiones S7 a través de CM/CP

Introducción

Las CPU S7-1200 con una versión de firmware V2.0 o superior soportan conexiones S7 unilaterales y bilaterales a través de interfaces CM/CP. De este modo aumenta la cantidad de redes y puertos Ethernet disponibles para las conexiones S7. Aunque la conexión se realiza entonces a través del CM/CP, la respectiva CPU S7-1200 es un punto final en la conexión. El otro punto final puede ser cualquier otro dispositivo en el caso de las conexiones bilaterales. Este otro dispositivo debe soportar también conexiones S7.

Volumen de datos y recursos disponibles

El número de conexiones soportadas por CM/CP se puede consultar en el manual de producto adjunto con cada CM/CP. Añadiendo más CM/CP se puede aumentar el número de conexiones por dispositivo.

Si un dispositivo incorpora varios CM/CP, al sobrepasar este límite se cambia automáticamente al siguiente CP. En caso necesario, asigne selectivamente las conexiones a través de la elección de vía.

Nota

Los CPs actuales soportan la transferencia de datos > 240 bytes.

Los CPs con una versión más antigua soportan la transferencia de datos con una longitud de hasta 240 bytes.

Tenga en cuenta las indicaciones del manual de producto del CP Ethernet.

Tareas del CM/CP Ethernet en el modo online

El CM/CP Ethernet asume las siguientes tareas durante la transferencia de datos a través de una conexión:

- Recibir
  Recepción de datos de Ethernet y transmisión al área de datos de usuario de la CPU.
- Enviar
  Aceptación de datos del área de datos de usuario de la CPU y transmisión de los datos a través de Ethernet.

El establecimiento de la conexión se realiza automáticamente en cuanto se puede acceder al interlocutor.

Conexión HMI

Introducción a la configuración de conexiones

Definición

Una conexión define la asignación lógica de dos interlocutores con el fin de ejecutar servicios de comunicación. Una conexión define lo siguiente:

- Interlocutores que intervienen
- Tipo de conexión (p. ej. conexión HMI)
- Propiedades especiales (p. ej. si una conexión permanece establecida o si se establece y deshace dinámicamente en el programa de usuario y si deben enviarse avisos sobre el estado operativo)
- Vía de conexión

Información importante para configurar conexiones

Al configurar conexiones se asigna un nombre de conexión local en una conexión HMI en calidad de identificación local unívoca.

En la vista de redes, junto a la ficha "Vista general de redes" aparece la ficha "Conexiones". Esta ficha incluye la tabla de conexiones. Una fila de dicha tabla representa una conexión configurada, p. ej. entre un dispositivo HMI y un PLC, junto con sus propiedades.

Información importante sobre el consumo de recursos de conexión

Introducción

Cada conexión requiere recursos de los dispositivos implicados tanto para el punto final como para el punto de transición. El número de recursos de conexión es específico del dispositivo.

Si todos los recursos de conexión de un interlocutor están ocupados, no podrá establecerse ninguna conexión nueva. Esto se reconoce por el hecho de que la nueva conexión se resaltará en rojo en la tabla de conexiones. En este caso, la configuración es incoherente y no puede compilarse.

Conexiones HMI

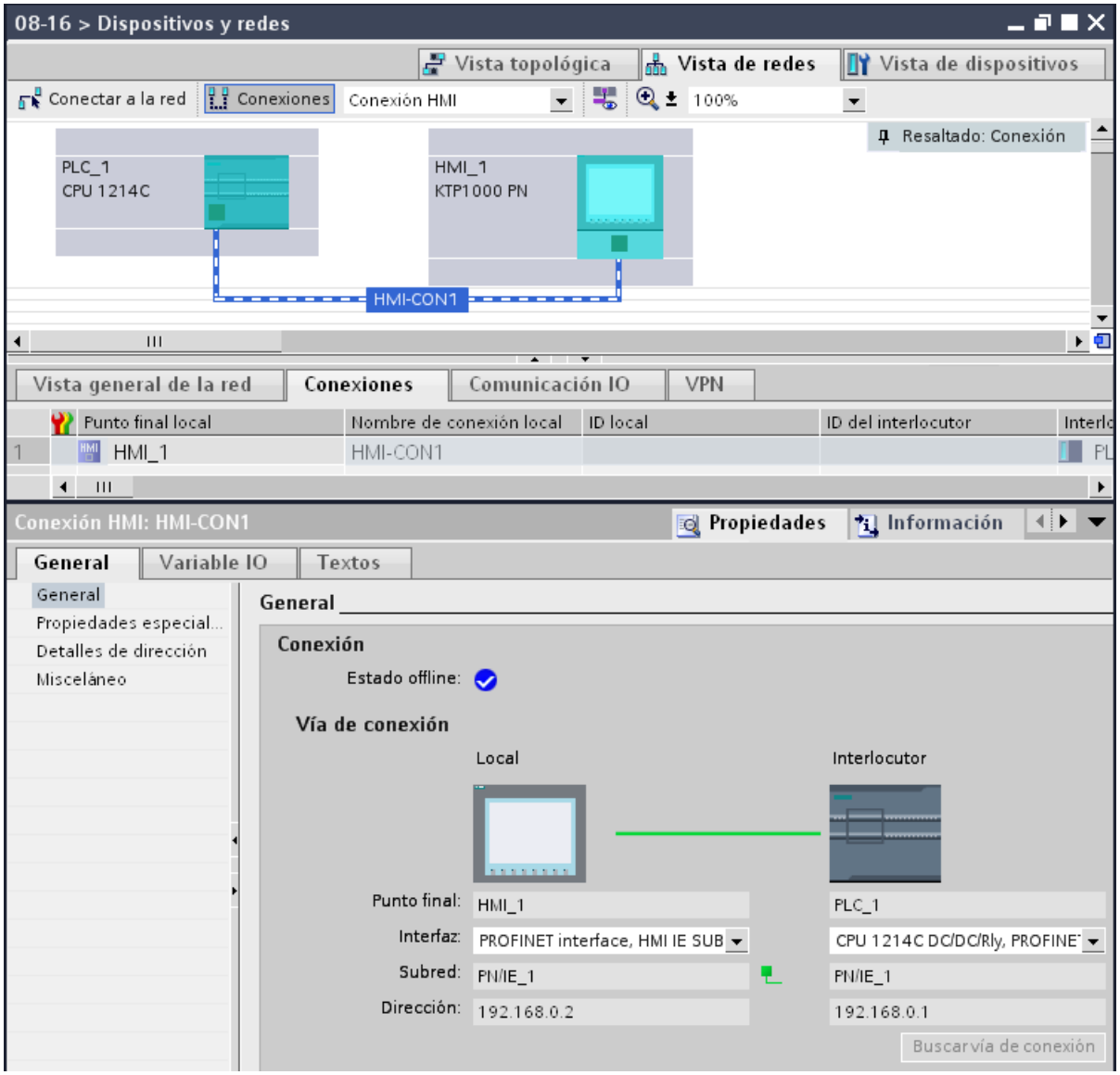
Cada conexión HMI establecida a través de la interfaz PN **integrada** ocupa un recurso de conexión para el punto final en el dispositivo HMI.

El interlocutor (PLC) también requiere un recurso de conexión.

Vistas con información sobre las conexiones configuradas

Desde las vistas descritas a continuación se accede a toda la información y las funciones relacionadas con la configuración y verificación de conexiones.

- Visualización de conexiones en la vista de redes
- Tabla de conexiones
- Ficha "Propiedades" de una conexión en la ventana de inspección



Ventajas

La información representada en estas vistas siempre es actual en cuanto a las acciones de usuario inmediatas. Esto significa que:

- La tabla de conexiones muestra todas las conexiones creadas.
- Si se ha seleccionado una conexión en la tabla de conexiones, rige lo siguiente:
  - La vía de conexión se representa gráficamente en la vista de redes.
  - La ficha "Propiedades" de la ventana de inspección muestra los parámetros de dicha conexión.

La tabla de conexiones

La tabla de conexiones ofrece las funciones siguientes:

- Relación de todas las conexiones del proyecto
- Selección de una conexión y su visualización en la vista de redes
- Cambio de interlocutores
- Visualización de información de estado

Ficha "Propiedades" de una conexión en la ventana de inspección

El cuadro de diálogo de propiedades tiene el significado siguiente:

- Visualización de los parámetros de la conexión
- Visualización de la vía de conexión
- Especificación posterior de las conexiones por medio del botón "Buscar vía de conexión"

Crear una conexión nueva

Alternativas para crear una conexión

Existen las posibilidades siguientes para crear una conexión en la vista de redes:

- Configuración gráfica de la conexión
- Configuración de la conexión desde un cuadro de diálogo

Los diferentes pasos que deben seguirse se explican en los capítulos siguientes.

Requisitos y resultado

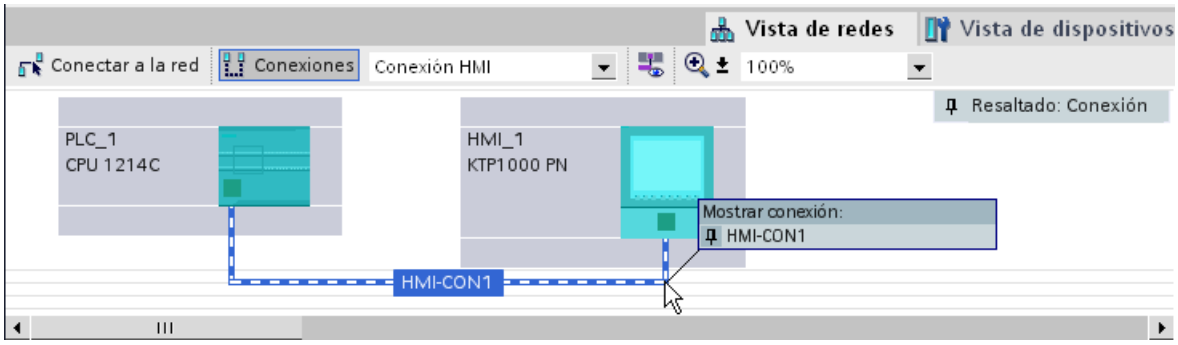
En la vista de redes se han creado los dispositivos con CPUs y los dispositivos HMI entre los que deben configurarse conexiones.

Especificar una conexión

Si ambos interlocutores están conectados a la misma red para el tipo de conexión seleccionado, cree una conexión completamente especificada entre ambos interlocutores mediante la selección gráfica o desde un cuadro de diálogo.

Dicha conexión se registrará automáticamente en la tabla de conexiones del dispositivo HMI y se le asignará un nombre de conexión local.

La representación siguiente muestra una conexión configurada en un dispositivo conectado en red:



Crear una conexión nueva gráficamente

Configurar conexiones gráficamente

En la configuración gráfica de las conexiones, el sistema requiere en algunos casos que se defina la vía de conexión. Seleccione los dispositivos que deben conectarse en la configuración actual.

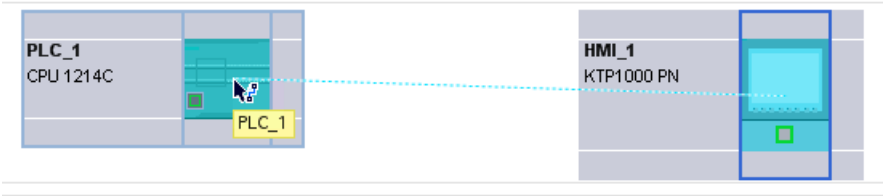
Determinar automáticamente la vía de conexión

Para crear una conexión gráficamente, proceda del siguiente modo:

1. Haga clic en el botón "Conexiones".

   De este modo se activa el modo de conexión para el tipo de conexión seleccionado.
   Esto se reconoce por el comportamiento siguiente:
   En la vista de redes están resaltados en color los dispositivos que pueden utilizarse en el proyecto para el tipo de conexión seleccionado.

2. Arrastre el cursor del ratón, manteniendo pulsado el botón del ratón, desde el dispositivo del que debe partir la conexión hasta el dispositivo en el que debe finalizar la conexión.

3. Suelte el botón del ratón en el dispositivo de destino para crear la conexión entre ambos.

Resultado

- Se ha creado una conexión especificada.
- La vía de conexión se representa resaltada.
- La conexión está registrada en la tabla de conexiones.

Crear una conexión nueva desde un cuadro de diálogo

Configurar conexiones desde un cuadro de diálogo

Determine el dispositivo local y su interlocutor.

Procedimiento

Para crear una conexión desde un cuadro de diálogo, proceda del siguiente modo:

1. En el menú contextual del interlocutor para el que debe crearse la conexión, elija el comando "Crear conexión nueva".
   Se abrirá el cuadro de diálogo "Crear conexión nueva".
2. Seleccione el punto final del interlocutor.
   En el lado derecho del cuadro de diálogo aparecerá una posible vía de conexión apropiada para el punto final seleccionado, si existiera. Las vías incompletas, como en el caso de una CPU no especificada, se marcan con un signo de exclamación sobre fondo rojo.
3. Para salir del cuadro de diálogo pulse "Aceptar".
   Para aplicar la conexión configurada y configurar otras conexiones referentes a otros puntos finales, haga clic en "Aplicar".

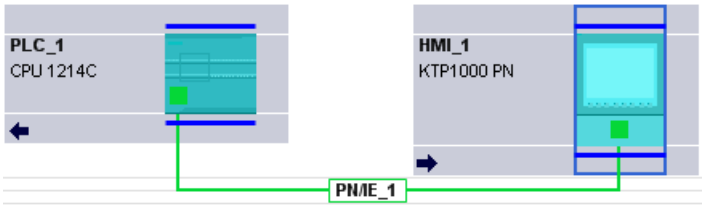
Trabajar en la vista de redes

Resaltar la vía de conexión y el interlocutor en la vista de redes

Para mostrar los interlocutores de todos o algunos tipos de conexión en la vista de redes, proceda del siguiente modo:

1. Haga clic en el botón "Conexiones".

2. En el menú contextual del dispositivo HMI cuyo interlocutor desea visualizar en la vista de redes, elija el comando "Resaltar interlocutores".
3. En el menú que aparece a continuación, elija "Todos los interlocutores".
   El dispositivo local y las CPUs de los dispositivos de destino están seleccionados. El interlocutor local presenta una flecha hacia la derecha y los interlocutores remotos una flecha hacia la izquierda.
4. Abra una lista con los datos de los dispositivos de destino seleccionando la flecha del dispositivo local. Esta función complementaria es útil en las configuraciones de red complejas en las que no se ven todos los dispositivos.



Nota

Es posible visualizar uno de los interlocutores que no se ve en el área de visualización actual de la vista de redes. Haga clic en el interlocutor en la lista que aparece. Resultado: la visualización se desplaza hasta que aparece el interlocutor.

Consulte también

Crear una conexión nueva gráficamente (Página 573)

Trabajar con la tabla de conexiones

Funciones básicas de las tablas

La tabla de conexiones soporta las siguientes funciones básicas para editar una tabla:

- Modificar el ancho de las columnas
- Explicar el significado de una columna, fila o campo mediante tooltips.

Modificar el ancho de las columnas

Para adaptar el ancho de una columna al contenido de modo que se lean los textos de todas las filas, proceda del siguiente modo:

1. Posicione el puntero del ratón en el encabezado de la tabla de conexiones, a la derecha de la columna que debe optimizarse, hasta que el puntero adopte la forma de dos líneas paralelas (como si quisiera modificar el ancho de la columna arrastrando el puntero del ratón).
2. Haga doble clic en este punto.

o

1. Abra el menú contextual en el encabezado de la tabla.
2. Haga clic en
   - "Optimizar ancho de columnas" o
   - "Optimizar ancho de todas las columnas".

Si las columnas son demasiado estrechas, se puede ver todo el contenido de los diferentes campos colocando el puntero del ratón brevemente sobre el campo en cuestión.

Mostrar/ocultar columnas

Mediante el menú contextual de los encabezados de la tabla de conexiones se controla la visualización de las diferentes columnas. Con el comando "Mostrar/ocultar columnas" del menú contextual aparece una vista general de las columnas disponibles. Con las casillas de verificación se controla la visibilidad de las columnas.

Navegar por la tabla de conexiones con las teclas de cursor

Las teclas de cursor FLECHA ARRIBA y FLECHA ABAJO permiten seleccionar una conexión en la tabla de conexiones; la conexión queda seleccionada y se resalta en la vista de redes.

Modificar las propiedades de la conexión

Algunos de los parámetros visualizados en la tabla de conexiones pueden editarse directamente. Para modificar el nombre de una conexión no es necesario navegar hasta la ventana de inspección.

Modificar el interlocutor

El interlocutor de una conexión se modifica del siguiente modo:

1. Seleccione la conexión.
2. Elija el nuevo interlocutor en la lista desplegable activada de la columna "Interlocutor".

Borrar conexiones

Las conexiones configuradas se borran en la vista de redes o en la tabla de conexiones.

En la vista de redes se puede borrar en cada caso una conexión resaltada. En la tabla de conexiones se borran una o varias conexiones.

Procedimiento

Para borrar una conexión, proceda del siguiente modo:

1. Seleccione la conexión que desea borrar:
   - En la vista de redes: seleccione la conexión que desea borrar.
   - En la tabla de conexiones: seleccione las filas de las conexiones que deben borrarse (posibilidad de selección múltiple).
2. Abra el menú contextual con el botón derecho del ratón.
3. Elija el comando "Borrar".

Resultado

La conexión seleccionada se borrará por completo.

Copiar conexiones

Introducción

Las conexiones no se copian individualmente, sino siempre en relación con el proyecto o dispositivo.

Es posible copiar:

- Proyectos enteros
- Uno o varios dispositivos dentro de un proyecto o en varios proyectos

Copiar un proyecto

Al copiar un proyecto se copiarán también todas las conexiones configuradas. No es necesario realizar ajustes en las conexiones copiadas porque siguen siendo coherentes.

Copiar dispositivos

Al copiar dispositivos para los que se han configurado conexiones (dispositivo HMI) también se copiarán las conexiones. Para completar la vía de conexión todavía hay que arrastrar la conexión en red.

Una CPU S7-1200 con un firmware V1.0 es simplemente un servidor de conexiones HMI y no tiene ninguna configuración de conexiones propia. Por ello, al copiar una CPU S7-1200 con un firmware V1.0 no se copiarán conexiones.

Conexiones incoherentes - conexiones sin asignación

Cuando una conexión es incoherente, significa que la estructura de los datos de conexión está destruida o bien la conexión no funciona en relación con el proyecto.

No es posible compilar ni cargar conexiones incoherentes, como tampoco lo es su operación.

En la tabla de conexiones, las conexiones incoherentes se reconocen por el color rojo.

Causas posibles de conexiones incoherentes

- Borrado o modificaciones de la configuración de hardware
- Falta de conexiones en red entre las interfaces del proyecto necesarias para una conexión.
- Rebase por exceso de los recursos de conexión
- Error al realizar una copia de seguridad de los datos por falta de memoria
- Conexiones con un interlocutor no especificado sin indicar la dirección del mismo.

Encontrará información detallada sobre la causa de la incoherencia en la ficha "Compilar" después de la compilación (Edición > Compilar).

Soluciones

Si la conexión no se repara al abrir, modificar o deshacer las propiedades de la misma en la configuración, posiblemente será necesario borrarla y volver a crearla.

Ajustes generales de la conexión HMI

Parámetros de conexión generales

En el grupo de parámetros "General" de las propiedades de la conexión se muestran los parámetros generales que identifican el punto final local de la conexión.

Aquí también existe la posibilidad de asignar la vía de conexión y especificar por completo a los interlocutores.

Propiedades de conexión especiales

Visualización de las propiedades de conexión (no modificable):

- Iniciativa local
  El dispositivo HMI es el que establece siempre la conexión. Esta opción está seleccionada por defecto si la dirección del interlocutor está especificada.
- Unilateral
  Unilateral significa que el interlocutor es el servidor de dicha conexión y que no puede enviar o recibir activamente.
- Enviar avisos de estado operativo
  No relevante para dispositivos HMI.

Detalles de dirección

Visualización de los detalles de dirección de la conexión HMI. Si un interlocutor no está especificado, pueden modificarse los valores de rack y slot. Todos los demás valores se tomarán de la configuración actual y no podrán modificarse.

Misceláneo

Visualización de los puntos de acceso para la conexión online entre dispositivo HMI e interlocutor.

Conexiones - significado y selección

Introducción a la configuración de conexiones

¿Para qué configurar conexiones?

Una conexión de comunicación o, simplemente, conexión define la asignación lógica de dos interlocutores con el fin de ejecutar servicios de comunicación.

Las conexiones las puede configurar uno mismo o, si se cumplen ciertos requisitos, también pueden configurarse por programa.

En este capítulo se explica cómo definir conexiones, qué particularidades hay que tener en cuenta y qué instrucciones de comunicación se pueden utilizar en el programa de usuario.

Definición

Una conexión define lo siguiente:

- los interlocutores que intervienen
- el tipo de conexión (p. ej. conexión S7, ISO-on-TCP, ISO-Transport, FDL, UDP, de correo electrónico o TCP)
- propiedades especiales como p. ej.
  - el comportamiento de la instalación
    Una conexión puede quedar establecida de forma permanente o puede establecerse y deshacerse según sea necesario.
  - la transmisión de avisos de estado operativo
- la vía de comunicación
  - dentro de los dispositivos (CPU - CP)
  - en la red (subredes / router)

Información importante para configurar conexiones

En la configuración de conexiones se asigna una identificación local unívoca por conexión, la "ID local". En la parametrización de los bloques de comunicación sólo se necesita esta ID local. La ID local está asignada unívocamente a las direcciones del interlocutor local y remoto.

Cuando un punto final de la conexión es una aplicación de una estación PC, se utiliza el nombre de la conexión como "ID local" para la identificación de la conexión.

Después de abrir el proyecto se visualiza la tabla de conexiones de todo el proyecto. Si desactiva la opción "Mostrar todas las conexiones" en el menú contextual de la tabla de conexiones, a partir de entonces se mostrarán sólo las conexiones específicas de los objetos seleccionados. Las conexiones creadas para los objetos siguientes se pueden visualizar aparte:

- Módulos programables que pueden ser punto final de una conexión
- Módulos de comunicación (CPs) que se utilizan en la vía de conexión
- Subredes

Las direcciones del interlocutor local y remoto están especificadas por la conexión de los dispositivos en red y se asignan a las conexiones durante la configuración. Las siguientes conexiones son excepciones:

- Conexión UDP libre
  En la conexión UDP libre la dirección se indica en la interfaz de comunicación del programa de usuario.
- Conexión programada

Nota

El término "conexión" también se utiliza en la comunicación SIMATIC con UDP. El motivo es que, durante la configuración, los interlocutores se asignan entre sí de forma que se "conectan" lógicamente. Realmente, con UDP no se establece ninguna conexión explícita entre los interlocutores.

La conexión en red y la selección de la vía determinan el estado de configuración

Dependiendo de si los interlocutores existen y están conectados en el proyecto, los parámetros de dirección de los puntos finales de la conexión pueden estar especificados por completo o sólo parcialmente.

- Especificados por completo
  - Las direcciones y parámetros de red del interlocutor local y remoto están definidos.
  - La conexión está lista para funcionar tras cargar los parámetros de conexión en el dispositivo.
- Sin especificar (parcialmente especificada)
  Sólo está definido el interlocutor local. El interlocutor para el tipo de conexión seleccionado no está conectado en red, o bien se encuentra fuera del proyecto, o bien no debe definirse debido a un manejo especial.
  Caso de aplicación:
  - La configuración del hardware está incompleta, de forma que todavía no es posible realizar la conexión en red completa de los dispositivos.
  - Debe establecerse una disponibilidad de recepción para los interlocutores deseados (no especificados).

  La conexión está lista para funcionar de forma condicionada tras cargar los parámetros de conexión en el dispositivo.

Conexiones programadas

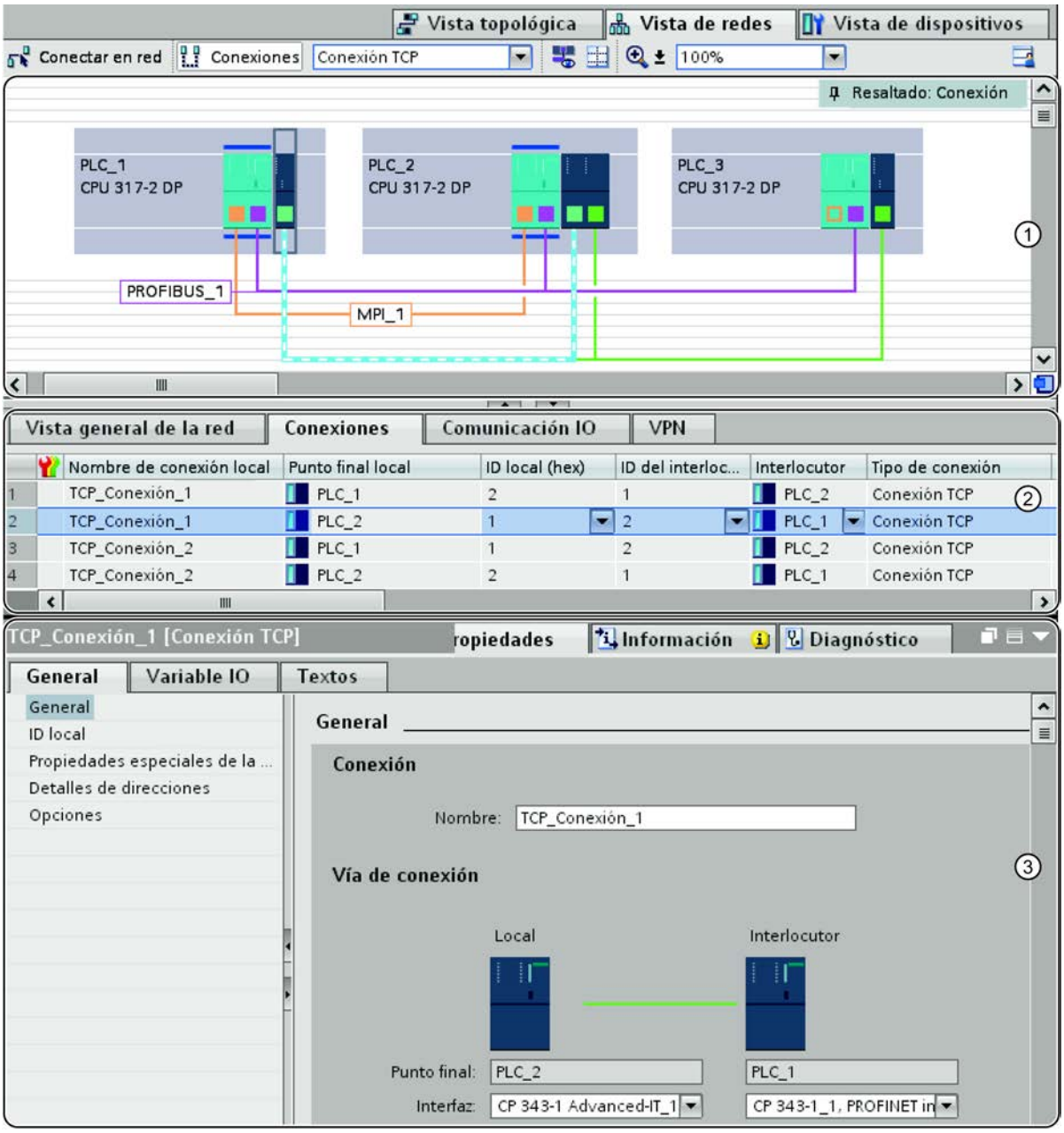
En determinados campos de aplicación es ventajoso no crear las conexiones de forma estática a través de la configuración. Por eso, de forma alternativa existe la posibilidad de establecer determinados tipos de conexión a través de una aplicación específica mediante un programa y hacerlo así de forma dinámica si es necesario.

Características de la configuración de conexiones

La vista de redes es una plataforma clara para la configuración de conexiones entre dispositivos configurados.

Funciones principales

- Configuración gráfica de conexiones en la vista de redes
  Con unos pocos clics del ratón se establece una conexión gráficamente.
- Resaltar posibles interlocutores
  Se selecciona el tipo de conexión y los posibles interlocutores se resaltan automáticamente.
- Búsqueda automática de vías de conexión
  Se ordena la asignación automática de las mejores vías de conexión posibles o la selección de vías de conexión alternativas.
- Aplicación automática de configuraciones
  Si se configuran módulos posteriormente y se conectan en red, las conexiones ya configuradas y soportadas por el módulo se asignan automáticamente.
- Selección cómoda de la visualización de conexiones
  Si se selecciona un objeto en la vista de redes, p. ej. una subred o la interfaz de un módulo apto para la comunicación, se muestra la correspondiente conexión configurada en una tabla de conexiones.
- Consideración de los recursos disponibles
  Mientras se necesiten recursos de conexión para un tipo de conexión, se tiene en cuenta una determinada limitación al establecer conexiones. El número máximo de recursos de conexión varía tanto con los tipos de dispositivo CPU como con los tipos de dispositivo CP.



① Visualización de conexiones en la vista de redes

② Tabla de conexiones

③ Ficha "Propiedades" de una conexión en la ventana de inspección

Campos de aplicación de los tipos de conexión

Introducción

Los siguientes apartados ofrecen una breve panorámica de los tipos de conexión que se pueden utilizar para la comunicación según el tipo de dispositivo y la finalidad de uso.

Información adicional

Para más información sobre las posibilidades de comunicación en SIMATIC y las finalidades de uso de los tipos de conexión consulte el manual sobre las comunicaciones con SIMATIC.

Conexiones S7

Las conexiones S7 ofrecen, entre otras, las siguientes características:

- Tipo de conexión utilizable en todos los dispositivos S7;
- Utilizables en los siguientes tipos de subred: MPI, PROFIBUS, Industrial Ethernet;
- Si se utilizan los SFBs BSEND/BRCV: Transferencia segura de datos entre estaciones SIMATIC S7-400/1500; p. ej. intercambio de contenidos de bloques de datos (hasta 64 KB);
- Con una CPU 317-2 PN/DP o con una CPU 31x y un CP también es posible la transferencia segura de datos a S7-300, S7-400 y S7-1500 utilizando los FBs BSEND/BRCV;
- Si se utilizan los SFBs USEND/URCV: transmisión rápida y no segura de datos independientemente de la ejecución temporal del interlocutor; p. ej. para avisos de operación y de mantenimiento;
- Acuse de la transferencia de datos del interlocutor en el nivel 7 del modelo de referencia ISO.

Conexión FDL

PROFIBUS FDL (Fieldbus Data Link) se identifica por las siguientes características:

- Transferencia de datos a un interlocutor (p. ej. SIMATIC S5 o PC) que soporte la transmisión o recepción según la función SDA (Send Data with Acknowledge);
- La recepción de los datos se confirma con un acuse del servicio FDL del interlocutor;
- tipo de subred utilizable: PROFIBUS;
- Cumple la norma EN 50170 Vol.2 PROFIBUS;
- Puede asignarse al nivel 2 del modelo de referencia ISO;
- Interfaz en el programa de usuario S7: SEND/RECEIVE;
- Servicios FDL disponibles en el PC como funciones C.

Conexión de transporte ISO

La conexión de transporte ISO se identifica por las siguientes características:

- "Bloqueo de datos" apto para grandes cantidades de datos;
- Comunicación con un interlocutor (p. ej. SIMATIC S5 o PC) que soporte la transmisión o recepción de datos según el transporte ISO.
- Tipo de subred utilizable: Industrial Ethernet;
- La recepción de los datos se confirma con un acuse del servicio de transporte ISO del interlocutor;
- Interfaz en el programa de usuario S7: SEND/RECEIVE y FETCH/WRITE;
- El servicio de transporte ISO (ISO 8073 class 4) corresponde al nivel 4 del modelo de referencia ISO;
- Servicios de transporte ISO disponibles en el PC como funciones C.

Conexión ISO-on-TCP

La conexión ISO-on-TCP se identifica por las siguientes características:

- Corresponde al estándar TCP/IP (Transmission Control Protocol/Internet Protocol) con la ampliación RFC 1006 según el nivel 4 del modelo de referencia ISO. RFC 1006 describe cómo se pueden reproducir en TCP los servicios del nivel 4 ISO;
- Comunicación con un interlocutor (p. ej. PC o sistema de terceros) que soporte la transmisión o recepción de datos según ISO-on-TCP.
- Tipo de subred utilizable: Industrial Ethernet;
- La recepción de los datos se confirma con un acuse;
- Interfaz en el programa de usuario STEP 7: SEND/RECEIVE y FETCH/WRITE;
- Servicios ISO-on-TCP disponibles en el PC como funciones C.

Conexión TCP

La conexión TCP se identifica por las siguientes características:

- Corresponde al estándar TCP/IP (Transmission Control Protocol/Internet Protocol);
- Permite la comunicación con un interlocutor (p. ej., un PC o un sistema de terceros) que soporte la transmisión o recepción de datos según TCP/IP;
- Tipo de subred utilizable: Industrial Ethernet;
- Interfaz en el programa de usuario STEP 7: SEND/RECEIVE y FETCH/WRITE;
- Normalmente es posible utilizar la implementación TCP/IP del sistema operativo del PC.

Conexión UDP

La conexión UDP (User Datagramm Protocol) se identifica por las siguientes características:

- Tipo de subred utilizable: Industrial Ethernet (protocolo TCP/IP);
- Transferencia no segura de bloques de datos relacionados entre dos dispositivos;
- Interfaz en el programa de usuario S7: SEND/RECEIVE.

Conexión de correo electrónico

La conexión de correo electrónico se identifica por las siguientes características:

- Tipo de subred utilizable: Industrial Ethernet (protocolo TCP/IP);
- Permite p. ej. la transmisión de datos de proceso de bloques de datos por correo electrónico a través de un CP con funcionalidad IT (IT-CP);
- Con la conexión de correo electrónico se establece el servidor de correo a través del cual se entregan todos los correos electrónicos enviados por un IT-CP.

Conexión FTP (conexión TCP configurada de forma especial)

La conexión FTP se utiliza para la ejecución de secuencias de petición FTP.

Con las funciones de transferencia de archivos (FTP) los CPs con funcionalidad IT ofrecen la posibilidad de transferir archivos desde o hacia dispositivos S7.

Conexión HMI

Auto-Hotspot

Consulte también

USEND: Envío de datos sin coordinación (Página 7683)

URCV: Recibir datos sin coordinación (Página 7685)

BSEND: Transmitir datos en segmentos (Página 7688)

BRCV: Recibir datos en segmentos (Página 7691)

USEND: Envío de datos sin coordinación (Página 5255)

URCV: Recibir datos sin coordinación (Página 5256)

BSEND: Transmitir datos en segmentos (Página 5258)

BRCV: Recibir datos en segmentos (Página 5261)

Selección de los tipos de conexión

Selección del tipo de conexión

El tipo de conexión que se selecciona depende, aparte de las propiedades de transferencia, de la subred y del protocolo de transferencia a través del cual se establece la conexión. Además, el tipo de dispositivo limita los tipos de conexión en los correspondientes puntos finales.

Las instrucciones o servicios que se utilizan en la interfaz (puntos finales de la conexión) dependen del tipo de conexión y del tipo de dispositivo.

La siguiente tabla muestra estas dependencias:

Tipo de conexión	Tipo de subred / protocolo	Tipo de dispositivo: conexión entre SIMATIC ...	Interfaz
Conexión S7	MPI, PROFIBUS, Industrial Ethernet	S7 - S7, S7 - PG/PC, S7 - PG/PC con WinCC, PG/PC - PG/PC Interlocutor S7 en otro proyecto (S7, PG/PC con WinCC)	**Instrucciones para S7:** USEND, URCV, BSEND, BRCV, GET, PUT, START, STOP, RESUME, STATUS, USTATUS **PG/PC:** servicios OPC o interfaces de programación
Conexión FDL	PROFIBUS (protocolo FDL)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - dispositivo de terceros, PG/PC - PG/PC Interlocutor S7 en otro proyecto (S7, S5, PG/PC, dispositivo de terceros)	**Instrucciones para S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV **PG/PC:** servicios OPC
Conexión de transporte ISO	Industrial Ethernet (protocolo de transporte ISO)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - dispositivo de terceros, PG/PC - PG/PC S7 - sin especificar Interlocutor S7 en otro proyecto (S7, S5, PG/PC, dispositivo de terceros, sin especificar)	**Instrucciones para S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV AG_LOCK AG_UNLOCK **PG/PC:** servicios OPC
Conexión ISO-on-TCP	Industrial Ethernet (protocolo TCP/IP)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - dispositivo de terceros, PG/PC - PG/PC, S7 - sin especificar Interlocutor S7 en otro proyecto (S7, S5, PG/PC, dispositivo de terceros, sin especificar)	**Instrucciones para S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV AG_LOCK AG_UNLOCK **PG/PC:** servicios OPC

Tipo de conexión	Tipo de subred / protocolo	Tipo de dispositivo: conexión entre SIMATIC ...	Interfaz
Conexión TCP	Industrial Ethernet (protocolo TCP/IP)	S7 - S7, S7 - S5, S7 - PC/PG, PG/PC - PG/PC S7 - dispositivo de terceros, S7 - sin especificar Interlocutor S7 en otro proyecto (S7, S5, PG/PC, dispositivo de terceros, sin especificar)	**Instrucciones para S7:** AG_SEND*, AG_RECV*, AG_LSEND**, AG_LRECV** AG_LOCK AG_UNLOCK **PG/PC:** servicios OPC
Conexión UDP	Industrial Ethernet (protocolo TCP/IP)	S7 - S7, S7 - S5, S7 - PC/PG, S7 - dispositivo de terceros, S7 - sin especificar Interlocutor S7 en otro proyecto (S7, S5, PG/PC, dispositivo de terceros, sin especificar)	**Instrucciones para S7:** AG_SEND, AG_RECV, AG_LSEND, AG_LRECV
Conexión de correo electrónico	Industrial Ethernet (protocolo TCP/IP)	S7 - sin especificar (S7 - servidor de correo electrónico), S7 - PC/PG (S7 - servidor de correo electrónico)	**Instrucciones para S7:** AG_SEND, AG_LSEND
Conexión FTP (conexión TCP configurada de forma especial)	Industrial Ethernet (protocolo TCP/IP)	S7 - sin especificar	**Instrucciones para S7:** FTP_CONNECT, FTP_STORE, FTP_RETRIEVE, FTP_DELETE, FTP_QUIT

Conexión a dispositivos Broadcast y Multicast

Nota

Para tipos de conexión especiales existe la posibilidad de no sólo seleccionar un único interlocutor, sino varios (dispositivos Broadcast y Multicast). Los interlocutores "Con todos los dispositivos Broadcast" o "...Multicast" pueden seleccionarse en el cuadro de diálogo para la introducción de una nueva conexión.

- Es posible establecer una conexión con "todos los dispositivos Broadcast" (transmisión simultánea a todos los receptores Broadcast) para los tipos de conexión FDL y UDP.
- Es posible establecer una conexión con "todos los dispositivos Multicast" (transmisión simultánea a varios dispositivos) para los tipos de conexión FDL y UDP.

Selección de los tipos de conexión

Selección del tipo de conexión

Según la tarea de automatización de que se trate, elija un servicio de comunicación u otro. La selección del servicio de comunicación influye en lo siguiente:

- las funciones disponibles
- el momento de establecimiento de la conexión
- las actividades que han de ejecutarse (p. ej. configuración de conexiones, programación de instrucciones)

La tabla siguiente ofrece un resumen de los servicios de comunicación disponibles.

Servicio de comunicación	funcionalidad	Vía interfaz:		
		PN	DP	serie
Comunicación PG	Para puesta en marcha, test, diagnóstico	X	X	-
Comunicación HMI	Para manejo y visualización	X	X	-
Comunicación E/S	Intercambio de datos, p. ej. entre: • controladores IO y dispositivos IO • maestro DP y esclavo DP	X	X	-
Comunicación S7	Intercambio de datos de cliente a servidor o de cliente a cliente Requiere configuración de conexiones Mediante instrucciones PUT/GET, BSEND/BRCV o USEND/URCV	X	X	-
Comunicación abierta vía TCP/IP	Intercambio de datos vía PROFINET con protocolo TCP/IP Con configuración de conexiones: instrucciones TSEND_C/TRCV_C o TSEND/TRCV Solo mediante programa de usuario: instrucciones TCON, TSEND, T_RCV, T_DISCON o TSEND_C y TRCV_C	X	-	-
Comunicación abierta vía ISO on TCP	Intercambio de datos vía PROFINET con protocolo ISO on TCP Con configuración de conexiones: instrucciones TSEND_C/TRCV_C o TSEND/TRCV Solo mediante programa de usuario: instrucciones TCON, TSEND, T_RCV, T_DISCON o TSEND_C y TRCV_C	X	-	-
Comunicación abierta vía UDP	Intercambio de datos vía PROFINET con protocolo UDP Con configuración de conexiones: instrucciones TSEND_C/TRCV_C o TSEND/TRCV Solo mediante programa de usuario: instrucciones TCON, TSEND, T_RCV, T_DISCON o TSEND_C y TRCV_C	X	-	-

Servicio de comunicación	funcionalidad	Vía interfaz:		
		PN	DP	serie
Comunicación abierta vía ISO (solo CPs con interfaz PN)	Intercambio de datos vía PROFINET con protocolo ISO Con configuración de conexiones: instrucciones TSEND_C/TRCV_C o TSEND/TRCV Solo mediante programa de usuario: instrucciones TCON, TSEND, T_RCV, T_DISCON o TSEND_C y TRCV_C	X	-	-
Comunicación abierta vía correo electrónico	Intercambio de datos vía correo electrónico Solo mediante programa de usuario: instrucción TMAIL_C	X	-	-
Comunicación abierta vía FTP (solo CPs con interfaz PN)	Intercambio de datos vía FTP (File Transfer Protocol) Solo mediante programa de usuario: instrucción FTP_CMD	X	-	-
Comunicación abierta vía FDL (solo CMs con interfaz DP)	Intercambio de datos vía PROFIBUS con protocolo FDL Con configuración de conexiones: instrucciones TSEND_C/TRCV_C o TSEND/TRCV Solo mediante programa de usuario: instrucciones TCON, TSEND, T_RCV, T_DISCON o TSEND_C y TRCV_C	-	X	-
Acoplamiento punto a punto	Intercambio de datos punto a punto con protocolo Freeport, 3964 (R), USS o Modbus Mediante instrucciones SEND_P2P/RCV_P2P o MB_COMM_LOAD, MB_MASTER , MB_SLAVE	-	-	X
Routing S7	Intercambio de datos fuera de los límites de red, p. ej. para parametrización, test y diagnóstico	X	X	-
Routing de registros	Ampliación del routing S7, p. ej. para parametrización y diagnóstico de dispositivos de campo en PROFIBUS vía un sistema de ingeniería en una interfaz PROFINET	X	X	-
Servidor web	Intercambio de datos vía Internet, p. ej. para diagnóstico	X	-	-
SNMP (Simple Network Management Protocol)	Intercambio de datos vía protocolo estándar SNMP, para diagnóstico y parametrización de redes	X	-	-
Sincronización horaria	Vía NTP (Network Time Protocol); la CPU es cliente	X	-	-
	La CPU es maestro o esclavo horario	-	X	-

Conexión a dispositivos Broadcast y Multicast

Nota

Para tipos de conexión especiales existe la posibilidad de seleccionar, no un único interlocutor, sino varios (dispositivos Broadcast y Multicast). Los interlocutores "Con todos los dispositivos Broadcast" o "...Multicast" pueden seleccionarse en el cuadro de diálogo para la introducción de una conexión nueva.

- Es posible establecer una conexión con "todos los dispositivos Broadcast" (transmisión simultánea a todos los receptores Broadcast) para los tipos de conexión FDL y UDP.
- Es posible establecer una conexión con "todos los dispositivos Multicast" (transmisión simultánea a varios dispositivos) para los tipos de conexión FDL y UDP.

Instrucciones para tareas de comunicación

Interfaz del programa de usuario

Para la interfaz de comunicación del programa de usuario de SIMATIC S7-300/400/1500 existen instrucciones especiales. Aquí se presenta un resumen de las instrucciones para cada uno de los tipos de conexión.

Para S7-1500 se emplean las instrucciones PUT/GET. Para las instrucciones de S7-300/400, véanse las tablas siguientes.

Instrucciones para conexiones S7

Los bloques de función de sistema asignados a las instrucciones están integrados en las CPUs de S7-400.

En las CPUs y los CPs más recientes para S7-300, existe la posibilidad de operar activamente (es decir, como cliente) la comunicación S7 a través de la interfaz del CP. Las instrucciones tienen el mismo nombre que los SFBs de la S7-400, pero tienen que llamarse cíclicamente en el programa de usuario de la CPU S7-300.

El CP tiene que soportar la función de cliente para comunicación S7.

Una CPU S7-300 con interfaz PROFINET integrada se puede configurar también como cliente para la comunicación S7. Se utilizan los mismos bloques que el caso mencionado anteriormente para S7-300 con CP. La funcionalidad de cliente sólo está disponible en la interfaz PROFINET.

Nombre	Descripción breve
USEND URCV	Intercambio de datos no coordinado a través de un SFB de transmisión y recepción. Longitud máx. SFB 8/9: 440 bytes divididos en 4x100 bytes. Longitud máx. FB 8/9: 160 bytes.
BSEND BRCV	Intercambio de bloques de datos de longitud variable entre un SFB de transmisión y un SFB de recepción. Longitud máx. SFB 12/13: 64 Kbytes Longitud máx. FB 12/13: 32 Kbytes

Nombre	Descripción breve
GET	Leer datos de un dispositivo remoto Longitud máx. SFB 14: 400 bytes divididos en 4x100 bytes Longitud máx. FB 14: 160
PUT	Escribir datos en un dispositivo remoto Longitud máx. SFB 15: 400 bytes divididos en 4x100 bytes Longitud máx. FB 15: 160
START	Ejecutar un rearranque completo (en caliente) en un dispositivo remoto
STOP	Poner un dispositivo remoto en el estado "STOP"
RESUME	Ejecutar un rearranque en un dispositivo remoto
STATUS	Consulta selectiva del estado de un dispositivo remoto
USTATUS	Recepción de avisos de estado de dispositivos remotos
CONTROL	Consultar el estado de la conexión perteneciente a una instancia SFB
C_CNTRL	Consultar el estado de una conexión (para CPUs S7-300)

Instrucciones para la comunicación S7 (Página 7666)

Instrucciones para conexiones punto a punto

Para el tipo de conexión punto a punto se pueden utilizar los SFBs BSEND, BRCV, GET, PUT y STATUS (véase tabla superior).

Además se puede utilizar el SFB PRINT:

Nombre	Descripción breve
PRINT	Enviar datos a una impresora

Instrucciones para conexión FDL, ISO-on-TCP, UDP y TCP, así como para conexión de correo electrónico

Nombre	Descripción breve
AG_SEND	Envía datos al interlocutor a través de una conexión configurada (en CPs más antiguos y en CPs para S7-400 es aplicable la limitación siguiente para este FC: <= 240 bytes).
AG_RECV	Recibe datos del interlocutor a través de una conexión configurada (en CPs más antiguos y en CPs para S7-400 es aplicable para este FC la limitación: <= 240 bytes y no correo electrónico).
AG_LSEND	Envía datos al interlocutor a través de una conexión configurada.
AG_LRECV	Recibe datos del interlocutor a través de una conexión configurada (no correo electrónico).
AG_LOCK	Bloqueo del acceso externo a los datos con FETCH/WRITE (no en UPD, correo electrónico).
AG_UNLOCK	Habilitación del acceso externo a los datos con FETCH/WRITE (no en UPD, correo electrónico).
AG_CNTRL	Diagnosticar conexiones con el programa de usuario.

Instrucciones para los Servicios de comunicación abiertos (interfaz SEND/RECEIVE) (Página 8052)

Instrucciones para los Servicios de comunicación abiertos (interfaz SEND/RECEIVE) (Página 8194)

Instrucciones para servicios FTP mediante conexiones TCP (conexión FTP)

Nombre	Descripción breve
FTP_CONNECT	Establecer conexión FTP
FTP_STORE	Transferir bloque de datos (File DB) del cliente FTP al servidor FTP.
FTP_RETRIEVE	Transferir archivo del servidor FTP al cliente FTP.
FTP_DELETE	Borrar archivo en el servidor FTP.
FTP_QUIT	Deshacer la conexión FTP designada mediante la ID.

Instrucciones para servicios FTP (Página 8167)

Instrucciones para servicios de comunicación abiertos y programados con conexión ISO-on-TCP, UDP y TCP, así como conexión de correo electrónico a través de CPs

Nombre	Descripción breve
IP_CONFIG	Configurar conexiones e IP del CP con control por programa.

Instrucciones para enlaces programados (Página 8148)

Instrucciones para servicios de comunicación abiertos y programados (Open User Communication) con conexión ISO-on-TCP, UDP y TCP a través de CPU o CP*)

*) Observación: La comunicación a través de CP con estas instrucciones sólo es posible mediante ISO-on-TCP.

Nombre	Descripción breve
TCON	Establecimiento de la conexión
TDISCON	Desconexión
TSEND	Envío de datos
TRCV	Recibir datos
TUSEND	Envío de datos; protocolo UDP orientado a la no-conexión según RFC 768
TURCV	Recepción de datos; protocolo UDP orientado a la no-conexión según RFC 768

Vista general (Página 7725)

Recursos de conexión

Recursos disponibles en la configuración de conexiones

Recursos de conexión

Algunos servicios de comunicación necesitan conexiones. Durante el funcionamiento, las conexiones de comunicación exigen recursos de memoria y programación de las CPUs, los CPs y CMs que intervienen en la ruta de comunicación (p. ej. áreas de memoria del sistema operativo de la CPU). Esto afecta, por ejemplo dentro de un controlador (PLC), a la CPU como

punto final de la conexión, así como a las ampliaciones con CPs como interfaces a las subredes. En la mayoría de los casos, para cada conexión se ocupa un recurso por cada CPU/CP/CM. En la comunicación HMI, por cada conexión HMI se necesitan hasta 3 recursos.

En cada sistema de automatización se dispone de un número diferente de recursos de conexión:

- En los sistemas de automatización S7-300, la CPU determina el número de recursos posibles para conexiones. Mediante CPs agregados puede aumentarse el número máximo de recursos para conexiones.
- En los sistemas de automatización S7-400, la CPU determina el número de recursos posibles para conexiones S7. Los recursos también pueden emplearse para otros tipos de conexiones; mediante los CPs agregados puede aumentarse el número máximo de recursos para otros tipos de conexiones.
- En sistemas de automatización S7-1500, la CPU empleada determina el límite superior de recursos de conexión válido para todo el sistema de automatización. La propia CPU controla parte de estos recursos de conexión; otros recursos se agregan vía CPs y CMs. Pero con independencia del número de CPs y CMs agregados, no puede sobrepasarse el número total máximo de recursos de conexión determinado por el tipo de CPU.

Recursos de conexión disponibles

La capacidad máxima dentro del sistema de automatización lo determina siempre la CPU, bien por el límite superior del sistema determinado por la CPU (S7-1500), bien por la suma de recursos útiles de la CPU y el CP (S7-300/400). En la mayoría de los casos los recursos de conexión disponibles de una CPU, un CP o un CM se distribuyen automáticamente en las interfaces en función de las necesidades concretas.

Cada CPU incluye en su configuración recursos de conexión reservados para comunicación PG, HMI y de servidor web. Paralelamente existen recursos disponibles que pueden utilizarse para cualquier comunicación HMI o S7 y para comunicación abierta (S7-1500).

La configuración de los dispositivos permite determinar los recursos disponibles del modo siguiente:

- Selección del tipo de dispositivo del punto final de la conexión (CPU)
- Número y tipo de módulos de comunicación adicionales (CP, CM)

¿Cuándo se ocupan los recursos de conexión?

Los recursos que realmente se requieren durante el funcionamiento dependen de qué funciones de comunicación se estén utilizando en un determinado momento y de qué conexiones están establecidas en un determinado momento.

Por eso, en el momento de la configuración sólo se pueden realizar comprobaciones limitadas manteniendo determinadas capacidades y la disponibilidad de recursos de comunicación. La configuración de conexiones ofrece la máxima ayuda posible.

¿Qué factores influyen en los recursos disponibles en la configuración de conexiones?

Al configurar los dispositivos se determinan los recursos de comunicación:

- Número de recursos de conexión S7 disponibles en la CPU (S7-300/400)
- Número de recursos de conexión para S7 y comunicación abierta en la estación (S7-1500)
- Número de recursos de conexión específicos de protocolo en los CPs o los CMs
- Número de CPs o CMs disponibles
- Creación automática de la conexión programada o configurada

¿Cómo se comporta la configuración de conexiones?

La configuración de conexiones comprueba durante la entrada las siguientes circunstancias y se comporta del siguiente modo:

- Al crear conexiones S7
  ¿Se ha asignado el número máximo de recursos de conexión S7 disponibles? En caso afirmativo, a través de la interfaz seleccionada no puede configurarse ninguna conexión S7 adicional.
  En S7-300 puede ampliarse la capacidad mediante otros CPs.
- Al crear cualquier tipo de conexión
  ¿Se ha asignado el número máximo de recursos de conexión específicos de protocolo en la interfaz (CP)? En caso afirmativo, a través de la interfaz seleccionada no puede configurarse ninguna conexión adicional del tipo seleccionado.
  En S7-300/400 puede ampliarse la capacidad mediante otros CPs.
- Al crear cualquier tipo de conexión con OPC (estaciones PC)
  En conexiones con OPC como punto final, generalmente se pueden configurar más conexiones de las que el CP utilizado en la estación PC puede operar en un determinado momento. Esto está relacionado con el hecho de que en las propiedades de conexión se diferencia entre "mantener de forma permanente" y "establecer a petición".

Si es necesario, al compilar la configuración de conexiones se emite una advertencia, o un mensaje de error en caso de que no pueda asignarse la interfaz.

Nota

Recursos de conexión S7 agotados

Cuando el número máximo de recursos de conexión S7 está reservado, es posible que durante el funcionamiento no quede disponible ningún recurso de conexión S7, que se requiere adicionalmente para la transferencia de datos entre CPU y CP. En tal caso, se rechazan las correspondientes peticiones de comunicación.

Por eso, debe asegurarse un número suficiente de recursos de conexión S7 no ocupados dependiendo de las funciones de comunicación que se requieren en total.

El número de recursos de conexión S7 reservados para la CPU S7-300 o S7-1500 seleccionada puede consultarse en "Recursos de conexión", dentro de la navegación local. Para S7-300 se muestran solo los recursos de conexión de la CPU; para S7-1500 se muestran también los recursos de los CPs y CMs.

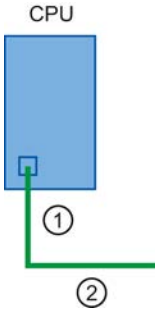
Consumo de recursos de conexión

Cada conexión requiere recursos de los dispositivos implicados tanto para el punto final como para el punto de transición (p. ej. CP, CM). El número de recursos de conexión disponibles es específico de la CPU, el CP o el CM.

Si todos los recursos de conexión de un interlocutor están ocupados, no podrá establecerse ninguna nueva conexión. A continuación se consideran todos los tipos de comunicación individualmente. Sin embargo, si se consideran los recursos de conexión disponibles, es posible cualquier combinación.

Conexiones S7

Cada conexión S7 establecida a través de la interfaz MPI/PROFIBUS DP/PN **integrada** ocupa un recurso de conexión para el punto final en la CPU. Esto se aplica a todas las CPUs S7-300/400/1500.



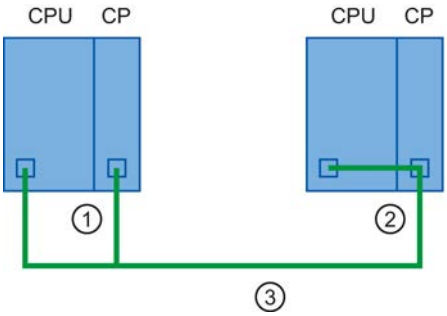
① Recurso de conexión S7 ocupado en la CPU

② Subred (p. ej. MPI, PROFIBUS DP o Industrial Ethernet o PROFINET)

- CPU S7-300/400
  Se ocupa un recurso de conexión S7 en la CPU.
- CPU S7-1500
  Se ocupa un recurso de conexión S7 en la CPU y, por tanto, también otro en la estación.

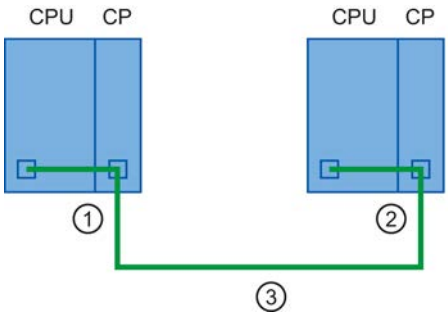
En las conexiones S7 a través de una interfaz CP **externa** deben distinguirse los siguientes casos:

- CPU S7-300
  - Conexión S7 unilateral
    En la CPU (para el punto final) y en el CP (punto de transición) se ocupa un recurso de conexión S7 por cada conexión S7.



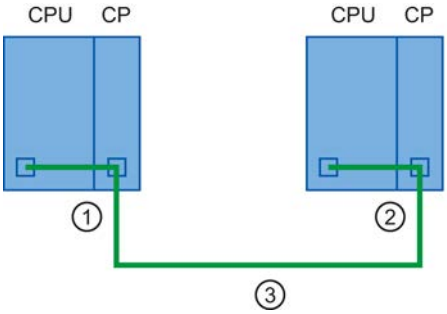
① Un recurso de conexión S7 en la CPU y uno en el CP del punto final local
② Interlocutor
③ Subred, p. ej. Industrial Ethernet

  - Conexión S7 bilateral
    El el CP se ocupa un recurso S7 por cada conexión S7.



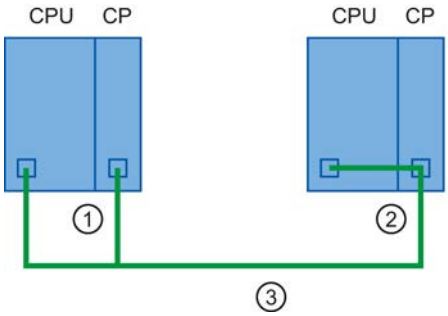
① Un recurso de conexión S7 en el CP del punto final local
② Interlocutor
③ Subred, p. ej. Industrial Ethernet

- CPU S7-400
  En la CPU (para el punto final) y en el CP (punto de transición) se ocupa un recurso de conexión S7 por cada conexión S7.



① Un recurso de conexión S7 en la CPU y uno en el CP del punto final local
② Interlocutor
③ Subred, p. ej. Industrial Ethernet

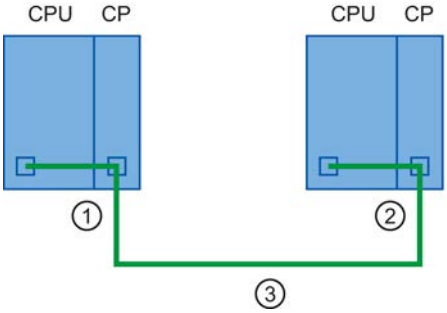
- CPU S7-1500
  – Conexión S7 unilateral
    Se ocupa un recurso de conexión S7 en la CPU por cada conexión S7 y, por tanto, también otro recurso de conexión S7 en la estación.



①
② Interlocutor
③ Subred, p. ej. Industrial Ethernet

  – Conexión S7 bilateral

Se ocupa un recurso de conexión S7 en la CPU por cada conexión S7 y, por tanto, también otro recurso de conexión S7 en la estación.



① Un recurso de conexión S7 en el CP y, por tanto, otro en la estación
② Interlocutor
③ Subred, p. ej. Industrial Ethernet

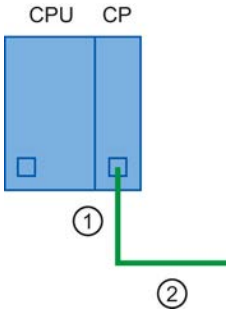
Nota

En "Recursos de conexión" en la navegación local, para determinados tipos de CPU S7-300 puede consultarse el número de recursos de conexión S7 reservados para la CPU seleccionada, y con S7-1500 para la CPU seleccionada y los módulos de comunicación conectados. Aquí se pueden reservar recursos de conexión S7 para la comunicación básica S7 y OP/PG.

Comunicaciones para servicios de comunicación abiertos (interfaz SEND/RECEIVE)

La comunicación a través de la interfaz SEND/RECEIVE se realiza exclusivamente con CPs. Para ello, en el CP se ocupa un recurso de conexión específico de protocolo para el punto final por cada conexión (es decir, conexión FDL, ISO-Transport, ISO-on-TCP, UDP y TCP).

En la CPU no se necesitan recursos de conexión para la conexión.



① 1 Recurso de conexión ocupado en el CP
② Subred, p. ej. Industrial Ethernet

Excepción: En la CPU S7-400, las conexiones con la propiedad "SPEED SEND/RECEIVE" requieren un recurso de conexión S7 adicional para cada conexión "SPEED SEND/RECEIVE" en la CPU.

Ocupación de los recursos de conexión

Activación de la conexión

A menudo existe la posibilidad de elegir entre una activación configurada o una programada. La activación programada facilita la habilitación de recursos de conexión tras la transferencia de datos. La activación configurada sirve de apoyo para administrar los recursos de conexión en STEP 7.

Tabla 8-1 Activación del servicio de comunicación

Servicio de comunicación	Activación automática	Activación programada	Activación configurada
Comunicación PG	X	-	-
Comunicación HMI	X	-	X
Comunicación S7	-	X	X
Comunicación abierta vía TCP/IP	-	X	X
Comunicación abierta vía ISO on TCP	-	X	X
Comunicación abierta vía UDP	-	X	X
Comunicación abierta vía ISO	-	X	X
Comunicación abierta vía correo electrónico	-	X	-
Comunicación abierta vía FTP	-	X	-
Comunicación abierta vía FDL	-	X	X
Routing S7	-	-	X

Ocupación de los recursos de conexión en función del dispositivo de conexión.

Dependiendo de cómo se realice la conexión, la ocupación de los recursos se efectúa de manera diferente:

- Activación configurada
  En las conexiones configuradas, el recurso de conexión se ocupa en cuanto la configuración hardware se carga en la CPU. Para servidores OPC los recursos se ocupan conforme se van necesitando.
- Activación programada
  Al llamar las instrucciones correspondientes para establecer la conexión (vía TSEND_C o T_CON), se ocupa un recurso de conexión.
- Activación automática
  En cuanto la PG o el dispositivo HMI se conecta físicamente y en línea con una CPU, se ocupan los recursos de conexión.

Recursos de conexión con activación configurada

Active la conexión en la vista de redes del editor de hardware y de redes dentro del contexto de una CPU.

Una vez transmitidos los datos no se deshace la conexión, es decir, los recursos de conexión permanecen ocupados permanentemente. Para deshacer la conexión, borre su configuración y cargue en la CPU la configuración modificada.

Las conexiones OPC se establecen y deshacen automáticamente en función de las necesidades.

Si se han **configurado** conexiones, en el sistema de automatización de supervisa que se respete el número máximo posible de recursos.

Recursos de conexión con activación programada

Active la conexión en el editor de programas en el contexto de una CPU parametrizando instrucciones para comunicación, p. ej. TSEND_C.

Una vez establecida la conexión con las instrucciones TCON o TSEND_C, se deshace parametrizando adecuadamente la instrucción TSEND_C o llamando la instrucción TDISCON después de la transferencia de datos. Cuando se deshace la conexión, los recursos de conexión vuelven a estar disponibles en la CPU, el CP o el CM. Si se mantiene la conexión, el tiempo de procesamiento de las instrucciones en una nueva transferencia de datos es menor, puesto que no es necesario volver a establecer la conexión.

Si las conexiones se establecen y deshacen **de forma programada** en el programa de usuario, hay que vigilar que se respeten los límites del sistema de automatización. En caso necesario recibirá un mensaje de error de comunicación en el parámetro de salida "STATUS" de la instrucción TSEND_C o T_CON, y podrá obtener información más detallada en el diagnóstico de la conexión.

Recursos de conexión con activación automática

La conexión se activa automáticamente (p. ej. conexión PG o HMI) en cuanto la interfaz PG-PG se conecta físicamente con una interfaz de la CPU y en el diálogo "Establecer conexión online" se ha realizado la asignación a la interfaz.

Para conexiones activadas **automáticamente** se vigila que en el sistema de automatización se respeten los recursos de conexión reservados.

Consulte también

TCON: Establecer una conexión (Página 5298)

TDISCON: Desconectar comunicación (Página 5305)

TCON: Establecer una conexión (Página 7748)

TDISCON: Desconectar comunicación (Página 7751)

TSEND_C: Enviar datos a través de Ethernet (Página 5263)

TSEND_C: Enviar datos a través de Ethernet (Página 5267)

Visualización de los recursos de conexión

Recursos de conexión en la ventana de inspección

Encontrará un resumen de los recursos de conexión reservados y disponibles de un sistema de automatización S7-1500 en las propiedades de la CPU, dentro de la ventana de inspección del editor de hardware y de redes.

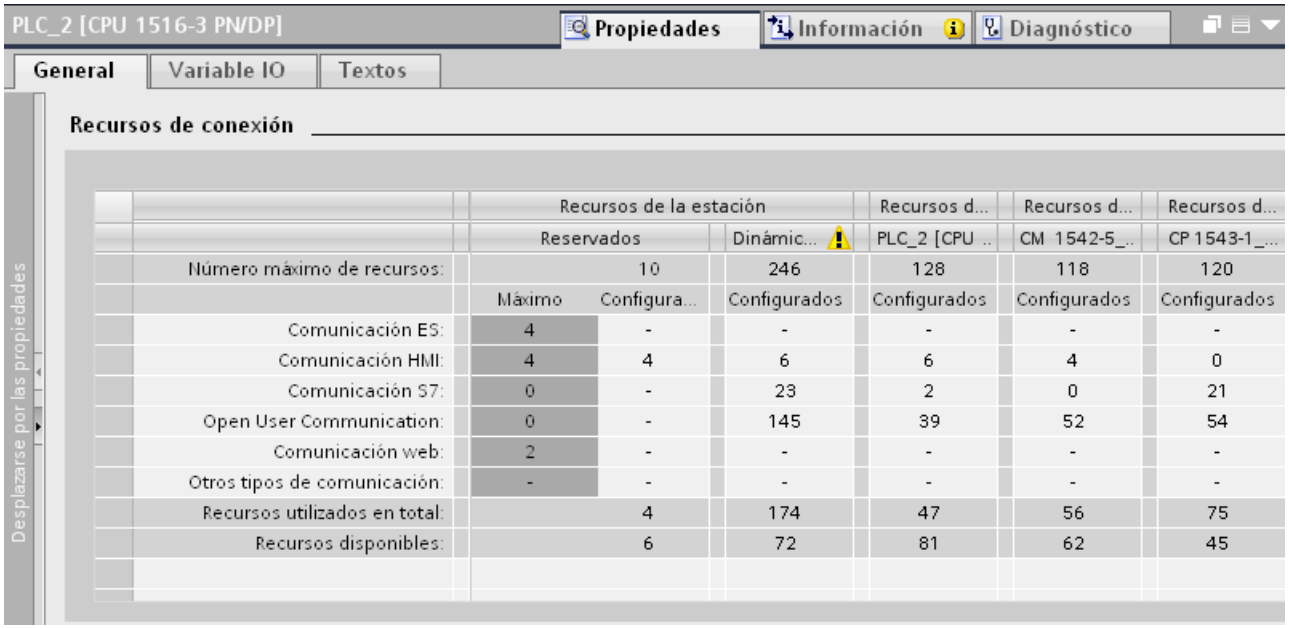
Al visualizar los recursos de conexión se distingue si se trata de la vista offline o de la vista online.

Visualización de los recursos de conexión en la vista offline

En la vista offline se muestran los recursos de conexión reservados y disponibles de las conexiones configuradas.

El número máximo de recursos de conexión disponibles en el sistema de automatización depende de la CPU empleada. Una vez que se alcanza el límite del sistema específico de la CPU, no importa que la CPU, los CPs y los CMs dispongan de otros recursos de conexión. Para ese dispositivo se han agotado los recursos de conexión.

En la figura siguiente, la estación dispone de 246 recursos de conexión dinámicos y 10 reservados.



	Recursos de la estación			Recursos d...	Recursos d...	Recursos d...
	Reservados		Dinámic...	PLC_2 [CPU ...	CM 1542-5_...	CP 1543-1_...
Número máximo de recursos:		10	246	128	118	120
	Máximo	Configura...	Configurados	Configurados	Configurados	Configurados
Comunicación ES:	4	-	-	-	-	-
Comunicación HMI:	4	4	6	6	4	0
Comunicación S7:	0	-	23	2	0	21
Open User Communication:	0	-	145	39	52	54
Comunicación web:	2	-	-	-	-	-
Otros tipos de comunicación:	-	-	-	-	-	-
Recursos utilizados en total:		4	174	47	56	75
Recursos disponibles:		6	72	81	62	45

Estos 10 últimos están reservados para:

- 4 para comunicación PG, p. ej. para funciones de test y diagnóstico o para la carga en la CPU
- 4 para comunicación HMI, para ser ocupados por las primeras conexiones HMI que se configuren
- 2 para comunicación vía servidor web, que en caso necesario se ocupan online

Como máximo hay 246 recursos de conexión dinámicos, es decir, disponibles para diferentes servicios de comunicación en el sistema de automatización. De ellos, 174 recursos están configurados para distintos servicios de comunicación y módulos. Quedan disponibles 72 recursos para el sistema de automatización.

Las columnas de los recursos específicos de módulos para CPU, CPs y CMs incluyen:

- el número máximo de recursos de conexión disponibles para CPU, CP o CM
- el número de los recursos necesarios para las distintas conexiones de comunicación configuradas
- el número de todos los recursos de conexión configurados y utilizados
- el número de recursos de conexión que quedan disponibles

La CPU dispone como máximo de 128 recursos de conexión. Para la CPU se han proyectado seis recursos para conexiones HMI, dos para conexiones S7 y 39 para conexiones de comunicación abierta, que ocupan en total 47 recursos en la CPU. Quedan disponibles en la CPU otros 81 recursos. Para el CP 1543-1, de los 118 recursos disponibles hay ocupados 56, quedando otros 62 recursos disponibles. Para el CP 1542-5, de los 120 recursos disponibles hay ocupados 75, quedando otros 45 recursos disponibles.

En la columna "Recursos de la estación", campo "Recursos dinámicos de la estación", aparece una señal de advertencia cuando el total de los recursos máximos disponibles de CPU, CP y CM (= 366 recursos de conexión) sobrepasa el límite de la estación (256).

Cuando el número de recursos disponibles es 0, es decir, cuando todos los recursos ya están reservados para conexiones, también aparece una señal de advertencia en la columna "Recursos dinámicos de la estación", línea "Recursos disponibles". En este caso no pueden crearse más conexiones.

Nota

Superación de los recursos de conexión disponibles

Si se sobrepasa el número de recursos de conexión específicos de una estación, STEP 7 lo avisa con un símbolo de error. En ese caso hay que emplear una CPU que permita un mayor número de recursos de conexión disponibles, o reducir las conexiones de comunicación.

Referencia

Aquí encontrará un ejemplo detallado para calcular los recursos de conexión: Ejemplo de cálculo de recursos de conexión (Página 603)

Visualización de los recursos de conexión en la vista offline

Si en la vista de redes del editor de hardware y de redes ha seleccionado una CPU conectada online, en la ficha "Información de la conexión" del campo "Diagnóstico" de la ventana de

inspección se muestra el estado actual de recursos de conexión ocupados y libres para el sistema de automatización:

PLC_2 [CPU 1516-3 PN/DP] | Propiedades | Información | Diagnóstico

Información de dispositivos | Información de la conexión | Visor de avisos

Recursos de conexión

	Recursos de la estación					Recursos del módulo	
		Reservados		Dinámicos		PLC_2 [CPU 1516-3 PN/DP]	
Número máximo de recursos:		10	10	194	194	86	86
	Máximo	Configurados	Utilizados	Configurados	Utilizados	Configurados	Utilizados
Comunicación PG:	4	-	4	-	0	-	0
Comunicación HMI:	4	4	4	4	4	8	8
Comunicación S7:	0	0	0	72	68	34	34
Open User Communication:	0	0	0	118	118	45	45
Comunicación web:	2	-	2	-	0	-	0
Otros tipos de comunicación:	-	-	0	-	0	-	0
Recursos utilizados en total:		4	10	194	190	82	82
Recursos disponibles:		6	0	0	4	4	4

Además de los recursos de conexión reservados y dinámicos, la vista online de la tabla "Recursos de conexión" contiene también columnas con los recursos de las conexiones programadas y activadas automáticamente que están ocupados en la estación. Por tanto, en la vista online se muestran **todos** los recursos ocupados del sistema de automatización, con independencia de cómo se haya activado la conexión. En la línea "Otra comunicación" se muestran los recursos ocupados para comunicación con dispositivos de terceros y comunicación vía routing de registros.

La tabla se actualiza automáticamente. Si existen incoherencias entre la configuración de hardware y la existente online, se emite el correspondiente mensaje.

Consulte también

Recursos para funciones de routing (Página 608)

Ejemplo de cálculo de recursos de conexión

Introducción

Este ejemplo de la tabla que contiene la visualización de los recursos muestra la composición de las distintas entradas de la tabla.

Requisitos

- 2 CPUs 1516-3 PN/DP con un CP 1543-1 y un CM 1542-1 cada una, para ampliar la capacidad como interlocutor
- 1 display TFT MP 377 12" Touch con WinAC MP 377 como interlocutor para conexiones HMI
- Los dispositivos están conectados en red entre sí.

Creación de las conexiones

Después de crear e interconectar los interlocutores (CPUs, CPs, CMs, MP 377) en la vista de redes del editor de hardware y de redes, proceda del siguiente modo:

1. En la vista de redes cambie al modo de conexión.
2. Cree las conexiones siguientes:
   - Conexiones S7: 2 vía CPU, 21 vía CM
   - Conexiones TCP: 2 vía CPU, 4 vía CM
   - Conexión ISO: 6 vía CP
   - ISO on TCP: 3 vía CPU, 4 vía CM
   - Conexiones UDP: 4 vía CPU, 106 vía CP
   - Conexiones HMI: 1 vía CPU, 2 vía CP

Resultado

En la visualización de los recursos de conexión se verán los recursos disponibles y los recursos que son necesarios para las conexiones creadas.

En la figura siguiente se destacan los recursos disponibles para dispositivos y módulos:

	Recursos de la estación			Recursos d...	Recursos d...	Recursos d...
	Reservados		Dinámic... ⚠	PLC_2 [CPU ...	CP 1543-1_...	CM 1542-5_...
Número máximo de recursos:		10	246	128	118	118
	Máximo	Configura...	Configurados	Configurados	Configurados	Configurados
Comunicación ES:	4	-	-	-	-	-
Comunicación HMI:	4	4	5	3	6	0
Comunicación S7:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Comunicación web:	2	-	-	-	-	-
Otros tipos de comunicación:	-	-	-	-	-	-
Recursos utilizados en total:		4	157	14	118	29
Recursos disponibles:		6	89	114	0	89

① Dependiendo de qué CPU se utilice, esta estación dispone de un máximo de 256 recursos. 10 recursos están reservados para determinados tipos de conexión y 246 están disponibles dinámicamente.

② La CPU proporciona 128 recursos de conexión, el CP y el CM 118 recursos adicionales cada uno.

③ Conjuntamente, la CPU, el CP y el CM proporcionan 128 + 118 + 118 = 364 recursos. Como a través de los dispositivos y módulos se proporcionan más recursos de los que puede manejar como máximo la estación (256), se muestra un símbolo de advertencia.

En la figura siguiente se destacan los recursos consumidos por las conexiones creadas:

	Recursos de la estación			Recursos d...	Recursos d...	Recursos d...
	Reservados		Dinámic...	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_...
Número máximo de recursos:		10	246	128	118	118
	Máximo	Configura...	Configurados	Configurados	Configurados	Configurados
Comunicación ES:	4	-	-	-	-	-
Comunicación HMI:	4	4	5	3	6	0
Comunicación S7:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Comunicación web:	2	-	-	-	-	-
Otros tipos de comunicación:	-	-	-	-	-	-
Recursos utilizados en total:		4	157	14	118	29
Recursos disponibles:		6	89	114	0	89

① Se trata de los recursos ocupados vía las conexiones HMI. En este ejemplo una conexión HMI ocupa 3 recursos. Para toda la estación se necesitan 3 + 6 = 9 recursos de conexión para la comunicación HMI.

② Hay 4 recursos reservados para la comunicación HMI.

③ Los 9 recursos necesarios para la comunicación HMI se distribuyen de la manera siguiente: Para los 4 primeros recursos se emplean los que ya estaban reservados para la comunicación HMI. Los otros 5 recursos necesarios se toman de los recursos dinámicos.

④ Se trata de los recursos ocupados vía las conexiones S7. Una conexión S7 consume 1 recurso de conexión. Para toda la estación se necesitan 2 + 21 = 23 recursos de conexión.

⑤ No hay recursos reservados para la comunicación S7. Los 23 recursos necesarios proceden de los recursos disponibles dinámicamente.

⑥ Se trata de los recursos de Open User Communication (para TCP, FDL, ISO, ISO on TCP y UDP). Para toda la estación se necesitan 9 + 112 + 8 = 129 recursos de conexión.

⑦ No hay recursos reservados para la Open User Communication. Los 129 recursos necesarios proceden de los recursos disponibles dinámicamente.

Cálculo de los recursos de conexión

A partir de los recursos disponibles y utilizados de cada dispositivo o módulo se calculan los recursos que quedan libres y los utilizados en total:

	Recursos de la estación			Recursos d...	Recursos d...	Recursos d...
	Reservados		Dinámic...	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_..
Número máximo de recursos:		10	246	128	118	118
	Máximo	Configura...	Configurados	Configurados	Configurados	Configurados
Comunicación ES:	4	-	-	-	-	-
Comunicación HMI:	4	4	5	3	6	0
Comunicación S7:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Comunicación web:	2	-	-	-	-	-
Otros tipos de comunicación:	-	-	-	-	-	-
Recursos utilizados en total:		4	157	14	118	29
Recursos disponibles:		6	89	114	0	89

① Aquí se muestran todos los recursos necesarios para cada tipo de comunicación de la CPU. Se necesitan 3 recursos para la comunicación HMI, 2 para la comunicación S7 y 9 para los tipos de conexión de Open User Communication.

② Debajo de la lista de recursos necesarios para los distintos tipos de comunicación se encuentra el número total de recursos necesarios. Para la CPU resultan 3 + 2 + 9 = 14 recursos necesarios.

③ El CP 1543-1 tiene 118 recursos de conexión disponibles. Si están ocupados, no pueden crearse otras conexiones para el CP.

④ Para el CP 1543-1 se han ocupado 118 recursos vía las 2 conexiones HMI (3 recursos cada una = 6 recursos) y 112 vía las conexiones de comunicación abierta.

⑤ El CP 1543-1 no tiene recursos libres. No pueden crearse más conexiones para el CP 1543-1.

La ilustración también muestra los recursos libres y el total de recursos utilizados para el conjunto de la estación:

	Recursos de la estación			Recursos d...	Recursos d...	Recursos d...
	Reservados		Dinámic...	PLC_2 [CPU ..	CP 1543-1_...	CM 1542-5_..
Número máximo de recursos:		10	246	128	118	118
	Máximo	Configura...	Configurados	Configurados	Configurados	Configurados
Comunicación ES:	4	-	-	-	-	-
Comunicación HMI:	4	4	5	3	6	0
Comunicación S7:	0	-	23	2	0	21
Open User Communication:	0	-	129	9	112	8
Comunicación web:	2	-	-	-	-	-
Otros tipos de comunicación:	-	-	-	-	-	-
Recursos utilizados en total:		4	157	14	118	29
Recursos disponibles:		6	89	114	0	89

① Recursos empleados en toda la estación para cada tipo de comunicación, recursos reservados (configurados) y recursos dinámicos. Los valores resultan de la suma de recursos utilizados en los dispositivos y módulos (CPU, CP, CM).

② Recursos empleados en toda la estación para todos los tipos de comunicación, recursos reservados (configurados) y recursos dinámicos.

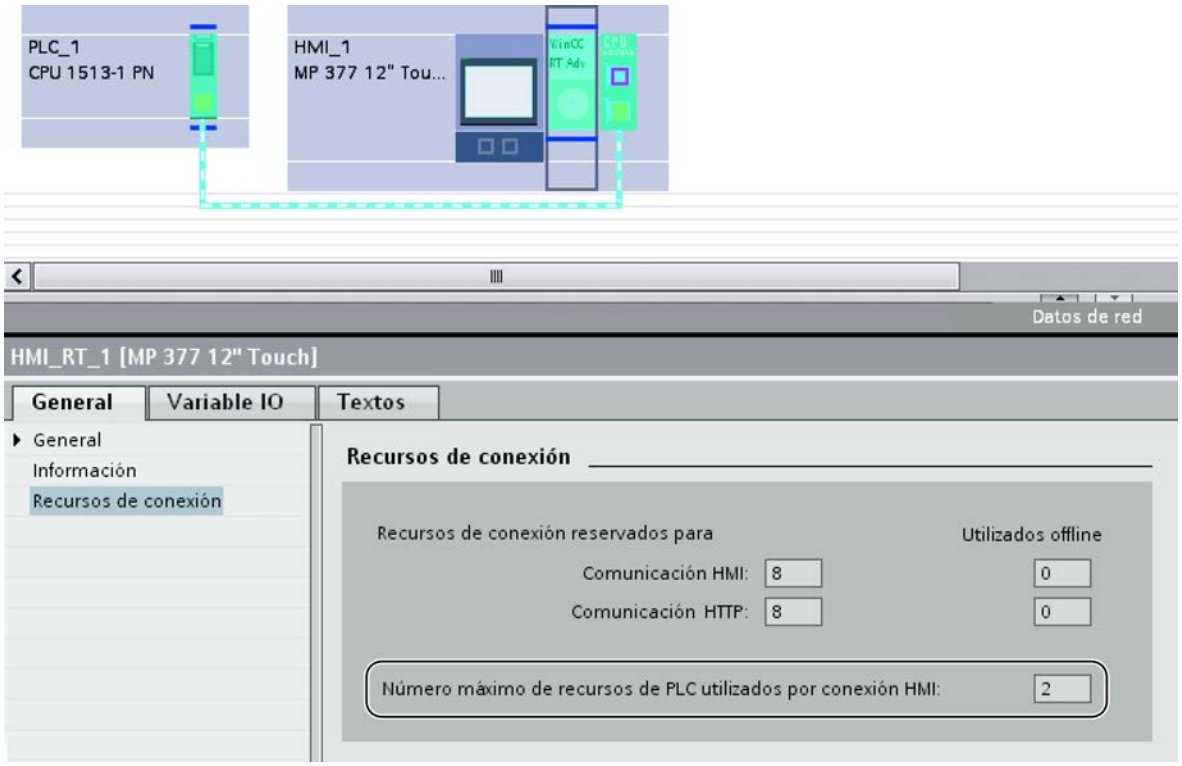
③ Número total de recursos reservados aún disponibles y de recursos dinámicos. Los valores se obtienen de restar al número máximo de recursos configurables el número de recursos utilizados.

Recursos para la comunicación HMI

Recursos de conexión para la comunicación HMI

En la comunicación HMI la ocupación de recursos de conexión en la CPU depende del dispositivo HMI empleado y de las aplicaciones en las que se utilice dicho dispositivo. Por ejemplo, un Basic Panel requiere hasta 2 recursos de conexión en la CPU para una conexión HMI, y un Comfort Panel hasta 3 recursos.

Puede consultarse información sobre la disponibilidad y ocupación de recursos para conexiones HMI en la vista offline del dispositivo HMI, en el campo "Recursos de conexión", dentro de Propiedades, en la ventana de inspección. También allí puede verse cuántos recursos se necesitan para una conexión HMI:



Si se sobrepasa el número máximo de recursos de conexión disponibles para un dispositivo HMI, se emite el correspondiente mensaje.

Información adicional

Para saber cuántos recursos precisa un dispositivo HMI, consulte la documentación del mismo.

Recursos para funciones de routing

Conexiones S7 con el exterior de la red (routing S7)

Para la transferencia de datos al exterior de la red ("routing S7") se establece una conexión S7 entre dos CPUs que abarque varias subredes. Las subredes están conectadas entre sí mediante transiciones de red denominadas routers. El router es el punto de paso de una conexión S7 y puede establecer conexiones S7. En S7-1500 pueden ser routers las CPUs, los CMs y los CPs.

En los puntos finales de la comunicación (CPUs) se toman en consideración los recursos de conexión para routing S7 y se muestran en la tabla "Recursos de conexión" en la comunicación S7. Además, en las transiciones entre redes debe haber disponibles recursos de conexión para el routing.

ATENCIÓN
Recursos limitados para routing S7 En la CPU 1516 se establecen como máximo 16 conexiones S7 enrutadas. Si configura más conexiones S7 enrutadas, las conexiones sobrantes no se establecen.

Routing de registros

El routing de registros también permite transferir datos al exterior de una red, desde una estación de ingeniería conectada a PROFINET y vía PROFIBUS, hasta diversos dispositivos de campo.

Para el routing de registros, en las transiciones entre redes se ocupan recursos de conexión. En el punto final de la comunicación (CPU) los recursos para el routing de registros se muestran en la vista online de la tabla "Recursos de conexión", en la línea "Otra comunicación".

Nota

Recursos de conexión en routing S7 y en routing de registros

Tanto en el routing S7 como en el routing de registros, en los routers se ocupan los correspondientes recursos de conexión. No es posible configurar y supervisar estos recursos de conexión en STEP 7. El usuario debe comprobar personalmente que se respetan los límites.

Conexiones a través de CPs

Volumen de datos y recursos disponibles

El número de conexiones soportadas por el correspondiente CP se puede consultar en el manual de producto adjunto con cada CP. Añadiendo más CPs se aumenta el número de conexiones por dispositivo.

Si en un dispositivo hay montados varios CPs, al sobrepasar este límite se cambia automáticamente al siguiente CP. En caso necesario, asigne selectivamente las conexiones a través de la elección de vía.

CP Ethernet

El CP Ethernet puede transferir a través de una conexión las siguientes cantidades de datos por petición:

	ISO-Transport	ISO-on-TCP	TCP	UDP
Enviar	8192 bytes	8192 bytes	8192 bytes	2048 bytes
Recibir	8192 bytes	8192 bytes	8192 bytes	2048 bytes

Nota

Los CPs actuales soportan la transferencia de datos > 240 bytes.

Los CPs con una versión más antigua soportan la transferencia de datos con una longitud de hasta 240 bytes.

Tenga en cuenta las indicaciones del manual de producto del CP Ethernet.

Tareas del CP Ethernet en el modo online

El CP Ethernet adopta las siguientes tareas durante la transferencia de datos a través de una conexión:

- Recibir
  Recepción de datos de Ethernet y transmisión al área de datos de usuario de la CPU.
- Enviar
  Aceptación de datos del área de datos de usuario de la CPU y transmisión de los datos a través de Ethernet.

El establecimiento de la conexión se realiza automáticamente en cuanto se puede acceder al interlocutor.

En una conexión UDP/FDL libre se añade:

- Al recibir
  Introducción del remitente que ha enviado el mensaje en la cabecera de la petición.
- Al enviar
  Evaluación de la cabecera de la petición y direccionamiento del interlocutor.

CP PROFIBUS

El CP PROFIBUS puede transferir a través de una conexión las siguientes cantidades de datos por petición:

Tipo de conexión FDL	Especificada	Sin especificar - acceso libre a nivel 2	Broadcast	Multicast
Enviar	240 bytes	236 bytes [1]	236 bytes [1]	236 bytes [1]
Recibir	240 bytes	236 bytes [1]	236 bytes [1]	236 bytes [1]

1) La cabecera de la petición ocupa 4 bytes adicionales además de los datos de usuario nombrados

Tareas del CP PROFIBUS en el modo online

El CP PROFIBUS adopta las siguientes tareas para ejecutar la transferencia de datos a través de una conexión FDL:

- En conexiones especificadas
  - Recibir
    Recepción de datos de PROFIBUS y transmisión al área de datos de usuario de la CPU.
  - Enviar
    Aceptación de datos del área de datos de usuario de la CPU y transmisión de los datos a través de PROFIBUS.
- En conexiones sin especificar adicionalmente
  - Al recibir
    Introducción del remitente y del servicio FDL en la cabecera de la petición.
  - Al enviar
    Evaluación de la cabecera de la petición y direccionamiento del interlocutor; ejecución del servicio FDL seleccionado.

Conexiones con estaciones PC

Puntos finales de la conexión de una estación PC

En el caso de estaciones PC, configure las conexiones con uno de los siguientes componentes como punto final:

- Servidor OPC como prestador de servicios de comunicación central
  Según el CP disponible, pueden utilizarse los siguientes tipos de conexión:
  - Conexión S7
  - Conexión ISO
  - Conexión ISO-on-TCP
  - Conexión TCP
  - Conexión FDL
- Aplicación estándar
  Con el tipo de conexión S7 es posible realizar la comunicación a través de Ind. Ethernet o PROFIBUS según el CP disponible.

Particularidades de la configuración de conexiones con servidor OPC

En conexiones con OPC como punto final, generalmente se pueden configurar más conexiones de las que el CP utilizado en la estación PC puede operar en un determinado momento.

Esto está relacionado con el hecho de que en las propiedades de conexión se diferencia entre "mantener de forma permanente" y "establecer a petición".

Por consiguiente, el número de conexiones con la propiedad "mantener de forma permanente" no puede sobrepasar el número máximo de recursos de conexión del CP. Si fuese posible establecer más conexiones a petición, deben mantenerse libres los correspondientes recursos de conexión del CP. Por ello, con la propiedad "mantener de forma permanente" configure un menor número de conexiones de las admitidas por el número máximo de recursos de conexión del CP.

Si es necesario, aparece una advertencia al compilar la configuración de conexiones.

Información adicional

Comunicación industrial con la PG/el PC banda 1 en la ID del artículo: 42783968 (http://support.automation.siemens.com/WW/view/es/42783968)

Comunicación industrial con la PG/el PC banda 2 en la ID del artículo: 42783660 (http://support.automation.siemens.com/WW/view/es/42783660)

Puesta en marcha de estaciones PC, manual en la ID del artículo: 13542666 (http://support.automation.siemens.com/WW/view/es/13542666)

Conexiones no especificadas

Conexiones no especificadas en Industrial Ethernet

Las conexiones con un dispositivo aún desconocido (p. ej., un dispositivo de diagnóstico) se configuran como conexiones "sin especificar". Una conexión no especificada se puede utilizar de diferentes maneras, como se explica a continuación con una conexión ISO-on-TCP a modo de ejemplo; el procedimiento que se debe usar con las conexiones ISO-Transport y con las conexiones TCP es equivalente.

- **Declaración de disponibilidad para la comunicación: establecimiento de conexión pasivo**
  El establecimiento de la conexión debe ajustarse como "pasivo".
  En tal caso, para la configuración de la dirección en una conexión ISO-on-TCP rige lo siguiente:
  la dirección IP remota y el TSAP remoto están vacíos, es decir, son irrelevantes para el CP. Por lo tanto, al establecer la conexión se aceptan todos los interlocutores (interlocutor = nombre de la conexión que direcciona el CP con dirección IP correcta y TSAP).
  También existe la posibilidad de establecer una especificación parcial, es decir, se autoriza la comunicación con un interlocutor cualquiera que coincida en el TSAP especificado.
- **Conexión con una estación concreta de un proyecto cualquiera**
  En tal caso, para la configuración de la dirección en una conexión ISO-on-TCP rige lo siguiente:
  Es posible indicar la dirección IP remota y el puerto para una estación de destino cualquiera. La estación de destino puede encontrarse dentro o fuera del proyecto STEP 7 actual.
  Utilice esta variante si no ha creado la estación interlocutora en el proyecto actual.
- **Conexión sin especificación del puerto**
  Las conexiones TCP no se especifican en los casos siguientes:
  - El puerto local no se indica (establecimiento activo de la conexión).
  - El puerto remoto no se indica (establecimiento no activo de la conexión).
- **Direccionamiento IP a través de DHCP**
  Si selecciona la opción de direccionamiento IP a través de DHCP, primeramente no se puede crear ninguna conexión completamente especificada en el proyecto, ya que se desconoce la dirección IP local. Por consiguiente, se debe seleccionar el tipo de conexión "sin especificar" sin establecimiento de conexión activo.

La tabla siguiente resume las distintas posibilidades.

Tabla 8-2 Configuración del parámetro de dirección "remoto"

Significado para el establecimiento de la conexión	Dirección IP / dirección MAC (remota)	TSAP / puerto (remoto)	Establecimiento activo de la conexión
desde un interlocutor cualquiera	vacía	vacía	no
desde un interlocutor cualquiera a través de un TSAP determinado	vacía	especificado	no

Significado para el establecimiento de la conexión	Dirección IP / dirección MAC (remota)	TSAP / puerto (remoto)	Establecimiento activo de la conexión
hacia o desde un interlocutor determinado	especificada	especificada	sí El puerto local puede quedar sin especificar (pero no necesariamente).
		No especificada	no

La conexión UDP libre representa otra variante. En este tipo de conexión, la dirección del interlocutor queda sin especificar durante la configuración. Los dispositivos interlocutores se determinan mediante la indicación de direcciones en la petición de comunicación del programa de usuario.

Conexiones no especificadas en PROFIBUS

Una conexión FDL no especificada con acceso libre Layer 2 permite el direccionamiento del interlocutor controlado por el programa y la comunicación entre dos dispositivos de PROFIBUS con las siguientes propiedades:

El dispositivo local ha sido fijado mediante configuración. El programa de usuario registra el dispositivo remoto en el encabezado de la petición del búfer de peticiones al llamar la instrucción AG_SEND. De esta forma es posible acceder a todos los dispositivos de PROFIBUS (direcciones PROFIBUS de 0 a 126).

Información adicional

Seguidamente encontrará más información al respecto en los capítulos dedicados a las conexiones.

Conexión UDP libre (Página 674)

Conexión FDL no especificada (acceso libre Layer 2) (Página 683)

Conexiones programadas a través de CP

Campo de aplicación

En determinados campos de aplicación no conviene crear las conexiones ni la configuración IP del CP con la configuración, sino con una aplicación específica controlada por programa.

Existe la posibilidad de configurar libremente las conexiones mediante un programa a través de los CPs Industrial Ethernet. Deben tenerse en cuenta las características funcionales del tipo de CP utilizado.

Un caso de aplicación típico es p. ej. el de los fabricantes de máquinas en serie. Para ofrecer a sus clientes una interfaz de usuario lo más cómoda posible, estos fabricantes adaptan los servicios de comunicación a las correspondientes peticiones de mando. El usuario final no necesita conocimientos sobre la configuración de conexiones.

Para estas aplicaciones se dispone de la instrucción IP_CONFIG, que permite la transferencia flexible de bloques de datos con datos de configuración a un CP Ethernet.

Interacción de programación y configuración

Las conexiones se configuran en general o a través del programa de usuario durante el tiempo de ejecución del dispositivo S7. Dentro de un CP no es posible mezclar ambas variantes.

Principio

La instrucción IP_CONFIG del programa de usuario transfiere al CP los datos de configuración para las conexiones.

El DB de configuración se puede cargar así en cualquier momento en el CP. Las conexiones y datos de configuración previos (dirección IP, máscara de subred, router predeterminado, servidor horario NTP y otros parámetros) se sobrescriben.

El CP Ethernet reconoce mediante los datos de configuración que las conexiones deben establecerse a través del programa de usuario.

Recursos disponibles

La instrucción IP_CONFIG admite un máximo de 64 conexiones. Sin embargo, es decisivo el número máximo de conexiones que soporta el tipo de CP utilizado.

Particularidades/restricciones

- Comprobación de coherencia sólo con una conexión configurada
  La configuración de conexiones está enlazada con comprobaciones de coherencia que no son posibles durante la configuración programada o lo son de forma limitada.
- Configuración de conexiones necesaria en el interlocutor
  En la configuración de conexiones especificadas se crea implícitamente la conexión para el interlocutor. Esto no es posible en el caso de una configuración programada. En este caso, tienen que configurarse para los interlocutores las correspondientes conexiones coherentes.
- Configuración de la protección de acceso IP
  Mediante la protección de acceso IP existe la posibilidad de limitar la comunicación a través del CP del dispositivo S7 local a interlocutores con direcciones IP muy concretas. Esta parametrización también es válida para conexiones programadas.
- Se soporta DHCP/DNS
  En la configuración programada, el direccionamiento IP también es posible a través de DHCP/DNS.

Consulte también

Sinopsis (Página 8148)

Conexiones con Open User Communication vía CPU y CP

Open User Communication vía Industrial Ethernet: conexiones programadas

La Open User Communication ofrece servicios para Industrial Ethernet que se pueden utilizar en el programa de usuario mediante la interfaz de instrucciones. Las conexiones se establecen por medio de la interfaz de instrucciones en el programa de usuario; no es necesario configurar las conexiones.

Servicios a través de la interfaz de la CPU

Con S7-1500, para el intercambio de datos a través de la interfaz Ethernet integrada de una CPU se dispone de los servicios siguientes de Open User Communication:

- TCP según RFC 793
- ISO on TCP según RFC 1006
- UDP según RFC 768 (broadcast o singlecast; no se soporta multicast)

Servicios a través de la interfaz del CP

Con S7-300/400/1500, para el intercambio de datos a través de la interfaz Ethernet de un CP se dispone de los servicios siguientes de Open User Communication:

- TCP según RFC 793
- ISO según RFC 905
- ISO on TCP según RFC 1006
- UDP según RFC 768 (broadcast o singlecast; no se soporta multicast)

Consulte también

Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)

Crear y editar conexiones

Crear una conexión: sinopsis

Alternativas para crear una conexión

Existen las posibilidades siguientes para crear una conexión en la vista de redes:

- Configuración gráfica de las conexiones
  Trace una línea de conexión con el interlocutor o asigne una conexión nueva al interlocutor seleccionado previamente. Es una manera sencilla de crear conexiones individuales en la vista de redes.
  Las interconexiones que faltan se agregan automáticamente o a través de cuadros de diálogo.
- Configuración de conexiones a través de cuadros de diálogo
  Seleccionando el interlocutor en una lista. Si se desea configurar un número mayor de conexiones, este procedimiento es más adecuado que el procedimiento gráfico.

Los diferentes pasos que deben seguirse se explican en los capítulos siguientes.

Asignar la vía de conexión

Según el caso de aplicación hay que distinguir entre los siguientes procedimientos:

- Selección del punto final de la conexión en los dispositivos fuente y de destino
  El software determina automáticamente la vía de conexión adecuada para el dispositivo de destino. Esta conexión se denomina entonces "vía de conexión estándar". Para el enrutamiento, el software considera los recursos de conexión disponibles.
  El punto final de la conexión puede ser una CPU o una aplicación para PC.
  Observación: Esta variante sólo es posible para la configuración gráfica de la conexión.
- Selección de las vías de conexión en los dispositivos fuente y de destino
  Siempre que se prevean varios componentes (CPs) y, en caso dado, también diferentes subredes a fin de distribuir la carga, también es posible configurar la vía de una conexión de forma específica. La vía de conexión se especifica a través del punto final de la conexión y la selección de la interfaz con la subred.

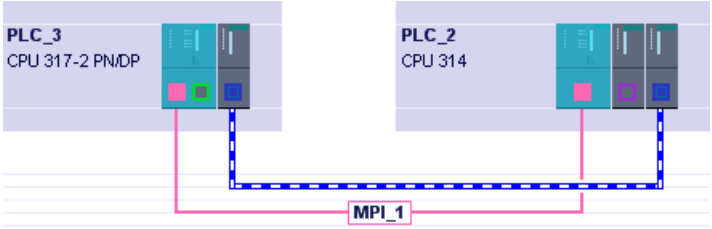
Requisitos y resultado

En la vista de redes se han especificado los dispositivos con CPUs y aplicaciones para PC, así como, dado el caso, los módulos con capacidad de comunicación (CP).

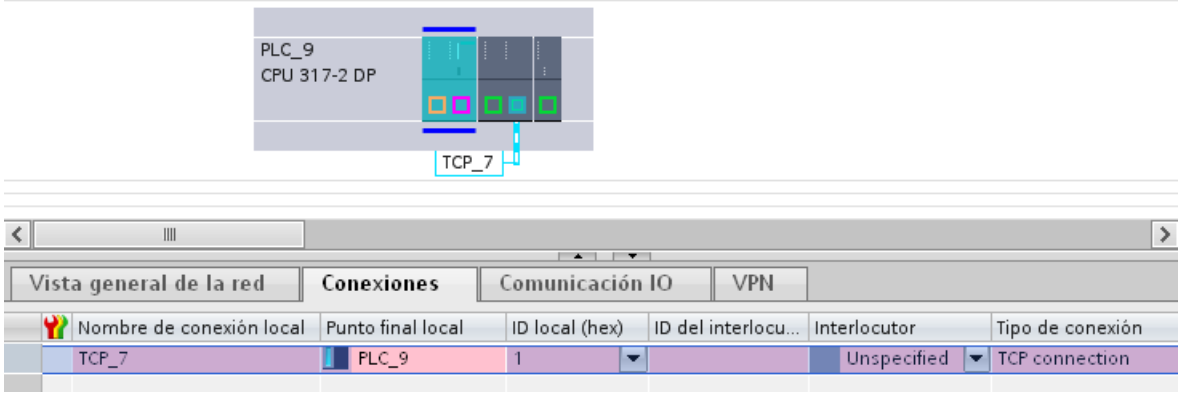
El resultado de la configuración depende en este caso de la configuración actual de los dispositivos y de la interconexión en el proyecto. Por tanto, con los procedimientos descritos en los siguientes capítulos se obtienen los siguientes resultados:

- Conexión especificada
  Cuando ambos interlocutores seleccionados se encuentran interconectados en la misma subred, se define una conexión totalmente especificada de forma gráfica o desde un cuadro de diálogo.
  Esta conexión totalmente especificada se registra automáticamente tanto en la tabla de conexiones del dispositivo local como en la del interlocutor. Para esta conexión se otorgan la ID local y la ID del interlocutor. Estas IDs son necesarias para programar las instrucciones de comunicación (valor para el parámetro del bloque "ID") o de las aplicaciones para el PC.
  Excepciones:
  - Conexiones S7 unilaterales
    En una conexión S7 unilateral, el interlocutor tiene la función de servidor. Por esta razón, para el interlocutor no se realiza ninguna entrada en la tabla de conexiones. Solo para el interlocutor local se otorga una ID de conexión.

  La siguiente representación muestra una conexión especificada:



- Conexión no especificada
  En los siguientes casos se definen conexiones no especificadas de forma específica seleccionando solo el interlocutor local de forma gráfica o desde un cuadro de diálogo:
  - Si determinadas propiedades de conexión requieren una conexión no especificada. Ejemplos: Conexión de correo electrónico, conexión FTP, conexión FDL con acceso libre Layer 2.
  - Cuando uno de ambos interlocutores (aún) no se encuentra disponible para el tipo de conexión seleccionado pero se desea aun así reservar un recurso de conexión.



Influencia de la interconexión: comportamiento cuando falta la conexión en red

La conexión de los interlocutores en red es un requisito para la univocidad y coherencia de los parámetros de conexión asignados. Si uno o ambos interlocutores no están conectados

a la misma subred, es necesario distinguir entre los comportamientos siguientes al configurar la conexión gráficamente:

Nota

Configuración de conexiones a través de cuadros de diálogo: interconexión no automática

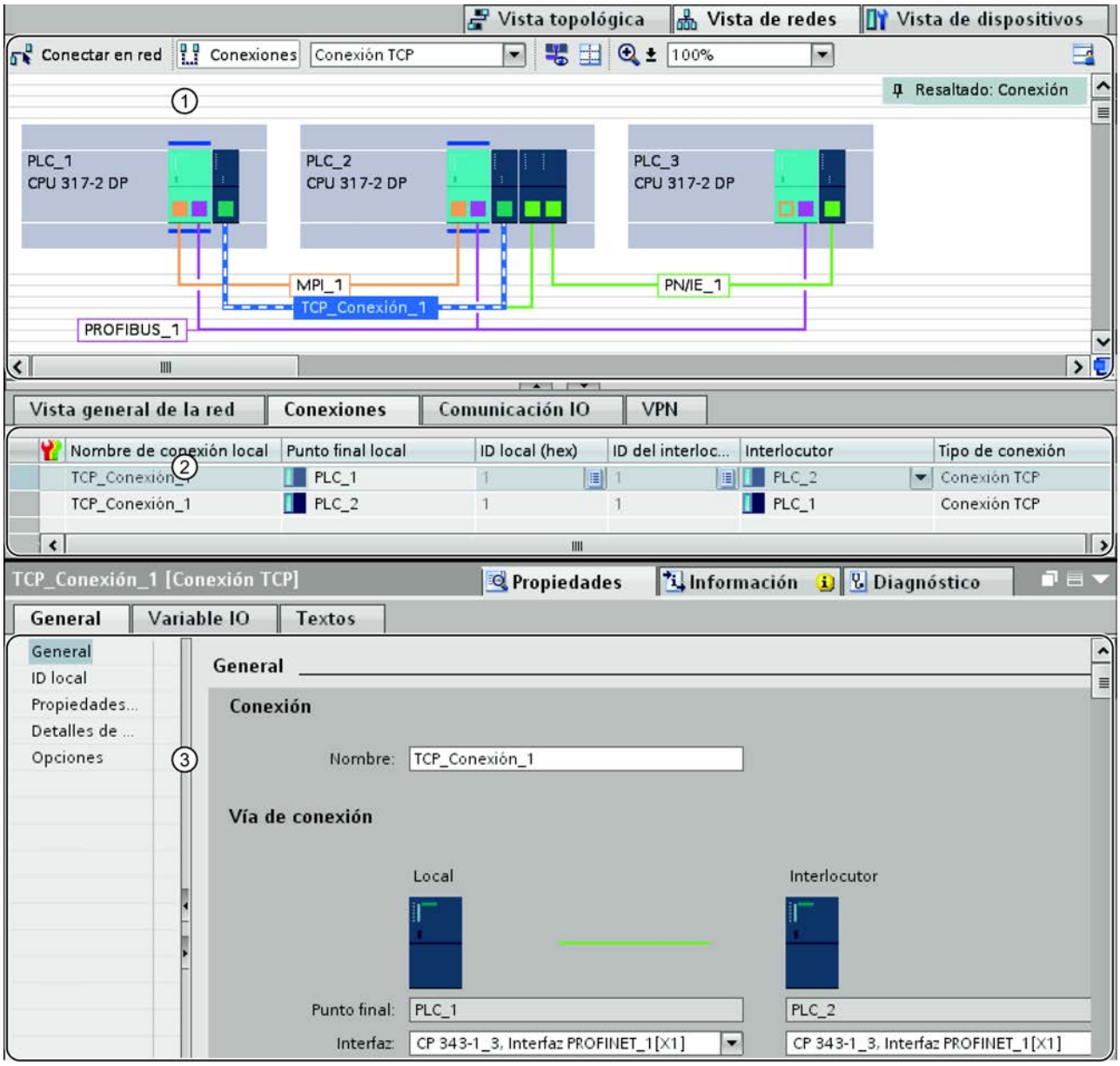
En la configuración de conexiones a través de cuadros de diálogo no se efectúan interconexiones automáticas.

- Caso a: Las interfaces de los interlocutores que se deben conectar no están conectadas en red
  Las interfaces de los interlocutores se conectan a través de una nueva subred. La conexión se puede crear de manera coherente y especificada.
- Caso b: Las interfaces de los interlocutores que se deben conectar primeramente no están conectadas en la misma subred
  Aparece el cuadro de diálogo "Conectar con subred", que ofrece las opciones siguientes:
  - Asignar subred existente
    Seleccione una subred de la lista de subredes disponibles. La conexión se puede crear de manera coherente y especificada.
  - "Conectar con subred nueva"
    Las interfaces de los interlocutores se conectan a través de una nueva subred. La conexión se puede crear de manera coherente y especificada.
  - "No conectar con subred"
    La conexión se crea de manera especificada; la coherencia no está asegurada debido a la falta de interconexión, por lo que la entrada aparece en color rojo en la tabla de conexiones. Las conexiones se representan en la vista de redes en forma de línea discontinua;
- Caso c: Varias interfaces/CPs en el interlocutor: como mínimo una de las interfaces disponibles del interlocutor no está conectada en red
  Según como se cree la conexión en este caso, rige lo siguiente:
  - Si la conexión se establece únicamente seleccionando la CPU o el servidor OPC como punto final, la vía de conexión se establecerá independientemente de la interconexión. Esto puede dar lugar al comportamiento descrito en el caso b, de manera que aparece el diálogo de selección.
  - Si como punto final se selecciona la interfaz del interlocutor que sigue libre y sin conectar, se produce el comportamiento descrito en el caso a.
    Las interfaces de los interlocutores se conectan a través de una nueva subred. La conexión se puede crear de manera coherente y especificada.

Vistas con información sobre las conexiones configuradas

Desde las vistas descritas a continuación se accede a toda la información y las funciones relacionadas con la configuración y verificación de conexiones.

Es posible acceder a la vista a través del árbol del proyecto en "Dispositivos y redes".



① Visualización de conexiones en la vista de redes

② Tabla de conexiones

③ Ficha "Propiedades" de una conexión en la ventana de inspección

Ventajas

La información representada en estas vistas siempre es actual en cuanto a las acciones de usuario inmediatas. Esto significa que:

- Si se ha seleccionado p. ej. una CPU en la vista de redes, la tabla de conexiones muestra todas las conexiones creadas para esta CPU.
  Observación: A través del menú contextual de la tabla de conexiones es posible escoger entre la visualización de todas las conexiones del proyecto y la visualización de las conexiones de un componente seleccionado.
- Si se ha seleccionado una conexión en la tabla de conexiones, rige lo siguiente:
  - Si el modo de comunicación se encuentra conectado, la vía de conexión se representa gráficamente en la vista de redes.
  - La vista Propiedades de la ventana de inspección muestra los parámetros de dicha conexión.

Visualización de conexiones en la vista de redes

La vista de redes muestra los dispositivos con sus componentes, así como las redes y las conexiones.

En referencia a las conexiones, la vista de redes muestra:

- Entre qué dispositivos se han creado conexiones.
- Desde qué dispositivos parten conexiones no especificadas.

La vista de redes ofrece las funciones siguientes:

- Crear conexiones gráficamente;
- Seleccionar las vías de conexión al crearlas gráficamente.

La tabla de conexiones

La tabla de conexiones ofrece las funciones siguientes:

- Relación de las conexiones con un objeto seleccionado en la vista de redes (red, CPU, CP);
- Relación de todas las conexiones del proyecto;
- Selección de una conexión con la consiguiente visualización en la vista de redes, así como visualización de las propiedades de una conexión seleccionada en la ventana de inspección;
- Cambio de interlocutores;
- Posibilidad de entrada de parámetros centrales;
- Visualización de información de estado;
- Acceso a la creación de conexiones nuevas a través de cuadros de diálogo (tiene que estar desactivada la opción "Mostrar todas las conexiones").

Ficha "Propiedades" de una conexión en la ventana de inspección

El cuadro de diálogo de propiedades tiene el significado siguiente:

- Visualización y posibilidad de ajuste de todos los parámetros de conexión;
- Ajuste de la vía de conexión; de forma automática o seleccionable
- Especificación posterior de conexiones no especificadas.

Crear una conexión gráficamente

Configurar conexiones gráficamente

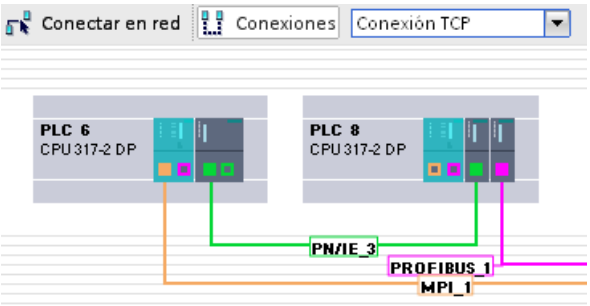
- En el caso más sencillo se seleccionan solo los dispositivos que se desea conectar. En la configuración actual se selecciona una vía de conexión posible y se denomina "vía de conexión estándar". La vía de conexión se selecciona entonces de tal forma que los recursos disponibles sean utilizados de la mejor forma. La definición se basa en el siguiente comportamiento:
  - En la medida de lo posible, los componentes que no están interconectados se interconectan automáticamente para poder crear conexiones totalmente especificadas directamente.
  - Los recursos de un componente se utilizan por completo antes de conmutar a otro componente.
  - Los tipos de red se ven sujetos a las siguientes reglas de preferencia para los tipos de conexión independientes de la red (conexiones S7): Industrial Ethernet antes que PROFIBUS.
- Como alternativa se determina específicamente de forma fija una vía de conexión sobre el punto final de la conexión, así como la interfaz con la subred.

Determinar automáticamente la vía de conexión

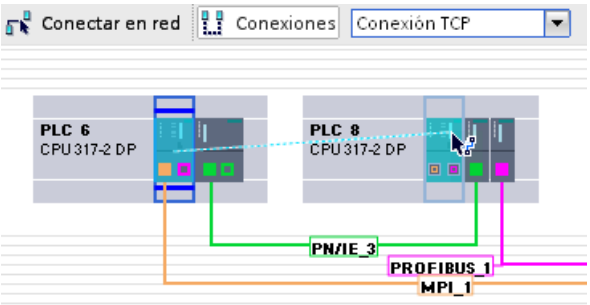
Para crear una conexión gráficamente sobre una vía de conexión por defecto, proceda del siguiente modo:

1. Haga clic en el símbolo "Conexiones" de la barra de herramientas. De esta forma se activa el modo de conexión.

2. Seleccione el tipo de conexión que necesite en la lista desplegable situada al lado.
   A continuación es posible observar el siguiente comportamiento:
   – En la vista de redes están resaltados en color los componentes que pueden utilizarse como puntos finales de la conexión en el proyecto para el tipo de conexión seleccionado. En el caso de los componentes puede tratarse de CPUs o aplicaciones para PC.



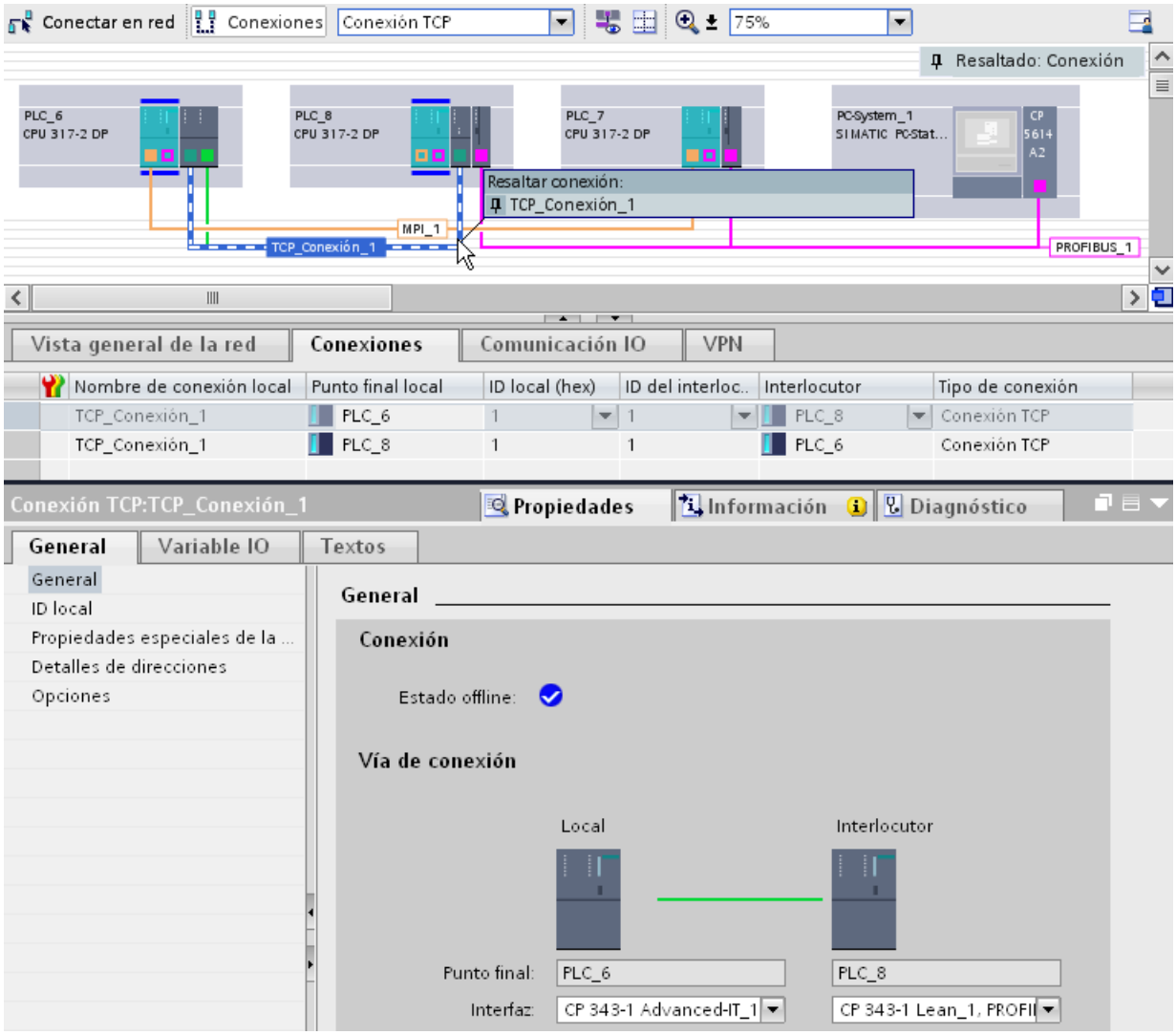
3. Seleccione el componente del que deba partir una conexión.
4. Para crear la conexión existen 2 posibilidades:
   – Haga clic en el componente del que debe partir una conexión y arrastre el cursor del ratón hasta el componente de destino. Confirme el punto final de la conexión mediante un nuevo clic.



   – Mientras mantiene presionada la tecla "Mayús", seleccione además el componentes de destino y elija con el botón derecho del ratón el comando "Agregar conexión".

Resultado

- Se ha creado una conexión especificada.
- La vía de conexión aparece resaltada. La vía de conexión se ha ajustado automáticamente conforme a los recursos de conexión e interfaces disponibles.
- La conexión ha sido registrada en las tablas de conexión de los componentes (CPU / aplicación para PC).

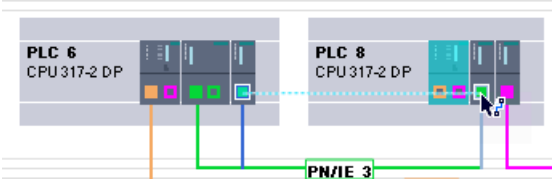


Seleccionar la vía de conexión de forma específica

Para crear una conexión con una vía de conexión seleccionada de forma explícita, proceda del siguiente modo:

1. Haga clic en el símbolo "Conexiones" de la barra de herramientas. De esta forma se activa el modo de conexión.

2. Seleccione el tipo de conexión necesario en la lista desplegable.
3. En la interfaz de subred, haga clic en el dispositivo del que deba partir una conexión.
4. Arrastre el puntero del ratón manteniendo el botón del mismo pulsado hasta la interfaz deseada en el dispositivo de destino y suelte el botón del ratón.



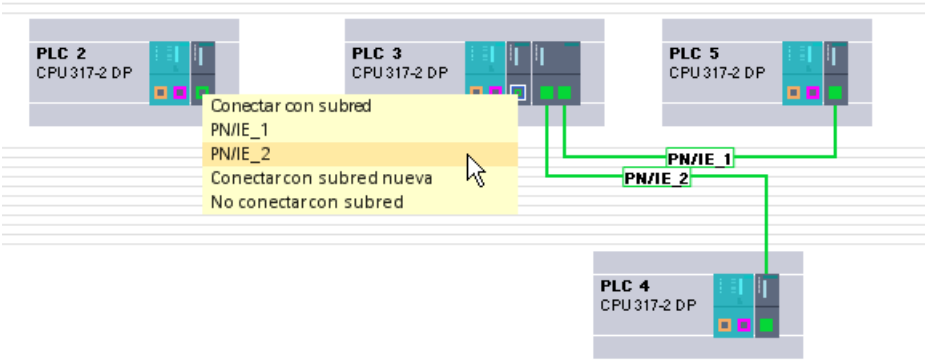
Resultado:

– Se ha creado una conexión especificada.
– La conexión creada entre los puntos finales utiliza las interfaces seleccionadas para la vía de conexión.
– La vía de conexión aparece resaltada.
– La conexión ha sido registrada en las tablas de conexión de los dispositivos (CPU / aplicación para PC).

Configurar una conexión cuando falta la asignación de red o ésta no es unívoca

Las interconexiones que faltan se establecen en la medida de lo posible de forma automática al crear una conexión. En caso de no ser posible una asignación de red unívoca, se produce una consulta al finalizar la configuración de la conexión. En este caso se ofrecen las subredes existentes para su selección.

Ejemplo en la siguiente figura: Al crear una conexión entre las estaciones PLC2 y PLC3 que aún no han sido interconectadas se produce una consulta.



Crear una conexión nueva a través de cuadros de diálogo

Configurar conexiones a través de cuadros de diálogo

La configuración de la conexión a través de cuadros de diálogo es muy útil para crear varias conexiones dentro de una estación en un solo paso de trabajo.

Por un lado se dispone del control completo sobre la selección de la vía de conexión, ya que el sistema muestra todas las variantes de interfaces. Por otro lado, tiene la posibilidad de delegar en el sistema la consideración de los recursos disponibles del sistema y, con ello, la selección de la vía de conexión.

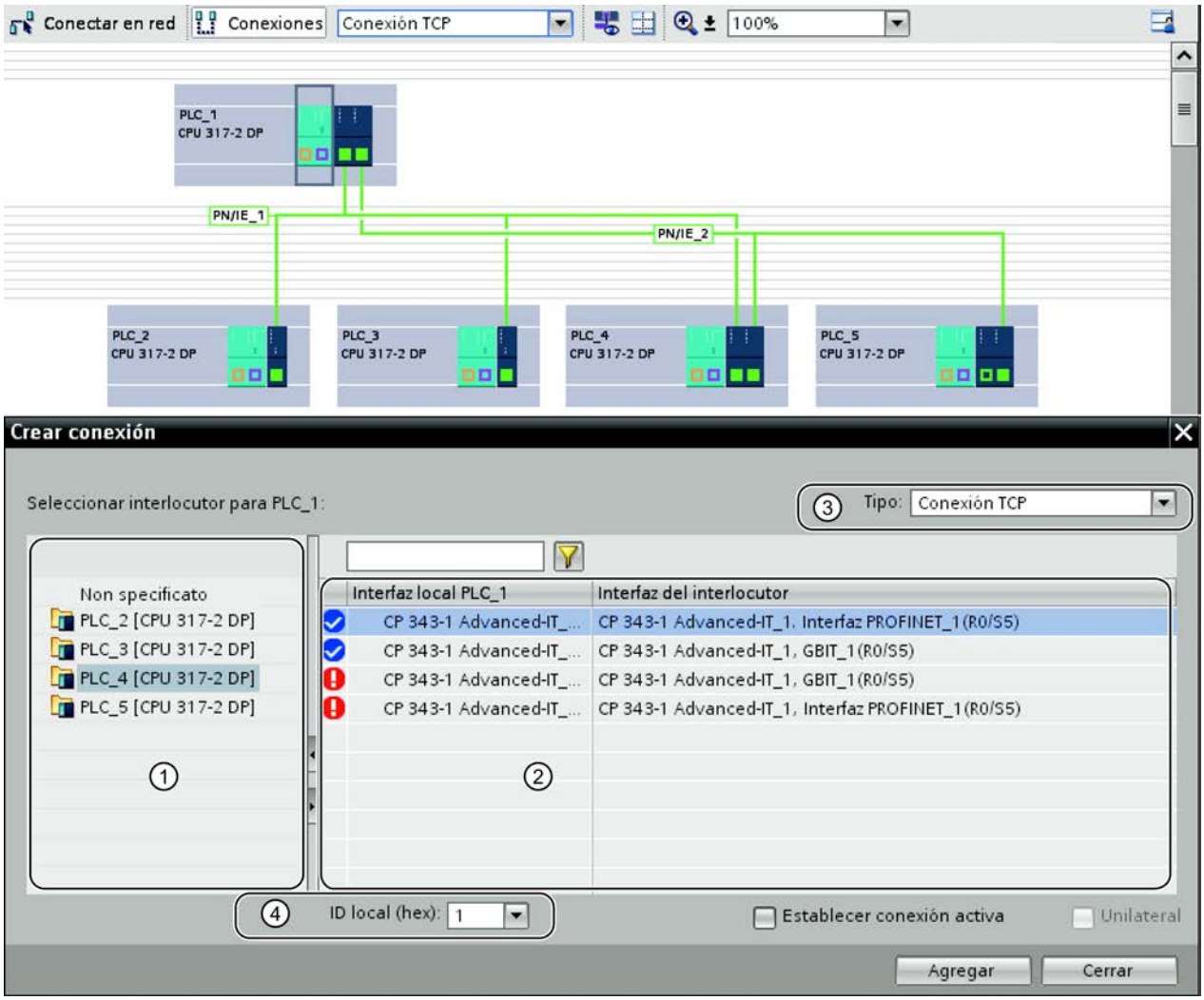
Nota

Configuración de conexiones a través de cuadros de diálogo: interconexión no automática

En la configuración de conexiones a través de cuadros de diálogo no se efectúan interconexiones automáticas.

Cuadro de diálogo "Crear conexión"

A continuación se muestra la estructura del cuadro de diálogo "Crear conexión". Se representa una interconexión entre varias estaciones. El modo de conexión está activado y se ha seleccionado la estación situada arriba.


Conectar en red
Conexiones
Conexión TCP
100%
PLC_1
CPU 317-2 DP
PN/IE_1
PN/IE_2
PLC_2
CPU 317-2 DP
PLC_3
CPU 317-2 DP
PLC_4
CPU 317-2 DP
PLC_5
CPU 317-2 DP
Crear conexión
Seleccionar interlocutor para PLC_1:
3
Tipo:
Conexión TCP
Non specificato
PLC_2 [CPU 317-2 DP]
PLC_3 [CPU 317-2 DP]
PLC_4 [CPU 317-2 DP]
PLC_5 [CPU 317-2 DP]
1
2
Interfaz local PLC_1
Interfaz del interlocutor
CP 343-1 Advanced-IT_...
CP 343-1 Advanced-IT_1, Interfaz PROFINET_1 (R0/S5)
CP 343-1 Advanced-IT_...
CP 343-1 Advanced-IT_1, GBIT_1 (R0/S5)
CP 343-1 Advanced-IT_...
CP 343-1 Advanced-IT_1, GBIT_1 (R0/S5)
CP 343-1 Advanced-IT_...
CP 343-1 Advanced-IT_1, Interfaz PROFINET_1 (R0/S5)
4
ID local (hex):
1
Establecer conexión activa
Unilateral
Agregar
Cerrar

① Visualización de los interlocutores posibles para el tipo de conexión seleccionado.
Seleccione de entre las siguientes alternativas:
- Dependiendo del tipo de conexión, una configuración especial:
  - "No especificada"
  - "Broadcast"
  - "Multicast"
- El interlocutor

② Área de contenido para la selección de las interfaces
Se muestran las posibles combinaciones de interfaces correspondientes a la configuración.
Solo se muestran las interfaces posibles para el tipo de conexión seleccionado.
La indicación del estado informa adicionalmente acerca de si es posible establecer la conexión dependiendo de la interconexión y los recursos disponibles.
Aunque exista una falta de interconexión entre las interfaces es posible establecer una conexión.
Consulte también Recursos disponibles en la configuración de conexiones (Página 592)
Ejemplo mostrado:
Entre PLC1 y PLC4 se encuentran interconectados en cada caso los puertos Gbit por un lado y las interfaces PROFINET por otro lado. En correspondencia, solo se muestran estas dos combinaciones como completamente asignables.

③ Lista desplegable para la selección del tipo de conexión

④ Parámetros de conexión preajustados

Indicación del estado

Icono	Significado
	Las interfaces están interconectadas y en los interlocutores seleccionados se dispone de recursos de conexión.
	Las interfaces no están interconectadas o no están conectadas a la misma subred.

Procedimiento

Para crear una conexión desde un cuadro de diálogo, proceda del siguiente modo:

1. Haga clic en el símbolo "Conexiones" de la barra de herramientas. De esta forma se activa el modo de conexión.

2. En el menú contextual del componente que debe convertirse en el punto final de la conexión (CPU o aplicación para PC), seleccione el comando "Agregar conexión".
   o
   Con la opción "Mostrar todas las conexiones" desactivada y con la CPU o aplicación seleccionada, elija el comando "Agregar conexión" en el menú contextual de la tabla de conexiones.
   Se abre el cuadro de diálogo "Crear conexión".
3. Seleccione en la lista desplegable "Tipo" el tipo de conexión deseado, p. ej. "Conexión FDL".

4. Compruebe en la línea inferior el ajuste previo de parámetros de conexión adicionales mostrados y corríjalos conforme a sus requisitos.

   Nota

   También puede modificar dichos parámetros a posteriori en la tabla de conexiones o en las propiedades de la conexión.

5. En el área izquierda del cuadro de diálogo, seleccione el interlocutor (dispositivo o CPU). En el área de contenido se muestran las posibles combinaciones de interfaces correspondientes a la configuración junto a la información de estado. Se encuentra seleccionada la primera combinación de interfaces a la que es posible asignar una conexión del tipo seleccionado.

   Nota

   Dependiendo del tipo de conexión seleccionado se ofrecen, además de los posibles interlocutores, tipos de conexión especiales tales como "no especificada", "Broadcast" o "Multicast".

6. Conserve la selección o seleccione la vía de conexión deseada.
7. Confirme la selección con el botón "Agregar".
   Se crea una conexión, que se muestra en la tabla de conexiones.
8. Si no ha cerrado el cuadro de diálogo, continúe alternativamente del modo siguiente:
   - Establezca otra conexión del mismo tipo con el mismo dispositivo de destino haciendo clic de nuevo en el botón "Agregar".
   - Repita el procedimiento indicado arriba con otro tipo de conexión de la lista desplegable u otro interlocutor.

Configurar una conexión cuando falta la interconexión

Las interconexiones que faltan se establecen en la medida de lo posible de forma automática al crear una conexión. En caso de ser posible una asignación de red unívoca, tras finalizar el cuadro de diálogo se mostrará la interfaz correspondiente interconectada en la vista de redes.

En caso de no ser posible una asignación de red unívoca, en la configuración de una conexión a través de cuadros de diálogo no se introducirá ninguna interconexión. Por esta razón, las conexiones creadas de esta forma aún no han sido completamente especificadas y se marcan en color rojo en la tabla de conexiones.

Trabajar en la vista de redes

Significado de la vista de redes en el modo de conexión

La vista de redes muestra dispositivos con sus componentes e interfaces a las subredes. En el modo de conexión, las vías de comunicación de las conexiones configuradas se pueden hacer visibles.

En lo relativo a las conexiones se puede apreciar:

- Entre qué dispositivos se han creado conexiones
- Qué tipos de conexión se han creado partiendo de un dispositivo
- Desde qué dispositivos parten conexiones no especificadas

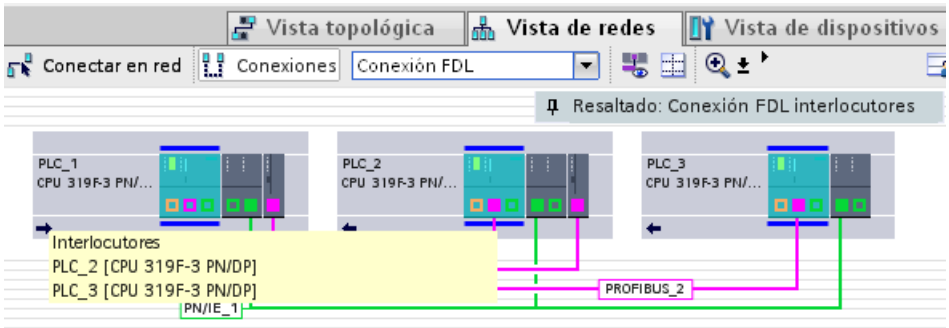
En el modo de conexión, la vista de redes ofrece las funciones siguientes:

- Crear conexiones gráficamente; consulte también Crear una conexión gráficamente (Página 621)
- Resaltar las conexiones de una subred de forma selectiva;
- Resaltar interlocutores.

Resaltar la vía de conexión y el interlocutor en la vista de redes

Para mostrar los interlocutores de todos o algunos tipos de conexión en la vista de redes, proceda del siguiente modo:

1. En el menú contextual del punto final de la conexión (CPU o aplicación para PC) cuyos interlocutores desea mostrar en la vista de redes, seleccione el comando de menú "Resaltar interlocutores".
2. En el menú que aparece a continuación, elija "Todos los interlocutores" o "Seleccionar el tipo de conexión..."
   Se seleccionan el punto final local y los puntos finales de los dispositivos de destino correspondientes a todos los tipos de conexión o al tipo de conexión correspondiente. Además, los dispositivos muestran una flecha.
3. Abra una lista con los datos de los dispositivos de destino seleccionando con el botón derecho del ratón la flecha situada en la línea inferior del dispositivo local. Esta función complementaria es útil en las configuraciones de red complejas en las que no se ven todos los dispositivos.



Nota

Es posible visualizar uno de los interlocutores que no se ve en el área de visualización actual de la vista de redes. Haga clic en el interlocutor en la lista que aparece. Resultado: La vista se desplaza de tal forma que se muestra el interlocutor.

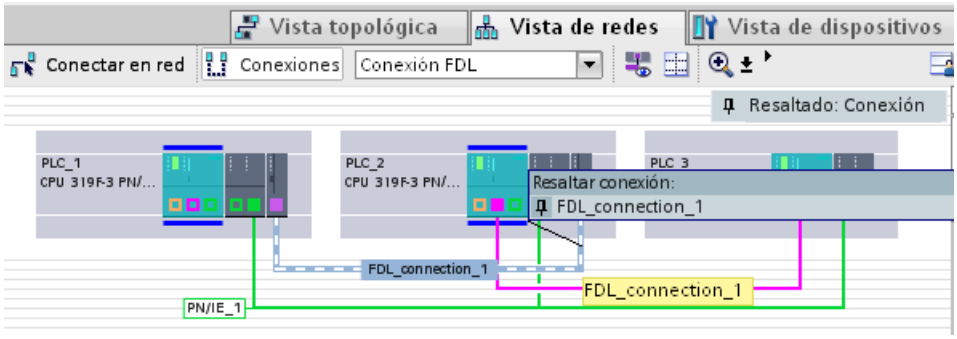
Resaltar / seleccionar una conexión

Es posible seleccionar conexiones individuales para que queden resaltadas. Con ello se consigue lo siguiente:

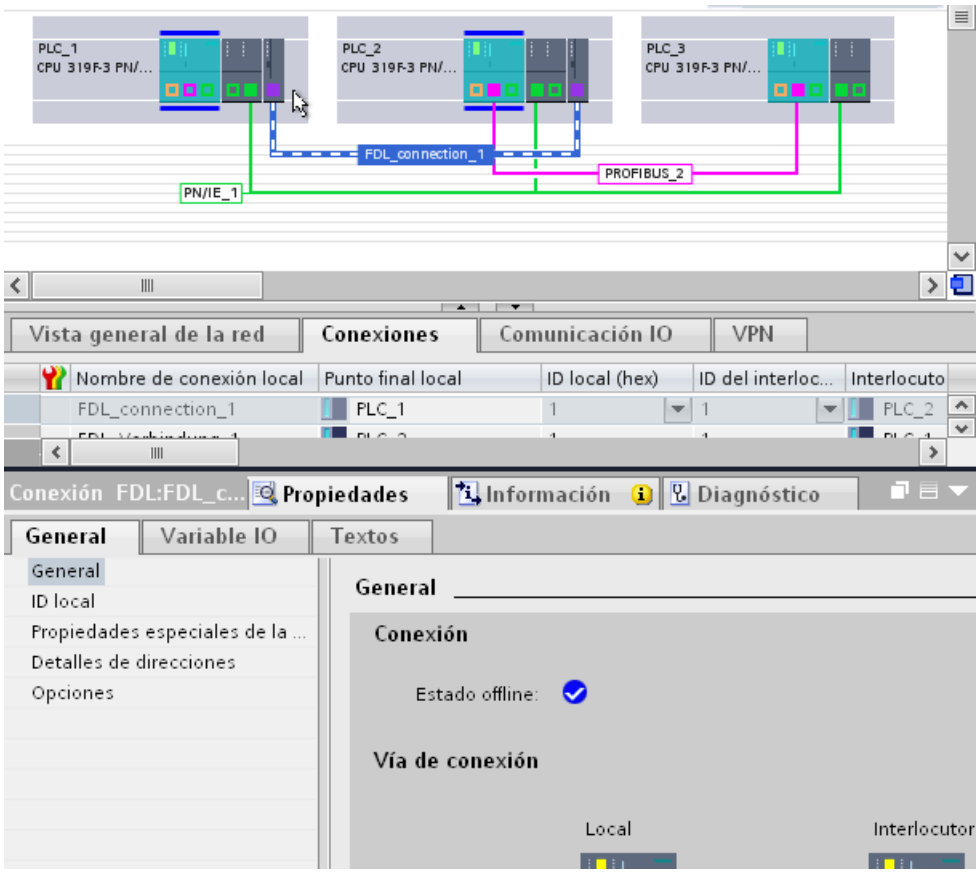
- La conexión está seleccionada en la tabla de conexiones; ello permite ajustar en ella determinadas propiedades.
- La conexión se muestra en la ventana de inspección con todas las posibilidades de parametrización.

Proceda del siguiente modo:

1. Sitúe el puntero del ratón sobre la subred.
2. Seleccione y marque la conexión deseada en el cuadro de diálogo que aparece "Resaltar la conexión".
   La vía de conexión se muestra en forma de línea discontinua azul clara.



3. Haga clic en la línea discontinua.
   El color de la línea pasa a azul oscuro; la conexión se muestra tanto en la tabla de conexiones como en la ventana de inspección.



Trabajar con la tabla de conexiones

Funciones

Además de la posibilidad de crear conexiones a través de cuadros de diálogo, la tabla de conexiones ofrece otras funciones:

- Relación de las conexiones con un objeto seleccionado en la vista de redes (red, CPU, CP)
- Relación de todas las conexiones del proyecto
- Selección de una conexión y su visualización en la vista de redes
- Cambio de interlocutores
- Posibilidad de entrada de parámetros centrales
- Visualización de información de estado

Crear conexiones a través de cuadros de diálogo

Crear una conexión nueva a través de cuadros de diálogo (Página 625)

Contenido de la tabla de conexiones

La tabla de conexiones permite visualizar en detalle todas las conexiones creadas en el proyecto:

- Mostrar todas las conexiones (ajuste predeterminado, seleccionable en el menú contextual)
  La tabla de conexiones muestra todas las conexiones creadas en el proyecto.
- Mostrar las conexiones relacionadas con la selección (ajuste recomendado para proyectos complejos)
  Desactive para ello la opción "Mostrar todas las conexiones" mediante el menú contextual.
  La tabla de conexiones muestra las conexiones correspondientes a una subred o una CPU/un CP seleccionado. Por tanto, se encuentra vacía cuando no se selecciona en la vista de redes ninguno de los objetos mencionados.
  También es posible seleccionar simultáneamente varios objetos del mismo tipo en cada caso (varias subredes, varias CPUs/CPs o varias interfaces). La tabla de conexiones muestra en este caso todas las conexiones correspondientes.

Funciones básicas de las tablas

La tabla de conexiones soporta las siguientes funciones básicas para editar una tabla:

- Mostrar y ocultar las columnas de la tabla de forma selectiva
  Seleccione para ello el comando "Mostrar/ocultar columnas" en el menú contextual de la fila superior de la tabla.
- Modificar manualmente el ancho de las columnas
  Desplace para ello la línea de separación situada en el lado derecho de la columna, en la fila superior.

- Optimizar automáticamente el ancho de las columnas
  Seleccione para ello el comando "Optimizar ancho de columnas" en el menú contextual del encabezado de la tabla.
- Mostrar el significado de una columna mediante tooltips.

Si desea guardar la disposición, el ancho y la visibilidad de las columnas, haga clic en la función "Recordar disposición" en la parte superior derecha de la vista de redes.

Símbolo	Significado
	Recordar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Navegar por la tabla de conexiones con las teclas de cursor

Las teclas de cursor FLECHA ARRIBA y FLECHA ABAJO permiten seleccionar una conexión en la tabla de conexiones; la conexión queda seleccionada y se resalta en la vista de redes cuando el modo de conexión se encuentra activado.

Modificar las propiedades de la conexión

Algunos de los parámetros visualizados en la tabla de conexiones pueden editarse directamente. Así, por ejemplo, el nombre de la conexión sólo puede modificarse en la tabla de conexiones.

Modificar el interlocutor

El interlocutor de una conexión se modifica del siguiente modo:

1. Seleccione la conexión.
2. Elija el nuevo interlocutor en la lista desplegable activada de la columna "Interlocutor".

Modificar el interlocutor

Para una conexión ya configurada, modifique en caso necesario el interlocutor según la siguiente descripción. Se mantienen la ID local y el tipo de conexión.

Solo son posibles las modificaciones de los puntos finales de la conexión en la tabla de conexiones o en la ventana de inspección.

Modificar conexiones a través de cuadros de diálogo

Para acceder a otro interlocutor en una vía de conexión predeterminada, proceda del siguiente modo:

1. Marque en la vista de redes el punto final local para el que desea asignar un interlocutor nuevo.
   En la tabla de conexiones se visualizan las conexiones correspondientes.
2. Marque la conexión deseada en la tabla de conexiones.
3. Elija el nuevo dispositivo de destino para la conexión mediante la lista desplegable de la columna de la tabla "Interlocutor".
   Como alternativa, seleccione el nuevo dispositivo de destino en "Vía de conexión" de la ventana de inspección.

Nota

Efecto en las propiedades ya configuradas

Tenga en cuenta que al modificar el interlocutor, las propiedades parametrizadas de la conexión se reinician al ajuste predefinido.

Utilice las posibilidades de entrada en la tabla de conexiones o en la ventana de inspección cuando desee modificar las "Propiedades" de la conexión.

Nota

Efecto en la interconexión

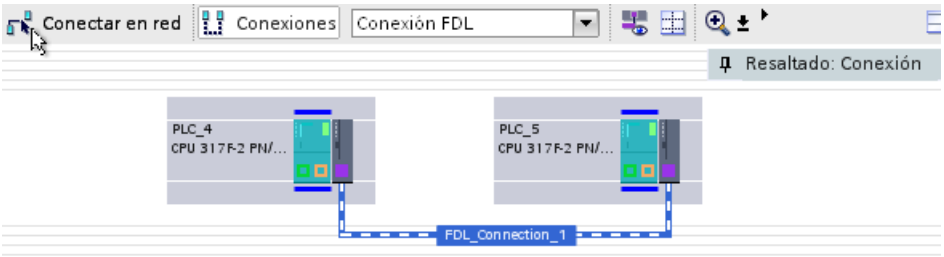
En el caso de los anteriores, para modificar el interlocutor no se crean subredes de forma automática cuando falta la interconexión con el nuevo punto final de la conexión.

Ajustar o modificar la vía de conexión

Dependiendo de los componentes e interfaces disponibles dentro de un dispositivo, es posible trazar conexiones a lo largo de diferentes vías dentro del dispositivo.

Dependiendo del estado de la configuración del dispositivo, en la configuración existen conexiones con el siguiente estado de conexión, las cuales se representan de la forma correspondiente:

- Vía de conexión cerrada
  Se encuentran disponibles ambos puntos finales (CPU o aplicación para PC), así como las interfaces necesarias para la comunicación (local en la CPU o CP adicional).



- Vía de conexión abierta
  - La vía de conexión no es completamente asignable debido a las interfaces que faltan.
  - La vía de conexión se ha definido únicamente en un punto final debido a una conexión no especificada.

A continuación se describe el procedimiento para modificar o volver a asignar una vía de conexión configurada en un dispositivo local o interlocutor.

La vía de conexión se ajusta o modifica en las siguientes vistas:

- Ventana de inspección en "Propiedades > General"

Modificar la vía de conexión en la ventana de inspección

Para asignar otra vía de conexión a una conexión existente, proceda del siguiente modo:

1. Seleccione en la vista de redes el punto final (CPU o aplicación para PC) para el que se desea modificar una conexión.
2. Seleccione en la tabla de conexiones la conexión cuya vía de conexión desea modificar.
   Sugerencia: En la vista de redes, la vía de conexión actual aparece resaltada si se ha ajustado la vista "Resaltar conexión" de la forma correspondiente.
3. Seleccione la interfaz nueva en "Propiedades > General > Interfaz" de la ventana de inspección. En el interlocutor se ofrecen las interfaces posibles en el contexto de la interfaz local (observación: en principio, las conexiones S7 se pueden trazar a través de distintos tipos de subred).
   En la vista de redes se representa la nueva vía de conexión resaltada.

Asignar la vía de conexión

Para asignar una vía de conexión cerrada a una vía de conexión existente, proceda del siguiente modo:

1. Amplíe la configuración del dispositivo de tal forma que las interfaces necesarias para el tipo de conexión se encuentren disponibles para ambos interlocutores.
2. Marque en la tabla de conexiones la conexión cuya vía de conexión desea asignar.
   Sugerencia: En la vista de redes se representa la vía de conexión actual resaltada si se ha ajustado la vista "Resaltar conexión" de la forma correspondiente.
3. Haga clic en el botón "Buscar vía de conexión" en "Propiedades > General > Interfaz" de la ventana de inspección.
   En la vista de redes se representa la nueva vía de conexión resaltada.

Crear una conexión no especificada

En los siguientes casos se definen conexiones no especificadas de forma específica seleccionando solo el interlocutor local de forma gráfica o desde un cuadro de diálogo:

- Si determinadas propiedades de servicio o conexión requieren una conexión no especificada (ejemplos: conexión de correo electrónico, conexión FTP, conexión FDL con la propiedad "Acceso libre Layer 2").
- Cuando es necesario establecer una conexión que debe estar preparada para la comunicación con cualquier interlocutor.
- Cuando uno de ambos interlocutores (aún) no se encuentra disponible para el tipo de conexión seleccionado pero se desea aun así reservar un recurso de conexión.
- Cuando el interlocutor se encuentra fuera del proyecto (p. ej. estaciones ajenas).
- Conexión UDP con Broadcast/Multicast:
  - véase UDP con Broadcast y Multicast (Página 671)
  - véase Conexión UDP libre (Página 674)

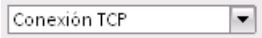
Requisitos

Está abierta la vista de redes.

Configurar gráficamente una conexión no especificada

Para crear una conexión no especificada gráficamente, proceda del siguiente modo:

1. En el modo de conexión, seleccione el tipo de conexión necesario en la lista desplegable, por ejemplo, "Conexión TCP".

   Las CPUs posibles para el tipo de conexión seleccionado han sido seleccionadas y se representan resaltadas.
2. Haga clic en la CPU de la que deba partir una conexión.
3. Desplace el puntero del ratón dentro de la CPU o sobre la interfaz del CP y confirme el punto final de la conexión haciendo doble clic con el ratón.

Resultado

- Se ha creado una conexión no especificada. Dependiendo del punto final seleccionado (CPU o CP) rige para la selección de la vía:
  - Punto final CPU: Se selecciona la mejor vía de conexión posible de forma automática y local. Consulte también Crear una conexión gráficamente (Página 621)
  - Punto final interfaz CP: La vía de conexión se determina mediante la introducción de datos.
- La conexión no especificada aparece resaltada mediante una línea corta discontinua.

- La conexión no especificada está registrada en la tabla de conexiones del dispositivo; en la columna "Interlocutor" aparece la entrada "desconocido".

Configurar una conexión no especificada a través de cuadros de diálogo

Para crear una conexión no especificada desde un cuadro de diálogo, proceda del siguiente modo:

1. En el menú contextual del dispositivo (CPU) para el que debe crearse una conexión, elija el comando "Agregar conexión".
   o
   Con la opción "Mostrar todas las conexiones" desactivada, seleccione el comando "Agregar conexión" en el menú contextual de la tabla de conexiones.
   Resultado: Se abre el cuadro de diálogo "Crear conexión".
2. Seleccione en el cuadro de diálogo "Crear conexión" el tipo de conexión deseado, p. ej. "Conexión FDL".
3. En la zona superior del cuadro de diálogo, seleccione la entrada "no especificada".
4. Si hay varias interfaces locales disponibles, la interfaz deseada se puede seleccionar de una lista. En la primera línea de la lista aparece preseleccionada la interfaz identificada por el sistema como la mejor posible.
5. Confirme la selección con el botón "Agregar".
   Se crea una conexión no especificada, que se muestra en la tabla de conexiones.
6. Repita el proceso o salga del cuadro de diálogo mediante "Aceptar".

Resultado

- Se ha creado una o varias conexiones no especificadas.
- Las conexiones no especificadas están registradas en la tabla de conexiones.

Nota

Recuerde que además se pueden parametrizar propiedades especiales para cada conexión.

Crear conexiones UDP/FDL con Multicast/Broadcast

Significado

Multicast y Broadcast son opciones de conexión especiales que son soportadas y configurables por CPs Industrial Ethernet en el caso de conexiones UDP y por PROFIBUS en el caso de conexiones FDL.

Crear una conexión a través de cuadros de diálogo: procedimiento

Para crear una conexión Multicast/Broadcast a través de cuadros de diálogo, proceda del siguiente modo:

1. En el menú contextual del componente que debe convertirse en el punto final de la conexión, seleccione el comando "Agregar conexión".
   o
   Con la opción "Mostrar todas las conexiones" desactivada, seleccione el comando "Agregar conexión" en el menú contextual de la tabla de conexiones.
   Se abre el cuadro de diálogo "Crear conexión".
2. En la lista desplegable "Tipo", seleccione el tipo de conexión Conexión UDP/Conexión FDL.
3. En el área izquierda del cuadro de diálogo, seleccione el tipo de conexión Broadcast o Multicast.
   En el área de contenido aparecen las interfaces locales posibles conforme a la configuración con su información de estado. Aparece preseleccionada la primera interfaz a la que puede asignarse una conexión del tipo seleccionado.
4. Conserve la selección o seleccione la vía de conexión deseada.
5. Confirme la selección con el botón "Agregar".
   Se crea una conexión, que se muestra en la tabla de conexiones.
6. Si no ha finalizado el cuadro de diálogo, continúe alternativamente del siguiente modo:
   - Establezca otra conexión del mismo tipo con el mismo dispositivo de destino haciendo clic de nuevo en el botón "Agregar".
   - Repita el procedimiento indicado arriba con otro tipo de conexión de la lista desplegable u otro interlocutor.

Crear la conexión gráficamente: procedimiento

1. En primer lugar, cree una conexión UDP/conexión FDL no especificada en la configuración gráfica de conexiones; Crear una conexión gráficamente (Página 621)
2. A continuación, configure los ajustes de la dirección del interlocutor conforme a los requisitos para Multicast o Broadcast. De manera alternativa, proceda del modo siguiente:
   - Seleccione el tipo de conexión deseado en la lista desplegable de la columna "Interlocutor" de la tabla de conexiones; los parámetros de dirección se asignan automáticamente.
     o
   - Con UDP, introduzca la dirección IP o el puerto directamente en "General > Detalles de direcciones". Con FDL, ajuste la dirección de dispositivo PROFIBUS 63.

Consulte también

UDP con Broadcast y Multicast (Página 671)

Conexión FDL con Broadcast (Página 684)

Conexión FDL con Multicast (Página 685)

Especificar una conexión no especificada

Características de la conexión no especificada

Para una conexión no especificada faltan, dependiendo del tipo de conexión, todos o parte de los parámetros necesarios para el acceso local a la red y para el direccionamiento del interlocutor.

Una conexión no especificada se representa como sigue:

- Vista de redes:
  Tras la creación de una nueva conexión o cuando la conexión ha sido marcada en la tabla de conexiones, la conexión no especificada aparece en la vista de redes como una línea discontinua.
- Tabla de conexiones:
  - Visualización en la columna "Interlocutor": "desconocido"

Una conexión no especificada se crea como sigue: Consulte también Crear una conexión no especificada (Página 637)

Al especificar a posteriori la conexión debe distinguirse en principio:

- El interlocutor se encuentra en el mismo proyecto
  La conexión se especifica mediante la selección a posteriori de los componentes disponibles en el proyecto.
- El interlocutor se encuentra fuera del proyecto
  La conexión se especifica introduciendo parámetros de dirección de los interlocutores disponibles fuera del proyecto.

Complementación automática de los parámetros de conexión que faltan

Si se soluciona la falta de recursos mediante la introducción de componentes, se asignarán conexiones en la medida de lo posible de forma automática, formando con ello una conexión totalmente especificada en caso dado.

Requisitos

- Los componentes de ambos dispositivos cuentan con recursos libres para el tipo de conexión.

Procedimiento en la tabla de conexiones

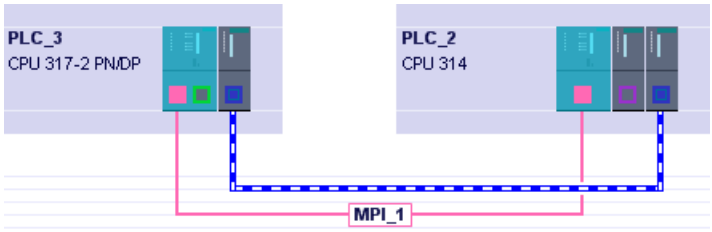
Para aportar a una conexión no especificada los parámetros de dirección de un dispositivo de destino disponible en el proyecto, proceda del siguiente modo:

1. Seleccione la CPU cuya conexión desea especificar.
   En la tabla de conexiones, todas las conexiones no especificadas configuradas para la CPU se identifican con la observación "desconocido" en la columna "Interlocutor".
2. Seleccione el dispositivo de destino deseado en la columna "Interlocutor" de la tabla de conexiones.

Resultado

- La conexión está especificada con los detalles de direcciones correspondientes para el interlocutor.
- En la vista de redes se representa la nueva vía de conexión resaltada como una línea discontinua.

La siguiente representación muestra una conexión especificada:



Borrar una conexión

Las conexiones configuradas se borran en la vista de redes o en la tabla de conexiones.

En la vista de redes se puede borrar en cada caso una conexión resaltada. En la tabla de conexiones se borran una o varias conexiones.

Procedimiento en la vista de redes

En la vista de redes se pueden borrar conexiones individuales mediante el menú contextual. Proceda del siguiente modo:

1. Conmute al modo de conexión.
2. Sitúe el puntero del ratón sobre la subred.
   El cuadro de diálogo "Resaltar la conexión" muestra las conexiones configuradas actualmente.
3. Seleccione la conexión que desea borrar.

4. Vuelva a situar el puntero del ratón sobre la subred.
5. Abra el menú contextual con el botón derecho del ratón.
6. Seleccione el comando "Borrar".
   La conexión seleccionada se borra por completo. Si para una conexión especificada el interlocutor se encuentra dentro del proyecto, la conexión se borra también de la tabla de conexiones del interlocutor.

Nota

Cargar los datos de conexión en el módulo

Cargue la tabla de conexiones con la conexión borrada en los módulos programables afectados por la operación de borrado.

Procedimiento en la tabla de conexiones / vista de redes

En la tabla de conexiones se pueden borrar una o más conexiones seleccionadas mediante el menú contextual. Proceda del siguiente modo:

1. Seleccione en la tabla las conexiones que desea borrar.
2. Seleccione el menú contextual con el botón derecho del ratón.
3. Elija el comando "Borrar".
   La conexión seleccionada en cada caso se borra por completo. Si para una conexión especificada el interlocutor se encuentra dentro del proyecto, la conexión se borra también de la tabla de conexiones del interlocutor.
4. Cargue la tabla de conexiones con las conexiones borradas en el módulo programable correspondiente.

Nota

Cargar los datos de conexión en el módulo

Cargue la tabla de conexiones con la conexión borrada en los módulos afectados por la operación de borrado.

Nota

Para borrar todas las conexiones del módulo se debe cargar una tabla de conexiones vacía.

Copiar una conexión

Introducción

Las conexiones no se copian individualmente, sino siempre en relación con el proyecto o dispositivo.

Es posible copiar:

- Proyectos enteros
- Uno o varios dispositivos dentro de un proyecto o de varios proyectos

Copiar un proyecto

Al copiar un proyecto se copian también todas las conexiones configuradas. No es necesario realizar ajustes en las conexiones copiadas porque siguen siendo coherentes.

Copiar dispositivos

- Copiar el dispositivo dentro de un proyecto
  Cuando se copian dispositivos individuales se crean conexiones no especificadas y es necesario volver a asignar los interlocutores al dispositivo local.
  Al copiar un grupo de dispositivos se copian también las conexiones dentro del grupo de dispositivos. No obstante, debido al duplicado se crean en este caso duplicados de los nombres de las conexiones. Modifique estos nombres de conexión tras el proceso de copia.
  Los dispositivos copiados se deben conectar nuevamente en red.
- Copiar dispositivos en otro proyecto
  Es posible copiar uno o varios dispositivos en otro proyecto. Con el fin de procesar en paralelo el otro proyecto, reinicie el TIA Portal.
  Las conexiones configuradas entre los dispositivos copiados permanecen sin cambios en el proyecto de destino. En el proyecto de destino se vuelve a asignar únicamente la interconexión.
  Si copia los dispositivos en el proyecto de destino uno tras otro, se generan conexiones temporales no especificadas. No obstante, al introducir otro dispositivo se vuelven a asignar los interlocutores en tanto sea posible.

Información adicional

Consulte también Especificar una conexión no especificada (Página 640)

Conexiones incoherentes - Conexiones sin asignación

Cuando una conexión es incoherente, significa que los datos de conexión no son completos o unívocos; la conexión no funciona en el contexto del proyecto.

Las conexiones incoherentes no pueden cargarse, como tampoco es posible el funcionamiento con una conexión de este tipo.

En la tabla de conexiones, las conexiones incoherentes se reconocen por el color rojo.

Nota

Conexiones no especificadas

En general, las conexiones no especificadas no son incoherentes aunque se marquen en color rojo en la tabla de conexiones.

Crear una conexión no especificada (Página 637)

Causas posibles de conexiones incoherentes

- La configuración hardware se ha borrado o modificado
- Falta de conexiones en red entre las interfaces del proyecto necesarias para una conexión.
- Rebase por exceso de los recursos de conexión
- Conexiones con un interlocutor no especificado sin indicar la dirección del mismo.

Puede consultar la información detallada acerca de la causa de la incoherencia en la ventana de inspección de la página de propiedades de una conexión; en ella los parámetros incoherentes se marcan en color rojo. Puede consultar la información adicional referente a dichos parámetros situando el puntero del ratón sobre el parámetro.

Ejemplos y ayuda para incoherencias en acciones típicas dentro de la configuración

A continuación se aclaran las acciones que pueden llevar a la pérdida o eliminación de la asignación de conexiones configuradas.

Nota

Tenga en cuenta que, al contrario que en las conexiones S7, se asigna una ID dependiente del CP a las conexiones de la interfaz SEND/RECEIVE. Por tanto, para las acciones que se describen a continuación pueden producirse adaptaciones de la ID, de tal forma que en el programa de usuario sea necesario adaptar asimismo la interfaz.

Si se sustituye un CP por otro, éste debe ofrecer como mínimo los mismos servicios y debe ser como mínimo de la misma versión. Solo de esta forma se garantiza que las conexiones configuradas mediante el CP permanezcan coherentes y puedan ser utilizadas.

Tabla 8-3 Acciones en una interfaz de CP que llevan a modificaciones en conexiones configuradas

Acción	Consecuencia para las conexiones	Qué es necesario hacer para volver a establecer la conexión
Borrar el CP (módulo).	Las conexiones se conservan en la tabla de conexiones sin asignación a un CP. La asignación a la CPU/aplicación PC se conserva.	Después de colocar e interconectar un CP en la configuración del hardware: 1. Asignar el CP a la conexión; 2. Comprobar la dirección inicial de los módulos LADDR y, en caso dado, adaptarla en el programa de usuario. 3. Adaptar las IDs de la conexión en el programa de usuario. 4. Cargar de nuevo la configuración de la conexión en el CP.
Borrar un dispositivo.	Todas las conexiones a este dispositivo se deshacen en el proyecto: • Las conexiones se conservan en el interlocutor como conexiones no especificadas. • Las conexiones sin interlocutor se eliminan por completo.	Volver a configurar el dispositivo y las conexiones. Reasignar o eliminar las conexiones no especificadas.
Modificar la asignación de subred del CP.	Las conexiones asignadas a través del CP se conservan, pero pueden resultar incoherentes.	Reasignar conexiones en caso necesario.

Propiedades de otros tipos de conexión

Conexión S7

Ajustes generales de la conexión S7

Parámetros de conexión generales

En el grupo de parámetros "General" de las propiedades de la conexión se muestran los parámetros generales que identifican el punto final local de la conexión.

Aquí existe la posibilidad de asignar la vía de conexión y especificar por completo a los interlocutores.

ID local

Aquí se muestra la ID local del módulo desde el que se observa la conexión (interlocutor local). La ID local se puede modificar. Esto es necesario cuando ya se han programado bloques de función de comunicación y debe utilizarse la ID local indicada allí para la conexión.

Propiedades de conexión especiales

Visualización de las propiedades de conexión (modificable en función de los componentes utilizables):

- Unilateral
  Unilateral significa que el interlocutor es el servidor de dicha conexión y que no puede enviar o recibir activamente.
  En las conexiones unilaterales no se pueden emplear instrucciones de conexión bilaterales ("parejas de módulos"), como USEND (Página 7683) y URCV (Página 7685), BSEND (Página 7688) y BRCV (Página 7691) o USTATUS (Página 7722).
- Establecimiento activo de la conexión
  Determine con esta opción si el establecimiento de la conexión debe realizarse desde este dispositivo.
  Si ha seleccionado como interlocutor "no especificado" al crear la conexión, la opción está deseleccionada por defecto. Si se activa esta opción, es necesario especificar la dirección del interlocutor en el registro "Direcciones".
- Enviar avisos de estado operativo
  Indica si el interlocutor local envía avisos de estado operativo al interlocutor.

Detalles de direcciones

Visualización de los detalles de direcciones de la conexión S7. Si un interlocutor no está especificado, pueden modificarse los valores de rack y slot. Todos los demás valores se tomarán de la configuración actual y no podrán modificarse.

Detalles de direcciones de la conexión S7

Significado

Un recurso de conexión, tal y como se muestra en "Propiedades > Detalles de direcciones", forma parte de TSAP (Transport Service Access Point) del módulo local o del interlocutor.

Existen limitaciones y reglas en referencia a la posibilidad de uso de los recursos de conexión. No todos los recursos de conexión son p. ej. utilizables para cada tipo de conexión.

Rangos de valores

La siguiente tabla informa acerca del significado de los valores y el tipo de conexión. Dependiendo del interlocutor y el tipo de conexión, el rango de valores se limita automáticamente a valores válidos, o el valor del recurso de conexión se asigna de forma fija.

Recurso de conexión (1er byte del TSAP)	Significado	Tipo	Significado
0x01 ("PG")	Conexión PG	Conexión libre (no está configurada)	Se ha reservado como mínimo un recurso por CPU para conexiones PG. En el caso de determinadas CPUs S7-300 es posible reservar varios recursos en las propiedades de la CPU.
0x02 ("OP")	Conexión OP	Conexión libre (no está configurada)	Se ha reservado como mínimo un recurso por CPU para conexiones OP. En el caso de determinadas CPUs S7-300 es posible reservar varios recursos en las propiedades de la CPU.
0x03	otras	Conexión libre (conexión configurada no especificada)	Este recurso de conexión puede dar servicio a varias conexiones. Utilización: ¡Conexión unilateral para interlocutor no especificado! No es necesario configurar el interlocutor cuando se direcciona el recurso de conexión 0x03.
0x10..0xDF	Conexiones con establecimiento de conexión estático o dinámico	Conexión configurada no especificada	Uno de estos recursos de conexión puede dar servicio exactamente a una conexión. Utilización: ¡Conexión bilateral para interlocutor no especificado!

Conexiones S7 libres

Las conexiones S7 libres se establecen de forma dinámica durante el tiempo de funcionamiento; existen las siguientes conexiones libres:

- Conexiones PG (0x01)
  Conexiones S7 establecidas normalmente por una programadora o un PC (con funcionalidad ES). La estación/el módulo direccionado se configura, programa, comprueba y pone en funcionamiento mediante una conexión de este tipo; por regla general, la conexión vuelve a eliminarse a continuación. Mediante este recurso de conexión es posible tanto la lectura como la escritura (p. ej. observar y cargar).
- Conexiones OP (0x02)
  Conexiones S7 establecidas normalmente desde un OP o un PC (con funcionalidad OS). Mediante una conexión de este tipo se observa la estación/el módulo en referencia al proceso controlado.

- Otras (0x03)
  La utilización no ha sido fijada. Durante la configuración de una conexión unilateral S7 desde un S7-400 hasta un S7-300 se utiliza p. ej. este recurso de forma automática.
- Comunicación básica S7 (0xFD)
  Conexiones establecidas normalmente desde una CPU hasta otro módulo (CPU, FM, ...) dentro de una subred. El establecimiento de conexión es iniciado por el programa de aplicación y no existe ninguna configuración de conexión. Mediante una conexión de este tipo es posible intercambiar datos de proceso entre los módulos. En el caso de determinadas CPUs S7-300 es posible reservar recursos para la comunicación básica S7.

Recomendación para la selección del recurso de conexión

Para el interlocutor local, STEP 7 propone siempre un recurso de conexión libre. Recomendamos utilizar el recurso propuesto.

Conexión de transporte ISO

Transporte ISO - Propiedades de conexión especiales

Referencia

En el grupo de parámetros "Propiedades > General > Propiedades especiales de la conexión" se pueden definir propiedades adicionales del punto final local de la conexión.

Establecer conexión activa

Determine con esta opción si el establecimiento de la conexión debe realizarse desde este dispositivo. Esta opción está seleccionada por defecto si la dirección del interlocutor está especificada.

- ON: La conexión se establece activamente.
- OFF: El interlocutor establece la conexión.

Si ha seleccionado como interlocutor "no especificado" al crear la conexión, la opción está deseleccionada por defecto. Si se activa esta opción, es necesario especificar la dirección del interlocutor en el registro "Direcciones".

Nota

Efecto en el modo de operación

Tenga en cuenta el efecto en el modo de operación.

- S7-300 / S7-400
  Si se debe utilizar el modo de operación FETCH o WRITE (véase el grupo de parámetros "Opciones"), deberá desactivarse la opción "Establecimiento activo de la conexión".
- Servidor OPC
  Si se debe utilizar el modo de operación FETCH o WRITE (véase el grupo de parámetros "Opciones"), deberá activarse la opción "Establecimiento activo de la conexión".

Nombre

Al crear la conexión se visualiza el nombre propuesto para el punto final de la conexión, que contiene un número de conexión como sufijo.

Utilice este campo para identificar el interlocutor en caso de conexiones no especificadas.

Detalles del establecimiento OPC ISO

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > Detalles de establecimiento"

Opciones del establecimiento de la conexión

En esta área sólo pueden realizarse entradas si en "Propiedades especiales de la conexión" se ha seleccionado el establecimiento activo de la conexión.

Elija una de las dos alternativas siguientes:

- Mantener la conexión establecida permanentemente
  La conexión se establece y se mantiene al arrancar la estación PC así como en caso de interrumpirse la conexión.
- Establecer la conexión a petición (acceso a variable)
  Aquí se indica adicionalmente durante cuánto tiempo debe mantenerse la conexión después de un acceso.

Monitorización del establecimiento de la conexión

- Tiempo excedido: Tiempo de espera de errores durante el establecimiento de la conexión. Si dentro del tiempo de espera no es posible establecer la conexión, ésta se inicializará y la petición pendiente se acusará con error.
  Rango de valores: 0 – 99999 ms
  Valor predeterminado: 15000 ms

Nota

La monitorización del establecimiento de la conexión que se ajusta aquí está relacionada del siguiente modo con la vigilancia de tiempo que puede ajustarse en la configuración de módulos: el valor válido es el más pequeño.

En la configuración de módulos, la vigilancia de tiempo se configura en "Propiedades de interfaz Ethernet".

Transporte ISO, detalles de la dirección

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Detalles de direcciones".

Parámetros de dirección

Una conexión de transporte ISO (ISO Transport) se especifica mediante los puntos finales local y remoto de la conexión.

- Direcciones locales:
  Dirección MAC local y TSAP (Transport Service Access Point) local
- Direcciones remotas:
  Dirección MAC remota y TSAP remoto

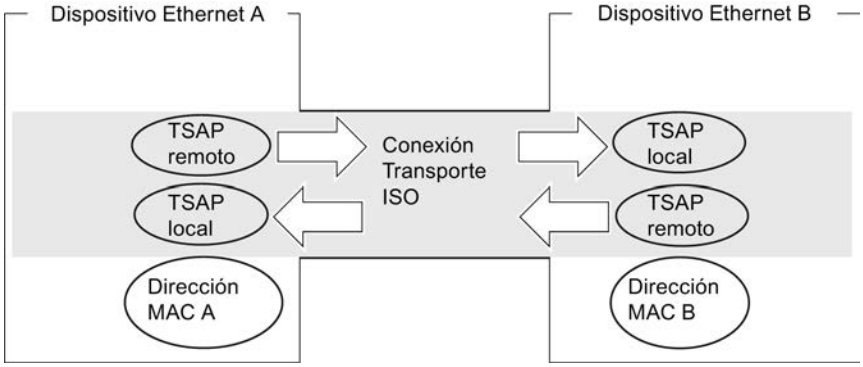


Figura 8-1 Emisión y recepción a través de una conexión de transporte ISO en cada caso

Formato TSAP

Las conexiones de transporte ISO tienen una longitud de TSAP de entre 1 y 16 bytes. Al realizar la entrada se muestra automáticamente la longitud actual (visualización: 16 caracteres ASCII). Los TSAPs locales y remotos pueden introducirse como valor hexadecimal o como string ASCII. En caso de una entrada ASCII, los caracteres introducidos también se visualizan en formato hexadecimal. En caso de entrada hexadecimal, los caracteres imprimibles se representan como valor ASCII (son visibles 8 caracteres hexadecimales). Si se introducen caracteres no imprimibles, la visualización ASCII queda atenuada (no es posible la entrada ASCII) y los caracteres no imprimibles se representan en forma de punto.

TSAPs locales y remotos

Los TSAPs locales y remotos pueden ser idénticos, ya que la conexión es unívoca debido a las distintas direcciones MAC. Si hay que configurar más de una conexión entre dos estaciones, habrá que diferenciar los TSAPs.

- Conexión especificada
  Los TSAPs de una conexión de transporte ISO deben coincidir como se indica a continuación:
  - TSAP remoto (en el CP Ethernet) = TSAP local (en la estación de destino)
  - TSAP local (en el CP Ethernet) = TSAP remoto (en la estación de destino)
- Conexión no especificada

Los parámetros de dirección quedan en parte sin especificar: Conexiones no especificadas (Página 612)

TSAPs predeterminados

Para configurar los TSAPs locales y remotos se proponen valores estándar (puede cambiarse) (p. ej. ISO-1 para la primera conexión entre dos interlocutores). Si se configuran nuevas conexiones entre los mismos interlocutores, los valores estándar se incrementan automáticamente (p. ej. ISO-2, etc.). Si se establece una nueva conexión con otro interlocutor distinto, se inicia de nuevo ISO-1.

Transporte ISO, opciones

Referencia

Configuración de conexión en el grupo de parámetros "Propiedades > General > Opciones"

Modos de operación

- SEND/RECV:
  Es el modo estándar con las FCs AG_SEND/AG_LSEND y AG_RECV/AG_LRECV.
- FETCH/WRITE:
  Si selecciona uno de los siguientes modos de operación para la conexión de transporte ISO, el acceso a las áreas de memoria del sistema en la SIMATIC S7 es posible desde los siguientes dispositivos:
  - SIMATIC S5
  - Estaciones SIMATIC PC
  - Dispositivos de terceros

  La conexión puede entonces utilizarse exclusivamente para este modo de operación. En este caso no es posible enviar o recibir mediante las instrucciones AG_SEND/AG_LSEND o AG_RECV/AG_LRECV.
  - FETCH
    Al seleccionar el modo de operación Fetch, se permite el acceso directo de lectura a los rangos de datos en la SIMATIC S7. Dicho acceso puede tener lugar por ejemplo desde una estación SIMATIC S5 o una estación de otro fabricante.
  - WRITE
    Al seleccionar el modo de operación Write, se permite el acceso directo de escritura a los rangos de datos en la SIMATIC S7. Dicho acceso puede tener lugar por ejemplo desde una estación SIMATIC S5 o una estación de otro fabricante.
- SPEED SEND/RECV:
  Al seleccionar el modo de operación SPEED SEND/RECV se permite la utilización de las instrucciones AG_SSEND y AG_SRECV.
  AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) ofrecen una transmisión optimizada (más rápida) dentro de una estación S7-400. En la documentación del dispositivo CP utilizado, consulte si el CP utilizado soporta este modo de operación y en combinación con la CPU seleccionada.

Requisitos para FETCH/WRITE: Tipo de establecimiento de la conexión

Seleccione la siguiente opción para el establecimiento de conexión en del tipo de estación en el grupo de parámetros "Propiedades > General > Propiedades especiales de la conexión":

- Estación SIMATIC S7: Establecimiento de conexión PASIVO
- Estación PC: Establecimiento activo de la conexión

Opción "Modo de direccionamiento S7" (solo para FETCH/WRITE)

En la configuración para el modo de operación FETCH / WRITE es posible seleccionar aquí el modo de direccionamiento.

- Modo de direccionamiento S7: Dirección de byte
- Modo de direccionamiento S5: Dirección de palabra

Esto permite que las aplicaciones accedan a las estaciones S5 o S7 sin adaptar las direcciones. Esto es interesante especialmente para aplicaciones S5 existentes que deben utilizarse tal como están para acceder a estaciones S7.

El modo de direccionamiento para el acceso a SIMATIC S7 (opción seleccionada) está seleccionado por defecto.

Consulte también

Configuración de conexión para FETCH/WRITE (Página 678)

Aplicación de FETCH/WRITE (Página 676)

Transporte ISO, dinámica

Referencia

Configuración de conexión en el grupo de parámetros "Propiedades > General > Dinámica"

Se visualizan los temporizadores y contadores relevantes de la conexión. Los valores predeterminados pueden aplicarse.

En caso necesario (p. ej. en acoplamientos a sistemas ajenos) es posible ajustar individualmente los temporizadores y contadores y, con ello, también el comportamiento dinámico de la conexión.

Propiedades OPC ISO

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > OPC"

Ajuste aquí las propiedades específicas para una conexión de transporte ISO utilizada por el servidor OPC.

Se pueden ajustar los parámetros siguientes:

Tiempo excedido para la petición

Si en el transcurso del tiempo de espera ajustado no es posible procesar una petición pendiente, la conexión se inicializa y la petición se acusa con un error.

Rango de valores: 0 – 99999 ms

- Ajuste predeterminado: 15000 ms

Tamaño del búfer de transmisión

Tamaño predeterminado del búfer de transmisión predeterminado. Durante el tiempo de ejecución es posible utilizar otros búferes de transmisión adicionales de una longitud cualquiera (máx. 65535 bytes).

Rango de valores:

- Ajuste predeterminado: 256 bytes

- Rango de valores: 65535 bytes como máximo

Opción "Optimización de acceso directo a servicios de variable"

Si se utiliza esta opción, el servidor OPC intenta agrupar los accesos directos a diferentes variables de modo que las peticiones que hay que transferir sean las menos posibles.

Dicha opción está activada por defecto.

Opción "Notificación inmediata al detectar una interrupción de la conexión"

Al seleccionar esta opción se determina que, en caso de que la conexión no esté disponible, la aplicación reciba inmediatamente una notificación acerca de una petición de comunicación que no pueda ejecutarse (FETCH o WRITE). La notificación tiene lugar independientemente de que funcione o no el intento actual de restablecer la conexión.

Si no se selecciona esta opción en caso de que la conexión no esté disponible, la aplicación no recibirá notificación hasta que no haya transcurrido el timeout para la conexión.

La opción aquí descrita sólo es relevante y sólo está disponible si se utilizan los dos modos de operación citados a continuación:

- La conexión está configurada como permanente.
  Las conexiones se pueden configurar como configuraciones establecidas permanentemente (en la sección "Detalles sobre el establecimiento de conexión"). De este modo, las peticiones pueden transmitirse al interlocutor con un tiempo de reacción lo más breve posible sin que sea necesario establecer primero una conexión.
  Si la conexión falla, se intentará volver a establecerla. En este estado pueden producirse retardos hasta que el aviso correspondiente se transmita a la aplicación.
- Se utilizan los tipos de petición FETCH o WRITE (posibilidad de selección en "Opciones").

Sólo con estos tipos de petición es necesaria una reacción inmediata en determinados casos de aplicación si falla la conexión.

Conexión ISO on TCP

ISO on TCP: propiedades de conexión especiales

Referencia

En el grupo de parámetros "Propiedades > General > Propiedades especiales de la conexión" se pueden definir propiedades adicionales del punto final local de la conexión.

Establecimiento activo de la conexión

Determine con esta opción si el establecimiento de la conexión debe realizarse desde este dispositivo. Esta opción está seleccionada por defecto si la dirección del interlocutor está especificada.

- ON: La conexión se establece activamente.
- OFF: El interlocutor establece la conexión.

Si ha seleccionado como interlocutor "no especificado" al crear la conexión, la opción está deseleccionada por defecto. Si se activa esta opción, es necesario especificar la dirección del interlocutor en el registro "Direcciones".

Nota

Efecto en el modo de operación

Tenga en cuenta el efecto en el modo de operación.

- S7-300 / S7-400
  Si se debe utilizar el modo de operación FETCH o WRITE (véase el grupo de parámetros "Opciones"), deberá desactivarse la opción "Establecimiento activo de la conexión".
- Servidor OPC
  Si se debe utilizar el modo de operación FETCH o WRITE (véase el grupo de parámetros "Opciones"), deberá activarse la opción "Establecimiento activo de la conexión".

Nombre

Al crear la conexión se visualiza el nombre propuesto para el punto final de la conexión, que contiene un número de conexión como sufijo.

Utilice este campo para identificar el interlocutor en caso de conexiones no especificadas.

Detalles del establecimiento OPC ISO-on-TCP

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > Detalles del establecimiento"

Opciones del establecimiento de la conexión

En esta área sólo pueden realizarse entradas si en "Propiedades especiales de la conexión" se ha seleccionado el establecimiento activo de la conexión.

Elija una de las dos alternativas siguientes:

- La conexión se mantiene establecida permanentemente
  La conexión se establece y se mantiene al arrancar la estación PC así como en caso de interrumpirse la conexión.
- La conexión se establece en caso necesario (acceso a variable)
  Aquí se indica adicionalmente durante cuánto tiempo debe mantenerse la conexión después de un acceso.

Monitorización del establecimiento de conexión

Tiempo excedido: Tiempo de espera de errores durante el establecimiento de la conexión.

Si dentro del tiempo de espera no es posible establecer la conexión, ésta se inicializará y la petición pendiente se acusará con error.

Rango de valores: 0 – 99999 ms

Valor predeterminado: 15000 ms

Nota

La monitorización del establecimiento de la conexión que se ajusta aquí está relacionada del siguiente modo con la vigilancia de tiempo que puede ajustarse en la configuración de módulos: el valor válido es el más pequeño.

En la configuración de módulos, la vigilancia de tiempo se configura en "Propiedades de interfaz Ethernet".

ISO on TCP, detalles de la conexión

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Detalles de direcciones".

Parámetros de dirección

Una conexión se especifica mediante los puntos finales local y remoto de la conexión.

- Direcciones locales:
  Dirección IP local y TSAP (Transport Service Access Point) local
- Direcciones remotas:
  Dirección IP remota y TSAP remoto

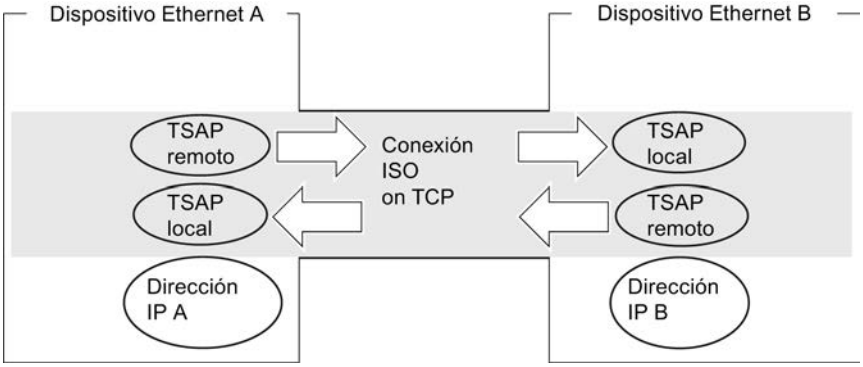


Figura 8-2 Emisión y recepción a través de una conexión ISO on TCP en cada caso

Registro Direcciones

En el registro Direcciones se muestra la información de direcciones locales y remotas relevantes en forma de valores propuestos. Tiene la posibilidad de ajustar de forma individual los TSAPs.

Formato TSAP

Las conexiones ISO on TCP tienen una longitud de TSAP de entre 1 y 16 bytes. Al realizar la entrada se muestra automáticamente la longitud actual (visualización: 16 caracteres ASCII). Los TSAPs locales y remotos pueden introducirse como valor hexadecimal o como string ASCII. En caso de una entrada ASCII, los caracteres introducidos también se visualizan en formato hexadecimal. En caso de entrada hexadecimal, los caracteres imprimibles se representan como valor ASCII (son visibles 8 caracteres hexadecimales). Si se introducen caracteres no imprimibles, la visualización ASCII queda atenuada (no es posible la entrada ASCII) y los caracteres no imprimibles se representan en forma de punto.

TSAPs locales y remotos

Los TSAPs locales y remotos pueden ser idénticos, ya que la conexión es unívoca debido a las distintas direcciones IP. Si hay que configurar más de una conexión entre dos estaciones, habrá que diferenciar también los TSAPs.

- Conexión especificada
  Para la configuración del CP Ethernet y en la estación de destino Ethernet, los TSAPs de una conexión ISO-on-TCP deben coincidir de forma cruzada:
  - TSAP remoto (en el CP Ethernet) = TSAP local (en la estación de destino)
  - TSAP local (en el CP Ethernet) = TSAP remoto (en la estación de destino)
- Conexión no especificada
  Los parámetros de dirección quedan en parte sin especificar: Conexiones no especificadas (Página 612)

TSAPs predeterminados

Para configurar los TSAPs locales y remotos hay un valor estándar "TCP-1" para la primera conexión entre ambos interlocutores (puede cambiarse). Si se establece una nueva conexión entre los mismos interlocutores, se propone el valor estándar "TCP-2". Si se establece una nueva conexión con otro interlocutor distinto, se utiliza de nuevo TCP-1.

ISO on TCP, opciones

Referencia

Configuración de conexión en el grupo de parámetros "Propiedades > General > Opciones"

Modos de operación

- SEND/RECV:
  Es el modo estándar con las FCs AG_SEND/AG_LSEND y AG_RECV/AG_LRECV.
- FETCH/WRITE:
  Si selecciona uno de los siguientes modos de operación para la conexión de transporte ISO, el acceso a las áreas de memoria del sistema en la SIMATIC S7 es posible desde los siguientes dispositivos:
  - SIMATIC S5
  - Estaciones PC SIMATIC
  - Dispositivos ajenos

  La conexión puede entonces utilizarse exclusivamente para este modo de operación. En este caso no es posible enviar o recibir mediante las instrucciones AG_SEND/AG_LSEND o AG_RECV/AG_LRECV.
  - FETCH
    Al seleccionar el modo de operación Fetch, se permite el acceso directo de lectura a los rangos de datos en la SIMATIC S7. Dicho acceso puede tener lugar por ejemplo desde una estación SIMATIC S5 o una estación de otro fabricante.
  - WRITE
    Al seleccionar el modo de operación Write, se permite el acceso directo de escritura a los rangos de datos en la SIMATIC S7. Dicho acceso puede tener lugar por ejemplo desde una estación SIMATIC S5 o una estación de otro fabricante.
- SPEED SEND/RECV:
  Al seleccionar el modo de operación SPEED SEND/RECV se permite la utilización de las instrucciones AG_SSEND y AG_SRECV.
  AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) ofrecen una transmisión optimizada (más rápida) dentro de una estación S7-400. Infórmese en la documentación del dispositivo del CP utilizado si se soporta este modo de operación.

Requisitos para FETCH/WRITE: Tipo de establecimiento de conexión

Seleccione la siguiente opción para el establecimiento de conexión en del tipo de estación en el grupo de parámetros "Propiedades > General > Propiedades especiales de la conexión":

- Estación SIMATIC S7: Establecimiento de conexión PASIVO
- Estación PC: Establecimiento activo de la conexión

Opción "Modo de direccionamiento S7" (solo para FETCH/WRITE)

En la configuración para el modo de operación FETCH / WRITE es posible seleccionar aquí el modo de direccionamiento.

- Modo de direccionamiento S7: Dirección de byte
- Modo de direccionamiento S5: Dirección de palabra

Esto permite que las aplicaciones accedan a las estaciones S5 o S7 sin adaptar las direcciones. Esto es interesante especialmente para aplicaciones S5 existentes que deben utilizarse tal como están para acceder a estaciones S7.

El modo de direccionamiento para el acceso a SIMATIC S7 (opción seleccionada) está seleccionado por defecto.

Consulte también

Configuración de conexión para FETCH/WRITE (Página 678)

Aplicación de FETCH/WRITE (Página 676)

Propiedades OPC ISO-on-TCP

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > OPC"

Ajuste aquí las propiedades específicas para una conexión ISO-on-TCP utilizada por el servidor OPC.

Se pueden ajustar los parámetros siguientes:

Tiempo excedido para la petición

Si en el transcurso del tiempo de espera ajustado no es posible procesar una petición pendiente, la conexión se inicializa y la petición se acusa con un error.

Rango de valores: 0 – 99999 ms

- Ajuste predeterminado: 15000 ms

Tamaño del búfer de transmisión

Tamaño predeterminado del búfer de transmisión predeterminado. Durante el tiempo de ejecución es posible utilizar otros búferes de transmisión adicionales de una longitud cualquiera (máx. 65535 bytes).

Rango de valores:

- Ajuste predeterminado: 256 bytes

- Rango de valores: 65535 bytes como máximo

Opción "Optimización de acceso directo a servicios de variable"

Si se utiliza esta opción, el servidor OPC intenta agrupar los accesos directos a diferentes variables de modo que las peticiones que hay que transferir sean las menos posibles.

Dicha opción está activada por defecto.

Opción "Notificación inmediata al detectar una interrupción de la conexión"

Al seleccionar esta opción se determina que, en caso de que la conexión no esté disponible, la aplicación reciba inmediatamente una notificación acerca de una petición de comunicación

que no pueda ejecutarse (FETCH o WRITE). La notificación tiene lugar independientemente de que funcione o no el intento actual de restablecer la conexión.

Si no se selecciona esta opción en caso de que la conexión no esté disponible, la aplicación no recibirá notificación hasta que no haya transcurrido el timeout para la conexión.

La opción aquí descrita sólo es relevante y sólo está disponible si se utilizan los dos modos de operación citados a continuación:

- La conexión está configurada como permanente.
  Las conexiones se pueden configurar como configuraciones establecidas permanentemente (en el apartado "Detalles sobre el establecimiento de la conexión"). De este modo, las peticiones pueden transmitirse al interlocutor con un tiempo de reacción lo más breve posible sin que sea necesario establecer primero una conexión.
  Si la conexión falla, se intentará volver a establecerla. En este estado pueden producirse retardos hasta que el aviso correspondiente se transmita a la aplicación.
- Se utilizan los tipos de petición FETCH o WRITE (posibilidad de selección en "Opciones"). Sólo con estos tipos de petición es necesaria una reacción inmediata en determinados casos de aplicación si falla la conexión.

Conexión TCP

TCP: Propiedades especiales de la conexión

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Propiedades especiales de la conexión"

Los parámetros de conexión definen propiedades adicionales del punto final local de la conexión.

Establecer conexión activa

Determine con esta opción si el establecimiento de la conexión debe realizarse desde este dispositivo. Esta opción está seleccionada por defecto si la dirección del interlocutor está especificada.

- ON: La conexión se establece activamente.
- OFF: El interlocutor establece la conexión.

Si ha seleccionado como interlocutor "no especificado" al crear la conexión, la opción está deseleccionada por defecto. Si se activa esta opción, es necesario especificar la dirección del interlocutor en el registro "Direcciones".

Nota

Efecto en el modo de operación

Tenga en cuenta el efecto en el modo de operación.

- S7-300 / S7-400
  Si se debe utilizar el modo de operación FETCH o WRITE (véase el grupo de parámetros "Opciones"), deberá desactivarse la opción "Establecimiento activo de la conexión".
- Servidor OPC
  Si se debe utilizar el modo de operación FETCH o WRITE (véase el grupo de parámetros "Opciones"), deberá activarse la opción "Establecimiento activo de la conexión".

Modo FTP (en el S7-300 / S7-400 - conexión no especificada)

Si se selecciona la opción, las repercusiones son las siguientes:

- La conexión TCP se utiliza ahora como conexión FTP.
- Grupo de parámetros "Detalles de direcciones": Las direcciones se especifican automáticamente (puerto = 21; observación: este ajuste no resulta visible en "Detalles de direcciones").
- Grupo de parámetros "Opciones": El modo de operación está ajustado a FTP de forma fija.

La opción "'Establecimiento activo de la conexión" es irrelevante aquí y no está operativa.

Requisitos: Esta opción sólo se puede seleccionar con una conexión TCP no especificada para determinados CPs del S7-300 / S7-400.

Detalles del establecimiento OPC TCP

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > Detalles del establecimiento"

Opciones del establecimiento de la conexión

En esta área sólo pueden realizarse entradas si en "Propiedades especiales de la conexión" se ha seleccionado el establecimiento activo de la conexión.

Elija una de las dos alternativas siguientes:

- La conexión se mantiene establecida permanentemente
  La conexión se establece y se mantiene al arrancar la estación PC así como en caso de interrumpirse la conexión.
- La conexión se establece en caso necesario (acceso a variable)
  Aquí se indica adicionalmente durante cuánto tiempo debe mantenerse la conexión después de un acceso.

Monitorización del establecimiento de conexión

- Timeout: Tiempo de espera de errores durante el establecimiento de la conexión.
  Si dentro del tiempo de espera no es posible establecer la conexión, ésta se inicializará y la petición pendiente se acusará con error.
  Rango de valores: 0 – 99999 ms
  Valor predeterminado: 15000 ms

Nota

La monitorización del establecimiento de la conexión que se ajusta aquí está relacionada del siguiente modo con la vigilancia de tiempo que puede ajustarse en la configuración de módulos: el valor válido es el más pequeño.

En la configuración de módulos, la vigilancia de tiempo se configura en "Propiedades de interfaz Ethernet".

TCP, detalles de dirección

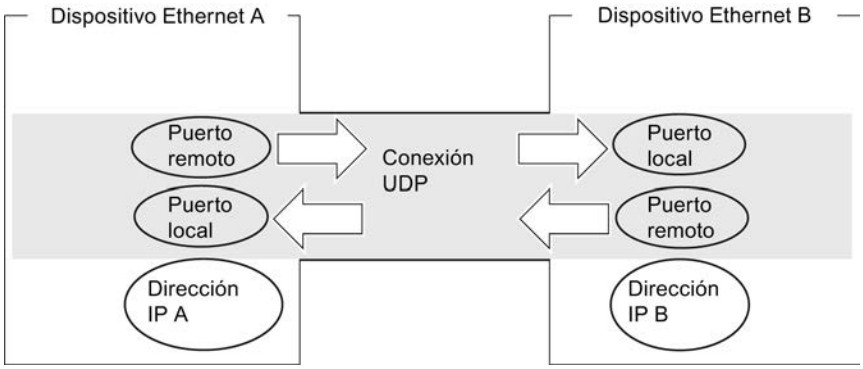
Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Detalles de direcciones".

Parámetros de dirección y tipos de conexión

Para TCP los interlocutores se direccionan mediante el punto final local y remoto como se indica a continuación.

- Direcciones locales: Dirección IP local y puerto local
- Direcciones remotas: Dirección IP remota y puerto remoto



Puertos

Los puertos o las direcciones de los puertos definen el punto de acceso al programa de usuario dentro de la estación/CPU. ¡Estos deben ser unívocos dentro de la estación/CPU!

- Conexión especificada
  Para la configuración del CP Ethernet y en la estación de destino Ethernet, los puertos de una conexión TCP deben coincidir de forma cruzada:
  - puerto remoto (en el CP Ethernet) = puerto local (en la estación de destino)
  - puerto local (en el CP Ethernet) = puerto remoto (en la estación de destino)
- Conexión no especificada
  Los parámetros de dirección quedan en parte sin especificar: Conexiones no especificadas (Página 612)

Respecto al rango de valores del puerto, véase también

Configuración de puerto (Página 667)

TCP, opciones

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Opciones"

Modo de operación

- SEND/RECV:
  Es el modo estándar con las FCs AG_SEND/AG_LSEND y AG_RECV/AG_LRECV.
- FETCH/WRITE:
  Si selecciona uno de los siguientes modos de operación para la conexión TCP, el acceso a las áreas de memoria del sistema en la SIMATIC S7 es posible desde los siguientes dispositivos:
  - SIMATIC S5
  - Estaciones SIMATIC PC
  - Dispositivos de terceros

  La conexión puede entonces utilizarse exclusivamente para este modo de operación. En este caso no es posible enviar o recibir mediante las FCs AG_SEND/AG_LSEND o AG_RECV/AG_LRECV.
  - FETCH
    Seleccionando este modo de operación, se permite el acceso directo en lectura a las áreas de datos en la SIMATIC S7. Dicho acceso puede tener lugar por ejemplo desde una estación SIMATIC S5 o una estación de otro fabricante.
  - WRITE
    Al seleccionar el modo de operación, se permite el acceso directo en escritura a las áreas de datos en el SIMATIC S7. Dicho acceso puede tener lugar por ejemplo desde una estación SIMATIC S5 o una estación de otro fabricante.
- FTP
  Si en "Propiedades > General > Propiedades especiales de la conexión" se ha seleccionado la opción "Utilizar para protocolo FTP", aquí se muestra "FTP"; en este caso, la conexión TCP se utiliza como "conexión FTP" y no existe ninguna otra posibilidad de selección.
  Requisitos: Sólo es posible seleccionar la opción para una conexión TCP no especificada.
- SPEED SEND/RECV
  Al seleccionar el modo de operación SPEED SEND/RECV se permite la utilización de las instrucciones AG_SSEND y AG_SRECV.
  Las instrucciones AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) ofrecen una transmisión optimizada (más rápida) dentro de una estación S7-400. Infórmese en la documentación del dispositivo CP utilizado si se soporta este modo de operación.

Requisitos para FETCH/WRITE: Tipo de establecimiento de la conexión

Dependiendo del tipo de estación, seleccione la siguiente opción para el establecimiento de la conexión en el registro "General":

- Estación SIMATIC S7: Establecimiento PASIVO de la conexión
- Estación PC: Establecimiento activo de la conexión

Opción "Modo de direccionamiento S7" (solo para FETCH/WRITE)

En la configuración para el modo de operación FETCH / WRITE es posible seleccionar aquí el modo de direccionamiento:

- Modo de direccionamiento S7: Dirección de byte
- Modo de direccionamiento S5: Dirección de palabra

Esto permite que las aplicaciones accedan a las estaciones S5 o S7 sin adaptar las direcciones. Esto es interesante especialmente para aplicaciones S5 existentes que deben utilizarse tal como están para acceder a estaciones S7.

El modo de direccionamiento para el acceso a SIMATIC S7 (opción seleccionada) está seleccionado por defecto.

Consulte también

Configuración de conexión para FETCH/WRITE (Página 678)

Aplicación de FETCH/WRITE (Página 676)

Propiedades OPC TCP

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > OPC"

Ajuste aquí las propiedades específicas para una conexión TCP utilizada por el servidor OPC.

Se pueden ajustar los parámetros siguientes:

Timeout de petición

Si en el transcurso del tiempo de espera aquí ajustado no es posible procesar una petición pendiente, la conexión se inicializa y la petición se acusa con un error.

Rango de valores:

- Ajuste predeterminado: 15000 ms

Tamaño del búfer de transmisión

Tamaño predeterminado del búfer de transmisión predeterminado. Durante el tiempo de ejecución es posible utilizar otros búferes de transmisión adicionales de una longitud cualquiera (máx. 65535 bytes).

Rango de valores:

- Ajuste predeterminado: 256 bytes
- Rango de valores: 65535 bytes como máximo

Optimización de acceso directo a servicios de variable

Si se utiliza esta opción, el servidor OPC intenta agrupar los accesos directos a diferentes variables de modo que las peticiones que hay que transferir sean las menos posibles.

Dicha opción está activada por defecto.

Utilizar el miniprotocolo

Con esta opción se determina si los datos deben transferirse en TCP como bloques individuales con una indicación cada uno acerca del tamaño del mismo (2 bytes en el búfer de peticiones). En este caso, el receptor lee los datos de acuerdo con la indicación de longitud. Esta propiedad debe configurarse de forma idéntica para ambos interlocutores.

Dicha opción está activada por defecto; el receptor lee los datos de acuerdo con el tamaño ajustado para el búfer de recepción.

- Tamaño del búfer de recepción
  Si no se utiliza ningún miniprotocolo, aquí es posible ajustar el tamaño del búfer de recepción.
  Rango de valores: 0 - 65535 bytes
  - Ajuste predeterminado: 256 bytes

Configuración de puerto

Configuración de puerto

Los puertos o las direcciones de los puertos definen el punto de acceso al programa de usuario dentro de la estación/CPU. ¡Estos deben ser unívocos dentro de la estación/CPU!

La siguiente tabla aporta información sobre el rango de valores:

Direcciones de puerto		Aplicación/nota
0		Asignado de forma fija; ¡no puede utilizarse!
1..1023		Asignado de forma estándar; no se debería utilizar (well known ports)
1024..49151	Puertos para protocolos específicos de aplicación	
	a partir de 2000...5000	Rango en el que la herramienta de configuración busca o asigna una dirección de puerto libre en cada caso. Es posible ajustar la dirección de puerto en este rango de forma individual.
	5001..49151	¡El sistema utiliza las direcciones de puerto a partir de 5000! Observación: Si desea utilizar estas direcciones de puerto, contacte en caso dado a su administrador del sistema para que compruebe su disponibilidad.
49152..65535		Puertos asignados de forma dinámica Se recomienda no utilizar estos puertos.

Los siguientes números locales de puerto han sido reservados, no deben ser utilizados para otra finalidad durante la configuración de la conexión.

Número de puerto	Protocolo	Servicio
20, 21	TCP	FTP
25	TCP	SMTP
80	TCP	HTTP
102	TCP	RFC1006
135	TCP	RPC-DCOM
161	UDP	SNMP_REQUEST
34964	UDP	PN IO
65532	UDP	NTP
65533	UDP	NTP
65534	UDP	NTP
65535	UDP	NTP

Conexión UDP

UDP, detalles de dirección

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Detalles de direcciones".

Parámetros de dirección y tipos de conexión

Para UDP los interlocutores se direccionan mediante el punto final local y remoto como se indica a continuación.

- Direcciones locales:
  Dirección IP local y puerto local
- Direcciones remotas:
  Dirección IP remota y puerto remoto

Nota

El término "Conexión" se utiliza aquí también para UDP, ya que para la configuración (así como p. ej. también para TCP), los interlocutores son asignados entre sí y, por tanto, quedan "conectados" de forma lógica. En realidad, en la operación de las estaciones no tiene lugar ningún establecimiento de conexión explícito entre los interlocutores para UDP.

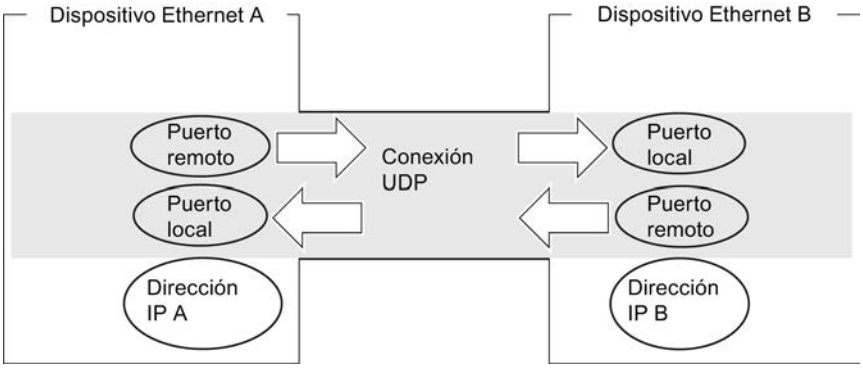


Figura 8-3 Emisión y recepción a través de una conexión UDP

Dependiendo del tipo de conexión deseado se especifican o se dejan abiertos los parámetros de dirección remotos durante la configuración.

- Conexión UDP especificada
  Se ha indicado una estación de destino al crear una nueva conexión.
- Conexión UDP especificada como Broadcast
  Se ha indicado "Broadcast" como interlocutor al crear una nueva conexión (la conexión se debe crear primeramente como conexión no especificada a través de cuadros de diálogo o gráficamente).
- Conexión UDP especificada como Multicast
  Se ha indicado "Multicast" como interlocutor al crear una nueva conexión (la conexión se debe crear primeramente como conexión no especificada a través de cuadros de diálogo o gráficamente).
- Conexión UDP no especificada
  Al crear la conexión ha seleccionado el tipo de conexión "sin especificar".

Conexión UDP no especificada

La conexión UDP no especificada puede utilizarse de dos formas:

- Conexión UDP libre
  Para configurar una conexión UDP libre seleccione la opción "Asignación de dirección en el bloque". Ya no es posible introducir los campos de entrada de la dirección IP remota y el puerto remoto, puesto que ahora el programa de usuario determina las direcciones de destino.
- Conexión con una "estación ajena" de otro proyecto
  Es posible indicar la dirección IP remota y el puerto para una estación de destino cualquiera. La estación de destino puede encontrarse dentro o fuera del proyecto actual.

Nota

Es preciso especificar las direcciones de interlocutor

Debido a que en UDP no tiene lugar ningún establecimiento de conexión (servicio de datagramas), la comunicación a través de la conexión UDP configurada solo es posible si también se han especificado las direcciones de interlocutor (dirección IP y puerto).

Consulte también

Conexiones no especificadas (Página 612)

Conexión con todos los dispositivos Broadcast

Al seleccionar como interlocutores "todos los dispositivos Broadcast" se determina que se envíen telegramas UDP a todos los dispositivos Broadcast accesibles.

En el grupo de parámetros "Detalles de direcciones" se propone una dirección Broadcast válida para la red en la dirección IP (IP) para el interlocutor.

En PUERTO se debe introducir una dirección de PUERTO adecuada para todos los interlocutores a los que se desea acceder.

Configuración de puerto, véase también

Configuración de puerto (Página 675)

UDP con Broadcast y Multicast

Aplicación

Para la selección del interlocutor, en las conexiones UDP se dispone de dos opciones adicionales:

- Conexión con todos los dispositivos Broadcast
  Al seleccionar como interlocutores "Todos los dispositivos Broadcast" se determina que serán enviados telegramas UDP a todos los dispositivos Broadcast accesibles.

  Nota

  En el caso de Broadcast, a través de los CPs S7 únicamente se puede enviar; un CP S7 no permite recibir datos de usuario a través de las conexiones Broadcast (véase más abajo)

- Conexión con todos los dispositivos Multicast
  Al seleccionar como interlocutores "Todos los dispositivos Multicast" se determina que serán enviados telegramas UDP a todos los dispositivos de un grupo Multicast y que es posible recibir telegramas Multicast.
  Multicast y Broadcast son opciones de conexión especiales que son soportadas y configurables por CPs Industrial Ethernet en conexiones UDP.

Los telegramas se envían sin acuse ya que el protocolo UDP no prevé confirmaciones. Esto ha sido previsto así para evitar "desbordamientos de datos" debidos a acuses. Si, por ejemplo, se envían telegramas a 100 interlocutores, se recibirán 100 acuses (1 por interlocutor) simultáneamente. El módulo emisor no es capaz de procesar tales caudales de datos.

Consulte también

Crear conexiones UDP/FDL con Multicast/Broadcast (Página 639)

¿Cuándo es preferible usar Multicast en lugar de Broadcast?

La opción de conexión "Multicast" para conexiones UDP ha sido añadida para permitir la transmisión simultánea de un telegrama a un gran número de interlocutores.

Al contrario que en la opción de conexión Broadcast, en este tipo de conexión también es posible la recepción de telegramas enviados a varios dispositivos en el círculo Multicast.

Mediante la orientación a un grupo de destinatarios determinado (círculo Multicast) se puede evitar la carga de los destinatarios no direccionados. Por esta razón, Multicast representa en cualquier caso una solución mejor que Broadcast cuando se desea enviar telegramas a grupos de estaciones interlocutoras.

¿Por qué un CP S7 no permite la recepción de datos de usuario en las conexiones Broadcast?

Los CPs S7 no reciben telegramas Broadcast de transferencia de datos de usuario. Solo los telegramas utilizables, p. ej. solicitud ARP, serán transferidos más allá del controlador LAN y evaluados. No obstante, con los CPs S7 se pueden enviar telegramas Broadcast a la red.

Motivo:

A menudo se desea enviar telegramas desde un dispositivo a múltiples dispositivos interlocutores. En este caso es importante que los telegramas sean enviados simultáneamente y que también lleguen a los dispositivos interlocutores de forma casi simultánea. Por esta razón se requiere siempre un envío y recepción de telegramas Broadcast. En el caso de un mensaje Broadcast, el telegrama es aceptado realmente por todos los dispositivos de la red.

Un caso de aplicación típico es la necesidad de telegramas Broadcast para la búsqueda de una dirección MAC relacionada con una dirección IP (solicitud ARP).

Por esta razón, un módulo de comunicación debe aceptar en general telegramas Broadcast y evaluarlos mediante software. En este caso, la gran desventaja consiste en que un número excesivo de telegramas Broadcast en la red reducen drásticamente el rendimiento. Esto es debido a que cada uno de los módulos debe procesar todos los telegramas Broadcast a fin de determinar si son destinatarios de los mismos.

Para evitar estas desventajas, el comportamiento de los CPs S7 en referencia al Broadcast es el siguiente:

- Tras la recepción, se filtran los telegramas Broadcast en todos los CPs Ethernet con la máxima prioridad. Esto significa que todos los telegramas que no sean utilizables serán directamente eliminados. Solo los telegramas utilizables, p. ej. solicitud ARP, serán transferidos más allá del controlador LAN y evaluados. De esta forma se evita un posible efecto negativo de los telegramas Broadcast sobre el resto de las conexiones.

Grupo de parámetros "Detalles de direcciones": conexión con todos los dispositivos Multicast

Al seleccionar como interlocutores "Todos los dispositivos Multicast" se determina lo siguiente:

- Los telegramas UDP enviados son entregados a todos los dispositivos Multicast accesibles del grupo Multicast;
- El dispositivo local para telegramas Multicast está preparado para la recepción en el grupo Multicast indicado.

El círculo Multicast se determina mediante la dirección IP y las direcciones de los puertos.

En la dirección IP (IP) para el interlocutor se propone una dirección IP válida en la red para conjuntos Multicast. En cada caso, el interlocutor para Multicast es un grupo de destinatarios (grupo Multicast).

Para todos los interlocutores a los que se desea llegar es necesario introducir en PUERTO una dirección de PUERTO. Debe ser adecuada para todos los interlocutores.

En principio es posible dirigirse a varios conjuntos Multicast bajo una dirección IP. Para ello es posible crear varias conexiones UDP con la misma dirección IP pero diferentes direcciones de PUERTO.

Nota

Es preciso asignar direcciones de puerto idénticas para el puerto local y el puerto del interlocutor dentro de un círculo Multicast. ¡Solo de esta manera es posible enviar y recibir telegramas desde el CP dentro de un círculo Multicast!

Tenga en cuenta el siguiente ejemplo para tres dispositivos en el círculo Multicast:

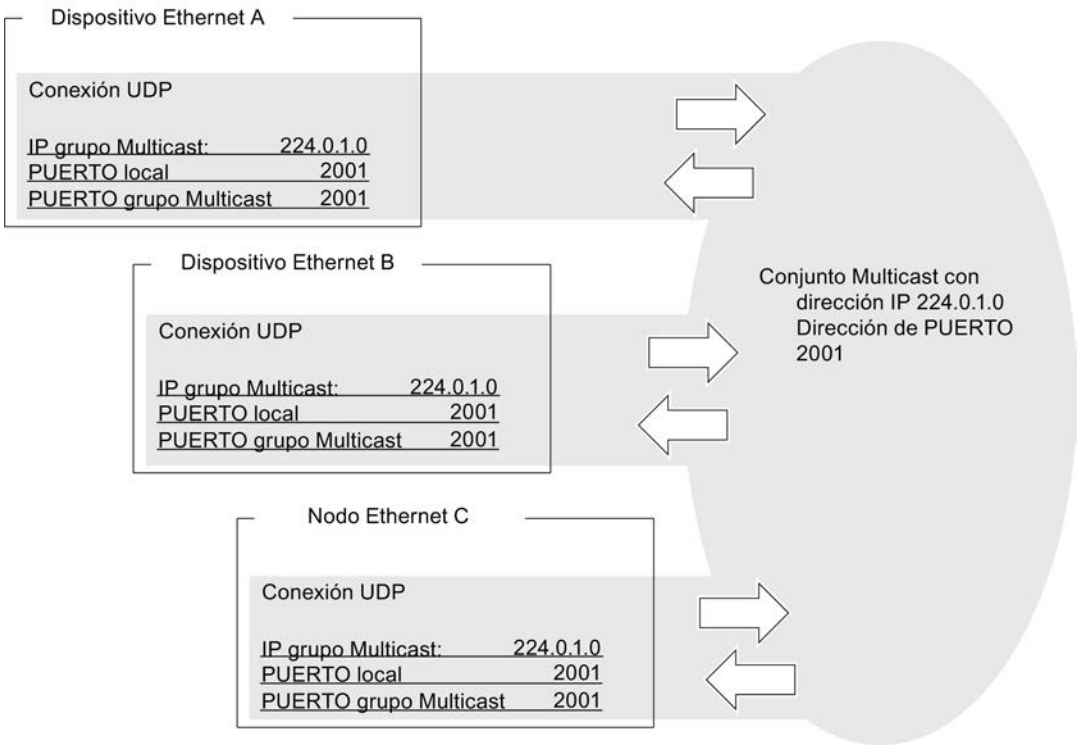
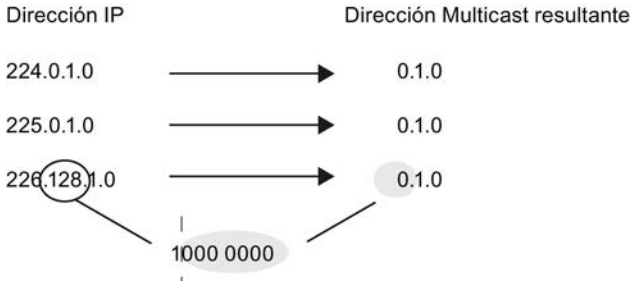


Figura 8-4 Envío y recepción en un círculo Multicast a través de direcciones de puerto idénticas

Direcciones IP para Multicast IP

- Rango de valores
  Para Multicast IP pueden utilizarse las direcciones IP desde 224.0.0.0 hasta 239.255.255.255.
  Debido a que las direcciones IP hasta 224.0.0.255 han sido reservadas para fines especiales, se recomienda utilizar para Multicast IP las direcciones IP a partir de 224.0.1.0 (ajuste por defecto).
- Identificación del círculo Multicast
  La identificación de un círculo Multicast no se realiza mediante la dirección IP completa; en realidad se ignoran el primer byte de dirección y el bit de mayor peso del segundo byte de dirección. Esta es una particularidad importante ya que de esta forma es posible direccionar direcciones IP aparentemente diferentes del mismo círculo Multicast.
  Ejemplo:
  Las siguientes direcciones IP direccionan en cada caso el mismo círculo Multicast.



Conexión UDP libre

Direccionamiento controlado por el programa

Una conexión UDP libre permite direccionar el interlocutor de forma controlada por el programa. La comunicación entre dos dispositivos en Industrial Ethernet tiene las siguientes propiedades:

- La transferencia de datos es bidireccional, es decir, en la conexión UDP es posible enviar y recibir de forma simultánea.
- El dispositivo local ha sido fijado mediante configuración. El programa de usuario registra el dispositivo remoto en el encabezado de la petición del búfer de peticiones en la llamada de AG_SEND. De esta forma es posible acceder a cada dispositivo de Ethernet/LAN/WAN.
- En el encabezado de la petición de AG_RECV es posible leer la dirección IP y el puerto del emisor.

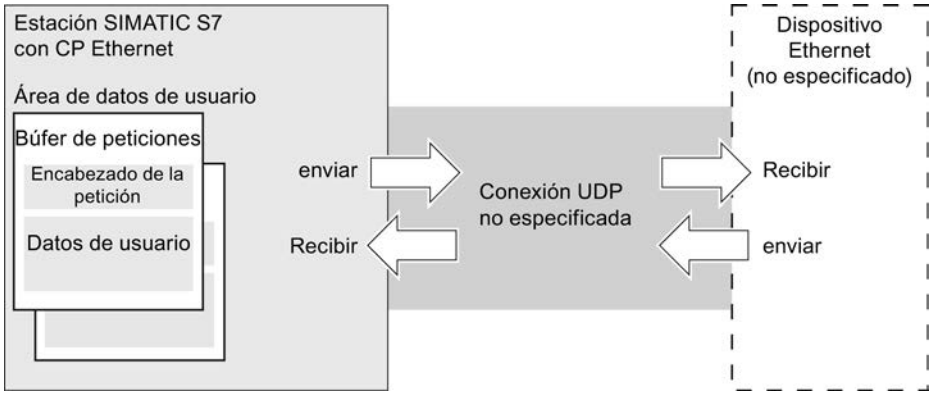


Figura 8-5 Emisión y recepción a través de una conexión UDP no especificada: direccionamiento por programa

Volumen de datos y equipamiento

Consulte cuántas conexiones UDP soporta cada CP Ethernet en el manual del dispositivo NO TAG que acompaña al CP Ethernet. Mediante la aceptación de otros CPs es posible aumentar el número de conexiones por estación.

Por cada búfer de peticiones es posible transmitir hasta 2042 bytes de datos de usuario. El encabezado de la petición ocupa adicionalmente 6 bytes.

Opciones UDP

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Opciones"

Modo de operación

- **SEND/RECV:**
  Es el modo estándar con las FCs AG_SEND/AG_LSEND y AG_RECV/AG_LRECV.
- **SPEED SEND/RECV**
  Al seleccionar el modo de operación SPEED SEND/RECV se permite la utilización de las instrucciones AG_SSEND y AG_SRECV.
  Las instrucciones AG_SSEND/AG_SRECV (SPEED SEND/RECEIVE) ofrecen una transmisión optimizada (más rápida) dentro de una estación S7-400. Infórmese en la documentación del dispositivo CP utilizado si se soporta este modo de operación.

Configuración de puerto

Configuración de puerto

Los puertos o las direcciones de los puertos definen el punto de acceso al programa de usuario dentro de la estación/CPU. ¡Estos deben ser unívocos dentro de la estación/CPU!

La siguiente tabla aporta información sobre el rango de valores:

Direcciones de puerto		Aplicación/nota
0		Asignado de forma fija; ¡no puede utilizarse!
1..1023		Asignado de forma estándar; no se debería utilizar (well known ports)
1024..49151	Puertos para protocolos específicos de aplicación	
	a partir de 2000...5000	Rango en el que la herramienta de configuración busca o asigna una dirección de puerto libre en cada caso. Es posible ajustar la dirección de puerto en este rango de forma individual.
	5001..49151	¡El sistema utiliza las direcciones de puerto a partir de 5000! Observación: Si desea utilizar estas direcciones de puerto, contacte en caso dado a su administrador del sistema para que compruebe su disponibilidad.
49152..65535		Puertos asignados de forma dinámica Se recomienda no utilizar estos puertos.

Los siguientes números locales de puerto han sido reservados, no deben ser utilizados para otra finalidad durante la configuración de la conexión.

Número de puerto	Protocolo	Servicio
20, 21	TCP	FTP
25	TCP	SMTP
80	TCP	HTTP
102	TCP	RFC1006
135	TCP	RPC-DCOM
161	UDP	SNMP_REQUEST

Número de puerto	Protocolo	Servicio
34964	UDP	PN IO
65532	UDP	NTP
65533	UDP	NTP
65534	UDP	NTP
65535	UDP	NTP

Modo de operación FETCH/WRITE

Aplicación de FETCH/WRITE

FETCH/WRITE

Los servicios FETCH/WRITE permiten acceder a áreas de memoria del sistema de la CPU S7.

- FETCH: Leer datos directamente
- WRITE: Escribir datos directamente

El acceso es posible desde los siguientes dispositivos:

- SIMATIC S5
- Estaciones PC SIMATIC
- Dispositivos ajenos

Tipos de conexión

Los servicios FETCH/WRITE pueden configurarse y utilizarse con SIMATIC S7 en los siguientes tipos de conexión:

- Conexiones de transporte ISO
- Conexiones ISO on TCP
- Conexiones TCP

Memoria de sistema

Mediante FETCH o WRITE es posible acceder a las siguientes áreas de operandos en la memoria de sistema de SIMATIC S7:

- Bloques de datos (DB)
  (tenga en cuenta la siguiente limitación para el acceso a DB: el número máximo de DB es 255)
- Marcas (M)
- Memoria imagen de proceso de las entradas (PAE)
- Memoria imagen de proceso de las salidas (PAA)

- Área de periferia de las entradas (PEW, PED, PEB)
- Área de periferia de las salidas (PAW, PAD, PAB)
- Contadores (Z)
- Temporizadores (T)

Acoplamiento con sistemas ajenos

FETCH y WRITE pueden utilizarse en principio desde cualquier dispositivo ajeno para el acceso a áreas de memoria del sistema S7.

Para que sea posible implementar este acceso también p. ej. para aplicaciones de PC, es necesario conocer la estructura de PDU para las peticiones. Tenga en cuenta para ello la documentación del dispositivo.

Mensajes en el búfer de diagnóstico

Debido a los accesos FETCH/WRITE pueden darse confirmaciones negativas por parte de la CPU S7. Esto lleva a las correspondientes entradas orientadas a la conexión en el búfer de diagnóstico.

Tabla 8-4 Codificación de mensajes en el búfer de diagnóstico en FETCH/WRITE

Codificación	Significado
01_H	Error de hardware
03_H	No se permite el acceso al objeto.
05_H	Dirección no válida (ID de sintaxis, rango, tipo, número de bit)
06_H	No se soporta el tipo de datos.
07_H	El tipo de datos no es coherente.
$0A_H$	El objeto no existe o se ha superado el final del área.
FF_H	Error interno de protocolo

Información adicional

La estructura de PDU para el acceso desde sistemas ajenos se describe en la documentación de dispositivos para CPs S7.

Configuración de conexión para FETCH/WRITE

Configurar el modo de operación en función del tipo de dispositivo

Según el tipo de estación es posible configurar los siguientes modos de operación para el punto final de la conexión:

- Estación SIMATIC S7: FETCH PASIVO / WRITE PASIVO
  Seleccionando uno de los modos de operación FETCH PASIVO o WRITE PASIVO para la conexión, es posible realizar accesos directos a las áreas de memoria del sistema en la SIMATIC S7 desde una estación SIMATIC S5 o una estación ajena (conexión no especificada).
  La conexión puede entonces utilizarse exclusivamente para este modo de operación. En este caso no es posible enviar o recibir mediante las FCs AG_SEND/AG_LSEND o AG_RECV/AG_LRECV.
  El establecimiento de conexión tiene lugar de forma pasiva, es decir, solo la estación interlocutora (estación SIMATIC S5, estación PC o una estación ajena) puede establecer la conexión. El ajuste correspondiente se realiza en "Propiedades > Opciones". Después de seleccionar el modo de operación ya no es posible modificar el tipo de establecimiento de conexión.
- Estación PC SIMATIC: FETCH ACTIVO / WRITE ACTIVO
  Seleccionando uno de los modos de operación FETCH ACTIVO o WRITE ACTIVO se pueden efectuar accesos directos a las áreas de memoria del sistema en la estación SIMATIC S7 o SIMATIC S5 desde la estación PC.
  El establecimiento de conexión tiene lugar de forma activa, es decir, la estación interlocutora debe esperar al establecimiento de conexión (establecimiento de conexión pasivo en el interlocutor). El ajuste correspondiente se realiza en "Propiedades > Opciones". Después de seleccionar el modo de operación ya no es posible modificar el tipo de establecimiento de conexión.

Opción "Modo de direccionamiento S7"

En la configuración para el modo de operación FETCH ACTIVO/WRITE ACTIVO es posible seleccionar el modo de direccionamiento. De esta manera se determina la forma en la que debe interpretarse la indicación de la dirección en la llamada FETCH/WRITE en la estación SIMATIC S7 para el acceso a DBs:

- Modo de direccionamiento S7: Dirección de byte
- Modo de direccionamiento S5: Dirección de palabra

Esto permite que las aplicaciones accedan a las estaciones S5 o S7 sin adaptar las direcciones. Esto es interesante especialmente para aplicaciones S5 existentes que deben utilizarse tal como están para acceder a estaciones S7.

El modo de direccionamiento para el acceso a SIMATIC S7 (opción seleccionada) está seleccionado por defecto.

Conexión FDL

Conexiones FDL: sinopsis

Aplicación

La transferencia de datos a través de una conexión FDL configurada es adecuada para la transmisión de bloques de datos relacionados entre dos o más dispositivos PROFIBUS.

Debe distinguirse entre

- Conexión FDL especificada
  Los dispositivos interlocutores se han definido de forma unívoca mediante la configuración de la conexión.
  El interlocutor puede encontrarse dentro o fuera del proyecto STEP7.
- Conexión FDL no especificada (acceso libre Layer 2)
  La dirección del interlocutor permanece abierta durante la configuración. Los dispositivos interlocutores se determinan mediante la indicación de direcciones en la petición de comunicación del programa de usuario. De esta forma es posible acceder a hasta 126 dispositivos a través de una conexión FDL configurada no especificada, en tanto dichos dispositivos soporten conexiones FDL.
  El interlocutor puede encontrarse dentro o fuera del proyecto.
- Conexión FDL con Broadcast
  Se accede a todos los dispositivos de PROFIBUS preparados para la recepción en Broadcast.
- Conexión FDL con Multicast
  Se accede a todos los dispositivos de PROFIBUS que pertenecen al conjunto Multicast.

Nota

Requisitos para la configuración

Si se utilizan conexiones FDL, **no** se permite ajustar el modo de operación CP del CP de PROFIBUS como **esclavo DP pasivo**.

Nota

Prioridad de los telegramas

Tenga en cuenta que los CPs de PROFIBUS para SIMATIC S7 envían los telegramas con la prioridad "LOW".

Las estaciones interlocutoras (SIMATIC S5, S7 o estaciones ajenas) deben utilizar asimismo la prioridad "LOW", ya que de lo contrario no tiene lugar ningún establecimiento de conexión.

FDL, detalles de dirección

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Detalles de direcciones".

Parámetros de dirección en conexiones FDL

Una conexión FDL se especifica mediante los puntos finales local y remoto de la conexión. Esto incluye:

- Dirección PROFIBUS del dispositivo al que se desea acceder.
- LSAP (Link ServiceAccessPoint) local:
  El LSAP local controla la disponibilidad para la recepción del CP PROFIBUS. Para el LSAP se proporcionan en el CP de PROFIBUS los recursos de recepción para la recepción de datos en la conexión FDL.
- LSAP (Link ServiceAccessPoint) remoto:
  El LSAP remoto controla el servicio de envío en el CP PROFIBUS. El CP PROFIBUS envía a través del LSAP al dispositivo a través de la conexión FDL. El dispositivo de destino debe estar preparado para la recepción desde este SAP.

Conexión FDL especificada

En una conexión FDL especificada se muestra la información de direcciones locales y remotas relevantes en forma de valores propuestos. En caso necesario, tiene la posibilidad de ajustar de forma individual las direcciones LSAP.

Conexión FDL no especificada

La conexión FDL no especificada puede utilizarse de dos formas:

- Conexión con una "estación ajena" de otro proyecto
  Es posible indicar la dirección PROFIBUS remota y el LSAP para una estación de destino cualquiera. La estación de destino puede encontrarse dentro o fuera del proyecto STEP 7 actual.
  En tanto no se haya especificado la dirección remota, no es posible la comunicación a través de la conexión FDL.
- Acceso libre Layer -2
  Para configurar un acceso libre Layer 2, seleccione la casilla de verificación correspondiente. Ya no es posible introducir los campos de entrada de la dirección PROFIBUS remota y el LSAP remoto, puesto que ahora el programa de usuario determina las direcciones de destino.

Conexión FDL con Broadcast

En la conexión FDL con Broadcast, los parámetros de dirección remota están ajustados de forma fija. Es posible acceder a todos los dispositivos Broadcast a través de la dirección

PROFIBUS 127. Todos los dispositivos Broadcast reciben los datos de recepción a través del LSAP 63.

Durante el envío, los parámetros de dirección local se incluyen en el encabezado de la petición del mensaje y se envían al destinatario. El programa de usuario del interlocutor remoto puede determinar quién es el emisor del telegrama Broadcast mediante dichos parámetros.

Nota

El CP recibe solo en una conexión Broadcast

Si se utiliza una conexión FDL con Broadcast, no es posible utilizar ninguna otra conexión Broadcast en el CP en cuestión.

⚠ PRECAUCIÓN

Función "Activar difusión cíclica de los parámetros de bus"

Si se ha activado esta opción para la subred en "General > Parámetros de bus", los parámetros de bus se envían cíclicamente como telegramas Broadcast durante el funcionamiento. A fin de evitar conflictos durante la manipulación de los telegramas en el programa de usuario encargado de recibir telegramas Broadcast, proceda del siguiente modo:

- Ignore todos los telegramas enviados con un LSAP >56;
  o
- Desactive la opción mencionada en "General > Parámetros de bus".

Conexión FDL con Multicast

En la conexión FDL con Multicast, los parámetros de dirección remota están ajustados de forma fija. Es posible acceder a todos los dispositivos Multicast a través de la dirección PROFIBUS 127. Todos los dispositivos del círculo Multicast envían y reciben datos a través del mismo LSAP (rango de 1 a 56). Por esta razón, el valor del LSAP solo se puede seleccionar de forma local y se acepta de forma automática en el LSAP remoto.

Durante el envío, los parámetros de dirección local se incluyen en el encabezado de la petición del mensaje y se envían al destinatario. El programa de usuario del interlocutor remoto puede determinar quién es el emisor del telegrama Multicast mediante dichos parámetros.

Opciones FDL

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Opciones"

Modo de operación

- SEND/RECV:
  Es el modo estándar con las FCs AG_SEND/AG_LSEND y AG_RECV/AG_LRECV. En las conexiones FDL, este modo de operación es el predeterminado y no se puede modificar.

Propiedades OPC FDL

Referencia

Configuración de conexiones en el caso de OPC en el grupo de parámetros "Propiedades > General > OPC"

Ajuste aquí las propiedades específicas para una conexión FDL utilizada por el servidor OPC.

Se pueden ajustar los parámetros siguientes:

Tiempo de espera de errores

Si dentro del tiempo de espera no es posible procesar una petición pendiente, la conexión se reseteará y la petición se acusará con error.

N.º máx. de repeticiones de transmisión

Indique aquí cuántas repeticiones de transmisión deben realizarse para una petición si el interlocutor no está accesible debido a una falta temporal de recursos, por ejemplo.

Tamaño del búfer de transmisión

Aquí se predefine el tamaño del búfer de transmisión predeterminado.

El ajuste realizado sirve tanto para transmisiones acusadas como no acusadas.

Durante el tiempo de ejecución es posible utilizar búferes de transmisión adicionales de una longitud cualquiera (máx. 244 ó 246 bytes).

Ajuste predeterminado:

244 bytes o 246 bytes (en SAPs predeterminados, cuando se ha ajustado el SAP predeterminado 255 para el SAP local y remoto)

N.º máx. de peticiones de transmisión en paralelo

Número máximo de peticiones de transmisión paralelas en la conexión configurada aquí.

Número de recursos para las peticiones recibidas

Indique el número de peticiones de recepción que el servidor OPC de la conexión FDL aquí configurada debe estar preparado para recibir simultáneamente.

Elija un valor equivalente al número de comunicaciones esperadas; es decir,

- En un número reducido de conexiones totales: elegir el valor más bien alto para asegurar un tiempo de reacción rápido.
- En un número elevado de conexiones totales: elegir el valor más bien bajo para impedir conflictos de recursos.
  - Ajuste predeterminado: 3

Enviar con prioridad alta

Indique si los telegramas FDL deben enviarse preferentemente vía LAN en la conexión configurada (opción activada).

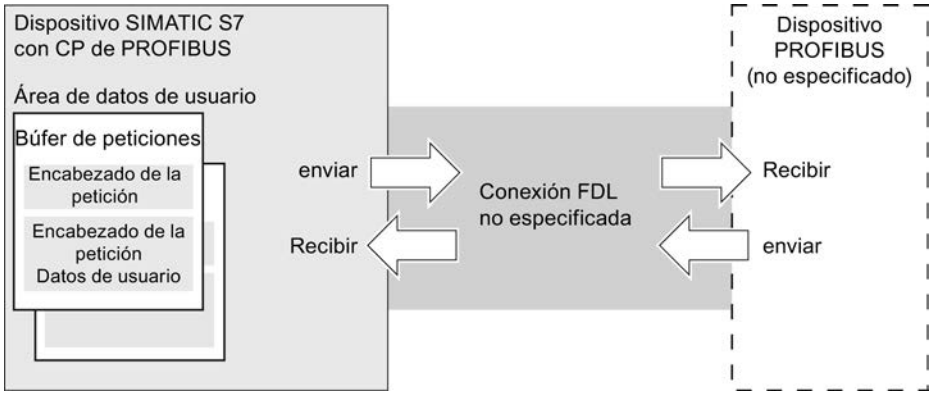
Conexión FDL no especificada (acceso libre Layer 2)

Propiedades

Una conexión FDL no especificada con acceso libre Layer 2 permite el direccionamiento del interlocutor controlado por el programa y la comunicación entre dos dispositivos de PROFIBUS con las siguientes propiedades:

- La transferencia de datos es bidireccional, es decir, en la conexión FDL es posible enviar y recibir de forma simultánea.
- El dispositivo local ha sido fijado mediante configuración. El programa de usuario registra el dispositivo remoto en el encabezado de la petición del búfer de peticiones al llamar la instrucción AG_SEND. De esta forma es posible acceder a todos los dispositivo de PROFIBUS (direcciones PROFIBUS de 0 a 126).
- En el encabezado de la petición de AG_RECV es posible leer la dirección PB, el LSAP y el servicio del emisor.

La siguiente representación ilustra el direccionamiento controlado por el programa en el búfer de peticiones:



Volumen de datos y equipamiento

Por cada búfer de peticiones es posible transmitir hasta 236 bytes de datos de usuario. El encabezado de la petición ocupa adicionalmente 4 bytes.

Conexión FDL con Broadcast

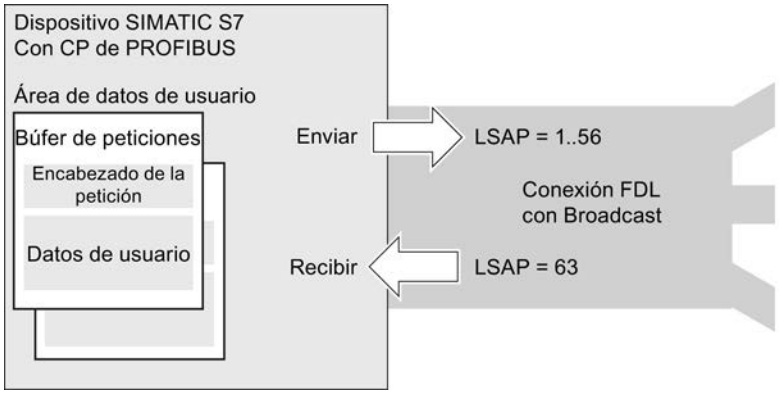
Propiedades

Una conexión Broadcast permite enviar un mensaje a varios destinatarios con **una** petición. Del mismo modo, es posible recibir mensajes en la misma conexión Broadcast que serán recibidos al mismo tiempo por otros dispositivos de PROFIBUS.

Es posible resumir las propiedades como sigue:

- La transferencia de datos es bidireccional, es decir, en la conexión Broadcast es posible enviar y recibir de forma simultánea.
- El envío y la recepción se realizan mediante el servicio FDL SDN (Send Data with No Acknowledge).
- La llamada de AG_SEND requiere un búfer de peticiones al enviar. Es necesario reservar el encabezado de la petición de este búfer.
- En el encabezado de la petición de AG_RECV es posible leer la dirección PB, el LSAP y el servicio del emisor Broadcast.
- Para el envío se utiliza el rango LSAP de 1 a 56. Para la recepción se reserva el LSAP 63 para todos los dispositivos Broadcast.

La siguiente representación ilustra la emisión y recepción a través de una conexión FDL con direccionamiento Broadcast por programa:



Configuración de una conexión FDL con Broadcast

Al crear la conexión FDL, como interlocutor/estación "Todos los dispositivos Broadcast".

Consulte también

Crear conexiones UDP/FDL con Multicast/Broadcast (Página 639)

Volumen de datos y equipamiento

El CP de PROFIBUS soporta en cada caso **una** conexión Broadcast.

Por cada búfer de peticiones es posible transmitir hasta 236 bytes de datos de usuario. El encabezado de la petición ocupa adicionalmente 4 bytes.

Nota

El CP recibe solo en una conexión Broadcast

Si se utiliza una conexión FDL con Broadcast, no es posible recibir mensajes en ninguna otra conexión Broadcast del CP en cuestión y tampoco en ninguna conexión FMS con Broadcast.
Motivo:
El LSAP de recepción para Broadcast (63) está ocupado por una conexión Broadcast.

Conexión FDL con Multicast

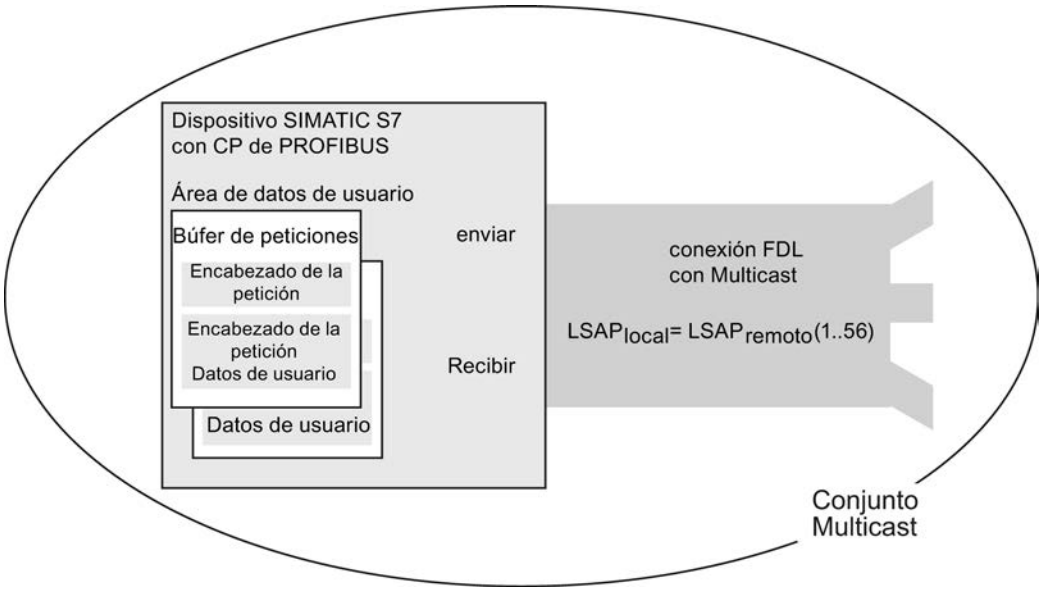
Propiedades

Una conexión FDL con Multicast permite el envío de un mensaje a varios destinatarios de un círculo Multicast mediante una petición.

Es posible resumir las propiedades como sigue:

- La transferencia de datos es bidireccional, es decir, en la conexión FDL con Multicast es posible enviar y recibir de forma simultánea.
- El envío y la recepción se realizan mediante el servicio FDL SDN (Send Data with No Acknowledge).
- El envío se realiza a través de un LSAP unívoco para el círculo Multicast (rango de 1 a 56).
- La llamada de AG_SEND requiere un búfer de peticiones al enviar. Es necesario reservar el encabezado de la petición de este búfer.
- En el encabezado de la petición de AG_RECV es posible leer la dirección PB, el LSAP y el servicio del emisor Multicast.

La siguiente representación ilustra la emisión y recepción a través de una conexión FDL con direccionamiento Multicast por programa:



Configuración de conexión FDL con Multicast

Al crear la conexión FDL, seleccione como interlocutor/estación "Todos los dispositivos Multicast".

Consulte también

Crear conexiones UDP/FDL con Multicast/Broadcast (Página 639)

Volumen de datos y equipamiento

Consulte el número de conexiones FDL soportadas por cada CP de PROFIBUS en la documentación de dispositivo relativa al CP de PROFIBUS. Mediante la aceptación de otros CPs es posible aumentar el número de conexiones por estación.

Por cada búfer de peticiones es posible transmitir hasta 236 bytes de datos de usuario. El encabezado de la petición ocupa adicionalmente 4 bytes.

Conexión de correo electrónico

Establecer una conexión de correo electrónico

El sistema de automatización puede enviar correo electrónico pero no puede recibirlo.

Para el envío de correo electrónico en el programa de usuario de la CPU S7 se utiliza la llamada de envío de la interfaz SEND/RECEIVE (FC AG_SEND / AG_LSEND).

Sinopsis para la configuración

Básicamente, para el envío de correo electrónico es necesario establecer **una** conexión de correo electrónico por CP. Mediante la conexión de correo electrónico se determina el servidor de correo a través del cual se entrega todo el correo electrónico enviado por el CP.

Es posible establecer una conexión de correo electrónico como sigue:

- Configuración de conexión (aplicación estándar)
  Este caso de aplicación se describe a continuación.
- Programa de usuario con FB CP_CONFIG y bloque de datos de configuración
  Existen áreas de aplicación en las que es ventajoso establecer las conexiones no a través de la interfaz de configuración, sino a través de aplicaciones específicas de forma controlada por programa.

Procedimiento: ejemplo de configuración gráfica

A continuación se describe la configuración de una conexión de correo electrónico. En principio, proceda como en el caso de una conexión no especificada: Crear una conexión no especificada (Página 637).

Seleccione el tipo de conexión de correo electrónico en la lista desplegable.

De esta forma se activa el modo de conexión: Las CPUs posibles para el tipo de conexión seleccionado han sido seleccionadas y se representan resaltadas.

1. Haga clic en la CPU de la que deba partir una conexión.
2. Desplace el puntero del ratón dentro de la CPU y confirme el punto final de la conexión mediante un nuevo clic.
   Se ha creado una conexión no especificada.
3. Introduzca los parámetros de dirección en "Propiedades > Detalles de direcciones" de la ventana de inspección conforme a los datos de los detalles de direcciones:
   Detalles de la dirección de correo electrónico (Página 688)

Tras la carga de los datos de configuración, el programa de usuario puede enviar correo electrónico a través de esta conexión de correo electrónico.

Uso del servidor OPC como servidor SMTP

En lugar de seleccionar una conexión no especificada, aquí puede seleccionar como interlocutor un servidor OPC como servidor SMTP. Con ello, en los detalles de direcciones del interlocutor se introduce automáticamente la dirección IP de la interfaz SMTP de la estación PC.

Nota

La interfaz del PC debe ser compatible con el protocolo SMTP

La conexión de correo electrónico a través del servidor OPC sólo se crea de manera coherente si en la estación PC está activada una interfaz compatible con el protocolo SMTP. Tenga en cuenta la información recogida en la documentación del módulo utilizado.

Detalles de la dirección de correo electrónico

Referencia

Configuración de conexiones en el grupo de parámetros "Propiedades > General > Detalles de direcciones".

Parámetros

Introduzca los parámetros necesarios para el envío del correo electrónico. Pueden introducirse los parámetros siguientes:

Parámetros	Descripción	Ejemplo
Servidor de correo electrónico (SMTP)	Dirección del servidor de correo a través del que se envían los correos electrónicos. La dirección IP puede indicarse en formato absoluto o simbólico. Para la entrada simbólica rige lo siguiente: Un nombre válido tiene entre 1 y 64 caracteres; de ellos, 1 carácter como mínimo debe ser una letra. Si en la entrada se utilizan uno o varios caracteres ".", rige además: Un nombre válido tiene detrás del último "." entre 1 y 3 caracteres; de ellos, 1 carácter como mínimo debe ser una letra. Para el formato simbólico es imprescindible que el CP Internet conozca la dirección del Domain Name System (DNS). Se debe realizar una entrada a este respecto en la configuración DNS al configurar el CP para Internet. Pueden introducirse 64 caracteres como máximo	absoluto:140.80.0.4 simbólico:t-online.de
Nombre predeterminado del remitente	Indicación de una dirección que se insertará como dirección del remitente en el correo electrónico siempre que la información del remitente (parámetro FROM) esté vacía en el encabezado del correo electrónico. Este dato no suele ser relevante para el envío de mensajes. Sirve de información para el destinatario del correo electrónico. Puesto que algunos servidores de correo no transmiten peticiones si el campo del remitente está vacío, es recomendable realizar una entrada. Pueden introducirse 126 caracteres como máximo. Nota: Tenga en cuenta que hay que indicar un nombre predeterminado de remitente si debe lanzarse un mensaje de test desde el diagnóstico especial.	Station2.CPU412@xy.werk2.de

Enviar un mensaje de correo electrónico

Procedimiento

Para enviar un mensaje de correo electrónico es necesario realizar los siguientes pasos:

1. Se preparan los datos de correo electrónico en un bloque de datos.
2. Se utiliza la instrucción AG_SEND o AG_LSEND en el programa de usuario.

Requisitos

Es posible enviar correo electrónico si se ha establecido la conexión de correo electrónico mediante la configuración de conexión. Se utiliza la ID definida durante la configuración de conexión al realizar la llamada a FC AG_SEND/AG_LSEND.

Bloque de datos

Todo el mensaje de correo electrónico, es decir, los datos de dirección y el propio mensaje, se generan en cualquier bloque de datos. A continuación se incluyen los datos correspondientes para la estructura de DB necesaria a partir de un ejemplo en notación AWL.

Utilice el editor de programación para crear e introducir los datos de DB.

La siguiente tabla contiene información acerca de los bloques de datos de correo electrónico en notación AWL.

Dirección	Nombre	Tipo	Valor inicial	Comentario	Entrada
0.0		STRUCT			
+0.0	TO[1)]	STRING[40]	'TO:name.name@t-online.de;'	Destinatario	Obligatorio
+42.0	CC[1)]	STRING[40]	'CC:name.name@t-online.de;'	CC destinatario	Opcional
+84.0	FROM	STRING[40]	'FROM:anlage.werk2@xyz-online.de;'	Remitente	Opcional
+126.0	SUB	STRING[40]	'SUB:Status Station 7;'	Asunto	Opcional
+168.0	Texto	STRING[100]	'TXT:Fallo en la sección 2 de la instalación;'	Texto del mensaje de correo	Obligatorio
+270.0	Anexo	STRING[4]	'BNY:'	Aquí comienza el anexo[3)]	Opcional
+276.0	Valor1	BYTE	**B#16#27**[2)]	Anexo/valor binario[3)]	Opcional
+277.0	Valor2	BYTE	**B#16#03**[2)]	Anexo/valor binario[3)]	Opcional
=278.0		END_STRUCT			

1) Se pueden indicar varios destinatarios. Las indicaciones deben separarse mediante comas.

2) Los datos marcados en negrita se entregan al destinatario como anexo

3) También es posible aportar los datos de forma dinámica.

Indicaciones relativas a la tabla

- Estructura y sintaxis de los datos en el DB de correo electrónico
  La estructura propuesta aquí con varios STRINGs es una de varias variantes. Son decisivas las entradas de la columna "Valor inicial" con los identificadores contenidos en ella (TO:, SUB:, CC:, FROM:, TXT:, BNY:), los cuales deben utilizarse exactamente con esta notación en el DB para la identificación de los contenidos del mensaje de correo. Es necesario cerrar aquí todas las entradas con un punto y coma; solo en la última entrada puede prescindirse del punto y coma.
  La longitud del string se indica en la tabla solo a modo de ejemplo; es posible adaptarla al número real de caracteres (excepción: La longitud del string para la identificación del anexo debe indicarse con [4]).
  Otra variante sería, por ejemplo, utilizar solo un STRING en total y asignar al mismo el texto completo con los identificadores.
- Si la introducción del carácter @ supone un problema, utilice la entrada ALT+64.
- Anexos
  Los datos útiles introducidos en el DB del correo electrónico también pueden ser entregados al destinatario en conjunto o parcialmente como anexo. Para ello, el remitente debe añadir a los datos el identificador 'BNY:' .
  Los datos introducidos a continuación de este identificador se entregan entonces al destinatario como anexo.
  En la tabla, el anexo tiene 2 bytes; ¡este es solo un ejemplo! Es posible introducir anexos con una complejidad aleatoria.
- Longitud de datos
  La longitud de datos indicada en AG_SEND/AG_LSEND debe cubrir como mínimo la longitud de los datos en el DB; tenga en cuenta los datos de la columna Dirección del editor de programación (nota: los datos se corresponden con el número de bytes).

Enviar un mensaje de correo electrónico con AG_SEND/AG_LSEND [1)]

Para enviar un mensaje de correo electrónico utilice la instrucción AG_SEND o, para longitudes de datos >240 bytes, AG_LSEND. Encontrará una descripción detallada relativa a los parámetros de llamada en la ayuda en pantalla relativa a las instrucciones.

Consulte también AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet) (Página 8056)

Ejemplo:

AWL	Explicación
call fc 50	//Llamada del bloque AG_LSEND
ACT := M 10.0	//Bit para el lanzamiento de la petición
ID := MW 12	//ID de la conexión (configuración de conexión)
LADDR := W#16#0100	//Dirección 256 dec. en la configuración hardware
SEND := P#db99.dbx10.0 byte 278,	//Dirección del bloque de datos; longitud de DB
LEN := MW 14	//Longitud del área de datos que se desea enviar
DONE := M 10.6	//Dirección para parámetro de retorno DONE

AWL	Explicación
ERROR := M 10.7	//Dirección para parámetro de retorno ERROR
STATUS := MW 16	//Dirección para parámetro de retorno STATUS

Conexión PtP

Comunicación PtP

Tipos de comunicación PtP

En la comunicación a través de una conexión PtP se distinguen dos tipos diferentes:

- Comunicación homogénea
- Comunicación no homogénea

Gracias a la compatibilidad con la comunicación PtP se pueden direccionar dispositivos externos, como impresoras.

Comunicación PtP homogénea

El CP PtP transmite los datos entre módulos compatibles con el protocolo S7. El usuario tiene a su disposición el conjunto completo de funciones de comunicación compatibles con los dispositivos finales interconectados entre sí a través del PtP (p. ej., CPU con CPU, o programadora con CPU).

En esta conexión, los CPs PtP no son el punto final de la conexión, sino que asumen la función de un router.

En el acoplamiento homogéneo se pueden usar todas las funciones de comunicación definidas en S7 que sean compatibles con ambos interlocutores. Por ello, no resulta necesario desarrollar un protocolo para llamadas específicas de instrucciones de comunicación en el CP.

Comunicación PtP no homogénea

El CP PtP funciona como gateway a través de otros protocolos, como RK512, 3964, controlador abierto y protocolos especiales. Determinados bloques siven como interfaz de usuario sincronizada con el programa de usuario. En la parte de recepción, el programa de usuario también puede ser direccionado a través de funciones de comunicación sincronizadas con el programa mediante el acoplamiento PtP no homogéneo.

En la comunicación no homogénea, y desde el punto de vista de la tecnología de comunicación, el CP PtP es el punto final de la comunicación S7. La comunicación entre CPU y CP se efectúa mediante elementos del protocolo S7. El CP se encarga de convertir estos elementos de protocolo al protocolo PtP y viceversa.

Para la comunicación no homogénea se pueden usar en la CPU las instrucciones de comunicación siguientes:

- BSEND
- BRCV

- PUT
- GET

En el CP PtP se deben implementar los elementos de protocolo S7 correspondientes a las instrucciones BSEND, BRCV, PUT y GET como cliente (el CP está activo) y PUT y GET como servidor (el CP está pasivo).

Instrucciones	Cliente	Servidor
BSEND (Página 7688)	sí	no
BRCV (Página 7691)	sí	no
PUT (Página 7680)	sí	sí
GET (Página 7677)	sí	sí

Funciones de comunicación

En función de la aplicación, la comunicación no homogénea se puede efectuar en dos calidades diferentes:

- Funciones de comunicación controladas:
  Las funciones de comunicación controladas permiten llevar a cabo la transmisión de manera controlada. Esto significa que la transmisión no sólo se puede iniciar desde la parte activa, sino que también se habilita de manera controlada desde la parte pasiva y se detecta el fin de la transmisión.
  En las instrucciones de comunicación en S7, esta funcionalidad se lleva a cabo con los parámetros "EN_R" (disponibilidad de recepción) y "NDR" (recepción de datos terminada).
- Funciones de comunicación no controladas:
  En las funciones de comunicación no controladas, la transmisión es efectuada por el interlocutor activo, sin que el interlocutor pasivo tenga posibilidad de intervenir.

Parametrizar conexión PtP

Configurar el hardware para PtP

Para establecer una conexión PtP se requiere un CP 441. Los módulos de comunicación del tipo CP 441 disponen de ranuras para uno (CP 441-1) o dos (CP 441-2) módulos de interfaz PtP.

Los módulos de interfaz PtP del tipo IF963 se encuentran en el catálogo de hardware del SIMATIC S7-400, bajo los módulos de comunicación para PtP. Existen módulos IF963 para los tipos de interfaz siguientes:

- RS232C
- TTY
- RS422/485

Según el tipo de interfaz, los módulos IF963 son compatibles como máximo con los protocolos de comunicación siguientes:

- ASCII
- 3964(R)
- RK512
- Impresora
- Maestro Modbus
- Esclavo Modbus

Si se utiliza una interfaz PtP en un CP 441, automáticamente se muestra una subred PtP y se inserta una conexión PtP no especificada.

Una vez seleccionado el CP 441 o tras seleccionar directamente la interfaz deseada, se puede elegir el protocolo en la ventana de inspección y configurarlo.

Conexión PtP con un dispositivo S7 como interlocutor

En la tabla de conexiones se pueden registrar 8 conexiones como máximo para el interlocutor local. Para el interlocutor se debe registrar sólo **una** conexión en la tabla de conexiones. El interlocutor puede recibir datos de las 8 conexiones a través de esta conexión.

Ejemplo: Definiciones para el interlocutor local:

- Protocolo 3964(R)
- Sentido de comunicación 1: Local --> interlocutor
- 8 conexiones máx., IDs locales: 1000; 1001, ..., 1007

Definiciones para el interlocutor:

- Protocolo 3964(R)
- Sentido de comunicación 2: Interlocutor --> local
- 1 conexión máx., ID local: 1000

Número de conexiones

El número máximo de conexiones configurables por dispositivo (CPU/FM) depende de los recursos del módulo empleado.

Sentido de comunicación PtP

Sentido de comunicación

En las conexiones PtP se pueden definir distintos sentidos de comunicación. Para ello, active la casilla de verificación correspondiente:

- Local → interlocutor:
  Conexiones a través de las cuales desea enviar telegramas.
- Interlocutor → local:
  Conexiones a través de las cuales desea recibir telegramas.
- Local ↔ interlocutor:
  Conexiones a través de las cuales desea enviar y recibir telegramas.

Nota

Encontrará información más detallada sobre cómo determinar el sentido de comunicación en la documentación de su módulo.

Vía de conexión PtP

Ajustes de la vía de conexión

En los ajustes del sentido de comunicación en "General > Vía de conexión" se puede configurar la vía que sigue la conexión que va del punto final local al interlocutor. Se puede elegir la vía en función de los CPs PtP configurados. Ejemplo: El interlocutor local contiene dos CPs PtP. En este caso se puede elegir a través de qué CP debe conducirse la conexión.

Particularidad: A diferencia de lo que ocurre en las conexiones S7, la configuración de conexiones PtP con un interlocutor no especificado no requiere la interconexión del interlocutor local. En este caso, los interlocutores se deben interconectar en la instalación real antes de poner la conexión en funcionamiento.

Posibilidades de selección de los interlocutores

- Vía CP:
  Para el interlocutor local se puede seleccionar el CP PtP a través del cual debe conducirse la conexión (nombre del CP con indicación del rack y el slot).
- Interfaz:
  Los CPs de SIMATIC S7 cuentan con más canales (interfaces) a través de los que se pueden establecer las conexiones PtP. Se muestra el canal y el protocolo usado para el canal (canal/protocolo). El protocolo se ha configurado con el software especial de configuración para el CP.
- Tipo:
  Aquí se visualiza el tipo de interfaz "PtP".

"Conexión seleccionada a través del n.º de CPU RK512:" (local)

Los sistemas de automatización con modo multiprocesador (como, p. ej., SIMATIC S7-400) pueden contener varias CPUs. Por ello, se debe indicar un número de CPU, a través del cual el interlocutor pueda direccionar la conexión.

Si se ha elegido como sentido de comunicación **interlocutor -> local** o bien **local <-> interlocutor** y se desea recibir telegramas del interlocutor a través de un SFB, introduzca aquí el número de la CPU (1 a 4) del **interlocutor local**.

El número de CPU aquí indicado debe coincidir con el número de CPU introducido en el encabezado del telegrama de comando del CP (la estructura del telegrama de comando se puede consultar en la documentación del CP empleado). Si coincide, el interlocutor direcciona la conexión a través de este número de CPU.

Nota

Si desea enviar y recibir telegramas a través de SFBs debe rellenar ambos campos en "Local" e "Interlocutor". El número máximo de conexiones a través de las que se pueden enviar/recibir telegramas se puede consultar en la documentación del CP empleado.

"N.º de CPU RK512:" (interlocutor)

Los sistemas de automatización con modo multiprocesador (como, p. ej., SIMATIC S7-400) pueden contener varias CPUs. Por ello, se debe indicar el número de la CPU a la que deba conducirse la conexión.

Si se ha elegido como sentido de comunicación **local -> interlocutor** o bien **local <-> interlocutor** y desea enviar telegramas al interlocutor a través de un SFB, introduzca aquí el número de la CPU (1 a 4) del **interlocutor**.

El CP introduce el número de CPU aquí indicado en el encabezado del telegrama de comando que se va a enviar. Por medio de este número de CPU se direcciona una de las 4 CPUs del interlocutor.

En caso de transmisión hacia dispositivos de terceros también se puede indicar "0" como número de CPU. Esto posibilita el acceso Broadcast en la estación interlocutora. Si el interlocutor es el CP441, se direcciona siempre la CPU 1.

Nota

Si desea enviar y recibir telegramas a través de SFBs debe rellenar ambos campos en "Local" e "Interlocutor". El número máximo de conexiones a través de las que se pueden enviar/recibir telegramas se puede consultar en la documentación del CP empleado.

PtP: Propiedades especiales de la conexión

Referencia

En el grupo de parámetros "Propiedades > General > Propiedades especiales de la conexión" se pueden definir propiedades adicionales del punto final local de la conexión.

Unilateral

Unilateral significa que el interlocutor es el servidor de dicha conexión y que no puede enviar o recibir activamente.

Establecer conexión activa

Determine con esta opción si el establecimiento de la conexión debe realizarse desde este dispositivo. Esta opción está seleccionada por defecto si la dirección del interlocutor está especificada.

- ON: La conexión se establece activamente.
- OFF: El interlocutor establece la conexión.

Si ha seleccionado como interlocutor "no especificado" al crear la conexión, la opción está deseleccionada por defecto. Si se activa esta opción, es necesario especificar la dirección del interlocutor en el registro "Direcciones".

Nota

No todos los dispositivos tienen la capacidad de establecer una conexión. En este caso, el interlocutor debe tomar la iniciativa; las casillas de verificación se muestran en color gris y no es posible modificarlas.

Enviar avisos de estado operativo

Los interlocutores también pueden intercambiar sus avisos de estado operativo a través de una conexión existente.

Active la casilla de verificación si el dispositivo local debe enviar sus avisos de estado operativo al interlocutor. Seguidamente, los avisos de estado se reciben en el interlocutor mediante la instrucción USTATUS (Página 7722) (SFB 23).

Utilizar Open User Communication (S7-1200, S7-1500)

Principios básicos de la Open User Communication

Introducción

Open User Communication es el nombre de un procedimiento de comunicación controlado por programa para la comunicación a través de la interfaz PN/IE integrada de las CPUs S7-1200/1500 o S7-300/400. Para este procedimiento de comunicación se dispone de diversos tipos de conexiones.

La Open User Communication destaca por una elevada flexibilidad en cuanto a las estructuras de datos que deben transferirse, con lo que permite la comunicación abierta con cualquier nodo de comunicación, siempre que éste soporte los tipos de comunicación disponibles. Puesto que la comunicación se controla exclusivamente por medio de las instrucciones del programa de usuario, es posible establecer y deshacer conexiones en función de un evento.

Asimismo, las conexiones pueden modificarse desde el programa de usuario durante el funcionamiento.

Para las CPUs con una interfaz PN/IE integrada, están disponibles los tipos de conexión TCP, UDP e ISO on TCP para la Open User Communication. Los interlocutores pueden ser dos PLCs SIMATIC o bien un PLC SIMATIC y un dispositivo de terceros adecuado.

Instrucciones para la Open User Communication

Después de abrir el editor de programación, en la Task Card "Instrucciones > Comunicación > Open user Communication" se dispone de distintas instrucciones para crear las conexiones, dependiendo del tipo de CPU:

- Instrucciones compactas para enviar o recibir datos con las funciones integradas para establecer o deshacer la conexión (solo S7-1200/1500):
  - TSEND_C (Página 5263) (establecer/deshacer conexión, enviar)
  - TRCV_C (Página 5274) (establecer/deshacer conexión, recibir)
- Instrucciones individuales para enviar y recibir datos o para establecer o deshacer las conexiones:
  - TCON (Página 5298) (establecer conexión)
  - TDISCON (Página 5305) (deshacer conexión)
  - TSEND (Página 5310) (TCP o ISO on TCP: enviar)
  - TRCV (Página 5314) (TCP o ISO on TCP: recibir)
  - TUSEND (Página 5323) (UDP: enviar)
  - TURCV (Página 5326) (UDP: recibir)

Establecimiento de la conexión

En la Open User Communication deben existir instrucciones para establecer y deshacer la conexión para ambos interlocutores. Un interlocutor envía los datos mediante TSEND, TUSEND o TSEND_C, mientras que el otro los recibe mediante TRCV, TURCV o TRCV_C.

Uno de los interlocutores empieza a establecer la conexión en calidad de interlocutor activo. El otro interlocutor reacciona y empieza a establecer su conexión en calidad de interlocutor pasivo. Cuando ambos interlocutores han lanzado su establecimiento de conexión, se establece por completo la conexión.

Parametrización de la conexión

El establecimiento de la conexión se parametriza a través de un DB de descripción de conexión con la estructura TCON_Param, TCON_IP_v4 o TCON_IP_RFC, del modo siguiente:

- Crear manualmente, parametrizar y escribir directamente en la instrucción.
- Soportado por la parametrización de la conexión.

La parametrización de la conexión soporta el establecimiento de la conexión, por lo que es recomendable utilizar este método antes que otros.

En la parametrización de la conexión se determina lo siguiente:

- Interlocutor
- Tipo de conexión
- ID de conexión
- DB de descripción de la conexión
- Detalles de la dirección en función del tipo de conexión seleccionado

Asimismo, se determina el interlocutor que activa el establecimiento de la conexión y el que responde a la petición del interlocutor estableciendo una conexión pasiva.

Consulte también

Funcionamiento de los protocolos orientados a la conexión (Página 708)

Parametrización de la conexión

Resumen de la parametrización de la conexión

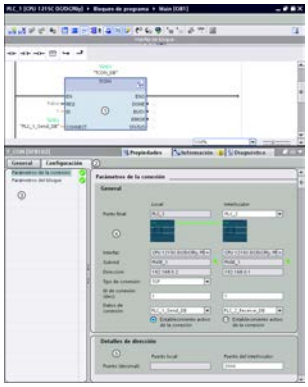
Introducción

La parametrización de la conexión se encuentra en la ventana de inspección del editor de programación si se desea programar una Open User Communication con las instrucciones de comunicación TSEND_C, TRCV_C o TCON.

La parametrización de la conexión soporta la funcionalidad flexible de la programación de la comunicación: En un DB global generado automáticamente y derivado de la estructura del tipo TCON_Param, TCON_IP_v4 o TCON_IP_RFC se guardan los parámetros introducidos en la parametrización de la conexión. En este DB de descripción de la conexión se modifican los parámetros de la misma.

Estructura de la parametrización de la conexión

La parametrización de la conexión incluye los componentes siguientes:



① Instrucción de comunicación para TCON, TSEND_C o TRCV_C

② Ficha "Configuración" en la ficha "Propiedades"

③ Navegación local de la ficha "Configuración"

④ Propiedades generales de los parámetros de conexión

⑤ Detalles de dirección de los parámetros de la conexión (con DBs de conexión seleccionados)

Ficha "Configuración"

Introduzca los parámetros de conexión deseados en la ficha "Configuración". Encontrará el grupo "Parámetros de la conexión" en la navegación local de la ficha "Configuración". Dicho grupo contiene la parametrización de la conexión. Aquí se introducen los parámetros de las conexiones y los detalles de dirección con soporte del sistema. También se cablean los parámetros de bloque CONNECT (TCON, TSEND_C, TRCV_C) o ID (TCON, TSEND, TRCV, TUSEND, TURCV) de las instrucciones de comunicación seleccionadas.

Cuando ya están todos los parámetros, se activa una marca de verificación delante del grupo "Parámetros de la conexión" de la navegación local.

Nota

La parametrización de la conexión no comprueba que las ID de conexión y los números de puerto (TCP, UDP) o TSAP (ISO on TCP, ISO) sean unívocos. Por consiguiente, al configurar la Open User Communication hay que procurar asignar parámetros unívocos dentro de un dispositivo.

Consulte también

Parámetros de conexión conforme a TCON_Param (Página 711)

Parámetros de conexión conforme a TCON_IP_v4 (Página 713)

Parámetros de conexión conforme a TCON_IP_RFC (Página 714)

Descripción de los parámetros de conexión

Resumen

La tabla siguiente muestra los parámetros de conexión generales:

Parámetro	Descripción
Punto final	Se muestra el nombre del punto final local y del punto final del interlocutor. El punto final local es la CPU para la que se programa TCON, TSEND_C o TRCV_C. Por tanto, el punto final local siempre se conoce. El punto final del interlocutor se selecciona en una lista desplegable. Dicha lista muestra todos los interlocutores disponibles, entre ellos también interlocutores sin especificar para dispositivos cuyos datos son desconocidos dentro del proyecto. En S7-1500 puede seleccionarse como punto final del interlocutor "Broadcast" (aviso a todos los dispositivos de la subred). En CPs o CMs S7-1500 puede seleccionarse como punto final del interlocutor "Multicast" (aviso a un grupo dentro de la subred). En este caso el tipo de conexión cambia automáticamente a UDP. Mientras no se haya ajustado ningún interlocutor, todos los demás parámetros de la máscara permanecerán desactivados.
Interfaz	Se muestra la interfaz del punto final local. Si hay disponibles varias interfaces, por ejemplo también a través de CPs o de CMs, la interfaz puede seleccionarse a través de la lista desplegable. La interfaz del interlocutor no se visualiza ni se selecciona hasta que se ha seleccionado un punto final especificado del interlocutor.
Subred	Se muestra la subred del punto final local, si existe. La subred del interlocutor no se visualiza hasta que se selecciona el punto final del interlocutor. Si como mínimo uno de los dos interlocutores no está conectado a una subred, se conectarán en red los dos interlocutores. Solo es posible conectar interlocutores a subredes distintas a través de un routing IP. La configuración del routing IP puede modificarse en las propiedades de las interfaces Ethernet en cuestión.
Dirección	Se muestra la dirección IP del punto final local. La dirección IP del interlocutor no se visualiza hasta que se selecciona el punto final del interlocutor. Si se ha seleccionado un interlocutor no especificado, el campo de entrada estará vacío y resaltado en rojo. En este caso hay que indicar una dirección IP válida. Broadcast (solo S7-1500): Si se ha configurado como punto final del interlocutor "Broadcast", para el interlocutor se registra automáticamente una dirección IP no modificable con la dirección de host 255. La parte de red es la del emisor. Ejemplo: Dirección IP local 192.168.0.1, dirección IP del interlocutor 192.168.0.255. Multicast (solo para CPs y CMs S7-1500): Si se ha configurado como punto final del interlocutor "Multicast", para el interlocutor se registra automáticamente la dirección IP modificable 224.0.1.0.
Tipo de conexión	Elija el tipo de conexión que debe utilizarse en la lista desplegable "Tipo de conexión": • TCP • ISO-on-TCP • UDP En el S7-1500 se puede seleccionar también el tipo de conexión ISO en TSEND_C y TRCV_C o TCON en el tipo de configuración de las conexiones configuradas. Los tipos de conexión solo pueden emplearse con interlocutores que sean asimismo compatibles con el protocolo correspondiente.

Parámetro	Descripción
Tipo de configuración (solo para S7-1500)	Para S7-1500 es posible ajustar en TSEND_C y TRCV_C dos tipos de configuración diferentes: • Las conexiones programadas utilizan bloques de programa para describir la conexión. • Las conexiones configuradas se guardan y se crean una vez cargada la configuración en el dispositivo, en el tiempo de ejecución. A través de la conexión configurada se puede seleccionar también el tipo de conexión ISO. El tipo de configuración predeterminado depende del tipo de conexión seleccionado. Si son posibles ambos tipos de configuración, está predeterminada la conexión programada. Se debe configurar el mismo tipo de conexión en ambos interlocutores.
ID de conexión	Introduzca la ID de conexión en el campo de entrada. La ID de conexión se modifica en los campos de entrada o bien se indica directamente en TCON. Procure asignar una ID de conexión unívoca dentro del dispositivo.
Datos de conexión	En las listas desplegables se muestran los nombres de los DB de descripción de la conexión estructurados según TCON_IP_v4, TCON_IP_RFC o TCON_Param. Después de seleccionar un interlocutor, la lista desplegable permanece vacía. Mediante la lista desplegable puede generarse un bloque de datos nuevo o seleccionarse uno ya existente. Al mismo se le asignarán automáticamente los valores de la parametrización de la conexión. El nombre del bloque de datos seleccionado se registra automáticamente en el parámetro de bloque CONNECT de las instrucciones seleccionadas TSEND_C, TRCV_C o TCON. Desde la lista desplegable también es posible referenciar otro bloque de datos válido. Si con el parámetro de entrada CONNECT de las instrucciones avanzadas TSEND_C, TRCV_C o TCON se hace referencia a un DB que no se corresponde con la estructura de un TCON_IP_v4, TCON_IP_RFC o TCON_Param, la lista desplegable aparecerá vacía y resaltada en rojo.
Nombre de conexión (solo para S7-1500)	Si para S7-1500 está ajustado el tipo de configuración de las conexiones configuradas en TSEND_C y TRCV_C, el parámetro "Datos de conexión" es reemplazado por el parámetro "Nombre de conexión". En este caso, el nombre de la conexión configurada sirve de fecha de la conexión. Después de seleccionar un interlocutor, la lista desplegable permanece vacía. Mediante la lista desplegable puede generarse una conexión nueva o seleccionarse una ya existente. Dado el caso, se genera un bloque de datos y se le asignan automáticamente los valores de la parametrización de la conexión. El nombre del bloque de datos se registra automáticamente en el parámetro de bloque CONNECT de la instrucción TSEND_C o TRCV_C. Desde la lista desplegable también es posible referenciar una conexión existente.
Iniciativa local	Determine el interlocutor activo de la Open User Communication mediante el botón de opción "Iniciativa local" (solo en TCP e ISO on TCP).
Puerto (solo con TCP y UDP)	Componente de dirección para una conexión TCP o UDP. El valor predeterminado después de crear una nueva conexión TCP es 2000. Los números de puerto pueden cambiarse pero tienen que ser unívocos dentro del dispositivo.
TSAP (solo con ISO on TCP)	Componente de dirección para una conexión ISO on TCP. El valor predeterminado después de crear una conexión ISO on TCP nueva es E0.01.49.53.4F.6F.6E.54.43.50.2D.31 (S7-1200/1500) o E0.02.49.53.4F.6F.6E.54.43.50.2D.31 (S7-300/400). La ID TSAP puede introducirse con extensión o como TSAP ASCII. Los TSAPs tienen que ser unívocos dentro del dispositivo.

Nota

Conexión UDP para configuración "Broadcast" (S7-300/400/1200)

Los parámetros de la conexión UDP para la configuración "Broadcast" en el punto final del interlocutor se guardan en un DB de descripción de la conexión TCON_IP_v4 : Respecto a la comunicación UDP con TCON y TUSEND/TURCV , al TCON_IP_v4 no se le asigna ningún parámetro de interlocutor (valor=0). Sin embargo, la dirección y el puerto interlocutor son necesarios para el envío de los datos, y el usuario debe registrarlos en TADDR_Param . El parámetro de bloque TUSEND-/TURCVhace referencia al TADDR_Param para la comunicación UDP como ADDR . Los valores para ambos parámetros pueden obtenerse de la parametrización de la conexión.

La configuración debe adaptarse también a los demás receptores de la comunicación UDP. Para recibir telegramas Broadcast, en el lado del receptor debe configurarse el puerto interlocutor. Para ello, en el bloque ADDRdebe rellenarse el parámetro RemotePort del TADDR_Param .

Nota

Comunicación vía TSEND_C y TRCV_C (S7-1500)

Si se emplean TSEND_C y TRCV_C, cada comunicación requiere su propia pareja de bloques TSEND_C y TRCV_C con una conexión configurada. Para la comunicación, la misma conexión configurada no puede ser utilizada por varias parejas de bloques TSEND_C y TRCV_C.

En los parámetros de conexión de la ventana de inspección pueden crearse, con los botones correspondientes, conexiones adicionales para una instrucción TSEND_C o TRCV_C, además de los datos de la conexión.

Las conexiones configuradas mediante TSEND_C y TRCV_C se muestran para el bloque seleccionado TSEND_C o TRCV_C en la ventana de inspección, dentro de una tabla de conexiones en "Propiedades > Configuración > Vista general de las conexiones configuradas".

Consulte también

Asignación de números de puerto (Página 716)

Estructura TSAP (Página 718)

Ejemplos de asignación TSAP (Página 720)

Capacidad de relectura de los parámetros para la descripción de la conexión (Página 717)

Crear y parametrizar conexiones (Página 704)

Parámetros de conexión conforme a TCON_Param (Página 711)

Parámetros de conexión conforme a TCON_IP_v4 (Página 713)

Parámetros de conexión conforme a TCON_IP_RFC (Página 714)

Iniciar la parametrización de la conexión

La parametrización de la conexión para la Open User Communication se activa en cuanto se selecciona una instrucción de comunicación TCON, TSEND_C o TRCV_C en un bloque de programa.

Requisitos

- El proyecto contiene una CPU S7 como mínimo.
- El editor de programación está abierto.
- Existe una red.

Procedimiento

Para insertar las instrucciones avanzadas para la Open User Communication, proceda del siguiente modo:

1. Abra la Task Card, paleta y carpeta "Instrucciones > Comunicación > Open User Communication".
2. Arrastre una de las instrucciones siguientes a una red con la función Drag & Drop:
   - TSEND_C
   - TRCV_C
   - TCON

   Se abre el cuadro de diálogo "Opciones de llamada".
3. Edite las propiedades del DB instancia en el cuadro de diálogo "Opciones de llamada". Existen las posibilidades siguientes:
   - Modificar el nombre predeterminado.
   - Seleccionar el campo de opción "manual" para asignar un número propio.
   - Para los bloques de función también puede utilizar el bloque de datos como multiinstancia.
4. Haga clic en "Aceptar" para concluir la entrada.

Resultado

Para la instrucción insertada TSEND_C, TRCV_C o TCON se crea, para una instancia individual, un DB de instancia correspondiente. En caso de una multiinstancia, se utiliza el DB instancia del bloque de función.

Si TSEND_C, TRCV_C o TCON está seleccionado, consulte la ficha "Configuración" en "Propiedades" de la ventana de inspección. El grupo "Parámetros de la conexión" de la navegación local contiene la parametrización de la conexión que debe realizarse ahora.

Consulte también

Insertar elementos FUP mediante la Task Card "Instrucciones" (Página 2724)

Insertar elementos KOP mediante la Task Card "Instrucciones" (Página 2681)

Crear y parametrizar conexiones (Página 704)

Principios básicos de las instancias individuales y multiinstancias (Página 2264)

Crear y parametrizar conexiones

En la parametrización de la conexión de la Open User Communication se crean y parametrizan conexiones del tipo TCP, UDP e ISO on TCP.

Requisitos

Existe una CPU con una instrucción de comunicación TCON, TSEND_C o TRCV_C.

Procedimiento

Para crear una conexión de la Open User Communication, proceda del siguiente modo:

1. Seleccione un bloque de la Open User Communication TCON, TSEND_C o TRCV_C en el editor de programación.
2. En la ventana de inspección abra la ficha "Propiedades > Configuración".
3. Seleccione el grupo "Parámetros de la conexión". Mientras no se haya seleccionado ningún interlocutor, solo estará activa la lista desplegable vacía del punto final del interlocutor. Todas las demás entradas posibles estarán desactivadas.
   Se mostrarán los parámetros de conexión que ya se conocen:
   - Nombre del punto final local
   - Interfaz del punto final local
   - Dirección IP del punto final local
4. Seleccione un interlocutor en la lista desplegable del punto final del interlocutor. El interlocutor puede ser un dispositivo sin especificar o una CPU del proyecto. Seguidamente, se registrarán automáticamente algunos parámetros de la conexión como valores predeterminados.
   Se ajustan los parámetros siguientes:
   - Nombre del punto final del interlocutor
   - Interfaz del punto final del interlocutor
   - Dirección IP del punto final del interlocutor

   Si los interlocutores están conectados en red, se muestra el nombre de la subred.
5. Para S7-1500 elija entre utilizar bloques de programa o conexiones configuradas en la lista desplegable "Tipo de configuración".

6. En la lista desplegable "Datos de conexión" seleccione un DB existente de descripción de la conexión o, para conexiones configuradas, una conexión existente en "Nombre de conexión". También es posible crear otro DB de descripción de la conexión u otra conexión configurada. Posteriormente puede seleccionar otros DBs de descripción de la conexión o conexiones configuradas o bien modificar los nombres de los DBs de descripción de la conexión para crear bloques de datos nuevos:
   - El bloque de datos seleccionado también se ve en el cableado del parámetro de entrada CONNECT de la instrucción seleccionada TCON, TSEND_C o TRCV_C.
   - Si ya se ha indicado un DB de descripción de la conexión para el interlocutor mediante el parámetro CONNECT de la instrucción TCON, TSEND_C o TRCV_C, podrá utilizarse dicho DB o bien crearse uno nuevo.
   - Si se edita el nombre del bloque de datos visualizado en la lista desplegable, se genera automáticamente un bloque de datos nuevo con el nombre modificado pero con la misma estructura y el mismo contenido y se utiliza para la conexión.
   - Los nombres modificados de un bloque de datos tienen que ser unívocos en el contexto del interlocutor.
   - Dependiendo del tipo de CPU y de la conexión, un DB de descripción de la conexión debe tener la estructura TCON_Param, TCON_IP_v4 o TCON_IP_RFC.
   - No es posible seleccionar un bloque de datos para un interlocutor sin especificar.

   Tras seleccionar o crear el DB de descripción de la conexión o la conexión configurada se determinan y registran más valores.
   Para interlocutores especificados rige lo siguiente:
   - Tipo de conexión ISO on TCP
   - ID de conexión con el valor predeterminado 1
   - Iniciativa local del interlocutor local
   - ID de TSAP
     para S7-1200/1500: E0.01.49.53.4F.6F.6E.54.43.50.2D.31
     para S7-300/400: E0.02.49.53.4F.6F.6E.54.43.50.2D.31

   Para interlocutores sin especificar rige lo siguiente:
   - Tipo de conexión TCP
   - Puerto interlocutor 2000

   Para una conexión configurada con interlocutor especificado rige lo siguiente:
   - Tipo de conexión TCP
   - ID de conexión con el valor predeterminado 257
   - Iniciativa local del interlocutor local
   - Puerto interlocutor 2000

   Para una conexión configurada con interlocutor no especificado rige lo siguiente:
   - Tipo de conexión TCP
   - Puerto local 2000

7. Indique una ID de conexión para el interlocutor si fuera necesario. Si hay un interlocutor sin especificar, no es posible asignarle ninguna ID de conexión.

   Nota

   Para un interlocutor conocido debe introducir una ID de conexión unívoca. La univocidad de la ID de conexión no es verificada por la parametrización de conexiones y al crear una conexión no se registra ningún valor predeterminado para la ID de la conexión.

8. Seleccione el tipo de conexión deseado en la lista desplegable correspondiente. Los detalles de la dirección se ajustan con valores estándar en función del tipo de conexión. Puede elegir entre:
   - TCP
   - ISO-on-TCP
   - UDP

   Para conexiones configuradas en el S7-1500 también vale ISO.
9. Los campos de entrada se editan en los detalles de dirección. Según sea el protocolo ajustado, se pueden editar los puertos (para TCP y UDP) o los TSAP (para ISO on TCP e ISO).
10. Ajuste el comportamiento de establecimiento de la conexión con los botones de opción "Iniciativa local" en TCP, ISO e ISO on TCP. Existe la posibilidad de elegir qué interlocutor debe establecer la conexión activamente.

La parametrización de la conexión comprueba inmediatamente si los valores modificados presentan errores de entrada y los registra en el bloque de datos para la descripción de la conexión.

Nota

La Open User Communication entre dos interlocutores no estará lista hasta que se haya cargado en el hardware también la parte del programa destinada al punto final del interlocutor. Para que la comunicación funcione, asegúrese de no cargar en el dispositivo únicamente la descripción de la conexión de la CPU local, sino también la de la CPU partner.

Consulte también

Descripción de los parámetros de conexión (Página 700)

Iniciar la parametrización de la conexión (Página 703)

Estructura TSAP (Página 718)

Asignación de números de puerto (Página 716)

Parámetros de conexión conforme a TCON_Param (Página 711)

Parámetros de conexión conforme a TCON_IP_v4 (Página 713)

Parámetros de conexión conforme a TCON_IP_RFC (Página 714)

Borrar conexiones

Introducción

Los datos de una conexión creada para la Open User Communication se almacenan en un DB de descripción de la conexión. La conexión se borra borrando el bloque de datos con la descripción de la conexión incluida.

Requisitos

Se ha creado una conexión de la Open User Communication.

Procedimiento

Para borrar una conexión, proceda del siguiente modo:

1. Seleccione un interlocutor de la Open User Communication en el árbol del proyecto.
2. Abra la carpeta "Bloques de programa > Bloques de sistema > Recursos de programa" situada debajo del interlocutor seleccionado.
3. Elija el comando "Borrar" del menú contextual del bloque de datos con la parametrización de la conexión.

Nota

Si no sabe exactamente qué bloque hay que borrar, abra la instrucción avanzada TCON, TSEND_C o TRCV_C. Encontrará el nombre del bloque de datos como parámetro de entrada CONNECT o en la parametrización de la conexión como parámetro "Datos de conexión".

Si sólo se borra el DB instancia de las instrucciones avanzadas TCON, TSEND_C o TRCV_C, no se borrarán las conexiones parametrizadas.

Nota

Si todavía hay bloques de las instrucciones avanzadas que están utilizando el DB de conexión, habrá que borrar también de la carpeta de bloques las llamadas correspondientes, sus DBs de instancia y, si existen, los bloques combinados TSEND_C y TRCV_C, siempre que no se utilicen en otro lugar.

De este modo se impide que el programa sea incoherente.

Resultado

Se ha borrado la conexión.

Nota

Vuelva a insertar una instrucción avanzada TCON, TSEND_C o TRCV_C para hacer referencia de nuevo mediante el parámetro "Datos de conexión" a una descripción de conexión ya existente de estructura TCON_Param, TCON_IP_v4 o TCON_IP_RFC.

Funcionamiento de los protocolos

Funcionamiento de los protocolos orientados a la conexión

Introducción

Los protocolos orientados a la conexión establecen una conexión lógica con el interlocutor antes de la transferencia de datos y la deshacen una vez concluida la transferencia, dado el caso. Los protocolos orientados a la conexión se utilizan especialmente cuando lo importante en la transferencia de datos es la fiabilidad. A través de una línea física es posible establecer varias conexiones lógicas.

La Open User Communication soporta los tipos de conexión siguientes:

- TCP
- ISO-on-TCP
- ISO (solo S7-1500)
- UDP

Para una conexión, los dos interlocutores deben soportar el mismo tipo de conexión. Si, por ejemplo, un interlocutor no soporta una conexión del tipo ISO on TCP, pruebe a utilizar el tipo de conexión TCP, si es que se soporta.

Para interlocutores que no se puedan configurar en el TIA Portal, por ejemplo dispositivos de terceros o PC, introduzca el punto final del interlocutor "sin especificar" en la parametrización de la conexión. El tipo de conexión necesario para dispositivos no especificados debe consultarse en la documentación correspondiente.

Nota

Conexiones con ISO

En las CPU S7-1500 es posible crear conexiones configuradas del tipo ISO mediante las instrucciones TSEND_C y TRCV_C. Para más información sobre estos tipos de conexiones consulte las descripciones generales de las conexiones.

Características de TCP

TCP es un protocolo de streaming en el que la longitud del flujo de datos se transmite al receptor para que pueda recibir el flujo de datos en forma de segmentos TCP individuales. Por tanto, en la transferencia de datos a través de una conexión TCP no se transmite información sobre el principio y el final de un mensaje. El receptor no puede saber dónde termina un mensaje en el flujo de datos y dónde empieza el siguiente a partir de los segmentos recibidos en el flujo de datos. Por ello, es recomendable asignar al número de bytes que deben recibirse (parámetro LEN, instrucción TRCV/TRCV_C) el mismo valor que al número de bytes que deben enviarse (parámetro LEN, instrucción TSEND/TSEND_C).

Si la longitud de los datos enviados no coincide con la de los datos esperados, se procede del siguiente modo:

- Longitud de los datos que deben recibirse (parámetro LEN, instrucción TRCV/TRCV_C) mayor que la longitud de los datos enviados (parámetro LEN, instrucción TSEND/TSEND_C):
  TRCV/TRCV_C no copia los datos recibidos en el área de recepción predefinida (parámetro DATA) hasta que se ha alcanzado la longitud parametrizada. Cuando se alcanza la longitud parametrizada, ya se reciben datos de una petición posterior. En este caso, en el área de recepción habrá datos de dos peticiones de transmisión distintas. Si no se conoce la longitud exacta del primer mensaje, no será posible reconocer el final del primero ni el principio del segundo.
- Longitud de los datos que deben recibirse (parámetro LEN, instrucción TRCV/TRCV_C) menor que la longitud de los datos enviados (parámetro LEN, instrucción TSEND/TSEND_C):
  TRCV/TRCV_C copia tantos bytes en el área de datos recibidos (parámetro DATA) como los predefinidos en el parámetro LEN. Seguidamente, pone el parámetro de estado NDR a TRUE (petición concluida correctamente) y asigna el valor de LEN a RCVD_LEN (número de datos realmente recibidos). Con cada llamada posterior se obtendrá un bloque adicional de los datos enviados.

A través de la variante de protocolo del modo Ad-hoc, en las instrucciones TRCV/TRCV_C se puede determinar un área de recepción con una longitud de datos fija.

Características de ISO on TCP

ISO on TCP es un protocolo orientado a mensajes que detecta el fin del mensaje en el lado del receptor e indica al usuario los datos pertenecientes al mensaje. Esto no depende de la longitud de recepción especificada del mensaje. Por tanto, al transferir datos a través de una conexión ISO on TCP, se suministra información relacionada con la longitud y el final de un mensaje.

Si la longitud de los datos enviados no coincide con la de los datos esperados, se procede del siguiente modo:

- Longitud de los datos que deben recibirse (parámetro LEN, instrucción TRCV/TRCV_C) mayor que la longitud de los datos enviados (parámetro LEN, instrucción TSEND/TSEND_C):
  TRCV/TRCV_C copia todos los datos enviados en el área de datos recibidos (parámetro DATA). Seguidamente, pone el parámetro de estado NDR a TRUE (petición concluida correctamente) y asigna la longitud de los datos enviados a RCVD_LEN (número de datos realmente recibidos).
- Longitud de los datos que deben recibirse (parámetro LEN, instrucción TRCV/TRCV_C) menor que la longitud de los datos enviados (parámetro LEN, instrucción TSEND/TSEND_C):
  TRCV/TRCV_C no copia datos en el área de datos recibidos (parámetro DATA), sino que proporciona la información de error siguiente: ERROR=1, STATUS=W#16#8088 (búfer de destino demasiado pequeño).

Características de UDP

UDP es un protocolo orientado a mensajes que detecta el fin del mensaje en el lado del receptor e indica al usuario los datos pertenecientes al mensaje. Esto no depende de la longitud de recepción especificada del mensaje. Por tanto, al transferir datos a través de una conexión UDP, se suministra información relacionada con la longitud y el final de un mensaje.

Si la longitud de los datos enviados no coincide con la de los datos esperados, se procede del siguiente modo:

- Longitud de los datos que deben recibirse (parámetro LEN, instrucción TURCV/TRCV_C) mayor que la longitud de los datos enviados (parámetro LEN, instrucción TUSEND/TSEND_C):
  TURCV/TRCV_C copia todos los datos enviados en el área de datos recibidos (parámetro DATA). Seguidamente, pone el parámetro de estado NDR a TRUE (petición concluida correctamente) y asigna la longitud de los datos enviados a RCVD_LEN (número de datos realmente recibidos).
- Longitud de los datos que deben recibirse (parámetro LEN, instrucción TURCV/TRCV_C) menor que la longitud de los datos enviados (parámetro LEN, instrucción TUSEND/TSEND_C):
  TURCV/TRCV_C copia tantos datos en el área de datos recibidos (parámetro DATA) como los solicitados en el parámetro LEN. No se generan más mensajes de error. En este caso, el usuario debe volver a llamar un T_URCV para recibir los bytes restantes.

Consulte también

Principios básicos de la Open User Communication (Página 696)

TSEND_C: Enviar datos a través de Ethernet (Página 5263)

TRCV_C: Recibir datos a través de Ethernet (Página 5274)

TSEND: Enviar datos a través de la conexión (Página 5310)

TRCV: Recibir datos a través de la conexión (Página 5314)

TUSEND: Enviar datos vía Ethernet (UDP) (Página 5323)

TURCV: Recibir datos vía Ethernet (UDP) (Página 5326)

Parámetros de conexión conforme a TCON_Param

Bloque de datos para la descripción de la conexión

Para parametrizar las conexiones en TCP, UDP e ISO on TCP, para algunas CPUs de la S7-1200 se utiliza un DB de descripción de la conexión con una estructura conforme a TCON_Param. La estructura de datos fija de TCON_Param contiene los parámetros necesarios para establecer la conexión. El DB de descripción de la conexión se crea automáticamente desde la parametrización de conexión de la Open User Communication cuando se utilizan las instrucciones TSEND_C, TRCV_C o TCON para una conexión nueva.

El parámetro de conexión CONNECT de los DBs de instancia para TSEND_C, TRCV_C o TCON contiene una referencia al bloque de datos utilizado.

Estructura de la descripción de la conexión conforme a TCON_Param

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	block_length	UINT	64	Longitud: 64 bytes (fija)
2 … 3	id	CONN_OUC	1	Referencia a esta conexión (rango de valores: de 1 a 4095). El valor de este parámetro debe indicarse en ID para la instrucción TSEND_C, TRCV_C o TCON.
4	connection_type	USINT	17	Tipo de conexión: • 17: TCP • 18: ISO-on-TCP • 19: UDP
5	active_est	BOOL	TRUE	Identificador del tipo de establecimiento de la conexión. En UDP siempre rige FALSE, porque es posible enviar o recibir datos mediante la ID local. Para TCP e ISO on TCP rige lo siguiente: • FALSE: establecimiento de conexión pasivo • TRUE: iniciativa local
6	local_device_id	USINT	1	Identificador de la interfaz local PN/IE.
7	local_tsap_id_len	USINT	0	Longitud utilizada del parámetro local_tsap_id en bytes; valores posibles: • 0 o 2, si el tipo de conexión = 17 (TCP) Para el lado activo solo se admite el valor 0. • de 2 a 16, si el tipo de conexión = 18 (ISO on TCP) • 2, si el tipo de conexión = 19 (UDP)
8	rem_subnet_id_len	USINT	0	Este parámetro no se utiliza.
9	rem_staddr_len	USINT	4	Longitud de la dirección del punto final del interlocutor en bytes: • 0: sin especificar, es decir, el parámetro rem_staddr es irrelevante. • 4: dirección IP válida en el parámetro rem_staddr (solo TCP e ISO on TCP)

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
10	rem_tsap_id_len	USINT	2	Longitud utilizada del parámetro rem_tsap_id en bytes; valores posibles: • 0 o 2, si el tipo de conexión = 17 (TCP) Para el lado pasivo solo se admite el valor 0. • de 2 a 16, si el tipo de conexión = 18 (ISO on TCP) • 0, si el tipo de conexión = 19 (UDP)
11	next_staddr_len	USINT	0	Este parámetro no se utiliza.
12 … 27	local_tsap_id	ARRAY [1..16] of BYTE	-	Componente de dirección local de la conexión: • TCP y UDP: n.º de puerto local (valores posibles: 1...49151; valores recomendados: 2000...5000); local_tsap_id[1] = byte High del n.º de puerto en representación hexadecimal; local_tsap_id[2] = byte Low del n.º de puerto en representación hexadecimal; local_tsap_id[3-16] = irrelevante • ISO on TCP: ID TSAP local: local_tsap_id[1] = B#16#E0; local_tsap_id[2] = rack y slot de los puntos finales locales (bits de 0 a 4: número de slot, bits de 5 a 7: número de rack); local_tsap_id[3-16] = extensión TSAP, opcional Nota: asegúrese de que todos los valores de local_tsap_id sean unívocos dentro de la CPU.
28 … 33	rem_subnet_id	ARRAY [1..6] of USINT	-	Este parámetro no se utiliza.
34 … 39	rem_staddr	ARRAY [1..6] of USINT	-	Solo TCP e ISO on TCP: Dirección IP del punto final del interlocutor, p. ej. para 192.168.002.003: • rem_staddr[1] = 192 • rem_staddr[2] = 168 • rem_staddr[3] = 002 • rem_staddr[4] = 003 • rem_staddr[5-6] = irrelevante
40 … 55	rem_tsap_id	ARRAY [1..16] of BYTE	-	Componente de dirección del interlocutor de la conexión • TCP: n.º de puerto interlocutor (valores posibles: 1...49151; valores recomendados: 2000...5000); rem_tsap_id[1] = byte alto del n.º de puerto en representación hexadecimal; rem_tsap_id[2] = byte bajo del n.º de puerto en representación hexadecimal; rem_tsap_id[3-16] = irrelevante • ISO on TCP: ID TSAP del interlocutor: rem_tsap_id[1] = B#16#E0; rem_tsap_id[2] = rack y slot del punto final del interlocutor (bits 0 a 4: número de slot, bits de 5 a 7: número de rack); rem_tsap_id[3-16] = extensión TSAP, opcional • UDP: Este parámetro no se utiliza.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
56 … 61	next_staddr	ARRAY [1..6] of BYTE	-	Este parámetro no se utiliza.
62 … 63	spare	WORD	W#16#0000	Reservado.

Nota

TCON_Param para la CPU S7-1500

El DB de descripción de la conexión con la estructura conforme a TCON_Param también es soportado por CPU S7-1500 por motivos de migración. Sin embargo, se recomienda utilizar las nuevas estructuras TCON_IP_v4 y TCON_IP_RFC.

Consulte también

Funcionamiento de los protocolos orientados a la conexión (Página 708)

Descripción de los parámetros de conexión (Página 700)

Capacidad de relectura de los parámetros para la descripción de la conexión (Página 717)

Principios básicos de la programación de bloques de datos (Página 2937)

Resumen de la parametrización de la conexión (Página 698)

Estructura TSAP (Página 718)

Asignación de números de puerto (Página 716)

Parámetros de conexión conforme a TCON_IP_v4

Bloque de datos para la descripción de la conexión

Para parametrizar las conexiones en TCP y UDP, para las CPU S7-1200 a partir de V4.0 y S7-1500 se utiliza un DB de descripción de la conexión con una estructura conforme a TCON_IP_v4. La estructura de datos fija de TCON_IP_v4 contiene los parámetros necesarios para establecer la conexión. El DB de descripción de la conexión se crea automáticamente desde la parametrización de conexión de la Open User Communication cuando se utilizan las instrucciones TSEND_C, TRCV_C o TCON para una conexión nueva.

El parámetro de conexión CONNECT de los DBs de instancia para TSEND_C, TRCV_C o TCON contiene una referencia al bloque de datos utilizado.

Estructura de la descripción de la conexión conforme a TCON_IP_v4

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	Interface_id	HW_ANY	64	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	id	CONN_OUC	1	Referencia a esta conexión (rango de valores: de 1 a 4095). El valor de este parámetro debe indicarse en ID para la instrucción TSEND_C, TRCV_C o TCON.
4	connection_type	BYTE	11	Tipo de conexión: • 11: TCP • 13: UDP
5	active_established	BOOL	TRUE	Identificador del tipo de establecimiento de conexión: • FALSE: establecimiento de conexión pasivo • TRUE: iniciativa local
6 … 9	remote_address	ARRAY [1..4] of BYTE	-	Dirección IP del punto final del interlocutor, p. ej. para 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1
10 … 11	remote_port	UINT	2000	Dirección de puerto del interlocutor remoto (rango de valores:. de 1 a 49151).
12 … 13	local_port	UINT	2000	Dirección de puerto del interlocutor local (rango de valores:. de 1 a 49151).

Consulte también

Funcionamiento de los protocolos orientados a la conexión (Página 708)

Descripción de los parámetros de conexión (Página 700)

Capacidad de relectura de los parámetros para la descripción de la conexión (Página 717)

Principios básicos de la programación de bloques de datos (Página 2937)

Resumen de la parametrización de la conexión (Página 698)

Asignación de números de puerto (Página 716)

Parámetros de conexión conforme a TCON_IP_RFC

Bloque de datos para la descripción de la conexión

Para parametrizar las conexiones en ISO on TCP, paralas CPU S7-1200 a partir de V4.0 y S7-1500 se utiliza un DB de descripción de la conexión con una estructura conforme a TCON_IP_RFC. La estructura de datos fija de TCON_IP_RFC contiene los parámetros necesarios para establecer la conexión. El DB de descripción de la conexión se crea

automáticamente desde la parametrización de conexión de la Open User Communication cuando se utilizan las instrucciones TSEND_C, TRCV_C o TCON para una conexión nueva.

El parámetro de conexión CONNECT de los DBs de instancia para TSEND_C, TRCV_C o TCON contiene una referencia al bloque de datos utilizado.

Estructura de la descripción de la conexión conforme a TCON_IP_RFC

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	Interface_id	HW_ANY	64	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	id	CONN_OUC	1	Referencia a esta conexión (rango de valores: de 1 a 4095). El valor de este parámetro debe indicarse en ID para la instrucción TSEND_C, TRCV_C o TCON.
4	connection_type	BYTE	12	Tipo de conexión: ISO-on-TCP
5	active_established	BOOL	TRUE	Identificador del tipo de establecimiento de conexión: • FALSE: establecimiento de conexión pasivo • TRUE: iniciativa local
8 … 11	remote_address	ARRAY [1..4] of BYTE	-	Dirección IP del punto final del interlocutor, p. ej. para 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1
12 … 45	remote_tselector	TSelector	-	TSelector del interlocutor remoto: • TSelLength = rango de valores de 0 a 32 como UINT • TSel[1-32] = rango de valores de 0 a 255 en bytes
46 … 79	local_tselector	TSelector	-	TSelector del interlocutor local: • TSelLength = rango de valores de 0 a 32 como UINT • TSel[1-32] = rango de valores de 0 a 255 en bytes

Consulte también

Funcionamiento de los protocolos orientados a la conexión (Página 708)

Descripción de los parámetros de conexión (Página 700)

Capacidad de relectura de los parámetros para la descripción de la conexión (Página 717)

Principios básicos de la programación de bloques de datos (Página 2937)

Resumen de la parametrización de la conexión (Página 698)

Estructura TSAP (Página 718)

Asignación de números de puerto

Introducción

Al crear una Open User Communication se asigna automáticamente el valor 2000 como número de puerto.

Los números de puerto admiten valores de 1 a 49151. Dentro de este rango, son libremente asignables. Sin embargo, puesto que algunos puertos ya son utilizados por el sistema, es recomendable utilizar números de puerto en un rango de 2000 a 5000.

Nota

Los números de puerto deben ser unívocos. Si se produjera doble asignación en los números de puerto, la configuración de la conexión o una correspondiente llamada de bloque se rechazarán con un error.

Resumen de los números de puerto

La tabla siguiente muestra un resumen de las reacciones del sistema a diferentes números de puerto.

N.º de puerto	Descripción	Reacción del sistema
2000 … 5000	Rango recomendado	Ninguna advertencia, ningún mensaje de error a la entrada El número de puerto se admite y aplica
1 … 1999, 5001 … 49151	Utilizable pero fuera del rango recomendado	Advertencia a la entrada El número de puerto se admite y aplica
0, 20, 21, 25, 80, 102, 135, 161, 34962 … 34964	Utilizable con limitaciones*	
53, 80, 102, 135, 161, 162, 443, 520, 9001, 34962 … 34964	Utilizable con limitaciones**	

* Puertos definidos para determinadas funciones:
0: ANY - el número de puerto es asignado automáticamente por la CPU S7-1500 como valor (>49151)
20: Transferencia de datos vía FTP
21: Control FTP
25: TMAIL_C (Simple Mail transfer Protocol)
80: Servidor web
102: ISO on TCP (RFC1006)
135: DCE Endpoint Mapper para PROFINET

161: SNMP (Simple Network Management Protocol)
34962 ... 34964: PROFINET

Nota

Para UDP/TCP, el usuario normalmente especifica en el punto final activo de la conexión el valor 0 para el puerto local. En este caso, el sistema operativo de la CPU selecciona el siguiente puerto libre por encima de 49151. Inversamente, en el punto final pasivo de la conexión, el puerto del interlocutor suele recibir 0. En la parametrización de la conexión se desactiva el correspondiente parámetro.

** Estos puertos están bloqueados según sea el volumen de funciones de la CPU utilizada. La ocupación de dichos puertos se desprende de la documentación de las CPUs correspondientes.

Consulte también

Descripción de los parámetros de conexión (Página 700)

Crear y parametrizar conexiones (Página 704)

Capacidad de relectura de los parámetros para la descripción de la conexión

Modificar los valores de parámetros en la descripción de la conexión

La parametrización de la conexión registra la descripción de una sola conexión de la Open User Communication en el DB de descripción de la conexión.

Existe la posibilidad de modificar los valores de los parámetros del DB de descripción de la conexión en el programa de usuario, fuera de la parametrización de la conexión. La parametrización de la conexión puede releer los DBs de descripción de la conexión que contienen los valores modificados con posterioridad por el usuario. En "Propiedades > Configuración > Parámetros de conexión" de la ventana de inspección se muestran solo los parámetros de conexión guardados en el DB de descripción de la conexión.

Nota

Mientras se ejecuta el programa de usuario, solo está permitido modificar valores si no se están procesando las instrucciones TCON, TSEND_C o TRCV_C, o si la conexión referenciada no se ha establecido.

Las descripciones de la conexión registradas con anidamiento en tipos de DB que solo se encuentran mediante referenciación de offset (p. ej. DB global) no son soportadas por la parametrización de la conexión.

La estructura de la descripción de la conexión no puede modificarse.

Capacidad de relectura de los diferentes parámetros de conexión

En el parámetro "Dirección" del interlocutor en una conexión TCP o ISO on TCP se muestra su dirección IP desde el parámetro "rem_staddr" de la descripción de la conexión.

Asimismo, los valores siguientes pueden volver a cargarse desde la descripción de la conexión:

- Tipo de conexión
- ID de conexión local
- Establecimiento de conexión activo/pasivo (no en UDP)
- TSAP local (solo con ISO on TCP)
- TSAP del interlocutor (solo con ISO on TCP)
- Puerto local (solo con TCP y UDP)
- Puerto interlocutor (solo con TCP)

Los valores de los parámetros de la ID de conexión del interlocutor, de los datos de conexión así como del establecimiento de la conexión no forman parte de la descripción del DB de descripción de la conexión local. Así pues, no es posible visualizar dichos parámetros cuando se vuelve a abrir la parametrización de la conexión. Sin embargo, el establecimiento de conexión del interlocutor resulta del establecimiento de conexión local, por lo que sí se visualiza.

La lista desplegable "Interlocutor" permite seleccionar un interlocutor nuevo en cualquier momento.

Si se selecciona una CPU detectada en el proyecto como interlocutor especificado, se mostrarán nuevamente las posibilidades de entrada de la ID de conexión y los datos de la misma.

Consulte también

Parámetros de conexión conforme a TCON_Param (Página 711)

Descripción de los parámetros de conexión (Página 700)

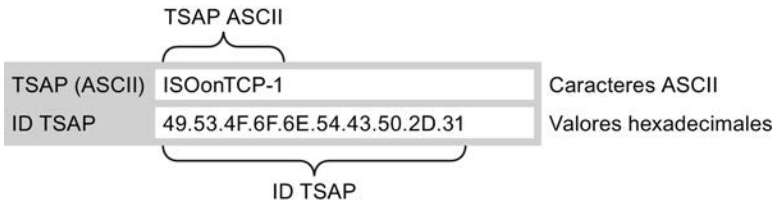
Estructura TSAP

Introducción

En una conexión del tipo ISO on TCP hay que asignar Transportation Service Access Points (TSAPs) a ambos interlocutores. Las IDs TSAP se asignan automáticamente después de crear una conexión ISO on TCP. Para garantizar la univocidad de las IDs TSAP dentro de un dispositivo, es posible modificar los TSAPs asignados en la parametrización de la conexión.

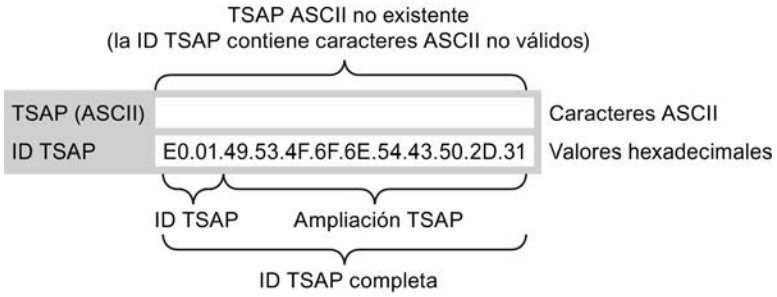
Estructura de los TSAPs

Para asignar libremente TSAPs hay que observar ciertas reglas. Un TSAP debe incluir un número determinado de bytes que se visualizan e introducen en la parametrización de la conexión en forma de valores hexadecimales (ID TSAP) o de caracteres ASCII (TSAP ASCII):



Las entradas o modificaciones de las ID TSAP o TSAP ASCII en los campos de entrada correspondientes repercuten siempre en el otro formato de visualización respectivo.

Si un TSAP no contiene ningún carácter ASCII válido, sólo se visualizará como ID TSAP, no como TSAP ASCII. Esto sucede después de crear una conexión. Los dos primeros caracteres hexadecimales como ID TSAP identifican el tipo de comunicación y el rack/slot. Puesto que estos caracteres no son caracteres ASCII válidos para una CPU, en este caso no se mostrará el TSAP ASCII:



Además de las reglas de longitud y estructura de los TSAPs hay que garantizar la univocidad de las ID TSAP. Los TSAPs no son unívocos automáticamente.

Longitud y contenido de los TSAPs

Un TSAP tiene la estructura siguiente:

- ID TSAP con extensión TSAP
  Longitud = de 2 a 16 bytes
  x_tsap_id[0] = 0xE0 (Open User Communication)
  x_tsap_id[1] (bits de 0 a 4) = número de slot de la CPU
  x_tsap_id[1] (bits de 5 a 7) = número de rack de la CPU
  x_tsap_id[2...15] = caracteres arbitrarios (extensión TSAP, opcional)
  (x = loc (local) o x = rem (interlocutor))
- ID TSAP como TSAP ASCII
  Longitud = de 3 a 16 bytes
  x_tsap_id[de 0 a 2] = 3 caracteres ASCII (de 0x20 a 0x7E)
  x_tsap_id[3...15] = caracteres arbitrarios (opcional)
  (x = loc (local) o x = rem (interlocutor))

La tabla siguiente muestra la estructura esquemática de una ID TSAP:

ID TSAP	tsap_id_len	tsap_id[0]	tsap_id[1]	tsap_id[2..15]	tsap_id[3..15]
...con extensión	2...16 bytes	0xE0	0x01 o 0x02 o 0x00*	Extensión (opcional)	Extensión (opcional)
...como TSAP ASCII	3...16 bytes	0x20...0x7E	0x20...0x7E	0x20...0x7	arbitrario (opcional)

* Una CPU S7-1200/1500 suele estar insertada en el rack 0 y el slot 1, y una CPU de la S7-300/400 en el rack 0 y el slot 2. Por consiguiente, la segunda cifra de la ID de TSAP con extensión es el valor hexadecimal 01 ó 02. Si el interlocutor es una CPU sin especificar, p. ej. un dispositivo de terceros, también se admite el valor hexadecimal 00 como dirección del slot.

Nota

Si los interlocutores están sin especificar, la ID TSAP local y del interlocutor admiten una longitud de 0 a 16 bytes, permitiéndose todos los valores hexadecimales de 00 a FF.

Tabla de códigos ASCII para introducir TSAPs ASCII

Para introducir un TSAP ASCII en la parametrización de la conexión sólo se admiten los valores hexadecimales de 20 a 7E:

Código	..0	..1	..2	..3	..4	..5	..6	..7	..8	..9	..A	..B	..C	..D	..E	..F
2..		!	"	#	$	%	&	’	(	)	*	+	,	-	.	/
3..	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4..	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5..	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6..	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7..	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	

Consulte también

Ejemplos de asignación TSAP (Página 720)

Descripción de los parámetros de conexión (Página 700)

Crear y parametrizar conexiones (Página 704)

Ejemplos de asignación TSAP

Los ejemplos siguientes muestran la ejecución de los TSAP para CPUs S7-1200/1500 (CPU en el slot 1) desde criterios diferentes:

- Ejemplo 1: crear una conexión nueva para una comunicación PLC-PLC
- Ejemplo 2: entrada de un TSAP ASCII local
- Ejemplo 3: entrada de una extensión TSAP en la ID TSAP

- Ejemplo 4: edición errónea de la ID TSAP
- Ejemplo 5: entrada de un TSAP ASCII desde el campo de entrada "ID TSAP"

Ejemplo 1: crear una conexión nueva para una comunicación PLC-PLC

Después de haber creado una conexión nueva con dos PLCs para la Open User Communication, se asigna automáticamente la extensión TSAP "ISOonTCP-1".

Dicha extensión resulta de la ID TSAP E0.01.49.53.4F.6F.6E.54.43.50.2D.31, que se registra automáticamente en el DB de descripción de la conexión y en los campos de entrada del TSAP local y del interlocutor. Los campos de entrada de los TSAPs ASCII quedan vacíos:

	TSAP local	TSAP del interlocutor
TSAP (ASCII)		
ID TSAP	E0.01.49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Es posible cambiar en cualquier momento los valores de los campos de entrada de la ID TSAP y del TSAP ASCII.

El campo de entrada de la ID TSAP muestra todo el TSAP guardado en el bloque de datos de la descripción de la conexión. La ID TSAP con extensión TSAP, limitada a 16 caracteres, no se muestra en el campo de entrada "TSAP (ASCII)" ya que el carácter E0 no representa ningún carácter válido para el TSAP ASCII.

Si la ID TSAP visualizada es un TSAP ASCII válido, se mostrará en el campo de entrada "TSAP (ASCII)".

Los cambios en los campos de entrada de ID TSAP y TSAP ASCII se influyen mutuamente.

Ejemplo 2: entrada de un TSAP ASCII local

Si se ha creado una conexión nueva y se asigna un valor ASCII al TSAP local en el campo de entrada "TSAP (ASCII)", p. ej. "ISOonTCP-1", la ID TSAP resultante se generará automáticamente.

Al salir del campo de entrada "TSAP (ASCII)" se comprobará automáticamente el mantenimiento de la limitación de caracteres ASCII de 3 a 16 y se registrará la ID TSAP resultante en el campo de entrada correspondiente:

	TSAP local	TSAP del interlocutor
TSAP (ASCII)	ISOonTCP-1	
ID TSAP	49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Ejemplo 3: entrada de una extensión TSAP en la ID TSAP

Si después de crear una conexión e introducir un TSAP ASCII (véanse los ejemplos 1 y 2) en el campo de entrada de la ID TSAP local se anteponen los caracteres "E0.01." al valor TSAP, una vez se salga del campo de entrada ya no se visualizará ningún TSAP ASCII:

	TSAP local	TSAP del interlocutor
TSAP (ASCII)		
ID TSAP	E0.01.49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Después de salir del campo de entrada de la ID TSAP se comprobará automáticamente si el primer carácter de la ID TSAP es un carácter ASCII válido. Puesto que el carácter "E0" de la ID TSAP no es válido para el TSAP ASCII, ya no se visualizará ningún TSAP ASCII en el campo de entrada "TSAP (ASCII)".

Si se utilizan caracteres ASCII válidos, se comprueba que se mantenga una longitud entre 2 y 16 caracteres.

Ejemplo 4: edición errónea de la ID TSAP

Si se elimina el valor hexadecimal "E0" de una ID TSAP que empieza por "E0.01", la ID TSAP empezará ahora por "01" y ya no será conforme a las reglas, por lo que no será válida:

	TSAP local	TSAP del interlocutor
TSAP (ASCII)		
ID TSAP	01.49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Después de salir del campo de entrada se emite un aviso, ya que la ID TSAP no es ni un TSAP ASCII válido (para ello debería tener un primer valor hexadecimal en un rango de 20 a 7E) ni una ID TSAP válida (para ello el primer valor debería tener la identificación "E0").

Ejemplo 5: entrada de un TSAP ASCII desde el campo de entrada "ID TSAP"

Si en la ID TSAP errónea del ejemplo 4 tras eliminar el valor "E0" se elimina también el valor "01", la ID TSAP empezará por el valor hexadecimal 49. Dicho valor está dentro del rango permitido para TSAPs ASCII:

	TSAP local	TSAP del interlocutor
TSAP (ASCII)		
ID TSAP	49.53.4F.6F.6E.54.43.50.2D.31	E0.01.49.53.4F.6F.6E.54.43.50.2D.31

Al salir del campo de entrada, la ID TSAP se detectará como TSAP ASCII válido y el TSAP ASCII resultante "ISOonTCP-1" se escribirá en el campo de entrada "TSAP (ASCII)".

Consulte también

Estructura TSAP (Página 718)

Descripción de los parámetros de conexión (Página 700)

Comunicación mediante instrucciones PUT y GET

Principios básicos de la comunicación mediante la instrucción PUT/GET

Principios básicos de las instrucciones PUT/GET

Para intercambiar datos entre dos CPUs a través de una conexión S7, utilice las instrucciones PUT y GET.

La instrucción GET permite leer datos de una CPU interlocutora. Con la instrucción PUT se puede controlar la escritura de variables en el interlocutor a través del programa de usuario. Además de las instrucciones PUT y GET, no se ha previsto ninguna otra función de comunicación para la lectura y escritura de variables.

Para facilitar la utilización de ambas instrucciones, defina todos los parámetros necesarios para la conexión, así como todos los parámetros de bloque en la ventana de inspección del editor de programación.

Requisitos

Para poder utilizar las instrucciones PUT y GET, es necesario que se cumplan los siguientes requisitos.

- Se debe haber incluido como mínimo una CPU S7-1200/1500 o S7-300/400 en el proyecto. En una CPU S7-1200 debe estar instalado el firmware 2.0 o superior. Si aún no se ha incluido una segunda CPU en el proyecto, se puede establecer primero la conexión con un interlocutor no especificado.
- Debe existir una conexión S7 entre ambas CPUs. Si aún no ha creado ninguna conexión entre dos CPUs, se establecerá automáticamente una conexión durante la configuración de las instrucciones.
- Para ambas instrucciones se requiere un bloque de datos de instancia, en el que se almacenan todos los datos utilizados por la instrucción. El bloque de datos de instancia se crea automáticamente en el momento en el que la instrucción PUT o GET se arrastra a un segmento en el editor de programación. Para la correcta ejecución del programa, los bloques de datos de instancia no pueden modificarse, por lo que tienen protección de know how. El usuario sólo tiene acceso de lectura a los bloques de datos de instancia.

Consulte también

Resumen de la parametrización de la conexión (Página 724)

Parametrizar el tipo de petición (Página 730)

PUT: Parametrizar área de escritura y transmisión (Página 730)

GET: Parametrizar un área de lectura y memoria (Página 731)

Parametrización de la conexión

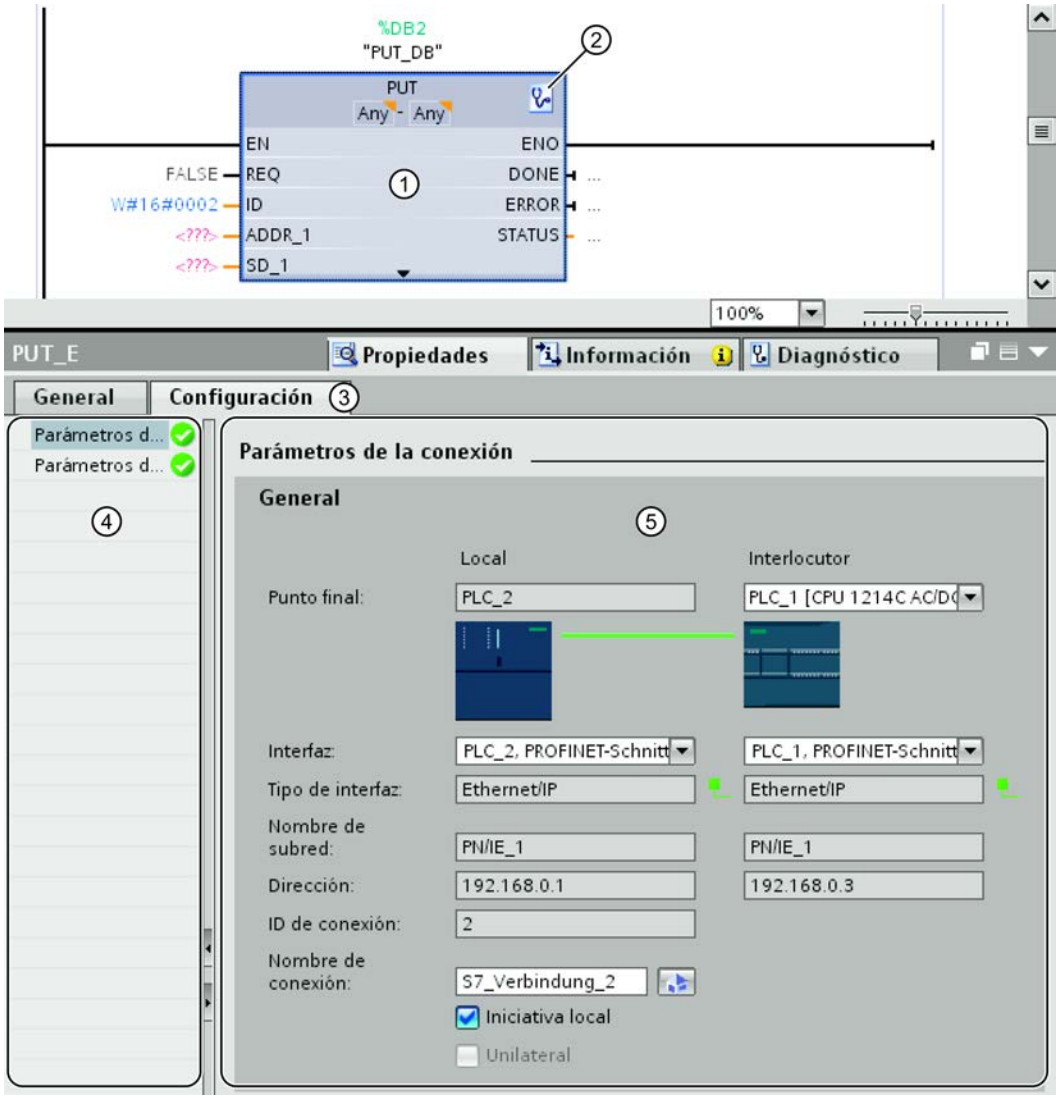
Resumen de la parametrización de la conexión

Introducción

La parametrización de la conexión para las instrucciones PUT y GET se realiza en la ventana de inspección del editor de programación. Todos los parámetros se almacenan en el bloque de datos de instancia correspondiente.

Estructura de la parametrización de la conexión

La parametrización de la conexión incluye los componentes siguientes:



① Instrucción de comunicación para PUT o GET
② Llamada a funciones online y de diagnóstico
③ Ficha "Configuración" en la ficha "Propiedades"
④ Navegación local de la ficha "Configuración"
⑤ Propiedades generales de los parámetros de conexión

Visualización de funciones online y de diagnóstico

Haciendo clic en el icono de arranque de las funciones online y de diagnóstico, la CPU correspondiente se conecta online automáticamente. En la vista de redes se abre la tabla de conexiones. Además, en la ventana de inspección se muestran la ficha "Diagnóstico" y los datos de la conexión.

Introducción de los parámetros de conexión

Introduzca los parámetros de conexión deseados en la ficha "Configuración". Encontrará el grupo "Parámetros de la conexión" en la navegación local de la ficha "Configuración". Dicho grupo contiene la parametrización de la conexión. Aquí se pueden introducir los parámetros de las conexiones con asistencia del sistema. Cuando todos los parámetros necesarios están disponibles, se activa una marca de verificación después del grupo "Parámetros de la conexión" en la navegación local.

Consulte también

Parametrizar el tipo de petición (Página 730)

PUT: Parametrizar área de escritura y transmisión (Página 730)

GET: Parametrizar un área de lectura y memoria (Página 731)

Descripción de los parámetros de conexión

Resumen

La tabla siguiente muestra los parámetros de conexión generales:

Parámetro	Descripción
Punto final	Se muestra el nombre del punto final local y del interlocutor. • Punto final local El punto final local es la CPU en la que se programa la instrucción PUT/GET. • Punto final del interlocutor El punto final del interlocutor se selecciona en una lista desplegable. Dicha lista muestra todos los interlocutores disponibles, entre ellos también interlocutores sin especificar para dispositivos cuyos datos son desconocidos dentro del proyecto. Mientras no se haya ajustado ningún interlocutor, todos los demás parámetros de la máscara permanecerán desactivados.
Interfaz	Se muestra la interfaz de la CPU interlocutora. La interfaz del interlocutor no se visualiza hasta que se selecciona una CPU interlocutora especificada.
Tipo de interfaz	Se visualiza el tipo de interfaz a través del cual tendrá lugar la comunicación.
Nombre de la subred	Se muestra la subred del punto final local, si existe. La subred del interlocutor no se visualiza hasta que se selecciona el punto final del interlocutor. Si como mínimo uno de los dos interlocutores no está conectado a una subred, se conectarán automáticamente en red los dos interlocutores. En este caso, el interlocutor que no está conectado en red se conectará a la misma subred a través de la cual el otro interlocutor se encuentra conectado a una red. Solo es posible conectar interlocutores a subredes distintas a través de un routing IP o S7. La configuración del routing IP puede modificarse en las propiedades de las interfaces Ethernet en cuestión.
Dirección	Se muestra la dirección IP del punto final local. La dirección IP del interlocutor no se visualiza hasta que se selecciona el punto final del interlocutor. Si se ha seleccionado un interlocutor no especificado, el campo de entrada estará vacío y resaltado en rojo. En este caso, hay que indicar una dirección IP válida para el interlocutor.

Parámetro	Descripción
ID de conexión	Se visualiza la ID de conexión actualmente ajustada. La ID de conexión se puede modificar en la tabla de conexiones, dentro de la vista de redes. Además, durante la parametrización de la conexión se puede acceder directamente a la tabla de conexiones. Para ello, haga clic en el icono "Crear conexión".
Nombre de conexión	Se visualiza el nombre de la conexión creada automáticamente al introducir la instrucción PUT/GET. El nombre de la conexión se puede modificar introduciendo otro nombre en el campo. Además es posible crear una conexión o editar conexiones ya existentes haciendo clic en el icono "Crear conexión".
Iniciativa local	Determine mediante el botón de opción "Establecimiento activo de la conexión" el interlocutor que debe iniciar la comunicación. Al crear la conexión, primero se determina el interlocutor local para el establecimiento de la conexión de forma automática. Si un dispositivo no soporta el establecimiento activo de la conexión, será necesario activar el establecimiento activo de la conexión en el otro interlocutor.
Unilateral	Si la casilla de verificación está activada, el interlocutor actúa como servidor para esta conexión. No puede enviar ni recibir de forma activa. Se corresponde con el comportamiento de las instrucciones PUT/GET. En este caso, no son posibles otras instrucciones. Si la casilla de verificación está desactivada, es también posible utilizar otras instrucciones para la comunicación.

Iniciar la parametrización de la conexión

La conexión para PUT y GET se puede parametrizar en la ventana de inspección en cuanto se ha insertado una instrucción PUT/GET en un bloque de programa y se ha seleccionado.

Procedimiento

Para insertar instrucciones PUT/GET, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones", carpeta "Comunicación > Comunicación S7".
2. Arrastre la instrucción PUT/GET a un segmento con la función Drag & Drop:
   Se abre el cuadro de diálogo "Opciones de llamada".
3. Opcionalmente: Edite las propiedades del DB instancia en el cuadro de diálogo "Opciones de llamada". Existen las posibilidades siguientes:
   – Modificar el nombre predeterminado.
   – Seleccionar el campo de opción "manual" para asignar un número propio.
4. Haga clic en "Aceptar".

Resultado

Para la instrucción insertada PUT o GET se crea un bloque de datos de instancia correspondiente. En el caso de las CPUs S7-300, se crea además un bloque de función en los recursos del programa.

Si hay una instrucción PUT/GET seleccionada, en la ventana de inspección se puede ver la ficha "Configuración" en "Propiedades". El grupo "Parámetros de la conexión" de la navegación local contiene la parametrización de la conexión que debe realizarse ahora.

Consulte también

Crear y parametrizar conexiones (Página 728)

Borrar conexiones (Página 729)

Crear y parametrizar conexiones

En la parametrización de conexión de las instrucciones PUT y GET es posible crear y parametrizar las conexiones S7. La parametrización de conexión comprueba inmediatamente si los valores modificados presentan errores de entrada.

Requisitos

Existe una CPU con una instrucción de comunicación PUT o GET.

Procedimiento

Para configurar una conexión S7 mediante instrucciones PUT/GET, proceda del siguiente modo:

1. En el editor de programación, seleccione la llamada de la instrucción PUT o GET.
2. En la ventana de inspección abra la ficha "Propiedades > Configuración".
3. Seleccione el grupo "Parámetros de la conexión". Mientras no se haya seleccionado ningún interlocutor, sólo estará activa la lista desplegable vacía del punto final del interlocutor. Todas las demás entradas posibles estarán desactivadas.
   Se mostrarán los parámetros de conexión que ya se conocen:
   - Nombre del punto final local
   - Interfaz del punto final local
   - Dirección IP del punto final local

4. Seleccione un interlocutor en la lista desplegable del punto final del interlocutor. El interlocutor puede ser un dispositivo sin especificar o una CPU del proyecto.
   Los siguientes parámetros se introducen automáticamente en cuanto se selecciona el interlocutor:
   - Nombre del punto final del interlocutor
   - Interfaz del punto final del interlocutor. Si se dispone de varias interfaces, se puede cambiar de interfaz en caso necesario.
   - Tipo de interfaz del punto final del interlocutor
   - Nombre de la subred de ambos puntos finales
   - Dirección IP del punto final del interlocutor
   - Nombre de la conexión que se utiliza para la comunicación. Si no existe ninguna conexión, se creará una de forma automática.
5. En caso necesario, cambie el nombre de conexión en el campo de entrada "Nombre de conexión". Si desea crear una conexión o editar una conexión ya existente, haga clic en el icono "Crear conexión".

Nota

Las instrucciones PUT y GET entre dos interlocutores no estarán listas hasta que se haya cargado en el hardware tanto la configuración de hardware como la parte del programa destinada al punto final del interlocutor. Para que la comunicación funcione, asegúrese de no cargar en el dispositivo únicamente la descripción de la conexión de la CPU local, sino también la de la CPU partner.

Borrar conexiones

Una conexión que se ha creado automáticamente al insertar una instrucción PUT/GET aparece, como cualquier otra conexión, en la tabla de conexiones de la vista de redes. De ese modo es posible borrarla en la tabla de conexiones.

Procedimiento

Para borrar una conexión, proceda del siguiente modo:

1. Abra la tabla de conexiones en la vista de redes.
2. En la tabla la conexiones, seleccione la conexión que desea borrar.
3. Haga clic en la conexión con el botón derecho del ratón y elija el comando "Borrar" en el menú contextual.

Resultado

La conexión se ha borrado. La instrucción PUT/GET y los bloques de datos de instancia correspondientes se conservan y, en caso necesario, deben borrarse manualmente.

Para continuar utilizando la instrucción PUT/GET hay que volver a configurar la conexión en la ventana de inspección del editor de programación, ya que al borrar la conexión también se

borran todos los parámetros correspondientes. En ese caso, especifique un nuevo interlocutor y una conexión adecuada.

Parametrización de bloques

Parametrizar el tipo de petición

Para iniciar una comunicación a través de la instrucción PUT/GET, hay que definir un evento que active la instrucción. Este evento se denomina parámetro de control (REQ). La petición de comunicación se activa con un flanco positivo en el parámetro de control REQ.

Recuerde que el parámetro de control REQ tiene asignado FALSE en la primera llamada.

Requisitos

- El editor de programación está abierto.
- Ya ha insertado una instrucción PUT/GET.
- Existe una conexión entre dos interlocutores.

Procedimiento

Para definir el parámetro de control REQ, proceda del siguiente modo:

1. Seleccione la instrucción PUT/GET en el editor de programación.
2. Abra la ficha "Configuración" en la ventana de inspección.
3. Seleccione la entrada "Parametrización de bloques" en la navegación local.
4. En el campo "REQ", seleccione una variable del tipo de datos "BOOL" para inicializar la ejecución de la instrucción. También es posible interconectar una instrucción previa en el editor de programación.

Consulte también

Coherencia de datos (Página 5244)

PUT: Parametrizar área de escritura y transmisión (Página 730)

GET: Parametrizar un área de lectura y memoria (Página 731)

PUT: Parametrizar área de escritura y transmisión

En la comunicación con la instrucción PUT hay que especificar en qué área de memoria de la CPU interlocutora deben escribirse los datos. Además hay que especificar el área de memoria de la CPU local de la cual deben leerse los datos.

Requisitos

- El editor de programación está abierto.
- Ya ha insertado una instrucción PUT.
- Existe una conexión entre dos interlocutores.

Procedimiento

Para definir el área de lectura y memoria de la instrucción, proceda del siguiente modo:

1. Seleccione la instrucción PUT en el editor de programación.
2. Abra la ficha "Configuración" en la ventana de inspección.
3. Seleccione la entrada "Parametrización de bloques" en la navegación local.
4. En el campo "Entradas/salidas > Área de escritura (ADDR_1) > Inicio" seleccione un puntero con el tipo de datos "REMOTE" al área de la CPU interlocutora que debe escribirse. Sólo se admite el direccionamiento absoluto.
   Ejemplo: P#DB10.DBX5.0 byte 10
5. En el campo "Longitud", indique la longitud del área de escritura y seleccione el tipo de datos del área de memoria en la lista desplegable.
6. En el campo "Entradas/salidas > Área de transmisión (SD_1) > Inicio", seleccione un puntero al área de la CPU local que contiene los datos que deben enviarse.
7. Indique en el campo "Longitud" la longitud del área de memoria que debe leerse y seleccione el tipo de datos en la lista desplegable.
   Sólo se permiten los tipos de datos BOOL (en un campo de bits debe indicarse como dirección "0" y como longitud un múltiplo entero de Byte), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER.
   Si el puntero VARIANT accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 Byte 10).

Consulte también

GET: Parametrizar un área de lectura y memoria (Página 731)

GET: Parametrizar un área de lectura y memoria

En la comunicación con la instrucción GET hay que especificar en qué área de memoria de la CPU local deben escribirse los datos. Además hay que definir el área de lectura de la CPU interlocutora de la cual deben leerse los datos.

Requisitos

- El editor de programación está abierto.
- Ya ha insertado una instrucción GET.
- Existe una conexión entre dos interlocutores.

Procedimiento

Para definir el área de lectura y memoria de la instrucción, proceda del siguiente modo:

1. Seleccione la instrucción GET en el editor de programación.
2. Abra la ficha "Configuración" de la ventana de inspección.
3. Seleccione la entrada "Parametrización de bloques" en la navegación local.
4. En el campo "Entradas/salidas > Área de lectura (ADDR_1) > Inicio" seleccione un puntero con el tipo de datos "REMOTE" al área de la CPU interlocutora que debe leerse.
   Sólo se admite el direccionamiento absoluto.
   Ejemplo: P#DB10.DBX5.0 byte 10
5. En el campo "Longitud", indique la longitud del área de lectura y seleccione el tipo de datos del área de memoria en la lista desplegable.
6. En el campo "Entradas/salidas > Área de memoria (RD_1) > Inicio", seleccione un puntero al área de la CPU local en la que deben depositarse los datos leídos.
7. Indique en el campo "Longitud" la longitud del área de memoria y seleccione el tipo de datos en la lista desplegable.
   Sólo se permiten los tipos de datos BOOL (en un campo de bits debe indicarse como dirección "0" y como longitud un múltiplo entero de Byte), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER.

Consulte también

PUT: Parametrizar área de escritura y transmisión (Página 730)

8.1.3.3 Mostrar y configurar topología

Sinopsis de la vista topológica

Funciones de la vista topológica

La vista topológica es una de las tres áreas de trabajo del editor de hardware y redes. Aquí se realizan las tareas siguientes:

- Mostrar la topología de Ethernet
  - Mostrar todos los dispositivos PROFINET y los componentes Ethernet pasivos del proyecto con puertos
  - Mostrar las interconexiones entre los puertos
  - Mostrar las respectivas redes lógicas
  - Mostrar la información de diagnóstico de todos los puertos
- Configurar la topología Ethernet
  - Crear, modificar y borrar las interconexiones de los puertos
  - Cambiar el nombre de estaciones, dispositivos, interfaces y puertos
  - Agregar los dispositivos PROFINET y los componentes pasivos Ethernet del catálogo de hardware al proyecto
- Determinar y minimizar diferencias entre la topología prevista y la topología real
  - Comparación offline/online de los módulos, puertos e interconexiones de puertos Ethernet
  - Aplicar la información de topología disponible online al proyecto offline

---

Nota

Dispositivos sin dirección IP válida

Sin una dirección IP válida no es posible obtener información de la topología (LLDP) de un dispositivo.

Para evitar que haya dispositivos sin dirección IP válida, en los ajustes de la configuración hardware del TIA Portal puede especificarse que se asigne una dirección IP temporal a los dispositivos que carezcan de IP válida.

Diferencias entre la vista de redes y la vista topológica

- La vista de redes muestra todas las subredes lógicas del proyecto.
  La vista topológica muestra todos los componentes Ethernet del proyecto. Entre ellos, los componentes pasivos como switches, convertidores de medio y líneas.

  Nota

  Además, se muestran estaciones con componentes no Ethernet si en la estación se encuentra por lo menos un componente Ethernet.

- La posición de un dispositivo en la vista de redes y su posición en la vista topológica son independientes la una de la otra, es decir, normalmente se encuentra el mismo dispositivo en las dos vistas en otro lugar.
- Si abre el catálogo de hardware desde la vista topológica, aparecerán sólo los dispositivos con interfaz Ethernet.

Estructura de la vista topológica

La vista topológica (Página 496) se compone básicamente de un área gráfica (en lo sucesivo denominada vista gráfica) y un área de tabla (en lo sucesivo denominada vista de tabla).

¿Qué funciones existen en la vista gráfica y en la vista de tabla?

- Mostrar la topología de Ethernet

Función	Vista gráfica	Vista de tabla
Mostrar todos los dispositivos PROFINET y los componentes Ethernet pasivos del proyecto con puertos	sí	sí
Mostrar interconexiones entre los puertos (incluido el tipo de medio)	sí	sí
Mostrar las respectivas redes lógicas	no	sí
Mostrar las propiedades de las líneas entre los puertos	no	sí
Mostrar la información de diagnóstico de todos los puertos	sí	sí

- Configurar la topología Ethernet

Función	Vista gráfica	Vista de tabla
Crear, modificar y borrar las interconexiones de los puertos	• Crear: sí • Modificar: no • Borrar: sí	• Crear: sí • Modificar: sí • Borrar: sí
Cambiar el nombre de estaciones, dispositivos, interfaces y puertos	no	sí
Agregar los dispositivos PROFINET y los componentes pasivos Ethernet del catálogo de hardware al proyecto	sí	no

- Determinar y minimizar diferencias entre la topología prevista y la topología real

Función	Vista gráfica	Vista de tabla
Comparación offline/online de los módulos, puertos e interconexiones de puertos Ethernet	no	sí
Aplicar la información de topología disponible online al proyecto offline	no	sí

Consulte también

Resumen de los ajustes de la configuración hardware (Página 497)

Iniciar la vista topológica

Requisitos

La vista de dispositivos o la vista de redes está abierta en el editor de hardware y redes.

Procedimiento

Para iniciar la vista topológica del proyecto, proceda del siguiente modo:

1. Haga clic en la ficha "Vista topológica".

O bien:

1. Abra la vista de redes del editor de hardware.
2. Seleccione un dispositivo PROFINET o un módulo PROFINET.
3. Selecciona el comando "Ir a la vista topológica" del menú contextual.

Resultado

Se inicia la vista gráfica de la vista topológica. Si ha llamado la vista topológica desde el menú contextual, el componente seleccionado se mantiene seleccionado después de la conmutación.

Mostrar topología

Mostrar la vista gráfica de la topología configurada

¿Qué se muestra?

La vista gráfica de la topología configurada muestra lo siguiente:

- Los dispositivos PROFINET configurados y componentes Ethernet pasivos con puertos
- Estaciones configuradas con componentes no Ethernet si en la estación se encuentra como mínimo un componente Ethernet
- Interconexiones configuradas entre los puertos

Modo de representación

La vista gráfica de la vista topológica y de la vista de redes son muy similares en lo que respecta al modo de representación:

- Los componentes se muestran simplificados en comparación con la vista de dispositivos.
- Las interconexiones entre los puertos se muestran en forma de líneas horizontales y verticales. Estas líneas son discontinuas si se trata de la interconexión entre el puerto de cambiador de herramientas y sus posibles puertos interlocutores.

Mostrar la vista de tabla de la topología configurada

¿Qué se muestra?

La vista de tabla de la topología configurada muestra lo mismo que la vista gráfica, a excepción de las subredes lógicas configuradas de PROFINET:

- Todos los dispositivos PROFINET configurados y componentes pasivos de Ethernet con puertos
- Todas las estaciones configuradas con componentes no Ethernet si en la estación se encuentra como mínimo un componente Ethernet
- Interconexiones configuradas entre los puertos
  A un puerto con la propiedad "Puerto interlocutor cambiante" pertenecen tantas filas rellenadas como el número de puertos interlocutores posibles más una fila en blanco.

Modo de representación

Como su nombre bien indica, la vista de tabla de la vista topológica consiste en una tabla, que es la tabla de la vista topológica general. Está estructurada como la tabla de la vista de redes general. La tabla contiene las columnas siguientes:

- Dispositivo/Puerto
  Esta es la columna principal de la tabla. Las entradas de esta columna están estructuradas de manera jerárquica y el último elemento de la jerarquía son los puertos PROFINET. Las entradas jerárquicas se pueden expandir y contraer. Por ejemplo, en el caso de una CPU, una entrada se compone de los siguientes elementos:
  - Nombre de estación
  - Nombre de dispositivo
  - Nombre de la interfaz PROFINET
  - Nombre de los puertos

  Nota: El resto de las columnas contienen entradas únicamente en las filas con los nombres de puertos.
- Tipo (como ajuste predeterminado no se muestra esta columna).
  Muestra a qué tipo de estación, dispositivo o interfaz se refiere la fila de la tabla o si pertenece a un puerto.
- Referencia (como ajuste predeterminado no se muestra esta columna)
  Referencia del dispositivo
- Subred (como ajuste predeterminado no se muestra esta columna).
  Subred configurada a la que pertenece la interfaz
- Sistema maestro/IO (como ajuste predeterminado no se muestra esta columna).
  Muestra si la interfaz pertenece a un sistema maestro PROFIBUS DP o a un sistema PROFINET IO.
- Dirección de dispositivo (como ajuste predeterminado no se muestra esta columna).
  Dirección configurada de la interfaz en la subred
- Estación del interlocutor
  Nombre de la estación que contiene el puerto interlocutor
- Dispositivo del interlocutor
  Nombre del dispositivo que contiene el puerto interlocutor
- Interfaz del interlocutor
  Interfaz a la que pertenece el puerto interlocutor
- Puerto interlocutor
- Datos de la línea
  Contiene la longitud de la línea y el tiempo de propagación de señal de la línea que conecta los puertos

Funciones básicas de las tablas

La tabla de la vista topológica general soporta las siguientes funciones básicas para editar una tabla:

- Mostrar y ocultar las columnas de la tabla
  Nota: No es posible ocultar las columnas relevantes para la configuración.
- Optimizar el ancho de las columnas
- Explicar el significado de una columna, fila o campo mediante tooltips.

Mostrar el estado de diagnóstico de puertos en la vista gráfica

Requisitos

La vista gráfica de la vista topológica está abierta.

Procedimiento

Para determinar el estado de diagnóstico de los puertos, proceda del siguiente modo:

1. Conéctese online con el componente o los componentes deseados.

Resultado

Aparecen los símbolos siguientes:

- Cada dispositivo muestra su respectivo símbolo de diagnóstico.
- Si hay como mínimo un error en un componente subordinado, se muestra también el símbolo de diagnóstico "Error en componente subordinado" en la esquina inferior izquierda del símbolo de diagnóstico.
- De cada puerto se muestra el respectivo símbolo de diagnóstico.
- Las líneas entre dos puertos que estén online reciben el color correspondiente a su estado de diagnóstico.

Encontrará los posibles símbolos de diagnóstico de los puertos y el color de las líneas Ethernet en la descripción del diagnóstico del hardware. Consulte: Mostrar el estado de diagnóstico y de comparación mediante iconos (Página 2160)

Mostrar el estado de diagnóstico de los componentes hardware en la vista de tabla

Requisitos

La vista de tabla de la vista topológica está abierta.

Procedimiento

Para determinar el estado de diagnóstico de los componentes hardware de la tabla de la vista topológica general, proceda del siguiente modo:

1. Conéctese online con el componente deseado.

Resultado

Los siguientes símbolos aparecen en el margen izquierdo de la tabla de la vista topológicao general, es decir en cada fila que pertenece al componente afectado:

- Se muestra el símbolo de diagnóstico que pertenece al componente hardware.
- En los componentes hardware con componentes subordinados se muestra además el símbolo de "error en componente subordinado" en la esquina izquierda bajo el símbolo de diagnóstico del componente hardware, si hay como mínimo un error en uno de los componentes subordinados.

Los posibles símbolos de diagnóstico de los componentes hardware se encuentran en la descripción del diagnóstico de hardware. Consulte: Mostrar el estado de diagnóstico y de comparación mediante iconos (Página 2160)

Nota

La visualización del estado del diagnóstico de los componentes hardware en la tabla de la vista topológica general y la de la tabla de la vista de redes general son idénticas.

Ejecutar una comparación offline/online y mostrar del resultado

Requisitos

La vista topológica está abierta. Una conexión online con uno o más dispositivos es posible pero no obligatoria.

Procedimiento

Para determinar las diferencias entre la topología configurada y la topología realmente existente, proceda del siguiente modo:

1. Haga clic en el botón "Comparar offline/online" de la barra de herramientas de la vista topológica general.

Resultado

Las columnas "Estación del interlocutor", "Interfaz del interlocutor" y "Datos de la línea" de la tabla de la vista topológica general se borran.

La tabla se amplía a la derecha con dos grupos de columnas que al principio aparecían vacías:

- A la derecha se agregan columnas para la topología por determinar online.
- Entre las columnas para la topología offline y online se agregan las columnas "Estado", "Acción" y "Descripción" para el resultado de la comparación offline/online.

Nota

Como ajuste predeterminado no se muestra la columna "Descripción".

Se activan los siguientes botones en la barra de herramientas de la tabla:

Botón	Nombre	Significado
	Actualizar	Se reinicia la determinación de la topología existente online.
	Sincronizar	• Aplicar al proyecto las interconexiones de puertos determinadas online (Página 749) • Aplicar los dispositivos determinados online al proyecto (Página 750)

Después de haber determinado la topología realmente existente, se rellenan las columnas agregadas. A continuación se describe con más detalle.

Nota

Para el puerto que está conectado con la PG/el PC que solo está disponible online, se muestra una diferencia entre la vista offline y online. Esto se debe a que la PG/el PC no puede configurarse offline.

Columnas de la topología determinada online

Se muestran las siguientes columnas:

- "Dispositivo/Puerto"
- "Tipo" (como ajuste predeterminado no se muestra esta columna).
- "Referencia" (como ajuste predeterminado no se muestra esta columna)
- "Dirección IP" (como ajuste predeterminado no se muestra esta columna).
- "Dispositivo del interlocutor"
- "Puerto interlocutor"
- "Datos de la línea"

Columnas del resultado de la comparación offline/online

Se muestran las siguientes columnas:

- "Estado"
  Aquí se muestra el resultado de la comparación offline/online mediante símbolos de diagnóstico. Pueden encontrarse los siguientes símbolos:

Símbolo de diagnóstico	Significado
	Distinta información topológica en un componente subordinado, como mínimo
	Información topológica idéntica
	La información topológica solo está disponible offline o el dispositivo está desactivado
	La información topológica solo está disponible online
	Distinta información topológica
	Si un dispositivo no soporta ninguna función de topología, la columna "Estado" queda vacía.

- "Acción"
  Aquí se muestran las acciones posibles mediante símbolos. Pueden encontrarse los siguientes símbolos:

Icono	Significado
=	Ninguna acción posible
←	Aplicar la interconexión determinada online

- "Descripción"
  Esta columna describe la acción elegida.

Configurar topología

Interconectar puertos

Sinopsis

Interconectar puertos en la vista topológica

En la vista topológica dispone de las siguientes posibilidades para interconectar los puertos:

- en la vista gráfica (Página 742)
- en la vista gráfica de un cambiador de herramientas (Página 744)
- en la vista de tabla (Página 744)

- en la vista de tabla de un cambiador de herramientas (Página 745)
- mediante la aplicación de las interconexiones de puertos determinadas online (Página 749)

¿Qué efectos tiene la interconexión de puertos en la vista de redes?

Nota

En la vista de redes, en las propiedades de una subred se puede especificar que al crear una interconexión de puertos entre dos dispositivos no interconectados se utilice esta subred.

Al crear una interconexión entre dos puertos son posibles los siguientes efectos en la vista de redes:

- Si las interfaces correspondientes no están conectadas: Si ha especificado una subred, se utilizará dicha subred. Si no, se creará una nueva subred para conectar las dos interfaces.
- Si sólo está conectada una de las dos interfaces correspondientes: La interfaz no conectada se conectará con la misma subred que la interfaz ya conectada.
- En otros casos: Las interfaces correspondientes no se conectan con una subred lógica.

Consulte también

Interconectar puertos (Página 1506)

Ajustes para la interconexión de dispositivos Ethernet (Página 553)

Interconectar los puertos en la vista gráfica

Requisitos

Se encuentra en la vista gráfica de la vista topológica.

Procedimiento: crear interconexiones entre dos puertos

Para interconectar un puerto de un dispositivo con un puerto de otro dispositivo, proceda del siguiente modo:

1. Coloque el puntero del ratón en el puerto que desee interconectar.
2. Haga clic con el botón izquierdo del ratón y manténgalo pulsado.
3. Desplace el puntero del ratón.
   El puntero del ratón indica con el símbolo de interconexión que se encuentra en el modo de interconexión. Al mismo tiempo, en el puntero del ratón aparece el símbolo de bloqueo, que no desaparece hasta que no se sitúa en una posición de destino válida.

4. Arrastre el puntero del ratón al puerto de destino. Puede mantener el botón izquierdo del ratón pulsado o soltarlo.
5. A continuación, suelte el botón izquierdo del ratón o, si ya lo ha soltado, haga clic de nuevo con dicho botón.

Resultado: Se creará una nueva interconexión de puertos.

Nota

Creación de un anillo en CPUs S7-300, S7-400 y S7-1500

Al crear un anillo en CPUs S7-300, S7-400 o S7-1500 mediante interconexiones de puertos, se genera automáticamente un dominio MRP.

Procedimiento: modificación de interconexión existente de puertos sin borrarla previamente

Proceda del siguiente modo:

1. Coloque el cursor del ratón sobre el puerto de una interconexión existente que deba recibir un nuevo puerto de interlocutor.
2. Arrástrelo hasta el nuevo puerto de interlocutor.

Resultado: la interconexión existente de puertos se borra. Se crea la nueva interconexión de puertos.

Procedimiento alternativo:

1. Coloque el cursor del ratón en un puerto no interconectado hasta el momento, y que deba conectarse con un puerto ya interconectado.
2. Arrástrelo hasta el puerto ya interconectado.

Resultado: la interconexión existente de puertos se borra. Se crea la nueva interconexión de puertos.

Procedimiento: interconexión de dos puertos ya interconectados, sin tener que borrar previamente las dos interconexiones existentes

Proceda del siguiente modo:

1. Coloque el cursor del ratón sobre un puerto ya interconectado que deba recibir un nuevo puerto de interlocutor.
2. Arrástrelo hasta el nuevo puerto de interlocutor que a su vez ya está interconectado.

Resultado: las dos interconexiones existentes de puertos se borran. Se crea la nueva interconexión de puertos.

Interconexión de los puertos en la vista de tabla

¿Qué acciones se pueden realizar con interconexiones de puertos en la vista de tabla?

En la vista de tabla se pueden realizar las siguientes acciones con interconexiones de puertos:

- Crear nueva interconexión de puertos
- Modificar interconexión de puertos existente
- Borrar interconexión de puertos existente

Requisitos

En la vista topológica general se ve la línea con el puerto cuya interconexión desea crear, modificar o borrar.

Procedimiento

Para crear, modificar o borrar la interconexión de un puerto por primera vez, proceda del siguiente modo:

1. Mueva el puntero del ratón a la línea del puerto de origen en la columna "Puerto interlocutor".
2. Haga clic en la lista desplegable.
3. Seleccione el puerto interlocutor deseado (al crear o modificar una interconexión de puertos) o la entrada "no interconectado" (al borrar una interconexión de puertos).

Resultado

Se realiza la acción deseada. En la columna "Puerto interlocutor" se muestra el nuevo puerto interlocutor (después de crear o modificar una interconexión de puertos) o la indicación "Seleccione puerto" (después de borrar una interconexión de puertos).

Interconexión de un puerto con varios puertos interlocutores en la vista gráfica

Requisitos

- Ha parametrizado un puerto de un dispositivo PROFINET con la propiedad "Interlocutores alternativos" y ha indicado los puertos interlocutores posibles.
- La vista gráfica de la vista topológica está abierta.

Procedimiento

1. Interconecte este puerto (en lo sucesivo denominado puerto de origen) con uno de los puertos interlocutores indicados (en lo sucesivo denominados puertos de destino).
2. Interconecte el puerto de origen con otro puerto de destino.
   Esto se puede realizar de diferentes maneras:
   – Arrastre el puntero del ratón de un puerto interlocutor ya interconectado a un puerto de destino.
   – Arrastre el puntero del ratón de una interconexión ya establecida a un puerto de destino.
   – Arrastre el puntero del ratón de un puerto de destino a un puerto interlocutor ya interconectado.
   – Arrastre el puntero del ratón de un puerto de destino a una interconexión ya establecida.
3. Repita el paso anterior una o varias veces, según sea necesario.

Resultado

Se establece una interconexión entre el puerto de origen y los puertos interlocutores cambiantes. Esta interconexión se muestra mediante una línea discontinua.

Interconexión de un puerto con varios puertos interlocutores en la vista de tabla

¿Qué acciones se pueden realizar con interconexiones de puertos a varios puertos interlocutores en la vista de tabla?

En un cambiador de herramientas, las acciones que se pueden realizar con interconexiones de puertos a varios puertos interlocutores en la vista de tabla son:

- Crear nueva interconexión de puertos
- Modificar interconexión de puertos existente
- Borrar interconexión de puertos existente

Requisitos

- Ha parametrizado un puerto de un dispositivo PROFINET con la propiedad "Interlocutores alternativos" y ha indicado los puertos interlocutores posibles.
- En la vista topológica general se ve la línea con el puerto cuya interconexión desea crear, modificar o borrar.

Procedimiento

Para crear por primera vez, modificar o borrar la interconexión de un puerto con otro de varios puertos interlocutores, proceda del siguiente modo:

1. Mueva el puntero del ratón a la línea del puerto de origen en la columna "Puerto interlocutor".
2. Haga clic en la lista desplegable.
3. Seleccione el puerto interlocutor deseado (al crear o modificar una interconexión de puertos) o la entrada "no interconectado" (al borrar una interconexión de puertos).

Resultado

Se realiza la acción deseada:

- En caso de crear un puerto se inserta una fila nueva en la tabla de la vista topológica general. En la columna "Puerto interlocutor" se muestra el nuevo puerto interlocutor.
- Al realizar una modificación, en la columna "Puerto interlocutor" se muestra el nuevo puerto interlocutor.
- En caso de borrar, se borra la fila con la interconexión de puertos existente hasta el momento.

Nota

En un cambiador de herramientas, un puerto con interconexiones a varios puertos interlocutores suele constar de varias filas. En este caso, la última fila siempre es una línea en blanco. La primera fila se puede editar, el resto son de solo lectura.

Cambiar el nombre de estaciones, dispositivos, interfaces y puertos

Cambiar el nombre de una estación, dispositivo, interfaz o puerto

Requisitos

La vista de tabla de la topología configurada está abierta.

Procedimiento

Para cambiar el nombre de una estación, dispositivo, interfaz o puerto, proceda del siguiente modo:

1. Haga clic dos veces en el campo correspondiente de la tabla de la vista topológica general (con el segundo clic se accede al modo de edición).
2. Introduzca el nombre nuevo y pulse la tecla INTRO (finaliza el modo de edición).

Resultado

Se cambia el nombre del objeto.

Comparar offline/online

Asignación automática de dispositivos mediante comparación offline/online

Sinopsis

Al comparar offline/online se compara la topología configurada con la topología realmente existente. Aquí se asignan automáticamente los dispositivos determinados online a los dispositivos configurados, siempre que sea posible.

Iniciar la determinación de disponibilidad

Se inicia la determinación de disponibilidad por primera vez mediante un clic en el botón "Comparar offline/online" de la barra de herramientas de la vista topológica.

La determinación de disponibilidad se reinicia haciendo clic en el botón "Actualizar".

Nota

La determinación de la disponibilidad puede requerir varios segundos. Durante este tiempo no se pueden realizar más entradas.

Asignación automática

Un dispositivo determinado online se asigna automáticamente a un dispositivo configurado si los dos dispositivos concuerdan en las siguientes propiedades:

- Nombre del dispositivo
- Referencia
- Número de puertos

A continuación, se describen los casos en los que puede ocurrir y qué medidas pueden tomarse dado el caso:

- Interconexiones de puertos idénticas
  Es el caso ideal. No es necesario tomar medidas.

Columna "Acción"	Significado
	ninguna acción

- Existen interconexiones en el dispositivo determinado y en el dispositivo configurado pero hay diferencias.
  Para ello existen las siguientes alternativas:
  – Si es posible aplicar la configuración online

Columna "Acción"	Significado
	Aplicar interconexión online (Página 749)
	ninguna acción

  – Si no es posible aplicar la configuración online

Columna "Acción"	Significado
	ninguna acción

- Una interconexión sólo existe online.
  Para ello existen las siguientes alternativas:
  – Si es posible aplicar la configuración online

Columna "Acción"	Significado
	Aplicar interconexión online (Página 749)
	ninguna acción

  – Si no es posible aplicar la configuración online

Columna "Acción"	Significado
	ninguna acción

- Una interconexión existe sólo en la configuración.
  Para ello existen las siguientes alternativas:

Columna "Acción"	Significado
	Aplicar interconexión online (Página 749), es decir, la interconexión se borra de la configuración
	ninguna acción

Asignación automática no posible

La asignación automática no es posible en los siguientes casos:

- No existe ningún dispositivo determinado online para el dispositivo configurado (por lo que las columnas correspondientes en el área "Topología online" de la tabla de la vista topológica están vacías).
  En este caso, debe agregar el dispositivo configurado a la instalación o borrar el dispositivo configurado de la configuración.
- No se puede asignar un dispositivo determinado online a ningún dispositivo configurado (por lo que las columnas correspondientes en el área "Topología online" de la tabla de la vista topológica están vacías).
  En este caso puede aplicar el dispositivo determinado online al proyecto (Página 750).

Aplicar al proyecto las interconexiones de puertos determinadas online

Requisitos

Ha realizado una comparación offline/online en la vista topológica. El resultado ha sido que como mínimo un dispositivo determinado online se ha asignado automáticamente a un dispositivo configurado pero hay diferencias en la interconexión.

Procedimiento

Para aplicar manualmente al proyecto una o varias interconexiones de puertos determinadas online, proceda del siguiente modo:

1. Seleccione el valor "Aplicar" en la columna "Acción" para un puerto de un dispositivo configurado al que ha sido asignado un dispositivo determinado online.
2. Repita este paso para otros puertos del mismo dispositivo configurado, según sea necesario.
3. Si es necesario, repita los pasos anteriores para otros dispositivos configurados a los que se han asignado dispositivos determinados online y que muestran diferencias de interconexión.
4. Haga clic en el botón de comando "Sincronizar".

Resultado

Para los dispositivos correspondientes se aplican al proyecto las interconexiones de puertos determinadas online y la información de línea. Si la aplicación se realiza correctamente, aparece el símbolo de diagnóstico "Información de topología idéntica" en cada puerto.

Nota

Si para un dispositivo determinado online se detectan interconexiones de puertos diferentes de las que existen en el proyecto, su aplicación al proyecto tendrá como resultado la sustitución de las interconexiones existentes en el proyecto por las interconexiones detectadas online. Si para un dispositivo determinado online no se detecta ninguna interconexión de puertos, la aplicación al proyecto tendrá como consecuencia la eliminación de todas las interconexiones de este dispositivo del proyecto.

Aplicar los dispositivos determinados online al proyecto

Requisitos

Ha realizado una comparación offline/online en la vista topológica. El resultado ha sido que como mínimo un dispositivo determinado online no se ha podido asignar a ningún dispositivo configurado.

Procedimiento

Para aplicar manualmente al proyecto uno o varios dispositivos determinados online, proceda del siguiente modo:

1. Mueva el puntero del ratón en la columna "Dispositivo/Puerto" de la topología online de un dispositivo configurado sin interlocutor online.
2. En la lista desplegable de este campo, seleccione el dispositivo que desea asignar al dispositivo configurado.
3. Si es necesario, repita los pasos anteriores para otros dispositivos configurados sin interlocutor online.

Resultado

Se desplaza el dispositivo determinado online seleccionado desde el final de la tabla hasta el principio. Después, se encontrará en la fila del dispositivo configurado al que acaba de asignarlo.

8.1.3.4 Industrial Ethernet Security

Configuración de la seguridad

General

Dispositivos soportados

Dispositivos soportados

Existe la posibilidad de configurar funciones de seguridad para los productos siguientes:

- SCALANCE S:
  - S602 V2/V3/V4
  - S612 V2/V3/V4
  - S613 V2
  - S623 V3/V4
  - S627-2M V4
- SOFTNET Security Client:
  - SOFTNET Security Client V4
- CPs S7: CP 343-1 GX31 Advanced, CP 443-1 GX30 Advanced, CP 1543-1
- CP PC: CP 1628
- Router de telefonía móvil: SCALANCE M875

Denominación genérica "módulo de seguridad"

En el presente apartado del sistema de información, los productos siguientes se denominan genéricamente "módulo de seguridad": SCALANCE S602 / SCALANCE S612 / SCALANCE S613 / SCALANCE S623 / SCALANCE S627-2M, CP 343-1 GX31 Advanced, CP 443-1 GX30 Advanced, CP 1543-1, CP 1628, SCALANCE M875.

Uso de las denominaciones "interfaz" y "puerto"

En la presente documentación se utilizan las siguientes denominaciones para los puertos de los módulos SCALANCE S:

- "Interfaz externa": el puerto externo del SCALANCE S602 / S612 / S623 o bien un puerto externo del SCALANCE S627-2M (marca roja)
- "Interfaz interna": el puerto interno del SCALANCE S602 / S612 / S623 o bien un puerto interno del SCALANCE S627-2M (marca verde)
- "Interfaz DMZ": el puerto DMZ del SCALANCE S623 / S627-2M (marca amarilla)

La denominación "puerto" se utilizará cuando se quiera destacar un puerto en particular de una interfaz.

Denominación genérica "CP x43-1 Adv."

En la presente documentación los siguientes productos se denominan genéricamente "CP x43-1 Adv.": CP 343-1 GX31 Advanced / CP 443-1 GX30 Advanced.

Estructura del presente apartado de ayuda

En el capítulo "General" encontrará los temas que son relevantes para todos los módulos de seguridad. Las informaciones que solo son relevantes para determinados tipos de módulo se encuentran en los apartados correspondientes específicos de cada módulo.

Vista general - Prestaciones y funcionamiento

Denominación general "STEP 7"

La configuración de funciones de seguridad se soporta a partir de STEP 7 V12. Por este motivo, en el presente apartado del sistema de información se utiliza la denominación "STEP 7" como término global para todas las versiones de STEP 7 a partir de V12.

Prestaciones

En STEP 7 pueden utilizarse las siguientes funciones de seguridad:

- Configuración de los módulos de seguridad
- Creación de datos de configuración VPN para SOFTNET Security Client V4
- Creación de datos de configuración VPN para SCALANCE M875
- Funciones de test y diagnóstico, indicaciones de estado

Vista de configuración offline y vista de diagnóstico online

Las funciones de seguridad se configuran en dos vistas:

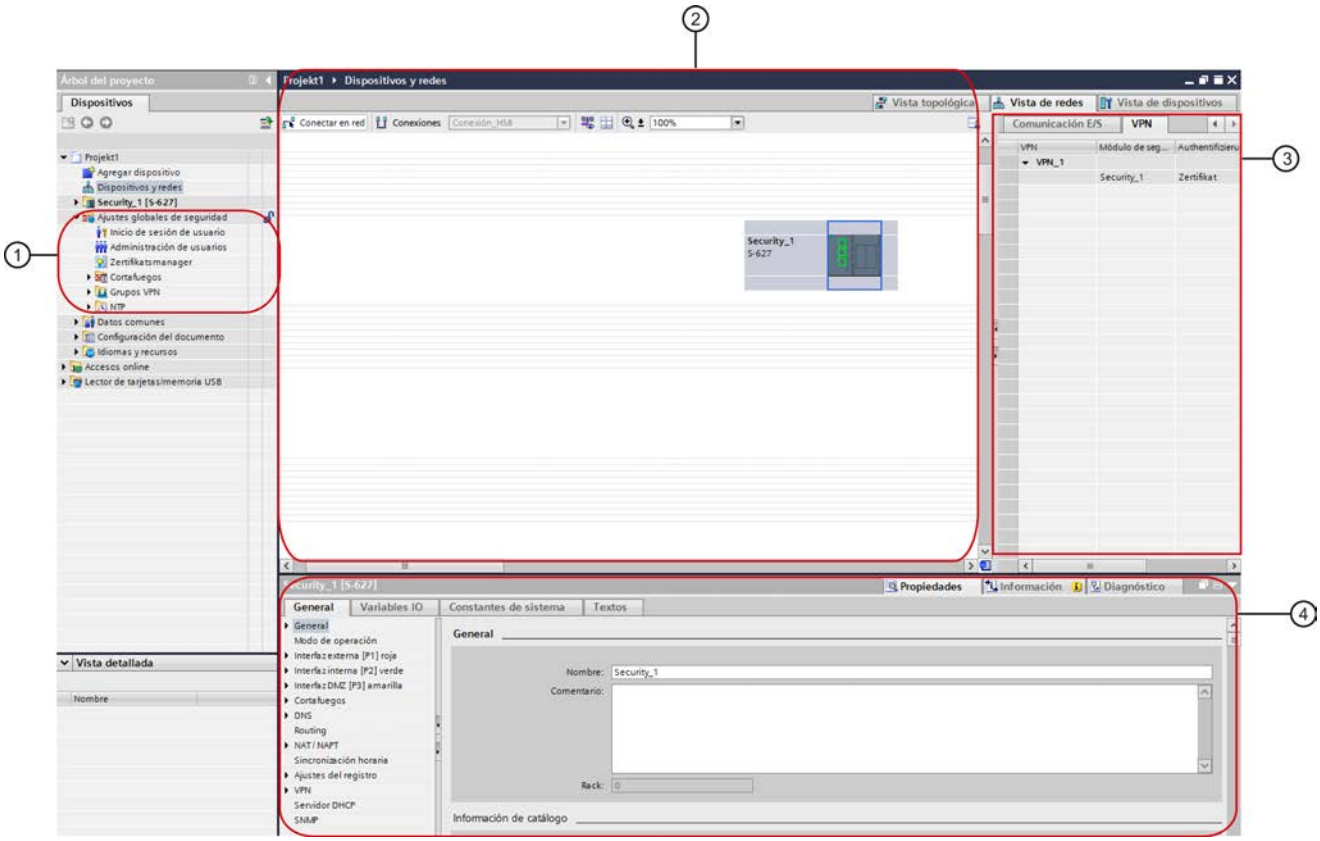
- Vista de configuración offline
  En la vista de configuración offline se ajustan los datos de configuración para los módulos de seguridad y el SOFTNET Security Client. Antes de realizar la carga no es necesario establecer para esto una conexión con los módulos de seguridad.
- Vista de diagnóstico online
  La vista de diagnóstico online sirve para diagnosticar los módulos de seguridad y ofrece la posibilidad de actualizar el firmware, entre otros.

Funcionamiento - Seguridad y coherencia

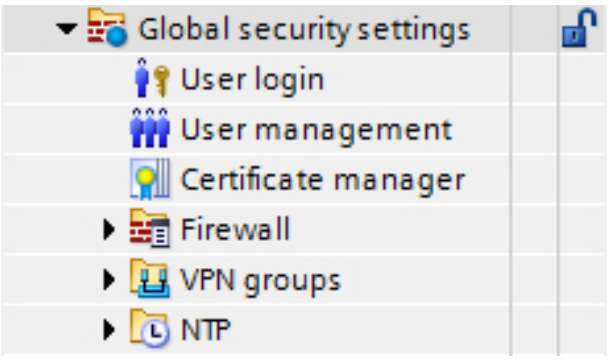
- Acceso solo para usuarios autorizados
  Las funciones de seguridad de cada proyecto están protegidas contra accesos no autorizados mediante un nombre de usuario y una contraseña.
- Datos de proyecto coherentes
  Ya durante la entrada en los distintos cuadros de diálogo se realizan comprobaciones de la coherencia. Adicionalmente se ejecutan pruebas de coherencia a nivel de proyecto en las que se incluyen todos los cuadros de diálogo.
  En los módulos de seguridad solo se pueden cargar datos de proyecto coherentes.
- Protección de datos de proyecto por cifrado
  Los datos de configuración y del proyecto relevantes para la seguridad está protegidos por cifrado. Según el módulo de seguridad, los datos se guardan en el proyecto y/o en el C-PLUG.

Interfaz de usuario - Estructura y comandos de menú

Interfaz de usuario para funciones de seguridad en STEP 7



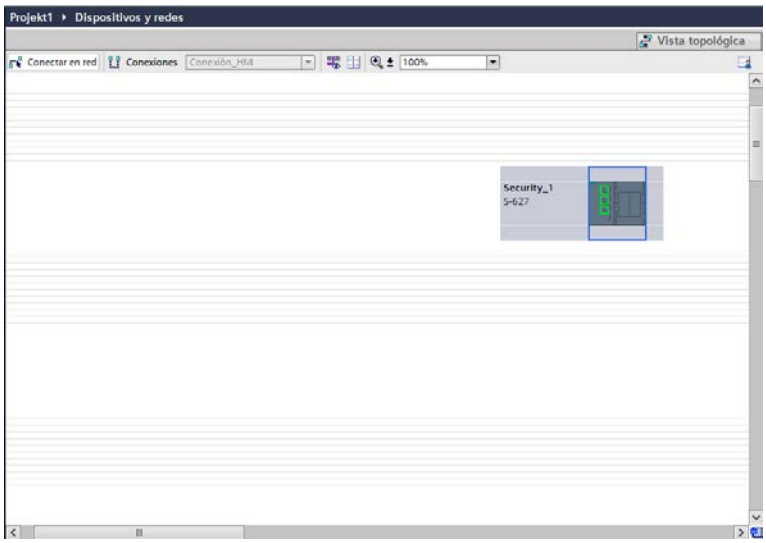
① Ajustes de seguridad global



En el árbol del proyecto se encuentran los ajustes de seguridad globales. Estos ajustes de seguridad pueden configurarse en función del módulo y a continuación asignarlos, en su caso, a módulos de seguridad individuales. Si el primer módulo de seguridad que debe configurarse es un CP, los ajustes de seguridad globales no se visualizan hasta que se activan las funciones de seguridad en los ajustes de seguridad locales del CP. Si el primer módulo de seguridad que debe configurarse es un módulo SCALANCE S, los ajustes de seguridad globales se muestran ya tras iniciar sesión en el proyecto de seguridad. En los ajustes de seguridad globales están disponibles las siguientes carpetas principales y entradas:

- Inicio de sesión de usuario
  Para la configuración de seguridad dentro de un proyecto existe una administración de usuarios propia. Inicie sesión en la configuración de seguridad mediante la entrada "Inicio de sesión de usuario". La primera vez que se inicia sesión en la configuración de seguridad se crea automáticamente un usuario con el rol "Administrador" definido por el sistema. En la administración de usuarios de la configuración de seguridad pueden crearse usuarios adicionales.
- Administración de usuarios
  En la administración de usuarios se crean usuarios, se definen derechos para roles y se asignan dichos roles a los usuarios.
- Administrador de certificados
  En el administrador de certificados se obtiene una relación de todos los certificados utilizados en el proyecto. Así, p. ej., es posible importar nuevos certificados así como exportar, modificar o sustituir certificados existentes.
- Cortafuegos
  En la entrada "Cortafuegos" se definen conjuntos de reglas globales para el cortafuegos IP y MAC así como conjuntos de reglas IP específicos del usuario (solo para módulos SCALANCE S) y se asignan módulos de seguridad. Las definiciones de los servicios IP y MAC permiten especificar de forma compacta y clara las reglas de cortafuegos IP y MAC.
- Grupos VPN
  Esta carpeta contiene todos los grupos VPN generados. Aquí podrá crear nuevos grupos VPN y asignar módulos de seguridad a dichos grupos VPN. Además, es posible adaptar las propiedades de grupos VPN ya creados.
- NTP
  Aquí puede crear servidores NTP seguros y asignarlos a uno o varios módulos de seguridad. Así se consigue que la sincronización horaria tenga lugar a través del servidor NTP asignado. Los servidores NTP no seguros pueden configurarse en los ajustes de seguridad locales.

② Área de trabajo con módulo de seguridad

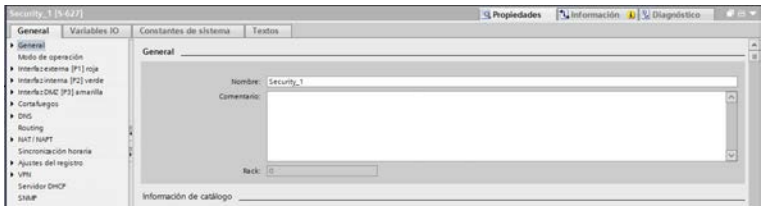


Después de haber seleccionado un módulo de seguridad en el área de trabajo, sus ajustes de seguridad locales pueden configurarse en "Propiedades" > "General". Si el módulo de seguridad seleccionado se encuentra en un grupo VPN, la información correspondiente se muestra en la ficha VPN.

③ Ficha VPN

En esta ficha se muestra información sobre todos los grupos VPN a los que pertenece el módulo de seguridad que se ha seleccionado en el área de trabajo. La información de cada uno de los participantes de un grupo VPN se puede mostrar u ocultar.

④ Ajustes de seguridad local



Los ajustes de seguridad local se configuran para un módulo de seguridad determinado. Después de haber seleccionado un módulo de seguridad en el área de trabajo, sus ajustes de seguridad locales están disponibles en la ventana de inspección dentro de "Propiedades" > "General".

Nota para CPs:

Antes de poder configurar ajustes de seguridad para CPs deben activarse los ajustes de seguridad locales. Para ello, inicie sesión en el proyecto de seguridad y active la casilla de verificación "Activar funciones de seguridad" en "Propiedades" > "General" > "Seguridad" de la ventana de inspección. Los ajustes de seguridad locales se visualizan a continuación dentro de la entrada "Seguridad". Al activar la casilla de verificación, los siguientes ajustes (siempre que estén activados) se migran automáticamente a los ajustes de seguridad locales.

CP x43-1 Advanced:

- SNMP
- Configuración FTP
- Sincronización horaria
- Servidor web
- Entradas de listas de acceso IP

CP 1543-1:

- SNMP
- Configuración FTP
- Sincronización horaria

CP 1628:

- SNMP
- Sincronización horaria

Adicionalmente se crean automáticamente para las conexiones configuradas reglas de cortafuegos que habilitan el establecimiento de conexión. Para registrar paquetes bloqueados están disponibles los ajustes de registro correspondientes.

Áreas de configuración seguras y no seguras

La interfaz de usuario puede dividirse en áreas de configuración seguras y no seguras.

Se consideran seguras las áreas en las que no es posible realizar la configuración hasta que se ha iniciado sesión en la configuración de seguridad. Estas áreas son cifradas, con lo que solo son accesibles a personas autorizadas dentro de la administración de usuarios, incluso si el proyecto es accesible a un grupo de personas más amplio.

En cambio, las funciones de las áreas no seguras pueden configurarse sin necesidad de iniciar sesión en la configuración de seguridad. Antes de cargar el proyecto en los componentes de la instalación debe comprobarse que los ajustes sean correctos, en caso de que un grupo amplio de personas pueda realizar cambios en el proyecto.

La lista siguiente muestra qué áreas de configuración de la interfaz de usuario son seguras y cuáles no. En parte, esto depende del módulo de seguridad para el que se realiza la configuración.

- Todos los ajustes de seguridad globales son seguros.
- Áreas de configuración seguras y no seguras para módulos SCALANCE S:
  - Todos los ajustes para interfaces y puertos, especialmente direcciones IP, son no seguros.
  - Los ajustes que se encuentran en la entrada "General" de los ajustes de seguridad locales son no seguros.
  - Los ajustes de nivel superior (p. ej. ajustes de MRP como administrador MRP, etc.) que no se configuran en el propio módulo de seguridad pero que pueden afectar al módulo de seguridad, son no seguros. Esto no afecta a los ajustes de seguridad globales.
  - Los demás ajustes son seguros.
- Áreas de configuración seguras y no seguras para CP 343-1 GX31 Advanced, CP 443-1 GX30 Advanced, CP 1628 y CP 1543-1:
  - Todos los ajustes que están fuera de la entrada "Seguridad" son no seguros.
  - Los ajustes de nivel superior (p. ej. ajustes de MRP como administrador MRP, ajustes de PROFINET, conexiones, etc.) que no se configuran en el propio módulo de seguridad pero que pueden afectar al módulo de seguridad, son no seguros. Esto no afecta a los ajustes de seguridad globales.
  - Todos los ajustes para interfaces y puertos, especialmente direcciones IP, son no seguros.
  - Los ajustes que se encuentran en la entrada "Seguridad" son seguros.

Realizar prueba de coherencia

Sinopsis

Existen las siguientes pruebas de coherencia:

- Pruebas de coherencia locales
- Pruebas de coherencia a nivel de proyecto

En las descripciones de los diferentes cuadros de diálogo de esta ayuda se listan bajo el término "Verificación de consistencia" las reglas que deben tenerse en cuenta al realizar cada entrada.

Pruebas de coherencia locales

Una prueba de coherencia se considera local si se puede realizar directamente dentro de un cuadro de diálogo. Para las acciones siguientes se llevan a cabo verificaciones de coherencia locales:

- al salir de un campo
- al salir de una fila de una tabla
- al confirmar un cuadro de diálogo con OK.

Pruebas de coherencia a nivel de proyecto

Las pruebas de coherencia a nivel de proyecto informan sobre la configuración correcta de los datos de proyecto. Para las acciones siguientes se comprueba la coherencia en todo el proyecto:

- al compilar una configuración
- al cargar una configuración

Nota

Solo se pueden cargar datos configurados en un módulo de seguridad si el proyecto es coherente en su conjunto.

Administrar certificados

Resumen de certificados

¿Cómo se administran los certificados?

En el administrador de certificados se obtiene una relación de todos los certificados utilizados en el proyecto, p. ej. certificados CA, con la información correspondiente a solicitante, emisor, validez, uso y presencia de una clave privada.

El certificado CA es un certificado emitido por un entidad emisora, llamada "Certificate Authority", y de él se derivan los certificados de dispositivo. Entre ellos se encuentran los certificados SSL y, en caso de que el módulo de seguridad forme parte de un grupo VPN, los certificados de grupo VPN. Los certificados SSL son necesarios para la autentificación en la comunicación segura entre un dispositivo de red y un módulo de seguridad. Las posibles entidades emisoras pueden ser:

- El propio STEP 7. Si el solicitante y el emisor son iguales, se trata de un certificado autofirmado, es decir, emitido por STEP 7.
- Una entidad emisora superior. Los certificados externos al proyecto de otros emisores se importan y se guardan en la memoria de certificados de STEP 7.

Los certificados creados por una de las dos entidades emisoras, siempre cuentan con una clave privada, que permite derivar los certificados de dispositivo.

El administrador de certificados incluye, además, las siguientes funciones:

- Modificación de certificados ya existentes (p. ej. periodo de validez).
- Importación de certificados y entidades emisoras nuevas.
- Importación de certificados SSL (solo con CP x43-1 Adv.), p. ej. para comunicación FTP.
- Exportar los certificados y entidades emisoras utilizados en el proyecto.
- Renovación de certificados y entidades emisoras caducados.
- Sustitución de entidades emisoras ya existentes por otras.
- Agregación de certificados y entidades emisoras de confianza.
- Borrado de certificados importados manualmente.

Nota

Cargar la configuración

Tras sustituir o renovar certificados es necesario cargar la configuración en los módulos de seguridad correspondientes.

Tras sustituir o renovar certificados CA es necesario cargar la configuración en todos los módulos de seguridad.

Nota

Fecha y hora actuales en los módulos de seguridad

Cuando utilice comunicación segura (p. ej. HTTPS, VPN...), asegúrese de que los módulos de seguridad afectados tienen la hora y fecha actuales. De lo contrario, los certificados utilizados se considerarán no válidos y la comunicación segura no funcionará.

Cómo se accede a esta función

Haga doble clic en la entrada "Administrador de certificados" de los ajustes de seguridad globales.

En las diferentes fichas se dispone de los comandos siguientes del menú contextual:

Comando	Significado
Importar/exportar	Importación / Exportación de certificados de dispositivos o certificados CA. Los certificados se transmiten al módulo de seguridad. Son posibles los siguientes formatos: *.cer (solo certificado) *.crt (solo certificado) *.pem (solo certificado) *.p12 (certificado con las claves privadas correspondientes)
Indicadores	Abre el cuadro de diálogo de certificados de Windows, en el que se muestra una vista general de todos los datos de certificado.

Comando	Significado
Renovar (solo en las fichas "CA" y "Certificados de dispositivos")	Abre el cuadro de diálogo "Crear nuevo certificado", en el que, si es necesario, se puede importar p. ej. un certificado comprometido o crear un certificado nuevo de STEP 7.
Reemplazar (solo en la ficha "CA")	Abre el cuadro de diálogo "Cambiar autoridad de certificación (CA)", que permite reemplazar una autoridad de certificación existente por otra nueva.
Borrar (solo para certificados importados manualmente)	Borra un certificado en la ficha "Certificados de confianza y entidades emisoras de certificados de origen".

Entidades emisoras

Ficha "CA"

Los certificados que se muestran aquí son generados por una entidad emisora.

- Certificados CA de un proyecto: Al crear un proyecto se genera un certificado CA para el proyecto. A partir de él se derivan los certificados SSL para los diferentes módulos de seguridad.
- Certificados de grupo CA: Al crear un grupo VPN nuevo, se genera un certificado CA para el grupo VPN. A partir de este certificado se derivan los certificados de grupos VPN de módulos de seguridad que se encuentren en el correspondiente grupo VPN.

Certificados de dispositivos

Ficha "Certificados de dispositivos"

Visualización de los certificados específicos de dispositivo que son generados por STEP 7 para un módulo de seguridad. Aquí se incluyen:

- Certificado SSL de un módulo de seguridad: para cada módulo de seguridad creado se genera un certificado SSL derivado del certificado CA del proyecto. Se recurre a los certificados SSL para autenticar la comunicación segura entre PG o PC y el módulo de seguridad así como al cargar la configuración.
- Certificado de grupo VPN de un módulo de seguridad: además se genera un certificado de grupo VPN para cada módulo de seguridad por cada grupo VPN en el que se encuentra.

Certificados de confianza y entidades emisoras de certificados de origen

Ficha "Certificados de confianza y entidades emisoras de certificados de origen"

Visualización de los certificados externos importados a STEP 7. Se pueden importar p. ej. certificados de servidor provenientes de servidores FTPS externos o certificados de proyecto provenientes de otros proyectos creados con STEP 7.

Con CPs, el certificado externo importado se transmite a todos los CPs administrados en el proyecto, los cuales comprueban el certificado. Si los módulos de seguridad clasifican el

certificado como de confianza, es posible establecer una conexión con un servidor FTPS, por ejemplo. El certificado importado no se utiliza adicionalmente en otro lugar de STEP 7.

Con módulos SCALANCE S, en esta ficha solo se muestran las entidades emisoras necesarias para la verificación de los módulos de seguridad por servicios externos, como DNS din.

Renovar certificados

Significado

En este cuadro de diálogo se renuevan certificados CA y certificados de dispositivos. De ser necesario, p. ej. en caso de un certificado comprometido, es posible importar un certificado o crear un certificado nuevo mediante STEP 7.

Cómo se accede a esta función

1. Haga clic con la tecla derecha del ratón en una entrada de la lista en el administrador de certificados.
2. Elija el comando "Renovar" del menú contextual.
3. Seleccione si el nuevo certificado debe ser autofirmado o bien firmado por una entidad emisora.
4. Si el certificado debe ser firmado por una entidad emisora, seleccione con el botón "Seleccionar" la entidad emisora que debe utilizarse. Para tal fin solo están disponibles las entidades emisoras que están guardadas en la memoria de certificados del proyecto actual.

5. Según sea el certificado, introduzca los valores siguientes en el campo de entrada "Solicitante" o "Nombre alternativo del solicitante":

Certificado a renovar	Parámetros	
	Solicitante	Nombre alternativo del solicitante
Certificados CA del proyecto	Nombre del certificado CA	-
Certificado de grupo CA	Nombre del certificado de grupo CA	-
Certificado SSL para CP S7	Nombre del módulo de seguridad	Direcciones IP de las interfaces Gigabit y PROFINET, separadas por una coma
Certificado SSL para CP PC	Nombre del módulo de seguridad	Dirección IP del módulo de seguridad
Certificado SSL para SCALANCE S, SCALANCE M y SOFTNET Security Client	Nombre del módulo de seguridad	Para SCALANCE S: dirección IP externa, dirección IP interna y, dado el caso, dirección IP de la interfaz DMZ Para SCALANCE M: dirección IP externa y dirección IP interna Para SOFTNET Security Client: nombre DNS
Certificado de grupo VPN del módulo de seguridad:	Nombre del certificado del grupo VPN	Derivado del certificado de grupo CA.

6. Elija un período de validez para el certificado. Por defecto, en los campos "Válido desde:" y "Válido hasta:" se introducen la hora actual y el valor del certificado actual, respectivamente.

Reemplazar certificados

Significado

En el cuadro de diálogo "Cambiar autoridad de certificación (CA)" se reemplaza el certificado CA existente en el proyecto o el certificado de grupo CA por uno nuevo.

Cómo se accede a esta función

1. Haga clic con el botón derecho del ratón en una entrada de la lista en la ficha "CA".
2. Elija el comando "Reemplazar" del menú contextual.
3. Se abre el cuadro de diálogo "Autoridad de certificación (CA)".

Todos los certificados que aparecen en la tabla "Certificados afectados" se vuelven a derivar. De este modo es posible reemplazar el certificado de grupo CA de un grupo VPN configurado previamente dentro del proyecto por el certificado de grupo CA de otro proyecto. Es decir, los certificados de grupo VPN de los miembros del grupo VPN se derivan en ambos proyectos del mismo certificado de grupo CA.

Una vez realizados los cambios en el administrador de certificados, la configuración debe cargarse en todos los módulos de seguridad afectados.

¿Qué formato puede tener el certificado?

De la autoridad de certificación importada se derivan otros certificados en STEP 7. Por lo tanto solo se pueden seleccionar certificados con clave privada.

- *.p12

Administrar usuarios y roles

Reglas para nombres de usuario, roles y contraseñas

¿Qué reglas son aplicables para nombres de usuario, nombres de rol y contraseñas?

Al crear o modificar un usuario, un rol o una contraseña, observe las siguientes reglas:

Tabla 8-5 Reglas para la administración de usuarios

Caracteres permitidos	Se admiten los siguientes caracteres del juego US-ASCII: 0123456789 A...Z a...z !#$%&()*+,-./:;<=>?@ [\]_{|}~¦^
Caracteres no permitidos	" '
Longitud del nombre de usuario	1 ... 32 caracteres
Longitud de la contraseña	8 ... 32 caracteres
Longitud del nombre de rol	1 ... 32 caracteres
Número máximo de usuarios por proyecto	128
Número máximo de usuarios en un módulo de seguridad	32 + 1 administrador al crear el proyecto
Número máximo de roles por proyecto	124 (120 definidos por el usuario + 4 definidos por el sistema)
Número máximo de roles en un módulo de seguridad	33 (29 definidos por el usuario + 4 definidos por el sistema)

Nota

Nombres de usuario y contraseñas

Como medida importante para aumentar la seguridad, asegúrese siempre de que los nombres de usuario y las contraseñas tienen 8 caracteres y contienen caracteres especiales, mayúsculas y minúsculas, números, etc.

Crear usuarios

Significado

Las funciones de seguridad configuradas en STEP 7 están protegidas de accesos no autorizados por una administración de usuarios propia. Antes de poder acceder a los ajustes de seguridad globales y locales de los módulos de seguridad es necesario iniciar sesión con un usuario en la configuración de seguridad.

Crear el primer usuario en el proyecto

Una vez creado el primer módulo de seguridad en el proyecto hay que crear un usuario. Para ello, haga clic en el botón "Inicio de sesión de usuario" de la entrada "Propiedades de seguridad" en los ajustes de seguridad locales del módulo de seguridad creado e indique los datos de inicio de sesión del usuario que debe crearse. Seguidamente, se iniciará sesión con el usuario creado, al que se le asignará el rol "Administrador" definido por el sistema. Dicho rol incluye derechos de configuración y de módulos completos.

Crear usuarios en la administración de usuarios

Cuando se ha iniciado sesión con un usuario en la configuración de seguridad, es posible crear más usuarios o borrar usuarios existentes en la entrada "Administración de usuarios" de los ajustes de seguridad globales.

Nota

Usuario con rol "administrator"

En el proyecto siempre debe haber como mínimo un usuario con derechos de configuración completos. El "administrator", que se crea automáticamente la primera vez que se activan las funciones de seguridad en el proyecto, solo puede borrarse mientras exista como mínimo otro usuario con el rol de "administrador" definido por el sistema.

Los parámetros siguientes están disponibles en la ficha "Usuario" de la administración de usuarios:

Tabla 8-6 Datos de la ficha "Usuario"

Parámetro	Significado
Nombre de usuario	Nombre del usuario que debe crearse. Haga clic en la entrada "Agregar nuevo usuario" de la columna "Nombre de usuario" para crear un usuario nuevo.
Contraseña	Contraseña con la que el usuario se autentica al iniciar la sesión.
Rol	Selección de un rol definido por el sistema o usuario.

Parámetro	Significado
Duración máxima de la sesión (solo para SCALANCE S V3 o superior)	Introducción de la duración tras la cual se cierra automáticamente la sesión de un usuario que ha iniciado sesión en la página web para conjuntos de reglas IP específicos del usuario de módulos SCALANCE S. La duración aquí indicada comienza tras el inicio de sesión y tras renovarse esta en la página web del módulo de seguridad. • Ajuste estándar: 30 minutos • Valor mínimo: 5 minutos • Valor máximo: 480 minutos
Comentario	Entrada opcional de un comentario.

Crear roles

Sinopsis

A cada usuario se le puede asignar un rol definido por el sistema o por el usuario. Los derechos de módulos de un rol definido por el usuario se definen para cada módulo de seguridad.

Roles definidos por el sistema

Están predefinidos los siguientes roles definidos por el sistema. Los roles tienen asignados determinados derechos, que son iguales en todos los módulos y que el administrador no puede cambiar ni borrar.

- Administrator
  Rol estándar al crear una configuración de seguridad.
  Derechos de acceso ilimitados a todos los datos de configuración y módulos de seguridad.
- Standard
  Rol con derechos de acceso restringidos.
- Diagnose
  Rol predeterminado al crear un usuario.
  - Acceso de lectura a configuraciones.
  - Acceso de lectura al módulo de seguridad en el modo de operación "Online" con fines de test y diagnóstico.
- Remote-Access
  Puede utilizarse como plantilla para crear roles definidos por el sistema.

Encontrará una lista detallada de los derechos de configuración y módulos asignados a los roles definidos por el sistema "Administrador", "Estándar" y "Diagnóstico" en las tablas 1-3 a 1-7 del capítulo Administrar derechos (Página 767).

Rol definido por el usuario

Además de los roles definidos por el sistema, se pueden crear roles definidos por el usuario. Para un rol definido por el usuario se seleccionan los derechos de configuración o de módulos, y se definen los derechos de módulos correspondientes para cada módulo de seguridad

utilizado en el proyecto. El rol definido por el usuario se asigna manualmente al usuario en cuestión.

Cómo se accede a esta función

1. Haga doble clic en la entrada "Administración de usuarios" de los ajustes de seguridad globales.
2. Elija la ficha "Roles" en la administración de usuarios.

Tabla 8-7 Datos de la ficha "Roles"

Parámetro	Significado
Rol	Nombre de rol libre. Haga doble clic en la entrada "Agregar nuevo rol" para crear un rol personalizado. A continuación es posible especificar los derechos para el rol creado.
Descripción	Información sobre el rol definido por el sistema. En los roles personalizados se muestra la cadena de caracteres "Rol definido por el usuario".
Duración máxima de la sesión (solo para SCALANCE S V3 o superior)	Introducción de la duración tras la cual se cierra automáticamente la sesión para un usuario con el rol asignado en la página web para conjuntos de reglas IP específicos del usuario de módulos SCALANCE S. La duración aquí indicada comienza tras el inicio de sesión y tras renovarse esta en la página web del módulo de seguridad. • Ajuste estándar: 30 minutos • Valor mínimo: 5 minutos • Valor máximo: 480 minutos
Comentario	Entrada de un comentario opcional adicional.

Nota

Borrar roles

Un rol definido por el usuario solo puede borrarse si no está asignado a ningún usuario. Dado el caso, asigne un rol diferente al usuario.

Los roles definidos por el sistema no se pueden borrar.

Administrar derechos

Cómo se accede a esta función

1. Haga doble clic en la entrada "Administración de usuarios" de los ajustes de seguridad globales.
2. Elija la ficha "Roles" en la administración de usuarios.

Crear y asignar un rol definido por usuario

1. Haga doble clic en la entrada "Agregar rol".
2. Introduzca un nombre de rol.
3. Dado el caso, en la lista desplegable que lleva por título "<Copiar derechos de>" seleccione el rol definido por el sistema cuyos derechos deben utilizarse como plantilla para el rol definido por el usuario. Los roles definidos por el usuario no pueden seleccionarse en la lista desplegable.
   Resultado: En la lista de derechos de los roles de usuario están seleccionados aquellos derechos asignados al rol seleccionado definido por el sistema.
4. Active o desactive para cada módulo de seguridad los derechos que deben asignarse al rol definido por usuario.
5. Asigne el rol a un usuario en la ficha "Usuario".

Derechos de configuración

Los derechos de configuración son independientes de los módulos y controlan las autorizaciones para la configuración en STEP 7.

Dependiendo del tipo de usuario están disponibles los siguientes derechos de configuración:

Tabla 8-8 Derechos de configuración

Derecho de configuración	Administrator	Standard	Diagnose
Diagnosticar seguridad	x	x	x
Configurar seguridad	x	x	-
Administrar usuarios y roles	x	-	-

Derechos de módulos

Los derechos de módulos se configuran por módulo. En la columna "Servicio" se muestra el servicio al que se refiere el derecho en cuestión. Los comandos "Copiar permisos" e "Insertar permisos" del menú contextual permiten transferir los derechos de un módulo a otro.

Dependiendo del tipo de usuario están disponibles los siguientes derechos de módulos:

Tabla 8-9 Derechos del módulo CP x43-1 Advanced

Derecho dentro del servicio	Administrator	Standard	Diagnose	Servicio
Web: Formatear el sistema de archivos del CP *	x	-	-	Sistema de archivos
FTP: Leer archivos del sistema de archivos del CP	x	x	x	
FTP: Escribir archivos del sistema de archivos del CP	x	x	-	
FTP: Leer archivos (DB) de la CPU S7 **	x	x	x	PLC
FTP: Escribir archivos (DB) en la CPU S7 ***	x	x	-	
Applet: Leer variables de símbolos configurados *	x	x	x	
Applet: Escribir variables de símbolos configurados *	x	x	-	

Derecho dentro del servicio	Administrator	Standard	Diagnose	Servicio
Applet: Leer variables de direcciones absolutas *	x	x	x	
Applet: Escribir variables de direcciones absolutas *	x	x	-	
Applet: Leer estado de los módulos en el bastidor *	x	x	x	
Applet: Consultar las referencias de los módulos que hay en el rack *	x	x	x	
SNMP: Leer MIB II	x	x	x	SNMP
SNMP: Escribir MIB II	x	x	-	
SNMP: Leer MIB Automation	x	x	x	
SNMP: Leer MIB LLDP	x	x	x	
SNMP: Leer MIB SNMPv2	x	x	x	
SNMP: Leer MIB MRP	x	x	x	
SNMP: Escribir MIB MRP	x	x	-	
TIA Portal: Ejecutar diagnóstico del módulo de seguridad ****	x	x	x	Seguridad
Web: Ampliar lista de control de acceso IP *	x	-	-	Web
Web: Acceder al diagnóstico web y al sistema de archivos de CP	x	x	x	
Web: enviar un mensaje de correo electrónico de prueba *	x	x	x	
Web: Actualizar el firmware *	x	x	-	Mantenimiento
Web: Cargar textos de diagnóstico *	x	x	-	

Tabla 8-10 Derechos del módulo CP 1628

Derecho dentro del servicio	Administrator	Standard	Diagnose	Servicio
SNMP: Leer MIB II	x	x	x	SNMP
SNMP: Escribir MIB II	x	x	-	
SNMP: Leer MIB Automation	x	x	x	
SNMP: Leer MIB SNMPv2	x	x	x	
TIA Portal: Ejecutar diagnóstico del módulo de seguridad ****	x	x	x	Seguridad

Tabla 8-11 Derechos del módulo SCALANCE S

Derecho dentro del servicio	Administrator	Standard	Diagnose	Servicio
Carga de los archivos de configuración	x	x	-	Seguridad
TIA Portal: Ejecutar diagnóstico del módulo de seguridad ****	x	x	x	
SNMP: leer MIB Automation	x	x	x	SNMP
SNMP: leer MIB-II	x	x	x	
SNMP: escribir MIB-II	x	x	-	
SNMP: leer MRP-MIB	x	x	x	
SNMP: escribir MRP-MIB	x	x	-	
SNMP: leer MIB SNMPv2	x	x	x	
Web: actualizar el firmware	x	x	-	Mantenimiento

Tabla 8-12 Derechos del módulo CP 1543-1

Derecho dentro del servicio	Administrator	Standard	Diagnose	Servicio
FTP: Leer archivos del sistema de archivos del CP	x	x	x	Sistema de archivos
FTP: Escribir archivos del sistema de archivos del CP	x	x	-	
TIA Portal: Ejecutar diagnóstico del módulo de seguridad ****	x	x	x	Seguridad
SNMP: Leer MIB Automation	x	x	x	SNMP
SNMP: Leer MIB IPv6	x	x	x	
SNMP: Leer MIB LLDP	x	x	x	
SNMP: Leer MIB II	x	x	x	
SNMP: Escribir MIB II	x	x	-	
SNMP: Leer MIB SNMPv2	x	x	x	
FTP: Leer archivos (DB) de la CPU S7 **	x	x	x	PLC
FTP: Escribir archivos (DB) en la CPU S7 ***	x	x	-	

* Para aplicar la función, también debe estar activado el derecho de módulos "Web: acceder al diagnóstico web y al sistema de archivos del CP".

** Para aplicar la función, también debe estar activado el derecho de módulos "FTP: Leer archivos del sistema de archivos del CP".

*** Para aplicar la función, también debe estar activado el derecho de módulos "FTP: Escribir archivos del sistema de archivos del CP".

**** Para aplicar la función, también debe estar activado el derecho de configuración "Diagnosticar seguridad".

Ajustar derechos de módulos antes y después de crear módulos de seguridad

Dentro de un rol definido por el usuario, los derechos de módulos se definen por separado para cada módulo de seguridad. Si antes de agregar un rol se ha creado un módulo de seguridad para el que deben ajustarse derechos de módulos dentro de dicho rol, STEP 7 asigna derechos de módulos al módulo de seguridad en función de la plantilla de derechos seleccionada. Los derechos de módulos preajustados pueden adaptarse al agregar el rol. Si se ha creado un módulo de seguridad después de agregar un rol, no se preasignan derechos a dicho módulo de seguridad. En este caso hay que editar el rol existente y ajustar posteriormente uno mismo todos los derechos de módulos para el módulo de seguridad.

Los derechos de módulos ya existentes también pueden aplicarse a otro módulo copiándolos y adaptándolos allí si fuera necesario. Para ello, elija el comando "Copiar permisos" e "Insertar permisos" del menú contextual de un módulo en los derechos de módulos.

Generar datos de configuración para módulos SCALANCE M

Referencia

Se pueden generar informaciones de VPN para la parametrización de un SCALANCE M con STEP 7. Para ello es necesario que el módulo esté como mínimo en un grupo VPN con un módulo de seguridad o un SOFTNET Security Client. Con los archivos creados se puede configurar el SCALANCE M utilizando el Web Based Management del dispositivo.

Archivos generados

Se generan los siguientes tipos de archivos:

- Archivo de exportación con los datos de configuración
  - Tipo de archivo: archivo *.txt en formato ASCII
  - Contiene la información de configuración exportada para el SCALANCE M, incluida una información sobre los certificados generados adicionalmente.
- Certificados de grupo VPN del módulo
  - Tipo de archivo de la clave privada: archivo *.p12
  - El archivo contiene el certificado de módulo y el material de clave.
  - El acceso está protegido por contraseña.
- Certificados CA de grupos VPN
  - Tipo de archivo: archivo *.cer

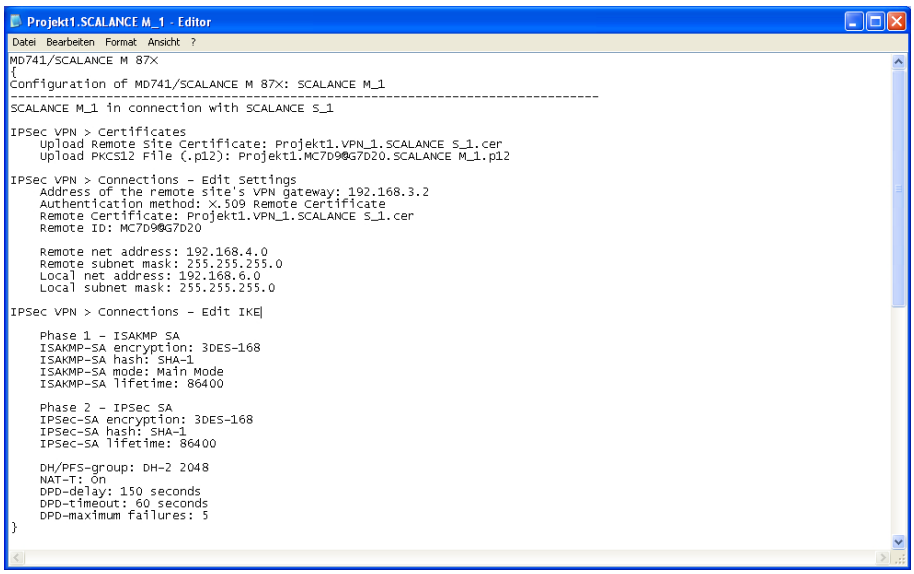


Figura 8-6 Archivo de configuración de SCALANCE M

Nota

No hay transferencia al módulo de seguridad

Los archivos de configuración no se transfieren al módulo de seguridad. Se genera un archivo ASCII con el que se puede configurar el SCALANCE M. Para ello, el SCALANCE M debe estar por lo menos en un grupo VPN con otro módulo de seguridad.

Proceda del siguiente modo

1. Seleccione el módulo del tipo "SCALANCE M".
2. En los ajustes de seguridad locales elija la entrada "Configuración de SCALANCE M".

3. Active la casilla de verificación "Generar archivos de SCALANCE M" y elija una ubicación para los archivos de configuración.
4. Compile la configuración del módulo SCALANCE M.
5. En el siguiente cuadro de diálogo, indique si para el archivo .p12 generado debe crearse una contraseña propia.
   Si elige "Cancelar", se asigna como contraseña el nombre del proyecto y no la contraseña del proyecto.

Resultado: los archivos (archivo .txt y certificados) se guardan en el directorio que ha indicado.

Configurar interfaces para módulos SCALANCE S

Sinopsis

Encontrará información sobre la configuración de las interfaces de módulos SCALANCE S en el apartado Auto-Hotspot del capítulo "SCALANCE S". La información incluida en dicho apartado explica las siguientes posibilidades de configuración:

- Modo de operación (modo Bridge / modo de enrutamiento): Ajustar el modo de operación (Página 844)
- Parámetros de dirección IP: Configurar parámetros de dirección IP (Página 844)
- Ajustes de puertos (solo para SCALANCE S ≥ V3): Configurar el modo de puerto (Página 846)
- Ajustes del Internet Service Provider (ISP) cuando una de las interfaces funciona vía PPPoE (solo para SCALANCE S ≥ V3): Configurar una conexión a Internet (Página 846)
- DNS dinámico (solo para SCALANCE S ≥ V3): Configurar DNS dinámico (Página 848)
- LLDP (solo para SCALANCE S ≥ V4 en modo de enrutamiento): Configurar un LLDP (Página 850)
- Redundancia de medios en topologías de anillo (cliente MRP, solo para SCALANCE S ≥ V4 en modo de enrutamiento): Auto-Hotspot

La configuración de las interfaces de los CPs se describe en los capítulos dedicados a los CPs.

Crear cortafuegos

Resumen del cortafuegos

Significado

La función de cortafuegos de los módulos de seguridad protege redes y estaciones de influencias externas e interferencias. De ese modo, solo se permiten relaciones de comunicación determinadas, definidas previamente. Los telegramas no autorizados son rechazados por el cortafuegos sin enviar una respuesta.

Para filtrar el tráfico de datos pueden utilizarse, entre otros, direcciones IP, subredes IP, números de puerto o direcciones MAC. Además se puede establecer una limitación del ancho de banda.

Las funciones de cortafuegos se pueden configurar para los siguientes niveles de protocolo:

- Cortafuegos IP con Stateful Packet Inspection (capa 3 y 4)
- Cortafuegos también para telegramas Ethernet "No IP" conforme a IEEE 802.3 (capa 2)

En un módulo de seguridad apto para VPN, el cortafuegos también puede emplearse para el tráfico de datos cifrado (túnel IPsec). En el módulo de seguridad SCALANCE S602 el cortafuegos solo puede utilizarse para el tráfico de datos no cifrado.

Reglas de cortafuegos

Las reglas de cortafuegos describen qué paquetes se permiten o prohíben en qué dirección.

Tipos de reglas de cortafuegos

- Conjuntos de reglas de cortafuegos globales: los conjuntos de reglas de cortafuegos globales se pueden asignar a varios módulos de seguridad al mismo tiempo. Los conjuntos de reglas de cortafuegos globales se configuran en los ajustes de seguridad globales.
- Reglas de cortafuegos locales: Las reglas de cortafuegos locales se configuran en los ajustes de seguridad locales de un módulo de seguridad.
- Conjuntos de reglas IP específicos del usuario (solo para SCALANCE S ≥ V3): los conjuntos de reglas IP específicos del usuario se pueden asignar a uno o varios módulos de seguridad al mismo tiempo. Los conjuntos de reglas IP específicos del usuario se configuran en los ajustes de seguridad globales, donde se asignan a uno o varios usuarios. Encontrará información sobre la configuración de conjuntos de reglas IP específicos del usuario para módulos SCALANCE S V3 en el siguiente capítulo:
  Auto-Hotspot

Definiciones de servicios

Adicionalmente se tiene la posibilidad de definir reglas de cortafuegos de forma compacta y clara con ayuda de definiciones de servicios. Las definiciones de servicios se configuran en los ajustes de seguridad globales y pueden utilizarse tanto en las reglas de cortafuegos globales como en las locales y en las específicas del usuario.

Adaptar reglas estándar para servicios IP

Para módulos SCALANCE S V3 o superior existe la posibilidad de adaptar las reglas de cortafuegos específicas del servicio que están ajustadas de forma estándar para las interfaces de los módulos de seguridad. Encontrará información sobre la configuración de estas reglas de cortafuegos en el apartado Adaptar reglas estándar para servicios IP (Página 860) del capítulo "SCALANCE S".

Reglas de cortafuegos generadas automáticamente para conexiones de CP

Para conexiones configuradas mediante CPs, STEP 7 crea automáticamente reglas de cortafuegos que autorizan la comunicación con el interlocutor correspondiente del CP en el sentido prescrito (CP activo/pasivo). Se tienen en cuenta los sentidos de establecimiento de las conexiones. Para visualizar dichas reglas de cortafuegos debe pulsarse el botón "Actualizar reglas de conexión" estando activado el modo de cortafuegos avanzado. Las reglas de cortafuegos se mostrarán seguidamente en el modo de cortafuegos avanzado.

En los apartados siguientes se explica qué reglas de cortafuegos se generan automáticamente.

- Para CPs S7-300/S7-400/PC: Reglas de cortafuegos automáticas referidas a conexiones (Página 885) en el capítulo "Seguridad para CPs S7-300/S7-400/PC".
- Para CPs S7-1500: Reglas de cortafuegos automáticas referidas a conexiones (Página 894) en el capítulo "Seguridad para CPs S7-1500".

Activar cortafuegos

En los ajustes de seguridad locales, las funciones de cortafuegos para un módulo de seguridad determinado se controlan mediante la casilla de verificación "Activar cortafuegos". Si la casilla de verificación está activada es posible configurar el cortafuegos, que será efectivo tras el proceso de carga. Si el módulo de seguridad está en un grupo VPN, la casilla de verificación "Activar cortafuegos" está activada de forma predeterminada y no puede desactivarse. Tras cambiar al modo de cortafuegos avanzado no es posible regresar al modo normal. Encontrará información más detallada sobre el modo normal y el modo de cortafuegos avanzado en el capítulo:

Resumen de reglas de cortafuegos locales (Página 784).

Conjuntos de reglas de cortafuegos globales

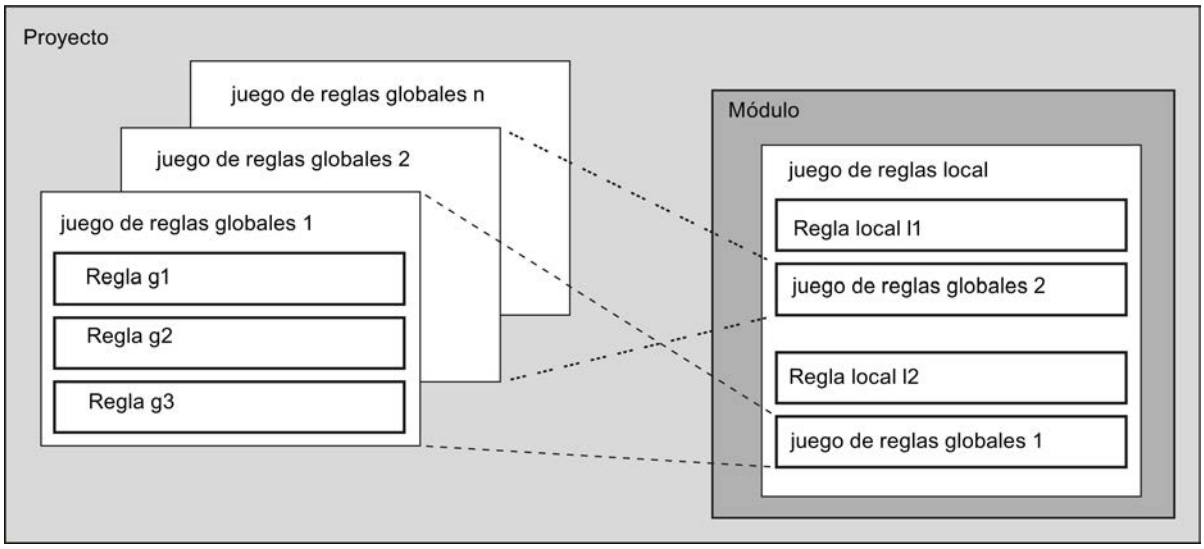
Aplicación

Los conjuntos de reglas de cortafuegos globales se configuran en los ajustes de seguridad globales en función del módulo. Un conjunto de reglas de cortafuegos consta de una o varias reglas de cortafuegos y se asigna a módulos de seguridad individuales.

Dentro de los conjuntos de reglas de cortafuegos globales, se distingue entre:

- conjuntos de reglas IP
- conjuntos de reglas MAC

La representación siguiente ilustra la relación entre los conjuntos de reglas de definición global y los conjuntos de reglas utilizados a nivel local.



Configuración

Al configurar reglas de cortafuegos globales es posible realizar un ajuste detallado del cortafuegos. Existe la posibilidad de activar servicios determinados para dispositivos concretos o habilitar para el dispositivo todos los servicios para el acceso a la estación o a la red.

¿Cuándo son convenientes las reglas de cortafuegos IP y MAC?

Las reglas de cortafuegos globales son convenientes cuando se desean definir criterios de filtrado idénticos para la comunicación.

Nota

Asignación de conjuntos de reglas de cortafuegos con reglas de cortafuegos incompatibles

Para un módulo de seguridad solo se aplican correctamente las reglas de conjuntos de reglas de cortafuegos que realmente son soportadas por el módulo de seguridad. Una regla incluida en un conjunto de reglas de cortafuegos global con el sentido "De: Externa" o "A: Any" no se asigna a un CP 1628. Las demás reglas del conjunto de reglas de cortafuegos global se aplican siempre que el CP las soporte.

Conjuntos de reglas de cortafuegos globales - Convenios

Los conjuntos de reglas de cortafuegos globales se utilizan localmente

Para la creación de un conjunto de reglas de cortafuegos global y la asignación a un módulo rigen los siguientes convenios:

- Vista de configuración
  Los conjuntos de reglas de cortafuegos globales se configuran en los ajustes de seguridad globales.
- Prioridad
  Por defecto, las reglas definidas localmente tienen mayor prioridad que los conjuntos de reglas de cortafuegos IP y MAC globales. Por consiguiente, los conjuntos de reglas de cortafuegos IP y MAC globales que se asignan de nuevo se insertan en un principio en último lugar de la lista de reglas local.
  La prioridad se puede modificar cambiando el emplazamiento en la lista de reglas.
- Introducir, modificar o borrar conjuntos de reglas
  Los conjuntos de reglas de cortafuegos globales no se pueden editar en la lista de reglas de cortafuegos locales de las propiedades del módulo. Allí solo se pueden ver y emplazar según la prioridad deseada
  Una regla individual no puede borrarse de un conjunto de reglas asignado en los ajustes de seguridad locales. Solo es posible eliminar de la lista de reglas local todo el conjunto de reglas. Los conjuntos de reglas de cortafuegos de los ajustes de seguridad globales no se ven afectados.

Crear conjuntos de reglas de cortafuegos globales

Cómo se accede a esta función

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Juegos de reglas globales para el cortafuegos" > "Conjuntos de reglas IP" o "Conjuntos de reglas MAC".
   Resultado: Debajo de la entrada seleccionada se muestran los conjuntos de reglas IP o MAC creados anteriormente.
2. Haga doble clic en la entrada "Agregar conjunto de reglas IP" o "Agregar conjunto de reglas MAC".
3. Introduzca los datos siguientes:
   - Nombre: designación unívoca del conjunto de reglas para todo el proyecto. El nombre aparece en la lista de reglas local del módulo de seguridad tras la asignación del conjunto de reglas.
   - Descripción (opcional): Introduzca la descripción del conjunto de reglas global.
4. Introduzca en la lista, por orden, las reglas de cortafuegos.
   Tenga en cuenta la descripción de los parámetros en los capítulos siguientes:
   Para conjuntos de reglas IP: Definir reglas de filtrado de paquetes IP (Página 785)
   Para conjuntos de reglas MAC: Definir reglas para filtrado de paquetes MAC (Página 788)

Resultado

Se ha creado el conjunto de reglas globales para el cortafuegos, que puede asignarse a los módulos de seguridad que se desee.
Tenga en cuenta las descripciones del capítulo siguiente:

Asignar conjuntos de reglas de cortafuegos globales (Página 777)

Asignar conjuntos de reglas de cortafuegos globales

Requisitos

Se ha activado el módulo de cortafuegos avanzado para los módulos de seguridad que deben asignarse a un conjunto de reglas de cortafuegos.

Procedimiento

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Juegos de reglas globales para el cortafuegos" > "Asignar módulo a un conjunto de reglas de cortafuegos".
2. En la lista desplegable "Conjunto de reglas" seleccione el conjunto de reglas al que desee asignar el módulo de seguridad.
   En la tabla de la derecha se muestran los módulos de seguridad que pueden asignarse al conjunto de reglas de cortafuegos seleccionado. En la tabla de la izquierda se muestran los módulos de seguridad que ya están asignados al conjunto de reglas de cortafuegos seleccionado.
3. En el área "Módulos disponibles" seleccione los módulos de seguridad que desee asignar al conjunto de reglas seleccionado.
4. Haga clic en el botón << para asignar los módulos seleccionados al conjunto de reglas seleccionado.

Resultado

Los módulos de seguridad asignados utilizan el conjunto de reglas global como conjunto de reglas local; el conjunto de reglas global aparece automáticamente al final de la lista de reglas de cortafuegos de los ajustes de seguridad locales.

Servicios IP

Definir servicios IP

Cómo se accede a esta función

En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas IP".

Procedimiento

La definición de los servicios IP permite definir de forma compacta y clara reglas de cortafuegos que se aplican a servicios determinados. Para esto se adjudica un nombre, al que se asignan los parámetros de servicio.

Además, los servicios así definidos se pueden reunir a su vez en grupos, con un nombre de grupo.

Para la configuración de las reglas de filtrado de paquetes se utiliza entonces ese nombre.

Parámetros para servicios IP

Los servicios IP se definen a través de los siguientes parámetros:

Tabla 8-13 Servicios IP: Parámetros

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Nombre	Nombre para el servicio; se utiliza para la identificación en la definición de reglas o en el agrupamiento. Los nombres de servicios predefinidos no pueden modificarse.	• El nombre debe empezar con una letra. • El nombre no puede contener caracteres especiales. • El nombre no puede estar repetido.
Protocolo	Selecciona el tipo de protocolo	• TCP • UDP • Todos
Puerto de origen	Se filtra a partir del número de puerto aquí indicado; este define el acceso al servicio para el destinatario de los telegramas.	Ejemplos: *: Puerto no se comprueba 20 o 21: Servicio FTP
Puerto de destino	Se filtra a partir del número de puerto aquí indicado; este define el acceso al servicio para el destinatario de los telegramas.	Ejemplos: *: Puerto no se comprueba TCP 80: Web-HTTP-Service TCP 102: protocolo S7

Definir servicios ICMP

Cómo se accede a esta función

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas IP".
2. Seleccione la ficha "ICMP".

Procedimiento

Con ayuda de la definición de los servicios ICMP se pueden definir de forma compacta y clara reglas de cortafuegos que se aplican a servicios determinados. Para esto se adjudica un nombre, al que se asignan los parámetros de servicio.

Además, los servicios definidos de este modo se pueden reunir en grupos, con un nombre de grupo.

Para la configuración de las reglas de filtrado de paquetes se utiliza entonces ese nombre.

Parámetros para servicios ICMP

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Nombre	Nombre de libre definición para el servicio; se utiliza para la identificación en la definición de reglas o en el agrupamiento. Los nombres de servicios ICMPv6 predefinidos no pueden modificarse.	• El nombre debe empezar con una letra. • El nombre no puede contener caracteres especiales. • El nombre debe tener 20 caracteres como máximo. • El nombre no puede estar repetido.
ICMPv6	Si se activa esta casilla de verificación, el servicio ICMP se declara como servicio ICMPv6 y es posible seleccionar para el servicio un tipo y un código específicos de ICMPv6. Un servicio ICMPv6 solo puede utilizarse en la regla de cortafuegos de un módulo de seguridad que soporte IPv6.	• activado • desactivado (predeterminado)
Tipo	Tipo del mensaje ICMPv4 o ICMPv6.	Si la casilla de verificación "ICMPv6" está desactivada puede seleccionarse tipos específicos de ICMPv4. Si la casilla de verificación está activada puede seleccionarse tipos específicos de ICMPv6.
Código	Código del tipo ICMP.	Los valores dependen del tipo seleccionado.

Crear grupos de servicios

Cómo se accede a esta función

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas IP".
2. Seleccione la ficha "Grupos de servicios".

Formación de grupos de servicios

Puede agrupar varios servicios formando grupos de servicios. De este modo se crean servicios más complejos que se pueden utilizar en las reglas de filtrado de paquetes seleccionando simplemente un nombre. Es posible reunir en un grupo servicios IPv4 e IPv6.

Los grupos se crean en la ficha "Grupos de servicios" abierta. A continuación asigne los servicios a un grupo en la ficha "Administración de grupos".

Proceda del siguiente modo

1. Primero cree en esta ficha grupos con un nombre adecuado a sus exigencias y un texto descriptivo opcional.
2. Seleccione luego la ficha "Administración de grupos". Allí puede asignar a los grupos aquí definidos los servicios IP establecidos con anterioridad.

Administrar grupos de servicios

Cómo se accede a esta función

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas IP".
2. Seleccione la ficha "Administración de grupos".

Formación de grupos de servicios

Puede agrupar varios servicios formando grupos de servicios. De este modo se crean servicios más complejos que se pueden utilizar en las reglas de filtrado de paquetes seleccionando simplemente un nombre. Es posible reunir en un grupo servicios IPv4 e IPv6.

En la ficha "Administración de grupos" asigne a un grupo de servicios seleccionado servicios que haya creado previamente en la ficha "Grupos de servicios".

Proceda del siguiente modo

1. En esta ficha seleccione primero, por medio de la lista desplegable "Grupos de servicios", un grupo que haya creado con anterioridad en la ficha "Grupos de servicios".
2. Luego asigne al grupo los servicios deseados del campo de selección de la derecha "Servicios disponibles".

Servicios MAC

Definir servicios MAC

Cómo se accede a esta función

En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas MAC".

Significado

La definición de los servicios MAC permite definir de forma compacta y clara reglas de cortafuegos que se aplican a servicios determinados. Se adjudica un nombre, al que se asignan los parámetros de servicio.

Además, los servicios definidos de este modo se pueden reunir en grupos, con un nombre de grupo.

Para la configuración de las reglas de filtrado de paquetes globales o locales se utilizan entonces esos nombres.

Parámetros para servicios MAC

Una definición de servicios MAC se forma mediante parámetros MAC específicos del protocolo:

Tabla 8-14 Parámetros de servicios MAC

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Nombre	Nombre de libre definición para el servicio; se utiliza para la identificación en la definición de reglas o en el agrupamiento.	• El nombre debe empezar con una letra. • El nombre no puede contener caracteres especiales. • El nombre no puede estar repetido.
Protocolo	Nombre del tipo de protocolo: • ISO ISO designa telegramas con las siguientes propiedades: Lengthfield <= 05DC (hex), DSAP= userdefined SSAP= userdefined CTRL= userdefined • SNAP SNAP designa telegramas con las siguientes propiedades: Lengthfield <= 05DC (hex), DSAP=AA (hex), SSAP=AA (hex), CTRL=03 (hex), OUI=userdefined, OUI-Type=userdefined • PROFINET IO • Como alternativa también es posible introducir un número de protocolo. Las entradas de protocolo 0800 (hex) y 0806 (hex) no se aceptan, ya que estos valores se aplican a telegramas IP y ARP, respectivamente.	• ISO • SNAP • PROFINET IO • 0x (entrada de código)
DSAP	Destination Service Access Point: Dirección de destinatario LLC	
SSAP	Source Service Access Point: Dirección de remitente LLC	
CTRL	LLC Control Field	
OUI	Organizationally Unique Identifier (los tres primeros bytes de la dirección MAC = identificación del fabricante)	
Tipo OUI	Tipo/Identificación de protocolo	

Nota

Procesamiento para CPs S7

Solo se procesan ajustes para frames ISO con DSAP=SSAP=FE (hex). Cualquier otro tipo de frame no es relevante para CPs S7, por lo que ya es rechazado por el cortafuegos antes del procesamiento.

Ajustes especiales para servicios SIMATIC NET

Utilice para el filtrado de servicios especiales SIMATIC NET los siguientes ajustes de protocolo:

- DCP (Primary Setup Tool) :
  PROFINET
- SiClock :
  OUI= 08 00 06 (hex), OUI-Type= 01 00 (hex)

Crear grupos de servicios

Cómo se accede a esta función

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas MAC".
2. Seleccione la ficha "Grupos de servicios".

Formación de grupos de servicios

Puede agrupar varios servicios formando grupos de servicios. De este modo se crean servicios más complejos que se pueden utilizar en las reglas de filtrado de paquetes seleccionando simplemente un nombre.

Los grupos se crean en la ficha "Grupos de servicios" abierta. A continuación asigne los servicios a un grupo en la ficha "Administración de grupos".

Proceda del siguiente modo

1. Primero cree en esta ficha grupos con un nombre adecuado a sus exigencias y un texto descriptivo opcional.
2. Seleccione luego la ficha "Administración de grupos". Allí puede asignar a los grupos aquí definidos los servicios MAC establecidos con anterioridad.

Administrar grupos de servicios

Cómo se accede a esta función

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Servicios" > "Definir servicios para reglas MAC".
2. Seleccione la ficha "Administración de grupos".

Formación de grupos de servicios

Puede agrupar varios servicios formando grupos de servicios. De este modo se crean servicios más complejos que se pueden utilizar en las reglas de filtrado de paquetes seleccionando simplemente un nombre.

En la ficha "Administración de grupos" asigne a un grupo de servicios seleccionado servicios que haya creado previamente en la ficha "Grupos de servicios".

Proceda del siguiente modo

1. En esta ficha seleccione primero, por medio de la lista desplegable "Grupos de servicios", un grupo que haya creado con anterioridad en la ficha "Grupos de servicios".
2. Luego asigne al grupo los servicios deseados del campo de selección de la derecha "Servicios disponibles".

Resumen de reglas de cortafuegos locales

Significado

Las reglas de cortafuegos locales se configuran en los ajustes de seguridad locales de un módulo de seguridad y solo son válidas para dicho módulo de seguridad. Después de activar la funcionalidad de cortafuegos es posible utilizar reglas de cortafuegos predefinidas o bien definir nuevas reglas de cortafuegos en el modo de cortafuegos avanzado.

Utilizar reglas de cortafuegos predefinidas

Aquí se recurre a reglas de cortafuegos sencillas y predefinidas. Solo pueden habilitarse reglas específicas de servicio. Los servicios habilitados están permitidos para todos los dispositivos en el sentido indicado. Encontrará información detallada sobre la definición de reglas de cortafuegos de esta ventana de diálogo en los capítulos siguientes específicos del módulo:

- Para SCALANCE S: Auto-Hotspot
- Para CPs S7-300/S7-400/PC: Auto-Hotspot
- Para CPs S7-1500: Auto-Hotspot

Definir reglas de cortafuegos en modo de cortafuegos avanzado

En el modo de cortafuegos avanzado se pueden definir ajustes de cortafuegos detallados. Existe la posibilidad de activar servicios determinados para dispositivos concretos o habilitar para el dispositivo todos los servicios para el acceso a la estación o a la red. Active el modo de cortafuegos avanzado mediante la casilla de verificación "Activar cortafuegos en modo avanzado". Seguidamente, en los ajustes de seguridad locales es posible configurar las reglas de cortafuegos con "Cortafuegos" > "Reglas IP" o "Reglas MAC". Consulte en detalle las posibilidades de configuración que existen en los capítulos siguientes:

Para reglas de filtrado de paquetes IP: Definir reglas de filtrado de paquetes IP (Página 785)

Para reglas de filtrado de paquetes MAC: Definir reglas para filtrado de paquetes MAC (Página 788)

Nota

No es posible desactivar el modo de cortafuegos avanzado

Una vez que se ha activado el modo de cortafuegos avanzado ya no se puede desactivar.

Capacidad

Número de reglas de cortafuegos (modo de cortafuegos avanzado)	
SCALANCE S	256 como máximo
CPs S7	256 como máximo
CP 1628	256 como máximo

Definir reglas de filtrado de paquetes IP

Significado

Utilizando reglas de filtrado de paquetes IP se puede filtrar por telegramas IP, p. ej. TCP, UDP o ICMP.

Dentro de una regla de filtrado de paquetes se puede acceder a definiciones de los servicios IP.

Registrar reglas de filtrado de paquetes IP

Introduzca en la lista, por orden, las reglas de cortafuegos. Tenga en cuenta la descripción de parámetros que sigue.

Tabla 8-15 Reglas IP: Parámetros

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Acción	Definición de la autorización (habilitación/bloqueo)	• Allow Autorizar telegramas según definición. • Drop Bloquear telegramas según definición. Para reglas de cortafuegos que se crearon automáticamente mediante una configuración de conexión y después se adaptaron manualmente: • Allow* • Drop* Si se modifican reglas de conexión creadas automáticamente, al seleccionar la opción "*" no volverán a generarse ni sobrescribirse desde STEP 7.
De/A	Selecciona los sentidos de comunicación para los que debe regir la regla.	Se describen en apartados separados. • Para módulos SCALANCE S: Sentidos de filtrado de paquetes IP SCALANCE S (Página 859) • Para CPs S7-300/S7-400/PC: Sentidos de filtrado de paquetes IP CPs S7-300/S7-400/PC (Página 883) • Para CPs S7-1500: Sentidos de filtrado de paquetes IP para CPs S7-1500 (Página 893)
IPv6 (solo para CP 1543-1)	Si activa esta casilla de verificación podrá utilizar en las reglas de cortafuegos un servicio ICMPv6 que se haya definido anteriormente. Para CP 1543-1 V1.1 es posible introducir adicionalmente direcciones IPv6 en los campos de entrada "Dirección IP de origen" y "Dirección IP de destino" después de activar la casilla de verificación. Para CP 1543-1 V1.0 no es posible introducir ninguna dirección IP de origen ni de destino para la regla de cortafuegos después de activar la casilla de verificación.	La casilla de verificación solo puede activarse y desactivarse si no hay entradas en los campos "Dirección IP de origen" y "Dirección IP de destino". Si se ha desactivado IPv6 en los ajustes locales del CP 1543-1, no es posible activar la casilla de verificación "IPv6" en los ajustes de seguridad locales del CP y, por tanto, tampoco es posible utilizar los servicios ICMPv6 o las direcciones IPv6 en las reglas del cortafuegos. Las reglas de cortafuegos existentes que utilizan IPv6 se representan atenuadas cuando IPv6 está desactivado.
Dirección IP de origen	La regla del cortafuegos se aplica a los telegramas cuyo remitente tiene la dirección IP indicada aquí. Si no se indica ninguna dirección IP, la regla del cortafuegos es válida para todos los dispositivos dentro del sentido de comunicación seleccionado en la columna "De".	Encontrará más información sobre direcciones IP en el apartado Direcciones IP en reglas de filtrado de paquetes IP (Página 790).

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Dirección IP de destino	La regla del cortafuegos se aplica a los telegramas cuyo destinatario tiene la dirección IP indicada aquí. Si no se indica ninguna dirección IP, la regla del cortafuegos es válida para todos los dispositivos dentro del sentido de comunicación seleccionado en la columna "A".	
Servicio	Nombre del servicio IP/ICMP o del grupo de servicios utilizado. Seleccione aquí uno de los servicios que ha definido en el cuadro de diálogo "Servicios IP": • Servicios IP o bien • Servicios ICMP Antes de seleccionar un servicio ICMPv6 hay que activar a casilla de verificación "IPv6".	En la lista desplegable se pueden seleccionar los servicios y grupos de servicios configurados en los ajustes de seguridad globales.
Ancho de banda (Mbits/s)	Posibilidad de ajuste para una limitación del ancho de banda Solo puede introducirse si para la acción está seleccionado "Allow". Un paquete pasa el cortafuegos si la regla "Allow" es correcta y si no se ha sobrepasado aún el ancho de banda permitido para esa regla.	CP x43-1, CP 1543-1 y SCALANCE S < V3.0: 0,001 ... 100 Mbits/s CP 1628 y SCALANCE S ≥ V3.0: 0,001 ... 1000 Mbits/s Para reglas globales y específicas de usuario: 0,001 ... 100 Mbits/s Nota: si se configura el sentido "De túnel a estación" en una regla de cortafuegos para el CP 1543-1 no es posible indicar ninguna limitación del ancho de banda.
Registro	Activación o desactivación del inicio de sesión para esta regla. Si el inicio de sesión está activado rigen los ajustes para el registro de filtros de paquetes que se han configurado en los ajustes de seguridad locales.	• activado • desactivado (predeterminado)
Número	Número de la regla asignado automáticamente	
Comentario	Espacio para explicación propia de la regla	Si un comentario está marcado con "AUTO", significa que ha sido creado para una regla de conexión automática. Para reglas de creación propia, la entrada de un comentario es opcional.

Tabla 8-16 Significados de las entradas del menú contextual

Entrada del menú contextual	Significado
Borrar	Sirve para borrar la regla seleccionada o el conjunto de reglas seleccionado. Indicaciones sobre la eliminación de un conjunto de reglas definido como global y asignado como local: Al eliminar el conjunto de reglas solo se anula la asignación al módulo de seguridad.
Guardar como conjunto de reglas global (solo para reglas de cortafuegos locales)	Copia la o las reglas de cortafuegos seleccionadas y las inserta en los ajustes de seguridad globales como conjunto de reglas global. Este proceso no afecta a la configuración de cortafuegos actual definida para el módulo de seguridad.
Mover hacia arriba	Sirve para mover la regla seleccionada o el conjunto de reglas seleccionado una posición hacia arriba en la lista. De esta forma la regla o el conjunto de reglas desplazado se trata con una prioridad mayor.
Mover hacia abajo	Sirve para mover la regla seleccionada o el conjunto de reglas seleccionado una posición hacia abajo en la lista. De esta forma la regla o el conjunto de reglas desplazado se trata con una prioridad menor.
Definir servicio para reglas IP	Sirve para abrir un cuadro de diálogo en el que se administran los servicios IP y los grupos de servicios.

Definir reglas para filtrado de paquetes MAC

Significado

Con reglas de filtrado de paquetes MAC se filtran telegramas MAC.

Dentro de una regla de filtrado de paquetes se puede acceder a definiciones de los servicios MAC.

Introducir reglas para el filtrado de paquetes MAC

Introduzca en la lista, por orden, las reglas de cortafuegos. Tenga en cuenta la descripción de parámetros que sigue.

Tabla 8-17 Reglas MAC: Parámetros

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Acción	Definición de la autorización (habilitación/bloqueo)	• Allow Autorizar telegramas según definición. • Drop Bloquear telegramas según definición.
De/A	Selecciona los sentidos de comunicación para los que debe regir la regla.	Se describen en apartados separados. • Para módulos SCALANCE S: Sentidos de filtrado de paquetes MAC SCALANCE S (Página 859) • Para CPs S7-300/S7-400/PC: Sentidos de filtrado de paquetes MAC CPs S7-300/S7-400/PC (Página 883) • Para CPs S7-1500: Sentidos de filtrado de paquetes MAC para CPs S7-1500 (Página 894)

Parámetro	Significado/comentario	Posibilidades de selección / Rangos de valores
Dirección MAC de origen	La regla del cortafuegos se aplica a los telegramas cuyo remitente tiene la dirección MAC indicada aquí. Si no se indica ninguna dirección MAC, la regla del cortafuegos es válida para todos los dispositivos dentro del sentido de comunicación seleccionado en la columna "De".	Dirección MAC en formato correcto
Dirección MAC de destino	La regla del cortafuegos se aplica a los telegramas cuyo destinatario tiene la dirección MAC indicada aquí. Si no se indica ninguna dirección MAC, la regla del cortafuegos es válida para todos los dispositivos dentro del sentido de comunicación seleccionado en la columna "A".	
Servicio	Nombre del servicio MAC o del grupo de servicios utilizado	En la lista desplegable se pueden seleccionar los servicios y grupos de servicios configurados.
Ancho de banda (Mbits/s)	Posibilidad de ajuste para una limitación del ancho de banda. Solo puede introducirse si para la acción está seleccionado "Allow". Un paquete pasa el cortafuegos si la regla "Allow" es correcta y si no se ha sobrepasado aún el ancho de banda permitido para esa regla.	CP x43-1, CP 1543-1 y SCALANCE S < V3.0: 0,001 ... 100 Mbits/s CP 1628 y SCALANCE S ≥ V3.0: 0,001 ... 1000 Mbits/s Para reglas globales: 0,001 ... 100 Mbits/s Nota: si se configura el sentido "De túnel a estación" en una regla de cortafuegos para el CP 1543-1 no es posible indicar ninguna limitación del ancho de banda.
Registro	Activación o desactivación del registro para esta regla. Si el inicio de sesión está activado rigen los ajustes para el registro de filtros de paquetes que se han configurado en los ajustes de seguridad locales.	• activado • desactivado (predeterminado)
Número	Número de la regla asignado automáticamente	
Comentario	Espacio para explicación propia de la regla	Si un comentario está marcado con "AUTO", significa que ha sido creado para una regla de conexión automática. Para reglas de creación propia, la entrada de un comentario es opcional.

Tabla 8-18 Significado de los comandos de menú

Botón	Significado
Borrar	Sirve para eliminar la regla seleccionada o el conjunto de reglas global seleccionado. Indicaciones sobre la eliminación de un conjunto de reglas definido como global y asignado como local: Al eliminar el conjunto de reglas solo se anula la asignación al módulo de seguridad.
Guardar como conjunto de reglas global (solo para reglas de cortafuegos locales)	Copia la o las reglas de cortafuegos seleccionadas y las inserta en los ajustes de seguridad globales como conjunto de reglas global. Este proceso no afecta a la configuración de cortafuegos actual definida para el módulo de seguridad.

Botón	Significado
Mover hacia arriba	Sirve para mover la regla seleccionada o el conjunto de reglas global seleccionado una posición hacia arriba en la lista. De esta forma la regla o el conjunto de reglas desplazado se trata con una prioridad mayor.
Mover hacia abajo	Sirve para mover la regla seleccionada o el conjunto de reglas global seleccionado una posición hacia abajo en la lista. De esta forma la regla o el conjunto de reglas desplazado se trata con una prioridad menor.
Definir servicio para reglas MAC	Sirve para abrir un cuadro de diálogo en el que se administran los servicios MAC y los grupos de servicios.

Direcciones IP en reglas de filtrado de paquetes IP

Entrada de direcciones IP en reglas de filtrado de paquetes IP

En las reglas de filtrado de paquetes IP existen las siguientes posibilidades de introducir direcciones IP:

- ninguna indicación
  La regla es válida para todas las direcciones IP.
- una dirección IP
  La regla es válida exactamente para la dirección indicada.
- Rango de direcciones
  La regla es válida para todas las direcciones IP incluidas en el rango de direcciones.
  Un rango de direcciones se define indicando la cantidad de posiciones de bits válidas en la dirección IP, a saber en la siguiente forma: [Dirección IP]/[Cantidad de bits a considerar]
  - [Dirección IP]/24 significa por consiguiente que solo los 24 bits de mayor valor de la dirección IP se tienen en cuenta en la regla de filtrado. Se trata de las tres primeras posiciones de la dirección IP.
  - [Dirección IP]/25 significa que solo se tienen en cuenta en la regla de filtrado las tres primeras posiciones y el bit de valor más alto de la cuarta posición de la dirección IP.
- Área de direcciones
  Para la dirección IP de origen puede introducirse un área de direcciones con el formato siguiente:
  [Dirección IP inicial]-[dirección IP final]

Direcciones IPv4

Una dirección IPv4 consta de 4 números decimales comprendidos en el rango de 0 a 255 y separados entre ellos por un punto.

Tabla 8-19 Ejemplos de rangos de direcciones IPv4

Dirección IP de origen o dirección IP de destino	Rango de direcciones		Número de direcciones *)
	de	a	
192.168.0.0/**16**	192.168.0.0	192.168.255.255	65.536
192.168.10.0/**24**	192.168.10.0	192.168.10.255	256

Dirección IP de origen o dirección IP de destino	Rango de direcciones		Número de direcciones *)
	de	a	
192.168.10.0/25	192.168.10.0	192.168.10.127	128
192.168.10.0/26	192.168.10.0	192.168.10.63	64
192.168.10.0/27	192.168.10.0	192.168.10.31	32
192.168.10.0/28	192.168.10.0	192.168.10.15	16
192.168.10.0/29	192.168.10.0	192.168.10.7	8
192.168.10.0/30	192.168.10.0	192.168.10.3	4
*) Nota: Tenga en cuenta que la dirección de red y la dirección Broadcast de un rango de direcciones no están disponibles como direcciones IP de dispositivos de red.			

Direcciones IPv6

Las direcciones IPv6 constan de 8 bloques de cuatro cifras hexadecimales cada uno (128 bits en total). Los bloques están separados por dos puntos. Las direcciones IPv6 solo pueden introducirse en reglas de filtrado de paquetes IP para el CP 1543-1 V1.1.

Ejemplo: fd00:ffff:ffff:ffff:ffff:ffff:2f33:8f21

Reglas / simplificaciones:

- Está permitido omitir los ceros a la izquierda dentro de un bloque.
  Ejemplo: en lugar de 2001:0db8:2426:08d3:1457:8a2e:0070:7344 también es posible la notación 2001:db8:2426:8d3:1457:8a2e:70:7344.
- Si uno o varios bloques tienen el valor 0 (o 0000) se admite una notación abreviada.
  Ejemplo: en lugar de 2001:0db8:0:0:0:0:1428:57ab también es posible la notación 2001:db8::1428:57ab.
  Con el fin de conservar la univocidad, dicha abreviación solo puede aplicarse una vez dentro de toda la dirección.
- Notación decimal con puntos
  Para los 2 últimos bloques o 4 bytes es posible utilizar la notación decimal tradicional con puntos.
  Ejemplo: la dirección IPv6 fd00::ffff.125.1.0.1 es equivalente a fd00::ffff:7d01:1.
- Notación de rango de direcciones en reglas de filtrado de paquetes IP: de forma análoga a las direcciones IPv4, también es posible la notación de direcciones IPv6 en forma de rangos de direcciones.
  Ejemplo: la entrada "2001:0db8:85a3:08d3:1319:8a2e:0:0 /96" incluye todas las direcciones IPv6 de 2001:0db8:85a3:08d3:1319:8a2e:0:0 a 2001:0db8:85a3:08d3:1319:8a2e:ffff:ffff.

Realizar ajustes de registro específicos del módulo

Resumen de ajustes de registro

Ajustes de registro en la configuración

Los ajustes de registro aquí adoptados se cargan en el módulo con la configuración y se activan al arrancar el módulo de seguridad.

Los ajustes configurados para el registro de filtrado de paquetes se pueden restringir, en caso necesario, en las funciones online. Por ejemplo, si se ha configurado un registro IP y MAC, en las funciones online puede ajustarse que únicamente se muestre el registro IP.

Métodos de registro y clases de eventos

Aquí puede definir qué datos deben ser registrados. De este modo activa ya el registro al cargar la configuración en el módulo de seguridad.

Además elige en la configuración uno de los métodos de registro o ambos:

- registro local
- red Syslog

El módulo de seguridad reconoce para cada uno de los dos métodos de registro los tres tipos de eventos siguientes:

- eventos de filtrado de paquetes
- eventos de auditoría
- eventos de sistema

Configurar registro local

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Configuración del registro" > "Local log store".

Configurar registro local

Tabla 8-20 Registro local - Ajustes para eventos de registro

Evento de registro	Significado	Observaciones
Registro de filtros de paquetes (cortafuegos)	El registro de filtros de paquetes registra determinados paquetes del tráfico de datos. Solamente se registran los paquetes de datos afectados por una regla de filtrado de paquetes configurada (cortafuegos), o para los que reacciona la protección básica (paquetes corruptos o no válidos). Condición para ello es que esté activado el registro para la regla de filtrado de paquetes. Puede definir la cantidad de paquetes de datos registrados en la lista desplegable "Paquetes para registrar": • "Todos los paquetes": se registran los paquetes de datos para los que es válida una regla de cortafuegos configurada. Además, los paquetes de respuesta se registran en los paquetes que han pasado el cortafuegos según una regla Allow configurada. • "Paquetes generadores de estado": se registran únicamente los paquetes de datos a los que afecta una regla de cortafuegos configurada y que inicialmente generan un estado en el cortafuegos. Los paquetes de datos que han pasado el cortafuegos utilizando este estado de cortafuegos, no se registran.	Los datos del registro de filtros de paquetes no son remanentes Los datos se guardan en una memoria volátil del módulo de seguridad, por lo que dejan de estar disponibles tras una desconexión de la alimentación eléctrica. Para el guardado remanente es posible guardar también en un archivo los datos de registro visualizados en el cuadro de diálogo "Online y diagnóstico".
Registro de auditoría	El registro de eventos de auditoría está siempre activado. Se almacena siempre en la memoria cíclica. El registro de auditoría registra de manera automática y continua eventos relevantes para la seguridad, por ejemplo acciones del usuario como activación o desactivación del registro de paquetes o carga de configuraciones en el módulo de seguridad.	Los datos de registro de auditoría son remanentes Se guardan en una memoria remanente del módulo de seguridad, por lo que están disponibles tras una desconexión de la alimentación eléctrica. Nota para CPs: Con CPs, los datos de registro de auditoría no son remanentes. Por ello, para guardar los datos es necesario utilizar un servidor Syslog.
Registro del sistema	El registro del sistema registra de forma automática y continua eventos de sistema como p. ej. el inicio de un proceso o el intento fallido de un usuario para iniciar sesión. Para configurar el filtro de eventos y el diagnóstico de líneas, elija la entrada "Configuración del registro" > "Configurar eventos del sistema".	Los datos de registro de sistema no son remanentes Los datos se guardan en una memoria volátil del módulo de seguridad, por lo que dejan de estar disponibles tras una desconexión de la alimentación eléctrica. Para el guardado remanente es posible guardar también en un archivo los datos de registro visualizados en el cuadro de diálogo "Online y diagnóstico".

Tabla 8-21 Registro local - método de almacenamiento para el registro de los datos

Método de almacenamiento	Significado
Memoria circulante	Cuando se alcanza el final del búfer el registro continúa al principio del búfer sobrescribiendo las entradas más antiguas.
Memoria lineal	El registro se detiene cuando el búfer está lleno.

Configurar eventos de sistema

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Configuración del registro" > "Configurar eventos del sistema".

Filtrado de los eventos de sistema

Ajuste en este cuadro de diálogo un nivel de filtrado para los eventos del sistema. Los valores predeterminados son los siguientes:

- SCALANCE S: nivel 3 (error)
- CP: nivel 3 (error)

El nivel de filtrado seleccionado debe tener una prioridad igual o menor que la severidad ajustada para el diagnóstico de la línea (véase la tabla "Parametrización del diagnóstico de líneas"; no para CPs).

Recomendación: seleccione como nivel de filtrado "Error" o un valor superior para evitar el registro de eventos generales, no críticos.

Nota para CP

Seleccione para CPs únicamente los niveles 3 o 6, pues para los CPs solo se generan eventos de estos niveles.

- Si selecciona el nivel 3, se muestran los mensajes de error de los niveles 0 a 3.
- Si selecciona el nivel 6, se muestran los mensajes de error de los niveles 0 a 6.

Propiedades de los eventos de sistema – Diagnóstico de líneas (no para CPs)

El diagnóstico de líneas genera un evento especial del sistema. Se genera un evento de sistema al alcanzarse el porcentaje de telegramas erróneos que se haya establecido. A este evento de sistema se le asignarán la severidad y facilidad ajustadas en este cuadro de diálogo.

Tabla 8-22 Parametrización del diagnóstico de líneas

Función / opción / parámetro	Significado
Activar	Activación y desactivación del registro.
Limit	Porcentaje ajustable de telegramas erróneos que, al alcanzarse, provoca un evento de sistema.
Facilidad	Seleccione en la lista desplegable una facilidad con la que identificar el evento de sistema captado para su registro.
Severidad	A través de la severidad se ponderan los eventos de sistema del diagnóstico de líneas respecto a la severidad de los eventos de sistema restantes.

Nota

Severidad de los eventos de sistema del diagnóstico de líneas

Asegúrese de no asignar a los eventos de sistema del diagnóstico de líneas una severidad menor que la que haya ajustado para el filtro. En caso de una severidad menor, esos eventos no pasarán el filtro y no serán registrados.

Configurar el registro del sistema en red

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Seleccione en los ajustes de seguridad locales la entrada "Configuración del registro" > "Red Syslog".

Configurar el registro del sistema en red

Tabla 8-23 Registro del sistema en red - Ajustes básicos

Opción/parámetros	
Activar registro del sistema en red	Activar y desactivar la transferencia de eventos de registro al servidor Syslog.
Servidor Syslog	Introduzca aquí la dirección IP del servidor Syslog. Para módulos SCALANCE S V4 también existe la posibilidad alternativa de introducir un FQDN. El servidor Syslog debe ser accesible desde el módulo de seguridad a través de la dirección indicada, y si fuese necesario, a través de la configuración del router en la entrada "Enrutamiento" de los ajustes de seguridad locales. Si el servidor Syslog no está accesible, no se enviarán mensajes de registro del sistema. Este estado operativo puede reconocerse por medio de los correspondientes avisos del sistema. Para activar de nuevo el envío de información Syslog, deberá actualizar en caso necesario la información de routing y efectuar un reinicio del módulo de seguridad.
Nombre del módulo	Aquí se muestra el nombre del módulo, que no puede modificarse.

Tabla 8-24 Registro del sistema en red - Ajustes para eventos de registro

Evento de registro	Configuración	Observaciones
Eventos de filtrado de paquetes (cortafuegos)	El registro de filtros de paquetes registra determinados paquetes del tráfico de datos. Solamente se registran los paquetes de datos afectados por una regla de filtrado de paquetes configurada (cortafuegos), o para los que reacciona la protección básica (paquetes corruptos o no válidos). Para ello es imprescindible que esté activado el registro para la regla de filtrado de paquetes. Mediante el ajuste de Facility y Severity se pueden clasificar los avisos de Syslog en función de su procedencia y de su gravedad. La asignación se realiza mediante listas desplegables. A cada evento se asignan la Severity y Facility que se configuren aquí.	El valor que se seleccione aquí depende de la evaluación en el servidor Syslog. Si deja configurado el valor estándar "default", se determina mediante el módulo de seguridad con qué combinación de Facility y Severity se visualizará el evento.
Eventos de auditoría	El registro de auditoría registra de manera automática y continua eventos relevantes para la seguridad, por ejemplo acciones del usuario como activación o desactivación del registro de paquetes o carga de configuraciones en el módulo de seguridad. La asignación de Severity y Facility se realiza mediante listas desplegables. A cada evento se asignan la Severity y Facility que se configuren aquí.	El valor que se seleccione aquí para Security y Facility depende de la evaluación en el servidor Syslog. Si deja configurado el valor estándar "default", se determina mediante el módulo de seguridad con qué combinación de Facility y Severity se visualizará el evento.
Eventos de sistema	El registro del sistema registra de forma automática y continua eventos de sistema como p. ej. el inicio de un proceso o el intento fallido de un usuario para iniciar sesión.	Para configurar el filtro de eventos y el diagnóstico de líneas, en los ajustes de seguridad locales elija la entrada "Configuración del registro" > "Configurar eventos del sistema".

Módulo de seguridad como router

Sinopsis de los ajustes de enrutamiento

Significado

Utilizando el módulo de seguridad en modo de enrutamiento, las redes se convierten en subredes separadas en las interfaces interna y externa. La interfaz DMZ (solo SCALANCE S623/S627-2M) se conecta en el modo de enrutamiento independientemente del modo de operación. En el modo de enrutamiento se reenvían los telegramas dirigidos a una dirección IP existente en la respectiva subred. Por lo demás son válidas las reglas de cortafuegos adoptadas para el respectivo sentido de transmisión.

Además, existen las siguientes posibilidades:

- Ajustar rutas específicas - configurable en "Enrutamiento" de los ajustes de seguridad locales (solo para SCALANCE S), véase Definir rutas (Página 862) en el capítulo "SCALANCE S".
- Utilizar router estándar - configurable en "Interfaz externa [P1] roja", "Interfaz interna [P2] verde" o "Interfaz DMZ [P3] amarilla" (solo SCALANCE S623/S627-2M) de los ajustes de seguridad locales, véase Configurar parámetros de dirección IP (Página 844) en el capítulo "SCALANCE S".
  Solo es posible utilizar un router estándar por interfaz.
- Enrutamiento NAT/NAPT - configurable en "NAT/NAPT" (no para CP 1628 ni CP 1543-1) de los ajustes de seguridad locales. Para utilizar el enrutamiento NAT/NAPT, el módulo de seguridad debe estar en modo de enrutamiento.

Activar el modo de enrutamiento (solo es necesario para módulos SCALANCE S)

Para este modo de operación es necesario configurar en los ajustes de seguridad locales una dirección IP interna y una máscara de subred interna para el direccionamiento del router en la subred interna. Todas las solicitudes de red que no pertenecen a una subred se transfieren a otra subred a través del módulo de seguridad.

Nota: A diferencia del modo Bridge del módulo de seguridad, en el modo de enrutamiento se pierden los identificadores VLAN. Los módulos SCALANCE S no soportan VLAN. Por este motivo, dichos módulos rechazan los telegramas con identificador VLAN.

1. En los ajustes de seguridad locales elija la opción "Modo de enrutamiento" en "Modo de operación".
2. En "Interfaz interna [P2] verde" > "Direcciones Ethernet" de los ajustes de seguridad locales, introduzca en los campos de entrada una dirección IP interna y una máscara de subred interna para el direccionamiento del router en la subred interna.

Sinopsis de NAT/NAPT

Función específica del módulo

Esta función no está disponible para CP 1628 ni para CP 1543-1.

Requisitos

El módulo de seguridad se encuentra en el modo de enrutamiento o la interfaz DMZ (solo SCALANCE S623 / S627-2M) está activada.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "NAT/NAPT".
3. Según se requiera, active una conversión de direcciones según NAT (Network Adress Translation) o NAPT (Network Address Port Translation).

Conversión de direcciones con NAT (Network Address Translation)

NAT es un método para la conversión de direcciones entre dos áreas de direcciones. La principal tarea es la conversión de direcciones IP privadas en públicas, es decir, en direcciones IP utilizadas y enrutadas en Internet. De ese modo se consigue que las direcciones IP de la red interna no se conozcan en la red externa. Los dispositivos internos solo se pueden ver en la red externa a través de las direcciones IP externas definidas en la lista de conversión de direcciones (tabla NAT). El hecho de que la dirección IP externa no sea la dirección del módulo de seguridad y que la dirección IP interna sea unívoca se denomina 1:1 NAT. Con 1:1 NAT la dirección interna se convierte a esta dirección externa sin conversión de puerto. En cualquier otro caso será n:1 NAT.

Conversión de direcciones con NAPT (Network Address Port Translation)

La conversión de direcciones en NAPT modifica la dirección de destino y el puerto de destino en una relación de comunicación (reenvío de puerto).

Se convierten telegramas procedentes de la red externa o la red DMZ y destinados a la dirección IP del módulo de seguridad. Si el puerto de destino del telegrama es idéntico a uno de los valores de la columna "Puerto de origen", el módulo de seguridad sustituye la dirección IP de destino y el puerto de destino de la forma indicada en la fila correspondiente de la tabla NAPT. En la respuesta, el módulo de seguridad aplica como dirección IP de origen y puerto de origen los valores que figuran como dirección IP de destino o puerto de destino en el telegrama inicial.

La diferencia respecto a NAT consiste en que en este protocolo también se pueden convertir puertos. Ya no hay una conversión 1:1 de la dirección IP. Solo existe una dirección IP pública, que se convierte agregando números de puerto a una serie de direcciones IP privadas.

Prueba de coherencia - reglas a considerar

Observe las reglas siguientes para la asignación de direcciones, con el fin de obtener entradas coherentes:

Prueba / Regla	Comprobación realizada	
	a nivel local	a nivel de proyecto
Las direcciones IP internas no deben ser idénticas a las direcciones IP del módulo.		x
Adopte para las direcciones IP la parte determinada por la máscara de subred: • Las direcciones IP que introduzca para el sentido "Externa" deben estar en la misma área de subred que la dirección IP externa del módulo de seguridad en la entrada "Interfaz externa [P1] roja". • Las direcciones IP que introduzca para el sentido "Interna" deben estar en la misma área de subred que la dirección IP interna del módulo de seguridad en la entrada "Interfaz interna [P2] verde". • Las direcciones IP que introduzca para el sentido "DMZ" deben estar en la misma área de subred que la dirección IP DMZ del módulo de seguridad en la entrada "Interfaz DMZ [P3] amarilla".		x
Una direcicón IP utilizada en la lista de conversión de direcciones NAT/NAPT no debe ser dirección Multicast ni dirección Broadcast.		x

Prueba / Regla	Comprobación realizada	
	a nivel local	a nivel de proyecto
Los puertos externos asignados a la conversión NAPT han de estar en el rango > 0 y ≤ 65535. Quedan excluidos los puertos 123 (NTP), 443 (HTTPS), 514 (Syslog), 161 (SNMP), 67+68 (DHCP) y 500+4500 (IPsec) y 161 (SNMP), siempre que estén activados los servicios correspondientes en el módulo de seguridad.	x	
La dirección IP externa del módulo de seguridad o la dirección IP de la interfaz DMZ solo se deben utilizar en la tabla NAT para la acción "Source-NAT".	x	
Control de duplicidad en la tabla NAT Una dirección IP externa o una dirección IP de la red DMZ utilizada en sentido "Destination-NAT", "Source-NAT + Destination-NAT" o "Double-NAT" solo puede utilizarse una vez en cada sentido indicado.	x	
Control de duplicidad en la tabla NAPT: cada número de puerto de origen solo puede introducirse una vez en cada interfaz.	x	
Los puertos NAPT internos pueden estar en el rango > 0 y ≤ 65535.	x	

Consulte también

Sinopsis de los ajustes de enrutamiento (Página 797)

Enrutamiento NAT/NAPT

Activar NAT

Se activa el campo de entrada para NAT. Las conversiones de direcciones NAT solo pasan a ser efectivas tras realizar las entradas descritas a continuación en la lista de conversión de direcciones. Tras crear las reglas NAT, se generan las reglas de cortafuegos correspondientes y se muestran en el modo de cortafuegos avanzado; véase el capítulo:

Relación entre router NAT/NAPT y cortafuegos (Página 805)

Si está activado PPPoE para la interfaz externa o la interfaz DMZ, la acción "Destination-NAT" no puede configurarse. Al configurar la acción "Source-NAT", la dirección IP no puede introducirse en el campo de entrada "Conversión de origen" porque se determina de forma dinámica en el tiempo de ejecución.

Acciones de conversión de direcciones posibles para NAT

La siguiente tabla recoge las posibilidades de entrada para la conversión de direcciones con NAT.

Acción "Destination-NAT" - "Redirect"

La acción "Destination-NAT" puede ejecutarse en el siguiente sentido:

- De externa a interna

Si la interfaz DMZ del módulo de seguridad (solo SCALANCE S623/S627-2M) está activada, la acción "Destination-NAT" también puede ejecutarse en los siguientes sentidos:

- De externa a DMZ
- De DMZ a interna
- De DMZ a externa

Para el sentido "De externa a interna" se aplica, por ejemplo, lo siguiente: se comprueba si la dirección IP de destino de un telegrama procedente de la red externa coincide con la dirección IP indicada en el campo de entrada "Dirección IP de destino". Si coincide, el telegrama se transfiere a la red interna sustituyendo la dirección IP de destino del telegrama por la dirección IP indicada en el campo de entrada "Conversión de destino". El acceso de externa a interna a través de la dirección IP externa es posible.

La siguiente tabla muestra el esquema de entrada para la acción "Destination-NAT".

Campo	Entradas posibles	Significado
Dirección IP de origen	No relevante para esta acción.	-
Conversión de origen	No relevante para esta acción.	-
Dirección IP de destino	Dirección IP en la red de origen	Dirección IP de destino en la red de origen a través de la cual se debe acceder a una dirección IP de la red de destino. Si en un telegrama la dirección IP de destino concuerda con la dirección introducida, la dirección se cambia por la dirección IP correspondiente en la red de destino. Si la dirección IP introducida aquí no es la dirección IP del módulo de seguridad, se convierte en dirección alias. Eso significa que la dirección IP indicada se registra además como dirección IP en la interfaz seleccionada. Asegúrese de que no existe ningún conflicto de dirección IP en la red con esta dirección alias.
Conversión de destino	Dirección IP en la red de destino	La dirección IP de destino se sustituye por la dirección IP indicada aquí.
N.º	-	Número consecutivo adjudicado por STEP 7 que se utiliza para la referencia a la regla de cortafuegos que STEP 7 genera para la regla NAT.

Acción "Source-NAT" - "Masquerading"

La acción "Source-NAT" puede ejecutarse en el siguiente sentido:

- De interna a externa

Si la interfaz DMZ del módulo de seguridad (solo SCALANCE S623/S627-2M) está activada, la acción "Source-NAT" también puede ejecutarse en los siguientes sentidos:

- De interna a DMZ
- De externa a DMZ
- De DMZ a externa

Para el sentido "De interna a externa" se aplica, por ejemplo, lo siguiente: se comprueba si la dirección IP de origen de un telegrama procedente de la red interna coincide con la dirección IP indicada en el campo de entrada "Dirección IP de origen". Si coincide, el telegrama se transfiere a la red externa con la dirección IP externa indicada en el campo de entrada "Conversión de origen" como nueva dirección IP de origen. En la red externa rige la dirección IP externa.

La siguiente tabla muestra el esquema de entrada para la acción "Source-NAT".

Campo	Entradas posibles	Significado
Dirección IP de origen	Dirección IP en la red de origen	La dirección IP de origen del dispositivo indicado se sustituye por la dirección IP indicada en el campo de entrada "Conversión de origen".
	Área de direcciones IP en la red de origen	El área de direcciones IP se sustituye por la dirección IP indicada en el campo de entrada "Conversión de origen".
	*	Las direcciones IP de todos los dispositivos de la red de origen se sustituyen por la dirección IP indicada en el campo de entrada "Conversión de origen".
Conversión de origen	Dirección IP en la red de destino	Entrada de la dirección IP que debe utilizarse como nueva dirección IP de origen. Si la dirección IP introducida aquí no es la dirección IP del módulo de seguridad, se convierte en dirección alias. Eso significa que la dirección indicada se registra además como dirección IP en la interfaz seleccionada. Asegúrese de que no existe ningún conflicto de dirección IP en la red con esta dirección alias.
Dirección IP de destino	No relevante para esta acción.	No relevante para esta acción.
Conversión de destino	No relevante para esta acción.	No relevante para esta acción.
N.º	-	Número consecutivo adjudicado por STEP 7 que se utiliza para la referencia a la regla de cortafuegos que STEP 7 genera para la regla NAT.

Puede configurar que las direcciones de todos los telegramas que van de una red de origen a una red de destino se conviertan a la dirección IP de los módulos de la red de destino. Además, el módulo de seguridad asigna un número de puerto a cada telegrama. En este caso, se trata de una conversión de direcciones n:1 NAT en la que varias direcciones IP de la red de origen se convierten a una dirección IP de la red de destino.

Introduzca, por ejemplo, los siguientes parámetros para el sentido "De interna a externa":

- Acción: "Source-NAT"
- De: "Interna"

- A "Externa"
- Dirección IP de origen: "*"
- Conversión de origen: dirección IP externa del módulo de seguridad

Acción "Source-NAT + Destination-NAT" - "1:1-NAT"

La acción "Source-NAT + Destination-NAT" puede ejecutarse en el siguiente sentido:

- De interna a externa

Si la interfaz DMZ del módulo de seguridad (solo SCALANCE S623/S627-2M) está activada, la acción "Source-NAT + Destination-NAT" también puede ejecutarse en los siguientes sentidos:

- De interna a DMZ
- De externa a DMZ
- De DMZ a externa

Para el sentido "De interna a externa" se aplica, por ejemplo, lo siguiente: en el acceso de interna a externa, se ejecuta la acción "Source-NAT". En el acceso de externa a interna, se ejecuta la acción "Destination-NAT".

La siguiente tabla muestra el esquema de entrada para la acción "Source-NAT+ Destination-NAT":

Campo	Entradas posibles	Significado
Dirección IP de origen	Dirección IP en la red de origen	La configuración siempre se indica en sentido Source-NAT. Por ello, las direcciones IP del sentido Destination-NAT siempre son insertadas automáticamente por STEP 7.
Conversión de origen	Dirección IP en la red de destino	
Dirección IP de destino	No relevante para esta acción.	
Conversión de destino	No relevante para esta acción.	
N.º	-	Número consecutivo adjudicado por STEP 7 que se utiliza para la referencia a las reglas de cortafuegos que STEP 7 genera para la regla NAT.

Acción "Double-NAT"

La acción "Double-NAT" puede ejecutarse en los siguientes sentidos:

- De interna a externa
- De externa a interna

Si la interfaz DMZ del módulo de seguridad (solo SCALANCE S623/S627-2M) está activada, la acción "Double-NAT" también puede ejecutarse en los siguientes sentidos:

- De interna a DMZ
- De externa a DMZ
- De DMZ a interna
- De DMZ a externa

Source-NAT y Destination-NAT se producen siempre en cada sentido.
Para el sentido "De externa a interna" se aplica, por ejemplo, lo siguiente: en el acceso de externa a interna, se sustituye la dirección IP de origen del dispositivo externo (Source-NAT). Además, el acceso a la red interna se realiza a través de la dirección IP externa (Destination-NAT) indicada en el campo de entrada "Dirección IP de destino".

Puede utilizar esta acción, por ejemplo, si se ha introducido un router estándar distinto al módulo de seguridad para un dispositivo al que se accede con ayuda de Destination-NAT. Los telegramas de respuesta de este dispositivo no se enviarán al router estándar introducido, sino a la interfaz correspondiente del módulo de seguridad.

La siguiente tabla muestra el esquema de entrada para la acción "Double-NAT":

Campo	Entradas posibles	Significado
Dirección IP de origen	Dirección IP en la red de origen	Dirección IP del dispositivo en la red de origen
Conversión de origen	-	La conversión de direcciones Source-NAT se realiza siempre a la dirección IP del módulo de seguridad en la red de destino. Por este motivo, el campo de entrada "Conversión de origen" no puede configurarse.
Dirección IP de destino	Dirección IP en la red de origen	Dirección IP de destino en la red de origen a través de la cual se debe acceder a una dirección IP de la red de destino. Si en un telegrama la dirección IP de destino concuerda con la dirección IP introducida, la dirección IP se sustituye por la dirección IP introducida en el campo de entrada "Conversión de destino". Si la dirección IP introducida aquí no es la dirección IP del módulo de seguridad, se convierte en dirección alias. Eso significa que la dirección indicada se registra además como dirección IP en la interfaz seleccionada. Asegúrese de que no existe ningún conflicto de dirección IP en la red con esta dirección alias.
Conversión de destino	Dirección IP en la red de destino	La dirección IP de destino se sustituye por la dirección IP indicada aquí.
N.º	-	Número consecutivo adjudicado por STEP 7 que se utiliza para la referencia a la regla de cortafuegos que STEP 7 genera para la regla NAT.

Activar NAPT

Se activa el campo de entrada para NAPT. Las conversiones NAPT solo pasan a ser efectivas tras realizar las entradas descritas a continuación en la lista. Tras crear las reglas NAPT, se generan las reglas de cortafuegos correspondientes y se muestran en el modo de cortafuegos

avanzado; véase el capítulo:
Relación entre router NAT/NAPT y cortafuegos (Página 805)

La conversión de direcciones IP con NAPT puede ejecutarse en el siguiente sentido:

- De externa a interna

Si la interfaz DMZ del módulo de seguridad (solo SCALANCE S623/S627-2M) está activada, la conversión de direcciones IP con NAPT también puede ejecutarse en los siguientes sentidos:

- De externa a DMZ
- De DMZ a interna
- De DMZ a externa

Para el sentido "De externa a interna" se aplica, por ejemplo, lo siguiente: los telegramas destinados a la dirección IP externa del módulo de seguridad y al puerto introducido en la columna "Puerto de origen" se transfieren a la dirección IP de destino de la red interna y al puerto de destino indicados.

La siguiente tabla muestra el esquema de entrada para la conversión de direcciones con NAPT:

Campo	Entradas posibles	Significado
Puerto de origen	Puerto o rango de puertos TCP/UDP Ejemplo de entrada de un rango de puertos: 78:99	Un dispositivo de la red de origen puede enviar un telegrama a un interlocutor de la red de destino utilizando este número de puerto.
Dirección IP de destino	Dirección IP en la red de destino	Los telegramas destinados a la dirección IP del módulo de seguridad de la red de origen, así como al puerto TCP/UDP indicado en el campo "Puerto de origen", se transfieren a la dirección IP aquí indicada.
Puerto de destino	Puerto TCP/UDP	Número de puerto al que se transfieren los telegramas procedentes de la red de origen.
Protocolo	• TCP+UDP • TCP • UDP	Selección de la familia de protocolos para los números de puerto indicados
N.º	-	Número consecutivo adjudicado por STEP 7 que se utiliza para la referencia a la regla de cortafuegos que STEP 7 genera para la regla NAPT.

Relación entre router NAT/NAPT y cortafuegos

Significado

Tras crear reglas NAT/NAPT, STEP 7 genera automáticamente reglas de cortafuegos que habilitan la comunicación en el sentido de conversión de direcciones configurado. Las reglas de cortafuegos generadas son visibles en el modo de cortafuegos avanzado y pueden ampliarse si es necesario (dirección IP adicional, servicios, ancho de banda). Asimismo, debería comprobarse la prioridad de las reglas de cortafuegos generadas respecto de su posición. Seguidamente, las reglas de cortafuegos generadas pueden desplazarse si fuera necesario. Los parámetros de cortafuegos generados por STEP 7 no pueden adaptarse.

Después de desactivar NAT/NAPT se desactivan las reglas de cortafuegos generadas por STEP 7.

Para simplificar la referencia entre las reglas NAT/NAPT y las correspondientes reglas de cortafuegos, estas están marcadas con los correspondientes números consecutivos.

La siguiente tabla muestra los esquemas de las reglas de cortafuegos generadas para las reglas NAT.

Tabla 8-25 Conversión de direcciones NAT y reglas de cortafuegos correspondientes

Acción NAT	Regla de cortafuegos creada				
	Acción	De	A	Dirección IP de origen	Dirección IP de destino
Source-NAT	Allow	Red de origen	Red de destino	Dirección IP del dispositivo indicado en el campo de entrada "Dirección IP de origen".	-
Destination-NAT	Allow	Red de origen	Red de destino	-	Dirección IP del dispositivo indicado en el campo de entrada "Conversión de destino".
Source-NAT + Destination-NAT	Allow	Red de origen	Red de destino	Dirección IP del dispositivo indicado en el campo de entrada "Dirección IP de origen".	-
	Allow	Red de destino	Red de origen	-	Dirección IP del dispositivo insertado por STEP 7 en el campo de entrada "Conversión de destino".
Double-NAT	Allow	Red de origen	Red de destino	Dirección IP del dispositivo indicado en el campo de entrada "Dirección IP de origen".	Dirección IP del dispositivo indicado en el campo de entrada "Conversión de destino".

La siguiente tabla muestra el esquema de las reglas de cortafuegos generadas para las reglas NAPT.

Tabla 8-26 Conversiones NAPT y reglas de cortafuegos creadas

Regla de cortafuegos creada					
Acción	De	A	Dirección IP de origen	Dirección IP de destino	Servicio
Allow	Red de origen	Red de destino	-	Dirección IP del dispositivo indicado en el campo de entrada "Dirección IP de destino".	[NúmPuerto_protocolo]

Configurar la sincronización horaria

Resumen de la sincronización horaria

Significado

Para comprobar la validez horaria de un certificado y para el sello de tiempo de entradas de registro se indican la fecha y la hora en el módulo de seguridad. Para CPs, la sincronización horaria está activa por defecto.

Se pueden configurar las siguientes alternativas:

- Ajustar la hora al cargar (solo para SCALANCE S): la hora del módulo se ajusta automáticamente a la hora del PC al cargar una configuración.
- SIMATIC (solo para CPs): si el módulo de seguridad recibe mensajes de hora MMS, su hora local se sincronizará, siempre que no se haya configurado el procedimiento NTP (MMS = Manufacturing Message Specification).
- NTP: ajuste automático y sincronización periódica de la hora a través de un servidor Network Time Protocol.

Nota

La sincronización horaria se refiere únicamente al módulo de seguridad y no puede utilizarse para la sincronización de dispositivos de la red interna de dicho módulo. Los CPs S7 pueden reenviar a hora a otros módulos de la estación.

Nota

Configurar el cortafuegos para la comunicación con el servidor NTP

Si no es posible acceder al servidor NTP desde el módulo de seguridad, los telegramas del servidor NTP se tienen que permitir explícitamente en el cortafuegos (UDP, puerto 123).

Definir un servidor NTP

Crear un servidor NTP en los ajustes de seguridad globales

En los ajustes de seguridad globales existe la posibilidad de crear únicamente servidores NTP del tipo "NTP (secure)" y asignarlos a CPs o módulos SCALANCE S V4. Por consiguiente, los servidores NTP no seguros para módulos SCALANCE S < V4 y CPs deben crearse en los ajustes de seguridad locales.

1. Haga doble clic en la entrada "NTP" de los ajustes de seguridad globales.
2. Haga doble clic en la entrada "Agregar servidor NTP nuevo".
3. Introduzca un nombre para el servidor NTP (secure).
4. Introduzca la dirección IP del servidor NTP (secure). Si únicamente deben asignarse al servidor NTP módulos SCALANCE S V4, es posible indicar alternativamente un FQDN.
5. Indique los parámetros de cifrado para el servidor NTP (secure).

Propiedad	Significado
ID de código	Valor numérico entre 1 ... 65534.
Autenticación	Seleccione el algoritmo de autenticación.
Hex/ASCII	Seleccione el formato del código NTP.
Código	Introduzca el código NTP con las siguientes longitudes: Hex: 22 ... 40 caracteres ASCII: 11 ... 20 caracteres

6. Asigne los módulos de seguridad deseados al servidor NTP creado (secure), consulte el capítulo:
   Asignar módulo de seguridad a un servidor NTP (secure) (Página 811).

Crear un servidor NTP en los ajustes de seguridad locales

1. Seleccione el módulo que desea editar.
2. Elija la entrada "Sincronización horaria" en los ajustes de seguridad locales.
3. Seleccione el modo de sincronización deseado.
4. Introduzca un nombre y la dirección IP del servidor NTP. Si se ha seleccionado el modo de sincronización "NTP (secure)", puede elegir en la columna "Nombre" un servidor NTP (secure) que haya creado en los ajustes de seguridad globales.

Capacidad para servidores NTP

Para la creación de servidores NTP rigen las siguientes reglas:

- Pueden crearse hasta 32 servidores NTP para todo el proyecto.
- A un módulo de seguridad se le pueden asignar como máximo 4 servidores NTP.

Importar y exportar servidores NTP (secure)

Los comandos "Importar" y "Exportar" del menú contextual permiten exportar la lista de códigos del servidor NTP (secure) seleccionado en ese momento en los ajustes de seguridad globales e importar el archivo a un servidor NTP (secure), o viceversa.

Configurar la sincronización horaria para un módulo de seguridad

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Elija la entrada "Sincronización horaria" en los ajustes de seguridad locales.
3. Active la casilla de verificación "Activar sincronización horaria".

Alternativas de la sincronización horaria

Se pueden configurar las siguientes alternativas:

Tabla 8-27 Sincronización horaria para CP y SCALANCE S

Posibilidades	Significado / repercusión
Ajustar la hora al cargar (solo para SCALANCE S)	La hora del módulo se ajusta automáticamente a la hora del PC al cargar una configuración.
SIMATIC (no para SCALANCE S)	Si el módulo de seguridad recibe mensajes de hora MMS, su hora local se sincronizará, siempre que no se haya configurado el procedimiento NTP (MMS = Manufacturing Message Specification).
NTP (no para SCALANCE S ≥ V3)	Ajuste automático y sincronización periódica de la hora a través de un servidor NTP.
NTP (secure) (no para SCALANCE S < V4)	Ajuste automático y sincronización periódica de la hora a través de un servidor NTP (secure).

Seleccionar el modo de sincronización horaria

Proceda del siguiente modo:

1. Seleccione el modo de sincronización horaria.
2. En función del modo seleccionado están disponibles las posibilidades de ajuste siguientes:

- **Ajustar la hora al cargar:** al cargar una configuración en un módulo SCALANCE S se sincroniza la hora del módulo con la del PC.
- **SIMATIC:** para CP x43-1, elija si el CP solo recibe la hora o si también la reenvía. En el CP 1628 la hora siempre se reenvía. Ajuste además para CPs el sentido para reenviar la hora.
  Sentidos disponibles:
  - Automático (solo para CP x43-1): el CP recibe la hora de la estación o LAN y la transfiere a la estación o LAN, respectivamente. Si se utilizan varios CPs en la estación, es posible que el ajuste automático provoque colisiones. Para impedirlas, defina el sentido del reenvío.
  - De estación (solo para CP x43-1).
  - De LAN.

  Si el reenvío de la hora está activado, la casilla de verificación "Utilizar hora corregida" permite especificar si debe utilizarse un factor de corrección que pueda estar incluido en el telegrama horario. Para el CP 1628 esta opción está activada por defecto y no puede desactivarse.
- **NTP:**
  - Zona horaria: en el procedimiento NTP se transfiere generalmente la hora UTC (Universal Time Coordinated), que equivale a la hora GMT (Greenwich Mean Time). Mediante la configuración de la zona horaria local se puede ajustar la diferencia de tiempo respecto de la hora UTC.
  - Intervalo de actualización en segundos: Define el espacio de tiempo entre las consultas de hora en segundos. Para SCALANCE S ≥ V3, el intervalo de tiempo para la consulta del servidor NTP se define automáticamente.

    Nota

    Ajuste del intervalo de actualización para CPs

    Si se activa la casilla de verificación "Activar funciones de seguridad" en los ajustes de seguridad locales de CPs, el ajuste del intervalo de actualización se toma de los ajustes locales del CP y se aplica a los ajustes de seguridad locales del CP.

  - Sincronización horaria al minuto completo (solo para CP x43-1/CP 1628): esta opción permite decidir si la hora se debe reenviar al bus K exactamente al cumplirse el minuto. Esta opción se requiere solo para algunas aplicaciones especiales.
  - Aceptar la hora de servidores NTP no sincronizados (solo para CP x43-1/CP 1628/CP 1543-1): aquí se ajusta si el módulo de seguridad también debe aceptar la hora de servidores NTP no sincronizados.
  - Reenviar hora a estación (solo para CP x43-1/CP 1628): desactive esta opción si la CPU pregunta la hora independientemente a un servidor NTP. De este modo se evita que la hora consultada directamente en el servidor NTP por la CPU se sobrescriba con la hora del CP. La transferencia a través del CP podría provocar una precisión menor.
  - Servidor NTP: La creación de servidores NTP en los ajustes de seguridad locales se describe en el capítulo Definir un servidor NTP (Página 808).

Asignar módulo de seguridad a un servidor NTP (secure)

Requisitos

- Se ha definido en los ajustes de seguridad globales un servidor NTP (secure).
- En los ajustes de seguridad locales del módulo de seguridad que quiera asignar a un servidor NTP (secure) se ha seleccionado como modo de sincronización "NTP" o "NTP (secure)".

Procedimiento

1. Haga doble clic en la entrada "NTP" de los ajustes de seguridad globales.
2. Haga doble clic sobre la entrada "Asignar módulo a un servidor NTP".
3. Seleccione de la lista desplegable "Servidor NTP" el servidor NTP (secure) al que quiera asignar un módulo de seguridad.
4. Seleccione en el área "Módulos disponibles" el módulo de seguridad que quiera asignar al servidor NTP (secure) seleccionado.
5. Haga clic sobre el botón << para asignar el módulo de seguridad seleccionado al servidor NTP (secure) seleccionado.

Resultado

Ha asignado el módulo de seguridad al servidor NTP (secure). El servidor NTP (secure) aparece en los ajustes de seguridad locales automáticamente en la lista de los servidores NTP.

Módulo de seguridad como servidor DHCP

Función específica del módulo

Solo es posible utilizar el módulo de seguridad como servidor DHCP con módulos SCALANCE S; consulte el apartado:
Auto-Hotspot en el capítulo "SCALANCE S".

Configurar SNMP

Resumen SNMP

¿Qué es SNMP?

El módulo de seguridad soporta la transmisión de información de administración a través del Simple Network Management Protocol (SNMP). Para ello se ha instalado en el módulo de seguridad un agente SNMP, que recibe y responde las solicitudes SNMP. La información sobre las propiedades de dispositivos compatibles con SNMP está almacenada en los

denominados archivos MIB (MIB = Management Information Base), para los que el usuario debe tener los derechos correspondientes.

En SNMPv1 también se envía el "community string". El community string es como una contraseña que se envía junto con la solicitud SNMP. Si el community string es correcto, el módulo de seguridad responde con la información solicitada. Si el community string es incorrecto, el módulo de seguridad rechaza la solicitud y no contesta. En SNMPv1 el community string se transfiere sin cifrar.

En SNMPv3 los datos pueden enviarse cifrados.

Configurar SNMP - Entrada "SNMP"

Función específica del módulo

La configuración de SNMP solo es posible para SCALANCE S V3 o superior, CPs S7 y CP 1628.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "SNMP".
3. Active la casilla de verificación "Activar SNMP".
4. Elija una de las siguientes versiones de protocolo SNMP:

   Nota
   Transferencia de datos cifrada en SNMPv3
   Para aumentar la seguridad, utilice SNMPv3, puesto que los datos se transfieren cifrados.

   – SNMPv1
     Para controlar los derechos de acceso en el agente SNMP el módulo de seguridad utiliza los siguientes valores estándar para los community strings. Estos valores estándar deberían adaptarse para aumentar la seguridad.
     Para acceso de lectura: public
     Para acceso de lectura y escritura: private
     Para activar el acceso de escritura a través de SNMP, elija la opción "Permitir acceso de escritura".
   – SNMPv3
     Seleccione solo un algoritmo de autenticación o bien un algoritmo de autenticación y uno de cifrado.
     Algoritmo de autenticación: Ninguno, MD5, SHA-1
     Algoritmo de cifrado: Ninguno, AES-128, DES

     Nota
     Evitar la utilización de DES
     DES es un algoritmo de cifrado no seguro. Solo ha de utilizarse por motivos de compatibilidad con versiones anteriores.

5. Si va a utilizarse SNMPv3, asigne a un usuario un rol en el que estén activados los correspondientes derechos SNMP para que pueda acceder al módulo vía SNMP. Encontrará una vista general de los derechos SNMP en el capítulo: Administrar derechos (Página 767)
6. En el área "Configuración avanzada", configure datos específicos del módulo relacionados con el autor, la ubicación y la dirección de correo electrónico, que sobrescriban los datos de las propiedades del proyecto.
   Para valores escritos por una herramienta SNMP en el módulo de seguridad por medio de un comando SNMP-SET rige lo siguiente:
   si se activa la casilla de verificación "Conservar los valores escritos por SNMP-SET", los valores no se sobrescribirán al volver a cargar una configuración de SCT en el módulo de seguridad.

Configurar Proxy ARP

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior, consulte el apartado: Configurar Proxy ARP (Página 866)

Activar el servidor web en módulo de seguridad

Función específica del módulo

Esta función solo está disponible para el CP x43-1 Advanced; consulte el apartado: Activar el servidor web en el CP x43-1 Advanced (Página 887) en el capítulo "Seguridad para CPs S7-300/S7-400/PC".

Túnel IPsec: Crear y asignar grupos VPN

Cómo crear túneles IPsec mediante grupos VPN

Función específica del módulo

Esta función no está disponible para SCALANCE S602 ni para CP 1543-1 V1.0. El CP 1543-1 soporta esta función a partir del firmware V1.1.

Requisitos

Nota

Fecha y hora actuales en los módulos de seguridad

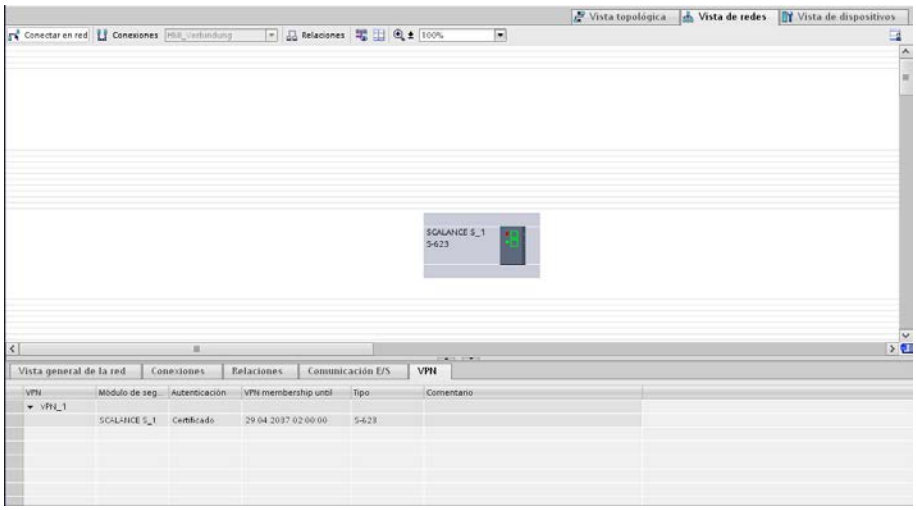
Cuando utilice comunicación segura (p. ej. HTTPS, VPN...), asegúrese de que los módulos de seguridad afectados tienen la hora y fecha actuales. De lo contrario, los certificados utilizados se considerarán no válidos y la comunicación segura no funcionará.

Cómo se accede a esta función

1. Haga doble clic en la entrada "Grupos VPN" > "Agregar grupo VPN nuevo" de los ajustes de seguridad globales para crear un grupo VPN.
2. Haga doble clic en la entrada "Grupos VPN" > "Asignar módulo a un grupo VPN" de los ajustes de seguridad globales y asigne al grupo VPN los módulos de seguridad y los módulos SOFTNET Security Client entre los que deben establecerse túneles VPN. Tenga en cuenta al respecto las reglas para formar grupos VPN. Las encontrará en el apartado: Modos de operación de grupos VPN (Página 819).

Visualización de los grupos VPN con sus propiedades

Si se selecciona un módulo de seguridad que está en uno o varios grupos VPN, en el área "Datos de red" se muestran las propiedades del o los grupos VPN en los que se encuentra el módulo de seguridad.



Las siguientes propiedades de los grupos VPN se visualizan por columnas en la ficha "VPN" del área "Datos de red":

Propiedad/columna	Significado
VPN	Nombres de los grupos VPN en los que se encuentra el módulo de seguridad seleccionado
Módulo de seguridad	Nombres de los módulos de seguridad asignados

Propiedad/columna	Significado
Autenticación	Tipo de autenticación: Preshared Key o certificado
Miembro del grupo hasta	Fecha y hora hasta las que es válido el certificado de grupo VPN del módulo de seguridad
Tipo	Números de modelos de los módulos de seguridad asignados
Comentario	Comentario

Ajustar la duración de certificados

Abra de la forma siguiente el cuadro de diálogo en el que puede introducir la fecha de caducidad del certificado:

1. En la ficha "VPN", seleccione el grupo VPN que desea editar.
2. En la ficha "Propiedades" > "General" de la ventana de inspección seleccione la entrada "Autenticación".

Nota

Caducidad de un certificado

La comunicación a través del túnel VPN continúa una vez caducado el certificado hasta que el túnel se deshaga o finalice la vida útil SA. Encontrará más información sobre certificados en el capítulo:
Auto-Hotspot.

Capacidad

Número de túneles IPSec	
SCALANCE S612 V2	64 como máximo
SCALANCE S612 V3/V4	128 como máximo
SCALANCE S613	128 como máximo
SCALANCE S623 V3/V4	128 como máximo
SCALANCE S627-2M V4	128 como máximo
CP x43-1 Advanced	32 como máximo
CP 1628	64 como máximo
CP 1543-1 V1.1	16 como máximo

Métodos de autenticación

Métodos disponibles

El método de autenticación se fija por grupo VPN y determina la forma de autenticación utilizada.

Son posibles métodos de autenticación basados en clave o en certificado:

- Preshared Key
  La autenticación se produce a través de una secuencia de caracteres establecida de antemano, que se distribuye a todos los módulos del grupo.
  Para ello, introduzca una Preshared Key en el campo "Clave" en "Autenticación" > "General" de las propiedades del grupo VPN.
- Certificado
  El ajuste estándar es el método de autenticación basado en certificado "Certificado". El comportamiento es el siguiente:
  – Al crear un grupo se genera automáticamente un certificado CA como certificado de origen.
  – Cada módulo de seguridad que forma parte del grupo recibe adicionalmente un certificado de grupo firmado con la clave del certificado CA.

  Todos los certificados se basan en el estándar ITU X.509v3 (ITU, International Telecommunications Union).
  Los certificados son generados por una entidad certificadora contenida en STEP 7.

Nota

Restricción para el modo VLAN

En la transmisión de telegramas IP a través del túnel VPN del módulo de seguridad no se transmiten identificadores de VLAN. Los identificadores de VLAN contenidos en los telegramas IP se pierden al pasar los módulos de seguridad, ya que para la transmisión de los telegramas IP se utiliza IPSec.

Con el ajuste predeterminado no es posible transmitir telegramas IP Broadcast o Mulitcast con IPsec a través de un túnel VPN de capa 3. A través de un túnel VPN de capa 2 del módulo de seguridad, los telegramas IP Broadcast o Multicast se "empaquetan" en UDP y se transmiten exactamente como paquetes MAC, incluido el encabezado Ethernet. Por ello, en esos paquetes también se conservan los identificadores de VLAN.

Propiedades de grupo para grupos VPN seleccionados

Propiedades del grupo VPN

Nota

Se requieren conocimientos sobre IPsec

Para poder ajustar estos parámetros necesita tener conocimientos en materia de IPsec. Si no efectúa ni modifica ningún ajuste, rigen los ajustes predeterminados.

En las propiedades de un grupo VPN pueden configurarse los ajustes siguientes:

- Método de autenticación (entrada: "General")
- Ajustes IKE (entrada: "Configuración avanzada fase 1")
- Ajustes IPsec (entrada: "Configuración avanzada fase 2")

Cómo se accede a esta función

1. En la entrada "Grupos VPN" de los ajustes de seguridad globales seleccione el grupo VPN cuyas propiedades desee configurar.
2. Elija el comando "Abrir" del menú contextual de esta entrada.
   Resultado: las propiedades del grupo VPN aparecen en el área de los ajustes de seguridad locales.
3. En la entrada "Autenticación" elija si para la autenticación debe utilizarse una Preshared Key o un certificado. Encontrará más información al respecto en el capítulo:
   Métodos de autenticación (Página 815).

Parámetros para ajustes ampliados Fase 1 - Ajustes IKE

Fase 1: Cambio de código (IKE = Internet Key Exchange):

Aquí se ajustan los parámetros para el protocolo de la administración de códigos IPsec. El cambio de clave se produce conforme al método estandarizado IKEv1, para el que se pueden ajustar los siguientes parámetro de protocolo:

Parámetro	Descripción
Modo IKE	Método de cambio de código: • Main Mode • Aggressive Mode La diferencia entre la modalidad Main y Aggressive es la "Identity-Protection" que se utiliza en el Main Mode. La identidad se transmite codificada en el Main Mode, en el Aggressive Mode no.
Fase 1 del grupo DH	Convenio de codificación Diffie-Hellman: • Group 1 • Group 2 • Group 5 • Group 14 Grupos Diffie-Hellman (algoritmos criptográficos seleccionables en el protocolo de cambio de códigos Oakley)
Tipo de vida útil SA	Phase 1 Security Association (SA): • Time: Limitación del tiempo en minutos Se limita el tiempo de vida útil para el material de codificación actual. Una vez transcurrido ese tiempo se negocia de nuevo el material de codificación.
Vida útil SA	Valor numérico: Rango de valores para Time: 1440 ... 2500000 minutos (predeterminado: 2500000)

Parámetro	Descripción
Fase 1 de la encriptación	Algoritmo de cifrado: • DES*: Data Encryption Standard (longitud de código 56 bits, modo CBC) • 3DES-168: DES triple (longitud de código 168 bits, modo CBC) • AES-128, 192, 256: Advanced Encryption Standard (longitud de código 128 bits, 192 bits o 256 bits, modo CBC)
Fase 1 de la autenticación	Algoritmo de autenticación: • MD5: Message Digest Algorithm 5 • SHA1: Secure Hash Algorithm 1

*DES es un algoritmo de cifrado no seguro. Solo ha de utilizarse por motivos de compatibilidad con versiones anteriores. El CP 1543-1 V1.1 no lo soporta.

Parámetros para Advanced Settings Phase 2 - Ajustes IPSec

Fase 2: Intercambio de datos (ESP = Encapsulating Security Payload)

Aquí se ajustan los parámetros para el protocolo del intercambio de datos IPsec. El intercambio de datos se produce en el "Quick Mode". Toda la comunicación que se produce en esta fase está cifrada a través del protocolo de seguridad estandarizado ESP, para el que pueden configurarse los siguientes parámetros:

Parámetro	Descripción
Tipo de vida útil SA	Phase 2 Security Association (SA): • Time: limitación del tiempo en minutos. Se limita el tiempo de vida útil para el material de codificación actual. Una vez transcurrido ese tiempo se negocia de nuevo el material de codificación. • Limit: Limitación del volumen de datos en MB
Vida útil SA	Valor numérico: • Rango de valores para Time: 60 ... 16666666 minutos (predeterminado: 2880) • Rango de valores para Limit: 2000 ... 500000 MB (predeterminado: 4000)
Fase 2 de la encriptación	Algoritmo de cifrado: • DES*: Data Encryption Standard (longitud de código 56 bits, modo CBC) • 3DES-168: DES triple (longitud de código 168 bits, modo CBC) • AES-128: Advanced Encryption Standard (longitud de código 128 bits, modo CBC)
Fase 2 de la autenticación	Algoritmo de autenticación: • MD5: Message Digest Algorithm 5 • SHA1: Secure Hash Algorithm 1
Perfect Forward Secrecy	Elija si antes de cada negociación de una SA IPsec tiene lugar una nueva negociación de la clave con ayuda del procedimiento Diffie-Hellman. Perfect Forward Secrecy garantiza que a partir de las claves generadas anteriormente no puedan deducirse las claves nuevas.

*DES es un algoritmo de cifrado no seguro. Solo ha de utilizarse por motivos de compatibilidad con versiones anteriores. El CP 1543-1 V1.1 no lo soporta.

Modos de operación de grupos VPN

Modos de operación VPN

Según el modo de operación en el que se encuentren los módulos de seguridad que se han agregado a un grupo VPN, se distingue entre diferentes modos de operación de grupos VPN. El modo de operación de un grupo VPN informa sobre qué módulos de seguridad pueden agregarse al grupo VPN y en qué modos de operación.

Reglas para la formación de grupos

Observe las reglas siguientes si desea formar grupos VPN:

- Para SCALANCE S612 / S613 / S623 / S627-2M / SCALANCE M
  El primer módulo asignado a un grupo VPN decide qué otros módulos se pueden agregar. Si el primer módulo SCALANCE S agregado está en modo de enrutamiento o si el primer módulo de seguridad es un módulo SCALANCE M, solo se podrán agregar módulos SCALANCE S con modo de enrutamiento activado o módulos SCALANCE M, pues los módulos SCALANCE M se utilizan siempre en modo de enrutamiento. Si el primer módulo SCALANCE S agregado está en modo Bridge, solo se podrán agregar módulos SCALANCE S en modo Bridge. Si se debe modificar el modo de un grupo VPN, hay que quitar todos los módulos contenidos en el grupo y volver a agregarlos. Un CP y un SSC puede agregarse a un grupo con un SCALANCE S en el modo Bridge o de enrutamiento.
- Para CP / SSC
  Si el primer módulo de un grupo VPN es un CP / SSC, es posible agregar módulos de seguridad en los modos que se desee. El siguiente módulo de seguridad que determina el modo de operación especifica también el modo de operación del grupo VPN. Un CP / SSC puede asignarse simultáneamente a varios grupos VPN con diferentes modos de operación. En tal caso el CP /SSC funcionará en modo combinado.
- No es posible agregar un módulo SCALANCE M a un grupo VPN que contenga un módulo en el modo Bridge.

Vea en la tabla siguiente qué módulos se pueden reunir en un grupo VPN:

Tabla 8-28 Módulos de seguridad y modos de operación VPN

Módulo	Se puede incluir en un grupo VPN en...	
	Modo Bridge	Modo de enrutamiento
SCALANCE S612 / S613 / S623 / S627-2M en modo Bridge	x	-*
SCALANCE S612 / S613 / S623 / S627-2M en modo de enrutamiento	-	x
CP x43-1 Adv.	x	x
CP 1543-1 V1.1	x	x
CP 1628	x	x

Módulo	Se puede incluir en un grupo VPN en...	
	Modo Bridge	Modo de enrutamiento
SOFTNET Security Client V4.0	x	x
SCALANCE M875	-	x

* Los módulos SCALANCE S623/S627-2M en modo Bridge pueden insertarse en un grupo VPN en modo de enrutamiento cuando su interfaz DMZ está activada (no simultáneamente).

Incluir un módulo de seguridad en un grupo VPN configurado

Las propiedades de grupo VPN configuradas se adoptan para los módulos de seguridad nuevos que se incluyen en un grupo VPN existente.

Procedimiento a seguir tras incluir un módulo de seguridad en un grupo VPN configurado

Dependiendo de si las propiedades de grupo VPN han cambiado o no desde el último proceso de carga, se tiene que proceder de forma distinta:

- **Caso a:** Si no ha cambiado las propiedades del grupo VPN y el módulo que se agrega establece activamente la conexión con los módulos previamente configurados:

1. Agregue el módulo de seguridad nuevo al grupo VPN.
2. Cargue la configuración en el módulo nuevo.

- **Caso b:**Si ha cambiado las propiedades del grupo VPN o si el módulo que se agrega no establece activamente la conexión con los módulos previamente configurados:

1. Agregue el módulo de seguridad nuevo al grupo VPN.
2. Cargue la configuración en todos los módulos pertenecientes al grupo VPN.

En el caso a no es necesario configurar de nuevo ni cargar los módulos de seguridad ya existentes y que ya se han puesto en servicio. La comunicación en curso no se ve afectada ni interrumpida.

Ajustes para estaciones con dirección IP desconocida

Las estaciones para las que no se conoce la dirección IP en el momento de la configuración (unknown peers) se pueden insertar en un grupo VPN ya existente. Dado que la mayoría de dispositivos suele tener un uso móvil y obtienen su dirección IP dinámicamente (p. ej. un SOFTNET Security Client o SCALANCE M), el túnel VPN solo puede establecerse si los ajustes de parámetros para la fase 1 se realizan conforme a una de las siguientes tablas ("Parámetro de cifrado 1" hasta "Parámetro de cifrado 4"). Si utiliza otros ajustes, no será posible establecer un túnel VPN con el dispositivo terminal.

Tabla 8-29 Parámetros de cifrado 1

Parámetro	Ajuste
Método de autenticación	Certificado
Fase 1 del grupo DH	Grupo 2
Vida útil SA	1440 … 2500000 minutos

Parámetro	Ajuste
Fase 1 de la encriptación	AES-256
Fase 1 de la autenticación	SHA-1

Tabla 8-30 Parámetros de cifrado 2

Parámetro	Ajuste
Método de autenticación	Certificado
Fase 1 del grupo DH	Grupo 2
Vida útil SA	1440 … 2500000 minutos
Fase 1 de la encriptación	3DES-168
Fase 1 de la autenticación	SHA-1

Tabla 8-31 Parámetros de cifrado 3

Parámetro	Ajuste
Método de autenticación	Certificado
Fase 1 del grupo DH	Grupo 2
Vida útil SA	1440 … 2500000 minutos
Fase 1 de la encriptación	DES
Fase 1 de la autenticación	MD5

Tabla 8-32 Parámetros de cifrado 4

Parámetro	Ajuste
Método de autenticación	Preshared Key
Fase 1 del grupo DH	Grupo 2
Vida útil SA	1440 … 2500000 minutos
Fase 1 de la encriptación	3DES-168
Fase 1 de la autenticación	SHA1

Limitaciones adicionales para el SOFTNET Security Client

Para el SOFTNET Security Client rigen además las siguientes limitaciones:

Tabla 8-33 Parámetros de cifrado para SOFTNET Security Client

Parámetro	Ajuste/particularidad
Fase 1 de la encriptación	AES-256 solo es posible con Windows 7
Fase 1 de la vida útil SA	1440 ... 2879 minutos
Tipo de vida útil SA	Debe ser idéntico para ambas fases.
Fase 2 de la encriptación	No es posible AES-128
Fase 2 de la vida útil SA	60 ... 2879 minutos
Fase 2 de la autenticación	No es posible MD5

Procedimiento a seguir tras quitar un dispositivo activo de un grupo VPN

Si retira una estación activa de un grupo VPN existente, la estación podrá establecer una conexión con las estaciones del grupo aunque haya vuelto a cargar el proyecto en todas ellas.

Si no desea que la estación activa retirada establezca conexión, renueve el certificado de grupo CA y vuelva a cargar el proyecto en las estaciones del grupo VPN. El certificado puede renovarse en las propiedades del grupo VPN o en el administrador de certificados, en la ficha "CA".

Configurar nodos de red internos

Resumen de la configuración de nodos de red internos

Configurar nodos de red internos

Para poder determinar la autenticidad de un telegrama, cada módulo de seguridad debe conocer los nodos de toda la red interna.

El módulo de seguridad debe conocer tanto su propio nodo interno como los nodos internos de los módulos de seguridad con los que comparte el grupo VPN. En un módulo de seguridad, esta información se utiliza para determinar qué paquete de datos se debe transmitir por qué túnel.

Al agregar un módulo de seguridad a un grupo VPN se notifican automáticamente al módulo de seguridad los nodos de red/las subredes locales e internos del módulo de seguridad. Para permitir la comunicación por el túnel VPN con otras subredes u otros dispositivos de otra subred, es necesario habilitar a través de la configuración dichas subredes o dichos dispositivos para la comunicación por túnel VPN.

Los módulos SCALANCE S permiten aprender los nodos de la red automáticamente o configurarlos estáticamente. Las posibilidades que están disponibles para el aprendizaje de nodos de red internos dependen también del modo de operación del módulo de seguridad.

SCALANCE S en modo Bridge

En el modo Bridge es posible configurar tanto los nodos IP/MAC internos como las subredes internas o, como alternativa, permitir el aprendizaje automático de nodos internos por parte del SCALANCE S.

SCALANCE S en modo de enrutamiento

En el modo de enrutamiento no hay ningún modo de aprendizaje automático disponible. En su lugar, aquí se introducen subredes completas que deben habilitarse para la comunicación tunelada.

CP x43-1 Advanced y CP 1628

- CP x43-1 Adv.
  Seleccione si la comunicación tunelada con el CP (interfaz Gigabit) y/o con la subred interna (subred PROFINET) está permitida para interlocutores VPN en modo de enrutamiento (SCALANCE S / M).
- CP 1628
  Introduzca los nodos NDIS que deben ser accesibles por el túnel de interlocutores VPN en modo de enrutamiento (SCALANCE S / M).

Aprendizaje automático de nodos de red internos

Función específica del módulo

Los módulos SCALANCE S en modo Bridge ofrecen un modo de aprendizaje que permite el aprendizaje automático de nodos de red internos durante el funcionamiento. Encontrará más información al respecto en el apartado:
Utilizar el modo de aprendizaje para aprender nodos internos (Página 867) en el capítulo "SCALANCE S".

Configurar manualmente nodos de red IP para SCALANCE S

Función específica del módulo

Para configurar nodos de red IP para módulos SCALANCE S, consulte el apartado:
Configurar manualmente nodos de red IP (Página 868) en el capítulo "SCALANCE S".

Configurar manualmente nodos de red MAC para SCALANCE S

Función específica del módulo

Para configurar nodos de red MAC para módulos SCALANCE S, consulte el apartado:
Configurar manualmente nodos de red MAC (Página 869) en el capítulo "SCALANCE S".

Configurar manualmente subredes internas para SCALANCE S

Función específica del módulo

Para configurar subredes internas para módulos SCALANCE S, consulte el apartado:
Configurar manualmente subredes internas (Página 869) en el capítulo "SCALANCE S".

Permitir el acceso a CPs S7-300/S7-400 para interlocutores VPN

Función específica del módulo

Para permitir el acceso a CPs S7-300/S7-400 para interlocutores VPN, consulte el apartado: Permitir el acceso a CPs S7-300/S7-400 para interlocutores VPN (Página 888) en el capítulo "Seguridad para CPs S7-300/S7-400/PC".

Configurar nodos NDIS accesibles por túnel para CPs PC

Función específica del módulo

Para configurar nodos NDIS accesibles por túnel en CPs PC, consulte el apartado: Configurar manualmente nodos NDIS accesibles por túnel en CPs PC (Página 888) en el capítulo "Seguridad para CPs S7-300/S7-400/PC".

Configurar ajustes VPN específicos del módulo y la conexión

Requisitos

El módulo forma parte de un grupo VPN.

Ajustes específicos del módulo y la conexión

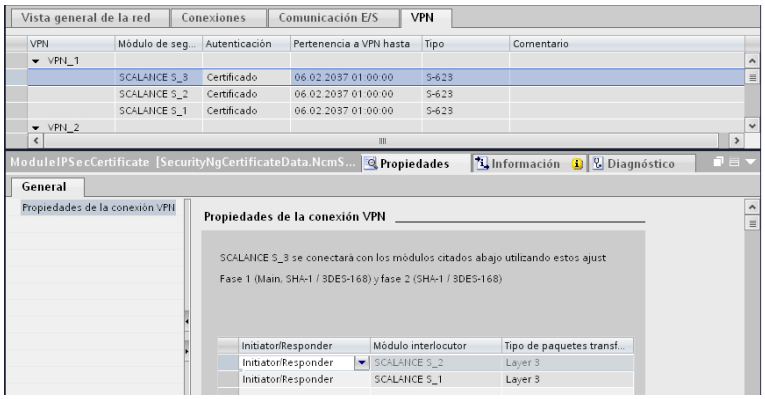
Los ajustes específicos del módulo y la conexión permiten configurar ajustes VPN concretos. Los ajustes específicos del módulo se configuran especialmente para un módulo de seguridad, mientras que los ajustes específicos de la conexión se configuran especialmente para un módulo de seguridad dentro de un grupo VPN concreto.

En la entrada "VPN" de los ajustes de seguridad locales pueden configurarse las siguientes propiedades **específicas del módulo**:

- Dead-Peer-Detection
- Permiso para iniciar el establecimiento de la conexión
- Dirección pública (dirección IP / FQDN) para la comunicación a través de pasarelas de Internet
- Nodos que deben habilitarse para la comunicación tunelada

Si en la lista de grupos VPN se selecciona un módulo de seguridad dentro del área "Datos de red", es posible ver y configurar los siguientes ajustes VPN **específicos de la conexión**:

- Permiso para iniciar el establecimiento de la conexión
- Módulos interlocutores con los que existen conexiones tuneladas
- Tipo de paquetes transmitidos
- Selección de la interfaz local del módulo de seguridad seleccionado que debe actuar de punto final del túnel
- Selección de la interfaz del interlocutor que debe actuar de punto final del túnel



Dead-Peer-Detection (DPD)

Como estándar está activado DPD.

Estando activado DPD, los módulos intercambian mensajes adicionales a intervalos de tiempo ajustables, siempre que en el instante en cuestión no se produzca ninguna comunicación. Eso permite detectar si la conexión IPsec todavía es válida o si es necesario volver a establecerla. Si ya no hay conexión, se finalizan prematuramente las "Security Associations" (SA) de fase 2. Con DPD desactivado, la SA no finaliza hasta haber concluido su vida útil. Para ajustar la vida útil de la SA consulte el apartado:
Propiedades de grupo para grupos VPN seleccionados (Página 816).

Permiso para iniciar el establecimiento de la conexión

Se puede limitar el permiso para iniciar el establecimiento de la conexión de VPN a determinados módulos de la VPN.

El factor decisivo para el ajuste del parámetro aquí descrito es la asignación de la dirección IP para la gateway del módulo configurado. En el caso de una dirección IP asignada estáticamente, el módulo puede ser encontrado por el interlocutor. En el caso de una dirección IP asignada dinámicamente, y por lo tanto constantemente cambiante, el interlocutor no puede establecer sin más una conexión.

Modo	Significado
Iniciar conexión con interlocutor (iniciador/respondedor) (predeterminado)	Con esta opción, el módulo está "activo", es decir, intenta establecer una conexión con el interlocutor con una dirección IP fija. Esta opción se recomienda si el proveedor ha asignado una dirección IP dinámica para la gateway del módulo de seguridad configurado. El direccionamiento del interlocutor tiene lugar a través de su dirección IP de módulo externa o de su dirección IP WAN configurada.
Esperar a interlocutor (respondedor)	Con esta opción, el módulo está "pasivo", es decir, espera a que el interlocutor inicie el establecimiento de la conexión. Esta opción se recomienda si el proveedor ha asignado una dirección IP estática para la gateway del módulo de seguridad configurado. Con esto se consigue que solo el interlocutor intente establecer la conexión. Dicho interlocutor puede tener una dirección IP WAN dinámica, por ejemplo.

Nota

No ponga todos los módulos de un grupo VPN a "Esperando al interlocutor", pues de hacerlo no se establece ninguna conexión.

Dirección IP WAN / FQDN - direcciones IP de los módulos y pasarelas en una VPN vía Internet

Para el funcionamiento de una VPN con túneles IPsec a través de Internet se necesitan, por regla general, direcciones IP adicionales para las gateways de Internet, como por ejemplo routers DSL. Los distintos módulos de seguridad o SCALANCE M tienen que conocer las direcciones IP externas de los módulos interlocutores de la VPN.

Nota

Para el uso de una WAN como red externa pública indique como dirección IP externa la dirección IP proporcionada por el proveedor, a través de la cual el módulo de seguridad estará accesible en la WAN (Internet). Para que el módulo de seguridad pueda enviar paquetes a través de la WAN hay que introducir el router DSL como "Router estándar".

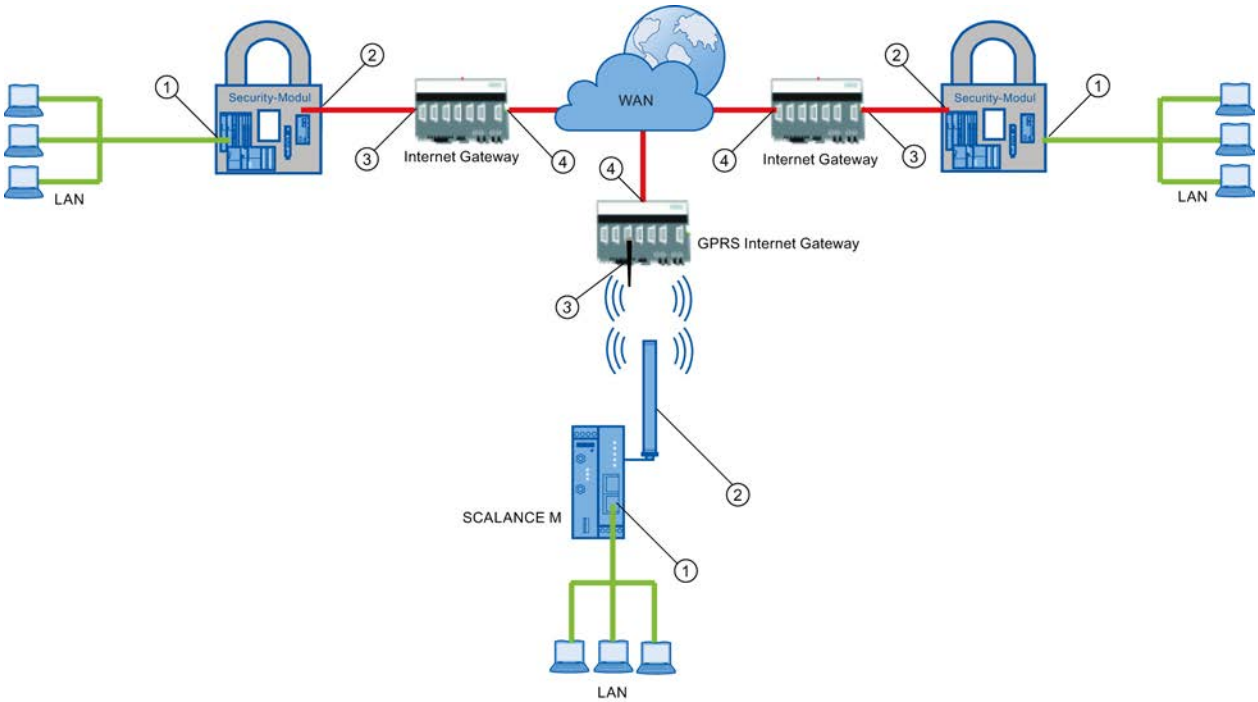
Si utiliza un router DSL como gateway de Internet, habilite en él al menos los puertos siguientes:

- Port 500 (ISAKMP)
- Port 4500 (NAT-T)

Para ello se puede asignar en la configuración del módulo una dirección IP externa como "dirección IP WAN". Al cargar la configuración del módulo, se comunica a los dispositivos del grupo las direcciones IP WAN de los módulos interlocutores. Como alternativa a una dirección IP WAN también puede introducir un FQDN. Si ha configurado, a la vez, DNS dinámico en el módulo de seguridad, este FQDN debe coincidir con el FQDN introducido en la configuración de DNS din. y registrado con un proveedor para DNS dinámico.

Puede definir en los ajustes VPN específicos del módulo de seguridad si debe utilizarse la dirección IP externa, la dirección IP de la interfaz DMZ (solo SCALANCE S623 / S627-2M) o la dirección IP WAN / el FQDN. Aquí también puede ajustar la interfaz a través de la que se comunicarán los dispositivos de un grupo VPN y cuál es el módulo de seguridad que tiene permitido establecer la conexión.

Si no especifica un punto de acceso aquí, se utilizará como punto final VPN la dirección IP externa o la dirección IP de la interfaz DMZ (solo SCALANCE S623 / S627-2M).



① Dirección IP interna - de un módulo
② Dirección IP externa - de un módulo
③ Dirección IP de una pasarela de Internet (p. ej. pasarela GPRS)
④ Dirección IP (dirección IP WAN) de una pasarela de Internet (p. ej. router DSL)

Configurar nodos de red internos

La configuración de nodos de red internos se describe en el capítulo siguiente: Auto-Hotspot

Consulte también

Cómo crear túneles IPsec mediante grupos VPN (Página 813)

Prueba de las funciones online / Diagnóstico y registro

Vista general del diagnóstico y registro

El módulo de seguridad se ha dotado de funciones de diagnóstico y registro con fines de comprobación y supervisión.

- Funciones de diagnóstico
  Con este término se conocen las funciones de sistema y de estado que pueden aplicarse al módulo de seguridad en caso de existir una conexión de red.
- Funciones de registro
  Se trata del registro de eventos del sistema y seguridad, así como de paquetes de datos.

Registrar eventos con funciones logging

Con los ajustes de registro del módulo de seguridad en cuestión se especifica qué eventos deben registrarse.

Para el registro pueden configurarse las siguientes variantes:

- Registro local
  Con esta variante se registran los eventos en el búfer local del módulo de seguridad. En el cuadro de diálogo "Online & Diagnóstico" es posible acceder a dichos registros, visualizarlos y archivarlos en la Service Station. Para la evaluación de las áreas de memoria temporal del módulo de seguridad se requiere una conexión de red con el módulo de seguridad seleccionado.
- Registro del sistema en red
  Para el registro del sistema en red se utiliza un servidor Syslog existente en la red. Este registra los eventos en función de la configuración de los ajustes de registro del módulo de seguridad en cuestión.

Archivar datos de registro y leerlos de un archivo

Los eventos registrados pueden guardarse en un archivo de registro con fines de archivación y abrirse después sin que haya una conexión de red con el módulo de seguridad. Encontrará más información en el capítulo:
Auto-Hotspot.

Vista general de las funciones del cuadro de diálogo Online

Funciones del cuadro de diálogo "Online y diagnóstico"

El módulo de seguridad ofrece las siguientes funciones en el cuadro de diálogo "Online y diagnóstico" de STEP 7:

Entrada en el cuadro de diálogo online	Función
Status	Visualización de la información de estado del módulo de seguridad seleccionado, como las direcciones IP actuales de las interfaces o la hora y fecha actuales.
Ajustes de interfaz (solo para SCALANCE S V3 o superior)	Vista general de los ajustes de las diferentes interfaces.
DNS din. (solo para SCALANCE S V3 o superior)	Vista general de los ajustes para DNS dinámico.
Registro del sistema	Visualización de eventos del sistema registrados, inicio y parada del registro (solo si existe una conexión online con módulos SCALANCE S) así como inicio y parada de la lectura de los datos de registro del búfer local del módulo de seguridad.
Registro de auditoría	Visualización de eventos de seguridad registrados, inicio y parada de la lectura de los datos de registro del búfer local del módulo de seguridad.
Registro de filtros de paquetes	Visualización de paquetes de datos registrados, inicio y parada del registro (solo si existe una conexión online con módulos SCALANCE S) así como inicio y parada de la lectura de los datos de registro del búfer local del módulo de seguridad.
Tabla ARP (solo para SCALANCE S V3 o superior)	Visualización de la tabla ARP del módulo de seguridad.
Usuarios dados de alta (solo para SCALANCE S V3 o superior)	Visualización de los usuarios que han iniciado sesión en la página de Internet para conjuntos de reglas IP específicas de usuario.
Estado de la comunicación (no para SCALANCE S602 y CP 1543-1 V1.0)	Visualización de la información de estado de las conexiones VPN tuneladas así como de los dispositivos de grupos VPN a los que pertenece el módulo de seguridad seleccionado.
Nodos internos (no para SCALANCE S602, CP 1543-1 V1.0 y CPs PC)	Visualización de los nodos de red internos (aprendidos o configurados) del módulo de seguridad.
Archivos de registro (vista offline)	
• Registro del sistema	Visualización de los eventos de sistema registrados, así como inicio y parada de la visualización.
• Registro de auditoría	Visualización de los eventos de seguridad registrados, así como inicio y parada de la visualización.
• Registro de filtros de paquetes	Visualización de los paquetes de datos registrados, así como inicio y parada de la visualización.
Reglas de cortafuegos actualizadas dinámicamente (solo para CPs S7)	Visualización de las direcciones IP habilitadas dinámicamente vía HTTP o HTTPS o cargadas posteriormente por un usuario.
Fecha y hora (solo para SCALANCE S)	Ajuste de la fecha y la hora.

Condiciones para el acceso

Para poder utilizar las funciones online de un módulo de seguridad, deben cumplirse los siguientes requisitos:

- existe una conexión de red con el módulo seleccionado
- está abierto el proyecto con el que se ha configurado el módulo
- un usuario ha iniciado sesión en el proyecto con los derechos necesarios

Cómo se accede a esta función

1. Haga clic con el botón derecho del ratón en el módulo que desea editar.
2. En el menú contextual, elija el comando de menú "Online y diagnóstico".
3. Si todavía no se ha establecido ninguna conexión online con el módulo de seguridad, haga clic en el botón "Establecer conexión online" de la entrada "Diagnóstico".

Ajustes online no se almacenan en la configuración

Los ajustes realizados en el modo online (p. ej. ajustes de la memoria de registro) no se guardan en la configuración del módulo de seguridad. Por ello, al rearrancar un módulo siempre se aplican los ajustes de la configuración.

Funciones del diagnóstico online

Información de estado del módulo de seguridad - entrada "Estado"

Significado

Visualización del estado del módulo de seguridad seleccionado en el proyecto.

Tabla 8-34 Online y diagnóstico: entrada "Estado"

Funciones de sistema y estado	Significado
Sinopsis	
Tipo de hardware	El tipo del módulo de seguridad.
Dirección IP externa	La dirección IP externa del módulo de seguridad. Para CP 1628: La dirección IP de la interfaz Industrial Ethernet. Para CPs S7: la dirección IP de la interfaz GBit.
Dirección IP interna	La dirección IP interna del módulo de seguridad. Para CP 1628: La dirección IP de la interfaz NDIS. En caso de varias direcciones NDIS solo se mostrará una. Para CPs S7: la dirección IP de la interfaz PROFINET.

Funciones de sistema y estado	Significado
Dirección IP DMZ (solo SCALANCE S623 / S627-2M)	La dirección IP DMZ del módulo de seguridad.
Número de serie	El número de serie del módulo de seguridad.
Referencia	La identificación de referencia del módulo de seguridad que se utiliza al realizar un pedido.
Versión de firmware	La versión de firmware del módulo de seguridad.
Modo de operación	Modo de operación del módulo de seguridad (modo Bridge / modo de enrutamiento)
Dirección MAC externa	La dirección MAC externa del módulo de seguridad. Para CP 1628: La dirección MAC de la interfaz Industrial Ethernet. Para CPs S7: la dirección MAC de la interfaz Gbit.
Dirección MAC interna	La dirección MAC interna del módulo de seguridad. Para CP 1628: La dirección MAC de la interfaz NDIS. Para CPs S7: la dirección MAC de la interfaz PROFINET.
Dirección MAC DMZ (solo SCALANCE S623 / S627-2M)	La dirección MAC DMZ del módulo de seguridad.
Revisión de HW	El estado de producto de hardware del módulo de seguridad.
C-PLUG	Indica si hay un C-PLUG enchufado o no.
Hora local	
Hora actual	Fecha y hora indicadas en el módulo de seguridad. Formato con idioma de interfaz "alemán": dd.mm.yyyy (fecha) hh:mm:ss (hora) Formato con idioma de interfaz "inglés": mm/dd/yyyy (fecha) hh:mm:ss AM/PM (hora) Formato con idioma de interfaz "francés": dd/mm/yyyy (fecha) hh:mm:ss (hora) Formato con idioma de interfaz "italiano": dd/mm/yyyy (fecha) hh:mm:ss (hora) Formato con idioma de interfaz "español": dd/mm/yyyy (fecha) hh:mm:ss (hora) **Nota (no para CPs)** Ajuste la hora local en el módulo SCALANCE S mediante "Funciones" > "Fecha y hora".

Funciones de sistema y estado	Significado
Tiempo de funcionamiento	Tiempo que ha transcurrido desde el último rearranque completo del módulo de seguridad. Formato con idioma de interfaz "alemán": dddd.hh:mm:ss Formato con idioma de interfaz "inglés": dddd.hh:mm:ss Formato con idioma de interfaz "francés": dddd.hh:mm:ss Formato con idioma de interfaz "italiano": dddd.hh.mm.ss Formato con idioma de interfaz "español": dddd.hh:mm:ss
Hora fuente	La fuente que suministra la fecha y hora.
Configuración	
Creado	Fecha y hora en las que fue creado el proyecto por primera vez. Formato con idioma de interfaz "alemán": dddd.hh:mm:ss hh:mm:ss (hora) Formato con idioma de interfaz "inglés": dddd.hh:mm:ss hh:mm:ss AM/PM (hora) Formato con idioma de interfaz "francés": dddd.hh:mm:ss hh:mm:ss (hora) Formato con idioma de interfaz "italiano": dddd.hh.mm.ss hh.mm.ss (hora) Formato con idioma de interfaz "español": dddd.hh:mm:ss hh:mm:ss (hora)
Nombre	Nombre de archivo que se utilizó para guardar el proyecto. STEP 7 predetermina el nombre, que cambia si cambia el nombre del proyecto.
Autor	Nombre del usuario que ha creado el proyecto. Se adopta de las propiedades del proyecto.
Cargado	Fecha y hora en las que el proyecto fue cargado por última vez en el módulo de seguridad. Formato con idioma de interfaz "alemán": dddd.hh:mm:ss hh:mm:ss (hora) Formato con idioma de interfaz "inglés": dddd.hh:mm:ss hh:mm:ss AM/PM (hora) Formato con idioma de interfaz "francés": dddd.hh:mm:ss hh:mm:ss (hora) Formato con idioma de interfaz "italiano": dddd.hh.mm.ss hh.mm.ss (hora) Formato con idioma de interfaz "español": dddd.hh:mm:ss hh:mm:ss (hora)
Ubicación	Indicación del lugar (p. ej. ciudad) que se ha ajustado mediante una herramienta SNMP. Para módulos SCALANCE S también es posible configurar la ubicación en los ajustes avanzados de SNMP.
Sistema de archivos (no para CPs)	

Funciones de sistema y estado	Significado
RAM	Se indica la cantidad de memoria RAM y flash ocupadas en el sistema de archivos.
Flash	

Sinopsis de las diferentes interfaces - Entrada "Ajustes de interfaz"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior, consulte el apartado: Sinopsis de las diferentes interfaces - Entrada "Ajustes de interfaz" (Página 871)

Sinopsis de los ajustes DDNS - Entrada "DNS din."

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior, consulte el apartado: Sinopsis de los ajustes DDNS - Entrada "DNS din." (Página 872)

Visualización de la tabla ARP - Entrada "Tabla ARP"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior, consulte el apartado: Visualización de la tabla ARP - Entrada "Tabla ARP" (Página 873)

Usuarios registrados en página web - Entrada "Usuarios dados de alta"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior, consulte el apartado: Usuarios registrados en página web - Entrada "Usuarios dados de alta" (Página 874)

Conexiones VPN del módulo de seguridad - entrada "Estado de la comunicación"

Función específica del módulo

Esta función no está disponible para módulos SCALANCE S602.

Significado

Muestra el estado de comunicación de los componentes de red siguientes:

- Otros módulos de seguridad del grupo VPN al que pertenece el módulo de seguridad seleccionado.
- Nodos de red internos de este módulo de seguridad.

Tabla 8-35 Online y diagnóstico: entrada "Estado de la comunicación"

Funciones de sistema y estado	Significado
Dispositivos o módulos de seguridad conocidos	Se muestran los dispositivos con los que comparte grupo VPN el módulo de seguridad seleccionado. También se muestra si el estado de túnel es activo o no. Para obtener información adicional sobre un dispositivo, selecciónelo en la lista. Nota: Los túneles configurados que no están activos se muestran solo para CPs.
Puntos finales	Visualización de información sobre los nodos de red internos del módulo de seguridad que se ha seleccionado en la tabla "Dispositivos o módulos de seguridad conocidos". Para cada nodo de red interno se muestra si se ha aprendido o configurado. También se indica la subred en la que se encuentra el nodo de red interno. Con módulos SCALANCE S, la subred de los nodos de red solo se muestra en el modo Bridge.
Propiedades del túnel	Visualización de las propiedades del túnel VPN que se ha establecido con el módulo de seguridad que se ha seleccionado en la tabla "Dispositivos o módulos de seguridad conocidos".

Nodos de red internos encontrados - Entrada "Nodos internos"

Función específica del módulo

Esta función no está disponible para SCALANCE S602 ni para CP 1543-1.

Significado

Visualización de todos los nodos de red aprendidos y configurados. También se indica si el modo de aprendizaje del módulo de seguridad está activado o desactivado.

Reglas de cortafuegos actualizadas - Entrada "Reglas del cortafuegos actualizadas de forma dinámica"

Función específica del módulo

Esta función solo está disponible para CP x43-1; consulte el apartado:
Reglas de cortafuegos actualizadas - Entrada "Reglas del cortafuegos actualizadas de forma dinámica" (Página 888) en el capítulo "Seguridad para CPs S7-300/S7-400/PC".

Visualización de la lista negra de cortafuegos - Entrada "Lista negra de cortafuegos"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V4 o superior, consulte el apartado:
Visualización de la lista negra de cortafuegos - Entrada "Lista negra de cortafuegos" (Página 874)

Ajustar la fecha y hora - Entrada "Fecha y hora"

Función específica del módulo

Esta función solo está disponible para módulos SCALANCE S, consulte el apartado:
Ajustar la fecha y hora - Entrada "Fecha y hora" (Página 875).

Funciones de registro

Registrar eventos del sistema - Entrada "Registros del sistema"

Significado

Visualización de eventos del sistema registrados así como inicio y parada de la lectura de los eventos del sistema de la memoria local del módulo de seguridad.

El registro del sistema registra de forma automática y continua eventos del sistema como p. ej. el inicio de un proceso. El registro se puede escalar basándose en clases de eventos.

Funciones de sistema y estado	Significado
Iniciando/Deteniendo registro (no para CPs)	Inicia o detiene el registro de eventos del sistema. Tanto el método como las clases de eventos que se registran se configuran en los ajustes de seguridad locales.
Iniciando/Deteniendo lectura	Inicia o detiene la lectura de eventos del sistema desde la memoria local del módulo de seguridad. Si se activa la casilla de verificación "Guardar archivo de registro" se guardarán los datos de registro registrados también como archivo. Seleccione la ubicación e indique un nombre para el archivo. **Nota** Si se ha activado la casilla de verificación "Guardar archivo de registro" después de iniciar el proceso de lectura, no se podrán guardar los datos leídos hasta el momento en un archivo de registro.
Borrar visualización	Borra los datos de registro mostrados en la tabla.

Para abrir eventos del sistema guardados en archivos de registro, consulte el capítulo: Evaluar eventos del sistema en modo offline - Entrada "Registros del sistema" (vista Offline) (Página 838).

Registrar eventos de seguridad - Entrada "Registros de auditoría"

Significado

Visualización de eventos de seguridad registrados así como inicio y parada de la lectura de los eventos de seguridad de la memoria local del módulo de seguridad.

El registro de auditoría registra de forma automática y continua eventos relevantes para la seguridad. Entre ellos, p. ej. acciones del usuario como conexión o desconexión del registro de paquetes.

Funciones de sistema y estado	Significado
Iniciando/Deteniendo lectura	Inicia o detiene la lectura de eventos de seguridad desde la memoria local del módulo de seguridad. Si se activa la casilla de verificación "Guardar archivo de registro" se guardarán los datos de registro registrados también como archivo. Seleccione la ubicación e indique un nombre para el archivo. **Nota** Si se ha activado la casilla de verificación "Guardar archivo de registro" después de iniciar el proceso de lectura, no se podrán guardar los datos leídos hasta el momento en un archivo de registro.
Borrar visualización	Borra los datos de registro mostrados en la tabla.

Para abrir eventos de seguridad guardados en archivos de registro, consulte el capítulo: Evaluar eventos de seguridad en modo offline - Entrada "Registros de auditoría" (vista Offline) (Página 838).

Registrar paquetes de datos - Entrada "Registros de filtrado de paquetes"

Significado

Visualización de los paquetes de datos registrados, así como inicio y parada de los eventos de filtrado de paquetes.

El registro de filtrado de paquetes registra determinados paquetes del tráfico de datos. Solo se registran paquetes de datos para los que sea válida una regla de filtrado de paquetes (cortafuegos) configurada o frente a los que reacciona la protección básica (paquetes corruptos o no válidos). Condición para ello es que esté activado el registro para la regla de filtrado de paquetes.

Encontrará información para activar el registro en el capítulo:
Auto-Hotspot.

Los datos registrados, además de ser leídos en el búfer y transferidos para su visualización, también pueden ser almacenados en un archivo.

Funciones de sistema y estado	Significado
Iniciando/Deteniendo registro (no para CPs)	Inicia o detiene el registro de paquetes de datos. El método con el que se registran los datos se configura en los ajustes de seguridad locales.
Iniciando/Deteniendo lectura	Inicia o detiene la lectura de paquetes de datos registrados desde la memoria local del módulo de seguridad. Si se activa la casilla de verificación "Guardar archivo de registro" se guardarán los datos de registro registrados también como archivo. Seleccione la ubicación e indique un nombre para el archivo. **Nota** Si se ha activado la casilla de verificación "Guardar archivo de registro" después de iniciar el proceso de lectura, no se podrán guardar los datos leídos hasta el momento en un archivo de registro.
Borrar visualización	Borra los datos de registro mostrados en la tabla.
Categoría de registro	Seleccione los paquetes de datos cuyo registro desea visualizar. La selección depende de los ajustes de seguridad locales configurados offline. Se registran solo los paquetes de datos para los que se haya activado el registro. Si se selecciona una categoría para la que no se ha activado el registro, tampoco se muestran datos para dicha categoría.

Para abrir los datos guardados de registro de filtrado de paquetes, consulte el capítulo: Evaluar eventos de filtrado de paquetes en modo offline - Entrada "Registros de filtrado de paquetes" (vista Offline) (Página 839).

Evaluar archivos de registro en modo offline

Evaluar eventos del sistema en modo offline - Entrada "Registros del sistema" (vista Offline)

Cómo se accede a esta función

1. Haga clic con el botón derecho del ratón en el módulo que desea editar.
2. Elija la entrada "Online y diagnóstico" en el menú contextual.
3. Elija la entrada "Diagnóstico" > "Archivos de registros (vista offline)" > "Registros del sistema".

Significado

Apertura de eventos del sistema registrados guardados como archivo en la vista online.

Encontrará más información en el capítulo Registrar eventos del sistema - Entrada "Registros del sistema" (Página 835).

Evaluar eventos de seguridad en modo offline - Entrada "Registros de auditoría" (vista Offline)

Cómo se accede a esta función

1. Haga clic con el botón derecho del ratón en el módulo que desea editar.
2. Elija la entrada "Online y diagnóstico" en el menú contextual.
3. Elija la entrada "Diagnóstico" > "Archivos de registros (vista offline)" > "Registros de auditoría".

Significado

Apertura de eventos de seguridad registrados guardados como archivo en la vista online.

Encontrará más información en el capítulo Registrar eventos de seguridad - Entrada "Registros de auditoría" (Página 836).

Evaluar eventos de filtrado de paquetes en modo offline - Entrada "Registros de filtrado de paquetes" (vista Offline)

Cómo se accede a esta función

1. Haga clic con el botón derecho del ratón en el módulo que desea editar.
2. Elija la entrada "Online y diagnóstico" en el menú contextual.
3. Elija la entrada "Diagnóstico" > "Archivos de registros (vista offline)" > "Registros de filtrado de paquetes".

Significado

Apertura de paquetes de datos registrados guardados como archivo en la vista online.

Encontrará más información en el capítulo Registrar paquetes de datos - Entrada "Registros de filtrado de paquetes" (Página 837).

Funciones de carga

Particularidades al cargar configuraciones de seguridad

Las configuraciones de seguridad pueden influir en la accesibilidad del módulo de seguridad desde el PC de configuración. Esto sucede, por ejemplo, cuando en una configuración de un módulo de seguridad se ha configurado una conexión tunelada con otro módulo de seguridad y dicha configuración se carga en el módulo de seguridad desde el PC de configuración. Después de la carga desde el PC de configuración ya no es posible acceder al módulo de seguridad y el test de seguridad que se lleva a cabo de forma estándar desde STEP 7 después de cargar una configuración, falla. El mensaje de error que emite a continuación STEP 7 hace referencia únicamente al test de accesibilidad; la carga de la configuración propiamente dicha está garantizada siempre que los datos del proyecto sean coherentes y la relación de direcciones IP entre el módulo de seguridad y el PC de configuración sea correcta.

Las particularidades al cargar configuraciones y firmware en módulos SCALANCE S están descritas en el apartado siguiente del capítulo "SCALANCE S":
Auto-Hotspot

Cargar configuraciones en la estación de ingeniería

No es posible cargar una configuración en una estación de ingeniería desde un módulo SCALANCE S o desde el CP 1628.

La carga de una configuración en una estación de ingeniería desde un CP S7 compatible con seguridad también es posible cuando se han configurado funciones de seguridad en la configuración. Sin embargo, las funciones de seguridad no se transfieren a la estación de ingeniería. Asimismo, la casilla de verificación "Activar funciones de seguridad" está desactivada en la configuración transferida a la estación de ingeniería.

SOFTNET Security Client

Uso del SOFTNET Security Client

Campo de aplicación - acceso a través de VPN

El software para PC SOFTNET Security Client (SSC) permite acceder remotamente desde la PG o el PC a autómatas programables protegidos por un módulo de seguridad, más allá de los límites de redes públicas. Para CPs S7-300/S7-400 y para el CP PC CP 1628 se necesita SOFTNET Security Client V4.0 HF1. Con SOFTNET Security Client ≤ V4.0, estos CPs no están habilitados. Para el CP 1543-1 no está habilitado el SOFTNET Security Client.

Mediante el SOFTNET Security Client se configura automáticamente una PG o un PC de manera que pueda establecer con uno o varios módulos de seguridad una comunicación por túnel IPsec protegida en la VPN (Virtual Private Network).

Por medio de esta comunicación por túnel IPsec es posible acceder de forma segura, mediante aplicaciones de PG/PC como Diagnóstico NCM, a dispositivos o redes que se encuentren en una red interna protegida por SCALANCE S.

¿Cómo funciona el SOFTNET Security Client?

El SOFTNET Security Client lee la configuración creada por STEP 7 y determina, sobre la base del archivo, los certificados que deben importarse.

El certificado CA y, dado el caso, las claves privadas se importan y se almacenan en la PG/el PC local.

A continuación, con los datos de la configuración, se adoptan los ajustes de Security para que las aplicaciones puedan acceder a direcciones IP que se encuentren detrás de los módulos de seguridad.

Si está activado el modo de aprendizaje para los dispositivos internos o los autómatas programables, se establece primero una directiva de seguridad para el acceso seguro a los módulos de seguridad. El SOFTNET Security Client interroga a continuación a los módulos de seguridad para determinar las direcciones IP de los respectivos dispositivos internos.

El SOFTNET Security Client registra esas direcciones IP en listas de filtrado especiales de esa directiva de seguridad. Después de esto las aplicaciones se pueden comunicar con los autómatas programables a través de VPN.

Crear un archivo de configuración en STEP 7

Configuración del módulo SOFTNET Security Client en el proyecto

El SOFTNET Security Client se habilita en el proyecto como módulo. A diferencia de los demás módulos de seguridad, no es necesario configurar otras propiedades.

Asigne el módulo SSC al grupo o a los grupos VPN en los que deben configurarse túneles IPsec a la PG o el PC.

Nota

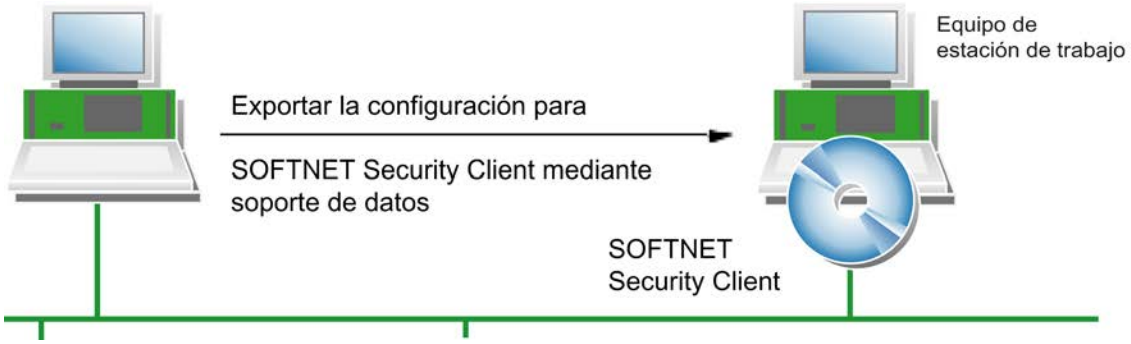
Tenga en cuenta las indicaciones sobre los parámetros en el capítulo:
Incluir un módulo de seguridad en un grupo VPN configurado (Página 820).

Nota

Si crea varios SOFTNET Security Clients dentro de un grupo, no se establece ningún túnel entre ellos, sino solo entre el cliente correspondiente y los módulos de seguridad.

Archivos de configuración para SOFTNET Security Client

La interfaz entre la herramienta de configuración STEP 7 y el SOFTNET Security Client funciona mediante archivos de configuración.



La configuración se guarda en los siguientes tipos de archivo:

- *.dat
- *.p12
- *.cer

Procedimiento

Para generar los archivos de configuración, realice los siguientes pasos en STEP 7:

1. Seleccione la ficha "Vista topológica" o la ficha "Vista de redes" en la vista "Dispositivos y redes".
2. Inserte en la ficha seleccionada un sistema PC del tipo "SOFTNET Security Client" desde el catálogo de hardware.
3. Asigne el SOFTNET Security Client a los grupos VPN en los que la PG o el PC se deba comunicar a través de túneles IPsec.
4. Asegúrese de que la casilla de verificación "Generar archivos SSC" está activada en la entrada "Configuración del SOFTNET Security Client" de los ajustes de seguridad locales del SOFTNET Security Client.

5. Seleccione la ruta de almacenamiento para los archivos de configuración.
6. Compile la configuración del SOFTNET Security Client para exportar el archivo de configuración.
7. Si ha elegido "Certificate" como método de autenticación, indique una contraseña para el certificado de la configuración de VPN. Si no asigna ninguna contraseña, se adopta el nombre del proyecto (no la contraseña del proyecto) como contraseña.
   Resultado: La exportación de los archivos de configuración ha concluido.
8. Transfiera los archivos del tipo *.dat, *.p12, *.cer a la PG o el PC en el que desee utilizar el SOFTNET Security Client.

SCALANCE S

Configurar interfaces para módulos SCALANCE S

Sinopsis

Configuración del modo de operación

El modo de operación permite definir el modo de enrutamiento de las interfaces (externa/interna). La interfaz DMZ del módulo de seguridad (solo SCALANCE S623/627-2M) siempre está conectada en el modo de enrutamiento. Encontrará más información en el apartado: Configurar parámetros de dirección IP (Página 844)

Configuración de las interfaces

Si debe configurarse la interfaz externa o la interfaz DMZ (solo SCALANCE S623/S627-2M) de un módulo de seguridad, estas deben estar activadas a través de la casilla de verificación "Activar interfaz". Defina las direcciones IP de cada interfaz y los ajustes de los diferentes puertos. Para asignar una dirección IP están disponibles los modos de asignación siguientes para la interfaz externa y la interfaz DMZ (solo SCALANCE S623/S627-2M) en la entrada "General":

- Dirección IP estática con máscara de subred. Encontrará más información en el apartado: Configurar parámetros de dirección IP (Página 844)
- Asignación de dirección mediante PPPoE. Encontrará más información en el apartado: Configurar una conexión a Internet (Página 846)
  La interfaz interna solo puede configurarse utilizando una dirección IP estática.

Nota

Interfaz externa e interfaz DMZ como acceso a Internet

No es posible el funcionamiento simultáneo de PPPoE en la interfaz externa y en la interfaz DMZ (Dual-ISP).

Point to Point Protocol over Ethernet (PPPoE)

Para permitir una conexión Internet/WAN directamente a través de un módem DSL, la asignación de la dirección IP se realiza en la interfaz externa o en la interfaz DMZ mediante PPPoE. PPPoE es un protocolo de llamada para la obtención de direcciones IP desde un Internet Service Provider (ISP). SCALANCE S funciona en ese caso en modo de enrutamiento.

Para utilizar este método de asignación de dirección IP, introduzca los datos del ISP en la entrada "PPPoE". La dirección IP, la máscara de subred, el router estándar y el servidor DNS de la interfaz son entonces predefinidos por el ISP.

Nota

Si hay configurado un router predeterminado, este no se tiene en cuenta si se utiliza PPPoE. Será preasignado dinámicamente al módulo por el ISP.

Nota

Ningún componente de red entre SCALANCE S y el módem DSL

Si la interfaz de un módulo SCALANCE S funciona mediante PPPoE, no debe haber ningún otro componente de red entre dicha interfaz y el módem DSL conectado, ya que los datos de marcación del Internet Service Provider se transfieren sin codificar en este tramo debido a necesidades de protocolo.

Configuración de módulos de medios

Además de las funciones del SCALANCE S623, el S627-2M posee dos slots para módulos de medios, en cada uno de los cuales se puede usar un módulo de medios eléctrico u óptico de dos puertos. Las interfaces externa e interna se amplían de esta forma con dos puertos cada una. En el modo de enrutamiento se pueden utilizar los puertos adicionales del módulo de seguridad para la conexión de las interfaces externa e interna a topologías de anillo.

Para integrar módulos de medios en el SCALANCE S627-2M, seleccione el módulo de seguridad y cambie a la vista de dispositivos. Seguidamente, seleccione los módulos de medios deseados en el catálogo de hardware.

Para los puertos del tipo "cobre" es posible definir manualmente la velocidad de transferencia y el modo dúplex/semidúplex por medio del modo de puerto. Para los puertos del tipo "óptico", el modo de puerto está predeterminado por el módulo de medios o el SFP utilizado y no se puede adaptar.

Encontrará información sobre la conexión de puertos para módulos de medios a anillos MRP en el siguiente capítulo:
Auto-Hotspot

Consulte también

Ajustar el modo de operación (Página 844)

Ajustar el modo de operación

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Modo de operación".

Modo de operación - Posibilidades de selección

Si el módulo de seguridad no se encuentra en ningún grupo VPN, es posible modificar el modo de operación en esta entrada. Si el módulo de seguridad se encuentra en un grupo VPN, no es posible modificar el modo de operación.

La selección es válida para el enrutamiento de interfaces entre la interfaz externa y la interna. La interfaz DMZ (solo SCALANCE S623 y S627-2M) siempre está conectada en el modo de enrutamiento.

Modo Bridge	Para el uso en redes planas. Las interfaces externa e interna están en la misma subred IP. Para S623 / S627-2M: las interfaces externa e interna están en la misma subred IP, la interfaz DMZ está en otra subred IP o está desactivada.
Modo de enrutamiento	Todas las interfaces están en diferentes subredes IP. Si se ha activado el modo de enrutamiento, hay que configurar una dirección IP interna y una máscara de subred interna para la interfaz interna del módulo de seguridad. **Nota** Si se ha activado el modo de enrutamiento para el módulo SCALANCE S no podrán definirse reglas para cortafuegos MAC.

Configurar parámetros de dirección IP

Significado

Definir parámetros de red como dirección IP y máscara de subred para la o las interfaces del módulo de seguridad.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Interfaz externa [P1] roja" o "Interfaz DMZ [P3] amarilla".

   Nota

   Configuración de la interfaz interna en modo de enrutamiento

   Si se ha seleccionado el modo de operación "Enrutamiento" para el módulo de seguridad, hay que configurar adicionalmente una dirección IP interna y una máscara de subred interna para la interfaz interna del módulo de seguridad. Se accede a esta función en los ajustes de seguridad locales mediante "Interfaz interna [P2] verde" > "Direcciones Ethernet".

3. En caso necesario, active la interfaz mediante la casilla de verificación "Activar interfaz".
4. Elija la entrada "Direcciones Ethernet".
5. Realice los ajustes que se describen en la tabla siguiente.

Parámetro	Significado
Dirección IP	Dirección IP de la interfaz externa. La dirección IP consta de 4 números decimales comprendidos en el rango de 0 a 255 y separados entre ellos por un punto; p. ej. 141.80.0.16
Máscara de subred	La máscara de subred consta de 4 números decimales separados entre sí por un punto, p. ej. 255.255.0.0.
Utilizar router	Active esta casilla de verificación si desea utilizar un router estándar e introduzca su dirección en el campo de entrada "Dirección del router".

Nota

Conexión en red de las interfaces

Conecte en red las diferentes interfaces del módulo de seguridad con las subredes correspondientes para evitar conflictos de direcciones IP.

Particularidades del router estándar

- Si está configurada la asignación de IP por "PPPoE", no es necesario configurar un router predeterminado, ya que la ruta predeterminada siempre conduce automáticamente a través de la interfaz PPPoE.
- Si está configurada la asignación de IP por "Dirección estática" y si el módulo de seguridad está conectado a Internet a través de un router DSL(NAPT), hay que especificar el router DSL como router predeterminado.

Configurar el modo de puerto

Significado

El modo de puerto define la velocidad de transferencia y el modo dúplex/semidúplex. Para módulos SCALANCE S V2, el modo de puerto está ajustado por defecto a "Autonegotiation". Esto significa que la velocidad de transferencia y el modo dúplex/semidúplex se seleccionan automáticamente. Además se soporta la función Autocrossing.

Modos de puerto seleccionables

Para SCALANCE S V3 o superior pueden configurarse los modos de puerto siguientes:

Modo de puerto	Significado
Autonegotiation	La velocidad de transferencia y el modo dúplex/semidúplex se seleccionan automáticamente. Nota Solo si Autonegotiation está seleccionada se soportan una velocidad de transferencia de 1000 Mbits/s y la función Autocrossing.
10 Mbps, dúplex y semidúplex	Velocidad de transferencia de 10 Mbits/s
100 Mbps, dúplex y semidúplex	Velocidad de transferencia de 100 Mbits/s

Configurar una conexión a Internet

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Requisitos

La entrada "PPPoE" solo se muestra en los ajustes de seguridad locales si se ha configurado el método de asignación IP "PPPoE" para una de las interfaces.

Cómo se accede a esta función

1. Seleccione el módulo de seguridad que desea editar.
2. En los ajustes de seguridad locales elija la entrada "PPPoE".

Significado

Si se ha ajustado una conexión a través de PPPoE para una de las interfaces del módulo de seguridad, realice los ajustes para el Internet Service Provider (ISP) en esta entrada.

Tabla 8-36 Ajustes de la cuenta ISP

Función	Descripción
Autenticación	Seleccione uno de los siguientes protocolos de autenticación o ninguno: • PAP (Password Authentication Protocol) • CHAP (Challenge Handshake Authentication Protocol) **Nota** Ambos interlocutores deben utilizar el mismo método de autenticación; de lo contrario no es posible establecer ninguna conexión.
Nombre de usuario	Introduzca el nombre para el inicio de sesión en la cuenta ISP.
Contraseña	Introduzca la contraseña para el inicio de sesión en la cuenta ISP.
Repetir contraseña	Vuelva a introducir la contraseña para el inicio de sesión en la cuenta ISP.

Tabla 8-37 Ajustes de la conexión

Función	Descripción
Conexión permanente	Conexión continua a Internet. Si el proveedor deshace la conexión, esta se vuelve a establecer automáticamente, aunque no haya que enviar ningún paquete en ese momento.
Conexión bajo demanda	La conexión a Internet se establece automáticamente si hay que enviar paquetes a Internet. Con este ajuste son posibles retardos al enviar los paquetes.
Tiempo máx. de inactividad (solo con el ajuste "Conexión bajo demanda")	Si no se envía ningún paquete dentro de un tiempo determinado, la conexión a Internet se deshace automáticamente. En el campo "Tiempo máx. de inactividad", indique el tiempo en segundos pasado el cual debe interrumpirse la conexión. Valores permitidos: 10 … 3600.
Desconexión forzada (solo con el ajuste "Conexión permanente")	Active esta casilla de verificación para adaptar la hora de la desconexión forzada por parte del módulo de seguridad.
Hora de la desconexión forzada (solo con el ajuste "Conexión permanente")	El proveedor deshace la conexión a Internet automáticamente tras un tiempo determinado. Si se introduce una hora en este campo, el módulo de seguridad deshará por sí mismo la conexión a Internet a esa hora. Esto permite, en determinadas circunstancias, aplazar una desconexión de Internet por parte del proveedor. Una desconexión forzada por iniciativa propia solo es posible si existe una conexión permanente. Entradas permitidas: 00:00 ... 23:59

Configurar DNS dinámico

Función específica del módulo

Es posible configurar FQDNs para SCALANCE S V3 o superior. A partir de SCALANCE S V4 es posible generar FQDNs mediante módulos SCALANCE S.

Significado

Con un DNS dinámico se puede acceder con un nombre determinado (FQDN) a una dirección IP siempre cambiante. Esto es necesario p. ej. para acceder a un servidor disponible a través de una dirección IP pública cambiante.

Funcionamiento

El módulo de seguridad notifica a un proveedor de DNS dinámico (p. ej. DynDNS.org, no-ip.com) la dirección IP WAN actual a través de la cual se puede acceder al módulo de seguridad. El proveedor garantiza que las peticiones de DNS al FQDN del módulo de seguridad se respondan con la dirección IP WAN actual del módulo de seguridad.

El DNS dinámico está permitido en las siguientes interfaces:

- Interfaz externa
- Interfaz DMZ

Requisitos para configurar el DNS dinámico

Requisitos:

- Se ha creado una cuenta para un proveedor de DNS dinámico y se ha registrado un FQDN.

Procedimiento para instalar el DNS dinámico:

1. Seleccione el módulo de seguridad que desea editar.
2. En los ajustes de seguridad locales elija la entrada "DNS".
3. Si el módulo de seguridad se encuentra después de un router o módem DSL, indique la dirección de un servidor DNS válido. Para ello existen dos opciones:

Opción	Significado
Obtener automáticamente la dirección del servidor DNS	La dirección del servidor DNS puede obtenerse automáticamente mediante PPPoE, siempre que el módulo de seguridad esté conectado a Internet a través de un módem DSL. Solo puede ajustarse para la interfaz externa y la interfaz DMZ.
Utilizar la siguiente dirección de servidor DNS:	Introduzca manualmente la dirección del servidor DNS preferido y del alternativo.

4. Active la casilla de verificación "Activar servicio" en el área "Servicio DNS din. primario" y realice los ajustes siguientes:

Ajuste	Significado
Proveedor	Elija el proveedor en el que ha configurado una cuenta para DNS dinámico.
Cuenta de usuario en el proveedor	Introduzca el nombre de usuario que definió al crear la cuenta.
Contraseña en el proveedor	Introduzca la contraseña que definió al crear la cuenta.
FQDN	Introduzca el nombre de host (p. ej. mysecuritymodule) y el nombre de dominio (p. ej. dyndns.org) registrado en el proveedor separados por un punto. Si en la entrada "VPN" se ha introducido también un FQDN, ambos deben coincidir.
Vigilar el cambio de dirección IP en el router DSL	Si el módulo de seguridad está conectado a Internet a través de un router DSL, activando esta función se activa el servicio de comprobación de IP. El módulo de seguridad envía periódicamente peticiones para determinar la dirección IP actual del router DSL y para detectar un cambio de dirección IP en el router DSL. La dirección IP determinada de este modo se envía al proveedor cada vez que se detecta un cambio.
Periodo	Indique en qué ciclo debe llamarse el servicio de comprobación de IP. Valores permitidos: 10 … 1440 minutos

5. Especifique un proveedor adicional para el caso de que fallase el primario en el área "Servicio DNS din. secundario" (ajuste opcional).

Procedimiento para instalar un proveedor personalizado:

En la lista desplegable "Proveedor", seleccione la entrada "definido por el usuario" y realice además las entradas siguientes:

Ajuste	Significado
Ignorar errores al verificar el certificado del servidor	Para que los datos de autenticación estén protegidos, el certificado del servidor de actualización se verifica de forma estándar. Si la verificación del certificado falla, la conexión HTTPS se termina y los datos de la cuenta no se transmiten. Si se activa la casilla de verificación, la función se desactiva, p. ej. si el certificado de servidor del servicio DNS dinámico no es válido (p. ej. porque ha caducado). Se recomienda no ignorar la comprobación y no activar la casilla de verificación.
URL de actualización de proveedor	En el caso de los proveedores predefinidos (DynDNS.org y No-IP.com), la URL ya aparece indicada.
URL de servicio de comprobación de IP	En el caso de los proveedores predefinidos (DynDNS.org y No-IP.com), la URL ya aparece indicada.

Configurar un LLDP

Función específica del módulo

Esta función solo está disponible para SCALANCE S V4 o superior.

Requisitos

El módulo de seguridad está en modo de enrutamiento.

Significado

LLDP (Link Layer Discovery Protocol) es un protocolo que se utiliza para detectar topologías de red. Un dispositivo apto para LLDP está en condiciones de enviar regularmente información sobre sí mismo a dispositivos vecinos y, al mismo tiempo, recibir información de ellos. La información recibida se almacena en cada dispositivo apto para LLDP en un archivo LLDP MIB. Los sistemas de gestión de redes pueden acceder a estos archivos LLDP MIB con ayuda de SNMP y así emular la topología de red existente.

Parámetros configurables

El grado de actividad del módulo de seguridad en relación con LLDP se puede configurar en la entrada "Modo LLDP" de la correspondiente interfaz.

Parámetro	Descripción
Nombre	Nombre del puerto para el que se configura el ajuste.
Modo LLDP	Modo LLDP configurado: • RxTx: enviar y recibir telegramas LLDP • Off: no enviar ni recibir telegramas LLDP

Redundancia de medios en topologías de anillo

Redundancia de medios con MRP

Función específica del módulo

Esta función solo está disponible para SCALANCE S627-2M.

Significado

Bajo el término "Redundancia de medios" se engloban distintos procedimientos para incrementar la disponibilidad de redes Industrial Ethernet en las que se puede acceder a dispositivos por diversas vías. Esto puede tener lugar por entrelazamiento de redes, conexión en paralelo de vías de transmisión o cerrando una topología lineal dándole forma de topología de anillo.

Método de redundancia de medios MRP

Para los productos SIMATIC NET existe redundancia de medios dentro de una topología de anillo si se aplica el método MRP (Media Redundancy Protocol), entre otros.

En este método se configura uno de los dispositivos como administrador de redundancia. Los demás dispositivos son clientes de redundancia. Los módulos SCALANCE S627-2M pueden adoptar exclusivamente el rol de un cliente MRP. El administrador de redundancia comprueba con telegramas de prueba la ausencia de interrupciones en el anillo. Los clientes de redundancia reenvían los telegramas de prueba. Si los telegramas de prueba del administrador de redundancia dejan de llegar al puerto en anillo del administrador de redundancia debido a una interrupción del anillo, el administrador de redundancia conecta sus dos puertos en anillo e informa inmediatamente del cambio a los clientes de redundancia.

El tiempo que necesitan los switches SCALANCE X para conectar sus puertos en anillo como administrador de redundancia es de 200 ms.

Nota sobre el uso de MRP

- MRP es compatible con topologías de anillo con un máximo de 50 dispositivos. Una superación del número de dispositivos puede hacer que falle el tráfico de datos.
- Se recomienda ajustar los puertos en anillo en cuestión a dúplex y 100 Mbits/s. En otro caso puede fallar el tráfico de datos.

Posibilidades de uso de MRP en puertos para módulos de medios

MRP se soporta exclusivamente en puertos para módulos de medios de SCALANCE S627-2M. La siguiente tabla muestra las posibilidades de uso de MRP en los puertos para módulos de medios de SCALANCE S627-2M:

Puertos en anillo	Módulo de medios 1		Módulo de medios 2	
	P4	P5	P6	P7
Cliente MRP	-	-	-	-
	Anillo 1	Anillo 1	-	-
	-	-	Anillo 2	Anillo 2
	Anillo 1	Anillo 1	Anillo 2	Anillo 2

Si hay dos anillos subordinados por cada módulo SCALANCE S, es posible la comunicación en el nivel 3 entre los anillos.

Configurar MRP para el módulo de seguridad

Requisitos

- El módulo de seguridad está en modo de enrutamiento.
- Las interfaces del módulo de seguridad que deben conectarse a anillos MRP están conectadas en red con los correspondientes administradores del anillo.

Cómo se accede a esta función

1. Seleccione el módulo de seguridad que desea editar.
2. En los ajustes de la interfaz deseada elija la entrada "Redundancia de medios".

Parámetros configurables

Parámetro	Significado	Posibilidades de selección
Dominio MRP (solo si se selecciona el rol de redundancia de medios "Cliente MRP")	Con ayuda de los dominios MRP se definen los dispositivos de un anillo MRP. Para las interfaces de todos los módulos que deban estar conectadas a un mismo anillo MRP, debe estar seleccionado el mismo dominio MRP.	Visualización del dominio MRP utilizado para la interfaz.
Rol de redundancia de medios	Selección del protocolo de redundancia de medios o desactivación de la redundancia de medios para la interfaz.	• No es estación del anillo • Cliente MRP
Puerto en anillo 1 (solo si se selecciona el rol de redundancia de medios "Cliente MRP")	Nombre del primer puerto en anillo de la interfaz seleccionada si se ha seleccionado para ella el rol de redundancia de medios "Cliente MRP".	-
Puerto en anillo 2 (solo si se selecciona el rol de redundancia de medios "Cliente MRP")	Nombre del segundo puerto en anillo de la interfaz seleccionada si se ha seleccionado para ella el rol de redundancia de medios "Cliente MRP".	-
Ajustes del dominio	A través de los ajustes del dominio pueden agregarse dominios MRP, editar los nombres de dominios MRP existentes o borrar estos dominios.	-

Parámetro	Significado	Posibilidades de selección
Protocolo de redundancia de medios alternativo	Active esta casilla de verificación para habilitar la interfaz del módulo de seguridad para otros protocolos de redundancia de medios.	• Habilitar interfaz para otros protocolos de redundancia de medios • Desactivar interfaz para otros protocolos de redundancia de medios (ajuste predeterminado)
Passive Listening	Active esta casilla de verificación si desea que la interfaz seleccionada se acople a redes externas en las que se utilice STP/RSTP (Spanning-Tree-Protocol/ Rapid-Spanning-Tree-Protocol).	• Activar Passive Listening • Desactivar Passive Listening (ajuste predeterminado)

Crear cortafuegos

Reglas de cortafuegos locales para módulos SCALANCE S

Configurar un cortafuegos con reglas predefinidas

Configurar un cortafuegos con reglas IP predefinidas

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Cortafuegos".

Cortafuegos activado de forma predeterminada

La casilla de verificación "Activar cortafuegos" está activada de forma predeterminada. Así, el cortafuegos está activado automáticamente y todos los accesos de externa al módulo de seguridad están bloqueados. Habilite las reglas de cortafuegos para los distintos sentidos activando las casillas de verificación correspondientes.

Nota

Ajustes de cortafuegos detallados en el modo de cortafuegos avanzado

En el modo de cortafuegos avanzado, las reglas de cortafuegos pueden limitarse a dispositivos determinados. Para cambiar al modo de cortafuegos avanzado, active la casilla de verificación "Activar cortafuegos en modo avanzado". Para más información sobre el modo de cortafuegos avanzado consulte el capítulo:
Resumen de reglas de cortafuegos locales (Página 784).

Configuración de cortafuegos con VPN

Si el módulo de seguridad está en un grupo VPN, la casilla de verificación "Solo comunicación tunelada" está activada de forma predeterminada. Lo que significa que a través de la interfaz externa o la interfaz DMZ solo está permitida la transferencia de datos IPsec cifrada.

Si se desactiva esta casilla de verificación, se permitirá la comunicación tunelada y, adicionalmente, los tipos de comunicación seleccionados en las otras casillas de selección.

Tabla 8-38 Reglas de cortafuegos y direcciones disponibles

Servicio	De interna a externa	De externa a interna	De interna a DMZ (solo para S623 / S627-2M)	De DMZ a interna (solo para S623 / S627-2M)	Puertos permitidos	Significado
Permitir comunicación IP	x	x	x	x	-	Se permite la comunicación IP para los sentidos de comunicación seleccionados.
Permitir protocolo S7	x	x	x	x	Puerto TCP 102	Se permite la comunicación de los dispositivos de red a través del protocolo S7.
Permitir FTP/FTPS (modo explícito)	x	x	x	x	Puerto TCP 20 Puerto TCP 21	Para la administración de archivos y el acceso a ellos entre servidor y cliente.
Permitir HTTP	x	x	x	x	Puerto TCP 80	Para la comunicación con un servidor web.
Permitir HTTPS	x	x	x	x	Puerto TCP 443	Para la comunicación segura con un servidor web, p. ej. diagnóstico web.
Permitir DNS	x	x	x	x	Puerto TCP 53 Puerto UDP 53	Se permite la conexión con un servidor DNS.
Permitir SNMP	x	x	x	x	Puerto TCP 161/162 Puerto UDP 161/162	Para vigilar dispositivos de red con capacidad SNMP.
Permitir SMTP	x	x	x	x	Puerto TCP 25	Para enviar correos electrónicos a través de un servidor SMTP.
Permitir NTP	x	x	x	x	Puerto UDP 123	Para la sincronización de la hora.
Permitir DHCP	x	x	x	x	Puerto UDP 67 Puerto UDP 68	Solo en el modo de operación Bridge: Se permite la comunicación con un servidor DHCP.

Tabla 8-39 Registro

Opción	Acción en caso de activación
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes IP transferidos a través del túnel.
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes IP entrantes que se rechazan.
Registro de paquetes salientes bloqueados	Se registran todos los paquetes IP salientes que se rechazan.

Los paquetes registrados se pueden consultar en la entrada "Registros de filtrado de paquetes" del cuadro de diálogo "Online y diagnóstico". Encontrará más información al respecto en el apartado Registrar paquetes de datos - Entrada "Registros de filtrado de paquetes" (Página 837).

Configurar un cortafuegos con reglas MAC predefinidas

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Cortafuegos".

Cortafuegos activado de forma predeterminada

La casilla de verificación "Activar cortafuegos" está activada de forma predeterminada. Así, el cortafuegos está activado automáticamente y todos los accesos de externa al módulo de seguridad están bloqueados. Habilite las reglas de cortafuegos para los distintos sentidos activando las casillas de verificación correspondientes.

Nota

Ajustes de cortafuegos detallados en el modo de cortafuegos avanzado

En el modo de cortafuegos avanzado, las reglas de cortafuegos pueden limitarse a dispositivos determinados. Para cambiar al modo de cortafuegos avanzado, active la casilla de verificación "Activar cortafuegos en modo avanzado". Para más información sobre el modo de cortafuegos avanzado consulte el capítulo Resumen de reglas de cortafuegos locales (Página 784).

Configuración de cortafuegos con VPN

Si el módulo de seguridad está en un grupo VPN, la casilla de verificación "Solo comunicación tunelada" está activada de forma predeterminada. Lo que significa que a través de la interfaz externa o la interfaz DMZ solo está permitida la transferencia de datos IPsec cifrada.

Si se desactiva esta casilla de verificación, se permitirá la comunicación tunelada y, adicionalmente, los tipos de comunicación seleccionados en las otras casillas de selección.

Reglas MAC y sentidos disponibles

Servicio	De interna a externa	De externa a interna	Significado
Permitir comunicación MAC	x	x	Se permite el tráfico MAC de interna a externa y viceversa.
Permitir protocolo ISO	x	x	Se permite el tráfico ISO de interna a externa y viceversa.
Permitir SiClock	x	x	Se permiten telegramas de hora SiClock de interna a externa y viceversa.
Permitir DCP	x	x	El tráfico DCP para adjudicar direcciones IP se permite de interna a externa y viceversa.

Tabla 8-40 Registro

Opción	Acción en caso de activación
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes MAC transferidos a través del túnel.
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes MAC entrantes que se rechazan.
Registro de paquetes salientes bloqueados	Se registran todos los paquetes MAC salientes que se rechazan.

Conjuntos de reglas IP específicos del usuario

Sinopsis

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Significado

En primer lugar se asignan uno o varios usuarios a los conjuntos de reglas IP específicos del usuario. A continuación se asignan los conjuntos de reglas IP específicos del usuario a uno o varios módulos de seguridad. De ese modo es posible permitir accesos específicos de usuario. Si, p. ej. están bloqueados de forma predeterminada todos los accesos a las redes situadas después de un módulo de seguridad, podrán habilitarse determinados dispositivos para un usuario mediante sus direcciones IP. El usuario tiene así el acceso permitido, mientras que este permanece bloqueado para otros usuarios.

Inicio de sesión del usuario a través de Internet

El usuario puede iniciar sesión a través de una página web de la interfaz externa o de la interfaz DMZ del módulo de seguridad. Si la autenticación es correcta, se activa el conjunto de reglas IP predefinido para ese usuario. La conexión con el módulo de seguridad se realiza vía HTTPS

utilizando la dirección IP del puerto enlazado y observando las reglas de enrutamiento aplicables:

Ejemplo:

Interfaz externa: 192.168.10.1

La página de inicio de sesión se abre mediante: https://192.168.10.1/

Pueden iniciar sesión usuarios con cualquier rol, siempre que el usuario esté asignado a un conjunto de reglas IP específico del usuario.

Los conjuntos de reglas IP específicos del usuario se utilizan localmente - Convenios

Rigen los mismos convenios que los descritos en el capítulo siguiente:
Conjuntos de reglas de cortafuegos globales - Convenios (Página 776)

Crear y asignar conjuntos de reglas IP específicos del usuario

Crear conjuntos de reglas IP específicos del usuario

1. En los ajustes de seguridad globales elija la entrada "Cortafuegos" > "Conjuntos de reglas IP específicos del usuario" > "Conjuntos de reglas IP".
   Resultado: debajo de la entrada seleccionada se muestran los conjuntos de reglas IP específicos del usuario que se han creado anteriormente.
2. Haga doble clic en la entrada "Agregar nuevo conjunto de reglas IP".
3. Introduzca los datos siguientes:
   - Nombre: designación unívoca del conjunto de reglas para todo el proyecto. El nombre aparece en la lista de reglas local del módulo de seguridad tras la asignación del conjunto de reglas.
   - Descripción (opcional): introduzca una descripción del conjunto de reglas IP específico del usuario.
4. Introduzca en la lista, por orden, las reglas de cortafuegos.
   Tenga en cuenta la descripción de los parámetros en los capítulos siguientes:
   Definir reglas de filtrado de paquetes IP (Página 785)

Asignar conjuntos de reglas IP específicos del usuario

1. Dentro del conjunto de reglas IP específico del usuario que se ha creado, cambie a la ficha "Usuarios".
2. En el área "Usuarios disponibles" seleccione los usuarios que desee asignar al conjunto de reglas IP específico del usuario.
3. Haga clic en el botón << para asignar los usuarios seleccionados al conjunto de reglas IP específico del usuario.

Nota

Asignación de conjuntos de reglas IP específicos del usuario

- Solo se puede asignar a un módulo de seguridad un conjunto de reglas IP específico del usuario por usuario.
- Con la asignación se activa de forma implícita para todos los usuarios asignados al conjunto de reglas IP el derecho para iniciar sesión en el módulo de seguridad.

Resultado

- El conjunto de reglas IP específico del usuario es utilizado como conjunto de reglas local por los módulos de seguridad y aparece automáticamente en la lista local de reglas de cortafuegos.
- El usuario puede iniciar sesión en el módulo de seguridad.

Sentidos de filtrado de paquetes IP SCALANCE S

Significado

Posibilidades de selección para los sentidos de comunicación "De" y "A" en las reglas IP del modo de cortafuegos avanzado.

Se dispone de los sentidos siguientes:

Posibilidades de selección / Rangos de valores		Módulo de seguridad		
De	A	S602	S61x	S623 / S627-2M
Interna	Externa	x	x	x
	Túnel	-	x	x
	Any	-	x	x
	DMZ	-	-	x
	Interna	x	x	x
Externa	Interna	x	x	x
	Any	-	-	x
	Túnel	-	-	x
	DMZ	-	-	x
Túnel	Interna	-	x	x
	Externa	-	x	x
	DMZ	-	-	x
Any	Interna	-	x	x
	Externa	-	-	x
	DMZ	-	-	x
DMZ	Interna	-	-	x
	Externa	-	-	x
	Any	-	-	x
	Túnel	-	-	x

x = posibilidad de configurar el sentido de la comunicación

- = no es posible configurar el sentido de la comunicación

Sentidos de filtrado de paquetes MAC SCALANCE S

Significado

Posibilidades de selección para los sentidos de comunicación "De" y "A" en las reglas MAC del modo de cortafuegos avanzado.

Se dispone de los sentidos siguientes:

Posibilidades de selección / Rangos de valores		Módulo de seguridad		
De	A	S602	S61x	S623 / S627-2M
Interna	Externa	x	x	x
	Túnel	-	x	x
	Any	-	x	x
Externa	Interna	x	x	x
	Any	-	-	x
	Túnel	-	-	x
Túnel	Interna	-	x	x
	Externa	-	x	x
Any	Interna	-	x	x
	Externa	-	-	x

x = posibilidad de configurar el sentido de la comunicación

- = no es posible configurar el sentido de la comunicación

Adaptar reglas estándar para servicios IP

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Requisitos

Esta función solo está disponible en el modo de cortafuegos avanzado.

Cómo se accede a esta función

1. Seleccione el módulo de seguridad que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Cortafuegos" > "Reglas estándar para servicios IP".

Significado de los ajustes avanzados

Parámetro	Significado en la activación
Utilizar opciones de estado ampliadas	Se limita el número de conexiones y estados del cortafuegos admisibles por unidad de tiempo. Si un dispositivo de red rebasa este límite, su dirección IP se incluye en la lista negra de IP del módulo de seguridad. La lista negra de IP del módulo de seguridad puede verse en el modo online.
Registrar todas las reglas activadas	Se registran los paquetes permitidos según las reglas estándar para los servicios IP.
Habilitar test ICMP para interfaces	Las solicitudes ping que entran en una interfaz del módulo de seguridad pueden transferirse a otras interfaces. De este modo, desde la red externa pueden realizarse por ejemplo solicitudes ping a la interfaz interna del módulo de seguridad.

Reglas de cortafuegos estándar para SCALANCE S

En la tabla siguiente se listan las reglas de cortafuegos estándar para los módulos SCALANCE S. En parte, las reglas de cortafuegos solo están activas cuando se utiliza el servicio correspondiente en el módulo de seguridad (p. ej. SNMP).

Servicio	Sentido	Interfaz X1 (roja)	Interfaz X2 (verde)	Interfaz X3 (amarilla) (solo para S623, S627-2M)	Interfaz de túnel (no para S602)
Re-enrutamiento por interfaz	saliente	-	x	-	-
HTTPS		x	x*	x	x*
ICMP	entrante	-	x	-	x
SNMP	entrante	x	x	x	x
Syslog	saliente	x	x	x	x
NTP	saliente	x	x	x	x
DNS	saliente	x	x	x	x
HTTP	saliente	x	-	x	-
VPN (IKE)		x	-	x	-
VPN (NAT Traversal)		x	-	x	-
Servidor BootP	entrante	-	x	x	-
Cliente BootP	saliente	-	x	x	-

x activado de forma predeterminada

- desactivado de forma predeterminada

* no puede adaptarse

Definir rutas

Significado

Indicar rutas para el direccionamiento de subredes a las que no puede accederse directamente desde el módulo de seguridad.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Enrutamiento".
3. Haga doble clic en la entrada "Agregar nuevo" de la tabla para agregar una ruta.
4. Introduzca los valores siguientes:

Parámetro	Función	Valor de ejemplo
ID de red	Las solicitudes a dispositivos de la subred con la ID de red indicada aquí y la máscara de subred indicada se envían a la subred por la dirección IP de router indicada. A partir de la ID de red y de la máscara de subred, el router reconoce si una dirección de destino está en la subred o fuera de ella. La ID de red indicada no puede estar en la misma subred que la dirección IP del módulo de seguridad.	192.168.11.0
Máscara de subred	La máscara de subred estructura la red. A partir de la ID de red y de la máscara de subred, el router reconoce si una dirección de destino está en la subred o fuera de ella.	255.255.255.0
Dirección IP del router	Dirección IP del router a través del que se accede a la subred. Para módulos SCALANCE S V4 también existe la posibilidad alternativa de introducir un FQDN. La dirección IP indicada para el router debe estar en la misma subred que la dirección IP del módulo de seguridad.	192.168.10.2
Activar re-enrutamiento (solo para módulos SCALANCE S V3/ V4)	Active esta casilla de verificación si los telegramas de la ruta introducida deben entrar y salir por la misma interfaz del módulo de seguridad (re-enrutamiento). El re-enrutamiento solo es soportado por la interfaz interna del módulo de seguridad.	

Módulo de seguridad como servidor DHCP

Resumen de servidor DHCP

Sinopsis

El módulo SCALANCE S puede utilizarse en la red interna y en la red DMZ como servidor DHCP (DHCP = Dynamic Host Configuration Protocol). Esto permite asignar automáticamente direcciones IP a los equipos conectados a la red interna.

Ambas interfaces pueden funcionar en modo servidor DHCP simultáneamente (solo SCALANCE S623/S627-2M).

Las direcciones IP se asignan en este caso dinámicamente desde una banda de direcciones definida por el usuario, o bien se asigna una dirección IP a un equipo concreto conforme a sus predeterminaciones. Si los dispositivos de la interfaz interna o DMZ deben obtener siempre la misma dirección IP para la configuración del cortafuegos, la asignación de direcciones solo puede ser estática a partir de la dirección MAC o de la ID de cliente.

Requisitos

Tiene que configurar los dispositivos en la red interna o DMZ de manera que obtengan la dirección IP de un servidor DHCP.

Dependiendo del modo de operación, el módulo de seguridad transmite a los dispositivos de la subred correspondiente una dirección IP del router predeterminado, o bien se tiene que comunicar una dirección IP de router a los dispositivos de la subred.

- Se transmite la dirección IP del router
  En los casos siguientes, el módulo de seguridad transmite a los dispositivos una dirección IP de router a través del protocolo DHCP:
  - El dispositivo está en la interfaz DMZ (solo SCALANCE S623/S627-2M)
    El módulo de seguridad transmite en este caso la dirección IP propia como dirección IP de router.
  - El dispositivo está en la interfaz interna y el módulo de seguridad está configurado para el modo de router
    El módulo de seguridad transmite en este caso la dirección IP propia como dirección IP de router.
  - El dispositivo está en la interfaz interna y el módulo de seguridad no está configurado para el modo de router, pero se ha indicado un router estándar en la configuración del módulo de seguridad
    El módulo de seguridad transmite en este caso la dirección IP del router estándar como dirección IP de router.
- No se transmite la dirección IP del router
  En estos casos tiene que introducir manualmente la dirección IP del router en los dispositivos:
  - El dispositivo está en la interfaz interna y el módulo de seguridad no está configurado para el modo de router. Adicionalmente, en la configuración del módulo de seguridad no se ha indicado ningún router estándar.

Consulte también

Configurar un servidor DHCP (Página 864)

Configurar un servidor DHCP

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Servidor DHCP".
3. Seleccione para qué interfaz desea realizar los ajustes de DHCP.
4. Realice la asignación de direcciones. Para la configuración tiene las dos posibilidades siguientes:
   – Asignación de direcciones estática
     A equipos con una dirección MAC o un Client-ID determinados se les asignan direcciones predeterminadas para cada caso. Introduzca para ello estos dispositivos en el campo de entrada "Asignación estática de direcciones" de la lista de direcciones.
   – Asignación dinámica de direcciones
     Los dispositivos cuya dirección MAC o ID de cliente no se haya indicado explícitamente reciben una dirección IP cualquiera de un rango de direcciones predeterminado. Para ello, active la casilla de verificación "Dynamic IP address range". A continuación active el área de direcciones en el área de entrada "Activar área dinámica de direcciones IP".

Nota

Asignación dinámica de direcciones - Comportamiento tras una interrupción de la alimentación eléctrica

Tenga en cuenta que las direcciones IP asignadas dinámicamente no se almacenan si se interrumpe la alimentación eléctrica. Tras restablecerse la alimentación eléctrica tiene que cuidar por tanto de que todos los dispositivos soliciten de nuevo una dirección IP.

Por esta razón, solo debería prever la asignación dinámica de direcciones para los siguientes dispositivos:

- dispositivos que se utilicen temporalmente en la subred (por ejemplo, equipos de mantenimiento).
- dispositivos que en caso de una nueva solicitud transmitan al servidor DHCP como "dirección preferida" una dirección IP anteriormente asignada (por ejemplo, estaciones de PC).

Para los dispositivos permanentes se debe dar preferencia a la asignación estática de direcciones indicando una ID de cliente o la dirección MAC.

Prueba de coherencia - reglas a considerar

Al realizar sus entradas debe tener en cuanta las reglas indicadas a continuación.

Prueba / Regla	Prueba realizada [1]	
	a nivel local	a nivel de proyecto/ módulo
Las direcciones IP asignadas en el campo de entrada "Asignación estática de direcciones" de la lista de direcciones no deben estar en el área de las direcciones IP dinámicas.		x
Las direcciones IP, direcciones MAC e IDs de cliente solo pueden aparecer una vez en la tabla "Asignación estática de direcciones" (con referencia al módulo de seguridad).		x
En el caso de las direcciones IP asignadas estáticamente, hay que indicar la dirección MAC o la ID de cliente (nombre del equipo).	x	
El Client-ID es una secuencia de como máximo 63 caracteres. Solo se permiten los caracteres siguientes: a-z, A-Z, 0-9 y - (guión). Nota En SIMATIC S7 es posible asignar a los dispositivos de la interfaz Ethernet una ID de cliente para la referencia a una dirección IP a través de DHCP. En PCs el procedimiento depende del sistema operativo utilizado; se recomienda utilizar la dirección MAC para la asignación.	x	
En el caso de las direcciones IP asignadas estáticamente, hay que indicar la dirección IP.	x	
Las siguientes direcciones IP no deben estar en el rango correspondiente al área de direcciones IP libres (direcciones IP dinámicas): • Todas las direcciones IP de router de la entrada "Enrutamiento" • Servidor Syslog • Router estándar • Dirección o direcciones del módulo de seguridad		x
DHCP es soportado por el módulo de seguridad en la interfaz con la subred interna y en la interfaz con la red DMZ. De este comportamiento del módulo de seguridad se derivan, además, los siguientes requisitos para las direcciones IP en el área de la asignación dinámica de direcciones: • Modo Bridge El área debe estar en la red definida por el módulo de seguridad. • Modo de enrutamiento El área debe estar en la subred interna definida por el módulo de seguridad. Nota La red DMZ representa siempre una subred separada. En caso de utilizar DHCP en la interfaz DMZ, hay que tener en cuenta que el área de direcciones IP libre (direcciones IP dinámicas) se encuentra dentro de la subred DMZ.		x

Prueba / Regla	Prueba realizada [1]	
	a nivel local	a nivel de proyecto/ módulo
El rango de direcciones IP libre se tiene que indicar por completo introduciendo la dirección inicial y la dirección final. La dirección final debe ser más alta que la inicial.	x	
Las direcciones IP introducidas en el campo de entrada "Asignación estática de direcciones" de la lista de direcciones tiene que estar en el área de direcciones de la subred interna o de la subred DMZ del módulo de seguridad.		x

Leyenda:

[1] Observe las explicaciones del capítulo:
Realizar prueba de coherencia (Página 758).

Configurar Proxy ARP

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Sinopsis

Proxy ARP permite a los routers responder a solicitudes ARP para hosts. Los hosts están en redes separadas por routers, pero utilizan la misma área de direcciones IP.

Si PC1 envía una solicitud ARP a PC2, recibe del módulo de seguridad situado en medio, y no del PC2, una respuesta ARP y la dirección de hardware de la interfaz (dirección MAC del puerto del módulo de seguridad) en la que se recibió la solicitud. El PC1 solicitante envía entonces sus datos al módulo de seguridad, que los transmite al PC2.

Cómo se accede a esta función

Esta función solo está disponible para la interfaz interna de un módulo de seguridad que forma parte de un grupo VPN y está en modo Bridge.

1. Seleccione el módulo de seguridad que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Proxy ARP".
3. Si el módulo de seguridad debe responder a una solicitud ARP de la LAN propia en lugar del interlocutor específico, introduzca la dirección IP correspondiente.

Túnel IPsec: crear y asignar grupos

Configurar nodos de red internos

Utilizar el modo de aprendizaje para aprender nodos internos

Buscar automáticamente dispositivos para la comunicación tunelada

Una gran ventaja para la configuración y el funcionamiento de la comunicación tunelada es que los módulos SCALANCE S en modo Bridge pueden localizar por sí mismos los dispositivos en la interfaz interna. Nuevos dispositivos son detectados por el módulo de seguridad durante el funcionamiento. Los dispositivos detectados se notifican a los módulos de seguridad pertenecientes al mismo grupo. Con esto está garantizado en todo momento el intercambio de datos en ambos sentidos dentro de los túneles de un grupo.

Dispositivos detectables

Se detectan los siguientes nodos:

- Nodos de red aptos para IP
  Se encuentran nodos de red aptos para IP si envían una respuesta ICMP al ICMP-Subnet-Broadcast.
  Nodos IP situados detrás de routers se pueden encontrar si los routers transmiten ICMP-Broadcasts.
- Nodos de red ISO
  Nodos de red que no sean aptos para IP, pero que a los que pueda accederse a través del protocolo ISO, también se pueden programar por aprendizaje.
  Condición para ello es que respondan a telegramas XID o TEST. TEST y XID (Exchange Identification) son protocolos auxiliares para el intercambio de informaciones en el nivel Layer 2. Enviando estos telegramas con una dirección Broadcast se pueden localizar estos nodos de red.
- Nodos PROFINET
  Con ayuda de DCP (Discovery and basic Configuration Protocol) se encuentran nodos PROFINET.

Los nodos de red que no cumplan estos requisitos se tienen que configurar manualmente.

Las subredes que están detrás de routers internos también deben configurarse manualmente.

Cómo se accede a la función

1. Seleccione el módulo.
2. En los ajustes de seguridad locales elija la entrada "VPN" > "Nodos".

Activación y desactivación del modo de aprendizaje

La función de aprendizaje está activada por defecto en la configuración para cada módulo de seguridad.

La programación por aprendizaje también se puede desactivar por completo para el SCALANCE S. Entonces se tienen que configurar manualmente todos los nodos de red internos que intervengan en la comunicación tunelada.

¿Cuándo es conveniente desactivar el modo de aprendizaje automático?

Los ajustes estándar para el módulo de seguridad parten de que las redes internas son siempre seguras; esto significa también que normalmente no se conectan a la red interna nodos de red que no sean dignos de confianza.

La desactivación del modo de aprendizaje puede ser conveniente si la red interna es estática, es decir, si no cambian el número ni las direcciones de los nodos internos.

Con la desconexión del modo de aprendizaje se suprime en la red interna la carga que los telegramas de programación por aprendizaje suponen para el medio y los nodos. También aumentan en cierta medida las prestaciones del módulo de seguridad, ya que no tiene que procesar los telegramas de programación por aprendizaje.

Observación: En el modo de aprendizaje se registran todos los nodos de la red interna. Los datos relativos a los recursos de la VPN se refieren solo a los nodos que se comuniquen en la red interna a través de VPN.

Nota

Si en la red interna se utilizan más de 128 nodos internos, se sobrepasa con esto el alcance admisible y se genera un estado operativo no permitido. Debido a la dinámica en el tráfico de la red ocurre entonces que los nodos internos ya programados por aprendizaje son reemplazados por nuevos nodos internos, hasta ahora desconocidos.

Consulte también

Configurar manualmente subredes internas (Página 869)

Configurar manualmente nodos de red IP

Significado

Como alternativa al modo de aprendizaje que se activa con la casilla de verificación "Permitir aprendizaje de nodos internos" y con el que el módulo de seguridad aprende dinámicamente los nodos de red internos, es posible introducir manualmente los nodos de red que deben aprenderse en la entrada "Nodos IP internos" y, con ello, habilitar la comunicación por túnel VPN. En este caso, la dirección MAC de un nodo de red es una entrada opcional.

Requisitos

- El módulo de seguridad está en modo Bridge.
- El módulo de seguridad forma parte de un grupo VPN.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "VPN" > "Nodos" > "Nodos IP internos".

Configurar manualmente nodos de red MAC

Significado

Como alternativa al modo de aprendizaje que se activa con la casilla de verificación "Permitir aprendizaje de nodos internos" y con el que el módulo de seguridad aprende dinámicamente los nodos de red internos, es posible introducir manualmente los nodos de red que deben aprenderse en la entrada "Nodos MAC internos" y, con ello, habilitar la comunicación por túnel VPN.

Requisitos

- El módulo de seguridad está en modo Bridge.
- El módulo de seguridad forma parte de un grupo VPN.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "VPN" > "Nodos" > "Nodos MAC internos".

Configurar manualmente subredes internas

Requisitos

- El módulo de seguridad forma parte de un grupo VPN.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "VPN" > "Nodos" > "Subredes internas".

Módulo de seguridad en modo Bridge - Entrada "Subredes internas"

Para habilitar manualmente subredes internas para la comunicación por túnel VPN hay que introducir los siguientes parámetros de dirección:

Parámetros	Función	Valor de ejemplo
ID de red	ID de red de la subred para la que debe habilitarse la comunicación por túnel VPN. A partir de la ID de red y de la máscara de subred, el router reconoce si una dirección de destino está en la subred o fuera de ella. No puede encontrarse en la misma subred que la dirección IP del módulo de seguridad.	192.168.11.0
Máscara de subred	La máscara de subred estructura la red. A partir de la ID de red y de la máscara de subred, el router reconoce si una dirección de destino está en la subred o fuera de ella.	255.255.255.0
Dirección IP del router	Dirección IP del router a través del que se accede a la subred que debe habilitarse. Debe estar en la misma subred que la dirección IP del módulo de seguridad.	192.168.10.2

Módulo de seguridad en modo de enrutamiento - Entrada "Subredes accesibles por túnel"

En el modo de enrutamiento siempre se establecen túneles para subredes completas. Para habilitar manualmente subredes internas para la comunicación por túnel VPN hay que introducir los siguientes parámetros de dirección:

Parámetro	Función	Valor de ejemplo
ID de red	ID de red de la subred para la que debe habilitarse la comunicación por túnel VPN. A partir de la ID de red y de la máscara de subred, el router reconoce si una dirección de destino está en la subred o fuera de ella. No puede encontrarse en la misma subred que la dirección IP del módulo de seguridad.	192.168.11.0
Máscara de subred	La máscara de subred estructura la red. A partir de la ID de red y de la máscara de subred, el router reconoce si una dirección de destino está en la subred o fuera de ella.	255.255.255.0
Comentario	Entrada de un comentario opcional adicional.	

Funciones online - Test / Diagnóstico y registro

Sinopsis de las diferentes interfaces - Entrada "Ajustes de interfaz"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Significado

Tabla 8-41 Diagnóstico online: Entrada "Ajustes de interfaz"

Funciones de sistema y estado	Significado
Interfaces	Tabla arriba: sinopsis de las interfaces del módulo de seguridad. Tabla abajo: indicaciones sobre la interfaz que se opera mediante PPPoE. • Estado: indica si se ha establecido o no una conexión con el Internet Service Provider (ISP). • Dirección IP actual: dirección IP actual de la interfaz • Pasarela: dirección IP de la pasarela • Servicio DNS din. primario: servicio primario para DNS dinámico • Servicio DNS din. secundario: servicio secundario para DNS dinámico • Código de error (numérico) información sobre el error en caso de que no se haya podido establecer una conexión con el ISP.
Redundancia de medios (solo SCALANCE S627-2M)	• Interfaz: interfaz conectada al anillo MRP • Protocolo: protocolo utilizado (MRP) • Puerto en anillo 1: nombre del primer puerto de módulo de medios de la interfaz que está conectado al anillo MRP • Puerto en anillo 2: nombre del segundo puerto de módulo de medios de la interfaz que está conectado al anillo MRP • Nombre de dominio: nombre del dominio MRP • Discrepancia: indicación de si hay un fallo en la configuración MRP • UUID de dominio: UUID del dominio MRP
Módulos de medios (solo SCALANCE S627-2M)	• Puerto: ID del o los puertos de módulos de medios • Nombre: nombre del módulo de medios • MLFB: referencia del módulo de medios • Revisión: versión del módulo de medios • Discrepancia: indicación de si hay un fallo en la configuración del módulo de medios

Sinopsis de los ajustes DDNS - Entrada "DNS din."

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Significado

Tabla 8-42 Diagnóstico online: "DNS dinámico"

Funciones de sistema y estado	Significado
Estado de cliente	Indica si se ha establecido una conexión con el servidor DNS dinámico o no.
Dirección IP actual	Dirección IP propia (es decir, registrada en el proveedor de DNS din.).
Dirección IP definida	dirección IP asignada vía PPPoE.
Hora actual	La hora actual.
Forzar la actualización	El módulo de seguridad determina la dirección IP actual de su acceso a Internet y envía una petición de actualización al o los servidores de actualización configurados. Así se publica la dirección IP actual en Internet. El estado se indica en los campos Servicio DNS din. primario y secundario. De esta manera se comprueba p. ej. si son correctos los datos configurados, como el nombre de usuario y la contraseña de la cuenta DNS din..
Deshabilitar suspensión	Anula la suspensión (bloqueo en el módulo de seguridad para actualizaciones de dirección IP en el proveedor de DNS din.), lo que significa que la contraseña de proveedor de DNS din. se ha corregido o se ha eliminado el fallo ocurrido.
Servicio DNS din. primario y secundario	
FQDN	Fully Qualified Domain Name registrado en el proveedor.
Dirección IP de servidor	Dirección IP del servidor de actualización utilizado.
Actualización satisfactoria	Actualización satisfactoria en el servicio de actualización DNS din.
Último intento de actualización	Último intento de actualización en el servicio de actualización DNS din.
Última actualización fallida	Último error de actualización en el servicio de actualización DNS din.
Código de error	Estado de error del último intento de actualización fallido en la actualización DNS din.

Significado de los avisos

Los avisos del último intento de actualización DDNS tienen el significado siguiente:

Aviso	Significado
Satisfactoria	
`DDNS_OK`	La solicitud de actualización ha sido satisfactoria.
Avisos de estado relativos a la conexión	
`DDNS_E_CON_UPD_SRV_RESOLV_ERR`	Nombre DNS del servidor de actualización desconocido, no se puede generar un FQDN a través del servidor DNS conocido.
`DDNS_E_CONUPD_SRV_UNREACHABLE`	Servidor de actualización no accesible ("timeout").
Avisos de estado relativos a la seguridad (error)	

Aviso	Significado
DDNS_E_CERT_SUBJECT_INVALID	El nombre común del sujeto en el certificado no coincide con el nombre de dominio del servidor de actualización o de su dirección IP.
DDNS_E_CERT_UNABLE_TO_GET_ISSUER_CERT	Certificado del emisor no encontrado. No se ha podido seguir la cadena de certificación hasta la CA raíz porque no se ha encontrado un certificado de emisor. La cadena de confianza está incompleta.
DDNS_E_CERT_SIGNATURE_INVALID	La firma de un certificado no ha podido leerse o no es válida.
DDNS_E_CERT_NO_TRUST	Un certificado de la cadena de confianza no es válido, es decir: • Todavía no es válido o ha caducado ya • Extensiones V3 no válidas • Una extensión V3 crítica no es soportada
DDNS_E_CERT_DEPTH_ZERO_SELFSIGNED_CERT	El servidor de actualización ha suministrado un certificado autofirmado y este certificado no está en la Certificate Store para certificados de CA raíz dignos de confianza.
DDNS_E_CERT_SELF_SIGNED_CERT_IN_CHAIN	Se ha establecido la cadena de certificación utilizando certificados no comprobados, pero no se ha encontrado ningún certificado de CA raíz apropiado en la Certificate Store para certificados dignos de confianza.
DDNS_E_CERT_CHAIN_TOO_LONG	La cadena de certificación excede la extensión de verificación máxima soportada.
DDNS_E_CERT_INVALID_CA	Un certificado CA no es válido, es decir, ha caducado, no es válido todavía o las extensiones V3 no son apropiadas para el fin de utilización previsto (por ejemplo, CA no en TRUE para certificados CA).
DDNS_E_CERT_KEYUSAGE_UNSUITED	El uso de clave de extensiones V3 o el uso de clave extendida puestos en un certificado de la cadena de confianza no es apto para la utilización del certificado.
DDNS_E_CERT_EXTENSION_UNSUPPORTED	Un certificado de la cadena de confianza utilizaba una extensión marcada como crítica que no es soportada.
Avisos de estado relativos al agente (error)	
DDNS_E_AGT_BAD_AGENT	• La solicitud de actualización no responde a la estructura requerida por cRSP, por ejemplo, faltan parámetros URL. • La solicitud de actualización ha sido enviada a una URL no permitida del servidor de actualización. • El string de actualización configurado contiene errores.

Visualización de la tabla ARP - Entrada "Tabla ARP"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Significado

Visualización de la tabla ARP del módulo de seguridad.

Tabla 8-43 Diagnóstico online: Ficha "Tabla ARP"

Funciones de sistema y estado	Significado
Tabla ARP	Indicación de las entradas estáticas (Proxy ARP) y dinámicas de la tabla ARP en el módulo de seguridad. En la ficha "Tipo de publicación" se indica si se trata de una entrada configurada estáticamente o de una entrada aprendida.

Usuarios registrados en página web - Entrada "Usuarios dados de alta"

Función específica del módulo

Esta función solo está disponible para SCALANCE S V3 o superior.

Significado

Visualización de los usuarios que han iniciado sesión en la página de Internet para conjuntos de reglas IP específicas de usuario.

Funciones de sistema y estado	Significado
Nombre de usuario	Nombre del usuario que ha iniciado sesión.
Dirección IP de origen	Dirección IP con la que el usuario ha iniciado sesión.
Tiempo de ejecución restante	Tiempo restante hasta que se cierre automáticamente la sesión del usuario.
Duración máxima de la sesión	Tiempo total configurado para la sesión.
Cerrar sesión	Si hace clic en el botón se cierra la sesión del usuario seleccionado.

Visualización de la lista negra de cortafuegos - Entrada "Lista negra de cortafuegos"

Significado

Visualización de las direcciones IP de dispositivos que han superado el número permitido de conexiones y estados de cortafuegos por tiempo. Estos dispositivos se registran en la lista negra de IP del cortafuegos.

El número de conexiones y estados de cortafuegos por tiempo solo está limitado si en la entrada "Reglas estándar para servicios IP" de los ajustes de seguridad locales está activada la casilla de verificación "Utilizar opciones de estado ampliadas".

Si se hace clic en el botón "Borrar todo" se eliminan las direcciones IP visualizadas de la lista negra de cortafuegos del módulo de seguridad. Además, dejarán de mostrarse las direcciones IP.

Ajustar la fecha y hora - Entrada "Fecha y hora"

Cómo se accede a esta función

1. Seleccione el módulo de seguridad cuya hora y fecha deban comprobarse o ajustarse.
2. En el menú contextual, elija el comando de menú "Online y diagnóstico".
3. En el diagnóstico online elija la entrada "Funciones" > "Fecha y hora".

Hora local en el módulo de seguridad

En esta área se leen y ajustan la hora y la fecha del módulo de seguridad. Si se hace clic en el botón "Aplicar" se asignan al módulo de seguridad la hora y fecha que están introducidas actualmente en los campos de entrada "Fecha" y "Hora".

Hora local en el PC

En esta área se muestran la hora y fecha actuales del PC en el que está instalado STEP 7. Si se hace clic en el botón "Aplicar al módulo" se asignan al módulo de seguridad la hora y fecha actuales del PC.

Funciones de carga

Cargar una configuración

Qué hay que tener en cuenta antes de cargar una configuración

- Dirección IP utilizada para cargar configuraciones
  Las configuraciones se cargan siempre en el módulo de seguridad mediante la dirección IP que se ha configurado para el puerto interno o externo en los ajustes de seguridad locales. Dicha dirección IP debe coincidir con la del módulo de seguridad.
- Conexiones
  En principio es posible cargar los datos de configuración tanto desde la interfaz externa como desde la interfaz interna del módulo de seguridad.
  Configure preferentemente los módulos de un grupo VPN a través de la red externa común de esos módulos (puerto 1 del dispositivo). Si el equipo de configuración se encuentra en una red interna, se tienen que habilitar explícitamente en el cortafuegos de ese módulo SCALANCE S las direcciones IP de los demás módulos del grupo, cargando luego el módulo en cuestión en primer lugar.

  Nota
  Seleccionar adaptador de red
  Si se utilizan varios adaptadores de red en su PC/PG, seleccione primero en la lista desplegable "Interfaz PG/PC" del cuadro de diálogo "Carga avanzada" el adaptador de red a través del que desea acceder al módulo SCALANCE S.

- Estado operativo
  Se pueden cargar configuraciones durante el funcionamiento normal de los equipos SCALANCE S. Para que sean eficaces los cambios realizados en una configuración debe rearrancarse el módulo SCALANCE S.

  Nota
  Particularidades
  - Mientras un módulo no haya ajustado todavía ningún parámetro IP (es decir, antes de la primera configuración), no debe haber ningún router entre el módulo y el ordenador de configuración.
  - Si enchufa un PC de la interfaz interna a la externa del SCALANCE S, los accesos de este PC al SCALANCE S se bloquearán durante un plazo de 20 minutos aproximadamente.

Transferencia de firmware

Qué hay que tener en cuenta antes de transferir un firmware nuevo

Para la transferencia de un nuevo firmware a un módulo de seguridad se han de cumplir los siguientes requisitos:

- Se poseen los derechos necesarios para transferir firmware, consulte el capítulo: Auto-Hotspot.
- El módulo de seguridad está configurado con una dirección IP.

La transferencia es segura

La transferencia del firmware tiene lugar a través de una conexión segura, por lo que se puede realizar también desde la red no protegida.

El propio firmware está firmado y codificado. Con esto se garantiza que solo se pueda cargar firmware auténtico en el módulo SCALANCE S.

La transferencia se puede realizar durante el funcionamiento normal

El firmware se puede transferir durante el funcionamiento normal de un módulo SCALANCE S. Un nuevo firmware cargado solo estará activo tras el reinicio del módulo SCALANCE S. Si la trasferencia ha sufrido una perturbación y se ha cancelado, el módulo vuelve a arrancar con la versión de firmware antigua.

Seguridad para CPs S7-300/S7-400/PC

Crear cortafuegos

Reglas de cortafuegos locales para CPs S7-300/S7-400/PC

Resumen de CPs S7-300/S7-400/PC

Activar reglas de filtrado de paquetes

Si en los ajustes de seguridad locales se activa la función de seguridad para los CPs, inicialmente estarán permitidos todos los accesos a y a través del CP. Para activar reglas concretas de filtrado de paquetes, haga clic en la casilla de verificación "Activar cortafuegos". A continuación autorice los servicios deseados. Las reglas de cortafuegos creadas

automáticamente debido a una configuración de conexión tienen preferencia sobre las reglas ajustadas manualmente.

Nota

Ajustes de cortafuegos detallados en el modo de cortafuegos avanzado

En el modo de cortafuegos avanzado, las reglas de cortafuegos pueden limitarse a dispositivos determinados. Para cambiar al modo de cortafuegos avanzado, active la casilla de verificación "Activar cortafuegos en modo avanzado".

Configuración de cortafuegos con VPN

Si el módulo de seguridad se agrega a un grupo VPN, el cortafuegos se activa por defecto. Además, se activa la casilla de verificación "Solo comunicación tunelada". Lo que significa que a través de la interfaz externa solo está permitida la transferencia de datos IPsec cifrada. El tráfico de datos a externa se bloquea.

Si se desactiva esta casilla de verificación, se permitirá la comunicación tunelada y, adicionalmente, los tipos de comunicación seleccionados en las otras casillas de selección.

Actualizar reglas de conexión

Al cambiar la configuración de conexión de CPs cambian también las reglas de cortafuegos relativas a la conexión. Para visualizar las reglas de cortafuegos modificadas hay que volver a pulsar el botón "Actualizar reglas de conexión". Las reglas de cortafuegos modificadas se mostrarán en el modo de cortafuegos avanzado.

Configurar cortafuegos con reglas de cortafuegos predefinidas - CP x43-1 Advanced

Configurar cortafuegos con reglas IP predefinidas - CP x43-1 Advanced

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Cortafuegos" > "Predefined IP rules".

Tabla 8-44 Servicios y sentidos disponibles

Servicio	De estación/ interna a externa	De externa a interna	De externa a estación	Puertos permitidos	Significado
Permitir comunicación IP	x	x	x	todos	Se permite el tráfico IP para los sentidos de comunicación seleccionados.
Permitir protocolo S7	x	x	x	Puerto TCP 102	Se permite la comunicación de los dispositivos de red a través del protocolo S7.

Servicio	De estación/ interna a externa	De externa a interna	De externa a estación	Puertos permitidos	Significado
Permitir FTP/ FTPS (modo explícito)	x	x	x	Puerto TCP 20 Puerto TCP 21	Para la administración de archivos y el acceso a ellos entre servidor y cliente.
Permitir HTTP	x	x	x	Puerto TCP 80	Para la comunicación con un servidor web.
Permitir HTTPS	x	x	x	Puerto TCP 443	Para la comunicación segura con un servidor web, p. ej. diagnóstico web.
Permitir DNS	x	x	-	Puerto TCP 53 Puerto UDP 53	Se permite la conexión con un servidor DNS.
Permitir SNMP	x	x	x	Puerto TCP 161/162 Puerto UDP 161/162	Para vigilar dispositivos de red con capacidad SNMP.
Permitir SMTP	x	x	-	Puerto TCP 25	Para enviar correos electrónicos a través de un servidor SMTP.
Permitir NTP	x	x	-	Puerto UDP 123	Para la sincronización de la hora.

Tabla 8-45 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Ajustes de registro IP		Acción	De	A
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes IP transferidos a través del túnel.	Allow	Estación	Túnel
		Allow	Túnel	Estación
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes IP entrantes que se rechazan.	Drop	Externa	Estación

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IP predefinidas" o "Reglas MAC predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Configurar un cortafuegos con reglas MAC predefinidas - CP x43-1 Advanced

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Seleccione la entrada "Cortafuegos" > "Reglas MAC predefinidas".

Tabla 8-46 Servicios y sentidos disponibles

Servicio	De estación a externa	De externa a estación	Significado
Permitir comunicación MAC	x	x	Se permite el tráfico MAC de estación a externa y viceversa.
Permitir protocolo ISO	x	x	Se permite el tráfico ISO de estación a externa y viceversa.

Tabla 8-47 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Configuración del registro MAC		Acción	De	A
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes MAC transferidos a través del túnel.	Allow	Estación	Túnel
		Allow	Túnel	Estación
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes MAC entrantes que se rechazan.	Drop	Externa	Estación
Registro de paquetes salientes bloqueados	Se registran todos los paquetes MAC salientes que se rechazan.	Drop	Estación	Externa

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IP predefinidas" o "Reglas MAC predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Configurar un cortafuegos con reglas predefinidas - CP1628

Configurar un cortafuegos con reglas IP predefinidas - CP1628

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Seleccione la entrada "Seguridad" > "Cortafuegos" > "Predefined IP rules".

Tabla 8-48 Servicios y sentidos disponibles

Servicio	De externa a estación	Puertos permitidos	Significado
Permitir comunicación IP	x	todos	Se permite el tráfico IP de externa a estación.
Permitir protocolo S7	x	Puerto TCP 102	Se permite la comunicación de los dispositivos de red a través del protocolo S7.
Permitir FTP/FTPS (modo explícito)	x	Puerto TCP 20 Puerto TCP 21	Para la administración de archivos y el acceso a ellos entre servidor y cliente.
Permitir HTTP	x	Puerto TCP 80	Para la comunicación con un servidor web.
Permitir HTTPS	x	Puerto TCP 443	Para la comunicación segura con un servidor web, p. ej. diagnóstico web.
Permitir DNS	x	Puerto TCP 53 Puerto UDP 53	Se permite la conexión con un servidor DNS.
Permitir SNMP	x	Puerto TCP 161/162 Puerto UDP 161/162	Para vigilar dispositivos de red con capacidad SNMP.
Permitir SMTP	x	Puerto TCP 25	Para enviar correos electrónicos a través de un servidor SMTP.
Permitir NTP	x	Puerto UDP 123	Para la sincronización de la hora.

Tabla 8-49 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Ajustes de registro IP		Acción	De	A
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes IP transferidos a través del túnel.	Allow	Estación	Túnel
		Allow	Túnel	Estación
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes IP entrantes que se rechazan.	Drop	Externa	Estación

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IP predefinidas" o "Reglas MAC predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Configurar un cortafuegos con reglas MAC predefinidas - CP1628

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Seleccione la entrada "Seguridad" > "Cortafuegos" > "Reglas MAC".

Tabla 8-50 Servicios y sentidos disponibles

Servicio	De estación a externa	De externa a estación	Significado
Permitir comunicación en nivel MAC	x	x	Se permite el tráfico MAC de externa a la estación y viceversa.
Permitir comunicación ISO	x	x	Se permite el tráfico ISO de externa a la estación y viceversa.
Permitir SiClock	x	x	Se permiten telegramas de hora SiClock de externa a la estación y viceversa.

Tabla 8-51 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Configuración del registro MAC		Acción	De	A
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes MAC transferidos a través del túnel.	Allow	Estación	Túnel
		Allow	Túnel	Estación
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes MAC entrantes que se rechazan.	Drop	Externa	Estación
Registro de paquetes salientes bloqueados	Se registran todos los paquetes MAC salientes que se rechazan.	Drop	Estación	Externa

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IP predefinidas" o "Reglas MAC predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Sentidos de filtrado de paquetes IP CPs S7-300/S7-400/PC

Significado

Posibilidades de selección para los sentidos de comunicación "De" y "A" en las reglas IP del modo de cortafuegos avanzado.

Se dispone de los sentidos siguientes:

Posibilidades de selección / Rangos de valores		Módulo de seguridad		Significado
De	A	CP x43-1 Adv.	CP 1628	
Interna	Estación	x	-	Acceso de la red interna a la estación.
	Any	x	-	Acceso de la red interna a la externa, al interlocutor de túnel VPN y a la estación.
Externa	Estación	x	x	Acceso de la red externa a la estación.
	Any	x	-	Acceso de la red externa a la interna y a la estación.
Estación	Interna	x	-	Acceso desde la estación a la red interna.
	Externa	x	x	Acceso desde la estación a la red externa.
	Túnel	x	x	Acceso desde la estación al interlocutor de túnel VPN.
Túnel	Estación	x	x	Acceso a la estación a través del interlocutor de túnel VPN.
	Any	x	-	Acceso del interlocutor de túnel VPN a la red interna y a la estación.
Any	Externa	x	-	Acceso desde la red interna y la estación a la red externa.

Sentidos de filtrado de paquetes MAC CPs S7-300/S7-400/PC

Referencia

Posibilidades de selección para los sentidos de comunicación "De" y "A" en las reglas MAC del modo de cortafuegos avanzado.

Se dispone de los sentidos siguientes:

Posibilidades de selección / Rangos de valores		Módulo de seguridad		Significado
De	A	CP x43-1 Adv.	CP 1628	
Externa	Estación	x	x	Acceso de la red externa a la estación.
Estación	Externa	x	x	Acceso desde la estación a la red externa.
	Túnel	x	x	Acceso desde la estación al interlocutor de túnel VPN.
Túnel	Estación	x	x	Acceso a la estación a través del interlocutor de túnel VPN.

Configurar una lista de acceso

Función específica del módulo

Esta función no está disponible para CP 1628.

Significado

Las listas de acceso IP permiten definir la protección de acceso para determinadas direcciones IP. Las entradas de la lista ya creadas en los ajustes locales del CP se muestran con sus derechos correspondientes en la entrada "Cortafuegos" > "Reglas IP" (modo de cortafuegos avanzado).

Nota

Cambio de respuesta tras activar la seguridad

- Tras activar la seguridad para un CP, la protección de acceso solo afecta a la interfaz externa. Para que la protección de acceso también afecte a la interfaz interna, configure las reglas de cortafuegos correspondientes en el modo de cortafuegos avanzado.
- El CP también responde a solicitudes ARP de direcciones IP no habilitadas (capa 2).
- Si la lista de acceso IP de un CP no contiene entradas y está activada la seguridad para el CP, se activa el cortafuegos y ya no es posible acceder al CP desde externa. Para que el CP sea accesible, configure las reglas de cortafuegos correspondientes en el modo de cortafuegos avanzado.

Efecto de las entradas de la lista de acceso IP al activar la seguridad

Si en los ajustes locales de un CP se activa la seguridad, en el modo de cortafuegos avanzado se crean las reglas correspondientes. Si en la lista de acceso se ha indicado una dirección IP, para la misma se crea una regla de cortafuegos "Allow" > "Externa" > "Estación". Como dirección IP de origen se toma la dirección IP de la lista de acceso IP. Las direcciones IP de un área de direcciones IP definida también se convierten a las reglas de cortafuegos correspondientes.

Requisitos de edición

Antes de que las reglas de cortafuegos creadas puedan editarse debe cumplirse el requisito siguiente:

- para la edición mediante STEP 7: derecho de configuración "Configurar seguridad".
- para la edición mediante un servidor web: derecho de módulos "Web: Ampliar lista de control de acceso IP".

Los requisitos para editar las listas de acceso IP fuera de los ajustes de seguridad locales se describen en los capítulos de los CPs correspondientes.

Reglas de cortafuegos automáticas referidas a conexiones

Significado

Para conexiones configuradas mediante CPs, STEP 7 crea automáticamente reglas de cortafuegos que autorizan la comunicación con el interlocutor correspondiente del CP en el sentido prescrito (CP activo/pasivo). Se tienen en cuenta los sentidos de establecimiento de las conexiones. Para visualizar dichas reglas de cortafuegos debe pulsarse el botón "Actualizar reglas de conexión" estando activado el modo de cortafuegos avanzado. Las reglas de cortafuegos se mostrarán seguidamente en el modo de cortafuegos avanzado.

Nota

Habilitar manualmente conexiones UDP Multicast y UDP Broadcast

Para las conexiones UDP Multicast y UDP Broadcast no se crean reglas de cortafuegos automáticas. Para habilitar las conexiones, agregue las correspondientes reglas de cortafuegos manualmente en el modo de cortafuegos avanzado.

Dependiendo de cómo esté configurado el establecimiento de conexión, se crean las siguientes reglas de cortafuegos de nivel 3. Si el módulo de seguridad se encuentra en un grupo VPN, el sentido "Externa" cambia a "Túnel". Esto solo es válido para CPs que soportan VPN.

En la columna "Dirección IP de origen" y "Dirección IP de destino" de estas reglas del cortafuegos se introduce en cada caso la dirección IP del interlocutor.

CP->externa	Acción	De	A
Activo	Allow	Estación	Externa
	Drop	Externa	Estación
Pasivo	Drop	Estación	Externa
	Allow	Externa	Estación
Activo y pasivo	Allow	Externa	Estación
	Allow	Estación	Externa

CP->interna	Acción	De	A
Activo	Allow	Estación	Interna
	Drop	Interna	Estación
Pasivo	Drop	Estación	Interna
	Allow	Interna	Estación
Activo y pasivo	Allow	Interna	Estación
	Allow	Estación	Interna

Para las conexiones de nivel 2 se crean reglas "Allow" para ambos sentidos. Si el módulo de seguridad se encuentra en un grupo VPN, el sentido "Externa" cambia a "Túnel". Esto solo es válido para CPs que soportan VPN.

En la columna "Dirección MAC de origen" y "Dirección MAC de destino" de estas reglas del cortafuegos se introduce en cada caso la dirección MAC del interlocutor.

CP->externa	Acción	De	A
activo, pasivo, activo y pasivo	Allow	Estación	Externa
	Allow	Externa	Estación

Nota

Cambiar la configuración de la conexión

Al cambiar la configuración de conexión de CPs cambian también las reglas de cortafuegos relativas a la conexión. Para visualizar las reglas de cortafuegos modificadas hay que volver a pulsar el botón "Actualizar reglas de conexión".

Convenios para reglas de cortafuegos creadas automáticamente

- Prioridad
  Las reglas tienen la máxima prioridad, por lo que en la lista de reglas local se incorporan en la parte superior.
- Borrar reglas
  Las reglas no se pueden borrar. El registro puede activarse y pueden asignarse servicios. Además es posible insertar un ancho de banda y un comentario.
- Cambiar una acción
  Si se cambia la acción "Allow" a "Drop" o viceversa, estas se sobrescribirán en la siguiente comparación del sistema. Si los cambios realizados deben conservarse, elija como acción "Allow*" o "Drop*". En ese caso solo se compara la dirección IP y la acción y el sentido se mantienen de la forma ajustada. Los ajustes de registro, servicio, ancho de banda y comentario se conservan cuando se produce una nueva comparación de sistema aunque no se cambie la acción a "Allow*" o "Drop*". Si se borra la conexión configurada, las reglas correspondientes se eliminan de la lista.

Módulo de seguridad en grupo VPN

La casilla de verificación "Solo comunicación tunelada" está activada de forma predeterminada. Si se desactiva la casilla de verificación, además de la comunicación

tunelada entre interlocutores tunelados existe la posibilidad de establecer comunicación con otros dispositivos de red con los que no existen túneles.

- La comunicación se produce sin túnel si la dirección del interlocutor pertenece a una estación conocida en STEP 7 con la que no hay configurado ningún túnel VPN.
- La comunicación transcurre a través del túnel VPN si la dirección del interlocutor es un punto final VPN.
- Si no es posible asignar unívocamente si una conexión debe transcurrir por dentro o por fuera del túnel VPN, la conexión se asigna al túnel VPN y se muestra la indicación correspondiente. La asignación puede adaptarse en el modo de cortafuegos avanzado, p. ej. cambiando el sentido "De" de "Túnel" a "Externa". Para que esta adaptación no se sobrescriba de nuevo al realizarse otra comparación del sistema, debe seleccionarse la acción "Allow*" o "Drop*".

Nota

En caso de ser necesario garantizar que solo sea posible la comunicación a través del túnel, cree las reglas de cortafuegos correspondientes en el modo de cortafuegos avanzado, p. ej. para dispositivos internos o direcciones NDIS.

Para permitir exclusivamente la comunicación tunelada para un CP, inserte una regla "Drop" > "Any" > "Externa". Para CP 1628 debe insertarse una regla "Drop" > "Estación" > "Externa". Además, deben eliminarse las reglas de cortafuegos existentes que permitan una comunicación sin túnel.

Activar el servidor web en el CP x43-1 Advanced

Función específica del módulo

Esta función solo está disponible para el CP x43-1 Advanced.

Significado

Tras activar el servidor web se obtiene acceso a las páginas web del módulo. En los ajustes de seguridad locales es posible limitar el acceso a dichas páginas web al protocolo HTTPS. El acceso se controla mediante la casilla de verificación "Permitir el acceso solo vía HTTPS". Además, es necesario configurar el cortafuegos adecuadamente.

Túnel IPsec: crear y asignar grupos

Configurar nodos de red internos - Entrada "Nodos"

Permitir el acceso a CPs S7-300/S7-400 para interlocutores VPN

Posibilidades de selección

Seleccione si los interlocutores VPN en modo de enrutamiento (SCALANCE S / M) deben tener acceso al CP y/o a la subred interna del CP.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Nodos".
3. Configure el acceso para los interlocutores VPN del CP en el modo de enrutamiento (SCALANCE S / M).
   - Permitir conexión al CP (interfaz Gigabit)
   - Permitir conexión a subred interna (subred PROFINET)

Configurar manualmente nodos NDIS accesibles por túnel en CPs PC

Configurar nodos NDIS accesibles por túnel

Los nodos internos se aprenden y se asignan directamente a las rutas. Ello afecta a las direcciones IP NDIS del PC Windows.

Proceda del siguiente modo

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Nodos" > "Nodos NDIS accesibles por túnel".
3. Introduzca la dirección IP NDIS.

Funciones online - Test / Diagnóstico y registro

Reglas de cortafuegos actualizadas - Entrada "Reglas del cortafuegos actualizadas de forma dinámica"

Significado

Visualización de las direcciones IP o áreas de direcciones IP habilitadas dinámicamente vía HTTP o HTTPS o cargadas posteriormente por un usuario. Para las direcciones IP habilitadas

se muestran los derechos de acceso al CP S7 que tienen asignadas. En esta ficha las direcciones IP se pueden actualizar mediante los eventos siguientes:

- Ampliación/Modificación de la lista IP Access Control
- Actualización de las reglas de cortafuegos
- Ampliaciones dinámicas registradas por el CP en el tiempo de ejecución, p.ej. dispositivos PROFINET IO

Como aquí solo se muestran las reglas de cortafuegos actualizadas dinámicamente, para considerar íntegramente el estado actual de cortafuegos del módulo hay que incluir también las reglas de cortafuegos que se han configurado offline y se han cargado en la estación.

Seguridad para CPs S7-1500

Crear cortafuegos

Reglas de cortafuegos locales para CPs S7-1500

Resumen de reglas de cortafuegos locales

Activar reglas de filtrado de paquetes

Si en los ajustes de seguridad locales se activa la función de seguridad para los CPs, inicialmente estarán permitidos todos los accesos a y a través del CP. Para activar reglas concretas de filtrado de paquetes, haga clic en la casilla de verificación "Activar cortafuegos". A continuación autorice los servicios deseados. Las reglas de cortafuegos creadas automáticamente debido a una configuración de conexión tienen preferencia sobre las reglas ajustadas manualmente.

Nota

Ajustes de cortafuegos detallados en el modo de cortafuegos avanzado

En el modo de cortafuegos avanzado, las reglas de cortafuegos pueden limitarse a dispositivos determinados. Para cambiar al modo de cortafuegos avanzado, active la casilla de verificación "Activar cortafuegos en modo avanzado".

Actualizar reglas de conexión

Al cambiar la configuración de conexión de CPs cambian también las reglas de cortafuegos relativas a la conexión. Para visualizar las reglas de cortafuegos modificadas hay que volver a pulsar el botón "Actualizar reglas de conexión". Las reglas de cortafuegos modificadas se mostrarán en el modo de cortafuegos avanzado.

Configurar un cortafuegos con reglas de cortafuegos predefinidas - CP 1543-1

Configurar un cortafuegos con reglas IP predefinidas - CP 1543-1

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. En los ajustes de seguridad locales elija la entrada "Cortafuegos" > "Predefined IP rules".

Tabla 8-52 Servicios y sentidos disponibles

Servicio	De externa a estación	Puertos permitidos	Significado
Permitir comunicación IP	x	todos	Se permite el tráfico IP de externa a estación.
Permitir protocolo S7	x	Puerto TCP 102	Se permite la comunicación de los dispositivos de red a través del protocolo S7.
Permitir FTP/ FTPS (modo explícito)	x	Puerto TCP 20 Puerto TCP 21	Para la administración de archivos y el acceso a ellos entre servidor y cliente.
Permitir HTTP	x	Puerto TCP 80	Para la comunicación con un servidor web.
Permitir HTTPS	x	Puerto TCP 443	Para la comunicación segura con un servidor web, p. ej. diagnóstico web.
Permitir SNMP	x	Puerto TCP 161/162 Puerto UDP 161/162	Para vigilar dispositivos de red con capacidad SNMP.
Permitir SMTP	x	Puerto TCP 25	Para enviar correos electrónicos a través de un servidor SMTP.
Permitir diagnostico de seguridad	x	Puerto TCP 8448	Permite o no permite el diagnóstico de seguridad.

Tabla 8-53 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Ajustes de registro IP		Acción	De	A
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes IP transferidos a través del túnel.	Allow	Estación	Túnel
		Allow	Túnel	Estación
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes IP entrantes que se rechazan.	Drop	Externa	Estación

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IP predefinidas" o "Reglas MAC predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Configurar un cortafuegos con reglas IPv6 predefinidas - CP 1543-1

Significado

Las reglas IPv6 predefinidas ofrecen la posibilidad de configurar el cortafuegos en relación a servicios en los que se utiliza IPv6. Al habilitar una regla IPv6 predefinida en los ajustes de seguridad locales del CP 1543-1 V1.1 se habilitan también en el cortafuegos los servicios ICMPv6 definidos por el sistema que se muestran en la ficha "ICMP" de los ajustes de seguridad globales, en "Cortafuegos" > "Servicios" > "Definir servicios para reglas IP". El cortafuegos del CP 1543-1 V1.0 deja pasar paquetes ICMPv6 aunque no esté habilitada una regla IPv6 predefinida.

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Seleccione en los ajustes de seguridad locales la entrada "Cortafuegos" > "Reglas IPv6 predefinidas".

Tabla 8-54 Servicios y sentidos disponibles

Servicio	De externa a estación	Puertos permitidos	Significado
Permitir tráfico IP	x	todos	Se permite el tráfico IP de externa a estación.
Permitir protocolo S7	x	Puerto TCP 102	Se permite la comunicación de los dispositivos de red a través del protocolo S7.
Permitir FTP/ FTPS (modo explícito)	x	Puerto TCP 20 Puerto TCP 21	Para la administración de archivos y el acceso a ellos entre servidor y cliente.
Permitir HTTP	x	Puerto TCP 80	Para la comunicación con un servidor web.
Permitir HTTPS	x	Puerto TCP 443	Para la comunicación segura con un servidor web, p. ej. diagnóstico web.
Permitir SNMP	x	Puerto TCP 161/162 Puerto UDP 161/162	Para vigilar dispositivos de red con capacidad SNMP.
Permitir SMTP	x	Puerto TCP 25	Para enviar correos electrónicos a través de un servidor SMTP.
Permitir diagnostico de seguridad	x	Puerto TCP 8448	Permite o no permite el diagnóstico de seguridad vía IPv6.

Tabla 8-55 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Ajustes de registro IP		Acción	De	A
Registro de paquetes tunelados	Solo está activado si el módulo de seguridad forma parte de un grupo VPN. Se registran todos los paquetes IP transferidos a través del túnel.	Allow	Estación	Túnel
		Allow	Túnel	Estación
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes IP entrantes que se rechazan.	Drop	Externa	Estación

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IPv6 predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Configurar un cortafuegos con reglas MAC predefinidas - CP 1543-1

Cómo se accede a esta función

1. Seleccione el módulo que desea editar.
2. Seleccione la entrada "Cortafuegos" > "Reglas MAC predefinidas".

Tabla 8-56 Servicios y sentidos disponibles

Servicio	De estación a externa	De externa a estación	Puertos permitidos	Significado
Permitir comunicación MAC	x	x	-	Se permite el tráfico MAC de externa a la estación y viceversa.
Permitir protocolo ISO	x	x	-	Se permite el tráfico ISO de externa a la estación y viceversa.
Permitir SiCLOCK	x	x	-	Se permite el tráfico SiCLOCK de externa a la estación y viceversa.

Tabla 8-57 Registro

Opción	Acción en caso de activación	Regla de cortafuegos pertinente		
Configuración del registro MAC		Acción	De	A
Registro de paquetes entrantes bloqueados	Se registran todos los paquetes MAC entrantes que se rechazan.	Drop	Externa	Estación
Registro de paquetes salientes bloqueados	Se registran todos los paquetes MAC salientes que se rechazan.	Drop	Estación	Externa

Nota

Relación entre los ajustes de registro en modo normal y las reglas del cortafuegos

Los ajustes de registro realizados en "Reglas IP predefinidas" o "Reglas MAC predefinidas" no afectan a las reglas del cortafuegos creadas automáticamente mediante una configuración de conexión. Así, por ejemplo, no es posible registrar telegramas tunelados de una conexión configurada. En el modo de cortafuegos avanzado existe la posibilidad de ampliar el registro a las reglas de cortafuegos creadas automáticamente para conexiones.

Sentidos de filtrado de paquetes IP para CPs S7-1500

Significado

Posibilidades de selección para los sentidos de comunicación "De" y "A" en las reglas IP del modo de cortafuegos avanzado.

Se dispone de los sentidos siguientes:

Posibilidades de selección / Rangos de valores		Significado
De	A	
Externa	Estación	Acceso de la red externa a la estación.
Estación	Externa	Acceso desde la estación a la red externa.
	Túnel	Acceso desde la estación al interlocutor de túnel VPN.*
Túnel	Estación	Acceso de interlocutores de túnel VPN a la estación.*

* No para CP 1543-1 V1.0.

Sentidos de filtrado de paquetes MAC para CPs S7-1500

Significado

Posibilidades de selección para los sentidos de comunicación "De" y "A" en las reglas MAC del modo de cortafuegos avanzado.

Se dispone de los sentidos siguientes:

Posibilidades de selección / Rangos de valores		Significado
De	A	
Externa	Estación	Acceso de la red externa a la estación.
Estación	Externa	Acceso desde la estación a la red externa.
	Túnel	Acceso desde la estación al interlocutor de túnel VPN.*
Túnel	Estación	Acceso de interlocutores de túnel VPN a la estación.*

* No para CP 1543-1 V1.0.

Reglas de cortafuegos automáticas referidas a conexiones

Significado

Para conexiones configuradas mediante CPs, STEP 7 crea automáticamente reglas de cortafuegos que autorizan la comunicación con el interlocutor correspondiente del CP en el sentido prescrito (CP activo/pasivo). Se tienen en cuenta los sentidos de establecimiento de las conexiones. Para visualizar dichas reglas de cortafuegos debe pulsarse el botón "Actualizar reglas de conexión" estando activado el modo de cortafuegos avanzado. Las reglas de cortafuegos se mostrarán seguidamente en el modo de cortafuegos avanzado.

Nota

Habilitar manualmente conexiones UDP Multicast y UDP Broadcast

Para las conexiones UDP Multicast y UDP Broadcast no se crean reglas de cortafuegos automáticas. Para habilitar las conexiones, agregue las correspondientes reglas de cortafuegos manualmente en el modo de cortafuegos avanzado.

Dependiendo de cómo esté configurado el establecimiento de conexión, se crean las siguientes reglas de cortafuegos de nivel 3. Si el CP 1543-1 V1.1 se encuentra en un grupo VPN, el sentido "Externa" cambia a "Túnel".

En la columna "Dirección IP de origen" y "Dirección IP de destino" de estas reglas del cortafuegos se introduce en cada caso la dirección IP del interlocutor.

CP->externa	Acción	De	A
Activo	Drop	Externa	Estación
	Allow	Estación	Externa

CP->externa	Acción	De	A
Pasivo	Drop	Estación	Externa
	Allow	Externa	Estación
Activo y pasivo	Allow	Externa	Estación
	Allow	Estación	Externa

Para las conexiones de nivel 2 se crean reglas "Allow" para ambos sentidos. Si el CP 1543-1 V1.1 se encuentra en un grupo VPN, el sentido "Externa" cambia a "Túnel".

En la columna "Dirección MAC de origen" y "Dirección MAC de destino" de estas reglas del cortafuegos se introduce en cada caso la dirección MAC del interlocutor.

CP->externa	Acción	De	A
activo, pasivo, activo y pasivo	Allow	Estación	Externa
	Allow	Externa	Estación

Nota

Cambiar la configuración de la conexión

Al cambiar la configuración de conexión de CPs cambian también las reglas de cortafuegos relativas a la conexión. Para visualizar las reglas de cortafuegos modificadas hay que volver a pulsar el botón "Actualizar reglas de conexión".

Convenios para reglas de cortafuegos creadas automáticamente

- Prioridad
  Las reglas tienen la máxima prioridad, por lo que en la lista de reglas local se incorporan en la parte superior.
- Borrar reglas
  Las reglas no se pueden borrar. El registro puede activarse y pueden asignarse servicios. Además es posible insertar un ancho de banda y un comentario.
- Cambiar una acción
  Si se cambia la acción "Allow" a "Drop" o viceversa, estas se sobrescribirán en la siguiente comparación del sistema. Si los cambios realizados deben conservarse, elija como acción "Allow*" o "Drop*". En ese caso solo se compara la dirección IP y la acción y el sentido se mantienen de la forma ajustada. Los ajustes de registro, servicio, ancho de banda y comentario se conservan cuando se produce una nueva comparación de sistema aunque no se cambie la acción a "Allow*" o "Drop*". Si se borra la conexión configurada, las reglas correspondientes se eliminan de la lista.

Módulo de seguridad en grupo VPN

La casilla de verificación "Solo comunicación tunelada" está activada de forma predeterminada. Si se desactiva la casilla de verificación, además de la comunicación

tunelada entre interlocutores tunelados existe la posibilidad de establecer comunicación con otros dispositivos con los que no existen túneles.

- La comunicación se produce sin túnel si la dirección del interlocutor pertenece a una estación conocida en STEP 7 con la que no hay configurado ningún túnel VPN.
- La comunicación transcurre a través del túnel VPN si la dirección del interlocutor es un punto final VPN.
- Si no es posible asignar unívocamente si una conexión debe transcurrir por dentro o por fuera del túnel VPN, la conexión se asigna al túnel VPN y se muestra la indicación correspondiente. La asignación puede adaptarse en el modo de cortafuegos avanzado, p. ej. cambiando el sentido "De" de "Túnel" a "Externa". Para que esta adaptación no se sobrescriba de nuevo al realizarse otra comparación del sistema, debe seleccionarse la acción "Allow*" o "Drop*".

Nota

En caso de ser necesario garantizar que solo sea posible la comunicación a través del túnel, cree las reglas de cortafuegos correspondientes en el modo de cortafuegos avanzado.

Para permitir exclusivamente la comunicación tunelada para un CP, inserte una regla "Drop" > "Any" > "Externa". Además, deben eliminarse las reglas de cortafuegos existentes que permitan una comunicación sin túnel.

8.1.4 Crear configuraciones

8.1.4.1 Información importante sobre el servidor web

Introducción

El servidor web permite a los usuarios autorizados vigilar y administrar la CPU a través de una red. Eso permite la evaluación y el diagnóstico a una gran distancia. Para ello solo se necesita un navegador web.

Los avisos y la información sobre el estado se muestran en páginas HTML.

Navegador web

Para acceder a las páginas HTML de la CPU se requiere un navegador web.

Los siguientes navegadores web se han probado para la comunicación con la CPU:

- Internet Explorer (versión 8)
- Mozilla Firefox (versión 21)
- mobileSafari (iOS5)

Acceso web a la CPU mediante PG/PC

Para acceder al servidor web, proceda de la siguiente manera:

1. Conecte el cliente (la PG/el PC) con la CPU a través de la interfaz PROFINET.
2. Abra el navegador web.
   En el campo "Dirección" del navegador web, introduzca la dirección IP de la CPU de la manera siguiente: http://ww.xx.yy.zz (entrada a modo de ejemplo: http://192.168.3.141). Se abrirá la página de inicio de la CPU. Desde allí podrá acceder al resto de las informaciones.

Información adicional

Encontrará más información sobre el servidor web de las diferentes familias de CPU bajo la palabra clave "Servidor web" en el sistema de información.

Encontrará información sobre cómo crear páginas web propias para el acceso a la CPU bajo la palabra clave "Páginas web definidas por el usuario" en el sistema de información.

Consulte también

Servidor web S7-1500 (http://support.automation.siemens.com/WW/view/es/59193560)

Documentación del S7-300 (http://support.automation.siemens.com/WW/view/es/12996906)

Documentación del S7-400 (http://support.automation.siemens.com/WW/view/es/44444467)

Servidor web S7-1200 (http://support.automation.siemens.com/WW/view/es/36932465)

8.1.4.2 Información importante sobre los modos de operación en PROFIBUS DP

Introducción

Se entienden por periferia descentralizada los sistemas maestros DP que constan de maestro DP y esclavos DP, están conectados con un bus y se comunican entre sí mediante el protocolo PROFIBUS DP.

A continuación se tratan los módulos con capacidad de comunicación e interfaz DP que pueden adoptar la función de maestro DP o de esclavo DP.

Opción "Maestro DP" y "Esclavo DP"

Los módulos aptos para comunicación, p. ej., las CPU con interfaz DP y los CP o CM con interfaz DP, cuentan con el área "Modo de operación" en las propiedades del módulo.

En las CPU S7-300 con interfaz DP integrada, p. ej., se pueden ajustar los modos de operación "Maestro DP" y "Esclavo DP". Una CPU o un CP que esté parametrizado como esclavo DP, también se denomina esclavo DP inteligente (esclavo I).

En las CPU S7-1500 con interfaz DP integrada solamente es posible el modo de operación "Maestro DP". Para que las CPU S7-1500 puedan funcionar como esclavo I, es necesario insertar el módulo de comunicaciones CM 1542-5 y parametrizarlo como esclavo DP.

Las CPU S7-1200 no tienen interfaces DP integradas. Para que una S7-1200 pueda funcionar como maestro DP o como esclavo DP, es necesario enchufar un módulo de comunicaciones CM 1243-5 (solo maestro DP) o un módulo de comunicaciones CM 1242-5 (solo esclavo DP; esclavo I).

Información adicional

Encontrará más información sobre la periferia descentralizada bajo las palabras clave "Periferia descentralizada" y "Esclavo I" en el sistema de información.

8.1.4.3 Configurar sistemas de automatización (S7-300, S7-400, S7-1500)

Configuración de módulos centrales

En los siguientes capítulos se explica cómo configurar los módulos centrales. Además se incluye información sobre el direccionamiento de módulos.

Sinopsis de la configuración de módulos centrales

Paso	Descripción
1	Seleccione un módulo central (Página 906).
2	Defina las propiedades del módulo. (Página 909).
3	Defina las direcciones de entrada y salida (Página 916).

Consulte también

Direccionamiento de módulos (S7-1500) (Página 915)

Direccionamiento de módulos (S7-300, S7-400) (Página 914)

Reglas para módulos (S7-300) (Página 898)

Reglas para módulos (S7-400) (Página 900)

Introducción a la configuración de hardware (Página 509)

Reglas para módulos (S7-300)

Introducción

En los módulos S7-300 existen reglas específicas para los slots.

Al seleccionar un módulo del catálogo de hardware en la vista de dispositivos, en el rack se marcan todos los slots posibles para el módulo seleccionado. Los módulos solo pueden arrastrarse a los slots seleccionados.

Si se inserta, desplaza o sustituye un módulo, también se aplican las reglas para los slots.

Reglas para los slots en los S7-300

Además de a las reglas generales de slots, el S7-300 está sujeto a las siguientes reglas adicionales:

- Pueden utilizarse hasta 4 racks.
  Rack 0:
  - Slot 1: sólo fuente de alimentación (p. ej., 6ES7 307-...) o vacío
  - Slot 2: sólo módulo central (p. ej., 6ES7 314-...)
  - Slot 3: módulo de interfaz (p. ej., 6ES7 360-.../365-...) o vacío
  - Slots 4 a 11: Módulos de señal o función, procesadores de comunicación o vacío

  Racks 1 a 3:
  - Slot 1: sólo módulo de fuente de alimentación (p. ej., 6ES7 307-...) o vacío
  - Slot 2: vacío
  - Slot 3: módulo de interfaz (p. ej., 6ES7 361-.../365-...)
  - Slots 4 a 11: Módulos de señal o función, procesadores de comunicación (dependiendo del módulo de interfaz utilizado) o vacío
- En el rack 3 no podrá utilizar el slot 11 si se está usando una CPU con periferia integrada.
- Los módulos deben sucederse sin dejar huecos.

Nota

En el S7-300, el slot 3 está reservado para el módulo de interfaz. Si se utiliza un rack individual no se ocupará este slot, ya que no es necesario insertar ningún módulo de interfaz. ¡En la estructura real no deberá haber ningún hueco, ya que de lo contrario se interrumpiría el bus de fondo!

El modelo ET 200M con bus de fondo activo sí puede presentar huecos en la ocupación.

Reglas especiales para el módulo comodín DM 370 Dummy

El módulo comodín DM 370 Dummy (6ES7 370-0AA01-0AA0) puede insertarse en lugar de un posterior módulo instalable.

El DM 370 Dummy reserva un área de dirección de 1 byte para un módulo en un esclavo modular DP (p. ej., para un módulo digital E/S).

Consulte también

Reglas generales para los slots (Página 511)

Reglas para módulos (S7-400)

Introducción

En los módulos S7-400 existen reglas específicas para los slots. Las reglas para la disposición de módulos en un rack del S7-400 dependen del tipo de rack utilizado.

Al seleccionar un módulo del catálogo de hardware en la vista de dispositivos, en el rack se marcan todos los slots posibles para el módulo seleccionado. Los módulos solo pueden arrastrarse a los slots seleccionados.

Si se inserta, desplaza o sustituye un módulo, también se aplican las reglas para los slots.

Reglas de slots del S7-400 para rack central

Además de a las reglas generales de slots, el S7-400 está sujeto a las siguientes reglas adicionales para módulos en un rack central:

- Insertar los módulos de fuente de alimentación únicamente en el slot 1 (excepción: módulos de fuente de alimentación redundantes)
- Insertar un máximo de 6 módulos de interfaz (IM de transmisión), de los cuales transferirán energía hasta 2 de ellos.
- Acoplar un máximo de 21 racks de expansión a través de módulos de interfaz al rack central
- Acoplar un máximo de 1 rack de expansión **con transferencia de energía** a una interfaz del IM de transmisión (IM 460-1 con IM 461-1)
- utilizar un máximo de 4 racks de expansión **sin transferencia de energía** (IM 460-0 con IM 461-0 o IM 460-3 con 461-3)

Reglas de slots del S7-400 para racks de expansión

Además de a las reglas generales de slots, el S7-400 está sujeto a las siguientes reglas adicionales para módulos en racks de expansión:

- Insertar los módulos de fuente de alimentación únicamente en el slot 1 (excepción: módulos de fuente de alimentación redundantes)
- Insertar el módulo de interfaz (IM de recepción) únicamente en el slot externo derecho (slot 9 o 18)
- Insertar los módulos de bus K únicamente en un rack cuyo número no sea superior a 6 (de lo contrario no podrá activarse)

Reglas especiales para los módulos de fuente de alimentación redundantes

Los módulos de fuente de alimentación redundantes pueden insertarse dos veces en un rack. Para saber si un módulo de fuente de alimentación es o no redundante, puede leer la Task Card "Catálogo de hardware" en la paleta "Información".

Deberán tenerse en cuenta las siguientes reglas:

- La inserción de módulos de alimentación redundantes sólo será posible en los racks previstos para ello (reconocible en el texto de información de la Task Card "Catálogo de hardware").
- Los módulos de alimentación redundantes deben insertarse en el slot 1 y siguientes (¡no dejar huecos!).
- En un mismo rack no podrán insertarse módulos de alimentación redundantes y no redundantes (no es posible un "Funcionamiento mixto").

Consulte también

Reglas generales para los slots (Página 511)

Reglas para módulos (S7-1500)

Utilización de módulos S7-1500

Introducción

Las CPUs del sistema de automatización S7-1500 son controladores modulares para soluciones de sistema de rendimiento medio y alto en la automatización de procesos discretos y continuos.

Campo de aplicación

El sistema de automatización S7-1500 está concebido como solución de sistema en la automatización de procesos y de la fabricación. El sistema de automatización S7-1500 es universal y constituye la solución óptima para aplicaciones con estructuras centralizadas y descentralizadas.

Su estructura escalable permite adaptar in situ y con precisión la estructura del controlador a las necesidades concretas.

El sistema de automatización S7-1500 está homologado para el tipo de protección IP20 y para el montaje en un armario eléctrico.

Estructura

El sistema de automatización S7-1500 se monta en un perfil de soporte y puede estar compuesto de un máximo de 32 módulos. Los módulos se conectan entre sí mediante un conector de bus. La alimentación eléctrica se realiza mediante el bus de fondo; la propia CPU también puede alimentar varios módulos. Si los módulos consumen más corriente de la que puede proporcionar la CPU, es posible conectar hasta tres fuentes de alimentación adicionales.

Consulte también

Reglas para los slots en módulos S7-1500 (Página 902)

Estructura de una estación S7-1500 (Página 903)

Equilibrado de potencia (Página 905)

Comprobación del equilibrio de potencia (Página 906)

Reglas para los slots en módulos S7-1500

En los módulos S7-1500 existen reglas específicas para los slots. Al seleccionar un módulo del catálogo de hardware en la vista de dispositivos, en el rack se marcan todos los slots posibles para el módulo seleccionado. Los módulos solo pueden arrastrarse a los slots seleccionados. Las reglas se aplican a los slots siempre que se inserte, se desplace o se cambie un módulo.

Reglas para los slots en S7-1500

Para S7-1500 se aplican las siguientes reglas de slots:

- Además de la CPU es posible conectar a un rack hasta 31 módulos.
- Se pueden enchufar como máximo tres fuentes de alimentación. El número mínimo de fuentes de alimentación necesario depende de la potencia consumida por cada módulo.
  - A la izquierda de la CPU (slot 0) se puede enchufar una fuente de alimentación. Junto con la CPU, esta fuente alimenta todos los dispositivos colocados a continuación en el rack. La fuente de alimentación y los módulos alimentados por ella constituyen un segmento de potencia.
  - En los slots situados a la derecha de la CPU se pueden conectar hasta dos fuentes de alimentación adicionales. Si en el slot 0 no hay instalada ninguna fuente de alimentación, es posible conectar hasta tres fuentes de alimentación a la derecha de la CPU.

Nota

Segmentos de potencia

Una fuente de alimentación y los módulos alimentados por ella constituyen un segmento de potencia. Una fuente de alimentación proporciona corriente siempre a los módulos que están instalados a su derecha. En la siguiente fuente de alimentación termina el segmento de potencia y comienza un nuevo segmento de potencia para la siguiente fuente de alimentación y para los módulos instalados a su derecha. Los módulos instalados a su izquierda son alimentados por la fuente de alimentación precedente o por la CPU.

Consulte también

Equilibrado de potencia (Página 905)

Estructura de una estación S7-1500 (Página 903)

Estructura de una estación S7-1500

Se dispone de diversos módulos de periferia para ampliar cada una de las CPUs S7-1500. Además de los módulos de entradas y salidas digitales es posible utilizar módulos de tecnología, de comunicación y especiales. A pesar de la gran capacidad de las CPUs S7-1500, si la estructura de hardware es muy amplia, existen límites relativos al número de módulos de periferia y de sistemas I/O configurados en función de las CPUs utilizadas. Por tanto, en la configuración hay que tener en cuenta los límites descritos en la hoja de datos técnicos de la CPU correspondiente.

Configuración máxima de una estación

Al compilar la configuración de hardware se comprueba si se han sobrepasado los límites máximos de configuración de una estación. Sin embargo, para la configuración de una estación S7-1500 se recomienda tener en cuenta las restricciones de cantidades que puedan existir. Al crear la configuración de hardware, compruebe sobre todo los criterios siguientes, consultando los datos técnicos de los dispositivos:

- Número máximo de módulos de comunicaciones para PROFINET, PROFIBUS e Industrial ETHERNET y para procesadores de comunicaciones
  Por ejemplo, en una S7-1513 (6ES7 513-1AL00-0AB0) son seis unidades.
- Número máximo de sistemas PROFINET IO y de sistemas PROFIBUS DP descentralizados
  Los sistemas IO conectados mediante un dispositivo I no se tienen en cuenta para calcular el número máximo. En tal caso se aplican las reglas para la configuración máxima del dispositivo I.
- Número máximo de unidades de periferia descentralizada (dispositivos IO y esclavos DP)
  Se cuentan todas las unidades de periferia descentralizada que están asignadas a la CPU, independientemente de la interfaz a la que estén conectadas (interfaz integrada o interfaz de los CPs/CMs). La CPU solo se incluye en el recuento de unidades de periferia si está configurada como I-Device.
  Si el sistema PROFINET IO tiene conectados I-Devices, no se cuentan los dispositivos IO que están en el sistema PROFINET IO subordinado de un I-Device.
- Número máximo de módulos/submódulos
  El número máximo de módulos insertables es muy elevado y pocas veces se alcanza en las estructuras de hardware habituales de un sistema de automatización S7-1500.
  Cuando se utilizan muchos módulos con submódulos en instalaciones muy grandes, existe la posibilidad de que se sobrepasen los recursos disponibles para módulos/submódulos.
  Los módulos de periferia sencillos que no tienen submódulos ocupan un recurso.
  Los módulos complejos pueden tener varios submódulos y ocupan un recurso para cada submódulo. Además, un módulo con submódulos ocupa otro recurso adicional (que representa el módulo completo).
  Ejemplo:
  un módulo de interfaz IM 155-5PNHF ocupa cinco recursos, repartidos del siguiente modo: dos recursos para los dos puertos de la interfaz PROFINET, un recurso para la interfaz de la interfaz PROFINET y un recurso para el submódulo que contiene los parámetros del módulo de interfaz. Tal como se ha explicado anteriormente, hay que agregar un recurso más, pues se trata de un módulo con submódulos.

Ejemplo

Las figuras siguientes muestran ejemplos de configuración hardware y explican la manera de contar descrita anteriormente:

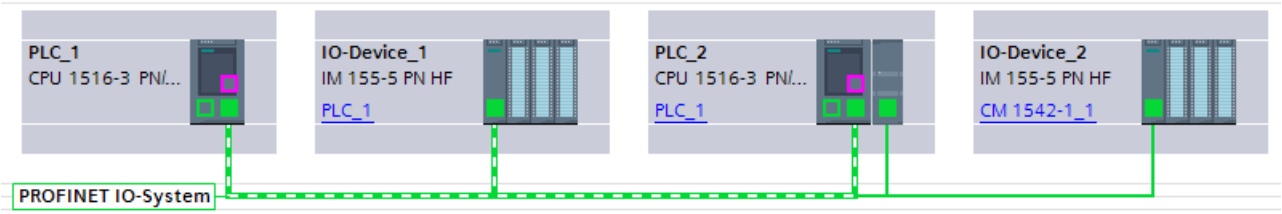


Figura 8-7 En esta figura, PLC_1 está configurado como controlador IO. IO-Device_1 y PLC_2 son dispositivos del sistema IO de PLC_1. IO-Device_2 está conectado a la misma red, pero no forma parte del sistema IO. Por tanto, el sistema IO de PLC_1 cuenta como dos dispositivos.

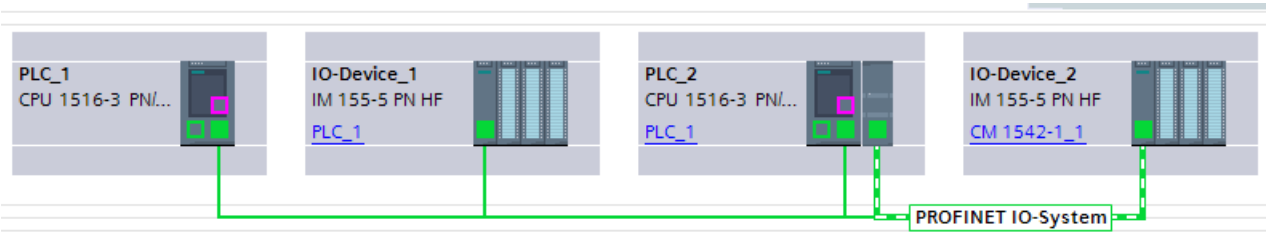


Figura 8-8 La segunda figura muestra que PLC_2 es un dispositivo I en el sistema IO de PLC_1, pero que al mismo tiempo es un controlador IO. A través del módulo de comunicación se establece un sistema IO subordinado, al que está conectado IO-Device_2. En total, para PLC_2 se cuentan dos dispositivos.

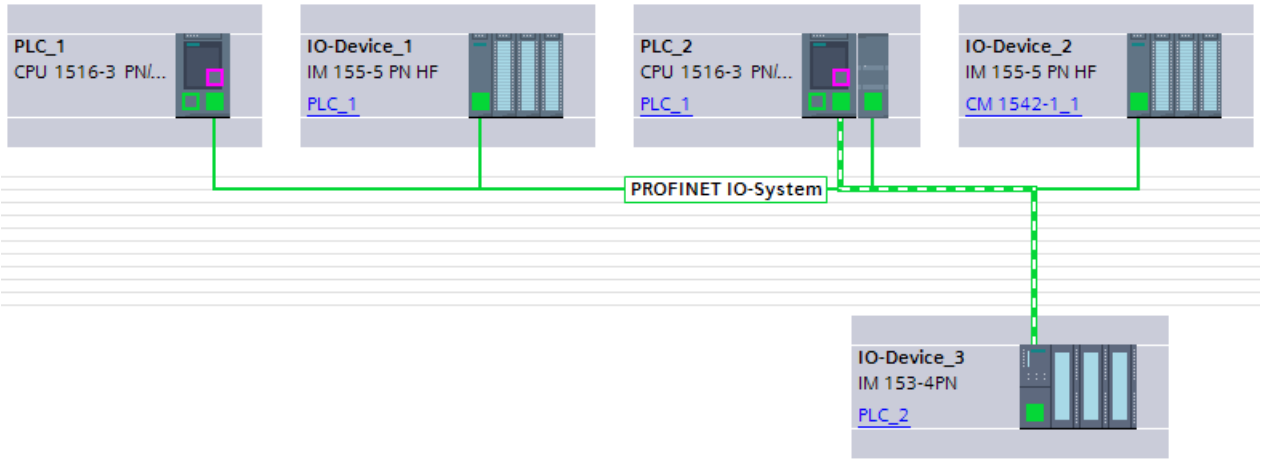


Figura 8-9 En la tercera figura hay otro sistema IO asignado a PLC_2. PLC_2 dispone ahora de dos sistemas IO en total, uno en la interfaz interna y otro en el módulo de comunicación. En el sistema IO nuevo se ha agregado otro dispositivo. En total, con esta configuración hardware se cuentan tres dispositivos para PLC_2: un I-Device (PLC_2 está asignado a PLC_1) y dos dispositivos IO (asignados a PLC_2).

Consulte también

Reglas para los slots en módulos S7-1500 (Página 902)

Equilibrado de potencia (Página 905)

Equilibrado de potencia

Introducción

Todos los módulos de periferia toman la potencia necesaria para activar su electrónica del bus de fondo. El bus de fondo se alimenta desde el módulo central, y en caso necesario, de fuentes de alimentación adicionales.

Comprobación del equilibrio de potencia durante la configuración

Para la comprobación del equilibrio de potencia se cuenta con el apoyo de STEP 7. En una tabla se muestran todos los módulos de periferia enchufados con el consumo eléctrico máximo de cada uno de ellos. El consumo eléctrico se compara con los módulos de alimentación del sistema enchufados. De esta manera se sabe si se dispone de alimentación suficiente para todos los módulos.

Se accede a la tabla en la ventana de inspección, en el área "Fuente de alimentación del sistema", después de seleccionar la CPU correspondiente y elegir la ficha "Propiedades".

Además, durante la configuración se comprueba si se dispone de suministro eléctrico suficiente. Si los módulos de periferia consumen más potencia de la que entra en el bus de fondo, las filas correspondientes de la tabla se marcan en rojo y se emite una advertencia al respecto después de compilar la configuración hardware.

Comprobación del equilibrio de potencia por la CPU

La CPU comprueba durante el funcionamiento si se cumple el equilibrio de potencia:

- en cada conexión (POWER ON)
- cada vez que cambia la estructura de hardware

Si la CPU detecta una sobrecarga del bus de fondo, desconecta todos los módulos de periferia alimentados desde la CPU o desde la fuente de alimentación y registra un mensaje en el búfer de diagnóstico. La CPU no arranca mientras el equilibrio de potencia sea negativo.

Causas de sobrecarga

La sobrecarga puede deberse a que la estructura de hardware no es congruente con la configuración. Por ejemplo, porque no están enchufadas o conectadas las fuentes de alimentación necesarias para el funcionamiento, o porque se han enchufado más módulos de los configurados.

Consulte también

Comprobación del equilibrio de potencia (Página 906)

Reglas para los slots en módulos S7-1500 (Página 902)

Estructura de una estación S7-1500 (Página 903)

Comprobación del equilibrio de potencia

Durante la configuración de una estación se puede ir comprobando si se dispone de potencia suficiente para todos los módulos de periferia. Para hacerlo, indique previamente si la CPU recibe alimentación de tensión.

Procedimiento

Para activar la fuente de alimentación de la CPU y evaluar el equilibrio de potencia, proceda del siguiente modo:

1. Seleccione la CPU S7-1500, p. ej. en la vista de redes.
2. Abra la ficha "Propiedades" de la ventana de inspección.
   Se muestran las propiedades de la CPU.
3. Seleccione la entrada "Fuente de alimentación del sistema" en la navegación local.
4. Si la CPU recibe tensión eléctrica, active la opción "Conexión a tensión de alimentación L +". Dicha opción está activada por defecto. De este modo, el consumo de potencia de la CPU se incluye en el equilibrio de potencia.
5. En la tabla "Equilibrado de potencia" compruebe si el equilibrio de potencia es positivo. Si es negativo, los módulos que no reciben alimentación suficiente estarán marcados en rojo.

Consulte también

Equilibrado de potencia (Página 905)

Seleccionar módulos centrales

Introducción

En la vista de redes se selecciona una CPU del catálogo de hardware y se crea junto con un rack. Los módulos deseados se arrastran del catálogo de hardware al dispositivo y se ordenan automáticamente en el rack.

Selección de los componentes en el catálogo de hardware

En el catálogo de hardware, todos los componentes se visualizan en forma de carpetas. Al abrir dichas carpetas, aparecerán las diferentes versiones de los componentes de hardware seleccionados junto con sus respectivas referencias.

Aquí se muestra a modo de ejemplo cómo crear una CPU con un rack en la vista de redes.

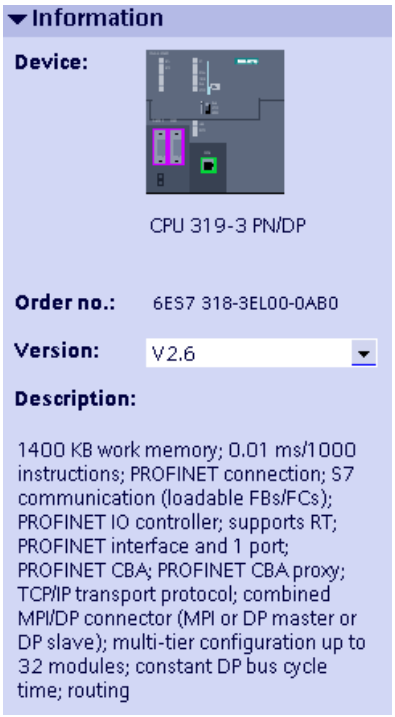
Requisitos

- El catálogo de hardware está abierto.
- Se encuentra en la vista de redes.

Procedimiento

Para seleccionar una CPU del catálogo de hardware, proceda del siguiente modo:

1. En el catálogo de hardware, navegue hasta la carpeta que contiene las CPUs deseadas.
2. Abra la carpeta con el tipo de CPU que busque. Aparecerán todas las referencias del tipo de CPU seleccionado.
3. Haga clic en la referencia de una CPU para obtener información acerca de la CPU seleccionada en la paleta "Información".



4. En la lista desplegable "Versión" ajuste la versión de firmware adecuada en su CPU.
5. Cree la CPU y un rack. Para ello existen las posibilidades siguientes:
   - Arrastre la CPU desde el catálogo de hardware hasta la vista de redes con la función Drag & Drop.
   - Haga doble clic en la entrada de la CPU del catálogo de hardware.

Consulte también

Agregar un dispositivo a la configuración de hardware (Página 520)

Insertar un módulo en un rack (Página 523)

Trabajar con racks (Página 514)

Examinar el catálogo de hardware (Página 512)

Seleccionar módulos de seguridad

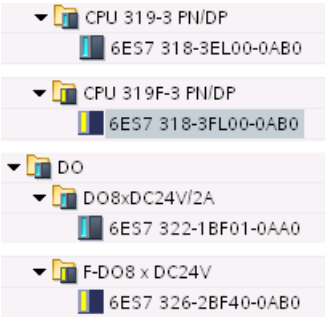
Introducción

Los sistemas de automatización de seguridad (sistemas F) sirven para controlar procesos con un estado seguro que se alcanza inmediatamente por desconexión. Los sistemas F controlan procesos cuya desconexión inmediata no supone ningún peligro ni para las personas ni para el medio ambiente.

Selección de los componentes F en el catálogo de hardware

En el catálogo de hardware, todos los componentes se visualizan en forma de carpetas. Los componentes de hardware de seguridad se muestran con símbolos amarillos en el catálogo de hardware, al contrario que los componentes que no son de seguridad, que se muestran en color azul.

La figura siguiente muestra extractos del catálogo de hardware con dos CPUs y dos módulos de salidas digitales, unos en versión normal (azules) y otros en versión F (amarillos):

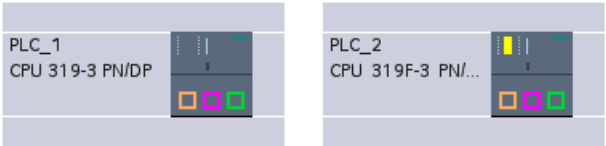


El color ayuda a identificar fácil y rápidamente los componentes F dentro del catálogo de hardware.

Visualización de los componentes F en el editor de hardware y redes

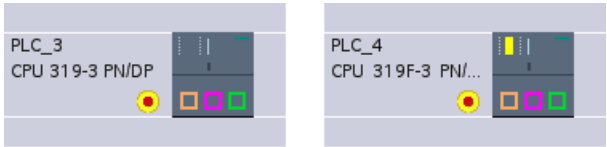
Si se utilizan dispositivos de seguridad y dispositivos "normales" en las vistas gráficas del editor de hardware y redes, es posible reconocer fácilmente los primeros por su identificación amarilla.

La figura siguiente muestra dos CPUs. La CPU F está marcada en amarillo:



En la vista de dispositivos también es posible distinguir los módulos de seguridad en racks o dispositivos gracias a su identificación amarilla. Puesto que los módulos no aptos para redes no aparecen en la vista de redes ni en la vista topológica, los dispositivos con componentes F no aptos para redes se muestran con un símbolo especial en la vista de redes y en la vista topológica.

La figura siguiente muestra una CPU "normal" y una CPU F en la vista de redes, y ambas disponen de componentes F:



La identificación cromática de los dispositivos y módulos de seguridad permite identificar rápidamente los componentes F incluso en las vistas gráficas del editor de hardware y redes.

Modificar las propiedades de los módulos centrales

Cada componente de hardware (módulo, interfaz o módulo de interfaz) tiene una serie de propiedades predeterminadas; p. ej., los módulos de entradas analógicas tienen tipos y áreas de medición predeterminados.

Propiedades de los módulos centrales

Para el comportamiento del sistema son muy importantes las propiedades de las CPUs. Una CPU permite ajustar, p. ej.:

- Comportamiento de arranque
- Áreas de datos locales
- Prioridades de alarma
- Áreas de memoria
- Comportamiento remanente
- Marcas de ciclo
- Nivel de protección
- Contraseña

Las posibilidades de entrada indican qué puede ajustarse y con qué rangos de valores. Los campos no editables están atenuados o no aparecen en la ventana de propiedades.

En las propiedades de una interfaz puede, p. ej., parametrizarse una interfaz MPI integrada o una interfaz PROFIBUS DP y conectarla con una subred existente.

Requisitos

Los componentes de hardware cuyas propiedades deben modificarse se han dispuesto en un rack.

Procedimiento

Para modificar las propiedades y los parámetros de los componentes de hardware, proceda del siguiente modo:

1. En la vista de dispositivos o de redes seleccione el componente de hardware que desee editar.
   También podrá seleccionar piezas individuales de un componente hardware, como p. ej.:
   - Interfaz
   - Entrada
   - Salida
2. Edite los ajustes del objeto seleccionado:
   - En la vista general de dispositivos se editan las direcciones y los nombres.
   - En la ventana de inspección existen otras posibilidades de ajuste adicionales.

No es necesario confirmar las entradas, los valores modificados se aplican inmediatamente.

Otras posibilidades de parametrización

Para los sistemas de automatización del S7-300/400 tiene además la posibilidad de, p. ej., ajustar parámetros en el programa del usuario en módulos analógicos. Para este objetivo acceda, en el programa del usuario, a las funciones de sistema WR_PARM (Página 6882) (SFC 55), DR_DPARM (Página 6881) (SFC 56) y PARM_MOD (Página 6877) (SFC 57). Sin embargo, estos ajustes se perderían en un arranque o reinicio (en caliente).

Información adicional

Para más información sobre las funciones estándar y de sistema del S7-300/400 tenga en cuenta el manual de referencia de software de sistema del S7-300/400.

Consulte también

Editar las propiedades y los parámetros (Página 529)

Acoplamiento de racks (S7-300)

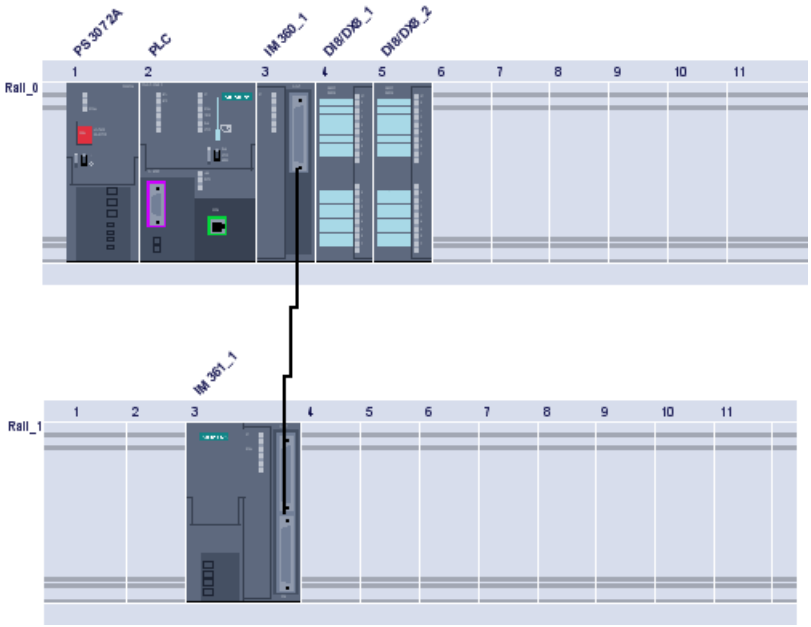
Introducción

Dependiendo del dispositivo utilizado, en algunas CPUs S7-300 existe la posibilidad de acoplar racks a través de módulos de interfaz.

Acoplamiento de racks

Para acoplar un rack a través de módulos de interfaz deberá arrastrarse otro rack desde el catálogo de hardware, en la vista de dispositivos. A continuación se insertará un módulo

emisor en el rack central y un módulo receptor en el rack nuevo. Automáticamente se arrastrará un acoplamiento entre el módulo emisor y el receptor.



En el S7-300 pueden acoplarse hasta cuatro racks. Los módulos de interfaz deben insertarse siempre en el slot 3.

Nota

Si su configuración no dispone de ampliaciones el slot 3 se marcará siempre como vacío, incluso aunque en la realidad esté directamente conectado el módulo del slot 4 a la CPU, no habiendo así ningún hueco.

Utilice los siguientes módulos de interfaz:

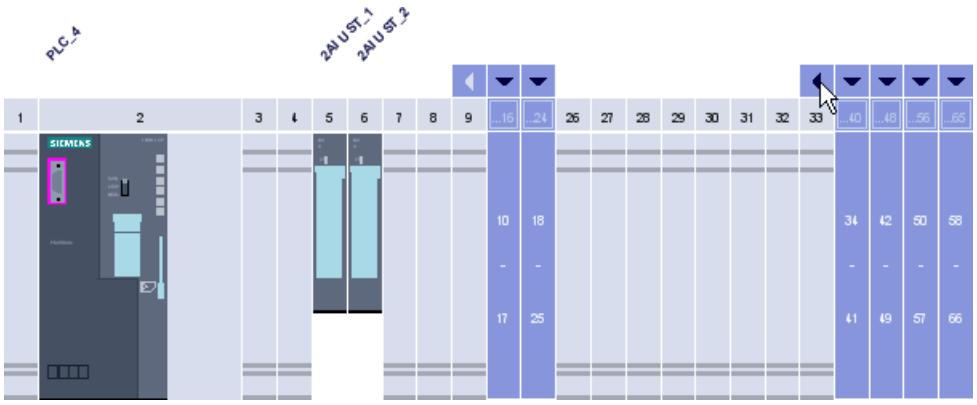
- Para la ampliación de un único rack:
  Rack 0 y 1: IM 365
- Para la ampliación de hasta 3 racks:
  Rack 0: IM 360
  Racks 1 a 3: IM 361

Si prefiere un acoplamiento manual, borre en primer lugar la línea de unión que representa el acoplamiento. A continuación, manteniendo pulsado el botón del ratón, arrastre una nueva línea de unión entre las interfaces del módulo emisor y el receptor.

Rack ampliable

Dependiendo del dispositivo utilizado, en algunas CPU (p. ej., IM 151-1) existen racks ampliables cuyos grupos de slots podrá abrir a través del símbolo de flecha situado encima del slot ampliable. Si están cerrados, se muestran el primer y último número de slot del grupo.

La siguiente imagen muestra un sector de un rack ampliable:



No es posible cerrar los grupos de slots con módulos ya insertados. Por este motivo tampoco puede cerrarse el primer grupo de slots con el dispositivo. Cuando esto sucede podrá detectarlo por el tono gris que adquiere el símbolo de flecha.

Consulte también

Trabajar con racks (Página 514)

Acoplamiento de racks (S7-400)

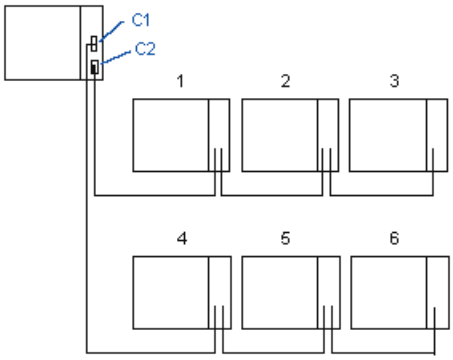
Ampliar un rack central con racks de ampliación

Configuración de racks de ampliación en el SIMATIC 400

En el SIMATIC 400 se dispone de posibilidades de ampliación más complejas gracias a los diversos racks y módulos de interfaz.

Todos los racks de ampliación que están conectados a una interfaz del IM emisor del rack central forman una línea.

La figura siguiente muestra tres racks de ampliación conectados a una interfaz del IM transmisor, respectivamente.



Reglas para acoplar racks de ampliación

Acoplar racks de ampliación

En caso de acoplar racks de ampliación (SIMATIC 400) a una interfaz del módulo de interfaz (IM emisor) del rack central, las propiedades siguientes deben coincidir tanto en el IM emisor como en el receptor:

- transferencia de intensidad (con/sin)
- tipo de acoplamiento (centralizado/descentralizado)
- transmisión del bus K (con/sin transmisión de alarmas)

Reglas para slots del S7-400 para racks de ampliación

Además de a las reglas generales para slots, el S7-400 está sujeto a las siguientes reglas adicionales para módulos en racks de ampliación:

- Insertar los módulos de alimentación en el slot 1
- Insertar el módulo de interfaz (IM de recepción) en el slot externo derecho (slot 9 o slot 18)
- Insertar los módulos de bus K únicamente en un rack de ampliación cuyo número no sea superior a 6 (de lo contrario no es posible acceder a él)

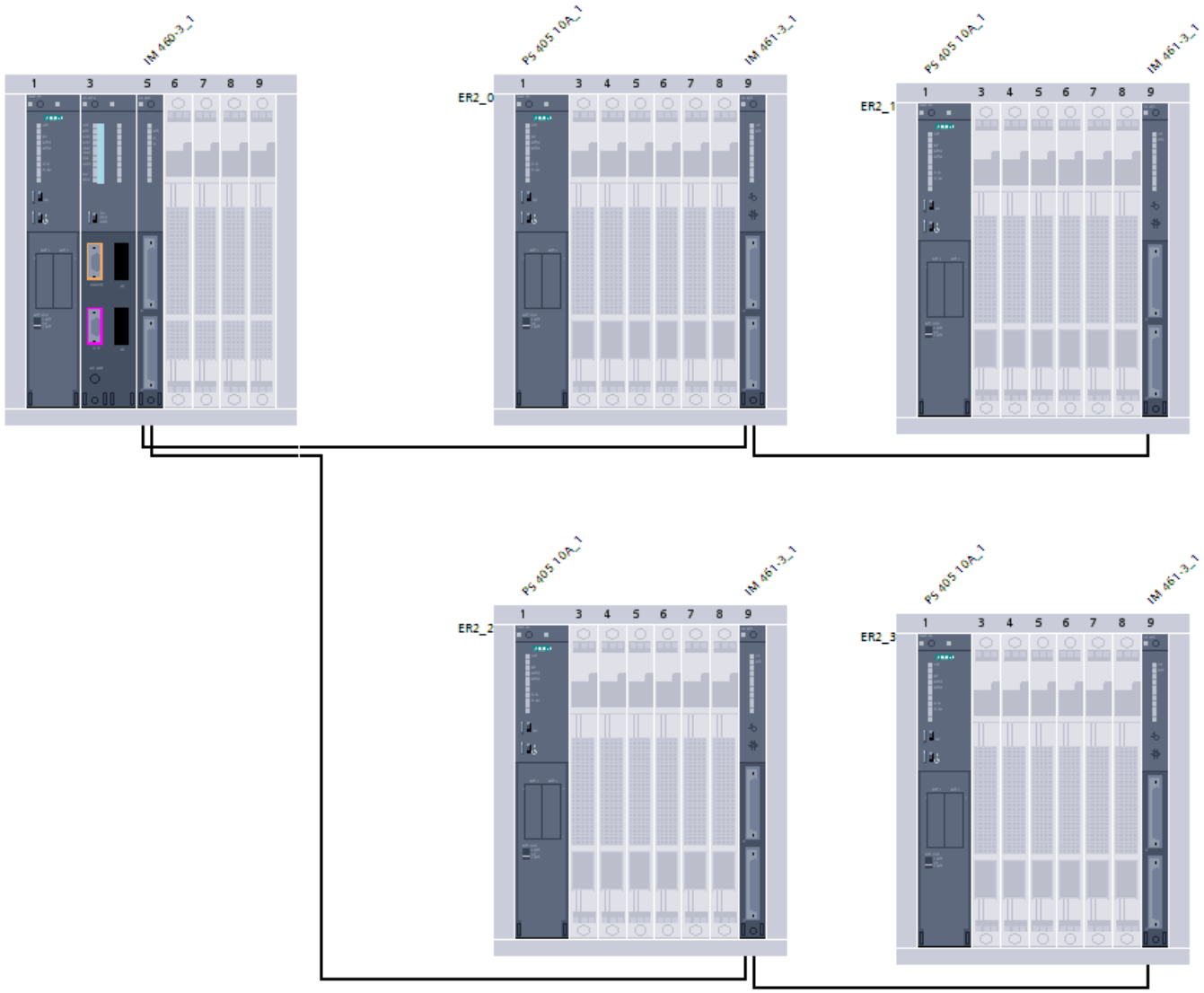
Disposición del rack de ampliación

Procedimiento general

Para disponer los racks de ampliación y realizar conexiones entre los módulos de interfaz de los racks, proceda del siguiente modo:

1. Seleccione un rack de CPU (p. ej., CPU 417) del "Catálogo de hardware".
2. Utilizando la función Drag&Drop, arrastre el controlador hasta la vista de dispositivos.
3. Asigne módulos al rack del controlador (p. ej., fuente de alimentación e IM 460).
4. Seleccione racks de ampliación adecuados (p. ej., ER1) del "Catálogo de hardware".
5. Utilizando la función Drag&Drop, arrastre sucesivamente los racks de ampliación hasta la vista de dispositivos.
6. Asigne módulos a cada rack (p. ej., fuente de alimentación o IM 461).
   Importante: los módulos de interfaz deben estar enchufados en todos los racks para poder acoplar los racks entre sí.
7. Conecte la interfaz del IM emisor con la del IM receptor del rack deseado.
   Seguidamente, la asignación entre los racks se muestra con líneas de conexión.

La figura siguiente muestra la conexión por parejas de los racks de ampliación a una interfaz del IM transmisor a través de un IM receptor.



Direccionar módulos

Direccionamiento de módulos (S7-300, S7-400)

Introducción

En las columnas Dirección I y Dirección Q de la vista general de dispositivos se ven las direcciones o áreas de direcciones de los módulos.

Las direcciones de entrada y salida se asignan automáticamente al insertar módulos en el rack. La dirección del primer canal representa la dirección inicial de un módulo. Las direcciones de los demás canales se derivan de dicha dirección inicial. La dirección final se deriva de la longitud de dirección específica del módulo.

Particularidad: Direcciones de entrada con asterisco (*)

En la vista de dispositivos se marcan con un (*) determinadas direcciones del área de la dirección de entrada. En este caso se trata de direcciones de diagnóstico, p. ej., de módulos de interfaz que no direccionan ningún dato de usuario, sino que son necesarios para el diagnóstico del sistema.

Diferenciación entre direcciones de nodos y direcciones de periferia

En la asignación de direcciones se diferencia entre las direcciones de nodos y de periferia (direcciones de entrada y salida):

- Las direcciones de nodos son direcciones de módulos programables (direcciones MPI, PROFIBUS e Industrial Ethernet). Se requieren para direccionar los distintos dispositivos de una subred, p. ej. para cargar un programa de usuario en una CPU.
- Las direcciones de periferia se necesitan para leer entradas y activar salidas en el programa de usuario.

Particularidad: Direcciones MPI de FM y CP (S7-300)

Los CPs y FMs con dirección MPI propia, esta viene predeterminada por STEP 7.

Las direcciones MPI se pueden modificar (en la ventana de inspección, a través de "Propiedades > Interfaz MPI").

En el caso de los CPs y FMs sin dirección MPI, la CPU direcciona estos módulos a través del bus K mediante el número de rack y el número de slot.

Consulte también

Definir las direcciones de entrada y salida (Página 916)

Asignar direcciones a una ubicación del programa (Página 928)

Direccionamiento de módulos (S7-1500)

Introducción

En las columnas Dirección I y Dirección Q de la vista general de dispositivos se ven las direcciones o áreas de direcciones de los módulos. Al lado hay otras direcciones que se explican a continuación.

Dirección E/S (dirección de periferia)

Las direcciones E/S (de entrada y salida) se necesitan para leer entradas y activar salidas en el programa de usuario.

Las direcciones de entrada y salida se asignan automáticamente al insertar módulos en el rack. La dirección del primer canal representa la dirección inicial de un módulo. Las direcciones de los demás canales se derivan de dicha dirección inicial. La dirección inicial se deriva de la longitud de dirección específica del módulo.

Dirección de dispositivo (p. ej. dirección Ethernet)

Las direcciones de dispositivos son direcciones de módulos programables (direcciones Industrial Ethernet). Se requieren para direccionar los distintos dispositivos de una subred, p. ej. para cargar un programa de usuario en una CPU.

ID de hardware para la identificación de módulos y submódulos

Además de las direcciones E y S, se asigna automáticamente un identificador de hardware (ID HW) que se utiliza para direccionar e identificar el módulo. También los submódulos (unidades de un módulo), como un contador integrado, obtienen una ID HW. La ID HW consiste en un número entero y es emitida por el sistema junto con avisos de diagnóstico con el fin de localizar el módulo o submódulo defectuoso. La ID HW se utiliza también para una serie de instrucciones para direccionar el módulo correspondiente. No es posible modificar la ID HW.

Consulte también

Información importante sobre identificadores de hardware (Página 917)

Ventana de inspección (Página 501)

Definir las direcciones de entrada y salida

Las direcciones de entrada y salida están predeterminadas de forma automática, aunque posteriormente podrá modificar dicho direccionamiento.

Requisitos

- Se encuentra en la vista de dispositivos.
- El módulo es un esclavo DP compacto o está enchufado en alguno de los objetos siguientes:
  - Rack central
  - Rack de ampliación
  - Esclavo DP
- La CPU permite la asignación libre de direcciones.

Procedimiento

Para modificar el área de direcciones predeterminada, proceda del siguiente modo:

1. En la vista de dispositivos, haga clic en el módulo cuya dirección inicial desea ajustar.
2. Haga clic en "Direcciones E/S", dentro de "Propiedades", en la ventana de inspección.
3. Introduzca la dirección inicial deseada en "Dirección inicial".
4. Pulse <Intro> o haga clic en cualquier otro objeto para aplicar el valor modificado.

Si ha introducido una dirección no válida aparecerá un aviso en el que podrá bien aceptar la dirección propuesta o bien introducir una.

Nota

También existe la posibilidad de modificar las direcciones directamente en la vista de dispositivos. Cuando no es posible realizar una norma de direccionamiento por parte de la CPU, las direcciones se representarán en color gris.

En los módulos dentro de un segmento de bus local formados por un FM (S7-300) o FM especiales (S7-400), asigne además otra dirección inicial. Además de la dirección inicial para la CPU, el módulo tiene también una dirección inicial para el FM. En este caso, en la vista de dispositivos siempre se muestra la dirección inicial desde el punto de vista del FM.

Consulte también

Editar las propiedades y los parámetros (Página 529)

Información importante sobre identificadores de hardware

Introducción

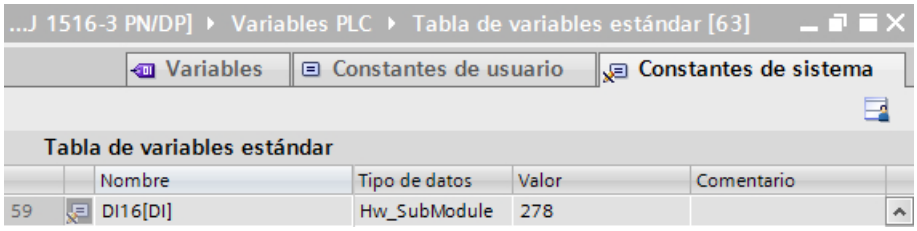
Todos los objetos de hardware, p. ej. un módulo, una interfaz, un puerto de un módulo o áreas de transferencia de dispositivos I, tienen un identificador de hardware asignado automáticamente por STEP 7 al configurar un componente de hardware.

En S7-1500 y S7-1200, dichos identificadores de hardware tienen la misma función que las direcciones de diagnóstico de componentes de hardware sin datos de usuario, o que las direcciones iniciales de objetos de hardware con datos de usuario (en S7-300/400). En caso de modificación de las direcciones E/S, sus identificadores de hardware se mantienen igual. En caso de modificación de las direcciones E/S, no hay que adaptar las instrucciones que direccionan a través de un identificador de hardware.

Identificadores de hardware en las propiedades de los componentes

Para los objetos de hardware mencionados anteriormente, en la ventana de inspección, ficha "General" (área Identificador de hardware) se muestra un número entero.

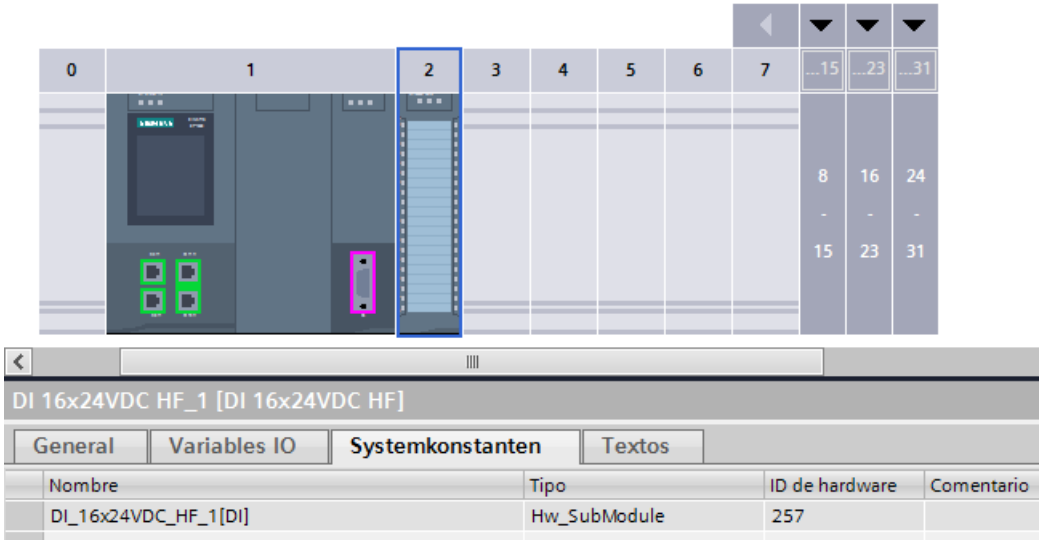
Ejemplo:



Identificadores de hardware en la ficha "Constantes de sistema" de la ventana de inspección

Si se selecciona un objeto de hardware en la vista de dispositivos y se elige la ficha "Constantes de sistema" en la ventana de inspección se muestran todos los identificadores de hardware del objeto.

Ejemplo:



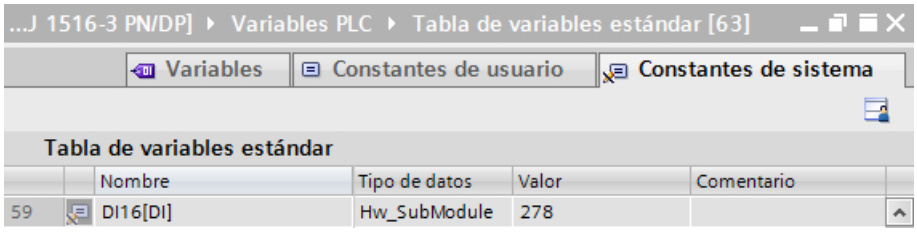
Identificadores de hardware en las constantes de sistema de la tabla de variables PLC

Adicionalmente, STEP 7 rellena entradas en las variables PLC (ficha "Constantes del sistema") para cada componente de hardware configurado. Dichas entradas son de solo lectura. Aparte del valor del número entero, también se asigna un nombre, un tipo de datos de hardware y, en el comentario, la ruta del componente de hardware.

El nombre es el nombre editable de las propiedades del componente (ventana de inspección).

Recomendación: Para facilitar la identificación de objetos de hardware, durante la configuración del hardware asígneles nombres que resulten "explicativos". Es conveniente que los nombres hagan patente la parte de instalación o la estación a la que pertenece el objeto.

Ejemplo:



Identificadores de hardware al programar

Si en una instrucción se direcciona un componente de hardware, p. ej. con el parámetro de entrada "ID" o "LADDR", al hacer doble clic en este parámetro de entrada se muestra una lista de todos los componentes de hardware seleccionables.

En la información de arranque sobre, por ejemplo, los OB de alarma de proceso, podrá identificar el módulo que lanza alarmas a través del identificador de hardware.

Consulte también

OBs de alarma de proceso (Página 1973)

Ejemplo de un evento de alarma de proceso (Página 1927)

Ejemplos de uso de identificadores de hardware (Página 919)

Ventana de inspección (Página 501)

Ejemplos de uso de identificadores de hardware

Los identificadores de hardware se utilizan para direccionar objetos de hardware como, p. ej., módulos, submódulos o dispositivos. Dependiendo de la instrucción, los identificadores de hardware tienen diferentes tipos de datos de hardware. El tipo de datos de hardware admisible aparece, para todos los componentes configurados, en la tabla de variables estándar, ficha "Constantes de sistema".

A continuación se explican los identificadores de hardware que deben utilizarse para algunas instrucciones seleccionadas.

Nota

Información sobre el respectivo identificador de hardware

Para toda instrucción avanzada que direccione objetos de hardware, el objeto de hardware que debe utilizarse figura en la ayuda. La ayuda correspondiente a la instrucción seleccionada se abre mediante la tecla <F1>.

Recomendación

En el editor de programas se puede introducir el nombre del objeto de hardware o el identificador numérico de hardware de dicho objeto.

Recomendación: Asigne nombres explicativos a los objetos de hardware, p. ej. a los módulos. Consulte el nombre y el identificador de hardware en la ficha "Constantes de sistema" de la vista de dispositivos e introdúzcalos en el editor de programación utilizando la instrucción correspondiente. Una vez haya asignado un nombre explicativo, lo encontrará fácilmente en la lista desplegable para seleccionar el identificador de hardware en el parámetro de bloque "ID".

Otra posibilidad para introducir el identificador de hardware

Si lo desea, puede arrastrar el identificador de hardware mediante Drag & Drop desde la tabla de variables al parámetro de bloque "ID".

Ventaja: el tipo de dato de hardware es comprobado por el editor de programas y, dado el caso, rechazado, si se ha seleccionado un objeto de hardware no admitido.

En caso de introducción numérica no se puede realizar una comprobación.

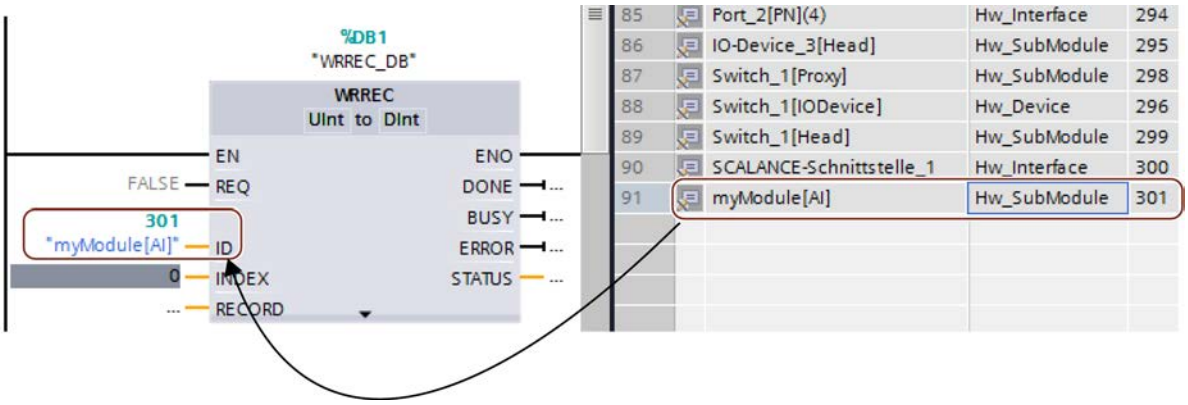
Si, p. ej., el objeto de hardware admisible es un dispositivo IO, marque en la vista de redes el dispositivo IO completo y lea el identificador de hardware en "Propiedades" (ficha "General") de la ventana de inspección. Si en este caso no se selecciona el dispositivo IO, sino el módulo de interfaz del dispositivo IO, el identificador de hardware visible en la ventana de inspección no sería admisible para esta instrucción. En caso de introducirse numéricamente el identificador de hardware durante la programación, este error pasa desapercibido.

Ejemplo de introducción de un identificador de hardware

En la tabla de variables estándar, ficha "Constantes de sistema", figura p. ej. el siguiente nombre de módulo:
`myModule[AI]`

1. Abra el editor de programación
2. Abra la tabla de variables estándar (carpeta "Variables PLC".)
3. Coloque las ventanas del editor de programación y de la tabla de variables estándar una junto a otra.
   Para ello, haga clic en el botón "Dividir área del editor verticalmente" de la barra de herramientas.

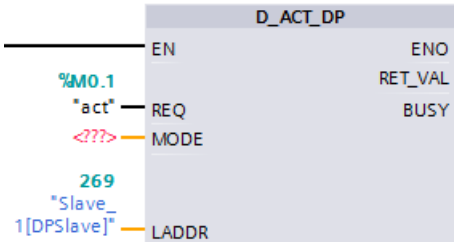
4. Arrastre "myModule[AI]" desde la ficha "Constantes de sistema" hasta el campo "ID" o "LADDR" de la instrucción en cuestión como se muestra en la figura inferior.



Ejemplos de direccionamiento con identificador de hardware

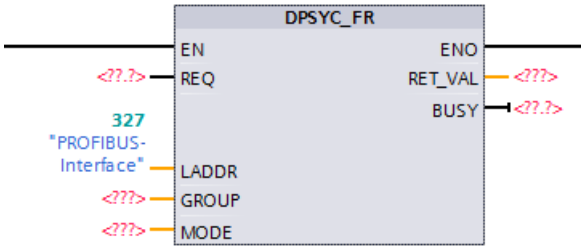
Instrucción avanzada D_ACT_DP: Desactivar/activar esclavo DP o dispositivo PROFINET IO.

El tipo de dato de hardware es Hw_DpSlave o Hw_Device. Los objetos de hardware admisibles para el parámetro de entrada "LADDR" son esclavos DP o dispositivos IO.



Instrucción avanzada DPSYC_FR: Sincronizar/congelar esclavos DP.

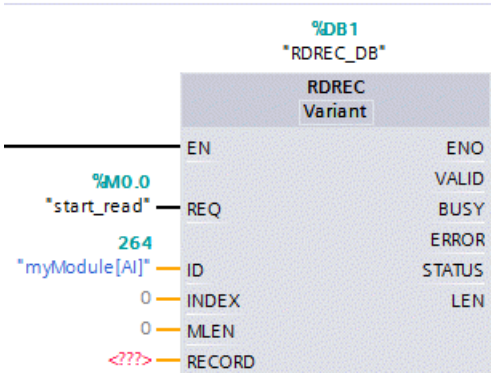
El tipo de dato de hardware es Hw_Interface. El objeto de hardware admisible para el parámetro de entrada "LADDR" es la interfaz de maestro DP.



Instrucción avanzada RDREC: Leer registro.

El tipo de dato de hardware es Hw_SubModule. Los objetos de hardware admisibles para el parámetro de entrada "ID" son módulos o submódulos cuyo registro debe leerse.

Este tipo de datos de hardware se utilizará también con la instrucción avanzada DPRD_DAT.



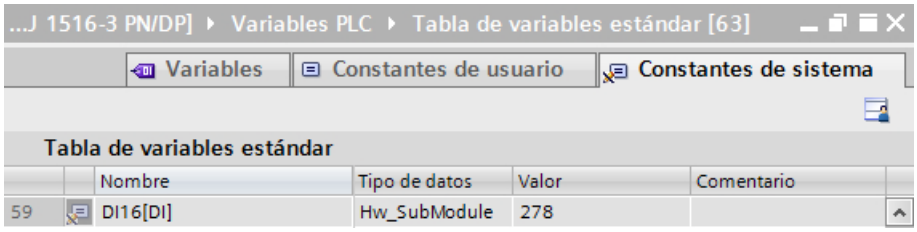
Instrucción avanzada WRREC: escribir registro. La instrucción se utiliza, p. ej., para escribir registros en un módulo o submódulo.

El tipo de dato de hardware es también Hw_SubModule. Los objetos de hardware admisibles para el parámetro de entrada "ID" son módulos o submódulos a los que debe transferirse el registro.

Este tipo de datos de hardware se utiliza también con la instrucción avanzada DPWR_DAT.

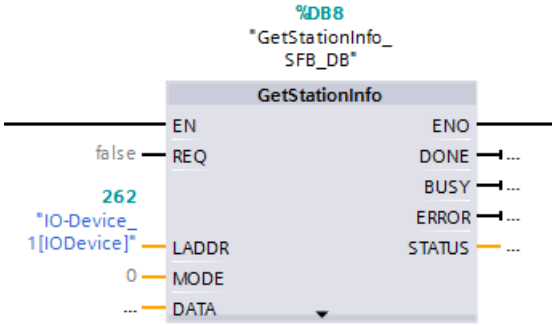
Notas:

- En caso de que un módulo de periferia esté dividido en varios submódulos (p. ej. 1 x 8 AI => 8 x 1AI), debe utilizarse el identificador de hardware de un submódulo. En este caso no debe emplearse el identificador de hardware del módulo.
- En los módulos de interfaz de una unidad de periferia descentralizada a los que debe transferirse un registro para el control de la configuración (configuración futura), debe utilizarse el identificador de hardware del módulo de cabecera [Head]; véase la figura.



Instrucción avanzada Station_Info: leer información de un dispositivo IO.

El tipo de dato de hardware es Hw_Device. El objeto de hardware admisible para el parámetro de entrada "ID" es un dispositivo IO.



Consulte también

Información importante sobre identificadores de hardware (Página 917)

Direcciones de entrada y salida en la vista de direcciones

Introducción

En la vista de direcciones podrán mostrarse, en forma de tabla, las direcciones de entrada y salida actualmente utilizadas. El sinóptico de direcciones se encuentra en "Propiedades" de la CPU, dentro de la ventana de inspección.

Estructura de la vista de direcciones

Podrá ajustar qué objetos deberán mostrarse en la vista de direcciones a través de distintas casillas opcionales:

- Entradas: Visualización de direcciones de entrada
- Salidas: Visualización de direcciones de salida
- Huecos en las direcciones: Visualización de áreas libres de dirección
- Slot: Visualización del número de slot

Normalmente, en la vista de direcciones se muestra la siguiente información:

Encabezado de la tabla	Significado
Tipo	Indica si el área de direcciones es de entrada o salida.
Direc. de	Dirección inicial del área de direcciones.
Direc. hasta	Dirección final del área de direcciones.
Módulo	Módulo desde el que se utilizará el área de direcciones.
IPP	N.º de la memoria imagen parcial de proceso Indica si el OB cíclico está en la memoria imagen parcial de proceso.
OB	Bloque de organización asignado.
DP	Número de sistema maestro. De esta forma puede determinarse qué esclavos están asignados a un maestro. El valor entre paréntesis indica la dirección PROFIBUS del componente de hardware.

Encabezado de la tabla	Significado
PN	Número del sistema IO. El valor entre paréntesis representa el número de dispositivo del componente de hardware.
Rack	Número del rack donde se encuentra insertado el componente hardware.
Slot	Número del slot donde se encuentra insertado el componente hardware en el rack.

Asignar memoria imagen de proceso/memoria imagen parcial de proceso (S7-300, S7-400)

Introducción

Si existen direcciones de entrada y de salida en una memoria imagen de proceso, en S7-300/400 es posible que se encuentre en la memoria imagen de proceso OB1 (MIP OB1) o en una memoria imagen parcial de proceso (IPP).

Una memoria imagen de proceso ofrece la ventaja de que durante el procesamiento cíclico del programa puede accederse a una imagen coherente de las señales de proceso. En S7-300/400 se puede elegir entre "MIP OB1" y "IPP 1" hasta "IPP 15".

Toda dirección de entrada o salida asignada con STEP 7 a una memoria imagen parcial de proceso desaparece de la memoria imagen de proceso OB1 de las entradas y salidas. Solo es posible preseleccionar una vez las direcciones de entrada o salida a través de la memoria imagen de proceso OB1 y de todas las memorias imagen parciales de proceso.

La memoria imagen parcial de proceso se define al asignar direcciones (qué direcciones de entrada o salida de los módulos se llevan a cada memoria imagen parcial de proceso). La memoria imagen parcial de proceso la actualiza el usuario con ayuda de instrucciones, o el sistema automáticamente mediante acoplamiento a un OB. Excepción: El sistema no actualiza las memorias imagen parciales de proceso de los OB de alarma de sincronismo aunque estén acopladas a un OB (OB 61 a OB 64).

Toda memoria imagen parcial de proceso puede actualizarse en el programa de usuario con instrucciones especiales. Utilice "UPDAT_PI (Página 6729)" (SFC 26) para la memoria imagen parcial de proceso de entradas y "UPDAT_PO (Página 6731)" (SFC 27) para la memoria imagen parcial de proceso de salidas.

Parametrice el acoplamiento de una memoria imagen parcial de proceso a un OB en las propiedades de la CPU (área "Alarmas").

Nota

Esta opción sólo puede activarse en módulos con entradas y salidas. Además, la CPU debe ser compatible con memorias imagen parciales de proceso. El número de las memorias imagen parciales de proceso disponibles también depende de la CPU.

Requisitos

- Se encuentra en la vista de dispositivos.
- El módulo es un esclavo DP compacto o está enchufado en alguno de los objetos siguientes:
  - rack central
  - rack de ampliación
  - esclavo DP
- La CPU permite la asignación libre de direcciones.

Procedimiento

Para modificar el área de la memoria imagen de proceso predeterminada, proceda del siguiente modo:

1. En la vista de dispositivos seleccione el módulo cuya dirección inicial desea ajustar.
2. Haga clic en "Direcciones E/S", dentro de "Propiedades", en la ventana de inspección.
3. En "Memoria imagen de proceso" seleccione la memoria imagen de proceso que desee.

Nota

Si en el campo "Memoria imagen de proceso" es visible la entrada "- - -" (no modificable), significa que no hay ninguna memoria imagen de proceso disponible para el área de direcciones indicada.

- En los módulos de entradas y salidas digitales seleccione una dirección inicial menor.
- En caso de que deba direccionarse un módulo analógico a través de una memoria imagen de proceso, seleccione también una dirección inicial menor. La CPU deberá soportar un desplazamiento de la dirección inicial en el área de direcciones para memorias imagen de proceso.
- Si la CPU tiene un tamaño parametrizable de la memoria imagen de proceso, se puede aumentar dicha memoria imagen (propiedades cíclicas de la CPU) para que la dirección indicada vuelva a encontrarse en la memoria imagen de proceso. El parámetro se llama "Tamaño de la memoria imagen de proceso ...".

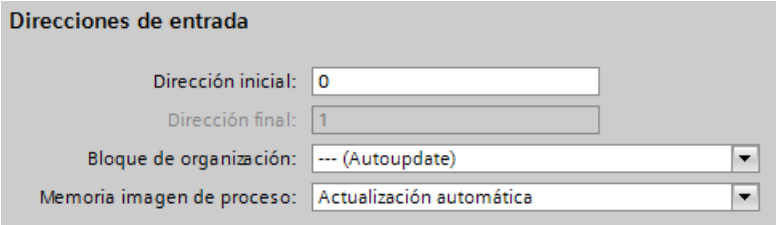
Asignar memoria imagen de proceso/memoria imagen parcial de proceso (S7-1500)

Introducción

Una memoria imagen de proceso ofrece la ventaja de que durante el procesamiento cíclico del programa puede accederse a una imagen coherente de las señales de proceso. En S7-1500 se puede elegir entre "Actualización automática" y "IPP 1" hasta "IPP 31".

Asignación de direcciones al área de memoria imagen de proceso de actualización automática

La memoria imagen de proceso se define al asignar direcciones (ventana de inspección, área "Direcciones E/S"). Si las direcciones de un módulo deben estar en un área de memoria imagen de proceso que el sistema actualice automáticamente en cada ciclo del programa, se deberá seleccionar el siguiente ajuste:



Comparación: En los módulos S7-300/400, este ajuste se llama "memoria imagen de proceso OB1".

Si las direcciones no deben estar en ningún área de memoria imagen de proceso, entonces seleccione "Ninguna" como memoria imagen de proceso.

Asignación de una dirección a una memoria imagen parcial de proceso

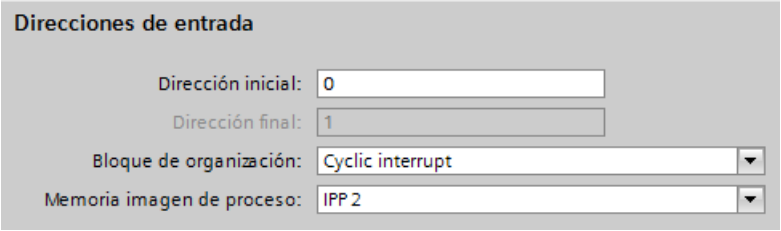
La memoria imagen parcial de proceso se define al asignar direcciones (ventana de inspección, área "Direcciones I/Q"). La memoria imagen parcial de proceso la actualiza el usuario con ayuda de instrucciones, o el sistema automáticamente mediante acoplamiento a un OB.

Acoplamiento de la memoria imagen parcial de proceso a un OB

En el área "Direcciones E/S" de un módulo también se acopla la memoria imagen parcial de proceso a un OB (parámetro "Bloque de organización").

Para otros OB (p. ej. del tipo "Cyclic Interrupt", alarma cíclica) se asigna automáticamente la siguiente memoria imagen parcial de proceso que esté libre, que puede ser modificada por el usuario. Tras el arranque del OB, el sistema actualiza la memoria imagen parcial de proceso asignada para las entradas. Al final del OB, el sistema escribe en las salidas de periferia las salidas de la memoria imagen parcial de proceso asignada. Las memorias imagen parcial de proceso están excluidas de la actualización automática.

Ejemplo de acoplamiento de la memoria imagen parcial de proceso 2 (TPA 2) en una alarma cíclica:



Excepción: El sistema no actualiza las memorias imagen parciales de proceso de los OB de alarma de sincronismo aunque estén acopladas a un OB (OB 61 a OB 64).

La memoria imagen (parcial) de proceso asignada se muestra en las propiedades del OB asignado (no modificable).

Actualización de una memoria imagen parcial de proceso en el programa de usuario

Toda memoria imagen parcial de proceso puede actualizarse en el programa de usuario con instrucciones especiales. Utilice "UPDAT_PI" para la memoria imagen parcial de proceso de las entradas y "UPDAT_PO" para la memoria imagen parcial de proceso de las salidas.

Requisitos

- Se encuentra en la vista de dispositivos.
- El módulo o el submódulo tiene direcciones I/Q.

Procedimiento

Para modificar el área de la memoria imagen de proceso predeterminada, proceda del siguiente modo:

1. En la vista de dispositivos, seleccione el módulo cuya dirección inicial desee ajustar.
2. Haga clic en el área "Direcciones E/S", dentro de "Propiedades", en la ventana de inspección.
3. En "Memoria imagen de proceso" seleccione la memoria imagen de proceso que desee.

Utilización de memorias imagen parcial de proceso en instrucciones

Introducción

Las instrucciones "UPDAT_PI", para actualizar entradas de la memoria imagen parcial de proceso, y "UPDAT_PO", para visualizar sus salidas en salidas de periferia, utilizan el parámetro de entrada "PART" para introducir la memoria imagen parcial de proceso.

A continuación se muestra la forma de definir una constante de usuario para una memoria imagen parcial de proceso y cómo utilizarla como entrada en el programa de usuario. Si se utilizan diversas memorias imagen parciales de proceso para configurar la periferia, los símbolos facilitan la programación.

Requisitos

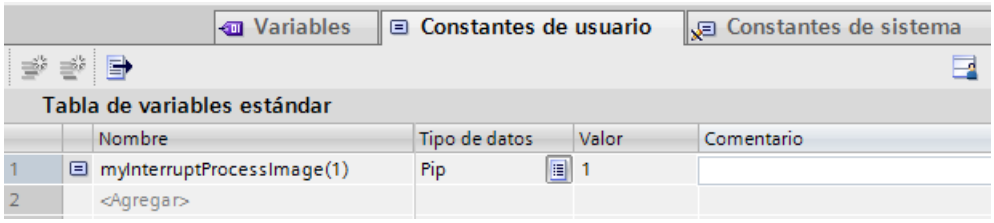
Se ha configurado la estructura de periferia, se han asignado direcciones y memorias imagen parciales de proceso y se han creado los bloques necesarios.

Para este ejemplo se ha seleccionado el lenguaje de programación KOP.

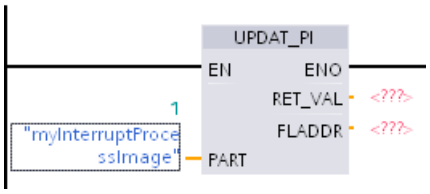
Procedimiento

Proceda del siguiente modo:

1. Abra la tabla de variables e introduzca constantes del tipo de datos "PIP" (Process image part) en la ficha "Constantes de usuario". Asigne un nombre a cada memoria imagen parcial de proceso utilizada, y si es necesario, también un comentario.



2. Abra el editor de programas (p. ej. haciendo doble clic en un OB en el árbol del proyecto).
3. Arrastre la instrucción UPDAT_PI al segmento con Drag & Drop.
4. Haga doble clic en el parámetro de entrada PART para abrir la selección de variables. En la selección de variables solo pueden seleccionarse constantes de la memoria imagen parcial de proceso.
5. Seleccione la memoria imagen parcial de proceso que desee.



Asignar direcciones a una ubicación del programa

Es posible asignar direcciones desde los canales E/S de los módulos directamente a las ubicaciones del programa o a una tabla de variables.

Requisitos

- Está abierta la vista de dispositivos del editor de hardware y redes.
- El factor de zoom en la vista de dispositivos debe ser del 200 % como mínimo para que se vean los diferentes canales E/S.
- Están abiertas la ventana de instrucciones del editor de programación o una tabla de variables.

Procedimiento

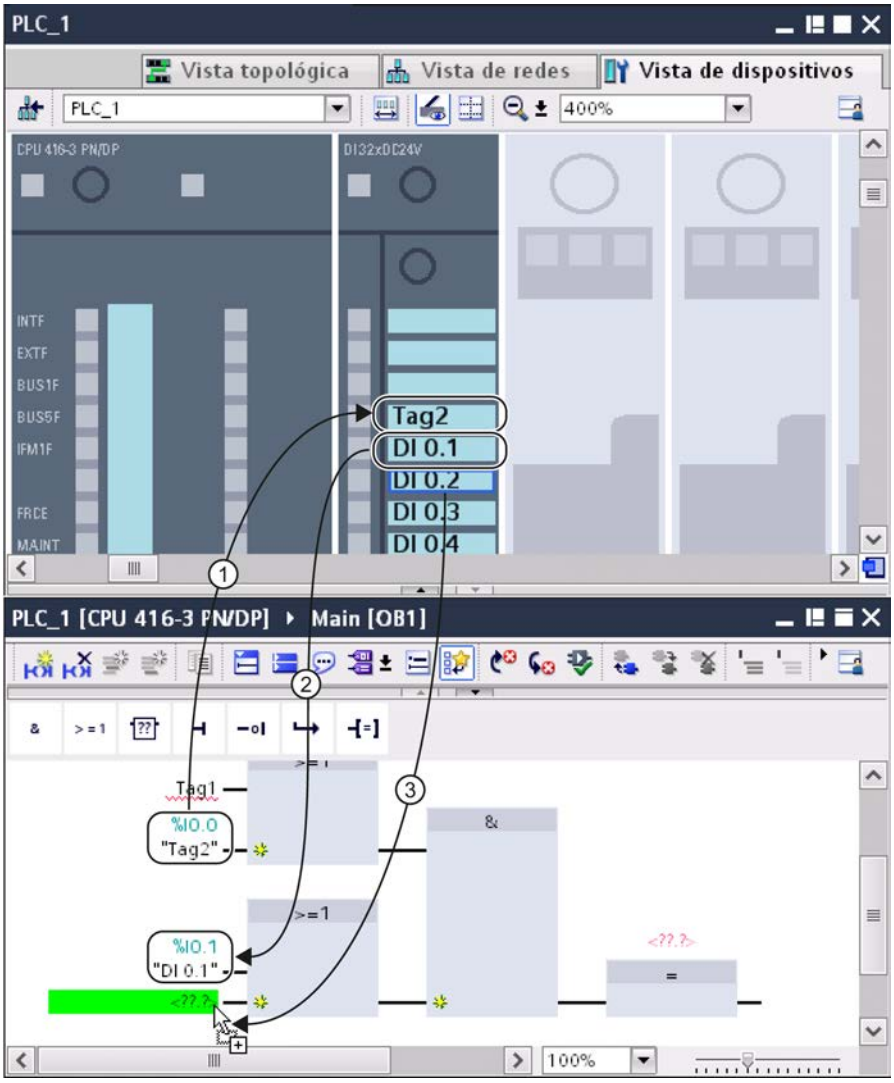
Para asignar canales E/S de módulos a las ubicaciones del programa o a una tabla de variables, proceda del siguiente modo:

1. En la vista de dispositivos, navegue hasta el módulo que tenga el canal E/S deseado.
2. Manteniendo pulsado el botón del ratón, arrastre la dirección E/S deseada hasta la ubicación correspondiente del bloque o hasta la tabla de variables.

La dirección del módulo se asignará a la ubicación del programa, o se introducirá como variable en la tabla de variables.

Nota

La variable para una entrada o salida de un bloque también puede arrastrarse hasta la entrada o la salida de un módulo para relacionar lógicamente la variable con el canal E/S del módulo.



① La ubicación de un bloque se ha arrastrado hasta un canal E/S.

② El nombre de un canal E/S se ha arrastrado hasta una ubicación del programa.

③ Manteniendo pulsado el botón del ratón, arrastre desde el nombre del canal E/S hasta la entrada de un bloque.

Consulte también

Sinopsis de las tablas de variables PLC (Página 2566)

Control de la configuración

Información importante sobre el control de la configuración

Principio de funcionamiento

El control de la configuración permite configurar la estructura de un controlador (en este caso: instalación centralizada de una S7-1500) y utilizar variantes (opciones) que difieran de esta configuración.

Mediante un registro de control (programado por el usuario) en el programa de arranque, se notifica a la CPU qué módulos faltan en la estructura real en comparación con la configuración o bien qué módulos se encuentran en un slot distinto al previsto en la configuración. El control de la configuración no afecta a la parametrización de los módulos.

El control de la configuración permite variar la instalación centralizada de forma flexible siempre que la configuración real derive de la configuración máxima prevista.

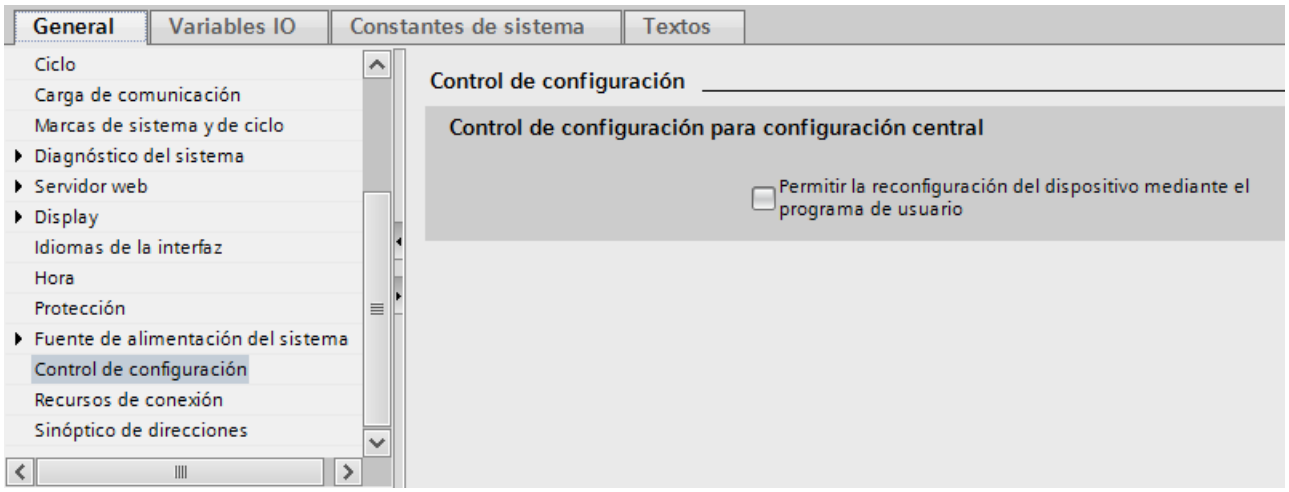
A continuación se describe la manera de activar el control de la configuración (parametrización de CPU) y la estructura que debe tener el registro necesario.

Requisitos

- STEP 7 Professional versión V13 o superior
- CPU S7-15XX versión de firmware V1.5 o superior
- Recomendación: Antes de cargar un programa nuevo con un registro de control modificado, realice un borrado total. Con esta medida se evitan errores que podrían producirse debido a la existencia de un registro de control incompatible.

Pasos necesarios

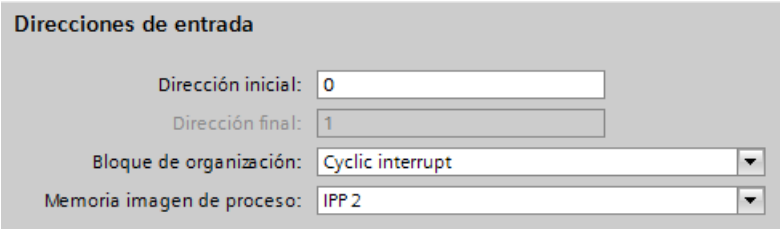
- Active en la configuración de la CPU el parámetro "Permitir la reconfiguración del dispositivo en el programa de usuario" (apartado "Control de configuración").



- Cree un registro de control (p. ej. en un bloque de datos) de acuerdo con la configuración actual siguiendo la muestra de registro de control descrita más abajo. El registro de control tiene el número 196.

		Nombre	Tipo de datos	Valor ...	Remane...	Accesi...	Visibl...	Valor...	Comentario
1		Static							
2		Block_length	Byte	9	☑	☑	☑	☐	4+number of slots
3		Block_ID	Byte	196	☑	☑	☑	☐	Data record number
4		Version	Byte	4	☑	☑	☑	☐	
5		Subversion	Byte	0	☑	☑	☑	☐	
6		slot_0	Byte	16#0	☑	☑	☑	☐	configured slot 0 < real slot 0
7		slot_1	Byte	16#1	☑	☑	☑	☐	configured slot 1 < real slot 1
8		slot_2	Byte	16#2	☑	☑	☑	☐	configured slot 2 < real slot 2
9		slot_3	Byte	16#FF	☑	☑	☑	☐	configured slot 3 < not used
10		slot_4	Byte	16#3	☑	☑	☑	☐	configured slot 4 < real slot 3

- Llame la instrucción avanzada WRREC (escribir registro) en el OB de arranque y transfiera el registro de control creado.



Si en el OB de arranque no se transfiere ningún registro de control válido, el control de la configuración no estará operativo. En este caso, la CPU interrumpe el arranque y vuelve al estado "STOP".

Disposición de los módulos

La siguiente tabla muestra la asignación de los números de slot:

Slot	Módulos	Observación
0	Módulo de alimentación (opcional)	delante de la CPU
1	CPU	El slot 1 es siempre la CPU
2 - 31	Módulos de periferia / módulos de alimentación del sistema, según la variante de configuración	detrás de la CPU

Registro de control

Para el control de configuración se define un registro de control 196 que contiene una asignación de slots.

Byte	Elemento	Codificación	Explicación
0	Longitud del bloque	4 + número de slots	Encabezado
1	ID de bloque	196	
2	Versión	4 (para periferia centralizada)	
3	Versión	0	
4	Asignación de slot 0 configurado	Slot 0 real	Elemento de control Describe en cada elemento qué slot real del dispositivo está asignado al slot configurado.
5	Asignación de slot 1 configurado	Slot 1 real (siempre 1, ya que la CPU siempre ocupa el slot 1)	
:	:	:	
4 + (n.º máx. slot)	Asignación del máximo slot configurado	Slot real	

Elemento de control

Cada elemento del registro de control debe incluir la siguiente información acerca del slot:

- Asignación de slot configurado ⇔ slot real

Byte	Bit	Significado
Asignación del slot configurado (p. ej., en byte 5)	0 a 7	11111111: Módulo no presente 00000000 hasta slot máximo: Slot real

Reglas

Tenga en cuenta las siguientes reglas:

- En la configuración centralizada no debe estar enchufado ningún CP/CM, es decir, ningún CM 1542-5 (maestro DP o esclavo DP), por ejemplo. Si se ha configurado un módulo de este tipo y se ha activado el control de configuración, no será posible cargar la configuración.
- Si se ha activado el control de la configuración, la CPU no estará operativa sin registro de control. Si en el OB de arranque no se transfiere ningún registro de control válido, la CPU interrumpe el arranque y vuelve al estado "STOP". En este caso no se inicializa la periferia centralizada. La causa del estado operativo "STOP" se registra en el búfer de diagnóstico.
- Para direccionar la instrucción WRREC, utilice el identificador de hardware 33 (decimal, para el parámetro de bloque LADDR) para escribir el registro de control.
  Encontrará los identificadores de hardware, p. ej., en **Variables PLC > Tabla de variables estándar > Constantes de sistema** o en la ficha "Constantes de sistema" de la ventana de inspección de la CPU.
- El registro de control se almacena de forma remanente en la CPU, de modo que si la configuración no varía, no es necesario escribir de nuevo el registro de control 196 al volver a arrancar. Antes de la puesta en marcha se recomienda efectuar un borrado total de la CPU a fin de borrar cualquier registro de control que pueda haber.
- Las entradas de slots en el registro de control que no estén en la configuración teórica configurada se ignoran.
- El registro de control se puede acortar. Deben estar incluidas las entradas hasta el último slot de la configuración teórica actual.
- Cada slot real solamente debe existir una vez en el registro de control.
- Un slot real solamente puede asignarse a un slot configurado.
- Los módulos de alimentación del sistema (PS) también pueden estar sometidos al control de la configuración. Para ello, tenga en cuenta las indicaciones del manual de sistema S7-1500 (http://support.automation.siemens.com/WW/view/es/59191792).
  Recomendamos prescindir de la reconfiguración, en especial para un módulo de alimentación del sistema (PS) en el slot 0.

Nota

Configuración modificada

La escritura de un registro de control con la configuración modificada provoca automáticamente la siguiente reacción de la CPU: borrado total y, a continuación, arranque con esta configuración modificada.

Con esta reacción se borra el registro 196 original guardado de forma remanente y se guarda de forma remanente el nuevo registro 196.

Comportamiento durante el funcionamiento

Repercusión de la discrepancia entre la configuración programada y la configuración real:

- Para la visualización online y la visualización en el búfer de diagnóstico (módulo en buen estado o módulo defectuoso) se utiliza siempre la configuración hardware, no la configuración real diferente de esta.
  Ejemplo: un módulo proporciona un diagnóstico. Este módulo está configurado para el slot 4, pero está enchufado realmente en el slot 3 (módulo ausente; ver ejemplo en el capítulo siguiente). En la vista online se muestra un slot 4 configurado como erróneo; en la estructura real, el módulo del slot 3 señaliza un error mediante un indicador LED.

Si en el registro de control se han introducido módulos como ausentes, el sistema de automatización se comporta de la siguiente manera:

- Los módulos marcados como no presentes en el registro de control no proporcionan ningún diagnóstico, y su estado siempre es OK. La información de calidad es OK.
- Acceso directo de escritura a las salidas o acceso de escritura a la memoria imagen de proceso de las salidas no presentes: continúa sin efecto; no se notifica ningún error de acceso.
- Acceso directo de lectura a las entradas o acceso de lectura a la memoria imagen de proceso de las entradas no presentes: se proporciona el valor "0"; no se notifica ningún error de acceso.
- Escribir registro en módulo no presente: continúa sin efecto; no se notifica ningún error.
- Leer registro de módulo no presente: se notifica un error, ya que no se puede devolver ningún registro válido.

Mensajes de error

Al escribir el registro de control, en caso de fallo se devuelven los siguientes mensajes de error:

Tabla 8-58 Mensajes de error

Código de error	Significado
$80B1_H$	Longitud no permitida; la indicación de longitud del registro 196 no es correcta.
$80B5_H$	Control de la configuración no parametrizado.
$80E2_H$	El registro se ha transferido a un contexto de OB erróneo. El registro debe transferirse en el programa de arranque.
$80B8_H$	Error de parámetros; el módulo notifica parámetros no válidos.

Consulte también

Manual de sistema S7-1500 (http://support.automation.siemens.com/WW/view/es/59191792)

Ejemplo de un control de la configuración

A continuación se programa una configuración compuesta por una fuente de alimentación del sistema, una CPU y 3 módulos de periferia.

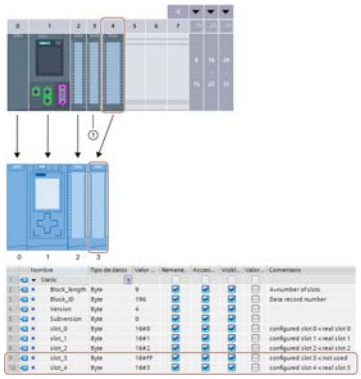
El módulo del slot 3 no está presente en el primer nivel de configuración y el control de la configuración lo "oculta".

En el segundo nivel de configuración, el módulo ocultado anteriormente se coloca en el último slot. El slot agregado se notifica a la CPU a través de un registro de control modificado.

Configuración real con módulo ausente

La configuración programada incluye todos los módulos que pueden estar presentes en una configuración final.

El módulo insertado en el slot 3 en la configuración programada falta en la configuración real. Por este motivo, el slot 3 debe codificarse en el registro de datos con "FF $_H$" (= no presente) de forma correspondiente.

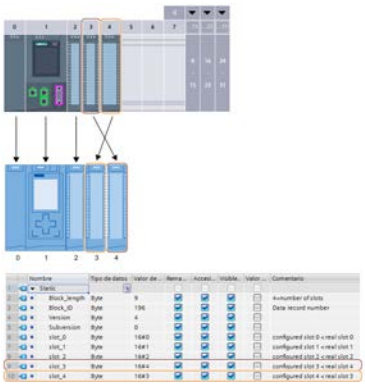


① El módulo falta en la configuración real

Configuración real con módulo agregado a posteriori

El módulo presente en la configuración programada en el slot 3 se "adjunta al final" en la configuración real; queda insertado como último módulo en el slot 4.

El registro de control se adapta de forma correspondiente.



Configuraciones para el servidor web

Información importante sobre el servidor web

Introducción

El servidor web permite a los usuarios autorizados vigilar y administrar la CPU a través de una red. Eso permite la evaluación y el diagnóstico a una gran distancia. Para ello solo se necesita un navegador web.

Los avisos y la información sobre el estado se muestran en páginas HTML.

Navegador web

Para acceder a las páginas HTML de la CPU se requiere un navegador web.

Los siguientes navegadores web se han probado para la comunicación con la CPU:

- Internet Explorer (versión 8)
- Mozilla Firefox (versión 21)
- mobileSafari (iOS5)

Leer información acerca del servidor web

La información detallada a continuación podrá leerse de la CPU. La disponibilidad de las respectivas páginas web depende de la CPU y de su versión de firmware.

En los siguientes manuales encontrará información detallada sobre cada una de las páginas web y sobre el volumen exacto de la información:

Documentación del S7-300 (http://support.automation.siemens.com/WW/view/es/12996906)

Documentación del S7-400 (http://support.automation.siemens.com/WW/view/es/44444467)

Documentación de la S7-1500 (http://support.automation.siemens.com/WW/view/es/59193560)

Información	Descripción
Página de inicio con información general acerca de la CPU	La página inicial incluye un resumen de la información general de la CPU, el nombre de la CPU, el tipo de CPU e información básica sobre el estado operativo actual.
Información identificativa	Visualización de la información identificativa estática como el número de serie, referencia y versión.
Contenido del búfer de diagnóstico	Visualización del contenido del búfer de diagnóstico con las entradas más recientes en primer lugar.
Estado del módulo	Visualización que muestra si los componentes de una estación están en orden, si p. ej., hay una petición de mantenimiento o si los componentes no son accesibles.

Información	Descripción
Avisos (sin posibilidad de acuse)	Visualización de mensajes de texto orientados a un proyecto con filtro y posibilidad de clasificación.
Información sobre la comunicación	Visualización de las conexiones de comunicación; visualización de los recursos; parámetros de interfaz y estadística de puertos.
Topología	Visualización de la topología con las vistas gráfica, de estado y en forma de tabla para dispositivos PROFINET conectados.
Estado de variables	Visualización del estado de operandos del programa de usuario para la observación de los valores.
Tablas de variables	Visualización de la tabla de variables generada con STEP 7 con la opción "Servidor web".
Páginas de usuario (en caso de que se hayan configurado y cargado páginas web definidas por el usuario)	Las páginas web del usuario proporcionan una lista de las páginas web con aplicaciones web específicas del cliente.
Explorador de archivos	S7-1500: el explorador de archivos lista archivos y directorios que se encuentran en la SIMATIC Memory Card. Según del tipo que sean, los archivos pueden descargarse, borrarse, renombrarse y cargarse. Los directorios solo pueden crearse, borrarse o renombrarse. Excepción: Los archivos de sistema no se muestran en el explorador de archivos, así que no pueden modificarse ni borrarse. Los archivos de sistema son el archivo de tareas y todos los directorios especiales (incluyendo su contenido) a los que se hace referencia en el archivo de tareas.

Acceso web a la CPU mediante PG/PC

Para acceder al servidor web, proceda de la siguiente manera:

1. Conecte el cliente (la PG/el PC) con la CPU a través de la interfaz PROFINET.
   Si no se accede a través de la interfaz PROFINET de la CPU, sino a través de la interfaz de un CP (p. ej. CP 1543-1), las páginas web tardan más en generarse. En tal caso, asegúrese de que la configuración de caché (archivos temporales de Internet) del navegador se encuentre en modo "Automático".
2. Abra el navegador web.
   En el campo "Dirección" del navegador web, introduzca la dirección IP de la CPU de la manera siguiente: http://ww.xx.yy.zz (entrada a modo de ejemplo: http://192.168.3.141).
   Se abrirá la página de inicio de la CPU. A partir de la página de inicio puede navegar hasta el resto de la información.
   Si no aparece la página de inicio, asegúrese de que se ha activado el servidor web en la configuración y que la configuración ha sido cargada.

Acceso web a una CPU S7-300/400 a través de dispositivos HMI y dispositivos móviles

El servidor web también soporta el servicio de Windows desde un terminal, de modo que, además de la programadora y el PC, también pueden utilizarse soluciones Thin-Client con dispositivos móviles (p. ej. PDA, MOBIC T8) y estaciones locales robustas (p. ej. SIMATIC MP370 con la opción ThinClient/MP) con Windows CE.

Para acceder al servidor web, proceda de la siguiente manera:

1. Conecte el cliente (dispositivo HMI, PDA) con la CPU a través de la interfaz PROFINET.
2. Abra el navegador web.
   En el campo "Dirección" del navegador web, introduzca la dirección IP de la CPU de la manera siguiente: http://ww.xx.yy.zz /basic (entrada a modo de ejemplo: http://192.168.3.141/basic).
   Se abrirá la página de inicio de la CPU. Desde allí podrá acceder al resto de las informaciones.

Para dispositivos HMI con el sistema operativo Windows CE, anterior a la V 5.x, la información de la CPU se edita en un navegador diseñado especialmente para Windows CE. En este navegador, la información se representa de forma simplificada.

Nota

SIMATIC Micro Memory Card en combinación con el servidor web (S7-300)

Los datos de configuración del servidor web se guardan en la SIMATIC Micro Memory Card. Por este motivo se recomienda el uso de una SIMATIC Micro Memory Card de al menos 512 KB.

También puede utilizar el servidor web sin una SIMATIC Micro Memory Card insertada. Para que funcione debe haber asignado una dirección IP a la CPU.

- El contenido del búfer de diagnóstico se visualiza en código hexadecimal.
- La página de inicio, la información sobre identificación y comunicación y el estado de las variables se muestran en texto explícito.
- Los indicadores siguientes permanecen vacíos:
  - Topología
  - Avisos
  - Estado del módulo
- La actualización automática de la página también está activada sin la SIMATIC Micro Memory Card o sin configuración.

Acceso web a una CPU S7-1500 a través de dispositivos HMI y dispositivos móviles

Para dispositivos HMI con el sistema operativo Windows CE preinstalado, anterior a la V 5.x, la información de la CPU se edita en un navegador diseñado especialmente para Windows CE. En este navegador, la información se representa de forma simplificada.

El servidor web ofrece páginas web de complejidad reducida que puedan ser útiles para dispositivos con pantallas pequeñas y baja potencia de procesador.

Requisito para el acceso con dispositivos móviles:

- conexión WLAN para la CPU (WLAN Access Point, p. ej. SCALANCE W788-1RR o SCALANCE W784-1)
- servidor web activado en la CPU
- Para el dispositivo móvil, la WLAN debe estar activada y se debe haber establecido una conexión con el punto de acceso.

Para un dispositivo móvil, el procedimiento es idéntico al de otros dispositivos (programadora/PC). Introduzca en el explorador la dirección IP de la interfaz de la CPU, descargue el certificado e instálelo a continuación.

Seguridad

Para evitar riesgos, asegúrese de emplear distintas técnicas para dejar protegida la CPU (p. ej. restringir el acceso a la red o usar firewalls).

El servidor web ofrece las funciones de seguridad siguientes:

- Acceso a través del protocolo de transmisión seguro https
- Permiso de usuario configurable mediante la lista de usuarios

Consulte también

Trabajar con proyectos multilingües (Página 348)

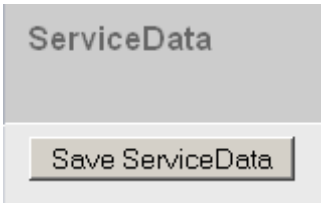
Condiciones previas para el acceso web (Página 940)

Lectura de datos de servicio

El servidor web de una CPU S7-1500 le ofrece la posibilidad de guardar los datos de servicio. Estos contienen, junto con el contenido del búfer de diagnóstico, otras muchas informaciones sobre el estado interno de la CPU. Si ocurriese con la CPU un problema que no pudiera solucionarse de otra forma, tiene la posibilidad de proporcionar los datos de servicio al equipo de Service & Support.

Procedimiento

1. Introduzca la dirección siguiente en la barra de direcciones de su navegador web: "http://<CPU IP address>/save_service_data.html", p. ej. "http://192.168.3.141/save_service_data.html"
2. En su pantalla aparecerá la vista de la página de datos de servicio con un botón para guardarlos.



3. Guarde los datos de servicio localmente en su visualizador haciendo clic en "Save ServiceData".

Resultado

Los datos se guardarán en un archivo .dmp con la siguiente convención de nomenclatura: "<MLFB><número de serie><sello de tiempo>.dmp". Podrá modificar posteriormente el nombre del archivo si lo desea.

Páginas web estándar

Condiciones previas para el acceso web

A continuación se explicarán las condiciones para el acceso a las páginas web estándar de la CPU así como las repercusiones la ausencia o presencia de información de configuración.

Requisitos

El servidor web tiene que estar abierto.

El servidor web se inicia al activarlo en el área "Servidor web" de las propiedades de la CPU.

Considere lo siguiente:

- Las páginas web se transfieren a través de una conexión no segura que no está exenta del peligro de ataques de terceros. Si desea transferir las páginas web codificadas al navegador, active la opción "Permitir el acceso sólo vía HTTPS" en las propiedades de la CPU. Tenga en cuenta que la URL de la CPU comienza, en este caso, con https://.
- El acceso vía HTTPS sólo funciona si se ha ajustado la hora en la CPU.
- Para una mayor protección de acceso, cree usuarios en el área "Servidor web" de las propiedades de la CPU.
- La funcionalidad completa de las páginas web relativas al estado del módulo, la topología y los avisos requiere en las CPU S7-300/400 la generación y carga de los datos de diagnóstico del sistema (Report System Error). Tras activar el servidor web aparece por tanto la consulta sobre si se desea activar el diagnóstico del sistema. Se recomienda que confirme la consulta con "Sí".

Iniciar sesión

No es necesario iniciar sesión para poder acceder a las páginas web estándar. Para realizar determinadas acciones como la modificación del estado operativo de la CPU o para acceder en escritura, debe haber un usuario registrado. Los campos de entrada para iniciar sesión son los que se encuentran en la esquina superior izquierda de todas las páginas web estándar.

Cuando se inicia sesión como usuario deben introducirse el nombre de usuario y la contraseña. El nombre de usuario y la contraseña se definen en las propiedades de la CPU, en el área "Servidor web".

Acceso web sin información de configuración

Sin configuración, sólo se dispone de información básica en las páginas web estándar. Esta información está disponible en varios idiomas:

- Las entradas del búfer de diagnóstico no se mostrarán como texto explícito sin información de configuración, sino sólo como código hexadecimal.
- En caso de que el servidor web de la CPU soporte tablas de variables (tablas de observación):
  La página de las tablas de variables no muestra ninguna variable sin configuración. El estado de las variables puede consultarse siempre.

Acceso web con información de configuración

En la información de configuración, los textos del idioma del país seleccionado contienen:

- Textos para las entradas del búfer de diagnóstico como texto breve o extenso
- Para CPU S7-300/400: Textos de aviso. p. ej., bloques de aviso AlarmS/SQ, AlarmD/DQ
- En caso de que se soporten las tablas variables (tablas de observación): Tablas de variables con los símbolos y comentarios contenidos

Los textos para las entradas del búfer de diagnóstico y mensajes se mostrarán durante el funcionamiento en el idioma ajustado en el navegador web.

Consulte también

Ajustes para el funcionamiento (Página 941)

Ajustes para el funcionamiento

Ajustes para el funcionamiento

Para poder utilizar el servidor web de una CPU y para determinar el comportamiento del servidor web, deberá marcar la CPU en la vista de redes, de dispositivos o de topología.

Realice los siguientes ajustes en la ventana de inspección bajo "Propiedades > General > Servidor web":

- Activar el servidor web
- Limitar el acceso a la web al protocolo seguro de transferencia de hipertexto (HTTPS)
- Activar la actualización automática
- Ajustar el idioma para web
- Completar la lista de usuarios
- Para CPU S7-300/400: Seleccionar los discriminadores de los avisos

Cada interfaz PROFINET de una CPU S7-1500, de un CP PROFINET o de un CM PROFINET puede ser asimismo autorizada o bloqueada en el servidor web para el acceso "desde fuera". Lo más sencillo es que se administre este acceso centralmente en las propiedades de la CPU. Para ello, haga clic en la ficha "Propiedades" de la ventana de inspección en el área "Servidor

web > Vista general de las interfaces" y active las interfaces a través de las que se permite el acceso.

Consulte también

Principios básicos de los textos del proyecto (Página 348)

Condiciones previas para el acceso web (Página 940)

Activar el servidor web

Activar el servidor web

En el ajuste básico, el servidor web permanece desactivado. Para que la CPU muestre páginas web deberán activarse las correspondientes casillas opcionales.

Este ajuste se encuentra, dentro de la ventana de inspección, en "Propiedades > General > Servidor web"; casilla opcional "Activar el servidor web en este módulo".

Cuando se activa el servidor web aparece para las CPU S7-300/400 un mensaje de consulta sobre si el diagnóstico de sistema (Report System Errors) debe o no activarse. Haga clic en "Sí" cuando deban generarse los bloques y textos necesarios para "Report System Errors", de manera que los textos de aviso estén disponibles en el navegador web durante el funcionamiento. En el caso de las CPU S7-1500, el diagnóstico de sistema siempre está activado (no puede desactivarse).

En las CPU S7-1500 con varias interfaces PROFINET o si se configuran CP con interfaces PROFINET, deberá activar adicionalmente las interfaces a través de las que se debe permitir el acceso al servidor web. Este ajuste lo encontrará en "Propiedades > General > Servidor web > Vista general de las interfaces".

Activar la actualización automática

Activar la actualización automática

Con la casilla opcional "Actualización automática" activada, las páginas web se actualizarán automáticamente en el intervalo de tiempo establecido en "Intervalo de actualización". La página web "Identificación" está excluida de la actualización.

Para activar la actualización automática, proceda del siguiente modo:

1. En "Propiedades > General > Servidor web > Actualización automática", active la casilla de verificación "Activar actualización automática".
2. Indique el intervalo de actualización.

Nota

Tiempo de actualización

El intervalo de actualización ajustado es el tiempo de actualización más corto. Un número mayor de datos o más conexiones HTTP/HTTPS incrementa el tiempo de actualización.

Ajustar el idioma para web

Ajustar el idioma para web

Para seleccionar el idioma de los textos dependientes del mismo que deban cargarse en la CPU, proceda del siguiente modo:

1. Navegue en la ventana de inspección hasta el área "Idiomas" desde "Propiedades > General > Servidor web > Idiomas".
2. Active los idiomas necesarios para la visualización en el navegador web en la primera columna; el número de idiomas seleccionable es específico de la CPU.
3. Asigne a los idiomas activados del servidor web un idioma de proyecto.
   Los idiomas de proyecto asignables deben estar activados y los textos correspondientes (traducciones) deben estar en el proyecto. Podrá encontrar la selección del idioma del proyecto en la navegación del proyecto, más concretamente en "Idiomas y recursos". Mediante estas asignaciones los textos de visualización no disponibles en la CPU se cargan en el idioma seleccionado (con código de país e idioma) al cargarlos en la CPU.

Enable	Idioma del servidor web	Asignar idioma del proyecto
☐	German	Ninguno
☐	English	Ninguno
☐	French	Ninguno
☐	Spanish	Ninguno
☐	Italian	Ninguno

Nota

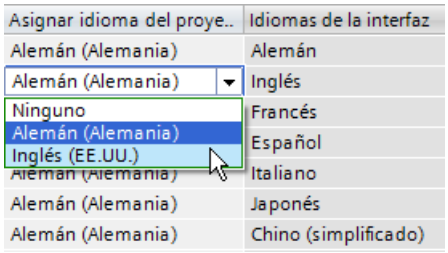
Si activa el servidor web y no asigna ningún idioma, los avisos y la información de diagnóstico se visualiza en código hexadecimal.

Ajustar el idioma para web

Ajustar el idioma de la interfaz para web

Para seleccionar el idioma de los textos dependientes del mismo que deban cargarse en la CPU, proceda del siguiente modo:

1. Con la CPU seleccionada, vaya en la ventana de inspección hasta el área "Idiomas de la interfaz" ("Propiedades > General > Idiomas de la interfaz").
2. Asigne a cada idioma de la interfaz un idioma de proyecto.
   Los idiomas de proyecto asignables deben estar activados y los textos correspondientes (traducciones) deben estar en el proyecto. Podrá encontrar la selección del idioma del proyecto en la navegación del proyecto, más concretamente en "Idiomas y recursos". Mediante estas asignaciones los textos de visualización no disponibles en la CPU se cargan en el idioma seleccionado (con código de país e idioma) al cargarlos en la CPU.



Nota

El número de idiomas cargables para el servidor web está limitado específicamente para la CPU. Si, por ejemplo, es posible tener dos idiomas, asigne en la columna "Idioma del servidor web" el correspondiente idioma del proyecto a su lengua materna. En el resto de filas, asigne al "Idioma del servidor web" en todos los casos el mismo idioma de sustitución.

Lista de usuarios

Administración de usuarios

La administración de usuarios ofrece las siguientes posibilidades:

- Crear usuarios
- Establecer derechos de ejecución
- Asignar contraseñas

Con esta asignación los usuarios disponen de las opciones asignadas de forma fija a los derechos de ejecución.

- Si se ha configurado un usuario, tras registrarse con su contraseña, éste podrá acceder a las páginas web conforme a sus derechos de acceso.
- Dependiendo de la CPU y el firmware utilizados, se pueden asignar diferentes derechos de usuario. Los derechos que su CPU no soporta no pueden seleccionarse.
- En la lista de usuarios se ha incluido de forma predeterminada un usuario con el nombre "Cualquiera" que posee derechos de acceso mínimos: acceso en lectura a las páginas Intro y de inicio. El usuario "Cualquiera" se ha configurado sin asignación de contraseña y no puede modificarse.

Acceso para HTTPS

Acceso sólo vía HTTPS

El HTTPS sirve para la encriptación y autenticación de la comunicación entre el navegador y el servidor web.

Para limitar la transferencia de datos entre el navegador y la CPU exclusivamente a través del protocolo HTTPS, active la opción "Acceso sólo vía HTTPS" en las propiedades de la CPU "Propiedades > General > Servidor web".

Requisitos

Para un acceso HTTPS a la CPU sin errores se aplican los siguientes requisitos:

- En la CPU debe haberse ajustado la hora actual
- un certificado válido e instalado.

De no haberlo, se mostrará un mensaje con la recomendación de no utilizar la página. Para poder ver la página deberá "Agregar excepción".

Obtendrá un certificado válido (Certification Authority) descargándolo de la página web "Intro", en "Download certificate". En la ayuda de su navegador encontrará la información necesaria para instalar dicho certificado.

Ejemplo del acceso a páginas web con el protocolo HTTPS

Para acceder a las páginas web de la CPU con el protocolo HTTPS, introduzca la URL del modo siguiente: https://ww.xx.yy.zz, donde la dirección IP de la CPU es ww.xx.yy.zz .

Entrada a modo de ejemplo: https://192.168.3.141

Una conexión encriptada se distingue por el símbolo de la cerradura en la barra de estado de la página web.

Clases de visualización de los mensajes

Clases de visualización de los mensajes

En el ajuste básico, todas las clases de visualización de mensajes están activadas. Los mensajes de las clases seleccionadas se muestran durante el funcionamiento en la página

web "Avisos". Los mensajes de las clases no seleccionadas no aparecen como texto completo sino en código hexadecimal.

Para configurar las clases de visualización, proceda de la manera siguiente:

- para "Report System Errors" vaya, en la ventana de inspección, a "Propiedades > General > Diagnóstico de sistema"
- para mensajes relativos a los módulos en el editor de mensajes

Nota

Reducir la necesidad de memoria

Puede reducir la necesidad de memoria para la configuración del servidor web seleccionando únicamente las clases de visualización de los mensajes que deban utilizarse.

Crear y cargar páginas web definidas por el usuario

Información importante sobre las páginas web definidas por el usuario

Concepto

El concepto de páginas web definidas por el usuario permite acceder con un navegador web a las páginas web de la CPU diseñadas por el usuario. El servidor web de la CPU pone a disposición esta función.

Para configurar la apariencia y funcionalidad de las páginas web definidas por el usuario no se requieren herramientas especiales. Puede utilizarse CSS para adaptar el diseño de las páginas, Javascript para incluir contenido dinámico o cualquier otro framwork que el usuario desee para crear las páginas web.

El conjunto de archivos que procesa el servidor web también recibe el nombre de "aplicación web".

Aplicación web y programa de usuario

Utilizando código HTML especial en páginas web definidas por el usuario, también pueden transferirse datos al programa de usuario de la CPU a través de un navegador web para su posterior procesamiento, así como visualizarse datos del área de operandos de la CPU en el navegador web.

Con instrucciones de script (p. ej. Javascript) puede optimizar sus páginas web, p. ej. modificar contenidos de forma dinámica o validar entradas de usuario.

Para sincronizar el programa de usuario y el servidor web, así como para la inicialización, es necesario llamar la instrucción WWW (SFC 99) del programa de usuario.

- Si no hace falta que la aplicación web y el programa de usuario interactúen, p. ej. si una página web sólo ofrece información estática, sólo es necesaria una inicialización en el programa de usuario.
- Si se requiere un intercambio de datos sencillo entre las variables PLC y las variables de la aplicación web, p. ej. para mostrar el contenido de las variables PLC o para escribir un valor en una variable PLC, deberá tener en cuenta la sintaxis para la lectura y escritura de variables. Asimismo, en el programa de usuario se requiere en este caso tan solo una inicialización, p. ej. en el OB de arranque.
- Si se requiere una mayor interacción entre la aplicación web y el programa de usuario, deberá manipular, además de la sincronización entre el servidor web y el programa de usuario, la información de estado y de control del DB Web Control. Esto será necesario, p. ej., en el caso de entradas de usuario que se transfieran al servidor web a través del navegador web para que la CPU las evalúe. A diferencia del intercambio de datos sencillo, el programa de usuario influye directamente en el momento en el que se devuelve al navegador web la página web solicitada. En este caso, es necesario que el usuario se familiarice con el concepto de los fragmentos manuales y las estructuras del DB Web Control.

Inicialización

Las páginas web definidas por el usuario se "comprimen" en bloques de datos para que la CPU pueda procesarlas. Durante la configuración, deben generarse los correspondientes bloques de datos a partir de los datos de origen (archivos HTML, imágenes, archivos Javascript, etc.) para poder cargar la aplicación web en la CPU. El DB Web Control (predeterminado: DB 333), que contiene información de estado y de control, así como referencias a otros bloques de datos con páginas web encriptadas, desempeña una función muy importante. Los bloques de datos que contienen las páginas web encriptadas reciben el nombre de "DBs de fragmento".

Si los bloques de datos se han cargado en la CPU, esta no "sabe" que las páginas web definidas por el usuario que contiene están cifradas. Con la instrucción "WWW" (SFC 99), p. ej. en un OB de arranque, se le informa de qué DB es el DB Web Control. Tras esta inicialización, las páginas web definidas por el usuario son accesibles desde un navegador web.

Sincronización

Si el programa de usuario debe intercambiar datos o interactuar con las páginas web definidas por el usuario, deberá utilizarse la instrucción WWW (SFC 99) en la parte cíclica del programa.

Ejemplos de interacción entre programa de usuario y página web:

- Comprobar los datos recibidos
- Recopilar y reenviar los datos para el navegador web solicitante

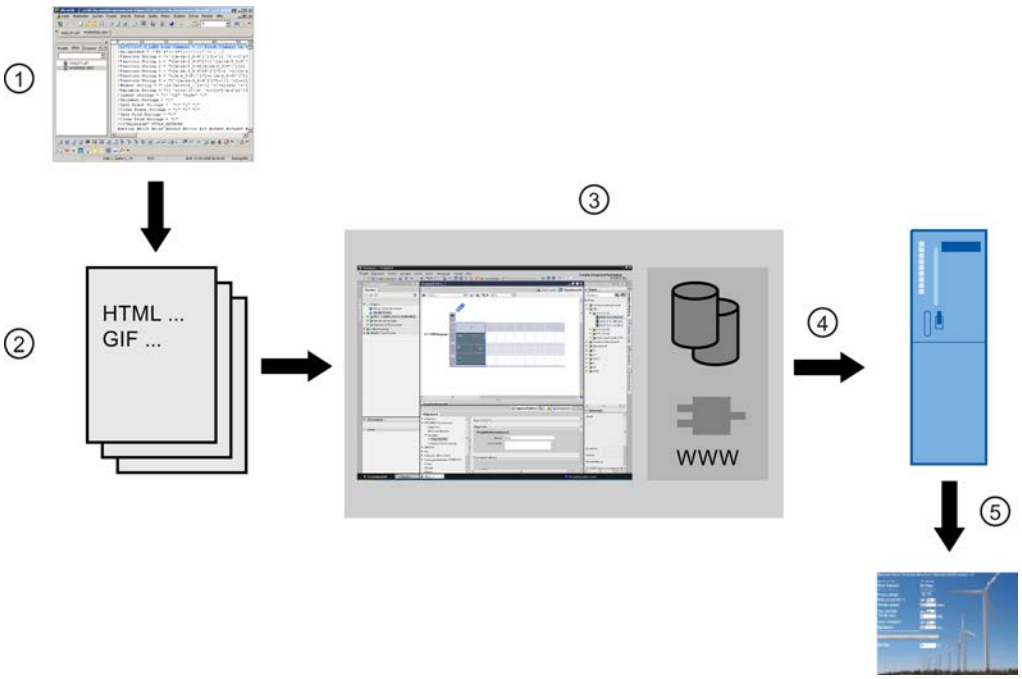
En este caso, la información de estado debe ser evaluable en el momento y la información de control debe comunicarse al servidor web, p. ej., para habilitar una página web solicitada.

Sinopsis del procedimiento

Básico

En este capítulo se describe paso a paso el procedimiento básico para crear, cargar y utilizar páginas web definidas para el usuario en la fase operativa.

El siguiente gráfico presenta el desarrollo para la creación y visualización de páginas web definidas para el usuario de forma simplificada:



① Programar una aplicación web (eventualmente con ayuda de las herramientas adecuadas, de ser necesario con los comandos AWP contenidos para las páginas dinámicas).

② Una aplicación web está formada por archivos de origen, p. ej., *.html, *.gif, *.js, ...

③ Con STEP 7:

- Generar los bloques de datos (DB de control web y DB de fragmentos) a partir de archivos de origen. Los DB contienen información meta y la aplicación web completa incluyendo las imágenes y las partes dinámicas y estáticas de dicha aplicación. Los DB se guardan dentro de la navegación del proyecto, en "Bloque de sistema".
- Acceder a la instrucción "WWW" (SFC 99) en el programa de usuario. La instrucción inicia el servidor web de la CPU para una aplicación web.
- De ser necesario, desprogramar la interacción ente el servidor web y el programa de usuario.

④ Cargar el bloque en la CPU.

⑤ Acceder a la página web desde el navegador. El acceso a las páginas web de la CPU se realiza introduciendo la dirección IP de la CPU.

Crear páginas web

Para crear páginas web definidas por el usuario pueden utilizarse herramientas de diseño web de distintos proveedores. Básicamente, las páginas web deben programarse y diseñarse según las convenciones del W3C (World Wide Web Consortium). En el servidor web de la CPU no se comprueba si se cumplen los criterios del W3C.

Reglas

- La herramienta debe ofrecer la posibilidad de editar el código HTML directamente para que los comandos AWP puedan insertarse en la página HTML.
  En la CPU sólo se analizan sintácticamente los comandos AWP y, por ejemplo, se sustituyen por valores del programa de usuario / memoria imagen de proceso de la CPU.
- Los archivos que contienen comandos AWP deben estar codificados con UTF-8. Por esta razón, ajuste el atributo charset a UTF-8 en los metadatos de la página HTML y guarde el archivo codificado en UTF-8.
- Los archivos que contienen comandos AWP no pueden contener la siguiente secuencia:]]
- Los archivos que contienen comandos AWP no pueden contener fuera de "Rangos de lectura variables" (:=<Nombre de la variable>:) la siguiente secuencia: :=
  Sugerencia: Sustituya el primer carácter de una secuencia prohibida por su código de carácter; para el carácter de dos puntos p. ej. :.

Un pequeño ejemplo de página web definida por el usuario pretende ilustrar la estructura básica.

Requisitos

- La CPU debe disponer de un servidor web y éste debe estar activado.
- El usuario debe haber iniciado sesión como "admin" para acceder en modo de escritura a las variables PLC como usuario de páginas web definidas por el usuario.
- Para el ejemplo inferior, deben haberse definido variables PLC para las variables PLC que deban mostrarse en la página web. Aquí se muestra el ejemplo de la primera variable PLC utilizada "Tank_below_max".

		Nombre	Tipo de datos	Dirección
1		Tank_below_max	Bool	%I0.0

Crear páginas web definidas por el usuario

El siguiente código para un ejemplo de página web lee valores de la memoria imagen de proceso y los representa en una tabla.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/
TR/html4/strict.dtd">
<html>
  <head>
   <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
   <title>Mix</title>
  </head>
   <body>
```

```
    <h1>Mix</h1>
    <h2> Actual State </h2>
    <table border="1">
     <tr>
      <th>Variable</th>
      <th>State</th>
     </tr>
     <tr>
      <td>Tank below max</td>
      <td>:="Tank_below_max":</td>
     </tr>
     <tr>
      <td>Tank above min</td>
      <td>:="Tank_above_min":</td>
     </tr>
    </table>
   </body>
</html>
```

Comandos AWP

Con los comandos AWP (Automation Web Programming) se declara la interfaz entre una aplicación web libremente programable para una CPU que cuenta con un servidor web y los datos de la CPU.

Para el desarrollo de aplicaciones web sólo existen las limitaciones que especifican los navegadores web. Con el programa de usuario se controla en uno de los lenguajes de programación de STEP 7 qué datos de la CPU deben mostrarse en el navegador web del visitante y en qué momento. Utilizando los comandos AWP, que se anotan en los archivos HTML, se acuerdan datos para la interacción pretendida entre la aplicación web y el programa de usuario.

Los comandos AWP se insertan en los archivos HTML como comentarios HTML con una sintaxis especial y declaran las siguientes características:

- leer variables PLC
- escribir variables PLC
- leer variables especiales
- escribir variables especiales
- definir los tipos Enum
- asignar variables a los tipos Enum
- definir fragmentos
- importar fragmentos

Sintaxis de los comandos AWP

Un comando AWP empieza por "`<!--AWP_`" y acaba por "`-->`". En los archivos de Javascript, los comandos deben estar incluidos en comentarios de Javascript ("`/*...*/`").

Reglas de notación para los nombres de variables PLC dentro de un comando AWP

Los comandos AWP "AWP_In_Variable" y "AWP_Out_Variable" contienen un atributo de nombre y, de forma opcional, un atributo de uso. Estos atributos tienen asignado un nombre de variable PLC a través del cual se escriben o leen variables PLC en el navegador. Para el uso de nombres de variables PLC en código HTML existen las siguientes reglas:

- Las variables PLC deben indicarse entre comillas (" ... ").
- En los comandos AWP, las variables PLC utilizadas deben indicarse, además de entre comillas simples ('" ... "') o con una barra invertida (Backslash), con comillas enmascaradas ("\" ... \"").
- Si el nombre de la variable PLC contiene el carácter \ (barra diagonal inversa), este carácter debe identificarse con la secuencia de escape \\ como carácter normal del nombre de la variable PLC.
- Si el nombre de la variable PLC en el comando AWP se encuentra adicionalmente entre comillas simples y dentro del nombre aparece la comilla simple ('), debe identificarse asimismo con la secuencia de escape \' como carácter normal.
- Si en el comando AWP se utiliza una dirección absoluta (entrada, salida, marca), deberá ir entre comillas simples.

Variable PLC	Variable PLC en código HTML
"Velocity"	<!-- AWP_In_Variable Name='"Velocity"' -->
	<!-- AWP_In_Variable Name="\"Velocity\"" -->
"abc\de"	<!-- AWP_In_Variable Name='"abc\\de"' -->
"abc'de"	<!-- AWP_In_Variable Name='"abc\'de"' -->
"abc'de"	<!-- AWP_In_Variable Name="abcde" Use'"abc\'de"' -->
"DB name".tag	<!-- AWP_In_Variable Name='"DB name".tag' -->
"DB name"."ta.g"	<!-- AWP_In_Variable Name='"DB name"."ta.g"' -->
-	<!-- AWP_Out_Variable Name=’flag1’ Use='M0.0' -->

Leer variables

Las páginas web definidas por el usuario pueden leer variables PLC.

Cada variable PLC debe estar especificada con un nombre de variable PLC.

Estas variables Out (dirección de salida vista desde el controlador) se introducen en un lugar cualquiera del texto HTML con la sintaxis que se describe a continuación.

Sintaxis

```
:=<varname>:
```

Durante el funcionamiento, el servidor web sustituye estas referencias por los correspondientes valores actuales de las variables PLC.

`<varname>` puede ser una variable PLC general sencilla, pero también una ruta completa de variables hacia un elemento de una estructura.

Reglas de notación para los nombres de variables PLC

- Las variables PLC en código HTML deben ir entre comillas (") si han sido definidas en la tabla de variables. En el caso de las variables de bloques de datos, el nombre del bloque de datos debe ir entre comillas. Si se utilizan caracteres especiales en los elementos de estructura del bloque de datos tales como, p. ej., el punto (.) o el espacio en blanco, esta parte debe ir asimismo entre comillas.
- No se utilizan las comillas para las direcciones absolutas de entradas, salidas o marcas.

Variable PLC	Variable PLC en código HTML
"DB_name".var_name	:="DB_name".var_name:
"DB_name"."var.name"	:="DB_name"."var.name":
"memory"	:="memory":
-	:=I0.0:
	:=Q0.0:
	:=MW100:
	:=%MW100:
"My_Data_Block".flag1	<!-- AWP_Out_Variable Name='flag1' Use='"My_Data_Block".flag1' -->
	...
	:=flag1:

- Si el nombre de variable PLC contiene los caracteres: (dos puntos) o \ (barra diagonal inversa), estos caracteres deben identificarse con la secuencia de escape \: o \\ como carácter normal del nombre de la variable PLC.

Variable PLC	Variable PLC en código HTML
"abc:de"	:="abc\:de":
"abc\de"	:="abc\\de":

- Caracteres especiales "<, &, >"
  Si aparecen estos caracteres en el nombre de la variable (p. ej. "a<b"), pueden darse problemas de representación.
  Evite expresiones tales como, p. ej. :="a<b": en la página HTML.
  A fin de evitar problemas de representación, utilice p. ej. un comando AWP con una expresión Use siguiendo el patrón que se muestra a continuación. El atributo Use define la variable PLC con los caracteres problemáticos, el atributo Name define el nombre sin caracteres problemáticos, tal y como se utiliza en la página HTML.

Variable PLC	Variable PLC en código HTML
"a<b"	<!—AWP_Out_Variable Name='simplename' Use='"a<b"' -->
	...
	:=simplename:

Escribir variables

Las páginas web definidas por el usuario pueden escribir datos en la CPU.

Para ello, se requiere un comando AWP que identifique la variable PLC que hay que escribir.

Además, cada variable PLC debe estar especificada con un nombre de variable PLC.

Las variables In (dirección de entrada vista desde el controlador) se activan en la página del navegador. Esto puede efectuarse, por ejemplo, en los formularios.

El navegador activa las variables en el HTTP-Request ya sea en el encabezado de HTTP (por cookie o método POST) o en la URL (método GET) y, a continuación, el servidor web las escribe en la correspondiente variable PLC.

Sintaxis

Para que las variables In puedan escribirse en la CPU, es necesario definirlas con una instrucción AWP explícita:

```
<!-- AWP_In_Variable Name='"<Varname1>"' Name='"<Varname2>"'
Name='"<Varname3>"' -->
```

En una instrucción puede haber varias variables definidas (como puede verse arriba).

En el caso en que el nombre de la variable que utilice para la aplicación web no sea idéntico al nombre de la variable PLC, puede realizarse la asignación a una variable PLC con el parámetro "Use".

```
<!-- AWP_In_Variable Name='<Varname_Webapp>' Use='<PLC_Varname>'
```

Ejemplo

El comando AWP "AWP_In_Variable" es imprescindible, especialmente para el tratamiento de formularios.

```
<form method="post" action="/awp/appl/x.html">
 <p>
  <input name='"var1"' type="text">
  <input value="set" name='"Button1"' type="submit">
 </p>
</form>
```

En el formulario antes definido, se transmiten las variables "var1" y "Button1" por "post" al servidor web. El usuario activa la variable "var1" en el campo del formulario. La variable "Button1" tiene el valor fijo "set". Para que las variables "var1" y "Button1" puedan escribirse en la CPU, debe incluirse la siguiente instrucción en el mismo fragmento:

```
<!-- AWP_In_Variable Name='"var1"' Name='"Button1"' -->
```

Puesto que las variables generales deben ir entre comillas, el nombre en el comando AWP debe ir entre apóstrofes o entre comillas con una barra inversa (\").

```
<!-- AWP_In_Variable Name='"Info".par1' -->
```

Condiciones para el acceso en escritura durante el funcionamiento

Para que un usuario pueda escribir en variables PLC desde una página web definida por el usuario, debe cumplirse el siguiente requisito:

- Se ha configurado un usuario y éste ha iniciado una sesión.

Esta regla es válida en principio para todos los accesos en escritura de páginas web a una CPU.

Consulte también

Lista de usuarios (Página 944)

Variables especiales

Las variables especiales son principalmente las variables HTTP que aparecen en las definiciones del World Wide Web Consortium (W3C) . También se utilizan variables especiales para las cookies y las variables de servidor.

Los comandos AWP para leer y escribir variables especiales sólo se diferencian por los parámetros adicionales de los comandos AWP para leer y escribir variables normales.

Leer variable especial

El servidor web puede leer variables PLC y entregarlas a variables especiales en el encabezado de respuesta HTTP. Puede p. ej. leer una URL para un desvío a otra página web y entregarla a la variable especial HEADER:Location sirviéndose de la variable especial HEADER:Location.

Pueden leerse las siguientes variables especiales:

Nombre	Descripción
COOKIE_VALUE:name	Valor de la cookie con el nombre: "name"
COOKIE_EXPIRES:name	Tiempo de ejecución de la cookie con el nombre: "name" en segundos (debe haberse activado con antelación).
HEADER:Status	Código de estado HTTP (si no se ha introducido ningún otro valor, se devolverá el código de estado 302).
HEADER:Location	Ruta para la derivación a otra página. El código de estado 302 debe estar activado.
HEADER:Retry-After	Tiempo durante el que previsiblemente el servicio no estará disponible. El código de estado 503 debe estar activado.
HEADER: …	El resto de las variables de encabezado también puede transmitirse de este modo.

Con el comando AWP "AWP_Out_Variable" se determinan las variables PLC que deben transferirse al navegador web en el encabezado de HTTP.

Estructura general:

```
<!-- AWP_Out_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Descripción de parámetros

- Name: tipo y nombre de la variable especial
- Use (parámetro opcional): en caso de que el nombre de la variable especial no sea idéntico al nombre de la variable PLC, puede realizarse la asignación a una variable PLC con el parámetro "Use".

Ejemplo:

```
<!-- AWP_Out_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Escribir una variable especial

En principio, el programa de usuario de la CPU puede evaluar todas las variables HTTP escritas por el navegador web en el encabezado de HTTP. Ejemplos de tipos de variables:

Nombre	Descripción
HEADER:Accept-Language	Idioma aceptado o preferente
HEADER:Authorization	Prueba de autorización para un recurso solicitado
HEADER:Host	Host y puerto del recurso solicitado
HEADER:User-Agent	Información sobre el navegador
HEADER: …	El resto de las variables de encabezado también puede transmitirse de este modo
SERVER:current_user_id	Indica si un usuario ha iniciado una sesión (current_user_id=0: ningún usuario ha iniciado una sesión)
SERVER:current_user_name	Nombre del usuario que ha iniciado la sesión
SERVER:GET	El método Request es GET
SERVER:POST	El método Request es POST
COOKIE_VALUE:name	Valor de la cookie con el nombre: "name"

Con el comando AWP "AWP_In_Variable" se determinan las variables especiales que deben evaluarse en el programa de usuario de la CPU.

Estructura general:

```
<!-- AWP_In_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Descripción de parámetros:

Name: tipo y nombre de la variable especial

Use (parámetro opcional): en caso de que el nombre de la variable especial no sea idéntico al nombre de la variable PLC, puede realizarse la asignación a una variable PLC con el parámetro Use .

Ejemplos:

```
<!-- AWP_In_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

El nombre de la variable en el encabezado de HTTP se sustituye por el nombre de la variable PLC indicado en Use . La cookie se escribe en la variable PLC "info".language .

```
<!-- AWP_In_Variable Name='COOKIE_VALUE:siemens' Use='"info".language' -->
```

El nombre de la variable en el encabezado de HTTP se sustituye por el nombre de la variable PLC indicado en Use. La cookie se escribe en la variable PLC "info".language .

```
<!-- AWP_In_Variable Name='"COOKIE_VALUE:siemens"' -->
```

La variable del encabezado HTTP se escribe en la variable PLC del mismo nombre.

Tipos de enumeraciones

Tipos de "enumerations" (Enums)

Con la ayuda de los Enums, los valores numéricos del programa del PLC pueden convertirse en texto y viceversa. La asignación de los valores numéricos también puede realizarse para más de un idioma.

Crear Enums

Introduzca un comando AWP utilizando la siguiente sintaxis al principio del archivo HTML:

```
<!-- AWP_Enum_Def Name="<Nombre del tipo de Enum>"
Values='0:"<Text_1>", 1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Para guardar, por ejemplo, valores alemanes como archivo HTML en la carpeta "de" del directorio HTML:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"an", 1:"aus", 2:"Störung"' -->
```

Para guardar, por ejemplo, valores ingleses como archivo HTML en la carpeta "en" del directorio HTML:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"error"' -->
```

Asignar Enums

La asignación de las variables del programa de usuario a cada uno de los textos Enum se realiza con un comando AWP propio:

```
<!-- AWP_Enum_Ref Name="<VarName>" Enum="<EnumTypeName>" -->
```

<VarName> representa el nombre simbólico del programa de usuario y <EnumTypeName>, el nombre definido anteriormente del tipo Enum.

Nota

En todos los fragmentos en los que una variable PLC referencie textos Enum, esta variable PLC debe asignarse junto con el correspondiente comando AWP al nombre del tipo de Enum.

Se debe tener en cuenta que entre una asignación Enum y el uso de Enum no existe ningún comando AWP para importar fragmentos debido a que esta importación hace que la asignación Enum se encuentre en otro fragmento que el uso de Enum.

Ejemplo

Se ha definido un tipo de Enum "estado" con los valores "0" y "1". "0" equivale a "off", mientras que "1" equivale a "on":

```
<!-- AWP_Enum_Def Name="estado" Values='0:"off", 1:"on"' -->
```

El código HTML de la página web que debe visualizarse contiene el siguiente código:

```
<!-- AWP_Enum_Ref Name="estado operativo" Enum="estado" -->
:=estado operativo:
```

En función del valor de la variable "estado operativo", ya no se muestra "0" ó "1" como resultado, sino "off" u "on".

Uso simplificado de tipos de enumeración

En las CPU S7-1200 a partir de la versión 4 del firmware es posible emplear enumeraciones directamente en comandos AWP para leer y escribir variables PLC.

Cree enumeraciones como se ha descrito en el párrafo anterior para poder utilizar los valores con comandos AWP de lectura y escritura.

Crear Enums

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Uso de Enums en comandos AWP de escritura y lectura

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<EnumType>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<EnumType>" -->
```

Ejemplo de lectura de variables PLC

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_Out_Variable Name='"Alarm"'
Enum="AlarmEnum" -->...<p>The current value of "Alarm"
is :="Alarm":</p>
```

Si el valor de "Alarma" está en la CPU "2", en la página HTML se muestra el siguiente texto:

'The current value of "Alarm" is Tank is empty' porque la definición de Enum asigna al string "Tank is empty" el valor numérico 2.

Ejemplo de escritura de variables PLC

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_In_Variable Name='"Alarm"'
Enum='AlarmEnum' -->...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /><p>
</form>
```

Como la definición de Enum asigna al string "Tank is full" el valor numérico "1", el valor "1" se escribe en la variable PLC"Alarm".

Definir fragmentos

Fragmentos

Se entiende por fragmentos cada una de las "unidades de significado" de una página web que debe procesar la CPU.

Los fragmentos suelen ser páginas enteras, aunque también pueden ser elementos individuales como archivos (p. ej. imágenes) o documentos enteros.

Definir fragmentos

```
<!-- AWP_Start_Fragment Name="<Name>" [Type="<Type>"] [ID="<Id>"]
[Mode=<Mode>]-->
```

Con este comando se define el inicio de un fragmento. Un fragmento se extiende hasta el principio del siguiente fragmento o hasta el final del archivo.

- <Name> indica el nombre del fragmento.
  El nombre debe comenzar con una letra [a-zA-Z] o un carácter de subrayado (_). Tras este primer carácter pueden aparecer letras, caracteres de subrayado o números [0-9].
- <Type> indica el tipo de fragmento.
  - "manual" El programa de usuario se informa mediante el Request de un fragmento de que es posible realizar cambios en la página web que debe devolverse con el programa de usuario.
  - "automatic" La página se edita automáticamente (por defecto).
- <id> para el fragmento puede predefinirse una ID numérica. Si no se especifica ninguna ID, se asignará automáticamente una ID al fragmento. En el caso de páginas manuales (<Type>=manual) puede accederse al fragmento en el programa de usuario de la CPU con esta ID.

  Nota

  Es necesario establecer una ID baja, ya que la ID más alta condiciona el tamaño del DB Web Control.

- <Mode> Los fragmentos son compatibles con ambos modos: visible y hidden.
  - "visible": el fragmento forma parte de la página web. Este modo está predeterminado y puede omitirse.
  - "hidden": el fragmento no forma parte de la página web, pero se guarda en el DB web y está disponible para que el programa de usuario lo agregue a una página web solicitada. Intercambiando los ID de fragmento (Variable del DB Web Control fragment_index) se agrega un fragmento "oculto" a la página web solicitada.

Con el comando "AWP_Start_Fragment", el documento Input se divide completamente en fragmentos. Por este motivo, no es necesario un "AWP_End_Fragment".

Sin un comando de inicio de fragmento, un archivo se reproduce como un fragmento cuyo nombre se deriva del nombre del archivo. Si un archivo se descompone en varios fragmentos (mediante "AWP_Start_Fragment"), el archivo debe empezar por el comando "AWP_Start_Fragment".

Importar fragmentos

Existe la posibilidad de declarar un fragmento en una página HTML e importar dicho fragmento a otras páginas web.

Ejemplo

El logotipo de un empresa debe mostrarse en todas las páginas web de una aplicación web.

El código HTML para el fragmento que muestra el logotipo de la empresa sólo existe una vez. El fragmento puede importarse tantas veces y a tantos archivos HTML como sea necesario.

Sintaxis

```
<!-- AWP_Import_Fragment Name = "<name>"-->
```

- <name> es el nombre del fragmento que se desea importar.

Ejemplo

Código HTML dentro de una página web que declara un fragmento:

```
<!-- AWP_Start_Fragment Name = "My_Company_Logo"-->
<p><img src = "compay_logo.jpg"></p>
```

Ejemplo

Código HTML dentro de otra página web que importa el fragmento declarado:

```
<!-- AWP_Import_Fragment Name = "My_Company_Logo"-->
```

Generar y cargar bloques de datos

Requisitos

- Se han creado todos los archivos de origen necesarios (*.html, *.js, *.png...) para la aplicación web.
- Los archivos de origen están en una carpeta, pero sólo los archivos de origen que son necesarios para la aplicación web. En esta carpeta no debe haber otros archivos.

Nota

Longitud de los nombres de archivos y variables

Si tiene una aplicación web amplia, con muchos archivos y directorios, es posible que fracase la generación de los bloques de datos web. En este caso, la generación se cancela con el mensaje "Lista de textos desbordada...". La causa son limitaciones internas del tamaño de la información administrativa que se guardará en el bloque de datos web.

Solución: utilice nombres de archivo y nombres de variable cortos.

Procedimiento

Para generar bloques de datos a partir de los archivos de origen para las páginas web definidas por el usuario en STEP 7, proceda del siguiente modo:

1. Seleccione la CPU, p. ej. en la configuración de dispositivos.
2. Seleccione las propiedades para las páginas web definidas por el usuario "Propiedades > General > Servidor web" de la ventana de inspección.
3. Como "fuente HTML" debe seleccionarse la carpeta en la que se encuentran los archivos de origen para la aplicación web.
4. Como página de inicio HTML debe introducirse la página HTML que deba abrirse al iniciar la aplicación web.
5. Si es necesario, introduzca el nombre de una aplicación.
6. Si es necesario, complemente la serie de extensiones de nombres de archivos como "archivos con contenido dinámico". Sólo deben introducirse las extensiones de nombres de archivos que contengan comandos AWP.
7. El número del DB Web Control y del inicio del DB de fragmento pueden conservarse si no están ocupados por el programa de usuario.
8. Haga clic en el botón "Generar" para generar DBs a partir de los archivos de origen.
   Los bloques de datos generados se depositan en la carpeta "Bloques de sistema" (subcarpeta "Servidor web") del árbol del proyecto.
9. En la vista de redes, seleccione la CPU que debe cargarse y elija el comando "Cargar en dispositivo" en el menú "Online" para cargar los bloques. Antes de la carga se iniciará implícitamente la compilación de los bloques.
   En caso de que se notifiquen fallos durante este proceso, deben solucionarse antes de cargar la configuración.

Estructura del programa del PLC

El programa de usuario debe llamar la instrucción " WWW" para que la aplicación web, es decir, las páginas web definidas por el usuario, estén disponibles en las páginas web estándar de la CPU y puedan abrirse.

El DB de Web Control generado a partir de los archivos de origen es el parámetro de entrada (CTRL_DB) para la instrucción "WWW". El DB de Web Control referencia el contenido de las

páginas web definidas por el usuario que está codificado en los DBs de fragmento y contiene información de estado y control.

Llamada de la instrucción "WWW" en el programa de arranque

Para evitar que el programa de usuario no influya en páginas web solicitadas, basta con llamar una sola vez la instrucción "WWW" en un OB de arranque. La instrucción inicializa la comunicación entre el servidor web y la CPU.

Llamada de la instrucción "WWW" en el programa cíclico

La llamada de la instrucción "WWW" también puede producirse en un OB procesado cíclicamente (p. ej. el OB 1). Esto ofrece la ventaja de que pueden aceptarse solicitudes del servidor web en el programa de usuario. Para ello, deben utilizarse fragmentos manuales.

En este caso, hay que evaluar información del DB de Web Control para identificar la página web solicitada o el fragmento solicitado. Por otra parte, es necesario activar un bit en el programa de usuario para habilitar explícitamente la página web que debe proporcionar el servidor web después de procesar la solicitud de página web.

La estructura del DB de Web Control se describe en el siguiente apartado.

DB de Web Control

El DB de Web Control (por defecto, DB 333) es generado por STEP 7 y contiene información sobre la estructura de las páginas de usuario, el estado de la comunicación y posibles errores producidos.

Además del DB de Web Control, también se crean los DBs de fragmento. Estos DBs de fragmento (también puede ser un solo DB de fragmento) se referencian en el DB de Web Control. Los DBs de fragmento contienen las páginas web y los datos de medios (p. ej. imágenes) codificados en fragmentos. El contenido de los DBs de fragmento no puede modificarse con el programa de usuario. Se crean automáticamente y sólo sirven para la gestión de datos.

A las variables de estado y control del DB de Web Control se accede de forma simbólica.

A continuación, se indican las variables del DB de Web Control necesarias para la evaluación de estado y el control de la interacción.

Pueden obtenerse dos tipos de información del DB de Web Control:

- Información de estado general: información no vinculada a ninguna solicitud de página web concreta (Request).
- Información de estado y control de Request: información sobre Requests pendientes.

Información de estado general

"WEB-Control_DB".commandstate.init	Activa e inicializa la aplicación web.
"WEB-Control_DB".commandstate.deactivate	Desactiva la aplicación web.
"WEB-Control_DB".commandstate.inititializing	Se inicializa la aplicación web (leer DB de Web Control, etc.).

"WEB-Control_DB".commandstate.error	No se ha podido inicializar la aplicación web. El motivo está codificado in "WEB-Control_DB".commandstate.last_error .
"WEB-Control_DB".commandstate.deactivating	Se finaliza la aplicación web.
"WEB-Control_DB".commandstate.initialized	Se ha inicializado la aplicación web y ésta ya está lista.
"WEB-Control_DB".commandstate.last_error	Tabla de valores de posibles errores (véase tabla siguiente).

Last_error	Descripción
1	El DB de fragmento es incoherente (no es compatible con el DB de Web Control).
2	Ya existe una aplicación web con este nombre.
3	Problema de memoria al inicializar en el servidor web.
4	Los datos del DB de Web Control son incoherentes.
5	Un DB de fragmento no está disponible (no está cargado).
6	Un DB de fragmento no tiene identificación AWP.
7	El fragmento Enum (contiene los textos e información para los tipos Enum) no está disponible.
8	Una acción solicitada a través de las marcas de comando en el DB de Web Control no está permitida en el estado actual.
9	La aplicación web no está inicializada (si no se produce una nueva inicialización tras la desactivación).
10	El servidor web está desactivado.
...	Se desactiva Last_error si la aplicación web se ha inicializado correctamente.

Información de estado de Request

La información de estado de Request está vinculada a uno de los cuatro Requests posibles, x = [1 … 4].

"WEB-Control_DB".requesttab[x].idle	No hay que hacer nada.
"WEB-Control_DB".requesttab[x].waiting	El programa de usuario debe responder a un Request de un fragmento manual e iniciar explícitamente el procesamiento posterior en el servidor web.
"WEB-Control_DB".requesttab[x].sending	El servidor web está ocupado con el procesamiento de Requests/fragmentos.
"WEB-Control_DB".requesttab[x].aborting	El servidor web cierra la conexión TCP.

Información de control de Request

La información de control de Request está vinculada a uno de los cuatro Requests posibles, x = [1 … 4].

"WEB-Control_DB".requesttab[x].continue	Habilita el fragmento que se acaba de procesar para su transmisión. Se inicia el procesamiento del siguiente fragmento.
"WEB-Control_DB".requesttab[x].repeat	Habilita el fragmento que se acaba de procesar para su transmisión. A continuación, el fragmento se procesa de nuevo.
"WEB-Control_DB".requesttab[x].abort	Cierra la conexión TCP.
"WEB-Control_DB".requesttab[x].finish	Habilita el fragmento que se acaba de procesar para su transmisión. Detiene el procesamiento posterior de fragmentos (finaliza el Request).

Ejemplo:

La variable para el DB es la siguiente: "WEB-Control_DB". Con el bit "WEB-Control_DB".commandstate.error del programa de usuario puede consultarse si se han producido errores durante la inicialización de la aplicación web.

En caso de que se haya producido un error, éste puede analizarse con el valor de "WEB-Control_DB".commandstate.last_error.

Interacción con el programa de usuario

Con ayuda de los fragmentos manuales conseguirá que el programa de usuario reaccione de forma sincrónica a los datos introducidos en el navegador, de manera que pueda prepararse la página devuelta por el programa de usuario.

Tipo de fragmento

Para poder reaccionar directamente a los datos recibidos en el programa de usuario deberá utilizarse el tipo de fragmento "manual" para el fragmento que escribe los datos (para "Páginas manuales"):

```
<!-- AWP_Start_Fragment Name="testfrag" ID="1" Type="manual" -->
```

La transferencia de los valores al servidor web de la CPU se realiza de la misma forma tanto en páginas manuales como automáticas:

Ejemplo:

```
<form method="POST" action="">
<p>
<input type="submit" value="Set new value">
<input type="text" name='"Velocity"' size="20">
</p>
</form>
```

Programa de usuario para fragmentos manuales

En el programa de usuario de la CPU debe accederse a la instrucción "WWW" (SFC 99) de forma cíclica durante el uso de páginas manuales.

Para poder reaccionar a los valores introducidos en el navegador será necesario evaluar, en el programa de usuario, el request que ajusta la página manual al servidor web. Para ello deberá examinar cíclicamente el DB de control web (p. ej., DB 333) en busca de requests pendientes. En el apartado "requesttab" del DB de control web se encuentra el array que administra cuatro requests. Cada elemento del array contiene, en una estructura, información sobre el request en cuestión.

Un ejemplo simple de programa AWL muestra cómo se comprobará mediante las variables del DB de control web la existencia de requests pendientes.

```
      U     "WEB-Control_DB".requesttab[1].waiting
      SPBN  R2
      L     "WEB-Control_DB".requesttab[1].fragment_index
      L     1
      SPA   PdRq                        // request pending

R2:   U     "WEB-Control_DB".requesttab[2].waiting
      SPBN  R3
      L     "WEB-Control_DB".requesttab[2].fragment_index
      L     2
      SPA   PdRq                        // request pending

R3:   U     "WEB-Control_DB".requesttab[3].waiting
      SPBN  R4
      L     "WEB-Control_DB".requesttab[3].fragment_index
      L     3
      SPA   PdRq                        // request pending

R4:   U     "WEB-Control_DB".requesttab[4].waiting
      SPBN  NoRq
      L     "WEB-Control_DB".requesttab[4].fragment_index
      L     4
      SPA   PdRq                        // request pending

NoRq: L     0
      T     #frag_index
      T     #req_index
      SPA   Exit

PdRq: T     #req_index
      POP
      T     #frag_index

Exit: BE
```

En el caso de que esté activado un request, escribirá el apartado de programa AWL la ID de fragmento en la variable #frag_index , y el número de request (área de valor 1-4) en la variable#req_index.

Con la información obtenida puede procesarse expresamente la información transferida en el request en el programa AWL para cada ID de fragmento (p. ej., prueba de plausabilidad).

Cuando el procesamiento del request concluye mediante el programa AWL, deberá responderse a dicho request y restablecer la entrada en cuestión desde el "requesttab" del DB de control web (p. ej., DB 333).

Un ejemplo simple de programa AWL para responder los requests:

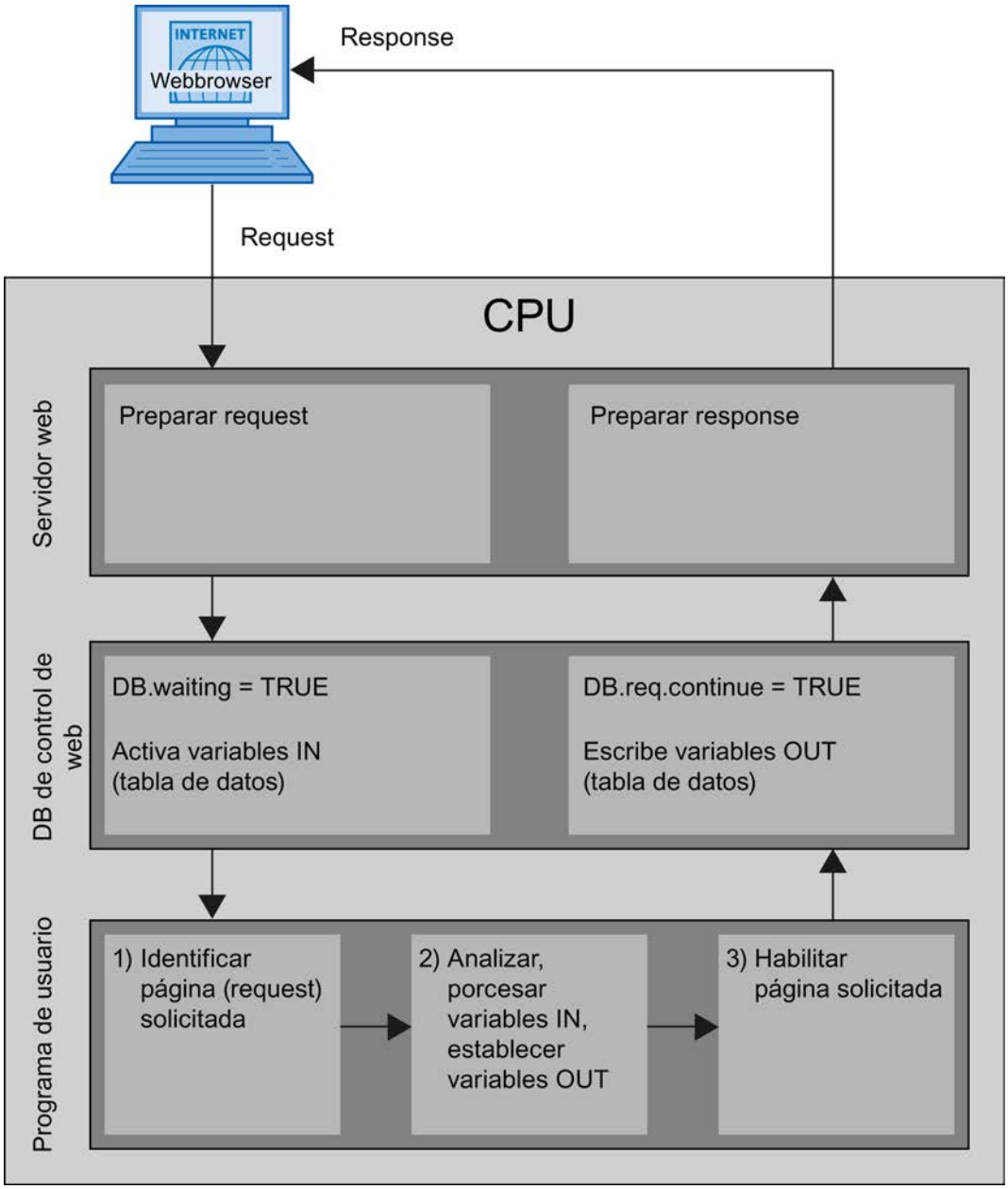
```
SET
      L     #req_index
      SPL   Con0
      SPA   Con0
      SPA   Con1
      SPA   Con2
      SPA   Con3
      SPA   Con4
Con0: SPA   Exit

Con1: =     "WEB-Control_DB".requesttab[1].continue
      SPA   Exit
Con2: =     "WEB-Control_DB".requesttab[2].continue
      SPA   Exit
Con3: =     "WEB-Control_DB".requesttab[3].continue
      SPA   Exit
Con4: =     "WEB-Control_DB".requesttab[4].continue
      SPA   Exit

Exit: BE
```

Desarrollo principal de un request de navegador con interacción del programa de usuario

La siguiente imagen muestra el desarrollo principal y simplificado del request del navegador sobre las repercusiones en los contenidos DB de control web y las acciones necesarias del programa de usuario hasta la devolución de la página web der preparada (response).



Visualización de páginas web definidas por el usuario en el navegador

Visualizar páginas web en el navegador

Las páginas web se abren desde las páginas web estándar del navegador web.

Las páginas web estándar disponen, además de los otros enlaces de la barra de navegación, de un enlace a las "Páginas de usuario".

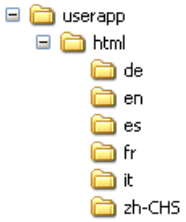
Al hacer clic sobre el enlace "Páginas de usuario", el navegador web abre la página web que se haya configurado como página de inicio HTML.

Crear páginas web en varios idiomas definidas para el usuario

Tiene la posibilidad de hacer que sus páginas web definidas para el usuario estén disponibles en distintos idiomas.

Condiciones

Las páginas HTML dependientes de idioma deberán guardarse en una estructura de carpeta que contenga las abreviaturas idiomáticas correspondientes:



Abreviatura idiomática establecida

Están establecidas las abreviaturas idiomáticas "de", "en", "fr", "es", "it" y "zh-CHS". No se soportarán otras carpetas de idioma o carpetas nombradas de distinta forma.

De ser necesario, podrá crear otras carpetas en la misma jerarquía de carpetas para otros archivos; p. ej., una carpeta "img" para imágenes y otra carpeta "script" para archivos Javascript.

Cambio de idioma para páginas web definidas por el usuario

Requisitos

Las páginas HTML se encuentran en las carpetas de idioma predefinidas, p. ej., páginas HTML con texto en alemán en la carpeta "de", páginas HTML con texto en inglés en la carpeta "en", etc.

Concepto del cambio de idioma

El cambio de idioma se basa en una cookie predefinida denominada "siemens_automation_language". Si se activa el valor "de" en la cookie, el servidor web se conectará a la página web de la carpeta "de" en la próxima solicitud o actualización de página web.

De forma análoga, el servidor web se conecta a la página web de la carpeta "en" si la cookie adopta el valor "en".

Ejemplo de un cambio de idioma

El ejemplo está formado de la siguiente manera:

- En las carpetas de idioma "de" y "en" encontrará los archivos HTML dependientes de idioma de igual nombre, p. ej., "langswitch.html". Los textos que se van a mostrar dentro de los dos archivos estarán, según el nombre de la carpeta, en alemán o en inglés.
- Además existe una carpeta "script" en la estructura de carpetas en la que se encuentra el archivo Javascript "lang.js". En este archivo se permutan funciones necesarias para el cambio de idioma.

Estructura del archivo "langswitch.html" (carpeta "de")

En el encabezado del archivo se hayan establecidos los datos meta "Content-Language", charset y ruta a los archivos Javascript.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Cambio de idioma página alemana</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

La selección de idioma se realiza en el cuerpo (body) del archivo con ayuda del elemento HTML "select". El elemento select dispone una lista de selección y contiene las opciones "de", con el nombre "Alemán", y "en", con el nombre "Inglés"; "de" está preseleccionado.

Con ayuda del gestor de evento (event-handler) "onchange" se accederá a la función "DoLocalLanguageChange(this)". El parámetro "this" transfiere el objeto select con la opción seleccionada a esta función. En cada modificación, "onchange" realiza la opción para acceder a la función.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" selected >Alemán</option>
                <option value="en" >Inglés</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Estructura del archivo "langswitch.html" (carpeta "en")

El encabezado del archivo HTML con texto en inglés es análogo al archivo HTML con texto en alemán.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
<html>
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

La selección de idioma también se realiza en el cuerpo (body) del archivo con ayuda del elemento HTML "select". A diferencia del archivo HTML alemán, la opción de inglés está preseleccionada y el texto y las etiquetas también están en inglés.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" >Alemán</option>
                <option value="en" selected >Inglés</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Estructura del archivo "langs.js" (en la carpeta "script")

En el archivo Javascript se define la función "DoLocalLanguageChange" que por su parte accede a la función "SetLangCookie" con el valor de selección de idioma. SetLangCookie reúne el nombre y el valor de la cookie y la activa a través de la propiedad correspondiente document.cookie. Para que el servidor web reaccione a la activación de las cookies con la visualización del idioma deseado, deberá volver a cargarse la página web (top.window.location.reload).

```
función DoLocalLanguageChange(oSelect) {
        SetLangCookie(oSelect.value);
        top.window.location.reload();
    }

 función SetLangCookie(value) {
        var strval = "siemens_automation_language=";
            // esta es la cookie por la que el servidor web
            // detecta el idioma deseado
            // el servidor web requiere este nombre
        strval = strval + value;
        strval = strval + "; path=/ ;";
            // establece la ruta hacia la aplicación, ya que de otra
forma
            // la ruta se establecería hacia la página solicitante
no obtendría la cookie
            /*
```

```
                usar expiración si esta cookie debe mantenerse más tiempo
                que la actual sesión del navegador
                var now      = new Date();
                var endttime = new Date(now.getTime() + expiration);
                strval = strval + "; expires=" + endttime.toGMTString()
+ ";";
                */
            document.cookie = strval;
        }
```

Información adicional

Información y ejemplos para las páginas definidas por el usuario

Encontrará ejemplos y explicaciones adicionales en el portal Service & Support.

- Encontrará un ejemplo del acceso al servidor web de una S7-1200 con un iPad y para la integración de la librería jQuery aquí (http://support.automation.siemens.com/WW/view/en/62046437?Datakey=47071380).
- Encontrará un ejemplo de aplicación para la creación de una página web definida por el usuario aquí (http://support.automation.siemens.com/WW/view/es/58862931).
- Encontrará un resumen sobre el tema Industrial Security aquí (http://support.automation.siemens.com/WW/view/es/50203404).

8.1.4.4 Configurar sistemas de automatización (S7-1200)

Direccionar módulos

Direccionamiento de módulos

Introducción

En las columnas Dirección I y Dirección Q de la vista general de dispositivos se ven las direcciones o áreas de direcciones de los módulos. Al lado hay otras direcciones que se explican a continuación.

Dirección E/S (dirección de periferia)

Las direcciones E/S (de entrada y salida) se necesitan para leer entradas y activar salidas en el programa de usuario.

Las direcciones de entrada y salida se asignan automáticamente al insertar módulos en el rack. La dirección del primer canal representa la dirección inicial de un módulo. Las direcciones de los demás canales se derivan de dicha dirección inicial. La dirección inicial se deriva de la longitud de dirección específica del módulo.

Dirección de dispositivo (p. ej. dirección Ethernet)

Las direcciones de dispositivos son direcciones de módulos programables (direcciones Industrial Ethernet). Se requieren para direccionar los distintos dispositivos de una subred, p. ej. para cargar un programa de usuario en una CPU.

ID de hardware para la identificación de módulos y submódulos

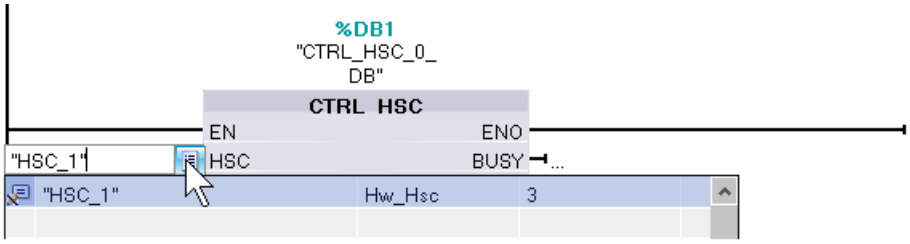
Además de las direcciones E y S, se asigna automáticamente un identificador de hardware (ID HW) que se utiliza para direccionar e identificar el módulo. También los submódulos (unidades funcionales de un módulo), como un contador integrado, obtienen una ID HW.

La ID HW consiste en un número entero y es emitida por el sistema junto con avisos de diagnóstico con el fin de localizar el módulo o submódulo defectuoso.

La ID HW se utiliza también para una serie de instrucciones para direccionar el módulo correspondiente.

No es posible modificar la ID HW.

Ejemplo: Identificación de contador rápido de la CPU S7-1200



La ID HW se asigna automáticamente al insertar componentes en la vista de dispositivos o redes y se introduce junto a las variables PLC (ficha Constantes del sistema). También se asigna automáticamente un nombre para la ID HW. Las constantes del sistema de las variables PLC tampoco se pueden modificar.

Consulte también

Definir las direcciones de entrada y salida (Página 972)

Asignar direcciones a una ubicación del programa (Página 973)

Introducción a la carga de una configuración (Página 1610)

Ventana de inspección (Página 501)

Definir las direcciones de entrada y salida

Las direcciones de entrada y salida se predeterminan automáticamente. De todas formas, es posible modificar posteriormente la asignación de direcciones.

Todas las direcciones de módulos están en el área de la memoria imagen de proceso. Ésta se actualiza cíclica y automáticamente.

Requisitos

Se encuentra en la vista de dispositivos.

Procedimiento

Para modificar el área de direcciones predeterminada, proceda del siguiente modo:

1. En la vista de dispositivos, haga clic en el módulo cuya dirección inicial desea ajustar.
2. En la ventana de inspección, vaya a "Propiedades" y, seguidamente, a "Direcciones de E/S".
3. Introduzca la dirección inicial deseada en "Dirección inicial".
4. Pulse <Intro> o haga clic en cualquier otro objeto para aplicar el valor modificado.

Si se ha introducido una dirección no válida, aparecerá un aviso con la próxima dirección libre.

Nota

También existe la posibilidad de modificar las direcciones directamente en la vista de dispositivos.

Consulte también

Editar las propiedades y los parámetros (Página 529)

Direcciones de entrada y salida en la vista de direcciones (Página 531)

Asignar direcciones a una ubicación del programa

Es posible asignar direcciones desde los canales E/S de los módulos directamente a las ubicaciones del programa o a una tabla de variables.

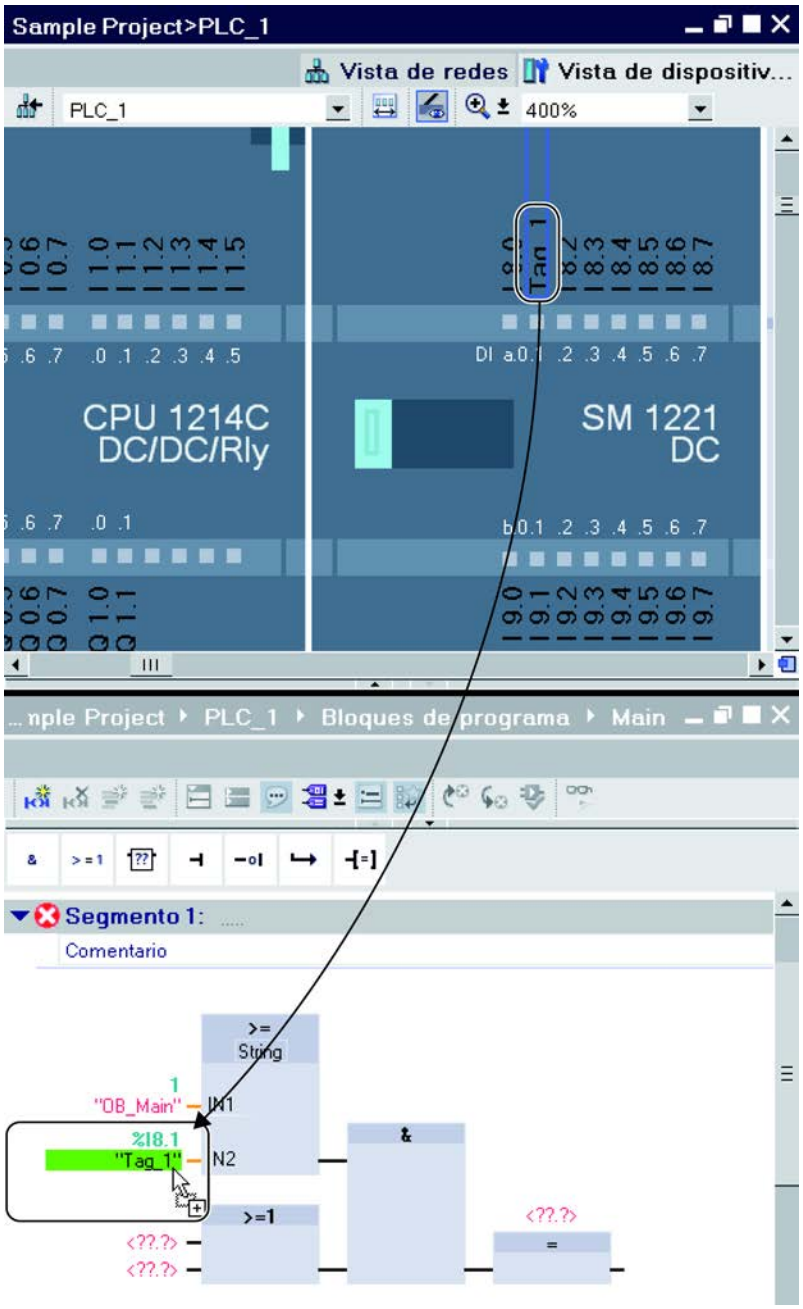
Requisitos

- Está abierta la vista de dispositivos del editor de hardware y redes.
- El factor de zoom en la vista de dispositivos debe ser del 200% como mínimo para que se vean los diferentes canales E/S.
- Están abiertas la ventana de instrucciones del editor de programación o una tabla de variables.

Procedimiento

Para asignar canales E/S de módulos a las ubicaciones del programa o a una tabla de variables, proceda del siguiente modo:

1. En la vista de dispositivos, navegue hasta el módulo que tenga el canal E/S deseado.
2. Manteniendo pulsado el botón del ratón, arrastre la dirección E/S deseada hasta la ubicación correspondiente del bloque o hasta la tabla de variables.



La dirección del módulo se asignará a la ubicación del programa, o se introducirá como variable en la tabla de variables.

Nota

La variable para una entrada o salida de un bloque también puede arrastrarse hasta la entrada o la salida de un módulo para relacionar lógicamente la variable con el canal E/S del módulo.

Signal Board

Insertar una Signal Board en la CPU

Introducción

Las Signal Boards ofrecen la posibilidad de aumentar el número de entradas y salidas propias de la CPU en las CPUs S7-1200. Como todos los demás componentes de hardware, las Signal Boards están en el catálogo de hardware. Las Signal Boards no se insertan en el rack como los demás módulos, sino directamente en un slot propio de la CPU.

Al utilizar una Signal Board hay que tener en cuenta lo siguiente:

- Cada CPU puede alojar una sola Signal Board.
- La Signal Board sólo se puede insertar si el slot de la CPU está libre.

Existen las posibilidades siguientes para insertar una Signal Board en una CPU:

- Doble clic en una Signal Board del catálogo de hardware si hay un slot libre en la CPU
- Drag & Drop del catálogo de hardware a un slot libre de la CPU
- Menú contextual de una Signal Board en el catálogo de hardware para "Copiar" y "Pegar"

Requisitos

- El catálogo de hardware está abierto.
- La CPU S7-1200 tiene un slot libre para la Signal Board.

Insertar una Signal Board en la CPU

Para insertar una Signal Board en una CPU, proceda del siguiente modo:

1. Navegue hasta la Signal Board deseada en el catálogo de hardware.
2. Seleccione la Signal Board deseada.
3. Arrastre la Signal Board hasta el slot libre de la CPU con la función Drag & Drop.

Ahora se ha insertado la Signal Board en el slot de la CPU.

Si está en la vista de redes, también puede arrastrar una Signal Board hasta un dispositivo con la función Drag & Drop. Si la CPU tiene un slot libre para una Signal Board, ésta se insertará automáticamente en dicho slot.

Configuraciones para el servidor web

Información importante sobre el servidor web

Introducción

El servidor web le proporciona la posibilidad de observar su CPU a través de Internet o de la Intranet de la empresa. Eso permite la evaluación y el diagnóstico a una gran distancia.

Los avisos y la información sobre el estado se muestran en páginas HTML.

Navegador web

Para acceder a las páginas HTML de la CPU se requiere un navegador web.

Los siguientes navegadores web para PC son aptos para la comunicación con la CPU, p. ej.:

- Internet Explorer (versión 8.0, 9.0)
- Mozilla Firefox (a partir de la versión 17.0.1)
- Google Chrome 23.0
- Apple Safari 5.1.7 (Windows)
- Apple Safari 6.0.2 (Mac)

También son aptos los siguientes navegadores web de dispositivos móviles:

- Internet Explorer 6.0 para paneles HMI
- Mobile Safari (iOS 5.0.1)
- Mobile Android Browser 2.3.4
- Mobile Google Chrome 23.0

Leer información acerca del servidor web

La información detallada a continuación podrá leerse de la CPU. La disponibilidad de las respectivas páginas web depende de la CPU y de su versión de firmware.

A partir de la versión de firmware 4, las páginas están disponibles en varios idiomas.

Página/información	Descripción
Intro/Introducción	Página de acceso para páginas web estándar
Start Page Página de inicio con información general acerca de la CPU	La página inicial incluye un resumen de la información general de la CPU, el nombre de la CPU, el tipo de CPU e información básica sobre el estado operativo actual.
Identification Información identificativa	Visualización de la información identificativa estática como el número de serie, referencia y versión.
Diagnostic Buffer Información de diagnóstico	Visualización del contenido del búfer de diagnóstico con las entradas más recientes en primer lugar.
Module Information Información del módulo	Indicación de si los componentes de una estación insertados de forma centralizada son correctos, de si hay, p. ej., solicitudes de mantenimiento o de si algunos componentes no se encuentran accesibles. A partir de la versión 4 de FW es posible actualizar el firmware a través de esta página web.
Communication Comunicación	Visualización de las conexiones de comunicación con una comunicación abierta (OUC), visualización de los recursos y los parámetros de dirección.
Varable Status Variables	Indicación del estado de los operandos del programa de usuario para observar y modificar los valores.

Página/información	Descripción
Data Logs (File Browser a partir de la versión 4 de FW)	Ficheros de datos en formato CSV para transferir al disco duro de la PG. Los ficheros de datos se crean con instrucciones Data Log en el programa de usuario y se llenan con datos. A partir de la versión 4 de FW se accede a través de la página web "File Browser" a los archivos de la memoria de carga interna y externa (Memory Card), p. ej., al contenido de los directorios "DataLogs" y "Recipes".
User Pages Páginas de usuario (en caso de que se hayan configurado y cargado páginas web definidas por el usuario)	Las páginas web del usuario proporcionan una lista de las páginas web con aplicaciones web específicas del cliente.

Acceso web a la CPU mediante PG/PC

Para acceder al servidor web, proceda de la siguiente manera:

1. Conecte el cliente (la PG/el PC) con la CPU a través de la interfaz PROFINET.
2. Abra el navegador web.
   En el campo "Dirección" del navegador web, introduzca la dirección IP de la CPU de la manera siguiente: http://ww.xx.yy.zz (entrada a modo de ejemplo: http://192.168.3.141). Se abrirá la página de inicio de la CPU. Desde allí podrá acceder al resto de las informaciones.

Consulte también

Servidor web S7-1200 (http://support.automation.siemens.com/WW/view/es/36932465)

Páginas web estándar

Requisitos para el acceso web

A continuación se explican los requisitos para el acceso a las páginas web estándar de la CPU, así como las repercusiones de la falta o la existencia de información de configuración.

Requisitos

El servidor web debe estar iniciado.

El servidor web no se inicia hasta que no se activa en las propiedades de la CPU, dentro de la sección "Servidor web".

Considere lo siguiente:

De forma estándar, las páginas web se transfieren a través de una conexión no segura y no están protegidas de los ataques de terceros. Si se desean transferir las páginas web al navegador de manera codificada, utilice la URL https://, seguida de la dirección IP de la CPU.

Iniciar sesión

Para acceder a las páginas web estándar en modo de lectura no es necesario iniciar sesión. Para realizar determinadas acciones como el cambio de estado operativo de la CPU o para acceder en escritura, es preciso que un usuario haya iniciado sesión.

CPU S7-1200 hasta la versión 3 de FW:

para las acciones mencionadas más arriba, el usuario debe haber iniciado sesión como "admin". Los campos de entrada para iniciar sesión se encuentran en la esquina superior izquierda de todas las páginas web estándar.

Al iniciar sesión como usuario "admin", deberá introducirse el nombre de usuario y la contraseña.

Nombre: admin.

Contraseña: contraseña configurada para la CPU (para CPU protegida por contraseña).

CPU S7-1200 a partir de la versión 4 de FW:

Los nombres de usuario y las contraseñas pueden elegirse a discreción (parámetro de CPU "Servidor web", área "Administración de usuarios").

Se deben asignar derechos a los usuarios, p. ej., el derecho a consultar el diagnóstico o a actualizar el firmware.

Javascript y cookies

Las páginas web estándar utilizan Javascript y cookies. Ambos deberán estar habilitados en el navegador web.

Si Javascript no está habilitado, aparecen, entre otras, las siguientes restricciones:

- Los datos de las páginas web estándar no se actualizan automáticamente.
- No es posible iniciar sesión como usuario.
- Los campos no pueden clasificarse (información del módulo)

Si las cookies no están habilitadas, no es posible iniciar sesión.

Consulte también

Acceso para HTTPS (Página 981)

Ajustes para el funcionamiento

Ajustes para el funcionamiento

Para poder utilizar el servidor web de una CPU S7-1200, hay que marcar la CPU en la vista de redes o dispositivos y realizar los siguientes ajustes en la ventana de inspección dentro de "Propiedades > General > Servidor web":

- Activar el servidor web
- Limitar los accesos a la CPU al protocolo de transferencia HTTPS (transferencia codificada) En este caso se bloquean los accesos a través del puerto 80. La comunicación sólo es posible a través del puerto 443.

- Activar la actualización automática de las páginas web
  El intervalo de actualización está predeterminado y no puede modificarse. La CPU actualiza las páginas web con contenidos variables (p. ej. información de estado o de diagnóstico) a intervalos regulares.
- Crear y gestionar usuarios
  El usuario dispondrá únicamente de las opciones que se le hayan asignado en los derechos de acceso.
  Dependiendo de la CPU y el firmware utilizados, se pueden asignar diferentes derechos de usuario. Los derechos que su CPU no soporta no pueden activarse. En la lista de usuarios se ha incluido de forma predeterminada un usuario con el nombre "Cualquiera" que posee derechos de acceso mínimos, aunque pueden ampliarse: Todo usuario que utilice el servidor web sin introducir la contraseña, tendrá los derechos de acceso del usuario "Cualquiera".
  Existe la posibilidad de parametrizar otros usuarios con derechos de acceso distintos. Estos usuarios deben iniciar sesión con el nombre de usuario y la contraseña parametrizados.

  ⚠ ADVERTENCIA

  Acceso no autorizado a la CPU a través del servidor web

  El acceso no autorizado a la CPU o la modificación de las variables PLC a valores no válidos puede causar interrupciones del proceso controlado por la CPU, así como provocar la muerte, lesiones graves o daños materiales.

  Dado que la activación del servidor web permite a personas autorizadas realizar tareas como cambiar los estados operativos, acceder a datos de la CPU en modo de escritura o actualizar el firmware, recomendamos tener en cuenta las siguientes medidas de seguridad:
  - Si es posible, limite los accesos al protocolo HTTPS.
  - Cree usuarios con contraseñas seguras. Una contraseña segura es aquella que, por ejemplo, solo se usa para una única aplicación, contiene más de 8 caracteres y está formada por letras mayúsculas y minúsculas, caracteres especiales y cifras (?!+% $1234...). Asimismo, no debe utilizarse ninguna cadena de caracteres habitual del teclado del ordenador o palabras del diccionario.
    Modifique la contraseña periódicamente.
  - No amplíe los derechos del usuario "Cualquiera".
  - Compruebe las variables PLC en el programa de usuario y limite el rango de valores a rangos admisibles, dado que los usuarios pueden ajustar valores no válidos a través del servidor web.

Acceso para HTTPS

Acceso a través de HTTPS

Un HTTP sirve para codificar y autenticar la comunicación entre el navegador y el servidor web.

Para llevar a cabo la transmisión de datos entre el navegador y la CPU a través del protocolo HTTPS, deberá introducirse la URL como https://ww.xx.yy.zz en la barra de dirección del navegador web, teniendo en cuenta que ww.xx.yy.zz hace referencia a la dirección IP de la CPU.

Para que el HTTPS pueda acceder correctamente a la CPU, es necesario tener instalado un certificado válido.

Si no hay ningún certificado instalado, aparece una advertencia con la recomendación de no utilizar la página. Para poder visualizar la página, debe seleccionarse explícitamente la opción "Agregar excepción".

En "Download certificate" de la página web "Intro" puede descargarse un certificado válido (Certification Authority) "SIMATIC CONTROLLER". En la Ayuda del navegador web correspondiente se explica cómo instalar un certificado.

Acceder a datos de la memoria de la CPU

A los datos que se encuentran en la memoria de carga interna o externa de la CPU se puede acceder a través de una página web estándar.

- En las CPU S7-1200 hasta la versión 3 de FW inclusive, utilice la página web "Data Logs". Desde esta página web se transmiten los Data Logs (registros de datos) desde la CPU hasta una unidad de su PC.
- En las CPU S7-1200 a partir de la versión 4 de FW, utilice la página web "File Browser". Desde esta página web se transmiten los datos, p. ej., desde las carpetas "Data Logs" o "Recipes", hasta una unidad de su PC.

Dependiendo del tipo de archivo y de los derechos de acceso que haya parametrizado para los usuarios del servidor web, los archivos se podrán descargar, borrar, renombrar o cargar. Los directorios propiamente dichos solo pueden crearse, borrarse o renombrarse.

Ejemplo: Data Logs

Para abrir un Data Log, hay que hacer clic en el enlace del Data Log deseado. Existe la posibilidad de abrir el archivo (.csv) p. ej. en Microsoft Excel u otro programa de su elección o de guardarlo.

Particularidad: los Data Logs están guardados en el formato CSV estadounidense, por lo que solo pueden abrirse directamente con la versión estadounidense de Microsoft Excel. En caso de disponer del programa en la versión de otro país, será necesario importar el archivo. En el asistente de importación deberá elegirse "Coma" como carácter separador.

Descargar un Data Log

Para descargar un Data Log, haga clic en el símbolo de descarga del Data Log deseado. Existe la posibilidad de abrir el archivo (.csv) p. ej. en Microsoft Excel u otro programa de su elección o de guardarlo.

Descargar y vaciar o borrar un Data Log

En una CPU con la versión de FW hasta V3.0:

Para poder descargar y borrar las entradas actuales del Data Log, es necesario haber iniciado una sesión. Para ello, es necesario hacer clic en el símbolo "Descargar y borrar" del correspondiente Data Log. Existe la posibilidad de abrir el archivo (.csv) p. ej. en Microsoft Excel u otro programa de su elección o de guardarlo.

En una CPU con la versión de FW a partir de V4.0:

Para inicializar el Data Log, proceda del siguiente modo:

1. Abra el archivo CSV, p. ej. con Excel.
2. Borre las filas entre la fila de título y la fila con la entrada "//END", si existe.
3. Guarde el archivo en una unidad de su PC.
4. Borre el Data Log (es decir, el archivo CSV) en la página web "File Browser" y cargue el archivo CSV preparado con el botón "Upload file" de la página web "File Browser" en la CPU.

Encontrará más información en el manual de sistema de la CPU S7-1500.

Consulte también

Data Logging - Übersicht (Página 4970)

Crear y cargar páginas web definidas por el usuario

Información importante sobre las páginas web definidas por el usuario

Concepto

El concepto de páginas web definidas por el usuario permite acceder con un navegador web a las páginas web de la CPU diseñadas por el usuario. El servidor web de la CPU pone a disposición esta función.

Para configurar la apariencia y funcionalidad de las páginas web definidas por el usuario no se requieren herramientas especiales. Puede utilizarse CSS para adaptar el diseño de las páginas, Javascript para incluir contenido dinámico o cualquier otro framwork que el usuario desee para crear las páginas web.

El conjunto de archivos que procesa el servidor web también recibe el nombre de "aplicación web".

Aplicación web y programa de usuario

Utilizando código HTML especial en páginas web definidas por el usuario, también pueden transferirse datos al programa de usuario de la CPU a través de un navegador web para su posterior procesamiento, así como visualizarse datos del área de operandos de la CPU en el navegador web.

Con instrucciones de script (p. ej. Javascript) puede optimizar sus páginas web, p. ej. modificar contenidos de forma dinámica o validar entradas de usuario.

Para sincronizar el programa de usuario y el servidor web, así como para la inicialización, es necesario llamar la instrucción WWW (SFC 99) del programa de usuario.

- Si no hace falta que la aplicación web y el programa de usuario interactúen, p. ej. si una página web sólo ofrece información estática, sólo es necesaria una inicialización en el programa de usuario.
- Si se requiere un intercambio de datos sencillo entre las variables PLC y las variables de la aplicación web, p. ej. para mostrar el contenido de las variables PLC o para escribir un valor en una variable PLC, deberá tener en cuenta la sintaxis para la lectura y escritura de variables. Asimismo, en el programa de usuario se requiere en este caso tan solo una inicialización, p. ej. en el OB de arranque.
- Si se requiere una mayor interacción entre la aplicación web y el programa de usuario, deberá manipular, además de la sincronización entre el servidor web y el programa de usuario, la información de estado y de control del DB Web Control. Esto será necesario, p. ej., en el caso de entradas de usuario que se transfieran al servidor web a través del navegador web para que la CPU las evalúe. A diferencia del intercambio de datos sencillo, el programa de usuario influye directamente en el momento en el que se devuelve al navegador web la página web solicitada. En este caso, es necesario que el usuario se familiarice con el concepto de los fragmentos manuales y las estructuras del DB Web Control.

Inicialización

Las páginas web definidas por el usuario se "comprimen" en bloques de datos para que la CPU pueda procesarlas. Durante la configuración, deben generarse los correspondientes bloques de datos a partir de los datos de origen (archivos HTML, imágenes, archivos Javascript, etc.) para poder cargar la aplicación web en la CPU. El DB Web Control (predeterminado: DB 333), que contiene información de estado y de control, así como referencias a otros bloques de datos con páginas web encriptadas, desempeña una función muy importante. Los bloques de datos que contienen las páginas web encriptadas reciben el nombre de "DBs de fragmento".

Si los bloques de datos se han cargado en la CPU, esta no "sabe" que las páginas web definidas por el usuario que contiene están cifradas. Con la instrucción "WWW" (SFC 99), p. ej. en un OB de arranque, se le informa de qué DB es el DB Web Control. Tras esta inicialización, las páginas web definidas por el usuario son accesibles desde un navegador web.

Sincronización

Si el programa de usuario debe intercambiar datos o interactuar con las páginas web definidas por el usuario, deberá utilizarse la instrucción WWW (SFC 99) en la parte cíclica del programa.

Ejemplos de interacción entre programa de usuario y página web:

- Comprobar los datos recibidos
- Recopilar y reenviar los datos para el navegador web solicitante

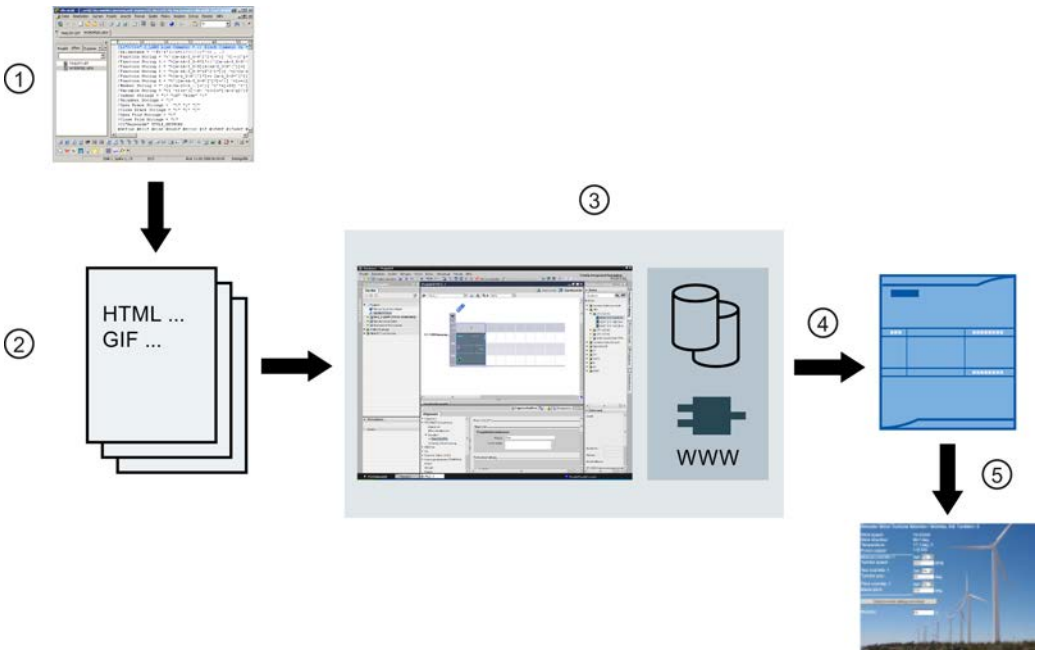
En este caso, la información de estado debe ser evaluable en el momento y la información de control debe comunicarse al servidor web, p. ej., para habilitar una página web solicitada.

Vista general del procedimiento

Principios básicos

En este capítulo se describe paso a paso el procedimiento fundamental que hay que seguir para crear, cargar y utilizar en la fase operativa páginas web definidas por el usuario.

El siguiente gráfico muestra el proceso para la creación y visualización de páginas web definidas por el usuario de forma simplificada:



① Programación de una aplicación web (si es necesario, utilizando herramientas adecuadas y, si se da el caso, con los comandos AWP incluidos para páginas dinámicas).

② La aplicación web se compone de distintos archivos de origen, p. ej. *.html, *.gif, *.js, etc.

③ Con STEP 7:

- Generar bloques de datos (DB de Web Control y DBs de fragmento) de los archivos de origen. Los DBs contienen metainformación y la aplicación web completa, incluidas las imágenes y las partes dinámicas y estáticas de la aplicación web. Los DBs se almacenan en el árbol del proyecto, dentro de "Bloques de sistema".
- Llamar la instrucción "WWW" en el programa de usuario. La instrucción inicializa el servidor web de la CPU para una aplicación web.
- Si es necesario, desprogramar la interacción entre el servidor web y el programa de usuario.

④ Cargar los bloques en la CPU.

⑤ Abrir la página web en el navegador web. La llamada de las páginas web de la CPU se realiza introduciendo la dirección IP de la CPU.

Información adicional

Encontrará más información y ejemplos sobre el servidor web de la S7-1200 en Internet (http://support.automation.siemens.com/WW/view/es/36932465).

Crear páginas web

Para crear páginas web definidas por el usuario pueden utilizarse herramientas de diseño web de distintos proveedores. Básicamente, las páginas web deben programarse y diseñarse según las convenciones del W3C (World Wide Web Consortium). En el servidor web de la CPU no se comprueba si se cumplen los criterios del W3C.

Reglas

- La herramienta debe ofrecer la posibilidad de editar el código HTML directamente para que los comandos AWP puedan insertarse en la página HTML.
  En la CPU sólo se analizan sintácticamente los comandos AWP y, por ejemplo, se sustituyen por valores del programa de usuario / memoria imagen de proceso de la CPU.
- Los archivos que contienen comandos AWP deben estar codificados con UTF-8. Por esta razón, ajuste el atributo charset a UTF-8 en los metadatos de la página HTML y guarde el archivo codificado en UTF-8.
- Los archivos que contienen comandos AWP no pueden contener la siguiente secuencia:]]
- Los archivos que contienen comandos AWP no pueden contener fuera de "Rangos de lectura variables" (:=<Nombre de la variable>:) la siguiente secuencia: :=
  Sugerencia: Sustituya el primer carácter de una secuencia prohibida por su código de carácter; para el carácter de dos puntos p. ej. :.

Un pequeño ejemplo de página web definida por el usuario pretende ilustrar la estructura básica.

Requisitos

- La CPU debe disponer de un servidor web y éste debe estar activado.
- El usuario debe haber iniciado sesión como "admin" para acceder en modo de escritura a las variables PLC como usuario de páginas web definidas por el usuario.
- Para el ejemplo inferior, deben haberse definido variables PLC para las variables PLC que deban mostrarse en la página web. Aquí se muestra el ejemplo de la primera variable PLC utilizada "Tank_below_max".

		Nombre	Tipo de datos	Dirección
1		Tank_below_max	Bool	%I0.0

Crear páginas web definidas por el usuario

El siguiente código para un ejemplo de página web lee valores de la memoria imagen de proceso y los representa en una tabla.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/
TR/html4/strict.dtd">
<html>
  <head>
   <meta http-equiv="Content-Type" content="text/html;
charset=utf-8">
   <title>Mix</title>
  </head>
   <body>
```

```
    <h1>Mix</h1>
    <h2> Actual State </h2>
    <table border="1">
     <tr>
      <th>Variable</th>
      <th>State</th>
     </tr>
     <tr>
      <td>Tank below max</td>
      <td>:="Tank_below_max":</td>
     </tr>
     <tr>
      <td>Tank above min</td>
      <td>:="Tank_above_min":</td>
     </tr>
    </table>
   </body>
</html>
```

Comandos AWP

Con los comandos AWP (Automation Web Programming) se declara la interfaz entre una aplicación web libremente programable para una CPU que cuenta con un servidor web y los datos de la CPU.

Para el desarrollo de aplicaciones web sólo existen las limitaciones que especifican los navegadores web. Con el programa de usuario se controla en uno de los lenguajes de programación de STEP 7 qué datos de la CPU deben mostrarse en el navegador web del visitante y en qué momento. Utilizando los comandos AWP, que se anotan en los archivos HTML, se acuerdan datos para la interacción pretendida entre la aplicación web y el programa de usuario.

Los comandos AWP se insertan en los archivos HTML como comentarios HTML con una sintaxis especial y declaran las siguientes características:

- leer variables PLC
- escribir variables PLC
- leer variables especiales
- escribir variables especiales
- definir los tipos Enum
- asignar variables a los tipos Enum
- definir fragmentos
- importar fragmentos

Sintaxis de los comandos AWP

Un comando AWP empieza por "`<!--AWP_`" y acaba por "`-->`". En los archivos de Javascript, los comandos deben estar incluidos en comentarios de Javascript ("`/*...*/`").

Reglas de notación para los nombres de variables PLC dentro de un comando AWP

Los comandos AWP "AWP_In_Variable" y "AWP_Out_Variable" contienen un atributo de nombre y, de forma opcional, un atributo de uso. Estos atributos tienen asignado un nombre de variable PLC a través del cual se escriben o leen variables PLC en el navegador. Para el uso de nombres de variables PLC en código HTML existen las siguientes reglas:

- Las variables PLC deben indicarse entre comillas (" ... ").
- En los comandos AWP, las variables PLC utilizadas deben indicarse, además de entre comillas simples ('" ... "') o con una barra invertida (Backslash), con comillas enmascaradas ("\" ... \"").
- Si el nombre de la variable PLC contiene el carácter \ (barra diagonal inversa), este carácter debe identificarse con la secuencia de escape \\ como carácter normal del nombre de la variable PLC.
- Si el nombre de la variable PLC en el comando AWP se encuentra adicionalmente entre comillas simples y dentro del nombre aparece la comilla simple ('), debe identificarse asimismo con la secuencia de escape \' como carácter normal.
- Si en el comando AWP se utiliza una dirección absoluta (entrada, salida, marca), deberá ir entre comillas simples.

Variable PLC	Variable PLC en código HTML
"Velocity"	<!-- AWP_In_Variable Name='"Velocity"' -->
	<!-- AWP_In_Variable Name="\"Velocity\"" -->
"abc\de"	<!-- AWP_In_Variable Name='"abc\\de"' -->
"abc'de"	<!-- AWP_In_Variable Name='"abc\'de"' -->
"abc'de"	<!-- AWP_In_Variable Name="abcde" Use'"abc\'de"' -->
"DB name".tag	<!-- AWP_In_Variable Name='"DB name".tag' -->
"DB name"."ta.g"	<!-- AWP_In_Variable Name='"DB name"."ta.g"' -->
-	<!-- AWP_Out_Variable Name=’flag1’ Use='M0.0' -->

Consulte también

Leer variables (Página 988)

Escribir variables (Página 990)

Variables especiales (Página 991)

Leer variables

Las páginas web definidas por el usuario pueden leer variables PLC.

Cada variable PLC debe estar especificada con un nombre de variable PLC.

Estas variables Out (dirección de salida vista desde el controlador) se introducen en un lugar cualquiera del texto HTML con la sintaxis que se describe a continuación.

Sintaxis

```
:=<varname>:
```

Durante el funcionamiento, el servidor web sustituye estas referencias por los correspondientes valores actuales de las variables PLC.

`<varname>` puede ser una variable PLC general sencilla, pero también una ruta completa de variables hacia un elemento de una estructura.

Reglas de notación para los nombres de variables PLC

- Las variables PLC en código HTML deben ir entre comillas (") si han sido definidas en la tabla de variables. En el caso de las variables de bloques de datos, el nombre del bloque de datos debe ir entre comillas. Si se utilizan caracteres especiales en los elementos de estructura del bloque de datos tales como, p. ej., el punto (.) o el espacio en blanco, esta parte debe ir asimismo entre comillas.
- No se utilizan las comillas para las direcciones absolutas de entradas, salidas o marcas.

Variable PLC	Variable PLC en código HTML
"DB_name".var_name	:="DB_name".var_name:
"DB_name"."var.name"	:="DB_name"."var.name":
"memory"	:="memory":
-	:=I0.0:
	:=Q0.0:
	:=MW100:
	:=%MW100:
"My_Data_Block".flag1	<!-- AWP_Out_Variable Name='flag1' Use='"My_Data_Block".flag1' -->
	...
	:=flag1:

- Si el nombre de variable PLC contiene los caracteres: (dos puntos) o \ (barra diagonal inversa), estos caracteres deben identificarse con la secuencia de escape \: o \\ como carácter normal del nombre de la variable PLC.

Variable PLC	Variable PLC en código HTML
"abc:de"	:="abc\:de":
"abc\de"	:="abc\\de":

- Caracteres especiales "<, &, >"
  Si aparecen estos caracteres en el nombre de la variable (p. ej. "a<b"), pueden darse problemas de representación.
  Evite expresiones tales como, p. ej. :="a<b": en la página HTML.
  A fin de evitar problemas de representación, utilice p. ej. un comando AWP con una expresión Use siguiendo el patrón que se muestra a continuación. El atributo Use define la variable PLC con los caracteres problemáticos, el atributo Name define el nombre sin caracteres problemáticos, tal y como se utiliza en la página HTML.

Variable PLC	Variable PLC en código HTML
"a<b"	<!—AWP_Out_Variable Name='simplename' Use='"a<b"' -->
	...

Variable PLC	Variable PLC en código HTML
	:=simplename:

Consulte también

Comandos AWP (Página 987)

Escribir variables

Las páginas web definidas por el usuario pueden escribir datos en la CPU.

Para ello, se requiere un comando AWP que identifique la variable PLC que hay que escribir.

Además, cada variable PLC debe estar especificada con un nombre de variable PLC.

Las variables In (dirección de entrada vista desde el controlador) se activan en la página del navegador. Esto puede producirse, por ejemplo, en el caso de los formularios.

El navegador activa las variables con Request de HTTP ya sea en el encabezado de HTTP (por cookie o método POST) o en la URL (método GET) y, a continuación, el servidor web las escribe en la correspondiente variable PLC.

Sintaxis

Para que las variables In puedan escribirse en la CPU, es necesario definirlas con una instrucción AWP explícita:

```
<!-- AWP_In_Variable Name='<PLC_Varname1>' Name='<PLC_Varname2>'
Name='<PLC_Varname3>' -->
```

En una instrucción puede haber varias variables definidas (como puede verse arriba).

El nombre concreto de la variable PLC se escribe entre comillas dobles; p. ej. <PLC_Varname1> = "myVar".

En caso de que el nombre de la variable que utilice para la aplicación web no sea idéntico al nombre de la variable PLC, puede realizarse la asignación a una variable PLC con el parámetro "Use":

```
<!-- AWP_In_Variable Name='<Webapp_Varname>' Use='<PLC_Varname>'
```

Ejemplo

El comando AWP "AWP_In_Variable" es imprescindible, especialmente para el tratamiento de formularios.

```
<form method='post' action='/awp/appl/x.html'>
 <p>
  <input name='"var1"' type='text'>
  <input value='set' name='Button1' type='submit'>
 </p>
</form>
```

En el formulario antes definido, se transmite la variable "var1" al servidor web con el método Request de HTTP "post". El usuario activa la variable "var1" en el campo del formulario. La variable 'Button1' tiene el valor 'set', pero no es necesaria para la CPU. Para que la variable

"var1" pueda escribirse en la CPU, debe incluirse la siguiente instrucción en el mismo fragmento:

```
<!-- AWP_In_Variable Name='"var1"' -->
```

Puesto que las variables PLC deben ir entre comillas dobles ("), el nombre en el comando AWP debe ir entre comillas simples (') o entre comillas dobles con una barra inversa (\"). Para evitar las numerosas secuencias Escape, es recomendable utilizar comillas simples.

```
<!-- AWP_In_Variable Name=’"Info".par1’ -->
<!-- AWP_In_Variable Name="\"Info".par1\"" -->
```

Condiciones para el acceso en escritura durante el funcionamiento

Para que un usuario pueda escribir en variables PLC desde una página web definida por el usuario, deben cumplirse los siguientes requisitos:

- La CPU está protegida por contraseña.
- El usuario ha iniciado una sesión de "admin".

Esta regla es válida en principio para todos los accesos en escritura de páginas web a una CPU.

Consulte también

Requisitos para el acceso web (Página 978)

Comandos AWP (Página 987)

Variables especiales

Las variables especiales son principalmente las variables HTTP que aparecen en las definiciones del World Wide Web Consortium (W3C) . También se utilizan variables especiales para las cookies y las variables de servidor.

Los comandos AWP para leer y escribir variables especiales sólo se diferencian por los parámetros adicionales de los comandos AWP para leer y escribir variables normales.

Leer variable especial

El servidor web puede leer variables PLC y entregarlas a variables especiales en el encabezado de respuesta HTTP. Puede p. ej. leer una URL para un desvío a otra página web y entregarla a la variable especial HEADER:Location sirviéndose de la variable especial HEADER:Location.

Pueden leerse las siguientes variables especiales:

Nombre	Descripción
COOKIE_VALUE:name	Valor de la cookie con el nombre: "name"
COOKIE_EXPIRES:name	Tiempo de ejecución de la cookie con el nombre: "name" en segundos (debe haberse activado con antelación).

Nombre	Descripción
HEADER:Status	Código de estado HTTP (si no se ha introducido ningún otro valor, se devolverá el código de estado 302).
HEADER:Location	Ruta para la derivación a otra página. El código de estado 302 debe estar activado.
HEADER:Retry-After	Tiempo durante el que previsiblemente el servicio no estará disponible. El código de estado 503 debe estar activado.
HEADER: …	El resto de las variables de encabezado también puede transmitirse de este modo.

Con el comando AWP "AWP_Out_Variable" se determinan las variables PLC que deben transferirse al navegador web en el encabezado de HTTP.

Estructura general:

```
<!-- AWP_Out_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Descripción de parámetros

- Name: tipo y nombre de la variable especial
- Use (parámetro opcional): en caso de que el nombre de la variable especial no sea idéntico al nombre de la variable PLC, puede realizarse la asignación a una variable PLC con el parámetro "Use".

Ejemplo:

```
<!-- AWP_Out_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

Escribir una variable especial

En principio, el programa de usuario de la CPU puede evaluar todas las variables HTTP escritas por el navegador web en el encabezado de HTTP. Ejemplos de tipos de variables:

Nombre	Descripción
HEADER:Accept-Language	Idioma aceptado o preferente
HEADER:Authorization	Prueba de autorización para un recurso solicitado
HEADER:Host	Host y puerto del recurso solicitado
HEADER:User-Agent	Información sobre el navegador
HEADER: …	El resto de las variables de encabezado también puede transmitirse de este modo
SERVER:current_user_id	Indica si un usuario ha iniciado una sesión (current_user_id=0: ningún usuario ha iniciado una sesión)

Nombre	Descripción
SERVER:current_user_name	Nombre del usuario que ha iniciado la sesión
SERVER:GET	El método Request es GET
SERVER:POST	El método Request es POST
COOKIE_VALUE:name	Valor de la cookie con el nombre: "name"

Con el comando AWP "AWP_In_Variable" se determinan las variables especiales que deben evaluarse en el programa de usuario de la CPU.

Estructura general:

```
<!-- AWP_In_Variable Name="<Typ>:<Name>" [Use="<Varname>"] -->
```

Descripción de parámetros:

Name: tipo y nombre de la variable especial

Use (parámetro opcional): en caso de que el nombre de la variable especial no sea idéntico al nombre de la variable PLC, puede realizarse la asignación a una variable PLC con el parámetro Use .

Ejemplos:

```
<!-- AWP_In_Variable Name="COOKIE_VALUE:siemens" Use='"info".language' -->
```

El nombre de la variable en el encabezado de HTTP se sustituye por el nombre de la variable PLC indicado en Use . La cookie se escribe en la variable PLC "info".language .

```
<!-- AWP_In_Variable Name='COOKIE_VALUE:siemens' Use='"info".language' -->
```

El nombre de la variable en el encabezado de HTTP se sustituye por el nombre de la variable PLC indicado en Use. La cookie se escribe en la variable PLC "info".language .

```
<!-- AWP_In_Variable Name='"COOKIE_VALUE:siemens"' -->
```

La variable del encabezado HTTP se escribe en la variable PLC del mismo nombre.

Consulte también

Comandos AWP (Página 987)

Tipos de enumeraciones

Tipos de "enumerations" (Enums)

Con la ayuda de los Enums, los valores numéricos del programa del PLC pueden convertirse en texto y viceversa. La asignación de los valores numéricos también puede realizarse para más de un idioma.

Crear Enums

Introduzca un comando AWP utilizando la siguiente sintaxis al principio del archivo HTML:

```
<!-- AWP_Enum_Def Name="<Nombre del tipo de Enum>"
Values='0:"<Text_1>", 1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Para guardar, por ejemplo, valores alemanes como archivo HTML en la carpeta "de" del directorio HTML:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"an", 1:"aus", 2:"Störung"' -->
```

Para guardar, por ejemplo, valores ingleses como archivo HTML en la carpeta "en" del directorio HTML:

```
<!-- AWP_Enum_Def Name="Enum1" Values='0:"on", 1:"off", 2:"error"' -->
```

Asignar Enums

La asignación de las variables del programa de usuario a cada uno de los textos Enum se realiza con un comando AWP propio:

```
<!-- AWP_Enum_Ref Name="<VarName>" Enum="<EnumTypeName>" -->
```

<VarName> representa el nombre simbólico del programa de usuario y <EnumTypeName>, el nombre definido anteriormente del tipo Enum.

Nota

En todos los fragmentos en los que una variable PLC referencie textos Enum, esta variable PLC debe asignarse junto con el correspondiente comando AWP al nombre del tipo de Enum.

Se debe tener en cuenta que entre una asignación Enum y el uso de Enum no existe ningún comando AWP para importar fragmentos debido a que esta importación hace que la asignación Enum se encuentre en otro fragmento que el uso de Enum.

Ejemplo

Se ha definido un tipo de Enum "estado" con los valores "0" y "1". "0" equivale a "off", mientras que "1" equivale a "on":

```
<!-- AWP_Enum_Def Name="estado" Values='0:"off", 1:"on"' -->
```

El código HTML de la página web que debe visualizarse contiene el siguiente código:

```
<!-- AWP_Enum_Ref Name="estado operativo" Enum="estado" -->
:=estado operativo:
```

En función del valor de la variable "estado operativo", ya no se muestra "0" ó "1" como resultado, sino "off" u "on".

Uso simplificado de tipos de enumeración

En las CPU S7-1200 a partir de la versión 4 del firmware es posible emplear enumeraciones directamente en comandos AWP para leer y escribir variables PLC.

Cree enumeraciones como se ha descrito en el párrafo anterior para poder utilizar los valores con comandos AWP de lectura y escritura.

Crear Enums

```
<!-- AWP_Enum_Def Name="<Name des Enum Typs>" Values='0:"<Text_1>",
1:"<Text_2>", ... , x:"<Text_x>"' -->
```

Uso de Enums en comandos AWP de escritura y lectura

```
<!-- AWP_In_Variable Name='<Varname>' Enum="<EnumType>" -->
<!-- AWP_Out_Variable Name='<Varname>' Enum="<EnumType>" -->
```

Ejemplo de lectura de variables PLC

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_Out_Variable Name='"Alarm"'
Enum="AlarmEnum" -->...<p>The current value of "Alarm"
is :="Alarm":</p>
```

Si el valor de "Alarma" está en la CPU "2", en la página HTML se muestra el siguiente texto:

'The current value of "Alarm" is Tank is empty' porque la definición de Enum asigna al string "Tank is empty" el valor numérico 2.

Ejemplo de escritura de variables PLC

```
<!-- AWP_Enum_Def Name='AlarmEnum' Values='0:"No alarms", 1:"Tank is
full", 2:"Tank is empty"' --><!-- AWP_In_Variable Name='"Alarm"'
Enum='AlarmEnum' -->...
<form method="POST">
<p><input type="hidden" name='"Alarm"' value="Tank is full" /></p>
<p><input type="submit" value='Set Tank is full' /><p>
</form>
```

Como la definición de Enum asigna al string "Tank is full" el valor numérico "1", el valor "1" se escribe en la variable PLC"Alarm".

Definir fragmentos

Fragmentos

Se entiende por fragmentos cada una de las "unidades de significado" de una página web que debe procesar la CPU.

Los fragmentos suelen ser páginas enteras, aunque también pueden ser elementos individuales como archivos (p. ej. imágenes) o documentos enteros.

Definir fragmentos

```
<!-- AWP_Start_Fragment Name="<Name>" [Type="<Type>"] [ID="<Id>"]
[Mode=<Mode>]-->
```

Con este comando se define el inicio de un fragmento. Un fragmento se extiende hasta el principio del siguiente fragmento o hasta el final del archivo.

- <Name> indica el nombre del fragmento.
  El nombre debe comenzar con una letra [a-zA-Z] o un carácter de subrayado (_). Tras este primer carácter pueden aparecer letras, caracteres de subrayado o números [0-9].
- <Type> indica el tipo de fragmento.
  - "manual" El programa de usuario se informa mediante el Request de un fragmento de que es posible realizar cambios en la página web que debe devolverse con el programa de usuario.
  - "automatic" La página se edita automáticamente (por defecto).

- <id> para el fragmento puede predefinirse una ID numérica. Si no se especifica ninguna ID, se asignará automáticamente una ID al fragmento. En el caso de páginas manuales (<Type>=manual) puede accederse al fragmento en el programa de usuario de la CPU con esta ID.

  Nota

  Es necesario establecer una ID baja, ya que la ID más alta condiciona el tamaño del DB Web Control.

- <Mode> Los fragmentos son compatibles con ambos modos: visible y hidden.
  - "visible": el fragmento forma parte de la página web. Este modo está predeterminado y puede omitirse.
  - "hidden": el fragmento no forma parte de la página web, pero se guarda en el DB web y está disponible para que el programa de usuario lo agregue a una página web solicitada. Intercambiando los ID de fragmento (Variable del DB Web Control fragment_index) se agrega un fragmento "oculto" a la página web solicitada.

Con el comando "AWP_Start_Fragment", el documento Input se divide completamente en fragmentos. Por este motivo, no es necesario un "AWP_End_Fragment".

Sin un comando de inicio de fragmento, un archivo se reproduce como un fragmento cuyo nombre se deriva del nombre del archivo. Si un archivo se descompone en varios fragmentos (mediante "AWP_Start_Fragment"), el archivo debe empezar por el comando "AWP_Start_Fragment".

Importar fragmentos

Existe la posibilidad de declarar un fragmento en una página HTML e importar dicho fragmento a otras páginas web.

Ejemplo

El logotipo de un empresa debe mostrarse en todas las páginas web de una aplicación web.

El código HTML para el fragmento que muestra el logotipo de la empresa sólo existe una vez. El fragmento puede importarse tantas veces y a tantos archivos HTML como sea necesario.

Sintaxis

```
<!-- AWP_Import_Fragment Name = "<name>"-->
```

- <name> es el nombre del fragmento que se desea importar.

Ejemplo

Código HTML dentro de una página web que declara un fragmento:

```
<!-- AWP_Start_Fragment Name = "My_Company_Logo"-->
<p><img src = "compay_logo.jpg"></p>
```

Ejemplo

Código HTML dentro de otra página web que importa el fragmento declarado:

```
<!-- AWP_Import_Fragment Name = "My_Company_Logo"-->
```

Generar y cargar bloques de datos

Requisitos

- Se han creado todos los archivos de origen necesarios (*.html, *.js, *.png...) para la aplicación web.
- Los archivos de origen están en una carpeta, pero sólo los archivos de origen que son necesarios para la aplicación web. En esta carpeta no debe haber otros archivos.

Nota

Longitud de los nombres de archivos y variables

Si tiene una aplicación web amplia, con muchos archivos y directorios, es posible que fracase la generación de los bloques de datos web. En este caso, la generación se cancela con el mensaje "Lista de textos desbordada...". La causa son limitaciones internas del tamaño de la información administrativa que se guardará en el bloque de datos web.

Solución: utilice nombres de archivo y nombres de variable cortos.

Procedimiento

Para generar bloques de datos a partir de los archivos de origen para las páginas web definidas por el usuario en STEP 7, proceda del siguiente modo:

1. Seleccione la CPU, p. ej. en la configuración de dispositivos.
2. Seleccione las propiedades para las páginas web definidas por el usuario "Propiedades > General > Servidor web" de la ventana de inspección.

3. Como "fuente HTML" debe seleccionarse la carpeta en la que se encuentran los archivos de origen para la aplicación web.
4. Como página de inicio HTML debe introducirse la página HTML que deba abrirse al iniciar la aplicación web.
5. Si es necesario, introduzca el nombre de una aplicación.
6. Si es necesario, complemente la serie de extensiones de nombres de archivos como "archivos con contenido dinámico". Sólo deben introducirse las extensiones de nombres de archivos que contengan comandos AWP.
7. El número del DB Web Control y del inicio del DB de fragmento pueden conservarse si no están ocupados por el programa de usuario.
8. Haga clic en el botón "Generar" para generar DBs a partir de los archivos de origen. Los bloques de datos generados se depositan en la carpeta "Bloques de sistema" (subcarpeta "Servidor web") del árbol del proyecto.
9. En la vista de redes, seleccione la CPU que debe cargarse y elija el comando "Cargar en dispositivo" en el menú "Online" para cargar los bloques. Antes de la carga se iniciará implícitamente la compilación de los bloques.
   En caso de que se notifiquen fallos durante este proceso, deben solucionarse antes de cargar la configuración.

Estructura del programa del PLC

El programa de usuario debe llamar la instrucción " WWW" para que la aplicación web, es decir, las páginas web definidas por el usuario, estén disponibles en las páginas web estándar de la CPU y puedan abrirse.

El DB de Web Control generado a partir de los archivos de origen es el parámetro de entrada (CTRL_DB) para la instrucción "WWW". El DB de Web Control referencia el contenido de las páginas web definidas por el usuario que está codificado en los DBs de fragmento y contiene información de estado y control.

Llamada de la instrucción "WWW" en el programa de arranque

Para evitar que el programa de usuario no influya en páginas web solicitadas, basta con llamar una sola vez la instrucción "WWW" en un OB de arranque. La instrucción inicializa la comunicación entre el servidor web y la CPU.

Llamada de la instrucción "WWW" en el programa cíclico

La llamada de la instrucción "WWW" también puede producirse en un OB procesado cíclicamente (p. ej. el OB 1). Esto ofrece la ventaja de que pueden aceptarse solicitudes del servidor web en el programa de usuario. Para ello, deben utilizarse fragmentos manuales.

En este caso, hay que evaluar información del DB de Web Control para identificar la página web solicitada o el fragmento solicitado. Por otra parte, es necesario activar un bit en el programa de usuario para habilitar explícitamente la página web que debe proporcionar el servidor web después de procesar la solicitud de página web.

La estructura del DB de Web Control se describe en el siguiente apartado.

DB de Web Control

El DB de Web Control (por defecto, DB 333) es generado por STEP 7 y contiene información sobre la estructura de las páginas de usuario, el estado de la comunicación y posibles errores producidos.

Además del DB de Web Control, también se crean los DBs de fragmento. Estos DBs de fragmento (también puede ser un solo DB de fragmento) se referencian en el DB de Web Control. Los DBs de fragmento contienen las páginas web y los datos de medios (p. ej. imágenes) codificados en fragmentos. El contenido de los DBs de fragmento no puede modificarse con el programa de usuario. Se crean automáticamente y sólo sirven para la gestión de datos.

A las variables de estado y control del DB de Web Control se accede de forma simbólica.

A continuación, se indican las variables del DB de Web Control necesarias para la evaluación de estado y el control de la interacción.

Pueden obtenerse dos tipos de información del DB de Web Control:

- Información de estado general: información no vinculada a ninguna solicitud de página web concreta (Request).
- Información de estado y control de Request: información sobre Requests pendientes.

Información de estado general

"WEB-Control_DB".commandstate.init	Activa e inicializa la aplicación web.
"WEB-Control_DB".commandstate.deactivate	Desactiva la aplicación web.
"WEB-Control_DB".commandstate.inititializing	Se inicializa la aplicación web (leer DB de Web Control, etc.).
"WEB-Control_DB".commandstate.error	No se ha podido inicializar la aplicación web. El motivo está codificado in "WEB-Control_DB".commandstate.last_error .
"WEB-Control_DB".commandstate.deactivating	Se finaliza la aplicación web.
"WEB-Control_DB".commandstate.initialized	Se ha inicializado la aplicación web y ésta ya está lista.
"WEB-Control_DB".commandstate.last_error	Tabla de valores de posibles errores (véase tabla siguiente).

Last_error	Descripción
1	El DB de fragmento es incoherente (no es compatible con el DB de Web Control).
2	Ya existe una aplicación web con este nombre.
3	Problema de memoria al inicializar en el servidor web.
4	Los datos del DB de Web Control son incoherentes.
5	Un DB de fragmento no está disponible (no está cargado).
6	Un DB de fragmento no tiene identificación AWP.
7	El fragmento Enum (contiene los textos e información para los tipos Enum) no está disponible.

Last_error	Descripción
8	Una acción solicitada a través de las marcas de comando en el DB de Web Control no está permitida en el estado actual.
9	La aplicación web no está inicializada (si no se produce una nueva inicialización tras la desactivación).
10	El servidor web está desactivado.
...	Se desactiva Last_error si la aplicación web se ha inicializado correctamente.

Información de estado de Request

La información de estado de Request está vinculada a uno de los cuatro Requests posibles, x = [1 … 4].

"WEB-Control_DB".requesttab[x].idle	No hay que hacer nada.
"WEB-Control_DB".requesttab[x].waiting	El programa de usuario debe responder a un Request de un fragmento manual e iniciar explícitamente el procesamiento posterior en el servidor web.
"WEB-Control_DB".requesttab[x].sending	El servidor web está ocupado con el procesamiento de Requests/fragmentos.
"WEB-Control_DB".requesttab[x].aborting	El servidor web cierra la conexión TCP.

Información de control de Request

La información de control de Request está vinculada a uno de los cuatro Requests posibles, x = [1 … 4].

"WEB-Control_DB".requesttab[x].continue	Habilita el fragmento que se acaba de procesar para su transmisión. Se inicia el procesamiento del siguiente fragmento.
"WEB-Control_DB".requesttab[x].repeat	Habilita el fragmento que se acaba de procesar para su transmisión. A continuación, el fragmento se procesa de nuevo.
"WEB-Control_DB".requesttab[x].abort	Cierra la conexión TCP.
"WEB-Control_DB".requesttab[x].finish	Habilita el fragmento que se acaba de procesar para su transmisión. Detiene el procesamiento posterior de fragmentos (finaliza el Request).

Ejemplo:

La variable para el DB es la siguiente: "WEB-Control_DB". Con el bit "WEB-Control_DB".commandstate.error del programa de usuario puede consultarse si se han producido errores durante la inicialización de la aplicación web.

En caso de que se haya producido un error, éste puede analizarse con el valor de "WEB-Control_DB".commandstate.last_error.

Interacción con el programa de usuario

Con la ayuda de los fragmentos manuales, se consigue que el programa de usuario reaccione a las entradas del navegador de forma sincrónica y pueda preparar la página web que se debe devolver.

Tipo de fragmento

Debe utilizarse el tipo de fragmento "manual" (para "páginas manuales") para el fragmento que escribe los datos con el fin de poder reaccionar ante los datos recibidos en el programa de usuario:

```
<!-- AWP_Start_Fragment Name="testfrag" ID="1" Type="manual" -->
```

La transmisión de los valores al servidor web de la CPU siempre se realiza del mismo modo en las páginas automáticas y manuales:

Ejemplo:

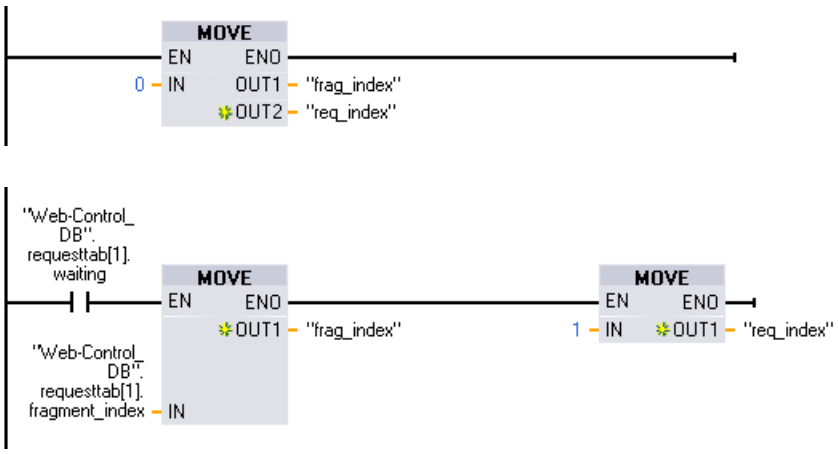
```
<form method="POST" action="">
<p>
<input type="submit" value="Set new value">
<input type="text" name='"Velocity"' size="20">
</p>
</form>
```

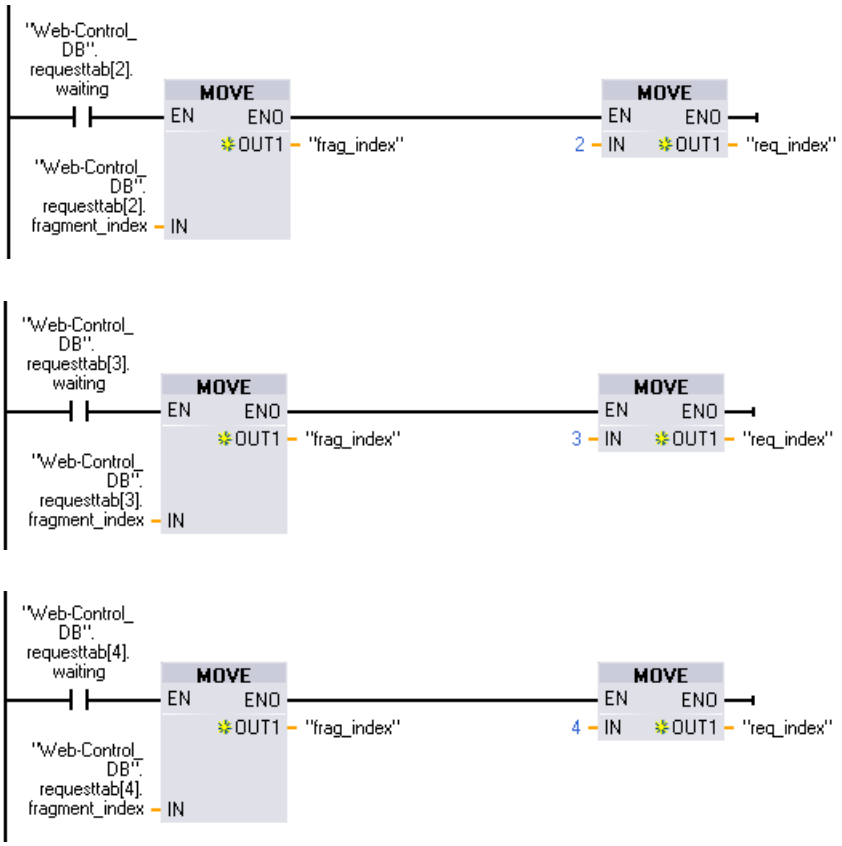
Programa de usuario para fragmentos manuales

En el programa de usuario de la CPU debe llamarse cíclicamente la instrucción "WWW" al utilizar páginas manuales.

Para poder reaccionar ante los valores indicados en el navegador, es necesario evaluar el Request que proporciona la página manual al servidor web en el programa de usuario. Para ello, el DB de Web Control (p. ej. DB 333) debe analizarse cíclicamente por si tiene algún Request pendiente. En el apartado "requesttab" del DB de Web Control se encuentra la matriz que gestiona cuatro Requests. Cada elemento de la matriz contiene información sobre el correspondiente Request en una estructura.

Un sencillo ejemplo de programa muestra cómo se comprueba la existencia de Requests pendientes con las variables del DB de Web Control.



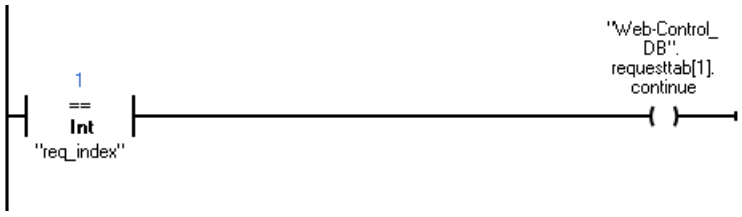


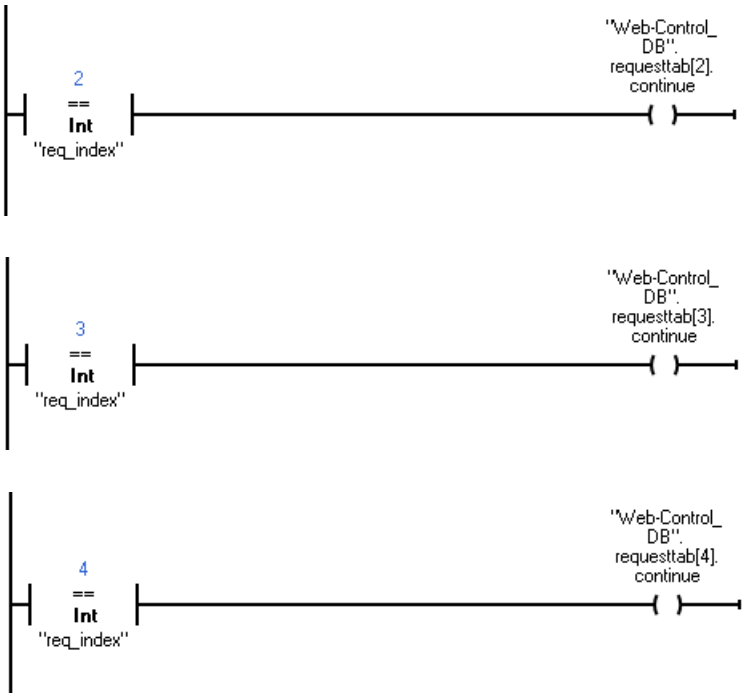
En caso de que exista un Request, esta sección de programa escribe la ID de fragmento en la variable #frag_index y el n.º de Request (rango de valores 1-4) en la variable #req_index.

Con la información obtenida de este modo, en el programa puede procesarse por separado la información transmitida en el Request para cada ID de fragmento (p. ej. comprobación de plausibilidad).

Cuando el programa haya finalizado el procesamiento del Request, debe responderse al Request y volver a restablecerse la correspondiente entrada en "requesttab" del DB de Web Control (p. ej. DB 333).

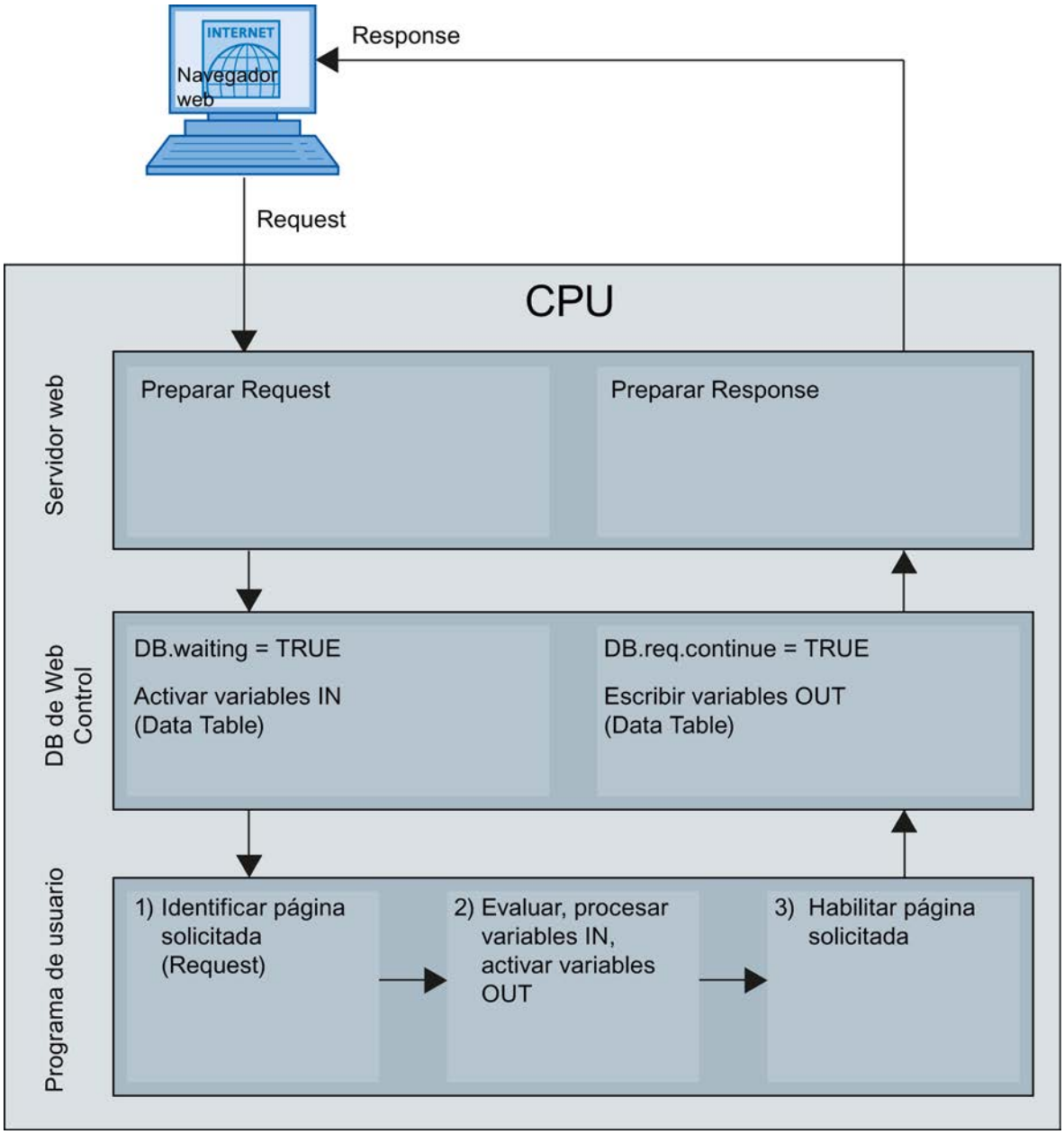
Un sencillo ejemplo de programa para responder a Requests:




2
==
Int
"req_index"
"Web-Control_
DB".
requesttab[2].
continue
3
==
Int
"req_index"
"Web-Control_
DB".
requesttab[3].
continue
4
==
Int
"req_index"
"Web-Control_
DB".
requesttab[4].
continue

Secuencia básica de un Request del navegador con interacción del programa de usuario

La figura siguiente muestra la secuencia básica y simplificada de un Request del navegador web sobre los efectos en los contenidos del DB de Web Control y las acciones necesarias del programa de usuario hasta la devolución de la página web preparada (Response).



Visualización de páginas web definidas por el usuario en el navegador

Visualizar páginas web en el navegador

Las páginas web se abren desde las páginas web estándar del navegador web.

Las páginas web estándar disponen, además de los otros enlaces de la barra de navegación, de un enlace a las "Páginas de usuario".

Al hacer clic sobre el enlace "Páginas de usuario", el navegador web abre la página web que se haya configurado como página de inicio HTML.

Crear páginas web definidas por el usuario en varios idiomas

Existe la posibilidad de poner a disposición las páginas web definidas por el usuario en distintos idiomas.

Requisitos

Las páginas HTML dependientes del idioma deben colocarse dentro de una estructura de carpetas que contenga carpetas con las abreviaturas de los correspondientes idiomas:

Abreviaturas definidas para los idiomas

Las abreviaturas "de", "en", "fr", "es", "it" y "zh" están definidas. No se soportan carpetas para otros idiomas o carpetas con otros nombres.

Si es necesario, pueden crearse otras carpetas en la misma jerarquía de carpetas para otros archivos; por ejemplo, una carpeta "img" para imágenes y una carpeta "script" para archivos Javascript.

Cambio de idioma para páginas web definidas por el usuario

Requisitos

Las páginas HTML se encuentran en las carpetas de idioma predefinidas, p. ej., páginas HTML con texto en alemán en la carpeta "de", páginas HTML con texto en inglés en la carpeta "en", etc.

Concepto del cambio de idioma

El cambio de idioma se basa en una cookie predefinida denominada "siemens_automation_language". Si se activa el valor "de" en la cookie, el servidor web se conectará a la página web de la carpeta "de" en la próxima solicitud o actualización de página web.

De forma análoga, el servidor web se conecta a la página web de la carpeta "en" si la cookie adopta el valor "en".

Ejemplo de un cambio de idioma

El ejemplo está estructurado de la manera siguiente:

- En las dos carpetas de idioma "de" y "en" se encuentran los archivos HTML dependientes del idioma con el mismo nombre, p. ej. "langswitch.html". Los textos que se muestran dentro de ambos archivos están en alemán o inglés en función del nombre de la carpeta.
- Además, en la estructura de carpetas hay una carpeta "script" en la que se encuentra el archivo Javascript "lang.js". A este archivo se han transferido las funciones necesarias para el cambio de idioma.

Estructura del archivo "langswitch.html" (carpeta "de")

En el encabezado del archivo están definidos los metadatos "Content-Language", charset y ruta para el archivo Javascript.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Language" content="de">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Cambio de idioma página alemana</title>
<script type="text/javascript" src="script/lang.js" ></script>
</head>
```

En el cuerpo del archivo se realiza la selección de idioma con la ayuda del elemento HTML "select". El elemento "select" inicia una lista de selección y contiene las opciones "de", con el nombre de "Alemán", y "en", con el nombre de "Inglés"; "de" está preseleccionado.

Con la ayuda del event handler "onchange", se llama la función "DoLocalLanguageChange(this)". El parámetro "this" transfiere el objeto "select" con la opción seleccionada a esta función. Ante cualquier cambio de la opción, "onchange" lleva a cabo la llamada de la función.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" selected >Alemán</option>
                <option value="en" >Inglés</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Estructura del archivo "langswitch.html" (carpeta "en")

El encabezado del archivo HTML con texto en inglés tiene una estructura análoga a la del archivo HTML con texto en alemán.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
<html>
<head>
<meta http-equiv="Content-Language" content="en">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>Language switching english page</title>
<script type="text/javascript" src="script/lang.js" ></script>
```

En el cuerpo del archivo también se realiza la selección de idioma con la ayuda del elemento HTML "select". A diferencia del archivo HTML alemán, la opción inglesa está preseleccionada y el texto o las etiquetas están en inglés.

```
<!-- Language Selection -->
<table>
    <tr>
        <td align="right" valign="top" nowrap>
         <!-- change language immediately on change of the selection
-->
          <select name="Language"
onchange="DoLocalLanguageChange(this)" size="1">
                <option value="de" >German</option>
                <option value="en" selected >English</option>
            </select>
        </td>
     </tr>
</table>
<!-- Language Selection End-->
```

Estructura del archivo "lang.js" (en la carpeta "script")

En el archivo Javascript está definida la función "DoLocalLanguageChange" que, a su vez, llama la función "SetLangCookie" con el valor de selección de idioma. SetLangCookie agrupa el nombre y el valor de la cookie y, finalmente, activa la cookie mediante la correspondiente propiedad document.cookie. Para que el servidor web reaccione ante la activación de la cookie con la visualización del idioma deseado, debe cargarse de nuevo la página web (top.window.location.reload).

```
function DoLocalLanguageChange(oSelect) {
        SetLangCookie(oSelect.value);
        top.window.location.reload();
    }

 function SetLangCookie(value) {
        var strval = "siemens_automation_language=";
            // this is the cookie by which the webserver
            // detects the desired language
            // this name is required by the webserver
        strval = strval + value;
        strval = strval + "; path=/ ;";
            // set path to the application, since otherwise
            // path would be set to the requesting page
            // would not get the cookie.
            // The path for user defined applications follows this
sample:
```

```
                // path=/awp/<application name>/<pagename>
                // example: path=/awp/myapp/myappstartpage.htm
                //(where myapp is the name of the web application
                // entered in the web server properties of the cpu)
                /*
                use expiration if this cookie should live longer
                than the current browser session
                var now      = new Date();
                var endttime = new Date(now.getTime() + expiration);
                strval = strval + "; expires=" + endttime.toGMTString()
+ ";";
                */
            document.cookie = strval;
        }
```

Configuraciones adicionales

Configuración de otras funciones

El sistema de automatización S7-1200 dispone de numerosas funciones adicionales que pueden utilizarse como funciones de CPU integradas o que están disponibles mediante módulos insertables (p. ej. módulos de comunicación). Encontrará la descripción en los enlaces que aparecen más abajo.

Consulte también

Resumen de la comunicación punto a punto (Página 1658)

Generalidades de los contadores rápidos (Página 1654)

Configurar PID_Compact V1 (Página 9981)

Configuración de PID_3Step V1 (Página 10016)

Funcionalidad Motion de la CPU S7-1200 (Página 10431)

8.1.4.5 Configuraciones para acoplamientos punto a punto (S7-1500)

Configuraciones para acoplamientos punto a punto

Introducción

Sinopsis de los módulos de comunicaciones

Los sistemas de automatización están compuestos por los más diversos componentes; entre ellos, los módulos de comunicaciones. La comunicación serie mediante conexiones punto a punto representa una opción sencilla de intercambio de datos.

Ajustando los parámetros de comunicación en el nivel inferior del modelo de niveles ISO (consulte el capítulo Seguridad de la transmisión (Página 1015)) es posible adaptarse a los más diversos interlocutores.

La comunicación mediante acoplamiento punto a punto en S7-1500, ET 200MP y ET 200SP se realiza exclusivamente a través de módulos de comunicaciones (CM) con interfaces serie.

SIMATIC S7 ofrece una serie de módulos que, para esta finalidad, ponen a disposición la interfaz física y los mecanismos de protocolo fundamentales.

- RS232: interfaz que puede coordinar la comunicación entre los interlocutores mediante señales cualificadoras adicionales.
- RS422/RS485: interfaz que permite líneas más largas utilizando tensiones diferenciales como tecnología de transmisión y que también posibilita estructuras con más de 2 dispositivos a través de hardware de bus (RS485).

Para transmitir los datos de la CPU a los módulos correspondientes, se dispone de instrucciones que asumen la coordinación entre la CPU y el CM (Communication module). Comunican al programa de usuario si la transmisión se ha realizado con éxito o si se han recibido datos nuevos (en sistemas sin una CPU SIMATIC, el usuario debe programar él mismo la función de estas instrucciones).

En este manual de funciones se describe el funcionamiento y el uso de los módulos de comunicaciones PtP.

Sinopsis de los componentes y las referencias

Tabla sinóptica de los módulos de comunicaciones y su compatibilidad

Módulo de comunicaciones	S7-1500	ET 200MP	ET 200SP	Referencia
CM PtP RS232 BA [1]	X	X	-	6ES7540-1AD00-0AA0
CM PtP RS422/485 BA	X	X	-	6ES7540-1AB00-0AA0
CM PtP RS232 HF [2]	X	X	-	6ES7541-1AD00-0AB0
CM PtP RS422/485 HF	X	X	-	6ES7541-1AB00-0AB0
CM PtP (ET 200SP)	-	-	X	6ES7137-6AA00-0BA0

[1] BA = Basic

[2] HF = High Feature

Sinopsis de los componentes y las interfaces

Tabla sinóptica de los módulos de comunicaciones y sus funciones.

Módulo de comunicaciones	Interfaz	Protocolos					Sistema de conexión	
		Freeport	3964(R)	Modbus Master	Modbus Slave	USS-Master	Sub-D 9 polos	Sub-D 15 polos
CM PtP RS232 BA	RS232	X	X	-	-	X	X	-
CM PtP RS422/485 BA	RS422	X	X	-	-	X	-	X
	RS485	X	-	-	-	X	-	X

Módulo de comunicaciones	Interfaz	Protocolos					Sistema de conexión	
CM PtP RS232 HF	RS232	X	X	X	X	X	X	-
CM PtP RS422/485 HF	RS422	X	X	X	X	X	-	X
	RS485	X	-	X	X	X	-	X
CM PtP (ET 200SP)	RS232	X	X	X	X	X	ET 200SP BaseUnit[1]	
	RS422 [2]	X	X	X	X	X		
	RS485	X	-	X	X	X		

1) BaseUnit con bornes en lugar de Sub-D; asignación según la física de transmisión

2) El módulo de comunicaciones CM PtP también se puede utilizar para acoplamiento multipunto en el modo RS422.

Sinopsis de los componentes y las velocidades de transferencia de datos

Los módulos de comunicaciones pueden enviar y recibir datos con distintas velocidades de transferencia de datos. La siguiente tabla muestra la asignación a los módulos de comunicaciones individuales.

Módulo de comunicaciones	Velocidad de transferencia de datos en bits/s										
	300	600	1200	2400	4800	9600	19200	38400	57600	76800	115200
CM PtP RS232 BA	X	X	X	X	X	X	X	-	-	-	-
CM PtP RS422/485 BA	X	X	X	X	X	X	X	-	-	-	-
CM PtP RS232 HF	X	X	X	X	X	X	X	X	X	X	X
CM PtP RS422/485 HF	X	X	X	X	X	X	X	X	X	X	X
CM PtP (ET 200SP)	X	X	X	X	X	X	X	X	X	X	X

Sinopsis de los componentes y capacidades del búfer de recepción

Cada módulo de comunicaciones dispone de un búfer para almacenar provisionalmente las tramas recibidas. La siguiente tabla muestra la asignación del tamaño máximo de una trama individual, así como el tamaño de la memoria de los módulos de comunicaciones individuales.

Módulo	Capacidad del búfer de recepción kB	Longitud máxima de la trama kB	Tramas en búfer
CM PtP RS232 BA	2	1	255
CM PtP RS422/485 BA	2	1	255
CM PtP RS232 HF	8	4	255
CM PtP RS422/485 HF	8	4	255
CM PtP (ET 200SP)	4	2	255

Señales cualificadoras y control del flujo de datos

- Control del flujo de datos de software con XON/XOFF
  El control del flujo de datos con XON/XOFF es posible en el protocolo Freeport mediante las interfaces RS232 y RS422.
- Control del flujo de datos de hardware con RTS/CTS
  El control del flujo de datos con RTS/CTS es posible en el protocolo Freeport mediante la interfaz RS232.
- Manejo automático de las señales cualificadoras
  Con los protocolos Freeport, maestro Modbus y esclavo Modbus es posible manejar las señales cualificadoras RS232 a través de la interfaz RS232. (Solo está disponible si no está activado el control del flujo de datos de hardware.)

Protocolos de los módulos de comunicaciones

En función de los módulos de comunicaciones empleados es posible establecer una conexión de comunicación con diferentes protocolos:

- Freeport: transferencia de cadenas de caracteres ASCII sin un formato de protocolo específico
- 3964(R): comunicación entre autómatas programables (comunicación maestro-maestro)
- USS: comunicación entre un autómata programable y un accionamiento (comunicación maestro-esclavo). La comunicación está adaptada a las necesidades de la tecnología de accionamientos. El módulo de comunicaciones solo puede ser maestro.
- Modbus RTU: comunicación entre autómatas programables (comunicación maestro-esclavo). El módulo de comunicaciones puede ser tanto maestro como esclavo.

Sinopsis de las operaciones

Acoplamiento punto a punto

Existen varias posibilidades de interconexión en red para intercambiar datos entre dos o más interlocutores. El acoplamiento punto a punto entre dos interlocutores es el método más sencillo de intercambio de información.

El módulo de comunicaciones (CM) constituye la interfaz entre un autómata programable y un interlocutor. En el acoplamiento punto a punto, la transferencia de datos se realiza en serie con el módulo de comunicaciones.

Configuración/parametrización

La configuración del módulo de comunicaciones incluye la disposición del módulo de comunicaciones en la configuración de dispositivos de STEP 7 V12, así como los ajustes de los parámetros específicos de los protocolos en el diálogo de propiedades del módulo de comunicaciones (configuración estática).

Programación

La programación incluye la conexión informática del módulo de comunicaciones a la CPU correspondiente a través del programa de usuario. La programación del módulo de comunicaciones se realiza con STEP 7 V12.

La comunicación entre la CPU, el módulo de comunicaciones y un interlocutor tiene lugar a través de instrucciones. Para el sistema de automatización S7-1500 se dispone de una serie de instrucciones que permiten iniciar y controlar la comunicación en el programa de usuario, así como modificar la configuración durante el tiempo de ejecución (configuración dinámica).

Para más información al respecto, consulte Resumen de las instrucciones (Página 1013) y la Ayuda en pantalla de STEP 7 V12.

Resumen de las instrucciones

Nota

Capacidad de la CPU

Las instrucciones punto a punto se comunican con los módulos de comunicaciones mediante lectura o escritura de registros.
Por ello, al utilizar las instrucciones debe tenerse en cuenta la capacidad de lectura y escritura de registros de la CPU correspondiente.
Si varias instrucciones de una CPU deben leer o escribir registros de forma simultánea, es posible que el programa de usuario tenga que llamar las instrucciones con un desfase temporal.

Resumen de instrucciones

En el módulo de comunicaciones se produce la conversión de los protocolos de transmisión. A través del protocolo se adapta la interfaz del módulo de comunicaciones a la interfaz del interlocutor.

La comunicación entre la CPU, el módulo de comunicaciones y un interlocutor se desarrolla a través de instrucciones especiales y los protocolos soportados por los módulos de comunicaciones correspondientes.

Las instrucciones conforman la interfaz de software entre la CPU y el módulo de comunicaciones. Deben llamarse cíclicamente desde el programa de usuario. Una transmisión de datos tiene lugar a lo largo de varios ciclos.

Las instrucciones forman parte de STEP 7 V12. Encontrará las instrucciones en la Task Card "Instrucciones", en Comunicación > Procesador de comunicaciones.

Las descripciones de las instrucciones correspondientes se encuentran en el sistema de información. Son aplicables a todos los módulos de comunicaciones indicados, siempre que estos soporten la función deseada.

Tabla 8-59 Instrucciones para PtP

Instrucción	Significado
Port_Config (Página 5399)	La instrucción Port_Config permite configurar parámetros básicos de la interfaz de forma dinámica.
Send_Config (Página 5401)	La instrucción Send_Config (parametrización de la transmisión) permite configurar dinámicamente parámetros de transmisión serie de un puerto.
Receive_Config (Página 5403)	La instrucción Receive_Config (parametrización de la recepción) permite parametrizar dinámicamente parámetros de recepción serie de un puerto.
P3964_Config (Página 5408)	La instrucción P3964_Config (configuración del protocolo) permite configurar los parámetros del procedimiento 3964(R) de forma dinámica.
Send_P2P (Página 5410)	La instrucción Send_P2P permite enviar datos a un interlocutor.
Receive_P2P (Página 5413)	La instrucción Recveive_P2P permite recibir datos de un interlocutor.
Receive_Reset (Página 5415)	La instrucción Receive_Reset permite borrar el búfer de recepción del módulo de comunicaciones.
Signal_Get (Página 5415)	La instrucción Signal_Get permite leer las señales cualificadoras RS232.
Signal_Set (Página 5416)	La instrucción Signal_Set permite activar las señales cualificadoras RS232.
Get_Features (Página 5418)	La instrucción Get_Features permite leer funciones avanzadas soportadas por el módulo de comunicaciones.
Set_Features (Página 5418)	La instrucción Set_Features permite activar funciones avanzadas soportadas por el módulo de comunicaciones.

Tabla 8-60 Instrucciones para el USS

Instrucción	Significado
USS_Port_Scan (Página 5438)	La instrucción USS_Port_Scan permite la comunicación a través de la red USS.
USS_Drive_Control (Página 5440)	La instrucción USS_Drive_Control permite intercambiar datos con el accionamiento.
USS_Read_Param (Página 5443)	La instrucción USS_Read_Param permite leer parámetros del accionamiento.
USS_Write_Param (Página 5444)	La instrucción USS_Write_Param permite modificar los parámetros del accionamiento.

Tabla 8-61 Instrucciones para el Modbus

Instrucción	Significado
Modbus_Comm_Load (Página 5465)	La instrucción Modbus_Comm_Load permite configurar el puerto del módulo de comunicaciones para Modbus RTU.
Modbus_Master (Página 5468)	La instrucción Modbus_Master permite comunicarse a través del puerto PtP en calidad de maestro Modbus.
Modbus_Slave (Página 5474)	La instrucción Modbus_Slave permite comunicarse a través del puerto PtP en calidad de esclavo Modbus.

Fundamentos de la comunicación serie

Transferencia de datos serie

En la transferencia de datos serie, los diferentes bits de un carácter correspondiente a la información que desea transferirse se transmiten uno detrás de otro en un orden fijo.

Tráfico de datos bidireccional - Modo de operación

En el módulo de comunicaciones, se distingue entre dos modos de operación para el intercambio de datos bidireccional:

- Modo semidúplex
  Los datos se transfieren alternativamente en ambas direcciones entre uno o más interlocutores. Modo semidúplex significa que en un momento determinado un interlocutor envía y el otro interlocutor recibe. Para ello se utiliza una línea que varía de dirección para la transmisión o la recepción.
- Modo dúplex
  Los datos se intercambian simultáneamente entre uno o más interlocutores, por lo que es posible enviar y recibir al mismo tiempo. Para ello se necesita una línea para la transmisión y una línea para la recepción.

Transferencia de datos asíncrona

En los módulos de comunicaciones se realiza una transmisión de datos serie asíncrona. El proceso llamado sincronización con el modelo temporal (modelo temporal fijo para la transmisión de una secuencia de caracteres determinada) solo se cumple durante la transmisión de un carácter. Cada uno de los caracteres que debe transferirse va precedido de un impulso de sincronización, también llamado bit de arranque. La longitud de la transmisión del bit de inicio determina la frecuencia. Uno o dos bits de parada establecen el fin de la transferencia de caracteres.

Convenciones

Además del bit de arranque y del bit de parada, para una transferencia de datos serie se requieren otras convenciones entre los interlocutores. Entre ellas se incluyen:

- velocidad de transferencia de datos
- criterios de inicio de trama y de fin (p. ej. tiempo de retardo de caracteres)
- paridad
- número de bits de datos (7 u 8 bits/carácter)
- número de bits de parada (1 o 2)

Seguridad de la transmisión

La seguridad de la transmisión es muy importante en el proceso de transferencia de datos y para la selección del método de transmisión. Por norma general puede decirse que cuantas

más capas del modelo de referencia se ejecuten, mayor será la seguridad de la transmisión.

Clasificación de los protocolos existentes

La clasificación de los protocolos existentes del módulo de comunicaciones en el modelo de referencia puede consultarse en la siguiente figura.

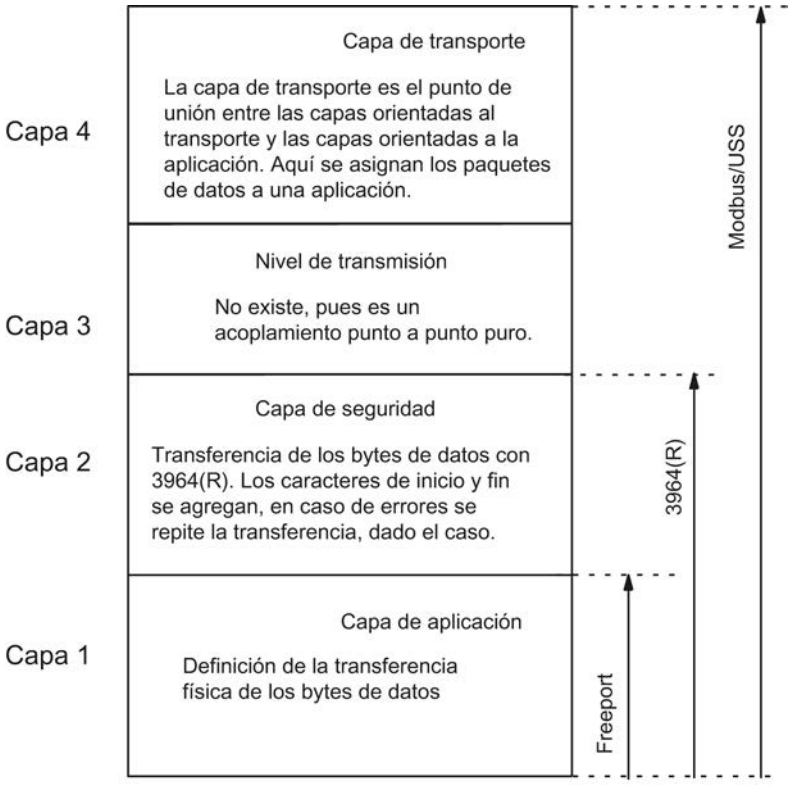


Figura 8-10 Catalogación de los protocolos existentes del módulo de comunicaciones en el modelo de referencia

Seguridad de la transmisión con Freeport

Seguridad de la transmisión en caso de utilizar Freeport:

- Dado que, en el transporte de datos con Freeport, aparte del bit de paridad no se aplica ninguna otra medida para la protección de los datos, la transferencia de datos con Freeport resulta muy eficaz en lo que respecta al flujo de datos, pero la seguridad del transporte de los datos es nula. Parametrizando las condiciones de inicio y fin de trama es posible conseguir cierta seguridad de los datos.
- Mediante el uso del bit de paridad se garantiza la inversión de un bit en un carácter que debe transmitirse. Si se invierten dos o más bits de un carácter, ya no se podrá garantizar que el fallo se detecte.

- Si fuera necesario aumentar la seguridad de la transmisión, puede hacerse introduciendo una suma de verificación, indicando la longitud de una trama o mediante condiciones de fin parametrizables. Estas medidas debe realizarlas el usuario.
- Otra forma de aumentar la seguridad de los datos consiste en introducir tramas de acuse en las tramas de transmisión y recepción. Esto ya ocurre en los protocolos de comunicación de datos de alto nivel (véase el modelo de referencia OSI de 7 capas).

Seguridad de la transmisión con 3964(R)

El bit de paridad sirve para aumentar la seguridad de los datos; complementa el número de bits de datos transferidos a un número par o impar según sea la parametrización.

Mediante el uso del bit de paridad se garantiza la inversión de un bit en un carácter que debe transmitirse. Si se invierten dos o más bits de un carácter, los fallos ya no podrán detectarse con seguridad.

Paridad "ninguna" indica que no se transmite ningún bit de paridad. Esto reduce la seguridad de la transmisión.

Dependiendo de si se desea transmitir los datos con o sin carácter de control de bloque, se distingue entre:

- Transferencia de datos sin carácter de control de bloque: 3964
  La seguridad de la transmisión se consigue mediante establecimiento y desconexión fijos de las tramas, así como con repeticiones de las mismas.
- Transferencia de datos con carácter de control de bloque: 3964R
  La elevada seguridad de transmisión se consigue mediante establecimiento y desconexión fijos de las tramas, repeticiones de los mismos así como la inclusión del carácter de control de bloque (BCC).

En las descripciones e indicaciones que se refieren a ambos métodos de transferencia de datos se utiliza en este manual la denominación 3964(R).

Seguridad de la transmisión con Modbus y USS

El bit de paridad sirve para aumentar la seguridad de la transmisión; complementa el número de bits de datos transferidos a un número par o impar según sea la parametrización.

Mediante el uso del bit de paridad se garantiza la inversión de un bit en un carácter que debe transmitirse. Si se invierten dos o más bits de un carácter, el fallo ya no podrá detectarse con seguridad.

Paridad "ninguna" indica que no se transmite ningún bit de paridad. Esto reduce la seguridad de la transmisión.

En el caso de Modbus se utiliza además el control cíclico de redundancia (CRC: cyclic redundancy check). Antes de la transferencia de datos se añade redundancia adicional, en forma de un valor denominado valor CRC, para cada bloque de datos de usuario. Se trata de un valor de ensayo calculado según un procedimiento determinado, con el cual se pueden detectar posibles errores producidos durante la transmisión.

Con USS se utiliza además un carácter de control de bloque BCC (block check character). Durante la recepción se forma el carácter de control de bloque y, tras la lectura de toda la trama, se compara con el BCC recibido. Si no coinciden, no se evalúa la trama. (Si un carácter

se transmite incorrectamente, se detectará un fallo con seguridad. Si se ha transmitido erróneamente un número par a un carácter, ya no se podrá detectar un fallo con seguridad.)

Modo RS232

El modo RS232 es compatible con los siguientes módulos de comunicaciones:

- CM PtP RS232 BA
- CM PtP RS232 HF
- CM PtP (ET 200SP)

En el modo RS232, la transmisión de datos se realiza a través de dos líneas. Una de las líneas está disponible para la transmisión y la otra para la recepción. Es posible enviar y recibir al mismo tiempo (dúplex).

Señales RS232

Cuando se emplea la física RS232, en el módulo de comunicaciones existen las siguientes señales RS232 además de las señales TXD (interfaz envía; Transmitted Data), RXD (interfaz recibe; Received Data) y GND (Ground; aislada):

DCD	(entrada)	Data Carrier detect; Señal portadora en caso de conexión de un módem. El interlocutor señaliza que ha detectado datos entrantes.
DTR	(salida)	Data terminal ready; DTR en "ON": módulo de comunicaciones encendido y listo para entrar en funcionamiento DTR en "OFF": módulo de comunicaciones apagado y no listo para entrar en funcionamiento
DSR	(entrada)	Data set ready; DSR en "ON": el interlocutor señaliza que está listo para entrar en funcionamiento DSR en "OFF": interlocutor apagado y no listo para entrar en funcionamiento
RTS	(Salida)	Request to send; RTS en "ON": módulo de comunicaciones preparado para enviar; señaliza al interlocutor que hay datos listos para ser enviados. RTS en "OFF": módulo de comunicaciones no preparado para enviar
CTS	(entrada)	Clear to send; El interlocutor puede recibir datos del módulo de comunicaciones (respuesta a RTS = ON del módulo de comunicaciones) CTS en "ON": señaliza al interlocutor la disponibilidad de recepción CTS en "OFF": señaliza al interlocutor la no disponibilidad de recepción
RI	(entrada)	Ring Indicator; Llamada entrante en caso de conexión de un módem

Tras conectar el módulo de comunicaciones las señales de salida están en estado OFF (inactivas).

La operación de las señales de control DTR/DSR y RTS/CTS puede parametrizarse en la interfaz del módulo de comunicaciones.

No es posible influir en las señales RS232 cuando:

- se ha parametrizado el control del flujo de datos "RTS de HW con protoc. intercambio" (equivale al manejo automático de las señales cualificadoras)
- se ha parametrizado el control del flujo de datos "RTS de HW siempre ON" (equivale al control del flujo de hardware con RTS/CTS)
- se ha parametrizado el control del flujo de datos "RTS de HW siempre ON, ignorar DTR/DSR"

Consulte también el capítulo Procedimiento Handshake (Página 1021).

Modo RS422

El modo RS422 es soportado por los siguientes módulos de comunicaciones:

- CM PtP RS422/485 BA
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

En el modo de operación RS422, la transmisión de datos se realiza a través de dos parejas de hilos (modo a cuatro hilos). Una de las parejas está disponible para la transmisión y la otra para la recepción. Es posible enviar y recibir al mismo tiempo (dúplex).

Cada interlocutor debe poder utilizar simultáneamente una dirección de transmisión y recepción.

Los datos pueden intercambiarse simultáneamente entre uno o varios interlocutores. En el modo multipunto RS422 solo debe enviarse un esclavo en un instante concreto.

Modos de operación de la interfaz

En la tabla siguiente se muestran los modos de operación de la interfaz para módulos de comunicaciones y protocolos en general.

El módulo de comunicaciones puede emplearse en el modo RS422 en las siguientes topologías:

- acoplamiento entre dos dispositivos: acoplamiento punto a punto
- acoplamiento entre varios dispositivos: acoplamiento multipunto (solo disponible con CM PtP (ET 200SP))

Modo de operación	Descripción
Dúplex (RS422) modo a 4 hilos (acoplamiento punto a punto)	En este modo de operación, ambos dispositivos tienen los mismos derechos.
Dúplex (RS422) modo a 4 hilos (maestro multipunto)	El módulo de comunicaciones puede emplearse como maestro multipunto.
Dúplex (RS422) modo a 4 hilos (esclavo multipunto)	El módulo de comunicaciones puede emplearse como esclavo multipunto.

En una topología maestro/esclavo multipunto en modo RS422 se aplica lo siguiente:

- El emisor del maestro se interconecta con los receptores de todos los esclavos.
- Los emisores de los esclavos se interconectan con el receptor del maestro.
- El estado inicial (asignación predeterminada) solo se produce en el receptor del maestro y en el receptor de un esclavo. Todos los demás esclavos trabajan sin asignación previa.

Modo RS485

El modo RS485 es soportado por los siguientes módulos de comunicaciones:

- CM PtP RS422/485 BA
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

En el modo RS485, la transferencia de datos se realiza a través de una pareja de hilos (modo a dos hilos). La pareja de hilos está disponible para la transmisión y la recepción de forma alternativa. Es posible o enviar o recibir (semidúplex). Tras una transmisión se conmuta inmediatamente a la recepción (disponibilidad de recepción). Si llega otra petición de transmisión, se conmuta de nuevo a transmisión.

Modo RS485

En la tabla siguiente se muestran los modos de operación de la interfaz para módulos de comunicaciones y protocolos en general.

Modo de operación	Descripción
Semidúplex (RS485) modo a 2 hilos	Modo de operación para acoplamiento punto a punto o acoplamiento multipunto (Multipoint) en modo a dos hilos. El módulo de comunicaciones puede ser tanto maestro como esclavo.

Si se utiliza Freeport en modo RS485 (semidúplex, modo a dos hilos), hay que procurar en el programa de usuario que nunca envíe más de un dispositivo a la vez. Si se realizan transmisiones simultáneas, las tramas se falsifican.

En el caso de Modbus, está garantizado automáticamente que nunca se envíe más de un dispositivo a la vez.

Tiempos de conmutación para el módulo de comunicaciones RS485 en modo semidúplex

El tiempo de conmutación máximo entre la transmisión y la recepción es de 0,1 ms.

Procedimiento Handshake

Introducción

Los procedimientos handshake controlan el flujo de datos entre dos interlocutores. Con el uso de procedimientos handshake se impide que se pierdan datos durante la transmisión entre dispositivos que trabajan a diferentes velocidades.

Básicamente se distinguen los procedimientos siguientes:

Tabla 8-62 Sinopsis de los procedimientos y las interfaces

Procedimiento	RS232	RS422	RS485
Control del flujo de datos de software XON/XOFF	X	X	-
Control del flujo de datos de hardware RTS/CTS	X	-	-
Manejo automático de las señales cualificadoras	X	-	-

Control del flujo de datos de software

El control del flujo de datos de software en el módulo de comunicaciones se desarrolla del siguiente modo:

- XON/XOFF
  - En cuanto el módulo de comunicaciones entra por parametrización en el modo de operación "XON/XOFF", envía el carácter XON y permite así que el interlocutor envíe.
  - Cuando se alcanza el número máximo de tramas parametrizado o bien 16 caracteres antes de que el búfer de recepción se desborde, el módulo de comunicaciones envía el carácter XOFF y pide así al interlocutor que interrumpa la transmisión. Si a pesar de ello el interlocutor continúa enviando, se genera un mensaje de error cuando el búfer de recepción se desborda. Los datos recibidos de la última trama se rechazan.
  - En cuanto la CPU recoge una trama y el búfer de recepción vuelve a estar listo para la recepción, el módulo de comunicaciones envía el carácter XON.
  - Si el módulo de comunicaciones recibe el carácter XOFF durante una transmisión, interrumpe la transmisión hasta que recibe de nuevo un XON del interlocutor. Si transcurrido un tiempo determinado parametrizable no se recibe ningún carácter XON, el proceso de transmisión se cancela y se genera el correspondiente mensaje de error.

Nota

Los caracteres para XON y XOFF pueden parametrizarse (cualquier carácter ASCII).

Al parametrizar el control del flujo de datos de software XON/XOFF, los datos de usuario no pueden contener ninguno de los caracteres XON o XOFF parametrizados.

Control del flujo de datos de hardware

Nota

Con la parametrización "RTS de HW siempre ON, ignorar DTR/DSR", las señales DTR/DSR no tienen que estar cableadas.

Con la parametrización "RTS de HW siempre ON" es necesario cablear por completo las señales de interfaz utilizadas. Asegúrese de que la RTS local (out) esté conectada con la CTS (in) del interlocutor y la CTS local con la RTS del interlocutor. En correspondencia, la DTR local debe estar conectada con la DSR del interlocutor y la DSR local con la DTR del interlocutor.

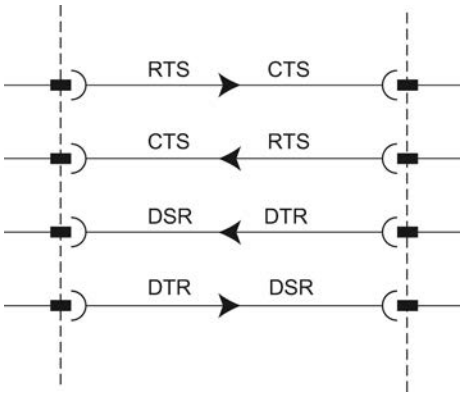


Figura 8-11 Cableado de las señales de interfaz

- RTS de HW siempre ON, ignorar DTR/DSR
  - En cuanto el módulo de comunicaciones entra por parametrización en un modo de operación con "RTS de HW siempre ON", indica su disponibilidad al interlocutor mediante RTS = ON.
  - Cuando se alcanza el número máximo de tramas parametrizado o bien 16 caracteres antes de que el búfer de recepción se desborde, RTS se pone a OFF.
    Si a pesar de ello el interlocutor continúa enviando, se genera un mensaje de error cuando el búfer de recepción se desborda. Los datos recibidos de la última trama se rechazan.
  - En cuanto la CPU recoge una trama y el búfer de recepción vuelve a estar preparado para la recepción, RTS se pone de nuevo a ON.
  - Si se produce un cambio de CTS a OFF durante la transmisión, el módulo de comunicaciones interrumpe el proceso de transmisión hasta que CTS vuelve a ponerse a ON. Si transcurrido un tiempo parametrizable CTS no vuelve a ponerse a ON, el proceso de transmisión se cancela y se genera el correspondiente mensaje de error.
- RTS de HW siempre ON
  El modo de operación "RTS de HW siempre ON" es equivalente al modo de operación "RTS de HW siempre ON, ignorar DTR/DSR", aunque hay que cablear adicionalmente DTR y DSR.
  - En cuanto el módulo de comunicaciones entra por parametrización en un modo de operación con "RTS de HW siempre ON", indica por lo general su disponibilidad al interlocutor poniendo DTR = ON y RTS = ON.
  - Cuando se alcanza el número máximo de tramas parametrizado o bien 16 caracteres antes de que el búfer de recepción se desborde, RTS se pone a OFF.
    Si a pesar de ello el interlocutor continúa enviando, se genera un mensaje de error cuando el búfer de recepción se desborda. Los datos recibidos de la última trama se rechazan.
  - En cuanto la CPU recoge una trama y el búfer de recepción vuelve a estar preparado para la recepción, RTS se pone de nuevo a ON.
  - Si se produce un cambio de CTS a OFF durante la transmisión, el módulo de comunicaciones interrumpe el proceso de transmisión hasta que CTS vuelve a ponerse a ON. Si transcurrido un tiempo parametrizable CTS no vuelve a ponerse a ON, el proceso de transmisión se cancela y se genera el correspondiente mensaje de error.
  - Con un cambio de DSR = ON a OFF, una petición de transmisión en curso se cancela con un mensaje de error.

Manejo automático de las señales cualificadoras

- HW RTS siempre activado
  La realización de "RTS de HW con protoc. intercambio" en el módulo de comunicaciones se desarrolla del siguiente modo:
  - En cuanto el módulo de comunicaciones entra por parametrización en el modo de operación con "RTS de HW con protoc. intercambio", pone las líneas RTS a OFF y DTR a ON (módulo de comunicaciones operativo).
    Solo es posible enviar tramas tras poner la línea DSR a ON. Mientras DSR permanezca en OFF, no se podrán transferir datos a través de la interfaz del RS232. Cualquier petición de transmisión se cancela con el correspondiente mensaje de error.
  - Si está pendiente una petición de transmisión, RTS pasa a ON y se inicia el retardo RTS ON parametrizado. Una vez transcurrido el tiempo de salida de datos se comprueba si el interlocutor ha puesto CTS a ON. En caso afirmativo, los datos se envían a través de la interfaz RS232.
  - Si la línea CTS no pasa a ON durante el retardo RTS ON, o si durante el proceso de transmisión CTS cambia a OFF, se cancela la petición de transmisión y se genera el correspondiente mensaje de error.
  - Tras la transmisión de los datos y una vez transcurrido el retardo RTS OFF parametrizado, la línea RTS pasa a OFF. No se espera a que la línea CTS pase a OFF.
  - Siempre es posible recibir datos a través de la interfaz RS232. Si existe riesgo de desbordamiento del búfer de recepción del módulo de comunicaciones, no se produce ninguna reacción.
  - Con un cambio de DSR = ON a OFF, una petición de transmisión en curso se cancela con un mensaje de error.

  Nota

  El "retardo RTS ON" debe definirse de tal manera que el interlocutor pueda pasar a modo de disponibilidad de recepción antes de que el tiempo haya transcurrido.

  El "retardo RTS OFF" debe definirse de tal manera que el interlocutor pueda recibir completos los últimos caracteres de la trama antes de que la línea RTS, y con ella la petición de transmisión, se desactiven.

  Nota

  En caso de parametrización de un control automático de las señales RS232 no es posible controlar RTS y DTR con las instrucciones correspondientes.

Cronograma

La siguiente figura muestra el desarrollo temporal de una petición de transmisión con control del flujo de datos "RTS de HW con protoc. intercambio" parametrizado.

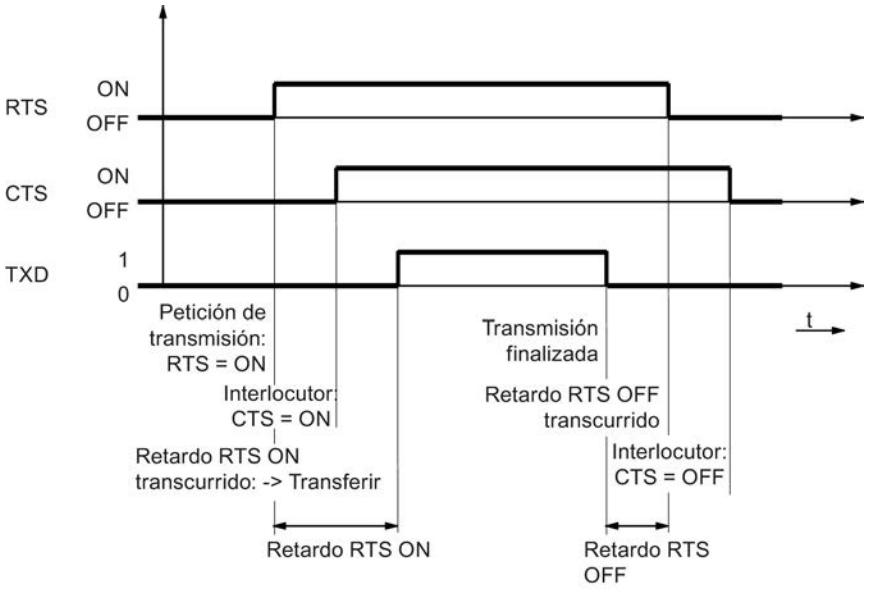


Figura 8-12 cronograma con RTS de HW con protocolo de intercambio

Información adicional

Nota

Con los ajustes siguientes, el módulo de comunicaciones se hace cargo del control de DTR/DSR y RTS/CTS:

- RTS de HW siempre ON, ignorar DTR/DSR
- RTS de HW siempre ON
- RTS de HW con protoc. intercambio

Configuración/parametrización

Configuración/parametrización de un módulo de comunicaciones

En los capítulos siguientes encontrará explicaciones sobre los siguientes protocolos y sus parámetros:

- Comunicación a través de Freeport (Página 1027)
- Comunicación a través de 3964(R) (Página 1033)
- Comunicación a través de USS (Página 1037)
- Comunicación a través de Modbus RTU (Página 1039)

Esta información es necesaria para poder realizar la parametrización y, seguidamente, la programación de la comunicación conforme al protocolo utilizado.

La configuración y parametrización se realiza en la vista de dispositivos de STEP 7 y en el diálogo de propiedades del módulo de comunicaciones. Algunas configuraciones se pueden modificar también durante el tiempo de ejecución con las instrucciones "Config" correspondientes (Port_Config, Send_Config, Receive_Config, P3964_Config).

Procedimiento para configurar una comunicación PtP

El procedimiento es independiente del módulo de comunicaciones empleado.

1. En la vista de dispositivos del editor de hardware de STEP 7 V12, configure una diseño S7-1500 con CPU y módulo de comunicaciones.
2. Parametrice la interfaz del módulo de comunicaciones (protocolo, parámetros de protocolo, direcciones) en el campo "General" de la ficha "Propiedades".

Información adicional

Encontrará información sobre la programación de la comunicación en el capítulo Programación - Comunicación mediante instrucciones (Página 1044).

Comunicación a través de Freeport

Transferencia de datos con Freeport

Introducción

Freeport es un protocolo basado en tramas que puede programarse libremente y que también se denomina protocolo ASCII.

El protocolo Freeport controla la transferencia de datos en un acoplamiento punto a punto entre el módulo de comunicaciones y un interlocutor. El protocolo Freeport contiene la capa física (capa 1).

El protocolo Freeport permite transferir y recibir mensajes con cualquier estructura (todos los caracteres desde 00H hasta FFH (en tramas con 8 bits de datos) o desde 00H hasta 7FH (en tramas con 7 bits de datos)).

Tanto para el sentido de transmisión como para el de recepción deben parametrizarse criterios de inicio y fin de una trama. Los criterios de inicio y fin se pueden parametrizar de forma diferente.

Para la comunicación con un interlocutor se dispone de instrucciones (consulte Programación - Comunicación mediante instrucciones (Página 1044)).

Transferir datos con Freeport

Definición de los ajustes para la transmisión

Para enviar un mensaje hay que comunicar al interlocutor el principio y fin de un mensaje. Estos ajustes pueden realizarse de forma fija en la configuración hardware o adaptarse

durante el tiempo de ejecución con la instrucción Send_CFG. Puede elegir entre las siguientes opciones o combinarlas:

- Enviar Break antes de iniciar la trama
  Existe la posibilidad de especificar que, al principio de cada transmisión de mensajes, se envíe adicionalmente un 'break' una vez transcurrido el tiempo de retardo RTS ON.
  La duración de la señal "Break" se indica en tiempos de bit.
  Si la sincronización se realiza con otros mecanismos, se puede desactivar dicha pausa.
- Enviar Idle Line
  Existe la posibilidad de especificar que, al principio de cada transmisión de mensajes, se emita adicionalmente la señal "Idle Line".
  La duración de la señal "Idle Line" se indica en tiempos de bit.
  Si la sincronización se realiza con otros mecanismos, se puede desactivar dicha pausa.
- Retardo RTS ON
  Es posible parametrizar el tiempo que debe transcurrir tras la solicitud de transmisión RTS (Request to send) antes de que empiece la transferencia de datos propiamente dicha (solo RS232).
- Retardo RTS OFF
  Es posible parametrizar el tiempo que debe transcurrir tras la transferencia completa antes de que se desactive la señal RTS (solo RS232).
- Enviar hasta carácter de fin inclusive
  Existe la posibilidad de parametrizar el número de caracteres de fin (1 o 2) y su valor.
  Se envía hasta el o los caracteres de fin, independientemente de la longitud de trama seleccionada. El delimitador de fin debe encontrarse en los datos que se van a enviar. Los datos se envían únicamente hasta el delimitador de fin incluido, aunque se haya especificado una mayor longitud de datos.
- Número de caracteres añadidos
  Entrada del número de caracteres añadidos. Se envía hasta la longitud parametrizada. El o los delimitadores de fin se añaden automáticamente. Dependiendo del número de delimitadores de fin se le envían al interlocutor de 1 a 5 caracteres más de los indicados en la instrucción.

Nota

Si se combinan "Enviar Break antes de iniciar la trama", "Enviar Idle Line" y "Retardo RTS ON", estos se procesarán en el orden: "Retardo RTS ON", "Enviar Break antes de iniciar la trama" y "Enviar Idle Line".

Recibir datos con Freeport

Definición del principio del mensaje

En la transferencia de datos con Freeport se puede elegir entre varios criterios de inicio distintos. El criterio de inicio establece cuándo empieza una trama. Cuando se cumple un criterio que indica el principio de un mensaje, se empieza a examinar el flujo de datos buscando criterios para el fin del mensaje. Seleccione aquí los ajustes correspondientes a las propiedades del interlocutor emisor.

Para detectar el principio de un mensaje existen dos métodos distintos:

- **Empezar con cualquier carácter**
  Cualquier carácter puede definir el principio de un mensaje (ajuste predeterminado). Esto significa que el primer carácter al principio de una comunicación o tras la detección de un fin de trama se detecta como primer carácter de un mensaje.
- **Iniciar con una condición especial**
  El principio de un mensaje se reconoce por las siguientes condiciones especificadas.
  - **Tras detectar Line Break**
    El principio de la trama solo se acepta si previamente se ha recibido un Break, es decir, el interlocutor tiene que enviar obligatoriamente un Break antes de enviar una trama.
  - **Tras detectar Idle Line**
    El principio de la trama solo se acepta una vez transcurrida la duración parametrizada de la Idle Line. Para ello se requiere una pausa mínima entre 2 tramas.
  - **Tras detectar el carácter de inicio**
    El principio de la trama se detecta cuando se detecta el carácter de inicio parametrizado.
  - **Tras detectar una o varias secuencias de inicio**
    El principio de la trama se detecta cuando se detecta la secuencia de caracteres parametrizada, que está formada por un máximo de 5 caracteres. Es posible parametrizar hasta 4 secuencias de inicio. Las secuencias de inicio formadas por un máximo de 5 caracteres también pueden contener "don't care characters".

Ejemplo:

Tabla 8-63 Condiciones de inicio parametrizadas

Condición de inicio	1.er carácter	2.º carácter	3.er carácter	4.º carácter	5.º carácter
1	0x68	xx	xx	0x68	xx
2	0x10	0xaa	xx	xx	xx
3	0xdc	0xaa	xx	xx	xx
4	0xe5	xx	xx	xx	xx
:					

Se ha recibido el siguiente mensaje: 68 10 aa 68 bb 10 aa 16

La evaluación de los criterios de inicio comienza al recibir el 1.er carácter: 0x68.
Los caracteres 2.º y 3.º pueden ser cualquiera.
Al recibir el 4.º carácter (segundo 0x68) se cumple la primera condición de inicio y comienza la consiguiente evaluación del mensaje.

Definición del fin del mensaje

En la transmisión de datos con el protocolo Freeport se puede elegir entre varios criterios de fin distintos. El criterio de fin establece el momento en el que se completa la recepción de una trama.

Los criterios de fin ajustables son:

- **Detectar fin del mensaje por rebase del tiempo máximo de espera**
- **Detectar fin del mensaje por rebase del tiempo de respuesta**

- Transcurrido el tiempo de retardo de caracteres (ajuste predeterminado)
- Tras recibir una longitud fija de la trama
- Tras recibir un número máximo de caracteres
- Leer longitud del mensaje en el mensaje
- Tras recibir una secuencia de fin

Tiempo de mensaje excedido

En la recepción de datos se detecta el fin de la trama una vez transcurrido el tiempo parametrizado para la transferencia de una trama. El tiempo se empieza a medir cuando se cumple el criterio de inicio.

Rebase del tiempo de respuesta

Con el tiempo de respuesta se vigila el comportamiento de respuesta del interlocutor. Si al concluir un proceso de transmisión no se detecta un inicio de trama válido, el proceso de transmisión se acusa con el mensaje correspondiente.

El criterio de fin real debe parametrizarse adicionalmente.

Transcurrido el tiempo de retardo de caracteres

Al recibir datos se detecta el final de la trama una vez rebasado el tiempo máximo ajustado entre caracteres consecutivos (tiempo de retardo de caracteres). El valor se indica en tiempos de bit.

En este caso, el tiempo de retardo de caracteres deberá ajustarse de tal modo que pueda transcurrir con seguridad entre dos tramas consecutivas. Esta indicación de tiempo debería ser lo suficientemente larga como para que no se confunda el final de la trama con una pausa de transmisión del interlocutor dentro de una trama.

Longitud fija de la trama

Al recibir datos se detecta el final de la trama una vez alcanzada la longitud parametrizada de la trama.

Si el tiempo de retardo de caracteres transcurre (si está activado) antes de que se alcance la longitud fija de la trama, se genera un mensaje de error y la trama se rechaza.

Cuando la longitud de la trama de los caracteres recibidos no coincida con la longitud fijada, tenga en cuenta lo siguiente:

- Todos los caracteres que se reciben tras alcanzar la longitud parametrizada de la trama se rechazan hasta que se detecta un nuevo criterio de inicio.
- Si se cumple otro criterio de fin (activado) antes de que se alcance la longitud de trama parametrizada, se genera un mensaje de error y se rechaza la trama.

Número máximo de caracteres

Al recibir datos se detecta el fin de la trama cuando se recibe el número parametrizado de caracteres.

Este ajuste puede combinarse con los ajustes de "tiempo de retardo de caracteres". La trama también se considera recibida sin errores cuando se cumple otra condición de fin, independientemente de si se ha alcanzado el número máximo de caracteres.

Cuando la longitud de trama de los caracteres recibidos no coincida con la longitud máxima parametrizada, tenga en cuenta lo siguiente:

- Todos los caracteres que se reciben tras alcanzar el número máximo de caracteres parametrizado se rechazan hasta que se detecta un nuevo criterio de inicio (p. ej. "Idle Line").
- Si se cumple otro criterio de fin (activado) antes de que se alcance el número máximo de caracteres parametrizado, esta "parte de la trama" se valora como trama válida y se espera un nuevo criterio de inicio. Todos los caracteres recibidos antes de que se cumpla un nuevo criterio de inicio se rechazan.

Nota

Si no hay ningún otro criterio de fin activado, la longitud fija de la trama y el número máximo de caracteres presentan el mismo comportamiento.

Longitud del mensaje dentro del mensaje

Al recibir datos se detecta el final de la trama cuando se alcanza la longitud de trama enviada en la trama de recepción.

Con los parámetros siguientes se especifican los caracteres que deben considerarse para evaluar la longitud del mensaje:

- **Offset del campo de longitud en el mensaje**
  Este valor determina la posición del carácter en el mensaje que debe considerarse para determinar la longitud del mensaje.
  En función del tamaño del búfer pueden ajustarse valores entre 0 y 4095 caracteres.
- **Tamaño del campo de longitud**
  Este valor indica cuántos caracteres a partir de la primera posición de evaluación deben considerarse para determinar la longitud del mensaje.
  Pueden ajustarse valores de 1, 2 y 4 caracteres.
- **Número de caracteres que se excluyen de la longitud**
  Número de caracteres añadidos a la trama pero que no se incluyen en la longitud de la trama. Este valor indica el número de bytes al final de la trama que no deben tenerse en cuenta al evaluar la longitud de la trama.
  Pueden ajustarse valores entre 0 y 255 caracteres.

Ejemplo:

Parametrizaciones para "Longitud del mensaje dentro del mensaje"

Offset del campo de longitud en el mensaje: 3.er byte (como Offset debe parametrizarse "2")
Tamaño del campo de longitud: 1 byte
Número de caracteres que se excluyen de la longitud: 3 bytes

Mensaje					Número de caracteres que se excluyen de la longitud		
Carácter de inicio	Dirección	Campo de longitud			Suma de verificación	Carácter de fin	
Byte 1	Byte 2	Byte 3	Byte...	Byte X	Byte X+1	Byte X+2	Byte X+3

Secuencia de fin

Al recibir datos se detecta el final de la trama cuando se recibe la secuencia de fin parametrizada (máx. 5 caracteres). La secuencia de fin formada por un máximo de 5 caracteres también puede contener "don't care characters". Los datos recibidos son aplicados por la CPU, incluida la secuencia de fin.

Si se trabaja con secuencia de fin, la transferencia no tendrá transparencia de código y deberá excluirse la posibilidad de que el o los delimitadores de fin estén incluidos en los datos de usuario.

Nota

Secuencia de fin de telegrama

Si solo hay 1 carácter de fin, la entrada **debe** realizarse forzosamente en la 5.ª línea.

Si hay 2 caracteres de fin, las entradas **deben** realizarse en las líneas 4.ª y 5.ª (sin dejar huecos).

Si se utilizan más caracteres, se aplican reglas análogas.

Transparencia del código

Transparencia del código

"Transparencia del código" significa que en los datos de usuario pueden darse todas las combinaciones de caracteres posibles sin que se detecte el criterio de fin.

La transparencia del código del procedimiento depende del criterio de fin parametrizado y del control del flujo de datos:

- Con secuencia de fin determinada o utilizando el control del flujo XON/XOFF
  - No hay transparencia del código
- Criterio de fin tiempo de retardo de caracteres, longitud fija de trama, longitud máxima de trama, tiempo de mensaje excedido o tiempo de respuesta excedido y longitud del mensaje dentro del mensaje
  - Con transparencia del código

Búfer de recepción

Búfer de recepción del módulo

Los módulos de comunicaciones disponen de un búfer de recepción, que almacena provisionalmente las tramas recibidas hasta que se transfieren a la CPU. El búfer de recepción es un búfer en anillo, es decir, las tramas se transmiten a la CPU siguiendo su orden de recepción hasta que el búfer de recepción está lleno. Si se reciben posteriormente más tramas, se sobrescribe la más antigua de las existentes. Si se ha parametrizado "Impedir sobrescritura", cuando el búfer de recepción se llene se generará el mensaje correspondiente. Todas las demás tramas se rechazan hasta que el búfer de recepción vuelva a estar listo para la recepción.

Durante la parametrización se puede indicar si se desea borrar el búfer de recepción al arrancar. Además puede indicarse el rango de valores (de 1 a 255) para el número de tramas de recepción guardadas en el búfer.

Dependiendo del módulo de comunicaciones utilizado, el búfer de recepción del módulo puede alojar hasta 8 kB (consulte el capítulo Introducción (Página 1009)). La longitud máxima de la trama es de 4 kB. De este modo, cada módulo de comunicaciones puede respaldar al menos dos tramas.

Si desea que se transfiera siempre la trama más reciente a la CPU, deberá parametrizar para el número de tramas en búfer el valor "1" y desactivar la protección contra sobrescritura.

Nota

Si la lectura continua de los datos recibidos en el programa de usuario se interrumpe durante un tiempo, al volver a solicitar los datos recibidos puede ocurrir que la CPU reciba primero una trama antigua del módulo de comunicaciones y después la última trama. La trama antigua se tomaba del búfer de recepción del módulo de comunicaciones en el momento de la interrupción y se preparaba para la transferencia a la CPU.

Comunicación a través de 3964(R)

Transferencia de datos con el procedimiento 3964(R)

Introducción

El procedimiento 3964(R) controla la transferencia de datos en un acoplamiento punto a punto entre el módulo de comunicación y un interlocutor. El procedimiento 3964(R) contiene, además de la capa física (capa 1), la capa de enlace de datos (capa 2).

Para la comunicación con un interlocutor se dispone de instrucciones (consulte Programación - Comunicación mediante instrucciones (Página 1044)).

Caracteres de control

Introducción

El procedimiento 3964(R) agrega caracteres de control a los datos de información (capa de enlace de datos) durante la transmisión de datos. Mediante estos caracteres de control, el interlocutor puede controlar si los datos le llegan de forma completa y sin errores.

Caracteres de control del procedimiento 3964(R)

El procedimiento 3964(R) evalúa los siguientes caracteres de control:

STX	Start of Text	Inicio de la secuencia de caracteres que debe transmitirse	02H
DLE	Data Link Escape	Conmutación de transmisión de datos	10H
ETX	End of Text	Fin de la secuencia de caracteres que debe transmitirse	03H
NAK	Negative Acknowledge	Respuesta negativa	15H
BCC	Block Check Character (solo con 3964R)	Carácter de control de bloque	

BCC se forma y vigila automáticamente en el módulo de comunicaciones. El carácter de control de bloque no se transmite a la CPU como contenido de la trama.

Nota

Si dentro de una trama se transmite el carácter DLE como carácter de información, este se enviará doblemente para distinguir el carácter de control DLE durante el establecimiento de la conexión y la desconexión en la línea de transmisión (duplicado DLE). El receptor anula el duplicado DLE.

Prioridad

En el procedimiento 3964(R), un interlocutor debe tener asignada una prioridad mayor que el otro. Si ambos interlocutores empiezan a establecer la conexión simultáneamente, el interlocutor que tiene la prioridad más baja retira su petición de transmisión.

Carácter de control de bloque

Carácter de control de bloque

En el protocolo de transmisión 3964R, la seguridad de datos se aumenta mediante un carácter de control de bloque (BCC = Block Check Character) enviado adicionalmente.

El carácter de control de bloque es la paridad longitudinal par (operación lógica EXOR de todos los bytes de datos) de un bloque enviado o recibido. La formación se inicia con el primer

byte de datos de usuario (1.er byte de la trama) tras establecerse la conexión y finaliza después del carácter DLE ETX en la desconexión.

Nota

En un duplicado DLE, el carácter DLE se incluye dos veces en la formación del BCC.

Transferir datos con 3964(R)

Establecimiento de la conexión al enviar

Para establecer la conexión, el procedimiento 3964(R) envía el carácter de control STX. Si el interlocutor responde con el carácter DLE antes de que transcurra el tiempo de retardo de acuse, el procedimiento pasa al modo de transmisión.

Si, por el contrario, el interlocutor responde con NAK, otro carácter cualquiera (con la excepción de DLE o STX) o si el tiempo de retardo de acuse transcurre sin que se produzca una reacción, el procedimiento repite el establecimiento de la conexión. Tras el número parametrizado de intentos fallidos, el procedimiento cancela el establecimiento y envía el carácter NAK al interlocutor. El módulo de comunicación emite el correspondiente mensaje de error.

Transferir datos

Si el establecimiento de la conexión tiene éxito, los datos de usuario contenidos en el búfer de salida del módulo de comunicaciones se envían al interlocutor con los parámetros de transmisión seleccionados (un DLE detectado en los datos de usuario se duplicará durante el proceso de transmisión). El interlocutor controla la distancia temporal de los caracteres entrantes. La distancia entre dos caracteres no debe ser mayor al tiempo de retardo de caracteres. La vigilancia del tiempo de retardo de caracteres empieza justo después de que se haya conseguido establecer la conexión.

Si el interlocutor envía el carácter NAK durante una transmisión, el procedimiento cancela el bloque y lo repite del modo arriba descrito, comenzando con el establecimiento de la conexión. Si el interlocutor envía cualquier otro carácter, el procedimiento espera primero a que transcurra el tiempo de retardo de caracteres y a continuación envía el carácter NAK para que el interlocutor pase a estado de reposo. Entonces el procedimiento comienza de nuevo la transmisión con el establecimiento de la conexión STX.

Desconexión al enviar

Una vez que se ha enviado el contenido del búfer, el procedimiento agrega los caracteres DLE, ETX y, solo en el caso de 3964R, el carácter de control de bloque BCC como delimitador de fin, y espera un carácter de acuse. Si el interlocutor envía el carácter DLE dentro del tiempo de retardo de acuse, significa que el bloque de datos se ha recibido sin errores. Si, por el contrario, el interlocutor responde con NAK, otro carácter cualquiera (con la excepción de DLE), un carácter erróneo, o si el tiempo de retardo de acuse transcurre sin que se produzca una reacción, el procedimiento comienza de nuevo la transmisión con el establecimiento de la conexión STX.

Tras un número parametrizado de intentos de transferencia, el procedimiento cancela el proceso y envía el carácter NAK al interlocutor. El módulo de comunicación emite el correspondiente mensaje de error.

Recibir datos con 3964(R)

Establecimiento de la conexión al recibir

En estado de reposo, cuando no debe procesarse ninguna petición de transmisión, el procedimiento espera a que el interlocutor establezca la conexión.

Si durante el establecimiento de la conexión con STX no hay ningún búfer de recepción disponible, se inicia un tiempo de espera (tiempo de espera = tiempo de retardo de acuse - 10 ms, aunque máximo 400 ms). Si transcurrido dicho tiempo no hay ningún búfer de recepción libre, se genera un mensaje de error. El procedimiento envía el carácter NAK y regresa al estado de reposo. En caso contrario, el procedimiento envía el carácter DLE y recibe los datos como se ha descrito.

El tiempo de retardo de acuse debería ajustarse al mismo valor en ambos interlocutores.

Si el procedimiento recibe cualquier carácter (con la excepción de STX o NAK) estando en reposo, espera primero a que transcurra el tiempo de retardo de caracteres y a continuación envía el carácter NAK. El módulo de comunicación emite el correspondiente mensaje de error.

Recibir datos

Una vez que la conexión se ha establecido correctamente, los caracteres de recepción entrantes se depositan en el búfer de recepción. Si se reciben dos caracteres DLE consecutivos, solo uno de ellos se registra en el búfer de recepción.

Una vez establecida correctamente la conexión y después de cada carácter recibido, se espera al siguiente durante el tiempo de retardo de caracteres. Si este transcurre sin recepción, se envía el carácter NAK al interlocutor. El módulo de comunicación emite el correspondiente mensaje de error. A continuación se espera una repetición.

Si durante la recepción se producen errores de transmisión (error de trama, error de paridad, etc.), se continúa recibiendo hasta que se produce la desconexión y, a continuación, se envía el carácter NAK al interlocutor. A continuación se espera una repetición. Si el bloque no puede recibirse sin errores tras el número de intentos especificado, o si el interlocutor no inicia la repetición dentro de un tiempo de espera de bloque de 4 s, el procedimiento cancela la recepción. El módulo de comunicación notifica la primera transmisión errónea y la cancelación definitiva.

Desconexión al recibir

Cuando el procedimiento 3964 reconoce la secuencia de caracteres DLE ETX, finaliza la recepción y envía el carácter DLE al interlocutor si ha recibido un bloque sin errores. Si, por el contrario, se produce un error de recepción, envía el carácter NAK al interlocutor. A continuación se espera una repetición.

Cuando el procedimiento 3964R reconoce la secuencia de caracteres DLE ETX BCC, finaliza la recepción. Compara el carácter de control de bloque recibido (BCC) con la paridad longitudinal calculada internamente. Si el carácter de control de bloque es correcto y no ha

ocurrido ningún otro error en la recepción, el procedimiento 3964R envía el carácter DLE y vuelve al estado de reposo. El módulo de comunicaciones informa al control de que existen nuevos datos recibidos.

Si hay un error en el BCC o si se produce cualquier otro error de recepción, se envía el carácter NAK al interlocutor. A continuación se espera una repetición.

Comunicación a través de USS

Resumen de la comunicación USS

Clasificación dentro del entorno del sistema

La siguiente descripción de USS se refiere al uso de los módulos de comunicaciones correspondientes.

- CM PtP RS232 BA
- CM PtP RS422/485 BA
- CM PtP RS232 HF
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Introducción

El protocolo USS® (protocolo de interfaz en serie universal) es un protocolo sencillo de transferencia de datos serie, que está adaptado a las necesidades de la tecnología de accionamientos.

El protocolo USS define una técnica de acceso de acuerdo con el principio maestroesclavo para la comunicación a través de un bus serie. Es posible conectar al bus un maestro y un máximo de 16 accionamientos (esclavos). El maestro selecciona los diferentes accionamientos en la trama con la ayuda de un carácter de dirección. Un accionamiento no puede enviar datos nunca sin que el maestro haya iniciado la transferencia previamente. Esto significa que no es posible una transferencia de datos directa entre accionamientos. La comunicación funciona en modo semidúplex. La función del maestro no puede transferirse.

La tecnología de accionamientos exige tiempos de reacción definidos para las tareas de control y regulación que deben realizarse y, por tanto, un tráfico de tramas con un ciclo estricto:

El maestro envía tramas de forma continua (tramas de petición) a los accionamientos y espera una trama de respuesta de cada accionamiento activado.

Un accionamiento debe enviar una trama de respuesta cuando

- haya recibido correctamente una trama y
- dicha trama estaba dirigida a él.

Un accionamiento no tiene permitido transmitir si no se cumplen las condiciones mencionadas o si el accionamiento ha sido activado en Broadcast (difusión).

Para el maestro, la conexión con los accionamientos en cuestión está establecida cuando recibe una trama de respuesta del accionamiento tras un tiempo de procesamiento definido (tiempo de retardo de respuesta).

Estructura de la trama

Cada trama empieza con un carácter de inicio (STX), seguido de la información sobre la longitud (LGE) y el byte de dirección (ADR). A continuación sigue el campo de datos. La trama finaliza con el carácter de comprobación de bloque (BCC). Para la longitud del telegrama se tienen en cuenta los datos de usuario (número n), el byte de dirección (ADR) y el carácter de protección de los datos (BCC).

STX	LGE	ADR	1	2	...	N	BCC

En el caso de datos compuestos por una palabra (16 bits), primero se envía el byte High y después el byte Low. En el caso de datos de palabras dobles, primero se envía la palabra High y, a continuación, la palabra Low. La longitud de una trama se indica en bytes.

Cifrado de datos

Los datos se encriptan del modo siguiente:

- STX: 1 byte, inicio de texto, 02H
- LGE: 1 byte, contiene la longitud de trama como número binario
- ADR: 1 byte, contiene la dirección del esclavo y el tipo de trama en código binario
- Campos de datos: Un byte cada uno; el contenido depende de la petición
- BCC: 1 byte, carácter de comprobación de bloque

Procedimiento de transmisión de datos

El maestro se ocupa de la transferencia de datos cíclica en tramas. El maestro direcciona sucesivamente todos los dispositivos esclavos con una trama de petición. Los dispositivos en cuestión responden con una trama de respuesta. De acuerdo con el procedimiento maestro-esclavo, el esclavo debe enviar la trama de respuesta al maestro después de haber recibido la trama de petición. Solo entonces el maestro puede comunicarse con el siguiente esclavo.

Campo de datos en la trama

El campo de datos está dividido en dos secciones: Parámetros (PKW) y datos de proceso (PZD).

STX	LGE	ADR	Parámetros (PKW)	Datos de proceso (PZD)	BCC

- Área de parámetros (PKW)
  El área PKW gestiona la transferencia de parámetros entre dos interlocutores (p. ej. autómata y accionamiento). Esto incluye, por ejemplo, leer y escribir valores de parámetros así como leer descripciones de parámetros y el texto correspondiente. La interfaz de PKW suele contener peticiones de manejo y visualización, mantenimiento y diagnóstico.
- Área de datos de proceso (PZD)
  El área PZD incluye señales necesarias para la automatización:
  – Palabras de control y consignas del maestro al esclavo
  – Palabras de estado y valores reales del esclavo al maestro

  Los accionamientos esclavos definen el contenido del área de parámetros y del área de datos de proceso.
  Encontrará más información al respecto en la documentación del accionamiento.

Sinopsis de funciones

Secuencia de transmisión

Las instrucciones procesan cíclicamente la transferencia de datos con un máximo de 16 accionamientos esclavos. Cada accionamiento solo tiene una petición activa.

Características de prestaciones:

- Creación de áreas de datos para la comunicación según la configuración de bus
- Ejecución y supervisión de peticiones de PKW
- Supervisión de todo el sistema y tratamiento de errores
- Comunicación con la CPU
- Acceso a las funciones del accionamiento
- Lectura de los parámetros del accionamiento
- Escritura de los parámetros del accionamiento

Comunicación a través de Modbus RTU

Resumen de la comunicación Modbus

Comunicación Modbus RTU

Modbus RTU (Remote Terminal Unit) es un protocolo estándar para la comunicación dentro de la red y utiliza la conexión eléctrica RS232 o RS422/485 para la transferencia de datos serie entre dispositivos Modbus de la red.

Modbus RTU utiliza una red de maestros y esclavos en la que toda la comunicación parte de un solo dispositivo maestro, mientras que los esclavos simplemente reaccionan a la solicitud del maestro. El maestro envía una solicitud a una dirección de esclavo y solo responde al

comando dicha dirección de esclavo (excepción: tramas Broadcast en dirección de esclavo 0 que no son acusadas por los esclavos).

Se utiliza un procedimiento semidúplex asíncrono y con trasparencia de códigos. La transmisión de datos se realiza sin procedimientos Handshake.

Clasificación dentro del entorno del sistema

La siguiente descripción de Modbus se refiere al uso de los módulos de comunicaciones correspondientes.

- CM PtP RS232 HF
- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Función del acoplamiento

Con los módulos de comunicaciones correspondientes y las instrucciones respectivas, es posible establecer una conexión de comunicación entre un sistema de control Modbus remoto y un SIMATIC S7.

Para la transmisión se utiliza el protocolo GOULD - MODBUS en el formato RTU.

En la comunicación entre un módulo de comunicaciones utilizado como esclavo Modbus y un sistema maestro se pueden utilizar los códigos de función 01, 02, 03, 04, 05, 06, 08, 15 y 16 (consulte Códigos de función (Página 1043)).

Si un módulo de comunicaciones SIMATIC S7 se utiliza como maestro Modbus, además están disponibles los códigos de función 11 y 12.

SIMATIC S7 como esclavo Modbus

El maestro tiene la iniciativa en la transmisión, el módulo de comunicaciones funciona como esclavo.

No es posible el intercambio de tramas entre esclavos.

La instrucción Modbus_Slave pone los datos a disposición en un área de datos SIMATIC conforme a la especificación de la figura o los guarda.

SIMATIC S7 como esclavo Modbus

Como maestro, el módulo de comunicaciones inicia la transmisión y, tras transmitir la trama de solicitud, espera la trama de respuesta del esclavo durante el tiempo de vigilancia de respuesta parametrizado. Si el esclavo no responde, el maestro repite la petición conforme a la parametrización antes de devolver un mensaje de error.

Estructura de la trama

El intercambio de datos "maestro-esclavo" y "esclavo-maestro" comienza con la **dirección del esclavo**, seguida del **código de función**. A continuación se transfieren los datos. La estructura

del campo de datos depende del código de función utilizado. Al final de la trama se transmitirá la comprobación CRC (CRC-Check).

ADDRESS	FUNCTION	DATA	CRC-CHECK
Byte/palabra	Bytes	n bytes	2 bytes

ADDRESS — Dirección del esclavo Modbus
- Dirección estándar: de 1 a 247 (bytes)
- Dirección de estación avanzada: de 1 a 65535 (palabra)

FUNCTION — Códigos de función (Página 1043) Modbus

DATA — Datos de la trama: datos de gestión y de usuario dependientes del código de función

CRC-CHECK — Suma de verificación de la trama

Dirección del esclavo

La dirección de esclavo puede encontrarse en el rango de 1 a 247 (byte) o de 1 a 65535 (palabra). Con la dirección se llega a un esclavo concreto del bus.

Mensaje Broadcast

Con la dirección de esclavo cero, el maestro se dirige a todos los esclavos del bus.

Los mensajes Broadcast se permiten únicamente en relación con los códigos de función 05, 06, 15 o 16 de escritura.

En un mensaje Broadcast, el esclavo no envía ninguna trama de respuesta.

Campo de datos DATA

En el campo de datos DATA se transfieren los datos específicos del código de función:

- Bytecount, Coil_Startaddress, Register_Startaddress; Number_of_Coils, Number_of_Registers...
  Consulte al respecto "Códigos de función (Página 1043)".

CRC-Check

El final de la trama forma la suma de verificación CRC-16 de dos bytes de longitud, formada según el polinomio $x^{16} + x^{15} + x^2 + 1$.

En primer lugar se transfiere el byte Low, y después, el byte High.

Fin de la trama

Se detecta el fin de la trama cuando no se realiza ninguna transferencia en el tiempo que tardan en transferirse 3,5 caracteres y medio (3,5 veces el tiempo de retardo de caracteres (véase Modbus Protocol Reference Guide).

Así, este TIME_OUT de fin de trama depende de la velocidad de transferencia de datos y se indica en tiempos de bit (35 tiempos de bit están codificados internamente de forma fija; en la instrucción se pueden parametrizar adicionalmente otros tiempos de bit).

Tras el TIME_OUT de fin de trama se evalúa y comprueba formalmente la trama Modbus recibida por el interlocutor.

Exception Responses

Cuando el esclavo detecta un error (p. ej., una dirección de registro no permitida) en la trama de solicitud del maestro, establece el bit de mayor valor en el código de función de la trama de respuesta.

A continuación se transfiere un byte de Exception Code que describe la causa del error.

El significado de dicho parámetro se describe en el "GOULD MODICON Modbus Protocol" (no forma parte de la presente documentación).

Trama de Exception Code

La estructura de la trama de Exception Code del esclavo es la siguiente:

- p. ej., dirección de esclavo 5, código de función 5, Exception Code 2

Trama de respuesta del esclavo EXCEPTION_CODE_xx:

05H	Dirección del esclavo
85H	Código de función
02H	Exception Code (1...7)
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cuando el driver recibe una trama de Exception Code, la petición en curso finalizará con error.

Los siguientes códigos de error están definidos según la especificación Modbus:

Código de error	Significado según la especificación Modbus	Causa - Resumen técnico *
1	Illegal function	Código de función no permitido
2	Illegal data address	Dirección de datos no permitida en el esclavo
3	Illegal data value	Valor de datos no permitido en el esclavo
4	Failure in associated device	Error interno en el esclavo
5	Acknowledge	La función será ejecutada
6	Busy, Rejected message	El esclavo no está preparado
7	Negative acknowledgement	Función no ejecutable
* Buscar causa detallada en el esclavo.		

Modo RS232

El modo RS232 es compatible con los siguientes módulos de comunicaciones:

- CM PtP RS232 HF
- CM PtP (ET 200SP)

Encontrará más información sobre el modo RS232 en el capítulo Modo RS232 (Página 1018).

Encontrará información sobre el control del flujo de datos de hardware y sobre el manejo automático de las señales cualificadoras en el capítulo Procedimiento Handshake (Página 1021).

Modo RS422/485

El modo RS422/485 es compatible con los siguientes módulos de comunicaciones:

- CM PtP RS422/485 HF
- CM PtP (ET 200SP)

Encontrará más información sobre el modo RS422/485 en los capítulos Modo RS422 (Página 1019) y Modo RS485 (Página 1020).

Códigos de función

Códigos de función utilizados

El código de función define el significado de la trama. Mediante el código de función se define también la estructura de la trama.

El módulo de comunicaciones soporta los siguientes códigos de función:

Código de función	Función según la especificación MODBUS	Rango
01	Read Coil Status	1 a 2000 bits/solicitud
02	Read Input Status	1 a 2000 bits/solicitud
03	Read Holding Registers	1 a 124/125 bits/solicitud (124 con dirección de estación avanzada)
04	Read Input Registers	1 a 124/125 bits/solicitud (124 con dirección de estación avanzada)
05	Force Single Coil	1 bit/solicitud
06	Preset Single Register	1 palabra/solicitud
08 *	Loop Back Test	Leer estado del esclavo o reiniciar contador de eventos en el esclavo
11 *	Fetch Communications Event Counter (solo maestro)	-

Código de función	Función según la especificación MODBUS	Rango
15	Force Multiple Coils	1 a 1968 bits/solicitud
16	Preset Multiple Registers	1 a 123 bits/solicitud

* Información de diagnostico para la comunicación del esclavo

El código de función Modbus 00 envía un mensaje Broadcast (difusión) a todos los esclavos (sin respuesta del esclavo).

Asignación de las direcciones Modbus a las direcciones SIMATIC

La tabla siguiente muestra la asignación de las direcciones Modbus a las direcciones SIMATIC.

Modbus				S7-1500	
FC[1]	Función	Declaración	Área de direcciones	Declaración	Dirección de la CPU
01	Leer bits	Output	1 - 9999	Imagen de proceso de las salidas	A0.0 - A1248.6
02	Leer bits	Input	10001 - 19999	Imagen de proceso de las entradas	E0.0 - 1248.6
03 [2]	Leer palabras	Holding Register	40001 - 49999 o 400001 - 465535	DW0 - DW19998 o DW0 - DW131068	El área de direcciones M depende de la CPU
04	Leer palabras	Input	30001 - 39999	Imagen de proceso de las entradas	EW0 - EW19996
05 [2]	Escribir bits	Output	1 - 9999	Imagen de proceso de las salidas	A0.0 - A1248.7
06	Escribir palabras	Holding Register	40001 - 49999 o 400001 - 465535	DW0 - DW19998 o DW0 - DW131068	El área de direcciones M depende de la CPU
15	Escribir bits	Output	1 - 9999	Imagen de proceso de las salidas	A0.0 - A1248.6
16 [2]	Escribir palabras	Holding Register	40001 - 49999 o 400001 - 465535	DW0 - DW19998 o DW0 - DW131068	El área de direcciones M depende de la CPU

1) FC = Código de función

2) Con el valor de HR_Start_Offset se determina si con los FC 03, 05 y 16 se direccionan áreas de datos o de marcas en la CPU SIMATIC.

Programación - Comunicación mediante instrucciones

Resumen de la programación PtP

Intercambio de datos mediante comunicación Freeport o 3964 (R)

Los datos enviados deben ponerse a disposición en bloques de datos o en el área de marcas en el programa de usuario de la CPU correspondiente. En el módulo de comunicaciones existe

un búfer de recepción para los datos recibidos. Se crea el bloque de datos correspondiente en la CPU.

En el programa de usuario de la CPU, las instrucciones Send_P2P y Receive_P2P se encargan de la transferencia de datos entre la CPU y el módulo de comunicaciones. La instrucción Receive_Reset permite borrar el búfer de recepción.

Configuración dinámica mediante el programa de usuario

Alternativa o complementariamente a la parametrización de la interfaz del módulo de comunicaciones descrita en el capítulo Configuración/parametrización de un módulo de comunicaciones (Página 1026), en determinados campos de aplicación puede ser ventajoso establecer la comunicación dinámicamente, es decir, controlada por programa mediante una aplicación específica.

Todas las parametrizaciones realizadas en el diálogo de propiedades del módulo de comunicaciones se pueden modificar también durante el tiempo de ejecución con las instrucciones "Config" correspondientes (Port_Config, Send_Config, Receive_Config, P3964_Config).

Secuencia de las llamadas de programa para la comunicación PtP

El gráfico siguiente ilustra la función de las instrucciones PtP para la secuencia de comunicación entre el programa de usuario y el interlocutor.

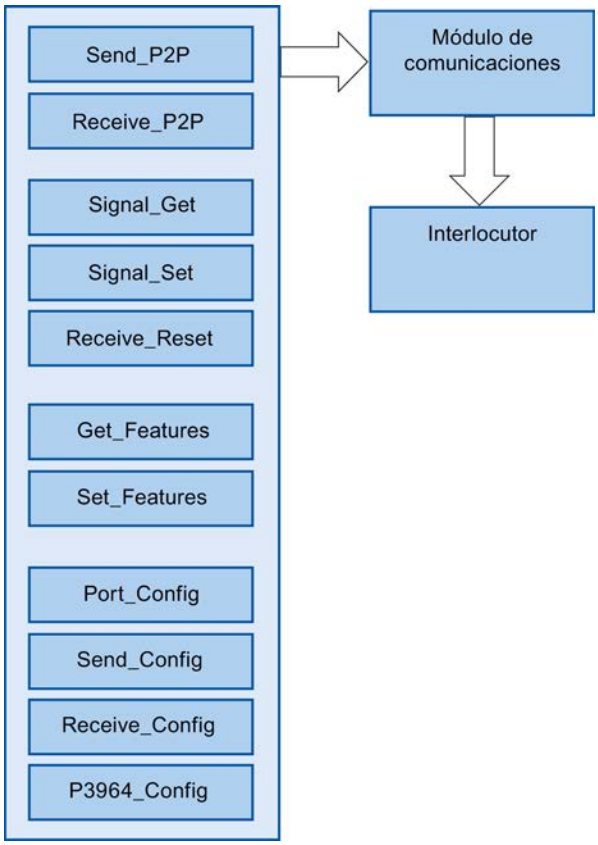


Figura 8-13 Secuencia de programa

Instrucciones PtP

- Instrucciones para la parametrización dinámica
  - Port_Config
    La instrucción Port_Config (configuración de puerto) permite configurar de forma dinámica parámetros básicos de la interfaz, como velocidad de transferencia de datos, paridad y control del flujo de datos, a través del programa de usuario.
  - Send_Config
    La instrucción Send_Config (parametrización de la transmisión) permite configurar de forma dinámica parámetros de transmisión serie, como retardo RTS ON/RTS OFF para una interfaz de comunicación PtP.
  - Receive_Config
    La instrucción Receive_Config (parametrización de la recepción) permite configurar dinámicamente parámetros de recepción serie de un módulo de comunicaciones. Esta instrucción parametriza las condiciones que marcan el principio y fin de un mensaje recibido.
  - P3964_Config
    La instrucción P3964_Config (configuración del protocolo) permite configurar de forma dinámica desde el programa los parámetros de protocolo del procedimiento 3964(R), como tiempo de retardo de caracteres, prioridad y comprobación de bloque.
- Instrucciones para el intercambio de datos (comunicación)
  - Send_P2P
    La instrucción Send_P2P (transferir datos punto a punto) permite transferir datos al interlocutor.
    Para transferir datos con el protocolo Freeport, llame la instrucción Send_P2P. Debe llamar la instrucción de forma cíclica hasta que reciba el acuse correspondiente en los parámetros de salida de la instrucción.
    Nota: Al parametrizar el control del flujo de datos XON/XOFF, los datos de usuario no deben contener ninguno de los caracteres XON o XOFF parametrizados. Los ajustes predeterminados son DC1 = 11H para XON y DC3 = 13H para XOFF.
  - Receive_P2P
    La instrucción Receive_P2P (recibir datos punto a punto) permite recoger los mensajes que ha recibido un interlocutor en el módulo de comunicaciones.
    Para recibir datos con el protocolo Freeport, llame la instrucción Receive_P2P de forma cíclica. La instrucción indica si hay nuevos datos recibidos en el parámetro NDR.
    Para indicar al receptor cuándo empieza la transmisión de un mensaje y cuándo ha finalizado, hay que definir en el protocolo Freeport criterios que identifiquen el principio y el fin de un mensaje.
  - Receive_Reset
    La instrucción Receive_Reset (borrar búfer de recepción) permite borrar el búfer de recepción del módulo de comunicaciones.

- Instrucciones para el control de las señales RS232
  - Signal_Get
    La instrucción Signal_Get (consultar señales RS232) permite leer los estados actuales de las señales RS232.
  - Signal_Set
    La instrucción Signal_Set (consultar señales RS232) permite activar los estados de las señales RS232 DTR y RTS.
- Instrucciones para funciones avanzadas
  - Get_Features
    La instrucción Get_Features (activar funciones avanzadas) permite consultar información sobre la compatibilidad con Modbus y sobre la generación de alarmas de diagnóstico.
  - Set_Features
    La instrucción Set_Features (ajustar funciones avanzadas) permite activar la generación de alarmas de diagnóstico, siempre que el módulo la soporte.

Procedimiento de configuración de la comunicación Freeport o 3964 (R)

Requisitos: la configuración y parametrización de una CPU y de un módulo de comunicaciones en la vista de dispositivos y en el diálogo de propiedades del módulo de comunicaciones han concluido.

1. Seleccione en el árbol del proyecto de la CPU la carpeta "Bloques de programa" y abra el Main (OB1) de la carpeta haciendo doble clic. Se abre el "Editor de programas".
2. Seleccione en la Task Card "Instrucciones", en el campo "Comunicación", las instrucciones Send_P2P y Receive_P2P y desplácelas a una red del Main (OB1) mediante arrastrar y soltar.
3. Parametrice las instrucciones según sus especificaciones.
4. Cargue la configuración hardware y el programa de usuario en la CPU.

Información adicional

Encontrará una lista de las instrucciones disponibles en el capítulo Resumen de las instrucciones (Página 1013).

Encontrará información sobre las instrucciones en el sistema de información de STEP 7 V12.

Resumen de la programación USS

Secuencia de las llamadas de programa para la comunicación USS

El gráfico siguiente ilustra la función de las instrucciones USS para la secuencia de comunicación entre el programa de usuario y el accionamiento USS. (Se requieren las instrucciones Send_P2P, Receive_P2P y las instrucciones "Config" subordinadas).

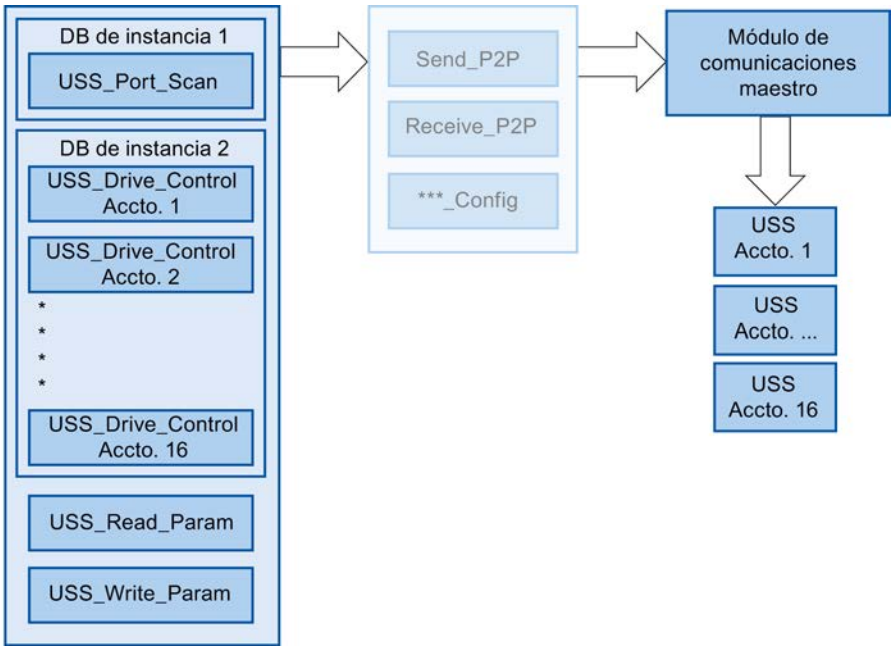


Figura 8-14 Secuencia de programa

Instrucciones USS

- USS_Port_Scan (Página 5438):
  La instrucción USS_Port_Scan permite la comunicación a través de un módulo de comunicaciones con hasta 16 accionamientos, mediante una red USS (debe ser llamada de forma cíclica).
  La instrucción USS_Port_Scan controla la comunicación entre la CPU y los accionamientos a través del puerto de comunicación PtP. Cada vez que se llama esta función se procesa una comunicación con un accionamiento. La instrucción USS_Port_Scan es requerida una vez.
  Puesto que la mayoría de los accionamientos disponen de una función interna parametrizable que vigila la integridad de la comunicación mediante un timeout, la instrucción USS_Port_Scan debe ser llamada desde un OB controlado por tiempo.
- USS_Drive_Control (Página 5440):
  La instrucción USS_Drive_Control permite preparar los datos enviados para un accionamiento y mostrar los datos recibidos.
  Las entradas y salidas de la instrucción corresponden a los estados y las funciones de control del accionamiento. La instrucción USS_Drive_Control debe llamarse una vez por accionamiento. Para todas las llamadas de la instrucción USS_Drive_Control a una red USS, solo es necesario un DB de instancia común. Interconecte todas las llamadas de la instrucción USS_Drive_Control para una red USS con el mismo DB de instancia.
  La instrucción USS_Drive_Control debe llamarse desde el Main (OB1) cíclico del programa principal.
- USS_Read_Param (Página 5443):
  La instrucción USS_Read_Param permite leer parámetros del accionamiento.
  Con la instrucción USS_Read_Param se leen los parámetros operativos del accionamiento que controlan las funciones internas del mismo.
  La instrucción USS_Read_Param debe llamarse desde el Main (OB1) cíclico del programa principal.
- USS_Write_Param (Página 5444):
  La instrucción USS_Write_Param permite modificar los parámetros del accionamiento.
  La instrucción USS_Write_Param debe llamarse desde el Main (OB1) cíclico del programa principal.

Procedimiento para configurar la comunicación USS

Requisitos: la configuración y parametrización de una CPU y de un módulo de comunicaciones en la vista de dispositivos y en el diálogo de propiedades del módulo de comunicaciones han concluido.

1. Seleccione en el árbol del proyecto de la CPU la carpeta "Bloques de programa" y abra el OB controlado por tiempo deseado haciendo doble clic. Se abre el "Editor de programas".
2. Seleccione en la Task Card "Instrucciones", en el campo "Comunicación", la instrucción USS_Port_Scan y desplácela a una red del OB controlado por tiempo mediante arrastrar y soltar.
   La instrucción USS_Port_Scan permite la comunicación a través de la red USS.
3. Seleccione en el árbol del proyecto de la CPU la carpeta "Bloques de programa" y abra el Main (OB1) de la carpeta haciendo doble clic. Se abre el "Editor de programas".

4. Seleccione en la Task Card "Instrucciones", en el campo "Comunicación", las instrucciones para la comunicación USS según la tarea planteada y desplácelas a una red del Main (OB1) mediante arrastrar y soltar:
   - La instrucción USS_Drive_Control sirve para el intercambio de datos con el accionamiento.
   - La instrucción USS_Read_Param sirve para leer parámetros del accionamiento.
   - La instrucción USS_Write_Param sirve para modificar parámetros del accionamiento.
5. Parametrice las instrucciones según sus especificaciones.
6. Cargue la configuración hardware y el programa de usuario en la CPU.

Información adicional

Encontrará una lista de las instrucciones disponibles en el capítulo Resumen de las instrucciones (Página 1013).

Encontrará información sobre las instrucciones en el sistema de información de STEP 7 V12.

Resumen de la programación Modbus

Secuencia de las llamadas de programa para la comunicación Modbus

El gráfico siguiente ilustra la función de las instrucciones Modbus para la secuencia de comunicación entre el programa de usuario y un dispositivo Modbus. (Se requieren las instrucciones Send_P2P, Receive_P2P y las instrucciones "Config" subordinadas).

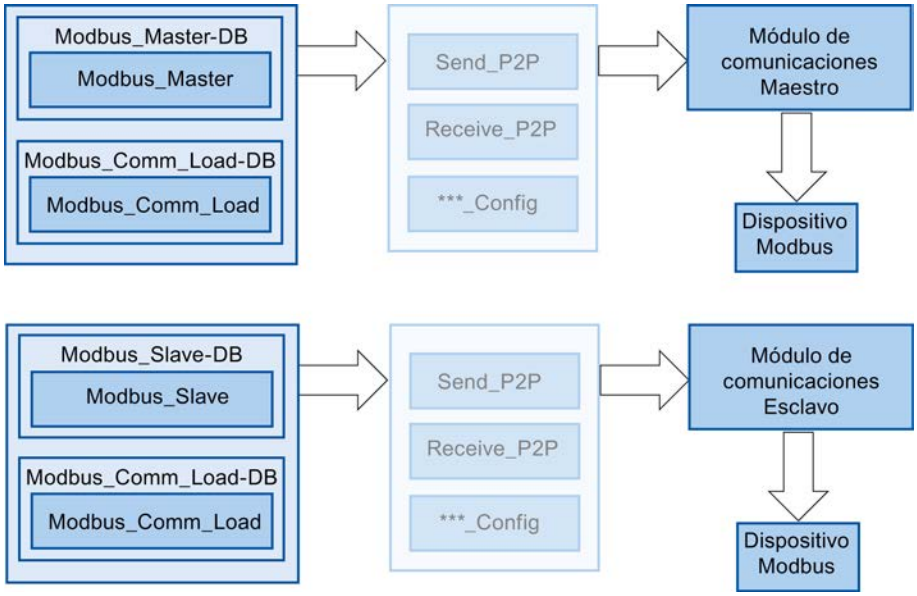


Figura 8-15 Secuencia de programa

Instrucciones Modbus

- Modbus_Comm_Load (Página 5465):
  La instrucción Modbus_Comm_Load permite configurar el puerto del módulo de comunicaciones para Modbus RTU.
  Es necesario ejecutar Modbus_Comm_Load para establecer parámetros de puerto PtP como velocidad de transferencia de datos, paridad y control del flujo. Una vez se ha configurado la interfaz para el protocolo Modbus RTU, solo pueden utilizarla la instrucción Modbus_Master o la instrucción Modbus_Slave .
- Modbus_Master (Página 5468):
  La instrucción Modbus_Master permite comunicarse a través del puerto PtP en calidad de maestro Modbus.
  Con la instrucción Modbus_Master es posible utilizar la CPU como dispositivo maestro Modbus RTU para la comunicación con uno o varios dispositivos esclavos Modbus.
- Modbus_Slave (Página 5474):
  La instrucción Modbus_Slave permite comunicarse a través del puerto PtP en calidad de esclavo Modbus.
  Con la instrucción Modbus_Slave es posible utilizar la CPU como dispositivo maestro Modbus RTU para la comunicación con uno o varios dispositivos esclavos Modbus.

Nota

Uso alternativo de Modbus_Slave y Modbus_Master

Un módulo de comunicaciones puede utilizarse como maestro o como esclavo.

Procedimiento para configurar la comunicación Modbus

Requisitos: la configuración y parametrización de una CPU y de un módulo de comunicaciones en la vista de dispositivos y en el diálogo de propiedades del módulo de comunicaciones han concluido.

1. Seleccione en el árbol del proyecto de la CPU la carpeta "Bloques de programa" y abra el Main (OB1) de la carpeta haciendo doble clic. Se abre el "Editor de programas".
2. Seleccione en la Task Card "Instrucciones", en el campo "Comunicación", las instrucciones para la comunicación Modbus según la tarea planteada y desplácelas a una red del Main (OB1) mediante arrastrar y soltar:
   - La instrucción Modbus_Comm_Load configura el puerto del CM para la comunicación Modbus.
     La instrucción Modbus_Comm_Load debe ser llamada en el Main (OB1) hasta que se muestre DONE (o ERROR).
   - La instrucción Modbus_Master se utiliza para la funcionalidad de maestro Modbus.
   - La instrucción Modbus_Slave se utiliza para la funcionalidad de esclavo Modbus.
3. Parametrice las instrucciones según sus especificaciones.
4. Cargue la configuración hardware y el programa de usuario en la CPU.

Información adicional

Encontrará una lista de las instrucciones disponibles en el capítulo Resumen de las instrucciones (Página 1013).

Encontrará información sobre las instrucciones en el sistema de información de STEP 7 V12.

Arranque y diagnóstico

Comportamiento de arranque

Transiciones de estado operativo

Tras el arranque del módulo de comunicaciones se intercambian todos los datos entre la CPU y el módulo de comunicaciones a través de instrucciones.

STOP de la CPU	Si hay una transferencia de datos en curso entre el módulo de comunicaciones y la CPU, ya sea una petición de transmisión o de recepción, se interrumpe.
RUN de la CPU	En el estado RUN de la CPU está garantizado el intercambio de datos, ya sea envío o recepción. Con la parametrización correspondiente en el cuadro de diálogo de propiedades del módulo se puede borrar automáticamente el búfer de recepción del módulo de comunicaciones durante el arranque de la CPU.

Desde el punto de vista del módulo de comunicaciones, no hay más estados operativos/ transiciones de estado operativo.

Funciones de diagnóstico

Introducción

Las funciones de diagnóstico del módulo de comunicaciones permiten localizar rápidamente los errores ocurridos. Para ello existen varias posibilidades de diagnóstico:

Diagnóstico a través de los elementos de indicación del módulo de comunicaciones	Los elementos de indicación informan sobre el estado operativo o posibles estados de error del módulo de comunicaciones. Los elementos de indicación proporcionan una primera vista general sobre los errores internos y externos ocurridos, así como los errores específicos de interfaz. Encontrará más información al respecto en el manual de producto del módulo de comunicaciones correspondiente.
Diagnóstico a través de la salida STATUS de las instrucciones	Las instrucciones disponen de una salida STATUS destinada al diagnóstico, que proporciona información sobre errores de comunicación entre el módulo de comunicaciones y la CPU. El parámetro STATUS se puede evaluar en el programa de usuario (permanece pendiente durante un ciclo exacto).
Alarma de diagnóstico	El módulo de comunicaciones puede lanzar una alarma de diagnóstico en la CPU que tiene asignada. El módulo de comunicaciones pone a disposición información de diagnóstico. Dicha información se evalúa a través del programa de usuario o leyendo el búfer de diagnóstico de la CPU.

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-

Código de error	Descripción	Solución
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).

Código de error	Descripción	Solución
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).

Código de error	Descripción	Solución
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Recibir datos con Freeport (Página 1028)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)

Código de error	Descripción	Solución
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.

Código de error	Descripción	Solución
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		

Código de error	Descripción	Solución
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

Código de error	Descripción	Solución
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 [1)]	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 [1)]	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 [1)]	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255

Código de error	Descripción	Solución
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.

Código de error	Descripción	Solución
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información

Código de error	Descripción	Solución
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.

Código de error	Descripción	Solución
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1)]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1)]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1)]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1)]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1)]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1)]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG

Código de error	Descripción	Solución
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.

Código de error	Descripción	Solución
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

1) Solo en instrucciones para CPU S7-300/400

8.1.4.6 Configuraciones para acoplamientos punto a punto (S7-300, S7-400)

Configuraciones para acoplamientos punto a punto

Introducción

Sinopsis de los componentes e interfaces

Contenido

Resumen tabular de los distintos tipos de módulos y sus funciones.

Para CP 340/341/440/441 y ET 200S 1SI

Módulo	Interfaz	Protocolo						
		ASCII	3964(R)	RK512	Driver de impresora	Maestro Modbus	Esclavo Modbus	Maestro USS
CP 340	RS232 C	X	X	-	X	-	-	-
CP 340	20 mA TTY	X	X	-	X	-	-	-
CP 340 1)	RS422	X	X	-	X	-	-	-
	RS485	X	-	-	-	-	-	-

Módulo	Interfaz	Protocolo						
		ASCII	3964(R)	RK512	Driver de impresora	Maestro Modbus	Esclavo Modbus	Maestro USS
CP 341	RS232 C	X	X	X	X	X 2)	X 2)	-
CP 341	20 mA TTY	X	X	X	X	X 2)	X 2)	-
CP 341 1)	RS422	X	X	X	X	X 2)	X 2)	-
	RS485	X	-	-	X	X 2)	X 2)	-
CP 440	RS422	X	X	-	-	-	-	-
	RS485	X	-	-	-	-	-	-
CP 441-1	RS232 C	X	X	-	X	-	-	-
CP 441-1	20 mA TTY	X	X	-	X	-	-	-
CP 441-1 1)	RS422	X	X	-	X	-	-	-
	RS485	X	-	-	X	-	-	-
CP 441-2	RS232 C	X	X	X	X	X 2)	X 2)	-
CP 441-2	20 mA TTY	X	X	X	X	X 2)	X 2)	-
CP 441-2 1)	RS422	X	X	X	X	X 2)	X 2)	-
	RS485	X	-	-	X	-	-	-
ET 200S 1SI para ASCII y 3964(R)	RS232 C	X	X	-	-	-	-	-
	RS422	X	X	-	-	-	-	-
	RS485	X	-	-	-	-	-	-
ET 200S 1SI para Modbus y USS	RS232 C	-	-	-	-	X	X	X
	RS422	-	-	-	-	X	X	-
	RS485	-	-	-	-	X	X	X
CPU 313C-2 PtP	RS422	X	X	-	-	-	-	-
	RS485	X	-	-	-	-	-	-
CPU 314C-2 PtP	RS422	X	X	X	-	-	-	-
	RS485	X	-	-	-	-	-	-

1) La diferencia entre RS 422 y RS 485 se define en el cuadro de diálogo de parámetros.

2) Para el protocolo Modbus es necesario un driver cargable con un dongle.

Señales cualificadoras y control del flujo de datos

Con los protocolos ASCII, Maestro Modbus y Esclavo Modbus es posible manejar las señales cualificadoras RS232C a través de la interfaz RS232C.

El control del flujo de datos con RTS/CTS en los protocolos ASCII y driver de impresora puede realizarse a través de la interfaz RS 232C.

El control del flujo de datos con XON/XOFF en los protocolos ASCII y driver de impresora puede realizarse a través de las interfaces RS 232C, 20mA TTY y RS 422.

Resumen de instrucciones

Resumen de instrucciones

La comunicación entre CPU, módulo de comunicación y un interlocutor se desarrolla a través de las instrucciones y los protocolos del módulo de comunicación.

Las instrucciones conforman la interfaz de software entre la CPU y el módulo de comunicación. Deben llamarse cíclicamente desde el programa de usuario.

En el módulo de comunicación se produce la conversión de los protocolos de transmisión. A través del protocolo se adapta la interfaz del módulo de comunicación a la interfaz del interlocutor.

Tabla 8-64 Instrucciones para el CP 340

Instrucción	Significado
P_SEND (Página 7890)	La instrucción P_SEND permite enviar a un interlocutor una área completa o parcial de un bloque de datos.
P_RCV (Página 7893)	La instrucción P_RCV permite recibir datos de un interlocutor y depositarlos en un bloque de datos.
P_PRINT (Página 7896)	La instrucción P_PRINT permite transmitir a una impresora un texto de aviso con hasta 4 variables.
P_RESET (Página 7896)	La instrucción P_RESET permite inicializar el búfer de recepción interno del CP 340.
V24_STAT_340 (Página 7902)	La instrucción V24_STAT permite leer los estados lógicos en la interfaz RS 232C del CP 340-RS 232C.
V24_SET_340 (Página 7903)	La instrucción V24_SET permite activar/desactivar las salidas de la interfaz RS 232C del CP 340-RS 232C.

Tabla 8-65 Instrucciones para el CP 341

Instrucción	Significado
P_SND_RK (Página 7911)	Con la instrucción P_SND_RK se puede tanto enviar a un interlocutor el área completa o parcial de un bloque de datos como recoger datos de dicho interlocutor (utilizando RK512).
P_RCV_RK (Página 7922)	Con la instrucción P_RCV_RK se puede tanto recibir datos de un interlocutor y depositarlos en un bloque de datos como preparar datos para dicho interlocutor (utillizando RK512).
P_PRINT341 (Página 7933)	La instrucción P_PRINT341 permite transmitir a una impresora un texto de aviso con hasta 4 variables.
V24_STAT (Página 7902)	La instrucción V24_STAT permite leer los estados de señal en la interfaz RS 232C del CP 341-RS 232C.
V24_SET (Página 7903)	La instrucción V24_SET permite activar/desactivar las salidas de la interfaz RS 232C del CP 341-RS 232C.

Tabla 8-66 Instrucciones para el CPU 313C-2 PTP

Instrucción	Significado
SEND_PTP (Página 8238)	Con la instrucción SEND_PTP se puede enviar a un interlocutor el área completa o parcial de un bloque de datos.
RCV_PTP (Página 8240)	Con la instrucción RCV_PTP se pueden recibir datos de un interlocutor y depositarlos en un bloque de datos.
RES_RCVB (Página 8240)	Con la instrucción RES_RCVB se puede inicializar el búfer de recepción del módulo.

Tabla 8-67 Instrucciones para el CPU 314C-2 PTP

Instrucción	Significado
SEND_PTP (Página 8238)	Con la instrucción SEND_PTP se puede enviar a un interlocutor el área completa o parcial de un bloque de datos.
RCV_PTP (Página 8240)	Con la instrucción RCV_PTP se pueden recibir datos de un interlocutor y depositarlos en un bloque de datos.
RES_RCVB (Página 8243)	Con la instrucción RES_RCVB se puede inicializar el búfer de recepción del módulo.
SEND_RK (Página 8245)	Con la instrucción SEND_RK se puede enviar a un interlocutor el área completa o parcial de un bloque de datos con el protocolo RK512.
FETCH_RK	Con la instrucción FETCH_RK se pueden solicitar datos de interlocutor con el protocolo RK512
SERVE_RK (Página 8254)	Con la instrucción SERVE_RK y el protocolo RK512 se puede tanto recibir datos de un interlocutor y depositarlos en un bloque de datos como preparar datos para dicho interlocutor.

Tabla 8-68 Instrucciones para el CP 440

Instrucción	Significado
SEND_440 (Página 7952)	Con la instrucción SEND_440 se puede enviar a un interlocutor el área completa o parcial de un bloque de datos.
RECV_440 (Página 7954)	Con la instrucción RECV_440 se pueden recibir datos de un interlocutor y depositarlos en un bloque de datos.
RES_RECV (Página 7957)	Con la instrucción RES_RECV se puede restablecer el búfer de recepción del CP 440.

Tabla 8-69 Instrucciones para el CP 441

Instrucción	Significado
V24_STAT_441 (Página 7967)	La instrucción V24_STAT_441 permite leer los estados lógicos en la interfaz RS 232C del CP 441.
V24_SET_441 (Página 7969)	La instrucción V24_SET_441 permite activar/desactivar las salidas en la interfaz RS 232C del CP 441 .
El sistema de automatización S7-400 ofrece una serie de instrucciones que inician y controlan la comunicación entre la CPU y el procesador de comunicación CP 441 en el programa de usuario. Estas instrucciones se encuentran depositadas de manera fija en la memoria de la CPU (véase el apartado "Instrucciones del sistema de automatización S7-400", situado más abajo).	

Tabla 8-70 Instrucciones para el ET 200S 1SI

Instrucción	Significado
S_SEND (Página 7999)	La instrucción S_SEND permite enviar a un interlocutor una área completa o parcial de un bloque de datos.
S_RCV (Página 7996)	La instrucción S_RCV permite recibir datos de un interlocutor y depositarlos en un bloque de datos.
S_VSTAT (Página 8002)	La instrucción S_VSTAT permite leer los estados de señal en la interfaz RS 232C del módulo ET 200S 1SI.
S_VSET (Página 8004)	La instrucción S_VSET permite establecer/restablecer las salidas de la interfaz RS 232C del módulo ET 200S 1SI.
S_XON (Página 8005)	Con la instrucción S_XON puede definir parámetros adicionales en caso de que se haya parametrizado el módulo para el control de flujo XON/XOFF.
S_RTS (Página 8007)	Con la instrucción S_RTS puede definir parámetros adicionales en caso de que se haya parametrizado el módulo para el control de flujo RTS/CTS.
S_V24 (Página 8009)	Con la instrucción S_V24 puede definir parámetros adicionales en caso de que se haya parametrizado el módulo para el uso automático de las señales V.24.
S_MODB (Página 8023)	La instrucción S_MODB permite establecer una comunicación con un sistema de control maestro Modbus y transformar las direcciones Modbus en áreas de memoria SIMATIC.
S_USST (Página 8033)	La instrucción S_USST gestiona la transmisión de los datos de red (PZD y eventualmente datos PKW) a los esclavos según la estructura de datos de red utilizada.
S_USSR (Página 8036)	La instrucción S_USSR gestiona la recepción de los datos de red (PZD y eventualmente datos PKW) a los esclavos según la estructura de datos de red utilizada.
S_USSI (Página 8038)	Si se activa la instrucción S_USSI durante el arranque del sistema S7, se generarán los DB de procesador de comunicación, de datos de red y de parametrización necesarios para la comunicación.

Tabla 8-71 Instrucciones para el Modbus

Instrucción	Significado
MODB_341 (Página 7985)	Con la instrucción MODB_341 y el driver correspondiente se permite una conexión de comunicación entre un sistema de control de maestro Modbus y el sistema esclavo compatible con Modbus CP 341 del módulo de comunicación.
MODB_441 (Página 7987)	Con la instrucción MODB_441 y el driver correspondiente se permite una conexión de comunicación entre un sistema de control de maestro Modbus y el sistema esclavo compatible con Modbus CP 441-2 del módulo de comunicación.

Instrucciones del sistema de automatización S7-400

En la tabla que sigue a continuación encontrará las instrucciones del sistema de automatización S7-400 disponibles para establecer la comunicación entre la CPU y el CP 441.

Tabla 8-72 Instrucciones del sistema de automatización S7-400

Instrucción	Significado
BSEND	Con la instrucción BSEND se pueden enviar datos de una área de datos S7 a un interlocutor con un destino indicado.
BRCV	Con la instrucción BRCV se pueden recibir datos de un interlocutor y transferirlos a una área de datos S7.
GET	Sólo con RK512: Con la instrucción GET se pueden recoger datos de un interlocutor.
PUT	Sólo con RK512: Con la instrucción PUT se pueden enviar datos a un interlocutor con una introducción dinámica y modificable de la dirección.
PRINT	La instrucción PRINT permite imprimir un texto de aviso con hasta cuatro variables.
STATUS	Con la instrucción STATUS se puede consultar el estado de una conexión de comunicación.

Más información

Encontrará una descripción detalladla de las instrucciones del sistema de automatización S7-400 y sus parámetros en: Resumen de instrucciones para la comunicación del S7 (Página 7666)

Consulte también

P_RESET: Borrar el búfer de recepción (Página 7900)

S_RCV: Recibir datos (Página 8016)

S_SEND: Enviar datos (Página 8020)

FETCH_RK: Recoger datos (RK 512) (Página 8249)

Información básica

Transferencia en serie de un carácter

Introducción

Existen distintas posibilidades de conexión para el intercambio de datos entre dos o más interlocutores. El acoplamiento punto a punto entre dos interlocutores es la forma más sencilla de intercambio de información.

Acoplamiento punto a punto

En el acoplamiento punto a punto, el módulo de comunicación crea una interfaz entre un autómata programable y un interlocutor. En el acoplamiento punto a punto, la transmisión de datos con el módulo de comunicación es serial.

En el módulo de comunicación se produce la conversión a los mecanismos de direccionamiento del procedimiento de transmisión seleccionado. Por ello, la conexión punto a punto ya termina en el módulo de comunicación y no en el interlocutor, como ocurre en otras conexiones.

Transmisión de datos en serie

En la transmisión de datos en serie, los bits individuales de un byte correspondiente a la información que desea transferirse se transmiten uno detrás de otro en un orden fijo.

Intercambio de datos unidireccional/bidireccional

La transmisión de datos con el interlocutor se desarrolla independientemente en una interfaz serie del módulo de comunicación. Para ello, el módulo de comunicación está equipado con tres drivers distintos.

- Intercambio de datos unidireccional:
  - Driver de impresora
- Intercambio de datos bidireccional:
  - Driver ASCII
  - Procedimiento 3964(R)
  - Protocolo para mainframe RK512
  - Modbus/USS

El módulo de comunicación desarrolla la transmisión de datos a través de una interfaz serie en función de la física de la interfaz y del driver seleccionado.

Intercambio de datos unidireccional - Proceso de impresión

En el proceso de impresión (driver de impresora) se envían a una impresora n bytes de datos útiles. Sin embargo, no se produce una recepción de caracteres. Se excluyen los caracteres de control individuales para el control de flujo (p. ej., XON/XOFF).

Intercambio de datos bidireccional - Modos de operación

En el módulo de comunicación, y para el intercambio de datos bidireccional, se distingue entre dos modos de operación:

- Modo semidúplex (procedimiento 3964(R), driver ASCII, RK 512, Modbus/USS)
  Los datos se transfieren entre uno o varios interlocutores variando en ambas direcciones. Un modo semidúplex es un modo en que, dependiendo del momento, se realizarán envíos o recepciones. La excepción en este caso puede venir dada por los caracteres de control individuales para el control del flujo de datos (p. ej., XON/XOFF), que también pueden resultar enviados o recibidos durante el modo de envío/recepción.
- Modo dúplex (driver ASCII)
  Los datos se intercambian simultáneamente entre uno o varios interlocutores; en un determinado punto puede tanto enviarse como recibirse. Cada interlocutor deberá poder utilizar simultáneamente una dirección de envío y recepción.

En el submódulo de interfaz X27 (RS 422/485) únicamente puede trabajarse con un ajuste RS 485 (2 hilos) en el modo semidúplex.

Transmisión de datos asíncrona

En el módulo de comunicación se realiza una transmisión de datos asíncrona en serie. La así llamada sincronización de tiempo (periodo de tiempo fijo durante la transferencia de una secuencia fija de caracteres) sólo se mantiene durante la transferencia de un carácter. Cada uno de los caracteres que vaya a transferirse recibe un impulso de sincronización, también llamado bit de arranque. La duración de la transferencia del bit de arranque establece el ritmo. El final de la transferencia de caracteres forma el bit de parada.

Convenciones

Además del bit de arranque y de parada existe una serie de convenciones entre ambos interlocutores necesarias para la transmisión de datos en serie. Entre ellas se incluyen:

- Velocidad de transmisión (baudrate)
- Tiempo de retardo entre caracteres y de acuse
- Paridad
- Número de bits de datos
- Número de bits de parada

Trama

Los datos que deban transferirse entre el módulo de comunicación y un interlocutor se transmiten a través de la interfaz en serie en una trama de entre 10 y 11 bits. Para cada trama hay varios formatos de datos disponibles. El formato deseado para la transmisión de datos se parametriza en el cuadro de diálogo de propiedades del módulo: Dependiendo del módulo de comunicación correspondiente, no están disponibles todas las tramas (y combinaciones de tramas) teóricamente posibles.

Trama de 10 bits

En la siguiente imagen se representan tres formatos de datos de la trama de 10 bits a modo de ejemplo.

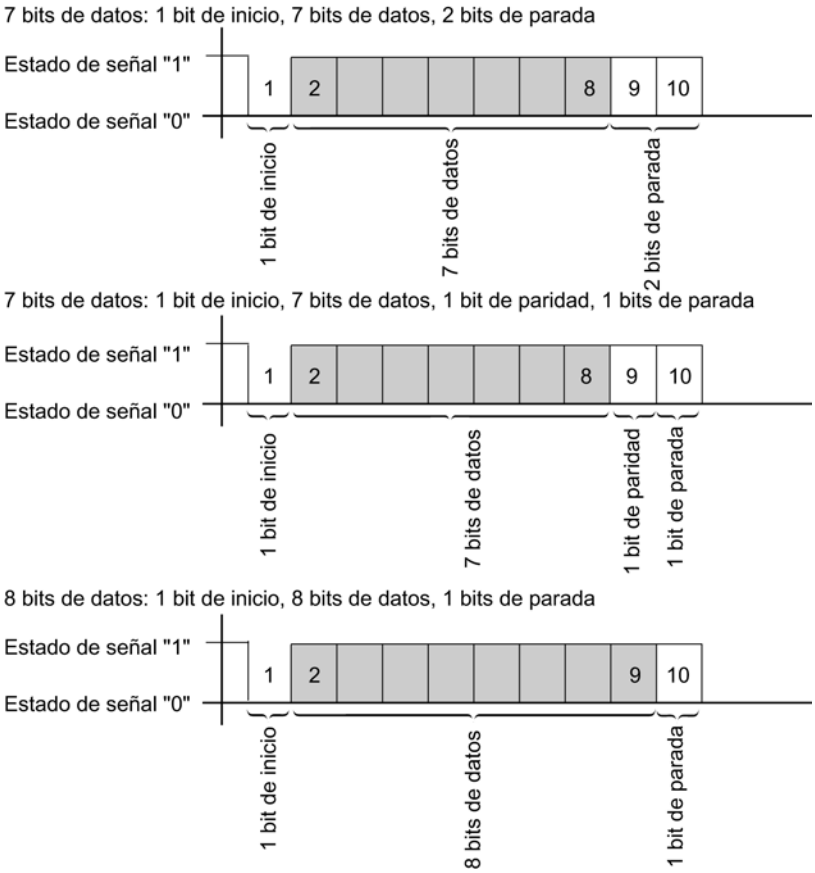


Figura 8-16 Trama de 10 bits

Trama de 11 bits

En la siguiente imagen se representan tres formatos de datos de la trama de 11 bits a modo de ejemplo.

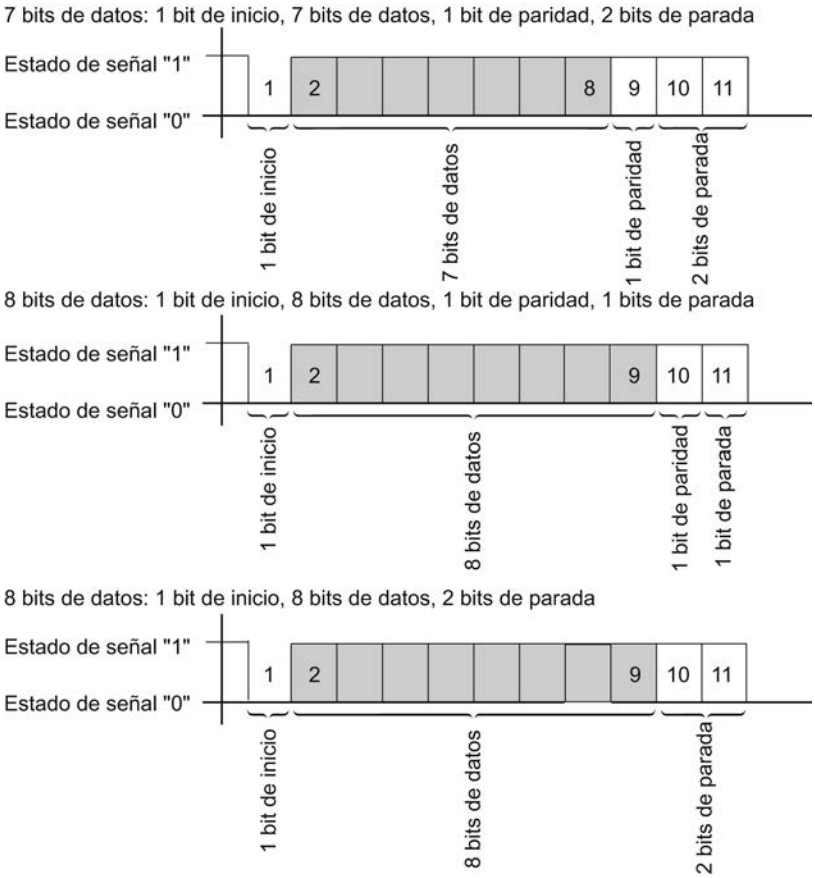


Figura 8-17 Trama de 11 bits

Tiempo de retardo de caracteres

En la siguiente imagen se representa el intervalo máximo de tiempo permitido entre dos caracteres recibidos dentro de un telegrama = tiempo de retardo entre caracteres representado.

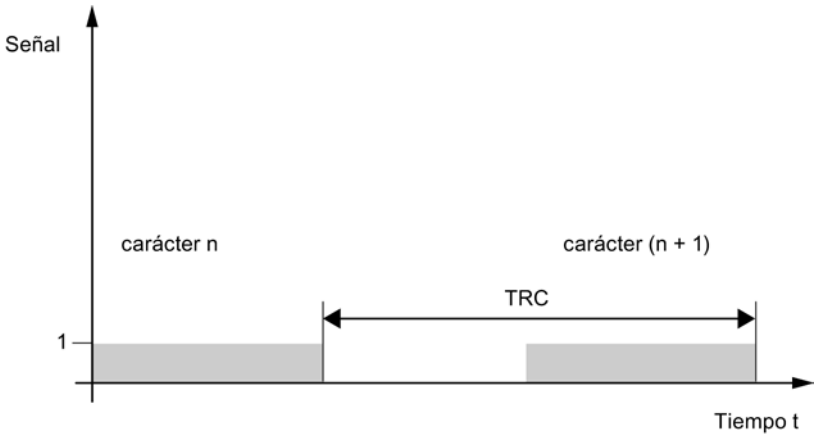


Figura 8-18 Tiempo de retardo de caracteres

Seguridad de transmisión

La seguridad de transmisión es muy importante en el proceso de transferencia de datos y para la selección del procedimiento de transmisión. Por norma general puede decirse que cuantos más capas del modelo de referencia se ejecuten, mayor será la seguridad de transmisión.

Clasificación de los protocolos existentes

La clasificación de los protocolos existentes del módulo de comunicación en el modelo de referencia puede consultarse en la siguiente figura:

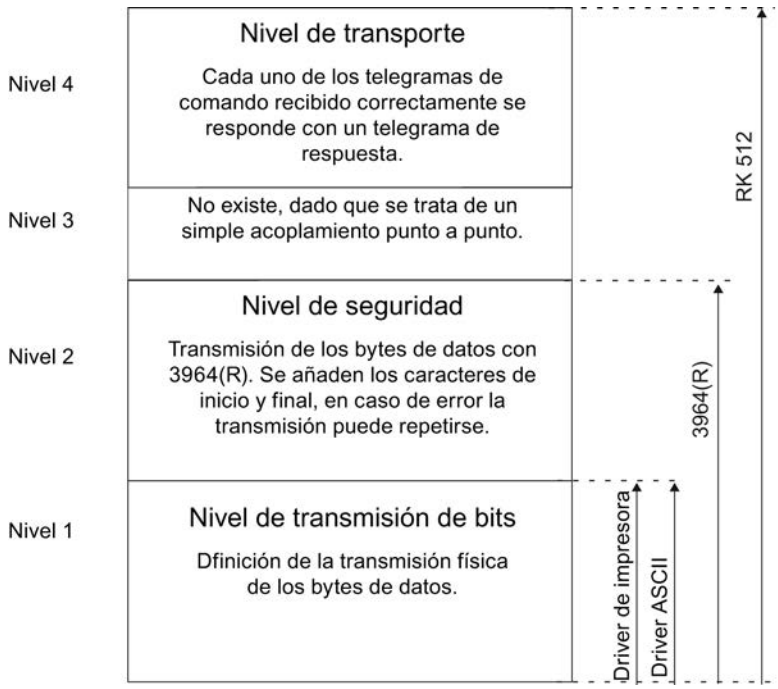


Figura 8-19 Catalogación de los protocolos existentes del módulo de comunicación en el modelo de referencia

Asignación de los protocolos a los módulos de comunicación

Protocolo	CP 340	CP 341	CPU 313C-2 PtP	CPU 314C-2 PtP	CP 440	CP 441	ET 200S 1SI
Driver de impresora	●	●				●	
Driver ASCII	●	●	●	●	●	●	●
Procedimiento 3964(R)	●	●	●	●	●	●	●
Protocolo para mainframe RK512		●		●		●[2]	
Maestro Modbus		●[1]				●[1] [2]	●
Esclavo Modbus		●[1]				●[1] [2]	●
Maestro USS							●

[1] Será necesario un driver cargable con un dongle para el protocolo Modbus.

[2] sólo CP 441-2

Seguridad de transmisión en el driver de impresora

Seguridad de los datos al utilizar el driver de impresora:

- No se ha previsto ninguna medida de seguridad para la transmisión de datos con el driver de impresora.
- Para impedir que se pierdan datos en caso de desbordarse el búfer de recepción de la impresora, utilice el control del flujo de datos (XON/XOFF, RTS/CTS).
- En la salida de la impresora se evalúa la señal BUSY de la impresora. La señal BUSY llega al módulo de comunicación como señal CTS y se evalúa como tal (véase driver ASCII). Hay que tener en cuenta que en el control de flujo con CTS/RTS debe ajustarse la polaridad de la señal BUSY a CTS = "OFF" en la impresora.

Seguridad de transmisión con el driver ASCII

Seguridad de los datos al utilizar el driver ASCII:

- Dado que durante el transporte de datos con el driver ASCII, a parte del bit de paridad, no se aplica ninguna otra medida para la protección de los datos (dependiendo de la trama de caracteres seleccionada también puede desactivarse), la transmisión de datos con el driver ASCII resulta muy eficaz en lo que respecta al flujo de datos, pero la seguridad del transporte de los datos es nula.
- Mediante el uso del bit de paridad se garantiza la inversión de un bit en un carácter a transmitir. Si se invierten dos o más bits de un carácter, el fallo ya no podrá detectarse.
- Si fuera necesario aumentar la seguridad de la transmisión, puede hacerse introduciendo una suma de verificación (checksum) y la indicación de longitud de un telegrama. Estas medidas debe realizarlas el usuario.
- Otra forma de aumentar la protección de los datos consiste en introducir telegramas de acuse de los telegramas de emisión y recepción. Esto ya ocurre en los protocolos de comunicación de datos de alto rendimiento (véase el modelo de referencia ISO de 7 capas).

Seguridad de transmisión con 3964(R)

Mayor seguridad de los datos mediante utilización del procedimiento 3964(R):

- La distancia Hamming con 3964(R) es de 3. La distancia Hamming es una medida para la seguridad de una transmisión de datos.
- Mediante el procedimiento 3964(R) se garantiza una elevada seguridad de transmisión en la línea de transmisión. La elevada seguridad de transmisión se consigue gracias al establecimiento y desconexión prefijadas del telegrama así como a la inclusión del carácter de comprobación de bloque (BCC).

Dependiendo de si se desea transmitir los datos con o sin carácter de comprobación de bloque, se distingue entre

- Transmisión de datos sin carácter de comprobación de bloque: **3964**
- Transmisión de datos con carácter de comprobación de bloque: **3964(R)**

En las descripciones e indicaciones que se refieren a ambos métodos de transmisión de datos se utiliza en este manual la denominación 3964(R).

Seguridad de transmisión con RK512

Protección de datos muy elevada gracias al uso del RK512:

- La distancia Hamming en el caso del RK512 con 3964R es de 4. La distancia Hamming es una medida para la seguridad de una transmisión de datos.
- Utilizando el protocolo para mainframe RK512 se garantiza una elevada seguridad de transmisión en la línea de transmisión (ya que con RK 512 se utiliza el procedimiento 3964R para el transporte de datos).
- Se garantiza el procesamiento posterior de los datos en el interlocutor (porque el intérprete del RK512 evalúa la indicación adicional de la longitud de los datos y, una vez depositados los datos en el área de datos de destino del interlocutor, se genera un telegrama de acuse sobre el éxito/el fracaso del transporte de los datos).
- El protocolo para mainframe RK512 garantiza el uso correcto del procedimiento 3964R y la evaluación/complemento de la indicación de longitud así como la generación automática de telegramas de respuesta. ¡El usuario no tiene que hacer nada! Tan solo tiene que evaluar el acuse positivo/negativo final.

Seguridad de transmisión con Modbus/USS

Seguridad de datos al utilizar el driver Modbus/USS:

- La distancia Hamming en el caso de Modbus/USS es de 4. La distancia Hamming es una medida para la seguridad de una transmisión de datos.
- El uso de Modbus/USS garantiza una seguridad de transmisión muy elevada en la línea de transmisión (con una comprobación de bloque CRC16 con Modbus y una comprobación XOR con USS).
- Se garantiza el procesamiento posterior en el interlocutor, porque tanto Modbus como USS evalúan una información de longitud adicional en el encabezado y porque se genera un telegrama de acuse (doblemente vigilado), que indica si el transporte de datos ha sido correcto o fallido.

Transmisión de datos con driver ASCII

Transmisión de datos con el driver ASCII

Para la asignación de los protocolos a los módulos de comunicación véase: Sinopsis de los componentes e interfaces (Página 1068)

Introducción

El driver ASCII controla la transmisión de datos en un acoplamiento punto a punto entre el procesador de comunicaciones y un interlocutor. El driver ASCII contiene el nivel de transmisión de bits (nivel 1).

La estructura de los telegramas se mantiene así abierta mediante la transmisión de todo el telegrama de envío al procesador de comunicaciones por parte del usuario de S7. Para la recepción debe parametrizarse el criterio de fin de un telegrama. La estructura de los telegramas de envío puede ser diferente de la de los telegramas de recepción.

Con el driver ASCII pueden enviarse y recibirse datos con cualquier estructura (todos los caracteres ASCII imprimibles, así como el resto de caracteres desde 00 hasta FFH (en tramas con 8 bits de datos) y desde 00 hasta 7FH (en tramas con 7 bits de datos)).

Envío de datos con el driver ASCII

Enviar datos

En el envío debe indicarse el número de bytes de datos útiles que deben transferirse en cada llamada de la instrucción de envío con el parámetro correspondiente a la longitud. Encontrará más información al respecto en el apartado "Resumen de instrucciones (Página 1070)".

Si se trabaja con el criterio de fin "Tiempo de retardo de caracteres transcurrido", el driver ASCII realiza una pausa entre dos telegramas también al enviarlos. Se puede acceder en todo momento a la instrucción de envío, aunque el driver ASCII no inicia la salida hasta que, desde el último telegrama enviado, haya transcurrido un tiempo superior al tiempo de retardo parametrizado entre caracteres.

Nota

Si se parametriza el control del flujo XON/XOFF, los datos útiles no pueden contener ninguno de los caracteres XON o XOFF parametrizados. Los ajustes predeterminados son DC1 = 11H para XON y DC3 = 13H para XOFF.

Si trabaja con el criterio de fin "Caracteres de fin", puede elegir entre 3 posibilidades:

- Enviar hasta el carácter de fin incluido
  El carácter de fin debe encontrarse en los datos que se van a enviar. Los datos se envían únicamente hasta el carácter de fin incluido, aunque en la instrucción se haya especificado una mayor longitud de datos.
- Enviar hasta la longitud parametrizada en la instrucción
  Se envían los datos hasta la longitud parametrizada en la instrucción. El último carácter debe ser un carácter de fin.
- Enviar hasta la longitud parametizada en la instrucción y agregar automáticamente el(los) carácter(es) de fin
  Se envían los datos hasta la longitud parametrizada en la instrucción. Adicionalmente se agrega automáticamente el(los) carácter(es) de fin. Esto es, los caracteres de fin no pueden encontrarse en los datos que se van a enviar. Dependiendo del número de caracteres de fin se le envían al interlocutor 1 ó 2 caracteres más de los indicados en la instrucción.

Si se trabaja con el criterio de fin "Longitud fija de telegrama", el número de datos que haya indicado en el parámetro de longitud de la instrucción se transfiere en el sentido de envío. En el sentido de recepción, esto es, en el DB de recepción, se registra el número de datos que se ha especificado en el cuadro de diálogo de propiedades del módulo a través del parámetro "Longitud fija de telegrama". Para garantizar un intercambio de datos correcto, ambos parámetros deberán escogerse de forma idéntica. Al enviar se mantendrá una pausa entre dos telegramas igual al tiempo de retardo de carácter (TRC) para que el interlocutor pueda sincronizarse (detección de inicio del telegrama).

Si la sincronización se realiza con otros mecanismos, puede desactivarse la pausa de envío en el cuadro de diálogo de propiedades del módulo.

Desarrollo durante el envío

En la siguiente figura se representan los procesos de envío.

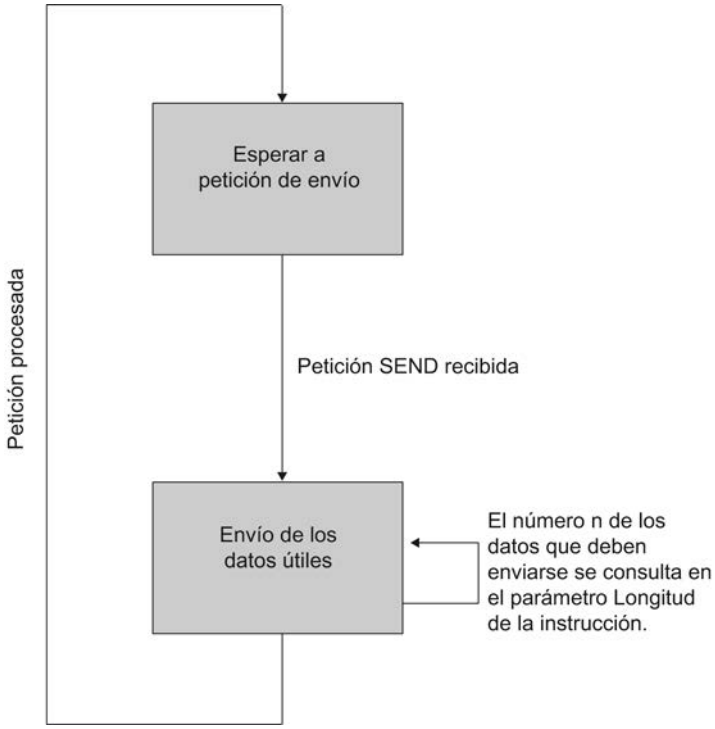


Figura 8-20 Esquema del proceso de envío del driver ASCII

Recepción de datos con el driver ASCII

Criterios de fin seleccionables

En la transmisión de datos con el driver ASCII puede elegir entre tres criterios distintos de fin. El criterio de fin determina cuándo se ha recibido un telegrama completo. Los criterios ajustables de fin son los siguientes:

- Tiempo de retardo de caracteres transcurrido
  El telegrama no tiene una longitud fija ni caracteres de fin definidos; una pausa en la línea determina el fin de dicho telegrama (tiempo de retardo de caracteres trascurrido).
- Recepción del/de los carácter(es) de fin
  Al final del telegrama hay uno o dos caracteres de fin definidos.
- Recepción de un número de caracteres determinado
  La longitud de los telegramas de recepción es siempre igual.

Transparencia del código

La transparencia del código del procedimiento depende de la elección del criterio de fin parametrizable y del control del flujo:

- Con uno o dos caracteres de fin
  - No hay transparencia del código
- Criterio de fin "Tiempo de retardo entre caracteres" o "Longitud fija de telegrama"
  - Con transparencia del código
- Si se utiliza el control del flujo XON/XOFF, la transparencia del código no es posible.

"Transparencia del código" significa que en los datos útiles pueden darse todas las combinaciones de caracteres posibles sin que se detecte el criterio de fin.

Criterio de fin "Tiempo de retardo de caracteres transcurrido"

En la recepción de datos se detecta el final del telegrama una vez que ha transcurrido el tiempo de retardo entre caracteres. Los datos recibidos los acepta la CPU.

El tiempo de retardo entre caracteres debe ajustare en este caso de tal modo que transcurra con seguridad entre dos telegramas consecutivos. Asimismo, ha de ser lo suficientemente largo como para garantizar que en las pausas de envío del interlocutor dentro de un mismo telegrama no se reconozca por error el final del mismo.

En la siguiente figura encontrará las fases de recepción con el criterio de fin "Tiempo de retardo de caracteres transcurrido".

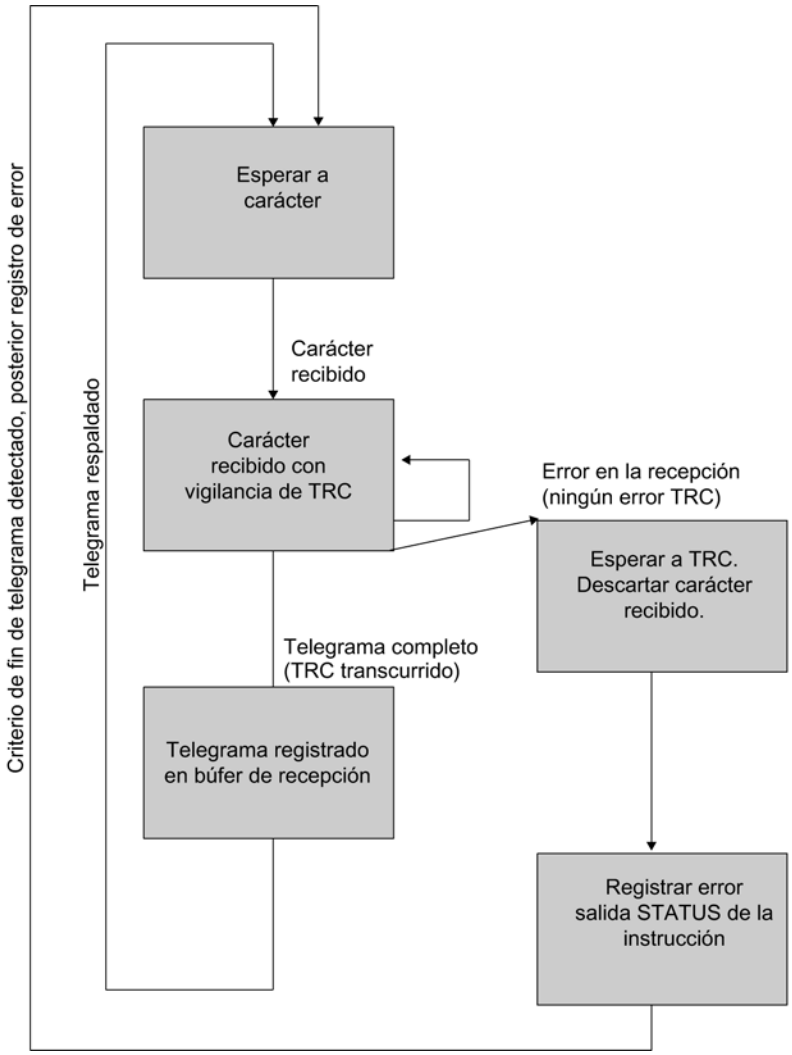


Figura 8-21 Desarrollo del proceso de recepción con el criterio de fin "Transcurrido el tiempo de retardo entre caracteres"

Criterio de fin "Carácter de fin"

Al recibir datos se detecta el fin del telegrama si se recibe(n) el(los) carácter(es) de fin parametrizado(s). Los datos recibidos, incluidos los caracteres de fin, los acepta la CPU.

El transcurso del tiempo de retardo entre caracteres durante la recepción implica el final de la recepción. Se genera un mensaje de error y el fragmento de telegrama se rechaza.

Cuando se trabaja con caracteres de fin, la transferencia no tiene transparencia del código y debe excluirse la posibilidad de que el(los) identificador(es) de fin aparezcan en los datos útiles.

Cuando en el telegrama recibido el último carácter no sea el carácter de fin, tenga en cuenta lo siguiente:

- Los caracteres de fin se encuentran en la posición deseada del telegrama:
  todos los caracteres, incluidos los caracteres de fin, se incluyen en el DB de recepción.
  Los caracteres que se encuentren detrás del carácter de fin
  - se rechazan una vez haya transcurrido el tiempo de retardo de carácter (TRC), al final del telegrama.
  - se fusionan con el siguiente telegrama al recibir uno nuevo antes de que transcurra el tiempo de retardo de carácter.
- El telegrama no contiene caracteres de fin:
  El telegrama
  - se rechaza una vez haya transcurrido el tiempo de retardo de carácter (TRC), al final del telegrama.
  - se fusiona con el siguiente telegrama al recibir uno nuevo antes de que transcurra el tiempo de retardo de carácter.

En la siguiente figura encontrará las fases de recepción con el criterio de fin "Caracteres de fin".

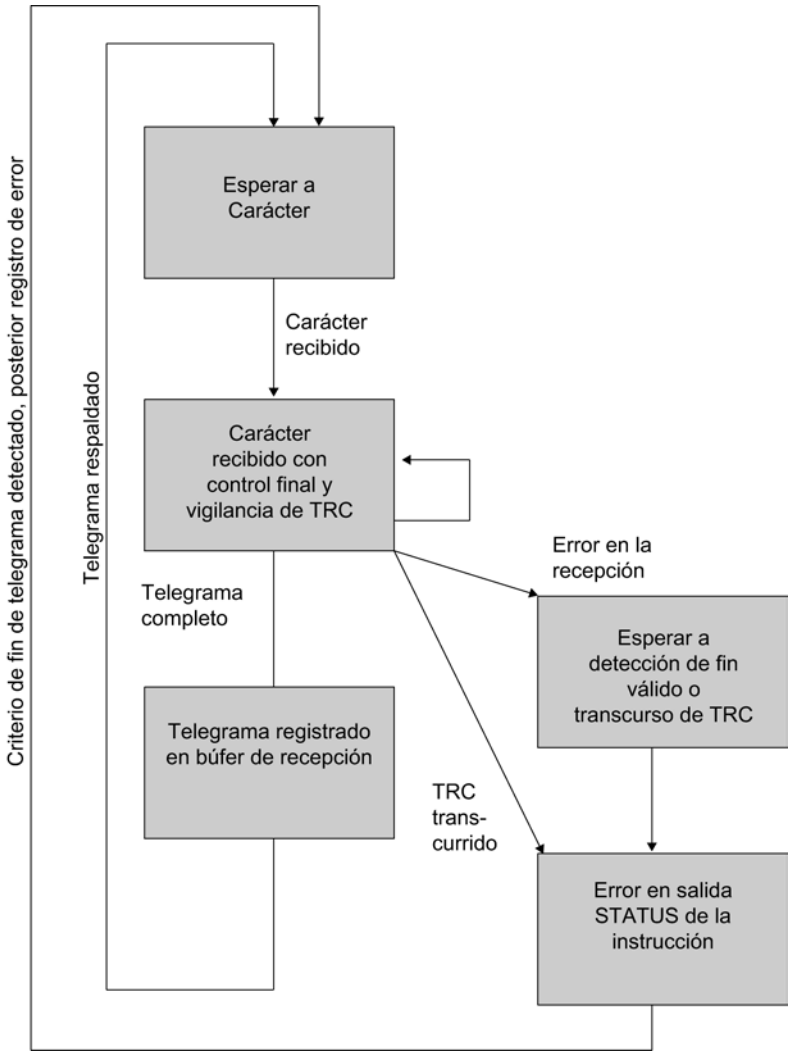


Figura 8-22 Desarrollo del proceso de recepción con el criterio de fin "Caracteres de fin"

Criterio de fin "Longitud fija de telegrama"

Al recibir datos se detecta el fin del telegrama si se ha recibido el número de caracteres parametrizado. Los datos recibidos los acepta la CPU.

El transcurso del tiempo de retardo entre caracteres antes de alcanzar el número parametrizado de caracteres implica el final de la recepción. Se genera un mensaje de error y el fragmento de telegrama se rechaza.

Cuando la longitud del telegrama de los caracteres recibidos no coincida con la longitud fijada, tenga en cuenta lo siguiente:

- Longitud del telegrama de los caracteres recibidos mayor que la longitud de telegrama fija parametrizada:
  Todos los caracteres que se reciban tras alcanzar la longitud parametrizada fija de telegrama
  - se rechazan una vez haya transcurrido el tiempo de retardo de carácter (TRC), al final del telegrama.
  - se fusionan con el siguiente telegrama al recibir uno nuevo antes de que transcurra el tiempo de retardo de carácter.
- Longitud del telegrama de los caracteres recibidos menor que la longitud de telegrama fija parametrizada:
  El telegrama
  - se rechaza una vez haya transcurrido el tiempo de retardo de carácter (TRC), al final del telegrama.
  - se fusiona con el siguiente telegrama al recibir uno nuevo antes de que transcurra el tiempo de retardo de carácter.

En la siguiente figura encontrará las fases de recepción con el criterio de fin "Longitud fija de telegrama".

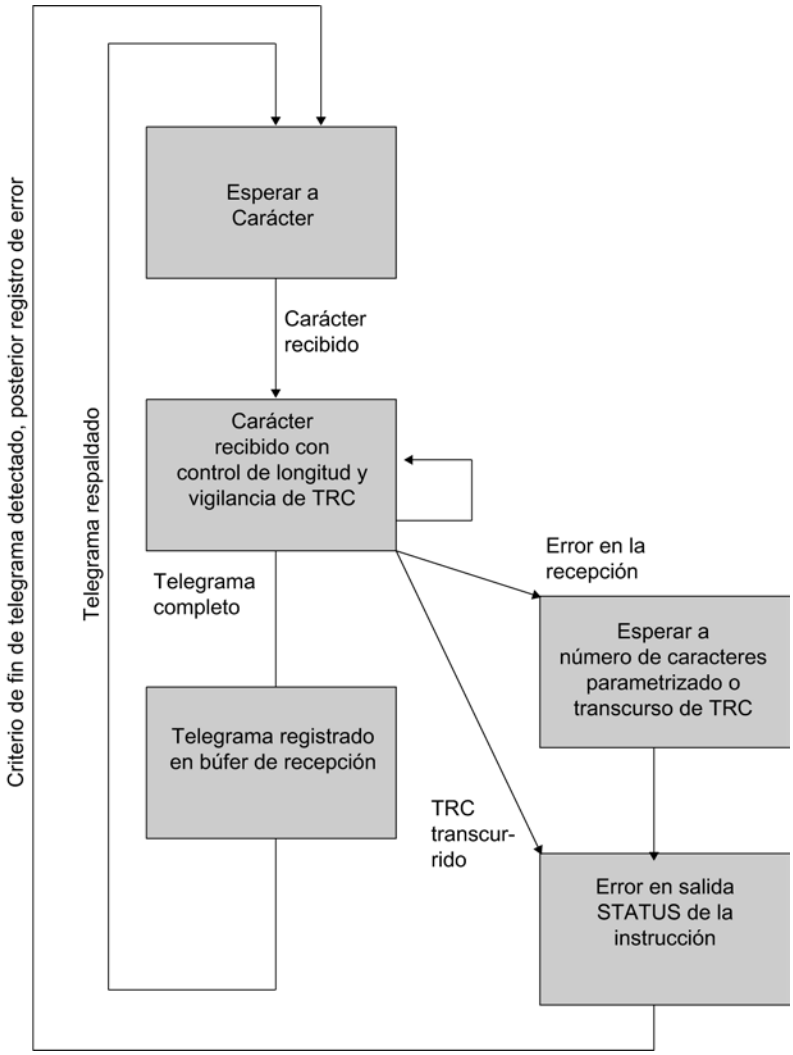


Figura 8-23 Desarrollo del proceso de recepción con el criterio de fin "Longitud fija de telegrama"

Búfer de recepción del módulo

Dependiendo del módulo de comunicación utilizado, el búfer de recepción del módulo puede alojar hasta 4096 bytes.

Durante la parametrización puede indicar si:

- CP y CPU 31xC-2 PtP
  el búfer de recepción debe borrarse durante el arranque y si debe evitarse una sobrescritura de los datos contenidos en el mismo. Además puede indicar el rango de valores (de 1 a 250) para el número de telegramas de recepción guardados en el búfer.
- ET 200S 1SI
  Se borra el búfer de recepción durante el arranque y si debe evitarse una sobrescritura de los datos contenidos en el mismo. Además puede activar o bloquear el respaldo de los telegramas recibidos.

El búfer de recepción del módulo es un búfer en anillo:

- Si se incluyen varios telegramas en el búfer de recepción del módulo: siempre se enviará el telegrama más antiguo del módulo a la CPU.
- Si se desea enviar siempre el telegrama más reciente a la CPU, tiene que:
  - CP y CPU 31xC-2 PtP
    parametrizar el valor "1" para el número de telegramas respaldados y desactivar la protección de sobrescritura.
  - ET 200S 1SI
    bloquear los telegramas dinámicos y desactivar la protección de sobrescritura.

Nota

Si la lectura continua de los datos recibidos en el programa de usuario se interrumpe durante un tiempo, al volver a solicitar los datos recibidos puede ocurrir que la CPU reciba primero telegramas antiguos del módulo y después el último telegrama.

Los telegramas más antiguos se encontraban de camino entre el módulo y la CPU durante la interrupción o la instrucción ya los habían recibido.

Modo RS485 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI)

Modo RS485

Si utiliza el driver ASCII en modo RS485 (semidúplex, modo de dos cables), debe garantizar mediante el programa de usuario que nunca pueda enviar más de una estación. Si envían más de una simultáneamente, el telegrama se falsifica.

Tiempos de conmutación para el módulo RS485 en modo semidúplex

El tiempo máximo de conmutación entre el envío y la recepción es de 1 ms.

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI)

Señales cualificadoras RS232C

Las siguientes señales cualificadoras RS232C estarán disponibles en los módulos al utilizar la física RS232C:

DCD	(Entrada)	Data Carrier detect, soporte de datos detectado
DTR	(Salida)	Data terminal ready; módulo listo para funcionar
DSR	(Entrada)	Data set ready, interlocutor listo para funcionar
RTS	(Salida)	Request to send, módulo listo para enviar

CTS	(Entrada)	Clear to send, el interlocutor puede recibir datos del módulo (respuesta a RTS = ON del módulo de comunicación)
RI	(Entrada)	Ring Indicator, indicador de llamada

Tras la conexión del módulo las señales de salida están en estado OFF (inactivas).

El procesamiento de las señales de control DTR/DSR y RTS/CTS puede parametrizarse en el cuadro de diálogo de propiedades del móduloo controlarse mediante las instrucciones del programa de usuario.

Manejo de señales cualificadoras RS232C

Las señales cualificadoras RS232C pueden utilizarse de la siguiente manera:

- en la parametrización del uso automático de todas las señales cualificadoras de RS232C
- con flujo de datos parametrizado (RTS/CTS)
- mediante las instrucciones V24_STAT y V24_SET

Nota

Si se parametriza el uso automático de las señales cualificadoras de RS232C no es posible ni el control del flujo de datos con RTS/CTS ni el control de RTS y DTR mediante la instrucción V24_SET. Si se parametriza un control del flujo de datos con RTS/CTS no es posible controlar RTS con la instrucción V24_SET. La lectura de todas las señales cualificadoras RS232C con la instrucción V24_STAT es posible en todos los casos.

A continuación se explica el procedimiento básico para controlar y evaluar las señales cualificadoras de RS 232C.

Manejo automático de las señales cualificadoras

El manejo automático de las señales cualificadoras RS232C en el módulo de comunicación se establece de la siguiente manera:

- Cuando el módulo entra por parametrización en un modo de operación con manejo automático de las señales cualificadoras de RS232C, pone las líneas RTS en estado OFF y las DTR en estado ON (módulo de comunicación preparado para entrar en funcionamiento).
  Sólo es posible enviar y recibir telegramas tras poner la línea DTR a ON. Mientras DTR permanezca en OFF, no se pueden recibir datos vía la interfaz del RS232C. Cualquier petición de envío se interrumpe con el correspondiente mensaje de error.
- Si está pendiente una petición de envío, RTS pasa al estado ON y se inicia el tiempo de espera de salida de datos parametrizado. Una vez transcurrido el tiempo de salida de datos y con CTS = ON, los datos se envían a través de la interfaz RS232C.
- Si durante el envío la línea CTS no pasa a ON durante el tiempo de espera, o si durante el proceso de salida, CTS cambia al estado OFF, se interrumpirá el envío mostrando el aviso correspondiente.
- Tras la emisión de los datos y una vez transcurrido el tiempo de RTS a OFF parametrizado, la línea RTS pasa al estado de inactividad. No se espera a que la línea CTS pase a OFF.

- Es posible recibir datos vía la interfaz RS232C tan pronto como se active la línea DSR (ON). Si existe riesgo de sobrecarga del búfer de recepción del módulo de comunicación, el módulo no reacciona.
- Al cambiar de DSR = ON a OFF se interrumpe tanto el envío en curso como la recepción de datos con un aviso de error. En el búfer de diagnóstico del módulo se registra el aviso “DSR = OFF (manejo automático de las señales V24)".

Nota

Si se parametriza el uso automático de las señales cualificadoras de RS232C no es posible ni el control del flujo de datos con RTS/CTS ni el control de RTS y DTR mediante la instrucción V24_SET.

El "tiempo RTS a OFF" debe definirse en el cuadro de propiedades del módulo de tal manera que el interlocutor pueda recibir completos los últimos caracteres del telegrama antes de que la línea RTS, y con ella la petición de envío, se desactiven. El "tiempo de salida de datos" debe definirse de tal manera que el interlocutor pueda pasar a modo de disponibilidad de recepción antes de que el tiempo haya transcurrido.

Cronograma

La figura muestra el desarrollo temporal de una petición de envío.

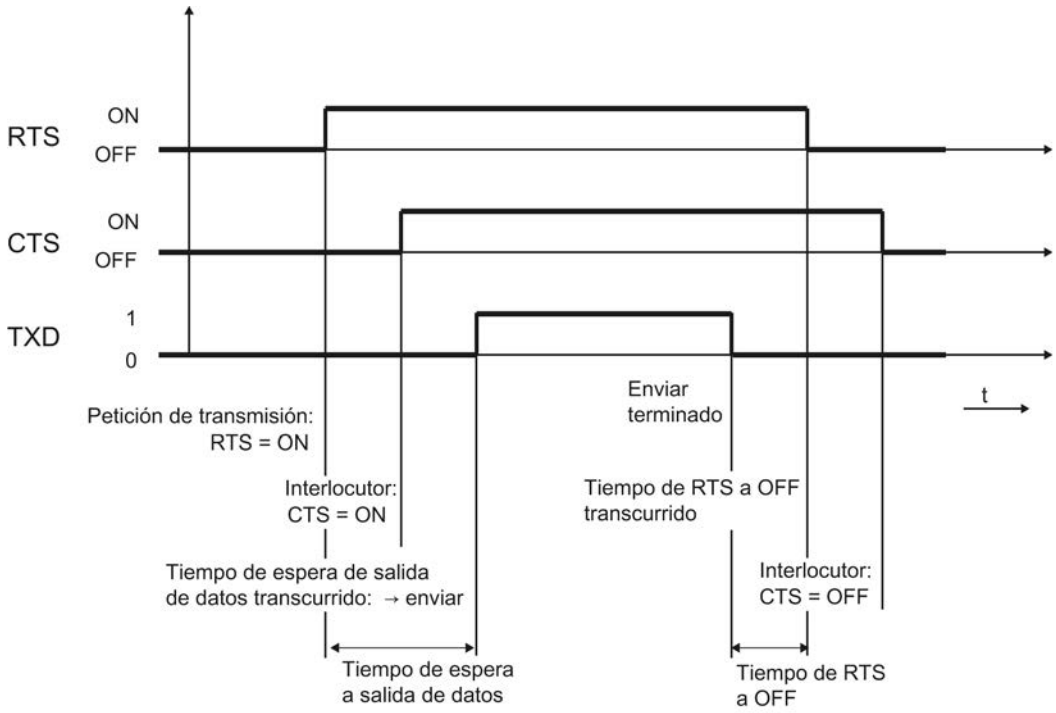


Figura 8-24 Cronograma para el manejo automático de las señales cualificadoras de RS232C

Funciones de la instrucción V24_STAT/SET

La instrucción V24_STAT permite determinar el estado de cada señal cualificadora RS232C. La instrucción V24_SET permite controlar las señales de salida DTR y RTS.

Handshake de software/hardware

Introducción

Los procedimientos handshake controlan el flujo de datos entre dos interlocutores. Con la utilización de procedimientos Handshake se evita que se pierdan datos durante la transmisión de datos entre dispositivos que trabajan a diferentes velocidades.

Procedimiento Handshake

Se distinguen dos procedimientos:

- Handshake de software (p. ej. XON/XOFF)
- Handshake de hardware (p. ej. RTS/CTS)

El control del flujo de datos en el módulo de comunicación se desarrolla de la siguiente manera:

- En cuanto el módulo pasa por parametrización a un modo de operación con control de flujo, envía el carácter XON o pone la línea RTS en estado ON.
- Al alcanzar el número de telegrama parametrizado o bien los 50 caracteres antes de que se desborde el búfer de recepción (capacidad del búfer de recepción: 4096 bytes), el módulo envía el carácter XOFF o pone la línea RTS en estado OFF. Si a pesar de ello el interlocutor continúa enviando, se genera un mensaje de error si el búfer de recepción se desborda. Los datos recibidos del último telegrama se descartan.
- En cuanto la CPU-S7 ha captado un telegrama, y una vez que el búfer de recepción está listo para recibir, el módulo envía el carácter XON o pone la línea RTS en estado ON.
- Si el módulo recibe el carácter XOFF o si la señal de control CTS pasa a estado OFF, el módulo interrumpe el proceso de envío. Si transcurrido un tiempo determinado parametrizable no se recibe ningún carácter XON o CTS no pasa a ON, el proceso de envío se interrumpe y se registra el correspondiente mensaje de error (0708H) en el área de registro de errores del módulo.

Nota

Al definir el control de flujo de datos con RTS/CTS es necesario cablear por completo las señales de interfaz utilizadas en la conexión por conector. Si se parametriza un control del flujo de datos con RTS/CTS no es posible controlar RTS con la instrucción V24_SET.

Transmisión de datos con 3964(R)

Transmisión de datos con el procedimiento 3964(R)

Para la asignación de los protocolos a los módulos de comunicación véase: Sinopsis de los componentes e interfaces (Página 1068)

Introducción

El procedimiento 3964(R) controla la transmisión de datos en un acoplamiento punto a punto entre el procesador de comunicaciones y un interlocutor. El procedimiento 3964(R) contiene, además del nivel de transmisión de bits (nivel 1), el nivel de seguridad (nivel 2).

Arranque

En la siguiente figura encontrará las fases de arranque del procedimiento 3964(R).

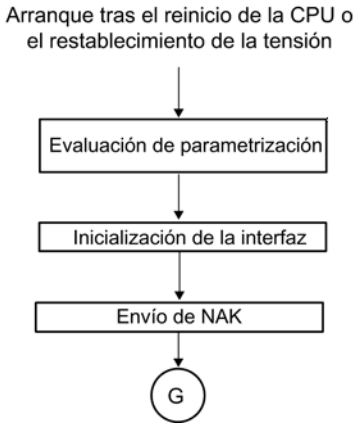


Figura 8-25 Esquema de desarrollo del arranque del procedimiento 3964(R)

Caracteres de control

Introducción

El procedimiento 3964(R) añade los caracteres de control de los datos de información (nivel de seguridad) durante la transmisión de datos. Mediante estos caracteres de control, el interlocutor puede controlar si los datos le llegan de forma completa y sin errores.

Caracteres de control del procedimiento 3964(R)

El procedimiento 3964(R) valora los siguientes caracteres de control:

- STX Start of Text; inicio de la secuencia de caracteres que debe transmitirse
- DLE Data Link Escape; conmutación de transmisión de datos
- ETX End of Text; fin de la secuencia de caracteres que debe transmitirse

- BCC Block Check Character (sólo con 3964(R)); carácter de comprobación de bloque
- NAK Negative Acknowledge; realimentación negativa

Nota

Si se transmite el carácter DLE como carácter de información, éste se enviará doblemente para distinguir el carácter de control DLE durante el establecimiento y la desconexión de la comunicación en la línea de transferencia (duplicado de DLE). El receptor anula el duplicado del carácter DLE.

Prioridad

En el procedimiento 3964(R) un interlocutor debe tener asignada una prioridad mayor que el otro. Si ambos interlocutores inician simultáneamente una petición de transmisión, el interlocutor que tiene la prioridad más baja retira su petición de transmisión.

Suma de comprobación de bloque

Suma de comprobación de bloque

En el protocolo de transmisión 3964(R) la seguridad de datos se aumenta mediante un carácter de comprobación de bloque (BCC = Block Check Character) enviado adicionalmente.

Telegrama:

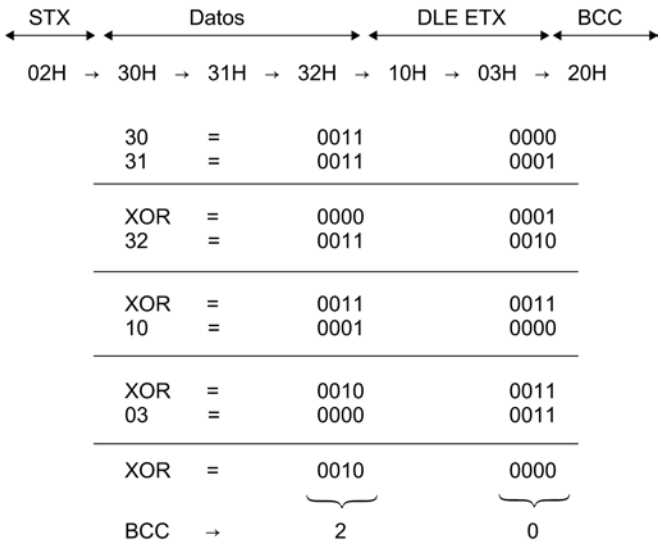


Figura 8-26 Suma de comprobación de bloque

La suma de comprobación de bloque es la paridad longitudinal par (operación lógica EXOR de todos los bytes de datos) de un bloque enviado o recibido. La formación se inicia con el

primer byte de datos útiles (1er byte del telegrama) tras el establecimiento de la conexión y finaliza después del carácter DLE ETX al interrumpir la conexión.

Nota

En un duplicado DLE, el carácter DLE se incluye dos veces en la formación del BCC.

Enviar datos con 3964(R)

Desarrollo del proceso de transmisión de datos en el envío

En la figura siguiente se representa el desarrollo de la transmisión de datos al enviar con el procedimiento 3964(R).

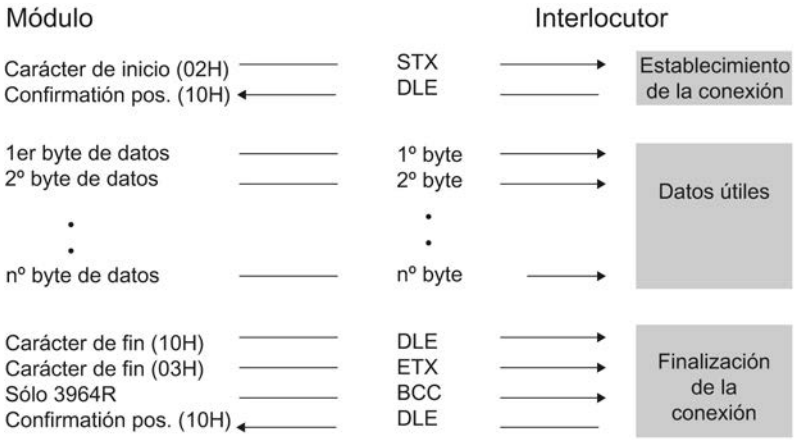


Figura 8-27 Intercambio de datos en el envío con el procedimiento 3964(R)

Establecimiento de la conexión en el envío

Para establecer la conexión, el procedimiento 3964(R) envía el carácter de control STX. Si el interlocutor responde con el carácter DLE antes de que transcurra el tiempo de retardo del acuse (TRA), el procedimiento entra en la fase de envío.

Si, por el contrario, el interlocutor responde con NAK, otro carácter cualquiera (con la excepción de DLE o STX) o si el tiempo de retardo del acuse transcurre sin que se produzca una reacción, el procedimiento repite el establecimiento de la conexión. Tras el número parametrizado de intentos fallidos, el procedimiento interrumpe el establecimiento y envía el carácter NAK al interlocutor. El módulo de comunicación registra el número de error correspondiente en su área de registro de errores.

Enviar datos

Si el establecimiento de la conexión tiene éxito, los datos útiles contenidos en el búfer de salida del procesador de comunicaciones se envían al interlocutor con los parámetros de

transmisión seleccionados. El interlocutor controla la distancia temporal de los caracteres entrantes. La distancia entre dos caracteres no debe ser mayor al tiempo de retardo de caracteres.

Si el interlocutor envía el carácter NAK durante un envío, el procedimiento aborta el proceso y lo repite del modo arriba descrito, comenzando con el establecimiento de la conexión. Si el interlocutor envía cualquier otro carácter, el procedimiento espera primero a que transcurra el tiempo de retardo de caracteres y a continuación envía el carácter NAK para que el interlocutor pase a estado de reposo. Entonces el procedimiento comienza de nuevo la fase de envío con el establecimiento de la conexión STX.

Interrupción de la conexión en el envío

Una vez que se ha enviado el contenido del búfer, el procedimiento añade los caracteres DLE, ETX en el caso de 3964R el carácter de comprobación de bloques BCC como carácter de fin, y espera a un carácter de acuse. Si el interlocutor envía el carácter DLE dentro del tiempo de retardo de acuse, el bloque de datos se recibe sin errores. Si, por el contrario, el interlocutor responde con NAK, otro carácter cualquiera (con la excepción de DLE), un carácter erróneo, o si el tiempo de retardo de acuse transcurre sin que se produzca una reacción, el procedimiento comienza de nuevo el envío con el establecimiento de la conexión STX.

Tras un número preestablecido de intentos de envío del bloque de datos, el procedimiento interrumpe el proceso y envía el carácter NAK al interlocutor. El módulo de comunicación registra el error en el área de registro de errores.

Fases del envío con el procedimiento 3964(R)

En la siguiente figura encontrará las fases de envío del procedimiento 3964(R).

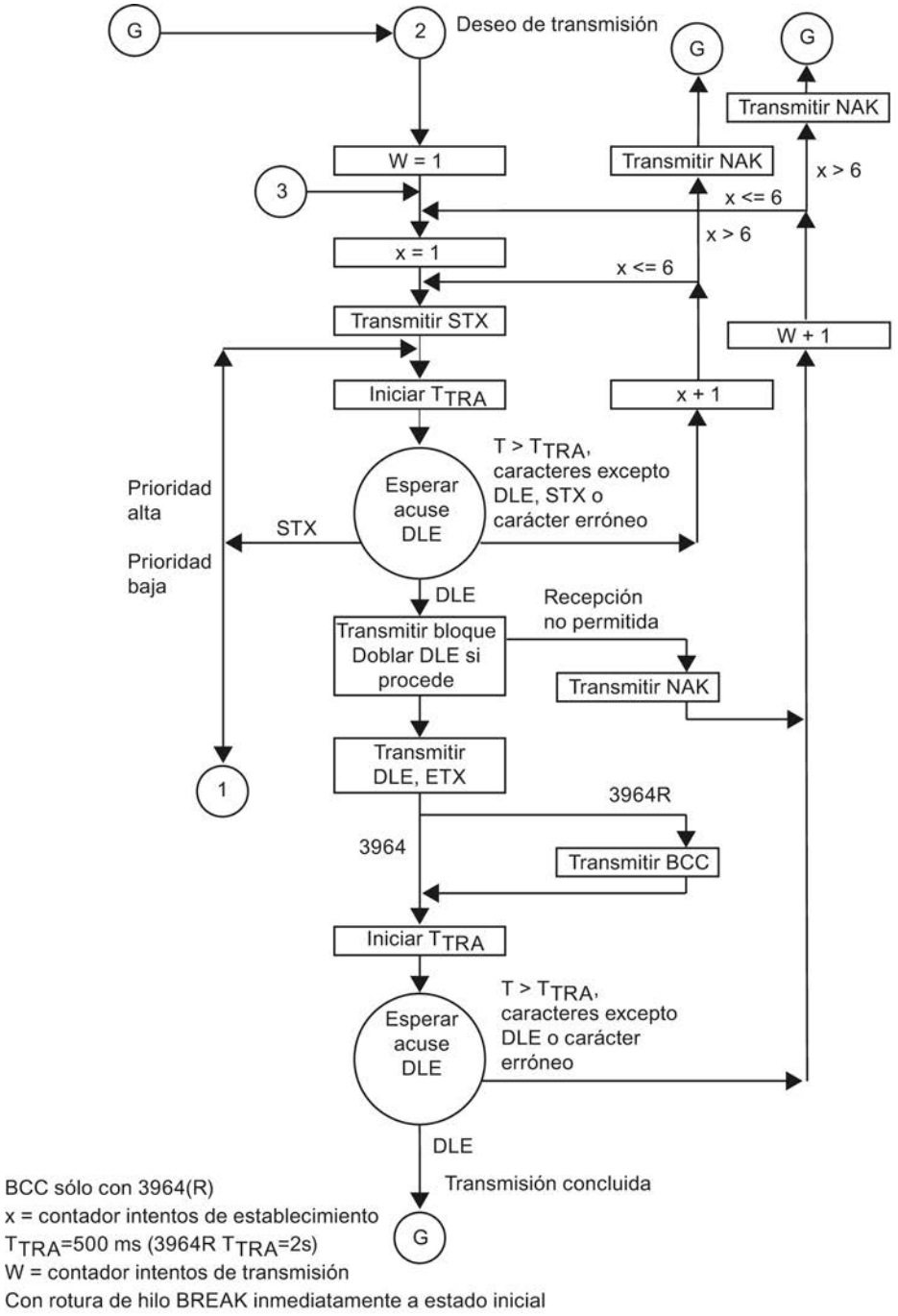


Figura 8-28 Esquema de desarrollo del envío con el procedimiento 3964(R)

Recibir datos con 3964(R)

Desarrollo del proceso de transmisión de datos en la recepción

En la siguiente imagen se representa el desarrollo de la transmisión de datos al recibir con el procedimiento 3964(R).

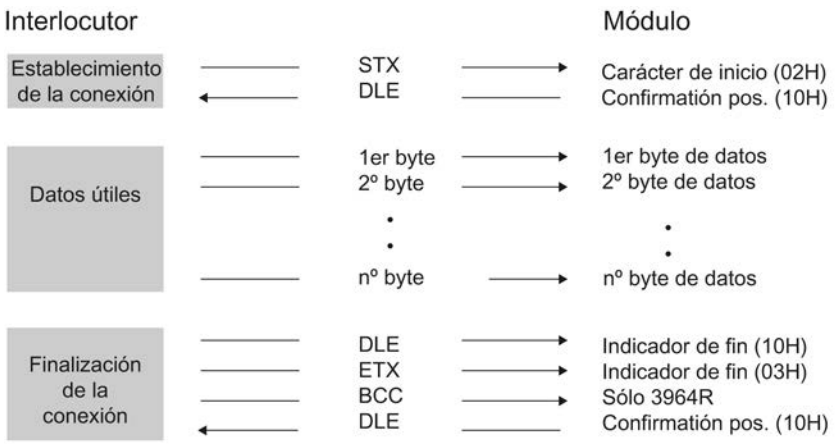


Figura 8-29 Intercambio de datos en la recepción con el procedimiento 3964(R)

Nota

En cuanto está preparado, el procedimiento 3964(R) comienza a enviar el carácter NAK al interlocutor para que éste pase a estado de reposo.

Establecimiento de la conexión en la recepción

En estado de reposo, cuando no debe procesarse ninguna orden de envío, el procedimiento espera a que el interlocutor establezca la conexión.

Si durante el establecimiento de la conexión con STX no hay ningún búfer de recepción disponible, se inicia un tiempo de espera de 400 ms. Si una vez transcurrido este tiempo continúa sin haber ningún búfer disponible, el programa registra el error en el área de registro de errores del módulo. El procedimiento envía un carácter NAK y regresa al estado de reposo. En caso contrario, el procedimiento envía el carácter DLE y recibe los datos como se ha descrito.

Si el procedimiento recibe cualquier carácter (con la excepción de STX o NAK) durante el modo de reposo, el procedimiento espera primero a que transcurra el tiempo de retardo de carácter y a continuación envía un carácter NAK. El módulo de comunicación registra el error en el área de registro de errores del módulo.

Recibir datos

Una vez que la conexión se ha establecido con éxito, los datos entrantes se recogen en el búfer de recepción. Si se reciben dos caracteres DLE consecutivos, sólo uno de ellos se registra en el búfer.

Después de cada carácter recibido, se espera al siguiente durante el tiempo de retardo de caracteres. Si éste transcurre sin ninguna recepción, se envía el carácter NAK al interlocutor. El módulo de comunicación registra el error en el área de registro de errores del módulo. El procedimiento 3964(R) no inicia una repetición.

Si durante la recepción se producen errores de transmisión (carácter perdido, error de trama, error de paridad, etc.), se continúa recibiendo hasta que se produce la interrupción de la conexión y a continuación se envía el carácter NAK al interlocutor. A continuación se espera una repetición. Si el bloque no puede recibirse sin errores tras el número de intentos especificado en el registro estático de parámetros, o si el interlocutor no inicia la repetición dentro de un tiempo de espera de 4 s, el procedimiento interrumpe la recepción. El módulo de comunicación registra la primera transmisión errónea y la interrupción final del proceso en el área de registro de errores del módulo.

Interrupción de la conexión en la recepción

Cuando el procedimiento 3964 reconoce la secuencia de caracteres DLE ETX, finaliza la recepción y envía el carácter DLE al interlocutor si ha recibido un bloque sin errores. Si, por el contrario, se produce un error de transmisión, envía el carácter NAK al interlocutor. A continuación se espera una repetición.

Cuando el procedimiento 3964R reconoce la secuencia de caracteres DLE ETX BCC, finaliza la recepción. Compara el carácter de comprobación de bloques BCC con la paridad longitudinal calculada internamente. Si el carácter de comprobación de bloques es correcto y no ha ocurrido ningún otro error en la recepción, el procedimiento 3964R envía el carácter DLE y vuelve al estado de reposo. Si hay un error en el BCC o si se produce cualquier otro error de recepción, se envía el carácter NAK al interlocutor. A continuación se espera una repetición.

Recepción con el procedimiento 3964(R)

En las siguientes figuras encontrará las fases de recepción del procedimiento 3964(R).

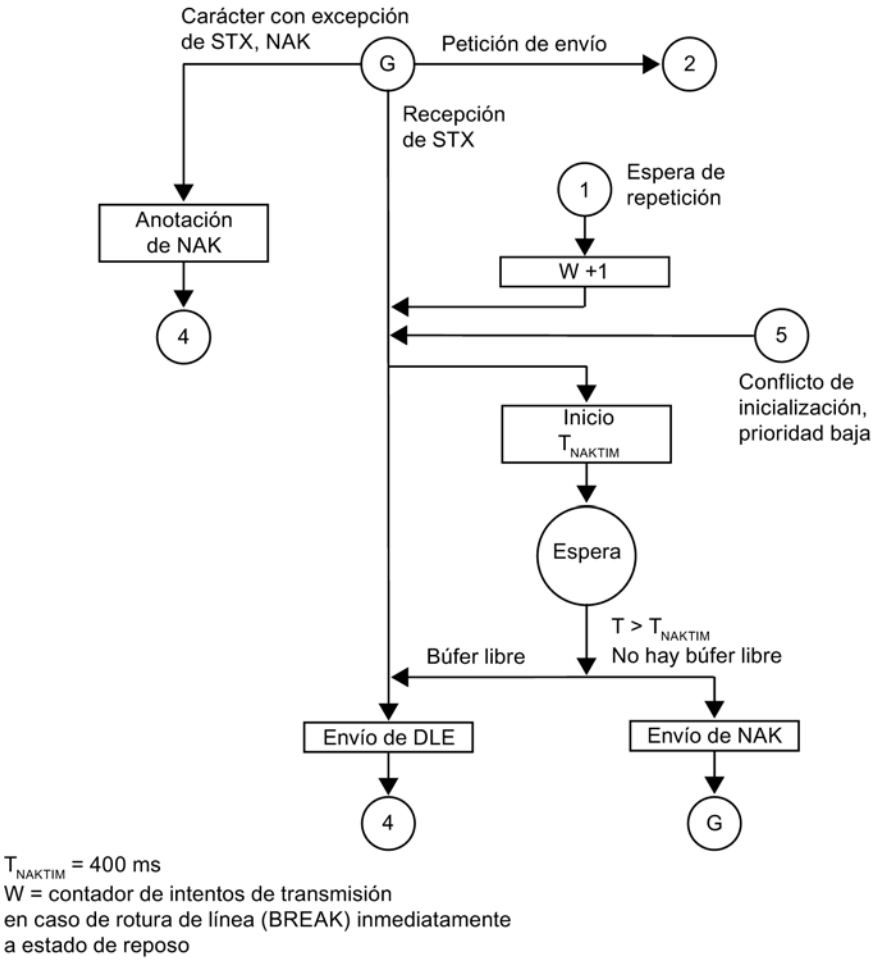


Figura 8-30 Esquema de procesos en la recepción con el procedimiento 3964(R) (parte 1)

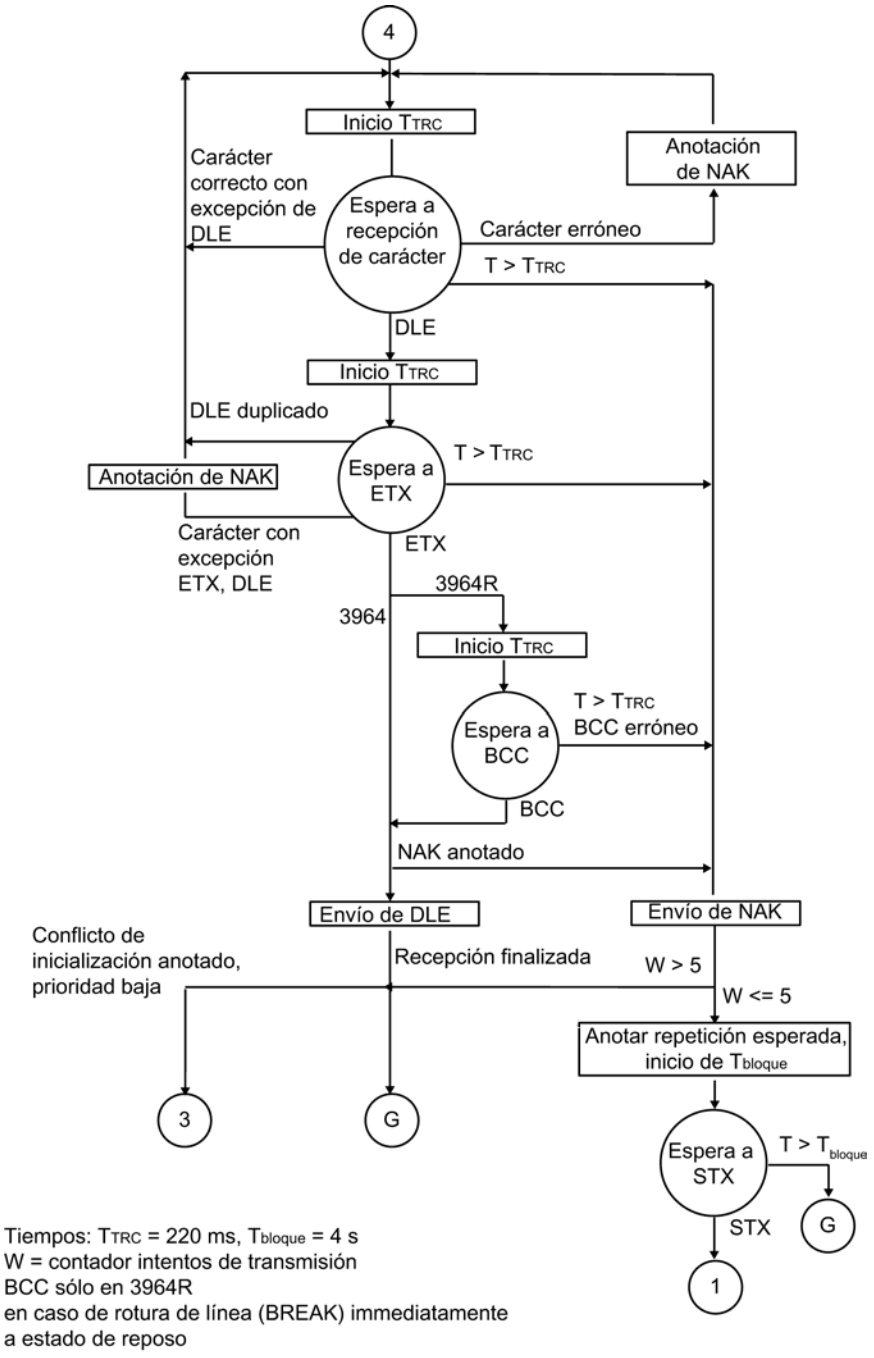


Figura 8-31 Esquema de procesos en la recepción con el procedimiento 3964(R) (parte 2)

Transmisión de datos con RK512

Transmisión de datos con el protocolo para mainframe RK512

Para la asignación de los protocolos a los módulos de comunicación véase: Sinopsis de los componentes e interfaces (Página 1068)

Introducción

El protocolo para mainframe RK512 controla la transmisión de datos en un acoplamiento punto a punto entre el módulo de comunicación y un interlocutor.

A diferencia del procedimiento 3964(R), el protocolo para mainframe RK512 contiene además del nivel de transmisión de bits (nivel 1) y del nivel de seguridad (nivel 2) el nivel de transporte (nivel 4). El protocolo para mainframe RK512 ofrece además una mayor protección de los datos y posibilidades de direccionamiento avanzadas.

telegrama de reacción

El protocolo para mainframe RK512 responde a cada telegrama de orden recibido correctamente con un telegrama de reacción al ordenante (nivel de transporte). De ese modo el remitente puede comprobar si sus datos han llegado correctamente a la CPU del interlocutor o bien si los datos solicitados están disponibles en la misma.

Telegrama de orden

Los telegramas de orden pueden ser telegramas SEND o bien telegramas FETCH.

Telegrama SEND

En un telegrama SEND, el módulo de comunicación envía un telegrama de orden con datos útiles y el interlocutor responde con un telegrama de reacción sin datos útiles.

Telegrama FETCH

En un telegrama FETCH, el módulo de comunicación envía un telegrama de orden sin datos útiles y el interlocutor responde con un telegrama de reacción con datos útiles.

Telegrama de continuación

Si la cantidad de datos supera los 128 bytes, en el caso de telegramas SEND y FETCH se envían automáticamente telegramas de continuación.

Encabezado de telegrama

Cada telegrama de RK512 comienza con un encabezado. Éste puede contener datos identificadores del telegrama, información sobre el destino y la fuente de los datos y un número de error.

Estructura del encabezado de telegrama

En la siguiente tabla puede verse la estructura del encabezado del telegrama de orden.

Tabla 8-73 Estructura del encabezado del telegrama de orden (RK512)

4º byte	Significado
1	El identificador de telegrama en telegramas de orden (00H), en telegramas de continuación de orden (FFH)
2	Identificador de telegrama (00H)
3	'A' (41H) para petición SEND con DB de destino o bien 'O' (4FH) para petición SEND con DX de destino o bien 'E' (45H) para petición FETCH
4	los datos que deben transferirse son (en el envío sólo es posible 'D'): 'D' (44H) = bloque de datos 'I' (45H) = bytes de entrada 'Q' (41H) = bytes de salida 'M' (4DH) = bytes de marca 'C' (5AH) = celdas de contador 'T' (54H) = celdas de temporizador
5 y 6	Destino de los datos en petición SEND/PUT o bien Fuente de los datos en petición GET p. ej. byte 5 = nº DB, byte 6 = nº DW. (El direccionamiento de RK512 describe la fuente y el destino de los datos con límites de palabra. El cálculo de las direcciones de bytes en SIMATIC S7 se produce automáticamente.)
7 y 8	Longitud High-Byte Longitud de los datos a transmitir según el tipo en bytes o bien Longitud Low-Byte Palabras
9	Número de bytes de la marca de acoplamiento; si no ha indicado ninguna marca de acoplamiento, aquí aparece FFH.
10	**Bit 0 a 3:** Número de bits de la marca de acoplamiento, si ha especificado una marca de acoplamiento, el protocolo registra aquí FH. **Bit 4 a 7:** Número de CPU (de 1 a 4); si no ha especificado ningún número de CPU, pero ha indicado una marca de acoplamiento, aquí aparece 0H; si no ha indicado ni un número de CPU ni una marca de acoplamiento, aquí aparece FH.

Las letras de los bytes 3 y 4 son caracteres ASCII.

El encabezado del telegrama de continuación de orden sólo se compone de los bytes 1 a 4.

Telegrama de reacción

Una vez transferido el telegrama de orden, RK512 espera un telegrama de reacción del interlocutor durante un tiempo de vigilancia fijo. La longitud del tiempo de vigilancia es de 20 s, independientemente de la velocidad de transferencia (baudrate).

Dependiendo del módulo PtP utilizado, el usuario puede reducir el tiempo de vigilancia mediante parametrización en el cuadro de diálogo de propiedades del módulo. Seleccionando la opción "según la velocidad de transferencia" se vigila con los tiempos de espera máximos indicados a continuación:

Tabla 8-74 Tiempo de vigilancia del telegrama de reacción en función de la velocidad de transferencia

Velocidad de transferencia	Tiempo de vigilancia
• 300 baudios	10 s
• 600 baudios	7 s
• 1200 baudios	5 s
• A partir de 38400 baudios	3 s

El campo gris "Tiempo de espera máximo" sólo indica el tiempo de vigilancia utilizado, pero no puede editarse.

Estructura y contenido del telegrama de reacción

El telegrama de reacción se compone de 4 bytes y contiene información sobre el desarrollo de la petición.

Byte	Significado
1	Identificador de telegrama en telegramas de respuesta (00H), en telegramas de continuación de respuesta (FFH)
2	Identificador de telegrama (00H)
3	Ocupado con 00H
4	Número de error del interlocutor en el telegrama de reacción: • **00H** si la transmisión ha transcurrido sin problemas • **> 00H** número de error El número de error en el telegrama de reacción provoca automáticamente la entrada de un número de error en el área de registro de errores del módulo.

Enviar datos con RK512

Desarrollo del proceso de transmisión de datos en el envío

La siguiente figura representa el desarrollo del proceso de transmisión de datos durante el envío con un telegrama de reacción con el protocolo para mainframe RK512.

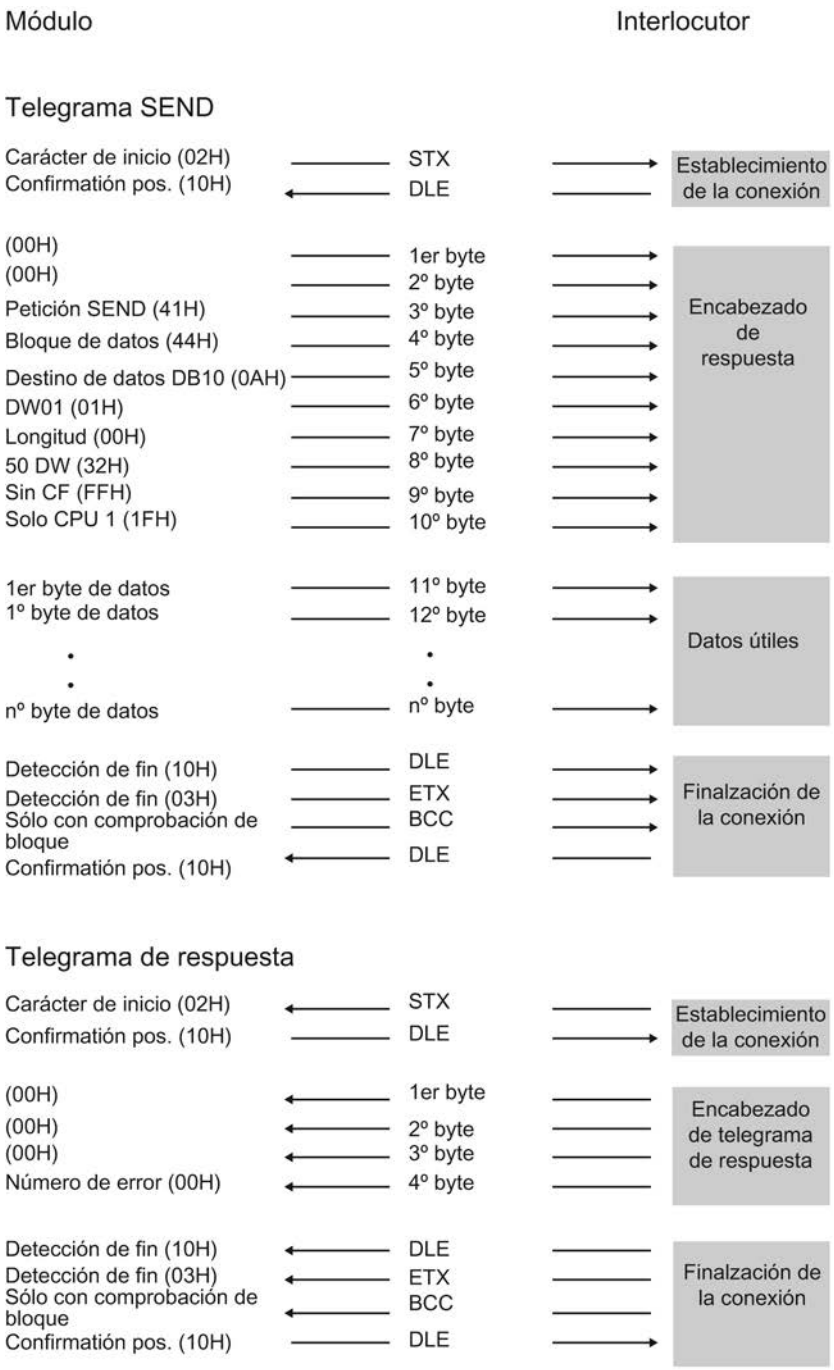


Figura 8-32 Intercambio de datos al enviar con telegrama de reacción

Enviar datos

La petición SEND se desarrolla en el siguiente orden:

- Interlocutor activo
  Envía un telegrama SEND. Éste contiene datos y el encabezado del telegrama.
- Interlocutor pasivo
  Recibe el telegrama, comprueba el encabezado y los datos y confirma con un telegrama de reacción tras realizar la transmisión de dichos datos a la CPU.
- Interlocutor activo
  Recibe el telegrama de reacción y termina la petición SEND.

Nota

Si la CPU no ha recibido los datos útiles de la petición SEND sin errores o si se ha producido un error en el encabezado del telegrama, el interlocutor inserta un número de error en el 4º byte del telegrama de reacción. Si se trata de fallos de protocolo, no se produce ninguna entrada en el telegrama de reacción.

Telegrama SEND de continuación

Se inicia una petición SEND de continuación cuando la cantidad de datos sobrepasa los 128 bytes. El procedimiento se corresponde con el de la petición SEND.

Si se envían más de 128 bytes, éstos se transmiten automáticamente en uno o más telegramas de continuación. Cada telegrama SEND de continuación se acusa con un telegrama de reacción de continuación

La siguiente figura representa el desarrollo del proceso de transmisión de datos al enviar un telegrama SEND de continuación con un telegrama de reacción de continuación.

El telegrama de reacción de continuación al último telegrama SEND de continuación no se envía hasta haber enviado todos los datos útiles a la CPU.

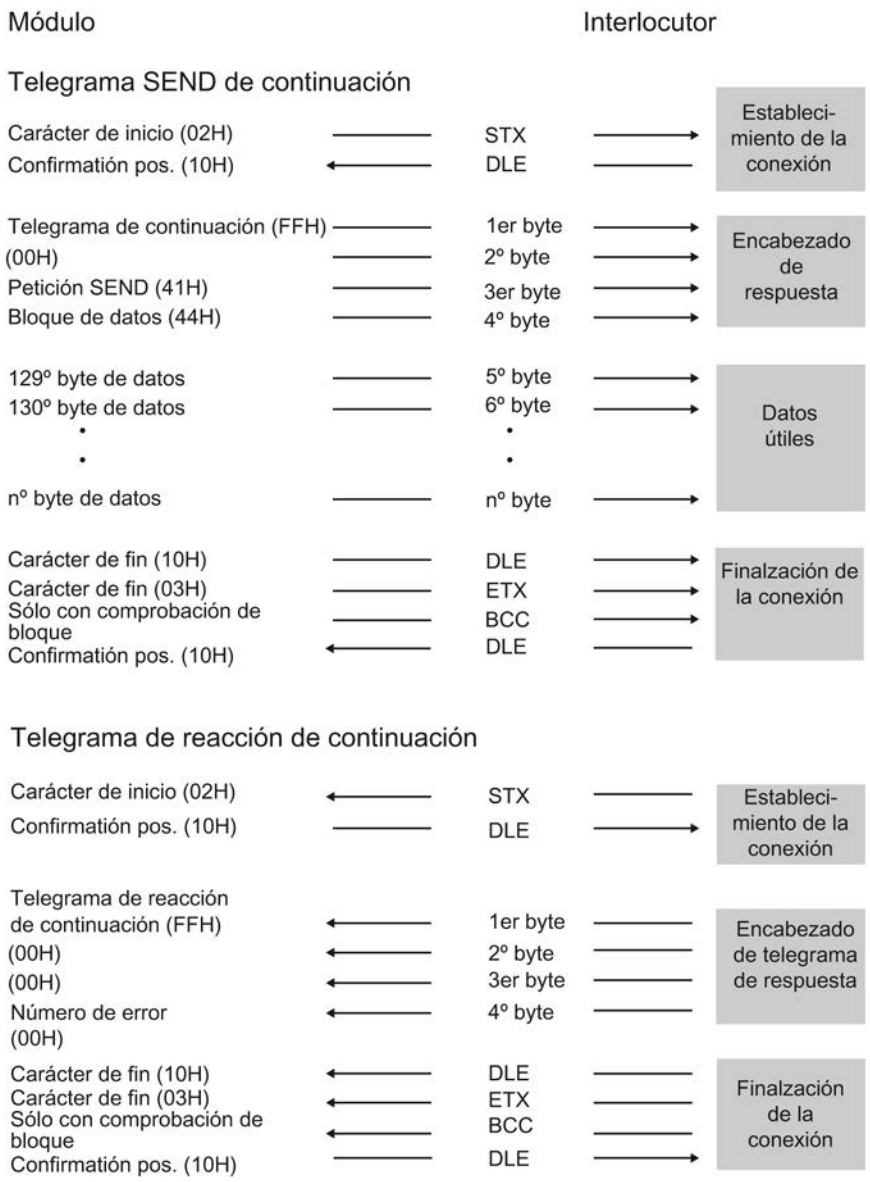


Figura 8-33 Desarrollo del telegrama SEND de continuación con un telegrama de reacción de continuación

Recoger datos con RK512

Fases de la recogida de datos con RK512

La siguiente figura representa el desarrollo del proceso de transmisión de datos durante la recogida con un telegrama de reacción con el protocolo para mainframe RK512.

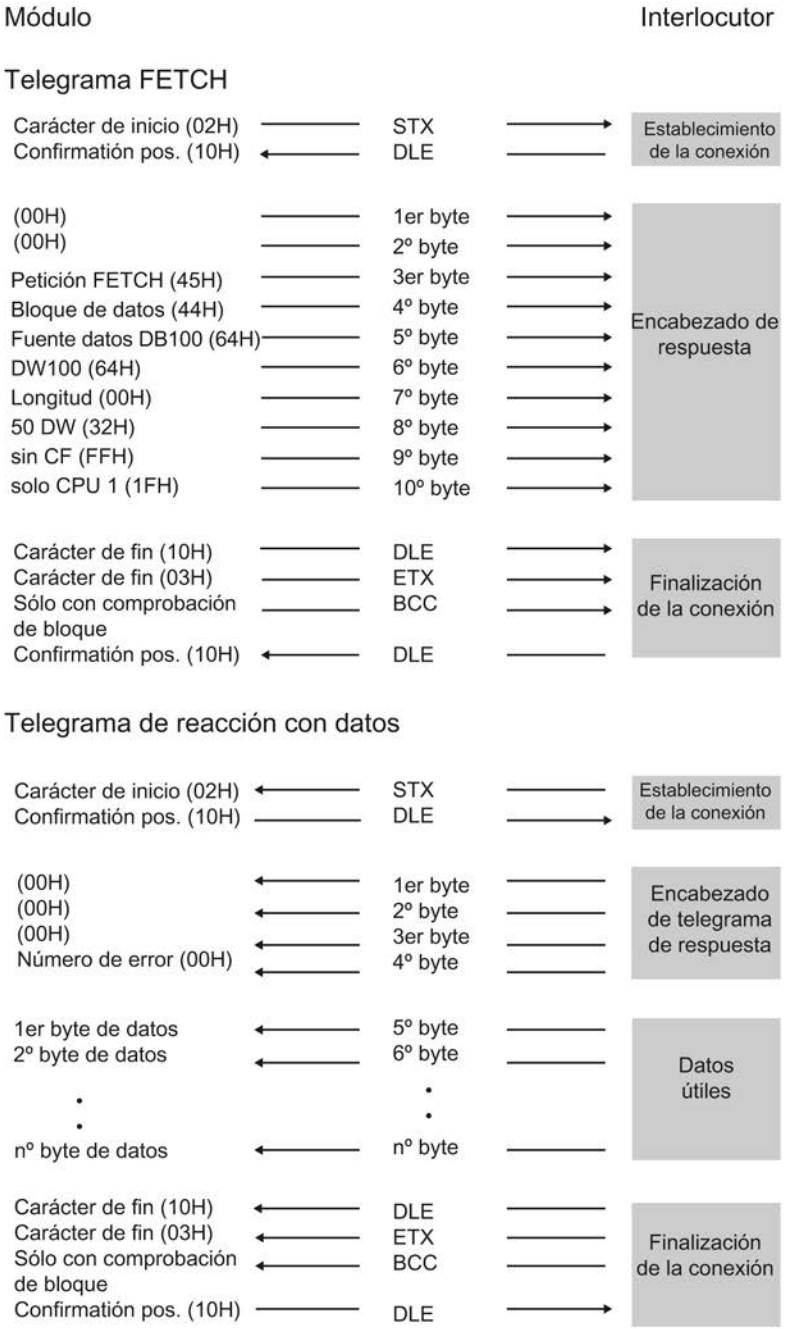


Figura 8-34 Intercambio de datos al recoger con telegrama de reacción

Recoger datos

La petición FETCH se desarrolla en el siguiente orden:

- Interlocutor activo
  Envía un telegrama FETCH. Éste contiene el encabezado.
- Interlocutor pasivo
  Recibe el telegrama, comprueba el encabezado, recoge los datos de la CPU y confirma la recepción con un telegrama de reacción. Éste contiene los datos.
- Interlocutor activo
  Recibe el telegrama de reacción, transfiere a la CPU los datos útiles recibidos en el telegrama de reacción y termina la petición FETCH.

Si el 4º byte lleva un número de error (no igual a 0), el telegrama de reacción no contiene datos.

Si se solicitan más de 128 bytes, éstos se transmiten automáticamente en uno o más telegramas de continuación. Cada telegrama FETCH de continuación se acusa con un telegrama de reacción de continuación (con más datos útiles).

Tras la recepción del último telegrama de reacción de continuación (con el último bloque de datos útiles) se transmite todo el conjunto de datos útiles a la CPU y se termina la petición FETCH.

Nota

Si la CPU no ha podido procesar correctamente el telegrama FETCH o si se ha producido un error en el encabezado del telegrama, el interlocutor inserta un número de error en el 4º byte del telegrama de reacción. Si se trata de fallos de protocolo no se produce ninguna entrada en el telegrama de reacción.

Telegrama FETCH de continuación

La siguiente figura representa el desarrollo del proceso de transmisión de datos al recoger datos con un telegrama de reacción de continuación.

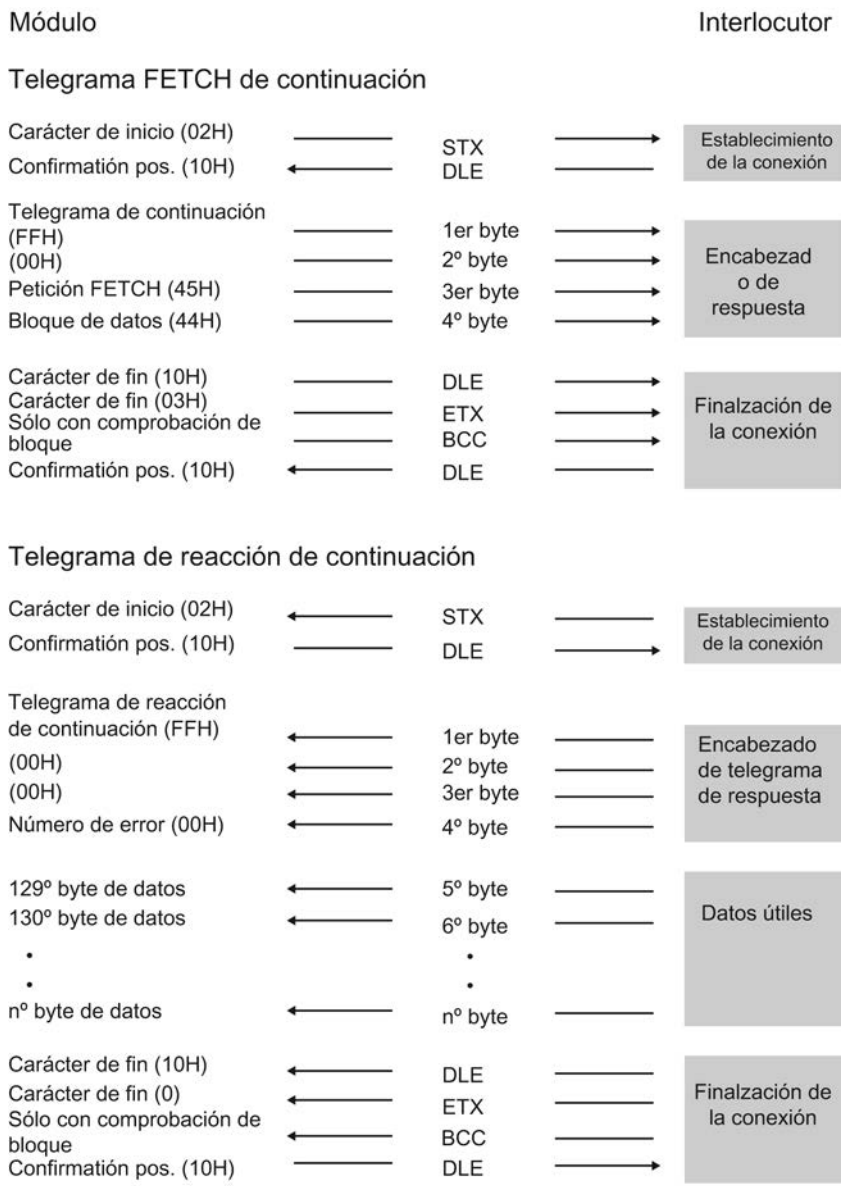


Figura 8-35 Desarrollo del telegrama FETCH de continuación con un telegrama de reacción de continuación

Modo casi dúplex

Casi dúplex significa: los interlocutores pueden enviar telegramas de orden y de respuesta en cualquier momento, menos cuando el otro interlocutor está enviando. La profundidad de anidado máximo de los telegramas de orden y de recepción es "1". Por ello, sólo es posible

procesar un nuevo telegrama de orden cuando el primero ha sido contestado con un telegrama de reacción.

En algunos casos - cuando ambos interlocutores quieren enviar - puede enviarse un telegrama SEND o FETCH del interlocutor antes del telegrama de reacción. Por ejemplo, cuando en el búfer de salida del módulo de comunicación se ha registrado un telegrama SEND o FETCH del interlocutor antes del telegrama de reacción.

En la siguiente figura el telegrama de reacción de continuación para el primer telegrama SEND se envía después del telegrama SEND del interlocutor .

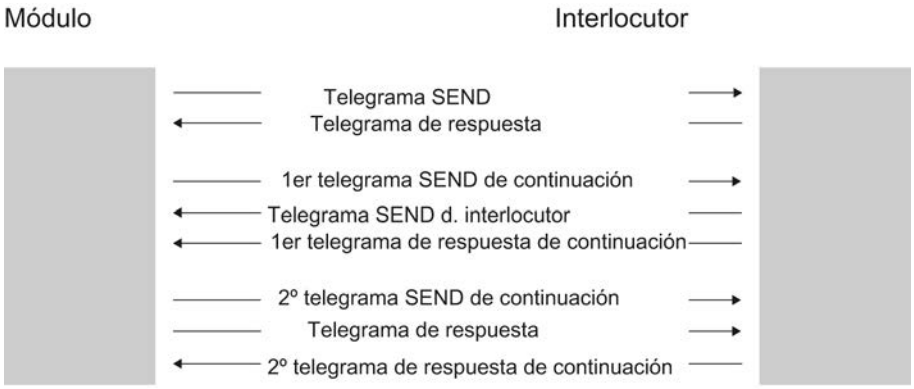


Figura 8-36 Modo casi dúplex

Peticiones RK512 de CPU

En la siguiente figura se representa el desarrollo del proceso de protocolo para mainframe RK512 mediante peticiones de CPU.

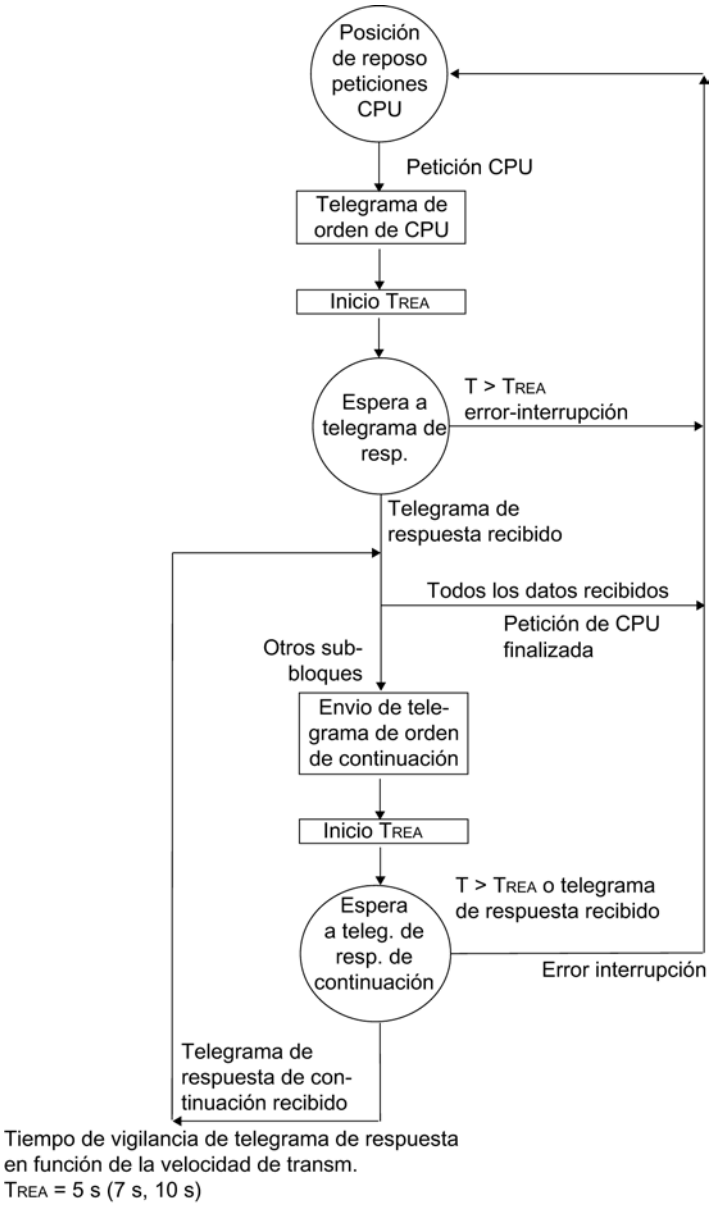
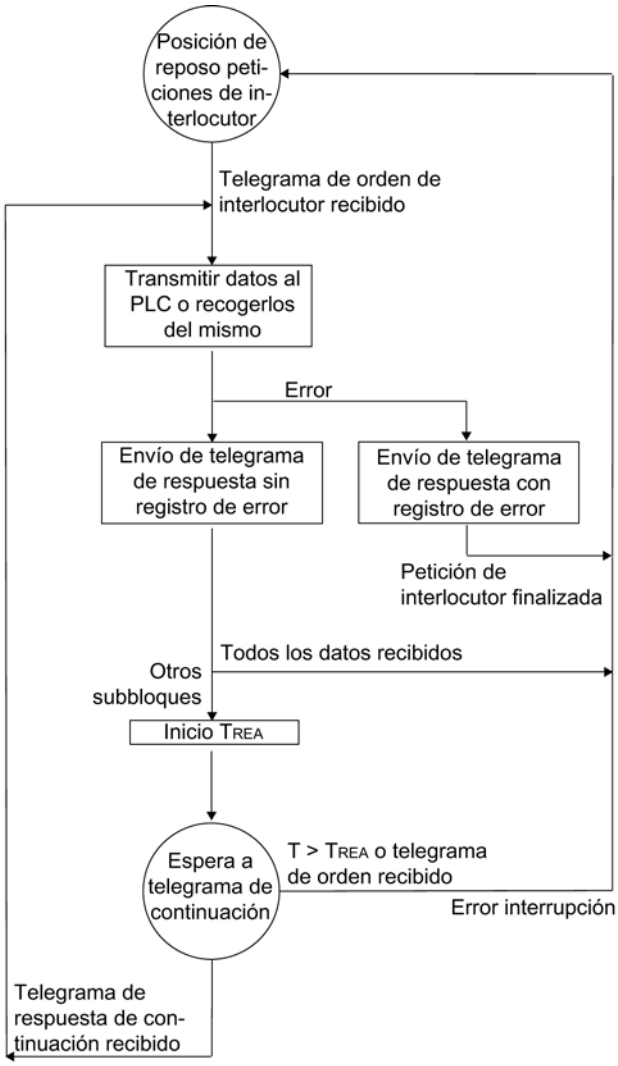


Figura 8-37 Fases de la transmisión de datos con RK512 mediante peticiones de CPU

Peticiones RK512 de interlocutor

En la siguiente figura se representa el desarrollo del proceso de protocolo para mainframe RK512 mediante peticiones de interlocutor.



Tiempo de vigilancia para telegrama de respuesta en función de la velocidad de transmisión
TREA = 5 s (7 s, 10 s)
PLC = autómata

Figura 8-38 Esquema de desarrollo de la transmisión de datos con RK512 mediante peticiones de interlocutor

Transmisión de datos con driver de impresora

Transmisión de datos con el driver de impresora

Para la asignación de los protocolos a los módulos de comunicación véase: Sinopsis de los componentes e interfaces (Página 1068)

Introducción

Con el driver de impresora se pueden imprimir textos de aviso con indicación de fecha y hora. De este modo es posible controlar procesos sencillos, imprimir avisos de error o alarmas o p.ej. dar instrucciones a los operarios.

El driver de impresora contiene el nivel de transmisión de bits (nivel 1).

Textos de aviso y parámetros para la impresión

En el cuadro de diálogo de propiedades del módulo se configuran los textos de aviso y se definen los parámetros (diseño de página, juego de caracteres, caracteres de control) para la impresión. Los textos de aviso y los parámetros para la impresión se transmiten al módulo de comunicación durante su arranque junto con los parámetros del módulo.

- Textos de aviso:
  Los textos de aviso se pueden configurar con variables e instrucciones de control (como p. ej. negrita, cursiva, subrayado, etc.). Cada texto de aviso recibe un número en la configuración. En los módulos CP 441, la impresión de un texto de aviso determinado se produce mediante la indicación de una referencia (en la célula de memoria que contiene el número del texto) en los parámetros de envío SD_1 a SD_4 de la instrucción. En los módulos CP 34x se define indicando el número de texto de aviso en una cadena de formato en la llamada de la instrucción.
  Los textos de aviso configurados pueden guardarse en un archivo de texto externo (nombre del SDB, máximo 8 caracteres ASCII) y modificarlos con el editor apropiado. De esta forma, por ejemplo, es posible traducir los textos y volver a leerlos.
  Estructura del archivo de textos de aviso:
  <Número de versión>
  <Número del aviso>:<Contenido del aviso>
- Diseño de página:
  Para el diseño de página es posible ajustar los márgenes, saltos de línea, así como los pies de página y los encabezamientos.
- Juego de caracteres:
  Con una tabla de conversión de caracteres se cambia del juego de caracteres ANSI de STEP 7 al juego de caracteres de impresión. Es posible hacer modificaciones en la tabla de conversión de caracteres propuesta para el tipo de impresión, y adoptar así p. ej. los caracteres específicos de un país.
- Caracteres de control:
  Una tabla de caracteres de control le permite modificar las instrucciones de control del texto de aviso para la emulación de impresora para activar y desactivar la letra negrita, cursiva, comprimida y expandida o el subrayado o incluir nuevos caracteres de control.

Variables

En el texto del aviso se pueden visualizar hasta 4 variables (3 + número de aviso). Los valores de las variables pueden transferirse de la CPU al modulo de comunicación. Como variables pueden visualizarse:

- Valores de cálculo del programa de usuario (p. ej.: niveles de llenado)
- Fecha y hora

- Cadenas (variable String)
- Otros textos de aviso

Para cada variable es necesario indicar en el texto de aviso configurado o en la cadena de formato una instrucción de conversión en la que esté codificado el significado y el formato de salida del valor de la variable.

Cadena de formato

Con la cadena de formato se define la representación y composición de un texto de aviso. La cadena de formato puede componerse de:

- Texto (todos los caracteres imprimibles, p. ej.: A las ... horas se ha alcanzado el nivel ... l)
- Las instrucciones de conversión para variables (p. ej. %N = puntero al número de texto de aviso x, donde x es un valor de variable (ver ejemplo 2 más abajo))
  Para cada variable debe haber exactamente una instrucción de conversión en la cadena de formato o en el texto de aviso configurado. Las instrucciones de conversión se aplican a las variables conforme a su orden.
- Las instrucciones de control con carácter de control para negrita, comprimida, expandida, cursiva y subrayado (p. ej. \B = activar negrita) o bien con caracteres de control definidos por el usuario

Se pueden utilizar caracteres de control adicionales registrándolos en el cuadro de diálogo de propiedades del módulo y reparametrizando el módulo de comunicación.

Otras funciones

Además de la impresón de textos de aviso, también se pueden utilizar las siguientes funciones de impresión. La ejecución de una de estas funciones también se indica en la cadena de formato.

- Ajustar el número de página (cadena de caracteres = %P)
- Iniciar página nueva (cadena de formato = \F)
- Imprimir con/sin salto de línea m(\x al final de la cadena de formato)

Recuerde que en la configuración estándar se realizar un salto de línea después de cada impresión.

Ejemplos

Ejemplo 1: A las "17:30" horas se ha alcanzado el nivel "200" l

Cadena de formato = A las %Z horas se ha alcanzado el nivel %i l.
Variable 1 = hora
Variable 2 = nivel

Ejemplo 2: La presión de la cámara "desciende"

Cadena de formato = %N %S
Variable 1 = 17 (texto de aviso n.º 17: La presión de la cámara ...)
Variable 2 = referencia a cadena (variable String: ... desciende)

Ejemplo 3: (Ajustar el número de página a 10)

Cadena de formato = %P
Variable 1 = 10 (número de página: 10)

Impresión

Para que pueda producirse la salida de n bytes de datos útiles a la impresora deben indicarse la cadena de formato y las variables del texto de aviso en la llamada de la instrucción en forma de parámetros.

Durante la impresión se preparan los datos que se van a imprimir. La impresión se procesa conforme a la parametrización del cuadro de diálogo de propiedades del módulo (diseño de página, juego de caracteres, caracteres de control, etc.).

En la salida por impresora no se reciben caracteres. Son excepción los caracteres de control para el control de flujo, si éstos se han parametrizado de la forma correspondiente. No se acepta ningún otro carácter que pueda aparecer.

Salida del texto de aviso

En la figura siguiente se representan los procesos de impresión.

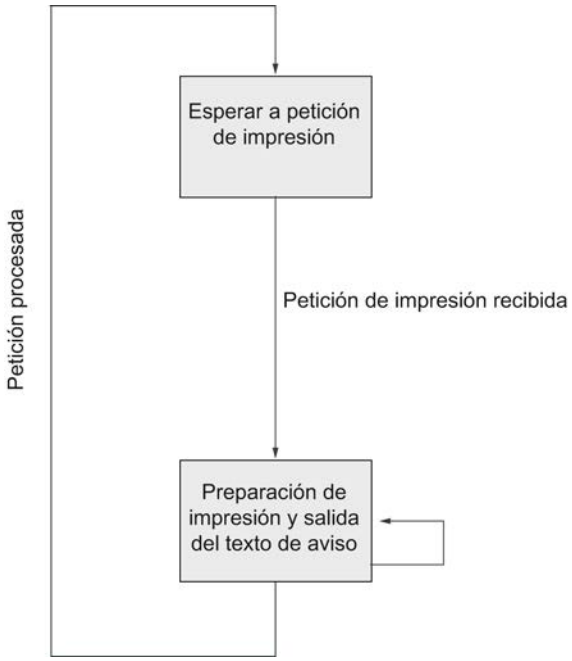


Figura 8-39 Esquema del proceso de impresión

Handshake de software/hardware

Los procedimientos handshake controlan el flujo de datos entre dos interlocutores. El procedimiento handschake evita que se pierdan datos durante la transmisión con equipos que trabajan a diferente velocidad.

En la impresión también es posible enviar los datos de salida con control de flujo de datos. Básicamente se distinguen dos procedimientos:

- Handshake de software (p. ej. XON/XOFF)
- Handshake de hardware (p. ej. RTS/CTS)

En la impresión, el control del flujo de datos en el módulo de comunicación se desarrolla de la siguiente manera:

- En cuanto el módulo de comunicación pasa por parametrización al modo de operación con control de flujo, envía el carácter XON o pone la línea RTS en estado ON.
- Si el módulo de comunicación recibe el carácter XOFF o si la señal de control CTS pasa a estado OFF, el módulo interrumpe la salida de caracteres. Si transcurrido un tiempo determinado parametrizable no se recibe ningún carácter XON o CTS no pasa a ON, la impresión se interrumpe y se registra el correspondiente mensaje de error (0708H) en la salida STATUS de la instrucción.

Nota

Si se parametriza con control de flujo RTS/CTS es necesario cablear por completo las señales de interfaz utilizadas en la conexión por conector.

Señal BUSY

El módulo de comunicación evalúa la señal de control "BUSY" de la impresora. Ésta notifica al CP 34x1 su disposición de recepción,

- En el caso del CP 34x-20mA-TTY: Con corriente en la línea RxD.
- En el caso del CP 34x–RS232C y el CP 34x-RS422/485: Con señal CTS = "ON".

Nota

Si se parametriza con control de flujo, en la impresora hay que ajustar la polaridad de la señal BUSY de la siguiente manera:

- Señal BUSY: CTS = "OFF"

Recuerde que algunas impresoras indican la señal BUSY con la señal DTR. En ese caso hay que realizar el cableado correspondiente en el cable de conexión que va al módulo de comunicación.

Detección del fin de un telegrama

Criterios de fin

En la transmisión de datos con el driver ASCII se puede elegir entre tres criterios de fin distintos para la recepción de datos:

- **Tiempo de retardo entre caracteres transcurrido**
- **Recepción del/de los carácter(es) de fin**
- **Recepción de un número de caracteres determinado**

Parámetros de protocolo

En la tabla siguiente encontrará una descripción de los parámetros de protocolo.

Tabla 8-75 Parámetros de protocolo (driver ASCII)

Parámetro	Descripción	Rango de valores	Valor predeterminado
Detección de fin de un telegrama de recepción	Determinación del criterio que determina el fin de telegramas.	• Transcurrido el tiempo de retardo de caracteres • Tras recibir el/los carácter(es) de fin • Tras recibir un número fijo de caracteres	Transcurrido el tiempo de retardo de caracteres
Carácter de fin 1 [1]	Código de la primera identificación de fin.	• Con 7 bits de datos: 0 a 7FH (Hex) [2] • Con 8 bits de datos: 0 a FFH (Hex) [2]	3 (03H = ETX)
Carácter de fin 2 [1]	Código de la segunda identificación de fin, si está activada.	• Con 7 bits de datos: 0 a 7FH (Hex) [2] • Con 8 bits de datos: 0 a FFH (Hex) [2]	0
[1] Sólo ajustable con el criterio de fin Carácter final. [2] Dependiendo de si para la trama de caracteres se parametrizan 7 u 8 bits de datos.			

Información adicional

Encontrará una descripción detallada de la recepción de datos con el driver ASCII en el apartado Recepción de datos con el driver ASCII (Página 1083).

Control del flujo de datos y señales cualificadoras

Introducción

El módulo de comunicación le permite crear un acoplamiento punto a punto con módulos SIMATIC y con productos externos.

Funciones compatibles con las interfaces

Dependiendo de la interfaz utilizada, pueden utilizarse diferentes funciones de los drivers:

Tabla 8-76 Funciones del módulo de comunicación dependiendo de la interfaz utilizada

	RS232C	20mA TTY	X27 (RS422/485)	
			RS422*	RS485*
Función CP 441				
Procedimiento 3964(R)	●	●	●	-
Protocolo para mainframe RK512	●	●	●	-
Driver ASCII:	●	●	●	●

	RS232C	20mA TTY	X27 (RS422/485)	
			RS422*	RS485*
• Manejo de las señales cualificadoras RS 232C	●	-	-	-
• Control/lectura de las señales cualificadoras RS 232C con instrucciones	●	-	-	-
• Control del flujo de datos con RTS/CTS	●	-	-	-
• Control del flujo de datos con XON/XOFF	●	●	●	-
Driver de impresora:	●	●	●	●
• Control del flujo de datos con RTS/CTS	●	-	-	-
• Control del flujo de datos con XON/XOFF	●	●	●	-
Función CP 440				
Procedimiento 3964(R)			●	-
Driver ASCII:			●	●
• Control del flujo de datos con XON/XOFF			●	-
Función CP 341				
Procedimiento 3964(R)	●	●	●	-
Protocolo para mainframe RK512	●	●	●	-
Driver ASCII:	●	●	●	●
• Manejo de las señales cualificadoras RS 232C	●	-	-	-
• Control/lectura de las señales cualificadoras RS 232C con instrucciones	●	-	-	-
• Control del flujo de datos con RTS/CTS	●	-	-	-
• Control del flujo de datos con XON/XOFF	●	●	●	-
Driver de impresora:	●	●	●	●
• Control del flujo de datos con RTS/CTS	●	-	-	-
• Control del flujo de datos con XON/XOFF	●	●	●	-
Función CP 340				
Procedimiento 3964(R)	●	●	●	-
Driver ASCII:	●	●	●	●
• Manejo de las señales cualificadoras RS 232C	●	-	-	-
• Control/lectura de las señales cualificadoras RS 232C con instrucciones	●	-	-	-
• Control del flujo de datos con RTS/CTS	●	-	-	-
• Control del flujo de datos con XON/XOFF	●	●	●	-
Driver de impresora:	●	●	●	●
• Control del flujo de datos con RTS/CTS	●	-	-	-
• Control del flujo de datos con XON/XOFF	●	●	●	-

Función ET 200S 1SI				
Procedimiento 3964(R)	●		●	-
Driver ASCII:	●		●	●
• Manejo de las señales cualificadoras RS 232C	●		-	-
• Control/lectura de las señales cualificadoras RS 232C con instrucciones	●		-	-

• Control del flujo de datos con RTS/CTS	•		-	-
• Control del flujo de datos con XON/XOFF	•		•	-
* La diferenciación entre RS422 y RS485 se consigue mediante parametrización.				

Aplicaciones posibles de los módulos de comunicación

Dependiendo del módulo de comunicación utilizado puede crearse un acoplamiento punto a punto con diferentes módulos Siemens, así como con productos externos:

- SIMATIC S5 vía driver 3964(R) o RK512 con el correspondiente módulo de interfaz en el lado de S5
- Terminales Siemens BDE de la familia ES 2 vía driver 3964(R)
- MOBY I (ASM 420/421, SIM), MOBY L (ASM 520) y estación de registro ES 030K vía driver 3964R
- PC mediante procedimiento 3964(R) (existen herramientas de desarrollo para programación en PC: PRODAVE DOS 64R (6ES5 897-2UD11) para MS-DOS, PRODAVE WIN 64R
  (6ES5 897-2VD01) para Windows o el driver ASCII)
- Lector de código de barras vía driver 3964(R) o driver ASCII
- PLC de otro fabricante vía 3964(R), driver ASCII o RK512
- Otros aparatos con estructuras de protocolo sencillas mediante la correspondiente adaptación del protocolo con el driver ASCII
- Otros equipos que también dispongan del driver 3964(R) o RK512
- Impresora (HP-Deskjet, HP-Laserjet, Postscript, Epson, IBM)

Modos de operación de la interfaz

Posibles modos de operación de la interfaz

En la siguiente tabla se muestran los modos de operación de la interfaz para módulos de comunicación y protocolos en general.

Modo de operación	Descripción
Semidúplex (RS485) a 2 hilos	Modo de operación para acoplamiento punto a punto o acoplamiento multipunto (Multipoint) en modo a dos hilos. El módulo de comunicación puede ser tanto maestro como esclavo. En el modo de operación RS485, la transmisión de datos se realiza a través de dos líneas (modo a dos hilos). Las dos líneas están disponibles para el envío y la recepción variando de dirección. Podrá bien enviarse o bien recibirse (semidúplex). Tras un proceso de envío se conmuta inmediatamente al proceso de recepción (tiempo máximo de conmutación de 1 ms).
Dúplex (RS422) a 4 hilos	En el modo de operación RS422, la transmisión de datos se realiza a través de cuatro líneas (modo a cuatro hilos). Dos de las líneas estarán disponibles para el envío y las otras dos para la recepción. Podrá enviarse y recibirse al mismo tiempo (dúplex). Los datos se intercambian simultáneamente entre uno o varios interlocutores; en un determinado punto puede tanto enviarse como recibirse. Cada interlocutor deberá poder utilizar simultáneamente una dirección de envío y recepción.
Dúplex (RS422) modo a 4 hilos (acoplamiento punto a punto) [1)]	El módulo de comunicación puede emplearse en el modo de operación RS422 en las siguientes topologías: • Acoplamiento entre dos estaciones: acoplamiento Punto a punto • Acoplamiento entre varias estaciones: acoplamiento multipunto El módulo de comunicación puede emplearse como maestro y como esclavo.

Modo de operación	Descripción
Dúplex (RS422) modo a 4 hilos (maestro multipunto) [1]	El módulo de comunicación puede emplearse en el modo de operación RS422 en las siguientes topologías: • Acoplamiento entre dos estaciones: Acoplamiento punto a punto • Acoplamiento entre varias estaciones: Acoplamiento multipunto El módulo de comunicación puede emplearse como maestro multipunto y como esclavo multipunto. En una topología maestro/esclavo en modo RS422 se aplica que: • El emisor del maestro se interconecta con los receptores de todos los esclavos. • Los emisores de los esclavos se interconectan con el receptor del maestro. • La asignación previa sólo se produce en el receptor del maestro y en el receptor de un esclavo. Todos los demás esclavos trabajan sin asignación previa.
Dúplex (RS422) modo a 4 hilos (maestro multipunto) [1]	El módulo de comunicación puede emplearse en el modo de operación RS422 en las siguientes topologías: • Acoplamiento entre dos estaciones: Acoplamiento punto a punto • Acoplamiento entre varias estaciones: Acoplamiento multipunto El módulo de comunicación puede emplearse como maestro multipunto y como esclavo multipunto. En una topología maestro/esclavo en modo RS422 se aplica que: • El emisor del maestro se interconecta con los receptores de todos los esclavos. • Los emisores de los esclavos se interconectan con el receptor del maestro. • La asignación previa sólo se produce en el receptor del maestro y en el receptor de un esclavo. Todos los demás esclavos trabajan sin asignación previa.

[1] sólo CP 440

Ocupación estándar de la línea de recepción

Interfaz X27 (RS422/485)

En la siguiente tabla se describen los parámetros del submódulo interfaz X27 (RS422/485). No puede emplearse el modo RS485 con los procedimientos 3964(R) y RK512 y la impresora.

Tabla 8-77 Módulo interfaz X27 (RS422/485)

Parámetros	Descripción	Rango de valores	Valor por defecto
Ocupación estándar de la línea de recepción	**ninguno**: este ajuste sólo tiene sentido para drivers especiales con capacidad para bus.	ninguno	Señal R(A) 5 V, señal R(B) 0 V (detección de rotura) [2]
	Señal R(A) 5V, señal R(B) 0V (detección de rotura): Con estos ajustes predeterminados, en el modo "Dúplex (RS 422) modo a 4 hilos" es posible la detección de rotura.	Señal R(A) 5 V, señal R(B) 0 V (detección de rotura) [1]	

Parámetros	Descripción	Rango de valores	Valor por defecto
	Señal R(A) 0V, señal R(B) 5V: Este estado inicial corresponde al estado de reposo (no hay emisor activo) en el modo "semidúplex (RS 485) dos cables". Con este ajuste predeterminado no es posible la detección de rotura.	Señal R(A) 0V, señal R(B) 5V	
(1) Sólo en "Dúplex (RS422) modo a cuatro hilos" (2) Sólo en "Dúplex (RS422) modo a cuatro hilos"; en "Semidúplex (RS485) modo a dos hilos" la configuración predeterminada es R(A) 0V, señal R(B) 5V			

Ocupación estándar de la línea de recepción

La figura muestra el cableado del receptor en la interfaz X27 (RS422/485):

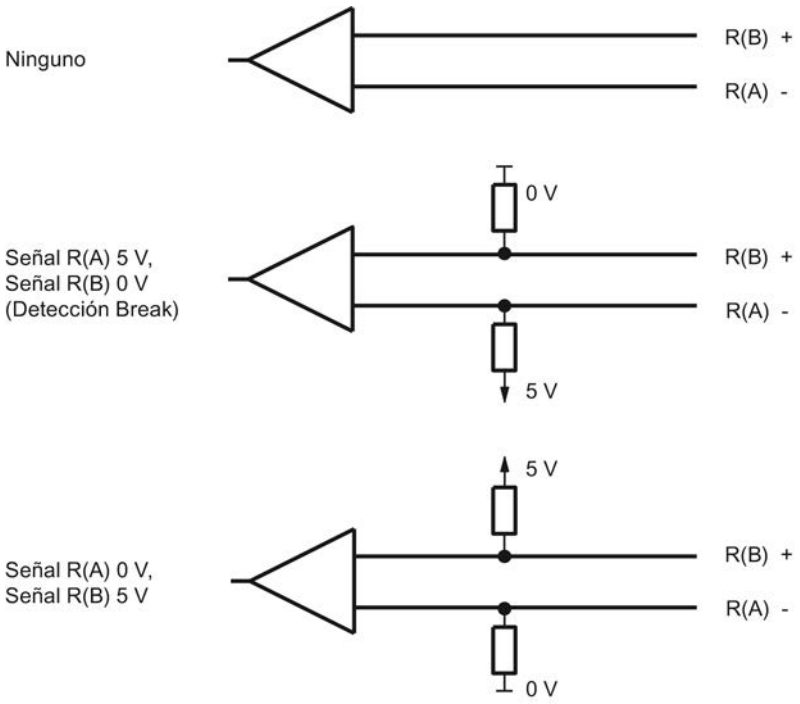


Figura 8-40 Cableado del receptor en la interfaz X27 (RS 422/485)

CP 440 en una topología multipunto con los protocolos RS422/485

Posibles aplicaciones

En los modos de operación RS422 y RS485, el CP 440 puede emplearse en diferentes topologías.

Se distingue entre acoplamientos con

- dos estaciones (**punto a punto**) y
- muchas estaciones (**multipunto/multipoint**).

El CP 440 puede emplearse como

- **maestro** o
- **esclavo**.

Modo RS422 o RS485 con conexiones punto a punto

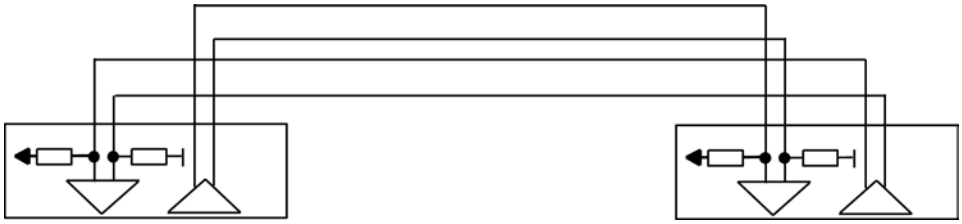


Figura 8-41 RS422 punto a punto

Figura 8-42 RS485 punto a punto

En una **topología maestro/esclavo**, en el programa de usuario debe suprimirse el telegrama correspondiente. Ejemplo: el maestro envía a todos los esclavos un telegrama con una información de dirección. Todos los esclavos los reciben y comparan la dirección con la suya propia. Si coinciden, el esclavo invocado envía su respuesta.

Los emisores de todos los esclavos deben ser capaces de activarse con alta resistencia óhmica.

Modo RS422 con conexiones multipunto

En una topología maestro/esclavo en modo RS422

- el emisor del maestro se interconecta con los receptores de todos los esclavos,
- los emisores de los esclavos se interconectan con el receptor del maestro,
- la asignación previa sólo se produce en el receptor del maestro y en el receptor de un esclavo. Todos los demás esclavos trabajan sin asignación previa.

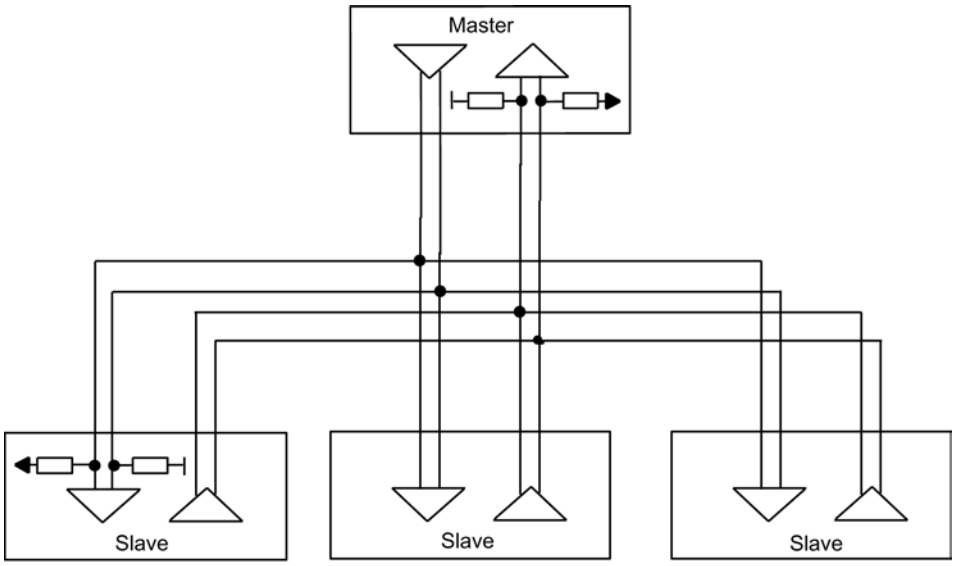


Figura 8-43 RS422 multipunto (multipoint)

Modo RS485 con conexiones multipunto

En una topología en modo RS485

- se interconectan la pareja de cables para envío/recepción de todas las estaciones,
- la asignación previa sólo se realiza en el receptor de una estación. Todos los demás módulos trabajan sin asignación previa.

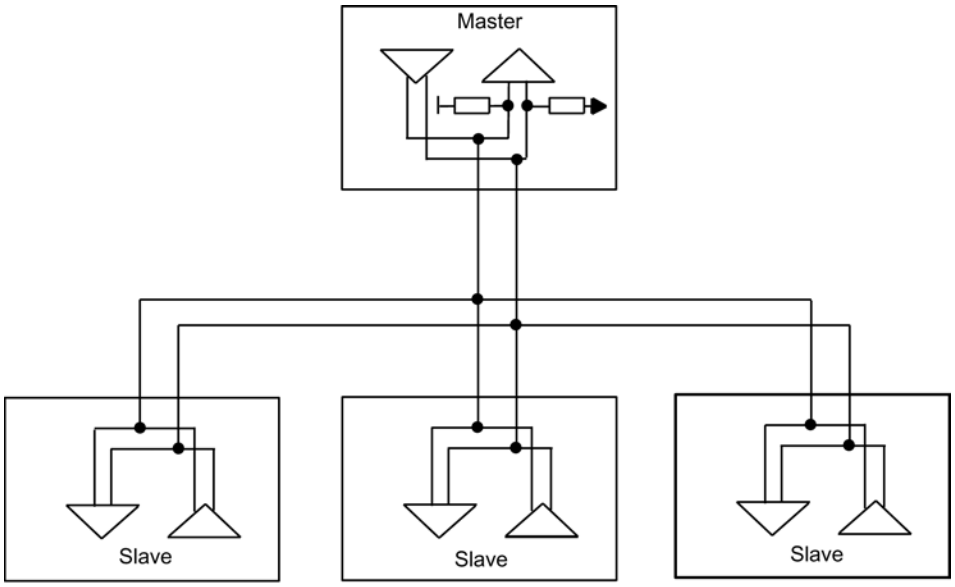


Figura 8-44 RS485 multipunto (multipoint)

Los ajustes necesarios para las diferentes topologías se realizan en el cuadro de diálogo de propiedades, en la máscara "Interfaz".

Nota

Si utiliza el driver ASCII en modo RS422 multipunto o en modo RS485, en el programa de usuario debe asegurarse de que nunca pueda enviar más de una estación. Si envían más de una simultáneamente, el telegrama se falsifica.

Búfer de recepción

Búfer de recepción del módulo

La tabla siguiente describe los parámetros del búfer de recepción del módulo.

Tabla 8-78 Búfer de recepción del módulo - Procedimiento 3964(R) / RK512 / driver ASCII

Parámetros	Descripción	Rango de valores	Valor por defecto
Borrar búfer de recepción del módulo en el arranque	El búfer de recepción del módulo no se borra durante el arranque de la CPU (cambio STOP RUN).	no (no modificable)	no
Utilizar un buzón de CPU (sólo CP 441)	Puede indicar si desea configurar un buzón en la CPU. Se debe configurar un buzón de recepción si en el programa de usuario de la CPU para el módulo no se ha programado ninguna instrucción BRCV. Si ha programado un BRCV, debe desactivar este parámetro, ya que si no lo hace los datos no serán procesados por el BRCV, sino que se recogerán en el buzón.	• sí • no	no
Número DB (1)	Número del bloque de datos para el buzón en la CPU	1 a 65535 (según la CPU)	1
(1) Sólo con "Utilizar buzón de CPU" = "sí". (2) Sólo si "Telegramas de recepción respaldados" = "1".			

Tabla 8-79 Búfer de recepción del módulo - adicionalmente en el driver ASCII

Parámetros	Descripción	Rango de valores	Valor por defecto
Telegramas de recepción respaldados	Se puede indicar el número de telegramas de recepción que se deben respaldar en el búfer de recepción del modulo de comunicación. Si indica el valor "1", y desactiva el siguiente parámetro "Impedir sobrescritura" y lee los datos recibidos cíclicamente en el programa de usuario, siempre se enviará un telegrama actual a la CPU.	1 a 250	250
Telegramas dinámicos ET 200S 1SI	En la recepción de avisos puede establecer si deberá respaldarse un único aviso o si, por el contrario, los avisos deben respaldarse de forma dinámica. Al activar los telegramas dinámicos, el módulo puede respaldar más avisos con distintas longitudes. El búfer en cuestión es un búfer en anillo. Si éste estuviera lleno, se sobrescribiría el aviso más antiguo, a no ser que estuviese activado el parámetro "Evitar sobrescritura de búfer". En este caso, se rechazará el nuevo aviso. En ambos casos una alarma de diagnóstico será la encargada de indicar que se han perdido datos.	• Activado • Bloqueado	Activado
Impedir sobrescritura	Si para el parámetro "Telegramas de recepción respaldados" indica el valor "1", puede desactivar este parámetro. De esta manera permite que se sobrescriba el telegrama de recepción respaldado.	• sí • no (2)	sí
(1) Sólo con "Utilizar buzón de CPU" = "sí". (2) Sólo si "Telegramas de recepción respaldados" = "1".			

Comunicación Modbus

Resumen de la comunicación Modbus

Clasificación dentro del entorno del sistema

La siguiente descripción de Modbus se refiere al uso del driver correspondiente cargable posteriormente en los procesadores de comunicaciones CP 341 (S7 300) y CP 441-2 (S7 400), así como al uso del módulo ET 200S 1SI en la variante Modbus/USS.

Función del acoplamiento

Con los módulos de comunicación debidamente equipados y las correspondientes instrucciones se puede establecer un enlace entre un sistema de control Modbus remoto (p. ej. autómatas Modicon o Honeywell TDC3000) y una SIMATIC S7.

Para la transmisión se utiliza el protocolo GOULD - MODBUS en el formato RTU.

En la comunicación entre un módulo de comunicación utilizado como esclavo Modbus y un sistema maestro se pueden utilizar los códigos de función 01, 02, 03, 04, 05, 06, 08, 15 y 16.

Si un módulo de comunicación SIMATIC S7 se utiliza como maestro Modbus, además están disponibles los siguientes códigos de función 07, 11 y 12.

Protocolo Modbus

El protocolo Modbus es un protocolo de comunicación basado en una arquitectura maestro/esclavo o cliente/servidor.

Se utiliza un procedimiento semidúplex asíncrono y con trasparencia de códigos. La transmisión de datos se realiza sin procedimientos Handshake.

SIMATIC S7 como esclavo Modbus

El maestro tiene la iniciativa en la transmisión, el CP/la CPU S7 funciona como esclavo.

No es posible el intercambio de telegramas entre esclavos.

SIMATIC S7 como esclavo Modbus

Como maestro, el módulo de comunicación inicia la transmisión y, tras la emisión del telegrama de solicitud, espera el telegrama de respuesta del esclavo durante el tiempo de vigilancia de respuesta parametrizado.

Estructura del telegrama

El intercambio de datos "maestro-esclavo" y "esclavo-maestro" comienza con la **dirección de esclavo**, seguida del **código de función**. A continuación se transfieren los datos. La estructura del campo de datos depende del código de función utilizado. Al final del telegrama se transmitirá la comprobación CRC (CRC-Check).

ADDRESS	FUNCTION	DATA	CRC-CHECK
Bytes	Bytes	n bytes	2 bytes

ADDRESS	Dirección del esclavo MODBUS
FUNCTION	Código de función MODBUS
DATA	Datos del telegrama: Byte_Count, Coil_Number, Data
CRC-CHECK	Suma de verificación del telegrama

Dirección del esclavo

La dirección del esclavo puede estar en el área de 1 a 255. Con la dirección se llega a un esclavo concreto del bus.

Mensaje Broadcast

Con la dirección de esclavo cero, el maestro se dirige a todos los esclavos del bus.

Los **mensajes Broadcast** se permiten únicamente en conexión con los **códigos de función 05, 06, 15 ó 16** de escritura.

En un mensaje Broadcast, el esclavo no envía ningún telegrama de respuesta.

Código de función

El código de función define el significado del telegrama. Mediante el código de función se define también la estructura del telegrama. El módulo de comunicación soporta los siguientes códigos de función:

Código de función	Función según la especificación MODBUS
01	Read Coil Status
02	Read Input Status
03	Read Holding Registers
04	Read Input Registers
05	Force Single Coil
06	Preset Single Register
07	Read Exception Status (sólo maestro)
08	Loop Back Test
11	Fetch Communications Event Counter (sólo maestro)
12	Fetch Communications Event Log (sólo maestro)
15	Force Multiple Coils
16	Preset Multiple Registers

Campo de datos DATA

En el campo de datos DATA se transfieren los datos específicos del código de función:

- Bytecount, Coil_Startaddress, Register_Startaddress; Number_of_Coils, Number_of_Registers, ...
  Véase el capítulo "Códigos de función (Página 1143)".

CRC-Check

El final del telegrama forma la suma de verificación CRC-16 de dos bytes de longitud, formada según el polinomio $x^{16} + x^{15} + x^2 + 1$.

En primer lugar se transfiere el byte Low, y después, el byte High.

Fin del telegrama

El driver cargable detecta el fin del telegrama cuando no se realiza ninguna transferencia en el tiempo que tardan en transferirse 3 caracteres y medio (3,5 veces el tiempo de retardo de caracteres (véase MODBUS Protocol Reference Guide).

Así, este TIME_OUT de fin de telegrama depende de la velocidad de transmisión.

Velocidad de transmisión	TIME_OUT
115200 bits/s	0,3 ms
76800 bits/s	0,5 ms
57600 bits/s	0,7 ms
38400 bits/s	1 ms

Velocidad de transmisión	TIME_OUT
19200 bits/s	2 ms
9600 bits/s	4 ms
4800 bits/s	8 ms
2400 bits/s	16 ms
1200 bits/s	32 ms
600 bits/s	64 ms
300 bits/s	128 ms

En el "Funcionamiento normal" se evalúa y comprueba de manera formal el fin de telegrama Modbus recibido por el interlocutor acoplado tras el fin de telegrama TIME_OUT.

En "Supresión de perturbaciones", el fin de telegrama se detecta gracias un telegrama de recepción correcto y formal con un código CRC igualmente correcto.

Exception Responses

Cuando el esclavo detecta un error (p. ej., una dirección de registro no permitida) en el telegrama de solicitud del maestro, establece el bit de mayor valor en el código de función del telegrama de respuesta.

A continuación se transfiere un byte de código de error (Exception Code) que describe la causa de dicho error.

El significado de dicho parámetro se describe en el "GOULD MODICON Modbus Protocol".

Telegrama de Exception Code

La estructura del telegrama de respuesta del código de error del esclavo es la siguiente:

- p. ej., dirección de esclavo 5, código de función 5, Exception Code 2

Telegrama de respuesta del esclavo EXCEPTION_CODE_xx:

05H	Dirección del esclavo
85H	Código de función
02H	Exception Code (1...7)
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cuando el driver recibe un telegrama de respuesta del código de error, la petición en curso finalizará con error.

Además, en el área de registro de errores se registra el número de error correspondiente al código de error recibido (Exception Code 1-7).

No se produce ninguna entrada en el bloque de datos de destino.

Los siguientes códigos de error están definidos según la especificación MODBUS:

Código de error	Significado de la especificación MODBUS	Causa - Resumen técnico *
1	Illegal function	Código de función no permitido
2	Illegal data address	Dirección de datos no permitida en el esclavo
3	Illegal data value	Valor de datos no permitido en el esclavo
4	Failure in associated device	Error interno en el esclavo
5	Acknowledge	La función será ejecutada
6	Busy, Rejected message	El esclavo no está preparado
7	Negative acknowledgement	Función no ejecutable
* Buscar causa detallada en el esclavo.		

Señales cualificadoras RS232C

Las siguientes señales cualificadoras RS232C estarán disponibles en el módulo de comunicación al utilizar el módulo de interfaz RS232C:

DCD	(Entrada)	Data carrier detect	Soporte de datos detectado
DTR	(Salida)	Data terminal ready	Interlocutor listo para funcionar
DSR	(Entrada)	Data set ready	Interlocutor listo para funcionar
RTS	(Salida)	Request to send	Módulo de comunicación preparado para enviar
CTS	(Entrada)	Clear to send	El interlocutor puede recibir datos del módulo de comunicación (respuesta a RTS = ON del CP)
RI	(Entrada)	Ring indicator	Indicador de llamada

Tras la conexión del módulo de comunicación las señales de salida están en estado OFF (inactivas).

El procesamiento de las señales de control DTR/DSR y RTS/CTS puede parametrizarse en el cuadro de diálogo de propiedades del módulo o controlarse mediante las instrucciones del programa de usuario.

Manejo de señales cualificadoras RS232C

Las señales cualificadoras RS232C pueden manejarse de la siguiente manera:

- En la parametrización del manejo automático de todas las señales cualificadoras de RS232C
- Mediante las instrucciones correspondientes. Encontrará más información al respecto en el apartado "Resumen de instrucciones (Página 1070)".

Nota

En caso de parametrización de un manejo automático de las señales cualificadoras RS232C no es posible controlar RTS y DTR con las instrucciones correspondientes (véase el apartado "Resumen de instrucciones (Página 1070)").

Sin embargo, la lectura de todas las señales cualificadoras RS232C con la instrucción correspondiente es posible en todos los casos.

A continuación se explica el procedimiento básico para controlar y evaluar las señales cualificadoras de RS232C.

Manejo automático de las señales cualificadoras

El manejo automático de las señales cualificadoras RS232C en el módulo de comunicación se establece de la siguiente manera:

- Cuando el módulo de comunicación entra por parametrización en un modo de operación con manejo automático de las señales cualificadoras de RS232C, pone las líneas RTS en estado OFF y las DTR en estado ON (módulo de comunicación preparado para entrar en funcionamiento).
- Sólo es posible enviar y recibir telegramas tras poner la línea DTR a ON. Mientras DTR permanezca en OFF, no se pueden recibir datos vía la interfaz del RS232C. Cualquier petición de envío se interrumpe con el correspondiente mensaje de error.
- Si está pendiente una petición de envío, RTS pasa al estado ON y se inicia el tiempo de espera de salida de datos parametrizado. Una vez transcurrido el tiempo de salida de datos y con CTS = ON, los datos se envían a través de la interfaz RS232C.
- Si durante el envío la línea CTS no pasa a ON durante el tiempo de espera o si, durante el proceso de salida, CTS cambia al estado OFF, se interrumpirá el envío y se mostrará el aviso correspondiente.
- Tras la emisión de los datos y una vez transcurrido el tiempo de RTS a OFF parametrizado, la línea RTS pasa al estado de inactividad. No se espera a que la línea CTS pase al estado OFF.
- Es posible recibir datos vía la interfaz RS232C tan pronto como se active la línea DSR (ON). Si existe riesgo de sobrecarga del búfer de recepción del módulo de comunicación, éste no reacciona.
- Al cambiar de DSR = ON a OFF se interrumpe tanto el envío en curso como la recepción de datos con un aviso de error. En el búfer de diagnóstico del módulo de comunicación se registra el aviso "DSR = OFF" (manejo automático de las señales V24).

Nota

En caso de parametrización de un manejo automático de las señales cualificadoras RS232C no es posible controlar RTS y DTR con las instrucciones correspondientes. Encontrará más información al respecto en el apartado "Resumen de instrucciones (Página 1070)".

Nota

Defina el "tiempo de RTS a OFF" en el cuadro de diálogo de propiedades del módulo de tal manera que el CP pueda recibir completos los últimos caracteres del telegrama antes de que la línea RTS, y con ella la petición de envío, se desactiven. Defina el "tiempo de salida de datos" de tal manera que el interlocutor pueda pasar a modo de disponibilidad de recepción antes de que el tiempo haya transcurrido.

Cronograma

La siguiente figura muestra el desarrollo temporal de una petición de envío:

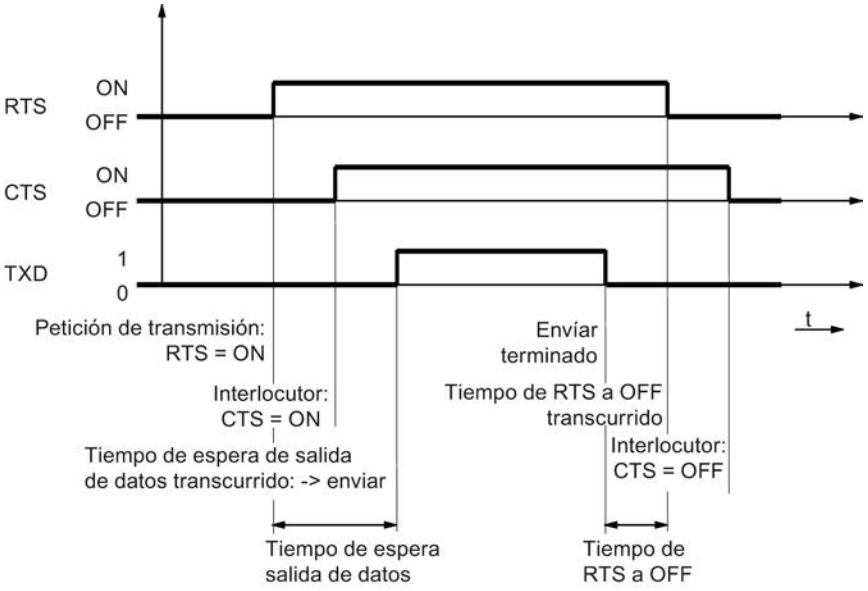


Figura 8-45 Cronograma para el manejo automático de las señales cualificadoras RS232C

Maestro Modbus

CP 341

Transferencia de datos de la CPU al módulo de comunicación con P_SND_RK (CP 341)

Activación

La ejecución de un código de función MODBUS se activa con un SFB **P_SND_RK** con un **flanco** en la entrada **REQ**.

Introduzca "S" en el parámetro SF para SEND.

La dirección lógica del módulo se introduce en LADDR.

Como tipo de área de la CPU del interlocutor debe introducirse "X" para el bloque de datos ampliado. No hay que introducir ningún valor para el resto de los parámetros de la CPU del interlocutor (R_...).

De esta forma se transfieren al driver los parámetros necesarios para la ejecución del código de función.

Fuente de datos

Si se activa el P_SND_RK, se transferirá al módulo de comunicación el **área de datos fuente** indicada con los parámetros **DB_NO** y **DBB_NO** y la longitud **LEN**.

Indicación de la longitud

La longitud **LEN** depende del código de función utilizado.

Código de función	Longitud LEN en bytes
01	6
02	6
03	6
04	6
05	6
06	6
07	2
08	6
11	2
12	2
15	>6
16	>6

Si en los códigos de función no se transfieren las cantidades de datos indicadas, no se realizará la petición y el P_SND_RK terminará con un flanco en la salida ERROR.

DB de origen SEND

En el área de datos fuente deben incluirse como datos útiles los parámetros necesarios para la ejecución de un código de función.

La estructura detallada del correspondiente DB de origen P_SND_RK se incluye en la descripción de los códigos de función individuales en el capítulo "Códigos de función (Página 1143)".

Generación de telegrama

Los telegramas de petición se envían al esclavo mediante los datos de origen P_SND_RK transferidos y estructurados por el módulo de comunicación.

A continuación, el driver comprueba que la longitud LEN indicada en el P_SND_RK se corresponda con la longitud prevista para este código de función.

Si no es así, la petición no se ejecuta y termina con un flanco en la salida ERROR del P_SND_RK.

Si se utilizan códigos de función distintos a los listados, la petición activada no se ejecuta y termina con ERROR en el P_SND_RK.

El módulo de comunicación forma los elementos "Contador de bytes" y "Comprobación CRC" contenidos en el telegrama de petición; no es necesaria una entrada en el DB de origen P_SND_RK.

Finalización de petición en funciones de escritura

En códigos de función de escritura, el P_SND_RK activado finaliza tras la correcta recepción del telegrama de respuesta. Este suceso se le comunica al programa de usuario SIMATIC con un flanco en la salida **DONE** del P_SND_RK.

Si se han detectado **errores** en el tráfico de telegramas o si el esclavo responde con un telegrama de respuesta de **código de error**, se informará de ello mediante un flanco en la salida **ERROR**.

Finalización de petición en funciones de lectura

En las funciones de lectura, el P_SND_RK activado finaliza tras la correcta recepción del telegrama de respuesta **y** la completa trasferencia a la CPU de los datos recibidos.

Este suceso se le comunica al programa de usuario SIMATIC con un flanco en la salida **DONE** del P_SND_RK.

En ese momento los datos recibidos ya están disponibles en la CPU.

Si se han detectado **errores** en el tráfico de telegramas o si el esclavo responde con un telegrama de respuesta de **código de error**, se informará de ello mediante un flanco en la salida **ERROR**.

En este caso no se transfiere ningún dato recibido a la CPU.

Entrada STATUS en final de petición

En los casos en que una petición finaliza con **ERROR** en P_SND_RK, se produce la entrada adicional de un código de error en el parámetro **STATUS**.

Mediante este código de error puede determinarse la causa exacta del mismo

Transferencia de datos del módulo de comunicación a la CPU con P_RCV_RK (CP 341)

Requisito

Todos los códigos de función de **lectura** necesitan un P_RCV_RK.

Destino de datos

Con la disponibilidad de recepción activada de la instrucción P_RCV_RK, aplica los datos recibidos del módulo de comunicación y los introduce en un destino de datos especificado por los parámetros **DB_N0** y **DBB_N0**.

Visualización de la recepción de datos

Al usuario se le comunica la recepción de datos en la CPU mediante un flanco en la salida **NDR**.

En ese momento, en el parámetro **LEN** se indica la longitud del bloque de datos recibido.

La finalización de la petición de Modbus completa puede detectarse en la salida DONE de la instrucción P_SND_RK.

Comportamiento en caso de error

En el caso de errores durante la recepción o la transferencia, no se enviarán los datos a la CPU. La instrucción P_SND_RK finalizará en este caso con un flanco en la salida ERROR.

DB de destino P_RCV_RK

Los datos útiles recibidos por un código de función de lectura se introducen en el área de destino P_RCV_RK.

La estructura detallada del correspondiente DB de destino P_RCV_RK se incluye en la descripción de los códigos de función en el capítulo "Códigos de función (Página 1143)".

La longitud de los datos introducidos se muestra en el parámetro LEN del P_RCV_RK.

CP 441-2

Transferencia de datos de la CPU al módulo de comunicación con BSEND (CP 441-2)

Enlace de comunicación

El **parámetro ID** describe el enlace de comunicación unívoco a un interlocutor. El ID local de la configuración de la conexión debe introducirse aquí.

Relación de bloque

El **parámetro R_ID** describe la relación de bloque unívoca dentro de un enlace de comunicación.

En este driver pueden incluirse los valores **deseados0..255** para el R_ID en BSEND.

El BRCV correspondiente a las peticiones de lectura debe tener el R_ID parametrizado de la misma forma que en BSEND.

Activación

La ejecución de un código de función MODBUS se activa con una instrucción **BSEND** con un **flanco** en la entrada **REQ**.

De esta forma se transfieren al driver los parámetros necesarios para la ejecución del código de función.

Fuente de datos

Al activar BSEND, el **área de datos fuente** introducida en el parámetro **SD_1** con la longitud **LEN** se transfiere al módulo de comunicación.

Indicación de la longitud

La longitud **LEN** depende del código de función utilizado.

Código de función	Longitud LEN en bytes
01	6
02	6
03	6
04	6
05	6
06	6
07	2
08	6
11	2
12	2
15	>6
16	>6

Si en los códigos de función no se transfieren las cantidades de datos indicadas, no se realizará la petición y el BSEND terminará con un flanco en la salida ERROR.

DB de origen BSEND

En el área de datos fuente deben incluirse como datos útiles los parámetros necesarios para la ejecución de un código de función.

La estructura detallada del correspondiente DB de origen BSEND se incluye en la descripción de los códigos de función individuales en el apartado "Códigos de función (Página 1143)".

Generación de telegrama

Los telegramas de petición se envían al esclavo mediante los datos de origen BSEND transferidos y estructurados por el módulo de comunicación.

A continuación, el driver comprueba que la longitud LEN indicada en el BSEND se corresponda con la longitud prevista para este código de función.

Si no es así, la petición no se ejecuta y termina con un flanco en la salida ERROR del BSEND.

Si se utilizan códigos de función distintos a los listados, la petición activada no se ejecuta y termina con ERROR en el BSEND.

El módulo de comunicación forma los elementos "Contador de bytes" y "Comprobación CRC" contenidos en el telegrama de petición; no es necesaria una entrada en el DB de origen BSEND.

Finalización de petición en funciones de escritura

En códigos de función de escritura, el BSEND activado finaliza tras la correcta recepción del telegrama de respuesta.

Este suceso se le comunica al programa de usuario SIMATIC con un flanco en la salida **DONE** del BSEND.

Si se han detectado **errores** en el tráfico de telegramas o si el esclavo responde con un telegrama de respuesta de **código de error**, se informará de ello mediante un flanco en la salida **ERROR**.

Finalización de petición en funciones de lectura

En las funciones de lectura, el BSEND activado finaliza tras la correcta recepción del telegrama de respuesta **y** la completa trasferencia a la CPU de los datos recibidos.

Este suceso se le comunica al programa de usuario SIMATIC con un flanco en la salida **DONE** del BSEND.

En ese momento los datos recibidos ya están disponibles en la CPU.

Si se han detectado **errores** el tráfico de telegramas o si el esclavo responde con un telegrama de respuesta de **código de error**, se informará de ello mediante un flanco en la salida **ERROR**.

En este caso no se transfiere ningún dato recibido a la CPU.

Entrada en el área de registro de errores con fin de petición

En los casos en que una petición finaliza con **ERROR** en BSEND, se produce la entrada adicional de un código de error en el área de registro de errores.

Mediante este código de error puede determinarse la causa exacta del mismo

Transferencia de datos del módulo de comunicación a la CPU con BRCV (CP 441-2)

Enlace de comunicación

El **parámetro ID** describe el enlace de comunicación unívoco a un interlocutor. El ID local de la configuración de la conexión debe introducirse aquí.

Relación de bloque

El **parámetro R_ID** describe la relación de bloque unívoca dentro de un enlace de comunicación.

Todos los códigos de función **de lectura** necesitan un BRCV.
El **R_ID** parametrizado en **BRCV** debe ser el **mismo R_ID** que el del BSEND correspondiente con el que se activó la petición (valores preferidos 0 ... 255).

De esta forma es posible programar varios pares BSEND/BRCV en el programa de usuario SIMATIC.

Los telegramas de respuesta recibidos por el esclavo Modbus se recogen en distintas áreas de destino, dependiendo del R_ID utilizado por esta petición.

Destino de datos

Con la disponibilidad para recepción activada de la instrucción BRCV, éste aplica los datos recibidos del módulo de comunicación y los introduce en el destino de datos indicado en el parámetro **RD_1**. De esta forma, el destino de datos es variable.

Visualización de la recepción de datos

Al usuario se le comunica la recepción de datos en la CPU mediante un flanco en la salida **NDR**.

En ese momento, en el parámetro **LEN** se indica la longitud del bloque de datos recibido.

La finalización de la petición de Modbus completa puede detectarse en la salida DONE de la instrucción BSEND.

Comportamiento en caso de error

En el caso de errores durante la recepción o la transferencia, no se enviarán los datos a la CPU. BSEND finalizará en este caso con un flanco en la salida ERROR.

DB de destino BRCV

Los datos útiles recibidos por un código de función de lectura se introducen en el área de destino BRCV.

La estructura detallada del correspondiente DB de destino BRCV se incluye en la descripción de los códigos de función en el apartado "Códigos de función (Página 1143)".

La longitud de los datos introducidos se muestra en el parámetro LEN del BRCV.

ET 200S 1SI

Transferencia de datos con el maestro Modbus ET 200S 1SI

Introducción

La transmisión de datos entre el módulo y la CPU se realiza mediante las instrucciones S_SEND y S_RCV.

Instrucción S_SEND: Enviar datos a un interlocutor

Para ejecutar una petición del maestro Modbus tienen que activarse las instrucciones S_SEND y S_RCV. La instrucción S_SEND se activa con un flanco en la entrada REQ cuando deben emitirse datos al módulo. La instrucción S_RCV se prepara con EN_R=1 para recibir datos del módulo. En todos los códigos de función de lectura es necesario un S_RCV. La figura

siguiente muestra el comportamiento global de los parámetros S_SEND y S_RCV cuando se ejecuta una petición Modbus.

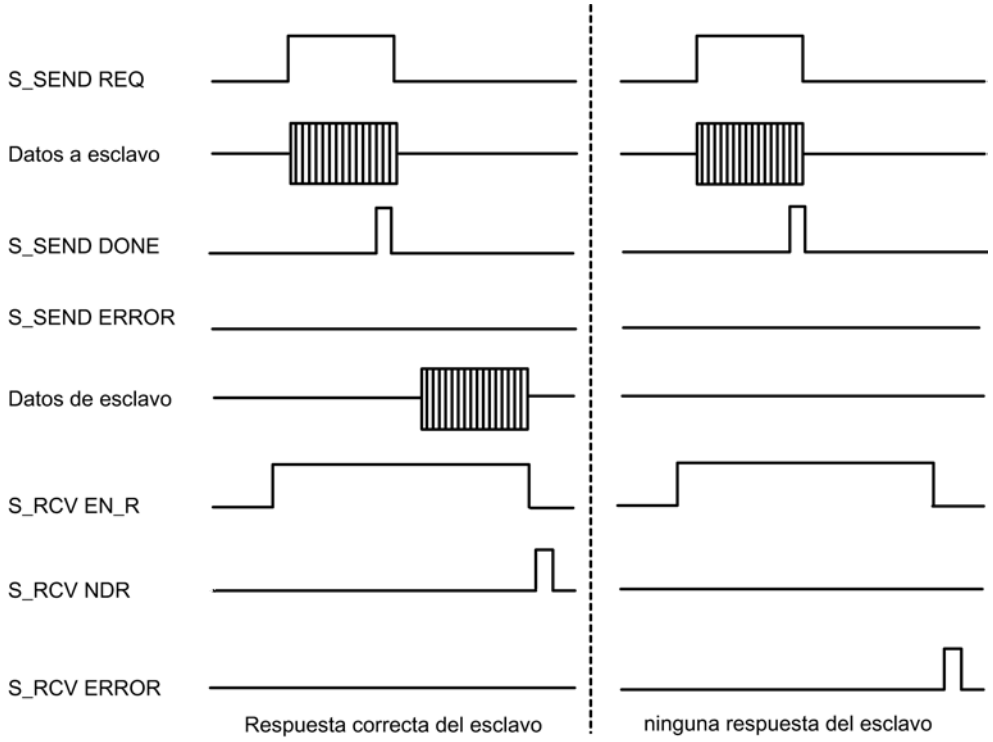


Figura 8-46 Cronograma de una petición Modbus

Con flanco positivo en la entrada REQ se inicia la transmisión de los datos. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción S_SEND se puede llamar en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al módulo y la instrucción S_SEND pasa al estado básico. Los datos que ya ha recibido el módulo se envían al interlocutor. Si hay un estado de señal "1" estático en la entrada R, la transmisión está desactivada.

En el parámetro LADDR se indica la dirección del módulo interfaz serie ET 200S Modbus/USS con el que debe realizarse la conexión.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número correspondiente de evento se indica en STATUS. Si no se produce ningún fallo, STATUS tiene el valor 0. DONE y ERROR/STATUS se indican también en caso de RESET de S_SEND. Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Petición de lectura del maestro Modbus

Dado que la interfaz entre el programa de usuario y el módulo interfaz funciona en modo semidúplex, hay que tener en cuenta lo siguiente:

Después de una petición de lectura del maestro Modbus con acuse positivo, hay que recoger los datos recibidos del módulo interfaz con la instrucción S_RCV antes de iniciar una nueva petición de envío del maestro Modbus.

Esclavo Modbus

Inicialización del módulo de comunicación

Introducción

Las instrucciones de comunicación Modbus (MODB_341, MODB_441 y S_MODB) para el driver cargable del esclavo Modbus deben llamarse en la CPU SIMATIC S7 en la parte cíclica.

La instrucción de comunicación Modbus inicializa el CP y ejecuta las funciones Modbus que el driver no puede ejecutar por sí mismo. La instrucción de comunicación Modbus también debe llamarse en el programa de usuario si el sistema maestro Modbus no utiliza estos códigos de función.

La comunicación entre CP y la instrucción se realiza mediante funciones del sistema operativo de la CPU y las instrucciones BSEND (CP 441-2) o P_SND_RK y P_RCV_RK (CP 341) llamadas por la instrucción.

Arranque, inicialización

Después de rearrancar o reiniciar la CPU, hay que inicializar la instrucción de comunicación Modbus.

La inicialización se activa con un flanco ascendente en la entrada CP_START.

La instrucción borra primero el DB de instancia, lee las áreas de operandos E, A, M, T, Z de la CPU y las almacena en el DB de instancia. De esta forma se puede comprobar si las solicitudes de escritura del sistema maestro Modbus sobrepasan el área.

Con una petición de envío, se comunica al CP el número del DB de instancia y el transcurso satisfactorio de la inicialización.

Si la inicialización termina con errores, no es posible una comunicación Modbus. Todas las solicitudes del sistema maestro Modbus se responden con un telegrama Exception Code.

DB de instancia

Todos los datos relevantes para la instrucción de comunicación Modbus se encuentran en un bloque de datos de instancia. Este bloque de datos es al mismo tiempo DB de instancia (múltiples instancias) para las instrucciones utilizadas y área de trabajo para la instrucción de comunicación Modbus. No es necesaria otra área de datos.

La instrucción de comunicación Modbus solo trabaja con el DB de instancia y con datos locales.

El DB de instancia solo permite acceso de lectura.

Tiempo de vigilancia

Tras encenderla, la CPU necesita unos segundos para hacer comprobaciones de hardware y de memoria antes de estar lista. Los intentos de inicialización realizados durante ese tiempo por la instrucción de comunicación Modbus finalizan con error. Por eso, durante ese tiempo de vigilancia se repite varias veces la petición de inicialización.

Leer intervalo de tiempo SYSTAT (solo CP 441-2)

Como leer periódicamente el SYSTAT en cada ciclo del PLC, o un ciclo de cada dos, sobrecarga innecesariamente el CP y el bus K y reduce el flujo de datos, se puede configurar un intervalo de tiempo para leer el SYSTAT (solo relevante para CP 441-2 (MODB_441)).

Códigos de función

Maestro Modbus (RTU)

Código de función 01 - Read Output Status

Función

Esta función permite leer bits individuales del esclavo.

Dirección inicial

El driver no comprueba el parámetro **Dirección inicial de bit** y lo envía sin modificarlo.

Número de bits

Como **número de bits**, number of coils, se admite cualquier valor entre **1** y **2040** (en el ET 200S 1SI Modbus hasta 2008).

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#1	Código de función
+2.0	bit_startadr	WORD	W#16#0040	Dirección inicial de bit
+4.0	bit_anzahl	INT	16	Número de bits

Ejemplo

Telegrama de solicitud FUNCTION 01:

05H	Dirección del esclavo
01H	Código de función
00H	Dirección inicial de bit "High"
40H	Dirección inicial de bit "Low"
00H	Número de bits "High"
10H	Número de bits "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 01:

05H	Dirección del esclavo
01H	Código de función
02H	Contador de bytes
01H	<Datos>
17H	<Datos>
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#1701	Datos

El driver registra los datos del telegrama de respuesta **por palabras** en el DB de destino.

El primer byte recibido se registra como byte Low de la primera palabra "data[1]", el tercer byte recibido como byte Low de la segunda palabra "data[2]", etc.

Si se leen menos de 9 bits o si sólo se ha leído un byte Low, en el byte High restante de la última palabra se registra el valor **00H**.

Código de función 02 - Read Input Status

Función

Esta función permite leer bits individuales del esclavo.

Dirección inicial

El driver no comprueba el parámetro **Dirección inicial de bit** y lo envía sin modificarlo.

Número de bits

Como **número de bits**, number of coils, se admite cualquier valor entre **1** y **2040** (en el ET 200S 1SI Modbus hasta 2008).

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#2	Código de función
+2.0	bit_startadr	WORD	W#16#0120	Dirección inicial de bit
+4.0	bit_anzahl	INT	24	Número de bits

Ejemplo

Telegrama de solicitud FUNCTION 02:

05H	Dirección del esclavo
02H	Código de función
01H	Dirección inicial de bit "High"
20H	Dirección inicial de bit "Low"
00H	Número de bits "High"
18H	Número de bits "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 02:

05H	Dirección del esclavo
02H	Código de función
03H	Contador de bytes
04H	<Datos>
26H	<Datos>
48H	<Datos>
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#2604	Datos
+2.0	data[2]	WORD	W#16#0048	Datos

El driver registra los datos del telegrama de respuesta **por palabras** en el DB de destino.

El primer byte recibido se registra como byte Low de la primera palabra "data[1]", el tercer byte recibido como byte Low de la segunda palabra "data[2]", etc.

Si se leen menos de 9 bits o si sólo se ha leído un byte Low, en el byte High restante de la última palabra se registra el valor **00H**.

Código de función 03 - Read Output Registers

Función

Esta función permite leer registros individuales del esclavo.

Dirección inicial

El driver no comprueba el parámetro **Dirección inicial del registro** y lo envía sin modificarlo.

Número de registros

Pueden leerse entre **1** y **127 registros como máximo** (con ET 200S 1SI Modbus máx. 125 registros) (1 registro = dos bytes).

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#3	Código de función
+2.0	reg_startadr	WORD	W#16#0040	Dirección inicial del registro
+4.0	reg_anzahl	INT	2	Número de registros

Ejemplo

Telegrama de solicitud FUNCTION 03:

05H	Dirección del esclavo
03H	Código de función
00H	Dirección inicial de registro "High"
40H	Dirección inicial de registro "Low"
00H	Número de registros "High"
02H	Número de registros "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 03:

05H	Dirección del esclavo
03H	Código de función
04H	Contador de bytes
21H	Dirección de registro 40H datos "High"
23H	Dirección de registro 40H datos "Low"
25H	Dirección de registro 41H datos "High"
27H	Dirección de registro 41H datos "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#2123	Datos
+2.0	data[2]	WORD	W#16#2527	Datos

Código de función 04 - Read Input Registers

Función

Esta función permite leer registros individuales del esclavo.

Dirección inicial

El driver no comprueba el parámetro **Dirección inicial del registro** y lo envía sin modificarlo.

Número de registros

Pueden leerse entre **1** y **127 registros como máximo** (con ET 200S 1SI Modbus máx. 125 registros) (1 registro = dos bytes).

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#4	Código de función
+2.0	reg_startadr	WORD	W#16#0050	Dirección inicial del registro
+4.0	reg_anzahl	INT	3	Número de registros

Ejemplo

Telegrama de solicitud FUNCTION 04:

05H	Dirección del esclavo
04H	Código de función
00H	Dirección inicial de registro "High"
50H	Dirección inicial de registro "Low"
00H	Número de registros "High"
03H	Número de registros "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 04:

05H	Dirección del esclavo
04H	Código de función
04H	Contador de bytes
31H	Dirección de registro 50H datos "High"
32H	Dirección de registro 50H datos "Low"
33H	Dirección de registro 51H datos "High"
34H	Dirección de registro 51H datos "Low"
35H	Dirección de registro 52H datos "High"
36H	Dirección de registro 52H datos "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#3132	Datos
+2.0	data[2]	WORD	W#16#3334	Datos
+4.0	data[3]	WORD	W#16#3536	Datos

Código de función 05 - Force Single Coil

Función

Con esta función puede activarse o borrarse un bit individual en el esclavo.

Dirección de bit

El driver no comprueba el parámetro **Dirección de bit** y lo envía sin modificarlo.

Estado de bit

El **estado de bit** admite los dos valores siguientes:

- FF00H = activar bit
- 0000H = borrar bit

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#5	Código de función
+2.0	bit_address	WORD	W#16#0019	Dirección de bit
+4.0	bit_state	WORD	W#16#FF00	Estado de bit

Ejemplo

Telegrama de solicitud FUNCTION 05:

05H	Dirección del esclavo
05H	Código de función
00H	Dirección de bit "High"
19H	Dirección de bit "Low"
FFH	Activar bit

00H	
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 05:

05H	Dirección del esclavo
05H	Código de función
00H	Dirección de bit "High"
19H	Dirección de bit "Low"
FFH	Estado de bit "High"
00H	Estado de bit "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

El telegrama de respuesta no se registra en un DB de recepción.

Código de función 06 - Preset Single Register

Función

Con este comando puede sobrescribirse un registro de esclavo con un valor nuevo.

Dirección de registro

El driver no comprueba el parámetro **Dirección de registro** y lo envía sin modificarlo.

Valor de registro

El **valor de registro** puede ser cualquier valor.

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#6	Código de función
+2.0	reg_address	WORD	W#16#0180	Dirección de registro
+4.0	reg_value	WORD	W#16#3E7F	Valor de registro

Ejemplo

Telegrama de solicitud FUNCTION 06:

05H	Dirección del esclavo
06H	Código de función
01H	Dirección de registro "High"
80H	Dirección de registro "Low"
3EH	Valor de registro "High"
7FH	Valor de registro "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 06:

05H	Dirección del esclavo
06H	Código de función
01H	Dirección de registro "High"
80H	Dirección de registro "Low"
3EH	Valor de registro "High"
7FH	Valor de registro "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

El telegrama de respuesta no se registra en un DB de recepción.

Código de función 07 - Read Exception Status

Función

Con este código de función pueden leerse 8 bits de evento del esclavo conectado.

El número de bit inicial del bit de evento está definido por el equipo conectado y, por lo tanto, el programa de usuario SIMATIC no tiene que preajustarlo.

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#7	Código de función

Ejemplo

Telegrama de solicitud FUNCTION 07:

05H	Dirección del esclavo
07H	Código de función
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 07:

05H	Dirección del esclavo
07H	Código de función
3EH	<Datos>
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#3Exx	Datos

El driver registra el único byte del telegrama de respuesta en el **byte High** del DB de destino data[1].

El byte Low de data[1] no se modifica.

Como longitud, en el parámetro LEN de la instrucción de recepción se indica el valor 1.

Código de función 08 - Loop Back Diagnostic Test

Función

Esta función sirve para comprobar el enlace de comunicación.

Para este código de función sólo se admite el **código de diagnóstico 0000**.

Código de diagnóstico

Para el parámetro **Código de diagnóstico** sólo se admite el valor 0000.

Valor de test

El **valor de test** puede ser cualquier valor.

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#8	Código de función
+2.0	diag_code	WORD	W#16#0000	Código de diagnóstico
+4.0	test_value	WORD	W#16#A5C3	Valor de test

Ejemplo

Telegrama de solicitud FUNCTION 08:

05H	Dirección del esclavo
08H	Código de función
00H	Código de diagnóstico "High"
00H	Código de diagnóstico "Low"
A5H	Valor de test "High"
C3H	Valor de test "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 08:

05H	Dirección del esclavo
08H	Código de función
00H	Código de diagnóstico "High"
00H	Código de diagnóstico "Low"
A5H	Valor de test "High"
C3H	Valor de test "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

El esclavo tiene que devolver el telegrama de solicitud recibido por el maestro como eco y sin modificarlo.

El telegrama de respuesta no se registra en un DB RCV.

Código de función 11 - Fetch Communications Event Counter

Función

Con este código de función el esclavo puede leerse una **"palabra de estado"** de 2 bytes y un **"contador de eventos"** de 2 bytes.

El significado de dicho parámetro se describe en el "GOULD MODICON Modbus Protocol".

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0B	Código de función

Ejemplo

Telegrama de solicitud FUNCTION 11:

05H	Dirección del esclavo
0BH	Código de función
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 11:

05H	Dirección del esclavo
0BH	Código de función
FEH	Palabra de estado "High"
DCH	Palabra de estado "Low"
01H	Contador de eventos "High"
08H	Contador de eventos "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#FEDC	Palabra de estado
+2.0	data[2]	WORD	W#16#0108	Contador de eventos

Código de función 12 - Fetch Communications Event Log

Función

Con este código de función el esclavo puede leer lo siguiente:

- **"palabra de estado"** de dos bytes;
- **"contador de eventos"** de dos bytes;
- **"contador de mensajes"** de dos bytes;
- **"bytes de eventos"** de 64 bytes.

El significado de dicho parámetro se describe en el "GOULD MODICON Modbus Protocol".

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0C	Código de función

Ejemplo

Telegrama de solicitud FUNCTION 12:

05H	Dirección del esclavo
0CH	Código de función
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 12:

05H	Dirección del esclavo
0CH	Código de función
46H	Contador de bytes

87H	Palabra de estado "High"
65H	Palabra de estado "Low"
01H	Contador de eventos "High"
08H	Contador de eventos "Low"
02H	Contador de mensajes "High"
20H	Contador de mensajes "Low"
01H	Byte de eventos 1
12H	Byte de eventos 2
:	:
C2H	Byte de eventos 63
D3H	Byte de eventos 64
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

DB de destino RCV

Contenido del área de destino RCV:

Dirección	Nombre	Tipo	Valor actual	Comentario
+0.0	data[1]	WORD	W#16#8765	Palabra de estado
+2.0	data[2]	WORD	W#16#0108	Contador de eventos
+4.0	data[3]	WORD	W#16#0220	Contador de mensajes
+6.0	bytedata[1]	BYTE	B#16#01	Byte de eventos 1
+7.0	bytedata[2]	BYTE	B#16#12	Byte de eventos 2
:	:			:
+68.0	bytedata[63]	BYTE	B#16#C2	Byte de eventos 63
+69.0	bytedata[64]	BYTE	B#16#D3	Byte de eventos 64

Código de función 15 - Force Multiple Coils

Función

Con este código de función se pueden modificar hasta 2040 bits (con ET 200S 1SI Modbus máx. 1744 bits) en el esclavo

Dirección inicial

El driver no comprueba el parámetro **Dirección inicial de bit** y lo envía sin modificarlo.

Número de bits

Como **número de bits**, number of coils, se admite cualquier valor entre **1** y **2040** (en el ET 200S 1SI Modbus hasta 1744 bits).

Con ello se define cuántos bits pueden sobrescribirse en el esclavo.

El driver crea el parámetro "Contador de bytes" incluido en el telegrama de solicitud a causa del parámetro "Número de bits" transferido.

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#0F	Código de función
+2.0	bit_startadr	WORD	W#16#0058	Dirección inicial de bit
+4.0	bit_anzahl	INT	10	Número de bits
+6.0	coil_state[1]	WORD	W#16#EFCD	Estado Coil 5FH..58H/57H..50H

Ejemplo

Telegrama de solicitud FUNCTION 15:

05H	Dirección del esclavo
0FH	Código de función
00H	Dirección de bit "High"
50H	Dirección de bit "Low"
00H	Número de bits "High"
0AH	Número de bits "Low"
02H	Contador de bytes
CDH	Estado Coil 50H..57H
EFH	Estado Coil 58H..59H
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 15:

05H	Dirección del esclavo
0FH	Código de función
00H	Dirección de bit "High"
50H	Dirección de bit "Low"
00H	Número de bits "High"
0AH	Número de bits "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

El telegrama de respuesta no se registra en un DB de recepción.

Código de función 16 - Preset Multiple Registers

Función

El código de función 16 permite sobrescribir un máximo de **127 registros** en (en ET 200S 1SI Modbus máx. 109 registros) en el esclavo con un telegrama de solicitud.

Dirección inicial

El driver no comprueba el parámetro **Dirección inicial del registro** y lo envía sin modificarlo.

Número de registros

Pueden leerse entre **1** y **127 registros como máximo** (con ET 200S 1SI Modbus máx. 109 registros) (1 registro = dos bytes).

El driver crea el parámetro "Contador de bytes" incluido en el telegrama de solicitud a causa del parámetro "Número de registros" transferido.

DB de origen SEND

Estructura del área de origen SEND:

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.0	Dirección	BYTE	B#16#5	Dirección del esclavo
+1.0	Función	BYTE	B#16#10	Código de función
+2.0	reg_startadr	WORD	W#16#0060	Dirección inicial del registro
+4.0	reg_anzahl	INT	3	Número de registros
+6.0	reg_data[1]	WORD	W#16#41A1	Datos de registro
+8.0	reg_data[2]	WORD	W#16#42A2	Datos de registro
+10.0	reg_data[3]	WORD	W#16#43A3	Datos de registro

Ejemplo

Telegrama de solicitud FUNCTION 16:

05H	Dirección del esclavo
10H	Código de función
00H	Dirección de registro "High"
60H	Dirección de registro "Low"
00H	Número de registros "High"
03H	Número de registros "Low"
06H	Contador de bytes

41H	<reg_data[1]> "High"
A1H	<reg_data[1]> "Low"
42H	<reg_data[2]> "High"
A2H	<reg_data[2]> "Low"
43H	<reg_data[3]> "High"
A3H	<reg_data[3]> "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta del esclavo FUNCTION 16:

05H	Dirección del esclavo
10H	Código de función
00H	Dirección de registro "High"
60H	Dirección de registro "Low"
00H	Número de registros "High"
03H	Número de registros "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

El telegrama de respuesta no se registra en el DB de recepción.

Esclavo Modbus (RTU)

Código de función 01 - Read Coil (Output) Status

Función

Esta función permite leer diferentes bits de las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro Modbus.

Telegrama de solicitud

ADDR	FUNC	start_address	bit_number	CRC

Telegrama de respuesta

ADDR	FUNC	Byte_count n	n-Byte DATA	CRC

start_address

La dirección de bit Modbus "**start_address**" la interpreta el driver del siguiente modo:

el driver comprueba si "start_address" se encuentra en una de las áreas que se indicaron en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 01, 05, 15**" (desde/hasta: marcas, salidas, temporizadores, contadores).

Si la dirección de bit MODBUS **start_address** se encuentra en una de estas áreas,	se accede a este **área de memoria SIMATIC**.	
De aaaaa a bbbbb	A partir de la marca	M uuuuu.0
De ccccc a ddddd	A partir de la salida	Q ooooo.0
De eeeee a fffff	A partir del temporizador	T ttttt
De ggggg a hhhhh	A partir del contador	C ccccc

La dirección para el acceso, la conversión de dirección, se calcula entonces del modo siguiente:

Acceso empezando con la		fórmula de conversión SIMATIC			
Byte de marca	=	((start_address	– aaaaa)	/ 8)	+ uuuuu
Byte de salida	=	((start_address	– ccccc)	/ 8)	+ ooooo
Temporizador	=	((start_address	– eeeee)	/ 16)	+ ttttt
Contadores	=	((start_address	– ggggg)	/ 16)	+ ccccc

Acceso a "marcas" y "salidas"

Al acceder a las áreas SIMATIC "Marcas" y "Salidas", se calcula el número de bit restante y se utiliza para direccionar el bit en cuestión dentro del primer/último byte de marca o de salida.

Acceso a "Temporizadores" y "Contadores"

Al calcular las direcciones, los siguientes resultados deben ser divisibles por 16 sin resto:

- (start_address - eeeee)
- (start_address - ggggg)

Sólo es posible el acceso por palabras empezando en el límite de palabra.

bit_number

Como **bit_number**, number of coils, se admiten valores entre **1** y **2040** (en el ET 200S 1SI Modbus hasta 1768 bits). Se lee esta cantidad de bits.

Acceso a "Temporizadores" y "Contadores"

Al acceder a las áreas SIMATIC "Temporizadores" y "Contadores", "bit_number" tiene que ser divisible por 16 (sólo acceso por palabras).

Tenga en cuenta que, al aplicar el CP 341, pueden leerse un máximo de 16 "temporizadores" o "contadores".

Nota

Observe las restricciones específicas de la CPU.

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-80 Conversión del direccionamiento MODBUS para los códigos de función FC 01, 05 y 15

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
De *0* a *2047*	A partir de la marca	M *1000.0*
De *2048* a *2559*	A partir de la salida	Q *256,0*
De *4096* a *4607*	A partir del temporizador	T *100*
De *4608* a *5119*	A partir del contador	C *200*

Telegrama de solicitud FUNCTION 01:

05H	Dirección del esclavo ADDR
01H	Código de función FUNC
00H	start_address "High"
40H	start_address "Low"
00H	bit_number "High"
20H	bit_number "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 01:

05H	Dirección del esclavo ADDR
01H	Código de función FUNC
04H	Byte_count
01H	<DATA 1> M 1008.0 - M 1008.7
17H	<DATA 2> M 1009.0 - M 1009.7
02H	<DATA 3> M 1010.0 - M 1010.7
18H	<DATA 4> M 1011.0 - M 1011.7
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "start_address" 0040 Hex (64 decimal) se encuentra en el área "Marcas":

Byte de marca	=	((start_address	- aaaaa)	/ 8)	+ uuuuu
	=	((64	- 0)	/ 8)	+ 1000
	=	1008 ;			

El número de bits restante da:

Nº bit restante	=	((start_address	- aaaaa)	% 8)	[Módulo 8]
	=	((64	- 0)	% 8)	
	=	0 ;			

Se produce un acceso que empieza con la marca M 1008.0 hasta M 1011.7 inclusive.

Número de bits:

El número de bits de Modbus “bit_number" 0020 Hex (32 decimal) significa que deben leerse 32 bits = 4 bytes.

Otros ejemplos

A continuación se muestra una tabla con más ejemplos de acceso.

Todos los ejemplos se basan en la clasificación de áreas anteriormente indicada.

start_address		Acceso a SIMATIC empezando					->	con
Hex	Decimal	(Decimal)						
0000	0	Marca	((0	- 0)	/ 8)	+ 1000	->	M 1000.0
0021	33	Marca	((33	- 0)	/ 8)	+ 1000	->	M 1004.1
0400	1024	Marca	((1024	- 0)	/ 8)	+ 1000	->	M 1128.0
0606	1542	Marca	((1542	- 0)	/ 8)	+ 1000	->	M 1192.6
0840	2112	Salida	((2112	- 2048)	/ 8)	+ 256	->	Q 264.0
09E4	2532	Salida	((2532	- 2048)	/ 8)	+ 256	->	Q 316.4
1010	4112	Temporizad ores	((4112	- 4096)	/ 16)	+ 100	->	T 101
10C0	4288	Temporizad ores	((4288	- 4096)	/ 16)	+ 100	->	T 112
1200	4608	Contadores	((4608	- 4608)	/ 16)	+ 200	->	C 200
13E0	5088	Contadores	((5088	- 4608)	/ 16)	+ 200	->	C 230

Código de función 02 - Read Input Status

Función

Esta función permite leer diferentes bits de las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro Modbus.

Telegrama de solicitud

ADDR	FUNC	start_address	bit_number	CRC

Telegrama de respuesta

ADDR	FUNC	Byte_count n	n-Byte DATA	CRC

start_address

La dirección de bit Modbus "**start_address**" la interpreta el driver del siguiente modo:

el driver comprueba si "start_address" se encuentra en una de las áreas que se indicaron en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 02**" (desde/hasta: marcas, entradas).

Si la dirección de bit MODBUS se encuentra en el área **start_address**,	se accede al **área de memoria SIMATIC**.	
De kkkkk a lllll	A partir de la marca	M vvvvv.0
De nnnnn a rrrrr	A partir de la entrada	I sssss.0

La dirección para el acceso, la conversión de dirección, se calcula entonces del modo siguiente:

Acceso empezando con la		fórmula de conversión SIMATIC
Byte de marca	=	((start_address – *kkkkk*) / 8) + *vvvvv*
Byte de entrada	=	((start_address – *nnnnn*) / 8) + *sssss*

Acceso a "marcas" y "entradas"

Al acceder a las áreas SIMATIC "Marcas" y "Entradas", se calcula el número de bit restante y se utiliza para direccionar el bit en cuestión dentro del primer/último byte de marca o de entrada.

bit_number

Como **bit_number**, number of coils, se admiten valores entre **1** y **2040** (en el ET 200S 1SI Modbus hasta 1768 bits). Se lee esta cantidad de bits.

Nota

Observe las restricciones específicas de la CPU.

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-81 **Conversión del direccionamiento MODBUS en el código de función FC 02**

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
De 0 a 4095	A partir de la marca	M 2000.0
De 4096 a 5119	A partir de la entrada	E 128.0

Telegrama de solicitud FUNCTION 02:

05H	Dirección del esclavo ADDR
02H	Código de función FUNC
10H	start_address "High"
30H	start_address "Low"
00H	bit_number "High"
18H	bit_number "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 02:

05H	Dirección del esclavo ADDR
02H	Código de función FUNC
03H	Byte_count
12H	<DATA 1> I 134.0 - I 134.7
34H	<DATA 2> I 135.0 - I 135.7
56H	<DATA 3> I 136.0 - I 136.7
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "start_address" 1030 Hex (4144 decimal) se encuentra en el área "Entradas":

Byte de entrada	=	((start_address	- nnnnn)	/ 8)	+ sssss
	=	((4144	- *4096*)	/ 8)	+ *128*
	=	134 ;			

El número de bits restante da:

Nº bit restante	=	((start_address	- nnnnn)	% 8)	[Módulo 8]
	=	((4144	- 4096)	% 8)	
	=	0 ;			

Se produce un acceso que empieza con la entrada I 134.0 hasta I 136.7 inclusive.

Número de bits:

El número de bits MODBUS “bit_number" 0018 Hex (24 decimal) significa que deben leerse 24 bits = 3 bytes.

Otros ejemplos

A continuación se muestra una tabla con más ejemplos de acceso.

Todos los ejemplos se basan en la clasificación de áreas anteriormente indicada.

start_address		Acceso a SIMATIC empezando					->	con
Hex	Decimal	(Decimal)						
0000	0	Marca	((0	- 0)	/ 8)	+ 2000	->	M 2000.0
0071	113	Marca	((113	- 0)	/ 8)	+ 2000	->	M 2014.1
0800	2048	Marca	((2048	- 0)	/ 8)	+ 2000	->	M 2256.0
0D05	3333	Marca	((3333	- 0)	/ 8)	+ 2000	->	M 2416.5
1000	4096	Entrada	((4096	- 4096)	/ 8)	+ 128	->	I 128.0
10A4	4260	Entrada	((4260	- 4096)	/ 8)	+ 128	->	I 148.4

Código de función 03 - Read Output Registers

Función

Esta función permite leer palabras de datos de un bloque de datos a través del sistema maestro MODBUS.

Telegrama de solicitud

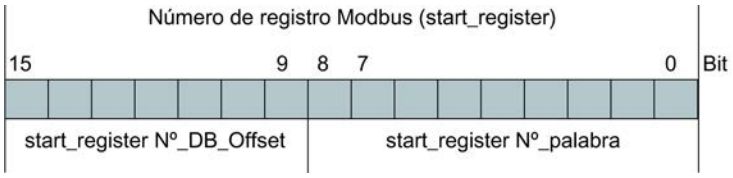
ADDR	FUNC	start_register	register_number	CRC

Telegrama de respuesta

ADDR	FUNC	Byte_count n	n/2-Register DATA (High,Low)	CRC

start_register

La dirección de registro MODBUS "**start_register**" la interpreta el driver del siguiente modo:



Para seguir creando direcciones, el driver utiliza el "número de DB básico" (a partir de DB xxxxx) registrado en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 03, 06, 16**".

La dirección para el acceso, la conversión de dirección, se calcula entonces del modo siguiente:

Acceso a la		fórmula de conversión SIMATIC
Bloque de datos DB (DB resultante)	=	(Número de DB básico *xxxxx* + start_register Nº_DB_Offset)
Palabra de datos DBW	=	(start_register Nº_palabra * 2)

Fórmula de cálculo para start_register

Si se conoce el DB resultante que se debe leer, la dirección MODBUS **start_register** requerida en el sistema maestro puede calcularse con la siguiente fórmula:

start_register = ((DB resultante - Número de DB básico) * 512) +
(Palabra de datos_DBW / 2)

Para ello sólo pueden emplearse números pares de palabra de datos.

register_number

Como **register_number**, number of coils, se admite cualquier valor entre **1** y **127** (en el ET 200S 1SI Modbus hasta 110 registros). Se lee esta cantidad de registros. Aquí hay que tener en cuenta la siguiente regla:

$$(\text{register_number})_{\text{máx.}} = 512 - \text{start_register}$$

Nota

Observe las restricciones específicas de la CPU.

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-82 Conversión del direccionamiento MODBUS para los códigos de función FC 03, 06 y 16

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
0	A partir del bloque de datos (número de DB básico)	DB 800

Telegrama de solicitud FUNCTION 03:

05H	Dirección del esclavo ADDR
03H	Código de función FUNC
00H	start_register "High"
50H	start_register "Low"
00H	register_number "High"
02H	register_number "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

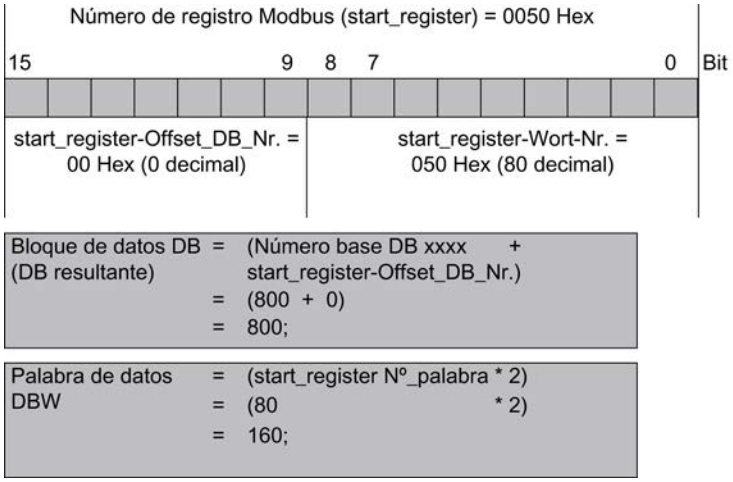
Telegrama de respuesta FUNCTION 03:

05H	Dirección del esclavo ADDR
03H	Código de función FUNC
04H	Byte_count
87H	<DATA 1> DBW 160 "High"
65H	<DATA 2> DBW 160 "Low"
43H	<DATA 3> DBW 161 "High"
21H	<DATA 4> DBW 161 "Low"

xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "start_register" 0050 Hex (80 decimal) se interpreta:



Se accede al DB 800, palabra de datos DBW 160.

Número de registros:

El número de registros MODBUS “register_number" 0002 Hex (2 decimal) significa que deben leerse 2 registros = 2 palabras de datos.

Otros ejemplos

A continuación se muestra una tabla con más ejemplos de acceso.

			start_register				
start_register		Nº de DB básico	Nº_DB_Offset	Número de palabra		DB resultante	DBW
Hex	Decimal	Decimal	Decimal	Hex	Decimal	Decimal	Decimal
0000	0	800	0	000	0	800	0
01F4	500	800	0	1F4	500	800	1000
0200	512	800	1	000	0	801	0
02FF	767	800	1	0FF	255	801	510
0300	768	800	1	100	256	801	512
03FF	1023	800	1	1FF	511	801	1022
0400	1024	800	2	000	0	802	0

Código de función 04 - Read Input Registers

Función

Esta función permite leer palabras de datos de un bloque de datos a través del sistema maestro MODBUS.

Telegrama de solicitud

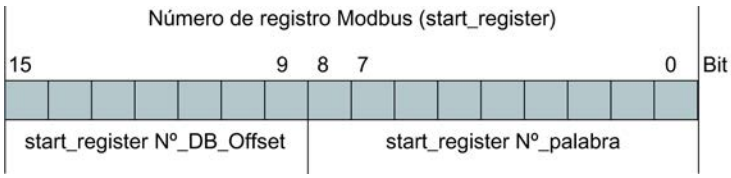
ADDR	FUNC	start_register	register_number	CRC

Telegrama de respuesta

ADDR	FUNC	Byte_count n	n/2-Register DATA (High,Low)	CRC

start_register

La dirección de registro MODBUS "**start_register**" la interpreta el driver del siguiente modo:



Para seguir creando direcciones, el driver utiliza el "número de DB básico" (a partir de DB xxxxx) introducido en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 04**".

El cálculo de la dirección para el acceso y la conversión de la dirección se realiza en dos niveles de la siguiente forma:

Acceso a la		fórmula de conversión SIMATIC
Bloque de datos DB (DB resultante)	=	(número DB básico *xxxxx* + start_register-Offset_DB_Nr.)
Palabra de datos DBW	=	(start_register Nº_palabra * 2)

Fórmula de cálculo para start_register

Si se conoce el DB resultante que se debe leer, la dirección MODBUS **start_register** requerida en el sistema maestro puede calcularse con la siguiente fórmula:

start_register = ((DB resultante - Número de DB básico) * 512) +(palabra de datos_DBW / 2)

Únicamente pueden emplearse números pares de palabras de datos.

register_number

Como **register_number**, number of coils, se admite cualquier valor entre **1** y **127** (en el ET 200S 1SI Modbus hasta 110 registros). Se lee esta cantidad de registros. Debe tenerse en cuenta la siguiente regla:

$(\text{register_number})_{\text{máx.}}$ = 512 - start_register

Nota

Observe las restricciones específicas de la CPU.

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-83 Conversión del direccionamiento MODBUS para el código de función FC 04

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
0	A partir del bloque de datos (número de DB básico)	DB 900

Telegrama de solicitud FUNCTION 04:

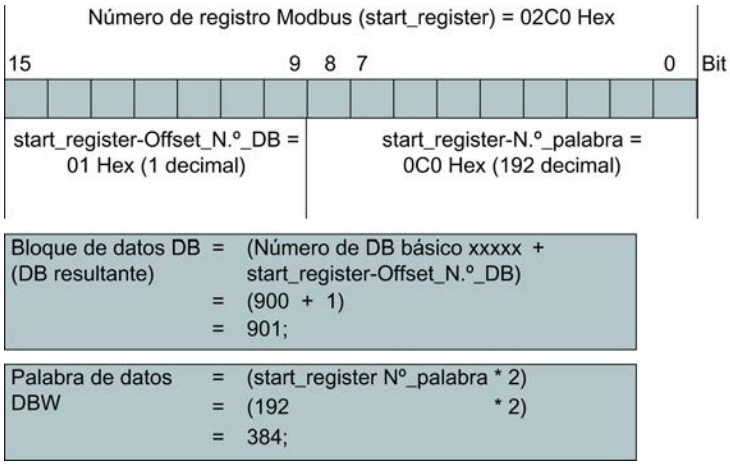
05H	Dirección del esclavo ADDR
04H	Código de función FUNC
02H	start_register "High"
C0H	start_register "Low"
00H	register_number "High"
03H	register_number "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 04:

05H	Dirección del esclavo ADDR
04H	Código de función FUNC
06H	Byte_count
A1H	<DATA 1> DBW 384 "High"
A2H	<DATA 2> DBW 384 "Low"
A3H	<DATA 3> DBW 385 "High"
A4H	<DATA 4> DBW 385 "Low"
A5H	<DATA 5> DBW 386 "High"
A6H	<DATA 6> DBW 386 "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "start_register" 02C0 Hex (704 decimal) se interpreta:



Se accede al DB 901, palabra de datos DBW 384.

Número de registros:

El número de registros MODBUS “register_number" 0003 Hex (3 decimal) significa que deben leerse 3 registros = 3 palabras de datos.

Otros ejemplos

A continuación se muestra una tabla con más ejemplos de acceso.

			start_register				
start_register		Nº de DB básico	Nº_DB_Offset	Número de palabra		DB resultante	DBW
Hex	decimal	decimal	decimal	Hex	decimal	decimal	decimal
0000	0	900	0	000	0	900	0
0064	100	900	0	064	100	900	200
00C8	200	900	0	0C8	200	900	400
0190	400	900	0	190	400	900	800
1400	5120	900	10	000	0	910	0
1464	5220	900	10	064	100	910	200
14C8	5320	900	10	0C8	200	910	400

Código de función 05 - Force Single Coil

Función

Esta función permite escribir un bit de las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro MODBUS.

Telegrama de solicitud

ADDR	FUNC	coil_address	DATA-on/off	CRC

Telegrama de respuesta

ADDR	FUNC	coil_address	DATA-on/off	CRC

coil_address

La dirección de bit MODBUS "**coil_address**" la interpreta el driver del siguiente modo:

el driver comprueba si "coil_address" se encuentra en una de las áreas que se introdujeron en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 01, 05, 15**" (desde/hasta: marcas, salidas, temporizadores, contadores).

Si la dirección de bit MODBUS **coil_address** se encuentra en el área,	se accede al **área de memoria SIMATIC**.	
De aaaaa a bbbbb	A partir de la marca	M uuuuu.0
De ccccc a ddddd	A partir de la salida	Q ooooo.0

El cálculo de la dirección para el acceso (conversión de la dirección) se realiza de la siguiente forma:

Acceso empezando con lafórmula de conversión SIMATIC

Byte de marca	=	((coil_address	- *aaaaa*)	/ 8)	+ *uuuuu*
Byte de salida	=	((coil_address	- *ccccc*)	/ 8)	+ *ooooo*

Acceso a "marcas" y "salidas"

Al acceder a las áreas SIMATIC "Marcas" y "Salidas", se calcula el número de bits restante y se utiliza para direccionar el bit en cuestión dentro del byte de marca o de salida.

Acceso a "Temporizadores" y "Contadores"

Con el código de función FC 05 no se permite el acceso al área SIMATIC Temporizadores y Contador; el driver avisa de ello con un telegrama de error.

DATA-on/off

Se permiten los dos siguientes valores como **DATA-on/off**:

- FF00H -> activar bit
- 0000H -> borrar bit

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-84 Conversión del direccionamiento MODBUS para los códigos de función FC 01, 05, 15

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
De 0 a 2047	A partir de la marca	M 1000.0
de 2048 a 2559	A partir de la salida	Q 256,0

Telegrama de solicitud FUNCTION 05:

05H	Dirección del esclavo ADDR
05H	Código de función FUNC
08H	coil_address "High"
09H	coil_address "Low" Q 257.1
FFH	DATA-on/off "High"
00H	DATA-on/off "Low"

xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 05:

05H	Dirección del esclavo ADDR
05H	Código de función FUNC
08H	coil_address "High"
09H	coil_address "Low" Q257.1
FFH	DATA-on/off "High"
00H	DATA-on/off "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "coil_address" 0809 Hex (2057 decimal) se encuentra en el área "Salidas":

Byte de salida	=	((coil_address	- ccccc)	/ 8)	+ ooooo
	=	((2057	- 2048)	/ 8)	+ 256
	=	257 ;			

El número de bits restante da:

Nº bit restante	=	((coil_address	- ccccc)	% 8)	[Módulo 8]
	=	((2057	- 2048)	% 8)	
	=	1 ;			

Se accede a la salida Q 257.1.

Otros ejemplos

Pueden consultarse más ejemplos de accesos a marcas y salidas en el FC 01.

Código de función 06 - Preset Single Register

Función

Esta función permite escribir palabras de datos en un bloque de datos de la CPU a través del sistema maestro MODBUS.

Telegrama de solicitud

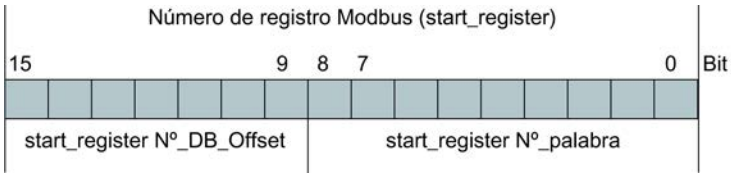
ADDR	FUNC	start_register	DATA-value (High,Low)	CRC

Telegrama de respuesta

ADDR	FUNC	start_register	DATA-value (High,Low)	CRC

start_register

La dirección de registro MODBUS "**start_register**" la interpreta el driver del siguiente modo:



Para seguir creando direcciones, el driver utiliza el "número de DB básico" (a partir de DB xxxxx) registrado en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 03, 06, 16**".

El cálculo de la dirección para el acceso y la conversión de la dirección se realiza en dos niveles de la siguiente forma:

Acceso a la		fórmula de conversión SIMATIC
Bloque de datos DB (DB resultante)	=	(número DB básico xxxxx + start_register-Offset_DB_Nr.)
Palabra de datos DBW	=	(start_register Nº_palabra * 2)

Fórmula de cálculo para start_register

Si se conoce el DB resultante en el que se debe escribir, la dirección MODBUS **start_register** requerida en el sistema maestro puede calcularse con la siguiente fórmula:

start_register = ((DB resultante - Número de DB básico) * 512) +(palabra de datos_DBW / 2)

Únicamente pueden emplearse números pares de palabras de datos.

DATA-value

El valor de registro, como DATA-**value**, puede ser cualquier valor.

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-85 Conversión del direccionamiento MODBUS para los códigos de función FC 03, 06 y 16

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
0	A partir del bloque de datos (número de DB básico)	DB 800

Telegrama de solicitud FUNCTION 06:

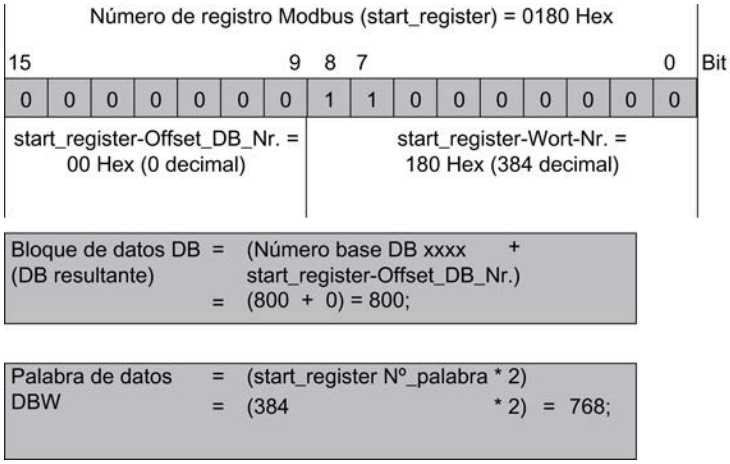
05H	Dirección del esclavo ADDR
06H	Código de función FUNC
01H	start_register "High"
80H	start_register "Low" DBW 768
2BH	DATA-value "High"
1AH	DATA-value "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 06:

05H	Dirección del esclavo ADDR
06H	Código de función FUNC
01H	start_register "High"
80H	start_register "Low" DBW 768
2BH	DATA-value "High"
1AH	DATA-value "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "start_register" 0180 Hex (384 decimal) se interpreta:



Se accede al DB 800, palabra de datos DBW 768.

Otros ejemplos

Pueden consultarse otros ejemplos de acceso en el FC 03.

Código de función 08 - Loop Back Diagnostic Test

Función

Esta función sirve para comprobar el enlace de comunicación.

Tiene efecto en la CPU S7 y los programas y datos de usuario. El driver reenvía independientemente el telegrama recibido al sistema maestro.

Telegrama de solicitud

ADDR	FUNC	Código de diagnóstico (High,Low)	Test-Data	CRC

Telegrama de respuesta

ADDR	FUNC	Código de diagnóstico (High,Low)	Test-Data	CRC

Código de diagnóstico

Únicamente se admite el **código de diagnóstico 0000**.

Test-Data

Valor de 16 bits deseado.

Ejemplo de aplicación

Telegrama de solicitud FUNCTION 08:

05H	Dirección del esclavo ADDR
08H	Código de función FUNC
00H	Código de diagnóstico "High"
00H	Código de diagnóstico "Low"
A5H	Valor de test "High"
C3H	Valor de test "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 08:

05H	Dirección del esclavo ADDR
08H	Código de función FUNC
00H	Código de diagnóstico "High"
00H	Código de diagnóstico "Low"
A5H	Valor de test "High"
C3H	Valor de test "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Código de función 15 - Force Multiple Coils

Función

Esta función permite escribir varios bits de las áreas de memoria SIMATIC listadas a continuación a través del sistema maestro MODBUS.

Telegrama de solicitud

ADDR	FUNC	start_address	quantity	byte_count n	n-DATA	CRC

Telegrama de respuesta

ADDR	FUNC	start_address	quantity	CRC

start_address

La dirección de bit MODBUS "**start_address**" la interpreta el driver del siguiente modo:

el driver comprueba si "start_address" se encuentra en una de las áreas que se indicaron en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 01, 05, 15**" (desde/hasta: marcas, salidas, temporizadores, contadores).

Si la dirección de bit MODBUS se encuentra en el área **start_address**,	se accede al **área de memoria SIMATIC**.	
De aaaaa a bbbbb	A partir de la marca	M uuuuu.0
De ccccc a ddddd	A partir de la salida	Q ooooo.0

El cálculo de la dirección para el acceso, la conversión de la dirección, se realiza de la siguiente forma:

Acceso empezando con la fórmula de conversión SIMATIC

Byte de marca = ((start_address - aaaaa) / 8) + uuuuu

Byte de salida = ((start_address - ccccc) / 8) + ooooo

Acceso a "marcas" y "salidas"

Al acceder a las áreas SIMATIC "Marcas" y "Salidas", se calcula el número de bits restante y se utiliza para direccionar el bit en cuestión dentro del byte de marca o de salida.

Acceso a "Temporizadores" y "Contadores"

Con el código de función FC 15 no se permite el acceso al área SIMATIC Temporizadores y Contador; el driver avisa de ello con un telegrama de error.

Quantity

Como **quantity**, número de bits, se admite cualquier valore entre **1** y **2040** (en el ET 200S 1SI Modbus hasta 1976).

Nota

Observe las restricciones específicas de la CPU.

DATA

En el **campo DATA** se encuentran los valores deseados como estados de bits.

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-86 Conversión del direccionamiento MODBUS para los códigos de función FC 01, 05 y 15

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
De 0 a 2047	A partir de la marca	M 1000.0
De 2048 a 2559	A partir de la salida	Q 256,0

Acción:

El sistema maestro MODBUS pretende describir las marcas M 1144.0 ... M 1144.7 y M 1145.0 ... M 1145.3 con los siguientes estados de bits:

Marcas M 1144	7	6	5	4	3	2	1	0	Bit
	ON	ON	OFF	OFF	ON	ON	OFF	ON	
Marcas M 1145	7	6	5	4	3	2	1	0	Bit
	-	-	-	-	ON	OFF	OFF	ON	

Telegrama de solicitud FUNCTION 15:

05H	Dirección del esclavo ADDR
0FH	Código de función FUNC
04H	start_address "High"
80H	start_address "Low" (M 1144.0 ...)
00H	quantity "High"
0CH	quantity "Low" (12 bits)
02H	bytecount
CDH	Coil de estado (M 1144.0 ... M 1144.7)
09H	Coil de estado (M 1145.0 ... M 1145.3)
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Telegrama de respuesta FUNCTION 15:

05H	Dirección del esclavo ADDR
0FH	Código de función FUNC

04H	start_address "High"
80H	start_address "Low"
00H	quantity "High"
0CH	quantity "Low"
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "coil_address" 0480 Hex (1152 decimal) se encuentra en el área "Marcas":

Byte de marca	=	((start_address	- aaaaa)	/ 8)	+ uuuuu
	=	((1152	- 0)	/ 8)	+ 1000
	=	1144 ;			

El número de bits restante da:

Nº bit restante	=	((start_address	- aaaaa)	% 8)	[Módulo 8]
	=	((1152	- 0)	% 8)	
	=	0 ;			

Se produce un acceso a la marca que comienza en M 1144.0.

Otros ejemplos

Pueden consultarse más ejemplos de accesos a marcas y salidas en el FC 01.

Código de función 16 - Preset Multiple Registers

Función

Este código de función permite escribir palabras de datos en un bloque de datos de la CPU SIMATIC a través del sistema maestro MODBUS.

Telegrama de solicitud

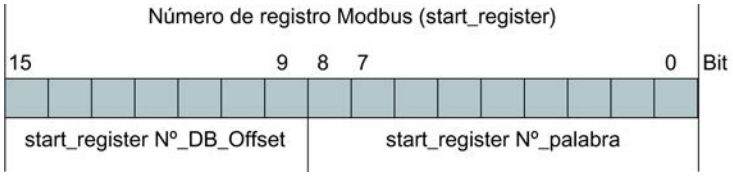
ADDR	FUNC	start_register	quantity	byte_count n	n-DATA (High,Low)	CRC

Telegrama de respuesta

ADDR	FUNC	start_register	quantity	CRC

start_register

La dirección de registro MODBUS "**start_register**" la interpreta el driver del siguiente modo:



Para seguir creando direcciones, el driver utiliza el "número de DB básico" (a partir de DB xxxxx) registrado en la parametrización de la máscara "**Conversión del direccionamiento MODBUS para FC 03, 06, 16**".

El cálculo de la dirección para el acceso y la conversión de la dirección se realiza en dos niveles de la siguiente forma:

Acceso a la	fórmula de conversión SIMATIC
Bloque de datos DB (DB resultante)	= (número DB básico *xxxxx* + start_register-Offset_DB_Nr.)
Palabra de datos DBW	= (start_register Nº_palabra * 2)

Fórmula de cálculo para start_register

Si se conoce el DB resultante en el que se debe escribir, la dirección MODBUS **start_register** requerida en el sistema maestro puede calcularse con la siguiente fórmula:

start_register = ((DB resultante - Número de DB básico) * 512) +(palabra de datos_DBW / 2)

Únicamente pueden emplearse números pares de palabras de datos.

Quantity

Como **quantity**, número de registros, se admite cualquier valor entre **1** y **127** (en el ET 200S 1SI Modbus hasta 123). Aquí hay que tener en cuenta la siguiente regla:

$(\text{Quantity})_{\text{máx.}}$ = 512 - start_register

Nota

Observe las restricciones específicas de la CPU.

DATA (High,Low)

Puede utilizarse cualquier valor como **DATA (High,Low)** (valor de registro).

Ejemplo de aplicación

Ejemplo para la parametrización:

Tabla 8-87 Conversión del direccionamiento MODBUS para los códigos de función FC 03, 06, 16

Dirección MODBUS en el telegrama de transferencia	Área de memoria SIMATIC	
0	A partir del bloque de datos (número de DB básico)	DB 800

Acción:

El sistema maestro MODBUS pretende describir las palabras de datos DBW 100, DBW 102, DBW 104 del DB 800 con los valores CD09 Hex, DE1A Hex, EF2B Hex.

Telegrama de solicitud FUNCTION 16:

05H	Dirección del esclavo ADDR
10H	Código de función FUNC
00H	start_register "High"
32H	start_register "Low" DBW 100
00H	quantity "High"
03H	quantity "Low" (3 registros)
06H	bytecount
CDH	register value -High (DBW100)
09H	register value -Low
DEH	register value -High (DBW102)
1AH	register value -Low
EFH	register value -High (DBW104)
2BH	register value -Low
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

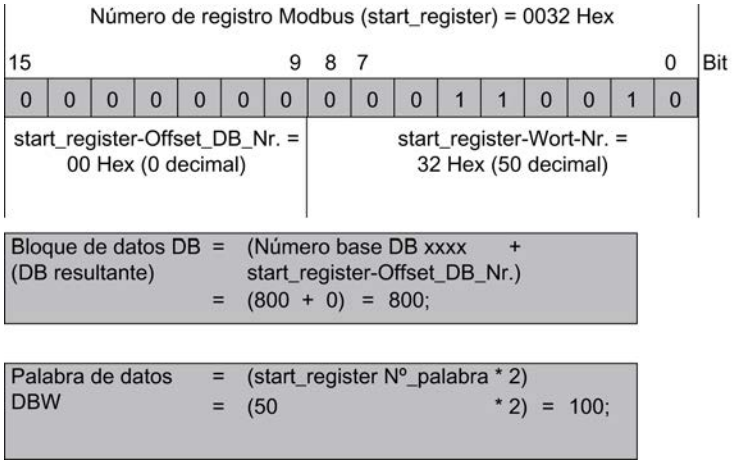
Telegrama de respuesta FUNCTION 16:

05H	Dirección del esclavo ADDR
10H	Código de función FUNC
00H	start_register "High"
32H	start_register "Low"
00H	quantity "High"

03H	quantity "Low" (3 registros)
xxH	Código de verificación CRC "Low"
xxH	Código de verificación CRC "High"

Cálculo de direcciones:

La dirección MODBUS "start_register" 0032 Hex (50 decimal) se interpreta:



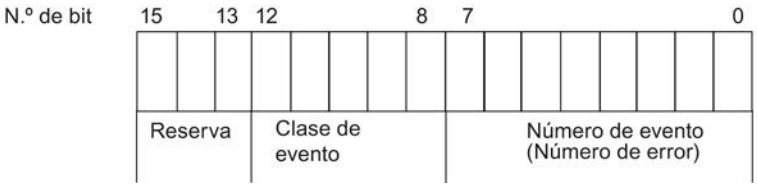
Se accede al DB 800, palabra de datos DBW 100.

Otros ejemplos

Pueden consultarse otros ejemplos de acceso en el FC 03.

Área de registro de errores

Esquema numérico clase de evento/número de evento



Clases de eventos

Clase de evento	Significado
2	Error en la inicialización
5	Error en la edición de una petición CPU

Clase de evento	Significado
8	Error de recepción
14 ($0E_H$)	Error general de edición
30 ($1E_H$)	Error durante la comunicación entre la interfaz serie y la CPU.

Clase de evento/número de evento

Código de error(W#16 #...)	Descripción	Solución
0201	No hay parametrización (válida).	Proporcione parámetros correctos al módulo. Dado el caso, compruebe que el montaje de la instalación es correcto.
0502	Petición no admisible en este estado operativo del módulo interfaz serie ET 200S Modbus/USS (ejemplo: La interfaz de dispositivo no está parametrizada).	Evalúe la alarma de diagnóstico y elimine el error de la forma correspondiente.
050E	Longitud de telegrama no válida	La longitud del telegrama de envío supera los 224 bytes. La petición de envío ha sido interrumpida por el módulo ET 200S Modbus/USS. Seleccione una longitud de telegrama inferior.
0518	• La longitud de transferencia en el envío es demasiado grande (> 4 kbytes) • La longitud de transferencia en SEND es demasiado pequeña.	Comprobar parámetro LEN en SEND.
0530	Petición de envío de maestro Modbus rechazada, ya que todavía no se ha llamado la respuesta del interlocutor a una petición de envío de lectura anterior del maestro Modbus.	Después de una petición de lectura del maestro Modbus correcta, primero hay que recoger del módulo la respuesta del interlocutor antes de iniciar una nueva petición de envío del maestro Modbus.
0551	Error de desarrollo de trama en la comunicación entre el módulo interfaz serie ET 200S Modbus/USS y la CPU. El error se ha producido durante la transmisión de un telegrama recibido del módulo interfaz serie ET 200S SI en la CPU.	El módulo y la CPU han interrumpido la transmisión. Repita la petición de recepción. El módulo interfaz serie ET 200S Modbus/USS vuelve a enviar el aviso recibido.
0806	Se ha sobrepasado el tiempo de retardo de caracteres ZVZ	Error en el dispositivo interlocutor o deben eliminarse interferencias importantes en la línea de transmisión.
080A	Desbordamiento del búfer de recepción en el maestro durante la recepción del telegrama de reacción.	Compruebe los ajustes de protocolo del esclavo.
080C	Se ha detectado un error de transferencia en uno de los caracteres (error de paridad, de overrun o de bit de parada (Frame)).	Comprobar si las interferencias afectan a la línea de transferencia. De ser necesario, modifique la estructura del sistema o el tendido de la línea. Compruebe si los parámetros de protocolo de la velocidad de transmisión, el número de bits de datos, la paridad, el número de bits de parada en el módulo de comunicación y el interlocutor acoplado están ajustados de la misma forma.

Código de error(W#16 #...)	Descripción	Solución
080D	BREAK Se ha interrumpido la línea de recepción hasta el dispositivo interlocutor.	Establezca la conexión entre los equipos o active el dispositivo interlocutor. En el modo TTY, comprobar si la corriente de línea continúa fluyendo en estado de reposo. Compruebe y modifique si fuera necesario la ocupación de la línea de recepción R(A), R(B) de 2 hilos en una conexión RS422/485 (X27).
0810	Error de paridad: Hay una interrupción en la línea de conexión (rotura de hilo) de ambos interlocutores.	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos interlocutores. Modifique la configuración de su sistema o el cableado.
0811	Error de trama de caracteres: Hay una interrupción en la línea de conexión (rotura de hilo) de ambos interlocutores.	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos interlocutores. Modifique la configuración de su sistema o el cableado.
0812	Se han recibido más caracteres después de que la interfaz serie haya puesto CTS en OFF.	Parametrice el interlocutor de nuevo o vacíe la interfaz serie con mayor rapidez.
0830	El tiempo de vigilancia de respuesta tras el envío de un telegrama de solicitud ha trascurrido sin que se haya detectado el inicio del telegrama de reacción.	Compruebe si la línea de transferencia se ha visto afectada por algún problema (de ser necesario, utilizar el analizador de interfaz). Compruebe si los parámetros de protocolo de la velocidad de transmisión, el número de bits de datos, la paridad, el número de bits de parada en el módulo de comunicación y el interlocutor acoplado están ajustados de la misma forma. Compruebe si el tiempo de vigilancia de respuesta parametrizado en el cuadro de diálogo de propiedades del módulo es suficiente. Compruebe si la dirección del esclavo indicada está o no disponible.
0831	El primer carácter del telegrama de reacción del esclavo no es igual a la dirección de esclavo enviado en el telegrama de solicitud (en el modo de operación "Normal").	Ha respondido un esclavo equivocado. Compruebe si la línea de transferencia se ha visto afectada por algún problema (de ser necesario, utilizar el analizador de interfaz).
0832	Desbordamiento del búfer de recepción en el módulo de comunicación al recibir el telegrama de reacción.	Compruebe los ajustes de protocolo del esclavo.
0833	El número de bits o registros con el código de función FC 15/16 y el elemento de telegrama byte_count no coinciden.	Corrija el número de bits/registros o byte_count.
0834	Codificación de bit no admisible para "Activar bit"/"Desactivar bit".	Para FC05 utilice sólo las codificaciones 0000Hex o FF00Hex.
0835	Detectado subcódigo de diagnóstico no admisible (no igual que 0000Hex) con el código de función FC 08 "Loop Back Test".	Para FC08 utilice sólo el subcódigo 0000Hex.

Código de error(W#16 #...)	Descripción	Solución
0836	El valor calculado internamente con la suma de comprobación CRC-16 no concuerda con la suma de comprobación CRC recibida.	Compruebe el cálculo de la suma de comprobación CRC en el sistema maestro Modbus.
0837	Error de proceso de telegrama: El sistema maestro Modbus envía un nuevo telegrama de petición antes que el driver haya transmitido el último telegrama de reacción.	Aumente el timeout para el telegrama de reacción del esclavo en el sistema maestro Modbus.
0850	Longitud del telegrama de recepción mayor de 224 bytes o bien mayor que la longitud de telegrama parametrizada.	Adaptar la longitud de telegrama del interlocutor.
0E01	Error en la inicialización del proceso SCC específico del driver.	Volver a parametrizar y cargar el driver.
0E02	Error en el arranque del driver: proceso SCC erróneo activo (driver SCC). El driver no funciona con este driver SCC.	Volver a parametrizar y cargar el driver.
0E03	Error en el arranque del driver: proceso erróneo de transferencia de datos activo (interfaz con las instrucciones). El driver no funciona con este proceso de transferencia de datos.	Volver a parametrizar y cargar el driver.
0E04	Error en el arranque del driver: submódulo de interfaz no permitido El driver no puede funcionar con el submódulo de interfaz parametrizado.	Comprobar y corregir la parametrización.
0E05	Error en el dongle del driver: no se ha conectado ningún dongle o se ha producido un error en el dongle conectado. El driver no está preparado para entrar en funcionamiento.	Compruebe si hay un dongle de driver conectado al módulo de comunicación. Si es así, sustituya el dongle erróneo por uno correcto.
0E06	Error en el dongle del driver: el contenido del dongle no es válido. El driver no está preparado para entrar en funcionamiento.	Solicite un dongle correcto a la filial de SIEMENS a la que adquirió el driver.
0E10	Error interno de procedimiento: rama predeterminada en el autómata del procedimiento.	Rearranque del módulo de comunicación (Power ON)
0E11	Error interno de procedimiento: rama predeterminada en estado de procedimiento enviar/recibir.	Rearranque del módulo de comunicación (Power ON)
0E12	Error interno en autómata activo: rama predeterminada	Rearranque del módulo de comunicación (Power ON)
0E13	Error interno en autómata pasivo: rama predeterminada	Rearranque del módulo de comunicación (Power ON)
0E20	El número de los bits de datos en este acoplamiento debe ser 8. El driver no está preparado para entrar en funcionamiento.	Corrija la parametrización del driver.
0E21	El factor de multiplicación parametrizado para el tiempo de retardo de caracteres no se encuentra dentro del ámbito de 1 a 10. El driver funciona con 1 como ajuste estándar.	Corrija la parametrización del driver.

Código de error(W#16 #...)	Descripción	Solución
0E22	No se permite el modo de operación parametrizado del driver. Deberá indicarse el modo "Normal" o de "Supresión de perturbaciones". El driver no está preparado para entrar en funcionamiento.	Corrija la parametrización del driver.
0E23	El valor no permitido para el tiempo de vigilancia de respuesta parametriza: Valores permitidos 5 … 65500 ms. El driver no está preparado para entrar en funcionamiento.	Corrija la parametrización del driver.
0E24	Se han parametrizado límites no permitidos para los accesos de escritura. El driver no está listo.	Corrija la parametrización del driver.
0E25	Se ha parametrizado una combinación "de/a" no válida en la introducción de las áreas "cambio del direccionamiento Modbus en la FC 01, 05, 15". (Área de marcas, salidas, temporizadores, contadores). El driver no está operativo.	Corrija la parametrización del driver.
0E26	Al introducir las áreas "Conversión del direccionamiento Modbus para FC 02", se ha parametrizado una combinación "desde/hasta" no permitida. (Área marcas, entradas). El driver no está listo.	Corrija la parametrización del driver.
0E27	Al introducir las áreas "Conversión del direccionamiento Modbus para FC 01, 05, 15", hay solapamiento en las combinaciones "desde/hasta". (Área marcas, salidas, temporizadores, contadores). El driver no está listo.	Corrija la parametrización del driver.
0E28	Al introducir las áreas "Conversión del direccionamiento Modbus para FC 02", hay solapamiento en las combinaciones "desde/hasta". (Área marcas, entradas). El driver no está listo.	Corrija la parametrización del driver.
0E2E	Se ha producido un error al leer el archivo de los parámetros de interfaz. El driver no está preparado para entrar en funcionamiento.	Rearranque del módulo de comunicación (Power ON).
0E30	Error interno de la transferencia a la CPU: Confirmación inesperada pasiva.	Puede ignorarse si sólo aparece de manera esporádica.
0E31	Time out al transferir datos a la CPU	Comprobar la interfaz CP-CPU.
0E32	Error al transferir datos con RCV a la CPU: analizar causa - El error en detalle se encuentra en SYSTAT, antes de esta entrada.	Comprobar la interfaz CP-CPU.
0E33	Error interno de la transferencia de datos a la CPU: Estado del autómata no permitido	Comprobar la interfaz CP-CPU.
:		
0E38	Se ha producido un error al acceder a una de las áreas SIMATIC "marcador, salida, temporizador, contador, entrada" con el código de función FC 01 o FC 02: P. ej.: No existe esa entrada, o se ha intentado leer más allá del final del área.	Compruebe que existe el área SIMATIC llamada y que no se ha intentado acceder más allá del final del área.

Código de error(W#16 #...)	Descripción	Solución
0E39	Se ha producido un error al acceder al área SIMATIC "Bloque de datos" con el código de función FC 03, 04, 06, 16: El bloque de datos no existe o es demasiado corto.	Compruebe que el bloque de datos llamado existe y que es suficientemente largo.
0E3A	Se ha producido un error al ejecutar la petición de escritura con el código de función FC 05,15: El bloque de datos de instancia de la instrucción Modbus no existe o es demasiado corto.	Compruebe que el DB de instancia parametrizado en la instrucción de comunicación Modbus existe y que es suficientemente largo.
0E3B	TimeOut (espera de acuse) en la ejecución de una petición de escritura por la instrucción de comunicación Modbus.	Comprobar la configuración de conexiones y la interfaz CP-CPU (BSEND) y, en caso necesario, volver a cargar la instrucción de comunicación Modbus.
0E3C	Petición no permitida en este driver.	Sólo se permiten SEND, RCV, STATUS (sólo con CP 441-2)
0E40	Se ha introducido un valor demasiado pequeño para el parámetro LEN en la instrucción SEND.	La longitud mínima es de 2 bytes.
0E41	Se ha introducido un valor demasiado pequeño para el parámetro LEN en la instrucción SEND. Se necesita una mayor longitud para el código de función transferido.	La longitud mínima en este código de función es de 6 bytes.
0E42	No se permite el código de función transferido.	Sólo se permiten los códigos de función incluidos en el capítulo " Códigos de función (Página 1143) ".
0E43	No se permite la dirección del esclavo 0 (= Broadcast) en este código de función.	Utilice una dirección del esclavo 0 sólo en los códigos de función adecuados para ello.
0E44	El valor del parámetro transferido "Número de bits" no se encuentra en el área 1... 2040.	El parámetro "Número de bits" debe encontrarse en el área 1 ... 2040.
0E45	El valor del parámetro transferido "Número de registro" no se encuentra en el área 1... 127.	El parámetro "Número de registro" debe encontrarse en el área 1 ... 127.
0E46	Código de función 15 ó 16: el valor del parámetro transferido "Número de bits " o "Número de registro" no se encuentra en el área 1 ... 2040 ni en la 1 ... 127.	El parámetro "Número de bits" o el parámetro "Número de registro" debe encontrarse en el área 1 ... 2040 ó 1 ... 127.
0E47	Código de función 15 ó 16: el parámetro LEN de la instrucción SEND no se corresponde con el parámetro transferido "Número de bits" ni "Número de registro". El parámetro LEN es demasiado bajo.	Aumente el parámetro LEN en SEND hasta que se transfiera un número suficiente de datos útiles al módulo de comunicación. Debido al "Número de bits" o al "Número de registro" debe transferirse un mayor número de datos útiles al módulo de comunicación.
0E48	Código de función 05: la codificación dada en DB de origen SEND para "Activar bit" (FF00H) o "Borrar bit" (0000H) es errónea.	Sólo se permiten las codificaciones FF00H y 0000H.
0E49	Código de función 08: El "Código de diagnóstico" dado en DB de origen SEND es erróneo.	Sólo se permite el "Código de diagnóstico " 0000H.
0E50	En las áreas de SIMATIC de temporizadores/contadores orientadas a palabras, el número de bits sobrante de la dirección Modbus ≠ 0.	Utilizar solo direcciones Modbus que den como resultado números de bits válidos.
0E51	La dirección Modbus recibida queda fuera de las áreas parametrizadas "desde/hasta".	Utilizar en el telegrama de solicitud solo direcciones ya definidas en la parametrización.

Código de error(W#16 #...)	Descripción	Solución
0E52	• Se ha rebasado un límite de área SIMATIC en un intento de acceso del sistema maestro Modbus: Número de DB resultante < 1 • Acceso de escritura en un área no autorizada (parametrización) • Acceso de escritura al DB de instancia de la instrucción de comunicación	Limitar las áreas de acceso a áreas de memoria SIMATIC válidas.
0E53	Se ha rebasado un límite de área SIMATIC en un intento de acceso por parte del sistema maestro Modbus: P. ej.: Desbordamiento en el cálculo del número de DB resultante (> 65535)	Limitar las áreas de acceso a áreas de memoria SIMATIC válidas.
0E54	Acceso más allá del final del área parametrizada o acceso más allá del final del área SIMATIC.	Limitar las áreas de acceso a áreas de memoria SIMATIC válidas.
0E55	No se permite el acceso de escritura en esta área de memoria SIMATIC.	Efectuar accesos de escritura solo en las áreas de datos SIMATIC de marcas y salidas.
0E56	No es posible el funcionamiento del acoplamiento, ya que la instrucción de comunicación no funciona.	Llame la instrucción de comunicación Modbus cíclicamente en el programa de usuario STEP 7. En caso necesario, vuelva a inicializar la instrucción de comunicación.
0E57	Al procesar el código de función Modbus se ha producido un error en la instrucción de comunicación.	Analice la cause exacta.
0E4A	La longitud de este código de función supera la longitud máxima.	En el manual encontrará los valores de longitud máxima para cada código de función.
0E4F	CP 441: La R_ID indicada en la instrucción SEND no se permite en este driver.	Sólo pueden utilizarse las R_ID 0 ... 255 (00 00 00 00 ... 00 00 00 FFH).
	CP 341: El R_TYP indicado en la instrucción SEND_RK no se permite en este driver.	Debe introducirse "X" como R_TYP.
0E50	Dirección del esclavo errónea: la dirección del esclavo no es igual a la dirección del esclavo enviada.	Ha respondido un esclavo equivocado. Compruebe si la línea de transferencia se ha visto afectada por algún problema (de ser necesario, utilizar el analizador de interfaz).
0E51	Código de función erróneo: el código de función recibido en el telegrama de reacción no es igual al código de función enviado.	Compruebe el equipo esclavo.
0E52	Flujo insuficiente de bytes: se han recibido menos caracteres de los generados por el contador de bytes del telegrama de reacción o de los que se esperaban en este código de función.	Compruebe el equipo esclavo.
0E53	Flujo excesivo de bytes: se han recibido más caracteres de los generados por el contador de bytes del telegrama de reacción o de los que se esperaban en este código de función.	Compruebe el equipo esclavo.
0E54	Contador de bytes demasiado pequeño: el contador de bytes recibido en el telegrama de reacción es demasiado pequeño.	Compruebe el equipo esclavo.

Código de error(W#16 #...)	Descripción	Solución
0E55	Contador de bytes erróneo: el contador de bytes recibido en el telegrama de reacción es erróneo.	Compruebe el equipo esclavo.
0E56	Eco erróneo: los datos del telegrama de reacción repetidos por el esclavo (número de bits, etc.) no son iguales a los datos enviados en el telegrama de solicitud.	Compruebe el equipo esclavo.
0E57	Error en la comprobación CRC: se ha detectado un fallo en la comprobación de la suma de verificación CRC16 del telegrama de reacción del esclavo.	Compruebe el equipo esclavo.
0E61	telegrama de reacción con Exception Code 01: Illegal function	Veáse "Manual del equipo esclavo"
0E62	telegrama de reacción con Exception Code 02: Illegal data address	Veáse "Manual del equipo esclavo"
0E63	telegrama de reacción con Exception Code 03: Illegal data value	Veáse "Manual del equipo esclavo"
0E64	telegrama de reacción con Exception Code 04: Failure in associated device	Veáse "Manual del equipo esclavo"
0E65	telegrama de reacción con Exception Code 05: Acknowledge	Veáse "Manual del equipo esclavo"
0E66	telegrama de reacción con Exception Code 06: Busy, Rejected message	Veáse "Manual del equipo esclavo"
0E67	telegrama de reacción con Exception Code 07: Negative Acknowledgement	Veáse "Manual del equipo esclavo"
1E0D	"Cancelación de petición por rearranque, rearranque completo o inicialización"	
1E0E	Error estático en la llamada de la instrucción DP_RDDAT. El valor de respuesta RET_VAL de la instrucción se deposita en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E0F	Error estático en la llamada de la instrucción DP_WRDAT. El valor de respuesta RET_VAL de la instrucción se deposita en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E10	Error estático en la llamada de la instrucción RD_LGADR. El valor de respuesta RET_VAL de la instrucción se deposita en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E11	Error estático en la llamada de la instrucción RDSYSST. El valor de respuesta RET_VAL de la instrucción se deposita en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E20	Parámetro fuera de rango	Indique un parámetro dentro del rango permitido para el bloque de función.
1E41	El número de bytes indicado en el parámetro LEN de la instrucción no es admisible.	Respete el rango de entre 1 y 256 bytes.

Comunicación USS

Resumen de la comunicación USS

Clasificación dentro del entorno del sistema

La siguiente descripción USS se refiere al uso del módulo ET 200S 1SI en la variante Modbus/USS.

Introducción

El protocolo USS es un protocolo sencillo de transferencia de datos serie, que está adaptado a las necesidades de la tecnología de accionamientos.

El protocolo USS define una técnica de acceso de acuerdo con el principio maestroesclavo para la comunicación a través de un bus serie. Es posible conectar al bus un maestro y un máximo de 31 esclavos. Maestros y esclavos forman una red USS. Los datos intercambiados entre maestro y esclavos se denominan datos de red. El bloque de datos de red contiene los datos de red del maestro. El maestro selecciona los diferentes esclavos en el telegrama con la ayuda de un carácter de dirección. Un esclavo no puede enviar nunca sin que el maestro lo haya activado antes. Esto significa que no es posible una transferencia de datos directa entre esclavos. La comunicación funciona en modo semidúplex. La función del maestro no puede transferirse. El sistema USS sólo dispone de un maestro.

Estructura de telegrama

Cada telegrama empieza con un carácter de inicio (STX), seguido de la información sobre la longitud (LGE) y el byte de dirección (ADR). A continuación sigue el campo de datos. El telegrama finaliza con el carácter de comprobación de bloque (BCC).

STX	LGE	ADR	1	2	...	N	BCC

En el caso de datos de un bloque de datos de red compuestos por una palabra (16 bits), primero se envía el byte High y después el byte Low. En el caso de datos de palabras dobles, primero se envía la palabra High y, a continuación, la palabra Low.

El protocolo no identifica ninguna tarea en los campos de datos.

Codificación de datos

Los datos se codifican del modo siguiente:

- STX: 1 byte, inicio de texto, 02H
- LGE: 1 byte, contiene la longitud de telegrama como número binario
- ADR: 1 byte, contiene la dirección del esclavo y el tipo de telegrama en código binario
- Campos de datos: Un byte cada uno, el contenido depende de la tarea
- BCC: 1 byte, carácter de comprobación de bloque

Procedimiento de transmisión de datos

El maestro se ocupa de la transferencia de datos cíclica en telegramas. El maestro direcciona sucesivamente todas las estaciones esclavo con un telegrama de tarea. Las estaciones en cuestión responden con un telegrama de respuesta. De acuerdo con el procedimiento maestro-esclavo, el esclavo debe enviar el telegrama de respuesta al maestro después de haber recibido el telegrama de tarea. Sólo entonces el maestro puede comunicarse con el siguiente esclavo.

Estructura general del bloque de datos de red

El bloque de datos de red está dividido en dos secciones: Parámetros (PKW) y datos de proceso (PZD).

STX	LGE	ADR	Parámetros (PKW)	Datos de proceso (PZD)	BCC

- Área de parámetros (PKW)
  El área PKW gestiona la transferencia de parámetros entre dos interlocutores (p. ej. autómata y accionamiento). Esto incluye, por ejemplo, leer y escribir valores de parámetros así como leer descripciones de parámetros y el texto correspondiente. La interfaz de PKW suele contener tareas para manejo e indicadores, mantenimiento y diagnóstico.
- Área de datos de proceso (PZD)
  El área PZD incluye señales necesarias para la automatización:
  - Palabras de control y valores consigna del maestro al esclavo
  - Palabras de estado y valores reales del esclavo al maestro

  Los accionamientos esclavos definen el contenido del área de parámetros y del área de datos de proceso. Encontrará más información al respecto en la documentación del accionamiento.

Maestro USS

Sinopsis de funciones

Procedimiento de transmisión de datos de red

Las instrucciones procesan cíclicamente la transferencia de datos de red con un máximo de 31 accionamientos esclavos de acuerdo con el orden indicado en la lista de sondeo (DB de parametrización). Cada esclavo sólo tiene una petición activa. El usuario guarda los datos de red para cada esclavo en un bloque de datos (bloque de datos de red) y los llama desde allí. Se transfieren al procesador de comunicaciones a través de otra área de datos (DB de procesador de comunicaciones) de acuerdo con la definición de programa en la lista de sondeo y se llaman desde allí.

Para este procedimiento se requieren dos llamadas de funciones (una instrucción de envío y una de recepción). Una tercera función soporta la creación y la asignación de valores predeterminados de los bloques de datos necesarios para la comunicación.

Características y prestaciones:

- Creación de áreas de datos para la comunicación según la configuración de bus
- Valores predeterminados de la lista de sondeo
- Estructura de telegramas de acuerdo con la especificación USS
- El intercambio de datos de red puede parametrizarse de acuerdo con la estructura necesaria de datos de red
- Ejecución y supervisión de peticiones de PKW
- Gestión de informes sobre modificación de parámetros
- Supervisión de todo el sistema y eliminación de fallos

Pueden usarse diferentes estructuras de datos de red para enviar datos de red. Es decir, que el contenido de los datos de red se puede seleccionar según la aplicación.

Según sea la estructura seleccionada, los datos de red disponen de un área PZD para los datos de proceso y de un área PKW para el procesamiento de parámetros.

En el área PKW, el maestro puede leer y escribir los valores de parámetros y el esclavo puede visualizar modificaciones de parámetros mediante los informes correspondientes.

El área PZD contiene las señales necesarias para el control de procesos, p. ej. palabras de control, el valor consigna del maestro al esclavo, así como palabras de estado y valores reales del esclavo al maestro.

El orden correcto de las llamadas de instrucciones es: S_USST, S_SEND, S_RCV, S_USSR. Esto es importante porque las salidas de las instrucciones S_SEND y S_RCV sólo son válidas en el ciclo actual del sistema de automatización.

La figura siguiente muestra el intercambio de datos entre el programa de usuario y el esclavo USS.

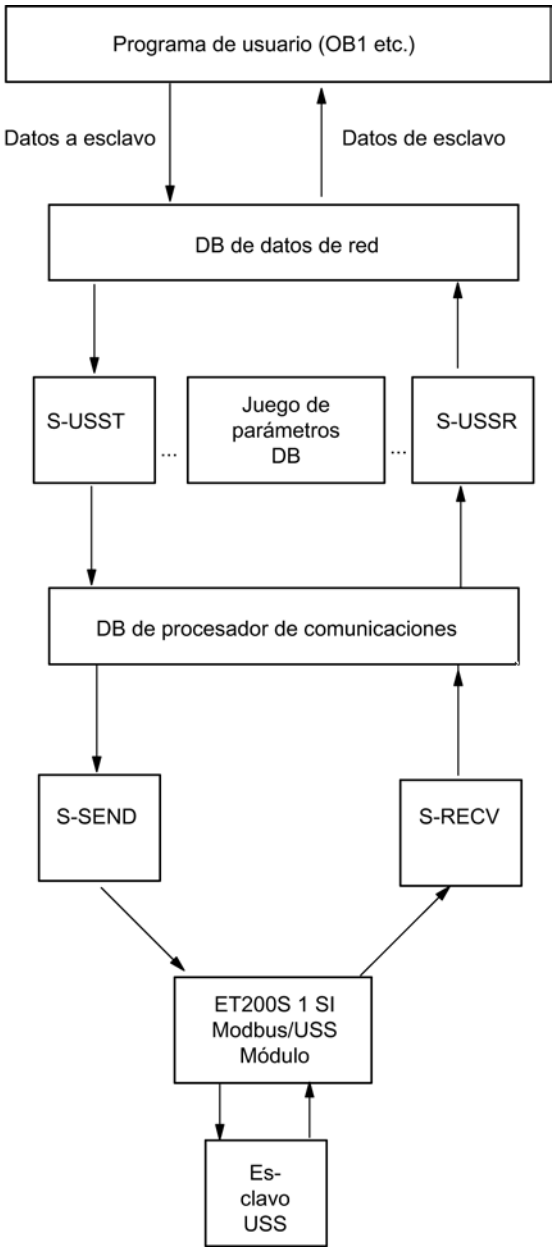


Figura 8-47 Intercambio de datos entre el programa de usuario y el esclavo USS

Información adicional

Encontrará información sobre los bloques de datos utilizados en:

- DB de datos de red (Página 8040)
- DB de parametrización (Página 8044)
- DB del procesador de comunicaciones (Página 8046)

Arranque y diagnóstico

Comportamiento de arranque

Comportamiento de arranque

El arranque del módulo de comunicación se divide en las siguientes fases:

- Inicialización (POWER ON del módulo de comunicación)
- Parametrización

Inicialización

En cuanto el módulo de comunicación recibe tensión, tras la ejecución de un programa de test de hardware se prepara el firmware en el módulo de comunicación para el servicio.

Parametrización

Durante la parametrización, el módulo de comunicación recibe los parámetros del módulo asignados al slot actual. El módulo de comunicación ya está listo para el servicio.

Transiciones de estado operativo

Tras el arranque del módulo de comunicación se intercambian todos los datos entre la CPU y el módulo de comunicación a través de instrucciones. La respuesta a la transición de estado operativo del módulo de comunicación depende del estado operativo de la CPU.

STOP de la CPU	Si hay un intercambio de datos entre módulo de comunicación y CPU se interrumpe, ya sea una petición de envío o de recepción, y el enlace se reinicia.
RUN de la CPU	En el estado RUN de la CPU es posible el intercambio de datos ilimitado, ya sea envío o recepción. En las primeras ejecuciones de instrucción tras rearranque de la CPU se sincronizan el módulo de comunicación y la instrucción en cuestión.
Rearranque de la CPU	En caso de rearranque de la CPU, las peticiones de la CPU se inicializan, de modo que todas las peticiones en curso entre la CPU y el módulo de comunicación se interrumpen automáticamente.
Rearranque completo de la CPU	En caso de rearranque completo de la CPU, las peticiones se siguen procesando. Con la parametrización correspondiente en el cuadro de diálogo de propiedades del módulo se puede borrar automáticamente el búfer de recepción del módulo de comunicación en el arranque de la CPU.

Funciones de diagnóstico

Introducción

Las funciones de diagnóstico del módulo de comunicación permiten localizar los errores ocurridos con rapidez. Para ello existen varias posibilidades de diagnóstico:

Diagnóstico a través de los elementos de indicación del módulo de comunicación	Los elementos de indicación informan sobre el estado operativo o posibles estados de error del módulo de comunicación. Los elementos de indicación proporcionan una primera vista general sobre los errores internos y externos ocurridos, así como los errores específicos de interfaz.
Diagnóstico a través de la salida STATUS de las instrucciones	Las instrucciones disponen de una salida STATUS destinada al diagnóstico, que proporciona información sobre errores de comunicación entre el módulo de comunicación y la CPU. El parámetro STATUS se puede evaluar en el programa de usuario.
Diagnóstico a través del área de registro de errores SYSTAT [1]	Mediante la programación de la instrucción STATUS en el programa de usuario se puede consultar el estado de la interfaz. Leyendo SYSTAT se obtiene información detallada sobre errores/eventos ocurridos durante la comunicación entre el módulo de comunicación y la CPU correspondiente y el interlocutor acoplado a esa interfaz.
Diagnóstico sobre los números de error en el telegrama de respuesta	Si se trabaja con el protocolo para mainframe RK512 y se produce un error en un telegrama SEND o FETCH en el interlocutor, éste envía un telegrama de respuesta con un número de error en el 4º byte.
Diagnóstico a través del búfer de diagnóstico del módulo de comunicación	Todos los errores/eventos también se registran en el búfer de diagnóstico [2] del módulo de comunicación. Igual que ocurre con el búfer de diagnóstico de la CPU, la información del búfer de diagnóstico del CP relevante para el usuario también puede leerse en texto claro y comprensible.
Alarma de diagnóstico	El módulo de comunicación puede lanzar una alarma de diagnóstico en la CPU asignada a él. El módulo de comunicación pone a disposición 4 bytes de información de diagnóstico. Dicha información se evalúa a través del programa de usuario o leyendo el búfer de diagnóstico de la CPU.

[1] Solo CP 441-1 y CP 441-2

[2] El CP 340, los módulos ET 200S 1SI y la CPU31xC-2 PtP no tienen búfer de diagnóstico

8.1.4.7 Configurar estaciones PC

Información de interés para estaciones PC

Introducción

Los controladores basados en PC ofrecen la misma funcionalidad que las CPUs SIMATIC S7-300 o S7-400 (controladores por hardware).

La estación SIMATIC PC representa un PC o una estación OS que puede contener componentes SIMATIC tales como aplicaciones (p. ej., WinCC) para tareas de automatización.

Comparación: SIMATIC S7 con una estación PC SIMATIC

Para las estaciones PC se utilizan términos adicionales debido a las diferencias funcionales entre la estación PC SIMATIC y SIMATIC S7:

- La estación PC en el editor de dispositivos y redes es un dispositivo capaz de acoger componentes de hardware tales como CPs y software, tales como aplicaciones o servidores OPC.
- En analogía con SIMATIC S7 se muestran estaciones SIMATIC PC en forma de rack con los slots que contienen, en los que se disponen los módulos y aplicaciones. A diferencia de la configuración S7 SIMATIC, en la estación SIMATIC PC se trata de una disposición supuesta y no de una asignación real de slots.
- En la vista de redes aparece únicamente el dispositivo (CPU, CP, etc.), p. ej., en el caso de un SIMATIC S7-300. Los módulos que inserte en la vista de dispositivos no se mostrarán en la vista de redes y tampoco pueden ser editados en la vista de redes.
  En la estación PC, en la vista de redes, puede ver también todos los componentes que ha insertado en la vista de dispositivos de la estación PC. En la vista de redes o de dispositivos puede seleccionar los componentes y editarlos en la ventana de inspección.
- En la asignación virtual de slots, con el objetivo de una diferenciación terminológica, no se utiliza ninguna "Dirección" o número de slot, sino que se asigna un "Índice". El índice, al igual que un slot, puede admitir exactamente un objeto del catálogo de hardware.

Modo de operación necesario

Para la puesta en servicio y el funcionamiento de la estación PC se diferenciará entre distintas áreas de uso. Seleccione, según el área de uso, el correspondiente modo de operación:

- Operación de la programadora: Modo de operación predeterminado para la programadora (PG/PC) y la estación HMI. En este modo de operación se utiliza la estación PC para la programación y prueba de otros dispositivos.
- Funcionamiento configurado: Modo de operación en el que la estación PC está configurada para una tarea de automatización y en el que se comunica con otros sistemas de automatización.

Sinopsis: Configurar el hardware y las conexiones de una estación SIMATIC PC

Con el tipo de estación "Estación SIMATIC PC" se lleva a cabo la configuración completa de una estación PC. Los componentes ahí configurados también están a disposición para la configuración de conexiones. En una estación SIMATIC PC pueden configurarse los siguientes componentes:

- Software PLC WinLC (versión V4.5 o superior)
- Slot PLC CPU 41x-2 PCI (WinAC Slot 412 y WinAC Slot 416)
- Módulos de comunicación para el PC
- SIMATIC HMI WinCC y Protool Pro
- Aplicaciones (p. ej., en aplicaciones de usuario basadas en SAPI)
- Servidor OPC

Información adicional

Para más información sobe el concepto de las estaciones PC tenga en cuenta el manual de configuración "SIMATIC NET - Commissioning PC Stations".

Componentes software de una estación PC

Una aplicación PC contiene una serie de propiedades configurables que son aplicables al cargarlas en la estación PC. Existen dos casos distintos de aplicación con distinto efecto en la generación y edición de los datos de configuración:

- Servidos OPC como proveedor central de servicios de comunicación
- Aplicación estándar

Según esta diferenciación y los distintos servicios utilizados, configure propiedades específicas de estos componentes.

Servidos OPC como proveedor central de servicios de comunicación

Los clientes OPC utilizan la interfaz al servidor OPC para comunicarse con un dispositivo de automatización, p. ej., con un SIMATIC S7-300.

Para la adaptación del servidor OPC configure sus propiedades. Los parámetros aquí ajustados o modificados serán efectivos tras cargar los datos de configuración en la estación PC y tras el inicio del servidor OPC.

Podrá utilizar aquí los siguientes servicios de comunicación dependiendo del módulo insertado:

- todos los tipos de conexión
- Servicios DP (DPV0)

Aplicación estándar

Con las aplicaciones estándar podrá comunicarse con otras aplicaciones y dispositivos a través de su módulo.

Podrá configurar aquí los siguientes servicios de comunicación dependiendo del módulo insertado:

- Conexiones S7
- Servicios DP

Información adicional

Para más información sobre el significado y manejo de las aplicaciones PC y del servidor OPC tenga en cuenta los manuales "SIMATIC NET - Puesta en funcionamiento de estaciones PC" y "SIMATIC NET - Comunicación industrial".

Consulte también

Conexiones con estaciones PC (Página 611)

Reglas de slots para una estación PC

Introducción

Los componentes tales como CP, programas de usuario basado en PC para la configuración de las conexiones y el servidor OPC pueden insertarse en el slot (índice) 1 a 32. Las estaciones PC están sujetas a las reglas de slots.

Inserción de un servidor OPC

En las estaciones PC pueden insertarse servidores OPC para clientes OPC que accedan a variables de sistemas de automatización a través de conexiones configuradas:

- En una estación PC puede insertarse como máximo un servidor OPC para clientes OPC.
- Si ya hay un servidor OPC insertado en una estación PC, podrá asignársele un CP.

En lugar de un servidor OPC o de forma adicional a un servidor OPC podrá utilizar también las aplicaciones estándar.

Insertar componentes

Las siguientes reglas se aplican a la numeración de índice y la inserción de componentes en la estación PC:

- Los números de índices pueden asignarse del 1 al 32.
- El primer componente insertado recibe siempre el índice 1.
- El resto de componentes insertados reciben el siguiente número de índice libre.
- Al borrar un componente los números de índice permanecen inalterados para el resto de los componentes. De esta forma pueden aparecer huecos en la numeración de índice.
- Se permite la inserción de componentes entre los componentes ya disponibles.
- Para la inserción de componentes en una ocupación disponible rige lo siguiente:
  - Si existen números de índice libres entre dos componentes se asignará en primer lugar el primer índice libre.
  - Si no hay números de índice libres y disponibles entre dos componentes, se elevará en un número el índice de todos los demás componentes hasta el primer índice libre. De esta forma podrá sustituirse un componente con el índice 32 en el depósito de los módulos no insertados. Así puede ampliar las estaciones PC sin perder ajustes importantes por un hacinamiento de componentes.

Inserción de submódulos

WinLC RTX soporta como submódulo

- una tarjeta CP 5611/21 como máximo o una interfaz integrada CP 5611/21
- una interfaz PROFINET como máximo (CP 1616, CP 1604 u.o.)
- cuatro submódulos como máximo en total. En los cuatro submódulos puede tratarse de un número cualquiera de interfaces de comunicación CP 5613.

Cuando un componente tiene submódulos tienen validez las siguientes reglas:

- Un controlador contiene un número fijo de sub-slots libres.
- Los sub-slots libres se muestran en la visa de dispositivos gráfica y en forma de tabla.
- Los submódulos pueden insertarse únicamente en los sub-slots libres.
- Al borrar un submódulo surge un sub-slot libre.
- Los sub-slots no ocupan ningún número de índice.

Consulte también

Área de módulos no enchufados (Página 516)

Configurar estaciones PC online

Introducción

Cuando desee poner en funcionamiento por primera vez un módulo PC, deberá realizar una primera configuración. Tras esta primera configuración de los módulos PC, la estación PC se ajusta para la recepción de datos de configuración.

Puede realizar una primera configuración para una estación PC accesible online y cargar los datos de configuración en la estación PC configurada.

Nota

Cuando la estación PC no está accesible online, la primera configuración puede realizarse también con un archivo XDB (Página 1202).

Requisitos

Se encuentra en la vista de redes.

Procedimiento

Para configurar una estación PC online, proceda del siguiente modo:

1. Arrastre un dispositivo para una estación PC a la vista de red desde el catálogo hardware, en "Sistemas PC".
2. Seleccione la estación PC.
3. En la ventana de inspección, en "Propiedades > General", modifique los nombres de la estación PC según sus necesidades.

4. En la estación PC arrastre los demás componentes necesarios desde el catálogo hardware con la función Drag & Drop. Se insertará un componente insertado en el nuevo índice.

Nota

Puede modificar el índice de un componente seleccionado en la ventana de inspección bajo "Propiedades > General > Número de posición". De esta forma pueden aparecer huecos en la numeración de índice. Estos huecos no se mostrarán en la representación gráfica de la vista de dispositivo. Verá una visualización de los huecos en el área en forma de tabla de la Vista de dispositivos.

5. Seleccione la estación PC.
6. Realice la primera configuración y/o la carga de los datos de configuración mediante:
   - Para realizar la primera configuración de la estación PC seleccione, en el menú de contexto, la orden "Estación PC online".
     En el cuadro de diálogo "Configurar" podrá realizar distintos ajustes a su red local, los ordenadores accesibles y la estación PC meta.
   - Para cargar los datos de configuración seleccione, en el menú de contexto, la orden "Cargar en dispositivo".
     La carga de los datos de configuración requiere una primera configuración ya realizada de la estación PC.

Consulte también

Vista de dispositivos (Página 493)

Configurar estación PC offline

Introducción

Cuando desee poner en funcionamiento por primera vez un módulo PC, deberá realizar una primera configuración. Tras esta primera configuración de los módulos PC, la estación PC se ajusta para la recepción de datos de configuración.

La primera configuración de una estación PC puede realizarse offline. Los datos para la primera configuración y el funcionamiento se guardarán como base de datos para la estación PC en un archivo XDB.

Nota

Cuando la estación PC es accesible online puede realizarse la primera configuración y transferencia de de datos de configuración también de forma directa (Página 1201).

Requisitos

Se encuentra en la vista de redes.

Procedimiento

Para configurar una estación PC offline a través de un archivo XDB, proceda del siguiente modo:

1. Arrastre un dispositivo para una estación PC a la vista de red desde el catálogo hardware, en "Sistemas PC".
2. Seleccione la estación PC.
3. En la ventana de inspección, en "Propiedades > General", modifique los nombres de la estación PC según sus necesidades.
4. En la estación PC arrastre los demás componentes necesarios desde el catálogo hardware con la función Drag & Drop. Se insertará un componente insertado en el nuevo índice.

   Nota

   Puede modificar el índice de un componente seleccionado en la ventana de inspección bajo "Propiedades > General > Número de posición". De esta forma pueden aparecer huecos en la numeración de índice. Estos huecos no se mostrarán en la representación gráfica de la vista de dispositivo. Verá una visualización de los huecos en el área en forma de tabla de la Vista de dispositivos.

5. Seleccione la estación PC.
6. En la ventana de configuración, en "Propiedades > Configuración XDB", active la casilla opcional "Generar archivo XDB".
   En el archivo XDB se guardan los datos de conexión y direcciones para CP y aplicaciones.
7. Establezca la ruta de archivo para los datos XDB.
8. Seleccione la estación PC.
9. Seleccione el comando "Traducir > ..." del menú contextual.
   Al guardar y traducir la configuración de una estación PC se generan datos de sistema y archivos de configuración XDB que podrá cargar y que deberá instalar en el sistema meta.

   Nota

   El lugar del archivo de configuración se establecerá en la estación PC meta a través del programa "Ajustar interfaz PG-PC" (en la ficha, "Configuración STEP 7").

Consulte también

Vista de dispositivos (Página 493)

Crear archivo Memory Card

Con este comando de menú podrá establecer un nuevo archivo de Memory Card.

Procedimiento

Para crear un archivo Memory Card, proceda del modo siguiente:

1. En el menú "Proyecto > Archivo Memory Card" seleccione la opción "Nuevo".
   Se abre un cuadro de diálogo.
2. Seleccione el directorio deseado.
3. Introduzca un nombre de archivo.
4. Confirme con "Crear"

Resultado

En la navegación del proyecto se mostrará un archivo Memory Card.

Podrá copiar bloques de archivos en la Memory Card cuando los mueva o los copie de otras carpetas de bloques, bibliotecas o de una Memory Card S7.

Abrir archivo existente

Para mostrar un archivo existente de Memory Card en la navegación del proyecto, seleccione la opción "Abrir" en el menú "Proyecto > Archivo Memory Card".

Particularidades del S7-modular Embedded Controller

Introducción y procedimiento general

Introducción

S7-modular Embedded Controller (S7-mEC) son PCs en forma de S7-300 y tienen el software preinstalado y preconfigurado:

- Sistema operativo Windows Embedded Standard
- WinAC RTX (F)
- SIMATIC NET
- WinCC Runtime (sólo en EC31-HMI/RTX)

Interfaces Ethernet del S7-modular Embedded Controller

Un S7-modular Embedded Controller tiene dos interfaces integradas Industrial Ethernet:

- La interfaz X1 PN LAN P1 / X1 PN LAN P2 puede configurarse como una interfaz PROFINET.
- La interfaz X2 P1 PN/IE LAN está asignada en IE General a la estación PC en el índice 3 y preconfigurada para la comunicación Industrial Ethernet.

Nota

Si se accede al S7-mEC a través del protocolo Remote Desktop, debe asegurarse de que el S7-mEC y la programadora o el PC de ingeniería estén en la misma subred Ethernet.

Para el acceso a S7-mEC a través del protocolo Remote Desktop debe utilizarse la conexión de red IE General (generalmente, la interfaz X2 P1 PN/IE LAN).

Configuración y programación básicas del S7-mEC

Puesto que los S7-mEC son sistemas preconfigurados, proceda del siguiente modo:

1. Cree un proyecto.
2. Inserte una estación PC.
3. Configure el hardware.
4. Cree un programa S7.
5. Configure la visualización en WinCC en EC31-HMI/RTX.
6. Descargue los datos de configuración en WinLC RTX EC (descarga).

EC31-HMI/RTX: configurar la visualización

Sólo en el EC31-HMI/RTX

El proyecto de visualización para el EC31-HMI/RTX se crea en el PC de ingeniería con SIMATIC WinCC.

Requisitos

- WinCC está instalado en el PC de ingeniería.
- Hay un objeto de WinCC RT insertado en la estación PC.

Procedimiento

Para crear un proyecto de visualización para el EC31-HMI/RTX proceda del modo siguiente:

1. Abra la vista de dispositivos de la estación PC del EC31-HMI/RTX.
2. Seleccione el objeto de WinCC RT.

3. Elija el comando "Abrir objeto".
   Se inicia WinCC.
4. Cree y configure los elementos necesarios: imágenes, avisos, variables, etc. con la ayuda de los editores de WinCC.
5. Guarde y pruebe la visualización.
6. Transfiera el proyecto al sistema de destino EC31-HMI/RTX.

Cargar datos de configuración en el sistema de destino

Posibilidades de carga

Los siguientes datos de configuración de STEP 7 pueden cargarse en el Embedded Controller:

- toda la estación PC (p. ej. IE General inclusive)
- sólo los datos de configuración de WinLC RTX / WinLC RTX F
- sólo el programa S7 de WinLC RTX / WinLC RTX F

En el EC31-HMI/RTX es necesario transferir los datos de configuración de WinCC flexible.

Requisitos

- El filtro de escritura avanzado (EWF) está desactivado en el Embedded Controller (si debe cargarse toda la estación PC).
- El Embedded Controller está conectado a la programadora o el PC vía Industrial Ethernet. La interfaz PG/PC está ajustada a TCP/IP.
- **Sólo EC31-HMI/RTX:** El proyecto de visualización está creado en WinCC.
- Para las funciones siguientes, el Embedded Controller debe estar conectado al sistema de ingeniería con un puerto de la interfaz X1 PN LAN:
  - Asignar nombres de dispositivos PROFINET IO
  - PROFINET CBA: Descarga de interconexiones y funciones online

Procedimiento

1. Desactive EWF en el EC31-RTX / EC31-RTX F / EC31-HMI/RTX (si todavía no está desactivado). Tras desactivar dicho filtro debe desconectarse y volver a conectarse el Embedded Controller.
2. Inicie WinLC.
3. Active la estación PC en el configurador de componentes (botón "Activar estación").
4. Ponga el Embedded Controller al estado operativo STOP.
5. Cargue la configuración hardware, el programa S7 y los datos de configuración en el sistema de destino.

Preparar para el funcionamiento

Para preparar el EC31-RTX / EC31-HMI/RTX para el funcionamiento, proceda del siguiente modo:

1. Ponga el filtro de escritura avanzado para la unidad C:\ en estado "activado".

ATENCIÓN
Fallo de la memoria flash Si deja el filtro de escritura avanzado en estado "desactivado", el acceso de escritura continuo del sistema operativo puede provocar un fallo prematuro de la memoria flash. Al activar al filtro de escritura avanzado se minimiza el número de accesos de escritura a la partición del sistema operativo y, con ello, se prolonga la vida útil de la memoria flash.

2. Documente todos los ajustes especiales (p. ej. cambios en las direcciones IP) que haya realizado al desarrollar la aplicación. Guárdelos junto con su proyecto.
3. Haga una copia de seguridad de los ajustes mediante un backup.

8.1.4.8 CM/CP S7-1200

CM/CP S7-1200

Telecontrol S7-1200

CP 1242-7

Copiar un CP 1242-7 en otro proyecto

Si copia un CP 1242-7 de un proyecto a otro, los siguientes parámetros se cambiarán en el grupo de parámetros "Identificación del CP" de la estación de destino:

- Número de proyecto del CP
- Número de estación del CP

Cargar en dispositivo

La función "Cargar en dispositivo" solo puede realizarse en el CP 1242-7 mediante una conexión TeleService del siguiente modo:

1. Seleccione el CP en STEP 7.
2. Seleccione el menú "Online" > "Cargar en dispositivo".
3. Seleccione la interfaz TeleService en el cuadro de diálogo visualizado "Carga avanzada".
4. Cargue los datos del proyecto desde el cuadro de diálogo "Carga avanzada".

Cargar de dispositivo

La función "Cargar de dispositivo" no es soportada por el CP 1242-7.

Configuración CP 1242-7

Modos de operación del CP GPRS

Modos de operación del CP

El CP 1242-7 permite que un S7-1200 pueda comunicarse como estación GPRS con una central u otras estaciones remotas a través de la red GSM. Para la comunicación a través de GPRS, el CP se ajusta a uno de los siguientes modos de operación:

- Telecontrol
  Este modo de operación del CP permite que la estación GPRS pueda intercambiar datos con los siguientes interlocutores:
  - Comunicación con el servidor Telecontrol
    Este modo de operación del CP permite que la estación GPRS intercambie datos con un servidor Telecontrol.
    El servidor Telecontrol es un PC conectado a Internet con la aplicación "TELECONTROL SERVER BASIC". Generalmente se encuentra en la central y sirve para vigilar y controlar las estaciones GPRS remotas. A través de la interfaz OPC integrada pueden intercambiarse datos con el cliente OPC de un sistema de control central.
    El PC del servidor Telecontrol no se configura en STEP 7. La aplicación "TELECONTROL SERVER BASIC" tiene una interfaz de configuración propia.
  - Comunicación con otra estación GPRS remota
    Los telegramas se transmiten a través del servidor Telecontrol.
  - Comunicación con una estación de ingeniería (para TeleService)

  La comunicación con el servidor Telecontrol se realiza a través de la red GSM y de Internet. Los requisitos para este modo de operación son disponer de una tarjeta SIM con servicio GPRS habilitado y de un servidor Telecontrol accesible para el CP.
- GPRS directo
  Este modo de operación del CP sirve para que las estaciones remotas puedan comunicarse directamente a través de la red GMS. No se requiere ningún servidor Telecontrol.
  Para poder acceder directamente a las estaciones en las redes públicas, es necesario acceder a ellas mediante una dirección fija. Para esto se utilizan tarjetas SIM con dirección IP fija que permiten direccionar las estaciones directamente.
  Los servicios de comunicación y las funciones de seguridad disponibles (p. ej. VPN) dependen de la oferta del proveedor de la red.
  Posibles interlocutores de la estación GPRS con el CP 1242-7 en el modo de operación "GPRS directo":
  - Un dispositivo accesible para el CP a través de una dirección IP (estación GPRS con CP 1242-7)
  - Una estación de ingeniería (para TeleService)

Establecimiento de conexión en la comunicación GPRS

Modos de conexión

- Modo de operación "GPRS directo"
  El modo de operación "GPRS directo" no incluye distintos modos de conexión.
- Modo de operación "Telecontrol"
  El CP puede configurarse para los siguientes modos de conexión.
  - Modo de conexión "permanente"
    Existe una conexión TCP permanente con el servidor de Telecontrol. Una vez se ha establecido la conexión se dispone de una conexión TCP permanente con el servidor de Telecontrol, incluso aunque no se transfieran datos continuamente.
  - Modo de conexión "temporal"
    Solo se establece una conexión con el servidor de Telecontrol en caso de necesidad.

Una vez establecida la conexión TCP, el envío de datos de proceso se produce en cuanto las instrucciones de Telecontrol se invoquen en la CPU.

Una conexión se establece siempre por medio del CP. Si se interrumpe una conexión establecida desde el CP, éste intenta automáticamente establecer de nuevo la conexión.

Activación del establecimiento de conexión para estaciones permanentes (modo de operación "Telecontrol")

En el modo de operación "Telecontrol" el establecimiento de la conexión permanente con el servidor de Telecontrol se lleva a cabo cuando arranca la estación. Si la conexión se interrumpe, el establecimiento puede iniciarse con un SMS de activación (vea a continuación).

Activación del establecimiento de conexión en el caso de estaciones temporales (modo de operación "Telecontrol")

Para las estaciones "temporales", el establecimiento de la conexión puede activarse con los siguientes eventos:

- Evento en la CPU local, que debe evaluarse obligatoriamente en el programa.
  En el programa se distinguen dos casos:
  - Los eventos que provocan un establecimiento de conexión único (p. ej. alarmas o comandos de un operador).
  - Expiración de un intervalo, que provoca un establecimiento de conexión cíclico (p. ej. una vez al día para la transmisión de datos)
- Solicitud desde un interlocutor (cliente OPC o estación S7)
  La solicitud del interlocutor provoca un establecimiento de conexión.
- Solicitud desde una estación de ingeniería para TeleService
  La solicitud, en la que el servidor de Telecontrol o la gateway de TeleService ejercen de intermediarios, no tiene que ser evaluada por el programa.
- SMS de activación del servidor de Telecontrol
  El SMS de activación puede lanzarse espontáneamente en el servidor de Telecontrol. También es posible configurar un envío cíclico en el servidor de Telecontrol.

- Llamada de activación de un teléfono
  La llamada de activación puede enviarse desde un teléfono cuyo número esté autorizado en el proyecto STEP 7. El teléfono debe soportar la función CLIP (transferencia del propio número).
  Se inicia el establecimiento de conexión con el servidor de Telecontrol (principal).
- SMS de activación de un teléfono
  El SMS de activación puede enviarse desde un teléfono cuyo número esté autorizado en el proyecto STEP 7. El teléfono debe soportar la función CLIP (transmisión del propio número de llamada) y la transmisión de SMS.
  Se inicia el establecimiento de conexión con el servidor de Telecontrol especificado en el SMS.

Al activar una estación temporal se transfieren todos los datos si éstos han sido modificados desde la última transferencia de datos.

Activación del establecimiento de conexión en el modo de operación "GPRS directo"

En el modo de operación "GPRS directo" el establecimiento de la conexión se activa con los siguientes eventos:

- Evento en la CPU local, que se evalúa en el programa.
- Solicitud desde un interlocutor (no desde una estación de ingeniería)
  La solicitud del interlocutor, la cual viene incluida en el telegrama recibido se evalúa en el programa tan pronto como se llamen las instrucciones de Telecontrol.
- Solicitud desde una estación de ingeniería para TeleService
  La solicitud, en la que el servidor de Telecontrol o la gateway de TeleService ejercen de intermediarios, no tiene que ser evaluada por el programa.

Autorización de alarma a través de "Números de llamada autorizados"

El requisito para que el CP acepte un SMS es la autorización del interlocutor emisor mediante su número de llamada. Estos números de llamada se configuran en STEP 7 con el CP en la lista "Números de llamada autorizados".

Nota

"Números de llamada autorizados" en el proyecto STEP 7

- Un número de llamada aquí registrado autoriza al remitente que transfiere adicionalmente este número de llamada a iniciar el establecimiento de la conexión.
- Si en la lista se introduce un asterisco (*), el CP aceptará SMS de todos los remitentes.
- Si la lista está vacía, no será posible activar el CP para el establecimiento de conexiones.

SMS despertador

En función del tipo de conexión y del servidor que la activa o del servidor TeleService transmisor, en el SMS despertador debe transferirse el texto siguiente:

- Para enlaces Telecontrol:
  - Texto para el SMS de activación para establecer una conexión con el servidor de Telecontrol:
    `TELECONTROL`
  - Texto para el SMS de activación para establecer una conexión con el servidor de Telecontrol principal:
    `TELECONTROL MAIN`
  - Texto para el SMS de activación para establecer una conexión con el servidor de Telecontrol de sustitución:
    `TELECONTROL BACKUP`

  La configuración de los servidores de Telecontrol para el CP GPRS en STEP 7 se realiza en "Interfaz Telecontrol > Modo de operación > Servidor Telecontrol principal o sustitutivo".

  Nota

  Activación a través de un teléfono móvil

  - En un SMS de activación se puede utilizar uno de los textos arriba mencionados.
  - Tras una llamada de activación la estación se conecta siempre con el servidor de Telecontrol principal.

- Para conexiones de TeleService:
  - Texto para el SMS de activación para establecer una conexión a través del primer servidor de TeleService configurado:
    `TELESERVICE`
    o bien
    `TELESERVICE 1`
  - Texto para el SMS de activación para establecer una conexión a través del segundo servidor de TeleService configurado:
    `TELESERVICE 2`

  La configuración de los servidores de TeleService para el CP GPRS se realiza en STEP 7, en "Interfaz Telecontrol > Autorización TeleService > 1er o 2º servidor TeleService".

Redes GSM preferentes

Selección de las redes GSM preferentes

Para elegir las redes GSM a las que el CP 1242-7 debe conectarse preferentemente, se ofrecen las siguientes opciones:

- Conexión automática
  El CP se conecta a la red GSM del operador de redes contractual según las especificaciones de la tarjeta SIM de máxima prioridad. Si la conexión a la red contratada no se realiza correctamente, el CP se conecta a otras redes GSM con las que el operador contractual haya firmado contratos de roaming y cuyos datos de acceso estén almacenados en la tarjeta SIM.
- Sólo red contratada
  El CP se conecta únicamente a la red GSM del operador contractual cuya tarjeta SIM está insertada en el CP. Sin roaming.
- Red contratada y redes alternativas
  El CP se conecta a la red contratada como primera preferencia. Si la conexión a la red contratada no se realiza correctamente, el CP se conecta a otras redes GSM alternativas, que se incluirán en la Lista de operadores de red preferentes con prioridad descendente. Las redes alternativas se incluyen en la lista como "Public Land Mobile Network" (PLMN). PLMN se compone de Mobile Country Code (MCC) y Mobile Network Code (MNC). Ejemplo: 26276
  Este es el PLMN para la red de prueba de la empresa Siemens AG con MCC = 262 y MNC = 76.

Sincronización horaria, procedimiento NTP - Parametrizacíon

Procedimiento NTP (NTP: Network Time Protocol)

Con el procedimiento NTP, el módulo envía regularmente consultas de hora a uno o más servidores NTP. El CP selecciona la hora más exacta según las respuestas del servidor.

La ventaja de este procedimiento es la posibilidad de sincronizar la hora más allá de los límites de la subred.

En el procedimiento NTP se transfiere generalmente la hora UTC (Universal Time Coordinated). Esto equivale a la hora GMT (Greenwich Mean Time).

CP 1242-7 TeleService

TeleService a través de GPRS

Vía de comunicación en TeleService a través de GPRS

Con TeleService para estaciones S7 con CP 1242-7 a través de GPRS, la conexión siempre se establece a través de un mediador entre la estación de ingeniería y la estación S7 remota.

También pueden actuar como mediadores:

- Un servidor Telecontrol (y, en su caso, un servidor Telecontrol de sustitución)
  El servidor Telecontrol puede ser un PC independiente o bien puede estar instalado como aplicación "TELECONTROL SERVER BASIC" en la estación de ingeniería.
- Un servidor TeleService
  Un servidor TeleService se utiliza cuando no hay ningún servidor Telecontrol disponible, es decir, cuando el CP funciona en el modo de operación "GPRS directo".

El servidor Telecontrol o el servidor TeleService pueden estar conectados vía LAN o Internet con la estación de ingeniería, desde donde se llama la función TeleService.

Requisitos para TeleService a través de GPRS

- Un servidor Telecontrol o un servidor TeleService.
- El proyecto de STEP 7 con los datos necesarios para la comunicación GPRS

Indicaciones sobre la configuración

El servidor Telecontrol y el servidor TeleService no se configuran en STEP 7.

Consulte también

Establecimiento de conexión en la comunicación GPRS (Página 1209)

Modos de operación del CP GPRS (Página 1208)

Establecer la conexión con TeleService a través de GPRS

Establecimiento de conexión con TeleService a través de GPRS

El requerimiento de establecimiento de conexión se produce desde la estación de ingeniería y se transmite a la estación con un SMS despertador. El CP 1242-7 de la estación S7-1200 establece una conexión con la estación de ingeniería a través de la red GSM e Internet.

Inicio de TeleService a través de GPRS

TeleService se inicia a través de GPRS de la siguiente manera:

1. Dentro del proyecto y en la estación de ingeniería autorizada, seleccione la estación S7 remota con la que se desea establecer una conexión con TeleService a través de GPRS.
2. Abra el cuadro de diálogo "Establecer conexión online" utilizando alternativamente las siguientes opciones:
   - Botón "Establecer conexión online"
   - Menú contextual "Establecer conexión online" (botón derecho del ratón)
   - Menú "Online" > "Establecer conexión online"

   Se abre el cuadro de diálogo "Establecer conexión online".

3. Seleccione en la lista desplegable "Tipo de interfaz PG/PC" el tipo de interfaz "TeleService GPRS".
4. Seleccione en la lista desplegable "Interfaz PG/PC" la opción "GPRS TeleService Board" si ésta no aparece de forma automática.
5. Haga clic en el símbolo "Conectar" que se encuentra junto a la lista desplegable "Interfaz PG/PC".
   Se abre el cuadro de diálogo "Establecer conexión remota".
6. Realice los ajustes necesarios en el cuadro de diálogo "Establecer conexión remota".
   Para más detalles al respecto, consulte los tooltips en cascada de STEP 7.

Para un correcto establecimiento de la conexión son necesarios los siguientes datos:

Datos necesarios para el establecimiento de la conexión con la estación GPRS

En el cuadro de diálogo "Establecer conexión remota" deben indicarse los siguientes datos:

- Dirección IP o nombre DNS del servidor Telecontrol
- Número de puerto TCP del servidor Telecontrol o del router ADSL a través del cual se ha establecido la conexión entre la estación de ingeniería y la estación GPRS remota.
- Contraseña de servidor de la estación de ingeniería para su autenticación en el servidor Telecontrol
  Sólo es necesaria si se ha configurado una contraseña específica de grupo en la aplicación "TELECONTROL SERVER BASIC".
- Número de usuario de TeleService
  Véase la configuración del CP en STEP 7.
- Contraseña de TeleService
  Véase la configuración del CP en STEP 7.
- ID de acceso del CP
  Sólo es necesaria cuando la estación cuenta con varios CP 1242-7. Véase la configuración del CP en STEP 7.

Estado

Estados de conexión en TeleService a través de GPRS

En el cuadro de diálogo "Establecer conexión remota" pueden mostrarse los siguientes estados de conexión:

- No conectado
  No existe ninguna conexión con la estación GPRS remota. Todavía no se ha iniciado el establecimiento de conexión.

Después de iniciar el establecimiento de la conexión haciendo clic en el botón "Conectar", se mostrarán los siguientes estados de forma consecutiva si la conexión se establece con éxito:

- Conectar con el servidor Telecontrol
  La estación de ingeniería se conecta con el servidor Telecontrol.
- Esperar a la estación S7
  El SMS despertador se ha depositado en la estación remota. Esperar una respuesta de la estación.
- Autenticación en la estación S7
  La estación S7 ha establecido una conexión IP con la estación de ingeniería a través de GPRS e Internet y comprueba los datos de inicio de sesión y autenticación recibidos.
- Conectado
  La estación se ha conectado correctamente con la estación de ingeniería.

Si la conexión no se establece con éxito, pueden mostrarse los siguientes estados:

- Servidor Telecontrol no accesible
  Causas posibles:
  - Se ha interrumpido la conexión entre la estación de ingeniería y el servidor Telecontrol.
  - El servidor Telecontrol está desconectado.
- Contraseña del servidor incorrecta
  Causa: En el cuadro de diálogo se ha especificado una contraseña de servidor incorrecta para el inicio de sesión y la autenticación de la estación en el servidor Telecontrol.
- La estación S7 no da señal
  Causas posibles:
  - La comunicación GSM entre el servidor Telecontrol y la estación es defectuosa.
  - La conexión entre la red GSM e Internet es defectuosa.
  - La conexión a Internet es defectuosa.
  - El servidor Telecontrol no ha podido enviar un SMS despertador.
  - El CP no ha recibido nigún SMS despertador.
  - El remitente del SMS no se ha configurado en la lista de números de llamada despertador autorizados.
- Nombre de usuario de TeleService o contraseña de TeleService incorrecta
  Causas posibles:
  - En el cuadro de diálogo se ha introducido un nombre de usuario de TeleService o una contraseña de TeleService incorrectos para la autenticación en el CP 1242-7.
  - El nombre de usuario de TeleService o la contraseña de TeleService no se han configurado en STEP 7.
- Todos los puntos de acceso de TeleService están ocupados.
- El servidor Telecontrol no conoce el CP.
  Causa: El CP procede de un proyecto de STEP 7 que no se corresponde con el proyecto del servidor Telecontrol.

- No hay recursos para TeleService en el CP: Póngase en contacto con la línea directa de atención al cliente.
- Error de protocolo
  Causa: Telegrama incorrecto o telegrama de un dispositivo incorrecto. Póngase en contacto con la línea directa de atención al cliente.

CP 1243-1

Ciclo de muestreo de la CPU

Estructura del ciclo de muestreo de la CPU

El ciclo (incluida la pausa) con el que el CP muestrea el área de memoria de la CPU consta de las fases siguientes:

- **Peticiones de lectura con prioridad alta**
  Para puntos de datos del tipo "Input" (entradas) que se configuran en el ajuste "Alta prioridad" de "General > Prioridad en el ciclo de muestreo" en la configuración de puntos de datos, se leen todas las variables PLC en cada ciclo de muestreo.
- **Peticiones de escritura**
  En cada ciclo se escriben en la CPU los valores de un número determinado de peticiones de escritura espontáneas. El número de variables que se escriben en cada ciclo se especifica para el CP con el parámetro "Número máx. de peticiones de escritura" en el grupo de parámetros "Comunicación con la CPU". Las variables cuyo número excede este valor se escriben en el próximo ciclo o en uno de los siguientes.
- **Peticiones de lectura con prioridad baja - proporcionalmente**
  Para puntos de datos del tipo "Input" (entradas) que se configuran en el ajuste "Baja prioridad" de "General > Prioridad en el ciclo de muestreo" en la configuración de puntos de datos, los valores de sus variables PLC se leen proporcionalmente en cada ciclo de muestreo.
  El número de variables que se leen en cada ciclo se especifica para el CP con el parámetro "Número máx. de peticiones de lectura" en el grupo de parámetros "Comunicación con la CPU". Las variables que pasan de dicho valor y, por tanto, no se leen en un ciclo, se leerán en el próximo ciclo o en otro ulterior.
- **Tiempo de pausa del ciclo**
  Es el tiempo de espera entre dos ciclos de muestreo. Sirve para reservar tiempo suficiente para otros procesos que acceden a la CPU por medio del bus de fondo de la estación.

Duración del ciclo de muestreo de la CPU

Puesto que para el ciclo no es posible configurar un tiempo fijo y las diferentes fases no tienen asignado un número fijo de objetos, la duración del ciclo de muestreo es variable y puede cambiar dinámicamente.

Tipos de puntos de datos

Al configurar los datos de usuario que debe transferir el CP, se asigna cada punto de datos a un tipo de punto de datos específico del protocolo. Los tipos de datos compatibles con el CP están listados a continuación, junto con los tipos de datos S7 compatibles en cada caso. Están agrupados por formado (memoria necesaria).

El sentido hace referencia al sentido de la transferencia (sentido de observación = "in", sentido de control = "out").

CP 1243-1 DNP3: tipos soportados de puntos de datos

Tabla 8-88 Tipos soportados de puntos de datos, grupos de objetos DNP3, variantes y tipos de datos S7 compatibles

Formato (memoria necesaria)	Tipo de punto de datos	Grupo de objetos DNP3 [variations]	Sentido	Tipos de datos S7	Área de operandos
Bit	Binary Input	1 [1, 2]	in	BOOL	I, Q, M
	Binary Input Event	2 [1, 2]	in	BOOL	I, Q, M
	Binary Output [1)]	10 [2]	out		
	Binary Output Event [1)]	11 [1, 2]	out		
	Binary Command	12 [1]	out	BOOL	I, Q, M
Integer (16 bits)	Counter Static	20 [2]	in	UINT, WORD	I, Q, M
	Frozen Counter [2)]	21 [2, 6]	in		
	Counter Event	22 [2, 6]	in	UINT, WORD	I, Q, M
	Frozen Counter Event [3)]	23 [2, 6]	in		
	Analog Input	30 [1]	in	INT	I, Q, M
	Analog Input Event	32 [2, 4]	in	INT	I, Q, M
	Analog Output Status [4)]	40 [2]	out		
	Analog Output	41 [2]	out	INT	I, Q, M
	Analog Output Event [4)]	42 [2, 4]	out		
Integer (32 bits)	Counter Static	20 [1]	in	UDINT, DWORD	I, Q, M
	Frozen Counter [2)]	21 [1, 5]	in		
	Counter Event	22 [1, 5]	in	UDINT, DWORD	I, Q, M
	Frozen Counter Event [3)]	23 [1, 5]	in		
	Analog Input	30 [2]	in	DINT	I, Q, M
	Analog Input Event	32 [1, 3]	in	DINT	I, Q, M
	Analog Output Status [4)]	40 [1, 3]	out		
	Analog Output	41 [1]	out	DINT	I, Q, M
	Analog Output Event [4)]	42 [1]	out		
Número en coma flotante (32 bits)	Analog Input	30 [5]	in	REAL	Q, M
	Analog Input Event	32 [5, 7]	in	REAL	Q, M
	Analog Output Status [4)]	40 [3]	out		
	Analog Output	41 [3]	out	REAL	Q, M
	Analog Output Event [4)]	42 [5, 7]	out		

Formato (memoria necesaria)	Tipo de punto de datos	Grupo de objetos DNP3 [variations]	Sentido	Tipos de datos S7	Área de operandos
Número en coma flotante (64 bits)	Analog Input	30 [6]	in	LREAL	Q, M
	Analog Input Event	32 [6, 8]	in	LREAL	Q, M
	Analog Output	41 [4]	out	LREAL	Q, M
	Analog Output Event [4]	42 [6, 8]	out		
Bloque de datos (1...64 bytes) [5]	Octet String	110 [-]	in	[5]	I, Q, M
	Octet String Event [5]	111 [-]	in	[5]	I, Q, M

[1] Este grupo de objeto puede configurarse mediante el grupo de objeto 12 en el editor de puntos de datos de STEP 7.

[2] Este grupo de objeto puede configurarse mediante el grupo de objeto 20 en el editor de puntos de datos de STEP 7.

[3] Este grupo de objeto puede configurarse mediante el grupo de objeto 22 en el editor de puntos de datos de STEP 7.

[4] Este grupo de objeto puede configurarse mediante el grupo de objeto 41 en el editor de puntos de datos de STEP 7.

[5] Con estos tipos de puntos de datos pueden transferirse áreas de memoria relacionadas de hasta 64 bytes de tamaño. Son compatibles todos los tipos de datos S7 de 1 a 64 bytes de tamaño. En caso de que cambie posteriormente el array deberá volver a crearse el punto de datos.

Configuración y retroalimentación de valores analógicos y binarios

- Valores binarios
  Los valores binarios (entradas) se configuran mediante los grupos de objeto 1 y 2.
  Los grupos de objeto 10 y 11 sirven para retroalimentar los valores actuales de la memoria imagen de proceso de la CPU al sistema de control.
- Valores analógicos
  Los valores analógicos se configuran mediante los grupos de objeto 30, 32 y 41, según el sentido de transferencia.
  Los grupos de objeto 40 y 42 sirven para retroalimentar los valores actuales de la memoria imagen de proceso de la CPU al sistema de control.

CP 1243-1 IEC: tipos soportados de puntos de datos

Tabla 8-89 Tipos soportados de puntos de datos, tipos IEC y tipos de datos S7 compatibles

Formato (memoria necesaria)	Tipo de punto de datos	Tipo IEC	Sentido	Tipos de datos S7	Área de operandos
Bit	Single point information	<1>	in	BOOL	I, Q, M
	Single point information with time tag	<30>	in	BOOL	I, Q, M
	Single command	<45>	out	BOOL	I, Q, M
Byte	Step position information	<5>	in	BYTE	I, Q, M
	Step position information with time tag	<32>	in	BYTE	I, Q, M
	Regulating step command	<47>	out	BYTE	I, Q, M
Integer (16 bits)	Measured value, normalized value	<9>	in	INT	I, Q, M
	Measured value, normalized value with time tag	<34>	in	INT	I, Q, M
	Measured value, scaled value	<11>	in	INT	I, Q, M
	Measured value, scaled value with time tag	<35>	in	INT	I, Q, M

Formato (memoria necesaria)	Tipo de punto de datos	Tipo IEC	Sentido	Tipos de datos S7	Área de operandos
	Set point command, normalised value	<48>	out	INT	I, Q, M
	Set point command, scaled value	<49>	out	INT	I, Q, M
Integer (32 bits)	Bitstring of 32 bits	<7>	in	DWORD, UDINT	I, Q, M
	Bitstring of 32 bits with time tag	<33>	in	DWORD, UDINT	I, Q, M
	Integrated totals	<15>	in	DWORD, UDINT	I, Q, M
	Integrated totals with time tag	<37>	in	DWORD, UDINT	I, Q, M
Número en coma flotante (32 bits)	Measured value, short floating point number	<13>	in	REAL	Q, M
	Measured value, short floating point number with time tag	<36>	in	REAL	Q, M
	Set point command, short floating point number	<50>	out	REAL	Q, M
Bloque de datos (1...32 Bit) [1]	Bitstring of 32 bits [1]	<7>	in	[1]	I, Q, M
	Bitstring of 32 bits with time tag [1]	<33>	in	[1]	I, Q, M
	Bitstring of 32 bits [1]	<51>	out	[1]	I, Q, M

[1] Con estos tipos de puntos de datos pueden transferirse áreas de memoria relacionadas de hasta 32 bits de tamaño. Son compatibles todos los tipos de datos S7 de 1 a 32 bits de tamaño. En caso de que cambie posteriormente el array deberá volver a crearse el punto de datos.

Tipos de transferencia y clases de eventos

Almacenamiento y transferencia de valores

Los valores de puntos de datos pueden asignarse a diferentes áreas de memoria del CP y, por tanto, a diferentes tipos de transferencia.

- Valores estáticos
  Los valores estáticos se introducen en la memoria imagen de proceso del CP.
  El valor de un punto de datos estático se envía tras una consulta del interlocutor de la comunicación.
- Eventos
  Los valores de puntos de datos que están configurados como evento también se introducen en la memoria imagen de proceso del CP.
  Adicionalmente, los valores de eventos se introducen en el búfer de transmisión del CP.
  El valor de un punto de datos configurado como evento se envía espontáneamente al interlocutor de la comunicación cuando esta función está habilitada por parte del maestro.
  El búfer de transmisión tiene el tamaño siguiente:
  - En DNP3: máx. 64000 eventos
  - En IEC: máx. 65535 eventos

  La capacidad del búfer de transmisión se reparte equitativamente entre todos los interlocutores activados.

Tipo de transferencia

Dependiendo del tipo de CP pueden seleccionarse los siguientes tipos de transferencia:

- Transferencia tras llamada
  El valor actual del punto de datos en cada caso se introduce en la memoria imagen del CP. Un valor nuevo sobrescribe al anterior en la memoria imagen.
  Tras una llamada del interlocutor de la comunicación se transfiere el valor actual en ese momento.
- Todos los valores disparados
  Siempre que se cumple la condición de disparo se introduce el valor individual del punto de datos por separado en el búfer de transmisión y se transfiere al interlocutor de la comunicación.
- Clase de evento ...
  El valor se introduce en el búfer de transmisión disparado como evento de la clase 1, 2 o 3. Si el maestro ha habilitado la transferencia espontánea, el valor se transfiere espontáneamente.
  Encontrará los detalles de las clases de eventos en los apartados siguientes.
- Valor actual disparado
  Solo se introduce en el búfer de transmisión y se transfiere al interlocutor de la comunicación el valor actual de un punto de datos en el momento del disparo.

Clases de eventos en el protocolo DNP3

Los datos de proceso de las diferentes clases de eventos reciben el tratamiento siguiente:

- Clase de evento 1 / Clase de evento 2
  Clases según el protocolo DNP3: Event, class 1 / class 2
  Cada cambio de valor se introduce en el búfer de transmisión en orden cronológico.
  El maestro debe evaluar la clasificación (1 o 2).
- Clase de evento 3
  Clase según el protocolo DNP3: Static event, class 3
  So se introduce en el búfer de transmisión y sobrescribe el último valor guardado allí el valor actual en el momento en que se cumple la condición de disparo.

Clases de eventos en el protocolo IEC

Los datos de proceso de las diferentes clases de eventos reciben el tratamiento siguiente:

- Clase de evento 1
  Clases según el protocolo IEC: Event, class 1
  Cada cambio de valor se introduce en el búfer de transmisión en orden cronológico.
- Clase de evento 3
  Clase según el protocolo IEC: Static event, class 3
  So se introduce en el búfer de transmisión y sobrescribe el último valor guardado allí el valor actual en el momento en que se cumple la condición de disparo.

Preprocesamiento de valores analógicos

Algunos CPs tienen un preprocesamiento de valores analógicos con el que se soportan las funciones descritas a continuación, entre otras.

Factor de filtrado

Los valores analógicos que sufren oscilaciones rápidas pueden estabilizarse utilizando la función de filtrado.

Los factores de filtrado se calculan siguiendo la fórmula siguiente, igual que en los módulos de entradas analógicas S7.

$$y_n = \frac{x_n + (k-1)y_{n-1}}{k}$$

siendo
y_n = valor filtrado en el ciclo actual n
x_n = valor captado en el ciclo actual n
k = factor de filtrado

Los valores siguientes pueden configurarse como factor de filtrado para el módulo.

- 1 = sin filtrado
- 4 = filtrado débil
- 32 = filtrado medio
- 64 = filtrado fuerte

Tiempo de supresión de errores

Un valor analógico en el rango de desbordamiento por exceso (32767 / $7FFF_h$) o defecto (-32768 / 8000_h) no se transfiere mientras dure el tiempo de supresión de errores. Lo mismo es válido para entradas Life Zero. Una vez transcurrido el tiempo de supresión de errores se transferirá el valor en el rango de desbordamiento por exceso o defecto, siempre que siga pendiente.

Si el valor vuelve a entrar en el rango asignado antes de que transcurra el tiempo de supresión de errores se transferirá el valor actual inmediatamente.

Un caso típico de aplicación para este parámetro es la supresión de valores de corriente de pico al arrancar motores potentes que, en caso de no hacerse, se notificarían como fallo al punto de control.

La supresión está ajustada a valores analógicos que son captados como valores brutos por los módulos de entradas analógicas S7. Dichos módulos suministran para todas las áreas de entrada los valores citados para el rango de desbordamiento por exceso y defecto, incluso para entradas Life Zero.

Si en el área de marcas o en un bloque de datos se preparan valores listos preprocesados por la CPU, solo es posible o aconsejable suprimir las perturbaciones cuando dichos valores también adoptan los valores citados de (32767 / $7FFF_h$) o (-32768 / 8000_h) en el rango de desbordamiento por exceso o defecto, respectivamente. En caso contrario, no debería activarse el parámetro para valores preprocesados.

Cálculo del valor medio

Con este parámetro se transfieren valores analógicos captados como valores medios.

El valor actualmente pendiente para un punto de datos de valor analógico se registra y totaliza cíclicamente. El número de valores captados por unidad de tiempo depende del ciclo de lectura de la CPU y del ciclo de muestreo de la CPU para el CP. A partir de los valores totalizados se calcula el valor medio en cuanto se lanza la transferencia por medio de un disparo de tiempo. A continuación se reinicia la totalización para calcular el próximo valor medio.

El valor medio también se calcula cuando la transferencia del telegrama de valores analógicos es lanzada por una consulta del interlocutor. En este caso, la duración del período de cálculo es el tiempo que transcurre entre la última transferencia (p. ej. lanzada por el disparo) y el instante de la consulta. Después de esta transferencia también se reinicia la totalización para calcular el próximo valor medio.

En caso de que el valor analógico captado esté en el rango de desbordamiento por exceso (32767 / $7FFF_h$) o defecto (-32768 / 8000_h), existe la posibilidad de tomar dicho valor en consideración inmediatamente o bien ocultarlo por un tiempo determinado para el cálculo del valor medio. El comportamiento deseado en cada caso se determina con el parámetro "Tiempo de supresión de errores":

- Tiempo de supresión de errores = 0
  En cuanto se capta un valor en el rango de desbordamiento por exceso o defecto se cancela inmediatamente el cálculo del valor medio. Para el período de cálculo en curso, el valor 32767 / $7FFF_h$ o -32768 / 8000_h se guarda como valor medio no válido y se transfiere la próxima vez que se lanza el telegrama de valores analógicos. Posteriormente se inicia un nuevo cálculo del valor medio. Si el valor analógico sigue estando en el rango de desbordamiento por exceso o defecto, también se guarda inmediatamente como valor medio no válido y se transfiere con el próximo lanzamiento del telegrama.
- Tiempo de supresión de errores > 0
  Si el valor analógico captado está en el rango de desbordamiento por exceso o defecto, los valores erróneos no se incluyen en el cálculo del valor medio, concretamente mientras dure el tiempo de supresión de errores. Una vez excedido dicho tiempo, el valor 32767 / $7FFF_h$ o -32768 / 8000_h se guarda como valor medio no válido y se transfiere la próxima vez que se lanza el telegrama de valores analógicos. El procedimiento es el mismo en cada nuevo período de cálculo, es decir, los valores erróneos se suprimen mientras dure el tiempo de supresión de errores.

La duración del tiempo de supresión de errores permite determinar indirectamente el porcentaje de valores no válidos por período de cálculo. Si el valor medio se calcula en un ciclo de 15 minutos y el tiempo de supresión de errores está ajustado a 5 minutos, por ejemplo, esto significa que el valor medio solo se transferirá como no válido cuando en el período de cálculo en curso haya más de un 33 % de valores analógicos captados en el rango de desbordamiento por exceso o defecto.

8.1.4.9 SCALANCE X, W y M

SCALANCE X

Información importante

VLAN

Definición de red independiente de la ubicación de las estaciones

VLAN (Virtuelles Local Area Network) divide una red física en varias redes lógicas separadas entre sí. Se forman grupos lógicos de dispositivos. Solo pueden direccionarse entre sí las estaciones de la misma VLAN. Dado que también los telegramas Multicast y Broadcast solo pueden enviarse dentro de la VLAN correspondiente, se habla de dominios Broadcast.

De aquí resulta, como ventaja especial de las VLAN, una menor carga de la red para las estaciones o segmentos de red de otras VLAN.

Para identificar qué paquete está asignado a qué VLAN, el telegrama se amplía con 4 bytes (VLAN Tagging (Página 1224)). Esta ampliación contiene también información de prioridad, además de la VLAN ID.

Posibilidades de asignación a VLAN

Hay diferentes posibilidades de asignación a VLAN:

- VLAN basada en puerto
  A cada puerto de un dispositivo se le asigna una VLAN ID. Una VLAN basada en puerto se configura en "Layer 2 > VLAN > Port-based VLAN".
- VLAN basada en protocolo
  A cada puerto de un dispositivo se le asigna un grupo de protocolos. Una VLAN basada en protocolo se configura en "Layer 2 > VLAN > Protocol Based VLAN Port".
- VLAN basada en subred
  A la dirección IP del dispositivo se le asigna una VLAN ID. Una VLAN basada en subred se configura en "Layer 2 > VLAN > Ipv4 Subnet Based VLAN".

Procesamiento de la asignación VLAN

Si en el dispositivo hay creadas varias asignaciones VLAN, estas se procesan en el siguiente orden:

1. VLAN basada en subred
2. VLAN basada en protocolo
3. VLAN basada en puerto

En el telegrama se comprueba primero la dirección IP. Si en la ficha "Ipv4 Subnet Based VLAN" se cumple una regla, el telegrama se envía a la VLAN correspondiente. Si no se cumple ninguna regla, se comprueba el tipo de protocolo del telegrama. Si en la ficha "Protocol Based

VLAN Port" se cumple una regla, el telegrama se envía a la VLAN correspondiente. Si no se cumple ninguna regla, el telegrama se envía a través de la VLAN basada en puerto. Las reglas para la VLAN basada en puerto se establecen en la ficha "Port Based VLAN".

Consulte también

General (Página 1310)

GVRP (Página 1312)

VLAN basada en puerto (Página 1313)

Grupo VLAN basado en protocolo (Página 1314)

Puerto VLAN basado en protocolo (Página 1315)

VLAN basada en subred Ipv4 (Página 1315)

VLAN Tagging

Extensión de los telegramas Ethernet en cuatro byte

Para CoS (Class of Service, priorización de telegramas) y VLAN (red virtual) se ha fijado en la norma IEEE 802.1 Q la extensión de los telegramas Ethernet con el identificador VLAN.

Nota

El identificador VLAN provoca un aumento de la longitud total permitida del telegrama de 1518 a 1522 bytes. En los IE Switches, el tamaño MTU estándar es de 1536 bytes. El tamaño MTU puede modificarse en valores entre 64 y 9216 bytes.
Hay que comprobar si las estaciones terminales de la red pueden procesar estas longitudes y este tipo de telegrama. De no ser así, a esas estaciones sólo se les deberán enviar telegramas de longitud estándar.

Los 4 bytes adicionales se encuentran en el encabezado del telegrama Ethernet, entre la dirección de origen y el campo de tipo / longitud Ethernet:

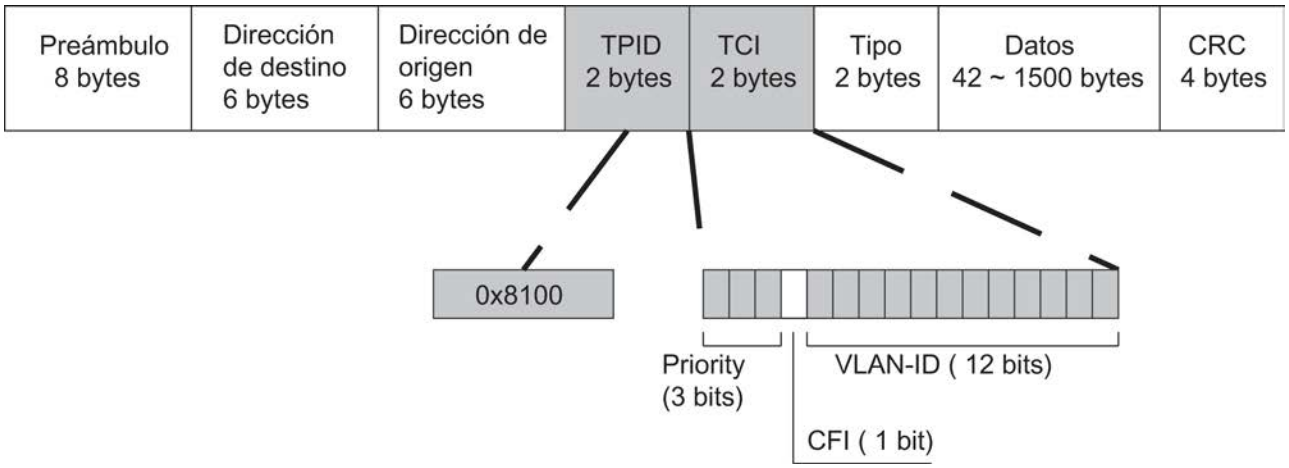


Figura 8-48 Estructura del telegrama Ethernet ampliado

Los bytes adicionales contienen el Tag Protocol Identifier (TPID) y la Tag Control Information (TCI).

Tag Protocol Identifier (TPID)

Los primeros 2 bytes forman el Tag Protocol Identifier (TPID) y están ocupados de forma fija con 0x8100. Este valor indica que el paquete de datos contiene información VLAN o indicaciones de prioridad.

Tag Control Information (TCI)

Los 2 bytes de la Tag Control Information (TCI) contienen la siguiente información:

CoS– Priorización

En el telegrama con tag hay 3 bits para la prioridad, que también se denominan Class of Service (CoS). La priorización según IEEE 802.1p es la siguiente:

Bits CoS	Tipo de datos
000	Transmisión de datos no crítica en cuanto al tiempo (less then best effort [ajuste básico])
001	Transmisión de datos normal (best effort [segundo plano])
010	Reservado (estándar)
011	Reservado (excellent effort)
100	Transmisión de datos con un retardo máx. de 100 ms
101	Servicio garantizado, multimedia interactivo
110	Servicio garantizado, transmisión de voz interactiva
111	Reservado

La priorización de los paquetes de datos presupone una cola de espera en los componentes, en la que puede guardar temporalmente los paquetes de datos de menor prioridad.

El dispositivo dispone de ocho colas de espera paralelas, en las que se procesan los telegramas de diferente prioridad. Primero se procesan los telegramas de mayor prioridad

(procedimiento "Strict Priority"). Este procedimiento garantiza que los telegramas de mayor prioridad se envíen en cualquier caso, incluso si la cantidad de datos es elevada.

Canonical Format Identifier (CFI)

El CFI posibilita la compatibilidad entre Ethernet y Token Ring.
Los valores tienen el siguiente significado:

Valor	Significado
0	El formato de la dirección MAC es canónico. En la representación canónica de la dirección MAC primero se transmite el bit de menor valor. Ajuste estándar para switches Ethernet.
1	El formato de la dirección MAC no es canónico.

VLAN ID

En el campo de datos de 12 bits se pueden formar hasta 4095 VLAN IDs. En ese caso rigen las siguientes especificaciones:

VLAN ID	Significado
0	El telegrama sólo contiene información sobre la prioridad (Priority Tagged Frames), pero no contiene un identificador VLAN válido.
1 - 4094	Identificador VLAN válido, el telegrama está asignado a una VLAN, puede contener también información sobre la prioridad.
4095	Reservado

SNMP

Introducción

El Simple Network Management Protocol (SNMP) permite vigilar y controlar elementos de red desde una estación central, p. ej. routers o switches. El SNMP regula la comunicación entre los dispositivos vigilados y la estación de vigilancia.

Tareas de SNMP:

- Vigilancia de componentes de red
- Control remoto y parametrización remota de componentes de red
- Detección y notificación de errores

Las versiones v1 y v2 de SNMP no disponen de mecanismos de seguridad. Todos los usuarios de la red pueden acceder a los datos con el software adecuado e incluso modificar parametrizaciones.

Para controlar derechos de acceso de forma sencilla sin aspectos de seguridad se utilizan Community Strings.

El Community String se transmite junto con la petición. Si el Community String es correcto, el agente SNMP responde y envía los datos solicitados. Si el Community String no es correcto, el agente SNMP rechaza la petición. Para los derechos de lectura y escritura se definen diferentes Community Strings. Los Community Strings se transmiten en texto claro.

Valores estándar de los Community Strings:

- public
  dispone solo de derechos de lectura
- private
  dispone de derechos de lectura y escritura

Nota

Por razones de seguridad, no deben utilizarse los valores estándar "public" ni "private". Modifique estos valores después de la primera instalación.

Otros mecanismos de protección sencillos a nivel de dispositivo:

- Allowed Host
  El sistema vigilado conoce las direcciones IP de los sistemas que vigilan.
- Read Only
  Si se asigna "Read Only" a un dispositivo vigilado, las estaciones de vigilancia podrán leer los datos, pero no modificarlos.

Los paquetes de datos SNMP no están codificados y es muy fácil leerlos.

La estación central también se denomina estación de gestión. En los dispositivos que deben vigilarse se instala un agente SNMP, con el que la estación de gestión intercambia datos.

La estación de gestión envía paquetes de datos del tipo siguiente:

- GET
  Solicitar un registro del agente
- GETNEXT
  Llama el próximo registro.
- GETBULK (disponible a partir de SNMPv2)
  Solicita varios registros al mismo tiempo, p. ej. varias filas de una tabla.
- SET
  Contiene datos de parametrización para el dispositivo correspondiente.

El agente SNMP envía paquetes de datos del tipo siguiente:

- RESPONSE
  El agente devuelve los datos solicitados por el administrador.
- TRAP
  Cuando se produce un evento determinado, el agente SNMP envía traps.

SNMPv1, SNMPv2 y SNMPv3 utilizan UDP (User Datagram Protocol). Los datos se describen en una Management Information Base (MIB).

SNMP v3

En comparación con las versiones anteriores SNMP v1 y SNMP v2, SNMP v3 incorpora un amplio concepto de seguridad.

SNMP v3 soporta:

- Autenticación de usuarios completamente codificada
- Codificación de todo el tráfico de datos
- Control de acceso de los objetos MIB a nivel de usuario/grupo

Spanning Tree

Evitar la formación de bucles en conexiones redundantes

El procedimiento Spanning Tree permite crear estructuras de red en las que existan varias conexiones entre dos estaciones. Un Spanning Tree impide que se formen bucles en la red, permitiendo solo una ruta y desactivando los demás puertos (redundantes) para el tráfico de datos. En caso de interrupción, los datos se pueden enviar por una ruta alternativa. La funcionalidad del método Spanning Tree se basa en el intercambio de telegramas de configuración y modificación de la topología.

Definición de la topología de red mediante telegramas de configuración

Para calcular la topología, los dispositivos intercambian entre sí telegramas de configuración, los llamados BPDU (Bridge Protocol Data Unit). Con estos telegramas se selecciona el Root Bridge y se crea la topología de red. Además, los telegramas BPDU provocan el cambio de estado de los puertos Root.

El Root Bridge es el Bridge (puente) que controla el procedimiento Spanning Tree de todos los componentes implicados.

Una vez que el Root Bridge está fijado, cada dispositivo determina un puerto Root. Este es el puerto con los menores costes de encaminamiento al Root Bridge.

Comportamiento en caso de cambios de la topología de la red

Si se incorporan estaciones a una red o se retiran de ella, esto puede tener repercusiones en la elección de la vía óptima para los paquetes de datos. Para tener en cuenta estos cambios, el Root Bridge envía mensajes de configuración a intervalos regulares. El tiempo que transcurre entre dos mensajes de configuración se puede ajustar con el parámetro "Hello Time".

Actualidad de la información de configuración

Con el parámetro "Max Age" se define la antigüedad máxima de las informaciones relativas a configuración. Si un Bridge recibe información de configuración más antigua de lo que se ha definido con Max Age, rechaza este mensaje e impulsa un nuevo cálculo de las vías.

Sin embargo, un Bridge no utiliza información de configuración nueva inmediatamente, sino sólo después del tiempo fijado en el parámetro "Forward Delay". Así se garantiza que el funcionamiento con la nueva topología no se inicie hasta que todos los Bridges tengan las informaciones necesarias.

RSTP, MSTP, CIST

Rapid Spanning Tree Protocol (RSTP)

Una desventaja del STP es que la red debe reconfigurarse si sufre alguna perturbación o falla un dispositivo: Los dispositivos comienzan a negociar nuevas rutas cuando se produce la interrupción. Esta operación dura hasta 30 segundos. Por esta razón se ha ampliado el STP para desarrollar el "Rapid Spanning Tree Protocol" (RSTP, IEEE 802.1w). Este se diferencia fundamentalmente del STP en que los dispositivos ya recopilan información sobre rutas alternativas cuando el funcionamiento es correcto, de modo que no deben esperar a que se produzca un fallo para adquirir estos datos. El tiempo de reconfiguración de una red controlada con RSTP se reduce así a unos pocos segundos.
Esto se logra con las siguientes funciones:

- Edge Ports (puerto de estación terminal)
  Los Edge Ports son puertos que están conectados a un dispositivo terminal.
  Un puerto que está definido como Edge Port se activa directamente después del establecimiento de una llamada. Si en un Edge Port se recibe una Spanning Tree BPDU, el puerto pierde su papel de Edge Port y participa de nuevo en (R)STP. Si no se recibe ninguna BPDU más una vez transcurrido un período de tiempo (3x Hello Time), el puerto pasa de nuevo al estado de Edge Port.
- Punto a punto (comunicación directa entre dos dispositivos vecinos)

  Por acoplamiento directo de los dispositivos se puede realizar una modificación del estado (reconfiguración del puerto) sin retardo.
- Puerto alternativo (en sustitución del Root Port)

  Se ha configurado un sustituto para el Root Port. En caso de pérdida de conexión con el Root Bridge, el dispositivo puede establecer así sin retardo, por reconfiguración, una conexión a través del puerto alternativo.
- Reacción a eventos

  Rapid Spanning Tree reacciona sin retardo a eventos, por ejemplo, a una interrupción de la conexión. Por lo tanto no es necesario esperar las señales de relojes, como en el caso de Spanning Tree.
- Número máximo de saltos de puente
  Número máximo de saltos de puente que un paquete puede efectuar antes de invalidarse automáticamente.
  Así, con Rapid Spanning Tree básicamente se preconfiguran alternativas para muchos parámetros o bien se consideran determinadas propiedades de la topología de red para acortar el tiempo de reconfiguración.

Multiple Spanning Tree Protocol (MSTP)

El Multiple Spanning Tree Protocol (MSTP) es una evolución del Rapid Spanning Tree Protocol. Entre otras cosas, ofrece la posibilidad de utilizar varias instancias RSTP dentro de diferentes VLAN o grupos de VLAN, lo cual permite, p. ej., disponer dentro de determinadas VLAN de rutas que el Rapid Spanning Tree Protocol sencillo bloquearía globalmente para el tráfico de datos.

Common and internal Spanning Tree (CIST)

CIST es un concepto de Multiple Spanning Tree. CIST designa la instancia utilizada internamente por el switch, que se asemeja en principio a una instancia RSTP interna.

Función Routing

Introducción

El término Routing describe el establecimiento de vías (rutas) para la comunicación entre diferentes redes, es decir, cómo debe llegar el paquete de datos desde la subred A a la subred B.

SCALANCE X soporta las siguientes funciones Routing:

- Routing estático
  En el routing estático las rutas se introducen manualmente en la tabla de Routing.
- Redundancia del router
  Con un VRRP (Virtual Router Redundancy Protocol) estandarizado, la disponibilidad de las pasarelas importantes aumenta gracias a routers redundantes.
- Routing dinámico
  Las entradas de la tabla de Routing son dinámicas y se actualizan de forma continua. Las entradas se generan con el protocolo Routing.
  - OSPF v2 (Open Shortest Path First)

Routing estático

La ruta se introduce manualmente en la tabla de Routing. La ruta se introduce en la tabla de Routing en la página WBM "Routes".

Redundancia del router con VRRP

El Virtual Router Redundancy Protocol (VRRP) permite detectar el fallo de un router en una red.

Varios routers VRRP de un segmento se unen en un grupo lógico formando un router virtual (VR). El grupo se define mediante la ID virtual (VRID). En el grupo la ID debe ser igual. Esta VRID no puede utilizarse para otros grupos.

Al router virtual se le asignan una dirección IP virtual y una dirección MAC virtual. Uno de los routers VRRP del grupo se establece como router maestro. El router maestro tiene la prioridad 255. Los demás routers VRRP son routers de reserva. El router maestro asigna a su interfaz de red las direcciones IP y MAC virtuales. El router maestro envía paquetes VRRP (Advertisements) a los routers de reserva en intervalos determinados. Con ellos señaliza que "sigue vivo". El router maestro responde también a las consultas ARP.

Si el router maestro virtual falla, un router de reserva adopta las funciones del maestro. El router de reserva con mayor prioridad se convierte en router maestro. Si la prioridad de los routers de reserva es la misma, decide la dirección MAC más alta. El router de reserva se convierte en el nuevo router maestro virtual.

El nuevo router maestro virtual adopta las direcciones MAC e IP virtuales. Con ello no es necesario actualizar las tablas de Routing o las tablas ARP. De este modo se minimizan las consecuencias del fallo de un dispositivo.

VRRP se configura en "Layer 3 > VRRP".

OSPFv2

Routing dinámico con OSPF v2

OSPF (Open Shortest Path First) es un protocolo de routing basado en los costes. Para calcular la ruta más corta y rentable se utiliza el algoritmo Short Path First de Dijkstra. El OSPF fue desarrollado por la IETF (Internet Engineering Task Force).
OSPFv2 se configura en "Layer 3 > OSPFv2".

OSPF v2 divide un sistema autónomo (AS) en diferentes áreas.

Áreas del OSPF

Existen las siguientes áreas:

- Backbone
  El área Backbone es el área 0.0.0.0. A ella están conectadas todas las áreas restantes. El área Backbone puede estar conectada a otras áreas directamente o a través de conexiones virtuales.
  En el área Backbone se encuentra toda la información de routing. Así, esta área es la responsable de transmitir la información entre diferentes áreas.
- Stub Area
  Esta área contiene las rutas dentro de su área, dentro del área autónoma y la ruta estándar de salida del sistema autónomo. Los destinos situados fuera del sistema autónomo se asignan a la ruta estándar.
- Totally Stubby Area
  Esta área solo contiene las rutas dentro de su área y la ruta estándar para salir del área.
- Not So Stubby Area (NSSA)
  Esta área puede transmitir paquetes de otros sistemas autónomos a las áreas del propio sistema autónomo (redistribute). Los paquetes se distribuyen desde el router NSSA.

Router del OSPF

El OSPF distingue entre los siguientes tipos de router:

- Router interno (IR)
  Todas las interfaces OSPF del router están asignadas a la misma área.
- Area Border Router (ABR)
  Las interfaces OSPF del router están asignadas a distintas áreas. Una interfaz OSPF está asignada al área Backbone. Si es posible, las rutas se agrupan.

- Backbone Router (BR)
  Al menos una de las interfaces OSPF está asignada al área Backbone.
- Autonomous System Area Border Router (ASBR)
  Una interfaz del router está conectada a otro AS, p. ej. un AS que utiliza el protocolo de routing RIP.

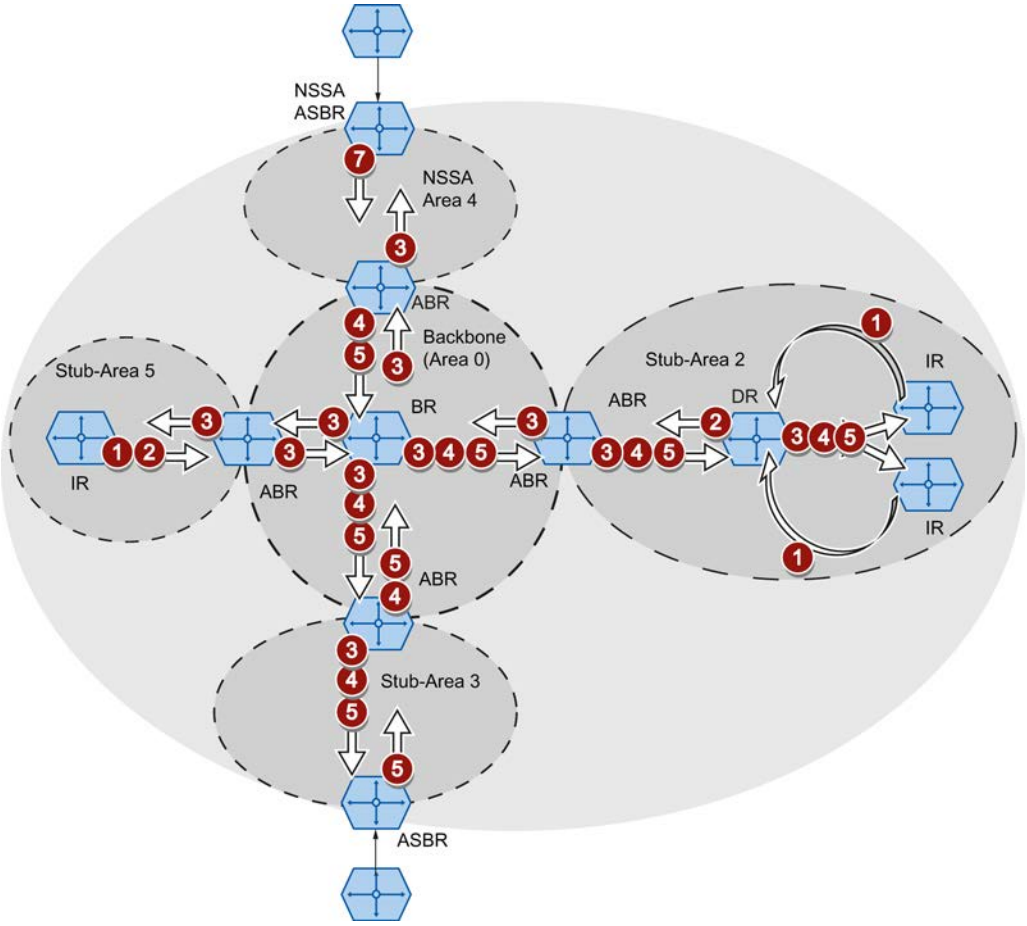
Conexión virtual

Cada área debe estar conectada físicamente al área Backbone. En algunos casos no es posible realizar una conexión física. El router remoto se conecta a un router Backbone mediante una conexión virtual. En el área por la que pasa el enlace virtual no se ha designado ningún router o router de reserva.

Tipos de LSA

Dentro del sistema autónomo se intercambian paquetes, que contienen información sobre las conexiones de un router y el aviso de estado de la conexión. Los paquetes también se conocen como LSAs (Link State Advertisments). Los LSAs se envían siempre del router al router vecino.

Si hay cambios en la red, los LSAs se envían a todos los routers de la red. La información depende del tipo de LSA.



1. **Router LSA (tipo de LSA 1)**
   El tipo de LSA 1 solo se envía dentro de un área. Para cada conexión activa del router que forma parte del área observada se genera un tipo de LSA 1. El tipo de LSA 1 contiene información sobre el estado y los costes de la conexión, p. ej. dirección IP, máscara de red y tipo de red.
2. **Network LSA (tipo de LSA 2)**
   El tipo de LSA 2 solo se envía dentro de un área. Para cada red que forma parte del área observada, el router genera un tipo de LSA 2. Si en una red hay varios routers interconectados, el router designado (DR) envía el tipo de LSA 2. El tipo de LSA 2 contiene entre otros la dirección de red, la máscara de red y una lista de los routers conectados a la red.
3. **Summary LSA (tipo de LSA 3 / tipo de LSA 4)**
   El Summary LSA es generado por el Area Border Router y se envía dentro del área. El Summary LSA contiene información sobre rutas que están fuera del área pero dentro del AS. Si es posible, las rutas se agrupan.
   - Summary LSA (tipo de LSA 3)
     El tipo de LSA 3 describe las rutas hacia las redes y propaga la ruta estándar por las áreas.
   - AS Summary LSA (tipo de LSA 4)
     El tipo de LSA 4 describe las rutas hacia los ASBR.
5. **External LSA (tipo de LSA 5 / tipo de LSA 7)**
   El External LSA es generado por el ASBR. El tipo de LSA depende del área.
   - AS External LSA (tipo de LSA 5)
     El tipo de LSA 5 es enviado por el AS Border Router a las áreas del sistema autónomo, excepto a las áreas Stub y NSSA. El LSA contiene información sobre las rutas hacia una red de otro AS. Las rutas se han creado manualmente o bien se han aprendido externamente. El ASBR utiliza el tipo de LSA 5 para distribuir rutas estándar al área Backbone.
   - NSSA External LSA (tipo de LSA 7)
     El tipo de LSA 7 es generado por el AS Border Router de un NSSA. Dicho router también se denomina NSSA ASBR. El tipo de LSA 7 se envía únicamente dentro del NSSA. Si en el tipo de LSA 7 el bit P = 1, el ABR convierte estos LSAs en el tipo de LSA 5 y los envía al área Backbone.

Establecer una vecindad

El router pasa por los estados siguientes para establecer una conexión con el router vecino.

1. Attempt state / Init state
   El router activa OSPF y empieza a enviar y recibir paquetes Hello. Por medio de los paquetes Hello recibidos, el router averigua los routers OSPF que están a su lado. El router comprueba el contenido del paquete Hello. El paquete Hello incluye también una lista de los routers vecinos (Neighbor Table) del "remitente".
2. Two way state
   Si, p. ej., la ID del área, el tipo de área y los ajustes de los temporizadores coinciden, es posible establecer una conexión (adjacencies) con el vecino. En una red punto a punto, la conexión se admite directamente. Si en una red debe accederse a varios routers vecinos, los paquetes Hello determinan el router designado (DR) y el router de reserva designado (DBR). El router con mayor prioridad es el router designado. Si dos routers tienen la misma prioridad, será router designado el que tenga la menor ID de router. El router establece una conexión con el router designado.
3. Exchangestart state
   Los routers vecinos establecen el router con el que se inicia la comunicación. El router con mayor ID se convierte en maestro.
4. Exchange state
   Los routers vecinos envían paquetes que describen el contenido de su base de datos de vecindad. La base de datos de vecindad (Link state database - LSDB) contiene información sobre la topología de la red.
5. Loading state
   El router completa la información recibida. Si el router todavía tiene dudas sobre el estado de una conexión concreta, envía una consulta (Link State Request). El router vecino envía una respuesta (Link State Update). La respuesta incluye un LSA correspondiente. El router confirma la recepción de la respuesta (Link State Acknowledge).
6. Full State
   El intercambio de información con el router vecino ha finalizado. La base de datos de vecindad de los routers vecinos es la misma. Utilizando el algoritmo Short Path First, el router calcula una ruta hacia cada destino. La ruta se introduce en la tabla de Routing.

Comprobar la vecindad

Los paquetes Hello no se utilizan únicamente para establecer relaciones de vecindad. Puesto que se envían cíclicamente, los paquetes Hello sirven también para comprobar la conexión con el router vecino. Si pasado un intervalo determinado (Dead Intervall) no se recibe ningún paquete Hello, la conexión con el vecino se marca como "fallida". Las entradas correspondientes se borran.

Actualizar la base de datos de vecindad

Una vez se ha creado la base de datos de vecindad, cuando se producen cambios en la topología se envían LSAs a todos los routers de la red.

RIPv2

Routing dinámico con RIPv2

El Routing Information Protocol (RIPv2) se utiliza para crear automáticamente tablas de routing. RIPv2 se emplea en sistemas autónomos (AS) con un máximo de 15 routers. Se basa en el algoritmo de vector de distancias.

RIPv2 fue desarrollado por la IETF (Internet Engineering Task Force) y está descrito en RFC 2453.
RIPv2 se configura en "Layer 3 > RIPv2".

Crear una tabla de routing

Puesto que, en un principio, un router solo conoce las redes que están directamente conectadas a él, envía una consulta a sus routers vecinos. La respuesta que recibe contiene las tablas de routing de los routers vecinos. Basándose en la información obtenida, el router crea una tabla de routing propia.

Dicha tabla contiene entradas para todos los destinos posibles. Cada entrada incluye la distancia hasta el destino y el primer router de la ruta.

La distancia también se denomina métrica. Indica el número de routers por los que se pasará siguiendo una ruta hasta el destino (Hop Count). La distancia máxima permitida es de 15 routers (Hops).

Actualizar la tabla de routing

Una vez se ha creado la tabla de routing, el router envía su tabla de routing a todos los routers vecinos en intervalos de 30 segundos.

El router compara la información de routing nueva con la tabla de la que dispone. Si la información nueva contiene rutas más cortas, se sobrescriben las rutas existentes. El router solo conserva la ruta más corta hasta un destino.

Comprobar un router vecino

Si un router no recibe mensajes de un router vecino durante más de 180 segundos, lo marca como no válido. El router asigna al router vecino la métrica 16.

Link Aggregation

Link Aggregation

Con la Link Aggregation o agregación de enlaces se agrupan varios enlaces físicos que transcurren en paralelo y que tienen la misma velocidad de transmisión en un único enlace lógico que ofrece mayor velocidad de transmisión. Este procedimiento conforme con IEEE 802.3ad también se denomina Port Trunking o Channel Bundling.

Link Aggregation funciona sólo con enlaces dúplex que tengan la misma velocidad de transmisión y funcionen en modo punto a punto. Así se consigue ampliar el ancho de banda

o, lo que es lo mismo, aumentar la velocidad de transmisión. Si falla una parte del enlace, el tráfico de datos se gestiona a través de las partes restantes.

Para el control se utiliza el Link Aggregation Control Layer (LACL) y el Link Aggregation Control Protocol (LACP).

Autenticación

Método de autenticación

Es posible configurar puerto a puerto los métodos de autenticación "802.1x" y "Autenticación MAC" así como la opción "Guest VLAN".

Las funciones están sometidas a un orden jerárquico. Si están activadas las tres funciones, primero se intenta autenticar el terminal mediante "802.1x". Si la autenticación no es correcta, se inicia "Autenticación MAC". Si esta tampoco funciona, el terminal se habilita para la comunicación en "Guest VLAN". "Solo es posible utilizar "Guest VLAN" si hay por lo menos un método de autenticación activo.

Ambos métodos de autenticación dependen del terminal. Si el terminal soporta EAP (Extensible Authentication Protocol) puede autenticarse utilizando el método "802.1x". Si el terminal no soporta EAP puede autenticarse utilizando "Autenticación MAC". En este caso, el IE Switch asume la función del terminal y utiliza la dirección MAC del dispositivo como parámetro de autenticación.

802.1x

El método de autenticación "802.1x" funciona del siguiente modo:
Un terminal que soporta EAP envía información de autenticación al IE Switch. El IE Switch transmite la información al servidor de autenticación. El servidor de autenticación comprueba la información y permite o deniega al terminal el acceso a la red.

Autenticación MAC

El método de autenticación "Autenticación MAC" funciona del siguiente modo:
En cuanto el IE Switch recibe un telegrama de un terminal, envía una consulta al servidor RADIUS para permitir o denegar el acceso del terminal a la red.

Editar propiedades y parámetros

Posibilidades de edición

Existen las posibilidades siguientes para editar las propiedades y los parámetros:

- Editor de hardware y redes
  Una vez se ha insertado el componente de red, existe la posibilidad de editar las propiedades y los parámetros, por ejemplo el nombre del dispositivo. Encontrará más información al respecto en "Editor de hardware y redes".
- Web Based Management (WBM)
  Es posible acceder a los parámetros y las propiedades utilizando páginas HTML (páginas del WBM) suministradas. Cada página del WBM tiene una página de ayuda propia que describe las propiedades y los parámetros. Encontrará más información en el manual de configuración "SCALANCE XM400/X-500 Web Based Management". Encontrará el manual de configuración en Siemens Industry Automation and Drives Service & Support en Internet, en la ID de artículo 67428305 (http://support.automation.siemens.com/WW/view/es/67428305).
- Command Line Interface
  La CLI permite especificar todos los ajustes para la configuración del dispositivo. La CLI ofrece las mismas posibilidades que el Web Based Management (WBM). Encontrará más información en el manual de configuración "SCALANCE XM400/X-500 Command Line Interface". Encontrará el manual de configuración en Siemens Industry Automation and Drives Service & Support en Internet, en la ID de artículo 67430663 (http://support.automation.siemens.com/WW/view/es/19440762).

Disponibilidad

La disponibilidad de los ajustes depende

- del tipo de puerto
- del modo de configuración

SCALANCE X distingue entre los siguientes tipos de puerto:

- Puerto Switch
- Router-Port

Existen los siguientes modos de configuración:

- Configuración offline
  La primera configuración de una estación PC puede realizarse offline. En este modo solo están disponibles los ajustes que no requieren conectarse al dispositivo.
- Modo de configuración online
  Cuando hay conexión con el dispositivo, la ventana de inspección presenta páginas adicionales. Dichas páginas incluyen la indicación "Esta página solo está disponible cuando hay conexión online con el dispositivo". Todos los ajustes incluyen el complemento "Solo disponible online".

Crear y borrar una entrada

A modo de ejemplo se creará y borrará una entrada en el cliente Syslog. El procedimiento es el mismo en todas las páginas.

Crear una entrada

1. Seleccione el dispositivo en la vista de redes o dispositivos.
2. Abra las propiedades del dispositivo en la ventana de inspección.
3. En la ventana de inspección, vaya a Sistema > Cliente Syslog.
4. Introduzca la dirección IP del servidor Syslog en Dirección IP.
5. Haga clic en la tabla.
6. Elija la entrada "Nueva entrada" del menú contextual.
   En la tabla se crea una nueva entrada.

Borrar una entrada

1. Seleccione la entrada deseada en la tabla.
2. Elija la entrada "Borrar" del menú contextual.

Botones utilizados con frecuencia

- **Actualizar la visualización con "Actualizar"**
  Las páginas que muestran parámetros actuales incluyen el botón "Actualizar" en el borde inferior. Haga clic en este botón si desea solicitar datos actuales del dispositivo para la página visualizada.
- **Guardar ajustes para todos los puertos con "Aplicar a tabla"**
  Las páginas en las que pueden configurarse varios puertos tienen 2 tablas. En la primera tabla se realizan los ajustes de todos los puertos, que se aplican a la segunda tabla. En la última columna de la primera tabla hay el botón "Aplicar a tabla". Haga clic en el botón para guardar los ajustes introducidos para todos los puertos.

Asignación de direcciones IP

Posibilidades de configuración

En el estado de suministro y tras restablecer la configuración de fábrica, el dispositivo no dispone de dirección IP.

Existen las siguientes posibilidades para asignar una dirección IP al dispositivo:

- DHCP (ajuste estándar)
- Primary Setup Tool
  Consulte el manual de configuración "Primary Setup Tool" para más información. Encontrará el manual de configuración en Siemens Industry Automation and Drives Service & Support en Internet, en la ID de artículo 19440762 (http://support.automation.siemens.com/WW/view/es/19440762).

- STEP 7
- CLI vía interfaz serie
  Consulte el manual de configuración "SCALANCE X500 Command Line Interface" para más información. Encontrará el manual de configuración en Siemens Industry Automation and Drives Service & Support en Internet, en la ID de artículo 67430663 (http://support.automation.siemens.com/WW/view/es/19440762).

Requisitos

- En las propiedades del dispositivo está activada la opción "Ajustar dirección IP en el dispositivo". Encontrará más información en "Direccionamiento de dispositivos PROFINET".

SCALANCE XR500 como dispositivo IO: "Compilar" y "Cargar en dispositivo"

En el caso de un SCALANCE XR500 configurado como dispositivo PROFINET IO y asignado a un controlador IO, al ejecutar las funciones "Compilar" y "Cargar en dispositivo" solo se cargan en el switch los datos que también pueden configurarse en Web Based Management (WBM) (Layer 2, Layer 3, System, Security).

Si desea ejecutar las funciones "Compilar" o "Cargar en dispositivo" para los datos de dispositivo PROFINET IO de XR500, seleccione antes el controlador IO asignado.

Información

Versiones

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra las versiones de hardware y software del dispositivo.

Descripción

La tabla 1 se divide en las siguientes columnas:

- **Hardware**
  Muestra el dispositivo utilizado.
- **Nombre**
  Indica el nombre del dispositivo o del módulo.
- **Versión**
  Indica la versión de hardware del dispositivo.
- **Referencia**
  Indica la referencia del dispositivo o del módulo descrito.

La tabla 2 se divide en las siguientes columnas:

- Software
  - Firmware
    Indica la versión actual de firmware. Cuando se haya cargado un archivo de firmware nuevo y el dispositivo todavía no se haya reiniciado, aquí se mostrará la versión de firmware del archivo de firmware que se ha cargado. Tras el siguiente reinicio se activa y utiliza el firmware que se ha cargado.
  - Bootloader
    Indica la versión del software de arranque que está guardado en el dispositivo.
- Descripción
  Muestra la descripción breve del software.
- Versión
  Muestra el número de versión del software instalado.
- Fecha
  Indica la fecha de creación del software instalado.

I&M

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página contiene datos del fabricante y de mantenimiento específicos del dispositivo, como la referencia, el número de serie, los números de versión, etc. En esta página no es posible realizar ajustes.

Descripción de los valores mostrados

La tabla se divide en las siguientes filas:

- ID de fabricante
  Muestra la identificación del fabricante.
- Referencia
  Muestra la referencia.
- Número de serie
  Muestra el número de serie.
- Versión de hardware
  Muestra la versión de hardware.
- Versión de software
  Muestra la versión de software.
- Contador de versión
  Muestra el contador de cambios de versión: contador de cambios de versión desde la primera puesta en marcha.
- Fecha de actualización
  Fecha y hora del último cambio de versión

- Identificador de la instalación
  Muestra el identificador de la instalación del dispositivo. El identificador de la instalación se establece en la fase de configuración del dispositivo con HW Config de STEP7.
- Identificador de ubicación
  Muestra el identificador de ubicación del dispositivo. El identificador de ubicación se establece en la fase de configuración del dispositivo con HW Config de STEP7.

Tabla ARP

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Asignación de dirección MAC y dirección IP

Por medio del Address Resolution Protocol (ARP) se asignan unívocamente las direcciones MAC a direcciones IPv4. Cada dispositivo de red guarda dicha asignación en su propia tabla ARP. La página muestra la tabla ARP del dispositivo.

Valores visualizados

La tabla se divide en las siguientes columnas:

- Interfaz
  Muestra la interfaz a través de la cual se aprendió la fila.
- Dirección MAC
  Muestra la dirección MAC del dispositivo de destino.
- Dirección IP
  Muestra la dirección IP del dispositivo de destino.
- Tipo de medio
  Muestra el tipo de conexión.
  - Dinámico
    El dispositivo ha reconocido de forma automática los datos de dirección.
  - Estático
    Las direcciones fueron registradas como direcciones estáticas.

Tabla de registro

Protocolización de eventos

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

El dispositivo ofrece la posibilidad de documentar los eventos que se produzcan; estos se pueden definir, en parte, en "Sistema > Eventos". Se puede documentar, por ejemplo, el momento en que falló un intento de autenticación o en que cambió el estado de conexión de un puerto. El contenido de la tabla de registro de eventos se conserva también tras desconectar el dispositivo.

Ajustes

- Filtro de severidad
  Las entradas de la tabla se pueden filtrar por grado de gravedad. Para visualizar todas las entradas, active o desactive todos los ajustes.
  - Info
    Información (6)
  - Warning
    Advertencias (4)
  - Critical
    Crítico (2)

Valores visualizados

La tabla se divide en las siguientes columnas:

- Reinicio
  Cuenta el número de rearranques completos desde la última vez que se restablecieron los ajustes de fábrica e indica tras qué rearranque del dispositivo se ha producido cada evento concreto.
- Tiempo de funcionamiento del sistema
  Indica el tiempo de funcionamiento del dispositivo desde el último rearranque, en el que se produjo el evento descrito.
  Cuando está ajustada la hora del sistema, también se muestra la hora a la que se ha producido el evento.
- Hora del sistema
  Muestra la fecha y la hora del dispositivo.
- Severidad
  Indica el grado de gravedad.
- Aviso de registro
  Muestra una descripción abreviada del evento que se ha producido.

Botones

- **Vaciar**
  Haga clic en este botón si desea borrar el contenido del archivo de registro de eventos. Con esta acción, también se vaciará la visualización. El contador de reinicios solo se reinicializará cuando se reinicie el dispositivo tras el restablecimiento de los ajustes de fábrica.

Nota

El número de entradas de esta tabla está limitado a 400. Cuando se alcanza este número, se sobrescriben las entradas más antiguas. La tabla se guarda de forma permanente en la memoria.

Error

Estado de error

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra los errores que se produzcan. Si no quedan mensajes de error por responder, se apaga el LED de error.

El cálculo de tiempo empieza después del último inicio de sistema. Al reiniciar el sistema se genera una entrada nueva con el tipo de inicio efectuado en la memoria de errores.

Descripción de los valores mostrados

- **Número de errores notificados**
  Muestra el número de errores notificados.

La tabla contiene las siguientes columnas:

- **Instante del error**
  Indica el tiempo de funcionamiento del dispositivo desde el último rearranque, en el que se produjo el error descrito.
- **Descripción del error**
  Visualización del estado de error para el dispositivo.
- **Borrar estado de error**
  Los errores que pueden borrarse tienen activado el botón "Borrar estado de error".

Redundancia

Spanning Tree

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra la información actual sobre Spanning Tree y los ajustes del Root-Bridge.

- Si Spanning Tree está desactivado se muestra únicamente información básica del dispositivo.
- Cuando Spanning Tree está activado se muestra lo siguiente:
  - parámetros de la instancia seleccionada
  - parámetros de los puertos de la instancia seleccionada

  La información depende del modo Spanning Tree seleccionado.

Valores visualizados

- **Modo Spanning Tree**
  Indica el modo ajustado. El modo se define en "Layer 2 > MSTP > General".
  Son posibles los siguientes valores:
  - '-'
  - STP
  - RSTP
  - MSTP
- **ID de instancia**
  Indica el número de la instancia. Este parámetro depende del modo configurado.
- **Prioridad Bridge / Prioridad Root**
  A partir de la prioridad Bridge se determina qué dispositivo pasa a ser Root-Bridge. El Bridge (puente) con la prioridad más alta (es decir, con el valor menor para este parámetro) es el Root-Bridge. Si en una red hay varios dispositivos con la misma prioridad, pasa a ser Root-Bridge el dispositivo cuya dirección MAC tenga el valor numérico más bajo. Ambos parámetros, prioridad de Bridge y dirección MAC, forman juntos el identificador Bridge. Dado que el Root-Bridge administra todos los cambios de ruta, debería estar dispuesto lo más central posible, debido al tiempo de ciclo de los telegramas. El valor de la prioridad Bridge es un múltiplo entero de 4096.
- **Dirección Bridge / Dirección Root**
  La dirección Bridge indica la dirección MAC del dispositivo y la dirección Root muestra la dirección MAC del Root-Bridge.
- **Costes Root**
  Muestra los costes de ruta desde el dispositivo hasta el Root-Bridge.
  En modo MSTP se muestran los costes de ruta hasta el Root-Bridge de CIST.

- Estado de Bridge
  Indica el estado del Bridge, p. ej. si el dispositivo es el Root-Bridge.
- Prioridad Root regional(solo disponible con MSTP)
  Véase la descripción en Prioridad Bridge / Prioridad Root.
- Dirección Root regional(solo disponible con MSTP)
  Indica la dirección MAC del dispositivo.
- Costes Root regionales (solo disponible con MSTP)
  Muestra los costes de la ruta desde el dispositivo hasta el Root-Bridge regional.

La tabla se divide en las siguientes columnas:

- Puerto
  Indica el puerto a través del cual se comunica el dispositivo.
- Prioridad
  Indica el estado del puerto. Son posibles los siguientes valores:
  - Disabled
    El puerto se ha eliminado manualmente del Spanning Tree, que ya no lo tiene en cuenta.
  - Designated
    Los puertos que salen del Root-Bridge.
  - Alternate
    El puerto con una ruta alternativa a un segmento de red
  - Backup
    Si un switch tiene varios puertos que van al mismo segmento de red, el "peor" puerto se convierte en reserva.
  - Root
    El puerto que ofrece la mejor ruta hacia el Root-Bridge.
  - Master
    Este puerto apunta a un Root-Bridge que está fuera de la región MST.
- Estado
  Indica el estado momentáneo en que se encuentra el puerto. Los valores solo se muestran. Este parámetro depende del protocolo configurado. Son posibles los siguientes estados:
  - Discarding
    El puerto recibe telegramas BPDU. Se rechazan otros telegramas entrantes o salientes.
  - Listening
    El puerto recibe y envía telegramas BPDU. El puerto está integrado en el algoritmo de Spanning Tree. Se rechazan otros telegramas entrantes y salientes.
  - Learning
    El puerto aprende activamente la topología, es decir, las direcciones de los dispositivos. Se rechazan otros telegramas entrantes y salientes.
  - Forwarding
    El puerto está activo en la red tras el tiempo de reconfiguración. El puerto recibe y envía telegramas de datos.
- Oper. Version
  Describe el tipo de Spanning Tree en el que trabaja el puerto

- **Prioridad**
  Si la ruta determinada por el Spanning Tree puede conducir alternativamente por varios puertos de un dispositivo, se selecciona el puerto con la prioridad más alta (es decir, con el valor menor para ese parámetro). Para la prioridad puede indicarse un valor de 0 a 240 en incrementos de 16. Si indica un valor no divisible por 16, el valor se ajustará automáticamente.
- **Costes de ruta**
  Este parámetro sirve para calcular la ruta que se ha de elegir. Se selecciona como ruta el trayecto con el valor más bajo. Si varios puertos de un dispositivo tienen el mismo valor, se elige el puerto con el número de puerto más bajo.
  Si el valor de "Calc. costes" es "0", se muestra el valor determinado automáticamente. En cualquier otro caso se indica el valor de "Calc. costes".
  La determinación de los costes de encaminamiento se basa prioritariamente en la velocidad de transmisión. Cuanto mayor es la velocidad de transmisión alcanzable, menor es el valor de los costes de encaminamiento.
  Valores típicos de costes de encaminamiento con Rapid Spanning Tree:
  - 10.000 Mbits/s = 2.000
  - 1000 Mbits/s = 20.000
  - 100 Mbits/s = 200.000
  - 10 Mbits/s = 2.000.000
- **Tipo Edge**
  Indica el tipo de conexión. Son posibles los siguientes valores:
  - Edge Port
    Hay un terminal en este puerto.
  - No Edge Port
    En este puerto hay un dispositivo Spanning Tree o Rapid Spanning Tree.
- **Tipo P.t.P.**
  Indica el tipo de conexión punto a punto. Son posibles los siguientes valores:
  - P.t.P.
    Con semidúplex se parte de una conexión punto a punto.
  - Shared Media
    Con una conexión dúplex no se parte de una conexión punto a punto.

Estadísticas VRRP

Introducción

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra las estadísticas del protocolo VRRP y todos los router virtuales configurados.

Descripción de los valores mostrados

Se muestran los siguientes campos:

- Errores VRID
  Indica el número de telegramas VRRP recibidos que contenían una VRID no soportada.
- Errores de versión
  Indica el número de telegramas VRRP recibidos que contenían un número de versión no válido.
- Errores de suma de verificación
  Indica el número de telegramas VRRP recibidos que contenían una suma de verificación no válida.

La tabla se divide en las siguientes columnas:

- Interfaces
  Interfaz a la que se refieren los ajustes.
- VRID
  Indica la ID del router virtual.
  Los valores válidos son 1...255.
- Transición a estado maestro
  Indica la frecuencia con la que este router virtual ha adoptado la función "Maestro".
- Notificaciones recibidas
  Indica la frecuencia con la que se ha recibido un telegrama VRRP que contenía una lista de direcciones errónea.
- Errores de intervalo de notificación
  Indica cuántos telegramas VRRP erróneos se han recibido cuyo intervalo no coincidía con el valor fijado localmente.
- Errores IP TTL
  Indica cuántos telegramas VRRP erróneos se han recibido cuyo valor TTL (Time to live) no coincidía en el encabezado de IP.
- Tel. Prio 0 recibidos
  Indica cuántos telegramas VRRP se han recibido con la prioridad 0. Los telegramas VRRP con la prioridad 0 se envían cuando se apaga un router maestro. Estos telegramas permiten pasar rápidamente al correspondiente router de reserva.
- Tel. Prio 0 enviados
  Indica cuántos telegramas VRRP se han enviado con la prioridad 0. Los telegramas con la prioridad 0 se envían cuando se apaga un router maestro. Estos telegramas permiten pasar rápidamente al correspondiente router de reserva.
- Tipo no válido
  Muestra la frecuencia con la que se ha recibido un paquete VRRP que tenía el tipo erróneo.
- Errores de listas de direcciones
  Muestra el número de errores en la lista de direcciones.
- Tipo de aut. no válido
  Indica cuántos telegramas VRRP erróneos se han recibido cuyo tipo de autenticación no era 0. Tipo 0 significa "sin autentificación".

- **Tipo de aut. divergente**
  Indica cuántos telegramas VRRP erróneos se han recibido cuyo tipo de autenticación no coincide.
- **Errores de longitud de telegrama**
  Indica cuántos telegramas VRRP erróneos se han recibido cuya longitud no es correcta.

Redundancia de anillo

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Al entregarse el dispositivo, está preajustada la Automatic Redundancy Detection (ARD). Si se debe utilizar el método High Speed Redundancy (HRP) que se utilizaba anteriormente, se tiene que configurar HRP.

- Tiempo de reconfiguración del tráfico de telegramas tras un cambio de redundancia en MRP: 200 ms
- Tiempo de reconfiguración del tráfico de telegramas tras un cambio de redundancia en HRP: 300 ms

La página le informa sobre los valores ajustados para el dispositivo.

Descripción

Se muestran los siguientes campos:

- **Función de redundancia**
  Se muestran los valores siguientes:
  - Disabled
    La redundancia de anillo está desactivada en el dispositivo.
  - MRP Manager
    El dispositivo trabaja como "Administrador MRP".
  - MRP Client
    El dispositivo trabaja como "Cliente MRP".
  - HRP Client
    El dispositivo trabaja como "Cliente HRP".
  - HRP Manager
    El dispositivo trabaja como "Administrador HRP".
- **Estado RM**
  La columna "Estado RM" indica si el IE Switch trabaja de administrador de redundancia y si en esta función ha abierto o interconectado el anillo.
  - Passive
    El IE Switch trabaja de administrador de redundancia y ha abierto el anillo, es decir, la línea de switches conectada a los puertos del anillo trabaja sin errores. El estado "Passive" se indica también cuando el IE Switch no trabaja de administrador de redundancia (RM Function Disabled).
  - Active
    El IE Switch trabaja de administrador de redundancia y ha cerrado el anillo, es decir, la línea de switches conectada a los puertos del anillo está interrumpida (error). El administrador de redundancia conmuta la conexión entre sus puertos de anillo y restablece una topología lineal continua.
  - Si la redundancia de medios en topologías de anillo está completamente desconectada, se visualizan los últimos puertos configurados en el anillo y aparece el texto "Ring Redundancy disabled".
- **Estado Observer**
  Muestra el estado actual del Observer.
- **Puertos de anillo**
  Los puertos 1 y 2 del anillo muestran los puertos utilizados para la redundancia.
- **Número de cambios a estado activo RM**
  Indica la frecuencia con la que el dispositivo ha conmutado al estado activo en calidad de administrador de redundancia, es decir, ha cerrado el anillo.
  Si la función de redundancia está desactivada o el dispositivo no es administrador HRP/MRP, aparece el texto "Administrador de redundancia desactivado".
- **Retardo máx. de los telegramas test RM [ms]**
  Muestra el tiempo de retardo máximo para telegramas de prueba del administrador de redundancia.
  Si la función de redundancia está desactivada o el dispositivo no es administrador HRP/MRP, aparece el texto "Administrador de redundancia desactivado".

Standby

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra el estado de disponibilidad del dispositivo.

Encontrará información sobre el cableado Ethernet y el emplazamiento topológico de maestro y esclavo en el "Manual de red Industrial Ethernet".

Descripción

Se muestran los siguientes campos:

- **Puertos Standby**
  Muestra los puertos utilizables.
- **Nombre Standby**
  Indica el nombre de la conexión de reserva. Con este nombre se define la pareja de dispositivos maestro/esclavo (ambos han de estar en el mismo anillo). Para esto se da el mismo nombre a dos dispositivos de un anillo. El nombre se puede elegir libremente, pero solo se debe utilizar para una pareja de dispositivos en toda la red.
- **Función Standby**
  Indica si la conexión de reserva está activada o desactivada.
  - Master
    El dispositivo tiene conexión con el interlocutor y trabaja de maestro. En caso de funcionamiento correcto, en este dispositivo está activo el puerto Standby.
  - Slave
    El dispositivo tiene conexión con el interlocutor y trabaja de esclavo. En caso de funcionamiento correcto, en este dispositivo está inactivo el puerto Standby.
  - Disabled
    El acoplamiento Standby está desactivado. El dispositivo no trabaja ni de maestro ni de esclavo. El puerto Standby funciona como puerto normal sin función Standby.
  - Waiting for Connection...
    Todavía no se ha establecido una conexión con el interlocutor. El puerto Standby está inactivo. En este caso, ocurre que la configuración del interlocutor no es coherente (p. ej. nombre de conexión incorrecto, acoplamiento Standby desactivado) o existe un defecto físico (p. ej. fallo del dispositivo, Link-Down).
  - Connection Lost
    Se ha perdido la conexión existente con el interlocutor. En este caso, existe un defecto físico (p. ej. fallo del dispositivo, Link-Down) o se ha modificado la configuración del interlocutor (p. ej. otro nombre de conexión, acoplamiento Standby desactivado).

- Estado Standby
  Muestra el estado del puerto Standby:
  - Active
    El puerto Standby de este dispositivo está activo, es decir, habilitado para el tráfico de telegramas.
  - Passive
    El puerto Standby de este dispositivo está inactivo, es decir, bloqueado para el tráfico de telegramas.
- Número de cambios a estado activo Standby
  Muestra el número de cambios del estado pasivo al estado activo.

Estadísticas de Ethernet

Estadística de interfaz

Estadística de interfaz

Esta página muestra la estadística de la tabla de interfaces de la Management Information Base (MIB).

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Valores visualizados

La tabla se divide en las siguientes columnas:

- Bytes recibidos
  Muestra el número de bytes recibidos.
- Bytes enviados
  Muestra el número de bytes enviados.
- Telegramas Unicast recibidos
  Muestra el número de telegramas Unicast recibidos.
- Telegramas no Unicast recibidos
  Muestra el número de telegramas recibidos que no son del tipo Unicast.
- Telegramas Unicast enviados
  Muestra el número de telegramas Unicast enviados.
- Telegramas no Unicast enviados
  Muestra el número de telegramas enviados que no son del tipo Unicast.

Longitud de telegrama

Telegramas clasificados por longitud

Esta página muestra cuántos telegramas se han recibido y enviado en cada puerto, así como el tamaño de dichos telegramas.

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Valores visualizados

La tabla se divide en las siguientes columnas:

- **Puerto**
  Indica los puertos y Link Aggregations disponibles.

  Nota

  Visualización de las estadísticas sobre telegramas

  Al considerar las estadísticas de la longitud de los telegramas, recuerde que se contabilizan tanto los telegramas entrantes como los salientes.

- **Longitudes de telegrama**
  El resto de columnas, situadas tras el número de puerto correspondiente, contienen los números absolutos de los telegramas registrados en función de la longitud de estos últimos. Para ello se distingue entre las siguientes longitudes de telegrama:
  - 64 bytes
  - 65 - 127 bytes
  - 128 - 255 bytes
  - 256 - 511 bytes
  - 512 - 1023 bytes
  - 1024 - máx

Botón

Inicializar contadores

Haga clic en "Inicializar contadores" para inicializar todos los contadores. Los contadores se inicializan con un rearranque completo.

Tipo de telegrama

Telegramas recibidos, clasificados por tipo

Esta página muestra cuántos telegramas del tipo "Unicast", "Multicast" y "Broadcast" se han recibido en cada puerto.

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Valores visualizados

La tabla se divide en las siguientes columnas:

- **Puerto**
  Indica los puertos y Link Aggregations disponibles.
- **Unicast/ Multicast / Broadcast**
  El resto de columnas tras los números de puerto contienen los valores absolutos de los telegramas entrantes según los tipos "Unicast", "Multicast" y "Broadcast".

Botón

Inicializar contadores

Haga clic en "Inicializar contadores" para inicializar todos los contadores. Los contadores se inicializan con un rearranque completo.

Error de telegrama

Telegramas recibidos con errores

Esta página muestra cuántos telegramas con errores se han recibido en cada puerto.

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Valores visualizados

La tabla se divide en las siguientes columnas:

- Puerto
  Indica los puertos y Link Aggregations disponibles.
- Tipos de error
  El resto de columnas tras los números de puerto contienen los valores absolutos de los telegramas entrantes según el tipo de error.
  Las columnas de la tabla dividen los telegramas según los siguientes tipos de errores:
  - CRC
    Telegramas cuyo contenido no coincide con la respectiva suma de verificación CRC.
  - Demasiado corto
    Telegramas de longitud inferior a 64 bytes.
  - Demasiado largo
    Telegramas rechazados por ser demasiado largos.
  - Fragmentos
    Telegramas de longitud inferior a 64 bytes y con suma de verificación CRC incorrecta.
  - Jabbers
    Telegramas con tag VLAN y suma de verificación CRC incorrecta que se rechazaron por ser demasiado largos.
  - Colisiones
    Colisiones detectadas.

Botón

Inicializar contadores

Haga clic en "Inicializar contadores" para inicializar todos los contadores. Los contadores se inicializan con un rearranque completo.

Historial

Muestras de las estadísticas

La página presenta para cada puerto muestras con información de las estadísticas.

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Ajustes

- **Puerto**
  Seleccione el puerto para el que desea visualizar el historial.

Valores visualizados

- **Entradas**
  Número máximo de muestras que se guardan simultáneamente.
- **Intervalo[s]**
  Intervalo tras el cual se guarda el estado actual de la estadística como muestra.

La tabla se divide en las siguientes columnas:

- **Muestra**
  Número de la muestra
- **Momento de la muestra**
  Tiempo de funcionamiento del sistema en el que se tomó la muestra.
- **Unicast**
  Número de telegramas Unicast recibidos.
- **Multicast**
  Número de telegramas Multicast recibidos.
- **Broadcast**
  Número de telegramas Broadcast recibidos.
- **CRC**
  Número de telegramas con suma de verificación CRC incorrecta.
- **Demasiado corto**
  Número de telegramas de longitud inferior a 64 bytes.
- **Demasiado largo**
  Número de telegramas rechazados por ser demasiado largos.
- **Fragmentos**
  Número de telegramas de longitud inferior a 64 bytes y con suma de verificación CRC incorrecta.
- **Jabbers**
  Número de telegramas con tag VLAN y suma de verificación CRC incorrecta que se rechazan por ser demasiado largos.
- **Colisiones**
  Número de colisiones de telegramas recibidos.
- **Carga**
  Carga del puerto mientras se toma una muestra.

Unicast

Esta página muestra las direcciones MAC Unicast aprendidas y estáticas.

Descripción

- ID de VLAN
  Indica la ID de VLAN asignada a esa dirección MAC.
- Dirección MAC
  Indica la dirección MAC del participante que el dispositivo ha aprendido o que el usuario ha configurado.
- Estado
  Muestra el estado de cada entrada de dirección:
  - Learnt
    La dirección indicada ha sido aprendida a raíz de la recepción de un telegrama de este dispositivo y se borrará una vez transcurrido el Aging Time si no se reciben más paquetes de este dispositivo.
  - Static
    Configuración por parte del usuario. Las direcciones estáticas están almacenadas de forma permanente, lo que significa que no se borran tras expirar el Aging Time ni al reiniciar el switch.
- Puerto
  Indica a través de qué puerto se puede acceder al dispositivo con la dirección especificada. Los telegramas recibidos por el dispositivo cuya dirección de destino coincida con esta dirección se transmiten a ese puerto.

Multicast

Esta página muestra las direcciones MAC Multicast aprendidas y estáticas.

Descripción

- ID de VLAN
  Indica la ID de la VLAN a la que está asignada la dirección MAC Multicast.
- Dirección MAC
  Indica la dirección MAC Multicast aprendida por el dispositivo o configurada por el usuario.
- Estado
  Muestra el estado de cada entrada de dirección. Al respecto son posibles los siguientes datos:
  - static
    La dirección ha sido introducida estáticamente por el usuario. Las direcciones estáticas están almacenadas de forma permanente, lo que significa que no se borran tras expirar el Aging Time ni al rearrancar el dispositivo. Deben ser borradas por el usuario.
  - IGMP
    El puerto de destino para esta dirección ha sido averiguado por configuración IGMP.
  - GMRP
    El puerto de destino para esta dirección ha sido registrado a través de un telegrama GMRP recibido.

Routing

Tabla de enrutamiento

Esta página muestra la tabla de enrutamiento del dispositivo.

Descripción

La tabla se divide en las siguientes columnas:

- **Red de destino**
  Indica la dirección de destino de esa ruta.
- **Máscara de subred**
  Indica la máscara de subred de esa ruta.
- **Gateway**
  Indica la pasarela para esa ruta.
- **Interfaz**
  Indica la interfaz para esa ruta.
- **Métrica**
  Indica la métrica de la ruta. Cuanto mayor es el valor, más tardan los paquetes en llegar a su destino.
- **Protocolo routing**
  Indica el protocolo de enrutamiento del que proviene la entrada de la tabla de enrutamiento. Son posibles las siguientes entradas:
  - Conectado: rutas conectadas
  - Estático: rutas estáticas
  - RIP: rutas vía RIP
  - OSPF: rutas vía OSPF
  - Otras: otras rutas

Interfaces OSPFv2

Sinopsis

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra la configuración de la interfaz OSPF.

Descripción de los valores mostrados

La tabla se divide en las siguientes columnas:

- Dirección IP
  Muestra la dirección IP de la interfaz OSPF.
- Area ID
  Indica la Area ID a la que pertenece la interfaz OSPF.
- Estado de la interfaz
  Muestra el estado de la interfaz:
  - Down
    La interfaz no está disponible.
  - Loopback
    Interfaz Loopback
  - Waiting
    Arranque y negociación de la interfaz.
  - Point to Point
    Conexión punto a punto
  - Designated Router
    El router es un router designado y genera Network LSA.
  - Backup D. Router
    El router es el router de reserva para el router designado.
  - Other D. Router
    La interfaz está arrancada. El router no es un router designado ni un router de reserva designado.
- Estado OSPF
  Indica el estado de OSPF.
  - Enabled: OSPF está activado en la interfaz.
  - Disabled: OSPF está desactivado en la interfaz.
- Designated Router
  Muestra la dirección IP del router designado para esta interfaz OSPF.
- Backup Designated Router
  Muestra la dirección IP del router de reserva designado para esta interfaz OSPF.
- Eventos
  Indica el número de cambios de estado de OSPF.

Vecinos OSPFv2

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra los routers vecinos registrados dinámicamente en las distintas redes.

Descripción de los valores mostrados

La tabla se divide en las siguientes columnas:

- **Dirección IP**
  Indica la dirección IP del router vecino en esta red.
- **ID de router**
  Indica la ID del router vecino. Ambas direcciones pueden coincidir.
- **Estado**
  Muestra el estado del router vecino. El estado puede adoptar los siguientes valores:
  - unknown
    Estado del router vecino desconocido.
  - down
    El router vecino no está accesible.
  - attempt y init
    Estados de corta duración durante la inicialización
  - two-way
    Recepción de paquetes Hello por ambos lados. Se especifican el router designado y el router de reserva designado.
  - exchangestart, exchange y loading
    Estado durante el intercambio de los LSAs
  - full
    La base de datos está completa y es síncrona dentro del área. Las rutas ya pueden determinarse.

  Nota

  Estado normal

  Si el router interlocutor es un router designado o un router de reserva designado, el estado es "full". De lo contrario, el estado es "two-way".

- **Assoc. Area Type**
  Indica el tipo de área a través de la cual existe la relación de vecindad. Existen los siguientes tipos de área:
  - Normal
  - Stub
  - NSSA
  - Backbone
- **Prioridad**
  Muestra la prioridad del router vecino. Solo tiene relevancia para la selección del router designado en una red. Este dato no es relevante en el caso de routers vecinos virtuales.
- **Paquetes Hello supr.**
  Indica si hay paquetes Hello suprimidos destinados al router vecino virtual.
  - no: no hay paquetes Hello suprimidos (predeterminado).
  - sí: hay paquetes Hello suprimidos.

- **Cola de espera Hello**
  Longitud de la cola de espera con los paquetes Hello aún por transmitir.
- **Eventos**
  Indica el número de cambios de estado.

Vecinos virtuales OSPFv2

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra los routers vecinos virtuales configurados.

Sinopsis

La tabla se divide en las siguientes columnas:

- **Dirección IP**
  Indica la dirección IP del router vecino virtual en esta red.
- **ID de router**
  Indica la ID del router vecino virtual.
- **Estado**
  Muestra el estado del router vecino. El estado puede adoptar los siguientes valores:
  - unknown
    Estado del router vecino desconocido.
  - down
    El router vecino no está accesible.
  - attempt y init
    Estados de corta duración durante la inicialización
  - two-way
    Recepción de paquetes Hello por ambos lados. Se especifican el router designado y el router de reserva designado.
  - exchangestart, exchange y loading
    Estado durante el intercambio de los LSAs
  - full
    La base de datos está completa y es síncrona dentro del área. Las rutas ya pueden determinarse.

    Nota
    Estado normal

    Si el router interlocutor es un router designado o un router de reserva designado, el estado es "full". De lo contrario, el estado es "two-way".
- **Trans. Area ID**
  Muestra la ID del área a través de la cual se genera la relación de vecindad virtual.

- **Paquetes Hello supr.**
  Indica si hay paquetes Hello suprimidos destinados al router vecino virtual.
  – no: no hay paquetes Hello suprimidos (predeterminado)
  – sí: hay paquetes Hello suprimidos.
- **Cola de espera Hello**
  Longitud de la cola de espera con los paquetes Hello aún por transmitir.
- **Eventos**
  Indica el número de cambios de estado.

OSPFv2 LSDB

Sinopsis .

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

La Link State Database es la base de datos central para la administración de todas las conexiones de un área. Consta de los llamados Link State Advertisements (LSAs). Los datos más importantes de estos LSAs se muestran en esta página del WBM.

Descripción de los campos mostrados

La tabla se divide en las siguientes columnas:

- Area ID
  Indica la ID del área a la que pertenece el LSA. Si el LSA es una conexión externa, se indica un '-'.
- Tipo de LSA
  Indica el tipo de LSA. Son posibles los siguientes valores:
  - unknown
    Tipo de LSA desconocido.
  - Router
    El Router LSA (tipo 1) se envía desde el router OSPF dentro de un área. El LSA contiene información sobre el estado de todas las interfaces del router.
  - Network
    El Network LSA (tipo 2) se envía desde el router designado dentro de un área. El LSA contiene una lista de routers conectados con la red.
  - NSSA External
    El NSSA External LSA (tipo 7) se envía desde el NSSA-ASBR dentro de un NSSA. El NSSA-ASBR recibe LSAs del tipo 5 y convierte la información a LSAs del tipo 7. El NSSA Router puede enviar estos LSAs dentro de un NSSA.
  - Summary
    El Summary LSA (tipo 3) se envía desde el ABR dentro de un área. El LSA contiene información sobre rutas a otras redes.
  - AS Summary
    El AS Summary LSA (tipo 4) se envía desde el Area Border Router dentro de un área. El LSA contiene información sobre rutas a otros sistemas autónomos.
  - AS External
    El AS External LSA (tipo 5) se envía desde el AS Border Router dentro de un sistema autónomo. El LSA contiene información sobre las rutas desde una red a otra red.
- Link State ID
  Indica la ID del LSA.
- ID de router
  Indica la ID del router que ha enviado este LSA.
- Número de secuencia
  Indica el número de secuencia del LSA. Con cada renovación de un LSA este número de secuencia aumenta en uno.

Estadística RIPv2

Sinopsis

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Esta página muestra la estadística de la interfaz RIP.

Descripción de los valores mostrados

La tabla se divide en las siguientes columnas:

- Dirección IP
  Muestra la dirección IP de la interfaz RIPv2.
- Telegramas ignorados
  Número de telegramas RIP recibidos, que se han borrado y por ello no se han tenido en cuenta.
- Rutas ignoradas
  Número de rutas de telegramas RIP válidos que no se han podido tener en cuenta.
- Actualizaciones enviadas
  Indica la frecuencia con la que el router ha enviado su tabla de routing a los routers vecinos.

Sistema

Configuración

En esta página se definen los servicios a través de los cuales se accede al dispositivo. Para algunos servicios hay más páginas de configuración, en las que se pueden realizar ajustes detallados.

Ajustes

- Servidor Telnet
  Active o desactive el servicio "Servidor Telnet" para el acceso no codificado a CLI.
- Servidor SSH
  Active o desactive el servicio "Servidor SSH" para el acceso codificado a CLI.
- Solo servidor HTTPS
  Active o desactive el acceso vía HTTP.
- Cliente DNS
  Active o desactive la función de cliente DNS del IE Switch. Puede realizar más ajustes en "Sistema > Cliente DNS".
- Cliente SMTP
  Active o desactive el cliente SMTP. Puede realizar más ajustes en "Sistema > Cliente SMTP".
- Cliente Syslog
  Active o desactive el cliente de eventos de sistema. Puede realizar más ajustes en "Sistema > Cliente Syslog".

- **Servidor DCP**
  Especifique si se debe poder acceder al dispositivo con DCP (Discovery and Configuration Protocol):
  - "-" (desactivado)
    DCP está desactivado. No pueden leerse ni modificarse los parámetros del dispositivo.
  - Lectura/escritura
    Con DCP es posible leer y modificar los parámetros del dispositivo.
  - Solo lectura
    Con DCP es posible leer los parámetros del dispositivo, pero no modificarlos.
- **Ajuste de la hora**
  Elija el ajuste deseado. Son posibles los siguientes ajustes:
  - Manual
    La hora del sistema se ajusta de forma manual.
  - Cliente SNTP
    La hora del sistema se ajusta mediante un servidor SNTP. Puede realizar más ajustes en "Sistema > Hora del sistema > Cliente SNTP".
  - Cliente NTP
    La hora del sistema se ajusta mediante un servidor NTP. Puede realizar más ajustes en "Sistema > Hora del sistema > Cliente NTP".
  - SIMATIC Time
    La hora del sistema se ajusta mediante un reloj SIMATIC. Puede realizar más ajustes en "Sistema > Hora del sistema > SIMATIC Time Client".
  - Cliente PTP
    La hora del sistema se ajusta mediante un servidor PTP. Puede realizar más ajustes en "Sistema > Hora del sistema > Cliente PTP".
    Esta función solo está disponible en SCALANCE X500.
- **SNMP**
  Elija el protocolo deseado. Son posibles los siguientes ajustes:
  - "-" (SNMP desactivado)
    No se puede acceder a los parámetros del dispositivo vía SNMP.
  - SNMPv1/v2c/v3
    Se puede acceder a los parámetros del dispositivo con las versiones 1, 2c o 3 de SNMP. Puede realizar más ajustes en "Sistema > SNMP > General".
  - SNMPv3
    Se puede acceder a los parámetros del dispositivo con la versión 3 de SNMP. Puede realizar más ajustes en "Sistema > SNMP > General".
- **SNMPv1/v2 solo lectura**
  Active o desactive el acceso de escritura a variables SNMP con SNMPv1/v2c.
- **Traps SNMPv1**
  Active o desactive el envío de traps SNMP (telegramas de alarma). Puede realizar más ajustes en "Sistema > SNMP > Traps".

- **NFC (Near Field Communikation)**
  Active o desactive la función "NFC".
  Esta función solo está disponible en SCALANCE XM400.
- **Modo de configuración**
  Elija el modo de operación deseado. Son posibles los siguientes modos de operación:
  - Guardar automáticamente
    Modo de seguridad automático. La configuración se guarda de forma automática aprox. 1 minuto después de la última modificación de parámetros o al reiniciar el dispositivo.
  - Trial
    En el modo de configuración "Trial" se aceptan modificaciones, pero no se guardan en el archivo de configuración (Startup Configuration).
    Para guardar modificaciones en el archivo de configuración, utilice el botón "Escritura de la configuración de arranque". El botón se muestra al ajustar este modo de configuración. Además, en el área de visualización aparece el aviso "Modo de prueba activo - Para guardar los ajustes, pulse el botón "Escribir config. de arranque"" en cuanto hay algún cambio sin guardar. Este aviso es visible en todas las páginas hasta que las modificaciones realizadas se guardan o el dispositivo se reinicia.

General

Dispositivo

Esta página contiene información general sobre el dispositivo.

Ajustes

- **Hora actual del sistema** (solo disponible online)
  Indica la hora actual del sistema. La hora del sistema la ajusta el usuario o se sincroniza mediante un telegrama de hora, que puede ser un telegrama de hora SINEC H1, NTP o SNTP. (Solo lectura)
- **Tiempo de funcionamiento del sistema** (solo disponible online)
  Indica el tiempo de funcionamiento del dispositivo desde el último rearranque completo. (Solo lectura)
- **Tipo de dispositivo** (solo disponible online)
  Indica la denominación de tipo del dispositivo. (Solo lectura)
- **Nombre del sistema**
  Puede introducir el nombre del dispositivo. El nombre introducido se muestra en el área de selección. Pueden utilizarse un máximo de 255 caracteres. El nombre del sistema se indica también en el símbolo del sistema CLI (Prompt). El número de caracteres está limitado en el símbolo del sistema CLI. El nombre del sistema se trunca después de 16 caracteres.

- **Persona de contacto**
  Puede introducir el nombre de una persona de contacto responsable de la administración del dispositivo.
- **Ubicación del dispositivo**
  Puede introducir el lugar de montaje del dispositivo. El lugar de montaje introducido se muestra en el área de selección.

Nota

En los campos de entrada se usan los caracteres ASCII 0x20 hasta 0x7e.

Al principio y al final de los campos de entrada "Nombre del sistema", "Persona de contacto" y "Ubicación del dispositivo" no se permiten los caracteres "<", ">" ni espacios en blanco.

Coordenadas

En esta página se configuran las coordenadas geográficas (latitud, longitud y altura sobre el elipsoide según WGS84). Son campos meramente informativos con una longitud máxima de 32 caracteres.

Determinación de las coordenadas

Para determinar las coordenadas geográficas, utilice los mapas que se suministran con el dispositivo.

Las coordenadas geográficas también se pueden determinar mediante un receptor GPS. Por lo general, estos dispositivos muestran las coordenadas geográficas directamente, por lo que basta con introducirlas en los campos de entrada de esta página.

Ajustes

- **Latitud**
  Introduzca la latitud norte o sur de la ubicación del dispositivo.
  P. ej., +49° 1' 31,67" significa que el dispositivo se encuentra a 49 grados, 1 minuto y 31,67 segundos de latitud norte.
  La latitud sur va precedida de un signo negativo.
  También se pueden añadir las letras N (latitud norte) o S (latitud sur) a las indicaciones numéricas (49° 1' 31,67" N).
- **Longitud**
  Introduzca la longitud este u oeste de la ubicación del dispositivo.
  P. ej., +8° 20' 58,73" significa que el dispositivo se encuentra a 8 grados, 20 minutos y 58,73 segundos de longitud este.
  La longitud oeste va precedida de un signo negativo.
  También se pueden añadir las letras O o E (longitud este) o W (longitud oeste) a las indicaciones numéricas (8° 20' 58,73" E).
- **Altitud**
  Introduzca el valor de la altitud en metros por encima o por debajo del nivel del mar.
  P. ej., 158 m significa que el dispositivo está a una altura de 158 m por encima del nivel del mar.
  Las altitudes por debajo del nivel del mar (p. ej. en el Mar Muerto) van precedidas de un signo negativo.

IP agente

Configuración de las direcciones IP

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

En esta página se realiza la configuración IP del dispositivo.

En los dispositivos con varias interfaces IP, esta llamada remite al comando de menú "Subredes > Configuración" del menú "Layer 3" y a la configuración de la interfaz TIA que aparece allí.

Nota

La dirección IP del puerto In-Band y la dirección IP del puerto Out-of-Band deben pertenecer a distintas subredes.

Ajustes

- **Dirección IP**
  En "In-Band" introduzca la dirección IP del puerto In-Band (puerto switch). En "Out-Band" introduzca la dirección IP del puerto Out-of-Band (puerto Ethernet).
  Si modifica la dirección IP, el navegador web debería ajustarse automáticamente a la nueva dirección. Si no es así, introduzca manualmente la nueva dirección en el navegador web.
- **Máscara de subred**
  Introduzca en "In-Band" la máscara de subred del puerto In-Band y, en "Out-Band", la del puerto Out-of-Band.
- **Gateway predeterminada**
  Si el dispositivo va a comunicarse con otros dispositivos (estaciones de diagnóstico, servidores de correo electrónico, etc.) integrados en otra subred, introduzca aquí la dirección IP de la pasarela estándar. No puede accederse al puerto Out-of-Band desde otra subred.
- **Agent VLAN ID**
  Seleccione en la lista desplegable la VLAN ID para la administración de In-Band. Solo pueden seleccionarse VLAN ya configuradas.

  Nota

  Modificación de la ID de VLAN agente

  Cuando el PC de configuración está conectado al dispositivo directamente a través de Ethernet y se modifica la ID de VLAN agente, el dispositivo deja de estar accesible vía Ethernet después de la modificación.

- **Dirección MAC**
  Muestra la dirección MAC del dispositivo. La dirección MAC va asociada al hardware y no se puede modificar.

Cliente DNS

El servidor DNS (Domain Name System) asigna un nombre de dominio a una dirección IP de modo que sea posible identificar unívocamente a un dispositivo.

Cuando esta función está activada, el IE Switch puede comunicarse con un servidor DNS como cliente DNS.

Nota

Para utilizar la función "Cliente DNS" debe haber un servidor DNS en la red.

Descripción

La página contiene los siguientes campos:

- **Cliente DNS**
  Active o desactive la función de cliente DNS del IE Switch.
- **Dirección del servidor DNS**
  Introduzca la dirección IP del servidor DNS.

Rearranque completo

En esta página encontrará un botón para rearrancar el dispositivo, así como diversas posibilidades para restablecer los ajustes predeterminados del mismo.

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Nota

Al reiniciar un dispositivo, tenga en cuenta los siguientes aspectos:

- Solo puede realizar un reinicio del dispositivo si tiene derechos de administrador.
- El rearranque del dispositivo se debe efectuar solo mediante los botones de esta página o con los correspondientes comandos CLI, y no desconectando y volviendo a conectar la alimentación eléctrica.
- Los cambios realizados solo se aplicarán tras hacer clic en el botón "Aplicar ajustes" de la correspondiente página del dispositivo. Si el dispositivo está en modo "Trial", los cambios de la configuración deben guardarse de forma manual antes de un rearranque completo. En el modo "Guardar automáticamente", las últimas modificaciones se guardan automáticamente antes de un rearranque completo.

Ajustes

Para el rearranque completo existen las siguientes posibilidades:

- Reinicio
  Haga clic en este botón para reiniciar el sistema. Debe confirmarse el reinicio en un cuadro de diálogo. Al efectuarse un reinicio, el dispositivo se reinicializa, carga de nuevo el firmware interno y realiza una autocomprobación. Se borran las entradas aprendidas en la tabla de direcciones. Puede dejar abierta la ventana del navegador mientras reinicia el dispositivo. Deberá iniciar de nuevo la sesión:
- Restablecimiento de los ajustes predeterminados y reinicio
  Haga clic en este botón para reiniciar el sistema. Debe confirmarse el reinicio en un cuadro de diálogo. Al efectuarse un reinicio, el dispositivo se reinicializa, carga de nuevo el firmware interno y realiza una autocomprobación. Se borran las entradas aprendidas en la tabla de direcciones. Puede dejar abierta la ventana del navegador mientras reinicia el dispositivo. Deberá iniciar de nuevo la sesión.

Nota

Al restablecer todos los ajustes a los valores de fábrica, se pierden también la dirección IP y las contraseñas. Tras ello solo se podrá acceder al dispositivo mediante la Primary Setup Tool o DHCP.

En función de la conexión, un dispositivo que antes estaba correctamente configurado puede generar telegramas circulantes indefinidamente y, con ello, el fallo del tráfico de datos.

Cargar y guardar

Cargar y guardar vía HTTP

Cargar y guardar datos vía HTTP

Esta página ofrece la posibilidad de almacenar datos del dispositivo en un archivo externo en su PC cliente o cargar dichos datos en los dispositivos desde un archivo externo del PC. De este modo puede cargar, p. ej., un firmware nuevo desde un archivo de su PC cliente.

Nota

Incompatibilidad con versiones anteriores

Si se instala una versión anterior, pueden perderse los datos de configuración. En ese caso, tras la instalación del firmware el dispositivo se enciende con los ajustes de fábrica.

Incompatibilidad con versiones anteriores con el PLUG insertado

Si se instala una versión anterior, pueden perderse los datos de configuración. En ese caso, tras la instalación del firmware el dispositivo se enciende con los ajustes de fábrica. Si, además, hay un PLUG insertado en el dispositivo, después del rearranque tendrá el estado "Not Accepted", puesto que en el PLUG sigue habiendo los datos de configuración de la versión más actual que estaba instalada antes. Esto permite regresar al firmware más actual que estaba instalado anteriormente sin perder datos de configuración. Si la configuración original del PLUG ya no se necesita, puede borrarse o sobrescribirse manualmente mediante > Sistema > PLUG.

Nota

Archivos de configuración y modo Trial/Guardar automáticamente

En el modo "Guardar automáticamente" se realiza una copia de seguridad automática antes de que se transmitan los archivos de configuración (ConfigPack y Config).
En modo Trial los cambios se aplican, pero no se guardan en los archivos de configuración (ConfigPack y Config). Utilice el botón "Escritura de la configuración de arranque" de la página del WBM "Sistema > Configuración" para guardar cambios en el archivo de configuración.

Ajustes

La tabla se divide en las siguientes columnas:

- Tipo
  Indica el tipo de archivo.
- Descripción
  Muestra la descripción breve del tipo de archivo.
- Cargar
  Este botón permite cargar archivos en el dispositivo. El botón se puede activar cuando esta función es compatible con el tipo de archivo.

- Guardar
  Este botón permite guardar archivos del dispositivo. El botón solo se puede activar cuando esta función es compatible con el tipo de archivo y el archivo existe en el dispositivo.
- Borrar
  Este botón permite borrar archivos del dispositivo. El botón solo se puede activar cuando esta función es compatible con el tipo de archivo y el archivo existe en el dispositivo.

Carga de un firmware nuevo

Una vez cargado correctamente el firmware, se le solicitará reiniciar el dispositivo.

Reinicie el dispositivo y continúe con la configuración con el firmware recién iniciado.

Procedimiento

Cargar datos vía HTTP

1. Para cargar datos, haga clic en uno de los botones "Cargar".
   Se abre un cuadro de diálogo para cargar un archivo.
2. Navegue hasta el archivo que desee cargar.
3. Haga clic en el botón "Abrir" del cuadro de diálogo.
   Ahora el archivo estará cargado.
4. Si aparece el mensaje solicitando reiniciar el dispositivo, haga clic en el botón "Sí" para hacerlo. Si hace clic en el botón "No", no se reiniciará el dispositivo. Los cambios no surtirán efecto mientras no reinicie el dispositivo.

Guardar datos vía HTTP

1. Para guardar datos, haga clic en uno de los botones "Guardar".
2. Se le solicitará que seleccione un lugar de almacenamiento y un nombre para el archivo. También puede aceptar el nombre de archivo propuesto. Para seleccionar el archivo, utilice el cuadro de diálogo correspondiente del navegador. Una vez realizada la selección, haga clic en el botón "Guardar".

Borrar datos vía HTTP

1. Para borrar datos, haga clic en uno de los botones "Borrar".
   Se borra el archivo en cuestión.

Reutilizar los datos de configuración

Si se desea asignar a varios dispositivos la misma configuración, y las direcciones IP se asignan a través de DHCP, se puede reducir el trabajo necesario para la reconfiguración almacenando y cargando de nuevo los datos de configuración.

Para reutilizar datos de configuración, haga lo siguiente:

1. Guarde en el PC los datos de configuración de un dispositivo ya configurado.
2. Cargue este archivo de configuración en todos los demás dispositivos que desee configurar de este modo.
3. Si algún dispositivo requiere ajustes individuales, debe realizarlos online en el dispositivo en cuestión.

Tenga en cuenta que los datos de configuración se guardan codificados. Por este motivo, los archivos no pueden modificarse con un editor de textos.

Cargar y guardar vía TFTP

Cargar y guardar datos a través de un servidor TFTP

En esta página puede configurar el servidor TFTP y los nombres de los archivos. Además, ofrece la posibilidad de almacenar datos del dispositivo en un archivo externo en su PC cliente o cargar dichos datos en los dispositivos desde un archivo externo del PC. De este modo puede cargar, p. ej., un firmware nuevo desde un archivo de su PC cliente.

Nota

Incompatibilidad con versiones anteriores

Si se instala una versión anterior, pueden perderse los datos de configuración. En este caso, después de instalar el firmware el dispositivo arranca con la configuración de fábrica.
Incompatibilidad con versiones anteriores con PLUG insertado
Al instalar una versión anterior es posible que se pierdan los datos de configuración. En ese caso, tras la instalación del firmware el dispositivo se enciende con los ajustes de fábrica. Si, además, hay un PLUG insertado en el dispositivo, después del rearranque tendrá el estado "Not Accepted", puesto que en el PLUG sigue habiendo los datos de configuración de la versión más actual que estaba instalada antes. Esto permite regresar al firmware más actual que estaba instalado anteriormente sin perder datos de configuración. Si la configuración original del PLUG ya no se necesita, es posible borrar o sobrescribir manualmente el PLUG en la página "Sistema > C-PLUG".

Nota

Archivos de configuración y modo Trial/Guardar automáticamente

En el modo de configuración "Guardar automáticamente" se realiza una copia de seguridad automática antes de que se transmitan los archivos de configuración (ConfigPack y Config). En el modo de configuración "Trial" los cambios se aplican, pero no se guardan en los archivos de configuración (ConfigPack y Config). Utilice el botón "Escritura de la configuración de arranque" de la página del WBM "Sistema > Configuración" para guardar cambios en el archivo de configuración.

Ajustes

- Dirección IP del servidor TFTP
  Introduzca aquí la dirección IP del servidor TFTP con el que se intercambian datos.
- Puerto del servidor TFTP
  Introduzca aquí el puerto del servidor TFTP a través del cual desee efectuar el intercambio de datos. En caso necesario, puede modificar el valor estándar 69 conforme a sus requisitos específicos.

La tabla se divide en las siguientes columnas:

- Tipo de archivo
  Muestra el tipo de archivo.
- Descripción
  Muestra la descripción abreviada del tipo de archivo.

- **Nombre de archivo**
  Introduzca un nombre de archivo.
- **Seleccionar acción** (solo disponible online)
  Seleccione la acción en la lista desplegable. La selección depende del tipo de archivo seleccionado; así, por ejemplo, el archivo de registro solo se puede guardar.
  Son posibles las siguientes acciones:
  - Guardar archivo
    Con esta selección se guarda un archivo en el servidor TFTP.
  - Cargar archivo
    Con esta selección se carga un archivo desde el servidor TFTP.

Eventos

Configuración

En esta página se define cómo debe reaccionar un dispositivo a eventos.

Ajustes

- **Comportamiento del contacto de aviso**
  Elija el comportamiento del contacto de señalización. Son posibles las siguientes respuestas:
  - Estándar
    Ajuste estándar para el contacto de señalización. Si se produce un error se señaliza en el LED de error y el contacto de señalización se abre. Cuando el estado de error desaparece, se apaga el LED de error y el contacto de señalización se cierra.
  - Definido por el usuario
    La función del contacto de señalización es independiente del error que aparece. El contacto de señalización puede abrirse o cerrarse aleatoriamente con acciones del usuario.
- **Estado del contacto de aviso**
  Elija el estado del contacto de señalización. Son posibles los siguientes estados:
  - Cerrado
    El contacto de señalización está cerrado.
  - Abierto
    El contacto de señalización está abierto.

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos de la tabla 2.
- **E-mail/Trap/Tabla de registro/Syslog/Error**
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- Evento
  La columna "Evento" contiene los siguientes valores:
  - Arranque en frío/en caliente
    Se ha conectado el dispositivo o el usuario ha realizado un reinicio.
  - Link Change
    Este evento solo se produce cuando el estado del puerto se somete a vigilancia y ha cambiado de la forma correspondiente, véase "Sistema > Control de errores > Link Change".
  - Error de autentificación
    Este evento se produce en caso de intento de acceso con una contraseña incorrecta.
  - Alarma RMON
    Se ha producido una alarma o un evento en relación con la vigilancia remota del sistema.
  - Cambio de alimentación
    Este evento solo se produce cuando se vigilan las líneas de alimentación 1 y 2. Indica que se ha producido un cambio en el cable 1 o en el cable 2, véase "Sistema > Control de errores > Alimentación de tensión".
  - Cambio de estado RM
    El estado del administrador de redundancia ha cambiado
  - Cambio de topología en Spanning Tree
    La topología STP y RSTP o MSTP ha cambiado.
  - Cambio del estado de error
    El estado de error ha cambiado. El estado de error puede estar relacionado con la vigilancia activa de puertos, con la reacción del contacto de señalización o con la vigilancia de la tensión eléctrica.
  - Cambio de estado VRRP(solo en caso de routing vía VRRP)
    El estado del router virtual ha cambiado.
  - Loop Detection
    Se ha detectado un bucle en el segmento.
  - Cambio de estado OSPF (solo en caso de routing vía OSPF)
    El estado de OSPF ha cambiado.
- E-mail
  El dispositivo envía un correo electrónico. Para ello es necesario que el servidor SMTP esté configurado y la función "Cliente SMTP" esté activada.
- Trap
  El dispositivo activa un trap SNMP. Para ello, en "Sistema > Configuración" debe estar activado "Traps SNMPv1".
- Tabla de registro
  El dispositivo escribe una entrada en la tabla de registro.

- Syslog
  El dispositivo escribe una entrada en el servidor de informes del sistema. Para ello es necesario que el servidor de informes del sistema esté configurado y la función "Cliente Syslog" esté activada.
- Error
  El dispositivo emite un error. El LED de error se enciende.

Filtro Severity

En esta página se ajustan los niveles de valor umbral para enviar notificaciones de eventos del sistema.

Ajustes

La tabla se divide en las siguientes columnas:

- Tipo de cliente
  Elija el tipo de cliente para el que se realizan los ajustes:
  - E-mail
    Envío de avisos de eventos de sistema por correo electrónico
  - Tabla de registro
    Entrada de eventos del sistema en la tabla de registro.
  - Syslog
    Entrada de eventos del sistema en el archivo Syslog.
- Severity
  Seleccione el nivel deseado. Son posibles los siguientes ajustes:
  - Info
    Los eventos del sistema se procesan a partir del grado de gravedad "Info".
  - Warning
    Los eventos del sistema se procesan a partir del grado de gravedad "Warning".
  - Critical
    Los eventos del sistema se procesan a partir del grado de gravedad "Critical".

Cliente SMTP

El dispositivo ofrece la posibilidad de que, al producirse un evento de alarma, se envíe automáticamente un e-mail (p. ej. al administrador de la red). El e-mail contiene la identificación del dispositivo remitente, una descripción de la causa de la alarma en texto claro y un sello horario. De este modo se puede crear para redes con pocos dispositivos un equipo sencillo y centralizado de vigilancia de la red sobre la base de un sistema de correo electrónico. Si llegan e-mails con avisos de fallo, se puede iniciar el WBM vía navegador a través de la identificación del remitente, lo que permite leer más información para el diagnóstico.

En esta página se pueden configurar hasta tres servidores SMTP y las correspondientes direcciones de correo electrónico.

Nota

Dependiendo de las características y la configuración del servidor SMTP, puede ser necesario modificar el campo de entrada "Remitente" para los correos electrónicos. Infórmese a través del administrador del servidor SMTP.

Ajustes

- **Cliente SMTP**
  Active o desactive el cliente SMTP.
- **Dirección de correo electrónico del remitente**
  Introduzca el nombre de remitente que se indica en el correo electrónico, p. ej. el nombre del dispositivo.
- **Puerto SMTP**
  Cambie el puerto si no puede acceder al servidor SMTP a través del puerto 25.
- **Dirección IP del servidor SMTP**
  Introduzca la dirección IP del servidor SMTP.

La tabla contiene las siguientes columnas:

- **Dirección IP del servidor SMTP**
  Muestra la dirección IP del servidor SMTP.
- **Dirección de correo electrónico del destinatario**
  Introduzca aquí la dirección de correo electrónico a la que el dispositivo debe enviar un correo electrónico en caso de fallo. La dirección de correo electrónico puede ser la de una sola persona o de una lista de distribución.

Sintaxis de las direcciones de correo electrónico

Las condiciones siguientes rigen para direcciones de correo electrónico:

- Están permitidos caracteres al-fa-nu-mé-ri-cos.
- Están permitidos los caracteres especiales siguientes:
  - @
  - _ (guión bajo)
  - - (guión)
  - . (punto)
- Debe haber un signo @.
- Solo puede haber un signo @.
- Los caracteres "@" y "." no pueden ser ni el primer ni el último carácter.

Cliente DHCP

Cuando está activado el modo DHCP, el cliente DHCP inicia una petición DHCP en un servidor DHCP configurado, y como respuesta se le asigna una dirección IP. El servidor administra un rango de direcciones desde el cual asigna direcciones IP. También es posible configurar el servidor de modo que, cuando el cliente solicite una dirección IP, siempre se le asigne la misma.

Ajustes

- Solicitud de configuración cliente DHCP (Opt. 66, 67)
  Active esta opción si desea que el cliente DHCP utilice las opciones 66 y 67 para descargar un archivo de configuración y activarlo luego.
- Modo de operación DHCP
  Elija el modo de operación DHCP. Son posibles los siguientes modos de operación:
  - Vía dirección MAC
    La identificación se realiza utilizando la dirección MAC.
  - Mediante ID de cliente DHCP
    La identificación se desarrolla a través de una ID de cliente DHCP definida libremente.
  - Vía nombre del dispositivo
    La identificación se realiza utilizando el nombre del dispositivo. Si el nombre del dispositivo tiene una longitud de 255 caracteres, el último no se utiliza para la identificación.

La tabla contiene las siguientes columnas:

- Interfaz
  Muestra las interfaces disponibles.
- DHCP
  Active o desactive el cliente DHCP.

SNMP

General

En esta página se describen los ajustes básicos de SNMP. Active las funciones que desee utilizar.

Ajustes

- **SNMP**
  Elija el protocolo SNMP. Son posibles los siguientes ajustes:
  - "-" (desactivado)
    SNMP desactivado.
  - SNMPv1/v2c/v3
    Se soporta SNMPv1/v2c/v3.
  - SNMPv3
    Solo se soporta SNMPv3.
- **SNMPv1/v2 solo lectura**
  Si activa esta opción, SNMPv1/v2c solo puede acceder a las variables SNMP en modo de lectura.

  Nota

  Community String

  Por razones de seguridad, no deben utilizarse los valores estándar "public" ni "private". Modifique los Community Strings después de la primera instalación.

- **SNMPv1/v2c Read Community String - Lectura**
  Introduzca el Community String para el acceso del protocolo SNMP.
- **SNMPv1/v2c Read/Write Community String - Lectura/escritura**
  Introduzca el Community String para el acceso de lectura y escritura del protocolo SNMP.
- **Traps SNMPv1**
  Active o desactive el envío de traps SNMP (telegramas de alarma). En la ficha "Trap" se determinan las direcciones IP de los dispositivos a los que se envían traps SNMP.
- **SNMPv1/v2c Trap Community String - Trap**
  Introduzca el Community String para el envío de mensajes SNMPv1/v2.

Traps

Cuando se produce un evento de alarma, un dispositivo puede enviar traps SNMP (telegramas de alarma) a un máximo de 10 estaciones de gestión distintas. Solo se envían traps cuando se producen los eventos definidos al efecto en "Eventos > Configuración".

Nota

Solo se envían traps SNMP si en "SNMP > General" se ha activado el ajuste "Traps SNMPv1".

Ajustes

- **Dirección IP**
  Introduzca la dirección IP de las estaciones a las que el dispositivo envía traps SNMP.

La tabla se divide en las siguientes columnas:

- Dirección IP
  Muestra las direcciones IP de las estaciones a las que el dispositivo envía traps SNMP.
- Trap
  Active o desactive el envío de traps SNMP. Las estaciones especificadas pero no seleccionadas no recibirán traps SNMP.

Grupos v3

Ajustes de seguridad y asignación de derechos

SNMP Versión 3 ofrece asignación de derechos, autentificación y codificación a nivel de protocolo. Los niveles de seguridad y los derechos de lectura/escritura se definen específicamente por grupos. Para cada miembro de un grupo se aplican automáticamente los ajustes correspondientes.

Ajustes

- Nombre del grupo
  Introduzca el nombre del grupo. Este nombre ha de tener al menos dos caracteres y como máximo 32 caracteres.
- Nivel de seguridad
  Seleccione el nivel de seguridad (autentificación, codificación) válido para el grupo seleccionado. En cuanto a los niveles de seguridad existen las posibilidades siguientes:
  - Sin aut./sin priv.
    Autentificación no activada / codificación no activada.
  - Aut./Sin priv.
    Autentificación activada / codificación no activada.
  - Aut./priv.
    Autentificación activada / codificación activada.

La tabla se divide en las siguientes columnas:

- Nombre del grupo
  Muestra los nombres de grupo definidos.

  Nota

  El nombre del grupo y el nivel de seguridad no pueden modificarse una vez creados y asignados. Si desea cambiar el nombre del grupo o el nivel de seguridad, debe borrar el grupo, crearlo con un nuevo nombre y configurarlo de nuevo.

- Nivel de seguridad
  Muestra el nivel de seguridad configurado.
- Lectura
  Active o desactive el acceso de lectura.

- Escritura
  Active o desactive el acceso de escritura.
- Persistencia
  Indica si el grupo está asignado a un usuario SNMPv3. Si el grupo no está asignado a ningún usuario SNMPv3 no se inicia el guardado automático y el grupo configurado desaparecerá tras realizar un rearranque completo del dispositivo.
  - Sí
    El grupo está asignado a un usuario SNMPv3.
  - No
    El grupo no está asignado a ningún usuario SNMPv3.

Usuarios v3

Ajustes de seguridad personalizados

En la página del WBM es posible crear usuarios SNMPv3, modificarlos o borrarlos. El modelo de seguridad basado en el usuario se centra en el nombre del usuario, es decir, se asigna a cada telegrama el identificador de usuario correspondiente. Este nombre de usuario y los ajustes de seguridad en cuestión son comprobados tanto por el remitente como por el destinatario.

Ajustes

- Nombre de usuario
  Introduzca un nombre de usuario cualquiera. Una vez aplicados los datos, no podrá cambiar el nombre.

La tabla se divide en las siguientes columnas:

- Nombre de usuario
  Muestra el usuario creado.
- Nombre del grupo
  Seleccione el grupo al que debe asignarse el usuario.
- Protocolo de autentificación
  Defina el protocolo de autentificación. Solo puede activarse si el grupo soporta la función. Existen los siguientes ajustes:
  - Ninguno
  - MD5
  - SHA
- Protocolo de encriptado
  Especifique si el usuario utiliza el algoritmo DES. Solo activable si el grupo soporta esta función.
- Contraseña de autenticación
  Introduzca la contraseña de autentificación en el primer campo.

- Confirmar contraseña de autenticación
  Confirme la contraseña volviéndola a introducir.
- Contraseña de encriptado
  Introduzca su contraseña de codificación.
- Confirmar contraseña de encriptado
  Confirme la contraseña de codificación volviéndola a introducir.
- Persistencia
  Indica si el usuario está asignado a un grupo SNMPv3. Si el usuario no está asignado a ningún grupo SNMPv3 no se inicia el guardado automático y el usuario configurado desaparecerá tras realizar un rearranque completo del dispositivo.
  - Sí
    El usuario está asignado a un grupo SNMPv3.
  - No
    El usuario no está asignado a ningún grupo SNMPv3.

Hora del sistema

Ajuste manual de la hora

En esta página se ajusta la fecha y la hora del sistema. Para utilizar este ajuste, es necesario activar "Ajuste manual de la hora".

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Ajustes

- Ajuste manual de la hora
  Active o desactive el ajuste manual de la hora.
- Hora del sistema
  Introduzca la fecha y la hora en el formato "MM/DD/YYYY HH:MM:SS". Solo puede editarse cuando está activado el ajuste manual de la hora.
  Tras el reinicio, la hora comienza con 01/01/2000 00:00:00.
- Utilizar la hora del PC
  Haga clic en el botón para aplicar el ajuste de la hora del PC.

- **Momento de la última sincronización**
  Este campo es de solo lectura y muestra cuándo ha tenido lugar la última sincronización horaria. Si no ha sido posible sincronizar la hora, el campo contendrá la entrada "Fecha/ hora no ajustadas".
- **Último mecanismo de sincronización**
  Este campo muestra cómo se ha efectuado la última sincronización horaria.
  - No ajustado
    La hora no se ha ajustado.
  - Manual
    Ajuste manual de la hora
  - SNTP
    Sincronización horaria automática mediante SNTP
  - NTP
    Sincronización horaria automática mediante NTP
  - SIMATIC
    Sincronización horaria automática mediante telegrama horario SIMATIC.
  - PTP
    Sincronización horaria automática mediante PTP

Sinopsis de DST

Esta página permite crear entradas para cambiar el horario de verano.

La tabla proporciona una sinopsis de las entradas existentes.

Ajustes

- **N.º DST**
  Muestra el número de la entrada.
  Al crear una entrada se crea una fila con un número unívoco.
- **Nombre**
  Muestra el nombre de la entrada.
- **Año**
  Indica el año para el que se creó la entrada.
- **Fecha de inicio**
  Indica el mes, el día y la hora del inicio del horario de verano.
- **Fecha de fin**
  Indica el mes, el día y la hora del final del horario de verano.
- **Tipo**
  Indica cómo se realiza el cambio del horario de verano:
  - Fecha
    Se ha introducido una fecha fija para el cambio de horario de verano.
  - Periódicamente
    Hay definida una regla para el cambio de horario de verano.

Configuración DST

Esta página permite configurar las entradas para cambiar el horario de verano. Gracias al cambio del horario de verano o invierno, la hora del sistema está bien ajustada a la zona horaria local.

Se puede definir una regla para el cambio del horario de verano o introducir una fecha determinada.

Ajustes

Nota

El contenido de esta página depende de la selección realizada en el campo "Tipo".

Los campos "N.º DST", "Tipo" y "Nombre" se muestran siempre.

- **N.º DST**
  Seleccione el número de la entrada.
- **Tipo**
  Seleccione cómo debe ser el cambio del horario de verano:
  - Fecha
    Es posible indicar una fecha fija para el cambio de horario de verano.
    Este ajuste es adecuado para regiones en las que el cambio del horario de verano no está sujeto a reglas.
  - Periódicamente
    Se puede definir una regla para el cambio de horario de verano.
    Este ajuste es adecuado para regiones en las que el horario de verano empieza y termina siempre en un día concreto de la semana.
- **Nombre**
  Indique un nombre para la entrada.

Ajustes con la selección "Fecha"

Se puede indicar una fecha fija para el inicio y el final del horario de verano.

- Año
  Indique el año para el cambio de horario de verano.
- Fecha de inicio
  Indique los siguientes valores para el inicio del horario de verano:
  - Día
  - Hora
  - Mes
- Fecha de fin
  Indique los siguientes valores para el final del horario de verano:
  - Día
  - Hora
  - Mes

Ajustes con la selección "Periódicamente"

Se puede crear una regla para el cambio de horario de verano.

- Fecha de inicio
  Indique los siguientes valores para el inicio del horario de verano:
  - Hora
  - Mes
  - Semana
    Se puede elegir entre las semanas 1 a 5 o la última semana del mes.
  - Día de la semana
- Fecha de fin
  Indique los siguientes valores para el final del horario de verano:
  - Hora
  - Mes
  - Semana
    Se puede elegir entre las semanas 1 a 5 o la última semana del mes.
  - Día de la semana

Cliente SNTP

El SNTP (Simple Network Time Protocol) sirve para la sincronización horaria en la red. Los telegramas correspondientes son enviados por un servidor SNTP integrado en la red.

Ajustes

- Cliente SNTP
  Active o desactive la sincronización horaria automática vía SNTP.
- Hora actual del sistema (solo disponible online)
  Muestra los valores de fecha y hora ajustados actualmente en el sistema.

- **Momento de la última sincronización** (solo disponible online)
  Este campo es de solo lectura y muestra cuándo ha tenido lugar la última sincronización horaria. Si no ha sido posible sincronizar la hora, el campo contendrá la entrada "Fecha/ hora no ajustadas".
- **Último mecanismo de sincronización** (solo disponible online)
  Este campo muestra cómo se ha efectuado la última sincronización horaria.
  - No ajustado
    La hora no se ha ajustado.
  - Manual
    Ajuste manual de la hora
  - SNTP
    Sincronización horaria automática mediante SNTP
  - NTP
    Sincronización horaria automática mediante NTP
  - SIMATIC
    Sincronización horaria automática mediante telegrama horario SIMATIC
  - PTP
    Sincronización horaria automática mediante PTP
- **Zona horaria**
  Introduzca la zona horaria utilizada en formato "+/- HH:MM". La zona horaria se basa en el tiempo universal coordinado (UTC). Los ajustes para verano o invierno se consideran en este campo indicando la diferencia horaria.
- **Tipo de sincronización SNTP**
  Seleccione el tipo de sincronización. Son posibles los siguientes tipos de sincronización:
  - Poll
    Si selecciona este tipo de protocolo, los campos de entrada "Dirección IP del servidor SNTP", "Puerto del servidor SNTP" e "Intervalo Poll" se muestran para continuar la configuración. Con este tipo de sincronización, el dispositivo actúa de forma activa y envía solicitudes de hora al servidor SNTP.
  - Listen
    Con este tipo de sincronización, el dispositivo actúa de forma pasiva y "escucha" telegramas SNTP que proporcionan la hora.
- **Dirección IP del servidor SNTP**
  Introduzca la dirección IP del servidor SNTP.
- **Puerto del servidor SNTP**
  Introduzca el puerto del servidor SNTP.
- **Intervalo Poll [s]**
  Introduzca aquí el intervalo de tiempo en segundos entre dos solicitudes de hora.

Cliente NTP

Si la sincronización horaria debe efectuarse mediante NTP, aquí puede realizar los correspondientes ajustes.

Ajustes

- Cliente NTP
  Active o desactive la sincronización horaria automática vía NTP.
- Hora actual del sistema (solo disponible online)
  Muestra los valores de fecha y hora ajustados actualmente en el sistema.
- Momento de la última sincronización (solo disponible online)
  Este campo es de solo lectura y muestra cuándo ha tenido lugar la última sincronización horaria. Si no ha sido posible sincronizar la hora, el campo contendrá la entrada "Fecha/ hora no ajustadas".
- Último mecanismo de sincronización (solo disponible online)
  Este campo muestra cómo se ha efectuado la última sincronización horaria.
  - No ajustado
    La hora no se ha ajustado.
  - Manual
    Ajuste manual de la hora
  - SNTP
    Sincronización horaria automática mediante SNTP
  - NTP
    Sincronización horaria automática mediante NTP
  - SIMATIC
    Sincronización horaria automática mediante telegrama horario SIMATIC
  - PTP
    Sincronización horaria automática mediante PTP
- Zona horaria
  Introduzca la zona horaria utilizada en formato "+/- HH:MM". La zona horaria se basa en el tiempo universal coordinado (UTC). Los ajustes para verano o invierno se consideran en este campo indicando la diferencia horaria.
- Dirección IP del servidor NTP
  Introduzca la dirección IP del servidor NTP.
- Puerto del servidor NTP
  Introduzca el puerto del servidor NTP.
- Intervalo Poll [s]
  Introduzca aquí el intervalo de tiempo en segundos entre dos solicitudes de hora.

SIMATIC Time Client

En esta página se configura la sincronización horaria utilizando el SIMATIC Time Client.

Ajustes

- SIMATIC Time Client
  Active o desactive el SIMATIC Time Client.
- Hora actual del sistema (solo disponible online)
  Muestra los valores de fecha y hora ajustados actualmente en el sistema.

- **Momento de la última sincronización** (solo disponible online)
  Este campo es de solo lectura y muestra cuándo ha tenido lugar la última sincronización horaria. Si no ha sido posible sincronizar la hora, el campo contendrá la entrada "Fecha/ hora no ajustadas".
- **Último mecanismo de sincronización** (solo disponible online)
  Este campo muestra cómo se ha efectuado la última sincronización horaria.
  - No ajustado
    La hora no se ha ajustado.
  - Manual
    Ajuste manual de la hora
  - SNTP
    Sincronización horaria automática mediante SNTP
  - NTP
    Sincronización horaria automática mediante NTP
  - SIMATIC
    Sincronización horaria automática mediante telegrama horario SIMATIC
  - PTP
    Sincronización horaria automática mediante PTP

Cliente PTP

En esta página se configura la sincronización horaria utilizando el PTP (Precision Time Protocol).

Esta función solo está disponible en SCALANCE X500.

Ajustes

- **Cliente PTP**
  Active o desactive la sincronización horaria vía PTP. Se pueden configurar otros ajustes en "Layer 2 > PTP".
- **Hora actual del sistema** (solo disponible online)
  Este campo es de solo lectura y muestra la hora actual del sistema.
- **Momento de la última sincronización** (solo disponible online)
  Este campo es de solo lectura y muestra cuándo ha tenido lugar la última sincronización horaria. Si no ha sido posible sincronizar la hora, el campo contendrá la entrada "Fecha/ hora no ajustadas".

- **Último mecanismo de sincronización** (solo disponible online)
  Este campo muestra cómo se ha efectuado la última sincronización horaria.
  - No ajustado
    La hora no se ha ajustado.
  - Manual
    Ajuste manual de la hora
  - SNTP
    Sincronización horaria automática mediante SNTP
  - NTP
    Sincronización horaria automática mediante NTP
  - SIMATIC
    Sincronización horaria automática mediante telegrama horario SIMATIC
  - PTP
    Sincronización horaria automática mediante PTP
- **Zona horaria**
  Introduzca la zona horaria utilizada en formato "+/- HH:MM". La zona horaria se basa en el tiempo universal coordinado (UTC).

Auto-Logout

En esta página se ajustan los tiempos tras los que se efectúa un cierre de sesión automático en WBM o CLI por inactividad del usuario.

Nota

No hay cierre de sesión automático en la CLI

Si la conexión no finaliza una vez transcurrido el tiempo configurado, verifique el ajuste del mecanismo "Keep alive" en el cliente Telnet.

Si el intervalo es inferior al tiempo configurado, la conexión se mantiene aunque no se transfieran datos de usuario. Ejemplo: para el cierre de sesión automático se han configurado 300 segundos, pero la función "Keep alive" tiene ajustados 120 segundos. En este caso se envía un paquete cada 120 segundos, lo que hace que se mantenga establecida la conexión.

- Desactive el mecanismo Keep Alive (intervalo=0)
  o bien
- Ajuste un intervalo tan alto que la conexión subordinada finalice en caso de inactividad.

Ajustes

- **Web Based Management [s]**
  Introduzca el tiempo en segundos para el cierre de sesión automático del WBM. Si introduce el valor 0, se desactiva el cierre de sesión automático.
- **CLI (TELNET, SSH, Serial) [s]**
  Introduzca el tiempo en segundos para el cierre de sesión automático del CLI. Si introduce el valor 0, se desactiva el cierre de sesión automático.

Pulsador

El pulsador "Select/Set" tiene las siguientes funciones:

- Cambiar el modo de visualización
- Activar el administrador de redundancia
- Restablecer la configuración de fábrica
- Definir la máscara de avisos y los indicadores LED

Encontrará una descripción detallada de las distintas funciones ejecutables con este pulsador en las instrucciones de servicio del dispositivo.

Ajustes

- **Restablecer configuración de fábrica**
  Active o desactive la función "Restablecer configuración de fábrica" del pulsador Select/Set.

  ⚠ **PRECAUCIÓN**

  Función "Restablecer configuración de fábrica" activa en el pulsador al arrancar

  Si se ha desactivado esta función en la configuración, este ajuste solo se mantiene durante el funcionamiento. En caso de arranque, por ejemplo tras un fallo de intensidad, la función está activa hasta que se cargue la configuración, con lo que existe la posibilidad de restablecer la configuración de fábrica incluso sin querer. Esto puede provocar fallos imprevistos en el funcionamiento de la red, pues el dispositivo tiene que volver a configurarse después de que esto suceda. Si hay un PLUG insertado, también se borrará y se pondrá al estado de suministro.

- **Administrador de redundancia**
  Active o desactive el administrador de redundancia con el pulsador Select/Set.
- **Activar la máscara de avisos**
  Active o desactive la función "Definir la máscara de avisos mediante el indicador LED" con el pulsador Select/Set. Esta función solo está disponible en el modo de visualización D.

Cliente Syslog

Syslog según RFC 3164 se usa en la red IP para la transmisión de mensajes de texto cortos y no codificados a través de UDP. Para ello se requiere un servidor de informes del sistema.

Condiciones para el envío de entradas de registro:

- La función de informes del sistema está activada en el dispositivo.
- La función de informes del sistema para el evento correspondiente está activada.
- En su red se encuentra un servidor de informes del sistema que recibe las entradas del registro. Como se trata de una conexión UDP, no hay respuesta al emisor.
- La dirección IP del servidor de informes del sistema está introducida en el dispositivo.

Ajustes

- Cliente Syslog
  Active o desactive la función de informes del sistema.
- Dirección IP del servidor Syslog
  Introduzca la dirección IP del servidor de informes del sistema.

La tabla contiene las siguientes columnas

- Dirección IP del servidor Syslog
  Muestra la dirección IP del servidor de informes del sistema.
- Puerto de servidor
  Introduzca el puerto que utiliza el servidor Syslog.

Puertos

Sinopsis de puertos

La página muestra la configuración para la transferencia de datos para todos los puertos del dispositivo.

Ajustes

- Puerto
  Muestra los puertos configurables.
- Nombre del puerto
  Muestra el nombre del puerto.
- Dirección MAC (solo disponible online)
  Muestra la dirección MAC del puerto.
- Modo de transmisión act. (solo disponible online)
  Muestra los parámetros de transferencia del puerto.
- Negotiation
  Indica si la configuración automática está activada o desactivada.
- Flow Ctrl. Type
  Indica si el control de flujo está activado o desactivado para el puerto.
- Flow Ctrl.
  Indica si el control de flujo trabaja en ese puerto.
- MTU
  Muestra el tamaño máximo de los paquetes.
- Tipo de puerto (solo con enrutamiento)
  Muestra el tipo del puerto. Son posibles los siguientes tipos:
  - Router-Port
  - Puerto Switch VLAN Hybrid
  - Puerto Switch VLAN Trunk

- **Estado**
  Indica si el puerto está activado o desactivado. El tráfico de datos solo es posible a través de un puerto activado.
- **Tipo de medio Combo Port** (solo con SCALANCE XM400)
  Muestra el modo del Combo Port:
  - auto
  - rj45
  - sfp
- **Link** (solo disponible online)
  Indica el estado de conexión con la red. Para el estado de conexión existen las siguientes posibilidades:
  - Up
    El puerto tiene una conexión válida con la red; se recibe una señal de "Link Integrity".
  - Down
    La conexión está interrumpida, por ejemplo por estar desactivado el dispositivo conectado.

Configuración

En esta página se configuran los puertos del dispositivo.

Ajustes

- **Puerto**
  Seleccione los puertos que deben configurarse. El puerto se compone del número de puerto y del número de slot, p. ej. el puerto 0.1 es el slot 0, puerto 1.
- **Estado**
  Defina si el puerto está activado o desactivado.
  - enabled
    El puerto está activado. El tráfico de datos solo es posible a través de un puerto activado.
  - disabled
    El puerto está desactivado pero la conexión sigue activa.
  - Link down
    El puerto está desactivado y la conexión con el dispositivo interlocutor se ha cancelado.
- **Nombre del puerto**
  Introduzca aquí un nombre para el puerto.
- **Dirección MAC** (solo disponible online)
  Muestra la dirección MAC del puerto.

- **Modo de transferencia**
  Indica la velocidad y el método de transferencia del puerto. Los ajustes de "Autonegotiation" y "Velocidad de transferencia" se definen en las opciones del puerto.

  Nota

  Para que el puerto y el puerto interlocutor puedan comunicarse, es necesario que los ajustes coincidan en ambas partes.

- **Modo de transmisión act.** (solo disponible online)
  Indica la velocidad y el método de transferencia que se utilizan actualmente.
- **Negotiation**
  Indica si la configuración automática de la conexión con el puerto interlocutor está activada o desactivada.
- **Flow Ctrl. Type**
  Indica si el control de flujo trabaja en ese puerto.

  Nota

  Activar y desactivar el control de flujo con Autonegotiation

  Solo es posible activar o desactivar el control de flujo si la función "Autonegotiation" está desactivada. Posteriormente puede volver a activarse la función.

- **Flow Ctrl.**
  Indica si el control de flujo trabaja en ese puerto.
- **MTU**
  Introduzca el tamaño de los paquetes.
- **Tipo de puerto** (solo con enrutamiento)
  Seleccione el tipo de puerto:
  - Router-Port
    El puerto es una interfaz IP. No soporta funciones de Layer 2.
  - Puerto Switch VLAN Hybrid
    El puerto envía telegramas con y sin tag. No forma parte automáticamente de una VLAN.
  - Puerto Switch VLAN Trunk
    El puerto envía solo telegramas con tag y forma parte automáticamente de todas las VLANs.

- **Tipo de medio Combo Port** (solo con SCALANCE XM400)
  Defina el modo del Combo Port:
  - auto
    Si selecciona este modo, tendrá prioridad el puerto del transceptor enchufable. Tan pronto como se inserta un transceptor enchufable, se desconecta la conexión existente en el puerto fijo RJ45. Si no hay ningún transceptor enchufable insertado, se puede establecer una conexión por medio del puerto RJ45 fijo.
  - rj45
    Si selecciona este modo, se utilizará el puerto RJ45 fijo, independientemente del puerto del transceptor enchufable.
  - sfp
    Si selecciona este modo, se utilizará el puerto del transceptor enchufable, independientemente del puerto RJ45 fijo.

  El ajuste de fábrica es el modo auto.
- **Link** (solo disponible online)
  Indica el estado de conexión con la red. Para el estado de conexión existen las siguientes posibilidades:
  - Up
    El puerto tiene una conexión válida con la red; se recibe una señal de "Link Integrity".
  - Down
    La conexión está interrumpida, por ejemplo por estar desactivado el dispositivo conectado.

Modificación de la configuración de puertos

Nota

Los puertos ópticos trabajan siempre con el método de transmisión dúplex completo y con la velocidad de transmisión máxima. Por esta razón no puede efectuar los siguientes ajustes en el caso de puertos ópticos:

- Configuración automática
- Velocidad de transmisión
- Procedimiento de transmisión

Nota

En caso de sobrecarga de un puerto, el dispositivo impide o reduce a través de ciertos mecanismos automáticos la retroacción sobre otros puertos y otras clases de prioridad (Class of Service). Estando activado el control de flujo, esto puede hacer que se rechacen telegramas.

Se producen sobrecargas de puertos cuando el dispositivo recibe más telegramas de los que puede enviar, p. ej. debido a velocidades de transmisión diferentes.

Vigilancia de errores

Alimentación

Configure si el sistema de señalización debe vigilar la alimentación. Dependiendo de la variante de hardware, puede haber una o dos conexiones de tensión (alimentación 1 / alimentación 2). Con alimentación redundante, configure la vigilancia para cada uno de los cables de alimentación.

El sistema de señalización indica un error si en una conexión vigilada (alimentación 1 o alimentación 2) falta la tensión o es insuficiente.

Nota

Encontrará los límites de tensión de servicio admisibles en las instrucciones de servicio resumidas del dispositivo.

Un error hace que se encienda el LED de error en el dispositivo. Dependiendo de la configuración, el error puede generar un trap, un e-mail o una entrada en la tabla de registro de eventos.

Ajustes

- **Alimentación 1**
  Active o desactive la vigilancia de la conexión de tensión 1.
- **Alimentación 2**
  Active o desactive la vigilancia de la conexión de tensión 2.

Link Change

En esta página se configura si debe emitirse un aviso de error al producirse una modificación del estado de una conexión de red.

Con la vigilancia de la conexión activada, se señaliza un error

- si en un puerto debe estar disponible un link y este falta.
- o si en un puerto no debe estar disponible ningún link y se detecta uno.

Un error hace que se dispare el contacto de señalización y se encienda el LED de error en el dispositivo. Dependiendo de la configuración, el error puede generar un trap, un e-mail o una entrada en la tabla de registro de eventos.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos.
- **Ajuste**
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Seleccione el ajuste. Existen las siguientes posibilidades:
  - Up
    El tratamiento de errores se inicia al pasarse al estado activo del puerto.
    (De "Link down" a "Link up")
  - Down
    El tratamiento de errores se inicia al pasarse al estado pasivo del puerto.
    (De "Link up" a "Link down")
  - "-" (desactivado)
    El tratamiento de errores no se inicia.

Redundancia

En esta página se configura si debe emitirse un aviso de error al producirse una modificación del estado de una conexión de red.

Ajuste

- **Pérdida de redundancia (solo HRP)**
  Active o desactive la vigilancia de la conexión. Si se pierde la redundancia de la conexión se señaliza un error.

PNIO

En esta página se configura el comportamiento del dispositivo para las entradas y salidas PROFINET.

Descripción

La página contiene los siguientes campos:

- **Modo runtime PNIO**
  Indica el estado de la ejecución de PNIO.
- **Modo runtime PNIO para el próximo arranque**
  Seleccione el modo runtime PNIO que debe activarse la próxima vez que se arranque el dispositivo.
- **Estado PNIO AR**
  Este campo muestra el estado de PROFINET IO Application Relation, es decir, si el IE Switch está conectado a un controlador PROFINET "online" o bien "offline".
  Online significa en este caso que existe una conexión con un PROFINET IO Controller, que el mismo ha cargado sus datos de configuración en el IE Switch y que el dispositivo puede transmitir datos de estado al PROFINET IO Controller. En este estado, que se conoce también como "in Data exchange", no se pueden configurar en el IE Switch los parámetros que se ajustan a través del PROFINET IO Controller.
- **Nombre de dispositivo PROFINET IO**
  Indique nombre del dispositivo PROFINET IO.
- **Permitir el intercambio de datos PNIO**
  Active o desactive el intercambio de datos PNIO.
- **Simular el intercambio de datos PNIO**
  Active o desactive la simulación del intercambio de datos PNIO.

PLUG

Configuración del PLUG

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

ATENCIÓN

No extraer ni insertar el C-PLUG / KEY-PLUG durante el funcionamiento

Un PLUG solo se puede extraer o insertar con el dispositivo desconectado.
El dispositivo comprueba cada varios segundos si hay un PLUG insertado. Si detecta que se ha retirado el PLUG, se produce un rearranque completo del sistema. Si había un KEY-PLUG válido insertado en el dispositivo, este pasa a un estado de error definido después del rearranque.

Información sobre la configuración del C-PLUG / KEY-PLUG

Esta página proporciona información detallada sobre la configuración que está depositada en el C-PLUG o KEY-PLUG. Además existe la posibilidad de restablecer la configuración de fábrica del C-PLUG o de dotarlo de un nuevo contenido.

Nota

La acción no se ejecuta hasta que se hace clic en el botón "Aplicar ajustes".

Esta acción no puede anularse.

Si decide no llevar a cabo la ejecución tras haber efectuado la selección, haga clic en el botón "Actualizar". De este modo, los datos de esta página vuelven a leerse del dispositivo y su selección se anula.

Nota

Incompatibilidad con versiones anteriores con el PLUG insertado

Si se instala una versión anterior, pueden perderse los datos de configuración. En ese caso, tras la instalación del firmware el dispositivo se enciende con los ajustes de fábrica. Si, además, hay un PLUG insertado en el dispositivo, después del rearranque tendrá el estado "NOT ACCEPTED", puesto que en el PLUG siguen estando los datos de configuración del firmware anterior, más actual. Esto permite regresar al firmware más actual que estaba instalado anteriormente sin perder datos de configuración.

Si la configuración original del PLUG ya no se necesita, puede borrarse o sobrescribirse manualmente mediante "Sistema > PLUG".

Ajustes

La tabla se divide en las siguientes filas:

- **Estado**
  Muestra el estado del PLUG. Existen las siguientes posibilidades:
  - ACCEPTED
    Existe un PLUG con una configuración válida y apropiada en el dispositivo.
  - NOT ACCEPTED
    La configuración del PLUG enchufado no es válida o es incompatible.
  - NOT PRESENT
    No hay ningún C-PLUG o KEY-PLUG enchufado en el dispositivo.
  - FACTORY
    El PLUG está enchufado y no contiene ninguna configuración. Este estado también se indica cuando se ha formateado el PLUG durante el funcionamiento.
  - MISSING
    No hay ningún PLUG insertado. En el dispositivo se han configurado funciones que requieren una licencia.
- **Familia de dispositivos**
  Indica qué línea de productos SIMATIC NET ha utilizado el C-PLUG o KEY-PLUG anteriormente.

- Tipo de dispositivo
  Dentro de de la línea de productos, indica el tipo de dispositivo que ha utilizado el C-PLUG o KEY-PLUG anteriormente.
- Versión de la configuración
  La versión de la estructura de configuración. Esta información se refiere a las posibilidades de configuración a que da soporte el dispositivo y no tiene nada que ver con la configuración de hardware concreta. Por lo tanto, el dato de la revisión no cambia si se agregan o se quitan componentes adicionales (p. ej. módulos o Extender), pero puede cambiar si se realiza una actualización del firmware.
- Sistema de archivos
  Indica el tipo del sistema de archivos existente en el PLUG.

ATENCIÓN
Nuevo sistema de archivos UBI A partir de la versión de firmware 3.0, UBI es el sistema de archivos estándar para el C-PLUG o KEY-PLUG. Si se detecta un C-PLUG con el sistema de archivos IECP anterior en uno de estos dispositivos, dicho C-PLUG se formatea para el sistema de archivos UBI y los datos se rescriben en el C-PLUG. El sistema de archivos también se modifica tras una actualización del firmware a V3.0. Un regreso a la versión anterior del firmware correspondiente puede comportar problemas. El firmware no puede leer el C-PLUG o KEY-PLUG ni puede escribir en él; tampoco es posible la función "Restablecer la configuración de fábrica del PLUG".

- Memoria disponible [bytes]
  Indica la capacidad de memoria máxima del sistema de archivos existente en el PLUG.
- Memoria ocupada [bytes]
  Indica la capacidad de memoria ocupada en el sistema de archivos del PLUG.
- Información
  Muestra información adicional sobre el dispositivo que había utilizado el PLUG anteriormente, por ejemplo la referencia, la denominación de tipo y las versiones de hardware y software. La versión de software mostrada equivale a la versión en la que fue modificada la configuración por última vez. Con el estado "NOT ACCEPTED" se muestra información adicional sobre la causa del problema.
- Modificar PLUG
  Elija el ajuste deseado.
  - Escribir la configuración actual en el PLUG
    Esta opción solo está disponible si el estado del PLUG es "NOT ACCEPTED" o "FACTORY".
    La configuración disponible en la memoria flash interna del dispositivo se copia en el PLUG.
  - Restablecer la configuración de fábrica del PLUG
    Borra todos los datos del PLUG y realiza un formateo Low Level.

Licencia PLUG

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

ATENCIÓN

No extraer ni insertar el C-PLUG / KEY-PLUG durante el funcionamiento

Un PLUG solo se puede extraer o insertar con el dispositivo desconectado.
El dispositivo comprueba cada varios segundos si hay un PLUG insertado. Si detecta que se ha retirado el PLUG, se produce un rearranque completo del sistema. Si había un KEY-PLUG válido insertado en el dispositivo, este pasa a un estado de error definido después del rearranque.

Si un dispositivo se ha configurado una vez con un PLUG, ya no podrá utilizarse sin él. Para poder volver a utilizar el dispositivo, restaure la configuración de fábrica.

Nota

Incompatibilidad con versiones anteriores con el PLUG insertado

Si se instala una versión anterior, pueden perderse los datos de configuración. En ese caso, tras la instalación del firmware el dispositivo se enciende con los ajustes de fábrica. Si, además, hay un PLUG insertado en el dispositivo, después del rearranque tendrá el estado "NOT ACCEPTED", puesto que en el PLUG siguen estando los datos de configuración de la versión más actual que estaba instalada antes. Esto permite regresar al firmware más actual que estaba instalado anteriormente sin perder datos de configuración.

Si la configuración original del PLUG ya no se necesita, puede borrarse o sobrescribirse manualmente mediante "Sistema > PLUG".

Información sobre la licencia del KEY-PLUG

Un C-PLUG solo puede guardar la configuración de un dispositivo. Un KEY-PLUG contiene, además de la configuración, una licencia que habilita determinadas funciones del dispositivo SIMATIC NET.

Esta página proporciona información detallada acerca de la licencia del KEY-PLUG. En este ejemplo, el KEY-PLUG contiene los datos para habilitar las funciones de capa 3 del dispositivo.

Valores visualizados

- **Estado**
  Muestra el estado del KEY-PLUG. Existen las siguientes posibilidades:
  - ACCEPTED
    El KEY-PLUG que hay en el dispositivo contiene una licencia adecuada y válida.
  - NOT ACCEPTED
    La licencia del KEY-PLUG insertado no es válida.
  - NOT PRESENT
    No hay ningún KEY-PLUG insertado en el dispositivo.
  - MISSING
    No hay ningún KEY-PLUG insertado o bien hay un C-PLUG insertado con el estado "FACTORY". En el dispositivo se han configurado funciones que requieren una licencia.
  - WRONG
    El KEY-PLUG insertado no concuerda con el dispositivo.
  - UNKNOWN
    Contenido desconocido del KEY-PLUG.
  - DEFECTIVE
    Contenido erróneo del KEY-PLUG.
- **Referencia**
  Muestra la referencia del KEY-PLUG. El KEY-PLUG está disponible para diferentes ampliaciones de función y sistemas de destino.
- **Número de serie**
  Muestra el número de serie del KEY-PLUG.
- **Información**
  Muestra información adicional sobre el dispositivo que había utilizado el KEY-PLUG anteriormente, por ejemplo la referencia, la denominación de tipo y las versiones de hardware y software. La versión de software mostrada equivale a la versión en la que fue modificada la configuración por última vez. Con el estado "NOT ACCEPTED" se muestra información adicional sobre la causa del problema.

Nota

Al guardar la configuración, se guarda también la información de si en ese momento hay un KEY-PLUG insertado en el dispositivo. Esta configuración solo podrá funcionar si hay enchufado un KEY-PLUG con la misma referencia o licencia.

Ping

Accesibilidad de una dirección en una red IP

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

La función Ping permite comprobar si una dirección IP determinada está disponible en la red.

Ajustes

La tabla se divide en las siguientes columnas:

- **Dirección IP**
  Introduzca la dirección IP del dispositivo.
- **Repetir**
  Introduzca el número de peticiones Ping.
- **Ping**
  Haga clic en este botón para iniciar la función Ping.
- **Salida Ping**
  Este campo muestra la salida de la función Ping.

PoE

General

En esta página se define la potencia máxima de la alimentación de tensión (PSE).

PSE (Power Sourcing Equipment)

El SCALANCE X-500 representa un PSE (Power Sourcing Equipment).

En un SCALANCE XM400, la función "Power over Ethernet" puede utilizarse a través del Port Extender PE408PoE. Cada grupo de cuatro puertos con capacidad PoE se considera un PSE. La numeración de los PSEs tiene una asignación fija y no cambia con el número de Port Extenders PoE insertados o con el slot. Pueden utilizarse un máximo de 4 PSEs.

Ajuste

- **PSE**
  Muestra el número de la alimentación de tensión PoE.
- **Potencia máxima**
  Potencia máxima que un PSE proporciona para la alimentación de dispositivos PoE.
- **Potencia distribuida**
  Potencia total reservada por los dispositivos PoE de acuerdo con la clasificación.

- **Potencia utilizada**
  Potencia total utilizada por los dispositivos terminales.
- **Umbral de potencia [%]**
  Cuando la potencia consumida por los terminales sobrepasa el porcentaje indicado aquí, se activa un evento.

Puerto

Ajustes para los puertos

Para cada uno de los puertos PoE se puede definir si debe tener lugar una alimentación de tensión vía Ethernet. Además se puede establecer una prioridad para cada consumidor conectado. En caso necesario, los dispositivos con una prioridad alta reciben un trato preferente respecto a otros en cuanto a la alimentación de tensión.

Ajustes

La tabla se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos PoE configurables.
  El puerto se compone del número de puerto y del número de slot, p. ej. el puerto 0.1 es el slot 0, puerto 1.
- **Ajuste**
  Habilite la alimentación PoE para este puerto o interrúmpala.
- **Prioridad**
  Seleccione la prioridad con la que debe tratarse este puerto en cuanto a la alimentación de tensión.
  Existen las siguientes posibilidades de ajuste con relevancia ascendente:
  - low
    prioridad baja
  - high
    prioridad media
  - critical
    prioridad alta

  Si se ha establecido la misma prioridad para dos puertos, en caso necesario se da preferencia al puerto con el número más bajo.
- **Tipo**
  Aquí se puede introducir una cadena de caracteres que describe con mayor detalle el dispositivo conectado.
- **Clasificación**
  La clasificación indica la clase de dispositivo, además de la potencia máxima del mismo.

- **Estado**
  Indica el estado del puerto.
  Existen los siguientes estados:
  - disabled
    La alimentación PoE para este puerto está desactivada.
  - delivering
    La alimentación PoE para este puerto está activada y hay un dispositivo conectado.
  - searching
    La alimentación PoE para este puerto está activada pero no hay ningún dispositivo conectado.

Nota

Cuando un dispositivo se conecta a un puerto con capacidad PoE, se comprueba si la potencia del puerto es suficiente para el dispositivo conectado. Si la potencia del puerto no es suficiente, PoE estará activado en "Ajuste", pero el puerto tendrá el estado "disabled". En ese caso el puerto ha sido desactivado por el PoE Power Management.

- **Potencia [mW]**
  Indica la potencia que suministra SCALANCE a este puerto.
- **Tensión [V]**
  Indica la tensión que llega a este puerto.
- **Intensidad [mA]**
  Indica la intensidad que se suministra a un dispositivo de este puerto.

Diagnóstico del puerto

Comprobador de cables

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Con esta página, cada uno de los puertos Ethernet puede realizar un diagnóstico de errores independiente en el cable. Esta prueba se realiza sin que el cable esté desenchufado, sin estar conectado un comprobador de cables y sin que esté instalado un módulo Loopback en el otro extremo. Con esto se pueden localizar cortocircuitos e interrupciones de cables con una precisión de pocos metros.

Nota

Tenga en cuenta que esta prueba solo se permite si en el puerto a comprobar no está establecida ninguna conexión de datos.

Ajustes

- **Puerto**
  Seleccione el puerto deseado en la lista desplegable.
- **Realizar test**
  Activa el diagnóstico de errores. El resultado se representa en la tabla.

La tabla contiene las siguientes columnas:

- **Par**
  Indica el par de conductores del cable.

  Nota

  Pares de conductores

  En cables de red de 10/100 Mbits no se utilizan los pares de conductores 4-5 ni 7-8.

  En cables de 1000 Mbits o Gigabit Ethernet se utilizan los 4 pares de conductores.

  La correspondencia entre par de conductores y asignación de pines es la siguiente (DIN EN 50173):

  Par 1 = pin 4-5

  Par 2 = pin 1-2

  Par 3 = pin 3-6

  Par 4 = pin 7-8

- **Estado**
  Indica el estado del cable.
- **Distancia [m]**
  Indica la distancia hasta el extremo del cable, el punto de rotura del cable o el cortocircuito.

Diagnóstico SFP

En esta página puede ejecutar un diagnóstico de errores independiente para cada puerto SFP. Esta prueba se realiza sin necesidad de desenchufar ningún cable, conectar un comprobador de cables o instalar un módulo Loopback en el otro extremo.

Nota

Tenga en cuenta que esta prueba solo se permite si en el puerto a comprobar no está establecida ninguna conexión de datos.

En caso de haber una conexión de datos en él, se interrumpirá durante un breve intervalo de tiempo.

El restablecimiento automático de la conexión puede fracasar; en dicho caso habrá que restablecerla manualmente.

Descripción

La página contiene los siguientes campos:

- Puerto
  Seleccione el puerto deseado en la lista desplegable.

Los valores se muestran en los siguientes campos:

- Nombre
  Muestra el nombre de la interfaz.
- Modelo
  Muestra el tipo de interfaz.
- Versión
  Indica la versión de hardware de SFP.
- Serie
  Muestra el número de serie de SFP.
- Tasa de bits nominal [Mbits/s]
  Muestra la tasa de bits nominal de la interfaz.
- Max. Link (50.0/125um) [m]
  Muestra la distancia máxima en metros, posible con este medio.
- Max. Link (62.5/125um) [m]
  Muestra la distancia máxima en metros, posible con este medio.

En la siguiente tabla se muestran los valores del transceptor enchufable SFP utilizado en este puerto:

- Temperatura [°C]
  Muestra la temperatura de la interfaz.
- Tensión [V]
  Muestra la tensión [V] aplicada a la interfaz.
- Intensidad [mA]
  Muestra el consumo de corriente [mA] de la interfaz.
- Potencia Rx [mW]
  Muestra la potencia de recepción [mW] de la interfaz.
- Potencia Tx [mW]
  Muestra la potencia de emisión [mW] de la interfaz.
- Actual
  Muestra el valor actual.
- Baja
  Muestra el valor más bajo.
- Alta
  Muestra el valor más alto.

Layer 2

Configuración

En esta página se configuran las funciones de la Layer 2. Para algunas funciones hay más páginas de configuración, en las que se pueden realizar ajustes detallados. En las páginas de configuración también pueden comprobarse los ajustes.

Ajustes

- VLAN basada en protocolo
  Active o desactive la VLAN basada en protocolo. Se pueden configurar otros ajustes en "Layer 2 > VLAN".
- VLAN basada en subred
  Active o desactive la VLAN basada en subred. Se pueden configurar otros ajustes en "Layer 2 > VLAN".
- Dynamic MAC Aging
  Active o desactive el mecanismo "Aging". Se pueden configurar otros ajustes en "Layer 2 > Dynamic MAC Aging".
- Tipo de redundancia
  Existen los siguientes ajustes:
  - "-" (desactivado)
    La función de redundancia está desactivada.
  - Ring
    Redundancia de anillo activada. Se pueden configurar otros ajustes en "Layer 2 > Redundancia de anillo > Anillo".
  - Spanning Tree
    Seleccionando esta opción se especifica el modo de redundancia deseado en "Método de redundancia".

- **Método de redundancia**
  Si selecciona "Spanning Tree" en "Tipo de redundancia", dispone de las siguientes posibilidades de selección:
  - **STP**
    Activa el Spanning Tree Protocol (STP). Los tiempos de reconfiguración típicos para Spanning Tree se sitúan entre 20 y 30 segundos. Se pueden configurar otros ajustes en "Layer 2 > Spanning Tree".
  - **RSTP**
    Activa el Rapid Spanning Tree Protocol (RSTP). Si en un puerto se detecta un telegrama Spanning Tree, este puerto retrocede de RSTP a Spanning Tree. Se pueden configurar otros ajustes en "Layer 2 > Spanning Tree".

    Nota

    Con RSTP (Rapid Spanning Tree Protocol) pueden producirse brevemente bucles con duplicación o rebases de telegramas. Si esto no es aceptable para su aplicación, deberá utilizar el procedimiento estándar Spanning Tree, más lento.
  - **MSTP**
    Activa el Multiple Spanning Tree Protocol (MSTP). Se pueden configurar otros ajustes en "Layer 2 > Spanning Tree".

  Si selecciona "Anillo" en "Tipo de redundancia", dispone de las siguientes posibilidades de selección:
  - **Automatic Redundancy Detection**
    Seleccione este ajuste para una configuración automática del método de redundancia de anillo. En este modo, el dispositivo determina automáticamente si hay un dispositivo con la función de "Administrador HRP" en el anillo. De ser así, el dispositivo adopta la función de "Cliente HRP". Si no se encuentra ningún "Administrador HRP", todos los dispositivos con el ajuste "Detección automática de redundancia" o "Administrador automático MRP" negocian entre sí cuál de ellos adopta la función de "Administrador MRP". El dispositivo con la dirección MAC de valor más bajo siempre pasa a ser "Administrador MRP". Los restantes dispositivos adoptan automáticamente el tipo de redundancia de anillo "Cliente MRP".
  - **MRP Auto-Manager**
    Administrador automático de la redundancia de medios
  - **MRP Client**
    Cliente de la redundancia de medios
  - **HRP Client**
    Cliente High Speed Redundancy Protocol
  - **HRP Manager**
    Administrador High Speed Redundancy Protocol
- **Standby**
  Active o desactive la función "Redundancia Standby". Se pueden configurar otros ajustes en "Layer 2 > Redundancia de anillo > Standby".
- **Passive Listening**
  Active o desactive la función "Passive Listening".

- RMON
  Si activa esta casilla de verificación, Remote Monitoring (RMON) permite recopilar datos de diagnóstico en el dispositivo, procesarlos y extraerlos a través de SNMP de una estación de gestión de red que soporte también RMON. Estos datos de diagnóstico, como pueden ser evoluciones de carga referidas a puertos, permiten detectar a tiempo problemas en la red y eliminarlos.
- Multicast dinámico
  Son posibles los siguientes ajustes:
  - "-" (desactivado)
  - IGMP Snooping
    Activa IGMP (Internet Group Management Protocol). Se pueden configurar otros ajustes en "Layer 2 > Multicast > IGMP".
  - GMRP
    Activa GMRP (GARP Multicast Registration Protocol). Se pueden configurar otros ajustes en "Layer 2 > Multicast > GMRP".

  Nota

  GMRP e IGMP no se pueden utilizar al mismo tiempo.

- GVRP
  Active o desactive "GVRP" (GARP VLAN Registration Protocol). Se pueden configurar otros ajustes en "Layer 2 > VLAN > GVRP".
- Mirroring
  Active o desactive la duplicación de puertos. Se pueden configurar otros ajustes en "Layer 2 > Mirroring > Port".
- Loop Detection
  Active o desactive la detección de bucles. Se pueden configurar otros ajustes en "Layer 2 > Loop Detection".
- PTP
  Especifique cómo el dispositivo debe procesar mensajes PTP. Se pueden configurar otros ajustes en "Layer 2 > PTP".
  - Off
    El dispositivo no procesa mensajes PTP. Sin embargo, los mensajes PTP son transferidos siguiendo las reglas del switch.
  - transparent
    El dispositivo adopta la función de un Transparent Clock y transfiere los mensajes PTP a otros dispositivos, no sin antes haber realizado entradas en el campo de corrección del mensaje PTP.

QoS

CoS-Map

En esta página se asignan prioridades CoS a determinadas colas de espera (Traffic Queues).

Ajustes

- **CoS**
  Indica la prioridad CoS de los paquetes entrantes.
- **Queue**
  Seleccione la cola de espera de reenvío (prioridad de emisión) a la que se asigna la prioridad CoS.
  Cuanto mayor sea el número de la cola de espera, mayor será la prioridad de envío.

DSCP-Map

En esta página se asignan ajustes de DSCP a diversas colas de espera (Traffic Queues).

Ajustes

- **DSCP**
  Indica la prioridad DSCP de los paquetes entrantes.
- **Queue**
  Seleccione la cola de espera de reenvío (prioridad de emisión) a la que se asigna el valor DSCP. Cuanto mayor sea el número de la cola mayor será la prioridad de envío.

Limitación de carga

Limitación de la velocidad de transferencia de datos entrantes y salientes

En esta página se configura la limitación de carga (número máximo de paquetes de datos por segundo) para cada uno de los puertos. Se puede definir a qué categoría de telegramas se aplicarán estos valores límite.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos.
- **Limit Ingress Unicast (DLF) / Limit Ingress Broadcast / Limit Ingress Multicast**
  Elija el ajuste deseado.
  - enabled: activa la función
  - disabled: desactiva la función
  - No Change: el ajuste de la tabla 2 no cambia
- **Velocidad de transmisión Ingress total [pkts/s]**
  Especifique el número máximo de paquetes entrantes que son procesados por el dispositivo. Si se ha introducido "No Change", la entrada de la tabla 2 permanece invariable.

- **Velocidad de transmisión Egress [kb/s]**
  Especifique la velocidad de transferencia de datos para todos los telegramas salientes. Si se ha introducido "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplican los ajustes a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Indica el puerto al que se refieren los ajustes.
- **Limit Ingress Unicast (DLF)**
  Active o desactive la velocidad de transferencia de datos para limitar los telegramas Unicast entrantes con dirección indisoluble (Destination Lookup Failure).
- **Limit Ingress Broadcast**
  Active o desactive la velocidad de transferencia de datos para limitar los telegramas Broadcast entrantes.
- **Limit Ingress Multicast**
  Active o desactive la velocidad de transferencia de datos para limitar los telegramas Multicast entrantes.
- **Velocidad de transmisión Ingress total [pkts/s]**
  Especifique el número máximo de paquetes entrantes que son procesados por el dispositivo.
- **Velocidad de transmisión Egress [kb/s]**
- Especifique la velocidad de transferencia de datos para todos los telegramas salientes.

> **Nota**
>
> **Redondeo de los valores, divergencia respecto al valor predeterminado**
>
> Al indicar los valores para las velocidades de transferencia, tenga en cuenta que el WBM redondea a valores correctos.
>
> En caso de haber valores configurados para Velocidad de transmisión Ingress total y Velocidad de transmisión Egress, los valores reales durante el funcionamiento pueden rebasar los valores ajustados en un 10% por defecto o por exceso.

VLAN

General

En esta página se define la VLAN y se fija el uso de los puertos.

> **Nota**
>
> **Modificación de la "ID VLAN agente"**
>
> Cuando el PC de configuración está conectado al dispositivo directamente a través de Ethernet y se modifica la "ID VLAN agente", el dispositivo deja de estar accesible vía Ethernet después de la modificación.

Reglas importantes para VLAN

Tenga en cuenta las siguientes reglas para la configuración y la operación de sus VLAN:

- Los telegramas con la ID de VLAN "0" se tratan como telegramas sin tag, pero conservan su valor de prioridad.
- Todos los puertos del dispositivo envían por defecto telegramas sin identificador (tag) de VLAN, para estar seguro de que el dispositivo terminal pueda recibir dichos telegramas.
- En los dispositivos SCALANCE X está preconfigurada la ID de VLAN "1" para todos los puertos.
- Si en un puerto hay conectado un dispositivo terminal, los telegramas salientes se deben enviar sin identificador (puerto de acceso estático). Si en ese puerto hay otro switch, el telegrama se debe dotar de identificador (Trunk Port).
- En un Trunk Port la asignación de VLAN es dinámica. Solo es posible realizar configuraciones estáticas si el puerto, además de la propiedad Trunk Port, se registra estáticamente como miembro de las VLANs correspondientes. Un ejemplo de configuración estática es la asignación de grupos Multicast en determinadas VLANs.

Ajustes

- ID de VLAN
  Introduzca la ID de VLAN.

La tabla se divide en las siguientes columnas:

- ID de VLAN
  Indica la ID de VLAN. La ID de VLAN se asigna una sola vez al crear un registro y no puede modificarse posteriormente. Para modificarla es necesario borrar y volver a crear todo el registro.
- Nombre
  Introduzca un nombre para la VLAN. El nombre solo tiene carácter informativo y no influye en la configuración.
  El nombre de la VLAN debe tener como máximo 32 caracteres.

- Estado
  Muestra el estado de la entrada. Si aparece "estático", significa que la dirección ha sido introducida estáticamente por el usuario. La indicación "GVRP" significa que la configuración se ha registrado a través de un telegrama GVRP. Sin embargo, para ello es necesario haber activado GVRP para el dispositivo.
- Lista de puertos
  Defina el uso del puerto. Existen las siguientes posibilidades:
  - "-"
  - El puerto no es miembro de la VLAN indicada.
    En el momento de la nueva definición, todos los puertos están identificados con "-".
  - M
    El puerto es miembro de la VLAN. Los telegramas enviados en esta VLAN llevan el identificador (tag) de VLAN correspondiente.
  - R
    El puerto es miembro de la VLAN. El registro se produce mediante un telegrama GVRP.
  - U (mayúscula)
    El puerto es miembro sin identificador de la VLAN. Los telegramas enviados en esta VLAN no llevan identificador (tag) de VLAN. Desde ese puerto se envían telegramas sin identificador de VLAN.
  - u (minúscula)
    El puerto es miembro sin identificador de la VLAN, pero la VLAN no está configurada como VLAN del puerto. Los telegramas enviados en esta VLAN no llevan identificador (tag) de VLAN.
  - F
    El puerto no es miembro de la VLAN indicada y no es posible que la VLAN se registre dinámicamente en ese puerto mediante GVRP. Se pueden configurar otros ajustes en "Layer 2 > VLAN > VLAN basada en puerto".

GVRP

Configuración de la función GVRP

Un telegrama GVRP permite que otro dispositivo se registre en el puerto del dispositivo para un VID determinado. Otro dispositivo puede ser p. ej. un dispositivo terminal o un switch. Además, el dispositivo también puede enviar telegramas GVRP a través de este puerto.

Ajustes

- GVRP
  Active o desactive la función "GVRP".

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste. Existen las siguientes posibilidades de configuración:
  - enabled
    Activa el envío de telegramas GVRP.
  - disabled
    Desactiva el envío de telegramas GVRP.
  - No Change
    No hay cambios en la tabla 2.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2. Si se ha seleccionado "No Change", el contenido de la tabla 2 no cambia.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Active o desactive el envío de telegramas GVRP.

VLAN basada en puerto

Procesamiento de telegramas recibidos

En esta página se define la configuración de las propiedades de puerto para la recepción de telegramas.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos.
- **Prioridad / VID de puerto / Tipos de telegrama aceptados / Filtrado Ingress**
  Seleccione el ajuste. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- Puerto
  Muestra los puertos y Link Aggregations disponibles.
- Prioridad
  Seleccione la prioridad que se da a telegramas sin identificador.
  La prioridad CoS (Class of Service) utilizada en el tag de VLAN. Si se recibe un telegrama sin tag, se le asigna esta prioridad. Esta prioridad define cómo se procesa este telegrama en comparación con otros telegramas.
  Existen en total ocho prioridades con los valores 0 a 7, siendo 7 la prioridad más alta (IEEE 802.1p Port Priority).
- VID de puerto
  Seleccione la ID de VLAN. Solo pueden seleccionarse las IDs de VLAN que se han definido en la página "VLAN > General".
  Si un telegrama recibido no tiene ningún identificador de VLAN, se complementa con un identificador que lleva la ID de VLAN aquí indicada, y se envía a través del puerto según las reglas correspondientes.
- Tipos de telegrama permitidos
  Especifique los tipos de telegrama que se aceptan. Existen las siguientes alternativas:
  - Tagged Frames Only
    El dispositivo rechaza todos los telegramas sin tag. De lo contrario rigen las reglas de envío según la configuración.
  - All
    El dispositivo reenvía todos los telegramas
- Ingress Filtering
  Especifique si la VID de los telegramas recibidos se evalúa.
  Existen las siguientes posibilidades:
  - enabled
    La ID de VLAN de telegramas recibidos determina el reenvío: Para el envío de un telegrama con tag de VLAN, el puerto receptor debe ser miembro de la misma VLAN. En el puerto receptor se rechazan telegramas de VLANs desconocidas.
  - disabled
    Todos los telegramas se reenvían.

Grupo VLAN basado en protocolo

En esta página se definen grupos a los que se asigna un protocolo.

Ajustes

- VLAN basada en protocolo
  Active o desactive la asignación de VLAN basada en protocolo.
- Valor de protocolo
  Introduzca el valor de protocolo hexadecimal.
  Algunos ejemplos son:
  - Profinet: 88:92
  - IP: 08:00
  - Novell: 81:37
  - netbios: f0:f0
  - appletalk: 80:9b
- ID del grupo
  Introduzca la ID del grupo.

La tabla se divide en las siguientes columnas:

- Valor de protocolo
  Muestra el valor del protocolo.
- ID del grupo
  Muestra la ID de grupo.

Puerto VLAN basado en protocolo

En esta página se define qué protocolo y qué VLAN se asignan a cada puerto.

Ajustes

- Puerto
  Seleccione el puerto deseado. Se muestran todos los puertos disponibles y las Link Aggregations.
- ID del grupo
  Seleccione la ID del grupo en la lista desplegable. La ID se define en "Layer 2 > VLAN > Grupo VLAN basado en protocolo".

La tabla se divide en las siguientes columnas:

- Puerto
  Se muestran todos los puertos disponibles y las Link Aggregations.
- ID del grupo
  Muestra la ID de grupo asignada al puerto.
- VLAN ID
  Seleccione la ID de VLAN que debe asignarse al puerto.

VLAN basada en subred Ipv4

En esta página se define qué ID de VLAN se asigna a la subred.

Ajustes

- VLAN basada en subred
  Active o desactive la asignación de VLAN basada en subred.
- Puerto
  Seleccione el puerto. Se muestran todos los puertos disponibles y las Link Aggregations.
- Dirección de subred
  Introduzca la dirección IP de la subred.
  Ejemplo: 192.168.10.0 para la red 192.168.10.x con los dispositivos 192.168.10.1 a 192.168.10.254.
- Máscara de subred
  Introduzca la máscara de subred.

La tabla se divide en las siguientes columnas:

- Puerto
  Se muestran todos los puertos disponibles y las Link Aggregations.
- Dirección de subred
  Muestra la subred asignada al puerto.
- Máscara de subred
  Muestra la máscara de subred.
- VLAN ID
  Seleccione la ID de VLAN que desea asignar al puerto o a la subred.

Mirroring

Principios básicos

Mirroring

El dispositivo ofrece la posibilidad de desviar flujos de datos entrantes o salientes paralelamente a otras interfaces para analizarlos o visualizarlos. Esto no produce perturbaciones en los flujos de datos observados. Este método se denomina "mirroring" o duplicado. En este punto del menú se activa y desactiva la función de duplicado y se ajustan parámetros.

Duplicación de puertos

Duplicar un puerto significa que el tráfico de datos en un puerto (puerto duplicado) del IE Switch se copia a otro puerto (puerto de monitor). En un puerto de monitor es posible duplicar uno o varios puertos.

Si se conecta un analizador de protocolos al puerto de monitor, puede registrarse el tráfico de datos en el puerto duplicado sin interrumpir la conexión en este. Esto permite investigar sin repercusiones el tráfico de datos. Condición necesaria para ello es que el dispositivo disponga de un puerto libre para actuar como puerto monitor.

General

En esta página se activa o desactiva la función Mirroring y se llevan a cabo los ajustes básicos.

Nota

Si la tasa de datos máxima del puerto duplicado es mayor que la del puerto monitor, se pueden producir pérdidas de datos y el puerto monitor ya no reproduciría los procesos del puerto duplicado. En un puerto monitor se pueden duplicar varios puertos simultáneamente.

La duplicación de un puerto no funciona cuando se sobrepasan los límites del núcleo del switch.

Debe desactivar la duplicación de puertos si conecta un terminal normal al puerto monitor.

Ajustes

- **Mirroring**
  Active o desactive la duplicación del tráfico de datos.
- **Monitor Barrier**
  Active o desactive esta opción para limitar la comunicación a través del puerto de monitor.

  Nota

  Monitor Barrier

  Si se activa Monitor Barrier se bloquea automáticamente el tráfico de datos en el puerto de destino (Broadcast, Multicast, Unicast, DCP-forwarding, LLDP), de modo que solo queda el tráfico duplicado. Para volver a permitir otro tipo de tráfico de datos es necesario configurarlo. Los estados anteriores de estas opciones no se restablecen una vez finalizado Monitor Barrier y hay que volver a configurarlos.

  – enabled
    El puerto Monitor está excluido del switching normal de telegramas.
  – disabled
    No existe ninguna restricción para la comunicación a través del puerto Monitor.

La tabla contiene las siguientes columnas:

- **Session ID**
  Active o desactive en el puerto deseado la escucha de los paquetes entrantes.
- **Tipo de sesión**
  Especifique el tráfico de datos que se duplicará. Existen las siguientes posibilidades:
  - '-'
    ninguno
  - Port Based
    Duplicado basado en puerto. Se pueden configurar otros ajustes en "Layer 2 > Mirroring > Port".
  - VLAN
    Duplicado basado en VLAN. Se pueden configurar otros ajustes en "Layer 2 > Mirroring > VLAN".
  - MAC ACL
    Duplicado de la MAC Access Control List. Se pueden configurar otros ajustes en "Layer 2 > Mirroring > MAC Flow".
  - IP ACL
    Duplicado de la IP Access Control List. Se pueden configurar otros ajustes en "Layer 2 > Mirroring > IP Flow".
- **Estado**
  Indica si el duplicado está activo o no.
- **Puerto de destino**
  Elija el puerto de destino en el que se realizará el duplicado para esta sesión.

Puerto

Duplicación de puertos

Los ajustes de esta página solo se pueden configurar si previamente se ha generado una ID de sesión de tipo "Port Based" en la pestaña "General".

Ajustes

- **ID de sesión**
  Seleccione la ID de sesión. Solo están disponibles IDs del tipo de sesión "Port Based".

La tabla se divide en las siguientes columnas:

- **Puerto**
  Indica el puerto que debe vigilarse.
- **Ingress Mirroring**
  Active o desactive en el puerto deseado el duplicado de los paquetes entrantes.
- **Egress Mirroring**
  Active o desactive en el puerto deseado el duplicado de los paquetes salientes.

VLAN

Orígenes VLAN de la función de duplicación de puertos

Los ajustes de esta página solo se pueden configurar si previamente se ha generado una ID de sesión de tipo "VLAN" en la pestaña "General".

En esta página se especifica la VLAN cuyo tráfico de datos entrante se duplicará en el puerto monitor.

Ajustes

- ID de sesión
  Seleccione la ID de sesión. Solo están disponibles IDs del tipo de sesión "VLAN".
- VLAN ID
  Introduzca una ID de VLAN. La ID de VLAN solo puede asignarse una sola vez al crear un registro y no puede modificarse posteriormente. Para modificarla es necesario borrar y volver a crear todo el registro.

La tabla se divide en las siguientes columnas:

- VLAN ID
  Indica la ID de VLAN.

MAC Flow

Los ajustes de esta página solo se pueden configurar si previamente se ha generado una ID de sesión de tipo "MAC ACL" en la pestaña "General".

El filtro MAC ACL determina los datos que se preparan en el puerto de monitor.

Ajustes

- ID de sesión
  Seleccione la ID de sesión. Solo están disponibles IDs del tipo de sesión "MAC ACL".

La tabla se divide en las siguientes columnas:

- Número de filtro ACL
  Muestra el número del filtro ACL. El filtro ACL MAC se configura en "Security > Puerto ACL MAC".
- Ingress Mirroring
  Indica si se duplican paquetes entrantes.

Nota

Reglas

Una regla que se haya seleccionado para el Ingress Mirroring solo se activará si se ha configurado como regla Port Ingress en un puerto como mínimo. Las reglas Port Ingress se configuran en "Security > Puerto MAC IP > Reglas Port Ingress".

- Dirección MAC de origen
  Muestra la dirección MAC del remitente.
- Dirección MAC de destino
  Muestra la dirección MAC del destinatario.
- Ingress Port
  Muestra una lista de todos los puertos para los que rige la regla.
- Egress Port
  Muestra una lista de todos los puertos para los que rige la regla.

IP Flow

Los ajustes de esta página solo se pueden configurar si previamente se ha generado una ID de sesión de tipo "IP ACL" en la pestaña "General".

El filtro IP ACL determina los datos que se duplican en el puerto de monitor.

Ajustes

- ID de sesión
  Seleccione la ID de sesión. Solo están disponibles IDs del tipo de sesión "IP ACL".

La tabla se divide en las siguientes columnas:

- Número de filtro ACL
  Muestra el número del filtro ACL. El filtro ACL IP se configura en "Security > Puerto ACL IP".
- Ingress Mirroring
  Indica si se duplican paquetes entrantes.

  Nota
  Reglas

  Una regla que se haya seleccionado para el Ingress Mirroring solo se activará si se ha configurado como regla Port Ingress en un puerto como mínimo. Las reglas Port Ingress se configuran en "Security > Puerto ACL IP > Reglas Port Ingress".

- Dirección IP de origen
  Muestra la dirección IP del remitente.
- Máscara de subred de origen
  Muestra la máscara de subred del remitente.
- Dirección IP de destino
  Muestra la dirección IP del destinatario.
- Máscara de subred de destino
  Muestra la máscara de subred del destinatario.
- Ingress Port
  Muestra una lista de todos los puertos Ingress para los que rige la regla.
- Egress Port
  Muestra una lista de todos los puertos Egress para los que rige la regla.

Dynamic MAC Aging

El dispositivo aprende automáticamente las direcciones de origen de los dispositivos conectados.

Esta información se utiliza en el dispositivo para transmitir directamente telegramas de datos a los dispositivos afectados. Se reduce la carga de red para los demás dispositivos.
Si, dentro un determinado período de tiempo, un dispositivo no recibe ningún telegrama cuya dirección de origen coincida con una dirección aprendida, se borra la dirección en cuestión. A este mecanismo se le da el nombre de "Aging". Mediante Aging se evita que se envíen telegramas a direcciones incorrectas, si por ejemplo un terminal (p. ej. una programadora) se conecta a otro puerto. Si la opción no está activada, el dispositivo no borra automáticamente las direcciones aprendidas.

Ajustes

- Dynamic MAC Aging
  Active o desactive la función para el Aging automático de direcciones MAC aprendidas:
- Aging Time [s]
  Introduzca el intervalo en segundos. Tras ese intervalo se borra una dirección aprendida si el dispositivo no recibe más telegramas de esa dirección.

Redundancia de anillo

Anillo

En esta página se selecciona el modo de operación deseado para la redundancia rápida del anillo. Al entregarse el dispositivo está preajustado el método de redundancia de anillo "Automatic Redundancy Detection".

Nota

La redundancia de anillo no puede activarse si en el dispositivo sigue activado Spanning Tree.

Ajustes

- **Redundancia de anillo**
  Active o desactive la redundancia de anillo.
- **Método de redundancia de anillo**
  Defina el modo de redundancia del anillo. Existen las siguientes posibilidades:
  - **"-" (desactivado)**
    La función de redundancia está desactivada.
  - **Automatic Redundancy Detection**
    Seleccione este ajuste para una configuración automática del método de redundancia de anillo. En este modo, el dispositivo determina automáticamente si hay un dispositivo con la función de "Administrador HRP" en el anillo. De ser así, el dispositivo adopta la función de "Cliente HRP". Si no se encuentra ningún "Administrador HRP", todos los dispositivos con el ajuste "Detección automática de redundancia" o "Administrador automático MRP" negocian entre sí cuál de ellos adopta la función de "Administrador MRP". El dispositivo con la dirección MAC de valor más bajo siempre pasa a ser "Administrador MRP". Los restantes dispositivos adoptan automáticamente el método de redundancia de anillo "Cliente MRP".
  - **MRP Auto Manager**
    Los dispositivos con el ajuste "Detección automática de redundancia" o "Administrador automático MRP" negocian entre sí cuál de ellos adopta el papel de "Administrador MRP". El dispositivo con la dirección MAC de valor más bajo siempre pasa a ser "Administrador MRP". A diferencia del ajuste "Detección automática de redundancia", los dispositivos no son capaces de reconocer si hay un "Administrador HRP en el anillo. Por lo tanto no adoptan nunca la función de "Cliente HRP".
  - **MRP Client**
    En un anillo cuyos dispositivos están configurados con MRP, al menos un dispositivo debe estar ajustado al modo "Detección automática de redundancia" o bien "Administrador automático MRP". En todos los demás dispositivos puede estar ajustada la función de "Cliente MRP". Si todos los dispositivos del anillo, menos uno, se configuran como "Cliente MRP", dicho dispositivo adopta automáticamente la función de "Administrador MRP".
    Seleccione el modo de operación "Cliente MRP" si desea utilizar el dispositivo en el anillo junto con componentes que no sean de Siemens.
  - **HRP Client**
    Aquí puede seleccionar la función de "Cliente HRP".
  - **HRP Manager**
    Al configurar un anillo HRP, debe ajustarse un dispositivo como "Administrador HRP". Todos los demás dispositivos deben configurarse como "Cliente HRP".

- **Puertos en anillo**
  Especifique los puertos que se deben utilizar como puertos en anillo para la redundancia de medios en topologías de anillo.
  El puerto en anillo seleccionado en el menú desplegable anterior es el "Isolated Port" en HRP.

  Nota

  Al restablecer los ajustes de fábrica se activa el método preajustado de redundancia de anillo "Detección automática de redundancia".

  Además se restablecen los puertos preajustados de fábrica para la configuración de puertos de anillo. Si antes se han utilizado otros puertos como puertos de anillo y se establece la conexión correspondiente, un dispositivo configurado previamente de forma correcta puede provocar una circulación continua de telegramas y, con ello, el fallo del tráfico de datos.

- **Observer**
  Active o desactive el Observer. La función "Observer" solo está disponible en anillos HRP.
  El puerto en anillo seleccionado en el menú desplegable anterior se conecta al "Isolated Port" de un administrador HRP.
  El Observer vigila funciones incorrectas del administrador de redundancia o configuraciones incorrectas de un anillo HRP.
  Cuando el Observer está activado puede interrumpir el anillo conectado en caso de detectar errores. Para ello, el Observer conmuta un puerto del anillo al estado "blocking". Una vez solucionado el error, el Observer vuelve a habilitar el puerto.
- **Reiniciar Observer**(solo disponible online)
  Si se producen muchos errores continuados, el Observer no vuelve a habilitar su puerto por cuenta propia. El puerto en anillo permanece en estado "blocking". Esto se señaliza mediante el LED de error y un texto de aviso.
  Una vez solucionado el error puede volver a habilitar el puerto pulsando el botón "Reiniciar Observer".

Standby

Standby-Manager

El Standby-Manager permite el acoplamiento redundante de dos anillos HRP. Para esto se tienen que configurar dentro de un anillo dos dispositivos vecinos como interlocutores Standby. Active el Standby-Manager para ambos interlocutores Standby y seleccione a través de qué puerto está conectado el dispositivo con el anillo que debe acoplarse. Como "Nombre de la conexión Standby" se tiene que asignar a ambos interlocutores un nombre unívoco en el anillo, con el que se puedan identificar los dos dispositivos correspondientes como interlocutores Standby.

Nota

Para utilizar la función, ha de estar activado HRP.

Ajustes

- **Standby**
  Active o desactive el Standby-Manager.

  Nota

  Si dos dispositivos están acoplados a través de la función Standby, tiene que estar activada la función "Standby" en ambos dispositivos.

  Nota

  El Standby-Manager exige siempre que esté activado un cliente HRP.

- **Nombre de la conexión Standby**
  Introduzca aquí el nombre de la conexión Standby. Con este nombre se define la pareja de dispositivos maestro/esclavo. Ambos deben estar en el mismo anillo. Para esto se da el mismo nombre a dos dispositivos de un anillo.

  Nota

  Asegúrese de que el nombre Standby (para una pareja de dispositivos) se utiliza solo una vez en la red.

- **Asignar el modo de maestro Standby al dispositivo**
  Si está activado este ajuste, el dispositivo se configurará como maestro Standby independientemente de su dirección MAC.
  - Si el ajuste no está seleccionado en ninguno de los dos dispositivos que tienen activada la función Standby, el dispositivo que tiene la dirección MAC más alta asume la función de maestro Standby, siempre que no haya errores.
  - Si este ajuste está activado en ambos dispositivos o si solo uno de los dispositivos soporta el ajuste, el maestro Standby también se selecciona en función de la dirección MAC.

  Este tipo de asignación es importante ante todo en caso de sustitución de dispositivos. Dependiendo de las direcciones MAC, el dispositivo que desempeñaba hasta el momento la función de esclavo puede asumir ahora la función de maestro Standby.

La tabla contiene las siguientes columnas:

- **Puerto**
  Indica el puerto al que se refiere el ajuste.
- **Ajuste**
  Especifique los puertos Standby. Los puertos Standby participan en el desvío del tráfico de datos.
  Si no hay perturbaciones, solo los puertos Standby del maestro están activos y se encargan del tráfico de datos en los anillos (líneas) HRP conectados. Si falla el maestro o la conexión Ethernet (link) de uno de los puertos Standby del maestro, se desconectan todos los puertos Standby del maestro y se activan los puertos Standby del esclavo. De este modo se restablece una conexión Ethernet funcional en los segmentos de red conectados (anillos/líneas HRP).

Spanning Tree

General

Ajustes generales de MSTP

En esta página se configuran los ajustes para el MSTP. Rapid Spanning Tree está activado por defecto y puede pasarse al modo compatible con MSTP, RSTP o STP con un interruptor.

En las páginas de configuración correspondientes de estas funciones pueden efectuarse ajustes detallados.

En función del modo de compatibilidad, puede ajustarse la función correspondiente en la página de configuración respectiva.

Ajustes

- Spanning Tree
  Active o desactive Spanning Tree.
- Compatibilidad de protocolo
  Seleccione el modo de compatibilidad de Spanning Tree. Si, p. ej., se selecciona RSTP, Spanning Tree se comporta como RSTP.
  Existen los siguientes ajustes:
  - STP
  - RSTP
  - MSTP

Uso de Link Aggregation en una instancia MSTP

Para usar Link Aggregation en una instancia MSTP, realice la configuración como se indica a continuación:

1. Cree una Link Aggregation en "Layer 2" > "Link Aggregation (Página 1335)".
2. Cree una instancia MSTP en "Layer 2" > "Spanning Tree" > "MST General (Página 1330)".
3. Configure la Link Aggregation en "Layer 2" > "Spanning Tree" > "Puerto MST (Página 1331)".

Activación automática de MRP en topologías redundantes

Si conecta switches SCALANCE X con estructuras de red redundantes en la vista topológica, en los switches es cuestión se activará automáticamente MRP.

Se desactivará cualquier configuración de otros mecanismos redundantes que pueda existir, como por ejemplo MSTP.

CIST General

En esta página se configura CIST.

Ajustes

- **Prioridad Bridge / Prioridad Root** (solo disponible online)
  Por medio de la prioridad Bridge se especifica qué dispositivo es Root Bridge. El Bridge con mayor prioridad será el Root-Bridge. Cuanto menor sea el valor, mayor será la prioridad. Si en una red hay varios dispositivos con la misma prioridad, pasa a ser Root Bridge el dispositivo cuya dirección MAC tenga el valor numérico más bajo. Ambos parámetros, prioridad Bridge y dirección MAC, forman juntos el identificador Bridge. Dado que el Root-Bridge administra todos los cambios de ruta, debería estar dispuesto lo más central posible, debido al tiempo de ciclo de los telegramas.
- **Dirección Bridge / Dirección Root** (solo disponible online)
  La dirección Bridge indica la dirección MAC del dispositivo y la dirección Root muestra la dirección MAC del Root-Bridge.
- **Root Port** (solo disponible online)
  Indica el puerto a través del cual el switch se comunica con el Root-Bridge.
- **Costes Root** (solo disponible online)
  Los costes de ruta desde este dispositivo hasta el Root-Bridge.
  En modo MSTP se muestran los costes de ruta hasta el Root-Bridge de CIST.
- **Cambios de topología** / **Último cambio de topología** (solo disponible online)
  Para el dispositivo se indica el número de reconfiguraciones debidas al mecanismo Spanning Tree desde el último arranque. Para el Root-Bridge el tiempo desde la última reconfiguración se indica del siguiente modo:
  - Segundos: se agrega sec tras la indicación numérica
  - Minutos: se agrega min tras la indicación numérica
  - Horas: se agrega hr tras la indicación numérica
- **Bridge Hello Time / Root Hello Time** (solo disponible online)
  Cada Bridge envía periódicamente telegramas de configuración (BPDU). El tiempo transcurrido entre dos de estos telegramas es el Hello Time. El valor estándar para ese parámetro es de 2 segundos.
- **Bridge Forward Delay / Root Forward Delay**
  Un bridge no utiliza información de configuración nueva inmediatamente, sino solo después del tiempo establecido en el parámetro. Así se garantiza que el funcionamiento con la nueva topología no se inicie hasta que todos los Bridges tengan las informaciones necesarias.
- **Bridge Max Age / Root Max Age** (solo disponible online)
  Bridge Max Age define la "edad" máxima que puede tener una BPDU recibida para ser aceptada como válida por el switch.
- **Bridge Max Hop Count**
  Este parámetro indica por cuántos dispositivos MSTP puede pasar una BPDU. Si se recibe una MSTP BPDU cuyo "Bridge Max Hop Count" supera el valor aquí configurado, se rechaza.
- **Prioridad Root regional** (solo disponible online)
  Véase la descripción en Prioridad Bridge / Prioridad Root
- **Dirección Root regional** (solo disponible online)
  La dirección MAC del dispositivo.

- **Costes Root regionales** (solo disponible online)
  Muestra los costes de la ruta desde el dispositivo hasta el Root-Bridge regional.
- **Nombre de la región**
  Introduzca el nombre de la región MSTP a la que pertenece este dispositivo. Aquí se indica por defecto la dirección MAC del dispositivo. Este valor debe ser el mismo en todos los dispositivos pertenecientes a la misma región MSTP.
- **Versión de la región**
  Introduzca el número de versión de la región MSTP en la que se encuentra el dispositivo. Este valor debe ser el mismo en todos los dispositivos pertenecientes a la misma región MSTP.
- **Inicializar contadores** (solo disponible online)
  Haga clic en este botón si desea poner a cero los contadores de esta página.

Puerto CIST

Configuración de los puertos MSTP CIST

Al acceder a esta página, se muestra en la tabla el estado actual de la configuración de los parámetros de puerto.

Para llevar a cabo la configuración, haga clic en los campos correspondientes de la tabla de puertos.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Estado MSTP**
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Estado MSTP**
  Defina si el puerto está integrado en el Spanning Tree o no.

Nota

Si desactiva la opción "MSTP Status" para un puerto, pueden formarse bucles. Debe observarse la topología.

- Prioridad
  Introduzca la prioridad del puerto. La prioridad solo se evalúa si los costes de ruta son iguales.
  El valor debe ser divisible entre 16. Si el valor no es divisible entre 16, se adapta automáticamente.
- Calc. Costes
  Introduzca el cálculo de los costes de encaminamiento. Si introduce el valor "0", en el campo "Costes de ruta" se muestra el valor determinado automáticamente.
- Costes de ruta (solo disponible online)
  Este parámetro sirve para calcular la ruta que se ha de elegir. Se selecciona como ruta el trayecto con el valor más bajo. Si varios puertos de un dispositivo tienen el mismo valor con los mismos costes de ruta, se selecciona el puerto con el número más bajo.
  Si el valor indicado en el campo "Calc. costes" es "0", se muestra el valor determinado automáticamente. En cualquier otro caso se indica el valor del campo "Calc. costes". La determinación de los costes de encaminamiento se basa prioritariamente en la velocidad de transmisión. Cuanto mayor es la velocidad de transmisión alcanzable, menor es el valor de los costes de encaminamiento.
  Valores típicos de costes de encaminamiento con Rapid Spanning Tree:
  - 10.000 Mbits/s = 2.000
  - 1000 Mbits/s = 20.000
  - 100 Mbits/s = 200.000
  - 10 Mbits/s = 2.000.000

  Sin embargo, los valores también pueden parametrizarse individualmente.
- Estado (solo disponible online)
  Indica el estado momentáneo en que se encuentra el puerto. Los valores solo se indican y no pueden parametrizarse. El parámetro "Estado" depende del protocolo configurado. Para el estado existen las siguientes posibilidades:
  - Disabled
    El puerto solo recibe y no toma parte en STP, MSTP ni RSTP.
  - Discarding
    En el modo "Discarding" se reciben telegramas BPDU. Se rechazan otros telegramas entrantes o salientes.
  - Listening
    En este estado se reciben y también se envían telegramas BPDU. El puerto está integrado en el algoritmo de Spanning Tree.
  - Learning
    Etapa previa al estado de reenvío; el puerto aprende de forma activa la topología (es decir, las direcciones de los dispositivos).
  - Forwarding
    El puerto está activo en la red tras el tiempo de reconfiguración; recibe y envía telegramas de datos.

- **Fwd. Trans** (solo disponible online)
  Indica el número de cambios del estado "Discarding" al estado "Forwarding".
- **Tipo Edge**
  Defina el tipo del puerto Edge. Existen las siguientes posibilidades:
  - "-"
    El puerto Edge está desactivado. El puerto se considera un "no Edge Port".
  - Admin
    Seleccione esta opción si en este puerto se encuentra siempre un terminal. De lo contrario se activa una reconfiguración de la red con cada cambio de conexión.
  - Auto
    Seleccione esta opción si debe detectarse automáticamente si hay conectado un terminal en este puerto. La primera vez que se establezca una conexión, el puerto se considerará un "no EdgePort".
  - Admin/Auto
    Seleccione estas opciones para utilizar una combinación de ambas en este puerto. La primera vez que se establezca una conexión, el puerto se considerará un "EdgePort".
- **Edge** (solo disponible online)
  Indica el estado del puerto.
  - Enabled
    Hay un terminal en este puerto.
  - Disabled
    En este puerto hay un dispositivo Spanning Tree o Rapid Spanning Tree.

  En un terminal, un switch puede conmutar el puerto con mayor rapidez, sin considerar los telegramas Spanning Tree. Si en contra de este ajuste se recibe un telegrama Spanning Tree, el puerto cambia automáticamente al ajuste "Desactivado" para switches.

- **Tipo P.t.P.**
  Seleccione la opción deseada. La selección depende del puerto ajustado.
  - "-"
    La conexión punto a punto se determina automáticamente. Si el puerto está en semidúplex, no se parte de una conexión punto a punto.
  - P.t.P.
    Se parte de una conexión punto a punto también con semidúplex.
  - Shared Media
    No se parte de una conexión punto a punto incluso si se trata de una conexión dúplex.

  Nota

  La conexión punto a punto significa una conexión directa entre dos dispositivos. Una conexión Shared Media sería, p. ej., una conexión a un hub.

- **Hello Time**
  Introduzca el intervalo tras el cual el Bridge envía BPDUs de configuración.

  Nota

  El ajuste específico de puerto del Hello Time solo es posible con un modo compatible con MSTP.

MST General

Configuración Multiple Spanning Tree

Adicionalmente a RSTP, con MSTP se pueden gestionar varias VLAN en una LAN con árboles RSTP propios.

Ajustes

- **ID de instancia MSTP**
  Introduzca el número de la instancia MSTP.

La tabla se divide en las siguientes columnas:

- **ID de instancia MSTP**
  Muestra el número de la instancia MSTP.
- **Dirección Root**
  Muestra la dirección MAC del Root-Bridge.
- **Prioridad Root**
  Muestra la prioridad del Root Bridge.
- **Prioridad Bridge**
  Introduzca la prioridad Bridge. El valor de la prioridad Bridge es un múltiplo entero de 4096.
- **VLAN ID**
  Introduzca la ID de VLAN. Aquí también puede indicar rangos con ID inicial, "-", ID final. Con "," se separan varios rangos o ID.

Puerto MST

Configuración de los parámetros Multiple Spanning Tree Port

En esta página se ajustan los parámetros de los puertos de las instancias Multiple Spanning Tree configuradas.

Ajustes

- **ID de instancia MSTP**
  Seleccione la ID de la instancia MSTP.

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos.
- **Estado MSTP**
  Seleccione el ajuste. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste para todos los puertos de la tabla 2.

La tabla se divide en las siguientes columnas:

- **Puerto**
  Muestra todos los puertos y Link Aggregations disponibles.
- **ID de instancia MSTP**
  ID de la instancia MSTP.
- **Estado MSTP**
  Active o desactive MSTP para este puerto.
- **Prioridad**
  Introduzca la prioridad del puerto. La prioridad solo se evalúa si los costes de ruta son iguales.
  El valor debe ser divisible entre 16. Si el valor no es divisible entre 16, se adapta automáticamente.
- **Calc. Costes**
  Introduzca el cálculo de costes de encaminamiento en el campo de entrada. Si introduce aquí el valor "0", en el siguiente campo "Costes de ruta" se muestra el valor determinado automáticamente.

- **Costes de ruta** (solo disponible online)
  Este parámetro sirve para calcular la ruta que se ha de elegir. Se selecciona como ruta el trayecto con el valor más bajo. Si varios puertos de un dispositivo tienen el mismo valor con los mismos costes de ruta, se selecciona el puerto con el número más bajo.
  Si el valor indicado en el campo "Calc. costes" es "0", se muestra el valor determinado automáticamente. En cualquier otro caso se indica el valor del campo "Calc. costes". La determinación de los costes de encaminamiento se basa prioritariamente en la velocidad de transmisión. Cuanto mayor es la velocidad de transmisión alcanzable, menor es el valor de los costes de encaminamiento.
  Valores típicos de costes de encaminamiento con Rapid Spanning Tree:
  - 10.000 Mbits/s = 2.000
  - 1000 Mbits/s = 20.000
  - 100 Mbits/s = 200.000
  - 10 Mbits/s = 2.000.000

  Sin embargo, los valores también pueden parametrizarse individualmente.
- **Estado** (solo disponible online)
  Indica el estado momentáneo en que se encuentra el puerto. Los valores solo se indican y no pueden parametrizarse. El parámetro "Estado" depende del protocolo configurado. Para el estado existen las siguientes posibilidades:
  - Disabled
    El puerto solo recibe y no toma parte en STP, MSTP ni RSTP.
  - Discarding
    En el modo "Discarding" se reciben telegramas BPDU. Se rechazan otros telegramas entrantes o salientes.
  - Listening
    En este estado se reciben y también se envían telegramas BPDU. El puerto está integrado en el algoritmo de Spanning Tree.
  - Learning
    Etapa previa al estado de reenvío; el puerto aprende de forma activa la topología (es decir, las direcciones de los dispositivos).
  - Forwarding
    El puerto está activo en la red tras el tiempo de reconfiguración; recibe y envía telegramas de datos.
- **Fwd. Trans** (solo disponible online)
  Indica el número de cambios del estado "Discarding" al estado "Forwarding".

Enhanced Passive Listening Compatibility

Activación de la función

En esta página se activa la compatibilidad avanzada para Passive Listening.

Ajustes

- **Enhanced Passive Listening Compatibility**
  Active o desactive esta función para todo el dispositivo.

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos de la tabla 2.
- **Ajuste**
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Indica el puerto del dispositivo.
- **Ajuste**
  - enabled
    Activa la función para todos los puertos del dispositivo.
  - disabled
    Desactiva la función para todos los puertos del dispositivo.

Loop Detection

Con la función "Loop Detection" se define para qué puertos se debe activar la identificación de bucles. Desde los puertos en cuestión se envían telegramas especiales denominados Loop Detection. Si estos telegramas se devuelven nuevamente al dispositivo significa que hay un bucle ("Loop").

Se habla de un "Local Loop" con la participación de este dispositivo cuando los telegramas se reciben de nuevo en otro puerto del mismo dispositivo. Si los telegramas enviados se reciben nuevamente en el mismo puerto significa que se ha producido un bucle "Remote Loop" en otros componentes de red.

Nota

Un bucle es un error en la estructura de la red que debe eliminarse. La identificación de bucles puede ayudar a encontrar el error más rápido, sin embargo no lo soluciona. La identificación de bucles no es apta para aumentar la disponibilidad de la red mediante el montaje directo de bucles.

Nota

Tenga en cuenta que una identificación de bucles solo es posible en los puertos que no se han configurado como puertos de anillo o Standby.

Ajustes

- Loop Detection
  Active o desactive la detección de bucles.
- VLAN Loop Detection
  Active o desactive la detección de bucles en VLAN.

La tabla 1 se divide en las siguientes columnas:

- 1.ª columna
  Indica que los ajustes son válidos para todos los puertos de la tabla 2.
- Umbral / Remote Reaction / Local Reaction
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- Aplicar a tabla
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- Puerto
  Muestra los puertos disponibles.
- Ajuste
  Determine cómo debe proceder el puerto con telegramas Loop Detection.

  Nota

  Los telegramas de prueba provocan una carga adicional de la red. Recomendamos configurar únicamente switches individuales, p. ej. en los ramales del anillo, como "sender" y los demás como "forwarder".

  – sender
    Se envían y reenvían telegramas Loop Detection.
  – forwarder
    Se reenvían telegramas Loop Detection de otros dispositivos.
  – blocked
    Se bloquea el reenvío de los telegramas Loop Detection.
- Valor umbral
  Especifique tras cuántos telegramas Loop Detection recibidos se supondrá la existencia de un bucle.
- Remote Reaction
  Especifique cómo debe reaccionar el puerto cuando se produce un Remote Loop. Seleccione una de las dos opciones en la lista desplegable:
  – no action
    Un bucle no afecta al puerto.
  – disable
    el puerto se bloquea.

- Local Reaction
  Especifique cómo debe reaccionar el puerto cuando se produce un Local Loop. Seleccione una de las dos opciones en la lista desplegable:
  - no action
    Un bucle no afecta al puerto.
  - disable
    el puerto se bloquea
- Estado (solo disponible online)
  Indica si la detección de bucles está activada o desactivada para el puerto.
- Source Port (solo disponible online)
  Muestra el puerto destinatario del telegrama Loop Detection que ha provocado la última reacción.
- Source VLAN (solo disponible online)
  Muestra la ID de VLAN del telegrama Loop Detection que ha provocado la última reacción. Para ello es imprescindible que previamente se haya activado "VLAN Support Enabled" en la página "Loop Detection Configuration".
- Reiniciar (solo disponible online)
  Después de eliminar un bucle de la red, haga clic en este botón para resetear nuevamente el puerto.

Link Aggregation

Agrupación de enlaces de red para obtener redundancia y una mayor amplitud de banda

La Link Aggregation según IEEE 802.3ad permite agrupar varias conexiones entre dispositivos vecinos, para alcanzar así una mayor amplitud de banda y procurar adicionalmente seguridad contra fallos.

Para esto se agrupan los puertos de ambos dispositivos interlocutores en las denominadas "Link Aggregations" y luego se conectan los dispositivos entre sí a través de estos puertos. Para asignar puertos (es decir, links) correctamente a un interlocutor, se usa el Link Aggregation Control Protocol (LACP) del estándar IEEE 802.3ad.

Pueden definirse hasta 8 Link Aggregations. A cada Link Aggregation pueden asignársele como máximo 8 puertos.

También es posible utilizar una Link Aggregation dentro de una instancia MSTP, véase "Layer 2" > "Spanning Tree" > "General (Página 1325)".

Ajustes

La tabla se divide en las siguientes columnas:

- Puerto
  Muestra el número de puerto virtual de esta Link Aggregation. La designación es asignada internamente por el firmware.
- Nombre de la Link Aggregation
  Introduzca un nombre para la Link Aggregation. El usuario puede indicar este nombre en la configuración. El nombre no es absolutamente necesario pero puede resultar útil para diferenciar las distintas agrupaciones de enlaces.
- Dirección MAC
  Muestra la dirección MAC.
- Estado
  Active o desactive la Link Aggregation.
- MTU
  Defina el tamaño de los paquetes.
- LACP
  - on
    Activa el envío de telegramas LACP.
  - off
    Desactiva el envío de telegramas LACP.
- Frame Distribution
  Defina el tipo de distribución de los paquetes entre los diferentes enlaces de una Aggregation.
  - Destination&Source Mac
    La distribución se basa en una combinación de dirección MAC de origen y de destino.
  - Destination&Source IP-MAC
    La distribución se basa en una combinación de dirección IP y MAC de origen y de destino.

- Modo VLAN
  Defina cómo se introduce la Link Aggregation en una VLAN:
  - Hybrid
    La Link Aggregation envía telegramas con y sin tag. No forma parte automáticamente de una VLAN.
  - Trunk
    La Link Aggregation envía solo telegramas con tag y forma parte automáticamente de todas las VLANs.
- Puerto
  Muestra los puertos pertenecientes a esta Link Aggregation. En la lista desplegable pueden seleccionarse los siguientes valores:
  - "-" (desactivado)
    La Link Aggregation está desactivada.
  - "a" (activo)
    El puerto envía telegramas LACP y solo participa en la Link Aggregation cuando se reciben telegramas LACP.
  - "p" (pasivo)
    El puerto solo participa en la Link Aggregation cuando se reciben telegramas LACP.
  - "o" (on)
    El puerto participa en la Link Aggregation y no envía telegramas LACP.

  Nota

  Dentro de una "Link Aggregation" solo se admiten puertos con la siguiente configuración:
  - todos los puertos con "o"
  - todos los puertos con "a" o "p".

Reenvío DCP

Aplicaciones

El protocolo DCP lo utilizan STEP 7 y PST-Tool para la configuración y el diagnóstico. Con el ajuste de fábrica, DCP está activado en todos los puertos, lo que significa que los telegramas DCP recibidos se transmiten a todos los puertos. Con esta opción tiene la posibilidad de desactivar la emisión de telegramas para puertos concretos, por ejemplo, para aislar sectores de la red de la configuración vía PST-Tool o para dividir el conjunto de la red en subredes más pequeñas para la configuración y el diagnóstico.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Especifique si el puerto debe bloquear o transmitir los telegramas DCP en la salida. Existen las siguientes posibilidades:
  - forward
    En este puerto se reenvían telegramas DCP.
  - block
    En este puerto no se reenvían telegramas DCP a la salida. Sin embargo, la recepción sigue siendo posible a través de este puerto.

LLDP

Aplicaciones

PROFINET utiliza el protocolo LLDP para el diagnóstico de topología. Con el ajuste de fábrica, LLDP está activado para todos los puertos, es decir, se envían y reciben telegramas LLDP en todos los puertos. Con esta función se tiene la posibilidad de activar o desactivar el envío y/o la recepción por cada puerto.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- Puerto
  Muestra el puerto.
- Ajuste
  Defina la funcionalidad LLDP. Existen las siguientes posibilidades:
  - Rx
    Este puerto solo puede recibir telegramas LLDP.
  - Tx
    Este puerto solo puede enviar telegramas LLDP.
  - Rx & Tx
    Este puerto puede recibir y enviar telegramas LLDP.
  - "-" (desactivado)
    Este puerto no puede recibir ni enviar telegramas LLDP.

Unicast

Filtro

Filtrado de direcciones

Esta página muestra el contenido actual de la tabla de filtros Unicast. En esta tabla se recogen las direcciones de origen de telegramas de direcciones Unicast. Las entradas mostradas se han realizado estáticamente, es decir, el usuario las ha parametrizado.

En esta página se definen también los filtros Unicast estáticos.

Ajustes

- ID de VLAN
  Seleccione la ID de VLAN en la que se configura estáticamente una nueva dirección MAC. Si no se especifica nada, está parametrizado "VLAN1" como ajuste básico.
- Dirección MAC
  Introduzca aquí la dirección MAC.

La tabla contiene las siguientes columnas:

- ID de VLAN
  Indica la ID de VLAN asignada a esa dirección MAC.
- Dirección MAC
  Indica la dirección MAC del participante que el dispositivo ha aprendido o que el usuario ha configurado.

- **Estado**
  Muestra el estado de cada entrada de dirección:
  - static
    Configuración por parte del usuario. Las direcciones estáticas están almacenadas de forma permanente, lo que significa que no se borran tras expirar el Aging Time ni al reiniciar el switch.
- **Puerto**
  Indica a través de qué puerto se puede acceder al dispositivo con la dirección especificada. Los telegramas recibidos por el dispositivo cuya dirección de destino coincida con esta dirección se transmiten a ese puerto.

Nota

Para direcciones Unicast solo se puede indicar **un** puerto.

Learning

Inicio/detención del aprendizaje

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

El aprendizaje automático permite introducir automáticamente todos los dispositivo conectados en la tabla de filtros Unicast. Mientras la función "Iniciar aprendizaje" está activada, todas las direcciones Unicast aprendidas se registran de inmediato como entradas Unicast estáticas.
El proceso de aprendizaje finaliza cuando se hace clic en el botón "Detener aprendizaje". De este modo, el aprendizaje puede durar desde unos pocos minutos hasta varias horas en grandes redes, a fin de localizar de modo efectivo todos los dispositivos. Solo pueden encontrarse los dispositivos que envían paquetes durante el proceso de aprendizaje. Activando a continuación la función Port Lock, en los puertos correspondientes solo se admitirán paquetes de los dispositivos conocidos tras la finalización del proceso de aprendizaje (entradas Unicast estáticas).

Nota

Si la función Port Lock ya está activa en puertos concretos antes del aprendizaje automático, en estos puertos no se aprenden direcciones. De este modo es posible aprender solo en determinados puertos. Así pues, active previamente la función Port Lock en los puertos que no deban aprender direcciones.

Ajustes

- **Iniciar aprendizaje**
  Haga clic en el botón "Iniciar aprendizaje" para iniciar el proceso de aprendizaje.
  El dispositivo introducirá las direcciones de dispositivos conectados hasta que el usuario detenga el proceso.
- **Detener aprendizaje**
  Haga clic en el botón "Detener aprendizaje" para detener el proceso de aprendizaje.
  Las entradas aprendidas se guardan.
- **Borrado de todas las entradas Unicast estáticas**
  Haga clic en el botón "Borrar todas las direcciones Unicast estáticas" para borrar todas las entradas estáticas.
  En grandes redes con un gran número de dispositivos, es posible que el aprendizaje automático dé lugar a un exceso de entradas estáticas indeseadas. Para no tener que borrar las entradas estáticas una a una, este botón ofrece la posibilidad de borrarlas todas.
  Esta función está desactivada durante el aprendizaje automático.

Nota

El borrado puede tardar cierto tiempo, en función del número de entradas.

Locked Ports

Activación del control de acceso

En esta página se pueden bloquear los diferentes puertos para dispositivos desconocidos.

Si la función Port Lock está activada, en ese puerto se rechazan inmediatamente los paquetes procedentes de direcciones MAC desconocidas. El puerto admite los paquetes de dispositivos conocidos.
Puesto que los puertos con la función Port Lock activada tampoco aprenden direcciones MAC, las direcciones aprendidas en estos puertos se eliminan automáticamente al activarse la función Port Lock.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Active o desactive la función Port Lock para el puerto.

Blocking

Bloqueo del envío de telegramas Unicast desconocidos

En esta página se bloquea el envío de telegramas Unicast para puertos determinados.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Active o desactive el bloqueo de telegramas Unicast.

Nota

Redundancia de anillo / Standby

Si están activados la redundancia de anillo o Standby, los puertos configurados en este sentido se excluirán del bloqueo.

Multicast

Grupos

Aplicaciones Multicast

En la mayoría de los casos se envía un telegrama con una dirección Unicast a un destinatario determinado. Si una aplicación debe enviar los mismos datos a varios destinatarios, se puede reducir el volumen de datos a transmitir enviando los datos a todos ellos a través de una dirección Multicast. Para algunas aplicaciones existen direcciones Multicast fijas (NTP, IETF1-Audio, IETF1-Video, etc.).

Reducción de la carga de red

En comparación con los telegramas Unicast, los telegramas Multicast suponen una mayor carga para el dispositivo. Esto se debe a que los telegramas Multicast se envían de manera general a todos los puertos. Existen tres posibilidades para reducir la carga causada por telegramas Multicast:

- Entrada estática de las direcciones en la tabla de filtros Multicast.
- Entrada dinámica de las direcciones por escucha de telegramas de parametrización IGMP (configuración IGMP).
- Asignación dinámica de direcciones por telegramas GMRP.

Todos los procedimientos mencionados tienen como consecuencia que los telegramas Multicast se envían únicamente a los puertos para los que se han introducido las correspondientes direcciones.

En "Multicast" se muestran los telegramas Multicast registrados actualmente en la tabla de filtros, con sus puertos de destino. Las entradas mostradas se han realizado estáticamente, es decir, el usuario las ha parametrizado.

Ajustes

- ID de VLAN
  Elija la ID de VLAN que se asignará a la dirección MAC Multicast.
- Dirección MAC
  Introduzca una nueva dirección MAC Multicast para configurar.

La tabla se divide en las siguientes columnas:

- ID de VLAN
  Indica la ID de la VLAN a la que está asignada la dirección MAC Multicast.
- Dirección MAC
  Indica la dirección MAC Multicast aprendida por el dispositivo o configurada por el usuario.
- Estado
  Muestra el estado de cada entrada de dirección. Al respecto son posibles los siguientes datos:
  - static
    La dirección ha sido introducida estáticamente por el usuario. Las direcciones estáticas están almacenadas de forma permanente, lo que significa que no se borran tras expirar el Aging Time ni al rearrancar el dispositivo. Deben ser borradas por el usuario.

IGMP

Función

Los IE Switches soportan "IGMP Snooping" y la función IGMP Querier. Si está activado "IGMP Snooping", los telegramas IGMP se evalúan y con esta información se actualiza la tabla de filtros Multicast. Si además está activado "IGMP Querier", los IE Switches envían también peticiones IGMP que provocan respuestas en los dispositivos compatibles con IGMP.

IGMP Snooping Aging Time

Con este menú puede configurar el Aging Time para la configuración IGMP. Una vez transcurrido este tiempo se borran de la tabla de direcciones las entradas generadas por IGMP, si estas no se actualizan por medio de un nuevo telegrama IGMP.

La especificación es válida para todos los puertos, y no es posible una configuración específica para puertos concretos.

IGMP Snooping Aging Time en función del Querier

SCALANCE XR500 como IGMP Querier

Si se utiliza un SCALANCE XR500 como IGMP Querier, el intervalo Query es de 125 segundos. Ajuste el "IGMP Snooping Aging Time" en 250 segundos como mínimo.

Otros IGMP Querier

Si se utiliza otro IGMP Querier, el valor de "IGMP Snooping Aging Time" debería ser al menos dos veces el intervalo Query.

Ajustes

- IGMP Snooping
  Active o desactive IGMP (Internet Group Management Protocol). Esta función permite asignar direcciones IP a grupos Multicast. Si la opción está activada, las entradas de IGMP se incluyen en la tabla y los telegramas IGMP se transmiten.
- IGMP Snooping Aging Time
  Introduzca en este campo el valor en segundos para el Aging Time.
- IGMP Querier
  Active o desactive "IGMP Querier". El dispositivo envía solicitudes IGMP.

GMRP

Activación de GMRP

Activando las opciones correspondientes se define individualmente para cada puerto si se aplica GMRP. Si "GMRP" está desactivado para un puerto, no se efectúa ningún registro para él y no puede enviar ningún telegrama GMRP.

Ajustes

- GMRP
  Active o desactive la función GMRP.

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplican los ajustes a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles y las Link Aggregations.
- **Ajuste**
- Active o desactive GMRP para el puerto o la Link Aggregation.

Blocking

En esta página se bloquea el envío de telegramas Multicast para puertos determinados.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Active o desactive el bloqueo de telegramas Multicast.

Broadcast

Bloqueo de la transmisión de telegramas Broadcast

En este menú se puede bloquear la transmisión de telegramas Broadcast para puertos determinados.

Nota

Algunos protocolos de comunicación funcionan solo con la ayuda de Broadcast. En esos casos, el bloqueo puede llevar al fallo de la comunicación de datos. Bloquee la función Broadcast solo si está seguro de que puede prescindir de ella.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que el ajuste es válido para todos los puertos de la tabla 2.
- **Ajuste**
  Seleccione el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplica el ajuste a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Se muestran todos los puertos disponibles y las Link Aggregations.
- **Ajuste**
  Active o desactive el bloqueo de telegramas Broadcast.

PTP

General

Nota

PTP solo está disponible en SCALANCE X500.

El Precision Time Protocol (PTP) conforme a IEEE 1588v2 permite sincronizar la hora de los dispositivos que están conectados a los puertos de otro dispositivo. Dichos dispositivos transfieren los telegramas de sincronización por la red utilizando el procedimiento "Transparent Clock" (TC). Se soportan los mecanismos de corrección "End-to-End" y "Peer-to-Peer".

Para utilizar IEEE 1588v2 hay que activar esta función y configurar cada puerto que está dentro de la ruta de sincronización así como los puertos que están bloqueados por

mecanismos de redundancia. IEEE 1588v2 también puede utilizarse con mecanismos de redundancia en el anillo, como HRP, acoplamiento de reserva de anillos, MRP y RSTP.

Ajuste

- Modo de operación 1588
  Son posibles los siguientes ajustes:
  - off
    El dispositivo no procesa mensajes PTP. Sin embargo, los mensajes PTP son transferidos siguiendo las reglas del dispositivo.
  - transparent
    El dispositivo adopta la función de un Transparent Clock y reenvía los mensajes PTP a otros dispositivos, no sin antes haber realizado entradas en el campo de corrección del mensaje PTP.

TC General

1588 Transparent Clock

En esta página se definen los ajustes generales de PTP.

Ajustes

- Mecanismo Delay
  Especifique el mecanismo Delay con el que debe trabajar el dispositivo:
  - End-to-End (se utiliza el mecanismo Delay Request Response)
  - Peer-to-Peer (se utiliza el mecanismo Peer Delay)
- Número de dominio
  Introduzca el número de identificación para el dominio horario. La sincronización solo afecta a los dispositivos que están dentro del dominio. El dispositivo ignora los mensajes PTP que tienen un número de dominio distinto. Un dispositivo SCALANCE solo puede estar asignado a un dominio de sincronización.

Puerto TC

En esta página se definen los puertos que pueden procesar mensajes PTP.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- **1.ª columna**
  Indica que los ajustes son válidos para todos los puertos de la tabla 2.
- **Ajuste / Mecanismo de transporte**
  Seleccione el ajuste. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- **Aplicar a tabla**
  Al hacer clic en este botón se aplican los ajustes a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- **Puerto**
  Muestra los puertos disponibles.
- **Ajuste**
  Active o desactive PTP. Si está activado, el puerto procesa mensajes PTP.
- **Flag de error**
  Muestra el estado de error en relación a PTP.
  - **verdadero**
    Se ha producido un error.
  - **falso**
    No hay errores en este puerto.
- **Mecanismo de transporte**
  Define el protocolo para la transferencia de los mensajes PTP. Este protocolo debe ser soportado por el interlocutor correspondiente del puerto.
  - Ethernet
  - UDP IPv4

RMON

Estadística

Estadística

En esta página se especifican los puertos para los que se visualizarán estadísticas.

Ajustes

- RMON
  Si activa esta casilla de verificación, Remote Monitoring (RMON) permite recopilar datos de diagnóstico en el dispositivo, procesarlos y extraerlos a través de SNMP de una estación de gestión de red que soporte también RMON. Estos datos de diagnóstico, como pueden ser evoluciones de carga referidas a puertos, permiten detectar a tiempo problemas en la red y eliminarlos.
- Puerto
  Seleccione los puertos para los que deban visualizarse estadísticas.

La tabla se divide en las siguientes columnas:

- 1.ª columna
  Activa la casilla de verificación para los puertos cuyas estadísticas no deban visualizarse más.
- Puerto
  Muestra los puertos para los que se visualizarán estadísticas.

Historial

Muestras de las estadísticas

En esta página se especifica si deben guardarse muestras de las estadísticas para un puerto. Es posible definir cuántas entradas deben guardarse y en qué intervalo deben tomarse muestras.

Ajustes

La tabla 1 se divide en las siguientes columnas:

- 1.ª columna
  Indica que los ajustes son válidos para todos los puertos.
- Ajuste
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- Entradas
  Introduzca el número máximo de muestras que se guardan simultáneamente. Si se ha introducido "No Change", la entrada de la tabla 2 permanece invariable.
- Intervalo[s]
  Introduzca el intervalo tras el cual se guarda el estado actual de la estadística como muestra. Si se ha introducido "No Change", la entrada de la tabla 2 permanece invariable.
- Aplicar a tabla
  Al hacer clic en este botón se aplican los ajustes a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- Puerto
  Indica el puerto al que se refieren los ajustes.
- Ajuste
  Active el puerto para el que desea visualizar el historial.
- Entradas
  Introduzca el número máximo de muestras que se guardan simultáneamente.
- Intervalo[s]
  Introduzca el intervalo tras el cual se guarda el estado actual de la estadística como muestra.

Layer 3

Configuración

Esta página contiene la vista general de las funciones de Layer 3 del dispositivo. Para utilizar las funciones "Routing", "VRRP", "OSPF" y "RIP", el dispositivo requiere un KEY PLUG.

Ajustes

- Routing (solo disponible para Layer 3)
  Active o desactive la función "Routing".
- DHCP Relay Agent
  Active o desactive el DHCP Relay Agent. Se pueden configurar otros ajustes en "Layer 3 > DHCP Relay Agent".
- VRRP (solo disponible para Layer 3)
  Active o desactive el routing vía VRRP. Para utilizar VRRP, active primero la función "Routing". Se pueden configurar otros ajustes en "Layer 3 > VRRP".
- OSPF (solo disponible para Layer 3)
  Active o desactive el routing vía OSPF. Para utilizar OSPF, active primero la función "Routing". Se pueden configurar otros ajustes en "Layer 3 > OSPFv2".
- RIP (solo disponible para Layer 3)
  Active o desactive el routing vía RIP. Para utilizar RIP, active primero la función "Routing". Se pueden configurar otros ajustes en "Layer 3 > RIPv2".

Subredes

Sinopsis

Subred

Esta página muestra las subredes para la interfaz seleccionada. Si en una interfaz hay varias subredes disponibles, la primera entrada de la interfaz tendrá el tipo de dirección "Primaria". Todas las demás subredes tienen el tipo de dirección "Secundaria".

En esta página pueden crearse más subredes, "Layer 3 > Subredes > Sinopsis". En "Layer 3 > Subredes > Configuración" se configuran los ajustes de las subredes creadas.

Ajustes

Para subredes del tipo de dirección "Secundaria" son posibles los ajustes siguientes:

- Interfaz
  Seleccione la interfaz en la que debe configurarse otra subred.

La tabla se divide en las siguientes columnas:

- Interfaz
  Muestra la interfaz.
- Interfaz TIA
  Indica si la entrada es una interfaz TIA.
- Nombre de interfaz
  Muestra el nombre de la interfaz.
- Dirección MAC
  Muestra la dirección MAC.
- Dirección IP
  Muestra la dirección IP de la subred.
- Máscara de subred
  Muestra la máscara de subred.
- Tipo de dirección
  Muestra el tipo de dirección. Son posibles los siguientes valores:
  - Primaria
    La primera dirección IP configurada en una interfaz.
  - Secundaria
    Todas las demás direcciones IP configuradas en una interfaz.

- **Método de asignación de direcciones IP**
  Indica cómo se asigna la dirección IP.
  - Estático
    La dirección IP es estática. Los ajustes IP se introducen en "Dirección IP" y "Máscara de subred".
  - Automático (DHCP)
    El dispositivo obtiene una dirección IP dinámica de un servidor DHCP.
- **Estado de la detección de colisiones de direcciones**
  Indica el estado de la función "Detección de colisiones de direcciones".
  - Idle
    La interfaz no está activa y no tiene dirección IP.
  - Starting
    Este estado designa la fase de arranque. En esta fase, el dispositivo envía primero una consulta acerca de si la dirección IP prevista ya existe. Si la dirección todavía no está asignada, el dispositivo envía una notificación indicando que a partir de ahora utilizará esta dirección IP.
  - Conflict
    La interfaz no está activa. La interfaz intenta utilizar una dirección IP que ya está asignada.
  - Defending
    La interfaz utiliza una dirección IP unívoca. Otra interfaz intenta utilizar la misma dirección IP.
  - Active
    La interfaz utiliza una dirección IP unívoca. No hay colisiones.
  - Not supported
    La función para detectar colisiones de direcciones no se soporta.

Configuración

En esta página se configura la subred. La subred se crea en "Layer 3 > Subredes > Sinopsis".

Ajustes

- **Interfaz (nombre)**
  Seleccione la interfaz deseada.
- **Nombre de interfaz**
  Introduzca la designación de la interfaz, p. ej. eth0; P3.
- **Dirección MAC** (solo disponible online)
  Muestra la dirección MAC de la interfaz seleccionada.
- **DHCP**
  Active o desactive el cliente DHCP para esta interfaz.
- **Dirección IP**
  Introduzca la dirección IP de la subred. No se permite el uso múltiple de la dirección IP.

- **Máscara de subred**
  Introduzca la máscara de la subred que se va a crear. Las subredes de distintas interfaces no deben solaparse.
- **Tipo de dirección**
  Muestra el tipo de dirección. Son posibles los siguientes valores:
  – Primaria
    La primera subred de la interfaz.
  – Secundaria
    Todas las subredes restantes de la interfaz.
- **Detección de colisiones de direcciones**
  Active o desactive la función.
  Si se activan direcciones IP nuevas en la red, esta función comprueba si pueden producirse colisiones de direcciones.
  Esta función permite detectar direcciones IP que deben asignarse por duplicado.

  Nota

  La función no realiza ninguna comprobación cíclica.

- **Interfaz TIA**
  Active o desactive el ajuste.
  Para la interfaz TIA rigen las condiciones siguientes:
  – Solo es posible activar como interfaz TIA las interfaces con el tipo de dirección "Primario".
  – Siempre debe haber una interfaz TIA.
  – Solo puede haber una interfaz TIA.
  – Una interfaz TIA es siempre una interfaz VLAN.

Interfaz TIA e interfaz PROFINET

La dirección IP de la interfaz TIA está enlazada con la dirección IP que se encuentra en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Direcciones Ethernet".

Ambas direcciones IP tienen siempre el mismo valor. Si se modifica el valor de una dirección IP, la otra dirección IP cambia en correspondencia.

Interfaz TIA

La interfaz TIA es una interfaz por la que transcurren todas las funciones PROFINET del IE Switch.

Interfaz TIA es el nombre que se da a la dirección IP para PROFINET desde el punto de vista del IE Switch.

Interfaz PROFINET

En la interfaz PROFINET se realiza el ajuste IP del dispositivo PROFINET en TIA.

Interfaz PROFINET es el nombre que se da a la dirección IP para PROFINET desde el punto de vista del TIA Portal.

Rutas

Ruta estática

En esta página se definen rutas estáticas. Las rutas estáticas no se incluyen en las adaptaciones automáticas y deben adaptarse manualmente.

Ajustes

- Red de destino
  Introduzca la dirección de la red de destino.
- Máscara de subred
  Introduzca la máscara de subred correspondiente.
- Pasarela
  Introduzca la dirección IP de la pasarela siguiente.
- Métrica
  Introduzca la métrica de la ruta. La métrica corresponde a la calidad de una conexión, p. ej. velocidad y costes.

La tabla se divide en las siguientes columnas:

- Red de destino
  Indica la dirección de la red de destino.
- Máscara de subred
  Muestra la máscara de subred correspondiente.
- Gateway
  Indica la dirección IP de la siguiente pasarela.
- Interfaz (solo disponible online)
  Muestra la interfaz de la ruta.
- Métrica
  Introduzca la métrica de la ruta. Al crear la ruta se registra automáticamente "No utilizado". La métrica corresponde a la calidad de una conexión, p. ej. velocidad y costes. Si hay varias rutas iguales, se utilizará la que tenga el valor métrico más pequeño.
- Estado (solo disponible online)
  Muestra el estado de la ruta.

Route Maps

General

Route Maps

Los Route Maps permiten controlar el procesamiento posterior de la información de routing. Existe la posibilidad de filtrar la información de routing y determinar si debe seguir procesándose, modificarse o rechazarse.

Los Route Maps trabajan de acuerdo con el principio siguiente:

- La información de routing se compara con los filtros de los Route Maps.
- La comparación continúa hasta que los filtros de un Route Map coinciden con las propiedades de una información.
- Seguidamente, la información se procesa de acuerdo con los ajustes del Route Map:
  - La información de routing se rechaza.
  - Las propiedades de la información de routing se modifican.

Ajustes

- Nombre
  Introduzca aquí el nombre del Route Map.
- Número de secuencia
  Introduzca un número para el Route Map.
  El número determina el orden en el que se procesan los Route Maps.

La tabla se divide en las siguientes columnas:

- Nombre
  Muestra el nombre del Route Map.
- Número de secuencia
  Muestra el número de secuencia del Route Map.
- Acción
  Especifique lo que pasará con la información de routing que coincide con los ajustes del Route Map:
  - permit
    La información de routing se seguirá procesando de acuerdo con los ajustes de la ficha "Ajustes".
  - deny
    La información de routing se rechaza.

Coincidencia entre interfaz y métrica

En esta página se especifica si la información de routing para un Route Map debe filtrarse por interfaces o por métrica.

Ajustes

- Route Map (nombre/n.º sec.)
  Seleccione un Route Map.
  Están disponibles los Route Maps creados.
- Tipo
  Elija el criterio de filtrado:
  - Interfaz
  - Métrica

- Interfaz
  Seleccione una interfaz.
  Este campo solo está activo si en la lista desplegable "Tipo" se ha seleccionado la entrada "Interfaz".
- Métrica
  Introduzca un valor para la métrica.
  Este campo solo está activo si en la lista desplegable "Tipo" se ha seleccionado la entrada "Métrica".

La tabla se divide en las siguientes columnas:

- Tipo
  Muestra el tipo seleccionado:
  – Interfaz
  – Métrica
- Valor
  Muestra la interfaz seleccionada o el valor de la métrica.

Coincidencia del origen

En esta página se especifica si la información de routing para un Route Map debe filtrarse por la dirección IP de origen.

Ajustes

- Route Map (nombre/n.º sec.)
  Seleccione un Route Map.
- Dirección IP
  Introduzca la dirección IP del origen que debe servir de filtro.
- Máscara de subred
  Introduzca la máscara de subred del origen que debe servir de filtro.

La tabla se divide en las siguientes columnas:

- Dirección IP
  Muestra la dirección IP del origen.
- Máscara de subred
  Muestra la máscara de subred del origen.

Coincidencia del destino

En esta página se especifica si la información de routing para un Route Map debe filtrarse por la dirección IP de destino.

Ajustes

- Route Map (nombre/n.º sec.)
  Seleccione un Route Map.
- Dirección IP
  Introduzca la dirección IP del destino que debe servir de filtro.
- Máscara de subred
  Introduzca la máscara de subred del destino que debe servir de filtro.

La tabla se divide en las siguientes columnas:

- Dirección IP
  Muestra la dirección IP del destino.
- Máscara de subred
  Muestra la máscara de subred del destino.

Coincidencia del siguiente salto

En esta página se especifica si para un Route Map debe filtrarse por el router que debe enviar la próxima información de routing.

Ajustes

- Route Map (nombre/n.º sec.)
  Seleccione un Route Map.
- Dirección IP
  Introduzca la dirección IP del router por el que debe enviarse la próxima información de routing.

La tabla se divide en las siguientes columnas:

- Dirección IP
  Muestra la dirección IP del próximo router.

Ajustes

En esta página se especifica si la información de routing para un Route Map debe modificarse.

Si, por ejemplo, se ha filtrado por una métrica concreta, aquí puede cambiarse el valor de dicha métrica. Seguidamente, la información de routing se reenviará con el nuevo valor.

Ajustes

- Route Map (nombre/n.º sec.)
  Seleccione un Route Map.

La tabla se divide en las siguientes columnas:

- Nombre
  Muestra el nombre del Route Map.
- Número de secuencia
  Muestra el número de secuencia del Route Map.

- Interfaz
  Introduzca la nueva interfaz por la que se reenviará la información de routing.
- Métrica
  Introduzca el nuevo valor de la métrica con el que se reenviará la información de routing.
- Dirección IP de siguiente salto
  Introduzca la dirección IP del router por el que debe enviarse la próxima información de routing.

DHCP Relay Agent

General

Si el servidor DHCP se encuentra en otra red, el dispositivo no puede acceder al servidor DHCP. El DHCP Relay Agent ejerce de intermediario entre un servidor DHCP y el dispositivo. Para ello, el DHCP Relay Agent transmite el número de puerto del dispositivo junto con la petición DHCP al servidor DHCP. Si un servidor DHCP no está accesible, el dispositivo puede cambiar a otro servidor DHCP.

Ajustes

- DHCP Relay Agent (Opt. 82)
  Active o desactive el DHCP Relay Agent.
- Dirección IP de servidor
  Introduzca la dirección IP del servidor DHCP.

La tabla se divide en las siguientes columnas:

- Dirección IP de servidor
  Muestra la dirección IP del servidor DHCP.

Opción

Parámetros de DHCP Relay Agent

En esta página se definen los parámetros del servidor DHCP, p. ej. la Circuit ID.
La Circuit ID describe la procedencia de la petición DHCP, p. ej. qué puerto ha recibido la petición DHCP.

Los servidores DHCP se definen en la ficha "General".

Ajustes

Configuración global

- Índice del router de Circuit ID
  Cuando está activado, el índice de router se agrega a la Circuit ID creada.
- ID de VLAN de receptor de Circuit ID
  Cuando está activado, la ID de VLAN se agrega a la Circuit ID creada.

- **Puerto receptor de Circuit ID**
  Cuando está activado, el puerto de receptor se agrega a la Circuit ID creada.

  Nota

  Hay que seleccionar como mínimo una opción.

- **ID remota**
  Muestra el identificador de dispositivo.

Configuración específica de la interfaz

- **Interfaz**
  Seleccione la interfaz deseada.

La tabla se divide en las siguientes columnas:

- **Interfaz**
  Muestra la interfaz seleccionada.
- **Tipo de ID remota**
  Seleccione el tipo de identificador de dispositivo. Existen las siguientes posibilidades:
  - Dirección IP
    Como identificador de dispositivo se utiliza la dirección IP actual de la interfaz y se copia en el campo "ID remota".

    Nota

    Sin actualización automática

    No hay ningún enlace entre el campo "ID remota" y la dirección IP ajustada actualmente.

    Si se cambia la dirección IP, la nueva dirección IP no se introduce automáticamente en el campo "ID remota". Solo cambia el valor del campo "Tipo de ID remota" en "Free Text". Para volver a utilizar la dirección IP actual, seleccione de nuevo "Dirección IP" en el campo "Tipo de ID remota".

  - Dirección MAC
    Como identificador de dispositivo se utiliza la dirección MAC del dispositivo.
  - Free Text
    Si se utiliza "Free Text", se puede introducir el nombre del dispositivo como identificador de dispositivo en el campo "ID remota", por ejemplo.
- **ID remota**
  Este campo solo es editable si en "Tipo de ID remota" se selecciona la entrada "Free Text". La ID remota tiene que se inequívoca.

- Tipo de Circuit ID
  Seleccione el tipo de Circuit ID en la lista desplegable. Existen las siguientes posibilidades:
  - Predefined
    La Circuit ID se genera automáticamente a partir del Router Index, la ID de VLAN o el puerto.
  - Free Number
    Si se utiliza "Free Number" se puede introducir la ID en "Circuit ID".
- Circuit ID
  Introduzca la Circuit ID.
  El campo solo es editable si en "Tipo de Circuit ID" se selecciona la entrada "Free Number".

VRRP

Router

En esta página se crean routers virtuales. Se pueden configurar otros parámetros en "Layer 3 > VRRP > Configuración".

Nota

- VRRP solo está disponible para Layer 3.
- VRRP solo puede utilizarse en combinación con interfaces VLAN. No se admiten puertos router.
- Active "VRRP" para configurar VRRP.

Ajustes

Existen los siguientes ajustes:

- VRRP
  Active o desactive el enrutamiento vía VRRP.
- Reacción en interfaces virtuales a solicitudes Ping
  Si está activado, también responden al Ping las direcciones IP virtuales.
- Interfaz
  Seleccione la interfaz que ejerce de router virtual.
- VRID
  Introduzca la ID del router virtual en el campo de entrada. Esta ID define el grupo de routers que forman un router virtual (VR). En el grupo la ID es igual. No puede utilizarse para otros grupos.

La tabla se divide en las siguientes columnas:

- Interfaz
  Indica la interfaz que ejerce de router virtual.
- VRID
  Muestra la ID del router virtual.

- **Dirección MAC virtual**
  Muestra la dirección MAC virtual del router virtual.
- **Dirección IP primaria**
  Indica la dirección IP prioritaria en esa VLAN. La entrada 0.0.0.0 significa que la dirección "Primary" se utiliza en esa VLAN. En cualquier otro caso, serán válidas todas las direcciones IP que hayan sido configuradas en esa VLAN en "Layer 3 > Subred".
- **Estado del router** (solo disponible online)
  Indica el estado actual del router virtual. Los valores posibles son:
  - Master
    El router es el router maestro y adopta la funcionalidad de routing para todas las direcciones IP asignadas.
  - Backup
    El router es el de reserva. Si el router maestro falla, el router de reserva adopta las funciones del maestro.
  - Initialize
    El Router virtual acaba de ser conectado. En breve, pasará al estado "Master" o "Backup".
- **Dirección IP maestra** (solo disponible online)
  Muestra la dirección IP del router maestro.
- **Prioridad**
  Muestra la prioridad del router virtual.
  El 255 se asigna automáticamente al router maestro actual. Todas las prioridades restantes pueden distribuirse libremente entre los routers VRRP. Cuanto mayor sea la prioridad, antes pasará el router VRRP a "Master".
- **Advert. Interna**
  Muestra el intervalo con el que el router maestro envía paquetes VRRP.
- **Interrumpir**
  Muestra los privilegios de un router a la hora de cambiar de función entre reserva y maestro.
  - sí
    Este router tiene preferencia para cambiar de función.
  - no
    Este router no tiene preferencia para cambiar de función.

Configuración

Introducción

En esta página se configura el router virtual.

Nota

VRRP solo está disponible para Layer 3.

Descripción de los valores mostrados

La página contiene los siguientes campos:

- **Interfaz / VRID**
  Seleccione la ID del router virtual que debe configurarse.
- **Dirección IP primaria**
  Seleccione la dirección IP prioritaria. Si el router va a ser el maestro, utilizará esta dirección IP.

  Nota

  Si en esta VLAN solo configura una subred, no es necesaria ninguna entrada. La entrada es 0.0.0.0.
  Si configura varias subredes en la VLAN y desea utilizar una dirección IP determinada como dirección de origen para paquetes VRRP, seleccione una dirección IP adecuada en la lista desplegable. En cualquier otro caso se utiliza la dirección IP prioritaria.

- **Master**
  Si esta opción está activada, se introducirá la dirección IP prioritaria en "Dirección IP asignada". De este modo, la dirección IP prioritaria del router VRRP se utiliza como dirección IP virtual del router maestro virtual. Los routers de reserva de este grupo tienen que desactivar esta opción y utilizar la dirección IP del router en "Dirección IP asignada".
- **Prioridad**
  Introduzca la prioridad de este router virtual.
  El 255 se asigna siempre al router maestro actual. Todas las prioridades restantes pueden distribuirse libremente entre los routers redundantes. Cuanto mayor sea la prioridad, antes pasará el router a "Master".
- **Intervalo Advertisement**
  Introduzca el intervalo en segundos tras el cual un router maestro vuelve a enviar un paquete VRRP.
- **Interrumpir router de baja prioridad**
  Permita los privilegios para cambiar de función entre "Backup" y "Master" basándose en el proceso de selección.

Sinopsis de direcciones

Sinopsis

Esta página muestra qué direcciones IP vigila el router virtual.

Nota

VRRP solo está disponible para Layer 3.

Ajustes

La tabla se divide en las siguientes columnas:

- Interfaz
  Indica la interfaz que ejerce de router virtual.
- VRID
  Muestra la ID de este router virtual.
- Número de direcciones
  Muestra el número de direcciones IP.
- Dirección IP asignada (1) ... Dirección IP asignada (4)
  Muestra las direcciones IP de router vigiladas por este router virtual. Si un router adopta el papel del maestro, la función Routing de todas estas direcciones IP es asumida también por ese router.

Configuración de direcciones

Nota

VRRP solo está disponible para Layer 3.

Ajustes

Son posibles los siguientes ajustes:

- Interfaz / VRID
  Seleccione el router virtual deseado.
- Dirección IP asignada
  Introduzca la dirección IP que debe vigilar el router virtual.

La tabla se divide en las siguientes columnas:

- Dirección IP asignada
  Muestra las direcciones IP vigiladas por el router virtual.

OSPFv2

Configuración

Introducción

En esta página se configura el Routing vía OSPF.

Nota

OSPF solo está disponible para Layer 3.

Ajustes

- **OSPFv2**
  Active o desactive el enrutamiento vía OSPF.
- **ID de router**
  Introduzca una denominación para una de las interfaces OSPF. La denominación se introduce en formato de dirección IP y no tiene que coincidir obligatoriamente con la propia dirección IP.
- **Compatibilidad OSPFv2 - RFC1583**
  Active esta opción si utiliza routers OSPF antiguos, no compatibles con RFC 2328.
- **Border Router**
  Muestra el estado del router OSPF. Si el sistema local es miembro activo de al menos 2 áreas, se trata de un Area Border Router.
- **AS Border Router**
  Especifique si el router es un AS Border Router. Un AS Border Router ejerce de intermediario entre varios sistemas autónomos, p. ej. si hay una red RIP adicional. Un AS Border Router también se necesita para agregar y distribuir rutas estáticas.
- **Nuevo LSA recibido** (solo disponible online)
  Muestra el número de LSAs que se han recibido.
  Las actualizaciones y los LSAs propios no se cuentan.
- **Nuevo LSA configurado** (solo disponible online)
  Indica el número de LSAs diferentes que han sido enviados por este sistema local.
- **Número máximo de LSAs externos**
  Para limitar las entradas de LSAs externos en la base de datos, introduzca el número máximo de LSAs externos.
- **Intervalo Exit [s]**
  Introduzca el intervalo tras el cual el router OSPF debe intentar de nuevo salir del estado Overflow. 0 significa que el router OSPF solo volverá a intentar salir del estado Overflow en un arranque.
- **Filtro de entrada**
  Seleccione un Route Map que filtre rutas entrantes.

- **Redistribución de rutas**
  Especifica qué rutas conocidas se transmiten vía OSPF.
  Existen los siguientes tipos de ruta:
  - Default
  - Connected
  - Static
  - RIPv2

  Nota

  Los ajustes solo pueden activarse en un AS Border Router. Especialmente la activación de las opciones "Ajuste predeterminado" y "Estático" puede causar problemas si se activan en demasiados puntos de la red, p. ej. Forwarding Loops.

- **Route Map**
  Seleccione un Route Map que filtre las rutas que se transfieren vía OSPF.

Areas

Sinopsis

Una red autónoma (Autonomous System) se puede dividir en áreas más pequeñas (Areas).

En esta página se pueden ver, crear, cambiar o borrar las áreas del router.

Nota

OSPF solo está disponible para Layer 3.

Ajustes

- **Area ID**
  Introduzca la ID del área. La base de datos se sincroniza para todos los routers de un área.
  Formato de entrada: x.x.x.x
  x = 0 ... 255

La tabla contiene las siguientes columnas:

- Area ID
  Indica la ID del área.
- Tipo de área
  Seleccione el tipo de área en la lista desplegable.
  - Normal
  - Stub
  - NSSA
  - Backbone
- Summary
  Especifique si para esta área se generan Summary LSAs.
  - Summary: los Summary LSAs se envían al área.
  - No Summary: los Summary LSAs no se envían al área.
- Métrica
  Muestra los costes de la interfaz OSPF.
- Actualizaciones
  Muestra el número de recálculos de las tablas de routing.
- Contador LSA
  Muestra el número de LSAs en la base de datos.
- Area BR
  Muestra el número de Area Border Router (ABR) accesibles dentro de esta área.
- AS BR
  Muestra el número de Autonomous System Border Router (ASBR) accesibles en esta área.

Area Range

Creación de un nuevo rango de área OSPFv2.

En esta página existe la posibilidad de agrupar redes en una Area ID. Este procedimiento se aplica solo a routers Area Border. Esto hace que un router Area Border propague al exterior únicamente una ruta para cara área de direcciones.

Nota

OSPF solo está disponible para Layer 3.

Ajustes

- **Area ID**
  Seleccione la ID de área. La ID se configura en "Layer 3 > OSPFv2 > Áreas".
- **Dirección de subred**
  Introduzca la dirección de la red que se agrupa.
- **Máscara de subred**
  Introduzca la máscara de subred de la red que se agrupa.

La tabla contiene las siguientes columnas:

- **Area ID**
  Indica la ID del área.
- **Dirección de subred**
  Indica la dirección de la red que se agrupa con otras redes.
- **Máscara de subred**
  Indica la dirección de subred de la red que se agrupa con otras redes.
- **Advertise**
  Active esta opción para publicar la red agrupada.

Interfaces

Sinopsis

En esta página se pueden configurar interfaces OSPF.

Nota

OSPF solo está disponible para Layer 3.

Ajustes

- **Dirección IP**
  Elija la dirección IP de la interfaz OSPF.
- **Area ID**
  Seleccione la ID de área con la que está conectada la interfaz OSPF.

  Nota

  Para tipos de dirección secundarios seleccione la misma Area ID que para el tipo de dirección primario correspondiente.
  Encontrará información para saber si se trata de un tipo de dirección primario o secundario en la columna "Tipo de dirección" de la página "Subredes > Sinopsis".

La tabla contiene las siguientes columnas:

- Dirección IP
  Muestra la dirección IP de la interfaz OSPF.
- Area ID
  Seleccione la Area ID con la que está conectada la interfaz OSPF.
- Estado OSPF
  Especifique si OSPF está activo en la interfaz.
  – Activado: OSPF está activado en la interfaz.
  – Desactivado: OSPF está desactivado en la interfaz.
- Métrica
  Introduzca los costes para la interfaz OSPF.
- Prioridad
  Introduzca la prioridad del router. La prioridad solo es relevante para la selección del router designado. Este parámetro puede ser diferente en routers de una misma subred.
- Trans. Delay
  Introduzca el retardo deseado al enviar una actualización de la conexión.
- Retrans. Delay
  Introduzca el tiempo tras el cual se vuelve a enviar un paquete OSPF si no se ha recibido ninguna confirmación.
- Intervalo Hello
  Introduzca la distancia entre dos paquetes Hello.
- Intervalo Dead
  Introduzca el intervalo tras el cual el router vecino se marcará como "fallido" si durante ese tiempo no se ha recibido ningún paquete Hello de él.

Autenticación de interfaz

Configuración de la autentificación de interfaz

En esta página se define la autentificación de la interfaz.

Nota

OSPF solo está disponible para Layer 3.

Ajustes

- Interfaz OSPF
  Seleccione la interfaz OSPF para la que desea configurar la autentificación.
- Tipo de autenticación
  Seleccione el método de autentificación de la interfaz OSPF. Existen las siguientes posibilidades:
  - Ninguno: sin autentificación
  - Simple: autentificación utilizando una contraseña no codificada
  - MD5: autentificación mediante MD5

Autentificación sencilla

- Contraseña
  Introduzca la contraseña para la "autentificación sencilla" si ha seleccionado este tipo de autenticación.
- Confirmación
  Confirme la contraseña introducida.

Autentificación MD5

- ID de clave de autentificación
  Si ha seleccionado este tipo de autenticación, introduzca la ID con la que debe usarse la contraseña como clave para la autenticación MD5. Dado que la ID de clave se transmite con el protocolo, en todos los routers vecinos tiene que estar guardada la misma clave bajo la misma ID de clave.

La tabla se divide en las siguientes columnas:

- ID de clave de autentificación
  Solo puede editarse si se ajusta el método de autentificación MD5. Solo en él se pueden utilizar varias claves.
- Clave MD5
  Introduzca la clave MD5.
- Confirmar clave MD5
  Confirme la clave introducida.
- ID de clave (solo disponible online)
  Indica si la clave MD5 es la ID de clave más reciente.

Virtual Links

Sinopsis

Por razones relacionadas con el protocolo, cada Area Border Router debe tener acceso al área Backbone. Si un router no está conectado directamente al área Backbone, se establece con él una conexión virtual.

Nota

OSPF solo está disponible para Layer 3.

Nota

Recuerde que al crear una conexión virtual es necesario que estén configurados previamente tanto el área Transit, como el área Backbone.

La conexión virtual tiene que tener la misma configuración en ambos lados.

Nota

- **Los enlaces virtuales no serán efectivos mientras el dispositivo no sea ABR.** (solo disponible online)
  Esta nota se muestra cuando se ha configurado al menos una entrada Virtual Link y el dispositivo no es un Area Border Router.

Ajustes

- **ID de router vecino**
  Introduzca la ID del router vecino situado en el otro lado de la conexión virtual.
- **Transit Area ID**
  Elija la ID del área que une ambos routers.

La tabla contiene las siguientes columnas:

- **Transit Area ID**
  Muestra la ID que une ambos routers.
- **ID de router vecino**
  Muestra la ID del router vecino situado en el otro lado de la conexión virtual.
- **Estado Virt. Link**
  Especifique el estado en el que se encuentra la conexión virtual. Son posibles los siguientes estados:
  - down: la conexión virtual está inactiva.
  - point-to-point: la conexión virtual está activa.
- **Trans. Delay**
  Introduzca el retardo deseado al enviar un paquete Link Update (en segundos).

- Retrans. Delay
  Introduzca el tiempo tras el cual se vuelve a enviar un paquete si no se ha recibido ninguna confirmación (en segundos).
- Intervalo Hello
  Introduzca la distancia entre dos paquetes Hello (en segundos).
- Intervalo Dead
  Introduzca el intervalo tras el cual el router vecino se considera "fallido" si durante ese tiempo no se ha recibido ningún paquete Hello de él (en segundos).

Autentificación de Virtual Link

Configuración del inicio de sesión en la conexión virtual

En esta página se define la autentificación de las conexiones virtuales.

Nota

OSPF solo está disponible para Layer 3.

Ajustes

- Conexión virtual (área/vecino)
  Seleccione la conexión virtual para la que desea configurar la autentificación.
- Tipo de autenticación
  Seleccione el método de autentificación de la interfaz OSPF. Existen las siguientes posibilidades:
  - Ninguno: sin autentificación
  - Simple: autentificación utilizando una contraseña no codificada.
  - MD5: autentificación mediante MD5

Autentificación sencilla

- Contraseña
  Introduzca la contraseña para la "autentificación sencilla".
- Confirmación
  Confirme la contraseña introducida.

Autentificación MD5

- ID de clave de autentificación
  Para la autentificación con MD5, introduzca la ID con la que debe usarse la contraseña como clave. Dado que la ID de clave se transmite con el protocolo, en todos los routers vecinos tiene que estar guardada la misma clave bajo la misma ID de clave.

La tabla se divide en las siguientes columnas:

- **ID de clave de autentificación**
  Solo puede editarse si se ajusta el método de autentificación MD5. Solo en él se pueden utilizar varias claves.
- **Clave MD5**
  Introduzca la clave MD5.
- **Confirmar clave MD5**
  Confirme la clave introducida.
- **ID de clave más reciente**
  Indica si la clave MD5 es la ID de clave más reciente.

RIPv2

Configuración

En esta página se configura el routing vía RIP.

Nota

RIPv2 solo está disponible para Layer 3.

Ajustes

- **RIPv2**
  Active o desactive el routing vía RIPv2 .
- **Filtro de entrada**
  Seleccione un Route Map que filtre rutas entrantes.
- **Redistribución de rutas**
  Especifique qué rutas conocidas se transmiten vía RIPv2.
  Existen los siguientes tipos de ruta:
  - Static Default
  - Connected
  - Static
  - OSPF
- **Route Map**
  Seleccione un Route Map que filtre las rutas que se transfieren vía RIPv2.

Interfaces

Sinopsis

En esta página se pueden configurar interfaces RIPv2.

Nota

RIPv2 solo está disponible para Layer 3.

Ajustes

- **Dirección IP**
  Seleccione la dirección IP de la interfaz RIPv2.

La tabla contiene las siguientes columnas:

- **Dirección IP**
  Muestra la dirección IP de la interfaz RIPv2.
- **Enviar actualizaciones**
  Elija el modo en que deben enviarse las actualizaciones:
  - no send
    No se envían actualizaciones.
  - RIPv1
    Se envían actualizaciones para RIPv1.
  - RIPv1-compat.
    Se envían actualizaciones RIPv2 como Broadcasts según las reglas de RIPv1.
  - RIPv2
    Se envían actualizaciones para RIPv2 como Multicast.
  - RIPv1 demand/RIPv2 demand
    Los paquetes RIP se envían solo como respuesta a demandas explícitas.
- **Recibir actualizaciones**
  Elija el modo en que deben aceptarse las actualizaciones recibidas:
  - no receive
    No se reciben actualizaciones.
  - RIPv1
    Se reciben solo actualizaciones de RIPv1.
  - RIPv2
    Se reciben solo actualizaciones de RIPv2.
  - RIPv1/v2
    Se reciben actualizaciones de RIPv1 y RIPv2.
- **Métrica predeterminada**
  Introduzca los costes para la interfaz RIPv2.

Seguridad

Contraseñas

Nota

Esta página solo está disponible cuando hay conexión online con el dispositivo.

Solo el administrador está autorizado a cambiar las contraseñas del dispositivo para administrador y usuarios.

Las contraseñas de fábrica al entregarse el dispositivo son las siguientes:

- Administrador: admin
- Usuario: user

Nota

Cuando inicie la sesión por primera vez o tras un "Restablecimiento de los ajustes predeterminados guardados y reinicio", se le solicitará que cambie la contraseña.

Nota

Cambiar contraseña en modo de configuración "Trial"

Si se cambia la contraseña en modo de configuración "Trial", dicho cambio también se guardará inmediatamente.

Ajustes

- **Contraseña de administrador actual**
  Introduzca la contraseña de administrador válida.
- **Nombre de usuario**
  Seleccione el usuario deseado. Existen las siguientes posibilidades:
  - Administrador: admin
  - Usuario: user
- **Contraseña nueva**
  Introduzca la contraseña nueva.
- **Confirmar contraseña**
  Confirme la contraseña nueva.

AAA

General

La denominación "AAA" utilizada en el menú significa "authentication, authorization, accounting" y sirve para identificar dispositivos de la red y permitirles preparar los servicios correspondientes y determinar el alcance de uso.

En esta página se configura el inicio de sesión.

Ajustes

- Autentificación de inicio de sesión
  Especifique cómo debe ser el inicio de sesión:
  - Local
    Inicio de sesión con nombre de usuario local y contraseña.
  - RADIUS
    Inicio de sesión vía servidor RADIUS.

Cliente RADIUS

La denominación "AAA" utilizada en el menú significa "authentication, authorization, accounting" y sirve para identificar dispositivos de la red y permitirles preparar los servicios correspondientes y determinar el alcance de uso.

Autentificación mediante un servidor externo.

El concepto de RADIUS se basa en un servidor de autenticación externo. Para un terminal, el acceso a la red es posible una vez que el dispositivo haya verificado los datos de conexión en el servidor de autentificación. Tanto el terminal como el servidor de autentificación deben admitir el protocolo EAP (Extensive Authentication Protocol).

Cada columna de la tabla contiene los datos de acceso para un servidor. Primero se consulta al servidor primario en el orden de búsqueda. Si no puede accederse al servidor primario, se consultan servidores secundarios en el orden introducido.

Si ninguno de los servidores responde, no se efectúa la autentificación. El cliente no obtiene acceso a la red aunque se indique un link en el puerto.

Ajustes

La tabla se divide en las siguientes columnas:

- Dirección IP de servidor
  Introduzca la dirección IP del servidor RADIUS.
- Puerto de servidor
  Introduzca el puerto de entrada del servidor RADIUS. Por defecto está ajustado el puerto de entrada 1812.
- Shared Secret
  Introduzca su identificación de acceso.

- Confirmar Shared Secret
  Confirme la identificación de acceso.
- Max. Retrans
  Introduzca aquí el número máximo de intentos de consulta antes de consultar a otro servidor RADIUS configurado o dar por fallido el inicio de sesión. El valor por defecto es de 3.
- Servidor primario
  Especifique si este servidor es el servidor primario. Puede elegir entre las opciones "sí" y "no".
- Estado
  Active o desactive el servidor RADIUS.

Nota

En esta página puede configurar como máximo dos servidores.

802.1x-Authenticator

La denominación "AAA" utilizada en el menú significa "authentication, authorization, accounting" y sirve para identificar dispositivos de la red y permitirles preparar los servicios correspondientes y determinar el alcance de uso.

Activación de la autentificación para puertos concretos

Seleccionando las correspondientes opciones, es posible especificar individualmente para cada puerto si se activa la protección de acceso a la red según IEEE 802.1x.

Ajustes

- Autenticación MAC
  Active o desactive la autenticación MAC para el dispositivo.
- Guest VLAN
  Active o desactive la función "VLAN invitada" para el dispositivo.

La tabla 1 se divide en las siguientes columnas:

- 1.ª columna
  Indica que los ajustes son válidos para todos los puertos de la tabla 2.
- Control de autenticación 802.1x
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- Reautenticación 802.1x
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- Autenticación MAC
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.

- Adoptar asignación VLAN de RADIUS
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- Aut. MAC: direcciones máximas permitidas
  Indique cuántos terminales pueden estar conectados al puerto simultáneamente. Si se ha introducido "No Change", la entrada de la tabla 2 permanece invariable.
- Guest VLAN
  Elija el ajuste deseado. Si está seleccionado "No Change", la entrada de la tabla 2 permanece invariable.
- ID de VLAN invitada
  Indique la ID de VLAN del puerto. Si se ha introducido "No Change", la entrada de la tabla 2 permanece invariable.
- VLAN invitada: direcciones máximas permitidas
  Indique cuántos terminales se admiten simultáneamente en la VLAN invitada.
- Aplicar a tabla
  Al hacer clic en este botón se aplican los ajustes a todos los puertos de la tabla 2.

La tabla 2 se divide en las siguientes columnas:

- Puerto
  Indica el puerto al que se refiere el ajuste.
  Los puertos para los que no es posible una configuración aparecen atenuados y sus ajustes no pueden modificarse.
- Control de autenticación 802.1x
  Defina la autenticación del puerto:
  - Forzar "No autorizar"
    El tráfico de datos a través del puerto está bloqueado.
  - Forzar "Autorizar"
    El tráfico de datos a través del puerto está permitido sin restricciones.
    Ajuste predeterminado
  - Auto
    Los terminales se autentican en el puerto con el método "802.1x".
    El tráfico de datos a través del puerto se permite o bloquea según el resultado de la autenticación.
- Reautenticación 802.1x
  Active esta opción si debe realizarse una reautenticación cíclica para un terminal ya autenticado.
- Autenticación MAC
  Active esta opción si deben autenticarse terminales con el método "Autenticación MAC".
- Adoptar asignación VLAN de RADIUS
  El servidor RADIUS transmite al IE Switch a qué VLAN pertenece el puerto.
  Active esta opción si debe tomarse en consideración la información del servidor. En este caso, el puerto forma parte de la VLAN correspondiente.
  Si la opción está desactivada se rechaza la información de VLAN.
- Aut. MAC: direcciones máximas permitidas
  Indique cuántos terminales pueden estar conectados al puerto simultáneamente.

- Guest VLAN
  Active esta opción si el terminal debe autorizarse en la VLAN invitada en caso de que falle la autenticación.
- ID de VLAN invitada
  Indique la ID de VLAN del puerto.
- VLAN invitada: direcciones máximas permitidas
  Indique cuántos terminales se admiten simultáneamente en la VLAN invitada.
- Aut. MAC: direcciones permitidas actualmente
  Muestra el número de terminales conectados en ese momento.
- Aut. MAC: direcciones bloqueadas actualmente
  Muestra el número de terminales bloqueados en ese momento.
- VLAN invitada: direcciones permitidas actualmente
  Indica cuántos terminales se admiten actualmente en la "VLAN invitada".

Port ACL MAC

Configuración de reglas

En esta página se definen las reglas ACL para el ACL basado en MAC.

Ajustes

La tabla se divide en las siguientes columnas:

- Número de regla
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.
- Dirección MAC de origen
  Introduzca la dirección MAC Unicast del origen.
  Dirección MAC de destino
  Introduzca la dirección MAC Unicast del destino.
- Acción
  Elija el comportamiento. Existen las siguientes posibilidades:
  - Forward
    Si el telegrama cumple la regla ACL, se reenvía.
  - Discard
    Si el telegrama cumple la regla ACL, no se reenvía.
- Puertos Ingress
  Muestra una lista de todos los puertos Ingress para los que rige la regla.
- Puertos Egress
  Muestra una lista de todos los puertos Egress para los que rige la regla.

Nota

Si se introduce la dirección "00:00:00:00:00:00" para dirección MAC de origen y/o destino, la regla creada es válida para todas las direcciones MAC de origen y destino.

Reglas Port Ingress

En esta página se define la regla ACL que debe regir el tratamiento que el puerto dará a los telegramas entrantes.

Ajustes

- **Puertos**
  Seleccione el puerto deseado.
- **Agregar regla**
  Elija la regla ACL que se asignará al puerto. La regla ACL se define en la página "Configuración de reglas".
- **Agregar**
  Para asignar la regla ACL de forma fija al puerto, haga clic en el botón "Agregar". La configuración se muestra en la tabla.
- **Eliminar regla**
  Elija la regla ACL que debe borrarse.
- **Eliminar**
  Para eliminar la regla ACL del puerto, haga clic en el botón "Eliminar".

La tabla se divide en las siguientes columnas:

- **Orden de las reglas**
  Muestra el orden de las reglas ACL.
- **Número de regla**
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.
- **Dirección MAC de origen**
  Muestra la dirección MAC Unicast del origen.
- **Dirección MAC de destino**
  Muestra la dirección MAC Unicast del destino.
- **Acción**
  Elija el comportamiento. Existen las siguientes posibilidades:
  - Forward
    Si el telegrama cumple la regla ACL, se reenvía.
  - Discard
    Si el telegrama cumple la regla ACL, no se reenvía.

Reglas Port Egress

En esta página se definen las reglas ACL que deben regir el tratamiento que el puerto dará a los telegramas salientes.

Descripción

- Puertos
  Seleccione el puerto deseado.
- Agregar regla
  Elija la regla ACL que se asignará al puerto. La regla ACL se define en la página "Configuración de reglas".
- Agregar
  Para asignar la regla ACL de forma fija al puerto, haga clic en el botón "Agregar". La configuración se muestra en la tabla.
- Eliminar regla
  Elija la regla ACL que debe borrarse.
- Eliminar
  Para eliminar la regla ACL del puerto, haga clic en el botón "Eliminar".

La tabla se divide en las siguientes columnas:

- Orden de las reglas
  Muestra el orden de las reglas ACL.
- Número de regla
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.
- Dirección MAC de origen
  Muestra la dirección MAC Unicast del origen.
- Dirección MAC de destino
  Muestra la dirección MAC Unicast del destino.
- Acción
  Elija el comportamiento. Existen las siguientes posibilidades:
  - Forward
    Si el telegrama cumple la regla ACL, se reenvía.
  - Discard
    Si el telegrama cumple la regla ACL, no se reenvía.

Port ACL IP

Configuración de reglas

En esta página se definen las reglas para el ACL basado en IP.

Ajustes

La tabla se divide en las siguientes columnas:

- Número de regla
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.
- Dirección IP de origen
  Introduzca la dirección IP del origen.
- Máscara de subred de origen
  Introduzca la máscara de subred en la que se encuentra el origen.
- Dirección IP de destino
  Introduzca la dirección IP del destino.
- Máscara de subred de destino
  Introduzca la máscara de subred en la que se encuentra el destino.
- Acción
  Seleccione la respuesta en la lista desplegable. Existen las siguientes posibilidades:
  - Forward
    Si el telegrama cumple la regla ACL, se reenvía.
  - Discard
    Si el telegrama cumple la regla ACL, no se reenvía.
- Puertos Ingress
  Muestra una lista de todos los puertos Ingress para los que rige la regla.
- Puertos Egress
  Muestra una lista de todos los puertos Egress para los que rige la regla.

Configuración de protocolo

En esta página se definen las reglas para protocolos.

Ajustes

La tabla se divide en las siguientes columnas:

- Número de regla
  Muestra el número de la regla de protocolo. Al crear una regla se inserta una fila nueva con un número unívoco.
- Protocolo
  Seleccione el protocolo para el que rige la regla.
- Número de protocolo
  Introduzca un número de protocolo para definir protocolos adicionales.
  Este campo solo se puede editar si se ha ajustado "Other Protocol" en Protocolo.
- Puerto de origen mín.
  Indique el número de puerto más bajo posible para el puerto de origen.
  Este campo solo se puede editar si se ha ajustado "TCP" o "UDP" en Protocolo.

- Puerto de origen máx.
  Indique el número de puerto más alto posible para el puerto de origen.
  Este campo solo se puede editar si se ha ajustado "TCP" o "UDP" en Protocolo.
- Puerto de destino mín.
  Indique el número de puerto más bajo posible para el puerto de destino.
  Este campo solo se puede editar si se ha ajustado "TCP" o "UDP" en Protocolo.
- Puerto de destino máx.
  Indique el número de puerto más alto posible para el puerto de destino.
  Este campo solo se puede editar si se ha ajustado "TCP" o "UDP" en Protocolo.
- Tipo de mensaje
  Introduzca un tipo de mensaje para determinar el formato del mensaje.
  Este campo solo se puede editar si se ha ajustado "ICMP" en Protocolo.
- Código de mensaje
  Introduzca un código de mensaje para especificar la función del mensaje.
  Este campo solo se puede editar si se ha ajustado "ICMP" en Protocolo.
- DSCP
  Introduzca un valor para clasificar la prioridad.
  Este campo no se puede editar si se ha ajustado "ICMP" en Protocolo.

Reglas Port Ingress

En esta página se define la regla ACL que debe regir el tratamiento que el puerto dará a los telegramas entrantes.

Ajustes

- Puertos
  Seleccione el puerto deseado.
- Agregar regla
  Elija la regla ACL que se asignará al puerto. La regla ACL se define en la página "Configuración de reglas".
- Agregar
  Para asignar la regla ACL de forma fija al puerto, haga clic en el botón "Agregar". La configuración se muestra en la tabla.
- Eliminar regla
  Elija la regla ACL que debe borrarse.
- Eliminar
  Para eliminar la regla ACL del puerto, haga clic en el botón "Eliminar".

La tabla se divide en las siguientes columnas:

- Orden de las reglas
  Muestra el orden de las reglas ACL.
- Número de regla
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.

- Protocolo
  Muestra el protocolo para el que rige la regla.
- Número de protocolo
  Muestra el número de protocolo.
- Dirección IP de origen
  Muestra la dirección IP del origen.
- Máscara de subred de origen
  Muestra la máscara de subred en la que se encuentra el origen.
- Dirección IP de destino
  Introduzca la dirección IP del destino.
- Máscara de subred de destino
  Introduzca la máscara de subred en la que se encuentra el destino.
- Acción
  Elija el comportamiento. Existen las siguientes posibilidades:
  - Forward
    Si el telegrama cumple la regla ACL, se reenvía.
  - Discard
    Si el telegrama cumple la regla ACL, no se reenvía.
- Puerto de origen mín.
  Muestra el número de puerto más bajo posible para el puerto de origen.
- Puerto de origen máx.
  Muestra el número de puerto más alto posible para el puerto de origen.
- Puerto de destino mín.
  Muestra el número de puerto más bajo posible para el puerto de destino.
- Puerto de destino máx.
  Muestra el número de puerto más alto posible para el puerto de destino.
- Tipo de mensaje
  Muestra el tipo de mensaje.
- Código de mensaje
  Muestra el código de mensaje.
- DSCP
  Muestra el valor para clasificar la prioridad.

Reglas Port Egress

En esta página se definen las reglas ACL que deben regir el tratamiento que el puerto dará a los telegramas salientes.

Ajustes

- **Puertos**
  Seleccione el puerto deseado.
- **Agregar regla**
  Elija la regla ACL que se asignará al puerto. La regla ACL se define en la página "Configuración de reglas".
- **Agregar**
  Para asignar la regla ACL de forma fija al puerto, haga clic en el botón "Agregar". La configuración se muestra en la tabla.
- **Eliminar regla**
  Elija la regla ACL que debe borrarse.
- **Eliminar**
  Para eliminar la regla ACL del puerto, haga clic en el botón "Eliminar".

La tabla se divide en las siguientes columnas:

- **Orden de las reglas**
  Muestra el orden de las reglas ACL.
- **Número de regla**
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.
- **Protocolo**
  Muestra el protocolo para el que rige la regla.
- **Número de protocolo**
  Muestra el número de protocolo.
- **Dirección IP de origen**
  Muestra la dirección IP del origen.
- **Máscara de subred de origen**
  Muestra la máscara de subred en la que se encuentra el origen.
- **Dirección IP de destino**
  Introduzca la dirección IP del destino.
- **Máscara de subred de destino**
  Introduzca la máscara de subred en la que se encuentra el destino.
- **Acción**
  Seleccione la respuesta en la lista desplegable. Existen las siguientes posibilidades:
  - Forward
    Si el telegrama cumple la regla ACL, se reenvía.
  - Discard
    Si el telegrama cumple la regla ACL, no se reenvía.
- **Puerto de origen mín.**
  Muestra el número de puerto más bajo posible para el puerto de origen.
- **Puerto de origen máx.**
  Muestra el número de puerto más alto posible para el puerto de origen.

- Puerto de destino mín.
  Muestra el número de puerto más bajo posible para el puerto de destino.
- Puerto de destino máx.
  Muestra el número de puerto más alto posible para el puerto de destino.
- Tipo de mensaje
  Muestra el tipo de mensaje.
- Código de mensaje
  Muestra el código de mensaje.
- DSCP
  Muestra el valor para clasificar la prioridad.

Management ACL

En esta página es posible aumentar la seguridad del dispositivo. Para especificar qué estaciones pueden acceder al dispositivo y con qué dirección IP, configure la dirección IP o todo un rango de direcciones.

Puede ajustar con qué protocolos y a través de que puertos la estación puede acceder al dispositivo. Se define en qué VLAN debe encontrarse la estación. De este modo se garantiza que solo tengan acceso al dispositivo determinadas estaciones dentro de una VLAN.

Nota

Tenga en cuenta que una configuración incorrecta puede imposibilitar posteriormente el acceso al dispositivo.

Ajustes

- Administración ACL
  Active o desactive la función.
- Dirección IP
  Introduzca la dirección IP o la dirección de red a la que debe aplicarse la regla. Si utiliza la dirección IP 0.0.0.0, los ajustes serán válidos para todas las direcciones IP.
- Máscara de subred
  Introduzca la máscara de subred. La máscara de subred 255.255.255.255 pertenece a una dirección IP determinada. Si desea permitir una subred, introduzca, por ejemplo, 255.255.255.0 para una subred C. La máscara de subred 0.0.0.0 sirve para todas las subredes.

La tabla se divide en las siguientes columnas:

- Orden
  Indica el número de regla ACL. Al crear la regla se inserta una fila nueva con un número unívoco.
- Dirección IP
  Muestra la dirección IP.
- Máscara de subred
  Muestra la máscara de subred.

- VLANs admisibles
  Introduzca el número de las VLANs desde las que se permite el acceso.
  Solo pueden acceder al dispositivo las estaciones que están en dichas VLANs. Si este campo de entrada se deja vacío, no hay restricciones.
  Es posible introducir varias VLANs individuales así como rangos de VLANs separadas por comas, p. ej. 1,5,10-12.
- SNMP
  Especifique si la estación o dirección IP puede acceder al dispositivo mediante el protocolo SNMP.
- TELNET
  Especifique si la estación o dirección IP puede acceder al dispositivo mediante el protocolo TELNET.
- HTTP
  Especifique si la estación o dirección IP puede acceder al dispositivo mediante el protocolo HTTP.
- HTTPS
  Especifique si la estación o dirección IP puede acceder al dispositivo mediante el protocolo HTTPS.
- SSH
  Especifique si la estación o dirección IP puede acceder al dispositivo mediante el protocolo SSH.
- Px.y
  Especifique si la estación o dirección IP puede acceder al dispositivo mediante este puerto.

8.1.4.10 Configuración de PROFIBUS DP

Configurar la periferia descentralizada

En el capítulo siguiente se explica cómo configurar una periferia descentralizada. Además, se presenta información básica sobre la configuración de un sistema maestro DP.

Información básica

- Principios básicos sobre la configuración de un sistema maestro DP (Página 1387)
- Esclavos DP en el catálogo de hardware (Página 1389)
- Estructura de una configuración con esclavos DP sencillos (sistema monomaestro) (Página 1390)
- Estructura de una configuración con esclavos inteligentes (I-Slaves) (Página 1391)
- Estructura de una configuración con dos sistemas maestros (Página 1392)

Resumen de configuración de una periferia descentralizada

Paso	Descripción
1	Cree un sistema maestro DP (Página 1394).
2	En caso necesario, configure las interfaces del sistema maestro DP (Página 1397).
3	Inserte un esclavo DP y configúrelo (Página 1398).
4	En caso necesario, asigne esclavos DP a un grupo SYNC/FREEZE (Página 1402).

Consulte también

Configurar dispositivos DPV1 (Página 1434)

Configurar la comunicación directa (Página 1424)

Principios básicos sobre la configuración de un sistema maestro DP

Periferia descentralizada

Se entienden por periferia descentralizada los sistemas maestros DP que constan de maestro DP y esclavos DP, están conectados con un bus y se comunican entre sí mediante el protocolo PROFIBUS DP.

Configuración de una periferia descentralizada

Puesto que el maestro DP y los esclavos DP pueden ser dispositivos distintos, aquí se explican únicamente los procedimientos básicos de configuración. No obstante, la configuración de la periferia descentralizada es casi idéntica al procedimiento de configuración de una estructura centralizada.

Crear un sistema maestro DP en la vista de redes

Después de colocar un maestro DP y un esclavo DP (p. ej., una CPU 315-2 DP y un IM 153-1) del catálogo de hardware en la vista de redes por medio de Drag & Drop, conecte ambos dispositivos con una subred PROFIBUS.

Puesto que en la vista de redes sólo se pueden configurar los dispositivos pero no los módulos, para la configuración del maestro DP y esclavo DP es necesario cambiar a la vista de dispositivos.

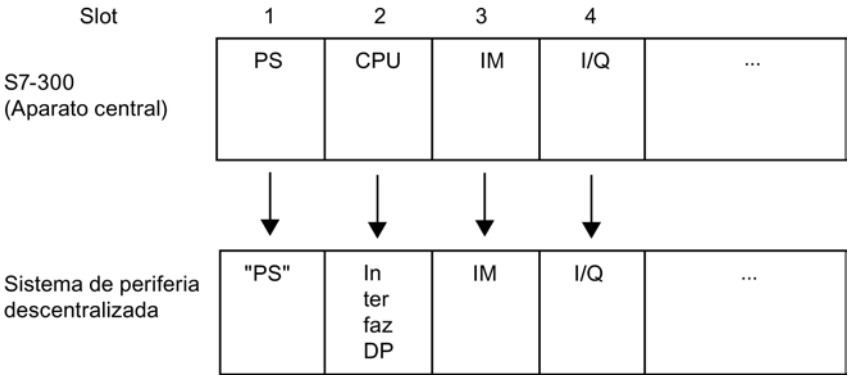
Configurar un esclavo DP

Cambie a la vista de dispositivos para configurar el esclavo DP. Asigne módulos al rack del esclavo DP en la vista de dispositivos y parametrice identificaciones DP y direcciones E/S. Los ajustes se pueden comprobar y editar en la vista de dispositivos y en la ventana de inspección.

Numeración de slots en unidades periféricas descentralizadas

Dependiendo del tipo de esclavo DP utilizado, los slots del esclavo DP comienzan por "0" o por "4".

La numeración de slots de los esclavos DP, como p. ej. con ET 200M, se deriva de la estructura de un S7-300:



Normas sobre slots para esclavos DP según S7-300

Para esclavos DP estructurados según el esquema de los S7-300, se aplican las siguientes reglas sobre slots:

- La periferia en sí (entradas/salidas) comienza siempre por el slot 4.
- El slot 1 está siempre reservado para un módulo de fuente de alimentación (PS), independientemente de si se inserta o no una PS en la instalación real.
- El slot 2 se reserva siempre para la interfaz DP.
- El slot 3 se reserva siempre para una interfaz de ampliación (IM), independientemente de que una unidad periférica real sea ampliable o no.

Este esquema se aplica a todos los tipos de esclavos DP modulares y compactos que se estructuran según el esquema del S7-300. Es importante la asignación de slots para evaluar los avisos de diagnóstico ("slot iniciador de diagnóstico").

Información relacionada

Lea la información complementaria sobre funcionalidad, procedimientos de acceso, etc. en los manuales de los correspondientes dispositivos o en la Ayuda en pantalla de las FCs especiales (p. ej. DP-SEND y DP-RECEIVE para CP 342-5).

Esclavos DP en el catálogo de hardware

Esclavos DP en el catálogo de hardware

Los esclavos DP se encuentran en la carpeta "Periferia descentralizada" del catálogo de hardware. Allí se encuentran esclavos DP compactos y modulares:

- Esclavos DP compactos
  Módulos con entradas y salidas digitales/analógicas integradas, p. ej. ET 200L
- Esclavos DP modulares
  Módulos de interfaz con módulos S7 asignados, p. ej. ET 200M

Los esclavos DP utilizables dependen del maestro DP existente y de la funcionalidad deseada.

Esclavos I en el catálogo de hardware

Los dispositivos que se pueden parametrizar como esclavos DP inteligentes se encuentran en la carpeta "PLC" del catálogo de hardware. los siguientes dispositivos pueden configurarse como esclavo I, por ejemplo:

- CP 342-5 DP
  en el catálogo de hardware en "PLC > SIMATIC S7-300 > CP > PROFIBUS".
- CPU 315-2 DP, CPU 317-2 DP, CPU 319-3 PN/DP
  en el catálogo de hardware en"PLC > SIMATIC S7-300 > CPU".
- IM 151-7 CPU
  en el catálogo de hardware en "PLC > SIMATIC ET200 PLC > ET 200S > CPU".

Consulte también

Examinar el catálogo de hardware (Página 512)

Acopladores DP/DP en el catálogo de hardware

Introducción

Un acoplador DP/DP sirve para conectar dos redes PROFIBUS DP como pasarela (gateway) y transferir así datos del maestro DP de una red al maestro DP de la otra red.

El tamaño máximo de los datos transferibles es de 244 bytes para datos de entrada y 244 bytes para datos de salida.

Acopladores DP/DP en el catálogo de hardware

Los acopladores DP/DP como gateway entre dos sistemas maestros DP se encuentran en la carpeta "Otros dispositivos de campo > PROFIBUS DP > Gateways" del catálogo de hardware.

Configuración del acoplador DP/DP

Los acopladores DP/DP se configuran en ambas redes PROFIBUS, en sistemas maestros propios.

Las áreas de entradas y salidas de ambas redes deben estar sintonizadas unas con otras. Los datos de salida de una parte del acoplador DP/DP se aceptan como datos de entrada de la otra parte y viceversa.

Consulte también

Instalar archivo GSD (Página 1431)

Crear un sistema maestro DP (Página 1394)

Configuraciones con PROFIBUS DP

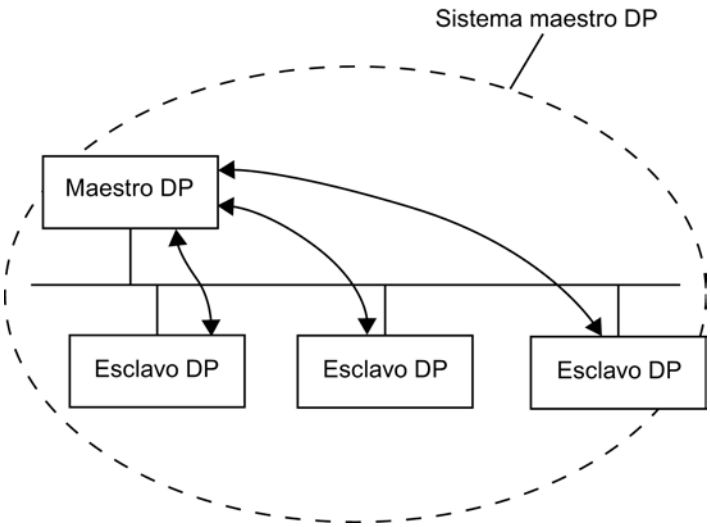
Configuraciones con esclavos DP "sencillos"

Intercambio de datos entre maestro DP y esclavo DP

En la configuración con esclavos DP sencillos, el intercambio de datos se efectúa entre maestro DP y esclavos DP sencillos, es decir, con módulos E/S a través del maestro DP. El maestro DP sondea sucesivamente cada esclavo DP configurado en su lista de llamada (Lista Polling) dentro del sistema maestro DP y transfiere los datos de salida o recibe sus valores de entrada devueltos.

Sistema monomaestro

La configuración con un solo maestro DP se denomina sistema monomaestro. Un único maestro DP está conectado a una subred PROFIBUS DP física con los correspondientes esclavos DP.



Configuraciones con varios sistemas maestros DP

Tipos de configuración con dos o más sistemas maestros DP

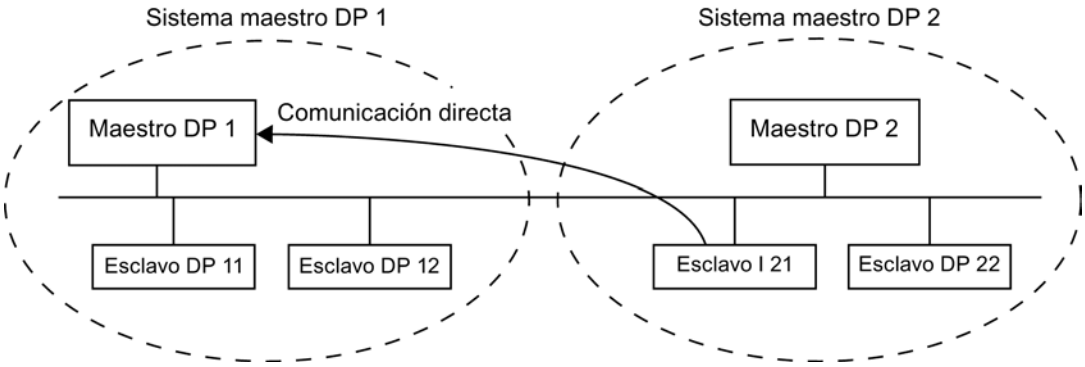
A un conjunto de sistemas maestros DP en una subred física PROFIBUS DP se le denomina también sistema multimaestro.

PROFIBUS DP permite conectar estos sistemas maestros DP de maneras distintas:

- Sin conexión lógica de los sistemas maestros DP entre sí
- Comunicación directa esclavo DP > maestro DP
- Comunicación directa esclavo DP > esclavo I

Comunicación directa esclavo DP > maestro DP

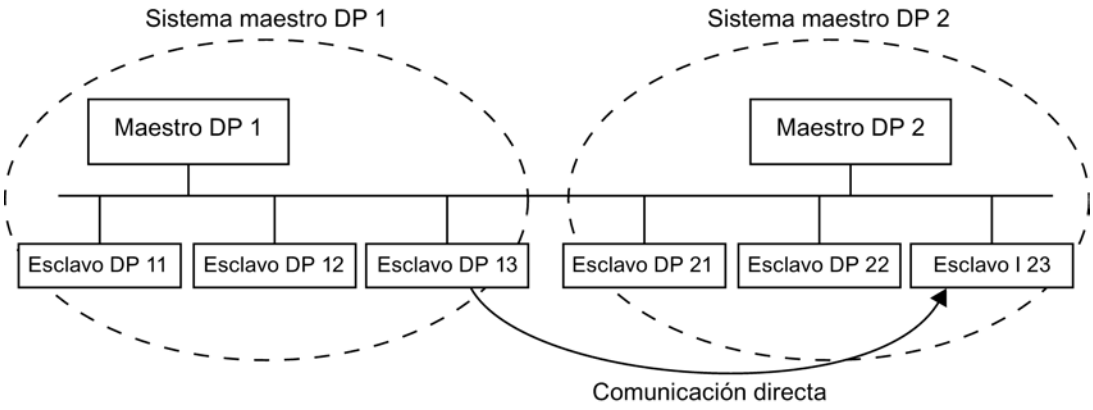
En la configuración con sistema multimaestro, los datos de entrada de los esclavos DP son leídos por los maestros DP en la misma subred física PROFIBUS DP.



Comunicación directa esclavo DP > esclavo I

Un esclavo DP inteligente (esclavo I), como, p. ej., una CPU 315-2 DP, en el sistema multimaestro también puede hacer transferir datos de entrada de esclavos DP, incluso desde distintos sistemas maestros DP, directamente a su área de datos de entrada.

En principio, los esclavos DP a partir de una determinada versión ponen a disposición datos de entrada seleccionados para la comunicación directa. Estos datos de entrada pueden ser utilizados a su vez por esclavos DP inteligentes.



Consulte también

Información importante sobre la comunicación directa (Página 1423)

Configurar la comunicación directa (Página 1424)

Configuraciones con esclavos DP inteligentes

Definición

Los esclavos DP con programa propio para preprocesamiento se denominan esclavos DP inteligentes (esclavos I). Ejemplos de esclavos DP inteligentes:

- CPU 315-2 DP
- CPU 317-2 DP
- CPU 319-3 PN/DP

Configuraciones con esclavos DP inteligentes

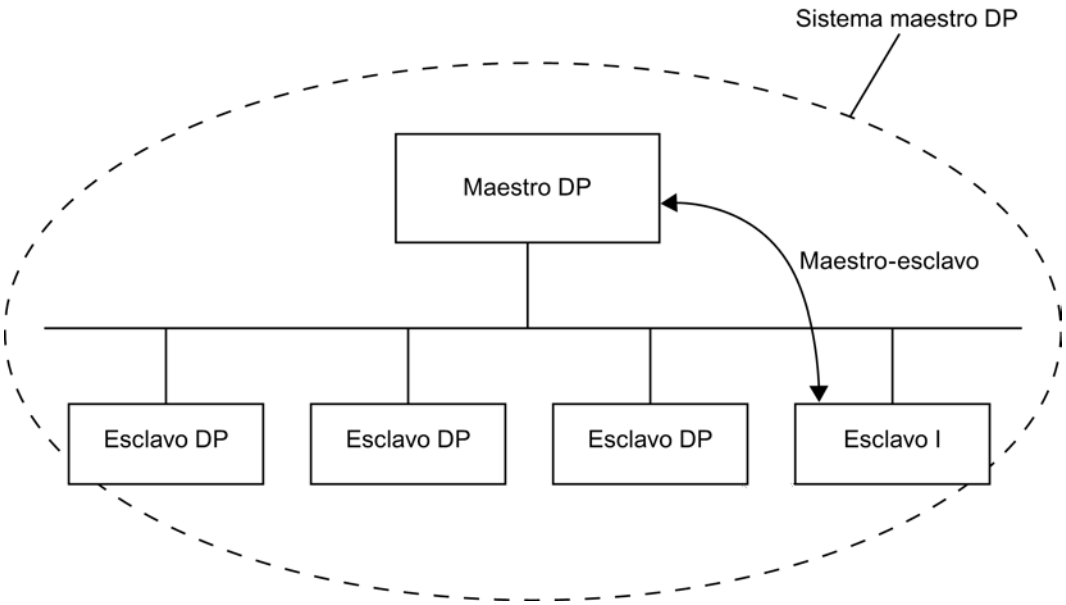
PROFIBUS DP permite configurar esclavos DP inteligentes de dos modos distintos:

- Intercambio de datos esclavo I <> maestro DP
- Comunicación directa esclavo DP > esclavo I

Intercambio de datos esclavo I <> maestro DP

Un sistema de automatización de nivel superior procesa la tarea de automatización dividida en tareas parciales. Las tareas de control que se han de realizar se procesan de manera independiente y eficiente como programas del preprocesamiento en una CPU. Esta CPU puede adoptar la forma de un esclavo DP inteligente.

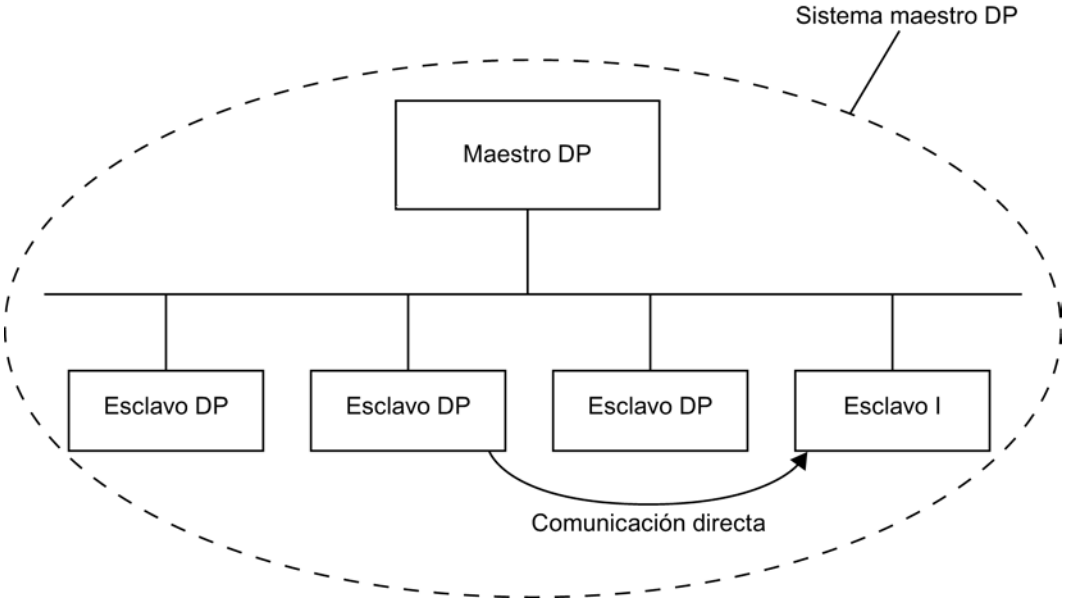
En configuraciones con esclavos DP inteligentes, como p. ej. una CPU 315-2 DP, el maestro DP no accede a los módulos E/S del esclavo DP inteligente, sino sólo al área de operandos de la CPU del esclavo I. El área de operandos no se puede ocupar para módulos E/S reales en el esclavo I. Hay que realizar esta asignación cuando se configura el esclavo I.



Comunicación directa esclavo DP > esclavo I

En esta configuración, los datos de entrada de esclavos DP sencillos se pueden transferir muy rápidamente a esclavos DP inteligentes en la subred PROFIBUS DP.

Así pues, en principio todos los esclavos DP sencillos (a partir de una determinada versión), y también los esclavos DP inteligentes, ponen a disposición sus datos de entrada para la comunicación directa. Se pueden utilizar esclavos DP inteligentes como receptores de estos datos.



Consulte también

Insertar esclavo I en un sistema maestro DP (Página 1415)

Ejemplo de configuración de un esclavo DP como esclavo I (Página 1422)

Información importante sobre la comunicación directa (Página 1423)

Configurar sistemas de periferia descentralizada

Crear un sistema maestro DP

Introducción

Para crear un sistema maestro DP se necesita un maestro DP y al menos un esclavo DP. En cuando se conecta un maestro DP con un esclavo DP, se produce un acoplamiento maestro-esclavo.

Maestro DP

Como maestro DP se pueden utilizar los dispositivos siguientes:

- CPU con interfaz de maestro DP enchufable o integrada de forma fija
- CP en conexión con una CPU
- Módulo de interfaz asignado a una CPU/FM.
- Módulo de interfaz con interfaz de maestro DP

El maestro DP debe tener una interfaz DP. Con algunos dispositivos, en la sección "Modo de operación" de la ventana de inspección de la interfaz DP se puede elegir entre los posibles modos de operación (como "Maestro" y "Esclavo"). Ajuste el modo de operación necesario en la lista desplegable.

Requisitos

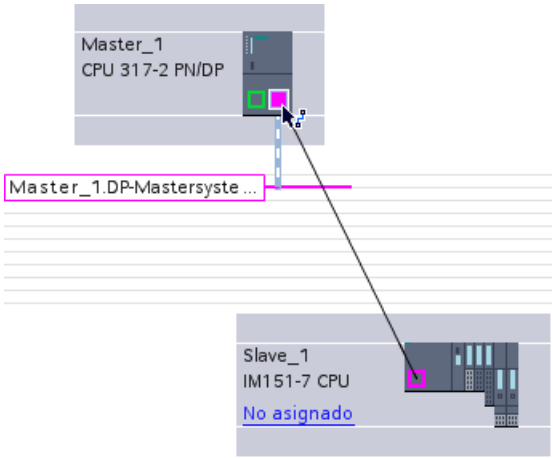
- Se encuentra en la vista de redes.
- El catálogo de hardware está abierto.

Procedimiento

Para crear un sistema maestro DP, proceda del siguiente modo:

1. Seleccione un maestro DP del catálogo de hardware.
2. Arrastre el maestro DP a la superficie libre en la vista de redes.
3. Haga clic con el botón derecho del ratón en la interfaz DP del maestro DP.
4. Seleccione "Crear sistema maestro" en el menú contextual.

Se crea un sistema maestro DP con un maestro DP como único nodo. Cuando se conecta la interfaz DP de un esclavo DP con la interfaz DP del maestro DP, el esclavo DP es incluido en el sistema maestro.



Si ya se ha colocado un maestro DP y un esclavo DP en la vista de redes, éstos se pueden enlazar con Drag&Drop y crear así un sistema maestro DP. Proceda del siguiente modo:

1. Haga clic en la interfaz DP del maestro DP o del esclavo DP.
2. Manteniendo pulsado el botón del ratón, arrastre una conexión de la interfaz DP seleccionada a la interfaz DP del interlocutor deseado.

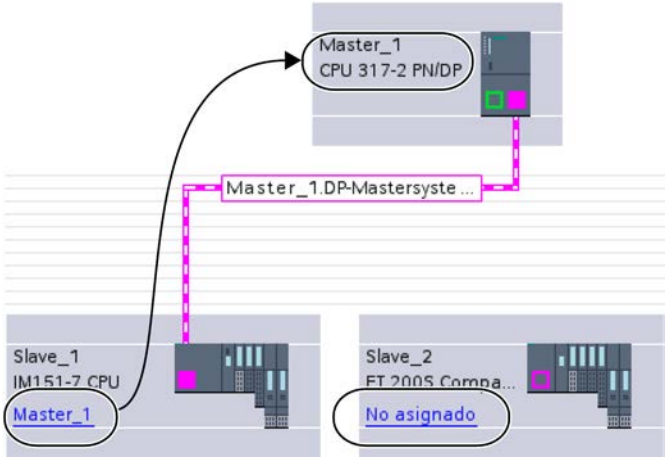
Se crea una subred con un sistema maestro DP entre maestro DP y esclavo DP.

Nota

Si se arrastra con el ratón una conexión entre la interfaz MPI de un maestro DP y la interfaz PROFIBUS de un esclavo DP, la interfaz MPI del maestro DP se configura automáticamente como interfaz PROFIBUS y se crea un sistema maestro DP.

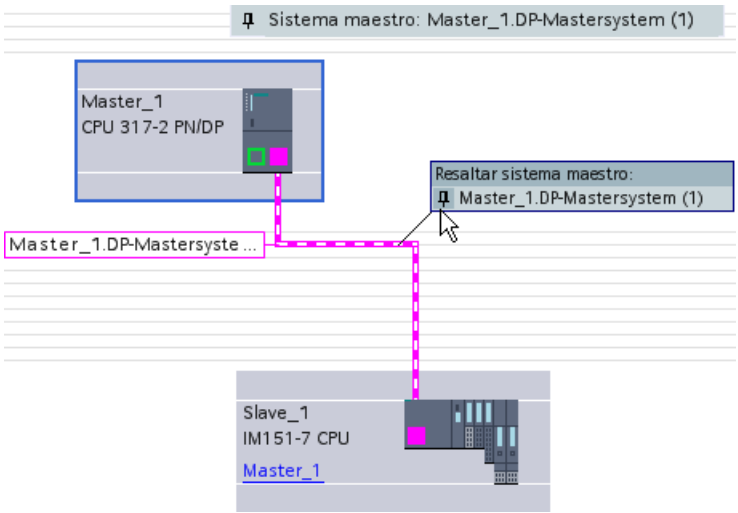
Visualizar el maestro DP en el esclavo DP

Si se conecta un módulo de cabecera que es esclavo DP (p. ej. del ET 200S) con un maestro DP, el nombre del maestro DP se muestra como hipervínculo en el esclavo DP. Si se hace clic en el hipervínculo, se selecciona el maestro DP asignado.



Resaltar el sistema maestro DP

Si se ha creado un nuevo sistema maestro DP, éste se representa resaltado. De este modo se reconoce rápidamente qué dispositivos pertenecen al sistema maestro DP. También se puede resaltar un sistema maestro DP arrastrando el puntero del ratón a una subred. Se visualizan a continuación los nombres de los sistemas maestros DP existentes. Al hacer clic con el ratón en uno de los sistemas maestros DP visualizados, se resalta el correspondiente sistema maestro DP.



Existen varias posibilidades para desactivar después el resalte de un sistema maestro DP:

- Resalte otro sistema maestro.
- Haga clic en el alfiler con la denominación del sistema maestro situada en la esquina superior derecha de la vista de redes.

Consulte también

Acopladores DP/DP en el catálogo de hardware (Página 1389)

Principios básicos sobre la configuración de un sistema maestro DP (Página 1387)

Editar sistemas maestros e interfaces DP

Introducción

Después de crear un sistema maestro DP, existe también la posibilidad de desconectar el sistema maestro DP y sus componentes. De ahí pueden generarse subredes con esclavos DP y sin maestro DP.

Casi nunca es necesario editar las interfaces de un maestro DP.

Se puede modificar el nombre y el número del sistema maestro DP.

Desconexión de sistemas maestros DP

Si se ha configurado una CPU con interfaz PROFIBUS DP integrada o un CP PROFIBUS (configurable como esclavo DP inteligente) como maestro DP con sistema maestro, entonces es posible desconectar el sistema maestro DP del maestro DP. El dispositivo ya no está conectado al sistema maestro DP.

La desconexión de la subred en el maestro DP provoca que el sistema maestro deje de existir, porque ya no hay asignado ningún maestro DP. Sin embargo, los esclavos DP individuales continúan conectados entre sí vía la subred y se mantiene la comunicación directa configurada.

Si se borran los esclavos DP o se desconectan del sistema maestro, este último se mantiene en el maestro DP.

Requisitos

- Se encuentra en la vista de redes.
- Hay un sistema maestro DP con un maestro DP y al menos un esclavo DP.

Desconectar el maestro DP del sistema maestro DP

Para desconectar el sistema maestro DP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la interfaz DP del maestro DP.
2. Seleccione "Desconectar del sistema maestro" en el menú contextual.

El maestro DP seleccionado se desconecta del sistema maestro DP. Queda una subred con los esclavos DP.

Agregar maestro DP al sistema maestro DP

Para asignar nuevamente un maestro DP a una subred, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la interfaz DP de un maestro DP.
2. Seleccione "Crear sistema maestro" en el menú contextual.
3. Arrastre el sistema maestro DP nuevo a las interfaces DP de los esclavos DP.

El maestro DP constituye con los esclavos DP otro sistema maestro DP.

Establecer interfaz PROFIBUS DP online

Para permitir que la interfaz MPI/DP integrada de una CPU participe activamente en el PROFIBUS DP, proceda del siguiente modo:

1. Seleccione el símbolo de interfaz de la CPU en la vista de dispositivos o de redes.
2. En "Dirección MPI" de la ventana de inspección, seleccione el parámetro "Tipo de interfaz: PROFIBUS".
3. Dado el caso, modifique la dirección PROFIBUS propuesta.
4. Si es necesario y si la hay, seleccione una subred existente en "Subred".

Nota

Si se arrastra con el ratón una conexión entre la interfaz PROFIBUS de un esclavo DP y la interfaz MPI de un maestro DP, la interfaz MPI del maestro DP se configura automáticamente como interfaz PROFIBUS y se crea un sistema maestro DP.

Editar las propiedades de un sistema maestro DP

Para editar las propiedades de un sistema maestro DP, proceda del siguiente modo:

1. Desplace el puntero del ratón a una subred con sistema maestro DP.
2. En el aviso que aparece sobre los sistemas maestros DP disponibles, haga clic en el sistema maestro DP que quiera editar. El sistema maestro DP se resalta ahora en color.
3. Haga clic en el el sistema maestro DP resaltado en color.
4. Edite los atributos del el sistema maestro DP en "Propiedades > General" de la ventana de inspección.

Nota

Si se hace clic en una subred sin que se resalte un sistema maestro DP, es posible editar las propiedades de toda la subred en "Propiedades" de la ventana de inspección.

Insertar y configurar esclavos DP en el sistema maestro

Inserte en la vista de redes diversos esclavos DP directamente con Drag&Drop o haciendo doble clic en el catálogo de hardware.

Tipos de esclavos DP

En la configuración de esclavos DP, éstos se clasifican como:

- Esclavos DP compactos
  (módulos con entradas y salidas digitales/analógicas integradas, p. ej. ET 200L)
- Esclavos DP modulares
  (Módulos de interfaz con módulos S5 o S7 asignados, p. ej. ET 200M)
- Esclavos DP inteligentes (esclavos I)
  (SIMATIC S7-300 con p. ej. CP 342-5, CPU 315-2 DP o ET 200S con IM 151-7 CPU)

Reglas para slots

- En un sistema maestro DP se utiliza sólo un maestro DP y uno o varios esclavos DP.
- En un sistema maestro DP no puede haber más esclavos DP de los indicados en el maestro DP.

Nota

Para configurar el sistema maestro DP, tenga en cuenta los datos técnicos del maestro DP (número máx. de nodos, número máx. de slots, número máx. de datos de usuario). Es posible que no se utilice el número máximo de nodos debido a la limitación por slots o datos de usuario.

Requisitos

- Se encuentra en la vista de redes.
- Se ha creado un sistema maestro DP.

Insertar un esclavo DP en el sistema maestro DP

Para insertar un esclavo DP del catálogo de hardware en el sistema maestro DP, proceda del siguiente modo:

1. Seleccione un esclavo DP modular (p. ej. ET 200M) del catálogo de hardware.
2. Arrastre el esclavo DP con Drag&Drop del catálogo de hardware a la vista de redes.
3. Arrastre una conexión de la interfaz DP del maestro DP o del sistema maestro DP resaltado a la interfaz DP del nuevo esclavo DP.

Se crea automáticamente un sistema maestro DP y el esclavo DP se conecta también automáticamente al maestro DP como esclavo DP.

Nota

En un sistema maestro DP resaltado se puede hacer doble clic en el esclavo DP deseado del catálogo de hardware. De este modo, el esclavo DP es incluido automáticamente en el sistema maestro DP resaltado.

Desconectar el esclavo DP del sistema maestro DP

Para desconectar un esclavo DP del sistema maestro DP, proceda del siguiente modo:

1. En la vista de redes, haga clic con el botón derecho del ratón en la interfaz DP del esclavo DP.
2. Seleccione el tipo de desconexión del sistema maestro DP en el menú contextual:
   – "Deshacer la conexión con la subred": la conexión PROFIBUS se deshace y el dispositivo ya no está conectado al sistema maestro DP o a una subred.
   – "Desconectar del sistema maestro": el esclavo DP queda conectado a la subred, pero ya no está asignado al sistema maestro DP como esclavo DP.

El esclavo DP seleccionado se desconecta del sistema maestro DP.

Asignar un esclavo DP a un sistema maestro DP nuevo

Para asignar un esclavo DP existente a un sistema maestro DP nuevo, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la interfaz DP del esclavo DP.
2. En el menú contextual, elija "Asignar al maestro nuevo". Es indiferente si el esclavo DP ya está asignado a otro sistema maestro DP.
3. En una ventana de selección se ve la lista de maestros DP disponibles. Seleccione el maestro DP a cuyo sistema maestro DP quiera conectar el esclavo DP.

Se ha asignado el esclavo DP seleccionado a un sistema maestro DP nuevo. Del mismo modo, es posible conectar un esclavo DP a otra subred mediante la función "Asignar a la subred nueva", sin acoplarlo a un sistema maestro DP existente.

Configurar esclavo DP

Para configurar un esclavo DP, proceda del siguiente modo:

1. Cambie a la vista de dispositivos del esclavo DP.
2. Seleccione los módulos deseados en el catálogo de hardware. En los esclavos DP modulares, los módulos posibles están situados en el catálogo de hardware debajo de la correspondiente "Familia" de esclavos DP. Utilice para ello la función de filtro del catálogo de hardware.
3. Arrastre los módulos deseados del catálogo de hardware al rack del esclavo DP.
4. Configure el esclavo DP y los módulos insertados en la ventana de inspección.

Consulte también

Crear un sistema maestro DP (Página 1394)

Esclavos DP en el catálogo de hardware (Página 1389)

Examinar el catálogo de hardware (Página 512)

Trabajar con racks (Página 514)

Insertar un módulo en un rack (Página 523)

Sugerencia: Configuración rápida de sistemas maestros

Si el sistema maestro DP tiene muchos esclavos DP, asigne todos los esclavos DP posicionados mediante Drag & Drop a un maestro en un solo paso.

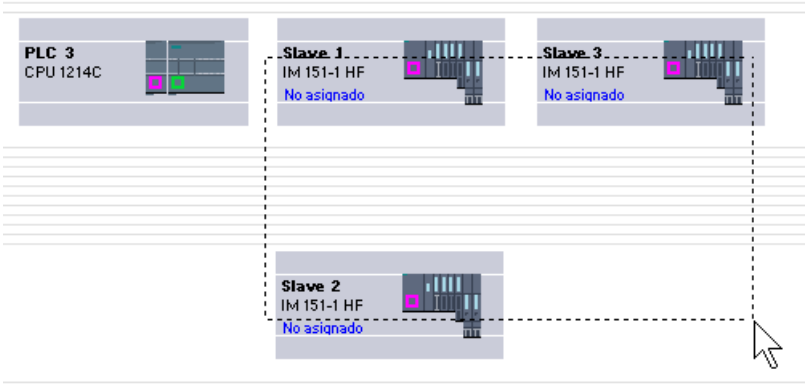
Requisitos

El maestro DP y los esclavos DP están posicionados en la vista de redes.

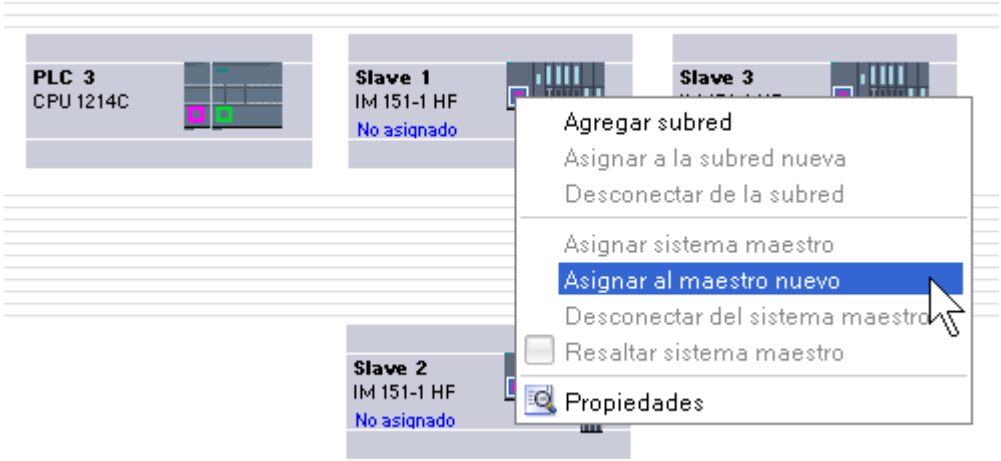
Asignar esclavos DP a un sistema maestro DP

Proceda del siguiente modo:

1. Elija un factor de zoom apropiado de manera que pueda ver tantos esclavos DP como sea posible en la vista de redes.
2. Disponga los esclavos DP en dos filas como máximo.
3. Seleccione todas las interfaces DP con el puntero del ratón (¡no los dispositivos!). Esto sólo resultará si empieza a tirar del puntero del ratón fuera del primer esclavo y suelta el botón del ratón en el último esclavo DP (selección con lazo).



4. Elija en el menú contextual "Asignar al maestro nuevo" y seleccione el cuadro de diálogo siguiente la interfaz DP correspondiente del maestro DP.



5. Los esclavos DP se conectan automáticamente en red con el maestro DP y forman conjuntamente un sistema maestro DP.

Nota

Con un sistema maestro DP resaltado puede hacer doble clic en un esclavo DP en el catálogo de hardware y agregar así rápidamente otros esclavos DP. De este modo, el esclavo DP es incluido automáticamente en el sistema maestro DP resaltado.

Asignar esclavo DP a grupo SYNC/FREEZE

Introducción

Un maestro DP con la respectiva funcionalidad puede enviar los comandos de control SYNC y/o FREEZE simultáneamente a un grupo de esclavos DP para sincronizar los esclavos DP. Para ello es necesario asignar los esclavos DP a los grupos SYNC/FREEZE.

Requisitos

Debe haberse creado un sistema maestro DP.

Procedimiento

Para asignar un esclavo DP a un grupo SYNC/FREEZE, proceda del siguiente modo:

1. Seleccione la interfaz del esclavo DP que quiera asignar a un grupo en la vista de dispositivos o de redes.
2. En el grupo "SYNC/FREEZE" de la ventana de inspección, seleccione la casilla de verificación de los grupos SYNC/FREEZE deseados.

Nota

Cada esclavo DP puede ser asignado como máximo a un grupo SYNC/FREEZE.

Excepción: En caso de utilizar un CP 342-5 como maestro DP, se puede asignar a cada esclavo DP un máximo de ocho grupos SYNC/FREEZE.

Tenga en cuenta la documentación del CP 342-5.

Información importante sobre los comandos de control SYNC y FREEZE

Con los comandos de control SYNC y FREEZE del maestro DP es posible sincronizar los esclavos DP de forma controlada por eventos. El maestro DP envía simultáneamente los comandos de control a un grupo de esclavos DP de su sistema maestro. No se consideran los esclavos DP que fallan o que están notificando un diagnóstico.

Para poder sincronizar mediante comandos de control es necesario haber asignado los esclavos DP a los grupos SYNC/FREEZE.

Para una CPU S7 utilice la instrucción DPSYC_FR (SFC 11) para sincronizar el esclavo DP.

Al seleccionar el maestro DP se ve una lista de los ocho grupos SYNC/FREEZE en la ventana de inspección bajo "Propiedades > Interfaz DP > SYNC/FREEZE" o "Propiedades > Interfaz MPI/DP > SYNC/FREEZE" (cuando opera como interfaz PROFIBUS).

Nota

Hay esclavos DP que o bien solo son aptos para SYNC, o bien solo para FREEZE. Para saber cuál de estas funciones está soportada por un esclavo DP, consulte el área "SYNC/FREEZE" de las propiedades de la interfaz DP.

Para poder asignar un esclavo DP de este tipo a un grupo SYNC o FREEZE, es necesario limitar el grupo en cuestión a la función soportada por el esclavo DP en las propiedades de la interfaz del maestro DP.

Ejemplo: En un esclavo DP que solo es apto para FREEZE, es necesario desactivar la casilla de verificación "SYNC" en el grupo SYNC/FREEZE del maestro DP.

Comando de control SYNC

Con el comando de control SYNC, el maestro DP hace que un grupo de esclavos DP congele los estados de sus salidas al valor actual.

Los esclavos DP almacenan los datos de salida del maestro DP en los telegramas siguientes. Sin embargo, los estados de las salidas de los esclavos DP no se modifican.

Con cada nuevo comando de control SYNC, el esclavo DP pone sus salidas a los valores que ha guardado como datos de salida del maestro DP.

Las salidas sólo se vuelven a actualizar cíclicamente cuando el maestro DP envía el comando de control UNSYNC.

Comando de control FREEZE

Después de recibir el comando de control FREEZE del maestro DP, los esclavos DP de un grupo congelan el estado actual de sus entradas. En los telegramas cíclicos siguientes, los esclavos DP envían estos datos de entrada congelados al maestro DP.

Con cada nuevo comando de control FREEZE, los esclavos DP vuelven a congelar el estado actual de sus entradas.

El estado de las entradas de los esclavos DP sólo se vuelve a enviar cíclicamente al maestro DP cuando éste envía el comando de control UNFREEZE.

Consulte también

Crear un sistema maestro DP (Página 1394)

Ajustar propiedades de PROFIBUS (Página 1455)

Ejemplo: Configurar un ET 200S como esclavo DP

Configurar ET 200S

Introducción

Los esclavos DP y los dispositivos IO de la familia ET 200S se configuran como otros esclavos DP y dispositivos IO modulares. Los módulos para el ET 200S se encuentran en el catálogo de hardware bajo "Periferia descentralizada".

Reglas de slots para la configuración de un ET 200S

Para configurar el ET 200S se aplican las reglas siguientes:

- Inserte los módulos del ET 200S uno junto a otro sin dejar espacio.
- Slot 1: sólo para módulos de potencia PM-E o PM-D
- A la izquierda de un módulo electrónico (EM): sólo un EM o un módulo de potencia (PM-E o PM-D)
- A la izquierda de un arrancador de motor (MS): sólo un MS o un módulo de potencia PM-D, PM-D Fx (1..x..4) o PM-X

- A la izquierda de un PM-X: sólo un arrancador de motor o un PM-D
- Se permiten como máximo 63 módulos y un módulo de interfaz IM

Nota

Tenga en cuenta la asignación correcta de los rangos de tensión PM-E y EM.

Funcionamiento si la configuración teórica difiere de la real

Este parámetro controla el comportamiento durante el arranque y al extraer e insertar módulos durante el funcionamiento (intercambio de datos cíclico).

La casilla de verificación está activada (se permite el funcionamiento si la configuración teórica difiere de la configuración real):

- El módulo arranca incluso si la configuración teórica difiere de la real.
- El módulo permanece en el intercambio de datos cíclico después de extraerlo e insertarlo (sin producirse un fallo de la estación).

La casilla de verificación está desactivada (se prohíbe el funcionamiento si la configuración teórica difiere de la configuración real):

- El módulo no arranca (no pasa al intercambio de datos cíclico) si la configuración teórica difiere de la real.
- Si durante el funcionamiento se extraen e insertan módulos, la estación falla y el módulo ya no participa en el intercambio de datos cíclico.

Observe las condiciones incluidas en el manual del módulo correspondiente, qué condiciones provocan un fallo de la estación o impiden la entrada en el intercambio de datos aunque esté permitido el funcionamiento si la configuración teórica difiere de la real.

Supresión de frecuencias perturbadoras

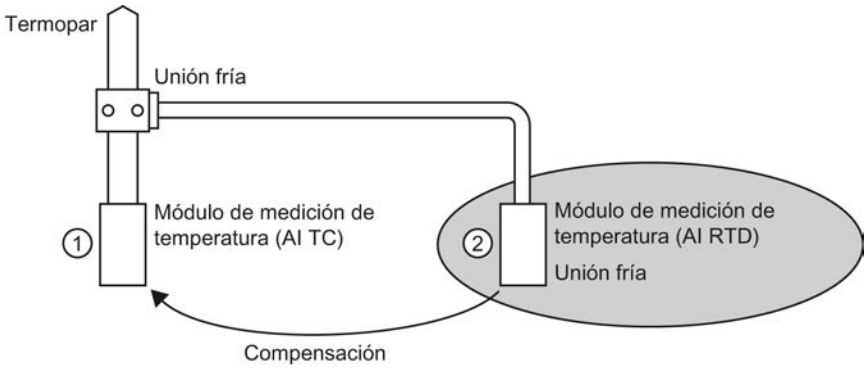
La frecuencia de la red de tensión alterna puede provocar interferencias en el valor medido, especialmente al medir en rangos de pequeña tensión y con termopares. Ajuste para todo el ET 200S la frecuencia de la tensión de red presente en su instalación.

Los módulos electrónicos analógicos AI ajustan el tiempo de integración y el tiempo de conversión en función de la selección de la supresión de frecuencia perturbadora.

Parametrizar una unión fría

Una unión fría es la conexión de un termopar con un cable de entrada (por norma general en la caja de bornes). La tensión que se genera aquí debido al efecto de la temperatura falsifica el valor de temperatura que mide el módulo.

En un ET 200S es posible parametrizar un canal del módulo AI RTD como unión fría. Gracias a la temperatura que mide dicho módulo en la unión fría, otros módulos AI TC pueden compensar los valores medidos.



① Parametrización del AI TC:
→ Selección de la unión fría utilizada

② Parametrización del AI RTD:
→ Activación de la unión fría
→ Definición de slot y canal del AI RTD

Particularidades de la parametrización de uniones frías

La parametrización de uniones frías se describe mediante ejemplos de uso de una termorresistencia Pt 100 "Climatiz." para registrar la temperatura de la unión fría.

Para parametrizar launión fría, proceda del siguiente modo:

1. Posicione un módulo electrónico RTD analógico en la vista de dispositivos del ET 200S, p. ej. un 2AI RTD HF. El módulo debe soportar el modo de medición Temorresistencia (lineal, a 4 hilos) y el rango de medición PT 100 climat.
2. Seleccione el módulo en el rack.
3. En la ventana de inspección, ajuste en "Propiedades > General > Entradas" un canal para la función de unión fría.
   Tipo de medición: Termorresistencia (lineal, a 4 hilos)
   Rango de medición: Pt 100 climat.
4. Seleccione el ET 200S (es decir, el módulo de interfaz).
5. En la ventana de inspección active la casilla de verificación "Unión fría" en "Propiedades > General > Parámetros del módulo > Unión fría" e indique el slot y el número de canal del módulo RTD en cuestión.
6. Posicione el módulo electrónico analógico para medir la temperatura con termopar (módulo TC) y seleccione en la ventana de inspección la opción "RTD" como unión fría en "Propiedades > General > Entradas > Canal x".

Información adicional

Para obtener información adicional sobre los diversos tipos y posibilidades de utilización de módulos en ET 200S, consulte las instrucciones de servicio y el manual del equipo "Sistema de periferia descentralizada ET 200S".

Para obtener información adicional sobre el procesamiento de valores analógicos, consulte la documentación del sistema de periferia descentralizada ET 200S.

Consulte también

Configurar ET 200S en modo DPV1 (Página 1413)

Documentación del ET 200S (http://support.automation.siemens.com/WW/view/de/6008567/0/es)

Comprimir direcciones

Introducción

Los esclavos DP y los dispositivos IO de la familia ET 200S se configuran como otros esclavos DP y dispositivos IO modulares. Además de las funciones habituales de los esclavos DP y dispositivos IO modulares, el ET 200S incorpora la función "Comprimir direcciones":

Los módulos electrónicos digitales con un área de direcciones necesaria de 2 ó 4 bits ocupan por principio 1 byte cuando se insertan en la vista de dispositivos. Con todo, el área de direcciones realmente ocupada se puede comprimir después de la configuración con la función "Comprimir direcciones".

	Ajuste predeterminado	Después de "Comprimir direcciones"
Módulo	Dirección I	Dirección I
2DI (2 bits)	Byte 10	10.0...10.1
4DI (4 bits)	Byte 11	10.2...10.5

Requisitos

- Se encuentra en la vista de dispositivos.
- Hay un ET 200S, p. ej. IM 151-1.
- Hay algunos módulos electrónicos digitales en los slots, p. ej. 2DI AC120V ST.

Comprimir direcciones

Para comprimir direcciones, proceda del siguiente modo:

1. Seleccione los módulos electrónicos cuyas direcciones quiera comprimir. Para seleccionar varios módulos electrónicos, existen las posibilidades siguientes:
   - Haga clic en los módulos electrónicos sucesivamente pulsando a la vez la tecla <Mayús> o <Ctrl>.
   - Haga clic fuera del rack y arrastre el ratón rodeando los módulos electrónicos que quiera seleccionar.
2. Haga clic en "Comprimir direcciones" del menú contextual de los módulos electrónicos seleccionados.

Se comprimen por separado las áreas de direcciones de entradas, salidas y arrancadores de motor. En las columnas Dirección I y Dirección Q de la vista general de dispositivos se ven las direcciones comprimidas.

Formación de direcciones y estructura de las direcciones comprimidas

En caso de utilizar la función "Comprimir direcciones", las direcciones de los módulos electrónicos seleccionados se comprimen de acuerdo con las reglas siguientes:

- El principio del área de direcciones está determinado por la dirección más baja de los módulos electrónicos seleccionados: X.0.
- Si la dirección del bit no es "0", automáticamente se utiliza la siguiente dirección (libre) del byte a partir de la cual se puede introducir el área seleccionada: (X+n).0.
- Cuando ya no queda ningún área conexa, se comprime automáticamente en los huecos de direcciones existentes.

Los módulos electrónicos con direcciones comprimidas y la misma dirección de byte constituyen un grupo de compresión.

Descomprimir direcciones

Para descomprimir direcciones, proceda del siguiente modo:

1. Seleccione uno o varios módulos electrónicos con dirección comprimida.
2. Haga clic en "Descomprimir direcciones" del menú contextual de los módulos electrónicos seleccionados.

Los grupos de compresión de los módulos electrónicos seleccionados se descomponen y las direcciones comprimidas de los respectivos módulos electrónicos se descomprimen.

Si se borran o desplazan módulos electrónicos de un grupo de compresión o bien se insertan módulos electrónicos en un slot libre dentro de un grupo de compresión, el grupo de compresión se descompone igualmente y las direcciones comprimidas se descomprimen.

Las direcciones iniciales de los módulos electrónicos descomprimidos se colocan en las siguientes direcciones de byte libres en cada caso.

Particularidades de los módulos electrónicos con direcciones comprimidas

Para un módulo electrónico con una dirección comprimida rigen las particularidades siguientes:

- Desde el punto de vista de la CPU, no es posible asignar slots al módulo electrónico. Por este motivo la instrucción GADR_LGC (SFC 5) devuelve para el slot real del módulo electrónico la información de error W#16#8099 "Slot no configurado".
- La instrucción LGC_GADR (SFC 49) y la ID SZL W#16#xy91 "Información de estado del módulo" de un módulo electrónico no se pueden evaluar.
- El módulo electrónico recibe una dirección de diagnóstico adicional mediante la funcionalidad DPV1, porque desde el punto de vista de la CPU no se pueden asignar alarmas debido a la dirección comprimida.
- La "Alarma de extracción/inserción" no es posible, porque las funciones "Comprimir direcciones" y "Alarma de extracción/inserción" se excluyen mutuamente.

Configuración futura con módulos de reserva

La configuración futura permite preparar el ET 200S con interfaz PROFIBUS para futuras ampliaciones (opciones). En el presente apartado se describe la configuración futura con módulos de reserva.

Para ello, monte, cablee, configure y programe la configuración máxima prevista del ET 200S y utilice primero los módulos de reserva económicos (138-4AA00 ó 138-4AA10) en lugar de los módulos electrónicos necesarios más tarde en el montaje

Nota

El ET 200S se puede precablear completamente con el cableado principal, porque un módulo de reserva no está conectado con los bornes del módulo de pines ni con el proceso.

Requisitos

- Módulo de interfaz ET 200S
  - IM 151-1 STANDARD (a partir de 6ES7 151-1AA03-0AB0)
  - IM 151-1 FO STANDARD (a partir de 6ES7 151-1AB02-0AB0)
- Módulo de potencia con configuración futura
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Procedimiento

Para activar la configuración futura, proceda del siguiente modo:

1. Seleccione en la vista de dispositivos el IM 151-1 y active la casilla de verificación "Configuración futura" en "Propiedades > General > Configuración futura" de la ventana de inspección.
2. Active ahora la casilla de verificación numerada para los slots que están ocupados en principio por módulos de reserva en lugar de módulos electrónicos.
3. Seleccione en la vista de dispositivos el módulo de potencia y active la casilla de verificación "Configuración futura" en "Propiedades > General > Direcciones" de la ventana de inspección. Reserve para la interfaz de control y notificación el área de direcciones necesaria para ello en la memoria imagen de proceso de las salidas (IPS) y en la memoria imagen de proceso de las entradas (IPE).

Más adelante se pueden sustituir los módulos de reserva montados por los módulos configurados sin tener que actualizar la configuración.

Nota

Las direcciones de estas interfaces quedan reservadas en cuanto se activa la configuración futura en el módulo de potencia. La función "Configuración futura" también debe estar activada en el esclavo DP (módulo de interfaz IM 151-1 STANDARD). Si no está activada, las direcciones reservadas para la interfaz de control y notificación se liberan.

Recuerde que la activación y desactivación repetida de la configuración futura puede modificar la dirección de la interfaz de control y notificación.

La configuración futura sólo debe estar activada si hay exactamente un módulo de potencia PM-E DC24..48V o PM-E DC24..48V/AC24..230V.

Información adicional

Para obtener información adicional sobre la ocupación y el significado de los bytes en la memoria imagen de proceso, la configuración futura del PROFIBUS y la utilización de módulos de reserva, consulte la documentación del sistema de periferia descentralizada ET 200S.

Funcionamiento de la configuración futura en el arranque

Cuando "Arranque si la configuración real difiere de la teórica" está bloqueado, el ET 200S funciona aunque haya insertado un módulo de reserva en lugar del módulo electrónico configurado y la configuración futura esté activada para este slot.

Funcionamiento de la configuración futura durante el servicio

Durante el servicio, el funcionamiento de la configuración futura se distingue de esta manera:

- Configuración futura conectada para un slot:
  En este slot puede haber un módulo de reserva (opcional) o el módulo electrónico configurado. Si hay otro módulo en este slot, se notifica un diagnóstico (ningún módulo o módulo incorrecto).
- Configuración futura desconectada para un slot:
  En este slot puede haber sólo el módulo electrónico configurado. Con cualquier otro módulo se notifica un diagnóstico (ningún módulo o módulo incorrecto).

Valores sustitutivos del módulo de reserva

- Valor sustitutivo para entradas digitales: 0
- Valor sustitutivo para entradas analógicas: 0x7FFF

Control y evaluación en el programa del usuario

El ET 200S dispone de una interfaz de control y notificación para la función "Configuración futura".

La interfaz de control se encuentra en la memoria imagen de proceso de las salidas (IPS). Cada bit de esta área de direcciones controla uno de los slots 2 a 63:

- Valor del bit = 0: rige la parametrización de la configuración futura. Se permiten módulos de reserva.
- Valor del bit = 1: la parametrización de la configuración futura está suprimida. Los módulos de reserva no se aceptan en este slot:

La interfaz de notificación se encuentra en la memoria imagen de proceso de las entradas (IPE). Cada bit de esta área de direcciones da información sobre el módulo realmente insertado en los slots 1 a 63:

- Valor del bit = 0: en el slot se encuentran el módulo de reserva, un módulo incorrecto o un módulo extraído.
- Valor del bit = 1: en el slot se encuentra el módulo configurado.

Consulte también

¿Qué módulos soportan la configuración futura? (http://support.automation.siemens.com/WW/view/es/22564754)

Configuración futura sin módulos de reserva

La configuración futura permite preparar el ET 200S para futuras ampliaciones (opciones) incluso sin el montaje de módulos de reserva. En el presente apartado se describe la configuración futura sin módulos de reserva.

Nota

ET 200S con interfaz PROFINET

Esta descripción se refiere a ET 200S con interfaz PROFIBUS. La configuración futura para ET 200S con interfaz PROFINET funciona en principio como se describe aquí sin módulos de reserva. En vez de los módulos de interfaz DP aquí incluidos, deben utilizarse módulos de interfaz PN. Encontrará más información sobre la configuración futura en ET 200S con interfaz PROFINET en los correspondientes manuales de producto.

Requisitos

- Módulo de interfaz ET 200S
  - IM 151-1 HIGH FEATURE (a partir de 6ES7151-1BA02)
  - IM 151-1 STANDARD (a partir de 6ES7 151-1AA05-0AB0)
- Módulo de potencia con configuración futura
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Procedimiento

Para activar la configuración futura, proceda del siguiente modo:

1. Seleccione en la vista de dispositivos el IM 151-1 y active la casilla de verificación "Configuración futura" en "Propiedades > General > Configuración futura" de la ventana de inspección.
2. Seleccione en la vista de dispositivos el módulo de potencia y active la casilla de verificación "Configuración futura" en "Propiedades > General > Direcciones" de la ventana de inspección. Reserve para la interfaz de control y notificación el área de direcciones necesaria para ello en la memoria imagen de proceso de las salidas (IPS) y en la memoria imagen de proceso de las entradas (IPE).
3. Configure la configuración máxima del esclavo. La activación/desactivación de opciones se controla a través del programa de usuario.

Nota

Las direcciones de estas interfaces quedan reservadas en cuanto se activa la configuración futura en el módulo de potencia. La función "Configuración futura" también debe estar activada en el esclavo DP (módulo de interfaz IM 151-1). Si no está activada, las direcciones reservadas para la interfaz de control y notificación se liberan.

Recuerde que la activación y desactivación repetida de la configuración futura puede modificar la dirección de la interfaz de control y notificación.

La configuración futura sólo debe estar activada si hay exactamente un módulo de potencia PM-E DC24..48V o PM-E DC24..48V/AC24..230V.

Información adicional

Para obtener información adicional sobre la ocupación y el significado de los bytes en la memoria imagen de proceso, la configuración futura del PROFIBUS y la utilización de módulos de reserva, consulte la documentación del sistema de periferia descentralizada ET 200S.

Control y evaluación en el programa del usuario

El ET 200S dispone de una interfaz de control y notificación para la función "Configuración futura".

La interfaz de control se encuentra en la memoria imagen de proceso de las salidas (IPS). Cada bit de esta área de direcciones controla uno de los slots 1 a 63:

- Valor del bit = 0: El slot no existe en la configuración real.
- Valor del bit = 1: El slot existe en la configuración real.

La interfaz de notificación se encuentra en la memoria imagen de proceso de las entradas (IPE). Cada bit de esta área de direcciones da información sobre el módulo realmente insertado en los slots 1 a 63:

- Valor del bit = 0: El slot corresponde a una opción inexistente o el estado del módulo no es correcto.
- Valor del bit = 1: En el slot se encuentra el módulo configurado.

Consulte también

Aplicación de ejemplo para ET 200S, configuración futura sin módulos de reserva (http://support.automation.siemens.com/WW/view/es/29430270)

Configurar ET 200S en modo DPV1

Con PROFIBUS DPV1 se facilitan funcionalidades PROFIBUS ampliadas.

Requisitos

- Se encuentra en la vista de redes.
- Hay un maestro DP con funcionalidad DPV1.
- Se establece una conexión maestro-esclavo con PROFIBUS.

Procedimiento

Para conectar el esclavo DP en DPV1, proceda del siguiente modo:

1. Seleccione el esclavo DP.
2. Seleccione el modo de operación "DPV1" en la lista desplegable "DP Alarm Mode" de "Propiedades > Parámetros del módulo" de la ventana de inspección.

o bien

1. Seleccione el maestro DP.
2. Seleccione en la tabla de comunicación E/S la fila con la conexión entre el maestro DP y el esclavo DP deseado.
3. Seleccione el modo de operación "DPV1" en la lista desplegable "DP Alarm Mode" de "Propiedades > Parámetros del módulo" de la ventana de inspección.

Particularidades

Existen dependencias entre los distintos parámetros, que se representan a continuación:

Parámetro	Modo de operación DPV0	Modo de operación DPV1
Funcionamiento si configuración teórica diferente a configuración real	Utilizable sin limitaciones	Utilizable sin limitaciones
Alarma de diagnóstico (OB 82)	No utilizable, no activado	Utilizable sin limitaciones
Alarma de proceso (OB 40 hasta 47)	No utilizable, no activado	Utilizable sin limitaciones
Alarma de extracción/inserción (OB 83)	No utilizable, no activado	Sólo utilizable si las direcciones no están comprimidas. Si la alarma de extracción/inserción está activada, se activa automáticamente "Arranque si la configuración real difiere de la teórica".

Alarmas en módulos con direcciones comprimidas

Si el módulo puede disparar alarmas y la dirección del bit difiere de 0 debido a las direcciones comprimidas, es necesario asignar una dirección de diagnóstico al ET 200S en el cuadro de diálogo de direcciones.

La dirección de diagnóstico es necesaria para asignar una alarma DPV1 al módulo como causante de la alarma. La CPU sólo puede asignar una alarma cuando un módulo tiene esta dirección "descomprimida" y hay depositados datos sobre la alarma en la información de arranque del OB de alarma o en el búfer de diagnóstico. La CPU no puede utilizar para ello una dirección "comprimida".

Desde el punto de vista del procesamiento de alarmas (OB de alarma), el módulo tiene la dirección de diagnóstico asignada; para el procesamiento de datos de entrada y salida en el programa de usuario, el módulo tiene las direcciones comprimidas.

Nota

Si las direcciones del módulo están comprimidas, la alarma de extracción/inserción está bloqueada para el ET 200S.

Consulte también

Introducción a PROFIBUS DPV1 (Página 1433)

Configurar dispositivos DPV1 (Página 1434)

Configurar ET 200S (Página 1404)

Configuraciones adicionales

Configuración de otras funciones

El sistema periférico descentralizado ET 200S cuenta con numerosas funciones adicionales disponibles a través de módulos insertables (p. ej., módulos de comunicación). Encontrará las descripciones correspondientes en los apartados "Uso de las funciones tecnológicas" y "Configuraciones para acoplamientos punto a punto".

Otros sistemas de periferia descentralizada

Encontrará más información sobre otros sistemas de periferia descentralizada en los enlaces de "Consulte también".

Consulte también

Configurar y parametrizar 1Count24V (Página 9616)

Configurar y parametrizar 1Count5V (Página 9678)

Configurar y parametrizar 1SSI (Página 10870)

Configurar y parametrizar 1PosU (Página 11139)

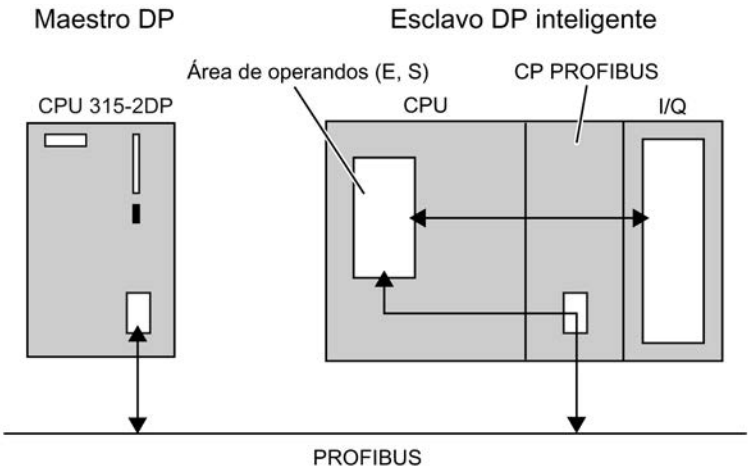
Configurar y parametrizar 1STEP 5V (Página 11189)

Configurar esclavos DP inteligentes

Insertar esclavo I en un sistema maestro DP

Introducción

Una característica del esclavo DP inteligente (esclavo I) es que los datos de entrada/salida no son facilitados al maestro DP directamente por una entrada/salida real, sino por una CPU de preprocesamiento. Dicha CPU constituye el esclavo I junto con el CP.



Diferencia: esclavo DP - esclavo DP inteligente

En un esclavo DP, el maestro DP accede a las entradas/salidas descentralizadas.

En un esclavo DP inteligente, el maestro DP no accede a las entradas/salida conectadas del esclavo DP inteligente, sino a un área de transferencia del área de direcciones E/S de la CPU de preprocesamiento. El programa de usuario de la CPU de preprocesamiento debe procurar el intercambio de datos entre el área de operandos y las entradas/salidas.

Nota

Las áreas E/S configuradas para el intercambio de datos entre maestro DP y esclavo DP no pueden ser utilizadas por módulos E/S.

Aplicaciones

Configuraciones con esclavos DP inteligentes:

- Intercambio de datos esclavo I <> maestro DP
- Comunicación directa esclavo DP > esclavo I

Procedimiento

Para insertar un esclavo I en un sistema maestro DP, proceda del siguiente modo:

1. Arrastre dos dispositivos con interfaz PROFIBUS DP del catálogo de hardware a la vista de redes para la configuración como maestro DP y esclavo I (p. ej. CPU 317-2 DP).
2. Ponga el dispositivo que debe utilizarse como esclavo I al modo de operación "Esclavo" en "Propiedades > Interfaz DP" de la ventana de inspección. El dispositivo opera ahora como esclavo DP inteligente.
3. Arrastre una línea de conexión entre las interfaces DP de ambos dispositivos. De este modo se conecta el esclavo I con un maestro DP en un sistema maestro DP.

Resultado

Se ha establecido un sistema maestro DP con un maestro DP y un esclavo I.

Tenga en cuenta los siguientes ejemplos para configurar esclavos I:

- Configurar CP como esclavo I (Página 1421)
- Configurar CPU como esclavo I (Página 1421)
- Configurar esclavo DP como esclavo I (Página 1422)

Consulte también

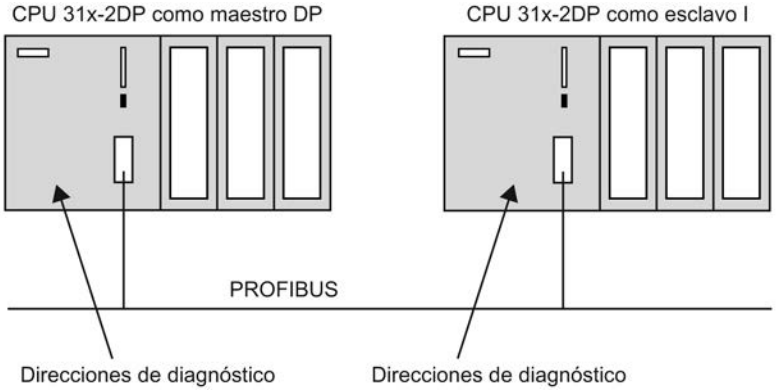
Alarmas en el esclavo I (Página 1418)

Configurar direcciones de diagnóstico (Página 1417)

Configurar direcciones de diagnóstico

Introducción

En la asignación de direcciones de diagnóstico para PROFIBUS DP, recuerde que se asignan direcciones de diagnóstico DP tanto al maestro DP como al I-Slave.



Direcciones de diagnóstico

En las direcciones de diagnóstico se distingue entre direcciones de diagnóstico en el maestro DP y en el I-Slave:

- Direcciones de diagnóstico en el maestro DP
  A través de las direcciones de diagnóstico asignadas al maestro DP, éste recibe información sobre el estado del esclavo DP o sobre una interrupción del bus.
- Direcciones de diagnóstico en el I-Slave
  A través de las direcciones de diagnóstico asignadas al esclavo I, éste recibe información sobre el estado del maestro DP o sobre una interrupción del bus.

Al crear la configuración maestro-esclavo I, STEP 7 asigna automáticamente direcciones de diagnóstico a las CPUs. La descripción siguiente muestra cómo modificar las direcciones de diagnóstico predefinidas.

Sugerencia: Para obtener una vista general de todas las direcciones asignadas a la CPU, incluidas las direcciones de diagnóstico, seleccione el área "Sinóptico de direcciones" en la ventana de inspección de la CPU seleccionada.

Requisitos

- Se encuentra en la vista de redes.
- Se ha creado un sistema maestro DP con maestro DP y I-Slave.

Procedimiento

Para editar las direcciones de diagnóstico, proceda del siguiente modo:

1. Seleccione el maestro DP o el I-Slave.
2. Seleccione la fila con el modo de operación "I-Slave" en la tabla de comunicación E/S.
3. Edite las direcciones de diagnóstico en "Propiedades > Comunicación de esclavo I" de la ventana de inspección.

O:

1. Seleccione el I-Slave.
2. Edite las direcciones de diagnóstico en "Propiedades > Interfaz DP > Modo de operación > Comunicación de esclavo I" de la ventana de inspección.

Pueden seleccionarse y editarse los campos con los valores de las direcciones de diagnóstico, tanto para el esclavo como para el maestro.

En las direcciones de diagnóstico se encuentran las direcciones para las transiciones de estado operativo y para el diagnóstico de dispositivo:

- Transiciones de estado operativo: transiciones de RUN > STOP o STOP > RUN
- Diagnóstico de dispositivo: p. ej. interrupción del bus por cortocircuito o conector desenchufado ("Fallo de estación")

Información adicional

Para obtener más información sobre direcciones de diagnóstico, consulte los manuales de cada una de las CPUs S7.

Consulte también

Direccionamiento de módulos (S7-300, S7-400) (Página 914)

Alarmas en el esclavo I

Tipos de alarma en función de DPV1 y DPV0

Los esclavos I pueden disparar alarmas en el maestro DP asignado con la instrucción avanzada "SALRM" (SFB 75). La tabla siguiente muestra los tipos de alarmas posibles en función de PROFIBUS DPV1 y DPV0:

Tipo de alarma	DPV0	DPV1
Alarma de diagnóstico (OB 82)	Sí	Sí
Alarma de proceso (OB 40 a 47)	Sí	Sí
Alarma de presencia de módulo (OB 83)	Sí (cuando el I-Slave soporta esta alarma)	Sí
Alarma de estado (OB 55)	No	Sí

Tipo de alarma	DPV0	DPV1
Alarma de actualización (OB 56)	No	Sí
Alarma específica del fabricante (OB 57)	No	Sí

Direcciones que disparan alarmas

Utilice las direcciones configuradas en "Propiedades > Interfaz DP > Modo de operación > Comunicación de esclavo I" de la ventana de inspección del esclavo I para disparar alarmas con la instrucción avanzada "SALRM". Estas direcciones no están asignadas a ningún módulo real, sino que corresponden a "slots" virtuales.

Las direcciones para transiciones del estado operativo y el diagnóstico de dispositivo no se pueden utilizar para disparar alarmas.

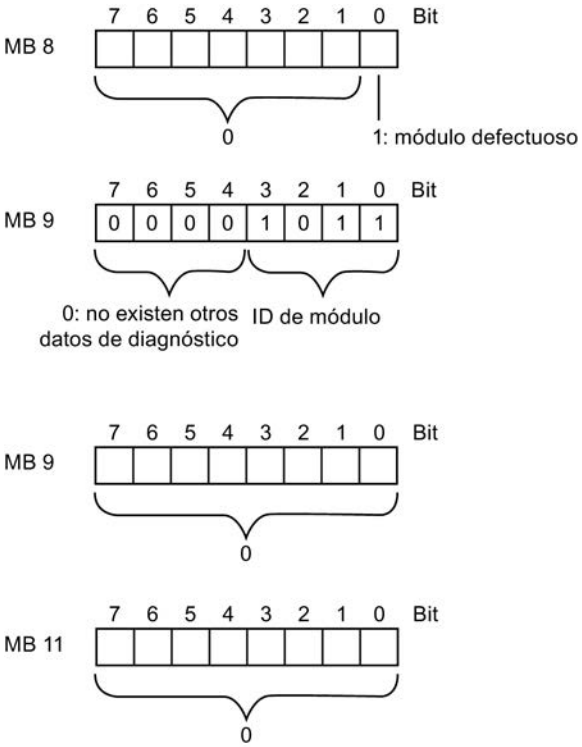
Procedimiento general de la generación de alarmas

El principio de la generación de alarmas se muestra con una alarma de diagnóstico.

- En el I-Slave se asigna la dirección de salida 0 a un slot virtual.
- La dirección de salida 0 se utiliza en el ejemplo para disparar una alarma de diagnóstico (OB 82) en el maestro DP.

Con la alarma de diagnóstico se transfieren datos mediante el programa de usuario (AINFO). Dichos datos deben corresponder a la configuración general de la información adicional de alarmas.

Se permite p. ej. la siguiente configuración simplificada:



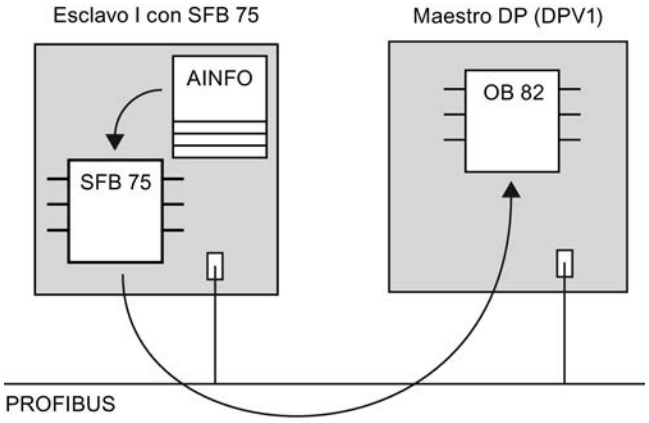
Información adicional

Para más información sobre la configuración representada más arriba, consulte el manual de referencia de las funciones estándar y de sistema del S7-300/400, tomo 2, cap. Datos de diagnóstico, y las de los manuales sobre las funciones de diagnóstico y de sistema del S7-1500..

Nota

Los datos de estado del módulo y el LED de error "SF" del I-Slave (S7-300/400) influyen en la información adicional de alarmas. Asimismo, la alarma influye en los datos de estado del módulo y el LED de error del maestro DP asignado. Por ello, es necesario tener en cuenta el significado de los registros de diagnóstico (registro 0 y registro 1) en la composición de los datos para la información adicional de alarmas.

La figura siguiente muestra el proceso de generación de alarmas:



Una llamada de la instrucción avanzada "SALRM" puede ser la siguiente:

AWL

```
CALL   FB    SALRM , DB75
       REQ   :=M0.0
       ID    :=DW#16#8000    //Dirección de salida 0
       ATYPE :=1             //Alarma de diagnóstico
       ASPEC :=1             //Alarma entrante, racx de ampliación averiado
       LEN   :=4             //Longitud de información de alarma 4 bytes
       DONE  :=M1.3
       BUSY  :=M1.4
       ERROR :=M1.5
       STATUS:=MD50
       AINFO :=P#M 8.0 BYTE 4 //Área para información de alarma
```

Consulte también

Introducción a PROFIBUS DPV1 (Página 1433)

Configurar dispositivos DPV1 (Página 1434)

Ejemplo de configuración de un CP como esclavo I

En este ejemplo se configura una estación con un CP 342-5 como esclavo I. Para ello, el CP 342-5 debe convertirse en un esclavo DP inteligente con el modo de operación "Esclavo DP".

Requisitos

- Se encuentra en la vista de redes.
- Hay un CP 342-5 en la vista de redes.
- Se ha equipado el CP 342-5 en la vista de dispositivos con módulos E/S.
- Hay un maestro DP (CPU con interfaz PROFIBUS DP integrada o CP con interfaz PROFIBUS DP) en la vista de redes.

Procedimiento

Para configurar un CP como esclavo I, proceda del siguiente modo:

1. Seleccione el CP 342-5.
2. Conmute el CP 342-5 al modo de operación "Esclavo DP" en "Propiedades > Modo de operación" de la ventana de inspección.
3. Arrastre una conexión de la interfaz DP del CP 342-5 a la interfaz DP del maestro DP.
4. Seleccione el CP 342-5.
5. Haga clic en "Comunicación del esclavo I" en "Propiedades > Modo de operación" de la ventana de inspección.
6. Edite la asignación de direcciones entre maestro DP y esclavo I.

Alternativamente, después de seleccionar el CP 342-5 en la tabla de comunicación E/S es posible elegir la fila con el modo de operación "Esclavo I" para seguidamente editar la asignación de direcciones en "Propiedades" y "Comunicación del esclavo I" de la ventana de inspección.

Información adicional

Para más información sobre el intercambio de datos entre la CPU de preprocesamiento y el CP 342-5 DP en el esclavo DP, consulte el manual para NCM S7 Industrial Ethernet.

Ejemplo de configuración de una CPU como esclavo I

En este ejemplo se configura una estación con una CPU 31X-2 DP con interfaz DP integrada (p. ej. CPU 315-2 DP) como esclavo I. La CPU 31X-2 DP precisa una interfaz DP integrada y debe ser convertida en un esclavo DP inteligente con el modo de operación "Esclavo DP".

Requisitos

- Se encuentra en la vista de redes.
- Hay una CPU 315-2 DP en la vista de redes.

- Se ha equipado la CPU 315-2 DP en la vista de dispositivos con módulos E/S.
- Hay un maestro DP (CPU con interfaz PROFIBUS DP integrada o CP con interfaz PROFIBUS DP) en la vista de redes.

Procedimiento

Para configurar una CPU como esclavo I, proceda del siguiente modo:

1. Seleccione la CPU 315-2 DP.
2. Conmute la CPU 315-2 DP al modo de operación "Esclavo DP" en "Propiedades > Interfaz DP > Modo de operación" de la ventana de inspección.
3. Arrastre una conexión de la interfaz DP de la CPU 315-2 DP a la interfaz DP del maestro DP.
4. Seleccione la CPU 315-2 DP.
5. Haga clic en "Comunicación del esclavo I" en "Propiedades > Interfaz DP > Modo de operación" de la ventana de inspección.
6. Edite la asignación de direcciones entre maestro DP y esclavo I.

Alternativamente, después de seleccionar la CPU 315-2 DP en la tabla de comunicación E/S es posible elegir la fila con el modo de operación "Esclavo I" para editar entonces la asignación de direcciones en "Propiedades > Comunicación del esclavo I" de la ventana de inspección.

Información adicional

Para más información sobre la configuración de CPUs como esclavos I, consulte los manuales de referencia de las correspondientes CPUs.

Ejemplo de configuración de un esclavo DP como esclavo I

En este ejemplo se configura un ET 200S con el módulo de interfaz IM 151-7 CPU como esclavo I.

Nota

El módulo de interfaz del ET 200S se puede encontrar en el catálogo de hardware en "PLC > SIMATIC ET 200 PLC > ET 200S CPU".

Requisitos

- Se encuentra en la vista de redes.
- Hay un ET 200S con IM 151-7 CPU en la vista de redes.
- El ET 200S se ha equipado con módulos de ampliación E/S en la vista de dispositivos.
- Hay un maestro DP (CPU con interfaz PROFIBUS DP integrada o CP con interfaz PROFIBUS DP) en la vista de redes.

Procedimiento

Para configurar un esclavo DP como esclavo I, proceda del siguiente modo:

1. Seleccione el IM 151-7 CPU.
2. Arrastre una conexión de la interfaz DP del IM 151-7 CPU a la interfaz DP del maestro DP.

   Nota

   Algunos dispositivos tienen interfaces ajustadas para maestro o esclavo. En "Propiedades > Interfaz MPI/DP" y/o "Propiedades > Interfaz DP" se puede encontrar el ajuste del correspondiente modo de operación.

3. Seleccione el IM 151-7 CPU.
4. Seleccione la fila con el modo de operación "Esclavo I" en la tabla de comunicación E/S.
5. Edite la asignación de direcciones entre maestro DP y esclavo I en "Propiedades > Comunicación del esclavo I" de la ventana de inspección.

Información adicional

Para más información sobre la configuración de esclavos DP como esclavos I, consulte los manuales de producto y las instrucciones de servicio de los sistemas de periferia descentralizada ET 200.

Configurar la comunicación directa

Información importante sobre la comunicación directa

Introducción

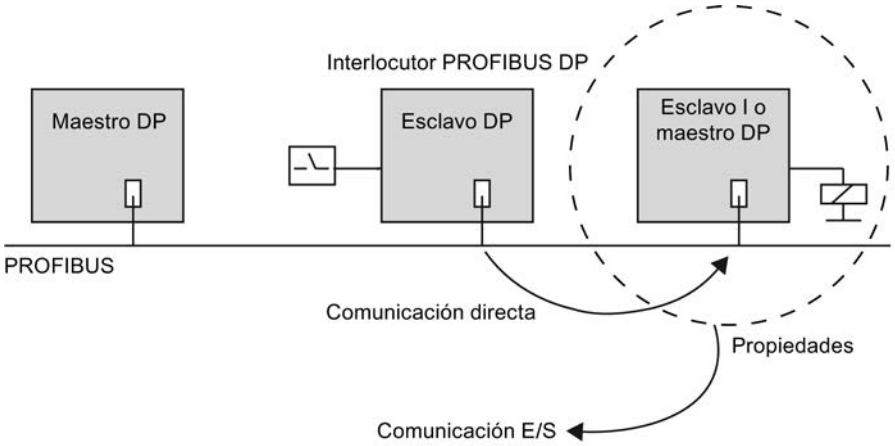
La comunicación directa es otra forma de comunicación E/S que puede ser efectiva más allá de un sistema maestro DP. En la comunicación directa, el receptor puede leer datos pero no escribirlos.

En un sistema maestro DP, un maestro DP controla exclusivamente los esclavos DP que le han sido asignados. Con la comunicación directa, otro maestro DP o esclavo DP puede "escuchar" como receptor en la subred PROFIBUS qué datos de entrada envía un esclavo DP a su maestro DP. Esta transferencia de datos entre esclavos DP se denomina comunicación directa.

Funcionamiento

En una configuración para la comunicación directa, las áreas de direcciones de entrada locales de un esclavo DP inteligente o de un maestro DP se asignan a las áreas de direcciones de entrada de un interlocutor PROFIBUS DP.

Con las áreas de direcciones de entrada asignadas, el esclavo DP inteligente o el maestro DP reciben datos de su interlocutor PROFIBUS DP.



Posibilidades de configuración

Posibles casos de aplicación para la comunicación directa:

- Configuración con esclavos DP inteligentes: esclavo DP > esclavo I
- Configuración con dos sistemas maestros DP: esclavo DP > maestro DP

Número de interlocutores PROFIBUS DP conectables

La suma de los interlocutores PROFIBUS DP que se conectan directamente a una interfaz DP o a los que se accede a través de dicha interfaz mediante la comunicación directa, está limitada por las interfaces. A la interfaz MPI/DP pueden acceder un máximo de 32 interlocutores PROFIBUS DP.

Consulte también

Configuraciones con varios sistemas maestros DP (Página 1391)

Configuraciones con esclavos DP inteligentes (Página 1392)

Configurar la comunicación directa

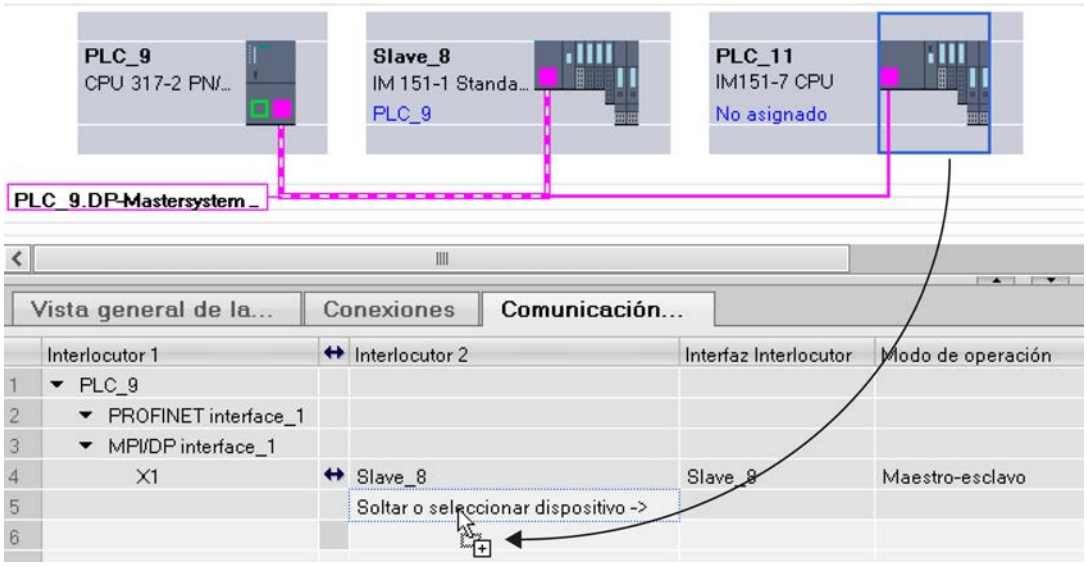
Requisitos

- Se encuentra en la vista de redes.
- Existe un sistema maestro DP con módulos de entrada en el esclavo DP.
- Un maestro DP o esclavo DP está conectado como "receptor" al sistema maestro DP.

Crear una comunicación directa

Para crear la comunicación directa, proceda del siguiente modo:

1. Seleccione el participante deseado de la comunicación (emisor o receptor) haciendo clic con el ratón.
2. Manteniendo pulsado el botón izquierdo del ratón, arrastre el segundo interlocutor al campo "Soltar o seleccionar dispositivo" de la columna "Interlocutor 2" de la tabla de comunicación E/S.



De manera alternativa, haga clic directamente en el campo "Soltar dispositivo aquí..." y seleccione el interlocutor deseado en la lista desplegable.

Se ha creado la relación de comunicación entre dos dispositivos con Drag&Drop dentro de la tabla. En la tabla de comunicación E/S se muestra el modo de operación "Comunicación directa" en una fila nueva.

Nota

En la comunicación directa entre dos esclavos I, siempre se selecciona en primer lugar el emisor. A continuación arrastre el receptor hasta el campo "Soltar o seleccionar dispositivo", o bien selecciónelo en la lista desplegable. El orden determina el sentido de la comunicación directa.

Configurar la comunicación directa

Para configurar la comunicación directa, proceda del siguiente modo:

1. Seleccione un dispositivo de la comunicación directa.
2. Seleccione la fila con el modo de operación "Comunicación directa" en la tabla de comunicación E/S.
3. En la ventana de inspección seleccione "Propiedades > Comunicación directa".

4. En "Comunicación directa", en la tabla del área de transferencia, cree un área de transferencia nueva:
   En la columna "Área de transferencia", haga clic en <"Agregar>" y, en la lista desplegable "Tipo", elija el tipo de comunicación "DX" para la comunicación directa. El área de transferencia nueva también se puede crear simplemente ajustando en una columna nueva el tipo de comunicación en "Tipo". El área de transferencia se crea a continuación automáticamente.
5. En "Comunicación directa", haga clic en el área de transferencia recién creada.
   Se abre la vista de detalle del área de transferencia.
6. Edite las propiedades del emisor y receptor y los parámetros generales de la comunicación directa.

Se ha configurado la comunicación directa.

Borrar la comunicación directa

Para borrar la comunicación directa, proceda del siguiente modo:

1. Seleccione la fila con el modo de operación "Comunicación directa" en la tabla de comunicación E/S.
2. Pulse <Supr> o seleccione el comando "Borrar" en el menú contextual.

Ejemplo de configuración de la comunicación directa

Introducción

En este ejemplo se configuran tres CPUs en una configuración maestro-esclavo para la comunicación directa. A continuación, se asignan nombres descriptivos y se configuran las áreas de direcciones.

Utilice las siguientes CPUs con la siguiente configuración:

CPU	Tarea	Dirección	Datos
317-2 PN/DP	Maestro DP	I 200	8 palabras del emisor
315-2 DP	Emisor	Q 100	8 palabras hacia el maestro
317-2 DP	Receptor	I 120	Primeros 2 bytes de las 8 palabras del emisor

Requisitos

- Se encuentra en la vista de redes.
- El catálogo de hardware está abierto.

Crear un sistema maestro PROFIBUS DP

Para crear un sistema maestro PROFIBUS DP, proceda del siguiente modo:

1. Arrastre con Drag&Drop las CPUs 315-2 DP, 317-2 DP y 317-2 PN/DP del catálogo de hardware hacia la vista de redes.
2. Introduzca nombres descriptivos para las CPUs:
   - Seleccione la CPU 317-2 PN/DP.
   - En "Propiedades > General" de la ventana de inspección, dé el nombre "Maestro DP" a la CPU.

   Proceda de forma análoga con las demás CPUs y póngales los nombres "Emisor" y "Receptor".
3. Seleccione la CPU 317-2 PN/DP y, si es necesario, modifique el tipo de interfaz de "MPI" a "PROFIBUS" en "Propiedades > Interfaz MPI/DP > Dirección MPI".
   Nota: Si el tipo de interfaz de la interfaz MPI/DP está ajustado a "PROFIBUS", la entrada no es "Dirección MPI" sino "Dirección PROFIBUS".
4. A continuación, seleccione las CPUs 315-2 DP y 317-2 DP sucesivamente y elija el modo de operación "Esclavo" en "Propiedades > Interfaz DP > Modo de operación" de la ventana de inspección.
   A partir de ahora, ambas CPUs 315-2 DP "Emisor" y 317-2 "Receptor" funcionan como esclavos I.
5. Manteniendo pulsado el botón del ratón, arrastre las conexiones PROFIBUS de la CPU maestra hacia las dos CPUs esclavas.

Se ha creado un sistema maestro PROFIBUS con la CPU 317-2 PN/DP como maestro DP y las CPUs 315-2 DP "Emisor" y 317-2 DP "Receptor" como esclavos I. En la tabla de comunicación E/S de las CPUs se han creado nuevas filas para los interlocutores. En la columna "Modo de operación" aparece "I-Slave".

Crear una comunicación directa

Para crear la comunicación directa dentro del sistema maestro PROFIBUS, proceda del siguiente modo:

1. Seleccione el "Emisor".
2. Manteniendo pulsado el botón izquierdo del ratón, arrastre el "Receptor" al campo "Soltar dispositivo aquí" de la columna "Interlocutor 2" de la tabla de comunicación E/S.
   Nota: También se puede abrir la lista desplegable del campo "Soltar dispositivo aquí" de la columna "Interlocutor 2" y seleccionar en ella el "Receptor".

Se ha creado una comunicación directa entre los dos I-Slaves. Se ha creado una nueva fila en la tabla de comunicación E/S de los dos I-Slaves. En la columna "Modo de operación" aparece "Comunicación directa".

Nota

Los símbolos de flecha de la tabla de comunicación E/S entre las columnas "Interlocutor 1" e "Interlocutor 2" muestran los sentidos de las relaciones de comunicación. En las relaciones maestro-esclavo o maestro-I-Slave, las flechas señalan en ambos sentidos. En las relaciones de comunicación entre I-Slaves o en la comunicación directa, las flechas señalan del emisor al receptor.

Configurar la comunicación maestro-esclavo

Para seleccionar la comunicación maestro-esclavo, proceda del siguiente modo:

1. Seleccione uno de los dos interlocutores de la comunicación maestro-esclavo (maestro DP CPU 317-2 PN/DP o I-Slave CPU 315-2 DP "Emisor")
2. En la tabla de comunicación E/S haga clic en la fila que contiene los dos interlocutores correspondientes y el modo de operación "I-Slave".
3. Seleccione "Propiedades > Comunicación de esclavo I" de la ventana de inspección.

O:

1. Seleccione el I-Slave CPU 315-2 DP "Emisor".
2. Seleccione "Propiedades > Interfaz DP > Modo de operación > Comunicación de esclavo I" de la ventana de inspección.

Para configurar la comunicación maestro-esclavo, proceda del siguiente modo:

1. Cree un área de transferencia nueva en "Comunicación de esclavo I" de la tabla de áreas de transferencia:
   En la columna "Área de transferencia", haga clic en "<Agregar>" y, en la lista desplegable "Tipo", seleccione el tipo de comunicación "MS" para la comunicación maestro-esclavo. El área de transferencia nueva también se puede crear simplemente ajustando en una columna nueva el tipo de comunicación en "Tipo". El área de transferencia se crea a continuación automáticamente.
2. En "Comunicación de esclavo I", haga clic en el área de transferencia recién creada.
   Se abre la vista de detalle del área de transferencia.
3. Introduzca los siguientes datos:
   Para "Maestro DP":
   - Tipo de dirección I
   - Dirección inicial 200
   - Longitud 8
   - Unidad WORD
   - Coherencia de la longitud total

   Para "Emisor":
   - Tipo de dirección Q
   - Dirección inicial 100

Una vez introducida la longitud de datos, se completan las direcciones de esclavo y maestro introducidas para la recopilación de datos necesaria. Al hacer clic en "Comunicación de esclavo I" de la ventana de inspección, en la vista general se ve la configuración actualizada de la comunicación maestro-esclavo.

Configurar la comunicación directa

Para configurar la comunicación directa, proceda del siguiente modo:

1. Seleccione uno de los dos interlocutores de la comunicación directa (esclavo I CPU 315-2 DP "Emisor" o esclavo I CPU 317-2 DP "Receptor").
2. Haga clic en la fila con el modo de operación "Comunicación directa" en la tabla de comunicación E/S.
3. En la ventana de inspección seleccione "Propiedades > Comunicación directa".

Para configurar la comunicación directa, proceda del siguiente modo:

1. Cree un área de transferencia nueva en "Comunicación directa" de la tabla de áreas de transferencia:
   En la columna "Área de transferencia", haga clic en "<Agregar>" y, en la lista desplegable "Tipo", seleccione el tipo de comunicación "DX" para la comunicación directa. El área de transferencia nueva también se puede crear simplemente ajustando en una columna nueva el tipo de comunicación en "Tipo". El área de transferencia se crea a continuación automáticamente.
2. En "Comunicación directa", haga clic en el área de transferencia recién creada.
   Se abre la vista de detalle del área de transferencia.
3. Introduzca los siguientes datos:
   Para "Emisor":
   – Dirección 100
   – Longitud de datos 1
   – Unidad WORD

   Para "Receptor":
   – Dirección 120

Nota

Si el emisor en comunicación directa es un esclavo DP normal no se configuran áreas de transferencia. En vez de ello, se debería agregar al emisor un módulo de entrada.

Particularidad

En principio se puede ajustar una dirección inicial mayor que 200, p. ej., 202. La longitud de los datos coherentes se adaptará automáticamente. Además, también puede ajustarse una longitud menor que la especificada por el emisor (p. ej. 1 byte).

Nota

Si en el emisor se configura una longitud coherente de 3 o más de 4 bytes y se transmiten los datos con la instrucción avanzada de la función de sistema "DPWR_DAT" (SFC 15), entonces el receptor deberá utilizar siempre la instrucción avanzada "DPRD_DAT" (SFC 14) aunque, por ejemplo, sólo se lea 1 byte.

Si para la lectura de 1 byte se utiliza una operación de carga (L EB..) en vez de DPRD_DAT, se leerá un "0" (valor incorrecto).

Llamada de DPWR_DAT en el emisor (CPU 315-2 DP)

```
AWL
CALL  "DPWR_DAT"
       LADDR  :=W#16#64                //Dirección inicial Q 100
       RECORD :=P#M 10.0 BYTE 16       //Rango de origen para datos de
usuario
```

```
AWL
       RET_VAL:=MW100                    //Valor de retorno
```

Llamada de DPRD_DAT en el receptor (CPU 317-2 DP)

```
AWL
CALL  "DPRD_DAT"
       LADDR  :=W#16#78                  //Dirección inicial I 120
       RET_VAL:=MW100                    //Valor de retorno
       RECORD :=P#M 10.0 BYTE 2          //Rango de destino para datos de
usuario
```

Utilizar archivos GSD

Revisiones GSD

Información importante sobre las revisiones GSD

Los archivos GSD ponen a disposición las propiedades de los esclavos DP para las herramientas de configuración.

Las ampliaciones de funciones en el ámbito de la periferia descentralizada repercuten en la especificación GSD, por ejemplo conducen a la definición de nuevas palabras clave.

De esta forma se obtiene un versionamiento de la especificación. La versión de la especificación en la que se basa un archivo GSD recibe el nombre de "Revisión GSD" en el caso de archivos GSD.

La revisión GSD debe incluirse de forma obligatoria como palabra clave "GSD_Revision" en los archivos GSD a partir de la revisión GSD 1. Los archivos GSD sin esta palabra clave serán interpretados por las herramientas de configuración como revisión GSD "0".

Los archivos GSD pueden interpretarse hasta la revisión GSD 5. De este modo, se soportan los esclavos DP que, por ejemplo, son compatibles con las siguientes funciones:

- Avisos de diagnóstico para bloques de alarmas
- Modo isócrono y equidistancia
- SYNC/FREEZE
- Sincronización horaria para esclavos DP

Instalar archivo GSD

Introducción

En un archivo GSD (archivo de datos maestros del dispositivo) se encuentran almacenadas todas las propiedades de un esclavo DP. Si desea configurar un esclavo DP que no aparece en el catálogo de hardware, es necesario instalar el archivo GSD suministrado por el

fabricante. Los esclavos DP instalados a partir de archivos GSD se muestran en el catálogo de hardware y pueden seleccionarse y configurarse.

Requisitos

- El editor de hardware y redes está cerrado.
- Tiene acceso a los archivos GSD necesarios en un directorio del disco duro.

Procedimiento

Para instalar un archivo GSD, proceda del siguiente modo:

1. Elija el comando "Instalar archivos de descripción del dispositivo" del menú "Herramientas".
2. Seleccione en el cuadro de diálogo "Instalar archivos de descripción del dispositivo" el directorio en el que se encuentran los archivos GSD.
3. Elija uno o varios de los archivos GSD que aparecen en la lista.
4. Haga clic en el botón "Instalar".
5. Para crear un archivo de informe de la instalación, haga clic en el botón "Guardar archivo de informe".
   Los posibles problemas que puedan aparecer durante la instalación pueden repasarse con el archivo de informe.

El nuevo esclavo DP instalado desde el archivo GSP se encuentra en una nueva carpeta del catálogo de hardware.

Nota

La instalación de un archivo GSD no puede deshacerse.

Consulte también

Resumen del editor de hardware y redes (Página 489)

Configurar un esclavo DP basado en GSD

Los esclavos DP que se insertan con la instalación de un archivo GSD pueden seleccionarse del modo habitual en el catálogo de hardware e insertarse en la vista de redes. Si se desean insertar los módulos de los esclavos DP basados en GSD, deben tenerse en cuenta algunas particularidades.

Requisitos

- Se ha instalado un esclavo DP a partir de un archivo GSD.
- Se ha insertado el módulo de cabecera en la vista de redes de la forma habitual.
- La vista general de dispositivos está abierta en la vista de dispositivos.
- El catálogo de hardware está abierto.

Procedimiento

Para agregar los módulos de un esclavo DP basado en GSD, proceda del siguiente modo:

1. Navegue en el catálogo de hardware hasta los módulos del esclavo DP basado en GSD. Los esclavos DP basados en GSD, también denominados esclavos DP normalizados, se encuentran en la carpeta "Otros dispositivos de campo" del catálogo de hardware.
2. Seleccione el módulo deseado.
3. Arrastre el módulo con Drag&Drop hasta un espacio libre de la vista general de dispositivos.
4. Seleccione el módulo en la vista general de dispositivos para poder editar sus parámetros.

Se ha insertado el módulo en un slot libre del esclavo DP basado en GSD y pueden editarse sus parámetros.

Nota

En el área gráfica de la vista de dispositivos sólo se ve el esclavo DP basado en GSD. Los módulos agregados de los esclavos DP basados en GSD sólo se encuentran en la vista general de dispositivos.

Configuración teórica

En los módulos con configuración teórica ajustable, ésta puede modificarse en "Propiedades > Configuración teórica" de la ventana de inspección.

Consulte también

Objetos en la vista de dispositivos (Página 515)

Insertar un módulo en un rack (Página 523)

Agregar un dispositivo a la configuración de hardware (Página 520)

Utilizar PROFIBUS DPV1

Introducción a PROFIBUS DPV1

Generalidades

La funcionalidad DPV 1 es una ampliación de la norma PROFIBUS para la periferia descentralizada. Respecto de la funcionalidad predecesora DPV0, la funcionalidad DPV1 presenta algunas ampliaciones y simplificaciones:

- PROFIBUS DPV0 procesa el intercambio cíclico de los datos y el diagnóstico
- PROFIBUS DPV1 procesa el intercambio acíclico y cíclico de los datos, además de las alarmas.

Funcionalidad adicional en los dispositivos DPV1 (maestro DP/esclavos DP)

Los maestros y esclavos DP que soportan DPV1 presentan las siguientes funciones adicionales respecto a los dispositivos que sólo soportan DPV0:

- Se soporta el intercambio de datos acíclico entre maestro y esclavo (leer/escribir registro, p. ej. para reparametrizar un esclavo durante el funcionamiento). Los registros de un módulo y la estructura de dichos registros se encuentra en la documentación del correspondiente módulo.
- Un esclavo DPV1 puede proporcionar alarmas que garantizan un procesamiento del evento que ha provocado la alarma en la CPU maestra. También en el estado operativo "STOP" se evalúan los datos de alarma en la CPU (actualización del búfer de diagnóstico y del estado del módulo). Sin embargo, no se produce ningún procesamiento del OB en STOP.
  Además de las alarmas conocidas en el SIMATIC, p. ej. la alarma de diagnóstico en ET 200M, ahora también se soportan las alarmas de estado, de actualización y las específicas del fabricante.

Consulte también

Integración S7 de esclavos DPV1 (http://support.automation.siemens.com/WW/view/es/10259221)

Configurar dispositivos DPV1

Algunos dispositivos, p. ej. la interfaz DP IM 151-1 HF, pueden conmutarse entre DPV0 y DPV1.

Requisitos

- Se encuentra en la vista de redes.
- Hay un maestro DP con funcionalidad DPV1.
- Se establece una conexión maestro-esclavo con PROFIBUS.

Procedimiento

Para cambiar el modo de operación del esclavo DP de funcionalidad DPV1 a DPV0, proceda del siguiente modo:

1. Seleccione el esclavo DP.
2. Elija entre los modos de operación "DPV0" y "DPV1" de la lista desplegable "Modo alarma DP" en "Propiedades > General > Parámetros del módulo > Parámetros del módulo" de la ventana de inspección.

o

1. Seleccione el maestro DP.
2. Seleccione en la tabla de comunicación E/S la fila con la conexión entre el maestro DP y el esclavo DP deseado.
3. Elija entre los modos de operación "DPV0" y "DPV1" de la lista desplegable "Modo alarma DP" en "Propiedades > General > Parámetros del módulo > Parámetros del módulo" de la ventana de inspección.

Utilización de dispositivos con funcionalidad PROFIBUS distinta

En caso de utilización mixta de dispositivos con funcionalidad DPV0 y DPV1, rigen las siguientes reglas:

- En una interfaz DP con funcionalidad DPV1 en principio también pueden utilizarse esclavos DPV0.
- En una interfaz DP con funcionalidad DPV0 en principio también pueden utilizarse esclavos DPV1. La funcionalidad DPV1 se desconectará automáticamente. Sin embargo, las reglas de configuración específicas del fabricante en determinados esclavos DP pueden requerir obligatoriamente un modo DPV1, de manera que estos esclavos DP no podrán utilizarse con el sistema de maestro DP.

Asignación de la dirección de diagnóstico

En los esclavos DPV1, la dirección de diagnóstico se asigna automáticamente al slot virtual 0 como representante de la estación.

Por lo general, rige la siguiente asignación:

- Al slot virtual 0 con su dirección de diagnóstico se asignan diagnósticos y alarmas que sólo hacen referencia al esclavo DP completo: p. ej. alarmas de módulos en slots no configurados, fallo de la estación / retorno de la estación (OB 86 (Página 1864)).
- A los demás slots con su correspondiente dirección inicial se asignan diagnósticos y alarmas que parten de este módulo (p. ej. de una interfaz DP IM 153-2 en el slot 2).

Consulte también

Alarmas en el esclavo I (Página 1418)

Programación de dispositivos DPV1

Para la funcionalidad DPV1 hay OBs y SFBs/SFCs que no pueden utilizarse para DPV0.

OBs de alarma para eventos DPV1

Los esclavos DPV1 pueden activar alarmas. Para las alarmas de diagnóstico, de proceso y de extracción/inserción pueden utilizarse los correspondientes OBs, que proporciona el sistema operativo de las CPUs S7.

En la siguiente tabla se representan los OBs de alarma para DPV1. Para información más detallada al respecto, consulte la descripción de los OBs.

Alarma DPV1	OB	Explicación
Alarma de estado	OB 55 (Página 1832)	La alarma de estado se activa al cambiar el estado operativo de un dispositivo o de un módulo, p. ej. de RUN a STOP. En la documentación del fabricante del esclavo DPV1 encontrará una descripción exacta de los eventos en los cuales se activa una alarma de estado.
Alarma de actualización	OB 56 (Página 1836)	Una alarma de actualización se activa cuando se reparametriza un slot. Un acceso local o un acceso del interlocutor a los parámetros, por ejemplo, puede provocar esta situación. En la documentación del fabricante del esclavo DPV1 encontrará una descripción exacta de los eventos en los cuales se activa una alarma de actualización.
Alarma específica del fabricante	OB 57 (Página 1840)	El fabricante de un esclavo DPV1 puede definir el evento con el cual se activa la alarma específica del fabricante.

Instrucciones para accesos a esclavos DPV1

En la siguiente tabla se representan las interfaces DPV1 con sus funciones. Encontrará información más detallada al respecto en la descripción de las instrucciones avanzadas.

Función	Interfaz (DPV1)	Observaciones
Leer registro	RDREC (Página 6735)	-
Escribir registro	WRREC (Página 6738)	-
Recibir alarma de un esclavo DP	RALRM (Página 6745)	El SFB debe llamarse en el OB que origina la alarma.

Nota

Si se utiliza un esclavo DPV1 y la interfaz DP del maestro DP está ajustada a "DPV0", no está permitido que los módulos E/S lean o escriban registros con las instrucciones "WR_REC"/"RD_REC" o "WRREC"/"RDREC" en el programa de usuario. En este caso, el maestro DP direcciona el slot incorrecto (slot configurado+3).

Solución: cambiar la interfaz del maestro DP a "DPV1".

Lista de comprobación para el cambio de DPV0 a DPV1

Deben tenerse en cuenta las siguientes secciones del programa de usuario existente al editarse la configuración y al cambiar la interfaz de "DPV0" a "DPV1":

Función	¿Qué hay que tener en cuenta?
Conversión de direcciones	Si en el programa de usuario se han utilizado conversiones de direcciones, p. ej. con las instrucciones avanzadas de las funciones de sistema "GADR_LGC (Página 7073)" (SFC 5), "LGC_GADR (Página 7075)" (SFC 49), "RD_LGADR (Página 7072)" (SFC 50), entonces deberá comprobarse la asignación slot <-> dirección inicial lógica para los esclavos DPV1. Además, el slot 0 tiene una dirección. Esclavo DPV1: Mientras que el primer módulo E/S del esclavo DP de las instrucciones avanzadas estaba asignado hasta ahora al slot 4, ahora el primer módulo E/S está asignado al slot 1 (como puede verse en la configuración de hardware). Esclavo DPV1 integrado en el catálogo de hardware (p. ej. ET 200 M): El módulo de interfaz (slot 2) tiene una dirección propia.
Leer diagnóstico con la función de sistema "DPNRM_DG (Página 6803)"	La dirección de diagnóstico asignada inicialmente sigue funcionando. Internamente se asigna el slot 0 a esta dirección. El registro de diagnóstico del esclavo DPV1 tiene una estructura distinta que el DPV0 (véase descripción de los esclavos DP, en ET 200M p. ej. también bajo la palabra clave "diagnóstico avanzado").
Leer/escribir registros	Si se transfieren registros a un esclavo DPV1 con la instrucción avanzada "WR_REC (Página 6774)" (SFC 58) o se leen registros de un esclavo DPV1 con la instrucción avanzada "RD_REC (Página 6769)" (SFC 59) y estos registros trabajan en modo DPV1, el maestro DP evalúa la información de error obtenida del esclavo de la siguiente manera: Si la información de error se encuentra dentro de los rangos W#16#8000 a W#16#80FF o W#16#F000 a W#16#FFFF, el maestro DP transmite la información de error a la SFC. Si se encuentra fuera de dichos rangos, transmite el valor W#16#80A2 a la SFC y suspende el esclavo. Consulte la descripción de las informaciones de error procedentes de los esclavos DPV1 en la instrucción "RALRM" (SFB 54). Encontrará más información en las descripciones de los saltos en las descripciones de idiomas, las ayudas de los bloques y los atributos del sistema.
Leer lista de estado del sistema	Si se utiliza la instrucción avanzada "RDSYSST (Página 6963)" para p. ej. leer la información de estado del módulo o la información de estado del rack / estación, también debe tenerse en cuenta el significado modificado de los slots, así como el slot adicional 0 (véase arriba).

Ejemplo de evaluación de informaciones de alarma

Introducción

Un módulo de entrada digital S7 con disposición descentralizada (dirección inicial 288) activa una alarma de proceso. En el OB 40 (Página 1830) debe leerse la información adicional de alarma de este módulo llamando la instrucción avanzada "RALRM (Página 6745)". Se comprueba si el primer canal ha activado una alarma de proceso.

En los módulos S7, las informaciones adicionales de alarmas también pueden leerse directamente de la información inicial del OB 40. La funcionalidad DPV1 permite hasta 59 bytes de información adicional de alarma. Este volumen, sin embargo, es excesivo para la información inicial del OB 40.

Evaluación de la información de alarma del OB 40 con "RALRM"

AWL

```
// ...
// ...
//Interruptor para dirección que activa alarma (288)
    L    DW#16#120
    T    "MD10"
    CALL  "RALRM" , "DB54"
      MODE  :=1                 //Modo de función: 1=activar todos los parámetros de salida
(es decir, F_ID sin efecto)
      F_ID  :="MD10"              //Dirección inicial del slot que autoriza una alarma
      MLEN  :=8                   //Longitud máx. de la información adicional de alarma en
bytes (p. ej. para estado de canal del módulo)
      NEW   :="Alarma_nueva"   //¿Alarma recibida? (sí = 1)
      STATUS:="DP_RALRM_STATUS"   //Valor de retorno con resultado de función/mensaje de
error
      ID    :="Alarma_dirección_slot" //Dirección inicial del slot del cual se ha recibido
una alarma
      LEN   :="Longitud_información_alarma"  //Longitud de la información adicional de
alarma (4 bytes de información de cabecera + p. ej. 4 bytes en módulos E/S de S7)
     TINFO :=P#M 100.0 BYTE 28       //Puntero para la información de inicio OB+información
de gestión: 28 bytes a partir de 100 MB
      AINFO :=P#M 130.0 BYTE 8    //Puntero para rango de destino de la información de
cabecera + información adicional de alarma (máx. 59 bytes)
    U    M    124.0               //¿Es la entrada 1 (bit 0) la que ha activado la alarma?
    SPB  Alrm
    BEA
Alrm:    S    A   0.0             // Procesamiento de alarmas
// ...
```

Ejemplo de evaluación de datos de diagnóstico

Introducción

El rango de destino para datos de diagnóstico debe ser suficiente para el diagnóstico estándar (6 bytes), para el diagnóstico específico de la identificación (3 bytes para 12 slots), así como

para la evaluación del diagnóstico específico del dispositivo (sólo estado del módulo, se trata de 7 bytes más). Para ello, se utiliza la instrucción avanzada "RALRM (Página 6745)" en el OB 82 (Página 1851).

Para la evaluación continuada (diagnóstico específico del canal) deberían reservarse bytes adicionales, siempre y cuando el esclavo DP soporte esta función.

Evaluación de datos de diagnóstico en el OB 82 con "RALRM" (SFB 54)

AWL

```
// ...
// ...
      L     120                        //Definir dirección inicial para dispositivo/módulo,
      T     "Diag_dirección_slot"        //del cual debe obtenerse el diagnóstico


      CALL  "RALRM" , "DB54"
       MODO  :="Todos_params"        // 1 = Se activan todos los parámetros de salida
      F_ID  :="Diag_dirección_slot"    //Dirección inicial del slot del cual debe obtenerse
el diagnóstico
       MLEN  :=20                     //Longitud máx. de los datos de diagnóstico en bytes
       NEW   :="nuevo"                //irrelevante
       STATUS:="RET_VAL"                //Resultado de función, mensaje de error
       ID    :="Alarma_dirección_slot" //Dirección inicial del slot del cual se ha recibido
una alarma
       LEN   :="Longitud_info_alarma"  //Longitud de la información adicional de alarma (4
bytes de información de cabecera + 16 bytes de datos de diagnóstico)
       TINFO :=P#M 100.0 BYTE 28        //Puntero para la información de inicio OB
+información de gestión: 28 bytes a partir de 100 MB
       AINFO :=P#M 130.0 BYTE 20        //Puntero hacia el rango de destino en el que deben
depositarse los datos de diagnóstico

// ...
//Estructura de los datos de diagnóstico depositados:
//   MB 130 a MB 133: información de cabecera (longitud, identificador, slot)
//   MB 134 a MB 139: diagnóstico estándar (6 bytes)
//   MB 140 a MB 142: diagnóstico específico de la identificación (3 bytes)
//   MB 143 a MB 149: estado del módulo (7 bytes)
// ...
      U     M    141.0                  //¿El slot 1 tiene un error?
      SPB   stp1
      BE

stp1: L     MB   147                    //Obtener estado del módulo slot 1 a 4
      UW    W#16#3                     //Filtrar slot 1
      L     W#16#2                     //Estado 2 bits 'wrong module' módulo insertado
incorrecto
      ==I
      S     A      0.1                 //Reacción a módulo incorrecto

      L     MB   147                    //Obtener estado del módulo slot 1 a 4
      UW    W#16#3                     //Filtrar slot 1
      L     W#16#1                     //Estado 2 bits 'invalid data' datos de usuario no
válidos
      ==I
```

```
AWL
      S     A      0.2                  //Reacción a datos de usuario no válidos
// ...
```

Modelo de slot de los esclavos DPV1 en los esclavos I

Introducción

A continuación, se trata de hacer que la asignación de las direcciones E/S y direcciones de diagnóstico a los slots en DPV1 sea visible. Deben observarse en particular las direcciones que no transportan datos de usuario, así como su configuración.

El modelo de slot en DPV1

En DPV1, un esclavo DP está estructurado en slots, exactamente igual que en DPV0. Los números de los slots son 0, 1, ... n. El slot 0 sólo es válido en DPV1 y tiene un significado destacado, porque es el representante del esclavo DP completo.

Ser representante significa que las alarmas activadas por el slot 0, por ejemplo, proceden de todo el esclavo DP y no de un slot concreto dentro del esclavo DP. Los diagnósticos procedentes de este slot están asignados al esclavo DP en su conjunto y no a un solo slot o módulo.

Direcciones para interfaces DP

Desde el punto de vista de la CPU, cada una de sus interfaces tiene una dirección lógica separada. La dirección de la correspondiente interfaz DP se muestran de distintas formas:

- Al seleccionar un dispositivo o una interfaz DP en la ventana de inspección dentro de "Propiedades > Dirección PROFIBUS (Xn)" o "Propiedades > Interfaz MPI/DP > Dirección PROFIBUS (Xn)".
- Con la función "Mostrar direcciones" activada en la barra de herramientas de la vista de redes: Entonces, se indica para cada interfaz el nombre de la posible subred conectada y, a continuación, la dirección PROFIBUS asignada.

Estas direcciones no tienen nada que ver con el modelo de slot de los esclavos DP, sino que sirven exclusivamente para la identificación interna de la CPU, p. ej. en caso de fallo de la interfaz. Para el programa de usuario, esta dirección tiene una importancia mínima.

Slots y direcciones para datos de usuario

En general, cada fabricante de un esclavo DP puede determinar libremente qué datos proceden de qué slot.

A diferencia de los esclavos DPV1 con datos de usuario a partir del slot 1, para los esclavos DPV0 se aplica la restricción de que el primer módulo E/S siempre se encuentra en el slot 4. Los esclavos DP instalados mediante un archivo GSD pueden tener datos de usuario a partir del slot 1. Puesto que los datos de la periferia descentralizada generalmente se activan mediante sus direcciones, como los datos de la periferia centralizada, los datos de usuario del esclavo DPV0 deben direccionarse a partir de la dirección inicial del slot 4. Esto también rige para los esclavos DP inteligentes.

En los esclavos DP inteligentes, las áreas de memoria E/S del esclavo I pueden asignarse a las áreas de memoria E/S del maestro DP a través del área de transferencia de la "Comunicación del esclavo I". Durante el funcionamiento (intercambio cíclico de datos), los datos transferidos en el programa de usuario del esclavo DP inteligente a su área de memoria E/S son transferidos seguidamente al área de memoria E/S del maestro DP a través del área de transferencia. La tabla para la asignación de áreas de memoria E/S en áreas de transferencia se encuentra, una vez seleccionada la conexión del esclavo I en la tabla de comunicación E/S de la ventana de inspección, en "Propiedades > Comunicación de esclavo I". Tras crear un área de transferencia, ésta se puede configurar en "Detalles del área de transferencia".

Al configurar las direcciones en "Detalles del área de transferencia" también se muestran los números de slot. Los slots no están ocupados por módulos reales, como p. ej. en el caso del ET 200M, sino por longitudes discrecionales de las áreas de E/S correspondientes. En este caso, también se habla de slots virtuales.

Para comprender las asignaciones de direcciones en el área de transferencia es importante tener en cuenta lo siguiente:

- El esclavo DP inteligente tiene, además de sus slots reales, slots virtuales que se encuentran en el área de memoria E/S.
- Los slots virtuales son activados, al igual que los slots reales, por direcciones lógicas. En el caso de un esclavo DP, como, p. ej., el ET 200M, por la dirección inicial de un módulo; en el caso de un esclavo I, por la dirección configurada de la "Comunicación de esclavo I".
- Las direcciones de los slots virtuales son diferentes desde el punto de vista del maestro DP que desde el punto de vista del esclavo DP. La asignación puede configurarse. Por este motivo, el mismo slot de esclavo DP generalmente es activado por el maestro DP con una dirección distinta a la del esclavo DP.

Ejemplo de asignación de direcciones para datos de usuario

La tabla siguiente muestra la asignación de los slots virtuales en el área de transferencia:

Dirección de ejemplo desde el punto de vista del esclavo DP	Significado para el esclavo DP	Slot	Significado para el maestro DP	Dirección de ejemplo desde el punto de vista del maestro DP
		0		
		1		
		2		
		3		
I 2	Mediante el byte de entrada 2 se lee lo ...	4	... que el maestro ha escrito en el byte de salida 4.	Q 4
Q 5	Lo que se ha escrito en el byte de salida 5 del esclavo ...	5	... puede leerse en el maestro como byte de entrada 6	I 6
I 8	...	6	...	Q 8
		...		
		35		

Nota

Con la vista de redes abierta, la asignación de slots con la conexión del esclavo I seleccionada en la tabla de comunicación E/S se encuentra en "Comunicación de esclavo I > Áreas de transferencia" de la ventana de inspección.

Direcciones para información del sistema

Las direcciones para informaciones de sistema se utilizan p. ej. para poder manejar informaciones de diagnóstico o informaciones sobre transiciones de estado operativo.

Direcciones del esclavo DP

Las informaciones de sistema de un esclavo DP están asignadas a direcciones que se pueden modificar. En este sentido, las direcciones siguientes resultan relevantes para DPV1:

- Direcciones para diagnóstico de dispositivo:
  - El maestro DP diagnostica un fallo o el retorno del esclavo I.
  - El esclavo I diagnostica un fallo o el retorno del maestro DP.
- Direcciones para transiciones de estado operativo:
  - El maestro DP puede determinar un cambio de estado operativo del esclavo DP.
  - El esclavo DP inteligente puede determinar un cambio de estado operativo del maestro DP.

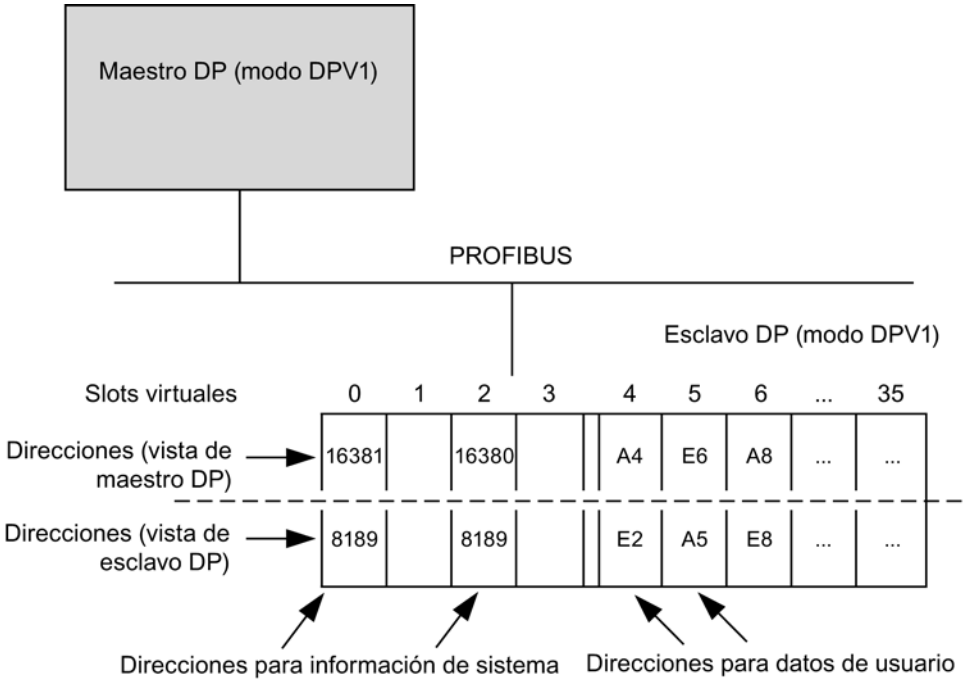
Las direcciones se adjudican automáticamente de arriba a abajo para que no se produzca ningún conflicto con datos de usuario. Las direcciones propuestas deben aceptarse, aunque pueden modificarse. Si el programa de usuario debe funcionar en distintas CPUs, es necesario comprobar si las direcciones también se encuentran dentro del área de direcciones de la CPU "más pequeña".

La tabla siguiente muestra la asignación de las direcciones en la comunicación del esclavo I:

Dirección de ejemplo desde el punto de vista del esclavo DP	Significado para el esclavo DP	Dirección de ejemplo desde el punto de vista del maestro DP	Significado para el maestro DP
8189	Fallo de la estación / retorno de la estación del maestro DP	16381	Fallo de la estación / retorno de la estación del esclavo DP
8188	Transición de estado operativo del maestro DP	16380	Transición de estado operativo del esclavo DP

Resumen

La configuración del esclavo I con slots virtuales de las áreas de transferencia es la siguiente:



Activar alarma de proceso con instrucción DP_PRAL

Con la función de sistema en la instrucción avanzada "DP_PRAL (Página 6794)" sepuede disparar una alarma de proceso para cada dirección configurada desde el programa de usuario de la CPU del I-Slave. Esto es válido tanto para direcciones de datos de usuario del área de E/S como para direcciones del slot virtual 2.

En el programa de usuario del I-Slave se utilizan direcciones E/S configuradas para la instrucción "DP_PRAL", p. ej. en la columna "local ...".

A continuación, se activará una alarma de proceso en el programa de usuario del maestro. Como dirección que activa la alarma se indica, en la información inicial del OB de alarma de proceso (p. ej. OB 40 (Página 1830)), la dirección que se ha configurado en la columna "Interlocutor PROFIBUS DP".

Equidistancia y sincronismo

Información importante sobre la equidistancia y el sincronismo

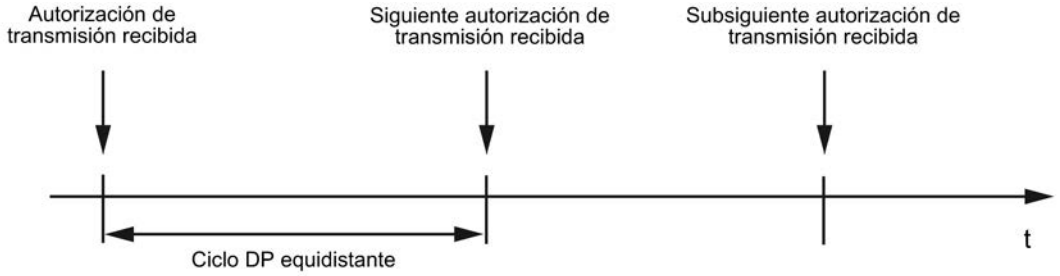
Ciclos de bus de la misma longitud en subredes PROFIBUS

Introducción

Las propiedades de PROFIBUS "sincronismo" y "equidistancia" forman la base de los ciclos de procesamiento sincronizados.

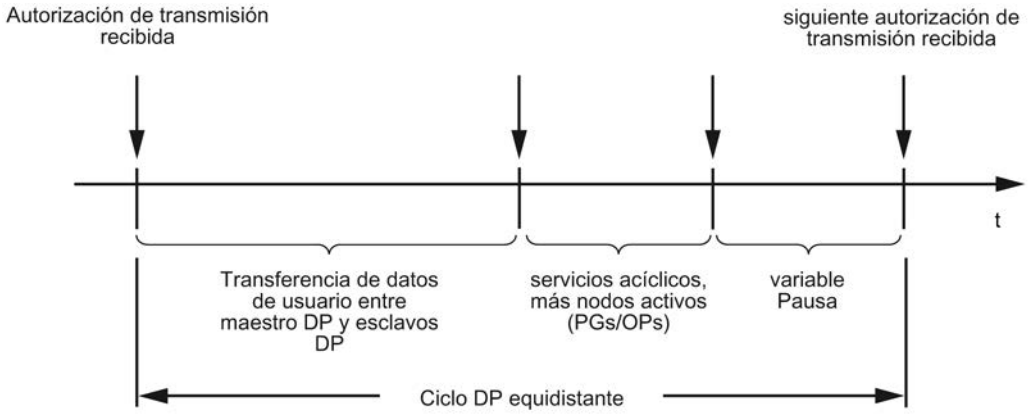
La equidistancia garantiza un intervalo exactamente de la misma longitud para ciclos de bus. "Ciclos de bus de la misma longitud" significa que el maestro PROFIBUS DP inicia el ciclo de bus DP una y otra vez tras el mismo intervalo de tiempo. Por este motivo, los esclavos conectados también obtienen sus datos del maestro en intervalos de tiempo exactamente iguales. En este caso, también se habla de "Sincronización del ciclo de bus".

La equidistancia es el requisito previo para el sincronismo.



Composición del tiempo de ciclo de bus

La figura siguiente muestra la composición de tiempo para un ciclo de bus.



La "pausa variable" que se muestra en la imagen siempre es mínima, si todavía quedan peticiones de comunicación pendientes p. ej. para otros nodos activos. El maestro, y más concretamente el maestro de equidistancia, controla los porcentajes de comunicación de manera que siempre se alcanza la misma duración para un ciclo de bus.

Requisitos

Deben haberse cumplido los siguientes requisitos para la equidistancia:

- El maestro DP debe soportar la función "Equidistancia".
- El maestro de equidistancia debe ser un maestro DP de clase 1. Una PG/un PC no es adecuado como maestro de equidistancia.
- El maestro de equidistancia es la única estación activa en PROFIBUS DP. Sólo puede haber un sistema de maestro DP en la subred PROFIBUS. Es posible conectarse adicionalmente PGs y/o PCs.
- No se han configurado grupos SYNC/FREEZE.
- La equidistancia sólo es posible con los perfiles de bus "DP" y "definido por el usuario".
- No puede haber ninguna CiR (Configuration in Run) configurada.
- En la subred PROFIBUS no puede haber ninguna CPU H conectada.

Tiempo para ciclo DP equidistante

El tiempo para el ciclo DP equidistante se calcula automáticamente:

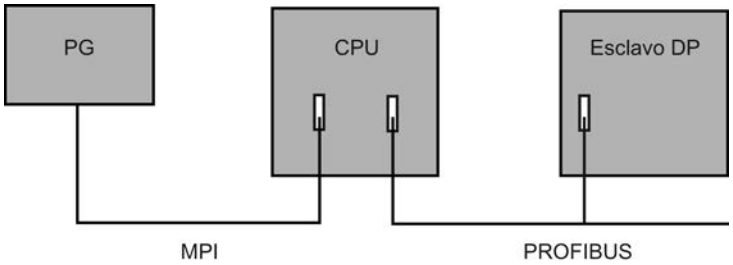
- A partir de la configuración PROFIBUS (número de nodos configurados)
- A partir de otras informaciones para el cálculo que pueden indicarse de forma opcional: p. ej. PGs que deben tenerse en cuenta pero que se encuentran fuera del proyecto.

El tiempo calculado para el ciclo DP equidistante tiene en cuenta el tráfico de datos de usuario del maestro PROFIBUS DP. Si se producen errores, es posible que se infrinja la equidistancia.

El tiempo calculado puede modificarse dentro del intervalo mostrado.

Influencia de nodos activos conectados (PGs/PCs y esclavos I)

Sólo deberá tenerse en cuenta una PG/un PC si está conectado directamente a PROFIBUS con su interfaz PROFIBUS. Si la PG está conectada mediante la interfaz MPI de la CPU, no hace falta tenerlo en cuenta, tal y como muestra la figura siguiente.



Si hay conectados esclavos DP inteligentes, p. ej. CPU 315-2 DP, el tiempo para el ciclo DP equidistante debe calcularse de forma más amplia.

Ciclos de procesamiento sincronizados

PROFIBUS permite el modo isócrono

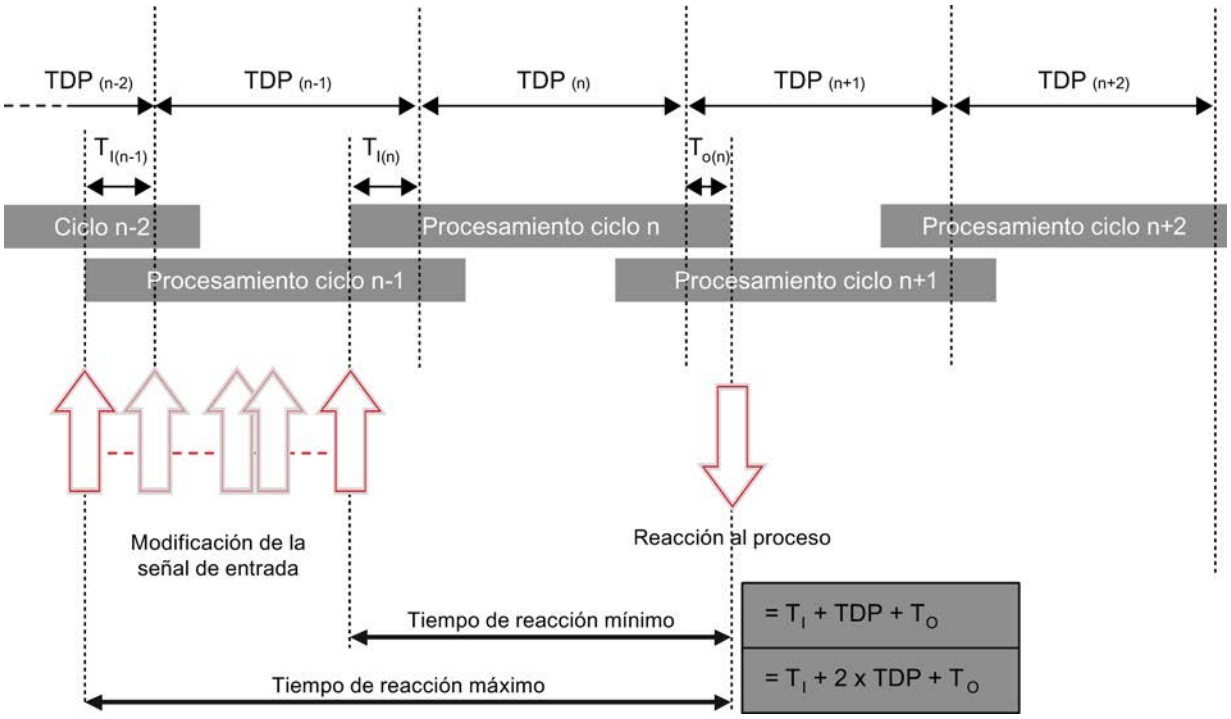
El PROFIBUS equidistante (isócrono) constituye la base de los ciclos de procesamiento sincronizados. Pone a disposición una sincronización básica. Con la propiedad de sistema "Modo isócrono" se acopla una solución de automatización al PROFIBUS equidistante.

Esto significa:

- La lectura de los datos de entrada se sincroniza con el ciclo DP. Todos los datos de entrada se leen en el mismo momento.
- El programa de usuario para el procesamiento de datos periféricos se sincroniza mediante los OBs de alarma de sincronismo OB 61 a OB 64 con el ciclo DP.
- La salida de los datos de salida se sincroniza con el ciclo DP; todos los datos de salida tienen validez en el mismo momento.
- Todos los datos de entrada y salida se transmiten de forma coherente. Esto significa que todos los datos de la memoria imagen de proceso pertenecen al mismo grupo lógico y temporal.

Sincronización de los ciclos individuales

La siguiente figura muestra el ancho de banda de posibles tiempos de reacción que surgen de la sincronización de la lectura de los datos de entrada para un ciclo de procesamiento a partir de una serie de ciclos individuales isócronos.



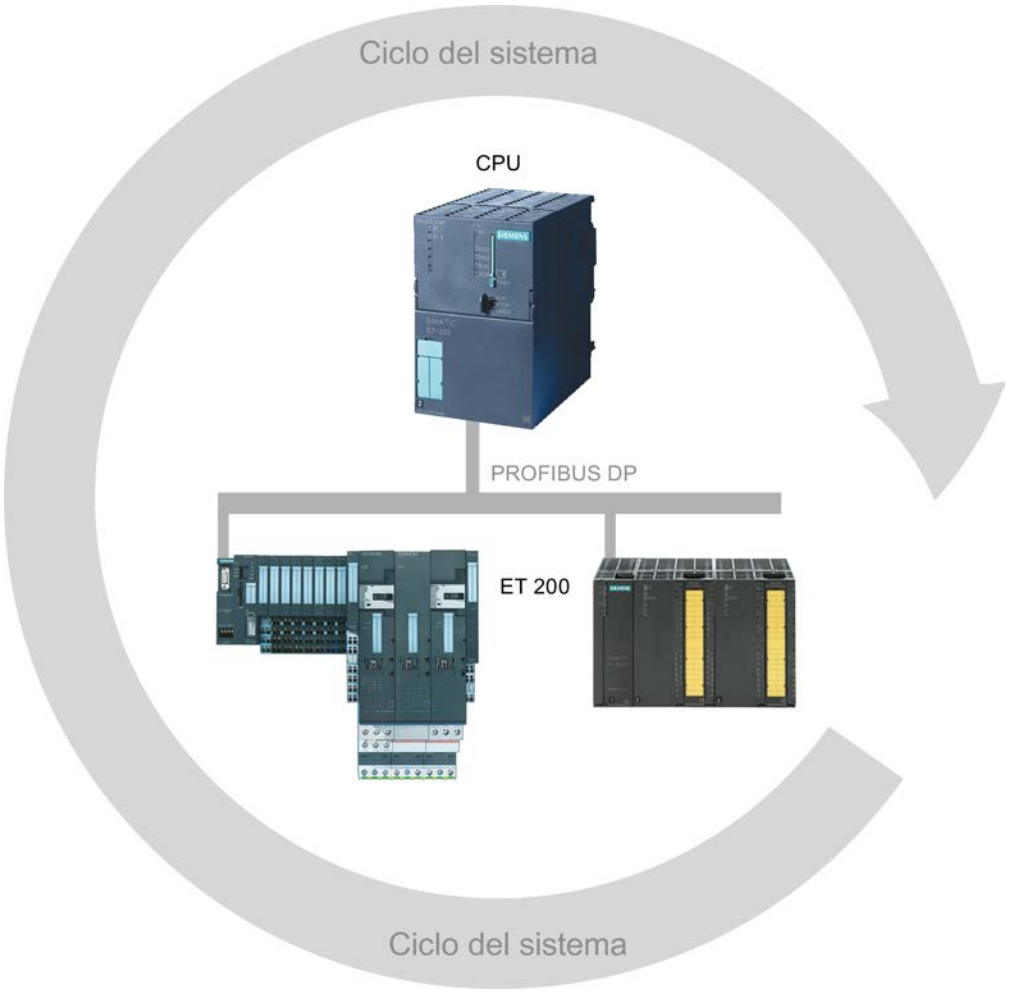
TDP	Reloj del sistema
Ti	Momento de la lectura de los datos de entrada
To	Momento de emisión de los datos de salida

Con la sincronización de los ciclos individuales es posible leer los datos de entrada en el ciclo "n-1", transmitir y procesar los datos en el ciclo "n" y transmitir los datos de salida calculados y conmutarlos a los "Terminales" al principio del ciclo "n+1". De ello resulta un tiempo de reacción de proceso real de "Ti +TDP +To" a "Ti + 2xTDP + To".

Con la propiedad de sistema "Modo isócrono", los tiempos de ejecución del sistema SIMATIC son constantes y SIMATIC es estrictamente determinista a través del bus.

Justo a tiempo

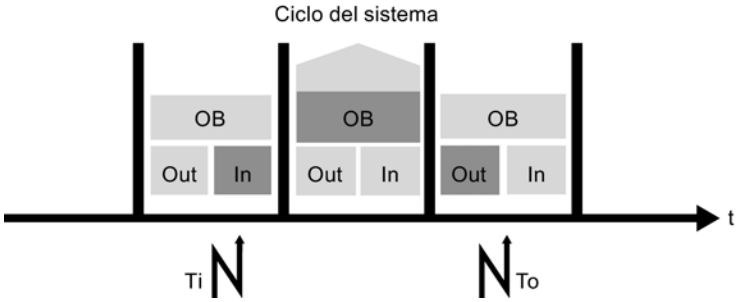
El tiempo de reacción rápido y fiable de un funcionamiento isócrono se basa en el hecho de que todos los datos se ponen a disposición justo a tiempo. Para ello, el ciclo DP equidistante (isócrono) marca el tiempo de ciclo.



Tiempo de preprocesamiento Ti

Para que todos los datos de entrada se encuentren disponibles para el transporte a través de la línea DP la próxima vez que empiece el ciclo DP, el ciclo de lectura de la periferia se inicia

previamente en un tiempo de preprocesamiento Ti. El tiempo de preprocesamiento Ti es configurado por el usuario o definido automáticamente.



Procesamiento de los datos de entrada

PROFIBUS transporta los datos de entrada hacia el maestro DP a través de la línea DP. El OB de alarma de sincronismo (OB 61, OB 62, OB 63 u OB 64) se llama. El programa de usuario en el OB de alarma de sincronismo determina la reacción del proceso y prepara los datos de salida a tiempo hasta el comienzo del próximo ciclo DP. Existe la posibilidad de hacer calcular automáticamente la longitud del ciclo DP o bien de configurarlo uno mismo.

Preparación de los datos de salida

Para los datos de salida rige lo siguiente:

- Preparación a tiempo al principio del siguiente ciclo DP
- Transferencia a los esclavos DP a través de línea DP
- Transmisión al proceso de modo isócrono, es decir, simultáneamente en el tiempo To

De ello se deriva el total "Ti + 2xTDP + To" como tiempo de reacción reproducible del borne de entrada al borne de salida.

Características del modo isócrono

El modo isócrono tiene tres características fundamentales:

- El programa de usuario está sincronizado con el procesamiento de la periferia, es decir, todos los procesos están sintonizados temporalmente entre sí. Todos los datos de entrada se registran en un momento definido. Asimismo, los datos de salida se hacen efectivos en un momento definido. Los datos de entrada y salida están sincronizados con el ciclo del sistema hasta el borne.
  Los datos de un tiempo de ciclo siempre se procesan en el siguiente tiempo de ciclo y empiezan a ser efectivos en los bornes en el ciclo posterior.
- Los datos de entrada y salida se procesan de forma equidistante (isócrona), es decir, los datos de entrada siempre se leen en los mismos intervalos de tiempo y los datos de salida siempre se emiten en los mismos intervalos de tiempo.
- Todos los datos de entrada y salida se transfieren de forma coherente, es decir, todos los datos de la memoria imagen de proceso pertenecen al mismo grupo lógico y temporal.

Ejemplo de detección isócrona de varios puntos de medición

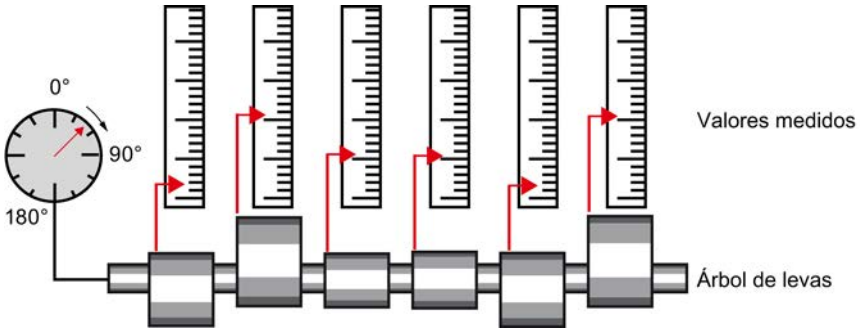
Introducción

Con la propiedad de sistema "Sincronismo" pueden detectarse valores medidos y datos de proceso en un ciclo de sistema fijo. En el mismo ciclo de sistema se produce el procesamiento de señales hasta la conmutación al terminal de salida. Así, el sincronismo contribuye a elevar la calidad del control y, con ello, a mejorar la precisión de fabricación. Con el sincronismo se reducen drásticamente las posibilidades de oscilación de los tiempos de reacción del proceso. El procesamiento asegurado temporalmente puede utilizarse para un ciclo de máquina más elevado.

En general, el sincronismo puede utilizarse allí donde deban registrarse valores medidos de modo síncrono, coordinarse movimientos y definirse y realizarse simultáneamente reacciones de procesos, como en el siguiente ejemplo.

Tarea

Dentro del proceso de producción de árboles de levas, éstos deben medirse de forma precisa para asegurar la calidad.



Proceso de trabajo convencional

1. Girar árbol de levas
2. Detener
3. Medir la posición angular
4. Medir todas las desviaciones de las levas
5. Continuar girando, detener, medir...

Proceso de trabajo con sincronismo

Con la utilización de la propiedad de sistema "Sincronismo" y la simultaneidad relacionada del registro de valores medidos, el proceso de medición puede realizarse de forma continua, con lo que se reduce el tiempo necesario para ello. Esto lleva al siguiente proceso de trabajo:

1. Girar el árbol de levas de forma continua
2. Durante el giro continuo, medir las posiciones y las desviaciones de las levas de modo síncrono
3. Procesar el siguiente árbol de levas

En caso de un único giro del árbol de levas, se miden de modo síncrono todas las posiciones del árbol de levas y los correspondientes valores medidos (rojos en la imagen). El ciclo de la máquina aumenta con una precisión de medición igual o mayor.

Reglas, requisitos y condiciones marco

Lo que debe tenerse en cuenta para la configuración del modo isócrono

- El modo isócrono no puede utilizarse en redes PROFIBUS ópticas.
- La equidistancia y el modo isócrono sólo son posibles con los perfiles de bus "DP" y "Definido por el usuario".
- El modo isócrono sólo es posible con las interfaces DP integradas en la CPU. Para PROFIBUS, no es posible el modo isócrono con CPs.
- En el PROFIBUS DP isócrono sólo se permite el maestro de equidistancia como estación activa. Los OPs y las programadoras (o PCs con funcionalidad de programadora) influyen en el comportamiento del tiempo de ciclo DP equidistante, por lo que no se recomiendan.
- No es posible el funcionamiento isócrono de varias líneas.
- Las direcciones de la periferia isócrona deben estar en el área de direcciones de las memorias imagen parciales de proceso, ya que la periferia isócrona sólo puede procesarse en memorias imagen parciales de proceso. Si no se utilizan memorias imagen parcial de proceso, no es posible una transmisión de datos coherente e isócrona. Se supervisa el cumplimiento de los alcances, ya que el número de esclavos y de bytes en el sistema maestro DP están limitados en cada memoria imagen parcial de proceso.
- En la misma memoria imagen parcial de proceso puede haber periferia no isócrona junto con la periferia isócrona. Debe haber un módulo/submódulo isócrono en la memoria imagen parcial de proceso.
- Sólo es posible un funcionamiento isócrono total de "terminal" a "terminal" si todos los componentes de una cadena soportan la propiedad de sistema "Modo isócrono". Al seleccionar el módulo en el catálogo asegúrese de que el campo de información del módulo contenga el término "modo isócrono".
- En caso de configurar el modo isócrono, no está permitido asignar ningún grupo SYNC/FREEZE al esclavo.

Consulte también

Asignar esclavo DP a grupo SYNC/FREEZE (Página 1402)

Manual de funcionamiento isócrono (http://support.automation.siemens.com/WW/view/es/15218045)

Configurar y parametrizar el sincronismo

Optimizar el comportamiento de tiempo seleccionando la topología adecuada

Influencia de la topología DP

El comportamiento de tiempo del sincronismo puede influirse positivamente eligiendo una topología DP adecuada.

Separación de periferia isócrona y no isócrona

Es posible combinar periferia descentralizada isócrona con periferia descentralizada no isócrona en un sistema maestro DP.

Sin embargo, para un buen rendimiento se recomienda repartir la periferia isócrona y no isócrona en distintos sistemas maestros DP.

Si se utilizan TDs, OPs o PGs, éstos no deberían utilizarse en el sistema maestro DP isócrono.

Distribución adecuada de los módulos de periferia

Para mantener lo más cortos posible los tiempos Ti y To, a la hora de planificar la topología DP deberá tenerse en cuenta una distribución adecuada de los grupos de periferia en los módulos de interfaz (ET 200M, ET 200S).

Nota

Pruebe distintas configuraciones y calcule el comportamiento de tiempo. Seleccione la configuración más apropiada.

Información adicional

Encontrará más información sobre la optimización del tiempo de equidistancia en las instrucciones de servicio de las familias de esclavos DP ET 200M y ET 200S.

Parametrizar equidistancia y sincronismo

Configurar equidistancia y sincronismo

Configuración del modo isócrono

La configuración del modo isócrono consta de los siguientes componentes isócronos:

- CPU con interfaz DP integrada (p. ej. CPU 319-3 PN/DP)
- Interfaces DP (p. ej. interfaz ET 200S IM 151-1 High Feature)
- Módulos de entradas/salidas descentralizados (p. ej. DI 2x24V DC, High Feature [131-4BB00], DO 2x24V DC/2A, High Feature [132-4BB30])

Encontrará una lista actual de componentes isócronos en el Service & Support Portal (http://support.automation.siemens.com/WW/view/es/14747677).

Procedimiento

Para configurar la equidistancia y el modo isócrono, proceda del siguiente modo:

1. Configure una CPU operable en modo isócrono y ajuste las propiedades para el modo isócrono:
   - Alarma de sincronismo
   - Sistema maestro DP asignado
   - Memoria imagen parcial de proceso
   - Tiempo de retardo
2. Configure los esclavos DP isócronos con módulos isócronos y ajuste las propiedades del modo isócrono de forma centralizada en las propiedades de PROFIBUS o en cada uno de los esclavos DP (propiedades de esclavo). En este caso, se trata fundamentalmente de los valores Ti/To y de cómo se calculan.

   Nota

   Todas las propiedades de la periferia isócrona pueden ajustarse de forma centralizada en las propiedades de PROFIBUS. Al mismo tiempo, se obtiene la vista general de todos los esclavos y sus ajustes en PROFIBUS.

3. Escriba el programa de usuario con accesos a la periferia isócrona.

En los siguientes apartados encontrará una descripción detallada del procedimiento a seguir.

Consulte también

Ajustar propiedades de la CPU (Página 1454)

Ajustar propiedades de PROFIBUS (Página 1455)

Ajustar el sincronismo de los esclavos DP (Página 1457)

Crear un programa de usuario (Página 1458)

Ajustar propiedades de la CPU

Los ajustes de la CPU para alarmas de sincronismo provocan la conexión isócrona del programa de usuario a través de los bloques de organización previstos para ello (p. ej. OBs de alarma de sincronismo).

Requisitos

La CPU debe soportar el modo isócrono y las condiciones marco no deben excluirlo.

Procedimiento

Para ajustar las propiedades de la CPU, proceda del siguiente modo:

1. Seleccione la CPU en la vista de dispositivos o redes.
2. Seleccione la propiedad de la CPU "Alarmas de sincronismo".
3. Deben realizarse los siguientes ajustes para cada OB de alarma de sincronismo:
   - Sistema maestro DP utilizado
   - Memoria o memorias imagen parcial de proceso deseadas
   - Tiempo de retardo

Información adicional sobre las memorias imagen parciales de proceso

Las memorias imagen parciales de proceso asignadas deben actualizarse en el OB de alarma de sincronismo llamando las instrucciones avanzadas "SYNC_PI" (SFC 126) y "SYNC_PO" (SFC 127). La llamada de "SYNC_PI" actualiza la memoria imagen parcial de proceso de las entradas, mientras que la llamada de "SYNC_PO" actualiza la memoria imagen parcial de proceso de las salidas.

Si las direcciones de la periferia descentralizada han sido asignadas a distintas memorias imagen parciales de proceso, deberá introducir los números de las memorias imagen parciales de proceso utilizadas, separados por comas (p. ej. 1,2,3).

Información adicional sobre el tiempo de retardo

- El tiempo de retardo es el tiempo que transcurre entre el telegrama Global Control y el inicio del OB 6x. Durante este tiempo, el maestro DP ejecuta el intercambio de datos cíclico con los esclavos DP.

  Nota

  Recomendación: al lado del campo para introducir el tiempo de retardo se encuentra la casilla de verificación "Automático mínimo". Si esta casilla de verificación está activada, no es necesario volver a ajustar el tiempo de retardo después de efectuar cambios en la configuración.

- El tiempo de retardo sólo puede ajustarse si la casilla de verificación "Automático mínimo" está desactivada.

  Nota

  Si el tiempo de retardo es 0, el OB 6x se inicia directamente con el telegrama Global-Control. En este caso es muy probable que un acceso a la periferia isócrona por medio de la instrucción avanzada "SYNC PI" (SFC 26) resulte fallido, ya que los datos de la periferia aún no están disponibles en el momento de inicio del OB.

  Si se eligiera un tiempo de retardo demasiado grande, el acceso a la periferia isócrona sería posible antes. Un tiempo de retardo demasiado grande provoca un tiempo de ejecución de OB innecesariamente elevado.

Consulte también

Configurar equidistancia y sincronismo (Página 1453)

Reglas, requisitos y condiciones marco (Página 1451)

Ajustar propiedades de PROFIBUS

Introducción

Para editar las propiedades del modo isócrono en PROFIBUS deben modificarse los ajustes en la subred o en el sistema maestro DP.

La equidistancia para el modo isócrono en una subred PROFIBUS equidistante se activa automáticamente en las siguientes condiciones:

- Asignación de un sistema maestro DP a un OB de alarma de sincronismo en las propiedades de la CPU para alarmas de sincronismo.
- Ajuste del modo isócrono para un esclavo DP en PROFIBUS.

Las propiedades de PROFIBUS sólo deben editarse si se desea modificar los tiempos manualmente. De lo contrario, todos los tiempos se determinarán automáticamente.

Requisitos

Subred PROFIBUS equidistante.

Procedimiento

Para modificar las propiedades del modo isócrono en PROFIBUS, proceda del siguiente modo:

1. Seleccione la subred PROFIBUS en la vista de redes y elija "Equidistancia" en "Propiedades" de la ventana de inspección.

   Nota

   Procure no seleccionar un sistema maestro resaltado. Sólo si la subred PROFIBUS está seleccionada, pueden utilizarse las propiedades PROFIBUS y, con ellas, también las propiedades del modo isócrono.

2. Ajuste las propiedades para el modo isócrono. En los siguientes apartados se describe cada uno de los ajustes.

Propiedades no seleccionables para modo isócrono

Si las propiedades para el modo isócrono no pueden seleccionarse, haga la siguiente comprobación:

- Conexión de un solo maestro DP en la subred PROFIBUS. El modo isócrono no es posible si hay varios maestros DP conectados.
- Soporte del modo isócrono por parte del maestro DP (p. ej. una CPU con interfaz DP integrada).
- Posibilidad de configuración de los grupos SYNC/FREEZE. Los grupos SYNC/FREEZE y el modo isócrono no son compatibles.

Ajustar manualmente el tiempo de ciclo DP y los valores Ti/To

Para poder ajustar manualmente el tiempo de ciclo DP, desactive la casilla de verificación "Tiempo de ciclo DP automáticamente al mínimo". De este modo podrá introducir un valor mayor que el tiempo de ciclo DP mínimo calculado automáticamente.

Para poder ajustar los valores Ti y To, desactive la casilla de verificación "Valores Ti/To automáticos".

Modo isócrono y ajustes Ti/To en la vista detallada

La vista detallada permite ajustar las siguientes propiedades para cada esclavo adecuado o para cada módulo adecuado del esclavo:

- Modo isócrono
- Valores Ti/To

La vista detallada consiste en una tabla. En el menú contextual de la barra de título de la tabla puede adaptar la representación de la tabla, p. ej. mostrar los valores de TDPMin.

No se puede activar o desactivar manualmente la casilla de verificación "Modo isócrono" de una CPU (PLC). La casilla de verificación se activa automáticamente si se asigna una alarma de sincronismo al sistema maestro DP en la sección "Alarmas" de las propiedades de la CPU.

Posibilidades de ajuste para el cálculo de los valores Ti/To en la vista detallada

Para el cálculo de los valores Ti/To existen las siguientes posibilidades de ajuste:

- De la subred
  Al activar el modo isócrono, se preselecciona automáticamente la opción Valores Ti/To "De la subred". Con esta opción, todos los esclavos DP que obtienen sus valores Ti/To de la red tienen tiempos Ti/To idénticos, concretamente los de la subred PROFIBUS.
- Automático mínimo
  Si p. ej. se modifica manualmente el tiempo de ciclo DP o los valores Ti/To de otros esclavos y esta modificación repercute en los tiempos Ti/To del esclavo, automáticamente se realizarán las adaptaciones necesarias. Con esta opción, los valores Ti/To del esclavo pueden ser diferentes entre sí.
- Manuales
  Con esta opción, tiene la posibilidad de introducir manualmente los valores Ti/To para el esclavo. Con esta configuración pueden producirse errores:
  - Tiempos demasiado pequeños
  - Los valores Ti/To de otros esclavos DP ya no pueden calcularse automáticamente debido a su configuración.

Indicación "-" con tiempo Ti (To)

Si no se muestra ningún valor para Ti o To en la subred PROFIBUS (indicación "-"), se debe a la siguiente causa: Para el cálculo automático del tiempo Ti/To para la subred PROFIBUS se considera la media de todos los rangos de valores Ti/To de todos los esclavos DP. De esta media se elige el valor más pequeño. Si no es posible calcular ninguna media, se muestra "-". En este caso, un esclavo DP no puede obtener ningún valor Ti/To de la red.

Consulte también

Configurar equidistancia y sincronismo (Página 1453)

Solapamiento de Ti y To (Página 1459)

Ajustar el sincronismo de los esclavos DP (Página 1457)

Asignar esclavo DP a grupo SYNC/FREEZE (Página 1402)

Ajustar el sincronismo de los esclavos DP

Ajustes en el esclavo DP

Excepto en el ciclo DP equidistante y los tiempos Ti/To para la subred PROFIBUS, pueden realizarse los mismos ajustes en las propiedades de esclavo DP que en las propiedades de PROFIBUS.

Procedimiento

Para ajustar el modo isócrono para módulos, proceda del siguiente modo:

1. Seleccione en la vista de redes el esclavo DP cuyas propiedades desea editar.
2. Adapte los parámetros a sus exigencias en "Propiedades > Modo isócrono".

Ajustar los mismos tiempos Ti/To para todos los esclavos

El comportamiento preajustado tras la inserción de esclavos DP en la vista de redes es: Los valores Ti/To se obtienen de la red, por lo que son iguales automáticamente.

En una configuración editada es posible ajustar los tiempos Ti/To al mismo valor manualmente o bien ajustar todos los esclavos de forma que obtengan sus valores Ti/To de la red.

Consulte también

Configurar equidistancia y sincronismo (Página 1453)

Ajustar propiedades de PROFIBUS (Página 1455)

Crear un programa de usuario

Procedimiento

Para crear el programa de usuario para sincronismo, proceda del siguiente modo:

1. Cree los OBs de alarma de sincronismo necesarios (p. ej. el OB 61).
2. Llame en el OB de alarma de sincronismo las siguientes instrucciones avanzadas para actualizar la memoria imagen parcial de proceso.
   - "SYNC_PI" (SFC 126) para actualizar la memoria imagen parcial de proceso de las entradas
   - "SYNC_PO" (SFC 127) para actualizar la memoria imagen parcial de proceso de las salidas

Debe utilizarse como memoria imagen parcial de proceso la parametrizada en la CPU (rango de parámetros "Alarmas de sincronismo").

Nota

Puede producirse la siguiente situación, sobre todo en caso de tiempos de ciclo DP muy cortos: El tiempo de ejecución del programa de usuario (OB 6x con SYNC_PI/SYNC_PO llamado) es mayor que el ciclo más pequeño (véase Datos técnicos de la CPU, apartado "Sincronismo"). En este caso, debe aumentarse el tiempo de ciclo DP calculado automáticamente de forma manual.

Consulte también

Configurar equidistancia y sincronismo (Página 1453)

Solapamiento de Ti y To

Solapamiento de Ti y To

Reducción del ciclo de sistema y del tiempo de reacción del proceso

Con un solapamiento de Ti y To se obtiene una reducción del ciclo del sistema TDP y del tiempo de reacción del proceso, es decir, el tiempo que transcurre entre la aparición y el registro de un evento, el procesamiento y la salida de una reacción en las salidas será menor.

Requisitos para el solapamiento de Ti y To

Para el solapamiento de Ti y To deben cumplirse los siguientes requisitos:

ET 200M: Módulo de interfaz IM 153-2; a partir de la referencia 6ES7 153-2BAx2-0XB0

ET 200S: Módulo de interfaz IM 151-1; a partir de la referencia 6ES7 151-2BAx2-0XB0

Nota

Los módulos que disponen tanto de tiempos de conversión de entrada (TWE) como de tiempos de conversión de salida (TWA) (en algunos casos, en función del modo de operación del módulo de entrada/salida) no pueden utilizarse de modo isócrono con solapamiento de Ti y To. El solapamiento de Ti y To sólo es posible si estos módulos **no** están activados para el modo isócrono.

Consulte también

Ajustar propiedades de PROFIBUS (Página 1455)

Principio de funcionamiento del solapamiento de Ti y To (Página 1459)

Ajuste del sincronismo con solapamiento de Ti y To (Página 1461)

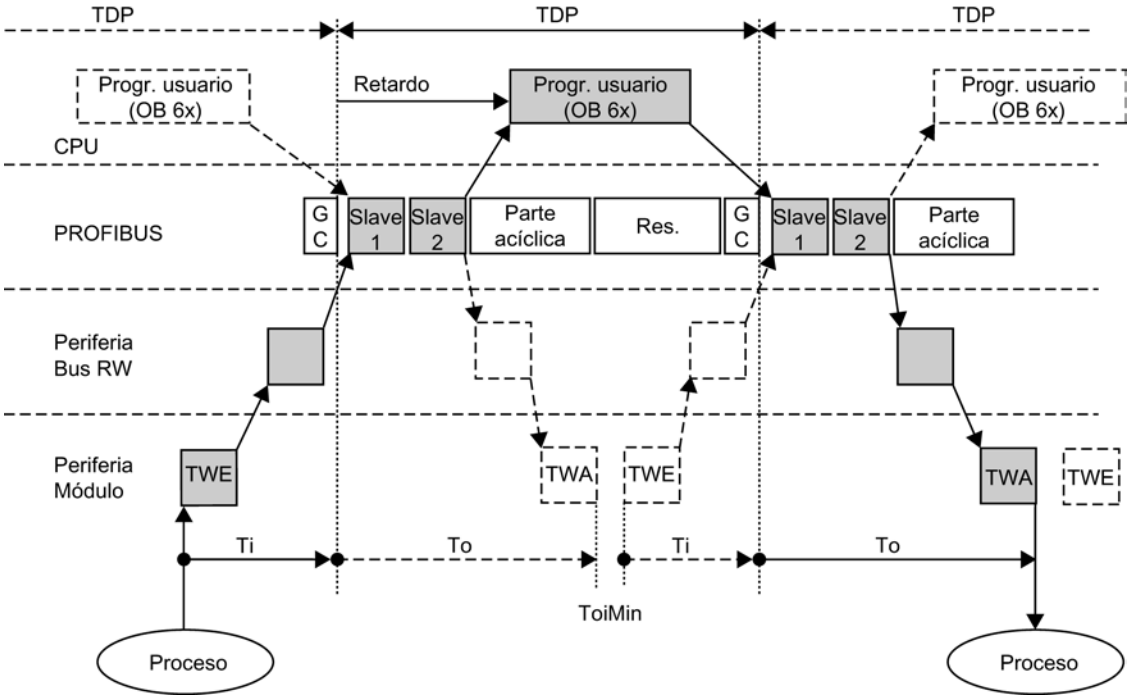
Principio de funcionamiento del solapamiento de Ti y To

Procesamiento solapado

El principio de funcionamiento en el solapamiento de Ti y To consiste en el hecho de que el módulo de entrada de periferia ya lee las entradas mientras el módulo de salida de periferia todavía conecta las reacciones de proceso del programa de usuario en las salidas. Esto lleva a una reducción del ciclo de sistema TDP y, con él, a tiempos de reacción de proceso más cortos.

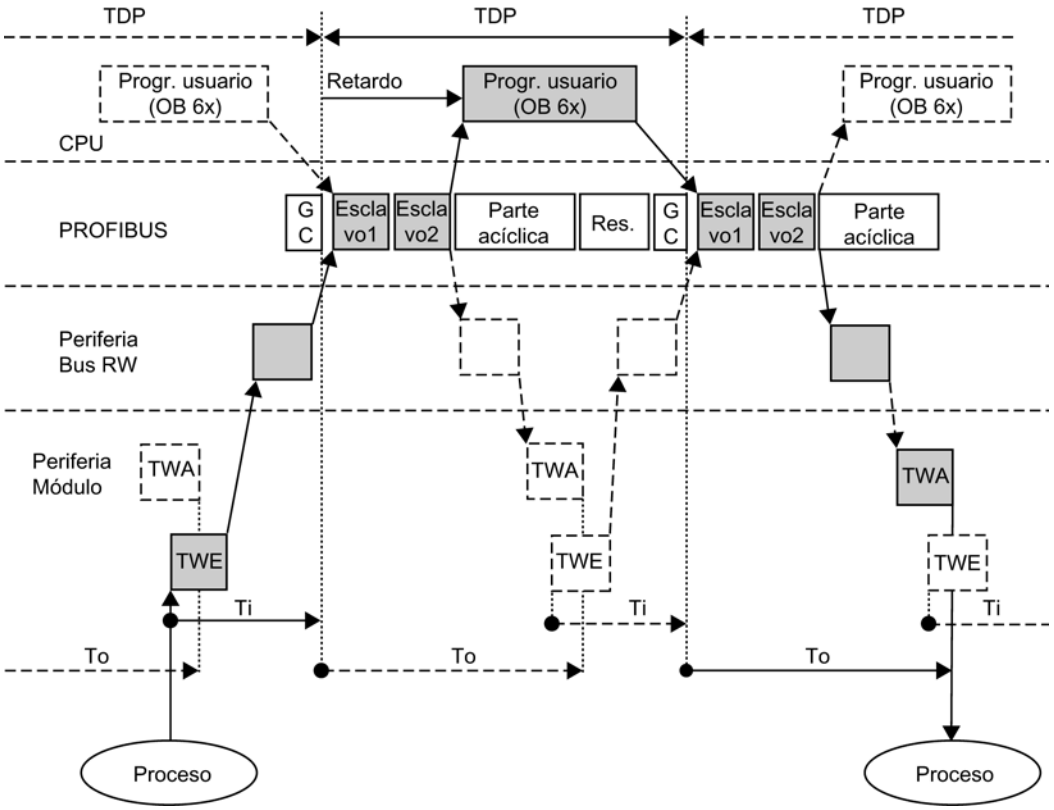
Ciclo de bus DP sin solapamiento de Ti y To

La siguiente figura representa un ciclo de bus DP equidistante con sincronismo, pero sin solapamiento de Ti y To.



Ciclo de bus DP con solapamiento de Ti y To

La siguiente figura representa un ciclo de bus DP equidistante con sincronismo y solapamiento de Ti y To.



Reglas para Ti, To y TDP

Rigen las siguientes reglas para Ti, To y TDP:

- Sin solapamiento, Ti + To es ≤ TDP, expresado de otro modo: TDP ≥ Ti + To
- Con solapamiento, Ti + To puede ser > TDP o TDP < Ti + To

Consulte también

Solapamiento de Ti y To (Página 1459)

Ajuste del sincronismo con solapamiento de Ti y To

Ajustar el modo isócrono con solapamiento de Ti y To

En el modo isócrono con solapamiento de Ti y To es necesario tener en cuenta lo siguiente:

El esclavo DP en el que es posible el solapamiento Ti/To no puede obtener sus valores Ti/To de la subred PROFIBUS. Seleccione para este esclavo DP, por ejemplo, la opción "Valores Ti/To automáticos".

Se ajustan automáticamente los mejores valores posibles para Ti y To, dependiendo del ciclo DP equidistante actualmente ajustado.

Reglas para Ti y To en caso de solapamiento

En el modo isócrono con solapamiento se aplican las siguientes reglas para Ti y To:

- Ti ≤ TDP
- To ≤ TDP

Localizar reservas

Para localizar reservas, es posible visualizar los procesos temporales con ayuda de una tabla de cálculo (http://support.automation.siemens.com/WW/view/es/23876584) y con el resultado optimizar en consecuencia los procesos en STEP 7.

Nota

Exención de responsabilidad

Siemens AG no asume ninguna responsabilidad por los daños derivados de la utilización de la tabla de cálculo, en especial por los daños personales, materiales o patrimoniales que guarden relación directa o indirecta con la utilización de la tabla de cálculo en calidad de daños consecuenciales.

No prestamos soporte para la utilización de la tabla de cálculo.

Nota

Tenga en cuenta las reglas para la optimización del tiempo de equidistancia y para el solapamiento de Ti y To.

Información adicional

Encontrará más información sobre la optimización del tiempo de equidistancia en las instrucciones de servicio del ET 200M y en las instrucciones de servicio del ET 200S.

Consulte también

Solapamiento de Ti y To (Página 1459)

8.1.4.11 Configurar PROFIBUS DP (S7-1200)

Principios básicos sobre la configuración de un sistema maestro DP

Periferia descentralizada

Se entienden por periferia descentralizada los sistemas maestros DP que constan de maestro DP y esclavos DP, están conectados con un bus y se comunican entre sí mediante el protocolo PROFIBUS DP.

Versión de firmware de la CPU S7-1200

El uso de las funcionalidades PROFIBUS en la S7-1200 requiere CPUs con una versión de firmware 2.0 o superior.

Configuración de una periferia descentralizada

Puesto que el maestro DP y los esclavos DP son dispositivos distintos, aquí se explican únicamente los procedimientos básicos de configuración. No obstante, la configuración de la periferia descentralizada es casi idéntica al procedimiento de configuración de una estructura centralizada.

Crear un sistema maestro DP en la vista de redes

Después de colocar un maestro DP y un esclavo DP (p. ej. un CM 1243-5 y un CM 1243-5) del catálogo de hardware en la vista de redes por medio de Drag & Drop, conecte ambos dispositivos con una subred PROFIBUS.

Información relacionada

Consulte la información complementaria sobre funcionalidad en los manuales del dispositivo correspondiente.

Esclavos DP en el catálogo de hardware

Esclavos DP en el catálogo de hardware

Los esclavos DP se encuentran en la carpeta "Periferia descentralizada" del catálogo de hardware. Allí se encuentran esclavos DP compactos y modulares:

- Esclavos DP compactos
  Módulos con entradas y salidas digitales/analógicas integradas, p. ej. ET 200L
- Esclavos DP modulares
  Módulos de interfaz con módulos S7 asignados, p. ej. ET 200M

Los esclavos DP utilizables dependen del maestro DP existente y de la funcionalidad deseada.

Esclavos I en el catálogo de hardware

Un esclavo DP que puede configurarse como esclavo DP inteligente es, por ejemplo, el CM 1242-5. Lo encontrará en el siguiente lugar del catálogo de hardware:

- CM 1242-5
  "PLC > SIMATIC S7 1200 > Módulo de comunicación > PROFIBUS"

Acopladores DP/DP en el catálogo de hardware

Introducción

Un acoplador DP/DP sirve para conectar dos redes PROFIBUS DP como pasarela (gateway) y transferir así datos del maestro DP de una red al maestro DP de la otra red.

El tamaño máximo de los datos transferibles es de 244 bytes para datos de entrada y 244 bytes para datos de salida.

Acopladores DP/DP en el catálogo de hardware

Los acopladores DP/DP como gateway entre dos sistemas maestros DP se encuentran en la carpeta "Otros dispositivos de campo > PROFIBUS DP > Gateways" del catálogo de hardware.

Configuración del acoplador DP/DP

Los acopladores DP/DP se configuran en ambas redes PROFIBUS, en sistemas maestros propios.

Las áreas de entradas y salidas de ambas redes deben estar sintonizadas unas con otras. Los datos de salida de una parte del acoplador DP/DP se aceptan como datos de entrada de la otra parte y viceversa.

Configuraciones con PROFIBUS DP

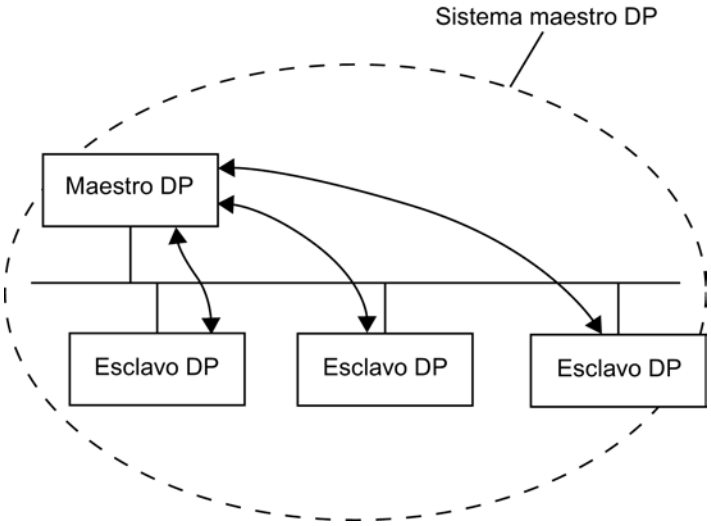
Configuraciones con esclavos DP sencillos

Intercambio de datos entre maestro DP y esclavo DP

En la configuración con esclavos DP sencillos, el intercambio de datos se efectúa entre maestro DP y esclavos DP sencillos, es decir, con módulos E/S a través del maestro DP. El maestro DP sondea sucesivamente cada esclavo DP configurado en su lista de llamada (Lista Polling) dentro del sistema maestro DP. Este transfiere los datos de salida a los esclavos y recibe los valores de entrada que estos devuelven.

Sistema monomaestro

La configuración con un solo maestro DP se denomina sistema monomaestro. Un único maestro DP está conectado a una subred PROFIBUS DP física con los correspondientes esclavos DP.



Configuraciones con esclavos DP inteligentes

Definición

Los esclavos DP con programa propio para preprocesamiento se denominan esclavos DP inteligentes (esclavos I).

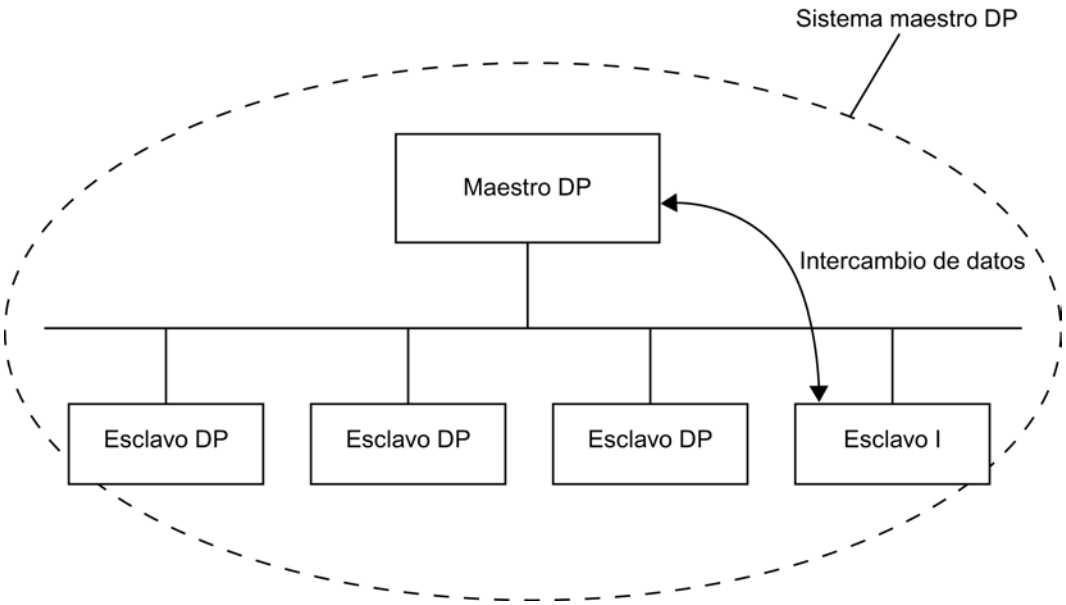
El CM 1242-5 es un esclavo DP inteligente.

Intercambio de datos esclavo I <> maestro DP

Un sistema de automatización de nivel superior procesa la tarea de automatización dividida en tareas parciales. Las tareas parciales se procesan en los sistemas de automatización de nivel inferior. Las tareas de control que se han de realizar se procesan de manera independiente y eficiente como programas del preprocesamiento en las CPUs.

En configuraciones con esclavos DP inteligentes, el maestro DP no accede a los módulos E/S del esclavo DP inteligente, sino solo al área de operandos de la CPU del esclavo I. El área de operandos no se puede ocupar para módulos E/S reales en el esclavo I. Hay que realizar esta asignación cuando se configura el esclavo I.

Las direcciones de los datos que van a intercambiarse entre maestro y esclavo se configuran en el área de transferencia del esclavo I.



Configuración de sistemas de periferia descentralizada

Sugerencia: Configuración rápida de sistemas maestros

Si el sistema maestro DP tiene muchos esclavos DP, asigne todos los esclavos DP posicionados mediante Drag & Drop a un maestro en un solo paso.

Requisitos

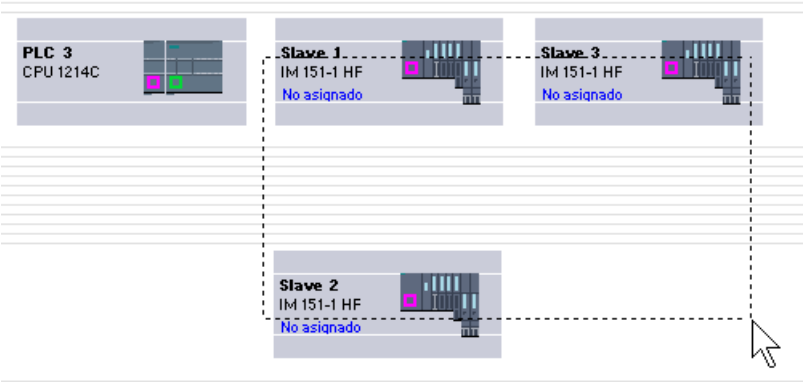
El maestro DP y los esclavos DP están posicionados en la vista de redes.

Asignar esclavos DP a un sistema maestro DP

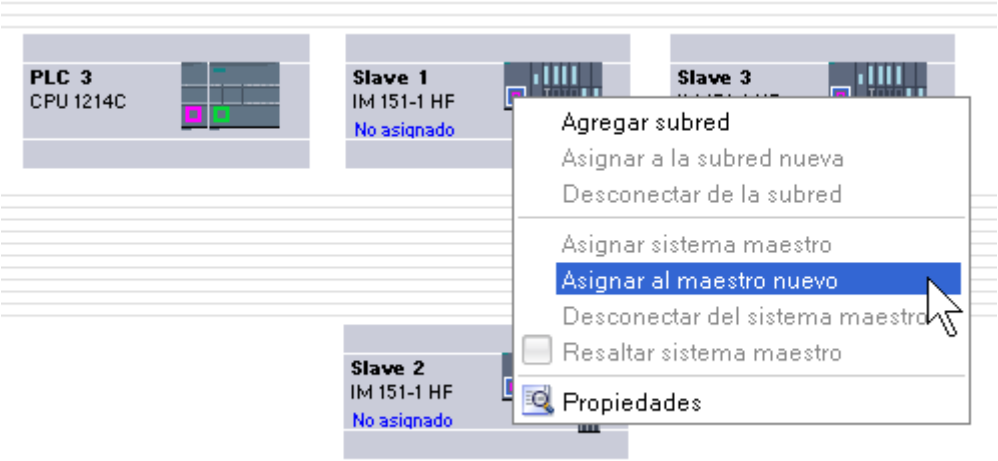
Proceda del siguiente modo:

1. Elija un factor de zoom apropiado de manera que pueda ver tantos esclavos DP como sea posible en la vista de redes.
2. Disponga los esclavos DP en dos filas como máximo.

3. Seleccione todas las interfaces DP con el puntero del ratón (¡no los dispositivos!). Esto sólo resultará si empieza a tirar del puntero del ratón fuera del primer esclavo y suelta el botón del ratón en el último esclavo DP (selección con lazo).



4. Elija en el menú contextual "Asignar al maestro nuevo" y seleccione el cuadro de diálogo siguiente la interfaz DP correspondiente del maestro DP.



5. Los esclavos DP se conectan automáticamente en red con el maestro DP y forman conjuntamente un sistema maestro DP.

Nota

Con un sistema maestro DP resaltado puede hacer doble clic en un esclavo DP en el catálogo de hardware y agregar así rápidamente otros esclavos DP. De este modo, el esclavo DP es incluido automáticamente en el sistema maestro DP resaltado.

Crear un sistema maestro DP

Introducción

Para crear un sistema maestro DP se necesita un maestro DP y al menos un esclavo DP. En cuando se conecta un maestro DP con un esclavo DP, se produce un acoplamiento maestro-esclavo.

Maestro DP

Como maestro DP se pueden utilizar los dispositivos siguientes:

- CM 1243-5

Requisitos

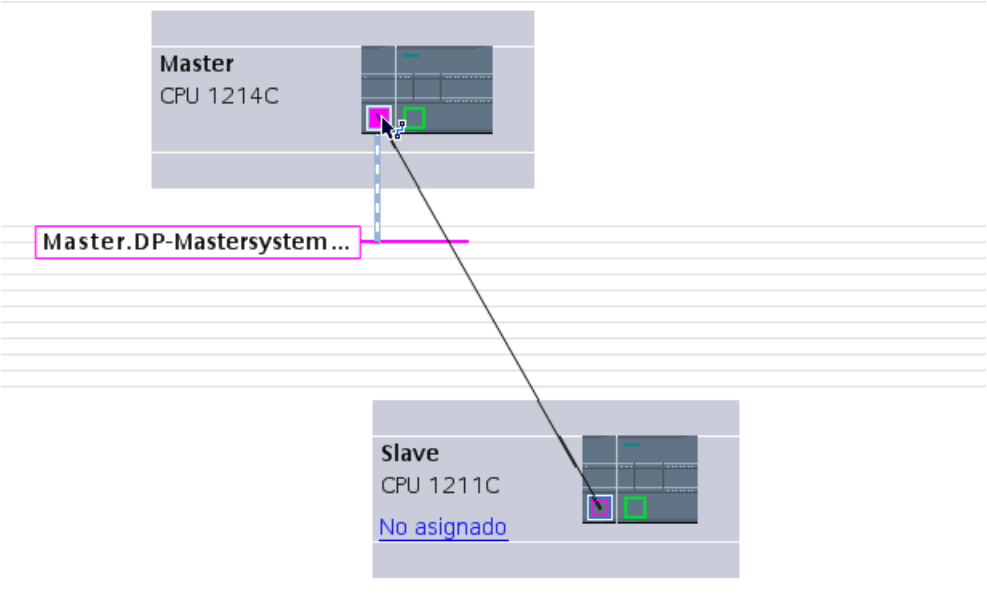
- Se encuentra en la vista de redes.
- El catálogo de hardware está abierto.

Procedimiento

Para crear un sistema maestro DP, proceda del siguiente modo:

1. Seleccione un maestro DP del catálogo de hardware.
2. Arrastre el maestro DP a la superficie libre en la vista de redes.
3. Haga clic con el botón derecho del ratón en la interfaz DP del maestro DP.
4. Seleccione "Crear sistema maestro" en el menú contextual.
   Se crea un sistema maestro DP con un maestro DP como único nodo.

Cuando se conecta la interfaz DP de un esclavo DP con la interfaz DP del maestro DP, el esclavo DP es incluido en el sistema maestro.



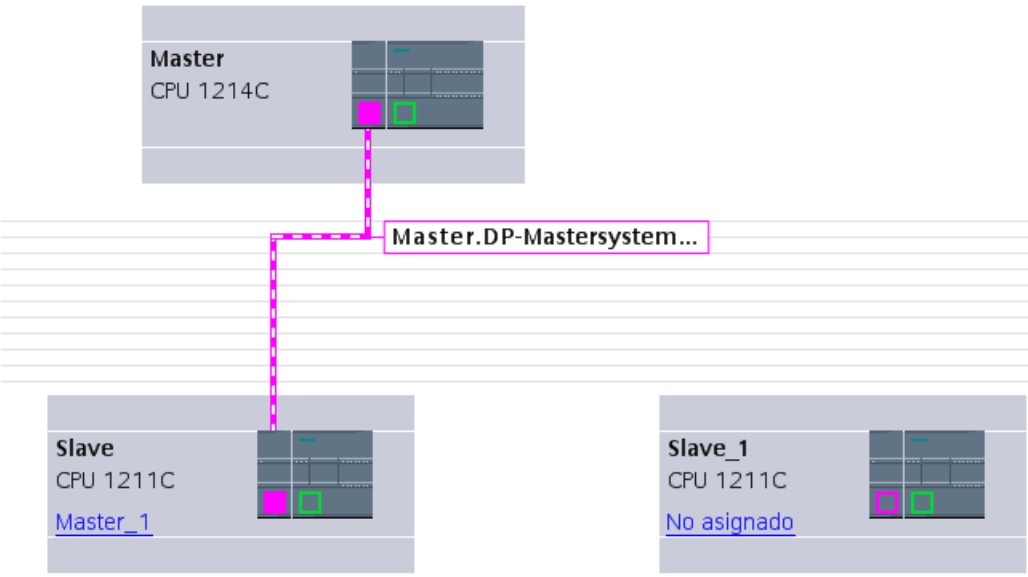
Si ya se ha colocado un maestro DP y un esclavo DP en la vista de redes, éstos se pueden enlazar con Drag&Drop y crear así un sistema maestro DP. Proceda del siguiente modo:

1. Haga clic en la interfaz DP del maestro DP o del esclavo DP.
2. Manteniendo pulsado el botón del ratón, arrastre una conexión de la interfaz DP seleccionada a la interfaz DP del interlocutor deseado.

Se crea una subred con un sistema maestro DP entre maestro DP y esclavo DP.

Visualizar el maestro DP en el esclavo DP

Si se conecta un esclavo DP con un maestro DP, el nombre del maestro DP se muestra como hipervínculo en el esclavo DP. Si se hace clic en el hipervínculo, se selecciona el maestro DP asignado.

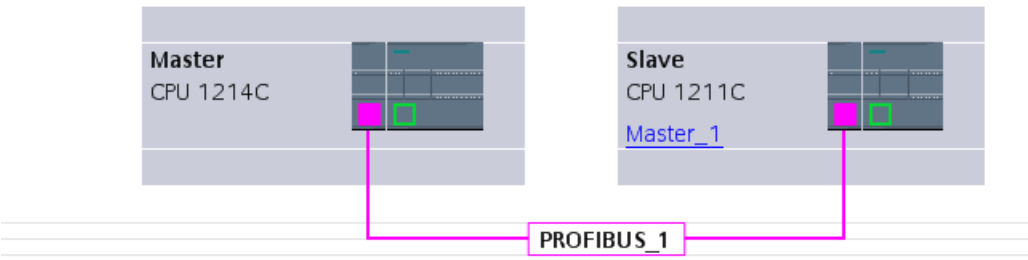


Resaltar el sistema maestro DP

Si se ha creado un nuevo sistema maestro DP, éste se representa resaltado. De este modo se reconoce rápidamente qué dispositivos pertenecen al sistema maestro DP. También se puede resaltar un sistema maestro DP arrastrando el puntero del ratón a una subred. Se visualizan a continuación los nombres de los sistemas maestros DP existentes. Al hacer clic con el ratón en uno de los sistemas maestros DP visualizados, se resalta el correspondiente sistema maestro DP.

Existen varias posibilidades para desactivar después el resalte de un sistema maestro DP:

- Resalte otro sistema maestro.
- Haga clic en el alfiler con la denominación del sistema maestro, situado en la esquina superior derecha de la vista de redes.



Editar sistemas maestros e interfaces DP

Introducción

Después de crear un sistema maestro DP, existe también la posibilidad de desconectar el sistema maestro DP de sus componentes. Así pueden generarse subredes con esclavos DP pero sin maestro DP.

Casi nunca es necesario editar las interfaces de un maestro DP.

Se puede modificar el nombre y el número del sistema maestro DP.

Desconexión del maestro o de esclavos de sistemas maestros DP

Si un CP PROFIBUS se ha configurado como maestro DP con sistema maestro, se puede desconectar el sistema maestro DP del maestro DP. El dispositivo ya no está conectado al sistema maestro DP.

La desconexión de la subred de un maestro DP provoca que el sistema maestro deje de existir, porque ya no hay asignado ningún maestro DP. Pero los esclavos DP continúan conectados entre sí a través de la subred.

Si se borran los esclavos DP o se desconectan del sistema maestro, este último se mantiene en el maestro DP.

Requisitos

- Se encuentra en la vista de redes.
- Hay un sistema maestro DP con un maestro DP y al menos un esclavo DP.

Desconectar el maestro DP del sistema maestro DP

Para desconectar el sistema maestro DP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la interfaz DP del maestro DP.
2. Seleccione "Desconectar del sistema maestro" en el menú contextual.

El maestro DP seleccionado se desconecta del sistema maestro DP. Queda una subred con los esclavos DP.

Agregar maestro DP al sistema maestro DP

Para asignar nuevamente un maestro DP a una subred, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la interfaz DP de un maestro DP.
2. Seleccione "Crear sistema maestro" en el menú contextual.
3. Arrastre el sistema maestro DP nuevo a las interfaces DP de los esclavos DP.

El maestro DP constituye con los esclavos DP otro sistema maestro DP.

Editar las propiedades de un sistema maestro DP

Para editar las propiedades de un sistema maestro DP, proceda del siguiente modo:

1. Desplace el puntero del ratón a una subred con sistema maestro DP.
2. En el aviso que aparece sobre los sistemas maestros DP disponibles, haga clic en el sistema maestro DP que quiera editar. El sistema maestro DP se resalta ahora en color.
3. Haga clic en el el sistema maestro DP resaltado.
4. Edite los atributos del el sistema maestro DP en "Propiedades > General" de la ventana de inspección.

Nota

Si se hace clic en una subred sin que se resalte un sistema maestro DP, es posible editar las propiedades de toda la subred en "Propiedades" de la ventana de inspección.

Insertar y configurar esclavos DP en el sistema maestro

Inserte en la vista de redes diversos esclavos DP directamente con Drag&Drop o haciendo doble clic en el catálogo de hardware.

Esclavos DP

En la configuración de esclavos DP, éstos se clasifican como:

- Esclavos DP compactos
  (módulos con entradas y salidas digitales/analógicas integradas, p. ej. ET 200L)
- Esclavos DP modulares
  (Módulos de interfaz con módulos S5 o S7 asignados, p. ej. ET 200M)
- Esclavos DP inteligentes (esclavos I)
  (CM 1242-5 o ET 200S con CPU IM 151-7)

Reglas

- En un sistema maestro DP se utiliza sólo un maestro DP y uno o varios esclavos DP.
- En un sistema maestro DP no puede haber más esclavos DP de los permitidos para el maestro DP en cuestión.

Nota

Para configurar el sistema maestro DP, tenga en cuenta los datos técnicos del maestro DP (número máx. de nodos, número máx. de slots, número máx. de datos de usuario). Es posible que no se utilice el número máximo de dispositivos debido a la limitación de los datos de usuario.

Requisitos

- Se encuentra en la vista de redes.
- Se ha creado un sistema maestro DP.

Insertar un esclavo DP en el sistema maestro DP

Para insertar un esclavo DP del catálogo de hardware en el sistema maestro DP, proceda del siguiente modo:

1. Seleccione un esclavo DP del catálogo de hardware.
2. Arrastre el esclavo DP con Drag&Drop del catálogo de hardware a la vista de redes.
3. Arrastre una conexión de la interfaz DP del maestro DP o del sistema maestro DP resaltado a la interfaz DP del nuevo esclavo DP.

Se crea automáticamente un sistema maestro DP y el esclavo DP se conecta también automáticamente al maestro DP.

Nota

En un sistema maestro DP resaltado se puede hacer doble clic en el esclavo DP deseado del catálogo de hardware. De este modo, el esclavo DP es incluido automáticamente en el sistema maestro DP resaltado.

Desconectar el esclavo DP del sistema maestro DP

Para desconectar un esclavo DP del sistema maestro DP, proceda del siguiente modo:

1. En la vista de redes, haga clic con el botón derecho del ratón en la interfaz DP del esclavo DP.
2. Seleccione el tipo de desconexión del sistema maestro DP en el menú contextual:
   - "Deshacer la conexión con la subred": la conexión PROFIBUS se deshace y el dispositivo ya no está conectado al sistema maestro DP o a una subred.
   - "Desconectar del sistema maestro": el esclavo DP queda conectado a la subred, pero ya no está asignado al sistema maestro DP como esclavo DP.

El esclavo DP seleccionado se desconecta del sistema maestro DP.

Asignar un esclavo DP a un sistema maestro DP nuevo

Para asignar un esclavo DP existente a un sistema maestro DP nuevo, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la interfaz DP del esclavo DP.
2. En el menú contextual, elija "Asignar al maestro nuevo".
   Es indiferente si el esclavo DP ya está asignado a otro sistema maestro DP.
3. Seleccione en la lista el maestro DP a cuyo sistema maestro DP quiera conectar el esclavo DP.
   El esclavo DP seleccionado se asigna a un sistema maestro DP nuevo.

Del mismo modo, es posible conectar un esclavo DP a otra subred mediante la función "Asignar a la subred nueva", sin acoplarlo a un sistema maestro DP existente.

Configurar esclavo DP

Para configurar un esclavo DP, proceda del siguiente modo:

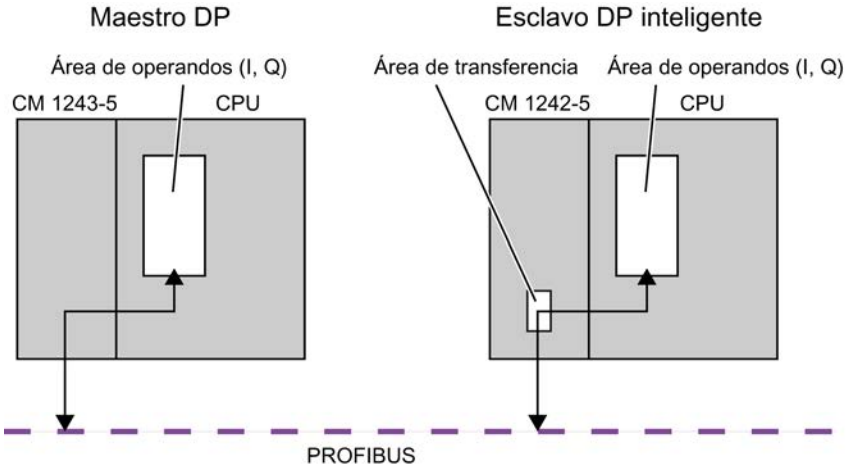
1. Cambie a la vista de dispositivos del esclavo DP.
2. Seleccione los módulos que desee utilizar.
3. Configure el esclavo DP en la ventana de inspección.

Configurar esclavos DP inteligentes

Insertar esclavo I en un sistema maestro DP

Introducción

Una característica del esclavo DP inteligente (esclavo I) es que los datos de entrada/salida no son facilitados al maestro DP directamente por una entrada/salida real, sino por una CPU de preprocesamiento. Dicha CPU constituye el esclavo I junto con el CP.



Diferencia: esclavo DP - esclavo DP inteligente

En un esclavo DP, el maestro DP accede a las entradas/salidas descentralizadas.

En un esclavo DP inteligente, el maestro DP no accede a las entradas/salida conectadas del esclavo DP inteligente, sino a un área de transferencia del área de direcciones E/S de la CPU

de preprocesamiento. El programa de usuario de la CPU de preprocesamiento debe procurar el intercambio de datos entre el área de operandos y las entradas/salidas.

Nota

Las áreas E/S configuradas para el intercambio de datos entre maestro DP y esclavo DP no pueden ser utilizadas por módulos E/S.

Aplicaciones

Configuraciones con esclavos DP inteligentes: Intercambio de datos esclavo I <> maestro DP

Procedimiento

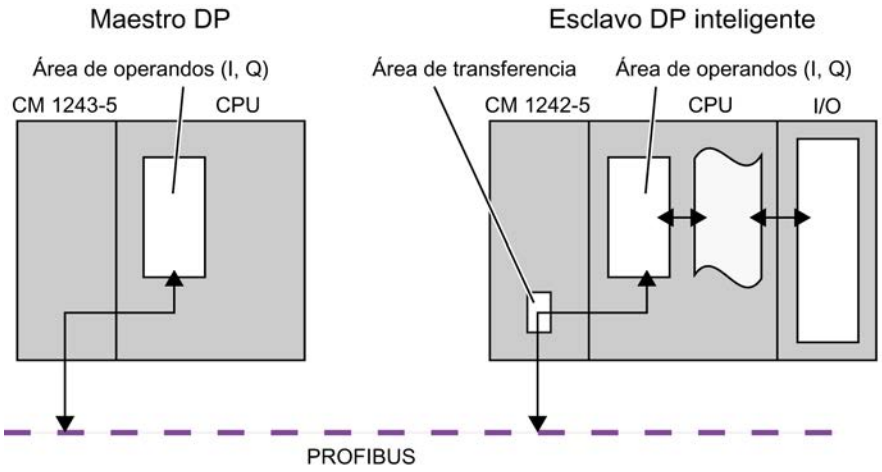
Para insertar un esclavo I en un sistema maestro DP, proceda del siguiente modo:

1. En la vista de redes arrastre un CM 1242-5 en función de esclavo I desde el catálogo de hardware hasta una estación, y un CM 1243-5 en función de maestro DP hasta otra estación.
2. Arrastre una línea de conexión entre las interfaces DP de ambos dispositivos.
   De este modo se conecta el esclavo I con un maestro DP en un sistema maestro DP.
   Resultado: Se ha establecido un sistema maestro DP con un maestro DP y un esclavo I.

Configurar el acceso a los datos del esclavo I

Acceso a datos

Para el CM 1242-5 en función de esclavo I: Las direcciones del área de transferencia de los datos y la dirección del módulo de E/S en el esclavo I son diferentes. Por ello, la dirección inicial que ocupa un módulo de E/S ya no se puede utilizar para la memoria de transferencia. Por consiguiente, si el maestro DP de orden superior debe acceder a datos de un módulo de E/S en el esclavo I, es necesario configurar este intercambio de datos entre el módulo de E/S y el área de transferencia en el programa de usuario del esclavo I.



Configuración del área de transferencia en el CM 1242-5 (área de transferencia)

En el CM 1242-5, el área de transferencia para el intercambio cíclico de datos PROFIBUS se configura en el grupo de parámetros "Interfaz PROFIBUS > Modo de operación > Comunicación de esclavo I".

Acceso directo a datos de CPU a CPU

Los CMs PROFIBUS S7-1200 sólo soportan el acceso directo a datos de CPU a CPU vía PROFIBUS utilizando los servicios PUT/GET.

Configurar esclavos DP de la periferia descentralizada

Configurar ET 200S

Reglas de slots para la configuración de un ET 200S

Para configurar el ET 200S se aplican las reglas siguientes:

- Inserte los módulos del ET 200S uno junto a otro sin dejar espacio.
- Slot 1: sólo para módulos de potencia PM-E o PM-D
- A la izquierda de un módulo electrónico (EM): sólo un EM o un módulo de potencia (PM-E o PM-D)
- A la izquierda de un arrancador de motor (MS): sólo un MS o un módulo de potencia PM-D, PM-D Fx (1..x..4) o PM-X
- A la izquierda de un PM-X: sólo un arrancador de motor o un PM-D
- Se permiten como máximo 63 módulos y un módulo de interfaz IM

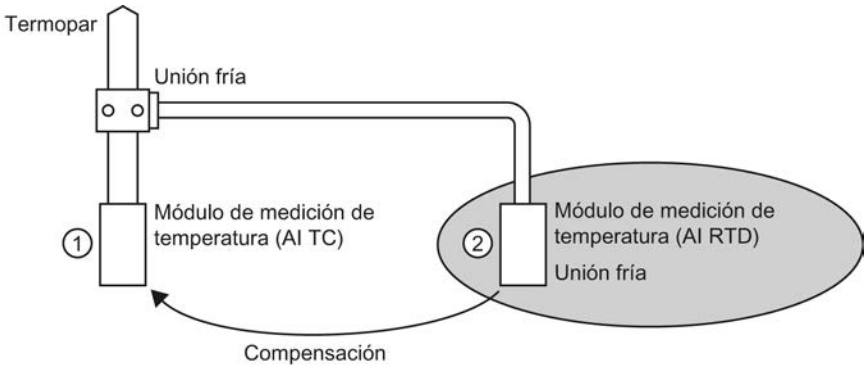
Nota

Tenga en cuenta la asignación correcta de los rangos de tensión PM-E y EM.

Parametrizar una unión fría

Una unión fría es la conexión de un termopar con un cable de entrada (por norma general en la caja de bornes). La tensión que se genera aquí debido al efecto de la temperatura falsifica el valor de temperatura que mide el módulo.

En un ET 200S es posible parametrizar un canal del módulo AI RTD como unión fría. Gracias a la temperatura que mide dicho módulo en la unión fría, otros módulos AI TC pueden compensar los valores medidos.



① Parametrización del AI TC:
→ Selección de la unión fría utilizada

② Parametrización del AI RTD:
→ Activación de la unión fría
→ Definición de slot y canal del AI RTD

Particularidades de la parametrización de uniones frías

La parametrización de uniones frías se describe mediante ejemplos de uso de una termorresistencia Pt 100 "Climatiz." para registrar la temperatura de la unión fría.

Para parametrizar la unión fría, proceda del siguiente modo:

1. Coloque un módulo electrónico analógico del ET 200, p. ej. 2AI RTD HF, en la vista de dispositivos.
2. Seleccione el módulo en el rack.
3. En la ventana de inspección, bajo "Propiedades > Entradas" , ajuste un canal para la función de unión fría al rango de medición "RTD-4L Pt 100 cl.".
4. Seleccione el ET 200S.
5. Active la casilla de verificación "Unión fría" en "Propiedades > Parámetros del módulo > Uniones frías" de la ventana de inspección e indique el slot y el número de canal del módulo RTD relevante.
6. Coloque el módulo electrónico analógico para medir la temperatura con termopar (módulo TC) y parametrícelo con el número de unión fría del módulo RTD.

Información adicional

Para obtener información adicional sobre los diversos tipos y posibilidades de utilización de módulos en ET 200S, consulte las instrucciones de servicio y el manual de producto "Sistema de periferia descentralizada ET 200S".

Para obtener información adicional sobre el procesamiento de valores analógicos, consulte la documentación del sistema de periferia descentralizada ET 200S.

Comprimir direcciones

Introducción

Los esclavos DP y los dispositivos IO de la familia ET 200S se configuran como otros esclavos DP y dispositivos IO modulares. Además de las funciones habituales de los esclavos DP y dispositivos IO modulares, el ET 200S incorpora la función "Comprimir direcciones":

Los módulos electrónicos digitales con un área de direcciones necesaria de 2 ó 4 bits ocupan por principio 1 byte cuando se insertan en la vista de dispositivos. Con todo, el área de direcciones realmente ocupada se puede comprimir después de la configuración con la función "Comprimir direcciones".

	Ajuste predeterminado	Después de "Comprimir direcciones"
Módulo	Dirección I	Dirección I
2DI (2 bits)	Byte 10	10.0...10.1
4DI (4 bits)	Byte 11	10.2...10.5

Requisitos

- Se encuentra en la vista de dispositivos.
- Hay un ET 200S, p. ej. IM 151-1.
- Hay algunos módulos electrónicos digitales en los slots, p. ej. 2DI AC120V ST.

Comprimir direcciones

Para comprimir direcciones, proceda del siguiente modo:

1. Seleccione los módulos electrónicos cuyas direcciones quiera comprimir. Para seleccionar varios módulos electrónicos, existen las posibilidades siguientes:
   - Haga clic en los módulos electrónicos sucesivamente pulsando a la vez la tecla <Mayús> o <Ctrl>.
   - Haga clic fuera del rack y arrastre el ratón rodeando los módulos electrónicos que quiera seleccionar.
2. Haga clic en "Comprimir direcciones" del menú contextual de los módulos electrónicos seleccionados.

Se comprimen por separado las áreas de direcciones de entradas, salidas y arrancadores de motor. En las columnas Dirección I y Dirección Q de la vista general de dispositivos se ven las direcciones comprimidas.

Formación de direcciones y estructura de las direcciones comprimidas

En caso de utilizar la función "Comprimir direcciones", las direcciones de los módulos electrónicos seleccionados se comprimen de acuerdo con las reglas siguientes:

- El principio del área de direcciones está determinado por la dirección más baja de los módulos electrónicos seleccionados: X.0.
- Si la dirección del bit no es "0", automáticamente se utiliza la siguiente dirección (libre) del byte a partir de la cual se puede introducir el área seleccionada: (X+n).0.
- Cuando ya no queda ningún área conexa, se comprime automáticamente en los huecos de direcciones existentes.

Los módulos electrónicos con direcciones comprimidas y la misma dirección de byte constituyen un grupo de compresión.

Descomprimir direcciones

Para descomprimir direcciones, proceda del siguiente modo:

1. Seleccione uno o varios módulos electrónicos con dirección comprimida.
2. Haga clic en "Descomprimir direcciones" del menú contextual de los módulos electrónicos seleccionados.

Los grupos de compresión de los módulos electrónicos seleccionados se descomponen y las direcciones comprimidas de los respectivos módulos electrónicos se descomprimen.

Si se borran o desplazan módulos electrónicos de un grupo de compresión o bien se insertan módulos electrónicos en un slot libre dentro de un grupo de compresión, el grupo de compresión se descompone igualmente y las direcciones comprimidas se descomprimen.

Las direcciones iniciales de los módulos electrónicos descomprimidos se colocan en las siguientes direcciones de byte libres en cada caso.

Particularidades de los módulos electrónicos con direcciones comprimidas

Para un módulo electrónico con una dirección comprimida rigen las particularidades siguientes:

- Desde el punto de vista de la CPU, no es posible asignar slots al módulo electrónico. Por este motivo la instrucción GADR_LGC (SFC 5) devuelve para el slot real del módulo electrónico la información de error W#16#8099 "Slot no configurado".
- La instrucción LGC_GADR (SFC 49) y la ID SZL W#16#xy91 "Información de estado del módulo" de un módulo electrónico no se pueden evaluar.
- El módulo electrónico recibe una dirección de diagnóstico adicional mediante la funcionalidad DPV1, porque desde el punto de vista de la CPU no se pueden asignar alarmas debido a la dirección comprimida.
- La "Alarma de extracción/inserción" no es posible, porque las funciones "Comprimir direcciones" y "Alarma de extracción/inserción" se excluyen mutuamente.

Configuración futura con módulos de reserva

La configuración futura permite preparar el ET 200S con interfaz PROFIBUS para futuras ampliaciones (opciones). En el presente apartado se describe la configuración futura con módulos de reserva.

Para ello, monte, cablee, configure y programe la configuración máxima prevista del ET 200S y utilice primero los módulos de reserva económicos (138-4AA00 ó 138-4AA10) en lugar de los módulos electrónicos necesarios más tarde en el montaje

Nota

El ET 200S se puede precablear completamente con el cableado principal, porque un módulo de reserva no está conectado con los bornes del módulo de pines ni con el proceso.

Requisitos

- Módulo de interfaz ET 200S
  - IM 151-1 STANDARD (a partir de 6ES7 151-1AA03-0AB0)
  - IM 151-1 FO STANDARD (a partir de 6ES7 151-1AB02-0AB0)
- Módulo de potencia con configuración futura
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Procedimiento

Para activar la configuración futura, proceda del siguiente modo:

1. Seleccione en la vista de dispositivos el IM 151-1 y active la casilla de verificación "Configuración futura" en "Propiedades > General > Configuración futura" de la ventana de inspección.
2. Active ahora la casilla de verificación numerada para los slots que están ocupados en principio por módulos de reserva en lugar de módulos electrónicos.
3. Seleccione en la vista de dispositivos el módulo de potencia y active la casilla de verificación "Configuración futura" en "Propiedades > General > Direcciones" de la ventana de inspección. Reserve para la interfaz de control y notificación el área de direcciones necesaria para ello en la memoria imagen de proceso de las salidas (IPS) y en la memoria imagen de proceso de las entradas (IPE).

Más adelante se pueden sustituir los módulos de reserva montados por los módulos configurados sin tener que actualizar la configuración.

Nota

Las direcciones de estas interfaces quedan reservadas en cuanto se activa la configuración futura en el módulo de potencia. La función "Configuración futura" también debe estar activada en el esclavo DP (módulo de interfaz IM 151-1 STANDARD). Si no está activada, las direcciones reservadas para la interfaz de control y notificación se liberan.

Recuerde que la activación y desactivación repetida de la configuración futura puede modificar la dirección de la interfaz de control y notificación.

La configuración futura sólo debe estar activada si hay exactamente un módulo de potencia PM-E DC24..48V o PM-E DC24..48V/AC24..230V.

Información adicional

Para obtener información adicional sobre la ocupación y el significado de los bytes en la memoria imagen de proceso, la configuración futura del PROFIBUS y la utilización de módulos de reserva, consulte la documentación del sistema de periferia descentralizada ET 200S.

Funcionamiento de la configuración futura en el arranque

Cuando "Arranque si la configuración real difiere de la teórica" está bloqueado, el ET 200S funciona aunque haya insertado un módulo de reserva en lugar del módulo electrónico configurado y la configuración futura esté activada para este slot.

Funcionamiento de la configuración futura durante el servicio

Durante el servicio, el funcionamiento de la configuración futura se distingue de esta manera:

- Configuración futura conectada para un slot:
  En este slot puede haber un módulo de reserva (opcional) o el módulo electrónico configurado. Si hay otro módulo en este slot, se notifica un diagnóstico (ningún módulo o módulo incorrecto).
- Configuración futura desconectada para un slot:
  En este slot puede haber sólo el módulo electrónico configurado. Con cualquier otro módulo se notifica un diagnóstico (ningún módulo o módulo incorrecto).

Valores sustitutivos del módulo de reserva

- Valor sustitutivo para entradas digitales: 0
- Valor sustitutivo para entradas analógicas: 0x7FFF

Control y evaluación en el programa del usuario

El ET 200S dispone de una interfaz de control y notificación para la función "Configuración futura".

La interfaz de control se encuentra en la memoria imagen de proceso de las salidas (IPS). Cada bit de esta área de direcciones controla uno de los slots 2 a 63:

- Valor del bit = 0: rige la parametrización de la configuración futura. Se permiten módulos de reserva.
- Valor del bit = 1: la parametrización de la configuración futura está suprimida. Los módulos de reserva no se aceptan en este slot:

La interfaz de notificación se encuentra en la memoria imagen de proceso de las entradas (IPE). Cada bit de esta área de direcciones da información sobre el módulo realmente insertado en los slots 1 a 63:

- Valor del bit = 0: en el slot se encuentran el módulo de reserva, un módulo incorrecto o un módulo extraído.
- Valor del bit = 1: en el slot se encuentra el módulo configurado.

Consulte también

¿Qué módulos soportan la configuración futura? (http://support.automation.siemens.com/WW/view/es/22564754)

Configuración futura sin módulos de reserva

La configuración futura permite preparar el ET 200S para futuras ampliaciones (opciones) incluso sin el montaje de módulos de reserva. En el presente apartado se describe la configuración futura sin módulos de reserva.

Nota

ET 200S con interfaz PROFINET

Esta descripción se refiere a ET 200S con interfaz PROFIBUS. La configuración futura para ET 200S con interfaz PROFINET funciona en principio como se describe aquí sin módulos de reserva. En vez de los módulos de interfaz DP aquí incluidos, deben utilizarse módulos de interfaz PN. Encontrará más información sobre la configuración futura en ET 200S con interfaz PROFINET en los correspondientes manuales de producto.

Requisitos

- Módulo de interfaz ET 200S
  - IM 151-1 HIGH FEATURE (a partir de 6ES7151-1BA02)
  - IM 151-1 STANDARD (a partir de 6ES7 151-1AA05-0AB0)
- Módulo de potencia con configuración futura
  - PM-E DC24..48V
  - PM-E DC24..48V/AC24..230V

Procedimiento

Para activar la configuración futura, proceda del siguiente modo:

1. Seleccione en la vista de dispositivos el IM 151-1 y active la casilla de verificación "Configuración futura" en "Propiedades > General > Configuración futura" de la ventana de inspección.
2. Seleccione en la vista de dispositivos el módulo de potencia y active la casilla de verificación "Configuración futura" en "Propiedades > General > Direcciones" de la ventana de inspección. Reserve para la interfaz de control y notificación el área de direcciones necesaria para ello en la memoria imagen de proceso de las salidas (IPS) y en la memoria imagen de proceso de las entradas (IPE).
3. Configure la configuración máxima del esclavo. La activación/desactivación de opciones se controla a través del programa de usuario.

Nota

Las direcciones de estas interfaces quedan reservadas en cuanto se activa la configuración futura en el módulo de potencia. La función "Configuración futura" también debe estar activada en el esclavo DP (módulo de interfaz IM 151-1). Si no está activada, las direcciones reservadas para la interfaz de control y notificación se liberan.

Recuerde que la activación y desactivación repetida de la configuración futura puede modificar la dirección de la interfaz de control y notificación.

La configuración futura sólo debe estar activada si hay exactamente un módulo de potencia PM-E DC24..48V o PM-E DC24..48V/AC24..230V.

Información adicional

Para obtener información adicional sobre la ocupación y el significado de los bytes en la memoria imagen de proceso, la configuración futura del PROFIBUS y la utilización de módulos de reserva, consulte la documentación del sistema de periferia descentralizada ET 200S.

Control y evaluación en el programa del usuario

El ET 200S dispone de una interfaz de control y notificación para la función "Configuración futura".

La interfaz de control se encuentra en la memoria imagen de proceso de las salidas (IPS). Cada bit de esta área de direcciones controla uno de los slots 1 a 63:

- Valor del bit = 0: El slot no existe en la configuración real.
- Valor del bit = 1: El slot existe en la configuración real.

La interfaz de notificación se encuentra en la memoria imagen de proceso de las entradas (IPE). Cada bit de esta área de direcciones da información sobre el módulo realmente insertado en los slots 1 a 63:

- Valor del bit = 0: El slot corresponde a una opción inexistente o el estado del módulo no es correcto.
- Valor del bit = 1: En el slot se encuentra el módulo configurado.

Consulte también

Aplicación de ejemplo para ET 200S, configuración futura sin módulos de reserva (http://support.automation.siemens.com/WW/view/es/29430270)

Configurar ET 200S en modo DPV1

Con PROFIBUS DPV1 se facilitan funcionalidades PROFIBUS ampliadas.

Requisitos

- Se encuentra en la vista de redes.
- Hay un maestro DP con funcionalidad DPV1.
- Se establece una conexión maestro-esclavo con PROFIBUS.

Procedimiento

Para conectar el esclavo DP en DPV1, proceda del siguiente modo:

1. Seleccione el esclavo DP.
2. Seleccione el modo de operación "DPV1" en la lista desplegable "Modo alarma DP" de "Propiedades > Parámetros del módulo" de la ventana de inspección.

o bien

1. Seleccione el maestro DP.
2. Seleccione en la tabla de comunicación E/S la fila con la conexión entre el maestro DP y el esclavo DP deseado.
3. Seleccione el modo de operación "DPV1" en la lista desplegable "Modo alarma DP" de "Propiedades > Parámetros del módulo" de la ventana de inspección.

Particularidades

Existen dependencias entre los distintos parámetros, que se representan a continuación:

Parámetro	Modo de operación DPV0	Modo de operación DPV1
Funcionamiento si configuración teórica diferente a configuración real	Utilizable sin limitaciones	Utilizable sin limitaciones
Alarma de diagnóstico (OB 82)	No utilizable, no activado	Utilizable sin limitaciones
Alarma de proceso (OB 40 hasta 47)	No utilizable, no activado	Utilizable sin limitaciones
Alarma de extracción/inserción (OB 83)	No utilizable, no activado	Sólo utilizable si las direcciones no están comprimidas. Si la alarma de extracción/inserción está activada, se activa automáticamente "Arranque si la configuración real difiere de la teórica".

Alarmas en módulos con direcciones comprimidas

Si el módulo puede disparar alarmas y la dirección del bit difiere de 0 debido a las direcciones comprimidas, es necesario asignar una dirección de diagnóstico al ET 200S en el cuadro de diálogo de direcciones.

La dirección de diagnóstico es necesaria para asignar una alarma DPV1 al módulo como causante de la alarma. La CPU sólo puede asignar una alarma cuando un módulo tiene esta dirección "descomprimida" y hay depositados datos sobre la alarma en la información de arranque del OB de alarma o en el búfer de diagnóstico. La CPU no puede utilizar para ello una dirección "comprimida".

Desde el punto de vista del procesamiento de alarmas (OB de alarma), el módulo tiene la dirección de diagnóstico asignada; para el procesamiento de datos de entrada y salida en el programa de usuario, el módulo tiene las direcciones comprimidas.

Nota

Si las direcciones del módulo están comprimidas, la alarma de extracción/inserción está bloqueada para el ET 200S.

Utilizar archivos GSD

Revisiones GSD

Información importante sobre las revisiones GSD

Los archivos GSD ponen a disposición las propiedades de los esclavos DP para las herramientas de configuración.

Las ampliaciones de funciones en el ámbito de la periferia descentralizada repercuten en la especificación GSD, por ejemplo conducen a la definición de nuevas palabras clave.

De esta forma se obtiene un versionamiento de la especificación. La versión de la especificación en la que se basa un archivo GSD recibe el nombre de "Revisión GSD" en el caso de archivos GSD.

La revisión GSD debe incluirse de forma obligatoria como palabra clave "GSD_Revision" en los archivos GSD a partir de la revisión GSD 1. Los archivos GSD sin esta palabra clave serán interpretados por las herramientas de configuración como revisión GSD "0".

Los archivos GSD pueden interpretarse hasta la revisión GSD 5. De este modo, se soportan los esclavos DP que, por ejemplo, son compatibles con las siguientes funciones:

- Avisos de diagnóstico para bloques de alarmas
- Modo isócrono y equidistancia
- SYNC/FREEZE
- Sincronización horaria para esclavos DP

Instalar archivo GSD

Introducción

En un archivo GSD (archivo de datos maestros del dispositivo) se encuentran almacenadas todas las propiedades de un esclavo DP. Si desea configurar un esclavo DP que no aparece en el catálogo de hardware, es necesario instalar el archivo GSD suministrado por el fabricante. Los esclavos DP instalados a partir de archivos GSD se muestran en el catálogo de hardware y pueden seleccionarse y configurarse.

Requisitos

- El editor de hardware y redes está cerrado.
- Tiene acceso a los archivos GSD necesarios en un directorio del disco duro.

Procedimiento

Para instalar un archivo GSD, proceda del siguiente modo:

1. Elija el comando "Instalar archivos de descripción del dispositivo" del menú "Herramientas".
2. Seleccione en el cuadro de diálogo "Instalar archivos de descripción del dispositivo" el directorio en el que se encuentran los archivos GSD.
3. Elija uno o varios de los archivos GSD que aparecen en la lista.
4. Haga clic en el botón "Instalar".
5. Para crear un archivo de informe de la instalación, haga clic en el botón "Guardar archivo de informe".
   Los posibles problemas que puedan aparecer durante la instalación pueden repasarse con el archivo de informe.

El nuevo esclavo DP instalado desde el archivo GSP se encuentra en una nueva carpeta del catálogo de hardware.

Nota

La instalación de un archivo GSD no puede deshacerse.

Configurar un esclavo DP basado en GSD

Los esclavos DP que se insertan con la instalación de un archivo GSD pueden seleccionarse del modo habitual en el catálogo de hardware e insertarse en la vista de redes. Si se desean insertar los módulos de los esclavos DP basados en GSD, deben tenerse en cuenta algunas particularidades.

Requisitos

- Se ha instalado un esclavo DP a partir de un archivo GSD.
- Se ha insertado el módulo de cabecera en la vista de redes de la forma habitual.

- La vista general de dispositivos está abierta en la vista de dispositivos.
- El catálogo de hardware está abierto.

Procedimiento

Para agregar los módulos de un esclavo DP basado en GSD, proceda del siguiente modo:

1. Navegue en el catálogo de hardware hasta los módulos del esclavo DP basado en GSD. Los esclavos DP basados en GSD, también denominados esclavos DP normalizados, se encuentran en la carpeta "Otros dispositivos de campo" del catálogo de hardware.
2. Seleccione el módulo deseado.
3. Arrastre el módulo con Drag&Drop hasta un espacio libre de la vista general de dispositivos.
4. Seleccione el módulo en la vista general de dispositivos para poder editar sus parámetros.

Se ha insertado el módulo en un slot libre del esclavo DP basado en GSD y pueden editarse sus parámetros.

Nota

En el área gráfica de la vista de dispositivos sólo se ve el esclavo DP basado en GSD. Los módulos agregados de los esclavos DP basados en GSD sólo se encuentran en la vista general de dispositivos.

Configuración teórica

En los módulos con configuración teórica ajustable, ésta puede modificarse en "Propiedades > Configuración teórica" de la ventana de inspección.

8.1.4.12 Configuraciones para PROFINET IO

Información importante sobre PROFINET IO

¿Qué es PROFINET IO?

PROFINET IO

PROFINET es un estándar de automatización basado en Ethernet de PROFIBUS Nutzerorganisation e.V. (PNO) que define un modelo de comunicación, automatización e ingeniería para todos los fabricantes.

Objetivos

Los objetivos de PROFINET son:

- una comunicación continua mediante bus de campo y Ethernet
- una automatización abierta y distribuida
- la utilización de estándares abiertos

Arquitectura

La PROFIBUS Nutzerorganisation e.V. (PNO) ha previsto los siguientes aspectos parciales para la arquitectura de PROFINET:

- comunicación entre controladores como componentes en sistemas distribuidos.
- comunicación entre aparatos de campo como p. ej. unidades periféricas y accionamientos

Aplicación en Siemens

La exigencia de una "comunicación entre controladores como componentes en sistemas distribuidos" es aplicada por "Component Based Automation" (CBA). Con Component Based Automation se elabora una solución de automatización distribuida basada en componentes preelaborados y soluciones parciales.

La exigencia de una "comunicación entre aparatos de campo" es aplicada por Siemens con "PROFINET IO". Como en PROFIBUS DP, la configuración y la programación completas de los componentes utilizados es posible con el Totally Integrated Automation Portal.

En los siguientes apartados se explica la configuración de la comunicación entre aparatos de campo con PROFINET IO.

Vista general de las clases RT

Clases RT en PROFINET IO

PROFINET IO es un sistema de comunicación en tiempo real escalable basado en la tecnología Ethernet. El principio escalable se expresa con diversas clases de tiempo real:

- **RT:** transferencia de datos en telegramas Ethernet priorizados, sin sincronismo. El ancho de banda necesario está dentro del rango de ancho de banda libre para la comunicación TCP/IP.
- **IRT:** transferencia isócrona de datos con estabilidad alta para aplicaciones de tiempo crítico (p. ej. Motion Control). El ancho de banda necesario está dentro del rango de ancho de banda reservado para datos cíclicos.

Dependiendo del dispositivo, no se soportan todas las clases de tiempo real.

Conexión de sistemas de bus existentes

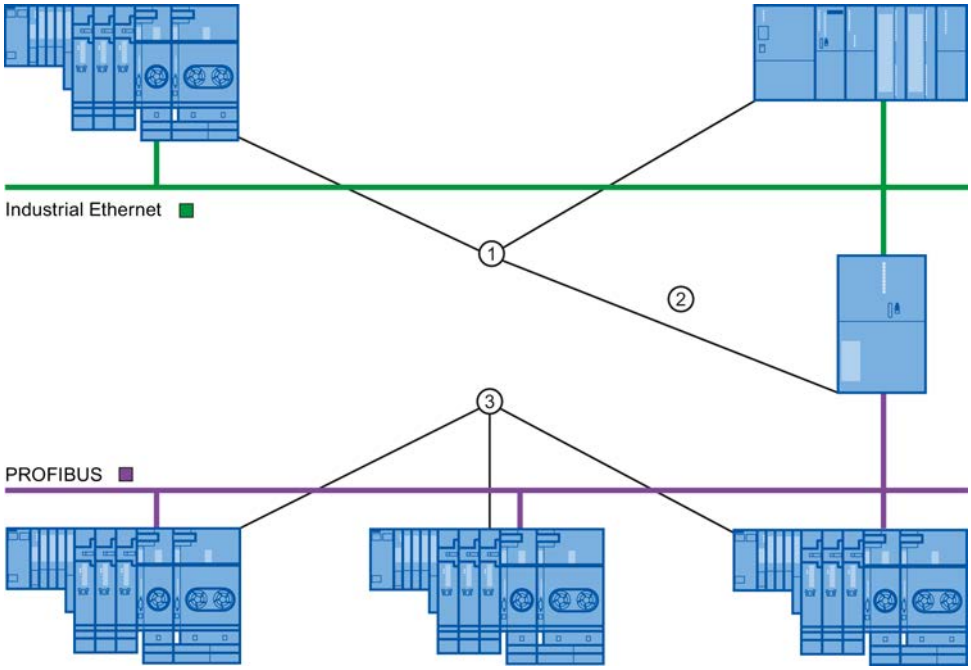
Acoplamiento de PROFINET y PROFIBUS

PROFINET IO y PROFIBUS DP pueden acoplarse entre sí utilizando los métodos siguientes:

- vía Industrial Ethernet:
  Para conectar los dos tipos de red Industrial Ethernet (nivel de gestión) y PROFIBUS (nivel de célula/campo) se utiliza, p. ej., el IE/PB-Link.
- vía Industrial Wireless LAN:
  Es posible acoplar los dispositivos PROFIBUS, p. ej. de forma inalámbrica, a PROFINET IO mediante un Wireless-LAN/PB-Link. De este modo se integran las configuraciones PROFIBUS existentes en PROFINET.

Los dispositivos con AS-Interface se pueden acoplar con un IE/AS-i Link PN IO a la interfaz de un dispositivo PROFINET. De este modo se integra la red AS-i existente en PROFINET.

La figura siguiente muestra la conexión de una subred PROFIBUS mediante un dispositivo PROFINET con funcionalidad proxy.



① Dispositivos PROFINET
② Dispositivo PROFINET con funcionalidad proxy (p. ej. IE/PB-Link)
③ Dispositivos PROFIBUS

Dispositivo PROFINET con funcionalidad proxy como sustituto de un dispositivo PROFIBUS

El dispositivo PROFINET con funcionalidad proxy es el sustituto de un dispositivo PROFIBUS en Ethernet. La funcionalidad proxy permite que un dispositivo PROFIBUS se comunique no solo con su maestro sino con todos los nodos en PROFINET.

Los sistemas PROFIBUS existentes se pueden integrar sin problemas en la comunicación PROFINET utilizando la funcionalidad proxy.

Si, por ejemplo, se conecta un dispositivo PROFIBUS a PROFINET mediante un IE/PB-Link, este último asume la comunicación vía PROFINET como sustituto de los componentes PROFIBUS.

Configuración con IE/PB-Link PN IO

Configuración con IE/PB-Link PN IO

Para integrar configuraciones PROFIBUS DP en PROFINET IO, es posible utilizar el IE/PB-Link PN IO.

Desde el punto de vista de la CPU, los esclavos PROFIBUS DP están ubicados en la misma red que el IE/PB-Link PN IO. Estos esclavos tienen el mismo nombre de dispositivo y dirección IP que el IE/PB-Link PN IO, pero distinto número de dispositivo. Además, estos incorporan otra dirección PROFIBUS específica.

En las propiedades del IE/PB-Link se muestran las direcciones PROFIBUS de los esclavos DP conectados junto a los números de dispositivo PROFINET, porque este dispositivo posee dos esquemas de direccionamiento.

Manejo de números de dispositivo y direcciones PROFIBUS en el sistema maestro

En la colocación se asigna la misma cifra para el número de dispositivo PROFINET y la dirección PROFIBUS.

En "Propiedades > General > Número de dispositivo PROFINET" de la ventana de inspección encontrará un resumen de los números de dispositivo utilizados y las direcciones PROFIBUS de un IE/PB-Link. Allí también existe la posibilidad de cambiar los números de dispositivo. También es posible especificar si el número de dispositivo y la dirección PROFIBUS deben ser siempre idénticos o no. Si la opción "Número de dispositivo PROFINET=Dirección PROFIBUS" está activada, ya no será necesario corregir el nombre de dispositivo cuando cambie la dirección PROFIBUS.

La dirección PROFIBUS se modifica en las propiedades del dispositivo PROFIBUS.

Restricciones

En la subred PROFIBUS de un IE/PB-Link rigen las siguientes restricciones para los esclavos DP en la configuración arriba descrita:

- No hay ningún IE/PB-Link insertable
- No hay ningún DP/PA-Link insertable
- No hay ningún Y-Link insertable
- No apto para CiR
- No hay ningún esclavo redundante insertable

- No es posible configurar sincronismo / equidistancia
- No se soportan las instrucciones SYNC/FREEZE ("DPSYC_FR") de una CPU en la subred Ethernet para esclavos DP detrás del IE/PB-Link.

Consulte también

Modos de operación del IE/PB Link PN IO (Página 2078)

Configuración con IWLAN/PB-Link

Número máximo de dispositivos en un segmento IWLAN

Si una subred Ethernet está estructurada como red por radiofrecuencia (IWLAN = Industrial Wireless LAN), el intercambio cíclico de datos entre controladores IO y dispositivos IO es posible mediante una línea inalámbrica.

En un lado de la línea inalámbrica hay Access Points instalados de forma fija (p. ej. SCALANCE W 788) y, en el otro lado, estaciones móviles (p. ej. con IWLAN/PB-Links con dispositivos PROFIBUS).

Si el radio de acción de las estaciones móviles es grandes, posiblemente deberán instalarse varios Access Points (SCALANCE W 788). Cada Access Point constituye un segmento con su alcance, de modo que la IWLAN se forma con una serie de segmentos.

Los dispositivos móviles "del otro lado" de la línea inalámbrica con sus IWLAN/PB-Links pueden moverse a lo largo de los segmentos.

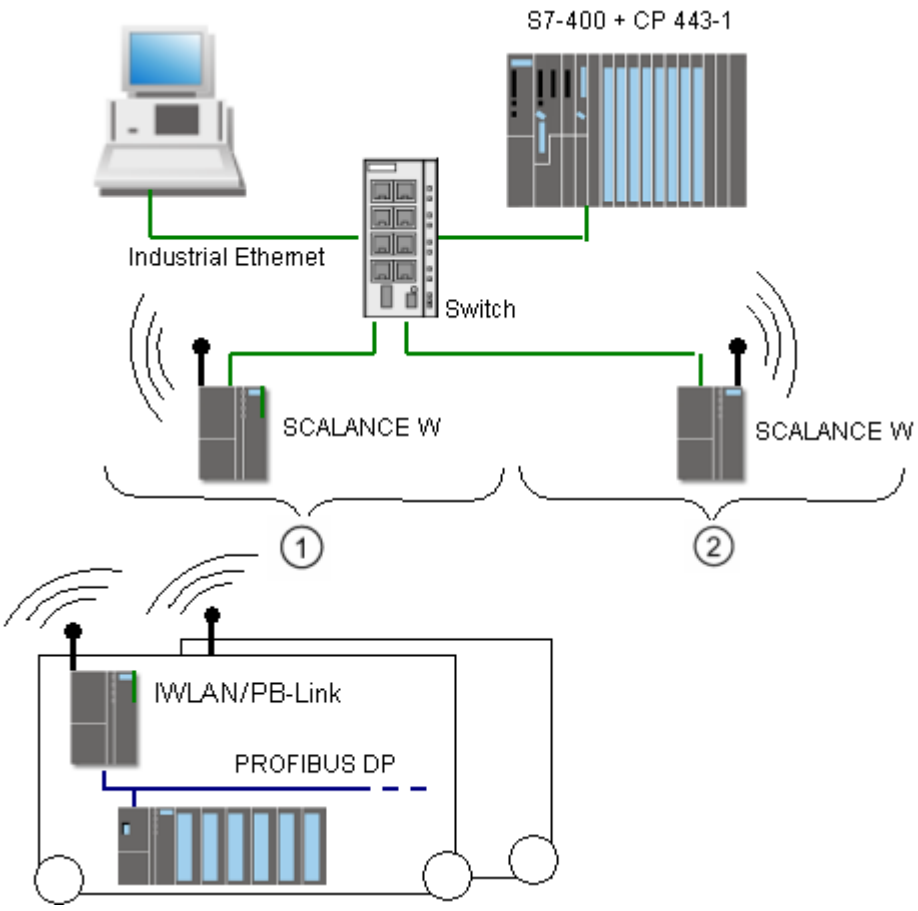
Particularidad

Si hay varios IWLAN/PB-Links dentro de un segmento, éstos deben compartir el ancho de banda disponible para la radiotransferencia. La consecuencia es un aumento del tiempo de actualización para dichos dispositivos.

Ejemplo

En el ejemplo siguiente hay dos dispositivos IO (IWLAN/PB-Links) dentro de un segmento.

Si no hay más de dos IWLAN/PB-Links en un segmento IWLAN al mismo tiempo, introduzca un "2".



① Segmento 1
② Segmento 2

Configurar PROFINET IO

Direccionamiento de dispositivos PROFINET

Asignación de direcciones y nombres para dispositivos PROFINET

En este capítulo se explican las convenciones de direcciones y nombres que se aplican a los dispositivos PROFINET.

Direcciones IP

Todos los dispositivos PROFINET operan con el protocolo TCP/IP y por ello precisan una dirección IP para funcionar con Ethernet.

Las direcciones IP se establecen en las propiedades de los módulos. Si la red forma parte de una red corporativa Ethernet existente, solicite estos datos al administrador de red.

Las direcciones IP de los dispositivos IO se asignan automáticamente y por lo general se asignan a los dispositivos IO durante el arranque de la CPU. Las direcciones IP de los dispositivos IO siempre tienen la misma máscara de subred que el controlador IO, y se asignan en orden creciente empezando por la dirección IP del controlador IO.

Nombre del dispositivo

Para que sea posible acceder a un dispositivo IO desde un controlador IO, es necesario que el dispositivo tenga nombre. En PROFINET se ha elegido este procedimiento porque es más sencillo manejar nombres que direcciones IP complejas.

Tanto los controladores IO como los dispositivos IO tienen un nombre de dispositivo. Con la opción "Generar automáticamente nombre de dispositivo PROFINET" activada, el nombre del dispositivo se deriva automáticamente del nombre configurado del dispositivo (CPU, CP o IM):

- El nombre del dispositivo PROFINET se compone del nombre del dispositivo (p. ej. de la CPU), del nombre de la interfaz (sólo si hay varias interfaces PROFINET) y, opcionalmente, del nombre del sistema IO:
  <nombre de la CPU>.<nombre de la interfaz>.<nombre del sistema IO>
  Este nombre no se puede modificar directamente. El nombre de dispositivo PROFINET se modifica indirectamente, cambiando el nombre de la CPU, del CP o del IM correspondiente en las propiedades generales del módulo. El nombre de dispositivo PROFINET se visualiza también p. ej. en la lista de los dispositivos accesibles. Si desea ajustar el nombre de dispositivo PROFINET independientemente del nombre del módulo, debe desactivar la opción "Generar automáticamente nombre de dispositivo PROFINET".
- Se genera un "nombre convertido" a partir del nombre de dispositivo PROFINET. Se trata del nombre de dispositivo que se carga realmente en el dispositivo.
  El nombre del dispositivo PROFINET se convierte únicamente si no cumple con las exigencias de la IEC 61158-6-10. Este nombre tampoco se puede modificar directamente.

Reglas para el nombre convertido

A continuación se citan las reglas para el nombre convertido. Si el nombre convertido **no** debe distinguirse del nombre del módulo, el nombre del módulo debe cumplir estas reglas.

- El nombre se compone de una o varias etiquetas (inglés: labels) separadas por un punto [.].
- Limitación a 240 caracteres en total (letras minúsculas, cifras, guión o punto)
- Un elemento del nombre de dispositivo, es decir, la cadena de caracteres entre dos puntos, no debe superar los 63 caracteres de longitud.
- Un elemento del nombre está formado por los caracteres [a-z, 0-9].
- El nombre de dispositivo no debe empezar ni acabar por el carácter "-".
- El nombre de dispositivo no debe empezar con una cifra.
- El nombre de dispositivo no debe tener la forma n.n.n.n (n = 0, ... 999).
- El nombre de dispositivo no debe empezar con la cadena de caracteres "port-xyz" o "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0, ... 9).

Ejemplo del nombre de dispositivo

```
device-1.machine-1.plant-1.vendor
```

Si se asigna este nombre a una CPU, p. ej., STEP 7 no lo convierte, porque corresponde a las reglas descritas arriba.

Número de dispositivo

Además del nombre de dispositivo, al insertar un dispositivo IO también se asigna automáticamente un número de dispositivo que se puede cambiar.

Número de dispositivos en una subred PROFINET

En una subred PROFINET se vigila el número máximo permitido de dispositivos durante la configuración.

Consulte también

Asignar el nombre de dispositivo y la dirección IP (Página 1493)

Remanencia de parámetros de dirección IP y nombres de dispositivo (Página 1501)

Asignar el nombre de dispositivo y la dirección IP

Primera asignación de la dirección IP y máscara de subred en un controlador IO

Para ello, existen varias opciones.

En la parametrización de la interfaz PROFINET debe especificarse si la dirección IP se ajustará en el proyecto (es decir, en la configuración hardware) o bien en el dispositivo.

Asignación de una dirección IP	Observaciones
Opción "Ajustar dirección IP en el proyecto": El controlador IO obtiene la dirección IP cargando la configuración hardware, p. ej. por medio de una de las interfaces PROFINET, la interfaz PROFIBUS o la interfaz MPI.	Al cargar la configuración hardware en el controlador IO (p. ej. CPU) se carga también la dirección IP y, en caso de estar ajustado así, el nombre de dispositivo. Ejemplo con la interfaz PROFINET: 1. Conecte la programadora o el PC a la misma red a la que está conectado el dispositivo PROFINET en cuestión. La interfaz de la PG o del PC tiene que estar ajustada a TCP/IP (Auto). 2. Visualice los dispositivos accesibles. 3. Seleccione el dispositivo de destino indicando su dirección MAC y cargue la configuración hardware, incluida la dirección IP configurada (así la dirección IP esta guardada de forma remanente). Si el dispositivo PROFINET dispone de una interfaz MPI o PROFIBUS DP, conecte la programadora o el PC directamente al dispositivo PROFINET a través de la interfaz MPI o PROFIBUS DP. La dirección IP configurada se aplica al cargar la configuración hardware.
Opción "Ajustar dirección IP en el dispositivo": • Asignar online • Asignación mediante programa de usuario (instrucción IP_CONFIG para S7-300/400, T_CONFIG para S7-1200/1500) • Asignar desde pantalla de CPU (S7-1500) • El controlador IO de nivel superior se encarga de la asignación (solo en I-Devices)	Si se elige esta opción en las propiedades de la interfaz PROFINET, la dirección IP se puede asignar desde el editor Online y diagnóstico, con la herramienta Primary Setup Tool o incluso con el programa de usuario (instrucción "IP_CONF"). Esta opción está ajustada automáticamente si en las propiedades del sistema PROFINET IO se ha activado la opción "Sistema IO de múltiples aplicaciones" (proyecto de maquinaria de serie). En la CPU S7-1200, recuerde que el acceso a la CPU no está protegido por contraseña. Si una CPU está protegida contra escritura, no es posible asignar una dirección IP ni tampoco un nombre de dispositivo directamente en el dispositivo.

Puesta en servicio de una interfaz PROFINET

Para más detalles sobre cómo poner en marcha una interfaz PROFINET, consulte las instrucciones de servicio de los dispositivos PROFINET de la familia de dispositivos SIMATIC.

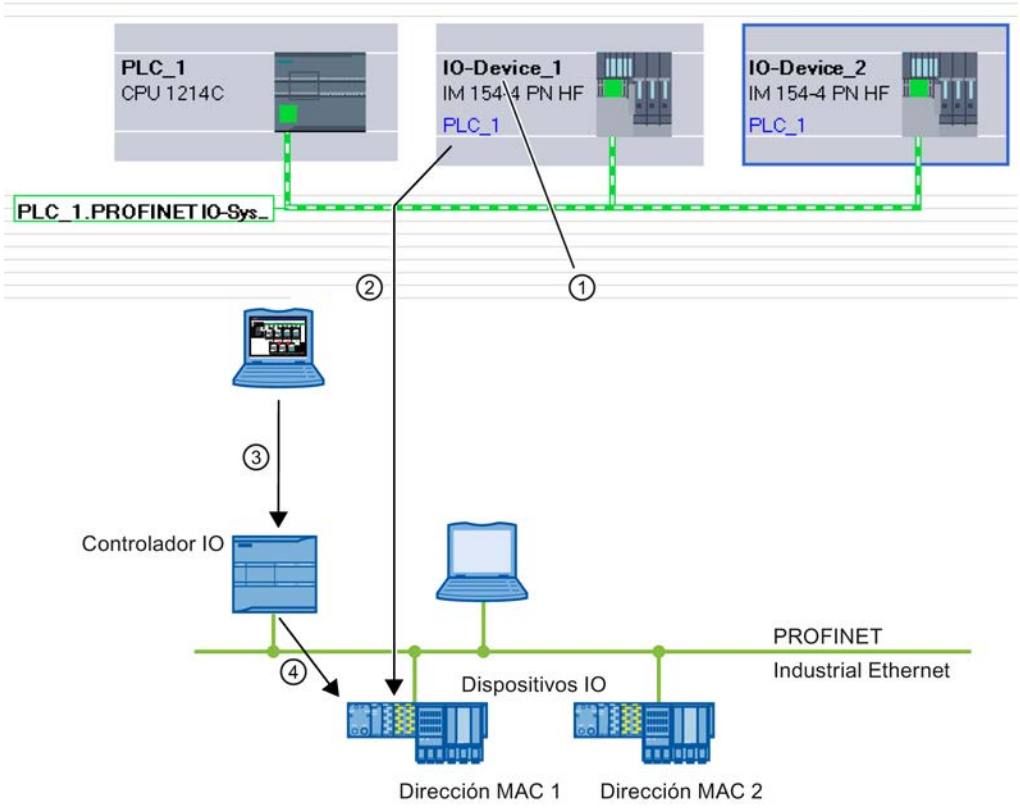
Asignación del nombre de dispositivo para dispositivos IO con la opción "Permitir sustitución de dispositivo sin medio de almacenamiento extraíble" activada

En el caso de controladores IO con la opción "Permitir sustitución de dispositivo sin medio de almacenamiento extraíble" activada, no deben asignarse nombres a los dispositivos IO in situ, p. ej., si se sustituye un dispositivo. Otro caso de aplicación es la puesta en servicio automática, en la que la CPU asigna automáticamente el nombre y los parámetros de dirección IP a los dispositivos IO durante el arranque.

Requisitos: los puertos de los dispositivos están interconectados y los dispositivos en cuestión soportan LLDP. Los dispositivos se han pasado al estado de suministro o, para las CPU S7-1500 a partir de la versión V1.5, la opción "Permitir sobrescribir el nombre del dispositivo PROFINET" está activada en el controlador IO (área "Direcciones Ethernet", apartado "PROFINET" de las propiedades de la interfaz PROFINET).

Asignación de nombre de dispositivo y dirección en un dispositivo IO

La siguiente figura muestra cómo se asignan el nombre del dispositivo y la dirección. Este procedimiento no es válido si la opción "Permitir sustitución de dispositivo sin medio de almacenamiento extraíble" está activada.



① Cada dispositivo recibe un nombre; STEP 7 asigna una dirección IP automáticamente.

② A partir del nombre, STEP 7 genera un nombre de dispositivo PROFINET, que después se asigna online a un dispositivo IO (dirección MAC) y se escribe en el dispositivo.

③ La configuración se carga en el controlador IO.

④ El controlador IO asigna la respectiva dirección IP al dispositivo IO en el arranque con el nombre de dispositivo PROFINET asignado.

Cambiar el nombre de dispositivo y la dirección IP

Es posible modificar manualmente el nombre y la dirección IP con posterioridad. El nombre de dispositivo debe cambiarse primero en la configuración para poder asignarse a

continuación al dispositivo IO a través de la Memory Card u online con la programadora o el PC.

Offline con Memory Card:

1. Guarde los datos configurados (nombre de dispositivo: p. ej. turbo 3) para el dispositivo IO en la Micro Memory Card insertada en la programadora o el PC. Utilice para ello el comando "SIMATIC Card Reader > Escribir nombre del dispositivo en Micro Memory Card" del menú "Proyecto".
2. A continuación inserte la Micro Memory Card en el dispositivo IO. El dispositivo IO aplica automáticamente el nombre configurado.

Online con la PG/el PC:

1. Conecte la programadora o el PC a la subred Ethernet a través de la interfaz PROFINET.
2. Seleccione la subred en la vista de redes y elija el comando "Asignar nombre de dispositivo" del menú contextual.
3. En el cuadro de diálogo "Asignar nombre de dispositivo PROFINET", seleccione la interfaz PG/PC adecuada para conectarse a la subred Ethernet. En la lista desplegable superior se pueden seleccionar todos los nombres de dispositivo PROFINET configurados. Elija allí un nombre de dispositivo PROFINET y, en la tabla inferior, seleccione el dispositivo IO que debe recibir este nombre. Es posible filtrar la visualización de los dispositivos en la tabla según diversos criterios.
4. El botón "Parpadear LED" permite identificar el dispositivo fácilmente.
5. Haga clic en el botón "Asignar nombre".

El controlador IO reconoce el dispositivo IO por su nombre y le asigna automáticamente la dirección IP configurada.

Asignación de la dirección IP para dispositivos IO especiales

Algunos dispositivos IO especiales, como p. ej. SCALANCE X o los CP S7-300, admiten la opción de no asignar la dirección IP desde el controlador IO durante el arranque. En ese caso, la dirección IP debe asignarse por otra vía. Esta opción se denomina "Ajustar dirección IP en el dispositivo". Encontrará más información en el manual de producto PROFINET correspondiente de la familia SIMATIC.

Otro caso especial es la opción "Ajustar dirección IP en el proyecto" en el apartado "Protocolo IP" de las propiedades de direcciones Ethernet de un dispositivo IO. Esta opción está ajustada automáticamente si en el sistema PROFINET IO correspondiente se ha activado la opción "Sistema IO de múltiples aplicaciones" para un proyecto de maquinaria de serie. En este caso, el controlador IO no asignará una dirección IP adaptada hasta que él mismo haya obtenido una dirección IP localmente.

Requisitos para el procedimiento ulterior de asignación de dirección IP y nombre de dispositivo

Si el dispositivo IO, como se ha indicado, no debe obtener la dirección IP o el nombre de dispositivo del controlador IO, proceda del siguiente modo:

1. Seleccione la vista de dispositivos o redes.
2. Abra las propiedades del dispositivo PROFINET en cuestión y seleccione allí el área "Interfaz PROFINET [X1]" > "Direcciones Ethernet".
3. Elija la opción "Ajustar dirección IP en el dispositivo" en "Parámetros IP" o bien la opción "Ajustar nombre de dispositivo PROFINET en el dispositivo" en "PROFINET".

Reglas

Si se utiliza la opción "Ajustar la dirección IP/el nombre de dispositivo en el dispositivo" para un dispositivo PROFINET, tenga en cuenta lo siguiente:

- La parte correspondiente a la subred en la dirección IP del dispositivo IO debe coincidir con la parte correspondiente a la subred en la dirección IP del controlador IO.
- El dispositivo PROFINET en cuestión no se puede utilizar como router.

Consulte también

Asignar un nombre en la vista Online y diagnóstico, abierta desde "Dispositivos accesibles" (Página 2198)

Activar la sustitución de dispositivo sin medio de almacenamiento extraíble (Página 1513)

Ejemplo de asignación del nombre de dispositivo

En este ejemplo se asignan nombres de dispositivo a un controlador PROFINET IO y a un dispositivo PROFINET IO. Los nombres de dispositivo deben incluir también el nombre del sistema PROFINET IO, para facilitar la correspondencia.

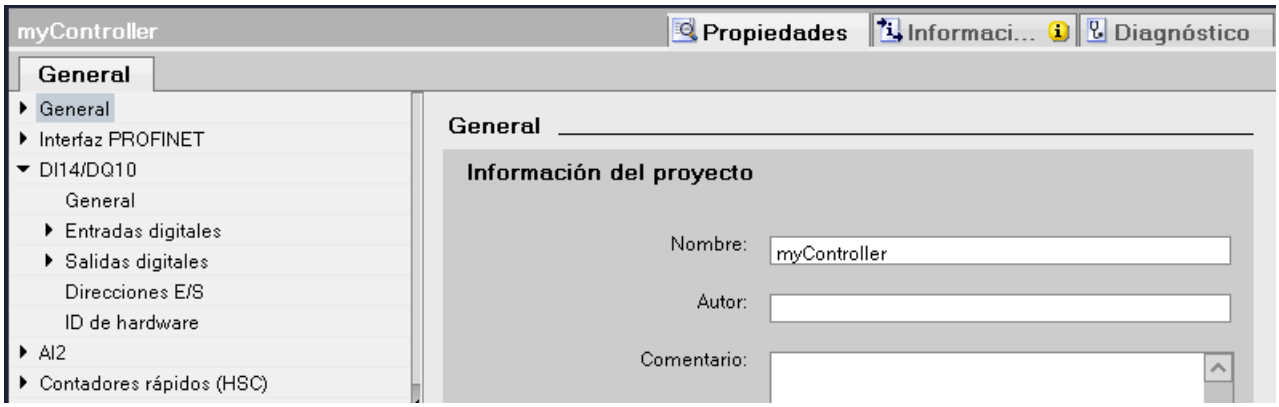
Requisitos

- Se encuentra en la vista de redes.
- Hay una CPU 1214C (a partir de V2.0) en la vista de redes.
- Hay un módulo de interfaz IM 151-3PN.
- Las interfaces PROFINET de ambos módulos están conectadas en red.

Procedimiento

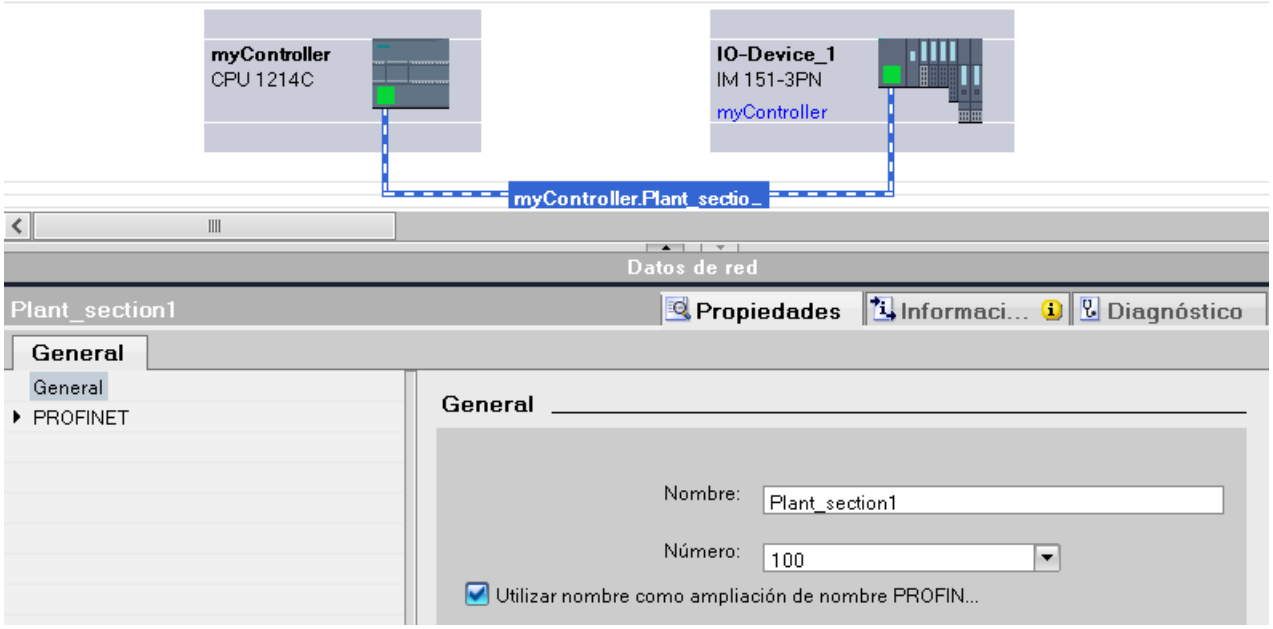
Para asignar los nombres, proceda del siguiente modo:

1. Seleccione la CPU.
   Asegúrese de seleccionar sólo la CPU y no el dispositivo completo.
2. Asigne el nombre "myController" en el área "General" de la ventana de inspección.

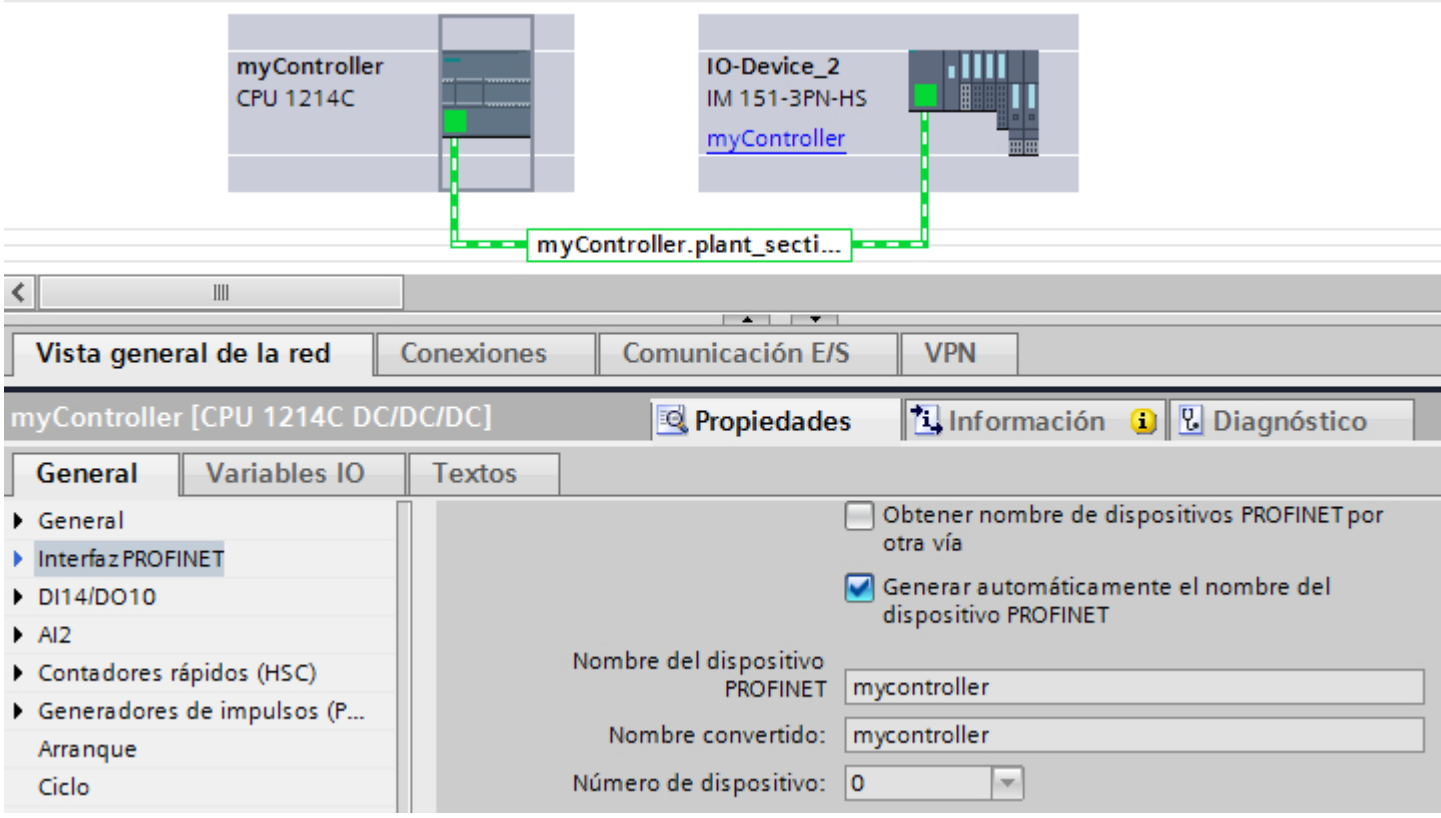


3. Seleccione el módulo de interfaz.
   Asegúrese de seleccionar sólo el módulo de interfaz y no el dispositivo completo ET 200S.
4. Asigne el nombre "Device_1" en el área "General" de la ventana de inspección.
5. Haga clic con el botón derecho del ratón en el sistema PROFINET IO y seleccione el comando "Propiedades".

6. Asigne el nombre "Plant_section1" al sistema IO y active la casilla de verificación "Utilizar nombre como ampliación de nombre PROFINET".



7. Los nombres de dispositivo PROFINET generados automáticamente se encuentran en el área "Interfaz PROFINET" de la ventana de inspección con el dispositivo seleccionado.



El nombre de dispositivo PROFINET corresponde al nombre del módulo (en este caso ampliado con el nombre del sistema IO) con la diferencia de que sólo se utilizan minúsculas. Motivo: al guardar el nombre no se distingue entre mayúsculas y minúsculas ("case insensitive").
Si desea ajustar el nombre de dispositivo independientemente del nombre del módulo, debe desactivar la opción "Generar automáticamente nombre de dispositivo PROFINET". En ese caso, el nombre de dispositivo PROFINET se puede editar.
Debajo se visualiza el nombre convertido. Es el nombre que se genera automáticamente a partir del nombre de dispositivo PROFINET y que cumple con las convenciones DNS. Si se trabaja con STEP 7 este nombre no es necesario. Se visualiza para fines de control y se corresponde con el nombre que se guarda en el dispositivo. Si se trabaja con otras herramientas que son capaces de grabar el intercambio de datos y leer el nombre de dispositivo real, se encontrará el nombre convertido.

Particularidades adicionales

En dispositivos PROFINET con varias interfaces PROFINET, el nombre de la interfaz se adjunta al nombre del módulo separado por un punto.

Ejemplo:

- Nombre del módulo: myController
- Nombre de la interfaz: Interface_1
- Nombre de dispositivo PROFINET: mycontroller.interface_1

Asignar nombre de dispositivo mediante Memory Card

Introducción

Es posible configurar offline el nombre de los dispositivos PROFINET IO. Para ello, guarde un nombre de dispositivo configurado en una Memory Card e insértela a continuación en el dispositivo IO previsto.

Si es necesario sustituir por completo un dispositivo IO en caso de defecto, el controlador IO efectúa automáticamente una parametrización y una configuración del dispositivo nuevo. Con la Memory Card es posible sustituir un dispositivo sin PG.

Requisitos

- La PG dispone de un lector de tarjetas para Memory Cards.
- El dispositivo IO debe soportar la asignación del nombre de dispositivo mediante Memory Card.
- Se ha configurado la estación con su sistema PROFINET IO.

Procedimiento

Para guardar un nombre de dispositivo en una Memory Card, proceda del siguiente modo:

1. Inserte la Memory Card en el lector de tarjetas.
2. Seleccione el dispositivo IO al que debe asignarse el nombre mediante Memory Card.
3. Elija el comando "Card Reader > Escribir nombre del dispositivo en Micro Memory Card" del menú "Proyecto".
   Si la Memory Card no está vacía, se notifica al usuario, que puede borrar la Memory Card antes de guardar.

Remanencia de parámetros de dirección IP y nombres de dispositivo

La remanencia de parámetros de dirección IP (dirección IP, máscara de subred, ajuste del router) y de nombres de dispositivo varía en función de cómo se realiza la asignación de direcciones.

Una asignación temporal no remanente significa:

- Los parámetros de la dirección IP y el nombre del dispositivo siguen siendo válidos durante el período de tiempo siguiente:
  - hasta la siguiente desconexión (POWER OFF)
  - hasta el siguiente borrado total
  - hasta que finaliza la conexión online (p. ej. después de cargar el programa)
    Después de desconexión (POWER OFF) / conexión (POWER ON) o borrado total, solo es posible acceder a la CPU mediante la dirección MAC.

Si los parámetros de dirección IP no son remanentes, después de los eventos citados anteriormente (p. ej. desconexión/conexión) no puede tener lugar ninguna comunicación basada en el protocolo IP.

La asignación de una dirección IP temporal también borra los parámetros de dirección IP guardados de forma remanente.

Asignar parámetros de dirección IP y nombre de dispositivo sin remanencia

Los parámetros de dirección IP y el nombre de dispositivo no son remanentes en los casos siguientes:

- Con la función "Dispositivos accesibles", una dirección IP temporal que no es remanente se asigna implícitamente si el dispositivo (p. ej. CPU) no tiene ninguna dirección IP.
- El dispositivo es un controlador IO "normal" (es decir, que no es I-Device) y en el programa de usuario (instrucción "IP_Conf") está especificado que los parámetros de dirección IP o el nombre del dispositivo no sean remanentes.

Asignar con remanencia parámetros de dirección IP y nombre de dispositivo

Los parámetros de dirección IP y el nombre de dispositivo son remanentes en los casos siguientes:

- En las propiedades de la interfaz PROFINET está ajustado que los parámetros de dirección IP estén configurados en el proyecto (opción "Configurar dirección IP en el proyecto").
- En las propiedades de la interfaz PROFINET se ha definido que la dirección IP se ajuste en el dispositivo.
  - Después de cargar la configuración, con STEP 7 o con una herramienta de configuración como PST se asignan los parámetros de dirección IP o el nombre de dispositivo (STEP 7: función online y de diagnóstico "Asignar dirección IP"). Los parámetros de dirección IP asignados son remanentes.
  - El dispositivo es un controlador IO "normal" (es decir, que no es I-Device) y en el programa de usuario (instrucción "IP_Conf") está especificado que los parámetros de dirección IP o el nombre del dispositivo son remanentes.

Particularidades con I-Device

En las propiedades de la interfaz PROFINET del I-Device se ha definido que los parámetros de dirección IP se ajusten en el dispositivo. Los parámetros de dirección IP para el I-Device se asignan a través del controlador IO de orden superior.

- Si hay ajustado un arranque priorizado, los parámetros de dirección IP son remanentes.
- Si **no** hay ajustado un arranque priorizado, los parámetros de dirección IP no son remanentes.

Recomendación

Siempre que sea posible, utilice la opción "Ajustar la dirección IP en el proyecto" y defina una dirección IP adecuada. En ese caso, la dirección IP está asignada con remanencia.

Restablecimiento de parámetros de dirección IP y nombres de dispositivo remanentes

La función online y de diagnóstico "Restablecer configuración de fábrica" restablece los parámetros de dirección IP y los nombres de dispositivo remanentes.

Nota

Consecuencias de la nueva asignación de parámetros de dirección IP a parámetros de dirección IP existentes

- Mediante la asignación temporal de parámetros de dirección IP / nombres de dispositivo se pueden restablecer parámetros de dirección IP / nombres de dispositivo guardados de forma remanente.
- En el caso de una asignación permanente de parámetros de dirección IP / nombres de dispositivo se sustituyen los parámetros guardados previamente de forma remanente por los parámetros recién asignados.

Nota

Reutilización de dispositivos

Ejecute la función "Restablecer configuración de fábrica" antes de instalar un dispositivo con parámetros de dirección IP o nombre de dispositivo remanentes en otras subredes o instalaciones, o antes de guardarlo en el almacén.

Crear un sistema PROFINET IO

Un sistema PROFINET IO consta de un controlador PROFINET IO y de dispositivos PROFINET IO asignados.

Para crear un sistema PROFINET IO se requiere un controlador IO (p. ej. CPU 1214C) y uno o varios dispositivos IO (p. ej. un módulo de cabecera de la familia de periferia descentralizada ET 200S).

En cuando se conecta un controlador IO con un dispositivo IO, se produce un acoplamiento controlador-dispositivo.

Procedimiento

Para crear un sistema PROFINET IO, proceda del siguiente modo:

1. Arrastre con Drag&Drop un controlador IO del catálogo de hardware (p. ej. CPU 1214C) al espacio libre de la vista de redes.
   El controlador IO se crea en el proyecto.
2. Arrastre con Drag&Drop un dispositivo IO del catálogo de hardware (p. ej. ET 200S) al espacio libre de la vista de redes.
3. Haga clic en la interfaz PROFINET del controlador IO o del dispositivo IO.
4. Con el botón del ratón pulsado, arrastre una conexión de la interfaz seleccionada a la interfaz del interlocutor.
   Se genera una subred con un sistema IO entre controlador IO y dispositivo IO.
5. Dado el caso, adapte las propiedades de la subred Ethernet o del controlador IO (p. ej. dirección IP) en "Propiedades" de la ventana de inspección.

Manejar el sistema PROFINET IO

Los comandos del menú contextual permiten borrar o crear sistemas PROFINET IO en la vista de redes y también conectar la interfaz con otra subred.

De este modo se corrige una configuración PROFINET existente en la vista de redes.

Crear un sistema PROFINET nuevo para un controlador IO

Para crear un sistema PROFINET IO nuevo para un controlador IO, proceda del siguiente modo:

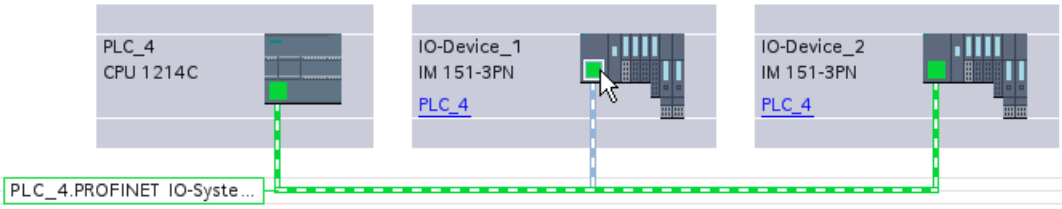
1. Asegúrese de que no hay ningún sistema IO asignado al controlador IO. Si ya hubiera un sistema IO asignado al controlador IO, el comando "Asignar sistema IO" del menú contextual no esta activado.
2. Seleccione la interfaz PROFINET y elija el comando "Asignar sistema IO" del menú contextual.

Se crea un sistema PROFINET IO nuevo en el controlador IO y es posible asignar dispositivos IO a este sistema IO.

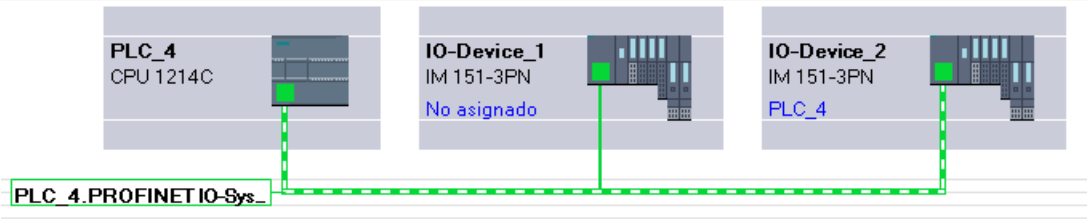
Desconectar dispositivos PROFINET del sistema PROFINET IO

Para desconectar dispositivos PROFINET IO ya conectados en red de su sistema PROFINET IO, proceda del siguiente modo:

1. Haga clic en la interfaz PROFINET de un dispositivo IO.



2. Seleccione el comando del menú contextual "Desconectar sistema IO".
   El dispositivo IO que estaba asignado a este sistema IO ya no lo está posteriormente.



Existe la posibilidad de crear un sistema IO nuevo y asignar los dispositivos IO no asignados a un controlador IO.

Asignar dispositivos PROFINET IO a otros controladores IO

La configuración de los sistemas PROFINET IO existentes se cambia fácilmente en la vista de redes:

1. Seleccione la interfaz de un dispositivo IO y elija el menú contextual. Para ello, existen las posibilidades siguientes:
   - Asignar el dispositivo IO a una subred nueva o desconectarlo de la subred existente
   - Asignar el dispositivo IO a un controlador IO nuevo
   - Asignar el dispositivo IO a un sistema IO nuevo o desconectarlo del sistema IO existente.
2. Para asignar el dispositivo IO a otro controlador IO, seleccione el comando "Asignar a controlador IO nuevo" del menú contextual.
   En caso de que no haya conexión en red, se crea una subred automáticamente y el dispositivo IO se asigna al sistema IO del controlador IO nuevo.

Sugerencia: configuración rápida de sistemas IO

Si el sistema IO tiene muchos dispositivos IO, asigne todos los dispositivos IO posicionados mediante Drag & Drop a un controlador IO en un solo paso.

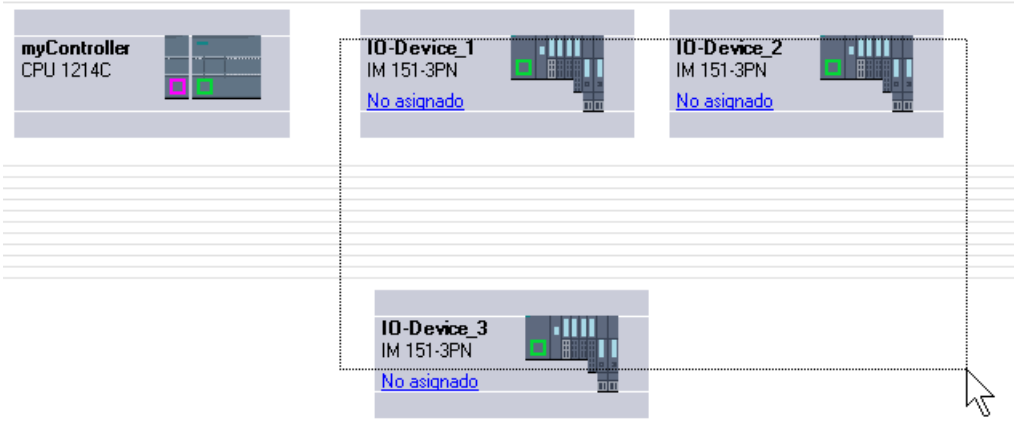
Requisitos

El controlador IO y los dispositivos IO están posicionados en la vista de redes.

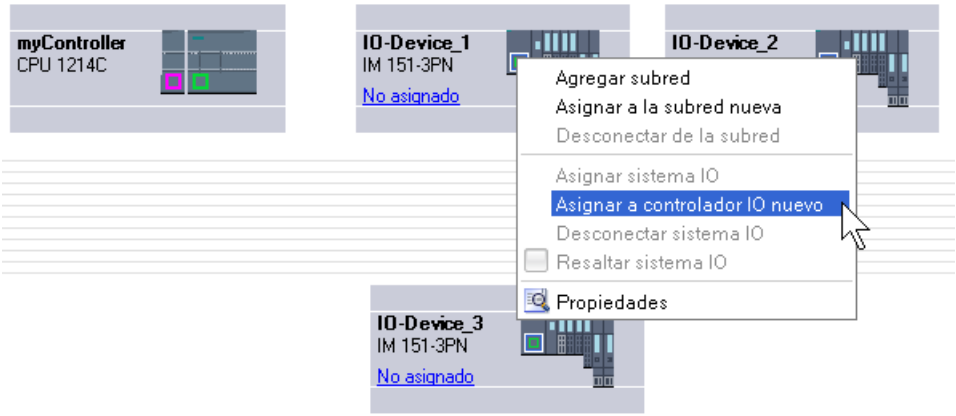
Asignar dispositivos IO a un sistema IO

Proceda del siguiente modo:

1. Elija un factor de zoom apropiado de manera que pueda ver tantos dispositivos IO como sea posible en la vista de redes.
2. Disponga los dispositivos IO en dos filas como máximo.
3. Seleccione todas las interfaces IO con el puntero del ratón (no los dispositivos). Esto sólo resultará si empieza a tirar del puntero del ratón fuera del primer dispositivo IO y suelta el botón del ratón en el último dispositivo IO (selección con lazo).



4. Elija en el menú contextual "Asignar a controlador IO nuevo" y seleccione en el cuadro de diálogo siguiente la interfaz IO correspondiente del controlador IO.



5. Los dispositivos IO se conectan automáticamente en red con el controlador IO y forman conjuntamente un sistema IO.

Nota

Con un sistema IO resaltado puede hacer doble clic en un dispositivo IO en el catálogo de hardware y agregar así rápidamente otros dispositivos IO. De este modo, el dispositivo IO es incluido automáticamente en el sistema IO resaltado.

Interconectar puertos

Cuando un dispositivo IO está asignado a un controlador IO, todavía no se ha especificado cómo están interconectados los puertos.

Una interconexión de puertos no es necesaria para poder utilizar la funcionalidad Ethernet/ PROFINET, pero ofrece las ventajas siguientes:

- Con la interconexión de puertos se especifica una topología teórica. Basándose en una comparación online-offline es posible realizar una comparación teórica-real en los dispositivos que soportan esta funcionalidad.
- Solo en la comunicación IRT: si se ha configurado una interconexión de puertos, STEP 7 puede determinar el ancho de banda necesario con más precisión. Por norma general, esto comporta un aumento del rendimiento.

Asegúrese de que con la interconexión de puertos no se creen estructuras en anillo no permitidas.

Es aconsejable utilizar la interconexión de puertos únicamente para aquellos dispositivos que soportan la configuración de topología.

Interconectar puertos en la ventana de inspección

Para interconectar los puertos, proceda del siguiente modo:

1. Seleccione el dispositivo Ethernet/PROFINET o la interfaz Ethernet/PROFINET.
2. Navegue hasta la propiedad de puerto "Interconexión de puertos".
   Con la interfaz Ethernet/PROFINET seleccionada, este ajuste se encuentra en la ventana de inspección del modo siguiente: Propiedades > General > Opciones avanzadas >Puerto [...] > Interconexión de puertos.
3. En el apartado "Puerto local" se encuentran los ajustes relacionados con el puerto local. Así, p. ej., para conductores de fibra óptica aquí pueden ajustarse las denominaciones de los cables.
   En el punto "Puerto interlocutor" haga clic en el triángulo negro del campo "Puerto interlocutor" para visualizar y seleccionar los puertos interlocutores disponibles.
4. Si se trata de una interconexión de puertos con el medio cobre y los dispositivos soportan la comunicación IRT, también se podrán ajustar la longitud del cable y el tiempo de propagación de señal.

Si la interfaz Ethernet/PROFINET no estaba conectada en red, se conectará automáticamente con esta acción. En las propiedades de la subred es posible especificar si debe utilizarse dicha subred para la conexión en red o no.

Consulte también

Sinopsis (Página 741)

Resumen de las propiedades de la CPU (Página 1931)

Ajustar tiempo de ciclo de emisión

Requisitos para modificar el tiempo de ciclo de emisión en el dispositivo PROFINET

No debe haber configurado ningún IRT (Isochronous Realtime), lo que significa:

- No debe haber configurado ningún dispositivo como esclavo Sync o maestro Sync en el sistema IO.
- Ningún dispositivo del sistema IO debe estar sincronizado.

Si está configurado IRT, es decir, si el controlador IO está configurado como maestro Sync, solo es posible ajustar el tiempo de ciclo de emisión en el dominio Sync.

Procedimiento

Para ajustar el tiempo de ciclo de emisión en el dispositivo PROFINET, proceda del siguiente modo:

1. Seleccione el controlador PROFINET IO en la vista de dispositivos o de redes.
2. Modifique el valor del intervalo de actualización más pequeño posible en las propiedades de la interfaz PROFINET, en "Interfaz PROFINET > Opciones avanzadas > Configuración en tiempo real > Comunicación IO > Tiempo de ciclo de emisión".

El tiempo de ciclo de emisión es válido para todos los dispositivos PROFINET del sistema IO. Si se ajusta la función de sincronización a un valor distinto de "Sin sincronizar", solo es posible ajustar el tiempo de ciclo de emisión en el dominio Sync, es decir, de forma centralizada en el sistema PROFINET IO.

Ajustar el tiempo de actualización

Tiempo de actualización

Dentro de este intervalo de tiempo, el controlador IO/dispositivo IO suministra datos nuevos a un dispositivo IO/controlador IO en el sistema PROFINET IO. El tiempo de actualización se puede configurar por separado para cada dispositivo IO; dicho tiempo determina el intervalo con que se envían los datos del controlador IO al dispositivo IO (salidas), así como del dispositivo IO al controlador IO (entradas).

En el ajuste predeterminado, STEP 7 calcula el tiempo de actualización automáticamente para cada dispositivo IO del sistema PROFINET IO, teniendo en cuenta el volumen de datos que deben intercambiarse y el tiempo de ciclo de emisión ajustado.

Ajustar el tiempo de actualización

Si no se quiere calcular el tiempo de actualización automáticamente, es posible modificar el ajuste.

Para modificar el tiempo de actualización, proceda del siguiente modo:

1. Seleccione en la vista de redes o de dispositivos la interfaz PROFINET del dispositivo IO.
2. Cambie el tiempo de actualización en "Opciones avanzadas > Configuración en tiempo real > Ciclo IO" de las propiedades de la interfaz.
   - Para calcular automáticamente un tiempo de actualización adecuado, seleccione "Automático".
   - Si el propio usuario quiere seleccionar el tiempo de actualización, se selecciona "Ajustable" y se indica el tiempo de actualización deseado en ms.
3. Para garantizar la coherencia entre tiempo de ciclo de emisión y tiempo de actualización, active la opción "Ajustar el tiempo de actualización si se modifica el tiempo de ciclo de emisión".
   Con esta opción se asegura que el tiempo de actualización ajustado no sea más corto que el tiempo de ciclo de emisión.

El ajuste no automático del tiempo de ciclo de emisión puede conducir a errores si el ancho de banda disponible es insuficiente o si se han rebasado otros límites (p. ej. demasiados dispositivos configurados).

Ajustar el tiempo de supervisión de respuesta

Tiempo de supervisión de respuesta

Para dispositivos PROFINET IO es posible configurar un tiempo de supervisión de respuesta.

Si el dispositivo IO no recibe datos de entrada o salida (datos IO) del controlador IO dentro del tiempo de supervisión de respuesta, éste pasa al estado seguro.

El tiempo de supervisión de respuesta no se introduce directamente, sino como "Número de ciclos de actualización aceptados sin datos IO". Esto facilita el ajuste, porque el tiempo de actualización puede ser más breve o más largo dependiendo de la capacidad de rendimiento del dispositivo IO y del ajuste.

A partir del "Número de ciclos de actualización aceptados sin datos IO" se calcula automáticamente el tiempo de supervisión de respuesta resultante.

Configurar tiempo de supervisión de respuesta

Para definir el tiempo de supervisión de respuesta, proceda del siguiente modo:

1. Seleccione en la vista de redes o de dispositivos la interfaz PROFINET del dispositivo IO.
2. Navegue hasta "Opciones avanzadas > Configuración en tiempo real > Ciclo IO" de las propiedades de la interfaz.
3. Seleccione el número deseado de ciclos en la lista desplegable "Lanzar supervisión de respuesta tras # ciclos con datos IO ausentes".

A continuación se calcula automáticamente el tiempo de supervisión de respuesta en base al factor especificado. No debe superar los 1,92 segundos.

Nota

El ajuste estándar sólo se puede modificar en casos excepcionales, p. ej. en la fase de puesta en servicio.

Ancho de banda calculado para datos IO cíclicos

Ancho de banda calculado para datos IO cíclicos

El sistema vigila el mantenimiento del ancho de banda máximo disponible para datos IO cíclicos. El ancho de banda máximo depende del tiempo de ciclo de emisión. Si los tiempos de ciclo de emisión son mayores o iguales a 1 ms, el ancho de banda máximo es de 0,5 ms. Si los tiempos de ciclo de emisión son menores, se reduce el ancho de banda máximo disponible.

El ancho de banda realmente necesario para datos IO cíclicos es definido por el sistema atendiendo al número de dispositivos IO y módulos IO configurados. Además, el ancho de banda necesario depende del tiempo de actualización utilizado.

Por regla general, el ancho de banda calculado aumenta en caso de:

- un número mayor de dispositivos IO
- un número mayor de módulos IO
- tiempos de actualización más cortos.

Ancho de banda máximo para datos IO cíclicos dependiendo del tiempo de ciclo de emisión

La tabla siguiente muestra cómo se comporta el ancho de banda máximo disponible para datos IO cíclicos en función del tiempo de ciclo de emisión:

Tiempo de ciclo de emisión	Ancho de banda máximo para datos IO cíclicos
250 µs – 468,75 µs	<< 125 µs
500 µs – 968,75 µs	= Tiempo de ciclo de emisión / 2
1 – 4 ms	= 500 µs

Definir opciones de puerto

Definición de opciones de puerto

Modificar los ajustes de conexión para el puerto PROFINET IO

En caso necesario, los ajustes de red para el puerto PROFINET IO se pueden modificar. La opción predeterminada es la definición automática de ajustes que, por norma general, permite comunicarse correctamente.

Posibilidades de ajuste del medio de transferencia/dúplex

Se pueden realizar los siguientes ajustes para "Medio de transferencia/dúplex" en función del dispositivo seleccionado:

- Ajuste automático
  Ajuste predeterminado recomendando para el puerto. La configuración de la transferencia con el puerto interlocutor se "negocia" automáticamente. En el ajuste predeterminado se activa también automáticamente la opción "Activar autonegotiation".
- TP/ITP con x Mbits/s dúplex (semidúplex)
  Ajuste de la velocidad de transferencia y del modo dúplex/semidúplex. El efecto depende de la opción ajustada "Activar autonegotiation":
  - Autonegotiation activada
    Es posible utilizar tanto cable cruzado como no cruzado.
  - Autonegotiation desactivada
    Asegúrese de utilizar el cable correcto (cable cruzado o no cruzado). El puerto también se vigila con este ajuste.
- Desactivado
  En función del tipo de módulo, en la lista desplegable aparece la opción "desactivado". Así, por ejemplo, tiene la posibilidad de prohibir el acceso a un puerto no utilizado por motivos de seguridad. Con este ajuste no se generan eventos de diagnóstico.

Opción "Vigilar"

Con esta opción se activa y desactiva el diagnóstico de puerto. Ejemplos de diagnóstico de puerto: se vigila el estado del enlace, es decir, se genera el diagnóstico en caso de fallo del enlace y se vigila la reserva del sistema en los puertos de fibra óptica.

Opción "Activar autonegotiation"

El ajuste de autonegotiation sólo es modificable si se ha seleccionado un medio específico (p. ej. TP 100 Mbits/s dúplex). De las propiedades del módulo depende que un medio específico se pueda ajustar o no.

Si se ha desactivado la autonegotiation, se fuerza el ajuste fijo del puerto, de manera similar a como se requiere p. ej. en caso de arranque priorizado del dispositivo IO.

Hay que procurar realizar los mismos ajustes en el puerto interlocutor, porque en esta opción los parámetros operativos de la red conectada no se reconocen y, en consecuencia, la velocidad de transferencia de datos y el modo de transferencia no se pueden ajustar óptimamente.

Nota

Cuando hay un puerto local interconectado, STEP 7 realiza el ajuste del puerto interlocutor siempre y cuando soporte este ajuste. Si el puerto interlocutor no soporta este ajuste, se genera un mensaje de error.

Consulte también

Normas de cableado con autonegotiation desactivada (Página 1511)

Limitaciones en el puerto (Página 1512)

Normas de cableado con autonegotiation desactivada

Requisitos

Se han realizado los ajustes siguientes p. ej. para acelerar el tiempo de arranque del dispositivo IO para el puerto correspondiente:

- Velocidad de transferencia fija
- Autonegotiation incl. autocrossing desactivada

De ese modo, durante el arranque se prescinde del tiempo de negociación de la velocidad de transferencia.

Si ha desactivado Autonegotiation, debe observar las normas de cableado.

Normas de cableado con autonegotiation desactivada

Los dispositivos PROFINET tienen los dos tipos de puertos siguientes:

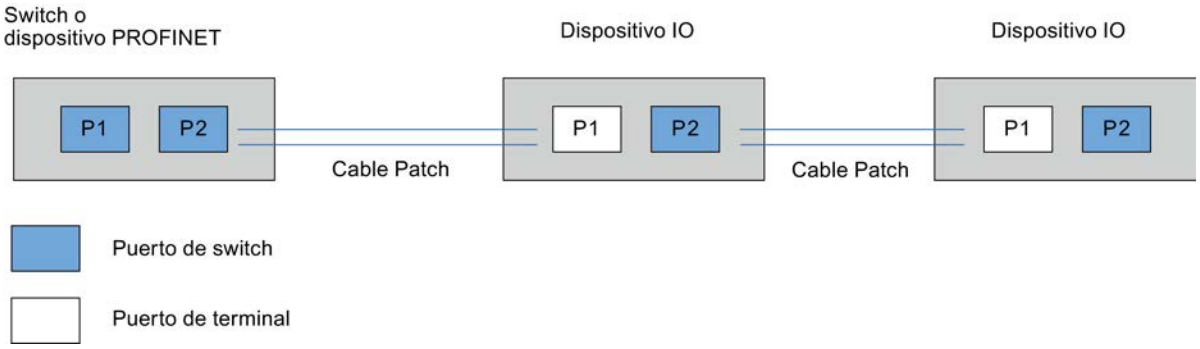
Tipo de puerto	Dispositivos PROFINET	Observación
Switchport con asignación de pines cruzada	En dispositivos IO: puerto 2 En CPUs S7 con 2 puertos: puerto 1 y puerto 2	Asignación de pines cruzada significa que la asignación de pines de los puertos para envío y recepción se intercambian internamente entre los dispositivos PROFINET afectados.
Puerto de terminal con asignación de pines no cruzada	En dispositivos IO: puerto 1 En CPUs S7 con un puerto: puerto 1	-

Validez de las normas de cableado

Las normas de cableado descritas en el siguiente apartado rigen exclusivamente para aquellos casos en los que se ha definido un ajuste de puerto fijo.

Normas para el cableado

Varios dispositivos IO pueden conectarse en línea con un cable de red (ambos conectores con cableado uno a uno). Para ello se conecta el puerto 2 (P2) del dispositivo IO con el puerto 1 (P1) del siguiente dispositivo IO. En el gráfico siguiente se representa un ejemplo con dos dispositivos IO.



Limitaciones en el puerto

Requisitos

Para poder trabajar con "boundaries" (limitaciones), el dispositivo correspondiente debe contar con más de un puerto. Si el dispositivo no soporta limitaciones para PROFINET, los parámetros correspondientes están desactivados.

Activar limitaciones

Se entiende por limitaciones los límites para transferir determinados frames en Ethernet. Es posible activar las siguientes limitaciones en un puerto:

- "Fin del registro de nodos accesibles"
  No se transfieren los frames DCP para el registro de nodos accesibles. Los nodos que están detrás de este puerto ya no se visualizan en "Nodos accesibles" del árbol del proyecto. La CPU ya no puede acceder a dichos nodos.
- "Fin de la detección de topología"
  No se transfieren los frames LLDP (Link Layer Discovery Protocol) para la detección de topología.
- "Fin del dominio Sync"
  No se transfieren frames Sync que se transmiten dentro de un dominio Sync para sincronizar los nodos.
  Si se opera p. ej. un dispositivo PROFINET con más de dos puertos en un anillo, es preciso impedir la inserción de frames Sync en el anillo activando una limitación Sync (en los puertos no incluidos en el anillo).
  Otro ejemplo: si se quieren utilizar varios dominios Sync, hay que configurar una limitación de dominio Sync para el puerto que está conectado con un dispositivo PROFINET del otro dominio Sync.

Restricciones

Hay que tener en cuenta las siguientes restricciones:

- Las casillas de verificación solo están operativas si el puerto soporta la correspondiente característica.
- Si se ha determinado un puerto interlocutor para el puerto, las siguientes casillas de verificación no están operativas:
  - "Fin del registro de nodos accesibles"
  - "Fin de la detección de topología"
- Si la autonegotiation está desactivada, no está operativa ninguna casilla de verificación.

Activar la sustitución de dispositivo sin medio de almacenamiento extraíble

Sustitución de un dispositivo IO sin medio de almacenamiento extraíble

A menudo es necesario sustituir dispositivos IO en los sistemas de automatización. Habitualmente, se asigna un nombre de dispositivo a los dispositivos IO al insertar el medio de almacenamiento extraíble o la programadora. El controlador IO identifica el dispositivo IO mediante estos nombres de dispositivos.

En determinadas condiciones, los dispositivos IO también pueden recibir su nombre sin tener que insertar un medio de almacenamiento extraíble (p. ej. Memory Card) o sin una PG. Para ello, el controlador IO analiza las relaciones de vecindad entre los diferentes dispositivos IO y el controlador IO mediante mecanismos Ethernet (protocolo LLDP; Link Layer Discovery Protocol). A partir de estas relaciones de vecindad, el controlador IO reconoce qué dispositivo IO se ha sustituido y le asigna el nombre de dispositivo configurado.

Requisitos

- Ya se ha configurado la interconexión de puertos.
- Los dispositivos IO que intervienen en el sistema de automatización tienen que permitir la sustitución del dispositivo sin un medio de almacenamiento extraíble (protocolo LLDP). Si algunos dispositivos IO del sistema de automatización no son compatibles con la sustitución del dispositivo sin el medio de almacenamiento extraíble, el dispositivo IO emite el mensaje correspondiente.

Nota

Para sustituir los dispositivos utilice solo dispositivos IO nuevos o vuelva a poner los dispositivos IO parametrizados al estado de suministro antes de la puesta en servicio.

En el caso de las CPU S7-1500 a partir de la versión de firmware V1.5, no es necesario volver a poner los dispositivos IO parametrizados al estado de suministro. La condición para ello es que la opción "Permitir sobrescribir el nombre de dispositivo" esté activada en el controlador IO (área "Direcciones Ethernet", apartado "PROFINET" de las propiedades de la interfaz PROFINET).

Procedimiento

Con el fin de activar la sustitución de un dispositivo IO sin el medio de almacenamiento extraíble, proceda del siguiente modo:

1. Seleccione en la vista de dispositivos o redes la interfaz PROFINET del controlador IO correspondiente.
2. Active en las propiedades de la interfaz en "Configuración avanzada > Opciones de interfaz" la casilla de verificación "Permitir sustitución de dispositivo sin medio de almacenamiento extraíble".

La opción "Permitir sustitución de dispositivo sin medio de almacenamiento extraíble" permite una puesta en servicio automática, es decir, el sistema IO puede ponerse en servicio con los dispositivos IO sin asignarles previamente un nombre de dispositivo.

Consulte también

Asignar el nombre de dispositivo y la dirección IP (Página 1493)

Permitir sobrescribir el nombre del dispositivo PROFINET (Página 1604)

Componentes con la función de sustitución de dispositivo sin medio de almacenamiento extraíble (http://support.automation.siemens.com/WW/view/es/36752540)

Uso de archivos GSD

Archivos GSD para dispositivos IO

Información básica sobre archivos GSD de dispositivos IO

Las propiedades de los dispositivos PROFINET IO no están guardadas en un archivo de texto basado en palabras clave (como en los esclavos PROFIBUS DP), sino en un archivo XML cuya estructura y cuyas reglas son determinadas por un esquema GSDML.

El lenguaje para la descripción de los archivos GSD es GSDML (Generic Station Description Markup Language). Se define con el esquema GSDML.

Un esquema GSDML contiene reglas de validez que permiten p. ej. comprobar la sintaxis de un archivo GSD. Los fabricantes de dispositivos IO de PROFIBUS International compran esquemas GSDML (en forma de archivos de esquema).

Las ampliaciones de funciones en el ámbito de PROFINET IO repercuten en las especificaciones GSDML y el correspondiente esquema. La ampliación de funciones genera una nueva versión de la especificación y del esquema.

Nombres de los archivos GSD para dispositivos IO

El nombre de un archivo GSD para dispositivos IO puede ser, por ejemplo:

"GSDML-V1.0-Siemens-ET200S-20030616.xml"

Componente del nombre	Explicación
GSDML	Cadena de caracteres con la que empieza cada archivo GSD para dispositivos IO
V1.0	Versión del esquema GSDML
Siemens	Fabricante
ET200S	Nombre del dispositivo
20030616	Identificación de la versión (fecha)
.xml	Extensión del archivo

Versiones de los archivos GSD para dispositivos IO

Para los archivos GSD, la información sobre la versión está dividida en dos partes:

Por un lado se indica la versión del esquema GSDML. De este modo se establece qué lenguaje utiliza un archivo GSD.

Por otro lado se indica la versión en forma de fecha. La edición de los archivos GSD se incrementa cuando p. ej. se ha subsanado un error o se ha introducido una ampliación de funciones.

Las ampliaciones de funciones pueden requerir una nueva versión del esquema GSDML. Es posible que una versión nueva del esquema GSDML sólo se soporte de manera limitada.

Instalar archivo GSD

Introducción

En un archivo GSD (archivo de descripción del dispositivo) se encuentran almacenadas todas las propiedades de un dispositivo IO. Si desea configurar un dispositivo IO que no aparece en el catálogo de hardware, es necesario instalar el archivo GSD suministrado por el fabricante. Los dispositivos IO instalados a partir de archivos GSD se muestran en el catálogo de hardware y pueden seleccionarse y configurarse.

Requisitos

- El editor de hardware y redes está cerrado.
- Se tiene acceso a los archivos GSD necesarios en un directorio del disco duro.

Procedimiento

Para instalar un archivo GSD, proceda del siguiente modo:

1. Elija el comando "Instalar archivos de descripción del dispositivo" del menú "Herramientas".
2. Seleccione en el cuadro de diálogo "Instalar archivos de descripción del dispositivo" el directorio en el que se encuentran los archivos GSD.
3. Seleccione uno o varios de los archivos GSD que aparecen en la lista.
4. Haga clic en el botón "Instalar".
5. Para crear un archivo de informe de la instalación, haga clic en el botón "Guardar archivo de informe".
   Los posibles problemas que puedan aparecer durante la instalación pueden repasarse con el archivo de informe.

Los dispositivos IO nuevos instalados con los archivos GSD se encuentran en "Otros dispositivos de campo > PROFINET" del catálogo de hardware.

Nota

La instalación de un archivo GSD no puede deshacerse.

Cambiar la versión de un archivo GSD

Cambiar la versión de un archivo GSD

La versión del archivo GSD para dispositivos IO se puede cambiar:

- Sólo para el dispositivo IO actual
- Para todos los dispositivos IO apropiados del sistema IO
- Para todos los dispositivos IO apropiados de todo el proyecto

Primero se muestran todos los archivos GSD disponibles para el dispositivo IO actual. Los archivos GSD visualizados se distinguen sólo por la versión. El archivo GSD utilizado actualmente está seleccionado.

Requisitos

- Los datos E/S son iguales para todos los dispositivos IO en los que debe modificarse la versión.
- La referencia no ha cambiado.
- El número de submódulos es el mismo.
- Los datos de parametrización no han cambiado.
- En los slots no debe haber ningún módulo o submódulo que no esté aceptado por el nuevo archivo GSD.

Procedimiento

Para modificar la versión de uno o varios dispositivos IO, proceda del siguiente modo:

1. Seleccione el dispositivo IO para cuyo archivo GSD desee modificar la versión.
2. Haga clic en el botón "Modificar versión" de "General > Información del catálogo" en las propiedades del dispositivo IO.
   Se abre el cuadro de diálogo "Modificar versión".
3. Seleccione la revisión GSD que quiera utilizar en la tabla "Versiones disponibles".
4. Elija en "Utilizar la versión marcada para" los dispositivos cuya versión sea necesario cambiar.
   - Sólo para el dispositivo IO actual
   - Para todos los dispositivos IO apropiados del sistema IO
   - Para todos los dispositivos IO apropiados del proyecto
5. Haga clic en el botón "Aplicar".

Configuraciones especiales PROFINET

Particularidades con S7-300/400/1500 y estaciones PC

Asignaciones de direcciones y nombres para dispositivos PROFINET

En este capítulo se explican las convenciones de direcciones y nombres que se aplican a los dispositivos PROFINET.

Direcciones IP

Todos los dispositivos PROFINET operan con el protocolo TCP/IP y por ello precisan una dirección IP para funcionar con Ethernet.

Las direcciones IP se establecen en las propiedades de los módulos. Si la red forma parte de una red corporativa Ethernet existente, solicite estos datos al administrador de red.

Las direcciones IP de los dispositivos IO se asignan automáticamente y por lo general se asignan a los dispositivos IO durante el arranque de la CPU. Las direcciones IP de los dispositivos IO siempre tienen la misma máscara de subred que el controlador IO, y se asignan en orden creciente empezando por la dirección IP del controlador IO.

Nombre del dispositivo

Para que sea posible acceder a un dispositivo IO desde un controlador IO, es necesario que el dispositivo tenga nombre. En PROFINET se ha elegido este procedimiento porque es más sencillo manejar nombres que direcciones IP complejas.

El nombre del dispositivo se deriva automáticamente del nombre configurado del dispositivo (CPU, CP o IM):

- El nombre del dispositivo PROFINET se compone del nombre del dispositivo (p. ej. de la CPU), del nombre de la interfaz (sólo si hay varias interfaces PROFINET) y, opcionalmente, del nombre del sistema IO:
  <nombre de la CPU>.<nombre de la interfaz>.<nombre del sistema IO>
  Este nombre no se puede modificar directamente. El nombre de dispositivo PROFINET se modifica indirectamente, cambiando el nombre de la CPU, del CP o del IM correspondiente en las propiedades generales del módulo. El nombre de dispositivo PROFINET se visualiza también p. ej. en la lista de los dispositivos accesibles. Si desea ajustar el nombre de dispositivo PROFINET independientemente del nombre del módulo, debe desactivar la opción "Generar automáticamente nombre de dispositivo PROFINET".
- Se genera un "nombre convertido" a partir del nombre de dispositivo PROFINET. Se trata del nombre de dispositivo que se carga realmente en el dispositivo.
  El nombre del dispositivo PROFINET se convierte únicamente si no cumple con las exigencias de la IEC 61158-6-10. Este nombre tampoco se puede modificar directamente.

Número de dispositivo

Además del nombre de dispositivo, al insertar un dispositivo IO también se asigna automáticamente un número de dispositivo que se puede cambiar.

Mediante este número de dispositivo se puede identificar un dispositivo IO en el programa de usuario, p. ej. mediante la instrucción "LOG_GEO (Página 7070)". En el programa de usuario el nombre del dispositivo permanece oculto, a diferencia de lo que ocurre con el número de dispositivo.

Número de dispositivos en una subred PROFINET

En una subred PROFINET se vigila el número máximo permitido de dispositivos durante la configuración.

Información importante sobre el tiempo de ciclo de emisión

Tiempo de ciclo de emisión: intervalo de actualización más breve posible

El tiempo de ciclo de emisión indica el periodo entre dos ciclos de comunicación consecutivos. Es el intervalo de emisión más pequeño posible para el intercambio de datos. Los tiempos de actualización calculados son un múltiplo del tiempo de ciclo de emisión.

Requisitos para modificar el tiempo de ciclo de emisión

El tiempo de ciclo de emisión se ajusta en el dispositivo para dispositivos que no están sincronizados; para dispositivos sincronizados (maestro Sync, esclavo Sync), se ajusta en el dominio Sync.

El tiempo de ciclo de emisión no se puede modificar en el dispositivo si en el sistema IO un dispositivo es el esclavo o el maestro Sync. En este caso, solo se muestra el tiempo de ciclo de emisión que se ha seleccionado en las propiedades del dominio de sincronización correspondiente. Para modificarlo tiene que llamar a las propiedades del dominio Sync correspondiente (botón "Configuración del dominio").

Es posible modificar el tiempo de ciclo de emisión en el dispositivo si ninguno de los dispositivos del sistema IO está sincronizado. Puede consultar los tiempos de actualización resultantes de los dispositivos IO individuales en la tabla Comunicación IO o en las propiedades de los dispositivos IO individuales.

Influencia del tiempo de ciclo de emisión en el ancho de banda máximo posible para datos IO cíclicos

En la tabla siguiente se muestra cómo influye el tiempo de ciclo de emisión en el ancho de banda máximo disponible para los datos IO cíclicos:

Tiempo de ciclo de emisión	Ancho de banda máximo para datos IO cíclicos
250 µs – 468,75 µs	<< 125 µs
500 µs – 968,75 µs	= tiempo de ciclo de emisión / 2
1 – 4 ms	= 500 µs

El tiempo de ciclo de emisión influye en el tiempo de actualización en dispositivos IO que tienen activadas las opciones "Automático" y "Ajustable" (con la opción "Adaptar el tiempo de actualización si se modifica el tiempo de ciclo de emisión" activada).

Los dispositivos IO que tienen activada la opción "Ajustar factor" pueden originar incoherencias en la configuración. En caso necesario tiene que adaptar posteriormente el tiempo de actualización de estos dispositivos IO.

Consulte también

Ajustar tiempo de ciclo de emisión (Página 1507)

Ajustar tiempo de ciclo de emisión

Requisitos para modificar el tiempo de ciclo de emisión en el dispositivo PROFINET

No debe haber configurado ningún IRT (Isochronous Realtime), lo que significa:

- No debe haber configurado ningún dispositivo como esclavo Sync o maestro Sync en el sistema IO.
- Ningún dispositivo del sistema IO debe estar sincronizado.

Si está configurado IRT, es decir, si el controlador IO está configurado como maestro Sync, solo es posible ajustar el tiempo de ciclo de emisión en el dominio Sync.

Procedimiento

Para ajustar el tiempo de ciclo de emisión en el dispositivo PROFINET, proceda del siguiente modo:

1. Seleccione el controlador PROFINET IO en la vista de dispositivos o de redes.
2. Modifique el valor del intervalo de actualización más pequeño posible en las propiedades de la interfaz PROFINET, en "Interfaz PROFINET > Opciones avanzadas > Configuración en tiempo real > Comunicación IO > Tiempo de ciclo de emisión".

El tiempo de ciclo de emisión es válido para todos los dispositivos PROFINET del sistema IO. Si se ajusta la función de sincronización a un valor distinto de "Sin sincronizar", solo es posible ajustar el tiempo de ciclo de emisión en el dominio Sync, es decir, de forma centralizada en el sistema PROFINET IO.

Consulte también

Configurar PROFINET IO con IRT (Página 1527)

Ajuste de tiempos de ciclo de emisión impares

Las CPUs S7-1500 soportan tiempos de ciclo de emisión pares e impares.

Tiempos de ciclo de emisión pares: 0,25 ms, 0,5 ms, 1 ms, 2 ms, ...

Tiempos de ciclo de emisión impares: 0,375 ms, 0,75 ms, 1,125 ms, ...

Para dispositivos IO que no soportan el tiempo de ciclo de emisión rápido de la CPU (es decir, del controlador IO) o que no soportan tiempos de ciclo de emisión impares, la CPU "adapta" el tiempo de ciclo de emisión de los dispositivos. P. ej., puede ocurrir que la CPU (controlador IO) trabaje con un tiempo de ciclo de emisión de 875 µs, que otro dispositivo IO conectado trabaje igualmente con 875 µs, y que otro lo haga con 2 ms.

Reglas

- La adaptación del tiempo de ciclo de emisión solo funciona con dispositivos IO no sincronizados (clase RT: "RT").
- Para dispositivos IO sincronizados (clase RT: "IRT") se aplica un tiempo de ciclo de emisión unificado que se configura en el dominio Sync. El tiempo de actualización corresponde en este caso al del ciclo de emisión.
- Para dispositivos IO que tengan configurada la opción de tiempo de actualización "Automático", STEP 7 configura un tiempo de actualización mínimo de 2 ms, es decir, con esta opción no son posibles tiempos de actualización menores de 2 ms.

Ejemplo

Un sistema maestro PROFINET IO consta de los componentes siguientes:

- Controlador IO: CPU 1516-3 PN/DP (6ES7 516-3AN00-0AB0)
- Dispositivo IO 1: ET200MP (6ES7 155-5AA00-0AB0) con un módulo DI
- Dispositivo IO 2: ET200S (6ES7 151-3BA23-0AB0 V6.0) con un módulo DI

El dispositivo IO 2 no soporta el tiempo de ciclo de emisión impar (aquí: 0,875 ms).

1. Asigne los dispositivos IO al controlador IO.
2. Para el dispositivo IO 1 ajuste la función de sincronización "Esclavo Sync" (ventana de inspección, área Interfaz PROFINET > Opciones avanzadas > Configuración en tiempo real > Sincronización).
3. Para el controlador IO, configure de igual modo la función de sincronización "Maestro Sync".
4. Interconecte los puertos del controlador IO y del dispositivo IO 1 (p. ej. en la vista topológica).
5. El dispositivo IO 2 permanece sin sincronizar; está configurada la opción de tiempo de actualización "Automático".
6. En el dominio Sync (propiedades de la subred), configure el tiempo de ciclo de emisión de 0,875 ms. Todos los dispositivos del dominio Sync tendrán ese tiempo de ciclo de emisión o ese tiempo de actualización.

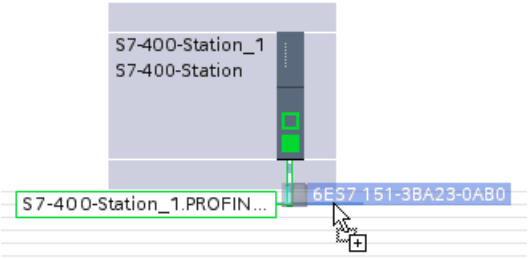
El dispositivo IO 2 no sincronizado recibe automáticamente un tiempo de actualización de 2 ms.

Configuración con un controlador IO externo

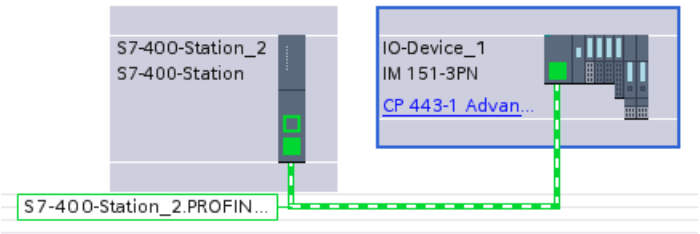
Configuración con un controlador IO externo

Los CP, que se pueden usar como controladores IO externos, soportan gran variedad de posibilidades de comunicación y, como consecuencia, no están "predeterminados" para su uso como controladores IO.

Si utiliza un controlador IO externo (p. ej., CP 443-1 Advanced), deberá agregar un sistema IO después de enchufar el controlador IO.



Los dispositivos IO deseados se pueden arrastrar desde el catálogo de hardware y soltar en este sistema PROFINET IO.



Configuración con la estación PC SIMATIC

Configuración con estaciones SIMATIC PC

Una "estación PC" es un PC con tarjetas de comunicación y componentes de software en una solución de automatización con SIMATIC.

Gracias al uso de tarjetas de comunicación y componentes de software apropiados es posible utilizar una estación PC como controlador PROFINET IO.

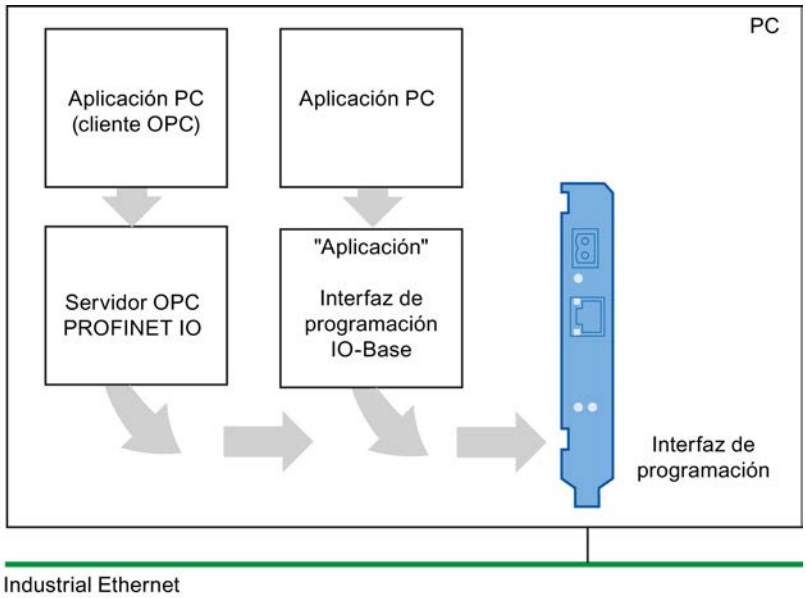
Las aplicaciones PC en la estación PC tienen las siguientes posibilidades de acceso al controlador PROFINET IO:

- Como cliente OPC a través del servidor OPC PROFINET IO
- Directamente mediante la interfaz de usuario PROFINET IO (interfaz de programación IO-Base)

En un momento dado las aplicaciones PC sólo pueden usar una de estas posibilidades de acceso (secuencia Open/Close).

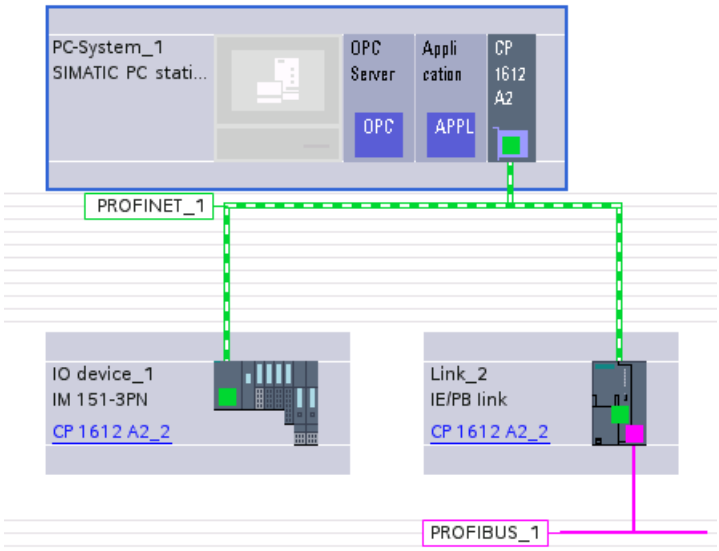
Funciones	Servidor OPC	Interfaz de programación Base-RTE
Leer y escribir datos IO	Sí	Sí
Leer y escribir registros	Sí	Sí
Recibir y acusar alarmas	No	Sí

En la imagen siguiente se muestra una estación PC con los componentes descritos.



Configuración en la vista de redes

En la imagen siguiente se muestra una estación PC con un servidor OPC y una aplicación PC. A través de la interfaz Ethernet insertada de un CP 1612, la estación PC se transforma en un controlador IO con dispositivos IO conectados. Además, se ha utilizado IE/PB-Link como transición de red de PROFINET a PROFIBUS.



Routing IO con CP 1616 / CP 1616 onboard / CP 1604

CP como router PROFINET IO: significado práctico

Puede utilizar una estación PC con el CP 1616 y el CP 1604 como router PROFINET IO utilizando el CP correspondiente al mismo tiempo como dispositivo PROFINET IO y como controlador PROFINET IO. Con este routing IO se intercambian datos de proceso entre dos sistemas PROFINET IO. Si existe la configuración correspondiente, el CP puede garantizar que, por ejemplo, un elemento del plano de control tenga acceso a determinadas entradas o salidas del nivel de proceso.

En una configuración típica, el CP desempeña al mismo tiempo las funciones siguientes:

- Dispositivo PROFINET IO para un controlador PROFINET IO superior del plano de control
- Controlador PROFINET IO con dispositivos IO conectados para el nivel de proceso inferior

Asignar área de transferencia de configuración "TM" (mapeo del módulo de transferencia)

En el área "Modo de operación" de la configuración seleccione "Dispositivo IO".

Acto seguido aparece el área de parametrización adicional "Comunicación de dispositivo I", que permite definir la configuración correspondiente para el routing IO. Una vez allí en la tabla "Área de transferencia" se asignan las áreas de datos para el intercambio de datos al controlador IO superior.

En el caso que nos ocupa existen las opciones siguientes en la columna "Tipo".

- CD: Dispositivo controlador
  Seleccione esta opción para definir un área de transferencia entre el controlador IO y el dispositivo IO. Observe en este punto el resto de informaciones que aparecen en la Ayuda en pantalla del área "Configuración de PROFINET IO".
- TM: Mapeo del módulo de transferencia
  Seleccione esta opción para el caso de aplicación del router IO. En la ventana de inspección dentro del área "Modo de operación / Comunicación de dispositivo I" aparece un área de parametrización adicional con el nombre que se acaba de definir. En esta área de parametrización puede verse la asignación del mapeo del módulo de transferencia en la tabla de mapeo.

Mapeo del módulo de transferencia

Los módulos de transferencia representan los datos enrutados hacia el sistema IO superior desde el sistema IO inferior.

Registre los dispositivos IO del sistema IO inferior de la tabla de mapeo, según corresponda.

La tabla contiene las columnas siguientes:

Parámetro	Significado
TM-Offset	Offset relativo al primer bit en el módulo de transferencia. Los datos tienen el formato <byte>.<bit> Ejemplo: 0.5
Dispositivo/módulo (slot)	Datos del dispositivo IO. El cuadro de diálogo muestra los dispositivos IO configurados en el sistema IO.
Tipo IO	Seleccione el tipo IO en la lista desplegable.
Principio	Dirección inicial (Offset) en el área de datos del dispositivo IO. Los datos tienen el formato <byte>.<bit> Ejemplo: 0.0
Fin	Dirección final (Offset) en el área de datos del dispositivo IO. Los datos tienen el formato <byte>.<bit> Ejemplo: 8.0

Diagnóstico de dispositivos PROFINET

Diagnóstico de dispositivos PROFINET

En PROFIBUS DP dispone de los mismos procedimientos de diagnóstico que con PROFINET IO. El procedimiento es idéntico. El análisis de la información de diagnóstico mediante las instrucciones avanzadas varía ligeramente para la periferia descentralizada en el programa de usuario de una CPU S7-300/400.

En PROFINET IO se aplica una estructura de cualquier fabricante para los registros con información sobre el diagnóstico. La información sobre el diagnóstico se genera sólo para los canales averiados.

Mediante las listas de estado del sistema (SZL) y las instrucciones avanzadas "RALRM (Página 6745)" y "RDREC (Página 6735)" puede consultar el estado de los sistemas PROFINET IO y la información sobre el diagnóstico en un programa de usuario S7:

- Con el fin de obtener una visión general del estado del sistema PROFINET IO, lea p. ej. la lista de estado del sistema 0x0X91 con "RDSYSST (Página 6963)".
- Para leer los registros de diagnóstico (Records) directamente de un módulo averiado, utilice la instrucción avanzada "RDREC" (leer registro). Podrá consultar toda la información del error relacionada con el estado.
- Para leer registros de diagnóstico relacionados con un evento (es decir, causados por el OB de error), use en el OB de error correspondiente la instrucción avanzada "RALRM" (leer la información adicional de la alarma).

Las instrucciones avanzadas "RALRM" y "RDREC" se pueden utilizar también con PROFIBUS DP.

Puede consultar aquí (http://support.automation.siemens.com/WW/view/es/19289930) la información sobre las listas de estado del sistema y los registros de diagnóstico que se han definido para PROFINET IO y la estructura de estos registros.

Configuración de la comunicación IRT

Introducción: Isochronous Realtime Ethernet

Introducción: Isochronous Realtime Ethernet

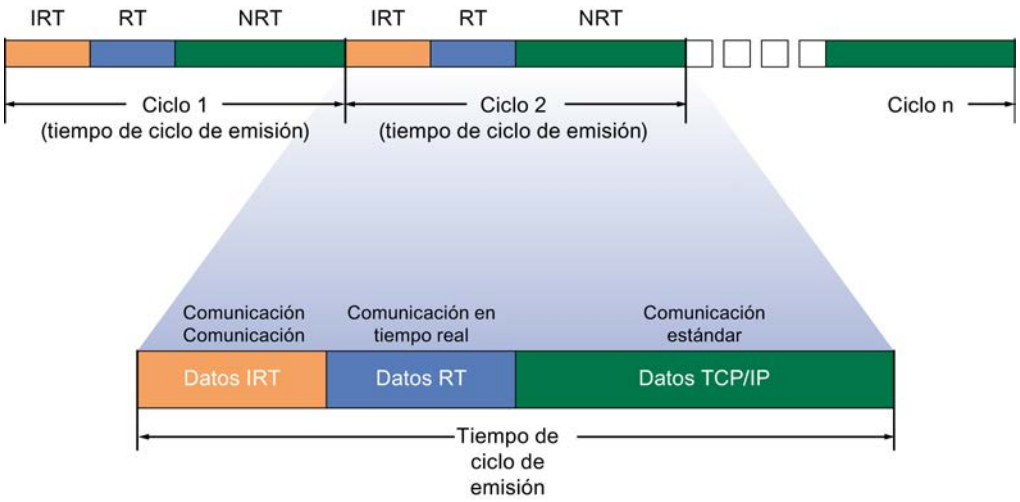
IRT es un procedimiento de transferencia en el que se sincronizan los dispositivos PROFINET con extrema precisión.

Un maestro Sync marca el tiempo de ciclo con el que se sincronizan los esclavos Sync. Un controlador IO y un dispositivo IO pueden tener la función del maestro Sync.

El maestro y los esclavos Sync siempre forman parte de un dominio Sync. En un dominio Sync se reserva el ancho de banda para la comunicación IRT. Es posible la comunicación Real Time y Non Real Time (comunicación TCP/IP) fuera del ancho de banda reservado.

Intervalos de tiempo del ciclo de comunicación

El ciclo de comunicación se divide en tres intervalos de tiempo que se muestran en el gráfico siguiente:



- Datos IRT (comunicación sincronizada)
  Puede reservar este intervalo de tiempo en función del tiempo de ciclo de emisión en pasos determinados. En este intervalo de tiempo se transmiten únicamente datos IRT.
- Datos RT (comunicación Real Time)
  En este intervalo de tiempo se transmiten los datos RT cíclicos. Los datos RT se priorizan en comparación con los datos TCP/IP "normales". Los datos TCP/IP o los telegramas Ethernet pueden tener una prioridad de 1 a 7. Los datos RT tienen la prioridad 6.
- Datos TCP/IP (comunicación estándar)
  La comunicación estándar (TCP/IP, etc.) se transmite en el intervalo restante del ciclo de comunicación.

Ámbitos de aplicación de IRT

PROFINET con IRT es especialmente adecuado para:

- Alto rendimiento y determinismo con grandes cantidades de datos en lo que respecta a la comunicación de datos de usuario (datos productivos)
- Alto rendimiento también con muchos dispositivos con topología en línea en lo que respecta a la comunicación de datos de usuario (datos productivos)
- Transmisión en paralelo de datos productivos y datos TCP/IP a través de una línea, incluso si el número de datos es muy elevado, con garantía de la transmisión de los datos productivos mediante reserva del ancho de banda para la transmisión.

Consulte también

Procedimiento básico para la configuración de IRT (Página 1527)

Configurar PROFINET IO con IRT (Página 1527)

Dominio Sync (Página 1529)

Procedimiento básico para la configuración de IRT

Procedimiento para la configuración de IRT

Si quiere ampliar a IRT en una instalación con PROFINET IO, realice los tres pasos siguientes:

1. Configure los dispositivos PROFINET IO. Los dispositivos PROFINET deben ser compatibles con IRT.
2. Defina el maestro Sync con el que deben sincronizarse los demás dispositivos. Tiene que configurar un dominio Sync con un maestro Sync y, al menos, un esclavo Sync.
3. Cargue la configuración en los dispositivos que intervienen.

Crear una configuración de PROFINET IO para IRT

Para la configuración de IRT es necesario que exista una configuración PROFINET IO, es decir, debe haber configuradas una o más estaciones con controlador IO y dispositivos IO.

En la Task Card "Información" puede ver qué componentes son adecuados para la comunicación IRT, tan pronto como haya seleccionado los componentes correspondientes en el catálogo de hardware.

Sólo se reúnen los requisitos para una comunicación IRT que funciona si se configura un sistema PROFINET IO con componentes compatibles con IRT.

Configurar PROFINET IO con IRT

Introducción

Para aumentar la precisión del intercambio cíclico de datos con IRT en PROFINET IO, configure en primer lugar los dispositivos PROFINET IO. Estos dispositivos PROFINET deben

ser compatibles con IRT. Seguidamente, defina el maestro Sync con el que deben sincronizarse los demás dispositivos. Para ello se debe configurar un dominio Sync con un maestro Sync y, al menos, un esclavo Sync.

Requisitos

- Se dispone de un sistema IO con un controlador IO y, por lo menos, un dispositivo IO.
- Todos los dispositivos que intervienen son compatibles con IRT.

Procedimiento

Con el fin de configurar IRT para un sistema IO existente, proceda del siguiente modo:

1. Seleccione la interfaz PROFINET del **controlador IO**.
2. En la ventana de inspección, vaya a "Opciones avanzadas > Configuración en tiempo real > Sincronización"
3. En "Función de sincronización", asigne al controlador IO la función de maestro Sync.
4. A continuación, seleccione las interfaces PROFINET del **dispositivo IO** correspondiente.
5. En la ventana de inspección, vaya a "Opciones avanzadas > Configuración en tiempo real > Sincronización".
6. Una vez allí, active la clase RT "IRT". Seguidamente, y de manera automática, se asigna al dispositivo IO la función de sincronización "Esclavo Sync".
7. El botón "Dominio Sync" permite comprobar y corregir la configuración en todo momento.
8. Interconecte los puertos entre los esclavos Sync y el maestro Sync.

O

1. En la vista de redes, resalte el sistema PROFINET IO.
2. Haga clic en el sistema PROFINET IO.
3. En la ventana de inspección, vaya al dispositivo del dominio Sync deseado.
4. Efectúe todos los ajustes necesarios directamente en las tablas:
   - Seleccione un sistema IO.
   - Ajuste la función de sincronización "Maestro Sync" en el controlador IO.
   - En el dispositivo IO, cambie la clase RT a "IRT". Con ello se asigna automáticamente al dispositivo IO la función de sincronización "Esclavo Sync".
5. Interconecte los puertos entre los esclavos Sync y el maestro Sync.

Ahora puede cargar la configuración con PROFINET IRT en los dispositivos que intervienen.

Nota

Asignación automática de un maestro Sync

También es posible configurar simplemente un I-Device o un dispositivo IO como esclavo Sync cambiando la clase RT de RT a IRT. El controlador IO asignado al I-device o dispositivo IO se configura automáticamente como maestro Sync.

Si ha configurado el controlador IO como esclavo Sync, este ajuste no se sobrescribirá. El controlador IO no cambia automáticamente de esclavo Sync a maestro Sync.

Si se han asignado varios controladores IO al dispositivo IO como Shared Device, el controlador IO se seleccionará automáticamente como maestro Sync que tiene asignada la interfaz del dispositivo IO.

Dominio Sync

Dominio Sync

Para la sincronización de dispositivos PROFINET IO, es necesario un dominio Sync. El dominio Sync garantiza que todos los dispositivos que pertenecen a este dominio se comuniquen con la misma base de tiempo.

Para la comunicación IRT se requiere un tiempo de ciclo de sincronización para todos los dispositivos PROFINET en un dominio Sync para la distribución de una base de tiempo común. Con esta sincronización básica se logra la ejecución simultánea del ciclo de transmisión de los dispositivos PROFINET en un dominio Sync. El maestro Sync (generalmente, un controlador IO) genera el tiempo de sincronización común y define la base de tiempo con la que se sincronizan los demás esclavos Sync (generalmente, dispositivos IO).

Si falla un maestro Sync, la comunicación de los dispositivos IRT retorna a calidad RT.

Procesos en segundo plano durante la configuración de un sistema IO

Si configura un controlador IO y lo conecta en red con una subred Ethernet, automáticamente se asignará al dominio Sync predeterminado de la subred Ethernet. El dominio Sync predeterminado siempre está disponible. El controlador IO funciona sin sincronizar.

Si asigna al sistema IO del controlador IO otro dispositivo IO, el dispositivo IO se asignará automáticamente al dominio Sync del controlador IO.

Consulte también

Trabajar con dominios Sync (Página 1530)

Trabajar con dominios Sync

Trabajar con dominios Sync

Todos los componentes que forman parte de la comunicación IRT tienen que pertenecer a un dominio Sync.

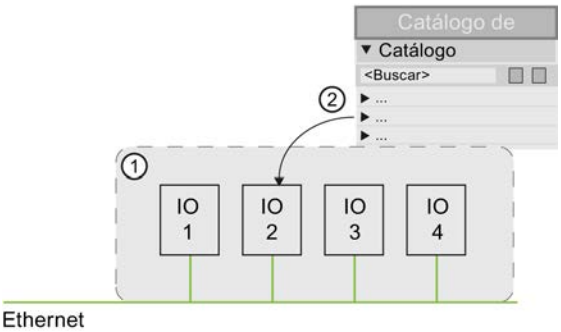
Un dominio Sync es un grupo de dispositivos PROFINET que se sincronizan con un tiempo de ciclo común. Un dispositivo tiene la función del maestro Sync (generador de impulsos) y los demás tienen la función de esclavos Sync.

Dominio Sync predeterminado

Si ha creado una subred Ethernet, se crea automáticamente un dominio Sync especial: el dominio Sync predeterminado. Todos los dispositivos PROFINET que configura para la subred Ethernet pertenecen de forma automática a este dominio Sync.

Las figuran siguientes explican esta relación:

① Dominio predeterminado; siempre disponible



①	Dominio predeterminado; siempre disponible
②	Los sistemas PROFINET IO recién configurados se encuentran primeramente en el dominio predeterminado.

Editar las propiedades de los dominios Sync

Editar las propiedades de los dominios Sync

A continuación se explica cómo definir los nombres y el tiempo de ciclo de emisión de un dominio Sync. Además puede aprender cómo se define un dominio Sync como dominio predeterminado.

Requisitos

- En el proyecto se ha creado al menos un dispositivo IO.
- Ya existe al menos un dominio Sync.

Edición de un Dominio Sync

Para cambiar el nombre de un dominio Sync, adaptar el tiempo de ciclo de emisión o definirlo como dominio predeterminado, proceda del siguiente modo:

1. Seleccione la subred Ethernet en la vista de redes en la que se ha creado el dominio Sync.
2. Abra las propiedades de la subred Ethernet.
3. Seleccione en la navegación local la entrada "Administración de dominios > Dominios Sync > <Nombre del dominio Sync>".
4. Realice los cambios:
   - Introduzca el nombre deseado del dominio Sync en el campo de entrada "Nombre del dominio Sync".
   - Seleccione el tiempo de ciclo de emisión en la lista desplegable "Tiempo de ciclo de emisión".
   - Active la casilla de verificación "Dominio predeterminado" si quiere definir un dominio Sync como el dominio predeterminado.

Anchos de banda para las diferentes clases RT

Junto a la configuración de nombre y tiempo de ciclo de emisión, se muestran también los anchos de banda disponibles y los anchos de banda calculados para las diferentes clases RT dentro de un dominio Sync.

Para ello, seleccione la sección "Detalles" en la navegación local dentro del nombre del dominio Sync.

Definir los dispositivos de un dominio Sync

Definir los dispositivos de un dominio Sync

Todos los componentes que forman parte de la comunicación IRT tienen que pertenecer a un dominio Sync. A continuación se explica cómo definir los dispositivos de un dominio Sync.

Requisitos

Ya están configurados los sistemas PROFINET IO que se encuentran en el dominio predeterminado.

Incluir un dispositivo nuevo en el dominio Sync

Para asignar dispositivos a un dominio Sync, proceda del siguiente modo:

1. Seleccione la subred Ethernet en la vista de redes en la que se ha creado el dominio Sync.
2. Abra las propiedades de la subred Ethernet.
3. Seleccione en la navegación local la entrada "Administración de dominios > Dominios Sync > <Nombre del dominio Sync> > Dispositivos".
4. En la tabla "Estación / Sistema IO" haga clic en la tabla "Agregar dispositivos".
   Se abre el cuadro de diálogo "Seleccionar sistema IO".
5. Seleccione todos los dispositivos que quiere asignar al dominio Sync y haga clic en "Aceptar".

Los sistemas IO seleccionados se eliminan del dominio Sync original y se asignan al dominio Sync actual.

Eliminar un dispositivo de un dominio Sync

Para eliminar un dispositivo de un dominio Sync, proceda del siguiente modo:

1. Seleccione la subred Ethernet en la vista de redes en la que se ha creado el dominio Sync.
2. Abra las propiedades de la subred Ethernet.
3. Seleccione en la navegación local la entrada "Administración de dominios > Dominios Sync > <Nombre del dominio Sync> > Dispositivos".
4. Seleccione en la tabla "Estación / Sistema IO" el dispositivo que quiere eliminar del dominio Sync y presione <Supr>.

Se borra el dispositivo y se vuelve a asignar al dominio predeterminado.

Configuración del maestro Sync redundante

Para dispositivos PROFINET IO en modo IRT es posible configurar un maestro Sync redundante.

Ejemplo: una CPU S7-1500 puede ser configurada como maestro Sync y un Scalance 6GK5 201-3BH00-2BA3 V5.0 puede ser configurado como maestro Sync redundante.

Funcionamiento

Un maestro Sync redundante pertenece al mismo dominio Sync que el maestro Sync "primario". En caso de fallo del maestro Sync primario, impide que falle la comunicación en el dominio Sync.

Tanto el maestro Sync como el maestro Sync redundante envían telegramas Sync. Un esclavo Sync se sincroniza con el telegrama Sync del maestro Sync. Si el maestro Sync falla, automáticamente un esclavo Sync se sincroniza con los telegramas Sync del maestro Sync redundante.

Reglas

- La función de maestro Sync redundante está asignada a un solo dispositivo del dominio Sync.
- En el dominio Sync hay solo dispositivos IRT con la opción "IRT de alto rendimiento".
- Entre el maestro Sync y el maestro Sync redundante hay como máximo 2 dispositivos.

Procedimiento

Para configurar un maestro Sync redundante, proceda del modo siguiente:

1. En la vista de dispositivos o de redes seleccione el dispositivo que desee configurar como maestro redundante.
2. En "Interfaz PROFINET > Configuración avanzada > Configuración en tiempo real > Sincronización > Función de sincronización", seleccione la opción "Maestro Sync redundante".

Reglas y notas sobre la configuración de IRT

Las siguientes reglas y notas se refieren a configuraciones de PROFINET. Aquí (Página 1488) encontrará otras recomendaciones de configuración para la optimización del sistema PROFINET.

Configurar PROFINET con IRT

Tenga en cuenta las reglas siguientes para instalar y utilizar un sistema PROFINET IO en el modo IRT. Estas reglas sirven para garantizar el funcionamiento correcto del sistema PROFINET IO.

- Si se utiliza IRT, es necesario configurar la topología. Para ello, se calculan con exactitud el tiempo de actualización, el ancho de banda y otros parámetros.
- Si quiere usar varios dominios Sync, configure un límite del dominio Sync para el puerto que está conectado con un dispositivo PROFINET de los demás dominios Sync.
- En un dominio Sync sólo puede configurar un maestro Sync.
- Un sistema PROFINET IO sólo puede pertenecer a un único dominio Sync.
- Si configura dispositivos PROFINET en un dominio Sync y quiere sincronizarlos con IRT, los dispositivos PROFINET en cuestión deben ser compatibles con la comunicación IRT.
- Si es posible, utilice el mismo dispositivo PROFINET como controlador PROFINET IO y maestro Sync.
- Si se sincroniza sólo una parte de los dispositivos PROFINET de un sistema PROFINET IO, tenga en cuenta lo siguiente: Coloque fuera del dominio Sync los dispositivos PROFINET que no participen en la comunicación IRT.

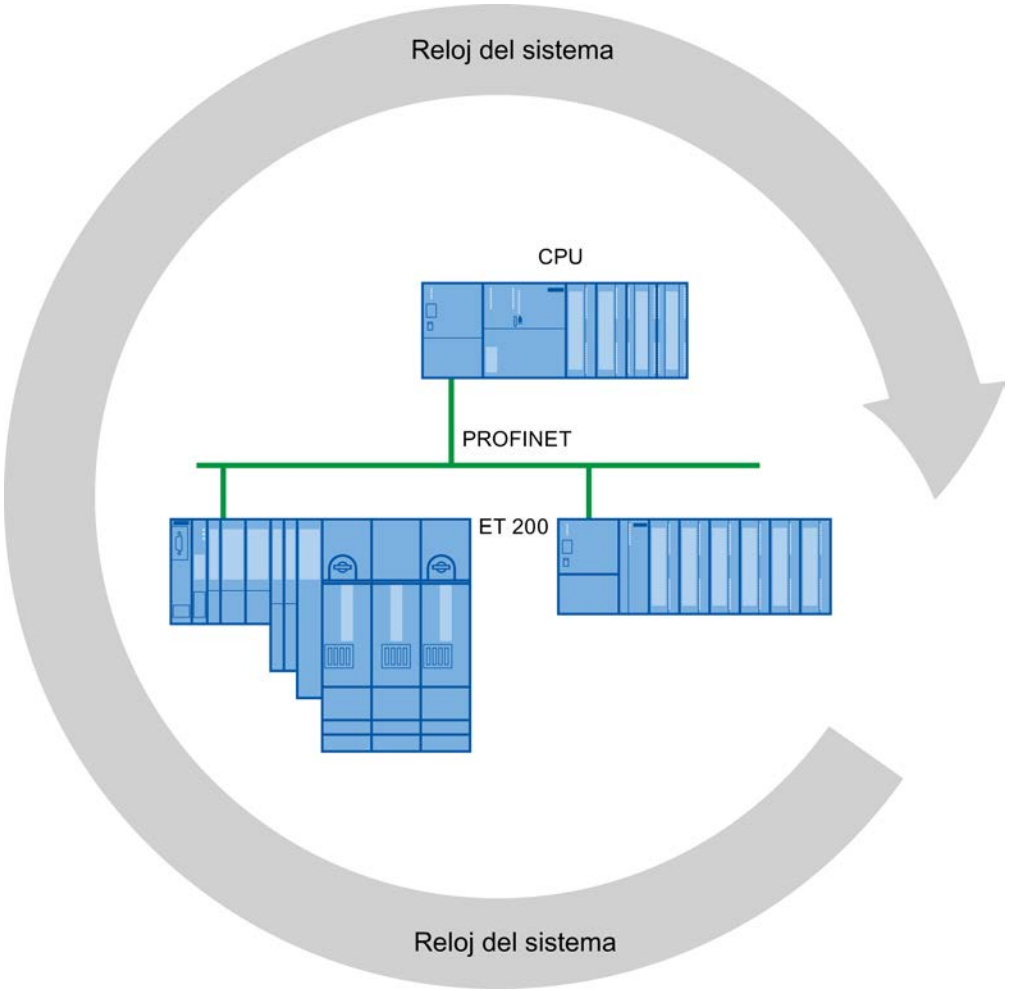
Configuración del modo isócrono

¿Qué es el modo isócrono?

¿Para qué sirve el modo isócrono?

Si el transporte público de cercanías circulara tan rápido como fuera posible y los tiempos de parada se redujeran al mínimo más absoluto, los pasajeros a menudo sólo verías los faroles de cola rojos. Sin embargo, la duración total del viaje viene determinada por los ciclos correspondientes del tren, autobús o metro, pues todo va mejor con unos ciclos bien sintonizados. Lo mismo es válido para la técnica de automatización descentralizada. No sólo cuentan los ciclos rápidos, sino que la sintonización y sincronización de los diferentes ciclos consiguen el caudal óptimo.

Just in Time



El tiempo de respuesta rápido y fiable de un modo isócrono se basa en que todos los datos se ponen a disposición Just in Time. Para ello, el ciclo PROFINET IO equidistante marca el tiempo de ciclo.

Empleo del modo isócrono

Empleo del modo isócrono

Con la propiedad de sistema "Modo isócrono" pueden detectarse valores medidos y datos de proceso en un ciclo de reloj del sistema fijo. En el mismo ciclo de reloj del sistema se produce el procesamiento de señales hasta la conmutación al "borne de salida". De esta manera el modo isócrono contribuye a lograr una gran calidad del control, y por tanto a mejorar la precisión de fabricación. Con el modo isócrono se reducen drásticamente las posibilidades de oscilación de los tiempos de reacción del proceso. El procesamiento en tiempo asegurado puede utilizarse para un ciclo de máquina mayor.

En general, el modo isócrono puede utilizarse donde sea necesario registrar sincrónicamente valores medidos, coordinar movimientos y definir y realizar simultáneamente reacciones de procesos.

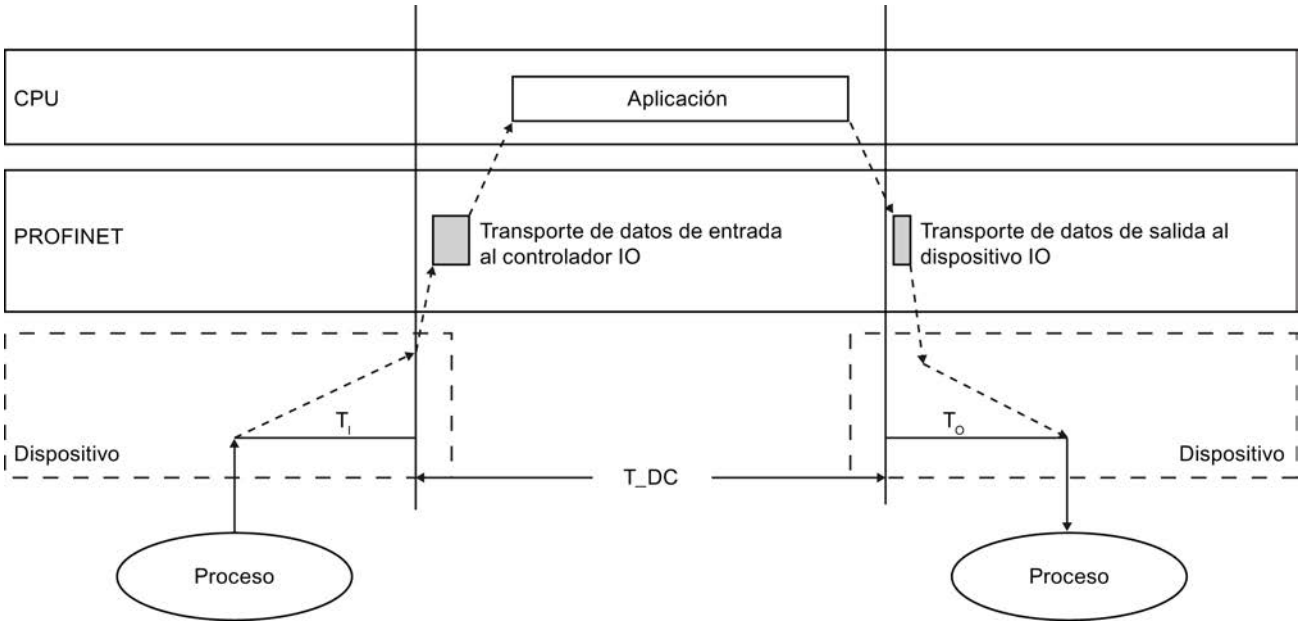
Secuencia temporal de la sincronización

Desde la lectura de los datos de entrada hasta la emisión de los datos de salida

A continuación va a explicarse el principio del proceso de ejecución de todos los componentes involucrados en la sincronización:

- Lectura isócrona de los datos de entrada
- Transporte de los datos de entrada vía subred PROFINET hasta el controlador IO (CPU)
- Reprocesamiento en la aplicación isócrona de la CPU

- Transporte de los datos de salida vía subred PROFINET hasta el dispositivo IO de salida
- Emisión isócrona de los datos de salida



T_DC	Ciclo de datos (Time_DataCycle)
T_I	Tiempo de lectura de los datos de entrada
T_O	Tiempo para emitir los datos de salida

Para que todos los datos de entrada estén preparados para ser transportados por la línea PROFINET IO en el momento de empezar el próximo ciclo PROFINET IO correspondiente, el ciclo de lectura de la periferia empieza anticipadamente con un tiempo de preprocesamiento T_I. T_I es el "flash" para las entradas; en ese instante se leen todas las entradas sincronizadas. T_I es necesario para compensar la conversión analógico-digital, los tiempos de bus de fondo y otros elementos similares. El tiempo de preprocesamiento T_I puede ser configurado por STEP 7 o por el usuario. Es recomendable dejar que STEP 7 determine automáticamente el tiempo de preprocesamiento T_I.

La línea PROFINET IO transporta los datos de entrada al controlador IO. Una vez transcurrido un tiempo de retardo parametrizable se llama el OB de alarma de sincronismo (OB 61, OB 62, OB 63 u OB 64). El programa de usuario en el OB de alarma de sincronismo determina la reacción del proceso y prepara los datos de salida a tiempo para el comienzo del próximo ciclo de datos. La longitud del ciclo de datos es configurada siempre por el usuario.

T_O es la compensación debida al bus de fondo y a la conversión digital-analógica dentro del dispositivo. T_O es el "flash" para las salidas; en ese instante se emiten las salidas sincronizadas. El tiempo T_O puede ser configurado por STEP 7 o por el usuario. Es recomendable dejar que STEP 7 determine automáticamente el tiempo de preprocesamiento T_O.

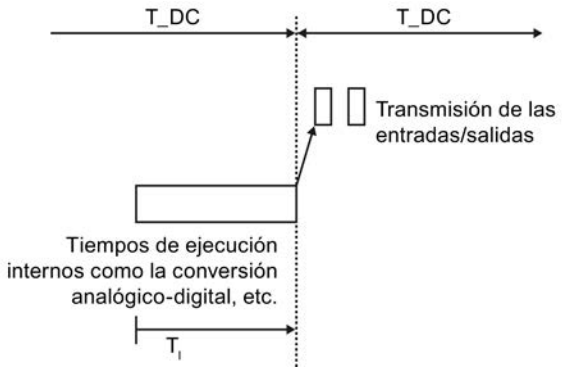
Periferia descentralizada isócrona y no isócrona

Es posible combinar periferia descentralizada isócrona con periferia descentralizada no isócrona en un controlador IO.

El valor Ti

Efecto de T_I

El efecto de $T_I$se dilucida en la figura siguiente:



T_DC	Ciclo de datos
T_I	Tiempo de lectura de los datos de entrada

Proceso

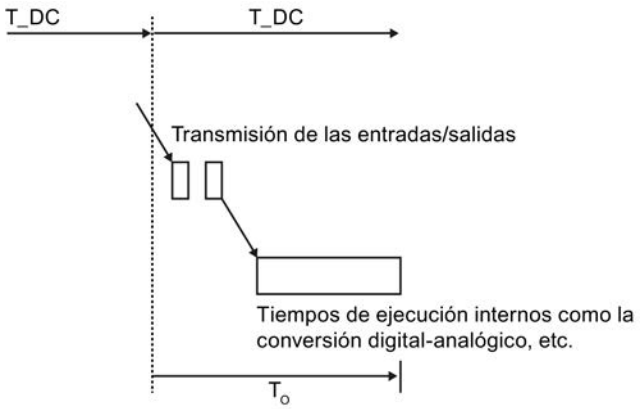
Para que, en el momento de iniciarse el nuevo tiempo de ciclo del sistema, sea posible transferir al controlador IO un estado coherente de las entradas, el proceso de lectura debe avanzarse en el tiempo T_I. Para un módulo de entrada determinado, el tiempo T_I abarca, como mínimo, el tiempo de preparación y conversión de señales en los módulos electrónicos y el tiempo de transferencia al módulo de interfaz en el bus de fondo del dispositivo IO.

En la instalación se consigue que los valores se lean simultáneamente poniendo el T_I de todos los módulos de entrada leídos en modo isócrono al mismo valor, que es mayor o igual al T_I mínimo más grande de todos los módulos de entrada isócronos. Con la configuración predeterminada, STEP 7 asegura de que se ajuste un T_I mínimo común.

El valor To

Efecto de T_O

El efecto se dilucida en la siguiente figura:



T_DC	Ciclo de datos
T_O	Momento de emisión de los datos de salida

Proceso

Para que, en el momento de iniciarse el reloj del sistema, sea posible transferir al proceso un estado coherente de las salidas, la emisión desde el borne no se produce hasta el momento T_O después del último toque. El tiempo T_O abarca para un módulo de salida determinado, como mínimo, el tiempo de transferencia del controlador IO al dispositivo IO (vía PROFINET IO) y en el dispositivo IO la transferencia de las salidas del módulo de interfaz al módulo electrónico (bus de fondo) (en alguna ocasión más el tiempo para la conversión digital-analógico).

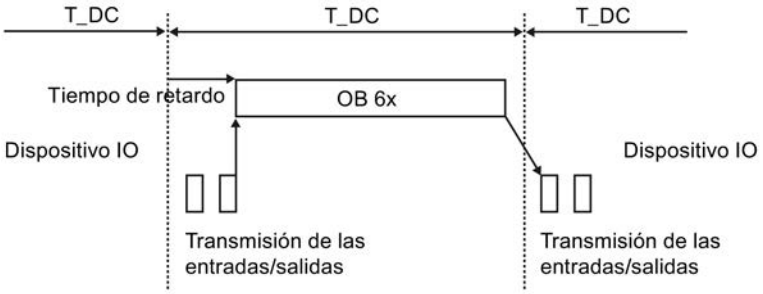
Para que se escriban los valores simultáneamente en la instalación debe ajustarse el T_O de todos los módulos de salidas isócronos al mismo valor. Este valor debe ser superior o igual al T_O mínimo más grande de todos los módulos de salida isócronos. STEP 7 calcula automáticamente el T_O común mínimo posible.

El OB de alarma de sincronismo (OB 6x)

Para conectar en modo isócrono el programa de usuario utilice el OB de alarma de sincronismo (p. ej. OB 61).

Llamada del OB de alarma de sincronismo

El OB de alarma de sincronismo se dispara a través del ciclo de reloj del sistema; en concreto, a través de un tiempo de retardo configurable posterior al ciclo de reloj del sistema. El efecto se dilucida en la siguiente figura:



T_DC Ciclo de datos

Tiempo de retardo

STEP 7 calcula primero automáticamente un valor adecuado para el tiempo de retardo. El tiempo de retardo compensa el tiempo de transferencia de las entradas de los I-devices isócronos al controlador IO a través de la red PROFINET IO. La ejecución del OB 6x está relacionada con el ciclo de reloj del sistema y el tiempo de retardo configurado. Si es necesario, el usuario puede corregir manualmente el tiempo de retardo. Por ejemplo, puede reducirse el tiempo de retardo para ejecutar desde el momento del arranque del OB 6x partes del programa que no acceden a la periferia isócrona. De este modo se gana tiempo de ejecución para el programa de usuario isócrono.

Conexión de la periferia isócrona al OB de alarma de sincronismo

Se accede a la periferia isócrona a través de la memoria imagen parcial de proceso, es decir, las direcciones de los módulos isócronos deben estar en una memoria imagen parcial de proceso.

Los accesos a la periferia isócrona deben programarse mediante las instrucciones SYNC_PI y SYNC_PO dentro del OB de alarma de sincronismo.

Llame la instrucción SYNC_PI al inicio del OB 6x, siempre que se haya seleccionado la configuración automática del tiempo de retardo. Llame la instrucción SYNC_PO al final del OB 6x.

TD_C: ciclo de aplicación y tiempo de ciclo de emisión

El tiempo TC_C designa el tiempo de un ciclo de datos, es decir, el tiempo mínimo posible para la lectura, procesamiento y emisión de los datos (modelo EVA). TC_C es siempre un múltiplo del tiempo de ciclo de emisión.

Si para un dispositivo IO está activado IRT, TC_C es igual al tiempo de ciclo de emisión. El tiempo de actualización se corresponde con el tiempo de ciclo de emisión.

El tiempo de aplicación es un múltiplo del ciclo de datos. Puede ser idéntico al ciclo de datos si el tiempo de ejecución del OB de alarma de sincronismo es suficientemente corto.

Ejecución del programa según el modelo EVA con ciclo corto

La característica principal del modelo EVA con factor de ciclo de aplicación = 1 es que los datos de E/S se procesan completamente dentro de un ciclo de reloj del sistema T_DC. Con este modelo se logran los tiempos de reacción más cortos.

Procesamiento del programa según el modelo EVA con tiempo largo

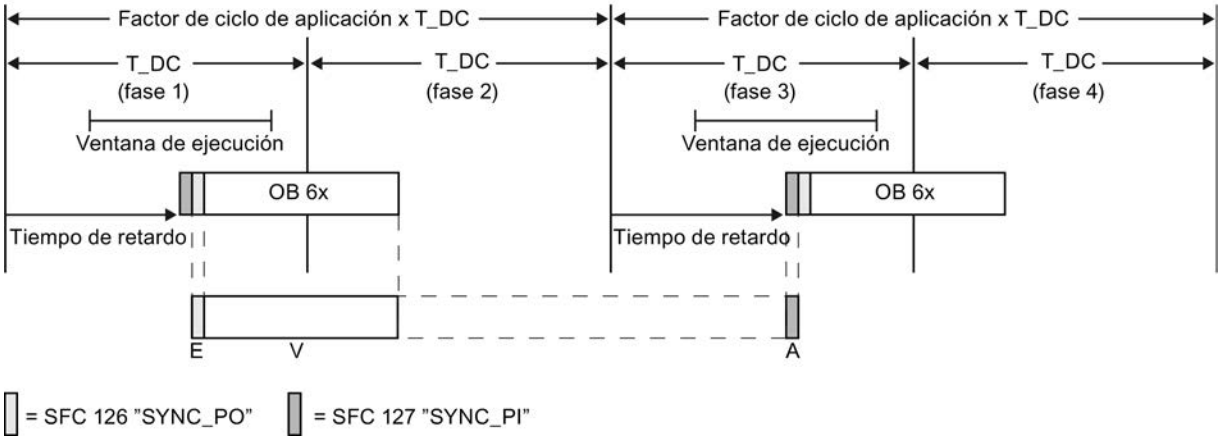
Distintivo del modelo EVA con factor de ciclo de aplicación > 1

Si el tiempo de ejecución del OB de sincronismo es mayor que un ciclo de datos T_DC, la salida debe retardarse un ciclo de aplicación (OB 6x). Por ese motivo, en este modelo EVA la salida se efectúa antes de la entrada.

A continuación se deben llamar las instrucciones SYNC_PI y SYNC_PO en la ventana de ejecución del primer ciclo de datos (sólo está disponible en ella si el factor de ciclo de aplicación > 1).

Se denomina ventana de ejecución el periodo en que es posible llamar las instrucciones SYNC_PI o SYNC_PO; es decir, desde que se han transferido los datos cíclicos mediante PROFINET hasta justo antes de que termine el ciclo de datos T_DC.

El procesamiento restante del programa de usuario en el OB 6x se realiza en el ciclo o los ciclos de datos posteriores.



La figura muestra el transcurso de las señales con factor de ciclo de aplicación = 2 (ciclo de aplicación = 2 x ciclo de datos) desde el registro (E) hasta la salida (A) de los valores del proceso, pasando por el procesamiento (V) en el controlador IO. STEP 7 calcula el tiempo de retardo. Durante este tiempo, los datos de entrada leídos están de camino en PROFINET IO.

Este modelo es especialmente adecuado para periferias de configuración extensa con un programa de usuario extenso en el OB 6x. Este modelo permite tiempos de cálculo más largos para procesar los datos de entrada y determinar los datos de salida correspondientes.

El tiempo de procesamiento constante desde el "borne de entrada" hasta el "borne de salida" es T_I + (factor de ciclo de aplicación + 1) x T_DC + T_O. Es posible asegurar T_I + (2 x factor de ciclo de aplicación + 1) x T_DC + T_O como tiempo de reacción del proceso.

Configuración del modo isócrono

Introducción

A continuación va a describirse cómo configurar el modo isócrono de un módulo utilizando como dispositivos IO las nuevas familias de periferia descentralizada ET 200MP. El procedimiento descrito también es válido para otros sistemas de periferia descentralizada (p. ej. ET 200S o ET 200SP).

El controlador IO es una CPU S7-1500.

Requisitos

- La vista de redes está abierta.
- Hay instalada una CPU S7-1500 (p. ej. CPU 1516-3 PN/DP).
- Hay instalado un dispositivo de periferia descentralizada (p. ej. IM 155-5 PN HF) y está conectado en red con la CPU.
- Se cumplen todos los requisitos para una configuración IRT, es decir:
  - Los puertos de la CPU y del IM están interconectados entre sí;
  - El IM tiene configurado "IRT" como clase "RT" (área "Opciones avanzadas > Configuración en tiempo real > Sincronización").

Procedimiento

Para crear una conexión isócrona de la periferia y el programa de usuario, proceda del siguiente modo:

1. Seleccione el dispositivo de periferia descentralizada (el IM 155-5 PN HF) en la vista de redes y cambie a la vista de dispositivos.
2. Inserte un módulo de periferia que pueda funcionar en modo isócrono (módulo con la extensión de nombre "HF", p. ej. DI 16X24VDC HF - 6ES7 521-1BH00).
3. En la ventana de inspección del módulo de periferia seleccionado, desplácese hasta el área "Direcciones E/S".
4. En el área "Direcciones E/S" realice la configuración siguiente:
   - Active la opción "Modo isócrono".
   - Seleccione una memoria imagen parcial de proceso, p. ej. Memoria imagen parcial de proceso 1.
   - Haga clic en la lista desplegable "OB asignado" y después haga clic en el botón "Agregar objeto".
     Se abre un cuadro de diálogo para seleccionar los módulos de organización.
   - Seleccione el OB "Synchronous Cycle" y confirme con "Aceptar".
     Si hay asignación automática de números, se genera y abre el OB 61. En la ventana de inspección está seleccionada el área "Modo isócrono", lo que permite continuar la configuración del ciclo de aplicación y del tiempo de retardo y comenzar la programación del OB en la parte de instrucciones.

5. Agregue otros dispositivos IO que necesite, conéctelos en red e interconéctelos, y adecue la configuración y los ajustes del modo isócrono.
6. Para acceder a informaciones sobre los anchos de banda calculados o para adaptar el tiempo de ciclo de emisión, seleccione la subred y desplácese hasta el área correspondiente de administración del dominio Sync en la ventana de inspección.

Configuración del modo isócrono

Introducción

A continuación va a describirse cómo configurar el modo isócrono de un módulo utilizando como dispositivos IO las nuevas familias de periferia descentralizada ET 200MP. El procedimiento descrito también es válido para otros sistemas de periferia descentralizada (p. ej. ET 200S o ET 200SP).

El controlador IO es una CPU S7-300/400.

Requisitos

- La vista de redes está abierta.
- Hay instalada una CPU S7-300/400 (p. ej. CPU 319-3 PN/DP).
- Hay instalado un dispositivo de periferia descentralizada (p. ej. IM 155-5 PN HF) y está conectado en red con la CPU.
- Se cumplen todos los requisitos para una configuración IRT, es decir:
  - Los puertos de la CPU y del IM están interconectados entre sí;
  - Está configurada la función de sincronización de la interfaz PROFINET "Maestro Sync" (Propiedades de la CPU, área "Interfaz PROFINET > Opciones avanzadas > Configuración en tiempo real > Sincronización").
  - El IM tiene configurado "IRT" como clase "RT" (área "Interfaz PROFINET > Opciones avanzadas > Configuración en tiempo real > Sincronización").

Procedimiento

Para crear una conexión isócrona de la periferia y el programa de usuario, proceda del siguiente modo:

1. Seleccione la CPU y desplácese en la ventana de inspección hasta el área "Alarmas > Alarmas de sincronismo".
2. Seleccione el dispositivo de periferia descentralizada (el IM 155-5 PN HF) en la vista de redes y cambie a la vista de dispositivos.
3. Inserte al menos un módulo de periferia que pueda funcionar en modo isócrono (módulo con la extensión de nombre "HF", p. ej. DI 16X24VDC HF - 6ES7 521-1BH00).
4. Asigne al módulo de periferia una memoria imagen parcial de proceso (ventana de inspección, área "Direcciones E/S").

5. En la vista de dispositivos o en la vista de redes seleccione los IM y desplácese en la ventana de inspección hasta el área "Interfaz PROFINET > Opciones avanzadas > Modo isócrono".
   - Active la opción "Modo isócrono".
   - Si desea cambiar el tiempo de ciclo de emisión o el ciclo de aplicación haga clic en el link verde situado junto a los campos correspondientes. Entrará en el dominio Sync, donde la configuración se puede modificar de forma centralizada.
   - Para los tiempos Ti/To, seleccione si deben calcularse automáticamente los tiempos mínimos o si deben configurarse manualmente. Con configuración manual es posible modificar los valores Ti/To.
   - En la vista predeterminada de los módulos: Active la opción "Modo isócrono" para los módulos que deban funcionar en dicho modo.
6. Agregue otros dispositivos IO que necesite, conéctelos en red e interconéctelos, y adecue la configuración y los ajustes del modo isócrono.
7. Para acceder a informaciones sobre los anchos de banda calculados o para adaptar el tiempo de ciclo de emisión, seleccione la subred y desplácese hasta el área correspondiente de administración del dominio Sync en la ventana de inspección.

Configuración del modo isócrono en la subred

Introducción

Las propiedades de la subred ofrecen, junto con las del dominio Sync y las características MRP, un resumen de la configuración del modo isócrono.

Requisitos

Hay configurado un sistema PROFINET IO con controlador IO y dispositivos IO.

Procedimiento

Para mostrar las características de modo isócrono de la subred, proceda del siguiente modo:

1. Seleccione la subred.
2. En las propiedades, seleccione el área "Modo isócrono".

Configuración de la vista general del modo isócrono

CPU	Selección de la CPU (controlador IO) de la subred
Aplicación isócrona	Selección de un OB de la CPU seleccionada, asignado a un sistema IO
Detalles de la aplicación isócrona	

Ciclo de aplicación	Visualización del ciclo de aplicación (múltiplo del ciclo de datos). Para S7-300/400, ajuste el tiempo en las propiedades de la CPU; para S7-1500 ajuste el tiempo en las propiedades del OB de sincronismo.
Tiempo de retardo	Visualización del tiempo transcurrido entre el ciclo de reloj del sistema y el arranque del OB de alarma de sincronismo (p. ej. OB 61).
Tiempo de ciclo de emisión	Visualización del periodo entre dos intervalos consecutivos para comunicación IRT o RT
N.º del sistema IO	Visualización del N.º del sistema PROFINET IO asignado al OB
Memoria imagen parcial de proceso	Visualización de los números de la memoria imagen parcial de proceso asignados al OB del sistema PROFINET IO.
Valores Ti/To en el OB	
Valores Ti/To automáticos	Si está activada esta opción, STEP 7 calcula un valor Ti y un valor To adecuados para todos los módulos que funcionan sincrónicamente y que están asignados al OB. En la tabla de dispositivos IO y módulos se puede hacer referencia a estos valores Ti/To comunes (valores Ti/To = del OB).
Tiempo Ti	Tiempo de lectura de los datos de entrada isócronos.
Tiempo To	Tiempo de emisión de los datos de salida isócronos
Dispositivos IO o módulos	
Tabla de valores Ti/To:	El funcionamiento isócrono puede activarse y desactivarse individualmente para cada dispositivo IO y para cada módulo de los mismos. Valores Ti/To: Son posibles tres configuraciones: • Automático: Ti/To se calculan automáticamente. • Manual: Ti/To pueden editarse. • Desde el OB: El OB acepta los valores Ti/To. Ti, To: Solo pueden editarse en configuración "Manual". TiMín./TiMáx.: tiempos Ti mínimo y máximo. ToMín./ToMáx.: tiempos To mínimo y máximo. Intervalo: tiempo predeterminado para valores Ti/To.

Reglas, requisitos y condiciones marco

Lo que debe tenerse en cuenta para la configuración del modo isócrono

Para configurar el modo isócrono tenga en cuenta las indicaciones siguientes:

Componentes e interfaces PROFINET

- El modo isócrono sólo es posible con las interfaces integradas en la CPU. Con S7-1500 de dos interfaces PROFINET, solo puede configurarse para PROFINET IO y modo isócrono la interfaz X1. No es posible el modo isócrono con CPs.
- Sólo es posible un funcionamiento isócrono total de "terminal" a "terminal" si todos los componentes de una cadena soportan la propiedad de sistema "Modo isócrono". Al seleccionar el módulo en el catálogo asegúrese de que el campo de información del módulo contenga el término "modo isócrono".
- Si un dispositivo está funcionando en modo isócrono (es decir, p. ej. tiene asignada la función de esclavo Sync.), en el dispositivo IO debe haber al menos un módulo o un submódulo que funcione en modo isócrono.

Modo isócrono de varias líneas

- Actualmente no es posible un funcionamiento isócrono que abarque varias líneas.

Memorias imagen parciales de proceso

- La periferia isócrona sólo puede procesarse en memorias imagen parcial de proceso. Si no se utilizan memorias imagen parcial de proceso, no es posible una transmisión de datos coherente e isócrona. Se supervisa el cumplimiento de los recursos disponibles, ya que el número de dispositivos IO y el número de bytes en el sistema IO están limitados en cada memoria imagen parcial de proceso.
- Las direcciones de la periferia isócrona deben estar en una memoria imagen parcial de proceso.
- La memoria imagen parcial de proceso para periferia isócrona también puede incluir periferia no isócrona.
- S7-300/400: Es posible que haya más de una memoria imagen parcial de proceso para un OB de alarma de sincronismo o para un sistema IO.
- S7-1500: Puede haber una y solo una memoria imagen parcial de proceso para un OB de alarma de sincronismo o para un sistema IO. Solo pueden estar en la misma memoria imagen parcial de proceso los módulos o submódulos de un mismo sistema IO o de un mismo OB de alarma de sincronismo

Temporizadores

- Cuando está activado el modo isócrono, el tiempo de actualización calculado por STEP 7 siempre es igual al tiempo de ciclo de emisión. No es posible modificar manualmente el tiempo de actualización.
- El tiempo del ciclo de aplicación está limitado a 32 ms.
- El tiempo de retardo para el OB de alarma de sincronismo debe estar entre 0 y (ciclo de datos - 1 µs).

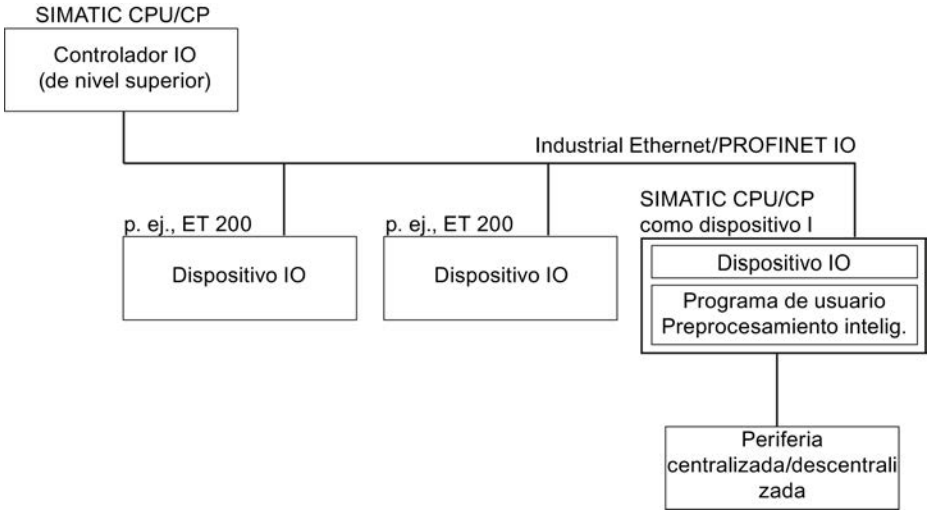
Configuración de dispositivos I

Funcionalidad I-device

Funcionalidad I-device

La funcionalidad "I-device" (dispositivo IO inteligente) de una CPU permite intercambiar datos de forma determinista con un controlador IO y, por consiguiente, emplear la CPU como una unidad inteligente de preprocesamiento de procesos parciales, por ejemplo. En este caso, el I-device está integrado como dispositivo IO a un controlador IO de nivel superior.

El preprocesamiento queda asegurado por el programa de usuario en la CPU. Los valores de proceso registrados en la periferia centralizada o descentralizada (PROFINET IO o PROFIBUS DP) son preprocesados por el programa de usuario y puestos a disposición de la CPU o del CP de un equipo de nivel superior a través de la interfaz de un dispositivo PROFINET IO.



Convención relativa al nombre "I-device"

En adelante, una CPU o un CP con la funcionalidad I-device se denominará simplemente "I-device".

Propiedades y ventajas del I-device

Campos de aplicación

Campos de aplicación del I-Device:

- Procesamiento distribuido
  Una tarea de automatización compleja puede repartirse entre unidades/procesos parciales menores. De este modo, resultan procesos abarcables que generan tareas parciales simplificadas.
- Desacoplamiento de procesos parciales
  Los procesos complicados, ampliamente distribuidos y voluminosos pueden dividirse en varios procesos parciales con interfaces abarcables gracias al uso de I-Devices. Estos procesos parciales, a su vez, se pueden almacenar en proyectos individuales que posteriormente se agrupan en un proyecto global.
- Protección de Know-how
  Partes de la instalación ya sólo pueden suministrarse con un archivo GSD en lugar de con un proyecto para la descripción de la interfaz del I-Device. Por lo tanto, el know-how del programa de usuario ya no tiene que ser revelado.

Propiedades

Propiedades del I-Device:

- Desacoplamiento de proyectos
  Las personas que crean y utilizan un I-Device pueden tener proyectos de automatización completamente separados. La interfaz entre los proyectos la constituye el archivo GSD. De este modo es posible el acoplamiento con controladores IO estándar a través de una interfaz estandarizada.
- Comunicación en tiempo real
  El dispositivo I se pone a disposición de un sistema PROFINET IO determinista mediante una interfaz PROFINET IO, con lo que es compatible con la comunicación en tiempo real RT (Real Time).

Ventajas

El I-Device ofrece las ventajas siguientes:

- Acoplamiento sencillo de controladores IO sin herramientas de software adicionales.
- Comunicación en tiempo real entre CPUs SIMATIC y con controladores IO estándar.
- Gracias a la distribución de la capacidad de procesamiento entre varios I-Devices, es posible reducir el rendimiento necesario de las diferentes CPU y, por supuesto, del controlador IO.
- Menor carga de comunicación gracias al procesamiento local de los datos de proceso.
- Claridad gracias al procesamiento de tareas parciales en CPUs o proyectos separados.

Intercambio de datos entre el sistema IO de nivel superior y el subordinado

Introducción

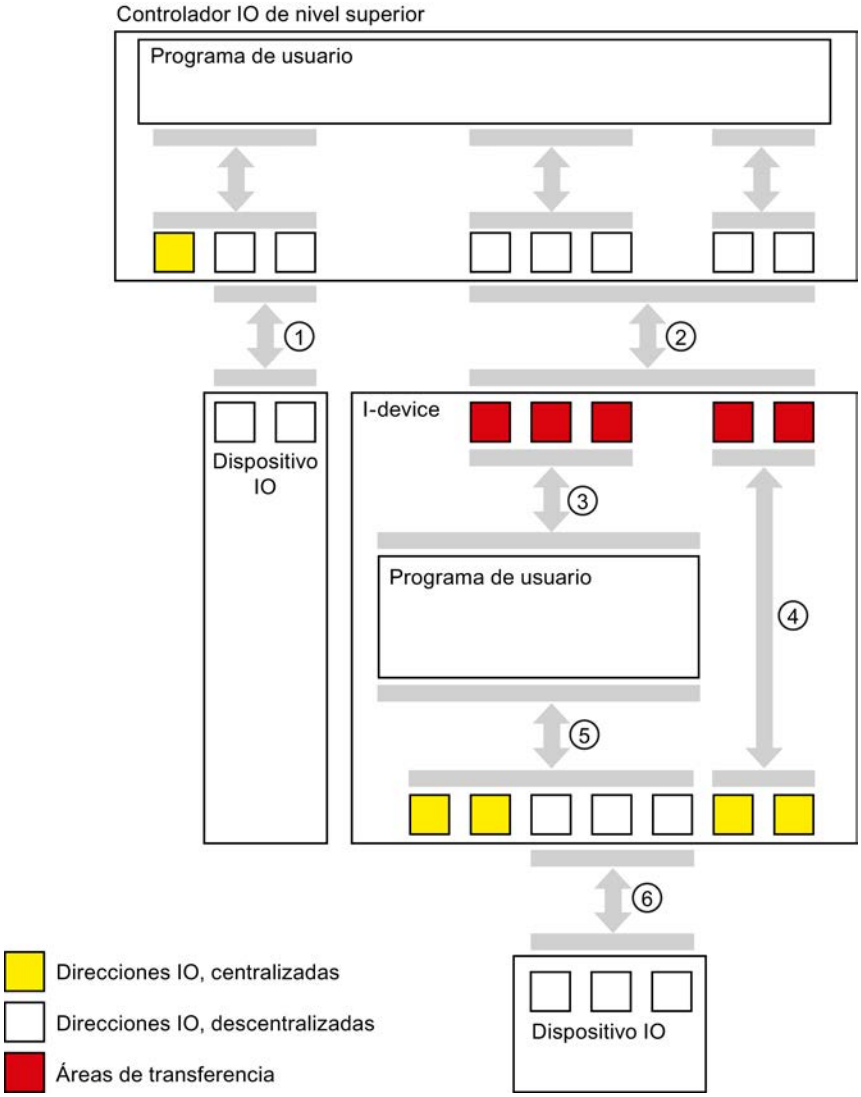
En el capítulo siguiente se presenta el intercambio de datos entre el sistema IO de nivel superior y el subordinado.

Áreas de transferencia

En las áreas de transferencia se preparan los datos para la comunicación entre el controlador IO y el dispositivo I. Un área de transferencia contiene una unidad informativa que se intercambia de forma coherente entre el controlador IO y el dispositivo I.

Las áreas de transferencia de aplicación son una interfaz con el programa de usuario de la CPU del I-device. Las entradas se procesan en el programa de usuario y las salidas son el resultado de un procesamiento en el programa de usuario.

En la figura siguiente se representa el intercambio de datos entre el sistema IO de nivel superior y el subordinado. Las diferentes relaciones de comunicación se explican a continuación con cifras.



① Intercambio de datos entre un controlador IO de nivel superior y un dispositivo IO normal

Los controladores IO y los dispositivos IO intercambian datos vía PROFINET.

② Intercambio de datos entre un controlador IO de nivel superior y un I-device

El controlador IO y el I-device intercambian datos vía PROFINET.

El intercambio de datos entre un controlador IO de nivel superior y un I-device se basa en la relación tradicional entre controlador IO y dispositivo IO.

Para el controlador IO de nivel superior, las áreas de transferencia del I-device representan submódulos de un equipo preconfigurado.

Los datos de salida del controlador IO son los datos de entrada del I-device. De forma análoga, los datos de entrada del controlador IO son los datos de salida del I-device.

③ **Relación de transferencia entre el programa de usuario y el área de transferencia**

El programa de usuario y el área de transferencia de aplicación intercambian datos de entrada y salida por esta vía.

④ **Relación de transferencia entre el área de transferencia y la periferia del I-device**

En la actualidad no es compatible con la configuración.

⑤ **Intercambio de datos entre el programa de usuario y la periferia del I-device**

El programa de usuario y la periferia centralizada y descentralizada intercambian datos de entrada y salida por esta vía.

⑥ **Intercambio de datos entre el I-device y un dispositivo IO subordinado**

El I-device y sus dispositivos IO intercambian datos por esta vía. Los datos se transfieren vía PROFINET.

Configurar un I-Device

Introducción

En principio hay dos posibilidades para realizar la configuración:

- Configurar un I-Device dentro de un proyecto
- Configurar un I-Device que se utiliza en otro proyecto o que se utiliza en otro sistema de ingeniería.

Si configura un I-Device para otro proyecto u otro sistema de ingeniería, STEP 7 se lo permite si exporta un I-Device configurado a un archivo GSD. Puede importar el archivo GSD en el otro proyecto o en el otro sistema de ingeniería, al igual que otros archivos GSD. En este archivo GSD se guardan, entre otras cosas, las áreas de transferencia para el intercambio de datos.

Configurar un I-Device dentro de un proyecto

1. Arrastre un controlador PROFINET IO desde el catálogo de hardware a la vista de redes mediante la función de arrastrar y soltar.
2. Arrastre un controlador PROFINET IO, que se pueda parametrizar como dispositivo IO, desde el catálogo de hardware a la vista de redes mediante la función de arrastrar y soltar. Este dispositivo se parametriza como I-Device (p. ej., CPU 319-3 PN/DP, FW 3.2).
3. Seleccione la interfaz PROFINET del I-Device.
4. En la ventana de inspección, seleccione en la navegación local la entrada "Modo de operación" y active la casilla de verificación "Dispositivo IO".
5. En la lista desplegable "Controlador IO asignado" puede seleccionar ahora el controlador IO.
   Cuando haya seleccionado el controlador IO, podrá apreciar la interconexión y el sistema IO entre ambos dispositivos en la vista de redes.

6. En la casilla de verificación "Parametrizar interfaz PN desde el controlador IO de nivel superior" puede definir si la interfaz y sus puertos son parametrizados por el propio I-Device o si lo hace el controlador IO de nivel superior.
   Si utiliza el I-Device con un sistema IO subordinado, la interfaz PROFINET (p. ej., parámetros de puerto) del I-Device no puede parametrizarse desde el controlador IO de nivel superior.
7. Configure las áreas de transferencia. Se trata de áreas de periferia a través de las cuales el I-Device intercambia datos con el controlador IO de nivel superior. Las áreas de transferencia se encuentran en la navegación local en el apartado "Comunicación de I-Device".
   - Haga clic en el primer campo de la columna "Áreas de transferencia". STEP 7 asigna un nombre predeterminado que se puede modificar.
   - Seleccione el tipo de relación de comunicación: en la actualidad sólo se puede seleccionar CD o F-CD como "Relación de comunicación controlador-dispositivo".
   - Las direcciones se establecen automáticamente; si es necesario, corrija las direcciones y defina la longitud del área de transferencia que debe transferirse de forma coherente.
8. Para cada área de transferencia se genera una entrada individual en la navegación local. Si selecciona una de estas entradas, puede adaptar o corregir y comentar los detalles del área de transferencia.
   Si la CPU es compatible con esta funcionalidad, también puede activar la opción "Mapeo de direcciones bidireccional". En este caso se establecen de forma automática áreas de direcciones "en dirección contraria", de forma que las CPUs ocupan simétricamente las áreas de entrada y salida.

Configurar un I-Device con ayuda de un archivo GSD

Si utiliza el I-Device en otro proyecto o si el I-Device se usa en otro sistema de ingeniería, configure el controlador IO de nivel superior y el I-Device de la forma antes descrita.

Asimismo, después de la configuración de las áreas de transferencia, haga clic en el botón "Exportar", para crear un archivo GSD a partir del I-Device. Este archivo GSD es el representante del I-Device configurado en el otro proyecto.

Encontrará el botón "Exportar" en el apartado "Comunicación de I-Device" en la ventana de inspección.

La configuración hardware se compila y se abre el cuadro de diálogo Exportar.

Asigne un nombre para el representante del I-Device y una descripción en los campos previstos. Haga clic en el botón "Exportar" para finalizar el proceso.

A continuación, importe el archivo GSD, p. ej. a otro proyecto.

Consulte también

Configurar el I-Device como Shared Device (Página 1584)

Condiciones límite al emplear I-devices

Condiciones marco

Cuando se emplean I-Devices deben observarse algunas condiciones marco en casos especiales.

Ancho de banda

El volumen de direcciones de las áreas de transferencia configuradas se traduce en el ancho de banda útil del I-Device:

Ancho de banda de las áreas de transferencia + ancho de banda del sistema IO subordinado = ancho de banda total utilizado en el I-Device.

Si el área de direcciones de las áreas de transferencia es demasiado grande, no queda suficiente ancho de banda para el sistema IO subordinado para poder conseguir tiempos de actualización breves.

Sugerencia: Mantenga el área de direccionamiento de las áreas de transferencia lo más pequeña posible.

Reglas para la comunicación RT e IRT

Los sistemas IO con I-Devices también son adecuados para crear aplicaciones en tiempo real con comunicación RT. A continuación se da por supuesto que se trata de un sistema jerárquico, compuesto por un controlador IO de nivel superior al que se ha asignado un I-Device (sistema IO de nivel superior). A su vez, se han asignado dispositivos IO al I-Device (sistema IO subordinado). Rige la regla siguiente:

Tanto el sistema IO de nivel superior como el subordinado son compatibles con la comunicación RT. Es posible utilizar a la vez la comunicación RT para ambos sistemas IO.

La comunicación IRT puede combinarse con la comunicación RT. Sin embargo, la comunicación IRT no puede tener lugar en ambos sistemas IO al mismo tiempo.

Ejemplo:

- Si en el I-Device está **activada** la opción "Parametrización de la interfaz PN y sus puertos en el controlador IO de nivel superior", el I-Device puede funcionar en el controlador IO de nivel superior con la clase RT "IRT". Para el sistema IO de nivel inferior del I-Device no es posible en este caso una comunicación IRT.
- Si en el I-Device está **desactivada** la opción "Parametrización de la interfaz PN y sus puertos en el controlador IO de nivel superior", el I-Device debe funcionar en el controlador IO de nivel superior con la clase RT "RT". Para el sistema IO de nivel inferior es posible la comunicación IRT.

Si configura el I-Device en un sistema IO con la clase RT "RT", no se pueden ajustar tiempos de ciclo de emisión impares.

Reglas para el acceso a los datos

- El controlador IO de nivel superior puede acceder a las áreas de transferencia cuando el I-Device está en RUN.
- En la CPU I-Device sólo se puede acceder a las áreas de transferencia de aplicación.
  - El acceso a áreas de transferencia de aplicación de entrada es posible, si el controlador IO de nivel superior está en RUN.
  - El acceso a áreas de transferencia de aplicación de entrada es posible, independientemente del estado operativo del controlador IO de nivel superior.

Parámetros de dirección IP y nombre de dispositivo

Igual que cualquier otro dispositivo IO, un I-Device también necesita parámetros de dirección IP / nombres de dispositivo para poder comunicarse a través de PROFINET. Los parámetros de dirección IP constan de tres partes: la dirección IP propiamente dicha, la máscara de subred y la dirección del router (gateway).

Los parámetros de dirección IP / nombres de dispositivo de un I-device pueden asignarse de dos modos distintos: desde el proyecto o directamente en el dispositivo.

Ajustar la dirección IP / el nombre de dispositivo en el proyecto:

- Los parámetros de dirección IP / el nombre de dispositivo se asignan de forma fija durante la configuración (en el proyecto del I-Device) en STEP 7. Este método es el estándar.

Ajustar la dirección IP / el nombre de dispositivo en el dispositivo:

- Parámetros de dirección IP / nombre de dispositivo vía DCP: Los parámetros de dirección IP / el nombre de dispositivo se asignan mediante DCP (Discovery and Configuration Protocol). Existen dos formas de hacerlo:
  - Mediante una herramienta de configuración como PST o STEP 7 (vía dispositivos accesibles).
  - A través del controlador IO de nivel superior.
- Parámetros de dirección IP / nombre de dispositivo vía programa de usuario: Los parámetros de dirección IP se asignan en el programa de usuario de la CPU del I-Device (a través de la instrucción "IP_CONFIG").

Nota

Visualización de la topología

Si el nombre de dispositivo y los parámetros de dirección IP de un I-device se ajustan directament en el dispositivo, el I-device puede aparecer varias veces en el editor topológico.

Diagnóstico y respuesta a alarmas

Diagnóstico y respuesta a alarmas

Las CPU S7 disponen de múltiples funciones de diagnóstico y alarma que, por ejemplo, notifican errores o fallos en los sistemas IO subordinados. Dichos avisos de diagnóstico reducen los tiempos de parada y facilitan la localización y solución de errores.

Diagnóstico del I-Device en el controlador IO de nivel superior y en el I-Device

El controlador IO de nivel superior y la CPU del I-Device disponen de los siguientes mecanismos para diagnosticar el estado de cada interlocutor:

- OB de extracción/inserción (OB 83) para alarma "Return-of-Submodule" (recuperación de submódulo)
- OB de error de ejecución del programa (OB 85) para, por ejemplo, errores de transferencia de la memoria imagen de proceso (sólo S7-300/400)
- OB de fallo del rack (OB 86) para fallo de aparatos de ampliación, sistemas IO o dispositivos de periferia descentralizada (fallo de estación, recuperación de la estación)
- OB de error de acceso a la periferia (OB 122), opcional en S7-1500
- Tratamiento de errores local (sólo S7-1500)

Nota

Los avisos de diagnóstico de la periferia pueden procesarse en el programa de usuario de la CPU I-Device y, desde allí, transferirse al controlador IO de nivel superior por medio de las áreas de transferencia.

Cambio del estado operativo y fallo/recuperación de la estación

En la tabla siguiente se muestran las consecuencias que puede tener un cambio del estado operativo o un fallo de una CPU (I-Device, controlador IO) sobre la(s) otra(s).

Tenga en cuenta que una CPU S7-1500, en caso de errores en el acceso directo a las áreas de transferencia (p. ej. porque la otra CPU pasa a STOP) llama al OB 122, si está disponible. Pero, a diferencia de una CPU S7-300/400, esta CPU no pasa a STOP si el OB 122 no está disponible. Además, con una CPU S7-1500 tiene la posibilidad de realizar el tratamiento de errores local dentro del bloque en el que ha aparecido el error. Esta circunstancia se refleja de forma abreviada en la tabla mediante la fórmula "Llamada del OB 122".

Eventos

Estado inicial	Evento	Comportamiento del I-Device	Comportamiento de controladores de nivel superior
Ambas CPU están en RUN (controlador IO de nivel superior e I-Device).	La CPU I-Device pasa a STOP	/	**S7-300/400/1500:** código de error en RET_VAL al actualizar la memoria imagen de proceso con la instrucción UPDAT_PI o UPDAT_PO. Al acceder directamente a las áreas de transferencia hacia el I-Device se llama al OB 122 (opcional en S7-1500). **S7-300/400**: si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso del sistema al I-Device se produce la llamada al OB 85.
La CPU I-Device está en STOP, el controlador IO de nivel superior está en RUN.	La CPU I-Device arranca	**S7-300/400/1500:** llamada del OB 100 (arranque). Llamada del OB 83 para áreas de transferencia de entrada al controlador IO de nivel superior. Hasta la llamada del OB 83, al acceder directamente a las áreas de transferencia de entrada al controlador IO de nivel superior se producen errores de acceso: Llamada del OB 122. **S7-300/400:** si está configurada la llamada del OB85 en caso de error de acceso a la periferia, hasta la llamada del OB 83 se sigue llamando al OB 85 en la transferencia de la memoria imagen de proceso del sistema al controlador IO de nivel superior.	**S7-300/400/1500:** llamada del OB 83: para todas las áreas de transferencia al I-Device. Hasta la llamada del OB 83, al acceder directamente a las áreas de transferencia al I-Device se sigue llamando al OB 122. **S7-300/400:** Si está configurada la llamada del OB85 en caso de error de acceso a la periferia, hasta la llamada del OB 83 se sigue llamando al OB 85 en la transferencia de la memoria imagen de proceso del sistema al I-Device.

Estado inicial	Evento	Comportamiento del I-Device	Comportamiento de controladores de nivel superior
Ambas CPU están en RUN (controlador IO de nivel superior e I-Device).	El controlador IO de nivel superior pasa a STOP	**S7-300/400/1500:** código de error en RET_VAL al actualizar la memoria imagen de proceso con la instrucción UPDAT_PI o UPDAT_PO. Al acceder directamente a las áreas de transferencia de entrada al controlador IO de nivel superior se produce la llamada al OB 122. Nota: Es posible seguir accediendo a las áreas de transferencia de salida. **S7-300/400**: si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso de las áreas de transferencia de entrada del sistema al controlador IO de nivel superior se produce la llamada al OB 85.	/
El controlador IO de nivel superior está en STOP, la CPU I-Device está en RUN	El controlador IO de nivel superior arranca	**S7-300/400/1500**: Llamada del OB 83 para áreas de transferencia de entrada al controlador IO de nivel superior. Hasta la llamada del OB 83, al acceder directamente a las áreas de transferencia de entrada al controlador IO de nivel superior se produce la llamada al OB 122. **S7-300/400**: si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso de las áreas de transferencia de entrada del sistema al controlador IO de nivel superior se produce la llamada al OB 85.	Llamada del OB 100 (arranque).

Estado inicial	Evento	Comportamiento del I-Device	Comportamiento de controladores de nivel superior
Ambas CPU están en RUN (controlador IO de nivel superior e I-Device).	Fallo de estación I-Device, p. ej. por interrupción de bus	**S7-300/400/1500**: Si el I-Device sigue funcionando sin conexión al bus: Llamada del OB 86 (fallo de estación). Código de error en RET_VAL al actualizar la memoria imagen de proceso con la instrucción UPDAT_PI o UPDAT_PO. Al acceder directamente a las áreas de transferencia al controlador IO de nivel superior se produce la llamada al OB 122. **S7-300/400**: si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso de las áreas de transferencia del sistema al controlador IO de nivel superior se produce la llamada al OB 85.	**S7-300/400/1500**: Llamada del OB 86 (fallo de estación). Código de error en RET_VAL al actualizar la memoria imagen de proceso con la instrucción UPDAT_PI o UPDAT_PO. Al acceder directamente a las áreas de transferencia hacia el I-Device se llama al OB 122. **S7-300/400**: si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso de las áreas de transferencia del sistema al I-Device se produce la llamada al OB 85.
Ambas CPU están en RUN (controlador IO de nivel superior e I-Device). La conexión de comunicación entre el controlador IO y el I-Device está interrumpida (interrupción de bus).	La conexión de bus entre el controlador IO y el I-Device se restablece y el I-Device es reincorporado al tráfico de datos de usuario por el controlador IO.	**S7-300/400/1500**: observe la advertencia bajo esta tabla sobre las particularidades al arrancar el I-Device. Llamada del OB 86 (recuperación de estación). Llamada del OB 83 para las áreas de transferencia de entrada al controlador IO de nivel superior. Hasta la llamada del OB 83, al acceder directamente a las áreas de transferencia de entrada al controlador IO de nivel superior se produce la llamada al OB 122. **S7-300/400:** si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso de las áreas de transferencia de entrada del sistema al controlador IO de nivel superior se produce la llamada al OB 85.	**S7-300/400/1500**: Llamada del OB 86 (recuperación de la estación). Hasta la llamada del OB 86, al acceder directamente a las áreas de transferencia al I-Device se llama al OB 122. **S7-300/400:** si está configurada la llamada del OB85 en caso de error de acceso a la periferia, en la transferencia de la memoria imagen de proceso de las áreas de transferencia del sistema al I-Device se produce la llamada al OB 85.

Nota

Particularidades al arrancar el I-Device

A diferencia del aviso de la recuperación de estación de los dispositivos IO en el controlador IO que está completamente incluido en una llamada del OB 86, la notificación de la recuperación de estación de un controlador IO de nivel superior en el I-Device se divide en 2 partes:

1. Llamada del OB 86: Se ajustan los valores iniciales para las salidas del I-Device. Sin embargo, los valores de entrada aún no son válidos, lo serán con la llamada del OB 86 en el controlador IO de nivel superior.
2. Llamada del OB 83 para cada área de transferencia de entrada. Con esta llamada se indica la validez de un área de transferencia de entrada. El arranque del I-Device sólo se considerará finalizado cuando se haya llamado al OB 83 para las áreas de transferencia de entrada. Este paso puede retrasarse significativamente o incluso suspenderse en las siguientes situaciones:
   - El controlador IO de nivel superior está en STOP: Al OB 83 se llama sólo en la transición de STOP→RUN del controlador IO de nivel superior.
   - La comunicación del IRT se ha interrumpido, p. ej. porque falla el maestro Sync, a causa de errores de topología, etc.: el OB 83 sólo arranca cuando se ha restaurado la comunicación IRT.

Consulte también

Principios básicos del tratamiento de errores (Página 2932)

OB de cambio de módulo (OB 83) (Página 1854)

OB de cambio de módulo (Página 1981)

OB de fallo del rack (OB 86) (Página 1864)

OB de fallo de rack (Página 1982)

OB de error de acceso a la periferia (OB 122) (Página 1883)

OB de error de acceso a la periferia (Página 1986)

OB de error de ejecución del programa (OB 85) (Página 1859)

Diagnóstico online de los dispositivos I

En principio el diagnóstico online de los I-Devices no se diferencia del diagnóstico online de los dispositivos IO "normales".

Visualización del estado de diagnóstico al establecer una conexión online

En función del dispositivo que haya seleccionado al establecer una conexión online, se muestra la información siguiente:

- Ha seleccionado el controlador IO de nivel superior:
  En el árbol del proyecto se muestra el estado de diagnóstico del I-Device en el controlador IO en la carpeta "Periferia descentralizada". En el árbol del proyecto no se muestra el estado de diagnóstico del I-Device.
- Ha seleccionado el I-Device que se ha asignado a este controlador IO.
  En el árbol del proyecto se muestra el estado de diagnóstico del I-Device. No se muestra el estado de diagnóstico del I-Device que se encuentra bajo el controlador IO en la carpeta "Periferia descentralizada".

Vista Online y diagnóstico

En función del dispositivo que haya seleccionado al iniciar la vista Online y diagnóstico, se muestra la información siguiente:

- Ha seleccionado el controlador IO de nivel superior y está conectado online con este dispositivo:
  La vista Online y diagnóstico muestra los datos de diagnóstico del I-Device desde el punto de vista del controlador IO de nivel superior. La visualización se corresponde con la de un controlador IO "normal".
- Ha seleccionado el I-Device que se ha asignado a este controlador IO y está conectado online con este dispositivo:
  La vista Online y diagnóstico muestra los datos de diagnóstico del I-Device desde el punto de vista local, es decir, desde el punto de vista del I-Device. La visualización se corresponde con la de un dispositivo IO "normal".

Visualización en la lista de los dispositivos accesibles

En la lista de los dispositivos accesibles se muestra el I-Device como una CPU "normal".

Dispositivos IO que cambian en funcionamiento

Docking Station - Dispositivos IO que cambian en funcionamiento

Uso de dispositivos IO que cambian durante el funcionamiento ("puerto interlocutor cambiante") en una Docking Station

En la siguiente figura se representa una célula de automatización con un Docking System y varias Docking Units.

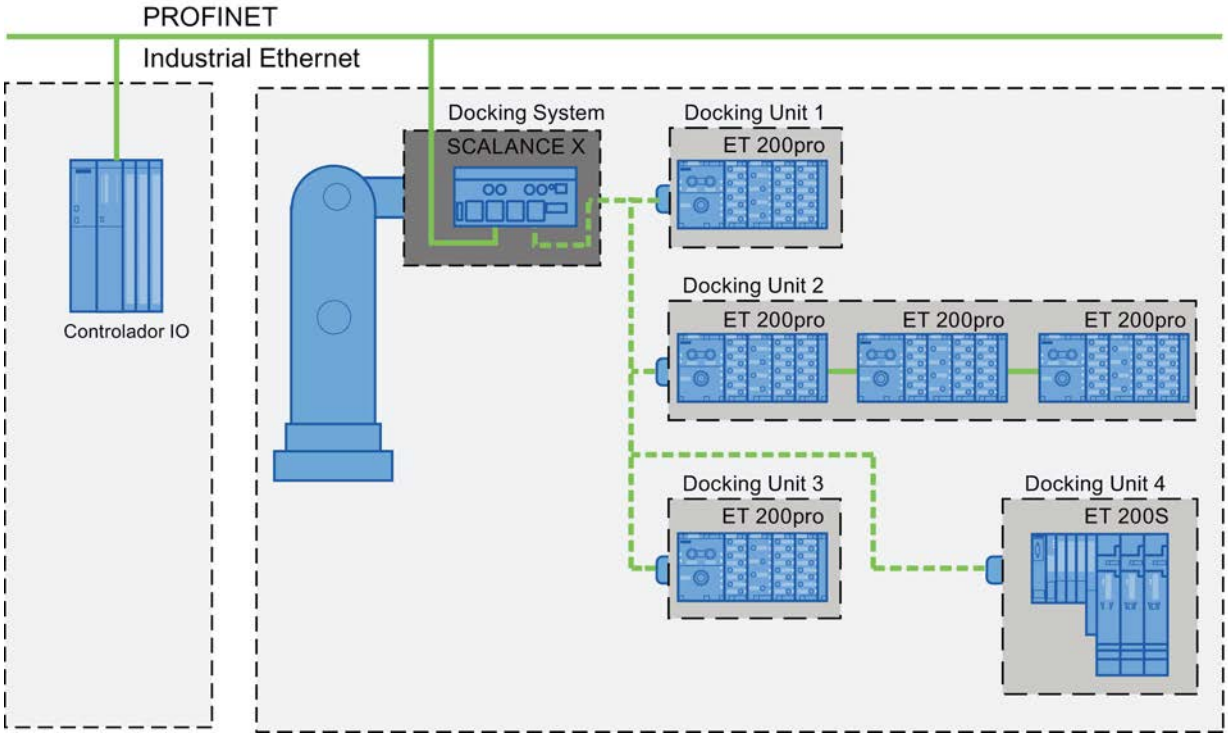


Figura 8-49 Dispositivos IO cambiantes (puertos interlocutores) en una Docking Station

Requisitos aplicativos

Al crear un Docking System con dispositivos IO que cambian durante el funcionamiento deben observarse los siguientes aspectos:

- En la configuración, los dispositivos IO de todas las Docking Units deben estar desactivados por defecto.
- En cada momento solo puede haber una Docking Unit activa, es decir, que solo pueden estar activos los dispositivos IO de una unidad. Todos los dispositivos IO de las otras Docking Units deben estar desactivados o deben desactivarse antes de poder activar los dispositivos IO de una Docking Unit. Esto se logra con la instrucción "D_ACT_DP".

- Para activar una Docking Unit se establece una conexión física con ella y con sus dispositivos IO y a continuación se conectan (Power ON) todos los dispositivos IO. Al mismo tiempo se deben activar en el programa de usuario todos los dispositivos IO de esa Docking Unit mediante la instrucción "D_ACT_DP".
- Una vez recibida la respuesta "Dispositivo IO activado", acceda al dispositivo IO con el comando de acceso directo a periferia.
- Llame la instrucción "D_ACT_DP" para activar y desactivar el dispositivo IO, a ser posible al comienzo del ciclo del OB 1.

Campo de aplicación de dispositivos IO que cambian durante el funcionamiento

La funcionalidad PROFINET de dispositivos IO que cambian durante el funcionamiento ("puerto interlocutor cambiante") puede utilizarse p. ej. para el cambio de herramientas en robots. Algunas herramientas típicas son p. ej.:

- Pinzas de soldar y
- Herramientas de fijación de piezas de producción

Nota

Número de dispositivos IO que cambian durante el funcionamiento ("puerto interlocutor cambiante") - Número de Docking Units

Para obtener tiempos de cambio de herramienta lo más cortos posible es necesario observar los siguientes puntos, que varían en función de la CPU o del CP utilizados:

- Solo pueden realizar un arranque optimizado los dispositivos IO configurados con la funcionalidad PROFINET "Arranque priorizado". El número de dispositivos IO configurados con esta funcionalidad PROFINET está limitado.
- Solo se puede activar al mismo tiempo un número determinado de dispositivos IO (en función de los recursos disponibles de la instrucción "D_ACT_DP"), por lo que una Docking Unit no debería contener más del número correspondiente de dispositivos IO. Si en una Docking Unit funcionan más dispositivos IO, deberán activarse sucesivamente uno tras otro, lo cual prolongará el tiempo necesario.

Ejemplo: Una CPU S7 319-3 PN/DP puede manejar como máximo 32 dispositivos IO con arranque priorizado y puede activar al mismo tiempo 8 dispositivos IO por cada instrucción "D_ACT_DP".

Por ello, para una aplicación optimizada en el tiempo una Docking Unit no debería contener más de 8 dispositivos IO y en todas las Docking Units cambiantes no deberían utilizarse más de 32 dispositivos IO.

Limitaciones en la interconexión

En los siguientes casos no es posible realizar una interconexión con un puerto interlocutor:

- El puerto interlocutor tiene un tipo de cable no apto. En tal caso hay que insertar el convertidor de medios desde el catálogo.
- El puerto interlocutor está bloqueado (desactivado).
- Los dos puertos a interconectar pertenecen a la misma interfaz (solo es posible interconectar puertos de diferentes interfaces en una estación).

- Está intentando establecer una conexión en anillo con un módulo que no es apto para la redundancia.
- Si los dos puertos a interconectar pertenecen a diferentes subredes Ethernet.
- El puerto de una interfaz PROFINET de un controlador IO no se puede configurar directamente con la función de cambio de dispositivo IO durante el funcionamiento ("puerto interlocutor cambiante").

Consulte también

D_ACT_DP: Activar / desactivar esclavos DP (Página 6764)

Configurar Docking Units en un Docking System

Configurar Docking Units en un Docking System

Para configurar varias Docking Units en un Docking System, proceda del siguiente modo:

1. Configure la configuración hardware de la forma habitual; considere los dispositivos IO que se corresponden con los partner cambiantes como dispositivos IO normales.
2. Abra en la ventana de inspección las propiedades del puerto en el que quiere usar los dispositivos IO cambiantes (puertos interlocutores cambiantes).
3. En "General > Interfaz PROFINET > Opciones avanzadas > Puerto RJ45 100 Mbits [Xn Pm] > Interconexión de puertos", active la casilla de verificación "Interlocutores alternativos".
4. Inserte el partner alternativo en la tabla de la forma descrita en el capítulo "Interconectar puertos".

Consulte también

Interconectar puertos con puertos partner cambiantes (Página 1562)

Interconectar puertos con puertos partner cambiantes

Requisitos

- Todos los dispositivos que intervienen admiten la configuración de topología
- Si el controlador IO, los dispositivos IO cambiantes (Docking Unit) y el switch (Docking System) en el que funcionan los dispositivos IO admiten esta funcionalidad.
- Si la Docking Unit está conectada con un switch compatible con la funcionalidad PROFINET "Arranque priorizado" (p. ej. de la familia SCALANCE X200IRT).

Ejemplo

- Controlador IO: CPU 416-3 PN/DP
- Docking System: Scalance X-200IRT
- Docking Units (dispositivos IO cambiantes): p. ej., 2 x ET 200pro (IM 154-4)

Procedimiento

En la descripción del procedimiento se presupone la configuración del ejemplo citado anteriormente. Para interconectar los puertos, proceda del siguiente modo:

1. Conecte entre sí todas las interfaces PROFINET de los dispositivos en la vista de redes.
2. Seleccione en la vista de redes o dispositivos el Docking System (el Scalance X200 IRT) cuyo puerto desea interconectar.
3. Abra las propiedades del dispositivo en la ventana de inspección.
4. En la lista desplegable "Puerto interlocutor" que se encuentra en "Propiedades > Interfaz PROFINET > Opciones avanzadas > Puerto RJ45 100 Mbit [Xn Pm] > Interconexión de puertos", seleccione el puerto interlocutor para la CPU (controlador IO).
   Los puertos de la CPU y del Docking System (Scalance) están interconectados entre sí.
5. El Docking System sigue seleccionado: Seleccione ahora para cualquier otro puerto no interconectado los puertos interlocutores para los dispositivos IO:
   - Active la casilla de verificación "Interlocutores alternativos".
     Se muestra una tabla de los puertos interlocutores.
   - Dentro de la tabla, haga clic en el triángulo negro de la primera fila para abrir una lista desplegable.
     En ella se muestran todos los dispositivos con sus puertos no interconectados para su selección.
   - Seleccione el puerto para la primera Docking Unit.
   - Haga clic en la fila siguiente de la tabla y seleccione otro puerto interlocutor alternativo adicional.
   - Repita los pasos arriba mencionados hasta que todos los puertos interlocutores necesarios estén registrados en la tabla.

Nota

Uso de un CP 1243-1 (WAN-CP)

Si se usa una CPU S7-1200 con CP 1243-1, se sigue el procedimiento anterior, pero en lugar de la interfaz PROFINET del controlador IO se utiliza la interfaz Ethernet del CP.

Interconexión de varios dispositivos IO con dispositivos IO que pueden cambiarse en funcionamiento

Los dispositivos IO de una Docking Unit interconectados en línea con dispositivos IO que pueden cambiar en funcionamiento ("interlocutores alternativos") se configuran de la forma habitual con el puerto del dispositivo IO cambiante, es decir, el dispositivo IO interconectado en línea no tiene entonces ningún puerto interlocutor cambiante.

Borrar el puerto interlocutor

Para eliminar la interconexión del puerto interlocutor que cambia en funcionamiento, proceda del siguiente modo:

1. En la tabla de interlocutores alternativos, haga clic en la fila en la que está registrado el puerto interlocutor alternativo que se desea eliminar y expanda la lista de selección de puertos alternativos (triángulo negro).
2. En la lista, seleccione la entrada "Cualquier interlocutor".
   El puerto interlocutor alternativo se elimina de la tabla.

Consulte también

Activar la sustitución de dispositivo sin medio de almacenamiento extraíble (Página 1513)

¿Qué es el arranque priorizado? (Página 1564)

Optimizar los tiempos de arranque

¿Qué es el arranque priorizado?

Definición

El término "arranque priorizado" hace referencia a la funcionalidad PROFINET de acelerar el arranque de dispositivos IO en un sistema PROFINET IO con comunicación RT e IRT. Reduce el tiempo que los dispositivos IO configurados necesitan para regresar al intercambio de datos de usuario cíclicos en los siguientes casos:

- Después de un corte de alimentación
- Después de recuperarse la estación
- Después de la activación de dispositivos IO

Ventajas

La funcionalidad PROFINET "Arranque priorizado" permite disponer de aplicaciones PROFINET IO con cambio permanente de piezas de máquinas o herramientas y de sus dispositivos IO. Los tiempos de espera de varios segundos entre los procesos temporales del rearranque se minimizan gracias a su optimización. Con ello se acelera el proceso de fabricación con dispositivos IO cambiantes, p. ej. en aplicaciones con cambiadores de herramientas, y permite una mayor rentabilidad de la producción.

La funcionalidad PROFINET "Arranque priorizado" también aporta una importante ganancia del rendimiento en aplicaciones en las que se requiere un tiempo de arranque breve de los dispositivos IO después de "Power ON", después de un fallo de la estación y su recuperación, o al activar los dispositivos IO.

Propiedades

Las siguientes propiedades están disponibles con la funcionalidad PROFINET "Arranque priorizado".

- Disponibilidad para la comunicación de los dispositivos IO (periferia descentralizada) hasta un mínimo de 500 ms.
- Puede utilizar el arranque priorizado para dispositivos IO (periferia descentralizada) tanto con comunicación RT como IRT.

Tiempos de arranque

La duración del tiempo de arranque de un dispositivo IO (periferia descentralizada) con la funcionalidad PROFINET "Arranque priorizado" depende de los siguientes aspectos:

- De los dispositivos IO utilizados (periferia descentralizada)
- De la extensión de la periferia del dispositivo IO (periferia descentralizada)
- De los módulos utilizados del dispositivo IO (periferia descentralizada)
- Del controlador IO utilizado
- Del switch utilizado
- Del ajuste de puertos
- Del cableado
- Clase RT configurada del dispositivo IO

Nota

Arranque priorizado después del primer arranque

El arranque priorizado de los dispositivos IO no está disponible hasta después de la primera parametrización del dispositivo IO en cuestión en el primer arranque del sistema PROFINET IO. También en caso de reponer una pieza o después de "Reset to factory Settings" (restablecimiento de los ajustes de fábrica), el primer arranque es un arranque estándar en los dispositivos IO configurados de la forma correspondiente.

Nota

En las siguientes situaciones, es posible que se produzcan tiempos de arranque de hasta 8 s a pesar del arranque priorizado:

- Un dispositivo IO se desacopla y se acopla nuevamente en 8 s.
- En una estación de acoplamiento se acoplan varios dispositivos IO físicos como dispositivo IO con un nombre de dispositivo específico y una configuración IP determinada (p. ej., una estación de acoplamiento para un sistema de transporte sin conductor).

Para conseguir el tiempo de arranque mínimo posible de 500 ms, deben aplicarse las siguientes medidas:

- Ajuste de los puertos en el dispositivo IO
- Cableado en función de los dispositivos PROFINET interconectados
- Medidas en el programa de usuario

Ajustar el arranque priorizado

Requisitos

La funcionalidad PROFINET "Arranque priorizado" sólo puede activarse en los dispositivos IO (periferia descentralizada) en los siguientes casos:

- El controlador IO utilizado puede priorizar dispositivos IO específicos en el arranque.
- El dispositivo IO utilizado admite la priorización.

Nota

Arranque priorizado

En el caso de un arranque acelerado (arranque priorizado) deben cumplirse en especial los requisitos referentes al ajuste de la interfaz PROFINET y del cableado para conseguir los tiempos de arranque mínimos posibles.

Procedimiento

1. Seleccione en la vista de redes o dispositivos el dispositivo IO cuyo arranque debe acelerarse.
2. Abra en la ventana de inspección las propiedades del dispositivo IO.
3. Vaya a "Interfaz PROFINET > Opciones avanzadas > Opciones de interfaz".
4. Active la casilla de verificación "Activar arranque priorizado".
5. Cargue la configuración en el controlador IO.

Observación:

En dispositivos en que puede configurarse el modo de operación DP, la casilla de verificación descrita anteriormente se encuentra en "Interfaz PROFINET > Modo de operación". Son de este tipo de dispositivos, por ejemplo el CP 1616 para estaciones PC.

Nota

Número de dispositivos IO (periferia descentralizada) con arranque priorizado

Dentro de un sistema PROFINET IO sólo puede utilizarse un número máximo de dispositivos IO con la funcionalidad PROFINET "Arranque priorizado". El número depende del controlador IO utilizado.

Consulte también

Optimizar los ajustes de los puertos en el dispositivo IO y el controlador IO (Página 1567)

Optimizar los ajustes de los puertos en el dispositivo IO y el controlador IO

Optimizar los ajustes de los puertos en el dispositivo IO y el controlador IO

Durante el arranque del dispositivo IO se realiza una comprobación del medio de transferencia y la opción dúplex en caso de cableado con CU.

Estas comprobaciones requieren algún tiempo, que se puede reducir gracias a la configuración precisa de estas opciones. Asegúrese de que la configuración realizada se corresponde con las condiciones reales (uso de los cables correctos).

Adaptar las propiedades del puerto local y del puerto interlocutor

Con el fin de adaptar los ajustes del puerto local y del puerto interlocutor, proceda del siguiente modo:

1. Seleccione primero el puerto local y después el puerto interlocutor y realice los siguientes pasos con ambos puertos.
2. En las propiedades de cada puerto adapte en "Opciones de puerto" el ajuste "Medio de transmisión/dúplex". Utilice el ajuste "TP 100 Mbits/s dúplex".
3. En "Opciones de puerto" desactive la casilla de verificación "Activar autonegotiation" en ambos puertos.

Consulte también

Normas de cableado con autonegotiation desactivada (Página 1511)

Configuración de redundancia de medios

Información importante sobre la redundancia de medios

Posibilidades de la redundancia de medios

Hay disponibles diferentes posibilidades que aumentan la disponibilidad de la red de una red Industrial Ethernet con topologías en línea ópticas o eléctricas:

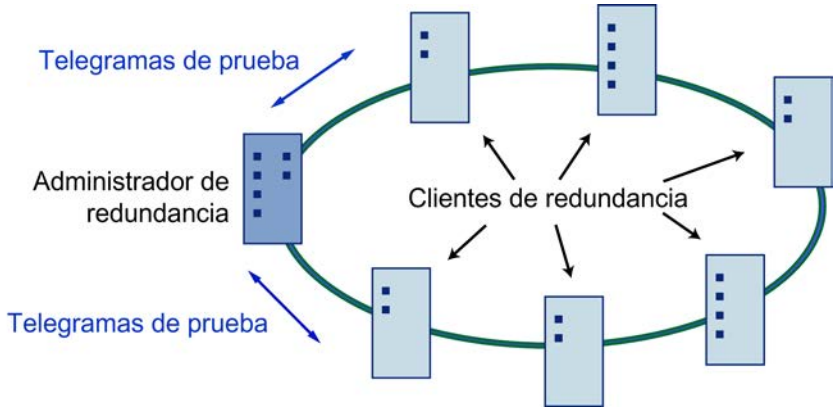
- Interconexión de redes
- Conexión en paralelo de rutas de transmisión
- Concentración de una topología en línea en una topología en anillo

Redundancia de medios en topologías en anillo

Los dispositivos de topologías en anillo pueden ser switches externos y/o los switches integrados de módulos de comunicación.

Para instalar una topología en anillo con redundancia de medios tiene que agrupar los dos extremos libres de una topología en red con forma de línea en un dispositivo. La concentración de la topología en línea en una topología en anillo se realiza mediante dos puertos (puertos en anillo) de un dispositivo en el anillo. Este dispositivo es el administrador de redundancia. Los demás dispositivos del anillo son los clientes de la redundancia.

En la imagen siguiente se muestra la interconexión en forma de anillo de los dispositivos IO:



los dos puertos en anillo de un dispositivo son los puertos que generan la conexión a los dos dispositivos vecinos en una topología en anillo. En la configuración de cada dispositivo se puede seleccionar y definir el puerto en anillo.

Antes de realizar la concentración física del anillo, cargue la configuración en cada dispositivo.

Funciones de la redundancia de medios en una topología en anillo

Si el anillo se interrumpe en algún punto, se vuelven a configurar automáticamente las rutas de datos entre los dispositivos individuales. Después de la reconfiguración, se puede volver a acceder a los dispositivos en la topología recién creada.

En el administrador de redundancia se separan los dos puertos en anillo si el funcionamiento en red no presenta interrupciones, con el fin de que no circulen telegramas de datos. Desde el punto de vista de la transmisión de datos, la topología en anillo se convierte en una línea. El administrador de redundancia vigila la topología en anillo, y envía telegramas de prueba desde el puerto en anillo 1 y el puerto en anillo 2. Estos telegramas recorren el anillo en ambas direcciones hasta que llegan al otro puerto del administrador de redundancia.

La causa de la interrupción del anillo puede ser un fallo de la conexión entre dos dispositivos o el fallo de un dispositivo en el anillo.

Si no circulan hasta el otro puerto en anillo los telegramas de prueba del administrador de redundancia al interrumpirse el anillo, este administrador interconecta ambos puertos en anillo. En esta ruta sustitutiva se vuelve a crear una conexión que funciona entre todos los dispositivos existentes con la forma de una topología en red con apariencia de línea.

Se denomina tiempo de reconfiguración al tiempo entre la interrupción del anillo y la recuperación de una topología en línea preparada para funcionar.

Tan pronto como se ha resuelto la interrupción, se vuelven a generar las rutas de transferencia originales, se separan los dos puertos en anillo en el administrador de redundancia y se informa del cambio a los clientes de la redundancia. Los clientes de la redundancia utilizan las rutas nuevas hasta los otros dispositivos.

Si se produce un fallo en el administrador de redundancia, el anillo se convierte en una línea preparada para funcionar.

Procedimientos de la redundancia de medios

En STEP 7 puede utilizar el procedimiento de la redundancia de medios MRP (Media Redundancy Protocol) para las topologías en anillo, ya que pueden participar hasta 50 dispositivos por anillo.

Media Redundancy Protocol (MRP)

El Media Redundancy Protocol (MRP)

El procedimiento "MRP" funciona de acuerdo con el Media Redundancy Protocol (MRP), que se especifica en la norma IEC 61158, tipo 10 "PROFINET".

El tiempo máximo de reconfiguración después de la interrupción del anillo es de 0,2 segundos.

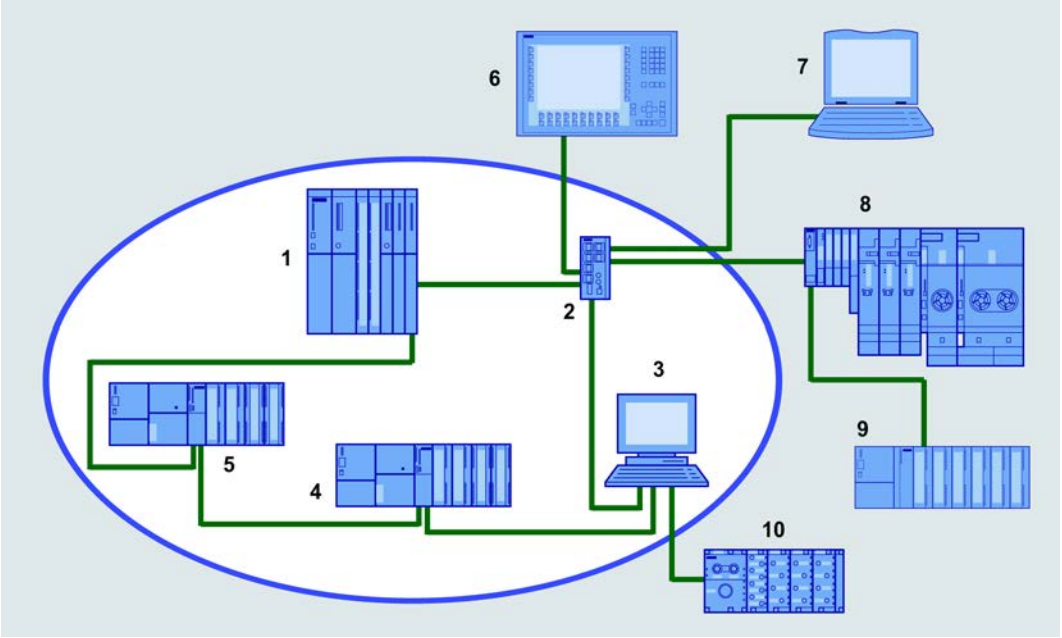
Requisitos para el funcionamiento sin averías

Los requisitos para el funcionamiento sin averías con el procedimiento de la redundancia de medios MRP son:

- MRP es compatible con topologías en anillo con hasta 50 dispositivos. Si se supera el número de dispositivos, puede causar un error en el intercambio de datos.
- Todos los dispositivos tienen que estar conectados entre sí mediante puertos en anillo.
- En todos los dispositivos del anillo debe estar activado "MRP".
- La configuración de la conexión (medio de transmisión / dúplex) debe estar ajustada en todos los puertos en anillo a dúplex y 100 Mbits/s como mínimo. En caso contrario, se puede producir un error en el intercambio de datos.
- En la configuración de STEP 7, en el diálogo de propiedades de todos los puertos incluidos en el anillo, en la ficha "Opciones" ajuste la conexión a "Configuración automática".
- El anillo en el que quiere utilizar MRP sólo puede estar compuesto de dispositivos compatibles con esta función. Se pueden utilizar, por ejemplo, los siguientes dispositivos:
  - Switches Industrial Ethernet como SCALANCE X 200 con versión de firmware V4.0 o superior y SCALANCE X 200 IRT con versión de firmware V4.0 o superior
  - Procesadores de comunicaciones como CP 443 1 Advanced (6GK7 443 1GX20 0XE0) con versión firmware V2.0 o superior, CP 343 1 Advanced (6GK7 343 1GX30 0XE0) con versión de firmware V1.0 o superior, CP 1616 (6GK1 161 6AA00) con versión de firmware V2.2 o superior, CP 1604 (6GK1 160 4AA00) con versión de firmware V2.2 o superior
  - Otros dispositivos MRP conformes con IEC 61158, tipo 10 "PROFINET".

Interconexión de puertos

En la siguiente figura se muestra una topología posible con dispositivos en un anillo con MRP. Los dispositivos dentro del óvalo blanco se encuentran en el dominio de la redundancia.



①	S7-400 con CP 443-1 Advanced
②	Switch SCALANCE X206-1
③	PC con CP 1616
④	S7-300 con CP 343-1 Advanced
⑤	S7-300 con CP 343-1 Advanced
⑥	Estación HMI
⑦	PG/PC
⑧	ET 200S
⑨	ET 200M
⑩	ET 200pro

Se aplican las siguientes reglas a la topología en anillo con redundancia de medios según el procedimiento MRP:

- Todos los dispositivos conectados de la topología en anillo pertenecen al mismo dominio de redundancia.
- Un dispositivo del anillo es el administrador de redundancia.
- Los demás dispositivos del anillo son los clientes de la redundancia.

Los dispositivos incompatibles con MRP pueden conectarse en el anillo mediante un switch SCALANCE X o un PC con CP 1616.

Arranque priorizado

Si configura MRP en un anillo, no podrá utilizar la función "Arranque priorizado" en los dispositivos que intervienen en las aplicaciones PROFINET .

Si quiere utilizar la función "Arranque priorizado", deberá desactivar MRP en la configuración.

Consulte también

Configurar la redundancia de medios (Página 1571)

Configurar la redundancia de medios

Requisitos

- Los componentes que intervienen tienen que ser compatibles con el Media Redundancy Protocol (MRP).
  Se trata, p. ej., de los componentes siguientes:
  - ET200S V7.0 o superior
  - ET 200SP
  - ET 200MP
  - CPUs S7-300 V3.2 o superior
  - CPUs S7-400 V6.0 o superior
  - CPUs S7-1500
- No hay configurada ninguna comunicación IRT, es decir, no hay ninguna interfaz PROFINET con configuración "Maestro Sync" ni "Esclavo Sync" (ventana de inspección "Opciones avanzadas > Configuración en tiempo real > Sincronización").

Procedimiento

Tiene que configurar la redundancia de medios en todos los dispositivos que deben funcionar con la redundancia de medios. Proceda del siguiente modo:

1. Configure los dispositivos PROFINET IO. Los dispositivos PROFINET deben ser compatibles con MRP.
2. Cree un anillo mediante interconexiones de puertos (p. ej. en la vista topológica).
   Si no se vulnera ninguna regla, se creará automáticamente un dominio MRP.

3. Controle las funciones de redundancia de medios, p. ej. cuál asume el papel de administrador de redundancia.
   Para ello, seleccione en las propiedades de la interfaz PROFINET, p. ej., de una CPU 319-3 PN/DP (FW 3.2), en "Configuración avanzada", la función de la redundancia de medios "Manager (Auto)". Puede haber varios dispositivos con este papel en un anillo; el administrador real se negocia entre los administradores (Auto).
   Los puertos en anillo predeterminados aparecen automáticamente en los campos "Puerto en anillo 1" o "Puerto en anillo 2", situados en la parte inferior.
   Si los puertos vienen definidos de fábrica, los campos aparecen desactivados. En caso contrario, seleccione el puerto que quiere configurar como puerto en anillo 1 o como puerto en anillo 2.
4. Seleccione para los demás dispositivos en el mismo lugar la función de redundancia de medios "Cliente".
   Los puertos en anillo predeterminados aparecen utomáticamente en los campos situados en la parte inferior.
5. Asigne a los dispositivos PROFINET que no quiera utilizar en una topología en anillo con MRP la opción "No es dispositivo del anillo".
6. Marque la opción "Alarmas de diagnóstico" si deben emitirse alarmas de diagnóstico sobre el estado de MRP en la CPU local.
7. Si un dispositivo PROFINET cuenta con más de dos puertos en un anillo, tiene que evitar que se introduzcan frames Sync en el anillo fijando un límite Sync (en los puertos que no se encuentran en el anillo). Puede realizar este ajuste en "Opciones avanzadas > Puerto [Xn Pm] > Opciones de puerto > Boundaries".
8. Cargue la configuración en los dispositivos que intervienen.
   La configuración de MRP es efectiva incluso después de efectuar un rearranque completo del dispositivo o tras un corte de alimentación y subsiguiente rearranque.

Reglas

- Todos los dispositivos deben ser compatibles con MRP.
- Solo está permitido que tenga la función de redundancia de medios "Manager" una interfaz PROFINET del anillo. No está permitido que otras interfaces PROFINET del anillo tengan la función "Manager" o "Manager (auto)".
- Puede haber varias interfaces PROFINET con la función de redundancia de medios "Manager (auto)".
- Al menos una interfaz PROFINET del anillo debe tener la función de redundancia de medios "Manager (auto)".
- Todas las interfaces PROFINET del anillo deben pertenecer al mismo dominio MRP. Una interfaz PROFINET solo puede pertenecer a un dominio MRP.

Alarmas de diagnóstico

Se pueden emitir las siguientes alarmas de diagnóstico:

- Fallos de cableado o de puerto
  Si se producen los siguientes fallos en los puertos en anillo se generan alarmas de diagnóstico:
  - Un vecino del puerto en anillo no es compatible con MRP
  - Un puerto en anillo está conectado a uno que no lo es
- Interrupción / recuperación (sólo administrador de redundancia)
  En caso de interrupción del anillo y de recuperación de la configuración original se generan alarmas de diagnóstico.
- Otras alarmas del administrador de redundancia
  Hay otras alarmas de diagnóstico, p. ej., si en el anillo hay varios dispositivos del tipo "Manager".

Nota

Con el fin de garantizar un funcionamiento correcto en caso de utilizar un dispositivo diferente como administrador de redundancia, tiene que asignar a los demás dispositivos del anillo la función "Cliente de la redundancia" antes de cerrar el anillo. En caso contrario, es posible que circulen telegramas de datos y se produzca un fallo de la red.

Consulte también

Media Redundancy Protocol (MRP) (Página 1569)

Administración de dominios MRP

Las configuraciones para redundancia de medios pueden controlarse y ajustarse de forma centralizada.

Requisitos

- Los componentes que intervienen tienen que ser compatibles con el Media Redundancy Protocol (MRP).
- Los dispositivos deben estar configurados y conectados en red.

Procedimiento

Para abrir el dominio MRP, proceda del siguiente modo:

1. Seleccione una interfaz PROFINET cualquiera y, en la ventana de inspección, seleccione el área "Opciones avanzadas > Redundancia de medios".
2. Haga clic en el botón "Ajustes del dominio".

O bien:

1. Seleccione la subred o el sistema IO.
2. En la ventana de inspección desplácese hasta el área "PROFINET > Administración de dominios > Dominios MRP".

Aquí se puede determinar de forma centralizada qué dominio MRP va a ser el dominio predeterminado; se pueden modificar los nombres de todos los dominios y revisar funciones de redundancia y puertos en anillo.

Para cada dominio MRP se ofrece un resumen donde se muestra cuántas interfaces PROFINET hay interconectadas en el anillo, cuántas interfaces PROFINET están fuera del anillo, y el número de administradores de redundancia y de clientes de redundancia.

Pueden mostrarse agrupados los dispositivos del anillo, con sus nombres, el dominio MRP correspondiente, la función de redundancia de medios y los puertos en anillo utilizados.

- Dominios MRP
  La lista muestra los dominios MRP existentes en el sistema IO seleccionado.
- <Nombre del dominio MRP>
  La lista muestra el número de interfaces PROFINET y de dispositivos con las correspondientes funciones de redundancia para el dominio MRP seleccionado.
- Dispositivos
  La lista muestra todos los dispositivos configurados del sistema IO seleccionado.

Configuración de PROFIenergy

Resumen de PROFIenergy

Ahorrar energía con PROFIenergy

PROFIenergy es una interfaz de datos basada en PROFINET que permite desconectar cargas de forma coordinada y centralizada durante pausas independientemente del fabricante y del equipo. De esta manera se pretende suministrar al proceso únicamente la energía absolutamente necesaria. La mayoría de la energía es ahorrada por el proceso, el dispositivo PROFINET sólo contribuye al potencial de ahorro algunos vatios.

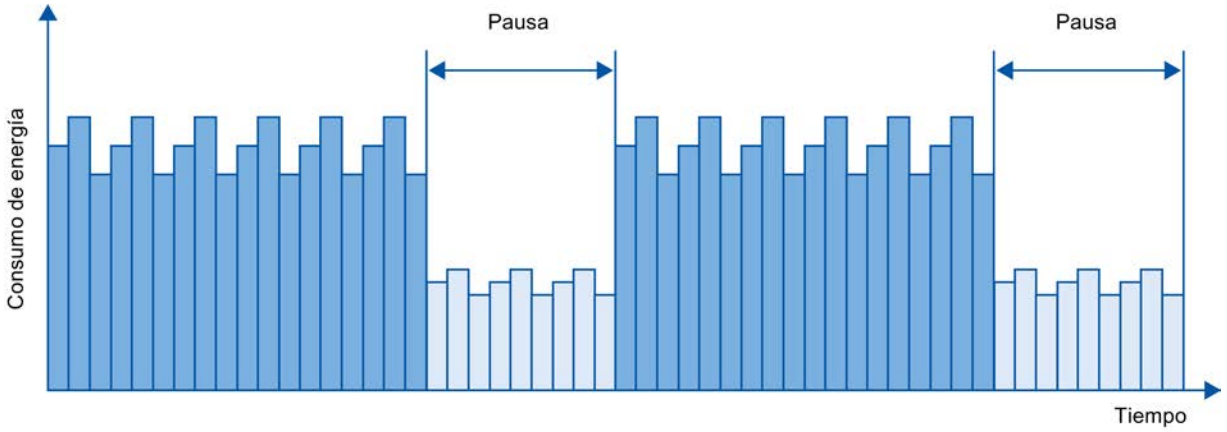


Figura 8-50 Ahorro de energía en pausas con PROFIenergy

Principios básicos

La desconexión de los dispositivos PROFINET o de los módulos de potencia se efectúa mediante comandos especiales en el programa de usuario del controlador PROFINET IO. Y no será necesario utilizar hardware adicional, los dispositivos PROFINET interpretan directamente los comandos de PROFIenergy.

Funcionamiento

Al principio y final de pausas, el responsable de la instalación activa o desactiva la función de pausa de la instalación; a continuación, el controlador IO envía el comando de PROFIenergy "Start_Pause" o "End_Pause" a los dispositivos PROFINET. El dispositivo interpreta posteriormente el contenido del comando de PROFIenergy y se desconecta/se vuelve a conectar.

A través de otras funciones adicionales de PROFIenergy se puede consultar información de los dispositivos durante las pausas. El usuario puede aprovecharlas para transmitir el comando "Start_Pause" o "End_Pause" a tiempo.

Instrucciones PROFIenergy para controladores IO

Para controlar y supervisar las funciones de PROFIenergy se necesitan dos instrucciones.

Con la instrucción PE_START_END se activa o desactiva de manera muy sencilla el estado de reposo de los dispositivos PROFINET. La activación o desactivación se realiza mediante un flanco entrante o un flanco saliente. La instrucción PE_START_END proporciona una interfaz sencilla para ejecutar los comandos de PROFIenergy Start_Pause y End_Pause.

Con la instrucción PE_CMD pueden transferirse todos los comandos de PROFIenergy, incluidos Start_Pause y End_Pause. Mediante los restantes comandos se pueden consultar, por ejemplo, el estado actual de los dispositivos PROFINET o el comportamiento durante las pausas. La instrucción PE_CMD permite un manejo cómodo de todas las funciones de PROFIenergy.

Instrucción de PROFIenergy para dispositivos I

Con la instrucción PE_I_DEV se puede ejecutar PROFIenergy también en dispositivos I. Esta instrucción recibe comandos de PROFIenergy en el dispositivo I y los pasa al programa de usuario para ser procesados. Una vez que el comando ha sido procesado por el programa de usuario, éste vuelve a llamar la instrucción PE_I_DEV para enviar el acuse al controlador IO. Para estas respuestas, el usuario dispone para cada comando de una instrucción auxiliar que proporciona a la instrucción los datos de respuesta.

Las instrucciones se encuentran en la Task Card "Instrucciones" del editor de programas de STEP 7.

Encontrará un ejemplo de aplicación en Internet (http://support.automation.siemens.com/WW/view/es/41986454) en el portal Service & Support.

Configuración y programación

Las funciones se pueden integrar fácilmente en las instalaciones existentes. Para la aplicación de PROFIenergy no se requiere ninguna configuración. Sin embargo, será necesario realizar algunas ampliaciones en el programa de usuario:

- Antes de emitir el comando "Start_Pause", el usuario debe asegurarse de que la instalación se encuentra en un estado seguro y apto para una pausa.
- Se deberá programar un control del tiempo restante para iniciar la pausa de los dispositivos y para volver a conectar las estaciones a tiempo que están paradas (en función de los tiempos necesarios de conexión y anticipación que requiere el dispositivo PROFINET correspondiente).
- Se deben evaluar los mensajes de error de la instrucción PE_CMD y programar la respuesta requerida en cada caso (p. ej. cancelación o continuación de otros comandos a dispositivos PROFINET subordinados).

Consulte también

PROFIenergy (Página 4820)

Configuración de PROFIenergy en ET 200S

El sistema de periferia descentralizada ET 200S presenta una particularidad: En determinados módulos de interfaz pueden configurarse cuáles son los grupos de potencial o los módulos de potencia del dispositivo IO que están activados para PROFIenergy.

Particularidades con módulos de interfaz ET 200S HF V7.0 y ET 200S FO V7.0

Se puede configurar el comportamiento inicial de los módulos de potencia (p. ej. el módulo de potencia PM-E DC24V/8A RO que puede conectarse) para los dispositivos IO mencionados, de manera que no sea necesario conmutar el dispositivo al modo de ahorro de energía en el programa de arranque mediante las instrucciones de PROFIenergy PE_Start_Ende_ Pause.

Active la función de ahorro de energía en las propiedades del módulo de potencia (área Parámetros del módulo, sección "PROFIenergy").

Tenga en cuenta que para aplicar las funciones de PROFIenergy es necesario adoptar ciertas precauciones en el programa de usuario.

Reglas

La posibilidad de configuración descrita más arriba no está disponible en los dispositivos ET 200S que han sido incorporados al catálogo de hardware mediante archivos GSDML.

Configuración de Shared Devices

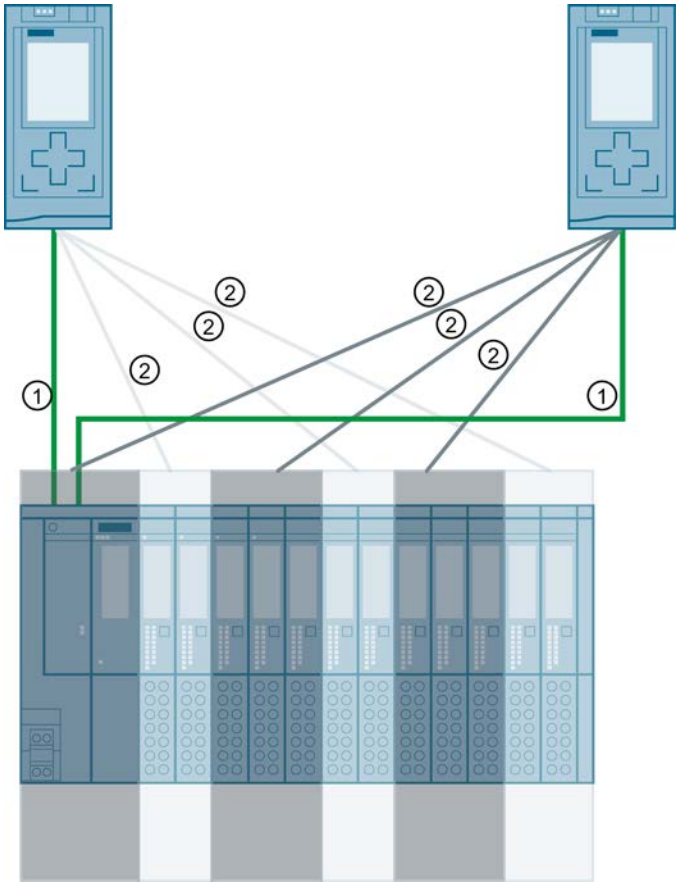
Información importante sobre los Shared Devices

Funcionalidad de 'Shared Device'

En instalaciones de mayor tamaño o de distribución extensa suelen utilizarse numerosos controladores IO.

Sin la función "Shared Device", cada módulo de periferia de un dispositivo IO está asignado al mismo controlador IO. Por tanto, si unos sensores que se encuentran muy próximos en el espacio deben suministrar datos a diferentes controladores IO, se necesitarán varios dispositivos IO.

La función "Shared Device" permite repartir los módulos o submódulos de un dispositivo IO entre varios controladores IO. Esta distribución hace posible realizar conceptos de automatización flexibles. Así, por ejemplo, se ofrece la posibilidad de agrupar módulos de periferia que se encuentren cerca en un único dispositivo IO.



① PROFINET
② Asignación lógica

Principio

El acceso a los submódulos del shared device se reparte entre los diferentes controladores IO. Cada submódulo del Shared Device está asignado exclusivamente a un controlador IO.

Requisitos (configuración GSD)

- STEP 7 a partir de V12 Service Pack 1
- CPU 15XX a partir de FW 1.1 como controlador IO
- El dispositivo IO soporta la función Shared Device, p. ej., el módulo de interfaz IM 155-5 PN ST
- El archivo GSD para la configuración del dispositivo IO está instalado

Observaciones:

- una CPU S7-1500 configurada como I-Device solo puede utilizarse como Shared Device a partir del firmware V 1.5. El I-Device debe integrarse en el sistema IO del controlador IO mediante una importación de GSD.
- Una CPU S7-300 o una CPU S7-400 configuradas como I-Device se pueden utilizar como Shared Device con la siguiente condición: El archivo GSD PROFINET para el I-Device debe exportarse desde STEP 7 (V5.5 y superiores) y luego importarse a STEP 7 (TIA Portal).

Configuración del acceso

Para conseguir que los módulos o submódulos de un dispositivo IO se puedan asignar a diferentes controladores IO, el dispositivo IO debe estar presente en varios proyectos. Para cada controlador IO se necesita un proyecto propio.

A través del parámetro "Shared Device" del módulo de interfaz se determina a qué módulos o submódulos tiene acceso el controlador IO:

- si el controlador IO local tiene acceso al módulo configurado, seleccione en la lista el nombre del controlador IO.
- Si el controlador IO que debe tener acceso al módulo configurado no es el controlador IO local sino el perteneciente a otro proyecto, seleccione la entrada "---".

Si cada módulo o submódulo está asignado a un controlador IO en un proyecto y solo uno, la configuración es coherente en cuanto al acceso.

Módulo o submódulo asignado a otro controlador IO

A continuación se tratarán las consecuencias del ajuste "---" del parámetro "Shared Device" desde el punto de vista del controlador IO local.

En este caso, el controlador IO local no tiene acceso al módulo configurado de esta forma. En particular, esto significa lo siguiente:

- No hay intercambio de datos con el módulo o el submódulo
- No hay recepción de alarmas y diagnósticos, es decir, no se muestra el estado de diagnóstico en la vista online
- No hay parametrización del módulo o submódulo

Ajuste de las características de tiempo real

STEP 7 calcula la carga de comunicación y con ella los tiempos de actualización resultantes. Para que sea posible un cálculo en configuraciones Shared Device, debe introducir el número de controladores IO externos al proyecto en el proyecto cuya interfaz PROFINET del Shared Device esté asignada al controlador IO.

El número máximo de controladores IO para el Shared Device depende del dispositivo. Este número se guarda en el archivo GSD del Shared Device.

En el caso de una CPU S7-1500 como controlador IO se puede ajustar un tiempo de ciclo de emisión muy corto. El tiempo de ciclo de emisión puede ser más corto que el tiempo de ciclo más corto soportado por el Shared Device. En este caso, el controlador IO hace funcionar el Shared Device con un tiempo de ciclo de emisión soportado por este (adaptación del tiempo de ciclo de emisión).

Ejemplo:

Una CPU S7-1500 soporta tiempos de ciclo de emisión a partir de 0,25 ms. Un dispositivo IO configurado soporta asimismo tiempos de ciclo de emisión a partir de 0,25 ms, y otro los soporta a partir de 1 ms. En este caso tiene la posibilidad de ajustar para la CPU el tiempo de ciclo de emisión corto de 0,25 ms. La CPU hace funcionar el dispositivo IO "lento" con el tiempo de ciclo de emisión de, p. ej., 1 ms.

Si no configura con STEP 7 uno de los controladores IO con acceso al Shared Device, esta función no es posible. En este caso debe leer en STEP 7 el tiempo de ciclo de emisión común en las propiedades del Shared Device e introducirlo en la herramienta de ingeniería para el controlador IO.

Reglas para la configuración

- Los controladores IO que utilizan el Shared Device se crean en diferentes proyectos. En cada proyecto hay que asegurarse de que el Shared Device esté configurado de forma idéntica en cada estación. Solo un controlador IO puede tener pleno acceso a un submódulo a la vez. Las incoherencias en la configuración provocan fallos en el Shared Device.
- Las direcciones de E/S de un módulo o submódulo solamente se pueden editar si un módulo o submódulo está asignado al controlador IO en el mismo proyecto.
- Un Shared Device debe tener en cada proyecto los mismos parámetros IP y el mismo nombre de dispositivo.
- El tiempo de ciclo de emisión debe ser igual en todos los controladores IO que tengan acceso al Shared Device.

- La ID de subred S7 a la que está conectado el Shared Device debe ser la misma en todos los proyectos.
- Solamente cuando la interfaz PROFINET del Shared Device está asignada al controlador IO local son posibles las siguientes funciones:
  - Modo IRT
  - Arranque optimizado
  - Parametrización de las propiedades del puerto

Condiciones marco

Puesto que una configuración de Shared Device se distribuye entre varios proyectos, resultan las siguientes condiciones marco:

- En la vista de direcciones de cada controlador IO que tiene acceso a un Shared Device faltan las direcciones de los módulos o submódulos que no están asignados a este controlador IO.
- Los módulos o submódulos no asignados no se incluyen en el cálculo del volumen de recursos para el Shared Device durante la comprobación de coherencia. Por tanto, el usuario debe verificar por sí mismo que no se supere el número máximo de submódulos o la cantidad máxima de datos IO cíclicos para el Shared Device.
- STEP 7 no detecta errores de configuración como la asignación de un módulo o submódulo a varios controladores IO.
- Las CPU que se cargan con una configuración de Shared Device no tienen información de que el dispositivo IO es un Shared Device. Los módulos o submódulos que están asignados a otros controladores IO y, por tanto, a otras CPU, no están presentes en la configuración cargada. Estos módulos o submódulos no se muestran, por tanto, ni en el servidor web de la CPU ni en la pantalla de la CPU.

Consulte también

Mostrar controladores IO que accedan a módulos de un Shared Device (Página 2173)

Configurar el I-Device como Shared Device (Página 1584)

Configurar un Shared Device (configuración GSD)

A continuación se describe cómo se configura un sistema de periferia descentralizada como Shared Device con STEP 7 V12, Service Pack 1 o superior.

En principio, es posible una configuración "distribuida" con diferentes herramientas de ingeniería para distintas familias de controladores IO. Sin embargo, la descripción del procedimiento está orientada exclusivamente a STEP 7 V12, Service Pack 1. La descripción se limita a dos controladores IO de la familia S7-1500 que comparten un Shared Device.

Se crean dos proyectos (Shared Device 1 y Shared Device 2) con un controlador IO cada uno (PLC1 y PLC2). El Shared Device debe crearse en los dos proyectos, aunque físicamente se trate de un único dispositivo IO.

Requisitos

- STEP 7 a partir de V12 Service Pack 1
- En el contexto de STEP 7 V12, Service Pack 1: solo CPU 15XX a partir de FW 1.1 configurable como controlador IO (no la CPU 31X, CPU 41X o CPU 12XX).
- En el contexto de otros controladores IO que deban tener acceso a un Shared Device: la correspondiente herramienta de ingeniería, p. ej. STEP 7 V5.5, Service Pack 3 para CPU 31x-3 PN/DP.
- El dispositivo IO soporta la función Shared Device, p. ej., el módulo de interfaz IM 155-5 PN ST V2.0.
- El archivo GSD para la configuración del dispositivo IO como Shared Device está instalado.

Procedimiento: creación del proyecto 1

Para crear el primer proyecto con un Shared Device, proceda del siguiente modo:

1. Inicie STEP 7.
2. Cree un nuevo proyecto con el nombre "Shared Device 1".
3. Inserte en la vista de red una CPU 1513-1 PN desde el catálogo de hardware. Asigne el nombre "PLC1".
4. Inserte un dispositivo IO con la función "Shared Device" desde el catálogo de hardware (catálogo de hardware: Otros dispositivos de campo > PROFINET IO > I/O).
5. Asigne al dispositivo IO el controlador IO "PLC1".
6. Haga doble clic en el dispositivo IO e inserte todos los módulos y submódulos necesarios desde el catálogo de hardware a la tabla con la vista de dispositivos.
7. Parametrice los módulos.
8. Guarde el proyecto.

Procedimiento - crear proyecto 2

Para crear el segundo proyecto con un Shared Device, proceda del siguiente modo:

1. Inicie STEP 7 otra vez.
   Se abre una nueva instancia de STEP 7.
2. Cree un nuevo proyecto con el nombre "Shared Device 2" en la nueva instancia.
3. Inserte en la vista de red una CPU 1513-1 PN. Asigne el nombre "PLC2".
4. Copie el dispositivo IO del proyecto "Shared Device 1" e insértelo en la vista de red del proyecto "Shared Device 2".
5. Asigne al dispositivo IO el controlador IO "PLC2".
6. Guarde el proyecto.

Los dos proyectos tienen ahora un dispositivo IO de idéntica estructura, que en el siguiente paso se debe parametrizar para los diferentes accesos a los controladores IO.

Procedimiento: parametrización de accesos al Shared Device

Los módulos y submódulos que se insertan en el Shared Device se asignan automáticamente a la CPU local. Para cambiar la asignación, proceda del siguiente modo:

1. Marque el módulo de interfaz en la vista de redes o en la vista de dispositivos del proyecto "Shared Device 1".
2. Seleccione el área "Shared Device" en la ventana de inspección.
   Una tabla muestra para todos los módulos configurados qué CPU tiene acceso al correspondiente módulo o submódulo. El ajuste predeterminado para todos los módulos y submódulos es que el acceso lo tiene la CPU local.
3. Para todos los módulos o submódulos que deban permanecer en el área de direcciones de la CPU local, deje el ajuste "PLC1".
   Para todos los módulos o submódulos que deben estar en el área de direcciones de la CPU del proyecto "Shared Device 2" ("PLC2"), seleccione el ajuste "---". Esto significa que un controlador IO fuera del proyecto debe tener acceso al módulo o submódulo.
4. Marque el módulo de interfaz en la vista de redes o en la vista de dispositivos del proyecto "Shared Device 2".
5. Seleccione el área "Shared Device" en la ventana de inspección.
   Una tabla muestra para todos los módulos configurados qué CPU tiene acceso al correspondiente módulo o submódulo.
6. Para todos los módulos o submódulos que deben estar en el área de direcciones de la CPU del proyecto "Shared Device 1" ("PLC1"), seleccione el ajuste "---".
7. A continuación, compruebe si para cada módulo o submódulo de ambos proyectos los ajustes para el acceso son "complementarios". Esto significa que cuando la CPU local tiene acceso en un proyecto, en el otro proyecto debe estar ajustada la opción "---" y viceversa. Particularidad: la opción "---" para la interfaz PROFINET y, con ello, también para los puertos, hace que los correspondientes parámetros solamente sean de lectura y no se puedan modificar. Los parámetros de la interfaz PROFINET y los parámetros de puerto solamente pueden editarse en el proyecto en el que la interfaz PROFINET está asignada a la CPU local. Independientemente de esto, los puertos se pueden interconectar en los dos proyectos.
8. Compruebe si están ajustados los mismos parámetros de dirección IP y los mismos nombres de dispositivo para el Shared Device en todos los proyectos.
   Compruebe si en todos los proyectos está ajustada la misma ID de subred S7 para la subred a la que está conectado el Shared Device (propiedades de subred, área "General" de la ventana de inspección).

Nota

Si realiza cambios en el Shared Device: ¡traslade a cada proyecto los cambios realizados en el Shared Device! ¡Tenga en cuenta que nunca puede acceder más de un controlador IO a un módulo o submódulo!

Procedimiento: adaptación de la configuración en tiempo real

Para garantizar que todos los controladores IO y Shared Devices funcionen con el tiempo de ciclo de emisión adecuado y que debido a la carga por comunicaciones los tiempos de actualización se calculen correctamente, se deben adaptar y comprobar los siguientes ajustes:

1. Seleccione el proyecto cuyo controlador IO tiene acceso a la interfaz PROFINET y a los puertos del Shared Device.
2. Marque el módulo de interfaz del Shared Device en la vista de redes.
3. En la ventana de inspección, vaya al área "Interfaz PROFINET > Opciones avanzadas > Configuración en tiempo real > Ciclo IO".
4. En el área "Shared Device", ajuste el número de controladores IO externos al proyecto. El número máximo depende del dispositivo IO (especificación en archivo GSD).
5. Para cada controlador IO que tenga acceso a módulos o submódulos del Shared Device, debe ajustarse el mismo tiempo de ciclo de emisión:
   - Si configura el controlador IO con STEP 7 (TIA Portal):
     Abra el proyecto correspondiente.
     Seleccione la interfaz PROFINET del controlador IO.
     Elija en la ventana de inspección el área "Opciones avanzadas > Configuración en tiempo real > Comunicación IO" y ajuste el mismo tiempo de ciclo de emisión.
   - Si configura el controlador IO con otra herramienta de ingeniería:
     Seleccione la interfaz PROFINET del Shared Device en STEP 7 (TIA Portal) y lea el tiempo de ciclo de emisión en el Shared Device (área "Opciones avanzadas > Configuración en tiempo real").
     Introduzca en la herramienta de ingeniería el tiempo de ciclo de emisión leído.
   - Particularidad: si configura en STEP 7 (TIA Portal o V5.5) **todos** los controladores IO que tienen acceso al Shared Device, en el controlador IO también se podrán ajustar tiempos de ciclo de emisión más cortos que los que admite el Shared Device (adaptación del tiempo de ciclo de emisión).

Compilación y carga

Las configuraciones de los distintos controladores IO se deben compilar y cargar sucesivamente en las CPUs.

Debido a la configuración distribuida en proyectos separados, STEP 7 no notifica errores de coherencia en caso de parametrización incorrecta del acceso. Ejemplos de parametrización incorrecta del acceso:

- Varios controladores IO tienen acceso al mismo módulo
- Los parámetros de dirección IP o los tiempos de ciclo de emisión no son idénticos

Estos fallos solo tienen efecto durante el funcionamiento y se notifican, p. ej., como errores de configuración.

Configurar el I-Device como Shared Device

A continuación se describe como caso especial cómo configurar una S7-1500 como I-Device con STEP 7 V13 o superior y utilizarla después en dos proyectos como Shared Device.

En principio, aquí también es posible una configuración "distribuida" con diferentes herramientas de ingeniería para distintas familias de controladores IO. El procedimiento descrito a continuación está orientado a STEP 7 V13 y se limita a una configuración con dos controladores IO de la familia S7-1500 que comparten las áreas de transferencia de un I-Device como Shared Device. El propio I-Device también es una CPU S7-1500.

Se crean tres proyectos con un controlador IO cada uno (PLC-I-Device, PLC_1 y PLC_2).

PLC-I-Device se utiliza para configurar el I-Device. La variante GSD PROFINET de PLC-I-Device se utiliza en los proyectos PLC_1 y PLC_2 para poder asignar las áreas de transferencia en el correspondiente controlador IO de nivel superior.

Concepto de Shared I-Device

Para introducir el concepto de Shared I-Device hay que explicar dos funciones:

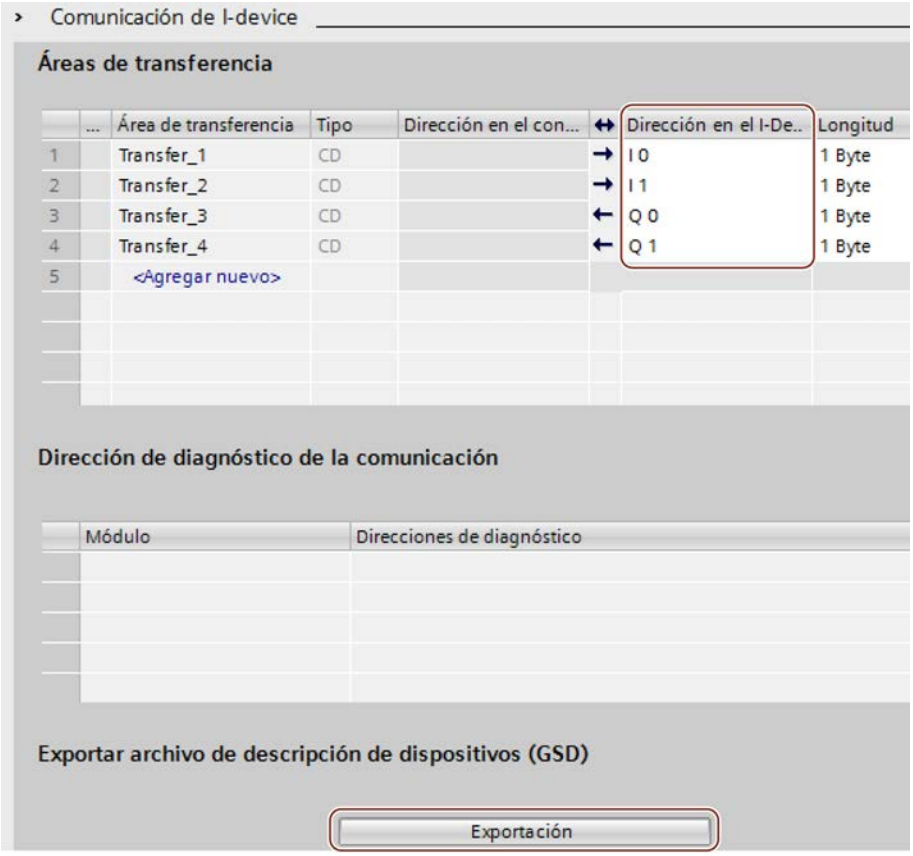
- La función de un fabricante (p. ej. un fabricante de máquinas): configura y programa un I-Device que asume una tarea de automatización determinada. Como interfaz de periferia con el operador de la máquina se definen áreas de transferencia que pueden asignarse a distintos controladores IO. Para la conexión a controladores IO de nivel superior, el fabricante proporciona un archivo GSD PROFINET y abre el acceso a las áreas de transferencia a través de las cuales se puede acceder al I-Device.
- La función del operador: utiliza el I-Device como variante GSD PROFINET durante la configuración de su sistema PROFINET IO y al hacerlo determina las direcciones E/S con las que los controladores IO acceden a las áreas de transferencia.

Vista del fabricante

Una CPU S7-1500 se parametriza como I-device con periferia centralizada y descentralizada, con las áreas de transferencia deseadas y con el número de controladores IO con acceso a este I-device (en el caso de Shared Device siempre mayor que 1).

Particularidad: el I-Device se configura sin controlador IO de nivel superior. De este modo, solo se dispone de las direcciones E/S locales del área de transferencia (= "Dirección en el I-Device") para crear el programa de usuario con el que se editarán las direcciones del área de transferencia. El I-Device, totalmente configurado a excepción de la conexión con el controlador IO de nivel superior, se carga en la CPU S7-1500.

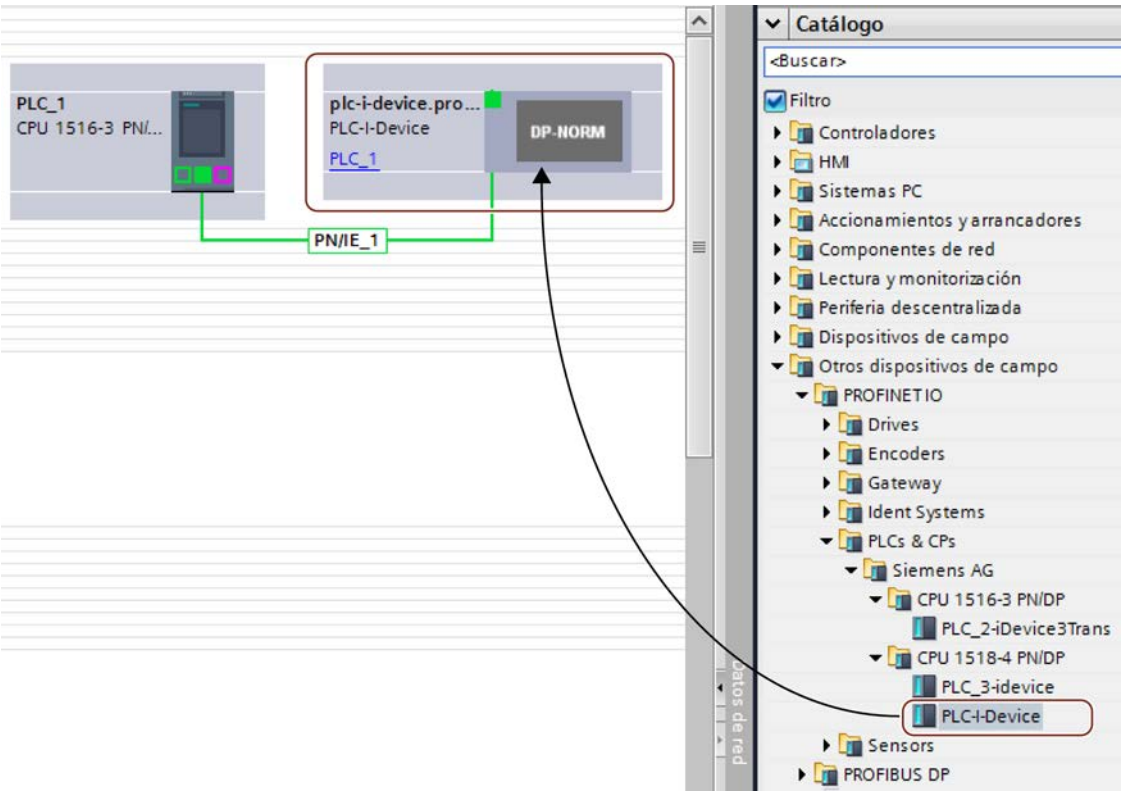
De la configuración del I-Device se exporta un archivo GSD PROFINET.



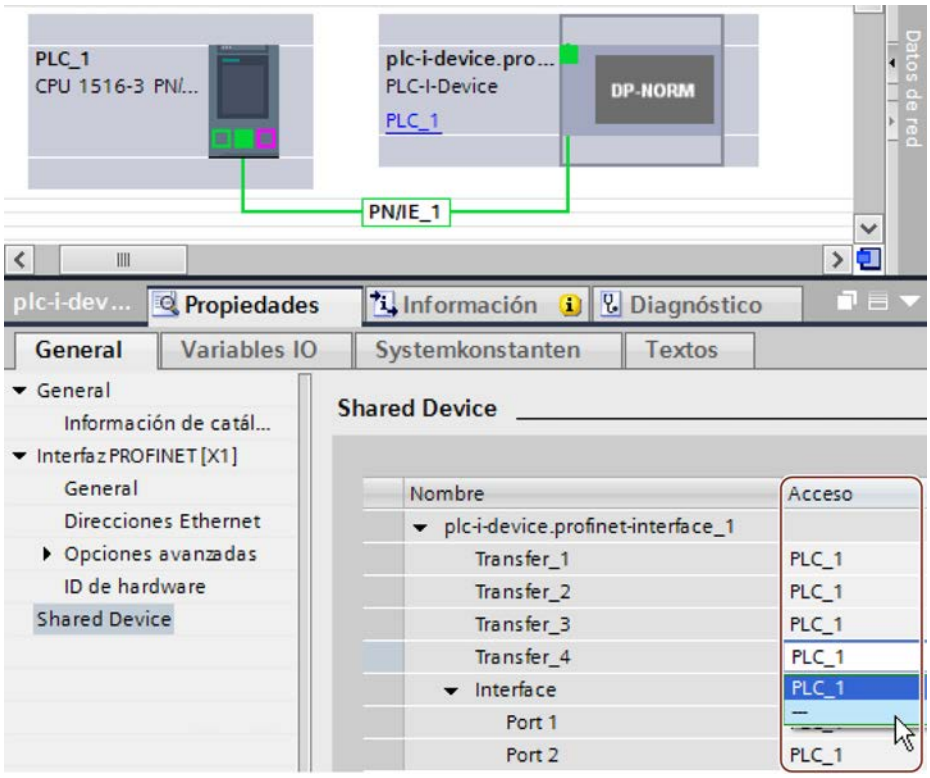
Vista de operador

El archivo GSD PROFINET creado a partir de la configuración del I-Device debe instalarse en todos los sistemas de ingeniería que intervengan en la configuración de un sistema PROFINET IO con este Shared I-Device. Si todos los usos de este I-Device se han configurado con STEP 7 V13, es suficiente instalar el archivo GSD en STEP 7.

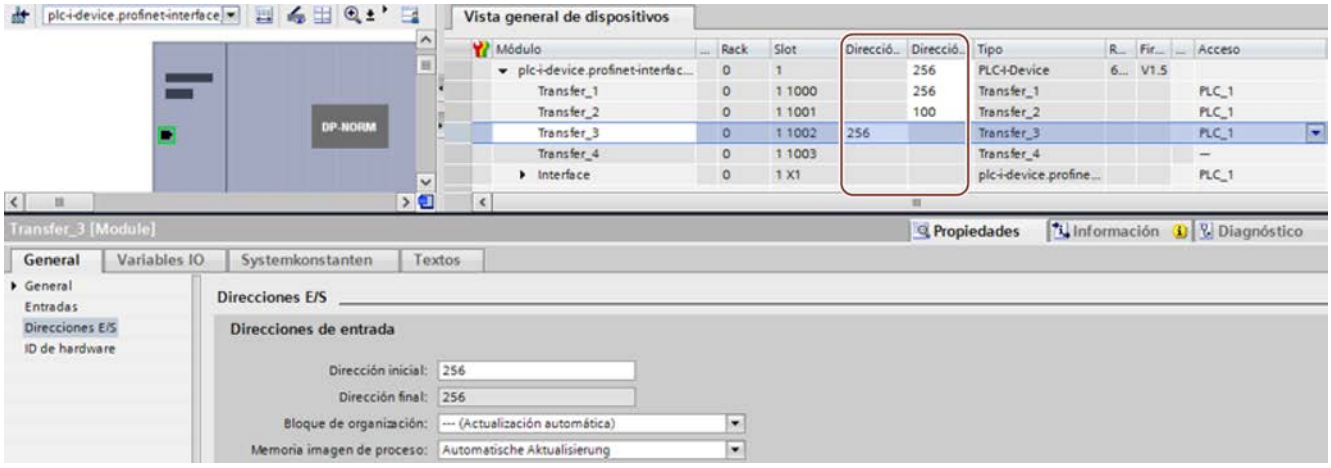
El I-Device se configura como variante GSD en el sistema PROFINET IO para los proyectos implicados. En STEP 7 V13, después de la instalación este I-Device se encuentra en "Otros dispositivos de campo > PROFINET IO > PLCs y CPs".



En cada uno de los proyectos implicados se determina qué áreas de transferencia están asignadas exclusivamente al controlador IO de nivel superior (valor predeterminado: todas). Las áreas de transferencia restantes se ajustan a "---" (no asignadas). Este ajuste provoca que el controlador IO local no tenga acceso a esta área de transferencia y, por lo tanto, pueda asignarse a otro controlador IO en otro proyecto.



Las direcciones desde el punto de vista del controlador IO se adaptan en la vista general de dispositivos. La vista general de dispositivos se abre haciendo doble clic en el I-Device.



Requisitos

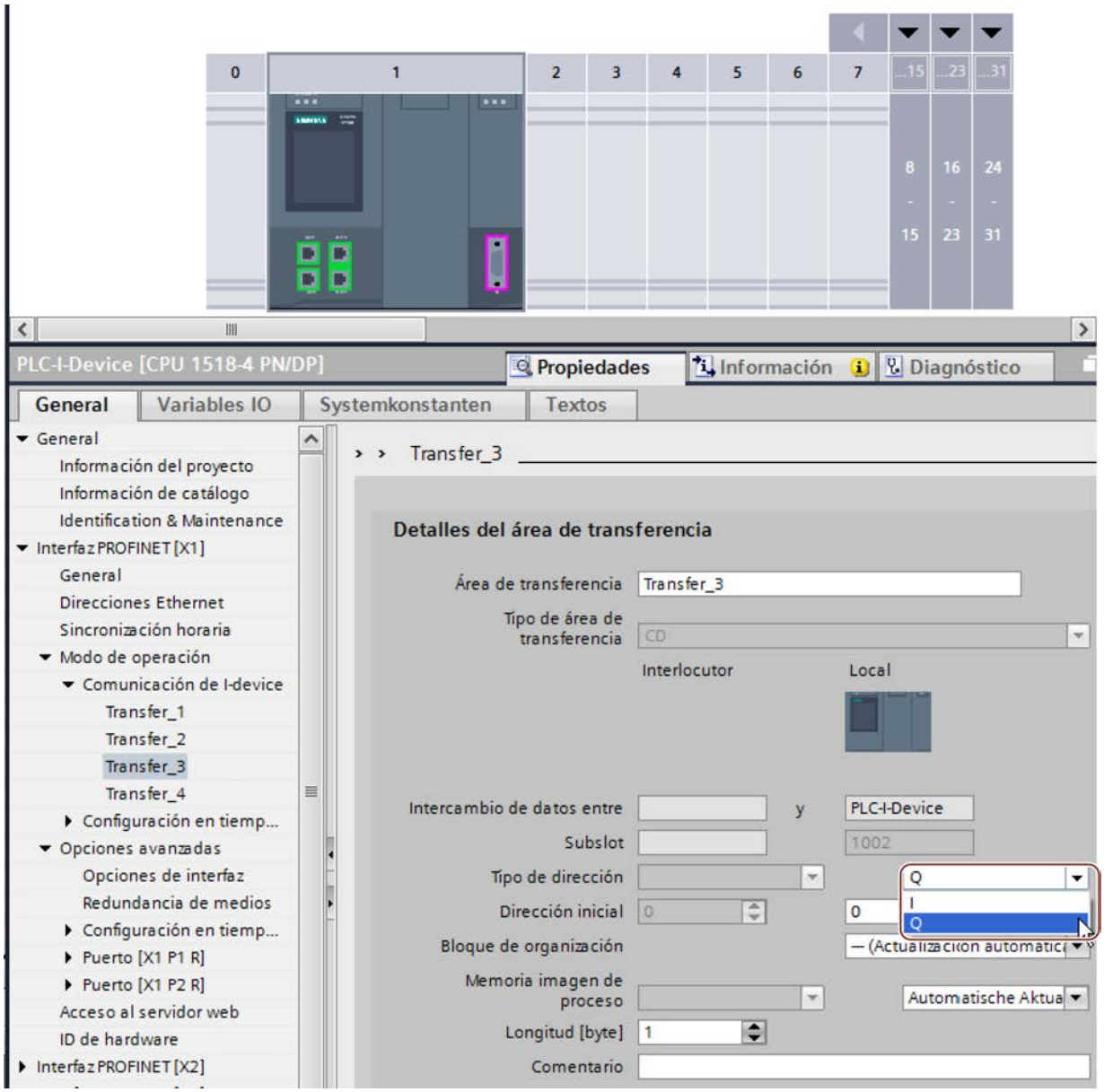
- STEP 7 V13 o superior
- En el contexto de STEP 7 V13: solo CPU 15XX a partir de FW 1.5 configurable como Shared I-Device (no una CPU 31X, CPU 41X o CPU 12XX).
- En el contexto de otros controladores IO que deban tener acceso a un Shared I-Device: la correspondiente herramienta de ingeniería, p. ej. STEP 7 V5.5, Service Pack 3 para CPU 31x-3 PN/DP.

Procedimiento: creación del proyecto PLC-I-Device

Para crear el proyecto con un Shared I-Device, proceda del siguiente modo:

1. Inicie STEP 7.
2. Cree un proyecto con el nombre "PLC-I-Device".
3. Inserte en la vista de redes una CPU 1518-4 PN/DP desde el catálogo de hardware. Asigne el nombre "PLC-I-Device".
4. Haga doble clic en el dispositivo IO y configure todos los módulos y submódulos necesarios.

5. Parametrice los módulos.
   Especialmente para la CPU son necesarios los siguientes ajustes en el área de la interfaz PROFINET [X1]:
   - Active la opción "Dispositivo IO" en el área "Modo de operación".
   - Configure las áreas de transferencia en el área "Modo de operación" > "Configuración de I-Device". La columna "Dirección en el controlador IO" queda vacía, dado que no se ha asignado ningún controlador IO.
     Nota: para convertir un área de entrada en un área de salida y viceversa, se debe ir al área de transferencia correspondiente.



   - Seleccione el número de controladores IO que pueden acceder al Shared I-Device durante el funcionamiento (área "Modo de operación" > "Configuración en tiempo real", apartado "Shared Device").

6. Guarde el proyecto.
7. Haga clic en el botón "Exportar" (área "Modo de operación" > "Configuración de I-Device", apartado "Exportar archivo de descripción del dispositivo (GSD)").
   Si no se cambia el nombre en el cuadro de diálogo Exportar, el archivo GSD tendrá, p. ej., un nombre de tipo "GSDML-V2.31-#Siemens-PreConf_PLC-I-Device-20130925-123456".

Procedimiento: creación del proyecto PLC_1

Para crear el primer proyecto con un Shared I-Device, proceda del siguiente modo:

1. Inicie STEP 7.
2. Instale el archivo GSD PROFINET de la exportación de la CPU I-Device (PLC-I-Device).
3. Cree un nuevo proyecto con el nombre "PLC_1".
4. Inserte en la vista de redes una CPU 1516-3 PN/DP. El nombre de la CPU debe ser "PLC_1".
5. Inserte el I-Device del catálogo de hardware (catálogo de hardware: Otros dispositivos de campo > PROFINET IO > PLCs y CPs).
6. Asigne al I-Device el controlador IO "PLC_1".
7. En las propiedades del I-Device, seleccione el área "Shared Device".
   En la tabla, todas las áreas de transferencia y la interfaz PROFINET están asignadas al controlador IO local (PLC_1).
8. Determine a qué áreas de transferencia **no** debe tener acceso la CPU PLC_1. Para esas áreas, seleccione la entrada "---". Esas áreas de transferencia están previstas para PLC_2.
9. Guarde el proyecto.

Procedimiento: creación del proyecto PLC_2

Para crear el segundo proyecto con un Shared Device, proceda del siguiente modo:

1. Inicie STEP 7 otra vez.
   Se abre una nueva instancia de STEP 7.
2. Cree un proyecto con el nombre "PLC_2" en la nueva instancia.
3. Inserte en la vista de redes una CPU 1516-3 PN/DP. Asigne el nombre "PLC_2".
4. Inserte el I-Device del catálogo de hardware (catálogo de hardware: Otros dispositivos de campo > PROFINET IO > PLCs y CPs).
5. Asigne al I-Device el controlador IO "PLC_2".
6. Adapte los accesos a las áreas de transferencia como en el proyecto PLC_1. Asegúrese de que no se produzca ninguna doble asignación.
7. Adapte los parámetros de subred e interfaz PROFINET. Puesto que el Shared I-Device es el mismo dispositivo en varios proyectos, estos datos deben coincidir.
8. Guarde el proyecto.

Ambos proyectos tienen ahora un Shared I-Device de idéntica estructura. Los accesos a los controladores IO y los parámetros de la interfaz PROFINET deben comprobarse en el siguiente paso en los diferentes proyectos.

Resumen: parametrización de accesos al Shared Device

Las áreas de transferencia se asignan automáticamente al controlador IO local. Para cambiar la asignación, proceda del siguiente modo:

1. En la vista de redes del proyecto "PLC_1", haga clic en el dispositivo "PLC_I-Device" y seleccione el área "Shared Device".
2. Una tabla muestra qué CPU tiene acceso a cada una de las áreas de transferencia configuradas. El ajuste predeterminado para todos los módulos y submódulos es que el acceso lo tiene la CPU local.
3. Para todas las áreas de transferencia que deban permanecer en el área de direcciones de la CPU local, deje el ajuste "PLC_1".
   Para todas las áreas de transferencia que deban estar en el área de direcciones de la CPU "PLC_2" del proyecto "PLC_2", seleccione el ajuste "---". Esto significa que un controlador IO fuera del proyecto debe tener acceso al área de transferencia.
4. Proceda de forma análoga con los demás proyectos.
5. A continuación, compruebe si para cada módulo o submódulo de ambos proyectos los ajustes para el acceso son "complementarios". Esto significa que cuando la CPU local tiene acceso en un proyecto, en el otro proyecto debe estar ajustada la opción "---" y viceversa. Particularidad: la opción "---" para la interfaz PROFINET y, con ello, también para los puertos, hace que los correspondientes parámetros solamente sean de lectura y no se puedan modificar. Los parámetros de la interfaz PROFINET y los parámetros de puerto solamente pueden editarse en el proyecto en el que la interfaz PROFINET está asignada a la CPU local. Independientemente de esto, los puertos se pueden interconectar en los dos proyectos.
6. Compruebe si están ajustados los mismos parámetros de dirección IP y los mismos nombres de dispositivo para el Shared Device en todos los proyectos.
   Compruebe si en todos los proyectos está ajustada la misma ID de subred S7 para la subred a la que está conectado el Shared Device (propiedades de subred, área "General" de la ventana de inspección).

Nota

Si realiza cambios en el I-Device (p. ej., cambios en el número o la longitud de las áreas de transferencia):

Vuelva a exportar el I-Device como archivo GSD. Vuelva a instalar el archivo GSD en cada proyecto que utilice el I-Device como Shared Device. Tenga en cuenta que nunca puede acceder más de un controlador IO a un área de transferencia.

Procedimiento: adaptación de la configuración en tiempo real

Para garantizar que todos los controladores IO y Shared Devices funcionen con el tiempo de ciclo de emisión adecuado y que debido a la carga por comunicaciones los tiempos de actualización se calculen correctamente, se deben adaptar y comprobar los siguientes ajustes:

1. Para cada controlador IO que tenga acceso a módulos o submódulos del Shared Device, debe ajustarse el mismo tiempo de ciclo de emisión:
   - Si configura el controlador IO con STEP 7 (TIA Portal):
     Abra el proyecto correspondiente.
     Seleccione la interfaz PROFINET del controlador IO.
     Elija en la ventana de inspección el área "Opciones avanzadas > Configuración en tiempo real > Comunicación IO" y ajuste el mismo tiempo de ciclo de emisión.
   - Si configura el controlador IO con otra herramienta de ingeniería:
     Seleccione la interfaz PROFINET del Shared Device en STEP 7 (TIA Portal) y lea el tiempo de ciclo de emisión en el Shared Device (área "Opciones avanzadas > Configuración en tiempo real").
     Introduzca en la herramienta de ingeniería el tiempo de ciclo de emisión leído.
   - Particularidad: si configura en STEP 7 (TIA Portal o V5.5) **todos** los controladores IO que tienen acceso al Shared I-Device, en el controlador IO también se podrán ajustar tiempos de ciclo de emisión más cortos que los que admite el Shared Device (adaptación del tiempo de ciclo de emisión).

Compilación y carga

Las configuraciones de los distintos controladores IO se deben compilar y cargar sucesivamente en las CPUs.

Debido a la configuración distribuida en proyectos separados, STEP 7 no notifica errores de coherencia en caso de parametrización incorrecta del acceso. Ejemplos de parametrización incorrecta del acceso:

- Varios controladores IO tienen acceso al mismo módulo
- Los parámetros de dirección IP o los tiempos de ciclo de emisión no son idénticos

Estos fallos solo tienen efecto durante el funcionamiento y se notifican, p. ej., como errores de configuración.

Consulte también

Configurar un I-Device (Página 1550)

Información importante sobre los Shared Devices (Página 1577)

Shared Input/Shared Output interna del módulo (MSI/MSO)

Introducción

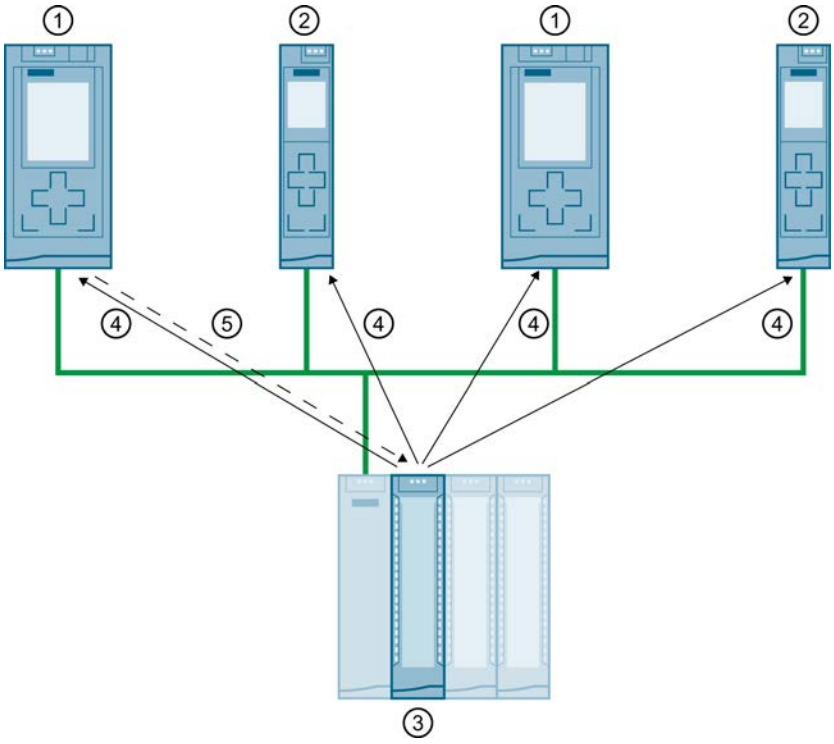
En este capítulo se describe la funcionalidad "Shared Input/Shared Output interna del módulo (MSI/MSO)" para los módulos de periferia S7-1500 que operan en un ET 200M conectado a PROFINET.

Funcionalidad "Shared Input/Shared Output interna del módulo"

La funcionalidad "Shared Input interna del módulo (MSI)" permite que un módulo de entrada ponga sus datos de entrada a disposición de hasta cuatro controladores IO. Cada controlador puede tener acceso de lectura a los mismos canales.

La funcionalidad "Shared Output interna del módulo (MSO)" permite que un módulo de salida ponga sus datos de salida a disposición de hasta cuatro controladores IO. El controlador IO tiene acceso de escritura. Los demás controladores IO pueden tener acceso de lectura a los mismos canales.

La figura siguiente muestra la funcionalidad MSI/MSO.



① CPU 1516-3 PN/DP como controlador IO

② CPU 1511-1 PN como controlador IO

③ Módulo E/S con MSI/MSO

④ Acceso de lectura a los canales del módulo E/S

⑤ Acceso de escritura a los canales del módulo E/S (solo en MSO)

Ventajas de MSI/MSO

La funcionalidad "Shared Input/Shared Output interna del módulo" ofrece las siguientes ventajas:

- Lectura en tiempo real en varias CPUs
- Menos costes gracias al ahorro de dispositivos IO y módulos adicionales
- Menos necesidad de espacio gracias al ahorro de dispositivos IO y módulos adicionales
- Menos carga de comunicación al suprimirse la comunicación CPU-CPU
- No es necesario ningún trabajo adicional de programación para la comunicación CPU-CPU

Requisitos para el uso de MSI/MSO

Tenga en cuenta los siguientes requisitos:

- El uso de MSI/MSO solo es posible en PROFINET IO.
- Software de configuración: STEP 7 (TIA Portal) V12 SP1 y superiores con archivo GSD.
- El módulo de interfaz IM 155-5 PN ST y los módulos soportan MSI/MSO a partir de la versión de firmware V2.0.0.

Condiciones para el uso de MSI/MSO

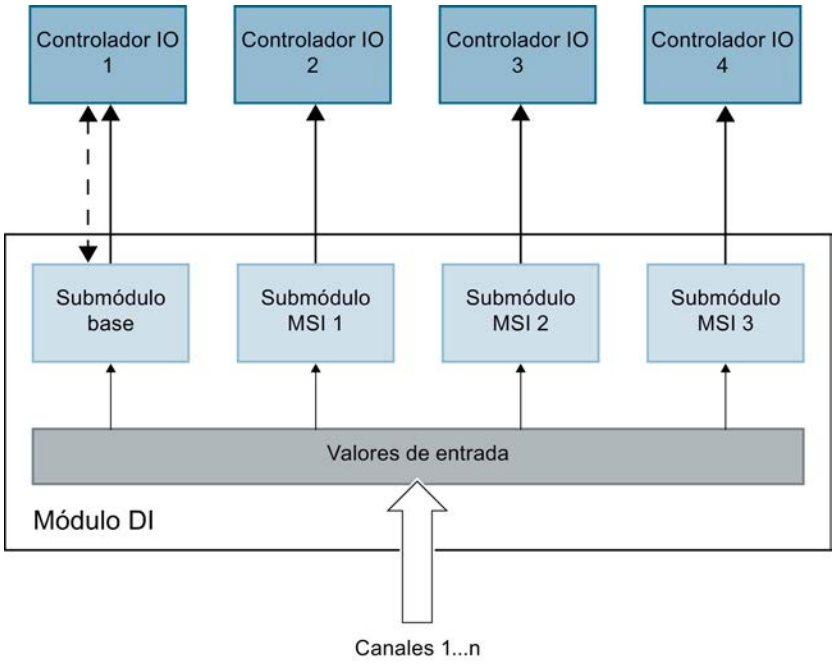
Tenga en cuenta las siguientes condiciones marco:

- En los módulos con grupos de canales no es posible utilizar MSI/MSO.
- Los módulos con MSI/MSO no se pueden utilizar en modo isócrono.

Submódulos MSI

En la configuración MSI de un módulo de entrada se copian los valores de entrada de todos los canales en un submódulo base y en hasta tres submódulos MSI adicionales. De este modo los canales del módulo están disponibles con valores de entrada idénticos en el submódulo base y en los submódulos MSI. Los submódulos pueden asignarse a hasta cuatro controladores IO si el módulo se utiliza en un Shared Device. Cada controlador IO puede tener acceso de lectura a los mismos canales.

La figura siguiente ilustra un módulo de entradas digitales con el submódulo base y tres submódulos MSI. Cada submódulo está asignado a un controlador IO. El diagnóstico y la parametrización del módulo de entradas digitales también son posibles a través del submódulo base del controlador IO 1.



Información de calidad del submódulo MSI (Quality Information, QI)

El significado de la información de calidad depende del submódulo en cuestión.

En el submódulo base (= submódulo 1), la información de calidad 0 indica que el valor es erróneo.

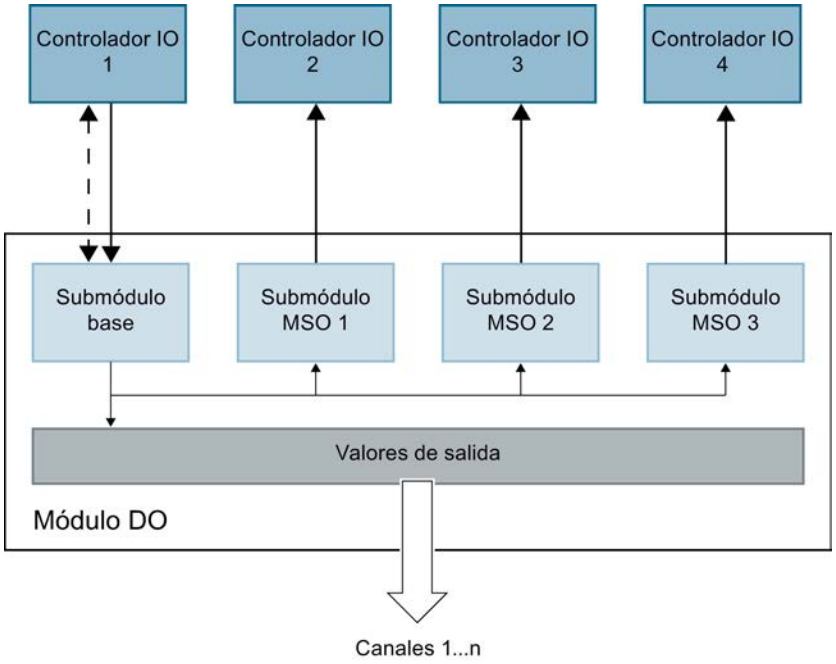
En un submódulo MSI (submódulos 2 a 4), la información de calidad 0 indica que el valor es erróneo o que el submódulo base aún no se ha parametrizado (no está listo para el funcionamiento).

Submódulos MSO

En la configuración MSO de un módulo de salida se copian los valores de salida de todos los canales del módulo de un submódulo base en hasta tres submódulos MSO adicionales. De este modo los canales del módulo están disponibles con valores idénticos en el submódulo base y en los submódulos MSO. Los submódulos MSO pueden asignarse a hasta cuatro controladores IO si el módulo se utiliza en un Shared Device.

- El controlador IO al que está asignado el submódulo base tiene acceso de escritura a las salidas del módulo. En consecuencia, el submódulo base ocupa direcciones de salida en la memoria imagen de proceso del controlador IO.
- Los controladores IO a los que están asignados los submódulos MSO tienen acceso de lectura a las salidas del módulo. En consecuencia, los submódulos MSO ocupan direcciones de entrada en la memoria imagen de proceso del controlador IO.

La figura siguiente ilustra un módulo de salidas digitales con el submódulo base y tres submódulos MSO. Cada submódulo está asignado a un controlador IO. El diagnóstico y la parametrización del módulo de salidas digitales también son posibles a través del submódulo base del controlador IO 1.



Información de calidad del submódulo MSO (Quality Information, QI)

El significado de la información de calidad depende del submódulo en cuestión.

En el submódulo base (= submódulo 1), la información de calidad 0 indica que el valor es erróneo.

En un submódulo MSO (=submódulos 2 a 4), el la información de calidad 0 indica que el valor es erróneo o que se ha producido uno de los siguientes errores:

- El submódulo base todavía no está parametrizado (no está operativo).
- Se ha interrumpido la conexión entre el controlador IO y el submódulo base.
- El controlador IO del submódulo base está en estado STOP o DESCONEXIÓN.

Configuración de módulos de periferia con submódulos MSI/MSO

Requisitos

- Software de configuración STEP 7 V12 Service Pack 1 o superior
- Archivo GSD para módulos de periferia con MSI/MSO.
- El dispositivo IO soporta MSI/MSO (p. ej. IM 155-5 PN ST a partir de la versión de firmware V2.0.0)

Procedimiento

1. Inserte un módulo de interfaz IM 155-5 PN ST V2.0 o superior en la vista de redes de STEP 7.
2. Haga doble clic en el dispositivo IO.
   Ahora se encuentra en la vista de dispositivos.
3. Coloque los módulos con MSI/MSO del catálogo de hardware en un slot adecuado.
4. Configure los módulos tal y como se describe en la información del producto correspondiente al módulo.

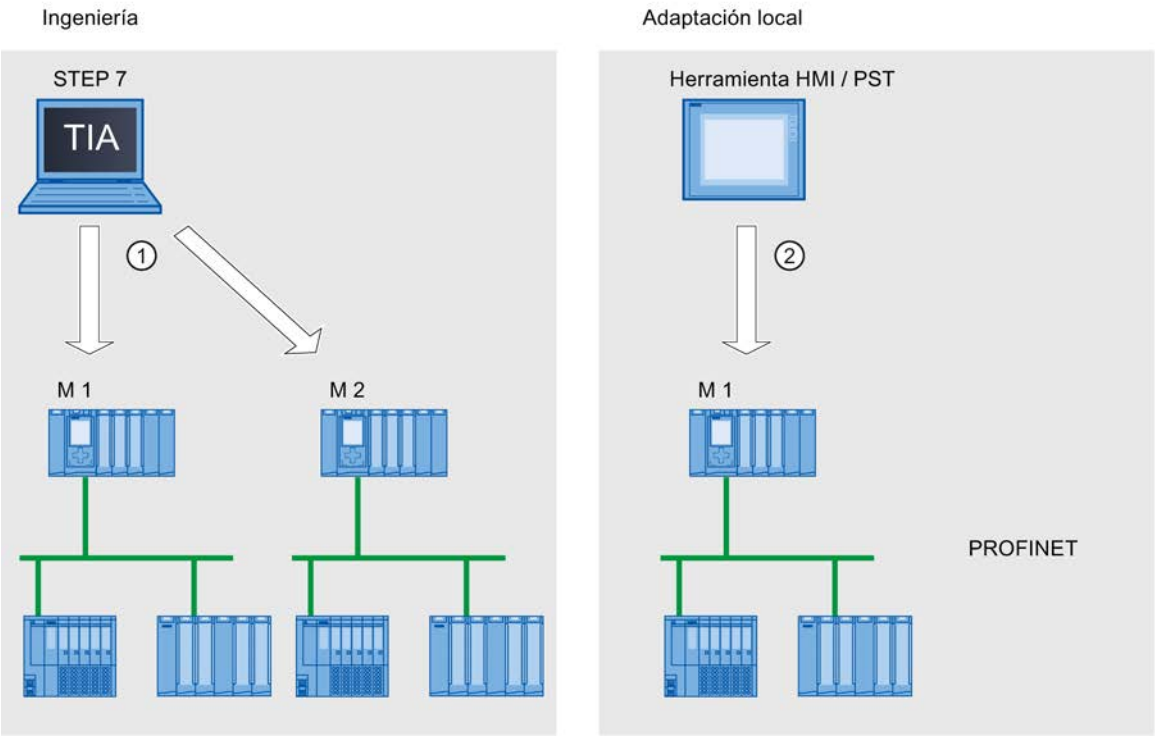
Crear un proyecto de maquinaria de serie

Información importante sobre proyectos de maquinaria de serie

Soluciones de automatización de múltiples aplicaciones

Para una solución de automatización flexible, como en el caso de la maquinaria de serie, se aplican los requisitos siguientes:

- La solución de automatización (máquina) se utiliza varias veces en las instalaciones de los clientes.
- La máquina se utiliza en diferentes instalaciones de distintos clientes.
- Un proyecto gestionado de forma centralizada (configuración y programa) puede cargarse sin cambios en diferentes máquinas del mismo tipo.
- Localmente solo son necesarias algunas adaptaciones sencillas para integrar la máquina en una infraestructura de red existente.



① Cargar configuración con un sistema IO de múltiples aplicaciones

② Ajustar la dirección IP y el nombre de dispositivo de forma local en el controlador IO

Principio

Los componentes de automatización para una máquina incluyen un sistema PROFINET IO, compuesto de un controlador IO (interfaz PROFINET de una CPU) y los dispositivos IO asignados a él.

El ajuste "Sistema IO de múltiples aplicaciones" en el sistema IO convierte un proyecto STEP 7 en un "proyecto de maquinaria de serie".

Con el ajuste "Sistema IO de múltiples aplicaciones" en el sistema IO, STEP 7 realiza diversos ajustes y comprobaciones de la configuración. Estos ajustes permiten que el sistema IO esté autocontenido y no existan dependencias de componentes externos al sistema IO.

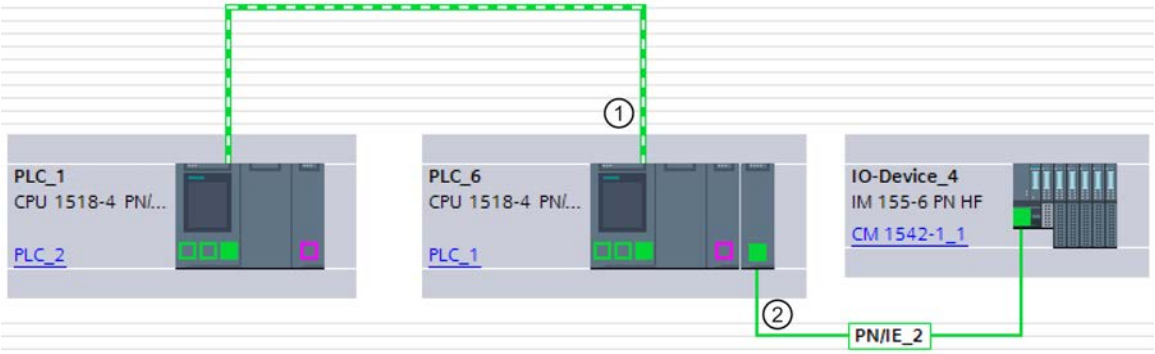
Requisitos

- STEP 7 V13 o superior
- CPU 15XX a partir de la versión de firmware V1.5 como controlador IO

Reglas

Para los sistemas IO de múltiples aplicaciones se aplican las siguientes reglas:

- Ningún dispositivo IO puede estar configurado como Shared Device.
- Los puertos de los dispositivos deben estar interconectados.
  Los dispositivos para los que no puede configurarse ninguna interconexión de puertos, como p. ej. módulo de interfaz IM 154-6 IWLAN (ET200pro PN), no se pueden utilizar con STEP 7 V13 como dispositivo IO en el sistema IO de múltiples aplicaciones.
- Si un dispositivo IO conectado al sistema IO de múltiples aplicaciones es un I-Device (CPU como dispositivo IO):
  - Si el I-Device tiene un sistema IO subordinado, este I-Device no debe estar conectado a la misma interfaz PROFINET que el controlador IO de nivel superior. Por ello una interfaz PROFINET conectada al I-Device solo debe operarse como "Controlador IO" o "Dispositivo IO".
    Nota: si el I-Device está configurado a través de un GSD PROFINET, STEP 7 no puede comprobar si se cumple esta regla. En tal caso, el propio usuario debe procurar que se cumpla esta regla.



① I-Device conectado al sistema IO de múltiples aplicaciones. La interfaz PROFINET se opera como dispositivo IO. No hay ningún otro sistema IO conectado.

② Un sistema IO subordinado conectado al I-Device está conectado a otra interfaz PROFINET.

  - La interfaz PROFINET del I-Device debe estar ajustada a "Parametrización de la interfaz PN por el controlador OP de nivel superior".
- Si está configurado MRP (Media Redundancy Protocol):
  - Todos los dispositivos IO del sistema IO de múltiples aplicaciones deben pertenecer al mismo dominio MRP.
- Si está configurado IRT (Isochronous Real Time):
  - Todos los dispositivos IO del sistema IO de múltiples aplicaciones deben pertenecer al mismo dominio Sync.
  - El dominio Sync no debe contener otros dispositivos IO.
- Los IE/PB Links no pueden operarse con STEP 7 V13 como dispositivo IO conectado al sistema IO de múltiples aplicaciones.

Configuración

En las propiedades del sistema IO se define si una configuración se puede usar varias veces.

Los demás ajustes de parámetros de los dispositivos configurados se establecen automáticamente en STEP 7 y se comprueban durante la compilación.

Condiciones marco

Para evitar que un proyecto de maquinaria de serie tenga dependencias de otros dispositivos externos a la máquina, tenga en cuenta lo siguiente:

- Configure en el proyecto de maquinaria de serie solo una CPU como controlador IO y los dispositivos IO correspondientes.
- No utilice para la comunicación conexiones bilaterales, sino solamente conexiones unilaterales o conexiones no especificadas si es necesario.
- Para la comunicación, utilice Open User Communication y asegúrese de que los parámetros de dirección puedan adaptarse fácilmente in situ.

Consulte también

Conexiones no especificadas (Página 612)

Principios básicos de la comunicación mediante la instrucción PUT/GET (Página 723)

Especificar una conexión no especificada (Página 640)

Crear una conexión: sinopsis (Página 616)

Configurar un proyecto de maquinaria de serie

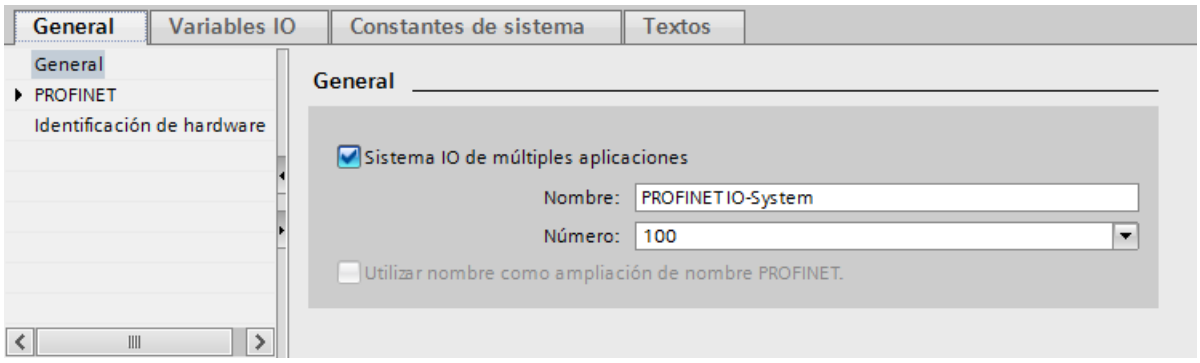
Requisitos

- CPU S7-1500 a partir de la versión de firmware V1.5 como controlador IO

Procedimiento

Para crear un proyecto de maquinaria de serie, proceda del siguiente modo:

1. Cree un proyecto.
2. Configure una CPU S7-1500 con versión de firmware V1.5 o superior como controlador IO.
3. Configure los dispositivos IO necesarios y asigne los dispositivos IO al controlador IO.
4. Configure la interconexión de puertos entre los dispositivos.
5. Seleccione el sistema IO de manera que pueda editar las propiedades en la ventana de inspección.
6. Active la casilla de verificación "Sistema IO de múltiples aplicaciones" en el apartado "General" de la ventana de inspección.



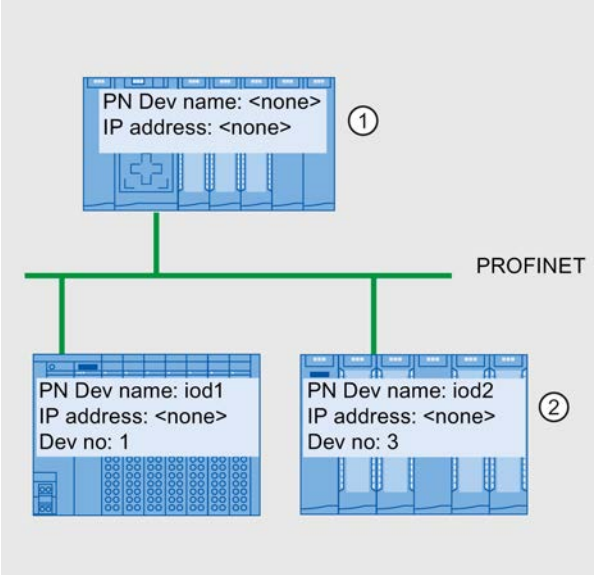
Resultado: STEP 7 establece los siguientes ajustes:

- El nombre de dispositivo del controlador IO (CPU) del proyecto de maquinaria de serie está fijado en "Permitir adaptar el nombre de dispositivo PROFINET directamente en el dispositivo". El controlador IO (CPU) carece al principio de nombre de dispositivo PROFINET.
- El protocolo IP del controlador IO (CPU) está ajustado de manera permanente a "Ajustar dirección IP en el dispositivo". La CPU carece al principio de dirección IP.
- La opción "Permitir sustitución de dispositivo sin medio de almacenamiento extraíble" se activa automáticamente. Esta opción hace posible una puesta en servicio automática. En tal caso, no es necesario que un técnico de puesta en marcha asigne localmente nombres de dispositivo a los dispositivos IO. El controlador IO asigna nombre de dispositivo y dirección IP a los dispositivos IO durante el arranque en función de la topología teórica y otros ajustes.
- El nombre de de dispositivo de los dispositivos IO está fijado en "Generar automáticamente el nombre del dispositivo PROFINET" (a partir del nombre configurado del dispositivo IO configurado).
- El protocolo IP de los dispositivos IO está fijado en "Ajustar dirección IP en el proyecto". Los dispositivos IO carecen al principio de dirección IP. Si un dispositivo IO no es un sistema de periferia descentralizada típico (p. ej. un sistema ET 200), sino otro dispositivo, como p. ej. un dispositivo HMI, cambie la opción a "Ajustar dirección IP en el dispositivo"; véase abajo.
- El número de dispositivo para los dispositivos IO se asigna automáticamente y se utiliza localmente para hacer que la dirección IP sea unívoca.

Para que el controlador IO pueda modificar más adelante el nombre de dispositivo, debe activarse la opción "Permitir sobrescribir el nombre del dispositivo PROFINET" (parámetros de CPU, propiedad de la interfaz PROFINET, apartado Direcciones Ethernet).

Esta opción está desactivada por defecto.

La figura siguiente muestra los ajustes de dirección IP y nombre de dispositivo PROFINET descritos anteriormente.



① Después de cargar la configuración del proyecto de maquinaria de serie, el controlador IO no tiene nombre de dispositivo ni dirección IP.

② Después de la carga, los dispositivos IO tienen un nombre y un número de dispositivo, pero no tienen dirección IP.

Cómo recibe un dispositivo IO una dirección IP localmente

A continuación se explican las opciones "Ajustar dirección IP en el proyecto" y "Ajustar dirección IP en el dispositivo", que pueden configurarse por lo general en un sistema IO de múltiples aplicaciones.

Si se ha ajustado la opción "Sistema IO de múltiples aplicaciones" en el sistema IO, STEP 7 activará automáticamente la opción "Ajustar dirección IP en el proyecto" en los dispositivos IO.

En ese caso, el dispositivo IO recibe del controlador IO una dirección IP derivada de la dirección IP del controlador IO ajustada localmente (consulte el siguiente capítulo). Esta opción resulta útil si el dispositivo IO es un dispositivo de campo, p. ej. ET 200MP, ET 200SP u otro sistema de periferia descentralizada.

Si el dispositivo IO no es un dispositivo de campo "normal" sino, p. ej., un dispositivo HMI con sistema operativo Windows, la opción "Ajustar dirección IP en el proyecto" descrita anteriormente no funciona. En tal caso, seleccione la opción "Ajustar dirección IP en el dispositivo". La dirección IP debe asignarse localmente en el dispositivo y el propio usuario debe asegurarse de que esta dirección concuerde con las direcciones IP de los demás dispositivos IO y con la dirección IP del controlador IO.

Consulte también

Adaptar maquinaria de serie localmente (Página 1603)

Adaptar maquinaria de serie localmente

Para adaptar la máquina cargada con el proyecto de maquinaria de serie, son necesarios unos pocos pasos de adaptación.

Localmente solo deben adaptarse el nombre de dispositivo y la dirección IP del controlador IO. Los nombres de dispositivo y las direcciones IP de los dispositivos IO se derivan de estas adaptaciones. En este ejemplo se describen las repercusiones de los ajustes locales para dos módulos de máquina concretos.

Los ajustes locales pueden realizarse, p. ej., con el display de la CPU, herramientas de puesta en servicio como Primary Setup Tool (PST) o PRONETA. Para estos ajustes no se necesita programadora con STEP 7, aunque también es posible realizarlos con ella.

Requisitos

- La máquina se ha cargado con un proyecto de maquinaria de serie (consulte Configurar un proyecto de maquinaria de serie (Página 1600)).
- El display está operativo o la herramienta deseada para asignar la dirección IP y el nombre de dispositivo está presente (p. ej. herramienta PST, STEP 7).
- Los puertos del controlador IO y los dispositivos IO están interconectados de acuerdo con la configuración.

Procedimiento

Tenga en cuenta las condiciones marco y las instrucciones para poner en marcha una S7-1500. Encontrará una referencia al manual de sistema S7-1500, en el que se describe la puesta en servicio, en "Consulte también".

Para adaptar una maquinaria de serie localmente, proceda del siguiente modo:

1. Integre la máquina en la red.
2. Conecte el dispositivo a la CPU para asignar la dirección IP y el nombre de dispositivo.
3. Asigne al controlador IO el nombre de dispositivo y la dirección IP deseados.

Durante el arranque, el controlador IO (CPU) asigna a los dispositivos IO el nombre de dispositivo PROFINET adaptado y una dirección IP unívoca.

Para la asignación se aplican las siguientes reglas:

- Los nombres de los dispositivos IO se derivan de la concatenación de los siguientes componentes de nombre, separados por un punto:
  <Nombre del dispositivo IO configurado en el proyecto de maquinaria de serie>.<Nombre del controlador IO respectivo ajustado en el dispositivo>
- Las direcciones IP de los dispositivos IO se derivan de la dirección IP del controlador IO correspondiente, configurada localmente, y del número de dispositivo (suma).

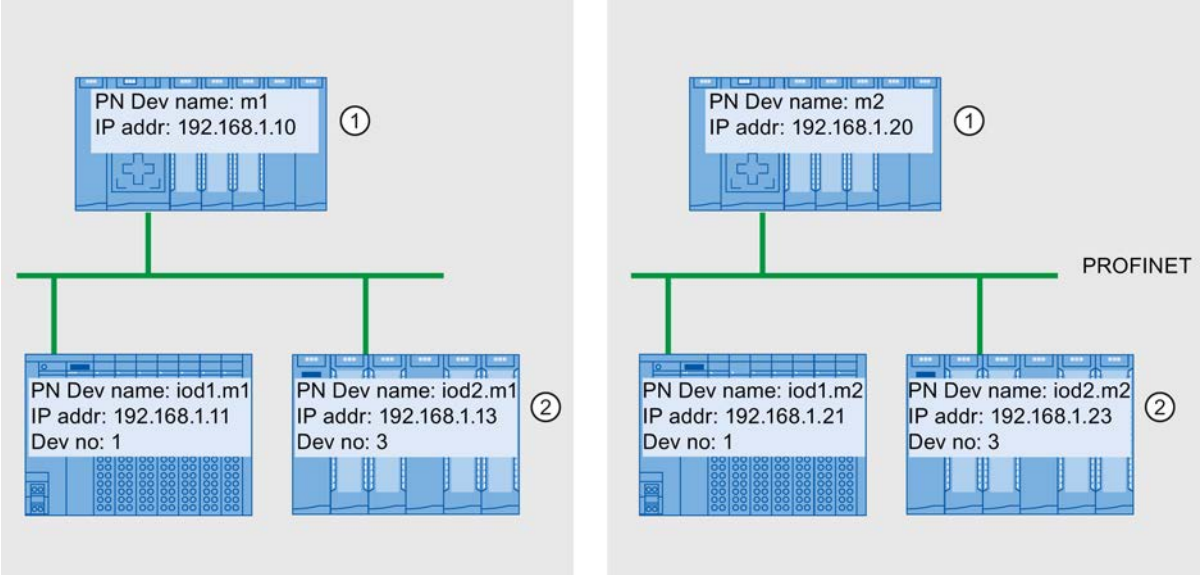
Nota

Tenga en cuenta que durante la asignación no pueden crearse en la subred direcciones IP duplicadas. En tal caso, el controlador IO no asigna una nueva dirección IP.

En la siguiente figura se ha asignado el nombre de dispositivo "m1" y la dirección IP 192.168.1.10 al controlador IO de la primera máquina.

Se ha asignado el nombre de dispositivo "m2" y la dirección IP 192.168.1.20 a la segunda máquina.

Los nombres de dispositivo y direcciones IP obtenidos de este modo pueden consultarse en la figura.



① Ajuste del nombre de dispositivo y la dirección IP en el controlador IO

② Tras el arranque, los dispositivos IO tienen un nombre de dispositivo actualizado (<Nombre de dispositivo configurado>.<Nombre de dispositivo del controlador IO>) y una dirección IP adaptada (= <Dirección IP del controlador IO> + <Número de dispositivo>)

Consulte también

Configurar un proyecto de maquinaria de serie (Página 1600)

Manual de sistema S7-1500 (http://support.automation.siemens.com/WW/view/es/59191792)

Permitir sobrescribir el nombre del dispositivo PROFINET

Con las CPU S7-1500 a partir de la versión de firmware V1.5 existe la posibilidad de sobrescribir nombres de dispositivo PROFINET de dispositivos IO durante el arranque de la CPU. Esta opción reduce el esfuerzo necesario a la hora de cambiar un dispositivo, por ejemplo en la puesta en marcha automática.

Efecto de la opción "Permitir sobrescribir el nombre del dispositivo PROFINET"

Si está activada la opción "Permitir sobrescribir el nombre del dispositivo PROFINET", el controlador IO (CPU) puede sobrescribir los nombres de dispositivo PROFINET de los dispositivos IO en el sistema IO.

Los sistemas IO de múltiples aplicaciones solo pueden operarse si esta opción está activada. Antes de sobrescribir, el controlador IO comprueba si el tipo de dispositivo IO coincide con el tipo configurado.

⚠ ADVERTENCIA

Si está activada la opción "Permitir sobrescribir el nombre del dispositivo PROFINET", los dispositivos conectados incorrectamente pueden recibir un nombre de dispositivo PROFINET incorrecto procedente de la configuración.

Esto puede provocar un funcionamiento incorrecto, con peligro de muerte, lesiones graves o daños materiales, según el tipo de periferia conectada.

Para evitar este tipo de riesgos, al sustituir un dispositivo compruebe siempre que esté conectado el dispositivo sustituto adecuado y que la interconexión de puertos se corresponda con la topología prevista.

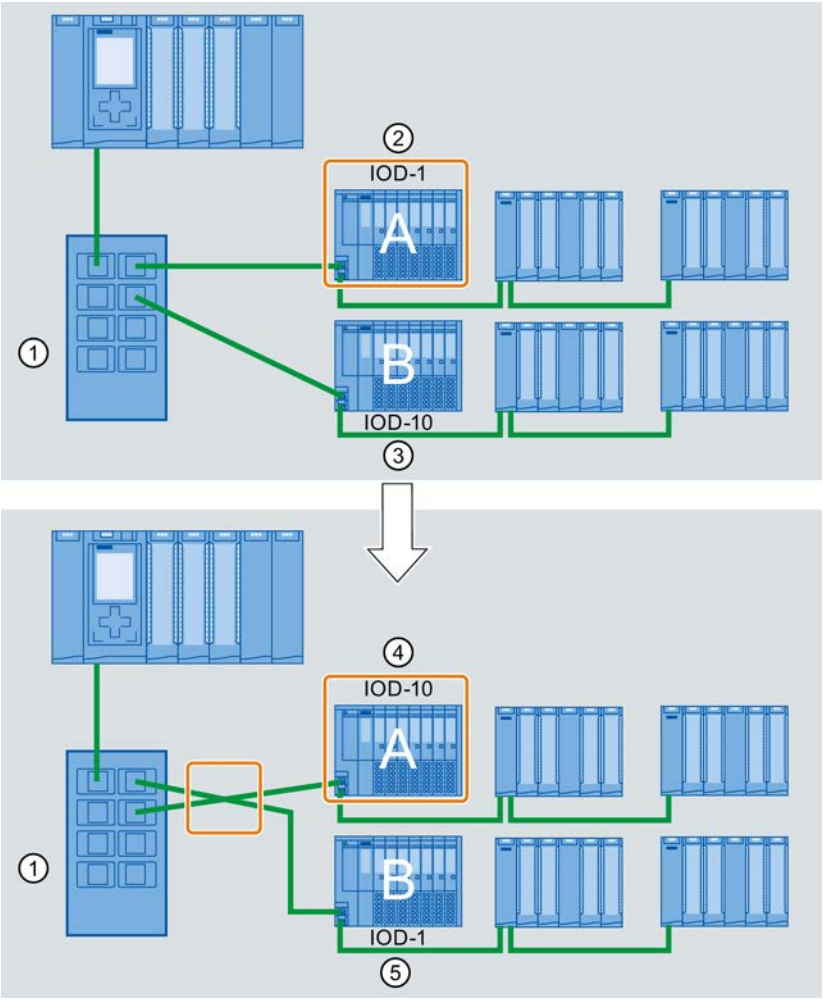
Si la opción no está activada, el controlador IO no puede sobrescribir los nombres de los dispositivos IO. En este caso, tras cambiar un nombre de dispositivo PROFINET en la configuración, o en caso de cambiar un dispositivo, p. ej., deberá asignar manualmente el nombre de dispositivo PROFINET al dispositivo IO o borrar el nombre de dispositivo de los dispositivos IO antes de una puesta en marcha automática.

Típica fuente de peligro

En principio, al sustituir un dispositivo IO ("supuesto normal"), se asegura hasta cierto punto que la conexión del dispositivo sustituido se realice de acuerdo con la interconexión de puertos configurada.

La siguiente figura muestra un supuesto en el que se confunden las conexiones de dos líneas PROFINET de idéntica estructura en dos switchports. Dado que el controlador IO asigna los nombres de dispositivo en función de la topología prevista, la conexión incorrecta de los dispositivos puede tener graves efectos en la asignación de nombres.

En este caso, al controlarse actuadores de distinta naturaleza, pueden producirse situaciones de peligro en la instalación.



① Switch con líneas PROFINET conectadas
② Dispositivo A, nombre de dispositivo "IOD-1": controla el motor 1
③ Dispositivo B, nombre de dispositivo "IOD-10": controla el motor 10
④ El dispositivo A controla el motor 10.
⑤ El dispositivo B controla el motor 1.

Procedimiento

Para modificar la opción "Permitir sobrescribir el nombre de dispositivo PROFINET", haga lo siguiente:

1. Seleccione en la vista de redes o en la vista de dispositivos la interfaz PROFINET de la CPU para la que desee cambiar esta opción.
2. Seleccione el área "Opciones avanzadas", apartado "Opciones de interfaz".
3. Modifique la opción.

Consulte también

Activar la sustitución de dispositivo sin medio de almacenamiento extraíble (Página 1513)

8.1.4.13 Acoplamiento de bus con acoplador PN/PN

Campo de aplicación y función

Campo de aplicación

El acoplador PN/PN sirve para interconectar dos subredes Ethernet e intercambiar datos entre ellas. Pueden emplearse datos de usuario mediante áreas de direcciones de entrada o salida o bien registros. El tamaño máximo de los datos de entrada y salida transferibles es de 1024 bytes. La repartición entre datos de entrada y datos de salida es aleatoria, de modo que pueden configurarse, p. ej. 800 bytes de datos de entrada y 200 bytes de datos de salida.

El acoplador PN/PN es un dispositivo con dos interfaces PROFINET, cada una de las cuales se conecta a una subred.

En la configuración, este acoplador PN/PN único se convierte en dos dispositivos IO, uno para cada estación con su subred. La otra parte del acoplador PN/PN, en cada caso, se denomina interlocutor acoplado. Cuando se finaliza la configuración, ambas partes se juntan.

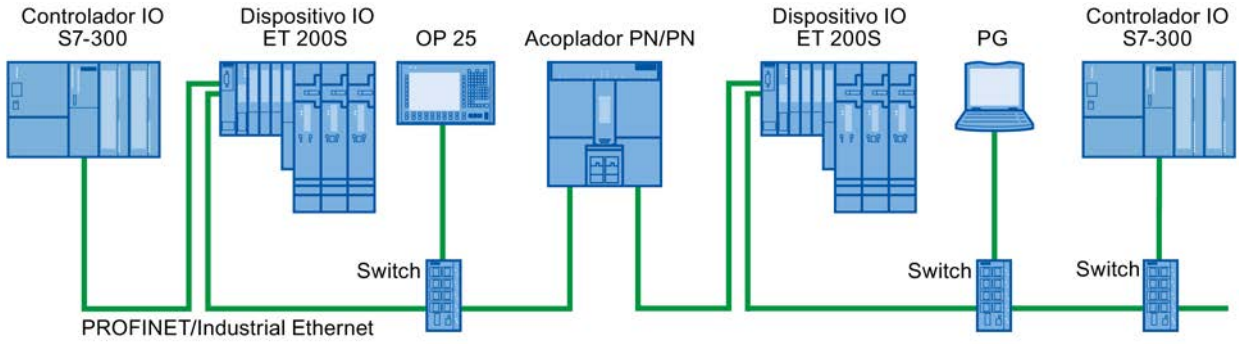


Figura 8-51 Acoplamiento de dos subredes PROFINET IO con un acoplador PN/PN

Información adicional

Encontrará más información relacionada con el tema "Acoplador PN/PN" en el Service & Support de Internet (http://support.automation.siemens.com/WW/view/es/44319532).

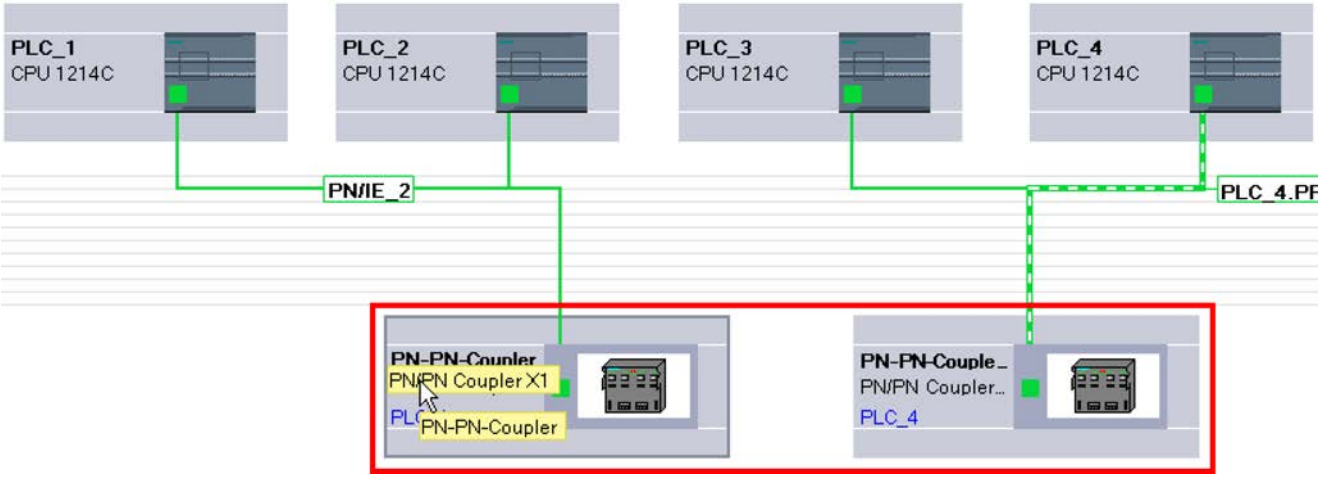
Acoplar subredes Ethernet

Acoplar subredes Ethernet con un acoplador PN/PN

El dispositivo normalizado acoplador PN/PN permite acoplar subredes Ethernet entre sí.

Para acoplar subredes Ethernet, proceda del siguiente modo:

1. Cree las subredes Ethernet.
2. Seleccione los aparatos de campo normalizados en el catálogo de hardware. En la carpeta "PROFINET IO" encontrará el acoplador PN/PN como módulo de cabecera.
3. Arrastre los componentes X1 y X2 de la versión deseada del acoplador PN/PN hasta la vista de redes mediante Drag & Drop. Los componentes forman un dispositivo, pero se muestran por separado para manejarlos mejor.
4. Conecte la interfaz Ethernet del acoplador PN/PN X1 con la primera subred Ethernet.
5. Conecte la interfaz Ethernet del acoplador PN/PN X2 con la segunda subred Ethernet. Ahora, las subredes Ethernet están acopladas mediante los dos componentes del acoplador PN/PN.



8.1.4.14 Integración de herramientas externas

Integración de herramientas externas a S7

Introducción

Para configurar dispositivos descentralizados es posible utilizar herramientas externas a STEP 7 ("Device Tools") con una interfaz de llamada especial (Tool Calling Interface). Dichos dispositivos también se denominan "aptos para TCI".

El volumen de suministro de estas herramientas excede las posibilidades de la configuración GSD, p. ej. ofrecen posibilidades avanzadas de entradas gráficas.

Se consideran los siguientes dispositivos descentralizados:

- Esclavos PROFIBUS DP
- Módulos dentro de un esclavo DP
- Dispositivos PROFINET IO
- Módulos dentro de un dispositivo IO

Nota

Garantía y responsabilidad

Siemens no asume responsabilidad alguna por el software de terceros (Device Tools) al que se acceda mediante TCI (Tool Calling Interface) ni por la correcta interacción con los correspondientes dispositivos.

Requisitos

La interfaz de llamada de la herramienta satisface la especificación TCI. Esta interfaz de llamada permite transferir parámetros y comandos al dispositivo descentralizado.

Estas herramientas deben instalarse utilizando los programas de instalación suministrados por el fabricante. La única excepción la constituye la Device Tool "S7-PCT" (Port Configuration Tool) para módulos maestros IO-Link y dispositivos IO-Link, que se suministra con STEP 7. Particularidad: después de la instalación, la herramienta no se visualiza en la lista de software instalado ni en la lista de productos de software del proyecto.

El archivo GSD del dispositivo descentralizado que va a configurarse con la Device Tool tiene que estar instalado.

Inicio de la Device Tool

El comando para iniciar Device Tool se encuentra en el menú contextual del dispositivo apto para TCI de la vista de dispositivos gráfica y en forma de tabla: "Iniciar Device Tool".

Consulte también

Inicio de SIMATIC S7-PCT (Página 1609)

Inicio de SIMATIC S7-PCT

Introducción

Con STEP 7 se instala la Device Tool "S7-PCT" (Port Configuration Tool).

Dicha herramienta permite parametrizar puertos de módulos IO-Link como 4SI IO-Link (S7-1200, ET 200S) o 4IOL+8DI+4DO (ET 200eco PN).

Requisitos

Ha configurado la correspondiente CPU, el esclavo DP o el dispositivo IO con un módulo IO-Link.

Procedimiento

Para iniciar la herramienta a través de la vista gráfica de dispositivos, proceda del siguiente modo:

1. Seleccione el módulo IO-Link en la vista de dispositivos.
2. Seleccione el comando "Iniciar Device Tool" del menú contextual.

O BIEN, para iniciar desde la vista de dispositivos en forma de tabla, proceda del siguiente modo:

1. Seleccione el módulo IO-Link en la vista de dispositivos.
2. Disponga las áreas en el área de trabajo de modo que se vea la vista de dispositivos en forma de tabla (se encuentra entre la vista de dispositivos y la ventana de inspección).
3. Seleccione la fila con el módulo IO-Link en la vista de dispositivos.
4. Seleccione el comando "Iniciar Device Tool" del menú contextual.

Resultado

La herramienta se inicia y permite configurar los puertos.

Consulte también

Integración de herramientas externas a S7 (Página 1608)

8.1.4.15 Cargar la configuración

Introducción a la carga de una configuración

Para poner en marcha un dispositivo es necesario que en la PG/el PC se hayan guardado idénticas configuraciones que en los dispositivos conectados. La sincronización entre PG/PC y los dispositivos conectados se realiza cargando una configuración. Los datos de configuración pueden cargarse en dos sentidos:

- de la PG/el PC al dispositivo
- del dispositivo a una PG/un PC

Consulte también

Cargar datos de proyecto de un dispositivo (Página 369)

Información general acerca de la carga (Página 364)

Carga de una configuración en un dispositivo (Página 1611)

Cargar los datos de proyecto en un dispositivo (Página 367)

Generalidades sobre la carga en la PG/el PC (Página 1612)

Particularidades del arranque (Página 1626)

Carga de una configuración en un dispositivo

Carga de la configuración de hardware

Si se ha insertado y configurado un dispositivo nuevo en el proyecto o si se ha modificado una configuración de hardware existente, primero hay que cargar la configuración actual en el dispositivo. De este modo se garantiza que tanto la PG/el PC como el dispositivo físico tengan la misma configuración. Utilice para ello el comando de menú "Online > Cargar en dispositivo (Página 367)".

En la primera carga se carga por completo la configuración hardware. En los procesos de carga posteriores se cargan únicamente los cambios en la configuración hardware.

Para cargar la configuración de hardware existen las posibilidades siguientes:

- Cargar en la vista de dispositivos o redes
- Cargar en el árbol del proyecto
- Cargar en un nodo accesible

> ⚠ ADVERTENCIA
>
> **Realizar el proceso de carga solo en STOP**
>
> En caso de una parametrización errónea, existe la posibilidad de que se produzca un comportamiento imprevisto de la máquina o del proceso después de cargar. Una CPU debe pasar obligatoriamente al estado operativo STOP para el proceso de carga con el fin de impedir que se dañe la instalación o se lesionen personas.

Particularidades de la carga de aplicaciones isócronas

Las aplicaciones isócronas tienen una parte de configuración hardware y otra de software.

Ejemplo: Si en la configuración hardware se cambia el número de un sistema IO, el tiempo de retardo o la asignación de una memoria imagen parcial de proceso de la periferia isócrona, se influye en los parámetros del OB de alarma de sincronismo y, por tanto, en la parte de software.

En aplicaciones isócronas debe cargarse siempre el proyecto completo (hardware y software). En caso de carga parcial (hardware y software independientemente en momentos distintos) pueden generarse incoherencias que, p. ej., obstaculicen el arranque de la CPU o provoquen que la aplicación no funcione en modo isócrono.

Consulte también

Información general acerca de la carga (Página 364)

Carga de una configuración en la PG/el PC

Generalidades sobre la carga en la PG/el PC

Introducción

Si va a una instalación con su programadora o PC y no está disponible el proyecto STEP 7 con el que se creó su configuración, cargue la configuración en su programadora o PC, p. ej. en un proyecto nuevo. Utilice para ello el comando de menú "Online > Cargar el dispositivo como nueva estación (hardware y software) (Página 369)".

La carga de un dispositivo en su programadora se efectúa básicamente a través de la lista de dispositivos accesibles en el árbol del proyecto. Es posible cargar varios dispositivos a la vez en el proyecto mediante selección múltiple.

Requisitos

- La configuración hardware del dispositivo debe haberse creado en el TIA Portal V12 o superior. Si la configuración hardware existente en el dispositivo se creó con una versión anterior, no podrá cargarse y deberá actualizarse (Página 343).
- Los módulos de GSD (ML), HSP o Service Packs del dispositivo deben estar instalados en la PG/el PC en el TIA Portal.
- Debe haber un proyecto abierto. Puede tratarse de un proyecto nuevo (vacío) o de un proyecto existente.
- El proyecto abierto se encuentra en modo offline.

Volumen de carga

La lista siguiente muestra un resumen de los componentes cargables de una configuración:

- El dispositivo (p. ej. una CPU) con todos los módulos de periferia y todos los ajustes de parámetros
- Sistemas maestros PROFIBUS y todos los ajustes relevantes para PROFIBUS
- Sistemas PROFINET IO y todos los ajustes relevantes para PROFINET
- I-devices y esclavos I
- Ajustes para la comunicación directa

Después de cargar una CPU se cargan automáticamente los demás módulos incluidos en el área de direcciones de la CPU.

Las siguientes conexiones se incluyen en la carga de la configuración:

- Conexiones S7 (también enrutadas) en redes PB/IE mixtas, también a través de interfaces IE-CP o PB-CM. Las conexiones S7 se aplican automáticamente como unilaterales cuando se carga la configuración del dispositivo, aunque la conexión S7 se hubiera configurado bilateralmente en el proyecto original. Si ambos interlocutores están cargados, la conexión se agrupa otra vez en la siguiente compilación.
- Conexiones TCP a través de una interfaz Ethernet propia de la CPU, conexiones UDP/ISOonTCP, conexiones TCP, UDP, ISO e ISOonTCP a través de una interfaz IE-CP
- Conexiones a través de la parametrización de la conexión OUC para proyectos STEP 7 V13 o superior

Nota

La configuración hardware cargada en la programadora o el PC no es idéntica a la configuración cargada originalmente en el dispositivo. Tenga en cuenta al respecto la información adicional sobre la carga, especialmente en lo relativo a los datos de configuración cargados parcialmente en caso de comunicación entre dispositivos.

No es posible la carga de sistemas de PC como, por ejemplo, WinAC o PC-based Automation.

Consulte también

Información general acerca de la carga (Página 364)

Carga de determinadas configuraciones de dispositivo

Información sobre la carga

Al cargar en la programadora o PC se transfieren al proyecto todos los parámetros configurados en el dispositivo. Si la CPU está conectada a una subred, se cargan todos los parámetros del dispositivo y la CPU se muestra como conectada en red en la vista de redes.

Nota

Las CPU en las que se ha restablecido la configuración de fábrica carecen de configuración hardware, por lo que en tal caso no se carga nada en el menú "Online" después de "Cargar en PG/PC".

Carga de configuraciones S7-300/400

Para evitar conflictos durante la carga de un dispositivo en un proyecto existente, deben cumplirse las siguientes reglas:

- Los nombres de dispositivo de CPU, esclavos PROFIBUS (esclavos DP, esclavos I) y dispositivos PROFINET (dispositivos IO, I-devices) deben ser unívocos.
- Combinación de nombre de red, ID de subred y dirección IP/DP unívoca para todos los módulos.

Si se produce un conflicto, se cancela la carga y se emite un aviso que informa sobre los problemas que hayan surgido. En tal caso se puede adaptar el proyecto como corresponda o bien reinstalar componentes que falten y volver a realizar la carga.

Las configuraciones de avisos no se cargan en la PG/el PC.

Carga de configuraciones S7-1200/1500

Tenga en cuenta lo siguiente al realizar la carga:

- No es posible cargar CPU S7-1200 con versión de firmware V1.0.
- No es posible cargar alarmas de diagnóstico del dispositivo del S7-1200. Para volver a generar las alarmas de diagnóstico del dispositivo del S7-1200 en la programadora o el PC es necesario compilar de nuevo la configuración hardware.
- Los comentarios de módulos de S7-1200/1500 se cargan en la PG/el PC desde el dispositivo si está ajustado el idioma del proyecto que también se utilizó al realizar la carga en el dispositivo. Si se desea, se puede deseleccionar la carga de los comentarios.

Nota

Todos los tipos de sistemas PC, como WinAC, Embedded Controller, CP 1616 o CP en PC, no permiten cargar en la programadora o PC.

Carga de periferia descentralizada

Se cargan las siguientes funcionalidades y ajustes de la periferia descentralizada:

- Sistemas maestros DP/sistemas IO con los correspondientes maestros DP/controladores IO (CPU y CP), esclavos DP/dispositivos IO, los módulos utilizados y sus parametrizaciones y propiedades, como p. ej. configuración futura, byte de estado o SYNC/FREEZE
- Conexión de memorias imagen parciales de proceso (IPP) a bloques de organización (OB). Esto se aplica a las propiedades de módulos y OBs
- Alarmas de proceso configuradas con las propiedades correspondientes
- Sistemas maestros DP con esclavo I
- CP como esclavo I PROFIBUS o I-Device PROFINET
- Comunicación directa

Las relaciones maestro-esclavo entre esclavo I/I-Device y maestro DP/controlador IO asignado solo se establecen en el proyecto si tanto el maestro como el esclavo se han cargado en la PG. Es indiferente si se carga primero el maestro DP/controlador IO o el esclavo I/I-Device. En cuanto ambos dispositivos están cargados, se restablecen también las relaciones maestro-esclavo/controlador-dispositivo.

Carga de subredes y dispositivos con MPI, PROFIBUS, Ethernet y PtP

Para la carga de subredes y puntos finales de conexiones para MPI, PROFIBUS, Ethernet y PtP con sus propiedades de conexión respectivas, se aplican las siguientes particularidades:

- Si se carga un dispositivo con interfaz PROFIBUS, los parámetros de bus del dispositivo difieren al principio de los ajustes del proyecto original. Solo cuando se han cargado todos los dispositivos implicados y no hay más dispositivos en el mismo bus, los parámetros de bus coinciden con el proyecto original.
- Los dispositivos de comunicación pasivos que no estén conectados con un sistema maestro o IO correspondiente como esclavos DP o dispositivos IO, no participan en el intercambio de datos. En consecuencia, no se cargan.
- En caso de configuraciones entre dispositivos deben cargarse todos los dispositivos afectados. Al compilar el proyecto aparece una advertencia para los dispositivos de red que faltan. Igualmente, la información de enrutado que falte por no haber cargado los dispositivos de comunicación también se notificará en forma de advertencia durante la compilación. Si la configuración vuelve a cargarse en el dispositivo desde la PG/el PC, se obtiene información de enrutado diferente.

Si se compila el proyecto después de cargar dispositivos en la programadora o el PC, STEP 7 comprobará si existen todos los dispositivos para los que se han configurado relaciones de comunicación. Si faltan dispositivos, aparece un aviso indicando el número de dispositivos de comunicación que faltan.

ATENCIÓN
Comunicación entre dispositivos Si se carga una configuración en la programadora o el PC con comunicación entre dispositivos también deben cargarse todos los dispositivos de red correspondientes. Si faltan dispositivos de red necesarios y la configuración se carga de nuevo en el dispositivo, no queda garantizado que siga funcionando la comunicación entre dispositivos.

Consulte también

Actualizar proyectos (Página 343)

Carga de configuraciones con servidor web

Información sobre la carga

La configuración hardware de una CPU incluye también los ajustes del servidor web. La carga de una configuración de servidor web en la programadora o PC está sujeta a algunas restricciones:

- La asignación de idioma del servidor web e idioma del proyecto en el S7-300/400 no se carga. Los textos del proyecto no se cargan y se emite un aviso indicando que no se han asignado idiomas del proyecto. En las CPU S7-1200/1500 los idiomas asignados en STEP 7 se cargan sin restricciones.
- Los datos de la administración de usuarios de S7-1200/1500 pueden cargarse pero no editarse. Una casilla de verificación permite seleccionar si se desea utilizar los datos existentes protegidos contra escritura o bien rechazarlos e introducir datos nuevos.
- No se cargan las tablas de observación del servidor web.

No se cargan los archivos fuente de las páginas web definidas por el usuario (páginas HTML, Java Scripts, etc.). Los bloques de programa generados durante la carga solo pueden editarse si el propio usuario introduce las propiedades y la página HTML.

Consulte también

Información importante sobre el servidor web (Página 976)

Información importante sobre el servidor web (Página 936)

Carga de configuraciones con PROFIBUS

Información sobre la carga

Un maestro DP se carga en la PG/el PC. El sistema maestro DP y todos los esclavos DP conectados se integran en el proyecto. Se mantienen los respectivos ajustes. Si se ha creado ya una subred PROFIBUS adecuada, los dispositivos cargados con interfaz PROFIBUS se conectan a la subred existente.

Para la carga de sistemas maestros DP con esclavos normalizados es imprescindible que los archivos GSD correspondientes estén instalados en el TIA Portal y estén disponibles en el catálogo de hardware. Si un archivo GSD necesario no está disponible en la misma versión que en el dispositivo, se muestran las diferencias en la comprobación de coherencia.

Nota

La comunicación directa en una configuración solo se carga si todos los interlocutores que intervienen en la comunicación directa se cargan en la PG/el PC. Los interlocutores no disponibles se sustituyen por un dispositivo comodín. De este modo sigue siendo posible compilar la configuración con intercambio de datos directo.

Modo isócrono

Al cargar sistemas maestros DP con la funcionalidad PROFIBUS "Modo isócrono" activada, tenga en cuenta lo siguiente:

- Tras cargar del dispositivo a la programadora o el PC, los parámetros de bus y los ajustes de modo isócrono no son idénticos hasta que no se han cargado todos los dispositivos relevantes para el cálculo del modo isócrono.
- Solo se soportan sistemas monomaestro con modo isócrono. Por lo tanto, solo se cargan configuraciones con un solo maestro DP en la subred PROFIBUS.

Carga de configuraciones con PROFINET

Información sobre la carga

Si se ha marcado una CPU en la lista de dispositivos accesibles y se efectúa una carga en la PG/el PC, se cargan todos los controladores IO y dispositivos IO pertenecientes a ese dispositivo con sus sistemas IO. Los ajustes de topología se aplican igualmente. Si existe ya una red Ethernet adecuada en el proyecto, los dispositivos cargados se integran en la red existente.

Las relaciones entre controladores IO y dispositivos IO solo se representan en el proyecto si tanto el controlador IO como el dispositivo I se han cargado en la PG. Es indiferente si se carga primero el controlador IO o los I-devices.

Funciones soportadas

Se cargan las siguientes funcionalidades y ajustes:

- Configuraciones PROFINET (RT e IRT) en sistemas IO con los controladores IO (CPU y CP) y dispositivos IO correspondientes y los módulos utilizados
- Direcciones lógicas y propiedades de interfaces
- Interconexiones de puertos
- Modo isócrono
- Dominios Sync/MRP
- Función de redundancia "Cliente" o "Administrador" en configuraciones MRP

Nota

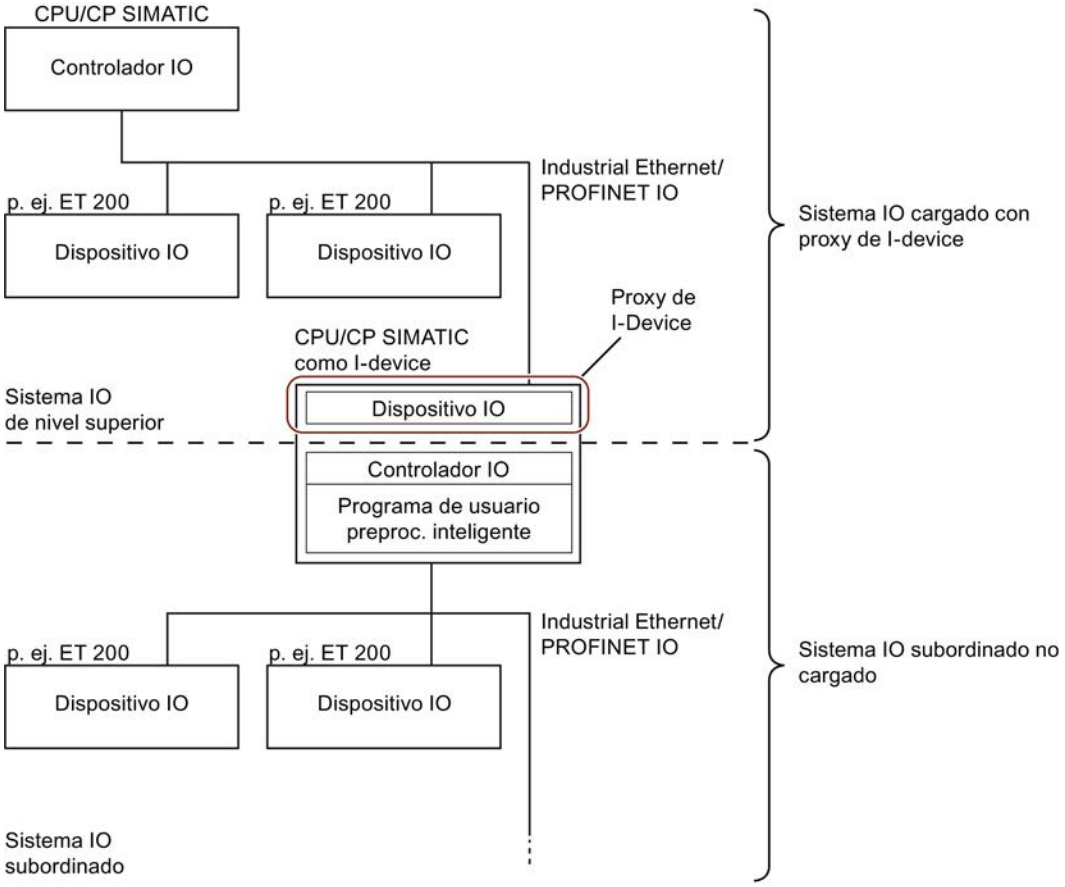
Los dominios Sync y MRP vacíos no se incluyen en la carga.

Dispositivos IO basados en GSD

Para la carga de dispositivos IO basados en GSD es imprescindible que los archivos GSD correspondientes estén instalados en el TIA Portal y estén disponibles en el catálogo de hardware. Si un archivo GSD necesario no está disponible en la misma versión que en el dispositivo, se muestran las diferencias en la comprobación de coherencia.

I-Device

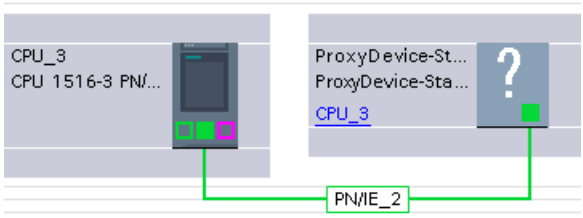
Si se carga el controlador IO desde un sistema IO con I-device enlazado, se cargan el controlador IO y sus dispositivos IO. Un I-device implicado solo se carga como proxy del I-device. El proxy del I-device carece de parametrización de CPU, incluida la parametrización del "propio" sistema IO (subordinado). Solamente se carga la interfaz con el controlador IO de nivel superior.



Para los proxy de I-Device se aplica lo siguiente:

- El controlador IO puede compilarse y cargarse.
- Las propiedades se muestran pero no se pueden modificar
- No se realiza el diagnóstico en el árbol del proyecto

En la vista de redes, los dispositivos comodín se representan con un signo de interrogación:



Para poder editar el I-Device en el proyecto, debe cargarse el I-Device del dispositivo en la PG/el PC. De este modo se sustituye el proxy del I-Device por el I-Device completo con sus dispositivos IO subordinados.

Es posible cargar un controlador IO con proxy de I-Device vinculado desde la programadora o el PC al dispositivo.

Nota

La sustitución de I-devices comodín solo es posible si el I-device necesario está disponible en el catálogo de hardware.

Un proxy de I-Device que represente a un I-Device de SIMOTION no puede cargarse ni sustituirse.

Configuraciones con IE/PB-Link

Si existe una de las siguientes configuraciones con IE/PB-Link PN IO como dispositivo PROFINET IO, se carga toda la configuración con todos los dispositivos PROFIBUS subordinados:

- CPU/CP de S7-300/400
- Estación PC y sistema maestro PROFIBUS conectado

La configuración completa se compone de:

- CPU
- Configuración de CP
- Sistema PROFINET IO con IE/PB-Link conectado
- Sistema maestro PROFIBUS del IE/PB-Link con esclavos DP conectados

Un ejemplo de configuración se compone de una CPU S7-300 con un CP como controlador PROFINET IO. En el controlador IO está conectado un IE/PB-Link como dispositivo IO. El IE/PB-Link, en calidad de maestro PROFIBUS DP, sondea un esclavo PROFIBUS DP, p. ej. ET 200L. Si se carga la CPU desde el dispositivo en la PG/el PC, se carga toda la configuración.

Nota

Si el IE/PB-Link no se utiliza como proxy de PROFINET IO, sino como transición de red en modo estándar, dicho IE/PB-Link funciona como módulo central y puede cargarse por separado.

Carga de Shared Devices

Para la carga de comentarios, se aplica lo siguiente: Si un módulo de entradas y salidas con la función Shared Input interna del módulo (MSI) o Shared Output interna del módulo (MSO) solo consta de un submódulo, dicho submódulo no posee comentario propio. En su lugar, se utiliza el comentario del módulo de entradas y salidas. Solo en caso de división del módulo en varios submódulos tienen campos de comentarios propios tanto el módulo de entradas y salidas como todos los submódulos.

Carga de dispositivos HMI

Información sobre la carga

Al cargar dispositivos HMI en la programadora o PC se distingue entre los siguientes casos:

- Los dispositivos HMI conectados con un maestro DP como esclavo DP o con un controlador IO como dispositivo IO se cargan como esclavo DP o dispositivo IO, respectivamente (por ejemplo, PP 17-I PROFIsafe).
- Los dispositivos HMI de un sistema maestro como esclavo I o de un sistema IO como I-device se cargan como proxy de esclavo I o proxy de I-Device (Página 1617) (por ejemplo, SIMATIC Comfort Panels). El acceso a los ajustes del proxy del dispositivo es de solo lectura.
- Los dispositivos HMI no se cargan si están conectados a una subred (PROFIBUS o PROFINET) pero no a un sistema maestro o sistema IO (por ejemplo, KP600 Basic color DP).

Conexión online de configuraciones cargadas

Se puede pasar a online con el proyecto entero cargado o bien solo con partes del mismo.

Requisitos

La configuración hardware cargada del dispositivo a la PG/el PC se ha compilado. Solo después de una compilación se muestran correctamente los estados de los módulos centralizados y descentralizados.

Nota

Si se realiza la conexión online antes de la compilación, se muestra el símbolo de diagnóstico "?" (diagnóstico imposible). En la ventana de inspección se muestra un aviso correspondiente en "Información > General".

Dependencias

En función de lo completa que haya sido la carga de la configuración hardware en la PG, existen restricciones para pasar a online y para el diagnóstico:

- Dispositivo cargado completamente con todos los módulos centralizados y descentralizados respectivos, como esclavos DP o dispositivos IO:
  Son posibles la conexión online y el diagnóstico.
- Dispositivo cargado con I-Devices/esclavos I conectados:
  - El I-Device/esclavo I no está cargado: es posible la conexión online para el dispositivo y sus módulos. Para los componentes dependientes no cargados de la configuración se utilizan proxys de dispositivos con un mínimo soporte de diagnóstico. El estado online se representa en forma de icono. En la vista Online y diagnóstico se muestra el diagnóstico estándar. Los datos I&M no se cargan.
  - El I-Device/esclavo I también está cargado: la conexión online es posible para todos los dispositivos, el diagnóstico se soporta por completo.

8.1.5 Información adicional sobre configuraciones

8.1.5.1 Funcionamiento de las CPUs S7-1200

Estados operativos

Principios básicos de los estados operativos de CPUs S7

Introducción

Los estados operativos describen el comportamiento de la CPU. Existen los estados operativos siguientes:

- ARRANQUE
- RUN
- STOP

En estos estados operativos, la CPU es apta para la comunicación, p. ej. a través de la interfaz PN/IE.

Otros estados operativos

Si la CPU no está lista para funcionar, está en uno de los dos estados operativos siguientes:

- Sin tensión, es decir, la tensión de red está desconectada.
- Avería, es decir, se ha producido un fallo interno.
  Si el estado "Avería" es provocado por un error de firmware, se señalizará con los LEDs de estado de la CPU (véase la descripción de la CPU). Para determinar la causa, proceda del siguiente modo:
  - Desconecte y vuelva a conectar el interruptor de red.
  - Lea el búfer de diagnóstico cuando la CPU arranque y envíe los datos al Customer Support para que los analicen.

  Si la CPU no arranca, cámbiela.

Consulte también

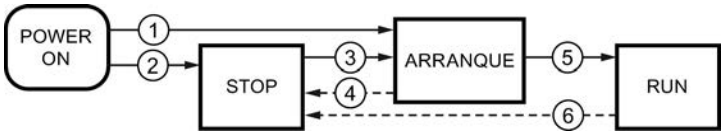
Estado operativo "STOP" (Página 1627)

Estado operativo "RUN" (Página 1627)

Transiciones de estado operativo

Resumen

La figura siguiente muestra los estados operativos y las transiciones de CPUs S7-1200:



La tabla siguiente muestra las condiciones en las que cambian los estados operativos:

N.º	Transición de estado operativo	Condiciones
①	POWER ON → ARRANQUE	Una vez conectada, la CPU pasa al estado operativo "ARRANQUE" si: • está ajustado el tipo de arranque "Arranque en caliente" y • la configuración de hardware y los bloques de programa son coherentes. La memoria no remanente se borra y en el contenido de los DBs no remanentes se restablecen los valores iniciales de la memoria de carga. La memoria remanente y el contenido de los DBs remanentes se conservan.
②	POWER ON → STOP	Si se ajusta el tipo de arranque "Sin arranque", la CPU pasa al estado operativo "STOP" después de conectar la tensión de alimentación. La memoria no remanente se borra y en el contenido de los DBs no remanentes se restablecen los valores iniciales de la memoria de carga. La memoria remanente y el contenido de los DBs remanentes se conservan.
③	STOP → ARRANQUE	La CPU pasa al estado operativo "ARRANQUE" si: • la CPU se pone a "RUN" desde la programadora y • la configuración de hardware y los bloques de programa son coherentes.

N.º	Transición de estado operativo	Condiciones
④	ARRANQUE → STOP	La CPU pasa al estado operativo "STOP" en los casos siguientes: • Durante el arranque se detecta un error. • La CPU se pone a "STOP" desde la programadora. • Se procesa un comando STOP en el OB de arranque.
⑤	ARRANQUE → RUN	Si el arranque es correcto, la CPU pasa a "RUN".
⑥	RUN → STOP	La CPU pasa al estado operativo "STOP" en los casos siguientes: • Se detecta un error que impide seguir trabajando. • La CPU se pone a "STOP" desde la programadora. • Se procesa un comando STOP en el programa de usuario.

Estado operativo "ARRANQUE"

Principios básicos del estado operativo "ARRANQUE"

Función

Antes de que la CPU empiece a procesar el programa de usuario cíclico después de la conexión, se procesa un programa de arranque.

En dicho programa se definen determinadas variables de inicialización para el programa cíclico programando de forma adecuada los OBs de arranque. El número de OBs de arranque no está prescrito. Es posible no programar ningún OB de arranque, programar sólo uno o bien varios.

Parametrización del comportamiento de arranque

Existe la posibilidad de determinar si la CPU debe permanecer en estado operativo STOP o bien si debe realizarse un arranque en caliente. Asimismo, en el grupo "Arranque" de las propiedades de la CPU se parametriza el comportamiento durante el arranque (RUN o estado operativo anterior).

Particularidades

Tenga en cuenta los puntos siguientes para el estado operativo "ARRANQUE":

- Los OBs de arranque se procesan. Independientemente del tipo de arranque seleccionado se procesan todos los OBs de arranque programados.
- No se pueden procesar programas controlados por tiempo.
- El procesamiento de programas controlados por alarmas se limita a:
  - OB 82 (alarma de diagnóstico)
- Las salidas de los módulos están bloqueadas.
- La memoria imagen de proceso no se actualiza, pero es posible un acceso directo de periferia a las entradas.

Consulte también

Editar las propiedades y los parámetros (Página 529)

Principios básicos de los estados operativos de CPUs S7 (Página 1621)

Bloques de organización para el arranque (Página 1675)

Rearranque (en caliente) (Página 1624)

Rearranque (en caliente)

Función

Con el rearranque (en caliente) se borran todas las marcas no remanentes y se restablecen los valores iniciales de la memoria de carga para todos los contenidos de DB no remanentes. Las marcas remanentes y contenidos remanentes de DBs se conservan.

El procesamiento del programa empieza con el primer OB de arranque.

Lanzar un rearranque (en caliente)

En los casos siguientes es posible realizar un "Rearranque (en caliente)" desde un comando de menú de la programadora:

- La CPU está en estado operativo "STOP".
- Después del borrado total
- Después de cargar un programa coherente y una configuración hardware coherente en estado operativo "STOP" de la CPU.

En caso de "Conexión (POWER ON)" se lanza un "Rearranque (en caliente)" si se ha parametrizado el comportamiento de arranque del siguiente modo:

- Tipo de arranque "Arranque en caliente - RUN" (independientemente del estado operativo de la CPU antes de la desconexión (POWER OFF)).
- "Arranque en caliente - modo de operación antes de desconexión (POWER OFF)" (en función del estado operativo de la CPU antes de la desconexión (POWER OFF). La CPU debía estar previamente en RUN.)

Consulte también

Áreas de memoria remanentes (Página 1632)

Actividades de arranque

Resumen

La tabla siguiente muestra qué actividades realiza la CPU durante el arranque:

Actividades en el orden de procesamiento	en caso de rearranque (en caliente)
Borrar marcas no remanentes	sí
Borrar todas las marcas	no
Borrar la memoria imagen de proceso de las salidas	sí
Procesamiento de los OBs de arranque	sí
Actualizar la memoria imagen de proceso de las entradas	sí
Habilitar las salidas tras transición de estado operativo a "RUN"	sí

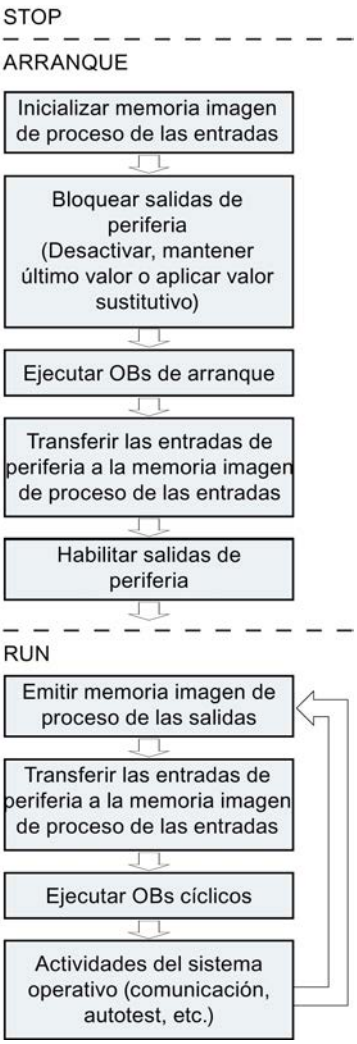
Ejecución

La figura siguiente muestra las actividades de la CPU en los estados operativos "STOP", "ARRANQUE" y "RUN".

El estado de las salidas de la periferia en el primer ciclo del programa de usuario se puede definir con las medidas siguientes:

- Utilizar módulos de salida parametrizables para aplicar valores sustitutivos o mantener el último valor.
- Preasignar las salidas en los OBs de arranque.

Durante el arranque, todos los eventos de alarma se colocan en una cola de espera y se procesan posteriormente en estado operativo RUN. En estado operativo RUN es posible procesar alarmas de proceso en cualquier momento.



Particularidades del arranque

Comportamiento si la configuración teórica difiere de la real

La configuración teórica se representa mediante la configuración cargada en la CPU. La configuración real es la que existe realmente en el sistema de automatización.

Si la configuración teórica difiere de la real, en principio la CPU pasa igualmente a RUN.

Cancelación de un arranque

Si durante el arranque se producen errores, éste se cancela y la CPU permanece en "STOP".

Un arranque no se realiza o se cancela en las condiciones siguientes:

- Si hay insertada una tarjeta SD no válida.
- Si no hay ninguna configuración hardware cargada.

Consulte también

Resumen de las propiedades de la CPU (Página 1646)

Estado operativo "RUN"

Función

En estado operativo "RUN" se procesa el programa cíclico, controlado por tiempo y por alarmas:

- Se emite la memoria imagen de proceso de las salidas.
- Se lee la memoria imagen de proceso de las entradas.
- Se procesa el programa de usuario.

El intercambio activo de datos entre CPUs S7-1200 a través de la Open User Communication sólo es posible en estado operativo "RUN".

Procesamiento del programa de usuario

Una vez que la CPU ha leído las entradas, el programa cíclico se procesa a partir de la primera instrucción y hasta la última instrucción.

Si se ha parametrizado un tiempo mínimo de ciclo, la CPU termina el ciclo una vez transcurrido el tiempo mínimo de ciclo aunque la ejecución del programa de usuario haya requerido menos tiempo.

Para garantizar que el programa cíclico se ejecute en un tiempo determinado, se ajusta un tiempo de vigilancia del ciclo que se puede adaptar en función de las necesidades. Si la ejecución del programa cíclico no ha terminado en ese tiempo, el sistema reacciona con un error de tiempo.

Otros eventos, p. ej. las alarmas de proceso o las alarmas de diagnóstico, pueden interrumpir el flujo del programa y prolongar el tiempo de ciclo.

Consulte también

Principios básicos de los estados operativos de CPUs S7 (Página 1621)

Eventos y OBs (Página 1636)

Estado operativo "STOP"

Función

En estado operativo "STOP" no se procesa el programa de usuario. Todas las salidas se desactivan o reaccionan del modo parametrizado: devuelven el valor sustitutivo parametrizado o mantienen el último valor emitido, con lo que el proceso controlado pasa a un estado operativo seguro.

La CPU comprueba los puntos siguientes:

- Hardware, p. ej. si todos los módulos están disponibles
- Si la CPU se rige por los valores predeterminados o si hay juegos de parámetros
- Si se cumplen las condiciones marco para el comportamiento de arranque programado

Consulte también

Principios básicos de los estados operativos de CPUs S7 (Página 1621)

Principios básicos del borrado total

Función

El borrado total de la CPU sólo es posible en estado operativo STOP.

En el borrado total, la CPU pasa a una especie de "estado original". Esto significa que:

- Si existe una conexión online entre la programadora o el PC y la CPU, se deshará.
- Tanto el contenido de la memoria de trabajo como los datos remanentes y no remanentes se borran.
- El búfer de diagnóstico, la hora, la dirección IP, la configuración hardware y las peticiones actuales de forzado permanente se conservan.
- Seguidamente, la memoria de carga (bloques lógicos y de datos) se copia en la memoria de trabajo. Esto hace que los bloques de datos ya no tengan valores actuales, sino sus valores iniciales configurados.

Áreas de memoria

Información importante sobre las Memory Cards

Función de la Memory Card

La SIMATIC Memory Card para una S7-1200 es una de las tarjetas de memoria SD preformateadas de Siemens para el programa de usuario de la CPU.

Sólo se pueden borrar archivos y carpetas. Si se formatea la Memory Card con Windows, p. ej. con un lector de tarjetas habitual, la Memory Card quedará inutilizada como medio de almacenamiento para una CPU S7.

Configurar el tipo de tarjeta

La Memory Card se puede utilizar como tarjeta de transferencia, tarjeta de programa o tarjeta de actualización de firmware.

Para configurar el tipo de tarjeta, inserte la Memory Card en el lector de tarjetas de la programadora y seleccione la carpeta "Lector de tarjetas/memoria USB" del árbol del proyecto. En las propiedades de la Memory Card marcada se puede establecer el tipo de tarjeta:

- Programa
  Si se utiliza como tarjeta de programa, es posible cargar el programa de usuario en la Memory Card. De esta manera, la memoria de carga interna de la CPU se sustituye por la Memory Card y se borra la memoria de carga interna. En este caso, el programa de usuario puede ejecutarse íntegramente desde la Memory Card. Si se extrae la Memory Card que contiene el programa de usuario, ya no existirá ningún programa en el dispositivo.
- Transferencia
  Si se utiliza como tarjeta de transferencia, es posible transferir el programa de usuario de la Memory Card a la memoria de carga interna de la CPU. Después se puede extraer la Memory Card.
- Firmware Card
  En una Memory Card se puede guardar firmware para módulos S7-1200. Por este motivo, es posible realizar una actualización de firmware mediante una Memory Card especialmente diseñada para ello. En la Memory Card también puede guardarse una copia de seguridad del firmware de un módulo.

Transferir objetos del proyecto a la Memory Card

Si la Memory Card está insertada en la programadora o en un lector de tarjetas externo, se pueden copiar en ella los siguientes objetos desde el árbol del proyecto:

- Bloques individuales (posibilidad de selección múltiple)
  En este caso se ofrece una transferencia coherente, es decir, se tiene en cuenta la interdependencia de los bloques por llamadas.
- PLC
  En este caso, todos los objetos relevantes para la ejecución, como los bloques y la configuración hardware, entre otros, se transfieren a la Memory Card, al igual que durante el proceso de carga.

La transferencia puede realizarse arrastrando y soltando los objetos, o con el comando "Lector de tarjetas/memoria USB > Escribir en Memory Card" del menú "Proyecto".

Transferir objetos de la Memory Card al proyecto

Los bloques individuales (posibilidad de selección múltiple) se transfieren al proyecto mediante la función Drag&Drop. Las configuraciones hardware no pueden transferirse de la Memory Card al proyecto.

Actualizar firmware con una Memory Card

Los archivos de firmware más recientes pueden adquirirse en Internet, a través de las páginas del "Service & Support":

http://support.automation.siemens.com (http://support.automation.siemens.com/WW/view/es/34143537)

Guarde los archivos de firmware en el disco duro e inserte la SIMATIC Memory Card en el lector de tarjetas de su programadora.

Para guardar el archivo en la Memory Card, seleccione en el árbol del proyecto la Memory Card en la carpeta "Lector de tarjetas/memoria USB". Seleccione el menú contextual "Lector de tarjetas/memoria USB > Crear actualización de firmware en Memory Card".

A continuación, siga las instrucciones del portal Service & Support para actualizar el firmware con la CPU.

Con la actualización del firmware, la CPU recibe una nueva versión del firmware. Si ha utilizado la CPU en el proyecto, la CPU ya configurada se debe actualizar mediante un cambio de dispositivo offline con la CPU que contiene la nueva versión del firmware, y después adaptar y cargar el programa y la configuración.

Consulte también

Sustituir un componente de hardware (Página 528)

Información importante sobre las versiones de firmware de la CPU y las versiones de STEP 7 (Página 1698)

Memoria de carga

Función

Cada CPU tiene una memoria de carga interna. El tamaño de dicha memoria depende de cada CPU.

La memoria de carga interna puede sustituirse externamente por Memory Cards. Si no hay ninguna Memory Card insertada, la CPU utiliza la memoria de carga interna; si hay una insertada, la CPU utiliza la Memory Card como memoria de carga.

Sin embargo, el tamaño de la memoria de carga externa utilizable no puede ser mayor que la memoria de carga interna, incluso si la tarjeta SD insertada dispone de más espacio libre.

Consulte también

Utilizar Memory Cards (Página 424)

Memoria de trabajo

Función

La memoria de trabajo es un área de memoria no remanente y almacena las partes del programa de usuario que son relevantes para la ejecución del programa. El programa de usuario se ejecuta exclusivamente en la memoria de trabajo y sistema.

Memoria de sistema

Áreas de memoria de sistema

Función

La memoria de sistema contiene los elementos de memoria que cada CPU pone a disposición del programa de usuario, p. ej. la memoria imagen de proceso de las entradas y salidas y las marcas.

Si se utilizan las operaciones adecuadas, los datos se direccionan en el programa directamente al área de operandos correspondiente.

La tabla siguiente muestra las áreas de operandos de la memoria de sistema:

Área de operandos	Descripción	Acceso vía unidades del tamaño siguiente:	Notación S7
Memoria imagen de proceso de las salidas	Al principio del ciclo, la CPU escribe los valores de la memoria imagen de proceso de las salidas en los módulos de salida.	Salida (bit)	Q
		Byte de salida	QB
		Palabra de salida	QW
		Palabra doble de salida	QD
Memoria imagen de proceso de las entradas	Durante el ciclo, la CPU lee las entradas de los módulos de entrada y guarda los valores en la memoria imagen de proceso de las entradas.	Entrada (bit)	I
		Byte de entrada	IB
		Palabra de entrada	IW
		Palabra doble de entrada	ID
Marca	Esta área pone a disposición espacio de memoria para resultados intermedios calculados en el programa.	Marca (bit)	M
		Byte de marcas	MB
		Palabra de marcas	MW
		Palabra doble de marcas	MD
Bloque de datos	Los bloques de datos guardan información del programa. Existe la posibilidad de definirlos de modo que todos los bloques lógicos accedan a ellos (DBs globales) o que estén asignados a un FB o SFB determinado (DB instancia).	Bit de datos	DBX
		Byte de datos	DBB
		Palabra de datos	DBW

Área de operandos	Descripción	Acceso vía unidades del tamaño siguiente:	Notación S7
	Requisitos: El atributo de bloque "Acceso optimizado al bloque" no está activado.	Palabra doble de datos	DBD
Datos locales	Esta área de memoria incluye los datos locales temporales de un bloque mientras dure el procesamiento del mismo. Requisitos: El atributo de bloque "Acceso optimizado al bloque" no está activado. Recomendación: Acceder a los datos locales (Temp) simbólicamente.	Bit de datos locales	L
		Byte de datos locales	LB
		Palabra de datos locales	LW
		Palabra doble de datos locales	LD
Área de periferia de las entradas	Las áreas de periferia de las entradas y salidas permiten acceder directamente a los módulos de entrada y salida descentralizados.	Bit de entrada de periferia	<variable>:P
		Byte de entrada de periferia	
		Palabra de entrada de periferia	
		Palabra doble de entrada de periferia	
Área de periferia de las salidas		Bit de salida de periferia	
		Byte de salida de periferia	
		Palabra de salida de periferia	
		Palabra doble de salida de periferia	

Consulte también

Búfer de diagnóstico (Página 1635)

Principios básicos de las memorias imagen de proceso (Página 1633)

Principios básicos de la programación de bloques de datos (Página 2937)

Declarar variables y constantes locales en la interfaz del bloque (Página 2648)

Estructura de la interfaz del bloque (Página 2642)

Acceso a las direcciones de periferia (Página 1636)

Áreas de memoria remanentes

Áreas de memoria remanentes

Para impedir pérdidas de datos en caso de un corte de alimentación, algunos datos pueden marcarse como remanentes. Entonces se guardarán en un área de memoria remanente. Un área de memoria remanente es un área cuyo contenido se conserva en caso de rearranque (en caliente), es decir, después de desconectar la tensión de alimentación y volverla a conectar en una transición de STOP a RUN.

Es posible definir como remanentes los datos siguientes:

- Marcas: el ancho exacto del área de memoria remanente se define para las marcas en la tabla de variables PLC o en el plano de ocupación.
- Variables de un bloque de función (FB): En la interfaz de un FB se pueden definir como remanentes variables seleccionadas si el acceso optimizado al bloque está activado para este bloque. Si el acceso optimizado al bloque no está activado para un FB, los ajustes de remanencia sólo se pueden efectuar en el bloque de datos de instancia asignado.
- Variables de un bloque de datos global: En un bloque de datos global, se pueden definir como remanentes algunas o todas las variables del bloque en función del ajuste del acceso:
  - Bloque con acceso optimizado: La remanencia se puede ajustar para cada variable individual.
  - Bloque con acceso estándar: el ajuste de remanencia es válido para todas las variables del DB; o todas las variables son remanentes o no lo es ninguna.

Consulte también

Ajustar la remanencia de variables PLC (Página 2578)

Rearranque (en caliente) (Página 1624)

Fundamentos del acceso a bloques (Página 2258)

Memoria imagen de proceso de las entradas y salidas

Principios básicos de las memorias imagen de proceso

Función

Si en el programa de usuario se activan las áreas de operandos de entradas (I) y salidas (Q), no se consultan o modifican los estados de señal en los módulos de señales digitales, sino un área de la memoria de sistema de la CPU. Dicha área de memoria se denomina memoria imagen de proceso.

Ventajas de la memoria imagen de proceso

El acceso a la memoria imagen de proceso tiene la ventaja, en comparación con el acceso directo a los módulos de entrada y salida, que la CPU dispone de una imagen coherente de las señales de proceso mientras dura el procesamiento cíclico del programa. Si durante el procesamiento del programa cambia un estado lógico en un módulo de entrada, dicho estado se conserva en la memoria imagen de proceso hasta que ésta se actualiza en el próximo ciclo. Esto permite consultar varias veces una señal de entrada en el programa y obtener siempre una información de entrada coherente.

Asimismo, el acceso a la memoria imagen de proceso requiere bastante menos tiempo que el acceso directo a los módulos de señales, porque la memoria imagen de proceso se encuentra en la memoria interna de la CPU.

Actualizar las memorias imagen de proceso

Ejecución

Las áreas de las memorias imagen de proceso son actualizadas cíclicamente por el sistema operativo, a menos que se haya definido lo contrario en la configuración. La actualización de las memorias imagen de proceso de las entradas y salidas se realiza en el orden siguiente:

1. Se ejecutan las tareas internas del sistema operativo.
2. La memoria imagen de proceso de todas las salidas (IPS) se escribe en las salidas de los módulos.
3. El estado de todas las entradas se lee en la memoria imagen de proceso de las entradas (IPE).
4. Se procesa el programa de usuario con todos los bloques que se han llamado en él.

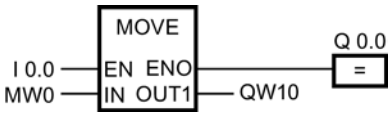
El sistema operativo controla independientemente la escritura de la memoria imagen de proceso de las salidas en las salidas de los módulos y la lectura de la memoria imagen de proceso de las entradas.

Particularidades

Existe la posibilidad de acceder directamente a las entradas y salidas a través de un acceso directo a la periferia.

- Si una instrucción accede directamente a una salida y la dirección de salida está dentro de la memoria imagen de proceso de las salidas, se corregirá la memoria imagen de proceso de la salida en cuestión.
- Si una instrucción accede directamente a una salida y la dirección de salida **no** está dentro de la memoria imagen de proceso de las salidas, **no** se corregirá la memoria imagen de proceso de la salida en cuestión.

Ejemplo de un acceso normal a la periferia a través de la memoria imagen de proceso



Actualizar QW10 en el área de periferia de las salidas con el valor MW0.

Error de acceso a la periferia al actualizar la memoria imagen de proceso

Si se produce un error al actualizar la memoria imagen de proceso (error de acceso a la periferia), la CPU reacciona con la reacción de sistema predeterminada "Ignorar".

Consulte también

Principios básicos del tratamiento de errores (Página 2932)

Dirección inicial de un módulo (Página 1635)

Acceso a las direcciones de periferia (Página 1636)

Actividades de arranque (Página 1625)

Búfer de diagnóstico

Función

El búfer de diagnóstico forma parte de la memoria de sistema de la CPU. Contiene los errores detectados por la CPU y los módulos aptos para diagnóstico. Aquí se incluyen los eventos siguientes, entre otros:

- Todos los cambios de estado operativo de la CPU (p. ej. conexión (POWER ON), transición al estado operativo STOP, transición al estado operativo RUN)
- Todas las alarmas de diagnóstico

Para la CPU S7-1200, el búfer de diagnóstico tiene una capacidad de 50 entradas, de las cuales se conservan las 10 últimas (las más recientes) en caso de una transición de desconexión (POWER OFF) - conexión (POWER ON).

Las entradas sólo se borran al restablecer la configuración de fábrica de la CPU.

El contenido del búfer de diagnóstico se lee desde la vista online y de diagnóstico.

Consulte también

Principios básicos del búfer de diagnóstico (Página 2204)

Área de datos de la periferia

Dirección inicial de un módulo

Definición

La dirección inicial es la dirección de byte más baja de un módulo. Representa la dirección inicial del área de datos de usuario del módulo.

Configurar las direcciones iniciales de los módulos

La asignación entre las direcciones utilizadas en el programa de usuario y los módulos se realiza mediante configuración de los módulos.

Las direcciones iniciales asignadas automáticamente al colocar los módulos se modifican en las propiedades del módulo (grupo "Direcciones E/S...").

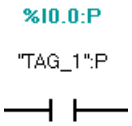
También existe la posibilidad de ajustar si las direcciones estarán en la memoria imagen de proceso o no.

Acceso a las direcciones de periferia

Direcciones de periferia

Cuando se inserta un módulo en la vista de dispositivos, sus datos de usuario se encuentran en la memoria imagen de proceso de la CPU S7-1200 (ajuste predeterminado). La CPU acepta automáticamente el intercambio de datos entre el módulo y el área de memoria imagen de proceso al actualizarse esta última.

Si el programa debe acceder al módulo directamente y no a través de la memoria imagen de proceso, la dirección de periferia se debe completar con ":P".



Esto sucede, p. ej., en un programa de tiempo crítico en el que las salidas deben influenciarse en el mismo ciclo.

Principios básicos del procesamiento del programa

Eventos y OBs

Eventos y OB

El sistema operativo de las CPU S7-1200 se basa en eventos. Hay que distinguir entre dos tipos de eventos:

- Eventos que pueden arrancar un OB
- Eventos que no pueden arrancar OB

Un evento que puede arrancar un OB provoca la siguiente reacción tras su aparición:

- Si se ha asignado un OB al evento, se llama dicho OB. Si por el momento no es posible llamar el OB, el evento se incorpora a una cola de espera de acuerdo con su prioridad.
- Si no se ha asignado ningún OB al evento, se ejecuta la reacción del sistema predeterminada.

Un evento que no puede iniciar ningún OB provoca la reacción del sistema predeterminada para la clase de evento correspondiente tras su aparición.

Por consiguiente, la ejecución del programa de usuario se basa en eventos, en la asignación de OB a los eventos y en el código que está dentro de los OB o que se llama desde allí.

La tabla siguiente ofrece una vista general de los eventos que pueden iniciar un OB, junto con las clases de eventos y OB correspondientes. Está ordenada por la prioridad predeterminada de los OB. La prioridad 1 es la más baja.

Clase de evento	N.º del OB	Cantidad de OB	Evento de arranque	Prioridad de OB (ajuste predeterminado)
Programa cíclico	1, >= 123	>= 1	Fin de arranque o fin del último OB de ciclo	1
Arranque	100, >= 123	>=0	Transición STOP-RUN	1
Alarma horaria	>= 10	Máx. 2	Se ha alcanzado la hora de arranque	2
Alarma de retardo	>= 20	Máx. 4	Tiempo de retardo finalizado	3
Alarma cíclica	>= 30		Intervalo de tiempo equidistante finalizado	8
Alarma de proceso	>= 40	Máx. 50 (pueden utilizarse más con DETACH y ATTACH)	• Flanco ascendente (máx. 16) • Flanco descendente (máx. 16)	18
			• HSC: valor de contaje=valor de referencia (máx. 6) • HSC: sentido de contaje modificado (máx. 6) • HSC: reset externo (máx. 6)	18
Alarma de estado	55	0 o 1	La CPU ha recibido una alarma de estado	4
Alarma de actualización	56	0 o 1	La CPU ha recibido una alarma de actualización	4
Alarma de fabricante o perfil	57	0 o 1	La CPU ha recibido una alarma de fabricante o perfil	4
Alarma de diagnóstico	82	0 o 1	El módulo ha detectado un error	5
Alarma de presencia de módulo	83	0 o 1	Extracción o inserción de módulos de la periferia descentralizada	6
Fallo de rack	86	0 o 1	Error en el sistema de entrada/salida de la periferia descentralizada	6
Error de tiempo	80	0 o 1	• Tiempo de vigilancia del ciclo excedido • El OB llamado todavía se está ejecutando • Alarma horaria perdida • Alarma horaria perdida en STOP • Desbordamiento de la cola de espera • Pérdida de alarmas por sobrecarga de alarmas	22

La tabla siguiente describe los eventos que no provocan el arranque de ningún OB y la reacción correspondiente del sistema operativo. Está ordenada por la prioridad de los eventos.

Clase de evento	Evento	Prioridad de evento	Reacción del sistema
Extracción/inserción de módulos centrales	Extracción/inserción de un módulo	21	STOP
Error de acceso a la periferia al actualizar la memoria imagen de proceso	Error de acceso a la periferia al actualizar la memoria imagen de proceso	22	Ignorar

Clase de evento	Evento	Prioridad de evento	Reacción del sistema
Error de programación	Error de programación en un bloque para el que se utilizan reacciones del sistema preparadas por el sistema operativo (nota: si se ha activado el tratamiento de errores local, será efectivo el tratamiento de errores programado en el bloque.)	23	RUN
Error de acceso a periferia	Error de acceso a la periferia en un bloque para el que se utilizan reacciones del sistema preparadas por el sistema operativo (nota: si se ha activado el tratamiento de errores local, será efectivo el tratamiento de errores programado en el bloque.)	24	RUN
Tiempo de vigilancia del ciclo excedido 2 veces	Tiempo de vigilancia del ciclo excedido 2 veces	27	STOP

Asignación entre OB y eventos

A excepción del programa cíclico y del programa de arranque, sólo es posible asignar un OB a un evento. En algunas clases de eventos, como las alarmas de proceso, es posible asignar el mismo OB a varios eventos.

La asignación entre OB y evento se realiza en la configuración del hardware. Una asignación ya realizada se modifica con las instrucciones ATTACH y DETACH en el tiempo de ejecución.

Prioridad de OB y comportamiento de ejecución

Las CPU S7-1200 soportan desde la prioridad 1 (más baja) hasta la 27 (más alta). Un OB obtiene la prioridad de su evento de arranque.

En principio, los OB se procesan por orden de prioridad: Los OB de mayor prioridad se procesan en primer lugar. Los eventos con la misma prioridad se procesan en el orden de aparición.

A partir de la versión de firmware V4.0 de las CPU S7-1200, en la configuración del dispositivo existe la posibilidad de especificar en las propiedades de la CPU si los OB pueden interrumpirse o no. Esta parametrización afecta a todos los OB excepto los OB de ciclo, que siempre pueden interrumpirse.

Para las CPU S7-1200 con versión de firmware < V4.0 rige lo siguiente:

- Todos los OB con prioridad >= 2 interrumpen el programa cíclico.
- Un OB de prioridad 2 a 25 no puede ser interrumpido por ningún evento de prioridad 2 a 25. Lo mismo es válido cuando se produce un evento con una prioridad mayor que la del OB activo en ese momento. Estos eventos se procesan en otro momento.
- La aparición de un error de tiempo (prioridad 26) interrumpe cualquier otro OB.

Para las CPU S7-1200 con versión de firmware V4.0 o superior rige lo siguiente:

Si se parametrizan los OB como que no pueden interrumpirse, se procesarán siempre hasta el final, incluso cuando se produzca un evento de prioridad superior durante su ejecución. En particular, esto significa lo siguiente:

- Todos los OB con prioridad >= 2 interrumpen el programa cíclico.
- Un OB de prioridad 2 a 25 no puede ser interrumpido por ningún evento. Lo mismo es válido cuando se produce un evento con una prioridad mayor que la del OB activo en ese momento, incluso para un error de tiempo. Estos eventos se procesan en otro momento.

Si se parametrizan los OB como que pueden interrumpirse y se produce un evento de prioridad superior durante la ejecución de un OB, dicho OB se interrumpirá y se procesará el OB correspondiente al evento que se ha producido. Una vez finalizado este último, se seguirá procesando el OB interrumpido. En particular, esto significa lo siguiente:

- Todos los OB con prioridad >= 2 interrumpen el programa cíclico.
- Un OB de prioridad 2 a 25 puede ser interrumpido por cualquier evento cuya prioridad sea superior a la del OB en ejecución. Lo mismo es válido cuando se produce un error de tiempo: la aparición de un error de tiempo (prioridad 26) interrumpe cualquier OB.

Información de arranque del OB

Algunos OBs tienen información de arranque, otros no. Esto se explica con más detalle en la descripción del OB correspondiente.

Consulte también

Procesamiento del programa basado en eventos (Página 1639)

Principios del tratamiento de errores local (Página 2934)

Bloques de organización (OB) (Página 2252)

Principios básicos del tratamiento de errores (Página 2932)

ATTACH: Asignar OB a evento de alarma (Página 4884)

DETACH: Deshacer asignación entre OB y evento de alarma (Página 4885)

Procesamiento del programa basado en eventos

Prioridad de OB y comportamiento de ejecución

Las CPU S7-1200 soportan desde la prioridad 1 (más baja) hasta la 27 (más alta). Un OB obtiene la prioridad de su evento de arranque.

Un OB de alarma sólo puede verse interrumpido por un OB de error de tiempo. Lo mismo es válido cuando se produce un evento con una prioridad mayor que la del OB activo en ese momento. Así pues, a excepción del OB de error de tiempo, sólo puede haber activo un OB de alarma.

Si durante el procesamiento de un OB de alarma se produce otro evento, dicho evento se incorporará a una cola de espera de acuerdo con su prioridad. Los eventos de arranque de una cola de espera se procesan posteriormente en el orden de aparición.

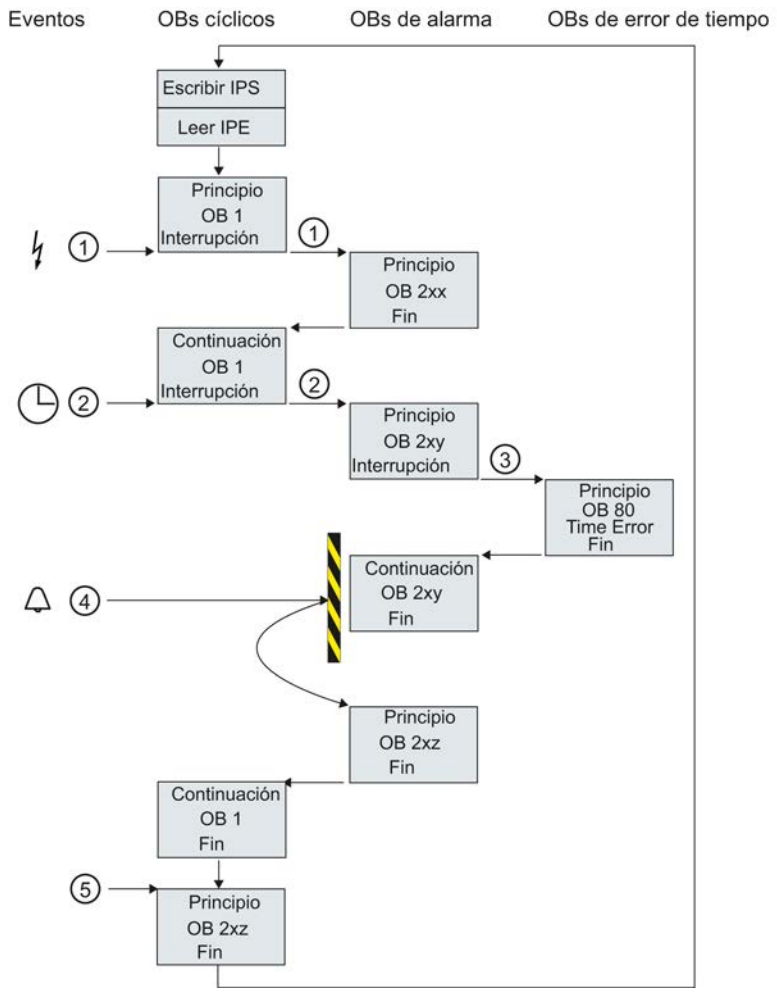
Procesamiento del programa en la CPU

Los OB cíclicos son interrumpidos por OB de alarma.

Reacciones a eventos que inician un OB de alarma:

- En las CPU hasta la versión de firmware V3: los OB de alarma solo pueden ser interrumpidos por OB de error de tiempo.
- En las CPU a partir de la versión de firmware V4: los OB de alarma pueden ser interrumpidos por OB de alarma de prioridad superior.

La figura siguiente muestra la secuencia básica en caso de que los OB de alarma no puedan interrumpirse (comportamiento hasta la versión de firmware V3):


Eventos
OBs cíclicos
OBs de alarma
OBs de error de tiempo
Escribir IPS
Leer IPE
Principio OB 1 Interrupción
Principio OB 2xx Fin
Continuación OB 1 Interrupción
Principio OB 2xy Interrupción
Principio OB 80 Time Error Fin
Continuación OB 2xy Fin
Principio OB 2xz Fin
Continuación OB 1 Fin
Principio OB 2xz Fin
1
2
3
4
5

① y ② Un evento (p. ej. una alarma de proceso) llama un OB asignado.

Un OB llamado con todos los bloques subordinados se procesa sin interrupción. Puesto que no hay ningún otro evento pendiente que provoque el arranque de un OB, después de procesar las alarmas se continúa con el procesamiento del OB cíclico.

③ Un OB de alarma sólo puede verse interrumpido por un OB de error de tiempo (OB 80).

④ Durante el procesamiento de la alarma se produce otro evento que lanza otra alarma.

Reacción en las CPU hasta la versión de firmware V3 inclusive:

Este evento nuevo se incorpora a una cola de espera. Sólo cuando ha concluido el procesamiento del OB de alarma actual, los eventos de la cola de espera llaman sucesivamente a los OB asignados en cada caso, y lo hacen siguiendo las reglas siguientes:

- Los eventos se procesan de acuerdo con su prioridad (empezando por la más alta).
- Los eventos con la misma prioridad se procesan en orden cronológico.

Reacción en las CPU a partir de la versión de firmware V4:

En las CPU con versión de firmware V4 o superior, la posibilidad de interrupción se ajusta con un parámetro de la CPU. Comportamiento predeterminado: los OB pueden interrumpirse. En este caso rige lo siguiente: si el nuevo evento tiene una prioridad mayor que el OB en ejecución, el OB que se ha iniciado con el nuevo evento interrumpe el OB en ejecución.

Si desactiva la opción, los OB de alarma no se interrumpirán.

⑤ Los OB cíclicos se procesan uno detrás de otro.

Indicaciones relativas a la cola de espera

- Para cada clase de prioridad (OB llamados con la misma prioridad) existe una cola de espera propia. Cada una de esta colas de espera tiene un tamaño predefinido.
- Si una cola de espera está llena y aparece otro evento, el mismo se rechazará y se perderá. Al mismo tiempo se generará un "Evento de error de tiempo". En la información de arranque del OB de error de tiempo (OB 80) se incluye la información relacionada con el OB que causa el error. Es posible programar una reacción adecuada en el OB de error de tiempo, p. ej. generar un aviso.

Ejemplo de un evento de alarma de proceso

El ejemplo de un módulo que origina una alarma de proceso describe el funcionamiento del procesamiento del programa orientado a eventos en la CPU S7-1200.

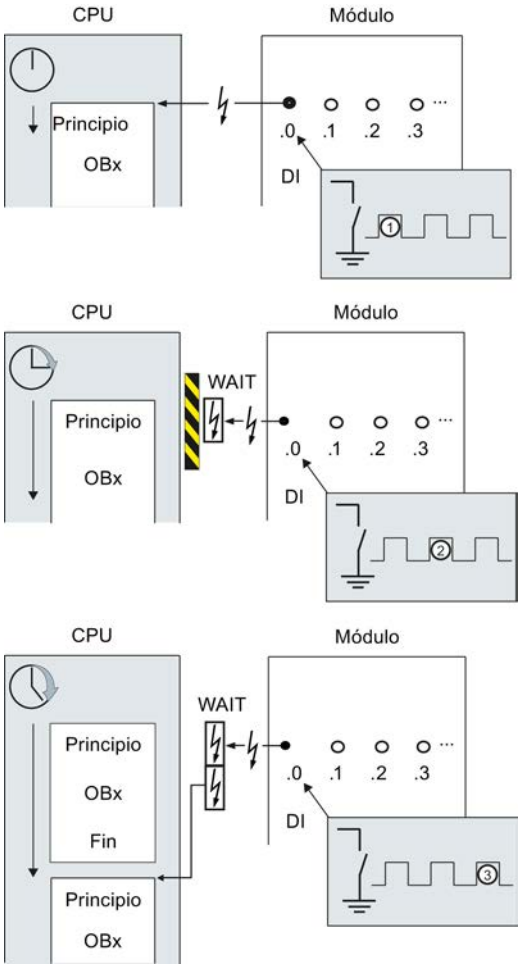
Eventos de proceso y su prioridad

Los eventos de proceso son eventos que lanza la periferia (p. ej. una entrada digital) y que provocan la llamada de un OB asignado en la CPU S7-1200. Por ello, los OBs asignados a un evento de proceso se denominan OBs de alarma de proceso.

Ejemplos de eventos de proceso y su prioridad:

- Evento de proceso "flanco ascendente" o "flanco descendente" de un módulo que lanza una alarma: el OB de alarma de proceso que se inicia seguidamente tiene siempre la prioridad 5.
- Eventos de proceso del contador rápido
  - El valor de contaje equivale al valor de referencia
  - Modificar sentido de contaje
  - Reset externo del contador rápido
    El OB de alarma de proceso que se inicia seguidamente tiene siempre la prioridad 6.

La figura siguiente muestra la evolución temporal del procesamiento de una alarma de proceso: En caso de eventos de alarma de proceso inmediatamente consecutivos, el segundo evento que dispara una alarma de proceso se retendrá en la cola de espera hasta que esté procesado el primer OBx. Este OBx deberá procesarse primero para que el siguiente evento que dispara una alarma de proceso pueda arrancar el correspondiente OBx. Los demás eventos que disparen alarmas de proceso se añadirán asimismo a la cola de espera siguiendo este principio.



Procesamiento de una alarma de proceso

① Un evento que lanza una alarma de proceso, p. ej. un flanco ascendente en la entrada, llama a un OB asignado a dicho evento.

② Si se produce otro evento que lanza una alarma de proceso mientras se procesa el OB, dicho evento se incorporará a una cola de espera.

③ El nuevo evento que lanza una alarma de proceso inicia el OB de alarma de proceso asignado a dicho evento.

Asignar el evento que lanza la alarma

El evento que lanza la alarma se asigna a un OB en la vista de dispositivos, propiedades de las entradas.

- Un evento que lanza una alarma sólo puede asignarse a un OB.
- Un OB puede estar asignado a varios eventos que lanzan alarmas.
  Así, p. ej., es posible asignar tanto flancos ascendentes como descendentes al mismo OB de alarma, con el fin de reaccionar directamente y del mismo modo a cada cambio de la señal de entrada.
- El OB iniciado puede interrumpir un OB de ciclo en cada instrucción. La coherencia en los accesos a datos se garantiza hasta el tamaño de palabra doble.
- Según el módulo es posible parametrizar diferentes eventos que lanzan alarmas, p. ej. un flanco ascendente y uno descendente en la entrada.
- La asignación entre eventos que lanzan alarmas y el OB iniciado se define al configurar el módulo que lanza alarmas. Sin embargo, la instrucción DETACH permite deshacer dicha asignación en el OB de alarma de proceso iniciado y la instrucción ATTACH, asignar otro OB al mismo evento. Esto permite reaccionar con flexibilidad a señales de proceso externas.

Ajustar el comportamiento operativo

Modificar las propiedades de los módulos

Configuración estándar

Todos los componentes de hardware parametrizables están ajustados de fábrica a los valores adecuados para las aplicaciones estándar. Estos valores estándar permiten utilizar los componentes sin tener que realizar más ajustes.

No obstante, el comportamiento y las propiedades de los componentes de hardware pueden parametrizarse y, con ello, adaptarse a las necesidades y particularidades de la instalación. Los componentes de hardware parametrizables son, p. ej., CPUs, módulos de comunicación y algunos módulos de señales analógicas y digitales.

Ajustar y cargar parámetros

Si se ha seleccionado un componente de hardware en la vista de dispositivos o redes, las propiedades se ajustan en la ventana de inspección. Al guardar la configuración de un dispositivo junto con sus parámetros, se generan datos que deben cargarse en la CPU. Desde allí, dichos datos se transfieren a los módulos correspondientes durante el arranque.

Propiedades de las CPUs

Para el comportamiento del sistema son muy importantes las propiedades de las CPUs. Una CPU permite ajustar, p. ej.:

- Interfaces
- Entradas y salidas
- Contadores rápidos
- Generadores de impulsos
- Comportamiento de arranque
- Hora
- Nivel de protección
- Marcas de sistema y ciclo
- Tiempo de ciclo
- Carga de comunicación

Las posibilidades de entrada indican qué puede ajustarse y con qué rangos de valores. Los campos no editables están atenuados o no aparecen en la ventana de propiedades.

Requisitos

Los componentes de hardware cuyas propiedades deben modificarse se han dispuesto en un rack.

Procedimiento

Para modificar las propiedades y los parámetros de los componentes de hardware, proceda del siguiente modo:

1. En la vista de dispositivos o redes seleccione el componente de hardware o la interfaz que desee editar.
2. Edite los ajustes del objeto seleccionado:
   - En la vista general de dispositivos se editan las direcciones y los nombres, p. ej.
   - En la ventana de inspección existen otras posibilidades de ajuste adicionales.

No es necesario confirmar las entradas, los valores modificados se aplican inmediatamente.

Consulte también

Editar las propiedades y los parámetros (Página 529)
Introducción a la carga de una configuración (Página 1610)

Propiedades de la CPU

Resumen de las propiedades de la CPU

Resumen

La tabla siguiente muestra una vista general de las propiedades de la CPU:

Grupo	Propiedades	Descripción
General	Información del proyecto	Información general que describe la CPU insertada. Es posible modificar todos los datos excepto el número de slot.
	Información del catálogo	Únicamente información legible del catálogo de hardware para esta CPU.
Interfaz PROFINET	General	Nombre y comentario de la interfaz PROFINET. El nombre está limitado a 110 caracteres.
	Direcciones Ethernet	Selección de si la interfaz PROFINET está conectada en red. Si ya se han creado subredes en el proyecto, estarán disponibles para su selección en la lista desplegable. En caso contrario, es posible crear una subred nueva con el botón "Agregar nueva subred". El protocolo IP incluye datos sobre la dirección IP, la máscara de subred y el uso de un router IP en la subred. Si se utiliza un router IP, es imprescindible indicar la dirección IP del mismo.
	Opciones avanzadas	Nombre, comentario y otras posibilidades de ajuste del puerto de la interfaz Ethernet.
	Sincronización horaria	Ajustes de la sincronización horaria en formato de hora NTP. NTP (network time protocol) es un procedimiento general para sincronizar relojes de sistema en redes locales y globales. Con el procedimiento NTP, la interfaz de la CPU envía regularmente consultas de hora (en modo cliente) al servidor NTP de la subred (LAN) cuyas direcciones deben parametrizarse aquí. Sobre la base de las respuestas del servidor, se determina la hora más exacta y fiable y se sincroniza. La ventaja de este procedimiento es la posibilidad de sincronizar la hora más allá de los límites de la subred. La precisión depende de la calidad del servidor NTP utilizado.
DI#/DO#	General	Nombre y comentario de las entradas digitales integradas en la CPU.

Grupo	Propiedades	Descripción
	Entradas digitales	Las entradas digitales permiten ajustar retardos a la entrada. Dichos retardos se ajustan por grupos (4 entradas cada uno). En cada entrada digital puede activarse la detección de un flanco ascendente y uno descendente. A este evento se le puede asignar un nombre y una alarma de proceso. En función de la CPU es posible activar tomas de impulso en distintas entradas. Si la toma de impulso está activada, los impulsos que sean más cortos que el tiempo de ciclo del programa también detectarán los flancos.
	Salidas digitales	Todas las salidas digitales permiten ajustar la reacción en caso de cambio de estado operativo de RUN a STOP: El estado se congela (equivale a mantener el último valor) o bien se ajusta un valor sustitutivo ("0" o "1").
	Direcciones E/S	Se definen el área de las direcciones de entrada y salida así como la memoria imagen de proceso.
	Identificador de hardware	Se muestra el identificador de hardware del dispositivo.
AI#	General	Nombre y comentario de las entradas analógicas integradas en la CPU.
	Entradas analógicas	En la reducción de ruido, las frecuencias perturbadoras de la frecuencia indicada (en Hz) se suprimen por medio del tiempo de integración ajustado. Dirección de canal, tipo de medición, rango de tensión, alisamiento y diagnóstico de rebase se ajustan en el grupo "Canal #". El tipo de medición está ajustado fijamente a tensión, entre 0 y 10 V. Gracias al alisamiento de los valores analógicos, se prepara una señal analógica estable para el procesamiento posterior. El alisamiento de los valores analógicos es aconsejable en caso de cambios lentos en los valores medidos, p. ej. en las mediciones de temperatura. Los valores medidos se alisan mediante filtrado digital. Para el filtrado, el módulo calcula valores medios a partir de un número definido de valores analógicos convertidos (digitalizados). El nivel parametrizado (débil, medio, fuerte) determina el número de señales analógicas que se tomarán para calcular el valor medio. Si el diagnóstico de rebase está activado, se generará un evento de diagnóstico en caso de rebase.
	Direcciones E/S	Se definen el área de las direcciones de entrada así como la memoria imagen de proceso.
	Identificador de hardware	Se muestra el identificador de hardware del dispositivo.
Contadores rápidos (HSC)	Contadores rápidos (HSC)#	Los contadores rápidos suelen emplearse como accionamiento de contadores de maniobras. Consulte: Configurar contadores rápidos (Página 1656)

Grupo	Propiedades	Descripción
Generadores de impulsos (PTO/PWM)	PTO#/PWM#	Un generador de impulsos se activa y recibe información del proyecto. Para parametrizar un generador de impulsos activado se ajusta el uso como PWM (Pulse Width Modulation; "modulación del ancho de pulso") o como PTO (Pulse Train Output; "secuencia de pulso"). Para PWM hay que definir la fuente de salida, base de tiempo, formato de duración de impulso, tiempo de ciclo y duración de impulso inicial. Como salida de hardware está prevista una salida de impulso. La salida PWM es controlada por la instrucción CTRL_PWM, véase CTRL_PWM (Página 4961). Para PTO hay que indicar el origen de la salida. Como salidas de hardware están previstas una salida de impulso y una de sentido. Los PTO se utiliza junto con un HSC en el tipo de contaje "Eje de movimiento" y se controlan con el objeto tecnológico Motion Control (véase la palabra clave "Motion Control S7-1200") En las direcciones E/S y de diagnóstico se muestra la identificación de hardware y, si se selecciona la función PWM, también es posible seleccionar el área de las direcciones de salida y la memoria imagen de proceso.
Arranque	Arranque tras conexión (POWER ON)	Ajuste del comportamiento de arranque tras una transición de desconexión (POWER OFF)/conexión (POWER ON). Consulte: Principios básicos del estado operativo "ARRANQUE" (Página 1623)
	Comparación entre configuración teórica y real	Determina si los módulos (SM, SB, CM, CP o también la CPU) pueden sustituirse: • Arranque de la CPU solo en caso de compatibilidad • Arranque de la CPU también si hay diferencias Ejemplo: un módulo de señales con 16 entradas digitales y 16 salidas digitales (DI16/DQ16) puede ser un sustituto aceptable para un módulo de señales con 8 salidas digitales (DQ8) o 4 entradas digitales (DI4).
	Tiempo de parametrización para la periferia descentralizada	Determina el intervalo de tiempo máximo (estándar: 60000 ms) en que la periferia descentralizada debe cargarse. (La CPU suministra a los módulos CM y CP tensión y parámetros de comunicación durante el arranque. Este tiempo de parametrización permite un intervalo durante el cual los módulos E/S conectados a CM o CP deben cargarse). La CPU pasa a RUN en cuanto la periferia descentralizada está cargada y lista para el servicio, independientemente del parámetro "Tiempo de parametrización para periferia descentralizada". Si la periferia descentralizada no está cargada dentro de este intervalo de tiempo, la CPU pasa a RUN sin la periferia descentralizada.
Ciclo	Tiempo de ciclo máximo y mínimo.	Definición de un tiempo de ciclo máximo y de un tiempo de ciclo mínimo fijo. Si el tiempo de ciclo excede el tiempo de vigilancia del ciclo, la CPU pasa al estado operativo STOP. Consulte: Tiempo de ciclo y tiempo de vigilancia del ciclo (Página 1650)

Grupo	Propiedades	Descripción
Carga de comunicación	Porcentaje máximo del ciclo para la comunicación (en %)	Controla hasta cierto punto la duración de los procesos de comunicación que también prolongan el tiempo de ciclo. Los procesos de comunicación pueden ser, p. ej.: transferencia de datos a otra CPU o carga de bloques (lanzada vía PC). Consulte: Carga de ciclo por comunicación (Página 1651)
Marcas de sistema y ciclo	Bits de marcas de sistema y bits de marcas de ciclo	Las marcas de sistema se utilizan en las consultas siguientes: • ¿El ciclo actual es el primer ciclo después de desconexión (POWER OFF)/conexión (POWER ON)? • ¿Han cambiado los estados de diagnóstico respecto del ciclo anterior? • Consulta en "1" (high) • Consulta en "0" (low) Las marcas de ciclo cambian periódicamente de valor, en intervalos definidos. Consulte: Uso de marcas de sistema (Página 1668) Consulte: Uso de marcas de ciclo (Página 1669)
Servidor web	Actualización automática	Envía periódicamente la página web solicitada con los datos actuales de la CPU al navegador web. La duración de los períodos se introduce en "Intervalo de actualización". La actualización automática solo se puede activar cuando el servidor web está activado. Consulte: Activar la actualización automática (Página 942)
	Páginas web definidas por el usuario	Permite acceder a páginas web de libre configuración de la CPU con un navegador web. Consulte: Información importante sobre las páginas web definidas por el usuario (Página 946)
Hora	Hora local y horario de verano	Ajuste de la zona horaria en la que se utiliza la CPU y ajuste del cambio entre horario de verano y de invierno.
Protección	Protección y contraseña del acceso de lectura/escritura	Ajuste de la protección contra lectura o escritura y de la contraseña de acceso a la CPU. Consulte: Posibilidades de ajuste para el nivel de protección (FW V1 a V3) (Página 1670) Consulte: Posibilidades de ajuste para la protección (FW a partir de V4) (Página 1671)
Recursos de conexión	-	Visualización de los recursos de conexión de la CPU disponibles, reservados y configurados.
Sinóptico de direcciones	-	Representación en forma de tabla de todas las direcciones utilizadas en la CPU para las entradas y salidas integradas así como para los módulos insertados. Las direcciones que no utiliza ningún módulo se representan como hueco. Es posible filtrar la vista por • Direcciones de entrada • Direcciones de salida • Huecos direcciones

Consulte también

Definir las direcciones de entrada y salida (Página 972)

Parametrizar OBs de alarma de proceso (Página 1693)

Acceso a las direcciones de periferia (Página 1636)

Direccionamiento de módulos (Página 971)

Particularidades del arranque (Página 1626)

Tiempo de ciclo y tiempo de vigilancia del ciclo

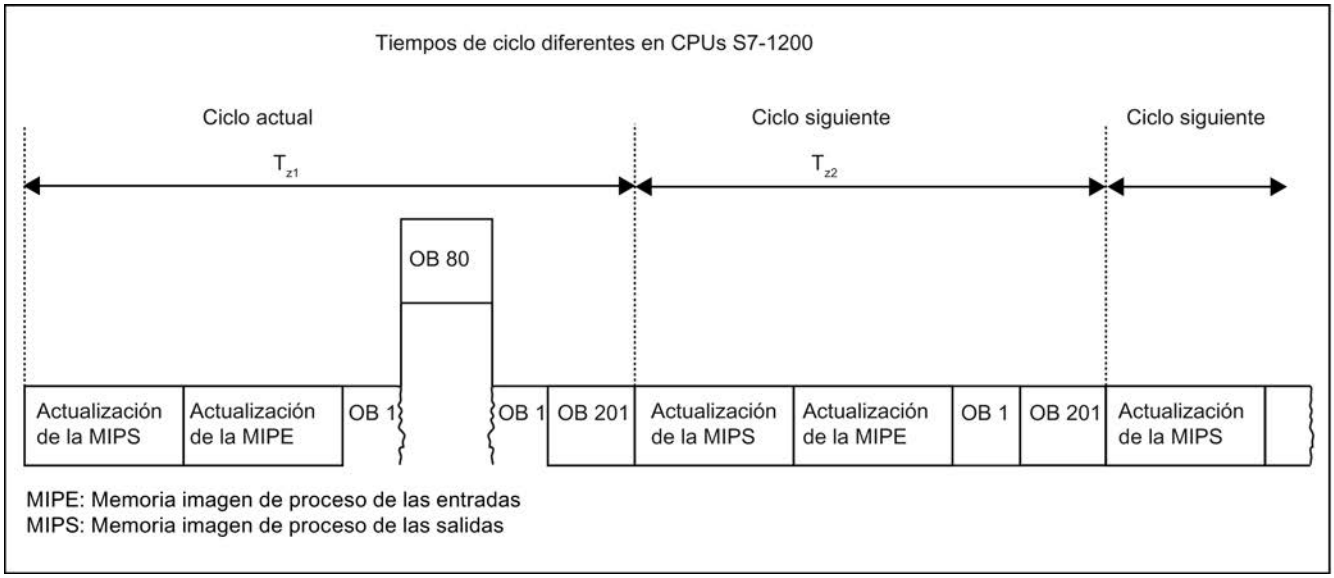
Función

El tiempo de ciclo es el tiempo que necesita el sistema operativo para el procesamiento cíclico del programa y para todas las partes del programa que interrumpen dicho ciclo. El procesamiento del programa se interrumpe debido a:

- Errores de tiempo y errores 2xMaxCycleTime
- Actividades del sistema, p. ej. actualización de la memoria imagen de proceso

Por consiguiente, el tiempo de ciclo (TC) no es igual en todos los ciclos.

La figura siguiente muestra un ejemplo de tiempos de ciclo diferentes (TC1 ≠ TC2) para CPU S7-1200:



En el ciclo actual, el OB cíclico utilizado (p. ej. el OB 1) es interrumpido por un error de tiempo (p. ej. el OB 80). A continuación del OB cíclico se procesa el siguiente OB del ciclo, el 201.

Tiempo de vigilancia del ciclo

El sistema operativo vigila si el tiempo de ejecución del programa cíclico alcanza un límite superior configurable, el tiempo de ciclo máximo. Dicho tiempo de vigilancia se reinicia llamando la instrucción RE_TRIGR en cualquier punto del programa.

Si el programa cíclico excede el tiempo de vigilancia del ciclo, el sistema operativo intenta arrancar el OB de error de tiempo (OB 80). Si el OB no está disponible, la CPU ignora que se ha excedido el tiempo de vigilancia del ciclo.

Además de vigilar el tiempo de ejecución para asegurar que no se excede el tiempo de vigilancia del ciclo, también se garantiza el mantenimiento de un tiempo de ciclo mínimo. Para ello, el sistema operativo retarda el inicio de un ciclo nuevo hasta que se ha alcanzado el tiempo de ciclo mínimo. Durante este tiempo de espera se procesan eventos nuevos y servicios del sistema operativo.

Si el tiempo de vigilancia del ciclo se excede por segunda vez, p. ej., mientras se procesa el OB de error de tiempo (error 2xMaxCycleTime), la CPU pasa al estado operativo STOP.

Carga de ciclo por comunicación

Función

El tiempo de ciclo de la CPU se prolonga debido a procesos de comunicación. Dichos procesos de comunicación incluyen, p. ej.:

- Transferencia de datos a otra CPU
- Carga de bloques lanzada por una programadora

La duración de estos procesos de comunicación se controla hasta cierto punto con el parámetro de la CPU "Carga de ciclo por comunicación".

Además de los procesos de comunicación, las funciones de test también prolongan el tiempo de ciclo. En este caso, el parámetro "Carga de ciclo por comunicación" influye en la duración.

Funcionamiento del parámetro

En el parámetro "Carga de ciclo por comunicación" se indica el porcentaje de capacidad de procesamiento de la CPU que debe estar disponible para los procesos de comunicación. La CPU asigna continuamente dicha capacidad a la comunicación. Si la comunicación no requiere esta capacidad de procesamiento, ésta queda a disposición del procesamiento del programa.

Repercusión en el tiempo de ciclo real

El uso del parámetro "Carga de ciclo por comunicación" prolonga el tiempo de ciclo del bloque de organización cíclico (p. ej. OB 1) en un factor que se calcula según la fórmula siguiente:

$$\frac{100}{100 - \text{"Carga de ciclo por comunicación"}}$$

La fórmula no tiene en cuenta la influencia de eventos asíncronos, como alarmas de proceso o cíclicas, en el tiempo de ciclo.

Si el tiempo de ciclo se prolonga debido a procesos de comunicación, también es posible que se produzcan más eventos asíncronos dentro del tiempo de ciclo del bloque de organización cíclico. Esto todavía prolonga más el ciclo. La prolongación depende de cuántos eventos aparezcan y de cuánto dure su procesamiento.

Ejemplo 1 - no hay eventos asíncronos adicionales:

Si se ajusta el parámetro "Carga de ciclo por comunicación" al 50 %, el tiempo de ciclo del bloque de organización cíclico puede duplicarse.

Ejemplo 2 - hay eventos asíncronos adicionales:

Con un tiempo de ciclo puro de 500 ms, el tiempo de ciclo real puede ascender hasta 1000 ms debido a una carga de comunicación del 50 %, siempre que la CPU tenga suficientes peticiones de comunicación para procesar. Si, paralelamente, se procesa una alarma cíclica cada 100 ms con un tiempo de procesamiento de 20 ms, dicha alarma sin carga de comunicación prolongaría el ciclo en un total de 5*20 ms = 100 ms, es decir, el tiempo de ciclo real sería de 600 ms. Puesto que una alarma cíclica también interrumpe la comunicación, repercute en el tiempo de ciclo con 10 * 20 ms con una carga de comunicación del 50 %, es decir, en este caso el tiempo de ciclo real no sería de 1000 ms, sino de 1200 ms.

Nota

Tenga en cuenta las indicaciones siguientes:

- Verifique las repercusiones de un cambio de valor del parámetro "Carga de ciclo por comunicación" en el funcionamiento de la instalación.
- La carga por comunicación debe tenerse en cuenta al ajustar el tiempo de ciclo mínimo, puesto que de lo contrario se pueden producir errores de tiempo.

Recomendaciones

- Aumente el valor sólo si la CPU se utiliza principalmente para fines de comunicación y el programa de usuario es de duración no crítica.
- En todos los demás casos, el valor sólo debería reducirse.

Funciones de reloj

Principios básicos de las funciones de reloj

Todas las CPUs S7-1200 disponen de un reloj interno. El respaldo permite visualizar la hora correcta en caso de interrupción de la fuente de alimentación durante un máximo de 10 horas.

Formato de hora

El reloj muestra siempre la hora con una resolución de 1 milisegundo y la fecha con el día de la semana. Se tiene en cuenta el cambio de hora provocado por el horario de verano.

Ajustar y leer la hora

Ajuste y lectura de la hora con instrucciones

La hora y la fecha del reloj de la CPU se pueden ajustar, iniciar y leer en el programa de usuario por medio de las instrucciones siguientes:

- Ajustar la hora: "WR_SYS_T"
- Leer la hora "RD_SYS_T"
- Leer la hora local "RD_LOC_T"
- Ajustar zona horaria "SET_TIMEZONE"

Ajuste manual

También es posible leer y ajustar la hora manualmente en la vista online y de diagnóstico, en "Funciones > Ajustar hora".

Consulte también

WR_SYS_T: Ajustar la hora (Página 4695)

RD_SYS_T: Leer la hora (Página 4696)

RD_LOC_T: Leer hora local (Página 4697)

SET_TIMEZONE: Ajustar zona horaria (Página 4700)

Parametrizar el reloj

Parámetros del reloj

Mediante los parámetros del reloj se ajusta lo siguiente:

- Activar sincronización horaria vía servidor NTP
  Active esta casilla de verificación si hay que sincronizar el reloj interno con el procedimiento de sincronización NTP.
- Servidor de hora de la red
  Deben configurarse las direcciones IP de hasta cuatro servidores NTP.
- Intervalo de actualización
  El intervalo de actualización define el espacio de tiempo entre las consultas de hora.

Consulte también

Sincronización horaria, procedimiento NTP - Parametrizacíon (Página 2000)

Contadores rápidos

Generalidades de los contadores rápidos

Introducción

Los contadores rápidos suelen emplearse como accionamientos de contadores de maniobras en los que un eje que funciona a una velocidad constante está equipado con un encoder incremental de posición. Dicho encoder procura un número determinado de valores de contaje por vuelta así como un impulso de reset una vez por vuelta. El o los generadores de impulsos de reloj y el impulso de reset del encoder incremental de posición proporcionan las entradas del contador rápido.

Las diferentes CPU S7-1200 disponen de un número variable de contadores rápidos:

CPU S7-1200	Número de HSC	Denominación HSC
CPU 1211C	3 (con Signal Board digital 4)*	HSC1…3 (y HSC5)*
CPU 1212C	4 (con Signal Board digital 5)*	HSC1…4 (y HSC5)*
CPU 1214C CPU 1215C CPU 1217C	6	HSC1…6

* con Signal Board DI2/DO2

Funcionamiento

El primer valor de varios preconfigurados se carga en el contador rápido. Las salidas deseadas se activan durante el intervalo de tiempo en que el valor actual del contador es inferior al valor preconfigurado. El contador se configura de modo que se produce una interrupción cuando el valor actual del contador es igual al valor preconfigurado o cuando el contador se inicializa.

Si el valor actual es igual al valor preconfigurado y se produce un evento de interrupción, se carga un valor preconfigurado nuevo y se activa el próximo estado lógico para las salidas. Si se produce un evento de interrupción debido a que el contador se inicializa, se activarán el primer valor preconfigurado y los primeros estados lógicos de las salidas, y se repetirá el ciclo.

Puesto que las interrupciones se producen a una velocidad muy inferior de la que cuenta el contador rápido, es posible implementar un control preciso de las operaciones rápidas con una influencia relativamente reducida en todo el ciclo del sistema de automatización. Puesto que existe la posibilidad de asignar interrupciones a determinados programas de interrupción,

cada nuevo ajuste predeterminado puede cargarse en un programa de interrupción separado, con lo que el estado se controla fácilmente.

Nota

También es posible procesar todos los eventos de interrupción en un solo programa.

Algoritmos de contaje de los distintos contadores

Todos los contadores funcionan de la misma manera, pero no todos los contadores rápidos soportan todos los algoritmos de contaje. Existen cuatro algoritmos básicos de contaje:

- Contador monofásico con control interno de sentido
- Contador monofásico con control externo de sentido
- Contador bifásico con 2 entradas de impulso de reloj
- Contador A/B

Todos los contadores rápidos pueden utilizarse con o sin entrada de reset. Si la entrada de reset se activa, se inicializa el valor actual. El valor actual permanece inicializado hasta que se desactiva la entrada de reset.

Consulte también

Configurar contadores rápidos (Página 1656)

Dependencias entre tipo de contador y entradas de contador (Página 1655)

Dependencias entre tipo de contador y entradas de contador

Generalidades del tipo de contador y las entradas de contador

A los contadores rápidos se les asignan tipos y entradas de contador y, por encima, las funciones como generador de impulsos de reloj, control de sentido y resetear. Rigen las reglas siguientes:

- No es posible utilizar una entrada para dos funciones distintas.
- Si el tipo actual de contador rápido definido no requiere una entrada, ésta estará disponible para otros fines.

Así, por ejemplo, si se utiliza HSC1 en el tipo de contador 1 que requiere las entradas I0.0 e I0.3, es posible utilizar I0.1 para interrupciones de flancos o para HSC2.

Si, por ejemplo, se utiliza HSC1 y HSC5, en los tipos de contador Contaje y Frecuencia se utilizarán siempre las entradas I0.0 (HSC1) e I1.0 (HSC5). Por ello ambas entradas no estarán disponibles para otras funciones cuando se utilicen los contadores.

Hay otras entradas disponibles si se utiliza una Signal Board digital.

Resumen de las dependencias entre el tipo de contador y las entradas de contador

Tipo de contador	Descripción	Entradas		
	HSC1	I0.0 (CPU) I4.0 (Signal Board)	I0.1 (CPU) I4.1 (Signal Board)	I0.3 (CPU) I4.3 (Signal Board)
	HSC2	I0.2 (CPU) I4.2 (Signal Board)	I0.3 (CPU) I4.3 (Signal Board)	I0.1 (CPU) I4.1 (Signal Board)
	HSC3*	I0.4 (CPU)	I0.5 (CPU)	I0.7 (CPU)
	HSC4 (solo CPU 1212/14/15/17C)	I0.6 (CPU)	I0.7 (CPU)	I0.5 (CPU)
	HSC5 (solo CPU 1214/15/17C)**	I1.0 (CPU) I4.0 (Signal Board)	I1.1 (CPU) I4.1 (Signal Board)	I1.2 (CPU) I4.3 (Signal Board)
	HSC6 (solo CPU 1214/15/17C)**	I1.3 (CPU)	I1.4 (CPU)	I1.5 (CPU)
Contaje / frecuencia	Contador monofásico con control interno de sentido	Generador de impulsos de reloj	-	-
Contaje				Resetear
Contaje / frecuencia	Contador monofásico con control externo de sentido	Generador de impulsos de reloj	Sentido	-
Contaje				Resetear
Contaje / frecuencia	Contador bifásico con 2 entradas de impulso de reloj	Generador de impulsos de reloj hacia delante	Generador de impulsos de reloj hacia atrás	-
Contaje				Resetear
Contaje / frecuencia	Contador A/B	Generador de impulsos de reloj A	Generador de impulsos de reloj B	-
Contaje				Resetear
Eje de movimiento	Generadores de impulsos PWM/ PTO	HSC1 y HSC2 soportan el modo de contaje Eje de movimiento para los generadores de impulsos PTO1 y PTO2: • HSC1 evalúa la salida Q0.0 para PTO1 por el número de impulsos • HSC2 evalúa la salida Q0.2 para PTO2 por el número de impulsos Q0.1 se utiliza como salida para el sentido del movimiento.		

* HSC3 sólo puede utilizarse para la CPU 1211 sin entrada de reset

** HSC5 también puede utilizarse para la CPU 1211/12 si se utiliza una Signal Board DI2/DO2

Consulte también

Generalidades de los contadores rápidos (Página 1654)

Configurar contadores rápidos (Página 1656)

Configurar contadores rápidos

Requisitos

Se ha insertado una CPU S7-1200 en la configuración hardware.

Procedimiento

Para configurar un contador rápido, proceda del siguiente modo:

1. Seleccione una CPU S7-1200 en la vista de dispositivos o redes.
2. En la ventana de inspección, haga clic en el contador rápido deseado en "Propiedades > Contadores rápidos":
   - CPU 1211C: de HSC1 a HSC3 (con Signal Board DI2/DO2 también HSC5)
   - CPU 1212C: de HSC1 a HSC4 (con Signal Board DI2/DO2 también HSC5)
   - CPU 1214C / 1215C / 1217C: de HSC1 a HSC6
3. Active el contador rápido en el grupo de parámetros "General" con la casilla de verificación correspondiente.

   Nota

   Si se utiliza una CPU 1211C o CPU 1212C con Signal Board DI2/DO2, también es posible activar el contador rápido HSC5.

   Nota

   Si se activan los generadores de impulsos y se emplean como PTO1 o PTO2, éstos utilizarán los contadores rápidos correspondientes HSC1 o HSC2 con el modo de contaje "Eje de movimiento" para evaluar las salidas de hardware. Si se configuran los contadores rápidos HSC1 o HSC2 para otras tareas de contaje, no estarán disponibles para los generadores de impulsos PTO1 o PTO2.

   En caso necesario, existe la posibilidad de introducir un nombre y un comentario para el contador rápido.
4. El funcionamiento del contador rápido se define en el grupo de parámetros "Función":
   - Modo de contaje: elija en la lista desplegable lo que debe contarse.
   - Fase operativa: elija el algoritmo de contaje en la lista desplegable.
   - Origen de entrada: elija en la lista desplegable las entradas integradas en la CPU o las entradas de una Signal Board digital opcional como origen de entrada para los impulsos de contaje.
   - Sentido de contaje indicado por: si se ha seleccionado una fase operativa monofásica, elija en la lista desplegable si el sentido de contaje debe indicarse internamente por medio de un parámetro SFB del programa de usuario o bien externamente por medio de una entrada digital.
   - Sentido de contaje inicial: si en el sentido de contaje se ha ajustado el programa de usuario como control interno de sentido, elija en la lista desplegable el sentido para iniciar el contaje.
   - Período de medición de frecuencia: si se ha ajustado la frecuencia como modo de contaje, elija en la lista desplegable la duración de los períodos de medición de frecuencia.

5. Defina los valores iniciales y la condición de reset del contador rápido en el grupo de parámetros "Restablecer a valores iniciales":
   - Valor de contador inicial: introduzca un valor inicial para el contador rápido.
   - Valor de referencia inicial: introduzca un valor máximo para el contador rápido.

   Aquí también se define si el contador rápido debe utilizar una entrada de reset y se ajusta el nivel de señal correspondiente para dicha entrada en la lista desplegable.
6. Configure la reacción del contador rápido a determinados eventos en el grupo de parámetros "Configuración de eventos". Los eventos siguientes lanzan una alarma:
   - El valor de contador equivale al valor de referencia.
   - Se ha generado un evento de reset externo.
   - Se ha lanzado un cambio de sentido.

   Active una reacción a la alarma mediante la casilla de verificación e introduzca un nombre y una interrupción de hardware para la alarma en la lista desplegable.
7. Asigne la dirección inicial del contador rápido en el grupo de parámetros "Direcciones de diagnóstico E/S".

Nota

En el grupo de parámetros "Entradas de hardware" se ve qué entradas de hardware y valores se utilizan para el reloj, la determinación del sentido, el impulso de reset y la velocidad de contaje máxima.

Resultado

Ahora, los parámetros del contador rápido están adaptados a los requerimientos del proyecto.

Consulte también

Generalidades de los contadores rápidos (Página 1654)

Dependencias entre tipo de contador y entradas de contador (Página 1655)

Comunicación punto a punto

Resumen de la comunicación punto a punto

La comunicación PtP es una comunicación a través de una interfaz serie que utiliza una transferencia de datos UART estandarizada (Universal Asynchronous Receiver Transmitter). La S7-1200 utiliza módulos de comunicación con una interfaz RS232 o RS485 para establecer la comunicación PtP.

Funciones de la comunicación punto a punto

La comunicación punto a punto (PtP) ofrece una gran cantidad de posibilidades de aplicación:

- Transmisión directa de información a un dispositivo externo, p. ej. una impresora o un lector de códigos de barras
- Recepción de información de dispositivos externos, p. ej. lectores de códigos de barras, lectores RFID, cámaras y sistemas ópticos de otros fabricantes y muchos otros dispositivos.
- Intercambio de información con dispositivos de otros fabricantes, p. ej. dispositivos GPS, radio módems y muchos más

El protocolo Freeport

La S7-1200 soporta el protocolo Freeport para la comunicación serie basada en caracteres. La comunicación Freeport permite configurar el protocolo de transferencia de datos utilizando exclusivamente el programa de usuario.

Siemens dispone de librerías con funciones de comunicación Freeport que pueden utilizarse en el programa de usuario:

- USS Drive Protocol
- Modbus RTU Master Protocol
- Modbus RTU Slave Protocol

Consulte también

Configurar el puerto de comunicación (Página 1660)

Uso de módulos de comunicación RS232 y RS485

Módulos de comunicación con interfaces RS232 y RS485

En una CPU S7-1200 es posible utilizar dos módulos de comunicación distintos:

- Módulo de comunicación RS232
- Módulo de comunicación RS485

Los módulos de comunicación se conectan a la CPU S7-1200 a través del canal I/O de la izquierda. Se admiten tres módulos como máximo.

Propiedades de los módulos de comunicación

Los módulos de comunicación se distinguen por las propiedades siguientes:

- Soporte del protocolo Freeport
- Configuración a través del programa de usuario mediante instrucciones avanzadas y funciones de librería

Configurar el puerto de comunicación

Configurar el puerto de comunicación

Después de insertar un módulo de comunicación con una interfaz RS232 o RS485, hay que ajustar los parámetros de la interfaz. La interfaz se parametriza en las propiedades de la misma, aunque también es posible controlar los parámetros de interfaz por medio del programa de usuario con la instrucción PORT_CFG. La descripción siguiente hace referencia a la configuración gráfica.

Nota

Si se modifica el ajuste del puerto por medio del programa de usuario, se sobrescribirán los ajustes de la configuración gráfica.

Asimismo, hay que tener en cuenta que los ajustes realizados desde el programa de usuario no se conservan en caso de una caída de tensión.

Requisitos

- Hay un módulo de comunicación insertado.
- Se encuentra en la vista de dispositivos.

Procedimiento

Para configurar el puerto de comunicación, proceda del siguiente modo:

1. Seleccione la interfaz en la representación gráfica de la vista de dispositivos.
   En la ventana de inspección aparecen las propiedades de la interfaz.
2. Elija el grupo "Configuración del puerto" en la navegación local de la ventana de inspección.
   Aparecen los ajustes del puerto.
3. En lista desplegable "Velocidad de transferencia" elija la velocidad para la transferencia de datos. Si la comunicación se ha programado libremente, tenga en cuenta la influencia de la velocidad de transferencia en el tiempo de conmutación.
4. En la lista desplegable "Paridad" seleccione el tipo de detección de palabras informativas que se transfieren con errores.
5. En la lista desplegable "Bits de datos" elija si un carácter debe tener ocho o siete bits.
6. En la lista desplegable "Bit de parada" seleccione con cuántos bits debe marcarse el final de una palabra transferida.

7. En la lista desplegable "Control de flujo" elija con qué procedimiento debe asegurarse un flujo de datos continuo entre el emisor y el receptor. Este parámetro sólo es ajustable para la interfaz RS232.
   - En el campo "Carácter XON" introduzca un valor HEX cuya aparición provoque la continuación de la transferencia de mensajes. Este parámetro sólo es ajustable para el control del flujo de datos por software.
   - En el campo "Carácter XOFF" introduzca un valor HEX cuya aparición provoque la suspensión de la transferencia de mensajes durante el tiempo de espera parametrizado. Este parámetro sólo es ajustable para el control del flujo de datos por software.
8. En el campo "Tiempo de espera" introduzca en ms el tiempo que deba esperarse tras finalizar un mensaje, hasta que pueda iniciarse la próxima transferencia.

Nota

La interfaz también puede configurarse en la vista de redes. Sin embargo, primero hay que seleccionar el módulo de comunicación en el área de tabla de la vista de redes y, seguidamente, seleccionar la interfaz en la ventana de inspección. Posteriormente, se procede del modo descrito anteriormente.

Consulte también

Ajustar el control del flujo de datos (Página 1661)

Ajustar el control del flujo de datos

Control del flujo de datos

El control del flujo de datos es un procedimiento que procura un comportamiento equilibrado de transmisión y recepción. En condiciones óptimas, con un control inteligente no se pierden datos. Dicho control asegura que un dispositivo no envíe más información de la que puede procesar el interlocutor receptor.

Existen dos procedimientos de control del flujo de datos:

- Control del flujo de datos por hardware
- Control del flujo de datos por software

En ambos métodos, las señales DSR de los interlocutores deben estar activas al empezar la transferencia. Si las señales DSR están inactivas, no se iniciará la transferencia.

El módulo de comunicación RS232 soporta ambos procedimientos, mientras que el módulo de comunicación RS485 no soporta ninguno.

Control del flujo de datos por hardware

El control del flujo de datos por hardware se realiza a través de las señales Request-to-send (RTS) y Clear-to-send (CTS). Con el módulo de comunicación RS232 se transfiere la señal RTS a través de la salida del pin 7. La señal CTS llega por el pin 8.

Si el control del flujo de datos por hardware está activado, la señal RTS se activará siempre que se envíen datos. Al mismo tiempo se vigila la señal CTS para determinar si el dispositivo receptor acepta los datos. Si la señal CTS está activa, el módulo puede transferir datos hasta que se desactive la señal CTS. Cuando se desactiva dicha señal, la transferencia de datos debe suspenderse durante el tiempo de espera parametrizado. Si la señal CTS sigue inactiva una vez transcurrido el tiempo de espera parametrizado, se cancelará la transferencia de datos y se notificará un error al programa de usuario.

Control del flujo de datos con handshake de hardware

Si el control del flujo de datos se regula por medio del handshake de hardware, el dispositivo emisor activa por defecto la señal RTS. En este caso, un dispositivo, p. ej. un módem, puede transferir datos en cualquier momento. No espera la señal CTS del receptor. El dispositivo emisor vigila él mismo su propia transferencia enviando únicamente un número limitado de frames (caracteres) del mensaje, con el fin de evitar un desbordamiento del búfer de recepción, por ejemplo. Si, a pesar de todo, se desborda el búfer de recepción, el dispositivo emisor debe retener el mensaje y notificar un error al programa de usuario.

Control del flujo de datos por software

El control del flujo de datos por software utiliza dentro de los mensajes determinados caracteres que controlan la transferencia. Éstos son caracteres ASCII que se parametrizan en XON y XOFF.

XOFF indica cuándo debe suspenderse una transferencia. XON indica cuándo puede continuar una transferencia.

Cuando el dispositivo emisor recibe el carácter XOFF, debe suspender la transmisión durante el tiempo de espera parametrizado. Si, una vez transcurrido el tiempo de espera parametrizado, se envía el carácter XON, se continuará con la transferencia. Si después del tiempo de espera no llega ningún carácter XON, se notifica un error al programa de usuario.

El control del flujo de datos por software requiere una comunicación dúplex, ya que el interlocutor receptor debe enviar el carácter XON durante la transferencia en curso.

Consulte también

Configurar el puerto de comunicación (Página 1660)

Configuración de la transferencia de mensajes

Comunicación libremente programable

El tráfico de datos entre un módulo de comunicación y un dispositivo conectado externamente a través de una interfaz serie se controla libremente. Para ello hay que definir un protocolo de comunicación. En la comunicación libremente programable se soportan protocolos ASCII y binarios para transferir mensajes.

Dentro del protocolo de comunicación deben definirse los criterios que deben facilitar la detección del principio y fin de un mensaje dentro del flujo de datos actual.

La comunicación libremente programable sólo puede activarse en estado operativo RUN. Si se pasa al estado operativo STOP, se detendrá la comunicación libremente programable.

Definición del protocolo de comunicación

El protocolo de comunicación se define del siguiente modo:

- Con el programa de usuario
  - El comportamiento durante la transmisión de datos se controla por medio de la instrucción SEND_CFG.
  - El comportamiento durante la recepción de datos se controla por medio de la instrucción RCV_CFG.
- Mediante parametrización gráfica en la ventana de inspección

Nota

Si se modifica el protocolo de comunicación a través del programa de usuario, se sobrescribirán los ajustes de la configuración gráfica.

Hay que tener en cuenta que los ajustes realizados desde el programa de usuario no se conservarán en caso de pérdida de tensión.

Consulte también

Comunicación libremente programable con dispositivos RS232 (Página 1663)

Definir los ajustes para la transmisión (Página 1665)

Definición del principio del mensaje (Página 1666)

Definición del fin del mensaje (Página 1667)

Comunicación libremente programable con dispositivos RS232

Cable multimaestro RS232/PPI y comunicación libremente programable con dispositivos RS232

Mediante el cable multimaestro RS232/PPI y la comunicación libremente programable es posible conectar a los módulos de comunicación de la S7-1200 muchos dispositivos compatibles con el estándar RS232. Sin embargo, primero hay que ajustar el cable para el modo "PPI/comunicación libremente programable".

Ajustes del cable

Los interruptores del cable deben ajustarse del siguiente modo:

- El interruptor 5 debe estar a 0
- El interruptor 6 ajusta el modo local (DCE) o el modo remoto (DTE):
  - Interruptor a 0 para el modo local
  - Interruptor a 1 para el modo remoto

Conmutación entre modo de transmisión y recepción

El cable multimaestro RS232/PPI está en modo de transmisión cuando se envían datos de la interfaz RS232 a la interfaz RS485. El cable está en modo de recepción cuando está en estado de reposo o cuando se envían datos de la interfaz RS485 a la interfaz RS232. El cable cambia inmediatamente del modo de recepción al de transmisión cuando se detectan caracteres en la línea de transmisión RS232.

Velocidades de transferencia soportadas

El cable multimaestro RS232/PPI soporta velocidades de transferencia entre 1200 baudios y 115,2 kbaudios. Mediante los interruptores DIP del cable PC/PPI se ajusta el cable multimaestro RS232/PPI a la velocidad de transferencia deseada.

La tabla siguiente muestra la posición de los interruptores para las velocidades de transferencia correspondientes:

Velocidad de transferencia	Tiempo de conmutación	Ajustes (1 = arriba)
115200 bits/s	0,15 ms	110
57600 bits/s	0,3 ms	111
38400 bits/s	0,5 ms	000
19200 bits/s	1,0 ms	001
9600 bits/s	2,0 ms	010
4800 bits/s	4,0 ms	011
2400 bits/s	7,0 ms	100
1200 bits/s	14,0 ms	101

El cable retorna al modo de recepción cuando la línea de transmisión RS232 está en modo de reposo durante un tiempo determinado, que se define como tiempo de conmutación del cable. La velocidad de transferencia ajustada influye en el tiempo de conmutación, tal como muestra la tabla.

Influencia del tiempo de conmutación

Al trabajar con un cable multimaestro RS232/PPI en un sistema que también utiliza la comunicación libremente programable, el programa debe considerar el tiempo de conmutación, por los motivos siguientes:

- El módulo de comunicación reacciona a los avisos enviados por el dispositivo RS232. Una vez el módulo de comunicación del dispositivo RS232 ha recibido una petición, debe retardar la transmisión del aviso de reacción por un intervalo de tiempo mayor o igual al tiempo de conmutación del cable.
- El dispositivo RS232 reacciona a avisos enviados por el módulo de comunicación. Una vez el módulo de comunicación del dispositivo RS232 ha recibido un aviso de reacción, debe retardar la transmisión del próximo aviso de petición por un intervalo de tiempo mayor o igual al tiempo de conmutación del cable.

En ambas situaciones, el cable multimaestro RS232/PPI tiene tiempo suficiente, gracias al retardo, para pasar del modo de recepción al de transmisión para que los datos puedan enviarse de la interfaz RS485 a la interfaz RS232.

Consulte también

Configuración de la transferencia de mensajes (Página 1662)

Definir los ajustes para la transmisión (Página 1665)

Definición del principio del mensaje (Página 1666)

Definición del fin del mensaje (Página 1667)

Definir los ajustes para la transmisión

Enviar mensajes

Existe la posibilidad de programar pausas entre los diferentes mensajes.

La tabla siguiente muestra qué pausas pueden preajustarse:

Parámetro	Definición
Retardo RTS ON	Se parametriza el tiempo que debe transcurrir después de la petición de transmisión RTS (Request to send) antes de que empiece la transferencia de datos propiamente dicha.
Retardo RTS OFF	Se parametriza el tiempo que debe transcurrir una vez finalizada la transferencia por completo antes de que se desactive la señal RTS.
Enviar pausa al inicio del mensaje	Se determina que, cada vez que inicie una transferencia de mensajes, se envíe adicionalmente una pausa si ha transcurrido el tiempo de retardo RTS ON. El tiempo de pausa se indica en tiempos de bit.
Enviar Idle Line tras la pausa	Se determina que, tras una pausa parametrizada al inicio del mensaje, se emita la señal "Idle Line" y, por tanto, la línea se señalice como "desocupada". Para activar el parámetro tiene que estar activado "Enviar pausa al inicio del mensaje". La duración de la señal "Idle Line" se indica en tiempos de bit.

Consulte también

Definición del principio del mensaje (Página 1666)

Definición del fin del mensaje (Página 1667)

Comunicación libremente programable con dispositivos RS232 (Página 1663)

Definición del principio del mensaje

Detección del principio del mensaje

Para indicar al receptor cuándo ha finalizado la transferencia de un mensaje y cuándo empieza la próxima transferencia, hay que definir criterios en el protocolo de transferencia que identifiquen el final y el principio de un mensaje.

Cuando se cumple un criterio que indica el principio de un mensaje, se empiezan a buscar criterios para el fin del mensaje en el flujo de datos.

Existen dos procedimientos distintos para detectar el principio del mensaje:

- Empezar con un carácter cualquiera:
  Cualquier carácter puede definir el principio de un mensaje. Éste es el procedimiento predeterminado.
- Empezar con una condición especial:
  El principio del mensaje se detecta gracias a condiciones definidas.

Condiciones para detectar el principio de un mensaje

La tabla siguiente muestra las diferentes posibilidades que existen para definir el principio de un mensaje:

Parámetro	Definición
Detectar inicio del mensaje por Line Break	El receptor detecta un Line Break cuando se interrumpe el flujo de datos recibidos durante un tiempo superior a la longitud de un carácter. En este caso, se determina el principio del mensaje por el Line Break.
Detectar inicio del mensaje por Idle Line	El principio de un mensaje se detecta cuando la línea de transferencia está en estado "Idle" durante un tiempo determinado (indicado en tiempos de bit) para enviar y va seguida de un evento, p. ej. la recepción de un carácter.
Detectar el inicio del mensaje por un solo carácter	El principio de un mensaje se detecta cuando aparece un carácter determinado. El carácter en cuestión se introduce como valor HEX.
Detectar el inicio del mensaje por una cadena de caracteres	El principio de un mensaje se detecta cuando aparece una cadena de caracteres definida en el flujo de datos. Es posible definir un máximo de cuatro cadenas de caracteres con cinco caracteres como máximo cada una.

Las diferentes condiciones pueden enlazarse entre sí del modo deseado.

Consulte también

Definir los ajustes para la transmisión (Página 1665)

Comunicación libremente programable con dispositivos RS232 (Página 1663)

Definición del fin del mensaje

Detección del fin del mensaje

Para indicar al receptor cuándo ha finalizado la transferencia de un mensaje y cuándo empieza la próxima transferencia, hay que definir criterios en el protocolo de transferencia que identifiquen el final y el principio de un mensaje.

Existen un total de seis procedimientos distintos para detectar el fin de un mensaje, que pueden enlazarse entre sí del modo deseado. La tabla siguiente muestra las diferentes posibilidades de ajuste:

Parámetro	Definición
Detectar fin del mensaje por tiempo de mensaje excedido	El fin del mensaje se detecta automáticamente cuando se excede la duración máxima predefinida de un mensaje. Se admiten valores entre 0 y 65535 ms.
Detectar fin del mensaje por tiempo de respuesta excedido	El fin del mensaje se detecta cuando no se recibe ninguna respuesta dentro del tiempo predefinido después de transferir datos. Se admiten valores entre 0 y 65535 ms.
Detectar fin del mensaje por tiempo excedido entre caracteres	El fin del mensaje se detecta cuando se excede un tiempo predefinido (en tiempos de bit) entre dos caracteres. Se admiten valores entre 0 y 2500 tiempos de bit. La CPU S7-1200 prevé un tiempo máximo de ocho segundos, aunque el tiempo ajustado sea superior.
Detectar fin del mensaje por longitud máxima	El fin del mensaje se detecta cuando se excede la longitud máxima de un mensaje. Se admiten valores entre 1 y 1023 caracteres.

Parámetro	Definición
Leer longitud del mensaje en el mensaje	El propio mensaje contiene información sobre su longitud. El fin del mensaje se alcanza cuando se llega al valor tomado del mensaje. Con los parámetros siguientes se definen los caracteres que deben considerarse para evaluar la longitud del mensaje: • Offset del campo de longitud en el mensaje El valor determina la posición del carácter en el mensaje que debe considerarse para determinar la longitud del mismo. Se admiten valores entre 1 y 1022 caracteres. • Tamaño del campo de longitud El valor indica cuántos caracteres deben considerarse a partir de la primera posición de evaluación para determinar la longitud del mensaje. Se admiten valores de 0, 1, 2 y 4 caracteres. • El campo de longitud que sigue a los datos (no forma parte de la longitud del mensaje) El valor indica el número de bytes posteriores al campo de longitud que no deben considerarse al evaluar la longitud del mensaje. Se admiten valores entre 0 y 255 caracteres.
Detectar fin del mensaje por una cadena de caracteres	El fin de un mensaje se detecta cuando aparece una cadena de caracteres definida en el flujo de datos. Se pueden especificar como máximo cinco caracteres que deben comprobarse dentro de la cadena. Si los caracteres especificados aparecen en la posición correcta dentro del mensaje, se detectará el fin del mensaje. Para poder detectar el fin del mensaje, por ejemplo, cuando los caracteres 1 y 3 posean un valor determinado, deben estar activadas las casillas de verificación para el carácter 1 y el carácter 3 y haberse introducido un valor de carácter.

Consulte también

Definir los ajustes para la transmisión (Página 1665)

Comunicación libremente programable con dispositivos RS232 (Página 1663)

Uso de marcas de sistema

Marcas de sistema

Una marca de sistema es una marca con valores definidos.

En la parametrización de la marca de sistema se determina el byte de marcas de la CPU que se convertirá en el byte de marcas de sistema.

Uso

Las marcas de sistema pueden utilizarse en el programa de usuario, p. ej. para ejecutar partes del programa sólo en el primer ciclo tras el arranque o para evaluar el diagnóstico al cambiar el estado del mismo. Dos marcas de sistema son permanentemente 1 ó 0.

Bits del byte de marcas de sistema

La tabla siguiente explica el significado de las marcas de sistema:

Bit del byte de marcas de sistema	7	6	5	4	3	2	1	0
Significado	Reservado (=0)	Reservado (=0)	Reservado (=0)	Reservado (=0)	=0	=1	=1 al cambiar el estado del diagnóstico	=1 en el primer ciclo tras el arranque, de lo contrario, 0

Nota

El byte de marcas seleccionado no puede emplearse para almacenar datos en la memoria intermedia.

Uso de marcas de ciclo

Marcas de ciclo

Una marca de ciclo es una marca que modifica periódicamente su estado binario en una relación 1:1 entre impulso y pausa.

En la parametrización de la marca de ciclo se determina el byte de marcas de la CPU que se convertirá en el byte de marcas de ciclo.

Uso

Las marcas de ciclo se utilizan en el programa de usuario, p. ej., para controlar indicadores luminosos con una luz intermitente o lanzar procesos periódicos, como la adquisición de un valor real.

Frecuencias posibles

Cada bit del byte de marcas de ciclo tiene asignada una frecuencia. La tabla siguiente muestra la asignación:

Bit del byte de marcas de ciclo	7	6	5	4	3	2	1	0
Duración del período (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frecuencia (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Nota

Las marcas de ciclo se ejecutan de forma asíncrona al ciclo de la CPU, es decir, en ciclos largos es posible que el estado de la marca de ciclo cambie varias veces.

El byte de marcas seleccionado no puede emplearse para almacenar datos en la memoria intermedia.

Protección

Posibilidades de ajuste para el nivel de protección (FW V1 a V3)

Nivel de protección

A continuación se explica cómo utilizar los diferentes niveles de protección de las CPU S7-1200 V1 a V3.

Repercusiones del nivel de protección ajustado

Existen los niveles de protección siguientes:

- Sin protección: Éste es el comportamiento predeterminado. No es posible introducir ninguna contraseña. El acceso de lectura y escritura está permitido siempre.
- Protección contra escritura: Sólo es posible un acceso en lectura. No se permite modificar datos de la CPU ni cargar bloques ni configuraciones. Están excluidos de la protección contra escritura los accesos al HMI y la comunicación entre varias CPU. Para seleccionar este nivel de protección, es necesario asignar una contraseña.
- Protección de escritura/lectura: no es posible acceder en modo de lectura ni de escritura al área "Dispositivos accesibles" ni a los dispositivos del proyecto que están conectados online. En el área "Dispositivos accesibles" del árbol del proyecto sólo se muestran el tipo de CPU y los datos de identificación. Es posible visualizar información online o bloques en el área "Dispositivos accesibles" o en el proyecto para dispositivos conectados online. Están excluidos de la protección contra escritura los accesos al HMI y la comunicación entre varias CPU. Para seleccionar este nivel de protección, es necesario asignar una contraseña.

Comportamiento de una CPU protegida por contraseña durante el funcionamiento

La protección de la CPU será efectiva una vez que los ajustes se hayan cargado en la CPU.

Antes de ejecutar una función online, se comprobarán los permisos. Si existe protección por contraseña, se pide que se introduzca la contraseña.

Ejemplo: el módulo se ha parametrizado con una protección contra escritura y debe ejecutarse la función "Forzar variable". Puesto que para ello se requiere un acceso de escritura, para ejecutar la función hay que introducir la contraseña parametrizada.

Las funciones protegidas por contraseña solo pueden ejecutarse desde una programadora o un PC en un momento dado. No se permite el acceso con contraseña de otra programadora u otro PC.

El permiso de acceso a los datos protegidos tiene validez mientras dure la conexión online o hasta que se desactive manualmente con "Online > Borrar derechos de acceso". Cuando se cierra el proyecto, el permiso de acceso también se pierde.

Nota

No es posible restringir las funciones para la conducción, visualización y comunicación del proceso.

Algunas funciones quedan protegidas porque se utilizan como datos online. Por consiguiente, las funciones RUN/STOP de la Task Card "Herramientas online" o "Ajustar la hora" del editor de diagnóstico y online están protegidas contra escritura.

Posibilidades de ajuste para la protección (FW a partir de V4)

Nivel de protección

A continuación se explica cómo utilizar los diferentes niveles de acceso de las CPU S7-1200 a partir de V4.

Las CPU S7-1200 ofrecen diferentes niveles de acceso para restringir el uso de determinadas funciones.

La parametrización de los niveles de acceso se realiza en una tabla. Las marcas de verificación verdes de las columnas de la derecha del respectivo nivel de acceso indican qué operaciones son posibles como máximo sin conocer la contraseña de este nivel de acceso. Para utilizar las funciones que no tienen marcada la casilla de verificación es necesario introducir una contraseña.

ATENCIÓN
La configuración de un nivel de acceso no sustituye la protección de know how La parametrización de niveles de acceso impide que se puedan hacer cambios no autorizados en la CPU limitando los derechos de descarga. En cualquier caso, los bloques de la Memory Card no están protegidos contra escritura o lectura. Para proteger el código de los bloques de la Memory Card se utiliza la protección de know-how.

Comportamiento predeterminado

El nivel de acceso predeterminado es "Acceso completo (sin protección)". Todo usuario puede leer y modificar la configuración hardware y los bloques. No hay ninguna contraseña parametrizada y tampoco se necesita para el acceso online.

Los distintos niveles de acceso

En una CPU S7-1200 se pueden parametrizar los siguientes niveles de acceso:

- Acceso completo (sin protección): la configuración hardware y los bloques pueden ser leídos y modificados por cualquier persona.
- Acceso de lectura: con este nivel de acceso solo es posible, sin introducir la contraseña, el acceso de lectura a la configuración hardware y a los bloques, es decir, se pueden cargar la configuración hardware y los bloques en la programadora. Además, también es posible el acceso a HMI y a los datos de diagnóstico.
  Sin introducir la contraseña no se pueden cargar en la CPU ningún módulo ni ninguna configuración hardware. Además, sin la contraseña **no** son posibles las funciones de test con acceso de escritura ni las actualizaciones de firmware.
- Acceso a HMI: con este nivel de acceso, si no se introduce la contraseña solo es posible el acceso a HMI y a los datos de diagnóstico.
  Sin contraseña no se pueden cargar bloques o configuración hardware en la CPU, ni tampoco de esta a la programadora. Además, sin introducir la contraseña **tampoco** es posible: funciones de test con acceso de escritura, cambio del estado operativo (RUN/STOP) y actualización de firmware.
- Sin acceso (protección completa): cuando la CPU dispone de protección completa no se tiene acceso ni de lectura ni de escritura a la configuración hardware y los bloques. Tampoco es posible el acceso HMI. La función de servidor para la comunicación PUT/GET está desactivada en este nivel de acceso (no puede modificarse).
  La legitimación con la contraseña proporciona acceso total a la CPU.

Comportamiento de un módulo protegido por contraseña durante el funcionamiento

La protección de la CPU será efectiva una vez que los ajustes se hayan cargado en la CPU.

Antes de ejecutar una función online, se comprobarán los permisos. Si existe protección por contraseña, se pide que se introduzca la contraseña.

Ejemplo: el módulo se ha parametrizado con acceso de lectura y se desea ejecutar la función "Forzar variable". Puesto que para ello se requiere un acceso de escritura, para ejecutar la función hay que introducir la contraseña parametrizada.

Las funciones protegidas por contraseña solo pueden ejecutarse desde una programadora o un PC en un momento dado. No se permite el acceso desde otra programadora o PC.

El permiso de acceso a los datos protegidos tiene validez mientras dure la conexión online o hasta que se desactive manualmente con "Online > Borrar derechos de acceso".

Todo nivel de acceso permite, incluso sin introducir una contraseña, el acceso ilimitado a determinadas funciones como, p. ej., la identificación mediante la función "Dispositivos accesibles".

Parametrización de niveles de acceso

A continuación se explica cómo parametrizar un nivel de acceso e introducir una contraseña para una CPU S7-1200 a partir de V4.

En una CPU S7-1200 es posible introducir varias contraseñas y, por tanto, establecer diferentes derechos de acceso para distintos grupos de usuarios.

Las contraseñas se introducen en una tabla, de modo que cada una tiene asignado exactamente un nivel de acceso.

El efecto de cada contraseña se muestra en la columna "Acceso" y en un texto explicativo bajo la tabla.

Ejemplo

Para una CPU estándar (es decir, no una CPU F) se elige el nivel de acceso "Sin acceso (protección completa)" y se introduce una contraseña propia para cada nivel de acceso situado encima en la tabla.

Para los usuarios que no conocen ninguna de las contraseñas, la CPU está completamente protegida. Ni siquiera son posibles accesos HMI.

Para los usuarios que conocen una de las contraseñas parametrizadas, el efecto depende de la fila de la tabla en la que se encuentre la contraseña:

- El efecto de la contraseña de la fila 1 (Acceso completo (sin protección)) es como si la CPU no estuviera protegida. Los usuarios que conocen esta contraseña tienen un acceso ilimitado a la CPU.
- El efecto de la contraseña de la fila 2 (acceso de lectura) es como si la CPU estuviera protegida contra escritura. Los usuarios que conocen esta contraseña tienen solo acceso de lectura a la CPU.
- El efecto de la contraseña de la fila 3 (acceso a HMI) es como si la CPU estuviera protegida contra escritura y lectura, de modo que los usuarios que conocen esta contraseña solo disponen de acceso HMI.

Procedimiento

Para parametrizar los niveles de acceso de una CPU S7-1200 proceda del siguiente modo:

1. Abra las propiedades del módulo en la ventana de inspección.
2. Abra la entrada "Protección" en la navegación local.
   En la ventana de inspección se muestra una tabla con los niveles de acceso posibles.

Nivel de acceso	Acceso			Permiso de acceso	
Protección	HMI	Leer	Escribir	Contraseña	Confirmación
Acceso completo (sin protección)	✓	✓	✓		
Acceso de lectura	✓	✓			
Acceso HMI	✓				
Sin acceso (protección completa)					

Figura 8-52 Protección de acceso

3. Active el nivel de acceso deseado en la primera columna de la tabla. Las marcas de verificación verdes de las columnas de la derecha del respectivo nivel de protección indican qué operaciones son posibles sin introducir la contraseña.

4. En la columna "Contraseña", asigne una contraseña en la primera fila para el acceso completo. Repita la contraseña elegida en la columna "Confirmación" con objeto de evitar entradas erróneas.
   Asegúrese de que la contraseña sea lo suficientemente segura, es decir, que no tenga un patrón reconocible por una máquina.
   La entrada de una contraseña en la primera fila "Acceso completo (sin protección)" es obligatoria y permite a la persona que conoce la contraseña acceder de forma ilimitada a la CPU, independientemente del nivel de protección seleccionado.
5. Asigne otras contraseñas a otros niveles de acceso según requiera, siempre que el nivel de protección seleccionado lo permita.
6. Cargue la configuración hardware para que el nivel de acceso se haga efectivo.

Resultado

La configuración hardware y los bloques están protegidos contra los accesos no autorizados conforme al nivel de acceso configurado. Si una operación no se puede ejecutar sin contraseña debido al nivel de acceso parametrizado, se muestra un cuadro de diálogo para introducir una contraseña.

Restricción de los servicios de comunicación

Introducción

La CPU puede ser el servidor de una serie de servicios de comunicación, es decir, que sin necesidad de configurar y programar conexiones para la CPU, otros dispositivos de comunicación pueden acceder a los datos de la CPU.

Con ello se prescinde de la posibilidad para la CPU local como servidor de controlar la comunicación con los clientes.

Si este tipo de comunicación durante el funcionamiento está autorizado o no para la CPU local, se determina mediante el parámetro "Mecanismos de conexión" en el área "Protección" de los parámetros de la CPU.

Permitir acceso vía comunicación PUT/GET del interlocutor remoto

En el ajuste predeterminado, la opción "Permitir acceso vía comunicación PUT/GET del interlocutor remoto (...)" está desactivada. En este caso, el acceso de lectura y de escritura a los datos de la CPU solo será posible a través de conexiones de comunicación que requieren una configuración o programación tanto para la CPU local como para el interlocutor de comunicación. Todavía son posibles, p. ej., los accesos a través de instrucciones BSEND/BRCV.

Las conexiones para las que la CPU local es solo servidor (es decir, cuando para la CPU local no hay una configuración/programación de la comunicación con el interlocutor) no son posibles durante el funcionamiento de la CPU; p. ej.

- en accesos PUT/GET, FETCH/WRITE o por FTP a través de módulos de comunicaciones
- en accesos PUT/GET de otras CPU S7
- en accesos HMI que se realizan a través de la comunicación PUT/GET

Si se desea permitir el acceso del cliente a los datos de la CPU, es decir, si no se desea restringir los servicios de comunicación de la CPU, se debe activar la opción "Permitir acceso vía comunicación PUT/GET del interlocutor remoto".

Bloques de organización

Bloques de organización para el arranque

Descripción

Es posible definir las condiciones marco para el comportamiento en arranque de la CPU, p. ej. los valores de inicialización para "RUN". Escriba para ello un programa de arranque. Este programa consta de uno o varios OBs de arranque (números de OB 100 o >= 123).

El programa de arranque se ejecuta una vez cuando el estado operativo cambia de "STOP" a "RUN". Para el programa de arranque no se dispone de valores actuales de la memoria imagen de proceso de las entradas, ni tampoco es posible activar salidas.

Una vez procesados por completo los OBs de arranque, se lee la memoria imagen de proceso de las entradas y se inicia el programa cíclico.

La ejecución del programa de usuario no está limitada en el tiempo. Por este motivo, no está activado el tiempo de vigilancia del ciclo. No es posible utilizar bloques de organización controlados por tiempo ni por alarmas.

Información de arranque

Un OB de arranque tiene la siguiente información de arranque:

Variable	Tipo de datos	Descripción
LostRetentive	BOOL	= 1, si se han perdido áreas de datos remanentes
LostRTC	BOOL	= 1, si se ha perdido el reloj en tiempo real

Consulte también

Bloques de organización (OB) (Página 2252)

Eventos y OBs (Página 1636)

Bloques de organización para la ejecución cíclica del programa

Introducción

Para poder comenzar a ejecutar el programa debe existir por lo menos un OB de ciclo en el proyecto. El sistema operativo llama este OB de ciclo una vez por ciclo, iniciando con ello la ejecución del programa de usuario. Es posible utilizar varios OBs de ciclo (números de OB >= 123). Si se utilizan varios OBs de ciclo, estos se llaman uno tras otro en el orden

correspondiente a su número de OB. El OB de ciclo con el menor número de OB se llama primero.

Los OBs de ciclo tienen la clase de prioridad 1, es decir, la prioridad más baja de todos los OBs. Por tanto, los eventos de cualquier otra clase de prioridad pueden interrumpir el programa cíclico.

Programar la ejecución cíclica del programa

La ejecución cíclica del programa se programa escribiendo el programa de usuario en los OBs de ciclo y los bloques llamados desde allí.

La primera ejecución cíclica del programa comienza tan pronto como haya finalizado sin errores el programa de arranque. El ciclo vuelve a comenzar cada vez que finaliza la ejecución cíclica del programa.

Pasos de la ejecución cíclica del programa

Un ciclo de ejecución del programa abarca los siguientes pasos:

1. El sistema operativo inicia el tiempo de vigilancia del ciclo.
2. El sistema operativo escribe los valores de la memoria imagen de proceso de las salidas en los módulos de salidas.
3. El sistema operativo lee el estado de las entradas en los módulos de entradas y actualiza la memoria imagen de proceso de las entradas.
4. El sistema operativo procesa el programa de usuario y ejecuta las operaciones correspondientes.
5. Al final de un ciclo, el sistema operativo puede ejecutar otras tareas pendientes, p. ej. cargar y borrar bloques, o bien llamar otros OBs de ciclo.
6. A continuación, la CPU regresa al comienzo del ciclo y reinicia la vigilancia del tiempo de ciclo.

Consulte también: Auto-Hotspot

Posibilidades de interrupción

Los siguientes eventos pueden interrumpir la ejecución cíclica del programa:

- Alarma
- Comando de STOP, disparado por
  - Intervención de la programadora
  - Instrucción "STP"
- Corte de alimentación
- Fallo de un dispositivo o error del programa

Información de arranque

- Ninguna
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
first_scan	BOOL	= TRUE en la primera llamada del OB con: • transición de STOP o PARADA a RUN • Tras recargar
remanence	BOOL	= TRUE si hay disponibles datos remanentes

Consulte también

Bloques de organización (OB) (Página 2252)

Eventos y OBs (Página 1636)

Bloques de organización para la ejecución del programa controlada por alarmas

Bloques de organización para alarmas horarias

Función

Los bloques de organización para alarmas horarias (números de OB >=123) pueden procesarse de la siguiente forma:

- Una sola vez en un instante especificado (con fecha y hora)
- De forma periódica, con un tiempo de arranque especificado y los siguientes intervalos:
  - Cada minuto
  - Cada hora
  - Cada día
  - Semanalmente
  - Mensualmente
  - Anualmente
  - A final de mes

Por este motivo, los OBs de alarma horaria se utilizan para ejecutar partes del programa de usuario de forma controlada por tiempo.

Estados de las alarmas horarias

La siguiente tabla contiene los distintos estados posibles de una alarma horaria, así como su significado.

Estado	Significado
Anulada	La alarma ya se ha ejecutado, o bien el evento de arranque de una alarma horaria que aún no se ha ejecutado se ha borrado con la instrucción avanzada CAN_TINT.
Ajustada	Se ha determinado el instante o instante de arranque de su ejecución.
Activada	Se ha establecido si la alarma debe ejecutarse una sola vez o de forma periódica, y se ha determinado el intervalo de tiempo en caso de ejecución periódica.

Reglas para alarmas horarias

Para utilizar alarmas horarias, se aplican las siguientes reglas:

- Una alarma horaria sólo puede ejecutarse si ha sido ajustada y activada, y si existe un bloque de organización correspondiente en el programa de usuario.
- Los tiempos de arranque de las alarmas horarias periódicas deben corresponder a una fecha real. Así pues, no es posible, por ejemplo, la repetición mensual de un bloque de organización cuya ejecución única tenga lugar el 31 de enero. En tal caso, el OB sólo se iniciaría en los meses que tuvieran 31 días.
- Una alarma horaria que se active durante el arranque llamando la instrucción avanzada ACT_TINT, no se ejecutará hasta que no haya finalizado el arranque.
- Cada vez que se arranque la CPU, las alarmas horarias ajustadas deberán activarse de nuevo.

Ajuste y activación de un OB de alarma horaria

Antes de que una alarma horaria pueda dispararse y de que el sistema operativo pueda ejecutar, con ello, el OB de alarma horaria asignado, es necesario ajustar y activar la alarma. Para ello existen las posibilidades siguientes:

Ajustar la alarma horaria	Activar la alarma horaria
Mediante configuración	Mediante configuración
Mediante configuración	Llamando la instrucción avanzada ACT_TINT
Llamando la instrucción avanzada SET_TINTL	Llamando la instrucción avanzada ACT_TINT

Nota

Si se configura una alarma horaria de modo que el OB correspondiente se ejecute una única vez, el tiempo de arranque no podrá encontrarse en el pasado (referido al reloj en tiempo real de la CPU).

Si se configura una alarma horaria de modo que el OB correspondiente se ejecute de forma periódica y el tiempo de arranque se encuentre, sin embargo, en el pasado, el OB de alarma horaria se ejecutará según la hora actual cuando se venza la siguiente hora.

Consultar el estado de una alarma horaria

Para consultar el estado de una alarma horaria, llame la instrucción avanzada QRY_TINT.

Anulación de una alarma horaria

Es posible anular alarmas horarias que no se hayan ejecutado todavía a través de la instrucción avanzada CAN_TINT.

Las alarmas horarias anuladas pueden volver a ajustarse con la instrucción avanzada SET_TINTL y volver a activarse con la instrucción avanzada ACT_TINT.

Condiciones que merman el funcionamiento de los OBs de alarma horaria

Puesto que una alarma horaria sólo se ejecuta en determinados intervalos, existen determinadas condiciones que pueden afectar a la función del OB correspondiente durante el procesamiento del programa. La siguiente tabla muestra algunas de estas condiciones y describe cómo afectan al procesamiento de un OB de alarma horaria.

Condición	Resultado
En el programa de usuario, se ejecutan la instrucción avanzada CAN_TINT.	El sistema operativo borra el evento de arranque (fecha y hora) de la alarma horaria. La alarma horaria deberá ajustarse y activarse de nuevo en caso de que deba llamarse nuevamente el correspondiente OB de alarma horaria.
Al sincronizar o corregir el reloj de sistema de la CPU, la hora se ha adelantado. Para ello, se ha saltado el tiempo de arranque de un OB de alarma horaria.	El sistema operativo ejecuta el OB de error de tiempo (OB 80) e introduce en la información de arranque el evento de arranque, el número y la prioridad del primer OB de alarma horaria que se ha saltado. Tras finalizar el OB 80, el sistema operativo procesa una única vez el OB de alarma horaria que se ha saltado.

Condición	Resultado
Al sincronizar o corregir el reloj de sistema de la CPU, la hora se ha retrasado. La hora corregida se encuentra por delante del tiempo de arranque de un OB de alarma horaria ya ejecutado.	El OB de alarma horaria se repite.
Un OB de alarma horaria se está procesando todavía cuando se produce el evento de arranque para su siguiente procesamiento.	El sistema operativo ejecuta el OB de error de tiempo (OB 80). Tras su procesamiento y el procesamiento posterior del OB de alarma horaria actual, se procesa a posteriori el OB solicitado.

Información de arranque

Un OB de alarma horaria tiene la siguiente información de arranque:

Variable	Tipo de datos	Descripción
CaughtUp	BOOL	= 1, si se ejecuta el OB a posteriori por haberse adelantado la hora
SecondTime	BOOL	= 1, si el OB se llama por segunda vez por haberse retrasado la hora (dicho de otro modo: si la hora planeada para el procesamiento actual del OB se encuentra antes o en el mismo momento de la hora planeada para el procesamiento anterior del OB). Nota: SecondTime se ajusta una única vez.

Bloque de organización para alarmas de estado

Descripción

El sistema operativo de la CPU S7-1200 llama el OB de alarma de estado cuando recibe una alarma de estado de un maestro DP o un controlador IO. Esto puede suceder cuando un módulo de un esclavo cambia de estado operativo, p. ej. de "RUN" a "STOP". Puede obtenerse más información sobre los eventos que originan una alarma de estado en la documentación del fabricante del esclavo o dispositivo.

Estructura de la información de arranque

El OB de alarma de estado tiene la siguiente información de arranque:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Dirección de hardware del componente causante de la alarma
Slot	UINT	Número de slot del componente causante de la alarma
Especificador	WORD	Especificador de alarma del telegrama de alarma

Consulte también

Eventos y OBs (Página 1636)

Bloque de organización para alarmas de actualización

Descripción

El sistema operativo de la CPU S7-1200 llama el OB de alarma de actualización cuando recibe una alarma de actualización de un maestro DP o un controlador IO. Esto puede suceder cuando se han modificado parámetros en un slot de un esclavo o dispositivo. Puede obtenerse más información sobre los eventos que originan una alarma de actualización en la documentación del fabricante del esclavo o dispositivo.

Estructura de la información de arranque

El OB de alarma de actualización tiene la siguiente información de arranque:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Dirección de hardware del componente causante de la alarma
Slot	UINT	Número de slot del componente causante de la alarma
Especificador	WORD	Especificador de alarma del telegrama de alarma

Consulte también

Eventos y OBs (Página 1636)

Bloque de organización para alarmas de fabricante o perfil

Descripción

El sistema operativo de la CPU S7-1200 llama el OB57 cuando recibe una alarma de fabricante o perfil de un maestro DP o un controlador IO. Puede obtenerse más información sobre los eventos que originan una alarma de este tipo en la documentación del fabricante del esclavo o dispositivo.

Estructura de la información de arranque

El OB de alarma de fabricante o perfil tiene la siguiente información de arranque:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Dirección de hardware del componente causante de la alarma
Slot	UINT	Número de slot del componente causante de la alarma
Especificador	WORD	Especificador de alarma del telegrama de alarma

Consulte también

Eventos y OBs (Página 1636)

Bloques de organización para alarmas de retardo

Descripción

El sistema operativo inicia un OB de alarma de retardo al cabo de un tiempo de retardo parametrizable. El tiempo de retardo comienza a transcurrir tras llamar la instrucción SRT_DINT.

En el programa pueden utilizarse como máximo cuatro OBs de alarma de retardo u OBs de alarma cíclica (números de OB >= 123). Si ya se están utilizando p. ej. dos OBs de alarma cíclica, solo se podrán insertar como máximo dos OBs de alarma de retardo en el programa.

El procesamiento de una alarma de retardo que no se haya iniciado aún se puede impedir con la instrucción CAN_DINT.

Funcionamiento de los OBs de alarma de retardo

Una vez transcurrido el tiempo de retardo transferido a la instrucción junto con un número de OB y una identificación, el sistema operativo inicia el OB asociado.

Si desea utilizar un OB de alarma de retardo, utilice las tareas siguientes:

- Llamar la instrucción SRT_DINT.
- Cargar el OB de alarma de retardo en la CPU como parte del programa de usuario.

La alarma de retardo se mide con una precisión de 1 ms. Un tiempo de retardo puede reiniciarse inmediatamente una vez transcurrido.

Los OBs de alarma de retardo se ejecutan únicamente si la CPU se encuentra en estado operativo "RUN". Un rearranque completo (en caliente) borra todos los eventos de arranque de los OBs de alarma de retardo.

El sistema operativo llama el OB de alarma de retardo cuando se presenta uno de los eventos siguientes:

- Cuando el sistema operativo intenta iniciar un OB que no está cargado y cuyo número se ha indicado al llamar la instrucción SRT_DINT.
- Cuando se presenta el siguiente evento de arranque de una alarma de retardo antes de finalizar el procesamiento del OB de alarma de retardo asociado.

Las alarmas de retardo se pueden inhibir y habilitar con las instrucciones DIS_AIRT y EN_AIRT, respectivamente.

Nota

Si, tras la ejecución de SRT_DINT, se inhibe una alarma con DIS_AIRT, esta alarma se procesará tan solo tras habilitarla con EN_AIRT. El tiempo de retardo se prolonga de la forma correspondiente.

Información de arranque

- Ninguna
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
sign	WORD	Identificador de usuario: parámetro de entrada SIGN de la llamada de la instrucción "SRT_DINT"

Consulte también

SRT_DINT: Iniciar alarma de retardo (Página 4899)

CAN_DINT: Anular alarma de retardo (Página 4901)

Bloques de organización (OB) (Página 2252)

Eventos y OBs (Página 1636)

Bloques de organización para alarmas cíclicas

Descripción

Los OBs de alarma cíclica sirven para iniciar programas en intervalos periódicos, independientemente de la ejecución cíclica del programa. Los tiempos de arranque de un OB de alarma cíclica se indican mediante el periodo y el desfase.

El periodo define el intervalo en el que se arranca el OB de alarma cíclica y es un múltiplo entero del ciclo base de 1 ms. El desfase es el tiempo de desfase del tiempo de arranque con respecto al ciclo base. Si se utilizan varios OBs de alarma cíclica, este desfase puede utilizarse para impedir un tiempo de arranque simultáneo, si los periodos de los OBs de alarma cíclica tienen un múltiplo común.

Como periodo se puede predeterminar un tiempo comprendido entre 1 ms y 60000 ms.

En el programa pueden utilizarse como máximo cuatro OBs de alarma cíclica u OBs de alarma de retardo (números de OB >= 123). Si ya se están utilizando p. ej. dos OBs de alarma de retardo, se podrán insertar como máximo dos OBs de alarma cíclica en el programa.

Nota

El tiempo de ejecución de todo OB de alarma cíclica debe ser muy inferior a su periodo. Si un OB de alarma cíclica no ha finalizado todavía, pero se tiene que ejecutar nuevamente porque ha finalizado el periodo, se arrancará el OB de error de tiempo. A continuación, se reactiva o se rechaza la alarma cíclica que ha causado el error.

Ejemplo de utilización del desfase

En el programa se han insertado dos OBs de alarma cíclica:

- OB1 de alarma cíclica
- OB2 de alarma cíclica

Para el OB1 de alarma cíclica se ha ajustado un periodo de 20 ms y, para el OB2 de alarma cíclica, de 100 ms. Una vez transcurrido el periodo de 100 ms, el OB1 de alarma cíclica alcanzará por quinta vez su tiempo de arranque, mientras que el OB2 de alarma cíclica lo hará por primera vez. No obstante, para procesar ambos OBs de alarma cíclica de forma desfasada, es preciso definir un desfase para uno de ellos.

Información de arranque

- Ninguna
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
first_scan	BOOL	= TRUE en la primera llamada del OB • En la transición de STOP o PARADA a RUN • Tras recargar
event_count	INT	Número de los eventos de arranque perdidos desde el último arranque del OB

Consulte también

Bloques de organización (OB) (Página 2252)

Parametrizar OBs de alarma cíclica (Página 1692)

Eventos y OBs (Página 1636)

Bloques de organización para alarmas de proceso

Descripción

Los OB de alarma de proceso pueden utilizarse para reaccionar a determinados eventos. A un evento que dispara una alarma solo se puede asociar un único OB de alarma de proceso. En cambio, a un OB de alarma de proceso pueden asociarse varios eventos.

Los contadores rápidos y canales de entrada pueden disparar alarmas de proceso. Es preciso parametrizar las siguientes propiedades para todo contador rápido y canal de entrada que deba disparar una alarma de proceso:

- El evento de proceso que debe disparar la alarma de proceso (p. ej. cambio del sentido de contaje de un contador rápido)
- El número del OB de alarma de proceso asociado a este evento de proceso

En el programa pueden utilizarse como máximo 50 OB de alarma de proceso (números de OB >= 123) independientes unos de otros.

Funcionamiento de un OB de alarma de proceso

Tras disparar una alarma de proceso, el sistema operativo identifica el canal de entrada o el contador rápido y determina el OB de alarma de proceso asociado.

Si no está activo ningún otro OB de alarma, se llamará el OB de alarma de proceso que se ha determinado. Si ya se está ejecutando otro OB de alarma, la alarma de proceso se colocará en la cola de espera de su clase de prioridad. La alarma de proceso se acusa tras finalizar la ejecución del OB de alarma de proceso asociado.

Si desde que se identifica hasta que se acusa una alarma de proceso se presenta nuevamente un evento de proceso en el mismo módulo, se aplicará lo siguiente:

- Si el evento se presenta en el canal que ha disparado la alarma de proceso actual, no se disparará ninguna otra alarma de proceso. Una alarma de proceso adicional puede dispararse tan solo tras haberse acusado la alarma de proceso actual.
- Si el evento se presenta en un canal diferente, se disparará una alarma de proceso.

Los OB de alarma de proceso solo se llaman en el estado operativo "RUN" de la CPU.

Información de arranque

- Ninguna
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
Laddr	HW_IO	Identificador de hardware del módulo que dispara la alarma de proceso
USI	WORD	Identificador para ampliaciones futuras (irrelevante para el usuario)
IChannel	USINT	Número del canal que dispara la alarma de proceso
EventType	BYTE	Identificador de tipo del evento que dispara la alarma (p. ej. flanco ascendente) Este identificador se indica en la descripción del módulo correspondiente.

Consulte también

Bloques de organización (OB) (Página 2252)

Parametrizar OBs de alarma de proceso (Página 1693)

Eventos y OBs (Página 1636)

Bloque de organización de error de tiempo

Descripción

El sistema operativo llama el OB de error de tiempo (OB 80) cuando se presenta uno de los eventos siguientes:

- El programa cíclico excede el tiempo de vigilancia del ciclo.
- El OB llamado se está ejecutando todavía (esto es posible en los OBs de alarma de retardo y de alarma cíclica).
- Se ha perdido una alarma horaria porque la hora se ha adelantado más de 20 segundos.
- Una alarma horaria se ha perdido en estado STOP.
- Se ha desbordado una cola de espera de OBs de alarma.
- Una alarma se ha perdido por sobrecarga de alarmas.

Si no se ha programado ningún OB de error de tiempo, la CPU S7-1200 se comporta del modo siguiente:

- CPUs con versión de firmware V1.0: La CPU permanece en RUN.
- CPUs con versión de firmware V2.0 o superior:
  - Al exceder el tiempo de vigilancia del ciclo, la CPU pasa a STOP.
  - Con todos los demás eventos de inicio del OB de error de tiempo, la CPU permanece en RUN.

En las CPUs con versión de firmware V1.0, un segundo rebase del tiempo de vigilancia del ciclo no provoca la llamada de un OB, sino un STOP de la CPU. El segundo rebase se puede impedir reiniciando la vigilancia del ciclo de la CPU con la instrucción RE_TRIGR.

En el programa solo se puede utilizar un OB de error de tiempo.

Información de arranque

El OB de error de tiempo tiene la siguiente información de arranque:

Variable	Tipo de datos	Descripción
fault_id	BYTE	• 0x01: Tiempo de vigilancia del ciclo excedido • 0x02: El OB llamado todavía se está ejecutando • 0x05: Alarma horaria expirada debido a salto de hora • 0x06: Alarma horaria expirada al reanudar en RUN • 0x07: Desbordamiento de la cola de espera • 0x09: pérdida de alarmas por sobrecarga de alarmas
csg_OBnr	OB_ANY	Número del OB procesado en el momento en el que ocurrió el error
csg_prio	UINT	Prioridad del OB procesado en el momento en el que ocurrió el error

Consulte también

Bloques de organización (OB) (Página 2252)

Eventos y OBs (Página 1636)

Bloque de organización para alarmas de diagnóstico

Descripción

La alarma de diagnóstico se puede habilitar para los módulos aptos para diagnóstico, de manera que el módulo pueda detectar cambios del estado de la periferia. Así, el módulo dispara una alarma de diagnóstico en los casos siguientes:

- Hay un fallo (evento entrante)
- El fallo se ha solucionado (evento saliente)

Si no está activo ningún otro OB de alarma, se llama el OB de alarma de diagnóstico (OB 82). Si ya se está ejecutando otro OB de alarma, la alarma de diagnóstico se colocará en la cola de espera de su clase de prioridad.

En el programa se puede utilizar un solo OB de alarma de diagnóstico.

Información de arranque

El OB de alarma de diagnóstico tiene la siguiente información de arranque:

Variable	Tipo de datos	Descripción
IO_state	WORD	Contiene el estado de la periferia del módulo apto para diagnóstico.
laddr	HW_ANY	Identificación HW
channel	UINT	Número de canal
multi_error	BOOL	= 1, si ha ocurrido más de un error

Variable IO_state

La tabla siguiente muestra los posibles estados de la periferia que puede contener la variable IO_state:

IO_state	Descripción
Bit 0	Configuración correcta: = 1, si la configuración es correcta = 0, si la configuración ya no es correcta
Bit 4	Fallo: = 1, si hay un fallo, p. ej. rotura de hilo = 0, si se ha solucionado el fallo

IO_state	Descripción
Bit 5	Configuración incorrecta: = 1, si la configuración no es correcta = 0, si la configuración vuelve a ser correcta
Bit 6	No se puede acceder a la periferia: = 1, si ha ocurrido un error de acceso a la periferia En este caso, laddr contiene la ID de hardware de la periferia con el error de acceso. = 0, si puede accederse nuevamente a la periferia

Consulte también

Bloques de organización (OB) (Página 2252)

Eventos y OBs (Página 1636)

Bloque de organización de cambio de módulo

Descripción

El sistema operativo de la CPU S7-1200 llama el OB de cambio de módulo (OB 83) cuando se extrae o inserta un módulo o submódulo de la periferia descentralizada configurado y no desactivado (PROFIBUS, PROFINET, AS-i).

Nota

La extracción o inserción de un módulo central provoca un STOP de la CPU.

Información de arranque

El OB de cambio de módulo tiene la siguiente información de arranque:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del módulo o submódulo afectado
Event_Class	BYTE	• B#16#38: (sub)módulo insertado • B#16#39: (sub)módulo extraído o no responde
Fault_ID	BYTE	Código de error (valores posibles: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58)

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB de cambio de módulo.

ev_class (B#16# ...)	fault_id (B#16# ...)	Significado
39	51	Módulo extraído
39	54	Submódulo extraído
38	54	Submódulo insertado equivalente al submódulo parametrizado

ev_class (B#16# ...)	fault_id (B#16# ...)	Significado
38	55	Submódulo insertado que no equivale al submódulo parametrizado
38	56	Submódulo insertado, pero error en la parametrización
38	57	Submódulo o módulo insertado, pero con fallo o mantenimiento
38	58	Error de acceso a submódulo solucionado

Consulte también

Eventos y OBs (Página 1636)

Bloque de organización de fallo de rack

Descripción

El sistema operativo de la CPU S7-1200 llama el OB86 en los siguientes casos:

- Se detecta el fallo de un sistema maestro DP o un sistema PROFINET IO (tanto con evento entrante como con evento saliente).
- Se detecta el fallo de un esclavo DP o un dispositivo IO (tanto con evento entrante como con evento saliente).
- Se detecta el fallo de una parte de los submódulos de un I-Device PROFINET.

Estructura de la información de arranque

El OB de fallo de rack tiene la siguiente información de arranque:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del objeto de hardware defectuoso
Event_Class	BYTE	• B#16#32: activación de un esclavo DP o un dispositivo IO • B#16#33: desactivación de un esclavo DP o un dispositivo IO • B#16#38: evento saliente • B#16#39: evento entrante
Fault_ID	BYTE	Código de error (valores posibles: B#16#C3, B#16#C4, B#16#C5, B#16#C6, B#16#C7, B#16#C8, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB86.

Ev_class B#16# ...	Fault_id B#16# ...	Significado
39	C3	Fallo de un sistema maestro DP
39/38	C4	Fallo/retorno de un esclavo DP

Ev_class B#16# ...	Fault_id B#16# ...	Significado
38	C5	Retorno de un esclavo DP, aunque defectuoso
38	C6	Retorno del aparato de ampliación, pero error en la parametrización del módulo
38	C7	Retorno de un esclavo DP, pero hay un fallo en la parametrización del módulo
38	C8	Retorno de un esclavo DP, pero con diferencias entre la configuración teórica y la configuración real
32/33	C9	Activación/desactivación de un esclavo DP con la instrucción "D_ACT_DP"
39	CA	Fallo de un sistema PROFINET IO
39/38	CB	Fallo/retorno de un dispositivo PROFINET IO
38	CC	Retorno de un dispositivo PROFINET IO con fallo o mantenimiento
38	CD	Retorno de un dispositivo PROFINET IO, pero la configuración teórica difiere de la configuración real
38	CE	Retorno de un dispositivo PROFINET IO, pero error en la parametrización del módulo
32/33	CF	Activación/desactivación de un dispositivo IO con la instrucción "D_ACT_DP"
39/38	F8	Fallo/retorno de una parte de los submódulos de un I-Device PROFINET
38	F9	Retorno de una parte de los submódulos de un I-Device PROFINET con diferencias en la configuración de dispositivos

Consulte también

Eventos y OBs (Página 1636)

Parámetros de bloques de organización

Principios básicos de los parámetros de bloque

Introducción

Algunos bloques de organización (OBs) tienen propiedades que permiten controlar su comportamiento o asignación a determinados eventos. Parametrizando estas propiedades se puede influir en ellas.

Resumen

Es posible parametrizar las propiedades de los siguientes bloques de organización:

- OBs de alarma horaria
- OBs de alarma cíclica
- OBs de alarma de proceso

Consulte también

Parametrizar OBs de alarma de proceso (Página 1693)

Parametrizar OBs de alarma cíclica (Página 1692)

Parametrizar un OB de alarma horaria

Procedimiento para configurar los parámetros

Para ajustar los parámetros de un OB de alarma horaria, proceda del siguiente modo:

1. Abra el cuadro de diálogo "Propiedades" del OB de alarma horaria en cuestión.
2. Haga clic en el grupo "Alarma horaria" de la navegación local.

Vista general de los parámetros ajustables

Se pueden configurar los parámetros siguientes:

- Ejecución
- Fecha de inicio y hora
- Botones de opción "Hora local" y "Hora del sistema"

Parámetro "Ejecución"

La lista desplegable "Ejecución" permite indicar con qué periodicidad debe ejecutarse el OB de alarma horaria. Los intervalos se basan en los ajustes de "Fecha de inicio" y "Hora".

Están disponibles los siguientes valores de "Ejecución":

- nunca
- una sola vez
- una vez por minuto
- una vez por hora
- una vez al día
- una vez por semana
- una vez al mes
- una vez al año
- a finales de mes

Nota

Con el valor "a finales de mes", el valor indicado bajo "Fecha de inicio" es irrelevante.

Parámetros "Fecha de inicio" y "Hora"

Permiten establecer en qué momento debe ejecutarse el OB de alarma horaria por primera vez.

Ejemplo: fecha de inicio = 05.07.2013, hora =11:16

En función del valor del parámetro "Ejecución", la CPU genera otras alarmas horarias periódicas. La hora de inicio se refiere, según la configuración, bien a la hora local, bien al tiempo universal coordinado UTC.

Nota

Si se configura el parámetro "Ejecución" como "mensual", los días 29, 30 y 31 no se pueden seleccionar como fecha de inicio. Si el OB de alarma horaria debe iniciarse a finales de mes, seleccione en su lugar "a finales de mes" en el parámetro "Ejecución".

"Hora local" u "Hora del sistema"

Permite definir a qué hora hace referencia la hora de inicio del OB de alarma horaria:

- "Hora local": la hora de inicio se refiere a la zona horaria configurada en la CPU.
- "Hora del sistema": la hora de inicio se refiere al tiempo universal coordinado UTC (Universal Time Coordinated).

Parametrizar OBs de alarma cíclica

Introducción

Los OB de alarma cíclica permiten iniciar programas en intervalos periódicos. Para ello es preciso especificar un periodo y un desfase para cada OB de alarma cíclica utilizado.

En el programa pueden utilizarse como máximo cuatro OB de alarma cíclica u OB de alarma de retardo (números de OB >= 200). Si ya se están utilizando p. ej. dos OB de alarma de retardo, se podrán insertar como máximo dos OB de alarma cíclica en el programa.

Nota

Si parametriza varios OB de alarma cíclica, es preciso asignar un tiempo de ciclo o un desfase diferente a cada OB de alarma cíclica, con objeto de impedir su ejecución simultánea y/o una cola de espera. Al crear un OB de alarma cíclica se especifican el tiempo de ciclo 100 y el desfase 0 como valor de arraque.

Procedimiento

Para especificar un periodo y un desfase para un OB de alarma cíclica, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del árbol del proyecto.
2. Haga clic con el botón derecho del ratón en un OB de alarma cíclica existente.

3. Elija el comando "Propiedades" del menú contextual.
   Se abre el diálogo "<Nombre del OB de alarma cíclica>".
4. Haga clic en el grupo "Alarma cíclica" de la navegación local.
   Se visualizan los campos de entrada para el periodo y el desfase.
5. Introduzca el periodo y el desfase.
6. Confirme las entradas realizadas con "Aceptar".

Consulte también

Crear bloques de organización (Página 2596)

Principios básicos de los parámetros de bloque (Página 1690)

Bloques de organización para alarmas cíclicas (Página 1683)

Parametrizar OBs de alarma de proceso

Introducción

Para todo canal de entrada y todo contador rápido que debe disparar una alarma de proceso es preciso activar el evento correspondiente y asignar los parámetros siguientes:

- Nombre del evento
- Número del OB de alarma de proceso asociado a este evento de proceso

Las alarmas de proceso se parametrizan en las propiedades del dispositivo correspondiente. Como máximo pueden parametrizarse 50 OBs de alarma de proceso.

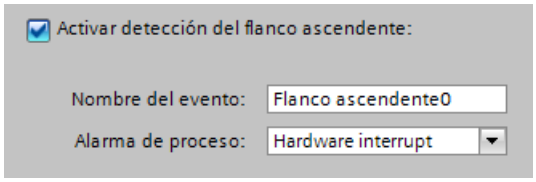
El OB de alarma de proceso que se desea parametrizar puede crearse antes o durante la activación de un evento.

Procedimiento

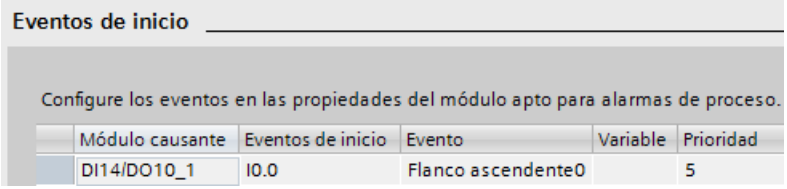
Para configurar un evento de alarma de proceso, proceda del siguiente modo:

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   El editor de hardware y redes se abre en la vista de red.
2. Cambie a la vista de dispositivos.
3. Si la ventana de inspección está cerrada en la vista de dispositivos, active la casilla de verificación "Ventana de inspección" del menú "Ver".
   Se abre la ventana de inspección.
4. Haga clic en la ficha "Propiedades".
5. En la vista de dispositivos, seleccione el módulo para el que desea parametrizar una alarma de proceso.

6. Active el evento que deba disparar una alarma de proceso, p. ej. un flanco ascendente.



7. Introduzca un nombre de evento.
8. Seleccione un OB de alarma de proceso de la lista desplegable "Alarma de proceso", o cree un OB de alarma de proceso nuevo. Si todavía no ha creado ningún OB de alarma de proceso, haga clic en el botón "Agregar nuevo bloque" en la lista desplegable.
   La información de arranque del OB de alarma de proceso correspondiente se actualiza con todos los datos sobre el evento que dispara la alarma.



9. Si desea parametrizar más alarmas de proceso, repita los pasos 5 a 8.

Para el evento, identificado por el nombre de evento unívoco, se crea automáticamente una constante de sistema del tipo de datos Event_Hwlnt. Las constantes de sistema se muestran en la tabla de variables estándar.

Consulte también

Crear bloques de organización (Página 2596)

Principios básicos de los parámetros de bloque (Página 1690)

Bloques de organización para alarmas de proceso (Página 1684)

Eventos y OBs (Página 1636)

Denominación simbólica y numérica de las instrucciones

Descripción

Las instrucciones de la Task Card se componen de funciones (FC), bloques de función (FB), funciones de sistema (SFC) y bloques de función de sistema (SFB) que están identificados internamente por medio de números.

Las tablas siguientes muestran las correspondencias entre la denominación numérica y la simbólica.

Bloques de función (FBs)

Denominación numérica	Denominación simbólica
FB 105	TC_CONFIG
FB 110	Port_Config
FB 111	Send_Config
FB 112	Receive_Config
FB 113	Send_P2P
FB 114	Receive_P2P
FB 115	Receive_Reset
FB 116	Signal_Get
FB 117	Get_Features
FB 118	Set_Features
FB 163	TC_SEND
FB 164	TC_RECV
FB 165	TC_CON
FB 166	TC_DISCON
FB 804	SET_TIMEZONE
FB 1030	TSEND_C
FB 1031	TRCV_S
FB 1071	USS_DRIVE
FB 1080	MB_COMM_LOAD
FB 1081	MB_MASTER
FB 1082	MB_SLAVE
FB 1084	MB_CLIENT
FB 1085	MB_SERVER
FB 1100	MB_Halt
FB 1101	MC_Home
FB 1102	MC_MoveAbsolute
FB 1103	MC_MoveJog
FB 1104	MC_MoveRelative
FB 1105	MC_MoveVelocity
FB 1107	MC_Power
FB 1108	MC_Reset
FB 1110	MC_MoveInterrupt
FB 1111	MC_ChangeDynamik
FB 1112	MC_CommandTable
FB 1113	MC_MoveLinearAbs_2D
FB 1114	MC_MoveLinearRel_2D
FB 1115	MC_MoveCircular_2D
FB 1130	PID_Compact
FB 1134	PID_3Step
FB 1140	HSC

Denominación numérica	Denominación simbólica
FB 2040	RecipeCreate
FB 2041	RecipeOpen
FB 2042	RecipeRead
FB 2043	RecipeWrite
FB 2044	RecipeAppend
FB 2045	RecipeClose

Funciones (FCs)

Denominación numérica	Denominación simbólica
FC 2 (1)	CONCAT
FC 4 (1)	DELETE
FC 11 (1)	FIND
FC 17 (1)	INSERT
FC 20 (1)	LEFT
FC 21 (1)	LEN
FC 22 (1)	LIMIT
FC 25 (1)	MAX
FC 26 (1)	MID
FC 27 (1)	MIN
FC 31 (1)	REPLACE
FC 32 (1)	RIGHT
FC 36 (1)	ENCO
FC 36 (1)	SEL
FC 37	DECO
FC 800	LED
FC 801	IM_DATA
FC 802	DeviceStates
FC 803	ModuleStates
FC 1070	USS_PORT
FC 1072	USS_RPM
FC 1073	USS_WPM
(1) Instrucción MC7+	

Tipos de datos de sistema (SDTs)

Denominación numérica	Denominación simbólica
SDT 99	WWW_CDB
SDT 513	CONDITIONS
SDT 581	Send_Conditions
SDT 582	Receive_Conditions

Bloques de función de sistema (SFBs)

Denominación numérica	Denominación simbólica
SFB 0 (1)	CTU
SFB 1 (1)	CTD
SFB 2 (1)	CTUD
SFB 3 (1)	TP
SFB 4 (1)	TON
SFB 5 (1)	TOF
SFB 27	START_OB
SFB 52	RDREC
SFB 53	WRREC
SFB 54	RALRM
SFB 105	T_CONFIG
SFB 106	TDIAG
SFB 107	TRESET
SFB 110	PORT_CFG
SFB 111	SEND_CFG
SFB 112	RCV_CFG
SFB 113	SEND_PTP
SFB 114	RCV_PTP
SFB 115	SGN_GET
SFB 116	SGN_SET
SFB 117	RCV_RST
SFB 120	CTRL_HSC
SFB 122	CTRL_PWM
SFB 140	DataLogCreate
SFB 141	DataLogOpen
SFB 142	DateLogWrite
SFB 143	DataLogClear
SFB 144	DataLogClose
SFB 145	DataLogDelete
SFB 146	DataLogNewFile

Funciones de sistema (SFCs)

Denominación numérica	Denominación simbólica
SFC 7	DP_PRAL
SFC 11	DPSYC_FR
SFC 13	DPNRM_DG
SFC 14	DPRD_DAT
SFC 16	RD_OBINF
SFC 23	DEL_DB

Denominación numérica	Denominación simbólica
SFC 28	SET_TINT
SFC 29	CAN_TINT
SFC 30	ACT_TINT
SFC 31	QRY_TINT
SFC 32	SRT_DINT
SFC 33	CAN_DINT
SFC 34	QRY_DINT
SFC 41	DIS_AIRT
SFC 42	EN_AIRT
SFC 43	RE_TRIGR
SFC 45	DE_ACT
SFC 46	STP
SFC 82	CREA_DBL
SFC 83	READ_DBL
SFC 84	WRIT_DBL
SFC 86	CREATE_DB
SFC 89	RST_EVOV
SFC 99	WWW
SFC 101	RTM
SFC 117	GET_DIAG
SFC 124	ATTR_DB
SFC 140	IO2MOD
SFC 143	RD_ADDR
SFC 154	RD_LOC_T
SFC 154	DPWR_DAT
SFC 161	WR_LOC_T
SFC 180	ID2LOG
SFC 181	LOG2ID
SFC 182	ID2GEO
SFC 190	SET_CINT
SFC 191	QRY_CINT
SFC 192	ATTACH
SFC 193	DETACH
MC7+ Anweisung	GET_ERROR
MC7+ Anweisung	GET_ERR_ID

Información importante sobre las versiones de firmware de la CPU y las versiones de STEP 7

Las CPU y el software de ingeniería para la configuración de CPU se encuentran en permanente desarrollo para mejorar sus prestaciones y su seguridad. De esta forma surgen nuevas versiones que presentan algunas particularidades en combinación con los componentes. Las siguientes secciones describen las particularidades de las CPU S7-1200 con versión de firmware V4 en comparación con las versiones de firmware V1 a V3.

Para comparar con detalle el volumen de funciones, lea el manual de sistema S7-1200 (descripción de instrucciones nuevas, bloques de organización nuevos y opciones de configuración avanzadas).

Software de ingeniería necesario

Las CPU S7-1200 V4 son configurables con STEP 7 a partir de V13.

Compatibilidad entre los contenidos de la Memory Card y la versión de firmware de la CPU

Las Memory Card (tarjeta de transferencia o tarjeta de programa) con configuración y programa para una CPU S7-1200 V1, V2 o V3 no funcionan en una CPU S7-1200 V4.
Las Memory Card con configuración y programas para una CPU S7-1200 V4 no funcionan en una CPU S7-1200 V1, V2 o V3.

La configuración de una CPU S7-1200 de V1 a V3 debe cambiarse a una configuración de CPU S7-1200 V4 (sustitución de dispositivos) y, seguidamente, cargarse en la CPU. Para las CPU S7-1200 V1 y V2 se requiere una sustitución gradual (véase abajo).

Si inserta la Memory Card en una CPU con una versión de firmware incompatible, la CPU no arranca. Si inserta una Memory Card para una CPU V1, V2 o V3 en una CPU S7-1200 V4, esta CPU indica un error de versión.

Paso a online y carga

Si ha configurado con STEP 7 una CPU S7-1200 con la versión de firmware V1, V2 o V3, es necesario que la CPU que quiere cargar o a la que quiere pasar online tenga también esta versión de firmware. Con una CPU S7-1200 V1, V2 o V3 configurada no se puede pasar online a una CPU S7-1200 V4.

Inversamente, con una CPU S7-1200 V4 configurada no se puede pasar online a una CPU S7-1200 V1, V2 o V3 ni se puede cargar esta CPU.

Sustitución de la CPU existente

Existe la posibilidad de sustituir una CPU S7-1200 V1, V2 o V3 por una CPU con una versión de firmware V4 o superior. En caso de tener una CPU S7-1200 V1 o V2, primero deberá sustituirse por una CPU S7-1200 V3: no es posible cambiarla directamente por una CPU S7-1200 V4.

1. CPU S7-1200 V1 (V2) > CPU S7-1200 V3
2. CPU S7-1200 V3 > CPU S7-1200 V4

Una sustitución de dispositivo puede deshacerse (comando "Deshacer" del menú "Edición") mientras no se haya cargado la configuración. Una vez cargada en la CPU la configuración de la nueva versión de firmware de la CPU ("configuración V4") ya no es posible regresar a una versión V3.

Por este motivo es recomendable guardar el proyecto existente, p. ej. con una configuración V3 como fichero de proyecto, para poder acceder a él posteriormente.

Particularidades de la sustitución de dispositivos (V3 > V4):

- El comportamiento de interrupción de los OB de alarma no cambia; están parametrizados para que no puedan interrumpirse. Este también es el comportamiento predeterminado en las CPU S7-1200 V1 a V3. En las CPU S7-1200 V4, el comportamiento de interrupción de los OB de alarma puede parametrizarse. Si se arrastra una CPU S7-1200 V4 directamente desde el catálogo de hardware hasta la vista de redes, esta opción está activada (los OB de alarma pueden interrumpirse).
- El comportamiento para el acceso PUT/GET de interlocutores remotos no cambia, el acceso está permitido. Este también es el comportamiento predeterminado en las CPU S7-1200 V1 a V3. En las CPU S7-1200 V4, el acceso vía comunicación PUT/GET por parte de interlocutores remotos es parametrizable (área "Protección" de los parámetros de la CPU). Si se arrastra una CPU S7-1200 V4 directamente desde el catálogo de hardware hasta la vista de redes, el acceso no está permitido y debe habilitarse explícitamente.
- El enunciado de los niveles de protección es distinto, pero el efecto de los ajustes es el mismo. Se ha agregado el nivel de acceso "Sin acceso (protección completa)".
- Los ajustes del servidor web para activar este último y los ajustes HTTP/HTTPS se aplican. Además existe la posibilidad de crear usuarios y asignarles derechos específicos (área Servidor web > Administración de usuarios de los parámetros de CPU). Sin la parametrización de usuarios, un usuario del servidor web solo tiene acceso a páginas web estándar. Las CPU S7-1200 V4 ya no soportan el usuario "admin" ni su contraseña.

Comunicación con dispositivos HMI

Si conecta un dispositivo HMI a una CPU S7-1200 V4, procure que la versión del software runtime del dispositivo HMI sea adecuada.

En caso necesario, debe transferir la última versión de runtime del HMI mediante el software de ingeniería WinCC.

Para que la comunicación CPU-HMI esté operativa durante el funcionamiento, la configuración HMI debe compilarse de nuevo y volver a cargarse en el dispositivo HMI.

Consulte también

Información importante sobre las Memory Cards (Página 1628)

8.1.5.2 Funcionamiento de las CPU S7-300/400

Estados operativos

Principios básicos de los estados operativos de CPUs S7

Los estados operativos describen el comportamiento de la CPU. Los estados operativos se pueden consultar y evaluar en todo momento para la programación del arranque, la comprobación del controlador y el diagnóstico de errores.

Las CPU S7 de las distintas familias, como, p. ej., S7-300, S7-400, ET 200S, ET 200Pro y también WinLC (controlador basado en PC) o Embedded Controller, tienen los estados operativos siguientes:

- STOP
- ARRANQUE
- RUN
- PARADA

En estos estados operativos, la CPU puede comunicarse a través de sus interfaces (p. ej., a través de una interfaz MPI).

Estado operativo "STOP"

En el estado operativo "STOP", la CPU comprueba si están presentes todos los módulos configurados o utilizados mediante direccionamiento estándar. En función de esto, la CPU pone la periferia en un estado básico predeterminado. En el estado operativo "STOP", no se procesa el programa de usuario.

Consulte también: Estado operativo "STOP" (Página 1703)

Estado operativo "ARRANQUE"

En el estado operativo "ARRANQUE" se distingue entre los tipos de arranque "Rearranque completo (en caliente)", "Rearranque normal" y "Arranque en frío".

Consulte también: Auto-Hotspot

Estado operativo "RUN"

En el estado operativo "RUN", la CPU procesa el programa de usuario, actualiza las entradas y salidas y procesa alarmas y mensajes de error.

Consulte también: Estado operativo "RUN" (Página 1712)

Estado operativo "PARADA"

En el estado operativo "PARADA" se interrumpe el procesamiento del programa de usuario. El estado operativo "PARADA" sólo se consigue al realizar pruebas con la programadora.

Consulte también: Estado operativo "PARADA" (Página 1712)

Otros estados operativos

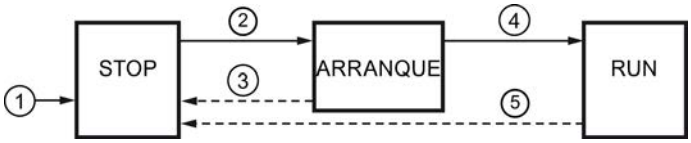
Si la CPU no está lista para funcionar, se encuentra en uno de estos dos estados operativos:

- Sin tensión, es decir, la tensión de red está desconectada.
- Avería, es decir, se ha producido un fallo que no se puede solventar.
  Compruebe si la CPU realmente está averiada. Para ello, póngala en "STOP" y apague y encienda el interruptor de alimentación. Para analizar el error, lea el búfer de diagnóstico cuando arranque la CPU. Si la CPU no arranca, cámbiela.

Transiciones de estado operativo

Resumen

La figura siguiente muestra los estados operativos y las transiciones de CPUs S7-300:



La tabla siguiente muestra las condiciones en las que cambian los estados operativos:

N.º	Transición de estado operativo	Condiciones
①	STOP	Después de conectar la tensión de alimentación, la CPU está en estado operativo "STOP", determina el tipo de arranque solicitado y adopta el siguiente estado operativo.
②	STOP → ARRANQUE	La CPU pasa al estado operativo "ARRANQUE" en los siguientes casos: • La CPU se pone en "RUN" desde el selector de modo o la programadora. • Después de disparar automáticamente un tipo de arranque mediante "Conexión (POWER ON)". • Se ejecuta la función de comunicación "RESUME" o "START". En los dos últimos casos, el selector de modo debe estar en "RUN".
③	ARRANQUE → STOP	La CPU pasa al estado "STOP" en los casos siguientes: • Durante el arranque se detecta un error. • La CPU se pone en "STOP" desde el selector de modo o la programadora. • Se procesa un comando STOP en el OB de arranque. • Se ejecuta la función de comunicación "STOP".
④	ARRANQUE → RUN	Si el arranque es correcto, la CPU pasa a "RUN".
⑤	RUN → STOP	La CPU pasa al estado "STOP" en los casos siguientes: • En el estado RUN se detecta un error y el correspondiente OB no está cargado. • La CPU se pone en "STOP" desde el selector de modo o la programadora. • Se procesa un comando STOP en el programa de usuario. • Se ejecuta la función de comunicación "STOP".

Prioridad de los estados operativos

Función

Si se solicitan al mismo tiempo varios cambios de estado operativo, se cambiará al estado operativo que tenga una mayor prioridad. Así, por ejemplo, si el selector de modo está en

"RUN" y desde la programadora se intenta cambiar la CPU a "STOP", la CPU pasa a "STOP", ya que este estado operativo tiene la máxima prioridad.

Prioridad	Estado operativo
Máxima	STOP (Página 1703)
	PARADA (Página 1712)
	ARRANQUE
Mínima	RUN (Página 1712)

Estado operativo "STOP"

Función

En estado operativo "STOP" no se procesa el programa de usuario. Todas las salidas se fijan a los valores sustitutivos, dejando el proceso controlado en un estado operativo seguro. La CPU comprueba los puntos siguientes:

- Hardware, p. ej. si todos los módulos están disponibles
- Si la CPU se rige por los valores predeterminados o si hay juegos de parámetros
- Si se cumplen las condiciones marco para el comportamiento de arranque programado

En el estado STOP se pueden recibir datos globales. Además, es posible la comunicación pasiva unilateral mediante bloques de función del sistema de comunicación para las conexiones configuradas y mediante funciones del sistema de comunicación para las conexiones no configuradas.

Consulte también

Principios básicos de los estados operativos de CPUs S7 (Página 1700)

Estado operativo "ARRANQUE"

Principios básicos del estado operativo "ARRANQUE"

Función

Antes de que la CPU, tras conectarse, empiece a procesar el programa de usuario, se ejecuta un programa de arranque. En dicho programa se pueden definir determinados valores predeterminados para el programa cíclico programando de forma adecuada los OBs de arranque.

Existen los siguientes tipos de arranque:

- Rearranque completo/arranque en caliente
- Rearranque normal
- Arranque en frío

Particularidades

Tenga en cuenta los puntos siguientes para el estado operativo "ARRANQUE":

- Se procesa el programa en el OB de arranque. El OB 100 es para "Rearranque completo (en caliente)", el OB 101 para rearranque y el OB 102 para "Arranque en frío".
- No se procesan programas controlados por tiempo o alarmas.
- Los temporizadores se actualizan.
- El contador de horas de funcionamiento está en marcha.
- Las salidas de los módulos están bloqueadas.

Consulte también

Principios básicos de los estados operativos de CPUs S7 (Página 1700)

Rearranque normal (Página 1705)

Arranque en frío (Página 1704)

OBs de arranque (OB 100, OB 101 y OB 102) (Página 1875)

Vista general completa del arranque (Página 1710)

Arranque en frío

Arranque en frío

En un rearranque en frío son repuestos todos los datos (imagen del proceso, marcas, temporizadores, contadores y bloques de datos) a los valores iniciales depositados en el programa (memoria de carga), independientemente de si fueron parametrizados como datos remanentes o no remanentes.

El programa vuelve a ejecutarse desde el principio (OB de arranque u OB 1).

Rearranque completo (en caliente)

Rearranque completo (en caliente) permitido

En los siguientes casos se puede ejecutar un "Rearranque completo (en caliente)":

- Después del borrado total
- Después de cargar el programa de usuario con la CPU en estado STOP

No se puede realizar un rearranque completo (en caliente) si el sistema ha solicitado un borrado total.

Rearranque completo (en caliente) manual

El rearranque completo (en caliente) manual se activa mediante:

- El selector de modo
- Un comando de menú de la PG o funciones de comunicación si el selector de modo está en "RUN".

Rearranque completo (en caliente) automático

En los siguientes casos puede producirse un rearranque completo (en caliente) automático al conectar la alimentación ("POWER ON"):

- Al desconectar la alimentación ("POWER OFF"), la CPU no estaba en "STOP".
- El selector de modo está en "RUN".
- El rearranque completo (en caliente) de la CPU se ha interrumpido debido a un corte en la alimentación eléctrica. Esto es independiente de la parametrización del tipo de arranque.

Rearranque completo (en caliente) automático sin respaldo

Si se utiliza la CPU sin pila de respaldo, por ejemplo, porque sea necesario el funcionamiento sin mantenimiento, tras conectar la alimentación o restablecer la tensión después de la desconexión se efectúa automáticamente un borrado total de la CPU y luego un rearranque completo (en caliente). Para poder volver a copiar el programa de usuario en la memoria de trabajo, éste debe estar en la Flash-EPROM (Memory Card).

Rearranque normal

Rearranque

Este tipo de arranque sólo es posible en el S7-400. El rearranque normal tiene las características siguientes:

- En el rearranque normal conservan su último valor válido todos los datos, incluida la imagen del proceso.
- La ejecución del programa prosigue exactamente con la instrucción que fue interrumpida.
- Las salidas no son modificadas hasta que acaba el ciclo actual.
- En caso de interrumpirse la alimentación, sólo se podrá efectuar un rearranque normal si el funcionamiento está respaldado.

Cuando se produce un corte de alimentación en RUN y seguidamente retorna la tensión, las CPUs S7-400 ejecutan una ruta de inicialización y a continuación efectúan automáticamente un rearranque normal. En caso de rearranque normal, el programa de usuario prosigue en el mismo punto en el que quedó interrumpido el procesamiento. La parte del programa de usuario que no se había procesado antes del corte de la alimentación eléctrica se denomina resto del ciclo. El resto del ciclo puede contar con partes del programa controladas por tiempo o alarmas.

Por principio, un rearranque normal sólo está permitido si el programa de usuario no se ha modificado en el estado STOP (p. ej., por recargar un bloque modificado) y si otros motivos

no hacen necesario un rearranque completo (en caliente). Se distingue entre rearranques de tipo manual y automático.

Rearranque manual

Un rearranque manual sólo resulta posible si la CPU cuenta con la parametrización correspondiente y tras los siguientes motivos de STOP:

- El selector de modo ha pasado de RUN a STOP
- STOPs programados por el usuario, STOPs tras llamada a OBs no cargados
- El estado STOP ha sido establecido por la programadora o por las funciones de comunicación

Un rearranque manual se puede disparar con un comando de menú desde la programadora o mediante funciones de comunicación (si el selector de modo se encuentra en RUN o en RUN-P). Para ello es imprescindible que el rearranque no se encuentre bloqueado por intervención (parámetros de arranque).

- Si se ha parametrizado el juego de parámetros de la CPU de rearranque manual

Rearranque automático

El rearranque automático se dispara al producirse la conexión (POWER ON) si la CPU no estaba en STOP o PARADA cuando tuvo lugar la desconexión (POWER OFF). Para ello es imprescindible que el rearranque no se encuentre bloqueado por intervención (parámetros de arranque).

Comportamiento de remanencia tras un corte de la alimentación

Áreas de datos remanentes tras un corte de la alimentación

Para evitar pérdida de datos en caso de corte de suministro, se puede determinar la remanencia de marcas, temporizadores, contadores y áreas de bloques de datos. Si el tipo de arranque es "Rearranque completo (en caliente)" —que es el valor predeterminado—, al restablecerse el suministro eléctrico se producirá un "rearranque automático con memoria".

Las CPUs S7-300 y S7-400 reaccionan de manera distinta al producirse el retorno de la tensión tras un fallo de red.

Las CPUs S7-300 sólo conocen el tipo de arranque "Rearranque completo". Para evitar la pérdida de datos en caso de corte de suministro, STEP 7 permite determinar la remanencia de marcas, temporizadores, contadores y áreas de bloques de datos. Al producirse el retorno de la tensión se efectúa un "rearranque completo automático con memoria".

Las CPUs S7-400 reaccionan frente al retorno de la tensión con un rearranque completo (en caliente) (tras la conexión (POWER ON) respaldada o no respaldada) o con un rearranque (posible únicamente tras conexión (POWER ON) respaldada), según la parametrización.

Comportamiento de remanencia en las CPUs con respaldo

La tabla siguiente muestra el comportamiento de remanencia en la memoria de trabajo de las CPUs S7-300 y S7-400 con respaldo en caso de "Rearranque completo (en caliente)" y "Arranque en frío":

Datos	Rearranque completo (en caliente)	Arranque en frío	Rearranque
Bloques en la memoria de carga	X	X	X
DB en la memoria de trabajo	X	0	X
Marcas, temporizadores, contadores parametrizados como remanentes	X	0	X
Marcas, temporizadores, contadores parametrizados como no remanentes	0	0	X
X : Los datos se conservan 0: Los datos vuelven a su valor original o se borran (contenido de los DBs)			

Comportamiento de remanencia en las CPUs sin respaldo

En las CPUs S7-400 sin respaldo no se permite efectuar rearranques.

La siguiente tabla muestra el comportamiento de remanencia en la memoria de trabajo de las CPUs S7-300 sin respaldo en caso de "Rearranque completo (en caliente)" y "Arranque en frío":

Datos	Rearranque completo (en caliente)		Arranque en frío	
	S7-300	S7-400	S7-300	S7-400
Bloques en la memoria de carga	VC	VC	VC	VC
DB en la memoria de trabajo parametrizado como remanente	VX	no es posible	V	no es posible
DB en la memoria de trabajo parametrizado como no remanente	V	V	V	V
Marcas, temporizadores, contadores parametrizados como remanentes	X	0	0	0
Marcas, temporizadores, contadores parametrizados como no remanentes	0	0	0	0
X : Los datos se conservan 0: Los datos vuelven a su valor original o se borran (contenido de los DBs) V: Los datos se restauran al valor predeterminado de la memoria EPROM VC: El bloque lógico de la EPROM se conserva; si el bloque lógico está sobrecargado se pierde VX: El bloque de datos sólo se conserva si los datos remanentes presentes en la EPROM se toman de la NV-RAM (los bloques de datos cargados o generados en la RAM se pierden)				

Actividades de arranque

Resumen

La tabla siguiente muestra qué actividades realiza la CPU durante el arranque:

Actividades en el orden de procesamiento	en caso de rearranque (en caliente)	en caso de arranque en frío	en caso de rearranque
Borrar la pila de interrupciones y bloques	X	X	0
Borrar marcas, temporizadores y contadores no remanentes	X	0	0
Borrar todas las marcas, temporizadores y contadores	0	X	0
Borrar la memoria imagen de proceso de las salidas	X	X	parametrizable
Restablecer las salidas de los módulos de señales	X	X	parametrizable
Rechazar alarmas de procesos	X	X	0
Rechazar alarmas de retardo	X	X	0
Rechazar alarmas de diagnóstico	X	X	X
Actualizar lista de estado del sistema (SZL)	X	X	X
Evaluar los parámetros del módulo y transferirlos al módulo o enviar valores estándar	X	X	X
Procesamiento del OBs de arranque correspondiente	X	X	X
Procesar el resto del ciclo (parte del programa de usuario que no ha podido terminar de procesarse debido a una desconexión "POWER OFF")	0	0	X
Actualizar la memoria imagen de proceso de las entradas	X	X	X
Liberar salidas digitales (cancelar OD (Output Disabled) de la señal) tras la transición del estado operativo a "RUN"	X	X	X
X : La actividad se lleva a cabo 0: La actividad no se lleva a cabo			

Consulte también

Vista general completa del arranque (Página 1710)

Cancelación de un arranque

Causas de la cancelación

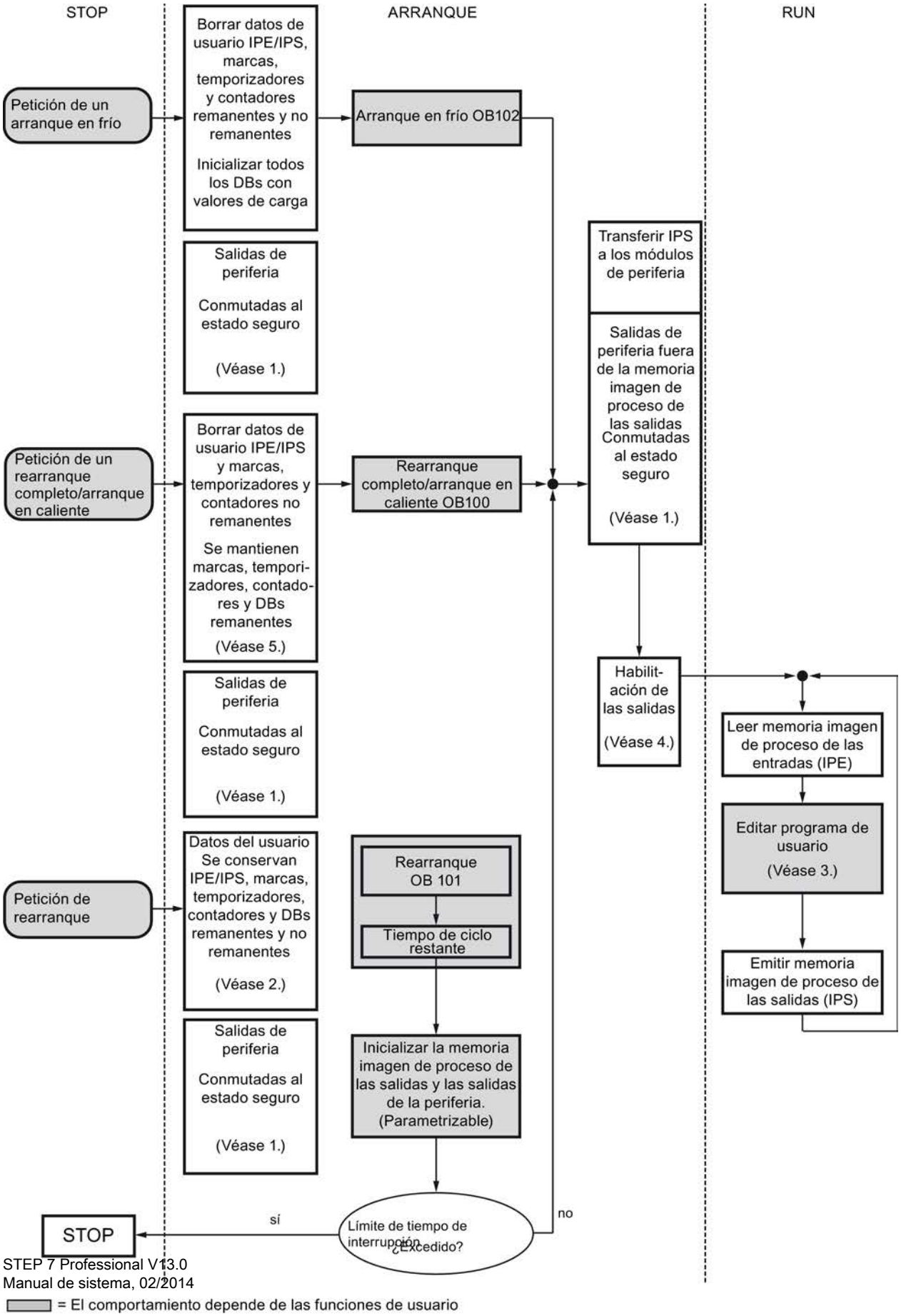
Si durante el arranque se producen errores, se cancela el arranque y la CPU pasa o permanece en "STOP". Esto tiene las siguientes consecuencias sobre el estado operativo Rearranque completo (en caliente):

- Hay que repetir el rearranque completo (en caliente) interrumpido.
- Si el rearranque ya ha comenzado, resulta posible tanto un rearranque completo (en caliente) como un rearranque.
- El rearranque completo (en caliente) o rearranque no se efectúa, o bien se interrumpe, si se dan las condiciones siguientes:
  - Si el selector de modo de la CPU está en "STOP".
  - Si se solicita un borrado total.
  - Si hay una Memory Card con ID de aplicación no permitida.
  - Si en el modo de procesador único hay enchufada más de una CPU.
  - Si el programa de usuario contiene un bloque de organización bloqueado o desconocido para la CPU.
  - Tras la conexión del suministro eléctrico, si la CPU constata que no están conectados realmente todos los módulos enumerados en la configuración creada con STEP 7 (no se permite parametrizar la diferencia teórico-real).
  - Si en la parametrización de los módulos se producen errores.
- El rearranque no se realiza, o bien se cancela, si se dan las condiciones siguientes:
  - Si previamente se había efectuado un borrado total de la CPU (tras un borrado total, sólo resulta posible el rearranque completo (en caliente)).
  - Si se ha superado el límite de interrupción (el tiempo de interrupción es el tiempo que transcurre desde que se sale de RUN hasta que se ha procesado el OB de arranque, tiempo de ciclo restante incluido).
  - Si la configuración de módulo ha cambiado (p. ej., sustitución de módulo).
  - Si la parametrización sólo permite un rearranque completo (en caliente).
  - Si en el STOP se han cargado, borrado o modificado bloques.

Vista general completa del arranque

Proceso

La figura siguiente muestra las actividades de la CPU en los estados operativos "STOP", "ARRANQUE" y "RUN".


STOP
ARRANQUE
RUN
Petición de un arranque en frío
Borrar datos de usuario IPE/IPS, marcas, temporizadores y contadores remanentes y no remanentes
Inicializar todos los DBs con valores de carga
Arranque en frío OB102
Salidas de periferia
Conmutadas al estado seguro
(Véase 1.)
Transferir IPS a los módulos de periferia
Salidas de periferia fuera de la memoria imagen de proceso de las salidas Conmutadas al estado seguro
(Véase 1.)
Petición de un rearranque completo/arranque en caliente
Borrar datos de usuario IPE/IPS y marcas, temporizadores y contadores no remanentes
Se mantienen marcas, temporizadores, contadores y DBs remanentes
(Véase 5.)
Rearranque completo/arranque en caliente OB100
Salidas de periferia
Conmutadas al estado seguro
(Véase 1.)
Habilitación de las salidas
(Véase 4.)
Leer memoria imagen de proceso de las entradas (IPE)
Editar programa de usuario
(Véase 3.)
Emitir memoria imagen de proceso de las salidas (IPS)
Petición de rearranque
Datos del usuario Se conservan IPE/IPS, marcas, temporizadores, contadores y DBs remanentes y no remanentes
(Véase 2.)
Rearranque OB 101
Tiempo de ciclo restante
Salidas de periferia
Conmutadas al estado seguro
(Véase 1.)
Inicializar la memoria imagen de proceso de las salidas y las salidas de la periferia. (Parametrizable)
Límite de tiempo de interrupción ¿Excedido?
sí
no
STOP
= El comportamiento depende de las funciones de usuario

Leyenda de la figura:

1. Todas las salidas de periferia están conectadas por hardware en el estado seguro a través de los módulos de periferia (valor predeterminado = "0"). Esto es independiente de si se utilizan en el programa de usuario dentro o fuera del área de memoria imagen de proceso. En caso de utilizar módulos de señales sustituibles, el comportamiento de las salidas se puede parametrizar, p. ej., mantener el último valor.
2. Es necesario para el procesamiento del resto del ciclo.
3. También los OBs de alarma disponen de una memoria imagen de proceso actual de las entradas en la primera llamada.
4. El estado de las salidas de periferia centralizadas y descentralizadas en el primer ciclo del programa de usuario se puede definir de la siguiente forma:
   - Utilizar módulos de salida parametrizables para aplicar valores sustitutivos o mantener el último valor.
   - En caso de rearranque: Activar el parámetro de arranque de CPU "Inicializar salidas al efectuar un rearranque" para emitir un "0" (ajuste predeterminado).
   - Predefinir las salidas en el OB de arranque (OB 100, OB 101, OB 102).
5. En los sistemas S7-300 sólo se conservan las áreas de DB configuradas como remanentes.

Estado operativo "RUN"

Función

En estado operativo "RUN" se procesa el programa cíclico, controlado por tiempo y por alarmas:

- Se lee la imagen de proceso de las entradas.
- Se procesa el programa de usuario.
- Se emite la memoria imagen de proceso de las salidas.

El intercambio activo de datos entre CPUs mediante la comunicación de datos globales (tabla de datos globales) y mediante instrucciones de comunicación para conexiones configuradas e instrucciones de comunicación para conexiones no configuradas sólo resulta posible en el estado RUN.

Consulte también

Principios básicos de los estados operativos de CPUs S7 (Página 1700)

Estado operativo "PARADA"

Función

El estado operativo "PARADA" ocupa una posición especial, ya que sólo se utiliza para pruebas en "ARRANQUE" o "RUN".

Estado operativo "PARADA":

- Todos los temporizadores se congelan. Los temporizadores y contadores de horas de funcionamiento no se procesan, los tiempos de vigilancia y los tiempos base de los niveles controlados por tiempo se detienen.
- El reloj de tiempo real sigue en marcha.
- Las salidas no están habilitadas, pero se pueden habilitar para realizar pruebas.
- Se pueden controlar las entradas y salidas.
- En caso de corte y restablecimiento de la alimentación, las CPUs respaldadas pasan de "PARADA" a "STOP" y no efectúan ningún "Rearranque completo (en caliente)". Las CPUs no respaldadas realizan un "Rearranque completo (en caliente)" automático no respaldado al restablecerse el suministro.
- Se pueden recibir datos globales. Además, es posible la comunicación pasiva unilateral mediante SFBs de comunicación para las conexiones configuradas y mediante SFCs de comunicación para las conexiones no configuradas.

Consulte también

Principios básicos de los estados operativos de CPUs S7 (Página 1700)

Principios básicos del borrado total

Función

En el estado STOP se puede efectuar un borrado total de la CPU. El borrado total se puede realizar manualmente con el selector de modo (MRES) o desde la programadora, p. ej. antes de cargar un programa de usuario.

Con el borrado total, la CPU vuelve a su "estado original". Esto significa que:

- El programa de usuario se borra íntegramente de la memoria de trabajo y de la memoria de carga, y lo mismo ocurre con todas las áreas de operandos.
- Los parámetros de sistema y los parámetros de la CPU y del módulo vuelven a su configuración estándar. Los parámetros MPI configurados antes del borrado total se conservan.
- Si hay insertada una Memory Card (Flash-EPROM) o Micro Memory Card, la CPU copia el programa de usuario de la Memory Card en la memoria de trabajo. Si los datos de configuración también están guardados en la Memory Card, también se copian los parámetros de la CPU y del módulo.
- El búfer de diagnóstico, los parámetros MPI, la hora y el contador de horas de funcionamiento no se ponen a cero.

Áreas de memoria

Qué hay que saber sobre las Micro Memory Cards para S7-300

Función de la Micro Memory Card

La Micro Memory Card SIMATIC es un módulo de memoria especial para una CPU S7-300. En principio, la Micro Memory Card se puede utilizar como memoria de carga o como soporte de datos portátil.

STEP 7 V11 permite su uso como memoria de carga.

Para el funcionamiento, la Micro Memory Card SIMATIC debe estar insertada en la CPU.

Si la Micro Memory Card se formatea con recursos Windows, como puede ser un lector de tarjetas de los que se encuentran habitualmente en el mercado, la Micro Memory Card quedará inutilizada como soporte de memoria para una CPU S7.

Qué se puede almacenar en una Micro Memory Card SIMATIC

En una Micro Memory Card se puede almacenar el programa de usuario (bloques y datos de sistema).

La Micro Memory Card también se puede utilizar para actualizar el firmware o para hacer una copia de seguridad del firmware de la CPU.

Transferir objetos del proyecto a la Micro Memory Card

Si la Micro Memory Card está insertada en la programadora o en un programador de EPROM externo, se pueden copiar en ella los siguientes objetos desde el árbol del proyecto:

- Bloques sueltos (también es posible seleccionar varios)
  En este caso, se ofrecerá la posibilidad de hacer la transferencia de forma coherente, es decir, teniendo en cuenta las dependencias entre los bloques mediante llamadas de bloque.
- PLC
  En este caso, todos los objetos relevantes para la ejecución, entre ellos, los bloques y la configuración del hardware, se transfieren a la Micro Memory Card, al igual que sucede al cargar.

La transferencia de los objetos se puede realizar mediante la función Drag&Drop o con el comando "Escribir en tarjeta de memoria" del menú "Proyecto".

Transferir objetos de la Micro Memory Card al proyecto

Los bloques sueltos (es posible la selección múltiple) se transfieren al proyecto mediante la función Drag&Drop. Una configuración de hardware no se puede transferir de la Memory Card al proyecto.

Copia de seguridad del firmware en la Micro Memory Card

Recomendamos hacer una copia de seguridad del firmware para casos de emergencia.

Para ello hay que insertar una Micro Memory Card vacía en la CPU y seguir un procedimiento determinado: La descripción se puede encontrar en la documentación de la CPU.

Documentación del S7-300 (http://support.automation.siemens.com/WW/view/de/13008499/0/es)

Actualizar el firmware con una Micro Memory Card

La versión más reciente del firmware se puede descargar de Internet como archivo *.UPD en las páginas de Service & Support:

http://support.automation.siemens.com (http://support.automation.siemens.com)

Guarde los archivos de firmware en el disco duro.

Para guardar el archivo en la Micro Memory Card, seleccione en el menú "Proyecto" el comando "Crear actualización de Firmware en tarjeta de memoria".

A continuación siga las instrucciones del portal Service & Support para actualizar el firmware con la CPU.

Consulte también

Áreas de memoria de las CPUs S7-300 (Página 1718)

Qué hay que saber sobre las Memory Cards para el S7-400

Función de la Memory Card

En todas las CPUs S7-400 se puede introducir una Memory Card. Esta sirve como ampliación de la memoria de carga. Dependiendo del tipo de Memory Card utilizada, el programa de usuario se conserva en la Memory Card aunque se desconecte la tensión.

Tipos de Memory Card

- Tarjeta RAM: Exclusivamente como ampliación de la memoria de carga integrada.
- Tarjeta FLASH: Para almacenar el programa de usuario aunque se desconecte la tensión sin tensión de respaldo o fuera de la CPU.

Transferir objetos del proyecto a la Memory Card

Si la Memory Card está insertada en la programadora o en un programador de EPROM externo, se pueden copiar en ella los siguientes objetos desde el árbol del proyecto:

- Bloques individuales (posibilidad de selección múltiple)
  En este caso, se ofrece una transferencia coherente, es decir, se tiene en cuenta la interdependencia de los bloques por llamadas.
- PLC
  En este caso, todos los objetos relevantes para la ejecución, como los bloques y la configuración hardware, entre otros, se transfieren a la Memory Card, al igual que durante el proceso de carga.

La transferencia de los objetos se puede realizar mediante la función Drag&Drop o con el comando "Escribir en tarjeta de memoria" del menú "Proyecto" y siguiendo las instrucciones que aparezcan en la pantalla.

Transferir objetos de la Memory Card al proyecto

Los bloques individuales (posibilidad de selección múltiple) se transfieren al proyecto mediante la función Drag&Drop. Las configuraciones hardware no se pueden transferir de la Memory Card al proyecto.

Cargar un programa de usuario en la Memory Card

El comando "Cargar programa de usuario en Memory Card" del menú "Online" permite guardar todo el programa de usuario en la tarjeta FLASH. La tarjeta FLASH se encuentra en la CPU. Antes de cargar, se borra todo lo que haya en la tarjeta FLASH.

- La CPU está en estado operativo STOP.
- Sólo se pueden cargar tarjetas FLASH con este comando de menú

Actualizar firmware con una Memory Card

La versión más reciente del firmware se puede descargar de Internet como archivo *.UPD en las páginas de Service & Support:

http://support.automation.siemens.com (http://support.automation.siemens.com)

Guarde los archivos de firmware en el disco duro.

Para guardar el archivo en la Memory Card, seleccione en el menú "Proyecto" el comando "Crear Memory Card con actualización de firmware".

A continuación, siga las instrucciones del portal Service & Support para actualizar el firmware con la CPU.

Consulte también

Áreas de memoria de las CPUs S7-400 (Página 1720)

Distribución de las áreas de memoria

Distribución

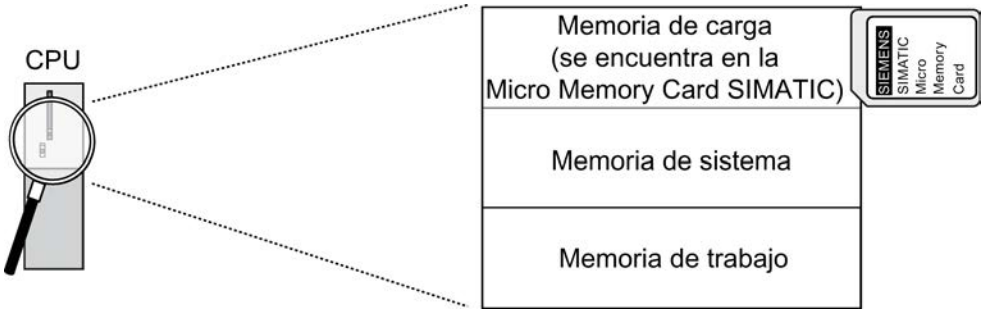
La memoria de las CPUs S7 puede distribuirse en tres áreas:

- Memoria de carga (según el tipo de CPU o de Memory Card será memoria RAM, EPROM o Flash)
- Memoria de trabajo (RAM integrada)
- Memoria de sistema (RAM integrada)

Áreas de memoria S7-300

Áreas de memoria de las CPUs S7-300

Las tres áreas de memoria de la CPU



Memoria de carga

La memoria de carga se encuentra en la Micro Memory Card SIMATIC y equivale exactamente al tamaño de la Micro Memory Card SIMATIC. Sirve para guardar bloques lógicos y bloques de datos, así como información del sistema (configuración, enlaces, parámetros del módulo, etc.). Los bloques que no se consideran relevantes para la ejecución se guardan exclusivamente en la memoria de carga. Además es posible almacenar todos los datos de configuración de un proyecto en la SIMATIC Micro Memory Card.

Nota

La transferencia de programas de usuario y, por consiguiente, el funcionamiento de la CPU sólo es posible si hay una SIMATIC Micro Memory Card insertada en la CPU.

Memoria de sistema

La memoria del sistema está integrada en la CPU y no se puede ampliar.

Contiene

- Las áreas de operandos Marcas, Temporizadores y Contadores
- Las imágenes del proceso de entradas y salidas
- Los datos locales

Memoria de trabajo

La memoria de trabajo está integrada en la CPU y no se puede ampliar. Sirve para procesar el código y los datos del programa de usuario. Este procesamiento tiene lugar exclusivamente en el área de la memoria de trabajo y en la memoria del sistema.

Consulte también

Qué hay que saber sobre las Micro Memory Cards para S7-300 (Página 1714)

Remanencia de la memoria de carga, sistema y trabajo

Su CPU posee una memoria remanente libre de mantenimiento, es decir, no se requiere pila de respaldo para el funcionamiento. Gracias a la remanencia se mantiene el contenido de la memoria remanente también tras desconectar la alimentación y tras un rearranque completo (en caliente).

Datos remanentes en la memoria de carga

Su programa en la memoria de carga siempre es remanente: Se almacena ya durante la carga en la Micro Memory Card SIMATIC de forma protegida contra cortes de alimentación y un borrado total.

Datos remanentes en la memoria del sistema

En el caso de marcas, temporizadores y contadores, es posible determinar mediante configuración (Propiedades de la CPU) qué partes deben ser remanentes y cuáles deben inicializarse a "0" en el rearranque completo (en caliente).

El búfer de diagnóstico, la dirección MPI y la velocidad de transferencia, así como el contador de horas de funcionamiento suelen almacenarse en el área de memoria remanente de la CPU. Gracias a la remanencia de la dirección MPI y la velocidad de transferencia se garantiza que la CPU pueda seguir comunicándose después de una caída de tensión, de un borrado total o de pérdida de los parámetros de comunicación (al retirar la Micro Memory Card SIMATIC o borrar los parámetros de comunicación).

Datos remanentes en la memoria de trabajo

Así, el contenido de los DBs remanentes es fundamentalmente remanente en caso de rearranque completo y de desconexión y conexión. Los bloques de datos remanentes se pueden cargar en la memoria de trabajo hasta el límite de remanencia máximo de dicha memoria.

Cuando se realiza un rearranque completo o una desconexión y conexión, los DBs no remanentes se inicializan con sus valores iniciales de la memoria de carga. Los bloques de datos no remanentes y los bloques de código se pueden cargar hasta el límite máximo de la memoria de trabajo.

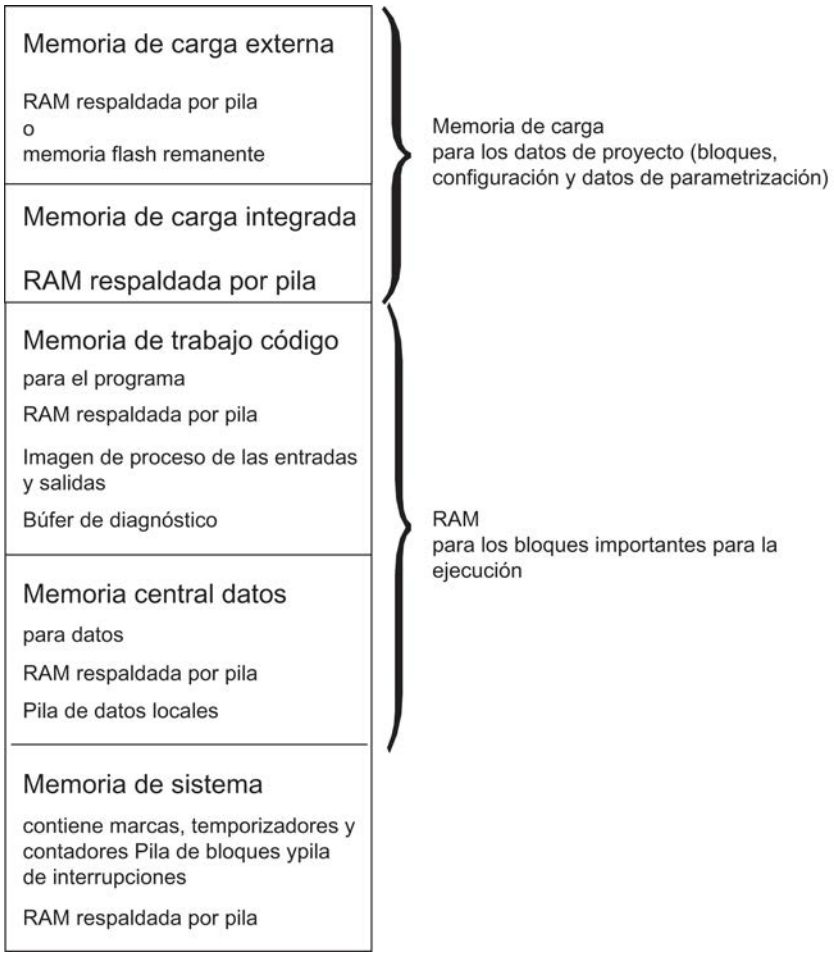
El tamaño de la memoria de trabajo remanente (para bloques de datos remanentes) de la CPU se puede encontrar en los datos técnicos de la CPU en cuestión.

Áreas de memoria S7-400

Áreas de memoria de las CPUs S7-400

Distribución de las áreas de memoria

La memoria de las CPUs S7-400 puede distribuirse en las áreas siguientes:



Tipos de memoria en las CPUs S7-400

- Memoria de carga para los datos del proyecto, p. ej. bloques, configuración y datos de parametrización.
- Memoria de trabajo para los bloques importantes para la ejecución (bloques lógicos y bloques de datos).

- Memoria del sistema (RAM) con los elementos de almacenamiento que cada CPU pone a disposición del programa de usuario, tales como marcas, temporizadores o contadores. La memoria del sistema contiene además la pila de bloques y la pila de interrupciones.
- La memoria del sistema de la CPU proporciona asimismo memorias intermedias (pila de datos locales, búfer de diagnóstico y recursos de comunicación), que se asignan al programa al solicitarse un bloque para sus datos temporales. Estos datos son válidos únicamente mientras está activado el bloque.
  Modificando los valores predeterminados para la memoria imagen de proceso, los datos locales, el búfer de diagnóstico y los recursos de comunicación (propiedades de la CPU parametrizables) es posible modificar la memoria de trabajo disponible para bloques relevantes para la ejecución.

Nota

Tenga en cuenta lo siguiente al ampliar la imagen del proceso de una CPU: Reconfigure los módulos cuyas direcciones deban ser superiores a la dirección más alta de la imagen de proceso de manera que las nuevas direcciones sigan siendo superiores a la dirección más alta de la imagen de proceso ampliada.

Nota importante para las CPUs después de la parametrización modificada de la distribución de la memoria de trabajo

Si modifica la división de la memoria de trabajo mediante parametrización, la memoria de trabajo se reorganizará al cargar los datos de sistema en la CPU. Esto tiene como consecuencia que los bloques de datos que hayan sido creados con una función del sistema (SFC) serán borrados y a los demás bloques de datos se les asignarán valores iniciales de la memoria de carga.

El tamaño disponible de la memoria de trabajo para bloques de código y de datos se modifica durante la carga de datos del sistema si se alteran los siguientes parámetros:

- Tamaño de la imagen de proceso (byte por byte; en la ficha "Ciclo/Marca de ciclo")
- Recursos para comunicación (sólo S7-400; en la ficha "Memoria")
- Tamaño del búfer de diagnóstico (en la ficha "Diagnóstico/Reloj")
- Número de datos locales para todas las clases de prioridad (ficha "Memoria")

Base de cálculo para estimar la memoria central necesaria

Con el fin de no sobrepasar el tamaño de la memoria central disponible en la CPU, al parametrizar se deberán tener en cuenta los siguientes requerimientos de memoria:

Tabla 8-90 Memoria necesaria

Parámetros	Memoria de trabajo necesaria	En la memoria de programa/datos
Tamaño de la imagen de proceso (entradas)	12 bytes por byte en la memoria imagen de proceso de las entradas A partir de la V6.0: 20 bytes por byte en la imagen de proceso de las entradas	Memoria de programa
Tamaño de la imagen de proceso (salidas)	12 bytes por byte en la memoria imagen de proceso de las salidas A partir de la V6.0: 20 bytes por byte en la imagen de proceso de las entradas	Memoria de programa
Recursos de comunicación (peticiones de comunicación)	72 bytes por petición de comunicación	Memoria de programa
Tamaño del búfer de diagnóstico	32 bytes por entrada del búfer de diagnóstico	Memoria de programa
Número de datos locales	1 byte por cada byte de datos locales	Memoria de datos

Capacidad de memoria flexible

- Memoria de trabajo:
  La capacidad de la memoria de trabajo se determina eligiendo de la variada gama de unidades CPU una CPU adecuada.
- Memoria de carga:
  Para los programas de volumen pequeño y medio es suficiente la memoria de carga integrada.
  Para los programas más extensos se puede ampliar la memoria de carga enchufando tarjetas de memoria RAM.
  Además, hay disponibles tarjetas de memoria FLASH para conservar los programas incluso sin pila tampón en caso de interrumpirse la tensión. Asimismo, estas tarjetas de memoria FLASH (a partir de 8 MB) son adecuadas para enviar y ejecutar actualizaciones del sistema operativo.

Respaldo

- La pila tampón respalda transitoriamente la parte integrada y externa de la memoria de carga, la sección de datos de la memoria de trabajo y la sección de código.

Consulte también

Qué hay que saber sobre las Memory Cards para el S7-400 (Página 1716)

Contenido de las memorias de trabajo y del sistema

Áreas de operandos de las CPUs

Resumen

Los elementos de memoria que cada CPU pone a disposición del programa de usuario, p. ej. la memoria imagen de proceso de las entradas y salidas, marcas, temporizadores y contadores. La memoria de trabajo y del sistema de las CPUs S7 se distribuye en áreas de operandos. Si se utilizan las operaciones adecuadas, los datos se direccionan en el programa directamente al área de operandos correspondiente.

La tabla siguiente muestra las áreas de operandos:

Área de operandos	Acceso vía unidades del tamaño siguiente:	Notación S7	Descripción
Memoria imagen de proceso de las entradas	Entrada (bit)	I	Al principio de cada ciclo, la CPU lee las entradas de los módulos de entrada y guarda los valores en la memoria imagen de proceso de las entradas.
	Byte de entrada	IB	
	Palabra de entrada	IW	
	Palabra doble de entrada	ID	
Memoria imagen de proceso de las salidas	Salida (bit)	Q	Durante el ciclo, el programa calcula los valores para las salidas y los guarda en la memoria imagen de proceso de las salidas. Al final del ciclo, la CPU escribe los valores calculados para las salidas en los módulos de salida.
	Byte de salida	QB	
	Palabra de salida	QW	
	Palabra doble de salida	QD	
Marcas	Marca (bit)	M	Esta área pone a disposición espacio de memoria para resultados intermedios calculados en el programa.
	Byte de marcas	MB	
	Palabra de marcas	MW	
	Palabra doble de marcas	MD	
Temporizadores	Temporizador (T)	T	En esta área hay disponibles temporizadores.
Contadores	Contador (C)	C	En esta área hay disponibles contadores.

Área de operandos	Acceso vía unidades del tamaño siguiente:	Notación S7	Descripción
Bloque de datos	Bloque de datos, abierto con "OPN DB":	DB	Los bloques de datos guardan información del programa. Existe la posibilidad de definirlos de modo que todos los bloques lógicos accedan a ellos (DBs globales) o que estén asignados a un FB o SFB determinado (DB instancia).
	Bit de datos	DBX	
	Byte de datos	DBB	
	Palabra de datos	DBW	
	Palabra doble de datos	DBD	
	Bloque de datos, abierto con "OPN DI":	DI	
	Bit de datos	DIX	
	Byte de datos	DIB	
	Palabra de datos	DIW	
	Palabra doble de datos	DID	
Datos locales	Bit de datos locales	L	Esta área de memoria aloja los datos temporales de un bloque mientras dura el procesamiento del mismo. La pila L-Stack también proporciona memoria para transferir parámetros de bloque y para guardar los resultados intermedios de los segmentos KOP.
	Byte de datos locales	LB	
	Palabra de datos locales	LW	
	Palabra doble de datos locales	LD	
Área de periferia: Entradas	Byte de entrada de periferia	PIB	Las áreas de periferia de las entradas y salidas permiten acceder directamente a los módulos de entrada y salida descentralizados.
	Palabra de entrada de periferia	PIW	
	Palabra doble de entrada de periferia	PID	
Área de periferia: Salidas	Byte de salida de periferia	PQB	
	Palabra de salida de periferia	PQW	
	Palabra doble de salida de periferia	PQD	

Memoria imagen de proceso de las entradas y salidas

Principios básicos de las memorias imagen de proceso

Función

Si en el programa de usuario se activan las áreas de operandos de entradas (I) y salidas (Q), no se consultan o modifican los estados de señal en los módulos de señales digitales, sino un área de la memoria de sistema de la CPU. Dicha área de memoria se denomina memoria imagen de proceso.

Ventajas de la memoria imagen de proceso

El acceso a la memoria imagen de proceso tiene la ventaja, en comparación con el acceso directo a los módulos de entrada y salida, que la CPU dispone de una imagen coherente de las señales de proceso mientras dura el procesamiento cíclico del programa. Si durante el procesamiento del programa cambia un estado lógico en un módulo de entrada, dicho estado se conserva en la memoria imagen de proceso hasta que ésta se actualiza en el próximo ciclo. Esto permite consultar varias veces una señal de entrada en el programa y obtener siempre una información de entrada coherente.

Asimismo, el acceso a la memoria imagen de proceso requiere bastante menos tiempo que el acceso directo a los módulos de señales, porque la memoria imagen de proceso se encuentra en la memoria interna de la CPU.

Actualizar las memorias imagen de proceso

Ejecución

La actualización de las memorias imagen de proceso de las entradas y salidas (memoria imagen de proceso OB 1) se efectúa en el orden siguiente:

1. Se ejecutan las tareas internas del sistema operativo.
2. La memoria imagen de proceso de las salidas (IPS) se escribe en las salidas de los módulos.
3. El estado de las entradas se lee en la memoria imagen de proceso de las entradas (IPE).
4. Se procesa el programa de usuario con todos los bloques que se han llamado en él.

El sistema operativo controla independientemente la escritura de la memoria imagen de proceso de las salidas en las salidas de los módulos y la lectura de la memoria imagen de proceso de las entradas.

Error de acceso a la periferia al actualizar la memoria imagen de proceso

Las CPUs S7-300 reaccionan de la siguiente forma en caso de producirse un error durante la actualización de la memoria imagen de proceso:

- No se efectúan entradas en el búfer de diagnóstico ni llamadas de OB. Los bytes de entrada correspondientes se ponen a "0" y permanecen así hasta que se elimine el error.

Las CPUs S7-400 reaccionan de la siguiente forma en caso de producirse un error durante la actualización de la memoria imagen de proceso:

- Entrada en el búfer de diagnóstico y arranque del OB 85 cada vez que se accede a la periferia en cada actualización de la correspondiente memoria imagen de proceso. Los bytes de entrada erróneos se ponen a "0" en cada acceso a la memoria imagen de proceso.

La parametrización de la reacción en caso de error de acceso a la periferia se puede modificar de una de las dos maneras siguientes:

- La CPU genera una entrada en el búfer de diagnóstico e inicia el OB 85 en cada acceso.
- La CPU genera una entrada en el búfer de diagnóstico y sólo inicia el OB 85 en caso de que entre o salga un error de acceso a la periferia. Antes de llamar el OB 85, los bytes de entrada defectuosos se ponen a "0", tras lo cual el sistema operativo no vuelve a sobrescribirlos hasta que salga el error de acceso a la periferia.
- La CPU muestra el comportamiento predefinido para el S7-300. No se efectúa llamada del OB-85, los bytes de entrada correspondientes se ponen a "0" y el sistema operativo no los sobrescribe hasta que se elimina el error.

Frecuencia de arranque del OB-85

Aparte de la reacción que se haya parametrizado para cuando se produzca un error de acceso a la periferia, el área de direcciones de un módulo también influye en la frecuencia de arranque del OB-85:

- En el caso de un módulo con un área de direcciones de hasta una palabra doble, el OB 85 se inicia una vez, por ejemplo, con un módulo digital con hasta 32 entradas o salidas o un módulo analógico con 2 canales.
- En el caso de los módulos con un área de direcciones mayor, el OB 85 se inicia tantas veces como se acceda a él con comandos de palabra doble, por ejemplo, en un módulo de señales analógicas de 4 canales será dos veces.

Pila de datos locales

Función

Al crear los bloques de organización es posible declarar variables temporales (TEMP) disponibles únicamente durante el procesamiento del bloque, después de lo cual vuelven a sobrescribirse. Además, cada bloque de organización necesita para su información de arranque 20 bytes de datos locales.

La CPU tiene una memoria limitada para las variables temporales de los datos locales de los bloques que se están procesando en ese momento. El tamaño de esta área de memoria,

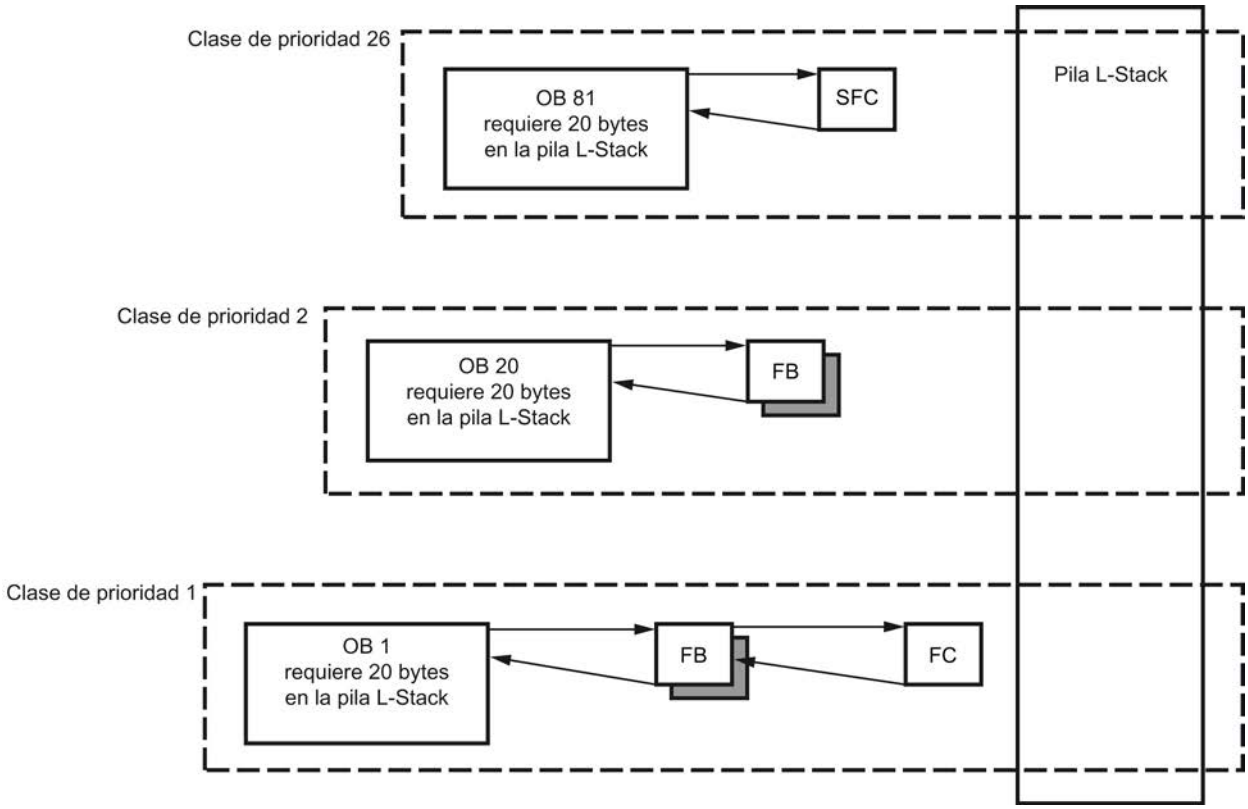
denominada pila de datos locales o pila L-Stack, depende de la CPU. De forma predeterminada, la pila de datos locales se reparte a partes iguales entre las clases de prioridades. Esto significa que cada clase de prioridad dispone de su propio espacio en la pila de datos locales. De esta forma se garantiza que también las clases de prioridades con mayor prioridad y sus OBs subordinados tengan espacio para sus datos locales.

Antes del primer acceso deben inicializarse los datos locales.

La pila de datos locales almacena los siguientes datos:

- Variables temporales de los datos locales de bloques
- Información de arranque de los bloques de organización
- Información para la transferencia de parámetros
- Resultados intermedios de la lógica en programas de esquemas de contactos

La figura siguiente muestra cómo se asignan los datos locales a las clases de prioridades:



En la pila de datos locales, el OB 1 es interrumpido primero por el OB 20, que a su vez será interrumpido por el OB 81.

⚠ PRECAUCIÓN

Todas las variables temporales de un OB y de sus bloques subordinados se almacenan en la pila de datos locales. Si se utilizan muchos niveles de anidamiento en el procesamiento de los bloques, existe riesgo de desbordamiento de la pila de datos locales.

Si se supera el tamaño permitido para la pila de datos locales para un programa, las CPUs S7 pasan al estado operativo "STOP".

La función "Mostrar información del programa" ofrece un resumen de los datos locales necesarios.

También hay que tener en cuenta los datos locales requeridos por los OBs de error síncrono.

Asignar datos locales a las clases de prioridades

Con S7-300: En el S7-300, el número máximo de datos locales por clase de prioridad y por bloque está limitado (ver Datos técnicos de la CPU).

Con S7-400: El número máximo total de datos locales está limitado (ver Datos técnicos de la CPU). Los datos locales disponibles se distribuyen libremente al parametrizar la CPU entre las distintas clases de prioridades. No se debe superar el número máximo de datos locales.

Pila de interrupciones

Función

Si el procesamiento del programa se ve interrumpido por un bloque de organización de mayor prioridad, el sistema operativo almacena los siguientes datos en la pila de interrupciones (USTACK):

- El contenido actual de los acumuladores y el registro de direcciones
- El número y tamaño de los bloques de datos abiertos

Cuando se termina de procesar el nuevo bloque de organización, el sistema operativo carga la información de la pila de interrupciones. A continuación se retoma el procesamiento del bloque interrumpido desde el mismo lugar en el que se produjo la interrupción.

Pila de bloques

Función

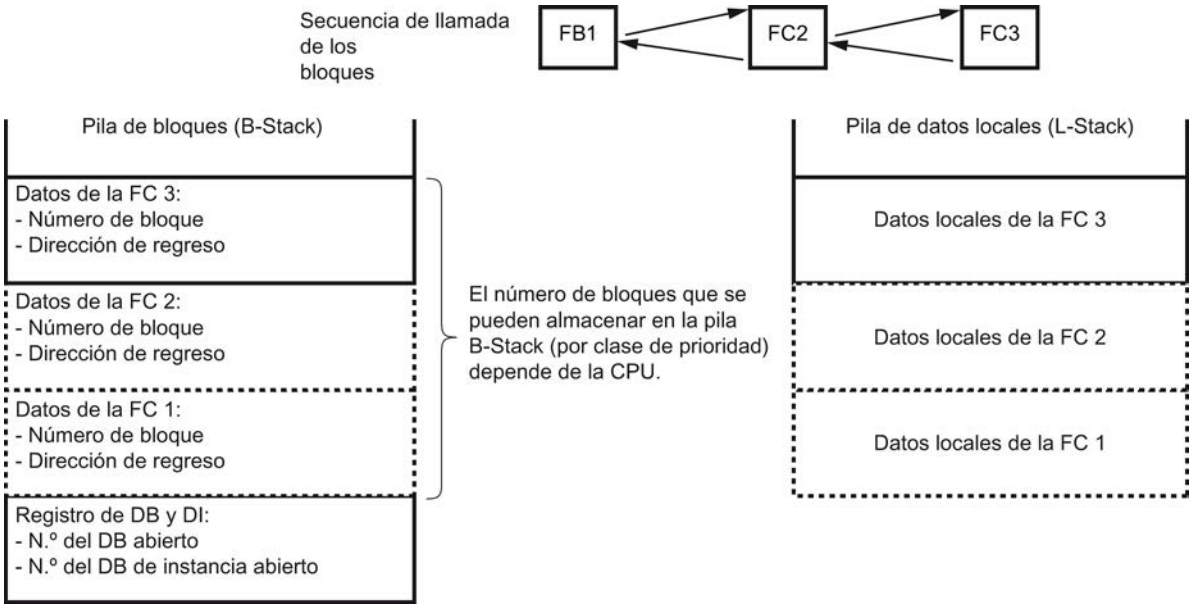
Si se interrumpe el procesamiento de un bloque por la llamada de otro bloque o por una prioridad mayor, por ejemplo, una alarma o tratamiento de errores, el sistema operativo almacena los siguientes datos en la pila de bloques (B-Stack):

- Número, tipo (OB, FB, FC, SFB, SFC) y dirección de salto del bloque interrumpido.
- Número de los bloques de datos del registro de DB y DI abiertos en el momento de la interrupción.

Con los datos almacenados, el programa de usuario puede continuar el procesamiento después de la interrupción.

En el estado operativo "STOP", la función de test "Jerarquía de llamada" permite leer en la PG la pila de bloques si el editor de programación está abierto. La pila de bloques indica todos los bloques que no habían terminado de procesarse en el momento en el que la CPU pasó al estado operativo "STOP".

La figura siguiente muestra la secuencia de llamada de los bloques:



Registro del bloque de datos

Existen dos registros de bloques de datos. Estos registros contienen los números de los bloques de datos abiertos:

- El registro de DB incluye el número del bloque de datos globales abierto.
- El registro de DI incluye el número del bloque de datos de instancia abierto.

Búfer de diagnóstico

Función

En el búfer de diagnóstico se registran los eventos de diagnóstico del sistema y definidos por el usuario en el orden en el que se producen. La primera entrada contiene el evento más reciente. El número de entradas del búfer de diagnóstico depende del módulo en cuestión y de su actual estado operativo.

Posibles eventos de diagnóstico:

- Fallo en un módulo
- Fallo en el cableado del proceso
- Fallo de sistema en la CPU
- Transiciones de estado operativo de la CPU
- Fallo en el programa de usuario
- Eventos de diagnóstico definidos por el usuario con la función de sistema SFC 52

La información que se registra en el búfer de diagnóstico al producirse un evento de diagnóstico del sistema es idéntica a la información de arranque que se envía al bloque de organización correspondiente.

Las entradas del búfer de diagnóstico no se pueden borrar; el contenido del búfer de diagnóstico se mantiene incluso tras un borrado completo.

Ventajas del búfer de diagnóstico

El búfer de diagnóstico ofrece las ventajas siguientes:

- En caso de parada de la instalación, se puede evaluar el último evento antes del "STOP" y encontrar la causa del mismo.
- Esto permite detectar con más rapidez las causas de errores, con lo que se aumenta la disponibilidad de la instalación.
- Es posible evaluar el comportamiento dinámico de la instalación y, seguidamente, optimizarlo.

Organización del búfer de diagnóstico

El búfer de diagnóstico está diseñado como búfer en anillo. El número máximo de entradas depende del módulo; para la CPU 314 S7-300, por ejemplo, son 100 entradas. Cuando se alcanza este número máximo de entradas, cada nuevo evento de diagnóstico borra la entrada más antigua. Esto hace que el resto de las entradas también se desplacen una posición. De esta forma la entrada de diagnóstico más reciente siempre ocupa la primera posición.

El número de entradas del búfer de diagnóstico que se muestran depende del módulo y de su actual estado operativo. En determinadas CPUs se puede parametrizar la longitud del búfer de diagnóstico.

Evaluación del búfer de diagnóstico

Existen las posibilidades siguientes para acceder al contenido del búfer de diagnóstico:

- Online con la vista de diagnóstico
- En el programa de usuario con SFC 51 "RDSYSST".

La CPU está configurada de tal forma que la última entrada de diagnóstico antes de pasar del estado "RUN" a "STOP" se envía automáticamente a un dispositivo de observación (PG, OP o TD). Esto permite encontrar y solucionar más rápidamente el problema que ha causado el cambio de estado operativo. Es posible modificar este ajuste predeterminado.

Nota

Para poder utilizar adecuadamente las indicaciones de tiempo de las entradas del búfer de diagnóstico, hay que ajustar y comprobar de vez en cuando la fecha y la hora del módulo.

Temporizadores y contadores

Definición

Los temporizadores y contadores son áreas de memoria de la memoria de sistema. Las funciones del temporizador o contador, como el retardo al conectar o el valor inicial del contador, se definen en el programa de usuario. El número de contadores y temporizadores disponibles depende de la CPU.

Nota

Tenga en cuenta las indicaciones siguientes sobre los temporizadores:

- Si en un programa de usuario se utilizan más temporizadores o contadores de los que permite la CPU, se notificará un error sincrónico y se iniciará el bloque de organización "OB 121".

Acceso al área de datos de la periferia

Área de datos de la periferia

Estructura del área de datos de la periferia

Área de datos de la periferia

El área de datos de la periferia se subdivide en:

- Datos útiles
- Datos de diagnóstico y de parámetros

Las dos áreas tienen un área de entrada con acceso de sólo lectura y un área de salida con acceso de sólo escritura.

Datos útiles

Definición

Los datos de usuario pueden ser:

- Señales digitales y analógicas de entrada y salida de módulos de señales
- Información de control y estado de los módulos de función
- Información para conexiones punto a punto y acoplamientos de bus de módulos de comunicación.

Acceso a los datos de usuario

En los módulos de señales digitales, los datos de usuario se direccionan con la dirección de byte del área de entrada o salida.

En los módulos de señales analógicas, el direccionamiento se efectúa con la dirección de palabra del área de entrada o salida.

Se puede acceder a los datos de usuario mediante comandos de carga y transferencia o funciones de comunicación (manejo y visualización, M+V), o con la transferencia de la memoria imagen de proceso.

Con comandos de carga y transferencia se puede alcanzar una coherencia de datos de 1, 2 y 4 bytes. Los datos de la memoria imagen de proceso siempre son coherentes.

Si se desea una coherencia de datos de 3 ó de más de 4 bytes, se debe utilizar la instrucción "DPRD_DAT" (SFC 14) o "DPWR_DAT" (SFC 15).

Para transferir 4 bytes de forma coherente y sin modificación, se utiliza la instrucción "Transferir palabra doble". Si se utiliza cuatro veces la instrucción "Transferir byte de entrada", es posible que se pierda la coherencia de los bytes, lo que puede tener efectos no deseados.

Consulte también

DPRD_DAT: Leer datos coherentes de un esclavo DP normalizado (Página 6776)

DPWR_DAT: Escribir datos coherentes de un esclavo DP normalizado (Página 6778)

Datos de diagnóstico y de parámetros

Registros de los datos de diagnóstico y de parámetros

Los datos de diagnóstico y de parámetros de un módulo no se pueden direccionar por separado, sino sólo agrupados como registros completos. Los datos de diagnóstico y de parámetros se transfieren de forma coherente.

Los datos de diagnóstico y de parámetros se direccionan con la dirección inicial del módulo en cuestión y el número de registro. Los registros se dividen en registros de datos de entrada y de salida. Los registros de entrada sólo se pueden leer; los registros de salida sólo se pueden escribir. Se puede acceder a los registros con las funciones del sistema o de comunicación.

La tabla siguiente muestra la asignación de los registros a los datos de diagnóstico y de parámetros:

Datos	Descripción
Datos de diagnóstico	En los módulos diagnosticables, al leer los registros 0 y 1 se obtienen los datos de diagnóstico del módulo.
Datos de parámetros	En los módulos parametrizables, al escribir los registros se transfieren los parámetros del módulo.

Acceder a los registros de un módulo

Instrucciones para el acceso a los registros

Para cambiar los parámetros de los módulos parametrizables y leer la información de diagnóstico de los módulos diagnosticables, se puede utilizar la información de los registros de un módulo.

La tabla siguiente muestra con qué instrucciones se puede acceder a los registros:

Instrucción	Aplicación
Parametrizar módulos	
WR_PARM	Transferir los parámetros modificables (registro 1) al módulo de señales direccionado
WR_DPARM	Transferir registros individuales de parámetros desde los SDBs 100 a 129 al módulo de señales direccionado
PARM_MOD	Transferir todos los parámetros desde los SDBs 100 a 129 al módulo de señales direccionado
WR_REC	Transferir un registro cualquiera al módulo de señales direccionado
Leer información de diagnóstico	
RD_REC	Leer los datos de diagnóstico

Nota

Si se ha configurado un esclavo DPV1 con un archivo GSD (GSD a partir de rev. 3) y la interfaz DP del maestro DP (CPU S7-400) está configurada como "Compatible S7", en el programa de usuario no se podrán leer ni escribir registros de los módulos de E/S ni con la instrucción "WR_REC/RD_REC" ni con la instrucción "WRREC/RDREC". En este caso, el maestro DP direcciona el slot incorrecto, es decir, el slot configurado +3. Para evitar este problema, cambie el ajuste de la interfaz del DP maestro a "DPV1".

Dirección inicial de un módulo

La dirección inicial de un módulo es la dirección de byte más baja de un módulo. Representa la dirección inicial del área de datos útiles del módulo y, en muchos casos, se utiliza como sustituto de todo el módulo.

Cuando, por ejemplo, se producen alarmas de proceso, de diagnóstico, de extracción/inserción o fallos de alimentación, la dirección inicial del módulo se registra en la información de arranque del correspondiente bloque de organización e identifica así el módulo que ha dado lugar a la alarma.

Configurar las direcciones iniciales de los módulos

La asignación entre las direcciones utilizadas en el programa de usuario y los módulos se realiza mediante configuración de los módulos.

Las direcciones iniciales asignadas automáticamente al colocar los módulos en el rack se pueden modificar en las propiedades del módulo (grupo "Direcciones").

Área de datos útiles

Acceso a las direcciones de periferia

Datos útiles en el área de operandos Periferia

Los datos útiles representan el área de operandos Periferia. Existen las siguientes áreas:

- Entradas de periferia PI
- Salidas de periferia PQ

Si los datos útiles están en la memoria imagen de proceso, la CPU acepta automáticamente el intercambio de datos entre el módulo y el área de la memoria imagen de proceso al actualizarse las memorias imagen de proceso.

Si los datos útiles no están en la memoria imagen de proceso, se accede a estos datos en el programa de usuario mediante acceso directo a la periferia (p. ej. L PEB 500).

Las direcciones de los módulos de entrada y salida analógica no suelen estar en el área de la imagen de proceso. No obstante, las direcciones de los módulos analógicos se pueden colocar en esta área.

Consulte también

Memorias imagen de proceso (área de memoria de las entradas y salidas) (Página 1809)

Acceso directo a la periferia (áreas de memoria PE y PA) (Página 1810)

Áreas de direcciones

Áreas de direcciones de la periferia

Las entradas y salidas disponen de su propia área de direcciones. Por eso, la dirección de un área de periferia, además de la indicación de byte o palabra, debe identificar también si se trata de entradas ("I") o salidas ("Q").

La siguiente tabla muestra las áreas de direcciones de periferia disponibles:

Área de operandos	Acceso vía unidades del tamaño siguiente:	Notación S7
Área de periferia: Entradas	Byte de entrada de periferia Palabra de entrada de periferia Palabra doble de entrada de periferia	PIB PIW PID
Área de periferia: Salidas	Byte de salida de periferia Palabra de salida de periferia Palabra doble de salida de periferia	PQB PQW PQD

Consulte también

Acceso directo a la periferia (áreas de memoria PE y PA) (Página 1810)

Configurar el comportamiento operativo

Introducción a la configuración del comportamiento operativo

Modificar las propiedades de los sistemas de automatización

Las propiedades de los sistemas de automatización que no vienen predeterminadas se pueden modificar de la siguiente forma:

- Configurando parámetros de sistema
- Utilizando funciones de sistema (SFC) en instrucciones avanzadas

Modificar parámetros del módulo

Configuración estándar

Todos los módulos parametrizables están ajustados de fábrica a los valores adecuados para las aplicaciones estándar. Estos valores estándar permiten utilizar los módulos sin necesidad de realizar más ajustes.

No obstante, es posible parametrizar el comportamiento y las propiedades de los módulos para adaptarlos a las necesidades y particularidades de la instalación. Los módulos

parametrizables son CPUs, FMs, CPs y algunos módulos de entrada y salida analógica y módulos de entrada y salida digital.

Parámetros del módulo

Para una mejor comprensión, los parámetros del módulo se agrupan, para las CPUs, en los siguientes grupos:

- Arranque
- Ciclo
- Interfaz PROFINET u otras interfaces
- Diagnóstico
- Remanencia
- Marcas de ciclo
- Alarmas
- Protección
- Reloj
- Recursos de conexión

Ajustar y cargar parámetros

Es posible establecer los ajustes de los parámetros de los módulos. Cuando se guarda una configuración de dispositivos con sus parámetros, se generan datos de sistema que se cargan en la CPU junto con el programa de usuario y, desde allí, se transfieren a los módulos correspondientes al arrancar.

Parametrizar con instrucciones

Aparte de la parametrización, también es posible modificar los parámetros de los módulos desde el programa con las instrucciones avanzadas.

La tabla siguiente muestra qué instrucciones pueden modificar qué parámetros de módulos:

Función de sistema	Aplicación
WR_PARM (Página 6882)	Transferir los parámetros modificables (registro 1) al módulo de señales direccionado.
WR_DPARM (Página 6883)	Transferir juegos individuales de parámetros desde los SDB correspondientes al módulo de señales direccionado.
PARM_MOD (Página 6877)	Transferir todos los parámetros desde los SDBs correspondientes al módulo de señales direccionado.
WR_REC (Página 6774)	Transferir un registro cualquiera al módulo de señales direccionado

Parámetros generales de la CPU

Propiedades generales de los módulos

El grupo de parámetros "General" informa acerca del tipo y la ubicación del módulo y, en los módulos con interfaces, ofrece información relevante para la red, como p. ej. la dirección de dispositivo.

Nombre abreviado, referencia/firmware, nombre, identificador de instalación, identificador de situación

El nombre abreviado, la información del mismo y la referencia/firmware son los mismos que aparecen en el catálogo de hardware.

En el campo "Nombre" aparece el nombre del módulo, que se puede modificar según las especificaciones del usuario. Si se modifica el nombre, el nuevo aparece también en la vista de dispositivos o en la de redes.

Las CPU ofrecen la posibilidad de establecer una identificación específica de la instalación. Con la identificación de la instalación se identifican unívocamente las partes de la instalación siguiendo criterios funcionales. Tiene una estructura jerárquica (según IEC 1346-1). También resulta posible indicar una identificación de situación de hasta 32 caracteres de longitud. Forma parte de la designación de referencia que identifica la localización precisa de, por ejemplo, un punto de instrumentación y control o un grupo de ellos dentro de una instalación de ingeniería de procesos (de forma similar a una calle en un mapa).

La identificación introducida en el campo "ID de instalación" o "ID de situación" se puede evaluar en el programa de usuario de la CPU. Esta posibilidad se puede utilizar para ejecutar el programa de usuario en función de la ID de la instalación o la ID de situación.

Comentario

Todos los módulos cuentan con un campo de comentario. En este campo puede introducirse p. ej. la finalidad de uso del módulo.

Arranque

Parámetros de arranque

El comportamiento de arranque de la CPU

A continuación se describen los parámetros de las CPUs S7 que determinan el comportamiento de arranque.

Arranque si la configuración real difiere de la teórica

La configuración teórica se representa mediante la configuración cargada en la CPU (bloques de datos de sistema). La configuración real es la que existe realmente en el sistema de automatización.

Si la casilla de verificación "Arranque si la configuración real difiere de la teórica" está desactivada, la CPU pasa al estado "STOP" en los siguientes casos:

- Si no hay, al menos, un módulo en el slot configurado.
- Si en el slot configurado hay un módulo de otro tipo, p. ej. durante la puesta en servicio.
- Si falta la periferia descentralizada.

Si aparte de los módulos definidos en la configuración hay también otros módulos (centralizados o descentralizados), estos no se tienen en cuenta al comparar la configuración real/teórica.

Si la casilla de verificación "Arranque si la configuración real difiere de la teórica" está activada, la CPU arrancará también aunque se den las circunstancias arriba indicadas.

La CPU no comprueba la configuración de la periferia (módulos centralizados y descentralizados, racks centralizados y esclavos DP). Una excepción a esto son las interfaces PROFIBUS DP. Debido a su estrecha relación con la CPU, los módulos de este tipo que se hayan configurado siempre deben estar insertados para que la CPU pueda arrancar.

De forma predeterminada, en la CPU está activada la casilla "Arranque si la configuración real difiere de la teórica", de modo que arranca aunque haya diferencias entre la configuración real y la teórica.

Tiempo de vigilancia para la señal "ready" de los módulos

Duración máxima del aviso de fin de todos los módulos configurados después de la conexión (POWER ON). Si, una vez transcurrido este tiempo de vigilancia, los módulos no han enviado una señal "ready" a la CPU, significa que la configuración real difiere de la teórica. La reacción de la CPU estará entonces determinada por el parámetro "Arranque si la configuración real difiere de la teórica".

En el caso, por ejemplo, de un montaje en varias líneas o de dispositivos conectados a la CPU a través de la interfaz maestra DP, durante este tiempo de vigilancia se pueden conectar todas las fuentes de alimentación sin necesidad de seguir el orden de conexión.

Tiempo de vigilancia para la transferencia de parámetros a los módulos

Duración máxima del "reparto" de los parámetros a los módulos parametrizables (empieza a contar después de recibirse la señal "ready" de los módulos). En las CPUs con interfaz maestra DP, este parámetro también define la vigilancia del tiempo de arranque de los esclavos DP. Esto significa que los esclavos DP deben haber arrancado y haber sido parametrizados por la CPU (como maestro DP) en el tiempo definido.

Si, una vez transcurrido este tiempo de vigilancia, no se han parametrizado todos los módulos/esclavos DP, significa que la configuración real difiere de la teórica. La reacción de la CPU estará entonces determinada por el parámetro "Arranque si la configuración real difiere de la teórica".

Parámetros de arranque sólo para CPUs S7-400

Inicializar salidas al efectuar un rearranque

Inicializar salidas al efectuar un rearranque

En un rearranque, las CPUs del S7-400 pueden inicializar las salidas (y, con ello, la memoria imagen de proceso de las salidas PAA).

Consulte también

Rearranque normal (Página 1705)

Bloquear rearranque en caso de arranque por intervención

Bloquear el rearranque en caso de arranque por intervención (p. ej., de la programadora) o por petición de comunicación (p. ej., de dispositivos MPI)

En el S7-400 existe la posibilidad de bloquear el tipo de arranque "Rearranque" en caso de que se active por intervención de la programadora o por una petición de comunicación. Así, el rearranque sólo se puede efectuar "in situ".

- Active esta casilla de verificación si el tipo de arranque activado por intervención o por petición de comunicación se debe limitar al rearranque completo (=rearranque completo manual) o al arranque en frío.
- Si la casilla de verificación está desactivada, todos los tipos de arranque resultan posibles.

Un rearranque por intervención o petición de comunicación (=rearranque normal manual) se activa cuando se cumplen los requisitos siguientes:

- La casilla de verificación "Bloquear rearranque..." no ha sido activada y esta parametrización se ha transferido a la CPU
- No hay ningún impedimento para un rearranque (batería en estado correcto, ningún cambio en el programa de usuario en estado STOP, ...)
- En la programadora se ha seleccionado el estado operativo de rearranque, o bien otro dispositivo provoca un rearranque mediante una petición de comunicación.

Consulte también

Rearranque normal (Página 1705)

Comportamiento cíclico

Ajustes para el comportamiento cíclico

Actualizar cíclicamente la memoria imagen de proceso del OB1

Si esta opción está activada, la memoria imagen de proceso del OB1 se actualiza cíclicamente (ajuste predeterminado). En las CPUs S7-400 la actualización del OB1 se puede desactivar con este parámetro.

Tiempo de vigilancia del ciclo

El tiempo de vigilancia del ciclo se indica en milésimas de segundo.

Si el tiempo de ciclo excede el tiempo de vigilancia del ciclo, la CPU pasa al estado operativo "STOP". Para evitar que pase a "STOP", se puede programar el bloque de organización OB 80.

Las posibles causas de que se rebase el tiempo son:

- Procesos de comunicación
- Acumulación de eventos de alarma
- Error en el programa de la CPU.

Control de los procesos de comunicación que cargan el ciclo

Con el parámetro "Carga del ciclo por comunicación" se puede controlar hasta cierto punto la duración de los procesos de comunicación que también alargan el tiempo de ciclo. Los procesos de comunicación pueden ser, p. ej.:

- Transferencia de datos a otra CPU
- Carga de bloques (iniciada desde la programadora)

Las funciones de test con la programadora apenas se ven afectadas por este parámetro, y pueden alargar considerablemente el tiempo de ciclo. El tiempo disponible para las funciones de test se puede limitar en el modo Proceso.

Funcionamiento del parámetro

El sistema operativo de la CPU siempre tiene a disposición de las funciones de comunicación el porcentaje de potencia de procesamiento total de la CPU que se haya configurado (técnica de segmentos de tiempo). Si para la comunicación no se necesita tanta potencia de procesamiento, esta quedará disponible para otros procesos.

Repercusión en el tiempo de ciclo real

Si no hay otros eventos asíncronos, el tiempo de ciclo del OB 1 se alarga en un factor que se calcula con la siguiente fórmula:

$$\frac{100}{100 - \text{“Carga de ciclo por comunicación“}}$$

Ajustar el tamaño de la memoria imagen de proceso

En las CPUs con un tamaño variable de la memoria imagen de proceso se puede definir el tamaño de la misma (en bytes). El área de la memoria imagen de proceso empieza siempre por el byte de entrada o salida 0.

En las CPUs S7-300 que soportan esta función, al crear el programa hay que tener en cuenta las particularidades para el acceso al área de la memoria imagen de proceso. Estas particularidades pueden consultarse en la documentación de la CPU.

Ejemplo: Actualmente, la configuración variable de la memoria imagen de proceso sólo afecta a la actualización de la memoria imagen de proceso en el punto de control del ciclo. La memoria imagen de proceso de las entradas sólo se actualiza hasta el tamaño que se ha configurado para ella, y a los módulos de salida sólo se les envía el área de la memoria imagen de proceso de las salidas que se ha parametrizado. El acceso del programa de usuario al área comprendida entre el tamaño configurado para la memoria imagen de proceso y el tamaño máximo de la misma no provoca un error de acceso sincrónico.

Consulte también

Ejemplos de carga del ciclo por comunicación (Página 1741)

Ejemplos de carga del ciclo por comunicación

A continuación hay dos ejemplos de cómo la comunicación puede afectar a la carga del ciclo.

Ejemplo de carga del ciclo por comunicación sin eventos asíncronos adicionales

Si se ha configurado la carga del ciclo por la comunicación al 50%, puede producirse una duplicación del tiempo de ciclo del OB 1.

Además, el tiempo de ciclo del OB 1 también se ve afectado por eventos asíncronos (p. ej., alarmas de proceso o cíclicas). La prolongación del tiempo de ciclo con el ratio de comunicación hace que, desde un punto de vista estadístico, también se produzcan más eventos asíncronos dentro de un ciclo del OB 1. Esto prolonga aún más el ciclo del OB 1. Esta prolongación depende de cuántos eventos aparezcan por cada ciclo del OB 1 y de cuánto dure su procesamiento.

Ejemplo de carga del ciclo por comunicación con eventos asíncronos adicionales

Con un tiempo neto de ejecución del OB 1 de 500 ms, una carga de comunicación del 50% puede hacer que el tiempo real del ciclo llegue a los 1000 ms (siempre que la CPU tenga suficientes peticiones de comunicación que procesar). Si, paralelamente, se procesa una alarma cíclica cada 100 ms con un tiempo de procesamiento de 20 ms, dicha alarma sin carga de comunicación prolongaría el ciclo en un total de 5*20 ms = 100 ms, es decir, el tiempo de ciclo real sería de 600 ms. Como una alarma cíclica también interrumpe la comunicación, con

una carga de comunicación del 50% la alarma repercute en el tiempo de ciclo con 10 * 20 ms, es decir, que en este caso el tiempo de ciclo real no sería de 1000 ms, sino de 1200 ms.

Nota

Verifique las repercusiones de un cambio de valor del parámetro "Carga de ciclo por comunicación" en el funcionamiento de la instalación.

La carga de comunicación debe tomarse en consideración al ajustar el tiempo de ciclo mínimo, puesto que de lo contrario se producen errores de tiempo.

Llamada del OB 85 en caso de error de acceso a la periferia

Ajustes para la llamada del OB 85

Se puede modificar la reacción predeterminada de la CPU en caso de error de acceso a la periferia durante la actualización de la memoria imagen de proceso por parte del sistema.

Recomendación: Para activar la llamada del OB 85 en caso de error de acceso a la periferia, seleccione la opción "Sólo para errores entrantes y salientes" para que no aumente el tiempo de ciclo de la CPU por llamadas repetidas del OB 85.

Las CPUs S7-300 están configuradas de forma estándar de tal forma que los errores de acceso a la periferia no llamen ningún OB 85 ni generen entradas en el búfer de diagnóstico.

Comportamiento remanente

Ajustes de remanencia para marcas, temporizadores y contadores

En la parametrización de la remanencia se determina qué áreas de memoria de la CPU se conservarán en caso de corte de la alimentación eléctrica o transición del estado operativo "STOP" a "RUN".

Número de bytes de marcas a partir de MB 0

Introduzca el número de bytes de marcas remanentes a partir del byte de marca 0.

Ejemplo

Si indica "2", los bytes de marca MB 0 y MB 1 serán remanentes.

Número de temporizadores S7 desde T0

Introduzca el número de temporizadores S7 remanentes a partir del T 0.

Cada temporizador S7 ocupa 2 bytes.

Ejemplo

Si indica "2", los temporizadores S7 T 0 y T 1 serán remanentes.

Número de contadores S7 a partir de C0

Introduzca el número de contadores S7 remanentes (contadores (C)) a partir de C0.

Ejemplo

Si indica "2", los contadores S7 C0 y C1 serán remanentes.

Ajustes de remanencia para bloques de datos

Función

En las propiedades de remanencia se puede determinar qué áreas de memoria se conservarán tras un corte de alimentación eléctrica y tras una transición de "STOP" a "RUN", incluso aunque no haya respaldo por pila en la memoria de respaldo de la CPU. Si se pueden o no configurar áreas remanentes, y cuántas, depende de la CPU.

Si hay una pila de respaldo insertada, los bloques de datos son siempre remanentes.

Áreas remanentes (sólo relevante para S7-300 sin respaldo por pila o sin Micro Memory Card)

Las áreas remanentes son áreas de datos de bloques de datos.

Para que las áreas de datos de la memoria de trabajo de la CPU sean remanentes, indique lo siguiente:

- N.º DB
  Números de los bloques de datos que deben ser remanentes.
- Dirección de byte (dirección inicial con la longitud del byte de datos)
  Direcciones de byte de los bloques de datos que deben ser remanentes.
- N.º de bytes
  Cantidad de bytes de datos remanentes a partir de la dirección de byte. Si, p. ej., indica "1", el byte de datos "0" será remanente.

Posibilidades de ajuste para alarmas

Ajustes para alarmas de proceso

Alarmas de proceso

Cuanta más alta sea la prioridad elegida, mayor será la prioridad de procesamiento de la correspondiente alarma de proceso. Con la prioridad "0" se deselecciona el OB de alarma. En las CPUs S7-300 no se puede modificar la prioridad predeterminada.

En las CPUs que soportan memorias imagen parcial de proceso se pueden asignar memorias imagen parcial de proceso a una alarma de proceso. Al iniciar el OB de alarma de proceso se actualiza la memoria imagen parcial de proceso asignada, de forma que el OB trabaja con valores de proceso actualizados. En el ajuste "---" no se asigna ninguna memoria imagen de proceso.

Consulte también

OBs de alarma de proceso (de OB 40 a OB 47) (Página 1830)

Ajustes para alarmas de retardo

Alarmas de retardo

Con OBs de alarma de retardo se pueden procesar partes del programa de usuario con retardo de tiempo.

Cuanto más alta es la prioridad elegida, tanto mayor es la prioridad con la que se procesa la alarma de retardo correspondiente. Con la prioridad "0" se deselecciona el OB de alarma. En las CPUs S7-300 no se puede modificar la prioridad predeterminada.

En las CPUs que soportan memorias imagen parcial de proceso se pueden asignar memorias imagen parcial de proceso a una alarma de retardo. Al iniciar el OB de alarma de proceso se actualiza la memoria imagen parcial de proceso asignada, de forma que el OB trabaja con valores de proceso actualizados. En el ajuste "---" no se asigna ninguna memoria imagen de proceso.

Las alarmas de retardo se manejan con las instrucciones siguientes:

- SRT_DINT Iniciar alarma de retardo
- CAN_DINT Anular alarma de retardo
- QRY_DINT Consultar el estado de una alarma de retardo

Consulte también

OBs de alarma de retardo (de OB 20 a OB 23) (Página 1827)

Ajustes para alarmas de error asíncrono

Alarmas de error asíncrono

En las CPUs S7-300 no se puede modificar la prioridad predeterminada.

La prioridad de los OB de error asíncrono se puede modificar para que la CPU siga en el estado operativo "RUN" también en las siguientes circunstancias:

- Frecuentemente se producen errores asíncronos que disparan el mismo OB de alarma (p. ej. alarma de diagnóstico, llamada de OB 82).
- Además, se produce otro evento de error asíncrono que dispara otro OB de alarma (p. ej. error de ejecución del programa, llamada de OB 85).
- Sólo debe depositarse un mensaje (en caso de fallo de estación, p. ej., cuál es la estación que falla). No es necesaria una reacción del proceso. En este caso se puede asignar una menor prioridad al OB 86 para no aumentar el tiempo de reacción a otras alarmas.

Ejemplo de necesidad de modificar la prioridad

Si se corta un cable de una línea de señales, todos los módulos de señales afectados por la rotura del cable podrían emitir una alarma de diagnóstico (llamada al OB 82) - en ese caso, el búfer para la información de arranque del OB 82 se llenaría enseguida.

Para que la CPU no pase al estado operativo "STOP" si se produce un evento que llama otro OB de error asíncrono (p. ej. error de ejecución de programa, OB 85), es posible asignar al OB 85 una prioridad mayor que al OB 82, por ejemplo.

Consulte también

OB de error de fuente de alimentación (OB 81) (Página 1848)

OB de alarma de diagnóstico (OB 82) (Página 1851)

OB de cambio de módulo (OB 83) (Página 1854)

OB de error de hardware en CPU (OB 84) (Página 1858)

OB de error de ejecución del programa (OB 85) (Página 1859)

OB de fallo del rack (OB 86) (Página 1864)

OB de error de comunicación (OB 87) (Página 1871)

Ajustes para alarmas horarias

Prioridad

La prioridad de las alarmas horarias determina el comportamiento de interrupción de los OBs de alarma. Actualmente no se puede modificar la prioridad en el S7-300.

Activo

Active la casilla de verificación del OB de alarma horaria si desea que se inicie automáticamente en caso de rearranque. O bien active la alarma horaria en el programa de usuario con el parámetro OB_NR en la instrucción ACT_TINT.

Ejecución

Seleccione con qué frecuencia deben ejecutarse las alarmas.

Están disponibles intervalos desde **cada minuto** hasta **anual**.

Los intervalos se basan en los ajustes de la fecha de inicio y la hora.

Nota

Para la **Periodicidad** = **Último del mes** no es relevante la fecha de inicio (ya que será siempre el último día del mes).

Hora de inicio

Indique en qué fecha y hora debe ejecutarse la alarma por primera vez.

Ejemplo

Fecha de inicio	Hora
30.03.2006	10:15

En función del parámetro "Periodicidad", la CPU genera otras alarmas horarias en intervalos determinados. La hora de inicio se refiere siempre a la hora patrón de la CPU.

Nota

Para la periodicidad "Mensual" no se puede elegir como fecha de inicio el día 29, 30 ó 31 del mes, ya que la alarma horaria no saltaría en el mes de febrero.

En lugar de ello, seleccione la Periodicidad "Último del mes". En este caso no es relevante la fecha de inicio (ya que será siempre el último día del mes).

Memoria imagen parcial de proceso

En las CPUs que soportan memorias imagen parcial de proceso se pueden asignar memorias imagen parcial de proceso a una alarma horaria. Al iniciar el OB de alarma se actualiza la memoria imagen parcial de proceso asignada, de forma que el OB trabaja con valores de proceso actualizados. En el ajuste "---" no se asigna ninguna memoria imagen parcial de proceso.

Caso particular: Se ha configurado para una única vez y el momento de inicio queda en el pasado

La configuración es la siguiente:

- Periodicidad: Una sola vez
- Activada: Sí
- Fecha de inicio/Hora: Queda en el pasado (según el reloj de tiempo real de la CPU)

Comportamiento de la CPU: Tras un arranque en frío o rearranque completo (en caliente) el sistema operativo **llama una vez** al correspondiente OB de alarma horaria.

Consulte también

OBs de alarma horaria (de OB 10 a OB 17) (Página 1823)

Ajustes para alarmas cíclicas

Una alarma cíclica es una señal periódica que provoca la llamada automática de un OB de alarma cíclica. A continuación se describen los posibles ajustes.

Prioridad

La prioridad determina el comportamiento de interrupción de los OBs de alarma. En las CPUs S7-300 no se puede modificar.

Ejecución

Indique (p. ej. milésimas de segundo) los intervalos de tiempo en los que deben procesarse los OB de alarma cíclica utilizando las unidades indicadas.

El momento inicial del ciclo es el cambio de estado operativo de "STOP" a "RUN".

Desfase

Si el desfase se puede configurar: Introduzca, en milésimas de segundo (ms), el tiempo que debe retardarse el momento de ejecución real de la alarma cíclica.

Nota

Un desfase tiene sentido si hay activas varias alarmas cíclicas. Sin desfase, las alarmas cíclicas se ejecutarían al mismo tiempo y se procesarían una tras otra en función de su prioridad.

Con el desfase se puede repartir el tiempo de procesamiento de las alarmas cíclicas más allá del ciclo.

Unidad

Se refiere a los parámetros "Periodicidad" y "Desfase". Sólo se puede modificar en algunas CPUs especiales que soportan un ajuste microsegundo a microsegundo de las alarmas cíclicas.

Memoria imagen parcial de proceso

En las CPUs que soportan memorias imagen parcial de proceso se pueden asignar memorias imagen parcial de proceso a una alarma cíclica. Al iniciar el OB de alarma cíclica se actualiza la memoria imagen parcial de proceso asignada, de forma que el OB trabaja con valores de proceso actualizados. En el ajuste "---" no se asigna ninguna memoria imagen de proceso.

Consulte también

OBs de alarma cíclica (de OB 30 a OB 38) (Página 1828)

Ajustes para alarmas DPV1

Alarmas para DPV1

Las alarmas las puede poner un esclavo DPV1 y garantizan el tratamiento del evento desencadenante de la alarma en la CPU maestra. También en el estado operativo "STOP" se evalúan en la CPU los datos de la alarma (actualización del búfer de diagnóstico y del estado del módulo). No obstante, en "STOP" no se procesa el OB.

Aparte de las alarmas conocidas de SIMATIC (p. ej. alarma de diagnóstico en ET 200M), también se soportan otras alarmas, como la alarma de estado, la de actualización y la del fabricante.

Consulte también

OB de alarma de estado (OB 55) (Página 1832)

OB de alarma de actualización (OB 56) (Página 1836)

OB de alarmas específicas del fabricante (OB 57) (Página 1840)

Ajustes para alarmas de sincronismo

Propiedades de las alarmas de sincronismo

Los OBs de alarma de sincronismo se procesan sincrónicamente con el ciclo equidistante.

Número de sistema maestro DP

Seleccione el sistema maestro DP cuyos esclavos DP deban asignarse al OB de alarma. Las entradas y salidas de dichos esclavos podrán sincronizarse con el ciclo equidistante y el programa de usuario en el OB de alarma.

Memoria(s) imagen parcial del proceso

Introduzca aquí el número de memorias imagen parcial de proceso a las que debe acceder el OB de alarma. A través de las memorias imagen parcial de proceso se actualiza la periferia descentralizada asignada de forma sincrónica con el ciclo equidistante.

Las memorias imagen de proceso parciales asignadas deben actualizarse en el OB de alarma de sincronismo con la instrucción "SYNC_PI" y "SYNC_PO". La instrucción "SYNC_PI" actualiza la memoria imagen de proceso parcial de las entradas, mientras que "SYNC_PO" actualiza la memoria imagen de proceso parcial de las salidas.

Si las direcciones de la periferia descentralizada están asignadas a distintas memorias imagen parcial de proceso, los números de estas deben indicarse separados por comas (p. ej. 1,2,3).

Tiempo de retardo

El tiempo de retardo es el tiempo entre el telegrama Global Control y el inicio del OB 61 (o del OB 62...OB 64). Durante este tiempo, el maestro DP ejecuta el intercambio de datos cíclico con los esclavos DP.

Si el tiempo de retardo = 0, el OB 6x se inicia con el telegrama Global-Control. En este caso es muy probable que un acceso a la periferia isócrona a través de "SYNC_PI" al principio del OB no resulte, ya que los datos de la periferia aún no están disponibles.

Si el tiempo de retardo es mayor que el valor estándar, se "regalará" tiempo de ejecución del OB, ya que el acceso a la periferia isócrona ya sería posible antes.

Consulte también

OBs de alarma de sincronismo (de OB 61 a OB 64) (Página 1844)

Funciones de reloj

Principios básicos de las funciones de reloj

Todas las CPU S7-300/400 tienen un reloj, que puede ser de tiempo real o de software. En el sistema de automatización, el reloj puede funcionar como reloj maestro o como esclavo con sincronización externa. Permite la utilización de alarmas horarias y contadores de horas de funcionamiento.

Formato de hora

El reloj muestra siempre la hora con una resolución mínima de 1 segundo, y la fecha con el día de la semana. En algunas CPUs también se pueden visualizar milésimas de segundo.

Ajustar y leer la hora

Ajustar la hora

La fecha y hora del reloj de la CPU se ajustan e inician con las instrucciones siguientes:

- "SET_CLK"
- "SET_CLKS", si está disponible en la CPU

Nota

Para evitar diferencias de visualización en el sistema HMI y en el reloj, ajuste el horario de invierno en la CPU o utilice una sincronización horaria centralizada.

Leer la hora

Con la instrucción "READ_CLK" o un comando de menú de la programadora se leen la fecha y hora actuales de la CPU.

Sincronización horaria

Referencia

Parametrización de las propiedades de las interfaces en los siguientes grupos de parámetros:

- Para la CPU: "Propiedades > General > Hora"
- Para la interfaz DP: "Propiedades > General > Interfaz DP > Sincronización horaria".

Parámetros del reloj

Mediante los parámetros del reloj se ajusta lo siguiente:

- Tipo de sincronización
  Las propiedades de sincronización se pueden ajustar interna (PLC) y externamente (MPI) para la CPU.
  Si está configurado el tipo de sincronización "Como maestro", se puede elegir el intervalo de tiempo para la sincronización automática.
  Si en una red hay más de un módulo con reloj, hay que determinar qué CPU actuará como maestra y cuál como esclava para la sincronización del reloj. Sólo puede haber un reloj maestro.
- Intervalo de actualización
  El intervalo de actualización define el espacio de tiempo entre las consultas de hora.
- Factor de corrección
  Con el factor de corrección se compensa una desviación del reloj cada 24 horas para la CPU.
  Se pueden introducir valores positivos o negativos en ms.
  Ejemplo: Si el reloj se atrasa 4 segundos cada 24 horas, habrá que indicar un factor de corrección de "+4000 ms".

Instrucciones para la sincronización de la hora

Para garantizar que la hora coincida en todos los módulos de la red, el programa de sistema sincroniza los relojes esclavos a intervalos regulares parametrizables. Con la instrucción "SNC_RTCB" se puede enviar la fecha y hora del reloj maestro al reloj esclavo, independientemente del intervalo de sincronización parametrizado.

Utilizar un contador de horas de funcionamiento

Definición

Un contador de horas de funcionamiento cuenta el tiempo, en total de horas de funcionamiento, durante el que está activo un recurso conectado o el tiempo que está en funcionamiento la CPU. Una CPU puede tener hasta 8 contadores de horas de funcionamiento. La numeración empieza en el 0.

En el estado operativo "STOP" se detiene el contador de horas de funcionamiento. Su valor se conserva incluso tras un borrado completo. En caso de "rearranque completo (en caliente)", el programa de usuario debe volver a poner en marcha el contador de horas de funcionamiento.

Programar el contador de horas de funcionamiento

El contador de horas de funcionamiento se puede programar con las instrucciones siguientes:

- Con la instrucción "SET_RTM" se puede ajustar el contador de horas de funcionamiento a un valor inicial.
- Con la instrucción "CTRL_RTM" se puede iniciar o detener el contador de horas de funcionamiento.
- Con la instrucción "READ_RTM" se puede leer la cantidad actual de horas de funcionamiento y el estado del contador, que puede ser "parado" o "contando".

Uso de marcas de ciclo

Marcas de ciclo

Una marca de ciclo es una marca que modifica periódicamente su estado binario en una relación 1:1 entre impulso y pausa.

En la parametrización de la marca de ciclo se determina el byte de marcas de la CPU que se convertirá en el byte de marcas de ciclo.

Uso

Las marcas de ciclo se utilizan en el programa de usuario, p. ej., para controlar indicadores luminosos con una luz intermitente o lanzar procesos periódicos, como la adquisición de un valor real.

Frecuencias posibles

Cada bit del byte de marcas de ciclo tiene asignada una frecuencia. La tabla siguiente muestra la asignación:

Bit del byte de marcas de ciclo	7	6	5	4	3	2	1	0
Duración del período (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frecuencia (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Nota

Las marcas de ciclo transcurren de forma asíncrona al ciclo de la CPU, es decir, en ciclos largos es posible que el estado de la marca de ciclo cambie varias veces.

El byte de marcas seleccionado no puede emplearse para almacenar datos en la memoria intermedia.

Configuración web

En este grupo de parámetros se puede elegir:

- Inicio del servidor web de la CPU
- Idiomas en los que se deben cargar los textos (los textos del búfer de diagnóstico o mensajes, p. ej., dependen del idioma)

Activar servidor web en este módulo

Si la casilla de verificación está activada, tras cargarse los datos de configuración se inicia el servidor web de la CPU y se puede leer información de la CPU con un navegador web.

Los ajustes restantes afectan a la actualización de las páginas web visualizadas. En caso necesario se puede ajustar el tiempo de actualización.

Permitir el acceso sólo vía HTTPS

Active esta casilla de verificación para que las páginas web del servidor web se transmitan encriptadas al navegador web.

Nota

El acceso vía HTTPS sólo funciona si se ha ajustado la hora en la CPU.

Selección de idiomas

Elija los idiomas para los textos que dependen del idioma que desea cargar en la CPU. El número de idiomas que se pueden seleccionar depende de la CPU.

Visualización y selección de idiomas específicos de la CPU

Se muestran los idiomas que se pueden ajustar para el acceso web a la CPU. Dependiendo de la CPU en cuestión, a estos "idiomas del servidor web" se les puede asignar un determinado número de idiomas de proyecto. En la CPU se cargan los textos de las entradas del búfer de diagnóstico o de los eventos en los idiomas seleccionados. Durante el funcionamiento, la información también se muestra en el navegador web en el idioma seleccionado.

Si durante el funcionamiento se selecciona en el navegador web un idioma que no se ha cargado previamente, las entradas del búfer de diagnóstico y los eventos aparecerán en código hexadecimal.

Hay que tener en cuenta que cada idioma necesita espacio de memoria en la CPU.

Administración de usuarios

Para acceder a la CPU a través del navegador web se pueden definir usuarios, concederles derechos y asignarles contraseñas.

También se pueden definir derechos para el usuario estándar que no indica su contraseña para iniciar sesión en el navegador web ("everybody").

Tablas de observación

Existe la posibilidad de que los usuarios puedan acceder a través del navegador web a las tablas de observación definidas. Seleccione las que desee entre las ya existentes en la lista desplegable y determine en la lista desplegable de la segunda columna si se debe permitir el acceso en lectura o escritura.

Discriminador de avisos

Los discriminadores de avisos seleccionados se muestran posteriormente en las páginas web de la CPU.

Se puede reducir el espacio requerido en memoria por los SDBs web seleccionando únicamente los discriminadores que se necesiten.

Los discriminadores se ajustan en las propiedades de la CPU "Atributos de aviso", en "Diagnóstico de sistema" en la ventana de inspección.

Páginas web definidas por el usuario

El apartado "Páginas web definidas por el usuario" permite cargar en la CPU páginas web propias y poner así a disposición las aplicaciones web propias a través del navegador web.

Consulte también

Información importante sobre el servidor web (Página 936)

Crear y cargar páginas web definidas por el usuario (Página 946)

Reservar recursos de comunicación

Funcionamiento de la asignación de recursos de conexión

Toda conexión de comunicación necesita recursos de conexión en la CPU S7 mientras dure esa comunicación. Dependiendo de los datos técnicos, todas las CPUs S7 tienen un determinado número de recursos de conexión que pueden ser ocupados por distintos servicios de comunicación (comunicación PG/OP, comunicación S7 o comunicación básica S7).

Asignación de los recursos de conexión en orden temporal

Cuando los servicios de comunicación inician sesión, se les asignan los recursos de conexión en el orden en que hayan iniciado sesión.

Asignación manual de los recursos de conexión

Si no se desea que la asignación de los recursos de conexión dependa únicamente del orden en el que ha iniciado sesión cada servicio de comunicación, es posible reservar recursos de conexión para los siguientes servicios:

- Comunicación PG y OP
- Comunicación básica S7

De forma predeterminada, la comunicación PG/OP tiene reservado al menos un recurso de conexión. No se admiten valores más pequeños.

Otros servicios de comunicación, como la comunicación S7 con las instrucciones avanzadas "PUT (Página 7680)" y "GET (Página 7677)", no pueden utilizar estos recursos aunque hayan establecido la conexión antes. En lugar de eso, ocuparán los recursos de conexión que aún queden disponibles y no estén reservados expresamente para un servicio.

Información sobre el número máximo y ya ocupado de recursos de conexión

En las propiedades de la CPU relativas a la reserva de recursos de conexión se muestra, además, la siguiente información (de sólo lectura):

- Número de recursos ya asignados para la comunicación S7. Se trata de recursos para conexiones S7 unilaterales desde otra CPU a esta, p. ej., o de conexiones S7 bilaterales. Estas conexiones ya se han configurado en la vista de redes.
- Número máximo de recursos de conexión para esta CPU.

Test, puesta en servicio y routing

Test, puesta en servicio y routing

Active la casilla de verificación si desea realizar con frecuencia funciones de programadora necesarias para la puesta en marcha y el test, por ejemplo, utilizando esta interfaz.

Si se activa la opción "Test, puesta en servicio y routing", la interfaz se convertirá en el dispositivo activo en PROFIBUS (es decir, la interfaz participa en la rotación del token de PROFIBUS). En este caso serán posibles las funciones siguientes:

- Programar (p. ej. cargar)
- Probar (estado/forzar)
- Routing S7 (esclavo I como transición de red)

El tiempo de ciclo puede prolongarse. En aplicaciones críticas y si no se necesitan el routing S7 ni la funcionalidad de cliente para la comunicación, no debería activarse esta opción.

Si la casilla de verificación "Test, puesta en servicio y routing" no está activada, la interfaz sólo trabaja como servidor de servicios de comunicación.

La funcionalidad de servidor significa que sólo el interlocutor (p. ej. PG, OP o sistema de automatización) puede iniciar la comunicación. Las particularidades siguientes deben tenerse en cuenta si la opción "Test, puesta en servicio y routing" no está activada:

- No es posible el routing S7
- La programación y el test son posibles pero lentamente (sin prolongación del tiempo de ciclo)

Ajustes del diagnóstico del sistema

Con el sistema de diagnóstico se puede detectar, valorar y notificar un fallo del sistema de automatización.

Ejemplos de errores

- Fallo en el programa de la CPU
- Fallo de módulos
- Rotura de cables de encoders y actuadores

Notificar causa del STOP

Esta opción se activa para que, en caso de pasar a "STOP", la CPU notifique la causa a los visualizadores que han iniciado sesión (p. ej. PG u OP).

El aviso también se registra en el búfer de diagnóstico.

Notificación controlada por acuse de la instrucción avanzada ALARM

Este parámetro es relevante para las CPUs utilizadas para Process Control Systems. Si esta función está activada, las instrucciones avanzadas "ALARM (Página 6930)", "ALARM_8

(Página 6923)" y "ALARM_8P (Página 6925)" (SFBs 33 a 35) no volverán a notificar un cambio de señal hasta que se acuse el anterior (es decir, el anterior mensaje entrante). De esta forma se evita que los mensajes se acumulen y no se puedan procesar.

Posibilidades de ajuste del nivel de protección

Nivel de protección

A continuación se explica cómo se utilizan los distintos niveles de protección de las CPUs S7 y cómo influye el tipo de CPU S7 sobre la parametrización y utilización de cada nivel de protección.

Comportamiento predeterminado

En el comportamiento predeterminado del nivel de protección no se asignan contraseñas. Este nivel de protección se denomina "Sin protección".

Aquí sólo se puede introducir una contraseña si se ha activado la opción "Eliminable con contraseña". Mediante esta opción también se puede acceder en escritura a la CPU en RUN si en el programa de usuario se ha usado la instrucción "PROTECT" para proteger la CPU contra escritura.

Nota

Si en la CPU está activado el nivel de protección predefinido y la CPU dispone de la instrucción "PROTECT (Página 5957)", esta instrucción permite cambiar entre el nivel de protección predefinido "Sin protección" y el nivel de protección "Protección contra escritura".

Otros niveles de protección protegidos por contraseña

Para los siguientes niveles de protección es necesario asignar una contraseña:

- Protección contra escritura: Con este nivel de protección, si no se introduce la contraseña correcta, únicamente será posible el acceso de lectura, independientemente de la posición del interruptor de llave. Para seleccionar este nivel de protección, es necesario asignar una contraseña.
- Protección de escritura/lectura: Con este nivel de protección, si no se introduce la contraseña correcta, no se tendrá acceso ni de lectura ni de escritura, independientemente de la posición del interruptor de llave. Para seleccionar este nivel de protección, es necesario asignar una contraseña.

A los usuarios que conocen la contraseña se les permite el acceso en lectura y escritura, independientemente del nivel de protección parametrizado.

Comportamiento de un módulo protegido por contraseña durante el funcionamiento

La protección de la CPU será efectiva una vez que los ajustes se hayan cargado en la CPU.

Antes de ejecutar una función online se comprueban los permisos y, si existe protección por contraseña, se pide que se introduzca la contraseña.

Ejemplo: el módulo se ha parametrizado con una protección contra escritura y debe ejecutarse la función "Forzar variable". Puesto que para ello se requiere un acceso de escritura, para ejecutar la función hay que introducir la contraseña parametrizada.

Las funciones protegidas por contraseña sólo pueden ejecutarse desde una programadora o un PC en un mismo momento. No se permite el acceso con contraseña de otra programadora u otro PC.

El permiso de acceso a los datos protegidos tiene validez mientras dure la conexión online o hasta que se vuelva a anular el permiso manualmente con "Online > Borrar derechos de acceso".

Nota

No se pueden restringir las funciones para la conducción, visualización y comunicación del proceso. Así, p. ej., la función "Ajustar fecha y hora" no se puede bloquear con una contraseña.

Modo Test y modo Proceso

Introducción

En algunos tipos de CPU se puede ajustar el modo de operación para el test del programa.

En el modo Proceso, las funciones de test, como Estado del programa u Observar/forzar variable, están restringidas para no exceder la prolongación admisible del tiempo de ciclo. No se puede efectuar el test con puntos de parada ni la ejecución del programa paso a paso.

En el modo Test desde la PG/el PC se pueden utilizar todas las funciones de test sin restricciones, aunque prolonguen considerablemente el tiempo de ciclo.

Posibilidades de ajuste del comportamiento operativo

Al realizar un test con el estado del programa en la ventana de inspección bajo "Propiedades > Modo de operación", se pueden ajustar los siguientes modos de operación:

- Modo de prueba
  Todas las funciones de prueba se pueden utilizar sin limitaciones. Es posible que aumente el tiempo de ciclo de la CPU, ya que, por ejemplo, en cada bucle se determina el estado de las instrucciones programadas en el mismo.
- Modo de proceso
  La función de prueba "Estado del programa" se limita para garantizar una carga lo más mínima posible del tiempo de ciclo:
  - Por ejemplo, no se permite ninguna condición de llamada.
  - La visualización del estado de un bucle programado se interrumpe en el punto de retorno.
  - No se puede ejecutar la función de prueba "PARADA" ni la ejecución del programa paso a paso.

Nota

Si la CPU está en modo Test, hay que asegurarse de que la CPU o el proceso puedan soportar una prolongación importante del tiempo de ciclo.

Con la opción "Modo Proceso" se evita que en el editor de programación se pueda configurar el entorno de llamada.

Parámetros de las CPUs compactas

Función Contaje

Propiedades - Contaje

Descripción

Si se ha seleccionado la CPU con función de contaje y medición integrada en la vista de redes o de dispositivos, es posible parametrizar la función de contaje para cada uno de los canales mencionados en "Propiedades > General > Contaje" de la ventana de inspección.

Si se utiliza el canal 0 ó 1 como canal de contaje no se podrá utilizar la tecnología "Posicionamiento".

Nota

El número de canales de contaje disponibles depende de la CPU utilizada.

En el campo de selección "Modo de operación" se puede definir el modo de contaje de la CPU para cada canal. Existen varias opciones para ello. Como ajuste predeterminado está seleccionada la opción "Sin parametrizar".

Consulte también

Documentación sobre funciones tecnológicas de la CPU (http://support.automation.siemens.com/WW/view/de/12429336/0/es)

Selección de alarmas

Para la función "Contaje" es posible seleccionar diversas alarmas.

Requisitos

- La CPU está seleccionada.
- La selección de alarmas se encuentra en "Propiedades > General > Contaje > Selección de alarmas".

Selección de alarmas (parámetros básicos)

Seleccione en la lista desplegable "Selección de alarmas" las alarmas que deberá disparar la CPU. Dispone de las siguientes posibilidades de selección:

- Ninguna
- Diagnóstico
- Proceso
- Diagnóstico y proceso

El ajuste predeterminado es "Ninguna".

Los eventos que deben disparar una alarma de proceso se seleccionan dependiendo del modo de operación ajustado.

Contaje

Eventos que pueden disparar una alarma de proceso:

- Abrir la puerta HW (si la puerta SW está abierta)
- Cerrar la puerta HW (si la puerta SW está abierta)
- Al alcanzar el valor de comparación (cuando reacciona el comparador)
- En caso de pulso de contaje
- Rebase por exceso (al rebasar el límite superior de contaje)
- Rebase por defecto (al rebasar el límite inferior de contaje)

Medición de frecuencia

Eventos que pueden disparar una alarma de proceso:

- Abrir la puerta HW (si la puerta SW está abierta)
- Cerrar la puerta HW (si la puerta SW está abierta)
- Fin de medición
- Rebase del límite superior
- Rebase del límite inferior

Modulación por ancho de pulso

Eventos que pueden disparar una alarma de proceso:

- Abrir la puerta HW (si la puerta SW está abierta)

Parámetros específicos de canal

General

En "Propiedades > General > Contaje > Canal 0 (... Canal n)" se ajustan los parámetros específicos de canal. Para el modo de operación hay seis opciones disponibles: "Sin configurar", "Contaje sin fin", "Contaje único", "Contaje periódico", "Medición de frecuencia" y "Modulación del ancho de impulso".

El modo de operación predeterminado es "Sin configurar".

Contaje

Parámetros operativos 'Contaje'

Sentido normal de contaje

En los modos de operación "Contaje único" y "Contaje periódico" se puede seleccionar en la lista desplegable "Sentido normal de contaje" si se desea incrementar o decrementar el contador sin sentido normal de contaje o con sentido normal de contaje. El ajuste predeterminado es "Ninguno".

En el modo de operación "Contaje sin fin", la lista desplegable "Sentido normal de contaje" no está disponible.

Función de puerta

En la lista desplegable "Función de puerta" puede elegir si cancelar o interrumpir el contaje cuando se cierre la puerta.

Si selecciona "Cancelar contaje", entonces el contaje comenzará a partir del valor de carga cuando se cierre la puerta y se vuelva a abrir. Por otro lado, si selecciona "Interrumpir contaje", el contaje continuará en el último valor de contaje actual cuando se cierre la puerta y se vuelva a abrir.

Como ajuste predeterminado está seleccionado "Cancelar contaje".

Valor inicial/Valor final

Dependiendo del sentido normal de contaje seleccionado en la lista desplegable correspondiente, en el campo de entrada "Valor inicial/valor final" podrá indicar un valor para el límite de contaje correspondiente.

Entre el sentido normal de contaje y esta entrada existe la siguiente relación:

- En caso de que el sentido normal de contaje sea Adelante, aquí se introducirá un valor final.
- En caso de que el sentido normal de contaje sea hacia atrás, aquí se introduce un valor inicial.

Si no hay seleccionado ningún sentido normal de contaje, este campo de entrada no esta disponible.

El ajuste predeterminado (sólo visible si hay seleccionado un sentido normal de contaje) es "2147483647".

Valor de comparación

En el campo de entrada "Valor de comparación" se puede introducir un valor que esté asignado a la salida digital, al bit de estado del comparador STS_CMP y a una alarma de proceso. El valor de contaje se compara con el valor de comparación. Dependiendo del resultado de dicha comparación, se puede activar la salida y el bit de estado del comparador STS_CMP y se puede generar la alarma de proceso "Alcanzar el valor de comparación".

En el grupo Salida se ajusta cómo se activará la salida en función del estado del contador.

El ajuste predeterminado del valor de comparación es "0".

Histéresis

Introduzca en el campo de entrada "Histéresis" un rango de histéresis entre 0 y 255 (0 y 1 significan: histéresis desactivada).

Un encoder puede detenerse en una posición determinada y, a continuación, "oscilar" alrededor de dicha posición. Como consecuencia de este estado, el estado del contador fluctuará en torno a un valor determinado. Si en esta banda de fluctuación se encontrase, por ejemplo, un valor de comparación, la salida correspondiente se activaría y desactivaría al ritmo de dichas fluctuaciones. Para evitar esta conmutación en caso de haya pequeñas fluctuaciones, la CPU está equipada con una histéresis parametrizable.

La histéresis se puede modificar mediante el programa de usuario con la instalación en marcha.

La histéresis afecta a la salida y a los bits de estado del comparador STS_CMP, del rebase por exceso STS_OFLW, del rebase por defecto STS_UFLW y del paso por cero STS_ZP.

Como ajuste predeterminado se indica "0".

Base de tiempo

Es posible elegir si la duración del período debe medirse en unidades de 125 ns o 1 µs en caso de una frecuencia de contaje máxima de 1 kHz. Si la frecuencia de contaje máxima es superior a 1 kHz no se mide la duración del período. Es posible medir la duración del período de la señal de contaje hasta una frecuencia de contaje máxima de 1 kHz.

Con una frecuencia de contaje máxima de 1 kHz se mide siempre el tiempo entre dos flancos de contaje sucesivos.

Con una base de tiempo de 1 µs pueden medirse duraciones de período de 4294 s = 71 min. 34 s como máximo y, con una base de tiempo de 125 ns, de 178 s = 2 min. 58 s como máximo.

Si los flancos de contaje están más separados en el tiempo, la duración medida del período será errónea, porque no se tiene en cuenta un rebase.

Entrada 0 (contaje)

Evaluación de señal

En la lista desplegable "Evaluación de señal" se selecciona qué señales transmite el encoder conectado. Puede seleccionar las siguientes señales:

- Impulso y sentido
- Encoder 1x
- Encoder 2x
- Encoder 4x

El valor predeterminado es "Impulso y sentido".

Puerta HW

La casilla de verificación "Puerta HW" permite seleccionar si desea utilizar una puerta de hardware o no.

Esta casilla de verificación no está activada de forma estándar.

Sentido de contaje invertido

La casilla de verificación "Sentido de contaje invertido" permite seleccionar si desea invertir el sentido de contaje o no.

Esta casilla de verificación no está activada de forma estándar.

Salida (contaje)

Comportamiento de la salida

Elija una de las siguientes opciones de la lista desplegable "Comportamiento de la salida":

- Sin comparación
  La salida funciona como una salida normal.
  Los parámetros de entrada SFB CTRL_DO y SET_DO no actúan.
  Los bits de estado STS_DO y STS_CMP (estado del comparador en el IDB) permanecen desactivados.
- Activar si estado del contador ≥ valor de comparación
  La salida se activa en cuanto el estado del contador alcanza el valor de comparación y se mantiene activada mientras el estado del contador sea mayor o igual al valor de comparación.

- Activar si estado del contador ≤ valor de comparación
  La salida se activa en cuanto el estado del contador alcanza el valor de comparación y se mantiene activada mientras el estado del contador sea menor o igual al valor de comparación.
- Impulso al alcanzar el valor de comparación
  Es posible predefinir una duración de impulso para adaptarse a los actuadores utilizados. La salida se activa por la duración de impulso ajustada en cuanto el estado del contador alcanza el valor de comparación. Si se ha ajustado un sentido normal de contaje, la salida se activa sólo cuando se alcanza el valor de comparación del sentido normal de contaje.

El ajuste predeterminado es "Sin comparación".

Duración de impulso

Si en la lista desplegable "Comportamiento de la salida" se ha seleccionado la alternativa "Impulso al alcanzar el valor de comparación", hay que introducir la duración del impulso de la señal de salida en ms en el campo de entrada "Duración de impulso". La duración máxima es de 510 ms. Sólo es posible introducir valores pares.

La duración de impulso comienza al activarse la salida digital correspondiente. La imprecisión de la duración de impulso es inferior a 2 ms.

Si se introduce 0 ms como duración de impulso, la salida queda activada hasta que la condición de comparación deje de cumplirse.

El ajuste predeterminado es "0 ms".

Nota

Si la salida se activa cuando se alcanza el valor de comparación, también se activa el bit STS_CMP, "Estado del comparador".

Frecuencia máx.

Señales de contaje/puerta HW

En esta lista desplegable se puede definir la frecuencia máxima de las señales vía A/impulso, vía B/dirección y puerta de hardware de acuerdo con una escala fija. El valor máximo es específico para cada CPU:

CPU	Rango de valores	Ajuste predeterminado
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	30 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	60 kHz

Si la CPU no soporta esta función, la lista desplegable está atenuada.

Congelar

En esta lista desplegable es posible ajustar la frecuencia máxima de la señal de congelación (latch) en una escala fija. El valor máximo es específico para cada CPU:

CPU	Rango de valores	Ajuste predeterminado
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	10 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	10 kHz

Si la CPU no soporta esta función, la lista desplegable está atenuada.

Alarma de proceso en (contaje)

Descripción

Si hace clic en las casillas de verificación correspondientes, el evento en cuestión dispara una alarma de proceso. Los siguientes eventos pueden disparar una alarma de proceso:

- Abrir la puerta HW (si la puerta SW está abierta)
- Cerrar la puerta HW (si la puerta SW está abierta)
- Al alcanzar el valor de comparación (cuando reacciona el comparador)
- En caso de pulso de contaje
- Rebase por exceso (al rebasar el límite superior de contaje)
- Rebase por defecto (al rebasar el límite inferior de contaje)

Para que la CPU también genere una alarma de proceso, deberá habilitar esta opción en la ficha "Parámetros básicos".

Como ajuste predeterminado no hay seleccionada ninguna casilla de verificación.

El disparo de una alarma de proceso en cada flanco de contaje provoca una fuerte carga de trabajo de la CPU en caso de frecuencias de contaje elevadas. Si las alarmas de proceso aparecen en el submódulo "Contaje" con más rapidez de la que pueden procesarse en el OB de alarma de proceso (OB 40), se produce un diagnóstico de "Alarma de proceso perdida" si la alarma de diagnóstico está habilitada.

Asignación de los datos de entrada

Asignación de los datos de entrada

Es posible elegir si con una frecuencia de contaje máxima de 1 kHz puede leerse el valor de contaje o bien la duración del período en los datos de entrada (datos E) del submódulo "Contaje". Si la frecuencia de contaje máxima es superior a 1 kHz sólo se admite "Valor de contaje".

Es posible medir el período de la señal de contaje hasta una frecuencia de contaje máxima de 1 kHz.

Con una frecuencia de contaje máxima de 1 kHz se mide siempre el tiempo entre dos flancos de contaje sucesivos. Si la frecuencia de contaje máxima es superior a 1 kHz no se mide la duración del período, el valor es 0.

Medición de frecuencia

Medida de frecuencia

Descripción

En este modo de operación, la CPU cuenta los impulsos que aparecen en un intervalo de tiempo predeterminado (tiempo de integración) y calcula la frecuencia a partir del número de impulsos y de la duración del tiempo de integración.

El valor de la frecuencia determinada se indica en la unidad "mHz".

Parámetros operativos 'Medida de frecuencia'

Descripción

En este modo de operación, la CPU cuenta los impulsos que aparecen durante el tiempo de integración parametrizado.

La medición se realiza durante el tiempo de integración. Una vez transcurrido el tiempo de integración, se actualiza el valor medido.

Si durante el tiempo de integración no ha habido como mínimo un flanco ascendente, se dará como respuesta el valor medido 0 o el valor medio, según sea la parametrización.

Tiempo de integración

En el campo de entrada "Tiempo de integración" se puede introducir un tiempo de integración de entre 10 ms y 10000 ms.

El ajuste predeterminado es "100 ms".

Valor límite inferior y valor límite superior

En los campos de entrada "Valor límite inferior" y "Valor límite superior" se indican los valores con los que se debe comparar la frecuencia determinada una vez transcurrido el tiempo de

integración. Para esta supervisión de límites se admiten los siguientes rangos de valores en función de la CPU utilizada:

- Límite inferior
  - CPU 312C: de 0 a 9 999 999 mHz
  - CPU 313C: de 0 a 29 999 999 mHz
  - CPU 314C: de 0 a 59 999 999 mHz
- Límite superior
  - CPU 312C: de 1 mHz a 10 000 000 mHz
  - CPU 313C: de 1 mHz a 30 000 000 mHz
  - CPU 314C: de 1 mHz a 60 000 000 mHz

Nota

Al introducir estos valores se comprueba si el límite superior es mayor que el límite inferior.

Como ajuste predeterminado, los límites inferiores están ajustados a 0 mHz. Como ajuste predeterminado, los límites superiores tienen asignados los siguientes valores en función de la CPU utilizada:

- CPU 312C: 10 000 000 mHz
- CPU 313C: 30 000 000 mHz
- CPU 314C: 60 000 000 mHz

Máx. frecuencia de contaje/puerta HW

En esta lista desplegable se puede definir la frecuencia máxima de las señales vía A/impulso, vía B/dirección y puerta de hardware de acuerdo con una escala fija. El valor máximo es específico para cada CPU:

CPU	Rango de valores	Ajuste predeterminado
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	30 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	60 kHz

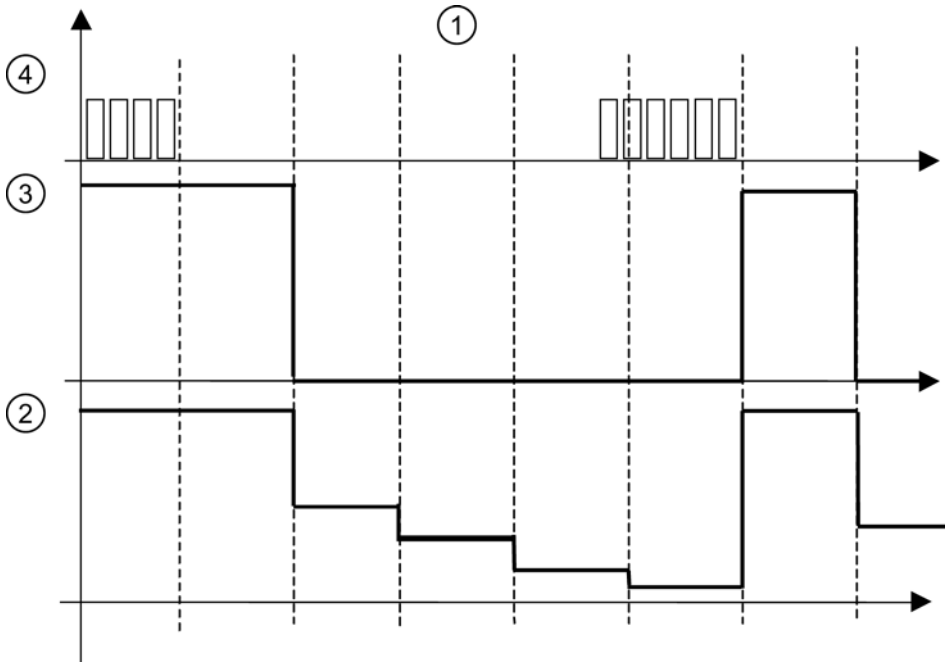
Indicar valor medido

Puede elegir si desea que el valor medido se indique directamente o como valor medio. Para ello, haga clic en la opción correspondiente "Directamente" o "Promediado" junto a "Salida valor medido".

Al final del tiempo de integración, en el que se produce al menos un flanco ascendente, se muestra la frecuencia medida.

Si durante el tiempo de integración parametrizado no ha habido como mínimo un flanco positivo:

- En caso de frecuencia con valor directo, se indicará el valor "0" al final del tiempo de integración.
- En caso de frecuencia con valor medio, se repartirá el último valor entre los siguientes intervalos de medición sin flanco ascendente (f ≥ 1 mHz). Esto implica una prolongación del tiempo de integración. Para ello se divide el último valor medido por el número de intervalos de medición sin flanco ascendente.
  Ejemplo: si el último valor medido es 12000 mHz y hay tres intervalos de medición, se obtiene el valor 4000 mHz.



① Intervalo t.
② Frecuencia media
③ Frecuencia directa
④ Impulsos

El ajuste predeterminado es la opción "Directamente" de la lista desplegable.

Entrada 0 (medición de frecuencia)

Evaluación de señal

En la lista desplegable "Sentido de contaje invertido" se selecciona qué señales transmite el encoder conectado. Se pueden seleccionar las siguientes señales:

- Impulso y sentido
- Encoder rotativo 1x

El ajuste predeterminado es "Impulso y sentido".

Puerta HW

La casilla de verificación "Puerta HW" permite seleccionar si desea utilizar una puerta de hardware o no.

Esta casilla de verificación no está activada de forma estándar.

Sentido de contaje invertido

La casilla de verificación "Sentido de contaje invertido" permite seleccionar si desea invertir el sentido de contaje o no.

Esta casilla de verificación no está activada de forma estándar.

Salida (medición de frecuencia)

Comportamiento de la salida

Elija una de las siguientes opciones de la lista desplegable "Comportamiento de la salida":

- Sin comparación
  La salida funciona como una salida normal.
  Los parámetros de entrada SFB MAN_DO y SET_DO no actúan.
  El bit de estado STS_DO queda desactivado.
- Activar cuando queda fuera de los límites
  La salida se pone a 1 en cuanto la frecuencia medida es mayor que el límite superior o menor que el límite inferior.
- Activar cuando queda por debajo del límite inferior
  La salida se pone a 1 en cuanto la frecuencia medida es menor que el límite inferior.
- Activar cuando queda por encima del límite superior
  La salida se activa en cuanto la frecuencia medida es mayor que el límite superior.

El ajuste predeterminado es la opción "Sin comparación".

Alarma de proceso en (medición de frecuencia)

Descripción

Si hace clic en las casillas de verificación correspondientes, el evento en cuestión dispara una alarma de proceso. Los siguientes eventos pueden disparar una alarma de proceso:

- Abrir la puerta HW (si la puerta SW está abierta)
- Cerrar la puerta HW (si la puerta SW está abierta)
- Fin de medición
- Valor límite superior rebasado por exceso
- Valor límite inferior rebasado por defecto

Para que la CPU también genere una alarma de proceso, debe habilitar esta opción en la ficha Parámetros básicos.

Como ajuste predeterminado no hay seleccionada ninguna casilla de verificación.

Modulación del ancho de impulso

Modulación del ancho de pulso

Descripción

La CPU transforma el valor de salida indicado por el usuario en una secuencia de impulsos con la relación impulso/pausa correspondiente (modulación de ancho de pulso).

La secuencia de impulsos se emite una vez transcurrido el retardo a la conexión parametrizado en la salida digital DO (secuencia de salida).

Parámetros operativos `Modulación del ancho de pulso´

Formato de salida

En la lista desplegable "Formato de salida" se selecciona el rango del valor de salida haciendo clic en la opción "Por mil" o "Valor analógico S7".

Si el valor de salida está entre 0 y 1000, haga clic en la opción "Por mil".

Si el valor de salida es un valor analógico SIMATIC S7 (entre 0 y 27648), haga clic en la opción "Valor analógico S7".

El ajuste predeterminado es el botón de opción "Por mil".

Base de tiempo

En la lista desplegable "Base de tiempo", seleccione la base de tiempo para la resolución y el rango de valores de la duración de período, la duración mínima de impulso y el retardo a la conexión.

Si se hace clic en la opción "1 ms", los tiempos se ajustan con una resolución de 1 ms.

Si se hace clic en la opción "0,1 ms", los tiempos se ajustan con una resolución de 0,1 ms.

El ajuste predeterminado es la opción "0,1 ms"

Retardo a la conexión

En el campo de entrada "Retardo a la conexión", introduzca un valor para el tiempo que transcurre desde el comienzo de la secuencia de salida hasta la salida del impulso.

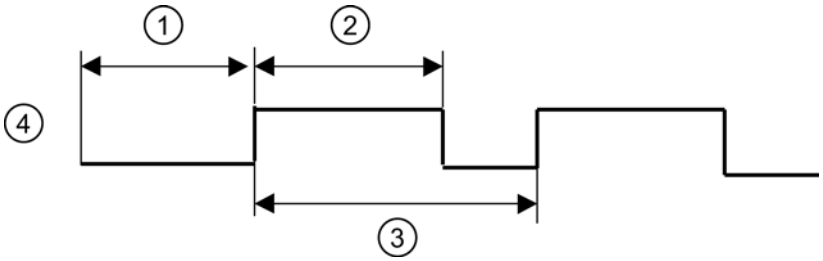
Introduzca el valor en la resolución ajustada anteriormente.

Si en la lista desplegable "Base de tiempo" ha hecho clic en la opción "1 ms", los tiempos se ajustan de 0 ms a 65535 ms con una resolución de 1 ms.

Si en la lista desplegable "Base de tiempo" ha hecho clic en la opción "0,1 ms", los tiempos se ajustan de 0 ms a 6553,5 ms con una resolución de 0,1 ms.

El ajuste predeterminado es "0".

La secuencia de impulsos se emite una vez transcurrido el retardo a la conexión en la salida digital DO (secuencia de salida).



① Retardo a la conexión
② Duración de impulso
③ Duración del período
④ Salida DO

El ajuste predeterminado es "0".

Duración del período

La duración del período define la longitud de la secuencia de salida, que consiste en duración de impulso y pausa entre impulsos.

Introduzca en el campo de entrada "Duración del período" un valor en la resolución ajustada anteriormente.

Si en la lista desplegable "Base de tiempo" ha hecho clic en la opción "1 ms", los tiempos se ajustan de 1 ms a 65535 ms con una resolución de 1 ms.

Si en la lista desplegable "Base de tiempo" ha hecho clic en la opción "0,1 ms", los tiempos se ajustan de 0,4 ms a 6553,5 ms con una resolución de 0,1 ms.

El ajuste predeterminado es "20000".

Duración mínima de impulso

Con la duración mínima de impulso se suprimen impulsos de salida cortos y pausas de impulso cortas. Se suprimen todos los impulsos y pausas que son más breves que la duración mínima de impulso.

Introduzca en el campo de entrada "Duración mínima de impulso" un valor en la resolución ajustada anteriormente.

Si en la lista desplegable "Base de tiempo" ha hecho clic en la opción "1 ms", los valores se introducen de 0 hasta una duración de período/2.

Si en la lista desplegable "Base de tiempo" ha hecho clic en la opción "0,1 ms", los valores se introducen de 2 hasta una duración de período/2.

Si se ha seleccionado la base de tiempo 1 ms y se introduce 0 como duración mínima de impulso, ésta se ajusta internamente a 0,2 ms.

El ajuste predeterminado es "2".

Entrada 0 (modulación del ancho de impulso)

Puerta HW

La casilla de verificación "Puerta HW" permite seleccionar si desea utilizar una puerta de hardware o no.

Esta casilla de verificación no está activada de forma estándar.

Sólo podrá iniciar la modulación del ancho de impulso con la puerta de hardware si primero abre la puerta de software y, a continuación, genera un flanco ascendente en la entrada digital "Puerta de hardware".

La modulación por ancho de impulso sólo se puede parar con un flanco descendente en la puerta de software. El estado de la puerta de hardware es irrelevante.

Frecuencia de filtro

Puerta HW

Esta lista desplegable sólo está activa si se ha seleccionado la opción "Puerta HW" en "Entrada". Aquí se ajusta la frecuencia máxima de la señal de la puerta de hardware en una escala fija. El valor máximo es específico para cada CPU:

CPU	Rango de valores	Ajuste predeterminado
CPU 312C	10, 5, 2, 1 kHz	10 kHz
CPU 313C CPU 313C-2 DP CPU 313C-2 PtP	30, 10, 5, 2, 1 kHz	30 kHz
CPU 314C-2 DP CPU 314C-2 PN/DP CPU 314C-2 PtP	60, 30, 10, 5, 2, 1 kHz	60 kHz

Alarma de proceso en (modulación del ancho de impulso)

Descripción

Si se hace clic en la casilla de verificación correspondiente, el evento en cuestión dispara una alarma de proceso. El siguiente evento puede disparar una alarma de proceso:

- Abrir la puerta HW (si la puerta SW está abierta)

Para que la CPU también genere una alarma de proceso, debe habilitar esta opción en la ficha Parámetros básicos.

Como ajuste predeterminado no hay seleccionada ninguna casilla de verificación.

Contaje sin fin

Descripción

El modo de operación "Contaje sin fin" se define del siguiente modo:

Cuando el contador alcanza el límite superior de contaje y aparece otro impulso de contaje, el contador salta al límite inferior y sigue contando desde ahí.

Cuando el contador alcanza el límite inferior y aparece otro impulso de contaje, el contador salta al límite superior y sigue contando desde ahí.

Los límites de contaje están ajustados de forma fija al rango máximo.

En este modo de operación, el contaje se inicia y se detiene mediante la función de puerta.

Contaje único

Descripción

Cuando el contador alcanza el valor final - 1 estando seleccionado como sentido normal de contaje el ajuste Adelante y aparece otro impulso, el contador salta al valor inicial (valor de carga) y se para aunque sigan apareciendo otros impulsos.

Cuando el contador alcanza el valor de contaje = 1 estando seleccionado como sentido normal de contaje el ajuste Atrás y aparece otro impulso de contaje, el contador salta al valor inicial (valor de carga) y permanece parado aunque aparezcan otros impulsos de contaje.

Si no ha ajustado el sentido normal de contaje, los límites de contaje corresponderán al rango máximo.

En este modo de operación, el contaje se inicia y se detiene mediante la función de puerta.

Cuando la puerta se abre, los impulsos acceden al contador y éste empieza a contar. Cuando se cierra, los impulsos ya no pueden acceder al contador y éste se para.

Contaje periódico

Descripción

Si el contador ha alcanzado el valor final - 1 al contar hacia delante en sentido normal de contaje y llega otro pulso de contaje, el contador salta al valor de carga y sigue contando desde allí.

Si el contador ha alcanzado el valor final = 1 al contar hacia atrás en sentido normal de contaje y llega otro pulso de contaje, el contador salta al valor inicial (valor de carga) y sigue contando desde allí.

Si no ha ajustado el sentido normal de contaje, los límites de contaje corresponderán al rango máximo.

En este modo de operación, el contaje se inicia y se detiene mediante la función de puerta.

Cuando la puerta se abre, los impulsos acceden al contador y éste empieza a contar. Cuando se cierra, los impulsos ya no pueden acceder al contador y éste se para.

Sentido normal de contaje

Descripción

Es posible parametrizar "adelante" o "atrás" como sentido normal de contaje. Así se define para los modos de operación Contaje único y Contaje periódico qué límite de contaje rige como valor inicial o final.

En el sentido normal de contaje hacia delante, el contador cuenta desde 0 o desde el valor de carga hasta el valor final parametrizado.

En el sentido normal de contaje hacia atrás, el contador comienza por el valor inicial parametrizado y cuenta hasta 0.

Aunque se haya parametrizado "atrás" como sentido normal de contaje, para contar hacia atrás es necesario crear una señal de sentido correspondiente o ajustar "Sentido de contaje invertido" en la parametrización.

Función de puerta

Puerta de software y puerta de hardware

La CPU tiene dos puerta combinadas lógicamente con Y:

Una **puerta de software** (puerta SW) que se controla mediante el programa de usuario.

La puerta de software puede abrirse únicamente con un cambio de flanco 0-1 del parámetro de entrada SW_GATE en el SFB. Se cierra desactivando dicho parámetro.

Una **puerta de hardware** (puerta HW) que se controla mediante la entrada digital "Puerta de hardware". En la ficha "Contaje" se ajusta si se quiere utilizar la puerta de hardware. Se abre con un cambio de flanco 0-1 en la entrada digital y se cierra con un cambio de flanco 1-0.

Función Posicionamiento

Selección de alarmas

Para la función "Posicionamiento" existe la posibilidad de seleccionar alarmas.

Requisitos

- La CPU está seleccionada.
- La selección de alarmas se encuentra en "Propiedades > General > Posicionamiento > Selección de alarmas".

Selección de alarmas

Seleccione en la lista desplegable "Selección de alarmas" si la CPU debe o no disparar la alarma de diagnóstico. Dispone de las siguientes posibilidades de selección:

- Ninguna
- Diagnóstico

El ajuste predeterminado es "Ninguna".

Para que los eventos correspondientes también disparen alarmas de diagnóstico, es necesario realizar ajustes adicionales.

Active la vigilancia deseada en "Propiedades > General > Posicionamiento > Canal 0" estando seleccionado el modo de operación "Posicionar con salida analógica":

- En "Accionamiento"
  Se puede vigilar el valor real, la aproximación al destino y la zona de destino.
- En "Eje"
  Para un eje lineal, aquí se puede establecer si se va a vigilar la zona de trabajo. El margen de desplazamiento se vigila siempre.
- En "Encoder"
  En caso de los encoders con señal cero (impulso de origen), se comprueba si el encoder transmite siempre el mismo número de impulsos entre dos señales cero consecutivas.

En "Canal 0 - Diagnóstico" se seleccionan los eventos que deben disparar una alarma de diagnóstico.

Para disparar una alarma de diagnóstico es necesario parametrizar los siguientes ajustes:

- Selección de alarmas en "Propiedades > General > Posicionamiento > Canal 0"
- La vigilancia correspondiente en el apartado "Accionamiento", "Eje" o "Encoder"
- La habilitación correspondiente en el apartado "Diagnóstico"

Modo de operación

Modo de operación

Con el parámetro "Modo de operación" se determina el modo en el que se va a realizar el posicionamiento. Dispone de las siguientes opciones:

- Salidas analógicas: posicionamiento con salida analógica
- Salidas digitales: posicionamiento con salidas digitales

Como ajuste predeterminado está seleccionada la opción "Sin parametrizar".

La selección del modo de operación se encuentra en "Propiedades > General > Posicionamiento > Canal 0".

Nota

La función tecnológica "Posicionamiento" no se puede parametrizar si en la función Contaje se ha parametrizado Canal 0 o Canal 1.

Posicionar con salida analógica sólo se puede parametrizar si en el apartado AI5/AO2 está desactivada la salida 0.

Los canales de entrada y salida ocupados por el posicionamiento ya no son accesibles desde el programa de usuario.

Consulte también

Documentación sobre funciones tecnológicas de la CPU (http://support.automation.siemens.com/WW/view/de/12429336/0/es)

Posicionamiento con salida analógica

Accionamiento

Activación del accionamiento y el tipo de control

Activación del accionamiento y el tipo de control

En el apartado "Canal 0 - accionamiento" de "Propiedades > Posicionamiento > Canal 0" se determinan los parámetros para el accionamiento del eje en el posicionamiento con salida analógica.

Descripción del accionamiento

El control del accionamiento se efectúa a través de dos salidas analógicas asignadas de forma fija, ya sea con una tensión de ±10 V (pin 16) o con una intensidad de ±20 mA (pin 17).

Nota

El posicionamiento con salida analógica sólo se puede parametrizar si en el apartado AI5/AO2 de la navegación local está desactivada la salida 0.

Descripción del tipo de control

La lista desplegable permite elegir entre 2 tipos de control. El tipo de control describe cómo se controla la etapa de potencia conectada.

- Tensión ±10 V o intensidad ±20 mA:

En los desplazamientos en sentido positivo (hacia delante) se aplica una tensión o intensidad positiva. En los desplazamientos en sentido negativo (hacia atrás) se aplica una tensión o intensidad negativa.

- Tensión 0 a 10 V e intensidad 0 a 20 mA y señal de sentido:

En los desplazamientos en sentido positivo (hacia delante) se aplica una tensión o intensidad positiva y la salida digital CONV_DIR está desactivada.

En los desplazamientos en sentido negativo (hacia atrás) se aplica una tensión o intensidad negativa y la salida digital CONV_DIR está activada.

El ajuste predeterminado es "±10".

Frecuencia max.

Lectura de recorrido

En esta lista desplegable se puede ajustar la frecuencia de contaje máxima de las señales de lectura de recorrido (señales del encoder A, B, N) en una escala fija.

Si la CPU no soporta esta función, la lista desplegable está atenuada.

El ajuste predeterminado es "60 kHz".

Señales cualificadoras

En esta lista desplegable es posible ajustar la frecuencia máxima de las señales de medición de longitud y sensor del punto de referencia en una escala fija.

Si la CPU no soporta esta función, la lista desplegable está atenuada.

El ajuste predeterminado es "10 kHz".

Aproximación a destino (analógica)

Zona de destino

En el campo de entrada "Zona de destino" se introduce un valor par entre 0 y 2 x 10 impulsos. Los valores impares se redondean internamente. La zona de destino está situada de forma simétrica en torno al destino. Si se introduce aquí el valor 0, el destino debe alcanzarse exactamente con el impulso. En este caso se desactiva la vigilancia de la zona de destino y la casilla de verificación correspondiente del área "Vigilancia" aparece atenuada.

El ajuste predeterminado es de "50 impulsos".

Tiempo de vigilancia

En el campo de entrada "Tiempo de vigilancia" se introduce un tiempo entre 0 y 100000 ms. La CPU redondea dicho valor a un ciclo de ejecución de 4 ms.

Con el tiempo de vigilancia, la CPU vigila el valor real y la aproximación a destino.

Si se introduce el valor 0 aquí, se desactivan las vigilancia del valor real y de la aproximación al destino y las casillas de verificación correspondientes en el área "Vigilancia" están atenuadas.

El ajuste predeterminado es de "2000 ms".

Velocidad máxima

En el campo de entrada "Velocidad máxima" se introduce un valor entre 10 y 1 000 000 impulsos/s, que equivale al nivel de 10 V o 20 mA en la salida analógica.

El ajuste predeterminado es de "1000 impulsos/s".

Velocidad lenta/de búsqueda de referencia

En el campo de entrada "Velocidad lenta/de búsqueda de referencia" se introduce un valor entre 10 impulsos/s y la velocidad máxima parametrizada.

Una vez alcanzado el punto de frenada se decelera a velocidad lenta.

En la búsqueda del punto de referencia, se decelera a la velocidad de búsqueda de referencia una vez alcanzado el sensor del punto de referencia.

El ajuste predeterminado es de "100 impulsos/s".

Habilitar etapa de potencia

Retardo a la desconexión

En el campo de entrada "Retardo a la desconexión" introduzca un tiempo en milisegundos. La CPU redondea dicho valor a un ciclo de ejecución de 4 ms. En caso de interrumpirse un desplazamiento (retirada de la habilitación para la etapa de potencia), la salida CONV_EN se desactiva con un retardo correspondiente a este tiempo.

Si controla un freno a través de la salida digital, mediante el retardo puede garantizar que el eje sea tan lento en el momento de activar la salida que el freno pueda disipar la energía del movimiento.

El ajuste predeterminado es "1000 ms".

Vigilancia (analógica)

Descripción

Aquí se activa y se desactiva la vigilancia del valor real, de la aproximación al destino y de la zona de destino.

Zona de destino

Haciendo clic en la casilla de verificación "Zona de destino" se vigila si, tras alcanzar la zona de destino, el accionamiento se detiene en la posición de destino del movimiento iniciado o si se aleja de ella.

Si en "Zona de destino" se ha ajustado un tiempo de vigilancia de 0 ms, no es posible activar la vigilancia de la zona de destino y los campos correspondientes están atenuados.

En caso de que la vigilancia reaccione se genera un error externo. Dicho error externo debe ser acusado mediante ERR_A. La vigilancia se desactiva. La vigilancia no se vuelve a activar hasta que no se inicia otro desplazamiento.

Esta vigilancia está desactivada de forma estándar.

Aproximación a destino

Si se ha hecho clic en la casilla de verificación "Aproximación a destino", se vigila si el eje alcanza la zona de destino dentro del tiempo de vigilancia después de alcanzar el punto de desconexión.

Si en "Aproximación a destino" se ha ajustado un tiempo de vigilancia de 0 ms, no es posible activar la vigilancia de la aproximación a destino y los campos correspondientes están atenuados.

Esta vigilancia está desactivada de forma estándar.

Valor real

Si se hace clic en la casilla de verificación "Valor real", se vigila el movimiento del eje desde el inicio del desplazamiento hasta alcanzar el punto de desconexión. Si el eje no se desplaza al menos un impulso en el sentido predeterminado durante el tiempo de vigilancia, se interrumpe el movimiento.

Si en "Aproximación a destino" se ha ajustado un tiempo de vigilancia de 0 ms, no es posible activar la vigilancia del valor real y los campos correspondientes están atenuados.

La CPU no detecta el fallo de una entrada digital. Mediante la vigilancia del valor real se puede detectar el fallo de una señal del encoder. Este fallo puede tener las siguientes causas:

- Fallo de la entrada digital
- Rotura de hilo
- Defecto del encoder
- Error en la etapa de potencia

Como ajuste predeterminado está activada esta vigilancia.

Eje

Tipo de eje (analógico)

Descripción

En "Propiedades > Posicionamiento > Canal 0" del apartado "Canal 0 - eje" se determinan los parámetros para el eje.

Tipo de eje

En "Tipo de eje" seleccione si desea controlar un eje lineal o un eje rotativo.

El ajuste predeterminado es "Eje lineal".

Inicio y fin del final de carrera por software

El área de trabajo de un eje lineal está entre el inicio y el final del final de carrera por software. Los finales de carrera por software forman parte del área de trabajo. Introduzca en los campos de entrada correspondientes valores entre - 5 x 10^8 y 5 x 10^8 impulsos. El inicio del final de carrera por software siempre debe ser menor que el final del final de carrera por software.

Los finales de carrera por software se vigilan cuando el eje está sincronizado y la vigilancia del área de trabajo está activada.

El ajuste predeterminado es de "-100000000 impulsos" para el inicio y "100000000 impulsos" para el final del final de carrera por software.

Fin de eje rotativo

En el campo de entrada "Fin eje rotativo" introduzca un valor entre 1 y 10^9 impulsos.

El valor "Fin de eje rotativo" es el valor teóricamente más grande que puede alcanzar el valor real. Tiene la misma posición física que el inicio del eje rotativo (0).

El ajuste predeterminado es de "100000 impulsos".

Medición de longitud (analógica)

Descripción

La función "Medición de longitud" permite medir longitudes durante un desplazamiento. El inicio y el fin de la medición de longitud tienen lugar mediante flancos en la entrada digital "Medición de longitud". Seleccione el tipo de flanco en la lista desplegable "Detección".

Dispone de las siguientes opciones:

- OFF
- Principio/fin con flanco ascendente
  Un flanco ascendente en la entrada digital inicia la medición de longitud, el siguiente flanco ascendente la finaliza.

- Principio/fin con flanco descendente
  Un flanco descendente en la entrada digital inicia la medición de longitud, el siguiente flanco descendiente lo finaliza.
- Inicio con flanco ascendente, fin con flanco descendente
  Un flanco ascendente en la entrada digital inicia la medición de longitud, el siguiente flanco descendente la finaliza.
- Inicio con flanco descendente, fin con flanco ascendente
  Un flanco descendente en la entrada digital inicia la medición de longitud, el siguiente flanco ascendente la finaliza.

El ajuste predeterminado es "OFF".

Punto de referencia (analógico)

Coordenadas

Tras una transición STOP-RUN de la CPU y si el eje todavía no está sincronizado, el valor real se pone al valor de la coordenada del punto de referencia.

Tras una búsqueda del punto de referencia, se asigna al punto de referencia el valor de la coordenada del punto de referencia.

Tratándose de un eje lineal, el valor de la coordenada del punto de referencia debe estar comprendido dentro del área de trabajo (incluidos el inicio y el final del final de carrera de software).

Tratándose de un eje rotativo, el valor de la coordenada del punto de referencia debe encontrarse en el rango de 0 a "Fin del eje rotativo - 1".

El ajuste predeterminado es de "0 impulsos".

Decalaje del punto de referencia con respecto al sensor

En la lista desplegable "Decalaje del punto de referencia con respecto al sensor", elija si el punto de referencia se encuentra en sentido más (+) (los valores reales se incrementan) o en sentido menos (-) (los valores reales se decrementan) con respecto al sensor del punto de referencia.

El ajuste predeterminado es "en sentido +".

Vigilancia del eje (analógica)

Área de trabajo

Para un eje lineal, aquí se puede establecer si se va a vigilar el área de trabajo. Si se hace clic en la casilla de verificación "Área de trabajo", se comprueba si el valor real de posición se encuentra fuera de los finales de carrera por software. La vigilancia tiene efecto solamente en caso de que el eje esté sincronizado.

Las coordenadas de los finales de carrera por software pertenecen al área de trabajo.

Si se dispara la vigilancia se interrumpe el desplazamiento.

Como ajuste predeterminado está activada esta vigilancia.

Rango de desplazamiento

Con la vigilancia del rango de desplazamiento se comprueba si se abandona el rango de desplazamiento admisible de - 5 x 10^8 a + 5 x 10^8. El rango de desplazamiento se vigila siempre. La vigilancia no se puede desactivar (siempre está activada en el parámetro "Vigilancia").

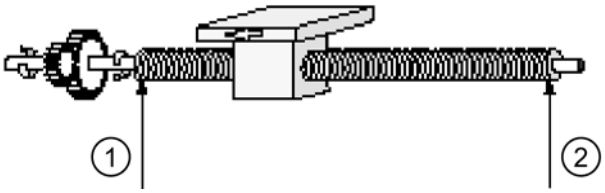
Si se dispara la vigilancia se interrumpe el desplazamiento.

Eje lineal

Descripción

El **eje lineal** es un eje que tiene un margen de desplazamiento físico limitado.

El área de trabajo de un eje lineal está limitada por los finales de carrera por software.

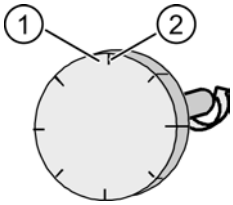


① Inicio físico
② Fin físico

Eje rotativo

Descripción

El **eje rotativo** es un eje cuyo rango de desplazamiento no está limitado por topes mecánicos.



① Valor más alto visualizable = Final del eje rotativo - 1
② Inicio del eje rotativo (coordenada 0) = Final del eje rotativo

Una vuelta del eje rotativo comienza por la coordenada "Cero" y termina por la coordenada "Final del eje rotativo" Físicamente, la coordenada "Cero" es idéntica a la coordenada "Final del eje rotativo" (= 0). En este punto salta la visualización del valor real de posición. La visualización del valor real de posición siempre es positiva.

Encoder

N.º de incrementos por vuelta

En el campo de entrada "N.º de incrementos por vuelta" se introduce el número de incrementos del encoder en una vuelta completa. Dicho valor se toma de la descripción del encoder.

La CPU evalúa los incrementos por cuadruplicado, es decir, un incremento equivale a cuatro impulsos.

El ajuste predeterminado es de "1000 incrementos".

Sentido de contaje

Con "Sentido de contaje" se adapta el sentido de la lectura del recorrido al sentido del movimiento del eje.

Haga clic en la opción "Normal" de la lista desplegable "Sentido de contaje", si los impulsos de contaje ascendentes corresponden a valores reales de posición ascendentes.

Haga clic en la opción "Invertido" de la lista desplegable "Sentido de contaje", si los impulsos de contaje ascendentes corresponden a valores reales de posición descendentes.

Tenga en cuenta también todos los sentidos de giro de los elementos de transmisión (como acoplamientos y transmisiones).

El ajuste predeterminado es la opción "Normal".

Vigilancia

En "Vigilancia" se determina si se deben vigilar los errores de señal del encoder. Esto sólo es posible en encoders cuyo valor en "N.º de incrementos por vuelta" sea divisible por 10 ó 16.

Haciendo clic en la casilla de verificación "Error en señal" se comprueba si el encoder devuelve siempre el mismo número de impulsos entre dos señales cero consecutivas.

Si se utiliza un encoder sin señal cero, habrá que desactivar la vigilancia de los errores en señal.

Esta vigilancia está desactivada de forma estándar.

Diagnóstico

Descripción

Seleccione en el apartado "Canal 0 - diagnóstico" de "Propiedades > Posicionamiento > Canal 0" en qué eventos es necesario generar una alarma de diagnóstico. Como ajuste predeterminado no hay seleccionada ninguna casilla de verificación.

Error en señal

Si se hace clic en la casilla de verificación "Error en señal", se genera una alarma de diagnóstico tan pronto como el encoder deja de transmitir el mismo número de impulsos entre

dos señales cero consecutivas. Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - encoder".

Rango de desplazamiento

Si se hace clic en la casilla de verificación "Rango de desplazamiento", se genera una alarma de diagnóstico en cuanto el eje abandona el rango de desplazamiento. La vigilancia correspondiente siempre está activada.

Zona de trabajo

Si se hace clic en la casilla de verificación "Área de trabajo", se genera una alarma de diagnóstico si el valor real de posición se encuentra fuera de los finales de carrera por software. Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - eje".

En caso de que el posicionamiento sea en un eje rotativo no se podrá utilizar esta alarma de diagnóstico. La casilla de verificación correspondiente aparecerá sombreada.

Valor real

Si ha marcado la casilla de verificación "Valor real", se generará una alarma de diagnóstico si durante un movimiento de desplazamiento el eje no se mueve como mínimo un impulso en el sentido indicado dentro del tiempo de vigilancia. En tal caso se interrumpirá el desplazamiento.

El valor real se vigila desde el inicio del desplazamiento hasta alcanzar el punto de desconexión.

Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - accionamiento".

La CPU no detecta el fallo de una entrada digital. Activando la vigilancia del valor real se puede detectar el fallo de una señal de sensor.

Esta casilla de verificación está seleccionada de forma estándar.

Aproximación a destino

Si se hace clic en la casilla de verificación "Aproximación a destino", se genera una alarma de diagnóstico si, tras alcanzar el punto de desconexión, el eje no alcanza la zona de destino durante el tiempo de vigilancia.

Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - accionamiento".

Zona de destino

Si se ha hecho clic en la casilla de verificación "Zona de destino", se genera una alarma de diagnóstico si el accionamiento no se detiene en una posición de destino del movimiento iniciado o si se aleja de ella después de alcanzar la zona de destino.

Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - accionamiento".

Posicionamiento con salidas digitales

Accionamiento

Activación del accionamiento

En el apartado "Canal 0 - accionamiento" de "Propiedades > Posicionamiento > Canal 0" se determinan los parámetros para el accionamiento del eje en el posicionamiento con salidas digitales.

Descripción

El accionamiento se activa a través de cuatro salidas digitales de 24 V con asignación fija.

La lista desplegable permite elegir entre 4 tipos de control. El tipo de control describe cómo las 4 salidas digitales (de DO + 1.0 a DO + 1.3) controlan la etapa de potencia.

A continuación se especifica la función de cada una de las salidas en los diversos tipos de control.

Salida	Tipo de control 1	Tipo de control 2	Tipo de control 3	Tipo de control 4
DO + 1.0	Marcha rápida	Marcha rápida/lenta	Marcha rápida	Marcha rápida +
DO + 1.1	Marcha lenta	Posición alcanzada	Marcha lenta	Marcha lenta +
DO + 1.2	Desplazamiento +	Desplazamiento +	Desplazamiento +	Marcha rápida -
DO + 1.3	Desplazamiento -	Desplazamiento -	Desplazamiento -	Velocidad lenta -

Frecuencia max.

Lectura de recorrido

En esta lista desplegable se puede ajustar la frecuencia de contaje máxima de las señales de lectura de recorrido (señales del encoder A, B, N) en una escala fija.

Si la CPU no soporta esta función, la lista desplegable está atenuada.

El ajuste predeterminado es "60 kHz".

Señales cualificadoras

En esta lista desplegable es posible ajustar la frecuencia máxima de las señales de medición de longitud y sensor del punto de referencia en una escala fija.

Si la CPU no soporta esta función, la lista desplegable está atenuada.

El ajuste predeterminado es "10 kHz".

Aproximación a destino (digital)

Zona de destino

Introduzca en el campo de entrada "Zona de destino" un valor par entre 0 y 2 x 10^8 impulsos. Los valores impares se redondean internamente. La zona de destino está situada de forma simétrica en torno al destino. Si se introduce aquí el valor 0, el destino debe alcanzarse exactamente con el impulso. En este caso se desactivará la vigilancia de la zona de destino y la casilla de opción correspondiente de la ficha "Vigilancia" aparecerá sombreada

El ajuste predeterminado es de "50 impulsos".

Tiempo de vigilancia

En el campo de entrada "Tiempo de vigilancia" se introduce un tiempo entre 0 y 100000 ms. La CPU redondea dicho valor a un ciclo de ejecución de 4 ms.

Con el tiempo de vigilancia, la CPU vigila el valor real y la aproximación a destino.

Si se introduce el valor 0 aquí, se desactivan las vigilancia del valor real y de la aproximación al destino y las casillas de verificación correspondientes en el área "Vigilancia" están atenuadas.

El ajuste predeterminado es "2000 ms".

Vigilancia (digital)

Descripción

Aquí se activa y se desactiva la vigilancia del valor real, de la aproximación al destino y de la zona de destino.

Aproximación a destino

Si ha marcado la casilla de verificación "Aproximación a destino", se vigila si el eje alcanza la zona de destino dentro del tiempo de vigilancia después de alcanzar el punto de desconexión.

Si se ha ajustado un tiempo de vigilancia de 0 ms en "Aproximación a destino", no se podrá activar la vigilancia del valor real y de la aproximación a destino y los campos correspondientes aparecerán sombreados.

Esta vigilancia está desactivada de forma estándar.

Zona de destino

Haciendo clic en la casilla de verificación "Zona de destino" se vigila si, tras alcanzar la zona de destino, el accionamiento se detiene en la posición de destino del movimiento iniciado o si se aleja de ella.

En caso de que la vigilancia reaccione se genera un error externo. Dicho error externo debe ser acusado mediante ERR_A. La vigilancia se desactiva. La vigilancia no se vuelve a activar hasta que no se inicia otro desplazamiento.

Esta vigilancia está desactivada de forma estándar.

Valor real

Si se hace clic en la casilla de verificación "Valor real", se vigila el movimiento del eje desde el inicio del desplazamiento hasta alcanzar el punto de desconexión. Si el eje no se desplaza al menos un impulso en el sentido predeterminado durante el tiempo de vigilancia, se interrumpirá el movimiento.

Si se ha ajustado un tiempo de vigilancia de 0 ms en "Aproximación a destino", no se podrá activar la vigilancia del valor real y de la aproximación a destino y los campos correspondientes aparecerán sombreados.

La CPU no detecta el fallo de una entrada digital. Mediante la vigilancia del valor real se puede detectar el fallo de una señal de sensor. Este fallo puede tener las siguientes causas:

- fallo de la entrada digital
- rotura de hilo
- defecto del encoder
- error en la etapa de potencia

Como ajuste predeterminado está activada esta vigilancia.

Eje

Tipo de eje (digital)

Descripción

En "Tipo de eje" seleccione si desea controlar un eje lineal o un eje rotativo.

El ajuste predeterminado es "Eje lineal".

Inicio y fin del final de carrera por software

El área de trabajo de un eje lineal está entre el inicio y el final del final de carrera por software. Los finales de carrera por software forman parte del área de trabajo. Introduzca en los campos de entrada correspondientes valores entre - 5 x 10^8 y 5 x 10^8 impulsos. El inicio del final de carrera por software siempre debe ser menor que el final del final de carrera por software.

Los finales de carrera por software se vigilan cuando el eje está sincronizado y la vigilancia del área de trabajo está activada.

El ajuste predeterminado es de "-100000000 impulsos" para el inicio y "100000000 impulsos" para el final del final de carrera por software.

Fin de eje rotativo

En el campo de entrada "Fin eje rotativo" introduzca un valor entre 1 y 10^9 impulsos.

El valor "Fin de eje rotativo" es el valor teóricamente más grande que puede alcanzar el valor real. Tiene la misma posición física que el inicio del eje rotativo (0).

El ajuste predeterminado es de "100000 impulsos".

Medición de longitud (digital)

Descripción

La función "Medición de longitud" permite medir longitudes durante un desplazamiento. El inicio y el fin de la medición de longitud tienen lugar mediante flancos en la entrada digital "Medición de longitud". Seleccione el tipo de flanco en la lista desplegable "Detección".

Dispone de las siguientes opciones:

- OFF
- Principio/fin con flanco ascendente
  Un flanco ascendente en la entrada digital inicia la medición de longitud, el siguiente flanco ascendente la finaliza.
- Principio/fin con flanco descendente
  Un flanco descendente en la entrada digital inicia la medición de longitud, el siguiente flanco descendiente lo finaliza.
- Inicio con flanco ascendente, fin con flanco descendente
  Un flanco ascendente en la entrada digital inicia la medición de longitud, el siguiente flanco descendente la finaliza.
- Inicio con flanco descendente, fin con flanco ascendente
  Un flanco descendente en la entrada digital inicia la medición de longitud, el siguiente flanco ascendente la finaliza.

El ajuste predeterminado es "OFF".

Punto de referencia (digital)

Coordenadas

Tras una transición STOP-RUN de la CPU y si el eje todavía no está sincronizado, el valor real se pone al valor de la coordenada del punto de referencia.

Tras una búsqueda del punto de referencia, se asigna al punto de referencia el valor de la coordenada del punto de referencia.

Tratándose de un eje lineal, el valor de la coordenada del punto de referencia debe estar comprendido dentro del área de trabajo (incluidos el inicio y el final del final de carrera de software).

Tratándose de un eje rotativo, el valor de la coordenada del punto de referencia debe encontrarse en el rango de 0 a "Fin del eje rotativo - 1".

El ajuste predeterminado es de "0 impulsos".

Decalaje del punto de referencia con respecto al sensor

En la lista desplegable "Decalaje del punto de referencia con respecto al sensor", elija si el punto de referencia se encuentra en sentido más (+) (los valores reales se incrementan) o en sentido menos (-) (los valores reales se decrementan) con respecto al sensor del punto de referencia.

El ajuste predeterminado es "en sentido +".

Vigilancia del eje (digital)

Área de trabajo

Para un eje lineal, aquí se puede establecer si se va a vigilar el área de trabajo. Si se hace clic en la casilla de verificación "Área de trabajo", se comprueba si el valor real de posición se encuentra fuera de los finales de carrera por software. La vigilancia tiene efecto solamente en caso de que el eje esté sincronizado.

Las coordenadas de los finales de carrera por software pertenecen al área de trabajo.

Si se dispara la vigilancia se interrumpe el desplazamiento.

Como ajuste predeterminado está activada esta vigilancia.

Rango de desplazamiento

Con la vigilancia del rango de desplazamiento se comprueba si se abandona el rango de desplazamiento admisible de - 5 x 10^8 a + 5 x 10^8. El rango de desplazamiento se vigila siempre. La vigilancia no se puede desactivar (siempre está activada en el parámetro "Vigilancia").

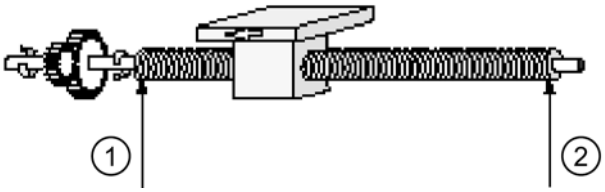
Si se dispara la vigilancia se interrumpe el desplazamiento.

Eje lineal

Descripción

El **eje lineal** es un eje que tiene un margen de desplazamiento físico limitado.

El área de trabajo de un eje lineal está limitada por los finales de carrera por software.

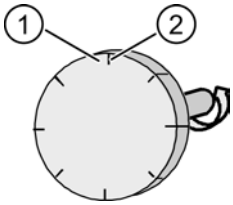


① Inicio físico
② Fin físico

Eje rotativo

Descripción

El **eje rotativo** es un eje cuyo rango de desplazamiento no está limitado por topes mecánicos.



① Valor más alto visualizable = Final del eje rotativo - 1

② Inicio del eje rotativo (coordenada 0) = Final del eje rotativo

Una vuelta del eje rotativo comienza por la coordenada "Cero" y termina por la coordenada "Final del eje rotativo" Físicamente, la coordenada "Cero" es idéntica a la coordenada "Final del eje rotativo" (= 0). En este punto salta la visualización del valor real de posición. La visualización del valor real de posición siempre es positiva.

Encoder

N.º de incrementos por vuelta

En el campo de entrada "N.º de incrementos por vuelta" se introduce el número de incrementos del encoder en una vuelta completa. Dicho valor se toma de la descripción del encoder.

La CPU evalúa los incrementos por cuadruplicado, es decir, un incremento equivale a cuatro impulsos.

El ajuste predeterminado es de "1000 incrementos".

Sentido de contaje

Con "Sentido de contaje" se adapta el sentido de la lectura del recorrido al sentido del movimiento del eje.

Haga clic en la opción "Normal" de la lista desplegable "Sentido de contaje", si los impulsos de contaje ascendentes corresponden a valores reales de posición ascendentes.

Haga clic en la opción "Invertido" de la lista desplegable "Sentido de contaje", si los impulsos de contaje ascendentes corresponden a valores reales de posición descendentes.

Tenga en cuenta también todos los sentidos de giro de los elementos de transmisión (como acoplamientos y transmisiones).

El ajuste predeterminado es la opción "Normal".

Vigilancia

En "Vigilancia" se determina si se deben vigilar los errores de señal del encoder. Esto sólo es posible en encoders cuyo valor en "N.º de incrementos por vuelta" sea divisible por 10 ó 16.

Haciendo clic en la casilla de verificación "Error en señal" se comprueba si el encoder devuelve siempre el mismo número de impulsos entre dos señales cero consecutivas.

Si se utiliza un encoder sin señal cero, habrá que desactivar la vigilancia de los errores en señal.

Esta vigilancia está desactivada de forma estándar.

Diagnóstico

Descripción

Seleccione en el apartado "Canal 0 - diagnóstico" de "Propiedades > Posicionamiento > Canal 0" en qué eventos es necesario generar una alarma de diagnóstico. Como ajuste predeterminado no hay seleccionada ninguna casilla de verificación.

Error en señal

Si se hace clic en la casilla de verificación "Error en señal", se genera una alarma de diagnóstico tan pronto como el encoder deja de transmitir el mismo número de impulsos entre dos señales cero consecutivas. Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - encoder".

Rango de desplazamiento

Si se hace clic en la casilla de verificación "Rango de desplazamiento", se genera una alarma de diagnóstico en cuanto el eje abandona el rango de desplazamiento. La vigilancia correspondiente siempre está activada.

Zona de trabajo

Si se hace clic en la casilla de verificación "Área de trabajo", se genera una alarma de diagnóstico si el valor real de posición se encuentra fuera de los finales de carrera por software. Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - eje".

En caso de que el posicionamiento sea en un eje rotativo no se podrá utilizar esta alarma de diagnóstico. La casilla de verificación correspondiente aparecerá sombreada.

Valor real

Si ha marcado la casilla de verificación "Valor real", se generará una alarma de diagnóstico si durante un movimiento de desplazamiento el eje no se mueve como mínimo un impulso en el sentido indicado dentro del tiempo de vigilancia. En tal caso se interrumpirá el desplazamiento.

El valor real se vigila desde el inicio del desplazamiento hasta alcanzar el punto de desconexión.

Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - accionamiento".

La CPU no detecta el fallo de una entrada digital. Activando la vigilancia del valor real se puede detectar el fallo de una señal de sensor.

Esta casilla de verificación está seleccionada de forma estándar.

Aproximación a destino

Si se hace clic en la casilla de verificación "Aproximación a destino", se genera una alarma de diagnóstico si, tras alcanzar el punto de desconexión, el eje no alcanza la zona de destino durante el tiempo de vigilancia.

Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - accionamiento".

Zona de destino

Si se ha hecho clic en la casilla de verificación "Zona de destino", se genera una alarma de diagnóstico si el accionamiento no se detiene en una posición de destino del movimiento iniciado o si se aleja de ella después de alcanzar la zona de destino.

Para ello se debe activar la vigilancia correspondiente en el apartado "Canal 0 - accionamiento".

Función Acoplamiento punto a punto

Protocolo ASCII

Transferencia PtP (ASCII)

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "ASCII" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "ASCII" hay 4 grupos de parámetros: transferencia, delimitador de fin, recepción de datos y ocupación de las señales.

Este apartado se ocupa de la transferencia PtP y está dividido en 3 secciones.

Velocidad de transferencia

En la lista desplegable "Velocidad de transferencia", seleccione la velocidad de transferencia de datos en bits/s (baudios). Puede seleccionar un valor entre 300 y 38400 baudios. En modo dúplex, la velocidad de transferencia máxima es de 19200 baudios.

El ajuste predeterminado es "9600 baudios".

Trama

Los datos se transmiten en una trama a través de la interfaz serie. Para cada trama se dispone de dos formatos de datos.

- Bits de datos
  Seleccione en la lista desplegable "Bits de datos" la cantidad de bits que representarán un carácter.
  El ajuste predeterminado es "8 bit".
- Bits de parada
  En la lista desplegable "Bits de parada", seleccione el número de bits que se pospondrán a cada carácter a transmitir durante la transmisión. Los bits de parada marcan el final de un carácter.
  El ajuste predeterminado es "1 bit".
- Paridad
  En la lista desplegable "Paridad" puede seleccionar si elevar el grado de protección de la transmisión mediante un bit de paridad. El parámetro de paridad "Sin" significa que no se transmitirá ningún bit de paridad. **Si se ha ajustado 7 bits de datos, no es posible seleccionar la opción "Sin".**
  Existe la posibilidad de ampliar una secuencia de bits de información añadiéndole otro bit, el bit de paridad, que completa el valor de todos los bits con un valor añadido ("0" o "1"), dotándoles así del mismo estado (par o impar).
  El ajuste predeterminado es "Par".

Transferencia de control del flujo de datos

- Control del flujo de datos
  Aquí solamente pueden ajustarse parámetros si está seleccionado el parámetro de control del flujo de datos "XON/XOFF".
  El ajuste predeterminado es "Sin".
- Carácter XON
  Introduzca un código para el carácter XON en el campo de entrada "Carácter XON".
  Introduzca el código en formato hexadecimal.
  Como ajuste predeterminado se indica "11".
- Carácter XOFF
  Introduzca un código para el carácter XOFF en el campo de entrada "Carácter XOFF".
  Introduzca el código en formato hexadecimal.
  Los valores de entrada posibles dependen de si se transfieren los datos con 7 u 8 bits de datos:
  - con 7 bits de datos: de 0 a 7FH
  - con 8 bits de datos: de 0 a FFH
    Como ajuste predeterminado se indica "13".
- Esperar a XON tras XOFF
  Introduzca en el campo de entrada "Esperar a XON tras XOFF" el tiempo que debe esperar el módulo durante la transmisión hasta recibir el carácter XON del interlocutor.
  Se puede introducir un valor entre 20 y 65530 ms (dividido en intervalos de 10 ms).
  Como ajuste predeterminado se indica "20000 ms".

Delimitador de fin PtP (ASCII)

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "ASCII" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "ASCII" hay 4 áreas PtP: transferencia, delimitador de fin, recepción de datos y ocupación de las señales.

Este apartado se ocupa del delimitador de fin PtP y está dividido en 2 secciones.

Detección de un telegrama de recepción

Seleccione la opción deseada en la lista desplegable "Indicador de fin de un telegrama de recepción". Existen tres opciones disponibles: "Transcurrido el tiempo de retardo de caracteres", "Mediante delim. de fin" y "Tras recibir un número de caracteres fijo".

- Transcurrido el tiempo de retardo de caracteres
  La elección "Transcurrido el tiempo de retardo de caracteres" en la lista desplegable permite establecer el criterio de fin de un telegrama una vez transcurrido el tiempo de retardo parametrizado. El tiempo de retardo de caracteres define entonces la distancia máxima permitida entre 2 caracteres recibidos consecutivamente.
  Esta opción está activada de forma estándar.
- Tiempo de retardo de caracteres
  En el campo de entrada "Tiempo de retardo de caracteres" debe introducir el tiempo en milisegundos y comprendido entre 1 y 65535.
  El tiempo mínimo de retardo de caracteres que puede seleccionarse depende de la velocidad de transferencia.

Velocidad de transferencia	Tiempo mínimo de retardo de caracteres
300 bits/s	130 ms
600 bits/s	65 ms
1200 bits/s	32 ms
2400 bits/s	16 ms
4800 bits/s	8 ms
9600 bits/s	4 ms
19200 bits/s	2 ms
38400 bits/s	1 ms

El ajuste predeterminado es "4 ms".

- Mediante delim. de fin
  Con la elección "Mediante delim. de fin" en la lista desplegable se determina que el final de un telegrama se detecte con el delimitador de fin.
- Tiempo de vigilancia si falta el indicador de fin
  En el campo de entrada "Tiempo de vigilancia si falta el indicador de fin" hay un tiempo de retardo de caracteres parametrizado que se utiliza en caso de que falte el indicador de fin. Esto es válido en los ajustes siguientes del indicador de fin:
  - Tras recibir un número fijo de caracteres
  - Tras recibir el/los carácter(es) de fin
    Aquí también podrá cambiar el tiempo de vigilancia. En este caso rigen las mismas condiciones que en el tiempo de retardo de caracteres.
    Deberá realizar manualmente los cálculos del emisor y la evaluación del carácter de control de bloque del receptor.
    Como ajuste predeterminado se indica "4 ms".
- Carácter de fin
  Con el ajuste "Mediante delim. de fin" del parámetro "Indicador de fin de un telegrama de recepción" se determina que el fin de un telegrama esté marcado con uno o dos delimitadores de fin definidos. Se puede trabajar con uno o con dos caracteres de fin. Si lo desea, también puede establecer la recepción adicional de uno o dos caracteres después del carácter de fin. Puede utilizar estos caracteres, por ejemplo, para enviar también un carácter de control de bloque (BCC).
  Seleccione una de las siguientes opciones de la lista desplegable correspondiente:
  - 1. Carácter
  - 1. Carácter con 1 BCC
  - 1. Carácter con 2 BCC
  - 1. Y 2º carácter
  - 1. Y 2º carácter con 1 BCC
  - 1. Y 2º carácter con 2 BCC
    Si se ha seleccionado esta opción, el ajuste predeterminado es "1er carácter".
- Carácter de fin receptor
  Introduzca en el campo de entrada "1er carácter de fin receptor" y, si correspondiese, en el campo de entrada "2º carácter de fin receptor" un código en cada campo para el indicador de fin correspondiente.
  Los valores de entrada posibles dependen de si se transfieren los datos con 7 u 8 bits de datos:
  - con 7 bits de datos: de 0 a 7FH
  - con 8 bits de datos: de 0 a FFH
    Los valores predeterminados son "3" para el primer carácter de fin y "0" para el segundo.
- Tras recibir un número fijo de caracteres
  La selección "Tras recibir un número fijo de caracteres" de la lista desplegable permite establecer el reconocimiento del final de un telegrama después de recibirse un número determinado de caracteres. La longitud de los telegramas de recepción será, de este modo, siempre la misma.
  Esta opción está desactivada de forma estándar.

- Pausa entre telegramas equivalente al tiempo de vigilancia
  Si ha marcado la casilla de verificación "Pausa entre telegramas equivalente al tiempo de vigilancia", entre el envío de dos telegramas se mantendrá una pausa de la duración del tiempo de vigilancia (por si falta la señal de fin) para que el interlocutor detecte la recepción del telegrama y pueda sincronizarse.
  Esta casilla de verificación está desactivada de forma estándar.
- Longitud del telegrama
  En el campo de entrada "Longitud del telegrama" puede ajustar el número de bytes del que debe componerse el telegrama. Puede introducir un valor entre 1 y 1024 bytes.
  El ajuste predeterminado es "200 bytes".

Enviar con señal de fin

Una vez definido el carácter de fin como señal de fin de un telegrama de recepción, podrá enviar datos con señal de fin.

Como parámetro predeterminado se ha ajustado "Enviar hasta la señal de fin inclusive".

- Enviar hasta la señal de fin inclusive
  En la lista desplegable "Criterio de fin > Enviar hasta delimitador de fin inclusive" se establece la inclusión obligatoria del carácter de fin en los datos que se transmiten. Sólo se enviarán los datos hasta el carácter de fin (éste inclusive), incluso en el caso de que en el SFB conste una longitud mayor de datos.
- Enviar hasta la longitud parametrizada en el bloque
  La lista desplegable "Criterio de fin > Enviar hasta la longitud parametrizada en el bloque" permite establecer que se envíen los datos hasta la longitud parametrizada en el SFB. El último carácter debe ser el carácter de fin.
- Enviar adjuntando automáticamente los caracteres de fin
  La lista desplegable "Criterio de fin > Adjuntar automáticamente los caracteres de fin" permite establecer que se envíen los datos hasta la longitud parametrizada en el SFB. Además, se agregan automáticamente el o los delimitadores de fin, es decir, los delimitadores de fin no pueden estar incluidos en los datos que se van a enviar. Según el número de delimitadores de fin, se envían al interlocutor 1 ó 2 caracteres más de los indicados por el SFB (máximo 1024 bytes).
- Carácter de fin emisor
  Si utiliza un carácter de fin, ahora puede establecer un código HEX. (hexadecimal). Del mismo modo podrá seleccionar si desea utilizar un segundo carácter de fin, pudiendo establecer también el código HEX de este carácter. Introduzca en el campo de entrada "1er carácter de fin emisor" y, si correspondiese, en el campo de entrada "2º carácter de fin emisor" un código en cada campo para el indicador de fin correspondiente.
  Los valores predeterminados son "3" para el primer carácter de fin y "0" para el segundo.

Recepción de datos PtP (ASCII)

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "ASCII" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "ASCII" hay 4 áreas PtP: transferencia, delimitador de fin, recepción de datos y ocupación de las señales.

Este apartado se ocupa de la recepción de datos PtP.

Respaldo de recepción en la CPU

- Borrar búfer de recepción de la CPU durante el arranque
  Haga clic en la casilla de verificación "Borrar búfer de recepción de la CPU durante el arranque". Así se borra el búfer de recepción de la función tecnológica al conectar la CPU (POWER ON) o cuando ésta pase de STOP a RUN.
  Esta casilla de verificación está desactivada de forma estándar.
- Impedir sobrescritura
  Haga clic en la casilla de verificación "Impedir sobrescritura" para que no se sobrescriban los datos del búfer de recepción cuando esté lleno.
  Esta casilla de verificación está activada de forma estándar.
- Aprovechar todo el búfer
  Haga clic en la casilla de verificación "Aprovechar todo el búfer" para aprovechar los 2048 bytes del búfer de recepción. Si se aprovecha todo el búfer, el número de telegramas de recepción respaldados sólo depende de la longitud de los telegramas.
  Esta casilla de verificación está activada de forma estándar.
- Número máx. de telegramas de recepción respaldados
  En el campo de entrada "Número máx. de telegramas de recepción respaldados" puede indicar la cantidad de telegramas de recepción respaldados por el búfer de recepción de la función tecnológica. Para ello hay que desactivar la casilla de verificación "Aprovechar todo el búfer". Puede indicar un valor entre 1 y 10.
  El ajuste predeterminado es "10".

Nota

Si se parametriza aquí el valor "1", se desactiva el parámetro "Impedir sobrescritura" y se leen los datos de recepción cíclicamente en el programa de usuario, se envía siempre un telegrama actual a la CPU.

Ocupación de las señales PtP (ASCII)

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "ASCII" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "ASCII" hay 4 áreas PtP: transferencia, delimitador de fin, recepción de datos y ocupación de las señales.

Este apartado se ocupa de la ocupación de las señales PtP y está dividido en 2 secciones.

Modo de operación

Seleccione uno de los tres modos de operación.

El ajuste predeterminado es "Dúplex (RS 422) modo a 4 hilos punto a punto".

Dúplex (RS 422) modo a 4 hilos punto a punto

Con el botón de opción "Dúplex (RS 422) modo a 4 hilos punto a punto" se selecciona el modo de operación apropiado para acoplamientos punto a punto en modo a 4 hilos. Como ocupación estándar de la línea de recepción dispone entonces de las siguientes variantes:

- Ninguno
- Señal R(A) 5 voltios, señal R(B) 0 voltios
- Señal R(A) 0 voltios, señal R(B) 5 voltios

Dúplex (RS 422) modo a 4 hilos maestro multipunto

Con el botón de opción "Dúplex (RS 422) modo a 4 hilos maestro multipunto" se selecciona el modo de operación apropiado para acoplamiento multipunto (Multipoint) en modo a 4 hilos si la CPU es un maestro. Como ocupación estándar de la línea de recepción existe la siguiente variante:

- Señal R(A) 0 voltios, señal R(B) 5 voltios

Semidúplex (RS 485) modo a 2 hilos

Con el botón de opción "Semidúplex (RS 485) modo a 2 hilos" se selecciona el modo de operación apropiado para acoplamiento punto a punto o multipunto (Multipoint) en modo a 2 hilos. La CPU puede ser maestro o esclavo. Como ocupación estándar de la línea de recepción dispone entonces de las siguientes variantes:

- Ninguno
- Señal R(A) 0 voltios, señal R(B) 5 voltios

Ocupación estándar de la línea de recepción

- Ninguno
  Si selecciona el botón de opción "Ninguno" habrá optado por una configuración exclusiva para los driver especiales para comunicaciones en bus.
- Señal R(A) 5 voltios, señal R(B) 0 voltios
  Con el botón de opción "Señal R(A) 5 voltios, señal R(B) 0 voltios" se selecciona un ajuste que permite detectar una rotura de hilo. (No ajustable con dúplex (RS422) a 4 hilos maestro multipunto ni con semidúplex (RS485) a 2 hilos)
- Señal R(A) 0 voltios, señal R(B) 5 voltios

- Con el botón de opción "Señal R(A) 0 voltios, señal R(B) 5 voltios" se selecciona un ajuste que corresponde al estado de reposo (ningún emisor activo). Con esta opción no es posible detectar una rotura de hilo.
- El ajuste predeterminado es Señal R(A) 5 voltios, señal R(B) 0 voltios.

Protocolo 3964(R)

Transferencia PtP (3964(R))

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "3964(R)" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "3964(R)" hay 3 áreas PtP: transferencia, recepción de datos y ocupación de las señales.

Este apartado se ocupa de la transferencia PtP y está dividido en 3 secciones.

Velocidad de transferencia

En la lista desplegable "Velocidad de transferencia", seleccione la velocidad de transferencia de datos en bits/s (baudios). Puede seleccionar un valor entre 300 y 38400 baudios.

El ajuste predeterminado es "9600 baudios".

Trama

Los datos se transmiten en una trama de caracteres a través de la interfaz serie. Para cada trama se dispone de dos formatos de datos.

- Bits de datos
  Seleccione en la lista desplegable "Bits de datos" la cantidad de bits que representarán un carácter.
  El ajuste predeterminado es "8 bit".
- Bits de parada
  En la lista desplegable "Bits de parada", seleccione el número de bits que se pospondrán a cada carácter a transmitir durante la transmisión. Los bits de parada marcan el final de un carácter.
  El ajuste predeterminado es "1 bit".

- Paridad
  En la lista desplegable "Paridad" puede seleccionar si elevar el grado de protección de la transmisión mediante un bit de paridad. El parámetro de paridad "Sin" significa que no se transmitirá ningún bit de paridad. **Si se ha ajustado 7 bits de datos, no es posible seleccionar la opción "Sin".**
  Una secuencia de bits de información puede ampliarse con un bit adicional, el bit de paridad, que con su valor añadido ("0" ó "1") complemente el valor de todos los bits a un estado predefinido (par o impar).
  El ajuste predeterminado es "par".
- Prioridad
  En la lista desplegable "Prioridad" se selecciona si la CPU debe tener una prioridad alta o baja.
  Un interlocutor tiene prioridad alta cuando su deseo de enviar tiene preferencia sobre el deseo de enviar del otro interlocutor. Un interlocutor tiene prioridad baja cuando su deseo de enviar tiene que esperar al deseo de enviar del otro interlocutor. En el procedimiento 3964(R) es necesario parametrizar ambos interlocutores con distinta prioridad, es decir, uno de los interlocutores obtendrá mayor prioridad que el otro.
  El ajuste predeterminado es "alta".

Parámetros del telegrama

- Con control de bloque
  Haga clic en la casilla de verificación "Con control de bloque" para determinar si los telegramas deben enviarse con o sin comprobación de bloque.
  - Sin control de bloque
    Los parámetros de protocolo pueden parametrizarse libremente.
    Si la CPU detecta la secuencia de caracteres DLE ETX, finaliza la recepción, enviando a su interlocutor los caracteres DLE si ha recibido un bloque sin errores (o NAK en el caso de recibirse un bloque erróneo).
  - Con control de bloque
    Los parámetros de protocolo tienen valores predeterminados.
    Cuando la CPU detecta la secuencia de caracteres DLE ETX BCC, finaliza la recepción. La CPU compara el carácter de control de bloque BCC recibido con la paridad de longitud interna. Si el carácter de control de bloque es correcto y no se presenta ningún otro error de recepción, la CPU envía el carácter DLE. (Si se detectase algún error, el carácter enviado al interlocutor sería NAK).

  La casilla de verificación "Con control de bloque" está activada de forma estándar.
- Utilizar valores predeterminados
  Haga clic en la casilla de verificación "Utilizar valores predeterminados" si desea que los parámetros de protocolo tengan valores predeterminados.
  Si el módulo detecta la secuencia de caracteres DLE ETX, finaliza la recepción, enviando a su interlocutor los caracteres DLE si ha recibido un bloque sin errores (o NAK en el caso de recibirse un bloque erróneo).
  Si no desea utilizar valores predeterminados, puede definir los siguientes parámetros de forma individual.
  La casilla de verificación "Utilizar valores predeterminados" está activada de forma estándar.
- Tiempo de retardo de caracteres
  El campo de entrada "Tiempo de retardo de caracteres" permite establecer el intervalo máximo de tiempo entre dos caracteres recibidos en un mismo telegrama. Se puede introducir un valor entre "20 ms" y "65530 ms" en incrementos de 10 ms.
  El tiempo mínimo de retardo de caracteres que puede seleccionarse depende de la velocidad de transferencia.

Velocidad de transferencia	Tiempo mínimo de retardo de caracteres
300 bits/s	60 ms
600 bits/s	40 ms
1200 bits/s	30 ms
2400 bits/s	20 ms
4800 bits/s	20 ms
9600 bits/s	20 ms
19200 bits/s	20 ms
38400 bits/s	20 ms

El ajuste predeterminado es "220 ms".

- Tiempo de retardo de acuse
  En el campo de entrada "Tiempo de retardo de acuse" se establece el tiempo máximo permitido que puede transcurrir hasta el acuse del interlocutor al establecer una conexión (tiempo transcurrido entre STX y el acuse DLE del interlocutor) o al desconectar (tiempo entre DLE ETX y el acuse DLE del interlocutor). Se puede introducir un valor entre "20 ms" y "65530 ms" en incrementos de 10 ms.
  El tiempo mínimo de retardo de acuse que puede seleccionarse depende de la velocidad de transferencia.

Velocidad de transferencia	Tiempo mínimo de retardo de acuse
300 bits/s	60 ms
600 bits/s	40 ms
1200 bits/s	30 ms
2400 bits/s	20 ms
4800 bits/s	20 ms
9600 bits/s	20 ms
19200 bits/s	20 ms
38400 bits/s	20 ms

El ajuste predeterminado es "2000 ms".

- Intentos de establecer conexión
  En la lista "Intentos de establecer conexión" se establece el número máximo de intentos del módulo de establecer una conexión. Puede seleccionar un valor entre "1" y "255".
  El ajuste predeterminado es "6".
- Intentos de transmisión
  En la lista "Intentos de transmisión" puede establecer el número máximo de intentos del módulo de transmitir un telegrama (incluido el primer telegrama) en caso de producirse algún error. Puede seleccionar un valor entre "1" y "255".
  El ajuste predeterminado es "6".

Recepción de datos PtP (3964(R))

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "3964(R)" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "3964(R)" hay 3 áreas PtP: transferencia, recepción de datos y ocupación de las señales.

Este apartado se ocupa de la recepción de datos PtP.

Búfer de recepción en la CPU

- Borrar búfer de recepción de la CPU durante el arranque
  Haga clic en la casilla de verificación "Borrar búfer de recepción de la CPU durante el arranque". Así se borra el búfer de recepción de la CPU al conectarla (POWER ON) o cuando ésta pase de STOP a RUN.
  Esta casilla de verificación está desactivada de forma estándar.
- Impedir sobrescritura
  Haga clic en la casilla de verificación "Impedir sobrescritura" para que no se sobrescriban los datos del búfer de recepción cuando esté lleno.
  Esta casilla de verificación está activada de forma estándar.
- Aprovechar todo el búfer
  Haga clic en la casilla de verificación "Aprovechar todo el búfer" si desea aprovechar los 2048 bytes del búfer de recepción de la CPU. Si se aprovecha todo el búfer, el número de telegramas de recepción respaldados sólo depende de la longitud de los telegramas.
  Esta casilla de verificación está activada de forma estándar.
- Número máx. de telegramas de recepción respaldados
  En el campo de entrada "Número máx. de telegramas de recepción respaldados" se indica el número de telegramas de recepción respaldados por el búfer de recepción de la CPU. Indique un valor entre 1 y 10.
  El ajuste predeterminado es "10".

Nota

Si se parametriza aquí el valor "1", se desactiva el parámetro "Impedir sobrescritura" y se leen los datos de recepción cíclicamente en el programa de usuario, se envía siempre un telegrama actual a la CPU.

Ocupación de las señales PtP (3964(R))

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "3964(R)" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "3964(R)" hay 3 áreas PtP: transferencia, recepción de datos y ocupación de las señales.

Este apartado se ocupa de la ocupación de las señales PtP.

Ocupación estándar de la línea de recepción

Para el protocolo 3964(R) se utiliza la interfaz X27 (RS 422/485) dúplex (RS 422) a 4 hilos. Éste es el modo de operación más adecuado para el acoplamiento punto a punto en el funcionamiento a 4 hilos.

Como ocupación estándar de la línea de recepción se dispone de las siguientes variantes:

- Ninguno
  Si selecciona el botón de opción "Ninguno" habrá optado por una configuración exclusiva para los driver especiales para comunicaciones en bus.
- Señal R(A) 5 V, señal R(B) 0 V (detección de rotura de hilo)
  Con este botón de opción se selecciona un ajuste que permite la detección de rotura de hilo.
- Señal R(A) 0 V, señal R(B) 5 V
  Con este botón de opción se selecciona un ajuste que corresponde al estado de reposo (ningún emisor activo). Con este ajuste no es posible detectar una rotura de hilo.

El ajuste predeterminado es "Señal R(A) 5 V, señal R(B) 0 V (detección de rotura de hilo)".

Protocolo RK512

Transferencia PtP (RK512)

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "RK512" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "RK512" hay 2 áreas PtP: transferencia y ocupación de las señales.

Este apartado se ocupa de la transferencia PtP y está dividido en 3 secciones.

Velocidad de transferencia

En la lista desplegable "Velocidad de transferencia", seleccione la velocidad de transferencia de datos en bits/s (baudios). Puede seleccionar un valor entre 300 y 38400 baudios.

El ajuste predeterminado es "9600 baudios".

Trama

Los datos se transmiten en una trama a través de la interfaz serie. Para cada trama se dispone de dos formatos de datos.

- Bits de datos
  La lista desplegable "Bits de datos" está sombreada, ya que en RK 512 siempre se mapean los caracteres con 8 bits.
- Bits de parada
  En la lista desplegable "Bits de parada", seleccione el número de bits que se pospondrán a cada carácter a transmitir durante la transmisión. Los bits de parada marcan el final de un carácter.
  El ajuste predeterminado es "1 bit".

- Paridad
  En la lista desplegable "Paridad" puede seleccionar si elevar el grado de protección de la transmisión mediante un bit de paridad. El parámetro de paridad "Sin" significa que no se transmitirá ningún bit de paridad. Existe la posibilidad de ampliar una secuencia de bits de información añadiéndole otro bit, el bit de paridad, que completa el valor de todos los bits con un valor añadido ("0" o "1"), dotándoles así del mismo estado (par o impar).
  El ajuste predeterminado es "par".
- Prioridad
  En la lista desplegable "Prioridad" se selecciona si la CPU debe tener una prioridad alta o baja.
  Un interlocutor tiene prioridad alta cuando su deseo de enviar tiene preferencia sobre el deseo de enviar del otro interlocutor. Un interlocutor tiene prioridad baja cuando su deseo de enviar tiene que esperar al deseo de enviar del otro interlocutor. En RK 512 es necesario parametrizar ambos interlocutores con distinta prioridad, es decir, uno de los interlocutores obtiene mayor prioridad que el otro.
  El ajuste predeterminado es "alta".

Parámetros del telegrama

- Con control de bloque
  Haga clic en la casilla de verificación "Con control de bloque" para determinar si los telegramas deben enviarse con o sin comprobación de bloque.
  - Sin control de bloque
    Los parámetros de protocolo pueden parametrizarse libremente.
    Si la CPU detecta la secuencia de caracteres DLE ETX, finaliza la recepción, enviando a su interlocutor los caracteres DLE si ha recibido un bloque sin errores (o NAK en el caso de recibirse un bloque erróneo).
  - Con control de bloque
    Los parámetros de protocolo tienen valores predeterminados.
    Cuando la CPU detecta la secuencia de caracteres DLE ETX BCC, finaliza la recepción. La CPU compara el carácter de control de bloque BCC recibido con la paridad de longitud interna. Si el carácter de control de bloque es correcto y no se presenta ningún otro error de recepción, la CPU envía el carácter DLE. (Si se detectase algún error, el carácter enviado al interlocutor sería NAK).

  La casilla de verificación "Con control de bloque" está activada de forma estándar.
- Utilizar valores predeterminados
  Haga clic en la casilla de verificación "Utilizar valores predeterminados" si desea que los parámetros de protocolo adopten valores predeterminados.
  Si el módulo detecta la cadena de caracteres DLE ETX, finaliza la recepción, enviando a su interlocutor los caracteres DLE si ha recibido un bloque sin errores (o NAK en el caso de recibirse un bloque erróneo).
  Si no desea utilizar valores predeterminados, puede definir los siguientes parámetros de forma individual.
  La casilla de verificación "Utilizar valores predeterminados" está activada de forma estándar.
- Tiempo de retardo de caracteres
  El campo de entrada "Tiempo de retardo de caracteres" permite establecer el intervalo máximo de tiempo entre dos caracteres recibidos en un mismo telegrama. Se puede introducir un valor entre "20 ms" y "65530 ms" en incrementos de 10 ms.
  El tiempo mínimo de retardo de caracteres que puede seleccionarse depende de la velocidad de transferencia.

Velocidad de transferencia	Tiempo mínimo de retardo de caracteres
300 bits/s	60 ms
600 bits/s	40 ms
1200 bits/s	30 ms
2400 bits/s	20 ms
4800 bits/s	20 ms
9600 bits/s	20 ms
19200 bits/s	20 ms
38400 bits/s	20 ms

El ajuste predeterminado es "220 ms".

- Tiempo de retardo de acuse
  En el campo de entrada "Tiempo de retardo de acuse" se establece el tiempo máximo permitido que puede transcurrir hasta el acuse del interlocutor al establecer una conexión (tiempo transcurrido entre STX y el acuse DLE del interlocutor) o al desconectar (tiempo entre DLE ETX y el acuse DLE del interlocutor). Se puede introducir un valor entre "20 ms" y "65530 ms" en incrementos de 10 ms.
  El tiempo mínimo de retardo de acuse que puede seleccionarse depende de la velocidad de transferencia.

Velocidad de transferencia	Tiempo mínimo de retardo de acuse
300 bits/s	60 ms
600 bits/s	40 ms
1200 bits/s	30 ms
2400 bits/s	20 ms
4800 bits/s	20 ms
9600 bits/s	20 ms
19200 bits/s	20 ms
38400 bits/s	20 ms

Como ajuste predeterminado se indica "2000 ms".

- Intentos de establecer conexión
  En la lista "Intentos de establecer conexión" se establece el número máximo de intentos del módulo de establecer una conexión. Puede seleccionar un valor entre 1 y 255.
  El ajuste predeterminado es "6".
- Intentos de transmisión
  En la lista "Intentos de transmisión" puede establecer el número máximo de intentos del módulo de transmitir un telegrama (incluido el primer telegrama). Puede seleccionar un valor entre "1" y "255".
  El ajuste predeterminado es "6".

Ocupación de las señales PtP (RK512)

Descripción

En "Propiedades > General > PtP > Selección de alarmas" se ajustan las alarmas, el comportamiento en caso de STOP de la CPU y el protocolo. Seleccione "RK512" para el protocolo.

Dependiendo del tipo de protocolo que se ha elegido cambia la subdivisión posterior en la navegación local.

Si se elige "RK 512" hay 2 áreas PtP: transferencia y asignación de las señales.

Este apartado se ocupa de la transferencia PtP y está dividido en 3 secciones.

Ocupación estándar de la línea de recepción

La interfaz X27 (RS 422/485) se utiliza en modo dúplex (RS 422) a 4 hilos. Este es el modo de operación más adecuado para el acoplamiento punto a punto en el funcionamiento a 4 hilos.

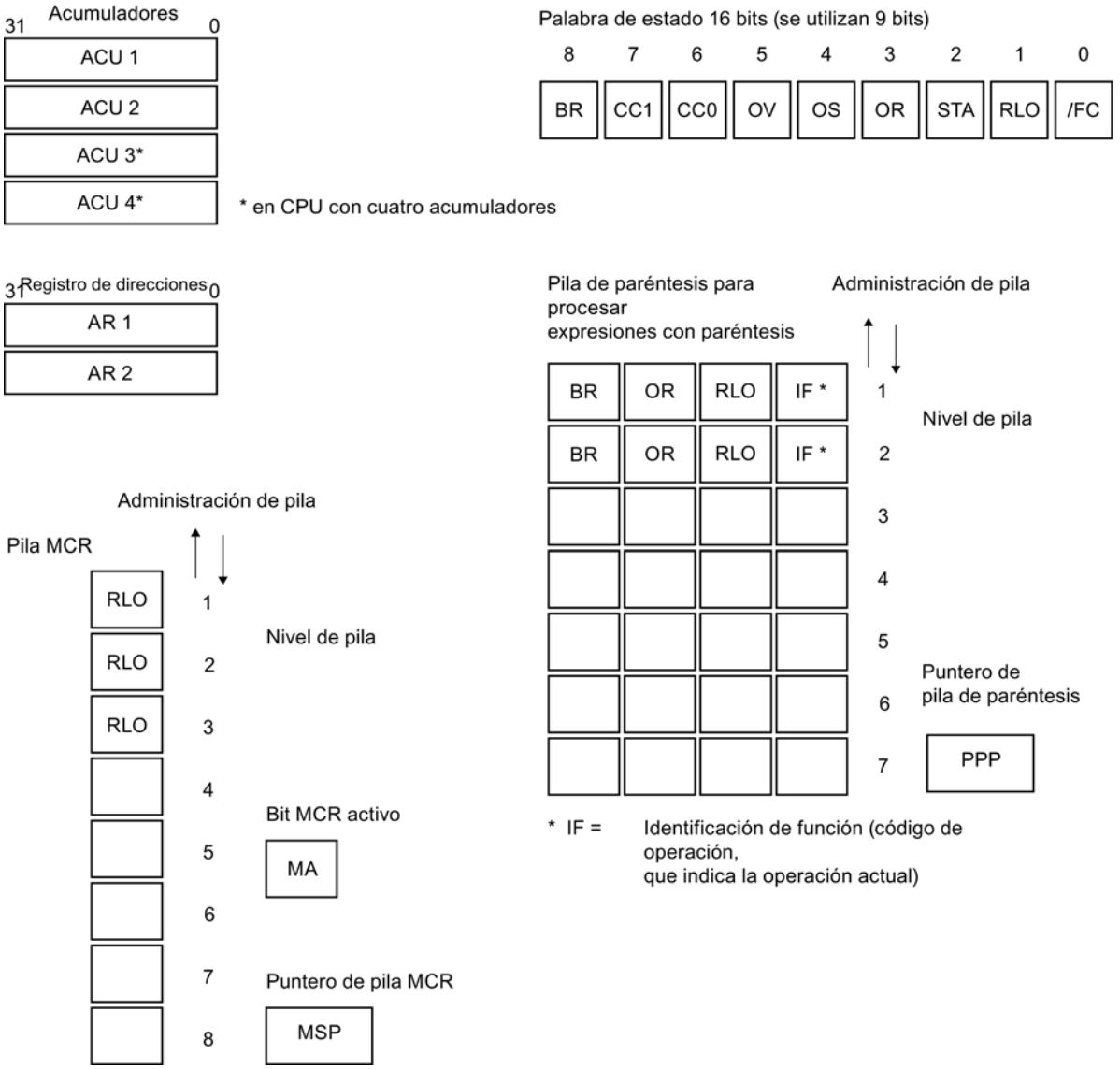
Como ocupación estándar de la línea de recepción se dispone de las siguientes variantes:

- Ninguna
  Si selecciona el botón de opción "Ninguno" habrá optado por una configuración exclusiva para los driver especiales para comunicaciones en bus.
- Señal R(A) 5 V, señal R(B) 0 V (detección de rotura de hilo)
  Con este botón de opción se selecciona un ajuste que permite la detección de rotura de hilo.
- Señal R(A) 0 V, señal R(B) 5 V
  Con este botón de opción se selecciona un ajuste que corresponde al estado de reposo (ningún emisor activo). Con este ajuste no es posible detectar una rotura de hilo.
- El ajuste predeterminado es "Señal R(A) 5 V, señal R(B) 0 V (detección de rotura de hilo)".

Información de la CPU

Registros de la CPU

Resumen



Acumuladores

Descripción

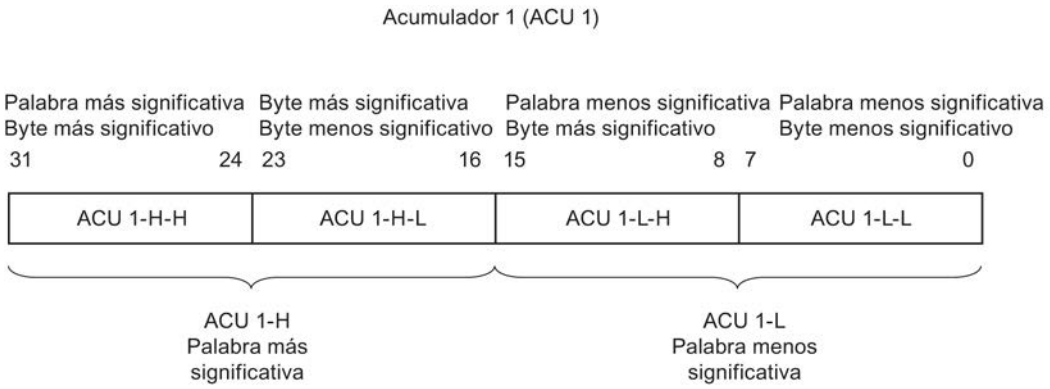
Los acumuladores de 32 bits son registros universales para procesar bytes, palabras y palabras dobles. Para ello, los operandos se cargan en los acumuladores, donde se combinan.

El resultado de la operación permanece siempre en el ACU 1, desde donde se puede transferir a una celda de memoria.

El mecanismo de pila para la gestión de los acumuladores funciona de la siguiente forma:

- Una operación de carga sólo actúa sobre el ACU 1 y guarda el contenido anterior en el ACU 2.
- Una operación de transferencia (operación de copiado) no modifica el contenido de los acumuladores.
- La operación TAK intercambia el contenido del ACU 1 con el del ACU 2.
- El resultado de las combinaciones entre ACU 1 y ACU 2 (operaciones aritméticas, de comparación, Y, O...) siempre se almacena en el ACU 1.

Definición de bytes y palabras en un registro de 32 bits (con el ejemplo del ACU 1)



Direccionamiento

Memorias imagen de proceso (área de memoria de las entradas y salidas)

Área de memoria de las memorias imagen de proceso

La memoria imagen de proceso del OB 1 (MIP OB1) es actualizada cíclicamente por el sistema operativo.

Esta imagen del proceso se almacena en el hardware de la periferia durante cada ciclo de la lista de instrucciones. Por eso, los bits del área de memoria I de la memoria imagen del proceso no varían durante los accesos a las instrucciones. Aunque los bits del área de memoria Q en la memoria imagen del proceso pueden mostrar distintos estados durante el ciclo del programa, al módulo de salida sólo se envía el último estado lógico.

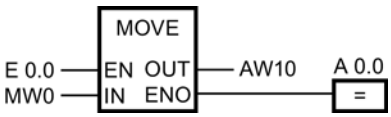
Consulte también: Datos útiles (Página 1732) y Actualizar las memorias imagen de proceso (Página 1725)

Excepciones

En los siguientes casos no se accede a la periferia a través de la memoria imagen de proceso:

- Direccionamiento directo de las entradas y salidas en el área de memoria de la periferia (P)
- Utilización de funciones que procesan inmediatamente los datos de la periferia.

Ejemplo de un acceso normal a la periferia a través de la memoria imagen de proceso



Actualizar QW10 en el área de periferia de las salidas con el valor MW0.

Consulte también

Acceso a las direcciones de periferia (Página 1734)

Acceso directo a la periferia (áreas de memoria PE y PA)

Acceso directo a la periferia

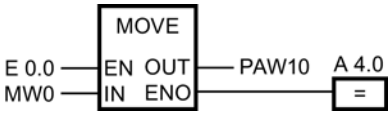
Si fuera necesario que el proyecto actualice los datos de la periferia más rápido de lo que la memoria imagen de proceso se los ofrece, se puede acceder directamente a los módulos de señales en el área de periferia de entradas y salidas (PI, PQ). El estado actual de los módulos de entrada se puede consultar en el área de memoria PI. Los módulos de salida se pueden ajustar escribiendo en el área de memoria PQ.

⚠ PRECAUCIÓN

En el área de periferia no se puede direccionar a bits (byte, bit) concretos. Para activar bits concretos, hay que direccionar los bits, palabras o palabras dobles en operaciones lógicas. Si se escribe un bit actualizado (dentro de un byte, palabra o palabra doble) en el área de memoria PQ, se actualizarán todos los bits de la dirección de esa área de periferia.

En la entrada **Memoria imagen de proceso** se puede encontrar más información sobre la relación entre el área de memoria de la periferia (área de memoria PI, PQ) y la memoria imagen de proceso (áreas de memoria I y Q). Allí también se describe cómo la memoria imagen de proceso memoriza (estabiliza) lo que se escribe y o envía desde los módulos de señales.

Ejemplo de un acceso directo a la periferia a través del área de memoria de la periferia



Actualizar PQW10 en el área de periferia de las salidas con el valor MW0.

Consulte también

Acceso a las direcciones de periferia (Página 1734)

Áreas de direcciones (Página 1735)

Sintaxis explícita para la entrada de constantes

Resumen

Forma abreviada	Forma ampliada
B#	BYTE#
W#	WORD#
DW#	DWORD#
D#	DATE#
T#	TIME#
TOD#	TIME_OF_DAY#
S5T#	S5TIME#
DT#	DATE_AND_TIME#
P#	POINTER#

Resumen de los identificadores de bit, byte, palabra y palabra doble

Resumen

Área de memoria de la CPU	Identificador de operando			
	Bit	Byte	Palabra	Palabra doble
Memoria imagen de proceso de las entradas	I	IB	IW	ID
Memoria imagen de proceso de las salidas	Q	QB	QW	QD
Marcas	M	MB	MW	MD
Datos locales	L	LB	LW	LD
Área directa de periferia de las entradas		PIB	PIW	PID

Área de memoria de la CPU	Identificador de operando			
	Bit	Byte	Palabra	Palabra doble
Área directa de periferia de las salidas		PQB	PQW	PQD
Bloque de datos [OPN] DB [OPN] DI	 DBX DIX	 DBB DIB	 DBW DIW	 DBD DID
Temporizador (T)	T		T	
Contador (C)	C		C	

Nota

Los dos operandos **IB0** e **IB1** (área de memoria I, formato: byte, bytes de dirección 0, 1) direccionan los dos mismos bytes que el operando **IW0** (área de memoria I, formato: palabra, 2 bytes, incremento de byte 0 a bytes de dirección 0, 1).

Identificadores de elementos y bloques

Resumen

T	Temporizador
DB	Bloque de datos globales
DI	Bloque de datos de instancia
FC	Función
FB	Bloque de función
OB	Bloque de organización
Z/C	Contador (depende del nemónico).
SDB	Bloque de datos de sistema
SFC	Función de sistema
SFB	Bloque de función de sistema

Denominación simbólica y numérica de las instrucciones

Descripción

Las instrucciones de la Task Card se componen de funciones (FC), bloques de función (FB), funciones de sistema (SFC) y bloques de función de sistema (SFB) que están identificados internamente por medio de números.

Las tablas siguientes muestran las correspondencias entre la denominación numérica y la simbólica.

Bloques de función (FBs)

Denominación numérica	Denominación simbólica
FB 2	S_RCV
FB 3	S_SEND
FB 4	S_VSTAT
FB 5	S_VSET
FB 6	S_XON
FB 7	S_RTS
FB 8	S_V24
FB 8 (1)	USEND_S
FB 9 (1)	URCV_S
FB 14 (1)	GET_S
FB 15 (1)	PUT_S
FB 40 (1)	FTP_CMD
FB 46	PG_DIAL
FB 47	AS_DIAL
FB 48	SMS_SEND
FB 49	AS_MAIL
FB 52 (1)	PNIO_RW_REC
FB 53	PE_DS3_Write_ET200S
FB 54 (1)	PNIO_ALARM
FB 55	IP_CONFIG
FB 56 (1)	LOGICAL_TRIGGER
FB 60	SET_SW
FB 61	SET_SW_S
FB 62	TIMESTMP
FB 80	LEAD_LAG
FB 81	DCAT
FB 81	S_MODB
FB 82	MCAT
FB 83	IMC
FB 84	SMC
FB 85	DRUM
FB 86	PACK
FB 210	FW_TCP
FB 220	FW_IOT
FB 450 (2)	RED_IN_MGP
FB 451 (2)	RED_OUT_MGP
FB 452 (2)	RED_DIAG_MGP
FB 453 (2)	RED_STATUS_MGP
FB 815	PE_Start_End_Pause
FB 816	PE_CMD

Denominación numérica	Denominación simbólica
FB 817 (1)	PE_I_DEV
(1) en el S7-300 (2) en el S7-400 (3) no en el WinAC (4) en el S7-400, no en el WinAC (5) en el WinAC	

Funciones (FCs)

Denominación numérica	Denominación simbólica
FC 0 (1)	PE_Error_RSP
FC 1 (1)	PE_Start_RSP
FC 1 (1)	DP_SEND
FC 2	CONCAT
FC 2 (1)	PE_End_RSP
FC 2 (1)	DP_RECV
FC 3 (1)	PE_List_Modes_RSP
FC 3 (1)	DP_DIAG
FC 4	DELETE
FC 4 (1)	PE_Get_Mode_RSP
FC 4 (1)	DP_CTRL
FC 5 (1)	PE_PEM_Status_RSP
FC 5 (1)	AG_SEND
FC 6 (1)	PE_Identify_RSP
FC 6 (1)	AG_RECV
FC 7	Asi_3422
FC 7 (1)	PE_Measurement_List_RSP
FC 7 (1)	AG_LOCK
FC 8 (1)	PE_Measurement_Value_RSP
FC 8 (1)	AG_UNLOCK
FC 10 (1)	AG_CNTRL
FC 11	FIND
FC 11 (1)	PNIO_SEND
FC 12 (1)	PNIO_RECV
FC 17	INSERT
FC 17	S_USST
FC 18	S_USSR
FC 19	S_USSI
FC 20	LEFT
FC 21	LEN
FC 22	LIMIT
FC 25	MAX

Denominación numérica	Denominación simbólica
FC 26	MID
FC 27	MIN
FC 31	REPLACE
FC 32	RIGHT
FC 36	ENCO
FC 36	SEL
FC 37	DECO
FC 60	LOC_TIME
FC 61	BT_LT
FC 62	LT_BT
FC 62 (1)	C_CNTRL
FC 63	S_LTINT
FC 74 (3)	I_ABORT
FC 80	TONR
FC 82	RSET
FC 83	SET
FC 84	ATT
FC 85	FIFO
FC 86	TBL_FIND
FC 87	LIFO
FC 88	TBL
FC 89	TBL_WRD
FC 90	WSR
FC 91	WRD_TBL
FC 92	SHRB
FC 93	SEG
FC 98	BCDPL
FC 99	BITSUM
FC 100	RSETI
FC 101	SETI
FC 102	DEV
FC 103	CDT
FC 104	TBL_TBL
FC 105	SCALE
FC 106	UNSCALE
FC 450 (2)	RED_INIT_MGP
FC 451 (2)	RED_DEPA_MGP
(1) en el S7-300 (2) en el S7-400 (3) no en el WinAC (4) en el S7-400, no en el WinAC (5) en el WinAC	

Bloques de función de sistema (SFBs)

Denominación numérica	Denominación simbólica
SFB 0	CTU
SFB 1	CTD
SFB 2	CTUD
SFB 3	TP
SFB 4	TON
SFB 5	TOF
SFB 16 (4)	PRINT
SFB 19 (2)	START
SFB 20 (2)	STOP
SFB 21 (4)	RESUME
SFB 22 (2)	STATUS
SFB 23 (2)	USTATUS
SFB 29 (1)	HS_COUNT_300C
SFB 30 (1)	FREQ_MES_300C
SFB 31 (2)	NOTIFY_8P
SFB 32	DRUM_X
SFB 33 (2)	ALARM
SFB 34 (2)	ALARM_8
SFB 35 (2)	ALARM_8P
SFB 36 (2)	NOTIFY_8P
SFB 37 (2)	AR_SEND
SFB 38 (1)	HSC_A_B_300C
SFB 39 (1)	POS_300C
SFB 44 (1)	ANALOG_300C
SFB 46 (1)	DIGITAL_300C
SFB 47 (1)	COUNT_300C
SFB 48 (1)	FREQUENC_300C
SFB 49 (1)	PULSE_300C
SFB 52	RDREC
SFB 53	WRREC
SFB 54	RALRM
SFB 60 (1)	SEND_PTP_300C
SFB 61 (1)	RCV_PTP_300C
SFB 62 (1)	RES_RCVB_300C
SFB 63 (1)	SEND_RK_300C
SFB 64 (1)	FETCH_RK_300C
SFB 65 (1)	SERVE_RK_300C
SFB 73	RCVREC
SFB 74	PRVREC
SFB 75	SALRM

Denominación numérica	Denominación simbólica
SFB 81	RD_DPAR
SFB 104	IP_CONF
SFB 65001 (5)	CREA_COM
SFB 65002 (5)	EXEC_COM
SFB 65002 (5)	ASYN_COM
(1) en el S7-300 (2) en el S7-400 (3) no en el WinAC (4) en el S7-400, no en el WinAC (5) en el WinAC	

Funciones de sistema (SFCs)

Denominación numérica	Denominación simbólica
SFC 2	SET_RTM
SFC 3	CTRL_RTM
SFC 4	READ_RTM
SFC 5	GADR_LGC
SFC 7 (3)	DP_PRAL
SFC 9 (2)	EN_MSG
SFC 10 (2)	DIS_MSG
SFC 11	DPSYC_FR
SFC 12	D_ACT_DP
SFC 13	DPNRM_DG
SFC 14	DPRD_DAT
SFC 17	ALARM_SQ
SFC 18	ALARM_S
SFC 19	ALARM_SC
SFC 20	BLKMOV
SFC 21	FILL
SFC 22	CREAT_DB
SFC 23	DEL_DB
SFC 24	TEST_DB
SFC 26 (2)	UPDAT_PI
SFC 27 (2)	UPDAT_PO
SFC 28	SET_TINT
SFC 29	CAN_TINT
SFC 30	ACT_TINT
SFC 31	QRY_TINT
SFC 32	SRT_DINT
SFC 33	CAN_DINT
SFC 34	QRY_DINT

Denominación numérica	Denominación simbólica
SFC 35 (4)	MP_ALM
SFC 36	MSK_FLT
SFC 37	DMSK_FLT
SFC 38	READ_ERR
SFC 39	DIS_IRT
SFC 40	EN_IRT
SFC 41	DIS_AIRT
SFC 42	EN_AIRT
SFC 43	RE_TRIGR
SFC 44	REPL_VAL
SFC 46	STP
SFC 47	WAIT
SFC 48 (4)	SNC_RTCB
SFC 49	LGC_GADR
SFC 50	RD_LGADR
SFC 51	RDSYSST
SFC 52	WR_USMSG
SFC 54 (2)	RD_DPARM
SFC 55	WR_PARM
SFC 56	WR_DPARM
SFC 57	PARM_MOD
SFC 58	WR_REC
SFC 59	RD_REC
SFC 60 (4)	GD_SND
SFC 61 (4)	GD_RCV
SFC 62 (2)	CONTROL
SFC 64	TIME_TCK
SFC 64	RD_SINFO
SFC 65 (3)	X_SEND
SFC 66 (3)	X_RCV
SFC 67 (3)	X_GET
SFC 68 (3)	X_PUT
SFC 69 (3)	X_ABORT
SFC 70	GEO_LOG
SFC 71	LOG_GEO
SFC 72 (3)	I_GET
SFC 73 (3)	I_PUT
SFC 75	SET_ADDR
SFC 78 (2)	OB_RT
SFC 79	SETP
SFC 80	RESETP
SFC 81	UBLKMOV
SFC 82 (1) (6)	CREA_DBL

Denominación numérica	Denominación simbólica
SFC 83 (1) (6)	READ_DBL
SFC 84 (1) (6)	WRIT_DBL
SFC 85 (6)	CREA_DB
SFC 87 (2)	C_DIAG
SFC 90 (4)	H_CTRL
SFC 99	WWW
SFC 100 (2)	SET_CLKS
SFC 101	RTM
SFC 102 (1)	RD_DPARA
SFC 103 (3)	DP_TOPOL
SFC 104 (2)	CIR
SFC 105	READ_SI
SFC 106	DEL_SI
SFC 107	ALARM_DQ
SFC 108	ALARM_D
SFC 109	PROTECT
SFC 112	PN_IN
SFC 113	PN_OUT
SFC 114	PN_DP
SFC 126	SYNC_PI
SFC 127	SYNC_PO
SFC 154	DPWR_DAT
(1) en el S7-300 (2) en el S7-400 (3) no en el WinAC (4) en el S7-400, no en el WinAC (5) en el WinAC	

Bloques de organización

Sinopsis de los OB

Bloques de organización

Los bloques de organización constituyen la interfaz entre el sistema operativo de la CPU y el programa de usuario. Los OB permiten ejecutar partes del programa de manera selectiva:

- Al arrancar la CPU
- En ejecución cíclica o con una cadencia determinada
- En caso de aparición de errores
- En caso de aparición de alarmas de proceso.

Los bloques de organización se procesan de acuerdo con la prioridad que tienen asignada.

OB disponibles

No todas las CPU pueden procesar todos los OB disponibles en S7. En la lista de operaciones de la CPU se indica qué OB están disponibles.

Eventos de arranque y clase de prioridad predeterminada

La tabla siguiente contiene el evento de arranque y la clase de prioridad predeterminada para cada OB.

OB	Evento de arranque	Clase de prioridad predeterminada	Explicación
OB 1	Fin del arranque o fin del OB 1	1	Ciclo libre
OB 10	Alarma horaria 0	2	Sin indicación de hora predeterminada
OB 11	Alarma horaria 1	2	
OB 12	Alarma horaria 2	2	
OB 13	Alarma horaria 3	2	
OB 14	Alarma horaria 4	2	
OB 15	Alarma horaria 5	2	
OB 16	Alarma horaria 6	2	
OB 17	Alarma horaria 7	2	
OB 20	Alarma de retardo 0	3	Sin indicación de hora predeterminada
OB 21	Alarma de retardo 1	4	
OB 22	Alarma de retardo 2	5	
OB 23	Alarma de retardo 3	6	
OB 30	Alarma cíclica 0 (predeterminado: ciclo de 5 s)	7	Alarmas cíclicas
OB 31	Alarma cíclica 1 (predeterminado: ciclo de 2 s)	8	
OB 32	Alarma cíclica 2 (predeterminado: ciclo de 1 s)	9	
OB 33	Alarma cíclica 3 (predeterminado: ciclo de 500 ms)	10	
OB 34	Alarma cíclica 4 (predeterminado: ciclo de 200 ms)	11	
OB 35	Alarma cíclica 5 (predeterminado: ciclo de 100 ms)	12	
OB 36	Alarma cíclica 6 (predeterminado: ciclo de 50 ms)	13	
OB 37	Alarma cíclica 7 (predeterminado: ciclo de 20 ms)	14	
OB 38	Alarma cíclica 8 (predeterminado: ciclo de 10 ms)	15	
OB 40	Alarma de proceso 0	16	Alarmas de proceso
OB 41	Alarma de proceso 1	17	
OB 42	Alarma de proceso 2	18	
OB 43	Alarma de proceso 3	19	
OB 44	Alarma de proceso 4	20	
OB 45	Alarma de proceso 5	21	

OB	Evento de arranque	Clase de prioridad predeterminada	Explicación
OB 46	Alarma de proceso 6	22	
OB 47	Alarma de proceso 7	23	
OB 55	Alarma de estado	2	Alarmas DPV1
OB 56	Alarma de actualización	2	
OB 57	Alarma de fabricante	2	
OB 61	Alarma de sincronismo 1	25	Alarmas de sincronismo
OB 62	Alarma de sincronismo 2	25	
OB 63	Alarma de sincronismo 3	25	
OB 64	Alarma de sincronismo 4	25	
OB 80	Error de tiempo	26, 28 [1]	Alarmas de error asíncronas
OB 81	Error de fuente de alimentación	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 82	Alarma de diagnóstico	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 83	Alarma de presencia de módulo	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 84	Error de hardware en CPU	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 85	Error de ejecución del programa	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 86	Fallo de un aparato de ampliación, de un sistema maestro DP o una estación en la periferia descentralizada	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 87	Error de comunicación	26, 28 [1] para S7-300, 25, 28 [1] para S7-400	
OB 88	Interrupción del procesamiento	28	
OB 90	Rearranque completo (en caliente) o arranque en frío o borrado de un bloque en procesamiento en el OB 90 o carga de un OB 90 en la CPU o fin de OB 90	29 [2)]	Ciclo en segundo plano
OB 100	Rearranque completo (en caliente)	27 [1]	Arranque
OB 101	Rearranque	27 [1]	
OB 102	Arranque en frío	27 [1]	

OB	Evento de arranque	Clase de prioridad predeterminada	Explicación
OB 121	Error de programación	Prioridad del OB que provoca errores	Alarmas de error síncronas
OB 122	Error de acceso a periferia	Prioridad del OB que provoca errores	
[1] Las clases de prioridad 27 y 28 son válidas en el modelo de clase de prioridad del arranque. [2] A la clase de prioridad 29 le corresponde la prioridad 0.29. En consecuencia, el ciclo en segundo plano tiene una prioridad inferior a la del ciclo libre.			

Programa cíclico (OB 1)

Descripción

El sistema operativo de la CPU S7 procesa el OB 1 de modo cíclico: Una vez finalizado el procesamiento del OB 1, el sistema operativo empieza nuevamente a procesar el OB 1. El procesamiento cíclico del OB 1 se inicia una vez finalizado el arranque. En el OB 1 pueden llamarse instrucciones.

Funcionamiento del OB 1

El OB 1 tiene la prioridad más baja de todos los OB con vigilancia en tiempo de ejecución. A excepción del OB 90, todos los demás OB pueden interrumpir el procesamiento del OB 1. Los siguientes eventos hacen que el sistema operativo llame el OB 1:

- Fin del procesamiento del arranque
- Fin del procesamiento del OB 1 (del ciclo anterior).

Una vez finalizado el procesamiento del OB 1, el sistema operativo envía datos globales. Antes de rearrancar el OB 1, el sistema operativo escribe la memoria imagen de proceso de las salidas en los módulos de salida, actualiza la memoria imagen de proceso de las entradas y recibe datos globales para la CPU.

S7 ofrece una vigilancia del tiempo de ciclo máximo, lo que garantiza el tiempo de reacción máximo. El tiempo de ciclo máximo predeterminado es de 150 ms; es posible reparametrizar este valor, así como reiniciar la vigilancia de tiempo en cada punto del programa con la instrucción "RE_TRIGR". Si el programa excede el tiempo de ciclo máximo para el OB 1, el sistema operativo llama el OB 80 (OB de error de tiempo). Si el OB 80 no está programado, la CPU pasa al estado operativo STOP.

Además de vigilar el tiempo de ciclo máximo, también se garantiza el mantenimiento de un tiempo de ciclo mínimo. Para ello, el sistema operativo retarda el inicio de un ciclo nuevo (escritura de la memoria imagen de proceso de las salidas en los módulos de salida) hasta que se ha alcanzado el tiempo de ciclo mínimo.

La modificación de los parámetros de tiempo de ciclo máximo y mínimo se realiza mediante configuración.

Datos locales del OB 1

La tabla siguiente contiene las variables temporales (TEMP) del OB 1. Se han usado como nombres de variable los nombres predeterminados del OB 1.

Variable	Tipo de datos	Descripción
OB1_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: El OB 1 está activo
OB1_SCAN_1	BYTE	• B#16#01: Finalización del rearranque completo (en caliente) • B#16#02: Finalización del rearranque • B#16#03: Finalización del ciclo libre • B#16#04: Finalización del arranque en frío • B#16#05: Primer ciclo de OB 1 de la nueva CPU maestra después de la conmutación de maestro a reserva y STOP del maestro anterior
OB1_PRIORITY	BYTE	Clase de prioridad: 1
OB1_OB_NUMBR	BYTE	N.º de OB (01)
OB1_RESERVED_1	BYTE	Reservado
OB1_RESERVED_2	BYTE	Reservado
OB1_PREV_CYCLE	INT	Tiempo de ejecución del ciclo anterior (ms)
OB1_MIN_CYCLE	INT	Tiempo de ciclo mínimo (ms) desde el último arranque
OB1_MAX_CYCLE	INT	Tiempo de ciclo máximo (ms) desde el último arranque
OB1_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

OBs de alarma horaria (de OB 10 a OB 17)

Descripción

S7 ofrece hasta ocho OBs (del OB 10 al OB 17), que se ejecutan una sola vez o periódicamente. Se puede parametrizar la CPU para que estos OBs se procesen en los siguientes intervalos:

- Una sola vez
- Una vez por minuto
- Una vez por hora
- Una vez al día
- Una vez por semana
- Una vez al mes

- Una vez al año
- A final de mes

Nota

Si se procesa el OB de alarma horaria con periodicidad mensual, solo se admiten en la fecha de arranque los días 1, 2, ... 28.

Funcionamiento de los OBs de alarma horaria

Para iniciar una alarma horaria, la alarma primero debe ajustarse y luego activarse. Existen tres posibilidades de inicio:

- Inicio automático de la alarma horaria. Se lleva a cabo si tanto el ajuste como la activación de la alarma horaria se han realizado mediante configuración. La tabla siguiente muestra las posibilidades básicas para activar una alarma horaria mediante configuración.
- Ajustar la alarma horaria mediante configuración y activarla llamando a la instrucción "ACT_TINT (Página 6889)" desde el programa.
- Ajustar la alarma horaria llamando a la instrucción "SET_TINT (Página 6887)" y activarla llamando a la instrucción "ACT_TINT (Página 6889)".

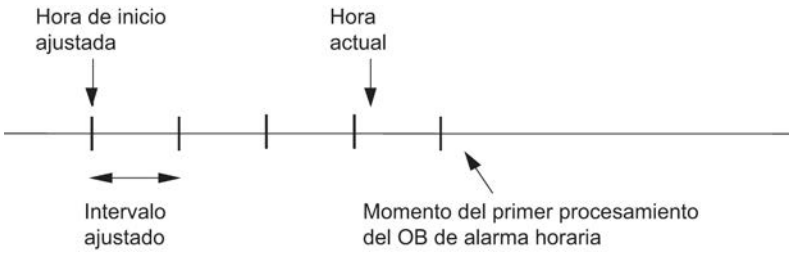
Intervalo	Descripción
No activado	El OB de alarma horaria no se procesa, ni siquiera cuando está cargado en la CPU. Se puede activar la alarma horaria llamando la instrucción "ACT_TINT".
Activación única	El OB de alarma horaria se anula automáticamente después de ser procesado una vez del modo indicado. Con la instrucción "SET_TINT", el programa puede ajustar de nuevo la alarma horaria y volver a activarla con la instrucción "ACT_TINT".
Activación periódica	En el momento de la alarma horaria, la CPU determina, a partir de la hora actual y del período fijado, el próximo momento de inicio de la alarma horaria.

Nota

Si se configura una alarma horaria para que el OB se procese una sola vez, la fecha y hora no pueden encontrarse en el pasado (en relación con el reloj de tiempo real de la CPU).

Si se configura una alarma horaria para que el OB se procese periódicamente, y la fecha y hora se encuentran en el pasado, el OB de alarma horaria se procesará en el próximo momento adecuado según la hora actual. La figura siguiente muestra el primer procesamiento de un OB de alarma horaria, en caso de que el momento de inicio se encuentre en el pasado y se haya ajustado la activación periódica.

Se puede bloquear la llamada al OB de alarma horaria con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".



Condiciones desfavorables para los OBs de alarma horaria

Dado que las alarmas horarias solo se producen en determinados intervalos, algunas circunstancias pueden entorpecer el funcionamiento de los correspondientes OBs durante la ejecución del programa. La tabla siguiente muestra algunas de esas circunstancias y describe sus efectos sobre el procesamiento del OB de alarma horaria.

Condición	Resultado
El programa llama la instrucción "CAN_TINT (Página 6888)" y anula una alarma horaria.	El sistema operativo borra el evento de arranque (fecha y hora) de la alarma horaria. Si desea que el OB vuelva a ser llamado, reajuste y active de nuevo el evento de arranque.
El programa ha intentado activar un OB de alarma horaria que en el momento de la activación no estaba cargado en la CPU.	El sistema operativo llama el OB 85. Si el OB 85 no está programado (cargado en la CPU), la CPU pasa al estado operativo STOP.
Al sincronizar o corregir el reloj del sistema de la CPU, se ha adelantado la hora y se ha omitido el evento de arranque, la fecha o la hora de un OB de alarma horaria.	El sistema operativo llama el OB 80 y codifica en el OB 80 el número del OB de alarma horaria y la información de evento de arranque. Tras ello, el sistema operativo procesa el OB de alarma horaria una vez, independientemente del número de veces que se hubiera fijado en principio para ese OB. La información de evento de arranque del OB 80 muestra la fecha y la hora en que se ha omitido por primera vez el OB de alarma horaria.
Al sincronizar o corregir el reloj del sistema de la CPU, se ha retrasado la hora y se ha repetido el evento de arranque, la fecha o la hora de un OB de alarma horaria.	CPUs S7-400: Si el OB de alarma horaria se activó antes de retrasar el reloj, no se volverá a llamar en las horas que ya hayan pasado. CPUs S7-300: Se ejecuta el OB de alarma horaria.

La CPU ejecuta un nuevo rearranque completo (en caliente) o un arranque en frío.	Cada OB de alarma horaria que se ha configurado mediante una instrucción vuelve a adoptar la configuración establecida inicialmente. Si se ha configurado una alarma horaria para activación única del OB correspondiente y se ha ajustado y activado mediante configuración, el sistema operativo llamará el OB una sola vez después de un rearranque completo (en caliente) o un arranque en frío, siempre que el momento de inicio configurado se encuentre en el pasado (en relación con el reloj en tiempo real de la CPU).
Un OB de alarma horaria sigue procesándose aunque ya se ha producido el evento de arranque correspondiente al siguiente intervalo.	El sistema operativo llama el OB 80. Si el OB 80 no está programado, la CPU pasa al estado operativo STOP. De lo contrario, después de procesarse el OB 80 y el OB de alarma horaria, se reactiva el procesamiento de OB solicitado.

Datos locales de los OBs de alarma horaria

La tabla siguiente contiene las variables temporales (TEMP) de un OB de alarma horaria. Se han escogido como nombres de variable los nombres predeterminados del OB 10.

Variable	Tipo de datos	Descripción
OB10_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: La alarma está activa
OB10_STRT_INFO	BYTE	B#16#11: Solicitud de inicio para el OB 10 (B#16#12: Solicitud de inicio para el OB 11) : : (B#16#18: Solicitud de inicio para el OB 17)
OB10_PRIORITY	BYTE	Clase de prioridad parametrizada; valor predeterminado: 2
OB10_OB_NUMBR	BYTE	N.º de OB (de 10 a 17)
OB10_RESERVED_1	BYTE	Reservado
OB10_RESERVED_2	BYTE	Reservado
OB10_PERIOD_EXE	WORD	El OB se procesa con el intervalo indicado: W#16#0000: Una sola vez W#16#0201: Una vez por minuto W#16#0401: Una vez por hora W#16#1001: Una vez al día W#16#1201: Una vez por semana W#16#1401: Una vez al mes W#16#1801: Una vez al año W#16#2001: A final de mes
OB10_RESERVED_3	INT	Reservado
OB10_RESERVED_4	INT	Reservado
OB10_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

OBs de alarma de retardo (de OB 20 a OB 23)

Descripción

S7 ofrece hasta cuatro OB (del OB 20 al OB 23) que se ejecutan una vez transcurrido un retardo parametrizable. Todos los OB de alarma de retardo se ejecutan mediante llamada de la instrucción "SRT_DINT (Página 6892)". El retardo es un parámetro de entrada de la instrucción.

Si el programa llama la instrucción "SRT_DINT (Página 6892)", debe indicarse el número de OB, el retardo y un identificador específico de usuario. El OB en cuestión se ejecuta una vez transcurrido el retardo especificado. También es posible anular el procesamiento de una alarma de retardo que todavía no se ha iniciado.

Funcionamiento de los OB de alarma de retardo

Una vez transcurrido el tiempo que se ha transmitido en ms a la instrucción "SRT_DINT (Página 6892)" junto con un número de OB, el sistema operativo ejecuta el OB correspondiente.

Si se desean utilizar OB de alarma de retardo, deben realizarse las siguientes tareas:

- Llamar la instrucción "SRT_DINT (Página 6892)".
- Cargar el OB de alarma de retardo en la CPU como parte del programa de usuario.

Los OB de alarma de retardo solo se procesan si la CPU está en el estado operativo RUN. El rearranque completo (en caliente) y el arranque en frío borran todos los eventos de arranque de los OB de alarma de retardo. Las alarmas de retardo todavía no activadas pueden anularse con la instrucción "CAN_DINT (Página 6893)".

El retardo se mide con una precisión de 1 ms. Un tiempo de retardo puede reiniciarse inmediatamente una vez transcurrido. La instrucción "QRY_DINT (Página 6894)" permite conocer el estado de una alarma de retardo.

El sistema operativo llama un OB para errores asíncronos en caso de que se produzca uno de los siguientes eventos:

- El sistema operativo intenta ejecutar un OB que no está cargado y cuyo número se ha indicado al llamar la instrucción "SRT_DINT (Página 6892)".
- Se produce el siguiente evento de arranque para una alarma de retardo antes de que haya finalizado el procesamiento del correspondiente OB de alarma de retardo.

Se puede bloquear la llamada del OB de alarma de retardo con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales de los OB de alarma de retardo

La tabla siguiente contiene las variables temporales (TEMP) de un OB de alarma de retardo. Se han escogido como nombres de variable los nombres predeterminados del OB 20.

Variable	Tipo de datos	Declaración	Descripción
OB20_EV_CLASS	BYTE	TEMP	Clase de evento e identificadores: B#16#11: La alarma está activa
OB20_STRT_INF	BYTE	TEMP	• B#16#21: Solicitud de inicio para el OB 20 • (B#16#22: Solicitud de inicio para el OB 21) • (B#16#23: Solicitud de inicio para el OB 22) • (B#16#24: Solicitud de inicio para el OB 23)
OB20_PRIORITY	BYTE	TEMP	Clase de prioridad parametrizada: De 3 (OB 20) a 6 (OB 23) (valores predeterminados) Valor predeterminado para CPU S7-1500: 3
OB20_OB_NUMBR	BYTE	TEMP	N.º de OB (de 20 a 23)
OB20_RESERVED_1	BYTE	TEMP	Reservado
OB20_RESERVED_2	BYTE	TEMP	Reservado
OB20_SIGN	WORD	TEMP	Identificador de usuario: Parámetro de entrada SIGN de la llamada de la instrucción "SRT_DINT (Página 6892)"
OB20_DTIME	TIME	TEMP	Retardo parametrizado en ms
OB20_DATE_TIME	DATE_AND_TIME	TEMP	Fecha y hora en las que se solicitó el OB

OBs de alarma cíclica (de OB 30 a OB 38)

Descripción

S7 ofrece hasta nueve OB de alarma cíclica (de OB 30 a OB 38). Éstos permiten ejecutar programas a intervalos equidistantes. La tabla siguiente muestra los valores predefinidos para el período y las clases de prioridad de los OB de alarma cíclica.

OB de alarma cíclica	Valor predeterminado del período	Valor predeterminado de la clase de prioridad
OB 30	5 s	7
OB 31	2 s	8
OB 32	1 s	9
OB 33	500 ms	10
OB 34	200 ms	11
OB 35	100 ms	12

OB de alarma cíclica	Valor predeterminado del período	Valor predeterminado de la clase de prioridad
OB 36	50 ms	13
OB 37	20 ms	14
OB 38	10 ms	15

Funcionamiento de los OB de alarma cíclica

Los puntos de inicio equidistantes de los OB de alarma cíclica se obtienen en cada caso a partir del ciclo de tiempo y el desfase correspondientes.

Nota

Debe procurarse que el tiempo de ejecución de todos los OB de alarma cíclica sea siempre claramente inferior a su ciclo de tiempo. Si un OB de alarma cíclica todavía no ha finalizado, pero, debido al tiempo transcurrido, el sistema operativo debe procesarlo nuevamente, se ejecuta el OB de error de tiempo (OB 80). A continuación se reactiva la alarma cíclica que haya causado el error.

Se puede bloquear la llamada del OB de alarma cíclica con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Los rangos de valores correspondientes a los parámetros de ciclo de tiempo, clase de prioridad y desfase pueden consultarse en los datos técnicos de la CPU. Las modificaciones de parámetros se realizan mediante configuración.

Datos locales de los OB de alarma cíclica

La tabla siguiente contiene las variables temporales (TEMP) de un OB de alarma cíclica.

Variable	Tipo de datos	Descripción
OB3x_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: La alarma está activa
OB3x_STRT_INF	BYTE	• B#16#30: Solicitud especial de arranque para un OB de alarma cíclica en el sistema H (tratamiento especial parametrizado el cambiar al estado del sistema "Redundante") • B#16#31: Solicitud de inicio para el OB 30 … • B#16#36: Solicitud de inicio para el OB 35 … • B#16#39: Solicitud de inicio para el OB 38 • B#16#3A: Solicitud de arranque para OB de alarma cíclica (de OB 30 a OB 38) con ciclo de alarma cíclica inferior a un milisegundo

Variable	Tipo de datos	Descripción
OB3x_PRIORITY	BYTE	Clase de prioridad parametrizada; valores predeterminados: De 7 (OB 30) a 15 (OB 38) Valores predeterminados para CPU S7-1500: 8 a 14; 16 a 17
OB3x_OB_NUMBR	BYTE	N.º de OB (de 30 a 38)
OB3x_RESERVED_1	BYTE	Reservado
OB3x_RESERVED_2	BYTE	Reservado
OB3x_PHS_OFFSET	INT	para OB 30 a 34 y 36 a 38: • si OB3x_STRT_INF=B#16#3A: Desfase en µs • En todos los demás casos: Desfase en ms
OB35_PHASE_OFFSET	WORD	para OB 35: • si OB35_STRT_INF=B#16#3A: Desfase en µs • En todos los demás casos: Desfase en ms
OB3x_RESERVED_3	INT	Reservado
OB3x_EXC_FREQ	INT	• si OB3x_STRT_INF=B#16#3A: Ciclo de tiempo en µs • En todos los demás casos: Ciclo de tiempo en ms
OB3x_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB.

OBs de alarma de proceso (de OB 40 a OB 47)

Descripción

S7 ofrece hasta ocho alarmas de proceso independientes entre sí con un OB para cada una.

Por cada configuración se debe establecer, para cada módulo de señal que deba disparar alarmas de proceso,

- qué canales deben disparar una alarma de proceso, así como las circunstancias en que deben hacerlo.
- qué OB de alarma de proceso se asigna a cada uno de los grupos de canales (de modo predeterminado, todas las alarmas de proceso son procesadas por el OB 40).

En los CP y FM debe utilizarse a estos efectos el software correspondiente para el módulo.

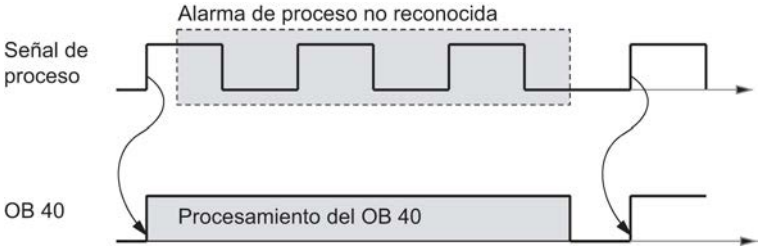
Las clases de prioridad para cada uno de los OB de alarma de proceso se definen mediante configuración.

Funcionamiento de los OB de alarma de proceso

Una vez disparada una alarma de proceso por el módulo, el sistema operativo identifica el slot y determina el OB de alarma de proceso correspondiente. Si el OB tiene una prioridad superior a la clase de prioridad actual, se procesará. La confirmación específica de canal se realiza una vez finalizado este OB de alarma de proceso.

Si en el intervalo entre la identificación y la confirmación de una alarma de proceso se produce nuevamente en el mismo módulo un evento que deba disparar una alarma de proceso, se aplica la siguiente regla:

- Si el evento se produce en el canal que ha disparado anteriormente la alarma de proceso, la alarma en cuestión se pierde. La figura siguiente muestra la relación entre una señal de proceso y el procesamiento del correspondiente OB de alarma de proceso, tomando como ejemplo un canal de un módulo de entrada digital. Supongamos que el evento desencadenante es el flanco ascendente. Se utiliza a modo de ejemplo el OB de alarma de proceso OB 40.



- Si el evento se produce en otro canal del mismo módulo, no puede dispararse por el momento ninguna alarma de proceso. Sin embargo, la alarma no se pierde, sino que se dispara una vez confirmada la alarma de proceso que se encuentre activa.

Al dispararse una alarma de proceso cuyo OB está actualmente activo debido a una alarma de proceso de otro módulo, la nueva solicitud se registra y el OB se ejecutará en su momento.

Se puede bloquear la llamada del OB de alarma de proceso con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Las alarmas de proceso de un módulo pueden parametrizarse con las instrucciones "WR_PARM (Página 6882)", " WR_DPARM (Página 6883)" y "PARM_MOD (Página 6877)".

Datos locales de los OB de alarma de proceso

La tabla siguiente contiene las variables temporales (TEMP) de un OB de alarma de proceso. Se han escogido como nombres de variable los nombres predeterminados del OB 40.

Variable	Tipo de datos	Descripción
OB40_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: La alarma está activa
OB40_STRT_INF	BYTE	• B#16#41: Alarma mediante cable de interrupción 1 • B#16#42: Alarma mediante cable de interrupción 2 (solo con S7-400) • B#16#43: Alarma mediante cable de interrupción 3 (solo con S7-400) • B#16#44: Alarma mediante cable de interrupción 4 (solo con S7-400) • B#16#45: WinAC: Alarma disparada desde el PC Nota: En modo multiprocesamiento, los cables de interrupción 1, ... 4 están asignados a las CPU 1, ... 4.
OB40_PRIORITY	BYTE	Clase de prioridad parametrizada: Valores predeterminados: De 16 (OB 40) a 23 (OB 47) Valor predeterminado para CPU S7-1500: 18

Variable	Tipo de datos	Descripción
OB40_OB_NUMBR	BYTE	N.º de OB (de 40 a 47)
OB40_RESERVED_1	BYTE	Reservado
OB40_IO_FLAG	BYTE	Módulo de entrada: B#16#54 Módulo de salida: B#16#55
OB40_MDL_ADDR	WORD	Dirección base lógica del módulo que ha disparado la alarma
OB40_POINT_ADDR	DWORD	• Con módulos digitales: Mapa de bits con las entradas del módulo que han disparado la alarma de proceso La asignación de un determinado bit OB40_POINT_ADDR a un determinado canal del módulo se puede consultar en la descripción del módulo correspondiente. • Con módulos analógicos: mapa de bits que indica qué canal ha rebasado un determinado límite (la estructura exacta puede consultarse en la descripción del módulo). • En CP o IM: Estado de alarma del módulo (no relevante para el usuario)
OB40_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Nota

Si se utiliza una CPU compatible con DPV1, puede usarse la instrucción "RALRM (Página 6745)" para obtener información acerca de la alarma, más allá de la información de arranque del OB. Esto también se aplica cuando el maestro DP se opera en el modo compatible con S7.

OB de alarma de estado (OB 55)

Descripción

El sistema operativo de la CPU activa el OB 55 cuando se ha disparado una alarma de estado desde un slot de un esclavo DPV1. Esto puede suceder cuando un componente (módulo) de un esclavo DPV1 cambia de estado operativo, p. ej. de RUN a STOP. Los eventos concretos que originan una alarma de estado pueden consultarse en la documentación del fabricante del esclavo DPV1.

Nota

Si las CPU S7-400 o los CPs PROFIBUS S7-400 están ajustados en el modo DP como "Compatible con S7", no se puede utilizar ningún OB de alarma de estado (OB 55).

Datos locales del OB de alarma de estado

La tabla siguiente contiene las variables temporales (TEMP) del OB de alarma de estado. Se han escogido como nombres de variable los nombres predeterminados del OB 55.

Variable	Tipo de datos	Descripción
OB55_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11 (evento entrante)
OB55_STRT_INF	BYTE	• B#16#55: Alarma de estado con DP • B#16#58: alarma de estado con PROFINET IO
OB55_PRIORITY	BYTE	Clase de prioridad parametrizada, valor predeterminado: 2 o 4 (en CPU S7-1500)
OB55_OB_NUMBR	BYTE	N.º de OB (55)
OB55_RESERVED_1	BYTE	Reservado
OB55_IO_FLAG	BYTE	Módulo de entrada: B#16#54 Módulo de salida: B#16#55
OB55_MDL_ADDR	WORD	Dirección base lógica del componente (módulo) que ha disparado la alarma
OB55_LEN	BYTE	Longitud del bloque lógico que emite la alarma
OB55_TYPE	BYTE	Identificador de tipo de alarma "Alarma de estado"
OB55_SLOT	BYTE	Número de slot del componente (módulo) que ha disparado la alarma
OB55_SPEC	BYTE	Especificador: • bits 0 a 1: Especificador de alarma • Bit 2: Add_Ack • Bits 3 a 7: N.º de sec.
OB55_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Nota

El significado, indicado en la tabla anterior, de OB55_LEN, OB55_TYPE, OB55_SLOT y OB55_SPEC solo es válido para alarmas de estado en DP. Si existe una alarma de estado en PROFINET IO, deben organizarse las variables locales como se indica en la tabla siguiente.

Nota

La información adicional completa de alarma procedente del telegrama DP puede consultarse llamando a instrucción "RALRM (Página 6745)" dentro del OB 55.

Si se desea programar el OB 55 en función de los eventos de arranque, se recomienda organizar las variables locales de la siguiente manera:

Variable	Tipo de datos
OB55_EV_CLASS	BYTE
OB55_STRT_INF	BYTE
OB55_PRIORITY	BYTE
OB55_OB_NUMBR	BYTE
OB55_RESERVED_1	BYTE
OB55_IO_FLAG	BYTE
OB55_MDL_ADR	WORD
OB55_Z2	WORD
OB55_Z3	WORD
OB55_DATE_TIME	DATE_AND_TIME

Dependiendo del evento de arranque, las variables OB55_Z2 y OB55_Z3 contienen distinta información. Esto se aclara a continuación.

Significado de OB55_Z2

OB55_STRT_INF	Significado de OB55_Z2
B#16#55	• low byte: Identificador de tipo de alarma "Alarma de estado" • high byte: Longitud del bloque lógico que emite la alarma
B#16#58	Identificador del tipo de alarma: • W#16#0000: reservado • W#16#0001: Alarma de diagnóstico (entrante) • W#16#0002: Alarma de proceso • W#16#0003: Alarma de extracción • W#16#0004: Alarma de inserción • W#16#0005: Alarma de estado • W#16#0006: Alarma de actualización • W#16#0007: Alarma de redundancia • W#16#0008: Controlado por el supervisor (controlled by supervisor) • W#16#0009: Habilitado • W#16#000A: Submódulo insertado erróneo • W#16#000B: Retorno del submódulo • W#16#000C: Alarma de diagnóstico (saliente) • W#16#000D: Mensaje de conexión de comunicación directa • W#16#000E: Mensaje de cambio de vecindad • W#16#000F: Mensaje de sincronización de ciclo (desde el bus) • W#16#0010: Mensaje de sincronización de ciclo (desde el dispositivo) • W#16#0011: Mensaje de componente de red • W#16#0012: Mensaje de sincronización de hora (desde el bus) • W#16#0013 a 001E: Reservado • W#16#001F: Alarma de extracción, módulo • W#16#0020 a 007F: Alarma de fabricante • W#16#0080 a FFFF: Reservado

Significado de OB55_Z3

OB55_STRT_INF	Significado de OB55_Z3
B#16#55	• low byte: Especificador – bits 0 a 1: Especificador de alarma – Bit 2: Add_Ack – Bits 3 a 7: Número de secuencia • high byte: Slot del componente (módulo) que ha disparado la alarma
B#16#58	Especificador de alarma • bits 0 a 10: Número de secuencia (rango de valores de 0 a 2047) • Bit 11: Diagnóstico de canal – 0: No existe diagnóstico de canal – 1: Existe diagnóstico de canal • Bit 12: Estado del diagnóstico específico de fabricante: – 0: No existe información de estado específica del fabricante – 1: Existe información de estado específica del fabricante • Bit 13: Estado del diagnóstico en el submódulo: – 0: No existe información de estado, se han eliminado todos los errores – 1: Existe por lo menos un diagnóstico de canal y/o información de estado • Bit 14: Reservado • Bit 15: Application Relationship Diagnosis State: – 0: ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico – 1: por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

OB de alarma de actualización (OB 56)

Descripción

El sistema operativo de la CPU activa el OB 56 cuando se ha disparado una alarma de estado desde un slot de un esclavo DPV1. Esto puede suceder cuando se han modificado parámetros en un slot de un esclavo DPV1 (mediante acceso local o remoto). Los eventos concretos que originan una alarma de actualización pueden consultarse en la documentación del fabricante del esclavo DPV1.

Nota

Si las CPU S7-400 o los CPs PROFIBUS S7-400 están ajustados en el modo DP como "Compatible con S7", no se puede utilizar ningún OB de alarma de actualización (OB 56).

Datos locales del OB de alarma de actualización

La tabla siguiente contiene las variables temporales (TEMP) del OB de alarma de actualización. Se han escogido como nombres de variable los nombres predeterminados del OB 56.

Variable	Tipo de datos	Descripción
OB56_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11 (evento entrante)
OB56_STRT_INF	BYTE	• B#16#56: Alarma de actualización con DP • B#16#59: Alarma de actualización con PROFINET IO
OB56_PRIORITY	BYTE	Clase de prioridad parametrizada, valor predeterminado: 2 o 4 (en CPU S7-1500)
OB56_OB_NUMBR	BYTE	N.º de OB (56)
OB56_RESERVED_1	BYTE	Reservado
OB56_IO_FLAG	BYTE	Módulo de entrada: B#16#54 Módulo de salida: B#16#55
OB56_MDL_ADDR	WORD	Dirección base lógica del componente (módulo) que ha disparado la alarma
OB56_LEN	BYTE	Longitud del bloque lógico que emite la alarma
OB56_TYPE	BYTE	Identificador de tipo de alarma "Alarma de actualización"
OB56_SLOT	BYTE	Número de slot del componente (módulo) que ha disparado la alarma
OB56_SPEC	BYTE	Especificador: • bits 0 a 1: Especificador de alarma • Bit 2: Add_Ack • Bits 3 a 7: N.º de sec.
OB56_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Nota

El significado, indicado en la tabla anterior, de OB56_LEN, OB56_TYPE, OB56_SLOT y OB56_SPEC solo es válido para alarmas de actualización en DP. Si existe una alarma de actualización en PROFINET IO, deben organizarse las variables locales como se indica en la tabla siguiente.

Nota

La información adicional completa de alarma procedente del telegrama DP puede consultarse llamando a instrucción "RALRM (Página 6745)" dentro del OB 56.

Si se desea programar el OB 56 en función de los eventos de arranque, se recomienda organizar las variables locales de la siguiente manera:

Variable	Tipo de datos
OB56_EV_CLASS	BYTE
OB56_STRT_INF	BYTE
OB56_PRIORITY	BYTE
OB56_OB_NUMBR	BYTE
OB56_RESERVED_1	BYTE
OB56_IO_FLAG	BYTE
OB56_MDL_ADR	WORD
OB56_Z2	WORD
OB56_Z3	WORD
OB56_DATE_TIME	DATE_AND_TIME

Dependiendo del evento de arranque, las variables OB56_Z2 y OB56_Z3 contienen distinta información. Esto se aclara a continuación.

Significado de OB56_Z2

OB56_STRT_INF	Significado de OB56_Z2
B#16#56	• low byte: Identificador de tipo de alarma "Alarma de estado" • high byte: Longitud del bloque lógico que emite la alarma
B#16#59	Identificador del tipo de alarma • W#16#0000: reservado • W#16#0001: Alarma de diagnóstico (entrante) • W#16#0002: Alarma de proceso • W#16#0003: Alarma de extracción • W#16#0004: Alarma de inserción • W#16#0005: Alarma de estado • W#16#0006: Alarma de actualización • W#16#0007: Alarma de redundancia • W#16#0008: Controlado por el supervisor (controlled by supervisor) • W#16#0009: Habilitado • W#16#000A: Submódulo insertado erróneo • W#16#000B: Retorno del submódulo • W#16#000C: Alarma de diagnóstico (saliente) • W#16#000D: Mensaje de conexión de comunicación directa • W#16#000E: Mensaje de cambio de vecindad • W#16#000F: Mensaje de sincronización de ciclo (desde el bus) • W#16#0010: Mensaje de sincronización de ciclo (desde el dispositivo) • W#16#0011: Mensaje de componente de red • W#16#0012: Mensaje de sincronización de hora (desde el bus) • W#16#0013 a 001E: Reservado • W#16#001F: Alarma de extracción, módulo • W#16#0020 a 007F: Alarma de fabricante • W#16#0080 a FFFF: Reservado

Significado de OB56_Z3

OB56_STRT_INF	Significado de OB56_Z3
B#16#56	• low byte: Especificador – bits 0 a 1: Especificador de alarma – Bit 2: Add_Ack – Bits 3 a 7: Número de secuencia • high byte: Slot del componente (módulo) que ha disparado la alarma
B#16#59	Especificador de alarma: • bits 0 a 10: Número de secuencia (rango de valores de 0 a 2047) • Bit 11: Diagnóstico de canal – 0: No existe diagnóstico de canal – 1: Existe diagnóstico de canal • Bit 12: Estado del diagnóstico específico de fabricante: – 0: No existe información de estado específica del fabricante – 1: Existe información de estado específica del fabricante • Bit 13: Estado del diagnóstico en el submódulo: – 0: No existe información de estado, se han eliminado todos los errores – 1: Existe por lo menos un diagnóstico de canal y/o información de estado • Bit 14: Reservado • Bit 15: Application Relationship Diagnosis State: – 0: ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico – 1: por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

OB de alarmas específicas del fabricante (OB 57)

Descripción

El sistema operativo de la CPU activa el OB 57 cuando se ha disparado una alarma específica del fabricante desde un slot de un esclavo DPV1.

Nota

Si las CPUs S7-400 o los CPs PROFIBUS S7-400 están ajustados en el modo DP como "Compatible con S7", no se puede utilizar ningún OB de alarma específica del fabricante (OB 57).

Datos locales del OB de alarmas específicas del fabricante

La tabla siguiente contiene las variables temporales (TEMP) del OB de alarmas específicas del fabricante. Se han escogido como nombres de variable los nombres predeterminados del OB 57.

Variable	Tipo de datos	Descripción
OB57_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11 (evento entrante)
OB57_STRT_INF	BYTE	• B#16#57: Manufacture Alarm con DP • B#16#5A: Manufacture Alarm con PROFINET IO • B#16#5B: IO: Profile Specific Alarm
OB57_PRIORITY	BYTE	Clase de prioridad parametrizada, valor predeterminado: 2 o 4 (en CPU S7-1500)
OB57_OB_NUMBR	BYTE	N.º de OB (57)
OB57_RESERVED_1	BYTE	Reservado
OB57_IO_FLAG	BYTE	Módulo de entrada: B#16#54 Módulo de salida: B#16#55
OB57_MDL_ADDR	WORD	Dirección base lógica del componente (módulo) que ha disparado la alarma
OB57_LEN	BYTE	Longitud del bloque lógico que emite la alarma
OB57_TYPE	BYTE	Identificador de tipo de alarma "Alarma específica del fabricante"
OB57_SLOT	BYTE	Número de slot del componente (módulo) que ha disparado la alarma
OB57_SPEC	BYTE	Especificador: • Bits 0 a 1: Especificador de alarma • Bit 2: Add_Ack • Bits 3 a 7: N.º de sec.
OB57_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Nota

El significado, indicado en la tabla anterior, de OB57_LEN, OB57_TYPE, OB57_SLOT y OB57_SPEC solo es válido para alarmas específicas del fabricante en DP. Si existe una alarma específica del fabricante en PROFINET IO, deben organizarse las variables locales como se indica en la tabla siguiente.

Nota

La información adicional completa de alarma procedente del telegrama DP puede consultarse llamando a instrucción "RALRM (Página 6745)" dentro del OB 57.

Si se desea programar el OB 57 en función de los eventos de arranque, se recomienda organizar las variables locales de la siguiente manera:

Variable	Tipo de datos
OB57_EV_CLASS	BYTE
OB57_STRT_INF	BYTE
OB57_PRIORITY	BYTE
OB57_OB_NUMBR	BYTE
OB57_RESERVED_1	BYTE
OB57_IO_FLAG	BYTE
OB57_MDL_ADR	WORD
OB57_Z2	WORD
OB57_Z3	WORD
OB57_DATE_TIME	DATE_AND_TIME

Dependiendo del evento de arranque, las variables OB57_Z2 y OB57_Z3 contienen distinta información. Esto se aclara a continuación.

Significado de OB57_Z2

OB57_STRT_INF	Significado de OB57_Z2
B#16#57	• low byte: Identificador de tipo de alarma "Alarma de estado" • high byte: Longitud del bloque lógico que emite la alarma
B#16#5A	Identificador del tipo de alarma: • W#16#0000: reservado • W#16#0001: Alarma de diagnóstico (entrante) • W#16#0002: Alarma de proceso • W#16#0003: Alarma de extracción • W#16#0004: Alarma de inserción • W#16#0005: Alarma de estado • W#16#0006: Alarma de actualización • W#16#0007: Alarma de redundancia • W#16#0008: Controlado por el supervisor (controlled by supervisor) • W#16#0009: Habilitado • W#16#000A: Submódulo insertado erróneo • W#16#000B: Retorno del submódulo • W#16#000C: Alarma de diagnóstico (saliente) • W#16#000D: Mensaje de conexión de comunicación directa • W#16#000E: Mensaje de cambio de vecindad • W#16#000F: Mensaje de sincronización de ciclo (desde el bus) • W#16#0010: Mensaje de sincronización de ciclo (desde el dispositivo) • W#16#0011: Mensaje de componente de red • W#16#0012: Mensaje de sincronización de hora (desde el bus) • W#16#0013 a 001E: Reservado • W#16#001F: Alarma de extracción, módulo • W#16#0020 a 007F: Alarma de fabricante • W#16#0080 a FFFF: Reservado

Significado de OB57_Z3

OB57_STRT_INF	Significado de OB57_Z3
B#16#57	• low byte: Especificador – Bits 0 a 1: Especificador de alarma – Bit 2: Add_Ack – Bits 3 a 7: Número de secuencia • high byte: Slot del componente (módulo) que ha disparado la alarma
B#16#5A	Especificador de alarma • Bits 0 a 10: Número de secuencia (rango de valores de 0 a 2047) • Bit 11: Diagnóstico de canal – 0: No existe diagnóstico de canal – 1: Existe diagnóstico de canal • Bit 12: Estado del diagnóstico específico de fabricante: – 0: No existe información de estado específica del fabricante – 1: Existe información de estado específica del fabricante • Bit 13: Estado del diagnóstico en el submódulo: – 0: No existe información de estado, se han eliminado todos los errores – 1: Existe por lo menos un diagnóstico de canal y/o información de estado • Bit 14: Reservado • Bit 15: Application Relationship Diagnosis State: – 0: ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico – 1: por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

OBs de alarma de sincronismo (de OB 61 a OB 64)

Descripción

Las alarmas de sincronismo permiten ejecutar programas de modo isócrono con respecto al ciclo DP o al tiempo de ciclo de emisión de PN. A la alarma de sincronismo TSAL y le corresponde el OB 6y, 1 <= y <= 4, como OB de interfaz. La prioridad de los OBs 61 a 64 puede ajustarse entre 0 (OB desactivado) y de 2 a 26.

⚠ PRECAUCIÓN

Accesos directos

Al acceder directamente con comandos L o T (p. ej. L PIB, T PQB), o si se utilizan las instrucciones "DPRD_DAT (Página 6776)" o "DPWR_DAT (Página 6778)", no debe accederse a áreas de la periferia que tengan asignadas memorias imagen parcial de proceso vinculadas a OB6x (alarmas de sincronismo).

Datos locales del OB de alarma de sincronismo

La tabla siguiente contiene las variables temporales (TEMP) del OB de alarma de sincronismo. Se han escogido como nombres de variable los nombres predeterminados del OB 61.

Variable	Tipo de datos	Descripción
OB61_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: La alarma está activa
OB61_STRT_INF	BYTE	B#16#64: Solicitud de arranque para el OB 61 : B#16#67: Solicitud de arranque para el OB 64
OB61_PRIORITY	BYTE	Clase de prioridad parametrizada; valor predeterminado: 25 o 21 (en CPU S7-1500)
OB61_OB_NUMBR	BYTE	N.º de OB: 61 ... 64
OB61_RESERVED_1	BYTE	Reservado
OB61_RESERVED_2	BYTE	Reservado
OB61_GC_VIOL	BOOL	Violación de GC en PROFIBUS DP
OB61_FIRST	BOOL	Primera ejecución después de arranque o estado de parada
OB61_MISSED_EXEC	BYTE	Número de arranques fallidos del OB 61 desde la última ejecución del OB 61
OB61_DP_ID	BYTE	ID de sistema maestro DP del sistema maestro DP isócrono (de 1 a 32) o ID de sistema PROFINET IO del sistema PNIO isócrono (de 100 a 115)
OB61_RESERVED_3	BYTE	Reservado
OB61_RESERVED_4	WORD	Reservado
OB61_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

OB de error de tiempo (OB 80)

Descripción

El sistema operativo de la CPU llama el OB 80 en caso de que durante el procesamiento se produzca uno de los siguientes errores:

- Rebase por exceso del tiempo de ciclo
- Error de acuse al procesar un OB
- Adelanto de la hora (salto de hora) al arrancar un OB

Si por ejemplo se produce un evento de arranque para un OB de alarma cíclica antes de que finalice el procesamiento anterior de ese mismo OB, el sistema operativo llama el OB 80.

Si el OB 80 no está programado, la CPU pasa al estado operativo STOP.

Se puede bloquear la llamada del OB de error de tiempo con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Nota

Si se llama al OB 80 dos veces en el mismo ciclo debido al rebase por exceso del tiempo de ciclo, la CPU pasa al estado STOP. Esto puede evitarse llamando la instrucción "RE_TRIGR" en el punto adecuado.

Nota sobre el uso de "Report System Errors"

"Report System Errors" retarda el procesamiento de las alarmas generadas durante la ejecución.

Si se cumple la condición **Tiempo de ejecución de "Report System Errors" > 50 % del tiempo de ciclo**, se refuerza el efecto (se considera que el tiempo de ejecución de "Report System Errors" es la suma de todos los tiempos de ejecución de "Report System Errors" dentro del tiempo de ciclo.). Puede conllevar la llamada del OB de error de tiempo o el cambio de la CPU a STOP (si no está cargado el OB de error de tiempo).

Para evitarlo existen las posibilidades siguientes:

- Si se utilizan los OBs de alarma de sincronismo (OB61 a OB64), hay que cerciorarse de que son llamados con menos frecuencia que antes dentro de cada unidad de tiempo. Puede lograrse incrementando en cada OB de alarma de sincronismo el parámetro "Factor de ciclo de aplicación" correspondiente.
- Incremente la relación (tiempo de ciclo - tiempo de ejecución de "Report System Errors")/(tiempo de ejecución de "Report System Errors"). Para ello, existen las dos posibilidades siguientes:
  - Si en los OBs de alarma cíclica (OB30 a OB38) se llama "Report System Errors", se aumentan los tiempos de ciclo de los OBs correspondientes.
  - Incremente en el OB1 el tiempo de ejecución de la sección de programa independiente de "Report System Errors". Hágalo, p. ej. utilizando la instrucción "WAIT".

Datos locales del OB de error de tiempo

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de tiempo. Se han escogido como nombres de variable los nombres predeterminados del OB 80.

Variable	Tipo de datos	Descripción
OB80_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#35
OB80_FLT_ID	BYTE	Código de error (valores posibles: B#16#01, B#16#02, B#16#05, B#16#06, B#16#07, B#16#08, B#16#09, B#16#0A, B#16#0B)

Variable	Tipo de datos	Descripción
OB80_PRIORITY	BYTE	Clase de prioridad: El OB 80 se ejecuta en RUN en la clase de prioridad 26, y en caso de desbordamiento del búfer de solicitud de OB, en la clase de prioridad 28 En las CPU S7-1500: Prioridad 22
OB80_OB_NUMBR	BYTE	N.º de OB (80)
OB80_RESERVED_1	BYTE	Reservado
OB80_RESERVED_2	BYTE	Reservado
OB80_ERROR_INFO	WORD	Información de error: En función del código de error
OB80_ERR_EV_CLASS	BYTE	Clase a la que pertenece el evento que ha provocado el error
OB80_ERR_EV_NUM	BYTE	Número del evento que ha provocado el error
OB80_OB_PRIORITY	BYTE	Información de error: En función del código de error
OB80_OB_NUM	BYTE	Información de error: En función del código de error
OB80_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Significado de las variables dependientes del código de error:

Código de error	Variable	Bit	Significado
B#16#01			Rebase por exceso del tiempo de ciclo
	OB80_ERROR_INFO		Tiempo de ejecución del último ciclo (ms)
	OB80_ERR_EV_CLASS		Clase a la que pertenece el evento que ha provocado la alarma
	OB80_ERR_EV_NUM		Número del evento que ha provocado la alarma
	OB80_OB_PRIORITY		Clase de prioridad del OB que se estaba procesando cuando se produjo el error
	OB80_OB_NUM		Número del OB que se estaba procesando cuando se produjo el error
B#16#02			El OB solicitado todavía está procesándose.
	OB80_ERROR_INFO		La correspondiente variable temporal del OB solicitado. Este se determina mediante OB80_ERR_EV_CLASS y OB80_ERR_EV_NUM.
	OB80_ERR_EV_CLASS		Clase a la que pertenece el evento que ha provocado la alarma
	OB80_ERR_EV_NUM		Número del evento que ha provocado la alarma
	OB80_OB_PRIORITY		Clase de prioridad del OB que provoca errores (p. ej.: "7" para el OB 30/clase de prioridad 7, que se debía ejecutar pero no se ha podido ejecutar)
	OB80_OB_NUM		Número del OB que provoca errores (p. ej.: "30" para el OB 30, que se debía ejecutar pero no se ha podido ejecutar)
B#16#05			Alarma horaria expirada debido a salto de hora
	OB80_ERROR_INFO	Bit 0 activado	El momento de inicio de la alarma 0 se encuentra en el pasado

Código de error	Variable	Bit	Significado
		...	
		Bit 7 activado	El momento de inicio de la alarma 7 se encuentra en el pasado
		Bit 8 a 15	Sin utilizar
	OB80_ERR_EV_CLASS		Sin utilizar
	OB80_ERR_EV_NUM		Sin utilizar
	OB80_OB_PRIORITY		Sin utilizar
	OB80_OB_NUM		Sin utilizar
B#16#06	Como con el código de error B#16#05		Alarma horaria expirada al reanudar en RUN después de PARADA
B#16#07	Como con el código de error B#16#02		Desbordamiento del búfer de solicitud de OB para la clase de prioridad actual (cada solicitud de inicio de OB para una clase de prioridad se registra en el correspondiente búfer de solicitud de OB; una vez finalizado el OB, la entrada se borra. Si para una determinada clase de prioridad existe un número de solicitudes de OB superior al máximo posible de entradas del correspondiente búfer de solicitud de OB, se llama el OB 80 con el código de error B#16#07.)
B#16#08	Como con el código de error B#16#02		Error de tiempo en alarma de sincronismo
B#16#09	Como con el código de error B#16#02		Pérdida de alarma por sobrecarga de alarmas
B#16#0A			Reanudación en RUN después de CiR
	OB80_ERROR_INFO		Tiempo de sincronización CiR en ms
B#16#0B			Error de tiempo en alarma de sincronismo tecnológica
	OB80_ERR_EV_NUM		Número del evento que ha provocado la alarma: W#16#116A
	OB80_OB_PRIORITY		Clase de prioridad del OB que se estaba procesando cuando se produjo el error
	OB80_OB_NUM		Número del OB que se estaba procesando cuando se produjo el error: 65

OB de error de fuente de alimentación (OB 81)

Descripción

El sistema operativo de la CPU llama el OB 81 cuando se produce un evento provocado por un error en la fuente de alimentación (solo para S7-400) en el respaldo (tanto con eventos entrantes como salientes).

En S7-400, se llama el OB 81 en caso de error de batería solo cuando está activada la comprobación de batería con la salida BATT.INDIC.

Si el OB 81 no está programado, la CPU no cambia al estado operativo STOP.

Se puede bloquear la llamada del OB de error de fuente de alimentación con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales del OB de error de fuente de alimentación

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de fuente de alimentación. Se han escogido como nombres de variable los nombres predeterminados del OB 81.

Variable	Tipo de datos	Descripción
OB81_EV_CLASS	BYTE	Clase de evento e identificadores: • B#16#38: Evento saliente • B#16#39: Evento entrante
OB81_FLT_ID	BYTE	Código de error (valores posibles: B#16#21, B#16#22, B#16#23, B#16#25, B#16#26, B#16#27, B#16#31, B#16#32, B#16#33)
OB81_PRIORITY	BYTE	Clase de prioridad: Ajustable mediante configuración P. ej., para el estado operativo RUN pueden ajustarse los valores del 2 al 26.
OB81_OB_NUMBR	BYTE	N.º de OB (81)
OB81_RESERVED_1	BYTE	Reservado
OB81_RESERVED_2	BYTE	Reservado
OB81_RACK_CPU	WORD	• Del bit 0 al bit 7: B#16#00 • Del bit 8 al bit 15: – Con una CPU estándar: B#16#00 – Con una CPU H: Del bit 8 al bit 10: N.º de rack, bit 11: 0=CPU de reserva, 1=CPU maestra, bits del 12 al 15: 1111
OB81_RESERVED_3	BYTE	Relevante solo para los códigos de error B#16#31, B#16#32 y B#16#33
OB81_RESERVED_4	BYTE	Relevante solo para los códigos de error B#16#31, B#16#32 y B#16#33
OB81_RESERVED_5	BYTE	Relevante solo para los códigos de error B#16#31, B#16#32 y B#16#33
OB81_RESERVED_6	BYTE	Relevante solo para los códigos de error B#16#31, B#16#32 y B#16#33
OB81_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Las variables OB81_RESERVED_i, 3 ≤ i ≤ 6, contienen los aparatos de ampliación en los que el respaldo de batería (con código de error B#16#31), la tensión de respaldo (con código de error B#16#32) o la alimentación de 24 V (con código de error B#16#33) han fallado o se han recuperado. La tabla siguiente muestra qué bit de las variables OB81_RESERVED_i, 3 ≤ i ≤ 6, pertenece a cada aparato de ampliación.

	OB81_RESERVED_6	OB81_RESERVED_5	OB81_RESERVED_4	OB81_RESERVED_3
Bit 0	Reservado	8. Ap. de ampl.	16. Ap. de ampl.	Reservado
Bit 1	1. Ap. de ampl.	9. Ap. de ampl.	17. Ap. de ampl.	Reservado.
Bit 2	2. Ap. de ampl.	10. Ap. de ampl.	18. Ap. de ampl.	Reservado
Bit 3	3. Ap. de ampl.	11. Ap. de ampl.	19. Ap. de ampl.	Reservado
Bit 4	4. Ap. de ampl.	12. Ap. de ampl.	20. Ap. de ampl.	Reservado
Bit 5	5. Ap. de ampl.	13. Ap. de ampl.	21. Ap. de ampl.	Reservado
Bit 6	6. Ap. de ampl.	14. Ap. de ampl.	Reservado	Reservado
Bit 7	7. Ap. de ampl.	15. Ap. de ampl.	Reservado	Reservado

Significado de los bits de las variables OB81_RESERVED_i (para el aparato de ampliación implicado):

En el evento entrante se marcan los aparatos de ampliación (están activados los bits correspondientes) para los que ha fallado por lo menos una batería, la tensión de respaldo o la alimentación de 24 V. No se muestran los aparatos de ampliación para los que ya ha fallado antes por lo menos una batería, la tensión de respaldo o la alimentación de 24 V. En el evento saliente se comunica la recuperación del respaldo en por lo menos un aparato de ampliación (están activados los bits correspondientes).

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB 81.

OB81_EV_CLASS	OB81_FLT_ID	Significado
B#16#39/B#16#38	B#16#21	Hay como mínimo una pila tampón vacía en el aparato central/eliminado (BATTF). Nota: El evento entrante se produce únicamente en caso de fallo de una de las dos baterías (si se usan pilas tampón redundantes). Si posteriormente falla también la otra batería, el evento no vuelve a producirse.
B#16#39/B#16#38	B#16#22	Falta la tensión de respaldo en el aparato central/ eliminado (BAF).
B#16#39/B#16#38	B#16#23	Fallo de la alimentación de 24 V en el aparato central/ eliminado.
B#16#39/B#16#38	B#16#25	Hay como mínimo una pila tampón vacía en por lo menos un aparato central redundante/eliminado (BATTF).
B#16#39/B#16#38	B#16#26	Falta la tensión de respaldo en por lo menos un aparato central redundante/eliminado (BAF).
B#16#39/B#16#38	B#16#27	Fallo de la alimentación de 24 V en por lo menos un aparato central redundante/eliminado
B#16#39/B#16#38	B#16#31	Hay como mínimo una pila tampón vacía en por lo menos un aparato de ampliación/eliminado (BATTF).
B#16#39/B#16#38	B#16#32	Falta la tensión de respaldo en por lo menos un aparato de ampliación/eliminado (BAF).
B#16#39/B#16#38	B#16#33	Fallo de la alimentación de 24 V en al menos un aparato de ampliación/ eliminado.

Programa de ejemplo para el OB 81

Usando un programa de ejemplo en AWL, se muestra la manera de leer el código de error en el OB 81.

El programa tiene la siguiente estructura:

El código de error en el OB 81 (OB81_FLT_ID) se lee y se compara con la identificación de los eventos "Hay como mínimo una pila tampón vacía en el aparato central" (B#16#21) y "Falta la tensión de respaldo en el aparato central" (B#16#22).

Si el código de error concuerda con uno de los eventos mencionados, el programa salta a la marca BF. De lo contrario se finaliza el bloque.

A partir de la etiqueta BF, el programa activa la variable *Error de batería* cuando existe un evento entrante. Cuando existe un evento saliente, el programa desactiva la variable.

```
L    B#16#21                                  //Identificador del evento "Hay como mínimo una
                                              pila tampón vacía en el aparato central"
L    #OB81_FLT_ID                             //Código de error OB 81
==I                                           //Si es igual,
SPB    BF                                     //saltar a BF
L    B#16#22                                  //Identificador del evento "Falta la tensión de
                                              respaldo en el aparato central"
==I                                           //Si es igual que el código de error OB 81,
SPB    BF                                     //saltar a BF
BEA                                           //no hay mensaje sobre error de batería

BF:    L    B#16#39                           //Identificador de evento entrante
L    #OB81_EV_CLASS                           //Clase de evento e identificadores al llamar el
                                              OB 81
==I                                           //Si es igual,
S    Error de batería                         //entonces activar error de batería (variable de
                                              la tabla de variables)
L    B#16#38                                  //Identificador de evento saliente
==I                                           //Si es igual que clase de evento e
                                              identificadores al llamar el OB 81,
R    Error de batería                         //entonces desactivar error de batería (variable
                                              de la tabla de variables)
```

OB de alarma de diagnóstico (OB 82)

Descripción

Cuando un módulo diagnosticable en el que se ha habilitado la alarma de diagnóstico detecta una modificación de su estado de diagnóstico, envía a la CPU una solicitud de alarma de diagnóstico:

- Hay un fallo, un componente requiere mantenimiento, o ambas cosas (evento entrante).
- Ya no hay ningún fallo, y ningún componente requiere mantenimiento (evento saliente).

Tras ello, el sistema operativo llama el OB 82.

El OB 82 contiene en sus variables locales la dirección base lógica, así como una información de diagnóstico de cuatro bytes de longitud acerca del módulo defectuoso (ver tabla siguiente).

Si no se ha programado el OB 82, la CPU pasa al estado operativo STOP.

Se puede bloquear la llamada del OB de alarma de diagnóstico con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Nota

Con controladores PROFINET IO, es posible especificar en la configuración si las alarmas de comunicación deben provocar o no la llamada del OB de alarma de diagnóstico. Estas alarmas son eventos de diagnóstico de la interfaz PROFINET.

La opción predeterminada es que estos eventos no provoquen la llamada del OB 82.

Datos locales del OB de alarma de diagnóstico

La tabla siguiente contiene las variables temporales (TEMP) del OB de alarma de diagnóstico. Se han escogido como nombres de variable los nombres predeterminados del OB 82.

Variable	Tipo de datos	Descripción
OB82_EV_CLASS	BYTE	Clase de evento e identificadores: • B#16#38: evento saliente: módulo sin errores (todos los errores solucionados). • B#16#39: evento entrante: el módulo tendrá errores o permanece con errores (con cambio del número de errores pendientes)
OB82_FLT_ID	BYTE	Código de error (B#16#42)
OB82_PRIORITY	BYTE	• Clase de prioridad; ajustable mediante configuración • En las CPU S7-1500: prioridad, valor predeterminado: 5
OB82_OB_NUMBR	BYTE	N.º de OB (82)
OB82_RESERVED_1	BYTE	Reservado
OB82_IO_FLAG	BYTE	• En el S7-300/S7-400: – Módulo de entrada: B#16#54 – Módulo de salida: B#16#55 • En el S7-1500: Cero
OB82_MDL_ADDR	WORD	• En el S7-300/S7-400: Dirección base lógica del módulo en el que se ha producido el error • En el S7-1500: identificador de hardware del objeto de hardware que ha disparado la alarma de diagnóstico
OB82_MDL_DEFECT	BOOL	Fallo de módulo
OB82_INT_FAULT	BOOL	Error interno
OB82_EXT_FAULT	BOOL	Error externo
OB82_PNT_INFO	BOOL	Existe error de canal
OB82_EXT_VOLTAGE	BOOL	Falta tensión auxiliar externa

Variable	Tipo de datos	Descripción
OB82_FLD_CONNCTR	BOOL	Falta conector frontal
OB82_NO_CONFIG	BOOL	Módulo no parametrizado
OB82_CONFIG_ERR	BOOL	Parámetro erróneo en módulo
OB82_MDL_TYPE	BYTE	• Bits 0 a 3: Clase de módulo • Bit 4: Existe información de canal • Bit 5: Existe información de usuario • Bit 6: Alarma de diagnóstico de sustituto • Bit 7: Mantenimiento necesario
OB82_SUB_MDL_ERR	BOOL	Módulo de usuario erróneo o no existe
OB82_COMM_FAULT	BOOL	Fallo de comunicación
OB82_MDL_STOP	BOOL	Estado operativo (0: RUN, 1: STOP)
OB82_WTCH_DOG_FLT	BOOL	La vigilancia de tiempo ha reaccionado
OB82_INT_PS_FLT	BOOL	Fallo de la tensión de alimentación interna del módulo
OB82_PRIM_BATT_FLT	BOOL	Batería vacía
OB82_BCKUP_BATT_FLT	BOOL	Fallo de todo el respaldo
OB82_RESERVED_2	BOOL	Mantenimiento solicitado
OB82_RACK_FLT	BOOL	Fallo de aparato de ampliación
OB82_PROC_FLT	BOOL	Fallo de procesador
OB82_EPROM_FLT	BOOL	Error de EPROM
OB82_RAM_FLT	BOOL	Error de RAM
OB82_ADU_FLT	BOOL	Error de ADU/DAU
OB82_FUSE_FLT	BOOL	Fallo de fusible
OB82_HW_INTR_FLT	BOOL	Alarma de proceso perdida
OB82_RESERVED_3	BOOL	Reservado
OB82_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Nota

Si se utiliza una CPU compatible con DPV1, puede usarse la instrucción "RALRM (Página 6745)" para obtener información acerca de la alarma, más allá de la información de arranque del OB. Esto también se aplica cuando el maestro DP se opera en el modo compatible con S7.

Programación del OB 82

El OB 82 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 82 y cárguelo en la CPU como parte del programa de usuario.

El OB 82 puede usarse por ejemplo de la siguiente manera:

- Se evalúa la información de arranque del OB 82.
- Se realiza un diagnóstico preciso del error producido.

Cuando se dispara una alarma de diagnóstico, el módulo erróneo registra automáticamente 4 bytes de datos de diagnóstico y su dirección inicial en la información de arranque del OB de alarma de diagnóstico y en el búfer de diagnóstico. De este modo puede conocerse en qué módulo y en qué momento se ha producido un error.

Con un programa al efecto pueden evaluarse en el OB 82 otros datos de diagnóstico del módulo erróneo (en qué canal se ha producido el error o de qué error se trata). Con la instrucción "RDSYSST (Página 6963)" pueden leerse datos de diagnóstico de módulo y registrar esa información en el búfer de diagnóstico con la instrucción "WR_USMSG (Página 6947)". Además es posible enviar un mensaje de diagnóstico personalizado a un dispositivo de observación convenientemente registrado.

OB de cambio de módulo (OB 83)

Descripción

El sistema operativo de la CPU llama el OB 83 tras extraer o introducir un módulo configurado.

Si no se ha programado el OB 83, la CPU pasa al estado operativo STOP.

Se puede bloquear la llamada del OB de cambio de módulo con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Extracción e inserción de módulos

Toda extracción e inserción de un módulo configurado (no permitido: fuentes de alimentación, CPUs, cápsulas de adaptación e IM) en los estados operativos RUN, STOP y ARRANQUE provoca una alarma de cambio de módulo. A su vez, esta alarma genera en la correspondiente CPU una entrada en el búfer de diagnóstico y en la lista de estado del sistema. Además de esto, en el estado operativo RUN se inicia el OB de cambio de módulo. Si no se ha programado este OB, la CPU pasa al estado operativo STOP.

El cambio de los módulos S7-400 se comprueba dentro del sistema una vez por segundo. Para que la CPU detecte el cambio, debe haber un intervalo mínimo de dos segundos entre la extracción y la inserción del módulo S7-400. Para otros módulos, este tiempo mínimo es mayor.

Al extraer un módulo configurado en el estado operativo RUN, se arranca el OB 83. Antes de ello puede detectarse un error de acceso en un acceso directo o en la actualización de la memoria imagen de proceso.

Al insertarse un módulo en un slot configurado en el estado RUN, el sistema operativo comprueba si el tipo del módulo insertado concuerda con el definido en la configuración. A continuación se arranca el OB 83, y si los tipos de módulo coinciden, se realiza la parametrización.

Particularidades con S7-300, ET 200S y ET 200Pro

- En S7-300 no se permite la extracción e inserción de la periferia centralizada.
- En las CPUs S7-300, la alarma de cambio de módulo solo existe para las CPUs 31x PN/DP y, concretamente, solo para los componentes PROFINET IO.
- En ET 200Pro con p. ej. CPU IM 154-4-8, la alarma de cambio de módulo solo existe para la periferia centralizada.
- En ET 200S con p. ej. CPU IM151-8 PN/DP, la alarma de cambio de módulo solo existe para la periferia centralizada.

Datos locales del OB de cambio de módulo

La tabla siguiente contiene las variables temporales (TEMP) del OB de cambio de módulo. Se han escogido como nombres de variable los nombres predeterminados del OB 83.

Variable	Tipo de datos	Descripción
OB83_EV_CLASS	BYTE	Clase de evento e identificadores: • B#16#32: Fin de reparametrización del módulo • B#16#33: Inicio de reparametrización del módulo • B#16#38: Módulo insertado • B#16#39: Módulo extraído, no accesible o fin de reparametrización
OB83_FLT_ID	BYTE	Código de error (valores posibles: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58, B#16#61, B#16#63, B#16#64, B#16#65, B#16#66, B#16#67, B#16#68)
OB83_PRIORITY	BYTE	Clase de prioridad; ajustable mediante configuración En CPU S7-1500: prioridad, valor predeterminado: 6
OB83_OB_NUMBR	BYTE	N.º de OB (83)
OB83_RESERVED_1	BYTE	Identificador de módulo o submódulo/módulo de interfaz
OB83_MDL_ID	BYTE	Rango: • B#16#54: Área de periferia de las entradas (PI) • B#16#55: Área de periferia de las salidas (PO)
OB83_MDL_ADDR	WORD	• PROFIBUS DP centralizado o descentralizado: Dirección base lógica del módulo afectado; en caso de módulo mixto, la dirección lógica menor utilizada del módulo. Si las direcciones lógicas I y O del módulo mixto son iguales, la dirección base lógica recibirá el identificador I. • PROFINET IO descentralizado: Dirección base lógica del módulo/submódulo

Variable	Tipo de datos	Descripción
OB83_RACK_NUM	INT	• Si OB83_RESERVED_1 =B#16#A0: N.º del submódulo/módulo de interfaz (low byte) • Si OB83_RESERVED_1 = B#16#C4: – Centralizado: N.º del rack – PROFIBUS DP descentralizado: N.º de la estación DP (low byte) e ID del sistema maestro DP (high byte) – PROFINET IO descentralizado: Dirección física: Bit de identificación (bit 15, 1 = PROFINET IO), ID del sistema IO (bits 11 a 14) y número de estación (bits 0 a 10)
OB83_MDL_TYPE	WORD	• PROFIBUS DP centralizado o descentralizado: tipo del módulo afectado (X: no relevante para el usuario). Los tipos de módulos que no aparecen listados aquí, están documentados en el manual del módulo correspondiente. – W#16#X5XX: Módulo analógico – W#16#X8XX: Módulo de función – W#16#XCXX: CP – W#16#XFXX: Módulo digital – W#16#8340: Identificador de tipo de sustitución para módulo de entrada – W#16#9340: Identificador de tipo de sustitución para módulo de salida – W#16#A340: Identificador de tipo de sustitución para módulo mixto (I/O) – W#16#F340: Identificador de tipo de sustitución para módulo vacío o no identificable de modo inequívoco (p. ej. direcciones comprimidas) – Los tipos de módulos que no aparecen listados aquí, están documentados en el manual del módulo correspondiente. • PROFINET IO descentralizado: – W#16#8101: El tipo del módulo insertado coincide con el tipo del módulo extraído – W#16#8102: El tipo del módulo insertado no coincide con el tipo del módulo extraído
OB83_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB 83.

OB83_EV_CLASS	OB83_FLT_ID	Significado
B#16#39	B#16#51	Módulo PROFINET IO extraído
B#16#39	B#16#54	Submódulo PROFINET IO extraído
B#16#38	B#16#54	Submódulo PROFINET IO insertado y coincidente con el submódulo parametrizado
B#16#38	B#16#55	Submódulo PROFINET IO insertado, pero no coincidente con el submódulo parametrizado

OB83_EV_CLASS	OB83_FLT_ID	Significado
B#16#38	B#16#56	Submódulo PROFINET IO insertado, pero error en parametrización del módulo
B#16#38	B#16#57	Submódulo o módulo PROFINET IO insertado, pero con fallo o mantenimiento
B#16#38	B#16#58	Submódulo PROFINET IO, error de acceso eliminado
B#16#39	B#16#61	Módulo extraído o no accesible OB83_MDL_TYPE: Tipo de módulo real
B#16#38	B#16#61	Módulo insertado, tipo de módulo correcto OB83_MDL_TYPE: Tipo de módulo real
B#16#38	B#16#63	Módulo insertado, pero tipo de módulo incorrecto OB83_MDL_TYPE: Tipo de módulo real
B#16#38	B#16#64	Módulo insertado, pero defectuoso (identificador de módulo ilegible) OB83_MDL_TYPE: Tipo de módulo teórico
B#16#38	B#16#65	Módulo insertado, pero error en parametrización del módulo OB83_MDL_TYPE: Tipo de módulo real
B#16#39	B#16#66	Módulo no accesible, error de tensión de carga
B#16#38	B#16#66	Módulo de nuevo accesible, error de tensión de carga eliminado
B#16#33	B#16#67	Inicio reparametrización de un módulo
B#16#32	B#16#67	Fin reparametrización de un módulo
B#16#39	B#16#68	Reparametrización de un módulo finalizada con errores

Nota

Si se utiliza una CPU compatible con DPV1 o PROFINET, puede usarse la instrucción "RALRM (Página 6745)" para obtener información acerca de la alarma, más allá de la información de arranque del OB. Esto también se aplica cuando el maestro DP se opera en el modo compatible con S7.

Programación del OB 83

El OB 83 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 83 y cárguelo en la CPU como parte del programa de usuario.

El OB 83 puede usarse por ejemplo de la siguiente manera:

- Se evalúa la información de arranque del OB 83.
- Para parametrizar el módulo que se acaba de insertar pueden usarse las instrucciones "WR_PARM (Página 6882)", "WR_DPARM (Página 6883)", "PARM_MOD (Página 6877)", "WR_REC (Página 6774)" y "RD_REC (Página 6769)".

OB de error de hardware en CPU (OB 84)

Descripción

El sistema operativo de la CPU llama el OB 84 en los siguientes casos:

- Después de detectarse y eliminarse errores de memoria
- Con WinAC RTX: Error en el sistema operativo del PC, p. ej. Blue Screen

Si no se ha programado el OB 84, la CPU no pasa al estado operativo STOP.

Se puede bloquear la llamada del OB de error de hardware en CPU con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales del OB de error de hardware en CPU

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de hardware en CPU. Se han escogido como nombres de variable los nombres predeterminados del OB 84.

Variable	Tipo de datos	Descripción
OB84_EV_CLASS	BYTE	Clase de evento e identificadores: • B#16#38: Evento saliente • B#16#35, B#16#39: Evento entrante
OB84_FLT_ID	BYTE	Código de error (B#16#82, B#16#83, B#16#85, B#16#86, B#16#87)
OB84_PRIORITY	BYTE	Clase de prioridad; ajustable mediante configuración (configuración de hardware)
OB84_OB_NUMBR	BYTE	N.º de OB (84)
OB84_RESERVED_1	BYTE	Reservado
OB84_RESERVED_2	BYTE	Reservado
OB84_RESERVED_3	WORD	Reservado
OB84_RESERVED_4	DWORD	Reservado
OB84_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB 84.

OB84_EV_CLASS	OB84_FLT_ID	Evento de arranque del OB 84
B#16#35	B#16#82	Error de memoria detectado por el sistema operativo y eliminado
B#16#35	B#16#83	Acumulación de errores de memoria detectados y corregidos
B#16#35	B#16#85	Error en el sistema operativo del PC
B#16#39	B#16#86	Rendimiento de un acoplamiento H-Sync, disminuido
B#16#35	B#16#87	Error de memoria multibit, detectado y corregido

Programación del OB 84

El OB 84 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 84 y cárguelo en la CPU como parte del programa de usuario.

El OB 84 puede usarse por ejemplo de la siguiente manera:

- Se evalúa la información de arranque del OB 84.
- Para enviar un mensaje al búfer de diagnóstico, utilice la instrucción "WR_USMSG (Página 6947)".

OB de error de ejecución del programa (OB 85)

Descripción

El sistema operativo de la CPU llama el OB 85 en caso de que se produzca uno de los siguientes eventos:

- Evento de arranque para un OB no cargado (excepto OB 80, OB 81, OB 82, OB 83 y OB 86)
- Error al acceder a un bloque a través del sistema operativo
- Error de acceso a la periferia durante la actualización de la memoria imagen de proceso desde el sistema, en caso de que no se haya suprimido la llamada del OB 85 mediante configuración.

Nota

Si el OB 85 no está programado, la CPU pasa al estado operativo STOP cuando se produce uno de los eventos mencionados.

Se puede bloquear la llamada del OB de error de ejecución del programa con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales del OB de error de ejecución del programa

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de ejecución del programa. Se han escogido como nombres de variable los nombres predeterminados del OB 85.

Variable	Tipo de datos	Descripción
OB85_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#35, B#16#38 (solo con código de error B#16#B3 y B#16#B4), B#16#39 (solo con código de error B#16#B1, B#16#B2, B#16#B3 y B#16#B4)
OB85_FLT_ID	BYTE	Código de error (valores posibles: B#16#A1, B#16#A2, B#16#A3, B#16#A4, B#16#B1, B#16#B2, B#16#B3, B#16#B4)
OB85_PRIORITY	BYTE	Clase de prioridad; ajustable mediante configuración

Variable	Tipo de datos	Descripción
OB85_OB_NUMBR	BYTE	N.º de OB (85)
OB85_RESERVED_1	BYTE	Reservado
OB85_RESERVED_2	BYTE	Reservado
OB85_RESERVED_3	INT	Reservado
OB85_ERR_EV_CLASS	BYTE	Clase a la que pertenece el evento que ha provocado el error
OB85_ERR_EV_NUM	BYTE	Número del evento que ha provocado el error
OB85_OB_PRIOR	BYTE	Clase de prioridad del OB que se estaba procesando cuando se produjo el error (solo con algunos códigos de error, para más información ver abajo)
OB85_OB_NUM	BYTE	Número del OB que se estaba procesando cuando se produjo el error (solo con algunos códigos de error, para más información ver abajo)
OB85_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Si se desea programar el OB 85 en función de los códigos de error posibles, se recomienda organizar las variables locales de la siguiente manera:

Variable	Tipo de datos
OB85_EV_CLASS	BYTE
OB85_FLT_ID	BYTE
OB85_PRIORITY	BYTE
OB85_OB_NUMBR	BYTE
OB85_DKZ23	BYTE
OB85_RESERVED_2	BYTE
OB85_Z1	WORD
OB85_Z23	DWORD
OB85_DATE_TIME	DATE_AND_TIME

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB 85 y qué valores contienen las variables dependientes del código de error.

OB85_EV_CLASS	OB85_FLT_ID	Significado
B#16#35	B#16#A1	El programa o el sistema operativo (en función de la configuración) genera un evento de arranque para un OB que no está cargado en la CPU. • OB85_Z1: La correspondiente variable temporal del OB solicitado. Este se determina mediante el OB85_Z23. • OB85_Z23: – high word: Clase y número del evento causante – low word, high byte: Nivel de programa activo en el momento del error low word, low byte: OB activo
B#16#35	B#16#A2	El programa o el sistema operativo (en función de la configuración) genera un evento de arranque para un OB que no está cargado en la CPU. OB85_Z1 y OB85_Z23 como con OB85_FLT_ID=B#16#A1
B#16#35	B#16#A3	Error al acceder a un bloque a través del sistema operativo • OB85_Z1: Identificador de error del sistema operativo – high byte: 1=función integrada, 2=temporizador CEI – low byte: 0=sin resolución de error, 1=bloque no cargado, 2=error de longitud de área, 3=error de protección contra escritura • OB85_Z23: – high word: Número de bloque – low word: Dirección relativa del comando MC7 que provoca errores. El tipo de bloque puede consultarse en el OB 85_DKZ23 (B#16#88: OB, B#16#8C: FC, B#16#8E: FB, B#16#8A: DB).
B#16#35	B#16#A4	DB interfaz PROFINET, no accesible
B#16#34	B#16#A4	DB interfaz PROFINET, de nuevo accesible

OB85_EV_CLASS	OB85_FLT_ID	Significado
B#16#39	B#16#B1	Error de acceso a periferia al actualizar la memoria imagen de proceso de las entradas • OB85_DKZ23: Identificador del tipo de transferencia de la memoria imagen de proceso en el que se ha producido el error de acceso a periferia – B#16#10: Acceso por byte – B#16#20: Acceso por palabra – B#16#30: Acceso por palabra doble – B#16#56 o B#16#57: Transferencia de un área de coherencia configurada • OB85_Z1: Reservado para uso interno de la CPU: dirección base lógica del módulo Si OB85_RESERVED_2 tiene el valor B#16#76, OB85_Z1 contiene el valor de retorno de la instrucción afectada ("DPRD_DAT (Página 6776)", "DPWR_DAT (Página 6778)", "UPDAT_PI (Página 6729)" o "UPDAT_PO (Página 6731)"). • OB85_Z23: – Byte 0: si OB85_DKZ23=B#16#57: n.º de memoria imagen parcial de proceso – Byte 1: irrelevante si OB85_DKZ23=B#16#10, 20 o 30; longitud del área de coherencia en bytes, si OB85_DKZ23=B#16#57 – Byte 0 y byte 1, si OB85_DKZ23=B#16#56: longitud del área de coherencia en bytes – Bytes 2 y 3: la dirección de periferia causante del error de acceso a la periferia, si OB85_DKZ23=B#16#10, 20 o 30; dirección lógica inicial del área de coherencia, si OB85_DKZ23=B#16#57
B#16#39	B#16#B2	Error de acceso a periferia al transferir la memoria imagen de proceso de las salidas a los módulos de salida OB85_DKZ23, OB85_Z1 y OB85_Z23 como con OB85_FLT_ID=B#16#B1
Si se ha configurado para la actualización de la memoria imagen de proceso desde el sistema la llamada repetida del OB 85 en caso de errores de acceso a periferia, se reciben los códigos de error B#16#B1 y B#16#B2.		

OB85_EV_CLASS	OB85_FLT_ID	Significado
B#16#39/B#16#38	B#16#B3	Error de acceso a periferia al actualizar la memoria imagen de proceso de las entradas, entrante/saliente • OB85_DKZ23: Identificador del tipo de transferencia de la memoria imagen de proceso en el que se ha producido el error de acceso a periferia – B#16#10: Acceso por byte – B#16#20: Acceso por palabra – B#16#30: Acceso por palabra doble – B#16#56 o B#16#57: Transferencia de un área de coherencia configurada • OB85_Z1: Reservado para uso interno de la CPU: dirección base lógica del módulo Si OB85_RESERVED_2 tiene el valor B#16#76, OB85_Z1 contiene el valor de retorno de la instrucción afectada ("DPRD_DAT (Página 6776)", "DPWR_DAT (Página 6778)", "UPDAT_PI (Página 6729)" o "UPDAT_PO (Página 6731)"). • OB85_Z23: – Byte 0: si OB85_DKZ23=B#16#57: n.º de memoria imagen parcial de proceso – Byte 1: irrelevante si OB85_DKZ23=B#16#10, 20 o 30; longitud del área de coherencia en bytes, si OB85_DKZ23=B#16#57 – Byte 0 y byte 1, si OB85_DKZ23=B#16#56: longitud del área de coherencia en bytes – Bytes 2 y 3: la dirección de periferia causante del error de acceso a la periferia, si OB85_DKZ23=B#16#10, 20 o 30; dirección lógica inicial del área de coherencia, si OB85_DKZ23=B#16#57
B#16#39/B#16#38	B#16#B4	Error de acceso a periferia al actualizar la memoria imagen de proceso de las salidas, entrante/saliente OB85_DKZ23, OB85_Z1 y OB85_Z23 como con OB85_FLT_ID=B#16#B3
Si se ha configurado para la actualización de la memoria imagen de proceso desde el sistema la llamada del OB 85 solo en caso de errores de acceso a periferia entrantes y salientes, se reciben los códigos de error B#16#B3 y B#16#B4. Después de un arranque en frío o un rearranque completo (en caliente), en la siguiente actualización de la memoria imagen de proceso, todos los accesos a entradas o salidas no existentes se comunicarán como errores de acceso a periferia entrantes.		

Programación del OB 85

El OB 85 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 85 y cárguelo en la CPU como parte del programa de usuario.

El OB 85 puede usarse por ejemplo de la siguiente manera:

- Al evaluar la información de arranque del OB 85, se comprueba qué módulo está defectuoso o no existe (indicación de la dirección inicial del módulo).
- La instrucción "LGC_GADR (Página 7075)" permite detectar el slot del módulo afectado.

OB de fallo del rack (OB 86)

Descripción

El sistema operativo de la CPU llama el OB 86 en los siguientes casos:

- Se detecta el fallo de un aparato de ampliación central (no para S7-300) (tanto con evento entrante como con evento saliente).
- Se detecta el fallo de un sistema maestro DP (tanto con evento entrante como con evento saliente).
- Se detecta el fallo de una estación en periferia descentralizada (PROFIBUS DP o PROFINET IO) (tanto con evento entrante como con evento saliente).
- Se ha desactivado una estación en periferia descentralizada (PROFIBUS DP o PROFINET IO) con la instrucción "D_ACT_DP (Página 6764)" mediante MODE=4.
- Se ha desactivado una estación en periferia descentralizada (PROFIBUS DP o PROFINET IO) con la instrucción "D_ACT_DP (Página 6764)" mediante MODE=3.
- Se detecta el fallo de un sistema PROFINET IO o de una estación PROFINET IO o de una parte de los submódulos de un I-Device PROFINET.

Si no se ha programado el OB 86 y se produce un error de ese tipo, la CPU pasa al estado operativo STOP.

Se puede bloquear la llamada del OB de fallo del rack con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales del OB de fallo del rack

La tabla siguiente contiene las variables temporales (TEMP) del OB de fallo del rack. Se han escogido como nombres de variable los nombres predeterminados del OB 86.

Variable	Tipo de datos	Descripción
OB86_EV_CLASS	BYTE	Clase de evento e identificadores: • B#16#32: Activar una estación con la instrucción "D_ACT_DP" mediante MODE=3 • B#16#33: Desactivar una estación con la instrucción "D_ACT_DP" mediante MODE=4 • B#16#38: evento saliente • B#16#39: evento entrante
OB86_FLT_ID	BYTE	Código de error (valores posibles: B#16#C1, B#16#C2, B#16#C3, B#16#C4, B#16#C5, B#16#C6, B#16#C7, B#16#C8, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)
OB86_PRIORITY	BYTE	Clase de prioridad; ajustable mediante configuración En las CPU S7-1500: prioridad, valor predeterminado: 6

Variable	Tipo de datos	Descripción
OB86_OB_NUMBR	BYTE	N.º de OB (86)
OB86_RESERVED_1	BYTE	Reservado
OB86_RESERVED_2	BYTE	Reservado
OB86_MDL_ADDR	WORD	En función del código de error
OB86_RACKS_FLTD	ARRAY [0 ..31] OF BOOL	En función del código de error
OB86_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Si se desea programar el OB 86 en función de los códigos de error posibles, se recomienda organizar las variables locales de la siguiente manera:

Variable	Tipo de datos
OB86_EV_CLASS	BYTE
OB86_FLT_ID	BYTE
OB86_PRIORITY	BYTE
OB86_OB_NUMBR	BYTE
OB86_RESERVED_1	BYTE
OB86_RESERVED_2	BYTE
OB86_MDL_ADDR	WORD
OB86_Z23	DWORD
OB86_DATE_TIME	DATE_AND_TIME

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB 86.

OB86_EV_CLASS	OB86_FLT_ID	Significado
B#16#39	B#16#C1	Fallo de aparato de ampliación • OB86_MDL_ADDR: Dirección base lógica del IM • OB86_Z23: Contiene un bit por cada aparato de ampliación posible: Cuando un aparato de ampliación provoca la llamada del OB 86, se comunica que ha fallado (están activados los bits correspondientes). Los aparatos de ampliación que hayan fallado antes ya no se muestran. – Bit 0: siempre 0 – Bit 1: 1. Aparato de ampliación – : – Bit 21: 21. Aparato de ampliación – Bits 22 a 29: siempre 0 – Bit 30: Fallo de por lo menos un aparato de ampliación en el rango SIMATIC S5 – Bit 31: siempre 0
B#16#38	B#16#C1	Retorno de un aparato de ampliación OB86_MDL_ADDR como con OB86_FLT_ID=B#16#C1. OB86_Z23 contiene los aparatos de ampliación recuperados (están activados los bits correspondientes).
B#16#38	B#16#C2	Recuperación del aparato de ampliación (fallo del aparato de ampliación saliente con diferencias entre configuración teórica y configuración real) • OB86_MDL_ADDR: Dirección base lógica del IM • OB86_Z23: Contiene un bit por cada aparato de ampliación posible; ver OB86_FLT_ID B#16#C1. Significado de un bit activado: En el aparato de ampliación afectado: – Existen módulos con identificador de módulo erróneo. – Faltan módulos configurados. – Hay por lo menos un módulo defectuoso.
B#16#39	B#16#C3	Periferia descentralizada: Fallo de un sistema maestro DP Solo los eventos entrantes provocan el arranque del OB 86 con código de error B#16#C3. Los eventos salientes provocan el arranque del OB 86 con código de error B#16#C4 y clase de evento B#16#38: El retorno de las estaciones DP subordinadas provoca el arranque del OB 86. • OB86_MDL_ADDR: Dirección base lógica del maestro DP • OB86_Z23: ID del sistema maestro DP: – Bits 0 a 7: Reservado – Bits 8 a 15: ID del sistema maestro DP – Bits 16 a 31: Reservado

OB86_EV_CLASS	OB86_FLT_ID	Significado
B#16#39/B#16#38	B#16#C4	Fallo de una estación DP • OB86_MDL_ADDR: Dirección base lógica del maestro DP • OB86_Z23: Dirección del esclavo DP afectado: – Bits 0 a 7: N.º de estación DP – Bits 8 a 15: ID del sistema maestro DP – Bits 16 a 30: Dirección base lógica con un esclavo S7 o dirección de diagnóstico con un esclavo DP normalizado – Bit 31: Identificador I/O
B#16#39/B#16#38	B#16#C5	Retorno de una estación DP, pero la estación es defectuosa OB86_MDL_ADDR y OB86_Z23 como con FLT_ID=B#16#C4
B#16#38	B#16#C6	Retorno del aparato de ampliación, pero error en parametrización del módulo • OB86_MDL_ADDR: Dirección base lógica del IM • OB86_Z23: Contiene un bit por cada aparato de ampliación posible: – Bit 0: siempre 0 – Bit 1: 1. Aparato de ampliación: – … – Bit 21: 21. Aparato de ampliación – Bits 22 a 30: Reservado – Bit 31: siempre 0 • Significado de un bit activado: En el aparato de ampliación afectado – Existen módulos con identificador de módulo erróneo. – Existen módulos con parámetros erróneos o inexistentes.
B#16#38	B#16#C7	Retorno de una estación DP, pero hay un fallo en la parametrización del módulo • OB86_MDL_ADDR: Dirección base lógica del maestro DP • Dirección del esclavo DP afectado: – Bits 0 a 7: N.º de estación DP – Bits 8 a 15: ID del sistema maestro DP – Bits 16 a 30: Dirección base lógica del esclavo DP – Bit 31: Identificador I/O

OB86_EV_CLASS	OB86_FLT_ID	Significado
B#16#38	B#16#C8	Retorno de una estación DP, pero con diferencias entre la configuración teórica y la configuración real • OB86_MDL_ADDR: Dirección base lógica del maestro DP • OB86_Z23: Dirección del esclavo DP afectado: – Bits 0 a 7: N.º de estación DP – Bits 8 a 15: ID del sistema maestro DP – Bits 16 a 30: Dirección base lógica del esclavo DP – Bit 31: Identificador I/O
B#16#32/B#16#33	B#16#C9	Activar/desactivar un esclavo DP con la instrucción "D_ACT_DP" mediante MODE=3/MODE=4 • OB86_MDL_ADDR: Dirección base lógica del maestro DP • OB86_Z23: Dirección del esclavo DP afectado – Bits 0 a 7: N.º de estación DP – Bits 8 a 15: ID del sistema maestro DP – Bits 16 a 30: Dirección base lógica con un esclavo S7 o dirección de diagnóstico con un esclavo DP normalizado – Bit 31: Identificador I/O
B#16#39	B#16#CA	Fallo de sistema de PROFINET IO • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: 0 (número de estación) – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 31: 0
B#16#39/38	B#16#CB	Fallo de estación/Recuperación de estación PROFINET IO • OB86_RESERVED_1: B#16#C4 • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O Nota: Antes de comprobar en el programa si la causa de un arranque del OB86 es un fallo de la estación PROFINET IO, hay que comprobar si existe un fallo del sistema PROFINET IO.

OB86_EV_CLASS	OB86_FLT_ID	Significado
B#16#/38	B#16#CC	Recuperación de estación PROFINET IO con fallo o mantenimiento • OB86_RESERVED_1: B#16#C4 • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O
B#16#38	B#16#CD	Retorno de estación PROFINET IO, pero la configuración teórica difiere de la configuración real • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O
B#16#38	B#16#CE	Retorno de estación PROFINET IO, pero error en la parametrización del módulo • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O
B#16#32/B#16#33	B#16#CF	Activar/desactivar un dispositivo PROFINET IO con la instrucción "D_ACT_DP" mediante MODE=3/MODE=4 • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O

OB86_EV_CLASS	OB86_FLT_ID	Significado
B#16#39/B#16#38	B#16#F8	Fallo/recuperación de una parte de los submódulos de un I-Device PROFINET • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O
B#16#38	B#16#F9	Recuperación de una parte de los submódulos de un I-Device PROFINET con diferencias en la configuración de dispositivos • OB86_MDL_ADDR: Dirección base lógica del controlador IO • OB86_Z23: – Bits 0 a 10: Número de estación – Bits 11 a 14: ID del sistema IO – Bit 15: 1 – Bits 16 a 30: Dirección base lógica de la estación – Bit 31: Identificador I/O

Nota

Si se utiliza una CPU compatible con DPV1 o PROFINET, puede usarse la instrucción "RALRM (Página 6745)" para obtener información acerca de la alarma, más allá de la información de arranque del OB. Esto también se aplica cuando el maestro DP se opera en el modo compatible con S7.

Programación del OB 86

El OB 86 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 86 y cárguelo en la CPU como parte del programa de usuario.

El OB 86 puede usarse por ejemplo de la siguiente manera:

- Al evaluar la información de arranque del OB 86, se comprueba qué rack es defectuoso o no existe.
- Con la instrucción "WR_USMSG (Página 6947)" se registra un mensaje en el búfer de diagnóstico, que se envía a un dispositivo de observación.

OB de error de comunicación (OB 87)

Descripción

El sistema operativo de la CPU activa el OB 87 cuando se produce un evento provocado por un error de comunicación.

Si no se ha programado el OB 87 y se produce un evento de arranque para el OB 87, la CPU se comporta de la manera siguiente:

- Una CPU S7-300 pasa al estado operativo STOP.
- Una CPU S7-400 no pasa al estado operativo STOP.

Se puede bloquear la llamada del OB de error de comunicación con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales del OB de error de comunicación

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de comunicación. Se han escogido como nombres de variable los nombres predeterminados del OB 87.

Variable	Tipo de datos	Descripción
OB87_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#39
OB87_FLT_ID	BYTE	Código de error (valores posibles: B#16#D2, B#16#D3, B#16#D4, B#16#D5, B#16#E1, B#16#E2, B#16#E3, B#16#E4, B#16#E5, B#16#E6)
OB87_PRIORITY	BYTE	Clase de prioridad; ajustable mediante configuración
OB87_OB_NUMBR	BYTE	N.º de OB (87)
OB87_RESERVED_1	BYTE	Reservado
OB87_RESERVED_2	BYTE	Reservado
OB87_RESERVED_3	WORD	En función del código de error
OB87_RESERVED_4	DWORD	En función del código de error
OB87_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Significado de las variables dependientes del código de error:

Código de error	Variable	Significado
B#16#D2		No es posible enviar los registros de diagnóstico en este momento.
B#16#D3		No se pueden enviar los telegramas de sincronización (maestro).
B#16#D4		Salto de hora no admisible debido a sincronización horaria
B#16#D5		Error al adoptar el tiempo de sincronización (esclavo)
	OB87_RESERVED_3	No contiene más información
	OB87_RESERVED_4	No contiene más información
B#16#E1		Identificación de telegrama errónea en comunicación de datos globales

Código de error	Variable	Significado
B#16#E2		No se puede registrar en un DB el estado de paquete GD
	OB87_RESERVED_3	Número de DB
	OB87_RESERVED_4	• High word: No contiene más información
		• Low word: Número de círculo GD (high byte), número de paquete GD (low byte)
B#16#E3		Error de longitud de telegrama en comunicación de datos globales
B#16#E4		Recibido número de paquete GD no admisible
	OB87_RESERVED_3	Identificador de interfaz (0: Bus K, 1: MPI)
	OB87_RESERVED_4	• High byte: Número de círculo GD • Low byte: No contiene más información
B#16#E5		Error al acceder al DB durante el intercambio de datos a través de bloques de función de comunicación
	OB87_RESERVED_1	Tipo de bloque: B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB
	OB87_RESERVED_3	Reservado para uso interno de la CPU:
	OB87_RESERVED_4	• High word: Número del bloque con el comando MC7 que provoca errores • Low word: Dirección relativa del comando MC7 que provoca errores
B#16#E6		No se puede registrar en el DB el estado global de GD
	OB87_RESERVED_3	Número de DB
	OB87_RESERVED_4	No contiene más información

Programación del OB 87

El OB 87 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 87 y cárguelo en la CPU como parte del programa de usuario.

El OB 87 puede usarse por ejemplo de la siguiente manera:

- Se evalúa la información de arranque del OB 87.
- Si falta el bloque de datos para la información de estado de la comunicación de datos globales, debe crearse un bloque de datos.

OB de interrupción del procesamiento (OB 88)

Descripción

El sistema operativo de la CPU H llama el OB 88 en caso de que se interrumpa el procesamiento de un programa. Posibles causas de la interrupción:

- Excesiva profundidad de anidamiento con errores síncronos
- Excesiva profundidad de anidamiento de llamadas de bloque (pila U)
- Error al asignar datos locales

Si no se ha programado el OB 88 y se produce una interrupción del procesamiento, la CPU pasa al estado operativo STOP (evento W#16#4570).

Si la interrupción se produce en la clase de prioridad 28, la CPU pasa a STOP.

Se puede bloquear la llamada del OB de error de interrupción del procesamiento con la instrucción "DIS_IRT (Página 6906)", volver a habilitarla con la instrucción "EN_IRT (Página 6908)" y retardarla con las instrucciones "DIS_AIRT (Página 6909)" y "EN_AIRT (Página 6910)".

Datos locales del OB de interrupción del procesamiento

La tabla siguiente contiene las variables temporales (TEMP) del OB de interrupción del procesamiento. Se han escogido como nombres de variable los nombres predeterminados del OB 88.

Variable	Tipo de datos	Descripción
OB88_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#35
OB88_SW_FLT	BYTE	Código de error Valores posibles: • B#16#71: Excesiva profundidad de anidamiento de niveles de paréntesis • B#16#72: Excesiva profundidad de anidamiento de Master Control Relais • B#16#73: Excesiva profundidad de anidamiento con errores síncronos • B#16#74: Excesiva profundidad de anidamiento de llamadas de bloque (pila U) • B#16#75: Excesiva profundidad de anidamiento de llamadas de bloque (pila B) • B#16#76: Error al asignar datos locales • B#16#78: Instrucción desconocida • B#16#7A: Instrucción de salto con destino fuera del módulo En la lista de operaciones de la CPU puede consultar cuáles son los códigos de error válidos para la CPU que se utiliza.
OB88_PRIORITY	BYTE	Clase de prioridad: 28
OB88_OB_NUMBR	BYTE	N.º de OB (88)
OB88_BLK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error: • B#16#88: OB • B#16#8C: FC • B#16#8E: FB • B#16##00: No se puede determinar la ubicación del error
OB88_RESERVED_1	BYTE	Reservado
OB88_FLT_PRIORITY	BYTE	Clase de prioridad del OB que provoca errores
OB88_FLT_OB_NUMBR	BYTE	Número del OB que provoca errores

Variable	Tipo de datos	Descripción
OB88_BLK_NUM	WORD	Número del bloque con el comando MC7 que provoca errores
OB88_PRG_ADDR	WORD	Dirección relativa del comando MC7 que provoca errores
OB88_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Programación del OB 88

El OB 88 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 88 y cárguelo en la CPU como parte del programa de usuario.

El OB 88 puede usarse por ejemplo de la siguiente manera:

- Se evalúa la información de arranque del OB 88.
- Debe registrarse la causa del error en un bloque de datos de notificación.

OB de segundo plano (OB 90)

Descripción

S7 ofrece la vigilancia de un tiempo de ciclo máximo y garantiza la observancia de un tiempo de ciclo mínimo. Si el tiempo de procesamiento del OB 1, incluidos todos los procesamientos de alarmas anidados y actividades del sistema, es menor que el tiempo de ciclo mínimo configurado,

- el sistema operativo llama el OB de segundo plano (si está presente en la CPU).
- el sistema operativo retarda el siguiente arranque del OB 1 (si el OB 90 no está presente en la CPU).

Funcionamiento del OB de segundo plano

El OB 90 tiene la prioridad más baja de todos los OBs. Cualquier actividad del sistema o cualquier procesamiento de alarma (incluido el OB 1, una vez finalizado el tiempo de ciclo mínimo) lo interrumpen. La excepción es el procesamiento de instrucciones iniciadas en el OB 90. Estas se procesan con la prioridad del OB 1 y por ello no se interrumpen por el procesamiento del OB 1. No hay ninguna vigilancia de tiempo para el OB 90.

El programa de usuario en el OB 90 se procesa en los siguientes casos empezando por el primer comando:

- Después de un rearranque completo (en caliente), un arranque en frío o un rearranque
- Después de cargar o borrar un bloque

- Después de cargar el OB 90 en la CPU en el estado operativo RUN
- Después de finalizar el ciclo de segundo plano

Nota

En las configuraciones en las que el tiempo de ciclo mínimo y el tiempo de vigilancia de ciclo están próximos, pueden producirse rebases de tiempo de ciclo inesperados al llamar instrucciones del OB de segundo plano.

Datos locales del OB de segundo plano

La tabla siguiente contiene las variables temporales (TEMP) del OB 90. Se han usado como nombres de variable los nombres predeterminados del OB 90.

Variable	Tipo de datos	Descripción
OB90_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#11: activo
OB90_STRT_INF	BYTE	• B#16#91: Rearranque completo (en caliente)/ arranque en frío/rearranque • B#16#92: Borrado de un bloque • B#16#93: Carga del OB 90 en la CPU en el estado operativo RUN • B#16#95: Finalización del ciclo de segundo plano
OB90_PRIORITY	BYTE	Clase de prioridad: 29 (corresponde a la prioridad 0.29)
OB90_OB_NUMBR	BYTE	N.º de OB (90)
OB90_RESERVED_1	BYTE	Reservado
OB90_RESERVED_2	BYTE	Reservado
OB90_RESERVED_3	INT	Reservado
OB90_RESERVED_4	INT	Reservado
OB90_RESERVED_5	INT	Reservado
OB90_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

OBs de arranque (OB 100, OB 101 y OB 102)

Tipos de arranque

Se distinguen los siguientes tipos de arranque:

- Rearranque (S7-400)
- Rearranque completo (en caliente)
- Arranque en frío

La tabla siguiente muestra cuál es el OB que el sistema operativo llama para el arranque en cada caso.

Tipo de arranque	OB correspondiente
Rearranque	OB 101
Rearranque completo (en caliente)	OB 100
Arranque en frío	OB 102

Eventos de arranque para el arranque

La CPU ejecuta un arranque

- Después de POWER ON
- Cuando lo solicita una función de comunicación (mediante comando de menú de la PG o al llamarse las instrucciones "START (Página 7714)" o "RESUME (Página 7718)" en otra CPU)

En función del evento de arranque, de la CPU presente y de los parámetros ajustados de ésta, se llama el correspondiente OB de arranque (OB 100, OB 101 u OB 102). En él se pueden realizar, mediante programación, determinados ajustes predeterminados para el programa cíclico.

Datos locales de los OBs de arranque

La tabla siguiente contiene las variables temporales (TEMP) de un OB de arranque. Se han usado como nombres de variable los nombres predeterminados.

Variable	Tipo de datos	Descripción
OB10x_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#13: activo
OB10x_STRTUP	BYTE	Solicitud de arranque: • B#16#81: Solicitud de rearranque completo (en caliente) manual • B#16#82: Solicitud de rearranque completo (en caliente) automática • B#16#83: Solicitud de rearranque manual • B#16#84: Solicitud de rearranque automática • B#16#85: Solicitud de arranque en frío manual • B#16#86: Solicitud de arranque en frío automática • B#16#87: Maestro Solicitud de arranque en frío manual • B#16#88: Maestro Solicitud de arranque en frío automática • B#16#8A: Maestro Solicitud de rearranque completo (en caliente) manual • B#16#8B: Maestro Solicitud de rearranque completo (en caliente) automática
OB10x_PRIORITY	BYTE	Clase de prioridad: 27 o 1 (en CPU S7-1500)
OB10x_OB_NUMBR	BYTE	N.º de OB (100, 101 ó 102)
OB10x_RESERVED_1	BYTE	Reservado

Variable	Tipo de datos	Descripción
OB10x_RESERVED_2	BYTE	Reservado
OB10x_STOP	WORD	Número del evento que ha puesto la CPU en el estado STOP
OB10x_STRT_INFO	DWORD	Información complementaria sobre el arranque actual
OB10x_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

La tabla siguiente muestra el contenido de la variable OB10x_STRT_INFO.

N.º de bit	Significado	Valores posibles binarios	Explicación
31-24	Información de arranque	0000 xxxx	N.º de rack 0 (solo en CPUs H)
		0100 xxxx	N.º de rack 1 (solo en CPUs H)
		1000 xxxx	N.º de rack 2 (solo en CPUs H)
		0001 xxxx	Multiprocesamiento (solo S7-400)
		0010 xxxx	Funcionamiento de varios módulos centrales en el rack segmentado (solo S7-400)
		xxxx xxx0	No hay diferencias entre la configuración teórica y la configuración real (solo S7-300)
		xxxx xxx1	Hay diferencias entre la configuración teórica y la configuración real (solo S7-300)
		xxxx xx0x	No hay ninguna diferencia entre la configuración teórica y la configuración real
		xxxx xx1x	Hay diferencia entre la configuración teórica y la configuración real
		xxxx x0xx	No hay una CPU H
		xxxx x1xx	CPU H
		xxxx 0xxx	Reloj para sello de tiempo en último POWER ON, no respaldado
		xxxx 1xxx	Reloj para sello de tiempo en último POWER ON, respaldado
23-16	Tipo de arranque efectuado en último lugar	0000 0001	Rearranque completo (en caliente) en multiprocesamiento sin intervención en la CPU de acuerdo con la parametrización (solo S7-400)
		0000 0011	Rearranque completo (en caliente) mediante selector de modo
		0000 0100	Rearranque completo (en caliente) mediante intervención MPI
		0000 0101	Arranque en frío en multiprocesamiento sin intervención en la CPU de acuerdo con la parametrización (solo S7-400)
		0000 0111	Arranque en frío mediante selector de modo
		0000 1000	Arranque en frío mediante intervención MPI

N.º de bit	Significado	Valores posibles binarios	Explicación
		0000 1010	Rearranque en multiprocesamiento sin intervención en la CPU de acuerdo con la parametrización (solo S7-400)
		0000 1011	Rearranque mediante selector de modo (solo S7-400)
		0000 1100	Rearranque mediante intervención MPI (solo S7-400)
		0001 0000	Rearranque completo (en caliente) automático después de POWER ON respaldado
		0001 0001	Arranque en frío después de POWER ON respaldado, de acuerdo con la parametrización
		0001 0011	Rearranque completo (en caliente) mediante selector de modo; último POWER ON respaldado
		0001 0100	Rearranque completo (en caliente) mediante intervención MPI; último POWER ON respaldado
		0010 0000	Rearranque completo (en caliente) automático después de POWER ON no respaldado (con borrado total desde el sistema)
		0010 0001	Arranque en frío después de POWER ON no respaldado (con borrado total desde el sistema)
		0010 0011	Rearranque completo (en caliente) mediante selector de modo; último POWER ON no respaldado
		0010 0100	Rearranque completo (en caliente) mediante intervención MPI; último POWER ON no respaldado
		1010 0000	Rearranque automático después de POWER ON respaldado, de acuerdo con la parametrización (solo S7-400)
15-12	Admisibilidad de los tipos de arranque automáticos	0000	Arranque automático no admisible, solicitud de borrado total
		0001	Arranque automático no admisible, se requiere modificación de parámetros, etc.
		0111	Rearranque completo (en caliente) automático, admisible
		1111	Rearranque completo (en caliente)/rearranque automático, admisible (solo S7-400)
11-8	Admisibilidad de los tipos de arranque manuales	0000	Arranque no admisible, solicitud de borrado total
		0001	Arranque no admisible, se requiere modificación de parámetros, etc.
		0111	Rearranque completo (en caliente) admisible
		1111	Rearranque completo (en caliente) y rearranque, admisibles (solo S7-400)
7-0	Última intervención válida o ajuste del	0000 0000	Ningún tipo de arranque

N.º de bit	Significado	Valores posibles binarios	Explicación
	tipo de arranque automático con POWER ON	0000 0001	Rearranque completo (en caliente) en multiprocesamiento sin intervención en la CPU de acuerdo con la parametrización (solo S7-400)
		0000 0011	Rearranque completo (en caliente) mediante selector de modo
		0000 0100	Rearranque completo (en caliente) mediante intervención MPI
		0000 0101	Arranque en frío en multiprocesamiento sin intervención en la CPU de acuerdo con la parametrización (solo S7-400)
		0000 0111	Arranque en frío mediante selector de modo
		0000 1000	Arranque en frío mediante intervención MPI
		0000 1010	Rearranque en multiprocesamiento sin intervención en la CPU de acuerdo con la parametrización (solo S7-400)
		0000 1011	Rearranque mediante selector de modo (solo S7-400)
		0000 1100	Rearranque mediante intervención MPI (solo S7-400)
		0001 0000	Rearranque completo (en caliente) automático después de POWER ON respaldado
		0001 0001	Arranque en frío después de POWER ON respaldado, de acuerdo con la parametrización
		0001 0011	Rearranque completo (en caliente) mediante selector de modo; último POWER ON respaldado
		0001 0100	Rearranque completo (en caliente) mediante intervención MPI; último POWER ON respaldado
		0010 0000	Rearranque completo (en caliente) automático después de POWER ON no respaldado (con borrado total desde el sistema)
		0010 0001	Arranque en frío después de POWER ON no respaldado (con borrado total desde el sistema)
		0010 0011	Rearranque completo (en caliente) mediante selector de modo; último POWER ON no respaldado
		0010 0100	Rearranque completo (en caliente) mediante intervención MPI; último POWER ON no respaldado
		1010 0000	Rearranque automático después de POWER ON respaldado, de acuerdo con la parametrización (solo S7-400)

Consulte también

Principios básicos del estado operativo "ARRANQUE" (Página 1703)

OB de error de programación (OB 121)

Descripción

El sistema operativo de la CPU activa el OB 121 cuando se produce un evento provocado por un error durante el procesamiento del programa. Si por ejemplo se llama en el programa un bloque que no está cargado en la CPU, se llama el OB 121.

Funcionamiento del OB de error de programación

El OB 121 se ejecuta en la misma clase de prioridad que el bloque interrumpido.

Si el OB 121 no está programado, la CPU cambia su estado operativo de RUN a STOP.

S7 dispone de las siguientes instrucciones para enmascarar y desenmascarar eventos de arranque del OB 121 durante la ejecución del programa:

- La instrucción "MSK_FLT (Página 6902)" enmascara determinados códigos de error.
- La instrucción "DMSK_FLT (Página 6903)" desenmascara los códigos de error que han sido enmascarados por la instrucción "MSK_FLT (Página 6902)".
- La instrucción "READ_ERR (Página 6903)" lee el registro de estado de evento.

Datos locales del OB de error de programación

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de programación. Se han escogido como nombres de variable los nombres predeterminados del OB 121.

Variable	Tipo de datos	Descripción
OB121_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#25
OB121_SW_FLT	BYTE	Código de error (valores posibles: B#16#21, B#16#22, B#16#23, B#16#24, B#16#25, B#16#26, B#16#27, B#16#28, B#16#29, B#16#30, B#16#31, B#16#32, B#16#33, B#16#34, B#16#35, B#16#3A, B#16#3C, B#16#3D, B#16#3E, B#16#3F)
OB121_PRIORITY	BYTE	Clase de prioridad del OB en el que se ha producido el error En las CPU S7-1500: prioridad, valor predeterminado: 7
OB121_OB_NUMBR	BYTE	N.º de OB (121)
OB121_BLK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error (en el caso de S7-300, no se registra aquí ningún valor válido): B#16#88: OB, B#16#8A: DB, B#16#8C: FC, B#16#8E: FB
OB121_RESERVED_1	BYTE	Reservado

Variable	Tipo de datos	Descripción
OB121_FLT_REG	WORD	Origen del error (en función del código de error), p. ej.: • Registro en el que se ha producido el error de conversión • Dirección errónea (error de lectura/escritura) • Número erróneo de un temporizador, un contador o un bloque • Área de memoria errónea
OB121_BLK_NUM	WORD	Número del bloque que contiene el comando MC7 que provoca errores (en el caso de S7-300, no se registra aquí ningún valor válido):
OB121_PRG_ADDR	WORD	Dirección relativa del comando MC7 que provoca errores (en S7-300 no se registra aquí ningún valor válido)
OB121_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Significado de las variables dependientes del código de error:

Código de error	Variable	Significado
B#16#21		Error de conversión BCD
	OB121_FLT_REG	Identificador del registro afectado (W#16#0000: Acu 1)
B#16#22		Error de longitud de área durante lectura
	OB121_RESERVED_1	• Del bit 7 al bit 4 Tipo de acceso: – 0: Acceso por bit, – 1: Acceso por byte, – 2: Acceso por palabra, – 3: Acceso por palabra doble • Del bit 3 al bit 0 Área de memoria: – 0: Área de periferia – 1: Memoria imagen de proceso de las entradas – 2: Memoria imagen de proceso de las salidas – 3: Marcas – 4: DB global – 5: DB de instancia – 6: Datos locales propios – 7: Datos locales del invocante
B#16#23		Error de longitud de área durante escritura
	OB121_RESERVED_1	Como con el código de error B#16#22
B#16#28		Acceso de lectura a un byte, palabra o palabra doble con un puntero cuya dirección de bit es distinta de 0
	OB121_RESERVED_1	Como con el código de error B#16#22

Código de error	Variable	Significado
B#16#29		Acceso de escritura a un byte, palabra o palabra doble con un puntero cuya dirección de bit es distinta de 0 Dirección de byte errónea. El área de datos y el tipo de acceso se pueden consultar en OB121_RESERVED_1
	OB121_RESERVED_1	Como con el código de error B#16#22
B#16#24		Error de área durante lectura
	OB121_FLT_REG	Contiene en el low byte el identificador del área errónea (B#16#86 área propia de datos locales)
B#16#25		Error de área durante escritura
	OB121_FLT_REG	Contiene en el low byte el identificador del área errónea (B#16#86 área propia de datos locales)
B#16#26		Error en número de temporizador
	OB121_FLT_REG	Número no admisible
B#16#27		Error en número de contador
	OB121_FLT_REG	Número no admisible
B#16#30		Acceso de escritura a un DB global protegido contra escritura
	OB121_FLT_REG	Número de DB no admisible
B#16#31		Acceso de escritura a un DB de instancia protegido contra escritura
	OB121_FLT_REG	Número de DB no admisible
B#16#32		Número erróneo de DB al acceder a un DB global
	OB121_FLT_REG	Número de DB no admisible
B#16#33		Número erróneo de DB al acceder a un DB de instancia
	OB121_FLT_REG	Número de DB no admisible
B#16#34		Error de número al llamar la FC
	OB121_FLT_REG	N.º de FC
B#16#35		Error de número FB al llamar el FB
	OB121_FLT_REG	N.º de FB
B#16#3A		Acceso a un DB no cargado; el número de DB está dentro del rango admisible
	OB121_FLT_REG	Número de DB
B#16#3C		Acceso a una FC no cargada; el número de FC está dentro del rango admisible
	OB121_FLT_REG	N.º de FC
B#16#3D		Acceso a una instrucción no presente; el número de SFC está dentro del rango admisible
	OB121_FLT_REG	N.º de SFC
B#16#3E		Acceso a un FB no cargado; el número de FB está dentro del rango admisible
	OB121_FLT_REG	N.º de FB
B#16#3F		Acceso a un SFB no presente; el número de SFB está dentro del rango admisible
	OB121_FLT_REG	N.º de SFB

Programación del OB 121

El OB 121 debe agregarse a los bloques de la CPU. Escriba en el bloque creado el programa que se deba procesar en el OB 121 y cárguelo en la CPU como parte del programa de usuario.

El OB 121 puede usarse por ejemplo de la siguiente manera:

- Se evalúa la información de arranque del OB 121.
- Debe registrarse la causa del error en un bloque de datos de notificación.

OB de error de acceso a la periferia (OB 122)

Descripción

El sistema operativo de la CPU activa el OB 122 cuando se produce un error al acceder a los datos de un módulo. Si por ejemplo la CPU detecta un error de lectura al acceder a los datos de un módulo de señal, el sistema operativo llama el OB 122.

Funcionamiento del OB de error de acceso a la periferia

El OB 122 se ejecuta en la misma clase de prioridad que el bloque interrumpido. Si el OB 122 no está programado, la CPU cambia su estado operativo de RUN a STOP.

S7 dispone de las siguientes instrucciones para enmascarar y desenmascarar eventos de arranque del OB 122 durante la ejecución del programa:

- La instrucción "MSK_FLT (Página 6902)" enmascara determinados códigos de error.
- La instrucción "DMSK_FLT (Página 6903)" desenmascara los códigos de error que han sido enmascarados por la instrucción "MSK_FLT (Página 6902)".
- La instrucción "READ_ERR (Página 6903)" lee el registro de estado de evento.

Datos locales del OB de error de acceso a la periferia

La tabla siguiente contiene las variables temporales (TEMP) del OB de error de acceso a la periferia. Se han escogido como nombres de variable los nombres predeterminados del OB 122.

Variable	Tipo de datos	Descripción
OB122_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#29
OB122_SW_FLT	BYTE	Código de error: • B#16#42: Error de acceso a la periferia, en lectura • B#16#43: Error de acceso a la periferia, en escritura
OB122_PRIORITY	BYTE	Clase de prioridad del OB en el que se ha producido el error En las CPU S7-1500: prioridad, valor predeterminado: 7
OB122_OB_NUMBR	BYTE	N.º de OB (122)
OB122_BLK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error (B#16#88: OB, B#16#8C: FC, B#16#8E: FB) (en S7-300 no se registra aquí ningún valor válido)

Variable	Tipo de datos	Descripción
OB122_MEM_AREA	BYTE	Área de memoria y tipo de acceso: • Del bit 7 al bit 4: Tipo de acceso – 0: Acceso por bit – 1: Acceso por byte – 2: Acceso por palabra – 3: Acceso por palabra doble • Del bit 3 al bit 0:Área de memoria – 0: Área de periferia – 1: Memoria imagen de proceso de las entradas – 2: Memoria imagen de proceso de las salidas
OB122_MEM_ADDR	WORD	Dirección de la memoria en la que se ha producido el error
OB122_BLK_NUM	WORD	Número del bloque que contiene el comando MC7 que provoca errores (en el caso de S7-300, no se registra aquí ningún valor válido):
OB122_PRG_ADDR	WORD	Dirección relativa del comando MC7 que provoca errores (en el caso de S7-300, no se registra aquí ningún valor válido):
OB122_DATE_TIME	DATE_AND_TIME	Fecha y hora en las que se solicitó el OB

Programa de ejemplo para el OB 122

En el siguiente programa de ejemplo se define un valor sustitutivo en la instrucción REPL_VAL. Si falla un módulo de entrada, el procesamiento de la instrucción L PEB0 provoca un error síncrono y arranca el OB 122. De modo predeterminado, el comando de carga lee el valor 0. Sin embargo, con la instrucción "REPL_VAL (Página 6110)" puede especificarse un valor sustitutivo cualquiera, siempre que sea válido para el proceso. Esta instrucción sustituye el contenido del ACU por el valor sustitutivo especificado.

```
AWL                                  Descripción
L    B#16#42                         //Identificador del evento "Error de acceso a periferia en
                                     lectura"
L    #OB122_SW_FLT                   //Código de error de OB 122
==I                                  //Si es igual,
SPB    Qfeh                          //entonces saltar a Qfeh
L    B#16#43                         //Identificador del evento "Error de acceso a periferia en
                                     escritura"
<>I                                  //Si es distinto al código de error de OB 122,
SPB    STOP                          //entonces saltar a STOP
Qfeh:
CALL    "REPL_VAL"                   //Llamada de la instrucción "REPL_VAL"
    VAL:= DW#16#2912                 //Valor sustitutivo que se carga en ACU 1
    RET_VAL:= #Error                 //Guardar el valor de retorno en #Error
L    #Error                          //Valor de retorno de la instrucción "REPL_VAL"
L    0
==I                                  //Si es igual que 0 (no se ha producido ningún error),
```

AWL	Descripción
BEB	//entonces finaliza el bloque
STOP:	
CALL "STP"	//Llamada de la instrucción "STP": La CPU pasa al estado operativo STOP.

Eventos

Eventos e ID del evento

Evento

Dentro de SIMATIC S7, todos los eventos están numerados. Esto permite asignar a un evento un texto de aviso.

ID de evento

Cada evento tiene asignada una ID de evento. La ID de evento tiene la siguiente estructura:

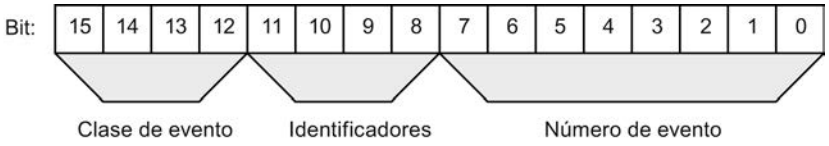


Figura 8-53 Estructura de la ID de evento

Clase de evento

Las clases de evento se clasifican como sigue:

Número	Clase de evento
1	Eventos de OB estándar
2	Eventos de errores síncronos
3	Eventos de errores asíncronos
4	Transiciones de estado operativo
5	Evento de procesamiento de estado operativo
6	Eventos de comunicación
7	Eventos para sistemas de seguridad y de alta disponibilidad
8	Datos de diagnóstico normalizados en módulos
9	Eventos de usuario predefinidos
A, B	Eventos de libre definición
C, D, E	Reservado
F	Eventos para módulos no centrales (p. ej. CP, FM)

Identificador

El identificador permite distinguir los eventos en función del tipo al que pertenecen. Significado de los cuatro bits de identificación:

Número de bit en la ID de evento	Significado
8	= 0: Evento saliente = 1: Evento entrante
9	= 1: Entrada en el búfer de diagnóstico
10	= 1: Error interno
11	= 1: Error externo

Clase de evento 1 - Eventos de OB estándar

Descripción de las IDs de evento

ID de evento	Evento
W#16#1381	Solicitud de rearranque completo (en caliente) manual
W#16#1382	Solicitud de rearranque completo (en caliente) automática
W#16#1383	Solicitud de rearranque manual
W#16#1384	Solicitud de rearranque automática
W#16#1385	Solicitud de arranque en frío manual
W#16#1386	Solicitud de arranque en frío automática
W#16#1387	CPU maestra: Solicitud de arranque en frío manual
W#16#1388	CPU maestra: Solicitud de arranque en frío automática
W#16#138A	CPU maestra: Solicitud de rearranque completo (en caliente) manual
W#16#138B	CPU maestra: Solicitud de rearranque completo (en caliente) automática
W#16#138C	CPU de reserva: Solicitud de arranque manual
W#16#138D	CPU de reserva: Solicitud de arranque automática

Clase de evento 2 - Eventos de error síncronos

Descripción de las IDs de evento

ID de evento	Evento	OB
W#16#2521	BCD - Error de conversión	OB 121
W#16#2522	Error de longitud de área durante lectura	OB 121
W#16#2523	Error de longitud de área durante escritura	OB 121
W#16#2524	Error de área durante lectura	OB 121
W#16#2525	Error de área durante escritura	OB 121
W#16#2526	Error de número de temporizador	OB 121
W#16#2527	Error de número de contador	OB 121

ID de evento	Evento	OB
W#16#2528	Error de alineación durante lectura	OB 121
W#16#2529	Error de alineación durante escritura	OB 121
W#16#2530	Error de escritura al acceder al DB	OB 121
W#16#2531	Error de escritura al acceder al DI	OB 121
W#16#2532	Error de número de bloque al abrir un DB	OB 121
W#16#2533	Error de número de bloque al abrir un DI	OB 121
W#16#2534	Error de número de bloque al llamar la FC	OB 121
W#16#2535	Error de número de bloque al llamar el FB	OB 121
W#16#253A	DB no cargado	OB 121
W#16#253C	FC no cargada	OB 121
W#16#253D	SFC no cargada	OB 121
W#16#253E	FB no cargado	OB 121
W#16#253F	SFB no cargado	OB 121
W#16#2942	Error de acceso a periferia en lectura	OB 122
W#16#2943	Error de acceso a periferia en escritura	OB 122

Clase de evento 3 - Eventos de error asíncronos

Descripción de las ID de evento

ID de evento	Evento	OB
W#16#3501	Rebase por exceso del tiempo de ciclo	OB 80
W#16#3502	Error de solicitud en interfaz de usuario (OB o FRB)	OB 80
W#16#3505	Alarma(s) horaria(s) expiradas debido a salto de hora	OB 80
W#16#3506	Alarma(s) horaria(s) expiradas al reanudar con RUN después de PARADA	OB 80
W#16#3507	Un error múltiple de solicitud de OB ha causado un desbordamiento interno del búfer	OB 80
W#16#3508	Error de tiempo en alarma de sincronismo	OB 80
W#16#3509	Pérdida de alarmas por sobrecarga de alarmas	OB 80
W#16#350A	Reanudación en RUN después de CiR	OB 80
W#16#350B	Error de tiempo en alarma de sincronismo tecnológica	OB 80
W#16#3921/3821	BATTF: Fallo de al menos una batería de respaldo en el aparato central/ eliminado Nota: El evento entrante se produce únicamente en caso de fallo de una de las baterías (si se usan baterías de respaldo redundantes). Si posteriormente falla también la otra batería, el evento no vuelve a producirse.	OB 81
W#16#3922/3822	BAF: Fallo de la tensión de respaldo en el aparato central/ eliminado	OB 81
W#16#3923/3823	Fallo de la alimentación de 24 V en el aparato central/ eliminado	OB 81

ID de evento	Evento	OB
W#16#3925/3825	BATTF: Fallo de al menos una batería de respaldo en un aparato central redundante/ eliminado	OB 81
W#16#3926/3826	BAF: Fallo de la tensión de respaldo en un aparato central redundante/ eliminado	OB 81
W#16#3927/3827	Fallo de la alimentación de 24 V en un aparato central redundante/ eliminado	OB 81
W#16#3931/3831	BATTF: Fallo de al menos una batería de respaldo en al menos un aparato de ampliación/ eliminado	OB 81
W#16#3932/3832	BAF: Fallo de la tensión de respaldo en al menos un aparato de ampliación/ eliminado	OB 81
W#16#3933/3833	Fallo de la alimentación de 24 V en al menos un aparato de ampliación/ eliminado	OB 81
W#16#3942	Alarma de diagnóstico (módulo/submódulo averiado o mantenimiento necesario o ambos)	OB 82
W#16#3842	Alarma de diagnóstico (módulo/submódulo en buen estado)	OB 82
W#16#3951	Módulo PROFINET IO desenchufado	OB 83
W#16#3954	Submódulo/módulo PROFINET IO desenchufado	OB 83
W#16#3854	Submódulo/módulo PROFINET IO enchufado y coincidente con el submódulo/módulo parametrizado	OB 83
W#16#3855	Submódulo/módulo PROFINET IO enchufado, pero no coincidente con el submódulo/módulo parametrizado	OB 83
W#16#3856	Submódulo/módulo PROFINET IO enchufado, pero error en parametrización del módulo	OB 83
W#16#3857	Submódulo/módulo PROFINET IO insertado, pero con fallo o mantenimiento	OB 83
W#16#3858	Submódulo PROFINET IO, error de acceso eliminado	OB 83
W#16#3861	Módulo/módulo de interfaz enchufado, tipo de módulo correcto	OB 83
W#16#3961	Módulo/módulo de interfaz desenchufado o no accesible	OB 83
W#16#3863	Módulo/módulo de interfaz enchufado, pero tipo de módulo incorrecto	OB 83
W#16#3864	Módulo/módulo de interfaz enchufado, pero defectuoso (identificador de módulo ilegible)	OB 83
W#16#3865	Módulo enchufado, pero error en parametrización del módulo	OB 83
W#16#3866	Módulo de nuevo accesible, error de tensión de carga eliminado	OB 83
W#16#3966	Módulo no accesible, error de tensión de carga	OB 83
W#16#3367	Inicio reparametrización de un módulo	OB 83
W#16#3267	Fin reparametrización de un módulo	OB 83
W#16#3968	Reparametrización de un módulo finalizada con errores	OB 83
W#16#3571	Excesiva profundidad de anidamiento de niveles de paréntesis	OB 88

ID de evento	Evento	OB
W#16#3572	Excesiva profundidad de anidamiento de Master Control Relais	OB 88
W#16#3573	Excesiva profundidad de anidamiento con errores síncronos	OB 88
W#16#3574	Excesiva profundidad de anidamiento de llamadas de bloque (pila U)	OB 88
W#16#3575	Excesiva profundidad de anidamiento de llamadas de bloque (pila B)	OB 88
W#16#3576	Error al asignar datos locales	OB 88
W#16#3578	Instrucción desconocida	OB 88
W#16#357A	Instrucción de salto con destino fuera del módulo	OB 88
W#16#3981	Error de interfaz, entrante	-
W#16#3881	Error de interfaz, saliente	-
W#16#3582	Error de memoria detectado por el sistema operativo y eliminado	OB 84
W#16#3583	Acumulación de errores de memoria detectados y corregidos	OB 84
W#16#3585	Error en el sistema operativo del PC (solo con controladores Win AC)	OB 84
W#16#3986	Rendimiento de un acoplamiento H-Sync, disminuido	OB 84
W#16#3587	Error de memoria multibit, detectado y corregido	OB 84
W#16#35A1	Interfaz de usuario (OB o FRB) no existente	OB 85
W#16#35A2	OB no cargado (iniciado por SFC o por el sistema operativo en función de la configuración)	OB 85
W#16#35A3	Error al acceder a un módulo a través del sistema operativo	OB 85
W#16#35A4	DB interfaz PROFInet, no accesible	OB 85
W#16#34A4	DB interfaz PROFInet, de nuevo accesible	OB 85
W#16#39B1	Error de acceso a periferia en actualización de memoria imagen de proceso de las entradas	OB 85
W#16#39B2	Error de acceso a periferia durante transferencia de la memoria imagen de proceso a los módulos de salida	OB 85
W#16#39B3/38B3	Error de acceso a periferia en actualización de memoria imagen de proceso de las entradas	OB 85
W#16#39B4/38B4	Error de acceso a periferia en la actualización de la memoria imagen de proceso de las salidas	OB 85
W#16#38C1	Retorno del aparato de ampliación (de 1 a 21), saliente	OB 86
W#16#39C1	Fallo del aparato de ampliación (de 1 a 21), entrante	OB 86
W#16#38C2	Retorno del aparato de ampliación con diferencias entre configuración teórica y real	OB 86
W#16#39C3	Periferia descentralizada: Fallo del sistema maestro, entrante	OB 86
W#16#39C4	Periferia descentralizada: Fallo de estación, entrante	OB 86
W#16#38C4	Periferia descentralizada: Fallo de estación, saliente	OB 86
W#16#38C5	Periferia descentralizada: Retorno de estación con fallos	OB 86

ID de evento	Evento	OB
W#16#38C6	Retorno del aparato de ampliación, pero error en parametrización del módulo	OB 86
W#16#38C7	DP: Retorno de la estación, pero error en parametrización del módulo	OB 86
W#16#38C8	DP: Retorno de la estación con diferencias entre configuración teórica y real	OB 86
W#16#39CA	Fallo de sistema de PROFINET IO	OB 86
W#16#39CB	Fallo de estación PROFINET IO	OB 86
W#16#38CB	Retorno de estación PROFINET IO	OB 86
W#16#38CC	Recuperación de estación PROFINET IO con fallo o mantenimiento	OB 86
W#16#39CD	Retorno de estación PROFINET IO, pero la configuración teórica difiere de la configuración real	OB 86
W#16#39CE	Retorno de estación PROFINET IO, pero error en la parametrización del módulo	OB 86
W#16#33CF/32CF	Fallo/recuperación de un dispositivo PROFINET IO debido a desactivación/activación mediante la instrucción D_ACT_DP	OB 86
W#16#39F8/38F8	Fallo/recuperación de una parte de los submódulos de un I-Device PROFINET	OB 86
W#16#38F9	Recuperación de una parte de los submódulos de un I-Device PROFINET con diferencias en la configuración de dispositivos	OB 86
W#16#35D2	No es posible enviar las entradas de diagnóstico en este momento	OB 87
W#16#35D3	No se pueden enviar los telegramas de sincronización	OB 87
W#16#35D4	Salto de hora no admisible debido a sincronización horaria	OB 87
W#16#35D5	Error al adoptar el tiempo de sincronización	OB 87
W#16#35E1	Identificación de telegrama errónea con GD	OB 87
W#16#35E2	No se puede registrar en el DB el estado de paquete GD	OB 87
W#16#35E3	Error de longitud de telegrama con GD	OB 87
W#16#35E4	Recibido número de paquete GD no admisible	OB 87
W#16#35E5	Error al acceder al DB con instrucciones de comunicación para conexiones S7 configuradas	OB 87
W#16#35E6	No se puede registrar en el DB el estado global de GD	OB 87

Clase de evento 4 - Eventos de parada y otras transiciones de estado operativo

Descripción de las IDs de evento

ID de evento	Evento
W#16#4300	POWER-ON respaldado
W#16#4301	Transición de estado operativo de STOP a ARRANQUE

ID de evento	Evento
W#16#4302	Transición de estado operativo de ARRANQUE a RUN
W#16#4303	STOP por accionamiento del interruptor de parada
W#16#4304	STOP por accionamiento de STOP en la programadora o mediante la instrucción STOP
W#16#4305	PARADA: Punto de parada alcanzado
W#16#4306	PARADA: Punto de parada abandonado
W#16#4307	Iniciar borrado total por intervención en PG
W#16#4308	Iniciar borrado total mediante accionamiento de interruptor
W#16#4309	Iniciar borrado total automáticamente (POWER-ON no respaldado)
W#16#430A	PARADA abandonada, transición a STOP
W#16#430D	STOP por otra CPU en caso de multiprocesamiento
W#16#430E	Borrado total ejecutado
W#16#430F	STOP del módulo mediante STOP de una CPU
W#16#4510	STOP debido a violación del rango de fechas de la CPU
W#16#4318	Inicio del proceso CiR
W#16#4319	Proceso CiR finalizado
W#16#4520	AVERÍA: STOP no accesible
W#16#4521	AVERÍA: Fallo del procesador de ejecución de comandos
W#16#4522	AVERÍA: Fallo del bloque de reloj
W#16#4523	AVERÍA: Fallo del generador de impulsos de reloj
W#16#4524	AVERÍA: Fallo de la actualización de la célula de tiempo
W#16#4525	AVERÍA: Fallo de sincronización en caso de multiprocesamiento
W#16#4926	AVERÍA: Fallo de la vigilancia de tiempo en accesos a periferia
W#16#4527	AVERÍA: Fallo de la vigilancia de acceso a la periferia
W#16#4528	AVERÍA: Fallo de la vigilancia de tiempo de ciclo
W#16#4530	AVERÍA: Error del test de memoria en la memoria interna
W#16#4931	STOP o AVERÍA: Error del test de memoria en la memoria del módulo
W#16#4532	AVERÍA: Fallo de recursos básicos
W#16#4933	Error de suma de verificación
W#16#4934	AVERÍA: No existe memoria
W#16#4935	AVERÍA: Interrupción debido a watchdog/processor exceptions
W#16#4536	AVERÍA: Selector de modo, averiado
W#16#4540	STOP: La ampliación de la memoria de trabajo interna presenta lagunas. La primera ampliación de memoria es demasiado pequeña o no existe.
W#16#4541	STOP debido al sistema de ejecución de las clases de prioridad
W#16#4542	STOP debido al sistema de administración de objetos
W#16#4543	STOP debido a test y puesta en servicio
W#16#4544	STOP debido al sistema de diagnóstico
W#16#4545	STOP debido al sistema de comunicación
W#16#4546	STOP debido a la administración de memoria de la CPU
W#16#4547	STOP debido a la administración de la memoria imagen de proceso
W#16#4548	STOP debido a la administración de la periferia
W#16#4949	STOP debido a alarma de proceso prolongada

ID de evento	Evento
W#16#454A	STOP debido a configuración: Al arrancar la CPU había un OB cargado que se había desmarcado en la configuración.
W#16#494D	STOP debido a errores de periferia
W#16#494E	STOP debido a fallo de red
W#16#494F	STOP debido a error de configuración
W#16#4550	AVERÍA: Error interno del sistema
W#16#4555	No es posible el rearranque porque se ha excedido el límite del tiempo de vigilancia
W#16#4556	STOP: Solicitud de borrado total debido a comunicación/incoherencia de datos
W#16#4357	Tiempo de vigilancia del módulo, iniciado
W#16#4358	Todos los módulos están listos para funcionar
W#16#4959	No todos los módulos están listos para funcionar
W#16#4562	STOP debido a error de programación (OB no cargado o no posible)
W#16#4563	STOP debido a error de acceso a periferia (OB no cargado o no posible)
W#16#4567	STOP debido a evento H
W#16#4568	STOP debido a error de tiempo (OB no cargado o no posible)
W#16#456A	STOP debido a alarma de diagnóstico (OB no cargado o no posible)
W#16#456B	STOP debido a extracción/inserción (OB no cargado o no posible)
W#16#456C	STOP debido a error de hardware de la CPU (OB no cargado o no posible)
W#16#456D	STOP debido a error de ejecución del programa (OB no cargado o no posible)
W#16#456E	STOP debido a error de comunicación (OB no cargado o no posible)
W#16#456F	STOP debido a fallo del rack (OB no cargado o no posible)
W#16#4570	STOP debido a interrupción del procesamiento (OB no cargado o no posible)
W#16#4571	STOP debido a error de pila de paréntesis
W#16#4572	STOP debido a error de pila del Master Control Relais
W#16#4573	STOP debido a rebase de la profundidad de anidamiento en errores síncronos
W#16#4574	STOP debido a anidamiento excesivo de la pila U en la pila de clases de prioridad
W#16#4575	STOP debido a anidamiento excesivo de la pila B en la pila de clases de prioridad
W#16#4576	STOP debido a error al asignar datos locales
W#16#4578	STOP debido a Opcode desconocido
W#16#457A	STOP debido a error de longitud de código
W#16#457B	STOP debido a DB no cargado en periferia integrada
W#16#497C	STOP debido a tecnología integrada
W#16#457D	Solicitud de borrado total porque se ha modificado la versión de la interfaz interna para la tecnología integrada
W#16#457F	STOP debido a comando STOP
W#16#4580	STOP: Contenido del búfer de respaldo, incoherente (no hay transición a RUN)
W#16#4590	STOP debido a sobrecarga de las funciones internas
W#16#49A0	STOP debido a error de parametrización o diferencia inadmisible entre configuración teórica y real: Arranque bloqueado
W#16#49A1	STOP debido a error de parametrización: Solicitud de borrado total
W#16#49A2	STOP debido a error al reparametrizar: Arranque bloqueado
W#16#49A3	STOP debido a error al reparametrizar: Solicitud de borrado total
W#16#49A4	STOP: Incoherencia de los datos de configuración

ID de evento	Evento
W#16#49A5	STOP: Periferia descentralizada: Incoherencias en la información de configuración cargada
W#16#49A6	STOP: Periferia descentralizada: Información de configuración no válida
W#16#49A7	STOP: Periferia descentralizada: No existe la información de configuración
W#16#49A8	STOP: Visualización de errores de la interfaz para la periferia descentralizada
W#16#43B0	Actualización o copia de seguridad del firmware, finalizada correctamente
W#16#49B1	Datos erróneos de actualización del firmware
W#16#49B2	Actualización de firmware: La versión del hardware no concuerda con el firmware
W#16#49B3	Actualización de firmware: El tipo de módulo no concuerda con el firmware
W#16#43B4	Error al efectuar copia de seguridad del firmware
W#16#43B6	Interrupción de la actualización del firmware por módulos redundantes
W#16#43D0	Rechazo de ACOPLAR debido a violación de las reglas de coordinación.
W#16#43D1	Interrupción de la secuencia ACOPLAR/SINCRONIZAR
W#16#49D2	STOP de la CPU de reserva debido a STOP de la CPU maestra durante el acoplamiento
W#16#43D3	STOP de una CPU de reserva
W#16#49D4	STOP de un maestro porque la CPU interlocutora también es maestra (error de acoplamiento)
W#16#43D5	Rechazo de ACOPLAR debido a que la capacidad de memoria del AS parcial es distinta
W#16#43D6	Rechazo de ACOPLAR debido a que el programa de sistema del AS parcial es distinto
W#16#43D7	Rechazo de ACOPLAR debido a cambio de la configuración
W#16#49D8	STOP/Modo de búsqueda de errores/AVERÍA: Error de hardware detectado debido a otro error
W#16#49D9	STOP debido a error del módulo de sincronización
W#16#49DA	STOP debido a error de sincronización entre CPUs H
W#16#43DC	Interrupción al acoplar con conmutación
W#16#43DD	Rechazo de ACOPLAR debido a funciones de test u otras funciones online en curso
W#16#43DE	Interrupción del proceso de sincronización debido al rebase de un tiempo de vigilancia después de n intentos, nuevo intento de sincronización iniciado
W#16#43DF	Interrupción definitiva del proceso de sincronización debido al rebase de un tiempo de vigilancia después del número máximo de intentos, se requiere nueva intervención
W#16#43E0	Cambio de Modo "Individual" a Acoplar
W#16#43E1	Cambio de Acoplar a Sincronizar
W#16#43E2	Cambio de estado del sistema de Sincronizar a Redundante
W#16#43E3	CPU maestra: Cambio de estado del sistema de Redundante a modo "Individual"
W#16#43E4	CPU de reserva: Cambio de estado del sistema de Redundante a BÚSQUEDA DE ERRORES
W#16#43E5	CPU de reserva: Cambio de BÚSQUEDA DE ERRORES a Acoplar o STOP
W#16#43E6	Interrupción de Acoplar de la CPU de reserva
W#16#43E7	Interrupción de Sincronizar de la CPU de reserva
W#16#43E8	CPU de reserva: Cambio de Acoplar a Arranque

ID de evento	Evento
W#16#43E9	CPU de reserva: Cambio de Arranque a Sincronizar
W#16#43F1	Conmutación de reserva a maestro
W#16#43F2	Acoplamiento de CPUs H incompatibles, bloqueado por el programa de sistema
W#16#42F3	Error de suma de verificación detectado por el sistema operativo y eliminado
W#16#43F4	CPU de reserva: Bloqueo de Acoplar/Sincronizar mediante la instrucción H_CTRL en la CPU maestra

Clase de evento 5 - Eventos de procesamiento de estado operativo

Descripción de las IDs de evento

ID de evento	Evento
W#16#530D	Nueva información de arranque en el estado operativo STOP
W#16#510F	En WinLC se ha producido un problema que ha provocado un STOP o una avería de la CPU.
W#16#5311	Arranque a pesar de falta de aviso de fin del/de los módulo(s)
W#16#5545	Inicio de la reparametrización en el marco de una modificación de la instalación durante el funcionamiento
W#16#5445	Fin de la reparametrización en el marco de una modificación de la instalación durante el funcionamiento
W#16#5961	Error de parametrización
W#16#5962	Error de parametrización con impedimento para el arranque
W#16#5963	Error de parametrización con solicitud de borrado total
W#16#5966	Error de parametrización al conmutar
W#16#5967	Error de parametrización: Datos de configuración desconocidos
W#16#5968/5868	Dirección de emergencia de una interfaz Ethernet, activada/desactivada
W#16#5969	Error de parametrización con impedimento para el arranque
W#16#596A	PROFINET IO: La dirección IP de un dispositivo IO ya existe
W#16#596B	La dirección IP de una interfaz Ethernet ya existe
W#16#596C	El nombre de una interfaz Ethernet ya existe
W#16#596D	La configuración de red existente no concuerda con los requisitos del sistema o la configuración.
W#16#5371	Periferia descentralizada: Fin de la sincronización con un maestro DP
W#16#5979/5879	Aviso de diagnóstico de la interfaz DP: LED EXTF apagado/encendido
W#16#597C	Fallo o aplazamiento del comando Global Control de DP
W#16#5380	Entradas de búfer de diagnóstico de eventos de alarma y asíncronos, bloqueadas
W#16#5581	Faltan una o varias licencias del software runtime.
W#16#5481	Todas las licencias del software runtime están completas nuevamente.
W#16#558A	Diferencia entre las referencias de la CPU configurada y la insertada
W#16#558B	Diferencia entre la versión de firmware de la CPU configurada y la insertada
W#16#5966	Error de parametrización al conmutar
W#16#597C	Fallo o aplazamiento del comando Global Control de DP

ID de evento	Evento
W#16#5395	Periferia descentralizada: Reset de un maestro DP
W#16#5598	Inicio de una incoherencia potencial con sistemas maestros DP debido a CiR
W#16#5498	Fin de una incoherencia potencial con sistemas maestros DP debido a CiR
W#16#59A0	Alarma en la CPU no asignable
W#16#59A1	Error de configuración de la tecnología integrada
W#16#53A2	Firmware de tecnología, cargado correctamente
W#16#59A3	Error al cargar la tecnología integrada
W#16#53A4	Fallo al cargar el DB de tecnología
W#16#55A5	Conflicto de versiones de la interfaz interna con la tecnología integrada
W#16#55A6	Se ha excedido el número máximo de objetos tecnológicos.
W#16#55A7	Ya existe un DB de tecnología de este tipo.
W#16#53FF	Reset al estado de suministro

Clase de evento 6 – Eventos de comunicación

Descripción de las IDs de evento

ID de evento	Evento
W#16#6500	Referencia de conexión (ID) a módulo, duplicada
W#16#6501	Recursos de conexión insuficientes
W#16#6502	Error en la descripción de la conexión
W#16#6905/6805	Problema de recursos en conexiones configuradas de modo pemanente/ eliminado
W#16#6510	Detectado error de estructura de CFB en el DB de instancia en evaluación EPROM
W#16#6514	Número de paquete GD en el módulo, duplicado
W#16#6515	Indicación de longitud incoherente en la información de configuración de GD
W#16#6316	Error de interfaz al arrancar el AS
W#16#6521	No existe módulo ni memoria interna
W#16#6522	Módulo no admisible: Se debe cambiar el módulo y efectuar un borrado total
W#16#6523	Solicitud de borrado total por error al acceder al módulo
W#16#6524	Solicitud de borrado total por error en encabezado de bloque
W#16#6526	Solicitud de borrado total debido a cambio de memoria
W#16#6527	Se ha cambiado la memoria y no es posible el rearranque
W#16#6528	Función de handling de objeto en STOP/PARADA, no es posible el rearranque
W#16#6529	No es posible el arranque durante la función "Cargar programa de usuario"
W#16#652A	No es posible el arranque porque el bloque está duplicado en la memoria de usuario
W#16#652B	No es posible el arranque porque la longitud del bloque es excesiva para el módulo: Se debe cambiar el módulo
W#16#652C	No es posible el arranque porque hay un OB no válido en el módulo

ID de evento	Evento
W#16#6532	No es posible el arranque porque hay información de configuración no válida en el módulo
W#16#6533	Solicitud de borrado total por contenido de módulo no válido
W#16#6534	No es posible el arranque: Bloque repetido varias veces en el módulo
W#16#6535	No es posible el arranque: Memoria insuficiente para recibir el bloque desde el módulo
W#16#6536	No es posible el arranque: El módulo contiene un número de bloque no válido
W#16#6537	No es posible el arranque: El módulo contiene un bloque de longitud no válida
W#16#6538	Datos locales o identificador de la protección contra escritura (con DB) de un bloque para CPU no válidos
W#16#6539	Comando no válido en el bloque (detectado por compilador)
W#16#653A	Solicitud de borrado total, porque los datos locales del OB en el módulo son demasiado cortos
W#16#6543	No es posible el arranque: Tipo de bloque no válido
W#16#6544	No es posible el arranque: Atributo "relevante para la ejecución" no válido
W#16#6545	Lenguaje de creación no válido
W#16#6546	Alcanzado el número máximo de bloques de un tipo de bloques
W#16#6547	Error al parametrizar módulos (no a través de bus P, sino interrupción de descarga)
W#16#6548	Error de plausibilidad en verificación de bloque
W#16#6549	Error de estructura en el bloque
W#16#6550	Un bloque presenta un error de verificación (CRC)
W#16#6551	Un bloque no presenta ningún valor de verificación (CRC)
W#16#6353	Actualización del firmware: Inicio de la descarga del firmware a través de la red
W#16#6253	Actualización del firmware: Fin de la descarga del firmware a través de la red
W#16#6560	SCAN-Overflow
W#16#6981	Error de interfaz, entrante
W#16#6881	Error de interfaz, saliente
W#16#6390	Se ha formateado una Micro Memory Card

Clase de evento 8 - Eventos de diagnóstico para módulos

Descripción de las IDs de evento

ID de evento	Evento	Tipo de módulo
W#16#8x00	Módulo averiado/correcto	Todos
W#16#8x01	Error interno	
W#16#8x02	Error externo	
W#16#8x03	Existe error de canal	
W#16#8x04	Falta tensión auxiliar externa	
W#16#8x05	Falta conector frontal	
W#16#8x06	Falta parametrización	

ID de evento	Evento	Tipo de módulo
W#16#8x07	Parámetro erróneo en módulo	
W#16#8x30	Módulo de usuario erróneo o no existe	
W#16#8x31	Fallo de comunicación	
W#16#8x32	Estado operativo RUN/STOP (STOP: entrante, RUN: saliente)	
W#16#8x33	La vigilancia de tiempo ha reaccionado (watchdog)	
W#16#8x34	Fallo de la tensión de alimentación interna del módulo	
W#16#8x35	Batería BATTF agotada	
W#16#8x36	Fallo de todo el respaldo	
W#16#8x40	Fallo de aparato de ampliación	
W#16#8x41	Fallo de procesador	
W#16#8x42	Error de EPROM	
W#16#8x43	Error de RAM	
W#16#8x44	Error de ADU/DAU	
W#16#8x45	Actuación fusible	
W#16#8x46	Alarma de proceso perdida	
W#16#8x50	Error de configuración/parametrización	Entrada analógica
W#16#8x51	Error de modo común (Common Mode)	
W#16#8x52	Cortocircuito P	
W#16#8x53	Cortocircuito M	
W#16#8x54	Rotura de hilo	
W#16#8x55	Error en canal de referencia	
W#16#8x56	Rebase por defecto del rango de medición	
W#16#8x57	Rebase por exceso del rango de medición	
W#16#8x60	Error de configuración/parametrización	Salida analógica
W#16#8x61	Error síncrono (error modo común)	
W#16#8x62	Cortocircuito P	
W#16#8x63	Cortocircuito M	
W#16#8x64	Rotura de hilo	
W#16#8x66	Falta tensión de carga	
W#16#8x70	Error de configuración/parametrización	Entrada digital
W#16#8x71	Defecto a masa	
W#16#8x72	Cortocircuito P (sensor)	
W#16#8x73	Cortocircuito M	
W#16#8x74	Rotura de hilo	
W#16#8x75	Falta alimentación del sensor	
W#16#8x80	Error de configuración/parametrización	Salida digital
W#16#8x81	Defecto a masa	
W#16#8x82	Cortocircuito P	
W#16#8x83	Cortocircuito M	
W#16#8x84	Rotura de hilo	
W#16#8x85	Actuación fusible	

ID de evento	Evento	Tipo de módulo
W#16#8x86	Falta tensión de carga	
W#16#8x87	Sobretemperatura	
W#16#8xB0	Contador FM: Señal A defectuosa	FM
W#16#8xB1	Contador FM: Señal B defectuosa	
W#16#8xB2	Contador FM: Señal N defectuosa	
W#16#8xB3	Contador FM: Se ha transferido un valor erróneo entre los canales	
W#16#8xB4	Contador FM: Alimentación de sensor de 5,2 V defectuosa	
W#16#8xB5	Contador FM: Alimentación de sensor de 24 V defectuosa	

Clase de evento 9 - Eventos de usuario estándar

Descripción de las IDs de evento

ID de evento	Evento
W#16#9001	Modo de operación automático
W#16#9101	Modo de operación manual
W#16#9x02	ABRIR/CERRAR, CONECTAR/DESCONECTAR
W#16#9x03	Habilitación de comando manual
W#16#9x04	Comando de protección de grupo (ABRIR/CERRAR)
W#16#9x05	Habilitación de proceso
W#16#9x06	Comando de protección de sistema
W#16#9x07	La vigilancia de valor real ha reaccionado
W#16#9x08	La vigilancia de magnitud manipulada ha reaccionado
W#16#9x09	Desviación de regulación superior a lo admisible
W#16#9x0A	Error de posición final
W#16#9x0B	Error en tiempo de ejecución
W#16#9x0C	Error de ejecución de comando (control secuencial)
W#16#9x0D	Estado operativo en marcha > ABRIR
W#16#9x0E	Estado operativo en marcha > CERRAR
W#16#9x0F	Bloqueo de comandos
W#16#9x11	Estado de proceso ABRIR/CONECTADO
W#16#9x12	Estado de proceso CERRAR/DESCONECTADO
W#16#9x13	Estado de proceso Posición intermedia
W#16#9x14	Estado de proceso CONECTADO mediante automático
W#16#9x15	Estado de proceso CONECTADO mediante manual
W#16#9x16	Estado de proceso CONECTADO mediante comando de protección
W#16#9x17	Estado de proceso DESCONECTADO mediante automático
W#16#9x18	Estado de proceso DESCONECTADO mediante manual
W#16#9x19	Estado de proceso DESCONECTADO mediante comando de protección

ID de evento	Evento
W#16#9x21	Fallo de función durante la aproximación
W#16#9x22	Fallo de función durante la retirada
W#16#9x31	Elemento activo (DE/WE) Posición final ABRIR
W#16#9x32	Elemento activo (DE/WE) Posición final No ABRIR
W#16#9x33	Elemento activo (DE/WE) Posición final CERRAR
W#16#9x34	Elemento activo (DE/WE) Posición final No CERRAR
W#16#9x41	Estado no permitido, tiempo de tolerancia transcurrido
W#16#9x42	Estado no permitido, tiempo de tolerancia no transcurrido
W#16#9x43	Error de bloqueo, tiempo de tolerancia = 0
W#16#9x44	Error de bloqueo, tiempo de tolerancia > 0
W#16#9x45	Respuesta no ocurrida
W#16#9x46	Posición final abandonada de modo no admisible, tiempo de tolerancia = 0
W#16#9x47	Posición final abandonada de modo no admisible, tiempo de tolerancia > 0
W#16#9x50	Límite superior rango de señales OSF
W#16#9x51	Límite superior rango de medición OMF
W#16#9x52	Límite inferior rango de señales USF
W#16#9x53	Límite inferior rango de medición UMF
W#16#9x54	Límite superior de alarma OOG
W#16#9x55	Límite inferior de advertencia OG
W#16#9x56	Límite superior de tolerancia OT
W#16#9x57	Límite inferior de tolerancia UT
W#16#9x58	Límite inferior de advertencia UG
W#16#9x59	Límite inferior de alarma UUG
W#16#9x60	GRAPH 7 Etapa entrante/saliente
W#16#9x61	GRAPH 7 Error de bloqueo
W#16#9x62	GRAPH 7 Error de ejecución
W#16#9x63	GRAPH 7 Error detectado
W#16#9x64	GRAPH 7 Error acusado
W#16#9x70	Tendencia excedida en sentido positivo
W#16#9x71	Tendencia excedida en sentido negativo
W#16#9x72	Respuesta no ocurrida
W#16#9x73	Posición final abandonada de modo no admisible
W#16#9x80	Límite excedido, tiempo de tolerancia = 0
W#16#9x81	Límite excedido, tiempo de tolerancia > 0
W#16#9x82	Límite inferior no alcanzado, tiempo de tolerancia = 0
W#16#9x83	Límite inferior no alcanzado, tiempo de tolerancia > 0
W#16#9x84	Gradiente rebasado por exceso, tiempo de tolerancia = 0
W#16#9x85	Gradiente rebasado por exceso, tiempo de tolerancia > 0
W#16#9x86	Gradiente inferior rebasado por defecto, tiempo de tolerancia = 0
W#16#9x87	Gradiente inferior rebasado por defecto, tiempo de tolerancia > 0
W#16#9190/9090	Error de parametrización del usuario, entrante/saliente
W#16#91F0	Desbordamiento
W#16#91F1	Rebase por defecto

ID de evento	Evento
W#16#91F2	División por 0
W#16#91F3	Operación aritmética no admisible

Clases de evento A y B - Eventos libres de usuario

Descripción de las IDs de evento

ID de evento	Evento
W#16#Axyz	Eventos de libre disposición
W#16#Bxyz	Eventos de libre disposición

Clases de evento reservadas

Reservado

Clases de evento reservadas para posteriores ampliaciones:

- C
- D
- E
- F reservada para módulos no centrales (p. ej. CP, FM)

8.1.5.3 Funcionamiento de las CPUs S7-1500

Estados operativos

Principios básicos de los estados operativos de CPUs S7

Introducción

Los estados operativos describen el estado de la CPU. Existen los estados operativos siguientes:

- ARRANQUE
- RUN
- STOP

En estos estados operativos, la CPU es apta para la comunicación, p. ej. a través de la interfaz PN/IE.

Otros estados operativos

Si la CPU no está lista para funcionar, está en uno de los dos estados operativos siguientes:

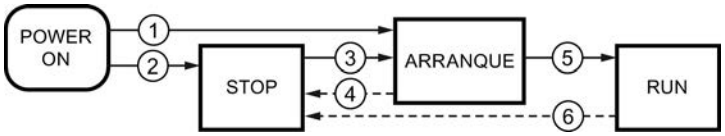
- Sin tensión, es decir, la tensión de red está desconectada.
- Avería, es decir, se ha producido un fallo interno.
  Si el estado "Defectuoso" es provocado por un error de firmware, se señalizará con los LEDs de estado de la CPU (véase la descripción de la CPU en los manuales de producto). Para determinar la causa, proceda del siguiente modo:
  - Desconecte y vuelva a conectar el interruptor de red.
  - Lea el búfer de diagnóstico cuando la CPU arranque y envíe los datos al Customer Support para que los analicen.

  Si la CPU no arranca, cámbiela.

Transiciones de estado operativo

Sinopsis

La figura siguiente muestra los estados operativos y las transiciones de CPUs S7-1500:



La tabla siguiente muestra las condiciones en las que cambian los estados operativos:

N.º	Transición de estado operativo	Condiciones
①	POWER ON → ARRANQUE	Una vez conectada, la CPU pasa al estado operativo "ARRANQUE" si: • está ajustado el tipo de arranque "Arranque en caliente" y • la configuración de hardware y los bloques de programa son coherentes. La memoria no remanente se borra y en el contenido de los DBs no remanentes se restablecen los valores iniciales de la memoria de carga. La memoria remanente y el contenido de los DBs remanentes se conservan.
②	POWER ON → STOP	Si se ajusta el tipo de arranque "Sin arranque", la CPU pasa al estado operativo "STOP" después de conectar la tensión de alimentación. La memoria no remanente se borra y en el contenido de los DBs no remanentes se restablecen los valores iniciales de la memoria de carga. La memoria remanente y el contenido de los DBs remanentes se conservan.
③	STOP → ARRANQUE	La CPU pasa al estado operativo "ARRANQUE" si: • la CPU se pone a "RUN" desde la programadora y • la configuración de hardware y los bloques de programa son coherentes.
④	ARRANQUE → STOP	La CPU pasa al estado operativo "STOP" en los casos siguientes: • Durante el arranque se detecta un error. • La CPU se pone a "STOP" desde la programadora. • Se procesa un comando STOP en el OB de arranque.

N.º	Transición de estado operativo	Condiciones
⑤	ARRANQUE → RUN	Si el arranque es correcto, la CPU pasa a "RUN".
⑥	RUN → STOP	La CPU pasa al estado operativo "STOP" en los casos siguientes: • Se detecta un error que impide seguir trabajando. • La CPU se pone a "STOP" desde la programadora. • Se procesa un comando STOP en el programa de usuario.

Estado operativo "ARRANQUE"

Principios básicos del estado operativo "ARRANQUE"

Función

Antes de que la CPU empiece a procesar el programa de usuario cíclico después de la conexión, se procesa un programa de arranque.

En dicho programa se definen determinadas variables de inicialización para el programa cíclico programando de forma adecuada los OBs de arranque. El número de OBs de arranque no está prescrito. Es posible no programar ningún OB de arranque, programar solo uno o bien varios.

Parametrización del comportamiento de arranque

Existe la posibilidad de determinar si la CPU debe permanecer en estado operativo "STOP" o bien si debe realizarse un rearranque normal. Asimismo, en el grupo "Arranque" de las propiedades de la CPU se parametriza el comportamiento durante el arranque ("RUN" o estado operativo anterior).

Particularidades

Tenga en cuenta los puntos siguientes para el estado operativo "ARRANQUE":

- Las salidas de los módulos están bloqueadas.
- La memoria imagen de proceso está inicializada.
- La memoria imagen de proceso no se actualiza.
  - Para leer el estado actual de las entradas durante el "ARRANQUE", existe la posibilidad de acceder a las entradas mediante un acceso directo a la periferia.
  - Para inicializar las salidas durante el ARRANQUE, es posible escribir valores mediante la memoria imagen de proceso o mediante acceso directo a la periferia. Los valores se emiten en las salidas en la transición al estado operativo "RUN".
- Las marcas no remanentes, los temporizadores y contadores están inicializados.
- Las variables no remanentes en bloques de datos están inicializadas.

- En el arranque no se ejecuta ninguna vigilancia del tiempo de ciclo.
- Los OBs de arranque se procesan. Independientemente del tipo de arranque seleccionado se procesan todos los OBs de arranque programados.
  Los siguientes OBs pueden iniciarse durante el arranque en caso de producirse un evento correspondiente:
  - OB 82: alarma de diagnóstico
  - OB 83: extracción/inserción de módulos
  - OB 86: error de rack
  - OB 121: error de programación (solo con tratamiento de errores global)
  - OB 122: error de acceso a la periferia (solo con tratamiento de errores global)

  Todos los demás OBs no se pueden iniciar hasta que se realice la transición al estado operativo "RUN".

Rearranque (en caliente)

Función

Con el rearranque (en caliente) se borran todas las marcas no remanentes y se restablecen los valores iniciales de la memoria de carga para todos los contenidos de DB no remanentes. Las marcas remanentes y contenidos remanentes de DBs se conservan.

El procesamiento del programa empieza con el primer OB de arranque.

Lanzar un rearranque (en caliente)

En los casos siguientes es posible realizar un "Rearranque (en caliente)" desde un comando de menú de la programadora:

- La CPU está en estado operativo "STOP".
- Después del borrado total
- Después de cargar un programa coherente y una configuración hardware coherente en estado operativo "STOP" de la CPU.

En caso de "Conexión (POWER ON)" se lanza un "Rearranque (en caliente)" si se ha parametrizado el comportamiento de arranque del siguiente modo:

- Tipo de arranque "Arranque en caliente - RUN" (independientemente del estado operativo de la CPU antes de la desconexión (POWER OFF)).
- "Arranque en caliente - modo de operación antes de desconexión (POWER OFF)" (en función del estado operativo de la CPU antes de la desconexión (POWER OFF). La CPU debía estar previamente en RUN.)

Actividades de arranque

Sinopsis

La tabla siguiente muestra qué actividades realiza la CPU durante el arranque:

Actividades en el orden de procesamiento	en caso de rearranque (en caliente)
Borrar marcas no remanentes	sí
Borrar todas las marcas	no
Borrar la memoria imagen de proceso de las salidas	sí
Procesamiento de los OBs de arranque	sí
Actualizar la memoria imagen de proceso de las entradas	sí
Habilitar las salidas tras transición de estado operativo a "RUN"	sí

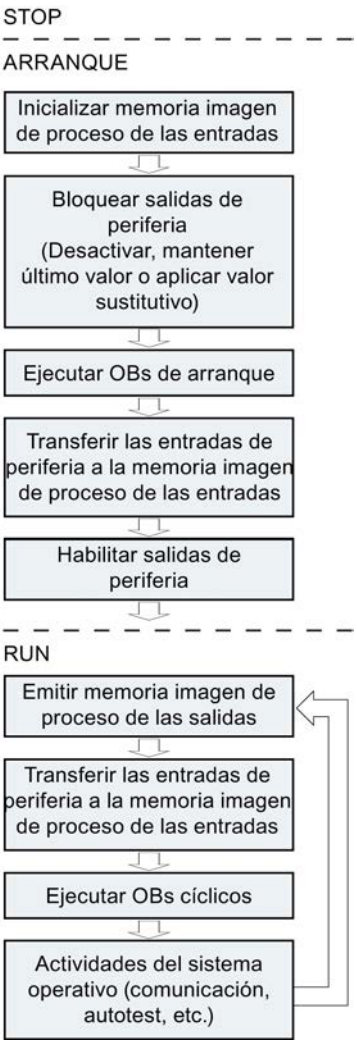
Proceso

La figura siguiente muestra las actividades de la CPU en los estados operativos "STOP", "ARRANQUE" y "RUN".

El estado de las salidas de periferia en el primer ciclo del programa de usuario se define con las medidas siguientes:

- Utilizar módulos de salida parametrizables para aplicar valores sustitutivos o mantener el último valor.
- Preasignar las salidas en los OBs de arranque.

Durante el arranque, todos los eventos de alarma se colocan en una cola de espera y se procesan posteriormente en estado operativo RUN. En estado operativo RUN es posible procesar alarmas de proceso en cualquier momento.



Particularidades del arranque

Comportamiento si la configuración teórica difiere de la real

La configuración teórica se representa mediante la configuración cargada en la CPU. La configuración real es la que existe realmente en el sistema de automatización.

Si la configuración teórica difiere de la real, el parámetro "Compatibilidad hardware soportada" será el que determine el comportamiento de la CPU.

Cancelación de un arranque

Si durante el arranque se producen errores, este se cancela y la CPU vuelve a "STOP".

Un arranque no se realiza o se cancela en las condiciones siguientes:

- Si no se ha insertado ninguna SIMATIC Memory Card o no es válida.
- Si no hay ninguna configuración hardware cargada.

Estado operativo "RUN"

Función

En estado operativo "RUN" se procesa el programa cíclico, controlado por tiempo y por alarmas. Las direcciones que se encuentren en la memoria imagen de proceso "Actualización automática" se actualizarán automáticamente en cada ciclo del programa:

- Se emite la memoria imagen de proceso de las salidas.
- Se lee la memoria imagen de proceso de las entradas.
- Se procesa el programa de usuario.

El intercambio activo de datos entre CPUs S7-1200 a través de la Open User Communication solo es posible en estado operativo "RUN".

Procesamiento del programa de usuario

Una vez que la CPU ha leído las entradas, el programa cíclico se procesa a partir de la primera instrucción y hasta la última instrucción.

Si se ha parametrizado un tiempo mínimo de ciclo, la CPU termina el ciclo una vez transcurrido el tiempo mínimo de ciclo aunque la ejecución del programa de usuario haya requerido menos tiempo.

Para garantizar que el programa cíclico se ejecute en un tiempo determinado, se ajusta un tiempo de vigilancia del ciclo que se puede adaptar en función de las necesidades. Si la ejecución del programa cíclico no ha terminado en ese tiempo, el sistema reacciona con un error de tiempo.

Otros eventos, p. ej. las alarmas de proceso o las alarmas de diagnóstico, pueden interrumpir el flujo del programa y prolongar el tiempo de ciclo.

Estado operativo "STOP"

Función

En estado operativo "STOP" no se procesa el programa de usuario. Todas las salidas se desactivan o reaccionan del modo parametrizado: devuelven el valor sustitutivo parametrizado o mantienen el último valor emitido, con lo que el proceso controlado pasa a un estado operativo seguro.

La CPU comprueba los puntos siguientes:

- Si la CPU se rige por los valores predeterminados o si hay juegos de parámetros
- Si se cumplen las condiciones marco para el comportamiento de arranque programado

Principios básicos del borrado total

Función

El borrado total de la CPU solo es posible en estado operativo "STOP".

En el borrado total, la CPU pasa a una especie de "estado original". Esto significa que:

- Si existe una conexión online entre la programadora o el PC y la CPU, se deshará.
- Tanto el contenido de la memoria de trabajo como los datos remanentes y no remanentes se borran.
- El búfer de diagnóstico, la hora y la dirección IP se conservan.
- A continuación, la CPU se inicializa con los datos de proyecto cargados (configuración hardware, bloques lógicos y de datos, peticiones de forzado permanente). Estos datos se copian de la memoria de carga a la memoria de trabajo.
  Resultado:
  - Si en la configuración hardware se ha ajustado una dirección IP (opción "Ajustar dirección IP en el proyecto"), esta dirección IP es válida después del borrado total.
  - Los bloques de datos ya no tienen valores actuales, sino sus valores iniciales configurados.
  - Las peticiones de forzado permanente siguen activas.

Áreas de memoria

Información de interés sobre las Memory Cards

Función de la Memory Card

La SIMATIC Memory Card para una S7-1500 es una de las tarjetas de memoria SD preformateadas de Siemens para el programa de usuario de la CPU.

Sólo se pueden borrar archivos y carpetas. Si se formatea la Memory Card con Windows, p. ej. con un lector de tarjetas habitual, la Memory Card quedará inutilizada como medio de almacenamiento para una CPU S7.

Configurar el tipo de tarjeta

La Memory Card se puede utilizar como tarjeta de programa o como tarjeta de actualización de firmware.

Para configurar el tipo de tarjeta, inserte la Memory Card en el lector de tarjetas de la programadora y seleccione la carpeta "Lector de tarjetas/memoria USB" del árbol del proyecto. En las propiedades de la Memory Card marcada se puede establecer el tipo de tarjeta:

- Programa
  Una tarjeta de programa sirve como memoria de carga externa para la CPU. Contiene el programa de usuario completo para la CPU. El programa de usuario se transfiere a la memoria de trabajo y se ejecuta aquí. Si se extrae la SIMATIC Memory Card que contiene el programa de usuario, ya no quedará ningún programa.
  Directorio: SIMATIC.S7S
- Firmware Card
  En una Memory Card se puede guardar firmware para módulos S7-1500. Por este motivo, es posible realizar una actualización de firmware mediante una Memory Card especialmente diseñada para ello.
  Directorio: FWUPDATE.S7S

Transferir objetos del proyecto a la Memory Card

Si la Memory Card está insertada en la programadora o en un lector de tarjetas externo, se pueden copiar en ella los siguientes objetos desde el árbol del proyecto:

- Bloques individuales (posibilidad de selección múltiple)
  En este caso, se ofrece una transferencia coherente, es decir, se tiene en cuenta la interdependencia de los bloques por llamadas.
- PLC
  En este caso, todos los objetos relevantes para la ejecución, como los bloques y la configuración hardware, entre otros, se transfieren a la Memory Card, al igual que durante el proceso de carga.

La transferencia puede realizarse arrastrando y soltando los objetos, o con el comando "Lector de tarjetas/memoria USB > Escribir en Memory Card" del menú "Proyecto".

Transferir objetos de la Memory Card al proyecto

Los objetos se transfieren seleccionando el símbolo de la SIMATIC Memory Card en el árbol del proyecto y arrastrándolo a una CPU del proyecto con la función Drag&Drop. El contenido de la SIMATIC Memory Card se debe corresponder con el tipo de CPU del árbol del proyecto. Si ya había bloques, estos se borran antes de la transferencia.

Actualizar firmware con una Memory Card

Los archivos de firmware más recientes pueden adquirirse en Internet, a través de las páginas del "Service & Support":

http://support.automation.siemens.com

Guarde los archivos de firmware en el disco duro e inserte la SIMATIC Memory Card en el lector de tarjetas de su programadora.

Para guardar el archivo en la Memory Card, seleccione en el árbol del proyecto la Memory Card en la carpeta "Lector de tarjetas/memoria USB". Seleccione el menú contextual "Lector de tarjetas/memoria USB > Crear actualización de firmware en Memory Card".

A continuación, siga las instrucciones del portal Service & Support o del manual de sistema S7-1500 para actualizar el firmware.

Con la actualización del firmware, el módulo, bien sea una CPU o un módulo de periferia, obtiene una nueva versión de firmware. Si ha utilizado el módulo en el proyecto, el módulo ya configurado se debe actualizar mediante un cambio de módulo offline con el módulo que contiene la nueva versión del firmware, y después adaptar y cargar el programa y la configuración.

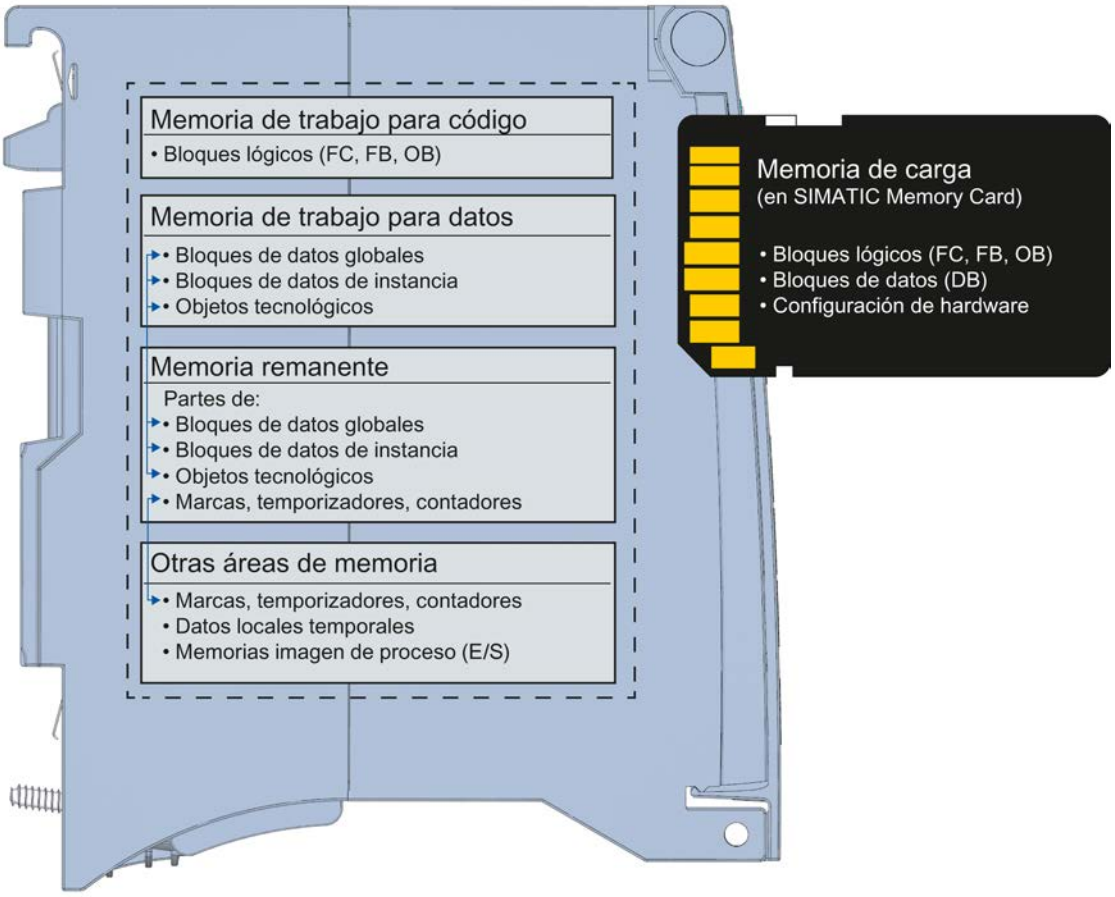
Peticiones de forzado permanente en la Memory Card

Tenga en cuenta que cualquier petición de forzado permanente almacenada en la Memory Card se ejecutará en cada CPU en la que se inserte esta Memory Card. Si no desea que esto suceda, finalice la petición de forzado permanente antes de insertar la Memory Card en otra CPU.

Áreas de memoria de la CPU

Áreas de memoria de la CPU S7-1500

La figura siguiente muestra las áreas de memoria de la CPU y la memoria de carga de la SIMATIC Memory Card. Además de en la memoria de carga, en la SIMATIC Memory Card se pueden encontrar otros datos como, p. ej., recetas, Data Logs o backups de HMI.



Memoria de carga

La memoria de carga es una memoria no volátil que contiene el código del programa, los bloques de datos y la configuración del hardware. Al cargar estos objetos en la CPU, primero se guardan en la memoria de carga. Esta memoria se encuentra en la SIMATIC Memory Card. Para el funcionamiento de la CPU es imprescindible insertar una SIMATIC Memory Card.

Memoria de trabajo

La memoria de trabajo es una memoria volátil que contiene los bloques lógicos y de datos. La memoria de trabajo está integrada en la CPU y no se puede ampliar.

La memoria de trabajo de S7-1500 se divide en dos áreas:

- Memoria de trabajo de código para partes relevantes para la ejecución del código del programa.
- Memoria de trabajo de datos para partes relevantes para la ejecución de bloques de datos y objetos tecnológicos. En las transiciones del estado operativo POWER ON → Arranque y STOP → Arranque se inicializan variables de bloques de datos globales, bloques de datos de instancia y objetos tecnológicos con sus valores de arranque. Las variables remanentes obtienen sus valores actuales guardados en la memoria remanente.

Memoria remanente

La memoria remanente es una memoria no volátil que sirve para guardar copia de una cantidad limitada de datos en caso de un fallo de alimentación. En la memoria remanente se guardan las variables definidas como remanentes y también áreas de operandos.

Dichos datos se conservan incluso después de una desconexión o un fallo de tensión. Todas las variables restantes se pierden, y se restablecen sus valores de arranque en las transiciones del estado operativo de Arranque a POWER ON y de Arranque a STOP.

El contenido de la memoria remanente se borra con las siguientes acciones:

- Borrado total
- Restablecimiento de la configuración de fábrica

En la memoria remanente se guardan también determinadas variables de los objetos tecnológicos. Estas no se borran con el borrado total.

Otras áreas de memoria

Además de las áreas de memoria descritas para el programa de usuario y los datos, la CPU dispone de otras áreas de memoria.

Entre estas áreas cabe mencionar las siguientes áreas relevantes para la ejecución:

- Marcas, temporizadores, contadores
- Datos locales temporales
- Memorias imagen de proceso

La capacidad de las distintas áreas específicas de la CPU se indica en los datos técnicos de la CPU.

Información sobre áreas de memoria

La información sobre las áreas de memoria de su S7-1500 CPU se puede consultar offline y online.

Offline:

al crear o modificar un programa puede determinar si este será demasiado grande para una CPU concreta. La información se encuentra, p. ej., en "Información del programa" del árbol del proyecto. La ficha "Carga de la memoria" ofrece información sobre la capacidad total de las áreas de memoria y sobre el espacio que ya está ocupado. En este punto también encontrará información acerca de las entradas y salidas ocupadas.

En la S7-1500 puede determinar la capacidad total de la memoria de carga (información del programa; el campo Memoria de carga - Total es una lista desplegable para seleccionar la capacidad de la memoria de carga).

Online:

Inicie "Online y diagnóstico" en el árbol del proyecto. La función "Memoria" ofrece información sobre el espacio lleno de todas las memorias de las CPU.

Encontrará más información sobre la carga de la memoria de la CPU utilizada en las fichas "Plano de ocupación" y "Estructura de llamadas" de la información del programa, así como en la tabla de variables PLC.

Consulte también

Resumen de la información del programa disponible (Página 3053)

Mostrar el nivel de llenado de todos los tipos de memoria de una CPU (Página 2178)

Áreas de memoria remanentes

Introducción

Las CPUs S7-1500 incorporan una memoria remanente para almacenar datos remanentes en POWER OFF. La capacidad de la memoria remanente está documentada en los datos técnicos de la CPU.

La carga de la memoria remanente de la CPU configurada se puede consultar offline en "Información del programa > Carga de la memoria" del árbol del proyecto y online en "Online y diagnóstico > Diagnóstico > Memoria" del árbol del proyecto.

Si define los datos como remanentes, su contenido se conserva tras un fallo de la alimentación, al arrancar la CPU y al cargar un programa modificado.

Los siguientes datos u objetos pueden definirse como remanentes:

- Marcas, temporizadores, contadores
- Variables de bloques de datos globales
- Variables de bloques de datos de instancia de un bloque de función

Determinadas variables de los objetos tecnológicos son siempre remanentes como, p. ej., valores de ajuste de encóders absolutos.

Marcas, temporizadores, contadores

El número de marcas, temporizadores y contadores remanentes se puede definir en la tabla de variables PLC a través del botón "Remanencia".

Variables de un bloque de datos global

En un bloque de datos global se pueden definir como remanentes algunas o todas las variables del bloque en función del ajuste del acceso:

- Acceso optimizado al bloque activado: La remanencia se puede ajustar para cada variable individual.
- Acceso optimizado al bloque desactivado: el ajuste de remanencia es válido para todas las variables del bloque de datos, es decir, o todas las variables son remanentes o no lo es ninguna.

Variables de un bloque de datos de instancia (bloque de función)

Las variables del bloque de datos de instancia de un bloque de función se pueden definir como remanentes. En función del acceso al bloque de datos de instancia, la remanencia puede definirse para variables individuales o conjuntamente para todas las variables del bloque:

- Acceso optimizado al bloque activado: puede definir variables individuales como remanentes tanto en la interfaz del bloque de función como también en el bloque de datos de instancia asignado.
- Acceso optimizado al bloque desactivado: puede definir conjuntamente la remanencia para todas las variables en el bloque de datos de instancia.

Generación de un bloque de datos en el programa de usuario

Con la instrucción "Create_DB" se puede crear un bloque de datos. En función de la selección del parámetro ATTRIB, el bloque de datos generado tiene la propiedad "remanente" o "no remanente". Aquí no se permite ajustar la remanencia para variables individuales.

Variables de objetos tecnológicos

Determinadas variables de los objetos tecnológicos son siempre remanentes como, p. ej., valores de ajuste de encóders absolutos. STEP 7 administra automáticamente la remanencia de objetos tecnológicos, de manera que no haga falta configurar ninguna remanencia. Estas variables están protegidas contra un borrado total y sólo pueden borrarse restableciendo la configuración de fábrica.

Resumen del comportamiento remanente

Comportamiento remanente de los objetos de memoria

Además de las áreas de memoria que se han descrito existen otros objetos con comportamiento remanente, p. ej., el búfer de diagnóstico. Estos objetos no ocupan memoria remanente.

La tabla siguiente explica el comportamiento remanente de los objetos de memoria

- en las transiciones de estado operativo STOP → Arranque y POWER OFF → Arranque
- en las funciones Borrado total y Restablecer configuración de fábrica, que afectan a la memoria

Tabla 8-91 Comportamiento remanente de los objetos de memoria ("x" = se conserva el contenido; "-" = se inicializa el contenido)

Objeto de memoria	Transiciones de estado operativo		Borrado total	Restablecimiento de la configuración de fábrica
	STOP → ARRANQUE	POWER OFF → ARRANQUE		
Valores actuales de los bloques de datos, bloques de datos de instancia	Ajustable en las propiedades del DB.		-	-
Marcas, temporizadores y contadores, configurables como remanentes	x	x	-	-
Marcas, temporizadores y contadores, configurables como no remanentes	-	-	-	-
Determinadas variables remanentes de objetos tecnológicos (p. ej. valores de ajuste de encóders absolutos)	x		x	-
Entradas del búfer de diagnóstico (área remanente)	x	x	x	-
Entradas del búfer de diagnóstico (área no remanente)	x	-	-	-
Contador de horas de funcionamiento	x	x	x	-
Hora	x	x	x	-

Búfer de diagnóstico

En las CPUs S7-1500, una parte del búfer de diagnóstico es remanente. Las entradas más actuales del búfer de diagnóstico se conservan tras un fallo de la alimentación y están protegidas contra un borrado total. La parte remanente del búfer de diagnóstico solo puede borrarse restableciendo la configuración de fábrica. El número de las entradas remanentes del búfer de diagnóstico depende de la CPU. Las entradas contenidas en el búfer de diagnóstico no requieren ninguna memoria remanente.

Contador de horas de funcionamiento

El contador de horas de funcionamiento de las CPUs S7-1500 es remanente y está protegido contra un borrado total. Restableciendo la configuración de fábrica se pone a cero.

Hora

La hora de las CPUs S7-1500 es remanente y está protegida contra un borrado total. Restableciendo la configuración de fábrica se pone la hora a 01.01.2012 00:00:00.

Comportamiento de memorización al cargar modificaciones de software

Introducción

La CPU S7-1500 ofrece la posibilidad de cargar modificaciones de software en STOP y en RUN de la CPU sin que la carga afecte a los valores actuales de las variables ya cargadas.

Cargue en STEP 7 las modificaciones del software en "Cargar en dispositivo > Software (sólo cambios)".

Efectos de las modificaciones de software en variables PLC

Puede cargar las siguientes modificaciones de software **sin** que se vean afectados los valores actuales de las variables ya cargadas:

- Modificación de nombres
- Modificación de comentarios
- Ampliación con nuevas variables
- Borrado de variables
- Modificación de los ajustes de remanencia para marcas, temporizadores y contadores.

Al cargar las siguientes modificaciones de software, se ven afectados los valores actuales:

- Modificación de tipos de datos
- Modificación de direcciones

Efectos de modificaciones de software en bloques de datos con reserva de memoria

Si utiliza la reserva de memoria para bloques de datos (atributo "Acceso optimizado al bloque" y botón "Cargar sin reinicializar" activado), puede cargar las siguientes modificaciones de software **sin** que se reinicialicen los valores actuales de variables ya cargadas:

- Modificación de valores de arranque
- Modificación de comentarios
- Ampliación con nuevas variables

Para cargar las siguientes modificaciones de software, desactive el botón "Cargar sin reinicializar". En la siguiente carga se reinicializan todos los valores actuales del bloque de datos:

- Modificación de nombres
- Modificación de tipos de datos
- Modificación de ajustes de remanencia
- Borrado de variables
- Modificación de ajustes de la reserva de memoria

Efectos de modificaciones de software en bloques de datos sin reserva de memoria

Si no utiliza reserva de memoria, puede cargar las siguientes modificaciones de software sin que se reinicialicen los valores actuales de variables ya cargadas:

- Modificación de valores de arranque
- Modificación de comentarios

Al cargar las siguientes modificaciones de software, se reinicializan todos los valores actuales del bloque de datos:

- Modificación del nombre
- Modificación de tipos de datos
- Modificación de ajustes de remanencia
- Ampliación con nuevas variables
- Borrado de variables

Reserva de memoria de bloques de datos

Cada bloque de función o de datos con el atributo "Acceso optimizado al bloque" activado contiene por ajuste predeterminado una reserva de memoria que puede utilizarse para posteriores modificaciones de interfaz. La reserva de memoria no se utiliza inicialmente. Si después de compilar y cargar el bloque comprueba que quiere recargar modificaciones de la interfaz, debe activar la reserva de memoria. Todas las variables que declare posteriormente se colocan en la reserva de memoria. Al realizar la carga, las nuevas variables se inicializan con sus valores de arranque, mientras que las variables que ya se han cargado no se reinicializan.

El ajuste de la reserva de memoria se encuentra en "Cargar sin reinicializar" de las propiedades del bloque de datos.

Consulte también

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)

Inicializar valores de ajuste en el programa online (Página 2964)

Información general acerca de la carga (Página 364)

Áreas de datos de periferia

Acceso a las direcciones de periferia

Direcciones de periferia

Cuando se inserta un módulo en la vista de dispositivos, sus datos de usuario se encuentran de forma estándar en la memoria imagen de proceso de la CPU S7-1500. La CPU acepta automáticamente el intercambio de datos entre el módulo y el área de memoria imagen de proceso al actualizarse esta última.

Si el programa debe acceder al módulo directamente y no a través de la memoria imagen de proceso, la dirección de periferia se debe completar con ":P".

En cualquier caso, el acceso directo a la periferia no ofrece un mejor rendimiento de las CPUs S7-1500, por lo que normalmente deberá direccionar la periferia a través de memorias imagen de proceso o memorias imagen parciales de proceso.

Dirección inicial de un módulo

Definición

La dirección inicial es la dirección de byte más baja de un módulo. Representa la dirección inicial del área de datos de usuario del módulo.

Configurar las direcciones iniciales de los módulos

La asignación entre las direcciones utilizadas en el programa de usuario y los módulos se realiza mediante configuración de los módulos.

Las direcciones iniciales asignadas automáticamente al colocar los módulos se modifican en las propiedades del módulo (área "Direcciones E/S...").

También existe la posibilidad de ajustar si las direcciones estarán en la memoria imagen de proceso o en qué memoria imagen parcial de proceso.

Además de la memoria imagen parcial de proceso, puede seleccionar también el bloque de organización al que está asignada la memoria imagen parcial de proceso. A excepción de los OB de alarma de sincronismo, al llamar al OB asignado se actualiza la memoria imagen parcial de proceso correspondiente. Inmediatamente antes del arranque de OB se actualiza la memoria imagen parcial de proceso de las entradas, e inmediatamente después de finalizar el OB se transfiere la memoria imagen parcial de proceso de las salidas a los módulos.

Seleccione la memoria imagen de proceso "Actualización automática" si la memoria imagen de proceso en la que se encuentran las direcciones del módulo debe actualizarse automáticamente en el punto de control del ciclo.

Evaluación del estado de valor

Introducción

La información de calidad (QI, Quality Information) aporta información de diagnóstico sobre los canales de la periferia que se ponen a disposición del programa de usuario a través de la memoria imagen de proceso de las entradas (IPE). La información de calidad se transfiere de manera síncrona con los datos de usuario.

Cada bit de la información de calidad está asignado a un canal y proporciona información sobre la validez del valor (0 = el valor es erróneo).

Por ejemplo, si existe una rotura de hilo en una entrada digital, la señal de los datos de usuario es lógicamente "0". Debido al diagnóstico de rotura de hilo, el módulo pone el bit correspondiente también a 0 en la información de calidad, lo que significa que existe la posibilidad de determinar que el valor no es válido consultando la información de calidad.

La asignación de los bytes de la información de calidad en la memoria imagen de proceso de las entradas depende del módulo utilizado. Encontrará información detallada en el manual de producto del respectivo módulo de periferia.

Activación de la información de calidad

Active la casilla de verificación "Información de calidad" en las propiedades del módulo de periferia si desea evaluar la información de calidad del canal. Activando la información de calidad se reservan direcciones adicionales en la memoria imagen de proceso de las entradas (IPE) para la información de calidad. En los módulos de entrada, STEP 7 reserva direcciones de entrada a continuación de los datos de usuario, mientras que en los módulos de salida se reservan las siguientes direcciones de entrada libres.

Particularidades de los módulos con MSI/MSO

Requisitos:

- Archivo GSD para módulos de periferia con MSI/MSO
- El dispositivo IO soporta MSI/MSO (p. ej. IM 155-5 PN ST a partir de la versión de firmware V2.0.0)

En los módulos aptos para diagnóstico para los que se ha configurado Shared Input interna del módulo (MSI) o Shared Output interna del módulo (MSO), la información de calidad tiene el significado siguiente:

MSI:

En el submódulo 1 (= submódulo base), la información de calidad 0 indica que el valor es erróneo.

En los submódulos 2 a 4 (= submódulos MSI), la información de calidad 0 indica que el valor es erróneo o que el submódulo base todavía no está parametrizado (no está operativo).

MSO:

En el submódulo 1 (= submódulo base), la información de calidad 1 indica que el valor de salida especificado por el programa de usuario es realmente el que se obtiene en borne del módulo.

Causas posibles para una información de calidad = 0:

- El valor es erróneo porque falta tensión de alimentación, por ejemplo.
- El controlador IO del submódulo base está en estado STOP.

En los submódulos 2 a 4 (= submódulos MSO), la información de calidad 1 indica que el valor de salida especificado por el programa de usuario es realmente el que se obtiene en el borne del módulo.

Causas posibles para una información de calidad = 0:

- El valor es erróneo porque falta tensión de alimentación, por ejemplo.
- El controlador IO del submódulo base está en estado STOP.
- El submódulo base todavía no está configurado.

Consulte también

OBs de ciclo (Página 1963)

Ejemplo de evaluación del estado de valor (Página 1919)

Diagnóstico del sistema (http://support.automation.siemens.com/WW/view/es/68052815)

Shared Input/Shared Output interna del módulo (MSI/MSO) (http://support.automation.siemens.com/WW/view/es/68052815)

Ejemplo de evaluación del estado de valor

El módulo DI 32x24VDC HF de S7-1500 puede configurarse en STEP 7 (vista de dispositivos) para diferentes ámbitos de aplicación. En función de la configuración se asignan direcciones adicionales/diferentes en la memoria imagen de proceso de las entradas.

En este ejemplo se muestran diferentes configuraciones con la respectiva información de calidad parametrizada y se describen las repercusiones en la asignación de direcciones.

Opciones de configuración de DI 32x24VDC HF

Son posibles las configuraciones siguientes:

- 1 x 32 canales sin información de calidad
- 1 x 32 canales con información de calidad
- 4 x 8 canales sin información de calidad
- 4 x 8 canales con información de calidad
- 1 x 32 canales con información de calidad para Shared Input interna del módulo (MSI) con hasta 4 submódulos

En las configuraciones con información de calidad, cada canal tiene asignado un bit adicional para la información de calidad. El bit de información de calidad indica si el valor digital leído es válido. (0 = el valor es erróneo).

Área de direcciones en la configuración como DI 32x24VDC HF de 32 canales con información de calidad

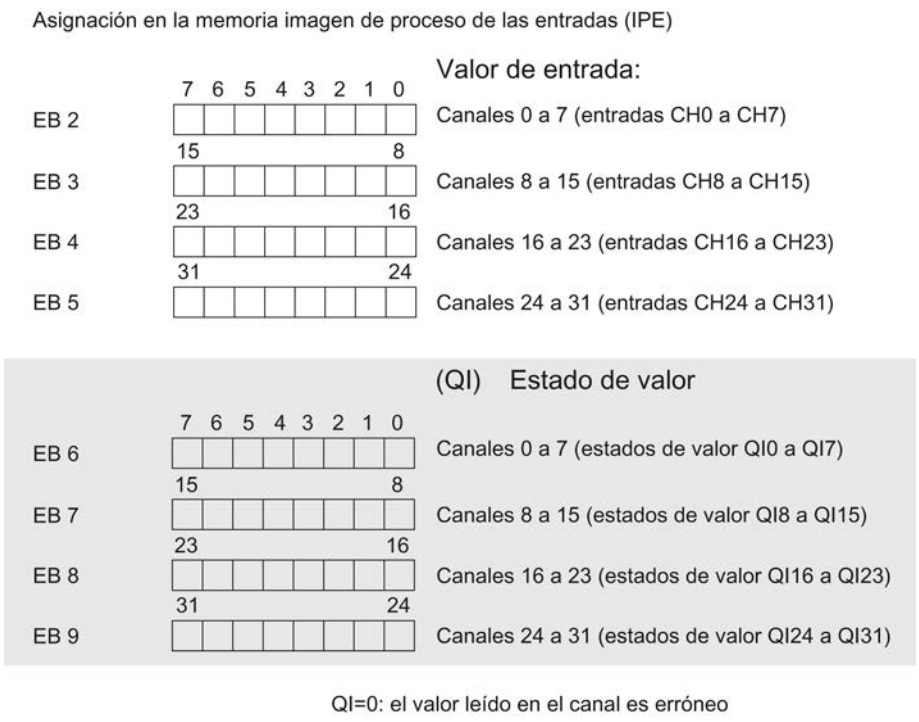
Al principio se conserva la configuración DI predeterminada (32 canales; ni división en submódulos ni configuración de Shared Input interna del módulo (MSI)), pero debe activarse la información de calidad.

- Active la casilla de verificación "Habilitar información de calidad" en el área "Configuración DI" de las propiedades del módulo (ventana de inspección).

La siguiente figura muestra la asignación del área de direcciones en la configuración como módulo de 32 canales con información de calidad. La dirección inicial del módulo puede asignarse libremente. Las direcciones de los canales se derivan de dicha dirección inicial.

Ejemplo: dirección inicial 2 (byte de entrada 2), dirección final 9 (byte de entrada 9).

STEP 7 reserva automáticamente las direcciones para los valores de entrada (4 bytes) y las direcciones siguientes para la información de calidad (también 4 bytes), con lo que se obtiene la siguiente asignación de direcciones:



Área de direcciones en la configuración como DI 32x24VDC HF de 4 x 8 canales con información de calidad

La configuración DI debe adaptarse de manera que los 32 canales del módulo DI 32x24VDC HF del S7-1500 puedan dividirse en cuatro submódulos de 8 canales cada uno. Estos

submódulos pueden asignarse a diferentes controladores IO si el módulo se utiliza en un Shared Device. En ese caso, la información de calidad también debe estar activada.

- Seleccione la configuración de submódulo "4 submódulos con 8 entradas digitales" en el área "Configuración DI" de las propiedades del módulo (ventana de inspección).
- Active la casilla de verificación "Habilitar información de calidad".

En la navegación local de la ventana de inspección aparecen los 4 submódulos con nombres propios (p. ej. DI 32x24VDC BA_1)

A diferencia de la configuración como módulo de 1 x 32 canales, cada uno de los cuatro submódulos posee una dirección inicial que se puede asignar libremente.

Ejemplo: las direcciones iniciales se asignan del modo siguiente:

- 1. submódulo: 10
- 2. submódulo: 20
- 3. submódulo: 30
- 4. submódulo: 40

STEP 7 reserva automáticamente un byte para la información de calidad de cada submódulo. El byte para la información de calidad tiene la dirección "Dirección inicial + 1", con lo que se obtiene la siguiente asignación de direcciones:

Asignación en la memoria imagen de proceso de las entradas (IPE)

	7 6 5 4 3 2 1 0	Valor de entrada:	
EB 10		Canales 0 a 7 (entradas CH0 a CH7)	1.er submódulo
EB 20	15 … 8	Canales 8 a 15 (entradas CH8 a CH15)	2.º submódulo
EB 30	23 … 16	Canales 16 a 23 (entradas CH16 a CH23)	3.er submódulo
EB 40	31 … 24	Canales 24 a 31 (entradas CH24 a CH31)	4.º submódulo

	7 6 5 4 3 2 1 0	(QI) Estado de valor	
EB 11		Canales 0 a 7 (estados de valor QI0 a QI7)	1.er submódulo
EB 21	15 … 8	Canales 8 a 16 (estados de valor QI8 a QI15)	2.º submódulo
EB 31	23 … 16	Canales 16 a 23 (estados de valor QI16 a QI23)	3.er submódulo
EB 41	31 … 24	Canales 24 a 31 (estados de valor QI24 a QI31)	4.º submódulo

QI = 0: el valor leído en el canal es erróneo

Área de direcciones en la configuración como DI 32x24VDC HF de 1 x 32 canales con MSI y información de calidad

La configuración mostrada utiliza la función "Shared Input interna del módulo (MSI)". Esta función permite la lectura simultánea de los valores de entrada por distintos controladores IO (CPU) si el módulo se utiliza en un Shared Device (ET 200MP).

En la configuración del módulo de 1 x 32 canales con Shared Input interna del módulo (MSI), los canales 0 a 31 del módulo se copian en hasta 4 submódulos. Por lo tanto, los canales 0 a 31 tienen valores de entrada idénticos en los distintos submódulos. Estos submódulos pueden asignarse a hasta cuatro controladores IO si el módulo se utiliza en un Shared Device. Cada controlador IO puede tener acceso de lectura a los mismos canales ("escucha").

El número de submódulos disponibles (= copias de las entradas) depende del módulo de interfaz utilizado. Lea las indicaciones respectivas del manual de producto que corresponda al módulo de interfaz.

- Active en el área "Configuración DI" de las propiedades del módulo (ventana de inspección) las siguientes opciones:
  - En "División del módulo": ninguna
  - En "Copias del módulo": una copia adicional de las entradas
    Este ajuste provoca que un controlador IO adicional pueda leer las entradas del módulo.

La información de calidad está habilitada automáticamente en configuraciones MSI (no modificable). El motivo de este ajuste es que un controlador IO "que escucha" del Shared Device puede determinar mediante la información de calidad si el módulo ya está operativo y ofrece valores válidos.

En lo sucesivo, los submódulos que surgen de los ajustes descritos anteriormente reciben los siguientes nombres:

- Submódulo base: se trata del submódulo que se asigna al controlador IO que realiza la parametrización. Solo el controlador IO que realiza la parametrización tiene acceso a diagnósticos del módulo y recibe alarmas de proceso.
- Submódulo MSI: se trata del submódulo asignado a un controlador IO "que escucha". El controlador IO que escucha no tiene acceso a diagnósticos del módulo y únicamente recibe información sobre la validez de los valores de entrada por medio de la información de calidad.

Información de calidad (Quality Information, QI)

El significado de la información de calidad depende del submódulo en cuestión.

En el submódulo 1 (= submódulo base), la información de calidad 0 indica que el valor es erróneo.

En los submódulos 2 a 4 (= submódulos MSI o copia de las entradas), la información de calidad 0 indica que el valor es erróneo o que el submódulo base todavía no está parametrizado (no operativo).

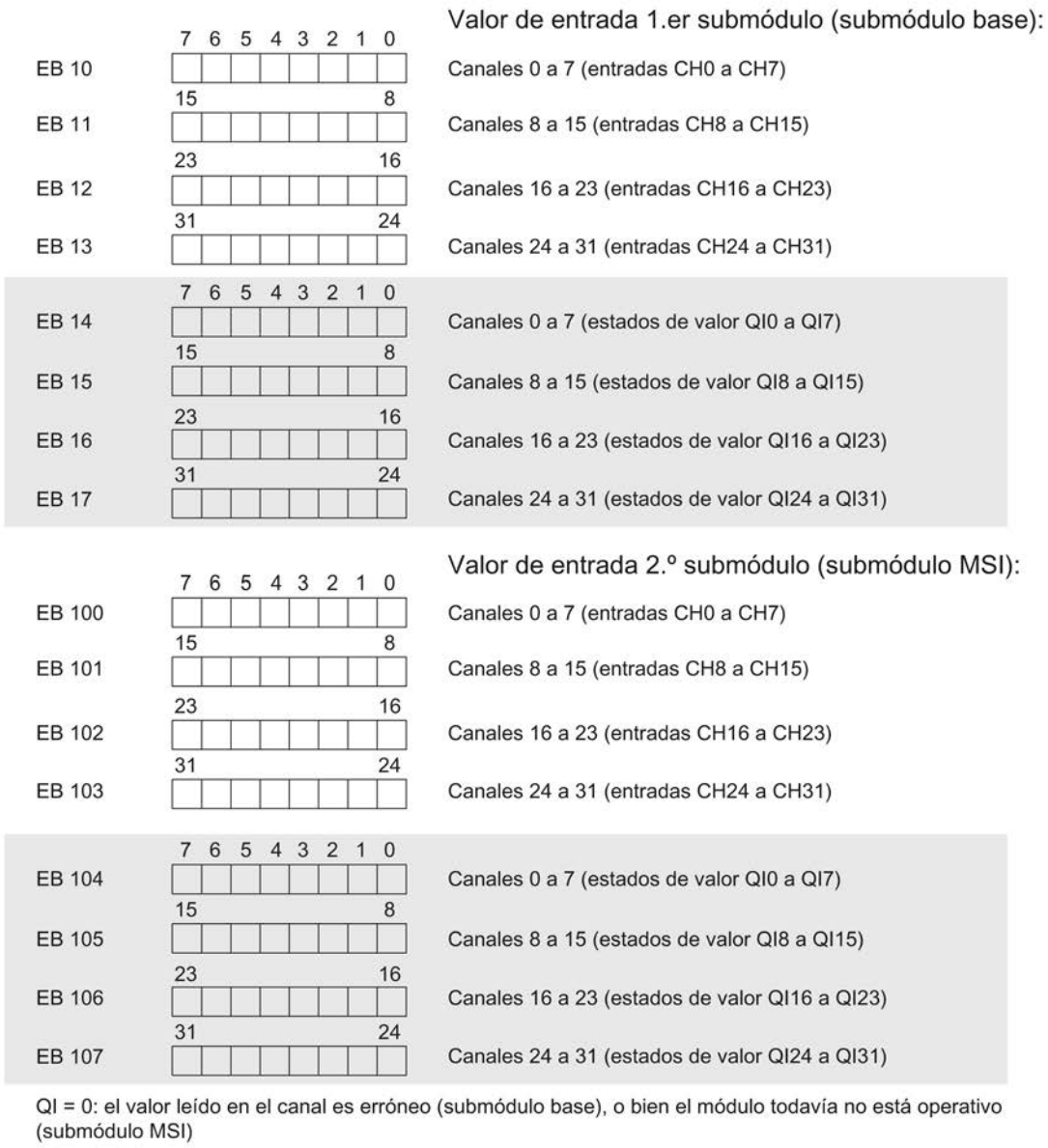
Ejemplo: el módulo DI 32x24VDC HF está configurado como se ha descrito anteriormente (una copia adicional de las entradas). Se obtienen un submódulo base y un submódulo MSI.

En este ejemplo, el submódulo base recibe la dirección inicial 10; el submódulo MSI, la dirección inicial 100. La dirección del submódulo MSI no es relevante para el controlador IO, pues en el transcurso de la configuración del Shared Device este submódulo se asignará a

otro controlador IO y, por lo tanto, desaparecerá del área de direcciones del controlador IO que realiza la parametrización.

La siguiente figura muestra la asignación del área de direcciones con los submódulos 1 y 2 y la información de calidad. Al principio, todos los submódulos están asignados al controlador IO.

Asignación en la memoria imagen de proceso de las entradas (IPE) para los submódulos 1 y 2



QI = 0: el valor leído en el canal es erróneo (submódulo base), o bien el módulo todavía no está operativo (submódulo MSI)

Consulte también

Evaluación del estado de valor (Página 1917)

Principios básicos del procesamiento del programa

Eventos y OBs

Eventos de arranque de OB

Un evento de arranque de OB provoca esta reacción después de aparecer:

- Si el evento procede de una fuente de eventos a la que se ha asignado un OB, se inicia la ejecución de este OB. Esto significa que el evento se pone en la lista de espera que corresponde según su prioridad.
- Si el evento procede de una fuente de eventos a la que no se ha asignado ningún OB, se ejecuta la reacción del sistema predeterminada.

Nota

Algunas fuentes de eventos están disponibles también sin su configuración, p. ej. arranque, extracción/inserción.

La siguiente tabla muestra un resumen de los eventos de arranque de OB junto con los valores posibles para la prioridad de OB, los números de OB posibles, la reacción del sistema predeterminada y la cantidad de OB. Está ordenada por la prioridad predeterminada de los OB. La prioridad 1 es la más baja.

Tipos de fuentes de eventos	Prioridades posibles (prioridad predeterminada)	Números de OB posibles	Reacción del sistema predeterminada	Número de OB soportados
Arranque	1	100, >= 123	Ignorar	100
Programa cíclico	1	1, >= 123	Ignorar	100
Alarma horaria	de 2 a 24 (2)	de 10 a 17, >= 123	no válido	20
Alarma de estado	de 2 a 24 (4)	55	Ignorar	1
Alarma de actualización	de 2 a 24 (4)	56	Ignorar	1
Alarma específica del fabricante o del perfil	de 2 a 24 (4)	57	Ignorar	1
Alarma de retardo	de 2 a 24 (3)	de 20 a 23, >= 123	no válido	20
Alarma cíclica	de 2 a 24 (de 8 a 17, en función del tiempo de ciclo)	de 30 a 38, >= 123	no válido	20
Alarma de proceso	de 2 a 26 (18)	de 40 a 47, >= 123	Ignorar	50
Alarma de sincronismo	de 16 a 26 (21)	de 61 a 64, >= 123	Ignorar	20 (uno por interfaz isócrona)
Alarma de servo MC	de 17 a 31 (25)	91	no válido	1
Alarma de interpolador MC	de 16 a 30 (24)	92	no válido	1
Error de tiempo	22	80	Ignorar	1
Tiempo de vigilancia del ciclo excedido una vez			STOP	
Alarma de diagnóstico	de 2 a 26 (5)	82	Ignorar	1

Tipos de fuentes de eventos	Prioridades posibles (prioridad predeterminada)	Números de OB posibles	Reacción del sistema predeterminada	Número de OB soportados
Extracción/inserción de módulos	de 2 a 26 (6)	83	Ignorar	1
error de rack	de 2 a 26 (6)	86	Ignorar	1
Error de programación (solo con tratamiento de errores global)	de 2 a 26 (7)	121	STOP	1
Error de acceso a la periferia (solo con tratamiento de errores global)	de 2 a 26 (7)	122	Ignorar	1

Asignación entre fuente de eventos y OB

El lugar donde se configura la asignación entre OB y fuente de eventos depende del tipo de OB:

- En alarmas de proceso y alarmas de sincronismo, la asignación se realiza durante la configuración del hardware o al crear el OB.
- En la alarma de MC.Servo y la alarma MC-Interpolator, STEP 7 asigna automáticamente los OB 91/92 en cuanto se agrega un objeto tecnológico "S7-1500 Motion Control".
- En los demás tipos de OB, la asignación se realiza al crear el OB, dado el caso, después de que se haya configurado la fuente de eventos.

Una asignación ya realizada puede modificarse en las alarmas de proceso con las instrucciones ATTACH y DETACH en el tiempo de ejecución. En este caso no se modifica la asignación configurada, sino solo la asignación realmente efectiva. La asignación configurada es efectiva después de cargar y en cada arranque.

Las alarmas de proceso que no tienen asignado ningún OB (porque están así configuradas o después de llamar la instrucción DETACH) se ignoran. La verificación de si un OB está asignado a un evento no tiene lugar cuando aparece el respectivo evento, sino cuando la alarma de proceso debe procesarse realmente.

Prioridad de OB y comportamiento de ejecución

Las CPU S7-1500 soportan desde la prioridad 1 (más baja) hasta la 26 (más alta). Un OB obtiene la prioridad de su evento de arranque.

Los OB se procesan atendiendo meramente a la prioridad. Esto significa que si existen varias solicitudes de OB al mismo tiempo, en primer lugar se procesará aquel OB con la prioridad más alta. Si se produce un evento con una prioridad más alta que la del OB actualmente activo, este OB se cancela. Los eventos con la misma prioridad se procesan en el orden de aparición.

Consulte también

Bloques de organización (OB) (Página 2252)

Principios básicos del tratamiento de errores (Página 2932)

Principios del tratamiento de errores local (Página 2934)

ATTACH: Asignar OB a evento de alarma (Página 4884)

DETACH: Deshacer asignación entre OB y evento de alarma (Página 4885)

Comportamiento de sobrecarga de la CPU

Principio del comportamiento de sobrecarga de la CPU

La ejecución del programa de usuario en las CPUs S7-1500 se basa en eventos de distinta prioridad. Un evento inicia la ejecución de un OB. Los OBs se procesan atendiendo meramente a la prioridad. Esto significa que si existen varias solicitudes de OB al mismo tiempo, en primer lugar se procesará aquel OB con la prioridad más alta. Si se produce un evento con una prioridad más alta que la del OB actualmente activo, este OB se cancela. Los eventos con la misma prioridad se procesan en el orden de aparición.

Para los escenarios de eventos que se contemplan a continuación, los requisitos son que se haya asignado un OB a cada fuente de evento y que estos OBs tengan la misma prioridad. El segundo requisito sirve únicamente para simplificar la representación.

Cuando se produce un evento, se inicia la ejecución del respectivo OB. Esto significa que se pone en la lista de espera que corresponde según su prioridad. Dependiendo de la prioridad de OB y de la carga actual del procesador, generalmente se produce un retardo en la ejecución del OB. De ahí que el mismo evento pueda volverse a presentar una o varias veces antes de que se haya procesado el OB que corresponde al evento anterior. La CPU maneja este tipo de situaciones de la siguiente manera: el sistema operativo clasifica los eventos en la lista de espera que corresponde a su prioridad atendiendo al orden de su aparición.

Se puede limitar el número de eventos pendientes que se encadenan desde una misma fuente con el fin de controlar situaciones de sobrecarga temporales. Una vez se alcanza el número máximo de eventos de arranque pendientes, p. ej. de un OB de alarma cíclica concreto, el siguiente evento se rechaza.

La sobrecarga se produce cuando aparecen más eventos procedentes de una misma fuente de los pueden procesarse.

En los siguientes apartados se ofrece información más detallada.

Recuperación y rechazo de eventos del mismo tipo

El término "eventos del mismo tipo" debe entenderse en lo sucesivo como los eventos procedentes de una fuente, como p. ej., los eventos del arrancador para un determinado OB de alarma cíclica.

Con el parámetro OB "Número de eventos en cola" se determina el número de eventos del mismo tipo que el sistema operativo pone en la correspondiente lista de espera y que, por tanto, corrige. Así, por ejemplo, si este parámetro tiene el valor 1, se guarda un evento en la memoria intermedia.

Nota

Muchas veces se prefiere no corregir eventos cíclicos, ya que esto puede derivar en una sobrecarga en el caso de OBs de igual prioridad o prioridad inferior. Por tanto, normalmente es más conveniente rechazar eventos correspondientes y reaccionar a la situación de sobrecarga en el siguiente procesamiento regular de OB. Un valor bajo del parámetro "Número de eventos en cola" contribuye a no complicar una situación de sobrecarga, sino a atenuarla.

En este caso, por ejemplo, si se alcanza en la lista de espera el número máximo de eventos de arranque para un OB de alarma cíclica, cada nuevo evento de arranque simplemente se cuenta y, a continuación, se rechaza. En el siguiente procesamiento regular de OB se informa del número de los eventos de arranque rechazados en el parámetro de entrada "event_count" (en la información de arranque optimizada). Entonces se puede reaccionar apropiadamente a la situación de sobrecarga. A continuación, la CPU pone a cero el contador de eventos perdidos.

A modo de ejemplo, si la CPU rechaza por primera vez un evento de arranque de un OB de alarma cíclica, su posterior comportamiento depende del parámetro OB "Entrada en el búfer de diagnóstico al desbordarse los eventos": Si la casilla de verificación está activada, la CPU registra una vez el evento DW#16#0002:3507 en el búfer de diagnóstico para la situación de sobrecarga en esta fuente de eventos. Las siguientes entradas del búfer de diagnóstico sobre el evento DW#16#0002:3507 que se refieran a esta fuente de eventos se suprimirán hasta que estén corregidos todos los eventos procedentes de esta fuente.

Mecanismo de valores umbrales para la solicitud del OB de error de tiempo

Con el parámetro OB "Habilitar error de tiempo" ("Enable time error") se determina si para eventos del mismo tipo se debe llamar al OB de error de tiempo cuando existe una determinada sobrecarga.

En caso afirmativo (casilla de verificación activada), con el parámetro OB "Umbral de eventos para error de tiempo" se define a qué número de eventos del mismo tipo incluidos en la lista de espera debe llamarse al OB de error de tiempo. Así, por ejemplo, si este parámetro tiene el valor 1, cuando se produce el segundo evento, el evento DW#16#0002:3502 se registra una vez en el búfer de diagnóstico y se solicita el OB de error de tiempo. Las siguientes entradas del búfer de diagnóstico sobre el evento DW#16#0002:3502 que se refieran a esta fuente de eventos se suprimirán hasta que estén corregidos todos los eventos procedentes de esta fuente.

De esta manera tiene la posibilidad de programar una reacción en caso de sobrecarga mucho antes de alcanzar el límite de eventos del mismo tipo y de rechazar eventos.

Para el parámetro "Umbral de eventos para error de tiempo" rige el siguiente rango de valores:
1 <= "Umbral de eventos para error de tiempo" <= "Número de eventos en cola"

Ejemplo de un evento de alarma de proceso

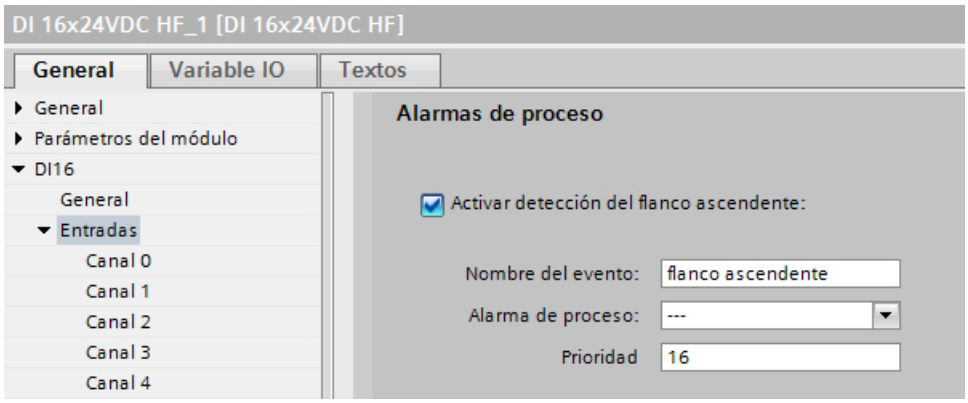
El ejemplo de un módulo que origina una alarma de proceso describe el funcionamiento de la ejecución del programa orientado a eventos en la CPU S7-1500.

Los eventos de proceso son eventos que lanza la periferia (p. ej. una entrada digital) y que provocan la llamada de un OB asignado en la CPU S7-1500. Por ello, los OBs asignados a un evento de proceso se denominan OBs de alarma de proceso.

Procedimiento

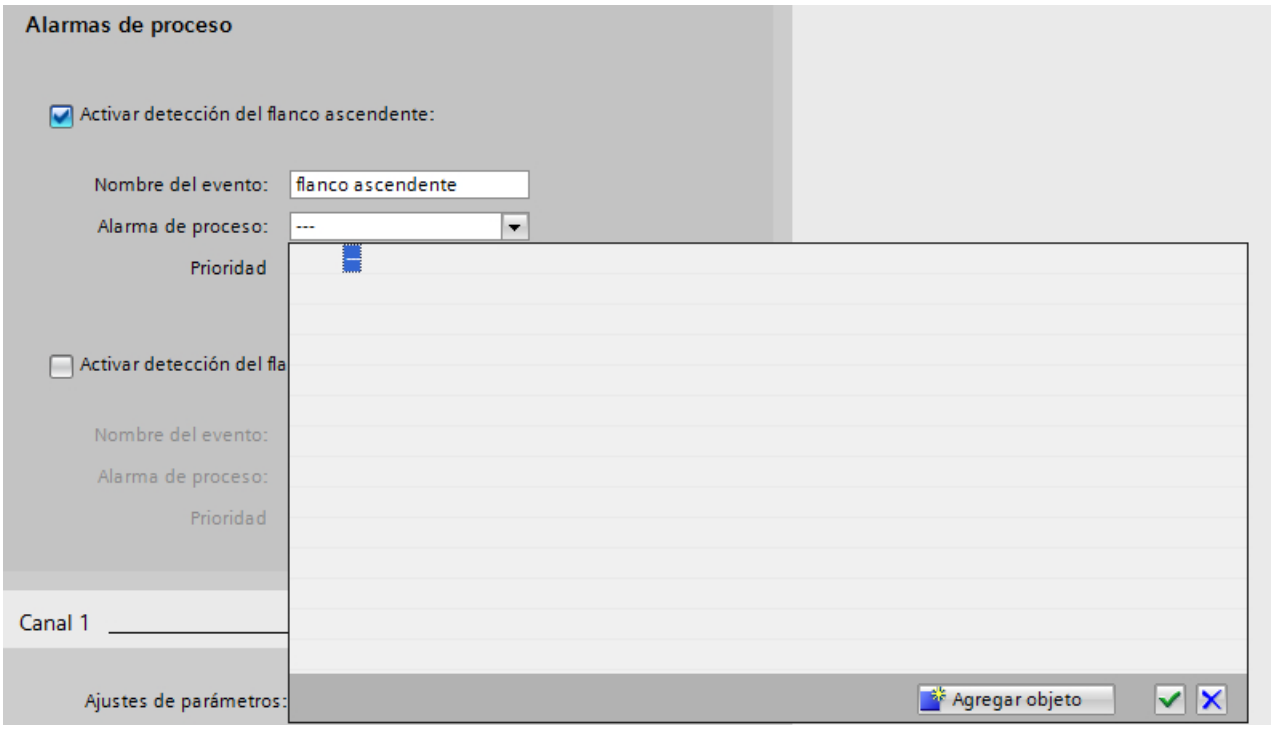
Para configurar una alarma de proceso, proceda del siguiente modo:

1. Configure una S7-1500 con CPU y con un módulo apto para alarmas de proceso, p. ej. DI 16x24VDC HF.
2. Seleccione el módulo apto para alarmas de proceso y desplácese por la navegación local al área "Entradas > Canal 0".
3. Active la opción "Activar detección de flanco ascendente".
4. Introduzca un nombre que describa el evento.
5. En caso necesario, modifique la prioridad del evento.
   En las variables PLC (ficha "Constantes del sistema") se crea para el evento una constante del sistema del tipo de datos Event_HwInt.

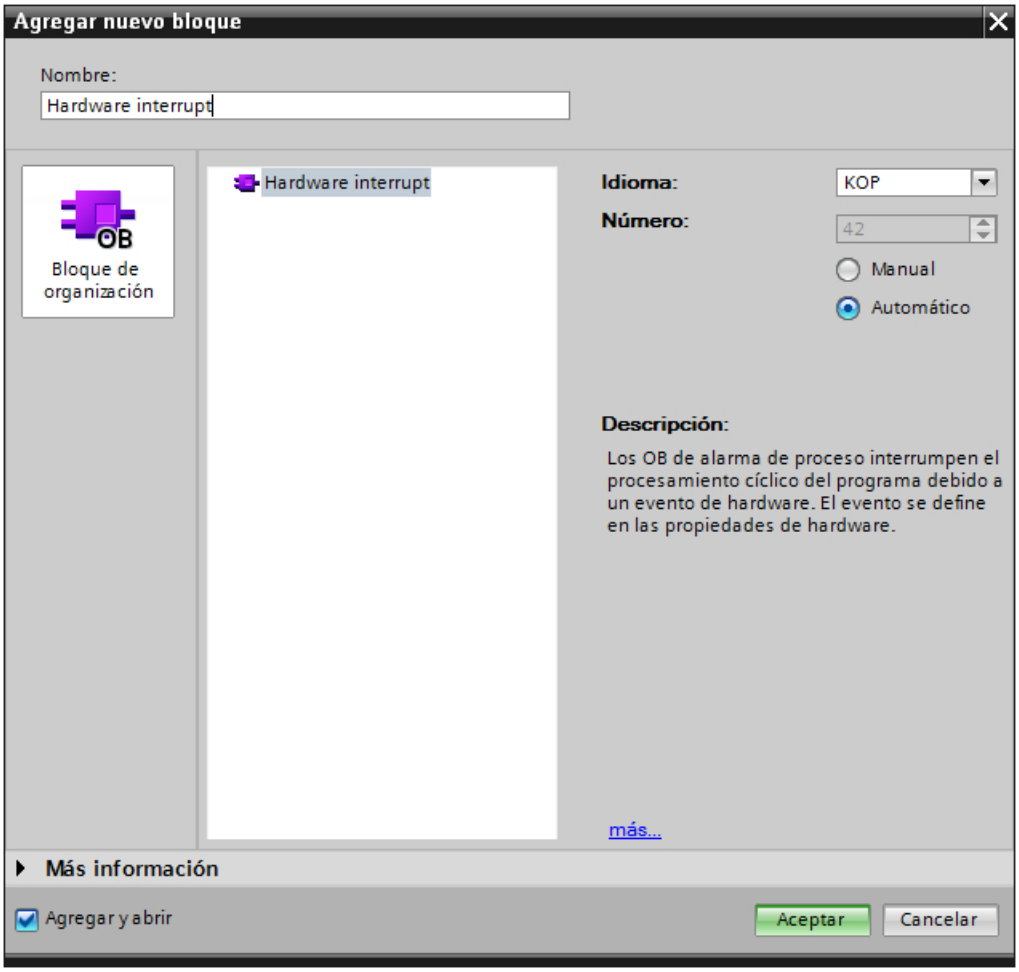


6. Haga clic sobre la lista desplegable "Alarma de proceso" para crear un nuevo OB de alarma de proceso. En caso de que ya exista un OB de alarma de proceso, se puede seleccionar.

7. Haga clic sobre el botón "Agregar objeto" para crear un nuevo OB de alarma de proceso.



8. En el diálogo de selección se puede decidir si el editor de programación debe arrancar inmediatamente con el OB de alarma de proceso abierto (opción "Agregar nuevo y abrir").

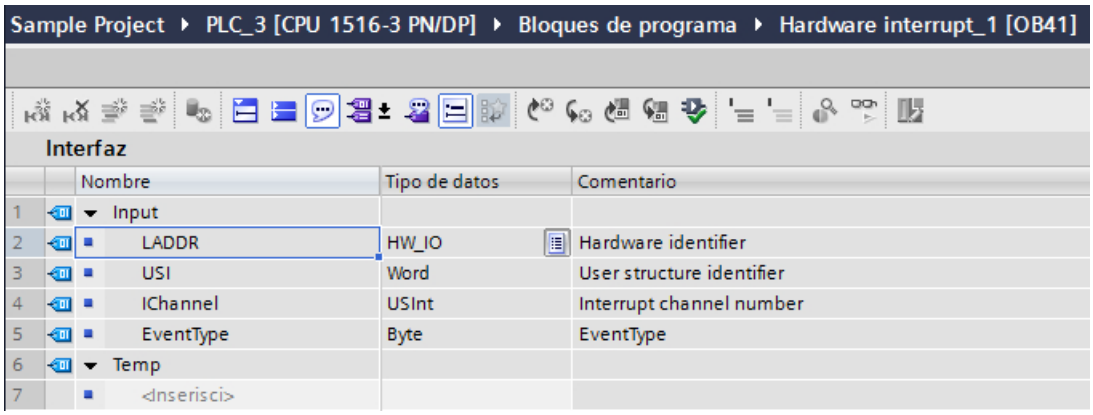


Propiedades de OB de alarma de proceso

En las propiedades del OB de alarma de proceso se puede encontrar, para cada evento que dispara alarma, la información correspondiente sobre el evento de arranque:

Programa de OB de alarma de proceso

En el programa de usuario se puede utilizar la información de arranque del OB de alarma de proceso para identificar y evaluar el evento de alarma de proceso.



Consulte también

Parametrizar OBs de alarma de proceso (Página 1990)

OBs de alarma de proceso (Página 1973)

Información importante sobre identificadores de hardware (Página 917)

Ajustar el comportamiento operativo

Propiedades de la CPU

Resumen de las propiedades de la CPU

Resumen

La tabla siguiente muestra una vista general de las propiedades de la CPU:

Grupo	Propiedades	Descripción
General	Información del proyecto	Información general que describe la CPU insertada. Es posible modificar todos los datos excepto el número de slot.
	Información del catálogo	Únicamente información legible del catálogo de hardware para esta CPU.
	Identificación y mantenimiento	Para guardar información específica de la aplicación, como el nombre de la instalación y el lugar de montaje.
Interfaz PROFINET	General	Nombre y comentario de la interfaz PROFINET. El nombre está limitado a 110 caracteres.

Grupo	Propiedades	Descripción
	Direcciones Ethernet	Selección de si la interfaz PROFINET está conectada en red. Si ya se han creado subredes en el proyecto, estarán disponibles para su selección en la lista desplegable. En caso contrario, es posible crear una subred nueva con el botón "Agregar nueva subred". El protocolo IP incluye datos sobre la dirección IP, la máscara de subred y el uso de un router IP en la subred. Si se utiliza un router IP, es imprescindible indicar la dirección IP del mismo. Consulte: Asignación de direcciones y nombres para dispositivos PROFINET (Página 1491)
	Sincronización horaria	Ajustes de la sincronización horaria en formato de hora NTP. NTP (network time protocol) es un procedimiento general para sincronizar relojes de sistema en redes locales y globales. Con el procedimiento NTP, la interfaz de la CPU envía regularmente consultas de hora (en modo cliente) al servidor NTP de la subred (LAN) cuyas direcciones deben parametrizarse aquí. Sobre la base de las respuestas del servidor, se determina la hora más exacta y fiable y se sincroniza. La ventaja de este procedimiento es la posibilidad de sincronizar la hora más allá de los límites de la subred. La precisión depende de la calidad del servidor NTP utilizado.
	Modo de operación	Selección del modo de operación de la CPU como controlador IO o dispositivo IO, incluyendo los ajustes necesarios en cada caso. En calidad de dispositivo IO (I-Device) debe asignarse un controlador IO y especificar si la interfaz PROFINET se parametrizará por medio del controlador IO de nivel superior o bien por medio del controlador IO local. Asimismo, para la función de dispositivo IO puede activarse el arranque optimizado, siempre que el controlador IO de nivel superior soporte dicha función. Si se ha parametrizado la CPU como dispositivo IO, se muestra la tabla para configurar las áreas de transferencia. Consulte: Configurar un I-device (Página 1550)

Grupo	Propiedades	Descripción
	Opciones avanzadas	Nombre, comentario y otras opciones de ajuste de la interfaz Ethernet y los puertos. Se trata, por ejemplo, de los siguientes ajustes: • Opciones de interfaz, como la reacción del programa de usuario en caso de errores de comunicación, si es posible la sustitución de dispositivos sin medio de almacenamiento extraíble o la puesta en servicio automática, ajustes relativos al modo LLDP conforme a IEEE (PROFINET V2.3) o intervalo de tiempo para el envío de Keep Alives para conexiones. • Ajustes de redundancia de medios, como la función de redundancia de medios (cliente, administrador (auto) o no es dispositivo del anillo de redundancia de medios), visualización de los puertos del anillo o si se generan alarmas de diagnóstico en caso de errores de redundancia. El botón "Ajustes del dominio" enlazado a los ajustes de dominio de la subred conectada (dominio MRP y dominio Sync). • Ajustes en tiempo real, como tiempo de ciclo de emisión y propiedades de sincronización, cada uno con un botón "Ajustes del dominio" enlazado con los ajustes de dominio de la subred conectada (dominio MRP y dominio Sync). En el dominio Sync se ajusta el tiempo de ciclo de emisión para sistemas IO isócronos y se determinan las clases RT (RT e IRT) así como las funciones de sincronización para IRT. • Puertos: para los puertos se ajusta la interconexión con dispositivos vecinos (alternativamente: desde la vista topológica), si fuera necesario. Se requieren interconexiones de puerto para aplicaciones isócronas (IRT) así como para la puesta en servicio automática (opción "Sustitución de dispositivos sin medio de almacenamiento extraíble" activada). Además, se ajustan opciones de puerto, como p. ej. si el puerto se utiliza (los puertos no utilizados pueden bloquearse). También se incluyen en las opciones de puerto los ajustes de la conexión (velocidad de transferencia/dúplex) con las opciones de vigilancia y autonegotiation. Asimismo, en los puertos se ajustan límites para el fin del registro de dispositivos accesibles, el fin de la detección de topología y el fin del dominio Sync. Para cada puerto se visualizan los identificadores de hardware. Consulte: Ajustar tiempo de ciclo de emisión (Página 1507) Interconectar puertos (Página 1506) Definición de opciones de puerto (Página 1510) Información importante sobre la redundancia de medios (Página 1567) Introducción: Isochronous Realtime Ethernet (Página 1526) Activar la sustitución de dispositivo sin medio de almacenamiento extraíble (Página 1513)
	Identificador de hardware	Se muestra el identificador de hardware de la interfaz. Consulte: Información importante sobre identificadores de hardware (Página 917)

Grupo	Propiedades	Descripción
	Acceso al servidor web	Permite el acceso a la función del servidor web.
Interfaz PROFIBUS	General	Nombre y comentario de la interfaz PROFIBUS. El nombre está limitado a 110 caracteres.
	Dirección PROFIBUS	Selección de si la interfaz PROFIBUS está conectada en red. Si ya se han creado subredes en el proyecto, estarán disponibles para su selección en la lista desplegable. En caso contrario, es posible crear una subred nueva con el botón "Agregar nueva subred". Se asigna la dirección PROFIBUS y se muestra la dirección más alta y la velocidad de transferencia.
	Modo de operación	Selección del modo de operación de la CPU como maestro DP o esclavo DP. Se puede asignar un maestro DP a un esclavo DP y parametrizar aquí la comunicación de esclavo I.
	Reloj	La hora puede sincronizarse. Se puede especificar la fuente horaria y la frecuencia de la sincronización.
	SYNC/FREEZE	El maestro DP puede enviar los comandos de control SYNC y/o FREEZE a un grupo de esclavos DP para sincronizar los esclavos DP. Para ello es necesario asignar los esclavos DP a los grupos SYNC/FREEZE.
	Identificador de hardware	Se muestra el identificador de hardware de la interfaz.
Arranque	Arranque tras conexión (POWER ON)	Ajuste del comportamiento de arranque tras una transición de desconexión (POWER OFF)/conexión (POWER ON). Consulte: Principios básicos del estado operativo "ARRANQUE" (Página 1902)

Grupo	Propiedades	Descripción
	Comparación entre configuración teórica y real	Define el comportamiento de arranque de la CPU en caso de que la configuración real de la estación S7-1500 no se corresponda con la configuración teórica: • Arranque de la CPU solo en caso de compatibilidad • Arranque de la CPU también si hay diferencias Con el ajuste "Arranque solo en caso de compatibilidad" debe haber un módulo en un slot configurado que sea compatible con el módulo configurado. Compatible significa que el módulo existente concuerde en número de entradas y salidas así como en las propiedades eléctricas y funcionales. Un módulo compatible debe poder reemplazar por completo a un módulo configurado; puede hacer más cosas, pero no menos. Ejemplo del ajuste "Arranque de la CPU solo en caso de compatibilidad": Una CPU puede ser un sustituto compatible de una CPU del mismo tipo con una versión de firmware inferior. Un módulo de entrada con 32 entradas digitales puede ser un sustituto compatible de un módulo de señales con 16 entradas digitales. La CPU arranca si está enchufado el módulo configurado o un módulo compatible. Si hay insertado un módulo no compatible, la CPU no arranca. Ejemplo del ajuste "Arranque de la CPU también si hay diferencias": En lugar de un módulo de entradas digitales configurado se inserta un módulo de salidas digitales o bien el slot se deja vacío. Aunque las entradas configuradas no están accesibles, la CPU arranca. Tenga en cuenta que en este caso el programa de usuario no funcionará correctamente, por lo que deben tomarse las medidas adecuadas. Nota: para módulos S7-1500 es posible ajustar individualmente para cada slot la comparación entre módulo teórico y real. El ajuste predeterminado es "De CPU" (es decir, para el slot rige el ajuste de la CPU). Para módulos de periferia de otras familias rige por lo general el ajuste de la CPU (no puede modificarse).
	Tiempo de parametrización para la periferia centralizada y descentralizada	Determina el intervalo de tiempo máximo (ajuste predeterminado: 60000 ms) en el que la periferia centralizada y descentralizada debe estar operativa. La CPU suministra a los módulos CM y CP tensión y parámetros de comunicación durante el arranque. El tiempo de parametrización limita el periodo durante el cual los módulos E/S conectados a CM o CP deben estar operativos. Si la periferia centralizada y descentralizada no está operativa dentro del tiempo de parametrización, el comportamiento de arranque de la CPU depende del ajuste del parámetro "Compatibilidad hardware soportada".
Hora	Hora local y horario de verano	Selección de la zona horaria en la que se utiliza la CPU y ajuste del horario de verano y de invierno.

Grupo	Propiedades	Descripción
Reloj	-	Definición para la sincronización de la hora a través del bus de fondo. Las posibilidades de ajuste dependen de la CPU. • Como esclavo: el reloj es sincronizado por otro reloj. • Como maestro: el reloj sincroniza otros relojes. • Ninguna: no se realiza ninguna sincronización. Los intervalos de tiempo para la sincronización pueden seleccionarse. Consulte: Auto-Hotspot
Protección	Protección y contraseña del acceso de lectura/escritura	Ajuste de la protección contra lectura o escritura y de la contraseña de acceso a la CPU. Consulte: Posibilidades de ajuste de la protección (Página 1955) Consulte: Parametrización de niveles de acceso (Página 1957)
Marcas de sistema y ciclo	Bits de marcas de sistema y bits de marcas de ciclo	Ajuste de un byte para las funciones de marcas de sistema y ajuste de un byte para las funciones de marcas de ciclo (para ello, cada bit se conecta y desconecta a una frecuencia predeterminado). Las marcas de sistema se utilizan en las consultas siguientes: • ¿El ciclo actual es el primer ciclo después de desconexión (POWER OFF)/conexión (POWER ON)? • ¿Han cambiado los estados de diagnóstico respecto del ciclo anterior? • Consulta en "1" (high) • Consulta en "0" (low) Las marcas de ciclo cambian periódicamente de valor, en intervalos definidos. Consulte: Uso de marcas de sistema (Página 1954) Consulte: Uso de marcas de ciclo (Página 1955)
Ciclo	Tiempo de ciclo máximo y Tiempo de ciclo mínimo	Definición de un tiempo de ciclo máximo y de un tiempo de ciclo mínimo fijo. Si el tiempo de ciclo excede el tiempo de vigilancia del ciclo, la CPU pasa al estado operativo STOP. Causas de un rebase de tiempo: • Procesos de comunicación • Acumulación de eventos de alarma • Error en el programa de la CPU. Si el tiempo de ciclo es más corto que el tiempo de ciclo mínimo introducido, la CPU espera hasta alcanzar el tiempo de ciclo mínimo. En este tiempo adicional de procesamiento del programa, la CPU procesa el OB 90 (OB de fondo) si está cargado. Consulte: Tiempo de ciclo y tiempo de vigilancia del ciclo (Página 1940)
Carga de comunicación	Carga de ciclo por comunicación	Controla hasta cierto punto la duración de los procesos de comunicación que también prolongan el tiempo de ciclo. Los procesos de comunicación pueden ser, p. ej.: transferencia de datos a otra CPU o carga de bloques (lanzada vía PC). Consulte: Carga de ciclo por comunicación (Página 1941)

Grupo	Propiedades	Descripción
Diagnóstico del sistema	General	El diagnóstico del sistema es la captación, valoración y notificación de un fallo dentro del sistema de automatización. Ejemplos de errores: • Fallo en el programa de la CPU • Fallo de módulos • Rotura de cables de sensores y actuadores Para determinar el diagnóstico del sistema existen textos de aviso predefinidos. Estos textos de aviso pueden modificarse según sea necesario e incluso añadirse nuevos. Los ajustes se guardan con el proyecto y no repercuten en los componentes implicados hasta compilar y cargar la configuración del hardware. Para el S7-1500, el diagnóstico de sistema está activado automáticamente (no puede desactivarse).
	Ajustes de avisos	Para cada una de las categorías de avisos, como error, mantenimiento solicitado, mantenimiento necesario e información, se especifica si deben visualizarse los avisos de esta categoría y si tienen que acusarse.
Sinóptico de direcciones	-	Representación en forma de tabla de todas las direcciones utilizadas en la CPU para las entradas y salidas integradas así como para los módulos insertados. Las direcciones que no utiliza ningún módulo se representan como hueco. Es posible filtrar la vista por • Direcciones de entrada • Direcciones de salida • Huecos direcciones
Recursos de conexión	-	Ofrece una vista general de los recursos reservados y asignados para las conexiones de la CPU. En la vista online se muestran también los recursos actualmente en uso. Consulte: Consumo de recursos de conexión (Página 595)
Display	General	El display permanece oscuro en el modo de espera y vuelve a activarse en cuanto se pulsa una tecla del display. El modo de espera también puede modificarse en el menú del display. En el modo de ahorro de energía, la pantalla muestra la información con una reducción del brillo. En cuanto se pulsa una tecla del display, el modo de ahorro de energía finaliza. El modo de ahorro de energía también puede modificarse en el menú del display. Idioma del display: el idioma cambia directamente después de cargar la configuración hardware con el idioma estándar ajustado. El idioma también puede modificarse en el menú del display.
	Actualización automática	Entrada del intervalo de tiempo en el que se actualiza el contenido del display.
	Contraseña	Posibilidad de introducir una contraseña para protegerse de accesos no autorizados. Puede ajustarse el tiempo que transcurrirá hasta la desconexión automática tras la introducir la contraseña en el display.
	Tablas de observación	Las tablas de observación configuradas pueden seleccionarse. Durante el funcionamiento pueden utilizarse las tablas de observación seleccionadas en el display.

Grupo	Propiedades	Descripción
	Logotipo personalizado	Puede seleccionarse un logotipo personalizado y cargarlo en la CPU con la configuración hardware.
Servidor web	-	Activa y configura la función del servidor web. Consulte: Activar el servidor web (Página 942)
	Actualización automática	Envía periódicamente la página web solicitada con los datos actuales de la CPU al navegador web. La duración de los períodos se introduce en "Intervalo de actualización". La actualización automática solo se puede activar cuando el servidor web está activado. Consulte: Activar la actualización automática (Página 942)
	Administración de usuarios	Aquí se introducen nombres de usuario con sus derechos de acceso. • Cualquiera: usuario predefinido, cuyos derechos de acceso se asignan desde una lista desplegable. El usuario "Cualquiera" no necesita iniciar la sesión introduciendo una contraseña en un acceso web. • User#: nombre editable de un usuario al que se asignan derechos de acceso y la contraseña correspondiente. Cuando se hace clic en una fila debajo de "Cualquiera", se muestra un nombre predeterminado, que se puede cambiar. Se pueden asignar distintos derechos de acceso a cualquier usuario a través de una lista desplegable. Hay derechos de acceso que están acoplados entre sí. Ejemplo: Si activa el derecho de acceso "Actualizar firmware", se activarán automáticamente los derechos de acceso "Modificar estado operativo" y "...consultar diagnóstico". Estos derechos son necesarios para poder ejecutar una actualización de firmware desde el servidor web. Consulte: Lista de usuarios (Página 944)
	Páginas web definidas por el usuario	Permite acceder a páginas web de libre configuración de la CPU con un navegador web. Consulte: Crear y cargar páginas web definidas por el usuario (Página 946)
	Resumen de interfaces	Representación en forma de tabla de todos los módulos con sus interfaces Ethernet para la funcionalidad de servidor web de este dispositivo. Aquí se permite o no el acceso al servidor web desde la interfaz correspondiente para cada interfaz Ethernet del dispositivo (CPU, CP o CM).
Idiomas de la interfaz	-	El servidor web y la pantalla de la CPU soportan diferentes idiomas de interfaz. Para visualizar los textos del proyecto es posible asignar un idioma del proyecto a cada idioma del servidor web (se soportan 2 como máximo). Aquí se especifican los idiomas seleccionables al acceder a la web y en la pantalla de la CPU. Se asigna un idioma de proyecto para el servidor web, p. ej., en la lista desplegable se selecciona el idioma de proyecto "Alemán (Luxemburgo)" para el idioma alemán del servidor web. Para poder seleccionar el idioma del proyecto en la lista desplegable, este debe estar activado en "Idiomas del proyecto" en el árbol del proyecto.

Grupo	Propiedades	Descripción
Control de la configuración	-	Activa el control de configuración. Ello permite cambiar la configuración en el programa de usuario hasta cierto punto. Consulte: Control de la configuración
Fuente de alimentación	Fuente de alimentación	Indica que la CPU está conectada con la fuente de alimentación. Por tanto, esta potencia a la CPU repercute positivamente en el equilibrio de potencia.
	Equilibrio de potencia	En el equilibrio de potencia se compara la potencia proporcionada por la CPU y los módulos de alimentación con la necesidad de potencia de los módulos de señales. En el caso de un equilibrio de potencia negativo, que significa que existe una necesidad de potencia mayor que la potencia que pueden proporcionar la CPU y los módulos de alimentación, se hace saber en la tabla.

Consulte también

Ajustar el idioma para web (Página 943)

Ajustar el arranque priorizado (Página 1566)

Asignar el nombre de dispositivo y la dirección IP (Página 1493)

Configuración de la CPU en el display (Página 1939)

Reglas para CPU con varias interfaces PROFINET (Página 1942)

LLDP conforme a IEEE (Página 1944)

Configuración de la CPU en el display

Las CPUs S7-1500 presentan un display con teclas de manejo. El display puede utilizarse para consultar información de estado relativa a los módulos configurados, para fines de diagnóstico y también para adaptar algunos ajustes de la configuración de la CPU.

Ejemplo: ajustes de la dirección IP y del nombre de dispositivo PROFINET en el display

Si desea asignar los parámetros de la dirección IP (dirección IP, máscara de subred así como ajustes del router) en el display, en el proyecto correspondiente para la interfaz PROFINET de la CPU debe seleccionar la siguiente opción:

- Ajustar dirección IP en el dispositivo

La misma pauta se aplica a los nombres de dispositivos PROFINET: si desea asignar los nombres de dispositivos PROFINET en el display, en el proyecto correspondiente para la interfaz PROFINET de la CPU debe seleccionar la siguiente opción:

- Ajustar el nombre de dispositivo PROFINET en el dispositivo

Tiempo de ciclo y tiempo de vigilancia del ciclo

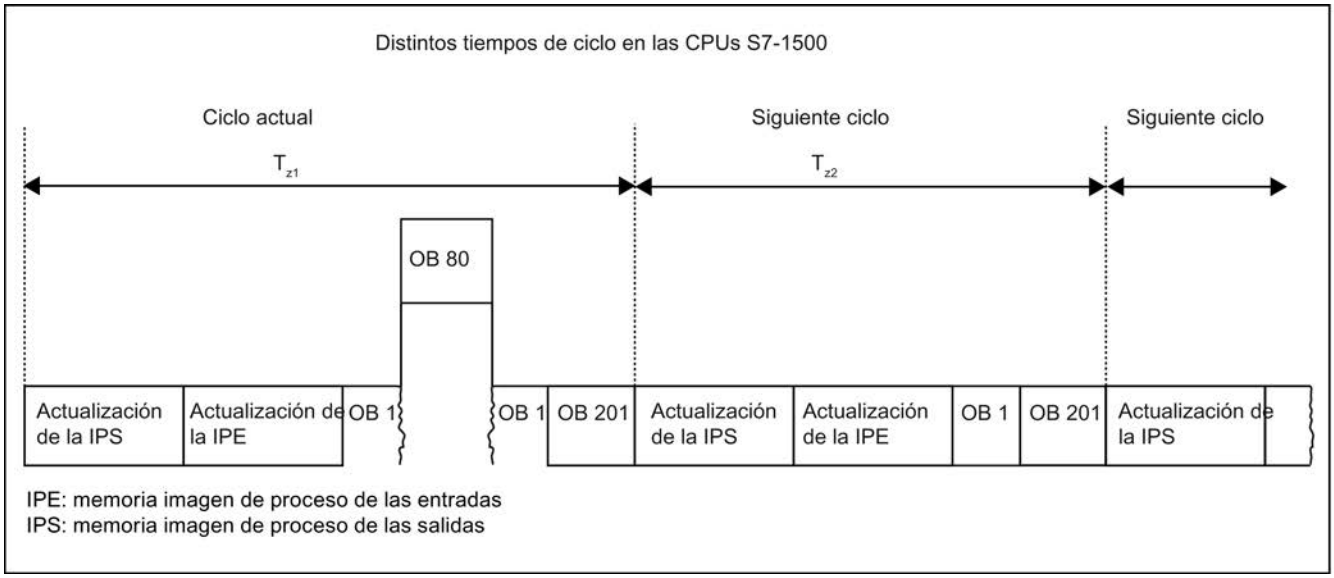
Función

El tiempo de ciclo es el tiempo que necesita el sistema operativo para el procesamiento cíclico del programa y para todas las partes del programa que interrumpen dicho ciclo. El procesamiento del programa se interrumpe debido a:

- Errores de tiempo y errores 2xMaxCycleTime
- Actividades del sistema, p. ej. actualización de la memoria imagen de proceso

Por consiguiente, el tiempo de ciclo (TC) no es igual en todos los ciclos.

La figura siguiente muestra un ejemplo de tiempos de ciclo diferentes (TC1 ≠ TC2) para CPUs S7-1500:



En el ciclo actual, el OB cíclico utilizado (p. ej. el OB 1) es interrumpido por un error de tiempo (p. ej. el OB 80). A continuación del OB cíclico se procesa el siguiente OB del ciclo, el 201.

Tiempo de vigilancia del ciclo

El sistema operativo vigila si el tiempo de ejecución del programa cíclico alcanza un límite superior configurable, el tiempo de ciclo máximo. Dicho tiempo de vigilancia se reinicia llamando la instrucción RE_TRIGR en cualquier punto del programa.

Si el programa cíclico excede el tiempo de vigilancia del ciclo, el sistema operativo intenta arrancar el OB de error de tiempo (OB 80). Si el OB no está disponible, la CPU ignora que se ha excedido el tiempo de vigilancia del ciclo.

Además de vigilar el tiempo de ejecución para asegurar que no se excede el tiempo de vigilancia del ciclo, también se garantiza el mantenimiento de un tiempo de ciclo mínimo. Para ello, el sistema operativo retarda el inicio de un ciclo nuevo hasta que se ha alcanzado el tiempo de ciclo mínimo. Durante este tiempo de espera se procesan eventos nuevos y servicios del sistema operativo.

Si el tiempo de vigilancia del ciclo se excede por segunda vez, p. ej., mientras se procesa el OB de error de tiempo (error 2xMaxCycleTime), la CPU pasa al estado operativo STOP.

Carga de ciclo por comunicación

Función

El tiempo de ciclo de la CPU se prolonga debido a procesos de comunicación. Dichos procesos de comunicación incluyen, p. ej.:

- Transferencia de datos a otra CPU
- Carga de bloques lanzada por una programadora

La duración de estos procesos de comunicación se controla hasta cierto punto con el parámetro de la CPU "Carga de ciclo por comunicación".

Además de los procesos de comunicación, las funciones de test también prolongan el tiempo de ciclo. En este caso, el parámetro "Carga de ciclo por comunicación" influye en la duración.

Funcionamiento del parámetro

En el parámetro "Carga de ciclo por comunicación" se indica el porcentaje de capacidad de procesamiento de la CPU que debe estar disponible para los procesos de comunicación.

Si la comunicación no requiere esta capacidad de procesamiento, esta queda a disposición del procesamiento del programa.

Para las clases de prioridad **inferiores o iguales a 15**, la CPU asigna continuamente dicha capacidad de procesamiento a la comunicación.

Las clases de prioridad **superiores a 15** no se interrumpen debido a la comunicación.

Esto significa que afectan a la capacidad de interrupción del OB y de los bloques que son llamados por el OB como consecuencia de la asignación de la prioridad de un evento o de un OB. Si de este modo se hace que amplias secciones del programa no puedan interrumpirse para minimizar los tiempos de ciclo, se retardan las funciones online de STEP 7.

Repercusión en el tiempo de ciclo real

El uso del parámetro "Carga de ciclo por comunicación" prolonga el tiempo de ciclo del bloque de organización cíclico (p. ej. OB 1) en un factor que se calcula según la fórmula siguiente:

$$\frac{100}{100 - \text{"Carga de ciclo por comunicación"}}$$

La fórmula no tiene en cuenta la influencia de eventos asíncronos, como alarmas de proceso o cíclicas, en el tiempo de ciclo.

Si el tiempo de ciclo se prolonga debido a procesos de comunicación, también es posible que se produzcan más eventos asíncronos dentro del tiempo de ciclo del bloque de organización cíclico. Esto todavía prolonga más el ciclo. La prolongación depende de cuántos eventos aparezcan y de cuánto dure su procesamiento.

Ejemplo 1 - no hay eventos asíncronos adicionales:

Si se ajusta el parámetro "Carga de ciclo por comunicación" al 50 %, el tiempo de ciclo del bloque de organización cíclico puede duplicarse.

Ejemplo 2 - hay eventos asíncronos adicionales:

Con un tiempo de ciclo puro de 500 ms, el tiempo de ciclo real puede ascender hasta 1000 ms debido a una carga de comunicación del 50 %, siempre que la CPU tenga suficientes peticiones de comunicación para procesar.

Si, paralelamente, se procesa una alarma cíclica cada 100 ms (clase de prioridad inferior o igual a 15) con un tiempo de procesamiento de 20 ms, dicha alarma sin carga de comunicación prolongaría el ciclo en un total de 5*20 ms = 100 ms, es decir, el tiempo de ciclo real sería de 600 ms.

Dado que la comunicación al 50 % de su carga prolonga tanto el programa cíclico como también la alarma cíclica, el tiempo de ciclo real no sería de 1000 ms, sino de 1200 ms.

Si la alarma cíclica tiene una prioridad superior a 15, no se prolonga el tiempo de procesamiento del OB de alarma cíclica por la comunicación, pero el tiempo de ciclo real se mantiene igual (1200 ms).

Nota

Tenga en cuenta las indicaciones siguientes:

- Verifique las repercusiones de un cambio de valor del parámetro "Carga de ciclo por comunicación" en el funcionamiento de la instalación.
- La carga por comunicación debe tenerse en cuenta al ajustar el tiempo de ciclo mínimo, puesto que de lo contrario se pueden producir errores de tiempo.

Recomendaciones

- Aumente el valor solo si la CPU se utiliza principalmente para fines de comunicación y el programa de usuario es de duración no crítica.
- En todos los demás casos, el valor solo debería reducirse.
- Si los OBs de alarma no pueden interrumpirse debido a la comunicación, ajuste su prioridad a un valor superior a 15.

CPU con varias interfaces PROFINET

Reglas para CPU con varias interfaces PROFINET

Dentro de la familia de las CPU S7-1500 existen CPU con dos o tres interfaces PROFINET, p. ej. CPU 1516-3 PN/DP.

Una interfaz PROFINET puede utilizarse para la comunicación PROFINET IO, mientras que las otras se utilizan, p. ej., para la comunicación con una red de orden superior (Backbone, Router, Internet) o para la comunicación con otra máquina o celda de automatización.

Tenga en cuenta que algunos ajustes solo pueden modificarse en la interfaz X1 y leerse en la otra interfaz (p. ej. X2).

Reglas

Para una CPU S7-1500 con varias interfaces PROFINET rige lo siguiente:

- Solo la primera interfaz PROFINET (X1) soporta PROFINET IO e Isochronous Realtime (IRT). Solo en esta interfaz puede efectuarse la configuración en tiempo real.
- La segunda y todas las demás interfaces PROFINET (p. ej. X2) soportan la funcionalidad básica de PROFINET (ninguna función de controlador IO/dispositivo IO).
  - La segunda y todas las demás interfaces pueden enviar topología por medio del protocolo LLDP, gestionar MIB de LLDP, responder y gestionar peticiones SNMP, por ejemplo, y soportan la función "Dispositivos accesibles" (protocolo DCP), etc.
- Las subredes IP de las dos interfaces deben ser diferentes. Es decir, la dirección IP de ambas interfaces debe diferir en el componente de subred. La comunicación a través de varias interfaces PROFINET solo resulta ventajosa si las interfaces se encuentran en distintas subredes IP.
  Ejemplo:
  - Máscara de subred de ambas interfaces: 255.255.255.0
  - Dirección IP de la primera interfaz: 192.168.0.10
  - Dirección IP de la segunda interfaz: 192.168.1.10
- Las interfaces pueden estar conectadas a la misma subred Ethernet.
- Como máximo puede estar activo un router. La dirección IP del router debe encontrarse en la subred IP de la interfaz PROFINET en que está configurado el router.
- Cuando el nombre de dispositivo se asigna automáticamente, además del nombre del dispositivo (CPU) se utiliza el nombre de la interfaz. Ejemplo: plc_1.profinet interface_1
- Sincronización horaria: la activación de la sincronización y los ajustes para el servidor NTP solo son posibles en la interfaz PROFINET X1. Los ajustes se aplican a la interfaz X2 y a las demás interfaces (allí son de solo lectura).
- Opción Keep Alive: para la activación de la opción avanzada "Enviar Keep Alives para conexiones ..." se aplican las mismas pautas que para la sincronización horaria: posibilidad de activar solo en la interfaz X1 y de leer solo en la interfaz X2 y demás interfaces.

Establecer una conexión online

En una CPU con varias interfaces PROFINET no basta con que una programadora que debe conectarse online con la CPU y las interfaces PROFINET se encuentren en la misma subred S7. También deben tenerse en cuenta las direcciones de subred IP de la programadora y las interfaces PROFINET.

Si desea acceder online con PG/PC equipado con STEP 7 a una CPU con dos o tres interfaces PROFINET, ejecute en segundo plano las siguientes actividades:

- al establecer la conexión online, STEP 7 comprueba si una de las interfaces PROFINET conectadas con la subred S7 se encuentra en la misma subred IP que la propia dirección IP (= dirección IP del acceso online de PG/PC) y pasa al estado online a través de esta dirección IP / interfaz PROFINET.
- En caso de que ninguna de las interfaces PROFINET se encuentre en la misma subred IP, STEP 7 pasa a online a través de las interfaces PROFINET para las que está configurado un router.
- Si no se cumplen los requisitos, STEP 7 emite un aviso con la opción de cambiar la dirección de subred IP de la interfaz PG/PC para que la programadora o el PC se encuentren en la misma subred IP.

LLDP conforme a IEEE

LLDP son las siglas del protocolo no propietario "Link Layer Discovery Protocol", que se define en la norma IEEE-802.1AB.

Con LLDP, los dispositivos Ethernet envían información propia en intervalos regulares a los dispositivos vecinos. Los dispositivos vecinos guardan la información.

Las herramientas SNMP leen estos datos y elaboran la topología de la red a partir de los mismos (SNMP son las siglas en inglés de "Simple Network Management Protocol").

LLDP e IEC 61158

La norma de bus de campo IEC 61158 V2.2, abreviada "IEC V2.2", traslada LLDP a dispositivos con una interfaz PROFINET. De este modo, las herramientas SNMP detectan la topología de redes Ethernet que tienen conectados dispositivos con una interfaz PROFINET.

Los dispositivos con varias interfaces PROFINET requieren una versión más reciente de esta norma, la IEC V2.3. De este modo, las herramientas de Siemens y de otros fabricantes detectan todas las interfaces de un dispositivo con varias interfaces PROFINET.

Todas las interfaces PROFINET conectadas en red deben estar ajustadas al mismo modo (ya sea IEC V2.3 o IEC V2.2).

Si todos los dispositivos de la subred PROFINET se configuran en un mismo proyecto, STEP 7 ajusta automáticamente el modo correcto y no es necesario ocuparse de más ajustes. A continuación se explican las reglas en función de las cuales STEP 7 lleva a cabo los ajustes de modo automático y se indica en qué casos debe ajustarse el modo manualmente.

¿Cómo se sabe qué modo soporta una interfaz PROFINET?

Para saber qué modo soporta una interfaz PROFINET, consulte las propiedades de la interfaz en la vista de dispositivos (con el dispositivo seleccionado) o en la vista de redes (con la interfaz seleccionada). En el apartado "Interfaz PROFINET > Opciones avanzadas" aparece la opción "Utilizar modo IEC V2.2 LLDP".

- Si la opción "Utilizar modo IEC V2.2 LLDP" está activada y no puede modificarse, la interfaz PROFINET soporta únicamente el modo V2.2.
- Si la opción "Utilizar modo IEC V2.2 LLDP" está desactivada y puede modificarse, la interfaz PROFINET soporta el modo V2.2 y también el V2.3.

Ajuste automático de las interfaces PROFINET

STEP 7 ajusta el modo de las interfaces PROFINET de acuerdo con las reglas siguientes:

- Si los dispositivos soportan el modo IEC V2.3, STEP 7 ajusta de forma unitaria al modo IEC V2.3 todas las interfaces PROFINET conectadas a la subred.
- Si un dispositivo solo soporta el modo IEC V2.2, STEP 7 ajustará al modo IEC V2.2 todas las interfaces PROFINET conectadas a la subred.

Ejemplo 1: Ajuste automático

En este ejemplo se creó un I-device en el proyecto 1 y, seguidamente, se insertó en el proyecto 2 como archivo GSD (dos proyectos).

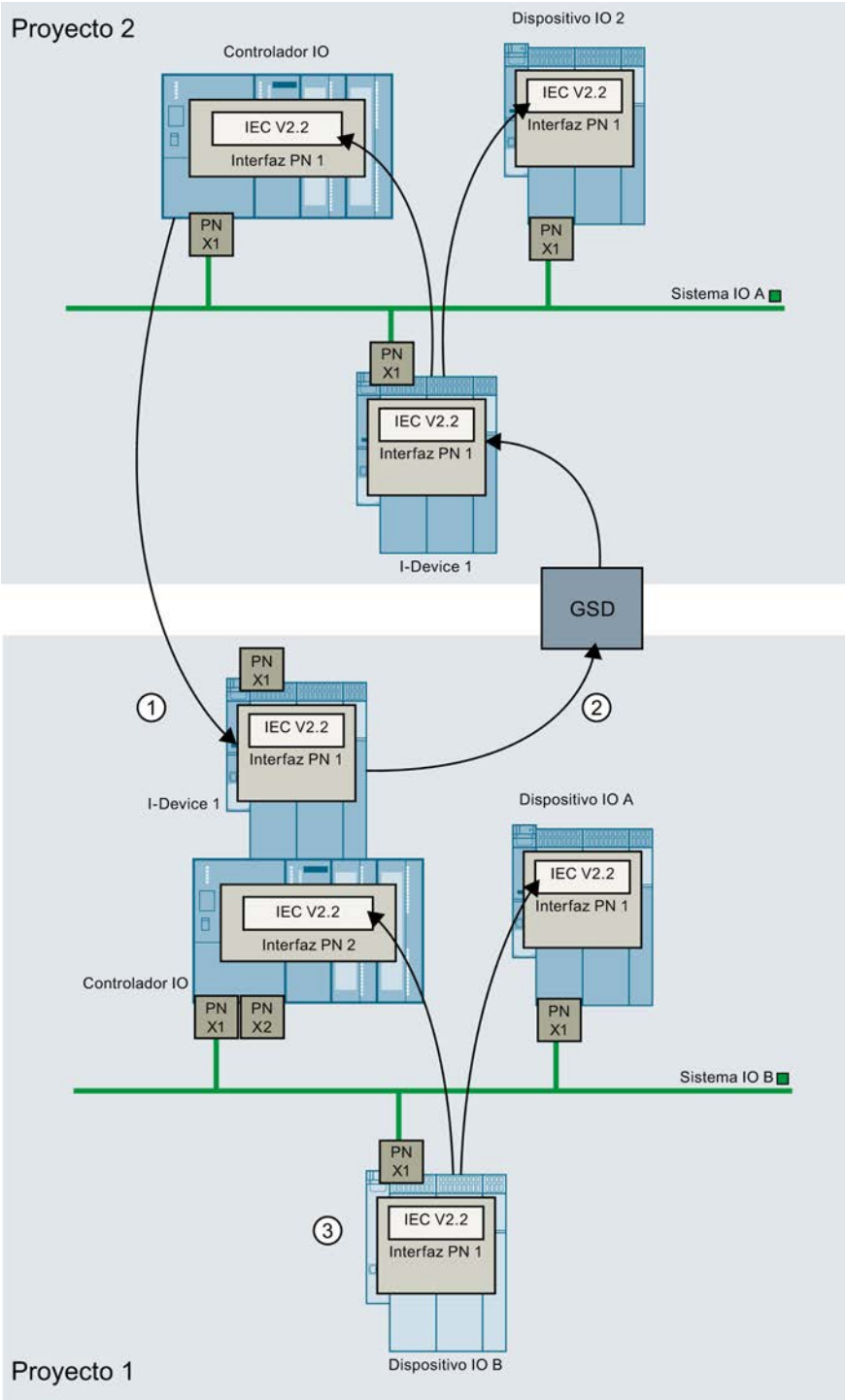
En el proyecto 1 hay un dispositivo (3) que solo soporta el modo V2.2. Por este motivo, STEP 7 ajustó todas las interfaces PROFINET de este proyecto al modo V2.2 y anotó este ajuste en el archivo GSD.

Seguidamente se exportó el proyecto de I-device a un archivo GSD (2) y se insertó en el proyecto 2.

Debido a la entrada en el archivo GSD, STEP 7 puso automáticamente todas las interfaces PROFINET del proyecto 2 al modo V2.2.

En este ejemplo, el controlador IO de nivel superior parametriza el sistema IO subordinado (1).

Las herramientas SNMP de este ejemplo detectan únicamente una interfaz por dispositivo. Las herramientas de Siemens detectan todas las interfaces PROFINET.



① El controlador IO de nivel superior parametriza la interfaz PROFINET del I-Device en el sistema IO A.

② El archivo GSD PROFINET especifica que solo se soporta el modo IEC V2.2

③ Los dispositivos con modo IEC V2.2 fuerzan el modo IEC V2.2 en todas las interfaces PROFINET del proyecto 1.

Ajuste manual de las interfaces PROFINET

El ajuste manual de una interfaz PROFINET es necesario en los siguientes casos:

- Se desea configurar una instalación en dos o más proyectos. En uno de los proyectos (proyecto 1) se utiliza un dispositivo que solo soporta el modo V2.2. En otro de los proyectos (proyecto 2) se utilizan únicamente dispositivos con el modo V2.3. En tal caso, debe ajustarse manualmente al modo V2.2 uno de los dispositivos del proyecto 2.
- Se utilizan dos proyectos. El primer proyecto contiene un I-Device. El segundo proyecto contiene un sistema IO de nivel superior al que se conecta el I-Device. Este caso se describe con más detalle a continuación.

Ajustar manualmente el modo IEC

Cuando se dan las condiciones siguientes es necesario ajustar manualmente una interfaz PROFINET al modo IEC V2.2:

- Se configura un I-device y se exporta el proyecto como archivo GSD.
- Se utiliza el archivo GSD en otro proyecto.
- Se inserta el I-device en un sistema IO de nivel superior.
- El proyecto con el I-Device utiliza únicamente dispositivos que soportan el modo V2.3. Este modo se ajusta automáticamente.
- En el sistema IO de nivel superior hay un dispositivo que solo soporta la norma IEC V2.2.

Para forzar al modo V2.2 la interfaz PROFINET de un dispositivo conectado al sistema IO subordinado, haga lo siguiente.

1. Abra el proyecto de I-Device en STEP 7.
2. Seleccione cualquier dispositivo.
3. En el apartado "Opciones de interfaz", active la opción "Utilizar modo IEC V2.2 LLDP".

A continuación, genere de nuevo el archivo GSD. Utilice en el sistema IO de nivel superior el nuevo archivo GSD.

Cargue en la CPU del I-Device la nueva configuración del I-Device.

Ejemplo 2: Ajustar manualmente el modo IEC

Proyecto 1: En el proyecto se ha creado un I-Device.

Todas las interfaces PROFINET soportan el modo V2.3.

Por este motivo, STEP 7 no ha cambiado el modo de las interfaces PROFINET y ha anotado el modo V2.3 en el archivo GSD (2).

Proyecto 2: Se inserta el I-Device del proyecto 1 (archivo GSD).

Uno de los dispositivos solo soporta el modo V2.2.

Por ello, STEP 7 ajusta todas las interfaces PROFINET de este proyecto al modo V2.2.

Las interfaces PROFINET de los proyectos 1 y 2 están configuradas de distinta manera.

Por ello es necesario volver a abrir el proyecto del I-Device (proyecto 1) en STEP 7 y ajustar manualmente una interfaz PROFINET al modo V2.2.

Para ajustar al modo V2.2 la interfaz PROFINET de un dispositivo conectado al sistema IO subordinado, haga lo siguiente.

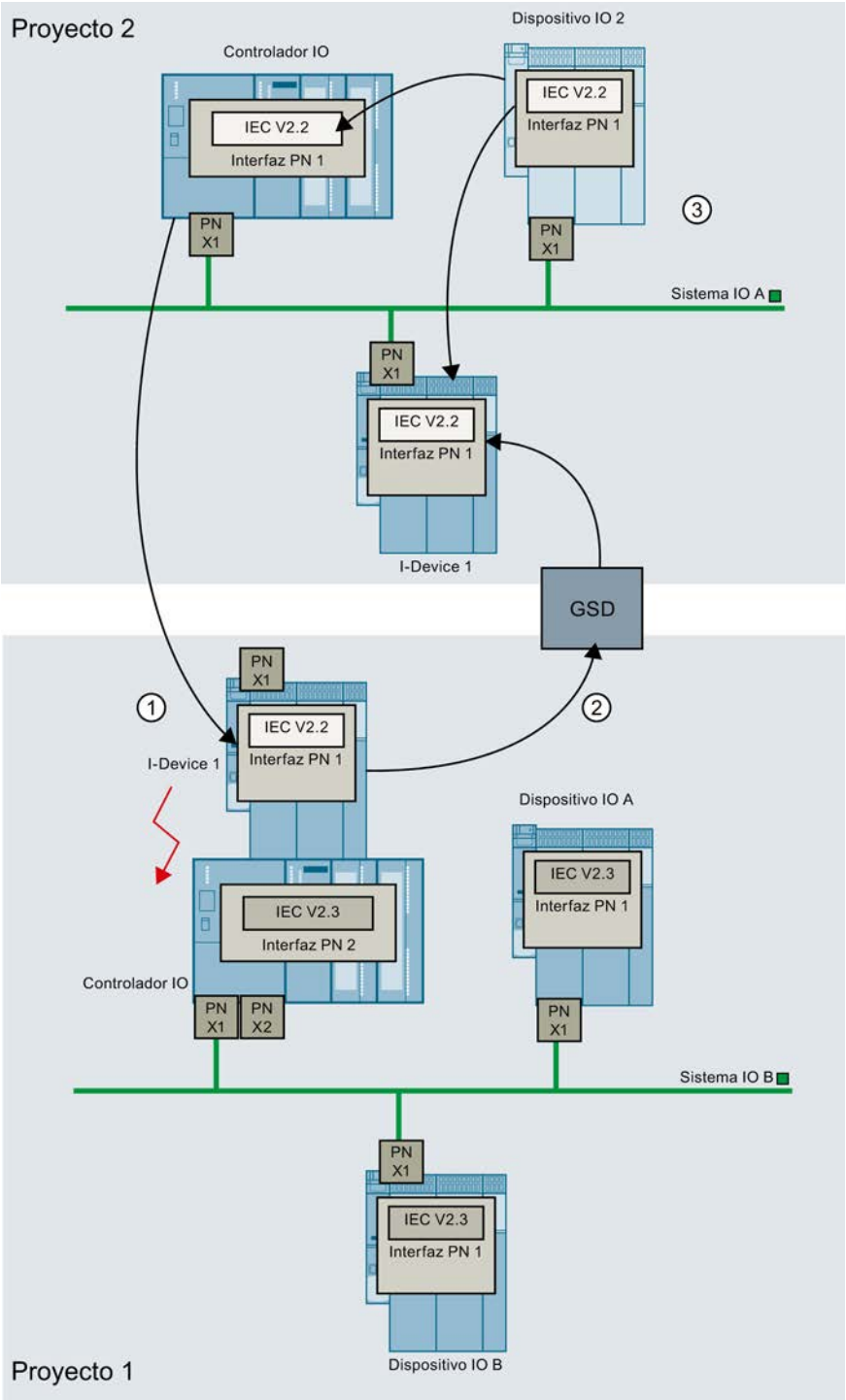
1. Abra el proyecto de I-Device en STEP 7.
2. Seleccione cualquier dispositivo.
3. En el apartado "Opciones de interfaz", active la opción "Utilizar modo IEC V2.2 LLDP".

A continuación genere un nuevo archivo GSD. Inserte el nuevo archivo GSD en el sistema IO de nivel superior (proyecto 2).

Vuelva a cargar el proyecto en la CPU del I-Device.

Las herramientas SNMP de este ejemplo detectan únicamente una interfaz por dispositivo. Las herramientas de Siemens detectan todas las interfaces PROFINET.

Si no se cambia el ajuste de la interfaz PROFINET, cada interfaz envía un aviso de diagnóstico en tiempo de ejecución (durante el funcionamiento).



① Las interfaces PROFINET del I-Device están configuradas de distinta manera (la del sistema IO A, para el modo V2.2, y la del sistema IO B, para el modo V2.3).

② El archivo GSD especifica que el I-Device soporta el modo IEC V2.3.

③ El dispositivo IO 2 soporta solo el modo V2.2. Por ello STEP 7 ajusta a este modo todas las demás interfaces PROFINET del proyecto 2 (incluida la interfaz PROFINET del I-Device del sistema IO A).

Ejemplo 3: Ajustar manualmente el modo IEC

Este ejemplo es bastante similar al ejemplo 2.

La única diferencia es que en el I-Device está guardada la configuración de las interfaces PROFINET (no se realiza la parametrización mediante el controlador IO de nivel superior como en el ejemplo 2).

en el sistema IO A, las interfaces PROFINET están ajustadas de distinta manera (I-Device para V2.3, controlador IO para V2.2, el dispositivo IO 2 soporta solo V2.2).

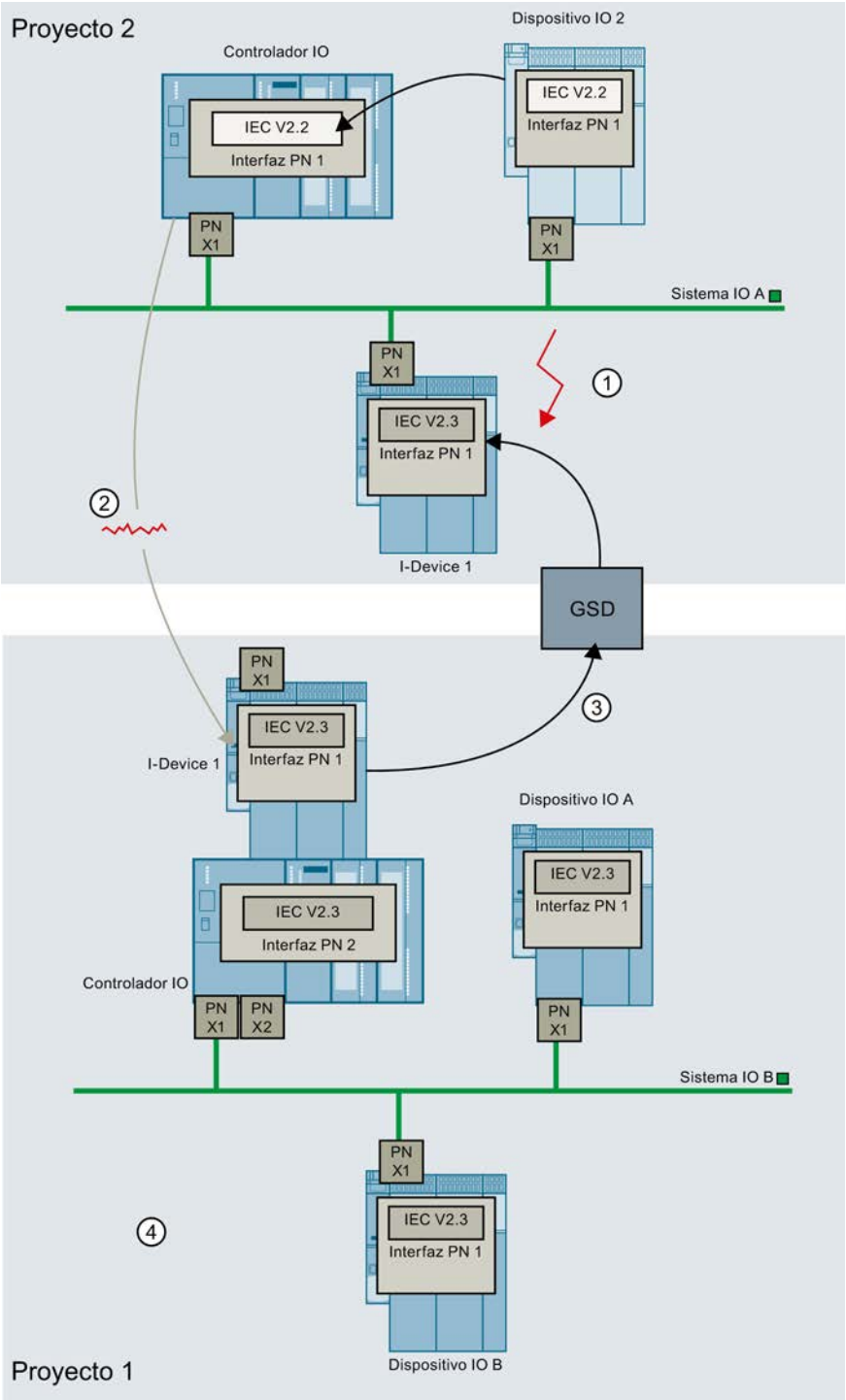
Por ello es necesario ajustar una interfaz PROFINET del sistema IO B al modo V2.2

Para ajustar al modo V2.2 la interfaz PROFINET de un dispositivo conectado al sistema IO subordinado, haga lo siguiente.

1. Abra el proyecto de I-Device en STEP 7.
2. Seleccione cualquier dispositivo.
3. En el apartado "Opciones de interfaz", active la opción "Utilizar modo IEC V2.2 LLDP".

Seguidamente, vuelva a generar el archivo GSD y utilice el nuevo archivo.

Cargue en la CPU del I-Device la nueva configuración del I-Device.


Proyecto 2
Controlador IO
Dispositivo IO 2
IEC V2.2
Interfaz PN 1
IEC V2.2
Interfaz PN 1
PN X1
PN X1
Sistema IO A
PN X1
IEC V2.3
Interfaz PN 1
I-Device 1
1
2
GSD
3
PN X1
IEC V2.3
Interfaz PN 1
I-Device 1
Dispositivo IO A
IEC V2.3
Interfaz PN 1
IEC V2.3
Interfaz PN 2
Controlador IO
PN X1
PN X2
PN X1
Sistema IO B
PN X1
IEC V2.3
Interfaz PN 1
4
Proyecto 1
Dispositivo IO B

① en el sistema IO A, las interfaces PROFINET están ajustadas de distinta manera (I-Device para V2.3, controlador IO para V2.2, el dispositivo IO 2 soporta solo V2.2).

② en el I-Device está guardada la configuración de las interfaces PROFINET (no se realiza la parametrización mediante el controlador IO de nivel superior).

③ En el archivo GSD se especifica que está ajustado el modo IEC V2.3.

④ Todos los dispositivos están ajustados automáticamente al modo V2.3. Por ello es necesario ajustar manualmente al modo V2.2 una interfaz PROFINET en el proyecto 1.

Funciones de reloj

Principios básicos de las funciones de reloj

Todas las CPUs S7-1500 disponen de un reloj interno. El respaldo permite visualizar la hora correcta en caso de interrupción de la fuente de alimentación durante un máximo de 10 horas.

Formato de hora

El reloj muestra siempre la hora con una resolución de 1 milisegundo y la fecha con el día de la semana. Se tiene en cuenta el cambio de hora provocado por el horario de verano.

Ajustar y leer la hora

Ajuste y lectura de la hora con instrucciones

La hora y la fecha del reloj de la CPU se pueden ajustar, iniciar y leer en el programa de usuario por medio de las instrucciones siguientes:

- Ajustar la hora: "WR_SYS_T"
- Leer la hora "RD_SYS_T"
- Leer la hora local "RD_LOC_T"
- Escribir la hora local "WR_LOC_T"

Ajuste manual

También es posible leer y ajustar la hora manualmente en la vista online y de diagnóstico, en "Funciones > Ajustar hora".

Consulte también

WR_SYS_T: Ajustar la hora (Página 4695)

RD_SYS_T: Leer la hora (Página 4696)

RD_LOC_T: Leer hora local (Página 4697)

WR_LOC_T: escribir hora local (Página 4698)

SET_TIMEZONE: Ajustar zona horaria (Página 4700)

Sincronización horaria

Referencia

Parametrización de las propiedades de la interfaz en el grupo de parámetros "Propiedades > General > Interfaz DP > Sincronización horaria".

Parámetros del reloj

Mediante los parámetros del reloj se ajusta lo siguiente:

- Tipo de sincronización
  Si está configurado el tipo de sincronización "Como maestro", se puede elegir el intervalo de tiempo para la sincronización automática.
  Si en una red hay más de un módulo con reloj, hay que determinar qué CPU actuará como maestra y cuál como esclava para la sincronización del reloj. Sólo puede haber un reloj maestro.
- Intervalo de actualización
  El intervalo de actualización define el espacio de tiempo entre las consultas de hora.

Instrucciones para la sincronización de la hora

Para garantizar que la hora coincida en todos los módulos de la red, el programa de sistema sincroniza los relojes esclavos a intervalos regulares parametrizables. Con la instrucción "SNC_RTCB" se puede enviar la fecha y hora del reloj maestro al reloj esclavo, independientemente del intervalo de sincronización parametrizado.

Sincronización horaria en el procedimiento NTP

Referencia

Parametrización de las propiedades de la interfaz en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Sincronización horaria". Está seleccionada la opción "Activar sincronización horaria vía servidor NTP".

Procedimiento NTP (NTP: Network Time Protocol)

Con el procedimiento NTP, el dispositivo envía regularmente consultas de hora (en modo cliente) al servidor NTP de la subred (LAN). Sobre la base de las respuestas de los servidores se determina la hora más exacta y fiable y se sincroniza la hora de la estación.

La ventaja de este procedimiento es la posibilidad de sincronizar la hora más allá de los límites de la subred.

Deben configurarse las direcciones IP de hasta cuatro servidores NTP. El intervalo de actualización define el espacio de tiempo entre las consultas de hora (en segundos). El rango de valores del intervalo está entre 10 segundos y un día.

En el procedimiento NTP se transfiere generalmente la hora UTC (Universal Time Coordinated), que equivale a la hora GMT (Greenwich Mean Time).

Parámetros de la sincronización horaria en el procedimiento NTP

Mediante los parámetros del reloj se ajusta lo siguiente:

- Activar sincronización horaria vía servidor NTP
  Active esta casilla de verificación si hay que sincronizar el reloj interno con el procedimiento de sincronización NTP.
- Servidor 1-4
  Deben configurarse las direcciones IP de hasta cuatro servidores NTP.
- Intervalo de actualización
  El intervalo de actualización define el espacio de tiempo entre las consultas de hora.

Uso de marcas de sistema

Marcas de sistema

Una marca de sistema es una marca con valores definidos.

En la parametrización de la marca de sistema se determina el byte de marcas de la CPU que se convertirá en el byte de marcas de sistema.

Uso

Las marcas de sistema pueden utilizarse en el programa de usuario, p. ej. para ejecutar partes del programa sólo en el primer ciclo tras el arranque o para evaluar el diagnóstico al cambiar el estado del mismo. Dos marcas de sistema son permanentemente 1 ó 0.

Bits del byte de marcas de sistema

La tabla siguiente explica el significado de las marcas de sistema:

Bit del byte de marcas de sistema	7	6	5	4	3	2	1	0
Significado	Reservado (=0)	Reservado (=0)	Reservado (=0)	Reservado (=0)	=0	=1	=1 al cambiar el estado del diagnóstico	=1 en el primer ciclo tras el arranque, de lo contrario, 0

Nota

El byte de marcas seleccionado no puede emplearse para almacenar datos en la memoria intermedia.

Uso de marcas de ciclo

Marcas de ciclo

Una marca de ciclo es una marca que modifica periódicamente su estado binario en una relación 1:1 entre impulso y pausa.

En la parametrización de la marca de ciclo se determina el byte de marcas de la CPU que se convertirá en el byte de marcas de ciclo.

Uso

Las marcas de ciclo se utilizan en el programa de usuario, p. ej., para controlar indicadores luminosos con una luz intermitente o lanzar procesos periódicos, como la adquisición de un valor real.

Frecuencias posibles

Cada bit del byte de marcas de ciclo tiene asignada una frecuencia. La tabla siguiente muestra la asignación:

Bit del byte de marcas de ciclo	7	6	5	4	3	2	1	0
Duración del período (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frecuencia (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Nota

Las marcas de ciclo se ejecutan de forma asíncrona al ciclo de la CPU, es decir, en ciclos largos es posible que el estado de la marca de ciclo cambie varias veces.

El byte de marcas seleccionado no puede emplearse para almacenar datos en la memoria intermedia.

Protección

Posibilidades de ajuste de la protección

Nivel de protección

A continuación se explica cómo utilizar los diferentes niveles de acceso de las CPU S7-1500.

Las CPU S7-1500 ofrecen diferentes niveles de acceso para restringir el uso de determinadas funciones.

La parametrización de los niveles de acceso se realiza en una tabla. Las marcas de verificación verdes de las columnas de la derecha del respectivo nivel de acceso indican qué operaciones son posibles como máximo sin conocer la contraseña de este nivel de acceso. Para utilizar las funciones que no tienen marcada la casilla de verificación es necesario introducir una contraseña.

ATENCIÓN
La configuración de un nivel de acceso no sustituye la protección de know how La parametrización de niveles de acceso ofrece una excelente protección contra modificaciones no autorizadas de la CPU, gracias a la limitación de los derechos de carga en la CPU. En cualquier caso, los bloques de la Memory Card no están protegidos contra escritura o lectura. Para proteger el código de los bloques de la Memory Card se utiliza la protección de know-how.

Comportamiento predeterminado

El nivel de acceso predeterminado es "Acceso completo (sin protección)". Todo usuario puede leer y modificar la configuración hardware y los bloques. No hay ninguna contraseña parametrizada y tampoco se necesita para el acceso online.

Los distintos niveles de acceso

En una CPU S7-1500 se pueden parametrizar los siguientes niveles de acceso:

- Acceso completo (sin protección): la configuración hardware y los bloques pueden ser leídos y modificados por cualquier persona.
- Acceso de lectura para bloques F (solo para CPU F): los bloques F del programa de seguridad no pueden modificarse sin legitimación mediante la contraseña de este nivel de acceso u otro superior.
  Encontrará más información al respecto en el manual de programación y manejo *SIMATIC Safety - Configurar y programar.*
- Acceso de lectura: con este nivel de acceso solo es posible, sin introducir la contraseña, el acceso de lectura a la configuración hardware y a los bloques, es decir, se pueden cargar la configuración hardware y los bloques en la programadora. Además, también es posible el acceso a HMI y a los datos de diagnóstico.
  Sin introducir la contraseña no se pueden cargar en la CPU ningún módulo ni ninguna configuración hardware. Además, sin la contraseña **no** son posibles las funciones de test con acceso de escritura ni las actualizaciones de firmware.

- Acceso a HMI: con este nivel de acceso, si no se introduce la contraseña solo es posible el acceso a HMI y a los datos de diagnóstico.
  Sin introducir la contraseña, los bloques o la configuración hardware no se pueden ni cargar en la CPU ni cargar desde la CPU a la programadora. Además, sin introducir la contraseña **tampoco** es posible: funciones de test con acceso de escritura, cambio del estado operativo (RUN/STOP) y actualización de firmware.
- Sin acceso (protección completa): cuando la CPU dispone de protección completa no se tiene acceso ni de lectura ni de escritura a la configuración hardware y los bloques. Tampoco es posible el acceso HMI. La función de servidor para la comunicación PUT/GET está desactivada en este nivel de acceso (no puede modificarse).
  La legitimación con la contraseña proporciona acceso total a la CPU.

Comportamiento de un módulo protegido por contraseña durante el funcionamiento

La protección de la CPU será efectiva una vez que los ajustes se hayan cargado en la CPU.

Antes de ejecutar una función online se comprueban los permisos y, si existe protección por contraseña, se pide que se introduzca la contraseña.

Ejemplo: el módulo se ha parametrizado con acceso de lectura y se desea ejecutar la función "Forzar variable". Puesto que para ello se requiere un acceso de escritura, para ejecutar la función hay que introducir la contraseña parametrizada.

Las funciones protegidas por contraseña solo pueden ejecutarse desde una programadora o un PC en un momento dado. No se permite el acceso desde otra programadora o PC.

El permiso de acceso a los datos protegidos tiene validez mientras dure la conexión online o hasta que se desactive manualmente con "Online > Borrar derechos de acceso".

Todo nivel de acceso permite, incluso sin introducir una contraseña, el acceso ilimitado a determinadas funciones como, p. ej., la identificación mediante la función "Dispositivos accesibles".

El acceso a una CPU S7-1500 protegida con contraseña puede limitarse localmente en el display. La restricción solo es efectiva si el selector de modo está en RUN.

Consulte también

Bloquear el acceso online a la CPU mediante el display (Página 1960)

Restricción de los servicios de comunicación (Página 1959)

Protección de los datos de proyecto (Página 389)

Parametrización de niveles de acceso (Página 1957)

Proteger bloques (Página 3045)

Imprimir los datos de proyecto (Página 407)

Parametrización de niveles de acceso

A continuación se explica cómo parametrizar un nivel de acceso e introducir una contraseña para una CPU S7-1500.

En una CPU S7-1500 es posible introducir varias contraseñas y, por tanto, establecer diferentes derechos de acceso para distintos grupos de usuarios.

Las contraseñas se introducen en una tabla, de modo que cada una tiene asignado exactamente un nivel de acceso.

En la columna "Nivel de acceso" se describen los efectos de la contraseña.

Ejemplo

Para una CPU estándar (es decir, no una CPU F) se elige el nivel de acceso "Sin acceso (protección completa)" y se introduce una contraseña propia para cada nivel de acceso situado encima en la tabla.

Para los usuarios que no conocen ninguna de las contraseñas, la CPU está completamente protegida. Ni siquiera son posibles accesos HMI.

Para los usuarios que conocen una de las contraseñas parametrizadas, el efecto depende de la fila de la tabla en la que se encuentre la contraseña:

- El efecto de la contraseña de la fila 1 (Acceso completo (sin protección)) es como si la CPU no estuviera protegida. Los usuarios que conocen esta contraseña tienen un acceso ilimitado a la CPU.
- El efecto de la contraseña de la fila 2 (Acceso de lectura) es como si la CPU estuviera protegida contra escritura. A pesar de conocer esta contraseña, los usuarios solo tienen acceso de lectura a la CPU.
- El efecto de la contraseña de la fila 3 (acceso HMI) es como si la CPU estuviera protegida contra escritura y lectura, de modo que los usuarios que conocen esta contraseña solo disponen de acceso HMI.

Procedimiento

Para parametrizar los niveles de acceso de una CPU S7-1500 proceda del siguiente modo:

1. Abra las propiedades del módulo en la ventana de inspección.
2. Abra la entrada "Protección" en la navegación local.
   En la ventana de inspección se muestra una tabla con los niveles de acceso posibles.

Nivel de acceso	Acceso			Permiso de acceso	
Protección	HMI	Leer	Escribir	Contraseña	Confirmación
Acceso completo (sin protección)	✓	✓	✓		
Acceso de lectura	✓	✓			
Acceso HMI	✓				
Sin acceso (protección completa)					

Figura 8-54 Protección de acceso

3. Active el nivel de acceso deseado en la primera columna de la tabla. Las marcas de verificación verdes de las columnas de la derecha del respectivo nivel de protección indican qué operaciones son posibles sin introducir la contraseña.

4. En la columna "Contraseña", asigne una contraseña en la primera fila para el nivel de acceso seleccionado. Repita la contraseña elegida en la columna "Confirmación" con objeto de evitar entradas erróneas.
   Asegúrese de que la contraseña sea lo suficientemente segura, es decir, que no tenga un patrón reconocible por una máquina.
   La entrada de una contraseña en la primera fila "Acceso completo (sin protección)" es obligatoria y permite a la persona que conoce la contraseña acceder de forma ilimitada a la CPU, independientemente del nivel de protección seleccionado.
5. Asigne otras contraseñas a otros niveles de acceso según requiera, siempre que el nivel de protección seleccionado lo permita.
6. Cargue la configuración hardware para que el nivel de acceso se haga efectivo.

Resultado

La configuración hardware y los bloques están protegidos contra los accesos no autorizados conforme al nivel de acceso configurado. Si una operación no se puede ejecutar sin contraseña debido al nivel de acceso parametrizado, se muestra un cuadro de diálogo para introducir una contraseña.

Consulte también

Posibilidades de ajuste de la protección (Página 1955)

Restricción de los servicios de comunicación

Introducción

La CPU puede ser el servidor de una serie de servicios de comunicación, es decir, que sin necesidad de configurar y programar conexiones para la CPU, otros dispositivos de comunicación pueden acceder a los datos de la CPU.

Con ello se prescinde de la posibilidad para la CPU local como servidor de controlar la comunicación con los clientes.

Si este tipo de comunicación durante el funcionamiento está autorizado o no para la CPU local, se determina mediante el parámetro "Mecanismos de conexión" en el área "Protección" de los parámetros de la CPU.

Permitir acceso vía comunicación PUT/GET del interlocutor remoto

En el ajuste predeterminado, la opción "Permitir acceso vía comunicación PUT/GET del interlocutor remoto (...)" está desactivada. En este caso, el acceso de lectura y de escritura a los datos de la CPU solo será posible a través de conexiones de comunicación que requieren una configuración o programación tanto para la CPU local como para el interlocutor de comunicación. Todavía son posibles, p. ej., los accesos a través de instrucciones BSEND/BRCV.

Las conexiones para las que la CPU local es solo servidor (es decir, cuando para la CPU local no hay una configuración/programación de la comunicación con el interlocutor) no son posibles durante el funcionamiento de la CPU; p. ej.

- en accesos PUT/GET, FETCH/WRITE o por FTP a través de módulos de comunicaciones
- en accesos PUT/GET de otras CPU S7
- en accesos HMI que se realizan a través de la comunicación PUT/GET

Si se desea permitir el acceso del cliente a los datos de la CPU, es decir, si no se desea restringir los servicios de comunicación de la CPU, se debe activar la opción "Permitir acceso vía comunicación PUT/GET del interlocutor remoto".

Bloquear el acceso online a la CPU mediante el display

En el display de una CPU S7-1500 puede bloquear el acceso a una CPU protegida con contraseña (bloqueo local).

El bloqueo de acceso sólo tiene efecto si el selector de modo está en RUN.

El bloqueo de acceso funciona independientemente de la protección por contraseña, es decir, si alguien accede a la CPU a través de una programadora conectada y ha introducido la contraseña correcta, el acceso a la CPU sigue sin estar permitido.

El bloqueo de acceso es ajustable para cada nivel de protección independientemente en el display, lo que significa que, p. ej., el acceso de lectura está permitido localmente, pero el acceso de escritura no está permitido localmente.

Adicionalmente se puede parametrizar una contraseña para el display en las propiedades de la CPU, de forma que la protección de acceso local cuente con una contraseña local.

Procedimiento

Para ajustar la protección de acceso local para una CPU S7-1500 en el display, proceda del siguiente modo:

1. Seleccione en el display el menú Ajustes > Protección.
2. Confirme la elección con "OK" y ajuste para cada nivel de protección si el acceso en RUN está permitido o no:
   Permitir: el acceso a la CPU es posible mediante la introducción de la contraseña.
   Bloquear en RUN: si el selector de modo está en RUN, no puede registrarse ningún usuario más con los derechos de este nivel de protección en la CPU, aunque conozca la contraseña. En STOP, el acceso es posible mediante la introducción de la contraseña.

Consulte también

Posibilidades de ajuste de la protección (Página 1955)

Test, puesta en servicio y routing

Test, puesta en servicio y routing

Active la casilla de verificación si desea realizar con frecuencia funciones de programadora necesarias para la puesta en marcha y el test, por ejemplo, utilizando esta interfaz.

Si se activa la opción "Test, puesta en servicio y routing", la interfaz se convertirá en el dispositivo activo en PROFIBUS (es decir, la interfaz participa en la rotación del token de PROFIBUS). En este caso serán posibles las funciones siguientes:

- Programar (p. ej. cargar)
- Probar (estado/forzar)
- Routing S7 (esclavo I como transición de red)

El tiempo de ciclo puede prolongarse. En aplicaciones críticas y si no se necesitan el routing S7 ni la funcionalidad de cliente para la comunicación, no debería activarse esta opción.

Si la casilla de verificación "Test, puesta en servicio y routing" no está activada, la interfaz sólo trabaja como servidor de servicios de comunicación.

La funcionalidad de servidor significa que sólo el interlocutor (p. ej. PG, OP o sistema de automatización) puede iniciar la comunicación. Las particularidades siguientes deben tenerse en cuenta si la opción "Test, puesta en servicio y routing" no está activada:

- No es posible el routing S7
- La programación y el test son posibles pero lentamente (sin prolongación del tiempo de ciclo)

Bloques de organización

Información de arranque del OB

Estructura de la información de arranque del OB

Al parametrizar las propiedades de un OB para CPU S7-1500, se define también la estructura de la información de arranque del OB:

- Como en las CPU S7-300 y S7-400
- Información de arranque optimizada

Información de arranque del OB como en las CPU S7-300 y S7-400

La información de arranque del OB forma parte de los datos locales y tiene la misma estructura que en las CPU S7-300 y S7-400.

Información de arranque optimizada

La información de arranque optimizada es un juego de variables de entrada.

Solamente proporciona la información necesaria para identificar el evento de arranque del OB.

En los distintos OB, la información que sea idéntica en cuanto a semántica se mostrará bajo el mismo nombre de variable dentro de la información de arranque optimizada.

Descripción de todos los tipos de bloques de organización

OBs de arranque

Descripción

El sistema operativo llama una sola vez a cada OB de arranque al conmutar del estado "STOP" a "RUN". Si existen varios OB de arranque, se llaman de acuerdo con el orden de sus números de OB, empezando con el OB que tiene el número más bajo.

El conjunto de todos los OB de arranque se denomina programa de arranque.

En el programa de arranque se pueden realizar ajustes predeterminados para su programa cíclico.

Aquí, todos los valores de la memoria imagen de proceso de las entradas tienen el valor cero. La ejecución del programa de usuario no está limitada en el tiempo. No es posible utilizar bloques de organización controlados por tiempo ni por alarmas.

Una vez que el programa de arranque se ha procesado completamente, el sistema operativo lee la memoria imagen de proceso de las entradas e inicia el programa cíclico.

Estructura de la información de arranque

Un OB de arranque tiene la siguiente información de arranque:

- Como en las CPU S7-300 y S7-400
  Consulte OBs de arranque (OB 100, OB 101 y OB 102) (Página 1875)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - OB_NUMBR=B#16#FF, si el número de OB es > 254
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LostRetentive	BOOL	=TRUE, si se han perdido los contenidos de las áreas de datos remanentes
LostRTC	BOOL	=TRUE, si se ha perdido la hora del reloj en tiempo real

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OBs de ciclo

Descripción

La base principal del programa de usuario es el programa cíclico. Sus partes de programa correspondientes se procesan cíclicamente: cuando finaliza su procesamiento, el sistema operativo lo vuelve a comenzar.

El programa cíclico presenta uno o varios OB de ciclo. Estos OB tienen la prioridad 1, es decir, la prioridad más baja de todos los OB. Por tanto, los eventos de cualquier otra clase de prioridad pueden interrumpir el programa cíclico.

Si ha creado varios OB de ciclo, estos se llaman uno tras otro en el orden correspondiente a su número de OB. El OB de ciclo con el menor número de OB se llama primero.

Los siguientes eventos inician el programa cíclico:

- Fin del procesamiento del arranque
- Fin del procesamiento de la ejecución anterior del programa cíclico

Una vez finalizado el programa cíclico, el sistema operativo actualiza las memorias imagen de proceso como sigue:

1. Escribe los valores de la memoria imagen de proceso de las salidas en los módulos de salidas.
2. Lee las entradas de los módulos de entradas y las transfiere a la memoria imagen de proceso de las entradas.

Vigilancia del tiempo de ciclo

Por tiempo de ciclo se entiende el tiempo de ejecución del programa cíclico más el tiempo de ejecución de todas las partes de programa conectadas de los OB de mayor prioridad. Por tanto, si ha creado varios OB de ciclo, cada uno de ellos contribuye al tiempo de ciclo.

El sistema operativo vigila si el tiempo de ciclo permanece por debajo del tiempo de ciclo máximo parametrizado. En caso de que el tiempo de ciclo exceda el tiempo de ciclo máximo, se llama al OB 80 (evento de arranque W#16#0002:3501). Esto puede evitarse iniciando de nuevo la vigilancia de tiempo en el punto adecuado del programa, es decir, llamando la instrucción RE_TRIGR. Si el tiempo de ciclo máximo se excede una segunda vez dentro de un ciclo (sin reiniciar la vigilancia de tiempo), la CPU pasa a STOP (evento W#16#0002:3500).

Tiempo de ciclo mínimo

Además de vigilar el tiempo de ciclo máximo, también se garantiza el mantenimiento de un tiempo de ciclo mínimo. Para ello, el sistema operativo retarda el inicio de un ciclo nuevo hasta que se ha alcanzado el tiempo de ciclo mínimo.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte Programa cíclico (OB 1) (Página 1822)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - SCAN_1=B#16#01 para la primera llamada y B#16#03 para las otras llamadas
  - OB_NUMBR=B#16#FF, si el número de OB es > 254

- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
Initial_Call	BOOL	=TRUE en la primera llamada de este OB con: • Transición de STOP o PARADA a RUN • Tras recargar
Remanence	BOOL	=TRUE si existen datos remanentes

Consulte también

Eventos y OBs (Página 1924)

Evaluación del estado de valor (Página 1917)

Información de arranque del OB (Página 1961)

OBs de alarma horaria

Descripción

Las CPU S7-1500 disponen de OB de alarma horaria que pueden iniciarse como sigue:

- Una sola vez a una hora determinada (fecha con hora)
- Periódicamente (aquí son posibles los intervalos de tiempo siguientes: cada minuto, cada hora, cada día, cada semana, cada mes, a finales de mes, cada año)

Inicio de un OB de alarma horaria

Para que la CPU pueda iniciar un OB de alarma horaria se deben cumplir los siguientes requisitos:

- La alarma horaria correspondiente debe haberse ajustado y activado previamente. (Por ajuste de una alarma horaria se entiende la especificación de su hora de inicio y la duración del período).
- El OB de alarma horaria debe estar cargado en la CPU.

Para ajustar una alarma horaria y activarla, tiene las siguientes tres opciones:

- Ajustar y activar la alarma horaria con STEP 7.
- Ajustar la alarma horaria con STEP 7 y activarla llamando la instrucción ACT_TINT desde el programa.
- Ajustar la alarma horaria llamando la instrucción SET_TINTL y activarla llamando la instrucción ACT_TINT cada vez desde el programa.

Reglas para alarmas horarias

- Si se ajusta una alarma horaria en la que el OB correspondiente debe ejecutarse una única vez, la hora de inicio no podrá encontrarse en el pasado (referido al reloj en tiempo real de la CPU).
- Si se ajusta una alarma horaria en la que el OB correspondiente debe ejecutarse de forma periódica, pero que tiene la hora de inicio en el pasado, el OB de alarma horaria se ejecutará según la hora actual cuando venza la siguiente hora.
- La fecha de las alarmas horarias periódicas debe corresponder a una fecha real. Así, por ejemplo, la repetición mensual de un OB de alarma horaria con fecha de inicio 31.1. no sería posible. En tal caso, el OB solo se iniciaría en los meses que tuvieran 31 días.
- Una alarma horaria que se activa durante el arranque no se ejecutará hasta que finalice el arranque.
- Un arranque borra todas las alarmas horarias que se han ajustado y activado por medio de instrucción en el programa de usuario.
- En la transición de PARADA a RUN, el sistema operativo comprueba si existen alarmas horarias expiradas. En caso afirmativo, llama al OB de error de tiempo. Si se ha seleccionado la misma información de arranque para este OB que para las CPU S7-300 y S7-400, el sistema operativo registra de 10 a 17 para los OB (y solo para estos) si las alarmas horarias han pasado. Una vez finalizado el procesamiento del OB 80 se recupera el primer OB que haya fallado.

Comportamiento de las alarmas horarias al adelantar o retrasar la hora

El comportamiento de las alarmas horarias depende de la diferencia de tiempo con que se adelanta o retrasa la hora. Esto se explica más detalladamente en la siguiente tabla:

Cambio de hora	Repercusión en la alarma horaria
< 20 s	• Al adelantar el reloj: se recupera cualquier alarma horaria que se haya saltado. La variable de información de arranque CaughtUp se activa. No se llama a OB 80. • Al retrasar el reloj: las alarmas horarias que ya se habían ejecutado en el intervalo de la hora retrasada o que estaban activas no se repiten.
> > = 20 s	• Al adelantar el reloj: si se habían saltado una o varias alarmas horarias, se llama a OB 80. La información de arranque proporciona para cada prioridad la información cuando se ha saltado mínimo una alarma horaria. Si no se borra la alarma horaria en el OB 80, se recupera la primera alarma horaria que se haya saltado y se activa la variable de información de arranque "CaughtUp". • Al retrasar el reloj: se repiten las alarmas horarias situadas en el intervalo de la hora retrasada. Si la corrección de tiempo es mayor que el periodo de una alarma horaria, el sistema operativo calcula la hora de la primera alarma que debe repetirse. Cuando se repite una alarma horaria, la variable de información de arranque "SecondTime" está activa.
Cambio del horario de invierno al horario de verano y viceversa	El cambio repercute únicamente cuando se utiliza la hora local. • Cuando se cambia del horario de invierno al de verano, se recuperan las alarmas horarias que se hayan saltado y se activa la variable de información de arranque "CaughtUp". No se llama al OB 80. • Cuando se cambia del horario de verano al de invierno, las alarmas horarias situadas en el intervalo de la hora retrasada se repiten. Cuando se repite una alarma horaria, la variable de información de arranque "SecondTime" está activa.
Cambio de la zona horaria (provoca el cambio de la hora local)	La hora de inicio de las alarmas horarias activas vuelve a calcularse. • Cuando se cambia la hora local, se recuperan las alarmas horarias que se hayan saltado y se activa la variable de información de arranque "CaughtUp". No se llama al OB 80. • Si se retrasa la hora local, las alarmas horarias que ya se han ejecutado se repiten según sea necesario. Cuando se repite una alarma horaria, la variable de información de arranque "SecondTime" está activa.

Vigilancia de tiempo

La vigilancia de tiempo por parte del sistema operativo puede provocar los siguientes eventos al trabajar con alarmas horarias:

- W#16#0002:3504
  La hora se ha adelantado debido al cambio de horario de invierno/verano. Con ello se puede haber saltado la hora de inicio de algunas alarmas horarias. Se llama el OB 80.
- W#16#0002:3505
  La hora se ha retrasado. Con ello se puede haber saltado la hora de inicio de algunas alarmas horarias. Se llama el OB 80.
- W#16#0002:3507
  Desbordamiento del búfer para eventos de arranque de OB. Se realiza una entrada en el búfer de diagnóstico.
- W#16#0002:3502
  El número de eventos de arranque de OB pendientes supera un límite predeterminado. Se llama el OB 80.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OBs de alarma horaria (de OB 10 a OB 17) (Página 1823)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - STRT_INF=B#16#1F, si el número de OB es > 122
  - OB_NUMBR=B#16#FF, si el número de OB es > 254
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
CaughtUp	BOOL	=TRUE, si se ha ejecutado un OB a posteriori por haberse adelantado la hora
SecondTime	BOOL	=TRUE, si se ha llamado a un OB por segunda vez debido al retraso de la hora. Nota: SecondTime se activa una única vez con la hora retrasada.

Consulte también

Parametrizar OB de alarma horaria (Página 1987)

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de alarma de estado

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de alarma de estado cuando recibe una alarma de este tipo. Esto puede suceder cuando un módulo de un esclavo cambia de estado operativo, p. ej., de "RUN" a "STOP". Puede obtenerse más información sobre los eventos que originan una alarma de estado en la documentación del fabricante del esclavo o dispositivo.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de alarma de estado (OB 55) (Página 1832)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del componente que ha disparado la alarma
Slot	UINT	Número de slot del componente causante de la alarma
Especificador	WORD	Especificador de alarma del telegrama de alarma: • PROFIBUS: – Bits 0 a 1: especificador de alarma – Bit 2: Add_Ack – Bits 3 a 7: n.º de sec. – Bits 8 a 15: reservados • Central y PNIO: – Bits 0 a 10: número de secuencia (rango de valores de 0 a 2047) – Bit 11: diagnóstico de canal: 0: no existe diagnóstico de canal 1: existe diagnóstico de canal – Bit 12: estado del diagnóstico específico de fabricante: 0: no existe información de estado específica del fabricante 1: existe información de estado específica del fabricante – Bit 13: estado del diagnóstico en el submódulo: 0: no existe información de estado, se han eliminado todos los errores 1: existe por lo menos un diagnóstico de canal y/o información de estado – Bit 14: reservado – Bit 15: Application Relationship Diagnosis State: 0: ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico 1: por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de alarma de actualización

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de alarma de actualización cuando recibe una alarma de este tipo. Esto puede suceder cuando se han modificado parámetros en un slot de un esclavo o dispositivo. Puede obtenerse más información sobre los eventos que originan una alarma de actualización en la documentación del fabricante del esclavo o dispositivo.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de alarma de actualización (OB 56) (Página 1836)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del componente que ha disparado la alarma
Slot	UINT	Número de slot del componente causante de la alarma
Especificador	WORD	Especificador de alarma del telegrama de alarma: • PROFIBUS: – Bits 0 a 1: especificador de alarma – Bit 2: Add_Ack – Bits 3 a 7: n.º de sec. – Bits 8 a 15: reservados • Central y PNIO: – Bits 0 a 10: número de secuencia (rango de valores de 0 a 2047) – Bit 11: diagnóstico de canal: 0: no existe diagnóstico de canal 1: existe diagnóstico de canal – Bit 12: estado del diagnóstico específico de fabricante: 0: no existe información de estado específica del fabricante 1: existe información de estado específica del fabricante – Bit 13: estado del diagnóstico en el submódulo: 0: no existe información de estado, se han eliminado todos los errores 1: existe por lo menos un diagnóstico de canal y/o información de estado – Bit 14: reservado – Bit 15: Application Relationship Diagnosis State: 0: ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico 1: por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB para la alarma específica del fabricante o perfil

Descripción

El sistema operativo de la CPU S7-1500 llama al OB57 cuando recibe una alarma específica del fabricante o del perfil. Puede obtenerse más información sobre los eventos que originan una alarma de este tipo en la documentación del fabricante del esclavo o dispositivo.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de alarmas específicas del fabricante (OB 57) (Página 1840)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  – STRT_INF=B#16#67
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del componente que ha disparado la alarma
Slot	UINT	Número de slot del componente causante de la alarma
Especificador	WORD	Especificador de alarma del telegrama de alarma: • PROFIBUS: – Bits 0 a 1: especificador de alarma – Bit 2: Add_Ack – Bits 3 a 7: n.º de sec. – Bits 8 a 15: reservados • Central y PNIO: – Bits 0 a 10: número de secuencia (rango de valores de 0 a 2047) – Bit 11: diagnóstico de canal: 0: no existe diagnóstico de canal 1: existe diagnóstico de canal – Bit 12: estado del diagnóstico específico de fabricante: 0: no existe información de estado específica del fabricante 1: existe información de estado específica del fabricante – Bit 13: estado del diagnóstico en el submódulo: 0: no existe información de estado, se han eliminado todos los errores 1: existe por lo menos un diagnóstico de canal y/o información de estado – Bit 14: reservado – Bit 15: Application Relationship Diagnosis State: 0: ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico 1: por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de alarma de retardo

Descripción

El sistema operativo llama a un OB de alarma de retardo al cabo de un tiempo de retardo parametrizable. El tiempo de retardo comienza a transcurrir con la llamada de la instrucción "SRT_DINT" (es el arranque de la alarma de retardo correspondiente).

Para que el sistema operativo pueda llamar a un OB de alarma de retardo, deben cumplirse los siguientes requisitos:

- La alarma de retardo debe haber sido activada llamando a la instrucción "SRT_DINT".
- El OB de alarma de retardo no debe haber sido desactivado mediante configuración.

El OB de alarma de retardo debe estar presente en la CPU.

Influir en las alarmas de retardo con instrucciones

Se puede anular una alarma de retardo ya arrancada, cuyo OB de alarma de retardo todavía no haya sido llamado por el sistema operativo, con la instrucción "CAN_DINT", o volver a arrancarla con la instrucción "SRT_DINT" (y, de este modo, asignarle otro tiempo de retardo). Asimismo, la instrucción "QRY_DINT" permite consultar el estado actual de una alarma de retardo.

Las alarmas de retardo se pueden inhibir y habilitar con las instrucciones "DIS_AIRT" y "EN_AIRT", respectivamente.

Nota

Si tras ejecutar "SRT_DINT", inhibe esta alarma con "DIS_AIRT", no se seguirá procesando hasta habilitarla con "EN_AIRT", por lo que el tiempo de retardo se prolongará de forma correspondiente.

Llamada de OB de error en relación con OB de alarma de retardo

Si transcurre el tiempo de retardo de una alarma de retardo antes de que finalice el procesamiento del OB de alarma de retardo correspondiente, el sistema operativo genera un error de tiempo (llamada del OB80).

Procesamiento de alarmas de retardo según el estado operativo

La llamada de "SRT_DINT" y, con ello, el transcurso del tiempo de retardo pueden producirse tanto en el arranque como en RUN. En cambio, los OB de alarma de retardo solo se procesan en el estado operativo RUN. Por este motivo, cuando el arranque de una alarma de retardo se produce en un OB de arranque y el tiempo de retardo ya ha transcurrido antes de que la CPU esté en RUN, la llamada del OB de alarma de retardo correspondiente se retrasa hasta que la CPU esté en RUN.

El arranque en frío y el arranque en caliente borran cualquier evento de arranque de un OB de alarma de retardo.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OBs de alarma de retardo (de OB 20 a OB 23) (Página 1827)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - STRT_INF=B#16#2F, si el número de OB es > 122
  - OB_NUMBR=B#16#FF, si el número de OB es > 254

- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
Sign	WORD	Identificador de usuario: parámetro de entrada SIGN de la llamada de la instrucción "SRT_DINT"

Consulte también

SRT_DINT: Iniciar alarma de retardo (Página 4899)

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OBs de alarma cíclica

Descripción

Con ayuda del OB de alarma cíclica, se pueden arrancar partes del programa en intervalos equidistantes.

Los momentos de arranque de un OB de alarma cíclica resultan del siguiente modo de su intervalo y su desfase:

Momentos de arranque = n * intervalo + desfase (n = 0, 1, 2...)

- El intervalo es el tiempo que transcurre entre dos llamadas. Es un múltiplo entero del ciclo básico de 1 µs.
- El desfase es el intervalo de tiempo en el que se desplazan los momentos de arranque respecto a los múltiplos del intervalo.
  Un desfase puede ser útil al utilizar varios OB de alarma cíclica. Si sus intervalos tienen múltiplos comunes, se puede utilizar un desfase para evitar que los momentos de arranque coincidan.

Ejemplo de utilización de un desfase

En su programa se utilizan dos OB de alarma cíclica: uno con un intervalo de 20 ms y otro con un intervalo de 100 ms. Para garantizar que, para los múltiplos enteros de 100 ms, estos dos OB no sean llamados de forma simultánea se utiliza un desfase.

Llamada de OB de error en relación con OB de alarma cíclica

El tiempo de ejecución de un OB de alarma cíclica debe ser muy inferior a su intervalo. De lo contrario, puede ocurrir que el intervalo de una alarma cíclica transcurra sin que se haya procesado el OB de alarma cíclica correspondiente. En ese caso, el sistema operativo genera un error de tiempo (llamada del OB80). A continuación, se reactiva o se rechaza la alarma cíclica que ha causado el error. Solo puede reactivarse un OB de alarma cíclica para cada prioridad.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OBs de alarma cíclica (de OB 30 a OB 38) (Página 1828)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - STRT_INF=B#16#3F, si el número de OB es > 122 y el desfase y el tiempo de ciclo se indican en ms
  - OB_NUMBR=B#16#FF, si el número de OB es > 254
- Información de arranque optimizada

Nombre	Tipo de datos	Significado
Initial_Call	BOOL	=TRUE en la primera llamada de este OB • En la transición de STOP o PARADA a RUN • Tras recargar
Event_count	INT	Número de eventos de arranque rechazados desde el último arranque de este OB

Consulte también

Parametrizar OB de alarma cíclica (Página 1989)

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OBs de alarma de proceso

Descripción

Los OB de alarma de proceso le ayudan a reaccionar a los eventos de los módulos aptos para alarmas de proceso que disparan una de estas alarmas. Mediante la configuración se establece de qué eventos se trata.

Solo se puede asociar un único OB de alarma de proceso a cada evento que dispara una de estas alarmas. En cambio, a un OB de alarma de proceso se le pueden asignar varios eventos (en un caso extremo, todos los eventos de un módulo que disparan alarmas de proceso).

En los módulos S7-1500, cada canal puede disparar alarmas de proceso individuales (p. ej., una entrada digital en caso de flanco ascendente).

Funcionamiento de un OB de alarma de proceso

Cuando un módulo o submódulo dispara una alarma de proceso, el sistema operativo determina el evento y el OB de alarma de proceso correspondientes. Si el OB que está activo en ese momento tiene una prioridad inferior a este OB de alarma de proceso, se arranca el OB de alarma de proceso. En caso contrario, el OB de alarma de proceso se pone en la lista de espera que corresponde según su prioridad. La alarma de proceso se acusa tras finalizar la ejecución del OB de alarma de proceso correspondiente.

Si desde que se identifica hasta que se acusa una alarma de proceso se presenta nuevamente un evento de proceso en el mismo módulo o submódulo, se aplicará lo siguiente:

- Si el evento se presenta en el canal que ha disparado la alarma de proceso actual, la alarma se perderá. En este canal solo se podrá disparar una nueva alarma de proceso tras haberse acusado la alarma de proceso actual.
- Si el evento se presenta en un canal diferente, se disparará una alarma de proceso.

Para la periferia descentralizada puede haber limitaciones adicionales, p. ej., que un módulo o submódulo solo pueda disparar una alarma de una categoría concreta hasta que se acuse la alarma actual.

Los OB de alarma de proceso solo se llaman en el estado operativo RUN de la CPU.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400: las variables USI, IChannel y EventType (significado expuesto en la Información de arranque optimizada) se copian tras "OB40_POINT_ADDR". Consulte OBs de alarma de proceso (de OB 40 a OB 47) (Página 1830)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - STRT_INF=B#16#41
  - OB_NUMBR=B#16#FF, si el número de OB es > 254

- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del módulo que dispara la alarma de proceso
USI	WORD	Identificador para ampliaciones futuras (irrelevante para el usuario)
IChannel	USINT	Número del canal que dispara la alarma de proceso
EventType	BYTE	Identificador del tipo de evento al que pertenece el evento que ha disparado la alarma (p. ej. flanco ascendente) Este identificador se indica en la descripción del módulo correspondiente.

Consulte también

Parametrizar OBs de alarma de proceso (Página 1990)

Eventos y OBs (Página 1924)

Ejemplo de un evento de alarma de proceso (Página 1927)

Información importante sobre identificadores de hardware (Página 917)

Información de arranque del OB (Página 1961)

OB de alarma de sincronismo

Descripción

Los OB de alarma de sincronismo permiten arrancar partes del programa de modo isócrono al ciclo DP o al tiempo de ciclo de emisión PN.

La memoria imagen parcial de proceso de las entradas y la correspondiente memoria imagen parcial de proceso de las salidas se actualizan llamando a las instrucciones SYNC_PI y SYNC_PO en el OB de alarma de sincronismo.

⚠ PRECAUCIÓN

Accesos directos

En los accesos directos, evite acceder a áreas de la periferia que tengan asignadas memorias imagen parcial de proceso vinculadas a OB de alarma de sincronismo.

Nota

Efecto del parámetro OB "Entrada en el búfer de diagnóstico al desbordarse los eventos"

La activación de la casilla de verificación "Entrada en el búfer de diagnóstico al desbordarse los eventos" no tiene efecto en los parámetros OB hasta que no se hayan cargado las propiedades de OB en la CPU y esta haya efectuado después una transición de STOP a RUN.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OBs de alarma de sincronismo (de OB 61 a OB 64) (Página 1844)
  A diferencia de la asignación para la información de arranque en las CPU S7-300 y S7-400 se aplica lo siguiente:
  - STRT_INF=B#16#6F, si el número de OB es > 122
  - OB_NUMBR=B#16#FF, si el número de OB es > 254

- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
Initial_Call	BOOL	=TRUE en la primera llamada de este OB durante la transición de STOP a RUN
PIP_Input	BOOL	TRUE: la memoria imagen de proceso correspondiente de las entradas es actual
PIP_Output	BOOL	TRUE: la memoria imagen de proceso correspondiente de las salidas se transfirió a tiempo a las salidas tras el último ciclo
IO_System	USINT	Número del sistema de periferia descentralizada que ha disparado la alarma
Event_Count	INT	• = n: número de ciclos perdidos • = -1: Se ha perdido un número desconocido de ciclos (p. ej. porque el ciclo ha cambiado).
SyncCycleTime	LTIME	Tiempo de ciclo calculado

Consulte también

Parametrizar OB de alarma de sincronismo (Página 1991)

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de servo MC

Descripción

Si se crea un objeto tecnológico para S7-1500 Motion Control, se creará automáticamente el bloque de organización "MC-Servo [OB91]" para el procesamiento de los objetos tecnológicos. La funcionalidad Motion Control de los objetos tecnológicos genera un nivel de ejecución propio y es llamada de acuerdo con el ciclo de aplicación en el sistema de ejecución de SIMATIC S7-1500.

El OB de servo MC está protegido contra escritura. El contenido no puede modificarse.

Dentro del OB MC-Servo se calculan los algoritmos de la regulación de posición de todos los objetos tecnológicos configurados en la CPU para Motion Control.

Se pueden ajustar el ciclo de aplicación y la prioridad del bloque de organización según sus requisitos de calidad de regulación y carga del sistema.

Ciclo de aplicación

El ciclo de aplicación en el que se llama al OB MC-Servo puede ajustarse en las propiedades del bloque de organización:

- Síncrono al bus
  El OB MC-Servo se llama de forma síncrona a un sistema de bus. El tiempo de ciclo de emisión se ajusta en las propiedades del sistema de bus seleccionado. Se puede elegir entre los siguientes sistemas de bus:
  - PROFIBUS DP isócrono
  - PROFINET IO isócrono

  El OB MC-Servo no puede llamarse de forma síncrona a un sistema de bus conectado a la CPU mediante un procesador de comunicaciones / módulo de comunicación (CP / CM).
- Cíclico
  El OB MC-Servo se llama de forma cíclica con el ciclo de aplicación indicado.

El ciclo de aplicación debe elegirse lo suficientemente grande como para procesar todos los objetos tecnológicos para Motion Control en un solo ciclo. Si no se mantiene el ciclo de aplicación, se producen desbordamientos. Si se produce un desbordamiento del OB de servo MC, la CPU pasa al estado operativo STOP.

Reducción de ciclo (a partir de CPU V1.5)

El ciclo de aplicación del OB MC-Servo puede reducirse al tiempo de ciclo de emisión de un sistema PROFINET IO isócrono. Como factor puede ajustarse un múltiplo entero del tiempo de ciclo de emisión. Para el ciclo de aplicación son posibles valores de hasta 14 veces el tiempo de ciclo de emisión (como máximo 32 ms).

Si se llaman un OB de alarma de sincronismo y el OB MC-Servo de forma síncrona al propio sistema PROFINET IO, debe ajustarse el mismo ciclo de aplicación en ambos bloques de organización.

Consulte también

Parametrizar el OB MC-Servo (Página 1993)

Eventos y OBs (Página 1924)

OB de interpolador MC

Descripción

Si se crea un objeto tecnológico para S7-1500 Motion Control, se creará automáticamente el bloque de organización "MC-Interpolator [OB91]" para el procesamiento de los objetos tecnológicos. La funcionalidad Motion Control de los objetos tecnológicos genera un nivel de ejecución propio y es llamada de acuerdo con el ciclo de aplicación del OB MC-Servo en el sistema de ejecución de SIMATIC S7-1500.

El OB de interpolador MC está protegido contra escritura. El contenido no puede modificarse.

Dentro del OB MC-Interpolator se realiza la evaluación de las instrucciones de Motion Control, así como la vigilancia y la generación de consignas para todos los objetos tecnológicos configurados en la CPU (Motion Control).

El OB de interpolador MC es llamado al finalizar el procesamiento del OB de servo MC. La relación de ciclos entre el OB de interpolador MC y el OB de servo MC es siempre 1:1.

Comportamiento de ejecución

El ciclo de aplicación del OB MC-Servo debe elegirse lo suficientemente grande como para procesar todos los objetos tecnológicos para Motion Control en un solo ciclo de aplicación. Si no se mantiene el ciclo de aplicación, se producen desbordamientos. Tenga en cuenta lo siguiente en relación con los desbordamientos:

- La CPU tolera como máximo tres desbordamientos consecutivos del OB MC-Interpolator.
- La ejecución de un OB de interpolador MC puede ser interrumpida como máximo por una llamada del OB de servo MC.

Si se producen más desbordamientos o interrupciones, la CPU pasará al estado operativo STOP.

Consulte también

Parametrizar el OB MC-Servo (Página 1993)

Eventos y OBs (Página 1924)

OB de error de tiempo

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de error de tiempo si se produce uno de los siguientes errores:

- Primer rebase por exceso del tiempo de ciclo dentro de un ciclo

  Nota

  Si el tiempo de ciclo se rebasa por exceso dos veces en un mismo ciclo, la CPU pasa al estado STOP. Esto puede evitarse llamando la instrucción "RE_TRIGR" en el punto adecuado.

- El número de peticiones solicitadas por un OB que no han terminado de procesarse ha alcanzado el límite de advertencia configurado. (En los programas migrados este límite de advertencia está ajustado a 1.)
  Si ha configurado el límite de advertencia a 1, significa lo siguiente: Si se produce un evento de arranque, p. ej. para un OB de alarma cíclica, antes de que finalice el procesamiento anterior de ese mismo OB, el sistema operativo llama el OB80.
  Esto se aplica a todos los eventos cuyo OB puede causar un error de tiempo.

- Alarma horaria expirada al reanudar en RUN después de que la CPU estuviera, p. ej., en PARADA.
- Alarma horaria omitida debido a un adelanto de la hora de más de 20 s (también en el cambio de horario de invierno a verano)

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de error de tiempo (OB 80) (Página 1845)
- Información de arranque optimizada

Nombre	Tipo de datos	Significado
Fault_ID	BYTE	Código de error. Valores posibles: • B#16#01: rebase por exceso del tiempo de ciclo • B#16#02: el OB solicitado todavía está procesándose. • B#16#05: alarma horaria expirada debido a salto de hora • B#16#06: alarma horaria expirada al reanudar en RUN después de PARADA • B#16#07: rebase por exceso del búfer de solicitud de OB para la clase de prioridad actual • B#16#08: error de tiempo en alarma de sincronismo • B#16#09: pérdida de alarma por sobrecarga de alarmas • B#16#0B: error de tiempo en alarma de sincronismo tecnológica
Csg_OBnr	OB_ANY	Número del OB que se estaba procesando cuando se produjo el error de tiempo
Csg_Prio	UINT	Prioridad del OB que se estaba procesando cuando se produjo el error de tiempo

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de alarma de diagnóstico

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de alarma de diagnóstico en los siguientes casos:

- Un módulo S7-1500 diagnosticable detecta una modificación de su estado de diagnóstico y envía a la CPU una solicitud de alarma de diagnóstico.
- Un módulo diagnosticable S7-300 o S7-400 en el que se ha habilitado la alarma de diagnóstico detecta una modificación de su estado de diagnóstico y envía a la CPU una solicitud de alarma de diagnóstico.

- Se produce un evento a raíz de un error en la fuente de alimentación o el respaldo por batería.

  Nota

  Un evento de este tipo en las CPU S7-300 y S7-400 causa la llamada del OB de error de fuente de alimentación.

- El sistema operativo ha detectado errores de memoria.

  Nota

  Un evento de este tipo en las CPU S7-300 y S7-400 causa la llamada del OB de error de hardware en CPU.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de alarma de diagnóstico (OB 82) (Página 1851)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
IOstate	WORD	Estado del objeto de hardware: • Bit 0: bueno • Bit 1: desactivado • Bit 2: mantenimiento necesario • Bit 3: mantenimiento solicitado • Bit 4: error • Bit 5: no accesible • Bit 6: cualificado • Bit 7: no disponible
LADDR	HW_ANY	Identificador de hardware del objeto de hardware que ha disparado la alarma de diagnóstico
Canal	UINT	Número de canal
MultiError	BOOL	=TRUE, si existe más de un error

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de cambio de módulo

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de cambio de módulo al insertar o extraer un módulo o submódulo de la periferia descentralizada configurado y no desactivado.

Nota

La inserción o extracción de un módulo central lleva a un STOP de la CPU.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de cambio de módulo (OB 83) (Página 1854)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LADDR	HW_IO	Identificador de hardware del módulo o submódulo afectado
Event_Class	BYTE	• B#16#38: (sub)módulo insertado • B#16#39: (sub)módulo extraído o no responde
Fault_ID	BYTE	Código de error (valores posibles: B#16#51, B#16#54, B#16#55, B#16#56, B#16#57, B#16#58, B#16#61, B#16#63, B#16#64, B#16#65, B#16#66)

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB de cambio de módulo.

ev_class B#16# ...	fault_id B#16# ...	Significado
39	51	Módulo PROFINET IO extraído
39	54	Submódulo PROFINET IO extraído
38	54	Submódulo PROFINET IO insertado y coincidente con el submódulo parametrizado
38	55	Submódulo PROFINET IO insertado, pero no coincidente con el submódulo parametrizado
38	56	Submódulo PROFINET IO insertado, pero error en parametrización del módulo
38	57	Submódulo o módulo PROFINET IO insertado, pero con fallo o mantenimiento
38	58	Submódulo PROFINET IO, error de acceso eliminado
39	61	Módulo extraído o no responde
38	61	Módulo insertado, tipo de módulo correcto
38	63	Módulo insertado, pero tipo de módulo incorrecto
38	64	Módulo insertado, pero defectuoso (identificador de módulo ilegible)

ev_class B#16# ...	fault_id B#16# ...	Significado
38	65	Módulo enchufado, pero error en parametrización del módulo
39	66	Módulo no accesible, error de tensión de carga
38	66	Módulo de nuevo accesible, error de tensión de carga eliminado

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de fallo de rack

Descripción

El sistema operativo de la CPU S7-1500 llama al OB86 en los siguientes casos:

- Se detecta el fallo de un sistema maestro DP o un sistema PROFINET IO (tanto con evento entrante como con evento saliente).
- Se detecta el fallo de un esclavo DP o un dispositivo IO (tanto con evento entrante como con evento saliente).
- Se ha activado o desactivado un esclavo DP o un dispositivo IO con la instrucción "D_ACT_DP".
- Se detecta el fallo de una parte de los submódulos de un I-Device PROFINET.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de fallo del rack (OB 86) (Página 1864)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
LADDR	HW_Device	Identificador de hardware del objeto de hardware defectuoso
Event_Class	BYTE	• B#16#32: activación de un esclavo DP o un dispositivo IO • B#16#33: desactivación de un esclavo DP o un dispositivo IO • B#16#38: evento saliente • B#16#39: evento entrante
Fault_ID	BYTE	Código de error (valores posibles: B#16#C3, B#16#C4, B#16#C5, B#16#C6, B#16#C7, B#16#C8, B#16#C9, B#16#CA, B#16#CB, B#16#CC, B#16#CD, B#16#CE, B#16#CF, B#16#F8, B#16#F9)

La tabla siguiente muestra cuál es el evento que ha provocado el arranque del OB86.

ev_class B#16# ...	fault_id B#16# ...	Significado
39	C3	Fallo de un sistema maestro DP
39/38	C4	Fallo/retorno de un esclavo DP
38	C5	Retorno de un esclavo DP, aunque defectuoso
38	C6	Retorno del aparato de ampliación, pero error en parametrización del módulo
38	C7	Retorno de un esclavo DP, pero hay un fallo en la parametrización del módulo
38	C8	Retorno de un esclavo DP, pero con diferencias entre la configuración teórica y la configuración real
32/33	C9	Activación/desactivación de un esclavo DP con la instrucción "D_ACT_DP"
39	CA	Fallo de un sistema PROFINET IO
39/38	CB	Fallo/retorno de un dispositivo PROFINET IO
38	CC	Retorno de un dispositivo PROFINET IO con fallo o mantenimiento
38	CD	Retorno de un dispositivo PROFINET IO, pero la configuración teórica difiere de la configuración real
38	CE	Retorno de un dispositivo PROFINET IO, pero error en la parametrización del módulo
32/33	CF	Activación/desactivación de un dispositivo IO con la instrucción "D_ACT_DP"
39/38	F8	Fallo/retorno de una parte de los submódulos de un I-Device PROFINET
38	F9	Retorno de una parte de los submódulos de un I-Device PROFINET con diferencias en la configuración de dispositivos

Consulte también

Eventos y OBs (Página 1924)

Información de arranque del OB (Página 1961)

OB de error de programación

Requisito para la llamada del OB de error de programación

Para que la CPU S7-1500 pueda llamar al OB de error de programación es imprescindible que se haya configurado el tratamiento de errores global.

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de error de programación si se produce un error de este tipo durante el procesamiento de una instrucción del programa de usuario. El OB de error de programación se procesa con la prioridad que se haya configurado.

Nota

Si se ha configurado el tratamiento de errores local, cuando se produzca un error de programación se aplicará el tratamiento de errores programado en el bloque correspondiente.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de error de programación (OB 121) (Página 1880)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
BlockNr	UINT	Número del bloque en el que se ha producido el error de programación
Reaction	USINT	• 0: ignorar error • 1: reemplazar valor incorrecto • 2: saltar comando • 3: Tratamiento de errores programado, disparado p. ej. por un acceso a campo con índice inadmisible o por un error en la transferencia de valores de parámetro de un FC o FB
Fault_ID	BYTE	Código de error (valores posibles: B#16#00, B#16#03, B#16#04, B#16#05, B#16#20, B#16#21, B#16#22, B#16#23, B#16#24, B#16#25, B#16#26, B#16#27, B#16#28, B#16#29, B#16#2C, B#16#30, B#16#31, B#16#32, B#16#33, B#16#34, B#16#35, B#16#38, B#16#39, B#16#3A, B#16#3B, B#16#3C, B#16#3D, B#16#3E, B#16#3F, B#16#50, B#16#51, B#16#75, B#16#76, B#16#A1, B#16#A2)
BlockType	USINT	Tipo de bloque en el que se ha producido el error: • OB: 1 • FC: 2 • FB: 3 • SFC: 4 • SFB: 5 • DB: 6
Area	USINT	Área en la que se ha producido el acceso incorrecto: • Datos locales: B#16#40 a 4E, 86, 87, 8E, 8F, C0 a CE • Memoria imagen de proceso de las entradas: B#16#01 • Memoria imagen de proceso de las salidas: B#16#02 • DB de tecnología: B#16#04 • E: B#16#81 • A: B#16#82 • M: B#16#83 • DB: B#16#84, 85, 8A, 8B
DBNr	Block_DB	DB-Nr, si AREA = DB o DI
Csg_OBNr	OB_ANY	N.º de OB (121)
Csg_Prio	USINT	Prioridad de OB
Width	USINT	Tipo de acceso en el que se ha producido el error: • Bit: B#16#00 • Byte: B#16#01 • Word: B#16#02 • DWord: B#16#03 • LWord: B#16#04

La tabla siguiente explica el significado del código de error:

fault_ID B#16# ...	Significado
00	Superada la profundidad de anidamiento máxima de las llamadas de bloque
03	Ha utilizado un puntero NULL para direccionar un operando.
04	Instrucción desconocida
05	Instrucción de parada
20	El string direccionado tiene una indicación de longitud errónea.
21	Error de conversión BCD
22	Error de longitud de área durante lectura
23	Error de longitud de área durante escritura
24	Error de área durante lectura
25	Error de área durante escritura
26	Error en el n.º del temporizador
27	Error en el n.º del contador
28	Acceso de lectura a un byte, palabra o palabra doble con un puntero cuya dirección de bit es diferente de cero
29	Acceso de escritura a un byte, palabra o palabra doble con un puntero cuya dirección de bit es diferente de cero
2C	Ha utilizado un puntero NULL para direccionar un operando.
30	Acceso de escritura a un DB global protegido contra escritura
31	Acceso de escritura a un DB de instancia protegido contra escritura
32	Número erróneo de DB al acceder a un DB global
33	Número erróneo de DB al acceder a un DB de instancia
34	Error de números en la llamada FC
35	Error de números en la llamada FB
38	El bloque de datos con seguridad de tipos no está cargado.
39	El bloque de datos no está definido con seguridad de tipos.
3A	Acceso a un DB no cargado; el número de DB está dentro del rango admisible
3B	El DB no existe
3C	Acceso a una FC no cargada; el número de FC está dentro del rango admisible
3D	Acceso a una SFC no cargada; el número de la SFC está dentro del rango admisible
3E	Acceso a un FB no cargado; el número de FB está dentro del rango admisible
3F	Acceso a un SFB no presente; el número de SFB está dentro del rango admisible
50	El bloque de datos con seguridad de tipos no está cargado.
51	El bloque de datos no está definido con seguridad de tipos.
75	Se ha superado la profundidad máxima de anidamiento para llamadas de bloque.
76	Se ha superado el máximo de datos locales disponibles.
A1	Acceso de escritura a una variable protegida contra escritura
A2	Acceso a una variable que contiene un valor numérico no válido

Consulte también

Principios básicos del tratamiento de errores (Página 2932)

Eventos y OBs (Página 1924)

Principios del tratamiento de errores local (Página 2934)

Información de arranque del OB (Página 1961)

OB de error de acceso a la periferia

Requisito para la llamada del OB de error de acceso a la periferia

Para que la CPU S7-1500 pueda llamar al OB de error de acceso a la periferia es imprescindible que se haya configurado el tratamiento de errores global.

Descripción

El sistema operativo de la CPU S7-1500 llama al OB de error de acceso a la periferia si se produce un error en el acceso directo a los datos de la periferia durante el procesamiento de una instrucción del programa de usuario. Es el caso, p. ej., de un error de lectura en el acceso directo a datos de un módulo de entrada. El OB de error de acceso a la periferia se procesa con la prioridad que se haya configurado.

Nota

Si se ha configurado el tratamiento de errores local, cuando se produzca un error de acceso a la periferia se aplicará el tratamiento de errores programado en el bloque correspondiente.

Estructura de la información de arranque

- Como en las CPU S7-300 y S7-400
  Consulte OB de error de acceso a la periferia (OB 122) (Página 1883)
- Información de arranque optimizada:

Nombre	Tipo de datos	Significado
BlockNr	UINT	Número del bloque en el que se ha producido el error de acceso a la periferia
Reaction	USINT	0: ignorar error, 1: reemplazar valor incorrecto, 2: saltar comando
Fault_ID	BYTE	Código de error: • B#16#42: error de acceso a la periferia, lectura • B#16#43: error de acceso a la periferia, escritura

Nombre	Tipo de datos	Significado
BlockType	USINT	Tipo de bloque en el que se ha producido el error: • OB: 1 • FC: 2 • FB: 3 • SFC: 4 • SFB: 5 • DB: 6
Area	USINT	Identificador del área en la que se ha producido el acceso incorrecto: • B#16#01: acceso directo a la entrada • B#16#02: acceso directo a la salida • B#16#81: acceso a la memoria imagen de proceso de las entradas • B#16#82: acceso a la memoria imagen de proceso de las salidas
DBNr	Block_DB	no relevante para el usuario
Csg_OBNr	OB_ANY	N.º de OB (122)
Csg_Prio	USINT	Prioridad de OB
Width	USINT	Tipo de acceso en el que se ha producido el error: • Bit: B#16#00 • Byte: B#16#01 • Word: B#16#02 • DWord: B#16#03 • LWord: B#16#04

Consulte también

Principios básicos del tratamiento de errores (Página 2932)
Eventos y OBs (Página 1924)
Principios del tratamiento de errores local (Página 2934)
Información de arranque del OB (Página 1961)

Ajuste de parámetros de bloques de organización

Parametrizar OB de alarma horaria

Procedimiento para configurar los parámetros

Para ajustar los parámetros de un OB de alarma horaria, proceda del siguiente modo:

1. Abra el diálogo "Propiedades" del OB de alarma horaria en cuestión.
2. Haga clic en el grupo "Alarma horaria" de la navegación local.

Resumen de los parámetros ajustables

Se pueden configurar los parámetros siguientes:

- Activo
- Ejecución
- Fecha de inicio y hora
- Botones de opción "Hora local" y "Hora del sistema"

Parámetro "Activo"

Si se activa la casilla de verificación, la alarma horaria se activa de forma automática tras un arranque.

Parámetro "Ejecución"

La lista desplegable "Ejecución" permite indicar con qué periodicidad debe ejecutarse el OB de alarma horaria. Los intervalos se basan en los ajustes de "Fecha de inicio" y "Hora".

Están disponibles los siguientes valores de "Ejecución":

- nunca
- una sola vez
- una vez por minuto
- una vez por hora
- una vez al día
- una vez por semana
- una vez al mes
- una vez al año
- a final de mes

Nota

En el valor "a final de mes", el valor indicado bajo "Fecha de inicio" es irrelevante.

Parámetros "Fecha de inicio" y "Hora"

Permiten establecer en qué momento debe ejecutarse el OB de alarma horaria por primera vez.

Ejemplo: Fecha de inicio = 05.01.2013, hora =11:16

En función del valor del parámetro "Ejecución", la CPU genera otras alarmas horarias periódicas. La hora de inicio se refiere, según la configuración, bien a la hora local, bien al tiempo universal coordinado UTC.

Nota

Si se configura el parámetro "Ejecución" como "mensual", los días 29, 30 y 31 no se pueden seleccionar como fecha de inicio. Si el OB de alarma horaria debe iniciarse a final de mes, seleccione en su lugar "a final de mes" en el parámetro "Ejecución".

"Hora local" u "Hora del sistema"

Permite definir a qué hora hace referencia la hora de inicio del OB de alarma horaria:

- "Hora local": la hora de inicio se refiere a la zona horaria configurada en la CPU.
- "Hora del sistema": la hora de inicio se refiere al tiempo universal coordinado UTC (Universal Time Coordinated).

Consulte también

OBs de alarma horaria (Página 1964)

Parametrizar OB de alarma cíclica

Procedimiento para configurar los parámetros

Para ajustar los parámetros de un OB de alarma cíclica, proceda del siguiente modo:

1. Abra el diálogo "Propiedades" del OB de alarma cíclica en cuestión.
2. Haga clic en el grupo "Alarma cíclica" de la navegación local.

Resumen de los parámetros configurables en el grupo "Alarma cíclica"

En el grupo "Alarma cíclica" pueden configurarse los siguientes parámetros:

- Tiempo de ciclo
- Desfase

Parámetro "Tiempo de ciclo"

El parámetro "Tiempo de ciclo" permite configurar el intervalo de tiempo entre dos llamadas del OB de alarma cíclica. Es un múltiplo entero de 1 µs.

Parámetro "Desfase"

Permite configurar el único intervalo de tiempo en el que se desplazan los tiempos de arranque respecto a los múltiplos del tiempo de ciclo.

Un desfase puede ser útil al utilizar varios OB de alarma cíclica. Si sus tiempos de ciclo tienen múltiplos comunes, se puede utilizar un desfase para evitar que los momentos de arranque coincidan.

Prioridad del OB de alarma cíclica

Al agregar un nuevo OB de alarma cíclica a un proyecto, indique su tiempo de ciclo. Al hacerlo, la prioridad del OB de alarma cíclica se predetermina de acuerdo con la siguiente tabla.

Tiempo de ciclo	Prioridad predeterminada
0,25 ms <= tiempo de ciclo <= 2 ms	17
2 ms < tiempo de ciclo <= 5 ms	16
5 ms < tiempo de ciclo <= 10 ms	14
10 ms < tiempo de ciclo <= 100 ms	13
100 ms < tiempo de ciclo <= 200 ms	12
200 ms < tiempo de ciclo <= 500 ms	11
500 ms < tiempo de ciclo <= 1000 ms	10
1000 ms < tiempo de ciclo <= 2000 ms	9
Tiempo de ciclo > 2000 ms	8

La prioridad predeterminada puede modificarse como sigue:

1. Abra el diálogo "Propiedades" del OB de alarma cíclica en cuestión.
2. Haga clic en el grupo "Atributos" de la navegación local.
3. Introduzca el valor que desee en el área "Prioridad" del campo de entrada "Prioridad".

Consulte también

OBs de alarma cíclica (Página 1972)

Parametrizar OBs de alarma de proceso

Procedimiento para configurar los eventos de arranque de un OB de alarma de proceso

Cada evento de arranque de un OB de alarma de proceso debe configurarse en las propiedades del módulo correspondiente. Proceda del siguiente modo:

1. Abra el diálogo "Propiedades" del módulo correspondiente.
2. En la navegación local, haga clic en el canal que deba poder disparar una alarma de proceso.

3. Marque las casillas de verificación cuyos eventos deban disparar una alarma de proceso. Si es necesario, defina los límites correspondientes.
4. Introduzca un nombre de evento, p. ej., "Flanco ascendente 0".
5. Seleccione en la lista desplegable "Alarma de proceso" un OB de alarma de proceso ya existente o cree uno nuevo.
6. Si es necesario, indique la prioridad del OB de alarma de proceso.

Para aquellos eventos que se identifican claramente por el nombre, se crea automáticamente una constante de sistema del tipo de datos "Event_HwInt". Esta se muestra en la tabla de variables estándar en la ficha "Constantes del sistema".

Resumen de los eventos de arranque de un OB de alarma de proceso

Para visualizar los eventos de arranque configurados actualmente para un OB de alarma de proceso, proceda del siguiente modo:

1. Abra el diálogo "Propiedades" del OB de alarma de proceso en cuestión.
2. Haga clic en el grupo "Eventos de arranque" de la navegación local.

Consulte también

OBs de alarma de proceso (Página 1973)

Ejemplo de un evento de alarma de proceso (Página 1927)

Parametrizar OB de alarma de sincronismo

Procedimiento para configurar los parámetros

Para ajustar los parámetros de un OB de alarma de sincronismo, proceda del siguiente modo:

1. Abra el diálogo "Propiedades" del OB de alarma de sincronismo en cuestión.
2. Haga clic en el grupo "Modo isócrono" de la navegación local.

Resumen de los parámetros ajustables

Se pueden configurar los parámetros siguientes:

- Ciclo de aplicación
- Ajuste automático
- Tiempo de retardo

Parámetro "Ciclo de aplicación"

- En PROFINET: múltiplo entero (valores posibles: 1, ... 14) del tiempo de ciclo de emisión
- En PROFIBUS DP: Ciclo DP

Casilla de verificación "Ajuste automático"

- Si la casilla de verificación está desactivada, se puede definir el tiempo de retardo de forma manual.
- Si la casilla de verificación está activada, el tiempo de retardo se calcula automáticamente. El campo de entrada correspondiente aparece sombreado.

Parámetro "Tiempo de retardo"

Nota

El parámetro "Tiempo de retardo" sólo puede definirse si la casilla de verificación "Ajuste automático" está desactivada.

El tiempo de retardo es el intervalo de tiempo que transcurre entre el tiempo de ciclo de emisión o el telegrama Global Control y el arranque del OB de alarma de sincronismo. Durante este tiempo, el controlador PROFINET IO o el maestro DP ejecutan en intercambio de datos cíclico con los dispositivos IO o los esclavos DP.

Si el tiempo de retardo es igual a cero, el OB de alarma de sincronismo se arranca con el tiempo de ciclo de emisión o el telegrama Global Control. En este caso es muy probable que un acceso a la periferia isócrona a través de la instrucción avanzada "SYNC_PI" al principio del OB no resulte, ya que los datos de la periferia aún no están disponibles.

Si el tiempo de retardo es mayor que el valor estándar, se "regalará" tiempo de ejecución del OB, ya que el acceso a la periferia isócrona ya sería posible antes.

Resumen de los parámetros mostrados

Se muestran los siguientes parámetros:

- Periferia descentralizada
- Memoria imagen parcial de proceso

"Periferia descentralizada"

Aquí se indica el sistema PROFINET IO o el sistema maestro DP cuyos dispositivos IO o esclavos DP están asignados al OB de alarma de sincronismo.

La definición de un sistema PROFINET IO o un sistema maestro DP se realiza durante la asignación entre la memoria imagen parcial de proceso y el OB de alarma de sincronismo.

"Memoria imagen parcial de proceso"

Aquí se muestra la memoria imagen parcial de proceso asignada al OB de alarma de sincronismo. Esta asignación se realizó durante la configuración de hardware.

Consulte también

OB de alarma de sincronismo (Página 1975)

Parametrizar el OB MC-Servo

Procedimiento para configurar los parámetros

Para ajustar los parámetros de un OB MC-Servo, proceda del siguiente modo:

1. Abra el cuadro de diálogo "Propiedades" del OB MC-Servo en cuestión.
2. Haga clic en el grupo "Tiempo de ciclo" de la navegación local.

Resumen de los parámetros ajustables

En el grupo "Tiempo de ciclo" se selecciona cómo se llama al OB MC-Servo en el sistema de ejecución de SIMATIC S7-1500:

- Cíclico
- Síncrono al bus

En función de la opción elegida pueden configurarse los siguientes parámetros:

- Ciclo de aplicación
- Periferia descentralizada
- Factor

Opción "Cíclico"

El OB MC-Servo se llama de forma cíclica con el ciclo de aplicación indicado.

En el campo "Ciclo de aplicación" se ajusta el ciclo en el que se llama el OB MC-Servo.

Opción "Sincronía con el bus"

El OB MC-Servo se llama de forma síncrona a un sistema de bus. El tiempo de ciclo de emisión se ajusta en las propiedades del sistema de bus seleccionado.

Para la opción "Sincronía con el bus" están disponibles los siguientes parámetros:

- **Periferia descentralizada**
  En la lista desplegable se selecciona el sistema de bus con el que se llama al OB MC-Servo de modo isócrono. Se puede elegir entre los siguientes sistemas de bus:
  - PROFINET IO isócrono
  - PROFIBUS DP isócrono

  El OB MC-Servo no puede llamarse de forma síncrona a un sistema de bus conectado a la CPU mediante un procesador de comunicaciones / módulo de comunicación (CP / CM).
- **Tiempo de ciclo de emisión (ms)**
  En este campo se muestra el tiempo ciclo de emisión o el ciclo DP ajustado en el sistema PROFINET IO seleccionado o en el sistema maestro DP.

- Factor
  El ciclo de aplicación del OB MC-Servo puede reducirse al tiempo de ciclo de emisión de un sistema PROFINET IO isócrono. En el campo "Factor" se ajusta un múltiplo entero del tiempo de ciclo de emisión. Para el ciclo de aplicación son posibles valores de hasta 14 veces el tiempo de ciclo de emisión (como máximo 32 ms).
- Ciclo de aplicación (ms)
  En este campo se muestra el ciclo de aplicación que se obtiene del tiempo de ciclo de emisión y el factor ajustado.

8.1.5.4 CPs, estaciones PC y componentes de red

CP Industrial Ethernet / PROFINET

Configuración

Protección de acceso a módulos / respaldo UDP / sistema de archivos

Referencia

Parametrización de ajustes especiales en el grupo de parámetros "Propiedades > General > Ajustes"

Dependiendo del tipo de CP, encontrará los siguientes rangos de ajuste, que se describen a continuación:

- Protección de acceso a módulos
- Respaldo UDP
- Sistema de archivos

Protección de acceso a módulos - Opción "Nivel de protección"

Esta opción permite proteger el CP de accesos involuntarios o no autorizados. En la lista desplegable pueden seleccionarse las siguientes opciones:

- No bloqueado
- Dependiente del estado
  Con este ajuste sólo se puede acceder al CP si la CPU se encuentra en estado STOP.

Desconectar el respaldo de telegramas UDP

El CP guarda en la memoria intermedia los telegramas UDP dependiendo de la carga por comunicación entre el CP y la CPU.

Con la opción Respaldo UDP (Desconectar respaldo de telegramas UDP) se puede elegir entre estos comportamientos:

- desactivado (ajuste predeterminado)
  El respaldo de telegramas UDP está activado. Los telegramas UDP recibidos por el CP se guardan en la memoria intermedia hasta que se transfieren a la CPU.
- activado
  El CP transfiere siempre el último telegrama recibido, es decir, el actual, a la CPU. Mientras no se pueda transmitir ningún telegrama UDP nuevo entre el CP y la CPU debido a la carga de comunicación actual, en el CP se guarda siempre en la memoria intermedia el último telegrama que ha llegado.

Al desactivar el respaldo de telegramas UDP se consigue un tiempo de reacción mínimo entre la llegada del telegrama UDP y su evaluación en la CPU. El comportamiento provocado al activar la opción puede ser crítico si llega una gran cantidad de telegramas en determinados casos de aplicación. Si se guardaran de manera intermedia muchos telegramas se podría producir un desfase temporal no deseado entre los telegramas registrados en la CPU y los telegramas registrados actualmente en la interfaz Ethernet.

Sistema de archivos (observar las mayúsculas y minúsculas) – Opción con CPs con funciones IT

Activado: En la asignación de nombres de archivo para RAM, el CP distingue entre mayúsculas y minúsculas dentro del nombre.

Transferencia de datos > 240 bytes

Referencia

Parametrización de ajustes especiales en el grupo de parámetros "Propiedades > General > Ajustes"

Dependiendo del tipo de CP, encontrará el rango de ajuste "Transferencia de datos > 240 bytes"

Indicaciones sobre la transferencia de datos > 240 bytes

- Los CPs con versiones recientes soportan la transferencia de datos > 240 bytes. Tenga en cuenta al respecto las indicaciones de la documentación del dispositivo CP Ethernet.
- Recuerde que con esta configuración en una S7-300 se ocupa un recurso de conexión (conexión libre para funciones S7) de la CPU S7-300. Los recursos de conexión de la CPU también son ocupados por CPs S7-300 en modo FMS o por programadoras u OPs, p. ej.

Nodo Ethernet / Interfaz PROFINET

Opciones avanzadas

Enviar Keep Alive para conexiones (sólo con conexiones TCP e ISO on TCP)

Referencia

Parametrización de ajustes especiales para la interfaz PROFINET en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Opciones avanzadas > Opciones de interfaz"

Enviar Keep Alive para conexiones (solo con conexiones TCP e ISO on TCP)

Aquí se puede ajustar el intervalo de tiempo en el que se enviarán telegramas de señal de vida (Keep Alive) al interlocutor de una conexión.

El CP Ethernet está configurado para todos los servicios relacionados con la conexión de modo que se envíen telegramas de señal de vida (Keep Alive). De este modo se garantiza que las conexiones finalicen después de que falle un interlocutor y que los recursos de conexión se liberen. El ajuste que se realice aquí servirá para todas las conexiones TCP o ISO on TCP que operen a través del CP; no es posible realizar un ajuste orientado a una conexión.

Rango de valores

- Ajuste predeterminado: 30 segundos
- Valor máximo: consulte la información visualizada en el campo de entrada
- Keep Alive desactivado: 0 segundos

Nota

Tenga en cuenta que el mecanismo de Keep Alive puede provocar que se mantengan conexiones subordinadas (p. ej. una conexión telefónica RDSI) aunque no se transfieran datos de usuario propiamente dichos. Para que esto suceda, se puede proceder del siguiente modo:

- Desactivar el mecanismo Keep Alive (intervalo =0)
  o bien
- Ajustar un intervalo de tiempo tan alto que la conexión subordinada finalice tras la ausencia de datos de usuario antes de que se envíe un telegrama Keep Alive.

Configuración IP

Referencia

Parametrización para el protocolo IP en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Direcciones Ethernet"

Significado

Aquí se especifica la vía o el procedimiento por el cual se determina y asigna la dirección IP de la interfaz local.

Mediante las variantes ofrecidas es posible asignar "dinámicamente" direcciones IP fuera de la configuración.

Por medio de la selección realizada se determina también si las conexiones se crean a través de la configuración o de la interfaz del programa de usuario (instrucción IP_CONFIG).

Las opciones siguientes están disponibles:

- Ajustar la dirección IP en el proyecto
  Este es el ajuste estándar para PLCs. La dirección IP se fija durante la conexión en red del dispositivo. De este modo, la dirección IP queda configurada de forma fija.
  Las conexiones deben configurarse con esta opción.
- Obtener la dirección IP de un servidor DHCP
  De este modo se especifica que la dirección IP se obtenga de un servidor DHCP al arrancar el dispositivo.
  Para ello, se transmite al servidor DHCP la dirección MAC de la interfaz o la ID del cliente, que puede introducirse en la configuración.
  La ID de cliente es una secuencia de 63 caracteres como máximo.
  Requisitos y limitaciones: Si se elige esta opción, en principio no es posible crear ninguna conexión completamente especificada en el proyecto, puesto que no se conoce la dirección IP local. Por consiguiente, debe seleccionarse el tipo de conexión "Sin especificar" con establecimiento pasivo.
- Ajustar la dirección IP en el programa de usuario
  De este modo se especifica que la dirección IP se determine mediante la interfaz del programa de usuario (bloque de función IP_CONFIG). Con ello, la dirección IP puede suministrarse dinámicamente durante el funcionamiento.
  En este caso de aplicación, las conexiones se crean exclusivamente a través de la interfaz del programa de usuario; la configuración de una conexión está excluida (afecta a las conexiones vía: TCP, ISO-on-TCP, UDP, ISO-Transport).
- Ajustar dirección IP en el dispositivo
  De este modo se especifica que la dirección IP se determine mediante otros servicios fuera de la configuración.
  En este caso de aplicación está excluida una configuración de la conexión (afecta a las conexiones vía: TCP, ISO-on-TCP, UDP, ISO-Transport).

Nota

Si ya se han configurado conexiones por medio de la interfaz configurada aquí y se elige un ajuste distinto al de "Ajustar la dirección IP en el proyecto", se visualiza un aviso indicando que las conexiones configuradas han dejado de funcionar.

Sincronización horaria

Sincronización horaria - Procedimiento

Referencia

Parametrización de las propiedades de interfaz en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Opciones avanzadas > Sincronización horaria".

Funcionamiento de los procedimientos relacionados con la sincronización horaria

- Procedimiento SIMATIC
  Cuando el CP recibe mensajes de hora MMS, su hora local se sincroniza, siempre que no se haya configurado el procedimiento NTP (MMS = Manufacturing Message Specification).
  La ventaja de este procedimiento radica en la mayor precisión general en comparación con el procedimiento NTP.
  Los mensajes de hora pueden ser recibidos por una CPU S7-41x o bien por LAN.
  Según donde se encuentre el reloj maestro, hay que distinguir entre los dos casos siguientes:
  - Los mensajes de hora pueden provenir de la subred (LAN) y transferirse a la estación. Para ello, la sincronización horaria de la CPU debe configurarse como esclavo.
  - Los mensajes de hora pueden provenir de la estación y transferirse a la subred (LAN). Para ello, la sincronización horaria de una CPU debe estar configurada como maestro o bien otro CP debe transferir los mensajes de hora al bus K.

  Si se utilizan varios CPs en la estación, debería considerarse el flujo de mensajes de hora en función del reloj maestro, ya que con la función aquí descrita es posible acoplar mensajes de hora de una red a otra. En una estación sólo puede haber un reloj maestro. Si en una estación hay varios CPs conectados a la misma red, sólo uno de estos CPs puede transferir los mensajes de hora.
- Procedimiento NTP (NTP: Network Time Protocol)
  Con el procedimiento NTP, el CP envía regularmente consultas de hora (en modo cliente) al servidor NTP de la subred (LAN). Sobre la base de las respuestas de los servidores se determina la hora más exacta y fiable y se sincroniza la hora de la estación.
  La ventaja de este procedimiento es la posibilidad de sincronizar la hora más allá de los límites de la subred.
  En este caso, los mensajes de hora MMS que se reciban se ignorarán.
  Deben configurarse las direcciones IP de hasta cuatro servidores NTP. El intervalo de actualización define el espacio de tiempo entre las consultas de hora (en segundos). El rango de valores del intervalo está entre 10 segundos y un día.
  En el procedimiento NTP se transfiere generalmente la hora UTC (Universal Time Coordinated), que equivale a la hora GMT (Greenwich Mean Time). Mediante la configuración de la zona horaria local se puede ajustar la diferencia de tiempo respecto de la hora UTC.

Sincronización horaria, procedimiento SIMATIC - Parametrizacíon

Referencia

Parametrización de las propiedades de interfaz en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Opciones avanzadas > Sincronización horaria"

Están seleccionados la opción "Activar sincronización horaria" y el método "Procedimiento SIMATIC".

Sincronización horaria en el procedimiento SIMATIC

Con el método "SIMATIC" se provoca que el CP reenvíe mensajes de hora MMS cuando los recibe de un emisor de hora SIMATIC. Los mensajes de hora pueden ser recibidos por el bus K (por una CPU S7-41x) o bien por LAN.

Según donde se encuentre el reloj maestro, hay que distinguir entre los dos casos siguientes:

- Los mensajes de hora pueden provenir de la subred (LAN) y transferirse a la estación.

  Nota

  Para ello es necesario configurar la sincronización horaria de la CPU (excepción: CPU 318) como esclavo.

- Los mensajes de hora pueden provenir de la estación y transferirse a la subred (LAN). Para ello, la sincronización horaria de una CPU debe estar configurada como maestro o bien otro CP debe transferir los mensajes de hora al bus K.

Si se utilizan varios CPs en la estación, debería considerarse el flujo de mensajes de hora en función del reloj maestro, ya que con la función aquí descrita es posible acoplar mensajes de hora de una red a otra. En una estación solo puede haber un reloj maestro.

Si en una estación hay varios CPs conectados a la misma subred, y si para ellos está activada la transmisión de la hora, rige: Se encarga de la transmisión el CP con el número de slot más bajo en cada caso. Todos los CPs restantes configurados para la transmisión de la hora activan la transmisión automáticamente.

Transmitir la hora - Posibilidades de ajuste del sentido:

- Automático
  El CP recibe el mensaje horario de la LAN o de la estación y lo reenvía a la estación o a la LAN.
  Si se operan varios CPs en la estación, este ajuste automático puede causar colisiones. Para evitarlo, es posible especificar el sentido de reenvío utilizando las opciones siguientes:
- De estación
- De LAN

Opción "Utilizar hora corregida"

La opción "Utilizar hora corregida" permite decidir si, dado el caso, debe tenerse en cuenta un posible factor de corrección enviado en un telegrama horario. El factor de corrección

especifica un offset de tiempo (en unidades de ½ hora), provocado por el ajuste de las zonas horarias o el ajuste del horario de verano/invierno del emisor de la hora.

Si se selecciona esta opción, el CP incluirá el factor de corrección en el cálculo de la hora, de modo que los sistemas conectados a continuación ya no tendrán que considerarlo.

Ejemplo:

En el telegrama horario se transmite como hora las 9:21. Además, el telegrama horario recibe el factor de corrección +2x0,5. Así, la hora visualizada transmitida por el CP es las 10:21.

Sincronización horaria, procedimiento NTP - Parametrizacíon

Referencia

Parametrización de las propiedades de interfaz en el grupo de parámetros "Propiedades > General > Interfaz PROFINET > Opciones avanzadas > Sincronización horaria"

Están seleccionados la opción "Activar sincronización horaria" y el método "NTP".

Procedimiento NTP (NTP: Network Time Protocol)

Con el procedimiento NTP, el módulo envía regularmente consultas de hora (en modo cliente) al servidor NTP de la subred (LAN). Sobre la base de las respuestas de los servidores se determina la hora más exacta y fiable y se sincroniza la hora de la estación.

La ventaja de este procedimiento es la posibilidad de sincronizar la hora más allá de los límites de la subred.

Deben configurarse las direcciones IP de hasta cuatro servidores NTP. El intervalo de actualización define el espacio de tiempo entre las consultas de hora (en segundos). El rango de valores del intervalo está entre 10 segundos y un día.

En el procedimiento NTP se transfiere generalmente la hora UTC (Universal Time Coordinated), que equivale a la hora GMT (Greenwich Mean Time).

Opción "Sincronización horaria al minuto completo"

La opción "Sincronización horaria al minuto completo" permite decidir si la hora debe transmitirse al bus K exactamente en el minuto completo.

Esta opción solo es necesaria para algunas aplicaciones especiales.

Opción "Transferir hora a la estación"

La opción "Transferir hora a la estación" está activada de forma predeterminada.

Algunas CPUs ofrecen la posibilidad de obtener la hora de forma autónoma de un servidor NTP. Si se utiliza esa posibilidad en la CPU, se debe desactivar la opción descrita en el CP. De ese modo se evita que la hora obtenida en la CPU directamente del servidor NTP se sobrescriba con la hora registrada en el CP. La opción de transmisión a través del CP ofrece una precisión menor.

Información adicional

Configuración de red individual

Configuración de red individual

Aquí es posible configurar la red de forma fija si es necesario. La opción predeterminada es "Configuración automática" que, por norma general, permite comunicarse correctamente.

Si surgen problemas durante la comunicación (p. ej. si no pueden establecerse conexiones o si se producen fallos frecuentes en la red) es posible que la configuración de red automática o la seleccionada no sea adecuada. En este caso, elija un ajuste de red adecuado a su configuración.

Rango de valores y ejemplo de dirección IP

Rango de valores de la dirección IP

La dirección IP consiste en 4 números decimales comprendidos en un rango de valores entre 0 y 255 y separados entre sí por un punto.

Formato: xxx.yyy.zzz.www

Ejemplo:

141.80.0.16

- Las direcciones siguientes tienen significados especiales (* indica un número cualquiera entre 0 y 255):
  - 127.*.*.* Loopback
  - 0.0.0.0 red no especificada
  - 255.255.255.255 Broadcast a todos los nodos (sólo válido si se ha seleccionado el interlocutor "Todos los nodos Broadcast")
  - Dirección de host 255 dirección Broadcast para todos los nodos de una red:
    en la clase A: xxx.255.255.255
    en la clase B: xxx.yyy.255.255
    en la clase C: xxx.yyy.zzz.255
- No se permiten:
  - Dirección de host 0
    en la clase A: xxx.0.0.0
    en la clase B: xxx.yyy.0.0
    en la clase C: xxx.yyy.zzz.0

Rango de valores y ejemplo de máscara de subred

Rango de valores de la máscara de subred

Para separar la ID de la red de la dirección de host en la dirección IP se introdujo la máscara de subred.

Dicha máscara consiste en 4 números decimales comprendidos en un rango de valores entre 0 y 255 y separados entre sí por un punto (p. ej. 255.255.0.0).

Los 4 números decimales de la máscara de subred deben incluir en su representación binaria una secuencia de valores "1" sin espacios partiendo de la izquierda y una secuencia de valores "0" sin espacios partiendo de la derecha.

Los valores "1" determinan el número de red dentro de la dirección IP. Los valores "0" determinan la dirección de host dentro de la dirección IP.

Ejemplo:

Valores correctos: 255.255.0.0 D = 1111 1111.1111 1111.0000 0000.0000 0000 B

255.255.128.0 D = 1111 1111.1111 1111.1000 0000.0000 0000 B

255.254.0.0 D = 1111 1111.1111 1110.0000 0000.0000.0000 B

Valor incorrecto: 255.255.1.0 D = 1111 1111.1111 1111.0000 0001.0000 0000 B

Relación entre la dirección IP y la máscara de subred

El primer decimal de la dirección IP (desde la izquierda) determina la estructura de la máscara de subred en cuanto al número de valores "1" (binarios), del siguiente modo ("x" representa la dirección de host o bien los "0" en la máscara de subred):

- Primer decimal de la dirección IP máscara de subred
  clase A: de 0 a 126 255.x.x.x
  clase B: de 128 a 191 255.255.x.x
  clase C: de 192 a 223 255.255.255.x

Nota

En el primer decimal de la dirección IP también puede introducirse un valor entre 224 y 255. Sin embargo, no es recomendable porque con estos valores STEP 7 no verifica la dirección.

Relación entre la dirección IP y la dirección de la transición de red

Relación entre la dirección IP y la dirección de la transición de red

La dirección IP y la dirección de la transición de red sólo pueden ser diferentes en las cifras que son "0" en la máscara de subred.

Ejemplo:

Se ha introducido: para máscara de subred 255.255.128.0; para dirección IP 141.30.0.5 y para dirección de la transición de red 141.30.128.1. La dirección IP y la dirección de la transición de red sólo pueden tener un valor diferente en el 4.º decimal. Sin embargo, en el ejemplo la 3.ª cifra ya es distinta.

Así pues, en el ejemplo debe cambiarse:

la máscara de subred por: 255.255.**0**.0 o bien

la dirección IP por: 141.30.**128**.5 o bien

la dirección de la transición de red por: 141.30.**0**.1

Parámetros de puerto

Configuración de red individual para cada puerto

En caso necesario, para cada interfaz (puerto) disponible es posible configurar la red de forma fija para las propiedades de transferencia. La opción predeterminada es "Configuración automática" que, por norma general, permite comunicarse correctamente.

Si surgen problemas durante la comunicación (p. ej. si no pueden establecerse conexiones o si se producen fallos frecuentes en la red) es posible que la configuración de red automática o la seleccionada no sea adecuada. En este caso, elija un ajuste de red adecuado a su configuración.

Funcionamiento de la conmutación automática

El CP ofrece en todos los puertos una conexión dúplex de 10/100 Mbits/s con detección automática (autosensing) y negociación automática (autonegotiation) de la configuración de red. Estas funciones se ejecutan del siguiente modo tras conectar el CP:

- El CP intenta detectar la velocidad de transferencia utilizada por el interlocutor.
- Si lo consigue, intentará negociar con el interlocutor un modo de operación dúplex óptimo.
- Si la negociación no es posible, el CP utilizará la velocidad de transferencia detectada anteriormente y el modo semidúplex.

La duración de los pasos descritos es de unos 2 segundos.

Además, la "Configuración automática" incluye un mecanismo de autocrossing; independientemente de las propiedades del dispositivo partner, podrá utilizar para la conexión un cable cruzado o no.

Conmutar a la configuración de red individual

En cuanto se realiza una configuración manual, la conmutación automática ya no es efectiva. Lo mismo es válido para el mecanismo de autocrossing integrado en el CP. Si el dispositivo partner conectado no soporta el mecanismo de autocrossing, deberá utilizarse un cable cruzado (en caso de switch) o no (en caso de terminal de datos), según sea el dispositivo partner.

Posibilidad de selección para "desactivar" un puerto:

En función del tipo de módulo, en la lista desplegable aparecerá la opción "- desactivar -". Así, por ejemplo, tiene la posibilidad de prohibir el acceso a un puerto no utilizado por motivos de seguridad.

Consulte también

Información relacionada

Tenga en cuenta las indicaciones adicionales del manual de producto del CP correspondiente.

Tamaño máximo de telegrama / Volumen de datos para dispositivos PROFINET IO

Tamaño de los datos de usuario en el CP 1616

El tamaño de los datos de usuario (datos de proceso) se calcula a partir del tamaño máximo de telegrama (volumen de datos bruto) menos los valores para los módulos insertados en este dispositivo IO. El resultado es el volumen de datos neto.

Tabla 8-92 Ejemplo de cálculo de los datos de usuario (bytes) del CP 1616 como dispositivo IO con módulos homogéneos (p. ej. solo módulos DI)

Bytes	Explicación
1434	Tamaño bruto del telegrama
-16	para el número máximo de módulos insertables
= 1418	Tamaño neto de los datos de usuario

Tabla 8-93 Ejemplo de cálculo de los datos de usuario (bytes) del CP 1616 como dispositivo IO con módulos heterogéneos (p. ej. DI/DO)

Bytes	Explicación
1434	Tamaño bruto del telegrama
-32	para el doble de módulos insertables como máximo
= 1370	Tamaño neto de los datos de usuario

Tamaño de los datos de usuario en el CP 343-1 (Lean/Standard/Advanced)

El tamaño de los datos de usuario (datos de proceso) se calcula a partir del tamaño máximo de telegrama (volumen de datos bruto) menos los valores para el CP y la propia interfaz, para el número de puertos y los módulos insertados en este dispositivo IO. El resultado es el volumen de datos neto.

Tabla 8-94 Ejemplo de cálculo de los datos de usuario (bytes) del CP 343-1 como dispositivo IO

Bytes	Explicación
548	Tamaño bruto del telegrama
-1	para el propio CP
-1	para la interfaz

Bytes	Explicación
-2	para 2 puertos
-32	para el número máximo de módulos insertables
= 512	Tamaño neto de los datos de usuario *

* Si se utilizan menos módulos pero más grandes es posible que aumente el volumen de datos de usuario.

Configuración DNS

Referencia

Parametrización de la dirección de servidor DNS en el grupo de parámetros "Propiedades > General > Configuración DNS"

Significado

En la configuración de una conexión de correo electrónico hay que especificar la dirección de un servidor de correo a través del cual se envían los mensajes. Esta dirección puede indicarse de forma absoluta o simbólica. En la indicación simbólica, la dirección absoluta se determina consultando al servidor DNS especificado.

Consulte también

Establecer una conexión de correo electrónico (Página 686)

Detalles de la dirección de correo electrónico (Página 688)

Protección de acceso IP

Protección de acceso IP

Referencia

Parametrización de las propiedades de interfaz en el grupo de parámetros "Propiedades > General > Protección de acceso IP".

Resumen

Para la protección de acceso IP pueden parametrizarse las opciones siguientes, según sea el tipo de CP:

- Opción "Activar servidor web"
  El CP le ofrece la función de un servidor web para acceder a través de un navegador web. Para ello, se almacenan determinadas páginas HTML con información del CP y funciones de diagnóstico en un área de memoria del CP.
  Active esta opción para obtener acceso a dichas páginas HTML. Con ello se habilitará el puerto 80 del CP.
  Por defecto, el acceso a través de un servidor web está activado.
- Opción "Activar servidor FTP"
  Elija esta opción si desea permitir los accesos FTP a la estación S7 a través del puerto 20/21 del CP.
  Si en la ficha "FTP" se configura el acceso FTP a los DBs file de la CPU, el acceso sólo será posible si se activa esta opción.
  Por defecto, el acceso a través de un servidor FTP por el puerto 20/21 está activado.
- Opción "Activar protección de acceso para comunicación IP"
  La interfaz Ethernet ofrece la posibilidad de limitar el acceso al dispositivo local a los interlocutores que tengan direcciones IP determinadas. De este modo, los interlocutores no autorizados no podrán acceder a los datos del dispositivo local a través de esta interfaz. Para ello se puede activar o desactivar la protección de acceso IP así como introducir direcciones IP determinadas en una lista IP Access Control (IP-ACL).
  Por defecto, la protección de acceso IP está **des**activada.
  Los intentos de acceso bloqueados se registran en el CP y pueden verse con el diagnóstico especial en el objeto de diagnóstico "Protección de acceso IP". En los CPs con función TI se creará además un archivo LOG en el sistema de archivos del CP que se puede ver con un navegador web.
  Siga las instrucciones de manejo del archivo LOG y las indicaciones para cargar los datos de configuración más adelante.

Protección de acceso IP para interlocutores configurados

Si desea restringir el acceso únicamente a los interlocutores indicados en la configuración, sólo hay que activar la protección de acceso. En este caso, no es necesario introducir ninguna dirección IP en la lista.

Los interlocutores que se toman en consideración son:

- Estaciones con las que se han configurado conexiones.
  Lo mismo es válido para conexiones (excepto conexiones S7) en las que el interlocutor está en una subred distinta.
  Sin embargo, tenga en cuenta que, en las conexiones sin especificar cuyas direcciones de interlocutor permanecen indeterminadas, todas las direcciones IP no introducidas no estarán autorizadas y, por tanto, se rechazarán. Las conexiones en PROFINET CBA también se considerarán conexiones sin especificar. Para estas conexiones deberá introducir explícitamente las direcciones IP en la lista IP Access Control.
- Dispositivos PROFINET IO cuando se utiliza el CP Ethernet como controlador PROFINET IO;
  Las direcciones IP de dispositivos PROFINET IO se introducen dinámicamente en la lista IP Access Control en estado RUN del CP.
  Si el CP pasa al estado STOP, las direcciones IP de los dispositivos PROFINET IO volverán a borrarse de la lista Access.
- Servidores NTP, servidores SMTP, servidores DNS y servidores DHCP.
  Las direcciones IP de servidores NTP, SMTP, DNS y DHCP también se introducirán y borrarán dinámicamente en los servidores cuando se realicen las consultas pertinentes.

La protección de acceso IP hace referencia a todos los tipos de conexión que se desarrollan a través del protocolo IP (TCP, ISO on TCP, UDP, S7)

Observación sobre direcciones IP asignadas dinámicamente:
Puesto que cada servicio administra su propia entrada dinámica en la ACL, es posible que la misma dirección IP aparezca varias veces en el diagnóstico de módulo.

Protección de acceso IP para interlocutores con direcciones IP concretas

Para restringir el acceso a direcciones IP concretas, introduzca dichas direcciones en la lista IP Access Control. Pueden ser, por ejemplo, las direcciones IP de interlocutores no especificados en la configuración de conexiones, de programadoras individuales o de interlocutores en PROFINET CBA.

Las direcciones IP indicadas en la configuración de conexiones están siempre incluidas en las direcciones IP admitidas y, por tanto, no deben introducirse explícitamente en la IP-ACL.

Ver el archivo LOG con un navegador web

En los CPs con función IT se crea un archivo LOG en el sistema de archivos del CP, que puede consultarse desde un navegador web. En comparación con el registro en el diagnóstico especial, el archivo LOG ofrece espacio para hasta 512 entradas.

El archivo LOG no está disponible hasta que se activa la protección de acceso IP por primera vez.

Encontrará el archivo LOG en formato de archivo HTML en el sistema de archivos del CP, en el directorio siguiente:

·ram/security/IPLogFile.htm

Otras propiedades:

El archivo LOG se crea como búfer circular; cuando se han registrado más de 512 entradas, se sobrescriben las entradas existentes, comenzando por la más antigua.

Las entradas son únicamente cronológicas; no hay ningún otro criterio de clasificación.

Recuerde que esta función solo se soporta en las versiones de firmware más recientes de los CPs; encontrará más información al respecto en la documentación de los CPs.

En los CPs Advanced a partir de CP 343-1 Advanced (GX30) y CP 443-1 Advanced (GX20) no se crea el archivo LOG. En ellos, los intentos de acceso fallidos pueden verse directamente en el diagnóstico web.

Especificación de los derechos de acceso en la lista IP Access Control (IP--ACL)

Referencia

Parametrización de las propiedades de interfaz en el grupo de parámetros "Propiedades > General > Protección de acceso IP".

Reglas para la introducción de las direcciones IP:

Tenga en cuenta las reglas siguientes al introducir las direcciones IP:

- En cada fila puede introducirse la dirección IP individualmente o como rango.
- También es posible introducir comentarios.
- El número máximo de direcciones que pueden introducirse es de 32
- El número máximo de áreas de direcciones que pueden introducirse es de 10
- No se comprueba si las direcciones introducidas individualmente también están incluidas en un área de direcciones.
- No se comprueba si un área incluye direcciones no válidas (así, por ejemplo, es posible indicar direcciones Broadcast aunque no pueden aparecer como dirección IP de un remitente).
- Los comentarios empiezan con el signo "#"

Ejemplos:

- Dirección IP individual con comentario: 141.80.0.16 #dirección de interlocutor x
- Área de direcciones IP con comentario: 141.80.0.16 - 141.80.0.25 #área de direcciones para nivel de gestión

Derechos

Los derechos de acceso aquí descritos pueden seleccionarse en la lista IP Access Control con las correspondientes abreviaciones.

Tabla 8-95 Derechos de acceso y abreviaciones

Derecho de acceso	Denominación breve	Significado / observación
Acceso a estación	A	Access Los interlocutores cuya dirección se encuentre dentro del rango especificado tienen acceso a la estación asignada al CP (CP/CPU). El derecho de acceso está activado implícitamente para las direcciones IP que se hayan indicado en la configuración de la conexión.
Modificación de la lista IP Access Control	M	Modify Los interlocutores cuyas direcciones se encuentren dentro del rango especificado tienen derecho de acceso a la lista IP Access Control. Estos interlocutores pueden enviar entradas para la lista IP Access Control al CP vía HTTP. Al seleccionar este derecho, el "A" también se activa automáticamente. Notas referentes a las entradas enviadas por HTTP: • Con cada lista enviada por HTTP se anula la lista enviada anteriormente por HTTP. • Con los derechos de acceso enviados vía HTTP es posible complementar entradas configuradas con STEP 7, pero en ningún caso borrarlas. • Una lista enviada por HTTP se borra con un fallo de la tensión de alimentación en el CP (tensión OFF). Encontrará información sobre la sintaxis de una lista enviada por HTTP en la documentación del CP.
Acceso por routing de IP a otra subred IP	R	Routing Los interlocutores cuya dirección se encuentre dentro del rango especificado tienen acceso a otras subredes conectadas al CP. El derecho de acceso no se activa automáticamente para las direcciones IP que se hayan indicado en la configuración de la conexión. Dado el caso, es necesario activar este derecho de acceso explícitamente en este punto.

Nota

- **Cargar datos de configuración a través de TCP/IP**
  Tenga en cuenta que al cargar a través de TCP/IP con la protección de acceso activada, debe incluirse en la IP-ACL la dirección IP de la programadora o el PC desde el que se carga la configuración. De lo contrario, la PG/el PC será detectado como no autorizado durante la operación de carga y éste se cancelará.
- **Cargar conexiones S7 con posterioridad**
  Si se cargan con posterioridad conexiones S7 en la estación S7, no se considerarán automáticamente en la IP-ACL. En este caso habrá que cargar de nuevo toda la configuración de la estación.
- **Aplicar y editar la configuración mediante carga en la PG/el PC**
  Si pretende editar la configuración de conexiones y ha activado la protección de acceso IP, en la PG/el PC se requerirá toda la información de la conexión. Por ello, habrá que cargar los datos de conexión actuales en la PG/el PC. La descarga posterior en la estación S7 requiere a continuación una carga completa de los datos de conexión y de la estación.

Nota

Por medio del diagnóstico especial se puede ver la IP-ACL actual en la estación S7 y, dado el caso, compararla con la IP-ACL configurada.

Editar derechos de acceso en la administración de usuarios

Referencia

Parametrización de los derechos de usuario en el grupo de parámetros "Propiedades > General > Administración de usuarios"

Significado de la administración de usuarios

Defina en la administración de usuarios qué derechos deben asignarse a qué usuarios para el acceso a la estación S7 a través de un navegador web o cliente FTP.

En una lista ordenada alfabéticamente encontrará en Nombre de usuario los usuarios especificados hasta el momento, para los que se han definido contraseñas.

La entrada "Everybody" está disponible de forma estándar. No puede modificarse. Tampoco se le puede asignar una contraseña.

Las entradas de usuario siempre están vinculadas a una contraseña. Excepción: La entrada "Everybody" no está protegida por contraseña.

Nota

En la entrada "Everybody" no hay derechos asignados por defecto. De todas formas, es posible asignarle derechos para fines de servicio.

De todas formas, hay que tener en cuenta que los derechos asignados al usuario "Everybody" están a disposición de todos los demás usuarios. Por consiguiente, es imprescindible retirar los derechos una vez terminada la asistencia técnica. En caso contrario, se concede el derecho de ejecutar servicios sin autorización en cualquier acceso.

Si hay un derecho de acceso activado bajo el usuario "Everybody", puede reconocerse porque la casilla de verificación está rellenada al asignar derechos a otro usuario.

Introducir un usuario nuevo

1. Seleccione la primera fila libre de la lista de usuarios.
   Se creará una entrada de usuario nuevo con el nivel de acceso "mínimo".
2. Adapte el nombre de usuario propuesto en la columna "Nombre".
3. Dentro de la fila recién creada, haga doble clic en la columna "Nivel de acceso".
   Aparecerá el símbolo de la lista desplegable.
4. Haga clic en dicho símbolo.
5. En el cuadro de diálogo que aparece a continuación, defina derechos específicos del usuario.
6. Confirme la entrada con "Aceptar".
7. Asigne una contraseña nueva en el cuadro de diálogo que aparece.

Modificar derechos de usuario o una contraseña:

1. En la fila de la entrada de usuario deseada, haga doble clic en la columna "Nivel de acceso".
   Aparecerá el símbolo de la lista desplegable.
2. Haga clic en dicho símbolo.
3. En el cuadro de diálogo que aparece a continuación, modifique los derechos específicos del usuario o haga clic en el botón "Contraseña" para asignar una contraseña nueva.
4. Confirme la entrada con "Aceptar".
5. Introduzca la contraseña en el cuadro de diálogo que aparece.

Borrar entradas de usuario:

1. En la fila de la entrada de usuario deseada, elija el comando "Borrar" del menú contextual.

Declaración de variables - Editar la lista de variables

Referencia

Parametrización de las variables configuradas en la CPU en el grupo de parámetros "Propiedades > General > Declaración de variables"

Significado de la declaración de variables

Para poder acceder a variables de las CPUs a través de un navegador web vía Java Applet es necesario indicar al CP los nombres, las direcciones y los derechos de acceso de las variables.

En la lista de símbolos de la declaración de variables se pueden ver las variables conocidas en ese momento con sus nombres simbólicos, los operandos correspondiente y los derechos de acceso asignados.

Agregar una entrada de variable nueva

1. Haga doble clic en la primera fila vacía de la tabla.
   Aparecerá el símbolo de selección por medio de una lista desplegable.
2. Seleccione el símbolo para abrir el visor de estructura para las variables del dispositivo actual.
3. Elija los símbolos de las variables o los elementos de estructura a los que se debe poder acceder desde un navegador web.
   Observación: los símbolos deben haberse almacenado previamente en el editor de símbolos.

Borrar una entrada de variable

1. En la fila de la variable que se va a borrar, elija el comando "Borrar" del menú contextual.
   Atención:

   Nota

   no se realiza ninguna consulta de seguridad, la entrada se borra inmediatamente. No es posible seleccionar varias entradas.

Configurar derechos de acceso - Opción "Escribible"

En el ajuste predeterminado, para las variables solo se otorgan derechos de lectura.

Para asignar los derechos de escritura/lectura para las variables introducidas aquí, proceda del siguiente modo:

1. Seleccione una o más variables en la lista de símbolos.
2. Ajuste la opción "Solo lectura" de la forma conveniente para las variables seleccionadas.

Declaración de variables - Editar la lista de variables (ERPC)

Referencia

Parametrización de las variables configuradas en la CPU en el grupo de parámetros "Propiedades > General > Declaración de variables (ERPC)"

Se selecciona la variable para hacerla accesible para la aplicación ERPC.

Información mostrada

En la lista se muestran las variables ERPC guardadas en el proyecto STEP 7. Las variables ERPC que puedan haberse cargado anteriormente en la estación no se visualizan.

En la lista se muestra la siguiente información relativa a las variables configuradas para ERPC:

- Info
  Información adicional procedente de la comprobación de modificaciones
  Con la comprobación de modificaciones, a las variables ERPC modificadas en el proyecto STEP 7 después de haberlo guardado la última vez se les dota de la correspondiente información adicional en la columna "Acerca de" (ver abajo).
- Variable
  Nombre de la variable
- Escribible
  Los derechos configurados para la comunicación ERPC (leer de la CPU, escribir en la CPU); en la opción predeterminada solo se asignan derechos de lectura.
- Comentario
  Comentarios de las variables ERPC específicos para la comunicación ERPC

Debajo de la tabla:

- Preferencia
  Activando la opción "Solo lectura" se asigna a las variables ERPC seleccionadas el derecho "Solo lectura del área de datos de CPU".
- Número de variables:
  La cantidad total de las variables utilizadas para ERPC.

Comprobación de modificaciones

Al abrir la declaración de variables tiene lugar una comprobación de las modificaciones. Las variables seleccionadas para la aplicación ERPC se comparan con las de la tabla de variables de la CPU asignada en el proyecto STEP 7 (tabla de variables offline).

Nota

No tiene lugar una comparación con variables ERPC cargadas eventualmente en la estación desde el proyecto STEP 7 en un momento anterior.

En la columna "Acerca de..." se muestra la información relevante acerca de las diferencias entre las variables ERPC y la tabla de variables de la CPU.

Indicación en la columna "Info"	Estado / Información
	Variable no modificada
ACC	La variable no se puede editar actualmente. Motivo: El bloque de datos en el que se utiliza la variable está abierto en este momento.
DEL	Variable borrada de la tabla de variables de la CPU
ADR	Dirección modificada en la tabla de variables de la CPU
TYP	Tipo de datos modificado en la tabla de variables de la CPU Observación: Si para una variable de la tabla de variables se han modificado la dirección o el tipo de datos, la información "ADR" se sobrescribe con la información "TYP".
DEF	Tipo de variable no soportado
UPL	Variable cargada desde la estación. (tras ejecutar la función de STEP 7 "Cargar en PG")

Si existen entradas del tipo "DEL", "ADR" y "TYP", se tienen que efectuar las correcciones correspondientes antes de guardar el proyecto STEP 7.

Selección múltiple de variables

Se puede realizar una selección múltiple de variables con las teclas <Mayús> y <Ctrl>.

Configuración FTP para el servicio de servidor FTP

CP Ethernet como servidor FTP para los datos de la CPU

Referencia

Registro de datos para la creación automática de una tabla de asignación file en el grupo de parámetros "Propiedades > General > Configuración FTP".

Los datos recogidos en la tabla de asignación file permiten acceder a bloques de datos de una o varias CPUs (hasta 4) de una estación S7.

Opción "Activar servidor FTP" (solo en determinados tipos de CP)

Elija esta opción para permitir accesos FTP a la estación S7 a través del puerto 20/21 del CP.

La activación de esta opción es indispensable para activar el acceso FTP en File-DBs de la CPU con la opción descrita a continuación.

Por defecto, el acceso a través de un servidor FTP por el puerto 20/21 está activado.

Opción "Utilizar servidor FTP para datos de CPU S7"

Al seleccionar esta opción, la tabla de asignación File aquí configurada se crea al cargar los datos de configuración en el CP y se almacena en el directorio /config del sistema de archivos del CP.

Si existe ya un archivo file_db.txt, se sobrescribe.

Nota

Si la seguridad y la opción "Permitir acceso solo vía FTPS" están activadas, rige lo siguiente:

- Los archivos se transfieren codificados.
- El usuario debe tener activados los derechos "FTP: leer archivos (DBs) de la CPU S7" o "FTP: escribir archivos (DBs) de la CPU S7".
- Si el cortafuegos está activado, los protocolos FTP/FTPS deben estar autorizados.

CP Ethernet como servidor FTP para los datos de la CPU S7

Para transferir datos a través de FTP se crean bloques de datos en la CPU de la estación S7; debido a la particularidad de su estructura, dichos bloques se denominan aquí DBs file.

En calidad de servidor FTP, el CP Ethernet determina, a partir de una tabla de asignación file (archivo file_db.txt) con un comando FTP cómo deben reproducirse en archivos (files) los bloques de datos utilizados en la estación S7 para la transferencia de archivos (files).

La tabla de asignación file se crea y transfiere al CP del siguiente modo:

- Introduciéndola en el grupo de parámetros aquí descrito "Propiedades > Configuración FTP";
  La tabla de asignación file se genera a partir de los datos de la configuración FTP y se carga automáticamente en el CP junto con los datos de configuración.
- Creando directamente un archivo file_db.txt.
  La tabla de asignación file creada de este modo se carga en el CP con un comando FTP.

La tabla de asignación file file_db.txt se almacena en el directorio /config del sistema de archivos del CP Ethernet.

Consulte también

Configuración y estructura de la lista de configuración FTP (Página 2016)

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Server (Página 2017)

Configuración y estructura de la lista de configuración FTP

Configuración y estructura de la lista de configuración FTP para crear la tabla de asignación file

Los campos de la tabla de la configuración FTP tienen el significado y la sintaxis siguientes:

Asignación de títulos en la tabla de la configuración FTP	"CPU"	"DB"	Nombre de archivo	Comentario
Significado	Asignación de rack/slot de la CPU Seleccionable por lista desplegable; la entrada no puede introducirse.	N.º del bloque La entrada puede introducirse.	Nombre de archivo asignado al DB file Propuesta de nombre automática. La entrada puede introducirse/ modificarse.	Comentario informal
Ejemplo 1	cpu1 [PLC_1]	20	cpu1_db20.dat	Valores medidos en el área Instalación1
Ejemplo 2	cpu2 [PLC_2]	35	Cpu2_db35.dat	Valores medidos en el área Instalación2

Indicaciones relativas a la sintaxis

- Los separadores válidos para las entradas son "espacios".
- Para el nombre de archivo (File Name) de un DB file rige lo siguiente:
  - El nombre de archivo empieza por "cpuX" (siendo X=1,2,3 ó 4);
  - "cpux" debe estar definido previamente en la asignación de rack/slot de la CPU;
  - Longitud: máximo 64 caracteres ("cpuX" inclusive);
- Máximo 100 entradas;
- Caracteres admisibles: letras "A-Z,a-z"; cifras "0-9", "_", "."

Nota

Observe el formato (minúsculas para "cpu" y sin espacios al principio de la línea). De lo contrario los archivos no se reconocerán.

El tabulador **no** está permitido como separador.

Consulte también

CP Ethernet como servidor FTP para los datos de la CPU (Página 2014)

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Server

Funcionamiento

Para la transmisión de datos mediante FTP hay que crear bloques de datos (File-DBs) en la CPU de la estación S7. Dichos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como archivos transmisibles. Constan de los siguientes sectores

- Sector 1: File-DB-Header (tiene una longitud fija de 20 byte y una estructura fija)
- Sector 2: datos de usuario (longitud y estructura variables)Tipo de datos "FILE_DB_HEADER"

Tipo de datos "FILE_DB_HEADER"

En la función "Agregar nuevo bloque" se asigna el tipo "FILE_DB_HEADER" directamente a un bloque de datos.

File-DB-Header para modo de servidor FTP

Observación: el File-DB-Header aquí descrito es prácticamente idéntico al File-DB-Header descrito para el modo cliente. Las diferencias residen en los parámetros

- WRITE_ACCESS
- FTP_REPLY_CODE

Parámetro	Tipo	Valor / Significado	Alimentación
EXIST	BOOL	El bit EXIST indica si hay datos válidos en el área de datos de usuario. El comando FTP retrieve solo procesa la petición si EXIST=1. • 0: el File-DB no contiene datos de usuario válidos ("el archivo no existe"). • 1: el File-DB contiene datos de usuario válidos ("el archivo existe").	El comando FTP dele pone EXIST=0; El comando FTP store pone EXIST=1;
LOCKED	BOOL	El bit LOCKED sirve para la protección de acceso al File-DB. • 0: se puede acceder al File-DB. • 1: el File-DB está bloqueado.	Los comandos FTP store y retr ponen LOCKED=1 durante la edición. Para una operación de escritura desde el programa de usuario rige: El programa de usuario almacenado en la CPU S7 puede activar o desactivar LOCKED para asegurar la coherencia durante un acceso de escritura. Procedimiento recomendado para el programa de usuario: 1. Comprobar el bit LOCKED; si =0 2. Poner el bit WRITEACCESS=0 3. Comprobar el bit LOCKED; si =0 4. Poner el bit LOCKED=1 5. Escribir datos 6. Poner el bit LOCKED=0
NEW	BOOL	El bit NEW informa sobre si los datos se han modificado desde la última operación de lectura. • 0: el contenido del File-DB no ha cambiado desde la última operación de escritura. El programa de usuario de la CPU S7 ha registrado el último cambio. • 1: el programa de usuario de la CPU S7 no ha registrado la última operación de escritura.	El comando FTP store pone NEW=1 tras la edición El programa de usuario de la CPU S7 tiene que poner NEW=0 tras la lectura de los datos, para hacer posible un nuevo store o para poder borrar el archivo con el comando FTP dele.

Parámetro	Tipo	Valor / Significado	Alimentación
WRITE_ACCESS	BOOL	0: el cliente FTP de la PG/el PC no tiene derecho de escritura para los File-DBs de la CPU S7. 1: el cliente FTP de la PG/el PC tiene derecho de escritura para los File-DBs de la CPU S7.	El bit se pone a un valor de inicialización al configurar el DB. Recomendación: A ser posible, el bit debería dejarse inalterado. En caso especiales es posible una adaptación durante el funcionamiento.
ACT_LENGTH	DINT	Longitud actual del área de datos de usuario. El contenido de este campo solo es válido si EXIST = 1.	La longitud actual se actualiza después de una operación de escritura.
MAX_LENGTH	DINT	Longitud máxima del área de datos de usuario (longitud total del DB menos 20 bytes del encabezado).	La longitud máxima se debería definir al configurar el DB. El valor puede ser modificado por el programa de usuario también durante el funcionamiento.
FTP_REPLY_CODE	INT	Este parámetro no tiene relevancia en el modo de servidor FTP.	El servidor FTP lo pone a "0".
DATE_TIME	DATE_AND_TIME	Fecha y hora de la última modificación del archivo. El contenido de este campo solo es válido si EXIST = 1.	La fecha actual se actualiza después de una operación de escritura. Si se utiliza la función "Transmisión de la hora", la entrada se corresponde con la hora transmitida. Si no se utiliza la función "Transmisión de la hora", se introduce una hora relativa. La referencia es el momento de arranque del CP IT (el valor de inicialización es 1.1.1994 0.0 horas).

Consulte también

CP Ethernet como servidor FTP para los datos de la CPU (Página 2014)

Direcciones E/S

Referencia

Parametrización para las direcciones de entrada/salida en el grupo de parámetros "Propiedades > General > Direcciones E/S"

Significado

- Entradas, salidas
  Asigne una dirección inicial al módulo. (Salidas solo si está seleccionada la opción "Ajuste de dirección para LOCK/UNLOCK con FETCH/WRITE")
- Opción "Ajuste de dirección para LOCK/UNLOCK con FETCH/WRITE"
  Seleccione esta opción si desea utilizar la función de coordinación de acceso con los FCs LOCK/UNLOCK en modo FETCH/WRITE.
  Esta función utiliza la salida de proceso a través del bus de fondo/P. Por ello, en cuanto se selecciona esta opción también pueden ajustarse las direcciones de salida.

Web

Servidor web en el CP

Referencia

Parametrización para los accesos al servidor web del CP en el grupo de parámetros "Propiedades > General > Servidor web"

Opción "Activar servidor web"

El CP le ofrece la función de un servidor web para acceder a través de un navegador web. Para ello, se almacenan determinadas páginas HTML con información del CP y funciones de diagnóstico en un área de memoria del CP.

Active esta opción para obtener acceso a dichas páginas HTML. Con ello se habilitará el puerto 80 del CP.

Por defecto, el acceso a través de un servidor web está activado.

Opciones del diagnóstico web (disponibilidad en función del tipo de CP)

- Opción "Visor de topología"
  Esta opción permite visualizar en el servidor web información de topología para las interfaces PROFINET IO del CP conectadas en red.
  Para determinar y almacenar la información de topología se ocupa espacio adicional en el CP. Por consiguiente, en caso de configuraciones de proyectos complejas con elevadas exigencias para los recursos de memoria es aconsejable activar esta opción solo provisionalmente para fines de servicio.
- Opción "Descarga de firmware vía web"
  Al activar esta opción se habilita la función para cargar el firmware del CP a través del centro de descargas del servidor web.
  Si la Security está desactivada, esta opción no está sometida a la administración de usuarios. Por ello, se recomienda habilitar la opción solo en caso necesario.
- Opción "Carga a posteriori de los archivos de idiomas para visores de diagnóstico a través de web"
  Los visores de diagnóstico de los CPs se visualizan con texto claro en el búfer de diagnóstico web. La visualización es específica del idioma.
  Al activar esta opción se habilita la función para recargar los archivos de idiomas que faltan a través del centro de descargas del servidor web.

Actualización automática / "Intervalo de actualización"

Si se elige la opción "Activar", el CP actualiza las páginas web visualizadas en el intervalo ajustado.

Rango de valores para el intervalo de actualización 1..999 s

Diagnóstico web en el CP

Referencia

Diagnóstico del navegador web en el grupo de parámetros "Propiedades > General > Diagnóstico web"

Requisitos:

Establezca un enlace físico entre la PG y la estación SIMATIC S7 y parametrice la interfaz PG/PC de manera que el CP esté accesible. Encontrará más ayuda en la función "Ajustar interfaz PG/PC...".

"Inicio del diagnóstico web"

Al hacer clic en el botón "Diagnóstico web" se muestra el resultado del diagnóstico del módulo en el navegador web. El servidor HTTP integrado del CP suministra los contenidos.

- Opción "Acceso vía"
  En la lista "Acceso vía" seleccione la interfaz desde la que se activará el CP.
  - En la lista desplegable pueden seleccionarse las interfaces disponibles.
  - Existe la posibilidad de introducir directamente una dirección IP.

Indicación si Security está activada

Si la opción "Permitir acceso solo vía HTTP" está activada, tenga en cuenta lo siguiente:

- Los datos se transfieren codificados.
- El usuario debe tener activado el derecho "Acceder a diagnóstico web".
- Si el firewall está activado, los protocolos HTTP/HTTPS deben estar autorizados.

SNMP

Referencia

SNMP se activa y configura en el grupo de parámetros "Propiedades > General > SNMP"

Opción "Activar SNMP"

- Si esta opción no está activada, los servicios SNMP no pueden responder (ni v1 ni v3).
- Si la opción está activada, el módulo actúa de forma estándar como agente SNMPv1. El módulo solo actúa como agente SNMPv3 si la seguridad está activada y se ha seleccionado SNMPv3 para la configuración SNMP.

CP/CM Industrial Ethernet

Configuración FTP

CP como servidor FTP para los datos de la CPU S7

CP Ethernet como servidor FTP para los datos de la CPU

Referencia

Parametrización del modo de operación Servidor FTP en el grupo de parámetros del CP Ethernet

- con la seguridad desactivada, en "Propiedades > General > Configuración FTP".
- con la seguridad activada, en "Propiedades > Seguridad > General > Configuración FTP".

CP S7-1500 como servidor FTP

La función aquí descrita permite transferir datos en forma de archivos (files) a bloques de datos o desde bloques de datos de una estación S7 utilizando los comandos FTP. Es posible utilizar los comandos FTP usuales para leer, escribir y administrar archivos.

Para la transmisión de datos mediante FTP hay que crear bloques de datos en la CPU de la estación S7. Debido a su estructura especial, se denominan File-DBs.

Con un comando FTP, el CP 1543-1 en calidad de servidor FTP determina a partir de una tabla de asignación File cómo deben reproducirse en archivos (Files) los bloques de datos utilizados para la transferencia de archivos en la estación S7.

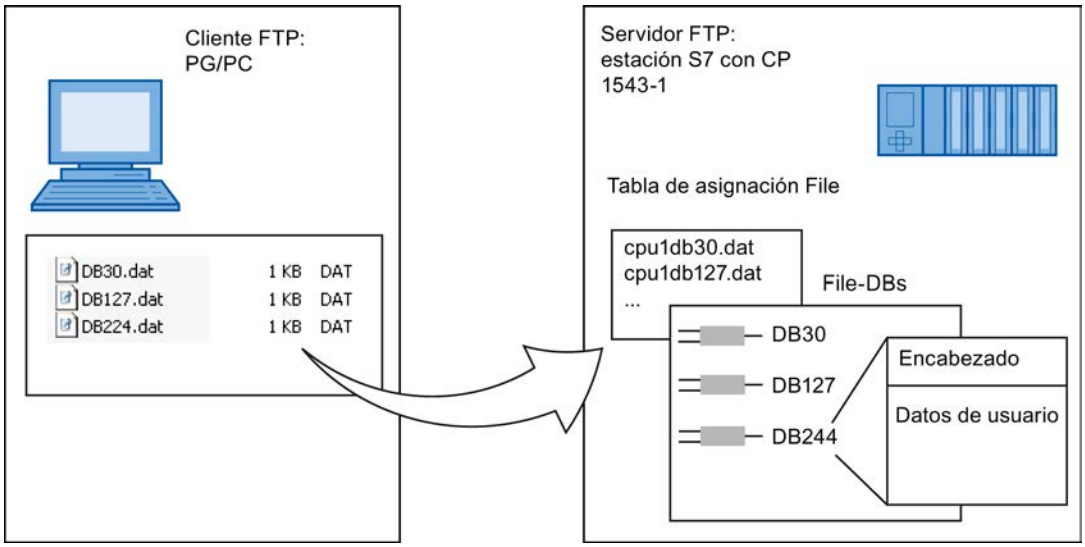


Figura 8-55 CPU S7 con CP 1543-1 como servidor FTP para los datos de la CPU S7

Tabla de asignación File

El CP 1543-1 utilizado como servidor FTP necesita información sobre cómo se deben reproducir en archivos los bloques de datos usados para la transferencia de archivos en la estación S7.

La tabla de asignación File contiene las entradas siguientes:

- Asignación de la CPU
- asignación de DB
- nombre de archivo

Requisitos en la configuración de la CPU

Utilice los ajustes siguientes para permitir el acceso FTP:

- En los datos de configuración de la CPU, en "Propiedades > General > Protección": desactive la opción "Desactivar comunicación PUT/GET".
- En todos los bloques de datos utilizados como File-DB desactive el atributo "Acceso optimizado al bloque"

Consulte también

Configuración y estructura de la lista de configuración FTP (Página 2024)

Comandos FTP en el cliente FTP (Página 2025)

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Server (Página 2027)

Configuración y estructura de la lista de configuración FTP

Configuración y estructura de la lista de configuración FTP para crear la tabla de asignación File

Los campos de la tabla de la configuración FTP tienen el significado y la sintaxis siguientes:

Asignación de títulos en la tabla de la configuración FTP	"CPU"	"DB"	Nombre de archivo	Comentario
Significado	Asignación de la CPU Seleccionable por lista desplegable; la entrada no puede introducirse.	N.º del bloque La entrada puede introducirse.	Nombre de archivo asignado al File-DB Propuesta de nombre automática. La entrada puede introducirse/ modificarse.	Comentario informal
Ejemplo	cpu1 [PLC_1]	20	cpu1_db20.dat	Valores medidos en el área Instalación1

Indicaciones relativas a la sintaxis

- Para el nombre de archivo (File Name) de un File-DB rige lo siguiente:
  - El nombre de archivo empieza por "cpuX" (siendo X=1 en S7-1500);
  - Longitud: máximo 64 caracteres ("cpuX" inclusive);

Nota

Observe el formato (minúsculas para "cpu" y sin espacios al principio de la línea). De lo contrario los archivos no se reconocerán.

Consulte también

CP Ethernet como servidor FTP para los datos de la CPU (Página 2023)

Comandos FTP en el cliente FTP

Un ejemplo de acceso

En la siguiente representación se ve a modo de ejemplo una secuencia de acceso típica en la ventana MS-DOS:

```
MS-Dos - ftp 141.73.10.33

c:\>ftp 141.73.10.33
Verbunden zu 141.73.10.33.
220 CP 443-1 IT FTP-Server V1.06 ready for new user
Benutzer (141.73.10.33:(none)): ftpadmin
331 User name okay, need password.
Kennwort:
230 User logged in, proceed.
Ftp> dir /cpu1
200 Command okay.
150 File status okay; about to open data connection.
-r--r--r--   1 root root          740 Sep 13 14:14 cpu1db20
-rw-rw-rw-   1 root root          987 Aug 28 14:16 cpu1db30
--w--w--w-   1 root root        64000 Mar 18  2001 cpu1db40
226 Transfer ok. Closing data connection.
183 Bytes empfangen in 0,04 Sekunden (4,58 KB/s)
Ftp> get cpu1db30
200 Command okay.
150 File status okay; about to open data connection.
226 Transfer ok. Closing data connection.
987 Bytes empfangen in 0,15 Sekunden (6,58 KB/s)
Ftp>
```

Cómo utilizan los comandos FTP típicos los comandos FTP admisibles

Consulte en la tabla siguiente qué comandos FTP se pueden ejecutar para el acceso a los File-DBs de la CPU. La tabla muestra también qué comandos FTP se emplean en las consolas de introducción típicas como, por ejemplo, el prompt de MS-DOS, para estos comandos FTP.

Tabla 8-96 Comandos FTP

Comandos FTP típicos						Comando FTP	Significado
open	dir	put	get	close	del		
x						User	Iniciar sesión
x						Pass	Autorización por contraseña
	x	x	x			Port	Indicación del puerto a través del que un cliente desea transmitir datos.
	x					List	Lista de los File-DBs en la CPU aludida.
					x	Dele	Borra un File-DB, poniendo a "0" el bit EXIST en el encabezado del File-DB.
			x			Retr	Lee los datos de usuario que están en el File-DB indicado en el archivo especificado del cliente FTP.
		x				Stor	Trasmite el archivo indicado desde el cliente FTP al área de datos de usuario del File-DB especificado.
				x		quit	Finaliza la conexión FTP actual.

Nota

Los comandos FTP "rename", "append", "rnfr" y "rnto" no se pueden utilizar para File-DBs.

Cómo se procesan comandos FTP en el CP

Para una mejor comprensión de la interfaz FTP destinada a los File-DBs existentes en la CPU, se explica aquí la secuencia de procesamiento con el ejemplo del comando stor.

El servidor FTP del CP procede del siguiente modo:

1. Identificación del File-DB aludido sobre la base de la entrada en la tabla de asignación File.
2. Comprobación de los bits en el encabezado del File-DB.
   El proceso de escritura únicamente se ejecuta cuando se comprueba que:
   el bit LOCKED = 0
   el bit NEW = 0
   el bit WRITEACCESS = 1
3. Escritura del contenido del archivo en el área de datos de usuario del File-DB en la CPU. Al comienzo del proceso de escritura se activa el bit LOCKED, que se desactiva tras concluir el proceso de escritura.
4. Una vez concluido el proceso de escritura se activa adicionalmente el bit NEW en el encabezado del File-DB y se introduce la fecha actual en el campo DATE_TIME.
5. El servidor FTP envía al cliente FTP un aviso relativo al resultado de la transferencia de archivos.

Modo de transmisión para la transferencia de archivos

La transferencia de archivos tiene lugar exclusivamente en el modo binario. Para ello, tras iniciar la sesión hay que introducir en la consola de introducción el comando "binary".

Consulte también

CP Ethernet como servidor FTP para los datos de la CPU (Página 2023)

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Server

Funcionamiento

Para la transmisión de datos mediante FTP hay que crear bloques de datos (File-DBs) en la CPU de la estación S7. Dichos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como archivos transmisibles. Constan de los siguientes sectores

- Sector 1: File-DB-Header (tiene una longitud fija de 20 byte y una estructura fija)
- Sector 2: datos de usuario (longitud y estructura variables)

Tipo de datos "FILE_DB_HEADER"

En la función "Agregar nuevo bloque" se asigna el tipo "FILE_DB_HEADER" directamente a un bloque de datos.

Requisitos en la configuración de la CPU

Utilice los ajustes siguientes para permitir el acceso FTP:

- En todos los bloques de datos utilizados como File-DB desactive el atributo "Acceso optimizado al bloque"

File-DB-Header para modo de servidor FTP

Observación: el File-DB-Header aquí descrito es prácticamente idéntico al File-DB-Header descrito para el modo cliente. Las diferencias residen en los parámetros

- WRITE_ACCESS
- FTP_REPLY_CODE

Parámetro	Tipo	Valor / Significado	Alimentación
EXIST	BOOL	El bit EXIST indica si hay datos válidos en el área de datos de usuario. El comando FTP retrieve solo procesa la petición si EXIST=1. • 0: el File-DB no contiene datos de usuario válidos ("el archivo no existe"). • 1: el File-DB contiene datos de usuario válidos ("el archivo existe").	El comando FTP dele pone EXIST=0; El comando FTP store pone EXIST=1;
LOCKED	BOOL	El bit LOCKED sirve para la protección de acceso al File-DB. • 0: se puede acceder al File-DB. • 1: el File-DB está bloqueado.	Los comandos FTP store y retr ponen LOCKED=1 durante la edición. Para una operación de escritura desde el programa de usuario rige: El programa de usuario almacenado en la CPU S7 puede activar o desactivar LOCKED para asegurar la coherencia durante un acceso de escritura. Procedimiento recomendado para el programa de usuario: 1. Comprobar el bit LOCKED; si =0 2. Poner el bit WRITEACCESS=0 3. Comprobar el bit LOCKED; si =0 4. Poner el bit LOCKED=1 5. Escribir datos 6. Poner el bit LOCKED=0

Parámetro	Tipo	Valor / Significado	Alimentación
NEW	BOOL	El bit NEW informa sobre si los datos se han modificado desde la última operación de lectura. • 0: el contenido del File-DB no ha cambiado desde la última operación de escritura. El programa de usuario de la CPU S7 ha registrado el último cambio. • 1: el programa de usuario de la CPU S7 no ha registrado la última operación de escritura.	El comando FTP store pone NEW=1 tras la edición El programa de usuario de la CPU S7 tiene que poner NEW=0 tras la lectura de los datos, para hacer posible un nuevo store o para poder borrar el archivo con el comando FTP dele.
WRITE_ACCESS	BOOL	0: el cliente FTP de la PG/el PC no tiene derecho de escritura para los File-DBs de la CPU S7. 1: el cliente FTP de la PG/el PC tiene derecho de escritura para los File-DBs de la CPU S7.	El bit se pone a un valor de inicialización al configurar el DB. Recomendación: A ser posible, el bit debería dejarse inalterado. En caso especiales es posible una adaptación durante el funcionamiento.
ACT_LENGTH	DINT	Longitud actual del área de datos de usuario. El contenido de este campo solo es válido si EXIST = 1.	La longitud actual se actualiza después de una operación de escritura.
MAX_LENGTH	DINT	Longitud máxima del área de datos de usuario (longitud total del DB menos 20 bytes del encabezado).	La longitud máxima se debería definir al configurar el DB. El valor puede ser modificado por el programa de usuario también durante el funcionamiento.
FTP_REPLY_CODE	INT	Este parámetro no tiene relevancia en el modo de servidor FTP.	El servidor FTP lo pone a "0".
DATE_TIME	DATE_AND_TIME	Fecha y hora de la última modificación del archivo. El contenido de este campo solo es válido si EXIST = 1.	La fecha actual se actualiza después de una operación de escritura. Si se utiliza la función "Transmisión de la hora", la entrada se corresponde con la hora transmitida. Si no se utiliza la función "Transmisión de la hora", se introduce una hora relativa. La referencia es el momento de arranque del CP IT (el valor de inicialización es 1.1.1994 0.0 horas).

Consulte también

CP Ethernet como servidor FTP para los datos de la CPU (Página 2023)

CP como cliente FTP para los datos de la CPU S7

Funcionamiento de un cliente FTP

Principio de un cliente FTP

Para la transmisión de datos mediante FTP hay que crear bloques de datos (File-DBs) en la CPU de la estación S7.

El programa de usuario envía peticiones de FTP que son ejecutadas por el CP 1543-1 como cliente FTP. En el programa de usuario se utiliza la instrucción FTP_CMS para establecer la conexión y transferir datos.

En la petición debe indicarse, entre otros:

- dirección IP del servidor FTP
- ubicación del archivo en el servidor FTP
- nombres de archivo
- información de acceso

Las ilustraciones siguientes muestran el funcionamiento si se utiliza la instrucción FTP_CMD.

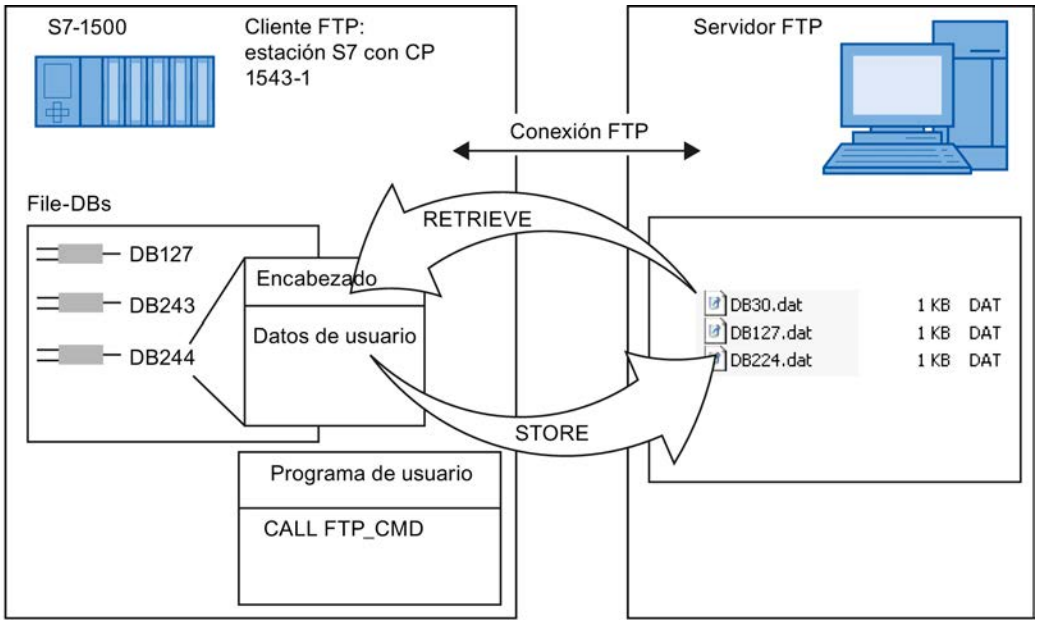


Figura 8-56 CPU con CP 1543-1 como cliente FTP para los datos de la CPU S7

Requisitos en la configuración de la CPU

Utilice los ajustes siguientes para permitir el acceso FTP:

- En los datos de configuración de la CPU, en "Propiedades > General > Protección": desactive la opción "Desactivar comunicación PUT/GET".
- En todos los bloques de datos utilizados como File-DB desactive el atributo "Acceso optimizado al bloque"

Informaciones adicionales

Consulte también el capítulo Auto-Hotspot.

Modo de operación FETCH/WRITE

Aplicación de FETCH/WRITE

FETCH/WRITE

Los servicios FETCH/WRITE permiten acceder a áreas de memoria del sistema de la CPU S7.

- FETCH: Leer datos directamente
- WRITE: Escribir datos directamente

El acceso es posible desde los siguientes dispositivos:

- SIMATIC S5
- Estaciones PC SIMATIC
- Dispositivos ajenos

Memoria de sistema

Mediante FETCH o WRITE es posible acceder a las siguientes áreas de operandos en la memoria de sistema de SIMATIC S7:

- Bloques de datos (DB)
  (tenga en cuenta la siguiente limitación para el acceso a DB: el número máximo de DB es 255)
- Marcas (M)
- Memoria imagen de proceso de las entradas (PAE)
- Memoria imagen de proceso de las salidas (PAA)
- Área de periferia de las entradas (PEW, PED, PEB)
- Área de periferia de las salidas (PAW, PAD, PAB)
- Contadores (Z)
- Temporizadores (T)

Requisitos en la configuración de la CPU

Utilice los ajustes siguientes para permitir el acceso FETCH/WRITE:

- En los datos de configuración de la CPU, en "Propiedades > General > Protección": desactive la opción "Desactivar comunicación PUT/GET".

Acoplamiento con sistemas ajenos

El modo de operación para FETCH y WRITE soportado en las conexiones de transporte ISO y en las conexiones ISO on TCP o TCP puede ser utilizado en principio desde cualquier dispositivo ajeno para el acceso a áreas de memoria del sistema de S7.

Para poder implementar también este acceso, por ejemplo, para aplicaciones de PC, se tiene que conocer la estructura de PDU para las peticiones. Consulte al respecto los capítulos siguientes.

Consulte también

Configurar el acceso FETCH/WRITE (Página 2032)

Acoplamiento con sistemas de terceros con FETCH/WRITE (Página 2033)

Configurar el acceso FETCH/WRITE

Referencia

Parametrización del modo de operación FETCH/WRITE en el grupo de parámetros "Propiedades > General > Opciones"

Configuración de FETCH/WRITE

Para habilitar el acceso FETCH/WRITE, primero hay que activar la opción.

Seguidamente, deben introducirse las identificaciones de acceso en las listas que aparecen. La identificación de acceso se consigue indicando lo siguiente:

- TSAP (Transportation Service Access Point)
  Para el acceso a través de conexiones ISO-Transport o ISO-on-TCP deben indicarse TSAPs. Se autorizarán los accesos FETCH/WRITE cuyos datos TSAP coincidan con una entrada de la lista.
- Puerto TCP
  Para el acceso a través de conexiones TCP deben indicarse puertos TCP. Se autorizarán los accesos FETCH/WRITE cuyos datos de puerto TCP coincidan con una entrada de la lista.

Opción "Modo de direccionamiento S7"

En la configuración para el modo de operación FETCH/WRITE es posible seleccionar el modo de direccionamiento. De esta manera se determina la forma en la que debe interpretarse la

indicación de la dirección en la llamada FETCH/WRITE en la estación SIMATIC S7 para el acceso a DBs:

- Modo de direccionamiento S7: Dirección de byte
- Modo de direccionamiento S5: Dirección de palabra

Esto permite que las aplicaciones accedan a las estaciones S5 o S7 sin adaptar las direcciones. Esto es útil, por ejemplo, para aplicaciones S5 existentes que deben utilizarse tal como están para acceder a estaciones S7.

El modo de direccionamiento para el acceso a SIMATIC S7 (opción seleccionada) está seleccionado por defecto.

Consulte también

Aplicación de FETCH/WRITE (Página 2031)

Configuración de puerto (Página 2033)

Configuración de puerto

Configuración de puerto

La tabla siguiente aclara el rango de valores y uso de las direcciones de puerto:

Direcciones de puerto		Aplicación/nota
0		Asignado de forma fija; ¡no puede utilizarse!
1..1023		Asignado de forma estándar; no se debería utilizar (well known ports)
1024..49151	Puertos para protocolos específicos de aplicación	
	a partir de 2000...5000	Rango en el que la herramienta de configuración busca o asigna una dirección de puerto libre en cada caso. Es posible ajustar la dirección de puerto en este rango de forma individual.
	5001..49151	¡El sistema utiliza las direcciones de puerto a partir de 5000! Observación: Si desea utilizar estas direcciones de puerto, contacte en caso dado a su administrador del sistema para que compruebe su disponibilidad.
49152..65535		Puertos asignados de forma dinámica Se recomienda no utilizar estos puertos.

Acoplamiento con sistemas de terceros con FETCH/WRITE

El modo de operación para FETCH y WRITE puede ser utilizado en principio por cualquier dispositivo ajeno para el acceso a áreas de memoria del sistema S7.

Para poder implementar también este acceso, por ejemplo, para aplicaciones de PC, se tiene que conocer la estructura de PDU para las peticiones. Los encabezados específicos de S7 o S5 necesarios para los telegramas de petición y acuse tienen una longitud predeterminada de 16 bytes; su estructura se expone a continuación:

a) Estructura para telegrama WRITE

El significado y los valores de los parámetros no ocupados con valores fijos en la tabla siguiente pueden consultarse en el apartado siguiente "Abastecimiento de parámetros".

Telegrama de petición WRITE			Telegrama de acuse WRITE		
0	Identificación del sistema	="S"	0	Identificación del sistema	="S"
1		="5"	1		="5"
2	Longitud de bytes del encabezado	=16d.	2	Longitud.Encabezado	=16d.
3	Identificación del código OP	=01	3	Identificación del código OP	=01
4	Longitud del código OP	=03	4	Longitud del código OP	=03
5	Código OP	=03	5	Código OP	=04
6	Bloque ORG	=03	6	Bloque de acuse	=0Fh
7	Longitud del bloque ORG	=08	7	Longitud del bloque de acuse	=03
8	Identificador ORG		8	Bloque de errores	=Nr
9	DBNR		9	Bloque vacío	=FFh
A	Dirección inicial	Byte High	A	Longitud de bloque vacío	=07
B		Byte Low	B		
C	Longitud	Byte High	C		
D		Byte Low	D	libre	
E	Bloque vacío	=FFh.	E		
F	Longitud de bloque vacío	=02	F		
	Conjunto de datos con hasta 64 K				

b) Estructura para telegrama FETCH

El significado y los valores de los parámetros no ocupados con valores fijos en la tabla siguiente pueden consultarse en el apartado siguiente "Abastecimiento de parámetros".

Telegrama de petición FETCH			Telegrama de acuse FETCH		
0	Identificación del sistema	="S"	0	Identificación del sistema	="S"
1		="5"	1		="5"
2	Longitud.Encabezado	=0x10	2	Longitud.Encabezado	=0x10
3	Id. código OP	=0x01	3	Id. código OP	=0x01
4	Longitud del código OP	=0x03	4	Longitud del código OP	=0x03
5	Código OP	=0x05	5	Código OP	=0x06
6	Bloque ORG	=0x03	6	Bloque de acuse	=0x0F
7	Longitud del bloque ORG	=0x08	7	Longitud del bloque de acuse	=0x03
8	Identificador ORG		8	Bloque de errores	=Nr
9	DBNR		9	Bloque vacío	=0xFF
A	Dirección inicial	Byte High	A	Longitud de bloque vacío	=0x07
B		Byte Low	B	libre	
C	Longitud	Byte High	C		

D		Byte Low	D		
E	Bloque vacío	=0xFF	E		
F	Longitud de bloque vacío	=0x02	F		
				Datos hasta 64 K, pero solo si n.º error = 0	

Abastecimiento de parámetros

Área de operandos S7	DB	M	E	A
Identificador ORG	01_H	02_H	03_H	04_H
	Datos de origen/ destino de/en bloque de datos en memoria principal	Datos de origen/ destino de/en área de marcas	Datos de origen/ destino de/en memoria imagen de proceso de las entradas (IPE)	Datos de origen/ destino de/en memoria imagen de proceso de las salidas (IPS)
DBNR	DB del que se toman los datos de origen o al que se transfieren los de destino	irrelevante	irrelevante	irrelevante
Rango permitido	1...255			
Dirección inicial	Número DW a partir del que se toman o escriben datos	N.º de byte de marcas a partir del que se toman o escriben datos	N.º de byte de entrada a partir del que se toman o escriben datos	N.º de byte de salida a partir del que se toman o escriben datos
Rango permitido	0...2047	Todos los bytes de marcas que proporciona una CPU.	0...127	0...127
Longitud	Longitud del conjunto de datos de origen/ destino en palabras	Longitud del conjunto de datos de origen/ destino en bytes	Longitud del conjunto de datos de origen/ destino en bytes	Longitud del conjunto de datos de origen/ destino en bytes
Rango permitido	Hasta 8192 bytes	Hasta 8192 bytes	1...128	1...128

Área de operandos S7	PI/PQ	C	T
Identificador ORG	05_H	06_H	07_H

	Datos de origen/ destino de/en módulos de periferia. Para datos de origen, módulos de entrada; para datos de destino, módulos de salida	Datos de origen/ destino de/en células de contador	Datos de origen/ destino de/en células de temporizador
DBNR	irrelevante	irrelevante	irrelevante
Dirección inicial	N.º de byte de periferia a partir del que se toman o escriben datos	Número de célula de contador a partir del que se toman o escriben datos	Número de célula de temporizador a partir del que se toman o escriben datos
Rango permitido	0...127 Periferia digit. 128...255 Periferia anal.	0...255	0...255
Número	Longitud del conjunto de datos de origen/ destino en bytes	Longitud del conjunto de datos de origen/ destino en palabras (célula contador = 1 palabra)	Longitud del conjunto de datos de origen/ destino en palabras (célula contador = 1 palabra)
Rango permitido	1...256	1	1

Consulte también

Aplicación de FETCH/WRITE (Página 2031)

CP PROFIBUS

Sincronización horaria

Referencia

Parametrización de las propiedades para CPs PROFIBUS en el grupo de parámetros "Propiedades > Ajustes > Sincronización horaria".

Sincronización horaria

Especifique si el CP debe transferir telegramas horarios o no.

Según donde se encuentre el reloj maestro, hay que distinguir entre los dos casos siguientes:

- Los telegramas horarios proceden de la subred y se transfieren a la estación.
- Los telegramas horarios proceden de la estación y se transfieren a la subred.

En algunos módulos nuevos se especifica de forma selectiva el sentido en el que deben transferirse los telegramas horarios. En módulos en los que la transferencia solo puede

activarse o desactivarse globalmente, parta del principio de que el CP puede transferir un telegrama horario en cualquier sentido.

Si utiliza varios CPs en la estación, tenga en cuenta que el flujo de telegramas horarios depende del reloj maestro. Dado que con la función aquí descrita se pueden acoplar telegramas horarios de una red a otra, pueden producirse conflictos.

Si en una estación hay varios CPs conectados a la misma red, solo uno de ellos puede transferir los mensajes de sincronización horaria.

Parametrización de aparatos de campo (routing de registros)

Referencia

Parametrización de las propiedades para CPs PROFIBUS en el grupo de parámetros "Propiedades > Ajustes > Parametrización de aparatos de campo".

Parametrización de aparatos de campo (routing de registros)

Seleccionando esta opción se puede utilizar el CP como router para registros destinados a aparatos de campo (esclavos DP). En ese caso, el CP desvía a los aparatos de campo los registros transferidos por dispositivos que no están conectados directamente al PROFIBUS, y que, por lo tanto, no tienen acceso directo a los aparatos de campo (esclavos DP).

Una herramienta que genera este tipo de registros para la parametrización de aparatos de campo es SIMATIC PDM (Process Device Manager).

La función está activada de forma estándar. Dado que la función requiere recursos de memoria adicionales, puede desconectarse si en general los recursos de memoria del CP se someten a una carga elevada (conexiones, etc.) y no se requiere la función "Routing de registros".

CPs con funcionalidad DP - Seleccionar el modo de operación

Referencia

Parametrización del modo de operación del CP para PROFIBUS DP en el grupo de parámetros "Propiedades > General > Modo de operación"

Relación entre el modo de operación CP y la función CP

Encontrará los modos de operación de CP configurables y, con ello, las funciones de CP posibles en la tabla siguiente:

Modo de operación CP	Funciones CP posibles				
	PG/BuB vía PROFIBUS	FDL vía PROFIBUS	Maestro DP	Esclavo DP	Conexiones S7 configuradas
Sin modo DP	X	X	-	-	X

Modo de operación CP	Funciones CP posibles				
Modo maestro DP	X	X	X	-	X
Esclavo DP activo (v. abajo)	X	X	-	X	-
Esclavo DP pasivo (v. abajo)	-	-	-	X	-

Leyenda: "X" = la función se soporta; "-" = la función no se soporta

Modo maestro DP / parámetro "Tiempo de retardo DP"

Aquí se puede ajustar un retardo adicional para la actualización de los datos E/S DP. Este tiempo de retardo es utilizado por el CP PROFIBUS para editar otros protocolos, como las conexiones FDL.

Los tiempos de reacción DP visualizados no se actualizarán hasta que se ejecute una de las acciones siguientes:

- Pulsar el botón "Recalcular..." o
- Seleccionar otro grupo de parámetros

Si se han configurado muchos esclavos DP, el recálculo puede durar varios segundos.

Maestro DP clase 2 (funciones de diagnóstico - sólo seleccionable con determinados tipos de CP)

En este modo de operación, el dispositivo puede leer los datos de entrada y salida de cualquier esclavo DP que no tenga asignado a través del CP PROFIBUS.

De este modo, por ejemplo, se puede registrar una señal de proceso de varios maestros DP, con lo que se pueden ahorrar sensores en el campo (shared input / shared output). Típicamente, los dispositivos de programación, diagnóstico o administración trabajan en calidad de maestro DP clase 2.

Modo esclavo DP / opción "Test, puesta en servicio y routing" (la función varía según el tipo de CP)

- Opción activada: esclavo DP activo
  Active la casilla de verificación si desea realizar con frecuencia funciones de programadora necesarias para la puesta en marcha y el test, por ejemplo, utilizando esta interfaz. Con ello se elige el modo de operación CP "Esclavo DP activo" (es decir, la interfaz participa en la rotación del token de PROFIBUS).
  En este caso son posibles las funciones siguientes (la función varía según el tipo de CP):
  - Programar (p. ej. cargar)
  - Probar (estado/forzar)
  - Routing S7 (esclavo I como transición de red)
  - Comunicación compatible con S5 (interfaz SEND/RECEIVE)
  - Comunicación S7 (sólo en modo servidor en conexiones configuradas unilateralmente sin bloques de comunicación; configuración en el interlocutor)

  Nota

  Repercusiones especiales del modo de operación "Esclavo DP activo"
  - En este modo de operación no se pueden configurar conexiones S7 para el intercambio de datos a través de bloques de comunicación.
  - El tiempo de ciclo puede prolongarse. En aplicaciones críticas y si no se necesitan el routing S7 ni la funcionalidad de cliente para la comunicación, no debería activarse esta opción.

- Opción desactivada: esclavo DP pasivo
  Si la casilla de verificación "Test, puesta en servicio y routing" no está activada, se selecciona el modo de operación CP "Esclavo DP pasivo", con lo que la interfaz sólo trabaja como servidor de servicios de comunicación y no participa en la rotación del token de PROFIBUS.
  La funcionalidad de servidor significa que sólo el interlocutor (p. ej. PG, OP o sistema de automatización) puede iniciar la comunicación. Las particularidades siguientes deben tenerse en cuenta si la opción no está activada:
  - No es posible el routing S7.
  - La programación y el test son posibles pero lentamente (sin prolongación del tiempo de ciclo).

El CP soporta varios tipos de esclavo DP

Referencia

Parametrización del tipo de esclavo DP para PROFIBUS DP en el grupo de parámetros "Propiedades > General > Modo de operación"

Modo DP

En los CPs que soportan distintos tipos de esclavos DP se puede elegir entre los siguientes modos DP, dependiendo del tipo de CP:

- Funcionalidad DPV1 (modo de operación predeterminado)
  Aquí se incluyen esclavos DP conformes a la norma PROFIBUS DPV0 y DPV1 así como esclavos DP Siemens.
  Este modo de operación presupone determinados tipos de CPU. Para el CP PROFIBUS tenga en cuenta las indicaciones del manual de producto de CPs S7.
- Compatible con DPV0
  Aquí se incluyen esclavos DP conformes a la norma PROFIBUS DPV0; los esclavos DP conformes a la norma DPV1 sólo pueden utilizarse con restricciones.
- Compatible con S7
  Aquí se incluyen esclavos DP conformes a la norma PROFIBUS DPV0 así como esclavos DP Siemens; los esclavos DP conformes a la norma DPV1 sólo pueden utilizarse con restricciones.

CP/CM PROFIBUS

Parametrizar el reloj / Sincronización horaria

Sincronización horaria a través del CM

Mediante los parámetros de la sincronización horaria se especifica el rol que desempeña el CM para transmitir telegramas horarios.

En función del modo de operación del dispositivo en PROFIBUS pueden ajustarse los siguientes tipos de sincronización:

- El CM es un dispositivo pasivo en PROFIBUS (p. ej. esclavo DP): esclavo, ninguna
- El CM es un dispositivo activo en PROFIBUS: maestro, esclavo, ninguna

Significado del tipo de sincronización horaria

- Esclavo:
  el reloj del CM es sincronizado por otro reloj. Los telegramas de sincronización provienen de PROFIBUS. Con este tipo de sincronización, el CM reenvía los telegramas horarios a la estación en un intervalo de tiempo especificado.
  Dentro de la estación S7 solo debe haber un reloj esclavo.
- Maestro:
  el reloj del CM envía telegramas de sincronización a PROFIBUS.
  Si está configurado el tipo de sincronización "Maestro", se puede elegir el intervalo de tiempo para la emisión de los telegramas de sincronización a PROFIBUS.
- Ninguna:
  no hay sincronización.

Si en una red hay más de un módulo con reloj, hay que determinar qué módulo actuará como maestro y cuál como esclavo para la sincronización horaria. Solo puede haber un reloj maestro.

- Intervalo de actualización
  El intervalo de actualización define, con el tipo de sincronización "Maestro", el intervalo en el que se emiten los telegramas horarios.

Ajustes de CP independientes de protocolo o de red

Cambio de módulo sin programadora

Referencia

Parametrización de ajustes especiales en el grupo de parámetros "Propiedades > General > Ajustes"

Dependiendo del tipo de CP, encontrará la posibilidad de ajuste "Cambio de módulo sin programadora"

Repercusión - almacenamiento seguro a largo plazo en la CPU

Si se ha seleccionado esta opción, el almacenamiento seguro a largo plazo se realiza en la CPU en lugar del EEPROM del CP. Sin embargo, tenga en cuenta que, también en la CPU, el almacenamiento seguro a largo plazo sólo es posible si dicha CPU está asegurada contra fallos de tensión mediante un respaldo por pila o una Memory Card S7.

Demanda de recursos en la CPU

Si se selecciona esta opción se ocuparán más recursos en la CPU. Al cargar los programas de usuario y los datos de configuración, se notificará si no hay memoria suficiente. La falta de recursos puede solucionarse utilizando una Memory Card S7.

Nota

Si hay que suponer que faltarán recursos y no se desea utilizar ninguna Memory Card S7, se puede renunciar en principio a esta opción y guardar los datos de configuración en el CP. Más adelante se puede configurar una Memory Card S7 con los datos de configuración de modo que la opción "Cambio de módulo sin programadora" esté activada dentro de la misma. Si, a continuación, se inserta la Memory Card S7 en la CPU, el CP podrá sustituirse en cualquier momento posterior. En este caso, los datos de configuración se cargarán automáticamente de la CPU o de la Memory Card S7 al arrancar el CP.

Multiplexar conexiones OP / asignar recurso de conexión interno de la CPU

Referencia

Parametrización de ajustes especiales en el grupo de parámetros "Propiedades > General > Ajustes"

Dependiendo del tipo de CP, encontrará la posibilidad de ajuste "Conexiones OP multiplexadas"

Modo multiplex

Para conectar TD/OPs o bien dispositivos HMI, es posible optimizar los recursos de conexión en la CPU S7-300 haciendo que un máximo de 16 dispositivos de este tipo se comuniquen a través de un solo recurso de conexión de la CPU (modo multiplex).

Si no se emplea el modo multiplex, el número de TD/OPs o dispositivos HMI que pueden utilizarse depende del número de recursos de conexión disponibles en la CPU utilizada.

Asignar recurso de conexión interno de la CPU

Las conexiones S7 configuradas a través del CP utilizan el mismo canal multiplex asignado a las conexiones HMI en el modo multiplex. Así pues, si se configuran conexiones S7, se asignará directamente un recurso de conexión de la CPU.

Nota

Las conexiones de la programadora no funcionan a través del multiplexor; para utilizar una programadora se asigna siempre un recurso de conexión.

Nota

Direccionamiento en modo multiplex

En modo multiplex, para las conexiones TD/OP/HMI debe indicarse la asignación de rack/slot del CP en lugar de la asignación de rack/slot de la CPU en el direccionamiento.

Las aplicaciones (p. ej. ProAgent) que requieren avisos de bloque (Alarm_S: SFC 17 , SFC 18 , SFC 19) no se soportan en modo multiplex.

Estación PC

Software SIMATIC NET PC para procesadores de comunicaciones de PC

Software SIMATIC NET PC para procesadores de comunicaciones de PC

Para los siguientes procesadores de comunicaciones se recomienda, dependiendo del sistema operativo, el uso de las versiones del software SIMATIC NET PC indicadas a continuación:

- CP 1613, CP 1613 A2, CP 1623, CP 1628
- CP 5613, CP 5613 A2, CP 5614, CP 5614 A2, CP 5623, CP 5624

Dependiendo del sistema operativo, utilice las siguientes versiones del software SIMATIC NET PC:

- Windows 8 (Professional, Enterprise), 32 y 64 bits, y Windows Server 2012
  DVD "SIMATIC NET PC Software" V12
- Windows 7 (Professional, Enterprise, Ultimate), 32 y 64 bits, y Windows Server 2008 R2
  DVD "SIMATIC NET PC Software" V12
- Windows XP, Windows Server 2003 y Windows Server 2008 32 bits
  CD "SIMATIC NET, PC Software, Edition 2008 + SP6", número de versión V7.1.6

Estación PC - Aplicaciones de usuario servidor OPC / aplicación

Servidor OPC

Ajustes generales específicos de protocolo

En general

Tiempo de ciclo

Repercusión del tiempo de ciclo ajustable

El tiempo de ciclo influye en la frecuencia con que el servidor OPC actualiza en su caché los valores de los OPC Items mediante una petición de comunicación nueva y comprueba los cambios. De este modo, el tiempo de ciclo influye en el rendimiento, la carga de comunicación y la carga del procesador.

Si los valores del tiempo de ciclo son demasiado pequeños se aumenta innecesariamente la carga del procesador.

Relación entre el tiempo de ciclo y el tiempo de actualización (UpdateRate)

El parámetro "Tiempo de actualización" (UpdateRate) que se ajusta desde el programa de usuario define el intervalo de tiempo más pequeño para comprobar los valores de los OPC Items de un grupo OPC activo. Justo después de pasado este intervalo, pero no antes, el servidor comprueba si el valor de los OPC Items ha cambiado. De este modo se limita la carga de comunicación.

Los tiempos de actualización (UpdateRate) utilizados por el servidor OPC para SIMATIC NET son múltiplos del tiempo de ciclo definido durante la configuración. El tiempo de actualización más pequeño posible (UpdateRate mínimo) equivale al tiempo de ciclo.

Relación entre los tiempos de ciclo ajustados de forma específica para los protocolos

Puesto que el servidor OPC SIMATIC NET puede utilizar a la vez varios protocolos,

el UpdateRate mínimo del servidor OPC resulta ser el valor más grande del tiempo de ciclo configurado para los protocolos activos (son aquellos para los que se han configurado conexiones).

Ejemplo:

Para S7 se ha configurado un tiempo de ciclo de 100 ms. No se utiliza ningún otro protocolo. El servidor OPC utiliza el valor 100 ms como tiempo de actualización (UpdateRate) mínimo. Adicionalmente se utiliza el protocolo DP, para el que se ha configurado un tiempo de ciclo de 50 ms. Consecuencia: Ahora el servidor OPC utiliza para todos los protocolos un UpdateRate mínimo de 50 ms.

Observación: El comportamiento descrito es el resultado de las exigencias de la norma OPC. Esta prescribe que los cambios de los datos no pueden notificarse con mayor rapidez que la derivada del tiempo de actualización (UpdateRate) mínimo.

Tiempo de actualización real (UpdateRate)

El tiempo de actualización (UpdateRate) que se alcanza realmente durante el funcionamiento depende de distintos factores, entre ellos:

- Protocolo de comunicación subordinado
- Interlocutores
- Número y tipo de items

Con ello, el tiempo de actualización (UpdateRate) que se alcanza realmente puede ser mayor que el valor mínimo configurado aquí.

Si el valor de proceso cambia varias veces dentro del intervalo de actualización real, no se transferirán todos los cambios de valor a la aplicación.

Observación:

El caudal de datos en peticiones asíncronas y síncronas no depende de los parámetros UpdateRate ni del tiempo de ciclo definido en la configuración.

Protección de acceso

Activar la protección de acceso

Referencia

Ajustes de servidor OPC / grupo de parámetros "Propiedades > General"; el nodo específico de protocolo está seleccionado.

Significado de los derechos de acceso para variables OPC

A través del servidor OPC se pueden definir derechos de acceso para variables o grupos de variables individuales específicamente para cada protocolo. La asignación de derechos de

acceso permite, por ejemplo, impedir que se sobrescriba una variable de cálculo de control interno.

- Derechos predeterminados
  Para objetos OPC no citados específicamente se puede definir un valor estándar:
  - RW: Acceso de lectura y escritura
  - R: Acceso de sólo lectura
  - W: Acceso de sólo escritura
  - NONE: Ni acceso de lectura ni de escritura
- Objeto OPC, derechos específicos

Seleccionando esta opción es posible asignar explícitamente derechos de acceso para objetos OPC determinados en el campo de entrada correspondiente.

Consulte también

Sintaxis - Asignar derechos de acceso específicos de OPC Items (Página 2045)

Sintaxis - Asignar derechos de acceso específicos de OPC Items

Es posible introducir derechos de acceso para uno o varios OPC Items.

Esto no afecta a los derechos predeterminados de los demás OPC Items.

La sintaxis de los derechos de acceso específicos para OPC ITEM está definida del siguiente modo:

<OPCItem>=<derechos>

siendo:

- <OPCItem>
  Indicación de uno o varios OPC Items de acuerdo con la sintaxis definida en la documentación OPC. Está permitido utilizar alias.
  Pueden emplearse los comodines siguientes:
  * un número cualquiera de caracteres irrelevantes
  ? un carácter exactamente es irrelevante
- <derechos>

RW	Acceso de lectura y escritura
R	Acceso de sólo lectura
W	Acceso de sólo escritura
NONE	Ni acceso de lectura ni de escritura

Reglas de evaluación

- Los derechos asignados aquí específicamente tienen prioridad sobre los derechos predeterminados que se han asignado en el área de entrada Protección de acceso.
- Los derechos asignados aquí específicamente se evalúan en el orden introducido aquí. En caso de asignación múltiple a un OPC Item siempre es válida la última asignación.
  Ejemplo:
  DP:[CP 5611]Slave040_AB*=RW
  DP:[CP 5611]Slave040_AB1=R
  DP:[CP 5611]Slave040_AB2=W
  DP:[CP 5611]Slave040_AB1=W
  DP:[CP 5611]Slave040_AB1*=R
  Debido a esta entrada tienen efecto los derechos de acceso siguientes, entre otros:
  DP:[CP 5611]Slave040_AB2=W
  DP:[CP 5611]Slave040_AB1=R
  Sugerencia: observe la descripción detallada de la sintaxis en la documentación OPC citada más abajo. Allí encontrará una serie de ejemplos, incluso para aplicaciones como PROFINET IO.
- Los ceros a la izquierda en los items y en las definiciones de los derechos de acceso se ignorarán por principio.
  El ejemplo siguiente explica la ventaja resultante:
  Se configura el derecho de acceso siguiente: Slave004_AB0=R
  Con ello se ha asignado implícitamente el derecho de acceso correspondiente a los items siguientes, p. ej.:
  - Item 1: Slave00004_AB000
  - Item 2: Slave4_AB00,1 (observación: la sintaxis con ",1" la predetermina el OPC Scout)

  Para el uso de los comodines "*" y "?" resulta de ello:
  No utilice dichos comodines en lugares en los que puede haber un cero a la izquierda en el item.
  Ejemplo: Ceros a la izquierda en items - ejemplo con comodines:

Información relacionada

Observe la descripción detallada de la sintaxis en la documentación OPC. Allí encontrará una serie de ejemplos, incluso para aplicaciones como PROFINET IO.

Ceros a la izquierda en items - ejemplo con comodines:

Se utiliza el item siguiente:

- Slave00004_AB000

Una definición del derecho de acceso que fallaría en la comparación sería:

- Slave0?4_AB0=R

Razón:

En la comparación se detecta un "0" a la izquierda en el item, de modo que éste se reduce a: Slave4_AB

A causa del comodín "?", en la definición el "0" antepuesto ya no se reconoce como un "0" a la izquierda durante la comparación. Por consiguiente, la definición del derecho de acceso no se reduce y la comparación falla.

Solución:

Para obtener la asignación de derechos deseada en este ejemplo, deberá realizar en su lugar las dos definiciones siguientes:

- Slave?4_AB0=R
- Slave4_AB0=R

Ajustes específicos de S7

Símbolos OPC

Utilizar símbolos

Referencia

Ajustes de servidor OPC / grupo de parámetros "Propiedades > S7 > Símbolos OPC".

Utilizar símbolos (a partir del servidor OPC V6.1)

Es posible determinar si deben incluirse símbolos en los datos de sistema para el servidor OPC desde las tablas de símbolos configuradas para una CPU. Esto es imprescindible para que las aplicaciones de usuario (cliente OPC) puedan depositar accesos simbólicos a variables a través del servidor OPC.

Se consultan las tablas de símbolos de las CPUs para las que se han configurado conexiones S7 para el servidor OPC. En este caso sólo se tendrán en cuenta los símbolos que hacen referencia a bloques de datos (DB).

Las alternativas que pueden elegirse aquí son:

- Ninguno
  No se incluirá ningún símbolo de STEP 7 en los datos de sistema del servidor OPC. Son válidos los derechos de acceso configurados para los OPC Items en la protección de acceso.
- Todos
  Se incluirán todos los símbolos relevantes de STEP 7 en los datos de sistema del servidor OPC.
  Los derechos de todos los accesos simbólicos a las variables asignadas se establecen en Read/Write (RW) con esta selección. Tienen prioridad sobre los derechos de acceso de los OPC_Items seleccionados en el área Protección de acceso.

- Configurar
  En los datos de sistema para el servidor OPC se incluyen símbolos seleccionados y configurados especialmente de STEP 7. A través del botón "Configurar" se accede al cuadro de diálogo "Configuración de símbolos", en el que se pueden seleccionar y configurar símbolos.
- Elementos de campo visibles en runtime (seleccionable a partir de servidor OPC V8.0)
  Se puede decidir si los elementos de matrices deben ser visibles en el servidor OPC en runtime. Los afectados son los campos (matrices) que están configurados como símbolos. La opción está desactivada por defecto, ya que la visibilidad de los elementos de campo puede traducirse en una gran carga de tiempo en el servidor OPC. En versiones anteriores de servidor OPC (< V8.0), los elementos de campo están visibles por principio y no es posible desactivar esta opción.

Nota

Coherencia

Si se selecciona la opción "Todos", al guardar y compilar se tendrán en cuenta las tablas de símbolos actuales de la CPU.

Si se utiliza la opción "Configurados", hay que tener en cuenta que los cambios posteriores en las tablas de símbolos tienen las siguientes consecuencias en los datos del sistema para el servidor OPC:

- Los símbolos agregados no se tienen en cuenta
- Los símbolos seleccionados retirados posteriormente de la tabla de símbolos no pueden guardarse.

Nota

Matrices de bits

Los símbolos que contienen datos del tipo "matriz de bits" no se incluirán en los datos de sistema del servidor OPC.

Nota

Acerca del procedimiento

Durante la aplicación de símbolos aquí descrita no puede estar abierto en KOP / AWL / FUP ningún FB en el que se utilicen símbolos. En caso contrario no se leerán todos los símbolos al aplicarlos.

Configuración de símbolos

Referencia

Ajustes de servidor OPC / grupo de parámetros "Propiedades > S7 > Símbolos OPC".

Acceso a la configuración de símbolos a través del botón "Configurar símbolos".

Resumen

En este cuadro de diálogo se pueden seleccionar y configurar los símbolos que deben aplicarse en los datos de sistema del servidor OPC.

Para la plantilla de los símbolos son relevantes las CPUs para las que hay configuradas conexiones S7 para el servidor OPC.

En el área de navegación, a la izquierda, se muestran las estaciones y CPUs correspondientes, y, dado el caso, los elementos de estructura. En el área de contenido, a la derecha, se muestran los símbolos definidos para la CPU o el elemento de estructura seleccionado. Para cada símbolo se indican el tipo de datos asignado y las propiedades configuradas en ese momento. Estas propiedades pueden configurarse en la parte inferior del cuadro de diálogo.

Seleccionar y configurar símbolos

1. Seleccione primero en el área de navegación el grupo de símbolos de una CPU o de un elemento de estructura.
2. A continuación, seleccione uno o varios símbolos en el área de contenido con el cursor del ratón.
3. Configure las propiedades de los símbolos seleccionados en la parte inferior del cuadro de diálogo según las posibilidades descritas a continuación.

Indicación "Elementos con rama activa"

La indicación tiene el siguiente significado:

- Símbolos: Número de elementos presentes en total en la rama o el elemento de estructura seleccionado (los elementos de estructura seleccionados también se cuentan).
- Elementos de matriz: Número de elementos en la matriz
  - Indicación de valor izquierda: Número de elementos marcados como "visibles"
  - Indicación de valor derecha: Total de elementos

Propiedades configurables

- Opción "Visible"
  Si se selecciona esta opción, el símbolo seleccionado se excluye del proceso de carga al servidor OPC.
- Opción "Cargar solo símbolos visibles"
  Si se selecciona esta opción, solo se aplicarán en los datos de sistema del servidor OPC los símbolos marcados como "visibles", con lo que serán visibles para el cliente OPC. La opción reduce considerablemente el tiempo de carga al servidor OPC si hay muchos símbolos marcados como "no visible".

- Derechos de acceso
  A los símbolos seleccionados se les pueden asignar los siguientes derechos de acceso desde la lista desplegable:
  - RW: Lectura y escritura
  - R: Leer
  - W: Escribir
  - None: Ningún acceso
    Los derechos aquí seleccionados tienen preferencia ante los derechos de acceso configurados para los objetos OPC en la protección de acceso.
- Conexión
  En la lista desplegable de conexiones configuradas, seleccione la conexión S7 a través de la cual debe producirse el acceso simbólico a variables.
- Campos de entrada "Edit EU Lo / Hi" - Definir el rango de tolerancia para valores medidos (valores límite EU Lo y EU Hi)
  Defina aquí valores límite para la determinación de un rango de tolerancia para los objetos OPC seleccionados. Los campos de entrada están activos si se da la siguiente condición:
  - Solo se ha seleccionado un símbolo;
  - El símbolo tiene asignado un tipo de datos elemental que admite la representación numérica (INT, DINT, REAL, BYTE, WORD, DWORD).
    Observación:
    La entrada solo es posible para servidores OPC a partir de la versión 6.4.
    Véase también Introducir valores límite para EU Lo y EU Hi (Página 2051)

Indicación "Ruta de acceso"

Aquí se indica la ruta de almacenamiento en el proyecto STP 7 actual para el objeto seleccionado en el área de navegación.

Particularidades

- Las matrices de bits no se aplican
  Los símbolos que contienen datos del tipo "matriz de bits" no se incluyen en los datos de sistema del servidor OPC. Esta restricción es válida aunque los símbolos se muestren dentro de la selección posible. También es válida independientemente de si se ajusta la opción "Visible" (ver arriba) para los símbolos.
- La representación de los tipos de datos "Array of UDT" y "Array of Struct" es diferente
  En el caso de los símbolos que contienen los tipos de datos compuestos "Array of UDT" o "Array of Struct" la estructura del tipo de datos "UDT" o "Struct" solo se muestra una vez en la parte izquierda del cuadro de diálogo. El hecho de que se trate de una matriz se reconoce por la indicación "Matriz" (ARRAY) en el campo "Tipo de datos" de la parte derecha.
  Al contrario de lo que ocurre en esta forma de representación, en otras herramientas, como el configurador de archivos de símbolos, los elementos de matriz se muestran de forma múltiple.
  Sin embargo, en principio se trata de estructuras de datos idénticas.
  Recuerde que el ajuste de la opción "visible" arriba descrita es efectivo para todos los elementos de una matriz; es decir, también para los elementos de matriz no representados.

Introducir valores límite para EU Lo y EU Hi

Introducir un rango de tolerancia para valores medidos (valores límite EU Lo y EU Hi)

Defina aquí valores límite para la determinación de un rango de tolerancia para los objetos OPC seleccionados (vea el significado más abajo).

Pueden indicarse valores numéricos como número entero, coma flotante o notación exponencial ("notación de ingeniería")

Nota: Como separador debe utilizarse el carácter "." en lugar del carácter ",".

- EU Lo (Engineering Unit Low)
  Límite inferior
- EU Hi (Engineering Unit High)
  Límite superior

Debe regir: EU Lo <= EU Hi

A los campos de entrada vacíos se les asigna automáticamente "0".

Observe los rangos de valores dependientes del tipo de datos: Rango de valores de los tipos de datos (Página 2052)

- Ejemplos:
  - EU Lo = 10 EU Hi = 25
  - EU Lo = 4.5 EU Hi = 14.5
  - EU Lo = 1.2e-2 EU Hi = 2.5e-2
- Observaciones:
  - La indicación del rango de tolerancia ("Percent Deadband" = rango de tolerancia porcentual) corresponde en cada caso a un grupo de objetos OPC.
  - La entrada solo es posible para servidores OPC a partir de la versión 6.4.

Significado de EU Lo y EU Hi

Pueden indicarse rangos de tolerancia para el registro de valores medidos. Un valor medido puede variar dentro de un rango de tolerancia ("Percent Deadband" = rango de tolerancia porcentual) sin que la variación se notifique.

Para ello se determina un valor relativo a partir de los parámetros indicados y según la siguiente fórmula. Este valor indica cuántas unidades puede variar como máximo una magnitud física hacia arriba o hacia abajo sin que el cambio se notifique.

A = PD / 100 x (EU HI – EU Lo)

---

siendo:

A = cantidad correspondiente a la variación (hacia arriba o hacia abajo)

PD = Percent Deadband / %

EU Lo = Engineering Unit Low

EU Hi = Engineering Unit High

Ejemplo

En una curva de medición de temperatura partimos del hecho de que en el rango entre 30°C y 40°C se da el rango de tolerancia típico para el valor medido. Como rango de tolerancia porcentual ("Percent Deadband") se fija el 10%.

Por ello se definen los siguientes valores:

EU Lo = 30

EU Hi = 40

PD = 10%

De ello se deriva: A = 10 / 100 x (40 – 30) = 1

Así, un cambio de temperatura solo se notifica como cambio si varía más de 1°C por encima o por debajo del valor medido anterior registrado.

Consulte también

Configuración de símbolos (Página 2048)

Rango de valores de los tipos de datos

Se aplican los siguientes rangos de valores para los tipos de datos utilizados:

BYTE= 0 .. 256

INT= -32768 ... 32767

WORD= 0 ... 65535

DWORD= 0 ... 4294967295.0

DINT= -2147483648 ... 2147483647.0

REAL POS UPPER = 3.402823e+38

REAL POS UNDER = 1.175495e-38

REAL NEG UPPER = -1.175495e-38 (límite superior negativo)

REAL NEG UNDER = -3.402823e+38 (límite inferior negativo)

Consulte también

Introducir valores límite para EU Lo y EU Hi (Página 2051)

Avisos de OPC

Configurar avisos de OPC

Referencia

Ajustes de servidor OPC / grupo de parámetros "Propiedades > S7 > Avisos de OPC".

Idioma / Ajuste de idioma del texto de aviso

Para avisos de OPC se define qué versiones de idioma de la librería de textos deben cargarse en el servidor OPC para los textos de aviso (avisos de bloque y de símbolos).

Mediante las dos casillas de verificación se ajusta lo siguiente:

- Idioma predeterminado (definido como predeterminado para visualizadores):
  Si el idioma predeterminado para avisos es uno de los siguientes idiomas estándar de STEP 7: Alemán (Alemania), Inglés (EE.UU.), Francés (Francia), Español (alfabetización tradicional), Italiano (Italia). En cualquier otro caso, la casilla de verificación está desactivada.
- Inglés (USA)
  (si está instalado el inglés (USA) como idioma estándar)
  Si el idioma estándar para visualizadores ya es de por sí el inglés (USA), la casilla de verificación estará desactivada.

Se puede seleccionar una de las dos opciones. Si se seleccionan ambas opciones, se comprueba si en el proyecto, en "Idioma para visualizador" se ha seleccionado una memorización del texto en varios idiomas. Si es así, en la descarga a la estación PC se cargan las librerías de texto correspondientes.

Sellos de tiempo para avisos

En este grupo de parámetros, seleccione de qué instancia debe colocarse el sello de tiempo de un aviso. Si se deja que sea la CPU la encargada de colocar sellos de tiempo, los avisos pueden registrarse exactamente con la indicación del momento en que se produjeron. La posible indicación de un offset de tiempo también permite registrar en un sistema de registro los avisos que se producen, por ejemplo, en instalaciones con diferentes zonas horarias, con una hora de referencia común.

- Sello de tiempo de la CPU
  Los avisos obtienen el sello de tiempo de la CPU que notifica el aviso. Excepción: Los StatepathEvents son avisos de generación local y obtienen la hora del PC (UTC).
- Hora de la CPU + offset
  Los avisos reciben el sello de tiempo de la CPU que notiifica el aviso; a dicho sello se le añade un retardo ajustable con precisión de segundo (offset). Excepción: Los StatepathEvents son avisos de generación local y obtienen la hora del PC (UTC).
  Rango de retardo: de -23:59:59 a + 23:59:59
- Hora del PC (UTC)
  En el momento en que se producen, los avisos reciben el sello de tiempo (UTC) de la estación PC que lo recibe (servidor OPC).

Tabla de asignación de avisos de OPC

Tabla de asignación de avisos de OPC

Referencia

Ajustes de servidor OPC / grupo de parámetros "Propiedades > S7 > Tabla de asignación de avisos de OPC".

Significado

Las categorías de avisos configuradas en un proyecto en "Datos comunes > Categorías" se representan en una tabla. Las categorías tienen asignada por defecto la categoría OPC "OffNormal". En caso necesario, asigne una categoría OPC diferente.

Acceso S7 optimizado a datos del PLC para servidores OPC

Para el acceso optimizado del servidor OPC V12 a bloques de datos, marcas, entradas, salidas, contadores y temporizadores sin seguridad de tipos de un SIMATIC S7-1500 y S7-1200 existe la posibilidad de configurar una conexión S7 optimizada con la CPU.

Las conexiones S7 para el acceso optimizado a datos de un S7-1500 o S7-1200 solo son posibles vía Industrial Ethernet.

Requisitos

- STEP 7 Professional V12 SP1
- Servidor OPC V12
- S7-1500 o S7-1200 V4.0 o superior

Nombre de la conexión para servidor OPC V12 y conexiones S7

Al crear una conexión S7 para un servidor OPC, STEP 7 propone un nombre válido. Existe la posibilidad de modificar en cualquier momento el nombre de la conexión S7. Algunos caracteres y nombres de conexión están ocupados por el sistema y no pueden utilizarse para asignar el nombre de la conexión. STEP 7 comprueba automáticamente que el nombre de la conexión no contenga caracteres inadmisibles.

Tabla de conexiones

En la tabla de conexiones de la vista de redes de la ficha "Conexiones" se muestran las conexiones creadas por el usuario junto con sus parámetros. En la columna "Interlocutor" existe la posibilidad de modificar el interlocutor del servidor OPC V12.

Parámetros de las propiedades especiales de la conexión

En la ventana de inspección de la conexión S7 unilateral seleccionada están activadas las casillas de verificación "Unilateral" y "Establecimiento activo de la conexión" en "Propiedades > Configuración > Propiedades especiales de la conexión". En el grupo de parámetros "Propiedades especiales de la conexión" no es posible realizar cambios.

Detalles de direcciones

En la ventana de inspección de la conexión S7 unilateral seleccionada se ven los TSAPs de los interlocutores en los recursos de la conexión, en "Propiedades > Configuración > Detalles de direcciones". El TSAP local tiene la estructura siguiente:

SNOPCCxxxxyyyyyy

- SNOPCC = SIMATIC NET OPC Configured connection
- xxxx = índice del servidor OPC V12
- yyyyyy = ID de la conexión S7

El TSAP del interlocutor es "SIMATIC-ROOT-OTH".

En el grupo de parámetros "Detalles de direcciones" no es posible realizar cambios.

Parámetros OPC

En la ventana de inspección de la conexión S7 unilateral seleccionada existen las siguientes propiedades de configuración en "Propiedades > Configuración > OPC".

- Activación del acceso optimizado
- Configuración del tamaño del búfer para el acceso optimizado a variables
- Administración del acceso optimizado a variables introduciendo la contraseña de acceso

Tamaño del búfer:

El tamaño predeterminado del búfer puede modificarse para casos de aplicaciones especiales con una capacidad especialmente elevada de variables de datos en una CPU (S7-1500 o S7-1200 V4.0). El acceso a variables de datos colindantes dentro de un área de datos se optimiza para mejorar el rendimiento. Los resultados óptimos se consiguen con tamaños de búfer que abarquen como mínimo la mayoría de variables de datos colindantes a las que se accede con regularidad y frecuencia.

Ejemplo: desde el bloque de datos "db10" se leen regularmente muchas variables de datos entre los bytes 1000 y 1999 y, en contadas ocasiones, también una variable de datos del byte 30000. En este caso basta con un tamaño de búfer de 1000 bytes.

Recursos de conexión

En las conexiones S7 configuradas para un servidor OPC V12 o superior con el fin de optimizar el acceso a un PLC S7-1500 o S7-1200 V4.0 o superior se muestran los recursos de conexión necesarios para ello en el lado del PLC. Los recursos de conexión necesarios se visualizan en las propiedades de la CPU correspondiente, en "Propiedades > General > Recursos de conexión" de la ventana de inspección.

Para conexiones S7 para el acceso optimizado de un servidor OPC V12 o superior, en el lado del PLC los recursos de este tipo de conexiones no se asignan a la fila "Comunicación S7" sino a las filas "Otra comunicación" (S7-1500) o "Conexiones libres disponibles" (S7-1200 V4.0 o superior). En una conexión S7 configurada para un servidor OPC V12 con un S7-1500 o S7-1200 V4.0 o superior se requieren varios recursos de conexión.

Sustituir servidor OPC

Desde el menú contextual "Sustituir dispositivo" de un servidor OPC seleccionado existe la posibilidad de sustituir un servidor OPC V12 que permite el acceso optimizado a S7-1500 o S7-1200 V4.0, por un servidor OPC < V12 y viceversa.

En tanto que se hayan configurado ya conexiones S7 para el servidor OPC que debe sustituirse, durante el intercambio de servidores se mostrará una nota indicando que las conexiones S7 existentes se borrarán. Una vez realizado el intercambio deberán crearse de nuevo.

Visualización de una conexión S7 unilateral entre un servidor OPC V12 y un S7-1500 o S7-1200 V4.0 o superior

En cuanto un servidor OPC V12 haya solicitado correctamente el acceso optimizado a datos de una S7-1500 o S7-1200 V4.0 o superior, dicha conexión se mostrará en STEP 7 en la tabla de conexiones online del PLC correspondiente.

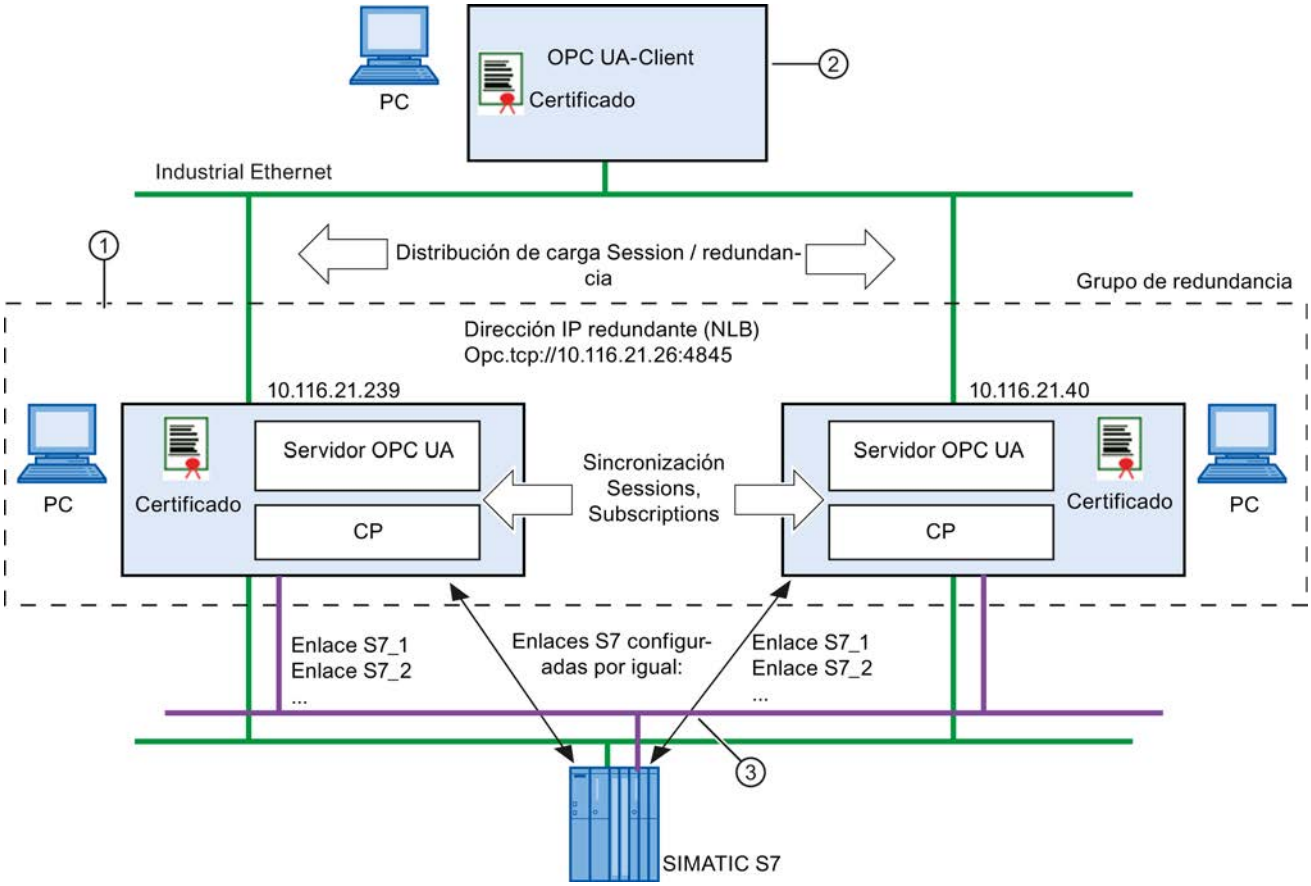
Este tipo de conexiones también se muestran en la ventana de inspección "Información de la conexión" de la CPU online seleccionada: aquí se visualizan los recursos de conexión online utilizados en el PLC.

Servidor OPC UA redundante

Servidor OPC UA redundante - Principios básicos

Servidor de redundancia OPC UA transparente

La siguiente representación muestra cómo, en la solución de redundancia OPC UA transparente, los servidores OPC UA están incrustados en un grupo de redundancia y configurados en correspondencia.



① Se configura una estación PC para cada servidor OPC UA redundante. Las estaciones PC redundantes asignadas a un cliente OPC UA forman un grupo de redundancia. Esta asignación se realiza mediante la configuración unitaria de la "dirección IP del grupo de redundancia" en los servidores OPC UA.

② El cliente OPC UA que utiliza la redundancia está en una estación PC de nivel superior.

La estación PC de nivel superior no debe configurarse para el uso de la redundancia OPC UA en el proyecto de STEP 7.

③ La comunicación entre los servidores OPC UA y los autómatas programables se consigue mediante conexiones S7 que se han creado de forma múltiple y se han configurado de forma idéntica.

Según sea la configuración, es posible utilizar conexiones S7 vía Ind.Ethernet o PROFIBUS.

Parametrizar un servidor OPC SIMATIC NET como servidor S7 OPC UA para la redundancia OPC UA transparente

La redundancia transparente utilizada por el cliente OPC UA de nivel superior se consigue básicamente del modo siguiente en la configuración:

1. Para utilizar estaciones PC con servidor OPC UA SIMATIC NET redundante, se crea en cada caso un componente del tipo servidor OPC >= V8.1. Para estos servidores OPC se activa la redundancia S7 UA.
2. Se adopta la dirección IP virtual. Esta dirección IP virtual se ha definido previamente en el administrador de compensación de carga de red (NLB) para ambas estaciones PC. La dirección IP virtual se reproduce en las direcciones IP reales del servidor OPC UA.
3. Se define un puerto que permite a los dos servidores OPC UA comunicarse con el fin de sincronizar los datos;
4. Se configuran conexiones S7 idénticas para la comunicación con los sistemas de automatización conectados para ambos servidores OPC UA;
5. Se asignan a las estaciones PC certificados de autenticación generados automáticamente. En caso necesario, se asignan certificados de autenticación nuevos a las dos estaciones PC redundantes.

Comprobación de coherencia

Al guardar y compilar la configuración se lleva a cabo una comprobación de coherencia. En relación a la redundancia OPC UA se realizan las comprobaciones siguientes:

- Las direcciones de servidor a servidor deben ser unívocas dentro de un grupo de redundancia. Del mismo modo, dentro de un proyecto solo está permitido utilizar una vez una dirección IP común de un grupo de redundancia.
- Cada miembro de un grupo de redundancia debe disponer de las mismas conexiones S7 con los mismos nombres e interlocutores.
- La configuración de hardware de las estaciones PC en los grupos de redundancia debe ser idéntica, es decir, los módulos utilizados para la función de redundancia deben ser idénticos en las estaciones PC redundantes.
- Los parámetros de las conexiones S7 tienen que ser idénticos; entre ellos se incluyen el establecimiento de la conexión, los avisos, las optimizaciones, el tamaño de PDU, el número de peticiones paralelas y los tiempos de vigilancia.
- Los ajustes S7 de los servidores OPC UA tienen que ser idénticos; entre ellos se incluyen el tiempo de ciclo, los derechos de acceso, las alarmas S7 y los símbolos S7.

Nota

Recomendación: conexiones S7 establecidas de forma permanente

Se recomienda encarecidamente configurar conexiones S7 establecidas de forma permanente y activa en los servidores OPC UA. De este modo se consiguen tiempos de conmutación más reducidos en caso de fallo de un servidor.

Procedimiento / recomendación

Las parametrizaciones de los servidores OPC UA redundantes deben corresponderse entre sí. Al guardar y compilar la configuración se lleva a cabo una comprobación de coherencia correspondiente.

Si se aprovechan en STEP 7 las posibilidades de copia, se reduce el peligro de que se produzcan indicaciones incoherentes. Es posible copiar toda la estación PC o bien solo el servidor OPC. Si ya se han configurado conexiones S7, también se copiarán. Todavía debe asignarse el interlocutor correspondiente a las conexiones de la estación PC copiada.

Consulte también

Configurar la redundancia S7 UA (Página 2059)

Configurar la redundancia S7 UA

Referencia

Parametrización de servidores OPC UA redundantes en el grupo de parámetros "Propiedades > General > Redundancia S7 UA

Redundancia OPC UA transparente

En este grupo de parámetros se configura la estación PC como servidor OPC UA redundante para el funcionamiento con la redundancia OPC UA transparente. Para ello se utiliza el servidor OPC SIMATIC NET en su versión 8.1

Las estaciones PC redundantes asignadas a un cliente OPC UA forman parte de un grupo de redundancia.

En este grupo de parámetros, proceda del siguiente modo:

1. Seleccione la opción "Activar redundancia".
2. Introduzca la dirección IP del grupo de redundancia. Existen para ello las siguientes posibilidades:
   - Introducir la dirección IP directamente o bien aplicar o modificar la dirección IP propuesta.
     Elija esta opción cuando se cree un grupo de redundancia nuevo.
   - Seleccionar el botón "Asignar...".
     Elija esta opción para asignar el servidor OPC a un grupo de redundancia que ya esté creado.

   Nota

   Después de la entrada o asignación, tenga en cuenta la tabla "Direcciones IP de los demás miembros del grupo de redundancia" al final del grupo de parámetros. Verá qué servidores OPC están asignados al mismo grupo de redundancia.

3. Defina el puerto de redundancia para la comunicación con el cliente OPC UA; dado el caso, aplique el ajuste predeterminado.
4. Defina los parámetros de los servidores para la comunicación entre servidores.
5. Dado el caso, compruebe el certificado de autenticación asignado al servidor OPC mediante el botón "Visualizar...".

Parámetros / Áreas de entrada / Botones

- Opción "Activar redundancia"
  Al activar esta opción se habilita la posibilidad de parametrización para la funcionalidad de servidor OPC UA.
- Botón "Asignar..."
  Se abre el cuadro de diálogo en el que pueden seleccionarse grupos de redundancia previamente creados.
- Parámetro "Dirección IP del grupo de redundancia"
  Dirección IP virtual mediante la que el cliente OPC UA accede al servidor OPC UA redundante (el grupo de redundancia).
  Esta dirección IP virtual se define en el administrador de compensación de carga de red (NLB) para ambas estaciones PC. La dirección IP virtual se reproduce en las direcciones IP reales del servidor OPC UA.
- Parámetro "Puerto de redundancia"
  Puerto por el que se lleva a cabo la comunicación con el cliente OPC UA.
  Predeterminado: 4845
- Parámetro "Puerto servidor-a-servidor"
  Puerto por el que se lleva a cabo la comunicación para fines de sincronización con los servidores OPC UA del grupo de redundancia.
  Predeterminado: 4846
- Parámetro "Interfaz servidor-a-servidor"
  Seleccione aquí la interfaz hacia la subred por la que se lleva a cabo la sincronización con los servidores OPC UA en el grupo de redundancia.
- Área de entrada "Certificado" / botón "Visualizar..."
  STEP 7 genera automáticamente un certificado que se transmite al cliente OPC UA al establecer la conexión. El cliente OPC UA lo utiliza para la autenticación en los procesos de comunicación subsiguientes.
  El certificado también se emplea para transferir los códigos para la codificación de los datos.
  El código privado permanece en el servidor y se utiliza para la codificación. A continuación, utilizando el código público correspondiente obtenido del servidor mediante el certificado, el cliente puede descodificar los mensajes cifrados del servidor.
  Aquí puede visualizar y comprobar el certificado válido actualmente.
  Observación:
  la visualización del certificado no detecta si ya se está visualizando un certificado. Cada vez que se pulsa el botón "Visualizar..." se abre una nueva visualización de certificado.
- Área de entrada "Certificado" / botón "Actualizar..."
  Se ordena a STEP 7 que cree un certificado nuevo.

Direcciones IP de los demás miembros del grupo de redundancia

Visualización de los demás miembros del grupo de redundancia. A un grupo de redundancia pertenecen las estaciones PC que tienen asignada la misma "dirección IP del grupo de redundancia".

Nota

Número máximo de miembros en un grupo de redundancia

En la versión actual de STEP 7 V12 se soportan **2** servidores OPC UA dentro de un grupo de redundancia.

Consulte también

Servidor OPC UA redundante - Principios básicos (Página 2057)

Asignar grupo de redundancia S7 UA

Referencia

Parametrización de servidores OPC UA redundantes en el grupo de parámetros "Propiedades > General > Redundancia S7 UA > Seleccionar maestro de redundancia S7 UA

Asignar un grupo de redundancia S7 UA existente

La tabla muestra los grupos de redundancia disponibles y seleccionables en el proyecto de STEP 7 junto con sus direcciones IP.

Siempre que se haya asignado una dirección IP nueva al servidor OPC actual, aparece la entrada siguiente en sustitución de dicha asignación:

- "(grupo de redundancia propio <dirección IP>)"

Proceda del siguiente modo:

1. Seleccione el grupo de redundancia deseado;
2. Confirme el cuadro de diálogo con el botón "Aceptar".

De este modo, la dirección IP del grupo de redundancia seleccionado se incorpora al grupo de parámetros "Redundancia S7 UA".

Consulte también

Servidor OPC UA redundante - Principios básicos (Página 2057)

Mensajes de CPU para servidores OPC

Servidor OPC SIMATIC NET: recomendaciones para la configuración de avisos de CPU

Si configura avisos de CPU de un servidor SIMATIC NET OPC de una estación PC, STEP 7 no comprobará si están configurados de forma coherente. Para conseguir que los avisos funcionen correctamente, tenga en cuenta las siguientes recomendaciones:

- Para recibir avisos configure por cada CPU una sola conexión S7 apta para alarmas.
- Utilice un solo idioma para los textos de aviso en cada proyeccto. El uso de textos de aviso en varios idiomas no es compatible.
- No introduzca fuentes específicas del usuario (texto adicional 1) en los avisos. Utilizando las fuentes predeterminadas se garantiza la univocidad de los mensajes.

Estación PC CP Ethernet

Inserción y extracción de CPs PC Ethernet

Inserción y extracción de CPs PC Ethernet

Los módulos PC Ethernet, como p. ej. las tarjetas de red USB, ya deben estar insertadas y activadas en el momento del arranque del sistema. Si algún módulo Ethernet está desactivado o no está presente en el momento de arrancar el sistema o bien se extrae y se vuelve a insertar durante el funcionamiento, la función "Dispositivos accesible" de STEP 7 no mostrará todos los dispositivos. Esto afecta también al cambio de estaciones de acoplamiento si el módulo Ethernet no está insertado en el portátil, sino en estas estaciones.

Active los módulos Ethernet desactivados en "Panel de control > Conexiones de red" y reinicie el sistema. Reinicie también el sistema si los módulos Ethernet fueron insertados durante el funcionamiento o si se cambió la estación de acoplamiento con el módulo Ethernet durante el funcionamiento.

Configuración OPC (CP Ethernet)

Referencia{"Configuración OPC"}

Parametrización de la configuración OPC en el PC CP Ethernet, en el grupo de parámetros "Propiedades > General > Configuración OPC".

El grupo de parámetros está presente cuando en el estación PC hay un servidor OPC configurado.

Opción "Acceso a través del servidor OPC"

Active el módulo para el funcionamiento a través de un servidor OPC configurado en el dispositivo.

Opción "Modo automático del controlador PROFINET IO"

Al seleccionar esta opción, el estado operativo del controlador PROFINET IO cambia automáticamente a OPERATE.

En estado operativo OPERATE, el controlador PROFINET IO tiene acceso al sistema PN IO conectado.

Esta opción está disponible cuando se ha seleccionado la opción "Acceso a través del servidor OPC".

- Activar opción (ajuste predeterminado)
  Si se elige esta opción, el controlador PROFINET IO pasará al estado operativo OPERATE en cuanto se produzca un acceso a un dispositivo IO. Dicho acceso puede producirse a través de la interfaz OPC-Item o de la interfaz de programación de usuario IO-Base, por ejemplo.
  Para inicializar el estado operativo no debe haber otras llamadas anteriores.
- Desactivar opción
  El controlador PROFINET IO debe ponerse al estado operativo OPERATE antes de acceder a los datos. Utilice para ello las llamadas de inicialización correspondientes de una interfaz en el programa de usuario.
  La interfaz del programa de usuario puede ser una interfaz OPC-Item o una interfaz de programación de usuario IO-Base.

Interfaz Ethernet

Sincronización horaria

Sincronización horaria

Aquí se determina si el CP debe procesar telegramas de hora o no.

Según donde debe encontrarse el reloj maestro, se ofrecen las dos opciones siguientes:

- Maestro (emisor de hora)
  Los mensajes de hora pueden provenir de la estación PC y transferirse a la subred (LAN). Una aplicación PC que ejecuta la función de reloj maestro transfiere la información de hora al CP a petición.
- Esclavo (receptor de hora)
  Los mensajes de hora pueden provenir de la subred (LAN) y transferirse a la estación. Una aplicación PC puede leer la hora transmitida con una petición adecuada.

Si se utilizan varios CPs en la estación, debería considerarse el flujo de telegramas de hora en función del reloj maestro, ya que con la función aquí descrita es posible acoplar telegramas de hora de una red a otra.

Si en una estación hay varios CPs conectados a la misma red, sólo **uno** de estos CPs puede transferir los mensajes de sincronización horaria.

Opciones avanzadas

Enviar Keep Alive para conexiones

Referencia

Parametrización de ajustes especiales para la interfaz Ethernet en el grupo de parámetros "Propiedades > General > Interfaz Ethernet> Opciones avanzadas > Opciones de interfaz"

Enviar Keep Alive para conexiones

Si el mecanismo de Keep Alive está activado, en caso de conexiones se envían telegramas de señal de vida a través de TCP/IP. De este modo se garantiza que las conexiones finalicen después de que falle un interlocutor y que los recursos de conexión se liberen.

Según el tipo de módulo se ofrecen 2 posibilidades para ajustar el mecanismo de Keep Alive:

- Keep Alive activable/desactivable con un intervalo fijo
  En este caso se puede activar y desactivar el mecanismo de vigilancia de la conexión. El intervalo de tiempo en el que se enviarán telegramas de señal de vida (Keep Alive) al interlocutor de una conexión está determinado en 30 segundos para el módulo.
  - Keep Alive desactivado: 0 segundos
  - Keep Alive activado: intervalo fijo de 30 segundos
- Keep Alive con intervalo ajustable
  Aquí se puede ajustar el intervalo de tiempo en el que se enviarán telegramas de señal de vida (Keep Alive) al interlocutor de una conexión.
  Rango de valores:
  - Ajuste predeterminado: 30 segundos
  - Valor máximo: 999 999 segundos
  - Keep Alive desactivado: 0 segundos

Indicaciones / recomendaciones:

Tenga en cuenta que el mecanismo de Keep Alive puede provocar que se mantengan conexiones subordinadas (p. ej. una conexión telefónica RDSI) aunque no se transfieran datos de usuario propiamente dichos. Si no desea que esto suceda, debe:

- Desactivar el mecanismo de Keep Alive (intervalo =0)

o

- ajustar un intervalo de tiempo tan alto que la conexión subordinada finalice tras la ausencia de datos de usuario antes de que se envíe un telegrama de Keep Alive.

Nota

Módulos de otros fabricantes

Si utiliza módulos de otros fabricantes observe sus indicaciones respecto de la configuración de Keep Alive.

Indicaciones sobre la vigilancia del establecimiento de conexión

Relación con la configuración de conexiones

La vigilancia de tiempo que se ajusta aquí está relacionada del siguiente modo con la vigilancia de tiempo que puede ajustarse en la configuración de conexiones: el valor válido es el más pequeño.

En la configuración de conexiones, la vigilancia de tiempo se configura en "Propiedades > Detalles del establecimiento de conexión".

Web

Servidor web en el CP PC

Referencia

Parametrización para los accesos al servidor web del CP PC en el grupo de parámetros "Servidor web"

Opción "Activar webserver en este módulo"

El CP le ofrece la función de un servidor web para el acceso mediante navegador web por el puerto 80. Para ello, se almacenan determinadas páginas HTML con información del CP y funciones de diagnóstico en un área de memoria del CP.

Active esta opción para obtener acceso a dichas páginas HTML.

- Opción activada
  Active la casilla de verificación para que se inicie el servidor web del CP después de cargar los datos de configuración. Posteriormente, con un navegador web podrá leer toda la información del CP que este ponga a disposición en páginas web.
- Opción desactivada
  Si la casilla de verificación está desactivada, solo será posible acceder de forma limitada a la información que el CP pone a disposición en páginas web. La desactivación puede ser adecuada en determinados casos de servicio para liberar recursos adicionales en el CP.
  Las páginas siguientes no se muestran:
  - búfer de diagnóstico
  - información del módulo
  - topología

Actualización automática

Si la casilla de verificación está activada, las páginas web se actualizan automáticamente en el intervalo definido en "Intervalo de actualización".

La página web "Identificación" está excluida de la actualización automática.

Selección de los idiomas

Seleccione los idiomas para textos vinculados a idiomas que deben cargarse en el CP. El número de idiomas seleccionables depende del CP. Están vinculados a idiomas, p. ej., los textos del búfer de diagnóstico.

Tenga en cuenta además que los idiomas seleccionados deben estar instalados en el proyecto. La selección de idiomas del proyecto se encuentra en el árbol del proyecto, bajo "Idiomas y recursos".

Si previamente no se han instalado los idiomas seleccionados, el servidor web solo podrá visualizar los textos en su idioma estándar.

Estación PC CP PROFIBUS

Funcionamiento de CPs PROFIBUS en equipos con STEP 7 V5.5

Funcionamiento de CPs PROFIBUS en equipos con STEP 7 V5.5

Si en un equipo están instalados tanto STEP 7 V5.5 como STEP 7 Professional y se desinstala STEP 7 V5.5, será necesario ejecutar la función de reparación de STEP 7 Professional antes de poder utilizar los siguientes procesadores de comunicaciones como adaptadores de red al cargar firmware:

- CP 561x, CP 5611 A2, CP 5612, CP 5613 A2, CP 5614 A2
- CP 5621, CP 5622, CP 5623, CP 5624
- CP 5711

Carga de firmware en CPs PROFIBUS S7-300/400

Carga de firmware en CPs PROFIBUS S7-300/400 a través del CP 5612 o CP 5622

En caso de que falte la parametrización de interfaz necesaria para la carga de firmware ("FWL" o "FWL_FAST_LOAD"), proceda del siguiente modo:

1. Abra el programa "Ajustar interface PG/PC". Encontrará el programa en el Panel de control del sistema operativo Windows.
2. Seleccione la parametrización de interfaz en función del sistema operativo y cópiela:
   - En un sistema operativo de 32 bits:
     "CP5612(PROFIBUS)" / "CP5622(PROFIBUS)"
   - En un sistema operativo de 64 bits:
     "CP5612.PROFIBUS.1" / "CP5622.PROFIBUS.1"
3. Asigne uno de los nombres siguientes a la copia, según corresponda:
   - En un sistema operativo de 32 bits:
     CP5612(FWL) / CP5622(FWL)
     CP5612(FWL_FAST_LOAD) / CP5622(FWL_FAST_LOAD)
   - En un sistema operativo de 64 bits:
     "CP5612.FWL.1" / "CP5622.FWL.1"
     "CP5612.FWL_FAST_LOAD.1" / "CP5622.FWL_FAST_LOAD.1"
4. Abra las propiedades de la entrada y realice los siguiente ajustes en la ficha "PROFIBUS":
   - Active la casilla de verificación "La PG/el PC es el único maestro del bus".
   - Asigne a la dirección el valor "0".
   - Para la velocidad de transmisión asigne el valor 187,5 kbits/s (o bien para FWL_FAST_LOAD el valor 1,5 Mbits/s).
   - Modifique la "Dirección de estación más alta" al valor 126.
   - Modifique el perfil a "Universal (DP/FMS)".
5. Cierre el programa "Ajustar interface PG/PC".

Resultado:

La parametrización de interfaz configurada se crea y está disponible seguidamente en el actualizador de firmware.

Configuración LSAP

Referencia

Parametrización de la configuración LSAP en el PC CP PROFIBUS, en el grupo de parámetros "Propiedades > General > Configuración LSAP".

Reservar LSAPs

Reserve LSAPs (Local Service Access Point) para otros protocolos utilizados en el CP.

Conexiones FDL de ejemplo: El protocolo FDL utiliza LSAPs que no pueden ser utilizados simultáneamente por otros protocolos.

Seleccione para ello el número correspondiente al LSAP en la lista.

Configuración OPC (CP PROFIBUS)

Referencia

Parametrización de la configuración OPC en el PC CP PROFIBUS, en el grupo de parámetros "Propiedades > General > Configuración OPC".

El grupo de parámetros está presente cuando en el estación PC hay un servidor OPC configurado.

Opción "Acceso a través del servidor OPC"

Active el módulo para el funcionamiento a través de un servidor OPC configurado en el dispositivo.

Opción "Poner el maestro DP automáticamente en Operate"

Si se selecciona esta opción, el modo DP se activa automáticamente en el arranque del servidor OPC.

Control de actividad del CP activado - Opción "Vigilancia de tiempo DP"

Si el servidor OPC ya no accede (debido a un error) a la interfaz DP, con el control de actividad activado el módulo reacciona del siguiente modo: en todos los esclavos DP asignados al servidor OPC se envían datos con el valor 0 (valores sustitutivos).

Parámetros ajustables:

- Timeout del watchdog de DP
  Con el control de actividad activado, la transmisión de los valores sustitutivos se activa si tras el tiempo ajustado aquí no se produce ningún acceso a la interfaz DP.
  Predeterminado: 6000 ms

Opción "Vigilancia de diagnóstico" activada

Si la vigilancia de diagnóstico está activada, el servidor OPC vigila automáticamente el flag de diagnóstico en los accesos a entradas y salidas de los esclavos DP, y en el caso de los esclavos DP normalizados, evalúa los datos de diagnóstico. Si en los datos de diagnóstico se detectan errores o problemas, el acceso a las entradas y salidas de los esclavos DP se impide durante el tiempo de bloqueo especificado aquí, y las peticiones de lectura o escritura se acusan con error.

Parámetros ajustables:

- Tiempo de bloqueo para Read / Write
  Predeterminado: 500 ms

Tiempo de espera a error para peticiones DP C1

Especifique aquí un tiempo de espera a error para servicios DPC1 (transmisión acíclica como maestro DP de clase 1).

Predeterminado: 15000 ms

Funciones de servidor OPC para el maestro DP clase 2

Aquí se pueden activar las funciones de servidor OPC para el maestro DP clase 2. Los ajustes realizados aquí afectan a las funciones de registro según maestro DP clase 2.

Se ofrece una opción de 2 niveles:

- Activar funciones de registro
  Las funciones de registro permiten leer información específica del dispositivo o escribir parámetros. Con esta opción se activa el servidor OPC que soporta dicha función.
- Utilizar en modo PROFIdrive
  Al seleccionar adicionalmente esta opción se activa el servidor de bus PROFIdrive. De este modo, las funciones de registro solo pueden utilizarse en el modo especial PROFIdrive.

Parámetros ajustables:

- Timeout para el establecimiento de conexión
  Si el interlocutor no responde a la solicitud de establecimiento de una conexión dentro del tiempo ajustado aquí, las peticiones OPC se cancelan con un mensaje de error.
  Dado que dependiendo de la estructura de la red, el correcto establecimiento de la conexión puede requerir un tiempo mayor, el timeout para el establecimiento de conexión debe ser mayor que el de la petición (consulte el tiempo de espera a error para peticiones asíncronas).
  Predeterminado: 15000 ms
- Tiempo de espera a error para peticiones asíncronas
  Si una petición de lectura o escritura a un interlocutor no se responde dentro del tiempo ajustado aquí, las peticiones OPC pendientes se cancelan con un mensaje de error.
  Predeterminado: 15000 ms

Configuración del archivo LDB

Referencia

Parametrización de la configuración LDB en el PC CP PROFIBUS en el grupo de parámetros "Propiedades > General > Configuración LDB"

Nota

Versión del software SIMATIC NET PC

El archivo LDB solo puede utilizarse para CPs que utilicen una versión del software SIMATIC NET PC anterior a mayo de 2000.

Crear un archivo LDB (solo válido para CPs con firmware < V6.0 SP5 o hasta el software SIMATIC NET PC anterior a mayo de 2000)

La configuración del sistema maestro DP debe exportarse a un archivo LDB (base de datos cargable) al guardar y compilar. Este archivo contiene la información del proyecto DP. Se necesita para las conexiones de sistema DP de la PG/el PC así como SOFTNET o CP 5614 en las estaciones PC.

Nota

Archivo LDB y determinación de las direcciones de entrada/salida de los módulos DP

Las direcciones de las áreas de entrada/salida de los esclavos DP se determinan automáticamente, en orden ascendente y con justificación sobre la base de la dirección PROFIBUS de los esclavos DP y de los módulos DP insertados y se almacenan en el archivo LDB.

Por consiguiente, los datos o los cambios de direcciones de entrada/salida de los esclavos DP en el sistema maestro DP configurado aquí no repercuten en la información de dirección almacenada en el archivo LDB.

Configuraciones de PC conectados en red

Cuadro de diálogo de configuración

Referencia

Con la estación PC seleccionada, se elige el comando de menú "Online > Configurar estación PC online".

En el cuadro de diálogo abierto, se transfiere la configuración seleccionada para una estación PC o se visualiza su configuración actual.

Requisitos

En el equipo de destino y en el de configuración debe estar instalado el DVD "SIMATIC NET PC-Software".

Si en el equipo de configuración o de destino hay más de un módulo de comunicación, primero habrá que definir el módulo correspondiente para la operación de configuración. Para ello existe el programa "SIMATIC Shell".

Cuando ya lo haya definido en el equipo de configuración o si solo hay un módulo de comunicación, se visualizará la conexión de red local, que no podrá modificarse.

Si en el proyecto de STEP 7 se ha seleccionado el equipo de destino y existe una asignación unívoca con los equipos accesibles, los campos "Equipos de destino accesibles" y "Equipos de destino" dispondrán ya de la asignación correspondiente y no deberán modificarse.

Nota

SIMATIC Shell

El programa "SIMATIC Shell" se instala junto con los productos del software SIMATIC NET PC. Observe la documentación del software SIMATIC NET PC.

Proceda del siguiente modo:

1. En la lista desplegable seleccione la conexión de red con la que debe activarse el equipo de destino.
2. Pulse el botón "Actualizar" para determinar los equipos accesibles.
3. En la lista "Equipos accesibles" seleccione el equipo de destino que desee o introduzca el nombre del equipo de destino directamente en el campo "Equipo de destino". La denominación del equipo de destino es idéntica al nombre del equipo en Windows.
4. Elija entre los procedimientos siguientes:
   - Mostrar la configuración actual del equipo de destino seleccionado: pulse para ello el botón "Mostrar...".
   - Cargar la configuración en el equipo de destino: pulse para ello el botón "Configurar...".
5. Una vez realizada la configuración observe los posibles avisos.

Asignar conexión de red y equipo de destino

Si resulta que no desea utilizar las asignaciones predefinidas o que no hay ninguna, proceda del siguiente modo:

1. Dado el caso, desactive la opción "Utilizar equipo de destino configurado".
2. En la lista desplegable seleccione la conexión de red con la que debe activarse el equipo de destino.

3. Pulse el botón "Actualizar" para determinar los equipos accesibles.
4. Seleccione el equipo de destino deseado en la lista "Ordenadores accesibles". Otra posibilidad consiste en introducir el nombre del equipo de destino directamente en el campo "Equipo de destino". La denominación del equipo de destino es idéntica al nombre del equipo en Windows.

Configurar equipo de destino

Proceda del siguiente modo:

1. Elija entre los procedimientos siguientes:
   – Mostrar la configuración actual del equipo de destino seleccionado: pulse para ello el botón "Mostrar...".
   – Cargar la configuración en el equipo de destino: pulse para ello el botón "Configurar...".
2. Una vez realizada la configuración observe los avisos que pueden aparecer.

Sugerencia:

Las funciones descritas para los botones también son accesibles mediante el botón derecho del ratón.

Indicaciones generales

Tenga en cuenta que tras iniciar el cuadro de diálogo de configuración ya no se registran cambios en el equipo local. Siguen siendo válidos los ajustes o los datos de configuración que se detectaron al inicio. Las acciones afectadas son:

- La activación o desactivación de los adaptadores de red.
- Los cambios en la configuración de STEP 7 de la estación PC.

Cuadro de diálogo "Configuración del equipo de destino"

Este cuadro de diálogo muestra la configuración actual con módulos y aplicaciones.

Aquí existe la posibilidad de comparar una configuración actual con una predefinida.

Estación

Aquí se indica el nombre de la estación.

Dicho nombre puede ser idéntico al nombre del equipo de destino, a menos que se haya asignado un nombre nuevo a la estación de destino después de instalar el software SIMATIC NET PC. Al arrancar se aplica primero el nombre del equipo como nombre de la estación.

Consulte también

Símbolos de estado al configurar (Página 2074)

Cuadro de diálogo de configuración (Página 2070)

Cuadro de diálogo "Configurar: equipo de destino seleccionado"

Este cuadro de diálogo muestra la configuración actual para el equipo de destino seleccionado con módulos y aplicaciones.

Desde aquí se lanza la transferencia para la primera configuración o la modificación de la misma.

Al abrir el cuadro de diálogo se comprueba si es plausible transferir la configuración al equipo de destino y, por tanto, si puede ejecutarse.

Estación

Aquí se indica el nombre de la estación.

Dicho nombre puede ser idéntico al nombre del equipo de destino, siempre que todavía no se haya asignado un nombre nuevo a la estación de destino después del arranque. Al arrancar se aplica primero el nombre del equipo como nombre de la estación.

Comprobar y aplicar la configuración configurada

Aquí se verifica la configuración configurada antes de realizar la transferencia de datos definitiva.

En la línea de información, debajo de la lista de configuración, se indica si es posible realizar la configuración. Para que sea posible es imprescindible que la configuración real (física) y la predefinida coincidan.

Observe los símbolos que indican particularidades.

Después de haber comprobado los indicadores y de iniciado la aplicación (botón Aceptar), deberá volver a confirmarla.

Cuándo se rechaza la configuración

La configuración no puede transferirse en los casos siguientes:

- Los módulos configurados no existen en la estación PC.
- Los módulos configurados no pueden asignarse unívocamente (consulte las reglas de asignación más adelante).
- La configuración predefinida excede el alcance permitido para la estación PC.
- No es posible establecer una conexión con la estación PC.
  La causa puede ser un firewall configurado en la estación PC que bloquee el acceso al puerto 102 utilizado por STEP 7. Libere el puerto 102 para TCP en la estación PC.

Observe las indicaciones y los avisos de los campos "Error" y "Estado". Así, por ejemplo, se recibirán advertencias si la asignación puede provocar un cambio de la configuración actual (ejemplo: un módulo ya configurado se desplazará a otro índice).

Haciendo doble clic en el texto de aviso se abre una ventana con información detallada acerca del aviso en cuestión.

Reglas de asignación

Al importar se comprueba si es posible una asignación de los módulos y, con ello, una aplicación, del siguiente modo:

- Los módulos predefinidos tienen que estar presentes.
- Si se encuentran varios módulos (por ejemplo CPs PROFIBUS), se intentará realizar la asignación:
  - sobre la base del string de parametrización asignado al módulo (en STEP 7)
  - sobre la base de la dirección (PC o MAC/IP)
  - sobre la base de la compatibilidad (volumen de funciones)

Nota

CPs Softnet

Tenga en cuenta que en cada configuración sólo puede haber un CP Softnet (PROFIBUS o Industrial Ethernet).

Consulte también

Símbolos de estado al configurar (Página 2074)

Cuadro de diálogo de configuración (Página 2070)

Símbolos de estado al configurar

Columna "Estado"

Observe la información adicional de la columna "Causas" en cada caso.

Error: componente no asignable / aplicable / no instalado

Advertencia: la asignación se ha adaptado o no es unívoca.

Línea de información debajo de la lista de configuración

La línea de información notifica un aviso de grupo que informa de la operación de importación:

La configuración no puede ejecutarse

La configuración puede ejecutarse pero hay advertencias: la asignación de los diferentes componentes se ha adaptado o no es unívoca.

La configuración puede ejecutarse

Consulte también

Cuadro de diálogo "Configurar: equipo de destino seleccionado" (Página 2073)

Estación PC, ajustes independientes de protocolo o red en un PC CP

Configuración XDB

Referencia

Parametrización de la configuración XDB en una estación PC en el grupo de parámetros "Propiedades > Configuración XDB"

Configuración XDB

- Compatibilidad - La opción "S7RTM" está instalada
  Si se configura una estación PC, la casilla de verificación está activada. En ese caso, en la estación PC de destino es necesario un "Administrador de estaciones en runtime" (S7RTM). La estación PC de destino debe estar configurada con el configurador de componentes (software SIMATIC NET PC a partir de aprox. 7/2001).
  Si la casilla de verificación está desactivada se generan archivos de configuración interpretables por estaciones PC "más antiguas".
- Ubicación del archivo de configuración
  La configuración con las descripciones de conexión para la estación PC SIMATIC se guarda en un archivo de configuración (*.XDB) al guardar y compilar. Ese archivo debe instalarse en la estación PC correspondiente.
  Con el botón "Examinar" se puede modificar la ruta predeterminada para el archivo de configuración.
  La indicación de ruta relativa (.\...) se refiere al directorio en el que se guarda el proyecto actual (<Install-Dir>\S7proj\<Projekt>\XDB). Si introduce una ruta absoluta (p. ej. C:\...), recuerde que el acceso solo funciona mientras el proyecto permanezca en el entorno actual.

Asignación de la parametrización de la interfaz

Comprobación de la parametrización de la interfaz

Al tomar la tarjeta de comunicación PC del catálogo de hardware y colocarla en el rack se asigna una parametrización de interfaz adecuada.

La parametrización válida se detecta a partir de la información del tipo de módulo existente en el catálogo (nombre del módulo y número de módulos configurables del mismo tipo) así como de los módulos ya configurados (asignación del número de tarjeta).

Si se configuran varios CPs del mismo tipo en la estación, las denominaciones de los módulos se complementarán con un número correlativo adicional (el número de tarjeta). Por medio de la lista desplegable que aparece pueden detectarse y seleccionarse las entradas admisibles.

En casos especiales, p. ej. cuando se intercambia la asignación de módulos idénticos en la estación PC, puede ser necesario adaptar la asignación de la interfaz; en este caso nos referimos a cambiar el hardware, no el slot "virtual" en HW Config.

Nota

Parametrización de la interfaz en el sistema de destino (estación PC)

Asegúrese de que el sistema de destino (estación PC) utilice la misma parametrización de interfaz que la indicada aquí. De lo contrario se producirá un conflicto de interfaces después de cargar el archivo de configuración.

Nota

Comprobación en el sistema de destino

Para la comprobación en el sistema de destino (estación PC) utilice la función Ajustar interfaz PG/PC. La encontrará en el Panel de control de Windows. En la ficha "Vía de acceso" verá qué interfaces están configuradas y si existen conflictos.

IE/PB Link PN IO

Números de dispositivo PROFINET

Referencia

Indicación y parametrización de las propiedades para dispositivos PROFINET-IO en el sistema maestro de un IE/PB Link en el grupo de parámetros "Propiedades > General > Número de dispositivo PROFINET".

Significado y parametrización

Para conseguir una correspondencia sencilla, al colocar el esclavo DP en el sistema maestro DP STEP 7 intenta asignar las mismas cifras para el número de dispositivo y la dirección PROFIBUS. Por ello, por defecto está activada la opción "Utilizar la dirección PROFIBUS como número de dispositivo", siempre que aún no se haya asignado ningún número de dispositivo que se corresponda con la dirección PROFIBUS actual.

- Opción "Arranque priorizado"
  Para conseguir que el controlador IO procese con preferencia el dispositivo PROFINET IO (aquí el esclavo DP como dispositivo PROFINET IO) y reduzca así el tiempo de arranque, active la casilla de verificación.
  Esta opción se puede seleccionar para hasta 8 esclavos DP conectados al sistema maestro DP.
  La opción solo puede seleccionarse si el controlador PROFINET IO soporta el arranque priorizado y siempre que aún no se haya alcanzado el número máximo citado anteriormente.
- Opción "Número de dispositivo = dirección PROFIBUS"
  Si se selecciona esta opción, el campo de entrada para el número de dispositivo PROFINET es editable (se visualiza la lista desplegable).
- Número de dispositivo PROFINET
  Rango de valores: Depende del controlador IO utilizado; si se supera el rango permitido al final del diálogo aparece un aviso indicando el rango permitido.
  Si tras introducir un valor se vuelve a seleccionar la opción "Utilizar la dirección PROFIBUS como número de dispositivo", el valor se restaura de la forma correspondiente (no puede verse hasta cerrar el cuadro de diálogo).

Nota

Números de dispositivo unívocos

Recuerde que los números de dispositivo asignados en el sistema PROFINET IO deben ser unívocos. Deben tenerse en cuenta también los dispositivos IO (esclavos DP) conectados al sistema maestro DP del IE/PB Link.

Modos de operación del IE/PB Link PN IO

Referencia

Configuración del modo de operación de la transición de red y la indicación del modo de operación resultante para PROFIBUS DP.

- Ajuste del modo de operación de la transición de red IE/PB Link PN IO en el grupo de parámetros "Propiedades > General > Modo de operación"
- Indicación del modo de operación para PROFIBUS DP en el grupo de parámetros "Propiedades > General > Interfaz DP/MPI > Modo de operación".

Significado y parametrización

La transición de red IE/PB Link PN IO se puede utilizar en los siguientes modos de operación:

- Transición de red como proxy de PROFINET IO
  Desde el punto de vista del controlador PROFINET IO conectado a Industrial Ethernet no hay diferencia en el acceso a dispositivos PROFINET IO conectados a Industrial Ethernet y a esclavos PROFIBUS DP conectados a PROFIBUS DP.
  En este caso, el IE/PB Link PN IO adopta el papel de proxy (modo de operación DP "Maestro DP" activo) para los esclavos DP conectados a PROFIBUS DP.
- Transición de red en modo estándar
  En este modo de operación existen las siguientes posibilidades de uso
  - Comunicación PG/OP
    La comunicación PG/OP permite cargar programas y datos de configuración para realizar funciones de test y diagnóstico, así como para manejar y observar (sistemas HMI) una instalación.
  - Parametrización de aparatos de campo (routing de registros)
    Se puede utilizar el IE/PB Link PN IO como router para registros destinados a aparatos de campo (esclavos DP). Así, los dispositivos que no están conectados directamente a PROFIBUS, y que, por lo tanto, carecen de acceso directo a los aparatos de campo (esclavos DP), puede transmitir registros a estos a través de IE/PB Link PN IO.
    Una herramienta que genera este tipo de registros para la parametrización de aparatos de campo es, por ejemplo, SIMATIC PDM (Process Device Manager).
  - Transición de red a un sistema maestro DP con equidistancia
    El IE/PB Link PN IO ejerce de transición de red entre Ethernet y los aparatos de campo de un sistema maestro DP. El IE/PB Link PN IO funciona como dispositivo activo junto con un maestro DP de un PROFIBUS parametrizado como equidistante.
  - Conexiones S7 comunes a varias subredes para modo HMI
    El IE/PB Link PN IO transmite la comunicación a través de conexiones S7.
    Este servicio se utiliza, por ejemplo, en aplicaciones HMI (estaciones PC).

Consulte también

Configuración con IE/PB-Link PN IO (Página 1489)

Aclaración sobre conexiones

Window Time e Inactivity Time - Relación

Referencia

Configuración de conexiones ISO-Transport / grupo de parámetros "Propiedades > General > Dinámica".

Impedir la cancelación de conexiones mediante una sincronización adecuada

La estación interlocutora responde a los telegramas de señal de vida con otro telegrama. Por ello se envían a la estación interlocutora en el intervalo del Window Time.

Para que no se cancele la conexión involuntariamente, el Inactivity Time debe ser al menos tres veces mayor que el Window Time.

Opción "Notificación inmediata al detectar una interrupción de la conexión"

Significado

Si se selecciona la opción "Notificación inmediata al detectar una interrupción de la conexión" se determina que, en caso de que una conexión no esté disponible, la aplicación reciba inmediatamente una notificación acerca de que una petición de comunicación no puede ejecutarse (FETCH o WRITE). La notificación se da independientemente de si funciona o no un intento actual de restablecer la conexión.

Si no se selecciona esta opción, en caso de que una conexión no esté disponible, la aplicación no recibirá la notificación hasta que haya transcurrido el timeout para la conexión.

La opción aquí descrita sólo es relevante y sólo está disponible si se utilizan los dos modos de operación citados a continuación:

- La conexión está configurada como permanente.
  Las conexiones pueden estar configuradas como permanentes (opción de la ficha "Detalles del establecimiento de conexión"). De este modo, las peticiones pueden transmitirse al interlocutor con un tiempo de reacción lo más breve posible sin que sea necesario establecer primero una conexión.
  Si la conexión falla, se intentará volver a establecerla. En este estado pueden producirse retardos hasta que el aviso correspondiente se transmita a la aplicación.
- Se utilizan los tipos de petición FETCH o WRITE (posibilidad de selección en la ficha "Opciones").

Sólo con estos tipos de petición es necesaria una reacción inmediata en determinados casos de aplicación si falla la conexión.

Datos de dirección - conexión de correo electrónico

Parámetros para el envío de correos electrónicos

En "Propiedades > Detalles de dirección" se indican los parámetros que se necesitan para enviar un correo electrónico. Pueden introducirse los parámetros siguientes:

Parámetro	Descripción	Ejemplo
Servidor de correo electrónico (SMTP)	Dirección del servidor de correo a través del que se envían los correos electrónicos. La dirección IP puede indicarse en formato absoluto o simbólico. Para la entrada simbólica rige lo siguiente: Un nombre válido tiene entre 1 y 64 caracteres; de ellos, 1 carácter como mínimo debe ser una letra. Si en la entrada se utilizan uno o varios caracteres ".", rige además: Un nombre válido debe tener detrás del último "." entre 1 y 63 caracteres, que solo pueden ser [a-z], [A-Z], [0-9] o [-] y de los cuales al menos uno debe ser una letra. Para el formato simbólico es imprescindible que el CP de Internet conozca la dirección del Domain Name System (DNS). Debe realizarse una entrada a este respecto en la configuración del CP de Internet en HW Config; encontrará más información al respecto en la Ayuda en pantalla correspondiente. Pueden introducirse 64 caracteres como máximo.	absoluto:140.80.0.4 simbólico:t-online.de
Nombre predeterminado del remitente	Indicación de una dirección que se insertará como dirección del remitente en el correo electrónico siempre que la información del remitente (parámetro FROM) esté vacía en el encabezado del correo electrónico. Este dato no suele ser relevante para el envío de mensajes. Sirve de información para el receptor del correo electrónico. Puesto que algunos servidores de correo no transmiten peticiones si el campo del remitente está vacío, **es recomendable realizar una entrada.** Pueden introducirse 126 caracteres como máximo. **Nota:** Tenga en cuenta que hay que indicar un nombre predeterminado del remitente si debe iniciarse un mensaje de test desde el diagnóstico.	Station2.CPU412@xy.werk2.de

Símbolos en estado offline

Símbolos en estado offline

El estado offline de una conexión se representa del siguiente modo:

Símbolo	Significado / estado de la conexión
	Coherente - la configuración de conexiones se puede cargar
	Incoherente - la configuración de conexiones no se puede cargar

Conexión ISO on TCP: TSAP

Formato TSAP

Las conexiones ISO on TCP tienen una longitud de TSAP de entre 1 y 16 bytes. Al realizar la entrada se muestra automáticamente la longitud actual. Los TSAPs locales y del interlocutor pueden introducirse como valor hexadecimal o como string ASCII. En caso de una entrada ASCII, los caracteres introducidos también se visualizan en formato hexadecimal. En caso de entrada hexadecimal, los caracteres imprimibles se representan como valor ASCII (son visibles 8 caracteres hexadecimales). Si se introducen caracteres no imprimibles, la visualización ASCII queda atenuada (no es posible la entrada ASCII) y los caracteres no imprimibles se representan en forma de punto.

Nota

Longitud TSAP

No deben utilizarse TSAPs con una longitud 1 ó 2, porque ya se utilizan para fines internos.

TSAPs locales y del interlocutor

Los TSAPs locales y del interlocutor pueden ser idénticos, ya que la conexión es unívoca debido a las distintas direcciones IP. Si hay que configurar más de una conexión entre dos dispositivos, habrá que diferenciar también los TSAPs.

TSAPs predeterminados

Para configurar los TSAPs locales y del interlocutor hay un valor predeterminado "ISO-on-TCP-1" para la primera conexión entre ambos interlocutores (puede cambiarse). Si se establece una nueva conexión entre los mismos interlocutores, se propone el valor predeterminado "ISO-on-TCP-2". Si se establece una nueva conexión con otro interlocutor distinto, se utiliza de nuevo "ISO-on-TCP-1".

Establecimiento de conexión para FETCH/WRITE

Referencia

Configuración de conexiones ISO-Transport / conexiones ISO-on-TCP / conexiones TCP - grupo de parámetros "Propiedades > General > Opciones".

Repercusiones del establecimiento de conexión en FETCH/WRITE

Para el modo de operación FETCH/WRITE tenga en cuenta las indicaciones siguientes:

- Para estaciones S7 rige lo siguiente:
  Si debe utilizarse el modo de operación FETCH/WRITE (véase el grupo de parámetros "Opciones"), hay que desactivar la opción "Establecimiento de conexión activo".
- Para estaciones PC / servidores OPC rige lo siguiente:
  Si debe utilizarse el modo de operación FETCH/WRITE (véase el grupo de parámetros "Opciones"), hay que activar la opción "Establecimiento de conexión activo".

Conexión S7

Parámetros de conexión OPC para vigilancia de tiempo

Timeout para establecimiento de conexión

Si una petición de establecer una conexión no recibe respuesta por parte del interlocutor dentro del tiempo ajustado aquí, las peticiones OPC se cancelan con un mensaje de error. Se admiten valores entre 2.000 y 100.000 ms.

Puesto que, según sea la estructura de la red, es posible que se tarde bastante tiempo en establecer una conexión correctamente, el timeout para el establecimiento de la conexión debería ser mayor que el timeout para peticiones.

El timeout para el establecimiento de la conexión comienza cuando se realiza la petición de establecer una conexión. Este instante depende de la configuración del comportamiento del establecimiento de la conexión (véase más arriba). Hay dos casos posibles:

- **Mantener la conexión de forma permanente:**la petición de establecer una conexión comienza en este caso con el arranque del servidor OPC para el protocolo S7. El arranque se inicia conectando un cliente OPC al servidor OPC y examinando el espacio del nombre S7 y/o agregando y observando activamente un S7 Item.
- **Conexión en caso necesario:**la petición de establecer una conexión comienza con el primer acceso a las variables de esta conexión S7.

Si la conexión S7 se interrumpe durante el funcionamiento, se intenta restablecerla. El timeout para el establecimiento de la conexión se resetea y se inicia de nuevo.

Timeout para peticiones

Con el timeout para peticiones se vigilan las peticiones de red dirigidas al interlocutor. Si no se responde a una petición de red dentro del timeout para peticiones, las correspondientes peticiones OPC se cancelan con un mensaje de error. Se admiten valores entre 1 y 100.000 ms.

El timeout para peticiones comienza cuando se envía una petición de red en el servidor OPC. Normalmente solo se establece una relación indirecta con peticiones de escritura/lectura/observación en la interfaz OPC. Se pueden procesar varios OPC Items por petición de red y, además, en cada petición de red varios clientes OPC pueden encargar los mismos OPC Items.

También hay que tener en cuenta los parámetros "Optimizar accesos de escritura" y "Optimizar accesos de lectura", porque permiten agrupar varias peticiones OPC en el nivel de la red.

Parámetros de conexión OPC para estado de la conexión

Notificación inmediata al detectar una interrupción de la conexión

Sólo es activable si la opción "Mantener la conexión de forma permanente" está seleccionada. Gracias a la notificación inmediata de las conexiones S7 no establecidas en peticiones OPC se impide que se repitan los timeouts para peticiones.

Funcionamiento: se detecta una interrupción de la conexión y se guarda el estado de la misma. Si la conexión todavía no está establecida o si se intenta volver a establecer una conexión interrumpida, las peticiones OPC se responden inmediatamente con una indicación de error correspondiente. La indicación de error no llega hasta haber transcurrido el tiempo de espera del timeout para peticiones o conexiones.

Optimizar accesos de escritura, optimizar accesos de lectura

Principio

Para accesos a variables a través de conexiones S7, el servidor OPC ofrece un algoritmo de optimización para SIMATIC NET, que permite aumentar la velocidad de la transferencia de datos.

Varias peticiones de acceso a diferentes variables se convierten internamente en un único acceso al interlocutor. Así se reduce el número de paquetes de datos transportados por la red. La carga de los paquetes se mejora y la proporción de datos de usuario de un paquete aumenta.

Esta optimización sirve tanto para accesos de lectura como de escritura y está activada por defecto. Para el cliente OPC, p. ej. un sistema de manejo o visualización apto para OPC, este algoritmo de optimización es invisible.

Informaciones adicionales

Encontrará más detalles en la documentación del servidor OPC:

- Comunicación industrial con PG/PC tomo 1, ID de artículo: 42783968 (http://support.automation.siemens.com/WW/view/es/42783968)
- Comunicación industrial con PG/PC tomo 2, ID de artículo: 42783660 (http://support.automation.siemens.com/WW/view/es/42783660)

Opción de conexión S7 unilateral

Significado

Si la casilla de verificación "Unilateral" está activada, para utilizar esta conexión sólo se admitirán bloques de comunicación unilaterales en el programa de usuario de una CPU S7 (instrucciones "PUT" y "GET"). Unilateral significa que el interlocutor es el servidor de dicha conexión y que no puede enviar o recibir activamente.

En las conexiones unilaterales no se pueden emplear bloques de comunicación bilaterales ("parejas de bloques"), como "USEND (Página 7683)" y "URCV (Página 7685)", "BSEND (Página 7688)" y "BRCV (Página 7691)" o "USTATUS (Página 7722)".

Puesto que, por norma general, las posibilidades de comunicación se identifican claramente por los puntos finales, esta casilla de verificación está preasignada y, dado el caso, no puede cambiarse.

Ejemplo 1:

La casilla de verificación está atenuada (es decir, no se puede cambiar) y activada con la configuración siguiente:

- El punto final local de una conexión es una CPU S7-400 o una CPU S7-300 con comunicación S7 cargable (mediante CP) y el punto final del interlocutor es una CPU S7-300 sin comunicación S7 cargable.

Ejemplo 2:

La casilla de verificación está atenuada (es decir, no se puede cambiar) y no está activada con la configuración siguiente:

- El punto final local de una conexión y el punto final del interlocutor son CPUs S7-400 En este caso, STEP 7 crea automáticamente una conexión bilateral. Se introduce una ID tanto para el punto final local como para el punto final del interlocutor (ID local e ID del interlocutor).

Consulte también

GET: Leer datos de una CPU remota (Página 5249)

PUT: Escribir datos en una CPU remota (Página 5252)

GET: Leer datos de una CPU remota (Página 7677)

PUT: Escribir datos en una CPU remota (Página 7680)

Servidor OPC - avisos

Avisos de bloque y avisos de diagnóstico

Las opciones sirven de filtro, de modo que determinan selectivamente el tipo de avisos que pueden entrar a través de las conexiones S7.

Los avisos de bloque (alarmas) son generados por el interlocutor mediante los bloques Alarm_8, Alarm_S o Alarm_SQ.

Los avisos de bloque (alarmas) pueden notificarse

- como Conditional Events (eventos condicionados)
- como Simple Events (eventos simples para la compatibilidad con aplicaciones antiguas que no suelen esperar Conditional Events).

Si se selecciona "Avisos de bloque", el servidor OPC deberá iniciar una sesión en esta conexión para las alarmas. Las alarmas recibidas se transmiten por la interfaz OPC Alarms&Events.

Los avisos de diagnóstico se determinan a partir de la lista de estado del sistema (SZL) del dispositivo S7. Los eventos de diagnóstico que todavía no se han notificado al cliente OPC se le notifican cuando inicia una sesión para avisos de diagnóstico. Los eventos de diagnóstico no tienen estado y no hay que acusarlos.

A través de la navegación local se accede a las funciones para ajustar las prioridades de avisos de alarma individuales y avisos de diagnóstico.

Informaciones adicionales

Encontrará más detalles en la documentación del servidor OPC:

- Comunicación industrial con PG/PC tomo 1, ID de artículo: 42783968 (http://support.automation.siemens.com/WW/view/es/42783968)
- Comunicación industrial con PG/PC tomo 2, ID de artículo: 42783660 (http://support.automation.siemens.com/WW/view/es/42783660)

AS-i

Configuración E/S en esclavos AS-i

Configuración E/S

La configuración E/S está asignada a los códigos E/S según la norma, del siguiente modo:

Tabla 8-97 Esclavo AS-i estándar

Código E/S	Configuración E/S			
	D0	D1	D2	D3
0	E	E	E	E
1	E	E	E	S
2	E	E	E	B
3	E	E	S	S
4	E	E	B	B
5	E	S	S	S
6	B	B	B	B
7	B	B	B	B
8	S	S	S	S
9	S	S	S	E
A	S	S	S	B
B	S	S	E	E
C	S	S	B	B
D	S	E	E	E
E	S	B	B	B
F	F	F	F	F

Leyenda: E=entrada; S=salida; B=bidireccional (entrada/salida); F=TriState

Tabla 8-98 Esclavo A/B AS-i

Código E/S	Configuración E/S			
	D0	D1	D2	D3
0	E	E	E	E
1	E	E	E	-
2	E	E	E	E
3	E	E	S	-
4	E	E	B	E
5	E	S	S	-
6	B	B	B	E
7	B	B	B	E
8	S	S	S	-

Código E/S	Configuración E/S			
	D0	D1	D2	D3
9	S	S	S	E
A	S	S	S	E
B	S	S	E	E
C	S	S	B	E
D	S	E	E	E
E	S	B	B	E
F	F	F	F	F

Leyenda: E=entrada; S=salida; B=bidireccional (entrada/salida); F=TriState

ASM 475

Información importante sobre ASM 475

Introducción

En un rack del SIMATIC S7-300 es posible insertar y operar un máximo de ocho módulos de interfaz ASM 475. En una configuración con varios racks (máximo cuatro), el ASM 475 puede insertarse y operarse en cada uno de ellos. De este modo, la configuración máxima de un SIMATIC S7-300 permite utilizar 32 ASM 475.

Al ASM 475 se le pueden conectar dos SLG (dispositivos de escritura y lectura) como máximo. Los SLG conectados se procesan en paralelo. La FC 45 hace posible una programación sencilla utilizando las herramientas de SIMATIC S7.

La FC 45 puede utilizarse tanto en el S7-300 como en el S7-400.

En el S7-400, el ASM 475 se conecta mediante un ET 200M.

Tenga en cuenta que

- el IM 153-1 debe tener como mínimo el n.º MLFB 6ES7 153-1AA03-0XB0 o 6ES7 153-1AA83-0XB0 y
- el IM 153-2 como mínimo el n.º MLFB 6ES7 153-2AA02-0XB0 o 6ES7 153-2AB01-0XB0

El acceso a los datos MDS se realiza mediante direccionamiento normal.

Direccionamiento físico de la MDS (memoria de datos móvil)

El direccionamiento físico de una MDS también se denomina **direccionamiento normal**. El usuario crea por sí mismo la estructura de una MDS. Sabe qué datos están escritos en qué dirección MDS física. El direccionamiento de la memoria MDS comienza normalmente con la dirección 0000 hex y termina en una dirección final adecuada al tamaño de la MDS.

Direccionamiento Filehandler de la MDS

En el direccionamiento Filehandler, el usuario indica un nombre de archivo para acceder a sus datos. Dicho nombre consta de ocho caracteres ASCII. El Filehandler administra independientemente los datos de usuario en la MDS. El usuario no tiene que crear ninguna estructura de datos en la MDS.

La MDS debe formatearse antes de usarse.

8.1.5.5 Sistemas de identificación

Sistemas RFID

Módulo de comunicación RF120C

Grupo de parámetros Lector

El grupo de parámetros "Lector" incluye lo siguiente:

- Diagnose: aquí se ajusta si deben emitirse avisos de diagnóstico de hardware.
- Sistema de lector: selección del sistema RFID conectado. En función de la selección, el grupo de parámetros "Sistema de lector" se adapta en correspondencia.

Parámetro "Diagnóstico"

Opciones de parametrización:

- Ninguno
  No se generan otros avisos de diagnóstico que los avisos de diagnóstico estándar.
- Hard Errors
  Con los siguientes eventos se generan avisos de diagnóstico avanzados.
  - Error de hardware (test de memoria)
  - Error de firmware (suma de verificación)
  - Interrupción de la conexión con el lector
  - Error de cortocircuito o interrupción, si está soportado por el hardware

Informaciones adicionales

Para obtener más información sobre el diagnóstico consulte la documentación del módulo de comunicación RF120C con bloques de aplicación para S7-1200 y S7-1500.

Grupo de parámetros Sistema de lector

En el grupo de parámetros "Sistema de lector" se ajustan los parámetros para el sistema de identificación seleccionado.

La tabla muestra los parámetros existentes en todos los sistemas de identificación (sistemas RFID y de lectura de códigos).

Tabla 8-99 Parámetros estándar para todos los sistemas de identificación

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
Velocidad de transferencia del lector	19,2 kBaud 57,6 kBaud 115,2 kBaud	115,2 kBaud	Tras cambiar la velocidad de transferencia es necesario desconectar y volver a conectar el lector.
Anwesenheitskontrolle	On Off (RF field off) Off (RF field on)	On	On = se notifica la presencia en cuanto hay un transpondedor en el campo de antena del lector Off (RF field on) = se suprime la indicación de presencia en el FB De todas formas, la antena del lector sigue conectada mientras no se desconecte por medio de un comando. Off (RF field off) = la antena solo se conecta cuando se envía un comando y después vuelve a desconectarse (solo para RF300)
Reset ERR LED	On Off	Off	On = cada reset del FB provoca el reset del parpadeo del LED de error en el RF120C. Off = el LED de error indica siempre el último error. Solo es posible resetear la indicación desconectando el RF120C.

Los parámetros siguientes son específicos del sistema, según lo que se haya seleccionado en el grupo de parámetros "Lector".

RF200

Tabla 8-100 RF200 allgemein

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
-	-	-	Solo están disponibles los parámetros estándar.

Tabla 8-101 RF290R

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
RF power	0,50 - 5,00 W	1,00 W	Ajuste de la potencia de salida del lector.

RF300

Tabla 8-102 RF300 allgemein

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
Tipo de transpondedor	RF300 ISO 15693	RF300	Selección de los transpondedores utilizados.

Tabla 8-103 RF380R

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
RF power	0,50 - 2,00 W	1,25 W	Ajuste de la potencia de salida del lector.
Tipo de transpondedor	RF300 ISO 15693	RF300	Selección de los transpondedores utilizados.

RF600

Tabla 8-104 Sistema de lector: RF600

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
Max. no. of transponders	1 - 80	1	Número de transpondedores esperados en el campo de antena. Valores admitidos: • 1 - 40 para RF620R • 1 - 80 para RF630R con 2 antenas • 1 - 40 para RF630R con 1 antena El valor que se almacena en "multitag" define el número previsto de transpondedores que deben leerse (EPC-ID) en el inventario. Dicho valor no limita el número de transpondedores que deben procesarse en el campo de antena. Para permitir un inventario productivo de transpondedores en el campo de antena, asegúrese de que los valores indicados aquí para el número máximo de transpondedores previstos en el campo de antena no difieren en más del 10 % aproximadamente.
Reader type	RF620R ETSI RF620R FCC RF620R CMIIT RF630R ETSI RF630R FCC RF630R CMIIT	RF620R ETSI	Selección del lector utilizado. Cuando se selecciona un lector se abre el grupo de parámetros "Tipo de lector". Los parámetros están descritos en la tabla siguiente.

Tipo de lector (solo para RF600)

En el grupo de parámetros "Tipo de lector" se ajustan parámetros específicos adicionales para el tipo de lector seleccionado en el grupo de parámetros "Sistema de lector".

Tabla 8-105 Menú del RF600: Reader type

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
radio profile ETSI	-	-	Selección del respectivo perfil de radiofrecuencia para ETSI, FCC o CMIT.
Multitag mode	UID = EPC-ID (8 Byte) UID = Handle-ID (4 Byte)	UID = EPC-ID (8 Byte)	• UID = EPC-ID (8 bytes) UID de 8 bytes de los bytes 5-12 de la EPC-ID de 12 bytes de longitud • UID = Handle-ID (4 bytes) UID de 4 bytes como Handle-ID para accesos a transpondedores con una EPC-ID de cualquier longitud
Intelligent Singletag Mode (ISTM)	On Off	Off	Activación o desactivación del algoritmo "Intelligent Singletag Mode ISTM" [1]
Black List	On Off	Off	Activación o desactivación de la "Blacklist" [1]
Radiated power internal antenna (RF620R)	0 - B	4	Ajuste del flujo óptico para la antena interna [1] [2]
Antena interna (RF620R)	-	-	Activar/desactivar la antena interna. En el RF620R solo puede ajustarse una antena, la interna o la externa.
Potencia de emisión (antena externa RF620R)	0 - 9	4	Ajuste de la potencia de emisión para la antena externa [1] [2]
Antena externa (se requiere SetAnt) (RF620R)	-	-	Activar/desactivar la antena externa. En el RF620R solo puede ajustarse una antena, la interna o la externa.
Potencia de emisión ANT 1 (RF630R)	0 - 9	4	Ajuste de la potencia de emisión para la antena 1[1] [2]
Potencia de emisión ANT 2 (RF630R)	0 - 9	4	Ajuste de la potencia de emisión para la antena 2[1] [2]
Velocidad de comunicación	Reliable detection Fast detection	Reliable detection	[1]
Tag Hold	On Off	Off	Activación o desactivación del "Tag Hold" [1]
Scanning Mode	On Off	Off	Activación o desactivación del "Scanning Mode" [1]
Channel assignment (only with wireless profile ETSI)	-	-	Selección de los canales de radiofrecuencia que deben utilizarse [1]

[1] Encontrará más información en el "RF620R/RF630R Configuration Manual". *

[2] Encontrará los valores de la potencia de emisión y el flujo óptico de las antenas en la tabla siguiente.

* Encontrará el RF620R/RF630R Configuration Manual aquí (http://support.automation.siemens.com/WW/view/es/10805817).

Tabla 8-106 Potencia de emisión y flujo óptico de las antenas

Valor Hex	RF630R Potencia de emisión	RF620R Flujo óptico (antena interna)			RF620R Potencia de emisión
	dBm / (mW)	ETSI dBm / (mW) ERP	FCC dBm / (mW) EIRP	CMIIT dBm / (mW) ERP	dBm / (mW)
0	18 / (63)	18 / (63)	20 / (100)	18 / (65)	18 / (63)
1	19 / (79)	19 / (79)	21 / (126)	19 / (79)	19 / (79)
...	...	...	...	...	...
4	22 / (158)	22 / (158)	24 / (251)	22 / (158)	22 / (158)
...	...	...	...	...	...
9	27 / (501)	27 / (501)	29 / (794)	27 / (501)	27 / (501)
A	27 / (501)	28 / (631)	30 / (1000)	28 / (631)	27 / (501)
B (...F)	27 / (501)	29 / (794)	31 / (1259)	29 / (794)	27 / (501)

SLG D10S

Tabla 8-107 SLG D10S

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
RF power	0,50 - 10,00 W	1,00 W	Ajuste de la potencia de salida del lector.
Transponder type	ISO I-Code (p. ej. MDS D139)	ISO	Selección de los transpondedores utilizados.

SLG D11S/D12S

Tabla 8-108 SLG D11S/D12S

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
Transponder type	ISO I-Code (p. ej. MDS D139)	ISO	Selección de los transpondedores utilizados.

MOBY U

Tabla 8-109 MOBY U

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
Standby time	0 - 1400 ms	0 ms	Tiempo en standby (scanning_time) para el transpondedor. Si el transpondedor recibe otro comando antes de que finalice el tiempo en standby, este puede procesarse inmediatamente. Si el transpondedor recibe un comando una vez finalizado el tiempo en standby, su procesamiento se retarda según el "sleep_time" del transpondedor.
Range limitation	0,2 m 0,5 m 1,0 m 1,5 m 2,0 m 2,5 m 3,0 m 3,5 m	1,5 m	-
Max. no. of transponders	1 - 12	1	Número máximo de transpondedores que pueden procesarse en el campo de antena paralelamente.

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
BERO mode	Sin BEROs 1 o 2 BEROs 1.er BERO activado, 2.º BERO desactivado Sincronización por conexión de cable	Sin BEROs	• Sin BEROs Los lectores no se sincronizan • 1 o 2 BEROs Los BEROs tienen una combinación lógica O entre sí. Mientras dure la activación de un BERO el campo de antena estará conectado. • 1.er BERO activado, 2.º BERO desactivado El primer BERO activa el campo de antena y el segundo BERO lo desactiva. Si hay dos BEROs y está parametrizado "Tiempo de BERO en s", el campo de antena se desactivará automáticamente si el segundo BERO no conmuta dentro de este tiempo de BERO. Si no está parametrizado "BERO time in s", el campo de antena permanece activado hasta que se activa el segundo BERO. • Sincronización por conexión de cable Activar la sincronización del lector mediante una conexión por cable (consulte el manual de configuración, montaje y servicio técnico de MOBY U).
BERO time in s	0 - 255 s	0 s	Solo puede ajustarse si el modo de operación BERO está ajustado a "1st BERO on, 2nd BERO off". • 0 La vigilancia de tiempo está desactivada. Para la desconexión del campo se requiere el segundo BERO. • 1 ... 255 s Tiempo de conexión para el campo del lector

Allgem. Reader

Tabla 8-110 Allgem. Reader

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
Campo de entrada para la serie de bytes (hexadecimal)	00000000	00000000	Modo experto Esta función permite indicar los parámetros de reset directamente en representación hexadecimal. Solo es posible seleccionar este ajuste si previamente un empleado de SIEMENS le ha facilitado la cadena de caracteres hexadecimales correspondiente.

Parámetros vía FB / Lectores de códigos

Tabla 8-111 Parámetros vía FB / Lectores de códigos

Parámetro	Valor de parámetro	Valor predeterminado	Descripción
MOBY Mode	RF200, RF300, RF600, MOBY D/U, MV	RF200, RF300, RF600, MOBY D/U, MV	Solo se soporta el modo de operación "RF200, RF300, RF600, MOBY D/U, MV". Para obtener más información sobre el bloque de función reset consulte la documentación del módulo de comunicación RF120C con bloques de aplicación para S7-1200 y S7-1500.*

* Encontrará información sobre la documentación aquí (http://support.automation.siemens.com/WW/view/es/10805817).

8.1.5.6 Periferia descentralizada

Sistemas de periferia descentralizada

SIMATIC ET 200 - La solución adecuada para cada aplicación

Con SIMATIC ET 200 es posible elegir entre distintos sistemas de periferia descentralizada.

- Soluciones utilizables en el armario de distribución
- Soluciones sin armario de distribución directamente en la máquina

Además, existen componentes utilizables en áreas con riesgo de explosión. Los sistemas SIMATIC ET 200 con estructura sin armario de distribución se instalan en el interior de una carcasa de plástico robusta, reforzada con fibra de vidrio y, por ello, están protegidos a prueba de golpes, son resistentes a la suciedad y estancos al agua.

La estructura modular permite p. ej. escalar y ampliar los sistemas ET 200 fácilmente y a pequeños pasos. Los módulos adicionales integrados reducen los costes y ofrecen al mismo tiempo posibilidades de aplicación muy diversas. Es posible elegir entre las más variadas posibilidades de combinación:

- entradas y salidas digitales y analógicas
- módulos inteligentes con funcionalidad CPU,
- técnica de seguridad,
- componentes neumáticos,
- convertidores de frecuencia
- y diversos módulos tecnológicos.

La comunicación a través de PROFIBUS y PROFINET, la ingeniería única, las posibilidades de diagnóstico transparente y la conexión óptima a controladores SIMATIC y dispositivos HMI dan prueba de la universalidad de Totally Integrated Automation.

La tabla siguiente muestra un resumen de las unidades periféricas de uso con armario de distribución:

Unidad periférica	Propiedades
ET 200S	• Estructura altamente modular con conexión a varios hilos • Multifuncionalidad gracias a la amplia gama de módulos • Aplicación en área con riesgo de explosión (zona 2)
ET 200S COMPACT	• Estructura altamente modular con conexión a varios hilos • Multifuncionalidad gracias a la amplia gama de módulos • Aplicación en área con riesgo de explosión (zona 2) • DI/DO integradas
ET 200L	• Periferia de bloque digital de bajo coste • Bloques electrónicos digitales hasta 32 canales
ET 200M	• Estructura modular con módulos estándar de SIMATIC S7-300 • Módulos E/S de seguridad positiva • Aplicación en área con riesgo de explosión hasta la zona 2, sensores y actuadores hasta la zona 1 • Alta disponibilidad de la instalación, p. ej. gracias a la extracción e inserción durante el funcionamiento.
ET 200iSP	• Estructura modular, con posibilidad de redundancia • Diseño robusto y seguro • Aplicación en área con riesgo de explosión hasta la zona 1/21; puede haber sensores y actuadores incluso en la zona 0/20 • Alta disponibilidad de la instalación, p. ej. gracias a la extracción e inserción durante el funcionamiento.

La tabla siguiente muestra un resumen de las unidades periféricas de uso sin armario de distribución:

Unidad periférica	Propiedades
ET 200pro	• Estructura modular con carcasa compacta • Montaje sencillo • Multifuncionalidad gracias a la amplia gama de módulos • Alta disponibilidad de la instalación gracias a la extracción e inserción durante el funcionamiento y al cableado fijo. • Diagnóstico completo
ET 200eco PN	• Periferia de bloque de bajo coste que ahorra espacio • Módulos digitales con un máximo de 16 canales (también parametrizables) • Módulos analógicos, maestro IO-Link y distribuidor de la tensión de carga • Conexión PROFINET con switch de 2 puertos en cada módulo • Distribución flexible mediante PROFINET en estructura lineal y/o de estrella directamente en la instalación
ET 200eco	• Periferia de bloque digital de bajo coste • Posibilidades de conexión flexibles • Módulos de seguridad positiva • Alta disponibilidad de la instalación
ET 200R	• Especial para uso en robots • Montaje directo en el chasis • Resistente a las salpicaduras de soldadura gracias a la robusta carcasa metálica

Consulte también

Documentación del ET 200L (http://support.automation.siemens.com/WW/view/de/1142908/0/es)

Documentación del ET 200S (http://support.automation.siemens.com/WW/view/de/6008567/0/es)

Documentación del ET 200M (http://support.automation.siemens.com/WW/view/de/1142798/0/es)

Documentación del ET 200pro (http://support.automation.siemens.com/WW/view/de/21210852/0/es)

Documentación del ET 200iSP (http://support.automation.siemens.com/WW/view/es/28930789/0/es)

Documentación del ET 200R (http://support.automation.siemens.com/WW/view/de/11966255/0/en)

Documentación del ET 200eco PN (http://support.automation.siemens.com/WW/view/de/29999018/0/en)

Documentación del ET 200eco (http://support.automation.siemens.com/WW/view/de/12403834/0/en)

ET 200iSP

Unidad periférica descentralizada ET 200iSP

Definición

La unidad periférica descentralizada ET 200iSP es un esclavo DP altamente modular y de seguridad intrínseca con grado de protección IP 30.

Campo de aplicación

La unidad periférica descentralizada ET 200iSP se puede utilizar en áreas con riesgo de explosión con atmósferas cargadas de gas o polvo:

Homologación	Estación ET 200iSP*	Entradas y salidas
ATEX	Zona 1, Zona 21	hasta Zona 0, Zona 20 **
IECEx	Zona 2, Zona 22	hasta Zona 0, Zona 20 **
* en combinación con una carcasa apropiada ** en módulo electrónico 2 DO Relay UC60V/2A: hasta Zona 1, Zona 21		

Evidentemente, la unidad periférica descentralizada ET 200iSP también se puede utilizar en áreas seguras.

Los módulos de periferia del ET 200iSP se pueden insertar casi en cualquier combinación posible, directamente junto al módulo de interfaz que transmite los datos al maestro DP. De esta manera, es posible adaptar la configuración a las demandas concretas del lugar de instalación.

Cada ET 200iSP consta de una fuente de energía, un módulo de interfaz y 32 módulos electrónicos como máximo (p. ej. módulos electrónicos digitales). Tenga en cuenta el consumo máximo.

Módulos electrónicos y de pines

La unidad periférica ET 200iSP está formada básicamente por varios módulos de pines pasivos, a los que se enchufan la fuente de alimentación y los módulos electrónicos.

El ET 200iSP se conecta al PROFIBUS RS 485-IS con un conector situado en el módulo de pines TM-IM/EM. Cada ET 200iSP es un esclavo DP en el PROFIBUS RS 485-IS.

Maestro DP

Todos los módulos del ET 200iSP se pueden comunicar con todos los maestros DP que cumplen la norma *IEC 61784-1:2002 Ed1 CP 3/1* con el protocolo de transferencia "DP" (DP significa Periferia Descentralizada).

Consulte también

Documentación del ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789/0/en)

Asignar canal y variable IEEE

Propiedades

Los módulos electrónicos analógicos 4 AI I 2WIRE/HART, 4 AI I 4WIRE/ HART y 4 AO I HART soportan hasta 4 variables IEEE.

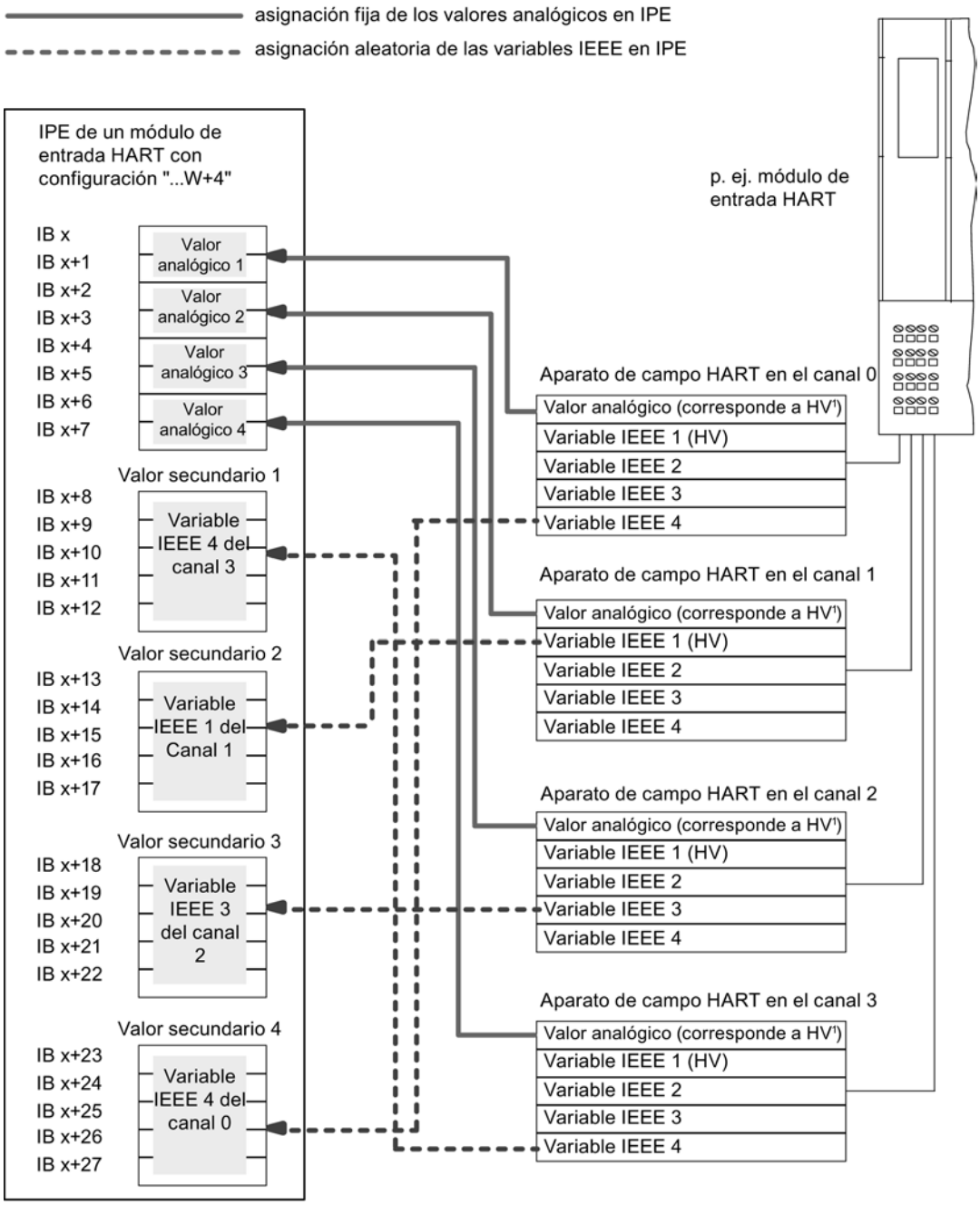
En la memoria imagen de proceso de las entradas (IPE) hay un máximo de 20 bytes en cada módulo para las variables IEEE. Así pues, para las 4 variables IEEE hay 4 bloques con 5 bytes cada uno dentro de la IPE.

Requisitos

El aparato de campo HART debe soportar el número parametrizado de variables IEEE.

Asignar variables IEEE

Las variables IEEE de los aparatos de campo se asignan a voluntad a los bloques de la IPE.



Consulte también

Documentación del ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789/0/en)

Parametrizar uniones frías para termopares

Compensación de la temperatura de las uniones frías

Existen varias posibilidades de registrar la temperatura de las uniones frías con el fin de calcular un valor de temperatura absoluto a partir de la diferencia de temperatura entre unión fría y punto de medición.

Tabla 8-112 Compensación de la temperatura de las uniones frías

Posibilidad	Explicación	Parámetro Unión fría
Sin compensación	No sólo se registra la temperatura del punto de medición. La temperatura de la unión fría (transición de cable Cu a cable de termopar) influye además en la tensión termoeléctrica. El valor medido tiene así un error.	Ninguno
Utilización de una termorresistencia Pt 100 Climatiz. para registrar la temperatura de las uniones frías (método idóneo)	Es posible registrar la temperatura de las uniones frías con una termorresistencia (Pt 100 Climatiz.). Este valor de temperatura se distribuye a los módulos 4 AI TC en el ET 200iSP con la respectiva parametrización y se calcula en los módulos con el valor de temperatura registrado en el punto de medición. Número de uniones frías: 2	La parametrización del IM152 y del 4 AI TC debe estar sintonizada: • 4 AI RTD parametrizado en Pt100 Climatiz. en el slot correcto; • 4 AI TC: unión fría: "sí"; seleccionar número de unión fría "1" ó "2" • IM 152-1: asignación de la unión fría a un slot con 4 AI RTD; selección de un canal;
Compensación interna 4 AI TC	El módulo de sensores TC (sensor de temperatura) se monta en los bornes del módulo de pines del EM 4 AI TC. El sensor de temperatura notifica la temperatura de los bornes al 4 AI TC. Dicho valor se calcula con el valor medido en el canal del módulo electrónico.	• 4 AI TC: número de unión fría "Interno"

Compensación con termorresistencia en el 4 AI RTD

Cuando los termopares conectados a las entradas de los 4 AI TC tienen la misma unión fría, se compensan con un 4 AI RTD.

Para los canales del módulo 4 AI TC es posible seleccionar "1", "2" o "Interno" como número de unión fría. Si se selecciona "1" ó "2", se utiliza siempre la misma unión fría (canal RTD) para los 4 canales.

Parametrización de la unión fría

Las uniones frías para módulos electrónicos 4 AI TC se ajustan con los parámetros siguientes:

Tabla 8-113 Parámetros de la unión fría

Parámetro	Módulo	Rango de valores	Explicación
Slot de unión fría 1 a slot 2	IM 152	ninguno, 4 a 35	Este parámetro permite asignar hasta 2 slots (ninguno, 4 a 35) en los que se encuentran los canales de medición de la temperatura de referencia (determinación del valor de compensación).
Entrada de unión fría 1 a 4 entrada de unión fría	IM 152	RTD a canal 0 RTD a canal 1 RTD a canal 2 RTD a canal 3	Con este parámetro se determina el canal (0/1/2/3) para la medición de la temperatura de referencia (determinación del valor de compensación) para el slot asignado.
Unión fría E0 a unión fría E3	4 AI TC	Ninguno sí	Este parámetro permite habilitar la utilización de la unión fría.
Número de unión fría	4 AI TC	1 2 Interno	Con este parámetro se asigna la unión fría (1, 2) que contiene la temperatura de referencia (valor de compensación).

Consulte también

Documentación del ET 200iSP (http://support.automation.siemens.com/WW/view/de/28930789/0/en)

Principios básicos del sellado de tiempo

Propiedades

El sellado de tiempo con el IM 152 es posible en aplicaciones de clientes que utilizan el FB 62 (FB TIMESTMP).

Funcionamiento

Una señal de entrada modificada se marca con un sello de tiempo y se guarda en un búfer (registro). Si hay señales con sello de tiempo o un registro está lleno, se genera una alarma de proceso en el maestro DP. El búfer se evalúa con "Leer registro". En los eventos que influyen en el sellado de tiempo (comunicación con maestro DP interrumpida, fallo de telegrama del reloj maestro, etc.) se generan avisos especiales.

Parametrizar

Con la parametrización se define qué datos de usuario del IM 152 se vigilan. En el sellado de tiempo se trata de las entradas digitales cuyos cambios de señal se vigilan.

Parámetro	Ajuste	Descripción
Sellado de tiempo	• bloqueado • habilitado	Active el sellado de tiempo para los canales del módulo electrónico 8 DI NAMUR.
Evaluación de flancos evento entrante	• Flanco ascendente • Flanco descendente	Determine el tipo de cambio de señal que se marca con un sello de tiempo.

Contaje

Propiedades Contaje

Funciones de contaje

El módulo electrónico 8 DI NAMUR incorpora funciones de contaje parametrizables:

- 2 contadores ascendentes de 16 bits (función de contaje normal) o
- 2 contadores descendentes de 16 bits (función de contaje periódica) o
- 1 contador descendente de 32 bits (función de contaje en cascada)
- Preajuste de una consigna mediante la IPS
- Función PUERTA
- La función de las señales de control de los contadores se puede configurar:
  - Configuración de canal 0..1: "Contador", canal 2..7: "DI": se configuran 2 contadores. Las señales de control de los contadores se han guardado en la IPS (memoria imagen de proceso de las salidas).
  - Configuración de canal 0..1: "Contador", canal 2..7: "Control": se configuran 2 contadores. Las señales de control de los contadores se han guardado en la IPS. Además son controladas por las entradas digitales del 8 DI NAMUR.

Consulte también

Funcionamiento (Página 2104)

Configurar contadores (Página 2106)

Parametrizar contadores (Página 2109)

Funcionamiento

Contador ascendente de 16 bits (función de contaje normal)

El rango de contaje es de 0 a 65535.

Con cada impulso de contaje en la entrada digital aumenta el estado del contador en el valor 1. Al alcanzarse el límite de contaje, el estado del contador se pone a 0 y se incrementa nuevamente a partir de este valor.

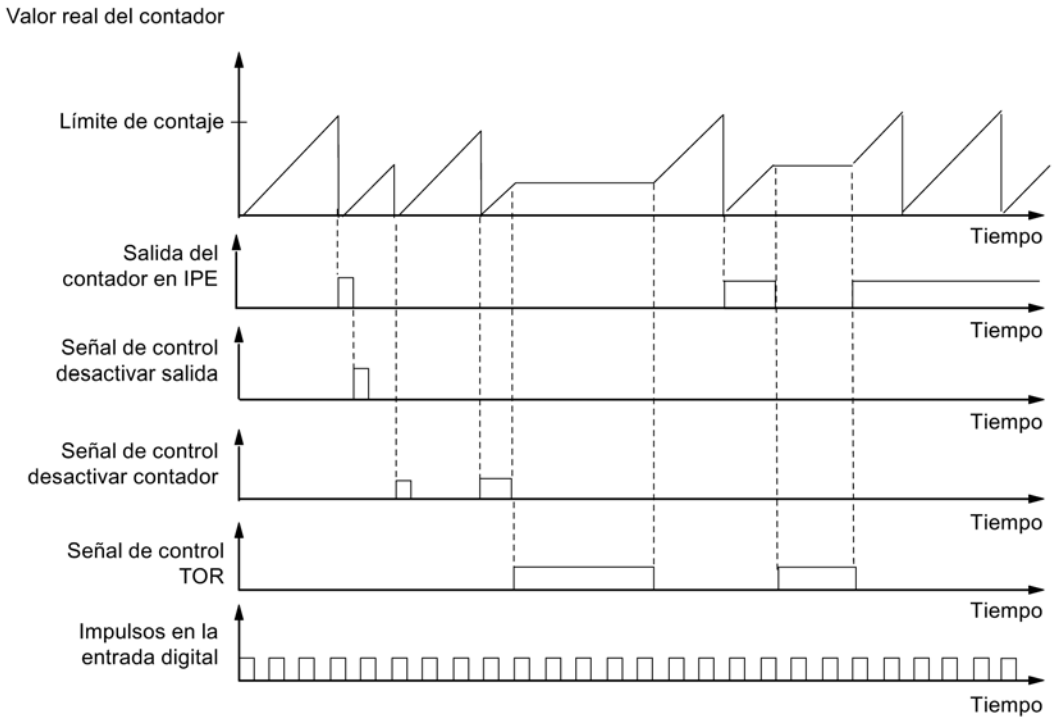
En caso de desbordamiento del contador, la salida correspondiente se activa en la IPE.

A través del flanco ascendente de la señal de control *Desactivar salida* se desactiva la salida en la IPE. El valor de contaje actual no se altera por ello.

En el contaje hacia delante de 16 bits no se activan salidas en la IPS. Éstas se encuentran en general desactivadas.

El flanco ascendente de la señal de control *Desactivar contador* pone el estado del contador a 0 y desactiva la salida del contador activada.

La señal de control *PUERTA* detiene el proceso de contaje en un flanco ascendente. No se vuelven a procesar impulsos de contaje en la entrada digital hasta que llega un flanco descendente. La señal de control *Desactivar contador* también funciona con la *PUERTA* activada.



Contador descendente de 16 bits (función de contaje periódica)

El rango de contaje máximo es de 65535 a 0.

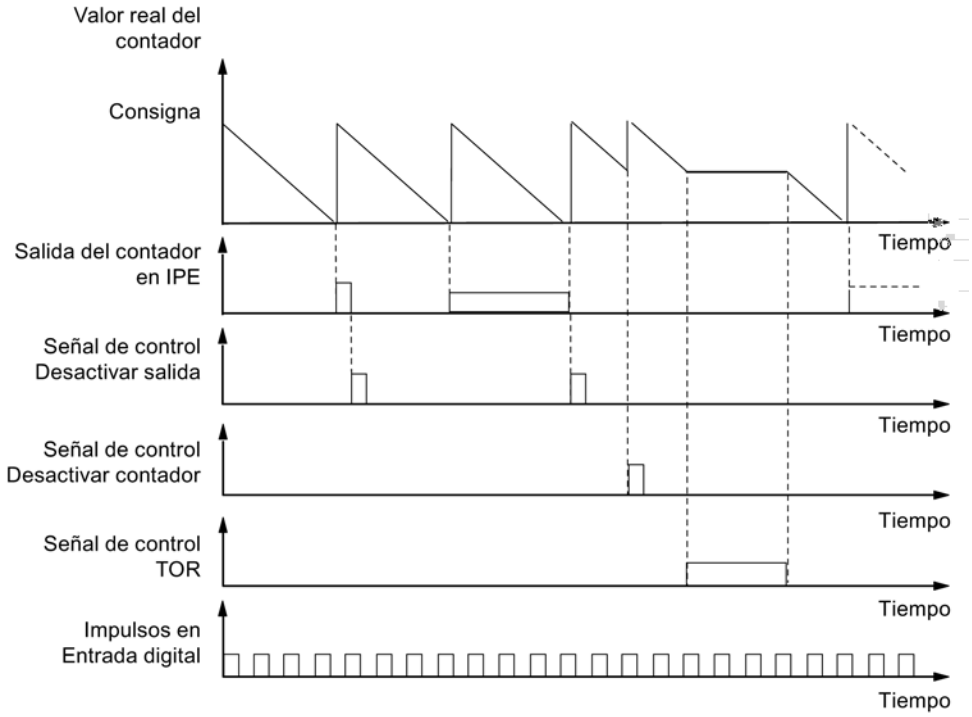
En el instante de inicio del contador, el valor real se pone a la consigna predeterminada. Con cada impulso de contaje se reduce el valor real en 1. Cuando el valor real llega a 0, se activa la respectiva salida en la IPE y el valor real se pone nuevamente a la consigna predeterminada. A partir de este valor se vuelve a contar hacia atrás.

El flanco ascendente de la señal de control *Desactivar contador* pone el valor real a la consigna predeterminada y desactiva la respectiva salida en la IPE.

A través del flanco ascendente de la señal de control *Desactivar salida* se desactiva la salida en la IPE. El valor de contaje actual no se altera por ello.

La señal de control *PUERTA* detiene el proceso de contaje en un flanco ascendente. Al mismo tiempo se desactiva la salida asignada en la IPE. No se vuelven a procesar impulsos de contaje en la entrada digital hasta que llega un flanco descendente. Las señales de control *Desactivar salida* y *Desactivar contador* también funcionan con la *PUERTA* activada.

La consigna del contador se preajusta y modifica mediante la IPS. La consigna se aplica con el flanco ascendente de la señal de control *Desactivar contador* o con un paso por cero del contador.



Contador descendente de 32 bits (función de contaje en cascada)

El rango de contaje máximo es de 4294967295 a 0.

El funcionamiento es idéntico al del contador descendente de 16 bits. El canal 1 no tiene ninguna función.

Consulte también

Propiedades Contaje (Página 2103)

Configurar contadores

Procedimiento

1. Arrastre el módulo 8 DI Namur con el ratón desde el catálogo de hardware hasta la unidad periférica descentralizada ET 200iSP.
2. Seleccione la configuración deseada (canal 0..1: "Contador", canal 2..7: "DI" o "Control"). En las propiedades del módulo (ventana de inspección) se encuentra este ajuste en "Parámetros > Entradas > Configuración".

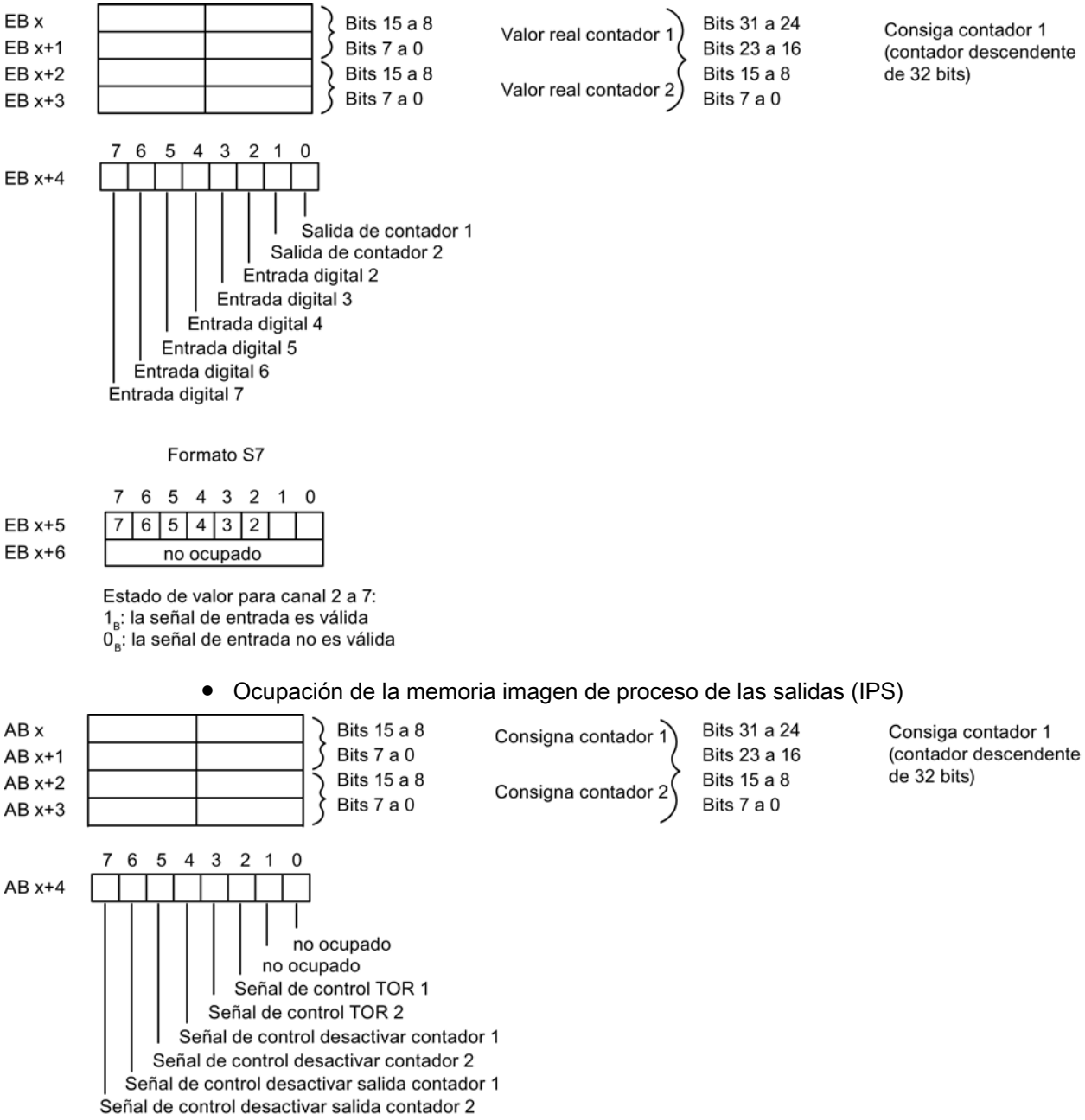
Configuración de canal 0..1: "Contador", canal 2..7: "DI"

- Ocupación de las entradas digitales en el módulo electrónico 8 DI NAMUR

Tabla 8-114 Ocupación de las entradas digitales en el canal 0..1: "Contador", canal 2..7: "DI":

Entrada digital	Borne	Ocupación
Canal 0	1, 2	Contador 1
Canal 1	5, 6	Contador 2 (irrelevante con contador descendente de 32 bits)
Canal 2	9, 10	Entrada digital 2
Canal 3	13, 14	Entrada digital 3
Canal 4	3, 4	Entrada digital 4
Canal 5	7, 8	Entrada digital 5
Canal 6	11, 12	Entrada digital 6
Canal 7	15, 16	Entrada digital 7

- Ocupación de la memoria imagen de proceso de las entradas (IPE)



- Ocupación de la memoria imagen de proceso de las salidas (IPS)

Configuración de canal 0..1: "Contador", canal 2..7: "CONTROL"

Esta configuración permite controlar los contadores además con las entradas digitales.

- Ocupación de las entradas digitales en el módulo electrónico 8 DI NAMUR
  Encontrará más información sobre la asignación de pines en los Datos técnicos del módulo electrónico 8 DI NAMUR.

Tabla 8-115 Ocupación de las entradas digitales en 2 Count/ 6 Control

Entrada digital	Borne	Ocupación
Canal 0	1, 2	Contador 1
Canal 1	5, 6	Contador 2 (irrelevante con contador descendente de 32 bits)
Canal 2	9, 10	Señal de control *PUERTA 1*
Canal 3	13, 14	Señal de control *PUERTA 2*
Canal 4	3, 4	Señal de control *Desactivar contador 1*
Canal 5	7, 8	Señal de control *Desactivar contador 2*
Canal 6	11, 12	Señal de control *Desactivar salida contador 1*
Canal 7	15, 16	Señal de control *Desactivar salida contador 2*

- Ocupación de la memoria imagen de proceso de las entradas (IPE)
  La ocupación es idéntica a la configuración 0..1: "Contador", canal 2..7: "DI".
- Ocupación de la memoria imagen de proceso de las salidas (IPS)
  La ocupación es idéntica a la configuración 0..1: "Contador", canal 2..7: "DI".

Consulte también

Propiedades Contaje (Página 2103)

Parametrizar contadores

Parámetros de la función de contaje

A continuación se explican sólo los parámetros relevantes para los contadores. Éstos forman parte de los parámetros del módulo electrónico 8 DI NAMUR y dependen de la configuración seleccionada:

Tabla 8-116 Parámetros de los contadores

Parámetro	Ajuste	Descripción
Tipo de encoder Entradas de contador	• Canal bloqueado • Encoder NAMUR • Contacto individual sin conectar	Para el canal 0 ó 1, seleccione el encoder del respecto contador.
Modo de operación Contador 1	• Función de contaje normal • Función de contaje periódica • Función de contaje en cascada	Seleccione el modo de operación del contador 1.
Modo de operación Contador 2	• Función de contaje normal • Función de contaje periódica • Función de contaje en cascada	Seleccione el modo de operación del contador 2. Este parámetro no es relevante si se ha ajustado el parámetro "Modo de operación Contador 1" en "Función de contaje en cascada".

Consulte también

Propiedades Contaje (Página 2103)

Medición de frecuencia

Propiedades de la medición de frecuencia

Propiedades

El módulo electrónico 8 DI NAMUR permite medir frecuencias en los canales 0 y 1:

- 2 frecuencímetros de 1 Hz a 5 kHz
- Ventana de medición parametrizable (PUERTA)
- Las señales de los frecuencímetros se leen con las entradas digitales del módulo electrónico.

Consulte también

Funcionamiento (Página 2110)

Configurar frecuencímetros (Página 2110)

Parametrizar frecuencímetros (Página 2112)

Funcionamiento

Medición de frecuencia

Las frecuencias de señal se determinan a partir de las señales de entrada del canal 0 ó 1 del módulo electrónico. Para calcular la frecuencia se miden las señales dentro de una ventana de medición parametrizable.

La frecuencia se representa como valor de 16 bits en formato de coma fija y se introduce en la IPE.

Los frecuencímetros calculan la frecuencia aplicando la fórmula siguiente:

$$\text{Frecuencia [Hz]} = \frac{\text{Número de flancos ascendentes en la entrada digital}}{\text{Ventana de medición [s]}}$$

Rebase por exceso de la frecuencia de entrada

Cuando la frecuencia de entrada supera los 5 kHz, se notifica $7FFF_H$ como valor real. Cuando la frecuencia de entrada es superior a 8 kHz ya no se pueden emitir más valores reales correctos.

Consulte también

Propiedades de la medición de frecuencia (Página 2109)

Configurar frecuencímetros

Procedimiento

1. Arrastre el módulo 8 DI Namur con el ratón desde el catálogo de hardware hasta la unidad periférica descentralizada ET 200iSP.
2. Seleccione la configuración deseada (canal 0..1: "Trace", canal 2..7: "DI"). En las propiedades del módulo (ventana de inspección) se encuentra este ajuste en "Parámetros > Entradas > Configuración".

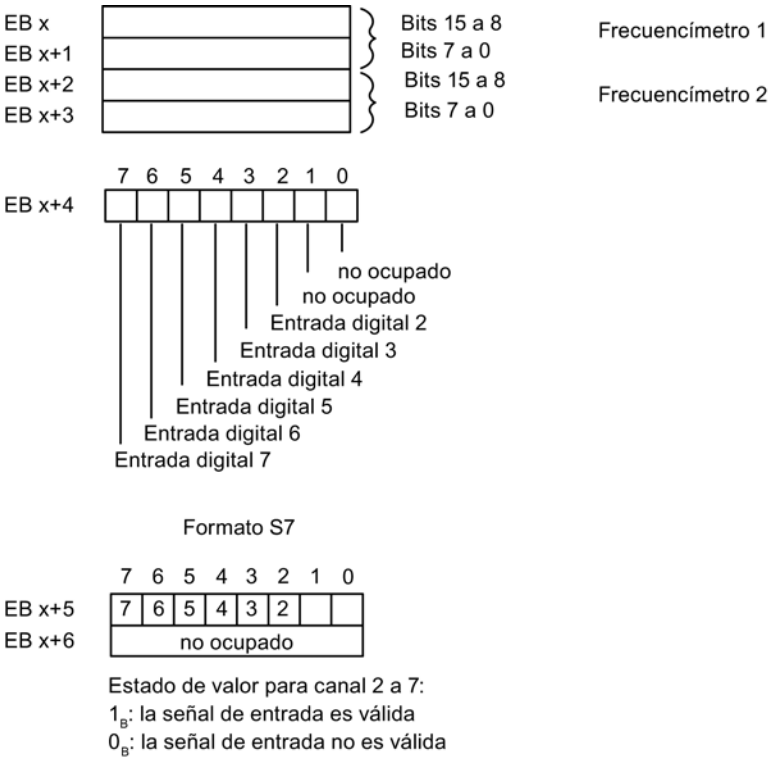
Configuración 0..1: "Trace", canal 2..7: "DI"

Ocupación de las entradas digitales en el módulo electrónico 8 DI NAMUR

Entrada digital	Borne	Ocupación
Canal 0	1, 2	Frecuencímetro 1
Canal 1	5, 6	Frecuencímetro 2
Canal 2	9, 10	Entrada digital 2
Canal 3	13, 14	Entrada digital 3
Canal 4	3, 4	Entrada digital 4
Canal 5	7, 8	Entrada digital 5

Entrada digital	Borne	Ocupación
Canal 6	11, 12	Entrada digital 6
Canal 7	15, 16	Entrada digital 7

Ocupación de la memoria imagen de proceso de las entradas (IPE) en la configuración del canal 0..1: "Trace", canal 2..7: "DI"



Ocupación de la memoria imagen de proceso de las salidas (IPS): La IPS no está ocupada.

Consulte también

Propiedades de la medición de frecuencia (Página 2109)

Parametrizar frecuencímetros

Parámetros de frecuencímetros

A continuación se explican sólo los parámetros relevantes para los frecuencímetros. Éstos forman parte de los parámetros del módulo electrónico 8 DI NAMUR:

Tabla 8-117 Parámetros de los frecuencímetros

Parámetro	Ajuste	Descripción
Tipo de encoder Entradas de frecuencia	• Canal bloqueado • Encoder NAMUR • Contacto individual sin conectar	Para el canal 0 ó 1, seleccione el encoder del respectivo frecuencímetro.
Ventana de medición (PUERTA)	• 50 ms • 200 ms • 1 s	Para el canal 0 ó 1, seleccione la ventana de medición necesaria. Para alcanzar una precisión lo más alta posible en la medición de frecuencia, siga las reglas siguientes: • Frecuencias altas (>4kHz): ajustar ventana de medición baja (50 ms) • Frecuencias variables/medias: ajustar ventana de medición media (200 ms) • Frecuencias bajas (<1kHz): ajustar ventana de medición alta (1 s)

Consulte también

Propiedades de la medición de frecuencia (Página 2109)

ET 200eco PN

Unidad periférica descentralizada ET 200eco PN

Definición

La unidad periférica descentralizada ET 200eco PN es un dispositivo PROFINET IO compacto con grado de protección IP 65/66 o IP 67 y UL Enclosure Tipo 4x, Indoor use only.

Campo de aplicación

Los campos de aplicación del ET 200eco PN resultan de sus características especiales.

- La robustez del diseño y el grado de protección IP 65/66 o IP 67 hacen que la unidad de periferia descentralizada ET 200eco PN sea especialmente apropiada para el uso en entornos industriales rudos.
- La estructura compacta del ET 200eco PN permite su uso en zonas de espacio reducido.
- La fácil manipulación del ET 200eco PN garantiza una puesta en marcha y un mantenimiento rápidos.

Propiedades

El ET 200eco PN presenta las siguientes características:

- Switch integrado con 2 puertos
- Servicios Ethernet soportados:
  - ping
  - arp
  - Diagnóstico de red (SNMP)
  - LLDP
- Alarmas
  - Alarmas de diagnóstico
  - Alarmas de mantenimiento
- Diagnóstico de puerto
- Comunicación Isochronous Real-Time
- Arranque priorizado
- Sustitución de dispositivo sin PG
- Redundancia de medios
- Conexión con sensores/actuadores inteligentes mediante el módulo de interfaz IO-Link Master.

Controlador IO

El ET 200eco PN puede comunicarse con todos los controladores IO que se comportan conforme a la norma IEC 61158.

El ET 200eco PN es configurable en una CPU con diagnóstico avanzado.

Consulte también

Documentación del ET 200eco PN (http://support.automation.siemens.com/WW/view/en/29999018)

Descripción de parámetros de entrada analógica

Diagnóstico de grupo

Con este parámetro se habilita y se bloquea en general el diagnóstico del dispositivo.

Los diagnósticos "Error" y "Error de parametrización" están habilitados siempre, con independencia del diagnóstico de grupo.

Diagnóstico falta 1L+

Si se habilita este parámetro, se habilita la comprobación de falta de tensión de alimentación.

Diagnóstico cortocircuito de la alimentación del encoder

Si se habilita este parámetro, se genera un diagnóstico en caso de cortocircuito a masa de la alimentación del encoder con el canal activado. La alimentación del encoder se vigila para los conectores X1, X3, X5 y X7. No es posible diferenciar en qué conector se produce el cortocircuito del encoder.

Supresión de frecuencias perturbadoras

Con este parámetro se ajusta el tiempo de integración del dispositivo en función de la selección de las frecuencias perturbadoras. Seleccione aquí la frecuencia de la tensión de red utilizada. Supresión de frecuencias perturbadoras **desactivada** significa 500 Hz, lo que equivale a un tiempo de integración de 2 ms en un canal de medición.

Unidad de temperatura

Ajuste aquí la unidad en que quiere registrar la temperatura.

Tipo de medición (por canales)

Con este parámetro se ajusta el tipo de medición, p. ej. la tensión. Si no se utiliza un canal, hay que elegir el ajuste **desactivado**. Cuando un canal está desactivado, se optimiza el tiempo de conversión y de integración del canal = 0 s y el tiempo de ciclo.

Rango de medición

Con este parámetro se ajusta el rango del tipo de medición seleccionado.

Coeficiente de temperatura (en RTD, termorresistencia)

El factor de corrección para los coeficientes de temperatura (valor α) indica cuál es el cambio relativo en la resistencia de un determinado material cuando la temperatura aumenta 1 °C.

Los valores α corresponden a las normas EN 60751, GOST 6651, JIS C 1604 y ASTM E-1137.

El coeficiente de temperatura depende de la composición química del material.

Alisamiento

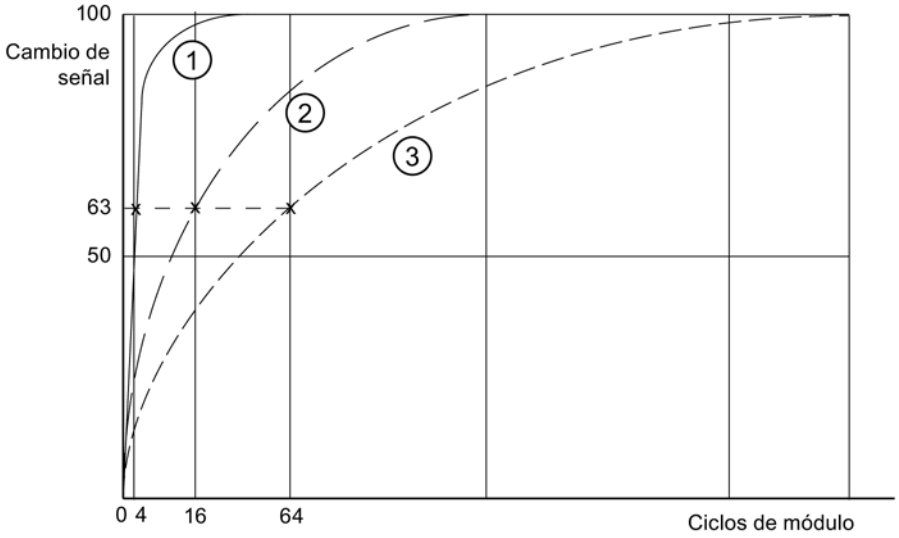
Gracias al alisamiento de los valores analógicos, se genera una señal analógica estable para el procesamiento posterior. El alisamiento de los valores analógicos es aconsejable en caso de cambios lentos en la señal útil (cambios en los valores medidos), p. ej. en las mediciones de temperatura.

Los valores medidos se alisan mediante filtrado digital. El alisamiento se consigue haciendo que el dispositivo calcule valores medios a partir de un número definido de valores analógicos convertidos (digitalizados).

El alisamiento se parametriza en máximo 4 niveles (ninguno, débil, medio, fuerte). El nivel determina el número de ciclos de módulos necesarios para calcular el valor medio.

Cuando más fuerte es el alisamiento, más estable es el valor analógico alisado y más tiempo pasa hasta que llega el valor analógico alisado tras un cambio de señal (véase ejemplo siguiente).

La figura siguiente muestra tras cuántos ciclos de módulo el valor analógico alisado se aproxima al 100% en una respuesta indicial, dependiendo del alisamiento ajustado. La figura se aplica a cualquier cambio de señal en la entrada analógica. El valor de alisamiento indica el número de ciclos que necesita el módulo para alcanzar el 63% del valor final de los cambios de señal.



① Alisamiento débil
② Alisamiento medio
③ Alisamiento fuerte

Diagnóstico Rotura de hilo

Si se habilita este parámetro, se genera el diagnóstico **Rotura de hilo** cuando se rompe un hilo.

En caso de rotura de hilo en los rangos de medición 1 a 5 V y 4 a 20 mA, tenga en cuentas las reglas siguientes:

Parámetro	Evento	Valor medido	Explicación
Rotura de hilo habilitada[1]	Rotura de hilo	$7FFF_H$	Diagnóstico **Rotura de hilo**
Rotura de hilo bloqueada[1] Rebase por defecto habilitado	Rotura de hilo	8000_H	Valor medido después de abandonar el rango de saturación por defecto Aviso de diagnóstico **Rebase del límite inferior**
Rotura de hilo bloqueada[1] Rebase por defecto bloqueado	Rotura de hilo	8000_H	Valor medido después de abandonar el rango de saturación por defecto
[1] Límites del rango de medición para detección de rotura de hilo y rebase por defecto del rango de medición: • de 1 a 5: a 0,296 V • de 4 a 20 mA: a 1,185 mA			

Diagnóstico Rebase por defecto

Cuando se habilita este parámetro, se genera el diagnóstico **Rebase por defecto** si el valor medido alcanza el rango de rebase por defecto.

Diagnóstico Rebase por exceso

Cuando se habilita este parámetro, se genera el diagnóstico **Rebase por exceso** si el valor medido alcanza el rango de rebase por exceso.

Unión fría para termorresistencia (TC)

Si se expone el punto de medición a una temperatura diferente a la de los extremos libres del termopar (punto de conexión), se genera una tensión entre los extremos libres, la tensión termoeléctrica. La intensidad de la tensión termoeléctrica depende de la diferencia entre la temperatura del punto de medición y la temperatura en los extremos libres, así como del tipo de combinación de materiales con que está hecho el termopar. Como siempre se registra una diferencia de temperatura con un termopar, los extremos libres en la unión fría deben mantenerse en una temperatura conocida para poder determinar la temperatura del punto de medición.

Cuando se ajusta la **Compensación interna**, se mide la temperatura del punto de medición en la carcasa de la unidad periférica. Al ajustar la **Compensación externa** es posible anteponer una caja de compensación para lograr una mayor exactitud en la medición de temperatura.

Descripción de parámetros salida analógica

Diagnóstico de grupo

Con este parámetro se habilita y se bloquea en general el diagnóstico del dispositivo.

Los diagnósticos "Error" y "Error de parametrización" están habilitados siempre, con independencia del diagnóstico de grupo.

Diagnóstico falta 1L+

Si se habilita este parámetro, se habilita la comprobación de falta de tensión de alimentación.

Diagnóstico cortocircuito de la alimentación del encoder

Si se habilita este parámetro, se genera un diagnóstico en caso de cortocircuito a masa de la alimentación del encoder. Este diagnóstico se habilita cuando se habilita el diagnóstico de grupo.

Respuesta a STOP de la CPU/el maestro

Seleccione cómo deben reaccionar las salidas del módulo en caso de STOP de la CPU:

- Desconectar
  La unidad periférica pasa al estado seguro. La memoria imagen de proceso de las salidas se borra (=0).
- Mantener último valor
  La unidad periférica mantiene el último valor emitido antes del STOP.
- Valor sustitutivo
  La unidad periférica emite el valor previamente ajustado para el canal.

Nota

Asegúrese de que la instalación esté siempre en un estado seguro en caso de "Mantener último valor".

Tipo de salida

Con este parámetro se ajusta el tipo de salida, p. ej. la tensión. Si no se utiliza un canal, hay que elegir el ajuste **Desactivado**. Cuando un canal está desactivado, se optimiza el tiempo de conversión y de integración del canal = 0 s y el tiempo de ciclo.

Rango de salida

Con este parámetro se ajusta el rango de salida del tipo de salida seleccionado.

Diagnóstico Rotura de hilo (en modo de intensidad)

Si se habilita este parámetro, se genera el diagnóstico **Rotura de hilo** cuando se rompe un hilo. Este diagnóstico no se reconoce en el área cero.

Diagnóstico Cortocircuito (en modo de tensión)

Cuando se habilita este parámetro, se genera un diagnóstico en caso de cortocircuito de la línea de salida. Este diagnóstico no se reconoce en el área cero.

Diagnóstico Sobrecarga

Si se habilita este parámetro, se genera el diagnóstico en caso de sobrecarga.

Valores sustitutivos

En este parámetro se introduce un valor sustitutivo que debe emitir el módulo en caso de STOP de la CPU. El valor sustitutivo debe situarse en el rango nominal, el rango de saturación por exceso o el rango de saturación por defecto.

ET 200SP

Sistema de periferia descentralizada ET 200SP

Definición

El ET 200SP es un sistema de periferia descentralizada altamente escalable y flexible que permite conectar las señales del proceso a un controlador central a través de un bus de campo.

Campo de aplicación

El ET 200SP es un sistema de periferia descentralizada multifuncional para diferentes campos de aplicación. Gracias a su estructura escalable es posible adaptar la configuración exactamente a las exigencias del proceso.

El ET 200SP está homologado para el grado de protección IP 20 y su montaje está previsto en un armario de distribución.

Estructura

El ET 200SP se monta sobre un perfil soporte y se compone de:

- un módulo de interfaz que se comunica con todos los controladores IO que se comportan conforme a la norma PROFINET IEC 61158.
- hasta un máximo de 32 módulos de periferia que se enchufan en BaseUnits pasivas en cualquier combinación.
- un módulo servidor que cierra la instalación del ET 200SP.

Parámetro Módulo de interfaz

Bytes de estado

Bytes de estado

Si se activa la opción "Bytes de estado", se reservan 4 bytes de datos de entrada para el estado de la tensión de alimentación de cada módulo de periferia.

	7	6	5	4	3	2	1	0
Byte 0	8	7	6	5	4	3	2	1
Byte 1	16	15	14	13	12	11	10	9
Byte 2	24	23	22	21	20	19	18	17
Byte 3	32	31	30	29	28	27	26	25

Slots de los módulos de periferia

Bit = 0: Falta tensión de carga o el módulo de periferia no existe

Bit = 1: Tensión de carga y módulo de periferia existentes

Nota

Un módulo de servidor, ya esté insertado o ausente, notifica siempre el bit = 0 para el slot.

Diagnóstico colectivo Falta tensión de alimentación L+

Diagnóstico colectivo Falta tensión de alimentación L+

Este diagnóstico es un diagnóstico agrupado del estado de la tensión de alimentación de todos los módulos de periferia de un grupo de potencial que reciben tensión a través de BaseUnits (BaseUnit BU...D clara).

El diagnóstico agrupado se forma a partir de los estados de la tensión de alimentación de los módulos de periferia enchufados dentro del grupo de potencial.

El diagnóstico agrupado no depende de la habilitación del parámetro "Falta tensión de alimentación L+" de los módulos de periferia.

El módulo servidor no influye en el diagnóstico agrupado Falta tensión de alimentación L+.

Requisitos para el funcionamiento correcto del diagnóstico agrupado Falta tensión de alimentación L+:

- En las BaseUnits claras y oscuras tiene que haber enchufados módulos de periferia o BU Cover.
  Si en las BaseUnit claras no hay ningún módulo de periferia enchufado, el módulo de interfaz no distingue el principio de este grupo de potencial; de este modo, los módulos de periferia de este grupo de potencial pertenecen al anterior. Como consecuencia, el error colectivo Falta tensión de alimentación L+ se asignará al grupo de potencial equivocado. Si se enchufa un módulo de periferia en la BaseUnit clara, el módulo de interfaz detecta el nuevo grupo de potencial, evalúa nuevamente el estado y, en caso de error, notifica un nuevo diagnóstico agrupado.
- El módulo servidor tiene que estar enchufado.
  Sin embargo, el módulo servidor por sí mismo no influye en el diagnóstico agrupado Falta tensión de alimentación L+.

Control de la configuración

Principio de funcionamiento

Gracias al control de la configuración es posible utilizar diferentes configuraciones (opciones) reales con una sola configuración del sistema de periferia descentralizada ET 200SP.

El control de la configuración permite configurar el sistema de periferia descentralizada ET 200SP con su configuración máxima y utilizarlo aunque falten módulos. Si posteriormente se añaden algunos módulos, no es necesario configurar de nuevo y, por consiguiente, tampoco volver a cargar la configuración hardware.

Utilizando registros de control que se transmiten al módulo de interfaz desde el programa de usuario, se define una configuración teórica actual.

- En un slot no está el módulo configurado.
  - En lugar del módulo de periferia configurado, puede encontrarse una BU Cover en dicho slot. Puesto que en el slot falta el módulo configurado, se habla también de un "control de la configuración con espacios libres".
  - En lugar del módulo configurado, en este slot puede haber el módulo que en la configuración se encuentra a la derecha del módulo ausente. La configuración real aparece juntada debido al módulo que falta. Como falta el módulo configurado pero no se producen huecos en la configuración, también se habla de un "control de la configuración sin espacios libres".
- La configuración se amplía con un módulo ya configurado.
  - En el control de la configuración con espacios libres, la configuración se amplía enchufando el módulo configurado en el espacio libre correspondiente.
  - En el control de la configuración sin espacios libres, el módulo configurado se enchufa a la derecha del último módulo del ET 200SP.

Direccionamiento del módulo de interfaz mediante identificador de hardware

Para transferir el registro 196 con la instrucción WRREC hay que introducir el identificador de hardware del módulo de interfaz como parámetro de entrada para la instrucción. La constante

de sistema de dicho identificador de hardware es, p. ej., "IO-Device_2 [Head]". Las constantes de sistema de un dispositivo seleccionado se muestran en la ficha "Constantes de sistema" de la vista de redes, por ejemplo. Utilice el valor correspondiente para el direccionamiento.

Información adicional y ejemplos

Encontrará información relacionada con las reglas del control de la configuración, la estructura del registro de control, el comportamiento durante el funcionamiento y ejemplos de la estructura del registro de control para diferentes configuraciones en el manual de producto del módulo de interfaz IM 155-6 PN (http://support.automation.siemens.com/WW/view/es/59768173).

Parámetro Módulos de salida

Comportamiento de valor sustitutivo

Comportamiento de valor sustitutivo

En el ET 200SP, el comportamiento de valor sustitutivo es definido por el controlador IO para cada slot.

la salida correspondiente se comporta conforme al comportamiento parametrizado del valor sustitutivo:

- "Desconectar"
- "Aplicar valor sustitutivo"
- "Mantener último valor"

El comportamiento de valor sustitutivo se da en los casos siguientes:

- Controlador IO en STOP
- Fallo del controlador (interrupción de la conexión)
- Actualización del firmware
- Restablecimiento de la configuración de fábrica
- Extracción de varios módulos de periferia al mismo tiempo

- Desactivación del dispositivo IO
- Parada de la estación
  - Ausencia del módulo servidor
  - Extracción de varios módulos de periferia al mismo tiempo
  - Al menos un módulo de periferia está enchufado en una BaseUnit errónea

Nota

Reducir una configuración

Si se reduce la configuración del ET 200SP y se carga en la CPU, los módulos que todavía existen pero ya no están configurados conservan su comportamiento original en lo que respecta al valor sustitutivo. Éste es válido hasta que se desconecta la tensión de alimentación en la BaseUnit BU...D o en el módulo de interfaz.

Parámetro Módulos de entrada

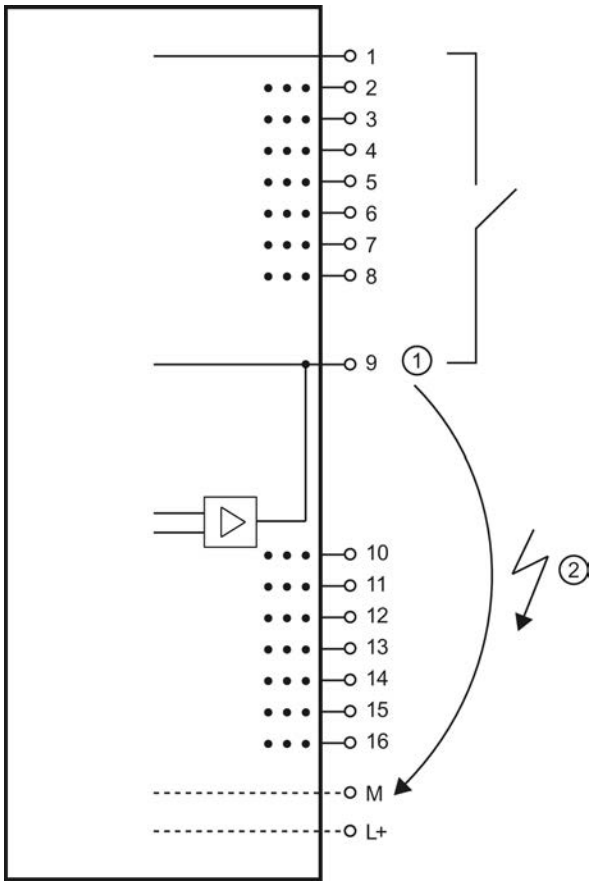
Parámetros de los módulos de entradas digitales

Diagnóstico Falta tensión de alimentación L+

Habilitación del aviso de diagnóstico cuando la tensión de alimentación L+ es insuficiente o falta.

Diagnóstico Cortocircuito a masa

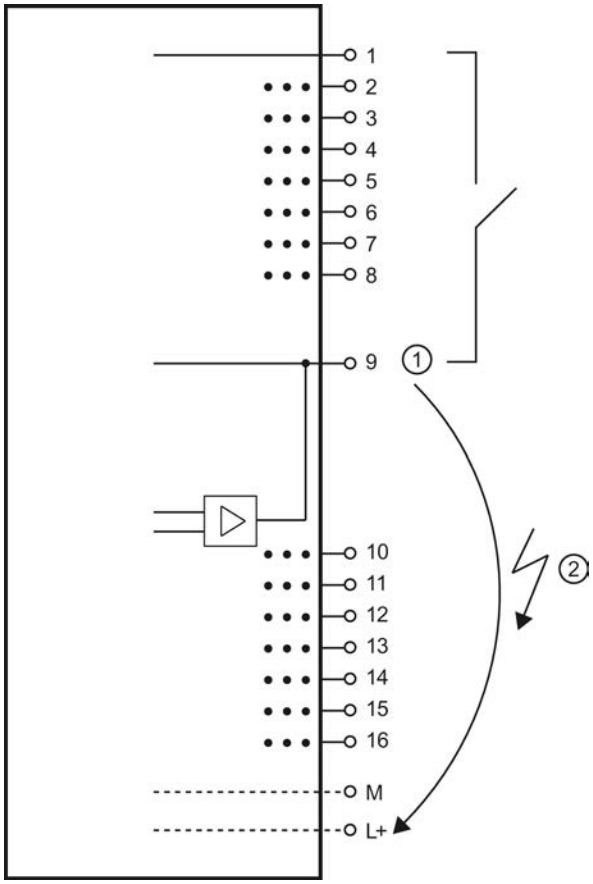
Habilitación del diagnóstico cuando aparece un cortocircuito a masa de la alimentación del sensor.



① Alimentación del sensor
② Cortocircuito

Diagnóstico Cortocircuito a L+

Habilitación del diagnóstico cuando aparece un cortocircuito a L+ de la alimentación del sensor.



① Alimentación del sensor
② Cortocircuito

Diagnóstico Rotura de hilo

Habilitación del diagnóstico si el cable que va al sensor está interrumpido.

Modo de operación

Determina si un canal está activado o desactivado.

Prolongación del impulso (solo para módulos High Feature)

La prolongación del impulso es una función que modifica una señal de entrada digital. Un impulso en una entrada digital se prolonga como mínimo hasta la longitud parametrizada. Si el impulso de entrada ya es más largo que la longitud parametrizada, no se modifica.

La prolongación del impulso se inicia siempre que el estado de la señal de entrada se modifica y si no hay activa ninguna prolongación del impulso para este canal.

Grupo de potencial del módulo izquierdo/nuevo grupo de potencial

Determina si el módulo de periferia se encuentra en una BaseUnit con entrada de tensión de alimentación (nuevo grupo de potencial) o en una BaseUnit sin entrada de tensión de alimentación (en este caso forma parte del grupo de potencial del módulo izquierdo).

Parámetros de los módulos de entradas analógicas

Falta tensión de alimentación L+

Habilitación del diagnóstico en caso de que falte tensión de alimentación L+ o sea insuficiente.

Unión fría (AI 4xRTD/TC 2-/3-/4-wire HF)

La unión fría puede ser una BaseUnit con sensor de temperatura interno (BU..T) o el canal 0 del módulo de periferia, siempre que este se haya parametrizado como "Termorresistencia Pt100 climatiz. Celsius".

A continuación se muestra una posible parametrización:

Tabla 8-118 Canal RTD

Ajuste	Descripción
Sin modo de canal de referencia	El valor de temperatura existente en el canal 0 puede utilizarse como valor de referencia para todo el módulo si los demás canales se configuran en correspondencia.
Canal de referencia del grupo 0	El canal actúa de emisor de la temperatura de la unión fría del grupo 0. La distribución se realiza a través del módulo de interfaz.

Tabla 8-119 Canal TC

Ajuste	Descripción
Canal de referencia del módulo	El canal TC correspondiente utiliza el canal 0 del mismo módulo como temperatura de la unión fría. Este debe estar parametrizado como "Termorresistencia Pt 100 climatiz. Celsius" y "Sin modo de canal de referencia", pues de lo contrario se dispara el diagnóstico Unión fría.
Unión fría interna	La temperatura de la unión fría se lee de un sensor de temperatura interno en la BaseUnit. En caso de que el tipo de BaseUnit sea incorrecto, se dispara el diagnóstico Unión fría.
Canal de referencia del grupo 0	Con el ajuste "TC" (termopar...), el canal actúa de receptor de la temperatura de la unión fría del grupo 0.
Temperatura de referencia fija	No se compensa la temperatura. La linealización se consigue con una temperatura supuesta de la unión fría de 0 °C.

Rebase por exceso

Habilitación del diagnóstico si el valor medido sobrepasa el rango de rebase por exceso.

Rebase por defecto

Habilitación del diagnóstico si el valor medido es inferior al rango de rebase por defecto.

Rotura de hilo

Habilitación del diagnóstico cuando el módulo no tiene flujo de corriente o o cuando la corriente es insuficiente para la medición en la entrada parametrizada en correspondencia.

Filtrado

Los diferentes valores medidos se alisan mediante filtrado digital. El filtrado puede ajustarse en 4 niveles, de modo que el factor de filtrado k multiplicado por el tiempo de ciclo del módulo de periferia equivale a la constante de tiempo del filtro. Cuanto mayor sea el filtrado mayor será la constante de tiempo del filtro.

La figura siguiente muestra la respuesta indicial con los diferentes factores de filtrado en función del número de ciclos del módulo.

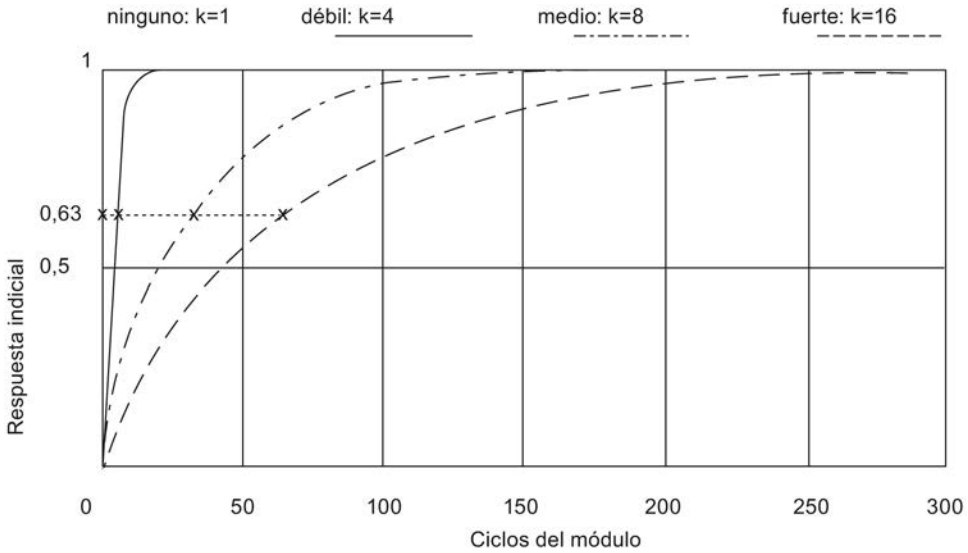


Figura 8-57 Alistamiento con AI 4×RTD/TC 2-/3-/4-wire HF

Supresión de frecuencias perturbadoras

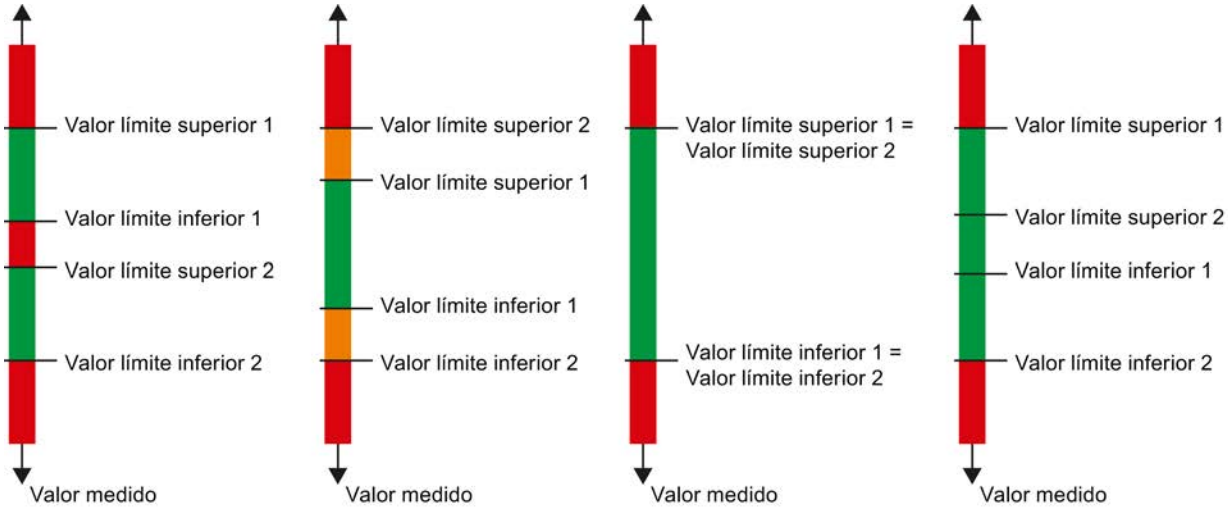
En los módulos de entradas analógicas, suprime las perturbaciones provocadas por la frecuencia de la red de tensión alterna utilizada.

La frecuencia de la red de tensión alterna puede provocar interferencias en el valor medido, especialmente al medir en rangos de pequeña tensión y con termopares. Con este parámetro, el usuario indica la frecuencia de red existente en la instalación.

Valores límite de la alarma de proceso

Si se rebasa el límite superior 1/2 por encima o el límite inferior 1/2 por debajo, el módulo dispara una alarma de proceso.

A continuación se muestran algunos ejemplos de selección de los valores límite 1 y 2.



Valor límite inferior 1/2

Especifique un umbral cuyo rebase por defecto dispare una alarma de proceso.

Valor límite superior 1/2

Especifique un umbral cuyo rebase por exceso dispare una alarma de proceso.

Grupo de potencial del módulo izquierdo/nuevo grupo de potencial

Determina si el módulo de periferia se encuentra en una BaseUnit con entrada de tensión de alimentación (nuevo grupo de potencial) o en una BaseUnit sin entrada de tensión de alimentación (en este caso forma parte del grupo de potencial del módulo izquierdo).

Coeficiente de temperatura (tipo de medición Termorresistencia)

El factor de corrección para el coeficiente de temperatura (valor α) indica el cambio relativo de la resistencia de un determinado material cuando la temperatura aumenta 1 °C.

El coeficiente de temperatura depende de la composición química del material. En Europa se utiliza un solo valor por tipo de sensor (valor predeterminado).

Los demás valores permiten ajustar el coeficiente de temperatura en función del sensor, con lo que se consigue una mayor precisión.

Consulte también

Particularidades del AI 4xRTD/TC 2/3/4 hilos HF (Página 2128)

Particularidades del AI 4xRTD/TC 2/3/4 hilos HF

Uso de sensores Cu10

- Seleccione en la parametrización "Termorresistencia 3 hilos" y "Cu10".
- Cablee el sensor Cu10 según el sistema de conexión a 3 hilos.
- Durante el funcionamiento se lleva a cabo una compensación interna automática de la resistencia del cable de medición ausente.

Nota

Para garantizar una compensación óptima del cable de Cu10, tenga en cuenta lo siguiente:

- La suma de la resistencia del cable y la resistencia de medición no debe exceder los 31 Ω.
- El cable debe presentar una resistencia máxima de 8 Ω si desea utilizar el rango de temperatura hasta los 312 °C.
  Ejemplo: un cable de Cu de 200 m de longitud con una sección de conductores de 0,5 mm^2 tiene aproximadamente 7 Ω. Una sección más pequeña reduce la longitud de cable permitida en correspondencia.

Uso de resistencias PTC

Los PTC son adecuados para vigilar la temperatura o como dispositivo de protección térmica de accionamientos complejos o arrollamientos de transformadores.

- Seleccione en la parametrización "Resistencia 2 hilos" y "PTC":
- Conecte el PTC según el sistema de conexión a 2 hilos.
- Utilice resistencias PTC del tipo A (termistor PTC) según DIN/VDE 0660, parte 302.
- Si el diagnóstico "Rebase por exceso/defecto" está habilitado, con valores de resistencia < 18 Ω se genera un diagnóstico "Rebase por defecto del límite inferior" que indica un cortocircuito.
- Datos del sensor relativos a la resistencia PTC:

Tabla 8-120 Uso de resistencias PTC

Propiedad	Datos técnicos	Observación
Puntos de conmutación	**Comportamiento al aumentar la temperatura**	
	< 550 Ω	**Rango normal:** • SIMATIC S7: bit 0 = "0", bit 2 = "0" (en PAE)
	de 550 Ω a 1650 Ω	**Rango de preaviso:** • SIMATIC S7: bit 0 = "0", bit 2 = "1" (en PAE)
	< 1650 Ω	**Rango de reacción:** • SIMATIC S7: bit 0 = "1", bit 2 = "0" (en PAE)
	Comportamiento al descender la temperatura	

Propiedad	Datos técnicos	Observación
	< 750 Ω	**Rango de reacción:** • SIMATIC S7: bit 0 = "1", bit 2 = "0" (en PAE)
	de 750 Ω a 540 Ω	**Rango de preaviso:** • SIMATIC S7: bit 0 = "0", bit 2 = "1" (en PAE)
	< 540 Ω	**Rango normal:** • SIMATIC S7: bit 0 = "0", bit 2 = "0" (en PAE)
(TNF-5) °C (TNF+5) °C (TNF+15) °C Tensión de medición Tensión en PTC	máx. 550 Ω mín. 1330 Ω mín. 4000 Ω máx. 7,5V	TNF= temperatura nominal de reacción

- Ocupación de la imagen de proceso de las entradas (PAE) en SIMATIC S7

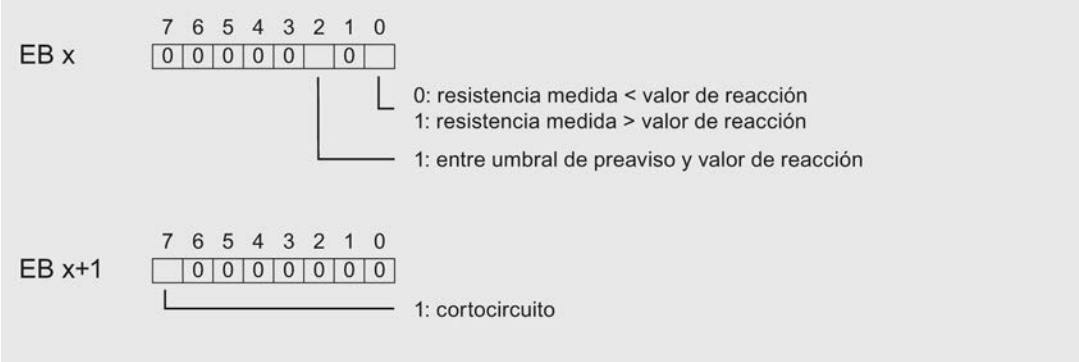


Figura 8-58 Ocupación de la memoria imagen de proceso de las entradas (PAE)

- Indicaciones de programación

Nota

En la imagen de proceso de las entradas solo son relevantes para la evaluación los bits 0+2. Los bits 0+2 permiten vigilar la temperatura de un motor, por ejemplo.

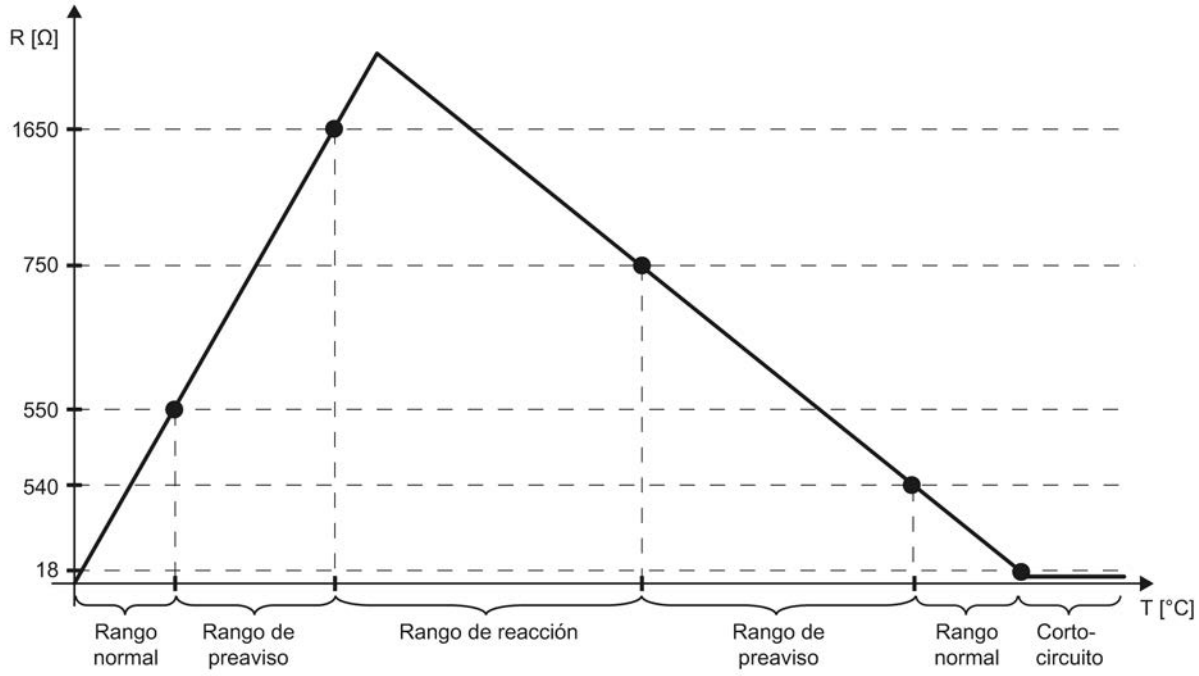
Los bits 0+2 de la imagen de proceso de las entradas no tienen memoria. Durante la parametrización, asegúrese de que un motor, por ejemplo, arranque de forma controlada (mediante un acuse).

Los bits 0+2 no pueden estar activados al mismo tiempo, sino que se activan sucesivamente.

Por motivos de seguridad, evalúe siempre las entradas de diagnóstico del AI 4×RTD/TC 2-/3-/4-wire HF, pues en caso de extracción de módulos de periferia, fallo en la tensión de alimentación del módulo de periferia, rotura de hilo o cortocircuito de los cables de medición no es posible medir.

Ejemplo

El diagrama inferior muestra la evolución de la temperatura y los correspondientes puntos de conmutación.



Consulte también

Parámetros de los módulos de entradas analógicas (Página 2125)

ET 200MP

Sistema de periferia descentralizada ET 200MP

Definición

El ET 200MP es un sistema de periferia descentralizada escalable y flexible que permite conectar las señales del proceso a un controlador central a través de un bus de campo.

Campo de aplicación

El ET 200MP es un sistema de periferia descentralizada multifuncional para diferentes campos de aplicación. Gracias a su estructura escalable es posible adaptar la configuración exactamente a las exigencias del proceso.

El ET 200MP está homologado para el grado de protección IP 20 y su montaje está previsto en un armario de distribución.

Estructura

El ET 200MP se monta sobre un perfil soporte y se compone de:

- un módulo de interfaz que se comunica con todos los controladores IO que se comportan conforme a la norma PROFINET IEC 61158,
- Pueden conectarse hasta 30 módulos (de alimentación y de periferia, de la gama de periferia de la S7-1500) a la derecha, junto al módulo de interfaz.
- Si se inserta un módulo de alimentación a la izquierda del módulo de interfaz, la configuración máxima posible es de 32 módulos.
- El número de módulos de periferia enchufables está limitado por su consumo eléctrico.

Reglas para slots

- Slot 0: módulo de alimentación (opcional)
- Slot 1: módulo de interfaz
- Slots 2 a 31: módulos de periferia o de alimentación

Parámetro Módulo de interfaz

Tensión de alimentación L+ conectada

Parámetro "Tensión de alimentación L+ conectada"

Este parámetro influye en el diagnóstico y en la comprobación del equilibrio de potencia.

- Diagnóstico del ET 200MP:
  Si la configuración real no se corresponde con la configuración teórica con respecto a la tensión de alimentación del módulo de interfaz, el módulo de interfaz genera un aviso de diagnóstico. Ejemplo: se ha desactivado la opción "Tensión de alimentación L+ conectada", pero en la instalación real se ha conectado la tensión de alimentación.
- Comprobación del equilibrio de potencia durante la configuración:
  En función del ajuste del parámetro se modifica el equilibrio de potencia: o bien el módulo de interfaz suministra potencia al bus de fondo o se alimenta a partir del bus de fondo.

El ajuste predeterminado (opción "Tensión de alimentación L+ conectada" está **activada**) significa que el módulo de interfaz se alimenta en el frente con 24 V DC y que suministra potencia al bus de fondo.

Si la opción "Tensión de alimentación L+ conectada" está **desactivada**, el módulo de interfaz no debe ser alimentado en el frente con 24 V DC.

En ese caso debe insertarse a la izquierda del módulo de interfaz una fuente de alimentación del sistema (PS) que alimente al módulo de interfaz y a los módulos a la derecha del módulo de interfaz.

Nota

Se recomienda alimentar el módulo de interfaz siempre desde el frente con 24 V DC. Si adicionalmente se inserta y conecta una fuente de alimentación del sistema (PS) **delante** o a la izquierda del módulo de interfaz, estarán disponibles para la instalación tanto la potencia de la fuente de alimentación del sistema (PS) como la potencia de la alimentación integrada del módulo de interfaz.

No es necesario cambiar el ajuste predeterminado del parámetro.

Control de la configuración

Principio de funcionamiento

Gracias al control de la configuración es posible utilizar diferentes configuraciones (opciones) reales con una sola configuración del sistema de periferia descentralizada ET 200MP.

El control de la configuración permite configurar el sistema de periferia descentralizada ET 200MP con su configuración máxima y utilizarlo aunque falten módulos. Si posteriormente se añaden algunos módulos, no es necesario configurar de nuevo y, por consiguiente, tampoco volver a cargar la configuración hardware.

Utilizando registros de control que se transmiten al módulo de interfaz desde el programa de usuario, se define una configuración teórica actual.

Se distingue entre los siguientes métodos:

- Realizar el control de la configuración sin slots vacíos (configuración futura)
- Ampliar la configuración (puesta en servicio paulatina)

Requisitos

Active la opción "Control de configuración" del módulo de interfaz (área "Control de configuración"). El control se efectúa a través del registro de control 196 para determinar la configuración deseada. Sin registro de control, el control de configuración no está operativo. El registro de control se transfiere con la instrucción WRREC en el programa de usuario.

Direccionamiento del módulo de interfaz mediante identificador de hardware

Para transferir el registro 196 con la instrucción WRREC hay que introducir el identificador de hardware del módulo de interfaz como parámetro de entrada para la instrucción. La constante de sistema de dicho identificador de hardware es, p. ej., "IO-Device._V2 [Head]". Las constantes de sistema de un dispositivo seleccionado se muestran en la ficha "Constantes de sistema" de la vista de redes, por ejemplo. Utilice el valor correspondiente para el direccionamiento.

Información adicional y ejemplos

Encontrará información relacionada con las reglas del control de la configuración, la estructura del registro de control, el comportamiento durante el funcionamiento y ejemplos de la estructura del registro de control para diferentes configuraciones en el manual de producto del módulo de interfaz IM 155-5 PN ST (http://support.automation.siemens.com/WW/view/es/67295970).

Parámetro Módulos de entrada

Parámetros de los módulos de entradas analógicas

Falta tensión de alimentación L+

Habilitación del diagnóstico en caso de que falte tensión de alimentación L+ o sea insuficiente.

Rotura de hilo

Habilitación del diagnóstico si en la correspondiente entrada parametrizada del módulo no circula corriente o la corriente es insuficiente para la medición o se aproxima a una tensión demasiado baja.

Límite de intensidad para diagnóstico de rotura de hilo

Valor umbral con el que se señaliza rotura de hilo. Según el sensor utilizado, el valor puede ajustarse a 1,185 mA o 3,6 mA.

Rebase por exceso

Habilitación del diagnóstico si el valor medido rebase por exceso el margen de saturación.

Rebase por defecto

Habilitación del diagnóstico si el valor medido rebase por defecto el margen de saturación.

Error en modo común

Habilitación del diagnóstico si se supera la tensión permitida en modo común.

Error en canal de referencia (sólo con AI 8xU/I/RTD/TC ST)

- Habilitación del diagnóstico si se produce un error en el canal para la compensación de temperatura, p. ej., rotura de hilo.
- El tipo de compensación Temperatura de referencia dinámica está parametrizado y todavía no se transferido ninguna temperatura de referencia al módulo.

Coeficiente de temperatura

El coeficiente de temperatura depende de la composición química del material. En Europa se utiliza un solo valor por tipo de sensor (valor predeterminado).

El factor de corrección para el coeficiente de temperatura (valor α) indica cuánto varía de forma relativa la resistencia de un determinado material cuando la temperatura aumenta 1 °C.

Los demás valores permiten ajustar el coeficiente de temperatura en función del sensor, con lo que se consigue una mayor precisión.

Supresión de frecuencias perturbadoras

En los módulos de entradas analógicas, suprime las perturbaciones causadas por la frecuencia de la red de corriente alterna utilizada.

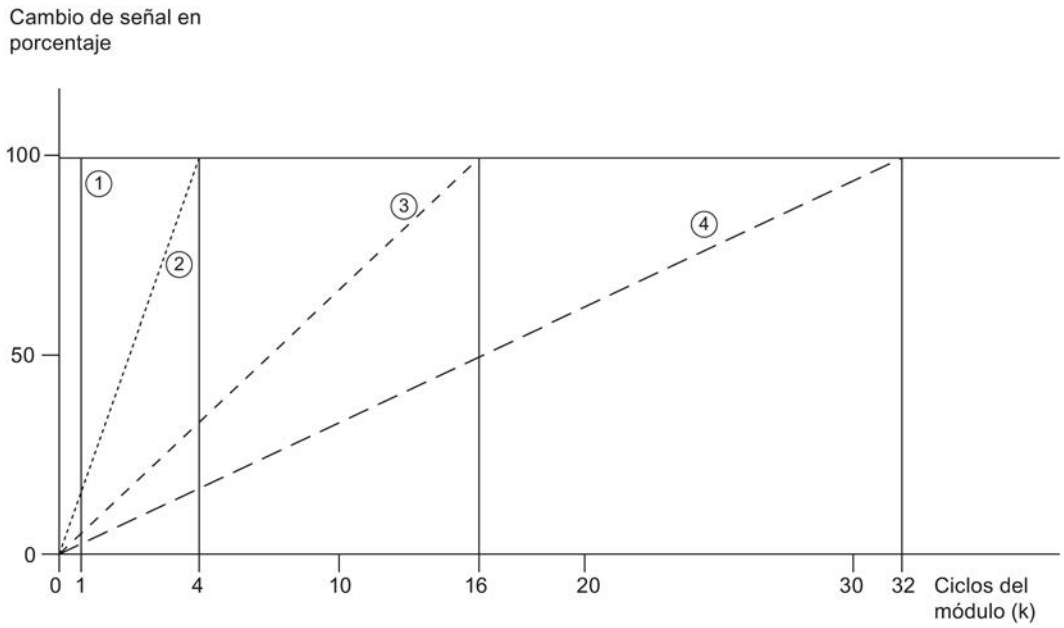
La frecuencia de la red de tensión alterna puede provocar interferencias en el valor medido, especialmente al medir en rangos de pequeña tensión y con termopares. El usuario indica mediante este parámetro la frecuencia de red que predomina en su instalación.

Filtrado

Los valores medidos son suavizados mediante filtrado. El alisamiento se puede ajustar en 4 niveles en los módulos de entradas analógicas AI 8xU/I/RTD/TC ST y AI 8xU/I HS.

Tiempo de alisamiento = número de ciclos del módulo (k) x tiempo de ciclo del módulo.

La figura siguiente muestra tras cuántos ciclos de módulo el valor analógico alisado se aproxima al 100%, dependiendo del alisamiento ajustado. La figura se aplica a cualquier cambio de señal en la entrada analógica.



① Ninguno (k = 1)
② Débil (k = 4)
③ Medio (k = 16)
④ Fuerte (k = 32)

Unión fría (sólo con AI 8xU/I/RTD/TC ST)

Para el parámetro Unión fría se pueden parametrizar los siguientes ajustes:

Tabla 8-121 Posibles parametrizaciones para el parámetro Unión fría

Ajuste	Descripción
Temperatura de referencia fija	La temperatura de la unión fría se parametriza y se almacena como valor fijo en el módulo.
Temperatura de referencia dinámica	La temperatura de la unión fría se transfiere en el programa de usuario de la CPU al módulo con la instrucción WRREC (SFB 53) mediante un registro comprendido entre 192 y 199.
Unión fría interna	La temperatura de la unión fría se determina con un sensor integrado en el módulo.
Canal de referencia del módulo	La temperatura de la unión fría se determina con una termorresistencia externa (RTD) en el canal de referencia del módulo (COMP).

Nota

Temperatura de referencia fija

En la parametrización de un termopar de tipo B, solo es posible el ajuste "Temperatura de referencia fija" con una temperatura de 0 °C.

Habilitación de alarma de proceso 1 ó 2

Habilitación de una alarma de proceso si se rebasa por exceso el límite superior 1 ó 2 o se rebasa por defecto el límite inferior 1 ó 2.

Límite inferior 1 ó 2

Especifique el umbral por debajo del cual se dispara la alarma de proceso 1 ó 2.

Límite superior 1 ó 2

Especifique el umbral por encima del cual se dispara la alarma de proceso 1 ó 2.

Compensación de temperaturas en termopares

Introducción

Existen varias posibilidades de registrar la temperatura de la unión fría para obtener un valor absoluto de temperatura basado en la diferencia de temperatura entre el punto de referencia y el de medición.

Según la situación (física) donde se requiera la unión fría, es posible aprovechar las distintas posibilidades de compensación.

Nota

En la parametrización de un termopar de tipo B, solo es posible el ajuste "Temperatura de referencia fija" con una temperatura de 0 °C.

Posibilidades para compensar la temperatura en la unión fría

Posibilidades de compensación	Explicación	Caso de aplicación
Unión fría interna	En esta compensación la temperatura de la unión fría se determina con un sensor integrado en el módulo. **Procedimiento** Conecte el termopar al módulo de periferia directamente o con cables de compensación.	• Para conectar los cables de compensación utilice el mismo material que el del termopar. • Si en la instalación la temperatura de la unión fría y la del módulo son iguales, también pueden emplearse cables de materiales distintos.
Canal de referencia del módulo	La temperatura de la unión fría se determina con una termorresistencia externa (RTD). **Procedimiento** Conecte el termopar, directamente o con cables de compensación, a la unión fría con los cables de entrada. Conecte los cables de entrada a los bornes correspondientes del módulo. Conecte la termorresistencia (RTD) al canal de referencia del módulo. La termorresistencia (RTD) debe estar ubicada en el área de la unión fría.	• La temperatura puede registrarse directamente en la unión fría. • La temperatura medida de todos los canales configurados para este tipo de compensación se corrige automáticamente con el valor de temperatura de la unión fría. • Desde la unión fría hasta el módulo pueden emplearse cables económicos (p. ej. de cobre).
Temperatura de referencia dinámica	La temperatura de la unión fría se determina a través de un módulo. Transfiera el valor de temperatura a otros módulos mediante un registro en el programa de usuario. **Procedimiento** Conecte la termorresistencia (RTD) para la unión fría a un canal cualquiera. La temperatura de la unión fría se transfiere de la CPU al módulo con la instrucción WRREC mediante un registro comprendido entre 192 y 199.	• Utilizando varios módulos en la unión fría podrá compensar todos los canales mediante un valor de temperatura común. • Para registrar el valor de temperatura basta con una termorresistencia (RTD). • Desde la unión fría hasta el módulo pueden emplearse cables económicos (p. ej. de cobre).
Temperatura de referencia fija	La temperatura de la unión fría se parametriza como valor fijo en el módulo. **Procedimiento** Conecte el termopar, directamente o con cables de compensación, a la unión fría con los cables de entrada. Conecte los cables de entrada a los bornes correspondientes del módulo. Al configurar el módulo predefina un valor de temperatura fijo para la unión fría (p. ej. 20 ºC).	• Mantenga constante la temperatura de la unión fría e identifique el valor de temperatura. • Desde la unión fría hasta el módulo pueden emplearse cables económicos (p. ej. de cobre).

Parámetro Módulos de salida

Parámetros de los módulos de salidas analógicas

Falta tensión de alimentación L+

Habilitación del diagnóstico en caso de que falte tensión de alimentación L+ o sea insuficiente.

Cortocircuito a masa

Habilitación del diagnóstico si aparece un cortocircuito a masa de la alimentación del actuador.

Rotura de hilo

Habilitación del diagnóstico si el cable que va al sensor está interrumpido.

Rebase por exceso

Habilitación del diagnóstico si el valor medido sobrepasa el rango de rebase por exceso.

Rebase por defecto

Habilitación del diagnóstico si el valor medido es inferior al rango de rebase por defecto.

Reacción a STOP de la CPU

Determina el comportamiento de la salida cuando la CPU pasa al estado operativo STOP.

Valor sustitutivo

Los valores sustitutivos son los que entregan las salidas (la salida) en caso de STOP de la CPU.

ET 200M

Configuración del ET 200M

Introducción

Para la familia ET 200M encontrará una amplia gama de módulos en el catálogo de hardware bajo "Periferia descentralizada".

Configuración y parametrización

Encontrará información sobre la configuración y parametrización en los capítulos siguientes.

Estructura del ET 200M

Definición

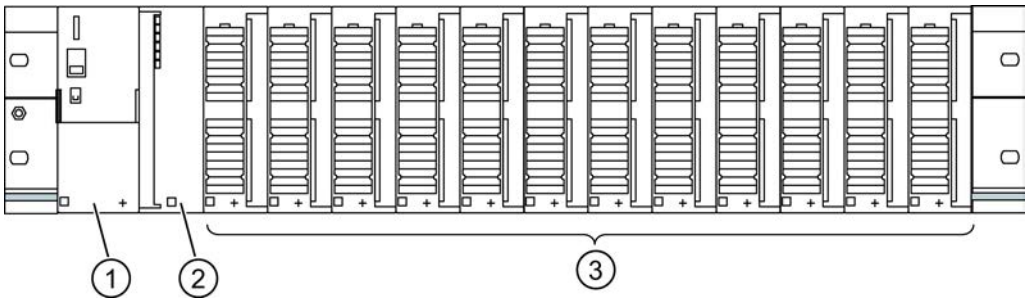
El sistema de periferia descentralizada ET 200M es un dispositivo de periferia modular con grado de protección IP 20.

El ET 200M presenta la técnica de montaje del sistema de automatización S7-300 y se compone del IM 153-x y módulos periféricos de la familia S7-300.

El ET 200M se puede comunicar con:

- todos los maestros DP que se comportan conforme a la norma IEC 61784-1:2002 Ed1 CP 3/1
- todos los controladores IO que se comportan conforme a la norma IEC 61158

Configuración de un ET 200M (ejemplo)



① Fuente de alimentación PS 307
② Módulo de interfaz IM 153-x
③ hasta 12 módulos periféricos (SM / FM / CP)

Configurar la función 'Sustitución de módulos en marcha'

Introducción

El ET 200M soporta la función "Sustitución de módulos en marcha" y la correspondiente alarma de extracción/inserción.

La función "Sustitución de módulos en marcha" permite extraer o insertar módulos del rack del ET 200M durante el funcionamiento.

Requisitos

Se ha configurado un módulo de interfaz que soporta la sustitución de módulos en marcha. (a partir de IM 153-1, ref. 153-1AA02-0XB0).

Además, la CPU configurada también debe soportar esta función, p. ej. para PROFIBUS una S7-400 con interfaz DP.

Para la configuración hardware debe utilizarse el bus de fondo activo (barra de bus con slots). El perfil soporte tradicional con conectores de bus entre los módulos no soporta esta función.

Configuración

Si se cumplen los requisitos para la configuración, en la ventana de inspección se ofrece el parámetro "Sustitución de módulos en marcha" en el área "Parámetros del módulo", el cual puede activarse. Para los módulos configurados se mostrará debajo de dicho parámetro una tabla que muestra los módulos de bus activos que se necesitan para la configuración hardware.

Para una configuración de PROFIBUS se muestra además la opción "Arranque si la configuración real difiere de la teórica". Dicha opción se activa automáticamente si está activada la "Sustitución de módulos en marcha".

Configuración de variables HART

Introducción

Un gran número de aparatos de campo HART ponen a disposición magnitudes de medida adicionales (p. ej. temperatura del sensor). Dichas magnitudes pueden leerse si están bien ajustadas en la configuración de los aparatos de campo. Por medio de variables HART es posible aplicar directamente los valores medidos ajustados del aparato de campo al área E/S del sistema de automatización.

En los módulos HART pueden parametrizarse un máximo de 8 variables HART (independientemente del número de canales configurados), 4 variables HART como máximo por canal. Las variables HART se asignan a un canal en las propiedades del módulo (área "Ajuste de variable HART").

Requisitos

El módulo HART está insertado en un ET 200M (a partir de IM 153-2, 6ES7 153-2BA02-0AB0).

Asignación de direcciones

El módulo HART ocupa 16 bytes de entrada/salida. Al configurar variables HART, el módulo ocupa 5 bytes adicionales para cada variable HART.

Si se utilizan las 8 variables HART, el módulo de entrada HART ocupa en total 56 bytes de entrada/salida (16 bytes + 8 x 5 bytes = 56 bytes).

La configuración "ninguno" no ocupa ningún byte de entrada adicional.

Configuración de variables HART

Para un canal es posible configurar un máximo de 4 variables HART

- PV (Primary Variable)
- SV (Secundary Variable)
- TV (Teritary Variable)
- QV (Quatenary)

CiR es un comodín y reserva el área de direcciones para una variable HART. Las variables HART que no se utilizan deben configurarse con el parámetro "ninguno".

Estructura de las variables HART

Las variables HART están estructuradas del siguiente modo:

Datos HART de 4 bytes	QC de 1 byte

Estructura del byte "Quality Code"

El Quality Code (QC) puede adoptar los valores siguientes:

Quality Code (QC)	Significado
0x4C o 0	Inicialización: valor 0 del IM y 4C del módulo
0x18	Interrupción de la comunicación / ninguna comunicación
0x0C	Error en el dispositivo HART
0x47	El dispositivo HART está ocupado
0x84	OK "Configuración modificada"
0x80	OK

Consulte también

Documentación de los módulos analógicos HART (http://support.automation.siemens.com/WW/view/ed/22063748)

Módulos de señales para automatización de procesos

Principios básicos

Introducción

Los módulos de señales para la automatización de procesos son módulos del S7-300, como el SM 321; DI 16xNAMUR o SM 322; DO 16xDC24V/0,5A.

Se utilizan en un esclavo DP (IM 153-2).

En comparación con los módulos estándar, ofrecen las siguientes funciones tecnológicas adicionales, por ejemplo prolongación del impulso y vigilancia de inestabilidad de señal.

Consulte también

Inversor (Página 2142)

Parámetros tecnológicos (Página 2143)

Inversor

Tipo de encoder "inversor"

Si las entradas digitales de un grupo de canales están parametrizadas como "inversores", el módulo realiza un diagnóstico para el tipo de encoder inversor de este grupo de canales.

Inversor

Un inversor es un bloque de contactos auxiliares con un solo contacto móvil que tiene una posición cerrada con el aparato de maniobra cerrado y una con el aparato abierto.

Tenga en cuenta la siguiente regla:

- Conecte siempre el contacto normalmente abierto al canal "par".
- Conecte siempre el contacto normalmente cerrado al canal "impar".

El tiempo de conmutación tolerado entre los dos canales es de 300 ms.

Si la verificación es negativa,

- El módulo identifica el estado de valor del canal del contacto normalmente abierto como "no válido"
- El módulo genera una entrada de diagnóstico para el canal del contacto normalmente abierto
- Se lanza una alarma de diagnóstico (si está habilitada)

La señal de entrada digital y el estado de valor se actualizan sólo para el canal del contacto normalmente abierto. Para el canal del contacto normalmente cerrado, la señal de entrada digital es "cero" y el estado de valor "no válido", ya que dicho canal sólo sirve para comprobar el sensor.

El diagnóstico depende del parámetro "Selección" (del encoder). Observe también las particularidades en el diagnóstico con el tipo de encoder inversor en el manual "Signal Modules for Process Automation".

Consulte también

Documentación de módulos para la automatización de procesos (http://support.automation.siemens.com/WW/view/de/7215812/0/en)

Parámetros tecnológicos

Prolongación del impulso y vigilancia de inestabilidad de señal

La prolongación del impulso es una función que modifica una señal de entrada digital. Un impulso en una entrada digital se prolonga como mínimo hasta la longitud parametrizada. Si el impulso de entrada ya es más largo que la longitud parametrizada, no se modifica.

Si los impulsos deben modificarse, haga clic en el campo para seleccionar el tiempo. Si los impulsos no deben prolongarse, seleccione la entrada "---".

La vigilancia de inestabilidad de señal es una función del sistema de control para señales de entradas digitales. Detecta y notifica cursos anómalos de las señales en la ingeniería de procesos, p. ej. oscilaciones demasiado frecuentes de la señal de entrada entre "0" y "1".

Para activar la vigilancia de inestabilidad de señal es imprescindible que el diagnóstico de grupo también esté habilitado en esta entrada.

Ventana de vigilancia y número de cambios de señal

El funcionamiento de la vigilancia de inestabilidad de señal se determina mediante los parámetros Ventana de vigilancia y Número de cambios de señal.

La primera vez que cambia la señal de entrada se inicia el tiempo que se ha parametrizado como ventana de vigilancia. Si, durante este tiempo, la señal de entrada cambia con más frecuencia de la parametrizada en el número de cambios de señal, se identifica como error de inestabilidad de señal. Si dentro de la ventana de vigilancia no se detecta ningún error de inestabilidad de señal, cuando vuelva a cambiar la señal se reiniciará la ventana de vigilancia.

Nota

Si se parametriza una prolongación del impulso en un canal de entrada, esto afectará a la vigilancia de inestabilidad de señal habilitada para dicho canal. La señal cuyo impulso se ha prolongado es la señal de entrada para la vigilancia de inestabilidad de señal. Por consiguiente, sincronice las parametrizaciones de la prolongación del impulso y la vigilancia de inestabilidad de señal.

Consulte también

Documentación de módulos para la automatización de procesos (http://support.automation.siemens.com/WW/view/de/7215812/0/en)

Módulo IQ-Sense

Propiedades de 8 IQ-SENSE

Propiedades

El módulo 8 IQ-SENSE se caracteriza por las siguientes particularidades:

- Conexión de sensores con IQ-SENSE®, detectores fotoeléctricos de proximidad: p.ej. barreras fotoeléctricas por reflexión, sensores difusos y sensores láser.
- Apto para el uso centralizado en S7-300 o descentralizado en el ET 200M.
- A cada módulo pueden conectarse hasta 8 sensores. Se requiere un cable de dos hilos por cada sensor.
- Reserva de función parametrizable.
- Funciones de tiempo parametrizables, histéresis de conmutación y modo síncrono.
- Prescripciones de valores de sensibilidad y distancia (*IntelliTeach* a través de FB "IQ-SENSE Opto")
- Teach in
- Extracción e inserción de sensores durante el funcionamiento (reparametrización automática)

Grupo de antiinterferencias

Solo para dispositivos IQ-Sense ópticos (ID de perfil IQ 1).

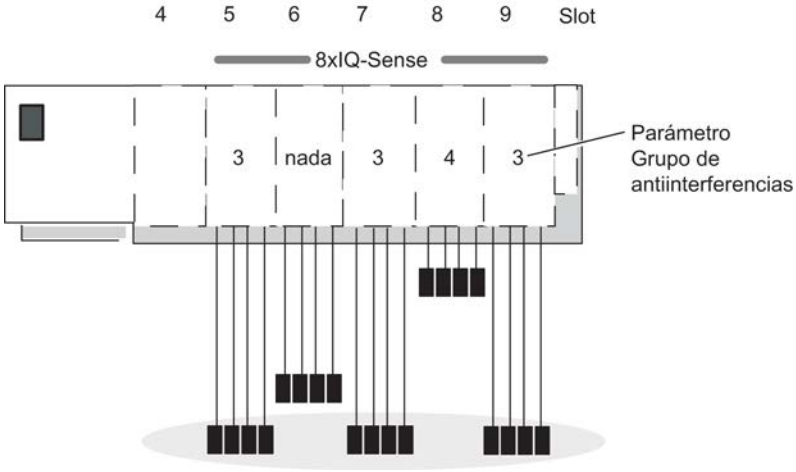
Para dispositivos IQ-Sense con ID de perfil IQ 128 (ultrasonido) consulte los parámetros por canal "Modo multiplex/síncrono".

Impedir una influencia perturbadora (interferencia, p. ej. luz parásita) asignando un grupo de antiinterferencias. Esto significa que:

- Grupo de antiinterferencias: ninguno (= ajuste predeterminado)
  En caso de disposición desfavorable, es posible que los sensores ópticos de uno o varios módulos se influyan mutuamente.
- Grupo de antiinterferencias: 3 o 4
  Los sensores ópticos del mismo módulo con el grupo de antiinterferencias 3 o 4 no pueden influirse mutuamente. Los sensores ópticos de diferentes módulos con el grupo de antiinterferencias 3 o 4 tampoco pueden influirse mutuamente. No es necesario mantener una distancia mínima entre los dispositivos IQ-Sense ópticos, por lo que dos barreras fotoeléctricas por reflexión pueden alinearse a un reflector común, por ejemplo.

Principio de funcionamiento

La figura siguiente ilustra el funcionamiento del parámetro Grupo de antiinterferencias:



Una interferencia parásita solo es posible entre los sensores ópticos de los módulos en los slots 5, 6, 7 y 9, ya que estos se encuentran en el mismo grupo de antiinterferencias 3 o bien están ajustados con "ninguno".

Nota

Los sensores que pertenecen a un mismo grupo de antiinterferencias deben montarse de manera que se respete la distancia mínima (véase la hoja adjunta al sensor) y así se puedan excluir interferencias entre dichos sensores.

Tipo de sensor

Con este parámetro se ajusta el tipo de sensor por canal:

- Barrera fotoeléctrica por reflexión o
- Sensor difuso o
- desactivado

Sensor difuso

Tabla 8-122 Sensor difuso

Sensor difuso	Objeto	
Emisor Receptor		Estado de conmutación 0: No se ha detectado ningún objeto, es decir, el objeto no se encuentra en la trayectoria de los rayos. El receptor no ve luz
Emisor Receptor		Estado de conmutación 1: Objeto detectado, es decir, el objeto se encuentra en la trayectoria de los rayos. El receptor no ve luz.

Barrera fotoeléctrica por reflexión

Tabla 8-123 Barrera fotoeléctrica por reflexión

Barrera fotoeléctrica por reflexión	Objeto	
Emisor Receptor		Estado de conmutación 0: No se ha detectado ningún objeto, es decir, el objeto no se encuentra en la trayectoria de los rayos. El receptor ve luz.
Emisor Receptor		Estado de conmutación 1: Objeto detectado, es decir, el objeto se encuentra en la trayectoria de los rayos. El receptor no ve luz.

Histéresis de conmutación

Un fallo en el sensor difuso o en el proceso de fabricación puede provocar una inestabilidad de la señal. En ese caso, el valor medido oscila alrededor del umbral de conmutación del 100% (objeto detectado – objeto no detectado). Esta fluctuación del umbral de conmutación se puede impedir con el parámetro Histéresis de conmutación. De ese modo se obtiene una señal de salida estable en el sensor.

Puede parametrizar los márgenes 5 / 10 / 20 / 50 % como histéresis de conmutación.

Requisitos

El parámetro Histéresis de conmutación sólo puede ajustarse para barreras fotoeléctricas por reflexión con supresión de fondo.

Principio de funcionamiento

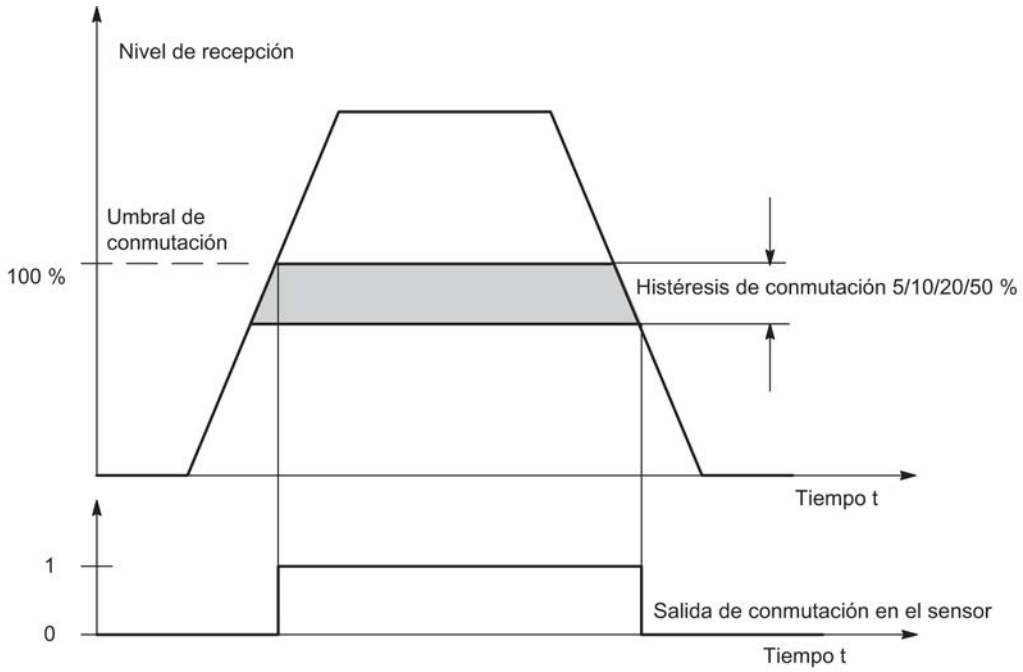


Figura 8-59 Parámetro Histéresis de conmutación

Funciones de tiempo, valor de tiempo

Con estos parámetros se puede adaptar el módulo electrónico a su aplicación.

Principio de funcionamiento

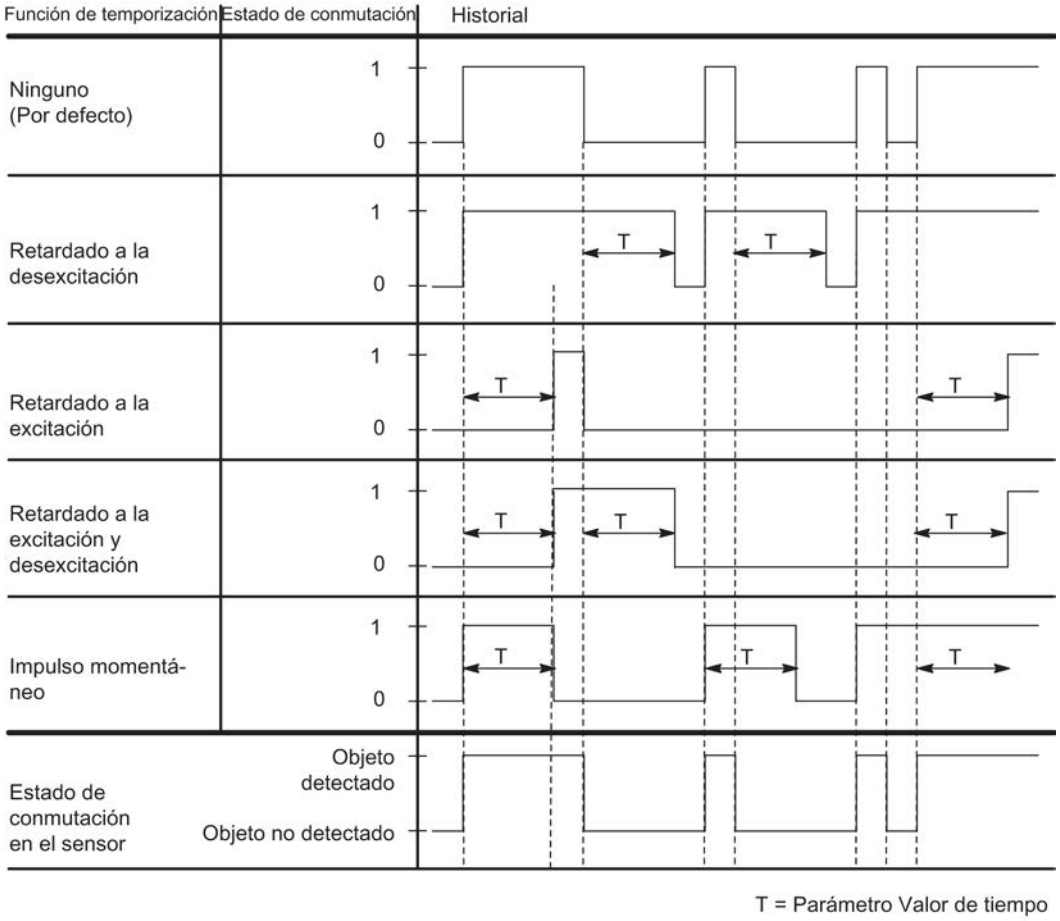


Figura 8-60 Parámetro Valor de tiempo, Funciones de tiempo

Modo multiplex/síncrono

Para evitar la influencia mutua de dispositivos ultrasonido IQ-Sense situados uno cerca del otro (dispositivos con ID de perfil IQ 128) se utiliza el parámetro "Modo multiplex/síncrono".

Ajustes del parámetro Modo multiplex/síncrono

Desactivado: es posible que los sensores ultrasonido IQ-Sense colindantes se influyan mutuamente (ajuste predeterminado). El tiempo de ciclo viene determinado por el sensor ultrasonido IQ-Sense.

Multiplex: los sensores ultrasonido IQ-Sense determinan el valor de proceso (distancia) uno detrás de otro, de modo que queda excluida una influencia mutua. El tiempo de ciclo es la suma de los tiempos de ciclo síncrono parametrizados para los sensores ultrasonido IQ-Sense en modo multiplex.

Sincronización: los sensores ultrasonido IQ-Sense determinan el valor de proceso (distancia) exactamente en el mismo instante, de modo que queda excluida una influencia mutua. El

tiempo de ciclo equivale al tiempo de ciclo síncrono más grande parametrizado para los sensores ultrasonido IQ-Sense en modo síncrono.

El modo síncrono se utiliza, p. ej., para una función de cortina en la que varios sensores ultrasonido situados en paralelo comparten un área de captura amplio. Los sensores emiten un impulso ultrasonido simultáneamente. Cuando un objeto entra en el área de captura, el sensor que está más cerca del objeto es el primero en recibir el eco. Esto permite no solo detectar el objeto sino además localizarlo.

Valor AFI

Con la ayuda del valor AFI (Application Family Identifier, definido en la norma internacional ISO 15693-3) se elige el transpondedor para diferentes aplicaciones. Solo se procesan los transpondedores cuyo valor AFI coincide con el valor ajustado en el sensor. Si un transpondedor tiene el valor AFI "0", es posible identificarlo y procesarlo independientemente del valor AFI del sensor.

Este parámetro solo es importante cuando es soportado por el sistema de identificación, de lo contrario puede tener un valor cualquiera (por lo general "0").

Tipo de transpondedor

Según sea el tipo de transpondedor, hay que ajustar si se trata de un transpondedor ISO o de un tipo específico del fabricante.

En los transpondedores que cumplen el estándar internacional ISO 15693 debe seleccionarse el valor "1"; para todos los demás tipos se ajusta "0". En base a este ajuste, en el sensor se selecciona uno de los dos drivers de interfaz aérea posibles.

Este parámetro solo es importante cuando es soportado por el sistema de identificación, de lo contrario puede tener un valor cualquiera (por lo general "0").

ET 200S

Configurar ET 200S

Introducción

Para la familia ET 200S encontrará una amplia gama de módulos en el catálogo de hardware bajo "Periferia descentralizada".

Parametrización

Encontrará información sobre la configuración y parametrización en "Consulte también".

Consulte también

Configurar ET 200S (Página 1404)

Convertidores de frecuencia

Empleo del convertidor de frecuencia

Convertidores de frecuencia

Los convertidores de frecuencia ICU24 e ICU24F (en su versión fail safe) tienen una estructura modular y están completamente incrustados en el sistema de periferia descentralizada ET 200S. A continuación se explica lo que debe tenerse en cuenta al parametrizar estos dos módulos.

Telegrama

El número de telegrama y el modo de operación del módulo sólo se visualizan y no pueden modificarse.

ID de aplicación

La ID de aplicación identifica la totalidad de los parámetros guardados en el convertidor de frecuencia. Introduzca una ID de aplicación comprendida en el rango de valores de 0 a 65535. Durante el arranque (o al extraer/insertar), esta identificación se compara con la ID de aplicación guardada en el convertidor.

Los convertidores que procesan aplicaciones idénticas suelen tener también la misma parametrización, por lo que deberían identificarse con la misma ID de aplicación. Los convertidores con la misma ID de aplicación pueden intercambiarse. Del mismo modo, sólo está permitido copiar la parametrización completa de un convertidor a otro, por ejemplo mediante una MMC, si ambos tienen la misma ID de aplicación.

Los convertidores que procesan aplicaciones diferentes y tienen parametrizaciones distintas deben identificarse con IDs de aplicación también distintas. De este modo se impide que un convertidor con una parametrización inadecuada arranque en un slot erróneo, es decir, en la aplicación equivocada. También se evita que la parametrización guardada en el convertidor se sobrescriba casualmente con otra parametrización guardada en una MMC.

Habilitar alarma de diagnóstico

Existe la posibilidad de habilitar la alarma de diagnóstico para los convertidores de frecuencia. Si la alarma de diagnóstico está habilitada, en la CPU debe haber un OB 82 que procese los eventos de diagnóstico.

Consulte también

Documentación del convertidor de frecuencia (http://support.automation.siemens.com/WW/view/es/26291825/0/en)

ET 200pro

Empleo del convertidor de frecuencia

Convertidores de frecuencia

Los convertidores de frecuencia ET 200pro FC y ET 200pro F-FC (en su versión fail safe) tienen una estructura modular y están completamente incrustados en el sistema de periferia descentralizada ET 200pro. A continuación se explica lo que debe tenerse en cuenta al parametrizar estos dos módulos.

Telegrama

El número de telegrama y el modo de operación del módulo solo se visualizan y no pueden modificarse.

ID de aplicación

La ID de aplicación identifica la totalidad de los parámetros guardados en el convertidor de frecuencia. Introduzca una ID de aplicación comprendida en el rango de valores de 0 a 65535. Durante el arranque (o al extraer/insertar), esta identificación se compara con la ID de aplicación guardada en el convertidor.

Los convertidores que procesan aplicaciones idénticas suelen tener también la misma parametrización, por lo que deberían identificarse con la misma ID de aplicación. Los convertidores con la misma ID de aplicación pueden intercambiarse. Del mismo modo, solo está permitido copiar la parametrización completa de un convertidor a otro, por ejemplo mediante una MMC, si ambos tienen la misma ID de aplicación.

Los convertidores que procesan aplicaciones diferentes y tienen parametrizaciones distintas deben identificarse con IDs de aplicación también distintas. De este modo se impide que un convertidor con una parametrización inadecuada arranque en un slot erróneo, es decir, en la aplicación equivocada. También se evita que la parametrización guardada en el convertidor se sobrescriba casualmente con otra parametrización guardada en una MMC.

Habilitar alarma de diagnóstico

Existe la posibilidad de habilitar la alarma de diagnóstico para los convertidores de frecuencia. Si la alarma de diagnóstico está habilitada, en la CPU debe haber un OB 82 que procese los eventos de diagnóstico.

8.1.5.7 Configuración IPv6

Protocolo IPv6

El protocolo de Internet en su versión 6 (en adelante abreviado: IPv6) amplía el protocolo de Internet en su versión 4 (IPv4), que es el más utilizado actualmente.

Formato de dirección IPv6: Notación

Las direcciones IPv6 constan de 8 bloques con cifras decimales de cuatro dígitos en cada uno (128 bits en total). Los bloques están separados por dos puntos.

Ejemplo:

fd00:ffff:ffff:ffff:ffff:ffff:2f33:8f21

Reglas / simplificaciones:

- Está permitido omitir los ceros a la izquierda dentro de un bloque.
  Ejemplo: fd01:0:ffff::2d12:7d23
- Si uno o varios bloques tienen el valor 0 se admite una notación abreviada.
  La dirección fd00:0:0:0:0:0:0:8f21 también puede anotarse abreviada, del siguiente modo:
  fd00::8f21
  Con el fin de conservar la univocidad, dicha abreviación solo puede aplicarse una vez dentro de toda la dirección.
- Notación decimal con puntos
  Para los 2 últimos bloques o 4 bytes es posible utilizar la notación decimal tradicional con puntos.
  Ejemplo: la dirección IPv6 fd00::ffff.125.1.0.1 es equivalente a fd00::ffff:7d01:1

Entrada y representación

La entrada de direcciones IPv6 es posible en las notaciones descritas anteriormente. Las direcciones IPv6 se muestran siempre en la notación con la que se han introducido.

Consulte también

IPv6 en el CP 1543-1 (Página 2152)

IPv6 en el CP 1543-1

Uso de IPv6 en el CP 1543-1

El CP soporta el protocolo de Internet en su versión 4 (IPv4) para todos los servicios IP.

La indicación adicional de direcciones en formato IPv6 puede utilizarse en el CP para los siguientes servicios y casos de aplicación:

- FETCH/WRITE
  Accesos directos de escritura/lectura de estaciones PC, SIMATIC S5 o dispositivos ajenos
- Cliente FTP
  Accesos FTP desde la CPU S7-1500 a un servidor FTP con el bloque de programa FTP_CMD
- Servidor FTP
  Accesos FTP desde un cliente FTP a áreas de datos de la CPU S7-1500

- SNMP
  Consulta de datos por medio de objetos MIB según SNMP
- E-mail
  Transferencia de datos desde la CPU S7-1500 con el bloque de programa T_Mail

8.1.6 Mostrar avisos

8.1.6.1 Sinopsis del visor de avisos

La función "Visor de avisos" permite emitir avisos asíncronos de eventos de diagnóstico y de avisos de diagnóstico personalizados, así como avisos de instrucciones ALARM.

Además, desde el visor de avisos se puede iniciar el editor de avisos con el comando del menú contextual "Editar aviso" y crear avisos de diagnóstico definidos por el usuario.

Símbolos

La siguiente tabla muestra los símbolos y su modo de funcionamiento:

Símbolo	Función
Vista de fichero	Muestra los avisos existentes en el fichero.
Avisos activos	Muestra los avisos actualmente pendientes. Los avisos de acuse obligado se muestran en color azul.
Ignorar	Ignora los avisos cuando éstos aparecen. Estos avisos ni se muestran en la ventana ni se guardan en el fichero.
Acusar	Confirma el aviso seleccionado como leído. Los avisos pendientes de acuse se muestran en color azul.
Vaciar el fichero	Borra todos los avisos del fichero.
Exportar el fichero	Exporta el fichero de avisos actual a un archivo en formato xml.

8.1.6.2 Vista de fichero

En la vista de fichero se muestran y archivan los avisos por orden cronológico. El tamaño del fichero (entre 200 y 3000 avisos) se puede establecer con el comando de menú "Opciones > Configuración > Online y diagnóstico". Cuando se excede el tamaño ajustado del fichero, se borra el aviso más antiguo.

Los avisos de acuse obligado se muestran en color azul y pueden ser acusados con el comando del menú contextual "Acusar aviso(s)".

El fichero se actualiza constantemente sin necesidad de guardarlo expresamente.

8.1.6.3 Estructura de los avisos en la vista de fichero

En la vista de fichero se registran todos los eventos que se han producido en las CPUs seleccionadas. Para cada evento se crea una entrada nueva, que se muestra en la tabla como fila adicional.

Estructura de la tabla

Todos los atributos de los avisos se pueden representar en forma de columnas. Es posible mostrar u ocultar las columnas de manera individual y modificar el ancho y el orden de las columnas. Esta configuración se guarda al cerrar el proyecto.

Los avisos se pueden mostrar en una o varias filas. En el modo de visualización de una sola fila, sólo se muestra la primera fila de los avisos de varias filas.

Los avisos pueden ser tanto de acuse obligado (avisos de acuse obligado) como no de acuse obligado. Las alarmas de acuse obligado pendientes de acuse se destacan en color azul y se pueden acusar de modo contextual con el respectivo botón de la barra de herramientas o con el comando del menú contextual "Acusar aviso(s)".

8.1.6.4 Recibir avisos

Para ver los avisos, primero debe ajustar la recepción de los avisos de cada CPU.

Procedimiento

Para recibir avisos, proceda de la manera siguiente:

1. Haga doble clic en la carpeta "Online y diagnóstico" de la CPU correspondiente en el árbol del proyecto.
2. Haga clic en el grupo "Accesos online" de la navegación local.
3. Active la opción "Recibir avisos".

Nota

Si elige este procedimiento, los avisos no se recibirán hasta que no vuelva a establecer una conexión online con el dispositivo.

O bien:

1. Seleccione la CPU correspondiente en la vista de dispositivos, redes o vista topológica.
2. Elija el comando "Recibir avisos" del menú "Online" o del menú contextual.

O bien:

1. Seleccione la CPU en el árbol del proyecto.
2. Elija el comando "Recibir avisos" del menú "Online" o del menú contextual.

Nota

Si elige uno de los dos procedimientos mencionados anteriormente, debe establecer primero una conexión online con el dispositivo.

8.1.6.5 Exportar el fichero

Para archivar avisos, puede exportar el fichero. Proceda del siguiente modo:

1. Cambie a la vista de fichero.
2. Haga clic en el símbolo "Exportar fichero".
3. Elija en el diálogo abierto la ruta de la ubicación a la que desea exportar el fichero.

Resultado

El fichero se guarda como archivo xml en la ubicación indicada por usted.

8.1.6.6 Vaciar el fichero

El fichero está concebido como búfer en anillo, es decir, cuando está lleno, se borra el aviso más antiguo del fichero con cada nuevo aviso que entra. Con el botón "Vaciar fichero" se borra todo el contenido del fichero.

Procedimiento

Para vaciar el fichero, proceda del siguiente modo:

1. Haga clic en el botón "Vaciar fichero" de la barra de herramientas del visor de avisos.

8.1.6.7 Vista "Avisos activos"

La vista "Avisos activos" es una imagen de la memoria de acuse de avisos de la(s) CPU seleccionada(s).

8.1.6.8 Estructura de los avisos en la vista "Avisos activos"

La vista "Avisos activos" muestra una imagen de la memoria de acuse de avisos de las CPUs seleccionadas. Por cada aviso pendiente se muestra una entrada en la tabla. Los eventos de un aviso ("Aparecido", "Desaparecido" y "Acusado") se muestran en una fila.

Estructura de la tabla

Todos los atributos de los avisos se pueden representar en forma de columnas. Es posible mostrar u ocultar las columnas de manera individual y modificar el ancho y el orden de las columnas. Esta configuración se guarda al cerrar el proyecto.

Los avisos se pueden mostrar en una o varias filas. En el modo de visualización de una sola fila, sólo se muestra la primera fila de los avisos de varias filas.

Los avisos pueden ser tanto de acuse obligado como no de acuse obligado. Las alarmas de acuse obligado pendientes de acuse se destacan en color azul y se pueden acusar de modo contextual con el respectivo botón de la barra de herramientas o con el comando del menú contextual "Acusar aviso(s)".

8.1.6.9 Estado de los avisos

En función de si se encuentra en la vista "Avisos activos" o en la vista de fichero, los avisos mostrados pueden tener diferentes estados.

Estado de los avisos en la vista "Avisos activos"

- AP: se ha recibido el aviso
- AAC: se ha recibido y acusado el aviso
- AD: ha desaparecido el aviso

Si se reciben más cambios de señales de los que se pueden enviar (Signal Overflow), en el estado se muestra una O y el estado se indica en rojo.

Estado de los avisos en la vista de fichero

- ninguna indicación: sólo para avisos que ha creado el PG/PC y se muestran en la ficha "Archivo", p. ej. estado de aviso, interrupción de la conexión, cambios del estado operativo
- AP: se ha recibido el aviso
- AC: se ha recibido y acusado el aviso
- D: ha desaparecido el aviso
- L: se ha borrado el aviso

Si se reciben más cambios de señales de los que se pueden enviar (Signal Overflow), en el estado se muestra una O y el estado se indica en rojo.

8.1.6.10 Acusar avisos

Los avisos de acuse obligado se muestran en color azul.

Procedimiento

Para acusar un aviso, proceda del siguiente modo:

1. Seleccione el o los aviso(s) deseado(s) en la tabla.
2. Haga clic en el botón "Acusar".

Nota

Además, puede seleccionar varios avisos pendientes de acuse simultáneamente. Para ello, mantenga pulsada la tecla <Ctrl> y seleccione a continuación los avisos deseados en la tabla.

Resultado

El aviso seleccionado ha sido acusado y se muestra con un estilo de fuente normal.

Nota

En la vista "Avisos activos" ya no se muestran los avisos acusados que ya han desaparecido.

8.1.6.11 Ignorar avisos

Ignorar avisos

Para ignorar avisos, proceda de la manera siguiente:

1. Haga clic en el botón "Ignorar".
   El botón aparece sobre fondo gris.

Resultado

A partir de este momento, se ignoran todos los avisos. En la vista de fichero se crea un aviso que indica que la visualización de avisos y eventos está desactivada.

Desactivar la función de ignorar avisos

Para desactivar la función de ignorar avisos, proceda del siguiente modo:

1. Haga clic en el botón "Ignorar".
   El botón aparece sobre fondo blanco.

Resultado

Todos los avisos, es decir, también aquellos avisos, que hay actualmente en la CPU y que han aparecido mientras estaba activada la función "Ignorar avisos", se vuelven a mostrar a

partir de este momento. En la vista de fichero se crea un aviso que indica que la visualización de avisos y eventos está activada.

8.1.6.12 Manejo del teclado en el visor de avisos

Visor de avisos

Función	Combinación de teclas (método abreviado)
Seleccionar todos los avisos	Ctrl+A
Acusar todos los avisos seleccionados	Ctrl+Q

8.2 Diagnosticar dispositivos y redes

8.2.1 Diagnosticar hardware

8.2.1.1 Resumen del diagnóstico de hardware

Posibilidades generales para diagnosticar el hardware

Posibilidades generales del diagnóstico de hardware

El hardware se diagnostica del modo siguiente:

- Desde la vista online y de diagnóstico
- Desde la Task Card "Herramientas online"
- Desde el área "Diagnóstico > Información del dispositivo" de la ventana de inspección
- Mediante símbolos de diagnóstico, p. ej. en la vista de dispositivos y en el árbol del proyecto

Estructura de la vista online y de diagnóstico

La vista online y de diagnóstico consta de dos ventanas colindantes:

- La ventana izquierda muestra una estructura de árbol con carpetas y grupos (si las carpetas están abiertas).
- La ventana derecha contiene información detallada sobre la carpeta o el grupo seleccionado.

Existen el grupo "Accesos online" y las carpetas "Diagnóstico" y "Funciones".

- Grupo "Accesos online": contiene la visualización de si actualmente existe o no una conexión online con el destino correspondiente. Asimismo, es posible establecer o deshacer la conexión online.
- "Diagnóstico": Contiene varios grupos de diagnóstico relacionados con el módulo seleccionado.
- "Funciones": Contiene varios grupos en los que se realizan ajustes en el módulo seleccionado o se dan comandos al módulo.

Función y estructura de la Task Card "Herramientas online"

En la Task Card "Herramientas online" se lee información de diagnóstico actual y se transfieren comandos al módulo en aquellos que tienen un estado operativo propio (p. ej. CPUs).

Se antes de activar la Task Card "Herramientas online" se ha seleccionado un módulo que no tiene estado operativo propio o bien varios módulos, la Task Card hará referencia a la CPU correspondiente.

La Task Card "Herramientas online" consta de las paletas siguientes:

- Panel de control de la CPU
- Tiempo de ciclo
- Memoria

Nota

Una paleta sólo tiene contenido si el módulo conoce las funciones correspondientes y si existe una conexión online.

Si no existe ninguna conexión online con el módulo en cuestión, en cada paleta se mostrará "Ninguna conexión online" en azul. Si se ha deshecho una conexión online existente, se muestra "No se puede acceder a este destino".

Estructura de la ficha "Diagnóstico" de la ventana de inspección

La ficha "Diagnóstico" de la ventana de inspección contiene a su vez varias fichas. De ellas, la siguiente es relevante para el diagnóstico de hardware:

- Información del dispositivo
  Esta ficha hace referencia a todas las CPUs del proyecto con las que existe una conexión online. Aquí se visualizan avisos si hay una o varias CPUs averiadas o si no están en estado operativo RUN.

Consulte también

Principios básicos de las Task Cards (Página 301)

Ventana de inspección (Página 299)

Determinar los dispositivos defectuosos que están conectados online

Resumen de los dispositivos defectuosos

El área "Diagnóstico > Información del dispositivo" de la ventana de inspección ofrece una vista general de los dispositivos defectuosos con los que existe o ha existido una conexión online.

El área "Diagnóstico > Información del dispositivo" de la ventana de inspección consta de los elementos siguientes:

- Encabezado con el número de dispositivos defectuosos
- Tabla con información detallada de cada dispositivo defectuoso

Cuando se intenta establecer una conexión online con un dispositivo, éste se considera defectuoso si no es accesible, si notifica un error como mínimo o si no está en estado operativo "RUN".

Estructura de la tabla con información detallada de los dispositivos defectuosos

La tabla contiene las columnas siguientes:

- Estado online: contiene el estado online como símbolo de diagnóstico y en palabras
- Estado operativo: contiene el estado operativo como símbolo y en palabras
- Dispositivo / módulo: nombre del dispositivo o módulo afectado
- Aviso: Explica la entrada de la columna anterior
- Detalles: el enlace abre la vista online y de diagnóstico correspondiente al dispositivo y la pone en primer plano. Si ya no existe ninguna conexión online, el enlace abre el diálogo para establecer la conexión.
- Ayuda: el enlace proporciona información adicional sobre el fallo aparecido.

Consulte también

Mostrar el estado de diagnóstico y de comparación mediante iconos (Página 2160)

Mostrar el estado de diagnóstico y de comparación mediante iconos

Determinar online el estado de diagnóstico y mostrarlo mediante iconos

Al establecer la conexión online con un dispositivo se determina también su estado de diagnóstico y el de sus componentes subordinados (si los tiene), así como, en su caso, el estado operativo.

A continuación se describe en qué vista se muestran qué iconos.

- Vista de dispositivos
  - Para cada componente de hardware (a excepción de la Signal board de la CPU) se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - En los componentes de hardware con estado operativo propio, se muestra adicionalmente el símbolo de estado operativo a la izquierda o encima del símbolo de diagnóstico.
  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: no se muestran símbolos de diagnóstico (debido a la configuración como dispositivo GSDML).
- Vista general de dispositivos
  - Para cada componente de hardware se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: para los módulos asignados a la CPU se muestra el correspondiente símbolo de diagnóstico (los módulos no asignados no obtienen símbolo de diagnóstico). De los submódulos enchufables de un módulo asignado se muestra el correspondiente símbolo de diagnóstico (los submódulos no enchufables no son visibles y por tanto tampoco reciben símbolo de diagnóstico).
- Vista de redes
  - Para cada dispositivo se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: se muestra un símbolo de diagnóstico. Corresponde a la parte de la estación que está asignada a la CPU.

- Vista general de redes
  - Para cada componente de hardware se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: se muestra un símbolo de diagnóstico. Corresponde a la parte de la estación que está asignada a la CPU.
- Vista topológica
  - Para cada dispositivo se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - Para cada puerto se muestra el correspondiente icono de diagnóstico. El significado de los diferentes colores se describe más abajo.
  - Cada cable entre dos puertos que están online se muestra en el color que corresponde a su estado de diagnóstico.
    El color del cable entre dos puertos depende del estado de los distintos puertos:

Color del primer puerto	Color del segundo puerto	Color del cable de conexión
Verde	Verde	Verde
Verde	Gris	Gris
Verde	Rojo	Rojo
Gris	Rojo	Rojo

  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: se muestra un símbolo de diagnóstico. Corresponde a la parte de la estación que está asignada a la CPU.
- Vista general de la topología
  - Para cada componente de hardware se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: se muestra un símbolo de diagnóstico. Corresponde a la parte de la estación que está asignada a la CPU.

- Árbol del proyecto
  - Detrás de cada componente de hardware se muestra el correspondiente icono de diagnóstico.
  - Si en un componente de hardware con componentes subordinados (p. ej. periferia descentralizada, Esclavo_1) hay un error de hardware en por lo menos un componente subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - En los componentes de hardware con estado operativo propio, se muestra adicionalmente el icono de estado operativo en la esquina superior derecha del icono de diagnóstico.
  - Si en una CPU está activado el forzado permanente, se visualiza una F roja en el borde izquierdo del símbolo de diagnóstico.
  - Detrás de la carpeta "Módulos locales" se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado" cuando hay un error de hardware en por lo menos uno de los módulos pertinentes.
  - Detrás de la carpeta "Periferia descentralizada" se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado" cuando hay un error de hardware en por lo menos uno de los componentes pertinentes.
  - Detrás de la carpeta del proyecto se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado" cuando detrás de por lo menos una de las carpetas "Módulos locales" y "Periferia descentralizada" se muestra el símbolo de diagnóstico "Error de hardware en componente subordinado".
  - Para los módulos o submódulos de un Shared Device de una CPU S7-1500 se aplica: de los módulos asignados a la CPU se muestra el correspondiente símbolo de diagnóstico (los módulos no asignados aparecen en color gris y no obtienen símbolo de diagnóstico). De los submódulos enchufables de un módulo asignado se muestra el correspondiente símbolo de diagnóstico (los submódulos no enchufables no son visibles y por tanto tampoco reciben símbolo de diagnóstico).

Nota

Si en un componente de hardware se presenta el diagnóstico "no es accesible desde la CPU", el símbolo de diagnóstico "Error de hardware en componente subordinado" no se muestra adicionalmente.

Símbolos de diagnóstico para módulos y dispositivos

La tabla siguiente muestra los iconos posibles y su respectivo significado.

Icono	Significado
	Se está estableciendo una conexión con una CPU.
	La CPU ya no es accesible en la dirección ajustada.

Icono	Significado
	La CPU configurada y la realmente existente no son compatibles debido al tipo.
	Al establecer la conexión online con una CPU protegida se canceló el cuadro de diálogo sin introducir la contraseña correcta.
	No hay fallos
	Mantenimiento necesario
	Mantenimiento solicitado
	Error
	El módulo o el dispositivo están desactivados.
	No es posible acceder al módulo o dispositivo desde la CPU (válido para módulos y dispositivos que están por debajo de una CPU).
	No hay datos de diagnóstico disponibles, porque los datos de configuración online actuales difieren de los datos de configuración offline.
	El módulo o dispositivo configurado y el módulo o dispositivo realmente existente son incompatibles (válido para módulos y dispositivos que están por debajo de una CPU).
	El módulo configurado no soporta la visualización del estado de diagnóstico (válido para módulos dentro de una CPU).
	La conexión está establecida, pero el estado del módulo todavía se está determinando o se desconoce.
	El módulo configurado no soporta la visualización del estado de diagnóstico.
	Error de hardware en componente subordinado: existe un error de hardware en por lo menos un componente de hardware subordinado (solo aparece como símbolo independiente en el árbol del proyecto).

Nota

Algunos módulos, como p. ej. el FM 450-1, tan solo se identifican como erróneos (caso de producirse un error) si el usuario ha habilitado la alarma de diagnóstico al parametrizar las propiedades del módulo en cuestión.

Símbolos del estado de comparación

Los símbolos de diagnóstico pueden combinarse en la parte inferior derecha con símbolos de estado más pequeños que muestran el resultado de la comparación online/offline. La tabla siguiente muestra los símbolos de comparación posibles y su significado.

Icono	Significado
	Error de hardware en componente subordinado: en por lo menos un componente de hardware subordinado las versiones online y offline son diferentes (solo en el árbol del proyecto).
	Error de software en componente subordinado: en por lo menos un componente de software subordinado las versiones online y offline son diferentes (solo en el árbol del proyecto).
	La versión online del objeto es distinta de la offline.
	El objeto solo existe online.
	El objeto solo existe offline.
	La versión online del objeto es igual que la offline.

Nota

Para que en la vista de dispositivos aparezcan en la parte inferior derecha tanto un símbolo de comparación como el símbolo de diagnóstico "Error en un componente subordinado", se aplica la regla siguiente: El icono de diagnóstico para el componente de hardware subordinado tiene prioridad por encima del icono de comparación. En consecuencia, el icono de comparación solo se muestra si no existe ningún error en los componentes de hardware subordinados.

Visualización de errores de software en el árbol del proyecto

- Detrás de cada bloque se muestra el símbolo de comparación correspondiente.
- Detrás de cada carpeta en la que solo hay bloques se muestra el símbolo de diagnóstico "Error de software en componente subordinado" cuando hay un error de software en por lo menos uno de los bloques pertinentes.
- Si en un componente de hardware con componentes de software subordinados no hay ningún error de hardware y hay un error en por lo menos un componente de software subordinado, el símbolo de diagnóstico tiene el aspecto siguiente: el símbolo de diagnóstico del componente de hardware se representa atenuado y, además, en el extremo inferior derecho se muestra el símbolo de diagnóstico "Error de software en componente subordinado".

Símbolos combinados de diagnóstico y comparación

La siguiente tabla muestra ejemplos de símbolos que se visualizan en el símbolo de diagnóstico.

Icono	Significado
	La carpeta contiene objetos con versiones online y offline diferentes (solo en el árbol del proyecto).
	El objeto solo existe online.

Símbolos de estado operativo para CPU y CP

La tabla siguiente muestra los símbolos posibles y sus respectivos estados operativos.

Icono	Estado operativo
	RUN
	STOP
	ARRANQUE
	PARADA
	AVERÍA
	Estado operativo desconocido
	El módulo configurado no soporta la visualización del estado operativo.

Nota

Si en una CPU está activado el forzado permanente, se visualiza una F roja con fondo rosa debajo y a la derecha del símbolo de estado operativo.

Código de color para puertos y cables Ethernet

La tabla siguiente muestra los colores posibles y su respectivo significado.

Color	Significado
	No hay fallos ni mantenimiento necesario
	Mantenimiento solicitado
	Fallo de comunicación o error de topología
	No diagnosticable

Iniciar la vista online y de diagnóstico

Resumen de las posibilidades de iniciar la vista online y de diagnóstico

La vista online y de diagnóstico de un módulo que debe diagnosticarse se puede abrir desde los siguientes lugares:

- Vista general
- Árbol del proyecto
- Vista de dispositivos
- Vista general de dispositivos
- Vista de redes
- Vista general de redes
- Vista topológica

A continuación se muestra con ejemplos cómo hay que proceder.

Requisitos

El proyecto con el módulo que debe diagnosticarse está abierto.

Nota

Este requisito no es imprescindible si se llama la vista online y de diagnóstico desde el árbol del proyecto después de haber determinado los nodos accesibles.

Procedimiento

Para iniciar la vista online y de diagnóstico de un módulo, proceda del siguiente modo:

1. Abra la carpeta de dispositivos correspondiente en el árbol del proyecto.
2. Haga doble clic en "Online y diagnóstico".

O bien:

1. Seleccione la carpeta de dispositivos correspondiente en el árbol del proyecto.
2. Elija el comando "Online y diagnóstico" del menú contextual o del menú principal "Online".

O bien:

1. Abra la carpeta "Accesos online" en el árbol del proyecto.
2. Abra la carpeta de la interfaz a través de la que desea establecer la conexión online.
3. Haga doble clic en "Mostrar /actualizar nodos accesibles".
4. Seleccione el módulo que debe diagnosticarse.
5. Elija el comando "Online y diagnóstico" del menú contextual o del menú principal "Online".

O bien:

1. Abra la carpeta "Módulos locales" en el árbol del proyecto.
2. Seleccione el dispositivo en cuestión o el módulo que debe diagnosticarse.
3. Elija el comando "Online y diagnóstico" del menú contextual o del menú principal.

O:

1. Abra la vista de dispositivos de la configuración de dispositivos.
2. Seleccione el módulo que debe diagnosticarse.
3. Elija el comando "Online y diagnóstico" del menú contextual o del menú principal "Online".

O:

1. Abra la vista de dispositivos de la configuración de dispositivos.
2. Establezca una conexión online con el módulo que se va a diagnosticar.
3. Haga doble clic sobre el símbolo de diagnóstico situado encima del módulo.

O bien:

1. Abra la vista de redes de la configuración de dispositivos.
2. Seleccione la estación con el módulo que debe diagnosticarse.
3. Elija el comando "Online y diagnóstico" del menú contextual o del menú principal "Online".

O bien:

Resultado

Se inicia la vista online y de diagnóstico del módulo que debe diagnosticarse. Si previamente existía una conexión online con la CPU correspondiente, la barra de título de la vista online y de diagnóstico se resaltará en color naranja.

Nota

Si al iniciar la vista online y de diagnóstico no existe ninguna conexión online, no aparecerá ninguna información online y los campos correspondientes estarán vacíos.

Activar la Task Card "Herramientas online"

Activar la Task Card "Herramientas online"

Esta Task Card se activa del siguiente modo:

1. Inicie la vista online y de diagnóstico.
2. Haga clic en la Task Card "Herramientas online".

O bien:

1. Inicie la vista de dispositivos.
2. Haga clic en la Task Card "Herramientas online".

O bien:

1. Inicie la vista de redes.
2. Haga clic en la Task Card "Herramientas online".

8.2.1.2 Mostrar valores actuales y no modificables de propiedades de módulos parametrizables

Mostrar propiedades generales e información relevante para la instalación relacionadas con un módulo

¿Dónde está la información buscada?

Las propiedades generales y la información relevante para la instalación relacionadas con un módulo están en el grupo "General" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse.

Estructura del grupo "General"

El grupo "General" incluye las áreas siguientes:

- Módulo
- Información del módulo
- Información del fabricante

Área "Módulo"

Esta área muestra los datos siguientes del módulo:

- Nombre abreviado, p. ej. CPU 1214C DC/DC/DC
- Referencia
- Hardware
- Firmware
- Rack
- Slot

Área "Información del módulo"

Esta área muestra los datos siguientes del módulo que se han ajustado al configurar el hardware:

- Nombre del módulo
- Fecha de montaje (no se muestra en todos los módulos)
- Información adicional (no se muestra en todos los módulos)

Área "Información del fabricante"

Esta área muestra los datos siguientes del módulo:

- Fabricante
- Número de serie
- Perfil: ID de perfil en formato de número hexadecimal

  Nota

  Encontrará el nombre del perfil correspondiente en la "Profile ID Table" en PROFIBUS International (véase "www.profibus.com").

- Detalles del perfil: tipo específico del perfil en formato de número hexadecimal

  Nota

  Encontrará el nombre correspondiente al tipo específico del perfil en la "Profile Specific Type Table" en PROFIBUS International (véase "www.profibus.com").

Mostrar los tiempos de ciclo parametrizados

¿Dónde está la información buscada?

Encontrará la información buscada en los lugares siguientes:

- En el grupo "Tiempo de ciclo" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse
- En la paleta "Tiempo de ciclo" de la Task Card "Herramientas online"

Estructura del grupo "Tiempo de ciclo" de la carpeta "Diagnóstico" de la vista online y de diagnóstico

El grupo "Tiempo de ciclo" incluye las áreas siguientes:

- Diagrama del tiempo de ciclo (vista gráfica de los tiempos de ciclo parametrizados y medidos)
- Tiempo de ciclo parametrizado (visualización de los tiempos de ciclo parametrizados en forma de valores absolutos)
- Tiempos de ciclo medidos (visualización de los tiempos de ciclo medidos en forma de valores absolutos)

Estructura de la paleta "Tiempo de ciclo" de la Task Card "Herramientas online"

La paleta "Tiempo de ciclo" muestra el diagrama del tiempo de ciclo y debajo de éste los tiempos de ciclo medidos, en forma de valores absolutos.

Tiempos de ciclo parametrizados

En el diagrama del tiempo de ciclo y en el área "Tiempo de ciclo parametrizado" se muestran los siguientes tiempos de ciclo parametrizados:

- Tiempo de ciclo mínimo
- Tiempo de vigilancia del ciclo

En el diagrama del tiempo de ciclo, el tiempo de ciclo mínimo y el tiempo de vigilancia de ciclo se corresponden con las dos marcas del eje de tiempo.

En el área "Tiempo de ciclo parametrizado" se muestran los tiempos de ciclo parametrizados en forma de valores absolutos.

Mostrar interfaces y propiedades de interfaces de un módulo

¿Dónde se encuentra la información buscada?

Las interfaces y las propiedades de interfaces de un módulo están en el siguiente grupo de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse:

- Interfaz PROFINET

Grupo "Interfaz PROFINET"

Este grupo se divide en las siguientes áreas:

- "Dirección Ethernet" con las subáreas "Conexión de red" y "Parámetro IP"
- "Puertos"

Subárea "Conexión de red" del área "Dirección Ethernet"

Esta subárea muestra los datos siguientes del módulo:

- Dirección MAC:
  Dirección MAC de la interfaz.
  La dirección MAC consta de dos partes. La primera parte ("dirección MAC básica") identifica al fabricante (Siemens, 3COM, ...). La segunda parte de la dirección MAC diferencia los distintos dispositivos Ethernet. Cada módulo Ethernet tiene asignada una dirección MAC unívoca.

Subárea "Parámetro IP" del área "Dirección Ethernet"

Esta subárea muestra los datos siguientes del módulo:

- Dirección IP:
  Dirección de protocolo de Internet del dispositivo conectado al bus (TCP/IP)
- Máscara de subred:
  La máscara de subred indica qué parte de la dirección IP establece la pertenencia a una determinada subred.

- Router predeterminado:
  Si la subred está conectada a otras subredes con un router, debe conocerse la dirección IP del router estándar. Sólo de este modo puede transmitirse un datagrama con una dirección de subred divergente.
- Configuración IP:
  Identificación de la ruta a través de la cual el dispositivo ha recibido su configuración IP (dirección IP, máscara de subred, router predeterminado).

Identificador	Significado
0	Dirección IP no inicializada
1	Mediante configuración (es decir, mediante los ajustes que se han cargado en el dispositivo desde la vista de dispositivos o la vista de redes)
2	Mediante el grupo "Asignar dirección IP" de la vista online y de diagnóstico
3	Mediante servidor DHCP (es decir, los parámetros IP se han obtenido a través de un servicio especial desde un servidor DHCP (Dynamic Host Configuration Protocol) y se han asignado por un tiempo limitado)
4	Dirección IP ajustada por un programa de usuario
5	Origen de la dirección IP desconocido

- Tiempo de ajuste IP:
  Sello de tiempo de la última modificación de la dirección IP directamente desde la conexión Ethernet del módulo

Área "Puertos"

Esta área muestra los datos siguientes del módulo:

- Puertos Ethernet
  Propiedades físicas de la interfaz PROFINET

Propiedad de la interfaz PROFINET	Significado
N.º de puerto	Número de puerto. Entre paréntesis se indica la denominación abreviada de la interfaz (X + n.º de la interfaz) y el puerto (P + n.º del puerto). Una "R" en la denominación abreviada de un puerto significa que se trata de un puerto anillo.
Estado	Visualización del estado del LED LINK correspondiente al puerto: • Si se muestra "OK", hay otro dispositivo (p. ej. un switch) conectado al puerto, y la conexión física existe. • Si se muestra "Desconectado", no hay ningún otro dispositivo conectado al puerto. • Si se muestra "Desactivado", el acceso al puerto está cerrado.
Configuración	• "Automática" en caso de configuración de red automática del dispositivo • Configuración de red en lo que respecta a la velocidad y el método de transferencia en caso de configuración de red manual del dispositivo
Modo de operación	Configuración de red en lo que respecta a la velocidad y el método de transferencia

Al seleccionar una línea en la tabla de puertos, se muestra información de ayuda adicional sobre el puerto en cuestión.

Mostrar controladores IO que accedan a módulos de un Shared Device

¿Dónde se encuentra la información buscada?

Los controladores IO que acceden a módulos de un Shared Device se muestran en la vista Online y diagnóstico del módulo de interfaz del Shared Device en la carpeta "Diagnóstico", en el área siguiente del grupo "Interfaz PROFINET":

- Controlador IO

Consulte también

Información importante sobre los Shared Devices (Página 1577)

Visualización de las propiedades del dominio Sync de un dispositivo PROFINET

¿Dónde se encuentra la información buscada?

Las propiedades del dominio Sync de un dispositivo PROFINET están en la vista Online y de diagnóstico del dispositivo que debe diagnosticarse, dentro de la carpeta "Diagnóstico" en el área siguiente del grupo "Interfaz PROFINET":

- Dominio

Área "Dominio"

Esta área se divide en las subáreas siguientes:

- Dominio Sync
- Dominio MRP

¿Qué es un dominio Sync?

Un dominio Sync es un grupo de dispositivos PROFINET que se sincronizan con un tiempo de ciclo común. Un único dispositivo tiene la función de maestro Sync (generador de impulsos) y los demás tienen la función de esclavos Sync. El maestro Sync es casi siempre un controlador IO o un switch.

Los dispositivos PROFINET no sincronizados no forman parte de un dominio Sync.

Subárea "Dominio Sync" del área "Dominio"

Esta subárea muestra las propiedades siguientes del dominio Sync:

- Nombre:
  Nombre del dominio Sync
- Función:
  función del dispositivo PROFINET en el dominio Sync. Son posibles las funciones siguientes:
  - maestro Sync
  - esclavo Sync
- Intervalo de sincronización:
  intervalo en el que se ejecuta la sincronización
- Tiempo de ciclo de emisión
  el intervalo de emisión mínimo posible para el intercambio de datos
- Precisión de jitter del tiempo de ciclo de emisión
- Ancho de banda reservado para la comunicación cíclica

Visualización de las propiedades del dominio MRP de un dispositivo PROFINET

¿Dónde se encuentra la información buscada?

Las propiedades del dominio MRP de un dispositivo PROFINET están en la vista Online y de diagnóstico del dispositivo que debe diagnosticarse, dentro de la carpeta "Diagnóstico" en el área siguiente del grupo "Interfaz PROFINET":

- Dominio

Área "Dominio"

Esta área se divide en las subáreas siguientes:

- Dominio Sync
- Dominio MRP

¿Qué es un dominio MRP?

Mediante el denominado Media Redundancy Protocol (MRP) es posible construir redes redundantes. Los trayectos de transmisión redundantes (topología de anillo) se encargan de proporcionar una ruta de comunicación alternativa en caso de que falle una vía de comunicación. Los dispositivos PROFINET que forman parte de la red redundante componen un dominio MRP.

Subárea "Dominio MRP" del área "Dominio"

Esta subárea muestra las propiedades siguientes del dominio MRP:

- Nombre:
  nombre del dominio MRP
- Función:
  función del dispositivo PROFINET en el dominio MRP. Son posibles las funciones siguientes:
  - Manager
  - Manager (auto)
  - Cliente
  - No es estación del anillo
- Puerto en anillo 1:
  El puerto del dispositivo PROFINET que tiene la propiedad "Puerto en anillo 1".
- Puerto en anillo 2:
  El puerto del dispositivo PROFINET que tiene la propiedad "Puerto en anillo 2".
- Estado del anillo MRP:
  Aquí se indica si el anillo está interrumpido (estado "abierto") o no (estado "cerrado").

Ver el firmware actual de un módulo

Ver firmware

Es posible ver el firmware instalado actualmente en un módulo.

Requisitos

- El módulo soporta actualización de firmware.
- El módulo está conectado online.

Procedimiento

Para ver el firmware actual, proceda del siguiente modo:

1. Abra el módulo en la vista online y de diagnóstico.
2. Elija el grupo "Actualización de firmware" de la carpeta "Funciones".
3. Puede consultar el firmware actual en el apartado "Firmware" del área "Datos online".

8.2.1.3 Mostrar valores actuales de propiedades de módulos dinámicas

Mostrar los tiempos de ciclo medidos

¿Dónde se encuentra la información buscada?

Encontrará los tiempos de ciclo medidos en los lugares siguientes:

- En la vista Online y diagnóstico del módulo que se desea diagnosticar, en la carpeta "Diagnóstico" del grupo "Tiempo de ciclo"
- En la paleta "Tiempo de ciclo" de la Task Card "Herramientas online"

Estructura del grupo "Tiempo de ciclo" en la carpeta "Diagnóstico" de la vista Online y diagnóstico

El grupo "Tiempo de ciclo" incluye las áreas siguientes:

- Diagrama del tiempo de ciclo (vista gráfica de los tiempos de ciclo parametrizados y medidos)
- Tiempo de ciclo parametrizado (visualización de los tiempos de ciclo parametrizados en forma de valores absolutos)
- Tiempos de ciclo medidos (visualización de los tiempos de ciclo medidos en forma de valores absolutos)

Estructura de la paleta "Tiempo de ciclo" de la Task Card "Herramientas online"

La paleta "Tiempo de ciclo" muestra el diagrama del tiempo de ciclo y debajo de éste los tiempos de ciclo medidos, en forma de valores absolutos.

Visualización gráfica de los tiempos de ciclo medidos

En el diagrama del tiempo de ciclo se muestran los siguientes tiempos de ciclo medidos:

- Tiempo de ciclo más corto: duración del ciclo más corto desde la última transición de STOP a RUN.
  Corresponde en el diagrama a la flecha gris discontinua de la izquierda.
- Tiempo de ciclo actual/último: duración del último ciclo.
  Corresponde en el diagrama a la flecha verde. Si el actual/último tiempo de ciclo sobrepasa el tiempo de vigilancia del ciclo, la flecha se vuelve roja.

  Nota

  Si la duración del último ciclo se aproxima al tiempo de vigilancia del ciclo, es posible que éste se sobrepase. Dependiendo del tipo de CPU, de la parametrización y del programa de usuario, la CPU puede pasar al estado operativo STOP. La observación de las variables del programa, p. ej., prolonga el tiempo de ciclo.

  Si el ciclo dura más del doble del tiempo de vigilancia del ciclo y el tiempo de vigilancia del ciclo no se vuelve a iniciar en el programa de usuario (llamando la instrucción avanzada RE_TRIGR), la CPU pasa a STOP.

- Tiempo de ciclo más largo: duración del ciclo más largo desde la última transición de STOP a RUN.
  Corresponde en el diagrama a la flecha azul discontinua de la derecha.

Entre las dos flechas discontinuas se extiende una franja azul que corresponde a la totalidad del rango de los tiempos de ciclo medidos. Si un tiempo de ciclo medido es mayor que el tiempo de vigilancia del ciclo, se muestra en rojo la parte de la franja que queda fuera de los límites parametrizados.

Visualización de los tiempos de ciclo medidos en forma de valores absolutos

En el área "Tiempos de ciclo medidos" y en la paleta "Tiempo de ciclo" se muestran los siguientes tiempos de ciclo medidos:

- Tiempo de ciclo más corto desde la última transición de STOP a RUN
- Tiempo de ciclo último o actual
- Tiempo de ciclo más largo desde la última transición de STOP a RUN

Mostrar el estado actual de los LEDs de una CPU

¿Dónde está la información buscada?

El estado actual de los LEDs de una CPU está en el área de visualización de la paleta "Panel de control de la CPU" de la Task Card "Herramientas online".

Área de visualización de la paleta "Panel de control de la CPU" de la Task Card "Herramientas online"

Esta área contiene los indicadores siguientes:

- Nombre de la estación y tipo de CPU (nombre abreviado)
- RUN / STOP (equivale al LED "RUN / STOP" de la CPU)
- ERROR (equivale al LED "ERROR" de la CPU)
- MAINT (equivale al LED "MAINT" de la CPU)

Mostrar el nivel de llenado de todos los tipos de memoria de una CPU

¿Dónde se encuentra la información buscada?

Encontrará el nivel de llenado de todos los tipos de memoria de una CPU en los lugares siguientes:

- En el área de visualización del grupo "Memoria" de la carpeta "Diagnóstico" de la vista Online y diagnóstico del módulo que debe diagnosticarse
- En el área de visualización de la paleta "Memoria" de la Task Card "Herramientas online"

Área de visualización del grupo "Memoria" de la carpeta "Diagnóstico" de la vista online y de diagnóstico

Esta área contiene la carga de memoria actual del módulo correspondiente, así como detalles de las distintas áreas de memoria.

La carga de memoria se muestra tanto en forma de diagrama de barras como porcentualmente en forma de valor numérico.

Se muestran las cargas de memoria siguientes:

- Memoria de carga
  Si no hay ninguna Memory Card insertada, se mostrará la memoria de carga interna.
  Si hay una Memory Card insertada, el sistema operativo utiliza exclusivamente la memoria de carga insertada, que es la que se muestra aquí.
- Memoria de trabajo
- Memoria remanente

Área de visualización de la paleta "Memoria" de la Task Card "Herramientas online"

Esta área contiene la carga actual de memoria del módulo correspondiente. La memoria libre se muestra tanto en forma de diagrama de barras como porcentualmente en forma de valor numérico. El valor numérico se redondea a un valor entero.

Nota

Si un área de memoria tiene una ocupación inferior al 1 %, en la parte libre de dicha área se mostrará "99 %".

Se muestran las cargas de memoria siguientes:

- Memoria de carga
  Si no hay ninguna Memory Card insertada, se mostrará la memoria de carga interna.
  Si hay una Memory Card insertada, el sistema operativo utiliza exclusivamente la memoria de carga insertada, que es la que se muestra aquí.
- Memoria de trabajo
- Memoria remanente

Consulte también

Memoria de carga (Página 1630)

Memoria de trabajo (Página 1631)

Áreas de memoria remanentes (Página 1632)

Visualización del nivel de llenado de todos los tipos de memoria de una CPU S7-1500

¿Dónde se encuentra la información buscada?

Encontrará el nivel de llenado de todos los tipos de memoria de una CPU S7-1500 en los lugares siguientes:

- En el área de visualización del grupo "Memoria" de la carpeta "Diagnóstico" de la vista Online y diagnóstico del módulo que debe diagnosticarse
- En el área de visualización de la paleta "Memoria" de la Task Card "Herramientas online"

Área de visualización del grupo "Memoria" de la carpeta "Diagnóstico" de la vista online y de diagnóstico

Esta área contiene la carga de memoria actual del módulo correspondiente, así como detalles de las distintas áreas de memoria.

La carga de memoria se muestra tanto en forma de diagrama de barras como porcentualmente en forma de valor numérico.

Se muestran las cargas de memoria siguientes:

- Memoria de carga

  Nota

  La memoria de carga se encuentra en la Memory Card SIMATIC.

- Memoria de trabajo para código: memoria de trabajo para código de programa
- Memoria de trabajo para datos: memoria de trabajo para bloques de datos
- Memoria remanente

Área de visualización de la paleta "Memoria" de la Task Card "Herramientas online"

Esta área contiene la carga actual de memoria del módulo correspondiente. La memoria libre se muestra tanto en forma de diagrama de barras como porcentualmente en forma de valor numérico. El valor numérico se redondea a un valor entero.

Nota

Si un área de memoria tiene una ocupación inferior al 1 %, en la parte libre de dicha área se mostrará "99 %".

Se muestran las cargas de memoria siguientes:

- Memoria de carga

  Nota

  La memoria de carga se encuentra en la Memory Card SIMATIC.

- Memoria de trabajo para código: memoria de trabajo para código de programa
- Memoria de trabajo para datos: memoria de trabajo para bloques de datos
- Memoria remanente

Consulte también

Áreas de memoria de la CPU (Página 1910)

8.2.1.4 Comprobar los fallos de un módulo

Determinar el estado de diagnóstico de un módulo

¿Dónde se muestra el estado de diagnóstico de un módulo?

El estado de diagnóstico de un módulo se muestra en el grupo "Estado de diagnóstico" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo.

El grupo "Estado de diagnóstico" incluye las áreas siguientes:

- Estado
- Diagnóstico estándar (en S7-300 y S7-400 sólo en módulos no CPU)

Área "Estado"

En esta área se muestra la siguiente información de estado:

- Estado del módulo desde el punto de vista de la CPU, p. ej.:
  - Módulo existente y en buen estado.
  - Módulo averiado.
    Si el módulo tiene un fallo y se ha habilitado la alarma de diagnóstico en la configuración, se mostrará el estado "Módulo averiado".
  - Módulo configurado pero no existente.
    Ej.: no hay datos de diagnóstico disponibles, porque la configuración online actual difiere de la configuración offline.
- Se han detectado divergencias entre el módulo configurado y el insertado. Se mostrará la referencia del tipo teórico y real, siempre que sea posible determinarla.

La cantidad de información visualizada depende del módulo seleccionado.

Área "Diagnóstico estándar"

En esta área se muestra la siguiente información de diagnóstico de módulos no CPU:

- Fallos internos y externos que se refieren a todo el módulo
- Eventos de diagnóstico relacionados

Ejemplos de dicha información de diagnóstico:

- Fallo de todo el respaldo
- Módulo averiado

Nota

Alarmas de diagnóstico

Sólo es posible notificar una alarma de diagnóstico a la CPU si el módulo es apto para esta función y se ha habilitado la alarma de diagnóstico.

La visualización de la alarma de diagnóstico es una captura instantánea. Los fallos esporádicos de un módulo se detectan en el búfer de diagnóstico de la CPU correspondiente.

Leer el búfer de diagnóstico de una CPU

¿Dónde se lee el búfer de diagnóstico de una CPU?

El búfer de diagnóstico de una CPU se lee en el grupo "Búfer de diagnóstico" de la carpeta "Diagnóstico" de la vista online y de diagnóstico.

Estructura del grupo "Búfer de diagnóstico"

El grupo "Búfer de diagnóstico" incluye las áreas siguientes:

- "Eventos"
- "Configuración"

Búfer de diagnóstico

El búfer de diagnóstico sirve de archivo de registro para los eventos de diagnóstico que se han producido en la CPU y los módulos asignados. Dichos eventos se registran siguiendo el orden de aparición, siendo el más reciente el que se representa en primer lugar.

Área "Eventos"

El área "Eventos" incluye los elementos siguientes:

- Casilla de verificación "El sello de tiempo de la CPU tiene en cuenta la hora local de la PG/el PC"
- Tabla de eventos
- Botón "Congelar lista" o "Descongelar"
- Detalles sobre el evento: N.º de evento, ID de evento, descripción, sello de tiempo, información entrante/saliente
- Botones "Ayuda del evento", "Abrir en el editor", "Guardar como..."

Casilla de verificación "El sello de tiempo de la CPU tiene en cuenta la hora local de la PG/el PC"

Si no se ha activado la casilla de verificación, se visualizarán las entradas del búfer de diagnóstico con la hora del módulo.

Si se ha activado la casilla de verificación, las entradas del búfer de diagnóstico se visualizarán con la hora resultante de la fórmula siguiente:

Hora visualizada = hora del módulo + offset de zona horaria de la PG / el PC

Para ello es imprescindible que la hora del módulo sea idéntica a la hora UTC.

Este ajuste se utiliza cuando se desea ver la hora local de la PG / el PC como hora en las entradas del búfer de diagnóstico del módulo.

Si se activa o desactiva la casilla de verificación, se adaptarán inmediatamente las indicaciones horarias de las entradas del búfer de diagnóstico.

Nota

Si el programa utiliza la instrucción "WR_SYS_T" o si se ajusta el reloj en tiempo real de la CPU a través de un panel de operador y no se emplea UTC, se recomienda desactivar la casilla de verificación "El sello de tiempo de la CPU tiene en cuenta la hora local de la PG/ el PC". En este caso, solo se tomará en consideración la hora del módulo.

Tabla de eventos

En la tabla se muestra la siguiente información para cada evento de diagnóstico:

- Número correlativo de la entrada
  La primera entrada contiene el evento más reciente.
- Fecha y hora del evento de diagnóstico
  Si no se indican ni fecha ni hora, significa que el módulo no tiene reloj integrado.
- Nombre abreviado del evento y, dado el caso, la reacción de la CPU

  Nota

  Si no es posible determinar un parámetro individual de un texto, en su lugar se indicará la cadena de caracteres "###".

  Si todavía no existe ningún texto de visualización para módulos o eventos nuevos, se indicarán los números de los eventos y los parámetros individuales en forma de valores hexadecimales.

- Icono para la información entrante/saliente
  La tabla siguiente muestra los iconos posibles y su respectivo significado.

Icono	Significado
	Evento entrante
	Evento saliente
	Evento entrante para el que no existe ningún evento saliente independiente
	Evento de diagnóstico definido por el usuario

- Solo en las CPUs S7-1200 y S7-1500: Icono del peso del evento
  La tabla siguiente muestra los iconos posibles y su respectivo significado.

Icono	Significado
	Sin mantenimiento o sin fallos
	Mantenimiento necesario

Icono	Significado
	Mantenimiento solicitado
	Error

Es posible modificar el orden de las columnas de la tabla de eventos, así como modificar su anchura y eliminarlas o volver a agregarlas individualmente. También pueden ordenarse del siguiente modo: por número correlativo, por "Fecha y hora" y por "Evento".

Botón "Congelar lista" o "Descongelar"

El botón "Congelar lista" o "Descongelar" solo está activo si existe una conexión online con la CPU.

El ajuste predeterminado es "Congelar lista".

Al hacer clic en el botón "Congelar lista" sucede lo siguiente:

- Se congela la visualización actual de las entradas del búfer de diagnóstico.
- La rotulación del botón cambia a "Descongelar".

Si se ha producido un error en la instalación, es posible que se produzcan eventos de diagnóstico con una sucesión demasiado rápida. Esto provoca una velocidad de actualización elevada de la visualización. En este caso, la congelación de la visualización permite examinar las circunstancias con más detalle y tranquilidad.

Cuando la visualización está congelada y se pulsa el botón "Descongelar", sucede lo siguiente:

- La visualización de las entradas del búfer de diagnóstico vuelve a actualizarse.
- La rotulación del botón cambia a "Congelar lista".

Nota

Cuando se congela la visualización del búfer de diagnóstico, la CPU sigue registrando eventos en el búfer.

Detalles sobre el evento

Al seleccionar una línea de la lista de eventos, se muestra información detallada sobre el evento correspondiente:

- Número correlativo del evento en el búfer de diagnóstico
- ID de evento
- Descripción del evento con información adicional dependiente del evento. Ejemplos de esta información adicional:
  - Comando que ha provocado el evento
  - Transición de estado operativo provocado por el evento de diagnóstico
- Sello de tiempo

- Solo en las CPUs S7-1200 y S7-1500: Datos I&M correspondientes (módulo, rack/slot, identificador de instalación, identificador de ubicación)
- Prioridad del evento
- Información sobre si se trata de un evento entrante o saliente

Botón "Ayuda del evento"

Si hace clic en este botón se describirá el evento seleccionado con más detalles y, dado el caso, se indicarán soluciones.

Nota

En unos pocos eventos el botón "Ayuda del evento" está atenuado.

Botón "Abrir en el editor"

La tabla siguiente muestra cuándo está activo el botón "Abrir bloque" y qué función alberga.

¿Cuándo está activo el botón "Abrir en el editor"?	¿Qué sucede al hacer clic en este botón?
Cuando en el evento de diagnóstico se hace referencia a una dirección relativa de un bloque. Esta es la dirección del comando que causa el evento.	La función "Abrir en el editor" abre el bloque referenciado en la vista offline por la instrucción de programación que causa el error. Esto permite comprobar el código fuente del bloque en el lugar indicado, modificarlo en caso necesario y, seguidamente, volver a cargarlo en la CPU.
Cuando el evento de diagnóstico ha sido desencadenado por un módulo.	La función "Abrir en el editor" abre la vista de dispositivos del módulo en cuestión.

Botón "Guardar como ..."

Si se hace clic en este botón, el contenido del búfer de diagnóstico se guardará en un archivo de texto. Como nombre de archivo se sugiere dependiendo del idioma "Diagnóstico" con la extensión ".txt". aunque se puede cambiar.

Área "Configuración"

El área "Configuración" está compuesta por los siguientes elementos:

- Lista "Mostrar eventos"
- Botón "Aplicar configuración como predeterminada"
- Casilla de verificación "Visualizar información del evento en formato hexadecimal"

Lista "Mostrar eventos:"

En esta lista hay una casilla de verificación para cada clase de evento (predeterminado: Todas las casillas de verificación están activadas.) Si se desactiva una casilla de verificación, dejan

de mostrarse en el área "Eventos" los eventos de la clase de evento correspondiente. Si vuelve a activar la casilla de verificación, se mostrarán de nuevo los eventos correspondientes.

Botón "Aplicar configuración como predeterminada"

Si se hace clic en este botón, la configuración seguirá estando activa las próximas veces que se abra la ficha "Eventos".

Casilla de verificación "Visualizar información del evento en formato hexadecimal"

Si se activa esta casilla de verificación, en la lista de eventos del área "Eventos" se muestra la ID de evento en formato hexadecimal. Si se desactiva esta casilla de verificación, la información del evento se muestra en forma de texto.

Consulte también

Principios básicos del búfer de diagnóstico (Página 2204)

Guardar datos de servicio

Finalidad

Durante el servicio técnico puede ser necesario que el Customer Support de SIEMENS le pida información muy específica acerca del estado de un módulo de su instalación con fines de diagnóstico.

Si se diera un caso de este tipo en su instalación, el Customer Support le solicitará que guarde los datos de servicio del módulo y que le proporcione el archivo generado.

¿Dónde se guardan los datos de servicio de un módulo?

Los datos de servicio de un módulo se guardan en la vista Online y diagnóstico, en la carpeta "Funciones" del grupo "Guardar datos de servicio"

El grupo "Guardar datos de servicio" consta de las áreas siguientes:

- Datos online
- Guardar datos de servicio

Área "Datos online"

Esta área muestra los datos siguientes del módulo:

- Referencia
- Versión del firmware
- Nombre del módulo (asignado durante la configuración del hardware).

- Rack
- Slot

Área "Guardar datos de servicio"

Para generar un archivo con datos de servicio especiales y guardarlo, proceda del siguiente modo:

1. Seleccione el lugar del sistema de archivos donde desea guardar el archivo:
   - Puede utilizar la ruta predeterminada en el campo "Ruta".
   - O puede hacer clic en el botón que tiene tres puntos. Se abrirá un cuadro de diálogo donde debe especificar la ruta deseada y definir el nombre del archivo.
2. Haga clic en el botón "Guardar datos".

8.2.1.5 Modificar las propiedades de un módulo o de la PG/el PC

Cambiar el estado operativo de una CPU

Requisitos

Existe una conexión online con la CPU cuyo estado operativo debe cambiarse.

Procedimiento

Para cambiar el estado operativo de una CPU, proceda del siguiente modo:

1. Active la Task Card "Herramientas online" de la CPU.
2. En la paleta "Panel de control de la CPU", haga clic en el botón "RUN" si desea poner la CPU al estado operativo RUN o en el botón "STOP" si desea poner la CPU al estado operativo STOP.

   Nota

   Sólo está activo el botón que puede seleccionarse en el estado operativo actual de la CPU.

3. Responda la consulta de seguridad haciendo clic en "Aceptar".

O bien:

1. Abra el menú "Online".
2. Seleccione la opción de menú "Arrancar CPU" si desea poner la CPU en el estado operativo RUN, o "Parar CPU" si desea poner la CPU en el estado operativo STOP.

   Nota

   Solo está activa la opción de menú que se puede seleccionar en el estado operativo actual de la CPU.

3. Responda la consulta de seguridad haciendo clic en "Aceptar".

O bien:

1. Haga clic en la barra de funciones en el botón "Arrancar CPU" si desea poner la CPU en el estado operativo RUN, o "Parar CPU" si desea poner la CPU en el estado operativo STOP.

   Nota

   Solo está activo el botón que se puede seleccionar en el estado operativo actual de la CPU.

2. Responda la consulta de seguridad haciendo clic en "Aceptar".

Resultado

La CPU pasará al estado operativo deseado.

Ejecutar un borrado total

Requisitos

- Existe una conexión online con la CPU en la que debe realizarse el borrado total.
- Dicha CPU está en estado operativo STOP.

Nota

Si la CPU todavía está en estado operativo RUN, éste pasará a STOP tras responder afirmativamente a una consulta de seguridad cuando se lance el borrado total.

Procedimiento

Para ejecutar un borrado total en una CPU, proceda del siguiente modo:

1. Active la Task Card "Herramientas online" de la CPU.
2. Haga clic en el botón "MRES" de la paleta "Panel de control de la CPU".
3. Responda la consulta de seguridad haciendo clic en "Aceptar".

Resultado

La CPU pasará al estado operativo STOP en caso necesario y se realizará el borrado total.

Consulte también

Principios básicos del borrado total (Página 1628)

Principios básicos del borrado total (Página 1713)

Principios básicos del borrado total (Página 1907)

Determinar y ajustar la hora de una CPU

¿Dónde están las funciones buscadas?

La determinación y modificación de la hora de una CPU se realiza en el grupo "Ajustar hora" de la carpeta "Funciones" de la vista online y de diagnóstico. Esto sólo es posible si existe una conexión online.

Estructura del grupo "Ajustar hora"

El grupo "Ajustar hora" incluye las áreas siguientes:

- Área para leer y ajustar la hora
- Sistema horario (esta área no existe para la S7-1200 y no se tiene en cuenta aquí.)

Estructura del área de lectura y ajuste de la hora

Esta área está compuesta por los siguientes elementos:

- Hora PG/PC
  Aquí se muestran la zona horaria ajustada y la fecha y la hora actuales de la PG / el PC.
- Hora del módulo
  Aquí se muestran los valores de fecha y hora que ha leído el módulo (p. ej. CPU) convertidos a hora local.
  Si la casilla de verificación "Adoptar de PG / PC" está activada, al hacer clic en el botón "Aplicar" se transferirán al módulo la fecha y la hora PG/PC convertida a hora UTC.
  Si la casilla de verificación "Adoptar de PG / PC" está desactivada, es posible predeterminar la fecha y hora del reloj integrado del módulo. Después de pulsar el botón "Aplicar", la fecha y la hora convertida a UTC se transferirán al módulo.

Actualizar el firmware de un módulo

Actualizar el firmware

Con ayuda de archivos de firmware se puede actualizar el firmware de un módulo.

Requisitos

- El módulo está conectado online.
- El módulo soporta actualización de firmware.
- Para aquellos módulos que necesiten que se aplique tensión de alimentación para una correcta ejecución de la actualización de firmware: la tensión de alimentación del módulo está garantizada. Para más información, consulte la documentación del módulo.

Procedimiento

Para efectuar la actualización del firmware, proceda del siguiente modo:

1. Abra el módulo en la vista Online y diagnóstico.
2. Elija el grupo "Actualización de firmware" de la carpeta "Funciones".

   Nota

   En CPUs S7-1500 este grupo está dividido en "PLC" y "Display"

3. Haga clic en el botón "Examinar" en el área "Actualizador de firmware" para escoger la ruta de los archivos de actualización de firmware.
4. Seleccione unos de estos archivos. En la tabla se muestran todos los módulos para los que es posible efectuar una actualización con el archivo de firmware seleccionado.
5. Opcionalmente: Active la casilla de verificación "Activar firmware tras actualización" para reiniciar automáticamente el módulo después de la carga e iniciar el nuevo firmware.
6. Haga clic en el botón "Iniciar actualización". Si el módulo puede interpretar el archivo seleccionado, éste se carga en el módulo. Si para ello se debe cambiar el estado operativo de la CPU, unos cuadros de diálogo le pedirán que lo haga.

⚠ ADVERTENCIA

Posibilidad de estados no admisibles de la instalación

Una CPU S7-1500 pasa inmediatamente al estado operativo STOP al iniciarse la actualización de firmware, lo cual puede repercutir sobre el funcionamiento de un proceso online o de una máquina.
Un funcionamiento inesperado de un proceso o de una máquina puede dar lugar a la muerte o a lesiones graves y/o daños materiales.

Nota

Tras la actualización del firmware, en la configuración hardware de su proyecto debe sustituirse el módulo afectado por el mismo módulo con la versión actual del firmware. Entonces la configuración volverá a corresponderse con la configuración realmente existente.

Casilla de verificación "Activar firmware tras actualización"

Si no ha activado la casilla de verificación "Activar firmware tras actualización", el firmware anterior permanece activo hasta que se inicialice el módulo (p. ej., con el cambio POWER OFF (desconexión) - POWER ON (conexión)). El firmware nuevo se activa una vez inicializado el módulo.

Si ha activado la casilla de verificación, el módulo se inicializa automáticamente después de que el procedimiento de carga finalice correctamente y, a continuación, pasa a funcionar con el firmware nuevo.

La activación del firmware tras la actualización tiene las consecuencias siguientes:

- Una estación efectúa un rearranque completo. Esto tiene como consecuencia el fallo de todos los módulos de la estación.
- Si la CPU correspondiente se encuentra en el estado operativo RUN, la activación del firmware puede dar lugar a errores de acceso u otros efectos negativos en el programa del usuario, incluido un STOP permanente de la CPU.

Nota

En algunas CPUs, la casilla de verificación "Activar firmware tras actualización" se muestra en color gris y desactivada. En este caso, la CPU se debe rearrancar de nuevo manualmente.

En las CPUs S7-1500, la casilla de verificación "Activar firmware tras actualización" se muestra en color gris y activada. En este caso el firmware nuevo se activará inmediatamente después del proceso de carga.

Consulte también

Sustituir un componente de hardware (Página 528)

Restablecer la configuración de fábrica de una CPU S7-1200

Requisitos

- No hay ninguna Memory Card insertada en la CPU.
- Existe una conexión online con la CPU cuya configuración de fábrica debe restablecerse.
- Dicha CPU está en estado operativo STOP.

Nota

Si la CPU todavía está en estado operativo RUN, se puede pasar a STOP tras responder afirmativamente a una consulta de seguridad al iniciar el proceso de restablecimiento de la configuración de fábrica.

Procedimiento

Para restablecer la configuración de fábrica de una CPU S7-1200, proceda del siguiente modo:

1. Abra la vista Online y diagnóstico de la CPU.
2. Elija el grupo "Restablecer configuración de fábrica" de la carpeta "Funciones".
3. Active el campo de opción "Conservar dirección IP" si desea mantener la dirección IP, o el campo de opción "Restablecer dirección IP" si desea borrarla.

   Nota

   Los dos botones de opción mencionados solo aparecen si el módulo que se desea restablecer tiene la capacidad de elegir entre mantener la dirección IP o borrarla.

4. Haga clic en el botón "Resetear".
5. Responda la consulta de seguridad haciendo clic en "Aceptar".

Resultado

El módulo pasará al estado operativo STOP en caso necesario y se restablecerá la configuración de fábrica. Esto significa que:

- La memoria de trabajo, la memoria de carga interna y todas las áreas de operandos se borrarán.
- Se restablecerán los valores predeterminados de todos los parámetros.
- El búfer de diagnóstico se borrará.
- La hora se inicializará.
- La dirección IP se conservará o borrará, según el ajuste que haya seleccionado.

Restablecer la configuración de fábrica de una CPU S7-1500

Requisitos

- Al abrir desde el contexto de proyecto el restablecimiento de las configuraciones de fábrica debe existir una conexión online de la CPU afectada.
- Dicha CPU está en estado operativo STOP.

  Nota

  Si la CPU todavía está en estado operativo RUN, se puede pasar a STOP tras responder afirmativamente a una consulta de seguridad al iniciar el proceso de restablecimiento de la configuración de fábrica.

Procedimiento

Para restablecer la configuración de fábrica de una CPU S7-1500, proceda del siguiente modo:

1. Abra la vista Online y de diagnóstico de la CPU (desde el contexto de proyecto o mediante "Dispositivos accesibles").
2. Elija el grupo "Restablecer configuración de fábrica" de la carpeta "Funciones".
3. Active el campo de opción "Conservar dirección IP" si desea mantener la dirección IP, o el campo de opción "Restablecer dirección IP" si desea borrarla.

   Nota

   Con "Restablecer dirección IP" se borran todas las direcciones IP, independientemente de cómo se ha establecido la conexión online.

   Si hay una Memory Card insertada, la activación del botón de opción "Restablecer dirección IP" tiene las repercusiones siguientes: las direcciones IP se borran y se restablece la configuración de fábrica de la CPU. Seguidamente, la configuración guardada en la Memory Card (incluida la dirección IP) se transfiere a la CPU (véase abajo). Si la Memory Card se ha formateado o está vacía antes de restablecer la configuración de fábrica, no se transferirá ninguna dirección IP a la CPU.

4. Haga clic en el botón "Resetear".
5. Responda la consulta de seguridad haciendo clic en "Aceptar".

Resultado

El módulo pasará al estado operativo STOP en caso necesario y se restablecerá la configuración de fábrica. Esto significa que:

- La memoria de trabajo, la memoria remanente del sistema y todas las áreas de operandos se borrarán.
- Se restablecerán los valores predeterminados de todos los parámetros.
- El búfer de diagnóstico se borrará.
- La hora se inicializará.
- Los datos I&M se borrarán, excepto los datos I&M0.
- Los contadores de horas de funcionamiento se inicializarán.
- La dirección IP se conservará o borrará, según el ajuste que haya seleccionado.
- Si antes de restablecer la configuración de fábrica había insertada una Memory Card, la configuración (hardware y software) guardada en ella se cargará en la CPU.

Formateado de una Memory Card S7-1500

Requisitos

- Al abrir desde el contexto de proyecto el formateado de la Memory Card debe existir una conexión online de la CPU afectada.
- Dicha CPU está en estado operativo STOP.

Nota

Si la CPU todavía está en estado operativo RUN, se puede pasar a STOP tras responder afirmativamente a una consulta de seguridad al iniciar el proceso de formateado.

Procedimiento

Para formatear una Memory Card S7-1500, proceda del siguiente modo:

1. Abra la vista Online y de diagnóstico de la CPU (desde el contexto de proyecto o mediante "Dispositivos accesibles").
2. Elija el grupo "Formatear Memory Card" de la carpeta "Funciones".
3. Haga clic en el botón "Formatear".
4. Responda la consulta de seguridad haciendo clic en "Aceptar".

Resultado

- Se formatea la Memory Card.
- Temporalmente no se puede acceder a la CPU.
- Se borran los datos de proyecto de la CPU, excepto la dirección IP.
- Si abre el formateado de la Memory Card desde el contexto de proyecto, la vista Online y de diagnóstico permanece abierta; si lo abre mediante "Dispositivos accesibles", se cierra.

Asignación de una dirección IP a un dispositivo PROFINET IO

Principios básicos para asignar una dirección IP a un dispositivo PROFINET IO

Resumen

Todos los dispositivos PROFINET IO operan con el protocolo TCP/IP, y por ello precisan una dirección IP para funcionar con Industrial Ethernet. Una vez que un dispositivo IO ha recibido una dirección IP, se puede acceder a él a través de dicha dirección. Por ejemplo, es posible cargar datos de configuración o realizar un diagnóstico.

Requisitos

- Debe estar establecida la conexión a LAN Ethernet.
- Debe estar accesible la interfaz Ethernet de la programadora o el PC.
- El dispositivo IO que debe recibir una dirección IP debe estar en la misma banda IP que la programadora o el PC.

Abrir la asignación de dirección mediante "Dispositivos accesibles"

Requisitos

- Se ha abierto la vista Online y diagnóstico del dispositivo PROFINET IO desde "Actualizar dispositivos accesibles" (en el árbol del proyecto) o desde "Dispositivos accesibles..." (menú "Online").

Procedimiento

1. Abra el grupo "Asignar dirección IP" de la carpeta "Funciones". En el campo "Dirección MAC" se muestra la dirección MAC del dispositivo PROFINET IO. El botón "Dispositivos accesibles" está en color gris.
2. Introduzca la dirección IP que desee.
3. Introduzca la máscara de subred.
4. Si hay que utilizar un router, active la casilla de verificación "Utilizar router" e introduzca su dirección IP.
5. Haga clic en el botón "Asignar dirección IP".

Resultado

Al dispositivo IO o a su interfaz PROFINET afectada se le asigna la dirección IP con carácter permanente. Dicha dirección se conserva incluso después de un arranque o un fallo de tensión.

Nota

En una CPU S7-1500, del modo descrito anteriormente se puede cambiar también la dirección IP de una interfaz PROFINET aunque anteriormente se haya cargado un proyecto en la CPU a través de dicha interfaz. Se sobrescribirá la dirección IP cargada a través del proyecto.

Consulte también

Remanencia de parámetros de dirección IP y nombres de dispositivo (Página 1501)

Abrir la asignación de dirección desde el contexto de proyecto

Requisitos

- Existe una conexión online con el dispositivo PROFINET IO.
- Se ha abierto la vista Online y diagnóstico del dispositivo PROFINET IO desde el contexto de proyecto.
- El dispositivo PROFINET IO no está asignado a ningún controlador IO.

Procedimiento

1. Abra el grupo "Asignar dirección IP" de la carpeta "Funciones".
2. Haga clic en el botón "Dispositivos accesibles" para determinar los dispositivos accesibles. Nota: En una CPU S7-1500 hay dos entradas porque tiene dos interfaces PROFINET.
3. Seleccione el dispositivo IO. Los campos "Dirección IP" y "Máscara de subred", la casilla de verificación "Utilizar router" y el campo "Dirección del router" están en color gris y contienen las propiedades del nodo a través del que se ha establecido el acceso online actual.
4. Haga clic en el botón "Asignar dirección IP".

Resultado

Al dispositivo IO o a su interfaz PROFINET afectada se le asigna la dirección IP con carácter permanente. Dicha dirección se conserva incluso después de un arranque o un fallo de tensión.

Consulte también

Remanencia de parámetros de dirección IP y nombres de dispositivo (Página 1501)

Asignar nombre de dispositivo PROFINET

Fundamentos para la asignación de nombre a un dispositivo PROFINET IO

Nombre de dispositivo

Para que sea posible acceder a un dispositivo IO desde un controlador IO, es necesario que el dispositivo tenga nombre. En PROFINET se ha optado por este procedimiento porque los nombres son más fáciles de manejar que las complejas direcciones IP.

La asignación de nombre a un dispositivo PROFINET IO es comparable al ajuste de la dirección PROFIBUS en un esclavo DP.

En estado de suministro, los dispositivos IO no tienen nombre. Para que un controlador IO pueda direccionar un dispositivo IO, p. ej. para transferir los datos de configuración (entre ellos la dirección IP) durante el arranque, o para el intercambio de datos útiles en modo de operación

cíclico, es necesario que previamente se le haya asignado al dispositivo un nombre de dispositivo con la programadora o el PC.

Reglas para nombres de dispositivo

Los nombres de dispositivo están sujetos a las siguientes restricciones:

- Limitación a 240 caracteres en total (letras minúsculas, cifras, guión o punto)
- Los componentes del nombre de dispositivo, es decir, las cadenas de caracteres entre dos puntos, no deben superar los 63 caracteres de longitud.
- No se admiten caracteres especiales como vocales acentuadas o con diéresis, los paréntesis, caracteres de subrayado, barras, espacios en blanco, etc. El único carácter especial permitido es el guión.
- El nombre de dispositivo no debe empezar ni acabar por el carácter "-".
- El nombre de dispositivo no debe empezar con una cifra.
- El nombre de dispositivo no debe tener la forma n.n.n.n (n = 0, ... 999).
- El nombre de dispositivo no debe empezar con la cadena de caracteres "port-xyz" o "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0, ... 9).

¿Dónde se encuentra la función buscada?

El nombre de un dispositivo PROFINET IO se puede asignar en los siguientes puntos:

- En su vista Online y diagnóstico, en el grupo "Asignar nombre" de la carpeta "Funciones". La forma de la interfaz de usuario de este grupo depende del modo en que se haya abierto la vista Online y diagnóstico:
  - Acceso mediante "Dispositivos accesibles"
  - Llamada desde el contexto del proyecto
- En el cuadro de diálogo "Asignar nombre de dispositivo PROFINET".

Consulte también

Asignar un nombre en la vista Online y diagnóstico, abierta desde "Dispositivos accesibles" (Página 2198)

Asignar un nombre en la vista Online y diagnóstico, abierta desde el contexto del proyecto (Página 2198)

Asignar un nombre en el cuadro de diálogo "Asignar nombre de dispositivo PROFINET" (Página 2200)

Asignar un nombre en la vista Online y diagnóstico, abierta desde "Dispositivos accesibles"

Requisitos

- Se ha abierto la vista Online y diagnóstico del dispositivo PROFINET IO desde "Actualizar dispositivos accesibles" (en el árbol del proyecto) o desde "Dispositivos accesibles..." (menú "Online").

Procedimiento

1. Abra el grupo "Asignar nombre" de la carpeta "Funciones". En el campo "Tipo" se muestra el tipo de módulo del dispositivo PROFINET IO.
2. Introduzca en el campo de entrada "Nombre de dispositivo PROFINET" el nombre del dispositivo deseado.
3. Opcionalmente: haga clic en la casilla de verificación "Prueba de parpadeo de LED" para efectuar un test de parpadeo de LEDs en el dispositivo PROFINET IO. Así se comprueba si el usuario se encuentra en el dispositivo IO deseado.

   Nota

   El test de parpadeo de LED no es compatible con todos los dispositivos PROFINET IO.

   El test de parpadeo de LED continúa hasta que es interrumpido por el usuario. Puede interrumpirse, p. ej., desactivando la casilla de verificación "Prueba de parpadeo de LED" o cerrando la vista Online y diagnóstico.
4. Haga clic en el botón "Asignar nombre".

Resultado

El nombre indicado se asigna al dispositivo PROFINET IO.

Asignar un nombre en la vista Online y diagnóstico, abierta desde el contexto del proyecto

Requisitos

- Se ha abierto la vista Online y diagnóstico del dispositivo PROFINET IO desde el contexto de proyecto.
- Se puede acceder al dispositivo PROFINET IO a través de una interfaz PG/PC como mínimo.

Procedimiento

1. Abra el grupo "Asignar nombre" de la carpeta "Funciones". En la lista desplegable "Nombre de dispositivo PROFINET" se muestra el nombre que existe actualmente en el proyecto offline y, en el campo "Tipo", se muestra el tipo de módulo del dispositivo PROFINET IO.

   Nota

   En las CPU con varias interfaces PROFINET se muestran los nombres de todas las interfaces PROFINET ya existentes en el proyecto offline.

2. En caso necesario, seleccione otro nombre en la lista desplegable.

   Nota

   En los pasos de 3 a 5 se determinan los dispositivos IO presentes en la subred PROFINET.

3. En la lista desplegable "Interfaz PG/PC para la asignación", seleccione la interfaz PG/PC a través de la cual desea establecer la conexión online.
4. Opcionalmente: Con las tres casillas de verificación, realice una selección de entre todos los dispositivos IO presentes online.
5. Haga clic en el icono para determinar los dispositivos IO presentes en la subred PROFINET. Tras ello, la tabla se actualiza.
6. Seleccione en la tabla el dispositivo IO deseado.
7. Opcionalmente: haga clic en la casilla de verificación "Prueba de parpadeo de LED" para efectuar un test de parpadeo de LEDs en el dispositivo PROFINET IO. Así se comprueba si el usuario se encuentra en el dispositivo IO deseado.

   Nota

   El test de parpadeo de LED no es compatible con todos los dispositivos PROFINET IO.

   El test de parpadeo de LED continúa hasta que es interrumpido por el usuario. Puede interrumpirse, p. ej. desactivando la casilla de verificación "Prueba de parpadeo de LED", seleccionando otro dispositivo IO de la tabla o cerrando la vista Online y diagnóstico.
8. Haga clic en el botón "Asignar nombre".

Resultado

El nombre seleccionado se asigna al dispositivo PROFINET IO o a una de sus interfaces PROFINET.

Asignar un nombre en el cuadro de diálogo "Asignar nombre de dispositivo PROFINET"

Requisitos

- Ha abierto el cuadro de diálogo "Asignar nombre de dispositivo PROFINET" desde la vista de redes (desde el menú contextual de una conexión PN/IE).
- Se puede acceder al dispositivo PROFINET IO a través de una interfaz PG/PC como mínimo.

Procedimiento

1. En la lista desplegable "Nombre de dispositivo PROFINET" se muestra lo siguiente:
   – el nombre existente actualmente en el proyecto online para la interfaz con la que se ha abierto el cuadro de diálogo
   – los nombres de los dispositivos IO que están conectados mediante dicha interfaz

   en el campo "Tipo" se muestra el tipo de módulo del dispositivo PROFINET IO.
   En caso necesario, seleccione otro nombre en la lista desplegable.

   Nota

   En los pasos de 2 a 4 se determinan los dispositivos IO presentes en la subred PROFINET.

2. En la lista desplegable "Interfaz PG/PC para la asignación", seleccione la interfaz PG/PC a través de la cual desea establecer la conexión online.
3. Opcionalmente: Con las tres casillas de verificación, realice una selección de entre todos los dispositivos IO presentes online.
4. Haga clic en el icono para determinar los dispositivos IO presentes en la subred PROFINET. Tras ello, la tabla se actualiza.
5. Seleccione en la tabla el dispositivo IO deseado.
6. Opcionalmente: haga clic en la casilla de verificación "Prueba de parpadeo de LED" para efectuar un test de parpadeo de LEDs en el dispositivo PROFINET IO. Así se comprueba si el usuario se encuentra en el dispositivo IO deseado.

   Nota

   El test de parpadeo de LED no es compatible con todos los dispositivos PROFINET IO.

   El test de parpadeo de LED continúa hasta que es interrumpido por el usuario. Puede interrumpirse, p. ej., desactivando la casilla de verificación "Prueba de parpadeo de LED" o seleccionando otro dispositivo IO de la tabla.
7. Haga clic en el botón "Asignar nombre".

Resultado

El nombre seleccionado se asigna al dispositivo PROFINET IO o a la interfaz con la que se ha abierto el cuadro de diálogo.

Calibración de un módulo analógico S7-1500

Calibración de un módulo analógico S7-1500: resumen

¿Cómo se calibra un módulo analógico S7-1500?

La calibración de un módulo analógico S7-1500 se lleva a cabo en su vista Online y de diagnóstico, en el grupo "Calibrar" de la carpeta "Funciones".

Resumen de las funciones de calibración disponibles

En el grupo "Calibración" es posible ejecutar las siguientes funciones en un módulo analógico S7-1500:

- Determinar la calibración actual de todos los canales
- Calibración de un canal
- Cancelación de un proceso de calibración en curso
- Restablecer la configuración de fábrica en la calibración de un canal

Requisitos para la función de calibración descrita a continuación

Para la función de calibración que se describe a continuación se presuponen los requisitos siguientes:

- Se ha abierto la vista Online y de diagnóstico desde el contexto de proyecto (es decir, no desde el árbol del proyecto ni desde el menú "Online").
- Existe una conexión online con el módulo analógico que va a calibrarse.
- Las configuraciones offline y online son idénticas.

Calibración de un módulo analógico S7-1500

Resumen de la calibración de un canal de un módulo analógico S7-1500

La calibración de un canal de un módulo analógico S7-1500 consta de los siguientes pasos:

1. Inicio del proceso de calibración
2. Ejecute todos los pasos del proceso de calibración menos el primero y el último.
3. Finalización del proceso de calibración

A continuación se describe con más detalle.

Requisitos

- Se ha abierto la vista Online y de diagnóstico del módulo analógico S7-1500 desde el contexto del proyecto y el usuario está en el grupo "Calibración" de la carpeta "Funciones".
- La CPU correspondiente está en modo online.
- Hay un proceso de calibración en curso en el módulo analógico (cuando se desea iniciar el proceso de calibración) o se ha realizado con éxito el último paso iniciado (cuando se desea continuar o cerrar el proceso de calibración).

Procedimiento para iniciar el proceso de calibración

Para iniciar el proceso de calibración, proceda del siguiente modo:

1. Seleccione en la tabla general la línea correspondiente al canal que va a calibrarse.
2. Haga clic en el botón "Iniciar calibración manual".

La interfaz de usuario cambia a la apariencia siguiente:

- Se desactivan la tabla general y los botones "Iniciar calibración manual" y "Restablecer configuración de fábrica"
- Se activa la visualización de etapas y se indica el número de la etapa actual y de la última etapa.
- Se activa el campo "Comando", que indica lo que el usuario debe hacer en la etapa de calibración siguiente.
- El campo "Estado" se activa y muestra el estado actual del proceso de calibración, p. ej. "Calibración iniciada correctamente".
- El campo "Valor medido" se activa. En un módulo de entrada aquí se muestra un valor, mientras que en un módulo de salida debe introducirse un valor.
- Se activa el botón "Cancelar".
- Se activa el botón "Siguiente". Puede ejecutarse la siguiente etapa del proceso de calibración.

Procedimiento para las etapas segunda a penúltima del proceso de calibración

Proceda del siguiente modo:

1. Haga clic en el botón "Siguiente".

Se actualizan los campos de la interfaz de usuario descritos más arriba.

Procedimiento para la última etapa del proceso de calibración

Proceda del siguiente modo:

1. Haga clic en el botón "Siguiente".

La interfaz de usuario cambia a la apariencia siguiente:

- Se activa la tabla sinóptica.
- Se actualiza la visualización del canal calibrado.

- Se activan los botones "Iniciar calibración manual" y "Restablecer configuración de fábrica".
- Se desactiva la visualización de etapas y los números de la etapa actual y de la última etapa están vacíos.
- El campo "Comando" se desactiva y queda vacío.
- Se desactiva el campo "Estado", que muestra el último estado del proceso de calibración, p. ej. "Calibración finalizada correctamente".
- El campo "Valor medido" se desactiva y queda vacío.
- Se desactiva el botón "Cancelar".
- Se desactiva el botón "Siguiente".

Aparición de un error

Si durante el proceso de calibración se produce un error, el módulo interrumpe el proceso. El canal que se estaba calibrando mantiene la misma configuración que tenían antes de iniciarse el proceso de calibración.

Una vez ocurrido el error, la interfaz de usuario tiene la misma apariencia que antes de iniciarse el proceso de calibración, excepto en el campo "Estado". El campo "Estado" muestra el error que ha detectado el módulo en el proceso de calibración.

Cancelación de un proceso de calibración en curso de un módulo analógico S7-1500

Requisitos

- Se ha abierto la vista Online y de diagnóstico del módulo analógico S7-1500 desde el contexto del proyecto y el usuario está en el grupo "Calibración" de la carpeta "Funciones".
- La CPU correspondiente está en modo online.
- Hay un proceso de calibración en curso en el módulo analógico.

Procedimiento

Para cancelar un proceso de calibración en curso, proceda del siguiente modo:

1. Haga clic en el botón "Cancelar".

Resultado

El proceso de calibración en curso se interrumpe, y el canal que se estaba calibrando mantiene la misma configuración que tenían antes de iniciarse el proceso de calibración.

En la interfaz de usuario se desactivan todos los elementos de manejo hasta que concluya la cancelación. La interfaz de usuario tiene la misma apariencia que antes de iniciarse el proceso de calibración, excepto en el campo "Estado". El campo "Estado" muestra el resultado de la cancelación.

Restablecer la configuración de fábrica de un módulo analógico S7-1500

Requisitos

- Se ha abierto la vista Online y de diagnóstico del módulo analógico S7-1500 desde el contexto del proyecto y el usuario está en el grupo "Calibración" de la carpeta "Funciones".
- La CPU correspondiente está en modo online.

Procedimiento

Para restablecer la configuración de fábrica de un canal de un módulo analógico S7-1500, proceda del siguiente modo:

1. En la tabla sinóptica seleccione la línea correspondiente al canal en el que se hará el restablecimiento.
2. Haga clic en el botón "Restablecer configuración de fábrica".

Resultado

En la interfaz de usuario se desactivan todos los elementos de manejo hasta que concluya el proceso de restablecimiento. La interfaz de usuario tiene la misma apariencia que antes de iniciarse el proceso de restablecimiento, excepto en el campo "Estado". El campo "Estado" muestra el resultado del proceso de restablecimiento.

8.2.1.6 Diagnóstico en estado operativo STOP

Principios básicos del búfer de diagnóstico

Función

El sistema operativo de la CPU registra en el búfer de diagnóstico los errores detectados por la CPU y los módulos aptos para diagnóstico siguiendo el orden de aparición. Aquí se incluyen los eventos siguientes, entre otros:

- Todos los cambios de estado operativo de la CPU (p. ej. conexión (POWER ON), transición al estado operativo STOP, transición al estado operativo RUN)
- Todas las alarmas de proceso y diagnóstico

La primera entrada contiene el evento más reciente. Las entradas del búfer de diagnóstico se guardan de forma permanente: se conservan incluso si falla la fuente de alimentación y sólo se borran al restablecer la configuración de fábrica de la CPU.

Una entrada del búfer de diagnóstico contiene los elementos siguientes:

- Sello de tiempo
- ID de error
- Otra información específica de la ID de error

Ventajas del búfer de diagnóstico

El búfer de diagnóstico ofrece las ventajas siguientes:

- Una vez la CPU ha pasado al estado operativo STOP, es posible evaluar los últimos eventos anteriores al STOP y, de este modo, aislar y determinar la causa del mismo.
- Esto permite detectar y solucionar con más rapidez las causas de errores, con lo que se aumenta la disponibilidad de la instalación.
- Es posible evaluar el comportamiento dinámico de la instalación y, seguidamente, optimizarlo.

Organización del búfer de diagnóstico

El búfer de diagnóstico es un búfer en anillo. El número máximo de entradas es de 50 en las CPUs S7-1200. Si el búfer de diagnóstico está lleno y hay que realizar otra entrada, todas las entradas existentes se corren un puesto (lo que supone que la entrada más antigua se borra) y la entrada nueva se registra en el primer lugar, que ha quedado libre (principio FIFO: first in, first out).

Evaluación del búfer de diagnóstico

Existen las posibilidades siguientes de acceder al contenido del búfer de diagnóstico:

- Desde la vista online y de diagnóstico

Evaluando los eventos anteriores al evento de error (p. ej. transición de estado operativo a STOP) se formará una idea de la causa posible y será posible aislarla o especificarla con más precisión (en función del tipo de error).

Lea detenidamente la información detallada de los eventos y utilice el botón "Ayuda del evento" para obtener más información y las posibles causas de las diferentes entradas.

Nota

Para poder utilizar adecuadamente las indicaciones de tiempo de las entradas del búfer de diagnóstico en instalaciones de tiempo crítico, es recomendable comprobar de vez en cuando la fecha y hora de la CPU y corregirlas en caso necesario.

Otra posibilidad consiste en sincronizar la hora a través de un servidor de hora NTP.

Consulte también

Restablecer la configuración de fábrica de una CPU S7-1200 (Página 2191)

Determinar la causa de STOP de una CPU (Página 2206)

Determinar y ajustar la hora de una CPU (Página 2189)

Parametrizar el reloj (Página 1653)

Determinar la causa de STOP de una CPU

Requisitos

La CPU que debe analizarse está en estado operativo STOP.

Procedimiento

Para determinar la causa de STOP de una CPU, proceda del siguiente modo:

1. Abra la vista online y de diagnóstico de la CPU.
2. Elija el grupo "Búfer de diagnóstico" de la carpeta "Diagnóstico".
3. Evalúe los eventos anteriores a la transición al estado operativo STOP. Con ello se formará una idea de la causa posible y será posible aislarla o especificarla con más precisión (en función del tipo de error).
   Lea detenidamente la información detallada de los eventos y utilice el botón "Ayuda del evento" para obtener más información y las posibles causas de las diferentes entradas.

Resultado

Se ha podido aislar y determinar con más precisión la causa de STOP de la CPU.

Nota

Si el análisis no le sirve de ayuda, póngase en contacto con el Customer Support. En este caso, guarde el contenido del búfer de diagnóstico en un archivo de texto con el botón "Guardar como" y entréguelo al Customer Support.

Consulte también

Leer el búfer de diagnóstico de una CPU (Página 2182)

8.2.1.7 Accesos online en la vista online y de diagnóstico

Ver el estado de la conexión online

Requisitos

- Se puede acceder al dispositivo en cuestión por lo menos a través de una interfaz PG/PC.

Procedimiento

1. Abra la vista online y de diagnóstico del dispositivo cuyo estado de conexión online desee ver.
2. Seleccione el grupo "Accesos online".

Nota

El grupo "Accesos online" solo existe en las CPUs y en algunos CPs. Si se ha llamado la vista Online y diagnóstico mediante la función "Mostrar/actualizar dispositivos accesibles", no se mostrará.

Resultado

En el área "Estado" se muestra, en gráficos y en palabras, el estado de la conexión online.

Definir interfaz PG/PC, establecer conexión online

Requisitos

- Se puede acceder al dispositivo en cuestión por lo menos a través de una interfaz PG/PC.
- En este momento no hay conexión online con el dispositivo correspondiente.

Procedimiento

1. Abra la vista Online y diagnóstico del dispositivo con el que desea establecer una conexión online.
2. Seleccione el grupo "Accesos online" y dentro de él, el área "Accesos online".

Nota

El grupo "Accesos online" existe en las CPU y en algunos CP. Si se ha abierto la vista Online y diagnóstico mediante la función "Mostrar/actualizar dispositivos accesibles", no se mostrará.

3. Si ya ha existido anteriormente una conexión online con el dispositivo, en las listas desplegables aparecerán de modo predeterminado los valores correspondientes a dicha conexión online. En este caso puede continuarse con la última etapa de esta instrucción, siempre que entre tanto no se haya cambiado la dirección IP en la vista Online y de diagnóstico.
4. Seleccione el tipo de interfaz en la lista desplegable "Tipo de interfaz PG/PC".
   En función de esta selección, aparecen en la lista desplegable "Interfaz PG/PC para el acceso online" solo aquellas interfaces PG/PC que coincidan con el tipo de interfaz seleccionado.
5. En la lista desplegable "Interfaz PG/PC para el acceso online", seleccione la interfaz PG/PC a través de la cual desea establecer la conexión online.
6. Opcionalmente: Para modificar las propiedades del CP correspondiente, haga clic en el botón "Propiedades".
7. En la lista desplegable "Conexión con subred", seleccione la subred a través de la cual el dispositivo está conectado con la interfaz PG/PC.

   Nota

   La interfaz de programadora o PC tiene que estar conectada con una interfaz de un dispositivo.

   Si se desea acceder exclusivamente a este dispositivo, seleccione en la lista desplegable el ajuste "Directamente en el slot <nombre de interfaz>".

   Por el contrario, si desea acceder a otro dispositivo mediante routing, cree en la configuración hardware una subred en esta interfaz y seleccione a continuación la subred en la lista desplegable.

8. Si el dispositivo es accesible a través de un gateway, seleccione en la lista desplegable "1.er gateway" el gateway que conecta entre sí las subredes implicadas.
9. Introduzca en el campo de entrada "Dirección del dispositivo" la dirección IP del dispositivo con el que desee establecer una conexión online.

   Nota

   Con CPUs que tienen varias direcciones IP seleccione en la lista desplegable "Dirección del dispositivo" la dirección IP de la interfaz PROFINET a través de la que desee establecer la conexión online.

10. Alternativa: Haga clic en el botón "Mostrar dispositivos accesibles" y seleccione en la lista el dispositivo con el que desea establecer una conexión online.
11. Haga clic en el botón "Establecer conexión online".

Resultado

Se establece la conexión online con el dispositivo deseado.

Deshacer la conexión online

Requisitos

- En este momento existe una conexión online con el dispositivo correspondiente.

Procedimiento

1. Abra la vista online y de diagnóstico del dispositivo cuya conexión online desea desactivar.
2. Seleccione el grupo "Accesos online" y dentro de él, el área "Accesos online".

   Nota

   El grupo "Accesos online" solo existe en las CPUs y en algunos CPs. Si se ha llamado la vista Online y diagnóstico mediante la función "Mostrar/actualizar dispositivos accesibles", no se mostrará.

3. Haga clic en el botón "Deshacer conexión online".

Resultado

Se deshace la conexión online con el dispositivo deseado.

Realizar el test de parpadeo con un dispositivo conectado online

Requisitos

- En este momento existe una conexión online con el dispositivo correspondiente.
- La función FORZADO PERMANENTE no está activa.

Procedimiento

1. Abra la vista online y de diagnóstico del dispositivo para el que desee realizar el test de parpadeo.
2. Seleccione el grupo "Accesos online" y dentro de él, el área "Estado".

   Nota

   El grupo "Accesos online" solo existe en las CPUs y en algunos CPs. Si se ha llamado la vista Online y diagnóstico mediante la función "Mostrar/actualizar dispositivos accesibles", no se mostrará.

3. Active la casilla de verificación "Prueba de parpadeo de LED".

Resultado

- En una CPU S7-1200 parpadean los LEDs RUN/STOP, ERROR y MAINT.
- En una CPU S7-1500 parpadean los LEDs RUN/STOP, ERROR y MAINT.
- En una CPU S7-300 o S7-400 parpadea el LED FRCE.

Los LEDs parpadean hasta que el test de parpadeo es cancelado por el usuario. Puede cancelarse, p. ej. desactivando la casilla de verificación "Prueba de parpadeo de LED", cambiando a otro grupo de la vista Online y de diagnóstico o cambiando la configuración en el área "Accesos online".

8.2.1.8 Comprobar fallos en subredes PROFIBUS DP

Principios básicos del repetidor de diagnóstico

¿Qué es el repetidor de diagnóstico?

El repetidor de diagnóstico es un repetidor capaz de vigilar durante el funcionamiento un segmento de una subred RS 485-PROFIBUS (cable de cobre) y comunicar al maestro DP mediante telegrama de diagnóstico los fallos presentes en el cable.

Gracias al diagnóstico de cable durante el funcionamiento, el repetidor de diagnóstico permite detectar, ubicar y visualizar a tiempo los fallos en el cable. Esto permite detectar a tiempo los fallos de la instalación y minimizar así las paradas.

Funcionamiento del repetidor de diagnóstico

El repetidor de diagnóstico puede efectuar un diagnóstico de cable en los segmentos DP2 y DP3 porque está provisto de un contacto de medición para dichos segmentos.

El diagnóstico de cable se efectúa en dos pasos:

- 1. Paso: Determinación de topología
  Para iniciar la determinación de topología, debe llamarse en el programa la instrucción avanzada "DP_TOPOL".
  El repetidor de diagnóstico determina las direcciones PROFIBUS y la distancia de los dispositivos, y crea una tabla de topología.
- 2. Paso: Determinación de los puntos de fallo
  El repetidor de diagnóstico comprueba los cables durante el funcionamiento del bus. Determina la distancia del punto de fallo, averigua la causa del fallo y emite un mensaje de diagnóstico con indicación relativa de la ubicación del fallo.

Visualización de información detallada sobre el punto de fallo detectado

En la vista oline y de diagnóstico del repetidor de diagnóstico se muestra información detallada sobre el punto de fallo detectado:

- Mediante iconos
- Mediante visualización gráfica y de texto

Consulte también

Visualizar el estado del diagnóstico de segmento mediante iconos (Página 2211)

Visualizar el estado del diagnóstico de segmento mediante gráficos y texto (Página 2211)

Visualizar el estado del diagnóstico de segmento mediante iconos

¿Dónde está la información buscada?

Los iconos que muestran el estado del diagnóstico de segmento se encuentran en el siguiente lugar:

- En la ventana de navegación de la vista online y de diagnóstico del correspondiente repetidor de diagnóstico, en la carpeta desplegada "Diagnóstico de segmento"

El icono de diagnóstico correspondiente al segmento aparece a continuación de la denominación del segmento. Debe tenerse en cuenta que solo se muestran los fallos de cable correspondientes a los segmentos DP2 y DP3. Los segmentos DP1 y PG no muestran los fallos en forma de icono de diagnóstico, sino que comunican solo unos pocos fallos de bus.

Iconos de diagnóstico

La tabla siguiente muestra los iconos posibles y su significado.

Icono	Significado
	Segmento sin fallos
	Segmento con fallos
	Segmento desconectado

Visualizar el estado del diagnóstico de segmento mediante gráficos y texto

¿Dónde se muestra el estado del diagnóstico de segmento mediante gráficos y texto?

La visualización del estado del diagnóstico de segmento mediante gráficos y texto se realiza en la vista online y de diagnóstico del correspondiente repetidor de diagnóstico, carpeta "Diagnóstico de segmento", grupos "DP1", "DP2", "DP3" y "PG".

Estructura de los grupos "DP1", "DP2", "DP3" y "PG"

Los grupos "DP1", "DP2", "DP3" y "PG" están formados por los siguientes elementos:

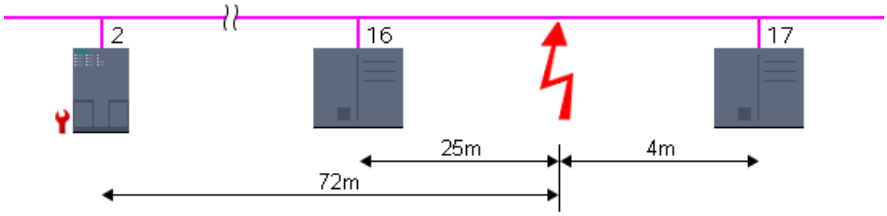
- Campo "Ubicación del fallo"
- Campo "Fallo"
- Campo "Solución"

- Botón "Ayuda del evento"
- Botón "Congelar lista" y "Descongelar"

Campo "Ubicación del fallo"

Aquí se muestra gráficamente la ubicación del fallo, a condición de que el repetidor de diagnóstico sea capaz de determinarla.

La imagen siguiente muestra un ejemplo de un fallo de cable producido en el segmento DP2.



En el ejemplo, el repetidor de diagnóstico tiene la dirección PROFIBUS 2, y se ha producido un fallo de cable entre los dispositivos con las direcciones PROFIBUS 16 y 17. El fallo de cable se encuentra a 25 m de distancia del dispositivo 16, a 4 m del dispositivo 17 y a 72 m del repetidor de diagnóstico.

Campo "Fallo"

Aquí se describe el fallo en lenguaje natural.

Campo "Solución"

Aquí se encuentran las instrucciones para la eliminación del fallo.

Botón "Ayuda del evento"

Al hacer clic en este botón, se describe con más detalle el fallo producido y, en su caso, se proporciona información adicional para su eliminación.

Botón "Congelar lista" y "Descongelar"

El botón "Congelar lista" o "Descongelar" solo está activo si existe una conexión online con el repetidor de diagnóstico.

El ajuste predeterminado es "Congelar visualización".

Al hacer clic en el botón "Congelar lista" sucede lo siguiente:

- Se congela la visualización actual del diagnóstico de segmento.
- La rotulación del botón cambia a "Descongelar".

Cuando la visualización está congelada y se pulsa el botón "Descongelar", sucede lo siguiente:

- Se actualiza de nuevo la visualización del diagnóstico de segmento.
- La rotulación del botón cambia a "Congelar lista".

8.2.2 Diagnosticar conexiones

8.2.2.1 Vista general del diagnóstico de conexiones

Principios básicos

En lo sucesivo se denomina "diagnóstico de conexiones" al diagnóstico de conexiones de comunicación.

El diagnóstico de conexiones se inicia cada vez que se establece la conexión online con un módulo (CPU o CP) que participa en uno o varios servicios de comunicación. La actualización del estado de conexión se efectúa automáticamente en segundo plano.

En las conexiones unilaterales, debe existir conexión online con el interlocutor que ha establecido la conexión.

En las conexiones bilaterales, se deben distinguir los dos casos siguientes:

- Cuando existe una conexión online exactamente a un punto final de la conexión, sólo se puede diagnosticar la parte de la conexión correspondiente a ese punto final de la conexión.
- Cuando existe una conexión online con ambos puntos finales de la conexión, se pueden diagnosticar las dos partes de la conexión (y, por tanto, la conexión completa).

Posibilidades básicas del diagnóstico de conexiones

Pueden diagnosticarse conexiones del modo que se indica a continuación:

- Mediante la visualización del estado de conexión con iconos
  Esta visualización se realiza en la tabla de conexiones.
- Mediante diagnóstico de conexiones detallado
  Este diagnóstico se realiza en el área "Diagnóstico > Información de conexión" de la ventana de inspección.

Requisitos para el diagnóstico de conexiones descrito a continuación

En la tabla de conexiones se pueden visualizar los detalles de todas las conexiones creadas en el proyecto (opción predeterminada) o sólo los detalles de las conexiones seleccionadas.

Para el diagnóstico de conexiones descrito a continuación es imprescindible que se visualicen los detalles de las conexiones seleccionadas. Desactive para ello la opción "Mostrar todas las conexiones" del menú contextual.

Consulte también

Trabajar con la tabla de conexiones (Página 633)

8.2.2.2 Ver el estado de conexión mediante iconos

Contenido de la tabla de conexiones cuando no hay ninguna conexión online establecida

- En el contexto de una CPU o un CP, si no está establecida la conexión online, se muestran en la tabla de conexiones las conexiones de comunicación configuradas offline (incluidas sus propiedades).

Contenido de la tabla de conexiones cuando hay una conexión online establecida

Una vez establecida la conexión online, a las propiedades de las comunicaciones enumeradas offline se añaden los iconos de diagnóstico para el estado de conexión (columna "Estado online").

Además, ahora la tabla de conexiones contiene entradas para todas las conexiones de comunicación que solo existen online (p. ej. conexiones para las instrucciones de Open User Communication, conexiones PG y OP conexiones para acceso a servidor web).

En el caso de las conexiones que sólo existen online u offline, el símbolo de diagnóstico se combina en la parte inferior derecha con un símbolo adicional más pequeño para el estado de comparación.

Iconos de diagnóstico para conexiones de comunicación

La tabla siguiente muestra los iconos de diagnóstico para las conexiones de comunicación.

Icono	Significado
	Conexión establecida
	Conexión no establecida o en vías de establecimiento
	Conexión no disponible

Iconos de diagnóstico para el estado de comparación

Los iconos de diagnóstico para las conexiones de comunicación pueden combinarse en la parte inferior derecha con iconos adicionales más pequeños que indican el resultado de la comparación online-offline. La tabla siguiente muestra los símbolos de comparación posibles y su significado.

Icono	Significado
	La conexión solo existe online
	La conexión solo existe offline

8.2.2.3 Diagnosticar conexiones detalladamente

Diagnosticar conexiones detalladamente - Resumen

¿Cómo se diagnostican las conexiones detalladamente?

El diagnóstico detallado de conexiones se efectúa en el área "Diagnóstico > Información de conexión" de la ventana de inspección.

¿Cómo se abre el área "Diagnóstico > Información de conexión" de la ventana de inspección?

Para abrir la ficha "Información de conexión" de la ventana de inspección, existen las siguientes posibilidades:

- Seleccione en la tabla de conexiones la línea que corresponde a la conexión en cuestión. A continuación, haga clic en la ventana de inspección en las fichas "Diagnóstico" e "Información de conexión", sucesivamente.
- Haga doble clic en la tabla de conexiones en el icono de diagnóstico de la conexión en cuestión.
- Se encuentra en el editor de programación, en una instrucción de la comunicación S7 o de la Open User Communication. Haga doble clic sobre el icono de diagnóstico de la instrucción (estetoscopio).

Estructura del área "Diagnóstico > Información de conexión" de la ventana de inspección.

La condición previa para ello es que el contenido de la ficha "Información de conexión" esté rellenado, es decir, que exista una conexión online con por lo menos un punto final de la conexión en cuestión.

Si el módulo está seleccionado (vista de redes), la ficha contiene el siguiente grupo:

- Recursos de conexión (con S7-1200 y S7-1500)

Si la conexión está seleccionada (tabla de conexiones), contiene los siguientes grupos:

- Detalles de conexión
- Detalles de la dirección de la conexión (S7-1200 y S7-1500)

Determinación de recursos de conexión online con S7-1200

¿Dónde se determinan los recursos de conexión online?

Los recursos de conexión online se obtienen del grupo "Recursos de conexión". Este grupo se encuentra en el área "Diagnóstico > Información de conexión" de la ventana de inspección. Solo se muestra si en la vista de redes se ha seleccionado un módulo con el que hay una conexión online.

Número de recursos de conexión

- Número máximo: Indica el número máximo posible de recursos de conexión disponibles del módulo.
- No ocupados: Indica cuántos recursos de conexión están todavía sin ocupar. En caso de que haya recursos de conexión ya reservados para determinados tipos de comunicación, los recursos de conexión no ocupados no podrán utilizarse siempre para cualquier tipo de conexión.

Recursos de conexión reservados y ocupados actualmente

A continuación se muestran los recursos de conexión reservados por el módulo y ocupados actualmente para los tipos de comunicación que se mencionan.

Tipo de comunicación	Significado
Comunicación PG	Recursos para conexiones entre el módulo y programadoras (p. ej. para el establecimiento de conexiones desde el árbol de proyecto, para diagnóstico online, etc.)
Comunicación HMI	Recursos para conexiones entre el módulo y dispositivos HMI
Open User Communication	Recursos para conexiones de instrucciones de Open User Communication
Comunicación S7	Recursos para conexiones S7 configuradas que permiten intercambiar datos en el programa de usuario mediante la llamada de instrucciones
Otra comunicación	Indica otros recursos de conexión ocupados para los que no hay recursos de conexión reservados

Determinación de recursos de conexión online con S7-1500

¿Dónde se determinan los recursos de conexión online?

Los recursos de conexión online se obtienen del grupo "Recursos de conexión". Este grupo se encuentra en el área "Diagnóstico > Información de conexión" de la ventana de inspección. Solo se muestra si en la vista de redes se ha seleccionado un módulo con el que hay una conexión online.

Descripción de la visualización detallada de los recursos de conexión

La visualización detallada de los recursos de conexión incluye:

- el número de recursos de conexión disponibles
- el número de recursos de conexión configurados
- el número de recursos de conexión que continúan disponibles.

Se pueden consultar sus descripciones aquí (Página 601) .

Determinar los detalles de conexión

¿Dónde se determinan los detalles de conexión?

Los detalles de conexión están indicados en el grupo "Detalles de conexión". Este grupo se encuentra en el área "Diagnóstico > Información de conexión" de la ventana de inspección.

¿Cuándo está ocupado el grupo "Detalles de conexión"?

Para que el grupo "Detalles de conexión" de la ficha "Información de conexión" esté ocupado, deben cumplirse los siguientes requisitos:

- Existe una conexión online con el punto final de la conexión en cuestión.
- Se ha seleccionado una fila en la tabla de conexiones.

Estructura del grupo "Detalles de conexión"

El grupo "Detalles de conexión" incluye los elementos siguientes:

- ID local (hex)
- Tipo de conexión (S7-1200 y S7-1500)
- Protocolo
- Estado de conexión: Icono y descripción
- Detalles
- Último cambio de estado (solo para S7-300 y S7-400)

Determinar los detalles de dirección de una conexión

¿Dónde se determinan los detalles de dirección de una conexión?

Los detalles de conexión están indicados en el grupo "Detalles de dirección de la conexión". Este grupo se encuentra en el área "Diagnóstico > Información de conexión" de la ventana de inspección.

¿En qué CPUs existe el grupo "Detalles de dirección de la conexión"?

El grupo "Detalles de las direcciones de la conexión" de la ficha "Información de la conexión" existe solo con las CPUs S7-1200 y S7-1500.

¿Cuándo está ocupado el grupo "Detalles de dirección de la conexión"?

Para que el grupo "Detalles de dirección de la conexión" de la ficha "Información de conexión" esté ocupado, deben cumplirse los siguientes requisitos:

- Existe una conexión online con los puntos finales de la conexión en cuestión.
- Se ha seleccionado una fila en la tabla de conexiones.

Estructura del grupo "Detalles de dirección de la conexión"

Se indican los detalles de las direcciones relevantes para el tipo de conexión para ambos interlocutores.

8.2.3 Diagnóstico del sistema con 'Report System Errors'

8.2.3.1 Introducción al diagnóstico del sistema con "Report System Errors"

Introducción

Los componentes de hardware y los dispositivos de terceros (esclavos cuyas propiedades están determinadas por su archivo GSD) pueden lanzar llamadas de bloques de organización al producirse un error de sistema.

Ejemplo: En caso de rotura de hilo un módulo apto para el diagnóstico puede llamar el OB 82.

Los componentes de hardware ofrecen información sobre el error de sistema ocurrido. La información del evento de arranque, es decir, los datos locales del OB asignado (y que contienen, entre otros, el registro 0), ofrece información general sobre la ubicación (p. ej., dirección lógica del módulo) y el tipo de error (p. ej. fallo de canal o fallo de módulo).

Además, la información de diagnóstico adicional (lectura del registro 1 con la instrucción "RDSYSST" o lectura del telegrama de diagnóstico de esclavos DP normalizados con la instrucción "DPNRM_DG") permite especificar el error con mayor precisión: p. ej. canal 0 ó 1, rotura de hilo o rebase del rango de medición.

El diagnóstico del sistema con "Report System Errors" (RSE) ofrece a los PLC S7-300/400, ET200S, ET200Pro y el software SPS una forma cómoda de evaluar esta información de diagnóstico y visualizarla en forma de avisos. Los bloques y textos de aviso necesarios se crean en las propiedades de cada PLC. Sólo hay que cargar los bloques creados en la CPU y, dado el caso, transferir los textos a los dispositivos HMI conectados.

Por ejemplo, para visualizar los eventos de diagnóstico gráficamente en un dispositivo HMI o vía un servidor web, es posible generar uno o varios DB de estado. Estos DB de estado son actualizados por los bloques de diagnóstico del sistema, por lo que contienen información sobre el estado actual del sistema.

8.2.3.2 Principios básicos del diagnóstico del sistema

El diagnóstico del sistema mediante "Report System Errors" permite generar bloques para analizar los errores del sistema y generar avisos con la descripción textual del error y su ubicación.

Estos avisos se definen por componente capaz de emitir avisos (p. ej. fallo de canal o fallo de rack) y están limitados a 255 avisos por componente capaz de emitir avisos. Una vez rebasado este límite, aparece un aviso indicándolo.

Los ajustes generales que pueden realizarse y modificarse en el diagnóstico del sistema se guardan junto con el proyecto. Estos ajustes influyen en el diagnóstico del sistema tan solo

después de generar los bloques, compilar la configuración hardware o cargar la configuración en los componentes implicados.

Procedimiento recomendado

Lleve a cabo los ajustes para la notificación de errores de sistema y la estructura de los avisos. Determine qué bloques de diagnóstico se crearán. Configure los OB, DB de estado, así como el PLC en STOP. Genere los bloques y cargue la configuración para aplicar los cambios.

Los siguientes capítulos ofrecen información detallada.

Nota

Al utilizar el diagnóstico del sistema puede cambiar el comportamiento del sistema de la instalación en caso de error. Por ejemplo, en algunas circunstancias la CPU no pasa al estado operativo "STOP", como sucede sin el diagnóstico del sistema.

Por lo tanto, debe asegurarse de que todos los mecanismos de protección de la instalación funcionan correctamente.

8.2.3.3 Módulos con posibilidades de diagnóstico limitadas

Introducción

Los componentes de las estaciones S7-300 y S7-400, los dispositivos PROFINET IO, esclavos DP y WinAC son asistidos por la función "Report System Errors" en la medida en que admitan funciones como la alarma de diagnóstico, la alarma de cambio de módulo y el diagnóstico de canal.

Componentes no soportados

"Report System Errors" no asiste los siguientes componentes:

- Configuraciones M7
- Configuraciones PROFIBUS-DP a través de interfaz maestra DP (CP 342-5 DP) en estaciones S7-300
- Dispositivos PROFINET IO a través de un controlador externo (CP 343-1 Advanced) en estaciones S7-300

En caso de rearranque, debe tenerse en cuenta además que pueden faltar avisos de alarma. Este comportamiento se debe a que, en caso de rearranque, la memoria de acuse de avisos de la CPU no se borra, pero "Report System Errors" restablece los datos internos. No se notifican todos los fallos de módulo o canal producidos antes del arranque o durante una avería.

Se notifican un máximo de 8 fallos de canal por módulo.

Nota

En caso de utilizar un CP 443-5 y de que este se encuentre en estado operativo STOP, durante el arranque no se notificará un fallo del sistema maestro.

PROFIBUS DP

Para todos los esclavos PROFIBUS DP se muestran los fallos de estación (fallo/retorno) con un mensaje en texto explícito.

Con todos los esclavos PROFIBUS DP se soporta el diagnóstico específico de fabricante con las siguientes limitaciones:

- Solo se soportan esclavos V1. Se trata de esclavos cuyo archivo GSD contiene la entrada "DPV1_Slave=1".
- El modo de alarma DP para este esclavo debe estar ajustado a "DPV0".

Repetidor de diagnóstico

En el modo DPV0, los avisos del repetidor de diagnóstico se emiten en texto explícito. Los textos se extraen del archivo GSD.

PROFINET IO

A continuación encontrará el diagnóstico de los distintos dispositivos PROFINET IO asistidos por "Report System Errors".

Fallos producidos durante el funcionamiento (CPU en RUN):

- Se soportan los fallos de dispositivo (fallo, retorno)
- Se soportan los fallos de módulo y submódulo (módulo/submódulo desenchufado, módulo/submódulo incorrecto, módulo/submódulo compatible)
- Se soportan los fallos de canal y se evalúa la información ampliada de fallo de canal

AS-Interface

En el caso de los esclavos AS-Interface, se envía un aviso si la configuración real difiere de la teórica.

Shared Devices

"Report System Errors" evalúa la información acerca de si un módulo o submódulo ha sido configurado como Shared Device. Dado que "Report System Errors" siempre está enfocado a una determinada CPU, solo se tienen en cuenta a efectos de diagnóstico los módulos/submódulos que tienen activado el acceso total a dicha CPU. No se toman en consideración los módulos/submódulos configurados como "sin asignar".

8.2.3.4 Bloques de diagnóstico para notificar errores de sistema

Una vez realizados los ajustes para la notificación de errores de sistema, al compilar nuevamente la configuración hardware se crean los bloques necesarios (FB con DB de instancia asignado y uno o varios DB globales, así como una FC y, dependiendo de los ajustes, OB todavía no disponibles) y los DB de estado.

Se crean los siguientes bloques:

- FB de diagnóstico (predeterminado: FB 49),
- DB de instancia para el FB de diagnóstico (predeterminado: DB 49),
- DB global (predeterminado: DB 50),
- FC de diagnóstico (predeterminado: FC 49),
- OB de error (se crean automáticamente si la CPU soporta esta función),
- OB cíclicos (si se ha elegido la opción correspondiente en el grupo "Configuración OB"),
- DB de estado opcionales

Los bloques creados se almacenan en el árbol de proyecto bajo "Bloques de programa > Bloques de sistema > Report System Errors" (excepto los OB).

Nota

Los números de bloque ajustados de los bloques de diagnóstico de sistema generados no pueden solaparse con números de bloques ya existentes. Tampoco pueden coincidir con los números de bloques contenidos en las librerías.

Si surgen problemas al generar los bloques, compruebe si se han modificado los números predeterminados (FB 49, DB 49, FC 49, DB 50). Si este no es el caso, intente repetir la generación con los números de bloque predeterminados o bien utilice números que con seguridad no se utilicen en otro lugar.

Consulte también

Propiedades de los bloques (Página 2221)

8.2.3.5 Propiedades de los bloques

Bloques de diagnóstico

Los bloques de diagnóstico creados (FB con DB de instancia asignado, uno o varios DBs globales y una FC) evalúan los datos locales del OB de error y, si la hubiere, leen la información de diagnóstico adicional de los componentes de hardware causantes del error.

Tienen las propiedades siguientes:

- Lenguaje de creación RSE (Report System Errors) (también sirve para los bloques mencionados anteriormente)
- Know-how-protected (también es aplicable a los bloques mencionados anteriormente)
- Alarmas que aparecen con retardo en runtime

DBs de estado

Los DB de estado sirven de interfaz para los bloques de diagnóstico y permiten visualizar eventos de diagnóstico gráficamente en un dispositivo HMI o un servidor web.

Consulte también

Ajustes para los bloques de diagnóstico del sistema (Página 2229)

8.2.3.6 OBs de error soportados

Los siguientes OB de error se crean automáticamente siempre que también sean soportados por la CPU configurada:

OB	Llamada del FB de diagnóstico	Descripción
OB 70 (error de redundancia en periferia)	Posible	Este OB sólo existe en las CPU H.
OB 72 (error de redundancia en CPU)	Posible	Este OB sólo existe en las CPU H.
OB 73 (error de redundancia en la comunicación)	Posible	Este OB sólo existe en algunas CPU H.
OB 81 (fallo de alimentación)	Posible	
OB 82 (OB de alarma de diagnóstico)	Posible	
OB 83 (alarma de presencia de módulo)	Posible	

OB	Llamada del FB de diagnóstico	Descripción
OB 85 (error de ejecución del programa)	No es posible	Si 'RSE' crea este OB al generar los bloques de diagnóstico, se insertarán segmentos adicionales que realicen la siguiente lógica: • En caso de errores al actualizar la memoria imagen de proceso (p. ej. desenchufar el módulo) se impedirá el estado STOP de la CPU para que se pueda procesar el FB de diagnóstico en el OB 83. Tras notificación de "RSE" tendrá efecto un posible ajuste de "STOP de la CPU" en el OB 83. • En el resto de eventos de error del OB 85, la CPU pasa al estado STOP.
OB 86 (fallo de un rack de ampliación, un sistema maestro DP o una unidad de periferia descentralizada)	Posible	

Si los OB de error ya existen...

Los OB de error existentes no se sobrescriben. Dado el caso, la llamada del FB de diagnóstico se añade al OB existente.

Si la configuración abarca la unidad de periferia descentralizada...

Para la evaluación de errores en la periferia descentralizada, el FB creado llama automáticamente la instrucción "DPNRM_DG" (leer datos de diagnóstico del esclavo DP).

Para asegurar esta función, el FB creado debe llamarse o bien sólo en el OB 1 o bien en un OB de alarma cíclica con un período corto y en los OB de arranque.

Nota

Considere lo siguiente:

- El OB 85 creado por el diagnóstico del sistema no provoca el paso de la CPU al estado STOP cuando se produce el evento de error "Error al actualizar la memoria imagen de proceso".
- La CPU también llama al OB 85 en los siguientes casos de error.
  - "Evento de error para un OB no cargado"
  - "Error al llamar o acceder a un bloque no cargado"
    En estos casos de error, el OB 85 creado por el diagnóstico del sistema sigue originando un estado STOP de la CPU, al igual que antes de utilizar el diagnóstico del sistema.

8.2.3.7 DB de estado de diagnóstico

En el diagnóstico de sistema existe la posibilidad de crear un DB de estado. Este DB de estado es actualizado por los bloques de diagnóstico del sistema, por lo que contiene información sobre el estado actual del sistema.

Interfaz para el DB de estado de diagnóstico

El bloque de datos generado (predeterminado: DB 127) ofrece la posibilidad de consultar el estado de sistema de un componente configurado y, dado el caso, de todos sus componentes subordinados.

Este bloque de datos es necesario para el soporte del diagnóstico del sistema a través del servidor web o también a través del visor de diagnóstico. Está activado de forma predeterminada.

Nota

Tenga en cuenta que el diagnóstico del sistema debe estar activado tanto en el servidor web o visor de diagnóstico como en la ventana de inspección del PLC para poder representar los datos de diagnóstico de los DBs de estado.

Tras efectuar un rearranque completo de una CPU con servidor web, el estado de los módulos se indicará con retardo. Para reducir el tiempo de espera se puede llamar el bloque de diagnóstico RSE en un OB de alarma cíclica con un periodo más corto.

Estructura del DB de estado de diagnóstico

Dirección	Nombre	Tipo de datos	Descripción
+0	Directory		
0	D_Version	WORD	Versión que soporta el diagnóstico del sistema
2	D_pGlobalState	WORD	Offset de byte hasta el principio de la parte "GlobalState"
4	D_pQuery	WORD	Offset de byte hasta el principio de la parte "Query"
6	D_pComponent	WORD	Offset de byte hasta el principio de la parte "Component"
8	D_pError	WORD	Offset de byte hasta el principio de la parte "Error"
10	D_pState	WORD	Offset de byte hasta el principio de la parte "State"
12	D_pAlarm	WORD	Offset de byte hasta el principio de la parte "Alarm"
14	D_pSubComponent	WORD	Offset de byte hasta el principio de la parte "Subcomponent"
+16	GlobalState		
0	G_EventCount	WORD	ID del último evento (contador)
2.0	G_StartReporting	BOOL	Evaluación del arranque en curso
+20	Query		
0	Q_ClientID_User	DWORD	ID del cliente; utilice aquí un valor entre 1 y 255. Asegúrese de que los distintos clientes utilizan ID distintas.
4	Q_ClientID_Intern	DWORD	ID del cliente (interna)
8.0	Q_WithSubComponent	BOOL	Con/sin estado de los componentes subordinados (más lento)
8.1	Q_SubComponentAlarm	BOOL	Maestro AS-i devuelve alarmas del AS-iSlave
8.2	Q_Reserved2	BOOL	Reservado
8.3	Q_Reserved3	BOOL	Reservado
8.4	Q_Reserved4	BOOL	Reservado
8.5	Q_Reserved5	BOOL	Reservado
8.6	Q_Reserved6	BOOL	Reservado
8.7	Q_Reserved7	BOOL	Reservado
9.0	Q_Start	BOOL	Iniciar consulta
10	Q_Error	BYTE	Error interno en la consulta
11	Q_Reserved8	BYTE	Reservado
+32	Component		
0	C_AddressMode	BYTE	Modo de direccionamiento del módulo
1	C_Reserved1	BYTE	Reservado
2	C_ComponentID	WORD	ID de hardware del componente (interna)
+36	Error		
0	E_ErrorNo	WORD	Índice del error solicitado/efectivo

Dirección	Nombre	Tipo de datos	Descripción
2.0	E_LastError	BOOL	Se activa cuando E_ErrorNo es distinto de 0. Valor TRUE si E_ErrorNo es el índice del último error; de lo contrario, FALSE
2.1	E_Reserved	ARRAY [1..15] BOOL	Reservado
+40	State		
0	S_Hierarchy	BYTE	Reservado
1	S_Periphery	BYTE	Reservado
2.0	S_SupFault	BOOL	No se puede acceder al componente.
2.1	S_NotAvailable	BOOL	El componente no está disponible.
2.2	S_Faulty	BOOL	El componente falla; la parte "Alarma" no está vacía
2.3	S_MoreErrors	BOOL	Hay más errores de los que puede guardar el diagnóstico del sistema
2.4	S_Maintenance1	BOOL	Mantenimiento necesario pendiente
2.5	S_Maintenance2	BOOL	Mantenimiento solicitado pendiente
2.6	S_Deactivated	BOOL	Se ha desactivado el componente *)
2.7	S_Reserved2	BOOL	Reservado
3.0	S_SubFault	BOOL	Fallo de un componente subordinado
3.1	S_SubMaintenance1	BOOL	Mantenimiento necesario en un componente subordinado
3.2	S_SubMaintenance2	BOOL	Mantenimiento solicitado para un componente subordinado
3.3	S_SubDeactivated	BOOL	Por lo menos un componente subordinado está desactivado
3.4	S_Reserved4	BOOL	Reservado
3.5	S_Reserved5	BOOL	Reservado
3.6	S_Reserved6	BOOL	Reservado
3.7	S_Reserved7	BOOL	Reservado
4	S_TIAMS	DWORD	Estado de mantenimiento del componente
8	S_TIAMSChannelExist	DWORD	Estado de mantenimiento: canales configurados
12	S_TIAMSChannelOK	DWORD	Estado de mantenimiento: fallan canales
16	S_ChannelCount	WORD	Número de canales; sólo es válido si está activado Q_WithSubComponent
18.0	S_ChannelVector	ARRAY [0..255] BOOL	Lista de canales afectados; sólo es válida si está activado Q_WithSubComponent
+90	Alarma		
0	A_ComponentID	WORD	ID de hardware del componente (interna)
2	A_TextID1	WORD	ID del primer texto de error
4	A_TextLexikonID1	WORD	ID de la primera librería de textos de error
6	A_HelpTextLexikonID1	WORD	ID de la primera librería de textos de ayuda
8	A_MapTextID	WORD	ID del primer texto de error (HMI)
10	A_MapHelpTextID	WORD	ID del primer texto de ayuda (HMI)

Dirección	Nombre	Tipo de datos	Descripción
12	A_TextID2	WORD	ID del segundo texto de error
14	A_TextLexikonID2	WORD	ID de la segunda librería de textos de error
16	A_HelpTextLexikonID2	WORD	ID de la segunda librería de textos de ayuda
18	A_MapTextID2	WORD	ID del segundo texto de error (HMI)
20	A_MapHelpTextID2	WORD	ID del segundo texto de ayuda (HMI)
22	A_AlarmID	DWORD	Número de aviso
26	A_ValueCount	WORD	Números de bytes restantes ocupados (12)
28	A_AssociatedValue	ARRAY [1..n] WORD	Valores asociados del aviso n = A_ValueCount / 2 (= 6)
+130	SubComponent		
0	U_SubComponentCount	WORD	Número de componentes subordinados
2	U_SubComponentFault	ARRAY [1..n] BYTE	Lista de componentes subordinados "n" depende de la configuración **)

*) Si se ha desactivado el componente, el índice del error solicitado/efectivo no cambia y "E_LastError" se pone a "true". Además, el rango de variables de la alarma no se llena.

**) La lista de componentes subordinados sólo es válida si está activado Q_WithSubComponent. Por cada componente configurado, el ARRAY contiene un byte de estado. En el caso de un maestro, el ARRAY contiene el estado de las estaciones configuradas por orden ascendente de la ID de estación. En el caso de una estación, el ARRAY contiene el estado de los slots configurados por orden ascendente de la ID del slot. Este campo puede contener un máximo de 4096 entradas (para un sistema IO); sólo se indica el tamaño máximo real.
El byte de estado por componente subordinado se define tal como se indica a continuación:
Bit 0 = SubFault: el componente no es accesible
Bit 1 = Fault: el componente no está disponible o falla
Bit 2 = Maintenance1: el componente ha notificado mantenimiento
Bit 3 = Maintenance2: el componente ha notificado mantenimiento
Bit 4 = Deactivated: el componente ha sido desactivado
Bit 5 = SubFault: falla un componente subordinado
Bit 6 = SubMaintenance1: un componente subordinado ha notificado mantenimiento
Bit 7 = SubMaintenance2: un componente subordinado ha notificado mantenimiento

8.2.3.8 Visualización de los ajustes del diagnóstico del sistema

Requisitos

El área "Propiedades" de la ventana de inspección del PLC deseado está abierta.

Procedimiento

Para mostrar y editar las propiedades del diagnóstico del sistema, proceda del siguiente modo:

1. Haga clic en "Diagnóstico del sistema" en la navegación local de la ventana de inspección. En la parte derecha de la ventana de inspección es posible visualizar y editar los distintos ajustes.

8.2.3.9 Configuración

Configuración general

Activar el diagnóstico de sistema para este PLC

Esta opción está desactivada por defecto.

Nota

Si la opción está desactivada, en el servidor web y en el visor de diagnóstico no se leerán datos de diagnóstico aunque la opción esté activada allí.

Si esta opción se desactiva en un PLC en el que estaba activada previamente, todos los datos de diagnóstico se borrarán la próxima vez que se compile la configuración hardware.

Ajustes de avisos

Para las categorías de aviso listadas a continuación se puede definir si emitir un aviso y si el aviso debe ser de acuse obligatorio o no.

- Error
- Mantenimiento solicitado
- Mantenimiento necesario
- Información

Active para ello los botones de opción deseados en las columnas "Aviso" y "Acuse".

En la columna "Categoría" se indica la categoría de aviso ajustada.

Configuración del soporte de diagnóstico

Con ayuda del soporte de diagnóstico el FB de diagnóstico genera datos de diagnóstico en un DB de estado especial. En tal caso, este puede representarse gráficamente, p. ej. en un dispositivo HMI.

DB de estado

Active el DB de estado para leer a través de este bloque de datos el estado de sistema actual de los componentes configurados.

Está predeterminado el siguiente DB de estado:

Bloque	Nombre	Número
DB de estado de diagnóstico	RSE_DIAGNOSTIC_STATUS_DB	127

No obstante, es posible asignar a este bloque otro nombre y/o número, siempre y cuando aún no estén en uso.

Si la funcionalidad "Activar webserver en este módulo" está activada en las propiedades de la CPU, el DB de estado de diagnóstico debe estar forzosamente activado. De forma predeterminada se genera el DB 127 si la CPU soporta el servidor web y éste está activado en las propiedades de la CPU.

Ajustes de diagnóstico avanzados

Para la consulta es imprescindible que el PLC incluya la instrucción "D_ACT_DP".

Active la opción "Consulta si el estado es "activado/desactivado" tras el arranque del PLC", si desea que se consulte el estado de los esclavos.

Active la opción "Enviar aviso si el estado cambia de/a activado o desactivado", si desea que aparezca un aviso cuando se produzca un cambio de estado. Para utilizar esta opción, debe utilizar la instrucción "D_ACT_DP" en modo 3/4 y haber seleccionado la llamada del FB de diagnóstico del OB 86 en el grupo "Configuración OB".

Consulte también

D_ACT_DP: Activar / desactivar esclavos DP (Página 6764)

Ajustes para los bloques de diagnóstico del sistema

Los bloques de diagnóstico del sistema tienen predefinidos los siguientes valores:

Bloque	Nombre	Número
FB de diagnóstico	RSE_FB	49
DB de diagnóstico	RSE_DB	49
primer DB global	RSE_GLOBAL_DB	50
FC de diagnóstico	RSE_FC	49

No obstante, es posible asignar a estos bloques otros números o nombres, siempre y cuando aún no estén en uso.

8.2.4 Diagnóstico del sistema para PLCs S7-1500

8.2.4.1 Introducción

Introducción

Los componentes de hardware y los dispositivos de terceros (esclavos cuyas propiedades están determinadas por su archivo GSD) pueden lanzar una reacción del sistema al producirse un error de sistema. Los componentes de hardware ofrecen información sobre el error de sistema ocurrido.

El diagnóstico del sistema ofrece para los PLC S7-1500 una manera cómoda de evaluar esta información de diagnóstico y visualizarla en forma de avisos.

Además, la información de diagnóstico adicional permite especificar el error con mayor precisión, p. ej., canal "Interruptor_Arriba", rotura de hilo o rebase del rango de medición.

8.2.4.2 Principios básicos del diagnóstico del sistema

El diagnóstico del sistema permite analizar los errores del sistema y generar avisos con la descripción textual del error y su ubicación.

Estos avisos se definen para cada componente que emite avisos (p. ej., fallo de canal o fallo de rack).

Procedimiento recomendado

Lleve a cabo los ajustes para el diagnóstico del sistema y la estructura de los avisos.

Los siguientes capítulos ofrecen información detallada.

8.2.4.3 Visualización de los ajustes del diagnóstico del sistema

Requisitos

El área "Propiedades" de la ventana de inspección del PLC deseado está abierta.

Procedimiento

Para mostrar y editar las propiedades del diagnóstico del sistema, proceda del siguiente modo:

1. Haga clic en "Diagnóstico del sistema" en la navegación local de la ventana de inspección. En la parte derecha de la ventana de inspección es posible visualizar y editar los distintos ajustes.

8.2.4.4 Configuración

Configuración general

Activar el diagnóstico de sistema para este PLC

Esta opción está activada por defecto y no puede desactivarse.

Ajustes de avisos

Para las categorías de aviso listadas a continuación se puede definir si emitir un aviso y si el aviso debe ser de acuse obligatorio o no.

- Error
- Mantenimiento solicitado
- Mantenimiento necesario
- Información

Active para ello los botones de opción deseados en las columnas "Aviso" y "Acuse".

En la columna "Categoría" se indica la categoría de aviso ajustada.

8.2.5 Utilizar opciones de diagnóstico adicionales

8.2.5.1 Diagnosticar el hardware

Mostrar valores actuales y no modificables de propiedades de módulos parametrizables

Mostrar propiedades generales e información relevante para la instalación relacionadas con un módulo

¿Dónde está la información buscada?

Las propiedades generales y la información relevante para la instalación relacionadas con un módulo están en el grupo "General" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse.

Estructura del grupo "General"

El grupo "General" incluye las áreas siguientes:

- Módulo
- Información del módulo
- Información del fabricante

Área "Módulo"

En S7-300 y S7-400, esta área muestra además el siguiente dato del módulo:

- Versión del bootloader

Área "Información del módulo"

En S7-300 y S7-400, esta área muestra además los siguientes datos del módulo, que se han ajustado durante la configuración del hardware:

- Nombre de la estación (que se asignó en la vista de redes.)
- Identificador de instalación
  Si una CPU, un esclavo, un dispositivo IO o un módulo de la estación soporta la parametrización de un identificador de instalación unívoco en toda la instalación, se muestra aquí (el identificador de instalación permite identificar de modo unívoco partes de la instalación con criterios funcionales. Tiene una estructura jerárquica conforme a IEC 1346-1.). Permite, p. ej., identificar una CPU desde el programa de usuario. El identificador de instalación puede leerse con la instrucción "RDSYSST".
- Identificador de ubicación
  Si una CPU, un esclavo, un dispositivo IO o un módulo de la estación soporta la parametrización de un identificador de ubicación unívoco en toda la instalación, se muestra aquí (el identificador de ubicación forma parte del identificador de recursos, que caracteriza p. ej. la posición exacta de un punto de instrumentación y control dentro de una instalación técnica de proceso, de modo análogo a como se ubica una calle en el plano de una ciudad).

Área "Información del fabricante"

En S7-300 y S7-400, esta área muestra además el siguiente dato del módulo:

- Entrada de derechos de autor

Ver las propiedades de comunicación de un módulo

¿Dónde está la información buscada?

Las propiedades de comunicación de un módulo están en el grupo "Comunicación" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse.

Se trata de las siguientes propiedades:

- Carga del ciclo por comunicación, configurada:
  Valor ajustado del parámetro "Carga de ciclo por comunicación", es decir, aquella parte del rendimiento de la CPU de la que se dispone en el ciclo para la comunicación.
- Recursos de conexión, número máximo:
  Indica el número máximo posible de recursos de conexión disponibles en el módulo. Siempre hay un determinado número de los recursos de conexión reservado para la comunicación PG y OP, y en S7-300 (a excepción de la CPU 318-2 DP), para la comunicación básica S7. Sin embargo, además de esto pueden crearse también otras conexiones para PG u OP; en tal caso, el número de los recursos de conexión no ocupados se reducirá proporcionalmente.

Ver las interfaces y las propiedades de interfaces de un módulo

¿Dónde está la información buscada?

Las interfaces y las propiedades de interfaces de un módulo están en los siguientes grupos de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse:

- Interfaz MPI
- Interfaz MPI/DP
- Interfaz DP
- Interfaz PROFINET

Grupo "Interfaz MPI"

Este grupo muestra el siguiente dato:

- Velocidad de transferencia:
  La velocidad de transferencia ajustada actualmente en la interfaz MPI del sistema de automatización

Grupo "Interfaz MPI/DP"

Este grupo muestra los siguientes datos:

- Velocidad de transferencia:
  La velocidad de transferencia ajustada actualmente en la interfaz MPI/DP del sistema de automatización
- Modo de operación DP:
  Modo de compatibilidad de la interfaz MPI/DP ajustada al tipo de interfaz "PROFIBUS". En los módulos que soportan el modo de operación "DPV1", este puede adoptar los valores "Compatible con S7" y "DPV1".

Nota

Si la interfaz MPI/DP está ajustada al tipo de interfaz "MPI", el campo "Modo de operación DP" no contiene ningún valor válido.

Grupo "Interfaz DP"

Este grupo muestra los siguientes datos:

- Modo de operación DP:
  Modo de compatibilidad de la interfaz DP. En los módulos que soportan el modo de operación "DPV1", este puede adoptar los valores "Compatible con S7" y "DPV1".

Grupo "Interfaz PROFINET"

Este grupo se divide en las siguientes áreas:

- "Dirección Ethernet" con las subáreas "Conexión de red" y "Parámetro IP"
- "Puertos"
- "Estadística" (esta área no se contempla aquí, ya que no se trata de valores reales de propiedades de módulo configurables.)

Nota

Las áreas "Dirección Ethernet" y "Puertos" ya se han descrito fuera del capítulo "Utilizar posibilidades de diagnóstico adicionales", ya que son aplicables a S7-1200 y S7-300/S7-400.

Ver las áreas de operandos, bloques de organización y bloques de sistema de una CPU

¿Dónde se encuentra la información buscada?

Las áreas de operandos, los bloques de organización y los bloques de sistema de una CPU están en el grupo "Datos de rendimiento" de la carpeta "Diagnóstico" de la vista Online y diagnóstico del módulo que debe diagnosticarse.

Grupo "Datos de rendimiento"

En el grupo "Datos de rendimiento" se muestra la siguiente información acerca de la CPU correspondiente:

- Áreas de operandos:
  Esta tabla informa acerca de
  - El tipo de operando, p. ej., memoria imagen de proceso de las entradas, contadores
  - El número de operandos o su tamaño de área
  - El área de direcciones disponible de los operandos, p. ej. de E0.0 a E127.7
- Bloques de organización:
  Todos los bloques de organización (OBs) junto con su función
  Las líneas de la tabla pueden ordenarse de la siguiente manera: Haciendo clic en un rótulo de columna, las líneas de la tabla se ordenan de manera que los contenidos de dicha columna aparecen en orden alfabético descendente.
- Bloques de sistema:
  Todas las funciones de sistema (SFCs) y todos los bloques de función de sistema (SFBs)
  Las líneas de la tabla pueden ordenarse de la siguiente manera: Haciendo clic en un rótulo de columna, las líneas de la tabla se ordenan de manera que los contenidos de dicha columna aparecen en orden alfabético descendente.

Ver las propiedades del sistema horario de un módulo

¿Dónde están las funciones buscadas?

Las propiedades del sistema horario de un módulo se encuentran en su vista online y de diagnóstico, en el grupo "Ajustar hora" de la carpeta "Funciones".

Estructura del grupo "Ajustar hora"

El grupo "Ajustar hora" incluye las áreas siguientes:

- Área para la lectura y ajuste de la hora (esta área no se contempla aquí)
- Sistema horario

Estructura del área "Sistema horario"

El área "Sistema horario" está compuesta por las siguientes subáreas:

- "Reloj"
- "Sincronización"

Subárea "Reloj"

La subárea "Reloj" incluye los elementos siguientes:

- Resolución:
  Indica la precisión con la que se realiza el cálculo y la presentación de la hora
- Reloj en tiempo real:
  Indica si la CPU está provista de reloj en tiempo real: En caso afirmativo, "existente", y en caso negativo, "no existente"
- Factor de corrección:
  Factor de corrección parametrizado para el reloj en tiempo real del módulo. El factor de corrección compensa la desviación diaria del reloj.

Subárea "Sincronización"

Los mecanismos de sincronización permiten sincronizar los relojes de diferentes sistemas parciales. Los ajustes para la sincronización horaria se realizan al parametrizar el módulo.

El reloj en tiempo real integrado -si existe- puede sincronizar los relojes de otros módulos (modo maestro) o puede ser sincronizado por los relojes de otros dispositivos (modo esclavo), o bien puede no participar en la sincronización. La sincronización puede realizarse en el AS o en MPI.

Mostrar valores actuales de propiedades de módulos dinámicas

Mostrar el estado actual de los LEDs y el selector de modo de una CPU

¿Dónde está la información buscada?

El estado actual de los LEDs y el selector de modo de una CPU se encuentra en el área de visualización de la paleta "Panel de control de la CPU" de la Task Card "Herramientas online".

Área de visualización de la paleta "Panel de control de la CPU" de la Task Card "Herramientas online"

Esta área contiene los indicadores siguientes:

- Nombre de estación y tipo de CPU
- Error (equivale al LED "SF" en las CPUs S7-300)
- RUN (equivale al estado operativo "RUN" de la CPU)
- STOP (equivale al estado operativo "STOP" de la CPU)
- FORCE (equivale al estado operativo "FRCE" de la CPU)
- Posición actual del selector de modo

Nota

Puede suceder que, en las CPUs S7-300 con selector de modo, se muestre erróneamente la posición "RUN-P". En este caso, el selector de modo se encuentra en la posición "RUN".

Ver los recursos de conexión todavía no ocupados y las conexiones ocupadas y reservadas de una CPU

¿Dónde está la información buscada?

Los recursos de conexión todavía no ocupados y las conexiones ocupadas y reservadas de una CPU se encuentran en el grupo "Comunicación" de la carpeta "Diagnóstico" en la vista online y de diagnóstico.

El área "Recursos de conexión" contiene los valores actuales de las siguientes propiedades dinámicas de módulos:

- Número de recursos de conexión todavía no ocupados
  Se encuentran en el apartado "No ocupados".

  Nota

  Si ya hay recursos de conexión reservados para determinados tipos de comunicación, los no ocupados no siempre pueden utilizarse para cualquier tipo de conexión.

- Comunicación PG "reservada" y "ocupada":
  Número de recursos de conexión que están reservados u ocupados para conexiones entre el módulo y las PGs.

  Nota

  El número de recursos de conexión ocupados puede ser mayor que el número de recursos de conexión reservados, ya que, siempre dentro del número máximo, pueden utilizarse recursos de conexión todavía libres para la comunicación PG. En las CPUs S7-300 con recursos de conexión parametrizables, es posible aumentar el número de recursos de conexión para la comunicación PG a costa de la comunicación básica S7.

- Comunicación OP "reservada" y "ocupada":
  Número de recursos de conexión que están reservados u ocupados para conexiones entre el módulo y los dispositivos HMI (TDs, OPs).

  Nota

  El número de recursos de conexión ocupados puede ser mayor que el número de recursos de conexión reservados, ya que, siempre dentro del número máximo, pueden utilizarse recursos de conexión todavía libres para la comunicación OP. En las CPUs S7-300 con recursos de conexión parametrizables, es posible aumentar el número de recursos de conexión para la comunicación OP a costa de la comunicación básica S7.

- Comunicación S7 "reservada" y "ocupada":
  Se visualiza en las CPUs S7-400, así como en la CPU 318-2 y en la CPU 317-2 PN/DP. Recursos reservados u ocupados para conexiones S7 configuradas que permiten intercambiar datos en el programa de usuario mediante la llamada de bloques de función de comunicación (SFBs). Estas conexiones pueden contemplarse como si estuvieran cableadas de modo permanente (p. ej. conexiones CPU-CPU). Pueden establecerse automáticamente al inicio de la comunicación o bien establecerse o deshacerse mediante instrucciones de programación. La comunicación S7 es posible a través de MPI, PROFIBUS e Industrial Ethernet.
- Comunicación básica S7 "reservada" y "ocupada":
  Se visualiza en las CPUs S7-300 (no en la CPU 318-2 DP ni en la CPU 317-2 PN/DP). Recursos reservados u ocupados para conexiones no configuradas que permiten intercambiar datos en el programa de usuario mediante la llamada de funciones de comunicación (SFCs). La comunicación básica S7 solo es posible a través de MPI o dentro del sistema de automatización.
- Otra comunicación "ocupada"
  Otros recursos de conexión ocupados que no tienen reservados recursos de conexión. Ejemplos:
  - Comunicación de una CPU S7-300 a través de un CP (solo en las CPUs con recursos de conexión parametrizables)
  - Gateway (routing S7) en CPUs S7-400

Nota

Los recursos de conexión ocupados por una función de servidor en la comunicación S7 (es decir, cuando una CPU S7-400 se comunica a través de las SFCs "PUT" o "GET" con una CPU S7-300) se muestran en las CPUs S7-300 en los siguientes recursos:

- Si la CPU no posee recursos de conexión parametrizables: En "Comunicación básica S7"
- Si la CPU posee recursos de conexión parametrizables: En las CPUs 317-2 PN/DP y 318-2, en "Comunicación S7", y en todas las demás en "Otra comunicación"

Ver los datos estadísticos de las transferencias de datos realizadas a través de la interfaz Ethernet

¿Dónde está la información buscada?

Los datos estadísticos de las transferencias de datos realizadas a través de la interfaz Ethernet se encuentran en el área "Estadística" del grupo "Interfaces PROFINET" en la carpeta "Diagnóstico" de la vista online y de diagnóstico.

Área "Estadística"

- Visualización
  Se muestran las estadísticas de paquetes de datos enviados y recibidos. Con ayuda de estos datos se puede evaluar la calidad de la transferencia de datos.
  El inicio del registro de las estadísticas se indica en la primera línea ("Paquetes de datos desde:"). Cada reinicio de los datos estadísticos reinicia ese tiempo (botón "Resetear").
  La visualización de datos estadísticos se actualiza de modo continuo.
- Botón "Resetear"
  El tiempo "Paquetes de datos desde" se reinicia y los datos estadísticos se ponen a "0".

Ver información acerca de los contadores de horas de funcionamiento de una CPU

¿Dónde está la información buscada?

La información acerca de los contadores de horas de funcionamiento está en el grupo "Contador de horas de funcionamiento" de la carpeta "Diagnóstico" de la vista online y de diagnóstico.

Grupo "Contador de horas de funcionamiento"

En el grupo "Contador de horas de funcionamiento" se muestra la siguiente información acerca de los contadores de horas de funcionamiento activos:

- N.º del contador de horas de funcionamiento junto con la información de si se trata de un contador de 16 bits o de 32 bits
- Horas de funcionamiento transcurridas
- Estado
- Información sobre si se ha producido un desbordamiento

Si la CPU tiene uno o varios contadores de horas de funcionamiento de 32 bits, al pasar de 32767 (valor máximo de un contador de horas de funcionamiento de 16 bits) el icono correspondiente aparece rodeado de un marco rojo. En la columna "Desbordamiento" aparece "no" si el contador de horas de funcionamiento no ha rebasado todavía el valor máximo de un contador de 32 bits.

El contador de horas de funcionamiento de 32 bits se activa, inicia, detiene y lee con la instrucción "RTM (Página 6694)".

Comprobar los fallos de un módulo

Realizar el diagnóstico específico de canal de módulos no CPU

¿Dónde se muestran los resultados del diagnóstico específico de canal de los módulos no CPU?

Los resultados del diagnóstico específico de canal de los módulos no CPU se muestran en el grupo "Diagnóstico de canal" en su vista online y de diagnóstico.

Grupo "Diagnóstico de canal"

La visualización está compuesta por los siguientes elementos para cada resultado:

- Slot del módulo averiado
- Número del canal averiado
  Tenga en cuenta que en esta visualización el recuento de los números de canal empieza por 0.
- Tipo de error

Nota

Alarmas de diagnóstico

Sólo es posible notificar una alarma de diagnóstico a la CPU si el módulo es apto para esta función y se ha habilitado la alarma de diagnóstico.

La visualización de la alarma de diagnóstico es una captura instantánea. Los fallos esporádicos de un módulo se detectan en el búfer de diagnóstico de la CPU correspondiente.

Al seleccionar una línea en la tabla de los errores de canal, se ofrece información adicional acerca de ese error (si existe).

Efectuar un diagnóstico de comunicación de los puertos PROFINET

¿Cómo se muestran los fallos específicos de puerto de los puertos PROFINET?

Los errores específicos de puertos de los puertos PROFINET se muestran en el apartado "Interfaz PROFINET" del grupo "Diagnóstico de comunicación" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse.

Estructura del grupo "Diagnóstico de comunicación"

El grupo "Diagnóstico de comunicación" incluye las áreas siguientes:

- Tabla con los errores específicos de puertos
- Campo de texto "Detalles"
- Campo de texto "Ayuda para la línea de diagnóstico seleccionada"

Estructura de la tabla de errores específicos de puertos

Cada línea de la tabla de errores corresponde a un error.

La tabla contiene las columnas siguientes:

- Nombre: N.º de puerto y denominación de puerto, así como el icono de diagnóstico
- Error: Descripción del error sucedido en ese puerto

Campo de texto "Detalles"

Al seleccionar una línea en la tabla de errores, el campo de texto "Detalles" contiene información acerca del error correspondiente (si existe).

Campo de texto "Ayuda para la línea de diagnóstico seleccionada"

Al seleccionar una línea en la tabla de errores, el campo de texto "Ayuda para la línea de diagnóstico seleccionada" contiene información de ayuda acerca del error correspondiente (si existe).

Diagnosticar componentes de PROFINET IO

¿Cómo se diagnostican los componentes de PROFINET IO?

El diagnóstico de los componentes PROFINET IO se realiza en el grupo "Diagnóstico de PROFINET IO" del apartado "Interfaz PROFINET" de la carpeta "Diagnóstico" de la vista online y de diagnóstico.

Grupo "Diagnóstico de PROFINET IO"

En esta área se muestran textos de diagnóstico, específicos de fabricante, para el dispositivo IO o para el módulo seleccionado del dispositivo IO.

Si puede leerse el diagnóstico desde el controlador IO, se muestran errores de comunicación y errores de configuración (diferencia online/offline).

Iniciar el diagnóstico de NCM S7

Requisitos

- No es necesario que haya conexión online con el módulo.

Procedimiento

Para iniciar el diagnóstico NCM S7, proceda del siguiente modo:

1. Abra el módulo en la vista online y de diagnóstico.
2. Elija el grupo "Diagnóstico especial" de la carpeta "Funciones".
3. Haga clic en el botón "Iniciar diagnóstico especial" del área "Diagnóstico especial".

Resultado

Se inicia el diagnóstico de NCM S7.

Modificar las propiedades de un módulo o de la PG/el PC

Particularidad al cambiar el estado operativo de una CPU S7-400

Preselección del tipo de arranque de una CPU S7-400 al cambiar al estado operativo "RUN"

Para pasar una CPU S7-400 al estado operativo "RUN", elija uno de los tipos de arranque siguientes:

- Rearranque (en caliente)
- Arranque en frío
- Rearranque

Registrarse en la CPU para acceder a avisos y resultados de diagnóstico del sistema

Requisitos

- A través de la interfaz PG/PC puede accederse a uno o varios dispositivos de una subred.

Procedimiento

1. Abra la vista online y de diagnóstico de la CPU en cuestión.
2. Seleccione el grupo "Accesos online".
3. En el área "Avisos", active la casilla de verificación "Recibir avisos".

Resultado

La próxima vez que se conecte online, la PG/el PC se registrará en la CPU en cuestión para acceder a avisos y resultados de diagnóstico del sistema.

Nota

Para visualizar los avisos, se requiere una conexión online permanente con cada una de las CPUs de las que se reciban avisos.

Registrarse con una conexión online ya existente

Si la conexión online con una CPU ya está establecida, es posible registrarse o darse de baja para avisos posteriormente de la siguiente manera:

- Con el comando "Recibir avisos" del menú "Online"
- En el árbol del proyecto, mediante el menú contextual de la CPU

- En la vista de dispositivos de la configuración de hardware, mediante el menú contextual de la CPU
- En la vista de redes de la configuración de hardware, mediante el menú contextual de la CPU

Consulte también

Recibir avisos (Página 2154)

Comprimir la memoria y copiarla de RAM a ROM

¿Dónde está la información buscada?

La reorganización de la memoria y su copia de RAM a ROM se encuentra en el área de comandos de la paleta "Memoria" de la Task Card "Herramientas online".

La reorganización de la memoria se encuentra también en el área de comandos del grupo "Memoria" de la carpeta "Diagnóstico" de la vista online y de diagnóstico del módulo que debe diagnosticarse.

Área de comandos de la paleta "Memoria" de la Task Card "Herramientas online"

Esta área puede contener los botones siguientes:

- Comprimir
  Con este botón se reorganiza la memoria para aumentar el tamaño de la mayor área de memoria coherente libre.
- Copiar de RAM a ROM
  Con este botón se copia el contenido de la memoria de carga RAM (programa de usuario) a una FEPROM:
  - En la Memory Card insertada en la CPU
  - En la FEPROM integrada

Nota

Los botones se muestran siempre únicamente en el caso de que el módulo soporte la función en cuestión.

Área de comandos del grupo "Memoria" de la carpeta "Diagnóstico" en la vista online y de diagnóstico

Esta área contiene los botones siguientes:

- Comprimir
  Con este botón se reorganiza la memoria para aumentar el tamaño de la mayor área de memoria coherente libre.

Inicialización de los parámetros de interfaz PROFINET de una CPU S7-300 o S7-400

Requisitos

- Se utiliza una CPU S7-300 de cualquier versión de firmware o una CPU S7-400 con versión de firmware >= V6.0.
- No hay ninguna Memory Card insertada en la CPU.
- Existe una conexión online con la CPU cuyos parámetros de interfaz PROFINET deben inicializarse.
- Dicha CPU está en estado operativo STOP.

Nota

Si la CPU todavía está en estado operativo RUN, se puede pasar a STOP tras responder afirmativamente a una consulta de seguridad al iniciar el proceso de restablecimiento de la configuración de fábrica.

Procedimiento

Para inicializar los parámetros de interfaz PROFINET de una CPU S7-300 o S7-400, proceda del siguiente modo:

1. Abra la vista Online y diagnóstico de la CPU.
2. Seleccione en la carpeta "Funciones" el grupo "Restablecimiento de los parámetros de interfaz PROFINET".

Nota

El aspecto de la interfaz de usuario del grupo "Restablecimiento de los parámetros de interfaz PROFINET" de la vista Online y diagnóstico depende de cómo se ha abierto dicha vista:

- Si la vista se abre desde el árbol de proyecto a través de "Actualizar dispositivos accesibles", la dirección MAC del módulo correspondiente se conoce y en consecuencia aparece atenuada.
- Si se abre la vista desde el contexto de proyecto, debe introducirse la correspondiente dirección MAC. De lo contrario, el botón "Restablecer" aparece en color gris.

3. Haga clic en el botón "Resetear".
4. Responda la consulta de seguridad haciendo clic en "Aceptar".

Resultado

El módulo pasará al estado operativo STOP en caso necesario y se restablecerán los parámetros de la interfaz PROFINET. Esto significa que:

- Se restablecerán las configuraciones de la interfaz Ethernet.
- La dirección IP se borrará.

Iniciar el cargador de firmware para un CP S7-300 o S7-400.

Requisitos

- El CP que se debe actualizar soporta actualización de firmware.
- No es necesario que haya conexión online con el CP.

Procedimiento

Para iniciar el cargador de firmware para un CP S7-300 o S7-400, proceda del siguiente modo:

1. Abra el CP en la vista Online y diagnóstico.
2. Elija el grupo "Actualización de firmware" de la carpeta "Funciones".
3. Haga clic en el botón "Iniciar cargador de firmware" del área "Actualización de firmware".

Resultado

Se inicia el cargador de firmware

Nota

La actualización del firmware de los CP S7-300 y S7-400 con el cargador de firmware difiere de los mecanismos estándar de actualización de firmware.

Asignación de una dirección IP a un dispositivo PROFINET IO

Principios básicos para asignar una dirección IP a un dispositivo PROFINET IO

Resumen

Para seleccionar una dirección IP hay básicamente dos posibilidades:

- La dirección IP es asignada por el usuario.
- La dirección IP se obtiene de un servidor DHCP (no es posible con todos los dispositivos IO). En este caso, dependiendo de la opción seleccionada, al servidor DHCP se le transfiere el ID de cliente, la dirección MAC o el nombre del dispositivo IO.

Requisitos

Para los pasos siguientes se presupone que la dirección IP del dispositivo IO se puede obtener también de un servidor DHCP. Así ocurre, p. ej., en algunos CPs S7-300.

Abrir la asignación de dirección mediante "Dispositivos accesibles"

Requisitos

- Se ha abierto la vista Online y diagnóstico del dispositivo PROFINET IO desde "Actualizar dispositivos accesibles" (en el árbol del proyecto) o desde "Dispositivos accesibles..." (menú "Online").

Procedimiento para la asignación de la dirección IP por parte del usuario

Para asignar al dispositivo IO una dirección IP preseleccionada, proceda del siguiente modo:

1. Abra la vista Online y diagnóstico del dispositivo IO.
2. Elija el grupo "Asignar dirección IP" de la carpeta "Funciones".
3. Seleccione la opción "Utilizar parámetros IP".
4. Introduzca la dirección IP que desee.
5. Introduzca la máscara de subred.
6. Si hay que utilizar un router, active la casilla de verificación "Utilizar router" e introduzca su dirección IP.
7. Haga clic en el botón "Asignar dirección IP".

Procedimiento para obtener la dirección IP desde un servidor DHCP

Para asignar al dispositivo IO una dirección IP preseleccionada de un servidor DHCP, proceda del siguiente modo:

1. Abra la vista Online y diagnóstico del dispositivo IO.
2. Elija el grupo "Asignar dirección IP" de la carpeta "Funciones".
3. Seleccione la opción "Obtener la dirección IP de un servidor DHCP".
4. Si el servidor DHCP debe identificar el dispositivo de destino por su ID de cliente, seleccione la opción "ID de cliente" Introduzca el ID de cliente en el cuadro de entrada homónimo.

   Nota

   El ID de cliente es una secuencia de 63 caracteres como máximo. Solo se permiten los caracteres siguientes: a-z, A-Z, 0 a 9 y - (guión)

5. Alternativa: Si el servidor DHCP debe identificar el dispositivo de destino por su dirección MAC, seleccione la opción "Dirección MAC".

6. Alternativa: Si el servidor DHCP debe identificar el dispositivo de destino por su nombre, seleccione la opción "Nombre de dispositivo".

   Nota

   Si se ha especificado que el servidor DHCP debe identificar el dispositivo de destino por su nombre, previamente ha debido asignarse al mismo un nombre.

7. Haga clic en el botón "Asignar dirección IP".

Resultado

Al dispositivo IO se le asigna la dirección IP con carácter permanente. Dicha dirección se conserva incluso después de un arranque o un fallo de tensión.

Consulte también

Remanencia de parámetros de dirección IP y nombres de dispositivo (Página 1501)

Abrir la asignación de dirección desde el contexto de proyecto

Requisitos

- Existe una conexión online con el dispositivo PROFINET IO.
- Se ha abierto la vista Online y diagnóstico del dispositivo PROFINET IO desde el contexto de proyecto.

Procedimiento

1. Abra el grupo "Asignar dirección IP" de la carpeta "Funciones". La opción "Utilizar parámetros IP" está activada.
2. Haga clic en el botón "Dispositivos accesibles" para determinar los dispositivos accesibles a través de direcciones MAC. Una vez efectuada la búsqueda, seleccione el dispositivo IO con la dirección MAC que conozca.
3. Haga clic en el botón "Asignar dirección IP".

Resultado

Al dispositivo IO se le asigna la dirección IP con carácter permanente. Dicha dirección se conserva incluso después de un arranque o un fallo de tensión.

Consulte también

Remanencia de parámetros de dirección IP y nombres de dispositivo (Página 1501)

8.2.5.2 Diagnosticar conexiones

Diagnosticar conexiones detalladamente

Determinar los detalles de conexión de una conexión NCM

¿Dónde se determinan los detalles de conexión de una conexión NCM?

Los detalles de conexión de una conexión NCM están indicados en el grupo "Detalles de conexión". Este grupo se encuentra en el área "Diagnóstico > Información de conexión" de la ventana de inspección.

¿Cuándo está ocupado el grupo "Detalles de conexión"?

Para que el grupo "Detalles de conexión" de la ficha "Información de conexión" esté ocupado, deben cumplirse los siguientes requisitos:

- Existe una conexión online con el módulo en cuestión.
- Se ha seleccionado una línea en la tabla de conexión.

Estructura del grupo "Detalles de conexión" de una conexión NCM

En una conexión NCM, el grupo "Detalles de conexión" incluye los elementos siguientes:

- ID de conexión local (hex)
- Nombre de conexión
- Estado de la conexión
- Estado de envío
- Estado de transmisión
- Número de bytes enviados
- Número de bytes recibidos
- Botón "Diagnóstico especial"
  El hacer clic en este botón se inicia el diagnóstico de conexión NCM.

Consulte también

Iniciar el diagnóstico de NCM S7 (Página 2241)

Programar el PLC 9

9.1 Cree el programa de usuario

9.1.1 Principios básicos de la programación

9.1.1.1 Sistema operativo y programa de usuario

Sistema operativo

Función

El sistema operativo está contenido en toda CPU y sirve para organizar todas las funciones y procesos de la CPU no relacionados con una tarea de control específica.

Algunas de las tareas del sistema operativo son p. ej.:

- Ejecutar un rearranque (en caliente)
- Actualización de las memorias imagen de proceso de las entradas y salidas
- Llamada del programa de usuario
- Registro de alarmas y llamada de los OBs de alarma
- Detección y tratamiento de errores
- Administración de áreas de memoria

El sistema operativo forma parte de la CPU y ya está contenido en ella en el momento de suministro.

Consulte también

Programa de usuario (Página 2249)

Programa de usuario

Función

El programa de usuario contiene todas las funciones necesarias para ejecutar la tarea de automatización específica.

Algunas de las tareas del programa de usuario son:

- Comprobación de los requisitos previos para un rearranque completo (en caliente) con ayuda de OBs de arranque, p. ej. final de carrera en posición correcta o dispositivo de seguridad activo
- Tratamiento de datos de proceso, p. ej. combinación lógica de señales binarias, lectura y evaluación de valores analógicos, definición de señales binarias para la salida, emisión de valores analógicos
- Reacción a alarmas, p. ej. alarma de diagnóstico al excederse por defecto el valor límite de un módulo de ampliación analógico
- Procesamiento de fallos durante la ejecución normal del programa

El usuario crea el programa de usuario y lo carga en la CPU.

Consulte también

Sistema operativo (Página 2249)

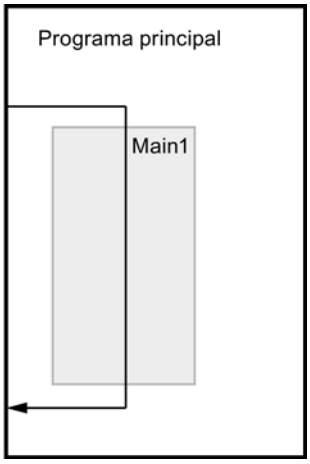
9.1.1.2 Bloques del programa de usuario

Programación lineal y estructurada

Programación lineal

Las tareas de automatización pequeñas pueden solucionarse escribiendo todo el programa de usuario linealmente en un OB de ciclo. Esto se recomienda sólo para programas sencillos.

La figura siguiente muestra esquemáticamente un programa lineal: El OB de ciclo "Main1" contiene el programa de usuario completo.



Programación estructurada

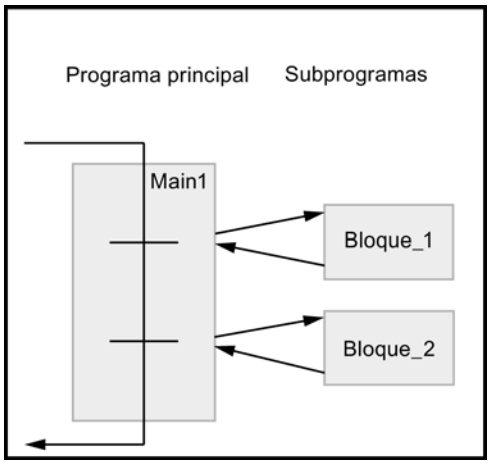
Para simplificar la creación y mantenimiento de tareas complejas de automatización, es conveniente subdividirlas en tareas parciales más pequeñas que correspondan a las diversas funciones tecnológicas del proceso de automatización o que puedan utilizarse varias veces.

En el programa de usuario, estas tareas parciales se representan mediante bloques. Cada bloque es una sección independiente del programa de usuario.

La estructuración del programa ofrece las ventajas siguientes:

- Los programas grandes pueden programarse de forma clara.
- Es posible estandarizar partes de programas y utilizarlas varias veces con diferentes parámetros.
- La organización del programa se simplifica.
- El programa se puede modificar más fácilmente.
- La comprobación del programa se simplifica, puesto que puede realizarse por secciones.
- La puesta en marcha se simplifica.

La figura siguiente muestra un esquema de un programa estructurado: El OB de ciclo "Main1" llama subprogramas consecutivamente que, a su vez, ejecutan tareas parciales definidas.



Sinopsis de los tipos de bloques

Tipos de bloques

Para realizar las tareas en un sistema de automatización se dispone de distintos tipos de bloques. La tabla siguiente muestra los tipos de bloques disponibles:

Tipo de bloque	Descripción breve
Bloques de organización (Página 2252) (OB)	Los bloques de organización determinan la estructura del programa de usuario.
Funciones (Página 2252) (FC)	Las funciones contienen rutinas de programa para tareas que se repiten con frecuencia. Carecen de "memoria".
Bloques de función (Página 2253) (FB)	Los bloques de función son bloques lógicos que depositan sus valores de forma permanente en bloques de datos instancia, de modo que siguen disponibles tras procesar el bloque.

Tipo de bloque	Descripción breve
Bloques de datos instancia (Página 2256)	Cuando se llama un bloque de función, se le asignan los bloques de datos instancia, en los que se almacenan los datos de programa.
Bloques de datos globales (Página 2254)	Los bloques de datos globales son áreas de datos para almacenar datos que pueden ser utilizados por cualquier bloque.

Bloques de organización (OB)

Definición

Los bloques de organización (OBs) constituyen la interfaz entre el sistema operativo y el programa de usuario. Son llamados por el sistema operativo y controlan p. ej. los siguientes procesos:

- Comportamiento en arranque del sistema de automatización
- Ejecución cíclica del programa
- Ejecución del programa controlada por alarmas
- Tratamiento de errores

Programando los bloques de organización se define el comportamiento de la CPU. En función de la CPU utilizada, se dispone de diferentes bloques de organización.

Encontrará más información acerca de los bloques de organización en las descripciones de los modos de funcionamiento de las CPU en el capítulo "Información adicional sobre configuraciones" de "Configurar dispositivos y redes".

Información de arranque de bloques de organización

Al arrancar algunos bloques de organización, el sistema operativo ofrece información que puede evaluarse en el programa de usuario. En las descripciones de los bloques de organización se indica si se ofrece información y de qué información se trata.

Consulte también

Crear bloques de organización (Página 2596)

Funciones (FC)

Definición

Las funciones (FCs) son bloques lógicos sin memoria. No poseen una memoria de datos que permita almacenar valores de parámetros de bloque. Por este motivo, al llamar una función es preciso asignar parámetros actuales a todos los parámetros formales.
Para almacenar los datos de forma permanente, las funciones disponen de bloques de datos globales.

Campo de aplicación

Una función contiene un programa que se ejecuta cada vez que la función es llamada por otro bloque lógico. Las funciones se pueden utilizar p. ej. para los siguientes fines:

- Devolver valores de función al bloque invocante, p. ej. en funciones matemáticas
- Ejecutar funciones tecnológicas, p. ej. controles individuales con operaciones lógicas binarias

Una función también se puede llamar varias veces en diferentes puntos de un programa. Esto facilita la programación de funciones de uso frecuente.

Nota

Transferencia de parámetros al llamar funciones

Para evitar errores al trabajar con funciones, consulte el capítulo "Transferencia de parámetros en la llamada de un bloque (Página 2268)".

Consulte también

Crear funciones y bloques de función (Página 2597)

Bloques de función (FB)

Definición

Los bloques de función son bloques lógicos que depositan sus parámetros de entrada, salida y entrada/salida de forma permanente en bloques de datos instancia, de modo que siguen estando disponibles después de editar el bloque. Por eso también se denominan "bloques con memoria".

Los bloques de función también pueden funcionar con variables temporales. No obstante, las variables temporales no se almacenan en el DB instancia, sino que únicamente permanecen disponibles durante un ciclo.

Campo de aplicación

Los bloques de función contienen subprogramas que se ejecutan cada vez que un bloque de función es llamado por otro bloque lógico. Un bloque de función también se puede llamar varias veces en diferentes puntos de un programa. Esto facilita la programación de funciones de uso frecuente.

Instancias de bloques de función

La llamada de un bloque de función se denomina instancia. Para cada instancia de un bloque de función se requiere un bloque de datos de instancia en el que se almacenen valores específicos de la instancia para los parámetros formales declarados en el FB.

El bloque de función puede depositar sus datos específicos de la instancia en un bloque de datos de instancia propio o en el bloque de datos de instancia del bloque que realiza la llamada.

Tipos de acceso

Los S7-1200 y S7-1500 ofrecen para los bloques de datos de instancia dos posibilidades de acceso diferentes que se asignan a un bloque de función al llamarlo:

- Bloques de datos con acceso optimizado
  Los bloques de datos de acceso optimizado carecen de una estructura de memoria definida. En la declaración se asigna a los elementos de datos un nombre simbólico, pero no una dirección fija dentro del bloque.
- Bloques de datos con acceso estándar (compatible con S7-300/400)
  Los bloques de datos de acceso estándar tienen una estructura de memoria fija. En la declaración se asigna a los elementos de datos un nombre simbólico y también una dirección fija dentro del bloque.

Nota

Para evitar errores al trabajar con bloques de función, consulte el capítulo "Transferencia de parámetros en la llamada de un bloque (Página 2268)".

Consulte también

Crear funciones y bloques de función (Página 2597)

Multiinstancias (Página 2266)

Bloques de datos instancia (Página 2256)

Fundamentos del acceso a bloques (Página 2258)

Bloques de datos globales (DB)

Definición

Los bloques de datos sirven para almacenar datos de programa. Así, los bloques de datos contienen datos variables con los que trabaja el programa de usuario. Los bloques de datos globales almacenan datos de usuario utilizables desde todos los demás bloques.

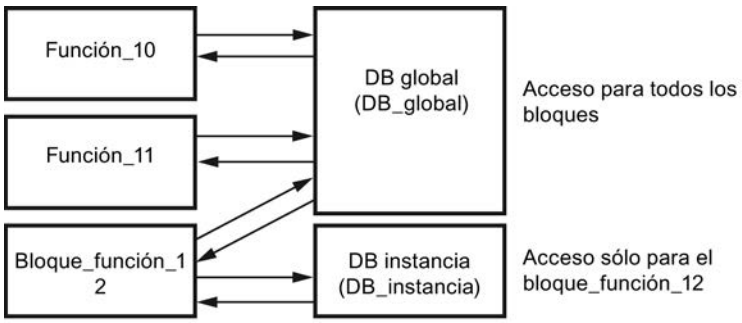
El tamaño máximo de los bloques de datos varía en función de la CPU. La estructura de bloques de datos globales puede definirse a discreción.

También es posible utilizar tipos de datos PLC (UDT) como plantilla para crear bloques de datos globales.

Bloques de datos globales en el programa de usuario

Todo bloque de función, toda función o todo bloque de organización puede leer datos de un bloque de datos global o escribir datos en este. Estos datos se conservan en el bloque de datos incluso al cerrarlo. Un bloque de datos global y un bloque de datos instancia pueden estar abiertos simultáneamente.

La figura siguiente muestra diferentes accesos a bloques de datos:



Tipos de acceso

Las S7-1200 y S7-1500 ofrecen dos posibilidades diferentes de acceso para bloques de datos globales:

- Bloques de datos con acceso optimizado
  Los bloques de datos con acceso optimizado carecen de una estructura definida. En la declaración se asigna a los elementos de datos un nombre simbólico, pero no una dirección fija dentro del bloque.
- Bloques de datos con acceso estándar (compatible con S7-300/400)
  Los bloques de datos con acceso estándar tienen una estructura fija. En la declaración se asigna a los elementos de datos un nombre simbólico y también una dirección fija dentro del bloque.

Bloques de datos ARRAY (S7-1500)

Un tipo especial de bloques de datos globales son los bloques de datos ARRAY. Están formados por un ARRAY de un tipo de datos cualquiera. Puede ser un ARRAY de un tipo de datos PLC (UDT), por ejemplo. El DB no contiene otros elementos más que el ARRAY. Gracias a su estructura plana, los bloques de datos ARRAY facilitan el acceso a los elementos ARRAY y su transmisión a bloques llamados.

Para los bloques de datos ARRAY, el atributo "Acceso optimizado al bloque" siempre está activado. No se admiten bloques de datos ARRAY con acceso estándar.

La sección "Transferencia" de la Task Card "Instrucciones" permite direccionar bloques de datos ARRAY.

Consulte también

Crear bloques de datos (Página 2598)

Programar tipos de datos PLC (Página 2967)

Fundamentos del acceso a bloques (Página 2258)

Programar bloques de datos (Página 2937)

Bloques de datos instancia

Definición

La llamada de un bloque de función se denomina instancia. Los datos con los que trabaja la instancia se almacenan en un bloque de datos instancia.

El tamaño máximo de los bloques de datos instancia varía en función de la CPU. Las variables declaradas en el bloque de función determinan la estructura del bloque de datos instancia.

Tipos de acceso

Los S7-1200 y S7-1500 ofrecen para los bloques de datos de instancia dos posibilidades de acceso diferentes que se asignan a un bloque de función al llamarlo:

- Bloques de datos con acceso optimizado
  Los bloques de datos con acceso optimizado carecen de una estructura definida. En la declaración se asigna a los elementos de datos un nombre simbólico, pero no una dirección fija dentro del bloque.
- Bloques de datos con acceso estándar (compatible con S7-300/400)
  Los bloques de datos con acceso estándar tienen una estructura fija. En la declaración se asigna a los elementos de datos un nombre simbólico y también una dirección fija dentro del bloque.

Consulte también: Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Consulte también

Crear bloques de datos (Página 2598)

Insertar llamadas de bloque en KOP (Página 2687)

Insertar llamadas de bloque en FUP (Página 2731)

Fundamentos del acceso a bloques (Página 2258)

Programar bloques de datos (Página 2937)

Bloques de datos de CPU

Definición

Los bloques de datos de CPU son bloques de datos que la CPU genera en runtime. Para ello, inserte en el programa de usuario la instrucción "CREATE_DB". Una vez que el bloque de datos se ha generado en runtime, se puede utilizar para guardar los datos.

Los bloques de datos de CPU se muestran dentro de un dispositivo accesible en la carpeta "Bloques de programa" con un pequeño símbolo de una CPU. Los valores actuales de las variables de un bloque de datos de CPU se pueden observar online como los de cualquier otro tipo de bloque de datos.

Los bloques de datos de CPU no se pueden crear offline en el proyecto.

Cargar bloques de datos de CPU

Una vez que el programa de usuario ha generado el bloque de datos de CPU con la instrucción "CREATE_DB", en un primer momento el bloque solo estará disponible en modo online en el dispositivo. En la siguiente carga completa en la que se carguen los bloques del dispositivo al proyecto también se cargarán en él todos los bloques de datos de CPU. Los bloques de datos de CPU se marcan con un pequeño símbolo de una CPU. Sin embargo, los bloques de datos de CPU ya no pueden volver a cargarse en el dispositivo.

Limitaciones de bloques de datos de CPU en el proyecto

Si los bloques de datos de CPU se cargaron en el proyecto offline con un proceso de carga, se podrán abrir y se podrá visualizar su contenido. No obstante, recuerde que los bloques de datos de CPU están protegidos contra escritura en el proyecto. Por ello, dentro del proyecto los bloques de datos de CPU están sometidos a las siguientes limitaciones:

- No es posible editar bloques de datos de CPU ni convertirlos a otro tipo de bloque de datos.
- Los bloques de datos de CPU no pueden dotarse con protección de know-how.
- El lenguaje de programación de los bloques de datos de CPU no puede cambiarse.
- Los bloques de datos de CPU no pueden compilarse ni cargarse en el dispositivo.

Comparar bloques de datos de CPU

Si los bloques de datos de CPU se cargaron en el proyecto offline con un proceso de carga, se podrá realizar una comparación online/offline de los bloques de datos de CPU cargados. En el editor de comparación se obtiene una vista general de las diferencias. No obstante, los bloques de datos de CPU que difieren en las versiones online y offline no pueden sincronizarse cargando la versión offline en el dispositivo.

Borrar bloques de datos de CPU

Los bloques de datos de CPU se pueden borrar tanto del proyecto como de la CPU.

Consulte también

Borrar bloques de datos de CPU (Página 2618)

Bloques con acceso optimizado

Fundamentos del acceso a bloques

Introducción

STEP 7 ofrece bloques de datos con distintas posibilidades de acceso:

- Bloques de datos con acceso optimizado (S7-1200/S7-1500)
- Bloques de datos con acceso estándar (S7-300 / S7-400 / S7-1200 / S7-1500)

Ambos tipos de bloques se pueden combinar en un programa.

Bloques de datos con acceso optimizado

Los bloques de datos con acceso optimizado carecen de una estructura definida. En la declaración se asigna a los elementos de datos un nombre simbólico, pero no una dirección fija dentro del bloque. Los elementos se disponen automáticamente en el área de memoria disponible del bloque de manera que no haya huecos en la memoria. De este modo se aprovecha al máximo la capacidad de memoria.

Las variables de estos bloques de datos se identifican por medio de su nombre simbólico. Para direccionar las variables debe indicarse su nombre simbólico. Por ejemplo, a la variable "Nivel" del DB "Datos" se accede del siguiente modo:
`"Datos".Nivel`

Los bloques con acceso optimizado ofrecen las ventajas siguientes:

- Es posible crear bloques de datos con una estructura cualquiera sin tener en cuenta la disposición física de los diferentes elementos de datos.
- El acceso a datos optimizados se realiza siempre con la mayor rapidez posible, puesto que el sistema optimiza y administra el almacenamiento de datos.
- Se excluyen los errores de acceso, p. ej. en caso de direccionamiento indirecto o desde la HMI.
- Las distintas variables pueden definirse como remanentes de forma selectiva.
- Los bloques optimizados disponen por defecto de una reserva de memoria que permite ampliar las interfaces de bloques de función o datos durante el funcionamiento. Los bloques modificados pueden cargarse sin necesidad de pasar la CPU a STOP y sin afectar a los valores de las variables ya cargadas.

Nota

El atributo "Acceso optimizado al bloque" siempre está activado para los bloques siguientes y no puede desmarcarse:

- bloques GRAPH
- bloques de datos ARRAY

Bloques de datos con acceso estándar

Los bloques de datos con acceso estándar tienen una estructura fija. En la declaración se asigna a los elementos de datos un nombre simbólico y también una dirección fija dentro del bloque. La dirección se muestra en la columna "Offset".

En estos bloques de datos, las variables se pueden direccionar de forma simbólica o absoluta.

```
"Datos".Nivel
DB1.DBW2
```

Remanencia con acceso optimizado o estándar

Cuando los datos se definen como remanentes, sus valores se conservan si se produce un fallo de tensión o alimentación. Una variable remanente no se inicializa tras el rearranque, sino que conserva el valor que tenía antes del fallo de tensión. Si una variable DB se define como remanente, queda depositada en el área de memoria remanente del bloque de datos.

Las posibilidades de ajustar la remanencia dependen del tipo de acceso del bloque:

- En los bloques de datos con acceso estándar no se puede definir por separado el comportamiento de remanencia de las distintas variables. La configuración de remanencia se aplica a todas las variables del bloque de datos.
- En los bloques de datos con acceso optimizado se puede definir el comportamiento de remanencia de las distintas variables.
  En las variables de tipo de datos estructurado, la configuración de remanencia se aplica siempre a la totalidad de la estructura. No es posible configurar por separado la remanencia de diferentes elementos dentro del tipo de datos.

Posibilidades de direccionamiento con acceso optimizado y estándar

Los bloques con acceso optimizado admiten únicamente accesos denominados "con seguridad de tipo". Dichos accesos direccionan variables utilizando solo el nombre simbólico. De este modo se descarta la posibilidad de que se produzcan incoherencias en el programa o errores de acceso, incluso en caso de cambios en el bloque o en su interfaz.

La tabla siguiente muestra las posibilidades de direccionamiento para datos optimizados:

Direccionamiento	Bloque con acceso estándar	Bloque con acceso optimizado
Direccionamiento simbólico	x	x
Direccionamiento indexado de ARRAYs		x
Accesos Slice	x	x
Solapamiento con AT	x	-
Direccionamiento absoluto	x	-
Direccionamiento indirecto mediante ANY	x	-
Direccionamiento indirecto mediante POINTER y VARIANT	x	solo con notación simbólica

Consulte también

Ajustar el acceso a un bloque (Página 2260)

Direccionar variables en bloques de datos (Página 2295)

Crear bloques (Página 2595)

Ajustar el acceso a un bloque

Introducción

El acceso al bloque se ajusta automáticamente al crear un bloque:

- En las CPU de la serie S7-1200/1500, cuando se genera un bloque este tiene acceso optimizado de forma predeterminada.
- En las CPU de la serie S7-300/S7-400, cuando se genera un bloque este tiene acceso estándar de forma predeterminada.

Si copia o migra un bloque a una CPU de otra serie, el acceso al bloque no cambia automáticamente. No obstante, puede ser conveniente modificar manualmente el acceso al bloque. p. ej. para poder utilizar todas las funciones de la CPU.

Tras cambiar el acceso al bloque, generalmente es necesario volver a compilar y cargar el programa.

ATENCIÓN
Acceso optimizado en bloques GRAPH El atributo "Acceso optimizado al bloque" siempre está activado para bloques GRAPH en la S7-1500 y no puede desmarcarse:

Procedimiento

Para ajustar el acceso al bloque, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" en el árbol del proyecto.
2. Haga clic con el botón derecho del ratón en el bloque cuyo acceso desee modificar.
3. Elija el comando "Propiedades" del menú contextual.
   Se abre el diálogo de propiedades del bloque.
4. Haga clic en "Atributos" en la navegación local.
5. Active o desactive la opción "Acceso optimizado al bloque".
6. Confirme las entradas con "Aceptar".

Limitaciones y particularidades

Básicamente el acceso al bloque solo puede cambiarse en CPU de las series S7-1200/1500, ya que solo estas soportan el tipo de acceso "optimizado".

Para ello existen las siguientes limitaciones y particularidades:

- Bloques de datos de instancia
  El acceso a los bloques de datos de instancia siempre se rige por el bloque de función asignado y no puede modificarse manualmente. Si cambia el tipo de acceso de un bloque de función, será necesario actualizar los bloques de datos de instancia asignados. Al actualizar se adapta el tipo de acceso del bloque de datos instancia.
- Bloques de sistema y bloques con protección de know-how
  En los bloques de sistema y en los bloques con protección de know-how no es posible modificar manualmente el acceso.
- Bloques de organización
  La información estándar de un OB con acceso estándar siempre se encuentra en la interfaz del bloque en los primeros 20 bytes de la sección "Temp". Por el contrario, la información de arranque de un OB con acceso optimizado se guarda en la sección "Input". Por ello, en el caso de los OB al cambiar el acceso al bloque también cambia la interfaz del bloque. Los detalles se describen en los capítulos siguientes.

Cambiar el acceso al bloque de "estándar" a "optimizado"

Al copiar o migrar un bloque de una CPU de las series S7-300/400 a una CPU S7-1200/1500, el bloque primero tendrá el tipo de acceso "estándar". Sin embargo, el rendimiento de la ejecución del programa mejora notablemente utilizando bloques con acceso optimizado. Por ello puede ser conveniente cambiar el tipo de acceso manualmente.

Al realizar el cambio, los bloques se adaptan del siguiente modo:

- Bloques de función
  Todos los parámetros de interfaz obtienen el ajuste de remanencia "no remanente".
- Bloques de datos globales
  El ajuste de remanencia que antes estaba ajustado de forma centralizada para todo el bloque de datos se transfiere a los diferentes parámetros de interfaz. Seguidamente es posible cambiar manualmente el ajuste de remanencia de los diferentes parámetros. Pero además rige la regla siguiente: En las variables de tipo de datos estructurado, la configuración de remanencia se aplica siempre a la totalidad de la estructura. No es posible configurar por separado la remanencia de diferentes elementos dentro de un tipo de datos estructurado. Así, en los bloques de datos basados en tipos de datos PLC tampoco se podrán realizar ajustes de remanencia individuales para las distintas variables.
- Bloques de organización
  Todos los parámetros de interfaz que se encuentran en los primeros 20 bytes de la sección "Temp" se borran. En la sección "Input" se crean nuevas informaciones de arranque específicas de la CPU. Si con ello aparecen conflictos de nombres con parámetros de interfaz definidos por el usuario, se cambia el nombre de estos últimos.

⚠ PRECAUCIÓN

El cambio del acceso al bloque tiene las siguientes consecuencias:

- Tras cambiar el acceso al bloque a "optimizado" ya no será posible direccionar los parámetros de interfaz del bloque de forma absoluta.
  Ejemplo: `#L0.1` ya no está permitido:
- Al cambiar el acceso a bloques de organización a "optimizado" cambia la interfaz del OB.

Debido a este cambio, puede ser necesario adaptar, volver a compilar y cargar el programa.

Consulte también: Compilar y cargar bloques (Página 3012)

Cambiar el acceso al bloque de "optimizado" a "estándar"

Para copiar o mover un bloque de una CPU de las series S7-1200/1500 a una CPU de las series S7-300/400, primero es necesario cambiar el acceso al bloque a "estándar".

Al realizar el cambio, los bloques se adaptan del siguiente modo:

- Bloques de función y bloques de datos globales:
  En el bloque de función ya no es posible realizar un ajuste de remanencia. El ajuste se realiza en el bloque de datos instancia.
  Todos los parámetros de interfaz del bloque de datos instancia o del bloque de datos global obtienen el mismo ajuste de remanencia. El cambio se produce conforme a la siguiente regla:
  - Si todos los parámetros de interfaz eran remanentes en el bloque de partida, tras el cambio todo el bloque será remanente.
  - Si todos los parámetros de interfaz eran no remanentes en el bloque de origen, tras el cambio todo el bloque será no remanente.
  - Si los parámetros de interfaz tenían diferentes ajustes de remanencia en el bloque de origen, todo el bloque será no remanente tras el cambio.
- Bloques de organización
  Se borran todos los parámetros de interfaz que se encuentran en la sección "Input". En la sección "Temp" se crean nuevas informaciones de arranque específicas de la CPU. Con ella se ocupan los primeros 20 bytes. Si con ello aparecen conflictos de nombres con parámetros de interfaz definidos por el usuario, se cambia el nombre de estos últimos.

⚠ PRECAUCIÓN

El cambio del acceso al bloque tiene las siguientes consecuencias:

Al cambiar el acceso al bloque a "estándar" los ajustes de remanencia de los parámetros de interfaz pueden cambiar. Debido a este cambio, puede ser necesario adaptar, volver a compilar y cargar el programa.

Consulte también: Compilar y cargar bloques (Página 3012)

Consulte también

Definir propiedades de bloques (Página 2607)

Fundamentos del acceso a bloques (Página 2258)

9.1.1.3 Llamadas de bloque

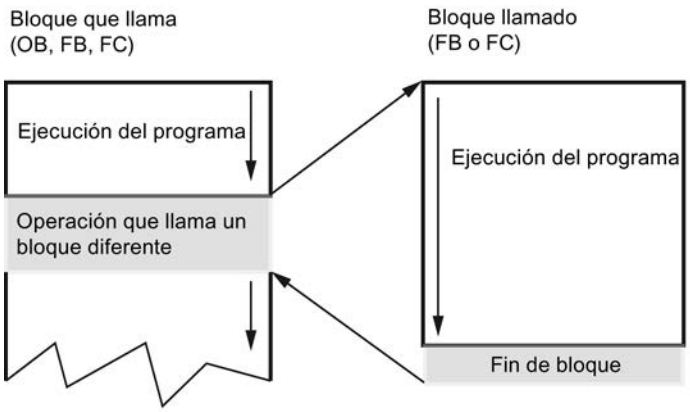
Principios básicos de las llamadas de bloque

Función de las llamadas de bloque

Para poder ejecutar los bloques en el programa de usuario, deben ser llamados desde un bloque diferente.

Si un bloque llama a otro bloque, se ejecutan las instrucciones del bloque llamado. El procesamiento del bloque invocante no se reanuda hasta que ha finalizado la ejecución del bloque llamado. La ejecución continúa con la instrucción que sigue a la llamada del bloque.

La figura siguiente muestra cómo se realiza una llamada de bloque en un programa de usuario:



Transferencia de parámetros

Al llamar un bloque, se asignan valores a los parámetros en la interfaz del bloque. Asignando valores a los parámetros de entrada se determina con qué datos debe procesarse el bloque. Asignando valores a los parámetros de salida se determina dónde deben almacenarse los resultados del procesamiento.

Consulte también

Jerarquía de llamada (Página 2264)

Principios básicos de las instancias individuales y multiinstancias (Página 2264)

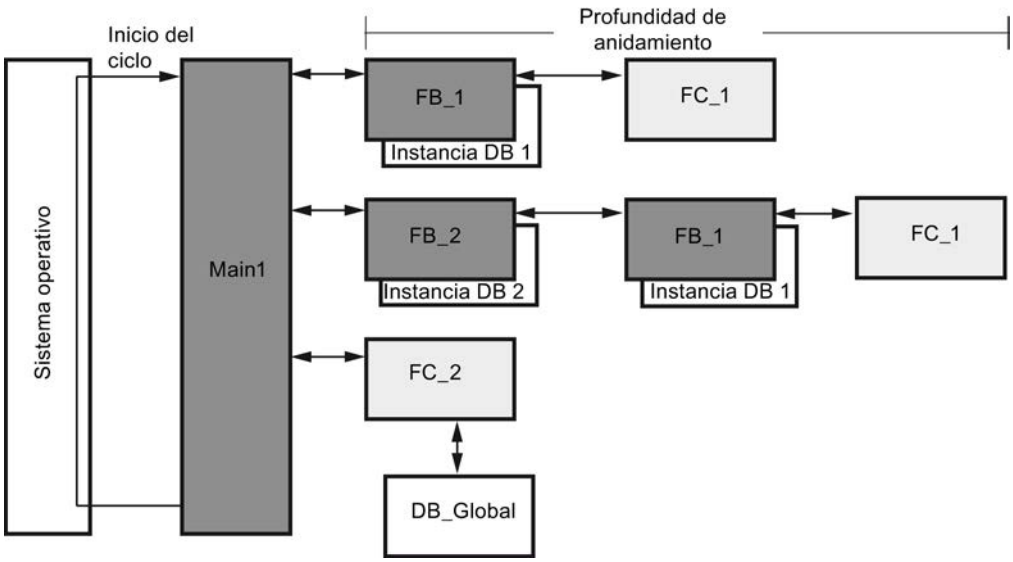
Transferencia de parámetros en la llamada de un bloque (Página 2268)

Jerarquía de llamada

Definición

Por jerarquía de llamada se entiende el orden y la profundidad de anidamiento de las llamadas de bloque. La profundidad de anidamiento permitida depende de la CPU.

La figura siguiente muestra un ejemplo del orden y la profundidad de anidamiento de las llamadas de bloque en un ciclo de ejecución:



Consulte también

Principios básicos de las instancias individuales y multiinstancias (Página 2264)

Principios básicos de las llamadas de bloque (Página 2263)

Llamada de bloques de función como instancia individual o multiinstancia

Principios básicos de las instancias individuales y multiinstancias

Utilización de instancias individuales y multiinstancias

Los bloques de función (FB) depositan sus datos en bloques de datos instancia. Los bloques de datos instancia almacenan los valores de los parámetros de bloque y los datos locales estáticos de los bloques de función.

Los bloques de datos instancia pueden asignarse como se indica a continuación:

- Instancia individual:
  Un bloque de datos instancia por cada instancia de un bloque de función
- Multiinstancia:
  Un bloque de datos de instancia para la instancia de un bloque de función y todas las instancias de los bloques de función llamados por él.

Consulte también

Direccionar variables en bloques de datos (Página 2295)

Principios básicos de las llamadas de bloque (Página 2263)

Multiinstancias (Página 2266)

Instancias individuales (Página 2265)

Jerarquía de llamada (Página 2264)

Instancias individuales

Definición

Por instancia individual se entiende la llamada de un bloque de función al que está asignado un bloque de datos instancia propio.

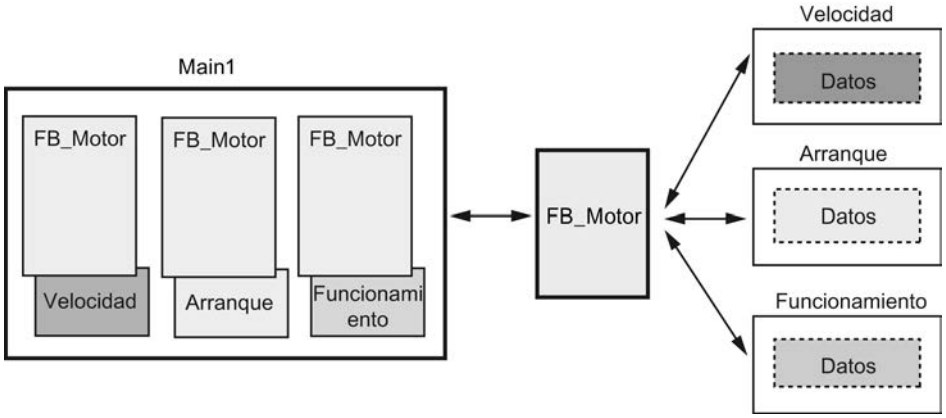
Al asignar el bloque de datos instancia se indica dónde deben guardarse los datos de instancia del FB. Asignando a cada llamada un bloque de datos instancia diferente, se puede utilizar el mismo FB varias veces con distintos datos de instancia.

Ejemplo de una instancia individual

Con un bloque de función pueden controlarse varios motores. A este efecto, se asigna a cada llamada del bloque de función un bloque de datos instancia diferente para el control del motor.

Los diferentes datos de cada uno de los motores (p. ej. número de revoluciones, tiempo de aceleración, tiempo total de servicio) se almacenan en los distintos bloques de datos instancia. En función del bloque de datos instancia asignado se controla un motor diferente.

La figura siguiente muestra el control de tres motores con un bloque de función y tres bloques de datos diferentes:



Consulte también

Principios básicos de las instancias individuales y multiinstancias (Página 2264)

Multiinstancias (Página 2266)

Multiinstancias

Definición

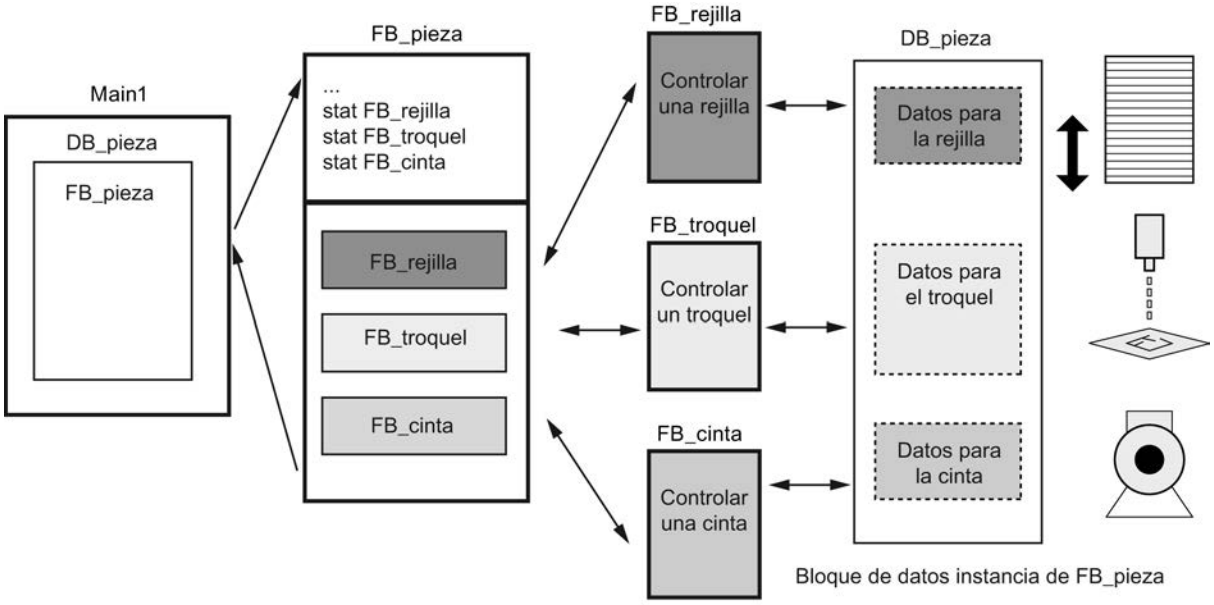
Las multiinstancias ofrecen a un bloque de función llamado la posibilidad de almacenar sus datos en el bloque de datos instancia del bloque de función que llama.

Esto permite concentrar los datos de instancia en un bloque de datos instancia y aprovechar mejor el número disponible de bloques de datos instancia.

Un bloque de datos instancia para las instancias de distintos bloques de función

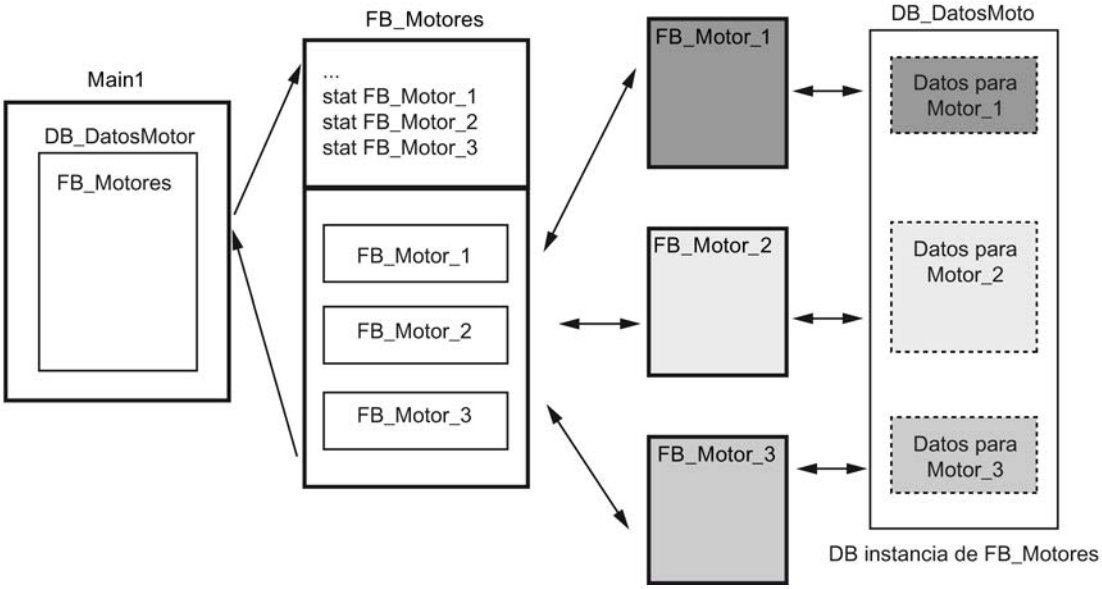
La figura siguiente muestra cómo varios bloques de función diferentes depositan sus datos en un bloque invocante. El FB_pieza llama sucesivamente el FB_rejilla, el FB_troquel y el

FB_cinta. Los bloques llamados depositan sus datos en el DB_pieza, que es el bloque de datos instancia del bloque que llama.



Un bloque de datos instancia para varias instancias de un bloque de función

La figura siguiente muestra cómo un bloque de función, que se llama en varias instancias, deposita los datos de todas las instancias en un bloque de datos instancia.



El bloque de función FB_motores llama tres instancias del FB_motor. Las instancias son "Motor_1", "Motor_2" y "Motor_3". Para cada llamada se utilizan diferentes datos de instancia. Sin embargo, todos los datos de instancia se almacenan en un único bloque de datos instancia, el DB_datosmotor.

Consulte también

Principios básicos de las instancias individuales y multiinstancias (Página 2264)

Instancias individuales (Página 2265)

Transferencia de parámetros en la llamada de un bloque

Principios básicos de los parámetros de bloque

Introducción

El bloque que llama envía al bloque llamado los valores con los que debe trabajar. Estos valores se denominan parámetros de bloque. Los parámetros de entrada comunican al bloque llamado los valores que debe procesar. El bloque devuelve los resultados a través de los parámetros de salida.

Así, los parámetros de bloque constituyen la interfaz entre el bloque que llama y el bloque llamado.

Si solo se desea consultar o leer valores, se utilizan parámetros de entrada, y si solo se desea definir o escribir valores, se utilizan parámetros de salida. Para leer y escribir parámetros de bloque, es necesario crearlos como parámetros de entrada/salida.

Parámetros formales y actuales

Los parámetros de bloque se definen en la interfaz del bloque llamado. Estos parámetros se denominan parámetros formales. Sirven de comodines para los valores que se transfieren al bloque al llamarlo. Los valores transferidos en la llamada se denominan parámetros actuales.

Reglas para utilizar los parámetros de bloque

Para la utilización de los parámetros de bloque dentro de un bloque rigen las reglas siguientes:

- Los parámetros de entrada solo pueden leerse.
- Los parámetros de salida solo pueden escribirse.
- Los parámetros de entrada/salida pueden leerse y escribirse.

Consulte también

Asignar valores a parámetros de bloques de función (Página 2272)

Asignar valores a parámetros de funciones (Página 2270)

Reglas generales para la transferencia de valores de parámetros (Página 2269)

Variables (Página 2286)

Palabras clave (Página 2280)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Asignar valores a los parámetros de bloque al llamarlos

Reglas generales para la transferencia de valores de parámetros

Introducción

Al llamar un bloque con parámetros de bloque, se transfieren parámetros actuales a sus parámetros formales. Para ello rigen las reglas indicadas a continuación.

Tipos de datos apropiados

Los tipos de datos de los parámetros actuales y formales deben ser idénticos o convertibles según las reglas de conversión de tipos de datos.

Transferir ARRAYs

Los ARRAYs se pueden transferir como parámetros. Si un bloque tiene un parámetro de entrada del tipo ARRAY, es preciso transferir un ARRAY con idéntica estructura como parámetro actual. Si el elemento tiene el mismo tipo de datos que el parámetro formal, los distintos elementos de un ARRAY también se podrán transferir como parámetros actuales.

Transferir tipos de datos PLC

Las variables declaradas como tipo de datos PLC también pueden transferirse como parámetros actuales. Si el parámetro formal está declarado en la declaración de variables como tipo de datos PLC, es necesario transferir como parámetro actual una variable que sea del mismo tipo de datos PLC.

El elemento de una variable declarada mediante un tipo de datos PLC también puede transferirse como parámetro actual al llamar un bloque, siempre y cuando el tipo de datos del elemento de la variable coincida con el tipo de datos del parámetro formal.

Transferir estructuras (STRUCT)

Las estructuras se pueden transferir como parámetros. Si un bloque tiene un parámetro de entrada del tipo STRUCT, es preciso transferir un STRUCT con idéntica estructura como parámetro actual. Esto implica que los nombres y los tipos de datos de todos los componentes de estructura deben ser idénticos.

Si el elemento tiene el mismo tipo de datos que el parámetro formal, los distintos elementos de un STRUCT también se podrán transferir como parámetros actuales.

Nota

Se recomienda programar estructuras como tipos de datos PLC. Los tipos de datos PLC facilitan la programación, puesto que pueden utilizarse varias veces y modificarse de forma centralizada.

Consulte también

Asignar valores a parámetros de bloques de función (Página 2272)

Asignar valores a parámetros de funciones (Página 2270)

Principios básicos de los parámetros de bloque (Página 2268)

Bases de AWL (Página 2753)

Direcciones totalmente especificadas en AWL (Página 229)

Direcciones parcialmente especificadas en AWL (Página 230)

Tipos de datos PLC (Página 2367)

Asignar valores a parámetros de funciones

Parámetros de funciones (FC)

Las funciones no poseen una memoria de datos que permita almacenar valores de parámetros de bloque. Por este motivo, al llamar una función es preciso asignar parámetros actuales a todos los parámetros formales.

Parámetros de entrada (Input)

Los parámetros de entrada solo se leen una vez por ciclo, antes de la llamada del bloque. Por ello rige la norma de que la escritura de un parámetro de entrada dentro del bloque no afecta al parámetro actual. Solo se escribe el parámetro formal.

Parámetros de salida (Output)

Los parámetros de salida solo se escriben una vez por ciclo, tras la llamada del bloque. Por ello rige la norma de que los parámetros de salida dentro del bloque no deben leerse. No obstante, si lee un parámetro de salida, recuerde que solo se lee el valor del parámetro formal. El valor del parámetro actual no puede leerse dentro del bloque.

Si un parámetro de salida de una función no se escribe en dicha función, se utiliza el valor predefinido para el tipo de datos indicado. Por ejemplo, el valor predefinido para BOOL es "false". No obstante, a los parámetros de salida estructurados no se les asigna ningún valor.

Para no continuar procesando el valor predefinido o un valor indefinido de forma accidental, recuerde lo siguiente al programar el bloque:

- Asegúrese de que en los parámetros de salida se registran valores para cada ruta de programa posible dentro del bloque. Recuerde, p. ej., que los comandos de salto pueden saltar sobre secuencias de instrucciones en las que se activan salidas.
- Tenga en cuenta que las instrucciones de activación y desactivación dependen del resultado lógico. Si el valor de un parámetro de salida se determina con estas instrucciones, no se forma ningún valor si el resultado lógico (RLO) es 0.
- Si es posible, asigne un valor predeterminado a los parámetros de salida de las funciones.

Parámetros de entrada/salida (InOut)

Los parámetros de entrada/salida se leen antes de la llamada del bloque y se escriben después de ella. Si se lee o escribe el parámetro dentro del bloque, solo se accede a su parámetro formal.

Los parámetros de entrada/salida con tipo de datos estructurado son una excepción. Los tipos de datos estructurados son los que se componen de varios elementos de datos, p. ej. ARRAY o STRUCT. Estos se transfieren al bloque llamado a través de un POINTER. Así, siempre se accede al parámetro actual si se lee o escribe un parámetro de entrada/salida estructurado dentro de un bloque.

Si no se escribe un parámetro de entrada/salida de una función en dicha función, el valor de salida o de entrada antiguo se utiliza como valor. No obstante, también en este caso debe observar las indicaciones anteriores sobre parámetros de salida para no seguir procesando valores antiguos involuntariamente.

Datos locales temporales (Temp)

Los datos locales temporales solo están disponibles durante un ciclo. Reciben un tratamiento diferente en función del tipo de bloque:

- Acceso estándar
  En los bloques lógicos con acceso estándar y para todas las variables con el ajuste de remanencia "Ajustar en IDB" rige la siguiente regla:
  Si utiliza datos locales temporales, se debe asegurar de que los valores se escriben dentro del ciclo en el que desea leerlos. De lo contrario, los valores serán aleatorios. Los datos temporales del tipo de datos STRING constituyen una excepción: se les asigna automáticamente la información de longitud correcta.
- Acceso optimizado
  En los bloques lógicos con acceso optimizado, se aplica la siguiente regla:
  Si una variable temporal no se escribe dentro de una función, se utiliza el valor predefinido para el tipo de datos indicado. Por ejemplo, el valor predefinido para BOOL es "false". A los elementos de los tipos de datos PLC se les preasigna el valor predeterminado especificado en la declaración del tipo de datos PLC (UDT). A los elementos ARRAY se les preasigna el valor "0", aunque se utilicen en un tipo de datos PLC. A los strings se les asigna la información de longitud correcta.

Valor de función (Return)

Por lo general, las funciones calculan un valor de función. Dicho valor puede devolverse al bloque que realiza la llamada a través del parámetro de salida RET_VAL. Para ello, el parámetro de salida RET_VAL debe estar declarado en la interfaz de la función. RET_VAL es siempre el primer parámetro de salida de una función. Para el parámetro RET_VAL se admiten todos los tipos de datos excepto ARRAY y STRUCT, así como los tipos de parámetros TIMER y COUNTER.

El lenguaje de programación SCL permite llamar directamente funciones en una expresión. El resultado de la expresión se obtiene a partir del valor de función calculado. Por este motivo, en SCL no se admite el tipo de datos ANY para el valor de función.

Consulte también

Asignar valores a parámetros de bloques de función (Página 2272)

Principios básicos de los parámetros de bloque (Página 2268)

Reglas generales para la transferencia de valores de parámetros (Página 2269)

Llamada de funciones (Página 2814)

Ejemplos de la llamada de funciones en SCL (Página 2817)

Asignar valores a parámetros de bloques de función

Asignar valores a parámetros de bloques de función (FB)

En el caso de los bloques de función, los valores de parámetro se guardan en los datos de instancia.

Si no se han transferido valores a los parámetros de entrada, salida y entrada/salida de un bloque de función, se utilizan los valores almacenados.

En algunos casos es imprescindible definir un parámetro actual.

La tabla siguiente muestra a qué parámetros de un bloque de función se deben asignar parámetros actuales:

Parámetros	Tipo de datos simple	Tipo de datos estructurado	Tipo de parámetro
Entrada (Input)	opcional	opcional	necesario
Salida (Output)	opcional	opcional	necesario
Entrada/salida (InOut)	opcional	necesario	Soportado solo en S7-1200; requiere transferencia de valores de parámetro
Temporal (Temp)	necesario S7-1500: opcional con acceso optimizado al bloque	necesario S7-1500: opcional con acceso optimizado al bloque	necesario

Consulte también

Principios básicos de los parámetros de bloque (Página 2268)

Reglas generales para la transferencia de valores de parámetros (Página 2269)

Asignar valores a parámetros de funciones (Página 2270)

Tipos de parámetros (Página 2366)

Acceso a parámetros de bloque durante la ejecución del programa

Introducción

Los parámetros de bloque de funciones y bloques de función se procesan de forma diferente durante la ejecución del programa. El tipo de acceso varía según la familia de la CPU, el tipo de bloque y el tipo de datos del parámetro de bloque.

Generalmente se diferencia entre los dos siguientes tipos de acceso:

- Los parámetros se transfieren como puntero de parámetro
  Al bloque llamado se transfiere un puntero de parámetro.
  Esto significa que el bloque llamado puede acceder directamente al operando especificado como parámetro actual. La escritura de un parámetro en el bloque llamado hace que se modifique el parámetro actual en el bloque que efectúa la llamada. Los accesos de lectura a un parámetro de bloque leen directamente el parámetro actual.
- Los parámetros se transfieren como copia
  Durante la llamada del bloque, el valor del parámetro actual se copia en los datos temporales del bloque llamado.
  Esto significa que el bloque llamado trabaja siempre con el valor que tenía el parámetro actual al producirse la llamada del bloque. No puede acceder directamente al operando especificado como parámetro actual. La escritura de un parámetro en el bloque llamado no hace que se modifique el parámetro actual en el bloque invocante. Los accesos de lectura acceden al parámetro formal, no al actual.
  El procedimiento de copia es el siguiente:
  - Parámetros de entrada:
    En la llamada del bloque, el valor del parámetro actual se copia en el parámetro formal del bloque llamado.
  - Parámetros de salida:
    El valor del parámetro formal se copia en el parámetro actual al salir del bloque.
  - Parámetros de entrada/salida:
    Antes de la llamada del bloque, el valor del parámetro actual se copia en el parámetro formal del bloque llamado y, después de salir de la función, se copia nuevamente en el parámetro actual.

Transferencia de parámetros con tipos de datos simples

En la siguiente tabla se muestra como las diferentes familias de CPU transfieren parámetros de bloque con tipos de datos simples. Tipos de datos simples son, p. ej.: Bool, Int, Byte.

Bloque llamado	Parámetro actual en el área	S7-300/400	S7-1500
FC	I, Q, M, P, L	Puntero	Puntero
	Dirección DB parcialmente especificada (p. ej. DBW 2)	Puntero	En S7-1500 no se soportan direcciones parcialmente especificadas.

Bloque llamado	Parámetro actual en el área	S7-300/400	S7-1500
	Dirección DB totalmente especificada (p. ej. "MyDB".Valor), parámetros transferidos del bloque que llama, parámetros estáticos del bloque que llama	Copia	Copia
FB	Todos los parámetros actuales	Copia	Copia

Transferencia de parámetros con tipos de datos estructurados

En la siguiente tabla se muestra cómo las diferentes familias de CPU transfieren parámetros de bloque con tipos de datos estructurados. Los tipos de datos estructurados son los que se componen de varios elementos, p. ej. ARRAY o STRUCT.

Bloque llamado	Parámetro actual en el área	S7-300/400	S7-1500
FC	IN, OUT, InOut	Puntero	Puntero
FB	IN, OUT	Copia	Copia
	InOut	Puntero	Puntero

Reenviar parámetros de bloque

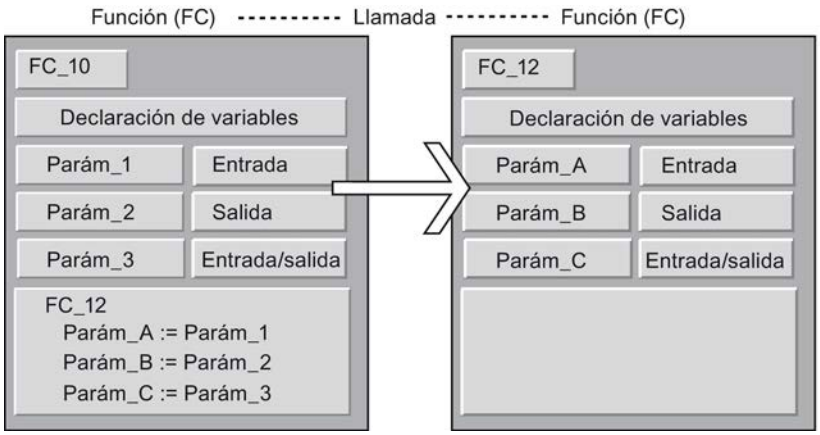
Fundamentos del reenvío de parámetros de bloque

Introducción

Definición

Existe un modo especial de uso de valores de parámetros denominado "reenvío" de parámetros de bloque. Consiste en reenviar los valores de los parámetros del bloque que llama a los parámetros del bloque llamado. El bloque llamado utiliza como parámetros actuales los valores asignados actualmente a los parámetros de bloque del bloque que llama.

La figura siguiente muestra cómo se reenvían los parámetros de la función FC_10 a la función FC_12:



Reglas para KOP/FUP

En KOP y FUP rigen las siguientes reglas generales:

- Los parámetros de entrada solo pueden reenviarse a parámetros de entrada.
- Los parámetros de salida solo pueden reenviarse a parámetros de salida.
- Los parámetros de entrada/salida pueden enviarse a cualquier tipo de parámetro.
- En los S7-300/400, ambos parámetros de bloque deben ser del mismo tipo de datos.
- En los S7-1200/1500, los parámetros también se pueden convertir según las normas de conversión implícita.

Reglas para AWL

En AWL rigen las siguientes reglas generales:

- Los parámetros de entrada solo pueden reenviarse a parámetros de entrada.
- Los parámetros de salida solo pueden reenviarse a parámetros de salida.
- Los parámetros de entrada/salida pueden enviarse a cualquier tipo de parámetro.
- Ambos parámetros de bloque deben ser del mismo tipo de datos. Esta regla rige para AWL en todas las familias de CPU.

Reglas para SCL

Las reglas que rigen para SCL son menos estrictas. Para que los programas de versiones anteriores de SCL puedan adoptarse con mayor facilidad, existen otras posibilidades a la hora de transferir parámetros, pero también se advierte sobre ellas. De este modo es posible, p. ej., reenviar un parámetro de entrada/salida a un parámetro de entrada, si bien se recibe una advertencia porque el programa no puede escribir el parámetro de entrada/salida transferido.

En los capítulos siguientes se describen otras reglas en detalle.

Consulte también

Llamada de una función desde otra función (Página 2276)
Llamada de una función desde un bloque de función (Página 2277)
Llamada de un bloque de función desde una función (Página 2278)
Llamada de un bloque de función desde otro bloque de función (Página 2279)

Llamada de una función desde otra función

Tipos de datos permitidos al llamar una función desde otra función

El reenvío de parámetros formales está regido por determinadas reglas. La tabla siguiente muestra las reglas que rigen para poder reenviar parámetros a las diferentes familias de CPU:

FC llama a FC		Tipos de datos					
Parámetro actual (bloque que efectúa la llamada)	Parámetro formal (bloque llamado)	Tipos de datos estándar	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Tipos de parámetros (TIMER, COUNTER, BLOCK_XX)	DB_Any
Input	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 a partir de V2 S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-

Consulte también

Fundamentos del reenvío de parámetros de bloque (Página 2274)

Llamada de una función desde un bloque de función

Tipos de datos permitidos al llamar una función desde un bloque de función

El reenvío de parámetros formales está regido por determinadas reglas. La tabla siguiente muestra las reglas que rigen para poder reenviar parámetros a las diferentes familias de CPU:

FB llama a FC		Tipos de datos					
Parámetro actual (bloque que efectúa la llamada)	Parámetro formal (bloque llamado)	Tipos de datos estándar	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Tipos de parámetros (TIMER, COUNTER, BLOCK_XX)	DB_Any
Input	Input	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 a partir de V2 S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-

Consulte también

Fundamentos del reenvío de parámetros de bloque (Página 2274)

Llamada de un bloque de función desde una función

Tipos de datos permitidos al llamar un bloque de función desde una función

El reenvío de parámetros formales está regido por determinadas reglas. La tabla siguiente muestra las reglas que rigen para poder reenviar parámetros a las diferentes familias de CPU:

FC llama a FB		Tipos de datos					
Parámetro actual (bloque que efectúa la llamada)	Parámetro formal (bloque llamado)	Tipos de datos estándar	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Tipos de parámetros (TIMER, COUNTER, BLOCK_XX)	DB_Any
Input	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-300/400 S7-1500	S7-1200 a partir de V2 S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-

Consulte también

Fundamentos del reenvío de parámetros de bloque (Página 2274)

Llamada de un bloque de función desde otro bloque de función

Tipos de datos permitidos al llamar un bloque de función desde otro bloque de función

El reenvío de parámetros formales está regido por determinadas reglas. La tabla siguiente muestra las reglas que rigen para poder reenviar parámetros a las diferentes familias de CPU:

FB llama a FB		Tipos de datos					
Parámetro actual (bloque que efectúa la llamada)	Parámetro formal (bloque llamado)	Tipos de datos estándar	ARRAY, STRUCT, STRING, WSTRING, DT	ANY, POINTER	VARIANT	Tipos de parámetros (TIMER, COUNTER, BLOCK_XX)	DB_Any
Input	Input	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	S7-300/400 S7-1500	S7-1200 a partir de V2 S7-1500
Output	Output	S7-300/400 S7-1200 S7-1500	S7-300/400 S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	Input	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-
InOut	Output	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	-	S7-1200 S7-1500	-	-
InOut	InOut	S7-300/400 S7-1200 S7-1500	S7-1200 S7-1500	S7-1500	S7-1200 S7-1500	-	-

Consulte también

Fundamentos del reenvío de parámetros de bloque (Página 2274)

9.1.1.4 Utilizar y direccionar operandos

Fundamentos de los operandos

Introducción

Al programar instrucciones se indican los valores de datos que la instrucción debe procesar. Estos valores se denominan operandos. P. ej. pueden usarse como operandos los siguientes elementos:

- Variables PLC
- Constantes
- Variables de bloques de datos instancia
- Variables de bloques de datos globales

Dirección absoluta y nombre simbólico

Los operandos se identifican mediante una dirección absoluta y un nombre simbólico. Los nombres y direcciones se definen en la tabla de variables PLC o en la declaración de variables de los bloques.

Bloques de datos de acceso optimizado (S7-1200, S7-1500)

A los elementos de datos de los bloques de datos con acceso optimizado se les asigna en la declaración un nombre simbólico, pero no una dirección absoluta. Encontrará más información al respecto en "Consulte también".

Consulte también

Mostrar operandos simbólicos y absolutos (Página 2630)

Fundamentos del acceso a bloques (Página 2258)

Palabras clave

SIMATIC conoce toda una serie de palabras clave que tienen un determinado significado en el programa. Estas palabras clave no deben usarse como nombres de variables ni de constantes.

Tabla de palabras clave

En la tabla siguiente figuran todas las palabras clave.

Palabras clave Nemónicos alemanes	Palabras clave Nemónicos ingleses	Descripción
&	&	Operación lógica Y de expresiones lógicas
A	Q	Salida, bit
A1	CC1	Bit indicador
A0	CC0	Bit indicador
AB	QB	Salida, byte
AD	QD	Salida, palabra doble
AND	AND	Operación lógica Y de expresiones lógicas
ANY	ANY	Tipo de datos, puntero
AR1	AR1	Registro de direcciones 1
AR2	AR2	Registro de direcciones 2
ARRAY	ARRAY	Inicio de la especificación de un array, seguida de la lista de índices entre "[" y "]"
AT	AT	Declaración de variables superpuestas
AUTHOR	AUTHOR	Nombre del autor, empresa, departamento u otro nombre (máx. 8 caracteres, sin espacios)
AW	QW	Salida, palabra
B	B	Byte
BEGIN	BEGIN	Inicio de la sección de instrucciones de bloques lógicos o sección de inicialización de bloques de datos
BIE	BR	Resultado binario
BLOCK_FB	BLOCK_FB	Tipo de parámetro para especificar un FB
BLOCK_FC	BLOCK_FC	Tipo de parámetro para especificar una FC
BLOCK_SDB	BLOCK_SDB	Tipo de parámetro para especificar un SDB
BOOL	BOOL	Tipo de datos
BY	BY	Incremento del bucle FOR
BYTE	BYTE	Tipo de datos
CALL	CALL	Llamada
CASE	CASE	Introducción de la instrucción CASE
CHAR	CHAR	Tipo de datos simple

Palabras clave Nemónicos alemanes	Palabras clave Nemónicos ingleses	Descripción
CODE_VERSION1	CODE_VERSION1	Identificación de si un FB es apto para multiinstancia o no. Si deben declararse multiinstancias, el FB no puede tener esta propiedad.
CONST	CONST	Inicio de la declaración de constante
CONTINUE	CONTINUE	Instrucción para salir de un bucle en SCL
COUNTER	COUNTER	Tipo de parámetro para especificar un contador
DATA_BLOCK	DATA_BLOCK	Inicio del bloque de datos
DATE	DATE	Tipo de datos
DATE_AND_TIME	DATE_AND_TIME	Tipo de datos
DB	DB	Bloque de datos
DB_ANY	DB_ANY	Tipo de datos
DBB	DBB	Bloque de datos, byte de datos
DBD	DBD	Bloque de datos, palabra doble de datos
DBLG	DBLG	Longitud del bloque de datos
DBNO	DBNO	Número del bloque de datos
DBW	DBW	Bloque de datos, palabra de datos
DBX	DBX	Bloque de datos, bit de datos
DI	DI	Bloque de datos de instancia
DIB	DIB	Bloque de datos de instancia, byte de datos
DID	DID	Bloque de datos de instancia, palabra doble de datos
DILG	DILG	Longitud de bloque de datos de instancia
DINO	DINO	Número de bloque de datos de instancia
DINT	DINT	Tipo de datos
DIW	DIW	Bloque de datos de instancia, palabra de datos
DIX	DIX	Bloque de datos de instancia, bit de datos
DO	DO	Inicio de la sección de instrucciones de FOR y WHILE
DT	DT	Tipo de datos
DTL	DTL	Tipo de datos
DWORD	DWORD	Tipo de datos
E	I	Entrada (mediante memoria imagen de proceso), bit

Palabras clave Nemónicos alemanes	Palabras clave Nemónicos ingleses	Descripción
EB	IB	Entrada (mediante memoria imagen de proceso), byte
ED	ID	Entrada (mediante memoria imagen de proceso), palabra doble
ELSE	ELSE	Rama alternativa en la instrucción IF y CASE
ELSIF	ELSIF	Condición alternativa de la instrucción IF
EN	EN	Operando de sistema del mecanismo EN/ENO
ENO	ENO	Operando de sistema del mecanismo EN/ENO
END_CASE	END_CASE	Fin de la instrucción CASE
END_DATA_BLOCK	END_DATA_BLOCK	Fin del bloque de datos
END_FOR	END_FOR	Fin de la instrucción FOR
END_FUNCTION	END_FUNCTION	Fin de la función
END_FUNCTION_BLOCK	END_FUNCTION_BLOCK	Fin del bloque de función
END_IF	END_IF	Fin de la instrucción IF
END_ORGANIZATION_BLOCK	END_ORGANIZATION_BLOCK	Fin del bloque de organización
END_REPEAT	END_REPEAT	Fin de la instrucción REPEAT
END_STRUCT	END_STRUCT	Fin de la especificación de una estructura
END_SYSTEM_FUNCTION	END_SYSTEM_FUNCTION	Fin de la función de sistema
END_SYSTEM_FUNCTION_BLOCK	END_SYSTEM_FUNCTION_BLOCK	Fin del bloque de función de sistema
END_TYPE	END_TYPE	Fin del tipo de datos PLC
END_VAR	END_VAR	Fin de un bloque de declaración
END_WHILE	END_WHILE	Fin de la instrucción WHILE
EW	IW	Entrada (mediante memoria imagen de proceso), palabra
EXIT	EXIT	Instrucción para salir de un bucle en SCL
FALSE	FALSE	Constante booleana predefinida: Condición lógica no se cumple, valor igual a 0
FAMILY	FAMILY	Nombre de la familia de bloques: p. ej. regulador
FB	FB	Bloque de función
FC	FC	Función
FOR	FOR	Inicio de la instrucción FOR
FUNCTION	FUNCTION	Inicio de la función
FUNCTION_BLOCK	FUNCTION_BLOCK	Inicio del bloque de función
GOTO	GOTO	Inicio de la instrucción GOTO
IF	IF	Inicio de la instrucción IF

Palabras clave Nemónicos alemanes	Palabras clave Nemónicos ingleses	Descripción
INSTANCE	INSTANCE	Tipo de datos
INT	INT	Tipo de datos
KNOW_HOW_PROTECT	KNOW_HOW_PROTECT	Protección de bloque
L	L	Bit de datos locales
LB	LB	Byte de datos locales
LD	LD	Palabra doble de datos locales
LDT	LDT	Tipo de datos
LINT	LINT	Tipo de datos
LTIME	LTIME	Tipo de datos
LTOD	LTOD	Tipo de datos
LW	LW	Palabra de datos locales
LWORD	LWORD	Tipo de datos
M	M	Bit de marcas
MB	MB	Byte de marcas
MD	MD	Palabra doble de marcas
MOD	MOD	Operador módulo
MW	MW	Palabra de marcas
NAME	NAME	Nombre del bloque
NETWORK	NETWORK	Segmento
NOT	NOT	Inversión lógica
NULL	NULL	Puntero Null
OB	OB	Bloque de organización
OF	OF	Inicio de la especificación del tipo de datos/de la sección de instrucciones de la instrucción CASE
OR	OR	Operación lógica O de expresiones lógicas
ORGANIZATION_BLOCK	ORGANIZATION_BLOCK	Inicio del bloque de organización
OS	OS	Desbordamiento, con memoria
OV	OV	Desbordamiento
PA	PQ	Salida (periferia directa), bit
PAB	PQB	Salida (periferia directa), byte
PAD	PQD	Salida (periferia directa), palabra doble
PAW	PQW	Salida (periferia directa), palabra
PE	PI	Entrada (periferia directa), bit
PEB	PIB	Entrada (periferia directa), byte
PED	PID	Entrada (periferia directa), palabra doble
PEW	PIW	Entrada (periferia directa), palabra
POINTER	POINTER	Tipo de datos

Palabras clave Nemónicos alemanes	Palabras clave Nemónicos ingleses	Descripción
READ_ONLY	READ_ONLY	Protección contra escritura para bloques de datos
REAL	REAL	Tipo de datos
REPEAT	REPEAT	Inicio de la instrucción REPEAT
RET_VAL	RET_VAL	Valor de retorno
RETURN	RETURN	Instrucción RETURN en SCL
S5T	S5T	Sintaxis para el tipo de datos S5TIME
S5TIME	S5TIME	Tipo de datos
S7_	S7_	Palabras clave para atributos del sistema
SDB	SDB	Bloque de datos de sistema
SFB	SFB	Bloque de función de sistema
SFC	SFC	Función de sistema
SINT	SINT	Tipo de datos
STRING	STRING	Tipo de datos
STRUCT	STRUCT	Inicio de la especificación de una estructura seguido de la lista de componentes
STW	STW	Palabra de estado
SYSTEM_FUNCTION	SYSTEM_FUNCTION	Función de sistema
SYSTEM_FUNCTION_BLOCK	SYSTEM_FUNCTION_BLOCK	Bloque de función de sistema
T	T	Temporizador (timer)
THEN	THEN	Inicio de la sección de instrucciones de una instrucción IF
THIS	THIS	Sintaxis para el acceso a un bloque de datos ARRAY
TIME	TIME	Tipo de datos simple para indicaciones horarias
TIME_OF_DAY	TIME_OF_DAY	Tipo de datos
TIMER	TIMER	Tipo de parámetro para especificar un temporizador
TITLE	TITLE	Título opcional de bloque o de segmento
TO	TO	Definición del valor final de una instrucción FOR
TOD	TOD	Tipo de datos
TRUE	TRUE	Constante booleana predefinida: Condición lógica se cumple, valor diferente de 0
TYPE	TYPE	Inicio del tipo de datos PLC
UDT	UDT	Tipo de datos PLC o global
UDINT	UDINT	Tipo de datos
UINT	UINT	Tipo de datos

Palabras clave Nemónicos alemanes	Palabras clave Nemónicos ingleses	Descripción
ULINT	ULINT	Tipo de datos
UNLINKED	UNLINKED	Identificación 'no relevante para la ejecución'
UNTIL	UNTIL	Fin de la sección de instrucciones de una instrucción REPEAT
USINT	USINT	Tipo de datos
UO	UO	Consulta de (A1=1) Y (A0=1)
VAR	VAR	Inicio de un bloque de declaración
VAR_IN_OUT	VAR_IN_OUT	Inicio de un bloque de declaración
VAR_INPUT	VAR_INPUT	Inicio de un bloque de declaración
VAR_OUTPUT	VAR_OUTPUT	Inicio de un bloque de declaración
VAR_TEMP	VAR_TEMP	Inicio de un bloque de declaración
VARIANT	VARIANT	Tipo de datos
VERSION	VERSION	Número de versión del bloque
VOID	VOID	La función no tienen ningún valor de retorno
WCHAR	WCHAR	Tipo de datos
WSTRING	WSTRING	Tipo de datos
WHILE	WHILE	Inicio de la instrucción WHILE
WORD	WORD	Tipo de datos
XOR	XOR	Operación lógica
Z	C	Contador (counter)

Variables

Definición

Una variable es un comodín que ocupa en un programa el lugar de un valor de datos modificable. El formato del valor de datos está establecido. La utilización de variables aumenta la flexibilidad del programa. Por ejemplo, a las variables declaradas en la interfaz de un bloque se pueden asignar valores diferentes en cada llamada del bloque. De esta manera, un bloque programado se puede reutilizar para distintos fines.

Una variable consta de los siguientes elementos:

- Nombre
- Tipo de datos

- Dirección absoluta
  - En los bloques con acceso estándar, las variables PLC y las variables DB tienen una dirección absoluta.
  - En los bloques con acceso optimizado, las variables DB no tienen una dirección absoluta.
- Valor (opcional)

Declaración de variables

En el programa se pueden definir variables con distintos ámbitos de validez:

- Variables PLC válidas en toda la CPU
- Variables DB de bloques de datos globales que pueden ser utilizadas por todos los bloques en toda la CPU.
- Variables DB de bloques de datos de instancia que se utilizan prioritariamente en el bloque en el que están declaradas.

La tabla siguiente muestra la diferencia entre los distintos tipos de variables:

	Variables PLC	Variables de bloques de datos de instancia	Variables de bloques de datos globales
Ámbito de validez	• Válidas en toda la CPU. • Pueden ser utilizadas por todos los bloques de la CPU. • El nombre es unívoco dentro de la CPU.	• Se utilizan prioritariamente en el bloque en el que están definidas. • El nombre es unívoco dentro del bloque de datos de instancia.	• Pueden ser utilizadas por todos los bloques de la CPU. • El nombre es unívoco dentro del bloque de datos global.
Caracteres permitidos	• Letras, cifras, caracteres especiales • No se permiten las comillas. • No se permiten las palabras clave reservadas.	• Letras, cifras, caracteres especiales • No se permiten las palabras clave reservadas.	• Letras, cifras, caracteres especiales • No se permiten las palabras clave reservadas.
Aplicación	• Señales E/S (I, IB, IW, ID, Q, QB, QW, QD) • Marcas (M, MB, MW, MD)	• Parámetros de bloque (parámetros de entrada, salida y entrada/salida) • Datos estáticos de un bloque	• Datos estáticos
Ubicación de la definición	Tabla de variables PLC	Interfaz del bloque	Tabla de declaración del bloque de datos global

Consulte también

Palabras clave (Página 2280)

Fundamentos de los operandos (Página 2280)

Mostrar operandos simbólicos y absolutos (Página 2630)

Nombres de variables PLC permitidos (Página 2570)

Direcciones y tipos de datos admisibles para variables PLC (Página 2570)

Constantes

Principios básicos de las constantes

Definición

Las constantes son datos que tienen un valor fijo que no cambia durante el tiempo de ejecución del programa. Las constantes pueden ser leídas por diferentes elementos del programa durante su ejecución, pero no se pueden sobrescribir. Para el valor de una constante hay notaciones definidas según el tipo y formato de datos. Básicamente se distingue entre notación con tipo y notación sin tipo.

Constantes sin tipo

En la notación sin tipo únicamente se introduce el valor de la constante, pero no un tipo de datos. Las constantes sin tipo obtienen su tipo de datos con la combinación aritmética o lógica en la que se utilizan.

El siguiente ejemplo muestra la notación de constantes sin tipo:

SCL

```
#My_Int1 := #My_Int2 + 12345        (*El tipo de datos de la constante "12345"
                                     resulta de la suma con My_Int 2. "12345" recibe
                                     el tipo de datos INT.*)
#My_Real1 := #My_Real2 + 12345      (*El tipo de datos de la constante "12345"
                                     resulta de la suma con My_Real2. "12345" recibe
                                     el tipo de datos REAL.*)
```

Constantes con tipo

En la notación con tipo, además del valor de la constante se indica explícitamente un tipo de datos.

El siguiente ejemplo muestra la notación de constantes con tipo:

SCL

```
#My_Int1 :=  INT#12345                    (*El tipo de datos de la constante es
                                          siempre INT.*)
```

Nota

Constantes del tipo BOOL

En S7-300/400 las constantes del tipo BOOL no pueden utilizarse como entradas en KOP o FUP.

Información adicional

Información adicional sobre tipos de datos y constantes así como sus formatos de entrada y rangos de valores:

Tipos de datos (Página 2321)

Consulte también

Declarar constantes globales (Página 2583)

Declarar variables y constantes locales (Página 2648)

Estructura de la interfaz del bloque (Página 2642)

Declaración de nombres simbólicos para constantes

Constantes simbólicas

Existe la posibilidad de declarar nombres simbólicos para constantes, a fin de poder utilizar en el programa valores constantes con un nombre. De ese modo se consigue una mejor legibilidad y un mantenimiento más sencillo del programa en caso de cambio de valores constantes.

Una constante simbólica está compuesta por los siguientes elementos:

- Nombre
- Tipo de datos
  Las constantes simbólicas tienen siempre un tipo de datos y no admiten la notación sin tipo.
- Valor de la constante
  Se puede elegir, como valor de constante, cualquier valor situado dentro del rango de valores del tipo de datos indicado. Encontrará indicaciones sobre los rangos de valores en el capítulo "Tipos de datos".

Declaración de constantes

Es posible definir constantes con ámbitos de validez distintos:

- Constantes globales válidas en toda la CPU
- Constantes locales válidas únicamente dentro de un bloque

La tabla siguiente muestra la diferencia entre los distintos tipos de constantes:

	Constantes globales	Constantes locales
Ámbito de validez	• Válidas en toda la CPU • El nombre es unívoco dentro de la CPU.	• Válidas en el bloque en el que fueron declaradas. • El nombre es unívoco dentro del bloque.
Caracteres permitidos	• Los caracteres permitidos en el nombre de una constante son letras, números y caracteres especiales.	• Los caracteres permitidos en el nombre de una constante son letras, números y caracteres especiales.
Ubicación de la definición	Ficha "Constantes" de la tabla de variables PLC	Interfaz del bloque
Representación	Entre comillas, p. ej.: `"Glob_Const"`	Con una almohadilla antepuesta, p. ej.: `#Loc_Const`

Nota

Cargar la declaración de constantes (S7-300/400)

Las declaraciones de constantes locales y globales no se cargan en la CPU. Al cargar un programa de un dispositivo es posible que ya no estén disponibles las declaraciones de constantes.

Información adicional

Información adicional sobre el procedimiento para declarar constantes:

Tipos de datos (Página 2321)

Declarar constantes globales (Página 2583)

Declarar variables y constantes locales (Página 2648)

Tipos de datos de las constantes

Tipos de datos permitidos

Para las constantes se admiten todos los tipos de datos básicos y sus derivados:

- Números binarios
- Secuencias de bits
- Enteros
- Números en coma flotante
- Temporizadores
- Fecha y hora
- Cadenas de caracteres

Rigen las reglas generales sobre la conversión explícita e implícita de tipos de datos.

Tipos de datos de constantes sin tipo

Las constantes sin tipo no contienen información explícita sobre el tipo de datos. Obtienen su tipo de datos con la combinación aritmética o lógica en la que se utilizan.

El siguiente ejemplo muestra el tratamiento de constantes sin tipo:

SCL

```
#My_Int1 := #My_Int2 + 12345        (*El tipo de datos de la constante "12345"
                                     resulta de la suma con My_Int 2. "12345" recibe
                                     el tipo de datos INT.*)
#My_Real1 := #My_Real2 + 12345      (*El tipo de datos de la constante "12345"
                                     resulta de la suma con My_Real2. "12345" recibe
                                     el tipo de datos REAL.*)
```

Nota

STEP 7 utiliza siempre el tipo de datos con la precisión más alta posible

Mientras dentro de una expresión no sea posible definir de forma unívoca el tipo de datos de una constante, se utilizará siempre el tipo de datos disponible en la CPU actual que tenga la precisión más alta posible.

Ejemplo:
`#My_Real := #My_Int / 3.5`

En esta expresión se combina una variable de entero con una constante de coma flotante sin tipo. En S7-300/400 se calcula la parte derecha de la asignación en formato REAL. En S7-1200/1500, el cálculo se realiza automáticamente con la precisión más alta posible, es decir, en este caso en LREAL. Como consecuencia, la asignación a una variable REAL no es válida o bien genera una advertencia.

Para definir con precisión el tipo de datos de una constante, utilice la notación con tipo:

Ejemplo:

`#My_Real := #My_Int / REAL#3.5`

Información adicional

Información adicional sobre tipos de datos y constantes así como sus formatos de entrada y rangos de valores:

Tipos de datos (Página 2321)

Consulte también

Declarar constantes globales (Página 2583)

Declarar variables y constantes locales (Página 2648)

Ejemplos de uso de constantes

Uso en instrucciones, asignaciones y expresiones

Las constantes pueden utilizarse en instrucciones o asignaciones en lugar de variables. En SCL también es posible utilizar constantes en expresiones. Puesto que las constantes no pueden escribirse, solo pueden utilizarse como entradas.

El siguiente ejemplo muestra los usos posibles de constantes:

SCL

```
#My_Int :=   3;
#My_Real1 := #My_Real2 * 3;
#My_Real1 := #My_Real2 * #My_local_const;
#My_Real1 := #My_Real2 * "My_global_const";
```

Uso como valor predeterminado

Una constante también puede utilizarse como valor predeterminado de una variable. Para hacerlo, introduzca el valor o el nombre simbólico de la constante en la columna "Valor predeterminado" de la interfaz del bloque. El tipo de datos de la constante debe ser igual al tipo de datos de la variable o bien debe ser convertible de acuerdo con las reglas de la conversión implícita de tipos de datos con comprobación IEC.

Es posible que se produzcan conflictos de nombres cuando se ha declarado con el mismo nombre simbólico una constante local y una global y se utiliza el nombre asignado por duplicado como valor predeterminado de una variable. En este caso se utiliza automáticamente la constante local.

Uso como límite de ARRAY

Las constantes locales del tipo de datos INT o DINT también pueden utilizarse como límites de ARRAY.

El ejemplo siguiente muestra el uso de constantes locales como límites de ARRAY:

SCL

```
Array[My_local_const1..My_local_const1] of REAL
```

Nota

Constantes como límites de ARRAY

Tenga en cuenta las siguientes indicaciones en lo que respecta a las "Constantes como límites de ARRAY":

- Las constantes globales no pueden utilizarse como límites de ARRAY.
- Las constantes locales que se utilizan como límites de ARRAY no se pueden modificar si está activada la reserva de memoria del bloque.

Consulte también

Declarar constantes globales (Página 2583)

Declarar variables y constantes locales (Página 2648)

Tipos de datos (Página 2321)

Direccionar operandos

Direccionar variables globales

Direccionar variables globales

Para direccionar una variable PLC global se puede utilizar la dirección absoluta o el nombre simbólico.

Nota

Los tipos de datos LWORD, LINT, ULINT, LREAL, LTIME, LTOD y LDT solo pueden ser direccionados con su nombre simbólico.

Direccionar variables globales de forma simbólica

Para el direccionamiento simbólico debe usarse el nombre de variable procedente de la tabla de variables PLC. El nombre simbólico de las variables globales se encierra automáticamente entre comillas.

Las variables estructuradas que se basan en un tipo de datos PLC se direccionan con el nombre simbólico de las variables PLC. También es posible indicar los nombres de los diferentes componentes separados por un punto.

Direccionar variables globales de forma absoluta

Para el direccionamiento absoluto indique la dirección de la variable obtenida de la tabla de variables PLC. Para el direccionamiento absoluto se utilizan direcciones numéricas que

empiezan por cero para cada área de operandos. Delante de la dirección absoluta de las variables globales se antepone automáticamente el identificador de dirección %.

Ejemplos

Ejemplos de aplicaciones del direccionamiento simbólico y absoluto:

Direccionamiento	Significado
%Q1.0	Dirección absoluta: Salida 1.0
%I16.4	Dirección absoluta: Entrada 16.4
%IW4	Dirección absoluta: Palabra de entrada 4
"Motor"	Dirección simbólica "Motor"
"Value"	Dirección simbólica "Value"
"Variable_estructurada"	Dirección simbólica de una variable basada en un tipo de datos PLC
"Variable_estructurada".Componente	Dirección simbólica del componente de una variable estructurada.

Consulte también: Direcciones y tipos de datos admisibles para variables PLC (Página 2570)

Consulte también

Mostrar operandos simbólicos y absolutos (Página 2630)

Declarar variables PLC (Página 2566)

Direccionar la periferia (Página 2294)

Direccionar la periferia

Descripción

La memoria imagen de proceso de la CPU se actualiza una vez por ciclo. No obstante, en aplicaciones de tiempo crítico puede ser necesario leer o transferir el estado actual de una entrada o salida digital con una frecuencia mayor. Para ello se puede acceder directamente a la periferia mediante un identificador de acceso a periferia en el operando.

Para leer la entrada directamente desde la periferia, utilice el área de memoria de entradas de periferia (PI) como contraposición a la memoria imagen de proceso de las entradas (I). El área de memoria de periferia se puede leer en formato de bits, bytes, palabras o palabras dobles.

Para leer la salida directamente desde la periferia, utilice el área de memoria de las salidas de periferia (PQ) en contraposición a la memoria imagen de proceso de las salidas (Q). El área de memoria de salidas de periferia se puede escribir en formato de bits, bytes, palabras o palabras dobles.

Para leer o escribir una señal directamente desde una entrada de periferia se puede agregar a un operando la identificación de acceso a la periferia ":P".

Los componentes de variables PLC estructuradas también pueden direccionarse con ":P". Sin embargo, no es posible acceder a la variable de nivel superior con ":P".

> ⚠ ADVERTENCIA
>
> **Escritura directa en la periferia**
>
> La escritura directa en la periferia puede provocar estados peligrosos, p. ej. cuando se direcciona varias veces en un ciclo de programa una salida de periferia en escritura.

Sintaxis

```
<Operando>:P
```

Ejemplo

Ejemplo de aplicaciones de la identificación de acceso a la periferia:

Direccionamiento	Significado
"Motor"	Direcciona la variable "Motor" en la memoria imagen de proceso.
"Motor":P	Direcciona la variable "Motor" en el área de memoria de periferia (PI o PQ).
"Variable_estructurada".Componente	Direcciona el componente de una variable PLC estructurada en la memoria imagen de proceso.
"Variable_estructurada".Componente:P	Direcciona el componente de una variable PLC estructurada en el área de memoria de periferia (PI o PQ).

Consulte también

Direccionar variables globales (Página 2293)

Direccionar variables en bloques de datos

Direccionar variables en bloques de datos globales

Descripción

En los bloques de datos globales, las variables se pueden direccionar de forma simbólica o absoluta. Para el direccionamiento simbólico, se utiliza el nombre del bloque de datos y el nombre de la variable, separados por un punto. El nombre del bloque de datos se pone entre comillas.

Para el direccionamiento absoluto, se utiliza el número del bloque de datos y la dirección absoluta de la variable en el bloque de datos, separados por un punto. Delante de la dirección absoluta se antepone automáticamente el identificador de dirección %.

En el S7-1200/1500 es posible acceder a un bloque de datos que se desconoce en el momento de la programación. Para ello se crea un parámetro de bloque del tipo de datos DB_Any en la interfaz del bloque que accede. En este parámetro se transmite el nombre o el número del bloque de datos en tiempo de ejecución. Para acceder a variables dentro del bloque de datos utilice el nombre del parámetro de bloque del tipo de datos DB_Any y la dirección absoluta de la variable separada por un punto.

Nota

Las siguientes variables no pueden ser direccionadas de forma absoluta:

- Variables de bloques con acceso optimizado.
- Variables de los tipos de variables LWORD, LINT, ULINT, LREAL, LTIME, LTOD y LDT.

Utilice para esas variables el direccionamiento simbólico, más cómodo.

Bloques de datos ARRAY

Un tipo especial de bloques de datos globales son los bloques de datos ARRAY. Están formados por un ARRAY de un tipo de datos cualquiera. Puede ser un ARRAY de un tipo de datos PLC (UDT), por ejemplo.

Los elementos de bloques de datos ARRAY se direccionan mediante la palabra clave "THIS". A continuación, el índice se indica entre corchetes. El índice puede ser tanto una constante como una variable. Como variables para el índice, se admiten números enteros de hasta 32 bits de ancho.
El área "Transferencia" de la Task Card "Instrucciones" ofrece opciones avanzadas para direccionar bloques de datos ARRAY. Estas instrucciones ofrecen, p. ej., la posibilidad de direccionar también el nombre del DB de forma indirecta.

Sintaxis

```
"<NombreDB>".NombreVariable
%<NúmeroDB>.DireccIónAbsoluta
#<Nombre_DBAny>.%DireccIónAbsoluta
"<ArrayDBname".THIS[#i].<Component>.<ComponentElement>
```

SCL:

```
"<ArrayDBname"."THIS"[#i].<Component>.<ComponentElement>
```

La tabla siguiente muestra las direcciones absolutas posibles de variables en bloques de datos:

Tipo de datos	Dirección absoluta	Ejemplo	Significado
BOOL	%DBn.DBXx.y	%DB1.DBX1.0	Bit de datos 1.0 en el DB1
BYTE, CHAR, SINT, USINT	%DBn.DBBy	%DB1.DBB1	Byte de datos 1 en el DB1

Tipo de datos	Dirección absoluta	Ejemplo	Significado
WORD, INT, UINT	%DBn. DBWy	%DB1.DBW1	Palabra de datos 1 en el DB1
DWORD, DINT, UDINT, REAL, TIME	%DBn.DBDy	%DB1.DBD1	Palabra de datos doble 1 en el DB1

Ejemplo

Ejemplos de direccionamiento de variables en bloques de datos globales:

Direccionamiento	Significado
"Motor".Value	Direccionamiento simbólico de la variable "Value" en el bloque de datos global "Motor".
%DB1.DBX1.0	Direccionamiento absoluto de la variable "DBX1.0" en el bloque de datos global "DB1".
#MyDBAny.%DBX30.0	Direccionamiento absoluto de la variable "DBX30.0" en el bloque de datos global, que en runtime se transmite en el parámetro "MyDBany".
"MyARRAY_DB".THIS[#MyIndex].MyComponent.MyComponentElement	Direccionamiento de un bloque de datos ARRAY. El índice de ARRAY se indica a través de la variable "MyIndex". El elemento ARRAY contiene dos subestructuras más: "MyComponent" y "MyComponentElement".

Consulte también

Direccionar variables estructuradas (Página 2298)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Fundamentos del direccionamiento indirecto (Página 2305)

Direccionar datos de instancia (Página 2297)

Direccionar datos de instancia

Descripción

Es posible direccionar elementos de datos desde la interfaz del bloque actual. Estas variables se almacenan en el bloque de datos instancia.

Nota

En los bloques de datos con acceso optimizado, las variables solo se pueden direccionar de forma simbólica.

Para direccionar una variable desde la interfaz del bloque actual, debe introducirse el carácter # seguido del nombre simbólico de la variable.

También es posible acceder a las variables de un bloque multiinstancia. Dentro del bloque multiinstancia también deben direccionarse los datos con el carácter # seguido del nombre de la variable. Para acceder a los datos del bloque multiinstancia desde el bloque que efectúa la llamada se utiliza #<NombreMultiinstancia.NombreVariable>.

Sintaxis

Para direccionar variables en bloques de datos instancia se utiliza la siguiente sintaxis:

```
#<NombreVariable>
#<NombreMultiinstancia.NombreVariable>
```

Ejemplos

Ejemplos de direccionamiento de variables en bloques de datos instancia:

Direccionamiento	Significado
#Value	Direccionamiento de la variable "Value" en el bloque de datos instancia.
#On	Direccionamiento de la variable "On" dentro del bloque multiinstancia
#Multi.On	Direccionamiento de la variable "On" del bloque multiinstancia desde el bloque que llama

Consulte también

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Direccionar variables en bloques de datos globales (Página 2295)

Direccionar variables estructuradas (Página 2298)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Fundamentos del direccionamiento indirecto (Página 2305)

Direccionar variables estructuradas

Direccionar elementos de datos de un ARRAY

Se accede a un elemento de un ARRAY mediante la sintaxis `ArrayName[i,j,k...]`.

El índice del elemento se indica entre corchetes. El índice contiene un valor entero por dimensión de ARRAY (-2147483648 ... 2147483647).

Los errores de acceso se producen cuando se accede en tiempo de ejecución a un elemento que está fuera de los límites de ARRAY declarados. Las diferentes familias de CPU reaccionan de diferente manera a los rebases de los límites de ARRAY:

- S7-300/400
  - La CPU cambia al estado operativo "STOP".
  - Para evitar esto, se puede programar el OB de error de ejecución del programa (OB 85).
  - En SCL también existe la posibilidad de activar el atributo "Comprobar límites de ARRAY" en las propiedades del bloque. Esto hace que, en caso de errores de acceso a ARRAY, la salida de habilitación ENO se ponga a FALSE.
- S7-1200
  - La CPU genera una entrada en el búfer de diagnóstico y permanece en estado operativo "RUN".
- S7-1500
  - La CPU cambia al estado operativo "STOP".
  - Para evitar esto, se puede programar el OB de error de programación (OB 121).
  - Además, existe la posibilidad de programar el tratamiento de errores local con las instrucciones "GET_ERROR: Consultar error localmente" o "GET_ERROR_ID: Consultar ID de error localmente".

Nota

Vigilancia de errores de acceso a ARRAY mediante ENO

La salida de habilitación ENO no adopta el estado lógico FALSE cuando se superan los límites de ARRAY al ejecutar una instrucción. Son excepción los bloques SCL en CPU de las series S7-300/400, para las que está activada la propiedad de bloque "Comprobar límites de ARRAY".

Consulte también:

Matriz (Página 2354)

Indexación indirecta de componentes de ARRAY (Página 2307)

Direccionar bloques de datos ARRAY

Una forma especial de ARRAY son los bloques de datos ARRAY. Los bloques de datos ARRAY son bloques de datos globales que constan exactamente de un ARRAY. Los elementos de bloques de datos ARRAY se direccionan con la siguiente sintaxis:

```
"<GlobArrayDBname".THIS[#i].<componentname>."<elementname>"
```

SCL:

```
"<GlobArrayDBname"."THIS"[#i].<componentname>."<elementname>".
```

El área "Transferencia" de la Task Card "Instrucciones" ofrece opciones avanzadas para direccionar bloques de datos ARRAY. Estas instrucciones ofrecen, p. ej., la posibilidad de direccionar también el nombre del DB de forma indirecta.

Direccionar elementos de datos en estructuras

A los distintos elementos de una estructura se accede a través de `NombreEstructura.NombreElemento`.

Consulte también:

Estructuras (Página 2358)

Direccionar elementos de datos de un tipo de datos PLC

Para acceder a los elementos de un tipo de datos PLC se utiliza la sintaxis `NombreTipoDatosPLC.NombreElemento`.

Consulte también:

Programar tipos de datos PLC (Página 2967)

Direccionar caracteres individuales de un STRING o WSTRING

Se accede a caracteres individuales de una variable STRING o WSTRING con la sintaxis `StringName[i]`. El índice de contaje "i" empieza por "1". Así pues, con el StringName[1] se accede al primer carácter del string.

No es posible acceder a caracteres individuales de una constante STRING o WSTRING.

Los errores se producen cuando se accede en tiempo de ejecución a un carácter que está fuera de la longitud de STRING. Las diferentes familias de CPU reaccionan de distinto modo a los rebases de la longitud de STRING:

- S7-300/400
  - Si la instrucción dispone de la salida de habilitación ENO, ENO adopta el estado lógico FALSE.
  - Si la instrucción no dispone de la salida de habilitación ENO, no se vigila el rebase de la longitud de STRING:
    Si el acceso se encuentra fuera del STRING, se accede al área de memoria incorrecta.
    Si el acceso se encuentra fuera del bloque de datos, la CPU pasa a STOP. Para evitar esto, se puede programar el OB de error de ejecución del programa (OB 85).
- S7-1200/1500
  - Si el acceso se encuentra fuera de la longitud declarada, la CPU pasa a STOP. Es posible elegir si se debe reaccionar a rebases con tratamiento de errores global de la CPU o con un tratamiento de errores local propio.
  - Si el acceso se encuentra dentro de la longitud de STRING declarada, pero fuera de la longitud efectiva, el rebase de la longitud de STRING no se vigila y la CPU no pasa a STOP. Si en el acceso de lectura se obtiene el carácter '$00' o '$0000', no se ejecuta ningún acceso de escritura.
    Si la instrucción dispone de la salida de habilitación ENO, ENO adopta el estado lógico FALSE.

Consulte también:

Cadenas de caracteres (Página 2349)

Ejemplos:

Ejemplos de direccionamiento de variables de tipo de datos estructurado:

Direccionamiento	Significado
Motor.Value_1x3[2]	Direccionamiento de una matriz unidimensional
Motor.Value_2x4[2,4]	Direccionamiento de una matriz bidimensional
Motor.Value_4x7[2,4,1,3]	Direccionamiento de una matriz tetradimensional
Batch_1.Temperature	Direccionamiento del elemento "Temperature" en la estructura "Batch_1"
Values.Temperature	Direccionamiento del elemento "Temperature" en la variable "Values", basada en un tipo de datos PLC.
STRING[3]	Direcciona el tercer carácter del string.

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Direccionar áreas de una variable con accesos Slice

Descripción

Existe la posibilidad de direccionar áreas concretas dentro de variables declaradas. Se puede acceder a áreas de 1 bit, 8 bits, 16 bits o 32 bits. Este tipo de acceso se denomina "acceso Slice".

Las constantes no pueden direccionarse mediante accesos Slice.

Sintaxis

Para el direccionamiento se utiliza la sintaxis siguiente:

```
<Variable>.X<Número de bit>
<Variable>.B<Número de BYTE>
<Variable>.W<Número de WORD>
<Variable>.D<Número de DWORD>
```

La sintaxis está formada por las partes siguientes:

Parte	Descripción
<Variable>	Variable a la que se accede. La variable debe ser del tipo de datos "Secuencia de bits". Si la verificación CEI está activada, también resulta posible acceder a variables del tipo de datos "Entero".
X	Identificación de la anchura de acceso "Bit (1 bit)"
B	Identificación de la anchura de acceso "Byte (8 bits)"
W	Identificación de la anchura de acceso "Palabra (16 bits)"
D	Identificación de la anchura de acceso "DWord (32 bits)"

Parte	Descripción
<Número de BIT>	Número del bit al que se accede dentro de la <Variable>. El número 0 accede al BIT menos significativo.
<Número de BYTE>	Número del byte al que se accede dentro de la <Variable>. El número 0 accede al BYTE menos significativo.
<Número WORD>	Número de la palabra a la que se accede dentro de la <Variable>. El número 0 accede a la WORD menos significativa.
<Número DWORD>	Número de la DWord a la que se accede dentro de la <variable>. El número 0 accede a la DWord menos significativa.

Ejemplos

Ejemplos de direccionamiento de bits individuales:

Direccionamiento	Significado
"Engine".Motor.X0 "Engine".Motor.X7	"Motor" es una variable del tipo de datos BYTE, WORD, DWORD o LWORD en el bloque de datos global "Engine". X0 direcciona la dirección de bit 0, X7 la dirección de bit 7 dentro de "Motor".
"Engine".Speed.B0 "Engine".Speed.B1	"Speed" es una variable del tipo de datos WORD, DWORD o LWORD en el bloque de datos global "Engine". B0 direcciona la dirección de byte 0, B1 la dirección de byte 1 dentro de "Speed".
"Engine".Fuel.W0 "Engine".Fuel.W1	"FUEL" es una variable del tipo de datos DWORD o LWORD en el bloque de datos global "Engine". W0 direcciona la dirección de palabra 0, W1 la dirección de palabra 1 dentro de "Fuel".
"Engine".Data.D0 "Engine".Data.D1	"Data" es una variable del tipo de datos LWORD en el bloque de datos global "Engine". D0 direcciona la dirección de palabra doble 0, D1 la dirección de palabra doble 1 dentro de "Data".

Superponer variables con AT

Descripción

Para acceder a determinadas áreas de datos dentro de una variable declarada, se puede superponer a las variables declaradas otra declaración más. De esta manera existe la posibilidad de direccionar una variable ya declarada con otro tipo de datos. Por ejemplo, cada bit de una variable del tipo de datos WORD se puede direccionar con un ARRAY of BOOL.

Reglas

Reglas generales válidas para la superposición de variables:

- En AWL, KOP, FUP y GRAPH es posible la superposición en S7-1200 y S7-1500.
- En SCL, la superposición es posible en todas las familias de CPU.

- La superposición de variables es posible en los siguientes bloques:
  - En bloques lógicos con acceso estándar
  - En bloques lógicos de acceso optimizado para variables con el ajuste de remanencia "Ajustar en IDB"
- El ancho de datos de la variable superpuesta debe ser menor o igual al de la variable sobre la que se superpone.
- Los tipos de datos VARIANT e INSTANCE no pueden superponerse.
- Los bloques de librerías que están declaradas como parámetros en la interfaz no pueden superponerse.
- Las variables PLC estructuradas que están declaradas como parámetros en la interfaz no pueden superponerse.

Nota

S7-1200/1500: utilizar AT en FC

En las FC del S7-1200/1500, el ancho de datos de las variables superpuestas deben ser idénticos a los de las variables sobre las que se superponen. Si esto no es posible en su programa, puede direccionar también áreas dentro de variables existentes mediante la sintaxis .X, .B, .W o .D.

Consulte también: Direccionamiento de áreas individuales de una variable (Página 2301)

Además, rigen las siguientes reglas de combinación:

		Variable superpuesta	Variable sobre la que se superpone			
Simple	Estructurada *	Any/Pointer	DB_ANY			
FB	Input	Simple	x	x		x
		Estructurada *	x	x	x	x
		Any/Pointer		x		
	Temp	Simple	x	x		
		Estructurada	x	x	x	
		Any/Pointer		x		
	Static, Output	Simple	x	x		x
		Estructurada	x	x		x
		Any/Pointer				
	InOut	Simple	x			x
		Estructurada		x		
		Any/Pointer				
FC	Temp	Simple	x	x		
		Estructurada	x	x	x	
		Any/Pointer		x		

		Variable superpuesta	Variable sobre la que se superpone			
	Input, Output, InOut	Simple (ambas variables deben tener el mismo ancho de bit)	x			x
		Estructurada		x	x	
		Any/Pointer				
OB	Temp	Simple	x	x		
		Estructurada	x	x	x	
		Any/Pointer		x		

* Los tipos de datos estructurados son los que se componen de varios elementos de datos, p. ej. ARRAY o STRUCT.

Declaración

Para superponer una variable, es necesario declarar una segunda variable directamente a continuación de la variable sobre la que se desea superponer, e identificarla con la palabra clave "AT".

Ejemplo

La figura siguiente muestra la declaración de una variable sobre la que se superpone en la interfaz de un FB:

Input		
MyByte		Byte
AT	AT "MyByte"	Array [0..7] of Bool
AT[0]		Bool
AT[1]		Bool
AT[2]		Bool
AT[3]		Bool
AT[4]		Bool
AT[5]		Bool
AT[6]		Bool
AT[7]		Bool

Al llamar un bloque con la declaración de variables indicada, se suministran valores a la variable "MyByte". Dentro del bloque existen dos posibilidades de interpretación de estos datos:

- como byte
- como ARRAY of BOOL unidimensional

Consulte también

Ajustar remanencia (Página 2951)

Direccionar operandos de modo indirecto

Fundamentos del direccionamiento indirecto

Introducción

El direccionamiento indirecto ofrece la posibilidad de direccionar operandos cuya dirección no se calcula hasta el momento de la ejecución. El direccionamiento indirecto también permite procesar varias veces partes del programa utilizando un operando diferente en cada ejecución.

> ⚠ ADVERTENCIA
>
> **Peligro de errores de acceso**
>
> Puesto que, en el direccionamiento indirecto, los operandos no se calculan hasta el momento de la ejecución, existe el peligro de que se produzcan errores de acceso y el programa funcione con valores incorrectos. Además, es posible sobrescribir áreas de memoria involuntariamente con valores incorrectos. En ese caso, el sistema de automatización puede reaccionar de forma inesperada.
>
> Por ello, el direccionamiento indirecto debe usarse siempre con precaución.

Direccionamiento indirecto

Fundamentos del direccionamiento indirecto

Posibilidades generales del direccionamiento indirecto en S7-1200 y S7-1500

Posibilidades de direccionamiento indirecto disponibles en todos los lenguajes de programación:

- Direccionamiento indirecto mediante punteros
- Indexación indirecta de componentes de ARRAY
- Direccionamiento indirecto de un bloque de datos mediante el tipo de datos DB_ANY.

Posibilidades de direccionamiento indirecto específicas del lenguaje

Además, los diferentes lenguajes de programación ofrecen las siguientes posibilidades de direccionamiento específicas:

- En AWL pueden direccionarse operandos de forma indirecta a través del registro de direcciones.
- En SCL se puede leer o escribir un área de memoria variable con las siguientes instrucciones:
  - POKE - escribir dirección de memoria
  - POKE_BOOL - escribir bit de memoria
  - PEEK - leer dirección de memoria
  - PEEK_BOOL - leer bit de memoria
  - POKE_BLK - escribir área de memoria

Encontrará una descripción detallada de estas posibilidades de direccionamiento en "Consulte también".

Consulte también

Direccionar variables en bloques de datos globales (Página 2295)

POKE: Escribir dirección de memoria (Página 4388)

POKE_BOOL: Escribir bit de memoria (Página 4390)

PEEK: Leer dirección de memoria (Página 4384)

PEEK_BOOL: Leer bit de memoria (Página 4386)

POKE_BLK: Escribir área de memoria (Página 4392)

Direccionamiento indirecto mediante punteros (Página 2306)

Indexación indirecta de componentes de ARRAY (Página 2307)

Direccionamiento indirecto en AWL (Página 2312)

Direccionamiento indirecto mediante punteros

Descripción

Para el direccionamiento indirecto se requiere un formato de datos especial, que contiene la dirección y, en su caso, también el rango y el tipo de datos de un operando. Este formato de datos se denomina puntero. Están disponibles los siguientes tipos de punteros:

- POINTER (S7-1500)
- ANY (S7-1500, válido solo para bloques con acceso estándar)
- VARIANT (S7-1200/1500)

Encontrará más información sobre tipos de datos de puntero en "Consulte también".

Nota

En SCL, el uso de POINTER está sujeto a limitaciones. Solo existe la posibilidad de reenviarlo a bloques ya llamados.

Ejemplo

El ejemplo siguiente muestra un direccionamiento indirecto con un puntero intraárea:

Direccionamiento en AWL	Significado
L P#10.0	// Cargar el puntero (P#10.0) en el Acumulador 1
T MD20	// Transferir el puntero al operando MD20
L MW [MD20]	// Cargar MW10 en el Acumulador 1
....	// Cualquier programa
L MD [MD20]	// Cargar MD10 en el Acumulador 1
....	// Cualquier programa
= M [MD20]	// Para RLO=1, activar el bit de marca M10.0

El puntero P#10.0 se transfiere al operando MD20. Si el operando MD20 está programado entre corchetes, en tiempo de ejecución se reemplazará por la dirección contenida en el puntero.

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Puntero (Página 2359)

Indexación indirecta de componentes de ARRAY

Descripción

Como índices para direccionar los componentes de un ARRAY no solo se pueden especificar constantes, sino también variables del tipo de datos Entero. Se admiten enteros con una longitud de hasta 32 bits. Si se usan variables, el índice se calcula en tiempo de ejecución. Así se puede utilizar un índice diferente para cada ejecución, p. ej. en bucles de programa.

Nota

Si se llama un bloque y se le transfiere un componente de ARRAY indexado de forma indirecta ("<Bloque de datos>".<ARRAY>["i"]) como parámetro de entrada/salida (InOut), no es posible modificar el valor de la variable de índice [i] durante el procesamiento del bloque. Es decir, el valor siempre se vuelve a escribir en el mismo componente de ARRAY del que se ha leído.

Sintaxis

Para indexar un ARRAY de forma indirecta se usa la siguiente sintaxis:

```
"<Bloque de datos>".<ARRAY>["i"] // ARRAY unidimensional
"<Bloque de datos>".<ARRAY>["i"].a // ARRAY of STRUCT unidimensional
"<Bloque de datos>".<ARRAY>["i", "j"] // ARRAY multidimensional
"<Bloque de datos>".<ARRAY>["i", "j"].a // ARRAY of STRUCT
multidimensional
```

La sintaxis está formada por las partes siguientes:

Parte	Descripción
Bloque de datos	Nombre del bloque de datos en el que se encuentra el ARRAY
ARRAY	Variable del tipo de datos ARRAY
i, j	Variables PLC del tipo de datos Entero que se emplean como punteros
a	Otras variables parciales de la estructura

Ejemplos

El siguiente ejemplo muestra la indexación indirecta de un componente ARRAY en AWL:

Deben moverse varios ejes a diferentes ángulos. Los valores correspondientes a número de eje y ángulo están depositados en el ARRAY bidimensional "control_axis".

Con la instrucción "SEL" se selecciona qué componente del ARRAY "control_axis" se escribe en el parámetro de salida "#out".

El número de eje se define mediante la constante "Constant_Axis_NoX" y "Constant_Axis_NoY", el ángulo se determina con la variable "#Angle".

Direccionamiento en AWL

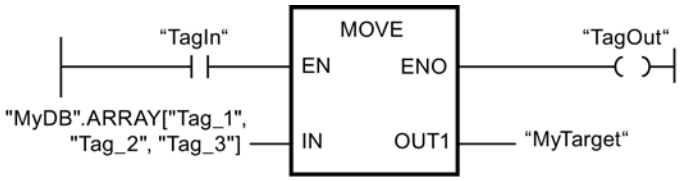
```
CALL  SEL
    value_type:=Int
    G   :="Select"
    IN0 :=#control_axis["Constant_Axis_NoX", #Angle]
    IN1 :=#control_axis["Constant_Axis_NoY", #Angle]
    OUT :=#out
```

Los ejemplos siguientes muestran la indexación indirecta de un componente de ARRAY (en este caso, con SCL). "MOTOR" es un ARRAY_of_INT unidimensional con tres filas. "VALUES" es una variable PLC del tipo de datos "Integer".

Direccionamiento en SCL	Significado
`MOTOR[2] := VALUES;`	`(*Direccionamiento directo: asignación de VALUES a la segunda fila del ARRAY MOTOR*)`
`MOTOR["Tag_1"] := VALUES;`	`(*Direccionamiento indirecto: Asignación de VALUES a la fila`

Direccionamiento en SCL	Significado
	especificada con "Tag_1" del ARRAY MOTOR*)
#MOTOR["Tag_2"+"Tag_3"] := #Values;	(*Direccionamiento indirecto: asignación de VALUES a la fila especificada con la expresión "Tag_2"+"Tag_3" del ARRAY MOTOR*)

El ejemplo siguiente muestra la indexación indirecta de un componente de ARRAY (en este caso, con KOP). "ARRAY" es un ARRAY tridimensional. "Tag_1", "Tag_2" y "Tag_3" son variables PLC del tipo de datos "Integer". En función de sus valores, uno de los componentes del "ARRAY" se copia en la variable "MyTarget".



Indexar componentes de ARRAY con las instrucciones "FieldRead" y "FieldWrite"

Para la indexación indirecta de componentes ARRAY en KOP y FUP también pueden utilizarse las siguientes instrucciones:

- FieldWrite - Escribir campo
- FieldRead - Leer campo
  Encontrará una descripción detallada de las instrucciones en el capítulo "Referencias".

Información adicional

Encontrará más información sobre el tipo de datos ARRAY en "Consulte también".

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Matriz (Página 2354)

Direccionar variables estructuradas (Página 2298)

Direccionamiento indirecto de caracteres individuales de un STRING

Descripción

Como índices para direccionar caracteres individuales de un STRING o WSTRING no solo se pueden especificar constantes, sino también variables. Las variables deben ser del tipo de

datos "Entero". Si se usan variables, el índice se calcula en tiempo de ejecución. Así se puede utilizar un índice diferente para cada ejecución, p. ej. en bucles de programa.

Si al llamar el bloque se transfiere un STRING o WSTRING indexado por variable a un parámetro de entrada/salida (InOut), tenga en cuenta lo siguiente: la variable de índice [i] se lee una vez al iniciar la llamada del bloque y, durante la ejecución del bloque llamado, no puede ser modificada por él.

Nota

Supervisión de accesos STRING en tiempo de ejecución

Si en tiempo de ejecución se escribe un STRING o WSTRING que supera la longitud definida, es posible que se produzcan reacciones inesperadas en el programa. En S7-1200/1500 se vigila el rebase de la longitud de STRING o WSTRING. Es posible elegir si se debe reaccionar a rebases con tratamiento de errores global de la CPU o con un tratamiento de errores local propio.

Sintaxis

Para indexar un STRING o WSTRING de forma indirecta se usa la siguiente sintaxis:

```
"<Bloque de datos>".<STRING>["i"]
"<Bloque de datos>".<WSTRING>["i"]
```

Ejemplo

El siguiente ejemplo muestra la indexación indirecta de un STRING a partir del ejemplo de SCL. "STRING" es una variable del tipo de datos STRING. "Tag_1" es una variable PLC del tipo de datos "Integer".

Direccionamiento en SCL	Significado
`STRING["Tag_1"] := CHARACTER;`	`(*Direccionamiento indirecto: Asignación de "CHARACTER" al carácter especificado con "Tag_1" del STRING*)`

Información adicional

Encontrará más información sobre el tipo de datos STRING en "Consulte también".

Consulte también

STRING (Página 2350)

Direccionamiento indirecto en AWL

Fundamentos de registros de direcciones

Introducción

Para direccionar operandos de forma indirecta existen dos registros de direcciones, el registro de direcciones 1 (AR1) y el registro de direcciones 2 (AR2). Los registros de direcciones son equivalentes y tienen una longitud de 32 bits. En los registros de direcciones pueden guardarse punteros intraárea y punteros interárea. Los datos guardados pueden llamarse en el programa para determinar la dirección de un operando.

El intercambio de datos entre los registros y las demás áreas de memoria disponibles se realiza con ayuda de instrucciones de carga y transferencia.

Nota

En el S7-1500 rigen reglas especiales para intercambiar datos a través del registro de bloques de datos y el registro de direcciones:

- los valores de los registros no se mantienen más allá de los límites del bloque.
- Al cambiar el lenguaje dentro de un bloque, los registros se inicializan.
- Solo es posible remitir a datos de bloques con acceso optimizado cuando estos disponen del ajuste de remanencia "Ajustar en IDB".
- No es posible remitir a datos locales en bloques con acceso optimizado mediante los registros de direcciones (interárea).

Información adicional

Encontrará más información sobre las instrucciones que utilizan registros de direcciones, así como sobre el direccionamiento indirecto, en el apartado "Consulte también".

Consulte también

LAR1: Cargar contenido del acumulador 1 en AR1 (Página 6254)

LAR1 <D>: Cargar palabra doble o puntero de área en AR1 (Página 6255)

LAR1 AR2: Cargar contenido del AR2 en AR1 (Página 6257)

LAR2: Cargar contenido del acumulador 1 en AR2 (Página 6258)

CAR: Intercambiar AR1 y AR2 (Página 6264)

TAR1: Transferir AR1 a acumulador 1 (Página 6265)

TAR1 <D>: Transferir AR1 a palabra doble (Página 6266)

TAR1 AR2: Transferir AR1 a AR2 (Página 6267)

TAR2: Transferir AR2 a acumulador 1 (Página 6268)

+AR1: Sumar acumulador 1 a AR1 (Página 6370)

+AR2: Sumar acumulador 1 a AR2 (Página 6371)

Direccionamiento indirecto en AWL (Página 2312)

Carga y transferencia (Página 4157)

Intercambio de datos en AWL (Página 2753)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Ajustar remanencia (Página 2951)

Direccionamiento indirecto en AWL

En AWL existen las siguientes posibilidades de direccionamiento indirecto:

- Direccionamiento indirecto por memoria
- Direccionamiento intraárea indirecto por registro
- Direccionamiento indirecto interárea por registro

Direccionamiento indirecto por memoria

En el direccionamiento indirecto por memoria se deposita la dirección en una variable. Esta variable puede ser del tipo de datos WORD o DWORD. La variable puede encontrarse en las áreas de memoria "Datos" (DB o DI), "Marcas" (M) o "Datos locales temporales" (L). En el S7-1500, los parámetros del FB también pueden utilizarse para depositar la dirección. Si la variable se encuentra en un bloque de datos, este debe tener acceso estándar.

Ejemplo de aplicaciones del direccionamiento indirecto por memoria:

Direccionamiento en AWL	Significado
`U E [MD 2]`	`// Ejecuta una operación lógica Y con un bit de entrada variable. La dirección del bit de entrada se encuentra en la palabra doble de marcas MD2.`
`= DIX [DBD 2]`	`// Asigna el RLO a un bit de datos variable. La dirección del bit de datos se encuentra en la palabra doble de datos DBD2.`
`L EB [DID 4]`	`// Carga un byte de entrada variable en ACU 1. La dirección del byte de entrada se encuentra en la palabra doble de instancia DID4.`
`AUF DB [LW 2]`	`// Abre un bloque de datos variable. El número del bloque de datos se encuentra en la palabra de datos locales LW2.`

Direccionamiento intraárea indirecto por registro

El direccionamiento indirecto por registro utiliza uno de los registros de direcciones (AR1 o AR2) para obtener la dirección del operando.

En el direccionamiento intraárea indirecto por registro solo se indexan la dirección de bit y la dirección de byte mediante el registro de direcciones (p. ej. `P#10.0`). El área de memoria para el que será válida la dirección del registro de direcciones no se indica hasta realizar que no

se programa la instrucción. De este modo, la dirección del registro de direcciones actuará en el área de memoria que se especifique en la instrucción.

Las áreas de memoria posibles son "Entradas" (I), "Salidas" (Q), "Periferia" (PI o PQ), "Marcas" (M), "Datos locales temporales" (L) y "Datos" (DB o DI). Si el operando se encuentra en un bloque de datos, este debe ser de acceso estándar.

Al introducir el direccionamiento intraárea indirecto por registro, después de indicar el registro de direcciones se especifica un offset que se añade al contenido del registro de direcciones sin modificarlo. Dicho offset también tiene el formato de un puntero. Es obligatorio indicar el puntero, lo cual debe realizarse como constante (p. ej. `P#0.0` o `P#2.0`).

Ejemplo de aplicación del direccionamiento indirecto intraárea por registro:

AWL	Significado
`LAR1 P#10.0`	`// Cargar el puntero (P#10.0) en el registro de direcciones 1`
`L IW [AR1, P#2.0]`	`// Aumentar el contenido del registro de direcciones 1 (P#10.0) con el offset P#2.0.`
	`// Cargar el contenido de la palabra de entrada IW12 en el acumulador 1`
`L IW [AR1, P#0.0]`	`// Aumentar el contenido del registro de direcciones 1 (P#10.0) con el offset P#0.0.`
	`// Cargar el contenido de la palabra de entrada IW10 en el acumulador 1`

Direccionamiento indirecto interárea por registro

En el direccionamiento interárea indirecto por registro se indexa la dirección completa del operando (es decir, tanto las direcciones de bit y byte como el área de memoria) mediante el registro de direcciones. Las áreas de memoria posibles son "Entradas" (I), "Salidas" (Q), "Periferia" (PI), "Marcas" (M), "Datos locales temporales" (L) y "Datos" (DB o DI). Si el operando se encuentra en un bloque de datos, este debe ser de acceso estándar o el operando debe tener el ajuste de remanencia "Ajustar en IDB".

En la instrucción únicamente se programa el ancho del operando. Los anchos del operando posibles son bit, byte, palabra y palabra doble.

Ejemplo de aplicación del direccionamiento interárea indirecto por registro:

`LAR1 P#M10.0`	`// Cargar el puntero interárea (P#M10.0) en el registro de direcciones 1`
`L W [AR1, P#2.0]`	`// Aumentar el contenido del registro de direcciones 1 (P#M10.0) con el offset P#2.0.`
	`// Cargar el contenido de la palabra de marca MW12 en el acumulador 1`
`LAR1 P#A10.0`	`// Cargar el puntero interárea (P#Q10.0) en el registro de direcciones 1`
`L W [AR1, P#2.0]`	`// Sumar el contenido del registro de direcciones 1 (P#Q10.0) con el offset P#2.0`

```
                    // Cargar el contenido de la palabra de salida QW12.0 en el
                    acumulador 1
```

Nota

Particularidades en S7-1500

En el S7-1500 rigen reglas especiales para intercambiar datos a través del registro de bloques de datos y el registro de direcciones:

- los valores de los registros no se mantienen más allá de los límites del bloque. Los registros también se inicializan al cambiar el lenguaje dentro de un bloque.
- Si se accede a un operando del tipo BYTE, WORD o DWORD mediante direccionamiento indirecto por registro, la dirección deberá comenzar en un límite de byte.
  Ejemplos:
  LAR1 P#0.0
  L MW [AR1, P#0.0] // P#0.0 + P#0.0 = P#0.0 - El direccionamiento está permitido ya que el puntero P#0.0 señala a un límite de byte.
  L MW [AR1, P#2.1] // P#0.0 + P#2.1 = P#2.1 - El direccionamiento no está permitido ya que el puntero P#2.1 no señala a un límite de byte.

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Direccionamiento indirecto de bloques de datos (Página 2315)

Direccionar variables estructuradas (Página 2298)

Fundamentos de registros de direcciones (Página 2311)

Intercambio de datos en AWL (Página 2753)

Direccionamiento indirecto

Direccionamiento indirecto mediante punteros

Descripción

Para el direccionamiento indirecto se requiere un formato de datos especial, que contiene la dirección y, en su caso, también el rango y el tipo de datos de un operando. Este formato de datos se denomina puntero. Tipos de punteros disponibles:

- POINTER (S7-300/400)
- ANY (S7-300/400)
- VARIANT (S7-1200/1500)

Encontrará más información sobre tipos de datos de puntero en "Consulte también".

Nota

En SCL, el uso del tipo de datos puntero está sujeto a limitaciones. Solo existe la posibilidad de reenviarlo a bloques ya llamados.

Ejemplo

El ejemplo siguiente muestra un direccionamiento indirecto con un puntero intraárea:

Direccionamiento en AWL	Significado
L P#10.0	// Cargar el puntero (P#10.0) en el Acumulador 1
T MD20	// Transferir el puntero al operando MD20
L MW [MD20]	// Cargar MW10 en el Acumulador 1
....	// Cualquier programa
L MD [MD20]	// Cargar MD10 en el Acumulador 1
....	// Cualquier programa
= M [MD20]	// Para RLO=1, activar el bit de marca M10.0

El puntero P#10.0 se transfiere al operando MD20. Si el operando MD20 está programado entre corchetes, en tiempo de ejecución se reemplazará por la dirección contenida en el puntero.

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Puntero (Página 2359)

Direccionamiento indirecto en SCL

Direccionamiento indirecto de bloques de datos

Direccionamiento indirecto de bloques de datos en SCL

Para direccionar bloques de datos de forma indirecta, puede utilizarse la función de conversión WORD_TO_BLOCK_DB. El número de DB se indica como variable o expresión con el tipo de datos WORD.

Sintaxis

Para direccionar un bloque de datos de forma indirecta se usa la siguiente sintaxis:
`WORD_TO_BLOCK_DB(índice).IdentificadorDelOperando(dirección)`

Ejemplos

Ejemplo 1: la variable global "Índice de dirección" del tipo de datos WORD se utiliza como número del DB.

Direccionamiento en SCL
`%M0.0:=WORD_TO_BLOCK_DB("índice de dirección").DX(0,0);`
`%MW0:=WORD_TO_BLOCK_DB("índice de dirección").DW(4);`

Ejemplo 2: la variable global "Índice de dirección" del tipo de datos WORD se utiliza como número del DB. El elemento de datos dentro del DB también se especifica mediante un índice.

Direccionamiento en SCL
`%M0.0:=WORD_TO_BLOCK_DB("índice de dirección").DX(#i,#y);`
`%MW0:=WORD_TO_BLOCK_DB("Índice de dirección").DW(#y);`

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Direccionamiento indirecto en AWL (Página 2312)

Direccionar variables estructuradas (Página 2298)

Direccionamiento indirecto de variables

Direccionamiento indirecto de variables en SCL

El direccionamiento indirecto se realiza de modo similar al direccionamiento absoluto. En lugar de la dirección se especifica un offset entre paréntesis. El offset se compone de una variable de byte, y si se usan operandos booleanos, de una variable de byte y una variable de bit. Las variables de byte y bit deben ser del tipo de datos INT.

Los temporizadores y contadores procedentes de la tabla de variables PLC no se pueden direccionar de forma indirecta de este modo.

Sintaxis

Para indexar una variable global de forma indirecta se usa la siguiente sintaxis:
`IdentificadorDelOperando(VariableDeByte)`
`IdentificadorDelOperando(VariableDeByte.VariableDeBit)`

Para indexar una variable DB de forma indirecta se usa la siguiente sintaxis:
`MyDB.IdentificadorDelOperando(VariableDeByte)`
`MyDB.IdentificadorDelOperando(VariableDeByte.VariableDeBit)`

Ejemplos

Ejemplo 1: En función del valor de la variable contador i se pone a 0 una de las palabras de entrada 2 a 8.

Direccionamiento en SCL

```
#i:=2;
FOR #i := 2 TO 8 DO
    %IW(#i) := 0 ;
END_FOR;
```

Ejemplo 2: En función del valor de la variable contador i se pone a 0 una de las palabras 2 a 8 en DB10.

Direccionamiento en SCL

```
#i:=2;
FOR #i := 2 TO 8 DO
    %DB10.DW(#i) := 0 ;
END_FOR;
```

Direccionamiento indirecto de componentes de ARRAY

Descripción

Como índices para direccionar los componentes de un ARRAY no solo se pueden especificar constantes, sino también variables del tipo de datos Entero. Si se usan variables, el índice se calcula en tiempo de ejecución. Así se puede utilizar un índice diferente para cada ejecución, p. ej. en bucles de programa.

Nota

La variable de índice [i] se lee una vez al iniciar la llamada al bloque y, durante la ejecución del bloque llamado, no puede ser modificada por él.

Si se llama un bloque y se le transfiere un componente de ARRAY indexado de forma indirecta ("<Bloque de datos>".<ARRAY>["i"]) como parámetro de entrada/salida (InOut), no es posible modificar el valor de la variable de índice durante el procesamiento del bloque. Es decir, el valor siempre se vuelve a escribir en el mismo componente de ARRAY del que se ha leído.

Sintaxis

Para indexar un ARRAY de forma indirecta se usa la siguiente sintaxis:

```
"<Bloque de datos>".<ARRAY>["i"] // ARRAY unidimensional
"<Bloque de datos>".<ARRAY>["i"].a // ARRAY of STRUCT unidimensional
"<Bloque de datos>".<ARRAY>["i", "j"] // ARRAY multidimensional
"<Bloque de datos>".<ARRAY>["i", "j"].a // ARRAY of STRUCT
multidimensional
```

La sintaxis está formada por las partes siguientes:

Parte	Descripción
Bloque de datos	Nombre del bloque de datos en el que se encuentra el ARRAY
ARRAY	Variable del tipo de datos ARRAY
i, j	Variables PLC del tipo de datos Entero que se emplean como punteros
a	Otras variables parciales de la estructura

Ejemplos

Los ejemplos siguientes muestran la indexación indirecta de un componente de ARRAY (en este caso, con SCL). MOTOR es un ARRAY_of_INT unidimensional con tres filas. VALUES es una variable PLC del tipo de datos "Integer".

Direccionamiento en SCL	Significado
`MOTOR[2] := VALUES;`	`(*Direccionamiento directo: asignación de VALUES a la segunda fila del ARRAY MOTOR*)`
`MOTOR["Tag_1"] := VALUES;`	`(*Direccionamiento indirecto: asignación de VALUES a la fila especificada con "Tag_1" del ARRAY MOTOR*)`
`#MOTOR["Tag_2"+"Tag_3"] := #Values;`	`(*Direccionamiento indirecto: asignación de VALUES a la fila especificada con la expresión "Tag_2"+"Tag_3" del ARRAY MOTOR*)`

Información adicional

Encontrará más información sobre el tipo de datos ARRAY en "Consulte también".

Direccionamiento indirecto en AWL

Fundamentos de registros de direcciones

Introducción

Para direccionar operandos de forma indirecta existen dos registros de direcciones, el registro de direcciones 1 (AR1) y el registro de direcciones 2 (AR2). Los registros de direcciones son equivalentes y tienen una longitud de 32 bits. En los registros de direcciones pueden guardarse punteros intraárea y punteros interárea. Los datos guardados pueden llamarse en el programa para determinar la dirección de un operando.

El intercambio de datos entre los registros y las demás áreas de memoria disponibles se realiza con ayuda de instrucciones de carga y transferencia.

Encontrará más información sobre las instrucciones que utilizan registros de direcciones, así como sobre el direccionamiento indirecto, en el apartado "Consulte también".

Consulte también

LAR1: Cargar contenido del acumulador 1 en AR1 (Página 6254)

LAR1 <D>: Cargar palabra doble o puntero de área en AR1 (Página 6255)

LAR1 AR2: Cargar contenido del AR2 en AR1 (Página 6257)

LAR2: Cargar contenido del acumulador 1 en AR2 (Página 6258)

CAR: Intercambiar AR1 y AR2 (Página 6264)

TAR1: Transferir AR1 a acumulador 1 (Página 6265)

TAR1 <D>: Transferir AR1 a palabra doble (Página 6266)

TAR1 AR2: Transferir AR1 a AR2 (Página 6267)

TAR2: Transferir AR2 a acumulador 1 (Página 6268)

+AR1: Sumar acumulador 1 a AR1 (Página 6370)

+AR2: Sumar acumulador 1 a AR2 (Página 6371)

Direccionamiento indirecto en AWL (Página 2319)

Carga y transferencia (Página 4157)

Intercambio de datos en AWL (Página 2753)

Direccionamiento indirecto en AWL

En AWL existen las siguientes posibilidades de direccionamiento indirecto:

- Direccionamiento indirecto por memoria
- Direccionamiento intraárea indirecto por registro
- Direccionamiento indirecto interárea por registro

Direccionamiento indirecto por memoria

En el direccionamiento indirecto por memoria se deposita la dirección en una variable. Esta variable puede ser del tipo de datos WORD o DWORD. La variable puede encontrarse en las áreas de memoria "Datos" (DB o DI), "Marcas" (M) o "Datos locales temporales" (L).

Ejemplo de aplicaciones del direccionamiento indirecto por memoria:

Direccionamiento en AWL	Significado
U E [MD 2]	// Ejecuta una operación lógica Y con un bit de entrada variable. La dirección del bit de entrada se encuentra en la palabra doble de marcas MD2.
= DIX [DBD 2]	// Asigna el RLO a un bit de datos variable. La dirección del bit de datos se encuentra en la palabra doble de datos DBD2.

Direccionamiento en AWL	Significado
L EB [DID 4]	// Carga un byte de entrada variable en ACU 1. La dirección del byte de entrada se encuentra en la palabra doble de instancia DID4.
AUF DB [LW 2]	// Abre un bloque de datos variable. El número del bloque de datos se encuentra en la palabra de datos locales LW2.

Direccionamiento intraárea indirecto por registro

El direccionamiento indirecto por registro utiliza uno de los registros de direcciones (AR1 o AR2) para obtener la dirección del operando.

En el direccionamiento intraárea indirecto por registro solo se indexan la dirección de bit y la dirección de byte mediante el registro de direcciones (p. ej. `P#10.0`). El área de memoria para el que será válida la dirección del registro de direcciones no se indica hasta realizar que no se programa la instrucción. De este modo, la dirección del registro de direcciones actuará en el área de memoria que se especifique en la instrucción.

Las áreas de memoria posibles son "Entradas" (I), "Salidas" (Q), "Periferia" (PI o PQ), "Marcas" (M), "Datos locales temporales" (L) y "Datos" (DB o DI).

Al introducir el direccionamiento intraárea indirecto por registro, después de indicar el registro de direcciones se especifica un offset que se añade al contenido del registro de direcciones sin modificarlo. Dicho offset también tiene el formato de un puntero. Es obligatorio indicar el puntero, lo cual debe realizarse como constante (p. ej. `P#0.0` o `P#2.0`).

Ejemplo de aplicación del direccionamiento indirecto intraárea por registro:

AWL	Significado
LAR1 P#10.0	// Cargar el puntero (P#10.0) en el registro de direcciones 1
L IW [AR1, P#2.0]	// Aumentar el contenido del registro de direcciones 1 (P#10.0) con el offset P#2.0.
	// Cargar el contenido de la palabra de entrada IW12 en el acumulador 1
L IW [AR1, P#0.0]	// Aumentar el contenido del registro de direcciones 1 (P#10.0) con el offset P#0.0.
	// Cargar el contenido de la palabra de entrada IW10 en el acumulador 1

Direccionamiento indirecto interárea por registro

En el direccionamiento interárea indirecto por registro se indexa la dirección completa del operando (es decir, tanto las direcciones de bit y byte como el área de memoria) mediante el registro de direcciones. Las áreas de memoria posibles son "Entradas" (I), "Salidas" (Q), "Periferia" (PI), "Marcas" (M), "Datos locales temporales" (L) y "Datos" (DB o DI). En la instrucción únicamente se programa el ancho del operando. Los anchos del operando posibles son bit, byte, palabra y palabra doble.

Ejemplo de aplicación del direccionamiento interárea indirecto por registro:

```
LAR1 P#M10.0          // Cargar el puntero interárea (P#M10.0) en el registro de
                      direcciones 1
L W [AR1, P#2.0]      // Aumentar el contenido del registro de direcciones 1
                      (P#M10.0) con el offset P#2.0.
                      // Cargar el contenido de la palabra de marca MW12 en el
                      acumulador 1
LAR1 P#A10.0          // Cargar el puntero interárea (P#Q10.0) en el registro de
                      direcciones 1
L W [AR1, P#2.0]      // Sumar el contenido del registro de direcciones 1 (P#Q10.0)
                      con el offset P#2.0
                      // Cargar el contenido de la palabra de salida QW12.0 en el
                      acumulador 1
```

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Direccionamiento indirecto de bloques de datos (Página 2315)

Direccionar variables estructuradas (Página 2298)

Fundamentos de registros de direcciones (Página 2318)

Intercambio de datos en AWL (Página 2753)

9.1.1.5 Tipos de datos

Vista general de los tipos de datos válidos

Validez de los grupos de tipos de datos

Los tipos de datos determinan las propiedades de los datos, p. ej. el formato de presentación del contenido y las áreas de memoria admisibles. En el programa de usuario se pueden utilizar tipos de datos predefinidos o definir tipos propios.

Las tablas siguientes muestran la disponibilidad de los tipos de datos predefinidos en las distintas CPUs S7:

Tabla 9-1 Números binarios

Números binarios	S7-300/400	S7-1200	S7-1500
BOOL (Página 2325)	X	X	X
Secuencias de bits			
BYTE (Página 2326)	X	X	X
WORD (Página 2327)	X	X	X
DWORD (Página 2327)	X	X	X
LWORD (Página 2328)	-	-	X

Tabla 9-2 Enteros

Enteros	S7-300/400	S7-1200	S7-1500
SINT (Página 2330)	-	X	X
INT (Página 2331)	X	X	X
DINT (Página 2333)	X	X	X
USINT (Página 2331)	-	X	X
UINT (Página 2332)	-	X	X
UDINT (Página 2334)	-	X	X
LINT (Página 2335)	-	-	X
ULINT (Página 2336)	-	-	X

Tabla 9-3 Números en coma flotante

Números en coma flotante	S7-300/400	S7-1200	S7-1500
REAL (Página 2338)	X	X	X
LREAL (Página 2339)	-	X	X

Tabla 9-4 Temporizadores

Temporizadores	S7-300/400	S7-1200	S7-1500
S5TIME (Página 2341)	X	-	X
TIME (Página 2343)	X	X	X
LTIME (Página 2343)	-	-	X

Tabla 9-5 Fecha y hora

Fecha y hora	S7-300/400	S7-1200	S7-1500
DATE (Página 2344)	X	X	X
TOD (TIME_OF_DAY) (Página 2345)	X	X	X
LTOD (LTIME_OF_DAY) (Página 2345)	-	-	X
DT (DATE_AND_TIME) (Página 2346)	X	-	X
LDT (Página 2347)	-	-	X
DTL (Página 2348)	-	X	X

Tabla 9-6 Carácter

Carácter	S7-300/400	S7-1200	S7-1500
CHAR (Página 2349)	X	X	X
WCHAR (Página 2349)	-	-	X
STRING (Página 2350)	X	X	X
WSTRING (Página 2351)	-	-	X

Tabla 9-7 Matriz

Matriz	S7-300/400	S7-1200	S7-1500
ARRAY [….] OF <tipo> (Página 2354)	X	X	X

Tabla 9-8 Estructuras

Estructuras	S7-300/400	S7-1200	S7-1500
STRUCT (Página 2358)	X	X	X

Tabla 9-9 Puntero

Puntero	S7-300/400	S7-1200	S7-1500
POINTER (Página 2359)	X	-	X
ANY (Página 2362)	X	-	X
VARIANT (Página 2364)	-	X	X

Tabla 9-10 Tipos de parámetros

Tipos de parámetros	S7-300/400	S7-1200	S7-1500
TIMER (Página 2366)	X	-	X
COUNTER (Página 2366)	X	-	X
BLOCK_FC (Página 2366)	X	-	X
BLOCK_FB (Página 2366)	X	-	X
BLOCK_DB (Página 2366)	X	-	-
BLOCK_SDB (Página 2366)	X	-	-
BLOCK_SFB (Página 2366)	X	-	-
BLOCK_SFC (Página 2366)	X	-	-
BLOCK_OB (Página 2366)	X	X	X
VOID (Página 2366)	X	X	X

Tabla 9-11 Tipos de datos PLC

Tipos de datos PLC	S7-300/400	S7-1200	S7-1500
Tipo de datos PLC (Página 2367)	X	X	X

Tabla 9-12 Tipos de datos de sistema

Tipos de datos de sistema	S7-300/400	S7-1200	S7-1500
IEC_TIMER (Página 2369)	X[1)]	X	X
IEC_LTIMER (Página 2369)	-	-	X
IEC_SCOUNTER (Página 2369)	-	X	X
IEC_USCOUNTER (Página 2369)	-	X	X
IEC_COUNTER (Página 2369)	X[2)]	X	X
IEC_UCOUNTER (Página 2369)	-	X	X
IEC_DCOUNTER (Página 2369)	-	X	X
IEC_UDCOUNTER (Página 2369)	-	X	X
IEC_LCOUNTER (Página 2369)	-	-	X
IEC_ULCOUNTER (Página 2369)	-	-	X
ERROR_STRUCT (Página 2369)	-	X	X
NREF (Página 2369)	-	X	X

Tipos de datos de sistema	S7-300/400	S7-1200	S7-1500
CREF (Página 2369)	-	X	X
FBTREF (Página 2369)	-	-	-
VREF (Página 2369)	-	-	-
STARTINFO (Página 2369)	X	-	X
SSL_HEADER (Página 2369)	X	-	X
CONDITIONS (Página 2369)	-	X	X
TADDR_Param (Página 2369)	-	X	X
TCON_Param (Página 2369)	-	X	X
1) En las CPUs S7-300/400, el tipo de datos se representa por medio de TP, TON y TOF. 2) En las CPUs S7-300/400, el tipo de datos se representa por medio de CTU, CTD y CTUD.			

Tabla 9-13 Tipos de datos de hardware

Tipos de datos de hardware	S7-300/400	S7-1200	S7-1500
REMOTE (Página 2371)	-	X	X
GEOADDR (Página 2371)	-	-	X
HW_ANY (Página 2371)	-	X	X
HW_DEVICE (Página 2371)	-	X	X
HW_DPMASTER (Página 2371)	-	-	X
HW_DPSLAVE (Página 2371)	-	X	X
HW_IO (Página 2371)	-	X	X
HW_IOSYSTEM (Página 2371)	-	X	X
HW_SUBMODULE (Página 2371)	-	X	X
HW_MODULE (Página 2371)	-	-	X
HW_INTERFACE (Página 2371)	-	X	X
HW_IEPORT (Página 2371)	-	X	X
HW_HSC (Página 2371)	-	X	X
HW_PWM (Página 2371)	-	X	X
HW_PTO (Página 2371)	-	X	X
AOM_AID (Página 2371)	-	X	X
AOM_IDENT (Página 2371)	-	X	X
EVENT_ANY (Página 2371)	-	X	X
EVENT_ATT (Página 2371)	-	X	X
EVENT_HWINT (Página 2371)	-	X	X
OB_ANY (Página 2371)	-	X	X
OB_DELAY (Página 2371)	-	X	X
OB_TOD (Página 2371)	-	X	X
OB_CYCLIC (Página 2371)	-	X	X
OB_ATT (Página 2371)	-	X	X
OB_PCYCLE (Página 2371)	-	X	X
OB_HWINT (Página 2371)	-	X	X
OB_DIAG (Página 2371)	-	X	X
OB_TIMEERROR (Página 2371)	-	X	X

Tipos de datos de hardware	S7-300/400	S7-1200	S7-1500
OB_STARTUP (Página 2371)	-	X	X
PORT (Página 2371)	-	X	X
RTM (Página 2371)	-	X	X
PIP (Página 2371)	-	-	X
CONN_ANY (Página 2371)	-	X	X
CONN_PRG (Página 2371)	-	X	X
CONN_OUC (Página 2371)	-	X	X
CONN_R_ID (Página 2371)	-	-	X
DB_ANY (Página 2371)	-	X	X
DB_WWW (Página 2371)	-	X	X
DB_DYN (Página 2371)	-	X	X

Nota

En función de la versión de la CPU, los tipos de datos efectivamente válidos pueden diferir ligeramente respecto a los de la tabla.

Números binarios

BOOL (bit)

Descripción

Un operando del tipo de datos BOOL representa un valor de bit y contiene uno de los siguientes valores:

- TRUE
- FALSE

La tabla siguiente muestra las propiedades del tipo de datos BOOL:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
1	Booleano	FALSE o TRUE BOOL#0 o BOOL#1 BOOL#FALSE o BOOL#TRUE	TRUE BOOL#1 BOOL#TRUE
	Enteros sin signo	0 ó 1	1
	Números binarios	2#0 ó 2#1	2#0
	Números octales	8#0 u 8#1	8#1
	Números hexadecimales	16#0 ó 16#1	16#1

Nota

Válido para las CPU de la serie S7-1500

Con un bloque con la propiedad "Acceso optimizado al bloque", el bit tiene 1 byte de longitud.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Secuencias de bits

BYTE (bytes)

Descripción

Un operando del tipo de datos BYTE es una secuencia de 8 bits.

La tabla siguiente muestra las propiedades del tipo de datos BYTE:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
8	Enteros sin signo[1)]	-128 a +127 o 0 a +255	15, BYTE#15, B#15
	Números binarios	de 2#0 a 2#11111111	2#00001111, BYTE#2#00001111, B#2#00001111
	Números octales	de 8#0 a 8#377	8#17, BYTE#8#17, B#8#17
	Números hexadecimales	de B#16#0 a B#16#FF, de 16#0 a 16#FF	16#0F, BYTE#16#0F, B#16#0F
[1)] El rango de valores depende de la interpretación o conversión correspondiente.			

Nota

El tipo de datos BYTE no puede someterse a una comprobación de mayor o menor. En la parte decimal solo puede alimentarse con los mismos datos que pueden procesar los tipos de datos SINT y USINT.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

WORD

Descripción

Un operando del tipo de datos WORD es una secuencia de 16 bits.

La tabla siguiente muestra las propiedades del tipo de datos WORD:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
16	Enteros sin signo	-32768 hasta 65535	61680, WORD#61680, W#61680
	Números binarios	2#0 a 2#1111111111111111	2#1111000011110000, WORD#2#1111000011110000, W#2#1111000011110000
	Números octales	8#0 a 8#177777	8#170360, WORD#8#170360, W#8#170360
	Números hexadecimales	de W#16#0 a W#16#FFFF, de 16#0 a 16#FFFF	16#F0F0, WORD#16#F0F0, W#16#F0F0
	BCD	de C#0 a C#999	C#55
	Secuencia decimal	B#(0, 0) a B#(255, 255)	B#(127, 200)

Nota

El tipo de datos WORD no puede someterse a una comprobación de mayor o menor. En la parte decimal solo puede alimentarse con los mismos datos que pueden procesar los tipos de datos INT y UINT.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

DWORD

Descripción

Un operando del tipo de datos DWORD es una secuencia de 32 bits.

La tabla siguiente muestra las propiedades del tipo de datos DWORD:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
32	Enteros sin signo	-2147483648 hasta 4294967295	15793935, DWORD#15793935, DW#15793935
	Números binarios	2#0 a 2#11111111111111111111111111111111	2#00000000111100001111111100001111, DWORD#2#00000000111100001111111100001111, DW#2#00000000111100001111111100001111
	Números octales	8#0 a 8#37777777777	8#74177417, DWORD#8#74177417, DW#8#74177417
	Números hexadecimales	de DW#16#00000000 a DW#16#FFFFFFFF, de 16#00000000 a 16#FFFFFFFF	16#00F0FF0F, DWORD#16#00F0FF0F, DW#16#00F0FF0F
	Secuencia decimal	B#(0, 0, 0, 0) a B#(255, 255, 255, 255)	B#(127, 200, 127, 200)

Nota

El tipo de datos DWORD no puede someterse a una comprobación de mayor o menor. En la parte decimal solo puede alimentarse con los mismos datos que pueden procesar los tipos de datos DINT y UDINT.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de las constantes (Página 2288)

Conversión de tipos de datos (Página 2373)

LWORD

Descripción

Un operando del tipo de datos LWORD es una secuencia de 64 bits.

La tabla siguiente muestra las propiedades del tipo de datos LWORD:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
64	Enteros sin signo	-9223372036854775808 hasta 18446744073709551615	26123590360715, LWORD#26123590360715, LW#26123590360715
	Números binarios	2#0 a 2#1111111111111111111111111111111111111111111111111111111111111111	2#0000000000000000000101111100001001011110101001011011 11010001011, LWORD#2#00000000000000000000101111100001001011110101001011011 01010001011, LW#2#0000000000000000000101111100001001011110101001011011 01010001011
	Números octales	8#0 a 8#1777777777777777777777	8#13724557213, LWORD#8#13724557213, LW#8#13724557213
	Números hexadecimales	LW#16#00000000 hasta LW#16#FFFFFFFFFFFFFFFF, 16#00000000 hasta 16#FFFFFFFFFFFFFFFF	16#000000005F52DE8B, LWORD#16#000000005F52DE8B, LW#16#000000005F52DE8B
	Secuencia decimal	B#(0, 0, 0, 0, 0, 0, 0, 0) a B#(255, 255, 255, 255, 255, 255, 255, 255)	B#(127, 200, 127, 200, 127, 200, 127, 200)

Nota

El tipo de datos LWORD no puede someterse a una comprobación de mayor o menor. En la parte decimal solo puede alimentarse con los mismos datos que pueden procesar los tipos de datos LINT y ULINT.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Sinopsis de la conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Conversión de tipos de datos (Página 2373)

Enteros

SINT (enteros de 8 bits)

Descripción

Un operando del tipo de datos SINT (short INT) tiene una longitud de 8 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 6 representan el valor del número. El estado lógico del bit 7 representa el signo. El signo puede adoptar el estado lógico "0" para positivo o "1" para negativo.

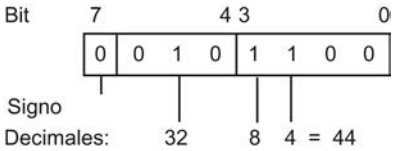
Un operando del tipo de datos SINT ocupa un BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos SINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
8	Enteros con signo	-128 hasta 127	+44, SINT#+44
	Números binarios	2#0 hasta 2#01111111	2#00101100, SINT#2#00101100
	Números octales (sólo positivos)	8#0 hasta 8#177	8#54, SINT#8#54
	Números hexadecimales (sólo positivos)	16#0 hasta 16#7F	16#2C, SINT#16#2C

Ejemplo

La figura siguiente muestra el entero +44 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

USINT (enteros de 8 bits)

Descripción

Un operando del tipo de datos USINT (Unsigned Short INT) tiene una longitud de 8 bits y contiene valores numéricos sin signo.

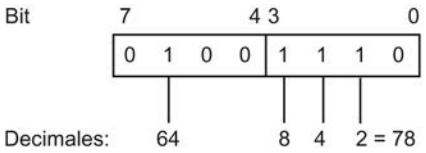
Un operando del tipo de datos USINT ocupa un BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos USINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
8	Enteros sin signo	de 0 a 255	78, USINT#78
	Números binarios	de 2#0 a 2#11111111	2#01001110, USINT#2#01001110
	Números octales	de 8#0 a 8#377	8#116, USINT#8#116
	Números hexadecimales	16#0 hasta 16#FF	16#4E, USINT#16#4E

Ejemplo

La figura siguiente muestra el entero 78 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

INT (enteros de 16 bits)

Descripción

Un operando del tipo de datos INT tiene una longitud de 16 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 14 representan el valor del número. El estado lógico del bit 15 representa el signo. El signo puede adoptar el estado lógico "0" para positivo o "1" para negativo.

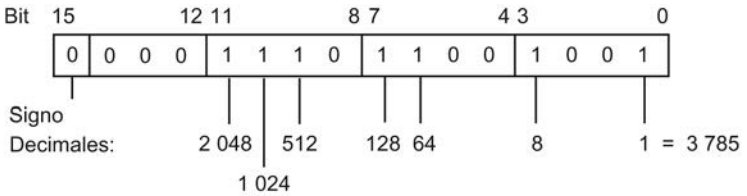
Un operando del tipo de datos INT ocupa dos BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos INT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
16	Enteros con signo	-32768 hasta 32767	+3785, INT#+3785
	Números binarios (sólo positivos)	2#0 hasta 2#0111111111111111	2#0000111011001001, INT#2#0000111011001001
	Números octales	8#0 hasta 8#77777	8#7311, INT#8#7311
	Números hexadecimales (sólo positivos)	16#0 hasta 16#7FFF	16#0EC9, INT#16#0EC9

Ejemplo

La figura siguiente muestra el entero +3785 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

UINT (enteros de 16 bits)

Descripción

Un operando del tipo de datos UINT (Unsigned INT) tiene una longitud de 16 bits y contiene valores numéricos sin signo.

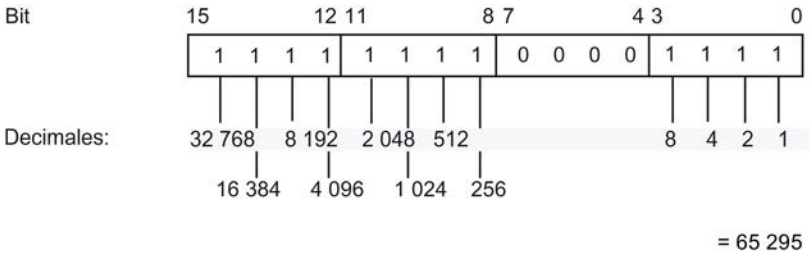
Un operando del tipo de datos UINT ocupa dos BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos UINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
16	Enteros sin signo	de 0 a 65535	65295, UINT#65295
	Números binarios	de 2#0 a 2#1111111111111111	2#1111111100001111, UINT#2#1111111100001111
	Números octales	de 8#0 a 8#177777	8#177417, UINT#8#177417
	Números hexadecimales	16#0 hasta 16#FFFF	16#FF0F, UINT#16#FF0F

Ejemplo

La figura siguiente muestra el entero 65295 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

DINT (enteros de 32 bits)

Descripción

Un operando del tipo de datos DINT (Double INT) tiene una longitud de 32 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 30 representan el valor del número. El estado lógico del bit 31 representa el signo. El signo puede adoptar el estado lógico "0" para positivo o "1" para negativo.

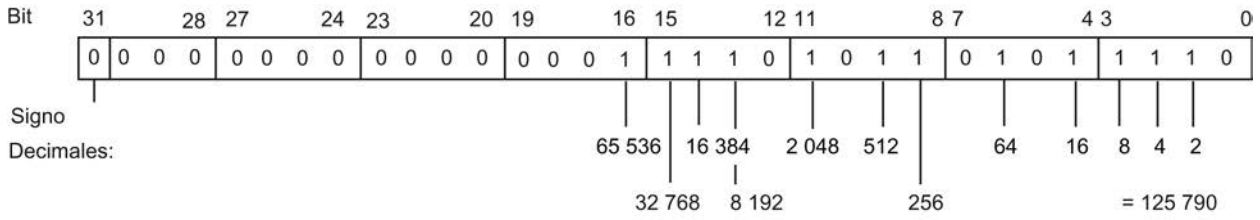
Un operando del tipo de datos DINT ocupa cuatro BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos DINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
32	Enteros con signo	-2147483648 hasta +2147483647	125790, DINT#125790, L#275
	Números binarios (sólo positivos)	2#0 hasta 2#01111111111111111111111111111111	2#00000000000000001111010 1101011110, DINT#2#00000000000000011 110101101011110
	Números octales (sólo positivos)	8#0 hasta 8#17777777777	8#365536, DINT#8#365536
	Números hexadecimales	16#00000000 hasta 16#7FFFFFFF	16#0001EB5E, DINT#16#0001EB5E

Ejemplo

La figura siguiente muestra el entero +125790 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

UDINT (enteros de 32 bits)

Descripción

Un operando del tipo de datos UDINT (Unsigned Double INT) tiene una longitud de 32 bits y contiene valores numéricos sin signo.

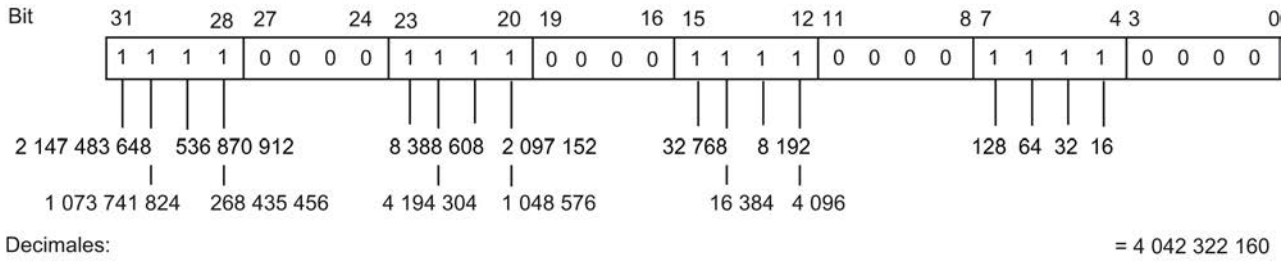
Un operando del tipo de datos UDINT ocupa cuatro BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos UDINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
32	Enteros sin signo	de 0 a 4294967295	4042322160, UDINT#4042322160
	Números binarios	de 2#0 a 2#11111111111111111111111111111111	2#11110000111100001111000011110000, UDINT#2#11110000111100001111000011110000
	Números octales	8#0 hasta 8#37777777777	8#36074170360, UDINT#8#36074170360
	Números hexadecimales	de 16#00000000 a 16#FFFFFFFF	16#F0F0F0F0, UDINT#16#F0F0F0F0

Ejemplo

La figura siguiente muestra el entero 4042322160 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

LINT (enteros de 64 bits)

Descripción

Un operando del tipo de datos LINT (Long INT) tiene una longitud de 64 bits y consta de dos componentes: Un signo y un valor numérico en complemento a dos. Los estados lógicos de los bits 0 a 62 representan el valor del número. El estado lógico del bit 63 representa el signo. El signo puede adoptar el estado lógico "0" para positivo o "1" para negativo.

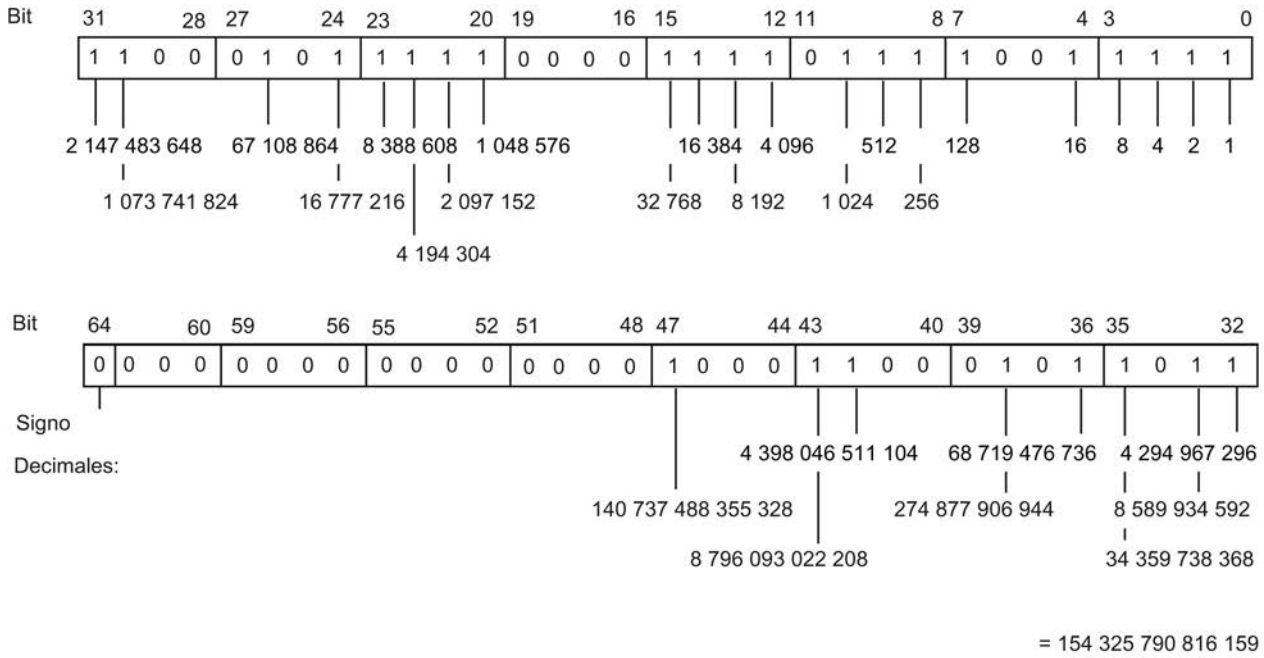
Un operando del tipo de datos LINT ocupa ocho BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos LINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
64	Enteros con signo	-9223372036854775808 hasta +9223372036854775807	+154325790816159, LINT# +154325790816159
	Números binarios (sólo positivos)	de 2#0 a 2#0111111111111111111111111111111111111111111111111111111111111111	2#0000000000000000100011000101101111000101111100001111011110011111, LINT#2#0000000000000000100011000101101111000101111100001111011110111110011111
	Números octales	8#0 hasta 8#7777777777777777777	8#4305570574173637, LINT#8#4305570574173637
	Números hexadecimales (sólo positivos)	de 16#0 a 16#7FFFFFFFFFFFFFFF	16#00008C5BC5F0F79F, LINT#16#00008C5BC5F0F79F

Ejemplo

La figura siguiente muestra el entero +154325790816159 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

ULINT (enteros de 64 bits)

Descripción

Un operando del tipo de datos ULINT (Unsigned Long INT) tiene una longitud de 64 bits y contiene valores numéricos sin signo.

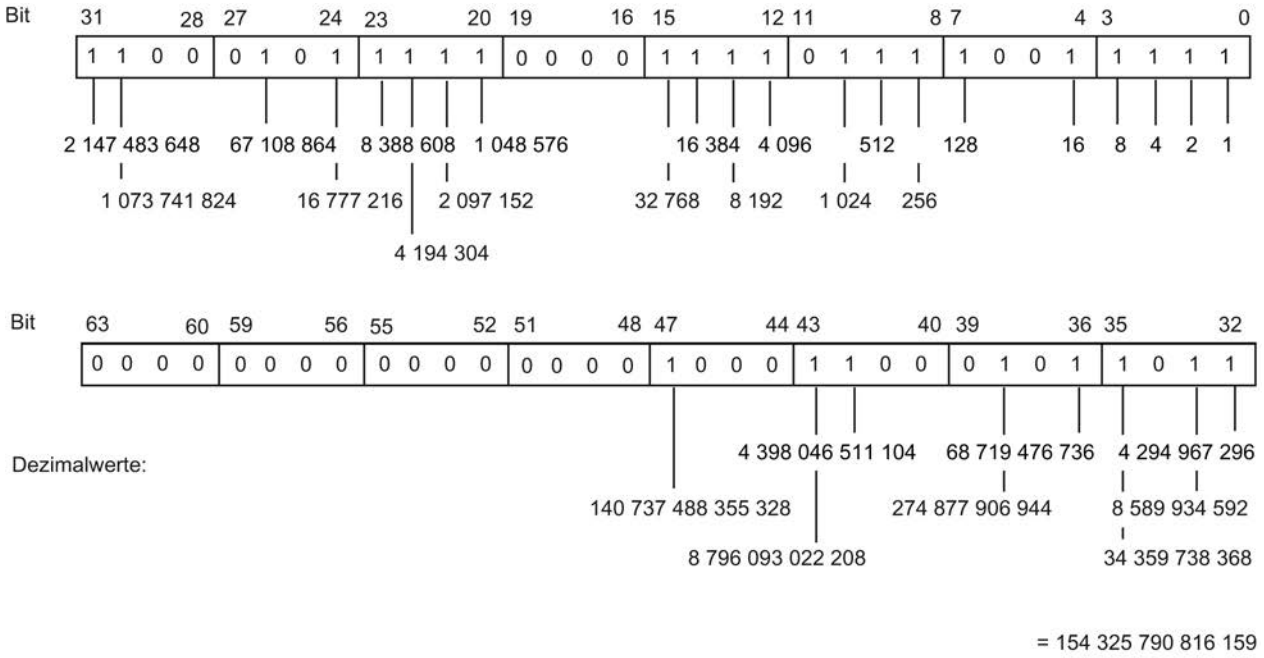
Un operando del tipo de datos ULINT ocupa ocho BYTE en la memoria.

La tabla siguiente muestra las propiedades del tipo de datos ULINT:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
64	Enteros sin signo	de 0 a 18446744073709551615	154325790816159, ULINT#154325790816159
	Números binarios	2#0 hasta 2#1111111111111111111111111111111111111111111111111111111111111111	2#0000000000000000100011000101101111000101111100001111011110011111, ULINT#2#0000000000000000100011000101101111000101111100001111011110111111
	Números octales	8#0 hasta 8#1777777777777777777777	8#4305570574173637, ULINT#8#4305570574173637
	Números hexadecimales	16#0 hasta 16#FFFFFFFFFFFFFFFF	16#00008C5BC5F0F79F, ULINT#16#00008C5BC5F0F79F

Ejemplo

La figura siguiente muestra el entero 154325790816159 como número binario:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Números en coma flotante

REAL

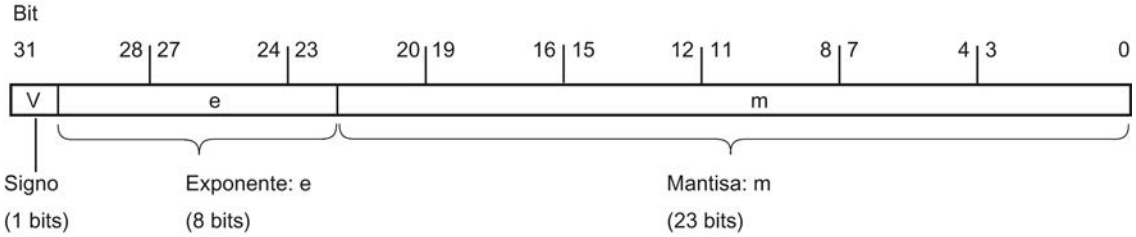
Descripción

Los operandos del tipo de datos REAL tienen una longitud de 32 bits y se utilizan para representar valores en coma flotante. Un operando del tipo de datos REAL consta de los tres componentes siguientes:

- Signo: el estado lógico del bit 31 determina el signo. El bit 31 puede adoptar los valores "0" (positivo) o "1" (negativo).
- Exponentes de 8 bits en base 2: El exponente se incrementa en una constante (base, +127), de modo que presenta un rango de valores de entre 0 y 255.
- Mantisa de 23 bits: se representa solo la fracción de la mantisa. En los números en coma flotante normalizados, la parte entera de la mantisa siempre es 1, y no se guarda.

El tipo de datos REAL se procesa con una precisión de 7 posiciones tras la coma.

La figura siguiente muestra la estructura del tipo de datos REAL:



Nota

En el caso de los números en coma flotante solo se guardan las precisiones definidas en la norma IEEE754. Las posiciones decimales indicadas adicionalmente se redondean según IEEE754.

En cálculos aritméticos anidados con frecuencia puede reducirse el número de posiciones decimales.

Si se indican más posiciones decimales que el tipo de datos puede guardar, el número se redondea al valor correspondiente a la precisión posible en este rango de valores.

La tabla siguiente muestra las propiedades del tipo de datos REAL:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
32	Números en coma flotante según IEEE754	-3.402823e+38 a -1.175495e-38 ±0.0 +1.175495e-38 a +3.402823e+38	1.0e-5, REAL#1.0e-5
	Números en coma flotante		1.0, REAL#1.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

LREAL

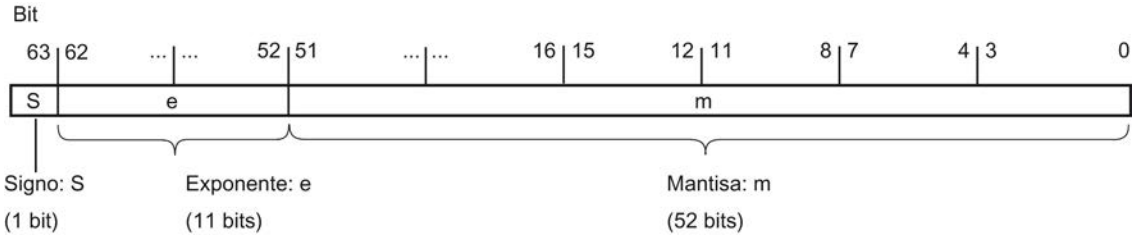
Descripción

Los operandos del tipo de datos LREAL tienen una longitud de 64 bits y se utilizan para representar valores en coma flotante. Un operando del tipo de datos LREAL consta de los tres componentes siguientes:

- Signo: el estado lógico del bit 63 determina el signo. El bit 63 puede adoptar los valores "0" (positivo) o "1" (negativo).
- Exponentes de 11 bits en base 2: El exponente se incrementa en una constante (base, +1023), de modo que presenta un rango de valores de entre 0 y 2047.
- Mantisa de 52 bits: se representa solo la fracción de la mantisa. En los números en coma flotante normalizados, la parte entera de la mantisa siempre es 1, y no se guarda.

El tipo de datos LREAL se procesa con una precisión de 15 posiciones tras la coma.

La figura siguiente muestra la estructura del tipo de datos LREAL:



La tabla siguiente muestra las propiedades del tipo de datos LREAL:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
64	Números en coma flotante según IEEE754	de -1.7976931348623158e+308 a -2.2250738585072014e-308 ±0.0	1.0e-5, LREAL#1.0e-5
	Números en coma flotante	de +2.2250738585072014e-308 a +1.7976931348623158e+308	1.0, LREAL#1.0

Nota

En el caso de los números en coma flotante solo se guardan las precisiones definidas en la norma IEEE754. Las posiciones decimales indicadas adicionalmente se redondean según IEEE754.

En cálculos aritméticos anidados con frecuencia puede reducirse el número de posiciones decimales.

Si se indican más posiciones decimales que el tipo de datos puede guardar, el número se redondea al valor correspondiente a la precisión posible en este rango de valores.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Números en coma flotante no válidos

Descripción

En los tipos de datos REAL y LREAL se diferencian cuatro rangos numéricos:

- Números normalizados que pueden guardarse con total precisión
- Números desnormalizados que no pueden guardarse con total precisión
- Números infinitos: +Inf/-Inf (Infinity)
- Números no válidos: NaN (Not a Number)

Nota

Los números en coma flotante se depositan como en la norma IEEE754. Los resultados de las conversiones o las funciones aritméticas con un número en coma flotante desnormalizado, infinito o NaN (Not a Number) dependen de la CPU.

Si no se trabaja con números en coma flotante normalizados en las funciones matemáticas, el resultado contenderá diferencias importantes, en función de la serie de CPU que se utilice.

Exceptuando las versiones de CPU antiguas de las series S7-300 y S7-400, una CPU no puede realizar cálculos con números en coma flotante desnormalizados; el patrón de bits de un número desnormalizado se interpreta como un cero. Si el resultado de un cálculo se incluye en este rango, se continúa con cero y los bits de estado OV y OS se activan (rebase por defecto del rango numérico).

Aunque los valores de los números en coma flotante no válidos solo pueden representarse con una exactitud limitada en las funciones matemáticas, los números con un exponente de -39 (p. ej., 2.4408e-039) pueden observarse en el TIA Portal y no por ello representan

necesariamente un resultado erróneo. En consecuencia, puede ocurrir que los valores de los números en coma flotante estén fuera de los valores numéricos válidos.

Nota

Para las CPU de las series S7-1200 V1, V2 y V3 rige lo siguiente:

La operación de comparación "Igual" utiliza el patrón de bits del número en coma flotante no válido. Si se comparan dos "números NaN" con el mismo patrón de bits, la salida de la operación de comparación "Igual" arroja el resultado TRUE.

Nota

Para las CPU de la serie S7-1200 V4 y S7-1500 vale:

Si se comparan dos números no válidos (NaN), el resultado es siempre FALSE, independientemente del patrón de bits del número no válido o de la relación (>, >, ...).

Nota

Comparación de números en coma flotante desnormalizados

Con la operación de comparación "Igual" con dos números en coma flotante desnormalizados, en las CPU de la serie S7-300/400 y la serie S7-1200/1500, la salida se pone al estado lógico "0" y "1", respectivamente.

Si las variables de entrada de una función matemática representan un número en coma flotante no válido, también se emite un número en coma flotante no válido como resultado.

Posibilidades de evaluar posibles errores derivados de números en coma flotante no válidos:

- En KOP/FUP y SCL puede consultarse FALSE en la salida de habilitación ENO
- En AWL puede evaluarse el bit de estado OV

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Temporizadores

S5TIME (tiempo)

Formato

El tipo de datos S5TIME guarda el tiempo en formato BCD. El tiempo es el producto de un valor de tiempo en el rango de 0 a 999 y una base de tiempo. La base de tiempo especifica el intervalo en el que un temporizador reduce el valor temporal en una unidad, hasta llegar a "0". La base de tiempo permite controlar la resolución de los valores temporales.

La tabla siguiente muestra el rango de valores para el tipo de datos S5TIME:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
16	Temporizadores de S7 en intervalos de 10 ms (valor estándar)	S5T#0MS a S5T#2H_46M_30S_0MS	S5T#10s, S5TIME#10s
	Números hexadecimales	16#0 a 16#3999	16#2

La tabla siguiente muestra la codificación de la base de tiempo para S5TIME:

Base de tiempo	Código binario para la base de tiempo
10 ms	00
100 ms	01
1 s	10
10 s	11

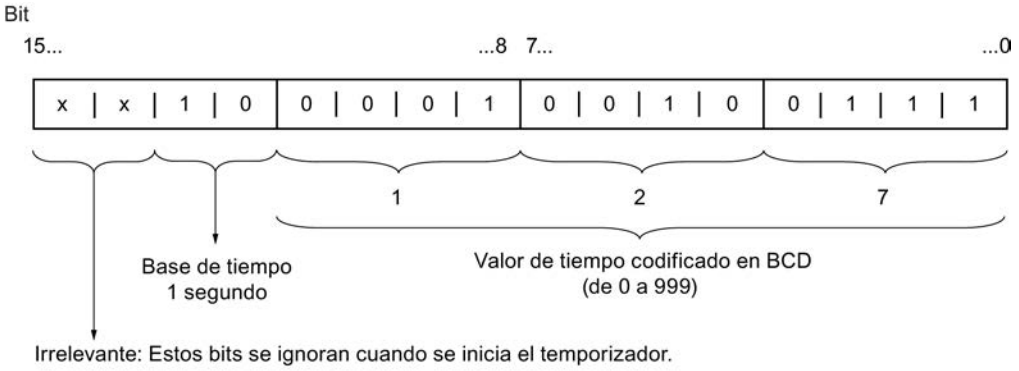
Si el tipo de datos S5TIME se utiliza con temporizadores, deben observarse los valores límite del rango y la resolución de los valores de tiempo. La tabla siguiente indica el rango correspondiente a cada una de las resoluciones:

Resolución	Rango
0,01 s	de 10 ms a 9 s 990 ms
0,1 s	de 100 ms a 1 m 39 s 900 ms
1 s	de 1 s a 16 m 39 s
10 s	de 10 s a 2 h 46 m 30 s

No se aceptan valores por encima de 2h46m30s.

Ejemplo

La figura siguiente muestra el contenido del operando de tiempo para un valor de tiempo de 127 y una base de tiempo de 1 s:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

TIME (tiempo IEC)

Descripción

El contenido de un operando del tipo TIME se interpreta como milisegundos. La representación contiene especificaciones de días (d), horas (h), minutos (m), segundos (s) y milisegundos (ms).

La tabla siguiente muestra las propiedades del tipo de datos TIME:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
32	Tiempo con signo	T#-24d20h31m23s648ms bis T#+24d20h31m23s647ms	T#10d20h30m20s630ms, TIME#10d20h30m20s630ms
	Números hexadecimales	16#00000000 hasta 16#7FFFFFFF	16#0001EB5E

No es necesario indicar todas las unidades de tiempo. Por ejemplo, T#5h10s es válido. Si se indica solo una unidad, el valor absoluto de días, horas y minutos no podrá exceder los límites superiores ni inferiores. Si se indica más de una unidad de tiempo, el valor correspondiente no podrá exceder 24 días, 23 horas, 59 minutos, 59 segundos o 999 milisegundos.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

LTIME (temporizador CEI)

Descripción

El contenido de un operando del tipo LTIME se interpreta como nanosegundos. La representación contiene especificaciones de días (d), horas (h), minutos (m), segundos (s), milisegundos (ms), microsegundos (us) y nanosegundos (ns).

La tabla siguiente muestra las propiedades del tipo de datos LTIME:

Longitud (bits)	Formato	Rango de valores	Ejemplos de entrada de valores
64	Tiempo con signo	de LT#-106751d23h47m16s854ms775us808ns a LT#+106751d23h47m16s854ms775us807ns	LT#11350d20h25m14s830ms652us315ns, LTIME#11350d20h25m14s830ms652us315ns
	Números hexadecimales	16#0 a 16#8000000000000000	16#2

No es necesario indicar todas las unidades de tiempo. Por ejemplo, LT#5h10s es válido. Si se indica sólo una unidad, el valor absoluto de días, horas y minutos no podrá exceder los límites superiores ni inferiores. Si se indica más de una unidad de tiempo, el valor correspondiente no podrá exceder 106751 días, 23 horas, 59 minutos, 59 segundos, 999 milisegundos, 999 microsegundos o 999 nanosegundos.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Fecha y hora

DATE

Formato

El tipo de datos DATE guarda una fecha como número entero sin signo. La representación contiene el año, el mes y el día.

El contenido de un operando del tipo de datos DATE corresponde al número de días desde 01/01/1990, expresado en formato hexadecimal (16#0000).

La tabla siguiente muestra las propiedades del tipo de datos DATE:

Longitud (bytes)	Formato	Rango de valores	Ejemplos de entrada de valores
2	Fecha IEC (año-mes-día)	de D#1990-01-01 a D#2168-12-31	D#2009-12-31, DATE#2009-12-31
	Números hexadecimales	de 16#0000 a 16#FF62	16#00F2

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

TOD (TIME_OF_DAY)

Formato

El tipo de datos TOD (TIME_OF_DAY) ocupa una palabra doble y guarda el número de milisegundos desde el inicio del día (0:00 h) como número entero sin signo.

La tabla siguiente muestra las propiedades del tipo de datos TOD:

Longitud (bytes)	Formato	Rango de valores	Ejemplos de entrada de valores
4	Hora del día (horas:minutos:segundos)	TOD#00:00:00.000 a TOD#23:59:59.999	TOD#10:20:30.400, TIME_OF_DAY#10:20:30.400

Es obligatorio especificar las horas, minutos y segundos. Los milisegundos se pueden especificar opcionalmente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

LTOD (LTIME_OF_DAY)

Formato

El tipo de datos LTOD (LTIME_OF_DAY) ocupa dos palabras dobles y guarda el número de nanosegundos desde el comienzo del día (0:00 h) como número entero sin signo.

La tabla siguiente muestra las propiedades del tipo de datos LTOD:

Longitud (bytes)	Formato	Rango de valores	Ejemplos de entrada de valores
8	Hora (horas:minutos:segundos.milisegundos.microsegundos.nanosegundos)	de LTOD#00:00:00.000000000 a LTOD#23:59:59.999999999	LTOD#10:20:30.400_365_215, LTIME_OF_DAY#10:20:30.400_365_215

Es obligatorio especificar las horas, minutos y segundos. La indicación de los milisegundos, microsegundos y nanosegundos es opcional.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

DT (DATE_AND_TIME)

Formato

El tipo de datos DT (DATE_AND_TIME) guarda información de fecha y hora en formato BCD.

La tabla siguiente muestra las propiedades del tipo de datos DT:

Longitud (bytes)	Formato	Rango de valores	Ejemplo de entrada de valores
8	Fecha y hora (año-mes-día-hora:minuto:segundo:milisegundo [3])	Mín.: DT#1990-01-01-00:00:00.000 Máx.: DT#2089-12-31-23:59:59.999	DT#2008-10-25-8:12:34.567, DATE_AND_TIME#2008-10-25-08:12:34.567

La tabla siguiente muestra la estructura del tipo de datos DT:

Byte	Contenido	Rango de valores
0	Año	de 0 a 99 (años 1990 a 2089) BCD#90 = 1990 ... BCD#0 = 2000 ... BCD#89 = 2089
1	Mes	BCD#0 a BCD#12
2	Día	BCD#1 a BCD# 31
3	Hora	BCD#0 a BCD#23
4	Minuto	BCD#0 a BCD#59
5	Segundo	BCD#0 a BCD#59
6	Los dos valores más significativos de MSEC	BCD#0 a BCD#999

Byte	Contenido	Rango de valores
7 (4MSB) [1]	El valor menos significativo de MSEC	BCD#0 a BCD#9
7 (4LSB) [2]	Día de la semana	BCD#1 a BCD#7 BCD#1 = domingo ... BCD#7 = sábado
[1] MSB: Most Significant Bit [2] LSB: Least Significant Bit [3] Número en coma fija		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

LDT (DATE_AND_LTIME)

Formato

El tipo de datos LDT (DATE_AND_LTIME) guarda información de fecha y hora en nanosegundos desde el 01.01.1970 0:0.

La tabla siguiente muestra las propiedades del tipo de datos LDT:

Longitud (bytes)	Formato	Rango de valores	Ejemplo de entrada de valores
8	Fecha y hora (año-mes-día-hora:minuto:segundo)	Mín.: LDT#1970-01-01-0:0:0.0000 00000 Máx.: LDT#2200-12-31-23:59:59.9 99999999	LDT#2008-10-25-8:12:34.567
	Números hexadecimales	de 16#0 a 16#7FFF_FFFF_FFFF_FFF F	16#7FFF

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

DTL

Descripción

Un operando del tipo de datos DTL tiene una longitud de 12 bytes y guarda datos de fecha y hora en una estructura predefinida.

La tabla siguiente muestra las propiedades del tipo de datos DTL:

Longitud (bytes)	Formato	Rango de valores	Ejemplo de entrada de valores
12	Fecha y hora (año-mes-día-hora:minuto:segundo.nanosegundo)	Mín.: DTL#1970-01-01-00:00:00.0 Máx.: DTL#2200-12-31-23:59:59.999999999	DTL#2008-12-16-20:30:20.250

La estructura del tipo de datos DTL consta de varios componentes, cada uno de los cuales puede tener un tipo de datos y un rango de valores distinto. El tipo de datos de un valor indicado debe coincidir con el tipo de datos del componente en cuestión.

La tabla siguiente muestra los componentes de la estructura del tipo de datos DTL y sus propiedades:

Byte	Componente	Tipo de datos	Rango de valores
0	Año	UINT	de 1970 a 2200
1			
2	Mes	USINT	de 1 a 12
3	Día	USINT	1 hasta 31
4	Día de la semana	USINT	de 1 (domingo) a 7 (sábado) El día de la semana no se tiene en cuenta al introducir los valores.
5	Hora	USINT	0 hasta 23
6	Minuto	USINT	0 hasta 59
7	Segundo	USINT	0 hasta 59
8	Nanosegundo	UDINT	de 0 a 999999999
9			
10			
11			

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Cadenas de caracteres

CHAR (carácter)

Descripción

Un operando del tipo de datos CHAR tiene una longitud de 8 bits y ocupa un BYTE en la memoria.

El tipo de datos CHAR almacena un solo carácter en formato ASCII. Encontrará información sobre la codificación de caracteres especiales en "Consulte también > STRING".

La tabla siguiente muestra el rango de valores del tipo de datos CHAR:

Longitud (bits)	Formato	Rango de valores	Ejemplo de entradas de valores
8	Caracteres ASCII	Juego de caracteres ASCII	'A', CHAR#'A'

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

STRING (Página 2350)

WCHAR (carácter)

Descripción

Un operando del tipo de datos WCHAR (wide characters) tiene una longitud de 16 bits y ocupa dos BYTE en la memoria.

El tipo de datos WCHAR almacena un carácter individual de un juego de caracteres ampliado en formato Unicode. No obstante, solo se cubre una parte de todo el rango Unicode. Si se introduce un carácter de control, se representa con un carácter de dólar.

La tabla siguiente muestra el rango de valores del tipo de datos WCHAR:

Longitud (bits)	Formato	Rango de valores	Ejemplo de entrada de valores
16	Unicode	$0000 - $D7FF	WCHAR#'a'

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Constantes (Página 2288)

STRING

Descripción

Un operando del tipo de datos STRING guarda varios caracteres en una cadena que puede estar formada por un máximo de 254 caracteres. En las cadenas de caracteres se admiten todos los caracteres del código ASCII. Los caracteres se introducen entre comillas sencillas.

La tabla siguiente muestra las propiedades de una variable STRING:

Longitud (bytes)	Formato	Rango de valores	Ejemplo de entrada de valores
n + 2 *	Cadena de caracteres ASCII, incluidos los caracteres especiales	0 a 254 caracteres	'Nombre', STRING#'NAME'
* Un operando del tipo de datos STRING ocupa en la memoria dos bytes más que la longitud máxima especificada.			

Una cadena de caracteres puede contener también caracteres especiales. Para especificar caracteres de control, caracteres de dólar y comillas simples se utiliza el carácter de escape $.

La tabla siguiente muestra ejemplos de notación de caracteres especiales:

Carácter	Hex	Significado	Ejemplo
$L o $l	0A	Avance de línea	'$LText', '$0AText'
$N	0A y 0D	Salto de línea El salto de línea ocupa 2 caracteres en la cadena.	'$NText', '$0A$0DText'
$P o $p	0C	Salto de página	'$PText', '$0CText'
$R o $r	0D	Retorno de carro (CR)	'$RText','$0DText'
$T o $t	09	Tabulación	'$TText', '$09Text'
$$	24	Carácter de dólar	'100$$t', '100$26'
$'	27	Comilla simple	'$'Text$','$27Text$27'

En la declaración de un operando, la longitud máxima de la cadena de caracteres puede indicarse entre corchetes a continuación de la palabra clave STRING (p. ej. STRING[4]). Si no se indica la longitud máxima, se utilizará para el correspondiente operando la longitud estándar de 254 caracteres.

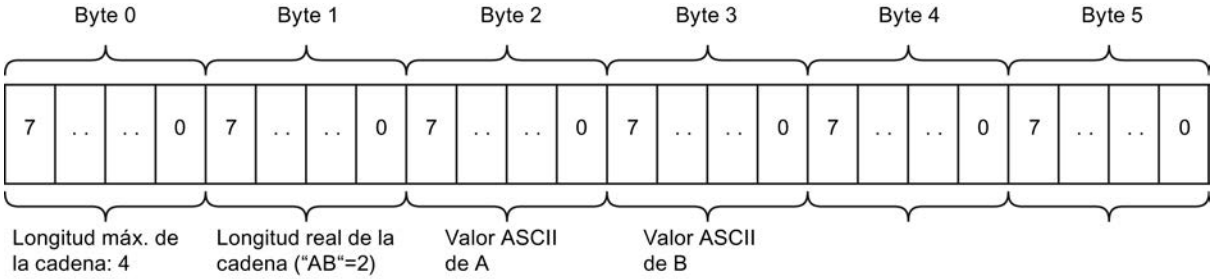
Si la longitud efectiva de una cadena de caracteres especificada es menor que la longitud máxima declarada, los caracteres se escriben justificados a la izquierda en la cadena y las posiciones sobrantes quedan sin definir. Al procesar los valores se consideran únicamente las posiciones de carácter ocupadas.

Nota

Para las CPU S7-300/400 se debe tener en cuenta lo siguiente: Si se ha definido una variable temporal del tipo de datos STRING, en el BYTE "longitud máx. de la cadena de caracteres" debe escribirse la longitud definida antes de utilizar la variable en el programa de usuario.

Ejemplo

El ejemplo siguiente muestra el orden de los bytes si se indica el tipo de datos STRING[4] con el valor de salida 'AB':



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

WSTRING

Descripción

Un operando del tipo de datos WSTRING (Wide String) guarda varios caracteres Unicode del tipo de datos WCHAR en una cadena de caracteres. Si no se indica longitud, la cadena tiene un valor predeterminado de 254 caracteres. En las cadenas de caracteres se admiten todos los caracteres del formato Unicode. Esto permite utilizar también caracteres chinos dentro de una cadena.

En la declaración de un operando del tipo de datos WSTRING es posible definir su longitud utilizando corchetes (p. ej. WSTRING[10]). Si no se indica ninguna longitud, el WSTRING se

ajustará a 254 caracteres. La longitud máxima que puede declararse es de 16382 caracteres (WSTRING[16382]).

Nota

Uso en la tabla de observación

Si desea observar o forzar una variable del tipo WSTRING en la tabla de observación, esta no deberá contener más de 254 caracteres.

Los caracteres se indican entre comillas simples y siempre con el indicador WSTRING#.

La tabla siguiente muestra las propiedades de una variable WSTRING:

Longitud (WORD)	Formato	Rango de valores	Ejemplo de entrada de valores
n + 2 *	Cadena de caracteres Unicode; n indica la longitud de la cadena de caracteres.	Valor predeterminado: 0 a 254 caracteres Valor máx. posible: 0 a 16382	WSTRING#'Hola mundo'
* Un operando del tipo de datos WSTRING ocupa en la memoria dos WORD más que la longitud máxima especificada.			

Una cadena de caracteres puede contener también caracteres especiales. Para especificar caracteres de control, caracteres de dólar y comillas simples se utiliza el carácter de escape $.

La tabla siguiente muestra ejemplos de notación de caracteres especiales:

Carácter	Hex	Significado	Ejemplo
$L o $l	000A	Avance de línea	'$LText', '$000AText'
$N	000A y 000D	Salto de línea El salto de línea ocupa 2 caracteres en la cadena.	'$NText', '$000A $000DText'
$P o $p	000C	Salto de página	'$PText', '$000CText'
$R o $r	000D	Retorno de carro (CR)	'$RText','$000DText'
$T o $t	0009	Tabulación	'$TText', '$0009Text'
$$	0024	Carácter de dólar	'100$$t', '100$0024t'
$'	0027	Comilla simple	'$'Text$','$0027Text $0027'

En la declaración de un operando, la longitud máxima de la cadena de caracteres puede indicarse entre corchetes a continuación de la palabra clave WSTRING (p. ej. WSTRING[4]). Si no se indica la longitud máxima, se utilizará para el correspondiente operando la longitud estándar de 254 caracteres.

Si la longitud efectiva de una cadena de caracteres especificada es menor que la longitud máxima declarada, los caracteres se escriben justificados a la izquierda en la cadena y las

posiciones sobrantes quedan sin definir. Al procesar los valores se consideran únicamente las posiciones de carácter ocupadas.

Nota

Conversión de variables WSTRING

No es posible la conversión implícita del tipo de datos WSTRING. En general, es posible la conversión explícita del tipo de datos WSTRING a STRING. Sin embargo, en todas las páginas de códigos de Windows solo funcionan por defecto las conversiones de los caracteres incluidos en el rango de código 0 - 127. En todos los caracteres que queden por encima de este rango, el carácter Codepage y el carácter Unicode deben estar exactamente en el mismo lugar para que la conversión funcione correctamente.

Acceso a parámetros de bloque del tipo de datos WSTRING

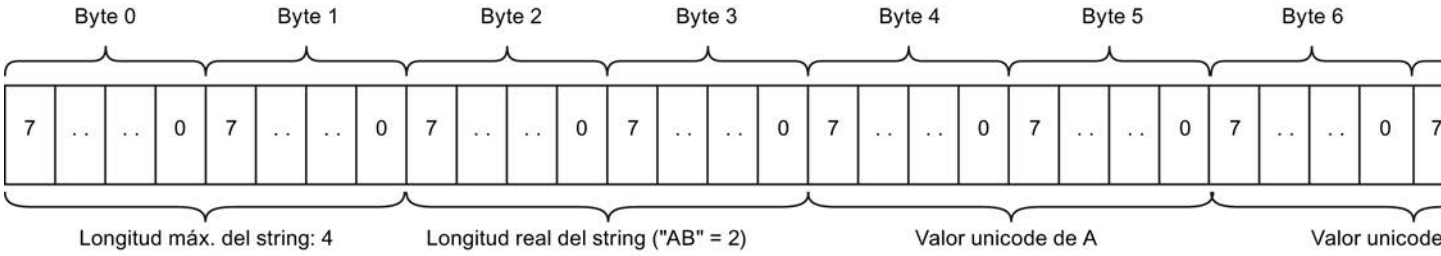
En los bloques con el tipo de acceso "optimizado" los operandos del tipo de datos WSTRING pueden transmitirse como parámetros hasta la longitud máxima.

En los bloques de función (FB) con el tipo de acceso "estándar", los operandos del tipo de datos WSTRING pueden declararse como parámetros en todas las secciones de la interfaz del bloque, excepto en la sección "InOut". En una función (FC) con el tipo de acceso "estándar" solo pueden transmitirse como parámetros operandos del tipo de datos STRING.

Otras excepciones a esta norma son el valor de función de una FC en el apartado "Return" de la interfaz del bloque y las expresiones en el lenguaje de programación SCL. En estos casos, la variable WSTRING no puede tener más de 1022 caracteres. Si necesita una variable WSTRING con más de 1022 caracteres, declare en la sección "Temp" de la interfaz del bloque una variable del tipo de datos "WSTRING" con la longitud de caracteres deseada y asigne el valor de función a la variable.

Ejemplo

El ejemplo siguiente muestra el orden de los bytes si se indica el tipo de datos WSTRING[4] con el valor de salida 'AB':



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Constantes (Página 2288)

Matriz

Formato de Array (límites de 16 bits)

Descripción

El tipo de datos Array representa una estructura de datos compuesta por un número fijo de componentes del mismo tipo de datos. Se admiten todos los tipos de datos, a excepción de Array.

Una variable del tipo de datos Array comienza siempre en un límite de WORD.

Para direccionar los componentes de campo se utilizan índices. Los límites de los índices se definen entre corchetes en la declaración del campo después de la palabra clave Array. El límite inferior debe ser menor o igual que el límite superior. Un campo puede contener hasta seis dimensiones, cuyos límites se especifican separados entre sí por comas.

La tabla siguiente muestra la propiedad del tipo de datos Array:

Longitud	Formato	Límites de índice	Tipo de datos
Número de componentes * longitud del tipo de datos	Array [valor límite inferior.. valor límite superior] of <tipo de datos>	[-32768..32767] of <tipo de datos>	Secuencias de bits, enteros, números en coma flotante, temporizadores, cadenas de caracteres, estructuras

Ejemplo

El siguiente ejemplo muestra cómo pueden declararse los operandos del tipo de datos Array:

Nombre	Declaración	Comentario
Valor medido	Array[1..20] of REAL	Campo unidimensional de 20 componentes
Hora	Array[-5..5] of INT	Campo unidimensional de 11 componentes
Carácter	Array[1..2, 3..4] of CHAR	Campo bidimensional de 4 componentes

Límites de Array máximos

Los límites de Array máximos dependen de los siguientes factores:

- Tipo de datos de los elementos Array
- Capacidad de memoria máxima de la CPU (encontrará información adicional en el manual de producto correspondiente)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Indexación indirecta de componentes de ARRAY (Página 2307)

Formato de Array (límites de 32 bits)

Descripción

El tipo de datos Array representa una estructura de datos compuesta por un número fijo de componentes del mismo tipo de datos. Se admiten todos los tipos de datos, a excepción de Array.

Para direccionar los componentes de campo se utilizan índices. Los límites de los índices se definen entre corchetes en la declaración del campo después de la palabra clave Array. El límite inferior debe ser menor o igual que el límite superior. Un campo puede contener hasta seis dimensiones, cuyos límites se especifican separados entre sí por comas.

Nota

Dependiendo de la CPU, la capacidad de memoria de un DB está limitada, y con ello también el número de componentes del Array. No obstante, el direccionamiento de los componentes de campo puede iniciarse en cualquier punto dentro de los límites de los índices.

La tabla siguiente muestra la propiedad del tipo de datos Array:

Longitud	Formato	Límites de índice	Tipo de datos
Número de componentes * longitud del tipo de datos	Array [valor límite inferior.. valor límite superior] of <tipo de datos>	[-2147483648..2147483647] of <tipo de datos>	Secuencias de bits, enteros, números en coma flotante, temporizadores, cadenas de caracteres, estructuras

Nota

La longitud del Array depende de si el bloque se ha creado con la propiedad "Estándar" o "con acceso optimizado".

Nota

Válido para las CPU de la serie S7-1500

Con un bloque con la propiedad "Acceso optimizado al bloque", un bit requiere 1 byte de memoria. Esto también es válido si se utiliza un ARRAY of <tipo de datos>. Así, p. ej., un ARRAY [0..1] of BOOL requiere 2 bytes en un bloque optimizado.

Ejemplo

El siguiente ejemplo muestra cómo pueden declararse los operandos del tipo de datos Array:

Nombre	Declaración	Comentario
Valor medido	Array[1..20] of REAL	Campo unidimensional de 20 componentes
Hora	Array[-5..5] of INT	Campo unidimensional de 11 componentes
Carácter	Array[1..2, 3..4] of CHAR	Campo bidimensional de 4 componentes

Límites de Array máximos

Los límites de Array máximos dependen de los siguientes factores:

- Tipo de datos de los elementos Array
- Reserva de memoria (solo en bloques con acceso optimizado)
  Encontrará más información al respecto en "Cargar cambios de bloque sin reinicializar".
- Tamaño máximo de un bloque de datos de una CPU (encontrará información adicional en el manual de producto correspondiente)
- Dentro de un bloque de datos se dispone de toda la longitud del Array. Dentro de un bloque de programa (OB, FB o FC) se reduce la longitud posible según la capacidad de memoria requerida por el código de programa.

Ejemplo de una CPU de la serie S7-1200

La tabla siguiente muestra el número máximo de elementos de un bloque con la propiedad "con acceso optimizado":

Ancho de tipo de datos (bits)	Número máximo de elementos	Observación
1	524272	= 65534*8
8	65534	El valor se puede consultar en el manual de producto de la CPU.
16	32767	= 65534/2 (división de entero, resto 0)
32	16383	= 65534/4 (división de entero, resto 2)
64	8191	= 65534/8 (división de entero, resto 6)

Debido a diferentes limitaciones técnicas o internas es posible que el área de memoria útil real sea aprox. 70 - 100 bytes menor. Asimismo, el área de memoria puede estar todavía más limitado por ajustes previos, p. ej. por la propiedad de bloque "Cargar sin reinicializar".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Indexación indirecta de componentes de ARRAY (Página 2307)

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)
Activar la reserva de memoria (Página 3033)

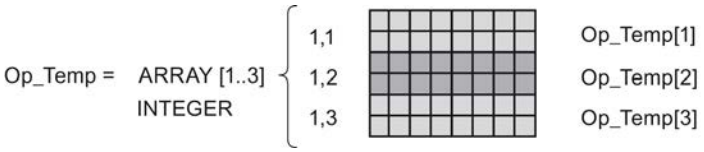
Ejemplo de un Array unidimensional

Declaración

La tabla siguiente muestra la declaración de una variable Array unidimensional:

Nombre	Tipo de datos	Comentario
Op_Temp	Array[1..3] of INT	Variable Array unidimensional con 3 componentes.

La figura siguiente muestra la estructura de la variable Array declarada:



Acceso al componente de campo

A los valores de los distintos componentes de campo se accede a través de un índice. El índice del primer componente de campo es [1], el del segundo [2], y el del tercero [3]. En este caso, para acceder al valor del segundo componente de campo es preciso indicar "Op_Temp[2]" en el programa.

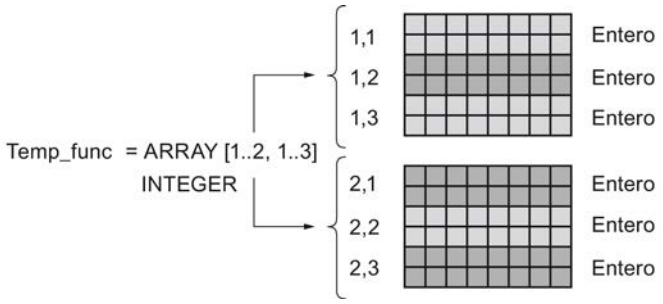
Ejemplo de Array multidimensional

Declaración

La tabla siguiente muestra la declaración de una variable Array bidimensional:

Nombre	Tipo de datos	Valor	Comentario
Betr_Temp	Array[1..2, 1..3] of INT	1,1,4(0)	Variable Array bidimensional de 6 componentes. A los dos primeros componentes se les asigna el valor "1". A los cuatro componentes restantes se les asigna el valor "0".

La figura siguiente muestra la estructura de la variable Array declarada:



Acceder a los componentes del campo

Para acceder a los valores de los distintos componentes del campo se utiliza un índice. El índice del primer componente del campo es p. ej. [1,1] y el índice del cuarto componente del campo, [2,1]. En el programa es necesaria, p. ej., la expresión "Betr_Temp[2,1]" para acceder al valor del cuarto componente del campo.

Otras posibilidades de acceso

La variable "Betr_Temp" se puede declarar también como campo hexadimensional. La tabla siguiente muestra un ejemplo de declaración de una variable Array hexadimensional:

Nombre	Tipo de datos	Valor	Comentario
Betr_Temp	Array[1..3, 1..2, 1..3, 1..4, 1..3, 1..4] of INT	-	Variable Array hexadimensional

En este caso, el índice del primer componente del campo es [1,1,1,1,1,1] y el índice del último componente es [3,2,3,4,3,4]. Para las posiciones intermedias se especifica el valor correspondiente a cada dimensión.

Estructuras

STRUCT

Descripción

El tipo de datos STRUCT representa una estructura de datos compuesta por un número fijo de componentes de diferentes tipos de datos. Los componentes de los tipos de datos STRUCT o ARRAY también pueden anidarse en una estructura. La profundidad de anidamiento se limita a ocho niveles. Con ayuda de estructuras es posible agrupar datos en función del control del proceso y transmitir parámetros como una unidad de datos.

En las CPU de las series S7-1200 o S7-1500 pueden crearse hasta 65534 estructuras. Cada una de ellas puede contener hasta 252 componentes. Además, pueden crearse hasta 65534 bloques de función, 65535 funciones y 65535 bloques de organización con hasta 252 componentes cada uno.

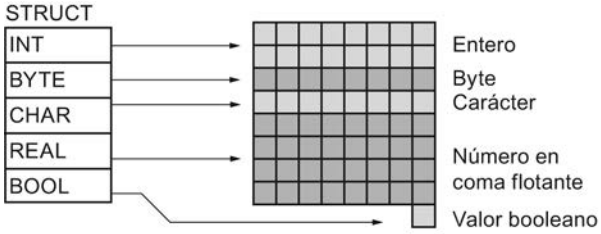
Un componente del tipo de datos ARRAY comienza siempre en un límite de WORD.

La tabla siguiente muestra las propiedades del tipo de datos STRUCT:

Longitud	Formato	Rango de valores	Ejemplo de entrada de valores
Una variable STRUCT comienza en un byte con dirección par y ocupa la memoria hasta el siguiente límite de palabra.	STRUCT	Son aplicables los rangos de valores de los tipos de datos utilizados.	Rigen las reglas de entrada de valores para los tipos de datos utilizados.

Ejemplo

La figura siguiente muestra un ejemplo de la estructura de una variable STRUCT:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Puntero

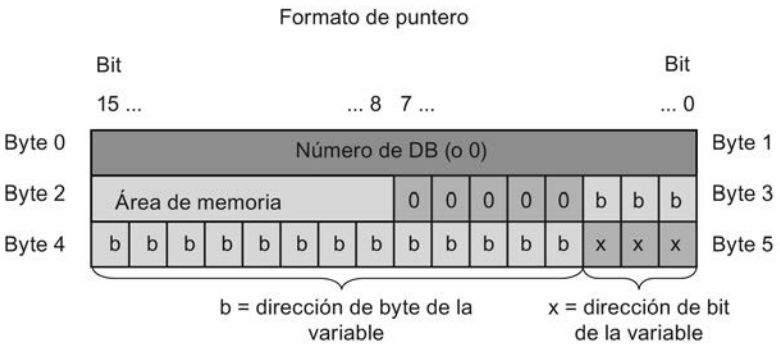
POINTER

Descripción

Un parámetro del tipo POINTER es un puntero que puede señalar a una variable específica. Ocupa 6 bytes (48 bits) en la memoria y puede contener la siguiente información acerca de una variable:

- Número del bloque de datos, o bien 0 en caso de que los datos no estén almacenados en un bloque de datos
- Área de memoria en la CPU
- Dirección de la variable

La figura siguiente muestra la estructura del tipo de parámetro POINTER:



Tipos de punteros

El tipo de parámetro POINTER permite declarar los cuatro tipos siguientes de punteros, en función de la información que deban contener:

- Puntero intraárea:
  Un puntero intraárea contiene información sobre la dirección de una variable.
- Puntero interárea:
  Un puntero interárea contiene información sobre el área de memoria y la dirección de un operando.
- Puntero DB:
  Un puntero DB permite apuntar a una variable de un bloque de datos. Un puntero DB contiene, además del área de memoria y la dirección de una variable, el número de un bloque de datos.
- Puntero cero:
  Con un puntero cero se muestra la falta de un valor. La falta de un valor puede significar por un lado que no existe ningún valor o que el valor todavía no se conoce. El valor cero equivale a la ausencia de valor, pero al mismo tiempo es un valor.

La tabla siguiente muestra los formatos para la declaración de los diferentes tipos de punteros:

Representación de P#Byte	Formato	Ejemplo de entrada de valores	Descripción
Simbólico	P#Byte.Bit	"MyTag"	Puntero intraárea
	P#ByteAreaOperando.Bit	"MyVariable"	Puntero interárea
	P#BloqueDatos.OperandoDatos	"MyDB"."MyTag"	Puntero DB
	P#Valor cero	-	Puntero cero
Absoluto	P#Byte.Bit	P#20.0	Puntero intraárea
	P#ByteAreaOperando.Bit	P#M20.0	Puntero interárea
	P#BloqueDatos.OperandoDatos	P#DB10.DBX20.0	Puntero DB
	P#Valor cero	P#0.0, NULL	Puntero cero

Para un parámetro del tipo POINTER se puede introducir el valor actual sin prefijo (P#). En tal caso, el texto introducido se convierte automáticamente al formato POINTER.

Nota

Si se utiliza el prefijo P#, solo es posible señalar a las áreas de memoria con tipo de acceso "estándar".

Áreas de memoria

La tabla siguiente muestra los códigos hexadecimales de las áreas de memoria del tipo de parámetro POINTER:

Código hexadecimal	Área de memoria	Descripción
B#16#80[1)]	P	Periferia en una CPU S7-300/400
16#1	P	Entradas de periferia en una CPU S7-1500
16#2	P	Salidas de periferia en una CPU S7-1500
B#16#81	I	Área de memoria de las entradas
B#16#82	Q	Área de memoria de las salidas
B#16#83	M	Área de memoria de las marcas
B#16#84	DBX	Bloque de datos
B#16#85	DIX	Bloque de datos de instancia
B#16#86	L	Datos locales
B#16#87	V	Datos locales anteriores
[1)] Estos tipos de datos solo pueden utilizarse para el puntero POINTER en una CPU S7-300/400.		

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

ANY

Descripción

Un parámetro del tipo ANY es un puntero que señala el inicio de un área de datos y especifica su longitud. Un puntero ANY ocupa 10 bytes en la memoria y puede contener la siguiente información:

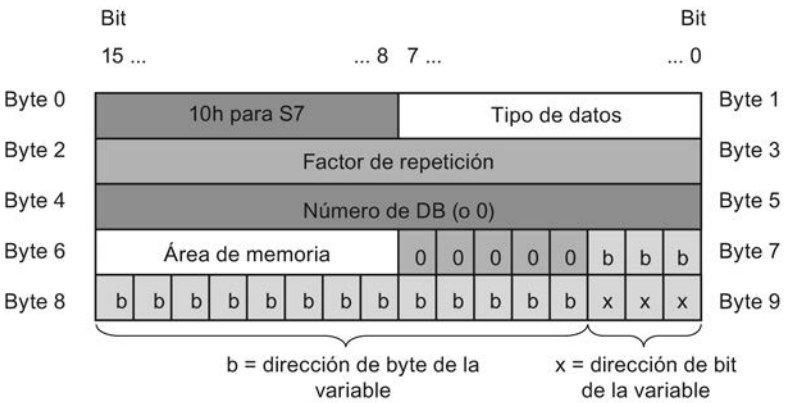
- Tipo de datos:
  Tipo de datos de los elementos del área de datos.
- Factor de repetición:
  Número de elementos del área de datos
- Número de DB:
  Bloque de datos en el que están declarados los elementos del área de datos.
- Área de memoria:
  Área de memoria de la CPU en la que están depositados los elementos del área de datos.
- Dirección inicial de los datos en formato "Byte.Bit":
  Inicio del área de datos al que apunta el punteroANY.
- Puntero cero:
  Con un puntero cero se muestra la falta de un valor. La falta de un valor puede significar por un lado que no existe ningún valor o que el valor todavía no se conoce. El valor cero equivale a la ausencia de valor, pero al mismo tiempo es un valor.

Nota

Área de memoria

El puntero ANY solo puede apuntar a las áreas de memoria con tipo de acceso "Estándar", incluso en una CPU S7-1500.

La figura siguiente muestra la estructura del puntero ANY:



Un puntero ANY no puede reconocer estructuras. Solo puede asignarse a variables locales.

La tabla siguiente muestra los formatos para la declaración de un puntero ANY:

Representación	Formato	Ejemplo de entrada de valores	Descripción
Simbólico	P#BloqueDatos.ÁreaMemoria DirecciónDatos Tipo Número	"MyDB".VariableStruct.FirstComponent	Área de 10 palabras en el DB global 11 a partir de DBB20.0
	P#ÁreaMemoria DirecciónDatos Tipo Número	"MyMarkerTag"	Área de 4 bytes a partir de MB20.0
		"MyTag"	Entrada I1.0
	P#Valor cero	-	Valor cero
Absoluto	P#BloqueDatos.ÁreaMemoria DirecciónDatos Tipo Número	P#DB11.DBX20.0 INT 10	Área de 10 palabras en el DB global 11 a partir de DBB20.0
	P#ÁreaMemoria DirecciónDatos Tipo Número	P#M20.0 BYTE 10	Área de 10 bytes a partir de MB20.0
		P#I1.0 BOOL 1	Entrada I1.0
	P#Valor cero	P#0.0 VOID 0, NULL	Valor cero

Codificación de los tipos de datos

La tabla siguiente muestra la codificación de los tipos de datos para el puntero ANY:

Código hexadecimal	Tipo de datos	Descripción
B#16#00	NIL	Puntero Cero
B#16#01[1)]	BOOL	Bits
B#16#02	BYTE	Bytes, 8 bits
B#16#03	CHAR	Caracteres de 8 bits
B#16#04	WORD	Palabras de 16 bits
B#16#05	INT	Enteros de 16 bits
B#16#06	DWORD	Palabras de 32 bits
B#16#07	DINT	Enteros de 32 bits
B#16#08	REAL	Números en coma flotante de 32 bits
B#16#0B	TIME	Tiempo
B#16#0C	S5TIME	Tiempo
B#16#09	DATE	Fecha
B#16#0A	TOD	Fecha y hora
B#16#0E	DT	Fecha y hora
B#16#13	STRING	Cadena de caracteres
B#16#17[1)]	BLOCK_FB	Bloque de función
B#16#18[1)]	BLOCK_FC	Función
B#16#19[1)]	BLOCK_DB	Bloque de datos
B#16#1A[1)]	BLOCK_SDB	Bloque de datos de sistema
B#16#1C[1)]	COUNTER	Contador
B#16#1D[1)]	TIMER	Temporizador
[1)] Estos tipos de datos solo pueden utilizarse para el puntero ANY en una CPU S7-300/400.		

Codificación de las áreas de memoria

La tabla siguiente muestra la codificación de las áreas de memoria para el puntero ANY:

Código hexadecimal	Área	Descripción
B#16#80[1)]	P	Periferia
B#16#81	I	Área de memoria de las entradas
B#16#82	Q	Área de memoria de las salidas
B#16#83	M	Área de memoria de las marcas
B#16#84	DBX	Bloque de datos
B#16#85	DIX	Bloque de datos de instancia
B#16#86	L	Datos locales
B#16#87	V	Datos locales anteriores
[1)] Estas áreas de memoria solo pueden utilizarse para el puntero ANY en una CPU S7-300/400.		

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

VARIANT

Descripción

Un parámetro del tipo VARIANT es un puntero que puede señalar a variables de distintos tipos de datos, excepto una instancia. El puntero VARIANT puede ser un objeto de un tipo de datos elemental como INT o REAL. También puede ser un STRING, DTL, Array of STRUCT, UDT o un Array of UDT. El puntero VARIANT puede reconocer estructuras y apuntar a componentes individuales de estas. Los operandos del tipo de datos VARIANT no ocupan espacio en el bloque de datos instancia o en la pila L-STACK. Pero dentro del área de memoria de la CPU sí que se ocupa lugar.

Una variable del tipo VARIANT no es un objeto, sino una referencia a otro objeto. Los diferentes elementos del tipo VARIANT solo pueden declararse dentro de la interfaz de bloque de una función. Por este motivo no es posible declararlos en un bloque de datos o una sección Static de la interfaz de un bloque de función, por ejemplo, pues no se conoce su tamaño. El tamaño de los objetos referenciados puede cambiar.

El tipo de datos VARIANT solo puede definirse en parámetros formales de la interfaz de un bloque.

Nota

Un bloque de datos completo solo se puede señalar si ha sido derivado originalmente de un tipo de datos definido por el usuario (UDT).

La tabla siguiente muestra las propiedades del puntero VARIANT:

Longitud (bytes)	Representación	Formato	Ejemplo de entrada de valores
0	Simbólico	Operando	"TagResult"
		NombreBloqueDatos.NombreOperando.Componente	"Datos_TIA_Portal".VariableStruct.PrimerComponente
	Absoluto	Operando	%MW10
		NúmeroBloqueDatos.Operando Tipo Longitud (válido solo para bloques con acceso estándar)	P#DB10.DBX10.0 INT 12
		P#Valor cero	P#0.0 VOID 0, NULL

Nota

Si se utiliza el prefijo P#, solo es posible señalar a las áreas de memoria con tipo de acceso "estándar".

Codificación de los tipos de datos

Si se utiliza el direccionamiento absoluto con P# se admiten los tipos de datos siguientes:

Código hexadecimal	Tipo de datos	Descripción
B#16#00	NIL	Puntero Cero
B#16#01	BOOL	Bits
B#16#02	BYTE	Bytes, 8 bits
B#16#03	CHAR	Caracteres de 8 bits
B#16#04	WORD	Palabras de 16 bits
B#16#05	INT	Enteros de 16 bits
B#16#06	DWORD	Palabras de 32 bits
B#16#07	DINT	Enteros de 32 bits
B#16#08	REAL	Números en coma flotante de 32 bits
B#16#0B	TIME	Tiempo
B#16#0C	S5TIME	Tiempo
B#16#09	DATE	Fecha
B#16#0A	TOD	Fecha y hora
B#16#0E	DT	Fecha y hora
B#16#13	STRING	Cadena de caracteres
B#16#17	BLOCK_FB	Bloque de función
B#16#18	BLOCK_FC	Función
B#16#19	BLOCK_DB	Bloque de datos
B#16#1A	BLOCK_SDB	Bloque de datos de sistema

Código hexadecimal	Tipo de datos	Descripción
B#16#1C	COUNTER	Contador
B#16#1D	TIMER	Temporizador

Ejemplo

El siguiente ejemplo muestra el funcionamiento de VARIANT tomando como base la instrucción AWL "MOVE": Copiar valor":

AWL	Significado
CALL MOVE	// Se llama la instrucción "Copiar valor".
VARIANT	// Tipo de datos de la instrucción
IN := "Datos_TIA_Portal".VariableStruct.PrimerComponente	// Se copia el contenido del operando "PrimerComponente" del DB "Datos_TIA_Portal".
OUT := "MotorDB".StructResult.TagResult	// Y se transfiere al operando "TagResult" del DB "MotorDB".

Consulte también

Fundamentos del direccionamiento indirecto (Página 2305)

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos (Página 2373)

Principios básicos de las constantes (Página 2288)

Fundamentos del acceso a bloques (Página 2258)

Principios básicos de las llamadas de bloque (Página 2263)

Uso de instrucciones VARIANT (Página 269)

Tipos de parámetros

Tipos de parámetros

Descripción

Los tipos de parámetros son tipos de datos para parámetros formales, que se transfieren a bloques llamados. Un tipo de parámetros puede ser también un tipo de datos PLC.

La tabla siguiente muestra los tipos de parámetros disponibles y su significado:

Tipo de parámetro	Longitud (bits)	Descripción
TIMER	16	Permite indicar un temporizador que se utiliza en el bloque lógico llamado. Cuando se transfiere un parámetro formal del tipo TIMER, el parámetro actual correspondiente debe ser siempre un temporizador. Ejemplo: T1
COUNTER	16	Permite indicar un contador que se utiliza en el bloque lógico llamado. Cuando se transfiere un parámetro formal del tipo COUNTER, el parámetro actual correspondiente debe ser siempre un contador. Ejemplo: Z10
BLOCK_FC	16	Permite indicar un bloque que se utiliza como entrada en el bloque lógico llamado. Al declarar un parámetro se determina el tipo de bloque (p. ej. FB, FC, DB) que se debe utilizar. Cuando se transfiere un parámetro formal del tipo BLOCK, el parámetro actual debe ser una dirección de bloque. Ejemplo: DB3
BLOCK_FB	16	
BLOCK_DB	16	
BLOCK_SDB	16	
BLOCK_SFB	16	
BLOCK_SFC	16	
BLOCK_OB	16	
VOID	-	El tipo de parámetro VOID no guarda valores. Este tipo de parámetro se utiliza cuando no se necesitan los valores de retorno de una salida. El tipo de parámetro VOID puede usarse p. ej. en la salida STATUS, cuando no se necesita la información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de los tipos de datos PLC (Página 2967)

Tipos de datos PLC

Tipos de datos PLC

Descripción

Los tipos de datos PLC son estructuras de datos definidas por el usuario, que pueden utilizarse varias veces en el programa. La estructura de un tipo de datos PLC consta de varios componentes, que pueden pertenecer a diferentes tipos de datos. El tipo de los componentes se define en la declaración del tipo de datos PLC.

En las CPU de las series S7-1200 o S7-1500 pueden crearse hasta 65534 tipos de datos PLC. Cada uno de estos tipos de datos PLC puede contener hasta 252 componentes.

Los tipos de datos PLC pueden utilizarse para los siguientes fines:

- Los tipos de datos PLC pueden utilizarse como tipos de datos de variables en la declaración de variables de bloques de código, o en bloques de datos.
- Los tipos de datos PLC pueden usarse como plantilla para la creación de bloques de datos globales con la misma estructura de datos.
- En S7-1200 y S7-1500 es posible utilizar tipos de datos PLC como plantilla para crear variables PLC estructuradas.

Consulte también

Programar tipos de datos PLC (Página 2967)

Direccionar variables estructuradas (Página 2298)

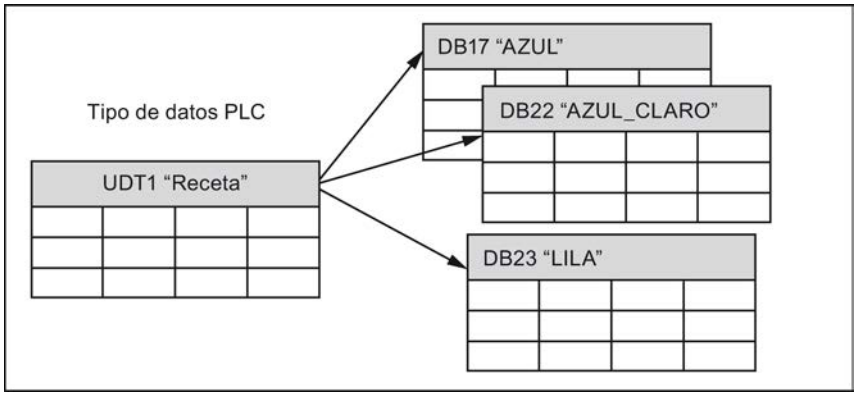
Ejemplo de tipo de datos PLC

Ejemplo

Al crear un bloque de datos puede asignársele el tipo de datos PLC. Tomando como base este tipo puede generarse un gran número de bloques de datos con la misma estructura de datos. Estos bloques de datos pueden adaptarse a la tarea correspondiente introduciendo los valores actuales necesarios en cada caso.

Por ejemplo, puede crearse un tipo de datos PLC para una receta de mezcla de pinturas. Este tipo de datos puede asignarse después a varios bloques de datos, cada uno de los cuales contendrá las cantidades requeridas en su caso.

La figura siguiente muestra este caso de aplicación:



Tipos de datos de sistema

Tipos de datos de sistema

Descripción

El sistema proporciona los tipos de datos de sistema (SDT), que tienen una estructura predefinida. La estructura de un tipo de datos de sistema comprende un número fijo de componentes, que pueden tener distintos tipos de datos. No es posible modificar la estructura de un tipo de datos de sistema.

Los tipos de datos de sistema sólo pueden utilizarse para determinadas aplicaciones. La tabla siguiente muestra los tipos de datos de sistema disponibles y su significado:

Tipo de datos de sistema	Longitud (bytes)	Descripción
IEC_TIMER	16	Estructura de un temporizador cuyos valores de tiempo son del tipo de datos TIME. Este tipo de datos se utiliza p. ej. en las instrucciones "TP", "TOF", "TON", "TONR", "RT" y "PT" .
IEC_LTIMER	32	Estructura de un temporizador cuyos valores de tiempo son del tipo de datos LTIME. Este tipo de datos se utiliza p. ej. en las instrucciones "TP", "TOF", "TON", "TONR", "RT" y "PT" .
IEC_SCOUNTER	3	Estructura de un contador cuyos valores de contaje son del tipo de datos SINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
IEC_USCOUNTER	3	Estructura de un contador cuyos valores de contaje son del tipo de datos USINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
IEC_COUNTER	6	Estructura de un contador cuyos valores de contaje son del tipo de datos INT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
IEC_UCOUNTER	6	Estructura de un contador cuyos valores de contaje son del tipo de datos UINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
IEC_DCOUNTER	12	Estructura de un contador cuyos valores de contaje son del tipo de datos DINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
IEC_UDCOUNTER	12	Estructura de un contador cuyos valores de contaje son del tipo de datos UDINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".

Tipo de datos de sistema	Longitud (bytes)	Descripción
IEC_LCOUNTER	24	Estructura de un contador cuyos valores de contaje son del tipo de datos LINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
IEC_ULCOUNTER	24	Estructura de un contador cuyos valores de contaje son del tipo de datos ULINT. Este tipo de datos se utiliza p. ej. en las instrucciones "CTU", "CTD" y "CTUD".
ERROR_STRUCT	28	Estructura de una información de error referente a un error de programación o de acceso a la periferia. Este tipo de datos se utiliza p. ej. en la instrucción "GET_ERROR".
CREF	8	Componente del tipo de datos ERROR_STRUCT en el que se guarda la información de dirección de un bloque.
NREF	8	Componente del tipo de datos ERROR_STRUCT en el que se guarda la información de dirección de un operando.
VREF	12	Sirve para archivar un puntero VARIANT. Este tipo de datos se utiliza, p. ej., en las instrucciones de S7-1200 Motion Control.
STARTINFO	12	Indica la estructura de datos en la que se guarda la información de arranque. Este tipo de datos se utiliza p. ej. en la instrucción "RD_SINFO".
SSL_HEADER	4	Indica la estructura de datos en la que se guarda la información referente a los registros al leer listas de estado del sistema. Este tipo de datos se utiliza p. ej. en la instrucción "RDSYSST".
CONDITIONS	52	Estructura de datos definida, que establece las condiciones para el inicio y final de una recepción de datos. Este tipo de datos se utiliza p. ej. en la instrucción "RCV_CFG".
TADDR_Param	8	Indica la estructura de un bloque de datos en el que se depositan las descripciones de conexiones para Open User Communication vía UDP. Este tipo de datos se utiliza p. ej. en las instrucciones "TUSEND" y "TURSV".
TCON_Param	64	Indica la estructura de un bloque de datos en el que se depositan las descripciones de conexiones para Open User Communication vía Industrial Ethernet (PROFINET). Este tipo de datos se utiliza p. ej. en las instrucciones "TSEND" y "TRSV".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Tipos de datos de hardware

Tipos de datos de hardware

Descripción

La CPU provee los tipos de datos de hardware. El número de tipos de datos de hardware disponibles depende de la CPU utilizada.

En función de los módulos parametrizados en la configuración de hardware, se crean constantes de un determinado tipo de datos de hardware. Si se inserta una instrucción para controlar o activar un módulo parametrizado en el programa de usuario, las constantes disponibles podrán utilizarse para asignar valores a los parámetros.

La tabla siguiente muestra los tipos de datos de hardware disponibles y su significado:

Tipo de datos	Tipos de datos base	Descripción
REMOTE	ANY	Permite especificar la dirección de una CPU remota. Este tipo de datos se utiliza p. ej. en las instrucciones "PUT" y "GET".
GEOADDR	HW_IOSYSTEM	Indicación de dirección geográfica
HW_ANY	WORD	Identificación de un componente de hardware cualquiera, p. ej. un módulo
HW_DEVICE	HW_ANY	Identificación de un esclavo DP o un dispositivo PROFINET IO
HW_DPMASTER	HW_INTERFACE	Identificación de un maestro DP
HW_DPSLAVE	HW_DEVICE	Identificación de un esclavo DP
HW_IO	HW_ANY	Número de identificación de la CPU o la interfaz El número se asigna automáticamente y se guarda en las propiedades de la CPU o de la interfaz en la configuración de hardware.
HW_IOSYSTEM	HW_ANY	Identificación de un sistema PN/IO o de un sistema maestro DP
HW_SUBMODULE	HW_IO	Identificación de un componente central de hardware
HW_MODULE	HW_IO	Identificación de un módulo
HW_INTERFACE	HW_SUBMODULE	Identificación de un componente de interfaz
HW_IEPORT	HW_SUBMODULE	Identificación de un puerto (PN/IO)
HW_HSC	HW_SUBMODULE	Identificación de un contador rápido Este tipo de datos se utiliza p. ej. en la instrucción "CTRL_HSC".
HW_PWM	HW_SUBMODULE	Identificación de una modulación de ancho de impulsos Este tipo de datos se utiliza p. ej. en la instrucción "CTRL_PWM".
HW_PTO	HW_SUBMODULE	Identificación de un encóder incremental de posición Este tipo de datos se utiliza para Motion Control
AOM_AID	DWORD	Solo se utiliza en combinación con un bloque de función de sistema.

Tipo de datos	Tipos de datos base	Descripción
AOM_IDENT	DWORD	Identificación de un objeto en el sistema runtime del AS
EVENT_ANY	AOM_IDENT	Sirve para indicar un evento cualquiera.
EVENT_ATT	EVENT_ANY	Permite especificar un evento que puede asignarse dinámicamente a un OB Este tipo de datos se utiliza p. ej. en las instrucciones "ATTACH" y "DETACH".
EVENT_HWINT	EVENT_ATT	Permite especificar un evento de alarma de proceso
OB_ANY	INT	Permite especificar un bloque de organización cualquiera.
OB_DELAY	OB_ANY	Permite especificar un bloque de organización al que se llama al producirse una alarma de retardo. Este tipo de datos se utiliza p. ej. en las instrucciones "SRT_DINT" y "CAN_DINT".
OB_TOD	OB_ANY	Indica el número de un bloque de organización de alarma horaria. Este tipo de datos se utiliza p. ej. en las instrucciones "SET_TINT", "CAN_TINT", "ACT_TINT" y "QRY_TINT".
OB_CYCLIC	OB_ANY	Permite especificar un bloque de organización al que se llama al producirse una alarma cíclica.
OB_ATT	OB_ANY	Permite especificar un bloque de organización que puede asignarse dinámicamente a un evento. Este tipo de datos se utiliza p. ej. en las instrucciones "ATTACH" y "DETACH".
OB_PCYCLE	OB_ANY	Permite especificar un bloque de organización que puede asignarse a un evento de la clase de evento "Programa cíclico".
OB_HWINT	OB_ATT	Permite especificar un bloque de organización al que se llama al producirse una alarma de proceso.
OB_DIAG	OB_ANY	Permite especificar un bloque de organización al que se llama al producirse una alarma de diagnóstico.
OB_TIMEERROR	OB_ANY	Permite especificar un bloque de organización al que se llama al producirse un error de tiempo.
OB_STARTUP	OB_ANY	Permite especificar un bloque de organización al que se llama al producirse un evento de arranque.
PORT	HW_SUBMODULE	Permite especificar un puerto de comunicación. Este tipo de datos se utiliza para la comunicación punto a punto.
RTM	UINT	Permite especificar el número de un contador de horas de funcionamiento. Este tipo de datos se utiliza p. ej. en la instrucción "RTM".
PIP	UINT	Permite crear e interconectar un OB "Synchronous Cycle". Este tipo de datos se utiliza en las SFC 26, 27, 126 y 127.
CONN_ANY	WORD	Sirve para indicar una conexión cualquiera.

Tipo de datos	Tipos de datos base	Descripción
CONN_PRG	CONN_ANY	Permite especificar una conexión para la comunicación abierta vía UDP.
CONN_OUC	CONN_ANY	Sirve para indicar una conexión para la comunicación abierta vía Industrial Ethernet (PROFINET).
CONN_R_ID	DWORD	Tipo de datos para el parámetro R_ID en los bloques de comunicación S7.
DB_ANY	UINT	Identificación (número) de un DB cualquiera El tipo de datos "DB_ANY" tiene la longitud 0 en la sección "Temp".
DB_WWW	DB_ANY	Número del DB que describe la aplicación web (p. ej. la instrucción "WWW") El tipo de datos "DB_WWW" tiene la longitud 0 en la sección "Temp".
DB_DYN	DB_ANY	Número del DB generado mediante el programa de usuario

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión de tipos de datos

Conversión de tipos de datos

Sinopsis de la conversión de tipos de datos

Introducción

Para combinar lógicamente varios operandos en una instrucción, es necesario verificar la compatibilidad de los tipos de datos. Esto también es aplicable a las asignaciones, o a la asignación de valores a los parámetros de bloque. Si los tipos de datos de los operandos son diferentes, será preciso realizar una conversión.

La conversión puede realizarse de dos maneras, a saber:

- conversión implícita
  Los lenguajes de programación KOP, FUP, SCL y GRAPH soportan la conversión implícita. La conversión implícita no es posible en el lenguaje de programación AWL.
- Conversión explícita
  Se utiliza una instrucción de conversión explícita antes de ejecutar la instrucción propiamente dicha.

Nota

Conversión de secuencias de bits en SCL

Todas las secuencias de bit (BYTE, WORD, DWORD y LWORD) incluidas en expresiones reciben el mismo tratamiento que los enteros sin signo (USINT, UINT, UDINT y ULINT). Por ello, p. ej. la conversión implícita de DWORD a REAL se lleva a cabo del mismo modo que una conversión de UDINT a REAL.

Conversión implícita

La conversión implícita se realiza automáticamente si los tipos de datos de los operandos son compatibles. La compatibilidad puede verificarse según criterios más o menos estrictos:

- Con verificación IEC (ajuste predeterminado)
  Si está activada la verificación IEC, se aplican las siguientes reglas:
  - La conversión implícita de BOOL a otros tipos de datos no es posible.
  - Solo los tipos de datos REAL, BYTE, WORD, DWORD, DINT, INT, SINT, UDINT, UINT, USINT, TIME, LDT, DTL, DT, TOD, WCHAR y CHAR se pueden convertir de forma implícita.
  - La longitud de bits del tipo de datos de origen no puede rebasar la longitud de bits del tipo de datos de destino. El operando del tipo de datos WORD no puede indicarse, por ejemplo, en un parámetro si en él se espera el tipo de datos BYTE.
- Sin verificación IEC
  Si no está activada la verificación IEC, se aplican las siguientes reglas:
  - La conversión implícita de BOOL a otros tipos de datos no es posible.
  - Solo los tipos de datos REAL, LREAL, BYTE, WORD, DWORD, LWORD, SINT, INT, DINT, LINT, USINT, UINT, UDINT, ULINT, TIME, LTIME, S5TIME, LDT, DTL, TOD, LTOD, DATE, STRING, WSTRING, WCHAR y CHAR se pueden convertir de forma implícita.
  - La longitud de bits del tipo de datos de origen no puede rebasar la longitud de bits del tipo de datos de destino. El operando del tipo de datos DWORD no puede indicarse, por ejemplo, en un parámetro si en él se espera el tipo de datos WORD.
  - En los parámetros de entrada/salida (InOut) la longitud de bits de un operando indicado debe coincidir con la longitud de bits programada para el parámetro correspondiente.

  Nota

  Conversión implícita sin verificación IEC

  El editor de programación identifica con un rectángulo gris los operandos que deben convertirse de forma implícita. Un rectángulo gris oscuro indica que es posible realizar una conversión implícita sin pérdidas de precisión, p. ej. si el tipo de datos INT se convierte a DINT. Un rectángulo gris claro indica que es posible realizar una conversión implícita, aunque pueden producirse fallos en tiempo de ejecución. Si, por ejemplo, el tipo de datos LINT se convierte a DINT y se produce un desbordamiento, la salida de habilitación ENO se pone a "0".

Encontrará más información sobre el ajuste de la verificación IEC y la conversión implícita en el apartado "Consulte también".

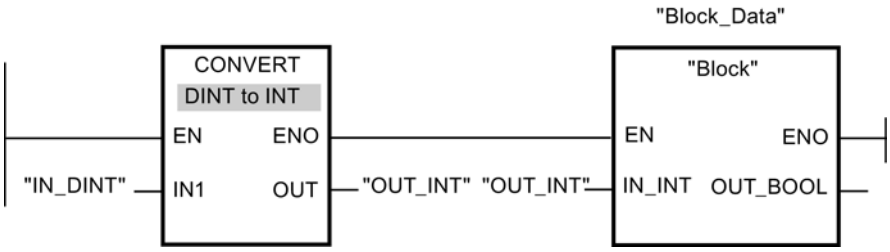
Conversión explícita

Si los operandos no son compatibles, por lo que no se puede realizar una conversión implícita, es posible utilizar una instrucción de conversión explícita. Para ello se pueden utilizar las instrucciones de conversión de la Task Card "Instrucciones" o bien se puede insertar la conversión en el programa manualmente. Encontrará la notación de las funciones de conversión explícita en "Consulte también".

Un posible desbordamiento se indica en la salida de habilitación ENO. Un desbordamiento tiene lugar cuando, p. ej., el valor del tipo de datos de origen es mayor que el valor del tipo de datos de destino.

Encontrará más información sobre la conversión explícita en "Consulte también".

La figura siguiente muestra un ejemplo en el que debe realizarse una conversión explícita de tipos de datos:



El bloque de función "Block" espera en el parámetro de entrada "IN_INT" una variable del tipo de datos INT. Por ello, el valor de la variable "IN_DINT" se debe convertir primero de DINT a INT. Si el valor de la variable "IN_DINT" se encuentra dentro del rango de valores permitido del tipo de datos INT, se realiza una conversión. De lo contrario se notifica un desbordamiento. No obstante, también en caso de desbordamiento se realiza una conversión, pero los valores se cortan y la salida de habilitación ENO se pone a "0".

Consulte también

Conversión implícita (Página 2375)

Conversión explícita (Página 2411)

Conversión implícita

Activar o desactivar la verificación IEC

Se comprueba si los tipos de datos de los operandos utilizados son compatibles. La compatibilidad puede verificarse según criterios más o menos estrictos. Si está activada la "Verificación IEC", se aplican criterios más estrictos.

La verificación IEC se puede ajustar de forma centralizada para todos los bloques nuevos del proyecto, o bien para un bloque en particular.

Ajustar la verificación IEC de bloques nuevos

Para ajustar la verificación IEC de todos los bloques nuevos del programa, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC > General" en la navegación local.
3. En el grupo "Ajuste predeterminado para bloques nuevos" active o desactive la casilla de verificación "Verificación IEC".
   La verificación IEC se activa o desactiva para todos los bloques nuevos del programa.

Ajustar la verificación IEC de un bloque en particular

Para ajustar la verificación IEC de un bloque en particular, proceda del siguiente modo:

1. Abra el bloque.
2. Abra la ficha "Propiedades" de la ventana de inspección.
3. Seleccione el grupo "Atributos" en la navegación local.
4. Active o desactive la casilla de verificación "Verificación IEC".
   La verificación IEC se activa o desactiva para este bloque. El ajuste se guarda junto con el proyecto.

Números binarios

Conversión implícita de BOOL

Posibilidades de conversión implícita

La conversión implícita del tipo de datos BOOL no es posible.

Consulte también

BOOL (bit) (Página 2325)

Sinopsis de la conversión de tipos de datos (Página 2373)

Activar o desactivar la verificación IEC (Página 2375)

Conversión explícita de BOOL (Página 2411)

Secuencias de bits

Conversión implícita de BYTE

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos BYTE:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
BYTE	BOOL	-	-	Ninguna conversión implícita
	WORD	X	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	DWORD	X	X	
	LWORD	X	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

BYTE (bytes) (Página 2326)
Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
Conversión explícita de BYTE (Página 2412)

Conversión implícita de WORD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos WORD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
WORD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El byte menos significativo se transfiere al tipo de datos de destino, y el byte más significativo se ignora.
	DWORD	X	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LWORD	X	X	
	SINT	-	X	El byte menos significativo se transfiere al tipo de datos de destino, y el byte más significativo se ignora.
	USINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LDT	-	-	Ninguna conversión implícita
	DTL	-	-	
	DT	-	-	
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	TOD	-	-	Ninguna conversión implícita
	LTOD	-	-	
	STRING	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

WORD (Página 2327)

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión explícita de WORD (Página 2414)

Conversión implícita de DWORD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DWORD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DWORD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	Los bytes derechos se transfieren al tipo de datos de destino, y los bytes izquierdos se ignoran.
	WORD	-	X	
	LWORD	X	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	-	-	Ninguna conversión implícita
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LTIME	-	-	Ninguna conversión implícita

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LTOD	-	-	Ninguna conversión implícita
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

DWORD (Página 2327)

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión explícita de DWORD (Página 2416)

Conversión implícita de LWORD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos LWORD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
LWORD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	Los bytes derechos se transfieren al tipo de datos de destino, y los bytes izquierdos se ignoran.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	UDINT	-	X	
	LINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	S5TIME	-	-	Ninguna conversión implícita
	LDT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	DTL	-	-	Ninguna conversión implícita
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	STRING	-	-	Ninguna conversión implícita
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

LWORD (Página 2328)

Conversión explícita de LWORD (Página 2418)

Enteros

Conversión implícita de SINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos SINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
SINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0".
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	USINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Transferencia de valores de p. ej. SINT #-1 -> INT #-1, no se rellena con "0".)
	INT	X	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	LINT	X	X	
	ULINT	-	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
SINT (enteros de 8 bits) (Página 2330)
Conversión explícita de SINT (Página 2421)

Conversión implícita de USINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos USINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
USINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0".
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. USINT #10 -> DINT #10, o USINT #128 -> SINT #-128)
	INT	X	X	
	UINT	X	X	
	DINT	X	X	
	UDINT	X	X	
	LINT	X	X	
	ULINT	X	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
USINT (enteros de 8 bits) (Página 2331)
Conversión explícita de USINT (Página 2424)

Conversión implícita de INT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos INT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
INT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. INT #-1 -> SINT #-1, o INT #-32 767 -> UINT #32 769)
	USINT	-	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	LINT	X	X	
	ULINT	-	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	TOD	-	-	Ninguna conversión implícita
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

INT (enteros de 16 bits) (Página 2331)

Conversión explícita de INT (Página 2426)

Conversión implícita de UINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos UINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
UINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. UINT #100 -> DINT #100, o UINT #60 000 -> INT #-5536)
	USINT	-	X	
	INT	-	X	
	DINT	X	X	
	UDINT	X	X	
	LINT	X	X	
	ULINT	X	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	X	X	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	TOD	-	-	Ninguna conversión implícita
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

UINT (enteros de 16 bits) (Página 2332)

Conversión explícita de UINT (Página 2428)

Conversión implícita de DINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> SINT #-1, o DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	UINT	-	X	
	UDINT	-	X	
	LINT	X	X	
	ULINT	-	X	
	REAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 8 388 608)
	LREAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LTIME	-	-	Ninguna conversión implícita
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LTOD	-	-	Ninguna conversión implícita
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

DINT (enteros de 32 bits) (Página 2333)

Conversión explícita de DINT (Página 2430)

Conversión implícita de UDINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos UDINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
UDINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> SINT #-1, o DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	LINT	X	X	
	ULINT	X	X	
	REAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 8 388 608)
	LREAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LTIME	-	-	Ninguna conversión implícita
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	LTOD	-	-	Ninguna conversión implícita
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
UDINT (enteros de 32 bits) (Página 2334)
Conversión explícita de UDINT (Página 2433)

Conversión implícita de LINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos LINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
LINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. LINT #-1 -> SINT #-1, o LINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	ULINT	-	X	
	REAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. LINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 8 388 608)
	LREAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. LINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 9 007 199 254 740 992)
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Tiempo en nanosegundos)
	S5TIME	-	-	Ninguna conversión implícita
	LDT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Nanosegundos desde el 1.1.1970)
	DTL	-	-	Ninguna conversión implícita

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	STRING	-	-	Ninguna conversión implícita
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

LINT (enteros de 64 bits) (Página 2335)

Conversión explícita de LINT (Página 2435)

Conversión implícita de ULINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos ULINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
ULINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. ULINT #-1 -> SINT #-1, o ULINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	REAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. LINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 8 388 608)
	LREAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. LINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 9 007 199 254 740 992)
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Tiempo en nanosegundos)
	S5TIME	-	-	Ninguna conversión implícita
	LDT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Nanosegundos desde el 1.1.1970)
	DTL	-	-	Ninguna conversión implícita
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	STRING	-	-	Ninguna conversión implícita
	WSTRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	WCHAR	-	X	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

ULINT (enteros de 64 bits) (Página 2336)

Conversión explícita de ULINT (Página 2438)

Números en coma flotante

Conversión implícita de REAL

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos REAL:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
REAL	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	LWORD	-	-	Ninguna conversión implícita
	SINT	-	X	El patrón de bits del valor de origen se redondea, se convierte y se transfiere al tipo de datos de destino. (Redondeo y conversión de valores de p. ej. REAL #2.5 -> INT #2, o números negativos REAL #-2.5 -> INT #-2 -> USINT #254. En caso de desbordamiento se determina el resto REAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	LREAL	X	X	El valor se transfiere al tipo de datos de destino.
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
REAL (Página 2338)
Conversión explícita de REAL (Página 2440)

Conversión implícita de LREAL

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos LREAL:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
LREAL	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	SINT	-	X	El patrón de bits del valor de origen se redondea, se convierte y se transfiere al tipo de datos de destino. (Redondeo y conversión de valores de p. ej. LREAL #2.5 -> INT #2, o números negativos LREAL #-2.5 -> INT #-2 -> USINT #254. En caso de desbordamiento se determina el resto LREAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	X	El valor se transfiere al tipo de datos de destino.
	TIME	-	-	Ninguna conversión implícita
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

LREAL (Página 2339)

Conversión explícita de LREAL (Página 2441)

Tiempos

Conversión implícita de S5TIME

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos S5TIME:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
S5TIME	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	DWORD	-	-	Ninguna conversión implícita
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	LTIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
S5TIME (tiempo) (Página 2341)
Conversión explícita de S5TIME (Página 2443)

Conversión implícita de TIME

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos TIME:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
TIME	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	LWORD	-	-	Ninguna conversión implícita
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	UDINT	-	X	
	LINT	-	-	Ninguna conversión implícita
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	S5TIME	-	-	
	LTIME	X	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en nanosegundos. (1 ms = 1 000 000 ns)
	LDT	-	-	Ninguna conversión implícita
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	X	Si el valor de origen está entre 0 y 84599.999 s, el patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. (Representación en nanosegundos). De lo contrario, el valor de destino no se modifica. El resultado de la conversión muestra el tiempo transcurrido desde medianoche.
	LTOD	-	-	Ninguna conversión implícita
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

TIME (tiempo IEC) (Página 2343)

Conversión explícita de TIME (Página 2444)

Conversión implícita de LTIME

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos LTIME:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
LTIME	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en nanosegundos.
	SINT	-	-	Ninguna conversión implícita
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en nanosegundos.
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	S5TIME	-	-	
	TIME	-	X	Si el valor de origen está fuera del rango de valores del tipo de datos de destino, el valor de destino no se cambia. (De 0.123456789 s se obtiene 0.123 s)
	LDT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en nanosegundos desde el 1.1.1970.
	DTL	-	-	Ninguna conversión implícita
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en nanosegundos desde el 1.1.1970.
	STRING	-	-	Ninguna conversión implícita
	WSTRING	-	-	
	CHAR	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

LTIME (temporizador CEI) (Página 2343)

Conversión explícita de LTIME (Página 2446)

Fecha y hora

Conversión implícita de DT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	LDT	X	X	El valor de origen se transfiere sin cambios, conservando su valor, al tipo de datos de destino. (El valor 24.12.2012 14:30 sigue siendo 24.12.2012 14:30)
	DTL	X	X	
	DATE	-	-	Ninguna conversión implícita
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

DT (DATE_AND_TIME) (Página 2346)

Conversión explícita de DT (Página 2447)

Conversión implícita de LDT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos LDT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
LDT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Nanosegundos desde el 1.1.1970)
	SINT	-	-	Ninguna conversión implícita
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	UDINT	-	-	
	LINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Nanosegundos desde el 1.1.1970)
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. (Nanosegundos desde el 1.1.1970)
	S5TIME	-	-	Ninguna conversión implícita
	DT	-	X	El patrón de bits del valor de origen se convierte al tipo de datos de destino con una pérdida de precisión. (De 24.12.2012 12:34:56.123456789 se obtiene 24.12.2012 12:34:56.123)
	DTL	X	X	El valor de origen se transfiere sin cambios, conservando su valor, al tipo de datos de destino. (El valor 24.12.2012 14:30 sigue siendo 24.12.2012 14:30)
	DATE	-	-	Ninguna conversión implícita
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión explícita de LDT (Página 2449)

LDT (DATE_AND_LTIME) (Página 2347)

Conversión implícita de DTL

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DTL:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DTL	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	X	X	El valor de origen se transfiere sin cambios, conservando su valor, al tipo de datos de destino. (El valor 24.12.2012 14:30 sigue siendo 24.12.2012 14:30)
	DT	-	X	El patrón de bits del valor de origen se convierte al tipo de datos de destino con una pérdida de precisión. (De 24.12.2012 12:34:56.123456789 se obtiene 24.12.2012 12:34:56.123)
	DATE	-	-	Ninguna conversión implícita
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

DTL (Página 2348)

Conversión explícita de DTL (Página 2450)

Conversión implícita de DATE

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DATE:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DATE	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión corresponde al número de días desde el 01.01.1990.
	DWORD	-	-	Ninguna conversión implícita
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión corresponde al número de días desde el 01.01.1990.
	UINT	-	X	
	DINT	-	-	Ninguna conversión implícita
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DT	-	-	
	DTL	-	-	
	TOD	-	-	
	LTOD	-	-	
	STRING	-	-	
	WSTRING	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

DATE (Página 2344)

Conversión explícita de DATE (Página 2452)

Conversión implícita de TOD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos TOD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
TOD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).
	LWORD	-	-	Ninguna conversión implícita
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).
	UDINT	-	X	
	LINT	-	-	Ninguna conversión implícita
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	LTIME	-	-	Ninguna conversión implícita
	S5TIME	-	-	
	LDT	-	-	
	DT	-	-	
	DTL	-	-	
	DATE	-	-	
	LTOD	X	X	El valor de origen se transfiere sin cambios, conservando su valor, al tipo de datos de destino. (De 12:34:56.123 se obtiene 12:34:56.123000000)
	STRING	-	-	Ninguna conversión implícita
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

TOD (TIME_OF_DAY) (Página 2345)

Conversión explícita de TOD (Página 2453)

Conversión implícita de LTOD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos LTOD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
LTOD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de nanosegundos desde el inicio del día (0:00 horas).
	SINT	-	-	Ninguna conversión implícita

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de nanosegundos desde el inicio del día (0:00 horas).
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de nanosegundos desde el inicio del día (0:00 horas).
	S5TIME	-	-	Ninguna conversión implícita
	LDT	-	-	
	DT	-	-	
	DTL	-	-	
	DATE	-	-	
	TOD	-	X	El valor de origen se transfiere sin cambios, conservando su valor redondeado, al tipo de datos de destino. (De 12:34:56.123456789 se obtiene 12:34:56.123)
	STRING	-	-	Ninguna conversión implícita
	WSTRING	-	-	
	CHAR	-	-	
	WCHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión explícita de LTOD (Página 2454)

LTOD (LTIME_OF_DAY) (Página 2345)

Cadenas de caracteres

Conversión implícita de CHAR

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos CHAR:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
CHAR	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0" desde la izquierda.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	WCHAR	-	-	
	STRING	X	X	El STRING se acorta hasta la longitud 1 y contiene el carácter.
	WSTRING	-	-	Ninguna conversión implícita
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)
Sinopsis de la conversión de tipos de datos (Página 2373)
CHAR (carácter) (Página 2349)
Conversión explícita de CHAR (Página 2456)

Conversión implícita de WCHAR

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos WCHAR:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
WCHAR	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0" desde la izquierda.
	WORD	-	X	
	DWORD	-	X	
	LWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LINT	-	X	
	ULINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	CHAR	-	-	
	STRING	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	WSTRING	X	X	El WSTRING se acorta hasta la longitud 1 y contiene el carácter.
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

WCHAR (carácter) (Página 2349)

Conversión explícita de WCHAR (Página 2457)

Conversión implícita de STRING

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos STRING:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
STRING	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	CHAR	-	X	Se devuelve el primer carácter del STRING si el STRING contiene uno o más caracteres. De lo contrario, el carácter se muestra con la codificación $00.
	WCHAR	-	-	Ninguna conversión implícita
	WSTRING	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

STRING (Página 2350)

Conversión explícita de STRING (Página 2458)

Conversión implícita de WSTRING

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos WSTRING:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
WSTRING	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	LWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	LINT	-	-	

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
	ULINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	LTIME	-	-	
	S5TIME	-	-	
	LDT	-	-	
	DTL	-	-	
	DT	-	-	
	DATE	-	-	
	TOD	-	-	
	LTOD	-	-	
	CHAR	-	-	
	WCHAR	-	X	Se devuelve el primer carácter del WSTRING si el WSTRING contiene uno o varios caracteres. De lo contrario, el carácter se muestra con la codificación $0000.
	STRING	-	-	Ninguna conversión implícita
x: Conversión posible -: Conversión no posible				

Consulte también

Activar o desactivar la verificación IEC (Página 2375)

Sinopsis de la conversión de tipos de datos (Página 2373)

WSTRING (Página 2351)

Conversión explícita de WSTRING (Página 2460)

Conversión explícita

Números binarios

Conversión explícita de BOOL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos BOOL y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
BOOL	BYTE	X	En el tipo de datos de destino solamente se activa el LSB (Least Significant Bit). La salida de habilitación ENO es siempre "1".	BOOL_TO_BYTE
	WORD	X		BOOL_TO_WORD
	DWORD	X		BOOL_TO_DWORD
	LWORD	X		BOOL_TO_LWORD
	SINT	X		BOOL_TO_SINT
	USINT	X		BOOL_TO_USINT
	INT	X		BOOL_TO_INT
	UINT	X		BOOL_TO_UINT
	DINT	X		BOOL_TO_DINT
	UDINT	X		BOOL_TO_UDINT
	LINT	X		BOOL_TO_LINT
	ULINT	X		BOOL_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	TIME	-		-
	LTIME	-		-
	S5TIME	-		-
	LDT	-		-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Conversión posible - : Conversión no posible				

Consulte también

Sinopsis de la conversión de tipos de datos (Página 2373)
Conversión implícita de BOOL (Página 2376)
BOOL (bit) (Página 2325)

Secuencias de bits

Conversión explícita de BYTE

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos BYTE y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
BYTE[1)]	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	BYTE_TO_BOOL
	WORD[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	BYTE_TO_WORD
	DWORD[1)]	X		BYTE_TO_DWORD
	LWORD[1)]	X		BYTE_TO_LWORD
	SINT	X		BYTE_TO_SINT
	USINT	X		BYTE_TO_USINT
	INT	X		BYTE_TO_INT
	UINT	X		BYTE_TO_UINT
	DINT	X		BYTE_TO_DINT
	UDINT	X		BYTE_TO_UDINT
	LINT	X		BYTE_TO_LINT
	ULINT	X		BYTE_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	BYTE_TO_TIME
	LTIME	X		BYTE_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	BYTE_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	BYTE_TO_TOD
	LTOD	X		BYTE_TO_LTOD
	DATE	X		BYTE_TO_DATE
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	BYTE_TO_CHAR
	WCHAR	X		BYTE_TO_WCHAR

x: conversión posible

- : conversión no posible

[1] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta en este caso como USINT; WORD, como UINT; DWORD, como UDINT y LWORD, como ULINT.

Consulte también

Sinopsis de la conversión de tipos de datos (Página 2373)

Conversión implícita de BYTE (Página 2377)

BYTE (bytes) (Página 2326)

Conversión explícita de WORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos WORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
WORD[1)]	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	WORD_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WORD_TO_BYTE
	DWORD[1)]	X		WORD_TO_DWORD
	LWORD[1)]	X		WORD_TO_LWORD

	SINT	X	ENO = TRUE
			#sint1 := WORD_TO_SINT(16#FFFF); // -1 a
			#sint1 := WORD_TO_SINT(16#FF80); // -128
			#sint1 := WORD_TO_SINT(16#0); // 0 a
			#sint1 := WORD_TO_SINT(16#007F); // 127
			
			ENO = FALSE
			#sint1 := WORD_TO_SINT(16#FF7F); // -129 a
			#sint1 := WORD_TO_SINT(16#8000); // -32768
			#sint1 := WORD_TO_SINT(16#0080); // 128 a
			#sint1 := WORD_TO_SINT(16#7FFF); // 32767
				WORD_TO_SINT

	USINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WORD_TO_USINT
	INT	X		WORD_TO_INT
	UINT	X		WORD_TO_UINT
	DINT	X		WORD_TO_DINT
	UDINT	X		WORD_TO_UDINT
	LINT	X		WORD_TO_LINT
	ULINT	X		WORD_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WORD_TO_TIME
	LTIME	X		WORD_TO_LTIME
	S5TIME	X		WORD_TO_S5TIME
	LDT	X		WORD_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WORD_TO_TOD
	LTOD	X		WORD_TO_LTOD
	DATE	X		WORD_TO_DATE
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WORD_TO_CHAR
	WCHAR	X		WORD_TO_WCHAR
WORD_ BCD16	INT	X	El valor que debe convertirse tiene el tipo de datos WORD y se aplica como valor codificado en BCD entre -999 y +999. Tras la conversión, el resultado es un número entero (representación binaria) del tipo INT. Se produce una conversión auténtica. Si el patrón de bits incluye una tétrada no válida, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	WORD_BCD16_TO_INT
BCD16	INT	X		BCD16_TO_INT

x: conversión posible

- : conversión no posible

[1] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta en este caso como USINT; WORD, como UINT; DWORD, como UDINT y LWORD, como ULINT.

Consulte también

Conversión implícita de WORD (Página 2378)

Sinopsis de la conversión de tipos de datos (Página 2373)

WORD (Página 2327)

Conversión explícita de DWORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DWORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DWORD[1)]	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	DWORD_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DWORD_TO_BYTE
	WORD[1)]	X		DWORD_TO_WORD
	LWORD[1)]	X		DWORD_TO_LWORD

	SINT	X	ENO = TRUE
			#sint1 := DWORD_TO_SINT(16#FFFF_FFFF); // -1 a
			#sint1 := DWORD_TO_SINT(16#FFFF_FF80); // -128
			#sint1 := DWORD_TO_SINT(16#0); // 0 a
			#sint1 := DWORD_TO_SINT(16#0000_007F); // 127
			
			ENO = FALSE
			#sint1 := DWORD_TO_SINT(16#FFFF_FF7F); // -129
			#sint1 := DWORD_TO_SINT(16#8000_0000); // -2147483648
			#sint1 := DWORD_TO_SINT(16#0000_0080); // 128 a
			#sint1 := DWORD_TO_SINT(16#7FFF_FFFF); // 2147483647
				DWORD_TO_SINT

	USINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DWORD_TO_USINT

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción

	INT	X	ENO = TRUE
			#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 a
			#int1 := DWORD_TO_INT(16#FFFF_8000); // -32768
			#int1 := DWORD_TO_INT(16#0); // 0 a
			#int1 := DWORD_TO_INT(16#0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769
			#int1 := DWORD_TO_INT(16#8000_0000); // -2147483648
			#int1 := DWORD_TO_INT(16#8000); // 32768 a
			#int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647
				DWORD_TO_INT

	UINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si no se producen errores durante la conversión, el estado lógico de ENO = 1; si se produce un error durante la ejecución, el estado lógico de ENO = 0.	DWORD_TO_UINT
	DINT	X		DWORD_TO_DINT
	UDINT	X		DWORD_TO_UDINT
	LINT	X		DWORD_TO_LINT
	ULINT	X		DWORD_TO_ULINT
	REAL	X		DWORD_TO_REAL
	LREAL	-	Ninguna conversión explícita	-
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DWORD_TO_TIME
	LTIME	X		DWORD_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DWORD_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DWORD_TO_TOD
	LTOD	X		DWORD_TO_LTOD
	DATE	X		DWORD_TO_DATE
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DWORD_TO_CHAR

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WCHAR	X		DWORD_TO_WCHAR
DWORD_BCD32	DINT	X	El valor que debe convertirse tiene el tipo de datos DWORD y se aplica como valor codificado en BCD entre -9999999 y +9999999. Tras la conversión, el resultado es un número entero (representación binaria) del tipo DINT. Se produce una conversión auténtica. Si el patrón de bits incluye una tétrada no válida, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	DWORD_BCD32_TO_DINT
BCD32	DINT	X		BCD32_TO_DINT
x: conversión posible - : conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta en este caso como USINT; WORD, como UINT; DWORD, como UDINT y LWORD, como ULINT.				

Consulte también

Conversión implícita de DWORD (Página 2379)

Sinopsis de la conversión de tipos de datos (Página 2373)

DWORD (Página 2327)

Conversión explícita de LWORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LWORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
LWORD[1]	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	LWORD_TO_BOOL
	BYTE[1]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_BYTE
	WORD[1]	X		LWORD_TO_WORD
	DWORD[1]	X		LWORD_TO_DWORD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción

	SINT	X	ENO = TRUE
			#sint1 := LWORD_TO_SINT(16#FFFF_FFFF_FFFF_FFFF); // -1 a
			#sint1 := LWORD_TO_SINT(16#FFFF_FFFF_FFFF_FF80); // -128
			#sint1 := LWORD_TO_SINT(16#0); // 0 a
			#sint1 := LWORD_TO_SINT(16#0000_0000_0000_007F); // 127
			
			ENO = FALSE
			#sint1 := LWORD_TO_SINT(16#FFFF_FFFF_FFFF_FF7F); // -129
			#sint1 := LWORD_TO_SINT(16#8000_0000_0000_0000); // -9223372036854775808
			#sint1 := LWORD_TO_SINT(16#0000_0000_0000_0080); // 128
			#sint1 := LWORD_TO_SINT(16#7FFF_FFFF_FFFF_FFFF); // 9223372036854775807
				LWORD_TO_SINT

	USINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_USINT

	INT	X	ENO = TRUE
			#int1 := LWORD_TO_INT(16#FFFF_FFFF_FFFF_FFFF); // -1 a
			#int1 := LWORD_TO_INT(16#FFFF_FFFF_FFFF_8000); // -32768
			#int1 := LWORD_TO_INT(16#0); // 0 a
			#int1 := LWORD_TO_INT(16#0000_0000_0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := LWORD_TO_INT(16#FFFF_FFFF_FFFF_7FFF); // -32769
			#int1 := LWORD_TO_INT(16#8000_0000_0000_0000); // -2147483648
			#int1 := LWORD_TO_INT(16#0000_0000_0000_8000); // 32768 a
			#int1 := LWORD_TO_INT(16#7FFF_FFFF_FFFF_FFFF); // 2147483647
				LWORD_TO_INT

	UINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_UINT

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción

	DINT	X	ENO = TRUE
			#dint1 := LWORD_TO_DINT(16#FFFF_FFFF_FFFF_FFFF); // -1 a
			#dint1 := LWORD_TO_DINT(16#FFFF_FFFF_8000_0000); // -2147483648
			#dint1 := LWORD_TO_DINT(16#0); // 0 a
			#dint1 := LWORD_TO_DINT(16#0000_0000_7FFF_FFFF); // 2147483647
			
			ENO = FALSE
			#dint1 := LWORD_TO_DINT(16#FFFF_FFFF_7FFF_FFFF); // -2147483649 a
			#dint1 := LWORD_TO_DINT(16#8000_0000_0000_0000); // -9223372036854775808
			#dint1 := LWORD_TO_DINT(16#0000_0000_8000_0000); // 2147483648 a
			#dint1 := LWORD_TO_DINT(16#7FFF_FFFF_FFFF_FFFF); // 9223372036854775807
				LWORD_TO_DINT

	UDINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_UDINT
	LINT	X		LWORD_TO_LINT
	ULINT	X		LWORD_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si no se producen errores durante la conversión, el estado lógico de ENO = 1; si se produce un error durante la ejecución, el estado lógico de ENO = 0.	LWORD_TO_LREAL
	TIME	X		LWORD_TO_TIME
	LTIME	X		LWORD_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_TOD
	LTOD	X		LWORD_TO_LTOD
	DATE	X		LWORD_TO_DATE
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LWORD_TO_CHAR

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WCHAR	X		LWORD_TO_WCHAR
x: conversión posible - : conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta en este caso como USINT; WORD, como UINT; DWORD, como UDINT y LWORD, como ULINT.				

Consulte también

Conversión implícita de LWORD (Página 2380)

Sinopsis de la conversión de tipos de datos (Página 2373)

LWORD (Página 2328)

Enteros

Conversión explícita de SINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos SINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
SINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	SINT_TO_BOOL
	BYTE1)	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si se convierte un valor negativo a un tipo de datos sin	SINT_TO_BYTE
	WORD1)	X		SINT_TO_WORD
	DWORD1)	X		SINT_TO_DWORD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LWORD[1)]	X	signo, la salida de habilitación ENO se pone a "0".	SINT_TO_LWORD
	USINT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0".	SINT_TO_USINT
	INT	X		SINT_TO_INT
	UINT	X		SINT_TO_UINT
	DINT	X		SINT_TO_DINT
	UDINT	X		SINT_TO_UDINT
	LINT	X		SINT_TO_LINT
	ULINT	X		SINT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "-1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "-1.0".	SINT_TO_REAL, NORM_X
	LREAL	X		SINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	SINT_TO_TIME
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	SINT_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	SINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0)	SINT_TO_TOD
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en nanosegundos desde 0:0)	SINT_TO_LTOD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1)	SINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa con un signo al principio. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	SINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		SINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0".	SINT_TO_CHAR
	WCHAR[1)]	X		SINT_TO_WCHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de SINT (Página 2382)

Sinopsis de la conversión de tipos de datos (Página 2373)

SINT (enteros de 8 bits) (Página 2330)

Conversión explícita de USINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos USINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
USINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	USINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	USINT_TO_BYTE
	WORD[1)]	X		USINT_TO_WORD
	DWORD[1)]	X		USINT_TO_DWORD
	LWORD[1)]	X		USINT_TO_LWORD
	SINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si al convertir se cambia el signo, la salida de habilitación ENO se pone a "0".	USINT_TO_SINT
	INT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir.	USINT_TO_INT
	UINT	X		USINT_TO_UINT
	DINT	X		USINT_TO_DINT
	UDINT	X		USINT_TO_UDINT
	LINT	X		USINT_TO_LINT
	ULINT	X		USINT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "1.0".	USINT_TO_REAL, NORM_X
	LREAL	X		USINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	USINT_TO_TIME

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	USINT_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	USINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en milisegundos desde 0:0)	USINT_TO_TOD
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en nanosegundos desde 0:0)	USINT_TO_LTOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en días desde 1990-1-1)	USINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	USINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		USINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir.	USINT_TO_CHAR
	WCHAR[1)]	X		USINT_TO_WCHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de USINT (Página 2383)

Sinopsis de la conversión de tipos de datos (Página 2373)

USINT (enteros de 8 bits) (Página 2331)

Conversión explícita de INT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos INT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
INT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	INT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	INT_TO_BYTE
	WORD[1)]	X		INT_TO_WORD
	DWORD[1)]	X		INT_TO_DWORD
	LWORD[1)]	X		INT_TO_LWORD
	SINT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	INT_TO_SINT
	USINT	X		INT_TO_USINT
	UINT	X		INT_TO_UINT
	DINT	X		INT_TO_DINT
	UDINT	X		INT_TO_UDINT
	LINT	X		INT_TO_LINT
	ULINT	X		INT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "-1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "-1.0".	INT_TO_REAL, NORM_X
	LREAL	X		INT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	INT_TO_TIME
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	INT_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	INT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0; verificación de límite de 24 h)	INT_TO_TOD
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en nanosegundos desde 0:0; verificación de límite de 24 h)	INT_TO_LTOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1; verificación de valor negativo)	INT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa con un signo al principio. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	INT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		INT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	INT_TO_CHAR
	WCHAR[1)]	X		INT_TO_WCHAR
	BCD16	X	El valor que debe convertirse tiene el tipo INT y se aplica como número entero con un valor entre -999 y +999. Tras la conversión, el resultado es un	INT_TO_BCD16

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	BCD16_WORD	X	número codificado en BCD del tipo WORD. Se produce una conversión auténtica. Si el valor está fuera del rango de destino, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	INT_TO_BCD16_WORD
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

Conversión implícita de INT (Página 2384)

Sinopsis de la conversión de tipos de datos (Página 2373)

INT (enteros de 16 bits) (Página 2331)

Conversión explícita de UINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos UINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
UINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	UINT_TO_BOOL
	BYTE1)	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si en este proceso se	UINT_TO_BYTE
	WORD1)	X		UINT_TO_WORD
	DWORD1)	X		UINT_TO_DWORD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LWORD[1)]	X	pierden bits, la salida de habilitación ENO se pone a "0".	UINT_TO_LWORD
	SINT	X		UINT_TO_SINT
	USINT	X		UINT_TO_USINT
	INT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si al convertir se cambia el bit de signo, la salida de habilitación ENO se pone a "0".	UINT_TO_INT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	UINT_TO_DINT
	UDINT	X		UINT_TO_UDINT
	LINT	X		UINT_TO_LINT
	ULINT	X		UINT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "1.0".	UINT_TO_REAL, NORM_X
	LREAL	X		UINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	UINT_TO_TIME
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	UINT_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	UINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en milisegundos desde 0:0; verificación de límite de 24 h)	UINT_TO_TOD
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en nanosegundos desde 0:0; verificación de límite de 24 h)	UINT_TO_LTOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en días desde 1990-1-1; verificación de valor negativo)	UINT_TO_DATE, T_CONV
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la	UINT_TO_STRING, S_CONV, VAL_STRG

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WSTRING	X	longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	UINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	UINT_TO_CHAR
	WCHAR[1)]	X		UINT_TO_WCHAR

x: conversión posible

- : conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de UINT (Página 2385)

Sinopsis de la conversión de tipos de datos (Página 2373)

UINT (enteros de 16 bits) (Página 2332)

Conversión explícita de DINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	DINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de	DINT_TO_BYTE
	WORD[1)]	X		DINT_TO_WORD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DWORD[1)]	X	destino. Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	DINT_TO_DWORD
	LWORD[1)]	X		DINT_TO_LWORD
	SINT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	DINT_TO_SINT
	USINT	X		DINT_TO_USINT
	INT	X		DINT_TO_INT
	UINT	X		DINT_TO_UINT
	UDINT	X		DINT_TO_UDINT
	LINT	X		DINT_TO_LINT
	ULINT	X		DINT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "-1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "-1.0".	DINT_TO_REAL, NORM_X
	LREAL	X		DINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	DINT_TO_TIME, T_CONV
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	DINT_TO_LTIME, T_CONV
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	DINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0)	DINT_TO_TOD
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en nanosegundos desde 0:0)	DINT_TO_LTOD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1)	DINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa con un signo al principio. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	DINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		DINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	DINT_TO_CHAR
	WCHAR[1)]	X		DINT_TO_WCHAR
	BCD32	X	El valor que debe convertirse tiene el tipo DINT y se aplica como número entero con un valor entre –999999 y +9999999. Tras la conversión, el resultado es un número codificado en BCD del tipo DWORD. Si se produce un desbordamiento, la salida de habilitación se pone a "0". Se produce una conversión auténtica. Si el valor está fuera del rango de destino, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	DINT_TO_BCD32
	BCD32_DWORD	X		DINT_TO_BCD32_DWORD

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de DINT (Página 2386)

Sinopsis de la conversión de tipos de datos (Página 2373)

DINT (enteros de 32 bits) (Página 2333)

Conversión explícita de UDINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos UDINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
UDINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	UDINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si en este proceso se pierden bits, la salida de habilitación ENO se pone a "0".	UDINT_TO_BYTE
	WORD[1)]	X		UDINT_TO_WORD
	DWORD[1)]	X		UDINT_TO_DWORD
	LWORD[1)]	X		UDINT_TO_LWORD
	SINT	X		UDINT_TO_SINT
	USINT	X		UDINT_TO_USINT
	INT	X		UDINT_TO_INT
	UINT	X		UDINT_TO_UINT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si al convertir se cambia el bit de signo, la salida de habilitación ENO se pone a "0".	UDINT_TO_DINT
	LINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	UDINT_TO_LINT
	ULINT	X		UDINT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "1.0".	UDINT_TO_REAL, NORM_X
	LREAL	X		UDINT_TO_LREAL, NORM_X

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	UDINT_TO_TIME
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	UDINT_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	UDINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0; verificación de límite de 24 h)	UDINT_TO_TOD, T_CONV
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en nanosegundos desde 0:0; verificación de límite de 24 h)	UDNT_TO_LTOD, T_CONV
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1; verificación de valor negativo)	UDINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	UDINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		UDINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	UDINT_TO_CHAR

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WCHAR[1)]	X		UDINT_TO_WCHAR
x: Conversión posible - : Conversión no posible [1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

Conversión implícita de UDINT (Página 2388)

Sinopsis de la conversión de tipos de datos (Página 2373)

UDINT (enteros de 32 bits) (Página 2334)

Conversión explícita de LINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
LINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	LINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	LINT_TO_BYTE
	WORD[1)]	X		LINT_TO_WORD
	DWORD[1)]	X		LINT_TO_DWORD
	LWORD[1)]	X		LINT_TO_LWORD
	SINT	X		LINT_TO_SINT
	USINT	X		LINT_TO_USINT
	INT	X		LINT_TO_INT

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	UINT	X		LINT_TO_UINT
	DINT	X		LINT_TO_DINT
	UDINT	X		LINT_TO_UDINT
	ULINT	X		LINT_TO_ULINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "-1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "-1.0".	LINT_TO_REAL, NORM_X
	LREAL	X		LINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	LINT_TO_TIME, T_CONV
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	LINT_TO_LTIME, T_CONV
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	LINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0)	LINT_TO_TOD
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en nanosegundos desde 0:0)	LINT_TO_LTOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1)	LINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa con un	LINT_TO_STRING, S_CONV, VAL_STRG

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WSTRING	X	signo al principio. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	LINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF)). Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	LINT_TO_CHAR
	WCHAR[1)]	X		LINT_TO_WCHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de LINT (Página 2389)

Sinopsis de la conversión de tipos de datos (Página 2373)

LINT (enteros de 64 bits) (Página 2335)

Conversión explícita de ULINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos ULINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
ULINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	ULINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	ULINT_TO_BYTE
	WORD[1)]	X		ULINT_TO_WORD
	DWORD[1)]	X		ULINT_TO_DWORD
	LWORD[1)]	X		ULINT_TO_LWORD
	SINT	X		ULINT_TO_SINT
	USINT	X		ULINT_TO_USINT
	INT	X		ULINT_TO_INT
	UINT	X		ULINT_TO_UINT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si al convertir se sobrescribe el bit de signo, la salida de habilitación ENO se pone a "0".	ULINT_TO_DINT
	UDINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	ULINT_TO_UDINT
	LINT	X		ULINT_TO_LINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino. El valor "1", p. ej., con la instrucción "Convertir valor" (CONVERT) se transforma en el valor "1.0".	ULINT_TO_REAL, NORM_X
	LREAL	X		ULINT_TO_LREAL, NORM_X

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	ULINT_TO_TIME
	LTIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como nanosegundos.	ULINT_TO_LTIME
	S5TIME	-	Ninguna conversión explícita	-
	LDT	X	El resultado se suministra en nanosegundos desde 1970-1-1-0:0:0.0.	ULINT_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en milisegundos desde 0:0)	ULINT_TO_TOD, T_CONV
	LTOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en nanosegundos desde 0:0)	ULINT_TO_LTOD, T_CONV
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Interpretación en días desde 1990-1-1)	ULINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	ULINT_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		ULINT_TO_WSTRING
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir.	ULINT_TO_CHAR
	WCHAR[1)]	X		ULINT_TO_WCHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de ULINT (Página 2390)

Sinopsis de la conversión de tipos de datos (Página 2373)

ULINT (enteros de 64 bits) (Página 2336)

Números en coma flotante

Conversión explícita de REAL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos REAL y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
REAL	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	REAL_TO_DWORD
	LWORD	-	Ninguna conversión explícita	-
	SINT	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada. Si durante la conversión se rebasa el rango permitido de valores del tipo de datos de destino o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0".	REAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		REAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		REAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		REAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		REAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		REAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LINT	X		REAL_TO_LINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	ULINT	X		REAL_TO_ULINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LREAL	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada, p. ej., TRUNC(2.5) = 2.0; CEIL(2.5) = 3.0	REAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Ninguna conversión explícita	-
	LTIME	-		-
	S5TIME	-		-
	LDT	-		-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DATE	-		-
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0". La longitud mín. de la cadena de caracteres es de 14 caracteres.	REAL_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		REAL_TO_WSTRING
	CHAR	-	Ninguna conversión explícita	-
	WCHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

Conversión implícita de REAL (Página 2392)

Sinopsis de la conversión de tipos de datos (Página 2373)

REAL (Página 2338)

Conversión explícita de LREAL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LREAL y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
LREAL	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LREAL_TO_LWORD
	SINT	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada. Si durante la conversión se rebasa el rango permitido de valores del tipo de datos de destino o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0".	LREAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		LREAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		LREAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		LREAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DINT	X		LREAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		LREAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LINT	X		LREAL_TO_LINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	ULINT	X		LREAL_TO_ULINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	REAL	X	El valor se convierte al tipo de datos de destino. Si al convertir se rebasa el rango de valores permitido o el valor que se desea convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0". Se tolera la pérdida de precisión.	LREAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Ninguna conversión explícita	-
	LTIME	-		-
	S5TIME	-		-
	LDT	-		-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0". La longitud mín. de la cadena de caracteres es de 21 caracteres.	LREAL_TO_STRING, S_CONV, VAL_STRG
	WSTRING	X		LREAL_TO_WSTRING
	CHAR	-	Ninguna conversión explícita	-
	WCHAR	-		-
x: Conversión posible - : Conversión no posible				

Consulte también

Conversión implícita de LREAL (Página 2393)

Sinopsis de la conversión de tipos de datos (Página 2373)

LREAL (Página 2339)

Tiempos

Conversión explícita de S5TIME

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos S5TIME y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
S5TIME	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	S5TIME_TO_WORD
	DWORD	-	Ninguna conversión explícita	-
	LWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	LINT	-		-
	ULINT	-		-
	REAL	-		-
	LREAL	-		-
	TIME	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de s5t#10ms se obtiene T#10ms)	S5TIME_TO_TIME
	LTIME	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de s5t#10ms se obtiene LTIME#10ms)	S5TIME_TO_LTIME
	LDT	-	Ninguna conversión explícita	-
	DT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WCHAR	-		-
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

Conversión implícita de S5TIME (Página 2394)

Sinopsis de la conversión de tipos de datos (Página 2373)

S5TIME (tiempo) (Página 2341)

Conversión explícita de TIME

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos TIME y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
TIME	BOOL	-	Ninguna conversión explícita	-
	BYTE1)	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	TIME_TO_BYTE
	WORD1)	X		TIME_TO_WORD
	DWORD1)	X		TIME_TO_DWORD
	LWORD1)	X		TIME_TO_LWORD
	SINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha, interpretado como milisegundos, al tipo de datos de destino.	TIME_TO_SINT
	USINT	X		TIME_TO_USINT
	INT	X		TIME_TO_INT
	UINT	X		TIME_TO_UINT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.	TIME_TO_DINT, T_CONV
	UDINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha, interpretado como milisegundos, al tipo de datos de destino. El cambio de signo da lugar a que la salida de habilitación ENO sea "0".	TIME_TO_UDINT
	LINT	X		TIME_TO_LINT
	ULINT	X		TIME_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	S5TIME	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de T#10ms se obtiene s5t#10ms)	TIME_TO_TIME
	LTIME	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de T#10ms se obtiene LTIME#10ms)	TIME_TO_LTIME
	LDT	-	Ninguna conversión explícita	-
	DT	-		-
	DTL	-		-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si el valor de origen está fuera del rango de valores de TOD, el tipo de datos de destino no se cambia.	TIME_TO_TOD
	LTOD	-	Ninguna conversión explícita	-
	DATE	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Conversión posible

- : Conversión no posible

[1] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de TIME (Página 2395)

Sinopsis de la conversión de tipos de datos (Página 2373)

TIME (tiempo IEC) (Página 2343)

Conversión explícita de LTIME

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LTIME y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
LTIME	BOOL	-	Ninguna conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LTIME_TO_BYTE
	WORD[1)]	X		LTIME_TO_WORD
	DWORD[1)]	X		LTIME_TO_DWORD
	LWORD[1)]	X		LTIME_TO_LWORD
	SINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha, interpretado como nanosegundos, al tipo de datos de destino. El cambio de signo da lugar a que la salida de habilitación ENO sea "0".	LTIME_TO_SINT
	USINT	X		LTIME_TO_USINT
	INT	X		LTIME_TO_INT
	UINT	X		LTIME_TO_UINT
	DINT	X		LTIME_TO_DINT
	UDINT	X		LTIME_TO_UDINT
	LINT	X		LTIME_TO_LINT, T_CONV
	ULINT	X		LTIME_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTIME#10ms se obtiene s5t#10ms)	LTIME_TO_S5TIME
	TIME	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTIMET#10ms se obtiene T#10ms)	LTIME_TO_TIME
	LDT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTIMET#10ms se obtiene LDT#1970-1-1-0:0:0.010000000)	LTIME_TO_LDT
	DT	-	Ninguna conversión explícita	-
	DTL	-		-
	TOD	-		-
	LTOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si el valor de origen está fuera del rango de valores de LTOD, el tipo de datos de destino no se cambia.	LTIME_TO_LTOD
	DATE	-	Ninguna conversión explícita	-
	STRING	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Conversión posible

- : Conversión no posible

[1] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de LTIME (Página 2397)

Sinopsis de la conversión de tipos de datos (Página 2373)

LTIME (temporizador CEI) (Página 2343)

Fecha y hora

Conversión explícita de DT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DT	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	LINT	-		-
	ULINT	-		-
	REAL	-		-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LDT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de DT#1990-1-2-0:0:1.0 se obtiene LDT#1990-1-2-0:0:1.0)	DT_TO_LDT
	DTL	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de DT#1990-1-2-0:0:1.0 se obtiene DTL#1990-1-2-0:0:1.0)	DT_TO_DTL
	TOD	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de DT#1990-1-2-0:0:1.0 se obtiene TOD#1990-1-2-0:0:1.0)	DT_TO_TOD
	LTOD	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de DT#1990-1-2-0:0:1.0 se obtiene LTOD#1990-1-2-0:0:1.0)	DT_TO_LTOD
	DATE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DT_TO_DATE
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: conversión posible - : conversión no posible				

Consulte también

Conversión implícita de DT (Página 2398)

Sinopsis de la conversión de tipos de datos (Página 2373)

DT (DATE_AND_TIME) (Página 2346)

Conversión explícita de LDT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LDT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
LDT	BOOL	-	Ninguna conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LDT_TO_BYTE
	WORD[1)]	X		LDT_TO_WORD
	DWORD[1)]	X		LDT_TO_DWORD
	LWORD[1)]	X		LDT_TO_LWORD
	SINT	X		LDT_TO_SINT
	USINT	X		LDT_TO_USINT
	INT	X		LDT_TO_INT
	UINT	X		LDT_TO_UINT
	DINT	X		LDT_TO_DINT
	UDINT	X		LDT_TO_UDINT
	LINT	X		LDT_TO_LINT
	ULINT	X		LDT_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LDT_TO_LTIME
	DT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LDT#1990-1-2-0:0:1.0 se obtiene DT#1990-1-2-0:0:1.0)	LDT_TO_DT
	DTL	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LDT#1990-1-2-0:0:1.0 se obtiene DTL#1990-1-2-0:0:1.0)	LDT_TO_DTL
	TOD	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LDT#1990-1-2-0:0:1.0 se obtiene TOD#1990-1-2-0:0:1.0)	LDT_TO_TOD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LTOD	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LDT#1990-1-2-0:0:1.0 se obtiene LTOD#1990-1-2-0:0:1.0)	LDT_TO_LTOD
	DATE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LDT_TO_DATE
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

Conversión implícita de LDT (Página 2399)

Sinopsis de la conversión de tipos de datos (Página 2373)

LDT (DATE_AND_LTIME) (Página 2347)

Conversión explícita de DTL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DTL y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DTL	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	LINT	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	ULINT	-		-
	REAL	-		-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	LDT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de DTL#1990-1-2-0:0:1.0 se obtiene LDT#1990-1-2-0:0:1.0)	DTL_TO_LDT
	DT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de DTL#1990-1-2-0:0:1.0 se obtiene DT#1990-1-2-0:0:1.0)	DTL_TO_DT
	TOD	X	Durante la conversión se extrae la hora del día del formato DTL y se escribe en el tipo de datos de destino.	DTL_TO_TOD, T_CONV
	LTOD	X		DTL_TO_LTOD
	DATE	X	Durante la conversión se extrae la fecha del formato DTL y se escribe en el tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	DTL_TO_DATE, T_CONV
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Conversión posible - : Conversión no posible				

Consulte también

Conversión implícita de DTL (Página 2401)

Sinopsis de la conversión de tipos de datos (Página 2373)

DTL (Página 2348)

Conversión explícita de DATE

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DATE y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DATE	BOOL	-	Ninguna conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	DATE_TO_BYTE
	WORD[1)]	X		DATE_TO_WORD
	DWORD[1)]	X		DATE_TO_DWORD
	LWORD[1)]	X		DATE_TO_LWORD
	SINT	X	Como resultado se muestra el número de días desde el 1.1.1990.	DATE_TO_SINT
	USINT	X		DATE_TO_USINT
	INT	X		DATE_TO_INT
	UINT	X		DATE_TO_UINT
	DINT	X		DATE_TO_DINT
	UDINT	X		DATE_TO_UDINT
	LINT	X		DATE_TO_LINT
	ULINT	X		DATE_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	DT	X	La conversión ajusta la fecha en el tipo de destino.	DATE_TO_DT
	LDT	X		DATE_TO_LDT
	DTL	X		DATE_TO_DTL
	TOD	-	Ninguna conversión explícita	-
	LTOD	-		-
	STRING	-		-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de DATE (Página 2402)
Sinopsis de la conversión de tipos de datos (Página 2373)
DATE (Página 2344)

Conversión explícita de TOD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos TOD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
TOD	BOOL	-	Ninguna conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	TOD_TO_BYTE
	WORD[1)]	X		TOD_TO_WORD
	DWORD[1)]	X		TOD_TO_DWORD
	LWORD[1)]	X		TOD_TO_LWORD
	SINT	X	Como resultado se muestra el número de milisegundos desde medianoche.	TOD_TO_SINT
	USINT	X		TOD_TO_USINT
	INT	X		TOD_TO_INT
	UINT	X		TOD_TO_UINT
	DINT	X		TOD_TO_DINT
	UDINT	X	El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).	TOD_TO_UDINT, T_CONV
	LINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	TOD_TO_LINT
	ULINT	X		TOD_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	-		-
	TIME	X	Como resultado se muestra el tiempo transcurrido desde medianoche.	TOD_TO_TIME
	LTIME	-	Ninguna conversión explícita	-
	DT	X	La conversión ajusta la hora en el tipo de datos de destino.	TOD_TO_DT
	LDT	X		TOD_TO_LDT
	DTL	X		TOD_TO_DTL
	DATE	-	Ninguna conversión explícita	-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LTOD	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de TOD#1:0:0.0 se obtiene LTOD#1:0:0.0)	TOD_TO_LTOD
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

Conversión implícita de TOD (Página 2403)

Sinopsis de la conversión de tipos de datos (Página 2373)

TOD (TIME_OF_DAY) (Página 2345)

Conversión explícita de LTOD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LTOD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
LTOD	BOOL	-	Ninguna conversión explícita	-
	BYTE1)	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LTOD_TO_BYTE
	WORD1)	X		LTOD_TO_WORD
	DWORD1)	X		LTOD_TO_DWORD
	LWORD1)	X		LTOD_TO_LWORD
	SINT	X	Como resultado se muestra el número de nanosegundos desde medianoche.	LTOD_TO_SINT
	USINT	X		LTOD_TO_USINT
	INT	X		LTOD_TO_INT
	UINT	X		LTOD_TO_UINT
	DINT	X		LTOD_TO_DINT
	UDINT	X		LTOD_TO_UDINT
	LINT	X		LTOD_TO_LINT
	ULINT	X		LTOD_TO_ULINT, T_CONV
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	TIME	-		-
	LTIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	LTOD_TO_LTIME
	DT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTOD#1:0:0.0 se obtiene DT#1:0:0.0)	LTOD_TO_DT
	LDT	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTOD#1:0:0.0 se obtiene LDT#1:0:0.0)	LTOD_TO_LDT
	DTL	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTOD#1:0:0.0 se obtiene DTL#1:0:0.0)	LTOD_TO_DTL
	DATE	-	Ninguna conversión explícita	-
	TOD	X	Aquí se produce una conversión del valor de origen al tipo de datos de destino (p. ej., de LTOD#1:0:0.0 se obtiene TOD#1:0:0.0)	LTOD_TO_TOD
	STRING	-	Ninguna conversión explícita	-
	WSTRING	-		-
	CHAR	-		-
	WCHAR	-		-

x: Conversión posible

- : Conversión no posible

[1] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y los tipos de datos CHAR y WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Conversión implícita de LTOD (Página 2404)

Sinopsis de la conversión de tipos de datos (Página 2373)

LTOD (LTIME_OF_DAY) (Página 2345)

Cadenas de caracteres

Conversión explícita de CHAR

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos CHAR y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
CHAR	BOOL	-	Ninguna conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	CHAR_TO_BYTE
	WORD[1)]	X		CHAR_TO_WORD
	DWORD[1)]	X		CHAR_TO_DWORD
	LWORD[1)]	X		CHAR_TO_LWORD
	SINT	X		CHAR_TO_SINT
	USINT	X		CHAR_TO_USINT
	INT	X		CHAR_TO_INT
	UINT	X		CHAR_TO_UINT
	DINT	X		CHAR_TO_DINT
	UDINT	X		CHAR_TO_UDINT
	LINT	X		CHAR_TO_LINT
	ULINT	X		CHAR_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	STRING	X	El valor se convierte en el primer carácter de la cadena de caracteres (STRING). Si la longitud de la cadena de caracteres no está definida, se ajustará la longitud "1" tras la conversión. Si la longitud de la cadena de caracteres está definida, esta no cambia tras la conversión.	CHAR_TO_STRING, S_CONV
	WSTRING	-	Ninguna conversión explícita	-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WCHAR	X		CHAR_TO_WCHAR
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y el tipo de datos WCHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

Conversión implícita de CHAR (Página 2406)

Sinopsis de la conversión de tipos de datos (Página 2373)

CHAR (carácter) (Página 2349)

Conversión explícita de WCHAR

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos WCHAR y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
WCHAR	BOOL	-	Ninguna conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WCHAR_TO_BYTE
	WORD[1)]	X		WCHAR_TO_WORD
	DWORD[1)]	X		WCHAR_TO_DWORD
	LWORD[1)]	X		WCHAR_TO_LWORD
	SINT	X		WCHAR_TO_SINT
	USINT	X		WCHAR_TO_USINT
	INT	X		WCHAR_TO_INT
	UINT	X		WCHAR_TO_UINT
	DINT	X		WCHAR_TO_DINT
	UDINT	X		WCHAR_TO_UDINT
	LINT	X		WCHAR_TO_LINT
	ULINT	X		WCHAR_TO_ULINT
	REAL	-	Ninguna conversión explícita	-
	LREAL	-		-
	S5TIME	-		-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	LTOD	-		-
	DATE	-		-
	STRING	-		-
	WSTRING	X	El valor se convierte en el primer carácter de la cadena de caracteres (WSTRING). Si la longitud de la cadena de caracteres no está definida, se ajustará la longitud "1" tras la conversión. Si la longitud de la cadena de caracteres está definida, esta no cambia tras la conversión.	WCHAR_TO_WSTRING
	CHAR	X		WCHAR_TO_CHAR

x: Conversión posible

- : Conversión no posible

[1] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho deseado, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

Sinopsis de la conversión de tipos de datos (Página 2373)

WCHAR (carácter) (Página 2349)

Conversión implícita de WCHAR (Página 2407)

Conversión explícita de STRING

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos STRING y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
STRING	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	X	La conversión comienza por el primer carácter de la cadena de caracteres (STRING) y termina en el fin de la cadena o bien en el primer carácter no permitido. Los caracteres siguientes están permitidos en la conversión:	STRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		STRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		STRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		STRING_TO_UINT, S_CONV, STRG_VAL

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DINT	X	• Cifra • Signo • Punto El primer carácter de la cadena de caracteres puede ser un signo (+, -) o una cifra. Se ignoran los espacios iniciales. El punto actúa como separador en la conversión de números en coma flotante. La notación exponencial "e" o "E" no está permitida. A la izquierda del punto decimal se permite usar comas para separar los millares, ya que se ignora. Si la estructura de la cadena de caracteres no es válida para la conversión o se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	STRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		STRING_TO_UDINT, S_CONV, STRG_VAL
	LINT	X		STRING_TO_LINT, S_CONV, STRG_VAL
	ULINT	X		STRING_TO_ULINT, S_CONV, STRG_VAL
	REAL	X		STRING_TO_REAL, S_CONV, STRG_VAL
	LREAL	X		STRING_TO_LREAL, S_CONV, STRG_VAL
	S5TIME	-	Ninguna conversión explícita	-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	CHAR	X	El primer carácter de la cadena de caracteres (STRING) se transfiere al tipo de datos de destino. Si la cadena de caracteres está vacía, se escribirá el valor "0" en el tipo de datos de destino.	STRING_TO_CHAR, S_CONV
	WCHAR	-	Ninguna conversión explícita	-
	WSTRING	X		STRING_TO_WSTRING
x: Conversión posible - : Conversión no posible				

Consulte también

Conversión implícita de STRING (Página 2408)

Sinopsis de la conversión de tipos de datos (Página 2373)

STRING (Página 2350)

Conversión explícita de WSTRING

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos WSTRING y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
WSTRING	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	LWORD	-		-
	SINT	X	La conversión comienza por el primer carácter de la cadena de caracteres (WSTRING) y termina en el fin de la cadena o bien en el primer carácter no permitido. Los caracteres siguientes están permitidos en la conversión: • Cifra • Signo • Punto El primer carácter de la cadena de caracteres puede ser un signo (+, -) o una cifra. Se ignoran los espacios iniciales. El punto actúa como separador en la conversión de números en coma flotante. La notación exponencial "e" o "E" no está permitida. A la izquierda del punto decimal se permite usar comas para separar los millares, ya que se ignora. Si la estructura de la cadena de caracteres no es válida para la conversión o se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	WSTRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		WSTRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		WSTRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		WSTRING_TO_UINT, S_CONV, STRG_VAL
	DINT	X		WSTRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		WSTRING_TO_UDINT, S_CONV, STRG_VAL
	LINT	X		WSTRING_TO_LINT, S_CONV, STRG_VAL
	ULINT	X		WSTRING_TO_ULINT, S_CONV, STRG_VAL
	REAL	X		WSTRING_TO_REAL, S_CONV, STRG_VAL
	LREAL	X		WSTRING_TO_LREAL, S_CONV, STRG_VAL
	S5TIME	-	Ninguna conversión explícita	-
	TIME	-		-
	LTIME	-		-
	DT	-		-
	LDT	-		-
	DTL	-		-
	TOD	-		-
	LTOD	-		-
	DATE	-		-
	CHAR	-		-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	WCHAR	X	El primer carácter de la cadena de caracteres (WSTRING) se transfiere al tipo de datos de destino. Si la cadena de caracteres está vacía, se escribirá el valor "0" en el tipo de datos de destino.	WSTRING_TO_WCHAR
	STRING	X		WSTRING_TO_STRING
x: Conversión posible - : Conversión no posible				

Consulte también

Sinopsis de la conversión de tipos de datos (Página 2373)

WSTRING (Página 2351)

Conversión implícita de WSTRING (Página 2409)

Conversión de tipos de datos

Sinopsis de la conversión de tipos de datos

Introducción

Para combinar lógicamente varios operandos en una instrucción, es necesario verificar la compatibilidad de los tipos de datos. Esto también es aplicable a las asignaciones, o a la asignación de valores a los parámetros de bloque. Si los tipos de datos de los operandos son diferentes, será preciso realizar una conversión.

La conversión puede realizarse de dos maneras, a saber:

- Conversión implícita
  La conversión se realiza automáticamente al ejecutar la instrucción.
- Conversión explícita
  Se utiliza una instrucción de conversión explícita antes de ejecutar la instrucción propiamente dicha.

Nota

Las posibilidades descritas para convertir tipos de datos siempre toman como base la versión más reciente de la CPU (V. 4). Es posible que algunas conversiones marcadas como posibles no estén disponibles en las versiones 1 - 3 de la CPU.

Nota

Conversión de secuencias de bits en SCL

Todas las secuencias de bit (BYTE, WORD, DWORD y LWORD) incluidas en expresiones reciben el mismo tratamiento que los enteros correspondientes sin signo (USINT, UINT, UDINT y ULINT). Por ello, p. ej., la conversión implícita de DWORD a REAL se lleva a cabo del mismo modo que una conversión de UDINT a REAL.

Conversión implícita

La conversión implícita se realiza automáticamente si los tipos de datos de los operandos son compatibles. La compatibilidad puede verificarse según criterios más o menos estrictos:

- Con verificación CEI (ajuste predeterminado)
  Si está activada la verificación CEI, se aplican las siguientes reglas:
  - La conversión implícita de BOOL a otros tipos de datos no es posible.
  - Solo los tipos de datos REAL, BYTE, WORD, DINT, INT, SINT, UDINT, UINT, USINT, TIME, DT, STRING y CHAR se pueden convertir de forma implícita.
  - La longitud de bits del tipo de datos de origen no debe superar la longitud de bits del tipo de datos de destino. El operando del tipo de datos WORD no puede indicarse, por ejemplo, en un parámetro si en él se espera el tipo de datos BYTE.
- Sin verificación CEI
  Si no está activada la verificación CEI, se aplican las siguientes reglas:
  - La conversión implícita de BOOL a otros tipos de datos no es posible.
  - Solo los tipos de datos REAL, LREAL, BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, TIME, DTL, TOD, DATE, STRING y CHAR se pueden convertir de forma implícita.
  - La longitud de bits del tipo de datos de origen no debe superar la longitud de bits del tipo de datos de destino. El operando del tipo de datos DWORD no puede indicarse, por ejemplo, en un parámetro si en él se espera el tipo de datos WORD.
  - En los parámetros de entrada/salida (InOut) la longitud de bits de un operando indicado debe coincidir con la longitud de bits programada para el parámetro correspondiente.

Nota

Conversión implícita sin verificación CEI

El editor de programación identifica los operandos que deben convertirse de forma implícita con un rectángulo gris. Un rectángulo gris oscuro indica que es posible realizar una conversión implícita sin pérdidas de precisión, p. ej., si el tipo de datos SINT se convierte a INT. Un rectángulo gris claro indica que es posible realizar una conversión implícita, aunque pueden producirse fallos en tiempo de ejecución. Si, por ejemplo, el tipo de datos DINT se convierte a INT y se produce un desbordamiento, la salida de habilitación ENO se pone a "0".

Encontrará más información sobre el ajuste de la verificación CEI y la conversión implícita en el apartado "Consulte también".

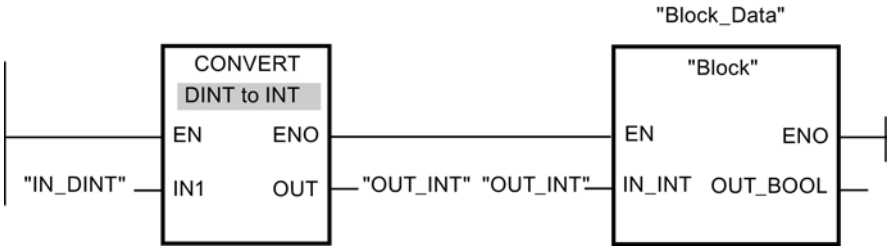
Conversión explícita

Si los operandos no son compatibles, por lo que no se puede realizar una conversión implícita, es posible utilizar una instrucción de conversión explícita. Encontrará las instrucciones de conversión en la Task Card "Instrucciones".

Un posible desbordamiento se indica en la salida de habilitación ENO. El desbordamiento se produce cuando, p. ej., el valor del tipo de datos de origen es mayor que el del tipo de datos de destino.

Encontrará más información sobre la conversión explícita en "Consulte también".

La figura siguiente muestra un ejemplo en el que debe realizarse una conversión explícita de tipos de datos:



El bloque de función "Bloque" espera en el parámetro de entrada "IN_INT" una variable del tipo de datos INT. Por este motivo, el valor de la variable "IN_DINT" debe convertirse primero de DINT a INT. Si el valor de la variable "IN_DINT" se encuentra dentro del rango de valores válido del tipo de datos INT, se produce la conversión. En caso contrario, se notifica un desbordamiento. No obstante, en caso de desbordamiento también tiene lugar una conversión, pero los valores se cortan y la salida de habilitación ENO se pone a "0".

Consulte también

Conversión implícita (Página 2463)

Conversión explícita (Página 2481)

Activar o anular la verificación CEI (Página 2463)

Conversión implícita

Activar o anular la verificación CEI

Se comprueba si los tipos de datos de los operandos utilizados son compatibles. La compatibilidad puede verificarse según criterios más o menos estrictos. Si está activada la "Verificación CEI" se aplican criterios más estrictos.

La verificación CEI se puede ajustar de forma centralizada para todos los bloques nuevos del proyecto, o bien para un bloque en particular.

Ajustar la verificación CEI para bloques nuevos

Para ajustar la verificación CEI en todos los bloques nuevos del proyecto, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC > General" en la navegación local.
3. En el grupo "Ajuste predeterminado para bloques nuevos", active o desactive la casilla de verificación "Verificación CEI".
   La verificación CEI se activa o desactiva para todos los bloques nuevos del programa.

Ajustar la verificación CEI para un bloque en particular

Para ajustar la verificación CEI en un bloque en particular, proceda del siguiente modo:

1. Abra el bloque.
2. Abra la ficha "Propiedades" de la ventana de inspección.
3. Seleccione el grupo "Atributos" en la navegación local.
4. Active o desactive la casilla de verificación "Verificación CEI".
   La verificación CEI se activa o desactiva para este bloque. El ajuste se guarda junto con el proyecto.

Números binarios

Conversión implícita de BOOL

Posibilidades de conversión implícita

No es posible la conversión implícita del tipo de datos BOOL.

Consulte también

BOOL (bit) (Página 2325)

Secuencias de bits

Conversión implícita de BYTE

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos BYTE:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
BYTE	BOOL	-	-	Sin conversión implícita
	WORD	x	x	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DWORD	x	x	
	SINT	-	x	
	USINT	-	x	
	INT	-	x	
	UINT	-	x	
	DINT	-	x	
	UDINT	-	x	
	REAL	-	-	Sin conversión implícita
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	x	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
x: conversión posible -: conversión no posible				

Consulte también

BYTE (bytes) (Página 2326)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de BYTE (Página 2482)

Conversión implícita de WORD

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos WORD:

Origen	Destino	Con verificación CEI	Sin verificación CEI	Significado
WORD	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El byte menos significativo se transfiere al tipo de datos de destino, y el byte más significativo se ignora.
	DWORD	X	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	SINT	-	X	El byte menos significativo se transfiere al tipo de datos de destino, y el byte más significativo se ignora.
	USINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Sin conversión implícita
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	STRING	-	-	Sin conversión implícita
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

WORD (Página 2327)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de WORD (Página 2483)

Conversión implícita de DWORD

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos DWORD:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
DWORD	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	Los bytes derechos se transfieren al tipo de datos de destino, y los bytes izquierdos se ignoran.
	WORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	UDINT	-	X	
	REAL	-	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	-	-	Sin conversión implícita
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DTL	-	-	Sin conversión implícita
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DATE	-	-	Sin conversión implícita
	STRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

DWORD (Página 2327)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de DWORD (Página 2485)

Enteros

Conversión implícita de SINT

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos SINT:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
SINT	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0".
	WORD	-	X	
	DWORD	-	X	
	USINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Transferencia de valores de p. ej. SINT #-1 -> INT #-1, no se rellena con "0".)
	INT	X	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Sin conversión implícita
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

SINT (enteros de 8 bits) (Página 2330)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de SINT (Página 2488)

Conversión implícita de USINT

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos USINT:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
USINT	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0".
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. USINT #10 -> DINT #10, o USINT #128 -> SINT #-128)
	INT	X	X	
	UINT	X	X	
	DINT	X	X	
	UDINT	X	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "1" se convierte en el valor "1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Sin conversión implícita
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

USINT (enteros de 8 bits) (Página 2331)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de USINT (Página 2490)

Conversión implícita de INT

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos INT:

Origen	Destino	Con verificación CEI	Sin verificación CEI	Significado
INT	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. INT #-1 -> SINT #-1, o INT #-32 767 -> UINT #32 769)
	USINT	-	X	
	UINT	-	X	
	DINT	X	X	
	UDINT	-	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Sin conversión implícita
	DTL	-	-	
	TOD	-	-	
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	STRING	-	-	Sin conversión implícita
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

INT (enteros de 16 bits) (Página 2331)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de INT (Página 2491)

Conversión implícita de UINT

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos UINT:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
UINT	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. UINT #100 -> DINT #100, o UINT #60 000 -> INT #-5536)
	USINT	-	X	
	INT	-	X	
	DINT	X	X	
	UDINT	X	X	
	REAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "1" se convierte en el valor "1.0", por ejemplo.)
	LREAL	X	X	
	TIME	-	-	Sin conversión implícita
	DTL	-	-	
	TOD	-	-	
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	STRING	-	-	Sin conversión implícita
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

UINT (enteros de 16 bits) (Página 2332)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de UINT (Página 2493)

Conversión implícita de DINT

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos DINT:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
DINT	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> SINT #-1, o DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	UDINT	-	X	
	REAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 8 388 608)
	LREAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "-1" se convierte en el valor "-1.0", por ejemplo.)
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DTL	-	-	Sin conversión implícita
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DATE	-	-	Sin conversión implícita
	STRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

DINT (enteros de 32 bits) (Página 2333)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de DINT (Página 2495)

Conversión implícita de UDINT

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos UDINT:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
UDINT	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> SINT #-1, o DINT #-1 -> USINT #255)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	REAL	-	X	El patrón de bits del valor de origen se convierte y se transfiere al tipo de datos de destino. (Conversión de valores de p. ej. DINT #-1 -> REAL #-1.0, pero se da una pérdida de precisión en aquellos números cuyo valor absoluto es mayor que 8 388 608)
	LREAL	X	X	El valor se convierte al formato del tipo de datos de destino. (El valor "1" se convierte en el valor "1.0", por ejemplo.)
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DTL	-	-	Sin conversión implícita
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	DATE	-	-	Sin conversión implícita
	STRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

UDINT (enteros de 32 bits) (Página 2334)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de UDINT (Página 2497)

Números en coma flotante

Conversión implícita de REAL

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos REAL:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
REAL	BOOL	-	-	Sin conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.
	SINT	-	X	El patrón de bits del valor de origen se redondea, se convierte y se transfiere al tipo de datos de destino. (Redondeo y conversión de valores de p. ej. REAL #2.5 -> INT #2, o números negativos REAL #-2.5 -> INT #-2 -> USINT #254. En caso de desbordamiento se determina el resto REAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	LREAL	X	X	El valor se transfiere al tipo de datos de destino.
	TIME	-	-	Sin conversión implícita
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

REAL (Página 2338)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de REAL (Página 2498)

Conversión implícita de LREAL

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos LREAL:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
LREAL	BOOL	-	-	Sin conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	SINT	-	X	El patrón de bits del valor de origen se redondea, se convierte y se transfiere al tipo de datos de destino. (Redondeo y conversión de valores de p. ej. REAL #2.5 -> INT #2, o números negativos REAL #-2.5 -> INT #-2 -> USINT #254. En caso de desbordamiento se determina el resto REAL #305.5 -> INT #306 -> USINT #50)
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Sin conversión implícita
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: conversión posible -: conversión no posible				

Consulte también

Conversión explícita de LREAL (Página 2500)

Temporizadores

Conversión implícita de TIME

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos TIME:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
TIME	BOOL	-	-	Sin conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	SINT	-	-	Sin conversión implícita
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	UDINT	-	X	
	REAL	-	-	Sin conversión implícita
	LREAL	-	-	
	DTL	-	-	
	TOD	-	X	Si el valor de origen es inferior a 24 horas (86 400 00 ms), el patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. En cualquier otro caso, el valor de destino no se modifica. El resultado de la conversión muestra el tiempo transcurrido desde medianoche.
	DATE	-	-	Sin conversión implícita
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

TIME (tiempo IEC) (Página 2343)

Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de TIME (Página 2501)

Fecha y hora

Conversión implícita de DTL

Posibilidades de conversión implícita

La conversión implícita del tipo de datos DTL no es posible.

Consulte también

Conversión explícita de DTL (Página 2504)

Conversión implícita de TOD

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos TOD:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
TOD	BOOL	-	-	Sin conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).
	SINT	-	-	Sin conversión implícita
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).
	UDINT	-	X	
	REAL	-	-	Sin conversión implícita
	LREAL	-	-	
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).
	DTL	-	-	Sin conversión implícita
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

TOD (TIME_OF_DAY) (Página 2345)
Activar o anular la verificación CEI (Página 2463)
Sinopsis de la conversión de tipos de datos (Página 2461)
Conversión explícita de TOD (Página 2503)

Conversión implícita de DATE

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos DATE:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
DATE	BOOL	-	-	Sin conversión implícita
	BYTE	-	-	
	WORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de días desde el 01/01/1990.
	DWORD	-	-	Sin conversión implícita
	SINT	-	-	
	USINT	-	-	
	INT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de días desde el 01/01/1990.
	UINT	-	X	
	DINT	-	-	Sin conversión implícita
	UDINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	DTL	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de días desde el 01/01/1990.
	TOD	-	-	Sin conversión implícita
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

DATE (Página 2344)
Activar o anular la verificación CEI (Página 2463)

Sinopsis de la conversión de tipos de datos (Página 2461)
Conversión explícita de DATE (Página 2502)

Cadenas de caracteres

Conversión implícita de CHAR

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos CHAR:

Origen	Destino	Con verificación CEI	Sin verificación CEI	Significado
CHAR	BOOL	-	-	Sin conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. El resto de los bits se rellenan con "0" desde la izquierda.
	WORD	-	X	
	DWORD	-	X	
	SINT	-	X	
	USINT	-	X	
	INT	-	X	
	UINT	-	X	
	DINT	-	X	
	UDINT	-	X	
	REAL	-	-	Sin conversión implícita
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	X	X	El STRING se acorta hasta la longitud 1 y contiene el carácter.
x: Conversión posible -: Conversión no posible				

Consulte también

CHAR (carácter) (Página 2349)
Activar o anular la verificación CEI (Página 2463)
Sinopsis de la conversión de tipos de datos (Página 2461)
Conversión explícita de CHAR (Página 2505)

Conversión implícita de STRING

Posibilidades de conversión implícita

La tabla siguiente indica las posibilidades de conversión implícita del tipo de datos STRING:

Origen	Destino	Con Verificación CEI	Sin Verificación CEI	Significado
STRING	BOOL	-	-	Sin conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	-	
	SINT	-	-	
	USINT	-	-	
	INT	-	-	
	UINT	-	-	
	DINT	-	-	
	UDINT	-	-	
	REAL	-	-	
	LREAL	-	-	
	TIME	-	-	
	DTL	-	-	
	DATE	-	-	
	TOD	-	-	
	CHAR	-	X	Se devuelve el primer carácter del STRING si el STRING contiene uno o más caracteres. De lo contrario, el carácter se muestra con la codificación $00.
x: Conversión posible -: Conversión no posible				

Consulte también

Conversión explícita de STRING (Página 2506)

Conversión explícita

Números binarios

Conversión explícita de BOOL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos BOOL y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
BOOL	BYTE	X	En el tipo de datos de destino solamente se activa el LSB (Least Significant Bit). La salida de habilitación ENO es siempre "1".	BOOL_TO_BYTE
	WORD	X		BOOL_TO_WORD
	DWORD	X		BOOL_TO_DWORD
	SINT	X		BOOL_TO_SINT
	USINT	X		BOOL_TO_USINT
	INT	X		BOOL_TO_INT
	UINT	X		BOOL_TO_UINT
	DINT	X		BOOL_TO_DINT
	UDINT	X		BOOL_TO_UDINT
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	TIME	-		-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	-		-
	CHAR	-		-
x: Conversión posible - : Conversión no posible				

Consulte también

BOOL (bit) (Página 2325)

Conversión implícita de BYTE (Página 2465)

Sinopsis de la conversión de tipos de datos (Página 2461)

Secuencias de bits

Conversión explícita de BYTE

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos BYTE y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
BYTE[1)]	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	BYTE_TO_BOOL
	WORD[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	BYTE_TO_WORD
	DWORD[1)]	X		BYTE_TO_DWORD
	SINT	X		BYTE_TO_SINT
	USINT	X		BYTE_TO_USINT
	INT	X		BYTE_TO_INT
	UINT	X		BYTE_TO_UINT
	DINT	X		BYTE_TO_DINT
	UDINT	X		BYTE_TO_UDINT
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	BYTE_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	BYTE_TO_TOD
	DATE	X		BYTE_TO_DATE
	STRING	-	Sin conversión explícita	-

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	BYTE_TO_CHAR
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta como USINT, WORD como UINT y DWORD como UDINT.				

Consulte también

BYTE (bytes) (Página 2326)

Conversión implícita de BYTE (Página 2465)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de WORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos WORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
WORD[1)]	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	WORD_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	WORD_TO_BYTE
	DWORD[1)]	X		WORD_TO_DWORD

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción

	SINT	X	ENO = TRUE
			#sint1 := WORD_TO_SINT(16#FFFF); // -1 a
			#sint1 := WORD_TO_SINT(16#FF80); // -128
			#sint1 := WORD_TO_SINT(16#0); // 0 a
			#sint1 := WORD_TO_SINT(16#007F); // 127
			
			ENO = FALSE
			#sint1 := WORD_TO_SINT(16#FF7F); // -129 a
			#sint1 := WORD_TO_SINT(16#8000); // -32768
			#sint1 := WORD_TO_SINT(16#0080); // 128 a
			#sint1 := WORD_TO_SINT(16#7FFF); // 32767
				WORD_TO_SINT

	USINT	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	WORD_TO_USINT
	INT	X		WORD_TO_INT
	UINT	X		WORD_TO_UINT
	DINT	X		WORD_TO_DINT
	UDINT	X		WORD_TO_UDINT
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	WORD_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	WORD_TO_TOD
	DATE	X		WORD_TO_DATE
	STRING	-	Sin conversión explícita	-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	WORD_TO_CHAR
WORD_BCD16	INT	X	El valor que debe convertirse tiene el tipo de datos WORD y se aplica como valor codificado en BCD entre -999 y +999. Tras la conversión, el resultado es un número entero (representación binaria) del tipo INT. Se produce una conversión auténtica. Si el patrón de bits incluye una tétrada no válida, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	WORD_BCD16_TO_INT

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
BCD16	INT	X		BCD16_TO_INT
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta como USINT, WORD como UINT y DWORD como UDINT.				

Consulte también

WORD (Página 2327)

Conversión implícita de WORD (Página 2466)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de DWORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DWORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
DWORD1)	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	DWORD_TO_BOOL
	BYTE1)	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DWORD_TO_BYTE
	WORD1)	X		DWORD_TO_WORD

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción

	SINT	X	ENO = TRUE
			#sint1 := DWORD_TO_SINT(16#FFFF_FFFF); // -1 bis
			#sint1 := DWORD_TO_SINT(16#FFFF_FF80); // -128
			#sint1 := DWORD_TO_SINT(16#0); // 0 bis
			#sint1 := DWORD_TO_SINT(16#0000_007F); // 127
			
			ENO = FALSE
			#sint1 := DWORD_TO_SINT(16#FFFF_FF7F); // -129
			#sint1 := DWORD_TO_SINT(16#8000_0000); // -2147483648
			#sint1 := DWORD_TO_SINT(16#0000_0080); // 128 bis
			#sint1 := DWORD_TO_SINT(16#7FFF_FFFF); // 2147483647
				DWORD_TO_SINT

	USINT	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DWORD_TO_USINT

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción

	INT	X	ENO = TRUE
			#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 bis
			#int1 := DWORD_TO_INT(16#FFFF_8000); // -32768
			#int1 := DWORD_TO_INT(16#0); // 0 bis
			#int1 := DWORD_TO_INT(16#0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769
			#int1 := DWORD_TO_INT(16#8000_0000); // -2147483648
			#int1 := DWORD_TO_INT(16#8000); // 32768 bis
			#int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647
				DWORD_TO_INT

	UINT	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DWORD_TO_UINT
	DINT	X		DWORD_TO_DINT
	UDINT	X		DWORD_TO_UDINT
	REAL	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. Si no se producen errores durante la conversión, el estado lógico de ENO = 1; si se produce un error durante la ejecución, el estado lógico de ENO = 0.	DWORD_TO_REAL
	LREAL	-	Sin conversión explícita	-
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DWORD_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DWORD_TO_TOD
	DATE	X		DWORD_TO_DATE
	STRING	-	Sin conversión explícita	-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DWORD_TO_CHAR
DWORD_BCD32	DINT	X	El valor que debe convertirse tiene el tipo de datos DWORD y se aplica	DWORD_BCD32_TO_DINT

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
BCD32	DINT	X	como valor codificado en BCD entre -9999999 y +9999999. Tras la conversión, el resultado es un número entero (representación binaria) del tipo DINT. Se produce una conversión auténtica. Si el patrón de bits incluye una tétrada no válida, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	BCD32_TO_DINT
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) se interpretan como número entero sin signo con la misma longitud de bits. El tipo de datos BYTE se interpreta como USINT, WORD como UINT y DWORD como UDINT.				

Consulte también

DWORD (Página 2327)

Conversión implícita de DWORD (Página 2467)

Sinopsis de la conversión de tipos de datos (Página 2461)

Enteros

Conversión explícita de SINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos SINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
SINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	SINT_TO_BOOL
	BYTE1)	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. Si se convierte un valor negativo a un	SINT_TO_BYTE
	WORD1)	X		SINT_TO_WORD

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	DWORD[1)]	X	tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	SINT_TO_DWORD
	USINT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0".	SINT_TO_USINT
	INT	X		SINT_TO_INT
	UINT	X		SINT_TO_UINT
	DINT	X		SINT_TO_DINT
	UDINT	X		SINT_TO_UDINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "-1" se transforma en el valor "-1.0" con la instrucción "Convertir valor" (CONVERT)).	SINT_TO_REAL, NORM_X
	LREAL	X		SINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	SINT_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0)	SINT_TO_TOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1)	SINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa precedida de signo. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	SINT_TO_STRING, S_CONV, VAL_STRG
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si se convierten valores negativos, la salida de habilitación ENO se pone a "0".	SINT_TO_CHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

SINT (enteros de 8 bits) (Página 2330)

Conversión implícita de SINT (Página 2468)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de USINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos USINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
USINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	USINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	USINT_TO_BYTE
	WORD[1)]	X		USINT_TO_WORD
	DWORD[1)]	X		USINT_TO_DWORD
	SINT	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si al convertir se cambia el signo, la salida de habilitación ENO se pone a "0".	USINT_TO_SINT
	INT	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	USINT_TO_INT
	UINT	X		USINT_TO_UINT
	DINT	X		USINT_TO_DINT
	UDINT	X		USINT_TO_UDINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "1" se transforma en el valor "1.0" con la instrucción "Convertir valor" (CONVERT)).	USINT_TO_REAL, NORM_X
	LREAL	X		USINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	USINT_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	USINT_TO_TOD
	DATE	X		USINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	USINT_TO_STRING, S_CONV, VAL_STRG

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.	USINT_TO_CHAR
x: Conversión posible - : Conversión no posible [1)] Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

USINT (enteros de 8 bits) (Página 2331)

Conversión implícita de USINT (Página 2469)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de INT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos INT y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
INT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	INT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	INT_TO_BYTE
	WORD[1)]	X		INT_TO_WORD
	DWORD[1)]	X		INT_TO_DWORD
	SINT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin	INT_TO_SINT
	USINT	X		INT_TO_USINT
	UINT	X		INT_TO_UINT
	DINT	X		INT_TO_DINT

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	UDINT	X	signo, la salida de habilitación ENO se pone a "0".	INT_TO_UDINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "-1" se transforma en el valor "-1.0" con la instrucción "Convertir valor" (CONVERT)).	INT_TO_REAL, NORM_X
	LREAL	X		INT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	INT_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0; verificación de límite de 24 h)	INT_TO_TOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1; verificación de valor negativo)	INT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa precedida de signo. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	INT_TO_STRING, S_CONV, VAL_STRG[)]
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0".	INT_TO_CHAR
	BCD16	X	El valor que debe convertirse tiene el tipo INT y se aplica como número entero con un valor entre -999 y +999. Tras la conversión, el resultado es un número codificado en BCD del tipo WORD. Se produce una conversión auténtica. Si el valor está fuera del rango de destino, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	INT_TO_BCD16
	BCD16_WORD	X		INT_TO_BCD16_WORD

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

INT (enteros de 16 bits) (Página 2331)

Conversión implícita de INT (Página 2470)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de UINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos UINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
UINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	UINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. Si en este proceso se pierden bits, la salida de habilitación ENO se pone a "0".	UINT_TO_BYTE
	WORD[1)]	X		UINT_TO_WORD
	DWORD[1)]	X		UINT_TO_DWORD
	SINT	X		UINT_TO_SINT
	USINT	X		UINT_TO_USINT
	INT	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si al convertir se cambia el bit de signo, la salida de habilitación ENO se pone a "0".	UINT_TO_INT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	UINT_TO_DINT
	UDINT	X		UINT_TO_UDINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "1" se transforma en el valor "1.0" con la instrucción "Convertir valor" (CONVERT)).	UINT_TO_REAL, NORM_X
	LREAL	X		UINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	UINT_TO_TIME
	DTL	-	Sin conversión explícita	-

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0; verificación de límite de 24 h)	UINT_TO_TOD
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1; verificación de valor negativo)	UINT_TO_DATE, T_CONV
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	UINT_TO_STRING, S_CONV, VAL_STRG
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	UINT_TO_CHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

UINT (enteros de 16 bits) (Página 2332)

Conversión implícita de UINT (Página 2471)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de DINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
DINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	DINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. Si se convierte un valor negativo a un tipo de datos sin signo, la salida de habilitación ENO se pone a "0".	DINT_TO_BYTE
	WORD[1)]	X		DINT_TO_WORD
	DWORD[1)]	X		DINT_TO_DWORD
	SINT	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0".	DINT_TO_SINT
	USINT	X		DINT_TO_USINT
	INT	X		DINT_TO_INT
	UINT	X		DINT_TO_UINT
	UDINT	X		DINT_TO_UDINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "-1" se transforma en el valor "-1.0" con la instrucción "Convertir valor" (CONVERT)).	DINT_TO_REAL, NORM_X
	LREAL	X		DINT_TO_LREAL, NORM_X
	TIME	X	El valor se transfiere al tipo de datos de destino y se interpreta como milisegundos.	DINT_TO_TIME, T_CONV
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0)	DINT_TO_TOD

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1)	DINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa precedida de signo. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	DINT_TO_STRING, S_CONV, VAL_STRG
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0".	DINT_TO_CHAR
	BCD32	X	El valor que debe convertirse tiene el tipo DINT y se aplica como número entero con un valor entre –999999 y +9999999. Tras la conversión, el resultado es un número codificado en BCD del tipo DWORD. Si se produce un desbordamiento, la salida de habilitación se pone a "0". Se produce una conversión auténtica. Si el valor está fuera del rango de destino, no se dispara ningún error síncrono, sino que solamente se activa el bit de estado OV.	DINT_TO_BCD32
	BCD32_DWORD	X		DINT_TO_BCD32_DWORD

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

DINT (enteros de 32 bits) (Página 2333)

Conversión implícita de DINT (Página 2472)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de UDINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos UDINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
UDINT	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	UDINT_TO_BOOL
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si en este proceso se pierden bits, la salida de habilitación ENO se pone a "0".	UDINT_TO_BYTE
	WORD[1)]	X		UDINT_TO_WORD
	DWORD[1)]	X		UDINT_TO_DWORD
	SINT	X		UDINT_TO_SINT
	USINT	X		UDINT_TO_USINT
	INT	X		UDINT_TO_INT
	UINT	X		UDINT_TO_UINT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si al convertir se cambia el bit de signo, la salida de habilitación ENO se pone a "0".	UDINT_TO_DINT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "1" se transforma en el valor "1.0" con la instrucción "Convertir valor" (CONVERT)).	UDINT_TO_REAL, NORM_X
	LREAL	X		UDINT_TO_LREAL, NORM_X
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha, interpretado como milisegundos, al tipo de datos de destino.	UDINT_TO_TIME
	DTL	-	Sin conversión explícita	-
	TOD	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en milisegundos desde 0:0; verificación de límite de 24 h)	UDINT_TO_TOD, T_CONV

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	DATE	X	El patrón de bits del valor de origen se transfiere al tipo de datos de destino sin convertir. (Del valor "-1" (16#FF) se obtiene el valor "-1" (16#FFFFFFFF). Cuando se convierte un valor negativo en un tipo de datos de destino sin signo, la salida de habilitación ENO se pone a "0". (Interpretación en días desde 1990-1-1; verificación de valor negativo)	UDINT_TO_DATE
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	UDINT_TO_STRING, S_CONV, VAL_STRG
	CHAR[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	UDINT_TO_CHAR

x: Conversión posible

- : Conversión no posible

[1)] Las secuencias de bits (BYTE, WORD, DWORD, LWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario (el signo inexistente se sustituye por ceros), y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.

Consulte también

UDINT (enteros de 32 bits) (Página 2334)

Conversión implícita de UDINT (Página 2473)

Sinopsis de la conversión de tipos de datos (Página 2461)

Números en coma flotante

Conversión explícita de REAL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos REAL y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
REAL	BOOL	-	Sin conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.	REAL_TO_DWORD

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	SINT	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada. Si durante la conversión se rebasa el rango permitido de valores del tipo de datos de destino o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0".	REAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		REAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		REAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		REAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		REAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		REAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	LREAL	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada, p. ej., TRUNC(2.5) = 2.0; CEIL(2.5) = 3.0	REAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Sin conversión explícita	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0". La longitud mín. de la cadena de caracteres es de 14 caracteres.	REAL_TO_STRING, S_CONV, VAL_STRG
	CHAR	-	Sin conversión explícita	-
x: Conversión posible -: Conversión no posible				

Consulte también

REAL (Página 2338)

Conversión implícita de REAL (Página 2474)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de LREAL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos LREAL y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
LREAL	BOOL	-	Sin conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	SINT	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada. Si al convertir se rebasa el rango de valores permitido o el valor que se desea convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0".	LREAL_TO_SINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	USINT	X		LREAL_TO_USINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	INT	X		LREAL_TO_INT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UINT	X		LREAL_TO_UINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	DINT	X		LREAL_TO_DINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	UDINT	X		LREAL_TO_UDINT, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	REAL	X	El valor se convierte al tipo de datos de destino. Si al convertir se rebasa el rango de valores permitido o el valor que se desea convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0". Se tolera la pérdida de precisión.	LREAL_TO_LREAL, ROUND, CEIL, FLOOR, TRUNC, NORM_X, SCALE_X
	TIME	-	Sin conversión explícita	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0". La longitud mín. de la cadena de caracteres es de 21 caracteres.	REAL_TO_STRING, S_CONV, VAL_STRG

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	CHAR	-	Sin conversión explícita	-
x: Conversión posible -: Conversión no posible				

Consulte también

LREAL (Página 2339)

Sinopsis de la conversión de tipos de datos (Página 2461)

Temporizadores

Conversión explícita de TIME

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos TIME y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
TIME	BOOL	-	Sin conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.	TIME_TO_BYTE
	WORD[1)]	X		TIME_TO_WORD
	DWORD[1)]	X		TIME_TO_DWORD
	SINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha, interpretado como milisegundos, al tipo de datos de destino.	TIME_TO_SINT
	USINT	X		TIME_TO_USINT
	INT	X		TIME_TO_INT
	UINT	X		TIME_TO_UINT
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.	TIME_TO_DINT, T_CONV
	UDINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha, interpretado como milisegundos, al tipo de datos de destino. El cambio de signo da lugar a que la salida de habilitación ENO sea "0".	TIME_TO_UDINT
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	DTL	-		-

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino. Si el valor de origen está fuera del rango de valores de TOD, el tipo de datos de destino no se cambia.	TIME_TO_TOD
	DATE	-	Sin conversión explícita	-
	STRING	-		-
	CHAR	-		-
x: Conversión posible -: Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

TIME (tiempo IEC) (Página 2343)

Conversión implícita de TIME (Página 2476)

Sinopsis de la conversión de tipos de datos (Página 2461)

Fecha y hora

Conversión explícita de DATE

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DATE y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
DATE	BOOL	-	Sin conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DATE_TO_BYTE
	WORD[1)]	X		DATE_TO_WORD
	DWORD[1)]	X		DATE_TO_DWORD
	SINT	X	Como resultado se muestra el número de días desde el 1.1.1990.	DATE_TO_SINT
	USINT	X		DATE_TO_USINT
	INT	X		DATE_TO_INT
	UINT	X		DATE_TO_UINT
	DINT	X		DATE_TO_DINT
	UDINT	X		DATE_TO_UDINT
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	TIME	-		-

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	DTL	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	DATE_TO_DTL
	TOD	-	Sin conversión explícita	-
	STRING	-		-
	CHAR	-		-
x: Conversión posible - : Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

DATE (Página 2344)

Conversión implícita de DATE (Página 2478)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de TOD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos TOD y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
TOD	BOOL	-	Sin conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.	TOD_TO_BYTE
	WORD[1)]	X		TOD_TO_WORD
	DWORD[1)]	X		TOD_TO_DWORD
	SINT	X	Como resultado se muestra el número de milisegundos desde medianoche.	TOD_TO_SINT
	USINT	X		TOD_TO_USINT
	INT	X		TOD_TO_INT
	UINT	X		TOD_TO_UINT
	DINT	X		TOD_TO_DINT
	UDINT	X	El resultado de la conversión equivale al número de milisegundos desde el inicio del día (0:00 horas).	TOD_TO_UDINT, T_CONV
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	TIME	X	Como resultado se muestra el tiempo transcurrido desde medianoche.	TOD_TO_TIME

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	DTL	X	Como resultado el día se pone a 1.1.1970.	TOD_TO_DTL
	DATE	-	Sin conversión explícita	-
	STRING	-		-
	CHAR	-		-
x: Conversión posible -: Conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario incluido el signo, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

TOD (TIME_OF_DAY) (Página 2345)

Conversión implícita de TOD (Página 2477)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de DTL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DTL y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
DTL	BYTE	-	Sin conversión explícita	-
	WORD	-		-
	DWORD	-		-
	SINT	-		-
	USINT	-		-
	INT	-		-
	UINT	-		-
	DINT	-		-
	UDINT	-		-
	REAL	-		-
	LREAL	-		-
	TIME	-		-
	TOD	X	Durante la conversión se extrae la hora del formato DTL y se escribe en el tipo de datos de destino.	DTL_TO_TOD, T_CONV
	DATE	X	Durante la conversión se extrae la fecha del formato DTL y se escribe en el tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	DTL_TO_DATE, T_CONV

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	STRING	-	Sin conversión explícita	-
	CHAR	-		-
x: Conversión posible -: Conversión no posible				

Consulte también

DTL (Página 2348)

Sinopsis de la conversión de tipos de datos (Página 2461)

Cadenas de caracteres

Conversión explícita de CHAR

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos CHAR y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
CHAR	BOOL	-	Sin conversión explícita	-
	BYTE[1)]	X	El patrón de bits del valor de origen se transfiere sin cambios y justificado a la derecha al tipo de datos de destino.	CHAR_TO_BYTE
	WORD[1)]	X		CHAR_TO_WORD
	DWORD[1)]	X		CHAR_TO_DWORD
	SINT	X		CHAR_TO_SINT
	USINT	X		CHAR_TO_USINT
	INT	X		CHAR_TO_INT
	UINT	X		CHAR_TO_UINT
	DINT	X		CHAR_TO_DINT
	UDINT	X		CHAR_TO_UDINT
	REAL	-	Sin conversión explícita	-
	LREAL	-		-
	TIME	-		-
	DTL	-		-
	TOD	-		-
	DATE	-		-

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	STRING	X	El valor se convierte en el primer carácter de la cadena de caracteres (STRING). Si la longitud de la cadena de caracteres no está definida, después de la conversión se ajusta la longitud "1". Si la longitud de la cadena de caracteres está definida, permanece inalterada después de la conversión.	CHAR_TO_STRING
x: conversión posible - : conversión no posible 1) Las secuencias de bits (BYTE, WORD, DWORD) y el tipo de datos CHAR se amplían en primer lugar hasta el ancho necesario, y a continuación se copian los bits. La interpretación se decide en función del tipo de origen.				

Consulte también

CHAR (carácter) (Página 2349)

Conversión implícita de CHAR (Página 2479)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión explícita de STRING

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos STRING y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
STRING	BOOL	-	Sin conversión explícita	-
	BYTE	-		-
	WORD	-		-
	DWORD	-		-
	SINT	X	La conversión comienza por el primer carácter de la cadena de caracteres (STRING) y termina en el fin de la cadena o bien en el primer carácter no permitido. Se admiten para la conversión los siguientes caracteres: • Cifras • Signos • Punto El primer carácter de la cadena puede ser un signo (+, -) o una cifra. Los espacios en blanco iniciales se ignoran. Al convertir números en coma flotante, se usa el punto como carácter de separación. La notación exponencial "e" o "E" no está permitida. A la izquierda del punto decimal se permite usar	STRING_TO_SINT, S_CONV, STRG_VAL
	USINT	X		STRING_TO_USINT, S_CONV, STRG_VAL
	INT	X		STRING_TO_INT, S_CONV, STRG_VAL
	UINT	X		STRING_TO_UINT, S_CONV, STRG_VAL
	DINT	X		STRING_TO_DINT, S_CONV, STRG_VAL
	UDINT	X		STRING_TO_UDINT, S_CONV, STRG_VAL
	REAL	X		STRING_TO_REAL, S_CONV, STRG_VAL

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	LREAL	X	comas para separar los millares, ya que se ignora. Si la estructura de la cadena de caracteres no es válida para la conversión o se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	STRING_TO_LREAL, S_CONV, STRG_VAL
	TIME	-	Sin conversión explícita	-
	DTL	-		-
	TOD	-		-
	DATE	-		-
	CHAR[1)]	X	El primer carácter de la cadena de caracteres (STRING) se transfiere al tipo de datos de destino. Si la cadena de caracteres está vacía, se escribe en el tipo de datos de destino el valor "0".	STRING_TO_CHAR, S_CONV
x: Conversión posible - : Conversión no posible				

Consulte también

STRING (Página 2350)

Sinopsis de la conversión de tipos de datos (Página 2461)

Conversión de tipos de datos

Sinopsis de la conversión de tipos de datos

Introducción

Para combinar lógicamente varios operandos en una instrucción, es necesario verificar la compatibilidad de los tipos de datos. Esto también es aplicable a las asignaciones, o a la asignación de valores a los parámetros de bloque. Si los tipos de datos de los operandos son diferentes, será preciso realizar una conversión.

La conversión puede realizarse de dos maneras, a saber:

- Conversión implícita
  Los lenguajes de programación KOP, FUP, SCL y GRAPH soportan la conversión implícita. La conversión implícita no es posible en el lenguaje de programación AWL.
- Conversión explícita

Nota

Conversión de secuencias de bits en SCL

Todas las secuencias de bit (BYTE, WORD, DWORD y LWORD) incluidas en expresiones reciben el mismo tratamiento que los enteros sin signo (USINT, UINT, UDINT y ULINT). Por ello, p. ej. la conversión implícita de DWORD a REAL se lleva a cabo del mismo modo que una conversión de UDINT a REAL.

Conversión implícita

La conversión implícita se realiza automáticamente si los tipos de datos de los operandos son compatibles. La compatibilidad puede verificarse según criterios más o menos estrictos:

- Con verificación IEC
  Si en los lenguajes de programación KOP, FUP y GRAPH está ajustada la verificación IEC, se aplican las siguientes reglas:
  - Sólo los tipos de datos BYTE y WORD pueden convertirse de forma implícita.
  - La longitud de bits del tipo de datos de origen no puede rebasar la longitud de bits del tipo de datos de destino. El operando del tipo de datos WORD no puede indicarse, por ejemplo, en un parámetro si él se espera el tipo de datos BYTE.

  Si en el lenguaje de programación SCL está ajustada la verificación IEC, se aplican las siguientes reglas:
  - La conversión implícita de secuencias de bits no es posible en otros tipos de datos. El operando del tipo de datos WORD no puede indicarse, por ejemplo, en un parámetro si en él se espera el tipo de datos INT.
  - La longitud de bits del tipo de datos de origen no puede rebasar la longitud de bits del tipo de datos de destino. El operando del tipo de datos WORD no puede indicarse, por ejemplo, en un parámetro si en el mismo se espera el tipo de datos BYTE.
- Sin verificación IEC (ajuste predeterminado)
  Si en los lenguajes de programación KOP, FUP y GRAPH no está ajustada la verificación IEC, se aplican las siguientes reglas:
  - Los tipos de datos BYTE, WORD, DWORD, INT, DINT, TIME, S5TIME, TOD, DATE y CHAR pueden convertirse de forma implícita.
  - La longitud de bits del tipo de datos de origen no puede rebasar la longitud de bits del tipo de datos de destino. El operando del tipo de datos DWORD no puede indicarse, por ejemplo, en un parámetro si en el mismo se espera el tipo de datos WORD.

  Si en el lenguaje de programación SCL no está ajustada la verificación IEC, se aplican las siguientes reglas:
  - La conversión implícita de secuencias de bits es posible en otros tipos de datos. El operando del tipo de datos WORD puede indicarse, por ejemplo, en un parámetro si en el mismo se espera el tipo de datos INT.
  - La conversión implícita de secuencias de bits no es posible en números en coma flotante. El operando del tipo de datos WORD no puede indicarse, por ejemplo, en un parámetro si en el mismo se espera el tipo de datos REAL.
  - La conversión implícita de secuencias de bits en los tipos de datos TIME, TOD, DATE y CHAR sólo es posible si éstos tienen la misma longitud de bits. El operando del tipo de datos DWORD no puede indicarse, por ejemplo, en un parámetro si en el mismo se espera el tipo de datos DATE.
  - La longitud de bits del tipo de datos de origen no puede rebasar la longitud de bits del tipo de datos de destino. El operando del tipo de datos DINT no puede indicarse, por ejemplo, en un parámetro si en el mismo se espera el tipo de datos INT.
  - En los parámetros de entrada/salida (InOut) la longitud de bits de un operando indicado debe coincidir con la longitud de bits programada para el parámetro correspondiente.

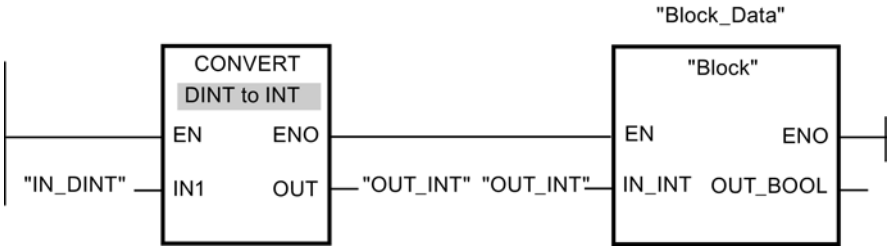
Conversión explícita

Si los operandos no son compatibles, por lo que no se puede realizar una conversión implícita, es posible utilizar una instrucción de conversión explícita. Encontrará las instrucciones de conversión en la Task Card "Instrucciones".

Un posible desbordamiento se indica en la salida de habilitación ENO. Un desbordamiento tiene lugar cuando, p. ej., el valor del tipo de datos de origen es mayor que el valor del tipo de datos de destino.

Encontrará más información sobre la conversión explícita en "Consulte también".

La figura siguiente muestra un ejemplo en el que debe realizarse una conversión explícita de tipos de datos:



El bloque de función "Block" espera en el parámetro de entrada "IN_INT" una variable del tipo de datos INT. Por ello, el valor de la variable "IN_DINT" se debe convertir primero de DINT a INT. Si el valor de la variable "IN_DINT" se encuentra dentro del rango de valores permitido del tipo de datos INT, se realiza una conversión. De lo contrario se notifica un desbordamiento. No obstante, también en caso de desbordamiento se realiza una conversión, pero los valores se cortan y la salida de habilitación ENO se pone a "0".

Consulte también

Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita (Página 2510)

Conversión explícita (Página 2522)

Conversión implícita

Activar o desactivar la verificación IEC

Se comprueba si los tipos de datos de los operandos utilizados son compatibles. La compatibilidad puede verificarse según criterios más o menos estrictos. Si está activada la "Verificación IEC", se aplican criterios más estrictos.

La verificación IEC se puede ajustar de forma centralizada para todos los bloques nuevos del proyecto, o bien para un bloque en particular.

Ajustar la verificación IEC de bloques nuevos

Para ajustar la verificación IEC de todos los bloques nuevos del programa, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC > General" en la navegación local.
3. En el grupo "Ajuste predeterminado para bloques nuevos" active o desactive la casilla de verificación "Verificación IEC".
   La verificación IEC se activa o desactiva para todos los bloques nuevos del programa.

Ajustar la verificación IEC de un bloque en particular

Para ajustar la verificación IEC de un bloque en particular, proceda del siguiente modo:

1. Abra el bloque.
2. Abra la ficha "Propiedades" de la ventana de inspección.
3. Seleccione el grupo "Atributos" en la navegación local.
4. Active o desactive la casilla de verificación "Verificación IEC".
   La verificación IEC se activa o desactiva para este bloque. El ajuste se guarda junto con el proyecto.

Consulte también

Sinopsis de la conversión de tipos de datos (Página 2507)

Números binarios

Conversión implícita de BOOL

Posibilidades de conversión implícita

La conversión implícita del tipo de datos BOOL no es posible.

Consulte también

BOOL (bit) (Página 2325)

Secuencias de bits

Conversión implícita de BYTE

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos BYTE:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Significado
BYTE	BOOL	-	-	Ninguna conversión implícita
	WORD	X	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	DWORD	X	X	
	INT	-	-	Ninguna conversión implícita
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	REAL	-	-	Ninguna conversión implícita
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
x: conversión posible -: conversión no posible				

Consulte también

BYTE (bytes) (Página 2326)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de BYTE (Página 2523)

Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de WORD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos WORD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
WORD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	Solamente el byte menos significativo se transfiere al tipo de datos de destino.
	DWORD	X	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.
	INT	-	X	
	DINT	-	X	
	REAL	-	-	Ninguna conversión implícita
	TIME	-	-	
	S5TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	DT	-	-	Ninguna conversión implícita
	TOD	-	-	
	DATE	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	STRING	-	-	Ninguna conversión implícita
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

WORD (Página 2327)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de WORD (Página 2524)

Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de DWORD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DWORD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DWORD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	WORD	-	X	
	INT	-	-	Ninguna conversión implícita
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	REAL	-	-	Ninguna conversión implícita
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	S5TIME	-	-	Ninguna conversión implícita
	DT	-	-	
	TOD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	DATE	-	-	Ninguna conversión implícita
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

DWORD (Página 2327)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de DWORD (Página 2526)

Activar o desactivar la verificación IEC (Página 2510)

Enteros

Conversión implícita de INT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos INT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
INT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	DWORD	-	-	Ninguna conversión implícita
	DINT	X	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	REAL	X	X	
	TIME	-	-	Ninguna conversión implícita
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

INT (enteros de 16 bits) (Página 2331)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de INT (Página 2528)

Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de DINT

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DINT:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DINT	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	INT	-	-	Ninguna conversión implícita
	REAL	-	-	
	TIME	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	S5TIME	-	-	Ninguna conversión implícita
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

DINT (enteros de 32 bits) (Página 2333)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de STRING (Página 2538)

Activar o desactivar la verificación IEC (Página 2510)

Números en coma flotante

Conversión implícita de REAL

Posibilidades para la conversión implícita

La conversión implícita del tipo de datos REAL no es posible.

Consulte también

REAL (Página 2338)
Sinopsis de la conversión de tipos de datos (Página 2507)
Conversión explícita de CHAR (Página 2537)
Activar o desactivar la verificación IEC (Página 2510)

Temporizadores

Conversión implícita de TIME

Posibilidades para la conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos TIME:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
TIME	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	INT	-	-	Ninguna conversión implícita
	DINT	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	REAL	-	-	Ninguna conversión implícita
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

TIME (tiempo IEC) (Página 2343)
Sinopsis de la conversión de tipos de datos (Página 2507)
Conversión explícita de TIME (Página 2532)
Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de S5TIME

Posibilidades para la conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos S5TIME:

Origen	Destino	Con verificación IEC	Sin verificación CEI	Explicación
S5TIME	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.
	DWORD	-	-	Ninguna conversión implícita
	INT	-	-	
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

S5TIME (tiempo) (Página 2341)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de S5TIME (Página 2533)

Activar o desactivar la verificación IEC (Página 2510)

Fecha y hora

Conversión implícita de DATE

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades para la conversión implícita del tipo de datos DATE:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
DATE	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de días desde el 01.01.1990.
	DWORD	-	-	Ninguna conversión implícita
	INT	-	-	
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

DATE (Página 2344)

Sinopsis de la conversión de tipos de datos (Página 2507)

Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de TOD

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades de la conversión implícita del tipo de datos TOD:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
TOD	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	-	
	WORD	-	-	
	DWORD	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión equivale al número de milisegundos desde el comienzo del día (0:00 horas).
	INT	-	-	Ninguna conversión implícita
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	DATE	-	-	
	STRING	-	-	
	CHAR	-	-	
x: Conversión posible -: Conversión no posible				

Consulte también

TOD (TIME_OF_DAY) (Página 2345)

Sinopsis de la conversión de tipos de datos (Página 2507)

Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de DT

Posibilidades de la conversión implícita

La conversión implícita del tipo de datos DT no es posible.

Consulte también

DT (DATE_AND_TIME) (Página 2346)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión explícita de DT (Página 2536)
Activar o desactivar la verificación IEC (Página 2510)

Cadenas de caracteres

Conversión implícita de CHAR

Posibilidades de conversión implícita

La tabla siguiente muestra las posibilidades de la conversión implícita del tipo de datos CHAR:

Origen	Destino	Con verificación IEC	Sin verificación IEC	Explicación
CHAR	BOOL	-	-	Ninguna conversión implícita
	BYTE	-	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
	WORD	-	-	Ninguna conversión implícita
	DWORD	-	-	
	INT	-	-	
	DINT	-	-	
	REAL	-	-	
	TIME	-	-	
	S5TIME	-	-	
	DT	-	-	
	TOD	-	-	
	DATE	-	-	
	STRING	X	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.
x: Conversión posible -: Conversión no posible				

Consulte también

CHAR (carácter) (Página 2349)
Sinopsis de la conversión de tipos de datos (Página 2507)
Conversión explícita de CHAR (Página 2537)
Activar o desactivar la verificación IEC (Página 2510)

Conversión implícita de STRING

Posibilidades de conversión implícita

La conversión implícita del tipo de datos STRING no es posible.

Consulte también

STRING (Página 2350)
Sinopsis de la conversión de tipos de datos (Página 2507)
Conversión explícita de STRING (Página 2538)
Activar o desactivar la verificación IEC (Página 2510)

Conversión explícita

Números binarios

Conversión explícita de BOOL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos BOOL y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
BOOL	BYTE	X	En el tipo de datos de destino solamente se activa el LSB (Least Significant Bit). La salida de habilitación ENO es siempre "1".	CONVERT
	WORD	X		
	DWORD	X		
	INT	X		BOOL_TO_INT
	DINT	X		BOOL_TO_DINT
	REAL	-	Ninguna conversión explícita	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

Conversión implícita (Página 2510)
Sinopsis de la conversión de tipos de datos (Página 2507)
BOOL (bit) (Página 2325)

Secuencias de bits

Conversión explícita de BYTE

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos BYTE y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
BYTE	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajustará según el LSB y la salida de habilitación ENO será "0".	CONVERT
	WORD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resto de los bits se ponen a "0".	CONVERT
	DWORD	X		
	INT	X		
	DINT	X		
	REAL	-	Ninguna conversión explícita	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	-		

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. El resto de los bits se ponen a "0".	CONVERT
x: conversión posible - : conversión no posible				

Consulte también

BYTE (bytes) (Página 2326)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de BYTE (Página 2512)

Conversión explícita de WORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos WORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
WORD	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	CONVERT
	BYTE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si se rebasa el rango de valores permitido del tipo de datos de destino, la salida de habilitación ENO se pone a "0". En ese caso, el resultado de la conversión no es válido.	
	DWORD	X		
	INT	X		
	DINT	X		

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	REAL	-	Ninguna conversión explícita	-
	TIME	-		
	S5TIME	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	MOVE
	DT	-	Ninguna conversión explícita	-
	TOD	-		
	DATE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	MOVE
	STRING	-	Ninguna conversión explícita	-
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	WORD_TO_CHAR
	BLOCK_DB	X	El patrón de bits de WORD se interpreta como número del bloque de datos.	WORD_TO_BLOCK_DB
WORD_BCD[1)]	INT	X	Si se rebasa el rango de valores permitido del tipo de datos de destino, o si existe una tétrada no válida en el rango A..F, la salida de habilitación ENO se pone a "0". En ese caso, el resultado de la conversión no es válido.	WORD_BCD_TO_INT
BCD[1)]	INT	X		BCD_TO_INT

x: Conversión posible

- : Conversión no posible

[1)] El valor que debe convertirse tiene el tipo de datos WORD y se aplica como valor codificado en BCD entre -999 y +999. Tras la conversión, el resultado es un número entero (representación binaria) del tipo INT.

Consulte también

WORD (Página 2327)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de WORD (Página 2513)

Conversión explícita de DWORD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DWORD y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
DWORD	BOOL	X	Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	CONVERT
	BYTE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	
	WORD	X		

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción

	INT	X	ENO = TRUE
			#int1 := DWORD_TO_INT(16#FFFF_FFFF); // -1 bis
			#int1 := DWORD_TO_INT(16#FFFF_8000); // -32768
			#int1 := DWORD_TO_INT(16#0); // 0 bis
			#int1 := DWORD_TO_INT(16#0000_7FFF); // 32767
			
			ENO = FALSE
			#int1 := DWORD_TO_INT(16#FFFF_7FFF); // -32769
			#int1 := DWORD_TO_INT(16#8000_0000); // -2147483648
			#int1 := DWORD_TO_INT(16#8000); // 32768 bis
			#int1 := DWORD_TO_INT(16#7FFF_FFFF); // 2147483647	

	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	MOVE
	REAL	X		
	TIME	X		
	S5TIME	-	Ninguna conversión explícita	-
	DT	-		
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	MOVE
	DATE	-	Ninguna conversión explícita	-
	STRING	-		
	CHAR	-		
DWORD_BCD[1)]	DINT	X	Si se rebasa el rango de valores permitido del tipo de datos de destino, o si existe una tétrada no válida en el rango A..F, la salida de habilitación ENO se pone a "0". En ese caso, el resultado de la conversión no es válido.	DWORD_BCD_TO_DINT

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
BCD[1)]	DINT	X		BCD_TO_DINT
x: conversión posible - : conversión no posible [1)] El valor que debe convertirse tiene el tipo de datos DWORD y se aplica como valor codificado en BCD entre -9999999 y +9999999. Tras la conversión, el resultado es un número entero (representación binaria) del tipo DINT.				

Consulte también

DWORD (Página 2327)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de DWORD (Página 2514)

Enteros

Conversión explícita de INT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos INT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
INT	BOOL	X	El valor se convierte internamente primero a WORD y a continuación a BOOL. Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	INT_TO_BOOL
	BYTE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si un valor negativo se convierte a un tipo de datos de destino sin signo o si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	CONVERT
	WORD	X		MOVE
	DWORD	X		
	DINT	X		CONVERT, ITD

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "1" se transforma en el valor "1.0" con la instrucción "Convertir valor" (CONVERT)).	CONVERT, SCALE, ITR
	TIME	-	Ninguna conversión explícita	-
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa con un signo inicial. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	S_CONV, CONVERT
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. En caso de convertir valores negativos o en caso de desbordamiento, la salida de habilitación ENO se pone a "0".	CONVERT
	BCD[1)]	X	Si se rebasa el rango de valores permitido del tipo de datos de destino, o si existe una tétrada no válida en el rango A..F, la salida de habilitación ENO se pone a "0". En ese caso, el resultado de la conversión no es válido.	INT_TO_BCD
	BCD_WORD [1)]	X		INT_TO_BCD_WORD

x: Conversión posible

- : Conversión no posible

[1)] El valor que debe convertirse tiene el tipo INT y se aplica como número entero con un valor entre -999 y +999. Tras la conversión, el resultado es un número codificado en BCD del tipo WORD.

Consulte también

INT (enteros de 16 bits) (Página 2331)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de INT (Página 2515)

Conversión explícita de DINT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DINT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DINT	BOOL	X	El valor se convierte internamente primero a DWORD y a continuación a BOOL. Pueden darse las siguientes posibilidades: • Cuando el valor de origen es "0", el tipo de datos de destino también es "0" y la salida de habilitación ENO es "1". • Si en el valor de origen solamente es "1" el LSB (Least Significant Bit), el tipo de datos de destino también es "1" y la salida de habilitación ENO es "1". • Si en el valor de origen hay bits distintos al LSB, el tipo de datos de destino se ajusta según el LSB y la salida de habilitación ENO es "0".	DINT_TO_BOOL
	BYTE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino. Si un valor negativo se convierte a un tipo de datos de destino sin signo o si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	CONVERT
	WORD	X		
	DWORD	X		MOVE
	INT	X		CONVERT
	REAL	X	El valor se convierte al formato del tipo de datos de destino (p. ej. el valor "1" se transforma en el valor "1.0" con la instrucción "Convertir valor" (CONVERT)).	CONVERT, DTR, SCALE
	TIME	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.	T_CONV, MOVE
	S5TIME	-	Ninguna conversión explícita	-
	DT	-		
	TOD	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	CONVERT
	DATE	X		
	STRING	X	El valor se convierte en una cadena de caracteres. La cadena de caracteres se representa con un signo inicial. Si se rebasa la longitud de la cadena de caracteres, la salida de habilitación ENO se pone a "0".	S_CONV, CONVERT

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	CHAR	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. En caso de convertir valores negativos o en caso de desbordamiento, la salida de habilitación ENO se pone a "0".	DINT_TO_CHAR
	BCD[1)]	X	Si se rebasa el rango de valores permitido del tipo de datos de destino, o si existe una tétrada no válida en el rango A..F, la salida de habilitación ENO se pone a "0". En ese caso, el resultado de la conversión no es válido.	DINT_TO_BCD
	BCD_DWORD[1)]	X		DINT_TO_BCD_DWORD
x: Conversión posible - : Conversión no posible				

Consulte también

DINT (enteros de 32 bits) (Página 2333)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita (Página 2510)

Números en coma flotante

Conversión explícita de REAL

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos REAL y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
REAL	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	-		
	DWORD	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino.	CONVERT, MOVE
	INT	X	El valor se convierte al tipo de datos de destino. El resultado de la conversión depende de la instrucción utilizada. Si durante la conversión se rebasa el rango permitido de valores del tipo de datos de destino o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0".	CONVERT, ROUND, RND, CEIL, RND+, FLOOR, RND-, TRUNC, UNSCALE
	DINT	X		

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	TIME	-	Ninguna conversión explícita	-
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	X	El valor se convierte en una cadena de caracteres. Si se rebasa la longitud de la cadena de caracteres o el valor a convertir es un número en coma flotante no válido, la salida de habilitación ENO se pone a "0".	S_CONV, CONVERT
	CHAR	-	Ninguna conversión explícita	-
x: Conversión posible - : Conversión no posible				

Consulte también

REAL (Página 2338)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita (Página 2510)

Temporizadores

Conversión explícita de TIME

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos TIME y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
TIME	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	-		
	DWORD	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.	CONVERT
	INT	-	Ninguna conversión explícita	-

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios al tipo de datos de destino. El resultado de la conversión muestra el tiempo en milisegundos.	T_CONV, CONVERT
	REAL	-	Ninguna conversión explícita	-
	S5TIME	X	El valor se convierte al formato del tipo de datos de destino. Si se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	T_CONV, CONVERT
	DT	-	Ninguna conversión explícita	-
	TOD	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

TIME (tiempo IEC) (Página 2343)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de TIME (Página 2517)

Conversión explícita de S5TIME

Posibilidades de la conversión explícita en KOP, FUP, AWL y GRAPH

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos S5TIME y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
S5TIME	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	X	El valor se convierte al formato del tipo de datos de destino.	S5TIME_TO_WORD
	DWORD	-	Ninguna conversión explícita	-
	INT	-		
	DINT	-		
	REAL	-		
	TIME	X	El valor se convierte al formato del tipo de datos de destino.	T_CONV, CONVERT
	DT	-	Ninguna conversión explícita	-
	TOD	-		

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
	DATE	-		
	STRING	-		
	CHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

S5TIME (tiempo) (Página 2341)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de S5TIME (Página 2518)

Fecha y hora

Conversión explícita de DATE

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DATE y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DATE	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	X	El valor se convierte al formato del tipo de datos de destino.	DATE_TO_WORD
	DWORD	-	Ninguna conversión explícita	-
	INT	X	El valor se convierte al formato del tipo de datos de destino.	DATE_TO_INT
	DINT	X		DATE_TO_DINT
	REAL	-	Ninguna conversión explícita	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	STRING	-		
	CHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

DATE (Página 2344)
Sinopsis de la conversión de tipos de datos (Página 2507)
Conversión implícita de DATE (Página 2519)

Conversión explícita de TOD

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos TOD y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
TOD	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	-		
	DWORD	X	El valor se convierte al formato del tipo de datos de destino.	TOD_TO_DWORD
	INT	-	Ninguna conversión explícita	-
	DINT	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	TOD_TO_DINT
	REAL	-	Ninguna conversión explícita	-
	TIME	X	El valor se convierte al formato del tipo de datos de destino.	TOD_TO_TIME
	S5TIME	-	Ninguna conversión explícita	-
	DT	-		
	DATE	-		
	STRING	-		
	CHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

TOD (TIME_OF_DAY) (Página 2345)
Sinopsis de la conversión de tipos de datos (Página 2507)
Conversión implícita de TOD (Página 2520)

Conversión explícita de DT

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos DT y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
DT	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	-		
	DWORD	-		
	INT	-		
	DINT	-		
	REAL	-		
	TIME	-		
	S5TIME	-		
	TOD	X	Durante la conversión, los datos de la hora se extraen del formato DTL y se transfieren al tipo de datos de destino.	T_CONV, CONVERT
	DATE	X	Durante la conversión, la información de la fecha se extrae del formato DTL y se transfiere al tipo de datos de destino.	
	STRING	-	Ninguna conversión explícita	-
	CHAR	-		
x: Conversión posible - : Conversión no posible				

Consulte también

DT (DATE_AND_TIME) (Página 2346)

Sinopsis de la conversión de tipos de datos (Página 2507)

Cadenas de caracteres

Conversión explícita de CHAR

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos CHAR y las correspondientes instrucciones:

Origen	Destino	Conversión	Explicación	Nemónicos de la instrucción
CHAR	BOOL	-	Ninguna conversión explícita	-
	BYTE	X	El patrón de bits del valor de origen se transfiere sin cambios y alineado a la derecha al tipo de datos de destino.	CONVERT
	WORD	X		
	DWORD	X		
	INT	X		
	DINT	X		
	REAL	-	Ninguna conversión explícita	-
	TIME	-		
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	STRING	X	El valor se convierte en el primer carácter de la cadena de caracteres (STRING). Si la longitud de la cadena de caracteres no está definida, se ajustará la longitud "1" tras la conversión. Si la longitud de la cadena de caracteres está definida, esta no cambia tras la conversión.	S_CONV, CONVERT
x: Conversión posible - : Conversión no posible				

Consulte también

CHAR (carácter) (Página 2349)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita de CHAR (Página 2521)

Conversión explícita de STRING

Posibilidades de conversión explícita

La tabla siguiente muestra las posibilidades de conversión explícita del tipo de datos STRING y las correspondientes instrucciones:

Origen	Destino	Conversión	Significado	Nemónicos de la instrucción
STRING	BOOL	-	Ninguna conversión explícita	-
	BYTE	-		
	WORD	-		
	DWORD	-		
	INT	X	La conversión comienza por el primer carácter de la cadena de caracteres (STRING) y termina en el fin de la cadena o bien en el primer carácter no permitido. Los caracteres siguientes están permitidos en la conversión: • Cifra • Signo • Punto El primer carácter de la cadena de caracteres puede ser un signo (+, -) o una cifra. Se ignoran los espacios iniciales. El punto actúa como separador en la conversión de números en coma flotante. La notación exponencial "e" o "E" no está permitida. A la izquierda del punto decimal se permite usar comas para separar los millares, ya que se ignora. Si la estructura de la cadena de caracteres no es válida para la conversión o se produce un desbordamiento, la salida de habilitación ENO se pone a "0".	S_CONV, CONVERT
	DINT	X		
	REAL	X		
	TIME	-	Ninguna conversión explícita	-
	S5TIME	-		
	DT	-		
	TOD	-		
	DATE	-		
	CHAR	X	El primer carácter de la cadena de caracteres (STRING) se transfiere al tipo de datos de destino. Si la cadena de caracteres está vacía, se escribirá el valor "0" en el tipo de datos de destino.	S_CONV, CONVERT
x: conversión posible - : conversión no posible				

Consulte también

STRING (Página 2350)

Sinopsis de la conversión de tipos de datos (Página 2507)

Conversión implícita (Página 2510)

Otras funciones de conversión

Otras funciones de conversión explícita

Posibilidades adicionales para la conversión explícita en SCL

La tabla siguiente muestra las posibilidades adicionales de conversión explícita en SCL y las correspondientes instrucciones:

Origen	Destino	Explicación	Nemónicos de la instrucción
WORD	BLOCK_DB	El patrón de bits de WORD se interpreta como número del bloque de datos.	WORD_TO_BLOCK_DB
BLOCK_DB	WORD	El número del bloque de datos se interpreta como patrón de bits de WORD.	BLOCK_DB_TO_WORD

9.1.1.6 Control del flujo de programa

Mecanismo EN/ENO

Principios básicos del mecanismo EN/ENO

Introducción

Al procesarse instrucciones pueden producirse errores de tiempo de ejecución que tienen como consecuencia la interrupción del programa. Para evitar dichas interrupciones, puede utilizarse el mecanismo EN/ENO. Este mecanismo puede usarse en dos niveles:

- Mecanismo EN/ENO para instrucciones individuales
- Mecanismo EN/ENO para llamadas de bloque

Mecanismo EN/ENO para instrucciones en KOP/FUP

En KOP y FUP, determinadas instrucciones poseen una entrada de habilitación EN (enable) y una salida de habilitación ENO (enable output).

Con la entrada de habilitación EN pueden establecerse condiciones para la ejecución de una instrucción. Las instrucciones solo se ejecutan si el estado lógico en la entrada de habilitación EN es "1".

Con la salida de habilitación ENO pueden consultarse errores en tiempo de ejecución en instrucciones y reaccionar a ellos.

La salida de habilitación ENO devuelve el estado lógico "1" si se cumplen las siguientes condiciones:

- No se ha producido ningún error durante la ejecución.

La salida de habilitación ENO devuelve el estado lógico "0" si se cumple una de las siguientes condiciones:

- La entrada EN devuelve el estado lógico "0".
- Se ha producido un error durante la ejecución.

El mecanismo EN/ENO se utiliza para las siguientes instrucciones básicas:

- Funciones matemáticas
- Transferencia
- Conversión
- Operaciones lógicas con palabras
- Desplazamiento y rotación

En KOP y FUP, la evaluación de la salida de habilitación ENO puede activarse y desactivarse de forma selectiva para instrucciones concretas a través del menú contextual.

Mecanismo EN/ENO para llamadas de bloque en KOP/FUP

Todos los bloques que se llaman en KOP o FUP reciben una entrada de habilitación EN y una salida de habilitación ENO al llamarlas. Esto vale para todos los bloques llamados, independientemente del lenguaje de programación en el que se hayan creado.

La entrada de habilitación EN se puede utilizar para llamar al bloque en función de determinadas condiciones. El bloque solo se ejecuta si el estado lógico en la entrada de habilitación EN es "1".

Con la salida de habilitación ENO puede consultarse el estado de error del bloque. Devuelve la señal "1" en cuanto se inicia el procesamiento del bloque. Si la salida ENO no se pone explícitamente a "0" en el código del programa, conservará la señal "1".
Sin embargo, puede ponerse explícitamente a "0" para notificar una información de error al bloque llamado. En KOP o FUP, la salida ENO se activa con la instrucción "RET: Retroceder".

Consulte también:

Ejemplo del mecanismo EN/ENO en KOP (Página 2542)

Ejemplo del mecanismo EN/ENO en FUP (Página 2543)

Mecanismo EN/ENO para AWL

En AWL, el mecanismo EN/ENO no es necesario para instrucciones básicas. Esta función se reproduce mediante secuencias de instrucciones específicas.

Los bloques que se llaman desde un bloque AWL no se dotan de los parámetros EN y ENO. Sin embargo, independientemente del lenguaje de programación en el que se hayan creado,

pueden transferir información de error al bloque AWL invocador a través del bit BR de la palabra de estado.

En AWL, el estado de error del bloque llamado puede evaluarse mediante una operación lógica del bit BR de la palabra de estado con el RLO. Devuelve la señal "1" en cuanto se inicia el procesamiento del bloque. Si no se pone explícitamente a "0" en el código del programa, conservará la señal "1".
Sin embargo, puede ponerse explícitamente a "0" para notificar una información de error al bloque invocante. En AWL, la información de error se activa con las instrucciones "SAVE" o "SPBNB".

Consulte también: Ejemplo de la emulación del mecanismo EN/ENO en AWL (Página 2544)

Mecanismo EN/ENO en SCL

En SCL, el uso del mecanismo EN/ENO para instrucciones es opcional. Se pueden activar con la propiedad de bloque "Asignar ENO automáticamente". Si la propiedad está activa, todos los bloques reciben implícitamente un tratamiento de error.

Con la entrada de habilitación EN puede realizarse una llamada de bloque condicional. La entrada de habilitación EN se utiliza como un parámetro de entrada normal en la lista de parámetros. Si EN devuelve señal "1" o si EN no se utiliza, se llama el bloque. Si EN devuelve señal "0", el bloque no se llama.

Nota

Cuando se llaman funciones en SCL, no se puede utilizar el mecanismo de habilitación mediante EN. En su lugar, utilice una instrucción IF para la llamada condicional de funciones.

Con la salida de habilitación ENO puede consultarse el estado de error del bloque. Si ENO devuelve señal "1", el bloque se ha procesado sin errores. Si ENO devuelve señal "0", se produjo un error durante el procesamiento. Para consultar el estado de la salida de habilitación, inserte en la lista de parámetros un parámetro de salida adicional con el nombre ENO en la llamada del bloque.

Consulte también: Ejemplo del mecanismo EN/ENO en SCL (Página 2543)

ENO en GRAPH (S7-1500)

El operando ENO de GRAPH permite evaluar si una acción se ha realizado correctamente. Para ello se puede activar la opción "Asignar ENO automáticamente".

- Active la opción en las propiedades del programa para que sea el ajuste predeterminado de bloques GRAPH nuevos.
- Active la opción en las propiedades de bloque para utilizarla en determinados bloques.

Si esta opción está activada se muestra el operando ENO al realizar una prueba con el estado del programa. El operando tiene el valor "TRUE" si la acción se ha realizado correctamente y "FALSE" en caso contrario.

El uso del operando ENO es posible en los casos siguientes:

- Al llamar bloques que disponen de una salida de habilitación ENO.
- En las instrucciones en las que pueden producirse errores, p. ej. conversiones o funciones matemáticas.

Mecanismo EN/ENO en errores de acceso a memoria y periferia

No es posible evaluar los errores de acceso a memoria y periferia por medio del mecanismo EN/ENO. Utilice para ello el tratamiento global de errores mediante OBs (S7-300/400 y S7-1200/1500) o el tratamiento local de errores mediante la instrucción "GetError" (solo S7-1200/1500). Si para una instrucción no se ha producido ningún error de acceso a memoria puede evaluar el ENO correspondiente.

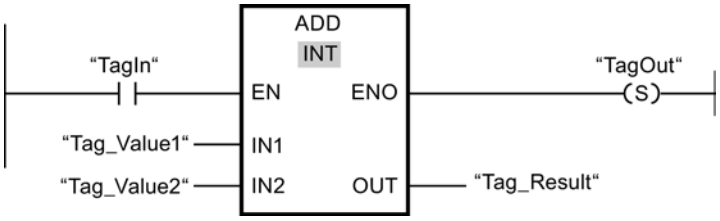
Consulte también

Activar y desactivar el mecanismo EN/ENO (Página 2699)

Ejemplo del mecanismo EN/ENO en KOP

Descripción

La figura siguiente muestra un sumador con cableado EN y ENO:



Después del contacto normalmente abierto, el RLO contiene el resultado de la operación lógica precedente:

- Si "TagIn" devuelve la señal "0", la suma no se realiza. Tanto EN como ENO tienen el estado lógico "0".
- Si "TagIn" devuelve la señal "1", la EN también es "1" y la suma se realiza. Si no se producen errores durante la ejecución de la instrucción, la salida ENO también devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

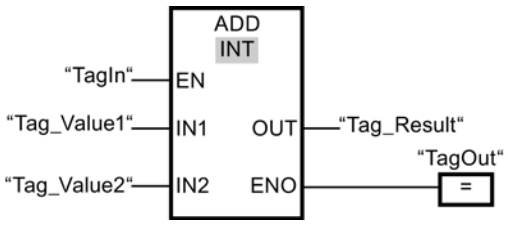
Principios básicos del mecanismo EN/ENO (Página 2539)

ADD: Sumar (Página 3400)

Ejemplo del mecanismo EN/ENO en FUP

Descripción

La figura siguiente muestra un sumador con conexión EN y ENO:



- Si "TagIn" devuelve la señal "1", la EN también es "1" y la suma se realiza. Si no se producen errores durante la ejecución de la instrucción, la salida ENO también devuelve el estado lógico "1" y se activa la salida "TagOut".
- Si "TagIn" devuelve la señal "0", la suma no se realiza. Tanto EN como ENO devuelven el estado lógico "0".

Consulte también

Principios básicos del mecanismo EN/ENO (Página 2539)

Ejemplo del mecanismo EN/ENO en SCL

Ejemplo del mecanismo EN/ENO para instrucciones básicas

Para utilizar el mecanismo EN/ENO para instrucciones en SCL, debe activarse la propiedad de bloque "Asignar ENO automáticamente". El siguiente ejemplo muestra el uso de la salida de habilitación ENO para la instrucción "a/b".

SCL

```
"MyoutputREAL" :=a/b;
IF ENO
    THEN "MyOutputBool":=1;
    ELSE "MyOutputBool":=0;
END_IF;
```

Si la instrucción "a/b" se ejecuta sin errores, MyOutputBool devuelve la señal "1".

Ejemplo del mecanismo EN/ENO para llamadas de bloque

El siguiente ejemplo muestra el uso de la salida de habilitación ENO para una llamada de bloque.

SCL

```
"MyDB"."MyFB"(EN:="MyTag1">"MyTag2",
              in1:="MyInputBool1",
              in2:="MyInputBool1",
              EN0=>"MyOutputBool");
```

Si MyTag1 no es mayor que MyTag2, la llamada del bloque no se procesa. Tanto EN como ENO tienen el estado lógico "0".

Si MyTag1 es mayor que MyTag2, EN devuelve la señal "1" y la llamada del bloque se ejecuta.

Si todas las instrucciones contenidas en MyFB se ejecutan sin errores, MyOutputBool devuelve la señal "1".

Consulte también

Principios básicos del mecanismo EN/ENO (Página 2539)

Ejemplo de la emulación del mecanismo EN/ENO en AWL

Descripción

El ejemplo siguiente muestra un segmento de programa para sumar valores con conexión EN y ENO:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual
JNB MyLABEL	// Evaluación de la entrada EN
	// Con RLO=0, saltar a la etiqueta "MyLABEL" y guardar el RLO actual en el BR
	// Si RLO ="1", ejecutar la instrucción siguiente
L"Tag_Input_2"	// Cargar el primer sumando
L"Tag_Input_3"	// Cargar el segundo sumando
+I	// Sumar valores
T "Tag_Result"	// Transferir la suma al operando "Tag_Result"
AN OV	// Consultar si se han producido errores
SAVE	// Transferir el estado lógico del RLO al bit BR
CLR	/// Poner RLO a "0"
MyLABEL: A BR	// Etiqueta "MyLABEL"
	// Consultar BR y combinarlo lógicamente a Y con el RLO

AWL	Explicación
= "Tag_Output"	// Asignar el estado lógico del RLO al operando"Tag_Output"

La consulta del operando "A "Tag_Input_1"" devuelve el resultado de la operación lógica precedente (RLO). La instrucción "Saltar si RLO = 0 y guardar RLO (JNB)" guarda el RLO en el BR. Además, la instrucción "Saltar si RLO = 0 y guardar RLO" evalúa el RLO y, en función de la evaluación, ejecuta una de las siguientes acciones:

- Si el RLO es "0", la ejecución del programa continúa en la etiqueta "MyLABEL" con la consulta del BR. No se realiza la suma. Se asigna el RLO actual al operando"Tag_Output".
- Si el RLO es "1", se realiza la suma. Consultando el bit de desbordamiento (OV) se determina si se ha producido un error en la suma. El resultado de la consulta se guarda en el BR. La operación "CLR" pone el RLO de nuevo a "0". A continuación se consulta si el estado lógico del BR es "1" y se combina el resultado lógicamente con Y con el RLO actual. El resultado se asigna al operando "Tag_Output". El estado lógico del BR y del operando "Tag_Output" indica si la suma se ha efectuado sin errores.

Consulte también

Principios básicos del mecanismo EN/ENO (Página 2539)

Palabra de estado

Principios básicos de la palabra de estado

Descripción

La palabra de estado agrupa bits de estado que son activados por la CPU durante el procesamiento digital o bien utilizados por la misma para controlar operaciones lógicas binarias. Los bits de estado se pueden consultar e influenciar de forma concreta.

La tabla siguiente muestra el orden de los bits de estado en la palabra de estado:

	Palabra de estado									
Número de bit	15-9	8	7	6	5	4	3	2	1	0
Contenido	0	BR	CC1	CC0	OV	OS	OR	STA	RLO	/FC

Los bits de estado /FC, RLO, STA, OR y BR son indicadores binarios que utiliza la CPU en las instrucciones binarias. Los bits de estado OS, OV, CC0 y CC1 son indicadores digitales que muestran principalmente resultados de funciones matemáticas.

/FC (primera consulta)

El estado lógico del bit /FC controla la ejecución de una cadena de operación lógica.

Una cadena de operación lógica comienza con el estado lógico "0" del bit /FC y una instrucción binaria de consulta (primera consulta). La primera consulta corresponde a la primera

operación lógica con bits o bien consulta binaria en una red. Esta pone el bit /FC a "1". La cadena de operación lógica finaliza con una asignación binaria de valores (p. ej. con la instrucción "Activar"), con un salto condicional o un cambio de bloque. Estos ponen el bit /FC a "0".

RLO (resultado lógico)

El bit de estado RLO es la memoria intermedia en las operaciones lógicas binarias.

La CPU transfiere el resultado de la consulta al RLO en la primera consulta. En todas las consultas siguientes, el resultado de la consulta se combina lógicamente con el RLO guardado y el resultado se deposita nuevamente en el RLO.

El RLO se puede activar, desactivar, negar o guardar en el resultado binario (BR) a través de las instrucciones correspondientes.

Con el RLO se controlan instrucciones de almacenamiento, temporización y contaje y se ejecutan determinadas instrucciones de salto.

STA (estado)

El bit de estado STA se corresponde con el estado lógico del operando binario consultado.

El estado de una operación lógica, la cual permite el acceso de lectura en la memoria, siempre es igual al valor del bit direccionado. El estado de una operación lógica, la cual permite el acceso de lectura a la memoria (p. ej. las instrucciones "Activar" o "Desactivar"), es igual al valor del bit en el que la instrucción escribe. En caso de no escribir, el estado es igual al valor del bit direccionado. El bit de estado no tiene ningún significado para las instrucciones de operación lógica que no acceden a la memoria. Estas instrucciones ponen el bit de estado a "1". La instrucciones no consultan el bit de estado. Este simplemente se calcula a través de variables de programa al mostrar el estado online.

OR (bit OR)

El bit de estado OR se requiere cuando se ejecuta una operación lógica Y antes de una operación lógica O.

El bit OR se activa cuando el RLO de la operación lógica Y es "1". De esta manera se anticipa el resultado de la operación lógica O. Cualquier otra instrucción binaria desactiva el bit OR.

OS (desbordamiento con memoria)

El bit de estado OS guarda una activación del bit de estado OV.

Cuando la CPU activa el bit de estado OV también activa el bit de estado OS. No obstante, mientras la siguiente instrucción ejecutada correctamente vuelve a desactivar el bit OV, el bit de estado OS permanece activado. Esto permite consultar más adelante, dentro del ciclo actual de la CPU, un desbordamiento del margen numérico o p. ej. la utilización de números en coma flotante no válidos.

OV (desbordamiento)

El bit de estado OV muestra un desbordamiento del margen numérico o la utilización de números en coma flotante no válidos.

El bit de estado OV se puede influenciar por medio de funciones matemáticas, instrucciones de conversión y comparaciones de números en coma flotante.

CC0 y CC1 (códigos de condición)

Los bits de estado CC0 y CC1 informan sobre el resultado de las siguientes instrucciones:

- Instrucciones de comparación
- Funciones matemáticas
- Operaciones lógicas con palabras
- Instrucciones de desplazamiento y rotación

BR (resultado binario)

El bit de estado BR sirve para ejecutar el mecanismo EN/ENO para cuadros o como condición en algunas instrucciones de salto (AWL). El bit de estado BR puede influenciarse a través de instrucciones determinadas (p. ej. SAVE).

Activar el bit de estado

Activar el bit de estado en operaciones lógicas con bits

Descripción

Las instrucciones binarias influyen en los bits de estado OR, STA, RLO y /FC.

Activar los bits de estado en KOP

La siguiente tabla muestra la activación de los bits de estado en el lenguaje de programación KOP:

Instrucción		Bit de estado			
Nemónicos	Título	OR	STA	RLO	/FC
-| |-	Contacto normalmente abierto	x	x	x	1
-| / |-	Contacto normalmente cerrado	x	x	x	1
-|NOT|-	Invertir RLO	-	1	x	-
-( )-	Asignación	0	x	-	0
-(R)-	Desactivar salida	0	x	-	0
-(S)-	Activar salida	0	x	-	0
SR	Flipflop de activación/desactivación	x	x	x	1
RS	Flipflop de desactivación/activación	x	x	x	1

Instrucción		Bit de estado			
Nemónicos	Título	OR	STA	RLO	/FC
-|P|-	Consultar flanco de señal ascendente de un operando	0	1	x	1
-|N|-	Consultar flanco de señal descendente de un operando	0	1	x	1
P_TRIG	Consultar flanco de señal ascendente del RLO	0	x	x	1
N_TRIG	Consultar flanco de señal descendente del RLO	0	x	x	1
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.					

Activar los bits de estado en FUP

La siguiente tabla muestra la activación de los bits de estado en el lenguaje de programación FUP:

Instrucción		Bit de estado			
Nemónicos	Título	OR	STA	RLO	/FC
&	Operación lógica Y	x	x	x	1
<=1	Operación lógica O	x	x	x	1
X	Operación lógica O-exclusiva	x	x	x	1
-|	Insertar entrada binaria	-	1	x	-
-o|	Invertir RLO	-	1	x	-
-( )-	Asignación	0	x	-	0
R	Desactivar salida	0	x	-	0
S	Activar salida	0	x	-	0
SR	Flipflop de activación/desactivación	x	x	x	1
RS	Flipflop de desactivación/activación	x	x	x	1
P	Consultar flanco de señal ascendente de un operando	0	1	x	1
N	Consultar flanco de señal descendente de un operando	0	1	x	1
P_TRIG	Consultar flanco de señal ascendente del RLO	0	x	x	1
N_TRIG	Consultar flanco de señal descendente del RLO	0	x	x	1
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.					

Activar los bits de estado en AWL

La siguiente tabla muestra la activación de los bits de estado en el lenguaje de programación AWL:

Instrucción		Bit de estado			
Nemónicos	Título	OR	STA	RLO	/FC
A	Operación lógica Y	0	x	x	1
AN	Operación lógica Y negada	0	x	x	1
O	Operación lógica O	0	x	x	1
ON	Operación lógica O negada	0	x	x	1
X	Operación lógica O-exclusiva	0	x	x	1
XN	Operación lógica O-exclusiva negada	0	x	x	1
O	Operación lógica O de funciones Y	0	1	-	0
A(	Operación lógica Y con abrir paréntesis	0	1	-	0
AN(	Operación lógica Y con abrir paréntesis	0	1	-	0
O(	Operación lógica O con abrir paréntesis	0	1	-	0
ON(	Operación lógica O negada con abrir paréntesis	0	1	-	0
X(	Operación lógica O-exclusiva con abrir paréntesis	0	1	-	0
XN(	Operación lógica O-exclusiva negada con abrir paréntesis	0	1	-	0
=	Asignación	0	x	-	0
R	Desactivar salida	0	x	-	0
S	Activar salida	0	x	-	0
NOT	Invertir RLO	-	1	x	-
SET	Poner RLO a 1	0	1	1	0
CLR	Poner RLO a 0	0	0	0	0
SAVE	Guardar RLO en bit BR	-	-	-	-
FN	Consultar flanco de señal descendente del RLO	0	x	x	1
FP	Consultar flanco de señal ascendente del RLO	0	x	x	1
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.					

Activar el bit de estado en instrucciones de temporización y contaje

Descripción

Las instrucciones de temporización y contaje influyen en los bits de estado OR, STA, RLO y /FC.

Activar los bits de estado en KOP y FUP

La tabla siguiente muestra la activación de los bits de estado en los lenguajes de programación KOP y FUP:

Instrucción		Bit de estado				
Nemónicos	Título	OS	OR	STA	RLO	/FC
S_PULSE	Parametrizar y arrancar temporizador como impulso	-	x	x	x	1
S_PEXT	Parametrizar y arrancar temporizador como impulso prolongado	-	x	x	x	1
S_ODT	Parametrizar y arrancar temporizador como retardo a la conexión	-	x	x	x	1
S_ODTS	Parametrizar y arrancar temporizador como retardo a la conexión con memoria	-	x	x	x	1
S_OFFDT	Parametrizar y arrancar temporizador como retardo a la desconexión	-	x	x	x	1
SP	Arrancar temporizador como impulso	-	0	-	-	0
SE	Arrancar temporizador como impulso prolongado	-	0	-	-	0
SD	Arrancar temporizador como retardo a la conexión	-	0	-	-	0
SS	Arrancar temporizador como retardo a la conexión con memoria	-	0	-	-	0
SF	Arrancar temporizador como retardo a la desconexión	-	0	-	-	0
S_CUD	Parametrizar e incrementar/decrementar contador	-	x	x	x	1
S_CU	Parametrizar e incrementar contador	-	x	x	x	1
S_CD	Parametrizar y decrementar contador	-	x	x	x	1
SC	Poner contador al valor inicial	-	0	x	-	0
CU	Contador ascendente	-	0	x	-	0
CD	Contador descendente	-	0	-	-	0
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.						

Activar los bits de estado en AWL

La siguiente tabla muestra la activación de los bits de estado en el lenguaje de programación AWL:

Instrucción		Bit de estado				
Nemónicos	Título	OS	OR	STA	RLO	/FC
FR	Habilitar temporizador	-	0	-	-	0
R	Inicializar temporizador	-	0	-	-	0

Instrucción		Bit de estado				
Nemónicos	Título	OS	OR	STA	RLO	/FC
SP	Arrancar temporizador como impulso	-	0	-	-	0
SE	Arrancar temporizador como impulso prolongado	-	0	-	-	0
SD	Arrancar temporizador como retardo a la conexión	-	0	-	-	0
SS	Arrancar temporizador como retardo a la conexión con memoria	-	0	-	-	0
SF	Arrancar temporizador como retardo a la desconexión	-	0	-	-	0
FR	Habilitar contador	-	0	-	-	0
R	Poner contador a 0	-	0	-	-	0
S	Inicializar contador	-	0	-	-	0
CU	Contador ascendente	-	0	-	-	0
CD	Contador descendente	-	0	-	-	0
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.						

Activar los bits de estado en instrucciones con números enteros

Descripción

Las instrucciones con números enteros influyen en los bits de estado CC1, CC0, OV y OS de la palabra de estado.

Si el resultado es negativo, los bits de estado CC1 y CC0 se ponen a "0". Si el resultado es positivo, el bit de estado CC1 se pone a "1" y el bit de estado CC0 a "0". El bit de estado CC1 se pone a "0" y el bit de estado CC0 a "1" cuando el resultado es negativo.

Si se produce un desbordamiento del margen numérico, los bits de estado OV y OS se ponen a "1". Si se produce una división entre cero, los bits de estado CC1, CC0, OV y OS devuelven el estado lógico "0".

Las siguientes tablas muestran la activación de los bits de estado cuando se utilizan números enteros:

Rango válido	CC 1	CC0	OV	OS
0 (cero)	0	0	0	*
16 bits: -32768 <= resultado < 0 (número negativo) 32 bits: -2147483648 <= resultado < 0 (número negativo)	0	1	0	*
16 bits: 32767 >= resultado > 0 (número positivo) 32 bits: 2147483647 >= resultado > 0 (número positivo)	1	0	0	*
* El bit OS no se ve afectado por el resultado de la operación.				

Rango no válido	CC 1	CC 0	OV	OS
Rebase por defecto en suma 16 bits: resultado = - 65536 32 bits: resultado = - 4294967296	0	0	1	1
Rebase por defecto en multiplicación 16 bits: resultado < -32768 (número negativo) 32 bits: resultado < -2147483648 (número negativo)	0	1	1	1
Desbordamiento en suma y resta 16 bits: resultado > 32767 (número positivo) 32 bits: resultado > 2147483647 (número positivo)	0	1	1	1
Desbordamiento en multiplicación y división 16 bits: resultado > 32767 (número positivo) 32 bits: resultado > 2147483647 (número positivo)	1	0	1	1
Rebase por defecto en suma y resta 16 bits: resultado < -32768 (número negativo) 32 bits: resultado < -2147483648 (número negativo)	1	0	1	1
División por 0	1	1	1	1

Activar los bits de estado en instrucciones con números en coma flotante

Descripción

Las instrucciones con números en coma flotante influyen en los bits de estado CC1, CC0, OV y OS de la palabra de estado.

Si el resultado es negativo, los bits de estado CC1 y CC0 se ponen a "0". Si el resultado es positivo, el bit de estado CC1 se pone a "1" y el bit de estado CC0 a "0". El bit de estado CC1 se pone a "0" y el bit de estado CC0 a "1" cuando el resultado es negativo.

Si se produce un desbordamiento del margen numérico, los bits de estado OV y OS se ponen a "1". En caso de número en coma flotante no válido, los bits de estado CC1, CC0, OV y OS devuelven el estado lógico "0".

Las siguientes tablas muestran la activación de los bits de estado cuando se utilizan números en coma flotante:

Rango válido	CC 1	CC 0	OV	OS
+0, -0 (cero)	0	0	0	*
-3,402823E+38 < resultado < -1,175494E-38 (número negativo)	0	1	0	*
+1,175494E-38 < resultado < 3,402824E +38 (número positivo)	1	0	0	*
* El bit OS no se ve afectado por el resultado de la operación.				

Rango no válido	CC 1	CC 0	OV	OS
Rebase por defecto -1,175494E-38 < resultado < -1,401298E-45 (número negativo)	0	0	1	1
Rebase por defecto +1,401298E-45 < resultado < +1,175494E-38 (número positivo)	0	0	1	1
Desbordamiento Resultado < -3,402823E+38 (número negativo)	0	1	1	1
Desbordamiento Resultado > 3,402823E+38 (número positivo)	1	0	1	1
Número en coma flotante no válido u operación no admisible (valor de entrada fuera del rango de valores válido)	1	1	1	1

Activar el bit de estado en operaciones de comparación

Descripción

Las instrucciones de comparación influyen en los bits de estado CC1, CC0, OV y OS de la palabra de estado.

Las instrucciones de comparación activan los bits de estado CC1 y CC0 independientemente de la relación de comparación.

Si se utilizan números en coma flotante no válidos, los bits de estado CC1, CC0, OV y OS se ponen a "1".

La siguiente tabla muestra la activación de los bits de estado en instrucciones de comparación:

Resultado	CC1	CC0	OV	OS
Igual (ACU2 = ACU1)	0	0	0	*
Mayor (ACU2 > ACU1)	1	0	0	*
Menor (ACU2 < ACU1)	0	1	0	*
Número en coma flotante no válido	1	1	1	1
* El bit OS no se ve afectado por el resultado de la operación.				

Activar el bit de estado en instrucciones de control del programa

Descripción

Las instrucciones de control del programa influyen en los bits de estado OS, OR, STA, RLO y /FC.

Activar los bits de estado en KOP y FUP

La tabla siguiente muestra la activación de los bits de estado en los lenguajes de programación KOP y FUP:

Instrucción		Bit de estado				
Nemónicos	Título	OS	OR	STA	RLO	/FC
JMP	Saltar si RLO = 1	-	0	1	1	0
JMPN	Saltar si RLO = 0	-	0	1	1	0
RET	Retroceder	0	0	1	1	0
CALL	Llamar bloque	0	0	1	-	0
OPN	Abrir bloque de datos global	-	-	-	-	-
OPNI	Abrir bloque de datos de instancia	-	-	-	-	-
MCR<	Abrir áreas MCR	-	0	1	-	0
MCR>	Cerrar áreas MCR	-	0	1	-	0
MCRA	Activar área MCR	-	-	-	-	-
MCRD	Desactivar área MCR	-	-	-	-	-
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.						

Activar los bits de estado en AWL

La siguiente tabla muestra la activación de los bits de estado en el lenguaje de programación AWL:

Instrucción		Bit de estado				
Nemónicos	Título	OS	OR	STA	RLO	/FC
JU	Salto incondicional	-	-	-	-	-
JC	Saltar si RLO = 1	-	0	1	1	0
JCN	Saltar si RLO = 0	-	0	1	1	0
JCB	Saltar si RLO = 1 y guardar RLO	-	0	1	1	0
JNB	Saltar si RLO = 0 y guardar RLO	-	0	1	1	0
JBI	Saltar si BR = 1	-	0	1	-	0
JNBI	Saltar si BR = 1	-	0	1	-	0
JO	Saltar si OV = 1	-	-	-	-	-
JOS	Saltar si OS = 1	0	-	-	-	-
SPZ	Saltar si el resultado es cero	-	-	-	-	-
SPN	Saltar si el resultado no es cero	-	-	-	-	-
SPP	Saltar si el resultado es mayor que cero	-	-	-	-	-
SPM	Saltar si el resultado es menor que cero	-	-	-	-	-
JPZ	Saltar si el resultado es mayor o igual a cero	-	-	-	-	-
SPMZ	Saltar si el resultado es menor o igual a cero	-	-	-	-	-
SPU	Saltar si el resultado no es válido	-	-	-	-	-

Instrucción		Bit de estado				
Nemónicos	Título	OS	OR	STA	RLO	/FC
JL	Definir lista de saltos	-	-	-	-	-
LOOP	Bucle	-	-	-	-	-
OPN	Abrir bloque de datos global	-	-	-	-	-
AUFDI	Abrir bloque de datos de instancia	-	-	-	-	-
CALL	Llamar bloque	0	0	1	-	0
CC	Llamada condicional	0	0	1	1	0
UC	Llamada incondicional	0	0	1	-	0
BE	Fin de bloque	0	0	1	-	0
BEC	Fin de bloque condicional	x	0	1	1	0
BEU	Fin de bloque incondicional	0	0	1	-	0
MCR(	Abrir áreas MCR	-	0	1	-	0
)MCR	Cerrar áreas MCR	-	0	1	-	0
MCRA	Activar área MCR	-	-	-	-	-
MCRD	Desactivar área MCR	-	-	-	-	-
x: la instrucción puede escribir "1" o "0". 1: la instrucción pone el bit de estado a "1". 0: la instrucción pone el bit de estado a "0". -: el bit de estado no se ve influenciado.						

Activar el bit de estado en operaciones lógicas con palabras

Descripción

Las operaciones lógicas con palabras influyen en los bits de estado CC1, CC0 y OV de la palabra de estado.

Si en el resultado de una operación lógica con palabras todos los bits son "0", el bit de estado CC1 se pone a "0". Si por lo menos un bit del resultado es "1", el bit de estado CC1 se pone a "1".

La siguiente tabla muestra la activación de los bits de estado en operaciones lógicas con palabras:

Resultado	CC1	CC0	OV
Cero	0	0	0
No cero	1	0	0

Activar el bit de estado en instrucciones de desplazamiento y rotación

Descripción

Las instrucciones de desplazamiento y rotación influyen en los bits de estado CC1, CC0 y OV de la palabra de estado.

En las operaciones de desplazamiento y rotación, el estado lógico del bit desplazado o rotado en último lugar se transfiere al bit de estado CC1. Los bits de estado CC0 y OV se resetean.

La siguiente tabla muestra la activación del bit de estado en instrucciones de desplazamiento y rotación:

Bit desplazado hacia fuera o rotado	CC1	CC0	OV
"0"	0	0	0
"1"	1	0	0
Con número de desplazamiento o rotación 0	*	*	*
* El bit no se ve afectado por el resultado de la operación.			

Consultar el bit de estado en AWL

Consultar el bit de estado con instrucciones de operaciones lógicas con bits

Descripción

El estado lógico de los bits en la palabra de estado se consulta con las siguientes operaciones lógicas con bits:

- A: Operación lógica Y
- AN: Operación lógica Y negada
- O: Operación lógica O
- ON: Operación lógica O negada
- X: Operación lógica O-exclusiva
- XN: Operación lógica O-exclusiva negada

La tabla siguiente muestra los operandos que pueden utilizarse para consultar los estados lógicos de los bits en la palabra de estado:

Operando	Consulta de la palabra de estado
== 0	Consulta de ((CC 0 = 0) Y (CC 1 = 0))
<> 0	Consulta de CC 0 <>CC 1
> 0	Consulta de ((CC 0 = 0) Y (CC 1 = 1))
< 0	Consulta de ((CC 0 = 1) Y (CC 1 = 0))
>= 0	Consulta de CC 0 = 0
<= 0	Consulta de CC 1 = 0
UO	Consulta de ((CC 1 = 1) Y (CC 0 = 1))
OV	Consulta de OV = 1
OS	Consulta de OS = 1
BR	Consulta de BR = 1

Consulte también

A: Operación lógica Y (Página 6158)
AN: Operación lógica Y negada (Página 6159)
O: Operación lógica O (Página 6161)
ON: Operación lógica O negada (Página 6162)
X: Operación lógica O-exclusiva (Página 6164)
XN: Operación lógica O-exclusiva negada (Página 6165)
O: Operación lógica O de funciones Y (Página 6167)
A(: Operación lógica Y con abrir paréntesis (Página 6168)
AN(: Operación lógica Y negada con abrir paréntesis (Página 6169)
O(: Operación lógica O con abrir paréntesis (Página 6170)
ON(: Operación lógica O negada con abrir paréntesis (Página 6171)
X(: Operación lógica O-exclusiva con abrir paréntesis (Página 6172)
XN(: Operación lógica O-exclusiva negada con abrir paréntesis (Página 6173)
): Cerrar paréntesis (Página 6175)
=: Asignación (Página 6176)
R: Desactivar (Página 6177)
S: Activar (Página 6179)
NOT: Invertir RLO (Página 6180)
SET: Poner RLO a 1 (Página 6181)
CLR: Poner RLO a 0 (Página 6182)
SAVE: Guardar RLO en bit BR (Página 6182)
FN: Consultar flanco de señal descendente del RLO (Página 6184)
FP: Consultar flanco de señal ascendente del RLO (Página 6185)

Consultar el bit de estado con instrucciones de salto

Descripción

La tabla siguiente muestra la evaluación de los bits de estado utilizando las instrucciones de salto:

RLO	BR	CC1	CC0	OV	OS	Nemónicos de la instrucción ejecutada
1	-	-	-	-	-	JC, JCB
0	-	-	-	-	-	JCN, JNB
-	1	-	-	-	-	JBI
-	0	-	-	-	-	JNBI

RLO	BR	CC1	CC0	OV	OS	Nemónicos de la instrucción ejecutada
-	-	0	0	-	-	JZ, JMZ, JPZ
-	-	1	0	-	-	JN, JP, JPZ
-	-	0	1	-	-	JN, JM, JMZ
-	-	1	1	-	-	SPU
-	-	-	-	1	-	JO
-	-	-	-	-	1	JOS

La tabla siguiente muestra la relación entre los bits de estado CC1 y CC0 y las instrucciones de salto condicionales:

CC1	CC0	Resultado	Nemónicos de la instrucción ejecutada
0	0	= 0	SPZ
1 o 0	0 o 1	<> 0	SPN
1	0	> 0	SPP
0	1	< 0	SPM
0 o 1	0	>= 0	JPZ
0	0 o 1	<= 0	SPMZ
1	1	UO (no válido)	SPU

Consulte también

JU: Salto incondicional (Página 6291)

JC: Saltar si RLO = 1 (Página 6292)

JCN: Saltar si RLO = 0 (Página 6293)

JCB: Saltar si RLO = 1 y guardar RLO (Página 6295)

JNB: Saltar si RLO = 0 y guardar RLO (Página 6296)

JBI: Saltar si BR = 1 (Página 6297)

JNBI: Saltar si BR = 0 (Página 6299)

JO: Saltar si OV = 1 (Página 6300)

JOS: Saltar si OS = 1 (Página 6301)

JZ: Saltar si el resultado es cero (Página 6303)

JN: Saltar si el resultado no es cero (Página 6305)

JP: Saltar si el resultado es mayor que cero (Página 6307)

JM: Saltar si el resultado es menor que cero (Página 6309)

JPZ: Saltar si el resultado es mayor o igual a cero (Página 6310)

JMZ: Saltar si el resultado es menor o igual a cero (Página 6312)

JUO: Saltar si el resultado no es válido (Página 6314)

JL: Definir lista de saltos (Página 6316)

LOOP: Bucle (Página 6317)

Palabra de estado

Principios básicos de la palabra de estado

Descripción

La palabra de estado agrupa bits de estado que son activados por la CPU durante el procesamiento digital o bien utilizados por la misma para controlar operaciones lógicas binarias. Los bits de estado se pueden consultar e influenciar de forma concreta.

La tabla siguiente muestra el orden de los bits de estado en la palabra de estado:

	Palabra de estado									
Número de bit	15-9	8	7	6	5	4	3	2	1	0
Contenido	0	BR	CC1	CC0	OV	OS	0	0	0	0

Los bits de estado OS, OV, CC0 y CC1 son indicadores digitales que muestran principalmente resultados de funciones matemáticas.

OS (desbordamiento con memoria)

El bit de estado OS guarda una activación del bit de estado OV.

Cuando la CPU activa el bit de estado OV también activa el bit de estado OS. No obstante, mientras la siguiente instrucción ejecutada correctamente vuelve a desactivar el bit OV, el bit de estado OS permanece activado. Esto permite consultar más adelante, dentro del bloque actual de la CPU, un desbordamiento del margen numérico o p. ej. la utilización de números en coma flotante no válidos.

OV (desbordamiento)

El bit de estado OV muestra un desbordamiento del margen numérico o la utilización de números en coma flotante no válidos.

El bit de estado OV se puede influenciar por medio de funciones matemáticas, instrucciones de conversión y comparaciones de números en coma flotante.

CC0 y CC1 (códigos de condición)

Los bits de estado CC0 y CC1 informan sobre el resultado de las siguientes instrucciones:

- Instrucciones de comparación
- Funciones matemáticas
- Operaciones lógicas con palabras
- Instrucciones de desplazamiento y rotación

BR (resultado binario)

El bit de estado BR sirve para ejecutar el mecanismo EN/ENO para cuadros o como condición en algunas instrucciones de salto (AWL). El bit de estado BR puede influenciarse a través de instrucciones determinadas (p. ej. SAVE).

RLO (resultado lógico)

El bit de estado RLO es la memoria intermedia en las operaciones lógicas binarias y no forma parte de la palabra de estado.

La CPU transfiere el resultado de la consulta al RLO en la primera consulta. En todas las consultas siguientes, el resultado de la consulta se combina lógicamente con el RLO guardado y el resultado se deposita nuevamente en el RLO.

El RLO se puede activar o desactivar a través de las instrucciones correspondientes.

Con el RLO se controlan instrucciones de almacenamiento, temporización y contaje y se ejecutan determinadas instrucciones de salto.

Nota

Dado que los bits de estado en parte solo se necesitan para el procesamiento interno, únicamente se muestran cuando están disponibles para la posterior ejecución del programa.

Consulta y activación de los bits de estado en AWL

Consultar el bit de estado con instrucciones de operaciones lógicas con bits

Descripción

El estado lógico de los bits en la palabra de estado se consulta con las siguientes operaciones lógicas con bits:

- A: Operación lógica Y
- AN: Operación lógica Y negada
- O: Operación lógica O
- ON: Operación lógica O negada
- X: Operación lógica O-exclusiva
- XN: Operación lógica O-exclusiva negada

La tabla siguiente muestra los operandos que pueden utilizarse para consultar los estados lógicos de los bits en la palabra de estado:

Operando	Consulta de la palabra de estado
== 0	Consulta de ((CC 0 = 0) Y (CC 1 = 0))
<> 0	Consulta de CC 0 <>CC 1
> 0	Consulta de ((CC 0 = 0) Y (CC 1 = 1))
< 0	Consulta de ((CC 0 = 1) Y (CC 1 = 0))
>= 0	Consulta de CC 0 = 0
<= 0	Consulta de CC 1 = 0
UO	Consulta de ((CC 1 = 1) Y (CC 0 = 1))
OV	Consulta de OV = 1
OS	Consulta de OS = 1
BR	Consulta de BR = 1

Consulte también

A: Operación lógica Y (Página 4062)

AN: Operación lógica Y negada (Página 4063)

O: Operación lógica O (Página 4065)

ON: Operación lógica O negada (Página 4066)

X: Operación lógica O-exclusiva (Página 4068)

XN: Operación lógica O-exclusiva negada (Página 4069)

O: Operación lógica O de funciones Y (Página 4071)

A(: Operación lógica Y con abrir paréntesis (Página 4072)

AN(: Operación lógica Y negada con abrir paréntesis (Página 4073)

O(: Operación lógica O con abrir paréntesis (Página 4074)

ON(: Operación lógica O negada con abrir paréntesis (Página 4075)

X(: Operación lógica O-exclusiva con abrir paréntesis (Página 4076)

XN(: Operación lógica O-exclusiva negada con abrir paréntesis (Página 4078)

): Cerrar paréntesis (Página 4079)

=: Asignación (Página 4080)

R: Desactivar (Página 4082)

S: Activar (Página 4083)

NOT: Invertir RLO (Página 4084)

SET: Poner RLO a 1 (Página 4085)

CLR: Poner RLO a 0 (Página 4086)

SAVE: Guardar RLO en bit BR (Página 4087)

FN: Consultar flanco de señal descendente del RLO (Página 4088)

FP: Consultar flanco de señal ascendente del RLO (Página 4090)

Consultar el bit de estado con instrucciones de salto

Descripción

La tabla siguiente muestra la evaluación de los bits de estado utilizando las instrucciones de salto:

RLO	BR	CC1	CC0	OV	OS	Nemónicos de la instrucción ejecutada
1	-	-	-	-	-	JC, JCB
0	-	-	-	-	-	JCN, JNB
-	1	-	-	-	-	JBI
-	0	-	-	-	-	JNBI
-	-	0	0	-	-	JZ, JMZ, JPZ
-	-	1	0	-	-	JN, JP, JPZ
-	-	0	1	-	-	JN, JM, JMZ
-	-	1	1	-	-	SPU
-	-	-	-	1	-	JO
-	-	-	-	-	1	JOS

La tabla siguiente muestra la relación entre los bits de estado CC0 y CC1 y las instrucciones de salto condicionales:

CC1	CC0	Resultado	Nemónicos de la instrucción ejecutada
0	0	= 0	JZ
1 o 0	0 o 1	<> 0	JN
1	0	> 0	JP
0	1	< 0	JM
0 o 1	0	>= 0	JPZ
0	0 o 1	<= 0	JMZ
1	1	UO (no válido)	SPU

Consulte también

JU: Salto incondicional (Página 6291)

JC: Saltar si RLO = 1 (Página 6292)

JCN: Saltar si RLO = 0 (Página 6293)

JCB: Saltar si RLO = 1 y guardar RLO (Página 6295)

JNB: Saltar si RLO = 0 y guardar RLO (Página 6296)

JBI: Saltar si BR = 1 (Página 6297)

JNBI: Saltar si BR = 0 (Página 6299)

JO: Saltar si OV = 1 (Página 6300)

JOS: Saltar si OS = 1 (Página 6301)

JZ: Saltar si el resultado es cero (Página 6303)

JN: Saltar si el resultado no es cero (Página 6305)

JP: Saltar si el resultado es mayor que cero (Página 6307)

JM: Saltar si el resultado es menor que cero (Página 6309)

JPZ: Saltar si el resultado es mayor o igual a cero (Página 6310)

JMZ: Saltar si el resultado es menor o igual a cero (Página 6312)

JUO: Saltar si el resultado no es válido (Página 6314)

JL: Definir lista de saltos (Página 6316)

LOOP: Bucle (Página 6317)

Etiqueta (Página 4197)

JU: Salto incondicional (Página 4198)

JC: Saltar si RLO = 1 (Página 4199)

JCN: Saltar si RLO = 0 (Página 4201)

JCB: Saltar si RLO = 1 y guardar RLO (Página 4202)

JNB: Saltar si RLO = 0 y guardar RLO (Página 4203)

JBI: Saltar si BR = 1 (Página 4205)

JNBI: Saltar si BR = 0 (Página 4206)

JO: Saltar si OV = 1 (Página 4207)

JOS: Saltar si OS = 1 (Página 4208)

JZ: Saltar si el resultado es cero (Página 4210)

JN: Saltar si el resultado no es cero (Página 4212)

JP: Saltar si el resultado es mayor que cero (Página 4214)

JM: Saltar si el resultado es menor que cero (Página 4216)

JPZ: Saltar si el resultado es mayor o igual a cero (Página 4218)

JMZ: Saltar si el resultado es menor o igual a cero (Página 4220)

JUO: Saltar si el resultado no es válido (Página 4222)

JL: Definir lista de saltos (Página 4223)

LOOP: Bucle (Página 4225)

Activar el bit de estado en operaciones lógicas con palabras

Descripción

Las operaciones lógicas con palabras influyen en los bits de estado CC0, CC1 y OV de la palabra de estado. Los bits de estado se activan únicamente si se utilizan durante el desarrollo del programa.

Si en el resultado de una operación lógica con palabras todos los bits son "0", el bit de estado CC1 se pone a "0". Si por lo menos un bit del resultado es "1", el bit de estado CC1 se pone a "1".

La siguiente tabla muestra la activación de los bits de estado en operaciones lógicas con palabras:

Resultado	CC1	CC0	OV
Cero	0	0	0
No cero	1	0	0

Consulte también

AW: Operación lógica Y por palabras (Página 4243)

OW: Operación lógica O por palabras (Página 4245)

XOW: Operación lógica O-exclusiva por palabras (Página 4247)

AD: Operación lógica Y por palabras dobles (Página 4249)

OD: Operación lógica O por palabras dobles (Página 4251)

XOD: Operación lógica O-exclusiva por palabras dobles (Página 4253)

Master Control Relay

El MCR (Master Control Relay)

Descripción de la funcionalidad MCR

El Master Control Relay (MCR) activa o desactiva el flujo de señales en una parte del controlador en los controladores de contactos.

Nota

Observe la información del apartado "Indicaciones importantes sobre el uso de la función MCR (Página 2565)"

⚠ ADVERTENCIA

Para evitar el riesgo de daños personales o materiales, no utilice nunca la funcionalidad MCR para sustituir un Master Control Relay mecánico que esté conectado permanentemente y que sirva como dispositivo de emergencia o seguridad.

El MCR afecta a todas las instrucciones que reescriben un valor en la memoria. Si la dependencia MCR está activada, dichas instrucciones reaccionan independientemente de que previamente haya una operación lógica binaria o digital.

Para la realización del MCR se dispone de las instrucciones siguientes:

- MCRA: Activar área MCR
- MCR: Abrir áreas MCR
- MCR: Cerrar áreas MCR
- MCRD: Desactivar área MCR

Si el resultado lógico es "1" justo antes de abrir un área MCR, se activa la dependencia MCR para dicha área. Si el RLO es "0" al abrir el área, el procesamiento en la misma se realiza sin dependencia MCR.

Indicaciones importantes sobre el uso de la función MCR

⚠ PRECAUCIÓN

En los bloques en los que se ha activado el Master Control Relay con la instrucción "Activar área MCR" (MCRA), debe tenerse en cuenta lo siguiente:

- Si el MCR está desactivado, todas las asignaciones escribirán el valor "0" en las secciones del programa que se encuentran entre las instrucciones "Abrir área MCR" (MCR( ) y "Cerrar área MCR" ( )MCR).
- El MCR está desactivado cuando el resultado lógico (RLO) es "0" antes de una instrucción "Abrir área MCR" (MCR( ).

⚠ PELIGRO

STOP de la CPU o comportamiento indefinido del tiempo de ejecución.

Para calcular las direcciones el compilador accede en modo de escritura a los datos locales que se encuentran después de las variables temporales definidas en VAR_TEMP. Por consiguiente, las siguientes secuencias de comandos ponen el AS a STOP o provocan un comportamiento indefinido del tiempo de ejecución (véanse las explicaciones siguientes).

Accesos a parámetros formales

- Accesos a componentes de parámetros FC complejos del tipo STRUCT, tipo de datos PLC (UDT), ARRAY, STRING
- Accesos a componentes de parámetros FB complejos del tipo STRUCT, tipo de datos PLC (UDT), ARRAY, STRING del área InOut en un bloque multiinstancia.
- Accesos a parámetros de un FB multiinstancia si su dirección es mayor que 8180.0.
- El acceso a un parámetro del tipo BLOCK_DB dentro del FB multiinstancia abre el DB 0. Los posteriores accesos a datos ponen la CPU a STOP. Con TIMER, COUNTER, BLOCK_FC y BLOCK_FB también se utiliza siempre T 0, C 0, FC 0 y FB 0, respectivamente.

Transferencia de parámetros

Se realiza en las llamadas de bloque en las que se transfieren parámetros.

Solución

Libere los comandos citados de la dependencia MCR:

1. Desactive el Master Control Relay con la instrucción "Desactivar área MCR" (MCRD) antes de la instrucción o del segmento en cuestión.
2. Active el Master Control Relay con la instrucción "Activar área MCR" (MCRA) después de la instrucción o del segmento en cuestión.

9.1.2 Declarar variables PLC

9.1.2.1 Sinopsis de las tablas de variables PLC

Introducción

Las tablas de variables PLC contienen la definición de las variables PLC y constantes simbólicas válidas en toda la CPU. Para toda CPU que se crea en el proyecto se genera automáticamente una tabla de variables PLC. Además, se pueden crear tablas de variables propias para ordenar y agrupar las variables y constantes.

En el árbol del proyecto hay una carpeta "Variables PLC" para cada CPU del proyecto. En ella se encuentran las tablas siguientes:

- Tabla "Todas las variables"
- Tabla de variables estándar
- Opcionalmente: Otras tablas de variables personalizadas

Todas las variables

La tabla "Todas las variables" muestra una vista general de todas las variables PLC, constantes de usuario y constantes de sistema de la CPU. Esta tabla no puede borrarse ni moverse.

Tabla de variables estándar

Hay una tabla de variables estándar para cada CPU del proyecto. Ésta no puede borrarse ni moverse, ni es posible cambiar su nombre. La tabla de variables estándar contiene variables PLC, constantes de usuario y constantes de sistema. Todas las variables PLC pueden declararse en la tabla de variables estándar, o se pueden crear otras tablas de variables personalizadas según necesidad.

Tablas de variables personalizadas

Para cada CPU pueden crearse varias tablas de variables personalizadas para agrupar variables según las necesidades. A las tablas de variables personalizadas se les puede cambiar el nombre, pueden reunirse en grupos o borrarse. Las tablas de variables personalizadas pueden contener variables PLC y constantes de usuario.

Consulte también

Estructura de la tablas de variables PLC (Página 2567)

Variables (Página 2286)

Principios básicos de las constantes (Página 2288)

9.1.2.2 Estructura de la tablas de variables PLC

Introducción

Todas las tablas de variables PLC contienen una ficha para variables y otra para constantes de usuario. La tabla de variables estándar y la tabla "Todas las variables" contienen además una ficha "Constantes de sistema".

Estructura de la ficha "Variables"

En la ficha "Variables" se declaran las variables PLC globales que se requieren en el programa. La figura siguiente muestra la estructura de la ficha. El número de columnas visualizadas puede variar.

	Nombre	Tipo de datos	Dirección	Remanencia	Visible en HMI	Accesible desde HMI	Comentario
	Motor1	Bool	%Q3.1	☐	☑	☑	
	Motor2	Bool	%Q3.2	☐	☑	☑	
	Control	Bool	%I3.3	☐	☑	☑	

En la siguiente tabla se explica el significado de las distintas columnas. El número de columnas visualizadas puede variar. Las columnas pueden mostrarse u ocultarse según necesidad.

Columna	Significado
	Símbolo en el que se puede hacer clic para arrastrar la variable mediante Drag & Drop hasta un programa y utilizarla allí como operando.
Nombre	Nombre de la variable, unívoco en toda la CPU.
Tipo de datos	Tipo de datos de la variable.
Dirección	Dirección de la variable.
Remanencia	Marca la variable como remanente. Los valores de variables remanentes se conservan tras desconectar la alimentación.
Accesible desde HMI	Indica si HMI puede acceder a esta variable en tiempo de ejecución.
Visible en HMI	Indica si la variable está visible en la selección de operandos de HMI mediante un ajuste predeterminado.
Valor de observación	Valor de datos actual en la CPU. Esta columna se visualiza solo si existe una conexión online y se hace clic en el botón "Observar todo".
Tabla de variables	Muestra en qué tabla de variables se encuentra la declaración de variables. Esta columna solo se encuentra en la tabla "Todas las variables".
Comentario	Comentario para documentar la variable.

Estructura de las fichas "Constantes de usuario" y "Constantes de sistema"

En la ficha "Constantes de usuario" se definen las constantes simbólicas válidas en toda la CPU. En la ficha "Constantes de sistema" se visualizan las constantes que requiere el sistema. Las constantes de sistema pueden ser, p. ej., identificadores de hardware que sirven para identificar módulos.

La siguiente figura muestra la estructura de las dos fichas. El número de columnas mostradas puede variar.

	Nombre	Tipo de datos	Valor	Comentario
	Const_1	Bool	true	
	Const_2	Byte	12	
	Const_3	Bool	false	
	Const_4	Real	1.0	

En la siguiente tabla se explica el significado de las distintas columnas. Las columnas pueden mostrarse u ocultarse según necesidad.

Columna	Significado
	Símbolo en el que se puede hacer clic para arrastrar la variable mediante Drag & Drop hasta un segmento y utilizarla allí como operando.
Nombre	Nombre de la constante, unívoco en toda la CPU.
Tipo de datos	Tipo de datos de la constante.
Valor	Valor de la constante.
Tabla de variables	Muestra en qué tabla de variables se encuentra la declaración de constantes. Esta columna solo se encuentra en la tabla "Todas las variables".
Comentario	Comentario para documentar la constante.

Consulte también

Tipos de datos de hardware (Página 2371)

Variables (Página 2286)

Principios básicos de las constantes (Página 2288)

Tipos de datos (Página 2321)

Sinopsis de las tablas de variables PLC (Página 2566)

Mostrar y ocultar las columnas de la tabla (Página 2591)

Editar tablas (Página 322)

9.1.2.3 Reglas para variables PLC

Nombres de variables PLC permitidos

Caracteres permitidos

Para los nombres de las variables PLC rigen las siguientes reglas:

- Se admiten letras, cifras y caracteres especiales.
- No se permiten las comillas.

Nombres de variable unívocos

Los nombres de las variables PLC tienen que ser unívocos en toda la CPU, aunque las variables se encuentren en tablas de variables diferentes de una CPU. Un nombre ya utilizado para un bloque, otra variable PLC o una constante en la CPU no se puede utilizar para una variable PLC nueva. En la verificación de univocidad no se distingue entre mayúsculas y minúsculas.

Si vuelve a introducir un nombre ya asignado, se añadirá automáticamente un número correlativo al segundo nombre. Por ejemplo, si introduce por segunda vez el nombre "Motor", la segunda entrada se transformará en "Motor(1)".

Nombres de tabla unívocos

Los nombres de las tablas de variables PLC también deben ser unívocos en toda la CPU. Al crear tablas de variables PLC personalizadas se propone automáticamente un nombre unívoco.

Consulte también

Variables (Página 2286)

Direcciones y tipos de datos admisibles para variables PLC (Página 2570)

Introducir la declaración de variables PLC (Página 2575)

Palabras clave (Página 2280)

Direcciones y tipos de datos admisibles para variables PLC

Las direcciones de las variables PLC constan de la indicación del área de operandos y la dirección dentro de ella.

Las direcciones deben ser unívocas en toda la CPU. Si introduce una dirección ya asignada a una variable diferente, la dirección se destacará en color amarillo en ambas ubicaciones y aparecerá un aviso de error.

Áreas de operandos

La tabla siguiente muestra las posibles áreas de operandos. Los tipos de datos disponibles dependen de la CPU utilizada:

Área de operandos		Significado	Tipo de datos	Formato	Área de direcciones:		
Nemónicos internacionales	Nemónicos alemanes				S7-1200	S7-300/400	S7-1500
I	E	Bit de entrada	BOOL	I x.y E x.y	0.0..1023.7	0.0..65535.7	0.0..32767.7
I	E	Entrada (64 bits)	LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL, tipo de datos PLC	I x.0 E x.0	-	-	0.0..32760.0
IB	EB	Byte de entrada	BYTE, CHAR, SINT, USINT, tipo de datos PLC	IB x EB y	0..1023	0..65535	0..32767
IW	EW	Palabra de entrada	WORD, INT, UINT, DATE, S5TIME, tipo de datos PLC	IW x EW y	0..1022	0..65534	0..32766
ID	ED	Palabra doble de entrada	DWORD, DINT, UDINT, REAL, TIME, TOD, tipo de datos PLC	ID x ED y	0..1020	0..65532	0..32764
Q	A	Bit de salida	BOOL	Q x.y A x.y	0.0..1023.7	0.0..65535.7	0.0..32767.7
Q	A	Salida (64 bits)	LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL, tipo de datos PLC	Q x.0 A x.0	-	-	0.0..32760.0
QB	AB	Byte de salida	BYTE, CHAR, SINT, USINT, tipo de datos PLC	QB x AB y	0..1023	0..65535	0..32767
QW	AW	Palabra de salida	WORD, INT, UINT, DATE, S5TIME, tipo de datos PLC	QW x AW y	0..1022	0..65534	0..32766

Área de operandos		Significado	Tipo de datos	Formato	Área de direcciones:		
Nemónicos internacionales	Nemónicos alemanes				S7-1200	S7-300/400	S7-1500
QD	AD	Palabra doble de salida	DWORD, DINT, UDINT, REAL, TIME, TOD, tipo de datos PLC	QD x AD y	0..1020	0..65532	0..32764
M	M	Bit de marcas	BOOL	M x.y	0.0..8191.7	0.0..65535.7	0.0..16383.7
M	M	Marca (64 bits)	LREAL	M x.0	0.0..8184.0	-	0.0..16376.0
M	M	Marca (64 bits)	LWORD, LINT, ULINT, LTIME, LTOD, LDT	M x.0	-	-	0.0..16376.0
MB	MB	Byte de marcas	BYTE, CHAR, SINT, USINT	MB x	0..8191	0..65535	0..16383
MW	MW	Palabra de marcas	WORD, INT, UINT, DATE, S5TIME	MW x	0..8190	0..65534	0..16382
MD	MD	Palabra doble de marcas	DWORD, DINT, UDINT, REAL, TIME, TOD	MD x	0..8188	0..65532	0..16380
T	T	Función de tiempo (solo en S7-300/400)	Temporizadores	T n	-	0..65535	0..2047
C	Z	Función de contaje (solo en S7-300/400)	Contadores	Z n C n	-	0..65535	0..2047

Direcciones

La siguiente tabla muestra las posibles direcciones de las variables:

Tipo de datos	Dirección	Ejemplo
BOOL	Las variables del tipo BOOL se direccionan con un número de byte y un número de bit. Para cada área de operando, la numeración de los bytes empieza desde 0. La numeración de los bits cubre del 0 al 7.	Q 1.0
BYTE, CHAR, SINT, USINT	Las variables del tipo de datos BYTE, CHAR, SINT y USINT se direccionan con un número de byte.	MB 1

Tipo de datos	Dirección	Ejemplo
WORD, INT, UINT, DATE, S5TIME	Las variables del tipo de datos WORD, INT, UINT, DATE, S5TIME constan de dos bytes. Estas se direccionan con el número del byte más bajo.	IW 1
DWORD, DINT, UDINT, REAL, TIME, TOD	Las variables del tipo de datos DWORD, DINT, UDINT, REAL, TIME, TOD constan de cuatro bytes. Estas se direccionan con el número del byte más bajo.	QD 1
LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL	Las variables del tipo de datos LWORD, LINT, ULINT, LTIME, LTOD, LDT, LREAL se componen de ocho bytes. Se direccionan con el número del byte más bajo en el número de bit 0.	I 1.0

Nemónicos utilizados

Las direcciones que se introducen en la tabla de variables PLC se adaptan automáticamente a los nemónicos utilizados.

Consulte también

Ajustar los nemónicos (Página 2630)

Variables (Página 2286)

Nombres de variables PLC permitidos (Página 2570)

Introducir la declaración de variables PLC (Página 2575)

Vista general de los tipos de datos válidos (Página 2321)

9.1.2.4 Crear y gestionar tablas de variables PLC

Crear una tabla de variables PLC

En una CPU se pueden crear varias tablas de variables PLC personalizadas. Todas las tablas de variables deben tener un nombre unívoco en toda la CPU.

Requisitos

La vista del proyecto está abierta.

Procedimiento

Para crear una tabla de variables PLC nueva, proceda del siguiente modo:

1. En el árbol del proyecto, abra la carpeta "Variables PLC" situada debajo de la CPU.
2. Haga doble clic en la entrada "Agregar nueva tabla de variables".
   Se crea una nueva tabla de variables PLC con el nombre estándar "Tabla de variables_x".

3. En el árbol del proyecto, seleccione la tabla de variables PLC.
4. Elija el comando "Cambiar nombre" del menú contextual.
5. Introduzca un nombre unívoco en toda la CPU.

Resultado

Se crea una nueva tabla de variables PLC. En ella pueden declararse variables y constantes.

Consulte también

Sinopsis de las tablas de variables PLC (Página 2566)

Estructura de la tablas de variables PLC (Página 2567)

Declarar variables PLC (Página 2575)

Declarar constantes globales (Página 2583)

Editar variables PLC con editores externos (Página 2591)

Agrupar tablas de variables PLC

Las tablas de variables personalizadas de una CPU pueden reunirse en grupos. Sin embargo, la tabla de variables estándar y la tabla "Todas las variables" no pueden moverse a un grupo.

Requisitos

En la carpeta "Variables PLC" de la CPU se encuentran varias tablas de variables personalizadas.

Procedimiento

Para agrupar varias tablas de variables PLC, proceda del siguiente modo:

1. En el árbol del proyecto, seleccione la carpeta "Variables PLC" situada debajo de la CPU.
2. Elija el comando de menú "Insertar > Grupo".
   Se inserta un grupo nuevo con el nombre estándar "Grupo_x".
3. Seleccione el grupo recién insertado en el árbol del proyecto.
4. Elija el comando "Cambiar nombre" del menú contextual.
5. Asígnele al grupo un nombre unívoco en toda la CPU.
6. Arrastre al nuevo grupo con Drag & Drop las tablas que se van a agrupar.

Resultado

Las tablas de variables se reúnen en el nuevo grupo.

Consulte también

Sinopsis de las tablas de variables PLC (Página 2566)
Estructura de la tablas de variables PLC (Página 2567)
Declarar variables PLC (Página 2575)
Declarar constantes globales (Página 2583)

Abrir la tabla de variables PLC

Procedimiento

Para abrir la tabla de variables PLC de una CPU, proceda del siguiente modo:

1. En el árbol del proyecto, abra la carpeta "Variables PLC" situada debajo de la CPU.
2. Haga doble clic en la tabla de variables PLC de la carpeta.
3. Seleccione la ficha deseada en la esquina superior derecha.

Resultado

Se abre la tabla de variables PLC perteneciente a la CPU. En ella pueden declararse las variables y constantes necesarias.

Consulte también

Sinopsis de las tablas de variables PLC (Página 2566)
Estructura de la tablas de variables PLC (Página 2567)
Declarar variables PLC (Página 2575)
Declarar constantes globales (Página 2583)

9.1.2.5 Declarar variables PLC

Introducir la declaración de variables PLC

Declarar variables en la tabla de variables PLC

Requisitos

La ficha "Variables" de una tabla de variables PLC está abierta.

Procedimiento

Para definir variables PLC, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca el tipo de datos deseado en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
   Las direcciones que concuerdan con el tipo de datos se completan automáticamente.
3. Opcionalmente: En la columna "Dirección" haga clic en el botón de dirección e introduzca un identificador del operando, un tipo de operando, una dirección y un número de bit en el siguiente diálogo.
4. Opcionalmente: Introduzca un comentario en la columna "Comentario".
5. Repita los pasos 1 a 4 para todas las variables necesarias.

Consulte también: Direcciones y tipos de datos admisibles para variables PLC (Página 2570)

Comprobación de sintaxis

Cada vez que se efectúa una entrada se comprueba la sintaxis automáticamente. Si hay errores, se destacan en color rojo. No es necesario corregir estos errores inmediatamente, sino que es posible continuar con la edición y efectuar las correcciones necesarias en un momento posterior. No obstante, si la declaración de variables contiene errores de sintaxis y la variable se utiliza en el programa, éste no se puede compilar.

Consulte también

Editar variables PLC con editores externos (Página 2591)

Utilizar la función automática de autocompletar (Página 2637)

Nombres de variables PLC permitidos (Página 2570)

Declarar variables PLC en el editor de programación (Página 2577)

Editar tablas de variables PLC (Página 2588)

Estructura de la tablas de variables PLC (Página 2567)

Editar tablas (Página 322)

Declarar variables PLC en el editor de programación

Requisitos

- El editor de programación está abierto.

Procedimiento

Para declarar operandos como variables PLC globales, proceda del siguiente modo:

1. Inserte una instrucción en el programa.
   Las cadenas de caracteres "<???>", "<??.?>" o "..." representan comodines de operandos.
2. Reemplace un comodín de operando con el nombre de la variable PLC que se va a crear.
3. Seleccione el nombre de la variable.
   Si desea declarar más variables PLC, seleccione los nombres de todas las variables que desea declarar.
4. Elija el comando "Definir variable" del menú contextual.
   Se abre el cuadro de diálogo "Definir variable". Éste muestra una tabla de declaración en la que ya figura el nombre de la variable.
5. En la columna "Sección", haga clic en el botón de dirección y seleccione una de las entradas siguientes:
   - Global Memory
   - Global Input
   - Global Output
6. Introduzca la dirección, el tipo de datos y el comentario en las respectivas columnas.
   Consulte también: Direcciones y tipos de datos admisibles para variables PLC (Página 2570)
7. Si la CPU contiene varias tablas de variables PLC, se puede especificar en la columna "Tabla de variables PLC" en qué tabla se debe insertar la variable. Si no se indica nada en la columna, la nueva variable se insertará en la tabla de variables estándar.
8. Haga clic en el botón "Definir" para finalizar la entrada.

Resultado

La declaración de variables se escribe en la tabla de variables PLC con validez para todos los bloques de la CPU.

Consulte también

Nombres de variables PLC permitidos (Página 2570)

Editar tablas (Página 322)

Declarar variables en la tabla de variables PLC (Página 2575)

Utilizar la función automática de autocompletar (Página 2637)

Ajustar la remanencia de variables PLC

Comportamiento remanente de variables PLC

Variables PLC remanentes

Toda CPU dispone de un área de memoria cuyo contenido permanece inalterado incluso tras desconectar la tensión de alimentación. Dicha área se denomina área de memoria remanente.

Para evitar pérdidas de datos en caso de un fallo de alimentación, pueden depositarse determinadas variables PLC en esta área de memoria. El ajuste de remanencia de las variables PLC se define en la tabla de variables PLC.

En función de la familia de CPU, el área de memoria remanente puede alojar diferentes tipos de variables PLC. La tabla siguiente ofrece una vista general de las posibilidades de las diferentes CPU:

Tipo de CPU	Marcas remanentes	Temporizadores SIMATIC remanentes	Contadores SIMATIC remanentes
Serie S7-300/400	✓	-	-
Serie S7-1200	✓	-	-
Serie S7-1500	✓	✓	✓

Consulte también

Ajustar el comportamiento remanente de variables PLC (Página 2578)

Ajustar el comportamiento remanente de variables PLC

Introducción

En la tabla de variables PLC se puede determinar el ancho del área de memoria remanente para variables PLC. Todas las variables direccionadas en esta área de memoria se marcan entonces como remanentes. Si en la columna "Remanencia" de la tabla de variables PLC aparece una marca de verificación, significa que se ha ajustado el comportamiento remanente para la variable en cuestión.

Requisitos

La ficha "Variables" de la tabla de variables PLC está abierta.

Procedimiento

Para definir el ancho del área de memoria remanente de variables PLC, proceda del siguiente modo:

1. Haga clic en el botón "Remanencia" de la barra de herramientas.
   Se abre el cuadro de diálogo "Memoria remanente".
2. Determine el ancho del área de memoria remanente introduciendo en el campo de entrada el número de bytes, temporizadores o contadores remanentes.
3. Haga clic en el botón "Aceptar".

Resultado

Se define el ancho del área de memoria remanente. En la columna "Remanencia" de la tabla de variables se activa automáticamente una marca de verificación para todas las variables que se encuentren en el área de memoria remanente.

Consulte también

Comportamiento remanente de variables PLC (Página 2578)

Editar tablas (Página 322)

9.1.2.6 Agrupar variables PLC en estructuras para entradas y salidas

Información importante sobre variables PLC estructuradas

Uso de variables PLC estructuradas (S7-1200 a partir de V4/S7-1500)

Para que el programa tenga un diseño más claro es posible agrupar varias direcciones de entrada o de salida en una variable PLC de nivel superior. La variable PLC de nivel superior representa una estructura que contiene diferentes entradas o salidas relacionadas lógicamente. Al llamar el bloque se transfiere la variable de nivel superior, con lo que solo se requiere un parámetro de entrada o salida para todas las entradas o salidas relacionadas.

Modo de funcionamiento

Para crear una variable PLC estructurada se define primero un tipo de datos PLC (UDT). En él se deben declarar los elementos de datos necesarios y definir sus nombres y tipos de datos.

Seguidamente se cambia a la tabla de variables PLC, donde se crea la variable PLC de nivel superior. Seleccione su tipo de datos PLC como tipo de datos para la variable. Ahora, el sistema reserva un número concreto de direcciones de entrada o salida a partir de la dirección inicial de la variable de nivel superior. El número de direcciones reservadas depende de la longitud del tipo de datos PLC.

Si se llama un bloque que requiere las entradas o salidas reservadas para procesar el programa, hay que transferir la variable de nivel superior como parámetro de bloque.

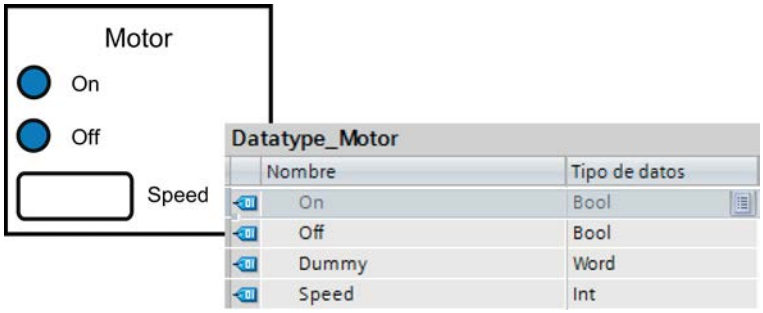
En el código del programa es posible direccionar las diferentes variables PLC como elementos de estructura.

En los siguientes capítulos encontrará una descripción detallada de los distintos pasos a seguir:

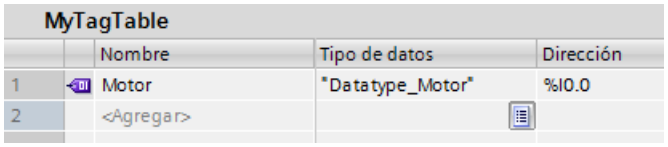
Ejemplo de aplicación

Existe la posibilidad de utilizar variables PLC estructuradas para agrupar entradas o salidas de un módulo de función. La figura siguiente muestra la representación esquemática de un motor. Para cada una de las tres entradas se ha creado un componente en el tipo de datos PLC "Datatype_Motor".

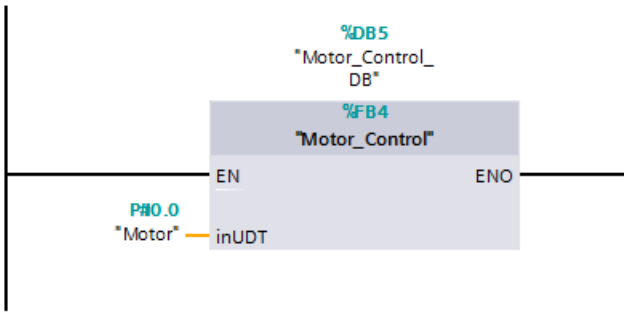
Las áreas de memoria de las variables declaradas no deben solaparse. En el ejemplo se ve que el componente "Speed" es del tipo de datos "Integer", con lo que debe empezar en una dirección de palabra. Por ese motivo se ha rellenado la primera palabra de entrada con la variable de relleno "Dummy". De este modo, "Speed" está en la segunda palabra de entrada.



La figura siguiente muestra la variable PLC de nivel superior "Motor" basada en el tipo de datos "Datatype_Motor". Mediante la declaración de "Motor" se reservan las direcciones EW0 y EW1 en el módulo de entrada.



La figura siguiente muestra la transferencia de la variable PLC "Motor" como parámetro de entrada del bloque "Motor_Control".



En el bloque "Motor_Control" se pueden direccionar los diferentes componentes de la variable.

Direccionamiento	Significado
"Motor"	Direccionamiento de la variable PLC de nivel superior.
"Motor".On	Direccionamiento de un componente de una variable PLC estructurada.
"Motor".On:P	Direccionamiento de una entrada o salida de periferia (PE o PA).

Reglas para utilizar variables PLC estructuradas

A la hora de crear y utilizar variables PLC estructuradas tenga en cuenta las reglas siguientes:

- Las variables PLC estructuradas pueden utilizarse en las áreas de operandos "entradas" y "salidas".
- En el área de marcas no se admiten variables PLC estructuradas.
- Desde HMI no es posible direccionar variables PLC estructuradas.

Al crear el tipo de datos PLC que debe utilizarse como base para una variable PLC tenga en cuenta las reglas siguientes:

- Las áreas de memoria de los diferentes elementos no deben solaparse.
  Consulte también: Direcciones y tipos de datos admisibles para variables PLC (Página 2570)
- No agrupe las entradas y salidas en un tipo de datos PLC, sino genere distintos tipos de datos PLC para entradas y salidas.
- No agrupe entradas o salidas de diferentes módulos en un tipo de datos PLC, pues no queda garantizado que las memorias imagen de proceso de los módulos se actualicen de forma síncrona.
- En los tipos de datos PLC subordinados se admiten todos los tipos de datos a excepción de "STRING".

Consulte también

Crear variables PLC estructuradas (Página 2582)

Direccionar variables globales (Página 2293)

Tipos de datos (Página 2321)

Crear variables PLC estructuradas

Reglas

A la hora de crear variables PLC estructuradas tenga en cuenta las reglas siguientes:

- Utilice tipos de datos PLC separados para las áreas de operandos "entradas" y "salidas".
- En el área de marcas no se admiten variables PLC estructuradas.
- No agrupe entradas o salidas de diferentes módulos en un tipo de datos PLC, pues no queda garantizado que las memorias imagen de proceso de los módulos se actualicen de forma síncrona.

Procedimiento

Para crear una variable PLC estructurada, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo tipo de datos" de la carpeta "Tipos de datos PLC" en el árbol del proyecto.
   Se crea y se abre una tabla de declaración nueva para la creación de un tipo de datos PLC.
2. Declare todos los componentes necesarios en el tipo de datos PLC. Se admiten todos los tipos de datos a excepción de "STRING".
3. Seleccione el tipo de datos PLC en el árbol del proyecto y elija el comando "Compilar > Software (solo cambios)" del menú contextual.
   El tipo de datos PLC se compila y puede utilizarse a continuación en la tabla de variables PLC.
   El programa debe compilarse de nuevo cuando se realizan cambios en los tipos de datos PLC existentes. De este modo se actualizan todas las ubicaciones del tipo de datos PLC.
4. Abra una tabla de variables PLC dentro de la misma CPU.
5. Declare una variable nueva o seleccione una existente.
6. Seleccione el tipo de datos PLC en la columna "Tipo de datos" y asígnelo a la variable PLC.
   La variable PLC adopta la estructura del tipo de datos PLC. Automáticamente se asigna una dirección adecuada. Las variables PLC estructuradas empiezan siempre en direcciones de palabra.
   En la tabla se representa solo el elemento de estructura más alto sin sus elementos subordinados.

Nota

Reglas de asignación y valores predeterminados

- Al declarar el tipo de datos PLC tenga en cuenta que las áreas de memoria de las diferentes variables no deben solaparse. Así, por ejemplo, las variables del tipo de datos "Integer" deben empezar en un límite de palabra. En caso necesario, inserte "variables de relleno" para impedir solapamientos.
  Consulte también: Direcciones y tipos de datos admisibles para variables PLC (Página 2570)
- No es posible asignar valores predeterminados a los diferentes componentes. Los valores introducidos en la columna "Valor predeterminado" no se evalúan. Por este motivo, es posible que algunas variables de los tipos de datos "DT" y "DTL" contengan valores no válidos.

Consulte también

Información importante sobre variables PLC estructuradas (Página 2579)

Direccionar variables globales (Página 2293)

Declarar variables PLC (Página 2575)

Programar tipos de datos PLC (Página 2967)

Tipos de datos (Página 2321)

9.1.2.7 Declarar constantes globales

Reglas para constantes globales

Caracteres permitidos

Los nombres de las constantes globales pueden constar de los siguientes caracteres:

- Se admiten letras, cifras y caracteres especiales.
- No se permiten las comillas.

Nombres de constante unívocos

Los nombres de las constantes globales tienen que ser unívocos en toda la CPU, aunque las constantes se encuentren en tablas de variables diferentes de una CPU. Un nombre ya utilizado para un bloque, una variable PLC u otra constante en la CPU no se puede utilizar para una constante nueva. En la verificación de univocidad no se distingue entre mayúsculas y minúsculas.

Si vuelve a introducir un nombre ya asignado, se añadirá automáticamente un número correlativo al segundo nombre. Por ejemplo, si introduce por segunda vez el nombre "Motor", la segunda entrada se transformará en "Motor(1)".

Tipos de datos permitidos

Para las constantes se permiten todos los tipos de datos soportados por la CPU, excepto los tipos de datos estructurados.

Valores permitidos

Se puede elegir, como valor de constante, cualquier valor situado dentro del rango de valores del tipo de datos indicado. Encontrará indicaciones sobre los rangos de valores en el capítulo "Tipos de datos".

Consulte también: Tipos de datos (Página 2321)

Consulte también

Principios básicos de las constantes (Página 2288)

Declarar constantes globales (Página 2584)

Constantes (Página 2288)

Declarar constantes globales

Introducción

Las constantes se declaran en la ficha "Constantes de usuario" de una tabla de variables PLC. Para declarar una constante es preciso introducir un nombre simbólico, un tipo de datos y un valor fijo. Se puede elegir, como valor de constante, cualquier valor situado dentro del rango de valores del tipo de datos indicado. Encontrará indicaciones sobre los rangos de valores en el capítulo "Tipos de datos".

Consulte también: Tipos de datos (Página 2321)

Procedimiento

Para declarar constantes, proceda del siguiente modo:

1. Abra una tabla de variables PLC.
2. Abra la ficha "Constantes de usuario".
   Se abre la tabla de constantes.
3. Introduzca un nombre de constante en la columna "Nombre".
4. Introduzca el tipo de datos deseado en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
5. En la columna "Valor", introduzca un valor de constante permitido para el tipo de datos seleccionado.
6. Si tiene observaciones sobre la constante, introduzca un comentario en la columna "Comentario". La introducción de un comentario es opcional.
7. Si desea declarar más constantes, ponga el cursor en la fila siguiente y repita los pasos 3 a 6.

Comprobación de sintaxis

Cada vez que se efectúa una entrada se comprueba la sintaxis automáticamente. Si hay errores, se destacan en color rojo. No es necesario corregir estos errores inmediatamente, sino que es posible continuar con la edición y efectuar las correcciones necesarias en un momento posterior. No obstante, si la declaración de variables contiene errores de sintaxis y la constante se utiliza en el programa, este no se puede compilar.

Consulte también

Abrir la tabla de variables PLC (Página 2575)

Insertar filas en la tabla de variables PLC (Página 2588)

Estructura de la tablas de variables PLC (Página 2567)

Reglas para constantes globales (Página 2583)

Editar tablas (Página 322)

Utilizar la función automática de autocompletar (Página 2637)

Constantes (Página 2288)

9.1.2.8 Editar propiedades

Editar las propiedades de variables PLC

Propiedades de variables PLC

Resumen

La tabla siguiente muestra una vista general de las propiedades de variables PLC. La representación de las propiedades puede variar en función del tipo de CPU.

Grupo	Propiedad	Descripción
General	Nombre	Nombre unívoco en la CPU.
	Tipo de datos	Tipo de datos de la variable.
	Dirección	Dirección de la variable.
	Remanente	Indica si la variable se encuentra en el área de memoria remanente.
	Comentario	Comentario acerca de la variable.
Historial	Fecha de creación	Fecha de creación de la variable (no modificable).
	Última modificación	Fecha de modificación de la variable (no modificable).
Utilización	Visible en HMI	Indica si la variable está visible en la lista desplegable de HMI mediante un ajuste predeterminado.
	Accesible desde HMI	Indica si HMI puede acceder a esta variable en tiempo de ejecución.

Consulte también

Editar las propiedades de variables PLC (Página 2586)

Editar las propiedades de variables PLC

Editar las propiedades en una tabla de variables PLC

Para editar las propiedades de una o más variables, proceda del siguiente modo:

1. En el árbol del proyecto, haga doble clic en la tabla de variables PLC en la que se encuentran las variables.
   Se abre la tabla de variables PLC.
2. Modifique las entradas en las columnas.

Editar las direcciones en el editor de programación

Para editar la dirección de una variable en el editor de programación, proceda del siguiente modo:

1. Seleccione el nombre de la variable.
2. Elija el comando "Reasignar variable" del menú contextual.
   Se abre el cuadro de diálogo "Reasignar variable". En él se visualiza una tabla de declaración.
3. Introduzca la dirección nueva en la columna "Dirección".
4. Haga clic en el botón "Modificar" para confirmar la entrada.

Editar nombres en el editor de programación

Para editar el nombre de una variable en el editor de programación, proceda del siguiente modo:

1. Seleccione el nombre de la variable.
2. Elija el comando "Cambiar nombre de la variable" del menú contextual.
   Se abre el cuadro de diálogo "Cambiar nombre de la variable". En él se visualiza una tabla de declaración.
3. Introduzca el nuevo nombre en la columna "Nombre".
4. Haga clic en el botón "Modificar" para confirmar la entrada.

Repercusión en el programa

En caso de modificar el nombre, el tipo de datos o la dirección de una variable, se actualizan automáticamente todas las ubicaciones de las variables en el programa.

Consulte también

Propiedades de variables PLC (Página 2585)

Editar propiedades de las constantes globales

Propiedades de las constantes globales

Vista general

La siguiente tabla proporciona una vista general de las propiedades de las constantes:

Grupo	Propiedad	Descripción
General	Nombre	Nombre unívoco en la tabla
	Tipo de datos	Tipo de datos de la constante
	Valor	Valor que se ha determinado para la constante. Este valor debe ser compatible con el tipo de datos indicado. Consulte también: Tipos de datos (Página 2321)
	Comentario	Comentario acerca de la constante
Historial	Fecha de creación	Fecha de creación de la constante (no modificable)
	Última modificación	Fecha de modificación de la constante (no modificable)

Editar propiedades de las constantes globales

Editar las propiedades en una tabla de variables PLC

Para editar las propiedades de una o más constantes, proceda del siguiente modo:

1. En el árbol del proyecto, haga doble clic en la tabla de variables PLC en la que se encuentran las constantes.
   Se abre la tabla de variables PLC.
2. Abra la ficha "Constantes de usuario".
3. Modifique las entradas de las columnas "Nombre", "Tipo de datos", "Valor" o "Comentario".

Repercusión en el programa

En caso de modificar el nombre, el tipo de datos o el valor de una constante, se actualizan automáticamente todas las ubicaciones de las constantes en el programa.

Consulte también

Editar tablas (Página 322)

9.1.2.9 Observar variables PLC

Observar variables PLC

Los valores de datos que adoptan actualmente las variables en la CPU pueden observarse directamente en la tabla de variables PLC.

Requisitos

Existe una conexión online con la CPU.

Procedimiento

Para observar los valores de datos, proceda del siguiente modo:

1. Abra una tabla de variables PLC.
2. Inicie la observación haciendo clic en el botón "Observar todos".
   En la tabla aparece la columna adicional "Valor de observación". En ella se visualizan los valores de datos actuales.
3. Para finalizar la observación, vuelva a hacer clic en el botón "Observar todos".

Nota

Además, existe la posibilidad de copiar las variables PLC en una tabla de observación o de forzado permanente para observarlas, forzarlas o forzarlas de manera permanente.

Consulte también

Estructura de la tablas de variables PLC (Página 2567)

Introducción al test con la tabla de observación (Página 3115)

Introducción al test con la tabla de forzado permanente (Página 3143)

Copiar entradas de la tabla de variables PLC (Página 2589)

9.1.2.10 Editar tablas de variables PLC

Insertar filas en la tabla de variables PLC

Procedimiento

Para insertar una fila encima de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila sobre la cual desea insertar una fila nueva.
2. Haga clic en el botón "Insertar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva sobre la seleccionada.

Consulte también

Editar tablas (Página 322)

Copiar entradas de la tabla de variables PLC

Las variables PLC se pueden copiar dentro de una tabla o en otras tablas.

Procedimiento

Para copiar una variable, proceda del siguiente modo:

1. Seleccione la variable que desea copiar.
   También puede seleccionar varias variables simultáneamente manteniendo pulsada la tecla <Ctrl> y haciendo clic consecutivamente en las variables que desea seleccionar, o bien en la primera y en la última.
2. Elija el comando "Copiar" del menú contextual.
3. Sitúe el cursor de inserción en la posición en la que desea insertar la variable.
4. Elija el comando "Pegar" del menú contextual.

O bien:

1. Seleccione la variable.
2. Mantenga pulsado el botón izquierdo del ratón.
3. Pulse simultáneamente la tecla <Ctrl>.
4. Arrastre la variable hasta el destino deseado.

Resultado

- La variable se copia en el destino.
- Si ocurre un conflicto de nombres, el nombre de la variable se complementará automáticamente con un número. Por ejemplo, "Variable" se transformará en "Variable(1)".
- Todas las demás propiedades de la variable se conservan.

Consulte también

Editar tablas (Página 322)

Borrar entradas de la tabla de variables PLC

Procedimiento

Para borrar elementos, proceda del siguiente modo:

1. Seleccione la fila que contiene el elemento que desea borrar. También puede seleccionar varias filas simultáneamente manteniendo pulsada la tecla <Ctrl> y haciendo clic consecutivamente en las filas que desea seleccionar, o bien en la primera y en la última.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Editar tablas (Página 322)

Ordenar las filas de las tablas de variables PLC

Las filas de las tablas se pueden ordenar de forma alfanumérica por nombre, tipo de datos o dirección.

Procedimiento

Para ordenar las filas de una tabla, proceda del siguiente modo:

1. Seleccione la columna por la cual desea ordenar.
2. Haga clic en el encabezado de la columna.
   La columna se ordena de forma ascendente.
   Una flecha hacia arriba indica el sentido de ordenación.
3. Para invertir el sentido de ordenación, haga clic en la flecha.
   La columna se ordena de forma descendente. Una flecha hacia abajo indica el sentido de ordenación.
4. Para restablecer la ordenación original, haga clic por tercera vez en el encabezado de la columna.

Consulte también

Editar tablas (Página 322)

Rellenar celdas automáticamente en la tabla de variables PLC

Es posible transferir el contenido de una o varias celdas de la tabla a las celdas situadas debajo, rellenando así las celdas consecutivas automáticamente.

Si se rellenan automáticamente las celdas de la columna "Nombre", cada uno de los nombres se complementará con un número correlativo. Por ejemplo, el nombre "Motor" se transforma en "Motor_1".

Si se rellenan automáticamente las celdas de la columna "Dirección", las direcciones aumentarán en función del tipo de datos indicado.

Procedimiento

Para rellenar celdas consecutivas automáticamente, proceda del siguiente modo:

1. Seleccione las celdas cuyo contenido desea transferir.
2. Haga clic en el símbolo de rellenar situado en la esquina inferior derecha de la celda.
   El puntero del ratón adopta la forma de una cruz con puntas de flecha.
3. Con el botón del ratón pulsado, arrastre el puntero hacia abajo hasta situarlo sobre las celdas que desea rellenar automáticamente.
4. Suelte el botón del ratón.
   Las celdas se rellenan automáticamente. Si ya existen entradas en las celdas que se van a rellenar automáticamente, aparece un cuadro de diálogo en el que se puede indicar si se desean sobrescribir las entradas existentes o insertar nuevas filas para las variables nuevas.

Consulte también

Editar tablas (Página 322)

Mostrar y ocultar las columnas de la tabla

Las columnas de las tablas pueden mostrase u ocultarse según necesidad.

Procedimiento

Para mostrar u ocultar las columnas de una tabla, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna.
2. Elija el comando "Mostrar/ocultar" del menú contextual.
   Aparece una selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación correspondiente.
4. Para ocultar una columna, desactive la casilla de verificación correspondiente.

Consulte también

Editar tablas (Página 322)

Editar variables PLC con editores externos

Bases para importar y exportar

Introducción

Las tablas de variables PLC pueden exportarse en un formato XLSX estandarizado y editarse con editores de tablas externos. También es posible importar al TIA Portal tablas de variables PLC que se crearon con editores de tablas externos.

Sobrescribir variables PLC y constantes existentes durante la importación

Durante la importación se sobrescriben las entradas ya existentes cuando tienen el mismo nombre que las entradas que se importarán.

Conexión con objetos existentes

Cuando en el proyecto existen ya referencias a variables PLC o constantes, dichas referencias se actualizan automáticamente durante la importación. La actualización se realiza tomando como base el nombre de las variables PLC y constantes.

Consulte también

Formato del archivo de exportación (Página 2592)

Exportación de variables PLC (Página 2593)

Importación de variables PLC (Página 2593)

Formato del archivo de exportación

Introducción

Al exportar tablas de variables PLC se genera un formato XSLX estandarizado que puede editarse con editores de tablas externos.

Dicho formato también es el previsto para la importación de tablas.

Formato del archivo de exportación

El nombre de la hoja es siempre "PLC Tags". Esta hoja puede contener las columnas mostradas. El orden de las columnas puede variar. No todas las columnas tienen que existir. Los valores que faltan se marcan con la indicación <no value> durante la importación.

Los títulos de columnas también tienen un nombre fijo que debe ser en inglés.

La siguiente tabla indica los contenidos previstos en las columnas individuales:

Elemento	Significado
Name	Nombre de la variable
Path	Grupo y nombre de la tabla de variables PLC
Data Type	La notación del tipo de datos se corresponde con la notación utilizada en la tabla de variables PLC.
Logical Address	La dirección puede estar indicada en nemotécnica alemana o internacional.
Comment	Comentario de libre elección
Hmi Visible	El valor previsto es TRUE o FALSE.
Hmi Accessible	El valor previsto es TRUE o FALSE.

Consulte también

Bases para importar y exportar (Página 2591)

Exportación de variables PLC (Página 2593)

Importación de variables PLC (Página 2593)

Exportación de variables PLC

Requisitos

Hay una tabla de variables PLC abierta.

Procedimiento

Para exportar variables PLC y constantes, proceda del siguiente modo:

1. Haga clic en el botón "Exportar" de la tabla de variables PLC.
   Se abre el cuadro de diálogo "Exportar a Excel".
2. Seleccione la ruta en la que se debe depositar el archivo de exportación.
3. Elija si quiere exportar variables y/o constantes.
4. Haga clic en el botón "Aceptar".

Resultado

Se crea el archivo de exportación. Los errores y las advertencias que se producen durante la exportación se muestran en la ficha "Información" de la ventana de inspección.

Consulte también

Bases para importar y exportar (Página 2591)

Formato del archivo de exportación (Página 2592)

Importación de variables PLC (Página 2593)

Importación de variables PLC

Requisitos

Hay una tabla que se corresponde con las especificaciones de formato.

Procedimiento

Para importar una tabla de variables PLC, proceda del siguiente modo:

1. Abra la tabla "Todas las variables".
2. Haga clic en el botón "Importar".
   Se abre el cuadro de diálogo "Importar de Excel".
3. Seleccione si quiere importar variables PLC y/o constantes.
4. Seleccione la tabla que desea importar.
5. Haga clic en el botón "Aceptar".

Resultado

Se importa la tabla de variables PLC.

Los errores y las advertencias que se producen durante la exportación se muestran en la ficha "Información" de la ventana de inspección.

Consulte también

Bases para importar y exportar (Página 2591)

Formato del archivo de exportación (Página 2592)

Exportación de variables PLC (Página 2593)

Editar variables PLC individuales con editores externos

Para editar variables PLC de forma individual en editores externos al TIA Portal, se pueden exportar y volver a importar mediante "Copiar y Pegar". Sin embargo, las variables estructuradas no se pueden copiar en un editor.

Requisitos

Hay una tabla de variables PLC y un editor externo abiertos.

Procedimiento

Para exportar variables PLC y volverlas a importar individualmente, proceda del siguiente modo:

1. Seleccione una o más variables PLC.
2. Elija el comando "Copiar" del menú contextual.
3. Cambie al editor externo y pegue la variable copiada.
4. Edite las variables de la forma necesaria.
5. Copie las variables en el editor externo.
6. Cambie nuevamente a la tabla de variables PLC.
7. Elija el comando "Pegar" del menú contextual.

Nota

Además existe la posibilidad de exportar o importar las variables PLC como datos masivos.

Consulte también: Editar variables PLC con editores externos (Página 2591)

9.1.3 Crear y gestionar bloques

9.1.3.1 Crear bloques

Carpeta de bloques

Función

El árbol del proyecto contiene la carpeta "Bloques de programa" en la que se pueden crear y gestionar los bloques siguientes:

- Bloques de organización (OB)
- Bloques de función (FB)
- Funciones (FC)
- Bloques de datos (DB)

Además, en la carpeta "Bloques de programa" se crea una subcarpeta "Bloques de sistema" con otra subcarpeta "Recursos de programa" al arrastrar por primera vez al programa una instrucción que ejerce de bloque de función en el programa. El bloque de datos de instancia del bloque de función interno del sistema se inserta así en la carpeta "Recursos de programa". Estos bloques de datos de instancia de la carpeta "Recursos de programa" se pueden mover, copiar, renombrar y borrar en cualquier carpeta. A su vez, los bloques de usuario también pueden moverse a la carpeta "Recursos de programa". Los bloques de la carpeta "Recursos de programa" que no sean necesarios para la ejecución del programa de usuario se eliminarán en el próximo proceso de compilación. Si, posteriormente, la carpeta "Recursos de programa" ya no contiene bloques, se borrará también junto con la carpeta "Bloques de sistema".

Para cada dispositivo se crea automáticamente un OB de ciclo y se inserta en la carpeta "Bloques de programa".

Consulte también

Crear funciones y bloques de función (Página 2597)

Crear bloques de datos (Página 2598)

Crear bloques de organización (Página 2596)

Utilizar bloques de librerías (Página 2599)

Crear bloques de organización

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para crear un bloque de organización, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo bloque".
   Se abre el diálogo "Agregar nuevo bloque".
2. Haga clic en el botón "Bloque de organización (OB)".
3. Seleccione el tipo del nuevo bloque de organización.
4. Introduzca un nombre para el nuevo bloque de organización.
5. Introduzca las propiedades del nuevo bloque de organización.
6. Haga clic en "Más información" para introducir más propiedades del nuevo bloque de organización.
   Se abre un área con campos de entrada adicionales.
7. Introduzca todas las propiedades deseadas.
8. Active la casilla de verificación "Agregar y abrir" si el bloque de organización debe abrirse inmediatamente después de crearse.
9. Confirme las entradas realizadas con "Aceptar".

Resultado

Se crea el nuevo bloque de organización. Éste se encuentra entonces en la carpeta "Bloques de programa" del árbol del proyecto. Algunos bloques de organización se pueden seguir parametrizando en la ventana de inspección o en la vista de dispositivos después de crearse. En la descripción del bloque de organización se puede leer si el bloque de organización creado recientemente tiene parámetros adicionales.

Consulte también

Bloques de organización (OB) (Página 2252)

Carpeta de bloques (Página 2595)

Crear funciones y bloques de función (Página 2597)

Crear bloques de datos (Página 2598)

Utilizar bloques de librerías (Página 2599)

Introducir el título del bloque (Página 2605)

Introducir el comentario de bloque (Página 2606)

Crear funciones y bloques de función

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para crear una función (FC) o un bloque de función (FB), proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo bloque".
   Se abre el diálogo "Agregar nuevo bloque".
2. Haga clic en el botón "Bloque de función (FB)" o "Función (FC)".
3. Introduzca un nombre para el nuevo bloque.
4. Introduzca las propiedades del nuevo bloque.
5. Haga clic en "Más información" para introducir más propiedades del nuevo bloque.
   Se abre un área con campos de entrada adicionales.
6. Introduzca todas las propiedades deseadas.
7. Active la casilla de verificación "Agregar y abrir" si el bloque debe abrirse inmediatamente después de crearse.
8. Confirme las entradas realizadas con "Aceptar".

Resultado

Se crea el nuevo bloque. Éste se encuentra entonces en la carpeta "Bloques de programa" del árbol del proyecto.

Consulte también

Bloques de función (FB) (Página 2253)

Funciones (FC) (Página 2252)

Fundamentos del acceso a bloques (Página 2258)

Carpeta de bloques (Página 2595)

Crear bloques de organización (Página 2596)

Crear bloques de datos (Página 2598)

Utilizar bloques de librerías (Página 2599)

Introducir el título del bloque (Página 2605)

Introducir el comentario de bloque (Página 2606)

Crear bloques de datos

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para crear un bloque de datos, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo bloque".
   Se abre el cuadro de diálogo "Agregar nuevo bloque".
2. Haga clic en el botón "Bloque de datos (DB)".
3. Seleccione el tipo de bloque de datos. Existen las posibilidades siguientes:
   - Para crear un bloque de datos global, seleccione la entrada de la lista "DB global".
   - Para crear un bloque de datos ARRAY, seleccione la entrada de la lista "DB ARRAY".
   - Para crear un bloque de datos instancia, seleccione de la lista el bloque de función al que desea asignar el bloque de datos instancia. En la lista solo aparecen bloques de función que se han creado con anterioridad para la CPU.
   - Para crear un bloque de datos basado en un tipo de datos PLC, seleccione en la lista el tipo de datos en cuestión. En la lista solo aparecen tipos de datos PLC que se han creado con anterioridad para la CPU.
   - Para crear un bloque de datos basado en un tipo de datos de sistema, seleccione en la lista el tipo de datos en cuestión. En la lista solo se muestran los tipos de datos del sistema que se han insertado con anterioridad en bloques de programa de la CPU.
4. Introduzca un nombre para el bloque de datos.
5. Introduzca las propiedades del nuevo bloque de datos.
6. Si se ha seleccionado como "tipo" un bloque de datos ARRAY, introduzca el tipo de datos ARRAY y el límite superior de ARRAY.
   El límite superior de ARRAY puede modificarse en todo momento en la ventana de propiedades del bloque generado. El tipo de datos ARRAY no puede modificarse a posteriori.
7. Haga clic en "Más información" para introducir más propiedades del nuevo bloque de datos.
   Se abre un área con campos de entrada adicionales.
8. Introduzca todas las propiedades deseadas.
9. Active la casilla de verificación "Agregar y abrir" si el bloque debe abrirse inmediatamente después de crearse.
10. Confirme la entrada realizada con "Aceptar".

Resultado

Se crea el nuevo bloque de datos. Este se encuentra entonces en la carpeta "Bloques de programa" del árbol del proyecto.

Consulte también

Bloques de datos globales (DB) (Página 2254)

Bloques de datos instancia (Página 2256)

Carpeta de bloques (Página 2595)

Crear bloques de organización (Página 2596)

Crear funciones y bloques de función (Página 2597)

Utilizar bloques de librerías (Página 2599)

Fundamentos del acceso a bloques (Página 2258)

Programar bloques de datos (Página 2937)

Programar tipos de datos PLC (Página 2967)

Tipos de datos de sistema (Página 2369)

Utilizar bloques de librerías

Los bloques se pueden depositar en la librería del proyecto o en una librería global para poder utilizarlos varias veces en el programa de usuario. Para ello, los bloques pueden insertarse como plantilla maestra o como tipo.

Consulte también: Principios básicos de las librerías (Página 427)

Requisitos

- Se visualiza la Task Card "Librerías".
- Las librerías globales no están protegidas contra escritura.

Agregar bloques como plantillas maestras a la librería del proyecto o a una librería global

Para agregar bloques nuevos como plantillas maestras a la librería del proyecto o a una librería global, proceda del siguiente modo:

1. Maximice la librería del proyecto o la librería global.
2. Arrastre el bloque que quiere agregar a la librería mediante Drag & Drop a la carpeta "Plantillas maestras" o a una subcarpeta cualquiera de las "Plantillas maestras" de la librería del proyecto o de una librería global. No suelte el botón izquierdo del ratón hasta que aparezca un signo "más" (+) pequeño debajo del puntero del ratón.

O bien:

1. Copie el elemento que desea agregar como plantilla maestra.
2. Maximice la librería del proyecto o la librería global.
3. Haga clic con el botón derecho del ratón en la carpeta "Plantillas maestras" o en cualquier subcarpeta de "Plantillas maestras".
4. Elija el comando "Pegar" del menú contextual.

Agregar bloques como tipos a la librería del proyecto o a una librería global

Para agregar bloques nuevos como tipos a la librería del proyecto o a una librería global, proceda del siguiente modo:

1. Maximice la librería del proyecto o la librería global.
2. Arrastre mediante Drag & Drop hasta la carpeta "Tipos" o a cualquier subcarpeta de "Tipos" de la librería del proyecto o de una librería global el elemento que desea agregar como tipo. No suelte el botón izquierdo del ratón hasta que aparezca un signo "más" (+) pequeño debajo del puntero del ratón.

O bien:

1. Copie el elemento que desea agregar como tipo.
2. Maximice la librería del proyecto o la librería global.
3. Haga clic con el botón derecho del ratón en la carpeta "Tipos" o en cualquier subcarpeta de "Tipos".
4. Elija el comando "Pegar" del menú contextual.

Utilizar bloques de la librería del proyecto o de una librería global

Para utilizar un bloque de la librería del proyecto o de una librería global en el proyecto, proceda del siguiente modo:

1. Maximice la librería del proyecto o la librería global, de manera que se visualice el bloque que desea utilizar.
2. Arrastre el bloque mediante Drag & Drop hasta la carpeta de bloques de la CPU. Los puntos de inserción no permitidos se identifican por el hecho de que el puntero del ratón se transforma en un círculo tachado (señal de prohibición).

Nota

Al derivar una instancia de un tipo de una librería global, el tipo también se agregará a la librería del proyecto. A partir de entonces la instancia solo estará enlazada al tipo de la librería del proyecto.

Consulte también

Utilizar librerías (Página 427)

Copiar e insertar bloques

Principios básicos de la copia e inserción de bloques

Función

Los bloques nuevos también se pueden crear copiando bloques existentes e insertando la copia. A este respecto, tenga en cuenta lo siguiente a la hora de copiar en las CPU de la misma familia de dispositivos:

- Los bloques de organización (OBs), funciones (FCs), bloques de función (FBs) y bloques de datos globales (DBs) pueden copiarse sin limitación alguna.
- Los bloques de datos instancia sólo se pueden copiar para el mismo bloque de función, puesto que la asignación al bloque de función no se puede modificar a posteriori. No obstante, la asignación se deshace si el bloque de datos instancia se copia a una CPU diferente. Sin embargo, si allí existe un bloque de función con el mismo nombre, el bloque de datos de instancia se asignará a este bloque de función. Si el bloque de datos instancia se copia junto con el bloque de función a la otra CPU, se asignará a la copia del bloque de función.

Las diferentes familias de dispositivos soportan en parte diferentes bloques, especialmente en el caso de los bloques de organización. No obstante, también es posible programar bloques de función y funciones en los diferentes dispositivos con distintos tipos de acceso. Por consiguiente, no todos los dispositivos soportan todos los bloques. Al copiar en otra familia de dispositivos, tenga en cuenta lo siguiente

- Copiar en una CPU S7-1200:
  - En una S7-1200 pueden copiarse bloques de organización con tipo de acceso "optimizado". Si la CPU S7-1200 soporta el tipo de OB copiado, el OB copiado conserva las propiedades de su evento. No obstante, es necesario compilarlo de nuevo.
  - En un S7-1200 pueden copiarse bloques de organización con tipo de acceso "estándar", pero la CPU no los soportará.
  - En un S7-1200 pueden copiarse bloques de función (FB), funciones (FC) y bloques de datos globales (DB) con tipo de acceso "optimizado". A continuación es necesario compilarlos de nuevo.
  - En un S7-1200 pueden copiarse bloques de función (FB), funciones (FC) y bloques de datos globales (DB) con tipo de acceso "estándar", pero la CPU no los soportará.
  - Bloques de datos de instancia: si en la CPU de destino ya existe un bloque de función con el nombre asignado al bloque de datos de instancia en la CPU de origen, el bloque de datos de instancia se asignará al bloque de función en la CPU de destino. Si el bloque de datos de instancia se copia a la CPU de destino junto con el bloque de función al que fue asignado en la CPU de origen, el bloque de datos de instancia se asignará a la copia del bloque de función.
- Copiar en una CPU S7-1500:
  - En un S7-1500 pueden copiarse bloques de organización con tipo de acceso "optimizado". Si la CPU S7-1500 soporta el tipo de OB copiado, el OB copiado conserva las propiedades de su evento. No obstante, es necesario compilarlo de nuevo. Los tipos de OB no soportados se marcan con el símbolo de "prohibido aparcar".
  - En un S7-1500 pueden copiarse bloques de organización con el tipo de acceso "estándar". Si el OB procede de una CPU S7-300/400, obtiene el evento estándar del tipo de OB correspondiente. Si el OB procede de una CPU S7-1200/1500, conserva las propiedades de su evento pero es necesario compilarlo de nuevo.
  - En un S7-1500 pueden copiarse bloques de función (FB), funciones (FC) y bloques de datos globales (DB) con tipo de acceso "optimizado". A continuación es necesario compilarlos de nuevo.
  - En un S7-1500 pueden copiarse bloques de función (FB), funciones (FC) y bloques de datos globales (DB) con tipo de acceso "estándar", pero la CPU no los soportará.
  - Bloques de datos de instancia: si en la CPU de destino ya existe un bloque de función con el nombre asignado al bloque de datos de instancia en la CPU de origen, el bloque de datos de instancia se asignará al bloque de función en la CPU de destino. Si el bloque de datos de instancia se copia a la CPU de destino junto con el bloque de función

al que fue asignado en la CPU de origen, el bloque de datos de instancia se asignará a la copia del bloque de función.

- Copiar en CPU S7-300/400:
  - Los bloques de organización pueden copiarse a voluntad entre S7-300 y S7-400.
  - Los bloques de organización de las CPU S7-1200/1500 se pueden copiar en las CPU S7-300/400, pero la CPU de destino no los soportará.
  - Los bloques de función (FB), las funciones (FC) y los bloques de datos globales (DB) se pueden copiar a voluntad entre S7-300 y S7-400.
  - Los bloques de función (FB), las funciones (FC) y los bloques de datos globales (DB) de las CPU S7-1200/1500 se pueden copiar en las CPU S7-300/400, pero la CPU de destino no los soportará.
  - Bloques de datos de instancia: si en la CPU de destino ya existe un bloque de función con el nombre asignado al bloque de datos de instancia en la CPU de origen, el bloque de datos de instancia se asignará al bloque de función en la CPU de destino. Si el bloque de datos de instancia se copia a la CPU de destino junto con el bloque de función al que fue asignado en la CPU de origen, el bloque de datos de instancia se asignará a la copia del bloque de función.

Los bloques no soportados se identifican en el árbol del proyecto con el símbolo de "prohibido aparcar". Los bloques con un símbolo de "prohibido aparcar" no pueden editarse, solo reutilizarse como fuentes para copiar.

Datos copiados

Al insertar se copian todos los datos del bloque y se transmiten a la copia. Algunos de estos datos son:

- Variables de la interfaz del bloque
- Todos los segmentos
- Comentarios en todas las traducciones existentes
- Avisos definidos en el bloque
- El código del programa completo del bloque copiado, incluyendo las instrucciones de llamada que contiene.
  Sin embargo, no se copian los bloques llamados ni los bloques de datos instancia asociados.

Evitar conflictos de nombre al insertar

Al insertar un bloque copiado cuyo nombre sea idéntico al de un bloque existente, se aplican los siguientes mecanismos para evitar conflictos de nombre:

- Inserción del bloque copiado en la misma CPU:
  La copia del bloque obtiene un nombre que se complementa con un número. Por ejemplo, si se copia el bloque "A", un nombre posible de la copia sería "A_1". La numeración no se realiza de forma correlativa, sino que se utiliza siempre el número libre más bajo. Por tanto, el nombre de la copia del bloque "A" también podría ser "A_25" si no está disponible ningún otro número más bajo.
- Inserción del bloque copiado en una CPU diferente:
  Se abre un diálogo en el que es posible seleccionar si se desea sustituir el bloque homónimo o insertar el bloque copiado con una identificación de duplicado (nombre_número).

Nota

Conflictos de números

No obstante, pueden ocurrir conflictos de nombre si el bloque insertado tiene el mismo número que un bloque existente. El número de bloque no se modifica automáticamente al realizar la inserción. Esto puede afectar p. ej. a las llamadas de bloques. Por tanto, al copiar bloques, compruebe con cuidado los números de bloque y, en caso necesario, corrija manualmente los números repetidos a través de las propiedades del bloque. Los números de bloque repetidos provocan un error de compilación.

Consulte también

Copiar bloques (Página 2604)

Insertar bloques (Página 2605)

Copiar bloques

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para copiar un bloque, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el bloque que desea copiar.
2. Elija el comando "Copiar" del menú contextual.

Resultado

El portapapeles contiene una copia del bloque que puede insertarse en la misma CPU o en otra diferente.

Consulte también

Principios básicos de la copia e inserción de bloques (Página 2601)

Insertar bloques (Página 2605)

Insertar bloques

Requisitos

Se ha copiado un bloque.

Procedimiento

Para insertar un bloque copiado con sus datos en una CPU, proceda del siguiente modo:

1. En el árbol del proyecto, abra la estructura de carpetas de la CPU en la que desea insertar el bloque copiado.

   Nota

   Tenga en cuenta que sólo se puede insertar el bloque copiado en una CPU que soporte el lenguaje y el tipo de bloque.

2. Haga clic con el botón derecho del ratón en la carpeta "Bloques de programa".
3. Elija el comando "Pegar" del menú contextual.
   - Si inserta el bloque en la CPU del bloque original, la copia se insertará con la extensión de nombre "_<número correlativo>".
   - Si inserta el bloque en una CPU diferente y ya existe un bloque homónimo, aparecerá el diálogo "Pegar". Seleccione la opción deseada y confírmela con "Aceptar".

Consulte también

Principios básicos de la copia e inserción de bloques (Página 2601)

Copiar bloques (Página 2604)

Introducir el título del bloque

El título del bloque es la cabecera del bloque. No es idéntico al nombre de bloque que se asignó al crear el bloque. La longitud del título del bloque está limitada a una línea. Es posible introducir títulos de bloque para los bloques tanto abiertos como cerrados.

Requisitos

Existe un bloque lógico.

Introducir el título de un bloque abierto

Para introducir el título de un bloque abierto, proceda del siguiente modo:

1. En el editor de programación, haga clic en en la barra de título del bloque.
2. Introduzca el título del bloque.

Introducir el título de un bloque cerrado

Para introducir el título de un bloque cerrado, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic con el botón derecho del ratón en el bloque.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo con las propiedades del bloque.
3. Seleccione la entrada "Información" en la navegación local.
4. Introduzca el título del bloque en el campo de entrada "Título".
5. Confirme las entradas realizadas con "Aceptar".

Consulte también

Crear bloques de organización (Página 2596)

Crear funciones y bloques de función (Página 2597)

Introducir el comentario de bloque (Página 2606)

Introducir el comentario de bloque

El comentario de bloque permite documentar todo el bloque lógico. Por ejemplo, se puede describir la finalidad del bloque, o bien llamar la atención sobre peculiaridades. Es posible introducir comentarios de bloque para los bloques tanto abiertos como cerrados.

Requisitos

Existe un bloque lógico.

Introducir el comentario de un bloque abierto

Para introducir el comentario de un bloque abierto, proceda del siguiente modo:

1. Haga clic en la flecha pequeña antepuesta al título del bloque.
   La flecha hacia la derecha se convierte en una flecha hacia abajo y aparece el área de comentario.
2. En el área de comentario, haga clic en "Comentario".
   El texto "Comentario" aparece seleccionado.
3. Introduzca el comentario de bloque.

Introducir el comentario de un bloque cerrado

Para introducir el comentario de un bloque cerrado, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic con el botón derecho del ratón en el bloque.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo con las propiedades del bloque.
3. Seleccione la entrada "Información" en la navegación local.
4. Introduzca el comentario de bloque en el campo de entrada "Comentario".
5. Confirme las entradas realizadas con "Aceptar".

Consulte también

Crear bloques de organización (Página 2596)

Crear funciones y bloques de función (Página 2597)

Introducir el título del bloque (Página 2605)

9.1.3.2 Definir propiedades de bloques

Principios básicos de las propiedades de bloques

Propiedades de bloques

Todo bloque tiene determinadas propiedades que se pueden visualizar y editar. Estas propiedades tienen, por ejemplo, las siguientes tareas:

- Identificar el bloque
- Visualizar la memoria necesaria y el estado de compilación del bloque
- Visualizar el sello de tiempo
- Visualizar la información de referencia
- Ajustar la protección de acceso

Consulte también

Sinopsis de las propiedades de bloques (Página 2608)
Sellos de tiempo de bloques (Página 2611)
Mostrar y editar propiedades de bloques (Página 2613)
Ajustar los nemónicos (Página 2630)

Sinopsis de las propiedades de bloques

Resumen

Las propiedades de los bloques son específicos del bloque y la CPU. Por eso, no todas las propiedades están disponibles para todos los bloques y en todas las familias de CPU. La siguiente tabla muestra una vista general de las propiedades del bloque:

Grupo	Propiedad	Descripción
General	Nombre	Nombre unívoco del bloque en la estación.
	Nombre de la constante	Nombre de la constante que se inserta para el OB en la tabla de variables PLC
	Tipo	Tipo de bloque (no se puede modificar)
	Número	Número de bloque
	Clase de evento	Clase de evento de un OB (no modificable)
	Lenguaje	Lenguaje de programación del bloque
	Lenguaje en segmentos	Es el lenguaje que se utiliza para programar las condiciones en los bloques GRAPH.
	Número de memoria imagen parcial del proceso	Indicación de las memorias imagen parciales de proceso asignadas al bloque de organización (no modificable)
	Tipo de datos ARRAY	Tipo de datos de un bloque de datos ARRAY (no modificable)
	Límite de ARRAY	Límite superior de un bloque de datos ARRAY La sección "Transferencia" de la Task Card "Instrucciones" permite direccionar bloques de datos ARRAY.
Información	Título	Título del bloque
	Comentario	Comentario del bloque
	Versión	Número de versión del bloque
	Familia	Nombre de la familia de bloques
	Autor	Nombre del autor, empresa, departamento u otro nombre
	ID personalizada	ID creada por el usuario
Sello de tiempo	Bloque	Fecha y hora de creación y modificación del bloque (no modificable)
	Interfaz	Fecha de modificación de la interfaz del bloque (no modificable)
	Código	Fecha de modificación del código (no modificable)
	Datos	Fecha de modificación de los datos (no modificable)
Compilación	Estado	Detalles de la última compilación (no modificable)
	Longitudes	Detalles de las longitudes de bloque (no modificable)

Grupo	Propiedad	Descripción
Protección	Protección	Ajustar protección de know-how y contra copia del bloque Consulte también: Proteger bloques (Página 3045)
Atributos	Acceso optimizado al bloque	En bloques con acceso optimizado, la declaración de variables solo contiene los nombres simbólicos de los elementos de datos. Las direcciones son optimizadas y gestionadas automáticamente por el sistema. El rendimiento de la CPU aumenta y no se producen errores de acceso, p. ej. desde SIMATIC HMI. Consulte también: Bloques con acceso optimizado (Página 2258)
	Verificación CEI	La compatibilidad de operandos en operaciones de comparación y aritméticas se verifica según CEI 61131. Los operandos no compatibles deben convertirse explícitamente. Consulte también: Sinopsis de la conversión de tipos de datos (Página 2461)
	Tratamiento de errores locales en el bloque	Tratamiento de errores en el bloque con las instrucciones GET_ERROR y/o GET_ERR_ID (no modificable). Consulte también: Tratar errores de ejecución del programa (Página 2932)
	Crear información avanzada de estado	Permite observar todas las variables de un bloque SCL. Sin embargo, la cantidad de memoria que requiere el programa y los tiempos de ejecución aumentan mediante esta opción.
	Comprobar límites de ARRAY	Comprueba durante el tiempo de ejecución de un bloque SCL si hay índices de campo en el área declarada para un ARRAY. Si un índice de campo está fuera del rango permitido, la salida de habilitación ENO del bloque se pone a "0".
	Asignar ENO automáticamente	Comprueba durante el tiempo de ejecución de un bloque SCL si se producen errores en el procesamiento de determinadas instrucciones. Si se produce un error en el tiempo de ejecución, la salida de habilitación ENO del bloque se pone a "0".
	Generar DB minimizado	Genera bloques de datos de instancia en formato minimizado para bloques GRAPH de las S7-300 y S7-400. Esta opción reduce considerablemente la cantidad de memoria que requiere el FB GRAPH. Por otro lado, solo se obtiene información limitada sobre el estado del programa.
	Saltar etapas	Si las transiciones en un bloque GRAPH se vuelven válidas al mismo tiempo antes y después de una etapa, esta no se activa y por consiguiente se salta.
	Acuse obligatorio de errores de supervisión	Si se produce un error de supervisión durante el funcionamiento de un bloque GRAPH, dicho error se debe acusar antes de que se siga ejecutando el programa.
	Procesamiento permanente de todos los interlocks en modo manual	Hace que las condiciones de interlock de un bloque GRAPH se comprueben permanentemente en modo manual.
	Bloqueo de selección de modo	Impide la selección de un modo de operación de un bloque GRAPH.
	Bloque de datos protegido contra escritura en el dispositivo	Indica si el bloque de datos está protegido contra escritura en el sistema de destino, por lo que no se puede sobrescribir durante la ejecución del programa (solo para bloques de datos)

Grupo	Propiedad	Descripción
	Depositar solo en la memoria de carga	Si se activa, el bloque de datos se depositará solo en la memoria de carga, no ocupará espacio en la memoria de trabajo y no se incorporará en el programa. La sección "Desplazamiento" de la Task Card "Instrucciones" permite transferir los bloques de datos a la memoria de trabajo (solo para bloques de datos).
	Información de arranque	En las CPU S7-1500, aquí se define la estructura de la información de arranque del bloque de organización: como en las CPUs S7-300 y S7-400 o bien información de arranque optimizada.
	Prioridad	En los bloques de organización muestra la prioridad ajustada. El que pueda cambiarse la prioridad dependerá de la familia de CPU utilizada y del tipo de bloque de organización.
	Asignación de parámetros a través de fichas	En un bloque AWL de una CPU S7-1500, permite la asignación de parámetros en fichas. Ello permite utilizar las instrucciones "Llamada condicional" (CC) y "Llamada incondicional" (UC) en el bloque.
	Bloque utilizable como elemento de librería con protección de know-how	Indica si el bloque de la librería puede utilizarse con protección de know-how.
	Activar relectura	Permite marcar diferentes parámetros del bloque como "relegibles". La función "Releer variables" es relevante cuando el bloque se utiliza en un esquema CFC.
	Representación del bloque	Determina cómo se representará el bloque en un esquema CFC.
Alarma horaria	Alarma horaria	Parámetros del OB de alarma horaria: Activo (sí o no), ejecución, fecha y hora de inicio, hora local o de sistema
Alarma cíclica	Alarma cíclica	Tiempo de ciclo y desfase del OB de alarma cíclica
Eventos de inicio	Eventos de inicio	Indicación de los eventos de inicio del OB de alarma de proceso
Modo isócrono	Modo isócrono	Parámetros del OB de alarma de sincronismo: Ciclo de aplicación, ajuste automático (sí o no), tiempo de retardo. Además se indica el sistema PROFINET IO o el sistema maestro DP cuyos dispositivos IO o esclavos DP, respectivamente, están asignados al OB de alarma de sincronismo.
Cargar sin reinicializar	Reserva en la memoria estándar	Tamaño de la reserva en la memoria estándar disponible para ampliaciones de interfaz. El número de bytes disponibles actualmente se indica entre paréntesis. Esta indicación se actualiza en cada proceso de compilación.
	Activar la carga sin reinicializar para variables remanentes	Permite definir una reserva en la memoria remanente.
	Reserva en la memoria remanente	Tamaño de la reserva en la memoria remanente disponible para ampliaciones de interfaz. El número de bytes disponibles actualmente se indica entre paréntesis. Esta indicación se actualiza en cada proceso de compilación.

Consulte también

Principios básicos de las propiedades de bloques (Página 2607)

Sellos de tiempo de bloques (Página 2611)

Mostrar y editar propiedades de bloques (Página 2613)

Fundamentos del acceso a bloques (Página 2258)

Configuración general de la programación PLC (Página 2640)

Sellos de tiempo de bloques

Introducción

A los bloques se les asignan diferentes sellos de tiempo, que permiten saber cuándo se ha creado el bloque y cuándo se ha modificado por última vez. Estos sellos de tiempo también se utilizan para comprobar la coherencia de forma automática antes de un proceso de compilación.

Sellos de tiempo para bloques lógicos

Para los bloques lógicos (OBs, FBs, FCs) se generan los siguientes sellos de tiempo:

- Bloque: Fecha de creación, fecha de la última modificación
- Interfaz: Fecha de la última modificación
- Código/datos: Fecha de la última modificación

Al compilar se indica un conflicto de sello de tiempo cuando el sello de tiempo del bloque que llama es más antiguo que el de la interfaz del bloque llamado.

Los sellos de tiempo de los bloques lógicos se actualizan del siguiente modo:

- Bloque: El sello de tiempo de la última modificación del bloque corresponde siempre al sello de tiempo de la interfaz o del código, dependiendo de qué área fue modificada por última vez.
- Interfaz: El sello de tiempo de la interfaz se actualiza cada vez que ésta se modifica. Aunque se deshaga manualmente la modificación de la interfaz, p. ej. se vuelva a cambiar el nombre, la modificación actualiza el sello de tiempo. Pero si se deshace la modificación mediante la función "Deshacer", el sello de tiempo se restablece al valor anterior a la modificación deshecha.
- Código/datos: El sello de tiempo del código se actualiza cada vez que se modifica el código del bloque. Aunque se deshaga manualmente la modificación del código, p. ej. se vuelva a quitar una instrucción, la modificación actualiza el sello de tiempo. Pero si se deshace la modificación mediante la función "Deshacer", el sello de tiempo se restablece al valor anterior a la modificación deshecha.

Sellos de tiempo para bloques de datos globales

Para los bloques de datos globales se generan los siguientes sellos de tiempo:

- Bloque: Fecha de creación, fecha de la última modificación
- Interfaz: Fecha de la última modificación
- Datos: Fecha de la última modificación

Al compilar un bloque de datos global basado en un tipo de datos PLC se indica un conflicto de sello de tiempo cuando el sello de tiempo del bloque de datos global es más antiguo que el del tipo de datos PLC utilizado.

Los sellos de tiempo de los bloques de datos globales se actualizan del siguiente modo:

- Bloque: El sello de tiempo de la última modificación de un bloque de datos global corresponde siempre al sello de tiempo de la interfaz y los datos.
- Interfaz y datos: Los sellos de tiempo de la interfaz y los datos se actualizan cada vez que se modifica el bloque de datos global. Aunque se deshaga manualmente la modificación, p. ej. se vuelva a quitar una variable, la modificación actualiza los sellos de tiempo. Pero si se deshace la modificación mediante la función "Deshacer", los sellos de tiempo se restablecen al valor anterior a la modificación deshecha.

Sellos de tiempo para bloques de datos de instancia

Para los bloques de datos de instancia se generan los siguientes sellos de tiempo:

- Bloque: Fecha de creación, fecha de la última modificación
- Interfaz: Fecha de la última modificación
- Datos: Fecha de la última modificación

Al compilar un bloque de datos de instancia se indica un conflicto del sello de tiempo cuando el sello de tiempo de las interfaces del bloque de datos de instancia y del bloque de función no son idénticos.

Los sellos de tiempo de los bloques de datos de instancia se actualizan del siguiente modo:

- Bloque: El sello de tiempo de la última modificación de un bloque de datos de instancia corresponde siempre al sello de tiempo de la interfaz y los datos.
- Interfaz y datos: Los sellos de tiempo de la interfaz y los datos se actualizan cada vez que se modifica el bloque de datos de instancia. Aunque se deshaga manualmente la modificación, p. ej. se vuelva a anular la remanencia de una variable, la modificación actualiza los sellos de tiempo. Pero si se deshace la modificación mediante la función "Deshacer", los sellos de tiempo se restablecen al valor anterior a la modificación deshecha.

Sellos de tiempo para tipos de datos PLC

Para los tipos de datos PLC se generan los siguientes sellos de tiempo:

- Bloque: Fecha de creación, fecha de la última modificación
- Interfaz: Fecha de la última modificación

Los sellos de tiempo de los tipos de datos PLC se actualizan del siguiente modo:

- Bloque: El sello de tiempo de la última modificación de un tipo de datos PLC corresponde siempre al sello de tiempo de la interfaz.
- Interfaz: El sello de tiempo de la interfaz se actualiza cada vez que se modifica el tipo de datos PLC. Aunque se deshaga manualmente la modificación, p. ej. se vuelva a borrar el contenido del tipo de datos PLC, la modificación actualiza el sello de tiempo. Pero si se deshace la modificación mediante la función "Deshacer", el sello de tiempo se restablece al valor anterior a la modificación deshecha.

Consulte también

Principios básicos de las propiedades de bloques (Página 2607)

Sinopsis de las propiedades de bloques (Página 2608)

Mostrar y editar propiedades de bloques (Página 2613)

Principios básicos de la compilación de bloques (Página 3012)

Mostrar y editar propiedades de bloques

Las propiedades de los bloques son específicos del bloque y la CPU. Por eso, no todas las propiedades están disponibles para todos los bloques y en todas las familias de CPU. Las propiedades que sólo se pueden mostrar están protegidas contra escritura.

Mostrar y editar las propiedades de un bloque cerrado

Para mostrar o editar las propiedades de un bloque cerrado, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del árbol del proyecto.
2. Haga clic con el botón derecho del ratón en el bloque cuyas propiedades desea mostrar o editar.
3. Elija el comando "Propiedades" del menú contextual.
   Se abre el diálogo de propiedades del bloque.
4. En la navegación local, haga clic en el grupo cuyas propiedades desea mostrar o editar.
5. Modifique la propiedad deseada.
6. Confirme las entradas realizadas con "Aceptar".

Mostrar y editar las propiedades de un bloque abierto

Para mostrar o editar las propiedades de un bloque abierto, proceda del siguiente modo:

1. Active la opción "Ventana de inspección" del menú "Ver".
   Se abre la ventana de inspección.
2. Haga clic en la ficha "Propiedades".
   Las propiedades del bloque se visualizan en la ficha "Propiedades" de la ventana de inspección.
3. En la navegación local, haga clic en el grupo cuyas propiedades desea mostrar o editar.
4. Modifique la propiedad deseada.

Resultado

Se modifican las propiedades del bloque. No obstante, los cambios no se guardarán hasta que se guarde el proyecto.

Consulte también

Principios básicos de las propiedades de bloques (Página 2607)

Sinopsis de las propiedades de bloques (Página 2608)

Sellos de tiempo de bloques (Página 2611)

9.1.3.3 Gestionar bloques

Abrir bloques

Los bloques se pueden abrir de las maneras siguientes:

- Abrir bloque directamente
  Puede abrir directamente un bloque si ha abierto la correspondiente carpeta de bloques en el árbol del proyecto o en la ventana general.
- Buscar y abrir bloque
  Puede buscar bloques dentro de un proyecto, un dispositivo y la carpeta "Bloques de programa" y después abrirlos.

Abrir bloque directamente

Para abrir un bloque directamente, proceda del siguiente modo:

1. Abra la carpeta con el bloque que desee abrir, en el árbol del proyecto o en la ventana general.
2. Haga doble clic en el bloque que desea abrir.

Buscar y abrir bloque

Para buscar un bloque y abrirlo, proceda de la siguiente manera:

1. Haga clic en el árbol del proyecto con el botón derecho del ratón en el proyecto, un dispositivo, la carpeta "Bloques de programa" o en cualquier subcarpeta de "Bloques de programa".
2. Elija el comando "Buscar y abrir en PLC" del menú contextual.
   Se abre el cuadro de diálogo "Buscar y abrir en PLC".
3. Introduzca el nombre, la dirección o el tipo de bloque que está buscando.
   Con cada letra introducida se sigue filtrando la lista de bloques. Si no hay ningún bloque que coincida con su entrada, se cierra la lista de bloques. Puede mostrar la lista completa de bloques en todo momento haciendo clic en el botón situado a la derecha del campo de texto. Sin embargo, tenga en cuenta que después no tiene lugar ningún filtrado. Si desea filtrar de nuevo por sus entradas, vuelva a hacer clic en el botón.
4. En la lista de bloques, haga clic en el bloque que desee abrir.

Resultado

El bloque se abre en el editor de programación.

Consulte también

Guardar bloques (Página 2615)

Cerrar bloques (Página 2615)

Cambiar el nombre de bloques (Página 2616)

Borrar bloques offline (Página 2616)

Borrar bloques online (Página 2617)

Abrir bloques con protección de know-how (Página 3050)

Guardar bloques

Los bloques se guardan siempre junto con el proyecto. También es posible guardar bloques erróneos. Los errores pueden solucionarse en cualquier momento.

Procedimiento

Para guardar un bloque, proceda del siguiente modo:

1. Elija el comando "Guardar" o "Guardar como" del menú "Proyecto".
   Consulte también: Guardar proyectos (Página 346)

Consulte también

Abrir bloques (Página 2614)

Cerrar bloques (Página 2615)

Cambiar el nombre de bloques (Página 2616)

Borrar bloques offline (Página 2616)

Borrar bloques online (Página 2617)

Cerrar bloques

Procedimiento

Para cerrar un bloque, proceda del siguiente modo:

1. Haga clic en el botón "Cerrar" de la barra de título del editor de programación.

Nota

Tenga en cuenta que el bloque no se guarda al cerrarlo.

Consulte también

Abrir bloques (Página 2614)
Guardar bloques (Página 2615)
Cambiar el nombre de bloques (Página 2616)
Borrar bloques offline (Página 2616)
Borrar bloques online (Página 2617)

Cambiar el nombre de bloques

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para cambiar el nombre de un bloque, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el bloque cuyo nombre desea cambiar.
2. Elija el comando "Cambiar nombre" del menú contextual.
   El nombre del bloque se transforma en un campo de entrada en el árbol del proyecto.
3. Introduzca el nuevo nombre para el bloque.
4. Confirme las entradas realizadas pulsando la tecla Intro.

Resultado

El nombre del bloque se cambia en todas las ubicaciones del programa.

Consulte también

Abrir bloques (Página 2614)
Guardar bloques (Página 2615)
Cerrar bloques (Página 2615)
Borrar bloques offline (Página 2616)
Borrar bloques online (Página 2617)

Borrar bloques offline

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para borrar un bloque que exista offline, proceda del siguiente modo:

1. En la carpeta "Bloques de programa" del árbol del proyecto, haga clic con el botón derecho del ratón en el bloque que desea borrar.
2. Elija el comando "Borrar" del menú contextual.
3. Confirme la consulta de seguridad con "Sí".
   El bloque se borra del proyecto offline.

Nota

Al borrar bloques de organización, recuerde que pueden tener eventos asignados. Si borra estos bloques de organización, el programa no podrá reaccionar a los eventos parametrizados.

Consulte también

Abrir bloques (Página 2614)

Guardar bloques (Página 2615)

Cerrar bloques (Página 2615)

Cambiar el nombre de bloques (Página 2616)

Borrar bloques online (Página 2617)

Cargar bloques (Página 3018)

Borrar bloques online

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para borrar un bloque que exista online, proceda del siguiente modo:

1. En la carpeta "Bloques de programa" del árbol del proyecto, haga clic con el botón derecho del ratón en el bloque que desea borrar del dispositivo.
2. Elija el comando "Borrar" del menú contextual.
   Se abre el cuadro de diálogo "Borrar".
3. Active el botón de opción "Borrar del dispositivo".
4. Haga clic en "Sí".
   El bloque se borra del dispositivo online.

Consulte también

Abrir bloques (Página 2614)

Guardar bloques (Página 2615)

Cerrar bloques (Página 2615)

Cambiar el nombre de bloques (Página 2616)

Borrar bloques offline (Página 2616)

Borrar bloques de datos de CPU

Los bloques de datos de CPU se pueden borrar tanto offline como online.

Borrar bloques de datos de CPU offline

Para borrar un bloque de datos de CPU disponible offline en el proyecto, proceda del siguiente modo:

1. En la carpeta "Bloques de programa" del árbol del proyecto, haga clic con el botón derecho del ratón en el bloque de datos de CPU que desea borrar.
2. Elija el comando "Borrar" del menú contextual.
3. Confirme la consulta de seguridad con "Sí".
   El bloque de datos de CPU se borra del proyecto offline.

Borrar un bloque de datos de CPU online

Para borrar un bloque de datos de CPU disponible online, proceda del siguiente modo:

1. Establezca una conexión online con el dispositivo que contiene el bloque de datos de CPU que desea borrar.
2. En la carpeta "Bloques de programa" del árbol del proyecto, haga clic con el botón derecho del ratón en el bloque de datos de CPU que desea borrar en el dispositivo.
3. Elija el comando "Borrar" del menú contextual.
   Se abre el cuadro de diálogo "Borrar".
4. Active el botón de opción "Borrar del dispositivo".
5. Haga clic en "Sí".
   El bloque de datos de CPU se borra online del dispositivo.

Consulte también

Bloques de datos de CPU (Página 2256)

9.1.4 Programar bloques

9.1.4.1 Editor de programación

Sinopsis del editor de programación

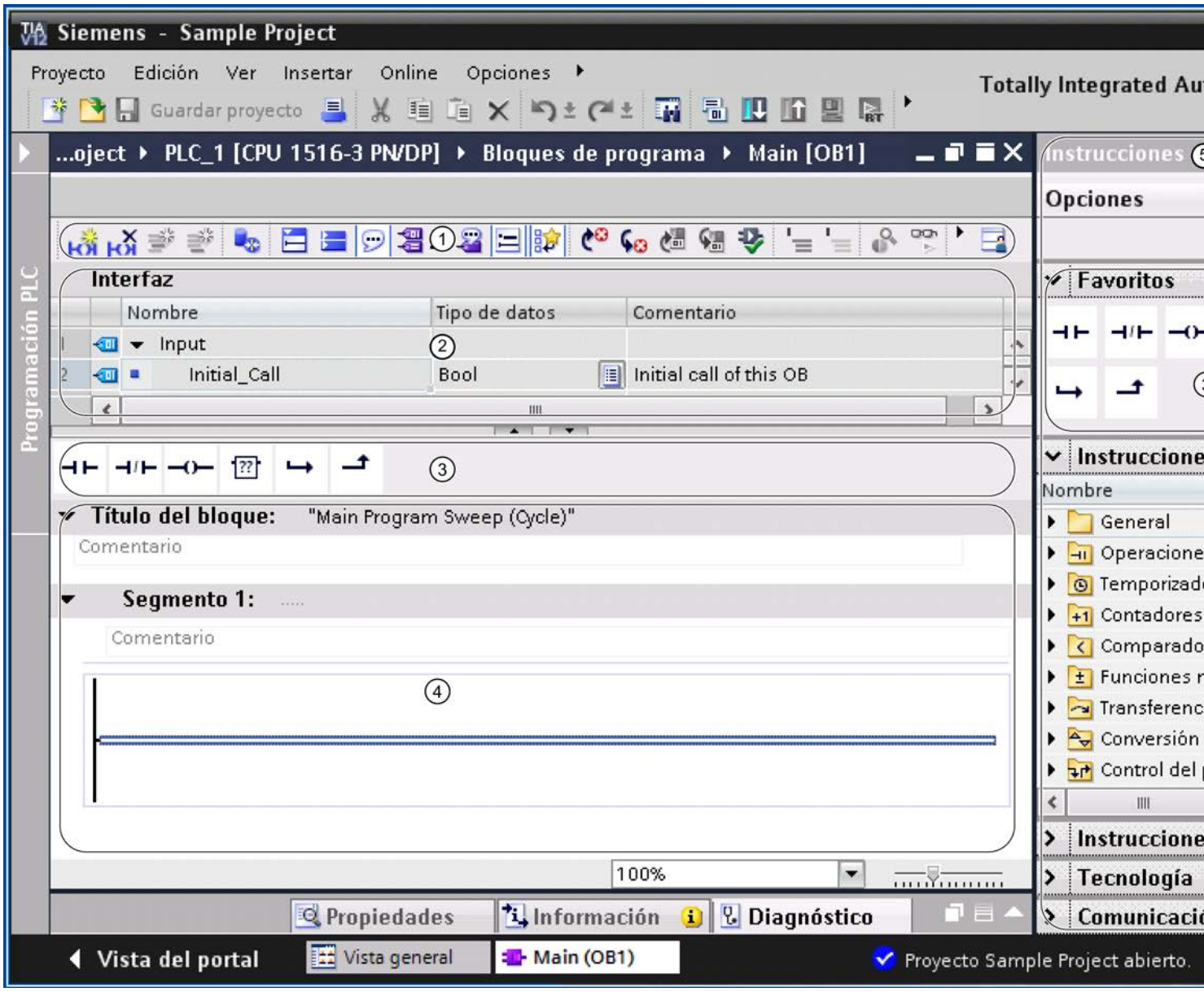
Función del editor de programación

El editor de programación es el entorno de desarrollo integrado para la programación de funciones, bloques de función y bloques de organización. Proporciona toda la ayuda posible para la creación de programas y la búsqueda de errores.

En función de la CPU utilizada, del lenguaje de programación y del tipo de bloque, el aspecto y las funciones del editor de programación pueden variar.

Estructura del editor de programación

La figura siguiente muestra los componentes del editor de programación, tomando como ejemplo KOP:



① Barra de herramientas
② Interfaz del bloque
③ Paleta "Favoritos" de la Task Card "Instrucciones" y Favoritos en el editor de programación
④ Ventana de programación
⑤ Task Card "Instrucciones"
⑥ Task Card "Test"

Barra de herramientas

La barra de herramientas permite acceder a las principales funciones del editor de programación, tales como:

- Mostrar y ocultar operandos absolutos
- Mostrar y ocultar Favoritos
- Saltar a errores de sintaxis
- Actualizar llamadas de bloque
- Mostrar y ocultar el estado del programa

Las funciones disponibles en la barra de herramientas pueden variar según el lenguaje de programación utilizado.

Interfaz del bloque

La interfaz del bloque contiene las declaraciones de las variables locales que se utilizan sólo en el bloque. Las secciones disponibles varían según el tipo de bloque.

Favoritos

Existe la posibilidad de crear favoritos para las instrucciones utilizadas con frecuencia. Estos favoritos se muestran en la paleta "Favoritos" de la Task Card "Instrucciones". Además, los favoritos también se pueden visualizar en el editor de programación a través de su barra de herramientas. De esta manera, se puede acceder a los favoritos sin que la Task Card "Instrucciones" esté visible.

Ventana de programación

La ventana de programación es el área de trabajo del editor de programación. Aquí se puede introducir el código. En función del lenguaje de programación utilizado, la ventana de programación puede variar en su aspecto y funcionalidad.

Task Card "Instrucciones"

La Task Card "Instrucciones" permite acceder fácilmente a todas las instrucciones que pueden utilizarse para crear el programa. Las instrucciones están divididas temáticamente en varias paletas. Mediante el botón "Mostrar títulos de columna y columnas adicionales" de la barra de herramientas de la Task Card es posible visualizar más información relacionada con las instrucciones. La disposición de las columnas se puede modificar haciendo clic en el título de la columna y moviendo la columna mediante Drag & Drop.

Si un perfil de instrucciones está activo, las instrucciones disponibles varían.
Consulte también: Utilizar perfiles de instrucciones (Página 2632)

Task Card "Test"

En la Task Card "Test" pueden realizarse ajustes que influyen en la búsqueda de errores con ayuda del estado del programa. Las funciones de la Task Card "Test" están disponibles

únicamente en modo online. Incluye las paletas siguientes, que se muestran en función de la CPU seleccionada y del lenguaje de programación configurado del bloque:

- Panel de control de la CPU
  El estado operativo de la CPU se puede conmutar en el Panel de control de la CPU.
- Puntos de parada
  Los bloques creados en uno de los lenguajes de programación textuales AWL o SCL pueden probarse en el modo de etapa individual. Para ello, inserte puntos de parada en el código del programa.
  En la paleta "Puntos de parada" puede encontrar todos los puntos de parada que ha introducido; se pueden activar, borrar, navegar hasta puntos de parada concretos o bien establecer el entorno de llamada del punto de parada.
- Registro AS
  En esta paleta se pueden leer los valores de los registros AS y los acumuladores.
- Control secuencial
  En esta paleta se establece el modo de operación para el test de las cadenas secuenciales de los bloques GRAPH.
- Ajustes de test
  En esta paleta se establece la configuración de test de los bloques GRAPH.
- Entorno de llamada
  En esta paleta se define el entorno de llamada del bloque.
- Jerarquía de llamada
  Esta paleta permite seguir la jerarquía de llamada de los bloques. La jerarquía de llamada se visualiza únicamente durante la observación de bloques.

Consulte también

Declarar la interfaz del bloque (Página 2642)

Estructura de la interfaz del bloque (Página 2642)

Crear el código del programa (Página 2667)

Ampliar la superficie de la ventana de programación (Página 2627)

Comandos de teclado del editor de programación

Navegar en el editor

Función	Combinación de teclas
Abrir la Task Card "Instrucciones"	<Ctrl+Mayús+C>
Abrir la Task Card "Test"	<Ctrl+Mayús+O>
Agregar nuevo bloque	<Ctrl+N>
Ampliar todos los segmentos	<Alt+F11>
Reducir todos los segmentos	<Alt+F12>

Función	Combinación de teclas
Navegar a la siguiente ubicación del bloque u operando seleccionado	<Ctrl+Mayús+F>
Navegar a la ubicación anterior del bloque u operando seleccionado	<Ctrl+Mayús+G>
Navegar al siguiente acceso de lectura o escritura	<Alt+F8>
Navegar al acceso de lectura o escritura anterior	<Alt+F9>

Navegar en el código del programa (KOP/FUP)

Función	Objeto seleccionado	Combinación de teclas
Navegar entre los objetos del segmento	Objeto del segmento	Botones de dirección
Navegar al primer elemento del segmento	Objeto del segmento	<Inicio>
Navegar al último elemento del segmento	Objeto del segmento	<Fin>
Navegar al siguiente elemento del segmento	Objeto del segmento	<Tab>
Navegar al elemento anterior del segmento	Objeto del segmento	<Mayús+Tab>
Insertar segmento	Cualquiera	<Ctrl+R>

Navegar en el código del programa (AWL/SCL)

Función	Posición del cursor	Combinación de teclas
Navegar en el código del programa	Línea	Botones de dirección
Una palabra hacia la derecha/izquierda	Línea	<Ctrl+botones de dirección>
Al inicio de la línea	Línea	<Inicio>
Al final de la línea	Línea	<Fin>
Al inicio de una sección del código	Línea	<Ctrl+Inicio>
Al final de una sección del código	Línea	<Ctrl+Fin>
Ir al segmento siguiente (solo AWL)	Título del segmento	<Flecha-hacia-abajo>
Ir al segmento siguiente (solo AWL)	Línea	<Tab> Repita la combinación de teclas hasta que la marca de inserción se encuentre en el siguiente segmento.
Ir al segmento anterior (solo AWL)	Título del segmento	<Flecha-hacia-arriba>
Ir al segmento anterior (solo AWL)	Línea	<Mayús+Tab> Repita la combinación de teclas hasta que la marca de inserción se encuentre en el segmento anterior.
Insertar segmento	Cualquiera	<Ctrl+R>

Insertar instrucciones (KOP)

Función	Objeto seleccionado	Combinación de teclas
Insertar un contacto normalmente abierto	Circuito	<Mayús+F2>
Insertar un contacto normalmente cerrado	Circuito	<Mayús+F3>
Insertar cuadro vacío	Circuito	<Mayús+F5>
Insertar asignación	Circuito	<Mayús+F7>
"Insertar "Abrir rama"	Circuito	<Mayús+F8>
"Insertar "Cerrar rama"	Circuito	<Mayús+F9>

Insertar instrucciones (FUP)

Función	Objeto seleccionado	Combinación de teclas
Insertar asignación	Segmento, entrada o salida	<Mayús+F7>
Insertar cuadro vacío	Segmento	<Mayús+F5>
"Insertar "Abrir rama"	Línea de conexión entre dos cuadros	<Mayús+F8>
Invertir RLO	Segmento, entrada o salida	<Ctrl+Mayús+4>
Insertar entrada	Segmento, entrada o salida	<Ctrl+Mayús+3>

Introducir operandos (KOP/FUP/GRAPH)

Función	Objeto seleccionado	Combinación de teclas
Activar el campo de entrada para el primer operando de la instrucción	Instrucción	<Intro> O bien: <Letra/número cualquiera> Al pulsar la tecla <Intro> se abre un campo de entrada; al pulsar letras o números, estos se introducen en el campo de entrada.
Activar el campo de entrada del operando	Operando	<F2>
Borrar operando	Operando	<Supr>
Definir variable	Operando	<Ctrl+Mayús+I>
Reasignar variable	Operando	<Ctrl+Mayús+P>
Cambiar nombre de la variable	Operando	<Ctrl+Mayús+T>
Introducir operandos	Campo de entrada de operandos	<Letra/número cualquiera>
Confirmar la entrada del operando	Campo de entrada de operandos	<Intro>

Función	Objeto seleccionado	Combinación de teclas
Abrir la función de autocompletar	Campo de entrada de operandos	<Ctrl+I>
Rechazar el cambio actual	Campo de entrada de operandos	<ESC> El campo de entrada se desactiva y se restablece el contenido anterior.

Editar instrucciones (AWL/SCL)

Función	Objeto seleccionado	Combinación de teclas
Aplicar sangría (solo SCL)	Línea	<Tab> o <Ctrl+R>
Anular sangría (solo SCL)	Línea	<Mayús+Tab> o <Ctrl+Mayús+R>
Formatear el texto seleccionado de forma automática (solo SCL)		<Ctrl+Mayús+W>
Abrir el cuadro de diálogo "Opciones de llamada"	Cursor detrás de una llamada de bloque	<Intro>
Definir variable	Operando	<Ctrl+Mayús+I>
Reasignar variable	Operando	<Ctrl+Mayús+P>
Cambiar nombre de la variable	Operando	<Ctrl+Mayús+T>
Expandir/reducir lista de parámetros (solo SCL)	Operando	<Ctrl+Mayús+barra espaciadora>
Expandir/contraer sección de código	Marca de inserción dentro de la sección de código	<Ctrl+Mayús+Num+> <Ctrl+Mayús+Num->
Expandir/contraer todas las secciones de código	Cualquiera	<Ctrl+Mayús+Num*> <Ctrl+Mayús+Num/>
Abrir la función de autocompletar	Cualquiera	<Ctrl+I> o <Ctrl+barra espaciadora>
Activar/borrar marcadores		<Ctrl+Mayús+M>
Ir al marcador siguiente		<Ctrl+Mayús+6>
Ir al marcador anterior		<Ctrl+Mayús+5>
Marcar la selección como comentario	Línea	<Ctrl+Mayús+Y>
Eliminar comentario	Línea	<Ctrl+Mayús+U>

Ventana de programación de GRAPH

Función	Área	Combinación de teclas
Retroceder/avanzar una página	Navegación, vista de etapa individual, vista de cadena, instrucciones permanentes	<Retroceder página>/ <Avanzar página>
Navegar por la navegación	Navegación	<Flecha-hacia-arriba> <Flecha-hacia-abajo>
Expandir/contraer objeto	Navegación	<+> o <Flecha-hacia-derecha>/ <-> o <Flecha-hacia-izquierda>
Cambiar entre la vista de etapa individual y la vista de cadena si hay una etapa o una transición seleccionada	Navegación	<Intro>
Cambiar entre navegación y área de trabajo	Navegación, vista de etapa individual, vista de cadena, instrucciones permanentes	<ALT+F6>
Saltar hasta el primer elemento de un segmento	Vista de etapa individual	<Inicio>
Saltar hasta el último elemento de un segmento	Vista de etapa individual	<Fin>
Cambiar a interlock	Vista de etapa individual	<Ctrl+Inicio>
Cambiar a transición	Vista de etapa individual	<Ctrl+Fin>
Navegar por la estructura	Vista de cadena	Botones de dirección
Saltar a la primera etapa	Vista de cadena	<Inicio> o <Ctrl+Inicio>
Saltar a la última etapa	Vista de cadena	<Fin> o <Ctrl+Fin>
Abrir rama	Vista de cadena	<Mayús+F8>
Cerrar rama	Vista de cadena	<Mayús+F9>
Insertar fin de cadena	Vista de cadena	<Mayús+F7>
Insertar salto	Vista de cadena	<Mayús+F12>
Insertar etapa y transición	Vista de cadena	<Mayús+F5>
Borrar elemento	Vista de cadena	<Supr>
Saltar hasta el primer elemento editable	Instrucciones permanentes	<Inicio>
Saltar hasta el siguiente elemento editable	Instrucciones permanentes	<Tab>
Saltar hasta el último elemento editable	Instrucciones permanentes	<Fin>
Saltar hasta el elemento editable anterior	Instrucciones permanentes	<Mayús+Tab>

Función	Área	Combinación de teclas
Saltar hasta el principio de la celda "Acción"	Acciones	<Inicio>
Saltar hasta el final de la celda "Acción"	Acciones	<Fin>
Insertar nueva acción	Acciones	<Intro>

Observar el programa

Función	Combinación de teclas
Posicionar/borrar punto de parada (AWL, SCL)	<Ctrl+Mayús+F9>
Saltar punto de parada (AWL, SCL)	<Ctrl+Mayús+F10>
Saltar a un bloque subordinado (AWL, SCL)	<Ctrl+Mayús+F11>
Regresar al bloque invocante (AWL, SCL)	<Ctrl+Mayús+F12>
Ejecutar el programa hasta la marca (posición del cursor) (AWL, SCL)	<Ctrl+F3>
Mostrar el estado del programa (AWL, SCL)	<Ctrl+T>
Activar todos los puntos de parada (AWL, SCL)	<Ctrl+Mayús+F2>
Desactivar todos los puntos de parada (AWL, SCL)	<Ctrl+Mayús+F3>
Forzar a 0 (KOP, FUP)	<Ctrl+Mayús+9>
Forzar a 1 (KOP, FUP)	<Ctrl+Mayús+1>
Forzar operando (KOP, FUP)	<Ctrl+Mayús+2>

Consulte también

Manejo del teclado en el TIA Portal (Página 314)

Uso de funciones de proyecto (Página 316)

Organización de ventanas (Página 316)

Editar tablas (Página 322)

Edición de textos (Página 321)

Ampliar la superficie de la ventana de programación

Introducción

Si se visualizan todos los elementos de la aplicación, la superficie de la ventana de programación es relativamente pequeña. Por este motivo, con programas extensos puede ser preciso desplazar a menudo el área de trabajo. Para evitarlo, es posible ocultar o minimizar la visualización de los siguientes componentes de la aplicación y del editor de programación:

- Árbol del proyecto
- Task Cards
- Interfaz del bloque
- Favoritos

- Comentarios
- Segmentos

Nota

Para las Task Cards, el árbol del proyecto y la ventana de inspección se puede utilizar de manera adicional la opción "Contraer automáticamente". Así, las ventanas se cierran automáticamente cuando no se necesitan.

Consulte también: Maximizar y minimizar el área de trabajo

Mostrar y ocultar el árbol del proyecto

Desde el árbol del proyecto se accede a todas las áreas del proyecto. Durante la creación del programa es posible ocultar el árbol del proyecto para ampliar el tamaño de la ventana de programación.

Para mostrar u ocultar el árbol del proyecto, proceda del siguiente modo:

1. Desactive la casilla de verificación "Árbol del proyecto" del menú "Ver" o haga clic en el botón "Contraer" de la barra de título del árbol del proyecto para ocultarlo.
2. Active la casilla de verificación "Árbol del proyecto" del menú "Ver" o haga clic en el botón "Expandir" de la barra de título del árbol del proyecto para mostrarlo.

Abrir y cerrar Task Cards

Las Task Cards se encuentran en el margen derecho de la ventana de programación.

Para abrir o cerrar Task Cards, proceda del siguiente modo:

1. Desactive la casilla de verificación "Task Card" del menú "Ver" o haga clic en el botón "Contraer" de la barra de título de las Task Cards para cerrarlas.
2. Active la casilla de verificación "Task Card" del menú "Ver" o haga clic en el botón "Expandir" de la barra de título de las Task Cards para abrirlas.

Mostrar y ocultar la interfaz del bloque

La interfaz de un bloque se encuentra en el área superior del editor de programación. Durante la programación es posible ocultarla y mostrarla de nuevo en caso necesario.

Para mostrar u ocultar la interfaz del bloque, proceda del siguiente modo:

1. Haga clic en la flecha hacia arriba o hacia abajo situada en el área inferior de la interfaz, dentro de la línea divisoria de ventanas.

Mostrar y ocultar Favoritos

Para mostrar u ocultar los Favoritos en el editor de programación, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Mostrar y ocultar comentarios

En un bloque es posible introducir un comentario para el bloque o para cada segmento. Estos dos tipos de comentarios se muestran y ocultan de distintas maneras.

Para mostrar u ocultar un comentario de bloque, proceda del siguiente modo:

1. Haga clic en el triángulo que aparece al comienzo de la fila que contiene el título del bloque.

Para mostrar u ocultar los comentarios de segmentos, proceda del siguiente modo:

1. Haga clic en el botón "Activar/desactivar comentarios de segmentos" de la barra de herramientas del editor de programación.

Nota

En función del lenguaje de programación utilizado, los comentarios disponibles pueden variar.

Expandir y contraer segmentos

Algunos lenguajes de programación utilizan segmentos. Éstos pueden abrirse o cerrarse según necesidad.

Para expandir o contraer un segmento en particular, proceda del siguiente modo:

1. Si desea expandir un segmento, haga clic en la flecha hacia la derecha antepuesta al título del segmento. Si desea contraer un segmento, haga clic en la flecha hacia abajo antepuesta al título del segmento.

Para expandir o contraer todos los segmentos, proceda del siguiente modo:

1. Haga clic en el botón "Abrir todos los segmentos" o "Cerrar todos los segmentos" de la barra de herramientas del editor de programación.

Nota

Los segmentos no se utilizan en todos los lenguajes de programación.

Consulte también

Sinopsis del editor de programación (Página 2619)

Maximizar y minimizar el área de trabajo (Página 293)

Ajustar los nemónicos

Los bloques pueden programarse utilizando nemónicos alemanes o internacionales. Los nemónicos internacionales están preajustados al abrir el TIA Portal por primera vez. No obstante, pueden cambiarse en todo momento.

Procedimiento

Para ajustar los nemónicos, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General" en la navegación local.
3. Seleccione los nemónicos deseados del grupo "Configuración general".
   Los nemónicos se cambian en todos los bloques.

Mostrar operandos simbólicos y absolutos

Los operandos pueden representarse en el editor de programación de las siguientes maneras:

- Representación simbólica
  En el programa se visualizan los operandos simbólicos. Las direcciones absolutas correspondientes se visualizan en tooltips al situar el puntero del ratón unos instantes sobre el operando.
- Representación absoluta
  En el programa se visualizan las direcciones absolutas. Los operandos simbólicos correspondientes se visualizan en tooltips.
- Representación simbólica y absoluta
  En el programa se visualizan operandos simbólicos y direcciones absolutas.

Requisitos

El editor de programación está abierto.

Procedimiento

Para cambiar la representación de los operandos, proceda del siguiente modo:

1. Haga clic en el botón "Operandos absolutos/simbólicos" de la barra de herramientas del editor de programación.
   Cada vez que se hace clic en el botón se obtiene una representación diferente y se modifica el símbolo del botón.

O bien:

1. Haga clic en la flecha pequeña que aparece junto al botón "Operandos absolutos/ simbólicos" de la barra de herramientas del editor de programación.
   Se abre una lista desplegable.
2. Seleccione la representación deseada de la lista desplegable.
   Se modifica el símbolo del botón.

Consulte también

Fundamentos de los operandos (Página 2280)

Utilizar versiones de instrucciones

Principios básicos de las versiones de instrucciones

Las instrucciones disponibles para programar el programa de usuario se gestionan en las librerías del sistema. Si debido a una actualización se instala una versión nueva de una librería de sistema, se instalarán también las versiones más recientes de las instrucciones de dicha librería, si las hubiere.

Si hay varias versiones de una instrucción, se listan detrás de las instrucción correspondiente en la Task Card "Instrucciones". Si las versiones de la instrucción están ocultas, es posible visualizarlas en la barra de herramientas de la Task Card "Instrucciones" mediante el botón "Mostrar títulos de columna y columnas adicionales". Seguidamente, en la lista desplegable de la columna "Versión" se seleccionan las versiones de las instrucciones que deben utilizarse en el programa. Si no se selecciona ninguna versión se utilizarán las más recientes.

Nota

Tenga en cuenta las indicaciones siguientes:

- Dentro de un dispositivo sólo es posible utilizar una misma versión de una instrucción.
- Si se modifica la versión de una instrucción de la que dependen otras instrucciones, se modificarán también las versiones de las instrucciones dependientes.
- Si se selecciona una versión de una instrucción que no puede ejecutarse en la CPU utilizada, la instrucción se mostrará atenuada. Esto significa que no es posible utilizar dicha versión de la instrucción en la CPU.
- Si se modifica la versión de una instrucción, es necesario compilar el bloque antes de que pueda indicarse el nuevo número de versión en las propiedades de la instrucción.
- Si una instrucción es un bloque en el sistema, el bloque recibe además un número de versión del bloque propio. Este número de versión puede diferir de la versión de la instrucción y también diferir según el tipo de CPU.

Modificaciones en las versiones

Las versiones nuevas pueden ser principales o secundarias. Las versiones principales nuevas, p. ej. 2.0 ó 3.0, incluyen cambios importantes. Por consiguiente, las versiones

principales nuevas pueden provocar cambios en la interfaz del bloque. Las versiones secundarias nuevas, p. ej. 1.3 ó 1.4, incluyen cambios menores o soluciones a errores.

Uso de versiones de instrucciones

Dentro de un dispositivo existe la posibilidad de decidir qué versión de una instrucción debe utilizarse. Si se selecciona una versión distinta de una instrucción, se aplicará la versión nueva en todas las ubicaciones de la instrucción dentro del programa. Estas instrucciones se marcan con un borde rojo en el programa. Seguidamente hay que cargar el programa en el dispositivo para que se utilicen las versiones nuevas de la instrucción.

Utilizar perfiles de instrucciones

Principios básicos de los perfiles de instrucciones

Introducción

El TIA Portal proporciona una serie de instrucciones para la programación del programa de usuario. No obstante, si no desea utilizar todas las instrucciones, puede filtrar las que no utilice. Para ese fin se pueden crear perfiles de instrucciones y definir dentro de cada perfil qué instrucciones deben aparecer en la Task Card "Instrucciones". Dentro de un proyecto se pueden crear varios perfiles de instrucciones, pero en cada caso solo puede estar activo un perfil. A través de librerías globales se pueden intercambiar perfiles de instrucciones con otros usuarios.

Nota

Tenga en cuenta las indicaciones siguientes:

- Si en un bloque se utilizan instrucciones que están excluidas por el perfil activo, se muestra un error al compilar el bloque. Eso puede ocurrir p. ej. con un bloque arrastrado al programa desde una librería.
- Las instrucciones de un perfil que no son soportadas por los productos instalados se borran del perfil en la siguiente ejecución del mismo. Si se transfiere el perfil a un sistema de ingeniería en el que las instrucciones sí que son soportadas por los productos instalados, las instrucciones estarán presentes en el perfil, pero estarán desactivadas. En caso necesario, es posible volver a activarlas en cualquier momento.
- Si se modifica el perfil activo, deberán compilarse nuevamente los bloques del proyecto. Esto es necesario también para desactivar y borrar el perfil activo o para activar un perfil.

Consulte también

Crear un perfil de instrucciones (Página 2633)

Abrir y editar un perfil de instrucciones (Página 2634)

Activar y desactivar un perfil de instrucciones (Página 2635)

Borrar un perfil de instrucciones (Página 2636)

Crear un perfil de instrucciones

Requisitos

La carpeta "Datos comunes > Perfiles de instrucciones" del árbol del proyecto está abierta.

Procedimiento

Para crear un perfil de instrucciones, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar perfil de instrucciones".
   El editor de perfiles de instrucciones se abre y se muestra un perfil nuevo. En el perfil nuevo están activadas todas las instrucciones.
2. Edite el perfil de instrucciones nuevo conforme a las necesidades.

El nombre del perfil nuevo se puede modificar si es necesario. Proceda para ello del siguiente modo:

1. Haga clic con el botón derecho del ratón en el nuevo perfil de instrucciones.
2. Elija el comando "Cambiar nombre" del menú contextual.
3. Introduzca un nombre para el nuevo perfil de instrucciones.

Nota

Si se trata del primer perfil de instrucciones creado, se utilizará como perfil activo. En este caso, hay que compilar todos los bloques del proyecto. Si ya existen otros perfiles de instrucciones, será necesario activar explícitamente el perfil nuevo para que se utilice como perfil activo. En el árbol del proyecto el perfil activo se reconoce por el símbolo.

Consulte también

Principios básicos de los perfiles de instrucciones (Página 2632)

Abrir y editar un perfil de instrucciones (Página 2634)

Activar y desactivar un perfil de instrucciones (Página 2635)

Borrar un perfil de instrucciones (Página 2636)

Abrir y editar un perfil de instrucciones

Una vez que se ha abierto un perfil de instrucciones, se puede editar del siguiente modo:

- Activar y desactivar instrucciones
  Para cada instrucción se puede definir si está permitida o no dentro del perfil de instrucciones.

  Nota

  Tenga en cuenta que existen dependencias entre determinadas instrucciones. Por esta razón, es posible que una única acción active o desactive varias instrucciones. El símbolo de la casilla de verificación permite reconocer en qué carpetas hay instrucciones desactivadas.

- Activar y desactivar versiones de instrucción
  Algunas instrucciones están disponibles en diferentes versiones. Si una instrucción tiene varias versiones, se puede definir para cada versión si está permitida o no dentro del perfil de instrucciones.
- Cambiar un número de bloque
  Si una instrucción es un bloque de función (FB) o una función (FC) internamente en el sistema, la instrucción recibe del sistema un número de bloque determinado. Ese número de bloque se puede cambiar por un número de bloque propio. Para algunas instrucciones existen varias implementaciones dentro de una versión. En ese tipo de instrucciones los números de bloque solo pueden modificarse individualmente para cada implementación.

  Nota

  Si en el programa se utiliza la instrucción de un perfil y el número de bloque definido ya está ocupado en el programa de usuario por otro bloque, el número de bloque definido para la instrucción se cambia por un número de bloque libre.

Requisitos

La carpeta "Datos comunes > Perfiles de instrucciones" del árbol del proyecto está abierta.

Abrir perfiles de instrucciones

Para abrir un perfil de instrucciones, proceda del siguiente modo:

1. Haga clic en el perfil de instrucciones que desea visualizar.
   El perfil se abre en el editor de perfiles de instrucciones.

Editar perfiles de instrucciones

Para editar un perfil en el editor de perfiles de instrucciones, proceda del siguiente modo:

1. En la lista desplegable "Familia de dispositivos" seleccione el dispositivo para el cual desea editar el perfil de instrucciones.
2. En la lista desplegable "Lenguaje" seleccione el lenguaje de programación para el cual desea editar el perfil de instrucciones.

3. Desactive las instrucciones o versiones de instrucción que desea excluir del perfil. Si desactiva una carpeta, se desactivarán todas las instrucciones que contiene.
4. Active las instrucciones o versiones de instrucción que desea permitir dentro del perfil.
5. Si es necesario, asigne números de bloque propios.

Nota

Para CPUs de la serie S7-1200/1500 es posible asignar números hasta 65535. Para CPUs de la serie S7300/400 encontrará las restricciones de los rangos numéricos en el correspondiente manual de la CPU.

Nota

Si se modifica el perfil activo, se precisará un nuevo proceso de compilación para todos los bloques del proyecto.

Consulte también

Principios básicos de los perfiles de instrucciones (Página 2632)

Crear un perfil de instrucciones (Página 2633)

Activar y desactivar un perfil de instrucciones (Página 2635)

Borrar un perfil de instrucciones (Página 2636)

Utilizar versiones de instrucciones (Página 2631)

Activar y desactivar un perfil de instrucciones

Para que el filtrado de las instrucciones de un perfil pueda aplicarse, primero hay que activar el perfil. El perfil activo puede desactivarse en cualquier momento, y entonces la Task Card "Instrucciones" volverá a contener las instrucciones estándar.

Nota

Será necesario un nuevo proceso de compilación para todos los bloques del proyecto.

Requisitos

La carpeta "Datos comunes > Perfiles de instrucciones" del árbol del proyecto está abierta.

Activar un perfil de instrucciones

Para activar un perfil de instrucciones, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el perfil de instrucciones que desea activar.
2. Elija el comando "Activar perfil de instrucciones" del menú contextual.
   El perfil de instrucciones seleccionado será entonces el perfil activo. Así, las instrucciones solo pueden utilizarse conforme a los ajustes de ese perfil.

Desactivar un perfil de instrucciones

Para desactivar el perfil de instrucciones activo, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el perfil de instrucciones que desea desactivar.
2. Elija el comando "Desactivar perfil de instrucciones" del menú contextual.
   Si no hay ningún perfil de instrucciones activo, en la Task Card "Instrucciones" estarán disponibles de nuevo todas las instrucciones.

Consulte también

Principios básicos de los perfiles de instrucciones (Página 2632)

Crear un perfil de instrucciones (Página 2633)

Abrir y editar un perfil de instrucciones (Página 2634)

Borrar un perfil de instrucciones (Página 2636)

Borrar un perfil de instrucciones

Requisitos

La carpeta "Datos comunes > Perfiles de instrucciones" del árbol del proyecto está abierta.

Procedimiento

Para borrar un perfil de instrucciones, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el perfil de instrucciones que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Nota

Si se borra el perfil activo, se precisará un nuevo proceso de compilación para todos los bloques del proyecto.

Resultado

El perfil de instrucciones seleccionado se borra. Si el perfil de instrucciones borrado era el perfil activo, no habrá ningún perfil activo y en la Task Card "Instrucciones" se mostrarán de nuevo todas las instrucciones disponibles de forma estándar.

Consulte también

Principios básicos de los perfiles de instrucciones (Página 2632)

Crear un perfil de instrucciones (Página 2633)

Abrir y editar un perfil de instrucciones (Página 2634)

Activar y desactivar un perfil de instrucciones (Página 2635)

Utilizar la función automática de autocompletar

Principios básicos de la función automática de autocompletar

Función

En la ventana de programación del editor de programación se puede utilizar la función automática de autocompletar para acceder fácilmente a las variables o instrucciones existentes durante la programación. La función automática de autocompletar significa que en un cuadro de diálogo se obtiene una lista específica de contexto, desde la cual se pueden seleccionar las variables e instrucciones deseadas.

Consulte también

Utilizar la función automática de autocompletar en los lenguajes de programación gráficos (Página 2638)

Utilizar la función automática de autocompletar en los lenguajes de programación textuales (Página 2639)

Utilizar la función automática de autocompletar en los lenguajes de programación gráficos

Insertar variables mediante la función automática de autocompletar

Para insertar variables en los lenguajes gráficos de programación mediante la función automática de autocompletar, proceda del siguiente modo:

1. Seleccione un operando de la instrucción al que desee asignar una variable.
   Se abre el campo de entrada del operando. Junto al campo de entrada aparece el botón de la función automática de autocompletar.
2. Haga clic en el botón para la función automática de autocompletar o pulse la combinación de teclas <Ctrl+I>.
   Se abre la función automática de autocompletar. Según el contexto, esta contiene solo las variables locales y globales, los bloques de datos y las multiinstancias permitidos para el operando. Con <Esc> se puede abandonar la función automática de autocompletar en cualquier momento.
3. Seleccione la variable deseada de la lista. Si lo desea, también puede filtrar la lista:
   - Introduzca p. ej. las iniciales de la variable o instrucción que desea insertar. Con cada letra introducida se sigue filtrando la función automática de autocompletar. Si no hay variables o instrucciones que comiencen con las letras introducidas, la función automática de autocompletar se mantiene en la última concordancia.
   - Introduzca # para acceder a las variables locales de la interfaz del bloque.
   - Introduzca " para acceder a las variables globales.
   - Introduzca % para acceder a direcciones absolutas.

   Si la variable es una variable estructurada, un bloque de datos o una multiinstancia, se muestra una flecha al final de la fila. Haga clic en la flecha para visualizar los elementos subordinados. De esta manera se puede navegar hasta el último nivel. Si para el operando está permitida una estructura como tipo de datos, se puede seleccionar "Ninguna entrada" en la lista. De ese modo se asigna al operando la estructura completa como variable. Utilice <Retroceso> para regresar al nivel anterior.
4. Pulse la tecla <Intro> para aplicar la variable.

Consulte también

Principios básicos de la función automática de autocompletar (Página 2637)

Utilizar la función automática de autocompletar en los lenguajes de programación textuales (Página 2639)

Utilizar la función automática de autocompletar en los lenguajes de programación textuales

Insertar variables e instrucciones mediante la función automática de autocompletar

Para insertar variables e instrucciones en los lenguajes de programación textuales mediante la función automática de autocompletar, proceda del siguiente modo:

1. Introduzca las iniciales de la variable o instrucción que desea insertar. Si lo desea, también puede filtrar por el tipo de variable:
   - Introduzca # para acceder a las variables locales de la interfaz del bloque.
   - Introduzca " para acceder a las variables globales.
   - Introduzca % para acceder a direcciones absolutas.

   Se abre la función automática de autocompletar. Según el contexto, esta contiene solo las variables locales y globales, los bloques de datos, las multiinstancias y las instrucciones permitidos en la posición actual. Con <Esc> se puede abandonar la función automática de autocompletar en cualquier momento.
2. Introduzca otras letras de la variable o instrucción que desea insertar. Con <Intro> o <Tab> se aplica la variable o instrucción y se cierra la función de autocompletar.
   Con cada letra introducida se sigue filtrando la función automática de autocompletar. Si no hay variables ni instrucciones que comiencen con las letras introducidas, la función de autocompletar contiene solo los resultados coincidentes hasta entonces.
3. Seleccione la variable o instrucción deseadas de la lista.
   Si una variable es una variable estructurada, un bloque de datos o una multiinstancia, seleccione primero la variable, el bloque de datos o la multiinstancia desde la función de autocompletar y aplique la selección con <Intro>. Para seleccionar los componentes restantes de la estructura, del bloque de datos o de la multiinstancia, introduzca un punto. A continuación se vuelve a abrir la función de autocompletar y se puede seleccionar el siguiente componente.
4. Pulse la tecla <Intro> para aplicar la variable.

Consulte también

Principios básicos de la función automática de autocompletar (Página 2637)

Utilizar la función automática de autocompletar en los lenguajes de programación gráficos (Página 2638)

Configuración general de la programación PLC

Sinopsis de la configuración general

Sinopsis

La tabla siguiente muestra los ajustes generales que se pueden efectuar:

Grupo	Ajuste	Descripción
Vista	con comentarios	Los comentarios de segmento se visualizan.
	Información de variable	En el editor de programación se visualiza información adicional de las variables utilizadas. Si selecciona la opción "Información de la variable con jerarquía", en las variables estructuradas se muestran también los comentarios de los niveles de orden superior de la estructura.
Compilación	Borrar parámetros actuales al sincronizar la interfaz	Los parámetros actuales se borran si el parámetro formal correspondiente se ha borrado del bloque llamado y se ejecuta la función "Actualizar llamada de bloque" o se compila el bloque.
Ajuste predeterminado para bloques nuevos	Verificación CEI	La compatibilidad de operandos en operaciones de comparación y aritméticas se verifica según las normas CEI. Los operandos no compatibles deben convertirse explícitamente.
Configuración avanzada	Definir el título del segmento automáticamente	Define el título de un segmento a partir del comentario del parámetro de salida de la primera instrucción de escritura en el segmento. Consulte también: Insertar título del segmento (Página 2675)
	Mostrar la función automática de autocompletar	La función automática de autocompletar se visualiza.
	Nemónicos	Representación alemana o internacional de las operaciones y operandos

Grupo	Ajuste	Descripción
Cargar sin reinicializar	Reserva de memoria	Define el tamaño de la reserva en la memoria estándar disponible para ampliaciones de interfaz.
Interfaz de bloque/Bloques de datos	Definir "Accesible desde HMI" para nuevos elementos y bloques de datos ARRAY	Activa la opción "Accesible desde HMI" para nuevas variables en la interfaz del bloque y los bloques de datos. Esta opción resulta especialmente útil al trabajar con grandes cantidades de datos en bloques de datos ARRAY. Consulte también: Principios básicos de la programación de bloques de datos (Página 2937)

Consulte también

Estructura de la interfaz del bloque (Página 2642)

Modificar la configuración (Página 2641)

Direcciones y tipos de datos admisibles para variables PLC (Página 2570)

Resumen de los ajustes de impresión (Página 282)

Fundamentos del acceso a bloques (Página 2258)

Activar o anular la verificación CEI (Página 2463)

Utilizar la función automática de autocompletar (Página 2637)

Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Modifique la configuración.

Resultado

El cambio se aplica sin necesidad de guardarlo explícitamente.

Consulte también

Sinopsis de la configuración general (Página 2640)

9.1.4.2 Programar bloques lógicos

Declarar la interfaz del bloque

Estructura de la interfaz del bloque

Introducción

La interfaz contiene las declaraciones de las variables y constantes locales que se utilizan en el bloque. Las variables se dividen en dos grupos:

- Parámetros de bloque que forman la interfaz del bloque para la llamada en el programa.
- Datos locales que sirven para almacenar resultados intermedios.

La declaración de variables permite determinar la interfaz de llamada de un bloque en el programa, así como el nombre y el tipo de datos de las variables y constantes que se desean utilizar en el bloque.

La interfaz de los bloques de función define además la estructura de las instancias que se asignan al bloque de función.

Estructura de la interfaz del bloque

La siguiente figura muestra la estructura de la interfaz del bloque. El número de columnas y secciones varía según el tipo de bloque.

Nombre	Tipo de datos	Valor predet.	Remanencia	Visible en HMI	Comentario
Input				☐	
My_Input1	Bool	false	No remane...	☑	
Output				☐	
My_Output1	Bool	1	Remanente	☑	
InOut				☐	
<Hinzufügen>				☐	
Static				☐	
Temp				☐	
Constant				☐	
My_Constant1	Int	22		☐	

Parámetros de bloque

La tabla siguiente muestra los tipos de parámetros de bloque:

Tipo	Sección	Función	Disponible en
Parámetros de entrada	Input	Parámetros cuyos valores lee el bloque.	Funciones, bloques de función y algunos tipos de bloques de organización
Parámetros de salida	Output	Parámetros cuyos valores escribe el bloque.	Funciones y bloques de función

Tipo	Sección	Función	Disponible en
Parámetros de entrada/salida	InOut	El bloque lee los valores de estos parámetros al efectuar la llamada y los vuelve a escribir en ellos tras la ejecución.	Funciones y bloques de función
Valor de retorno	Return	Valor que se devuelve al bloque que realiza la llamada.	Funciones

Según el tipo del bloque abierto se pueden visualizar más secciones.

Datos locales

La tabla siguiente muestra los tipos de datos locales:

Tipo	Sección	Función	Disponible en
Datos locales temporales	Temp	Variables que sirven para almacenar resultados intermedios temporales. Los datos temporales se conservan solo durante un ciclo. Si utiliza datos locales temporales, se debe asegurar de que los valores se escriben dentro del ciclo en el que desea leerlos. De lo contrario, los valores serán aleatorios.	Funciones, bloques de función y bloques de organización Nota: en los bloques de datos de instancia no se visualizan los datos locales temporales.
Datos locales estáticos	Static	Variables que sirven para almacenar resultados intermedios estáticos en el bloque de datos de instancia. Los datos estáticos se conservan hasta que se vuelven a escribir, también a lo largo de varios ciclos. Los nombres de los bloques que son llamados como multiinstancia en un bloque lógico también se depositan en datos locales estáticos.	Bloques de función
Constante	Constant	Constantes con nombres simbólicos declarados que se utilizan dentro del bloque.	Funciones, bloques de función y bloques de organización Nota: en los bloques de datos de instancia no se visualizan las constantes locales.

Significado de las columnas

En la siguiente tabla se explica el significado de las distintas columnas. Las columnas pueden mostrarse u ocultarse según necesidad. El número de columnas mostradas varía en función de la familia de CPU y del tipo de objeto abierto.

Columna	Significado
	Símbolo en el que se puede hacer clic para arrastrar un elemento mediante Drag & Drop hasta un programa y utilizarlo allí como operando.
Nombre	Nombre del elemento.
Tipo de datos	Tipo de datos del elemento.

Columna	Significado
Offset	Dirección relativa de una variable. Esta columna solo es visible en los bloques con acceso estándar.
Valor predeterminado	Valor que permite predeterminar determinadas variables en la interfaz del bloque lógico o bien valor de una constante local. La indicación del valor predeterminado es opcional para variables. Si no se especifica ningún valor, se utilizará el valor predefinido para el tipo de datos indicado. Por ejemplo, el valor predefinido para BOOL es "false". El valor predeterminado de una variable se aplica como valor de arranque en el respectivo bloque de datos de instancia. Los valores aplicados pueden sustituirse en el bloque de datos de instancia por los valores de arranque específicos de la instancia. Las constantes siempre tienen el valor predeterminado que se declaró en la interfaz del bloque. No se visualizan en los bloques de datos de instancia y por tanto tampoco se les puede asignar valores específicos de la instancia.
Remanencia	Marca una variable como remanente. Los valores de variables remanentes se conservan tras desconectar la alimentación. Esta columna solo es visible en la interfaz de bloques de función con acceso optimizado.
Visible en HMI	Indica si una variable está visible en la lista de selección de HMI mediante un ajuste predeterminado.
Accesible desde HMI	Indica si HMI puede acceder a esta variable en tiempo de ejecución.
Valor de ajuste	Marca una variable como valor de ajuste. Los valores de ajuste son valores que requieren un ajuste fino en la puesta en marcha. Esta columna solo existe en la interfaz de bloques de función.
Comentario	Comentario para documentar el elemento.

Consulte también

Variables (Página 2286)

Palabras clave (Página 2280)

Transferencia de parámetros en la llamada de un bloque (Página 2268)

Tipos de datos válidos en la interfaz de bloque (Página 2647)

Remanencia de las variables locales (Página 2660)

Ajustar los valores de datos durante la puesta en marcha (Página 2962)

Principios básicos de las constantes (Página 2288)

Reglas de la declaración de la interfaz del bloque

Reglas generales de la declaración de la interfaz del bloque

Uso de parámetros de bloque

Para la utilización de los parámetros de bloque dentro de un bloque rigen las reglas siguientes:

- Los parámetros de entrada solo pueden leerse.
- Los parámetros de salida solo pueden escribirse.
- Los parámetros de entrada/salida pueden leerse y escribirse.

Asignar valores predeterminados a los parámetros de bloque

Es posible asignar valores predeterminados a determinados parámetros en la interfaz de los bloques de función. La posibilidad de asignar estos valores depende de la sección de declaración y del tipo de datos de cada parámetro.

La tabla siguiente muestra a qué parámetros puede asignarse un valor predeterminado:

Tipo de parámetro	Sección	Es posible asignar un valor predeterminado		
		Tipos de datos simples	Tipos de datos estructurados	Tipos de parámetros
Parámetros de entrada	Input	X	X	-
Parámetros de salida	Output	X	X	-
Parámetros de entrada/salida	InOut	X	- (1)	-
Datos locales estáticos	Static	X	X	-
Datos locales temporales	Temp	-	-	-
Constantes	Constant	X	-	-

(1) Excepción: en los bloques con acceso optimizado, en determinadas condiciones existe la posibilidad de utilizar tipos de datos PLC como valores predeterminados.

Consulte también: Declarar tipos de datos PLC (UDT) como parámetros actuales para parámetros de entrada/salida (Página 2663)

Consulte también

Variables (Página 2286)

Palabras clave (Página 2280)

Tipos de datos válidos en la interfaz de bloque

Tipos de datos válidos en la interfaz de bloque, en S7-300/400

La tabla siguiente muestra a qué tipos de datos pueden asignarse los parámetros en las diferentes secciones de la interfaz.

Sección	Estándar Tipos de datos	ARRAY STRUCT STRING DT	Tipos de parámetros	VOID	POINTER	ANY
Bloque de organización						
Temp	X	X	-	-	-	X
Constant	X	X(3)	-	-	-	-
Bloque de función						
Input	X	X	X	-	X	X
Output	X	X	-	-	-	-
InOut	X	X(1)	-	-	X	X
Static	X	X	-	-	-	-
Temp	X	X	-	-	-	X
Constant	X	X(3)	-	-	-	-
Función						
Input	X	X(1)	X	-	X	X
Output	X	X(1)	-	-	X	X
InOut	X	X(1)	-	-	X	X
Temp	X	X	-	-	-	X
Return	X	X	-	X	X	X(2)
Constant	X	X(3)	-	-	-	-

(1) STRING solo puede definirse en la longitud estándar de 254 caracteres.
(2) En SCL, ANY no se admite como valor de función.
(3) No se admiten constantes con los tipos de datos ARRAY o STRUCT.

Consulte también

Transferencia de parámetros en la llamada de un bloque (Página 2268)

Tipos de datos (Página 2321)

Tipos de datos válidos en la interfaz de bloque

Tipos de datos válidos en la interfaz de bloque, en S7-1200

La tabla siguiente muestra qué tipos de datos pueden asignarse en los parámetros de las diferentes secciones de la interfaz.

Sección	Estándar Tipos de datos	ARRAY STRUCT STRING DT	VOID	VARIANT
Bloque de organización				
Temp	X	X	-	X
Constant	X	X(2)	-	-
Bloque de función				
Input	X	X	-	X
Output	X	X	-	-
InOut	X	X(1)	-	X
Static	X	X	-	-
Temp	X	X	-	X
Constant	X	X(2)	-	-
Función				
Input	X	X(1)	-	X
Output	X	X(1)	-	X
InOut	X	X(1)	-	X
Temp	X	X	-	X
Return	X	X	X	-
Constant	X	X(2)	-	-
(1) STRING solo puede definirse en la longitud estándar de 254 caracteres. (2) No se admiten constantes con los tipos de datos ARRAY o STRUCT.				

Tipos de datos válidos en la interfaz de bloque, en S7-1500

La tabla siguiente muestra qué tipos de datos pueden asignarse en los parámetros de las diferentes secciones de la interfaz.

Sección	Estándar Tipos de datos	ARRAY STRUCT STRING/ WSTRING DT	Tipos de parámetros	VOID	DB_ANY	POINTER	ANY	VARIANT
Bloque de organización								
Temp	X	X	-(4)	-	X	-	X(3)	X
Constant	X	X (5)	-	-	-	-	-	-

Sección	Estándar Tipos de datos	ARRAY STRUCT STRING/ WSTRING DT	Tipos de parámetros	VOID	DB_ANY	POINTER	ANY	VARIANT
Bloque de función								
Input	X	X	X	-	X	X	X	X
Output	X	X	-	-	X	-	-	-
InOut	X	X(1)	-(4)	-	X	X	X	X
Static	X	X	-	-	X	-	-	-
Temp	X	X	-(4)	-	-	-	X(3)	X
Constant	X	X (5)	-	-	-	-	-	-
Función								
Input	X	X(1)	X	-	X	X	X	X
Output	X	X(1)	-	-	X	X	X	X
InOut	X	X(1)	-(4)	-	X	X	X	X
Temp	X	X	-(4)	-	X	-	X(3)	X
Return	X	X	-	X	X	X	x(2)	-
Constant	X	X (5)	-	-	-	-	-	-

(1) En estas secciones no se pueden indicar declaraciones de longitud para STRING y WSTRING. Estos STRING siempre tienen una longitud estándar de 254 y los WSTRING, de 16832. No se admite una declaración en formato MyString[3]. En estas secciones, WSTRING solo está permitido en bloques con acceso optimizado.

(2) En SCL, ANY no se admite como valor de función.

(3) ANY solo puede utilizarse en bloques con acceso estándar de la sección "Temp".

(4) El tipo de parámetros "INSTANCE" es la única excepción permitida en las secciones "TEMP" e "InOut".

(5) No se admiten constantes con los tipos de datos ARRAY o STRUCT.

Consulte también

Transferencia de parámetros en la llamada de un bloque (Página 2268)

Tipos de datos (Página 2321)

Declarar variables y constantes locales

Declarar variables y constantes locales en la interfaz del bloque

Requisitos

La interfaz del bloque está abierta.

Procedimiento

Para declarar una variable o constante de un tipo de datos simple, proceda del siguiente modo:

1. Seleccione la sección de declaración deseada en la interfaz.
2. En la columna "Nombre", introduzca un nombre para el elemento.
3. Introduzca el tipo de datos deseado en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
4. Introduzca un valor para constantes en la columna "Valor predeterminado".
5. Opcionalmente: Modifique las propiedades que se visualizan en las otras columnas de la interfaz del bloque.

Resultado

Se crea el elemento.

Comprobación de sintaxis

Cada vez que se efectúa una entrada se comprueba la sintaxis. Si hay errores, se destacan en color rojo. No es necesario corregir estos errores inmediatamente, sino que es posible continuar con la edición y efectuar las correcciones necesarias en un momento posterior. No obstante, mientras que la declaración de variables contenga errores de sintaxis, no se podrá compilar el programa.

Nota

Si se modifica la interfaz de un bloque, las llamadas del bloque serán posiblemente incoherentes en el programa. Los puntos de llamada se actualizan automáticamente si es posible.

Si no es posible una actualización automática, se tendrán que actualizar manualmente las llamadas de bloque incoherentes.

Consulte también:

Actualizar llamadas de bloque en KOP (Página 2688)

Actualizar llamadas de bloque en FUP (Página 2732)

Consulte también

Editar tablas (Página 322)

Editar la interfaz del bloque (Página 2664)

Principios básicos de los valores de arranque (Página 2948)

Variables (Página 2286)

Palabras clave (Página 2280)

Propiedades de variables y constantes locales (Página 2658)

Remanencia de las variables locales (Página 2660)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Constantes (Página 2288)

Declarar una variable local en el editor de programación

Requisitos

El editor de programación está abierto.

Procedimiento

Para declarar una variable local, proceda del siguiente modo:

1. Inserte una instrucción en el programa.
   Las cadenas de caracteres "<???>", "<??.?>" o "..." representan comodines de operandos.
2. Reemplace un comodín de operando con el nombre de la variable que se va a crear.
3. Seleccione el nombre del elemento.
   Si desea declarar varios elementos, seleccione los nombres de todos los elementos que desea declarar.
4. Elija el comando "Definir variable" del menú contextual.
   Se abre el cuadro de diálogo "Definir variable". Muestra una tabla de declaración en la que ya figura el nombre del elemento.
5. Para declarar una variable local, seleccione una de las secciones siguientes:
   - Local In
   - Local Out
   - Local InOut
   - Local Static
   - Local Temp
6. Introduzca el tipo de datos y el comentario en las respectivas columnas.
7. Haga clic en el botón "Definir" para finalizar la entrada.

Resultado

La declaración se escribe directamente en la interfaz del bloque y es válida en todo el bloque.

Nota

Si se modifica la interfaz de un bloque, las llamadas del bloque serán posiblemente incoherentes en el programa. Los puntos de llamada se actualizan automáticamente si es posible.

Si no es posible una actualización automática, se tendrán que actualizar manualmente las llamadas de bloque incoherentes.

Consulte también:

Actualizar llamadas de bloque en KOP (Página 2688)

Actualizar llamadas de bloque en FUP (Página 2732)

Consulte también

Editar tablas (Página 322)

Variables (Página 2286)

Palabras clave (Página 2280)

Editar la interfaz del bloque (Página 2664)

Principios básicos de los valores de arranque (Página 2948)

Propiedades de variables y constantes locales (Página 2658)

Remanencia de las variables locales (Página 2660)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Declarar variables del tipo de datos ARRAY

Requisitos

La interfaz del bloque está abierta.

Procedimiento

Para declarar una variable del tipo de datos ARRAY, proceda del siguiente modo:

1. Seleccione la sección de declaración deseada en la interfaz.
2. Introduzca un nombre de variable en la columna "Nombre".
3. Haga clic en el botón para la selección del tipo de datos de la columna "Tipo de datos".
   Se abre una lista de los tipos de datos permitidos.
4. Seleccione el tipo de datos "Array".
   Se abre el cuadro de diálogo "Array".

5. Indique el tipo de datos de los elementos Array en el campo de entrada "Tipo de datos".
6. A continuación, indique el límite inferior y superior de cada dimensión en el campo de entrada "Límites de matriz".
   Ejemplo de un ARRAY unidimensional:
   `[0..3]`
   Ejemplo de un ARRAY tridimensional:
   `[0..3, 0..15, 0..33]`
7. Confirme la entrada.
8. Opcionalmente: Modifique las propiedades de las variables que se visualizan en las otras columnas de la interfaz del bloque.

Resultado

Se crea la variable del tipo de datos ARRAY.

Nota

Para los elementos ARRAY no se pueden definir valores predeterminados específicos. No obstante, es posible asignarles valores de arranque en la instancia.

Consulte también

Utilizar la función automática de autocompletar (Página 2637)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Matriz (Página 2354)

Variables (Página 2286)

Palabras clave (Página 2280)

Propiedades de variables y constantes locales (Página 2658)

Remanencia de las variables locales (Página 2660)

Editar tablas (Página 322)

Declarar variables del tipo de datos STRUCT

Requisitos

La interfaz del bloque está abierta.

Procedimiento

Para declarar una variable del tipo de datos STRUCT, proceda del siguiente modo:

1. Seleccione la sección de declaración deseada en la interfaz.
2. Introduzca un nombre de variable en la columna "Nombre".

3. Introduzca "Struct" en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
   Debajo de las nuevas variables se inserta una fila vacía sangrada.
4. Inserte el primer elemento de estructura en la fila vacía.
   Debajo del elemento se inserta nuevamente una fila vacía.
5. Seleccione un tipo de datos para el elemento de estructura.
6. Opcionalmente: Modifique las propiedades del elemento de estructura que se visualizan en las otras columnas de la interfaz del bloque.
7. Repita los pasos 4 a 7 para todos los demás elementos de estructura.
   No es necesario finalizar explícitamente la estructura. La estructura termina con el último elemento introducido.
8. Para insertar una nueva variable después de la estructura, deje una fila vacía al final de la estructura e introduzca la nueva variable en la segunda fila vacía.

Resultado

Se crea la variable del tipo de datos STRUCT.

Consulte también

Utilizar la función automática de autocompletar (Página 2637)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Variables (Página 2286)

Palabras clave (Página 2280)

Propiedades de variables y constantes locales (Página 2658)

Remanencia de las variables locales (Página 2660)

Editar tablas (Página 322)

Declarar variables basadas en un tipo de datos PLC

Requisitos

En la CPU actual está declarado un tipo de datos PLC.

Procedimiento

Para declarar una variable basada en el tipo de datos PLC, proceda del siguiente modo:

1. Seleccione la sección de declaración deseada en la interfaz.
2. Introduzca el tipo de datos PLC en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.

Resultado

La variable se crea.

Nota

Los valores predeterminados de las variables en un tipo de datos PLC se definen al crear el tipo de datos PLC. En la ubicación del tipo de datos PLC, estos valores no se pueden modificar.

Si modifica o borra tipos de datos PLC utilizados en la interfaz del bloque, la interfaz se volverá incoherente. Para corregir esta incoherencia se debe actualizar la interfaz.

Consulte también: Actualizar la interfaz del bloque (Página 2656)

Consulte también

Editar tablas (Página 322)

Principios básicos de los tipos de datos PLC (Página 2967)

Declarar variables superpuestas

Introducción

Para acceder a determinadas áreas de datos dentro de una variable declarada, se puede superponer a las variables declaradas otra declaración más. De esta manera existe la posibilidad de direccionar una variable ya declarada con otro tipo de datos. Por ejemplo, cada bit de una variable del tipo de datos WORD se puede direccionar con un ARRAY of BOOL.

Superponer variable

Para superponer una variable con un tipo de datos nuevo, proceda del siguiente modo:

1. Abra la interfaz del bloque.
2. En la interfaz, seleccione la variable que desea superponer con un tipo de datos nuevo.
3. Haga clic en el botón "Agregar fila" de la barra de herramientas.
   Debajo de la variable superpuesta se agrega una fila. La variable que se va a superponer debe declararse en la fila inmediatamente después de la variable superpuesta.
4. Introduzca un nombre de variable en la columna "Nombre".

5. Introduzca la entrada "AT" en la columna "Tipo de datos". Se dispone de la ayuda de la función de autocompletar.
   La entrada de la columna "Nombre" se complementa con el siguiente suplemento.
   `"AT<Nombre de la variable superpuesta>"`
6. Haga clic nuevamente en el botón para seleccionar el tipo de datos y seleccione el tipo de datos para la variable nueva.
   La variable se crea. En ella aparecen los mismos datos de la variable superpuesta pero se interpretan con el tipo de datos nuevo.

Eliminar superposición

Para eliminar la superposición de una variable, proceda del siguiente modo:

1. Seleccione la variable superpuesta que desea eliminar.
2. Elija el comando "Borrar" del menú contextual.
3. La superposición se elimina.

Consulte también

Editar tablas (Página 322)

Superponer variables con AT (Página 2302)

Declarar multiinstancias

Requisitos

- El bloque de función que debe llamarse se encuentra disponible en el árbol del proyecto y es apto para multiinstancia.
- La interfaz del bloque de función que efectúa la llamada está abierta.

Procedimiento

Para declarar como multiinstancia un bloque de función que debe llamarse, proceda del siguiente modo:

1. Introduzca una denominación para la llamada de bloque en la columna "Nombre" de la sección "Static".
2. Introduzca en la columna "Tipo de datos" un nombre simbólico para el bloque de función que debe llamarse.

Nota

El editor de programación declara automáticamente la multiinstancia si se programa una llamada de bloque en un segmento y en el cuadro de diálogo siguiente "Opciones de llamada" se indica que el bloque debe llamarse como multiinstancia.

Consulte también

Actualizar la interfaz del bloque (Página 2656)

Actualizar la interfaz del bloque

Introducción

Si modifica o borra tipos de datos PLC o multiinstancias que se utilizan en la interfaz del bloque, la interfaz se volverá incoherente. Para corregir esta incoherencia se debe actualizar la interfaz.

Existen dos posibilidades de actualizar la interfaz del bloque:

- Actualización explícita de la interfaz del bloque.
  Los tipos de datos PLC y multiinstancias que se utilizan se actualizan. Los bloque de datos instancia pertenecientes al bloque no se actualizan implícitamente.
- Actualización implícita durante la compilación.
  Todos los tipos de datos PLC y multiinstancias que se utilizan, así como los bloques de datos instancia correspondientes, se actualizan.

Actualización explícita de la interfaz del bloque

Para actualizar la interfaz del bloque explícitamente, proceda del siguiente modo:

1. Abra la interfaz del bloque.
2. Elija el comando "Actualizar interfaz" del menú contextual.

Actualización implícita durante la compilación

Para actualizar implícitamente durante la compilación todos los usos de los tipos de datos PLC y las multiinstancias, así como los bloques de datos instancia, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Principios básicos de los tipos de datos PLC (Página 2967)

Declarar variables basadas en un tipo de datos PLC (Página 2653)

Editar tablas (Página 322)

Editar la interfaz del bloque (Página 2664)

Principios básicos de los valores de arranque (Página 2948)

Variables (Página 2286)

Palabras clave (Página 2280)

Propiedades de variables y constantes locales (Página 2658)

Remanencia de las variables locales (Página 2660)

Actualizar llamadas de bloque en KOP (Página 2688)

Declarar multiinstancias (Página 2655)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Ampliar una interfaz de bloque

Descripción

Para modificar con posterioridad programas de PLC que ya se han puesto en servicio y que funcionan sin problemas en una instalación, las CPU de la serie S7-1500 y la mayoría de las CPU de la serie S7-1200 V4 permiten ampliar las interfaces de bloques de función durante el funcionamiento.

Los bloques modificados pueden cargarse sin necesidad de pasar la CPU a STOP y sin afectar a los valores de las variables ya cargadas.

De ese modo es posible realizar cambios en el programa con facilidad. El proceso controlado no se ve afectado por esta carga (cargar sin reinicializar).

Modo de funcionamiento

Cada bloque de función dispone de forma predeterminada de una reserva de memoria. En un primer momento, la reserva no se utiliza. Si compila y carga el bloque y después se da cuenta de que desea cargar cambios de interfaz con posterioridad, entonces active la reserva de memoria. Todas las variables declaradas después se cargan en la reserva de memoria. La siguiente carga no influirá en los valores de las variables ya cargadas, por lo que el funcionamiento en curso no se verá afectado.

Para modificar el programa más tarde, mientras la instalación no esté funcionando, se dispone además de la posibilidad de modificar el formato de memoria de uno o varios bloques en un único paso. Todas las variables de la reserva de memoria se mueven al área de memoria habitual. La reserva de memoria se libera de nuevo para otras ampliaciones de interfaz.

Requisitos

La función "Cargar sin reinicializar" está disponible si se cumplen los siguientes requisitos:

- El proyecto está disponible en el formato "TIA Portal V12" o en una versión superior.
- Trabaja con una CPU que admite "Cargar sin reinicializar".
- Los bloques se crearon en KOP, FUP, AWL o SCL.
- Los bloques han sido creados por el usuario, es decir, no son bloques suministrados.
- Se trata de bloques de acceso optimizado.

Pasos fundamentales

Para ampliar la interfaz de un bloque de función y cargar a continuación el bloque sin reinicializar, siga los siguientes pasos.

1. Todos los bloques tienen de forma predeterminada una reserva de memoria de 100 bytes. En caso necesario, el tamaño de la reserva puede adaptarse.
2. Active la reserva de memoria.
3. Amplíe la interfaz del bloque.
4. Compile el bloque.
5. Vuelva a cargar el bloque en la CPU de la forma acostumbrada.

Encontrará descripciones detalladas de cada uno de los pasos en el capítulo "Cargar bloques (S7-1200/1500) ".

Nota

La función "Cargar sin reinicializar" solo está disponible con plena funcionalidad en las CPU de las series S7-1500 y S7-1200 V4.

No obstante, todas las familias de CPU ofrecen los siguientes métodos para ampliar la interfaz de bloques de función y cargar variables de nueva declaración sin ninguna repercusión:

- En la sección "Temp" se pueden agregar variables nuevas y cargarlas sin influir en el proceso.
- En la sección "IOut" se pueden crear variables nuevas de un tipo de datos estructurado y cargarlas sin influir en el proceso.

Editar las propiedades de variables y constantes locales

Propiedades de variables y constantes locales

Propiedades

La tabla siguiente muestra una vista general de las propiedades de las variables y constantes locales:

Grupo	Propiedad	Descripción
General	Nombre	Nombre del elemento.
	Tipo de datos	Tipo de datos del elemento.

Grupo	Propiedad	Descripción
	Valor predeterminado	Valor que permite predeterminar determinadas variables en la interfaz del bloque lógico o bien valor de una constante local. La indicación del valor predeterminado es opcional para variables. Si no se especifica ningún valor, se utilizará el valor predefinido para el tipo de datos indicado. Por ejemplo, el valor predefinido para BOOL es "false". El valor predeterminado de una variable se aplica como valor de arranque en la respectiva instancia. Estos valores aplicados pueden sustituirse por valores de arranque específicos de la instancia.
	Comentario	Comentario sobre el elemento.
Atributos	Remanencia	Marca la variable como remanente. Los valores de variables remanentes se conservan tras desconectar la alimentación. Este atributo solo está disponible en la interfaz de bloques de función con acceso optimizado.
	Accesible desde HMI	Indica si la variable puede utilizarse en HMI. Si el atributo está definido, puede acceder a la variable desde HMI en modo de escritura o lectura. Si el atributo no está definido, no puede acceder a la variable desde HMI. No obstante, tenga en cuenta que con el atributo "Accesible desde HMI" no se consigue una protección de acceso general para la variable. El acceso en modo de escritura o lectura a través de otras aplicaciones puede ser posible aunque el atributo no esté activado.
	Visible en HMI	Indica si la variable está visible en la lista desplegable de HMI mediante un ajuste predeterminado.
	Parámetros ocultos	Indica si debe ocultarse la variable durante la llamada del bloque. Esto solo es posible si se ha indicado un parámetro actual predefinido válido.
	Parámetro actual predefinido	Define un parámetro que debe utilizarse como parámetro actual durante la llamada del bloque.
	Parametrizable	Identifica si un parámetro es parametrizable en CFC.
	Para test	Identifica si un parámetro está registrado para el modo Test de CFC.
	Visible	Identifica si un parámetro está visible en CFC.
	Interconectable	Identifica si un parámetro es interconectable en CFC.
	Habilitar la relectura de variables	Identifica si un parámetro es relevante para la función de CFC "Leer esquema de PLC".
	Textos enumerativos	Asigna un parámetro a una enumeración en CFC.
	Unidad de ingeniería	Asigna un parámetro a una unidad en CFC.
	Límite inferior	Define el valor límite inferior para el parámetro en CFC.
	Límite superior	Define el valor límite superior para el parámetro en CFC.

Consulte también

Remanencia de las variables locales (Página 2660)

Cambiar las propiedades de variables y constantes locales (Página 2661)

Palabras clave (Página 2280)

Remanencia de las variables locales

Introducción

Los bloques de función almacenan sus datos en una instancia. Para impedir la pérdida de estos datos en caso de un corte de alimentación, los datos pueden marcarse como remanentes. Entonces se guardarán en un área de memoria remanente. Las posibilidades de ajustar la remanencia dependen del tipo de acceso configurado para el bloque de función.

Comportamiento remanente en los bloques con acceso estándar

En los bloques con acceso estándar, el comportamiento remanente de las distintas variables no puede ajustarse. Sólo se pueden definir como remanentes en la instancia asignada. Todas las variables que contiene son consideradas entonces remanentes.

Remanencia en el acceso optimizado a bloques

En los bloques de datos con acceso optimizado se puede definir el comportamiento de remanencia de las distintas variables.

En las variables de tipo de datos estructurado, la configuración de remanencia se aplica siempre a la totalidad de la estructura. No se puede ajustar la remanencia individualmente para cada uno de los elementos de la estructura.

En la sección "InOut" no se pueden crear variables remanentes de tipos de datos estructurados. Los parámetros de entrada/salida con tipo de datos estructurado, p. ej. ARRAY, STRUCT o STRING siempre son no remanentes.

Existen los siguientes ajustes:

- Remanente
  Los valores de la variable o estructura están disponibles incluso después de un corte de alimentación.
- No remanente
  Los valores de la variable o estructura se pierden en caso de corte de alimentación.
- Ajustar en IDB
  La remanencia se puede ajustar en el bloque de datos instancia. El ajuste realizado en el bloque de datos instancia rige, sin embargo, de forma centralizada para todas las variables que están marcadas con "Ajustar en IDB".

Consulte también

Propiedades de variables y constantes locales (Página 2658)

Fundamentos del acceso a bloques (Página 2258)

Cambiar las propiedades de variables y constantes locales

Editar las propiedades de un elemento en la interfaz del bloque

Para editar las propiedades de un elemento en la interfaz del bloque, proceda del siguiente modo:

1. Abra la interfaz del bloque.
2. Seleccione el elemento deseado en la tabla.
3. Modifique las entradas en las columnas.

Editar las propiedades de varios elementos en la interfaz del bloque

Las columnas "Remanencia", "Visible en HMI", "Accesible desde HMI" y "Valor de ajuste" pueden activarse y desactivarse simultáneamente para varios elementos seleccionados.

Para cambiar una de estas propiedades para varios elementos, proceda del siguiente modo:

1. Abra la interfaz del bloque.
2. Mantenga pulsada la tecla CTRL.
3. En la columna deseada, seleccione sucesivamente las celdas cuyos valores desee cambiar.
4. Elija los comandos "Activar <propiedad>" o "Desactivar <propiedad>" del menú contextual.

Editar las propiedades en la ventana de propiedades

Para editar las propiedades de una variable o constante en particular, proceda del siguiente modo:

1. Seleccione un elemento de la tabla.
   Las propiedades del elemento se visualizan en la ventana de inspección.
2. Modifique las entradas deseadas en la ventana de inspección.

Cambiar el nombre de variables directamente en el editor de programación

Para cambiar el nombre de uno o varios elementos, proceda del siguiente modo:

1. Seleccione uno o varios elementos en el programa.
2. Elija el comando "Cambiar nombre de la variable" del menú contextual.
   Se abre el cuadro de diálogo "Cambiar nombre de la variable". Muestra una tabla de declaración con los elementos seleccionados.

3. Modifique las entradas de la columna "Nombre".
4. Confirme las entradas realizadas haciendo clic en el botón "Modificar".

Editar el tipo de datos o el comentario en el editor de programación

Para editar el tipo de datos o el comentario de una variable en el editor de programación, proceda del siguiente modo:

1. Seleccione el nombre de la variable.
2. Elija el comando "Reasignar variable" del menú contextual.
   Se abre el cuadro de diálogo "Reasignar variable". En él se visualiza una tabla de declaración.
3. Modifique la entrada en la columna "Tipo de datos" o "Comentario".
4. Haga clic en el botón "Modificar" para confirmar la entrada.

Repercusión en el programa

En caso de modificar el nombre, el tipo de datos o la dirección de una variable o constante, se actualizan automáticamente todas las ubicaciones de las variables en el programa.

Nota

Si se modifica la interfaz de un bloque, puede ocurrir que el programa resulte incoherente. Las incoherencias se actualizan automáticamente si es posible.

Si no es posible una actualización automática, las llamadas incoherentes se marcan en color rojo. En ese caso, las incoherencias deben actualizarse manualmente.

Consulte también:

Actualizar llamadas de bloque en KOP (Página 2688)

Actualizar llamadas de bloque en FUP (Página 2732)

Consulte también

Estructura de la interfaz del bloque (Página 2642)

Editar tablas (Página 322)

Editar la interfaz del bloque (Página 2664)

Propiedades de variables y constantes locales (Página 2658)

Remanencia de las variables locales (Página 2660)

Principios básicos de los valores de arranque (Página 2948)

Variables (Página 2286)

Palabras clave (Página 2280)

Actualizar la interfaz del bloque (Página 2656)

Reglas de la declaración de la interfaz del bloque (Página 2645)

Declarar tipos de datos PLC (UDT) como parámetros actuales para parámetros de entrada/salida

Uso de tipos de datos PLC como parámetros actuales predefinidos

En los bloques con acceso optimizado, en determinadas condiciones existe la posibilidad de declarar tipos de datos PLC como parámetros actuales para parámetros de entrada/salida (InOut). Esto puede ser útil cuando se utilizan bloques de programa como elementos de librería y se desea almacenar junto con el elemento de librería información sobre los parámetros actuales que deben utilizarse.

Además, existe la opción de ocultar en la llamada del bloque parámetros de entrada/salida que tienen un parámetro actual predefinido y válido. En principio, los parámetros ocultos no son visibles en la llamada del bloque, aunque pueden visualizarse utilizando una pequeña flecha situada en el borde inferior del cuadro.

Requisitos

- El parámetro de entrada/salida (InOut) se basa en un tipo de datos PLC o un tipo de datos de sistema.
- El parámetro de entrada/salida tiene el ajuste de remanencia "remanente" o "no remanente". Para parámetros de entrada/salida con el ajuste "Ajustar en IDB" no es posible predefinir parámetros actuales.
- El bloque de programa es un bloque con acceso optimizado.

Procedimiento

Para predefinir el parámetro actual de un parámetro de entrada/salida, proceda del siguiente modo:

1. Abra la interfaz del bloque.
2. Seleccione un parámetro de entrada/salida (InOut) en la interfaz del bloque.
3. Abra la ficha "Propiedades" en la ventana de inspección.
4. Seleccione el grupo "Atributos" en la navegación local.
5. Introduzca el parámetro actual deseado en el campo de entrada "Parámetro actual predefinido".
6. Opcionalmente, seleccione la casilla de verificación "Parámetro oculto".

Resultado

- Hay un parámetro actual predefinido. Si se guarda el bloque de programa como elemento de librería, se incluirá también la información sobre el parámetro actual que debe utilizarse.
- Si se utiliza el elemento de librería en el programa, se comprueba si se puede direccionar el parámetro actual que ha predefinido. En caso afirmativo se utiliza automáticamente como parámetro actual.
- Si no se encuentra el parámetro actual en el programa se notifica un error de sintaxis. El parámetro no se oculta y la transferencia de valores de parámetros debe realizarse manualmente.

Editar la interfaz del bloque

Insertar filas en la tabla

Procedimiento

Para insertar una fila encima de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila sobre la cual desea insertar una fila nueva.
2. Haga clic en el botón "Insertar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva sobre la seleccionada.

Consulte también

Editar tablas (Página 322)

Agregar filas en la tabla

Procedimiento

Para agregar una fila nueva detrás de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila tras la que desea insertar la fila nueva.
2. Haga clic en el botón "Agregar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva y vacía detrás de la fila seleccionada.

Consulte también

Editar tablas (Página 322)

Borrar variables

Procedimiento

Para borrar elementos, proceda del siguiente modo:

1. Seleccione la fila que contiene el elemento que desea borrar. También puede seleccionar varias filas simultáneamente manteniendo pulsada la tecla <Ctrl> y haciendo clic consecutivamente en las filas que desea seleccionar, o bien en la primera y en la última.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Editar tablas (Página 322)

Rellenar celdas consecutivas automáticamente

Es posible transferir el contenido de una o varias celdas de la tabla a las celdas situadas debajo, rellenando así las celdas consecutivas automáticamente.

Si se rellenan automáticamente las celdas de la columna "Nombre", cada uno de los nombres se complementará con un número correlativo. Por ejemplo, el nombre "Motor" se transforma en "Motor_1".

Es posible definir como área de origen una celda individual o varias celdas, o bien filas completas.

Si la tabla abierta contiene menos filas de las que deben rellenarse, en primer lugar se deben insertar más filas vacías.

Requisitos

- La tabla está abierta.
- Se dispone de filas de declaración suficientes.

Procedimiento

Para rellenar celdas consecutivas automáticamente, proceda del siguiente modo:

1. Seleccione las celdas cuyo contenido desea transferir.
2. Haga clic en el símbolo de rellenar situado en la esquina inferior derecha de la celda.
   El puntero del ratón adopta la forma de una cruz con puntas de flecha.
3. Con el botón del ratón pulsado, arrastre el puntero hacia abajo hasta situarlo sobre las celdas que desea rellenar automáticamente.
4. Suelte el botón del ratón.
   Las celdas se rellenan automáticamente.
5. Si ya existen entradas en las celdas que van a rellenarse automáticamente, aparece un cuadro de diálogo. En este diálogo puede indicarse si se quieren sobrescribir las entradas existentes o si se quieren insertar nuevas filas para las variables nuevas.

Consulte también

Editar tablas (Página 322)

Mostrar y ocultar las columnas de la tabla

Las columnas de las tablas pueden mostrase u ocultarse según necesidad.

Procedimiento

Para mostrar u ocultar las columnas de una tabla, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna.
2. Elija el comando "Mostrar/ocultar" del menú contextual.
   Aparece una selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación correspondiente.
4. Para ocultar una columna, desactive la casilla de verificación correspondiente.

Editar variables con editores externos

Para editar variables individuales en editores de tablas, p. ej., en Excel, éstas se pueden copiar y pegar para exportarlas o importarlas. Sin embargo, las variables estructuradas no se pueden copiar en un editor.

Requisitos

Hay una interfaz de bloque y un editor externo abiertos.

Procedimiento

Para exportar distintas variables a un editor externo y volverlas a importar, proceda del siguiente modo:

1. Seleccione una o más variables.
2. Elija el comando "Copiar" del menú contextual.
3. Cambie al editor externo y pegue la variable copiada.
4. Edite las variables de la forma necesaria.
5. Copie las variables en el editor externo.
6. Seleccione las variables en el editor externo.
7. Cambie nuevamente a la interfaz del bloque.
8. Elija el comando "Pegar" del menú contextual.

Crear el código del programa

Crear programas KOP

Principios básicos de KOP

Lenguaje de programación KOP

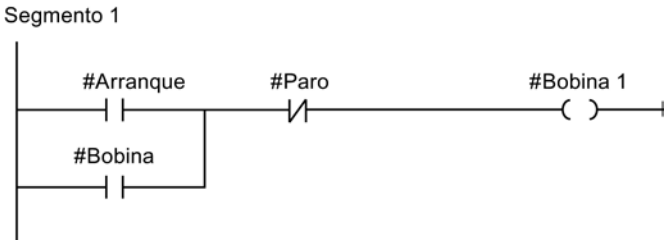
Sinopsis del lenguaje de programación KOP (Esquema de contactos)

KOP es un lenguaje de programación gráfico. Su representación es similar a los esquemas de circuitos.

El programa se mapea en uno o varios segmentos. Un segmento contiene en el margen izquierdo una barra de alimentación de la que parten los circuitos. Las consultas de las señales binarias se disponen en los circuitos en forma de contactos. Si los elementos se disponen en serie en un circuito, se crea una conexión en serie. Si se disponen en ramas paralelas, se crea una conexión en paralelo. Las funciones complejas se representan mediante cuadros.

Ejemplo de segmentos en KOP

La figura siguiente muestra un segmento KOP con dos contactos normalmente abiertos, un contacto normalmente cerrado y una bobina:



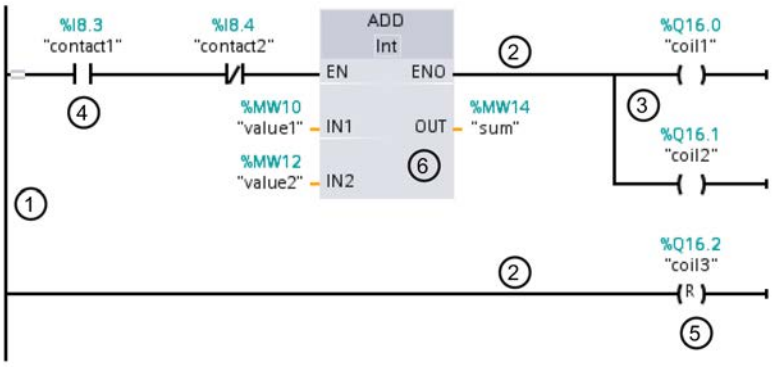
Vista general de los elementos KOP

Elementos KOP

Un programa KOP consta de distintos elementos que pueden conectarse en serie o en paralelo a la barra de alimentación de un segmento. Es preciso asignar variables a la mayoría de los elementos del programa.

Desde la barra de alimentación parte por lo menos un circuito. La programación del segmento comienza en el margen izquierdo del circuito. La barra de alimentación se puede ampliar con varios circuitos y ramas.

La figura siguiente muestra elementos de un segmento KOP a título de ejemplo:



1) Barra de alimentación

2) Circuito

3) Rama

4) Contacto

5) Bobina

6) Cuadro

Barra de alimentación

Todo segmento KOP comprende una barra de alimentación que contiene por lo menos un circuito. Un segmento se puede ampliar insertando circuitos adicionales. En los distintos circuitos es posible programar conexiones en paralelo con ayuda de ramas.

Contactos

Los contactos sirven para establecer o deshacer una conexión conductora entre dos elementos. La corriente fluye de izquierda a derecha. Los contactos permiten consultar el estado lógico o el valor de un operando y controlar el flujo de corriente en función del resultado.

Los siguientes tipos de contactos están disponibles en un programa KOP:

- Contacto normalmente abierto:
  Los contactos normalmente abiertos transmiten la corriente si el estado lógico de un operando binario indicado es "1".
- Contactos normalmente cerrados:
  Los contactos normalmente cerrados transmiten la corriente si el estado lógico de un operando binario indicado es "0".
- Contactos con función adicional:
  Los contactos con función adicional transmiten la corriente si se cumple una condición determinada. Estos contactos también permiten realizar una función adicional, p. ej. una evaluación de flancos o una comparación.

Bobinas

Las bobinas permiten controlar operandos binarios. En función del estado lógico del resultado lógico, las bobinas pueden activar o desactivar un operando binario.

Los siguientes tipos de bobinas están disponibles en un programa KOP:

- Bobinas simples:
  Las bobinas simples activan un operando binario cuando fluye corriente hasta la bobina. Un ejemplo de una bobina simple es la instrucción "Asignación".
- Bobinas con función adicional:
  Además de la evaluación del resultado lógico, estas bobinas ofrecen funciones adicionales. Por ejemplo, las bobinas de evaluación de flancos y control del programa son bobinas con función adicional.

Cuadros

Los cuadros son elementos KOP con funciones complejas. El cuadro vacío es una excepción al respecto. Un cuadro vacío es un comodín en el que puede seleccionarse la instrucción deseada.

Los siguientes tipos de cuadros están disponibles en un programa KOP:

- Cuadros sin mecanismo EN/ENO:
  Un cuadro se ejecuta en función del estado lógico de sus entradas. No es posible consultar el estado de errores de ejecución.
- Cuadros con mecanismo EN/ENO:
  Un cuadro se ejecuta únicamente si la entrada de habilitación "EN" tiene el estado lógico "1". Si el cuadro se ejecuta correctamente, la salida de habilitación "ENO" tendrá el estado lógico "1". Si ocurren errores durante la ejecución, se desactiva la salida de habilitación "ENO".

Las llamadas de bloques lógicos también se representan en el segmento como cuadros con mecanismo EN/ENO.

Consulte también

Reglas para utilizar elementos KOP (Página 2678)

Mecanismo EN/ENO (Página 2539)

Configuración de KOP

Sinopsis de la configuración de KOP

Resumen

La tabla siguiente muestra los ajustes que se pueden efectuar:

Grupo	Ajuste	Descripción
Fuente	Tamaño de fuente	Tamaño de fuente en el editor de programación
Vista	Formato	Compacto o ancho Cambia la distancia vertical entre los operandos y otros objetos (p. ej. operandos y contacto). El cambio efectuado se visualiza sólo tras volver a abrir el bloque.
	Con dirección absoluta	Indicación adicional de las direcciones absolutas
Campo de operando	Ancho máximo	Número máximo de caracteres que pueden introducirse horizontalmente en el campo del operando. Con este ajuste, el diseño de los segmentos se calcula nuevamente.
	Altura máxima	Número máximo de caracteres que pueden introducirse verticalmente en el campo del operando. Con este ajuste, el diseño de los segmentos se calcula nuevamente.

Consulte también

Modificar la configuración (Página 2670)

Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Modifique la configuración.

Resultado

El cambio se aplica sin necesidad de guardarlo explícitamente.

Consulte también

Sinopsis de la configuración de KOP (Página 2670)

Trabajar con segmentos

Utilizar segmentos

Función

El programa de usuario se crea en el bloque, dentro de segmentos. Para poder programar un bloque lógico, éste debe contener como mínimo un segmento. El programa de usuario también se puede dividir en varios segmentos para facilitar la claridad.

Consulte también

Insertar título del segmento (Página 2675)

Introducir el comentario del segmento (Página 2676)

Navegar en segmentos (Página 2677)

Insertar segmentos

Requisitos

Un bloque está abierto.

Procedimiento

Para insertar un segmento nuevo, proceda del siguiente modo:

1. Seleccione el segmento debajo del cual desea insertar un segmento nuevo.
2. Elija el comando "Insertar segmento" del menú contextual.

Resultado

Se inserta un nuevo segmento vacío en el bloque.

Consulte también

Seleccionar segmentos (Página 2672)

Copiar e insertar segmentos (Página 2673)

Borrar segmentos (Página 2673)
Contraer y expandir segmentos (Página 2674)
Insertar título del segmento (Página 2675)
Introducir el comentario del segmento (Página 2676)
Navegar en segmentos (Página 2677)

Seleccionar segmentos

Requisitos

Existe un segmento.

Seleccionar un segmento

Para seleccionar un segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento que desea seleccionar.

Seleccionar varios segmentos

Para seleccionar varios segmentos individuales, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Ctrl>.
2. Haga clic en todos los segmentos que desea seleccionar.

Para seleccionar varios segmentos consecutivos, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Mayús>.
2. Haga clic en el primer segmento que desea seleccionar.
3. Haga clic en el último segmento que desea seleccionar.
   Se seleccionan el primer y último segmento, así como todos los segmentos que se encuentran entre éstos.

Consulte también

Insertar segmentos (Página 2671)
Copiar e insertar segmentos (Página 2673)
Borrar segmentos (Página 2673)
Contraer y expandir segmentos (Página 2674)
Insertar título del segmento (Página 2675)
Introducir el comentario del segmento (Página 2676)
Navegar en segmentos (Página 2677)

Copiar e insertar segmentos

Los segmentos copiados pueden insertarse en el mismo bloque o en un bloque diferente. Los segmentos creados en KOP o FUP también pueden insertarse en los bloques de cada uno de los lenguajes de programación diferentes.

Requisitos

Existe un segmento.

Procedimiento

Para copiar e insertar un segmento, proceda del siguiente modo:

1. Seleccione el o los segmentos que desea copiar.
2. Elija el comando "Copiar" del menú contextual.
3. Seleccione el segmento debajo del cual desea insertar el segmento copiado.
4. Elija el comando "Pegar" del menú contextual.

Consulte también

Insertar segmentos (Página 2671)
Seleccionar segmentos (Página 2672)
Borrar segmentos (Página 2673)
Contraer y expandir segmentos (Página 2674)
Insertar título del segmento (Página 2675)
Introducir el comentario del segmento (Página 2676)
Navegar en segmentos (Página 2677)

Borrar segmentos

Requisitos

Existe un segmento.

Procedimiento

Para borrar un segmento, proceda del siguiente modo:

1. Seleccione el segmento que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Insertar segmentos (Página 2671)
Seleccionar segmentos (Página 2672)
Copiar e insertar segmentos (Página 2673)
Contraer y expandir segmentos (Página 2674)
Insertar título del segmento (Página 2675)
Introducir el comentario del segmento (Página 2676)
Navegar en segmentos (Página 2677)

Contraer y expandir segmentos

Requisitos

Existe un segmento.

Expandir y contraer un segmento

Para abrir un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha situada en la barra de título del segmento.

Para cerrar un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia abajo situada en la barra de título del segmento.

Expandir y contraer todos los segmentos

Para expandir o contraer todos los segmentos, proceda del siguiente modo:

1. Haga clic en el botón "Abrir todos los segmentos" o "Cerrar todos los segmentos" de la barra de herramientas.

Consulte también

Insertar segmentos (Página 2671)
Seleccionar segmentos (Página 2672)
Copiar e insertar segmentos (Página 2673)
Borrar segmentos (Página 2673)
Insertar título del segmento (Página 2675)
Introducir el comentario del segmento (Página 2676)
Navegar en segmentos (Página 2677)

Insertar título del segmento

El título de segmento es la cabecera de un segmento. La longitud del título del segmento está limitada a una línea. Puede introducir el título manualmente o definirlo de forma automática. Si elige el ajuste automático, puede aplicarlo a segmentos individuales o establecer en la configuración que el título del segmento debe definirse de forma automática.

Para la inserción automática del título del segmento se evalúa el comentario del operando de una de las siguientes instrucciones del segmento:

- Asignación
- Activar salida
- Desactivar salida

Para ello, se utiliza la instrucción que aparece primero en el segmento.

El título del segmento solo se inserta automáticamente si se cumplen las siguientes condiciones:

- El segmento todavía no tiene ningún título.
- El operando de la instrucción que se utiliza para el comentario tiene un comentario.

Nota

Al insertar el título del segmento de forma automática, tenga en cuenta las siguientes limitaciones:

- Si modifica el comentario del operando con posterioridad, el título del segmento no se actualiza.
- Si cambia el operando de la instrucción, el título del segmento no se actualiza.
- El título del segmento solo se define mediante las instrucciones de escritura citadas antes.
- Si el operando es del tipo de datos Array, se utiliza el comentario del Array y no los comentarios de los elementos Array.
- No se tienen en cuenta los comentarios de operandos no válidos.

Introducir el título del segmento manualmente

Para introducir un título de segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento.
2. Introduzca el título de segmento.

Definir el título del segmento automáticamente

Para establecer que el título del segmento debe definirse fundamentalmente de forma automática, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Active la casilla de verificación "Establecer título del segmento automáticamente" en el grupo "Configuración avanzada".
   Desde este momento, los títulos de los segmentos se definirán automáticamente si se cumplen las condiciones anteriores.

Para definir automáticamente un título de segmento concreto, proceda del siguiente modo:

1. Haga clic en la barra de título de un segmento con el botón derecho del ratón y seleccione "Segmento <Número del segmento>".
2. Elija el comando "Establecer título del segmento automáticamente" del menú contextual.
   El título del segmento seleccionado se define a partir del comentario del operando si se cumplen las condiciones anteriores.

Consulte también

Utilizar segmentos (Página 2671)

Insertar segmentos (Página 2671)

Seleccionar segmentos (Página 2672)

Copiar e insertar segmentos (Página 2673)

Borrar segmentos (Página 2673)

Contraer y expandir segmentos (Página 2674)

Introducir el comentario del segmento (Página 2676)

Navegar en segmentos (Página 2677)

Introducir el comentario del segmento

Los comentarios de segmentos sirven para documentar el contenido del programa en los distintos segmentos. Por ejemplo, es posible describir la función del segmento, o bien llamar la atención sobre peculiaridades.

Requisitos

Existe un segmento.

Procedimiento

Para introducir un comentario de segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha antepuesta al título del segmento.
2. Si el área de comentarios no resulta visible, haga clic en el botón "Activar/desactivar comentarios de segmento" de la barra de herramientas.
   Se abre el área de comentarios.
3. En el área de comentario, haga clic en "Comentario".
   El texto "Comentario" aparece seleccionado.
4. Introduzca el comentario del segmento.

Consulte también

Utilizar segmentos (Página 2671)

Insertar segmentos (Página 2671)

Seleccionar segmentos (Página 2672)

Copiar e insertar segmentos (Página 2673)

Borrar segmentos (Página 2673)

Contraer y expandir segmentos (Página 2674)

Insertar título del segmento (Página 2675)

Navegar en segmentos (Página 2677)

Navegar en segmentos

Existe la posibilidad de navegar en un bloque directamente hasta una posición determinada.

Procedimiento

Para navegar en un bloque hasta una posición determinada, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la sección de lógica de la ventana de programación.
2. Elija el comando "Ir a > Segmento/línea" del menú contextual.
   Se abre el cuadro diálogo "Ir a".
3. Introduzca el segmento hasta el que desea navegar.
4. Introduzca el número de la fila del segmento hasta el que desea navegar.
5. Confirme la entrada realizada con "Aceptar".

Resultado

Si es posible, se muestra la fila correspondiente. En caso de que el segmento o la fila deseados no existan, se visualiza el último segmento existente, o bien, la última fila existente en el segmento deseado.

Consulte también

Utilizar segmentos (Página 2671)

Insertar segmentos (Página 2671)

Seleccionar segmentos (Página 2672)

Copiar e insertar segmentos (Página 2673)

Borrar segmentos (Página 2673)

Contraer y expandir segmentos (Página 2674)

Insertar título del segmento (Página 2675)

Introducir el comentario del segmento (Página 2676)

Insertar elementos KOP

Reglas para utilizar elementos KOP

Reglas

Tenga en cuenta las reglas siguientes al insertar elementos KOP:

- Todo segmento KOP se debe finalizar con una bobina o un cuadro. Sin embargo, los siguientes elementos KOP no se pueden utilizar para finalizar un segmento:
  - Cuadros de comparación
  - Instrucciones de evaluación de flancos positivos o negativos
- El punto inicial de la rama para una conexión de cuadro debe ser siempre la barra de alimentación. No obstante, en la rama que precede al cuadro pueden existir operaciones lógicas u otros cuadros.
- En ramas paralelas con combinaciones previas sólo pueden insertarse contactos. El contacto para negar el resultado lógico (-|NOT|-) es una excepción. Tanto dicho contacto como las bobinas y los cuadros se pueden utilizar en ramas paralelas si éstas parten directamente de la barra de alimentación.
- A los contactos normalmente cerrados y normalmente abiertos no se les puede asignar constantes, p. ej. con TRUE o FALSE. En cambio, es posible utilizar operandos del tipo de datos BOOL.
- En un segmento sólo se puede insertar una instrucción de salto.
- Sólo se puede insertar una etiqueta por segmento.
- Las instrucciones con evaluación de flancos ascendentes o descendentes no se pueden colocar en el extremo izquierdo del circuito, ya que requieren una operación lógica previa.

Reglas de posicionamiento para CPU S7-1200/1500

La siguiente tabla muestra las instrucciones que sólo pueden posicionarse al final del segmento:

Instrucción		Operación lógica previa necesaria
Nemónicos	Nombre	
SET_BF	Activar mapa de bits	No
RESET_BF	Desactivar mapa de bits	No
JMP	Saltar si RLO = 1	No
JMPN	Saltar si RLO = 0	Sí
JMP_LIST	Definir lista de saltos	No
SWITCH	Distribuidor de saltos	No
RET	Retroceder	No

Reglas de posicionamiento para CPU S7-300/400

La siguiente tabla muestra las instrucciones que sólo pueden posicionarse al final del segmento:

Instrucción		Operación lógica previa necesaria
Nemónicos	Nombre	
S	Activar salida	Sí
R	Desactivar salida	Sí
SP	Arrancar temporizador como impulso	Sí
SE	Arrancar temporizador como impulso prolongado	Sí
SD	Arrancar temporizador como retardo a la conexión	Sí
SS	Arrancar temporizador como retardo a la conexión con memoria	Sí
SF	Arrancar temporizador como retardo a la desconexión	Sí
SC	Poner contador al valor inicial	Sí
CU	Contador ascendente	Sí
CD	Contador descendente	Sí
JMP	Saltar si RLO = 1	No
JMPN	Saltar si RLO = 0	Sí
RET	Retroceder	No
OPN	Abrir bloque de datos global	No
OPNI	Abrir bloque de datos de instancia	No
CALL	Llamar bloque	No
SAVE	Guardar RLO en bit BR	No
MCRA	Activar área MCR	No
MCRD	Desactivar área MCR	No
MCR<	Abrir áreas MCR	No
MCR>	Cerrar áreas MCR	No

Consulte también

Mecanismo EN/ENO (Página 2539)

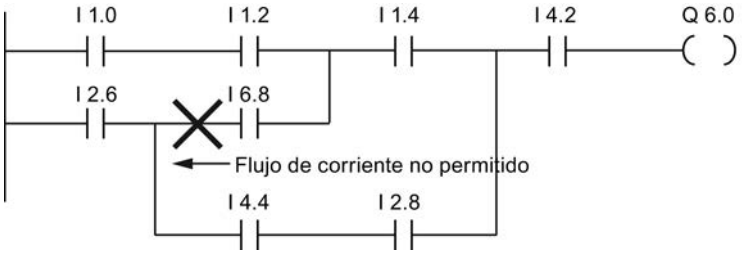
Interconexiones no permitidas en KOP (Página 2680)

Vista general de los elementos KOP (Página 2667)

Interconexiones no permitidas en KOP

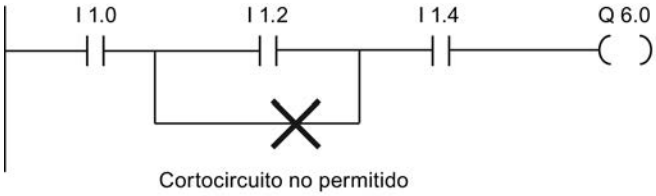
Circulación de corriente de derecha a izquierda

No se pueden programar ramas que puedan causar un flujo invertido de la corriente.



Cortocircuito

No se pueden programar ramas que causen un cortocircuito.



Operaciones lógicas

Para las operaciones lógicas rigen las siguientes normas:

- Sólo las entradas booleanas pueden ir precedidas de una operación lógica.
- Sólo la primera salida booleana se puede combinar mediante una operación lógica.
- Existe sólo un circuito lógico completo por segmento. Los circuitos no interconectados pueden conectarse.

Consulte también

Reglas para utilizar elementos KOP (Página 2678)

Insertar elementos KOP mediante la Task Card "Instrucciones"

Requisitos

Existe un segmento.

Procedimiento

Para insertar un elemento KOP en un segmento mediante la Task Card "Instrucciones", proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Navegue hasta el elemento KOP que desea insertar.
3. Arrastre el elemento hasta la posición deseada del segmento mediante Drag & Drop.
   Si el elemento ejerce de bloque de función (FB) internamente en el sistema, se abre el cuadro de diálogo "Opciones de llamada". En este cuadro de diálogo se puede crear para el bloque de función un bloque de datos de instancia como instancia individual o como multiinstancia en el que se guarden los datos del elemento insertado. El nuevo bloque de datos de instancia se encuentra, después de crearse, en la carpeta "Recursos de programa" del árbol del proyecto, bajo "Bloques de programa > Bloques de sistema". Si ha seleccionado "Multiinstancia", la encontrará en la sección "Static" de la interfaz del bloque.

O bien:

1. Seleccione la posición del segmento en la que desea insertar el elemento.
2. Abra la Task Card "Instrucciones".
3. Haga doble clic en el elemento que desea insertar.
   Si el elemento ejerce de bloque de función (FB) internamente en el sistema, se abre el cuadro de diálogo "Opciones de llamada". En este cuadro de diálogo se puede crear para el bloque de función un bloque de datos de instancia como instancia individual o como multiinstancia en el que se guarden los datos del elemento insertado. El nuevo bloque de datos de instancia se encuentra, después de crearse, en la carpeta "Recursos de programa" del árbol del proyecto, bajo "Bloques de programa > Bloques de sistema". Si ha seleccionado "Multiinstancia", la encontrará en la sección "Static" de la interfaz del bloque.

Resultado

El elemento KOP seleccionado se inserta con comodines para los parámetros.

Insertar elementos KOP mediante un cuadro vacío

Requisitos

Existe un segmento.

Procedimiento

Para insertar un elemento KOP en un segmento mediante un cuadro vacío, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "General > Cuadro vacío".
3. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
4. Sitúe el puntero del ratón sobre el triángulo de la esquina superior derecha del cuadro vacío.
   Se abre una lista desplegable.
5. Seleccione el elemento KOP deseado de la lista desplegable.
   Si el elemento ejerce de bloque de función (FB) internamente en el sistema, se abre el cuadro de diálogo "Opciones de llamada". En este cuadro de diálogo se puede crear para el bloque de función un bloque de datos de instancia como instancia individual o como multiinstancia en el que se guarden los datos del elemento insertado. El nuevo bloque de datos de instancia se encuentra, después de crearse, en la carpeta "Recursos de programa" del árbol del proyecto, bajo "Bloques de programa > Bloques de sistema". Si ha seleccionado "Multiinstancia", la encontrará en la sección "Static" de la interfaz del bloque.

Resultado

El cuadro vacío se transforma en el elemento KOP correspondiente. Para los parámetros se insertan comodines.

Seleccionar el tipo de datos de un elemento KOP

Seleccionar un tipo de datos

Introducción

Algunas instrucciones pueden ejecutarse con varios tipos de datos diferentes. Si en el programa se utiliza alguna de estas instrucciones, en la ubicación concreta del programa es preciso definir un tipo de datos permitido para la instrucción en cuestión. Para algunas instrucciones es necesario seleccionar los tipos de datos para las entradas y para las salidas por separado.

Nota

El sistema predetermina el tipo de datos permitido (BOOL) para las variables de la entrada de habilitación EN y la salida de habilitación ENO. Este tipo de datos no se puede modificar.

Los tipos de datos admisibles para una instrucción se indican en la lista desplegable correspondiente. Seleccionando una entrada de la lista desplegable se determina el tipo de datos de la instrucción. Si el tipo de datos de un operando difiere del tipo de datos de la

instrucción y no puede convertirse de forma implícita, el operando se muestra en color rojo y aparece una ventana desplegable con el mensaje de error correspondiente.

Selección de un tipo de datos de instrucciones matemáticas

Algunas instrucciones matemáticas ofrecen la posibilidad de ajustar automáticamente el tipo de datos adecuado respecto a los tipos de datos de los operandos. En la lista desplegable para seleccionar el tipo de datos, dichas instrucciones tienen, además de los tipos de datos propiamente dichos, la entrada "Auto". Si se selecciona esta entrada y se activa el primer operando, el tipo de datos del operando se utiliza como tipo de datos para la instrucción. La entrada de la lista desplegable cambia a "Auto (<tipo de datos>)", p. ej. "Auto (Real)". Si se activan otros operandos, el tipo de datos de la instrucción, ajustado automáticamente, se adapta según los criterios siguientes:

- Transfieren variables del mismo tipo de datos al resto de operandos:
  El tipo de datos de la instrucción no se modifica.
- Transfieren al resto de operandos variables cuyo tipo de datos es inferior al tipo de datos de la instrucción:
  El tipo de datos de la instrucción no se modifica. Para los operandos del tipo de datos inferior se realiza, si es necesario, una conversión implícita.
- Transfieren a otro operando una variable cuyo tipo de datos es superior al tipo de datos de la instrucción:
  El tipo de datos de la instrucción cambia al tipo de datos superior. Para los operandos que difieren del tipo de datos de la instrucción recién ajustado se realiza, si es necesario, una conversión implícita.

Cualquier modificación del tipo de datos de un operando puede provocar otros cambios en el tipo de datos de la instrucción. De este modo, es posible que otros operandos también se conviertan de forma implícita. Los operandos para los que se realiza una conversión implícita se marcan con un cuadrado gris.

Nota

Tenga en cuenta también la información relativa a la conversión de tipos de datos de su dispositivo y, particularmente en este caso, las indicaciones en materia de verificación CEI.

Consulte también: Conversión de tipos de datos (Página 2373)

Consulte también

Definir el tipo de datos de una instrucción (Página 2683)

Definir el tipo de datos de una instrucción

Introducción

Algunas instrucciones pueden ejecutarse con varios tipos de datos diferentes. Si se insertan este tipo de instrucciones en el programa, es preciso definir el tipo de datos adecuado para ellas en la ubicación concreta del programa.

Definir el tipo de datos mediante la lista desplegable

Para definir el tipo de datos de una instrucción mediante la lista desplegable, proceda del siguiente modo:

1. Arrastre la instrucción mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Haga clic en el triángulo situado en la esquina superior de la lista desplegable.
   Ésta se abre y se visualizan los tipos de datos admisibles para la instrucción.
3. Seleccione un tipo de datos de la lista desplegable.
   Aparece el tipo de datos seleccionado.
4. Si la instrucción dispone de dos listas desplegables, seleccione el tipo de datos de las entradas en la lista desplegable izquierda y el de las salidas en la derecha.

Definir el tipo de datos asignando variables

Para definir el tipo de datos de una instrucción asignando variables, proceda del siguiente modo:

1. Arrastre la instrucción mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Introduzca en una entrada o salida una variable válida cuyo tipo de datos deba adoptarse como tipo de datos de la instrucción.
   El tipo de datos de la variable aparece en la lista desplegable.
3. Si deben definirse tipos de datos tanto para las entradas como para las salidas de la instrucción, indique sendas variables válidas en una entrada y una salida. La variable indicada en la entrada determina el tipo de datos de las entradas y, la indicada en la salida, el tipo de datos de las salidas de la instrucción.

Definir automáticamente el tipo de datos de instrucciones matemáticas

Para definir automáticamente el tipo de datos de las instrucciones matemáticas, proceda del siguiente modo:

1. Arrastre la instrucción matemática mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Seleccione la entrada "Auto" en la lista desplegable.
3. Indique una variable válida en una entrada o salida.
   El tipo de datos de la variable se adoptará como tipo de datos de la instrucción. La entrada de la lista desplegable cambia a "Auto (<tipo de datos>)".

Consulte también: Seleccionar un tipo de datos (Página 2682)

Consulte también

Seleccionar un tipo de datos (Página 2682)

Utilizar Favoritos en KOP

Agregar elementos KOP a los favoritos

Requisitos

- Hay un bloque abierto.
- La Task Card "Instrucciones" tiene configurado el modo multipaleta, o bien los Favoritos se muestran también en el editor.

Procedimiento

Para agregar instrucciones a los Favoritos, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Maximice la paleta "Instrucciones básicas".
3. En la paleta "Instrucciones básicas", navegue hasta la instrucción que desea agregar a los Favoritos.
4. Arrastre la instrucción hasta la paleta "Favoritos" o hasta el área de Favoritos del editor de programación mediante Drag & Drop.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Quitar elementos KOP de los favoritos (Página 2686)

Sinopsis del editor de programación (Página 2619)

Insertar elementos KOP mediante Favoritos

Requisitos

- Hay un bloque abierto.
- Existen Favoritos.

Procedimiento

Para insertar una instrucción en un programa desde los Favoritos, proceda del siguiente modo:

1. Arrastre la instrucción deseada mediante Drag & Drop desde los Favoritos hasta la posición deseada.

O bien:

1. Marque el punto del programa en el que quiere insertar la instrucción.
2. En los Favoritos, haga clic en la instrucción que desea insertar.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Quitar elementos KOP de los favoritos (Página 2686)

Sinopsis del editor de programación (Página 2619)

Quitar elementos KOP de los favoritos

Requisitos

Hay un bloque lógico abierto.

Procedimiento

Para eliminar instrucciones de los Favoritos, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la instrucción que desea eliminar.
2. Elija el comando "Eliminar instrucción" del menú contextual.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Agregar elementos KOP a los favoritos (Página 2685)

Insertar elementos KOP mediante Favoritos (Página 2685)

Sinopsis del editor de programación (Página 2619)

Insertar llamadas de bloque en KOP

Insertar llamadas de bloque mediante Drag & Drop

Las llamadas de funciones (FC) y bloques de función (FB) existentes pueden insertarse desde el árbol del proyecto mediante Drag & Drop. Al llamar bloques de función desde otros bloques de función, es posible hacerlo como instancia individual o multiinstancia. Un bloque de función llamado como instancia individual guarda sus datos en un bloque de datos propio. Un bloque de función llamado como multiinstancia guarda sus datos en el bloque de datos instancia del bloque de función invocante.

Requisitos

- Existe un segmento.
- El bloque que se debe llamar existe.

Insertar una llamada de una función (FC)

Para insertar una llamada de una función (FC) mediante Drag & Drop en un segmento, proceda del siguiente modo:

1. Arrastre la función desde el árbol del proyecto hasta el segmento deseado.

Insertar una llamada de un bloque de función (FB)

Para insertar una llamada de un bloque de función (FB), proceda del siguiente modo:

1. Arrastre el bloque de función desde el árbol del proyecto hasta el segmento deseado. Se abre el cuadro de diálogo "Opciones de llamada".
2. Indique en el cuadro de diálogo si quiere llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca seguidamente un nombre en el campo de entrada "Nombre" para el bloque de datos que se debe asignar al bloque de función.
   - Si hace clic en el botón "Multiinstancia", introduzca seguidamente en el campo de entrada "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
3. Confirme las entradas con "Aceptar".

Resultado

La función o el bloque de función se inserta con sus respectivos parámetros. Los valores de estos parámetros pueden asignarse más tarde.

Consulte también: Transferencia de parámetros en la llamada de un bloque (Página 2268)

Nota

Si al llamar un bloque de función especifica un bloque de datos instancia que no exista aún, éste se creará. Si ha llamado un bloque de función como multiinstancia, éste se introducirá en la interfaz como variable estática.

Consulte también

Actualizar llamadas de bloque en KOP (Página 2688)

Cambiar el tipo de instancia (Página 2689)

Instancias individuales (Página 2265)

Multiinstancias (Página 2266)

Llamadas de bloque (Página 2263)

Actualizar llamadas de bloque en KOP

Si cambian los parámetros de interfaz de un bloque llamado, la llamada no podrá ejecutarse correctamente. Actualice las llamadas de bloque con objeto de evitar llamadas incoherentes.

Existen dos posibilidades de actualizar llamadas de bloque, a saber:

- Actualización explícita en el editor de programación.
  Se actualizan las llamadas de bloque del bloque abierto.
- Actualización implícita durante la compilación.
  Se actualizan todas las llamadas de bloque en el programa, así como los tipos de datos PLC utilizados.

Actualizar llamadas de bloque en el editor de programación

Para actualizar una llamada de bloque en un bloque, proceda del siguiente modo:

1. Abra el bloque en el editor de programación.
2. Haga clic en el botón "Actualizar llamadas de bloque incoherentes" de la barra de herramientas.

O bien:

1. Abra el bloque en el editor de programación.
2. Haga clic con el botón derecho del ratón en la instrucción que contiene la llamada de bloque.
3. Elija el comando "Actualizar" del menú contextual.
   Se abre el cuadro de diálogo "Sincronización de la interfaz". En él se visualizan las diferencias entre la interfaz utilizada y la interfaz modificada del bloque llamado.
4. Si desea actualizar la llamada de bloque, haga clic en "Aceptar". Para cancelar la actualización, haga clic en "Cancelar".

Actualizar llamadas de bloque durante la compilación

Para actualizar implícitamente todas las llamadas de bloque y usos de tipos de datos PLC durante la compilación, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Llamadas de bloque (Página 2263)

Insertar llamadas de bloque mediante Drag & Drop (Página 2687)

Cambiar el tipo de instancia (Página 2689)

Cambiar el tipo de instancia

Tipo de instancia

Los bloques de función se pueden llamar de dos maneras diferentes, a saber:

- Como instancia individual
- Como multiinstancia

Consulte también: Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

El tipo de instancia definido se puede modificar en todo momento.

Requisitos

El programa de usuario contiene una llamada de bloque.

Procedimiento

Para cambiar el tipo de instancia de un bloque de función, proceda del siguiente modo:

1. Abra el bloque lógico y seleccione la llamada de bloque.
2. Elija el comando "Cambiar instancia" del menú contextual.
   Se abre el cuadro de diálogo "Opciones de llamada".

3. Haga clic en el botón "Instancia individual" o "Multiinstancia".
   - Si selecciona el tipo de instancia "Instancia individual", introduzca un nombre para el bloque de datos que debe asignarse al bloque de función.
   - Si selecciona el tipo de instancia "Multiinstancia", introduzca en el campo de texto "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
4. Confirme las entradas con "Aceptar".

Nota

Las instancias individuales y multiinstancias anteriores no se borran automáticamente.

Consulte también

Llamadas de bloque (Página 2263)

Insertar llamadas de bloque mediante Drag & Drop (Página 2687)

Actualizar llamadas de bloque en KOP (Página 2688)

Insertar instrucciones KOP complejas

Utilizar la instrucción "Calcular"

Requisitos

Existe un segmento.

Procedimiento

Para utilizar la instrucción "Calcular", proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "Funciones matemáticas > CALCULATE".
3. Arrastre el elemento hasta la posición deseada del segmento mediante Drag & Drop.
   La instrucción "Calcular" se inserta con una expresión comodín y signos de interrogación para el tipo de datos.
4. Introduzca el tipo de datos para el cálculo.

5. Introduzca los operandos para el cálculo.

Nota

El cálculo se lleva a cabo con las entradas de la instrucción "Calcular". Por ello, si desea utilizar constantes debe insertar también las entradas correspondientes para las constantes.

6. Haga clic en el botón "Editar instrucción "Calcular"" para reemplazar la expresión comodín por la expresión correcta.
   Se abre el cuadro de diálogo "Editar instrucción "Calcular"".
7. Introduzca la expresión deseada en el campo de texto "OUT:= ".

Nota

En el área "Ejemplo" dispone de un ejemplo de expresión válida y de las posibles instrucciones que se pueden utilizar.

P. ej., para determinar un valor mediante el teorema de Pitágoras introduzca la expresión "OUT := SQRT (SQR (IN1) + SQR (IN2))".

8. Confirme las entradas con "Aceptar".

Consulte también

CALCULATE: Calcular (Página 3396)

Utilizar comentarios libres

Principios básicos de la utilización de comentarios libres en KOP

Introducción

Los comentarios libres permiten comentar el código fuente de los lenguajes de programación gráficos, de manera similar a los comentarios de línea de los lenguajes textuales.

Los comentarios libres pueden utilizarse para los elementos siguientes:

- Cuadros
- Bobinas

Consulte también

Insertar comentarios libres (Página 2692)

Editar comentarios libres (Página 2692)

Borrar comentarios libres (Página 2693)

Insertar comentarios libres

Requisitos

Existe un segmento con instrucciones.

Procedimiento

Para añadir un comentario libre relativo a una instrucción, proceda del siguiente modo:

1. Si es preciso, pulse el botón "Mostrar/ocultar comentarios libres" de la barra de herramientas.
2. Haga clic con el botón derecho del ratón en la instrucción para la que desea insertar un comentario libre.
3. Elija el comando "Insertar comentario" del menú contextual.
   Se abre un cuadro de comentario con un comentario estándar. El cuadro de comentario está unido con la instrucción correspondiente por medio de una flecha.
4. Introduzca el comentario deseado en el cuadro de comentario.

Consulte también

Principios básicos de la utilización de comentarios libres en KOP (Página 2691)

Editar comentarios libres (Página 2692)

Borrar comentarios libres (Página 2693)

Editar comentarios libres

Introducción

Los comentarios libres pueden editarse como se indica a continuación:

- Modificar el texto del comentario
- Modificar la posición y el tamaño del cuadro de comentario
- Acoplar el comentario a un elemento diferente
- Mostrar y ocultar comentarios libres

Modificar el texto del comentario

Para modificar el texto de un comentario libre, proceda del siguiente modo:

1. Haga clic en el cuadro de comentario.
2. Introduzca el texto deseado.

Modificar la posición del cuadro de comentario

Para modificar la posición del cuadro de comentario, proceda del siguiente modo:

1. Haga clic en el cuadro de comentario y mantenga oprimido el botón del ratón.
2. Arrastre el cuadro de comentario hasta la posición deseada.

Modificar el tamaño del cuadro de comentario

Para modificar el tamaño del cuadro de comentario, proceda del siguiente modo:

1. Haga clic en el cuadro de comentario.
2. Arrastre el controlador de tamaño de la esquina inferior derecha del cuadro de comentario hasta alcanzar el tamaño deseado.

Acoplar el comentario a un elemento diferente

Para acoplar un comentario libre a un elemento diferente, proceda del siguiente modo:

1. Haga clic en la punta de la flecha que une el cuadro de comentario con la instrucción y mantenga oprimido el botón del ratón.
2. Arrastre la flecha hasta el elemento al que desea acoplar el comentario. Los puntos de inserción posibles se identifican con un cuadrado verde.
3. Suelte el botón del ratón.

Mostrar y ocultar comentarios libres

Para mostrar y ocultar los comentarios libres, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar comentarios libres" de la barra de herramientas.

Consulte también

Principios básicos de la utilización de comentarios libres en KOP (Página 2691)

Insertar comentarios libres (Página 2692)

Borrar comentarios libres (Página 2693)

Borrar comentarios libres

Procedimiento

Para borrar un comentario libre, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el comentario libre que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Principios básicos de la utilización de comentarios libres en KOP (Página 2691)
Insertar comentarios libres (Página 2692)
Editar comentarios libres (Página 2692)

Editar elementos KOP

Seleccionar elementos KOP

Se pueden seleccionar varios elementos individuales o todos los elementos de un segmento.

Requisitos

Existen elementos KOP.

Seleccionar varios elementos KOP individuales

Para seleccionar varios elementos KOP individuales, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Ctrl>.
2. Haga clic en los elementos KOP que desea seleccionar.
3. Suelte la tecla <Ctrl>.

Seleccionar todos los elementos KOP de un segmento

Para seleccionar todos los elementos KOP de un segmento, proceda del siguiente modo:

1. Haga clic en el segmento cuyos elementos desea seleccionar.
2. Elija el comando "Seleccionar todo" del menú "Edición" o utilice la combinación de teclas <Ctrl+E>.

Consulte también

Copiar elementos KOP (Página 2695)
Cortar elementos KOP (Página 2695)
Insertar elementos KOP desde el portapapeles (Página 2696)
Reemplazar elementos KOP (Página 2697)
Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)
Quitar entradas y salidas (Página 2698)

Activar y desactivar el mecanismo EN/ENO (Página 2699)
Borrar elementos KOP (Página 2700)

Copiar elementos KOP

Requisitos

Existe un elemento KOP.

Procedimiento

Para copiar un elemento KOP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el elemento KOP que desea copiar.
2. Elija el comando "Copiar" del menú contextual.

Resultado

El elemento KOP se copia y se guarda en el portapapeles.

Consulte también

Seleccionar elementos KOP (Página 2694)
Cortar elementos KOP (Página 2695)
Insertar elementos KOP desde el portapapeles (Página 2696)
Reemplazar elementos KOP (Página 2697)
Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)
Quitar entradas y salidas (Página 2698)
Activar y desactivar el mecanismo EN/ENO (Página 2699)
Borrar elementos KOP (Página 2700)

Cortar elementos KOP

Requisitos

Existe un elemento KOP.

Cortar

Para cortar un elemento KOP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el elemento KOP que desea cortar.
2. Seleccione el comando "Cortar" del menú contextual.

Resultado

El elemento KOP se corta y se guarda en el portapapeles.

Consulte también

Seleccionar elementos KOP (Página 2694)
Copiar elementos KOP (Página 2695)
Insertar elementos KOP desde el portapapeles (Página 2696)
Reemplazar elementos KOP (Página 2697)
Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)
Quitar entradas y salidas (Página 2698)
Activar y desactivar el mecanismo EN/ENO (Página 2699)
Borrar elementos KOP (Página 2700)

Insertar elementos KOP desde el portapapeles

Requisitos

Existe un elemento KOP.

Procedimiento

Para insertar un elemento KOP desde el portapapeles, proceda del siguiente modo:

1. Copie o corte un elemento KOP.
2. Haga clic con el botón derecho del ratón en la posición del segmento en la que desea insertar el elemento.
3. Elija el comando "Pegar" del menú contextual.

Consulte también

Seleccionar elementos KOP (Página 2694)
Copiar elementos KOP (Página 2695)
Cortar elementos KOP (Página 2695)
Reemplazar elementos KOP (Página 2697)
Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)
Quitar entradas y salidas (Página 2698)
Activar y desactivar el mecanismo EN/ENO (Página 2699)
Borrar elementos KOP (Página 2700)

Reemplazar elementos KOP

Los elementos KOP se pueden reemplazar muy fácilmente por otros del mismo tipo. La ventaja es que los parámetros se conservan, por lo que no es necesario introducirlos de nuevo. Por ejemplo, es posible sustituir un contacto normalmente abierto por uno normalmente cerrado o un flipflop de desactivación/activación por uno de activación/desactivación.

Requisitos

Existe un segmento con un elemento KOP como mínimo.

Procedimiento

Para reemplazar un elemento KOP por otro, proceda del siguiente modo:

1. Seleccione el elemento KOP que desea reemplazar.
2. Sitúe el puntero del ratón sobre el triángulo de la esquina superior derecha del elemento KOP.
   Se abre una lista desplegable.
3. Seleccione el elemento KOP de la lista desplegable por el que desea reemplazar el elemento KOP actual.

Consulte también

Seleccionar elementos KOP (Página 2694)

Copiar elementos KOP (Página 2695)

Cortar elementos KOP (Página 2695)

Insertar elementos KOP desde el portapapeles (Página 2696)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Activar y desactivar el mecanismo EN/ENO (Página 2699)

Borrar elementos KOP (Página 2700)

Insertar entradas y salidas adicionales en los elementos KOP

Introducción

Mediante entradas adicionales es posible añadir elementos KOP que realicen instrucciones aritméticas conmutativas. Estos elementos son p. ej. las instrucciones "Sumar" (ADD) y "Multiplicar" (MUL). Los cuadros de las instrucciones "Copiar valor" (MOVE) y "Desmultiplexar" (DEMUX) se pueden ampliar mediante salidas adicionales.

Requisitos

Existe un elemento KOP que permite insertar entradas o salidas adicionales.

Insertar una entrada adicional

Para insertar una entrada adicional en el cuadro de un elemento KOP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una entrada existente del elemento KOP.
2. Elija el comando "Insertar entrada" del menú contextual.
   El cuadro del elemento KOP se complementa con una entrada adicional.

O bien:

1. Haga clic en el asterisco amarillo junto a la última entrada del cuadro de instrucción.
   El cuadro del elemento KOP se complementa con una entrada adicional.

Insertar una salida adicional

Para insertar una salida adicional en el cuadro de un elemento KOP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una salida existente del elemento KOP.
2. Elija el comando "Insertar salida" del menú contextual.
   El cuadro del elemento KOP se complementa con una salida adicional.

O bien:

1. Haga clic en el asterisco amarillo junto a la última entrada del cuadro de instrucción.
   El cuadro del elemento KOP se complementa con una salida adicional.

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar elementos KOP (Página 2694)

Copiar elementos KOP (Página 2695)

Cortar elementos KOP (Página 2695)

Insertar elementos KOP desde el portapapeles (Página 2696)

Reemplazar elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Activar y desactivar el mecanismo EN/ENO (Página 2699)

Borrar elementos KOP (Página 2700)

Quitar entradas y salidas

Introducción

Las entradas y salidas que se han insertado adicionalmente para una instrucción pueden quitarse nuevamente.

Requisitos

Existe un elemento KOP en el que se han insertado entradas o salidas adicionales.

Quitar una entrada

Para quitar una entrada, proceda del siguiente modo:

1. Seleccione la entrada que desea quitar.
2. Elija el comando "Borrar" del menú contextual.
   La entrada del elemento KOP se quita.

Quitar una salida

Para quitar una salida, proceda del siguiente modo:

1. Seleccione la salida que desea quitar.
2. Elija el comando "Borrar" del menú contextual.
   La salida del elemento KOP se quita.

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar elementos KOP (Página 2694)

Copiar elementos KOP (Página 2695)

Cortar elementos KOP (Página 2695)

Insertar elementos KOP desde el portapapeles (Página 2696)

Reemplazar elementos KOP (Página 2697)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Activar y desactivar el mecanismo EN/ENO (Página 2699)

Borrar elementos KOP (Página 2700)

Activar y desactivar el mecanismo EN/ENO

En KOP y FUP, determinadas instrucciones poseen una salida de habilitación ENO (enable output) y utilizan por consiguiente el mecanismo EN/ENO. Dicho mecanismo permite consultar errores en tiempo de ejecución en instrucciones y reaccionar a ellos. Para aumentar el rendimiento de la CPU, el mecanismo EN/ENO está desactivado por defecto. Por consiguiente, en un principio no existe la posibilidad de reaccionar a errores en tiempo de ejecución de la instrucción a través del valor ENO. Sin embargo, el mecanismo EN/ENO se puede activar en cualquier momento si fuera necesario.

El mecanismo EN/ENO se puede activar individualmente para cada instrucción con el fin de generar el ENO. Si se activa el mecanismo EN/ENO para una instrucción, el resto de instrucciones que se arrastren al programa posteriormente se insertarán con el mecanismo EN/ENO también activado. Si no desea utilizar la evaluación de ENO para una instrucción, el

mecanismo EN/ENO puede volver a desactivarse en cualquier momento. Cualquier otra instrucción que se arrastre al programa posteriormente se insertará sin el mecanismo EN/ENO.

Consulte también: Principios básicos del mecanismo EN/ENO (Página 2539)

Activar el mecanismo EN/ENO

Para activar el mecanismo EN/ENO de una instrucción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el programa sobre la instrucción para la que desea activar el mecanismo EN/ENO.
2. Elija el comando "Generar ENO" del menú contextual.
   Para esa instrucción se vuelve a generar el valor ENO. Cualquier otra instrucción añadida se insertará con salida de habilitación.

Desactivar el mecanismo EN/ENO

Para desactivar el mecanismo EN/ENO de una instrucción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el programa sobre la instrucción para la que desea desactivar el mecanismo EN/ENO.
2. Elija el comando "No generar ENO" del menú contextual.
   Para esa instrucción ya no se generará más el valor ENO. Cualquier otra instrucción añadida se insertará sin salida de habilitación.

Consulte también

Seleccionar elementos KOP (Página 2694)

Copiar elementos KOP (Página 2695)

Cortar elementos KOP (Página 2695)

Insertar elementos KOP desde el portapapeles (Página 2696)

Reemplazar elementos KOP (Página 2697)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Borrar elementos KOP (Página 2700)

Borrar elementos KOP

Requisitos

Existe un elemento KOP.

Procedimiento

Para borrar un elemento KOP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el elemento KOP que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Insertar elementos KOP (Página 2678)
Seleccionar elementos KOP (Página 2694)
Copiar elementos KOP (Página 2695)
Cortar elementos KOP (Página 2695)
Insertar elementos KOP desde el portapapeles (Página 2696)
Reemplazar elementos KOP (Página 2697)
Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)
Quitar entradas y salidas (Página 2698)
Activar y desactivar el mecanismo EN/ENO (Página 2699)

Cablear operandos en instrucciones KOP

Cablear operandos

Cuando se inserta un elemento KOP, las cadenas de caracteres "<???>", "<??.?>" y "..." se insertan como comodines de los parámetros. Las cadenas de caracteres "<???>" y "<??.?>" representadas en color rojo identifican los parámetros que se deben cablear. La secuencia de caracteres "..." representada en color negro identifica los parámetros que se pueden cablear. "<??.?>" representa comodines booleanos.

Nota

Al situar el puntero del ratón sobre el comodín se muestra el tipo de datos esperado.

Requisitos

Existe un elemento KOP.

Procedimiento

Para cablear los parámetros de un elemento KOP, proceda del siguiente modo:

1. Haga doble clic en el comodín del parámetro.
   Se abre un campo de entrada y el comodín aparece seleccionado.
2. Introduzca el parámetro correspondiente.

   Nota

   Si introduce la dirección absoluta de un parámetro ya definido, ésta se sustituirá por el nombre simbólico del parámetro tan pronto como confirme la entrada. Si el parámetro no se ha definido aún, en la tabla de variables PLC se introduce una variable nueva con esta dirección absoluta y el nombre estándar "Tag_<n>". Cuando confirme la entrada, la dirección absoluta se sustituirá por el nombre simbólico "Tag_<n>".

3. Confirme el parámetro pulsando la tecla Intro.
4. Si todavía no ha definido el parámetro, puede hacerlo mediante el menú contextual directamente en el editor de programación.
   Consulte también:
   Declarar variables PLC en el editor de programación (Página 2577)
   Declarar variables locales en el editor de programación (Página 2650)

O mediante Drag & Drop desde la tabla de variables PLC:

1. En el árbol del proyecto, seleccione la carpeta "Variables PLC" y abra la tabla de variables PLC.
2. Si está abierta la tabla de variables PLC, arrastre el símbolo desde la primera columna de la variable deseada hasta la respectiva posición del programa. Si la tabla de variables PLC no está abierta, abra la vista detallada. Arrastre la variable deseada desde la vista detallada hasta la respectiva posición del programa.

O mediante Drag & Drop desde la interfaz del bloque:

1. Abra la interfaz del bloque.
2. Arrastre el operando deseado desde la interfaz del bloque hasta la ventana de instrucciones.

Resultado

- Si la sintaxis es correcta, el parámetro se representará en color negro. El editor salta al siguiente comodín.
- Si hay un error de sintaxis, no se saldrá del campo de entrada y aparecerá un aviso de error correspondiente en la barra de estado. Si pulsa nuevamente la tecla Intro, el campo de entrada se cerrará y la entrada errónea se representará en rojo y en cursiva.

Consulte también

Utilizar y direccionar operandos (Página 2280)

Utilizar la función automática de autocompletar (Página 2637)

Cablear parámetros ocultos

Introducción

Dependiendo de la CPU utilizada, en el programa de usuario se pueden usar instrucciones complejas, suministradas con el TIA Portal. Estas instrucciones pueden contener parámetros declarados como ocultos.

Si una instrucción contiene parámetros ocultos, aparece una flecha pequeña en el margen inferior del cuadro de la instrucción. Los parámetros ocultos se reconocen porque se muestran en fuente blanca.

Los parámetros ocultos se pueden mostrar y cablear en todo momento.

Mostrar u ocultar parámetros ocultos

Para mostrar u ocultar parámetros ocultos, proceda del siguiente modo:

1. Para mostrar los parámetros ocultos, haga clic en la flecha hacia abajo situada en el margen inferior del cuadro de la instrucción.
2. Para ocultar los parámetros ocultos, haga clic en la flecha hacia arriba situada en el margen inferior del cuadro de la instrucción.

Cablear parámetros ocultos

Para cablear parámetros ocultos, proceda del siguiente modo:

1. Cablee los parámetros ocultos de la misma manera que los parámetros visibles.
   El parámetro oculto se convierte en un parámetro visible.

Consulte también

Utilizar librerías (Página 427)

Mostrar u ocultar informaciones de variables

Introducción

Se puede mostrar la siguiente información sobre las variables empleadas en el editor de programación:

- Nombre de la variable
- Dirección de la variable
- Comentarios simples o jerárquicos para la documentación de las variables

La información se recoge de la interfaz del bloque en el caso de las variables locales y de DB, y de la tabla de variables PLC en el caso de las variables válidas para toda la CPU.

Es posible visualizar la información de las variables para todos los bloques o solo para bloques abiertos individuales. Si selecciona la información de variables para todos los bloques, se muestra la información de las variables de todos los bloques abiertos actuales y futuros.

La información de las variables se puede volver a ocultar en cualquier momento. Si se ha ocultado la información de variable de todos los bloques, se puede volver a ver para bloques abiertos individualmente.

Si selecciona la visualización de la información de variable con comentarios jerárquicos, en las variables estructuradas se muestran también los comentarios de los niveles de orden superior de la estructura. En este caso, la información se muestra entre paréntesis detrás del comentario de las variables y los comentarios de los distintos niveles se separan con un punto. Si no existe ningún comentario en un nivel de la estructura de una variable, se suprime de la visualización, lo que se puede detectar porque aparecen dos puntos consecutivos.

Mostrar u ocultar información de variables para todos los bloques

Para mostrar u ocultar la información de variables para todos los bloques, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Si desea visualizar la información de variable, seleccione en la lista desplegable "Información de la variable" bien la opción "Mostrar", bien la opción "Información de la variable con comentarios jerárquicos", en función de si desea visualizar comentarios simples o jerárquicos.
4. Si desea ocultar la información de variable, seleccione la opción "Ocultar" en la lista desplegable "Información de variable".
   La información de las variables se muestra u oculta para todos los bloques abiertos. Si abre más bloques, la información de las variables se mostrará u ocultará dependiendo del ajuste seleccionado.

Mostrar u ocultar información de variables para un bloque abierto

Para mostrar u ocultar la información de variables para un bloque abierto, proceda del siguiente modo:

1. Si desea visualizar la información de variable, seleccione en la lista desplegable "Muestra información sobre la variable." bien la opción "Mostrar información de la variable", bien la opción "Información de la variable con comentarios jerárquicos", en función de si desea visualizar comentarios simples o jerárquicos.
2. Si desea ocultar la información de variable, seleccione la opción "Ocultar información de la variable" en la lista desplegable "Muestra información sobre la variable".
   Se muestra u oculta la información sobre las variables.

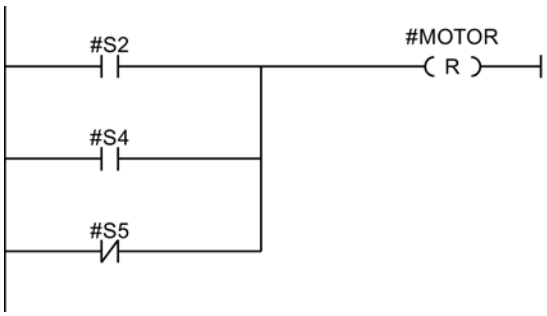
Ramas en KOP

Principios básicos de las ramas en KOP

Definición

En el lenguaje de programación Esquema de contactos (KOP) se utilizan ramas para programar conexiones en paralelo. Las ramas se insertan en el circuito principal. En una rama se pueden insertar varios contactos, creando así una conexión en paralelo de conexiones en serie. Esto permite programar esquemas de contactos complejos.

La figura siguiente muestra un ejemplo de la utilización de ramas:



El MOTOR tiene el estado lógico 1 si se cumple una de las condiciones siguientes:

- S2 o S4 tienen el estado lógico 1.
- S5 tiene el estado lógico 0.

Consulte también

Reglas para ramas en KOP (Página 2705)

Insertar ramas en segmentos KOP (Página 2706)

Cerrar ramas en segmentos KOP (Página 2706)

Borrar ramas en segmentos KOP (Página 2707)

Reglas para ramas en KOP

Reglas

Para las ramas paralelas rigen las reglas siguientes:

- Sólo se puede insertar una rama paralela si ya existe un elemento KOP en la rama principal.
- Las ramas paralelas se abren hacia abajo o se conectan directamente a la barra de alimentación y se cierran hacia arriba.
- Las ramas paralelas se abren detrás del elemento KOP seleccionado.

- Las ramas paralelas se cierran detrás del elemento KOP seleccionado.
- Para borrar una rama paralela deben borrarse todos sus elementos KOP. Al borrar el último elemento KOP de la rama, se borrará también el resto de la misma.

Consulte también

Principios básicos de las ramas en KOP (Página 2705)

Insertar ramas en segmentos KOP (Página 2706)

Borrar ramas en segmentos KOP (Página 2707)

Cerrar ramas en segmentos KOP (Página 2706)

Insertar ramas en segmentos KOP

En un segmento pueden crearse varias ramas.

Requisitos

- Existe un segmento.
- El segmento contiene elementos.

Procedimiento

Para insertar una rama nueva en un segmento, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "General > Abrir rama".
3. Arrastre el elemento hasta la posición deseada del segmento mediante Drag & Drop. Para insertar una rama directamente en la barra de alimentación, arrastre el elemento hasta la barra.

Consulte también

Principios básicos de las ramas en KOP (Página 2705)

Reglas para ramas en KOP (Página 2705)

Borrar ramas en segmentos KOP (Página 2707)

Cerrar ramas en segmentos KOP

Las ramas se deben volver a cerrar en los puntos adecuados. En caso necesario, las ramas se disponen de manera que no se crucen.

Requisitos

Existe una rama.

Procedimiento

Para cerrar una rama abierta, proceda del siguiente modo:

1. Seleccione la rama abierta.
2. Pulse y mantenga oprimido el botón izquierdo del ratón.
   Al mover el puntero del ratón aparece una línea discontinua.
3. Arrastre la línea discontinua hasta una posición apropiada del segmento. Las conexiones permitidas se identifican mediante líneas verdes.
4. Suelte el botón izquierdo del ratón.

Consulte también

Principios básicos de las ramas en KOP (Página 2705)

Reglas para ramas en KOP (Página 2705)

Borrar ramas en segmentos KOP

Requisitos

Existe una rama.

Procedimiento

Para borrar una rama, proceda del siguiente modo:

1. Seleccione la línea que conecta la rama con la rama principal.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Principios básicos de las ramas en KOP (Página 2705)

Reglas para ramas en KOP (Página 2705)

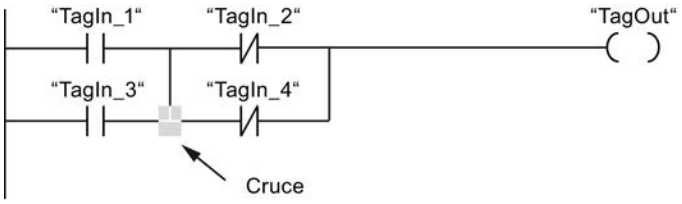
Insertar ramas en segmentos KOP (Página 2706)

Cruces en KOP

Principios básicos de los cruces en KOP

Definición

Un cruce es una posición de un segmento KOP en la que se cierra una rama al mismo tiempo que se abre otra rama.



"TagOut" tiene el estado lógico 1 si se cumplen las dos condiciones siguientes:

- "TagIn_1" o "TagIn_3" presentan el estado lógico 1
- "TagIn_2" o "TagIn_4" presentan el estado lógico 0

Insertar cruces

Los cruces se pueden insertar en un segmento KOP creando conexiones entre la rama principal y una rama adicional, o bien entre diferentes ramas.

Requisitos

Existe una rama.

Procedimiento

Para insertar un cruce en un segmento KOP, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "General > Abrir rama".
3. Arrastre el elemento y suéltelo detrás de la rama existente.
4. Inserte un elemento cualquiera en la rama abierta.
5. Detrás del elemento insertado, haga clic en la flecha de la rama abierta.
6. Mantenga pulsado el botón izquierdo del ratón y arrastre la línea de conexión discontinua hasta la rama principal.
7. Suelte el botón izquierdo del ratón.

Consulte también

Cambiar la posición de cruces (Página 2709)

Borrar cruces (Página 2709)

Insertar ramas en segmentos KOP (Página 2706)

Cambiar la posición de cruces

Requisitos

Existe un cruce.

Procedimiento

Para cambiar la posición de un cruce, proceda del siguiente modo:

1. Seleccione la línea de conexión que define los cruces en las respectivas ramas.
2. Elija el comando "Borrar" del menú contextual.
3. Abra la Task Card "Instrucciones".
4. En la paleta "Instrucciones básicas", navegue hasta "General > Abrir rama".
5. Arrastre el elemento mediante Drag & Drop hasta la posición del segmento en la que desea insertar el cruce nuevo.
6. Haga clic en la flecha de la rama abierta.
7. Mantenga pulsado el botón izquierdo del ratón y arrastre la línea de conexión discontinua hasta la rama secundaria en la que desea insertar el cruce nuevo.
8. Suelte el botón izquierdo del ratón.

Consulte también

Insertar cruces (Página 2708)

Borrar cruces (Página 2709)

Borrar cruces

Requisitos

Existe un cruce.

Procedimiento

Para borrar un cruce, proceda del siguiente modo:

1. Seleccione la línea de conexión que define los cruces en las respectivas ramas.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Insertar cruces (Página 2708)

Cambiar la posición de cruces (Página 2709)

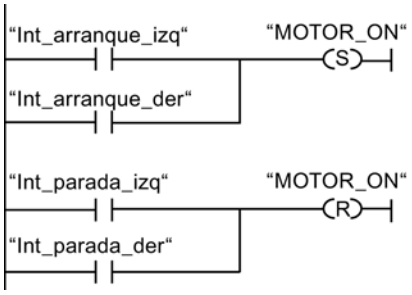
Circuitos en KOP

Principios básicos de los circuitos en KOP

Utilización de circuitos

El programa se mapea en uno o varios segmentos. Un segmento contiene en el margen izquierdo una barra de alimentación de la que pueden partir uno o varios circuitos. Las consultas de las señales binarias se disponen en los circuitos en forma de contactos. Si los elementos se disponen en serie en un circuito, se crea una conexión en serie. Si se disponen en ramas paralelas, se crea una conexión en paralelo. Un circuito se termina con una bobina o un cuadro en los que se escribe el resultado lógico.

La figura siguiente muestra un ejemplo de utilización de varios circuitos en un segmento:



Reglas

Observe las reglas siguientes si utiliza varios circuitos:

- No se permiten las conexiones entre los circuitos.
- Sólo se permite una instrucción de salto por segmento. Las reglas de posicionamiento de instrucciones de salto conservan su validez.

Procesamiento de circuitos

Los circuitos y segmentos se procesan de arriba abajo y de izquierda a derecha. Esto significa que primero se procesa la primera instrucción del primer circuito del primer segmento. Luego

se procesan todas las demás instrucciones de este circuito. A continuación se procesan todos los demás circuitos del primer segmento. Una vez procesados todos los circuitos, se procesa el siguiente segmento.

Diferencias entre las ramas y los circuitos

La diferencia entre las ramas y los circuitos es que estos últimos son ramas independientes que también pueden encontrarse en un segmento diferente. En cambio, las ramas permiten programar una conexión en paralelo.

Consulte también

Insertar un circuito (Página 2711)

Borrar un circuito (Página 2711)

Insertar un circuito

Requisitos

- Un bloque está abierto.
- Existe un segmento.

Procedimiento

Para insertar un circuito nuevo en un segmento, proceda del siguiente modo:

1. Inserte una bobina cualquiera en la barra de alimentación.
   Se inserta un circuito nuevo y la bobina se posiciona al final del circuito.
2. Inserte instrucciones adicionales en el circuito nuevo.

Consulte también

Principios básicos de los circuitos en KOP (Página 2710)

Borrar un circuito (Página 2711)

Borrar un circuito

Requisitos

Existe un circuito.

Procedimiento

Para borrar un circuito, proceda del siguiente modo:

1. Pulse y mantenga oprimido el botón izquierdo del ratón y trace un marco alrededor del circuito. Vigile que se seleccionen todas las instrucciones del circuito. Alternativamente, pulse y mantenga oprimida la tecla <Mayús> y seleccione la primera y última instrucción del circuito.
2. Haga clic con el botón derecho del ratón en una de las instrucciones del circuito.
3. Elija el comando "Borrar" del menú contextual.

Consulte también

Principios básicos de los circuitos en KOP (Página 2710)

Insertar un circuito (Página 2711)

Crear programas FUP

Principios básicos de FUP

Lenguaje de programación FUP

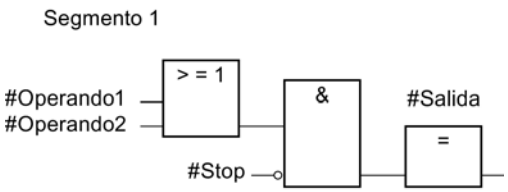
Resumen del lenguaje de programación FUP (Diagrama de funciones)

FUP es un lenguaje de programación gráfico. Su representación es similar a los diagramas de circuitos electrónicos.

El programa se mapea en uno o varios segmentos. Un segmento contiene uno o varios circuitos lógicos. Las consultas de las señales binarias se combinan lógicamente mediante cuadros. Para representar la lógica se utilizan los símbolos lógicos gráficos del álgebra booleana.

Ejemplo de segmentos en FUP

La figura siguiente muestra un segmento FUP con cuadros de operaciones lógicas Y y O, así como una asignación:



Consulte también

Trabajar con segmentos (Página 2715)

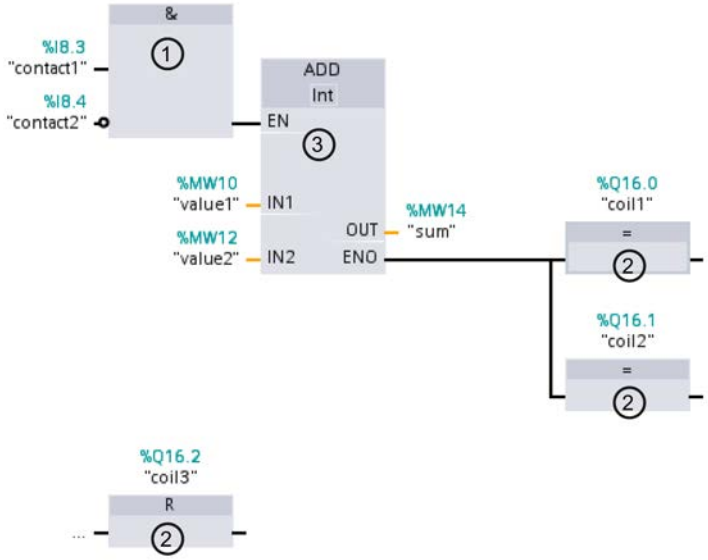
Vista general de los elementos FUP

Elementos FUP

Un programa FUP consta de distintos elementos conectados unos con otros a través del flujo de señales binarias. Es preciso asignar variables a la mayoría de los elementos del programa.

Un segmento FUP se programa de izquierda a derecha.

La figura siguiente muestra elementos de un segmento FUP a título de ejemplo:



1) Función binaria

2) Cuadro simple

3) Cuadro complejo

Funciones binarias

Las funciones binarias sirven para consultar los operandos binarios y combinar lógicamente sus estados lógicos. Las operaciones lógicas "Y", "O" y "O-exclusiva" son algunos ejemplos de funciones binarias.

Cuadros simples

Los cuadros simples sirven para controlar operandos binarios, evaluar flancos o ejecutar funciones de salto en el programa. Generalmente, los cuadros simples tienen sólo una entrada.

Cuadros complejos

Los cuadros complejos son elementos del programa con funciones complejas. El cuadro vacío es una excepción al respecto. Un cuadro vacío es un comodín en el que puede seleccionarse la instrucción deseada.

Los siguientes tipos de cuadros están disponibles en un programa FUP:

- Cuadros complejos sin mecanismo EN/ENO:
  El cuadro se ejecuta independientemente del estado lógico de sus entradas. No es posible consultar el estado de errores de ejecución.
- Cuadros complejos con mecanismo EN/ENO:
  Un cuadro se ejecuta únicamente si la entrada de habilitación "EN" tiene el estado lógico "1". Si el cuadro se ejecuta correctamente, la salida de habilitación "ENO" tendrá el estado lógico "1". Si ocurre un error durante el procesamiento, se desactiva la salida de habilitación "ENO".
  Si la entrada de habilitación EN no está interconectada, el cuadro se ejecuta siempre.

Las llamadas de bloques lógicos también se representan en el segmento como cuadros complejos con mecanismo EN/ENO.

Consulte también

Mecanismo EN/ENO (Página 2539)

Configuración de FUP

Resumen de la configuración de FUP

Resumen

La tabla siguiente muestra los ajustes que se pueden efectuar:

Grupo	Ajuste	Descripción
Fuente	Tamaño de fuente	Tamaño de fuente en el editor de programación
Vista	Formato	Compacto o ancho Cambia la distancia vertical entre los operandos y otros objetos (p. ej. operandos y contacto). El cambio efectuado se visualiza sólo tras volver a abrir el bloque.
	Con dirección absoluta	Indicación adicional de las direcciones absolutas

Grupo	Ajuste	Descripción
Campo de operando	Ancho máximo	Número máximo de caracteres que pueden introducirse horizontalmente en el campo del operando. Con este ajuste, el diseño de los segmentos se calcula nuevamente.
	Altura máxima	Número máximo de caracteres que pueden introducirse verticalmente en el campo del operando. Con este ajuste, el diseño de los segmentos se calcula nuevamente.

Consulte también

Modificar la configuración (Página 2715)

Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Modifique la configuración.

Resultado

El cambio se aplica sin necesidad de guardarlo explícitamente.

Consulte también

Resumen de la configuración de FUP (Página 2714)

Trabajar con segmentos

Utilizar segmentos

Función

El programa de usuario se crea en el bloque, dentro de segmentos. Para poder programar un bloque lógico, éste debe contener como mínimo un segmento. El programa de usuario también se puede dividir en varios segmentos para facilitar la claridad.

Consulte también

Insertar título del segmento (Página 2719)
Introducir el comentario del segmento (Página 2721)
Navegar en segmentos (Página 2721)

Insertar segmentos

Requisitos

Un bloque está abierto.

Procedimiento

Para insertar un segmento nuevo, proceda del siguiente modo:

1. Seleccione el segmento debajo del cual desea insertar un segmento nuevo.
2. Elija el comando "Insertar segmento" del menú contextual.

Resultado

Se inserta un nuevo segmento vacío en el bloque.

Consulte también

Insertar título del segmento (Página 2719)
Introducir el comentario del segmento (Página 2721)
Navegar en segmentos (Página 2721)

Seleccionar segmentos

Requisitos

Existe un segmento.

Seleccionar un segmento

Para seleccionar un segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento que desea seleccionar.

Seleccionar varios segmentos

Para seleccionar varios segmentos individuales, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Ctrl>.
2. Haga clic en todos los segmentos que desea seleccionar.

Para seleccionar varios segmentos consecutivos, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Mayús>.
2. Haga clic en el primer segmento que desea seleccionar.
3. Haga clic en el último segmento que desea seleccionar.
   Se seleccionan el primer y último segmento, así como todos los segmentos que se encuentran entre éstos.

Consulte también

Insertar segmentos (Página 2716)

Insertar título del segmento (Página 2719)

Introducir el comentario del segmento (Página 2721)

Navegar en segmentos (Página 2721)

Copiar e insertar segmentos

Los segmentos copiados pueden insertarse en el mismo bloque o en un bloque diferente. Los segmentos creados en KOP o FUP también pueden insertarse en los bloques de cada uno de los lenguajes de programación diferentes.

Requisitos

Existe un segmento.

Procedimiento

Para copiar e insertar un segmento, proceda del siguiente modo:

1. Seleccione el o los segmentos que desea copiar.
2. Elija el comando "Copiar" del menú contextual.
3. Seleccione el segmento debajo del cual desea insertar el segmento copiado.
4. Elija el comando "Pegar" del menú contextual.

Consulte también

Insertar segmentos (Página 2716)
Seleccionar segmentos (Página 2716)
Insertar título del segmento (Página 2719)
Introducir el comentario del segmento (Página 2721)
Navegar en segmentos (Página 2721)

Borrar segmentos

Requisitos

Existe un segmento.

Procedimiento

Para borrar un segmento, proceda del siguiente modo:

1. Seleccione el segmento que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Insertar segmentos (Página 2716)
Seleccionar segmentos (Página 2716)
Copiar e insertar segmentos (Página 2717)
Insertar título del segmento (Página 2719)
Introducir el comentario del segmento (Página 2721)
Navegar en segmentos (Página 2721)

Contraer y expandir segmentos

Requisitos

Existe un segmento.

Expandir y contraer un segmento

Para abrir un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha situada en la barra de título del segmento.

Para cerrar un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia abajo situada en la barra de título del segmento.

Expandir y contraer todos los segmentos

Para expandir o contraer todos los segmentos, proceda del siguiente modo:

1. Haga clic en el botón "Abrir todos los segmentos" o "Cerrar todos los segmentos" de la barra de herramientas.

Consulte también

Insertar segmentos (Página 2716)

Seleccionar segmentos (Página 2716)

Copiar e insertar segmentos (Página 2717)

Borrar segmentos (Página 2718)

Insertar título del segmento (Página 2719)

Introducir el comentario del segmento (Página 2721)

Navegar en segmentos (Página 2721)

Insertar título del segmento

El título de segmento es la cabecera de un segmento. La longitud del título del segmento está limitada a una línea. Puede introducir el título manualmente o definirlo de forma automática. Si elige el ajuste automático, puede aplicarlo a segmentos individuales o establecer en la configuración que el título del segmento debe definirse de forma automática.

Para la inserción automática del título del segmento se evalúa el comentario del operando de una de las siguientes instrucciones del segmento:

- Asignación
- Activar salida
- Desactivar salida

Para ello, se utiliza la instrucción que aparece primero en el segmento.

El título del segmento solo se inserta automáticamente si se cumplen las siguientes condiciones:

- El segmento todavía no tiene ningún título.
- El operando de la instrucción que se utiliza para el comentario tiene un comentario.

Nota

Al insertar el título del segmento de forma automática, tenga en cuenta las siguientes limitaciones:

- Si modifica el comentario del operando con posterioridad, el título del segmento no se actualiza.
- Si cambia el operando de la instrucción, el título del segmento no se actualiza.
- El título del segmento solo se define mediante las instrucciones de escritura citadas antes.
- Si el operando es del tipo de datos Array, se utiliza el comentario del Array y no los comentarios de los elementos Array.
- No se tienen en cuenta los comentarios de operandos no válidos.

Introducir el título del segmento manualmente

Para introducir un título de segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento.
2. Introduzca el título de segmento.

Definir el título del segmento automáticamente

Para establecer que el título del segmento debe definirse fundamentalmente de forma automática, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Active la casilla de verificación "Establecer título del segmento automáticamente" en el grupo "Configuración avanzada".
   Desde este momento, los títulos de los segmentos se definirán automáticamente si se cumplen las condiciones anteriores.

Para definir automáticamente un título de segmento concreto, proceda del siguiente modo:

1. Haga clic en la barra de título de un segmento con el botón derecho del ratón y seleccione "Segmento <Número del segmento>".
2. Elija el comando "Establecer título del segmento automáticamente" del menú contextual.
   El título del segmento seleccionado se define a partir del comentario del operando si se cumplen las condiciones anteriores.

Consulte también

Utilizar segmentos (Página 2715)

Insertar segmentos (Página 2716)

Seleccionar segmentos (Página 2716)

Copiar e insertar segmentos (Página 2717)

Borrar segmentos (Página 2718)

Contraer y expandir segmentos (Página 2718)

Introducir el comentario del segmento (Página 2721)

Navegar en segmentos (Página 2721)

Introducir el comentario del segmento

Los comentarios de segmentos sirven para documentar el contenido del programa en los distintos segmentos. Por ejemplo, es posible describir la función del segmento, o bien llamar la atención sobre peculiaridades.

Requisitos

Existe un segmento.

Procedimiento

Para introducir un comentario de segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha antepuesta al título del segmento.
2. Si el área de comentarios no resulta visible, haga clic en el botón "Activar/desactivar comentarios de segmento" de la barra de herramientas.
   Se abre el área de comentarios.
3. En el área de comentario, haga clic en "Comentario".
   El texto "Comentario" aparece seleccionado.
4. Introduzca el comentario del segmento.

Consulte también

Utilizar segmentos (Página 2715)

Insertar segmentos (Página 2716)

Seleccionar segmentos (Página 2716)

Copiar e insertar segmentos (Página 2717)

Borrar segmentos (Página 2718)

Contraer y expandir segmentos (Página 2718)

Insertar título del segmento (Página 2719)

Navegar en segmentos (Página 2721)

Navegar en segmentos

Existe la posibilidad de navegar en un bloque directamente hasta una posición determinada.

Procedimiento

Para navegar en un bloque hasta una posición determinada, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la sección de lógica de la ventana de programación.
2. Elija el comando "Ir a > Segmento/línea" del menú contextual.
   Se abre el cuadro diálogo "Ir a".
3. Introduzca el segmento hasta el que desea navegar.
4. Introduzca el número de la fila del segmento hasta el que desea navegar.
5. Confirme la entrada realizada con "Aceptar".

Resultado

Si es posible, se muestra la fila correspondiente. En caso de que el segmento o la fila deseados no existan, se visualiza el último segmento existente, o bien, la última fila existente en el segmento deseado.

Consulte también

Utilizar segmentos (Página 2715)

Insertar segmentos (Página 2716)

Seleccionar segmentos (Página 2716)

Copiar e insertar segmentos (Página 2717)

Borrar segmentos (Página 2718)

Contraer y expandir segmentos (Página 2718)

Insertar título del segmento (Página 2719)

Introducir el comentario del segmento (Página 2721)

Insertar elementos FUP

Reglas para utilizar elementos FUP

Reglas

Tenga en cuenta las reglas siguientes al insertar elementos FUP:

- Un segmento FUP puede constar de varios elementos. Todos los elementos de un circuito lógico deben estar conectados entre sí según IEC 61131-3.
- A los cuadros con operaciones lógicas binarias (p. ej. Y, O) se pueden añadir cuadros estándar (flipflops, contadores, temporizadores, operaciones aritméticas, etc.) como salida. Los cuadros de comparación son una excepción.
- Sólo las entradas booleanas de una instrucción pueden ir precedidas de una operación lógica.

- Sólo la última salida booleana de una instrucción se puede combinar mediante una operación lógica.
- Es posible cablear la entrada de habilitación EN y/o la salida de habilitación ENO de cuadros, pero no es absolutamente necesario.
- A las operaciones lógicas binarias no se pueden asignar constantes, p. ej. con TRUE o FALSE. Utilice variables del tipo de datos BOOL en su lugar.
- En un segmento sólo se puede insertar una instrucción de salto.
- Sólo se puede insertar una etiqueta por segmento.
- Las instrucciones con evaluación de flancos positivos o negativos no se pueden colocar en el extremo izquierdo del segmento, ya que requieren una operación lógica previa.

Reglas de posicionamiento para CPU S7-1200/1500

La siguiente tabla muestra las instrucciones que sólo pueden posicionarse al final del segmento:

Instrucción		Operación lógica previa necesaria
Nemónicos	Nombre	
SET_BF	Activar mapa de bits	No
RESET_BF	Desactivar mapa de bits	No
JMP	Saltar si RLO = 1	No
JMPN	Saltar si RLO = 0	SÍ
JMP_LIST	Definir lista de saltos	No
SWITCH	Distribuidor de saltos	No
RET	Retroceder	No

Reglas de posicionamiento para CPU S7-300/400

La siguiente tabla muestra las instrucciones que sólo pueden posicionarse al final del segmento:

Instrucción		Operación lógica previa necesaria
Nemónicos	Nombre	
S	Activar salida	Sí
R	Desactivar salida	Sí
SP	Arrancar temporizador como impulso	Sí
SE	Arrancar temporizador como impulso prolongado	Sí
SD	Arrancar temporizador como retardo a la conexión	Sí
SS	Arrancar temporizador como retardo a la conexión con memoria	Sí
SF	Arrancar temporizador como retardo a la desconexión	Sí
SC	Poner contador al valor inicial	Sí
CU	Contador ascendente	Sí
CD	Contador descendente	Sí

Instrucción		Operación lógica previa necesaria
Nemónicos	Nombre	
JMP	Saltar si RLO = 1	No
JMPN	Saltar si RLO = 0	Sí
RET	Retroceder	No
OPN	Abrir bloque de datos global	No
OPNI	Abrir bloque de datos de instancia	No
CALL	Llamar bloque	No
SAVE	Guardar RLO en bit BR	No
MCRA	Activar área MCR	No
MCRD	Desactivar área MCR	No
MCR<	Abrir áreas MCR	No
MCR>	Cerrar áreas MCR	No

Insertar elementos FUP mediante la Task Card "Instrucciones"

Requisitos

Existe un segmento.

Procedimiento

Para insertar un elemento FUP en un segmento mediante la Task Card "Instrucciones", proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Navegue hasta el elemento FUP que desea insertar.
3. Arrastre el elemento hasta la posición deseada del segmento mediante Drag & Drop.
   Si el elemento ejerce de bloque de función (FB) internamente en el sistema, se abre el cuadro de diálogo "Opciones de llamada". En este cuadro de diálogo se puede crear para el bloque de función un bloque de datos de instancia como instancia individual o como multiinstancia en el que se guarden los datos del elemento insertado. El nuevo bloque de datos de instancia se encuentra, después de crearse, en la carpeta "Recursos de programa" del árbol del proyecto, bajo "Bloques de programa > Bloques de sistema". Si ha seleccionado "Multiinstancia", la encontrará en la sección "Static" de la interfaz del bloque.

O bien:

1. Seleccione la posición del segmento en la que desea insertar el elemento.
2. Abra la Task Card "Instrucciones".
3. Haga doble clic en el elemento que desea insertar.
   Si el elemento ejerce de bloque de función (FB) internamente en el sistema, se abre el cuadro de diálogo "Opciones de llamada". En este cuadro de diálogo se puede crear para el bloque de función un bloque de datos de instancia como instancia individual o como multiinstancia en el que se guarden los datos del elemento insertado. El nuevo bloque de datos de instancia se encuentra, después de crearse, en la carpeta "Recursos de programa" del árbol del proyecto, bajo "Bloques de programa > Bloques de sistema". Si ha seleccionado "Multiinstancia", la encontrará en la sección "Static" de la interfaz del bloque.

Resultado

El elemento FUP seleccionado se inserta con comodines para los parámetros.

Consulte también

Reglas para utilizar elementos FUP (Página 2722)

Insertar elementos FUP mediante un cuadro vacío

Requisitos

Existe un segmento.

Procedimiento

Para insertar un elemento FUP en un segmento mediante un cuadro vacío, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "General > Cuadro vacío".
3. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
4. Sitúe el puntero del ratón sobre el triángulo de la esquina superior derecha del cuadro vacío.
   Se abre una lista desplegable.
5. Seleccione el elemento FUP deseado de la lista desplegable.
   Si el elemento ejerce de bloque de función (FB) internamente en el sistema, se abre el cuadro de diálogo "Opciones de llamada". En este cuadro de diálogo se puede crear para el bloque de función un bloque de datos de instancia como instancia individual o como multiinstancia en el que se guarden los datos del elemento insertado. El nuevo bloque de datos de instancia se encuentra, después de crearse, en la carpeta "Recursos de programa" del árbol del proyecto, bajo "Bloques de programa > Bloques de sistema". Si ha seleccionado "Multiinstancia", la encontrará en la sección "Static" de la interfaz del bloque.

Resultado

El cuadro vacío se transforma en el elemento FUP correspondiente. Para los parámetros se insertan comodines.

Seleccionar el tipo de datos de un elemento FUP

Seleccionar un tipo de datos

Introducción

Algunas instrucciones pueden ejecutarse con varios tipos de datos diferentes. Si en el programa se utiliza alguna de estas instrucciones, en la ubicación concreta del programa es preciso definir un tipo de datos permitido para la instrucción en cuestión. Para algunas instrucciones es necesario seleccionar los tipos de datos para las entradas y para las salidas por separado.

Nota

El sistema predetermina el tipo de datos permitido (BOOL) para las variables de la entrada de habilitación EN y la salida de habilitación ENO. Este tipo de datos no se puede modificar.

Los tipos de datos admisibles para una instrucción se indican en la lista desplegable correspondiente. Seleccionando una entrada de la lista desplegable se determina el tipo de datos de la instrucción. Si el tipo de datos de un operando difiere del tipo de datos de la instrucción y no puede convertirse de forma implícita, el operando se muestra en color rojo y aparece una ventana desplegable con el mensaje de error correspondiente.

Selección de un tipo de datos de instrucciones matemáticas

Algunas instrucciones matemáticas ofrecen la posibilidad de ajustar automáticamente el tipo de datos adecuado respecto a los tipos de datos de los operandos. En la lista desplegable para seleccionar el tipo de datos, dichas instrucciones tienen, además de los tipos de datos propiamente dichos, la entrada "Auto". Si se selecciona esta entrada y se activa el primer operando, el tipo de datos del operando se utiliza como tipo de datos para la instrucción. La entrada de la lista desplegable cambia a "Auto (<tipo de datos>)", p. ej. "Auto (Real)". Si se

activan otros operandos, el tipo de datos de la instrucción, ajustado automáticamente, se adapta según los criterios siguientes:

- Transfieren variables del mismo tipo de datos al resto de operandos:
  El tipo de datos de la instrucción no se modifica.
- Transfieren al resto de operandos variables cuyo tipo de datos es inferior al tipo de datos de la instrucción:
  El tipo de datos de la instrucción no se modifica. Para los operandos del tipo de datos inferior se realiza, si es necesario, una conversión implícita.
- Transfieren a otro operando una variable cuyo tipo de datos es superior al tipo de datos de la instrucción:
  El tipo de datos de la instrucción cambia al tipo de datos superior. Para los operandos que difieren del tipo de datos de la instrucción recién ajustado se realiza, si es necesario, una conversión implícita.

Cualquier modificación del tipo de datos de un operando puede provocar otros cambios en el tipo de datos de la instrucción. De este modo, es posible que otros operandos también se conviertan de forma implícita. Los operandos para los que se realiza una conversión implícita se marcan con un cuadrado gris.

Nota

Tenga en cuenta también la información relativa a la conversión de tipos de datos de su dispositivo y, particularmente en este caso, las indicaciones en materia de verificación CEI.

Consulte también: Conversión de tipos de datos (Página 2373)

Consulte también

Definir el tipo de datos de una instrucción (Página 2727)

Definir el tipo de datos de una instrucción

Introducción

Algunas instrucciones pueden ejecutarse con varios tipos de datos diferentes. Si se insertan este tipo de instrucciones en el programa, es preciso definir el tipo de datos adecuado para ellas en la ubicación concreta del programa.

Definir el tipo de datos mediante la lista desplegable

Para definir el tipo de datos de una instrucción mediante la lista desplegable, proceda del siguiente modo:

1. Arrastre la instrucción mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Haga clic en el triángulo situado en la esquina superior de la lista desplegable.
   Ésta se abre y se visualizan los tipos de datos admisibles para la instrucción.

3. Seleccione un tipo de datos de la lista desplegable.
   Aparece el tipo de datos seleccionado.
4. Si la instrucción dispone de dos listas desplegables, seleccione el tipo de datos de las entradas en la lista desplegable izquierda y el de las salidas en la derecha.

Definir el tipo de datos asignando variables

Para definir el tipo de datos de una instrucción asignando variables, proceda del siguiente modo:

1. Arrastre la instrucción mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Introduzca en una entrada o salida una variable válida cuyo tipo de datos deba adoptarse como tipo de datos de la instrucción.
   El tipo de datos de la variable aparece en la lista desplegable.
3. Si deben definirse tipos de datos tanto para las entradas como para las salidas de la instrucción, indique sendas variables válidas en una entrada y una salida. La variable indicada en la entrada determina el tipo de datos de las entradas y, la indicada en la salida, el tipo de datos de las salidas de la instrucción.

Definir automáticamente el tipo de datos de instrucciones matemáticas

Para definir automáticamente el tipo de datos de las instrucciones matemáticas, proceda del siguiente modo:

1. Arrastre la instrucción matemática mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Seleccione la entrada "Auto" en la lista desplegable.
3. Indique una variable válida en una entrada o salida.
   El tipo de datos de la variable se adoptará como tipo de datos de la instrucción. La entrada de la lista desplegable cambia a "Auto (<tipo de datos>)".

Consulte también: Seleccionar un tipo de datos (Página 2726)

Consulte también

Seleccionar un tipo de datos (Página 2726)

Emplear favoritos en FUP

Agregar elementos FUP a los Favoritos

Requisitos

- Hay un bloque abierto.
- La Task Card "Instrucciones" tiene configurado el modo multipaleta, o bien los Favoritos se muestran también en el editor.

Procedimiento

Para agregar instrucciones a los Favoritos, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Maximice la paleta "Instrucciones básicas".
3. En la paleta "Instrucciones básicas", navegue hasta la instrucción que desea agregar a los Favoritos.
4. Arrastre la instrucción hasta la paleta "Favoritos" o hasta el área de Favoritos del editor de programación mediante Drag & Drop.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Quitar elementos FUP de los Favoritos (Página 2730)

Sinopsis del editor de programación (Página 2619)

Insertar elementos FUP mediante Favoritos

Requisitos

- Hay un bloque abierto.
- Existen Favoritos.

Procedimiento

Para insertar una instrucción en un programa desde los Favoritos, proceda del siguiente modo:

1. Arrastre la instrucción deseada mediante Drag & Drop desde los Favoritos hasta la posición deseada.

O bien:

1. Marque el punto del programa en el que quiere insertar la instrucción.
2. En los Favoritos, haga clic en la instrucción que desea insertar.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Sinopsis del editor de programación (Página 2619)

Quitar elementos FUP de los Favoritos (Página 2730)

Quitar elementos FUP de los Favoritos

Requisitos

Hay un bloque lógico abierto.

Procedimiento

Para eliminar instrucciones de los Favoritos, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la instrucción que desea eliminar.
2. Elija el comando "Eliminar instrucción" del menú contextual.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Agregar elementos FUP a los Favoritos (Página 2729)

Insertar elementos FUP mediante Favoritos (Página 2729)

Sinopsis del editor de programación (Página 2619)

Insertar llamadas de bloque en FUP

Insertar llamadas de bloque mediante Drag & Drop

Las llamadas de funciones (FC) y bloques de función (FB) existentes pueden insertarse desde el árbol del proyecto mediante Drag & Drop. Al llamar bloques de función desde otros bloques de función, es posible hacerlo como instancia individual o multiinstancia. Un bloque de función llamado como instancia individual guarda sus datos en un bloque de datos propio. Un bloque de función llamado como multiinstancia guarda sus datos en el bloque de datos instancia del bloque de función invocante.

Requisitos

- Existe un segmento.
- El bloque que se debe llamar existe.

Insertar una llamada de una función (FC)

Para insertar una llamada de una función (FC) mediante Drag & Drop en un segmento, proceda del siguiente modo:

1. Arrastre la función desde el árbol del proyecto hasta el segmento deseado.

Insertar una llamada de un bloque de función (FB)

Para insertar una llamada de un bloque de función (FB), proceda del siguiente modo:

1. Arrastre el bloque de función desde el árbol del proyecto hasta el segmento deseado. Se abre el cuadro de diálogo "Opciones de llamada".
2. Indique en el cuadro de diálogo si quiere llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca seguidamente un nombre en el campo de entrada "Nombre" para el bloque de datos que se debe asignar al bloque de función.
   - Si hace clic en el botón "Multiinstancia", introduzca seguidamente en el campo de entrada "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
3. Confirme las entradas con "Aceptar".

Resultado

La función o el bloque de función se inserta con sus respectivos parámetros. Los valores de estos parámetros pueden asignarse más tarde.

Consulte también: Transferencia de parámetros en la llamada de un bloque (Página 2268)

Nota

Si al llamar un bloque de función especifica un bloque de datos instancia que no exista aún, éste se creará. Si ha llamado un bloque de función como multiinstancia, éste se introducirá en la interfaz como variable estática.

Consulte también

Actualizar llamadas de bloque en FUP (Página 2732)

Cambiar el tipo de instancia (Página 2733)

Instancias individuales (Página 2265)

Multiinstancias (Página 2266)

Actualizar llamadas de bloque en FUP

Si cambian los parámetros de interfaz de un bloque llamado, la llamada no podrá ejecutarse correctamente. Actualice las llamadas de bloque con objeto de evitar llamadas incoherentes.

Existen dos posibilidades de actualizar llamadas de bloque, a saber:

- Actualización explícita en el editor de programación.
  Se actualizan las llamadas de bloque del bloque abierto.
- Actualización implícita durante la compilación.
  Se actualizan todas las llamadas de bloque en el programa, así como los tipos de datos PLC utilizados.

Actualizar llamadas de bloque en el editor de programación

Para actualizar una llamada de bloque en un bloque, proceda del siguiente modo:

1. Abra el bloque en el editor de programación.
2. Haga clic en el botón "Actualizar llamadas de bloque incoherentes" de la barra de herramientas.

O bien:

1. Abra el bloque en el editor de programación.
2. Haga clic con el botón derecho del ratón en la instrucción que contiene la llamada de bloque.
3. Elija el comando "Actualizar" del menú contextual.
   Se abre el cuadro de diálogo "Sincronización de la interfaz". En él se visualizan las diferencias entre la interfaz utilizada y la interfaz modificada del bloque llamado.
4. Si desea actualizar la llamada de bloque, haga clic en "Aceptar". Para cancelar la actualización, haga clic en "Cancelar".

Actualizar llamadas de bloque durante la compilación

Para actualizar implícitamente todas las llamadas de bloque y usos de tipos de datos PLC durante la compilación, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Llamadas de bloque (Página 2263)

Insertar llamadas de bloque mediante Drag & Drop (Página 2731)

Cambiar el tipo de instancia (Página 2733)

Cambiar el tipo de instancia

Tipo de instancia

Los bloques de función se pueden llamar de dos maneras diferentes, a saber:

- Como instancia individual
- Como multiinstancia

Consulte también: Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

El tipo de instancia definido se puede modificar en todo momento.

Requisitos

El programa de usuario contiene una llamada de bloque.

Procedimiento

Para cambiar el tipo de instancia de un bloque de función, proceda del siguiente modo:

1. Abra el bloque lógico y seleccione la llamada de bloque.
2. Elija el comando "Cambiar instancia" del menú contextual.
   Se abre el cuadro de diálogo "Opciones de llamada".

3. Haga clic en el botón "Instancia individual" o "Multiinstancia".
   - Si selecciona el tipo de instancia "Instancia individual", introduzca un nombre para el bloque de datos que debe asignarse al bloque de función.
   - Si selecciona el tipo de instancia "Multiinstancia", introduzca en el campo de texto "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
4. Confirme las entradas con "Aceptar".

Nota

Las instancias individuales y multiinstancias anteriores no se borran automáticamente.

Consulte también

Insertar llamadas de bloque mediante Drag & Drop (Página 2731)

Actualizar llamadas de bloque en FUP (Página 2732)

Insertar instrucciones FUP complejas

Utilizar la instrucción "Calcular"

Requisitos

Existe un segmento.

Procedimiento

Para utilizar la instrucción "Calcular", proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "Funciones matemáticas > CALCULATE".
3. Arrastre el elemento hasta la posición deseada del segmento mediante Drag & Drop. La instrucción "Calcular" se inserta con una expresión comodín y signos de interrogación para el tipo de datos.
4. Introduzca el tipo de datos para el cálculo.
5. Introduzca los operandos para el cálculo.

Nota

El cálculo se lleva a cabo con las entradas de la instrucción "Calcular". Por ello, si desea utilizar constantes debe insertar también las entradas correspondientes para las constantes.

6. Haga clic en el botón "Editar instrucción "Calcular"" para reemplazar la expresión comodín por la expresión correcta.
   Se abre el cuadro de diálogo "Editar instrucción "Calcular"".
7. Introduzca la expresión deseada en el campo de texto "OUT:= ".

   Nota

   En el área "Ejemplo" dispone de un ejemplo de expresión válida y de las posibles instrucciones que se pueden utilizar.

   P. ej., para determinar un valor mediante el teorema de Pitágoras introduzca la expresión "OUT := SQRT (SQR (IN1) + SQR (IN2))".

8. Confirme las entradas con "Aceptar".

Consulte también

CALCULATE: Calcular (Página 3708)

Utilizar comentarios libres

Principios básicos de la utilización de comentarios libres en FUP

Introducción

Los comentarios libres permiten comentar el código fuente de los lenguajes de programación gráficos, de manera similar a los comentarios de línea de los lenguajes textuales.

Los comentarios libres pueden utilizarse para todos los cuadros no binarios.

Consulte también

Insertar comentarios libres (Página 2735)

Editar comentarios libres (Página 2736)

Borrar comentarios libres (Página 2737)

Insertar comentarios libres

Requisitos

Existe un segmento con instrucciones.

Procedimiento

Para añadir un comentario libre relativo a una instrucción, proceda del siguiente modo:

1. Si es preciso, pulse el botón "Mostrar/ocultar comentarios libres" de la barra de herramientas.
2. Haga clic con el botón derecho del ratón en la instrucción para la que desea insertar un comentario libre.
3. Elija el comando "Insertar comentario" del menú contextual.
   Se abre un cuadro de comentario con un comentario estándar. El cuadro de comentario está unido con la instrucción correspondiente por medio de una flecha.
4. Introduzca el comentario deseado en el cuadro de comentario.

Consulte también

Principios básicos de la utilización de comentarios libres en FUP (Página 2735)

Editar comentarios libres (Página 2736)

Borrar comentarios libres (Página 2737)

Editar comentarios libres

Introducción

Los comentarios libres pueden editarse como se indica a continuación:

- Modificar el texto del comentario
- Modificar la posición y el tamaño del cuadro de comentario
- Acoplar el comentario a un elemento diferente
- Mostrar y ocultar comentarios libres

Modificar el texto del comentario

Para modificar el texto de un comentario libre, proceda del siguiente modo:

1. Haga clic en el cuadro de comentario.
2. Introduzca el texto deseado.

Modificar la posición del cuadro de comentario

Para modificar la posición del cuadro de comentario, proceda del siguiente modo:

1. Haga clic en el cuadro de comentario y mantenga oprimido el botón del ratón.
2. Arrastre el cuadro de comentario hasta la posición deseada.

Modificar el tamaño del cuadro de comentario

Para modificar el tamaño del cuadro de comentario, proceda del siguiente modo:

1. Haga clic en el cuadro de comentario.
2. Arrastre el controlador de tamaño de la esquina inferior derecha del cuadro de comentario hasta alcanzar el tamaño deseado.

Acoplar el comentario a un elemento diferente

Para acoplar un comentario libre a un elemento diferente, proceda del siguiente modo:

1. Haga clic en la punta de la flecha que une el cuadro de comentario con la instrucción y mantenga oprimido el botón del ratón.
2. Arrastre la flecha hasta el elemento al que desea acoplar el comentario. Los puntos de inserción posibles se identifican con un cuadrado verde.
3. Suelte el botón del ratón.

Mostrar y ocultar comentarios libres

Para mostrar y ocultar los comentarios libres, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar comentarios libres" de la barra de herramientas.

Consulte también

Principios básicos de la utilización de comentarios libres en FUP (Página 2735)

Insertar comentarios libres (Página 2735)

Borrar comentarios libres (Página 2737)

Borrar comentarios libres

Procedimiento

Para borrar un comentario libre, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el comentario libre que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Principios básicos de la utilización de comentarios libres en FUP (Página 2735)

Insertar comentarios libres (Página 2735)

Editar comentarios libres (Página 2736)

Editar elementos FUP

Seleccionar elementos FUP

Se pueden seleccionar varios elementos individuales o todos los elementos de un segmento.

Requisitos

Existen elementos FUP.

Seleccionar varios elementos FUP individuales

Para seleccionar varios elementos FUP individuales, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Ctrl>.
2. Haga clic en los elementos FUP que desea seleccionar.
3. Suelte la tecla <Ctrl>.

Seleccionar todos los elementos FUP de un segmento

Para seleccionar todos los elementos FUP de un segmento, proceda del siguiente modo:

1. Haga clic en el segmento cuyos elementos desea seleccionar.
2. Elija el comando "Seleccionar todo" del menú "Edición" o utilice la combinación de teclas <Ctrl+E>.

Consulte también

Copiar elementos FUP (Página 2738)
Cortar elementos FUP (Página 2739)
Insertar elementos FUP desde el portapapeles (Página 2740)
Reemplazar elementos FUP (Página 2740)
Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)
Quitar entradas y salidas de una instrucción (Página 2742)
Activar y desactivar el mecanismo EN/ENO (Página 2743)
Borrar elementos FUP (Página 2744)

Copiar elementos FUP

Requisitos

Existe un elemento FUP.

Procedimiento

Para copiar un elemento FUP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el elemento FUP que desea copiar.
2. Elija el comando "Copiar" del menú contextual.

Resultado

El elemento FUP se copia y se guarda en el portapapeles.

Consulte también

Seleccionar elementos FUP (Página 2738)
Cortar elementos FUP (Página 2739)
Insertar elementos FUP desde el portapapeles (Página 2740)
Reemplazar elementos FUP (Página 2740)
Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)
Quitar entradas y salidas de una instrucción (Página 2742)
Activar y desactivar el mecanismo EN/ENO (Página 2743)
Borrar elementos FUP (Página 2744)

Cortar elementos FUP

Requisitos

Existe un elemento FUP.

Cortar

Para cortar un elemento FUP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el elemento FUP que desea cortar.
2. Seleccione el comando "Cortar" del menú contextual.

Resultado

El elemento FUP se corta y se guarda en el portapapeles.

Consulte también

Seleccionar elementos FUP (Página 2738)
Copiar elementos FUP (Página 2738)
Insertar elementos FUP desde el portapapeles (Página 2740)

Reemplazar elementos FUP (Página 2740)
Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)
Quitar entradas y salidas de una instrucción (Página 2742)
Activar y desactivar el mecanismo EN/ENO (Página 2743)
Borrar elementos FUP (Página 2744)

Insertar elementos FUP desde el portapapeles

Requisitos

Existe un elemento FUP.

Procedimiento

Para insertar un elemento FUP desde el portapapeles, proceda del siguiente modo:

1. Copie o corte un elemento FUP.
2. Haga clic con el botón derecho del ratón en la posición del segmento en la que desea insertar el elemento.
3. Elija el comando "Pegar" del menú contextual.

Consulte también

Seleccionar elementos FUP (Página 2738)
Copiar elementos FUP (Página 2738)
Cortar elementos FUP (Página 2739)
Reemplazar elementos FUP (Página 2740)
Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)
Quitar entradas y salidas de una instrucción (Página 2742)
Activar y desactivar el mecanismo EN/ENO (Página 2743)
Borrar elementos FUP (Página 2744)

Reemplazar elementos FUP

Los elementos FUP se pueden reemplazar muy fácilmente por otros del mismo tipo. La ventaja es que los parámetros se conservan, por lo que no es necesario introducirlos de nuevo. Por ejemplo, es posible sustituir una operación lógica O por una Y, un flipflop de desactivación/activación por uno de activación/desactivación, así como funciones de comparación o instrucciones de salto.

Requisitos

Existe un segmento con un elemento FUP como mínimo.

Procedimiento

Para reemplazar un elemento FUP por otro, proceda del siguiente modo:

1. Seleccione el elemento FUP que desea reemplazar.
   Si existen elementos compatibles con el elemento FUP seleccionado, aparecerá un triángulo en la esquina superior derecha del elemento.
2. Sitúe el puntero del ratón sobre el triángulo del elemento FUP.
   Se abre una lista desplegable.
3. Seleccione el elemento FUP de la lista desplegable por el que desea reemplazar el elemento FUP actual.

Consulte también

Seleccionar elementos FUP (Página 2738)

Copiar elementos FUP (Página 2738)

Cortar elementos FUP (Página 2739)

Insertar elementos FUP desde el portapapeles (Página 2740)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Activar y desactivar el mecanismo EN/ENO (Página 2743)

Borrar elementos FUP (Página 2744)

Insertar entradas y salidas adicionales en los elementos FUP

Introducción

Algunos elementos FUP que realizan operaciones aritméticas o binarias se pueden ampliar con entradas adicionales. Estos elementos son p. ej. las instrucciones "Sumar" (ADD) y "Multiplicar" (MUL), así como las operaciones lógicas Y u O. Los cuadros de las instrucciones "Copiar valor" (MOVE) y "Desmultiplexar" (DEMUX) se pueden ampliar mediante salidas adicionales.

El nombre de las entradas y salidas nuevas está formado por el tipo del elemento insertado y un número correlativo. Así pues, el nombre de una entrada nueva podría ser , p. ej., "IN2", y el nombre de una salida nueva, p. ej., "OUT2".

Requisitos

Existe un elemento FUP que permite insertar entradas o salidas adicionales.

Insertar una entrada adicional

Para insertar una entrada adicional en el cuadro de un elemento FUP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una entrada existente del elemento FUP.
2. Elija el comando "Insertar entrada" del menú contextual.
   El cuadro del elemento FUP se complementa con una entrada adicional.

O bien:

1. Haga clic en el asterisco amarillo junto a la última entrada del cuadro de instrucción.
   El cuadro del elemento FUP se complementa con una entrada adicional.

Insertar una salida adicional

Para insertar una salida adicional en el cuadro de un elemento FUP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una salida existente del elemento FUP.
2. Elija el comando "Insertar salida" del menú contextual.
   El cuadro del elemento FUP se complementa con una salida adicional.

O bien:

1. Haga clic en el asterisco amarillo junto a la última salida del cuadro de instrucciones.
   El cuadro del elemento FUP se complementa con una salida adicional.

Consulte también

Insertar elementos FUP (Página 2722)

Seleccionar elementos FUP (Página 2738)

Copiar elementos FUP (Página 2738)

Cortar elementos FUP (Página 2739)

Insertar elementos FUP desde el portapapeles (Página 2740)

Reemplazar elementos FUP (Página 2740)

Quitar entradas y salidas de una instrucción (Página 2742)

Activar y desactivar el mecanismo EN/ENO (Página 2743)

Borrar elementos FUP (Página 2744)

Quitar entradas y salidas de una instrucción

Introducción

Las entradas y salidas que se han insertado adicionalmente para una instrucción pueden quitarse nuevamente.

Requisitos

Existe un elemento FUP en el que se han insertado entradas o salidas adicionales.

Quitar una entrada

Para quitar una entrada, proceda del siguiente modo:

1. Seleccione la entrada que desea quitar.
2. Elija el comando "Borrar" del menú contextual.
   La entrada del elemento FUP se quita.

Quitar una salida

Para quitar una salida, proceda del siguiente modo:

1. Seleccione la salida que desea quitar.
2. Elija el comando "Borrar" del menú contextual.
   La salida del elemento FUP se quita.

Consulte también

Insertar elementos FUP (Página 2722)

Seleccionar elementos FUP (Página 2738)

Copiar elementos FUP (Página 2738)

Cortar elementos FUP (Página 2739)

Insertar elementos FUP desde el portapapeles (Página 2740)

Reemplazar elementos FUP (Página 2740)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Activar y desactivar el mecanismo EN/ENO (Página 2743)

Borrar elementos FUP (Página 2744)

Activar y desactivar el mecanismo EN/ENO

En KOP y FUP, determinadas instrucciones poseen una salida de habilitación ENO (enable output) y utilizan por consiguiente el mecanismo EN/ENO. Dicho mecanismo permite consultar errores en tiempo de ejecución en instrucciones y reaccionar a ellos. Para aumentar el rendimiento de la CPU, el mecanismo EN/ENO está desactivado por defecto. Por consiguiente, en un principio no existe la posibilidad de reaccionar a errores en tiempo de ejecución de la instrucción a través del valor ENO. Sin embargo, el mecanismo EN/ENO se puede activar en cualquier momento si fuera necesario.

El mecanismo EN/ENO se puede activar individualmente para cada instrucción con el fin de generar el ENO. Si se activa el mecanismo EN/ENO para una instrucción, el resto de instrucciones que se arrastren al programa posteriormente se insertarán con el mecanismo EN/ENO también activado. Si no desea utilizar la evaluación de ENO para una instrucción, el

mecanismo EN/ENO puede volver a desactivarse en cualquier momento. Cualquier otra instrucción que se arrastre al programa posteriormente se insertará sin el mecanismo EN/ENO.

Consulte también: Principios básicos del mecanismo EN/ENO (Página 2539)

Activar el mecanismo EN/ENO

Para activar el mecanismo EN/ENO de una instrucción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el programa sobre la instrucción para la que desea activar el mecanismo EN/ENO.
2. Elija el comando "Generar ENO" del menú contextual.
   Para esa instrucción se vuelve a generar el valor ENO. Cualquier otra instrucción añadida se insertará con salida de habilitación.

Desactivar el mecanismo EN/ENO

Para desactivar el mecanismo EN/ENO de una instrucción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el programa sobre la instrucción para la que desea desactivar el mecanismo EN/ENO.
2. Elija el comando "No generar ENO" del menú contextual.
   Para esa instrucción ya no se generará más el valor ENO. Cualquier otra instrucción añadida se insertará sin salida de habilitación.

Consulte también

Seleccionar elementos FUP (Página 2738)

Copiar elementos FUP (Página 2738)

Cortar elementos FUP (Página 2739)

Insertar elementos FUP desde el portapapeles (Página 2740)

Reemplazar elementos FUP (Página 2740)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Borrar elementos FUP (Página 2744)

Borrar elementos FUP

Requisitos

Existe un elemento FUP.

Procedimiento

Para borrar un elemento FUP, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el elemento FUP que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Insertar elementos FUP (Página 2722)
Seleccionar elementos FUP (Página 2738)
Copiar elementos FUP (Página 2738)
Cortar elementos FUP (Página 2739)
Insertar elementos FUP desde el portapapeles (Página 2740)
Reemplazar elementos FUP (Página 2740)
Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)
Quitar entradas y salidas de una instrucción (Página 2742)
Activar y desactivar el mecanismo EN/ENO (Página 2743)

Cablear operandos en instrucciones FUP

Cablear operandos

Cuando se inserta un elemento FUP, las cadenas de caracteres "<???>", "<??.?>" y "..." se insertan como comodines de los parámetros. Las cadenas de caracteres "<???>" y "<??.?>" representadas en color rojo identifican los parámetros que se deben cablear. La secuencia de caracteres "..." representada en color negro identifica los parámetros que se pueden cablear. "<??.?>" representa comodines booleanos.

Nota

Al situar el puntero del ratón sobre el comodín se muestran los tipos de datos posibles en un tooltip.

Requisitos

Existe un elemento FUP.

Procedimiento

Para cablear los parámetros de un elemento FUP, proceda del siguiente modo:

1. Haga clic en el comodín del parámetro.
   Aparece un campo de entrada.
2. Introduzca el parámetro correspondiente, p. ej. una variable PLC, una variable local o una constante.

   Nota

   Si introduce la dirección absoluta de un parámetro ya definido, ésta se sustituirá por el nombre simbólico del parámetro tan pronto como confirme la entrada. Si el parámetro no se ha definido aún, en la tabla de variables PLC se introducirá una nueva variable con esta dirección absoluta y el nombre estándar "Tag_1". Cuando confirme la entrada, la dirección absoluta se sustituirá por el nombre simbólico "Tag_1".

3. Confirme el parámetro pulsando la tecla Intro.
4. Si todavía no ha definido el parámetro, puede hacerlo mediante el menú contextual directamente en el editor de programación.
   Consulte también: "Declarar variables PLC en el editor de programación (Página 2577)".

O mediante Drag & Drop desde la tabla de variables PLC:

1. En el árbol del proyecto, seleccione la carpeta "Variables PLC" y abra la tabla de variables PLC.
2. Si está abierta la tabla de variables PLC, arrastre la variable deseada hasta la respectiva posición del programa. Si la tabla de variables PLC no está abierta, abra la vista detallada. Arrastre la variable deseada desde la vista detallada hasta la respectiva posición del programa.

O mediante Drag & Drop desde la interfaz del bloque:

1. Abra la interfaz del bloque.
2. Arrastre el operando deseado desde la interfaz del bloque hasta la respectiva posición del programa.

Resultado

- Si la sintaxis es correcta, el parámetro se representará en color negro.
- Si hay un error de sintaxis, no se saldrá del campo de entrada y aparecerá un mensaje de error correspondiente en la ficha "Información > Sintaxis" de la ventana de inspección.

Consulte también

Utilizar y direccionar operandos (Página 2280)

Utilizar la función automática de autocompletar (Página 2637)

Cablear parámetros ocultos

Introducción

Dependiendo de la CPU utilizada, en el programa de usuario se pueden usar instrucciones complejas, suministradas con el TIA Portal. Estas instrucciones pueden contener parámetros declarados como ocultos.

Si una instrucción contiene parámetros ocultos, aparece una flecha pequeña en el margen inferior del cuadro de la instrucción. Los parámetros ocultos se reconocen porque se muestran en fuente blanca.

Los parámetros ocultos se pueden mostrar y cablear en todo momento.

Mostrar u ocultar parámetros ocultos

Para mostrar u ocultar parámetros ocultos, proceda del siguiente modo:

1. Para mostrar los parámetros ocultos, haga clic en la flecha hacia abajo situada en el margen inferior del cuadro de la instrucción.
2. Para ocultar los parámetros ocultos, haga clic en la flecha hacia arriba situada en el margen inferior del cuadro de la instrucción.

Cablear parámetros ocultos

Para cablear parámetros ocultos, proceda del siguiente modo:

1. Cablee los parámetros ocultos de la misma manera que los parámetros visibles.
   El parámetro oculto se convierte en un parámetro visible.

Consulte también

Utilizar librerías (Página 427)

Mostrar u ocultar informaciones de variables

Introducción

Se puede mostrar la siguiente información sobre las variables empleadas en el editor de programación:

- Nombre de la variable
- Dirección de la variable
- Comentarios simples o jerárquicos para la documentación de las variables

La información se recoge de la interfaz del bloque en el caso de las variables locales y de DB, y de la tabla de variables PLC en el caso de las variables válidas para toda la CPU.

Es posible visualizar la información de las variables para todos los bloques o solo para bloques abiertos individuales. Si selecciona la información de variables para todos los bloques, se muestra la información de las variables de todos los bloques abiertos actuales y futuros.

La información de las variables se puede volver a ocultar en cualquier momento. Si se ha ocultado la información de variable de todos los bloques, se puede volver a ver para bloques abiertos individualmente.

Si selecciona la visualización de la información de variable con comentarios jerárquicos, en las variables estructuradas se muestran también los comentarios de los niveles de orden superior de la estructura. En este caso, la información se muestra entre paréntesis detrás del comentario de las variables y los comentarios de los distintos niveles se separan con un punto. Si no existe ningún comentario en un nivel de la estructura de una variable, se suprime de la visualización, lo que se puede detectar porque aparecen dos puntos consecutivos.

Mostrar u ocultar información de variables para todos los bloques

Para mostrar u ocultar la información de variables para todos los bloques, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Si desea visualizar la información de variable, seleccione en la lista desplegable "Información de la variable" bien la opción "Mostrar", bien la opción "Información de la variable con comentarios jerárquicos", en función de si desea visualizar comentarios simples o jerárquicos.
4. Si desea ocultar la información de variable, seleccione la opción "Ocultar" en la lista desplegable "Información de variable".
   La información de las variables se muestra u oculta para todos los bloques abiertos. Si abre más bloques, la información de las variables se mostrará u ocultará dependiendo del ajuste seleccionado.

Mostrar u ocultar información de variables para un bloque abierto

Para mostrar u ocultar la información de variables para un bloque abierto, proceda del siguiente modo:

1. Si desea visualizar la información de variable, seleccione en la lista desplegable "Muestra información sobre la variable." bien la opción "Mostrar información de la variable", bien la opción "Información de la variable con comentarios jerárquicos", en función de si desea visualizar comentarios simples o jerárquicos.
2. Si desea ocultar la información de variable, seleccione la opción "Ocultar información de la variable" en la lista desplegable "Muestra información sobre la variable".
   Se muestra u oculta la información sobre las variables.

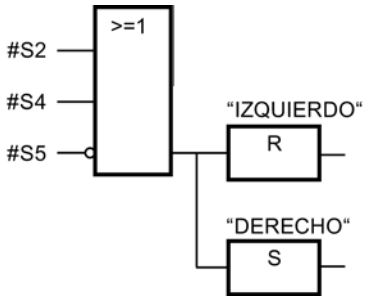
Ramas en FUP

Principios básicos de las ramas en FUP

Definición

El lenguaje de programación Diagrama de funciones (FUP) permite programar ramas paralelas. Para ello se utilizan ramas que se insertan entre los cuadros. En la rama pueden insertarse cuadros adicionales para programar diagramas de funciones complejos.

La figura siguiente muestra un ejemplo de la utilización de ramas:



Consulte también

Reglas para ramas en FUP (Página 2749)

Insertar ramas en segmentos FUP (Página 2750)

Borrar ramas en segmentos FUP (Página 2750)

Reglas para ramas en FUP

Reglas

Para la utilización de ramas en FUP rigen las siguientes reglas:

- Las ramas se abren hacia abajo.
- Las ramas sólo pueden insertarse entre dos elementos FUP.
- Para borrar una rama deben borrarse todos los elementos FUP, incluyendo la rama en sí.
- Si se borra la conexión entre dos ramas, los elementos FUP de la rama rota quedan situados libremente en el segmento.

Consulte también

Principios básicos de las ramas en FUP (Página 2749)

Insertar ramas en segmentos FUP (Página 2750)

Borrar ramas en segmentos FUP (Página 2750)

Insertar ramas en segmentos FUP

Requisitos

Existe un segmento.

Procedimiento

Para insertar una rama nueva en un segmento, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "General > Cerrar rama".
3. Arrastre el elemento mediante Drag & Drop y suéltelo en la posición deseada sobre una línea de conexión entre dos cuadros.

Consulte también

Reglas para ramas en FUP (Página 2749)

Principios básicos de las ramas en FUP (Página 2749)

Borrar ramas en segmentos FUP (Página 2750)

Borrar ramas en segmentos FUP

Requisitos

Existe una rama.

Procedimiento

Para borrar una rama, proceda del siguiente modo:

1. Seleccione la línea que conecta la rama con la rama principal.
2. Elija el comando "Borrar" del menú contextual.

Resultado

La rama se borra. Los cuadros que estaban conectados a la rama borrada se disponen libremente en el segmento.

Consulte también

Reglas para ramas en FUP (Página 2749)

Principios básicos de las ramas en FUP (Página 2749)

Insertar ramas en segmentos FUP (Página 2750)

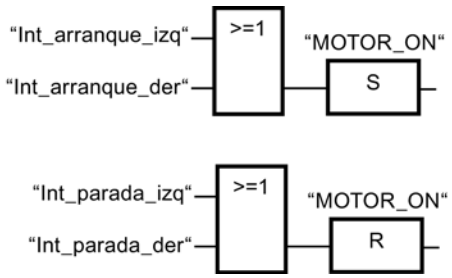
Circuitos lógicos en FUP

Principios básicos de los circuitos lógicos en FUP

Utilización de circuitos lógicos

El programa de usuario se representa en uno o varios segmentos. En los segmentos pueden encontrarse uno o varios circuitos lógicos, en los que las señales binarias están dispuestas en forma de cuadros.

La figura siguiente muestra un ejemplo de la utilización de varios circuitos lógicos en un segmento:



Reglas

Observe las reglas siguientes al utilizar circuitos lógicos:

- No se permiten conexiones entre los circuitos lógicos.
- Sólo se permite una instrucción de salto por segmento. Las reglas de posicionamiento de instrucciones de salto conservan su validez.

Procesamiento de circuitos lógicos

Los circuitos lógicos y segmentos se procesan de arriba abajo y de izquierda a derecha. Esto significa que primero se procesa la primera instrucción del primer circuito lógico del primer segmento. Luego se procesan todas las demás instrucciones de dicho circuito. A continuación se procesan todos los demás circuitos lógicos del primer segmento. Una vez procesados todos los circuitos lógicos, se procesa el siguiente segmento.

Si se utilizan saltos, se rodea el procesamiento regular de los circuitos lógicos y la instrucción se ejecuta en la meta de salto.

Diferencias entre las ramas y los circuitos lógicos

La diferencia entre las ramas y los circuitos lógicos es que éstos son ramas independientes que también pueden encontrarse en un segmento diferente. En cambio, las ramas permiten programar una conexión en paralelo y tienen una operación lógica precedente común.

Consulte también

Insertar un circuito lógico (Página 2752)

Borrar un circuito lógico (Página 2752)

Insertar un circuito lógico

Requisitos

- Un bloque está abierto.
- Existe un segmento.

Procedimiento

Para insertar un circuito lógico nuevo en un segmento, proceda del siguiente modo:

1. Inserte una instrucción cualquiera en un segmento, de manera que no tenga conexión alguna con las instrucciones existentes.
   Se inserta un circuito lógico nuevo.
2. Inserte una asignación al final del circuito lógico nuevo.
3. Inserte instrucciones adicionales en el circuito lógico nuevo.

Consulte también

Principios básicos de los circuitos lógicos en FUP (Página 2751)

Borrar un circuito lógico (Página 2752)

Borrar un circuito lógico

Requisitos

Existe un circuito lógico.

Procedimiento

Para borrar un circuito lógico, proceda del siguiente modo:

1. Pulse y mantenga oprimido el botón izquierdo del ratón y trace un marco alrededor del circuito lógico. Vigile que se seleccionen todas las instrucciones del circuito lógico. Alternativamente, pulse y mantenga oprimida la tecla <Mayús> y seleccione la primera y última instrucción del circuito lógico.
2. Haga clic con el botón derecho del ratón en una de las instrucciones del circuito lógico.
3. Elija el comando "Borrar" del menú contextual.

Consulte también

Principios básicos de los circuitos lógicos en FUP (Página 2751)

Insertar un circuito lógico (Página 2752)

Crear programas AWL

Bases de AWL

Lenguaje de programación AWL

Vista general del lenguaje de programación AWL (lista de instrucciones)

AWL es un lenguaje de programación basado en un texto con el que se podrán programar bloques lde código.

El programa AWL se divide en segmentos. Cada segmento puede tener una o varias líneas. La numeración de las líneas comienza en cada segmento con 1 y continuará de manera ascendente con cada nueva línea. En las líneas del segmento se programarán las instrucciones individuales de AWL, pudiéndose indicar sólo una instrucción AWL por línea. Cada instrucción representa una prescripción de trabajo para la CPU. La CPU ejecuta las instrucciones de arriba abajo.

El siguiente ejemplo muestra la programación de AWL de un segmento:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual
S "Tag_Output"	// Poner el operando a "1" si el RLO es "1"

Intercambio de datos en AWL

Intercambio de datos S7-300/400

El intercambio de información entre las diferentes áreas de memoria se efectúa en AWL mediante acumuladores para las CPU de la serie S7-300/400. Los acumuladores son registros especiales en el procesador que sirven de memoria intermedia. Todas las CPU S7-300/400 tienen dos memorias intermedias que son el acumulador 1 (ACU 1) y el acumulador 2 (ACU 2). Las CPU S7-400 y la CPU S7-318 tienen adicionalmente dos memorias intermedias más que son los acumuladores 3 (ACU 3) y 4 (ACU 4).

Intercambio de datos S7-1500

Las CPU de la serie S7-1500 ya no tienen registros en el procesador, pero pueden emular los registros de direcciones, ACU 1, el ACU 2, la palabra de estado y los registros de bloques de datos DB y DI. Los datos se intercambian, principalmente, a través de la interfaz del bloque, bloques de datos globales o variables PLC. Si a pesar de todo se utilizan los acumuladores para el intercambio de datos, la ejecución del programa se ralentiza.

Al realizar la programación en AWL para S7-1500, deberán tenerse en cuenta las siguientes indicaciones generales en lo que respecta a los registros:

- Los contenidos de los registros, de los acumuladores y de la palabra de estado solo están disponibles en segmentos AWL. Si un segmento KOP o FUP sigue a un segmento AWL, desde el segmento KOP o FUP no se podrá acceder a los contenidos de registro que antes se habían colocado en AWL. No obstante, en un segmento AWL posterior los contenidos de registro vuelven a estar disponibles.
  El bit RLO es una excepción: al realizar un cambio de idioma se pone en "indefinido" y en los siguientes segmentos ya no estará disponible.
- Los valores de los registros, acumuladores y de la palabra de estado no se transfieren a los bloques llamados. La única excepción son las instrucciones "CC" y "UC". Si utiliza "UC" o "CC", para transferir al bloque llamado parámetros a través de registros, de la palabra de estado o de acumuladores, active la opción "Alimentación de parámetros a través de registros" en las propiedades del bloque llamado. Tenga en cuenta que esta opción solo está disponible para bloques AWL con acceso estándar y que el bloque no puede tener parámetros formales. Si esta opción está activada, los contenidos de registro pueden transferirse entre bloques. También en este caso el bit RLO es una excepción: al salir del bloque se pone en "indefinido" y ya no estará disponible tras una llamada de bloque.
- Después de cada acceso a un bloque de datos indicando una dirección totalmente especificada (p. ej. %DB10.DBW10), el registro de bloques de datos DB se pone a "0". Un acceso posterior parcialmente especificado provoca un error durante la compilación.
- Para transferir una información de error al bloque invocante, se puede utilizar el bit BR. La información de error primero debe guardarse en el bloque llamado con la instrucción "SAVE" en el bit BR. A continuación se puede leer el bit BR en el bloque invocante.
- Si se direcciona simbólicamente un parámetro formal local desde la interfaz de bloque en S7-1500 (p. ej. con la instrucción L #myIn), siempre se accede al bloque de datos que se ha indicado como instancia en la llamada del bloque. Las instrucciones OPNDI, L AR2, +AR2, TDB y TAR cambian el contenido del registro DI o de direccionamiento, pero los registros ya no se evalúan en el direccionamiento de parámetros formales locales.

Cargar y transferir

Al intercambiar los datos, se indentificará la dirección del flujo de la información. Cuando se transfieren los datos desde un área de memoria al acumulador 1, se habla de cargar los datos. La transferencia de datos desde el acumulador 1 al área de memoria se denominará "Transferir". Para cargar y transferir en AWL se emplearán las instrucciones de carga y transferencia.

Es necesario cargar y transferir en especial a la hora de procesar los valores digitales. Por ej. cuando se suman dos valores, éstos se deberán cargar en la memoria intermedia y se ejecutará la correspondiente instrucción sumadora. Primero se cargará el primer valor acumulativo en el acumulador 1. Al cargar el segundo sumando se desplazará al acumulador 2 el primer valor que se encuentra en el acumulador 1. Después de ejecutar la instrucción

sumadora (por ej. I+) se guardará la suma en el acumulador 1. Será posible transferir la suma a cualquier operando gracias a la instrucción de transferencia.

Todos los acumuladores tienen una longitud de 32 bits y están organizados en bytes. La transferencia de datos entre las áreas de memoria y el acumulador 1 puede realizarse en bytes, palabras, o palabras dobles.

Intercambio de datos entre los acumuladores

Se podrán intercambiar datos entre los acumuladores individuales. Para ello se emplearán las instrucciones para acumuladores "TAK", "PUSH" y "POP". Estas permiten transferir datos entre los acumuladores disponibles e intercambiar bytes en el acumulador 1. Tenga en cuenta que en el S7-1500 ya no se dispone de las instrucciones para acumuladores "ENT" y "LEAVE".

Consulte también

Carga (Página 6251)

Transferir (Página 6260)

Acumulador (Página 6363)

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Alimentación de parámetros a través de registros

Alimentación de parámetros a través de registros en la llamada de bloques en AWL

En AWL pueden realizarse llamadas de bloques mediante las siguientes instrucciones:

- CALL: Llamar bloque
- UC: Llamada incondicional
- CC: Llamada condicional

En la llamada de bloque, se transfieren generalmente valores entre el bloque llamado y el bloque invocante. Las posibilidades del intercambio de datos difieren según la familia de CPU. Consulte también: Intercambio de datos en AWL (Página 2753)

Valores transferidos del bloque invocante al bloque llamado

La siguiente tabla muestra los registros, acumuladores y bits de estado disponibles en el bloque llamado directamente después de una llamada de bloque y antes de procesarse el primer segmento:

	S7-300/400		S7-1500	
	UC/CC	CALL	UC/CC con opción "Alimentación de parámetros a través de registros"	CALL
Bit de estado OS	0	0	0	0
Bit de estado OV	W	W	W	0
Bit de estado CC0	W	W	W	0
Bit de estado CC1	W	W	W	0
BR	W	W	W	1
RLO	W	W	W	U
Acu 1	W	W	W	U
Acu 2	W	W	W	U
Registro de direcciones 1	W	W	W	0
Registro de direcciones 2	W	Call FC: W Call FB: offset de instancia	W	Call FC: 0 Call FB: offset de instancia
Registro DB	W	W	W	U
Registro DI	W	Call FC: W Call FB: número DB de instancia	W	Call FC: U Call FB: número DB de instancia

Leyenda:
0 = registro se pone a "0" en el cambio de bloque.
1 = registro de pone a "1" en el cambio de bloque.
W = contenido del registro puede activarse en el bloque invocante y leerse en el bloque llamado.
U = contenido del registro no se transfiere al bloque llamado. El contenido es indefinido, el acceso provoca un error al compilar.
Offset de instancia = offset en DB de instancia
Número DB de instancia = número del DB de instancia

Valores devueltos del bloque llamado al bloque invocante

La siguiente tabla muestra los registros, acumuladores y bits de estado disponibles en el bloque invocante directamente después de procesarse la llamada de bloque:

	S7-300/400		S7-1500	
	UC/CC	CALL	UC/CC con opción "Alimentación de parámetros a través de registros"	CALL
Bit de estado OS	0	0	0	0
Bit de estado OV	W	W	W	-
Bit de estado CC0	W	W	W	-
Bit de estado CC1	W	W	W	-

	S7-300/400		S7-1500	
BR	W	W	W	W
RLO	W	W1	W	U
Acu 1	W	W1	W	-
Acu 2	W	W1	W	-
Registro de direcciones 1	W	W1	W	-
Registro de direcciones 2	W	Call FC: W Call FB: -	W	-
Registro DB	-	X	-	-
Registro DI	-	X	-	-

Leyenda:
0 = registro se pone a "0" en el cambio de bloque.
1 = registro se pone a "1" en el cambio de bloque.
W = contenido del registro puede activarse en el bloque llamado y leerse en el bloque invocante.
W1 = los parámetros de salida y de entrada/salida del bloque llamado pueden modificar el valor devuelto.
- = el contenido del registro no se devuelve al bloque invocante, la llamada de bloque no modifica el contenido.
U = el contenido del registro no se transfiere al bloque llamado. El contenido es indefinido, el acceso provoca un error al compilar.
X = contenido del registro no se devuelve al bloque invocante. Sin embargo, los parámetros del bloque llamado pueden modificar implícitamente el valor.

Consulte también

Transferencia de valores a través de registros o de la palabra de estado durante la llamada de bloque (Página 226)

Transferencia de valores a través de registros con las instrucciones CC y UC (Página 227)

Activar alimentación de parámetros a través de registros (Página 2757)

Activar alimentación de parámetros a través de registros

Requisitos

- La opción "Acceso optimizado al bloque" está desactivada.
- La interfaz del bloque no contiene parámetros.

Procedimiento

Para activar la alimentación de parámetros a través de registros, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del árbol del proyecto.
2. Haga clic con el botón derecho del ratón en el bloque para el que desea permitir la alimentación de parámetros a través de registros.
3. Elija el comando "Propiedades" del menú contextual.
4. Se abre el diálogo de propiedades del bloque.
5. Haga clic en "Atributos" en la navegación local.
6. Active la opción "Alimentación de parámetros a través de registros".
7. Confirme las entradas con "Aceptar".

Consulte también

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Intercambio de datos en AWL (Página 2753)

Configuración de AWL

Vista general de configuraciones de AWL

Sinopsis

La siguiente tabla muestra la configuración que usted puede realizar para AWL:

Grupo	Ajuste	Descripción
Vista	Ajuste de líneas con Intro	Añade un ajuste de líneas al pulsar la tecla Intro.

Consulte también

Modificar la configuración (Página 2759)

Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Modifique la configuración.

Resultado

El cambio se aplica sin necesidad de guardarlo explícitamente.

Consulte también

Vista general de configuraciones de AWL (Página 2758)

Trabajar con segmentos

Utilizar segmentos

Función

El programa de usuario se crea en el bloque, dentro de segmentos. Para poder programar un bloque lógico, éste debe contener como mínimo un segmento. El programa de usuario también se puede dividir en varios segmentos para facilitar la claridad.

Consulte también

Insertar segmentos (Página 2760)

Seleccionar segmentos (Página 2760)

Copiar e insertar segmentos (Página 2761)

Borrar segmentos (Página 2762)

Contraer y expandir segmentos (Página 2763)

Insertar título del segmento (Página 2763)

Introducir el comentario del segmento (Página 2765)

Navegar en segmentos (Página 2766)

Insertar segmentos

Requisitos

Un bloque está abierto.

Procedimiento

Para insertar un segmento nuevo, proceda del siguiente modo:

1. Seleccione el segmento debajo del cual desea insertar un segmento nuevo.
2. Elija el comando "Insertar segmento" del menú contextual.

Resultado

Se inserta un nuevo segmento vacío en el bloque.

Consulte también

Utilizar segmentos (Página 2759)
Seleccionar segmentos (Página 2760)
Copiar e insertar segmentos (Página 2761)
Borrar segmentos (Página 2762)
Contraer y expandir segmentos (Página 2763)
Insertar título del segmento (Página 2763)
Introducir el comentario del segmento (Página 2765)
Navegar en segmentos (Página 2766)

Seleccionar segmentos

Requisitos

Existe un segmento.

Seleccionar un segmento

Para seleccionar un segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento que desea seleccionar.

Seleccionar varios segmentos

Para seleccionar varios segmentos individuales, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Ctrl>.
2. Haga clic en todos los segmentos que desea seleccionar.

Para seleccionar varios segmentos consecutivos, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Mayús>.
2. Haga clic en el primer segmento que desea seleccionar.
3. Haga clic en el último segmento que desea seleccionar.
   Se seleccionan el primer y último segmento, así como todos los segmentos que se encuentran entre éstos.

Consulte también

Utilizar segmentos (Página 2759)

Insertar segmentos (Página 2760)

Copiar e insertar segmentos (Página 2761)

Borrar segmentos (Página 2762)

Contraer y expandir segmentos (Página 2763)

Insertar título del segmento (Página 2763)

Introducir el comentario del segmento (Página 2765)

Navegar en segmentos (Página 2766)

Copiar e insertar segmentos

Los segmentos copiados pueden insertarse en el mismo bloque o en un bloque diferente. Los segmentos creados en KOP o FUP también pueden insertarse en los bloques de cada uno de los lenguajes de programación diferentes.

Requisitos

Existe un segmento.

Procedimiento

Para copiar e insertar un segmento, proceda del siguiente modo:

1. Seleccione el o los segmentos que desea copiar.
2. Elija el comando "Copiar" del menú contextual.
3. Seleccione el segmento debajo del cual desea insertar el segmento copiado.
4. Elija el comando "Pegar" del menú contextual.

Consulte también

Utilizar segmentos (Página 2759)
Insertar segmentos (Página 2760)
Seleccionar segmentos (Página 2760)
Borrar segmentos (Página 2762)
Contraer y expandir segmentos (Página 2763)
Insertar título del segmento (Página 2763)
Introducir el comentario del segmento (Página 2765)
Navegar en segmentos (Página 2766)

Borrar segmentos

Requisitos

Existe un segmento.

Procedimiento

Para borrar un segmento, proceda del siguiente modo:

1. Seleccione el segmento que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Utilizar segmentos (Página 2759)
Insertar segmentos (Página 2760)
Seleccionar segmentos (Página 2760)
Copiar e insertar segmentos (Página 2761)
Contraer y expandir segmentos (Página 2763)
Insertar título del segmento (Página 2763)

Introducir el comentario del segmento (Página 2765)

Navegar en segmentos (Página 2766)

Contraer y expandir segmentos

Requisitos

Existe un segmento.

Expandir y contraer un segmento

Para abrir un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha situada en la barra de título del segmento.

Para cerrar un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia abajo situada en la barra de título del segmento.

Expandir y contraer todos los segmentos

Para expandir o contraer todos los segmentos, proceda del siguiente modo:

1. Haga clic en el botón "Abrir todos los segmentos" o "Cerrar todos los segmentos" de la barra de herramientas.

Consulte también

Utilizar segmentos (Página 2759)

Insertar segmentos (Página 2760)

Seleccionar segmentos (Página 2760)

Copiar e insertar segmentos (Página 2761)

Borrar segmentos (Página 2762)

Insertar título del segmento (Página 2763)

Introducir el comentario del segmento (Página 2765)

Navegar en segmentos (Página 2766)

Insertar título del segmento

El título de segmento es la cabecera de un segmento. La longitud del título del segmento está limitada a una línea. Puede introducir el título manualmente o definirlo de forma automática. Si elige el ajuste automático, puede aplicarlo a segmentos individuales o establecer en la configuración que el título del segmento debe definirse de forma automática.

Para la inserción automática del título del segmento se evalúa el comentario del operando de una de las siguientes instrucciones del segmento:

- Asignación
- Activar salida
- Desactivar salida

Para ello, se utiliza la instrucción que aparece primero en el segmento.

El título del segmento solo se inserta automáticamente si se cumplen las siguientes condiciones:

- El segmento todavía no tiene ningún título.
- El operando de la instrucción que se utiliza para el comentario tiene un comentario.

Nota

Al insertar el título del segmento de forma automática, tenga en cuenta las siguientes limitaciones:

- Si modifica el comentario del operando con posterioridad, el título del segmento no se actualiza.
- Si cambia el operando de la instrucción, el título del segmento no se actualiza.
- El título del segmento solo se define mediante las instrucciones de escritura citadas antes.
- Si el operando es del tipo de datos Array, se utiliza el comentario del Array y no los comentarios de los elementos Array.
- No se tienen en cuenta los comentarios de operandos no válidos.

Introducir el título del segmento manualmente

Para introducir un título de segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento.
2. Introduzca el título de segmento.

Definir el título del segmento automáticamente

Para establecer que el título del segmento debe definirse fundamentalmente de forma automática, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Active la casilla de verificación "Establecer título del segmento automáticamente" en el grupo "Configuración avanzada".
   Desde este momento, los títulos de los segmentos se definirán automáticamente si se cumplen las condiciones anteriores.

Para definir automáticamente un título de segmento concreto, proceda del siguiente modo:

1. Haga clic en la barra de título de un segmento con el botón derecho del ratón y seleccione "Segmento <Número del segmento>".
2. Elija el comando "Establecer título del segmento automáticamente" del menú contextual. El título del segmento seleccionado se define a partir del comentario del operando si se cumplen las condiciones anteriores.

Consulte también

Utilizar segmentos (Página 2759)

Insertar segmentos (Página 2760)

Seleccionar segmentos (Página 2760)

Copiar e insertar segmentos (Página 2761)

Borrar segmentos (Página 2762)

Contraer y expandir segmentos (Página 2763)

Introducir el comentario del segmento (Página 2765)

Navegar en segmentos (Página 2766)

Introducir el comentario del segmento

Los comentarios de segmentos sirven para documentar el contenido del programa en los distintos segmentos. Por ejemplo, es posible describir la función del segmento, o bien llamar la atención sobre peculiaridades.

Requisitos

Existe un segmento.

Procedimiento

Para introducir un comentario de segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha antepuesta al título del segmento.
2. Si el área de comentarios no resulta visible, haga clic en el botón "Activar/desactivar comentarios de segmento" de la barra de herramientas.
   Se abre el área de comentarios.
3. En el área de comentario, haga clic en "Comentario".
   El texto "Comentario" aparece seleccionado.
4. Introduzca el comentario del segmento.

Consulte también

Utilizar segmentos (Página 2759)

Insertar segmentos (Página 2760)

Seleccionar segmentos (Página 2760)

Copiar e insertar segmentos (Página 2761)

Borrar segmentos (Página 2762)

Contraer y expandir segmentos (Página 2763)

Insertar título del segmento (Página 2763)

Navegar en segmentos (Página 2766)

Navegar en segmentos

Existe la posibilidad de navegar en un bloque directamente hasta una posición determinada.

Procedimiento

Para navegar en un bloque hasta una posición determinada, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la sección de lógica de la ventana de programación.
2. Elija el comando "Ir a > Segmento/línea" del menú contextual.
   Se abre el cuadro diálogo "Ir a".
3. Introduzca el segmento hasta el que desea navegar.
4. Introduzca el número de la fila del segmento hasta el que desea navegar.
5. Confirme la entrada realizada con "Aceptar".

Resultado

Si es posible, se muestra la fila correspondiente. En caso de que el segmento o la fila deseados no existan, se visualiza el último segmento existente, o bien, la última fila existente en el segmento deseado.

Consulte también

Utilizar segmentos (Página 2759)

Insertar segmentos (Página 2760)

Seleccionar segmentos (Página 2760)

Copiar e insertar segmentos (Página 2761)

Borrar segmentos (Página 2762)

Contraer y expandir segmentos (Página 2763)

Insertar título del segmento (Página 2763)

Introducir el comentario del segmento (Página 2765)

Añadir instrucciones AWL

Reglas sobre AWL

Reglas

Con la entrada incremental de instrucciones AWL se deberán observan las siguientes reglas:

- La instrucción consiste en la entrada de una marca opcional de salto, de una operación, de un operando y de un comentario opcional. Ejemplo:
  `M001: U E1.0 //Comentario`
- Cada instrucción está en una línea.
- Se puede introducir un máximo de 800 segmentos por bloque.
- Al introducir operandos o direcciones absolutas no habrá ninguna diferencia entre mayúsculas y minúsculas.
- Si desea recibir ayuda para abrir el bloque, deberá programar el bloque abierto antes que el bloque que desea abrir.

Las instrucciones no admisibles se marcarán en rojo. La transferencia del contenido de un almacenador a una memoria intermedia y la ventana de inspección nos indican que debemos corregir errores.

Operaciones de salto

En operaciones de salto asegúrese de que la meta siempre es el inicio de una cadena lógica y de que se encuentra dentro de otro nivel de paréntesis (no aplicable para la CPU 318-2). La meta no puede estar dentro de una cadena lógica o de un nivel de paréntesis.

Consulte también

Introducir manualmente las instrucciones AWL (Página 2768)

Insertar instrucciones AWL empleando la Task Card "Instrucciones" (Página 2769)

Emplear favoritos en AWL (Página 2774)

Insertar llamadas de bloque en AWL (Página 2776)

Insertar comentarios (Página 2781)

Insertar operandos (Página 2771)

Introducir manualmente las instrucciones AWL

Al crear un nuevo segmento, ya sea de manera automática o manual, aparece un campo de entrada de dos líneas dentro del segmento. En este campo de entrada se introducirán las instrucciones AWL.

Nota

S7-1500: Las instrucciones de la Task Card "Instrucciones" que utilizan registros y acumuladores emulados no pueden utilizar variables con una longitud de 64 bits como operandos, p. ej. LInt. Si desea emplear variables de 64 bits, utilice las instrucciones de la librería global "Long Functions".

Requisitos

Existe un segmento.

Procedimiento

Para introducir las instrucciones AWL en un segmento, se deberá proceder de la siguiente manera:

1. Pulsar dentro de un segmento en la primera línea vacía del campo de entrada.
2. Introduzca la instrucción.
3. Deje un espacio.
4. Introduzca el operando.

   Nota

   Mediante Drag & Drop podrá insertar un operando ya definido, procedente de la tabla de variables del PLC o de la vista detallada.

5. Introduzca un comentario de línea. Este paso es opcional.
6. Confirme la instrucción con la tecla Intro.
7. Repita los pasos 2 a 6 hasta que haya insertado todas las instrucciones necesarias.

Resultado

Después de confirmar la instrucción con la tecla Intro, el editor programador llevará a cabo las siguientes acciones:

- Comprobación de sintaxis: Las entradas erróneas aparecerán en rojo y en cursiva después de la comprobación. Además aparece un mensaje de error detallado en la ficha "Sintaxis" de la ventana de información.
- Cambio de mayúsculas y minúsculas: Las instrucciones se cambiarán en letras mayúsculas.
- Si ha introducido direcciones absolutas en la programación simbólica, se cambiarán a nombres simbólicos. Si no hay ningún nombre simbólico, se insertará "Tag_< Número>".

- Llamada de bloque: Se insertarán y visualizarán los parámetros necesarios.
- En el campo de entrada se insertará otra línea.

Consulte también

Reglas sobre AWL (Página 2767)

Insertar instrucciones AWL empleando la Task Card "Instrucciones" (Página 2769)

Insertar operandos (Página 2771)

Emplear favoritos en AWL (Página 2774)

Insertar llamadas de bloque en AWL (Página 2776)

Insertar comentarios (Página 2781)

Insertar instrucciones AWL empleando la Task Card "Instrucciones"

Al crear un nuevo segmento, ya sea de manera automática o manual, aparece un campo de entrada de dos líneas dentro del segmento. En este campo de entrada se introducen las instrucciones AWL.

Nota

S7-1500: Las instrucciones de la Task Card "Instrucciones" que utilizan registros y acumuladores emulados no pueden utilizar variables con una longitud de 64 bits como operandos, p. ej. LInt. Si desea emplear variables de 64 bits, utilice las instrucciones de la librería global "Long Functions".

Requisitos

Existe un segmento.

Procedimiento

Para insertar en un segmento las instrucciones AWL empleando la Task Card "Instrucciones", deberá proceder de la siguiente manera:

1. Abra la Task Card "Instrucciones".
2. Para insertar la instrucción se deberá seleccionar uno de los siguientes pasos:
   - Navegue hasta al elemento AWL que desea insertar y arrástrelo mediante Drag & Drop al lugar deseado del segmento.
   - Marque el lugar del segmento en el que desea insertar el elemento y haga doble clic en el segmento a insertar.

3. Introduzca los operandos necesarios.

Nota

Es posible insertar un operando ya definido mediante Drag & Drop desde la tabla de variables PLC, desde un bloque de datos, desde la vista detallada o desde la interfaz del bloque.

4. Introduzca un comentario de línea. Este paso es opcional.
5. Repita los pasos 2 a 4 hasta que haya insertado todas las instrucciones necesarias.

Resultado

Después de confirmar la instrucción con la tecla Intro, el editor programador llevará a cabo las siguientes acciones:

- Comprobación de sintaxis: Las entradas erróneas aparecerán en rojo y en cursiva después de la comprobación. Además aparecerá un mensaje detallado de errores en la ventana de información.
- Cambio de mayúsculas y minúsculas: Las instrucciones se cambiarán en letras mayúsculas.
- Si ha introducido direcciones absolutas en la programación simbólica, se cambiarán a nombres simbólicos. Si no hay ningún nombre simbólico, se insertará "Tag_< Número>".
- Llamada de bloque: Se insertan y visualizan los parámetros necesarios.
- En el campo de entrada se inserta otra línea.

Consulte también

Reglas sobre AWL (Página 2767)

Introducir manualmente las instrucciones AWL (Página 2768)

Insertar operandos (Página 2771)

Emplear favoritos en AWL (Página 2774)

Insertar llamadas de bloque en AWL (Página 2776)

Insertar comentarios (Página 2781)

Seleccionar el tipo de datos de una instrucción AWL

Seleccionar un tipo de datos

Introducción

Algunas instrucciones pueden ejecutarse con varios tipos de datos diferentes. Si en el programa se utiliza alguna de estas instrucciones, en la ubicación concreta del programa es preciso definir un tipo de datos permitido para la instrucción en cuestión.

Los tipos de datos admisibles para una instrucción se indican en la lista desplegable correspondiente. Seleccionando una entrada de la lista desplegable se determina el tipo de datos de la instrucción. Si el tipo de datos de un operando difiere del tipo de datos de la instrucción y no puede convertirse de forma implícita, el operando se muestra en color rojo y aparece una ventana desplegable con el mensaje de error correspondiente.

Definir el tipo de datos de una instrucción

Introducción

Algunas instrucciones pueden ejecutarse con varios tipos de datos diferentes. Si se insertan este tipo de instrucciones en el programa, es preciso definir el tipo de datos adecuado para ellas en la ubicación concreta del programa.

Procedimiento

Para definir el tipo de datos de una instrucción mediante la lista desplegable, proceda del siguiente modo:

1. Arrastre la instrucción mediante Drag & Drop hasta la posición deseada del programa.
   En la lista desplegable de la instrucción insertada se visualiza la entrada "???" (no definida).
2. Haga clic en el triángulo situado en la esquina superior de la lista desplegable.
   Ésta se abre y se visualizan los tipos de datos admisibles para la instrucción.
3. Seleccione un tipo de datos de la lista desplegable.
   Aparece el tipo de datos seleccionado.

Insertar operandos

Los operandos de una instrucción AWL se pueden insertar manualmente o bien arrastrarlos mediante Drag & Drop desde la tabla de variables PLC, desde un bloque de datos, desde la vista detallada o desde la interfaz del bloque.

Nota

S7-1500: Las instrucciones de la Task Card "Instrucciones" que utilizan registros y acumuladores emulados no pueden utilizar variables con una longitud de 64 bits como operandos, p. ej. LInt. Si desea emplear variables de 64 bits, utilice las instrucciones de la librería global "Long Functions".

Procedimiento

Para insertar manualmente los operandos de una instrucción AWL, se deberá proceder de la siguiente manera:

1. Ponga el cursor a la derecha al lado de la operación.
2. Deje un espacio y a continuación inserte el operando.

Nota

Si en la programación simbólica introduce la dirección absoluta de un parámetro ya definido, esta dirección absoluta se sustituirá por el nombre simbólico del parámetro en cuanto confirme la entrada. Si el parámetro no se ha definido aún, en la tabla de variables PLC se introducirá una nueva variable con esta dirección absoluta y el nombre estándar "Tag_1". Cuando confirme la entrada, la dirección absoluta se sustituirá por el nombre simbólico "Tag_1".

3. Confirme el parámetro pulsando la tecla Intro.
4. Si todavía no ha definido el parámetro, puede hacerlo mediante el menú contextual directamente en el editor programador.
   Consulte también: "Declarar directamente el contenido de variables en el editor programador (Página 2577)"

O mediante Drag & Drop desde la tabla de variables PLC:

1. En el árbol del proyecto, seleccione la carpeta "Variables PLC" y abra la tabla de variables PLC.
2. Si está abierta la tabla de variables PLC, arrastre el símbolo desde la primera columna de la variable deseada hasta la respectiva posición del programa. Si la tabla de variables PLC no está abierta, abra la vista detallada. Arrastre la variable deseada desde la vista detallada hasta la respectiva posición del programa.

O mediante Drag & Drop desde la interfaz del bloque:

1. Abra la interfaz del bloque.
2. Arrastre el operando deseado desde la interfaz del bloque hasta la ventana de instrucciones.

Consulte también

Reglas sobre AWL (Página 2767)

Introducir manualmente las instrucciones AWL (Página 2768)

Insertar instrucciones AWL empleando la Task Card "Instrucciones" (Página 2769)

Emplear favoritos en AWL (Página 2774)

Insertar llamadas de bloque en AWL (Página 2776)

Insertar comentarios (Página 2781)

Utilizar y direccionar operandos (Página 2280)

Utilizar la función automática de autocompletar (Página 2637)

Mostrar u ocultar informaciones de variables

Introducción

Al margen de la representación absoluta o simbólica de los operandos, podrá mostrar u ocultar los comentarios sencillos o jerárquicos para documentar las variables. Esta información se recoge de la interfaz del bloque en el caso de las variables locales y las variables DB, y de la tabla de variables PLC en el caso de las variables válidas para toda la CPU.

Es posible visualizar la información de las variables para todos los bloques o solo para bloques abiertos individuales. Si selecciona la información de variables para todos los bloques, se muestra la información de las variables de todos los bloques abiertos actuales y futuros.

La información de las variables se puede volver a ocultar en cualquier momento. Si se ha ocultado la información de las variables de todos los bloques, se puede volver a ver para bloques abiertos individualmente.

Si se selecciona la visualización de información de variables con comentarios jerárquicos, en el caso de las variables estructuradas también se muestran los comentarios de los niveles de estructura de orden superior. En ese caso, esa información se muestra entre paréntesis después del comentario de la variable, y los comentarios de los diferentes niveles se separan con un punto. Si en un nivel de estructura no hay ningún comentario para una variable, el comentario en cuestión no se visualiza, lo que puede reconocerse por dos puntos seguidos.

Mostrar u ocultar información de variables para todos los bloques

Para mostrar u ocultar la información de variables para todos los bloques, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Para visualizar la información de las variables, seleccione las opciones "Mostrar" o "Información de la variable con jerarquía" en la lista desplegable "Información de la variable", en función de si desea visualizar los comentarios sencillos o jerárquicos.
4. Para ocultar la información de las variables, seleccione la opción "Ocultar" en la lista desplegable "Información de la variable".
   La información de las variables se muestra u oculta para todos los bloques abiertos. Si abre más bloques, la información de las variables se mostrará u ocultará dependiendo del ajuste seleccionado.

Mostrar u ocultar información de variables para un bloque abierto

Para mostrar u ocultar la información de variables para un bloque abierto, proceda del siguiente modo:

1. Para visualizar la información de las variables, seleccione las opciones "Mostrar información de la variable" o "Información de la variable con jerarquía" en la lista desplegable "Muestra información sobre la variable", en función de si desea visualizar los comentarios sencillos o jerárquicos.
2. Para ocultar la información de las variables, seleccione la opción "Ocultar información de la variable" en la lista desplegable "Muestra información sobre la variable".
   Se muestra u oculta la información sobre las variables.

Emplear favoritos en AWL

Insertar elementos AWL en favoritos

Requisitos

- Hay un bloque abierto.
- La Task Card "Instrucciones" tiene configurado el modo multipaleta, o bien los Favoritos se muestran también en el editor.

Procedimiento

Para agregar instrucciones a los Favoritos, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Maximice la paleta "Instrucciones básicas".
3. En la paleta "Instrucciones básicas", navegue hasta la instrucción que desea agregar a los Favoritos.
4. Arrastre la instrucción hasta la paleta "Favoritos" o hasta el área de Favoritos del editor de programación mediante Drag & Drop.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Insertar instrucciones AWL empleando favoritos (Página 2775)

Eliminar instrucciones AWL retirándolas de favoritos (Página 2775)

Sinopsis del editor de programación (Página 2619)

Insertar instrucciones AWL empleando favoritos

Requisitos

- Hay un bloque abierto.
- Existen Favoritos.

Procedimiento

Para insertar una instrucción en un programa desde los Favoritos, proceda del siguiente modo:

1. Arrastre la instrucción deseada mediante Drag & Drop desde los Favoritos hasta la posición deseada.

O bien:

1. Marque el punto del programa en el que quiere insertar la instrucción.
2. En los Favoritos, haga clic en la instrucción que desea insertar.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Sinopsis del editor de programación (Página 2619)

Insertar elementos AWL en favoritos (Página 2774)

Introducir manualmente las instrucciones AWL (Página 2768)

Insertar instrucciones AWL empleando la Task Card "Instrucciones" (Página 2769)

Eliminar instrucciones AWL retirándolas de favoritos (Página 2775)

Eliminar instrucciones AWL retirándolas de favoritos

Requisitos

Hay un bloque lógico abierto.

Procedimiento

Para eliminar instrucciones de los Favoritos, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la instrucción que desea eliminar.
2. Elija el comando "Eliminar instrucción" del menú contextual.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Insertar elementos AWL en favoritos (Página 2774)

Insertar instrucciones AWL empleando favoritos (Página 2775)

Sinopsis del editor de programación (Página 2619)

Insertar llamadas de bloque en AWL

Insertar manualmente llamadas de bloque

Podrá insertar llamadas para funciones (FCs) y bloques de función (FB). Los bloques de función se pueden llamar como instancia individual o como multiinstancia. Si se llama un bloque de función como instancia individual, guardará sus datos en un bloque de datos propio. Si se llama un bloque de función como multiinstancia, guardará sus datos en el bloque de datos de instancia del bloque de función que se está abriendo.

Requisitos

Existe un segmento.

Insertar llamada de una función (FC)

Para llamar una función en un segmento AWL, se deberá proceder de la siguiente manera:

1. Inserte "CALL <Nombre de la función que se desea abrir>".
   Sustituya el fichero "<Nombre de la función que se desea abrir>" por el nombre de la función que quiere llamar.
2. Confirme la entrada realizada pulsando la tecla Intro.
   - Si ya existe la función, se insertarán en la llamada todos los parámetros de función.
   - Si todavía no existe la función, usted mismo deberá insertar los parámetros.

Insertar llamada de un bloque de función (FB)

Para insertar la llamada para un bloque de función (FB), deberá proceder de la siguiente manera:

1. Inserte "CALL <Nombre del bloque de función que se desea abrir>[, <Nonmbre del bloque de datos de instancia>]".
   Sustituya el fichero "<Nombre del bloque de función que se desea abrir>" por el nombre del bloque de función que quiere llamar. Es opcional la entrada del bloque de datos de instancia.
2. Confirme la entrada realizada pulsando la tecla Intro.
   Si no se indica el bloque de datos de instancia o si el bloque indicado no existe, se abrirá el cuadro de diálogo "Opciones de llamada". En este caso ejecute los pasos 3 y 4.
3. Indique en él si desea llamar el bloque como instancia individual o como multiinstancia.
   - Después de hacer clic en el botón de "Instancia individual", deberá indicar un nombre en el cámpo de entrada "Nombre" para el bloque de datos al cual se deberá asignar la llamada.
   - Después de hacer clic en el botón de "Multiinstancia", deberá indicar en el campo de entrada "Nombre en la interfaz" el nombre de la variable con el que registrará el bloque abierto de función como variable estática en la interfaz del bloque que se desea llamar.
4. Confirme las entradas realizadas con "Aceptar".

Resultado

Al llamar las funciones existentes y los módulos de función existentes con el bloque de datos de instancia, se insertarán en la llamada los parámetros de la función o del bloque de función. A continuación usted deberá suministrar estos parámetros.

Consulte también: Transferencia de parámetros en la llamada de un bloque (Página 2268)

Si al llamar un bloque de función, especifica un bloque de datos de instancia que no existe aún, éste se creará.

Consulte también

Instancias individuales (Página 2265)

Multiinstancias (Página 2266)

Insertar llamadas de bloque mediante Drag & Drop (Página 2777)

Actualizar las llamadas de bloque en AWL (Página 2779)

Cambiar el tipo de instancia (Página 2780)

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Insertar llamadas de bloque mediante Drag & Drop

Podrá insertar llamadas para las funciones existentes (FCs) y módulos de función (FBs) procedentes del árbol del proyecto mediante Drag & Drop. Los bloques de función se pueden llamar como instancia individual o como multiinstancia. Si se llama un bloque de función como

instancia individual, guardará sus datos en un bloque de datos propio. Si se llama un bloque de función como multiinstancia, guardará sus datos en el bloque de datos de instancia del bloque de función que se está abriendo.

Requisitos

- Existe un segmento.
- Existe la función que se desea llamar (FC) o el bloque de función que se abre (FB).

Insertar llamada de una función (FC)

Para llamar una función en un segmento AWL empleando Drag & Drop, se deberá proceder de la siguiente manera:

1. Arrastre la función desde el árbol del proyecto hasta el segmento.

Insertar llamada de un bloque de función (FB)

Para insertar la llamada para un bloque de función (FB), deberá proceder de la siguiente manera:

1. Arrastre el bloque de función desde el árbol del proyecto hasta el segmento.
   Se abre el cuadro de diálogo "Opciones de llamada".
2. Indique en él si desea llamar el bloque como instancia individual o como multiinstancia.
   - Después de hacer clic en el botón "Instancia individual", indique un nombre en el campo de entrada "Nombre" para el bloque de datos de instancia al cual se deberá asignar la llamada.
   - Si hace clic en el botón "Multiinstancia", introduzca en el campo de texto "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
3. Confirme las entradas realizadas con "Aceptar".

Resultado

La función o el bloque de función se inserta con sus respectivos parámetros. A continuación usted deberá suministrar estos parámetros.

Consulte también: Transferencia de parámetros en la llamada de un bloque (Página 2268)

Si al llamar un bloque de función, especifica un bloque de datos de instancia que no existe aún, éste se creará. Si ha llamado un bloque de función como multiinstancia, éste se introducirá en la interfaz como variable estática.

Consulte también

Instancias individuales (Página 2265)

Multiinstancias (Página 2266)

Insertar manualmente llamadas de bloque (Página 2776)

Actualizar las llamadas de bloque en AWL (Página 2779)

Cambiar el tipo de instancia (Página 2780)

Actualizar las llamadas de bloque en AWL

Si cambian los parámetros de interfaz de un bloque llamado, la llamada no podrá ejecutarse correctamente. Actualice las llamadas de bloque con objeto de evitar llamadas incoherentes.

Existen dos posibilidades de actualizar llamadas de bloque, a saber:

- Actualización explícita en el editor de programación.
  Se actualizan las llamadas de bloque del bloque abierto.
- Actualización implícita durante la compilación.
  Se actualizan todas las llamadas de bloque en el programa, así como los tipos de datos PLC utilizados.

Actualizar llamadas de bloque en el editor de programación

Para actualizar una llamada de bloque en un bloque, proceda del siguiente modo:

1. Abra el bloque en el editor de programación.
2. Haga clic en el botón "Actualizar llamadas de bloque incoherentes" de la barra de herramientas.

O bien:

1. Abra el bloque en el editor de programación.
2. Haga clic con el botón derecho del ratón en la instrucción que contiene la llamada de bloque.
3. Elija el comando "Actualizar" del menú contextual.
   Se abre el cuadro de diálogo "Sincronización de la interfaz". En él se visualizan las diferencias entre la interfaz utilizada y la interfaz modificada del bloque llamado.
4. Si desea actualizar la llamada de bloque, haga clic en "Aceptar". Para cancelar la actualización, haga clic en "Cancelar".

Actualizar llamadas de bloque durante la compilación

Para actualizar implícitamente todas las llamadas de bloque y usos de tipos de datos PLC durante la compilación, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Llamadas de bloque (Página 2263)

Insertar manualmente llamadas de bloque (Página 2776)

Insertar llamadas de bloque mediante Drag & Drop (Página 2777)

Cambiar el tipo de instancia (Página 2780)

Cambiar el tipo de instancia

Tipo de instancia

Los bloques de función se pueden llamar de dos maneras diferentes, a saber:

- Como instancia individual
- Como multiinstancia

Consulte también: Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

El tipo de instancia definido se puede modificar en todo momento.

Requisitos

El programa de usuario contiene una llamada de bloque.

Procedimiento

Para cambiar el tipo de instancia de un bloque de función, proceda del siguiente modo:

1. Abra el bloque lógico y seleccione la llamada de bloque.
2. Elija el comando "Cambiar instancia" del menú contextual.
   Se abre el cuadro de diálogo "Opciones de llamada".
3. Haga clic en el botón "Instancia individual" o "Multiinstancia".
   - Si selecciona el tipo de instancia "Instancia individual", introduzca un nombre para el bloque de datos que debe asignarse al bloque de función.
   - Si selecciona el tipo de instancia "Multiinstancia", introduzca en el campo de texto "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
4. Confirme las entradas con "Aceptar".

Nota

Las instancias individuales y multiinstancias anteriores no se borran automáticamente.

Consulte también

Insertar manualmente llamadas de bloque (Página 2776)

Insertar llamadas de bloque mediante Drag & Drop (Página 2777)

Actualizar las llamadas de bloque en AWL (Página 2779)

Insertar comentarios

Comentar el código de programa

Tiene varias posibilidades para comentar los programas AWL:

- Comentario de línea
  Los comentarios de línea comienzan con "//" y se extienden hasta el final de la línea.
- Sección de comentario dentro de una línea
  Una sección de comentario se inicia con "(*" y se cierra con "*)". Puede haber una sección de comentario en cualquier posición de la línea, pero no debe abarcar más de una línea.

Requisitos

Existe un segmento.

Insertar comentarios de línea

Para insertar un comentario de línea en una línea de instrucción, se deberá proceder de la siguiente manera:

1. Inserte al final de la línea de instrucción "//".
2. Introduzca el comentario.

Insertar una sección de comentario en una línea

Para insertar una sección de comentario eb una línea de instrucción, proceda del siguiente modo:

1. Introduzca "(*" en la línea de instrucción en el lugar que desee.
2. Introduzca el comentario detrás de "(*".
3. Introduzca "*)" detrás del comentario.

Desactivar una o varias líneas mediante comentarios

Para desactivar código del programa mediante comentarios, proceda del siguiente modo:

1. Seleccione las líneas de código que desea comentar.
2. Haga clic en el botón "Marcar la selección como comentario" del editor.
3. En las líneas marcadas se inserta "//" al principio de la línea. El código siguiente se interpreta como comentario. Si se desactivan líneas que ya tienen un comentario de línea, también se inserta "//". Si a continuación se vuelven a activar las líneas, se mantendrán los comentarios iniciales.

Activar líneas de comentario

Para volver a activar líneas de comentario como código, proceda del siguiente modo:

1. Seleccione las líneas de código que desea activar.
2. Haga clic en el botón "Eliminar comentario" del editor.
3. Se elimina el identificador "//" de los comentarios de líneas al principio de la línea.

Resultado

Los comentarios se formatearán con un resalte.

Consulte también

Reglas sobre AWL (Página 2767)

Introducir manualmente las instrucciones AWL (Página 2768)

Insertar instrucciones AWL empleando la Task Card "Instrucciones" (Página 2769)

Insertar operandos (Página 2771)

Emplear favoritos en AWL (Página 2774)

Insertar llamadas de bloque en AWL (Página 2776)

Procesar instrucciones AWL

Seleccionar instrucciones AWL

Podrá seleccionar instrucciones individuales o todas las instrucciones del segmento.

Requisitos

Hay instrucciones AWL.

Seleccionar instrucciones AWL individuales

Para seleccionar instrucciones AWL individuales, se deberá proceder de la siguiente manera:

1. Ponga la marca de insertar delante del primer signo que desea marcar.
2. Pulse y mantenga pulsada la tecla izquierda del ratón.
3. Desplace el puntero del ratón detrás del último signo que desea marcar.
4. Suelte la tecla izquierda del ratón.

Seleccionar todas las instrucciones AWL del segmento

Para seleccionar todas las instrucciones del segmento, se deberá proceder de la siguiente manera:

1. Haga clic en el segmento cuyas instrucciones desea seleccionar.
2. Elija la instrucción de "Seleccionar todo" en el menú "Editar" o emplee la combinación de teclas <Strg+A>.

Consulte también

Copiar instrucciones AWL (Página 2783)

Cortar instrucciones AWL (Página 2784)

Insertar instrucciones AWL procedentes del portapapeles (Página 2784)

Borrar instrucciones AWL (Página 2785)

Copiar instrucciones AWL

Requisitos

Hay una instrucción AWL.

Procedimiento

Para copiar una instrucción AWL, se deberá proceder de la siguiente manera:

1. Marque la instrucción AWL que desee copiar.
2. Elija el comando "Copiar" del menú contextual.

Resultado

La instrucción AWL se copia y se guarda en el portapapeles.

Consulte también

Seleccionar instrucciones AWL (Página 2782)
Cortar instrucciones AWL (Página 2784)
Insertar instrucciones AWL procedentes del portapapeles (Página 2784)
Borrar instrucciones AWL (Página 2785)

Cortar instrucciones AWL

Requisitos

Hay una instrucción AWL.

Procedimiento

Para cortar una instrucción AWL, se deberá proceder de la siguiente manera:

1. Marque la instrucción AWL que desee cortar.
2. Elija el comando "Cortar" del menú contextual.

Resultado

La instrucción AWL se corta y se guarda en el portapapeles.

Consulte también

Seleccionar instrucciones AWL (Página 2782)
Copiar instrucciones AWL (Página 2783)
Insertar instrucciones AWL procedentes del portapapeles (Página 2784)
Borrar instrucciones AWL (Página 2785)

Insertar instrucciones AWL procedentes del portapapeles

Requisitos

Hay una instrucción AWL.

Procedimiento

Para insertar una instrucción AWL procedente del portapapeles, se deberá proceder de la siguiente manera:

1. Copiar una instrucción AWL o cortar una instrucción AWL.
2. Haga clic en el lugar del segmento en el que desee insertar la instrucción AWL.
3. Elija el comando "Insertar" del menú contextual.

Consulte también

Seleccionar instrucciones AWL (Página 2782)

Copiar instrucciones AWL (Página 2783)

Cortar instrucciones AWL (Página 2784)

Borrar instrucciones AWL (Página 2785)

Borrar instrucciones AWL

Requisitos

Hay una instrucción AWL.

Procedimiento

Para borrar una instrucción AWL, se deberá proceder de la siguiente manera:

1. Marque la instrucción AWL que desee borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Seleccionar instrucciones AWL (Página 2782)

Copiar instrucciones AWL (Página 2783)

Cortar instrucciones AWL (Página 2784)

Insertar instrucciones AWL procedentes del portapapeles (Página 2784)

Creación de programas SCL

Principios básicos de SCL

Lenguaje de programación SCL

Lenguaje de programación SCL

SCL (Structured Control Language) es un lenguaje de programación de alto nivel que se orienta a PASCAL. Este lenguaje se basa en la norma DIN EN-61131-3 (internacional: IEC 1131-3).

Esta norma estandariza los lenguajes de programación para autómatas programables. El lenguaje de programación SCL cumple el PLCopen Basis Level del lenguaje ST (texto estructurado) definido en esta norma.

Elementos del lenguaje

Además de incluir elementos típicos del PLC, como entradas, salidas, temporizadores o marcas, SCL incluye elementos de lenguajes de programación de alto nivel:

- Expresiones
- Asignaciones de valor
- Operadores

Control del programa

SCL ofrece instrucciones prácticas para el control del programa que permiten realizar, por ejemplo, ramas, bucles o saltos del programa.

Campo de aplicación

Por estos motivos, SCL es especialmente adecuado para los siguientes campos de aplicación:

- Gestión de datos
- Optimización de procesos
- Gestión de recetas
- Tareas matemáticas/estadísticas

Expresiones

Descripción

Las expresiones se calculan durante la ejecución del programa y devuelven un valor. Una expresión se compone de operandos (p. ej., de constantes, variables o llamadas de función)

y, opcionalmente, de operadores (p. ej., *, /, + ó -). Las expresiones se pueden combinar entre sí mediante operadores o estar anidadas las unas en las otras.

Orden de evaluación

La evaluación de la expresión se lleva a cabo en un orden determinado por los siguientes factores:

- Prioridad de los operadores implicados
- Orden izquierda-derecha
- Paréntesis

Tipos de expresiones

En función del operador, se pueden distinguir los siguientes tipos de expresiones:

- Expresiones aritméticas
  Las expresiones aritméticas están compuestas por un valor numérico o bien combinan dos valores o expresiones con operadores aritméticos.
- Expresiones de comparación
  Las expresiones de comparación comparan los valores de dos operandos y devuelven un valor booleano. El resultado es TRUE si se cumple la comparación y FALSE si no se cumple.
- Expresiones lógicas
  Las expresiones lógicas combinan dos operandos mediante operadores lógicos (AND, OR, XOR) o niegan operandos (NOT).

Uso de expresiones

El resultado de una expresión se puede usar de diferente manera:

- como asignación de valor a una variable
- como condición de una instrucción de control
- como parámetro para llamar un bloque o una instrucción

Consulte también

Operadores y su jerarquía (Página 2793)

Expresiones aritméticas (Página 2788)

Expresiones de comparación (Página 2790)

Expresiones lógicas (Página 2792)

Expresiones aritméticas

Descripción

Las expresiones aritméticas están compuestas por un valor numérico o bien combinan dos valores o expresiones con operadores aritméticos.

Los operadores aritméticos pueden procesar los tipos de datos compatibles con la CPU utilizada. Si dos operandos intervienen en la operación, el tipo de datos del resultado se determina según los criterios siguientes:

- Si ambos operandos son enteros con signo y tienen longitudes diferentes, el resultado obtiene el tipo de datos del entero más largo (p. ej. INT + DINT = DINT).
- Si ambos operandos son enteros sin signo y tienen longitudes diferentes, el resultado obtiene el tipo de datos del entero más largo (p. ej. USINT + UDINT = UDINT).
- Si un operando es un entero con signo y el otro operando es un entero sin signo, el resultado obtiene el siguiente tipo de datos superior con signo que cubre el entero sin signo (p. ej. SINT + USINT = INT).
  Una operación en que intervengan este tipo de operandos solo se puede ejecutar si no está ajustada la verificación CEI.
- Si un operando es un entero y el otro operando es un número en coma flotante, el resultado obtiene el tipo de datos del número en coma flotante (p. ej. INT + REAL = REAL).
- Si ambos operandos son números en coma flotante y tienen longitudes diferentes, el resultado obtiene el tipo de datos del número en coma flotante más largo (p. ej. REAL + LREAL = LREAL).
- El tipo de datos del resultado de una operación en que intervienen operandos del tipo de datos de los grupos "Temporizadores" y "Fecha y hora" figura en la tabla que aparece en el apartado "Tipos de datos de las expresiones aritméticas".
  Si la verificación CEI está activada, no es posible utilizar los tipos de datos de los grupos "Temporizadores" y "Fecha y hora".

Tipos de datos de las expresiones aritméticas

En la siguiente tabla se muestran los tipos de datos que se pueden utilizar en expresiones aritméticas:

Operación	Operador	1. Operando	2. Operando	Resultado
Potencia	**	Entero/número en coma flotante	Entero/número en coma flotante	Entero/número en coma flotante
Más unario	+	Entero/número en coma flotante TIME, LTIME	-	Entero/número en coma flotante TIME, LTIME
Menos unario	-	Entero/número en coma flotante TIME, LTIME	-	Entero/número en coma flotante TIME, LTIME
Multiplicación	*	Entero/número en coma flotante	Entero/número en coma flotante	Entero/número en coma flotante
		TIME, LTIME	Entero	TIME, LTIME

Operación	Operador	1. Operando	2. Operando	Resultado
División	/	Entero/número en coma flotante	Entero/número en coma flotante (distinto de 0)	Entero/número en coma flotante
		TIME, LTIME	Entero	TIME, LTIME
Función módulo	MOD	Entero	Entero	Entero
Suma	+	Entero/número en coma flotante	Entero/número en coma flotante	Entero/número en coma flotante
		TIME	TIME	TIME
		TIME	DINT	TIME
		LTIME	TIME, LTIME	LTIME
		LTIME	LINT	LTIME
		TOD	TIME	TOD
		TOD	DINT	TOD
		LTOD	TIME, LTIME	LTOD
		LTOD	LINT	LTOD
		DATE	LTOD	DTL
		DATE	TOD	• S7-300/400: DT • S7-1200/1500: DTL
		DT	TIME	DT
		LDT	TIME, LTIME	LDT
		DTL	TIME, LTIME	DTL
Resta	-	Entero/número en coma flotante	Entero/número en coma flotante	Entero/número en coma flotante
		TIME	TIME	TIME
		TIME	DINT	TIME
		LTIME [1)]	TIME, LTIME	LTIME
		LTIME	LINT	LTIME
		TOD	TIME	TOD
			DINT	TOD
		LTOD	TIME, LTIME	LTOD
		LTOD	LINT	LTOD
		DATE	DATE	• S7-300/400/1200: TIME • S7-1500: LTIME
		DT	TIME	DT
		LDT	TIME, LTIME	LDT
		DTL	TIME, LTIME	DTL
		DTL	DTL	• S7-1200: TIME • S7-1500: LTIME
[1)] Dentro de las expresiones no es posible combinar nanosegundos y milisegundos.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra una expresión aritmética:

```
SCL
"MyTag1":= "MyTag2" * "MyTag3";
```

Consulte también

Expresiones (Página 2786)

Tipos de datos (Página 2321)

Operadores y su jerarquía (Página 2793)

Expresiones de comparación

Descripción

Las expresiones de comparación comparan los valores de dos operandos y devuelven un valor booleano. El resultado es TRUE si se cumple la comparación y FALSE si no se cumple.

Los operadores de comparación pueden procesar los tipos de datos compatibles con la CPU utilizada. El tipo de datos del resultado es siempre BOOL.

Para formar expresiones de comparación se deben respetar las reglas siguientes:

- Todas las variables son comparables en los grupos siguientes de tipos de datos:
  - Enteros/números en coma flotante
  - Números binarios
  - Cadenas de caracteres
- En los siguientes tipos de datos/grupos de tipos de datos sólo son comparables las variables del mismo tipo:
  - TIME, LTIME
  - Fecha y hora
- Las cadenas de caracteres se comparan según el juego de caracteres ASCII. Para la comparación se utiliza la longitud de la variable y el valor numérico de cada carácter ASCII.
- No se permite usar las variables S5 TIME como operandos de comparaciones. Resulta necesario efectuar una conversión explícita de S5TIME a TIME o LTIME.

Tipos de datos de las expresiones de comparación

En la siguiente tabla se muestran los tipos de datos/grupos de tipos de datos que se pueden utilizar en expresiones de comparación:

Operación	Operador	1. Operando	2. Operando	Resultado
Comparación de igualdad, desigualdad	=, <>	Entero/número en coma flotante	Entero/número en coma flotante	BOOL
		Número binario	Número binario	BOOL
		Secuencia de caracteres	Secuencia de caracteres	BOOL
		TIME, LTIME	TIME, LTIME	BOOL
		Fecha y hora	Fecha y hora	BOOL
Comparación de menor, menor o igual, mayor, mayor o igual	<, <=, >, >=	Entero/número en coma flotante	Entero/número en coma flotante	BOOL
		Secuencias de bits (solo S7-1200/1500)	Secuencias de bits (solo S7-1200/1500)	BOOL
		Secuencia de caracteres	Secuencia de caracteres	BOOL
		TIME, LTIME	TIME, LTIME	BOOL
		Fecha y hora	Fecha y hora	BOOL

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El ejemplo siguiente muestra una expresión de comparación:

SCL

```
IF a > b THEN c:= a;
IF A > 20 AND B < 20 THEN C:= TRUE;
IF A<>(B AND C) THEN C:= FALSE;
```

Nota

La comparación en STRING y DT se realiza en el S7-300/400 internamente mediante instrucciones avanzadas. Para estas funciones no se admiten los siguientes operandos:

- Parámetro de una FC.
- Parámetro de entrada/salida de un FB del tipo STRUCT o ARRAY

Consulte también

Expresiones (Página 2786)

Tipos de datos (Página 2321)

Operadores y su jerarquía (Página 2793)

Expresiones lógicas

Descripción

Las expresiones lógicas combinan dos operandos mediante operadores lógicos (AND, OR, XOR) o niegan operandos (NOT).

Los operadores lógicas pueden procesar los tipos de datos compatibles con la CPU utilizada. El resultado de una expresión lógica es del tipo de datos BOOL si ambos operandos son del tipo de datos BOOL. Si al menos uno de los dos operandos es una secuencia de bits, el resultado también es una secuencia de bits y está determinado por el tipo del operando de mayor valor. Si, por ejemplo, se combina un operando del tipo BYTE con un operando del tipo WORD, el resultado es del tipo WORD.

Para combinar un operando de tipo BOOL con una secuencia de bits, primero debe convertirse explícitamente en una secuencia de bits.

Tipos de datos de las expresiones lógicas

En la siguiente tabla se muestran los tipos de datos que puede utilizar en expresiones lógicas:

Operación	Operador	1. Operando	2. Operando	Resultado
Negación	NOT	BOOL	-	BOOL
Operación lógica Y	AND o &	BOOL	BOOL	BOOL
		Secuencia de bits	Secuencia de bits	Secuencia de bits
Operación lógica O	OR	BOOL	BOOL	BOOL
		Secuencia de bits	Secuencia de bits	Secuencia de bits
Operación lógica O-exclusiva	XOR	BOOL	BOOL	BOOL
		Secuencia de bits	Secuencia de bits	Secuencia de bits

Ejemplo

El siguiente ejemplo muestra una expresión lógica:

SCL

```
IF "MyTag1" AND NOT "MyTag2" THEN c:=a;
MyTag:=ALPHA OR BETA;
```

Consulte también

Expresiones (Página 2786)

Tipos de datos (Página 2321)

Operadores y su jerarquía (Página 2793)

Operadores y su jerarquía

Operadores y su orden de evaluación

Las expresiones se pueden combinar entre sí mediante operadores o estar anidadas las unas en las otras.

El orden de evaluación de las expresiones depende de la jerarquía de los operadores y los paréntesis. Rigen las reglas siguientes:

- Los operadores aritméticos se evalúan antes de los operadores de comparación y estos antes de los operadores lógicos.
- Los operadores de la misma jerarquía se evalúan por orden de aparición de izquierda a derecha.
- Primero se evalúan las operaciones entre paréntesis.

La tabla siguiente ofrece una vista general de los operadores y su jerarquía:

Operador	Operación	Jerarquía
Expresiones aritméticas		
+ (Página 2788)	Más unario	2
- (Página 2788)	Menos unario	2
** (Página 2788)	Potencia	3
* (Página 2788)	Multiplicación	4
/ (Página 2788)	División	4
MOD (Página 2788)	Función módulo	4
+ (Página 2788)	Suma	5
- (Página 2788)	Resta	5
Expresiones de comparación		
< (Página 2790)	Menor	6
> (Página 2790)	Mayor	6
<= (Página 2790)	Menor o igual	6
>= (Página 2790)	Mayor o igual	6
= (Página 2790)	Igual	7
<> (Página 2790)	Diferente	7
Expresiones lógicas		
NOT (Página 2792)	Negación	3
AND (Página 2792) o & (Página 2792)	Y booleano	8
XOR (Página 2792)	O-exclusiva	9

Operador	Operación	Jerarquía
OR (Página 2792)	O booleano	10
Otras operaciones		
( ) (Página 2786)	Paréntesis	1
:= (Página 2794)	Asignación	11

Asignaciones de valor

Definición

Con una asignación de valor asigna el valor de una expresión a una variable. En la parte izquierda de la asignación se encuentra la variable que adopta el valor de la expresión que se encuentra a la derecha.

Como expresión se puede introducir también el nombre de una función. La función es llamada por la asignación de valor y devuelve su valor de función a la variable que se encuentra a la izquierda.

El tipo de datos de la asignación de valor está definido por el tipo de datos de la variable que se encuentra a la izquierda. El tipo de datos de la expresión que se encuentra a la derecha debe se compatible con este tipo de datos.

Encontrará más información sobre la compatibilidad y la conversión de tipos de datos en "Consulte también".

Asignaciones de valor para los tipos de datos STRUCT o PLC

Es posible asignar toda una estructura a otra estructura si ambas presentan una estructura idéntica y si coinciden tanto los tipos de datos como los nombres de sus elementos.

A un elemento individual de la estructura se le puede asignar una variable, una expresión u otro elemento de la estructura.

Asignaciones de valor para el tipo de datos ARRAY

Es posible asignar todo un ARRAY a otro ARRAY si coinciden los tipos de datos de los elementos del ARRAY así como los límites de los ARRAY.

A un elemento individual de un ARRAY se le puede asignar una variable, una expresión u otro elemento de ARRAY.

Asignaciones de valor del tipo de datos STRING

Es posible asignar un STRING a otro STRING.

A un elemento individual de STRING se le puede asignar otro elemento de STRING.

Asignaciones de valor para el tipo de datos ANY

Las variables con el tipo de datos ANY pueden asignarse solo a los siguientes objetos:

- Parámetros de entrada o datos locales temporales de FB que tengan también el tipo de datos ANY.
- Datos locales temporales de FC que tengan también el tipo de datos ANY.

Tenga en cuenta, que el puntero ANY permite apuntar solo a las áreas de memoria con el tipo de acceso "estándar".

Asignaciones de valor para el tipo de datos POINTER

En SCL no puede utilizarse POINTER en asignaciones de valor.

Ejemplos

En la siguiente tabla se muestran ejemplos de asignaciones de valor:

SCL	
`"MyTag1" := "MyTag2";`	`(* Asignación de una variable*)`
`"MyTag1" := "MyTag2" * "MyTag3";`	`(* Asignación de una expresión*)`
`"MyTag" := "MyFC"();`	`(* Llamada de una función que asigna su valor de función a la variable "MyTag"*)`
`#MyStruct.MyStructElement := "MyTag";`	`(* Asignación de una variable a un elemento de la estructura*)`
`#MyArray[2] := "MyTag";`	`(* Asignación de una variable a un elemento de ARRAY*)`
`"MyTag" := #MyArray[1,4];`	`(* Asignación de un elemento de ARRAY a una variable*)`
`#MyString[2] := #MyOtherString[5];`	`(* Asignación de un elemento de STRING a otro elemento de STRING*)`

Consulte también

Direccionar operandos (Página 2293)

Tipos de datos (Página 2321)

Operadores y su jerarquía (Página 2793)

Configuración de SCL

Sinopsis de la configuración de SCL

Sinopsis

Las tablas siguientes muestran los ajustes que se pueden efectuar en SCL:

Configuración del editor

Grupo	Ajuste	Descripción
Vista	Resaltar palabras clave	Notación en la que se muestran las palabras clave del lenguaje de programación. Se puede elegir entre mayúsculas y minúsculas o una notación conforme a las convenciones del lenguaje de programación Pascal.

Ajuste predeterminado para bloques nuevos

Al crear bloques, se utilizan los siguientes ajustes como valores estándar. Estos ajustes se pueden modificar posteriormente en las propiedades del bloque.

Grupo	Ajuste	Descripción
Compilación	Crear información avanzada de estado	Permite observar todas las variables de un bloque. No obstante, la memoria necesaria del programa y los tiempo de ejecución aumentan al usar esta opción.
	Comprobar límites de ARRAY[1)]	Comprueba en el tiempo de ejecución si los índices del campo se encuentran en el rango declarado para un ARRAY. Si un índice del campo supera el rango admisible, la salida de habilitación ENO del bloque se pone a "0".
	Asignar ENO automáticamente	Comprueba en el tiempo de ejecución si se han producido errores al procesar determinadas instrucciones. Si se produce un error en el tiempo de ejecución, la salida de habilitación ENO del bloque se pone a "0".
[1)]Para las CPU de las series S7-300/400 vale: si se rebasan por exceso los límites de ARRAY, la salida de habilitación ENO devuelve FALSE. Para las CPU de las series S7-1200/1500 vale: si se rebasan por exceso los límites de ARRAY, la salida de habilitación ENO no devuelve FALSE. Véase "Direccionar variables estructuradas (Página 2298)" para las posibilidades de consulta de errores.		

Consulte también

Mecanismo EN/ENO (Página 2539)

Modificar la configuración (Página 2796)

Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Modifique la configuración.

Resultado

El cambio se aplica sin necesidad de guardarlo explícitamente.

Consulte también

Sinopsis de la configuración de SCL (Página 2795)

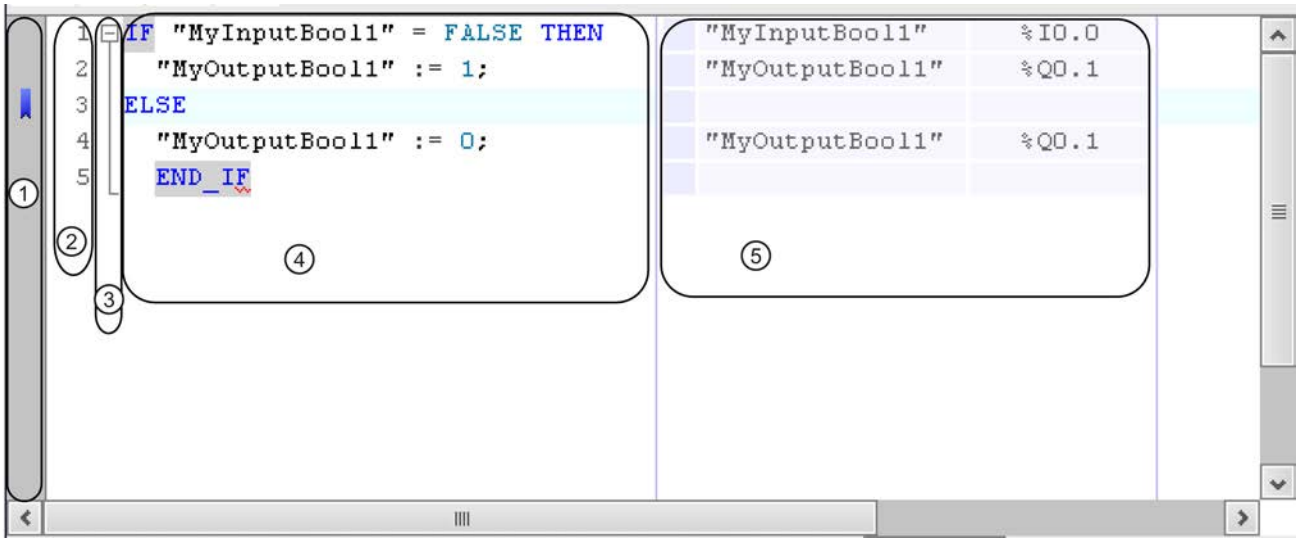
La ventana de programación de SCL

Vista general de la ventana de programación

Función

La ventana de programación es el área de trabajo en la que se introduce el programa SCL.

La figura siguiente muestra la ventana de programación de SCL:



La ventana de programación incluye las áreas siguientes:

Rango	Significado
① Barra lateral	En la barra lateral se pueden insertar marcadores y puntos de parada.
② Números de línea	A la izquierda se muestran los números de línea junto al código del programa.
③ Vista de esquema	La vista de esquema marca las secciones de código relacionadas.
④ Área del código	En el área del código se edita el programa SCL.
⑤ Visualización de los operandos absolutos	En esta tabla se muestra la asignación de los operandos simbólicos a las direcciones absolutas.

Consulte también

Uso de marcadores (Página 2801)

Ajuste de la ventana de programación (Página 2798)

Formatear un código SCL (Página 2799)

Expandir y contraer secciones de código (Página 2800)

Ajuste de la ventana de programación

Introducción

La visualización de la ventana de programación y del código de programa se puede ajustar de la siguiente manera:

- Ajuste del tipo, tamaño y color de la fuente
- Ajuste del ancho de tabulación
- Visualización de los números de línea
- Mostrar u ocultar los operandos absolutos

Ajuste del tipo, tamaño y color de la fuente

Para ajustar el tipo, tamaño y color de la fuente, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General > Editores scripts/textos".
3. Ajuste el tipo y tamaño de fuente deseado o seleccione los colores de la fuente para los diferentes elementos del lenguaje.

Ajuste del ancho de tabulación

Para que la visualización del programa sea más clara, se insertan sangrías en las líneas en función de la sintaxis. El ancho de la sangría se define con el ancho de tabulación.

Para ajustar el ancho de tabulación, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General > Editores scripts/textos".
3. Ajuste el ancho de tabulación.

Visualización de los números de línea

Para mostrar los números de línea, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "General > Editores scripts/textos".
3. Active la opción "Mostrar números de línea".

Mostrar y ocultar los operandos absolutos

En caso necesario, puede ver la asignación de operandos simbólicos y absolutos en una tabla junto al código del programa.

Para mostrar u ocultar los operandos absolutos, proceda del siguiente modo:

1. Haga clic en el botón "Operandos absolutos/simbólicos" de la barra de herramientas.
   Aparecen los operandos absolutos.
2. Para desplazar el visor, haga clic en la tabla y arrástrelo hasta la posición deseada mientras mantiene pulsado el botón del ratón.
3. Para modificar el ancho de la tabla, haga clic en el borde derecho o izquierdo de la tabla y arrastre hacia la derecha o la izquierda mientras mantiene pulsado el botón del ratón.

Consulte también

Uso de marcadores (Página 2801)

Vista general de la ventana de programación (Página 2797)

Formatear un código SCL (Página 2799)

Expandir y contraer secciones de código (Página 2800)

Formatear un código SCL

Introducción

Para proporcionar una estructura más clara al programa, es posible activar o desactivar manualmente la sangría de líneas o formatear secciones de código. Al formatear secciones de código observe los siguientes puntos:

- Se formatea según los ajustes generales de sangría, pero se sangra siempre la línea o sección como mínimo. Si ha seleccionado el ajuste "Smart, se eliminarán asimismo espacios innecesarios de la instrucción SCL.
- Solo pueden formatearse secciones de código sintácticamente correctas.
- Si coloca una marca de inserción en la primera o la última línea de una instrucción para el control del programa, p. ej. en una instrucción IF en la línea con el "IF", se formateará toda la instrucción.
- Si selecciona texto, solo se formateará el texto seleccionado.

Anulación y aplicación individual de sangrías

Para anular y aplicar sangrías en líneas individuales, proceda del siguiente modo:

1. Haga clic en la línea en la que desea aplicar o anular la sangría.
2. Seleccione en la barra de herramientas del editor del programa el botón "Aplicar sangría" o "Anular sangría".

Nota

El ancho de la sangría se define en "Herramientas > Configuración".

Formatear secciones de código

Para formatear secciones de código, proceda del siguiente modo:

1. Seleccione el texto que desea formatear o sitúe la marca de inserción en la línea en cuestión.
2. Seleccione el botón "Formatear automáticamente el texto seleccionado" en la barra de herramientas del editor de programación.

Consulte también

Uso de marcadores (Página 2801)

Vista general de la ventana de programación (Página 2797)

Ajuste de la ventana de programación (Página 2798)

Expandir y contraer secciones de código (Página 2800)

Sinopsis de la configuración de los editores de scripts y de textos (Página 281)

Expandir y contraer secciones de código

Introducción

Las instrucciones SCL pueden abarcar varias líneas. Ejemplos de ello son las instrucciones de control del programa o las llamadas de bloque.

Estas instrucciones se marcan del siguiente modo:

- Una vista de esquema entre el número de la línea y el código del programa marca toda la sección de código.
- Si marca la palabra clave de apertura, se marca automáticamente la palabra clave de cierre.

Procedimiento

Para expandir o contraer las secciones de código, proceda del siguiente modo:

1. Haga clic en el símbolo menos de la vista de esquema.
   La sección de código se contrae.
2. Haga clic en el símbolo más de la vista de esquema.
   La sección de código se expande.

Consulte también

Uso de marcadores (Página 2801)

Vista general de la ventana de programación (Página 2797)

Ajuste de la ventana de programación (Página 2798)

Formatear un código SCL (Página 2799)

Uso de marcadores

Principios básicos de los marcadores

Función

Los marcadores permiten marcar puntos del programa en programas extensos para encontrarlos fácilmente con el fin de retocarlos posteriormente. Los marcadores se muestran en la barra lateral de la ventana de programación. Mediante comandos de menú se navega entre varios marcadores de un bloque.

Los marcadores se guardan con el proyecto y, por tanto, están disponibles para todas las personas que editen el bloque. Sin embargo, no se cargan en dispositivos.

Durante la comparación de bloques, los marcadores no se evalúan.

Consulte también

Inserción de marcadores (Página 2801)

Navegación por marcadores (Página 2802)

Borrado de marcadores (Página 2803)

Inserción de marcadores

Requisitos

El bloque SCL está abierto.

Procedimiento

Para activar un marcador, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la línea deseada de la barra lateral.
2. Elija el comando "Marcadores > Activar" del menú contextual.

O bien:

1. Haga clic en la línea en la que desea colocar el marcador.
2. Haga clic en el botón "Activar/borrar marcadores" de la barra de herramientas.

O bien:

1. Mantenga pulsada la tecla <Ctrl>.
2. Haga clic en la línea de la barra lateral en la que quiere activar el marcador.

Resultado

Se inserta un marcador en el código del programa.

Consulte también

Principios básicos de los marcadores (Página 2801)

Navegación por marcadores (Página 2802)

Borrado de marcadores (Página 2803)

Navegación por marcadores

Requisitos

En un bloque se han activado varios marcadores.

Procedimiento

Para navegar entre marcadores, proceda del siguiente modo:

1. Inserte la marca de inserción en el código del programa.
2. Seleccione en el menú "Edición" el comando "Ir a > Marcador siguiente" o "Ir a > Marcador anterior".

O bien:

1. Inserte la marca de inserción en el código del programa.
2. Haga clic en el botón "Ir al marcador siguiente" o "Ir al marcador anterior" de la barra de herramientas del editor de programas.

O bien:

1. Haga clic en la barra lateral.
2. Elija el comando "Marcadores > Siguiente " o "Marcadores > Anterior" del menú contextual.

Resultado

Se resalta la línea con el marcador.

Consulte también

Principios básicos de los marcadores (Página 2801)
Inserción de marcadores (Página 2801)
Borrado de marcadores (Página 2803)

Borrado de marcadores

Es posible borrar marcadores individuales o todos los marcadores del bloque o la CPU.

Borrado de marcadores individuales

Para borrar un marcador individual, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la línea de la barra lateral en la que desea borrar el marcador.
2. Elija el comando "Marcadores > Borrar" del menú contextual.

O bien:

1. Haga clic en la línea en la que desea borrar el marcador.
2. Elija el comando "Marcador > Borrar" del menú "Edición".

O bien:

1. Haga clic en la línea en la que desea borrar el marcador.
2. Haga clic en el botón "Activar/borrar marcadores" de la barra de herramientas.

Borrado de todos los marcadores de un bloque

Para borrar todos los marcadores de un bloque, proceda del siguiente modo:

1. Haga clic en la barra lateral con el botón derecho del ratón.
2. Elija el comando "Marcadores > Borrar todos del bloque" del menú contextual.

O bien:

1. Elija el comando "Marcadores > Borrar todos del bloque" del menú "Edición".

Consulte también

Principios básicos de los marcadores (Página 2801)

Inserción de marcadores (Página 2801)

Navegación por marcadores (Página 2802)

Entrada de instrucciones SCL

Reglas sobre las instrucciones SCL

Instrucciones en SCL

SCL conoce los siguientes tipos de instrucciones:

- Asignaciones de valor
  Las asignaciones de valor sirven para asignar a una variable un valor constante, el resultado de una expresión o el valor de otra variable.
- Instrucciones para el control del programa
  Las instrucciones para el control del programa sirven para realizar ramas, bucles o saltos del programa.
- Otras instrucciones de la Task Card "Instrucciones"
  La Task Card "Instrucciones" ofrece una amplia selección de instrucciones estándar que pueden utilizarse en el programa SCL.
- Llamadas de bloque
  Las llamadas de bloque sirven para editar los subprogramas que han sido trasvasados a otros bloques y para posprocesar sus resultados.

Reglas

Al introducir instrucciones SCL, es necesario observar las reglas siguientes:

- Las instrucciones SCL pueden abarcar varias líneas.
- Todas las instrucciones acaban con un punto y coma (;).
- No se distingue entre mayúsculas y minúsculas.
- Los comentarios sólo sirven como documentación del programa. No influyen en la ejecución del programa.

Ejemplos

En los siguientes ejemplos se muestran los diferentes tipos de instrucciones:

SCL

```
// Ejemplo de una asignación de valor
"MyTag":= 0;
// Ejemplo de una llamada de bloque
```

```
SCL
"MyDB"."MyFB" (ParamInput:= 10);
// Ejemplo de una instrucción para el control del programa
WHILE "Counter" < 10 DO
    "MyTAG" := "MyTag" + 2;
END_WHILE;
```

Consulte también

Principios básicos de SCL (Página 2786)

Entrada manual de instrucciones SCL

Requisitos

Hay un bloque SCL abierto.

Procedimiento

Para introducir una instrucción SCL, proceda del siguiente modo:

1. Introduzca la sintaxis de la instrucción mediante el teclado.
   Al introducirla, se dispone de la función de autocompletar. Ofrece todas las instrucciones y operandos que están permitidos en la posición actual.
2. Seleccione con la función de autocompletar la instrucción o el operando deseados.
   Si selecciona una instrucción que requiere que se indiquen operandos, en el programa se insertarán comodines para los operandos. Los comodines de los operandos se resaltan en color amarillo. Se selecciona el primer comodín.
3. Sustituya el comodín por un operando.
4. Navegue con la tecla <TAB> por todos los comodines y sustitúyalos por operandos.

Nota

Es posible arrastrar un operando ya definido desde la tabla de variables PLC o la interfaz del bloque al programa mediante la función Drag & Drop. Para sustituir un operando ya insertado, deje el puntero del ratón brevemente sobre el operando que desea sustituir antes de soltar el botón del ratón. De ese modo el operando se marca y al soltar el botón del ratón se sustituye por el nuevo.

Resultado

Se inserta la instrucción.

El editor de programación realiza una comprobación de la sintaxis. Las entradas incorrectas se muestran en color rojo y cursiva. Asimismo, se muestra un mensaje de error detallado en la ventana de inspección.

Consulte también

Utilizar la función automática de autocompletar en los lenguajes de programación textuales (Página 2639)

Conversión de tipos de datos (Página 2373)

Ampliar y reducir una lista de parámetros (Página 2821)

Inserción de instrucciones SCL desde la Task Card "Instrucciones"

La Task Card "Instrucciones" ofrece una amplia selección de instrucciones que pueden utilizarse en el programa SCL. En la Task Card "Instrucciones" también están disponibles las instrucciones específicas de SCL para el control del programa.

Requisitos

Hay un bloque SCL abierto.

Procedimiento

Para insertar una instrucción SCL en un programa mediante la Task Card "Instrucciones", proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Para insertar la instrucción, elija uno de los siguientes pasos:
   - Navegue hasta la instrucción SCL que quiere insertar y arrástrela mediante Drag & Drop a la línea deseada del código de programa. El punto de inserción se marca con un rectángulo verde.
   - Marque el punto del código de programa en el que quiere insertar la instrucción y haga doble clic en la instrucción que quiere insertar.

   La instrucción inserta en el programa. Los comodines de los operandos se resaltan en color amarillo. Se selecciona el primer comodín.
3. Sustituya el comodín por un operando. También puede arrastrar una variable de la interfaz o la tabla de variables PLC mediante Drag & Drop hasta el comodín.
4. Navegue con la tecla <TAB> por todos los comodines y sustitúyalos por operandos.

Resultado

Se inserta la instrucción.

El editor de programación realiza una comprobación de la sintaxis. Las entradas incorrectas se muestran en color rojo y cursiva. Asimismo, se muestra un mensaje de error detallado en la ventana de inspección.

Consulte también

Utilizar la función automática de autocompletar en los lenguajes de programación textuales (Página 2639)

Conversión de tipos de datos (Página 2373)

Ampliar y reducir una lista de parámetros (Página 2821)

Definir el tipo de datos de una instrucción SCL

Principios básicos de los tipos de datos de instrucciones SCL

Introducción

Las instrucciones SCL que se emplean para programar los bloques utilizan tipos de datos determinados para determinar el valor de la función. Entre ellas hay instrucciones SCL para las que solo puede emplearse un tipo de datos concreto. En estas instrucciones no es posible cambiar el tipo de datos. No obstante, la mayoría de las instrucciones SCL puede trabajar con distintos tipos de datos. Se distinguen los dos tipos de instrucciones siguientes:

- Instrucciones en las que el tipo de datos del valor de la función queda determinado por el tipo de datos de los parámetros de entrada. La mayoría de instrucciones son de este tipo.
- Instrucciones en las que el tipo de datos está predeterminado. Son las instrucciones que se listan en la tabla inferior.

En el segundo grupo es necesario modificar el tipo de datos predeterminado cuando éste difiere del tipo de datos del parámetro de entrada utilizado. En principio es posible modificar el tipo de datos con la sintaxis siguiente:

```
_<tipo de datos>
```

Instrucciones SCL con tipo de datos predeterminado

La tabla siguiente muestra las instrucciones SCL con tipos de datos predeterminados:

Instrucción	Tipo de datos predeterminado
CEIL	DINT
DECO	DWORD
FLOOR	DINT
NORM_X	REAL
PEEK	BYTE
SCALE_X	INT
TRUNC	DINT
CONCAT	STRING

Consulte también

Modificar el tipo de datos de una instrucción SCL (Página 2808)

Ejemplo de modificación del tipo de datos de una instrucción SCL (Página 2809)

Modificar el tipo de datos de una instrucción SCL

Procedimiento

Para insertar una instrucción SCL y modificar a continuación su tipo de datos, proceda del siguiente modo:

1. Arrastre la instrucción mediante Drag & Drop hasta la posición deseada del programa.
2. Indique los operandos para la instrucción.
   El tipo de datos del valor de función se define de acuerdo con los parámetros de entrada o bien se utiliza el tipo de datos predeterminado de la instrucción.
3. Complemente el nombre de la instrucción con "_<tipo de datos>".
   "<tipo de datos>" es el tipo de datos que se requiere para la instrucción.

Consulte también

Principios básicos de los tipos de datos de instrucciones SCL (Página 2807)

Ejemplo de modificación del tipo de datos de una instrucción SCL (Página 2809)

Tipo de datos de temporizadores y contadores CEI

Los temporizadores y contadores CEI son bloques de función internos del sistema que requieren un bloque de datos de instancia. El bloque de datos de instancia se puede crear como instancia individual o como multiinstancia. El tipo de datos del bloque de datos de instancia depende de la instrucción correspondiente. Sin embargo, en las CPU de las series S7-1200 y S7-1500 es posible ejecutar las instrucciones con distintos tipos de datos para adaptarlas a sus necesidades.

Si el tipo de datos ajustado para el bloque de datos de instancia no coincide con el tipo de datos de los parámetros de entrada, se lleva a cabo una conversión implícita, si es posible. Si la conversión no es posible, aparece un mensaje de error.

Procedimiento

Para modificar eltipo d edatos del bloque de datos de instancia de temporizadores o contadores CEI, proceda del siguiente modo:

1. Abra el bloque en el que vaya a llamar al temporizador o contador CEI.
   Dependiendo del tipo de instancia del bloque de datos de instancia, aparecerá un rectángulo con borde verde delante (multiinstancia) o detrás (instancia individual) del nombre del bloque de datos de instancia.
2. Haga clic en el rectángulo con el borde verde.
   Se abrirá una lista desplegable con los tipos de datos posibles para el bloque de datos de instancia.
3. Seleccione el tipo de datos que desee utilizar.

Ejemplo de modificación del tipo de datos de una instrucción SCL

Modificación del tipo de datos predeterminado de la instrucción "Descodificar" (DECO)

Si se inserta la instrucción "Descodificar" en el programa, el tipo de datos DWORD ya estará predeterminado.

```
"Tag_Result" := DECO(IN := "Tag_Value");
```

Para modificar el tipo de datos de DWORD a BYTE posteriormente, modifique el código de programa del siguiente modo:

```
"Tag_Result_BYTE" := DECO_BYTE(IN := "Tag_Value");
```

Consulte también

Principios básicos de los tipos de datos de instrucciones SCL (Página 2807)

Modificar el tipo de datos de una instrucción SCL (Página 2808)

Mostrar u ocultar informaciones de variables

Introducción

Al margen de la representación absoluta o simbólica de los operandos, podrá mostrar u ocultar los comentarios sencillos o jerárquicos para documentar las variables. Esta información se recoge de la tabla de variables PLC.

Es posible visualizar la información de las variables para todos los bloques o solo para bloques abiertos individuales. Si selecciona la información de variables para todos los bloques, se muestra la información de las variables de todos los bloques abiertos actuales y futuros.

La información de las variables se puede volver a ocultar en cualquier momento. Si se ha ocultado la información de las variables de todos los bloques, se puede volver a ver para bloques abiertos individualmente.

Si se selecciona la visualización de información de variables con comentarios jerárquicos, en el caso de las variables estructuradas también se muestran los comentarios de los niveles de estructura de orden superior. En ese caso, esa información se muestra entre paréntesis

después del comentario de la variable, y los comentarios de los diferentes niveles se separan con un punto. Si en un nivel de estructura no hay ningún comentario para una variable, el comentario en cuestión no se visualiza, lo que puede reconocerse por dos puntos seguidos.

Mostrar u ocultar información de variables para todos los bloques

Para mostrar u ocultar la información de variables para todos los bloques, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Para visualizar la información de las variables, seleccione las opciones "Mostrar" o "Información de la variable con jerarquía" en la lista desplegable "Información de la variable", en función de si desea visualizar los comentarios sencillos o jerárquicos.
4. Para ocultar la información de las variables, seleccione la opción "Ocultar" en la lista desplegable "Información de la variable".
   La información de las variables se muestra u oculta para todos los bloques abiertos. Si abre más bloques, la información de las variables se mostrará u ocultará dependiendo del ajuste seleccionado.

Mostrar u ocultar información de variables para un bloque abierto

Para mostrar u ocultar la información de variables para un bloque abierto, proceda del siguiente modo:

1. Para visualizar la información de las variables, seleccione las opciones "Mostrar información de la variable" o "Información de la variable con jerarquía" en la lista desplegable "Muestra información sobre la variable", en función de si desea visualizar los comentarios sencillos o jerárquicos.
2. Para ocultar la información de las variables, seleccione la opción "Ocultar información de la variable" en la lista desplegable "Muestra información sobre la variable".
   Se muestra u oculta la información sobre las variables.

Utilizar Favoritos en SCL

Agregar instrucciones SCL a los Favoritos

Requisitos

- Hay un bloque abierto.
- La Task Card "Instrucciones" tiene configurado el modo multipaleta, o bien los Favoritos se muestran también en el editor.

Procedimiento

Para agregar instrucciones a los Favoritos, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Maximice la paleta "Instrucciones básicas".
3. En la paleta "Instrucciones básicas", navegue hasta la instrucción que desea agregar a los Favoritos.
4. Arrastre la instrucción hasta la paleta "Favoritos" o hasta el área de Favoritos del editor de programación mediante Drag & Drop.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Sinopsis del editor de programación (Página 2619)

Inserción de instrucciones de SCL mediante Favoritos (Página 2811)

Eliminación de instrucciones SCL de los Favoritos (Página 2812)

Inserción de instrucciones de SCL mediante Favoritos

Requisitos

- Hay un bloque abierto.
- Existen Favoritos.

Procedimiento

Para insertar una instrucción en un programa desde los Favoritos, proceda del siguiente modo:

1. Arrastre la instrucción deseada mediante Drag & Drop desde los Favoritos hasta la posición deseada.

O bien:

1. Marque el punto del programa en el que quiere insertar la instrucción.
2. En los Favoritos, haga clic en la instrucción que desea insertar.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Sinopsis del editor de programación (Página 2619)

Agregar instrucciones SCL a los Favoritos (Página 2810)

Eliminación de instrucciones SCL de los Favoritos (Página 2812)

Eliminación de instrucciones SCL de los Favoritos

Requisitos

Hay un bloque lógico abierto.

Procedimiento

Para eliminar instrucciones de los Favoritos, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la instrucción que desea eliminar.
2. Elija el comando "Eliminar instrucción" del menú contextual.

Nota

Para mostrar los Favoritos en el editor de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Sinopsis del editor de programación (Página 2619)

Agregar instrucciones SCL a los Favoritos (Página 2810)

Inserción de instrucciones de SCL mediante Favoritos (Página 2811)

Inserción de llamadas de bloque en SCL

Principios básicos de la llamada de bloque en SCL

Llamada de bloques de función

Sintaxis de una llamada

Para llamar un bloque de función como instancia individual o multiinstancia se utiliza la siguiente sintaxis:

- Instancia individual:
  - Si el bloque de función procede del proyecto:
    `<NombreDB> (lista de parámetros)`
  - Si el bloque de función procede de la Task Card "Instrucciones":
    `<NombreDB>.<NombreInstrucción> (lista de parámetros)`
- Multiinstancia
  `<#NombreInstancia> (lista de parámetros)`

Llamada como instancia individual o multiinstancia

Los bloques de función se pueden llamar como instancia individual o como multiinstancia.

- Llamada como instancia individual
  El bloque de función llamado almacena sus datos en un bloque de datos propio.
- Llamada como multiinstancia
  El bloque de función llamado almacena sus datos en el bloque de datos de instancia del bloque de función que lo llama.

Encontrará más información sobre los tipos de llamada en "Consulte también".

Lista de parámetros

Si llama otro bloque lógico desde un bloque SCL, deberá indicar los parámetros actuales de los parámetros formales del bloque llamado.

La entrada de los parámetros es análoga a una asignación de valor. Mediante esta asignación de valor, asigna un valor (parámetro actual) a los parámetros que ha definido en el bloque llamado.

Los parámetros formales del bloque lógico llamado se listan entre paréntesis directamente después de la llamada. Los parámetros de entrada y de entrada/salida tienen el identificador de asignación ":=", mientras que los parámetros de salida tienen el identificador de asignación "=>". Un comodín detrás del parámetro indica el tipo de datos necesario y el tipo de parámetro.

Reglas sobre la transferencia de valores de parámetros

Se aplican las siguientes reglas a la transferencia de valores de parámetros:

- Como parámetros actuales se pueden usar constantes, variables y expresiones.
- Se puede elegir cualquier orden para las asignaciones.
- El tipo de datos de los parámetros formales y actuales debe coincidir.
- Las asignaciones individuales se han de separar mediante una coma.
- Si el bloque llamado sólo tiene un parámetro, es suficiente con indicar el parámetro actual entre paréntesis. No es necesario indicar el parámetro formal.

Consulte también

Inserción manual de llamadas de bloque (Página 2818)

Insertar llamadas de bloque mediante Drag & Drop (Página 2819)

Ejemplos de la llamada de un bloque de función en SCL (Página 2816)

Llamadas de bloque (Página 2263)

Utilizar y direccionar operandos (Página 2280)

Llamada de funciones

Sintaxis de una llamada

Para llamar una función se utiliza la siguiente sintaxis:

```
<nombre de la función> (lista de parámetros);     //Llamada estándar
<operando>:=<nombre de la función> (lista de parámetros); // Llamada
en una expresión
```

Valor de función

Las funciones que devuelven un valor de retorno se pueden utilizar en SCL en cualquier expresión en lugar de un operando. Por eso en SCL el valor de retorno también se denomina "valor de función".

Las posibilidades de llamada de funciones en SCL dependen de si la función devuelve un valor de función al bloque que efectúa la llamada.

El valor de función se define en el parámetro RET_VAL. Si el parámetro RET_VAL es del tipo de datos VOID, la función no devuelve ningún valor al bloque que efectúa la llamada. Si el parámetro RET_VAL es de otro tipo de datos, la función devuelve un valor de función de este tipo de datos.

Para el parámetro RET_VAL se admiten en SCL todos los tipos de datos excepto ANY, ARRAY, STRUCT y VARIANT, así como los tipos de parámetros TIMER y COUNTER.

Posibilidades de llamada

Existen dos llamadas de función posibles en SCL:

- Llamada estándar de funciones con y sin valor de función
  En una llamada estándar los resultados de la función están disponibles como parámetros de salida y de entrada/salida.
- Llamada en una expresión de funciones con valor de función
  Las funciones que devuelven un valor de función se pueden utilizar en cualquier expresión en lugar de utilizar un operando (p. ej. en una asignación de valor).
  La función calcula el valor de función, que tiene el mismo nombre de la función, y lo transmite al bloque que efectúa la llamada. Allí el valor sustituye a la llamada de función. Después de la llamada, los resultados de la función están disponibles como valor de función o como parámetro de salida y de entrada/salida.

Lista de parámetros

Si llama otro bloque lógico desde un bloque SCL, tiene que indicar los parámetros actuales de los parámetros formales del bloque llamado.

La entrada de los parámetros es análoga a una asignación de valor. Mediante esta asignación de valor, asigna un valor (parámetro actual) a los parámetros que ha definido en el bloque llamado.

Los parámetros formales del bloque lógico llamado se listan entre paréntesis directamente después de la llamada. Los parámetros de entrada y de entrada/salida tienen el identificador de asignación ":=", mientras que los parámetros de salida tienen el identificador de asignación "=>". Un comodín sobre fondo gris detrás del parámetro indica el tipo de datos necesario y el tipo de parámetro.

Reglas sobre la transferencia de valores de parámetros

Se aplican las siguientes reglas para la transferencia de valores a los parámetros de funciones:

- Todos los parámetros de la función deben recibir valores.
- Se puede elegir cualquier orden para las asignaciones.
- Como parámetros actuales se pueden usar constantes, variables y expresiones.
- El tipo de datos de los parámetros formales y actuales debe coincidir.
- Las asignaciones individuales se han de separar mediante una coma.
- Si el bloque llamado solo tiene un parámetro, es suficiente con indicar el parámetro actual entre paréntesis. No es necesario indicar el parámetro formal.
- Cuando se llaman funciones en SCL, no se puede utilizar el mecanismo de habilitación mediante EN. En su lugar, utilice una instrucción IF para la llamada condicional de funciones.

Consulte también

Inserción manual de llamadas de bloque (Página 2818)

Insertar llamadas de bloque mediante Drag & Drop (Página 2819)

Ejemplos de la llamada de funciones en SCL (Página 2817)

Llamadas de bloque (Página 2263)

Utilizar y direccionar operandos (Página 2280)

Ejemplos de la llamada de un bloque de función en SCL

Llamada como instancia individual

El siguiente ejemplo muestra la llamada de un FB como instancia individual:

```
SCL
// Llamada como instancia individual
"MyDB" (MyInput:=10, MyInout:= "Tag1");
```

Resultado

Después de ejecutar la llamada, en el bloque de datos "MyDB" queda disponible el valor determinado en "Tag1" para el parámetro de entrada/salida "MyInout".

Llamada como multiinstancia

En el siguiente ejemplo se muestra la llamada de un FB como multiinstancia:

```
SCL
// Llamada como multiinstancia
"MyFB" (MyInput:= 10, MyInout:= "Tag1");
```

Resultado

Después de ejecutar el bloque "MyFB" queda disponible el valor determinado para el parámetro de entrada/salida "MyInout" en "Tag1" en el bloque de datos del bloque lógico que efectúa la llamada.

Consulte también

Llamada de bloques de función (Página 2813)

Llamadas de bloque (Página 2263)

Utilizar y direccionar operandos (Página 2280)

Inserción manual de llamadas de bloque (Página 2818)

Insertar llamadas de bloque mediante Drag & Drop (Página 2819)

Ejemplos de la llamada de funciones en SCL

Llamada estándar

El siguiente ejemplo muestra una llamada de función estándar:

SCL

```
// Llamada de función estándar
"MyFC" (MyInput := 10, MyInOut := "Tag1");
```

Resultado

Después de ejecutar el bloque "MyFC", queda disponible el valor determinado en "Tag1" para el parámetro de entrada/salida "MyInOut" en el bloque invocante, donde debe procesarse.

Llamada en una asignación de valor

El siguiente ejemplo muestra una llamada de función en una asignación de valor:

SCL

```
(*Llamada en una asignación de valor; se ha definido un valor de función
para "MyFC" *)
#MyOperand := "MyFC" (MyInput1 := 3, MyInput2 := 2, MyInput3 := 8.9,
MyInOut := "Tag1");
```

Resultado

El valor de función de "MyFC" se transmite a "#MyOperand".

Llamada en una expresión aritmética

El siguiente ejemplo muestra una llamada de función en una expresión aritmética:

SCL

```
(*Llamada en una expresión aritmética; se ha definido un valor de función
para "MyFC" *)
#MyOperand := "Tag2" + "MyFC" (MyInput1 := 3, MyInput2 := 2, MyInput3 :=
8.9);
```

Resultado

El valor de función de "MyFC" se suma a "Tag2" y el resultado se transmite a "MyOperand".

Consulte también

Llamada de funciones (Página 2814)

Inserción manual de llamadas de bloque (Página 2818)

Insertar llamadas de bloque mediante Drag & Drop (Página 2819)

Inserción manual de llamadas de bloque

Es posible insertar llamadas de funciones (FC) y de bloques de función (FB).

Inserción de una llamada de una función (FC)

Para insertar una llamada de función, proceda del siguiente modo:

1. Introduzca el nombre de la función.
2. Confirme las entradas realizadas pulsando la tecla Intro.
   La sintaxis de la llamada de función, junto con la lista de parámetros, se inserta en el programa SCL. Los comodines de los parámetros actuales se resaltan en color amarillo. Se selecciona el primer comodín.
3. Sustituya el comodín por un parámetro actual. También puede arrastrar una variable de la interfaz o la tabla de variables PLC mediante Drag & Drop hasta el comodín.
4. Navegue con la tecla <TAB> por los demás comodines y sustitúyalos por parámetros actuales.

Insertar una llamada de un bloque de función (FB)

Para insertar una llamada de un bloque de función (FB), proceda del siguiente modo:

1. Introduzca el nombre del bloque de función.
2. Confirme las entradas realizadas pulsando la tecla Intro.
   Se abre el cuadro de diálogo "Opciones de llamada".
3. Indique en el cuadro de diálogo si quiere llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca en el campo de entrada "Nombre" el nombre del bloque de datos que se debe asignar a la llamada.
   - Si hace clic en el botón "Multiinstancia", introduzca en el campo de entrada "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
4. Confirme las entradas con "Aceptar".
   La sintaxis de la llamada del bloque de función, junto con la lista de parámetros, se inserta en el programa SCL. Los comodines de los parámetros actuales se resaltan en color amarillo. Se selecciona el primer comodín.
5. Sustituya el comodín por un parámetro actual. También puede arrastrar una variable de la interfaz o la tabla de variables PLC mediante Drag & Drop hasta el comodín.
6. Navegue con la tecla <TAB> por los demás comodines y sustitúyalos por parámetros actuales.

Resultado

Se inserta la llamada de bloque.

Si al llamar un bloque de función especifica un bloque de datos instancia que no exista aún, éste se creará.

Consulte también

Actualización de llamadas de bloque (Página 2820)

Ampliar y reducir una lista de parámetros (Página 2821)

Principios básicos de la llamada de bloque en SCL (Página 2813)

Llamadas de bloque (Página 2263)

Utilizar la función automática de autocompletar en los lenguajes de programación textuales (Página 2639)

Insertar llamadas de bloque mediante Drag & Drop

Las llamadas de funciones (FC) y bloques de función (FB) existentes pueden insertarse desde el árbol del proyecto mediante Drag & Drop.

Requisitos

La función (FC) o el bloque de función (FB) que se va a llamar está disponible.

Inserción de una llamada de una función (FC)

Para insertar una llamada de función mediante Drag & Drop, proceda del siguiente modo:

1. Arrastre la función desde el árbol del proyecto hasta el programa.
   La sintaxis de la llamada de función, junto con la lista de parámetros, se inserta en el programa SCL. Los comodines de los parámetros actuales se resaltan en color amarillo. Se selecciona el primer comodín.
2. Sustituya el comodín por un parámetro actual. También puede arrastrar una variable de la interfaz o la tabla de variables PLC mediante Drag & Drop hasta el comodín.
3. Navegue con la tecla <TAB> por los demás comodines y sustitúyalos por parámetros actuales.

Insertar una llamada de un bloque de función (FB)

Para insertar una llamada de un bloque de función (FB) mediante Drag & Drop, proceda del siguiente modo:

1. Arrastre el bloque de función desde el árbol del proyecto hasta el programa.
   Se abre el cuadro de diálogo "Opciones de llamada".
2. Indique en el cuadro de diálogo si quiere llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca en el campo de entrada "Nombre" el nombre del bloque de datos que se debe asignar a la llamada.
   - Si hace clic en el botón "Multiinstancia", introduzca en el campo de entrada "Nombre de la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
3. Confirme las entradas con "Aceptar".
   La sintaxis de la llamada del bloque de función, junto con la lista de parámetros, se inserta en el programa SCL. Los comodines de los parámetros actuales se resaltan en color amarillo. Se selecciona el primer comodín.
4. Sustituya el comodín por un parámetro actual. También puede arrastrar una variable de la interfaz o la tabla de variables PLC mediante Drag & Drop hasta el comodín.
5. Navegue con la tecla <TAB> por los demás comodines y sustitúyalos por parámetros actuales.

Resultado

Se inserta la llamada de bloque.

Si al llamar un bloque de función especifica un bloque de datos instancia que no exista aún, éste se creará.

Consulte también

Actualización de llamadas de bloque (Página 2820)

Ampliar y reducir una lista de parámetros (Página 2821)

Principios básicos de la llamada de bloque en SCL (Página 2813)

Llamadas de bloque (Página 2263)

Utilizar la función automática de autocompletar en los lenguajes de programación textuales (Página 2639)

Actualización de llamadas de bloque

Si cambian los parámetros de interfaz de un bloque llamado, la llamada no podrá ejecutarse correctamente. Actualice las llamadas de bloque con objeto de evitar llamadas incoherentes.

Existen dos posibilidades de actualizar llamadas de bloque, a saber:

- Actualización explícita en el editor de programación.
  Se muestran las incoherencias en el bloque abierto y se pueden actualizar.
- Actualización implícita durante la compilación.
  Se actualizan todas las llamadas de bloque en el programa, así como los tipos de datos PLC utilizados.

Actualizar llamadas de bloque en el editor de programación

Para actualizar una llamada de bloque en un bloque, proceda del siguiente modo:

1. Abra el bloque en el editor de programación.
2. Haga clic en el botón "Actualizar llamadas de bloque incoherentes" de la barra de herramientas.
   Se muestran las llamadas incoherentes.
3. Corrija las incoherencias.

Actualizar llamadas de bloque durante la compilación

Para actualizar implícitamente todas las llamadas de bloque y usos de tipos de datos PLC durante la compilación, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Inserción manual de llamadas de bloque (Página 2818)

Insertar llamadas de bloque mediante Drag & Drop (Página 2819)

Ampliar y reducir una lista de parámetros

Si en SCL se llaman bloques o se insertan instrucciones que son bloques de función internos del sistema, la sintaxis se inserta en el programa SCL, incluida la lista de parámetros con los comodines para los parámetros actuales. Para una mejor claridad del código de programa, de la lista de parámetros se retiran los parámetros opcionales no cableados al ejecutar otras instrucciones. Sin embargo, se pueden volver a incluir en cualquier momento. Del mismo modo, también es posible reducir explícitamente la lista de parámetros una vez finalizado el cableado de los parámetros.

Ampliación de la lista de parámetros

Para ampliar la lista de parámetros, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la llamada de bloque o la instrucción.
2. En el menú contextual, elija el comando "Ampliar lista de parámetros" o pulse la combinación de teclas <Ctrl+Mayús+Barra espaciadora>.
   La lista de parámetros vuelve a mostrarse completa.

Reducción de la lista de parámetros

Para reducir la lista de parámetros, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la llamada de bloque o la instrucción.
2. En el menú contextual, elija el comando "Reducir lista de parámetros" o pulse la combinación de teclas <Ctrl+Mayús+Barra espaciadora>.
   Todos los parámetros opcionales no cableados se ocultan.

Consulte también

Entrada manual de instrucciones SCL (Página 2805)

Inserción de instrucciones SCL desde la Task Card "Instrucciones" (Página 2806)

Inserción manual de llamadas de bloque (Página 2818)

Insertar llamadas de bloque mediante Drag & Drop (Página 2819)

Inserción de comentarios

Comentar el código de programa

Existen diferentes posibilidades para comentar programas SCL:

- Comentario de línea
  Los comentarios de línea comienzan con "//" y se extienden hasta el final de la línea.
- Sección de comentario
  Una sección de comentario se inicia con "(*" y se cierra con "*)". Esta sección puede abarcar varias líneas.

Insertar comentarios de líneas

Para insertar un comentario de línea, proceda del siguiente modo:

1. Introduzca "//" en el punto en el que quiere poner el comentario. No tiene que ser el principio de la línea.
2. Introduzca el texto del comentario.

INsertar sección de comentario

Para insertar una sección de comentario, proceda del siguiente modo:

1. Introduzca "(*" en el punto en el que quiere poner el comentario. No tiene que ser el principio de la línea.
2. Introduzca el texto del comentario.
3. Cierre el comentario con "*)".

Desactivar una o varias líneas mediante comentarios

Para desactivar código del programa mediante comentarios, proceda del siguiente modo:

1. Seleccione las líneas de código que desea comentar.
2. Haga clic en el botón "Marcar la selección como comentario" del editor.
   En las líneas marcadas se inserta "//" al principio de la línea. El código siguiente se interpreta como comentario. Si se desactivan líneas que ya tienen un comentario de línea, también se inserta "//". Si a continuación se vuelven a activar las líneas, se mantendrán los comentarios iniciales.

Activar líneas de comentario

Para volver a activar líneas de comentario como código, proceda del siguiente modo:

1. Seleccione las líneas de código que desea activar.
2. Haga clic en el botón "Eliminar comentario" del editor.
   Se elimina el identificador "//" de los comentarios de líneas al principio de la línea.

Ejemplo

El siguiente código contiene secciones de comentario y comentarios de líneas

```
(*****************************************************************************
  Aquí se incluye una descripción de las siguientes instrucciones
*****************************************************************************)
IF "MyVal1" > 0 THEN //ninguna división entre 0
    "MyReal" := "MyVal2" (* valor de entrada *) / "MyVal1" (* valor medido *);
END_IF;
//Conversión de tipos de datos
"MyInt" := REAL_TO_INT("MyReal");
```

Edición de instrucciones SCL

Selección de instrucciones

Es posible seleccionar instrucciones individuales o todas las instrucciones de un bloque.

Requisitos

Hay un bloque SCL abierto.

Selección de instrucciones individuales

Para seleccionar instrucciones individuales, proceda del siguiente modo:

1. Inserte la marca de inserción delante del primer carácter que quiere marcar.
2. Pulse y mantenga oprimido el botón izquierdo del ratón.

3. Desplace el puntero del ratón detrás del último carácter que quiere marcar.
4. Suelte el botón izquierdo del ratón.

Selección de todas las instrucciones de un programa

Para seleccionar todas las instrucciones, proceda del siguiente modo:

1. Seleccione en el menú "Edición" el comando "Seleccionar todo" o use la combinación de teclas <Ctrl+A>.

Nota

Si marca la palabra clave de apertura de una instrucción, se marca automáticamente la palabra clave de cierre.

Copiar, cortar y pegar instrucciones

Copiar instrucciones

Para copiar una instrucción, proceda del siguiente modo:

1. Marque la instrucción que quiere copiar.
2. Elija el comando "Copiar" del menú contextual.

Cortar instrucciones

Para cortar una instrucción, proceda del siguiente modo:

1. Marque la instrucción que quiere cortar.
2. Seleccione el comando "Cortar" del menú contextual.

Insertar instrucciones desde el portapapeles

Para insertar una instrucción desde el portapapeles, proceda del siguiente modo:

1. Copie o corte una instrucción.
2. Haga clic en la posición en la que desea insertar la instrucción.
3. Elija el comando "Pegar" del menú contextual.

Borrado de instrucciones

Requisitos

Hay un bloque SCL abierto.

Procedimiento

Para borrar una instrucción, proceda del siguiente modo:

1. Marque la instrucción que quiere borrar.
2. Elija el comando "Borrar" del menú contextual.

Crear programas GRAPH

Principios básicos de GRAPH

Lenguaje de programación GRAPH

Resumen

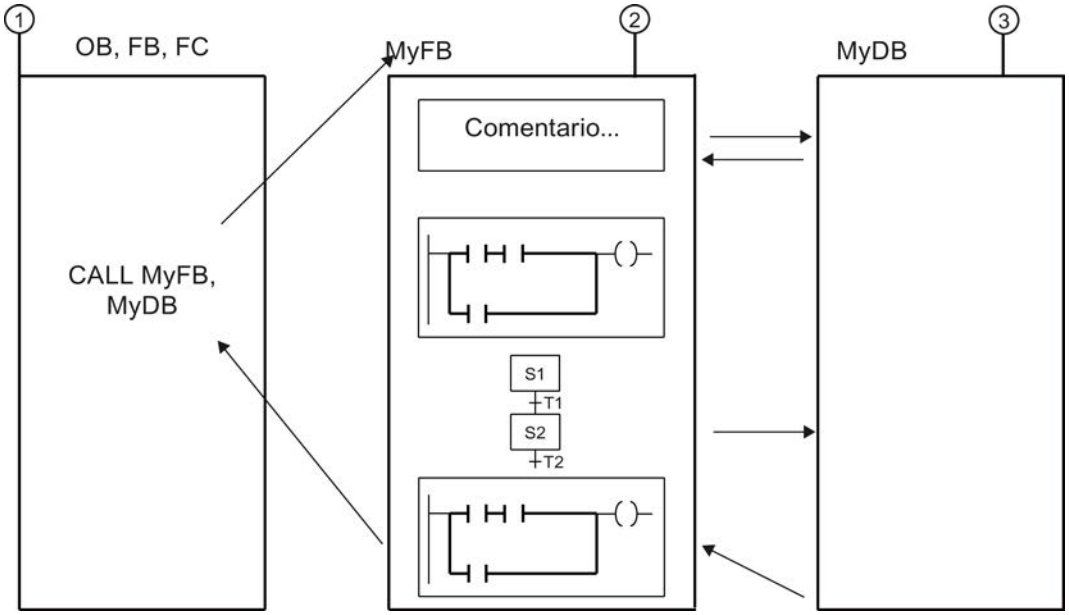
GRAPH es un lenguaje de programación gráfico para crear controles secuenciales. Permite programar controles secuenciales de manera clara y rápida utilizando cadenas secuenciales. El proceso se descompone en etapas individuales con un alcance funcional delimitable y se organiza en las cadenas secuenciales. En las etapas individuales se determinan las acciones que se deben ejecutar. El paso entre las etapas lo forman las transiciones. Estas contienen condiciones para pasar a la etapa siguiente.

Bloques del control secuencial

Un control secuencial controla el proceso en un orden predeterminado y en función de ciertas condiciones. La complejidad del control secuencial depende de la tarea de automatización. Un control secuencial está formado siempre por tres bloques como mínimo:

- Bloque de función GRAPH
  En el bloque de función GRAPH se describen las diferentes etapas y transiciones del control secuencial dentro de una o varias cadenas secuenciales.
- Bloque de datos de instancia
  El bloque de datos de instancia contiene los datos y parámetros del control secuencial. Está asignado al bloque de función GRAPH y puede ser generado por el sistema automáticamente.
- Bloque lógico que efectúa la llamada
  Para que el bloque de función GRAPH pueda ejecutarse dentro del ciclo, debe ser llamado por un bloque lógico de nivel superior. Dicho bloque puede ser un bloque de organización (OB), una función (FC) u otro bloque de función (FB). El bloque de función GRAPH se llama siempre como instancia individual.

La figura siguiente muestra la colaboración entre los bloques en un control secuencial:



① Bloque que efectúa la llamada

② Bloque de función GRAPH

③ Bloque de datos de instancia

El ciclo de un bloque de función GRAPH está definido por el ciclo del bloque que efectúa la llamada. En cada ciclo del bloque de función GRAPH se ejecutan en primer lugar las instrucciones permanentes anteriores. Seguidamente se procesan las acciones de las etapas activas. Por último se ejecutan las instrucciones permanentes posteriores.

Nota

Las instrucciones permanentes se ejecutan en cada ciclo, aunque no haya ninguna etapa activa.

Consulte también

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Cadenas secuenciales

Principios básicos de las cadenas secuenciales

Introducción

Dentro de un bloque de función GRAPH es posible programar el programa en forma de cadenas secuenciales. Las cadenas secuenciales pueden procesar tareas independientes, aunque también es posible descomponer una tarea compleja en varias cadenas secuenciales. Si las cadenas secuenciales procesan tareas independientes se procesarán paralelamente en la secuencia del programa. Si se utilizan varias cadenas secuenciales para descomponer una tarea compleja en segmentos más pequeños, habrá que enlazar las cadenas secuenciales relacionadas con saltos.

En una cadena secuencial, el programa se divide en etapas individuales. En casos sencillos, las etapas se procesan linealmente una tras otra. No obstante, el uso de ramas alternativas o simultáneas permite generar también cadenas secuenciales con estructuras más complejas. El procesamiento del programa empieza siempre con una etapa definida como etapa inicial. Una cadena secuencial puede tener una o varias etapas iniciales. Las etapas iniciales pueden ocupar cualquier posición dentro de la cadena secuencial.

Cuando una etapa está activa se ejecutan sus acciones. También existe la posibilidad de que varias etapas estén activas al mismo tiempo, p. ej. dentro de una rama simultánea. Mientras una etapa está activa se comprueban continuamente las condiciones de la transición siguiente. En cuanto se cumplen todas las condiciones y no hay ningún error de supervisión pendiente, la transición pasa a la etapa siguiente. Ahora, esta es la etapa activa. Una cadena secuencial puede terminarse con un salto o con un fin de cadena. Con un salto se pasa a cualquier otra etapa de la misma cadena secuencial o de otra distinta. Esto permite procesar cíclicamente la cadena secuencial.

Principio de procesamiento de una cadena secuencial

El procesamiento de una cadena secuencial empieza activando la o las etapas iniciales de la cadena. Solo se admiten varias etapas iniciales en ramas simultáneas. Mientras una etapa está activa se ejecutan sus acciones. Para ello se tienen en cuenta las condiciones de interlock para las diferentes acciones. Una vez que se han ejecutado todas las acciones posibles, se comprueba ante todo si hay algún error de supervisión pendiente. En caso negativo y si además se cumple la transición siguiente, se activa la próxima etapa de la cadena. En caso de haber un error de supervisión o de que no se cumpla la transición, la etapa en cuestión se mantiene activa hasta que se haya solucionado el error o se cumpla la transición, según corresponda. Al final de una cadena secuencial existe la posibilidad de procesar cíclicamente la cadena secuencial mediante saltos o bien de finalizarla con un fin de cadena.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Programar la estructura de una cadena secuencial (Página 2883)

Lenguaje de programación GRAPH (Página 2825)

Operandos específicos de GRAPH (Página 2834)

Instrucciones permanentes (Página 2846)
Condiciones (Página 2847)
Avisos de interlock y supervisión (Página 2847)
Etapas y transiciones (Página 2831)
Acciones (Página 2836)
Interfaz de un bloque de función GRAPH (Página 2848)

Elementos de una cadena secuencial

Resumen

La tabla siguiente muestra los posibles elementos de una cadena secuencial y sus símbolos:

Símbolo	Nombre
	Etapa y transición (Página 2831)
	Etapa (Página 2831)
	Transición (Página 2831)
	Rama simultánea
	Rama alternativa
	Cerrar rama
	Salto a etapa
	Fin de cadena

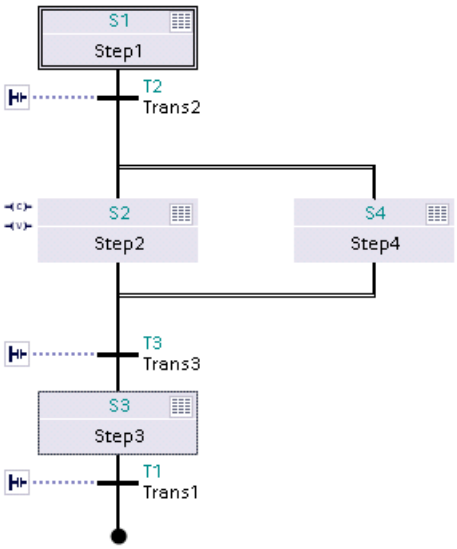
Rama simultánea

Las ramas simultáneas permiten programar ramas Y. Esto significa que con una transición se activan varias etapas cuyas acciones se ejecutarán a continuación. Por ese motivo, las ramas simultáneas empiezan siempre con una etapa.

Las transiciones posteriores de las ramas simultáneas se encuentran en la rama principal, de modo que las diferentes ramas simultáneas pueden cerrarse en diferentes puntos de la rama principal. Tenga en cuenta que las ramas que se agrupan en una transición no se conmutarán hasta la etapa siguiente, cuando se hayan ejecutado por completo todas las ramas.

Es posible programar un máximo de 249 ramas simultáneas en una cadena secuencial.

La figura siguiente muestra la representación gráfica de una rama simultánea:

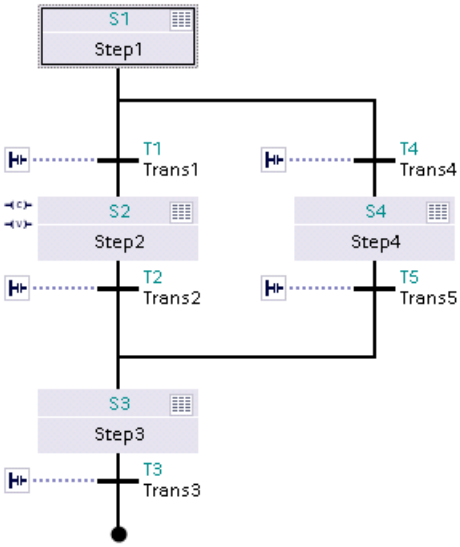


Rama alternativa

Las ramas alternativas permiten programar ramas O. Esto significa que, después de una etapa, se insertan ramas que empiezan con una transición. En función de la transición que se cumpla primero se ejecutará la rama correspondiente. Si se cumplen varias transiciones al mismo tiempo, tendrá prioridad la transición que esté más a la izquierda y se ejecutará la rama correspondiente.

Es posible programar un máximo de 125 ramas alternativas en una cadena secuencial.

La figura siguiente muestra la representación gráfica de una rama alternativa:



Cerrar rama

El elemento "Cerrar rama" permite cerrar ramas simultáneas o alternativas y llevarlas a su rama matriz. Esto debe hacerse cuando la rama no se cierra con un salto o un fin de cadena.

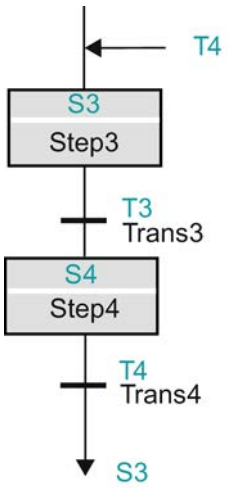
Salto a etapa

Con un salto se prosigue la ejecución del programa en una etapa cualquiera dentro del bloque de función GRAPH. Los saltos pueden insertarse al final de la rama principal, de una rama alternativa o de una rama simultánea, de modo que sea posible procesar la cadena secuencial cíclicamente. El salto y su destino se representan en la cadena secuencial en forma de flecha, de modo que en el destino del salto se indica la transición de origen y en el salto la etapa de destino.

Nota

Se deben evitar saltos desde una transición hasta la etapa inmediatamente anterior. Si, a pesar de todo, es necesario un salto de este tipo, inserte una etapa vacía con una transición sin condiciones.

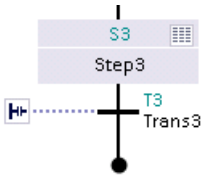
La figura siguiente muestra la representación gráfica de un salto:



Fin de cadena

Las cadenas secuenciales o ramas pueden cerrarse con el elemento "fin de cadena". Con él finaliza la cadena secuencial o rama. En una rama simultánea, sin embargo, el fin de cadena debe estar precedido de una transición. El elemento "fin de cadena" se representa en la cadena secuencial con un punto negro.

La figura siguiente muestra la representación gráfica de un fin de cadena:



Nota

Si todas las ramas de una cadena secuencial se cierran con un fin de cadena, es posible volver a iniciar la cadena secuencial con el parámetro "INIT_SQ" o con el botón "Inicializar" de la Task Card "Test" en la paleta "Control secuencial".

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)

Programar la estructura de una cadena secuencial (Página 2883)

Lenguaje de programación GRAPH (Página 2825)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Etapas y transiciones

Principios básicos de las etapas

Introducción

Las etapas sirven para dividir las tareas de automatización complejas en tareas más pequeñas y claras que, seguidamente, se llevan a cabo por medio de acciones. Las diferentes etapas se organizan en cadenas secuenciales, de modo que al ejecutar el programa cada etapa se procesa en el orden especificado. Cada etapa debe tener un nombre y un número unívocos.

Las acciones de una etapa se ejecutan cuando la etapa se activa. Para ello es necesario que una de las etapas sea la inicial o bien que se cumpla la transición precedente. Una etapa también puede llamarse con una acción controlada por evento. Adicionalmente es posible programar condiciones de interlock y supervisión en una etapa.

La etapa con la que debe empezar el procesamiento de la cadena secuencial se define libremente. Para ello se especifica una etapa cualquiera como etapa inicial. En las ramas simultáneas es posible definir varias etapas como iniciales. En caso de operación cíclica de

una cadena secuencial, la etapa inicial recibe el mismo tratamiento que cualquier otra etapa mientras no se efectúe una inicialización de la cadena secuencial con el parámetro "INIT_SQ". Cuando se reinicializa la cadena secuencial se activan todas las etapas iniciales, siempre que haya un flanco de señal ascendente en "INIT_SQ". Todas las demás etapas quedan desactivadas. En la navegación y la vista de cadena, las etapas iniciales están marcadas con un marco para identificarlas fácilmente.

Una etapa en la que no se programan acciones es una etapa vacía. Puesto que una etapa vacía no tiene que procesar acciones, la cadena secuencial pasa inmediatamente a la transición siguiente. Esto permite programar dos transiciones consecutivas.

Los nombres y números de las etapas pueden modificarse en cualquier momento. Además, existe la posibilidad de documentar etapas utilizando comentarios.

Consulte también

Elementos de una etapa (Página 2832)

Principios básicos de las transiciones (Página 2833)

Programar la estructura de una cadena secuencial (Página 2883)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Elementos de una etapa

En una etapa es posible programar los componentes siguientes:

- Interlock
- Supervisión
- Acciones

Interlock

Un interlock permite programar condiciones de las que depende la ejecución de las diferentes acciones. Las acciones de una etapa enlazadas con el interlock solo se ejecutan si se cumple la condición del interlock. Si la condición del interlock no se cumple, hay un fallo. Para estos casos es posible definir avisos que se mostrarán cuando corresponda. Sin embargo, el fallo no influye en la transición a la etapa siguiente. Tenga en cuenta además que al desactivar la etapa se cancela automáticamente el interlock. También existe la opción de que el operador tenga que acusar un aviso de interlock para cerrar el aviso.

Además, es posible no programar condiciones en el interlock pero enlazar el interlock con acciones de todos modos. En ese caso, el interlock siempre se cumple.

Supervisión

Una supervisión permite programar condiciones con las que se supervisa una etapa completa. Si se cumple la condición significa que hay un fallo y la etapa permanece activa. En ese caso, la cadena secuencial no pasa a la etapa siguiente, sino que se queda en la etapa activa hasta que se ha eliminado el fallo. También en la supervisión es posible definir avisos y activar la obligación de acuse.

Acciones

En las etapas se programan acciones que controlan entradas o salidas, activan y desactivan otras etapas de la cadena secuencial o llaman bloques, por ejemplo. Por consiguiente, las acciones incluyen instrucciones para controlar el proceso.

Se distingue entre acciones estándar y acciones controladas por evento. Las acciones estándar se ejecutan mientras la etapa está activa. Las acciones controladas por evento son las que se han enlazado con un evento.

Consulte también

Principios básicos de las etapas (Página 2831)

Principios básicos de las transiciones (Página 2833)

Lenguaje de programación GRAPH (Página 2825)

Operandos específicos de GRAPH (Página 2834)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Programar la estructura de una cadena secuencial (Página 2883)

Cadenas secuenciales (Página 2827)

Acciones (Página 2836)

Interfaz de un bloque de función GRAPH (Página 2848)

Principios básicos de las transiciones

Introducción

Las transiciones están entre las etapas y contienen las condiciones para continuar con la siguiente etapa. Es decir, la cadena secuencial solo pasa a las etapas siguientes cuando se cumplen las condiciones de la transición. Durante el proceso se desactivan las etapas correspondientes a la transición y se activan las etapas posteriores. Si no se cumplen las condiciones, las etapas correspondientes a la transición se mantienen activas. Cada transición

debe tener un nombre y un número unívocos. Las condiciones de una transición se programan en KOP o FUP.

Una transición en la que no se programan condiciones es una transición vacía. En ese caso, la cadena secuencial pasa inmediatamente a las etapas siguientes.

Los nombres y números de las transiciones pueden modificarse.

Consulte también

Programar la estructura de una cadena secuencial (Página 2883)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Elementos de una etapa (Página 2832)

Operandos específicos de GRAPH

Introducción

En GRAPH están disponibles operandos predefinidos que permiten leer determinada información y procesarla posteriormente. Estos operandos se utilizan en transiciones, interlocks, supervisiones, acciones e instrucciones permanentes.

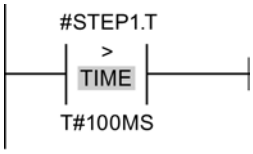
La tabla siguiente muestra una relación de los operandos específicos de GRAPH:

Operando	Significado	Utilización
<Nombre de etapa>.T	Tiempo de activación actual o último de la etapa indicada	Comparador, asignación
<Nombre de etapa>.U	Tiempo total de activación de la etapa indicada sin contar el tiempo de avería	Comparador, asignación
<Nombre de etapa>.X	Indica si la etapa indicada está activa	Contacto normalmente abierto/ contacto normalmente cerrado
<Nombre de transición>.TT	Indica si todas las condiciones de la transición indicada se cumplen	Contacto normalmente abierto/ contacto normalmente cerrado

Ejemplo de utilización de la instrucción "Mayor" con el operando "<Nombre de etapa>.T"

Para vigilar la duración de un proceso existe la posibilidad de programar una condición de supervisión utilizando la instrucción "Mayor" y el operando "<Nombre de etapa>.T". Con ello se vigila la duración total de una etapa.

La figura siguiente muestra el funcionamiento de la instrucción:

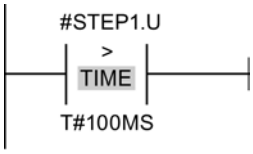


Mientras el tiempo de activación de "Step1" esté por debajo de 100 ms, el resultado lógico (RLO) será "0". En cuanto el tiempo de activación sobrepasa los 100 ms, el RLO cambia a "1".

Ejemplo de utilización de la instrucción "Mayor" con el operando "<Nombre de etapa>.U"

Si desea determinar el tiempo de activación sin contar los fallos, utilice el operando "<Nombre de etapa>.U". Con ello se restan del tiempo de activación los momentos en los que la etapa ha estado perturbada.

La figura siguiente muestra el funcionamiento de la instrucción:



Mientras el tiempo de activación de "Step1", sin tener en cuenta las posibles averías, esté por debajo de 100 ms, el RLO será "0". En cuanto el tiempo de activación sobrepasa los 100 ms, el RLO cambia a "1".

Consulte también

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Acciones

Principios básicos de las acciones

Introducción

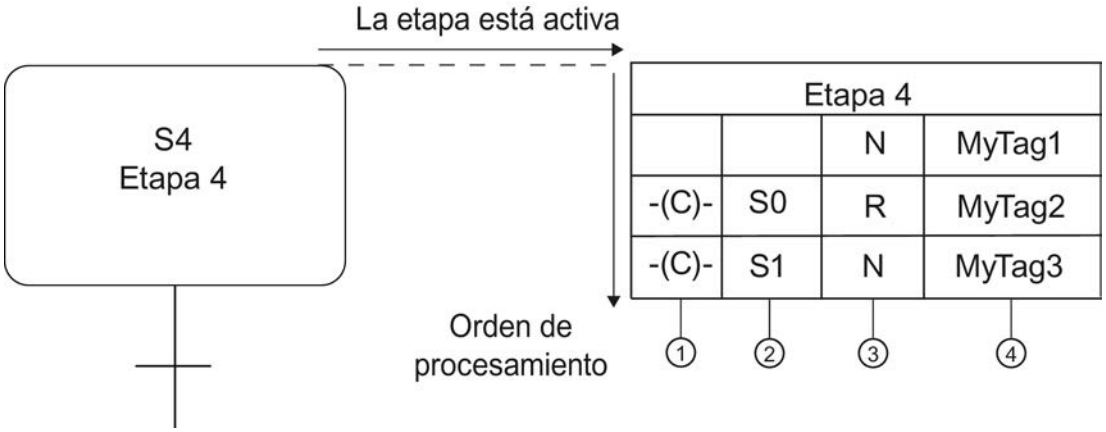
Las acciones permiten controlar entradas o salidas, activar y desactivar otras etapas de la cadena secuencial o llamar bloques, por ejemplo. Las acciones se ejecutan cuando se activa la etapa correspondiente (acciones estándar) o cuando se produce un evento relacionado con la acción (acciones controladas por evento). En las acciones también pueden utilizarse temporizadores y contadores.

La tabla siguiente muestra ejemplos de programación de acciones:

Interlock	Evento	Identificador	Acción	Descripción
		N	MyTag1	Acción compuesta por instrucción y operando sin interlock
-(C)-		N	MyTag2	Acción compuesta por instrucción y operando con interlock
	S1	S	MyTag3	Acción compuesta por evento, instrucción y operando sin interlock

Las acciones se ejecutan en orden descendente cuando la etapa está activa o se produce el evento.

La figura siguiente muestra un ejemplo del orden de procesamiento de las acciones de una etapa:



① Interlock

② Evento opcional

③ Identificador

④ Operando, asignación o llamada de bloque

Una acción consta de los componentes siguientes:

- Interlock (opcional)
  Es posible enlazar una acción con un interlock para influir en su ejecución.
- Evento (opcional)
  Un evento define cuándo debe ejecutarse una acción. Para algunos identificadores es obligatorio indicar un evento.
- Identificador (obligatorio)
  El identificador define el tipo de acciones que debe ejecutarse, p. ej. la activación o desactivación de un operando.
- Acción (obligatorio)
  La acción determina el operando con el que se ejecuta la misma.

Consulte también

Acciones estándar (Página 2837)

Acciones controladas por evento (Página 2839)

Llamadas de bloque (Página 2842)

Temporizadores (Página 2843)

Contadores (Página 2844)

Programar acciones (Página 2901)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Acciones estándar

Introducción

Las acciones estándar se ejecutan mientras una etapa está activa. La tabla siguiente muestra las posibles acciones estándar con sus identificadores:

Identificador	Tipo de datos del operando	Significado
N – Activar mientras la etapa esté activa	BOOL FB, FC, SFB, SFC	Mientras la etapa está activa, el estado lógico del operando es "1". Mientras la etapa está activa se efectúa la llamada del bloque indicado.
S - Poner a 1	BOOL	En cuanto la etapa se activa, el operando se pone a "1" y permanece posteriormente en "1".
R - Poner a 0	BOOL	En cuanto la etapa se activa, el operando se pone a "0" y permanece posteriormente en "0".
D - Retardo a la conexión	BOOL y Time/ DWORD	n segundos después de la activación de la etapa, el operando se pone a "1" y permanece en "1" mientras dura la activación de la etapa. No es aplicable si el tiempo de activación de la etapa es inferior a n segundos. El tiempo puede indicarse como constante o como variable PLC del tipo de datos TIME/DWORD.
L - Activar por tiempo limitado	BOOL y Time/ DWORD	Cuando la etapa está activa, el operando se pone a "1" durante n segundos. Después, el operando vuelve a ponerse a "0". El operando también se pone a "0" cuando el tiempo de activación de la etapa es inferior a la duración. El tiempo puede indicarse como constante o como variable PLC del tipo de datos TIME/DWORD.

Todas estas acciones estándar pueden asociarse a un interlock. Si una acción está enlazada con un interlock, dicha acción solo se ejecutará cuando se cumplan las condiciones del interlock.

La tabla siguiente muestra algunos ejemplos de uso de constantes de tiempo:

Identificador	Acción	Descripción
D	"MyTag", T#2s	2 segundos después de la activación de la etapa, el operando "MyTag" se pone a "1" y permanece en "1" mientras dura la activación de la etapa. No es aplicable si la duración de la activación de la etapa es inferior a 2 segundos. Con la desactivación de la etapa, el operando se pone a "0" (sin memoria).
L	"MyTag", T#20s	Cuando la etapa está activa, el operando "MyTag" se pone a "1" durante 20 segundos. Después, el operando se pone a "0" (sin memoria). El operando también se pone a "0" cuando el tiempo de activación de la etapa es inferior a 20 segundos.

Consulte también

Principios básicos de las acciones (Página 2836)

Acciones controladas por evento (Página 2839)

Llamadas de bloque (Página 2842)

Temporizadores (Página 2843)

Contadores (Página 2844)

Programar acciones (Página 2901)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)
Etapas y transiciones (Página 2831)
Operandos específicos de GRAPH (Página 2834)
Instrucciones permanentes (Página 2846)
Condiciones (Página 2847)
Avisos de interlock y supervisión (Página 2847)
Interfaz de un bloque de función GRAPH (Página 2848)

Acciones controladas por evento

Introducción

Existe la posibilidad de enlazar una acción con un evento para que la ejecución de dicha acción dependa de determinadas condiciones. Las acciones con los identificadores "D", "L" y "TF" no pueden enlazarse con eventos. Si una acción está asociada a un evento, el estado lógico del evento se detecta mediante evaluación de flancos. Es decir, las instrucciones solo se ejecutan en el mismo ciclo en el que se produce el evento.

La tabla siguiente muestra los eventos que pueden enlazarse con acciones:

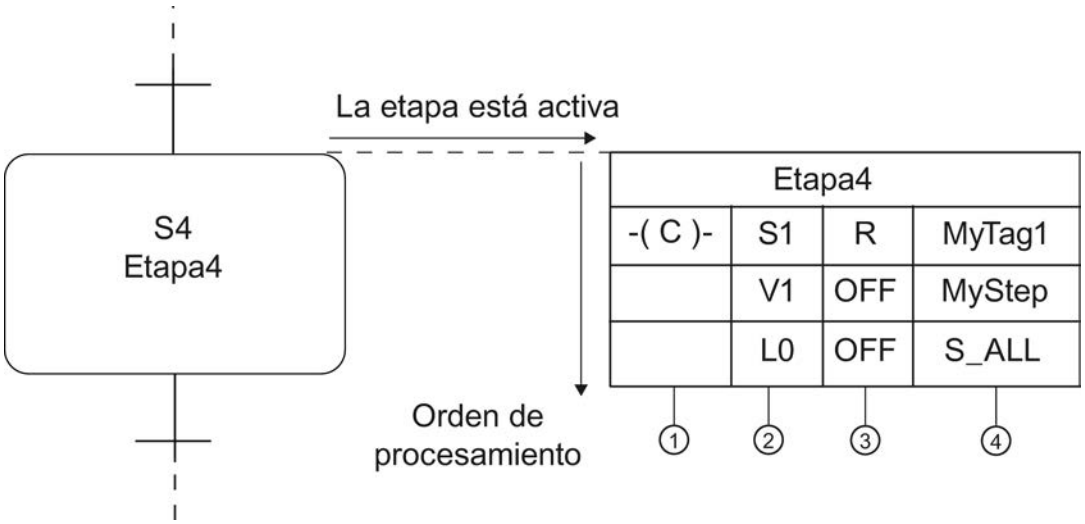
Evento	Evaluación de señal	Descripción
S1	Flanco ascendente	La etapa se activa (estado lógico = "1")
S0	Flanco descendente	La etapa se desactiva (estado lógico = "0")
V1	Flanco ascendente	La supervisión se cumple, es decir, hay un fallo (estado lógico = "1")
V0	Flanco descendente	La supervisión ya no se cumple, es decir, se ha solucionado el fallo (estado lógico = "0")
L0	Flanco ascendente	El interlock se cumple, es decir, se ha solucionado el fallo (estado lógico = "1")
L1	Flanco descendente	El interlock no se cumple, es decir, hay un fallo (estado lógico = "0")
A1	Flanco ascendente	Se acusa un aviso.
R1	Flanco ascendente	Llegada de un registro.

Los estados lógicos de un evento se pueden utilizar para programar otras acciones. De ese modo, no solo se pueden supervisar e influir las etapas individuales, sino todo el control secuencial.

Las acciones para las que se utilizan los identificadores "ON" (activar etapa) y "OFF" (desactivar etapa) deben enlazarse siempre con un evento. El evento determina cuándo se activa o desactiva la etapa. Si una etapa se activa y desactiva en un mismo ciclo, la desactivación tiene mayor prioridad.

Las acciones para las que se utilizan los eventos "S1", "V1", "A1" o "R1" pueden enlazarse adicionalmente con un interlock. De ese modo se ejecutan únicamente si las condiciones del interlock se cumplen.

La figura siguiente muestra un ejemplo de una etapa que contiene acciones controladas por evento:



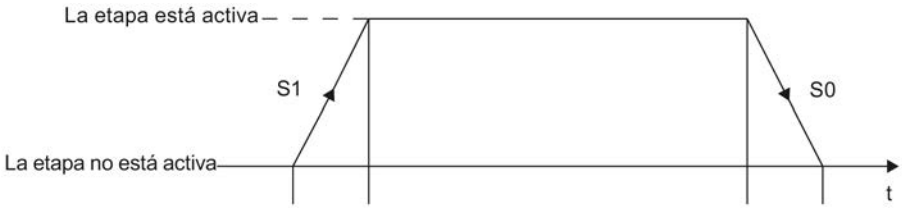
En cuanto la etapa "Step4" se activa y el interlock se cumple, "MyTag1" se pone a 0 y permanece posteriormente en "0". Cuando se produce un error de supervisión se desactiva la etapa "MyStep". Cuando se produce un interlock se desactivan todas las etapas.

Registro

Un registro es un evento que se lanza fuera del bloque y se consulta mediante un flanco de señal ascendente en uno de los parámetros de entrada "REG_S" o "REG_EF". Si el registro se efectúa a través del parámetro de entrada "REG_S", el evento se transmite únicamente a la etapa activa que se muestra en el parámetro de salida "S_NO". Si el registro se efectúa a través del parámetro de entrada "REG_EF", el evento se transmite a todas las etapas que se encuentren activas.

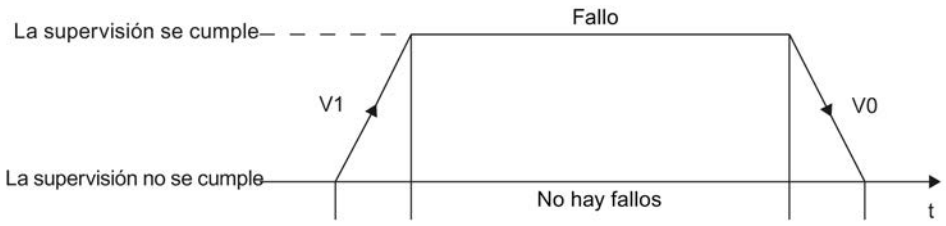
Ejemplos de la evaluación de señales de eventos

La figura siguiente muestra la evaluación de señales de una etapa:



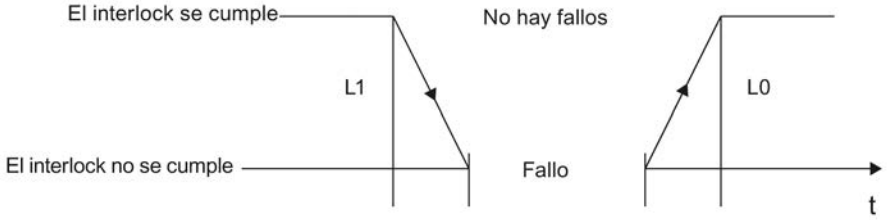
- S1: la etapa se activa
- S0: la etapa se desactiva

La figura siguiente muestra la evaluación de señales de una supervisión:



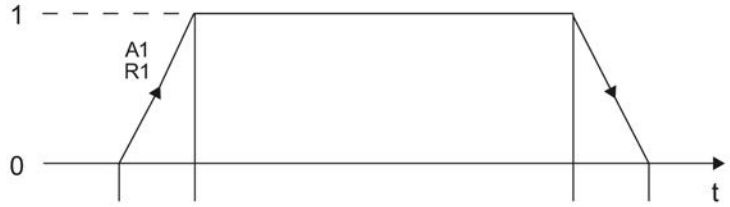
- V1: se produce un error de supervisión
- V0: el error de supervisión se ha solucionado

La figura siguiente muestra la evaluación de señales de un interlock:



- L1: el interlock ya no se cumple (fallo entrante)
- L0: el interlock se cumple (fallo saliente)

La figura siguiente muestra la evaluación de señales de un aviso y el registro:



- A1: se acusa un aviso
- R1: llega un registro (flanco ascendente en la entrada REG_EF / REG_S)

Consulte también

Principios básicos de las acciones (Página 2836)
Acciones estándar (Página 2837)
Llamadas de bloque (Página 2842)
Temporizadores (Página 2843)
Contadores (Página 2844)
Programar acciones (Página 2901)
Lenguaje de programación GRAPH (Página 2825)
Cadenas secuenciales (Página 2827)
Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Llamadas de bloque

Introducción

Dentro de las acciones es posible llamar otros bloques de función y otras funciones en los que se ejecutan determinadas tareas parciales. Esto permite estructurar mejor el programa.

Llamada de bloques de función (FB)

Para la llamada se debe utilizar la sintaxis siguiente:
```
CALL "<NombreFB>", "<NombreDB>" (lista de parámetros)
```

La tabla siguiente muestra un ejemplo de llamada de un bloque de función:

Interlock	Evento	Identificador	Acción
		N	CALL "FB_KOP", "FB_KOP_DB" (MyInOut := "MyTag2")

El bloque de función "FB_KOP" se llama con la instrucción "CALL". La instrucción "CALL" está siempre asociada al identificador "N". Tras el nombre del bloque llamado se encuentra el nombre del bloque de datos que contiene los datos y parámetros del bloque de función. En la lista de parámetros se asigna la variable "MyTag2" al parámetro "MyInOut".

Llamada de funciones (FC)

Para llamar una función se utiliza la siguiente sintaxis:
```
CALL "<NombreFC>" (lista de parámetros)
```

La tabla siguiente muestra un ejemplo de llamada de una función:

Interlock	Evento	Identificador	Acción
		N	CALL "FC_KOP" (MyInput := "MyTag1")

La función "FC_KOP" se llama con la instrucción "CALL". En la lista de parámetros se asigna la variable "MyTag1" al parámetro "MyInput".

Consulte también

Llamadas de bloque (Página 2263)

Inserción manual de llamadas de bloque (Página 2922)

Insertar llamadas de bloque mediante Drag & Drop (Página 2924)
Actualizar llamadas de bloque en GRAPH (Página 2925)
Principios básicos de las acciones (Página 2836)
Acciones estándar (Página 2837)
Acciones controladas por evento (Página 2839)
Temporizadores (Página 2843)
Contadores (Página 2844)
Lenguaje de programación GRAPH (Página 2825)
Operandos específicos de GRAPH (Página 2834)
Instrucciones permanentes (Página 2846)
Condiciones (Página 2847)
Avisos de interlock y supervisión (Página 2847)
Programar acciones (Página 2901)
Cadenas secuenciales (Página 2827)
Etapas y transiciones (Página 2831)
Interfaz de un bloque de función GRAPH (Página 2848)

Temporizadores

Introducción

Dentro de las acciones existe la posibilidad de utilizar temporizadores. Excepto "TF", todos los temporizadores dependen de un evento que determina el instante de activación del temporizador. El temporizador "TF" es activado por la propia etapa.

En caso de utilizar los temporizadores "TL", "TD" y "TF" hay que indicar el tiempo. Utilice para ello la sintaxis siguiente:

```
<Tiempo> = IWy, QWy, MWy, LWy, DBWy o DIWy; variable del tipo S5TIME
o WORD; constante S5T#
```

y = 0 ... 65534

Las constantes también se pueden introducir como datos de tiempo normales, p. ej. "5s".

En la siguiente tabla se muestran los temporizadores que se pueden utilizar en acciones:

Evento	Identificador	Descripción
S1, S0, L1, L0, V1, V0, A1, R1	TL	Impulso prolongado: el temporizador se inicia en cuanto se produce el evento definido. Durante el tiempo indicado, el temporizador tiene el estado lógico "1". Una vez finalizado el temporizador, pasa al estado lógico "0".
S1, S0, L1, L0, V1, V0, A1, R1	TD	Retardo a la conexión con memoria: el temporizador se inicia en cuanto se produce el evento definido. Durante el tiempo indicado, el temporizador tiene el estado lógico "0". Una vez finalizado el temporizador, pasa al estado lógico "1".
S1, S0, L1, L0, V1, V0, A1, R1	TR	Detener temporizador e inicializar: el temporizador se detiene en cuanto se produce el evento definido. El estado y el valor del temporizador se ponen a "0".
-	TF	Retardo a la desconexión: el estado del temporizador se pone a "1" en cuanto se activa la etapa. Al desactivarse la etapa, el temporizador está activado y no se pone a "0" hasta que finaliza el temporizador.

Consulte también

Principios básicos de las acciones (Página 2836)

Acciones estándar (Página 2837)

Acciones controladas por evento (Página 2839)

Llamadas de bloque (Página 2842)

Contadores (Página 2844)

Programar acciones (Página 2901)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Contadores

Introducción

Dentro de las acciones existe la posibilidad de utilizar contadores. Los contadores requieren siempre un evento para determinar el instante de activación del contador. Esto significa que

el contador se activará cuando se produzca el evento correspondiente. Las acciones para las que se utilizan los eventos S1, V1, A1 o R1 pueden enlazarse adicionalmente con un interlock. De este modo se ejecutan únicamente si las condiciones del interlock se cumplen.

En la siguiente tabla se muestran los contadores que se pueden utilizar en acciones:

Evento	Identificador	Tipo de datos del operando	Significado
S1, S0, L1, L0, V1, V0, A1, R1	CS	COUNTER	Poner contador al valor inicial: El contador se pone al valor de contaje indicado en cuanto se produce el evento definido. El valor de contaje puede indicarse como variable o constante del tipo de datos WORD (C#0 a C#999).
S1, S0, L1, L0, V1, V0, A1, R1	CU	COUNTER	Contador ascendente: El contador incrementa el valor de contaje en "1" en cuanto se produce el evento definido. El valor de contaje puede seguir incrementándose hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje ya no se incrementa más al detectarse un flanco ascendente.
S1, S0, L1, L0, V1, V0, A1, R1	CD	COUNTER	Contador descendente: El contador decrementa el valor de contaje en "1" en cuanto se produce el evento definido. El valor de contaje puede seguir decrementándose hasta alcanzar el límite "0". Una vez alcanzado el valor límite, el valor de contaje ya no se decrementa más al detectarse un flanco ascendente.
S1, S0, L1, L0, V1, V0, A1, R1	CR	COUNTER	Poner contador a 0: El contador se pone a "0" en cuanto se produce el evento definido.

Consulte también

Principios básicos de las acciones (Página 2836)

Acciones estándar (Página 2837)

Acciones controladas por evento (Página 2839)

Llamadas de bloque (Página 2842)

Temporizadores (Página 2843)

Programar acciones (Página 2901)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Instrucciones permanentes

Introducción

Las instrucciones permanentes ofrecen la posibilidad de crear un código del programa que se ejecute antes o después del procesamiento de la cadena secuencial. Esto permite programar condiciones y llamadas de bloque que se procesen en cada ciclo con independencia de la cadena secuencial.

Las instrucciones permanentes se crean en los lenguajes de programación KOP y FUP, con un máximo de 32 instrucciones KOP o FUP en cada instrucción permanente. Dentro de un bloque de función GRAPH es posible utilizar un máximo de 250 instrucciones permanentes anteriores y 250 posteriores. El resultado lógico de las condiciones se transfiere a una asignación que también dispone de las funciones de memoria "Activar" y "Desactivar". El operando utilizado (p. ej. una marca) se puede consultar en un segmento de transición, interlock y supervisión y en otros segmentos permanentes.

Consulte también

Vista "Instrucciones permanentes anteriores" (Página 2873)

Vista "Instrucciones permanentes posteriores" (Página 2878)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Condiciones

Introducción

Las condiciones son estados binarios del proceso que, en forma de elementos KOP o FUP (p. ej. contactos normalmente abiertos, contactos normalmente cerrados, operaciones lógicas Y, operaciones lógicas O o comparadores), se combinan entre sí en el esquema de contactos o en el diagrama de funciones según los principios de la lógica booleana. El resultado lógico (RLO) puede influir sobre acciones individuales de una etapa, sobre la etapa entera, la transición a la etapa siguiente o la cadena secuencial en su conjunto.

Las condiciones permiten programar los componentes siguientes del programa GRAPH:

- Instrucciones permanentes
- Interlocks
- Supervisiones
- Transiciones

En cada segmento es posible programar un máximo de 32 instrucciones con un máximo de 32 operandos interconectados. Encontrará información adicional sobre las variables utilizadas visualizando la información sobre las variables.

Consulte también

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Avisos de interlock y supervisión (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Avisos de interlock y supervisión

Introducción

Existe la posibilidad de detectar fallos acerca de interlocks y supervisiones. Para poder solucionar estos fallos con rapidez es posible definir avisos que describan con detalle los fallos. Para ello debe activarse primero la generación de avisos y adaptar seguidamente los textos de aviso a las necesidades particulares. También es posible determinar que los avisos deban acusarse.

Consulte también

Definir avisos de interlock y supervisión (Página 2926)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Interfaz de un bloque de función GRAPH (Página 2848)

Interfaz de un bloque de función GRAPH

Principios básicos de la interfaz de un bloque de función GRAPH

Juegos de parámetros de interfaz

Al crear un bloque GRAPH puede elegirse entre los siguientes juegos de parámetros de interfaz, que le proporcionan una cantidad variable de parámetros de entrada y salida:

- Juego de parámetros de interfaz máximo
- Juego de parámetros de interfaz estándar

El juego de parámetros de interfaz estándar permite la operación de la cadena secuencial en distintos modos de operación y ofrece la posibilidad de acusar avisos. Con el juego de parámetros de interfaz máximo se dispone de posibilidades adicionales de diagnóstico. En ambos juegos de parámetros se pueden borrar o agregar manualmente parámetros individuales.

Nota

A la hora de realizar la selección tenga en cuenta también el espacio de memoria disponible en la CPU, pues al aumentar el número de parámetros aumenta también la necesidad de espacio para el bloque GRAPH y el bloque de datos de instancia correspondiente.

Consulte también:

Parámetros de entrada (Página 2851)

Parámetros de salida (Página 2855)

Modelos de espacio de memoria para el bloque de datos instancia (S7-300/400)

Para CPUs de la serie S7-300/400 también es posible especificar el modelo de espacio de memoria para el bloque de datos de instancia. Existen las siguientes posibilidades:

- bloque de datos estándar
- bloque de datos minimizado

Al seleccionar el modelo de memoria se define el número de parámetros estáticos. Básicamente, la información sobre las etapas y transiciones individuales se almacena como parámetros estáticos en ambos modelos de memoria. Para cada etapa y cada transición se crea una estructura propia con información detallada. En un bloque de datos estándar se obtiene información más detallada que en un bloque de datos minimizado, pero el bloque de datos estándar requiere para ello una cantidad de espacio en memoria notablemente superior.

En un bloque de datos minimizado existen las restricciones siguientes:

- Algunos elementos de las estructuras de etapa y de transición no están disponibles.
- La opción "Saltar etapas" no está disponible.
- No hay avisos disponibles.
- Los números de etapa y de transición deben ser correlativos. Si no es así, las etapas y transiciones se renumeran automáticamente durante la compilación.
- No se crean datos para el análisis de criterios.

Consulte también: Parámetros estáticos (Página 2857)

Acceso a los parámetros de interfaz

Por principio, en una CPU S7-1500 no es posible acceder en escritura a los parámetros de interfaz internos de GRAPH Sin embargo, los siguientes componentes de la estructura "MOP" en la sección "Static" que gestionan los modos de operación sí permiten un acceso en escritura:

- AUTO
- MAN
- TAP
- TOP
- LOCK
- SUP
- OFF
- INIT
- HALT
- TMS_HALT
- OPS_ZERO
- S_PREV
- S_NEXT

- S_ON
- S_OFF
- T_PUSH
- ACK
- IL_PERM
- T_PERM
- ILP_MAN

Estos parámetros reaccionan a un flanco ascendente. Asimismo, en las CPU S7-300/400 evite acceder en escritura a otros parámetros que los mencionados, pues en caso contrario no queda asegurada una ejecución correcta de la cadena secuencial.

Consulte también: Otros parámetros estáticos (Página 2860)

Consulte también

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Etapas y transiciones (Página 2831)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Parámetros de entrada

Resumen de los parámetros de entrada de un bloque de función GRAPH

Los parámetros de entrada de un bloque de función GRAPH reaccionan a un flanco ascendente. La tabla siguiente muestra una relación de los parámetros de entrada de un bloque de función GRAPH:

Parámetro	Tipo de datos	Descripción	Juego de parámetros estándar	Juego de parámetros máximo
OFF_SQ	BOOL	OFF_SEQUENCE: desconectar cadena secuencial, es decir, desactivar todas las etapas Tipo: Requisito	x	x
INIT_SQ	BOOL	INIT_SEQUENCE: activar etapas iniciales, resetear cadena secuencial Tipo: Requisito	x	x
ACK_EF	BOOL	ACKNOWLEDGE_ERROR_FAULT: acusar una avería, forzar transición Tipo: Requisito	x	x
REG_EF	BOOL	REGISTRATE_ERROR_FAULT: registrar todos los errores y averías Tipo: Requisito	-	x
ACK_S	BOOL	ACKNOWLEDGE_STEP: acusar la etapa mostrada en el parámetro de salida "S_NO" Tipo: Requisito	-	x
REG_S	BOOL	REGISTRATE_STEP: registrar la etapa mostrada en el parámetro de salida "S_NO" Tipo: Requisito	-	x
HALT_SQ	BOOL	HALT_SEQUENCE: detener/reactivar cadena secuencial Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x
HALT_TM	BOOL	HALT_TIMES: detener/reactivar todos los tiempos de activación de etapa y las operaciones en función del tiempo (L y D) de la cadena secuencial Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x

Parámetro	Tipo de datos	Descripción	Juego de parámetros estándar	Juego de parámetros máximo
ZERO_OP	BOOL	ZERO_OPERANDS: poner a cero todos los operandos con el identificador N, D, L en etapas activas y no ejecutar las instrucciones CALL en acciones/reactivar los operandos e instrucciones CALL Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x
EN_IL	BOOL	ENABLE_INTERLOCKS: desactivar interlock (la cadena secuencial se comporta como si el interlock se cumpliera)/reactivarlo Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x
EN_SV	BOOL	ENABLE_SUPERVISIONS: desactivar supervisión (la cadena secuencial se comporta como si el interlock no se cumpliera)/reactivarlo Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x
EN_ACKREQ	BOOL	ENABLE_ACKNOWLEDGE_REQUIRED: activar acuse obligatorio Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x
EN_SSKIP	BOOL	ENABLE_STEP_SKIPPING: activar salto de etapa Tipo: Estado, la reinicialización se produce con un nuevo flanco ascendente	-	x
DISP_SACT	BOOL	DISPLAY_ACTIVE_STEPS: visualizar solo las etapas activas Tipo: Estado, la reinicialización se produce con un flanco ascendente en DISP_SEF, DISP_SALL	-	x
DISP_SEF	BOOL	DISPLAY_STEPS_WITH_ERROR_OR_FAULT: visualizar solo las etapas con errores y averías Tipo: Estado, la reinicialización se produce con un flanco ascendente en DISP_SACT, DISP_SALL	-	x
DISP_SALL	BOOL	DISPLAY_ALL_STEPS: visualizar todas las etapas Tipo: Estado, la reinicialización se produce con un flanco ascendente en DISP_SACT, DISP_SEF	-	x
S_PREV	BOOL	PREVIOUS_STEP:modo automático: retroceder página por las etapas actualmente activas, visualización del número de etapa en el parámetro de salida "S_NO" Modo manual: visualizar la etapa anterior en "S_NO" (número más pequeño) Tipo: Requisito	x	x

Parámetro	Tipo de datos	Descripción	Juego de parámetros estándar	Juego de parámetros máximo
S_NEXT	BOOL	NEXT_STEP: modo Automático: avanzar página por las etapas actualmente activas, visualización del número de etapa en el parámetro de salida "S_NO" Modo manual: visualizar la etapa siguiente en "S_NO" (número más grande) Tipo: Requisito	x	x
SW_AUTO	BOOL	SWITCH_MODE_AUTOMATIC: cambio de modo de operación: modo automático Tipo: Estado, la reinicialización se produce con un flanco ascendente en SW_TAP, SW_TOP, SW_MAN	x	x
SW_TAP	BOOL	SWITCH_MODE_TRANSITION: cambio de modo de operación: modo semiautomático Tipo: Estado, la reinicialización se produce con un flanco ascendente en SW_AUTO, SW_TOP, SW_MAN	x	x
SW_TOP	BOOL	SWITCH_MODE_TRANSITION_OR_PUSH: cambio de modo de operación: modo automático o semiautomático Tipo: Estado, la reinicialización se produce con un flanco ascendente en SW_AUTO, SW_TAP, SW_MAN	x	x
SW_MAN	BOOL	SWITCH_MODE_MANUAL: cambio de modo de operación: modo manual, no se lanza ninguna secuencia automática Tipo: Estado, la reinicialización se produce con un flanco ascendente en SW_AUTO, SW_TAP, SW_TOP	x	x
S_SEL	INT	STEP_SELECT: elegir el número de etapa del parámetro de salida "S_NO" en modo manual, activar/desactivar con "S_ON", "S_OFF" Tipo: N/A	x	x
S_SELOK	BOOL	STEP_SELECT_OK: aplicar valor de "S_SEL" para el parámetro de salida "S_NO" Tipo: Requisito	-	x
S_ON	BOOL	STEP_ON: modo manual: activar la etapa visualizada Tipo: Requisito	x	x
S_OFF	BOOL	STEP_OFF: modo manual: desactivar la etapa visualizada Tipo: Requisito	x	x
T_PREV	BOOL	PREVIOUS_TRANSITION: visualizar la anterior transición válida en el parámetro de salida "T_NO" Tipo: Requisito	-	x

Parámetro	Tipo de datos	Descripción	Juego de parámetros estándar	Juego de parámetros máximo
T_NEXT	BOOL	NEXT_TRANSITION: visualizar la siguiente transición válida en el parámetro de salida "T_NO" Tipo: Requisito	-	x
T_PUSH	BOOL	PUSH_TRANSITION: la transición conmuta si la condición se cumple y "T_PUSH" (flanco) Requisitos: modo automático o modo manual Tipo: Requisito	x	x

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)
Parámetros de salida (Página 2855)
Parámetros estáticos (Página 2857)
Definir el juego de parámetros de interfaz (Página 2868)
Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)
Lenguaje de programación GRAPH (Página 2825)
Cadenas secuenciales (Página 2827)
Etapas y transiciones (Página 2831)
Operandos específicos de GRAPH (Página 2834)
Acciones (Página 2836)
Instrucciones permanentes (Página 2846)
Condiciones (Página 2847)
Avisos de interlock y supervisión (Página 2847)

Parámetros de salida

Resumen de los parámetros de salida de un bloque de función GRAPH

La tabla siguiente muestra una relación de los parámetros de salida de un bloque de función GRAPH:

Parámetro	Tipo de datos	Descripción	Juego de parámetros estándar	Juego de parámetros máximo
S_NO	INT	STEP_NUMBER: muestra el número de etapa	x	x
S_MORE	BOOL	MORE_STEPS: hay más etapas activas	x	x
S_ACTIVE	BOOL	STEP_ACTIVE: la etapa visualizada está activa	x	x
S_TIME	TIME	STEP_TIME: tiempo de activación de la etapa	-	x
S_TIMEOK	TIME	STEP_TIME_OK: tiempo de activación de la etapa sin errores	-	x
S_CRITLOC (solo S7-300/400)	DWORD	STEP_CRITERIA_INTERLOCK: bits de criterios de interlock	-	x
S_CRITLOCERR (solo S7-300/400)	DWORD	S_CRITERIA_IL_LAST_ERROR: bits de criterios de interlock con evento L1	-	x
S_CRITSUP (solo S7-300/400)	DWORD	STEP_CRITERIA_SUPERVISION: bits de criterios de supervisión	-	x
S_STATE	WORD	STEP_STATE: bits de estado de etapa	-	x
T_NO	INT	TRANSITION: número de transición válido	-	x
T_MORE	BOOL	MORE_TRANSITIONS: otras transiciones válidas disponibles para visualización	-	x
T_CRIT	DWORD	TRANSITION_CRITERIA: bits de criterios de transición	-	x
T_CRITOLD	DWORD	T_CRITERIA_LAST_CYCLE: bits de criterios de transición del último ciclo	-	x
T_CRITFLT	DWORD	T_CRITERIA_LAST_FAULT: bits de criterios de transición con evento V1	-	x
ERROR	BOOL	INTERLOCK_ERROR: error de interlock (cualquier etapa)	-	x

Parámetro	Tipo de datos	Descripción	Juego de parámetros estándar	Juego de parámetros máximo
FAULT	BOOL	SUPERVISION_FAULT: error de supervisión (cualquier etapa)	-	x
ERR_FLT	BOOL	IL_ERROR_OR_SV_FAULT: avería de grupo	x	x
SQ_ISOFF	BOOL	SEQUENCE_IS_OFF: la cadena secuencial está desactivada (ninguna etapa activa)	-	x
SQ_HALTED	BOOL	SEQUENCE_IS_HALTED: cadena secuencial detenida	-	x
TM_HALTED	BOOL	TIMES_ARE_HALTED: temporizadores detenidos	-	x
OP_ZEROED	BOOL	OPERANDS_ARE_ZEROED: operandos reseteados	-	x
IL_ENABLED	BOOL	INTERLOCK_IS_ENABLED: el interlock se tiene en cuenta	-	x
SV_ENABLED	BOOL	SUPERVISION_IS_ENABLED: la supervisión se tiene en cuenta	-	x
ACKREQ_ENABLED	BOOL	ACKNOWLEDGE_REQUIRED_IS_ENABLED: acuse obligatorio activado	-	x
SSKIP_ENABLED	BOOL	STEP_SKIPPING_IS_ENABLED: salto de etapa activado	-	x
SACT_DISP	BOOL	ACTIVE_STEPS_WERE_DISPLAYED: visualizar solo etapas activas en "S_NO"	-	x
SEF_DISP	BOOL	STEPS_WITH_ERROR_FAULT_WERE_DISPLAYED: visualizar solo las etapas con errores y averías en "S_NO"	-	x
SALL_DISP	BOOL	ALL_STEPS_WERE_DISPLAYED: visualizar todas las etapas en "S_NO"	-	x
AUTO_ON	BOOL	AUTOMATIC_IS_ON: visualización del modo de operación automático	x	x
TAP_ON	BOOL	T_AND_PUSH_IS_ON: visualización del modo semiautomático	x	x
TOP_ON	BOOL	T_OR_PUSH_IS_ON: visualización del modo semiautomático	x	x
MAN_ON	BOOL	MANUAL_IS_ON: visualización del modo manual	x	x

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)
Parámetros de entrada (Página 2851)
Parámetros estáticos (Página 2857)
Definir el juego de parámetros de interfaz (Página 2868)
Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)
Lenguaje de programación GRAPH (Página 2825)
Cadenas secuenciales (Página 2827)
Etapas y transiciones (Página 2831)
Operandos específicos de GRAPH (Página 2834)
Acciones (Página 2836)
Instrucciones permanentes (Página 2846)
Condiciones (Página 2847)
Avisos de interlock y supervisión (Página 2847)

Parámetros estáticos

Parámetros estáticos de una etapa

Resumen de los parámetros estáticos de una etapa

La tabla siguiente muestra los parámetros que se crean en la sección "Static" de la interfaz del bloque para cada etapa del bloque de función GRAPH:

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
S1	La etapa se activa	BOOL	x	x
L1	El interlock desaparece	BOOL	x	x
V1	La supervisión aparece	BOOL	x	x
R1	Reservado	BOOL	x	x
A1	El error se acusa	BOOL	x	x
S0	La etapa se desactiva	BOOL	x	x
L0	El interlock aparece	BOOL	x	x
V0	La supervisión desaparece	BOOL	x	x
X	La etapa está activa	BOOL	x	x
LA	El interlock no se cumple	BOOL	x	x

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
VA	La supervisión está pendiente	BOOL	x	x
RA	Reservado	BOOL	x	x
AA	Reservado	BOOL	x	x
SS	Interno del sistema	BOOL	x	x
LS	Resultado directo del interlock programado	BOOL	x	x
VS	Resultado directo de la supervisión programada	BOOL	x	x
SNO	Número de etapa de usuario	INT		x
T	Tiempo total de activación de etapa	TIME	x	x
U	Tiempo de activación de etapa sin averías	TIME		x
CRIT_LOC (solo S7-300/400)	Estado de los máx. 32 elementos KOP/ FUP en el interlock en el ciclo de procesamiento actual	DWORD		x
CRIT_LOC_ ERR (solo S7-300/400)	Copia de CRIT_LOC si el interlock desaparece	DWORD		x
CRIT_SUP (solo S7-300/400)	Estado de los máx. 32 elementos KOP/ FUP en la supervisión en el ciclo de procesamiento actual	DWORD		x
SM	Interno del sistema	BOOL		x
LP (solo S7-300/400)	Interno del sistema	BOOL	x	x
LN (solo S7-300/400)	Interno del sistema	BOOL	x	x
VP (solo S7-300/400)	Interno del sistema	BOOL	x	x
VN (solo S7-300/400)	Interno del sistema	BOOL	x	x
H_IL_ERR	Interno del sistema	BYTE		x
H_SV_FLT	Interno del sistema	BYTE		x
RESERVED (solo S7-300/400)	Reservado	DWORD		x

Acceso a los componentes

Dado que los valores de los parámetros influyen en la ejecución de la cadena, el acceso de escritura no resulta recomendable. El acceso de lectura se efectúa a través del nombre de etapa seguido del nombre del elemento de estructura. Así, p. ej., algunos accesos posibles a las condiciones y acciones son:

- SECAR.X
- SECAR.T

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)
Parámetros de entrada (Página 2851)
Parámetros de salida (Página 2855)
Parámetros estáticos de una transición (Página 2859)
Otros parámetros estáticos (Página 2860)
Definir el juego de parámetros de interfaz (Página 2868)
Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)
Lenguaje de programación GRAPH (Página 2825)
Cadenas secuenciales (Página 2827)
Etapas y transiciones (Página 2831)
Operandos específicos de GRAPH (Página 2834)
Acciones (Página 2836)
Instrucciones permanentes (Página 2846)
Condiciones (Página 2847)
Avisos de interlock y supervisión (Página 2847)

Parámetros estáticos de una transición

Estructura de una transición en la interfaz

La tabla siguiente muestra los parámetros que se crean en la sección "Static" de la interfaz del bloque para cada transición del bloque de función GRAPH:

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
TV	La transición es válida	BOOL	x	x
TT	La transición se cumple	BOOL	x	x
TS	La transición conmuta	BOOL	x	x
CF_IV (solo S7-300/400)	La entrada "CRIT_FLT" no es válida	BOOL		x
TNO	Visualización del número de transición definido por el usuario	INT		x
CRIT (solo S7-300/400)	Estado de los máx. 32 elementos KOP/FUP de la transición en el ciclo de procesamiento actual	DWORD		x
CRIT_OLD (solo S7-300/400)	Estado de los máx. 32 elementos KOP/FUP de la transición en el ciclo de procesamiento anterior	DWORD		x
CRIT_FLT (solo S7-300/400)	Copia de "CRIT" si se produce un error	DWORD		x

Acceso a los componentes

Dado que los valores de los parámetros influyen en la ejecución de la cadena, el acceso de escritura no resulta recomendable. El acceso de lectura se efectúa a través del nombre de transición seguido del nombre del elemento de estructura. Así, p. ej., algunos accesos posibles a las condiciones y acciones son:

- VÁLVULA_ON.TT
- VÁLVULA_ON.CRIT

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)

Parámetros de entrada (Página 2851)

Parámetros de salida (Página 2855)

Parámetros estáticos de una etapa (Página 2857)

Otros parámetros estáticos (Página 2860)

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Otros parámetros estáticos

Resumen

Los parámetros estáticos incluyen la información siguiente, además de los parámetros de etapa y transición:

- Parámetros internos "S_DISPLAY", "S_SEL_OLD", "S_DISPIDX" y "T_DISPIDX"
- Componente "Modo de operación"
- Componente "Ciclos de tiempo"
- Componente "Marcas de cadena"

Nota

Un bloque de datos instancia específico de GRAPH contiene otros parámetros más. No obstante, estos están destinados únicamente al uso interno. No resulta recomendable utilizar estos parámetros en el programa.

Parámetros internos

La tabla siguiente muestra los parámetros internos que se encuentran en la sección "Static". Sirven para administrar los números de etapa y de transición:

Componente	Descripción	Tipo de datos	Longitud en bytes	DB minimizado	DB estándar
S_DISPLAY	Copia del parámetro de salida "S_NO"	INT	2	x	x
S_SEL_OLD	Número de etapa de parámetro de entrada "S_SEL" del último ciclo de la CPU	INT	2	x	x
S_DISPIDX	Índice para la etapa en el parámetro de salida "S_NO"	BYTE	1	x	x
T_DISPIDX	Índice para la transición en el parámetro de salida "T_NO"	BYTE	1	x	x

Modo de operación

La tabla siguiente muestra los componentes de la estructura "MOP" de la sección "Estático" que administran los modos de operación:

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
AUTO	Cambio al modo automático	BOOL	x	x
MAN	Cambio al modo manual	BOOL	x	x
TAP	Cambio al modo de operación "Jog"	BOOL	x	x
TOP	Cambio al modo de operación "Automático" o "Transición"	BOOL	x	x
ACK_S	Requisitos: acusar la etapa pendiente en el parámetro de salida "S_NO"	BOOL	x	x
REG_S	Requisitos: registrar la etapa pendiente en el parámetro de salida "S_NO"	BOOL	x	x
T_PREV	Requisitos: emitir la anterior transición válida en el parámetro de salida "T_NO"	BOOL	x	x
T_NEXT	Requisitos: emitir la siguiente transición válida en el parámetro de salida "T_NO"	BOOL	x	x
LOCK	Conectar/desconectar tratamiento de interlock	BOOL	x	x

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
SUP	Conectar/desconectar tratamiento de supervisión	BOOL	x	x
ACKREQ	Requisitos: acusar la etapa pendiente en el parámetro de salida "S_NO"	BOOL	x	x
SSKIP	Estado: "Salto de etapa" activado	BOOL	x	x
OFF	Desactivar cadena secuencial	BOOL	x	x
INIT	Activar etapas iniciales	BOOL	x	x
HALT	Detener la cadena/dejar que prosiga	BOOL	x	x
TMS_HALT	Detener los temporizadores/dejar que prosigan	BOOL	x	x
OPS_ZERO	Desconectar/conectar acciones	BOOL	x	x
SACT_DISP	Estado: visualizar solo las etapas activas	BOOL	x	x
SEF_DISP	Estado: visualizar solo las etapas con errores y averías	BOOL	x	x
SALL_DISP	Estado: visualizar todas las etapas	BOOL	x	x
S_PREV	Modo de operación "Automático": visualizar la etapa anterior simultáneamente activa en el parámetro de salida "S_NO" Modo de operación "Manual": visualizar el número de etapa siguiente más pequeño	BOOL	x	x
S_NEXT	Modo de operación "Automático": visualizar la etapa posterior simultáneamente activa en el parámetro de salida "S_NO" Modo de operación "Manual": visualizar el número de etapa siguiente más grande	BOOL	x	x
S_SELOK	Requisitos: emitir en el parámetro de salida "S_NO" el número de etapa pendiente en el parámetro de entrada "S_SEL"	BOOL	x	x
S_ON	Modo de operación "Manual": activar la etapa visualizada	BOOL	x	x
S_OFF	Modo de operación "Manual": desactivar la etapa visualizada	BOOL	x	x
T_PUSH	Modo de operación "Jog": la transición conmuta si la condición cumplida se aplica con "T_PUSH"	BOOL	x	x
REG	Requisitos: registrar todos los errores de interlock y de supervisión	BOOL	x	x
ACK	Acuse de la avería	BOOL	x	x
IL_PERM	Ejecución de interlock permanente	BOOL	x	x
T_PERM	Ejecución de transición permanente	BOOL	x	x
ILP_MAN	Ejecución de interlock permanente en modo de operación "Manual"	BOOL	x	x

En las CPUs de la serie S7-1500, la estructura "MOP" se encuentra dentro de la estructura "RT_DATA".

Ciclos de reloj (solo S7-300/400)

La tabla siguiente muestra los componentes de la estructura "TICKS" de la sección "Estático" que administran la información para el cálculo del tiempo de activación. Esta información solo puede ser evaluada internamente por el sistema.

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
DELTA	Diferencia de tiempo entre dos ciclos	TIME	x	x
OLD	Estado de contador cíclico 10 ms en el último ciclo	TIME	x	x
NEW	Estado de contador cíclico 10 ms en este ciclo	TIME	x	x

Marcas de cadena

La tabla siguiente muestra los componentes de la estructura "SQ_FLAGS" de la sección "Estático" que vigilan el estado interno de la cadena:

Componente	Descripción	Tipo de datos	DB minimizado	DB estándar
ERR_FLT	Avería de grupo	BOOL	x	x
ERROR	Error de interlock	BOOL	x	x
FAULT	Error de supervisión	BOOL	x	x
RT_FAIL	Error de tiempo de ejecución	BOOL	x	x
NO_SNO	La etapa seleccionada no existe	BOOL	x	x
NF_OFL	Demasiadas etapas que activar o desactivar	BOOL	x	x
SA_OFL	Demasiadas etapas activas	BOOL	x	x
TV_OFL	Demasiadas transiciones válidas	BOOL	x	x
MSG_OFL	Demasiados avisos	BOOL	x	x
NO_SWI	No conmutar en este ciclo	BOOL	x	x
CYC_OP	Procesamiento cíclico de la cadena secuencial tras la inicialización	BOOL	x	x
AS_MSG	Avisos activados o desactivados en tiempo de ejecución por instrucción	BOOL	x	x
AS_SEND	Enviar avisos de "WR_USMSG" o introducirlos únicamente en el búfer de diagnóstico	BOOL	x	x
SQ_BUSY	Marca de flanco interna para ejecución de proceso	BOOL	x	x
SA_BUSY	Marca de flanco interna para ejecución de proceso	BOOL	x	x
AS_SIG	Marca de flanco para avisos de "Alarm_S" y "Alarm_SQ"	BOOL	x	x

En las CPUs de la serie S7-1500, la estructura "SQ_FLAGS" se encuentra dentro de la estructura "RT_DATA".

Acceso a los componentes

Dado que los valores de los parámetros influyen en la ejecución de la cadena, el acceso de escritura no resulta recomendable. No obstante, el acceso de lectura es posible.

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)

Parámetros de entrada (Página 2851)

Parámetros de salida (Página 2855)

Parámetros estáticos de una etapa (Página 2857)

Parámetros estáticos de una transición (Página 2859)

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Lenguaje de programación GRAPH (Página 2825)

Cadenas secuenciales (Página 2827)

Etapas y transiciones (Página 2831)

Operandos específicos de GRAPH (Página 2834)

Acciones (Página 2836)

Instrucciones permanentes (Página 2846)

Condiciones (Página 2847)

Avisos de interlock y supervisión (Página 2847)

Configuración de GRAPH

Sinopsis de la configuración de GRAPH

Sinopsis

La tabla siguiente muestra los ajustes que se pueden efectuar:

Grupo	Área	Ajuste	Descripción
Vista		Longitud de los nombres de etapas y transiciones	Especifica si los nombres de las etapas y transiciones deben visualizarse en una o dos líneas. El ajuste "Mostrar dos filas" permite ver también nombres largos.
Cargar		Desactivar cadena antes de cargar el DB	Antes de cargar el bloque de datos de instancia se desactivan todas las etapas activas.

Grupo	Área	Ajuste	Descripción
Ajuste predeterminado para vigilancias de tiempo		Tiempo de activación de la etapa sin interrupciones (U)	Tiempo de activación para la vigilancia de tiempo de una etapa mediante la instrucción "Tiempo de activación de etapa ininterrumpido mayor".
		Tiempo de activación de la etapa (T)	Tiempo de activación para la vigilancia de tiempo de una etapa mediante la instrucción "Tiempo de activación de etapa mayor".
Ajuste predeterminado para bloques nuevos	Editor	Lenguaje en segmentos	Selección del lenguaje de programación para programar las condiciones.
	Interfaz	Parámetros máximos de interfaz	Inserta el número máximo de parámetros en la interfaz del bloque al crear bloques de función GRAPH.
	Compilar	Generar DB minimizado	Genera durante la compilación un bloque de datos de instancia con un juego reducido de parámetros estáticos. De todas formas, esto provoca restricciones en el funcionamiento. Este ajuste puede volver a modificarse posteriormente para bloques individuales. Consulte también: Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)
	Propiedades de cadena	Saltar etapas	Salta las etapas en las que la transición siguiente se cumple con la simple activación de la etapa. En este caso no se activa la etapa. Este ajuste puede modificarse posteriormente para bloques individuales.
		Acuse obligatorio de errores de supervisión	Determina que los errores de supervisión deban acusarse antes de seguir procesando la cadena secuencial.
		Procesamiento permanente de todos los interlocks en modo manual	Permite el procesamiento permanente de todos los interlocks en modo manual.
		Bloqueo de selección de modo	Impide el cambio manual del modo de operación mediante la Task Card "Test". Este ajuste puede modificarse posteriormente para bloques individuales.
		Asignar ENO automáticamente	Muestra el operando adicional "ENO" al ejecutar un test con el estado del programa en las CPU de la serie S7-1500.
	Propiedades de avisos	Activar avisos	Activa los avisos de interlock y supervisión para bloques de función GRAPH nuevos. Este ajuste puede modificarse posteriormente para bloques individuales.
	Avisos de interlock	Acuse obligatorio de avisos de interlock	Determina que los avisos de interlock deban acusarse antes de seguir procesando la etapa. Este ajuste puede modificarse posteriormente para bloques individuales.
		Texto de aviso	Define el texto de los avisos de interlock. Este texto puede modificarse posteriormente para bloques individuales.

Grupo	Área	Ajuste	Descripción
	Avisos de supervisión	Acuse obligatorio de avisos de supervisión	Determina que los avisos de supervisión deban acusarse para cerrarlos.
		Texto de aviso	Define el texto de los avisos de supervisión. Este texto puede modificarse posteriormente para bloques individuales.
Sincronización		Transición anterior cumplida	Al buscar posibles puntos de sincronización, marca las etapas cuya transición anterior se ha cumplido. No se cumple la siguiente transición.
		El interlock se cumple	Al buscar posibles puntos de sincronización, marca las etapas cuyo interlock se ha cumplido. Se cumple la siguiente transición.

Consulte también

Sinopsis de las propiedades de bloques (Página 2608)

Modificar la configuración (Página 2866)

Definir el lenguaje de programación en segmentos (Página 2867)

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Modificar ajustes para parámetros internos (Página 2870)

Modificar la configuración

Procedimiento

Para modificar la configuración, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Modifique la configuración.

Resultado

El cambio se aplica sin necesidad de guardarlo explícitamente.

Consulte también

Sinopsis de la configuración de GRAPH (Página 2864)

Definir el lenguaje de programación en segmentos (Página 2867)

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Modificar ajustes para parámetros internos (Página 2870)

Definir el lenguaje de programación en segmentos

Para bloques de función GRAPH no es posible cambiar el lenguaje de programación del bloque. Sin embargo, es posible definir el lenguaje de programación de los segmentos para condiciones. Puede elegir entre KOP y FUP. El lenguaje de programación se define en el árbol del proyecto o en la ventana de inspección.

Definir el lenguaje de programación para segmentos en el árbol del proyecto

Para definir el lenguaje de programación para segmentos en el árbol del proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic con el botón derecho del ratón en el bloque de función GRAPH.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo con las propiedades del bloque.
3. Seleccione la entrada "General" en la navegación local.
4. En el área "Bloque", seleccione el lenguaje de programación que desea utilizar en la lista desplegable "Lenguaje en segmentos".
5. Confirme la selección con "Aceptar".

Definir el lenguaje de programación para segmentos en la ventana de inspección

Para definir el lenguaje de programación para segmentos en la ventana de inspección, proceda del siguiente modo:

1. Abra el bloque de función GRAPH cuyo lenguaje de programación desea cambiar.
2. Abra la ficha "Propiedades" en la ventana de inspección.
3. Seleccione la entrada "General" en la navegación local de la ventana de inspección.
4. En el área "Bloque", seleccione el lenguaje de programación que desea utilizar en la lista desplegable "Lenguaje en segmentos".

Consulte también

Sinopsis de la configuración de GRAPH (Página 2864)

Modificar la configuración (Página 2866)

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Modificar ajustes para parámetros internos (Página 2870)

Definir el juego de parámetros de interfaz

El juego de parámetros de interfaz para bloques de función GRAPH se define de uno de los siguientes modos:

- globalmente para todos los bloques de función GRAPH nuevos
- localmente para un bloque de función GRAPH abierto

El ajuste local tiene prioridad frente al ajuste global.

Nota

En todo momento se pueden borrar o agregar manualmente parámetros individuales de entrada y salida en la interfaz del bloque.

Definir el juego de parámetros de interfaz globalmente para todos los bloques de función GRAPH nuevos

Para definir el juego de parámetros de interfaz globalmente para todos los bloques de función GRAPH nuevos, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Si desea utilizar el juego de parámetros de interfaz máximo, active la opción "Parámetros de interfaz máximos".
4. Si desea utilizar el juego de parámetros de interfaz estándar, desactive la opción "Parámetros de interfaz máximos".

Definir el juego de parámetros de interfaz localmente para un bloque de función GRAPH abierto

Para definir el juego de parámetros de interfaz localmente para un bloque de función GRAPH abierto, proceda del siguiente modo:

1. Si desea utilizar el juego de parámetros de interfaz máximo, seleccione el comando "Parámetros de interfaz máximos" del menú "Edición".
2. Si desea utilizar el juego de parámetros de interfaz estándar, seleccione el comando "Parámetros de interfaz estándar" del menú "Edición".

La interfaz del bloque de función GRAPH se adapta.

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)

Sinopsis de la configuración de GRAPH (Página 2864)

Modificar la configuración (Página 2866)

Definir el lenguaje de programación en segmentos (Página 2867)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

Modificar ajustes para parámetros internos (Página 2870)

Definir el modelo de espacio de memoria para el bloque de datos

El modelo de espacio de memoria para bloques de función GRAPH se define de uno de los siguientes modos:

- globalmente para todos los bloques de función GRAPH nuevos
- localmente para un bloque de función GRAPH abierto

El ajuste local tiene prioridad frente al ajuste global.

Nota

Los parámetros específicos de GRAPH que no están disponibles en el bloque de datos minimizado no se pueden insertar manualmente en la interfaz del bloque.

Definir el modelo de espacio de memoria globalmente para todos los bloques de función GRAPH nuevos

Para definir el modelo de espacio de memoria globalmente para todos los bloques de función GRAPH nuevos, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Si desea utilizar un bloque de datos minimizado, active la opción "Generar DB minimizado".
4. Si desea utilizar un bloque de datos con el modelo de espacio de memoria estándar, desactive la opción "Generar DB minimizado".

Definir el modelo de espacio de memoria localmente para un bloque de función GRAPH abierto

Para definir el modelo de espacio de memoria localmente para un bloque de función GRAPH abierto, proceda del siguiente modo:

1. Abra la ficha "Propiedades" en la ventana de inspección.
2. Seleccione la entrada "Atributos" en la navegación local de la ventana de inspección.
3. Si desea utilizar un bloque de datos minimizado, active la opción "Generar DB minimizado" en el grupo "Compilar".
4. Si desea utilizar un bloque de datos con el modelo de espacio de memoria estándar, desactive la opción "Generar DB minimizado" en el grupo "Compilar".

La interfaz del bloque GRAPH se adapta.

Consulte también

Principios básicos de la interfaz de un bloque de función GRAPH (Página 2848)

Sinopsis de la configuración de GRAPH (Página 2864)

Modificar la configuración (Página 2866)

Definir el lenguaje de programación en segmentos (Página 2867)

Definir el juego de parámetros de interfaz (Página 2868)

Modificar ajustes para parámetros internos (Página 2870)

Modificar ajustes para parámetros internos

Los parámetros internos de un bloque GRAPH pueden declararse visibles y accesibles desde HMI. En las CPU de la serie S7-1500, los parámetros internos pueden declararse además remanentes. Si los parámetros internos se declaran remanentes, sus valores se conservan incluso tras un cambio de estado operativo de "STOP" a "RUN", p. ej., después de un corte de alimentación. Como consecuencia, la cadena secuencial permanece en el estado actual y, después del cambio a "RUN", continúa con la ejecución en el punto en el que se interrumpió.

Los parámetros que haya declarado el propio usuario en la interfaz de un bloque GRAPH no se ven afectados por estos ajustes.

Declarar parámetros internos visibles y accesibles desde HMI

Para declarar todos los parámetros internos de un bloque GRAPH visibles y accesibles desde HMI, proceda del siguiente modo:

1. Abra el bloque GRAPH cuyo ajuste desea modificar.
2. Active la casilla de verificación "Parámetros internos visibles/accesibles desde HMI" en el menú "Edición". Desactive esta casilla de verificación si desea quitar este ajuste.

Declarar parámetros internos remanentes (solo S7-1500)

Para declarar remanentes todos los parámetros internos de un bloque GRAPH, proceda del siguiente modo:

1. Abra el bloque GRAPH cuyo ajuste desea modificar.
2. Active la casilla de verificación "Parámetros internos remanentes" en el menú "Edición". Desactive esta casilla de verificación si desea quitar este ajuste.

Consulte también

Sinopsis de la configuración de GRAPH (Página 2864)

Modificar la configuración (Página 2866)

Definir el lenguaje de programación en segmentos (Página 2867)

Definir el juego de parámetros de interfaz (Página 2868)

Definir el modelo de espacio de memoria para el bloque de datos (Página 2869)

La ventana de programación de GRAPH

Vista general de la ventana de programación

Función

La ventana de programación es el área en la que se crea el programa GRAPH. Además del área de trabajo propiamente dicha, contiene un área de navegación que facilita el acceso a todas las áreas del bloque de función GRAPH.

Las tareas siguientes se realizan en la ventana de programación:

- programar instrucciones permanentes anteriores y posteriores
- programar las cadenas secuenciales
- definir los avisos de interlock y supervisión

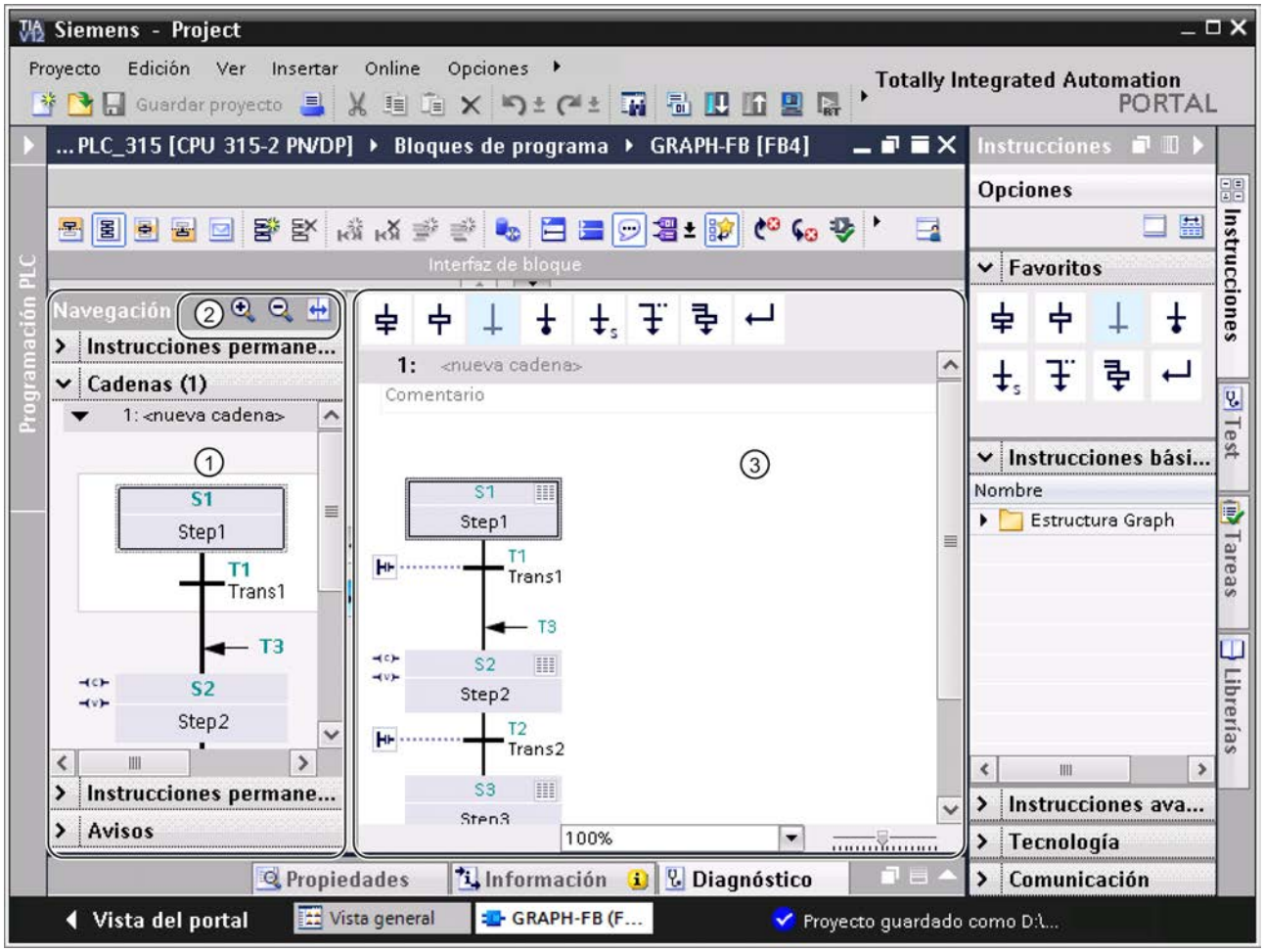
En función de lo que debe programarse es posible cambiar entre las vistas siguientes:

- Instrucciones permanentes anteriores (Página 2873)
- Vista de cadena (Página 2874)
- Vista de etapa individual (Página 2876)
- Instrucciones permanentes posteriores (Página 2878)
- Vista de avisos (Página 2879)

El área de trabajo así como las instrucciones y los Favoritos ofrecidos varían en función de la vista.

Estructura de la ventana de programación

La figura siguiente muestra la ventana de programación de GRAPH en la vista de cadena:



① Navegación

② Barra de herramientas de navegación

③ Área de trabajo

Navegación

La navegación contiene las paletas siguientes, desde las que se abren las vistas que se indican a continuación:

- Instrucciones permanentes anteriores
- Vista de cadena
- Instrucciones permanentes posteriores
- Vista de avisos

Adicionalmente, la navegación muestra vistas generales gráficas sobre las instrucciones permanentes y las cadenas secuenciales, y ofrece posibilidades de edición básicas por medio del menú contextual.

Haciendo doble clic en una etapa dentro de la navegación se cambia entre las vistas de cadena y etapa individual.

Barra de herramientas de navegación

Mediante la barra de herramientas de navegación pueden realizarse las siguientes acciones:

- Aumentar o reducir el tamaño de los elementos dentro de la navegación
  Utilizando los botones de ampliación y reducción se aumentan y reducen los elementos gráficos de las instrucciones permanentes y cadenas dentro de la navegación. El factor de zoom puede ajustarse individualmente para cada paleta. Al cerrar el bloque de función GRAPH se rechaza el cambio del factor de zoom. Si desea conservar el factor de zoom debe pulsar el botón "Memorizar disposición".
- Sincronizar la navegación
  Cuando el botón está activo, la navegación y el área de trabajo se sincronizan entre sí, de forma que siempre se muestran los mismos elementos. Si desactiva el botón, podrán mostrarse diferentes objetos en la navegación y en el área de trabajo.

Área de trabajo

En el área de trabajo se programan los componentes individuales de la cadena secuencial. Esto permite ver el programa GRAPH desde diferentes vistas. Dichas vistas pueden escalarse utilizando la función de zoom.

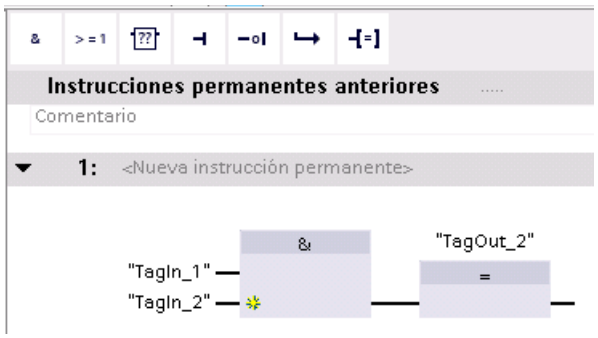
Vista "Instrucciones permanentes anteriores"

Función de la vista "Instrucciones permanentes anteriores"

En la vista "Instrucciones permanentes anteriores" se programan instrucciones que se ejecutan antes de procesar la cadena secuencial.

Estructura de la vista "Instrucciones permanentes anteriores"

La figura siguiente muestra un ejemplo de la vista "Instrucciones permanentes anteriores" utilizando el lenguaje de programación FUP:



Consulte también

Instrucciones permanentes (Página 2846)

Vista general de la ventana de programación (Página 2871)

Vista de cadena (Página 2874)

Vista de etapa individual (Página 2876)

Vista "Instrucciones permanentes posteriores" (Página 2878)

Vista de avisos (Página 2879)

Vista de cadena

Función de la vista de cadena

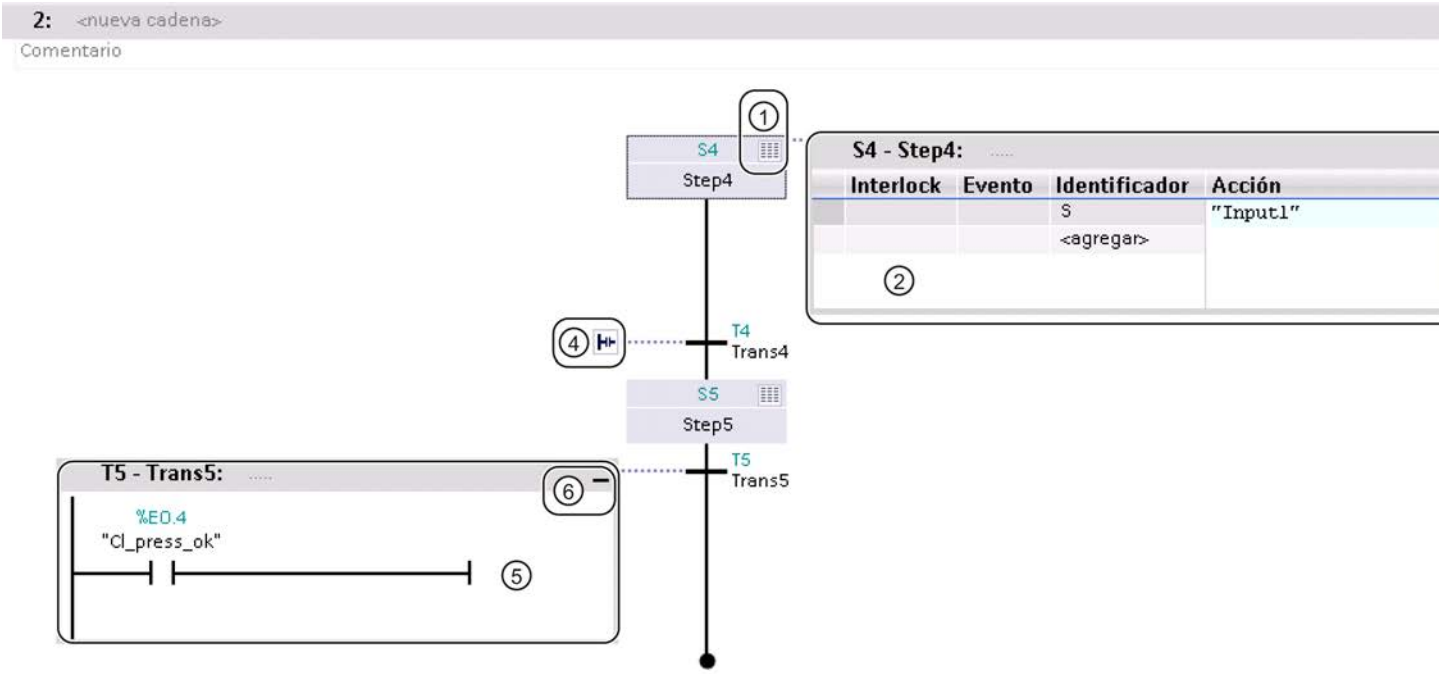
La vista de cadena muestra una vista general de la estructura de la cadena secuencial y permite ampliarla con los elementos siguientes:

- etapas
- transiciones
- operaciones de salto
- ramas
- fines de cadena

Además, es posible ampliar etapas y transiciones con un clic del ratón para visualizar o editar acciones de una etapa y condiciones de una transición.

Estructura de la vista de cadena

La figura siguiente muestra un ejemplo de la vista de cadena:



① Botón para ampliar o reducir una etapa

② Etapa en la vista avanzada

③ Botón para reducir una etapa

④ Botón para ampliar una transición

⑤ Transición en la vista avanzada

⑥ Botón para reducir una transición

Consulte también

Vista general de la ventana de programación (Página 2871)

Vista "Instrucciones permanentes anteriores" (Página 2873)

Vista de etapa individual (Página 2876)

Vista "Instrucciones permanentes posteriores" (Página 2878)

Vista de avisos (Página 2879)

Vista de etapa individual

Función de la vista de etapa individual

La vista de etapa individual ofrece la posibilidad de programar los elementos siguientes de una etapa:

- Interlock
- Supervisión
- Acciones
- Transición

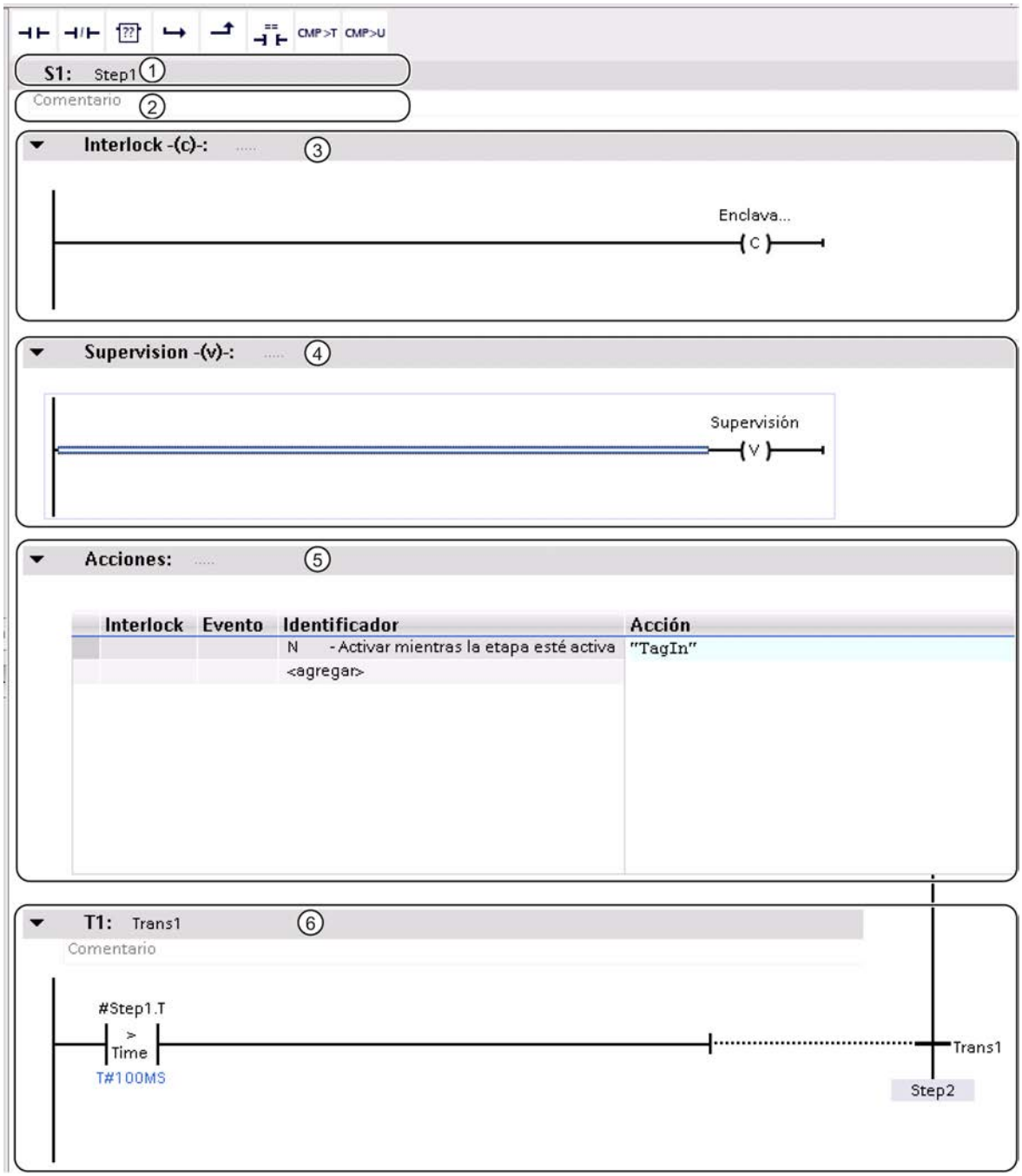
Además, es posible definir el título de la etapa e introducir un comentario sobre la misma.

Nota

Las acciones de la etapa y la transición correspondiente puede programarse también en la vista de cadena.

Estructura de la vista de etapa individual

La figura siguiente muestra una etapa en la vista de etapa individual:



① Título de la etapa

② Comentario sobre la etapa

③ Interlock

④ Supervisión

⑤ Acciones

⑥ Transición

Consulte también

Elementos de una etapa (Página 2832)

Vista general de la ventana de programación (Página 2871)

Vista "Instrucciones permanentes anteriores" (Página 2873)

Vista de cadena (Página 2874)

Vista "Instrucciones permanentes posteriores" (Página 2878)

Vista de avisos (Página 2879)

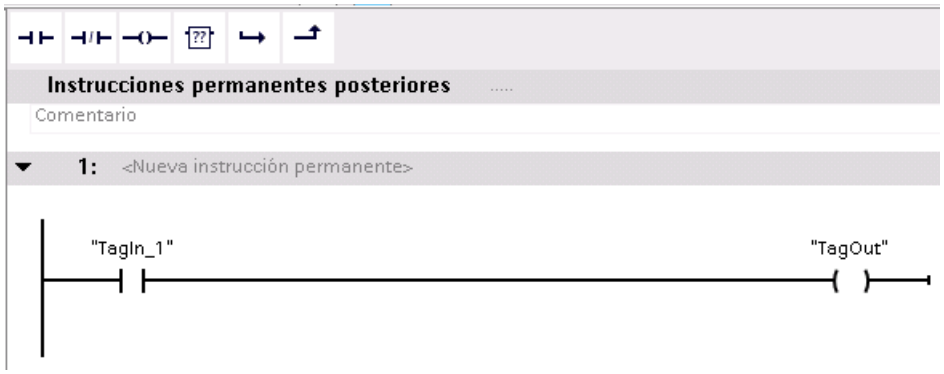
Vista "Instrucciones permanentes posteriores"

Función de la vista "Instrucciones permanentes posteriores"

En la vista "Instrucciones permanentes posteriores" se programan instrucciones que se ejecutan después de procesar la cadena secuencial.

Estructura de la vista "Instrucciones permanentes posteriores"

La figura siguiente muestra un ejemplo de la vista "Instrucciones permanentes posteriores" utilizando el lenguaje de programación KOP:



Consulte también

Instrucciones permanentes (Página 2846)

Vista general de la ventana de programación (Página 2871)

Vista "Instrucciones permanentes anteriores" (Página 2873)

Vista de cadena (Página 2874)

Vista de etapa individual (Página 2876)

Vista de avisos (Página 2879)

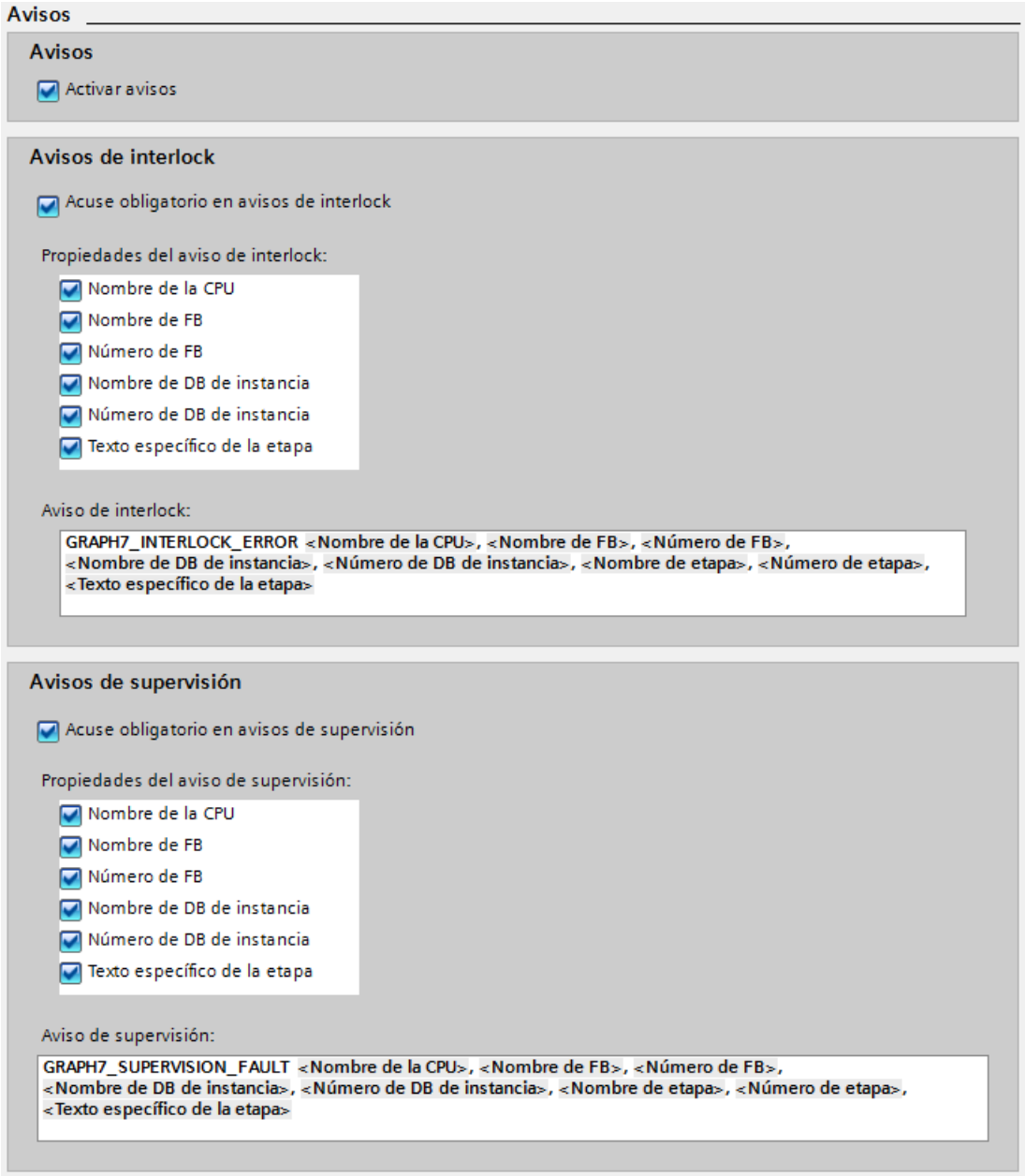
Vista de avisos

Función de la vista de avisos

En la vista de avisos se activa el visor de avisos y se definen avisos de interlock y supervisión. También es posible especificar si los avisos deben acusarse.

Estructura de la vista de avisos

La figura siguiente muestra la vista de avisos con los textos estándar de interlock y supervisión:



Consulte también

Definir avisos de interlock y supervisión (Página 2926)

Vista general de la ventana de programación (Página 2871)

Vista "Instrucciones permanentes anteriores" (Página 2873)

Vista de cadena (Página 2874)

Vista de etapa individual (Página 2876)

Vista "Instrucciones permanentes posteriores" (Página 2878)

Utilizar Favoritos en GRAPH

Agregar elementos GRAPH a los Favoritos

Requisitos

- Hay un bloque de función GRAPH abierto en una de las vistas siguientes:
  - Instrucciones permanentes anteriores
  - Vista de cadena
  - Vista de etapa individual
  - Instrucciones permanentes posteriores
- La Task Card "Instrucciones" tiene configurado el modo multipaleta, o bien los Favoritos se muestran también en la ventana de programación GRAPH.

Procedimiento

Para agregar instrucciones a los Favoritos, proceda del siguiente modo:

1. Abra la Task Card "Instrucciones".
2. Maximice la paleta "Instrucciones básicas".
3. En la paleta "Instrucciones básicas", navegue hasta la instrucción que desea agregar a los Favoritos.
4. Arrastre la instrucción hasta la paleta "Favoritos" o hasta el área de Favoritos de la ventana de programación GRAPH mediante Drag & Drop.

Nota

Para mostrar los Favoritos en la ventana de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Insertar elementos GRAPH mediante Favoritos (Página 2882)

Quitar elementos GRAPH de los Favoritos (Página 2882)

Insertar elementos GRAPH mediante Favoritos

Requisitos

- Hay un bloque de función GRAPH abierto en una de las vistas siguientes:
  - Instrucciones permanentes anteriores
  - Vista de cadena
  - Vista de etapa individual
  - Instrucciones permanentes posteriores
- Existen Favoritos.

Procedimiento

Para insertar una instrucción en un programa desde los Favoritos, proceda del siguiente modo:

1. Arrastre la instrucción deseada mediante Drag & Drop desde los Favoritos hasta la posición deseada.

O:

1. Marque el punto del programa en el que quiere insertar la instrucción.
2. En los Favoritos, haga clic en la instrucción que desea insertar.

Nota

Para mostrar los Favoritos en la ventana de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Agregar elementos GRAPH a los Favoritos (Página 2881)

Quitar elementos GRAPH de los Favoritos (Página 2882)

Quitar elementos GRAPH de los Favoritos

Requisitos

- Hay un bloque de función GRAPH abierto en una de las vistas siguientes:
  - Instrucciones permanentes anteriores
  - Vista de cadena
  - Vista de etapa individual
  - Instrucciones permanentes posteriores
- Existen Favoritos.

Procedimiento

Para eliminar instrucciones de los Favoritos, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la instrucción que desea eliminar dentro del área de Favoritos.
2. Elija el comando "Eliminar instrucción" del menú contextual.

Nota

Para mostrar los Favoritos en la ventana de programación, haga clic en el botón "Mostrar Favoritos también en el editor" de la barra de herramientas del editor de programación.

Consulte también

Agregar elementos GRAPH a los Favoritos (Página 2881)

Insertar elementos GRAPH mediante Favoritos (Página 2882)

Programar la estructura de una cadena secuencial

Insertar cadenas secuenciales

Si se abre un bloque de función GRAPH, la paleta "Cadenas" se expande en la navegación. En un bloque de función GRAPH nuevo ya existe una cadena secuencial con una etapa y una transición. Es posible insertar más cadenas secuenciales en todo momento.

Requisitos

Hay un bloque de función GRAPH abierto.

Procedimiento

Para insertar una nueva cadena secuencial, proceda del siguiente modo:

1. En la navegación, haga clic con el botón derecho del ratón en la barra de título de la paleta "Cadenas" o en la barra de título de una cadena secuencial ya existente.
2. Elija el comando "Insertar cadena" del menú contextual.
   La nueva cadena secuencial se insertará después de la seleccionada. Si no hay ninguna cadena secuencial seleccionada, se insertará delante de la primera.

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)

Elementos de una cadena secuencial (Página 2828)

Cambiar el nombre de cadenas secuenciales (Página 2884)

Comentar cadenas secuenciales (Página 2884)

Insertar elementos en una cadena secuencial (Página 2885)
Finalizar cadenas secuenciales (Página 2886)
Borrar cadenas secuenciales (Página 2887)
Utilizar etapas y transiciones (Página 2887)
Utilizar ramas (Página 2896)
Utilizar saltos (Página 2899)

Cambiar el nombre de cadenas secuenciales

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Procedimiento

Para cambiar el nombre de una cadena secuencial, proceda del siguiente modo:

1. Haga clic en la barra de título de la cadena secuencial dentro de la vista de cadena.
   De manera estándar aparece el comodín "<Nueva cadena>".
2. Introduzca el nuevo nombre para la cadena secuencial.

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)
Elementos de una cadena secuencial (Página 2828)
Insertar cadenas secuenciales (Página 2883)
Comentar cadenas secuenciales (Página 2884)
Insertar elementos en una cadena secuencial (Página 2885)
Finalizar cadenas secuenciales (Página 2886)
Borrar cadenas secuenciales (Página 2887)
Utilizar etapas y transiciones (Página 2887)
Utilizar ramas (Página 2896)
Utilizar saltos (Página 2899)

Comentar cadenas secuenciales

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Procedimiento

Para introducir un comentario sobre una cadena secuencial, proceda del siguiente modo:

1. Haga clic en el área de comentario de la cadena secuencial dentro de la vista de cadena. De manera estándar aparece el comodín "Comentario".
2. Introduzca un comentario sobre la cadena secuencial.

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)

Elementos de una cadena secuencial (Página 2828)

Insertar cadenas secuenciales (Página 2883)

Cambiar el nombre de cadenas secuenciales (Página 2884)

Insertar elementos en una cadena secuencial (Página 2885)

Finalizar cadenas secuenciales (Página 2886)

Borrar cadenas secuenciales (Página 2887)

Utilizar etapas y transiciones (Página 2887)

Utilizar ramas (Página 2896)

Utilizar saltos (Página 2899)

Insertar elementos en una cadena secuencial

Requisitos

Hay una cadena secuencial abierta en la vista de cadena.

Procedimiento

Para insertar elementos en una cadena secuencial, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la posición de la cadena secuencial en la que desea insertar el elemento nuevo.
2. En el submenú "Insertar elemento" del menú contextual, elija el elemento que desea insertar. Solo será posible seleccionar los elementos que se admiten en la posición seleccionada.

O:

1. Abra la paleta "Instrucciones básicas" de la Task Card "Instrucciones".
2. Abra la carpeta "Estructura Graph".
3. Arrastre el elemento que desea insertar hasta la posición deseada de la cadena secuencial mediante Drag & Drop. Tenga en cuenta que no es posible insertar todos los elementos en todas las posiciones. Las posiciones permitidas se muestran con un pequeño cuadrado verde.

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)

Elementos de una cadena secuencial (Página 2828)

Insertar cadenas secuenciales (Página 2883)

Cambiar el nombre de cadenas secuenciales (Página 2884)

Comentar cadenas secuenciales (Página 2884)

Finalizar cadenas secuenciales (Página 2886)

Borrar cadenas secuenciales (Página 2887)

Utilizar etapas y transiciones (Página 2887)

Utilizar ramas (Página 2896)

Utilizar saltos (Página 2899)

Finalizar cadenas secuenciales

Todas las cadenas secuenciales deben finalizarse con un salto a otra etapa dentro de la misma cadena secuencial u otra distinta o bien con el elemento "fin de cadena". Con un salto a otra etapa se consigue un procesamiento cíclico de la cadena secuencial.

Requisitos

Hay una cadena secuencial abierta en la vista de cadena.

Procedimiento

Para finalizar una cadena, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la flecha doble situada al final de la cadena secuencial.
2. En el submenú "Insertar elemento" del menú contextual, elija el elemento "Fin de cadena".

O:

1. Haga clic con el botón derecho del ratón en la flecha doble situada al final de la cadena secuencial.
2. En el submenú "Insertar elemento" del menú contextual, elija el elemento "Salto".
   Se abre una vista general de todas las etapas existentes del bloque de función GRAPH.
3. Elija una etapa como destino de salto.

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)

Elementos de una cadena secuencial (Página 2828)

Insertar cadenas secuenciales (Página 2883)

Cambiar el nombre de cadenas secuenciales (Página 2884)

Comentar cadenas secuenciales (Página 2884)

Insertar elementos en una cadena secuencial (Página 2885)

Borrar cadenas secuenciales (Página 2887)

Utilizar etapas y transiciones (Página 2887)

Utilizar ramas (Página 2896)

Utilizar saltos (Página 2899)

Borrar cadenas secuenciales

Las cadenas secuenciales pueden volver a borrarse en cualquier momento. Sin embargo, en el bloque de función GRAPH debe permanecer una cadena secuencial como mínimo. Al borrar cadenas secuenciales tenga en cuenta que es posible que también se borren destinos de saltos.

Procedimiento

Para borrar una cadena secuencial, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la barra de título de la cadena secuencial dentro de la navegación.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Principios básicos de las cadenas secuenciales (Página 2827)

Elementos de una cadena secuencial (Página 2828)

Insertar cadenas secuenciales (Página 2883)

Cambiar el nombre de cadenas secuenciales (Página 2884)

Comentar cadenas secuenciales (Página 2884)

Insertar elementos en una cadena secuencial (Página 2885)

Finalizar cadenas secuenciales (Página 2886)

Utilizar etapas y transiciones (Página 2887)

Utilizar ramas (Página 2896)

Utilizar saltos (Página 2899)

Utilizar etapas y transiciones

Insertar etapas y transiciones

Cuando se inserta un bloque de función GRAPH nuevo en el proyecto, ya contiene una cadena secuencial con una etapa y una transición. Es posible insertar más etapas y transiciones en la cadena secuencial.

Tenga en cuenta las reglas siguientes al insertar etapas y transiciones:

- Las etapas y transiciones deben intercalarse siempre. Esto significa que no puede haber dos etapas o dos transiciones consecutivas.
- Por consiguiente, solo es posible insertar etapas y transiciones al final de una rama o en lugares en los que falta una etapa o una transición.
- Una rama simultánea debe empezar siempre con una etapa.
- Una rama alternativa debe empezar siempre con una transición.
- Es posible insertar una etapa y una transición conjuntamente como pareja. Puesto que de este modo queda garantizado que la etapa está seguida por una transición, puede hacerse en cualquier lugar de la cadena secuencial.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Insertar etapa

Para insertar una etapa, proceda del siguiente modo:

1. En la cadena secuencial, haga clic con el botón derecho del ratón en un lugar en el que esté permitido insertar una etapa.
2. Elija el comando "Insertar elemento > Etapa" del menú contextual.

O:

1. Arrastre mediante Drag & Drop el elemento "Etapa" desde los Favoritos o la Task Card "Instrucciones" hasta la posición correspondiente de la cadena secuencial.

Insertar transición

Para insertar una transición, proceda del siguiente modo:

1. En la cadena secuencial, haga clic con el botón derecho del ratón en un lugar en el que esté permitido insertar una transición.
2. Elija el comando "Insertar elemento > Transición" del menú contextual.

O:

1. Arrastre mediante Drag & Drop el elemento "Transición" desde los Favoritos o la Task Card "Instrucciones" hasta la posición correspondiente de la cadena secuencial.

Insertar etapa y transición conjuntamente como pareja

Para insertar una etapa y una transición conjuntamente como pareja, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la posición de la cadena secuencial en la que desea insertar la etapa y la transición.
2. Elija el comando "Insertar elemento > Etapa y transición" del menú contextual.

O:

1. Arrastre mediante Drag & Drop el elemento "Etapa y transición" desde los Favoritos o la Task Card "Instrucciones" hasta la posición correspondiente de la cadena secuencial.

Consulte también

Etapas y transiciones (Página 2831)

Establecer y desmarcar una etapa inicial (Página 2889)

Renumerar las etapas y transiciones (Página 2890)

Renombrar las etapas y transiciones (Página 2892)

Comentar etapas (Página 2893)

Copiar y pegar etapas y transiciones (Página 2894)

Saltar etapas (Página 2895)

Borrar etapas y transiciones (Página 2895)

Establecer y desmarcar una etapa inicial

Cuando se inserta un bloque de función GRAPH nuevo en el proyecto, ya contiene una cadena secuencial con una etapa y una transición. La etapa está establecida de manera estándar como etapa inicial. Esto significa que el procesamiento de la cadena secuencial debe empezar siempre con esta etapa. En cualquier momento es posible definir otra etapa como inicial. En las ramas simultáneas también se admiten varias etapas iniciales. El número máximo de etapas iniciales es de 250. Una etapa inicial puede convertirse de nuevo en una etapa normal desmarcándola.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Establecer la etapa inicial

Para establecer una etapa como inicial, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la etapa que desea establecer como inicial.
2. En el menú contextual, active la casilla de verificación "Etapa inicial".

Desmarcar etapa inicial

Para desmarcar una etapa inicial y convertirla de nuevo en una etapa normal, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la etapa que desea establecer como inicial.
2. En el menú contextual, desactive la casilla de verificación "Etapa inicial".

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Renumerar las etapas y transiciones (Página 2890)

Renombrar las etapas y transiciones (Página 2892)

Comentar etapas (Página 2893)

Copiar y pegar etapas y transiciones (Página 2894)

Saltar etapas (Página 2895)

Borrar etapas y transiciones (Página 2895)

Renumerar las etapas y transiciones

Cuando se insertan etapas y transiciones en la cadena secuencial, estas reciben como número el siguiente valor libre. Es posible modificar este número posteriormente. Para ello existen las dos posibilidades siguientes:

- Asignar manualmente números nuevos a las etapas y transiciones que desee renumerar. Al hacerlo, tenga en cuenta que no puede haber números duplicados.
- Realizar la renumeración con la ayuda del programa. Esto permite renumerar automáticamente todas las etapas y transiciones de una rama, una cadena secuencial o un bloque. Tenga en cuenta las siguientes diferencias:
  - Renumerar dentro de una rama:
    Dentro de la rama se renumeran todas las etapas o transiciones a partir de la etapa o transición seleccionada. La renumeración finaliza cuando empieza una rama nueva.
  - Renumerar dentro de una cadena secuencial:
    Se renumeran todas las etapas o transiciones dentro de la cadena secuencial. Las primera etapa o transición de la cadena secuencial obtiene el número inicial que se ha introducido. Tenga en cuenta que al realizar la renumeración es posible que también se cambien los números de etapas en otras cadenas secuenciales para que no se den números duplicados.
  - Renumerar dentro de un bloque:
    Se renumeran todas las etapas o transiciones dentro del bloque GRAPH. Las primera etapa o transición dentro de la primera cadena secuencial del bloque obtiene el número inicial que se ha introducido.

También es posible renumerar solo las etapas o solo las transiciones.

Nota

Sin embargo, tenga en cuenta que esta opción no está disponible para bloques de función GRAPH que tienen activado el atributo "Generar DB minimizado".

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Renumerar manualmente las etapas y transiciones

Para renumerar manualmente las etapas y transiciones, proceda del siguiente modo:

1. Seleccione la etapa cuyo número desea cambiar.
2. Haga clic en el número de la etapa.
3. Introduzca el nuevo número.

Renumerar etapas y transiciones dentro de una rama con la ayuda del programa

Para renumerar las etapas y transiciones dentro de una rama con la ayuda del programa, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la etapa o transición a partir de la que desea renumerar la rama.
2. Elija el comando "Renumerar" del menú contextual.
   Se abre el cuadro de diálogo "Renumerar".
3. Si desea renumerar las etapas, active la casilla de verificación "Etapas" e introduzca el número que debe recibir la etapa seleccionada.
4. Si desea renumerar las transiciones, active la casilla de verificación "Transiciones" e introduzca el número que debe recibir la transición posterior a la etapa seleccionada.
5. Seleccione el botón de opción "Rama" en el área "Renumerar a".
6. Confirme con "Aceptar".

Renumerar etapas y transiciones dentro de una cadena secuencial con la ayuda del programa

Para renumerar las etapas y transiciones dentro de una cadena secuencial con la ayuda del programa, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una etapa o transición dentro de la cadena secuencial para la que desea realizar la renumeración.
2. Elija el comando "Renumerar" del menú contextual.
   Se abre el cuadro de diálogo "Renumerar".
3. Si desea renumerar las etapas, active la casilla de verificación "Etapas" e introduzca el número que debe recibir la primera etapa de la cadena secuencial.
4. Si desea renumerar las transiciones, active la casilla de verificación "Transiciones" e introduzca el número que debe recibir la primera transición de la cadena secuencial.
5. Seleccione el botón de opción "Cadena" en el área "Renumerar a".
6. Confirme con "Aceptar".

Renumerar etapas y transiciones dentro de un bloque de función GRAPH con la ayuda del programa

Para renumerar las etapas y transiciones dentro de un bloque de función GRAPH con la ayuda del programa, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una etapa o transición dentro del bloque de función GRAPH para el que desea realizar la renumeración.
2. Elija el comando "Renumerar" del menú contextual.
   Se abre el cuadro de diálogo "Renumerar".
3. Si desea renumerar las etapas, active la casilla de verificación "Etapas" e introduzca el número que debe recibir la primera etapa del bloque de función GRAPH.
4. Si desea renumerar las transiciones, active la casilla de verificación "Transiciones" e introduzca el número que debe recibir la primera transición del bloque de función GRAPH.
5. Seleccione el botón de opción "Bloque" en el área "Renumerar a".
6. Confirme con "Aceptar".

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Establecer y desmarcar una etapa inicial (Página 2889)

Renombrar las etapas y transiciones (Página 2892)

Comentar etapas (Página 2893)

Copiar y pegar etapas y transiciones (Página 2894)

Saltar etapas (Página 2895)

Borrar etapas y transiciones (Página 2895)

Renombrar las etapas y transiciones

Cuando se insertan etapas y transiciones en la cadena secuencial, en un principio reciben el nombre "Step<número de etapa>" y "Trans<número de transición>". Los nombres pueden modificarse para que el programa sea más claro. Al hacerlo, tenga en cuenta que no puede haber nombres duplicados.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Renombrar una etapa

Para renombrar una etapa, proceda del siguiente modo:

1. Seleccione la etapa cuyo nombre desea cambiar.
2. Haga clic en el nombre de la etapa.
3. Introduzca un nombre nuevo.

Renombrar una transición

Para cambiar el nombre de una transición, proceda del siguiente modo:

1. Haga doble clic en el nombre la transición que desee cambiar.
2. Introduzca un nombre nuevo.

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Establecer y desmarcar una etapa inicial (Página 2889)

Renumerar las etapas y transiciones (Página 2890)

Comentar etapas (Página 2893)

Copiar y pegar etapas y transiciones (Página 2894)

Saltar etapas (Página 2895)

Borrar etapas y transiciones (Página 2895)

Comentar etapas

Existe la posibilidad de comentar etapas. Esto permite describir la finalidad de la etapa o bien llamar la atención sobre peculiaridades, por ejemplo.

Requisitos

Hay una etapa abierta en la vista de etapa individual.

Procedimiento

Para comentar una etapa, proceda del siguiente modo:

1. En el área de comentario, haga clic en "Comentario".
   Se borra el texto "Comentario" y aparece una marca de inserción.
2. Introduzca el comentario.

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Establecer y desmarcar una etapa inicial (Página 2889)

Renumerar las etapas y transiciones (Página 2890)

Renombrar las etapas y transiciones (Página 2892)

Copiar y pegar etapas y transiciones (Página 2894)

Saltar etapas (Página 2895)

Borrar etapas y transiciones (Página 2895)

Copiar y pegar etapas y transiciones

Es posible copiar etapas y transiciones y pegarlas en la misma cadena secuencial u otra distinta. De este modo se reduce el trabajo de programación cuando una etapa nueva debe realizar tareas similares a las de una etapa ya existente. No obstante, tenga en cuenta las particularidades siguientes:

- Al copiar etapas y transiciones también debe observarse la secuencia de etapa y transición. Por este motivo se insertan comodines ("Missing Step", "Missing Transition") si hubiera dos etapas o transiciones consecutivas después del proceso de inserción.
- Se cambian el nombre y el número del elemento insertado para evitar que estén duplicados.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Copiar etapas y transiciones

Para copiar una etapa o una transición, proceda del siguiente modo:

1. Para copiar una etapa, haga clic con el botón derecho del ratón en la etapa que desea copiar. Para copiar una transición, haga clic con el botón derecho del ratón en la transición que desea copiar.
2. Elija el comando "Copiar" del menú contextual.

Insertar etapas y transiciones

Para insertar una etapa o una transición, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la posición de la cadena secuencial en la que desea insertar la etapa o la transición.
2. Elija el comando "Pegar" del menú contextual.
3. Sustituya los comodines por elementos válidos, si fuera el caso.

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Establecer y desmarcar una etapa inicial (Página 2889)

Renumerar las etapas y transiciones (Página 2890)

Renombrar las etapas y transiciones (Página 2892)

Comentar etapas (Página 2893)

Saltar etapas (Página 2895)

Borrar etapas y transiciones (Página 2895)

Saltar etapas

Existe la posibilidad de impedir que se ejecuten etapas en las que las transiciones anterior y posterior se validan al mismo tiempo. Sin embargo, en las CPUs de la serie S7-300/400 tenga en cuenta que esta opción no puede utilizarse junto con la opción "Generar DB minimizado".

Procedimiento

Para saltar etapas para las que las transiciones anterior y posterior se validan al mismo tiempo, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Active la opción "Saltar etapas" en "Propiedades de cadena".

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Establecer y desmarcar una etapa inicial (Página 2889)

Renumerar las etapas y transiciones (Página 2890)

Renombrar las etapas y transiciones (Página 2892)

Comentar etapas (Página 2893)

Copiar y pegar etapas y transiciones (Página 2894)

Borrar etapas y transiciones (Página 2895)

Borrar etapas y transiciones

Es posible eliminar de nuevo etapas y transiciones de la cadena secuencial. No obstante, tenga en cuenta las particularidades siguientes:

- Al borrar etapas y transiciones también debe observarse la secuencia de etapa y transición. Por este motivo se insertan comodines ("Missing Step", "Missing Transition") si hubiera dos etapas o transiciones consecutivas después del proceso de borrado.
- No es posible borrar todas las etapas de una cadena secuencial, pues esta debe tener siempre una etapa como mínimo.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Procedimiento

Para borrar una etapa o una transición, proceda del siguiente modo:

1. Para borrar una etapa, haga clic con el botón derecho del ratón en la etapa que desea borrar. Para borrar una transición, haga clic con el botón derecho del ratón en la transición que desea borrar.
2. Elija el comando "Borrar" del menú contextual.
3. Sustituya los comodines por elementos válidos, si fuera el caso.

Consulte también

Etapas y transiciones (Página 2831)

Insertar etapas y transiciones (Página 2887)

Establecer y desmarcar una etapa inicial (Página 2889)

Renumerar las etapas y transiciones (Página 2890)

Renombrar las etapas y transiciones (Página 2892)

Comentar etapas (Página 2893)

Copiar y pegar etapas y transiciones (Página 2894)

Saltar etapas (Página 2895)

Utilizar ramas

Crear ramas alternativas

Requisitos

Hay una cadena secuencial abierta en la vista de cadena.

Procedimiento

Para crear una rama alternativa, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la etapa tras la cual desea insertar una rama alternativa.
2. En el submenú "Insertar elemento" del menú contextual, elija el comando "Abrir rama alternativa".

O:

1. Abra la paleta "Instrucciones básicas" de la Task Card "Instrucciones".
2. Abra la carpeta "Estructura Graph".
3. Arrastre el elemento "Rama alternativa" hasta la posición deseada de la cadena secuencial mediante Drag & Drop. Las posiciones permitidas se muestran con un cuadrado verde.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Crear una rama simultánea (Página 2897)

Cerrar ramas (Página 2897)

Borrar ramas (Página 2898)

Crear una rama simultánea

Requisitos

Hay una cadena secuencial abierta en la vista de cadena.

Procedimiento

Para crear una rama simultánea, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la transición tras la cual desea insertar una rama simultánea.
2. En el submenú "Insertar elemento" del menú contextual, elija el comando "Abrir rama simultánea".

O:

1. Abra la paleta "Instrucciones básicas" de la Task Card "Instrucciones".
2. Abra la carpeta "Estructura Graph".
3. Arrastre el elemento "Rama simultánea" hasta la posición deseada de la cadena secuencial mediante Drag & Drop. Las posiciones permitidas se muestran con un cuadrado verde.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Crear ramas alternativas (Página 2896)

Cerrar ramas (Página 2897)

Borrar ramas (Página 2898)

Cerrar ramas

Las ramas simultáneas y alternativas se cierran del siguiente modo:

- Pueden reagruparse con su rama matriz.
- Pueden cerrarse con un fin de cadena.
- Pueden cerrarse con un salto.

Sin embargo, en una rama simultánea solo es posible insertar un fin de cadena o un salto tras una transición.

Requisitos

Hay una rama alternativa o simultánea.

Reagrupar ramas con la rama matriz

Para reagrupar una rama abierta con su rama matriz, proceda del siguiente modo:

1. Seleccione la flecha doble situada al final de la rama.
2. Pulse y mantenga pulsado el botón izquierdo del ratón.
   Al mover el puntero del ratón aparece una línea discontinua.
3. Arrastre la línea discontinua hasta una posición apropiada del segmento. Las posiciones permitidas se muestran con un cuadrado verde.
4. Suelte el botón izquierdo del ratón.

Cerrar ramas con un fin de cadena

Para cerrar una rama con un fin de cadena, proceda del siguiente modo:

1. En caso necesario, inserte una transición al final de la rama en una rama simultánea.
2. Haga clic con el botón derecho del ratón en la flecha doble situada al final de la rama.
3. En el submenú "Insertar elemento" del menú contextual, elija el elemento "Fin de cadena".

Cerrar ramas con un salto

Para cerrar una rama con un salto, proceda del siguiente modo:

1. En caso necesario, inserte una transición al final de la rama en una rama simultánea.
2. Haga clic con el botón derecho del ratón en la flecha doble situada al final de la rama.
3. En el submenú "Insertar elemento" del menú contextual, elija el elemento "Salto".
   Se abre una vista general de todas las etapas existentes del bloque de función GRAPH.
4. Elija una etapa como destino de salto.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Crear ramas alternativas (Página 2896)

Crear una rama simultánea (Página 2897)

Borrar ramas (Página 2898)

Borrar ramas

Requisitos

Hay una rama alternativa o simultánea.

Procedimiento

Para borrar una rama, proceda del siguiente modo:

1. Seleccione la rama que desea borrar.
2. Haga clic con el botón derecho del ratón en la rama seleccionada.
3. Elija el comando "Borrar" del menú contextual.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Crear ramas alternativas (Página 2896)

Crear una rama simultánea (Página 2897)

Cerrar ramas (Página 2897)

Utilizar saltos

Insertar saltos

Los saltos solo pueden insertarse al final de una cadena secuencial o una rama. En una rama simultánea, el último elemento debe ser una transición.

Requisitos

Hay una cadena secuencial abierta en la vista de cadena.

Procedimiento

Para insertar un salto en una cadena secuencial o una rama, proceda del siguiente modo:

1. En caso necesario, inserte una transición al final de la rama en una rama simultánea.
2. Haga clic con el botón derecho del ratón en la flecha doble situada al final de la cadena secuencial o rama.
3. En el submenú "Insertar elemento" del menú contextual, elija el elemento "Salto".
   Se abre una vista general de todas las etapas existentes del bloque de función GRAPH.
4. Elija una etapa como destino de salto.

Resultado

El salto se inserta. El salto y su destino se representan en la cadena secuencial en forma de flecha, de modo que en el destino del salto se indica la transición de origen y en el salto la etapa de destino.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Cambiar destino de salto (Página 2900)

Navegar hasta el destino de salto (Página 2900)

Borrar saltos (Página 2901)

Cambiar destino de salto

El destino de un salto se puede modificar a posteriori.

Requisitos

- Hay un bloque de función GRAPH abierto en la vista de cadena.
- Se ha creado un salto.

Procedimiento

Para cambiar el destino de salto a posteriori, proceda del siguiente modo:

1. Haga doble clic en el número de etapa del destino de salto actual.
   Se abre una vista general de todas las etapas existentes del bloque de función GRAPH.
2. Elija una etapa como nuevo destino de salto.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Insertar saltos (Página 2899)

Navegar hasta el destino de salto (Página 2900)

Borrar saltos (Página 2901)

Navegar hasta el destino de salto

Desde un salto es posible navegar hasta la etapa definida como destino del salto.

Requisitos

- Hay un bloque de función GRAPH abierto en la vista de cadena.
- Se ha creado un salto.

Procedimiento

Para navegar desde un salto hasta la etapa definida como destino, proceda del siguiente modo:

1. Haga clic en el salto con el botón derecho del ratón.
2. Elija el comando "Ir a > Destino de salto" del menú contextual.

Consulte también

Elementos de una cadena secuencial (Página 2828)
Insertar saltos (Página 2899)
Cambiar destino de salto (Página 2900)
Borrar saltos (Página 2901)

Borrar saltos

Requisitos

- Hay un bloque de función GRAPH abierto en la vista de cadena.
- Se ha creado un salto.

Procedimiento

Para borrar un salto, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en el salto que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Elementos de una cadena secuencial (Página 2828)
Insertar saltos (Página 2899)
Cambiar destino de salto (Página 2900)
Navegar hasta el destino de salto (Página 2900)

Programar acciones

Insertar acciones

Introducción

Existe la posibilidad de definir acciones que se ejecutan al procesar la etapa. Para comentar acciones o adaptar llamadas de bloque puede activarse una entrada de varias líneas. El ajuste

predeterminado es la entrada de una línea. Las acciones se insertan en la vista de etapa individual o en la vista de cadena.

Para tener una visión más clara de las acciones programadas es posible visualizar descripciones de los eventos e identificadores.

En las acciones se asigna a un operando el valor de otro operando o de una expresión matemática o bien el resultado de una instrucción. Utilice para ello la sintaxis siguiente:

- Asignar el valor de un operando: A:=B
- Asignar el valor de una expresión matemática: A:=B<operador>C
  Están disponibles los siguientes operandos: + (suma), - (resta), * (multiplicación), / (división), % (módulo), AND (operación lógica Y), OR (operación lógica O), XOR (operación lógica O exclusiva)
- Asignar el resultado de una instrucción: A:=<instrucción>(B)
  Las instrucciones disponibles se encuentran en la Task Card "Instrucciones".

Asegúrese de que los tipos de datos de los operandos sean compatibles entre sí.

Insertar acciones en la vista de etapa individual

Para insertar una acción en la vista de etapa individual, proceda del siguiente modo:

1. Abra la vista de etapa individual.
2. Si las acciones están contraídas, haga clic en la flecha situada delante de "Acciones". La flecha hacia la derecha se convierte en una flecha hacia abajo y aparece la tabla con las acciones.
3. Si desea activar la entrada de varias líneas, haga clic con el botón derecho del ratón en la fila de la nueva acción y active el comando "Permitir entrada de varias líneas" del menú contextual.
4. Si desea enlazar la nueva acción con un interlock, haga clic en la celda de la columna "Interlock" y elija la entrada "-(C)- - Interlock" de la lista desplegable.
5. Si desea enlazar la nueva acción con un evento, haga clic en la celda de la columna "Evento" y elija el evento correspondiente de la lista desplegable.

6. Haga clic en la celda de la columna "Identificador" y elija el identificador de la nueva acción en la lista desplegable.
7. Introduzca la acción que debe ejecutarse en la columna "Acción". Existen las siguientes posibilidades:
   – Reemplazar los comodines con los operandos o valores que desee utilizar para la acción. Estos se insertan con la función Drag & Drop o autocompletar.
   – Utilizar instrucciones de la Task Card "Instrucciones". Para ello, arrástrelas mediante Drag & Drop desde la Task Card hasta la columna "Acción".
   – Arrastrar bloques mediante Drag & Drop desde el árbol del proyecto hasta la columna "Acción" para abrirlos.

Nota

Al efectuar llamadas de instrucciones que dependen del tipo de datos se debe indicar un tipo de datos. El tipo de datos puede introducirse manualmente o bien seleccionarse de una lista desplegable haciendo clic en el pequeño rectángulo verde situado debajo del tipo de datos existente.

Insertar acciones en la vista de cadena

Para insertar una acción en la vista de cadena, proceda del siguiente modo:

1. Abra la vista de cadena.
2. Amplíe la etapa en la que deben insertarse acciones. Para ello, haga clic en el botón situado en el extremo superior derecho de la etapa o seleccione la etapa y pulse <Intro>.
3. Si desea activar la entrada de varias líneas, haga clic con el botón derecho del ratón en la fila de la nueva acción y active el comando "Permitir entrada de varias líneas" del menú contextual.
4. Si desea enlazar la nueva acción con un interlock, haga clic en la celda de la columna "Interlock" y elija la entrada "-(C)- - Interlock" de la lista desplegable.
5. Si desea enlazar la nueva acción con un evento, haga clic en la celda de la columna "Evento" y elija el evento correspondiente de la lista desplegable.

6. Haga clic en la celda de la columna "Identificador" y elija el identificador de la nueva acción en la lista desplegable.
7. Introduzca la acción que debe ejecutarse en la columna "Acción". Existen las siguientes posibilidades:
   – Reemplazar los comodines con los operandos o valores que desee utilizar para la acción. Estos se insertan con la función Drag & Drop o autocompletar.
   – Utilizar instrucciones de la Task Card "Instrucciones". Para ello, arrástrelas mediante Drag & Drop desde la Task Card hasta la columna "Acción".
   – Arrastrar bloques mediante Drag & Drop desde el árbol del proyecto hasta la columna "Acción" para abrirlos.

Nota

Al efectuar llamadas de instrucciones que dependen del tipo de datos se debe indicar un tipo de datos. El tipo de datos puede introducirse manualmente o bien seleccionarse de una lista desplegable haciendo clic en el pequeño rectángulo verde situado debajo del tipo de datos existente.

Mostrar la descripción de los eventos

Para visualizar la descripción de los eventos, proceda del siguiente modo:

1. Haga clic en la tabla de acciones con el botón derecho del ratón.
2. Active el comando "Mostrar descripción de los eventos" del menú contextual.

Mostrar la descripción de los identificadores

Para visualizar la descripción de los identificadores, proceda del siguiente modo:

1. Haga clic en la tabla de acciones con el botón derecho del ratón.
2. Active el comando "Mostrar descripción de los identificadores" del menú contextual.

Consulte también

Acciones (Página 2836)

Modificar acciones (Página 2907)

Copiar, cortar y pegar acciones (Página 2908)

Desplazar acciones (Página 2909)

Borrar acciones (Página 2909)

Definir el tipo de datos de una instrucción GRAPH

Principios básicos de los tipos de datos de instrucciones GRAPH

Introducción

El uso de instrucciones GRAPH puede llevarse a cabo mediante uno de los dos mecanismos siguientes:

- Como llamada de bloque
  En las instrucciones que se inserten como llamada de bloque en el programa, es posible seleccionar el tipo de datos de una lista desplegable.
- Como función interna de GRAPH
  Las funciones internas de GRAPH se asignan directamente a una variable. Para ello, el operando debe ser de un tipo de datos compatible con la función. La siguiente tabla muestra una relación de los tipos de datos admisibles de las funciones internas de GRAPH.

Funciones internas de GRAPH

La siguiente tabla muestra las funciones internas de GRAPH y sus tipos de datos:

Funcionamiento	S7-300/400	S7-1500
BCD_TO_INT	WORD	WORD
BCD_TO_DINT	DWORD	DWORD
INT_TO_BCD	INT	INT
DINT_TO_BCD	DINT	DINT
INT_TO_DINT	INT	INT
DINT_TO_REAL	DINT	DINT
ROUND	REAL	REAL, LREAL
TRUNC	REAL	REAL, LREAL
CEIL	REAL	REAL, LREAL
FLOOR	REAL	REAL, LREAL
NEG	INT, DINT, REAL	SINT, INT, DINT, REAL, LINT, LREAL
ABS	REAL	SINT, INT, DINT, LINT, REAL, LREAL
SQR	REAL	REAL, LREAL
SQRT	REAL	REAL, LREAL
LN	REAL	REAL, LREAL
EXP	REAL	REAL, LREAL
SIN	REAL	REAL, LREAL
COS	REAL	REAL, LREAL
TAN	REAL	REAL, LREAL
ASIN	REAL	REAL, LREAL
ACOS	REAL	REAL, LREAL
ATAN	REAL	REAL, LREAL

Funcionamiento	S7-300/400	S7-1500
NOT	WORD, DWORD	USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, BYTE, WORD, DWORD, LWORD
RLDA	DWORD	-
RRDA	DWORD	-
SWAP	WORD, DWORD	WORD, DWORD, LWORD
AND	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD
OR	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD
XOR	BYTE, WORD, DWORD	BYTE, WORD, DWORD, LWORD
SHL	WORD, DWORD	BYTE, USINT, SINT, WORD, UINT, INT, DWORD, UDINT, DINT, LWORD, ULINT, LINT
SHR	WORD, DWORD	BYTE, WORD, DWORD, USINT, UINT, UDINT, SINT, INT, DINT, LWORD, ULINT, LINT
ROL	DWORD	BYTE, WORD, DWORD, LWORD
ROR	DWORD	BYTE, WORD, DWORD, LWORD
MOD	DINT	USINT, UINT, UDINT, SINT, INT, DINT, ULINT, LINT
-: Función no disponible		

Modificar el tipo de datos de una instrucción GRAPH

Las instrucciones GRAPH que se insertan en el programa mediante una llamada de bloque, se insertan con un tipo de datos predeterminado. Este tipo de datos se puede modificar si es necesario. Los tipos de datos modificables aparecen marcados con un rectángulo verde bajo la inicial del tipo de datos. En las funciones internas de GRAPH se puede modificar el operando.

Modificar el tipo de datos de una instrucción llamada

Para modificar el tipo de datos de una instrucción llamada, proceda del siguiente modo:

1. Haga clic en el tipo de datos de la instrucción.
   Se abrirá la lista desplegable para seleccionar el tipo de datos.
2. Haga clic en el tipo de datos que desea utilizar.

Modificar el tipo de datos de una función interna de GRAPH

Para modificar el tipo de datos de una función interna de GRAPH, proceda del siguiente modo:

1. Indique como operando un valor o una variable. Tenga en cuenta el rango admisible para el operando.

O bien:

1. Mantenga pulsado el botón derecho del ratón y arrastre una variable del tipo de datos admisible por encima del operando.
2. Espere a que aparezca un marco verde en torno al operando.
3. Suelte el botón del ratón.

Modificar acciones

Es posible realizar los cambios siguientes en acciones:

- enlazar la acción con un interlock o deshacer el enlace con un interlock
- enlazar la acción con un evento o deshacer/modificar el enlace con un evento
- cambiar el identificador de la acción
- modificar la acción, p. ej. sustituir operandos o utilizar otra instrucción

Tenga en cuenta que determinados identificadores de acciones requieren el uso de eventos y que en estos identificadores no es posible deshacer el enlace con un evento.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de etapa individual.

Procedimiento

Para modificar una acción, proceda del siguiente modo:

1. Si desea enlazar una acción con un interlock, haga clic en la celda de la columna "Interlock" y elija la entrada "-(C)- - Interlock" de la lista desplegable.
2. Si desea deshacer el enlace con un interlock, haga clic en la celda de la columna "Interlock" y elija la entrada " - Ninguna condición" de la lista desplegable.
3. Si desea enlazar la acción con un evento o modificar el evento enlazado, haga clic en la celda de la columna "Evento" y elija el evento correspondiente de la lista desplegable.
4. Si desea deshacer el enlace con un evento, haga clic en la celda de la columna "Evento" y elija la entrada " - Ningún evento" de la lista desplegable. Si el identificador de la acción requiere un evento, el fondo de la celda se marcará en rojo y se mostrará "<???>". En ese caso, vuelva a asignar un evento o seleccione otro identificador.

5. Si desea modificar el identificador de la acción, haga clic en la celda de la columna "Identificador" y elija el nuevo identificador de la acción en la lista desplegable.
6. Si desea modificar la acción, haga clic en la acción dentro de la columna "Acción" y realice uno de los cambios siguientes:
   - Cambie el o los operandos de la acción.
   - Para cambiar el tipo de datos de un operando, haga clic en el pequeño rectángulo verde y elija el nuevo tipo de datos de la lista desplegable.
   - Para utilizar otra instrucción, borre la acción existente y arrastre otra instrucción desde la Task Card "Instrucciones" hasta la columna "Acción". Introduzca los operandos.
   - Para llamar otro bloque, borre la acción existente y arrastre un bloque mediante Drag & Drop desde el árbol del proyecto hasta la columna "Acción" para llamarlo.

Consulte también

Acciones (Página 2836)

Insertar acciones (Página 2901)

Copiar, cortar y pegar acciones (Página 2908)

Desplazar acciones (Página 2909)

Borrar acciones (Página 2909)

Copiar, cortar y pegar acciones

Requisitos

Hay un bloque de función GRAPH abierto en la vista de etapa individual.

Copiar acciones

Para copiar una acción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la primera celda de la acción que desea copiar.
2. Elija el comando "Copiar" del menú contextual.

Cortar acciones

Para cortar una acción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la primera celda de la acción que desea cortar.
2. Elija el comando "Cortar" del menú contextual.

Pegar acciones desde el portapapeles

Para pegar una acción desde el portapapeles, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la posición en la que desea insertar la acción.
2. Elija el comando "Pegar" del menú contextual.

Consulte también

Acciones (Página 2836)
Insertar acciones (Página 2901)
Modificar acciones (Página 2907)
Desplazar acciones (Página 2909)
Borrar acciones (Página 2909)

Desplazar acciones

Las acciones se pueden desplazar dentro de la tabla de acciones.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de etapa individual.

Procedimiento

Para desplazar una acción dentro de la tabla de acciones, proceda del siguiente modo:

1. Haga clic en la primera celda de la acción que desea desplazar y mantenga pulsado el botón del ratón.
   La acción se selecciona.
2. Arrastre la acción hasta la posición deseada mediante Drag & Drop.

Consulte también

Acciones (Página 2836)
Insertar acciones (Página 2901)
Modificar acciones (Página 2907)
Copiar, cortar y pegar acciones (Página 2908)
Borrar acciones (Página 2909)

Borrar acciones

Requisitos

Hay un bloque de función GRAPH abierto en la vista de etapa individual.

Procedimiento

Para borrar una acción, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la primera celda de la acción que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Acciones (Página 2836)

Insertar acciones (Página 2901)

Modificar acciones (Página 2907)

Copiar, cortar y pegar acciones (Página 2908)

Desplazar acciones (Página 2909)

Programar condiciones

Programar interlocks

En el segmento de interlock es posible utilizar un máximo de 32 instrucciones con un máximo de 32 operandos interconectados. En KOP, las instrucciones pueden insertarse tanto en un circuito como en la barra de alimentación.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de etapa individual.

Procedimiento

Para programar interlocks, proceda del siguiente modo:

1. Abra el segmento de interlock haciendo clic en la pequeña flecha situada delante de "Interlock -(c)-".
2. Abra la Task Card "Instrucciones".
3. Navegue hasta la instrucción que desea insertar.
4. Arrastre la instrucción hasta la posición deseada del segmento mediante Drag & Drop.
5. En la columna "Interlock" de la tabla de acciones, haga clic en la celda de la acción que desea enlazar con el interlock y elija la entrada "-(C)- - Interlock" de la lista desplegable.

O:

1. Abra el segmento de interlock haciendo clic en la pequeña flecha situada delante de "Interlock -(c)-".
2. Abra la Task Card "Instrucciones".
3. En la paleta "Instrucciones básicas", navegue hasta "KOP/FUP > General > Cuadro vacío".

4. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
5. Haga clic en "??".
   Se abre un campo de texto.
6. Haga clic en el botón para la función automática de autocompletar o pulse la combinación de teclas <Ctrl+I>.
   Se abre la función automática de autocompletar. Contiene solo los elementos admisibles en función del contexto.
7. Seleccione el elemento deseado de la lista desplegable.
8. En la columna "Interlock" de la tabla de acciones, haga clic en la celda de la acción que desea enlazar con el interlock y elija la entrada "-(C)- - Interlock" de la lista desplegable.

Consulte también

Condiciones (Página 2847)

Programar supervisiones (Página 2911)

Programar transiciones (Página 2912)

Programar instrucciones permanentes (Página 2914)

Mostrar u ocultar informaciones de variables (Página 2921)

Definir el lenguaje de programación en segmentos (Página 2867)

Programar supervisiones

En el segmento de supervisión es posible utilizar un máximo de 32 instrucciones con un máximo de 32 operandos interconectados. En KOP, las instrucciones pueden insertarse tanto en un circuito como en la barra de alimentación.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de etapa individual.

Procedimiento

Para programar supervisiones, proceda del siguiente modo:

1. Abra el segmento de supervisión haciendo clic en la pequeña flecha situada delante de "Supervisión -(v)-".
2. Abra la Task Card "Instrucciones".
3. Navegue hasta la instrucción que desea insertar.
4. Arrastre la instrucción hasta la posición deseada del segmento mediante Drag & Drop.

O:

1. Abra el segmento de supervisión haciendo clic en la pequeña flecha situada delante de "Supervisión -(v)-".
2. Abra la Task Card "Instrucciones".
3. En la paleta "Instrucciones básicas", navegue hasta "KOP/FUP > General > Cuadro vacío".
4. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
5. Haga clic en "??".
   Se abre un campo de texto.
6. Haga clic en el botón para la función automática de autocompletar o pulse la combinación de teclas <Ctrl+I>.
   Se abre la función automática de autocompletar. Contiene solo los elementos admisibles en función del contexto.
7. Seleccione el elemento deseado de la lista desplegable.

Consulte también

Condiciones (Página 2847)

Programar interlocks (Página 2910)

Programar transiciones (Página 2912)

Programar instrucciones permanentes (Página 2914)

Mostrar u ocultar informaciones de variables (Página 2921)

Definir el lenguaje de programación en segmentos (Página 2867)

Programar transiciones

En el segmento de transición es posible utilizar un máximo de 32 instrucciones con un máximo de 32 operandos interconectados. En KOP, las instrucciones pueden insertarse tanto en un circuito como en la barra de alimentación. Las transiciones se programan en la vista de etapa individual o en la vista de cadena.

Programar transiciones en la vista de etapa individual

Para programar transiciones en la vista de etapa individual, proceda del siguiente modo:

1. Abra la vista de etapa individual.
2. Abra el segmento de transición haciendo clic en la pequeña flecha situada delante de "<número de transición> - <nombre de transición>".
3. Abra la Task Card "Instrucciones".
4. Navegue hasta la instrucción que desea insertar.
5. Arrastre la instrucción hasta la posición deseada del segmento mediante Drag & Drop.

O:

1. Abra el segmento de transición haciendo clic en la pequeña flecha situada delante de "<número de transición> - <nombre de transición>".
2. Abra la Task Card "Instrucciones".
3. En la paleta "Instrucciones básicas", navegue hasta "KOP/FUP > General > Cuadro vacío".
4. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
5. Haga clic en "??".
   Se abre un campo de texto.
6. Haga clic en el botón para la función automática de autocompletar o pulse la combinación de teclas <Ctrl+I>.
   Se abre la función automática de autocompletar. Contiene solo los elementos admisibles en función del contexto.
7. Seleccione el elemento deseado de la lista desplegable.

Programar transiciones en la vista de cadena

Para programar transiciones en la vista de cadena, proceda del siguiente modo:

1. Abra la vista de cadena.
2. Amplíe la transición que desea programar. Para ello, haga clic en el botón situado junto a la transición o seleccione la transición y pulse <Intro>.
3. Abra la Task Card "Instrucciones".
4. Navegue hasta la instrucción que desea insertar.
5. Arrastre la instrucción hasta la posición deseada del segmento mediante Drag & Drop.

O:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "KOP/FUP > General > Cuadro vacío".
3. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
4. Haga clic en "??".
   Se abre un campo de texto.
5. Haga clic en el botón para la función automática de autocompletar o pulse la combinación de teclas <Ctrl+I>.
   Se abre la función automática de autocompletar. Contiene solo los elementos admisibles en función del contexto.
6. Seleccione el elemento deseado de la lista desplegable.

Consulte también

Condiciones (Página 2847)

Programar interlocks (Página 2910)

Programar supervisiones (Página 2911)

Programar instrucciones permanentes (Página 2914)
Mostrar u ocultar informaciones de variables (Página 2921)
Definir el lenguaje de programación en segmentos (Página 2867)

Programar instrucciones permanentes

Trabajar con segmentos

Utilizar segmentos

Función

Las instrucciones permanentes se programan dentro de los segmentos en un bloque de función GRAPH. Para poder programar una instrucción permanente debe haber como mínimo un segmento. El programa también se puede dividir en varios segmentos para facilitar la claridad.

Consulte también

Insertar segmentos (Página 2914)
Seleccionar segmentos (Página 2915)
Copiar e insertar segmentos (Página 2916)
Borrar segmentos (Página 2916)
Contraer y expandir segmentos (Página 2917)
Insertar título del segmento (Página 2918)
Navegar en segmentos (Página 2919)

Insertar segmentos

Requisitos

Un bloque está abierto.

Procedimiento

Para insertar un segmento nuevo, proceda del siguiente modo:

1. Seleccione el segmento debajo del cual desea insertar un segmento nuevo.
2. Elija el comando "Insertar segmento" del menú contextual.

Resultado

Se inserta un nuevo segmento vacío en el bloque.

Consulte también

Utilizar segmentos (Página 2914)
Seleccionar segmentos (Página 2915)
Copiar e insertar segmentos (Página 2916)
Borrar segmentos (Página 2916)
Contraer y expandir segmentos (Página 2917)
Insertar título del segmento (Página 2918)
Navegar en segmentos (Página 2919)

Seleccionar segmentos

Requisitos

Existe un segmento.

Seleccionar un segmento

Para seleccionar un segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento que desea seleccionar.

Seleccionar varios segmentos

Para seleccionar varios segmentos individuales, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Ctrl>.
2. Haga clic en todos los segmentos que desea seleccionar.

Para seleccionar varios segmentos consecutivos, proceda del siguiente modo:

1. Pulse y mantenga oprimida la tecla <Mayús>.
2. Haga clic en el primer segmento que desea seleccionar.
3. Haga clic en el último segmento que desea seleccionar.
   Se seleccionan el primer y último segmento, así como todos los segmentos que se encuentran entre éstos.

Consulte también

Utilizar segmentos (Página 2914)
Insertar segmentos (Página 2914)
Copiar e insertar segmentos (Página 2916)

Borrar segmentos (Página 2916)
Contraer y expandir segmentos (Página 2917)
Insertar título del segmento (Página 2918)
Navegar en segmentos (Página 2919)

Copiar e insertar segmentos

Los segmentos copiados pueden insertarse en el mismo bloque o en un bloque diferente. Los segmentos creados en KOP o FUP también pueden insertarse en los bloques de cada uno de los lenguajes de programación diferentes.

Requisitos

Existe un segmento.

Procedimiento

Para copiar e insertar un segmento, proceda del siguiente modo:

1. Seleccione el o los segmentos que desea copiar.
2. Elija el comando "Copiar" del menú contextual.
3. Seleccione el segmento debajo del cual desea insertar el segmento copiado.
4. Elija el comando "Pegar" del menú contextual.

Consulte también

Utilizar segmentos (Página 2914)
Insertar segmentos (Página 2914)
Seleccionar segmentos (Página 2915)
Borrar segmentos (Página 2916)
Contraer y expandir segmentos (Página 2917)
Insertar título del segmento (Página 2918)
Navegar en segmentos (Página 2919)

Borrar segmentos

Requisitos

Existe un segmento.

Procedimiento

Para borrar un segmento, proceda del siguiente modo:

1. Seleccione el segmento que desea borrar.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Utilizar segmentos (Página 2914)
Insertar segmentos (Página 2914)
Seleccionar segmentos (Página 2915)
Copiar e insertar segmentos (Página 2916)
Contraer y expandir segmentos (Página 2917)
Insertar título del segmento (Página 2918)
Navegar en segmentos (Página 2919)

Contraer y expandir segmentos

Requisitos

Existe un segmento.

Expandir y contraer un segmento

Para abrir un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia la derecha situada en la barra de título del segmento.

Para cerrar un segmento, proceda del siguiente modo:

1. Haga clic en la flecha hacia abajo situada en la barra de título del segmento.

Expandir y contraer todos los segmentos

Para expandir o contraer todos los segmentos, proceda del siguiente modo:

1. Haga clic en el botón "Abrir todos los segmentos" o "Cerrar todos los segmentos" de la barra de herramientas.

Consulte también

Utilizar segmentos (Página 2914)
Insertar segmentos (Página 2914)
Seleccionar segmentos (Página 2915)
Copiar e insertar segmentos (Página 2916)

Borrar segmentos (Página 2916)

Insertar título del segmento (Página 2918)

Navegar en segmentos (Página 2919)

Insertar título del segmento

El título de segmento es la cabecera de un segmento. La longitud del título del segmento está limitada a una línea. Puede introducir el título manualmente o definirlo de forma automática. Si elige el ajuste automático, puede aplicarlo a segmentos individuales o establecer en la configuración que el título del segmento debe definirse de forma automática.

Para la inserción automática del título del segmento se evalúa el comentario del operando de una de las siguientes instrucciones del segmento:

- Asignación
- Activar salida
- Desactivar salida

Para ello, se utiliza la instrucción que aparece primero en el segmento.

El título del segmento solo se inserta automáticamente si se cumplen las siguientes condiciones:

- El segmento todavía no tiene ningún título.
- El operando de la instrucción que se utiliza para el comentario tiene un comentario.

Nota

Al insertar el título del segmento de forma automática, tenga en cuenta las siguientes limitaciones:

- Si modifica el comentario del operando con posterioridad, el título del segmento no se actualiza.
- Si cambia el operando de la instrucción, el título del segmento no se actualiza.
- El título del segmento solo se define mediante las instrucciones de escritura citadas antes.
- Si el operando es del tipo de datos Array, se utiliza el comentario del Array y no los comentarios de los elementos Array.
- No se tienen en cuenta los comentarios de operandos no válidos.

Introducir el título del segmento manualmente

Para introducir un título de segmento, proceda del siguiente modo:

1. Haga clic en la barra de título del segmento.
2. Introduzca el título de segmento.

Definir el título del segmento automáticamente

Para establecer que el título del segmento debe definirse fundamentalmente de forma automática, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Active la casilla de verificación "Establecer título del segmento automáticamente" en el grupo "Configuración avanzada".
   Desde este momento, los títulos de los segmentos se definirán automáticamente si se cumplen las condiciones anteriores.

Para definir automáticamente un título de segmento concreto, proceda del siguiente modo:

1. Haga clic en la barra de título de un segmento con el botón derecho del ratón y seleccione "Segmento <Número del segmento>".
2. Elija el comando "Establecer título del segmento automáticamente" del menú contextual.
   El título del segmento seleccionado se define a partir del comentario del operando si se cumplen las condiciones anteriores.

Consulte también

Utilizar segmentos (Página 2914)

Insertar segmentos (Página 2914)

Seleccionar segmentos (Página 2915)

Copiar e insertar segmentos (Página 2916)

Borrar segmentos (Página 2916)

Contraer y expandir segmentos (Página 2917)

Navegar en segmentos (Página 2919)

Navegar en segmentos

Existe la posibilidad de navegar en un bloque directamente hasta una posición determinada.

Procedimiento

Para navegar en un bloque hasta una posición determinada, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la sección de lógica de la ventana de programación.
2. Elija el comando "Ir a > Segmento/línea" del menú contextual.
   Se abre el cuadro diálogo "Ir a".
3. Introduzca el segmento hasta el que desea navegar.
4. Introduzca el número de la fila del segmento hasta el que desea navegar.
5. Confirme la entrada realizada con "Aceptar".

Resultado

Si es posible, se muestra la fila correspondiente. En caso de que el segmento o la fila deseados no existan, se visualiza el último segmento existente, o bien, la última fila existente en el segmento deseado.

Consulte también

Utilizar segmentos (Página 2914)

Insertar segmentos (Página 2914)

Seleccionar segmentos (Página 2915)

Copiar e insertar segmentos (Página 2916)

Borrar segmentos (Página 2916)

Contraer y expandir segmentos (Página 2917)

Insertar título del segmento (Página 2918)

Insertar instrucciones permanentes

Al desplegar una paleta para instrucciones permanentes anteriores o posteriores, esta contiene por defecto un segmento vacío como mínimo. En cada segmento es posible utilizar un máximo de 32 instrucciones con un máximo de 32 operandos interconectados. En KOP, las instrucciones pueden insertarse tanto en un circuito como en la barra de alimentación. Dentro de un bloque de función GRAPH es posible utilizar un máximo de 250 instrucciones permanentes anteriores y 250 posteriores.

Requisitos

Hay un bloque de función GRAPH abierto en la vista de cadena.

Procedimiento

Para programar instrucciones permanentes, proceda del siguiente modo:

1. Abra la paleta "Instrucciones permanentes anteriores" o "Instrucciones permanentes posteriores" en la navegación.
2. Haga clic en un segmento.
   En el área de trabajo se abre la vista de instrucciones permanentes anteriores o posteriores.
3. Abra la Task Card "Instrucciones".
4. Navegue hasta la instrucción que desea insertar.
5. Arrastre la instrucción hasta la posición deseada del segmento mediante Drag & Drop.

O:

1. Abra la Task Card "Instrucciones".
2. En la paleta "Instrucciones básicas", navegue hasta "General > Cuadro vacío".

3. Arrastre el elemento "Cuadro vacío" hasta la posición deseada del segmento mediante Drag & Drop.
4. Haga clic en "??".
   Se abre un campo de texto.
5. Haga clic en el botón para la función automática de autocompletar o pulse la combinación de teclas <Ctrl+I>.
   Se abre la función automática de autocompletar. Contiene solo los elementos admisibles en función del contexto.
6. Seleccione el elemento deseado de la lista desplegable.

Consulte también

Condiciones (Página 2847)

Programar interlocks (Página 2910)

Programar supervisiones (Página 2911)

Programar transiciones (Página 2912)

Mostrar u ocultar informaciones de variables (Página 2921)

Definir el lenguaje de programación en segmentos (Página 2867)

Mostrar u ocultar informaciones de variables

Introducción

Para instrucciones permanentes, interlocks, supervisiones y transiciones está disponible la información siguiente sobre las variables utilizadas en la ventana de programación:

- Nombre de la variable
- Dirección de la variable
- Comentarios sencillos o jerárquicos para la documentación de variables

La información se recoge de la interfaz del bloque en el caso de las variables locales y de DB, y de la tabla de variables PLC en el caso de las variables válidas para toda la CPU.

Es posible visualizar la información de las variables para todos los bloques o solo para bloques abiertos individuales. Si selecciona la información de variables para todos los bloques, se muestra la información de las variables de todos los bloques abiertos actuales y futuros.

La información de las variables se puede volver a ocultar en cualquier momento. Si se ha ocultado la información de las variables de todos los bloques, se puede volver a ver para bloques abiertos individualmente.

Si se selecciona la visualización de información de variables con comentarios jerárquicos, en el caso de las variables estructuradas también se muestran los comentarios de los niveles de estructura de orden superior. En ese caso, esa información se muestra entre paréntesis después del comentario de la variable, y los comentarios de los diferentes niveles se separan con un punto. Si en un nivel de estructura no hay ningún comentario para una variable, el comentario en cuestión no se visualiza, lo que puede reconocerse por dos puntos seguidos.

Mostrar u ocultar información de variables para todos los bloques

Para mostrar u ocultar la información de variables para todos los bloques, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC" en la navegación local.
3. Para visualizar la información de las variables, seleccione las opciones "Mostrar" o "Información de la variable con jerarquía" en la lista desplegable "Información de la variable", en función de si desea visualizar los comentarios sencillos o jerárquicos.
4. Para ocultar la información de las variables, seleccione la opción "Ocultar" en la lista desplegable "Información de la variable".
   La información de las variables se muestra u oculta para todos los bloques abiertos. Si abre más bloques, la información de las variables se mostrará u ocultará dependiendo del ajuste seleccionado.

Mostrar u ocultar información de variables para un bloque abierto

Para mostrar u ocultar la información de variables para un bloque abierto, proceda del siguiente modo:

1. Para visualizar la información de las variables, seleccione las opciones "Mostrar información de la variable" o "Información de la variable con jerarquía" en la lista desplegable "Muestra información sobre la variable", en función de si desea visualizar los comentarios sencillos o jerárquicos.
2. Para ocultar la información de las variables, seleccione la opción "Ocultar información de la variable" en la lista desplegable "Muestra información sobre la variable".
   Se muestra u oculta la información sobre las variables.

Consulte también

Programar interlocks (Página 2910)

Programar supervisiones (Página 2911)

Programar transiciones (Página 2912)

Programar instrucciones permanentes (Página 2914)

Insertar llamadas de bloque en GRAPH

Inserción manual de llamadas de bloque

En las acciones y acciones permanentes es posible llamar otros bloques.

Insertar llamadas de bloque manualmente en una acción

Para insertar manualmente una llamada de bloque en una acción, proceda del siguiente modo:

1. Abra el bloque de función GRAPH en la vista de etapa individual.
2. Haga clic en la celda de la acción en la columna "Acción".
3. Introduzca "CALL <nombre del bloque que debe llamarse>" y confirme la entrada con la tecla Intro.
4. Sustituya el "<nombre del bloque que debe llamarse>" por el nombre del bloque que quiere llamar.
   Si el bloque que debe llamarse es un bloque de función, se abre el cuadro de diálogo "Opciones de llamada". En este caso ejecute los pasos 5 y 6.
5. Indique en él si desea llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca seguidamente un nombre en el campo de entrada "Nombre" para el bloque de datos que se debe asignar al bloque de función. Seleccione "Instancia individual" o "Multiinstancia" e introduzca si es preciso el número del bloque de datos.
   - Si hace clic en el botón "Multiinstancia", introduzca seguidamente en el campo de entrada "Nombre en la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
6. Confirme la entrada realizada con "Aceptar".

Insertar una llamada de bloque manualmente en una instrucción permanente

Para insertar manualmente una llamada de bloque en una instrucción permanente, proceda del siguiente modo:

1. Abra la vista "Instrucciones permanentes anteriores" o "Instrucciones permanentes posteriores".
2. Haga clic con el botón derecho del ratón en la posición del segmento en la que desea insertar la llamada de bloque.
3. Elija el comando "Insertar cuadro vacío" del menú contextual.
4. Haga clic en "??".
   Se abre un campo de texto.
5. Introduzca el nombre del bloque que desea llamar o abra la función de autocompletar y seleccione el bloque.
   Si el bloque que debe llamarse es un bloque de función, se abre el cuadro de diálogo "Opciones de llamada". En este caso ejecute los pasos 6 y 7.

6. Indique en él si desea llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca seguidamente un nombre en el campo de entrada "Nombre" para el bloque de datos que se debe asignar al bloque de función. Seleccione "Instancia individual" o "Multiinstancia" e introduzca si es preciso el número del bloque de datos.
   - Si hace clic en el botón "Multiinstancia", introduzca seguidamente en el campo de entrada "Nombre en la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
7. Confirme la entrada realizada con "Aceptar".

Consulte también

Llamadas de bloque (Página 2842)

Insertar llamadas de bloque mediante Drag & Drop (Página 2924)

Actualizar llamadas de bloque en GRAPH (Página 2925)

Insertar llamadas de bloque mediante Drag & Drop

En las acciones y acciones permanentes es posible llamar otros bloques. Sin embargo, tenga en cuenta que solo es posible llamar los bloques de función como instancia individual.

Insertar llamadas de bloque en una acción mediante Drag & Drop

Para insertar una llamada de bloque en una acción mediante Drag & Drop, proceda del siguiente modo:

1. Abra el bloque de función GRAPH en la vista de etapa individual.
2. Arrastre el bloque que desea llamar desde el árbol del proyecto hasta la celda de la acción en la columna "Acción".
   Si el bloque que debe llamarse es un bloque de función, se abre el cuadro de diálogo "Opciones de llamada". En este caso ejecute los pasos 3 y 4.
3. Indique en él si desea llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca seguidamente un nombre en el campo de entrada "Nombre" para el bloque de datos que se debe asignar al bloque de función. Seleccione "Instancia individual" o "Multiinstancia" e introduzca si es preciso el número del bloque de datos.
   - Si hace clic en el botón "Multiinstancia", introduzca seguidamente en el campo de entrada "Nombre en la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
4. Confirme la entrada realizada con "Aceptar".

Insertar llamadas de bloque en una instrucción permanente mediante Drag & Drop

Para insertar una llamada de bloque en una instrucción permanente mediante Drag & Drop, proceda del siguiente modo:

1. Abra la vista "Instrucciones permanentes anteriores" o "Instrucciones permanentes posteriores".
2. Arrastre el bloque que desea llamar desde el árbol del proyecto hasta la posición del segmento de la instrucción permanente en la que desea insertar la llamada de bloque. Si el bloque que debe llamarse es un bloque de función, se abre el cuadro de diálogo "Opciones de llamada". En este caso ejecute los pasos 3 y 4.
3. Indique en él si desea llamar el bloque como instancia individual o como multiinstancia.
   - Si hace clic en el botón "Instancia individual", introduzca seguidamente un nombre en el campo de entrada "Nombre" para el bloque de datos que se debe asignar al bloque de función. Seleccione "Instancia individual" o "Multiinstancia" e introduzca si es preciso el número del bloque de datos.
   - Si hace clic en el botón "Multiinstancia", introduzca seguidamente en el campo de entrada "Nombre en la interfaz" el nombre de la variable con el que el bloque de función llamado se debe introducir como variable estática en la interfaz del bloque que efectúa la llamada.
4. Confirme la entrada realizada con "Aceptar".

Consulte también

Llamadas de bloque (Página 2842)

Inserción manual de llamadas de bloque (Página 2922)

Actualizar llamadas de bloque en GRAPH (Página 2925)

Actualizar llamadas de bloque en GRAPH

Si cambian los parámetros de interfaz de un bloque llamado, la llamada no podrá ejecutarse correctamente. Actualice las llamadas de bloque con objeto de evitar llamadas incoherentes.

Existen dos posibilidades de actualizar llamadas de bloque, a saber:

- Actualización explícita en el editor de programación.
  Se actualizan las llamadas de bloque del bloque abierto.
- Actualización implícita durante la compilación.
  Se actualizan todas las llamadas de bloque del programa, así como los tipos de datos PLC utilizados.

Actualizar llamadas de bloque en el editor de programación

Para actualizar una llamada de bloque en un bloque, proceda del siguiente modo:

1. Abra el bloque en el editor de programación.
2. Haga clic en el botón "Actualizar llamadas de bloque incoherentes" de la barra de herramientas.

Actualizar llamadas de bloque durante la compilación

Para actualizar implícitamente todas las llamadas de bloque y usos de tipos de datos PLC durante la compilación, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Llamadas de bloque (Página 2842)

Inserción manual de llamadas de bloque (Página 2922)

Insertar llamadas de bloque mediante Drag & Drop (Página 2924)

Definir avisos de interlock y supervisión

Activar y desactivar el visor de avisos

Es posible activar o desactivar el visor de avisos globalmente para bloques GRAPH nuevos o localmente para bloques individuales. El ajuste local de un bloque tiene prioridad frente al ajuste global.

Activar o desactivar globalmente el visor de avisos

Para activar o desactivar el visor de avisos globalmente para bloques de función GRAPH nuevos, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Si desea activar el visor de avisos, active la casilla de verificación "Activar avisos" en el área "Ajuste predeterminado para bloques nuevos > Propiedades de aviso > Avisos". Si desea desactivar el visor de avisos, desactive la casilla de verificación "Activar avisos".

Activar o desactivar localmente el visor de avisos

Para activar o desactivar el visor de avisos para un bloque de función GRAPH existente, proceda del siguiente modo:

1. Abra la vista de avisos para el bloque de función GRAPH.
2. Si desea activar el visor de avisos, active la casilla de verificación "Activar avisos" en el área "Avisos". Si desea desactivar el visor de avisos, desactive la casilla de verificación "Activar avisos".

Nota

En bloques de función GRAPH de S7-300/400 en los que se utiliza la opción "Generar DB minimizado" no es posible modificar el visor de avisos.

Consulte también

Avisos de interlock y supervisión (Página 2847)

Editar avisos (Página 2927)

Establecer el acuse obligatorio (Página 2928)

Editar avisos

Los avisos de interlock y supervisión se editan de uno de los siguientes modos:

- globalmente para todos los bloques de función GRAPH nuevos
- localmente para bloques de función GRAPH existentes
- para etapas individuales

Editar avisos de interlock y supervisión globalmente para todos los bloques de función GRAPH nuevos

Para editar avisos de interlock y supervisión globalmente para bloques de función GRAPH nuevos, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Si desea editar los avisos de interlock, introduzca el texto deseado en el campo de texto "Texto de aviso" del área "Avisos de interlock".
4. Si desea editar los avisos de supervisión, introduzca el texto deseado en el campo de texto "Texto de aviso" del área "Avisos de supervisión".

Editar avisos de interlock y supervisión localmente para un bloque de función GRAPH existente

Para editar avisos de interlock y supervisión localmente para un bloque de función GRAPH existente, proceda del siguiente modo:

1. Abra la vista de avisos para el bloque de función GRAPH.
2. En el área "Avisos de interlock" y "Avisos de supervisión", active los bloques de aviso que deben visualizarse en el aviso.
3. Si desea editar el texto de aviso, introduzca el texto deseado delante de los bloques de aviso. En caso necesario, reemplace el texto estándar o el texto especificado globalmente.

Nota

En bloques de función GRAPH de S7-300/400 en los que se utiliza la opción "Generar DB minimizado" no es posible editar los textos de aviso.

Editar el aviso de interlock de una etapa individual

Para editar el aviso de interlock de una etapa individual, proceda del siguiente modo:

1. Abra la etapa en la vista de etapa individual.
2. Seleccione el interlock "-(C)-".
3. Abra la ventana de inspección y cambie a la ficha "Propiedades", si fuera necesario.
4. Introduzca el texto de aviso deseado en el campo de texto "Texto de aviso".
   El texto se adjuntará al final del aviso de interlock por medio del bloque de aviso "<texto específico de etapa>".

Editar el aviso de supervisión de una etapa individual

Para editar el aviso de supervisión de una etapa individual, proceda del siguiente modo:

1. Abra la etapa en la vista de etapa individual.
2. Seleccione la supervisión "-(v)-".
3. Abra la ventana de inspección y cambie a la ficha "Propiedades", si fuera necesario.
4. Introduzca el texto de aviso deseado en el campo de texto "Texto de aviso".
   El texto se adjuntará al final del aviso de supervisión por medio del bloque de aviso "<texto específico de etapa>".

Consulte también

Avisos de interlock y supervisión (Página 2847)

Activar y desactivar el visor de avisos (Página 2926)

Establecer el acuse obligatorio (Página 2928)

Establecer el acuse obligatorio

Es posible establecer que los avisos de interlock y supervisión deban ser acusados por el operador.

Procedimiento

Para establecer el acuse obligatorio de avisos de interlock y supervisión, proceda del siguiente modo:

1. Abra la vista de avisos para el bloque de función GRAPH.
2. Si desea activar el acuse obligatorio para avisos de interlock, active la casilla de verificación "Acuse obligatorio en avisos de interlock" en el área "Avisos de interlock". Si desea quitar el acuse obligatorio, desactive la casilla de verificación "Acuse obligatorio en avisos de interlock".
3. Si desea activar el acuse obligatorio para avisos de supervisión, active la casilla de verificación "Acuse obligatorio en avisos de supervisión" en el área "Avisos de supervisión". Si desea quitar el acuse obligatorio, desactive la casilla de verificación "Acuse obligatorio en avisos de supervisión".

Nota

En bloques de función GRAPH de S7-300/400 en los que se utiliza la opción "Generar DB minimizado" no es posible modificar el acuse obligatorio.

Consulte también

Avisos de interlock y supervisión (Página 2847)

Activar y desactivar el visor de avisos (Página 2926)

Editar avisos (Página 2927)

Corregir errores de sintaxis en el programa

Principios básicos de los errores de sintaxis

Error de sintaxis

Los siguientes errores se consideran errores de sintaxis:

- Separadores ausentes o la utilización de demasiados separadores
- Fallos de notación de palabras clave
- Notación errónea de las etiquetas
- Falta de concordancia entre la notación y los nemónicos ajustados (p. ej. "I2.3" en lugar de "I2.3")
- Utilización de palabras clave como operandos

Identificación de errores de sintaxis

Los errores de sintaxis se resaltan con subrayado rojo o con texto rojo.

Esta marca permite detectar a primera vista las entradas incorrectas, y saltar de error a error para corregirlo. Asimismo, los errores de sintaxis se listan con el respectivo mensaje de error en la ficha "Información" en la ventana de inspección.

Consulte también

Buscar errores de sintaxis en el programa (Página 2930)

Buscar errores de sintaxis en el programa

Procedimiento

Para buscar errores de sintaxis en el programa de usuario, proceda del siguiente modo:

1. Seleccione la posición del programa en la que desea buscar los errores.
2. Haga clic en el botón "Ir al error siguiente" de la barra de herramientas.
   Se marca el primer error desde la posición seleccionada.

Los botones "Ir al error siguiente" e "Ir al error anterior" de la barra de herramientas permiten encontrar y corregir todos los errores del bloque.

O bien:

1. En la ventana de inspección, elija "Información > Sintaxis" para abrir la lista de errores.
   Todos los errores de sintaxis se listan en la tabla junto con una breve descripción del error.
2. Si aparece, haga clic en el signo de interrogación azul situado junto al texto de error para obtener indicaciones adicionales sobre cómo solucionar el error.
3. Haga doble clic en el error que desea corregir.
   Se selecciona el error en cuestión.

Consulte también

Principios básicos de los errores de sintaxis (Página 2929)

Cambiar el lenguaje de programación

Reglas para cambiar el lenguaje de programación

Reglas

Observe las reglas siguientes al cambiar el lenguaje de programación de un bloque:

- Todas las familias de CPU:
  - Sólo es posible cambiar bloques completos. No es posible cambiar segmentos individuales.
  - No es posible cambiar entre bloques programados con los lenguajes de programación SCL o GRAPH. No obstante, en los bloques GRAPH se puede cambiar el lenguaje de los segmentos entre KOP y FUP.
- S7-300/400:
  - Se puede cambiar entre los lenguajes de programación KOP, FUP y AWL.
  - Se pueden crear segmentos en un bloque con un lenguaje de programación diferente y copiarlos luego al bloque deseado.
  - Los segmentos del bloque cuyo lenguaje de programación no se puede cambiar se visualizan en el lenguaje original.
- S7-1200/1500:
  - Se puede cambiar entre los lenguajes de programación KOP y FUP.
- S7-1500:
  - Se pueden crear segmentos AWL dentro de bloques KOP y FUP. Lo que no es posible es copiar entre AWL y KOP/FUP.

Cambiar el lenguaje de programación

Procedimiento

Para cambiar el lenguaje de programación, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic con el botón derecho del ratón en el bloque.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo con las propiedades del bloque.
3. Seleccione la entrada "General" en la navegación local.
4. Seleccione el nuevo lenguaje de programación de la lista desplegable "Lenguaje".
5. Confirme la selección con "Aceptar".

Consulte también

Reglas para cambiar el lenguaje de programación (Página 2931)

Tratar errores de ejecución del programa

Principios básicos del tratamiento de errores

Introducción

Los errores de acceso a la periferia y de programación se consideran errores de ejecución del programa. Para reaccionar ante los errores de ejecución del programa existen diferentes posibilidades según la CPU empleada.

Tratamiento de errores de ejecución del programa para S7-300/400

Para las CPU S7-300/400 se puede programar el OB de error de ejecución del programa (OB 85). Si se produce un error de ejecución del programa y no se utiliza el OB correspondiente, la CPU cambia al estado operativo "STOP".

Encontrará más información sobre el OB de error de ejecución del programa en la descripción del modo de funcionamiento de las CPU S7-300/400.

Tratamiento de errores de ejecución del programa para S7-1200/1500

Para las CPU de las series S7-1200 y S7-1500 se puede escoger el tipo de tratamiento de errores. Existen dos posibilidades:

- Se utiliza el tratamiento de errores global de la CPU:
  - S7-1200: la CPU genera una entrada en el búfer de diagnóstico y permanece en estado operativo "RUN".
  - S7-1500: para las CPU S7-1500 se pueden programar el OB de error de programación (OB 121) y el OB de error de acceso a la periferia (OB 122). Si el OB de error de programación no está disponible en la CPU, esta cambia al estado operativo "STOP" en caso de producirse un error de programación. Si se produce un error de acceso a la periferia, la CPU permanece siempre en el estado operativo "RUN", independientemente de si el OB de error de acceso a la periferia está disponible.
    No obstante, recuerde que un OB existente de error de acceso a periferia o de programación no se llama de forma sincrónica al error. Por ello es posible que los OBs de error de acceso a periferia o de programación no se ejecuten inmediatamente al producirse el error, sino con retardo en función de la prioridad ajustada. Si se producen más errores antes de que se haya completado la ejecución del OB de error de acceso a periferia o de programación, no se llamará ningún otro OB de dichos tipos. Para impedir el descarte de OBs de error de acceso a periferia o de programación, ajuste una prioridad suficientemente alta.
    Para las instrucciones "Leer campo" (FieldRead), "Escribir campo" (FieldWrite), "Leer dirección de memoria" (PEEK) y "Escribir dirección de memoria" (POKE) es posible utilizar la salida de habitación ENO para detectar errores de acceso a periferia y programación.
    Encontrará más información sobre los OB de error en la descripción del modo de funcionamiento de las CPU S7-1500.
- Se utiliza un tratamiento de errores locales propios. Por tratamiento de errores local se entiende el tratamiento de errores dentro de un bloque. El tratamiento de errores local aporta las ventajas siguientes:
  - En la memoria de sistema se almacena una información de error, que se puede consultar y evaluar.
  - La información de error permite programar en el bloque una reacción al error ocurrido.
  - Las evaluaciones y reacciones a errores programadas no interrumpen el ciclo del programa.
  - El tratamiento de errores local no supone una carga innecesaria para el rendimiento del sistema. Si no se presentan errores, no se ejecutan las evaluaciones y reacciones a errores programadas.
    El tratamiento de errores local se aplica sólo a los bloques para los que se ha ajustado explícitamente. Si se ha ajustado un tratamiento de errores local para un bloque, no se realizarán tratamientos de errores globales para errores de dicho bloque.

Nota

Tenga en cuenta las indicaciones siguientes:

- Todos los errores de acceso a la memoria y a la periferia deben ser interceptados mediante un tratamiento de errores global o local.
- Si no se produce ningún error de acceso a la memoria mediante los parámetros de una instrucción, se puede consultar el correspondiente ENO.

Consulte también

GET_ERROR: Consultar error localmente (Página 3840)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

GET_ERROR: Consultar error localmente (Página 3527)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Consultar y eliminar errores en el código del programa (Página 265)

Tratamiento de errores local

Principios del tratamiento de errores local

Introducción

El tratamiento de errores local permite consultar si se han producido errores en un bloque y evaluar la información correspondiente. El tratamiento de errores local se puede ajustar para bloques de organización (OB), bloques de función (FB) y funciones (FC). Si está activado el tratamiento de errores local, se ignorará la reacción del sistema.

El tratamiento de errores local se aplica sólo a los bloques para los que esté ajustado explícitamente. El ajuste del tratamiento de errores local no se aplica a un bloque que llama, ni se transfiere a bloques llamados. Para los bloques de orden superior e inferior sigue siendo aplicable la configuración del sistema, si no se ha programado ningún tratamiento de errores propio para ellos.

Procesamiento general del tratamiento de errores local

Si ocurren errores durante la ejecución de un bloque para el que se ha ajustado el tratamiento de errores local, se disparará una reacción predeterminada en función de los siguientes tipos de error:

- Errores de escritura: los errores se ignoran y la ejecución del programa prosigue sin más.
- Errores de lectura: la ejecución del programa continúa con el valor sustitutivo "0".
- Errores de ejecución: se cancela la ejecución de la instrucción. La ejecución del programa continúa con la siguiente instrucción.

En la memoria de sistema se guarda información sobre el primer error ocurrido. Esta información se puede consultar y visualizar mediante una instrucción (GET_ERROR o GET_ERR_ID). La información de error se visualiza en formato editable. También puede evaluarse mediante instrucciones adicionales, para programar una reacción al error ocurrido.

Cuando se consulta la información acerca del primer error ocurrido, se libera memoria para errores en la memoria de sistema. Si posteriormente ocurren más errores, se muestra la información acerca del siguiente error ocurrido.

Instrucciones para el tratamiento de errores local

Para el tratamiento de errores local se puede utilizar una de las instrucciones indicadas a continuación:

- GET_ERROR: Consultar error localmente
- GET_ERR_ID: Consultar ID de error localmente

Las instrucciones se diferencian en la profundidad de la información de error mostrada.

Encontrará más información sobre las instrucciones en "Consulte también".

Consulte también

GET_ERROR: Consultar error localmente (Página 3840)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

GET_ERROR: Consultar error localmente (Página 3527)

Prioridades en la salida de errores

Vista general de las prioridades

En el tratamiento de errores local se muestra la información acerca del primer error ocurrido. Si, durante la ejecución de una instrucción, se presentan simultáneamente varios errores, éstos se muestran según su prioridad. La tabla siguiente muestra las prioridades de los distintos tipos de error:

Prioridad	Tipo de error
1	Error en el código del programa
2	Falta la referencia
3	Rango no válido
4	El DB no existe
5	Los operandos no son compatibles
6	El ancho del rango indicado no es suficiente
7	No existen temporizadores o contadores
8	No se puede escribir en un DB
9	Error de periferia

Prioridad	Tipo de error
10	La instrucción no existe
11	El bloque no existe
12	Profundidad de anidamiento no válida

La prioridad más alta es 1 y la más baja es 12.

Consulte también

GET_ERROR: Consultar error localmente (Página 3840)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

GET_ERROR: Consultar error localmente (Página 3527)

Activar el tratamiento de errores local para un bloque

Introducción

El tratamiento de errores local para un bloque se activa insertando una de las siguientes instrucciones en un segmento:

- GET_ERROR: Consultar error localmente
- GET_ERR_ID: Consultar ID de error localmente

Encontrará más información sobre las instrucciones en "Consulte también".

Si está activado el tratamiento de errores local para un bloque, se ignorarán las reacciones del sistema ajustadas para el bloque en cuestión.

Requisitos

- El bloque está abierto.
- La Task Card "Instrucciones" está abierta.

Procedimiento

Para activar el tratamiento de errores local para un bloque, proceda del siguiente modo:

1. Navegue hasta la paleta "Instrucciones básicas" de la Task Card "Instrucciones".
2. Abra la carpeta "Control del programa".
3. Arrastre la instrucción "Consultar error localmente" (GET_ERROR) o "Consultar ID de error localmente" (GET_ERR_ID) hasta el segmento deseado.

Resultado

El tratamiento de errores local se activa para el bloque abierto. La casilla de verificación "Tratamiento de errores locales en el bloque" se activa en la ventana de inspección, en "Propiedades > Atributos". Este ajuste no se puede modificar en la ventana de inspección. El tratamiento de errores local se puede desactivar borrando las instrucciones correspondientes insertadas para dicho tratamiento.

Consulte también

GET_ERROR: Consultar error localmente (Página 3840)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

GET_ERROR: Consultar error localmente (Página 3527)

9.1.4.3 Programar bloques de datos

Principios básicos de la programación de bloques de datos

Un bloque de datos (DB) sirve para guardar valores que se escriben durante la ejecución del programa.

Al contrario que los bloques lógicos, los de datos solo contienen declaraciones de variables. No contienen ni segmentos ni instrucciones. Las declaraciones de variables definen la estructura del bloque de datos.

Tipos de bloques de datos

Existen dos tipos de bloques de datos, a saber:

- Bloques de datos globales
  Un bloque de datos global no está asignado a ningún bloque lógico. Se puede acceder a los valores de un bloque de datos global desde cualquier bloque lógico. Un bloque de datos global contiene únicamente variables estáticas.
  La estructura del bloque de datos global puede definirse libremente. En la tabla de declaración del bloque de datos se declaran los elementos de datos que debe contener el bloque de datos global.
- Bloques de datos de instancia
  Un bloque de datos de instancia es aquel que está asignado directamente a un bloque de función (FB). La estructura de un bloque de datos de instancia no puede definirse libremente, sino que está determinada por la declaración de interfaz del bloque de función. El bloque de datos de instancia contiene exactamente los parámetros de bloque y las variables declaradas en él.
  No obstante, en el bloque de datos de instancia pueden definirse valores específicos de la instancia, p. ej. valores de arranque para las variables declaradas.

Bloques de datos ARRAY (S7-1500)

Los bloques de datos ARRAY son bloques de datos globales que constan de un ARRAY. Dicho ARRAY puede basarse en cualquier tipo de datos. Puede ser un ARRAY de un tipo de datos PLC (UDT), por ejemplo. El DB no contiene otros elementos más que el ARRAY. Gracias a su estructura plana, los bloques de datos ARRAY facilitan el acceso a los elementos ARRAY y su transmisión a bloques llamados.

La sección "Transferencia" de la Task Card "Instrucciones" permite direccionar bloques de datos ARRAY.

Tipos de datos PLC como plantillas para bloques de datos globales

Los tipos de datos PLC pueden usarse como plantilla para la creación de bloques de datos globales con la misma estructura de datos. La estructura se define solo una vez como tipo de datos PLC y a continuación se generan los bloques de datos requeridos a través de la asignación del tipo de datos PLC.

Tipos de datos del sistema como plantillas para bloques de datos globales

Los tipos de datos del sistema también pueden servir como plantilla para la creación de bloques de datos globales con la misma estructura de datos. Los tipos de datos del sistema ya tienen una estructura definida de manera fija. El tipo de datos del sistema se inserta solo una vez en el programa y a continuación, a través de la asignación del tipo de datos del sistema, se generan otros bloques de datos con la misma estructura.

Tipos de acceso

Existen dos métodos diferentes para acceder a valores de datos contenidos en bloque de datos:

- Bloques de datos con acceso optimizado (solo S7-1200)
  Los bloques de datos con acceso optimizado carecen de una estructura definida. En la declaración se asigna a los elementos de datos un nombre simbólico, pero no una dirección fija dentro del bloque. A los valores de datos de estos bloques se accede a través del nombre simbólico.
  Para los bloques de datos ARRAY, el atributo "Acceso optimizado al bloque" siempre está activado.
- Bloques de datos con acceso estándar (todas las familias de CPU)
  Los bloques de datos con acceso estándar tienen una estructura fija. En la declaración se asigna a los elementos de datos un nombre simbólico y también una dirección fija dentro del bloque. Se puede acceder a los valores de datos de estos bloques a través del nombre simbólico o la dirección.
  No se admiten bloques de datos ARRAY con acceso estándar.

Remanencia de los valores de datos

Para evitar pérdidas de datos en caso de un fallo de alimentación, los valores de datos pueden depositarse en un área de memoria remanente.

Consulte también

Crear bloques de datos (Página 2598)

Bloques con acceso optimizado (Página 2258)

Bloques de datos globales (DB) (Página 2254)

Bloques de datos instancia (Página 2256)

Estructura de la tabla de declaración de bloques de datos

Estructura de la tabla de declaración de bloques de datos

La siguiente figura muestra la estructura de la tabla de declaración para bloques de datos. La representación varía según el tipo de bloque y acceso.

Nombre	Tipo de datos	Valor de arranq...	Remanencia	Visible en HMI	Comentario
Input					
MyInput1	Bool	false	☑	☑	
Output					
MyOutput1	Byte	0	☐	☑	
InOut					
Static					

Representación de valores específicos de instancia

En los bloques de datos de instancia se pueden adoptar valores ya definidos provenientes de la interfaz del bloque de función asignado, o bien definir valores de arranque específicos de la instancia. Los valores que se adoptan desde el bloque de función se representan en color gris. Los valores en gris se pueden reemplazar por valores específicos de instancia. Los valores que ya se han modificado específicamente por instancia no aparecen en gris.

Significado de las columnas

En la siguiente tabla se explica el significado de las distintas columnas. Las columnas pueden mostrarse u ocultarse según necesidad. El número de columnas visualizadas varía en función de la familia de CPU.

Columna	Significado
	Símbolo en el que se puede hacer clic para mover o copiar la variable. Ésta puede arrastrarse hasta un programa mediante Drag & Drop y utilizarse allí como operando.
Nombre	Nombre de la variable.
Tipo de datos	Tipo de datos de la variable.
Offset	Dirección relativa de la variable. Esta columna sólo es visible en los bloques de datos con acceso estándar.

Columna	Significado
Valor predeterminado	Valor que se ha predeterminado para la variable en la interfaz de un bloque lógico de orden superior o un tipo de datos PLC. Los valores que contiene la columna "Valores predeterminados" pueden modificarse sólo en el bloque lógico de orden superior o en el tipo de datos PLC. En el bloque de datos, los valores sólo se muestran.
Valor de arranque	Valor que debe adoptar la variable durante el arranque. Al crear el bloque de datos, los valores predeterminados definidos en un bloque lógico se utilizan como valores de arranque. Estos valores aplicados pueden sustituirse aquí por valores de arranque específicos de la instancia. La indicación de un valor de arranque es opcional. Si no se especifica ningún valor, la variable adopta el valor predeterminado durante el arranque. Si tampoco se ha definido ningún valor predeterminado, se utiliza el valor estándar válido para el tipo de datos. Por ejemplo, el valor estándar de BOOL es "FALSE".
Valor de observación	Valor de datos actual en la CPU. Esta columna se visualiza si existe una conexión online y se hace clic en el botón "Observar todo".
Instantánea	Muestra los valores que se han cargado desde el dispositivo.
Remanencia	Marca la variable como remanente. Los valores de variables remanentes se conservan tras desconectar la alimentación.
Visible en HMI	Indica si la variable está visible en la lista desplegable de HMI mediante un ajuste predeterminado.
Accesible desde HMI	Indica si HMI puede acceder a esta variable en tiempo de ejecución.
Valor de ajuste	Los valores de ajuste son valores que requieren un ajuste fino en la puesta en marcha. Tras la puesta en marcha, los valores de estas variables pueden transferirse como valores de arranque al programa offline y guardarse allí.
Comentario	Comentario para documentar la variable.

Consulte también

Editar las propiedades de las variables en bloques de datos (Página 2953)

Ajustar los valores de datos durante la puesta en marcha (Página 2962)

Crear bloques de datos (Página 2598)

Programar la estructura de datos de bloques de datos globales (Página 2944)

Principios básicos de los valores de arranque (Página 2948)

Definir valores de arranque (Página 2948)

Ajustar remanencia (Página 2951)

Crear bloques de datos

Requisitos

La carpeta "Bloques de programa" del árbol del proyecto está abierta.

Procedimiento

Para crear un bloque de datos, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo bloque".
   Se abre el cuadro de diálogo "Agregar nuevo bloque".
2. Haga clic en el botón "Bloque de datos (DB)".
3. Seleccione el tipo de bloque de datos. Existen las posibilidades siguientes:
   - Para crear un bloque de datos global, seleccione la entrada de la lista "DB global".
   - Para crear un bloque de datos ARRAY, seleccione la entrada de la lista "DB ARRAY".
   - Para crear un bloque de datos instancia, seleccione de la lista el bloque de función al que desea asignar el bloque de datos instancia. En la lista solo aparecen bloques de función que se han creado con anterioridad para la CPU.
   - Para crear un bloque de datos basado en un tipo de datos PLC, seleccione en la lista el tipo de datos en cuestión. En la lista solo aparecen tipos de datos PLC que se han creado con anterioridad para la CPU.
   - Para crear un bloque de datos basado en un tipo de datos de sistema, seleccione en la lista el tipo de datos en cuestión. En la lista solo se muestran los tipos de datos del sistema que se han insertado con anterioridad en bloques de programa de la CPU.
4. Introduzca un nombre para el bloque de datos.
5. Introduzca las propiedades del nuevo bloque de datos.
6. Si se ha seleccionado como "tipo" un bloque de datos ARRAY, introduzca el tipo de datos ARRAY y el límite superior de ARRAY.
   El límite superior de ARRAY puede modificarse en todo momento en la ventana de propiedades del bloque generado. El tipo de datos ARRAY no puede modificarse a posteriori.
7. Haga clic en "Más información" para introducir más propiedades del nuevo bloque de datos.
   Se abre un área con campos de entrada adicionales.
8. Introduzca todas las propiedades deseadas.
9. Active la casilla de verificación "Agregar y abrir" si el bloque debe abrirse inmediatamente después de crearse.
10. Confirme la entrada realizada con "Aceptar".

Resultado

Se crea el nuevo bloque de datos. Este se encuentra entonces en la carpeta "Bloques de programa" del árbol del proyecto.

Consulte también

Bloques de datos instancia (Página 2256)

Programar bloques de datos (Página 2937)

Programar tipos de datos PLC (Página 2967)

Bloques de datos globales (DB) (Página 2254)

Bloques con acceso optimizado (Página 2258)

Sinopsis de las propiedades de bloques (Página 2608)

Actualizar bloques de datos

Introducción

Las modificaciones en la interfaz de un bloque de función o un tipo de datos PLC pueden provocar que los bloques de datos correspondientes se vuelvan incoherentes. Estas incoherencias se marcan en color rojo tanto en la tabla de declaración como en el punto de llamada del bloque. Para corregir las incoherencias es necesario actualizar los bloques de datos.

Existen tres posibilidades de actualizar bloques de datos:

- Actualización explícita en la tabla de declaración de los bloques de datos.
  Se actualiza el bloque de datos. Se adoptan las modificaciones procedentes de la interfaz del bloque de función asignado y las de los tipos de datos PLC utilizados.
- Actualización explícita en el editor de programación.
  Se actualizan las llamadas de bloque del bloque abierto. También se adaptan los bloques de datos instancia respectivos.
- Actualización implícita durante la compilación.
  Se actualizan todas las llamadas de bloque en el programa, así como los tipos de datos PLC utilizados y los bloques de datos instancia respectivos.

Actualización explícita en la tabla de declaración de los bloques de datos

Para actualizar explícitamente un bloque de datos en particular, proceda del siguiente modo:

1. Abra el bloque de datos.
2. Elija "Actualizar interfaz" en el menú contextual.

Actualización explícita en el editor de programación

Para actualizar todas las llamadas de bloque o una determinada en un bloque, proceda del siguiente modo:

1. Abra el bloque en el editor de programación.
2. Haga clic con el botón derecho del ratón en la instrucción que contiene la llamada de bloque.
3. Elija el comando "Actualizar" del menú contextual.
4. Se abre el cuadro de diálogo "Sincronización de la interfaz". En él se visualizan las diferencias entre la interfaz utilizada y la interfaz modificada del bloque llamado.
5. Si desea actualizar la llamada de bloque, haga clic en "Aceptar". Para cancelar la actualización, haga clic en "Cancelar".

Actualización implícita durante la compilación

Para actualizar implícitamente todas las llamadas de bloque, las ubicaciones de los tipos de datos PLC y los bloques de datos instancia durante la compilación, proceda del siguiente modo:

1. Abra el árbol del proyecto.
2. Seleccione la carpeta "Bloques de programa".
3. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".

Consulte también

Modificar propiedades de las variables en bloques de datos de instancia (Página 2955)

Ampliar bloques de datos

Descripción

Para modificar con posterioridad programas de PLC que ya se han puesto en servicio y que funcionan sin problemas en una instalación, las CPU de la serie S7-1500 y la mayoría de las CPU de la serie S7-1200 V4 permiten ampliar los bloques de datos globales durante el funcionamiento.

Los bloques modificados pueden cargarse sin necesidad de pasar la CPU a STOP y sin afectar a los valores de las variables ya cargadas.

De ese modo es posible realizar cambios en el programa con facilidad. El proceso controlado no se ve afectado por esta carga (cargar sin reinicializar).

Modo de funcionamiento

Cada bloque de datos dispone de forma predeterminada de una reserva de memoria. En un primer momento, la reserva no se utiliza. Si compila y carga el bloque y después se da cuenta de que desea cargar cambios de interfaz con posterioridad, entonces active la reserva de memoria. Todas las variables declaradas después se cargan en la reserva de memoria. Al realizar la carga a continuación, los valores de las variables ya cargadas no se ven afectados.

Para modificar el programa más tarde, mientras la instalación no esté funcionando, se dispone además de la posibilidad de modificar el formato de memoria de uno o varios bloques en un único paso. Todas las variables de la reserva de memoria se mueven al área de memoria habitual. La reserva de memoria se libera de nuevo para otras ampliaciones de interfaz.

Requisitos

La función "Cargar sin reinicializar" está disponible si se cumplen los siguientes requisitos:

- El proyecto está disponible en el formato "TIA Portal V12" o en una versión superior.
- Trabaja con una CPU que admite "Cargar sin reinicializar".
- Los bloques se crearon en KOP, FUP, AWL o SCL.

- Los bloques han sido creados por el usuario, es decir, no son bloques suministrados.
- Se trata de bloques de acceso optimizado.

Pasos fundamentales

Para ampliar un bloque de datos y cargar a continuación el bloque sin reinicializar, siga los siguientes pasos.

1. Todos los bloques tienen de forma predeterminada una reserva de memoria de 100 bytes. En caso necesario, el tamaño de la reserva puede adaptarse.
2. Active la reserva de memoria.
3. Amplíe la interfaz del bloque.
4. Compile el bloque.
5. Vuelva a cargar el bloque en la CPU de la forma acostumbrada.

Referencia

Encontrará descripciones detalladas de cada uno de los pasos en el capítulo "Cargar bloques (S7-1200/1500) ".

Programar la estructura de datos de bloques de datos globales

Declarar variables del tipo de datos simple

Requisitos

Hay un bloque de datos global abierto.

Nota

Las estructuras de bloques de datos de instancia y bloques de datos basados en un tipo de datos PLC no se pueden modificar directamente, puesto que están definidas por el bloque de función o el tipo de datos PLC.

El tipo de bloque de datos está introducido en las propiedades del bloque.

Procedimiento

Para declarar una variable de un tipo de datos simple, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Haga clic en el botón para la selección del tipo de datos de la columna "Tipo de datos". Se abre una lista de los tipos de datos permitidos.
3. Seleccione el tipo de datos que desee utilizar.

4. Opcionalmente: Modifique las propiedades de las variables que se visualizan en las otras columnas.
5. Repita los pasos 1 a 4 para todas las variables que desea declarar.

Consulte también

Mostrar y editar propiedades de bloques (Página 2613)

Declarar variables del tipo de datos ARRAY (Página 2945)

Declarar variables del tipo de datos STRUCT (Página 2946)

Editar tablas (Página 322)

Editar la tabla de declaración de bloques de datos (Página 2958)

Definir valores de arranque (Página 2948)

Ajustar remanencia (Página 2951)

Observar valores de datos online (Página 2961)

Declarar variables del tipo de datos ARRAY

Requisitos

Hay un bloque de datos global abierto.

Procedimiento

Para declarar una variable del tipo de datos ARRAY, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca el tipo de datos "Array" en la columna "Tipo de datos". Se dispone de la ayuda de la función de autocompletar.
   Se abre el cuadro de diálogo "Array".
3. Indique el tipo de datos de los elementos Array en el campo de entrada "Tipo de datos".
4. A continuación, indique el límite inferior y superior de cada dimensión en el campo de entrada "Límites de matriz".
   Ejemplo de un ARRAY unidimensional:
   `[0..3]`
   Ejemplo de un ARRAY tridimensional:
   `[0..3, 0..15, 0..33]`
5. Confirme la entrada.
6. Opcionalmente: Modifique las propiedades de las variables que se visualizan en las otras columnas.

Se crea la variable, pero permanece contraída. Para expandir el ARRAY, haga clic en el triángulo situado delante de la variable. Tenga en cuenta que los ARRAY muy grandes no pueden expandirse para garantizar una mayor claridad.

Introducir los valores de arranque de elementos ARRAY

Para preseleccionar los valores de arranque de los distintos elementos de un ARRAY, proceda del siguiente modo:

1. Haga clic en el triángulo situado delante de la variable del tipo de datos ARRAY.
   El ARRAY se expande y cada uno de sus elementos aparece en una fila.
2. Introduzca los valores deseados en la columna "Valor de arranque".

Consulte también

Definir valores de arranque (Página 2948)

Ajustar remanencia (Página 2951)

Editar la tabla de declaración de bloques de datos (Página 2958)

Matriz (Página 2354)

Utilizar la función automática de autocompletar (Página 2637)

Declarar variables del tipo de datos STRUCT

Requisitos

Hay un bloque de datos global abierto.

Procedimiento

Para declarar una variable del tipo de datos STRUCT, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca "Struct" en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
   Debajo de las nuevas variables se inserta una fila vacía sangrada.
3. Inserte el primer elemento de estructura en la fila vacía.
   Debajo del elemento se inserta nuevamente una fila vacía.
4. Seleccione un tipo de datos para el elemento de estructura.
5. Opcionalmente: Modifique las propiedades del elemento de estructura que se visualizan en las otras columnas de la interfaz del bloque.
6. Repita los pasos 4 a 7 para todos los demás elementos de estructura.
   No es necesario finalizar explícitamente la estructura. La estructura termina con el último elemento introducido.
7. Para insertar una nueva variable después de la estructura, deje una fila vacía al final de la estructura e introduzca la nueva variable en la segunda fila vacía.

Resultado

Se crea la variable del tipo de datos STRUCT.

Introducir los valores de arranque de los elementos de estructuras

Para predeterminar los valores de arranque de los distintos elementos de una estructura, proceda del siguiente modo:

1. Haga clic en el triángulo situado delante de la variable del tipo de datos STRUCT.
   La estructura se expande y los elementos de estructura individuales se muestran en sus propias filas.
2. Introduzca los valores deseados en la columna "Valor de arranque".

Consulte también

Definir valores de arranque (Página 2948)

Ajustar remanencia (Página 2951)

Editar la tabla de declaración de bloques de datos (Página 2958)

STRUCT (Página 2358)

Utilizar la función automática de autocompletar (Página 2637)

Declarar variables basadas en un tipo de datos PLC

Requisitos

- Hay un bloque de datos global abierto.
- En la CPU actual está declarado un tipo de datos PLC.

Procedimiento

Para declarar una variable basada en el tipo de datos PLC, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca el tipo de datos PLC en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
3. Opcionalmente: Modifique las propiedades de las variables que se visualizan en las otras columnas de la tabla.

Resultado

La variable se crea.

Consulte también

Estructura de la interfaz del bloque (Página 2642)

Editar la tabla de declaración de bloques de datos (Página 2958)

Utilizar la función automática de autocompletar (Página 2637)

Definir valores de arranque

Principios básicos de los valores de arranque

Definición de "Valor de arranque"

El valor de arranque de una variable es un valor definido por el usuario que la variable debe adoptar tras un arranque de la CPU.

Las variables remanentes constituyen un caso particular. Sus valores adoptan el valor de arranque definido sólo después de un "Arranque en frío". Tras un "Rearranque completo (en caliente)" se conservan los valores y no se ponen al valor de arranque.

Definición de "Valor predeterminado"

La estructura de los bloques de datos puede derivarse de elementos de orden superior:

- Un bloque de datos de instancia se basa p. ej. en la interfaz de un bloque lógico de orden superior.
- Un bloque de datos global puede basarse en un tipo de datos PLC predefinido.

En este caso, se puede definir un valor predeterminado para cada variable en el elemento de orden superior. Al crear el bloque de datos, se utilizan estos valores predeterminados como valores de arranque. Estos valores aplicados pueden sustituirse luego en el bloque de datos por valores de arranque específicos de la instancia.

La indicación de un valor de arranque es opcional. Si no se especifica ningún valor, la variable adopta el valor predeterminado durante el arranque. Si tampoco se ha definido ningún valor predeterminado, se utiliza el valor estándar válido para el tipo de datos. Por ejemplo, el valor estándar de BOOL es "FALSE".

Consulte también

Definir valores de arranque (Página 2949)

Estructura de la tabla de declaración de bloques de datos (Página 2939)

Declarar variables y constantes locales en la interfaz del bloque (Página 2648)

Aplicar valores del programa online como valores de arranque (Página 2965)

Definir valores de arranque

Definir valores de arranque

Para definir valores de arranque para las variables de un bloque de datos, proceda del siguiente modo:

1. Abra el bloque de datos.
   La columna "Valor predeterminado" muestra los valores predeterminados definidos para las variables en la interfaz de un bloque lógico de orden superior o en un tipo de datos PLC.
2. Para mostrar todos los elementos de tipos de datos estructurados, haga clic en el botón "Modo avanzado".
3. Introduzca los valores iniciales deseados en la columna "Valor de arranque". El valor se debe corresponder con el tipo de datos de la variable y no puede exceder el rango de valores del tipo de datos.
   Los valores de arranque se han definido. Durante el arranque, la variable adopta el valor definido, siempre que no haya sido declarada como remanente.

Inicializar una variable al valor predeterminado

Para poner una variable al valor predeterminado para la que se ha definido un valor de arranque, proceda del siguiente modo:

1. Seleccione un valor modificado en la tabla.
2. Borre el valor.
   Se introduce el valor predeterminado. Este se representa desactivado.

Inicializar todas las variables al valor predeterminado

Para poner todas las variables al valor predeterminado para la que se ha definido un valor de arranque, proceda del siguiente modo:

1. Elija el botón "Resetear valores de arranque" en la barra de herramientas.
   Los valores predeterminados se transfieren a la columna "Valor de arranque". Los valores de arranque protegidos contra escritura no se sobrescriben.

Consulte también

Principios básicos de los valores de arranque (Página 2948)

Aplicar valores del programa online como valores de arranque (Página 2965)

Editar la tabla de declaración de bloques de datos (Página 2958)

Carga de valores modificados

Introducción

Para aplicar en el programa online valores de arranque modificados del programa offline, es necesario cargar los cambios. Deben distinguirse los siguientes casos:

- Cargar valores de arranque modificados de variables no remanentes
- Cargar valores de arranque modificados de variables remanentes
- Cargar valores de arranque modificados de valores de ajuste

Requisitos

Los valores de arranque del programa offline se han modificado.

Procedimiento

Para cargar valores de arranque modificados de variables no remanentes, proceda del siguiente modo:

1. Seleccione los bloques que desea cargar en el árbol del proyecto.
2. Elija el comando "Cargar en dispositivo > Software (solo cambios)" del menú contextual.
   Los bloques se compilan y se cargan.
   Los valores de arranque de las nuevas variables definidas se depositan en la memoria de carga de la CPU. En la siguiente transición de STOP a RUN, el programa arrancará con los valores de arranque nuevos.

Para cargar valores de arranque modificados de variables remanentes, proceda del siguiente modo:

1. Seleccione los bloques que desea cargar en el árbol del proyecto.
2. En el menú contextual, elija el comando "Compilar > Software (compilar todos los bloques)".
3. A continuación, elija el comando "Cargar e inicializar programa PLC en dispositivo" del menú "Online".
   Los bloques online se borran y se sobrescriben con los bloques nuevos. Con ello se inicializan todas las variables, incluidas las remanentes.

Encontrará información sobre la carga de valores de ajuste modificados así como información general sobre la carga en "Consulte también".

Consulte también

Compilar y cargar bloques (Página 3012)

Inicializar valores de ajuste en el programa online (Página 2964)

Ajustar remanencia

Remanencia de variables en bloques de datos

Comportamiento remanente

Para impedir la pérdida de datos en caso de un corte de alimentación, algunos datos pueden marcarse como remanentes. Entonces se guardarán en un área de memoria remanente. Las posibilidades de ajuste de la remanencia dependen del tipo de bloque de datos y del tipo de acceso a bloques configurado.

Consulte también

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Ajustar la remanencia en un bloque de datos global (Página 2952)

Ajustar la remanencia en un bloque de datos de instancia

Introducción

La posibilidad de modificar el comportamiento remanente en un bloque de datos instancia depende del tipo de acceso del bloque de función de orden superior:

- Bloque de función con acceso estándar
  Los datos instancia se pueden definir en conjunto como remanentes o no remanentes. Los ajustes de remanencia individuales no son posibles para algunas variables.
- Bloque de función con acceso optimizado
  En el bloque de datos instancia se pueden definir los ajustes de remanencia de las variables marcadas en la interfaz del bloque con "Ajustar en IDB". El comportamiento remanente de estas variables tampoco puede ajustarse individualmente. Los ajustes de remanencia repercuten en todas las variables marcadas en la interfaz del bloque con "Ajustar en IDB".

Ajustar la remanencia con acceso estándar

Para ajustar la remanencia de forma centralizada de todas las variables en bloques de datos con acceso estándar, proceda del siguiente modo:

1. Abra el bloque de datos instancia.
2. Active la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables se definen como remanentes.
3. Para inicializar la configuración de remanencia de todas las variables, desactive la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables se definen como no remanentes.

Ajustar la remanencia con acceso optimizado

Para ajustar el comportamiento remanente en los bloques de datos con acceso optimizado de las variables marcadas con "Ajustar en IDB", proceda del siguiente modo:

1. Abra el bloque de datos instancia.
2. Active la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables marcadas en la interfaz del bloque con "Ajustar en IDB" se definen como remanentes.
3. Para inicializar la configuración de remanencia de las variables, desactive la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables marcadas en la interfaz del bloque con "Ajustar en IDB" se definen como no remanentes.

Consulte también

Fundamentos del acceso a bloques (Página 2258)

Remanencia de variables en bloques de datos (Página 2951)

Editar la tabla de declaración de bloques de datos (Página 2958)

Ajustar la remanencia en un bloque de datos global

Introducción

La posibilidad de modificar el comportamiento remanente en un bloque de datos global depende del tipo de acceso:

- Bloque de datos global con acceso estándar
  Los datos se pueden definir en conjunto como remanentes o no remanentes. Los ajustes de remanencia individuales no son posibles para algunas variables.
- Bloque de datos global con acceso optimizado
  Los ajustes de remanencia de las variables se pueden definir individualmente. En variables de tipos de datos estructurados, los ajustes de remanencia se aplican a todos los elementos de las variables.

Ajustar la remanencia con acceso estándar

Para ajustar la remanencia de forma centralizada de todas las variables en bloques de datos con acceso estándar, proceda del siguiente modo:

1. Abra el bloque de datos global.
2. Active la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables se definen como remanentes.
3. Para inicializar la configuración de remanencia de todas las variables, desactive la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables se definen como no remanentes.

Ajustar la remanencia con acceso optimizado

Para ajustar individualmente la remanencia de las variables en un bloque de datos con acceso optimizado, proceda del siguiente modo:

1. Abra el bloque de datos global.
2. Active la casilla de verificación de la columna "Remanencia" para la variable para la que desea configurar el comportamiento remanente.
   La variable seleccionada se define como remanente.
3. Para inicializar la configuración de remanencia de las variables, desactive la casilla de verificación en la columna "Remanencia" de una variable.
   Todas las variables seleccionadas se definen como no remanentes.

Consulte también

Fundamentos del acceso a bloques (Página 2258)

Remanencia de variables en bloques de datos (Página 2951)

Editar la tabla de declaración de bloques de datos (Página 2958)

Editar las propiedades de las variables en bloques de datos

Propiedades de variables en bloques de datos

Propiedades

La siguiente tabla muestra una vista general de las propiedades de las variables en los bloques de datos:

Grupo	Propiedad	Descripción
General	Nombre	Nombre de la variable.
	Tipo de datos	Tipo de datos de la variable.
	Valor predeterminado	Valor que se ha predeterminado para la variable en la interfaz de un bloque lógico de orden superior o un tipo de datos PLC. Los valores que contiene la columna "Valores predeterminados" pueden modificarse sólo en el bloque lógico de orden superior o en el tipo de datos PLC. En el bloque de datos, los valores sólo se muestran.

Grupo	Propiedad	Descripción
	Valor de arranque	Valor que debe adoptar la variable al arrancar la CPU. Al crear el bloque de datos, los valores predeterminados definidos en un bloque lógico se utilizan como valores de arranque. Estos valores aplicados pueden sustituirse aquí por valores de arranque específicos de la instancia. La indicación de un valor de arranque es opcional. Si no se especifica ningún valor, la variable adopta el valor predeterminado durante el arranque. Si tampoco se ha definido ningún valor predeterminado, se utiliza el valor estándar válido para el tipo de datos. Por ejemplo, el valor estándar de BOOL es "FALSE".
	Comentario	Comentario acerca de la variable.
Atributos	Remanencia	Marca la variable como remanente. Los valores de variables remanentes se conservan tras desconectar la alimentación. Este atributo sólo está disponible en la interfaz de bloques de función con acceso optimizado.
	Accesible desde HMI	Indica si la variable puede utilizarse en HMI. Si el atributo está definido, puede acceder a la variable desde HMI en modo de escritura o lectura. Si el atributo no está definido, no puede acceder a la variable desde HMI. No obstante, tenga en cuenta que con el atributo "Accesible desde HMI" no se consigue una protección de acceso general para la variable. El acceso en modo de escritura o lectura a través de otras aplicaciones puede ser posible aunque el atributo no esté activado.
	Visible en HMI	Indica si la variable está visible en la lista desplegable de HMI mediante un ajuste predeterminado.
	Parámetros ocultos	Indica si debe ocultarse la variable durante la llamada del bloque. Esto solo es posible si se ha indicado un parámetro actual predefinido válido.
	Parámetro actual predefinido	Define un parámetro que debe utilizarse como parámetro actual durante la llamada del bloque.
	Visible	Identifica si un parámetro está visible en CFC.
	Parametrizable	Identifica si un parámetro es parametrizable en CFC.
	Para test	Identifica si un parámetro está registrado para el modo Test de CFC.
	Interconectable	Identifica si un parámetro es interconectable en CFC.
	Habilitar la relectura de variables	Identifica si un parámetro es relevante para la función de CFC "Leer esquema de PLC".
	Textos enumerativos	Asigna un parámetro a una enumeración en CFC.
	Unidad de ingeniería	Asigna un parámetro a una unidad en CFC.
	Límite inferior	Define el valor límite inferior para el parámetro en CFC.
	Límite superior	Define el valor límite superior para el parámetro en CFC.

Consulte también

Modificar propiedades de las variables en bloques de datos de instancia (Página 2955)

Modificar propiedades de las variables en bloques de datos globales (Página 2956)

Ajustar remanencia (Página 2951)

Definir valores de arranque (Página 2948)

Modificar propiedades de las variables en bloques de datos de instancia

Propiedades de variables específicas de instancia

Existen dos posibilidades para definir las propiedades de las variables:

- Las propiedades de las variables se adoptan desde la interfaz del bloque de función asignado.
  Las propiedades que se adoptan desde el bloque de función se representan en color gris en las columnas de la tabla de declaración. Las propiedades "Nombre" y "Tipo de datos" siempre se adoptan.
- Estas definen las propiedades específicas de la instancia.
  Algunas propiedades se pueden modificar específicamente para cada instancia. Algunos valores modificables son "Comentario" o "Visible en HMI". Las propiedades que se han modificado de forma específica de la instancia no aparecen en color gris en las columnas de la tabla de declaración. Las modificaciones específicas de instancia se mantienen aunque la interfaz del bloque de función de orden superior se modifique y el bloque de datos instancia se actualice después.

Editar las propiedades de un elemento en la tabla de declaración

Para editar las propiedades de un elemento, proceda del siguiente modo:

1. Abra el bloque de datos instancia.
2. Seleccione el elemento deseado en la tabla.
3. Modifique las entradas en las columnas.

Editar las propiedades de varios elementos en la tabla de declaración

Las columnas "Remanencia", "Visible en HMI", "Accesible desde HMI" y "Valor de ajuste" pueden activarse y desactivarse simultáneamente para varios elementos seleccionados.

Para cambiar una de estas propiedades para varios elementos, proceda del siguiente modo:

1. Abra el bloque de datos.
2. Mantenga pulsada la tecla CTRL.
3. En la columna deseada, seleccione sucesivamente las celdas cuyos valores desee cambiar.
4. Elija los comandos "Activar <propiedad>" o "Desactivar <propiedad>" del menú contextual.

Editar las propiedades en la ventana de propiedades

Para editar las propiedades de una variable en particular, proceda del siguiente modo:

1. Seleccione una variable de la tabla.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre la ventana de propiedades. Muestra las propiedades de la variable en las áreas "General" y "Atributos".
3. Seleccione el área deseada en la navegación local.
4. Modifique las entradas en los campos de entrada.

Inicializar propiedades individuales al valor predeterminado

Para inicializar propiedades individuales de variables al valor definido en el bloque de función como valor predeterminado, proceda del siguiente modo:

1. Seleccione en la tabla un valor modificado específico de instancia.
2. Borre el valor.
   El valor específico de instancia se borra y se introduce el valor predeterminado obtenido en la interfaz del bloque de función. El valor predeterminado se representa en color gris.

Consulte también

Actualizar bloques de datos (Página 2942)

Propiedades de variables en bloques de datos (Página 2953)

Editar la tabla de declaración de bloques de datos (Página 2958)

Modificar propiedades de las variables en bloques de datos globales

Introducción

Existen dos posibilidades para definir las propiedades de las variables:

- Las propiedades de las variables se adoptan desde el tipo de datos PLC.
  Las propiedades que se adoptan desde el tipo de datos PLC se representan en color gris en las columnas de la tabla de declaración. Las propiedades "Nombre" y "Tipo de datos" siempre se adoptan.
- Se definen propiedades específicas.
  Algunas propiedades se pueden modificar en el bloque de datos global. Algunos valores modificables son "Comentario" o "Visible en HMI". Las propiedades que se han modificado no aparecen en color gris en las columnas de la tabla de declaración. Las modificaciones se mantienen aunque el tipo de datos PLC se modifique y el bloque de datos global se actualice después.

Editar las propiedades de un elemento en la tabla de declaración

Para editar las propiedades de un elemento, proceda del siguiente modo:

1. Abra el bloque de datos global.
2. Seleccione el elemento deseado en la tabla.
3. Modifique las entradas en las columnas.

Editar las propiedades de varios elementos en la tabla de declaración

Las columnas "Remanencia", "Visible en HMI", "Accesible desde HMI" y "Valor de ajuste" pueden activarse y desactivarse simultáneamente para varios elementos seleccionados.

Para cambiar una de estas propiedades para varios elementos, proceda del siguiente modo:

1. Abra el bloque de datos.
2. Mantenga pulsada la tecla CTRL.
3. En la columna deseada, seleccione sucesivamente las celdas cuyos valores desee cambiar.
4. Elija los comandos "Activar <propiedad>" o "Desactivar <propiedad>" del menú contextual.

Editar las propiedades en la ventana de propiedades

Para editar las propiedades de una variable en particular, proceda del siguiente modo:

1. Seleccione una variable de la tabla.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre la ventana de propiedades. Muestra las propiedades de la variable en las áreas "General" y "Atributos".
3. Seleccione el área deseada en la navegación local.
4. Modifique las entradas en los campos de entrada.

Inicializar propiedades individuales al valor predeterminado

Para inicializar propiedades individuales de variables al valor definido en el tipo de datos PLC como valor predeterminado, proceda del siguiente modo:

1. Seleccione un valor modificado en la tabla.
2. Borre el valor.
   Se introduce el valor predeterminado obtenido en el tipo de datos PLC. El valor predeterminado se representa en color gris.

Consulte también

Propiedades de variables en bloques de datos (Página 2953)

Editar la tabla de declaración de bloques de datos (Página 2958)

Editar la tabla de declaración de bloques de datos

Insertar filas en la tabla

Procedimiento

Para insertar una fila encima de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila sobre la cual desea insertar una fila nueva.
2. Haga clic en el botón "Insertar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva sobre la seleccionada.

Consulte también

Editar tablas (Página 322)

Agregar filas en la tabla

Procedimiento

Para agregar una fila nueva detrás de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila tras la que desea insertar la fila nueva.
2. Haga clic en el botón "Agregar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva y vacía detrás de la fila seleccionada.

Consulte también

Editar tablas (Página 322)

Borrar variables

Requisitos

Hay un bloque de datos global abierto.

Procedimiento

Para borrar una variable, proceda del siguiente modo:

1. Seleccione la fila que contiene la variable que desea borrar. También puede seleccionar varias filas simultáneamente manteniendo pulsada la tecla <Ctrl> y haciendo clic consecutivamente en las filas que desea seleccionar, o bien manteniendo pulsada la tecla <Mayús> y haciendo clic en la primera y en la última fila.
2. Elija el comando "Borrar" del menú contextual.

Nota

La estructura de los bloques de datos instancia y globales basados en un tipo de datos PLC no se puede modificar directamente, puesto que la estructura de estos bloques está definida en el objeto de orden superior.

El tipo de bloque de datos está introducido en las propiedades del bloque.

Consulte también: Mostrar y editar propiedades de bloques (Página 2613)

Consulte también

Editar tablas (Página 322)

Rellenar celdas consecutivas automáticamente

Es posible transferir el contenido de una o varias celdas de la tabla a las celdas situadas debajo, rellenando así las celdas consecutivas automáticamente.

Si se rellenan automáticamente las celdas de la columna "Nombre", cada uno de los nombres se complementará con un número correlativo. Por ejemplo, el nombre "Motor" se transforma en "Motor_1".

Es posible definir como área de origen una celda individual o varias celdas, o bien filas completas.

Si la tabla abierta contiene menos filas de las que deben rellenarse, en primer lugar se deben insertar más filas vacías.

Requisitos

- La tabla está abierta.
- Se dispone de filas de declaración suficientes.

Procedimiento

Para rellenar celdas consecutivas automáticamente, proceda del siguiente modo:

1. Seleccione las celdas cuyo contenido desea transferir.
2. Haga clic en el símbolo de rellenar situado en la esquina inferior derecha de la celda.
   El puntero del ratón adopta la forma de una cruz con puntas de flecha.
3. Con el botón del ratón pulsado, arrastre el puntero hacia abajo hasta situarlo sobre las celdas que desea rellenar automáticamente.
4. Suelte el botón del ratón.
   Las celdas se rellenan automáticamente.
5. Si ya existen entradas en las celdas que van a rellenarse automáticamente, aparece un cuadro de diálogo. En este diálogo puede indicarse si se quieren sobrescribir las entradas existentes o si se quieren insertar nuevas filas para las variables nuevas.

Mostrar y ocultar las columnas de la tabla

Las columnas de las tablas pueden mostrase u ocultarse según necesidad.

Procedimiento

Para mostrar u ocultar las columnas de una tabla, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna.
2. Elija el comando "Mostrar/ocultar" del menú contextual.
   Aparece una selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación correspondiente.
4. Para ocultar una columna, desactive la casilla de verificación correspondiente.

Editar variables con editores externos

Para editar variables individualmente en editores externos al TIA Portal, éstas se pueden exportar o importar mediante "Copiar y Pegar". Sin embargo, las variables estructuradas no se pueden copiar en un editor.

Requisitos

Hay un bloque de datos y un editor externo abiertos.

Procedimiento

Para exportar variables PLC y volverlas a importar individualmente con Drag & Drop, proceda del siguiente modo:

1. Seleccione una o más variables.
2. Elija el comando "Copiar" del menú contextual.
3. Cambie al editor externo y pegue la variable copiada.

4. Edite las variables de la forma necesaria.
5. Copie las variables en el editor externo.
6. Cambie nuevamente a la tabla de declaración.
7. Elija el comando "Pegar" del menú contextual.

Observar valores de datos online

Observar valores de datos en bloques de datos online

Los valores de datos que adoptan actualmente las variables en la CPU pueden observarse directamente en la tabla de declaración.

Requisitos

- Existe una conexión online.
- El bloque de datos está cargado en la CPU.
- La edición del programa está activa (CPU en "RUN")
- El bloque de datos está abierto

Procedimiento

Para observar los valores de datos, proceda del siguiente modo:

1. Inicie la observación haciendo clic en el botón "Observar todo".
   En la tabla aparece la columna adicional "Valor de observación". En ella se visualizan los valores de datos actuales.
   Consulte también: Estructura de la tabla de declaración de bloques de datos (Página 2939)
2. Para finalizar la observación, vuelva a hacer clic en el botón "Observar todas las variables".

Visualizar valores de datos cargados del dispositivo

Al cargar un bloque de datos desde un dispositivo, los valores de las variables actuales también se cargan. Estos valores se pueden visualizar.

Requisitos

Se ha cargado un bloque de datos del dispositivo.

Procedimiento

Para visualizar los valores actuales, proceda del siguiente modo:

1. Abra el bloque de datos.
2. Haga clic en el encabezado de una columna.
3. Elija el comando "Mostrar/ocultar columnas" del menú contextual.
   Aparece una selección de las columnas disponibles.
4. Active la casilla de verificación de la columna "Instantánea".

Resultado

Los valores actuales se visualizan en la columna "Instantánea".

Nota

Si más adelante se modifica la estructura del bloque de datos, se pierde la vista de los valores actuales. La columna "Instantánea" queda vacía.

Ajustar los valores de datos durante la puesta en marcha

Principios básicos para adaptar los valores de datos durante la puesta en marcha

Introducción

Durante la puesta en marcha de una instalación, a menudo es necesario ajustar valores de datos para adaptar el programa de forma óptima a las condiciones locales. Para tal fin, la tabla de declaración para bloques de datos ofrece las funciones siguientes:

- Inicializar valores de datos en el estado operativo "RUN"
  Esta función permite modificar valores de datos online con el fin de determinar rápidamente los valores de variables óptimos.
- Aplicar valores de variables del programa online como valores de arranque para el programa offline
  Si se determinan los valores de variables óptimos, estos pueden adoptarse en el programa offline como valores de arranque. De esta forma, se garantiza que el programa arranque con valores optimizados después de la siguiente carga.

Para poder utilizar estas funciones, defina primero determinadas variables como "Valores de ajuste" en el programa. Los valores de ajuste son valores que requieren un ajuste fino en la puesta en marcha.

Consulte también

Observar valores de datos online (Página 2961)

Definir valores de arranque (Página 2948)

Marcar datos como valores ajustables

Determinadas variables pueden marcarse como "Valores de ajuste" en el programa. Los valores de ajuste son valores que requieren un ajuste fino en la puesta en marcha.

Reglas

Es posible marcar variables como "Valor de ajuste" en los siguientes tipos de bloque:

- En bloques de función (FB), pero solo en la sección "Static"
- En bloques de datos globales (DB)
- En tipos de datos PLC (UDT)
  No obstante, el ajuste solo es efectivo en tipos de datos PLC cuando el UDT se utiliza en la sección "Static" de un bloque de función o datos.

No es posible definir valores de ajuste en los siguientes tipos de bloque:

- En bloques de datos basados en un tipo de datos PLC y en bloques de datos de instancia. Estos heredan el ajuste de FB o UDT de orden superior.
- No es posible marcar variables de bloques de datos ARRAY como "Valor de ajuste".
- En el punto de llamada a una multiinstancia tampoco es posible marcar variables como "Valor de ajuste". Es necesario realizar el ajuste en la interfaz del bloque de función llamado como multiinstancia.
- En los bloques con protección de know-how no puede modificarse la marca "Valor de ajuste". Para ello es preciso eliminar primero la protección de know-how.

Requisitos

Están abiertos un bloque de función, un bloque de datos global o un tipo de datos PLC (UDT).

Procedimiento

Para marcar una variable como "Valor de ajuste", proceda del siguiente modo:

1. Seleccione una variable de la sección "Static".
2. Active la casilla de verificación en la columna "Valor de ajuste".
   - No es posible definir el elemento superior de una estructura o un tipo de datos PLC como "Valor de ajuste". Es necesario realizar el ajuste para cada uno de los elementos subordinados por separado.
   - En los ARRAY solo se puede marcar como "Valor de ajuste" el elemento de orden superior. Los elementos subordinados heredan el ajuste.
   - En los ARRAY of STRUCT solo es posible marcar como valores de ajuste los elementos por debajo de la primera estructura. Los elementos de otras estructuras heredan el ajuste.

Resultado

Las variables están marcadas como valores de ajuste. Durante la puesta en marcha, estas variables pueden inicializarse online. Para ello, no es necesario poner la CPU en el estado operativo "STOP", sino que puede permanecer en "RUN". Asimismo, los valores de variables actuales pueden transferirse como valores de arranque al programa offline y guardarse allí.

Consulte también

Observar valores de datos online (Página 2961)

Definir valores de arranque (Página 2948)

Inicializar valores de ajuste en el programa online

Principios básicos para inicializar valores de ajuste

Todas las variables marcadas como "valor de ajuste" pueden inicializarse con valores nuevos en el programa online. Para ello, los valores de arranque del programa offline se cargan en el programa online. La CPU permanece en el estado operativo "RUN". Todas las variables marcadas como valor de ajuste se inicializan una vez en el siguiente punto de control del ciclo. Esto es válido tanto para variables remanentes como para no remanentes. A continuación se sigue procesando el programa con los nuevos valores de variables.

⚠ PELIGRO

Peligro causado por la modificación de valores de variables

La modificación de los valores de variables con la instalación en marcha puede causar graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

Asegúrese de que no pueden darse estados peligrosos antes de reinicializar los valores de ajuste.

Requisitos

- Existe una conexión online con la CPU.
- La estructura del bloque de datos es idéntica tanto offline como online.
- Una o varias variables están marcadas como "valor de ajuste".

Procedimiento

Para inicializar todos los valores de ajuste del bloque de datos, proceda del siguiente modo:

1. Abra un bloque de datos global o un bloque de datos de instancia.
2. Introduzca los valores deseados en la columna "Valor de arranque". Los valores de arranque deben corresponderse con los tipos de datos indicados.
3. Haga clic en el botón "Inicializar valores de ajuste".

Resultado

Los valores de ajuste del programa online se inicializan en el siguiente punto de control del ciclo con los valores de arranque del programa offline.

El número máximo de variables inicializables depende de la CPU. Si el número de valores de ajuste marcados es demasiado elevado se notificará con un aviso. En ese caso, las variables pueden insertarse en una tabla de observación e inicializarse con la función "Forzar" de dicha tabla. Otra posibilidad es cargar todo el bloque de datos. Encontrará más información al respecto en "Consulte también".

Consulte también

Carga de valores modificados (Página 2950)

Aplicar valores del programa online como valores de arranque

Para aplicar valores de variables del programa online como valores de arranque en el programa offline, genere primero una instantánea de los valores de variables del programa online. Seguidamente, aplíquelos al programa offline. Asegúrese de copiar siempre los valores de la instantánea. Aquí no se comprueba si todos los valores provienen de un mismo ciclo.

Los valores de arranque protegidos contra escritura no se sobrescriben.

Para adoptar los valores, por lo general existen las siguientes posibilidades:

- Aplicar los valores de un bloque de datos abierto
  En un bloque de datos abierto es posible aplicar como valores de arranque todos los valores o bien solo los valores de las variables marcadas como "valor de ajuste".
- Aplicar los valores de varios bloques en el árbol del proyecto
  En el árbol del proyecto se pueden aplicar como valores de arranque todos los valores de ajuste o todos los valores remanentes.

Requisitos

- Existe una conexión online con la CPU.
- Como mínimo un bloque de datos está cargado en la CPU.

Procedimiento

Para aplicar en un bloque de datos todos los valores o bien solo los valores de las variables marcadas como "valor de ajuste", proceda del siguiente modo:

1. Abra el bloque de datos.
2. Inicie la observación haciendo clic en el botón "Observar todas las variables".
   En la tabla aparece la columna "Valor de observación". En ella se visualizan los valores de datos actuales.

3. Haga clic en el botón "Instantánea de los valores de observación" de la barra de herramientas.
   Los valores de observación actuales se adoptan en la columna "Instantánea".
4. Haga clic en uno de los siguientes botones de la barra de herramientas:
   – "Aplicar valores de ajuste de la instantánea como valores de arranque"
   – "Aplicar todos los valores de la instantánea como valores de arranque"

Los valores se incorporan de la columna "Instantánea" a la columna "Valor de arranque".

Para aplicar los valores de observación de varios bloques de datos en el árbol del proyecto, proceda del siguiente modo:

1. Marque los bloques en el árbol del proyecto.
2. En el menú contextual, elija el comando "Instantánea de los valores de observación".
   Los valores de observación actuales de todos los bloques marcados se adoptan en la columna "Instantánea".
   Cuando finaliza el proceso, aparece un mensaje en la ventana de inspección.
3. Elija a continuación uno de los comandos siguientes del menú contextual:
   – "Aplicar valores de observación como valores de arranque > Solo valores de ajuste"
   – "Aplicar valores de observación como valores de arranque > Solo valores remanentes"

Los valores se incorporan de la columna "Instantánea" a la columna "Valor de arranque".

Resultado

Los valores de arranque nuevos se guardan en el programa offline.

Nota

Aplicar valores de variables individuales

También es posible transferir de la columna "Instantánea" a la columna "Valores de arranque" aquellos valores de variables individuales que no se han marcado previamente como valor de ajuste. Para ello, copie los valores utilizando los comandos "Copiar" y "Pegar" del menú contextual y péguelos en la columna "Valor de arranque". Tenga en cuenta que solo se copian los valores que se encuentran en el área visible de la tabla en estos momentos.

Consulte también

Principios básicos de los valores de arranque (Página 2948)

Definir valores de arranque (Página 2949)

Observar valores de datos online (Página 2961)

9.1.4.4 Programar tipos de datos PLC

Principios básicos de los tipos de datos PLC

Descripción

Los tipos de datos PLC son estructuras de datos definidas por el usuario, que pueden utilizarse varias veces en el programa. La estructura de un tipo de datos PLC consta de varios componentes, que pueden pertenecer a diferentes tipos de datos. El tipo de los componentes se define en la declaración del tipo de datos PLC.

En las CPU de las series S7-1200 o S7-1500 pueden crearse hasta 65534 tipos de datos PLC. Cada uno de estos tipos de datos PLC puede contener hasta 252 componentes.

Los tipos de datos PLC pueden utilizarse para los siguientes fines:

- Los tipos de datos PLC pueden utilizarse como tipos de datos de variables en la declaración de variables de bloques de código, o en bloques de datos.
- Los tipos de datos PLC pueden usarse como plantilla para la creación de bloques de datos globales con la misma estructura de datos.
- En S7-1200 y S7-1500 es posible utilizar tipos de datos PLC como plantilla para crear variables PLC estructuradas.

Consulte también

Crear tipos de datos PLC (Página 2968)

Estructura de la tabla de declaración para los tipos de datos PLC

Estructura de la tabla de declaración para los tipos de datos PLC

La siguiente figura muestra la estructura de la tabla de declaración para tipos de datos PLC.

	Nombre	Tipo de datos	Valor predet.	Visible en HMI	Comentario
	Motor	Bool	false	☑	
	MyTag1	Struct		☑	
	Element1	Bool	false	☑	
	Element2	Bool	false	☑	
	MyTag2	Bool	false	☑	

Significado de las columnas

En la siguiente tabla se explica el significado de las distintas columnas. Las columnas pueden mostrarse u ocultarse según necesidad. El número de columnas visualizadas varía en función de la familia de CPU.

Columna	Significado
	Símbolo en el que se puede hacer clic para mover o copiar la variable.
Nombre	Nombre de la variable.
Tipo de datos	Tipo de datos de la variable.
Valor predeterminado	Valor predeterminado para la variable en la declaración del tipo de datos PLC. La indicación del valor predeterminado es opcional. Si no se especifica ningún valor, se utilizará el valor predefinido para el tipo de datos indicado. Por ejemplo, el valor predefinido para BOOL es "false".
Visible en HMI	Indica si la variable está visible en la lista desplegable de HMI mediante un ajuste predeterminado.
Accesible desde HMI	Indica si HMI puede acceder a esta variable en tiempo de ejecución.
Valor de ajuste	Los valores de ajuste son valores que requieren un ajuste fino en la puesta en marcha. Tras la puesta en marcha, los valores de estas variables pueden transferirse como valores de arranque al programa offline y guardarse allí.
Comentario	Comentario para documentar la variable.

Consulte también

Crear tipos de datos PLC (Página 2968)

Mostrar y ocultar las columnas de la tabla (Página 2975)

Ajustar los valores de datos durante la puesta en marcha (Página 2962)

Crear tipos de datos PLC

Requisitos

La carpeta "Tipos de datos PLC" del árbol del proyecto está abierta.

Procedimiento

Para crear un tipo de datos PLC, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo tipo de datos" de la carpeta "Tipos de datos PLC".
   Se crea y se abre una tabla de declaración nueva para la creación de un tipo de datos PLC.
2. Seleccione el tipo de datos PLC y elija el comando "Cambiar nombre" del menú contextual.
3. Introduzca un nombre para el tipo de datos PLC.

Resultado

Se crea el nuevo tipo de datos PLC. Éste se encuentra en la carpeta "Tipos de datos PLC" del árbol del proyecto.

Consulte también

Estructura de la tabla de declaración para los tipos de datos PLC (Página 2967)

Principios básicos de los tipos de datos PLC (Página 2967)

Borrar tipos de datos PLC

Requisitos

El tipo de datos PLC que desea borrar no está abierto.

Procedimiento

Para borrar un tipo de datos PLC, proceda del siguiente modo:

1. Abra la carpeta "Tipos de datos PLC" en el árbol del proyecto.
2. Seleccione el tipo de datos PLC que se desea borrar. También se pueden seleccionar varios tipos de datos PLC simultáneamente manteniendo pulsada la tecla <Ctrl> y haciendo clic consecutivamente en los tipos de datos que se desea seleccionar, o bien en el primero y en el último.
3. Elija el comando "Borrar" del menú contextual.

Nota

Si se borra un tipo de datos PLC, los bloques que lo utilizan se vuelven incoherentes. Estas incoherencias se marcan en el bloque utilizado con color rojo. Para corregir las incoherencias es necesario actualizar los bloques.

Consulte también:

Actualizar la interfaz del bloque (Página 2656)

Actualizar bloques de datos (Página 2942)

Programar la estructura de los tipos de datos PLC

Declarar variables del tipo de datos simple

Requisitos

Hay un tipo de datos PLC abierto.

Procedimiento

Para declarar una variable, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca el tipo de datos deseado en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
3. Opcionalmente: Modifique las propiedades de las variables que se visualizan en las otras columnas.
4. Repita los pasos 1 a 3 para todas las variables que deban declararse.

Consulte también

Editar tablas (Página 322)

Editar la tabla de declaración para tipos de datos PLC (Página 2974)

Declarar variables del tipo de datos ARRAY

Requisitos

Hay un tipo de datos PLC abierto.

Procedimiento

Para declarar una variable del tipo de datos ARRAY, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca el tipo de datos "Array" en la columna "Tipo de datos". Se dispone de la ayuda de la función de autocompletar.
   Se abre el cuadro de diálogo "Array".
3. Indique el tipo de datos de los elementos Array en el campo de entrada "Tipo de datos".
4. A continuación, indique el límite inferior y superior de cada dimensión en el campo de entrada "Límites de matriz".
   Ejemplo de un ARRAY unidimensional:
   `[0..3]`
   Ejemplo de un ARRAY tridimensional:
   `[0..3, 0..15, 0..33]`
5. Confirme la entrada.
6. Opcionalmente: modifique las propiedades de las variables que se visualizan en las otras columnas.

Nota

Para los elementos ARRAY no se pueden definir valores predeterminados específicos. No obstante, es posible asignarles valores de arranque en el lugar de uso del bloque de datos.

Consulte también

Matriz (Página 2354)

Estructura de la tabla de declaración para los tipos de datos PLC (Página 2967)

Utilizar la función automática de autocompletar (Página 2637)

Declarar variables del tipo de datos STRUCT

Requisitos

Hay un tipo de datos PLC abierto.

Procedimiento

Para declarar una variable del tipo de datos STRUCT, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca "Struct" en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.
   Debajo de las nuevas variables se inserta una fila vacía sangrada.
3. Inserte el primer elemento de estructura en la fila vacía.
   Debajo del elemento se inserta nuevamente una fila vacía.
4. Seleccione un tipo de datos para el elemento de estructura.
5. Opcionalmente: Modifique las propiedades del elemento de estructura que se visualizan en las otras columnas.
6. Repita los pasos 3 a 5 para todos los demás elementos de estructura.
   No es necesario finalizar explícitamente la estructura. La estructura termina con el último elemento introducido.
7. Para insertar una nueva variable después de la estructura, deje una fila vacía al final de la estructura e introduzca la nueva variable en la segunda fila vacía.

Resultado

Se crea la variable del tipo de datos STRUCT.

Consulte también

STRUCT (Página 2358)

Estructura de la tabla de declaración para los tipos de datos PLC (Página 2967)

Utilizar la función automática de autocompletar (Página 2637)

Declarar variables basadas en otro tipo de datos PLC

Requisitos

- Hay un bloque de datos global abierto.
- En la CPU actual está declarado un tipo de datos PLC.

Procedimiento

Para declarar una variable basada en otro tipo de datos PLC, proceda del siguiente modo:

1. Introduzca un nombre de variable en la columna "Nombre".
2. Introduzca el tipo de datos PLC en la columna "Tipo de datos". Al introducirlo, se dispone de la función de autocompletar.

Resultado

La variable se crea.

Nota

Los valores predeterminados de las variables en un tipo de datos PLC se definen al crear el tipo de datos PLC. En la ubicación del tipo de datos PLC, estos valores no se pueden modificar.

Consulte también

Principios básicos de los tipos de datos PLC (Página 2967)

Estructura de la tabla de declaración para los tipos de datos PLC (Página 2967)

Utilizar la función automática de autocompletar (Página 2637)

Editar propiedades de variables en tipos de datos PLC

Propiedades de variables en tipos de datos PLC

Propiedades

La siguiente tabla muestra una vista general de las propiedades de las variables en los tipos de datos PLC:

Grupo	Propiedad	Descripción
General	Nombre	Nombre de la variable.
	Tipo de datos	Tipo de datos de la variable.

Grupo	Propiedad	Descripción
	Valor predeterminado	Valor que se ha predeterminado para la variable en la interfaz de un bloque lógico de orden superior o un tipo de datos PLC. Los valores que contiene la columna "Valores predeterminados" pueden modificarse sólo en el bloque lógico de orden superior o en el tipo de datos PLC. En el bloque de datos, los valores sólo se muestran.
	Valor de arranque	Irrelevante en los tipos de datos PLC
	Comentario	Comentario acerca de la variable.
Atributos	Remanencia	Irrelevante en los tipos de datos PLC
	Visible	Identifica si un parámetro está visible en CFC.
	Parametrizable	Identifica si un parámetro es parametrizable en CFC.
	Para test	Identifica si un parámetro está registrado para el modo Test de CFC.
	Interconectable	Identifica si un parámetro es interconectable en CFC.

Consulte también

Modificar propiedades de variables en tipos de datos PLC (Página 2973)

Principios básicos de los tipos de datos PLC (Página 2967)

Estructura de la tabla de declaración para los tipos de datos PLC (Página 2967)

Modificar propiedades de variables en tipos de datos PLC

Editar propiedades generales en la tabla de declaración

Para editar las propiedades generales de una o varias variables, proceda del siguiente modo:

1. Abra el tipo de datos PLC.
2. Modifique las entradas en las columnas.

Editar las propiedades detalladas en la ventana de propiedades

Para editar propiedades detalladas de una variable en particular, proceda del siguiente modo:

1. Seleccione una variable de la tabla.
2. Elija el comando "Propiedades" del menú contextual.
   La ventana de inspección muestra las propiedades de las variables en las áreas "General" y "Atributos".
3. Seleccione el área deseada en la navegación local.
4. Modifique las entradas en los campos de entrada.

Consulte también

Actualizar la interfaz del bloque (Página 2656)
Actualizar bloques de datos (Página 2942)

Editar la tabla de declaración para tipos de datos PLC

Insertar filas en la tabla

Procedimiento

Para insertar una fila encima de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila sobre la cual desea insertar una fila nueva.
2. Haga clic en el botón "Insertar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva sobre la seleccionada.

Agregar filas en la tabla

Procedimiento

Para agregar una fila nueva detrás de la fila seleccionada, proceda del siguiente modo:

1. Seleccione la fila tras la que desea insertar la fila nueva.
2. Haga clic en el botón "Agregar fila" de la barra de herramientas de la tabla.

Resultado

Se inserta una fila nueva y vacía detrás de la fila seleccionada.

Borrar variables

Procedimiento

Para borrar elementos, proceda del siguiente modo:

1. Seleccione la fila que contiene el elemento que desea borrar. También puede seleccionar varias filas simultáneamente manteniendo pulsada la tecla <Ctrl> y haciendo clic consecutivamente en las filas que desea seleccionar, o bien en la primera y en la última.
2. Elija el comando "Borrar" del menú contextual.

Consulte también

Actualizar la interfaz del bloque (Página 2656)

Actualizar bloques de datos (Página 2942)

Rellenar celdas consecutivas automáticamente

Es posible transferir el contenido de una o varias celdas de la tabla a las celdas situadas debajo, rellenando así las celdas consecutivas automáticamente.

Si se rellenan automáticamente las celdas de la columna "Nombre", cada uno de los nombres se complementará con un número correlativo. Por ejemplo, el nombre "Motor" se transforma en "Motor_1".

Es posible definir como área de origen una celda individual o varias celdas, o bien filas completas.

Si la tabla abierta contiene menos filas de las que deben rellenarse, en primer lugar se deben insertar más filas vacías.

Requisitos

- La tabla está abierta.
- Se dispone de filas de declaración suficientes.

Procedimiento

Para rellenar celdas consecutivas automáticamente, proceda del siguiente modo:

1. Seleccione las celdas cuyo contenido desea transferir.
2. Haga clic en el símbolo de rellenar situado en la esquina inferior derecha de la celda.
   El puntero del ratón adopta la forma de una cruz con puntas de flecha.
3. Con el botón del ratón pulsado, arrastre el puntero hacia abajo hasta situarlo sobre las celdas que desea rellenar automáticamente.
4. Suelte el botón del ratón.
   Las celdas se rellenan automáticamente.
5. Si ya existen entradas en las celdas que van a rellenarse automáticamente, aparece un cuadro de diálogo. En este diálogo puede indicarse si se quieren sobrescribir las entradas existentes o si se quieren insertar nuevas filas para las variables nuevas.

Mostrar y ocultar las columnas de la tabla

Las columnas de las tablas pueden mostrase u ocultarse según necesidad.

Procedimiento

Para mostrar u ocultar las columnas de una tabla, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna.
2. Elija el comando "Mostrar/ocultar" del menú contextual.
   Aparece una selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación correspondiente.
4. Para ocultar una columna, desactive la casilla de verificación correspondiente.

9.1.4.5 Utilizar archivos fuente externos

Principios básicos de la utilización de archivos fuente externos

Función

Para los lenguajes de programación textual AWL y SCL existe la posibilidad de introducir el código de programa en un editor ASCII cualquiera y guardarlo como un archivo fuente externo. Aquí pueden realizarse p. ej. las siguientes acciones:

- Declarar variables
- Definir propiedades de bloques
- Programar bloques

Estos archivos fuente externos se pueden importar al proyecto para generar bloques a partir de ellos. De esta manera se pueden generar varios bloques desde un archivo fuente. Cuando genere bloques a partir de un archivo fuente, tenga en cuenta las siguientes particularidades:

- Si en el proyecto ya existe un bloque con el mismo nombre simbólico, el bloque se sobrescribe en el proyecto.
- Ponga siempre el nombre a los bloques de organización según el tipo, p. ej. "Main" o "Cyclic interrupt", para conservar la asignación de eventos. Si se utilizan varios bloques de organización del mismo tipo, se puede complementar el nombre con guiones bajos y números, p. ej. "Main_2" o "Main_2_1".
- Si un bloque se ha programado en el archivo fuente con su número de bloque absoluto en lugar de con un nombre simbólico, y ese número ya está asignado a otro bloque del proyecto, el bloque recién generado recibirá el siguiente nombre simbólico libre.
- Si en el archivo fuente externo no ha definido explícitamente el tipo de acceso para un bloque, el tipo de acceso para el bloque se ajustará en función de la familia de CPU:
  - Los bloques generados para una CPU de la serie S7-1200/1500 reciben de forma predeterminada el tipo de acceso "optimizado".
  - Los bloques generados para una CPU de la serie S7-300/400 reciben de forma predeterminada el tipo de acceso "estándar".

  Son excepción los bloques de organización que, en ese caso, siempre reciben de forma predeterminada el tipo de acceso "estándar", independientemente de la familia de CPU. El acceso al bloque se puede cambiar manualmente.

- Puede suceder que no todos los comentarios del archivo fuente se adopten en el bloque.
- Si en el archivo de origen externo se utiliza el direccionamiento absoluto, al crear el bloque se genera una variable simbólica para cada dirección absoluta. Los nombres de dichas variables se componen de "Tag_" y un sello de tiempo. De esta composición pueden resultar nombres de variables relativamente largos, que pueden modificarse manualmente en caso necesario.
- Si en el archivo fuente externo utiliza instrucciones de una versión distinta al dispositivo de destino, esto puede provocar errores de compilación. En ese caso, corrija las instrucciones correspondientes e inicie de nuevo el proceso de compilación. Otra posibilidad es seleccionar otra versión para el dispositivo de destino.

Asimismo, existe la posibilidad de guardar bloques existentes como archivos fuente externos.

Consulte también

Reglas para la programación de archivos fuente externos (Página 2977)

Guardar un bloque como archivo fuente externo (Página 2978)

Integrar archivos fuente externos (Página 2979)

Abrir y editar archivos fuente externos (Página 2980)

Generar bloques a partir de archivos fuente externos (Página 2981)

Reglas para la programación de archivos fuente externos

Un archivo fuente externo consta, en principio, de un texto continuo. Sin embargo, para poder compilarlo en bloques se deben respetar estructuras y normas sintácticas determinadas.

Reglas de sintaxis

La sintaxis de las instrucciones en los archivos fuente externos se parece a la utilizada en la creación del programa de usuario en el editor de programación con AWL o SCL. Pero tenga en cuenta las siguientes reglas sintácticas adicionales:

- Llamada de bloque
  Al llamar un bloque, transfiera los parámetros en una secuencia definida en el editor ASCII. De lo contrario, las asignaciones de comentario de estas filas posiblemente no concordarán.
  Indique los parámetros entre paréntesis. Los distintos parámetros se separan entre sí mediante una coma.
- Mayúsculas o minúsculas
  Por lo general, el editor de programación no tiene en cuenta mayúsculas o minúsculas. Excepto en las etiquetas. Al introducir cadenas de caracteres (tipo de datos "STRING") también se deben tener en cuenta las mayúsculas y minúsculas. Las palabras clave se representan en mayúscula. En la compilación no se tienen en cuenta las mayúsculas o minúsculas, así que las palabras clave se pueden indicar en mayúsculas, minúsculas o ambas mezcladas.

- Punto y coma
  Identifique el final de las instrucciones y declaraciones de variables con un punto y coma. Se pueden introducir varias instrucciones por fila.
- Barras inclinadas
  Comience cada comentario con dos barras inclinadas (//) y finalícelo con la tecla <Intro>.

Consulte también

Principios básicos de la utilización de archivos fuente externos (Página 2976)

Guardar un bloque como archivo fuente externo (Página 2978)

Integrar archivos fuente externos (Página 2979)

Abrir y editar archivos fuente externos (Página 2980)

Generar bloques a partir de archivos fuente externos (Página 2981)

Guardar un bloque como archivo fuente externo

En función del lenguaje de programación empleado para el bloque, existen las siguientes posibilidades de guardar bloques como archivo fuente externo:

- AWL y SCL: copiar un bloque como texto.
- SCL: generar un archivo fuente externo a partir de uno o varios bloques.

Copiar un bloque como texto

Para copiar un bloque como texto y exportarlo a un archivo fuente externo, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic con el botón derecho del ratón en el bloque que desea exportar a un archivo fuente externo.
2. Seleccione el comando "Copiar como texto" del menú contextual.
3. Abra un editor de texto externo.
4. En el editor de texto, inserte el texto copiado desde el portapapeles.
5. Guarde el archivo con una de las siguientes extensiones de nombre de archivo:
   – ".scl", si quiere generar un archivo fuente para SCL
   – ".awl", si quiere generar un archivo fuente para AWL

Generar un archivo fuente externo a partir de bloques SCL

Para generar un archivo fuente externo a partir de bloques SCL, proceda del siguiente modo:

1. En el árbol del proyecto o en la ventana general, seleccione los bloques SCL desde los que desea generar un archivo fuente externo.
2. Elija el comando "Generar fuente a partir de bloques" del menú contextual.
   Se abre el cuadro de diálogo "Generar fuente a partir de bloques".

3. Especifique una ruta y un nombre para la fuente externa.
4. Haga clic en "Aceptar".

También se puede generar un archivo fuente externo a partir de un bloque SCL abierto. Proceda para ello del siguiente modo:

1. Haga clic en el botón "Generar fuente a partir de bloques" del editor de programación. Se abre el cuadro de diálogo "Generar fuente a partir de bloques".
2. Especifique una ruta y un nombre para la fuente externa.
3. Haga clic en "Aceptar".

Resultado

El bloque se ha guardado como archivo fuente externo. Este archivo fuente se puede integrar en un proyecto en el TIA Portal para generar volver a generar bloques a partir de él. Pero tenga en cuenta que los archivos fuente AWL solo pueden utilizarse en las CPU S7-300/400/1500.

Consulte también

Principios básicos de la utilización de archivos fuente externos (Página 2976)

Reglas para la programación de archivos fuente externos (Página 2977)

Integrar archivos fuente externos (Página 2979)

Abrir y editar archivos fuente externos (Página 2980)

Generar bloques a partir de archivos fuente externos (Página 2981)

Integrar archivos fuente externos

Requisitos

- Existe un archivo fuente externo y se corresponde con las reglas de sintaxis y estructura.
- La carpeta "Fuentes externas" del árbol del proyecto está abierta.

Procedimiento

Para integrar un archivo fuente externo, proceda del siguiente modo:

1. Haga doble clic en el comando "Agregar nuevo archivo externo". Se abre el cuadro de diálogo "Abrir".
2. Navegue hasta el archivo fuente externo existente y selecciónelo.
3. Confirme la selección con "Abrir".

Resultado

El nuevo archivo fuente se inserta en la carpeta "Fuentes externas".

Consulte también

Principios básicos de la utilización de archivos fuente externos (Página 2976)

Reglas para la programación de archivos fuente externos (Página 2977)

Guardar un bloque como archivo fuente externo (Página 2978)

Abrir y editar archivos fuente externos (Página 2980)

Generar bloques a partir de archivos fuente externos (Página 2981)

Abrir y editar archivos fuente externos

Si enlaza archivos que cuentan con las extensiones "awl" y "scl" con un editor, existe la posibilidad de abrir y editar directamente los archivos de origen externos con estos formatos. Como editor se debe utilizar Notepad, ya que con otros editores de texto posiblemente no puedan abrirse varias fuentes al mismo tiempo.

De esta manera no es necesario volver a integrar los archivos fuente externos después de editarlos.

Enlazar archivos con las extensiones "awl" y "scl" con un editor

Para enlazar los archivos con las extensiones "awl" y "scl" con un editor, proceda del siguiente modo:

1. Abra el Explorador de Windows.
2. Haga clic con el botón derecho del ratón en un archivo AWL.
3. Elija "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo "Propiedades".
4. Haga clic en "Cambiar" en el área "Tipo de datos" de la ficha "General".
   Se abre el cuadro de diálogo "Abrir con".
5. Elija el editor de texto que desea enlazar con el tipo de archivo "awl".
6. Confirme la selección con "Aceptar".
7. Cierre el cuadro de diálogo "Propiedades" con "Aceptar".
8. Repita los pasos 2 a 7 con un archivo SCL.

Abrir y editar archivos fuente externos

Para abrir un archivo fuente externo, proceda del siguiente modo:

1. Abra la carpeta "Fuentes externas" del árbol del proyecto.
2. Haga doble clic en el archivo fuente externo que desea abrir.
   El archivo fuente externo se abre en el editor enlazado y se puede editar.

Consulte también

Principios básicos de la utilización de archivos fuente externos (Página 2976)

Reglas para la programación de archivos fuente externos (Página 2977)

Guardar un bloque como archivo fuente externo (Página 2978)

Integrar archivos fuente externos (Página 2979)

Generar bloques a partir de archivos fuente externos (Página 2981)

Generar bloques a partir de archivos fuente externos

Requisitos

- La carpeta "Fuentes externas" del árbol del proyecto está abierta.
- Existe un archivo fuente externo.

Procedimiento

Para generar bloques desde un archivo fuente externo, proceda del siguiente modo:

1. Seleccione el archivo fuente externo desde el que desea generar los bloques.
2. Elija el comando "Generar fuente a partir de bloques" del menú "Edición".
3. Una consulta de seguridad le informa de que los bloques existentes posiblemente se sobrescribirán.
4. Confirme la consulta de seguridad con "Sí".

Resultado

Los bloques del archivo fuente externo se generan e insertan en la carpeta "Bloques de programa" del árbol del proyecto. Si se producen errores, encontrará información sobre este respecto en la ventana de inspección. Sin embargo, esta información se refiere al archivo fuente externo y no al bloque generado.

Consulte también

Definir propiedades de bloques (Página 2607)

Principios básicos de la utilización de archivos fuente externos (Página 2976)

Reglas para la programación de archivos fuente externos (Página 2977)

Guardar un bloque como archivo fuente externo (Página 2978)

Integrar archivos fuente externos (Página 2979)

Abrir y editar archivos fuente externos (Página 2980)

9.1.5 Comparar programas de PLC

9.1.5.1 Principios básicos para comparar programas de PLC

Introducción a la comparación de programas de PLC

Función

Existe la posibilidad de comparar los siguientes objetos de un programa de PLC entre sí para determinar posibles diferencias:

- Bloques lógicos con otros bloques lógicos
- Bloques de datos con otros bloques de datos
- Variables PLC de una tabla de variables PLC con las variables PLC de otra tabla de variables PLC
- Tipos de datos PLC con otros tipos de datos PLC

Tipos de comparación y grados de comparación

En principio, se pueden utilizar dos tipos de comparación diferentes:

- Comparación online/offline:
  Los objetos del proyecto se comparan con los objetos del dispositivo respectivo. Para ello es necesario una conexión online con el dispositivo.
- Comparación offline/offline:
  Se comparan los objetos de dos dispositivos dentro de un proyecto o de varios proyectos o librerías. Para esta comparación no es necesaria una conexión online.

Tenga en cuenta que no es posible realizar simultáneamente tantas comparaciones como se desee, sino que solo es posible una por tipo de comparación (online/offline u offline/offline).

Dependiendo de la profundidad con la que se desee comparar los objetos, se puede elegir entre los siguientes grados de comparación:

- Editor de comparación
- Comparación detallada

Si se inicia una comparación, se obtiene un resumen en el editor de comparación, como primera medida. Para algunos objetos es posible iniciar a continuación una comparación detallada, en la que los objetos comparados se abren uno junto al otro, cada uno de ellos en una instancia propia del editor de programación. Si existen diferencias, se resaltan.

La siguiente tabla muestra un resumen sobre el tipo y grado de comparación que se puede emplear para los distintos objetos:

Objeto	Online/offline		Offline/offline	
Editor de comparación	Comparación detallada	Editor de comparación	Comparación detallada	
Bloque KOP	X	X	X	X

Objeto	Online/offline		Offline/offline	
Bloque FUP	X	X	X	X
Bloque AWL[1]	X	X	X	X
Bloque SCL	X	X[3 4]	X	X
Bloque GRAPH[2]	X	X[4]	X	X[5]
Bloque de datos globales	X	X	X	X
Bloque de datos de instancia	X	X	X	X
Variables PLC	-	-	X	X
Tipo de datos PLC	X[4]	X[4]	X	X

Leyenda:
X: Disponible
-: No disponible
[1]: AWL no está disponible para S7-1200
[2]: GRAPH no está disponible para S7-1200
[3]: no para S7-1200 anterior a la versión 2.0
[4]: no para S7-300/400
[5]: comparable solo con un bloque GRAPH de la misma familia de CPU

Nota

Tenga en cuenta las indicaciones siguientes:

- No es posible llevar a cabo comparaciones detalladas para los bloques con protección de know-how.
- Si en la comparación detallada las diferencias solo están en los tipos de datos de las variables locales, siendo que offline es un tipo de datos de alarma (C_ALARM C_ALARM_S C_ALARM_8 C_ALARM_8P C_ALARM_T C_AR_SEND C_NOTIFY C_NOTIFY_8P) y online una DWORD, la diferencia no se marca como tal.
- No pueden realizarse comparaciones detalladas de tipos y plantillas maestras de librerías.

Consulte también

Principios básicos de la comparación de datos de proyecto (Página 373)

Comparación de bloques lógicos (Página 2984)

Comparación de bloques de datos (Página 2985)

Comparación de variables PLC y tipos de datos PLC (Página 2985)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realizar una comparación detallada de bloques (Página 2988)

Comparación de bloques lógicos

Introducción

En la comparación de los bloques lógicos, los bloques que deben compararse se asignan entre sí según los siguientes criterios:

- Comparación online/offline: Direcciones, p. ej. FB 100
- Comparación offline/offline: Nombres simbólicos de los bloques

En la comparación se evalúan los sellos de tiempo de los bloques y se visualiza un resumen de los resultados en el editor de comparación. A continuación se puede definir mediante acciones cómo se deben tratar las diferencias. Además, es posible iniciar comparaciones detalladas para los distintos bloques. Aquí se abren las versiones comparadas de un bloque y se resaltan las diferencias.

En la comparación de bloques lógicos se incluyen tanto sus interfaces como los distintos segmentos. Además, se determinan las diferencias en los nombres de variables. En una comparación online/offline quedan excluidos todos los comentarios y cualquier otro atributo de bloque.

Si se modifica la interfaz del bloque, cambia el sello de tiempo de la interfaz del bloque del bloque lógico. Esta modificación también ocasiona un cambio del sello de tiempo del código del programa. Por eso, para la comparación de las interfaces del bloque se comparan, en primera instancia, los sellos de tiempo del código de programa. Si los sellos de tiempo son iguales, se parte del principio de que las interfaces también lo son. Si los sellos de tiempo de las interfaces son diferentes, a continuación se comparan los tipos de datos de las interfaces por secciones. Las multiinstancias y los tipos de datos PLC se incluyen en la comparación. Si los tipos de datos de las distintas secciones son idénticos, se comparan los valores de arranque de las variables. Se indican todas las diferencias existentes.

Al comparar los segmentos, primero se determinan los segmentos insertados o borrados. Luego se comparan los demás segmentos. Las instrucciones son iguales si el operador y el operando son iguales. Sólo se muestra la primera diferencia de cada instrucción. Sin embargo, pueden visualizarse varias diferencias por segmento.

Consulte también

Introducción a la comparación de programas de PLC (Página 2982)

Comparación de bloques de datos (Página 2985)

Comparación de variables PLC y tipos de datos PLC (Página 2985)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realizar una comparación detallada de bloques (Página 2988)

Comparación de bloques de datos

Introducción

En la comparación de los bloques de datos, los bloques que deben compararse se asignan entre sí según los siguientes criterios:

- Comparación online/offline: Direcciones, p. ej. DB 100
- Comparación offline/offline: Nombres simbólicos de los bloques

En la comparación de bloques de datos se comparan sus sellos de tiempo. Si son iguales, se parte del principio de que las estructuras de datos también lo son. Si los sellos de tiempo son diferentes, las estructuras se comparan hasta la primera diferencia en la estructura de datos. Si la estructura de datos de las distintas secciones es igual, se comparan los valores iniciales y actuales. Se indican todas las diferencias existentes. Además, se determinan las diferencias en los nombres de variables. Los comentarios y estructuras de los tipos de datos PLC utilizados en el bloque de datos quedan excluidos de la comparación.

Consulte también

Introducción a la comparación de programas de PLC (Página 2982)

Comparación de bloques lógicos (Página 2984)

Comparación de variables PLC y tipos de datos PLC (Página 2985)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realizar una comparación detallada de bloques (Página 2988)

Comparación de variables PLC y tipos de datos PLC

Introducción

Si se lleva a cabo una comparación offline/offline, las tablas de variables PLC y los tipos de datos PLC de los dispositivos también se visualizan en el editor de comparación. En ese caso, las tablas de variables PLC y los tipos de datos PLC se asignan entre sí en función de su nombre y se proporciona la siguiente información:

- Estado: Se marca mediante un símbolo si las variables PLC/tipos de datos PLC son idénticos o si presentan diferencias.
- Tablas de variables PLC/tipos de datos PLC ausentes: Se puede ver a simple vista si las tablas de variables PLC/tipos de datos PLC existen en ambos dispositivos.

En una comparación online/offline, se obtiene la siguiente información para las CPU de la serie S7-1200/1500:

- Variables PLC: Se muestra mediante un símbolo si las variables PLC son idénticas o si presentan diferencias. Puesto que las tablas de variables PLC no pueden cargarse en el dispositivo durante el proceso de carga, no pueden mostrarse en una comparación online/offline.
- Tipos de datos PLC: Se muestra el símbolo de estado para cada tipo de datos PLC. De este modo, se reconoce a primera vista si existen tipos de datos PLC en los dos dispositivos.

Consulte también

Introducción a la comparación de programas de PLC (Página 2982)

Comparación de bloques lógicos (Página 2984)

Comparación de bloques de datos (Página 2985)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Realizar una comparación detallada de bloques (Página 2988)

9.1.5.2 Comparar bloques

Comparar bloques en el editor de comparación

Existen las siguientes posibilidades de comparar bloques en el editor de comparación:

- Comparación online/offline
  Los bloques del proyecto se comparan con los bloques del dispositivo seleccionado.
- Comparación offline/offline automática
  Todos los bloques de los dispositivos seleccionados se comparan offline.
- Comparación offline/offline manual
  Los bloques seleccionados de los dispositivos se comparan offline.

Realizar una comparación online/offline de bloques

Para realizar una comparación online/offline, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación online/offline.
2. Elija el comando "Comparar > offline/online" del menú contextual.
3. Si hasta ahora no se ha establecido ninguna conexión online para ese dispositivo, aparecerá el cuadro de diálogo "Establecer conexión online". En ese caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Conectar".
   La conexión online se establece y se abre el editor de comparación.
4. Abra la carpeta "Bloques de programa".
   Los símbolos del área de estado y acción indican el estado. Dependiendo del estado de los objetos se pueden definir determinadas acciones. No obstante, tenga en cuenta que en una acción de sincronización solo es posible ejecutar acciones en un sentido.

Realizar una comparación offline/offline automática de bloques

Para realizar una comparación offline/offline automática de bloques, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   El dispositivo comparado puede ser del mismo proyecto, de un proyecto de referencia o de la librería.
4. Abra la carpeta "Bloques de programa".
   Los símbolos del área de estado y acción indican el estado de los objetos. Dependiendo del estado de los objetos se pueden definir determinadas acciones. Si se selecciona un objeto, sus propiedades y las del objeto correspondiente del dispositivo asignado se muestran sinópticamente en la comparación de propiedades.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Realizar una comparación offline/offline manual de bloques

Para realizar una comparación offline/offline manual de bloques, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   El dispositivo comparado puede ser del mismo proyecto, de un proyecto de referencia o de la librería.
4. En el área de estado y acción haga clic en el botón para conmutar entre la comparación automática y manual.
5. Seleccione los objetos que desea comparar.
   Se muestra la comparación de propiedades. Los símbolos indican el estado de los objetos. Dependiendo del estado de los objetos se pueden definir determinadas acciones.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Consulte también

Introducción a la comparación de programas de PLC (Página 2982)

Utilizar el editor de comparación (Página 375)

Realizar una comparación detallada de bloques (Página 2988)

Comparar variables PLC (Página 3009)

Comparar tipos de datos PLC (Página 3010)

Realizar una comparación detallada de bloques

Iniciar la comparación detallada

Es posible iniciar una comparación detallada para los bloques. Las versiones comparadas de un bloque se abren una junto a la otra y se resaltan las diferencias.

Nota

Tenga en cuenta las indicaciones siguientes:

- La comparación detallada no está disponible para los bloques creados en el lenguaje de programación SCL en las CPU de la serie S7-1200 con una versión inferior a 2.0.
- En los bloques creados con el lenguaje de programación GRAPH, la comparación detallada solo se puede iniciar mediante el editor de comparación.

Iniciar la comparación detallada mediante el editor de comparación

Para iniciar una comparación detallada de un bloque mediante el editor de comparación, proceda del siguiente modo:

1. Realice una comparación online/offline u offline/offline.
   El editor de comparación se abre.
2. En el editor de comparación, seleccione el bloque para el que desea realizar una comparación detallada.
3. Haga clic en el botón "Iniciar comparación detallada" de la barra de herramientas.

Iniciar la comparación detallada en el editor de programación

Para el tipo de comparación online/offline, la comparación detallada se puede iniciar directamente en el editor de programación. Proceda para ello del siguiente modo:

1. Abra el bloque para el que desea realizar una comparación detallada.
2. Establezca una conexión online.
   Consulte también: Establecer y deshacer una conexión online

   Nota

   Tenga en cuenta que el bloque debe estar disponible online para que se pueda iniciar la comparación detallada para el bloque dentro del editor de programación.

3. Haga clic en el botón "Comparación detallada" de la barra de herramientas.
4. Confirme el cuadro de diálogo para cerrar el bloque con "Sí".

Resultado

Para las versiones comparadas entre sí del bloque se abren sendas instancias del editor de programación, dispuestas una junto a la otra. Si existen diferencias, se marcan.

Si existen diferencias estructurales para los bloques GRAPH, en la vista de cadena solo se muestran los resultados de la comparación hasta la primera diferencia estructural.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación offline/offline (Página 374)

Realizar una comparación online/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Representación del resultado de la comparación para KOP/FUP (Página 2989)

Navegar en la comparación detallada (Página 3006)

Modificar bloques durante la comparación detallada (Página 3007)

Actualizar resultados de la comparación (Página 3008)

Representación del resultado de la comparación

Representación del resultado de la comparación para KOP/FUP

Introducción

La comparación detallada permite identificar exactamente las diferencias entre las versiones comparadas de un bloque. Las marcas siguientes permiten encontrar las diferencias rápidamente:

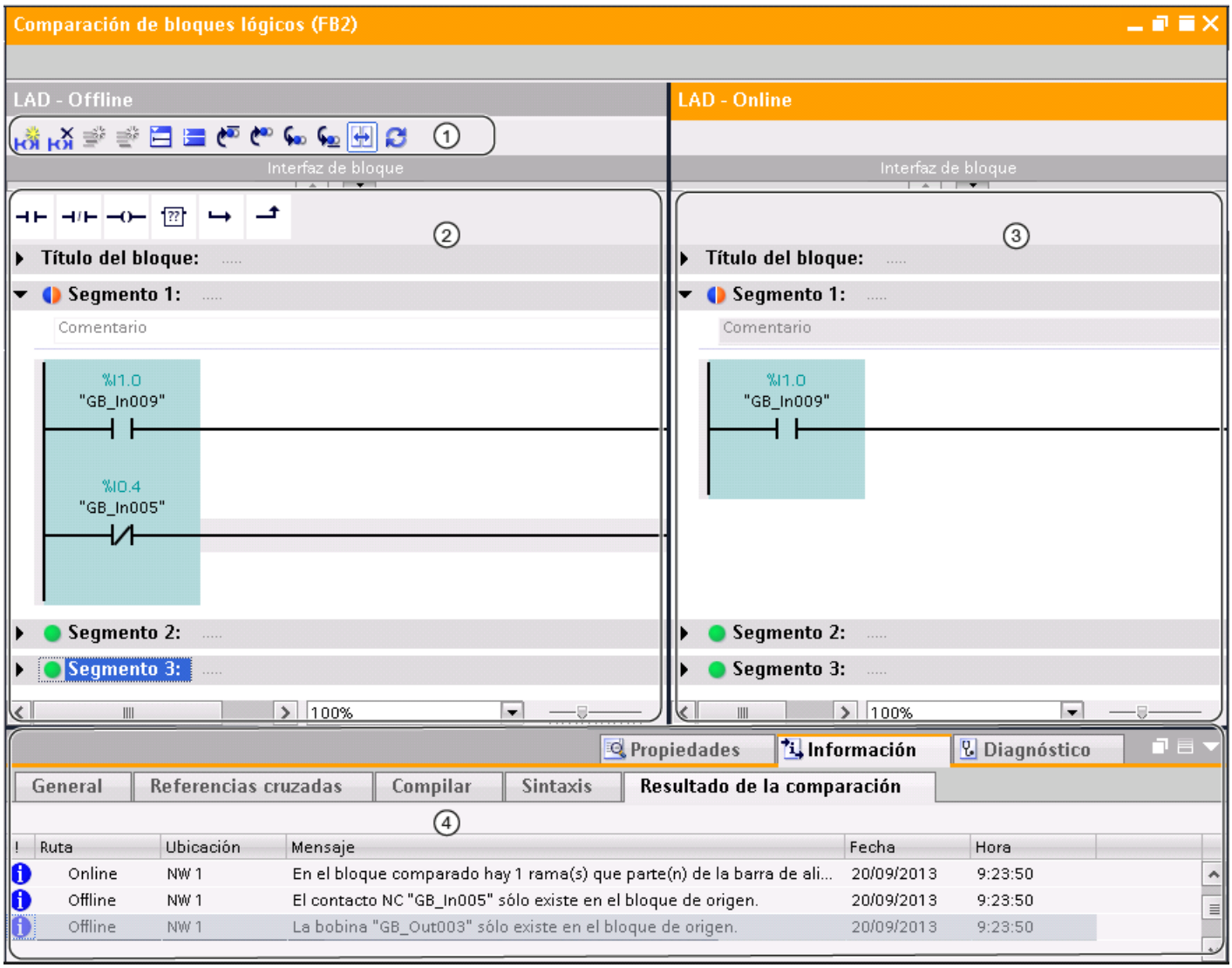
- Las filas que presentan diferencias se destacan con fondo gris.
- Los operandos e instrucciones diferentes se destacan con fondo verde.
- Si el número de segmentos es diferente, se insertan pseudo-segmentos para permitir una representación sincronizada de los segmentos idénticos. Estos pseudo-segmentos se destacan con fondo gris y contienen en su barra de título el texto "No se ha encontrado ningún segmento equivalente.". Los pseudo-segmentos no se pueden editar.

Para facilitar la claridad, no se resaltan todas las diferencias, sino sólo la primera diferencia de cada operación. Por ejemplo, si en una instrucción con varias entradas todas ellas se diferencian en las versiones offline y online de un bloque, solo la primera se resaltará como diferente. El usuario puede entonces solucionar esta diferencia y actualizar la lista de comparación. A continuación, se resalta la siguiente entrada como diferencia.

Por tanto, el número de diferencias resaltadas en un segmento depende del número de instrucciones existentes.

Estructura de la comparación detallada

La figura siguiente muestra un ejemplo de la comparación detallada online/offline para el lenguaje de programación KOP:



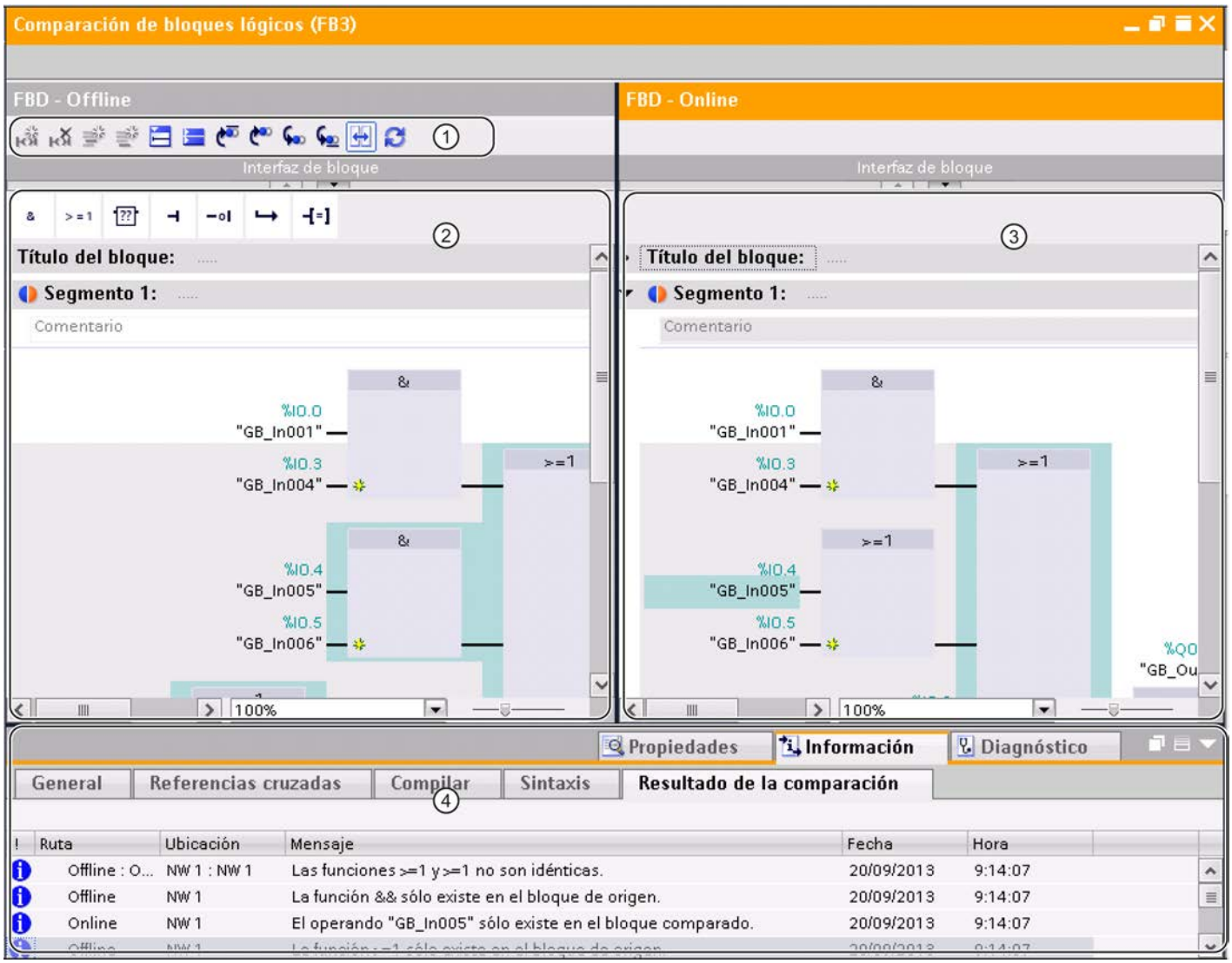
① Barra de herramientas de la comparación detallada para KOP

② Bloque de partida

③ Bloque comparado

④ Resultado de la comparación en la ventana de inspección

La figura siguiente muestra un ejemplo de la comparación detallada online/offline para el lenguaje de programación FUP:



① Barra de herramientas de la comparación detallada para FUP

② Bloque de partida

③ Bloque comparado

④ Resultado de la comparación en la ventana de inspección

Nota

La indicación de los nombres simbólicos de la versión online de un bloque solo es posible para S7-1200 y S7-1500.

Barra de herramientas de la comparación detallada

La barra de herramientas permite acceder a las siguientes funciones:

- Funciones generales
  - Insertar segmento
  - Borrar segmento
  - Insertar fila
  - Agregar fila
  - Abrir todos los segmentos
  - Cerrar todos los segmentos
- Funciones específicas de comparación
  - Posicionar en la primera diferencia
  - Posicionar en la anterior diferencia
  - Posicionar en la siguiente diferencia
  - Posicionar en la última diferencia
  - Sincronizar desplazamiento de los editores
  - Actualizar los resultados de la comparación

Bloque de partida

El bloque de partida se muestra en la ventana izquierda. En una comparación online/offline, el bloque de partida es la versión offline del bloque.

Bloque comparado

El bloque comparado se muestra en la ventana derecha. En una comparación online/offline, el bloque comparado es la versión online del bloque.

Resultado de la comparación en la ventana de inspección

En la ficha "Información > Resultado de la comparación" de la ventana de inspección se muestra una tabla que recoge las diferencias. Haga doble clic en una fila para acceder a la diferencia correspondiente en el bloque.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Navegar en la comparación detallada (Página 3006)

Modificar bloques durante la comparación detallada (Página 3007)

Actualizar resultados de la comparación (Página 3008)

Representación del resultado de la comparación para AWL

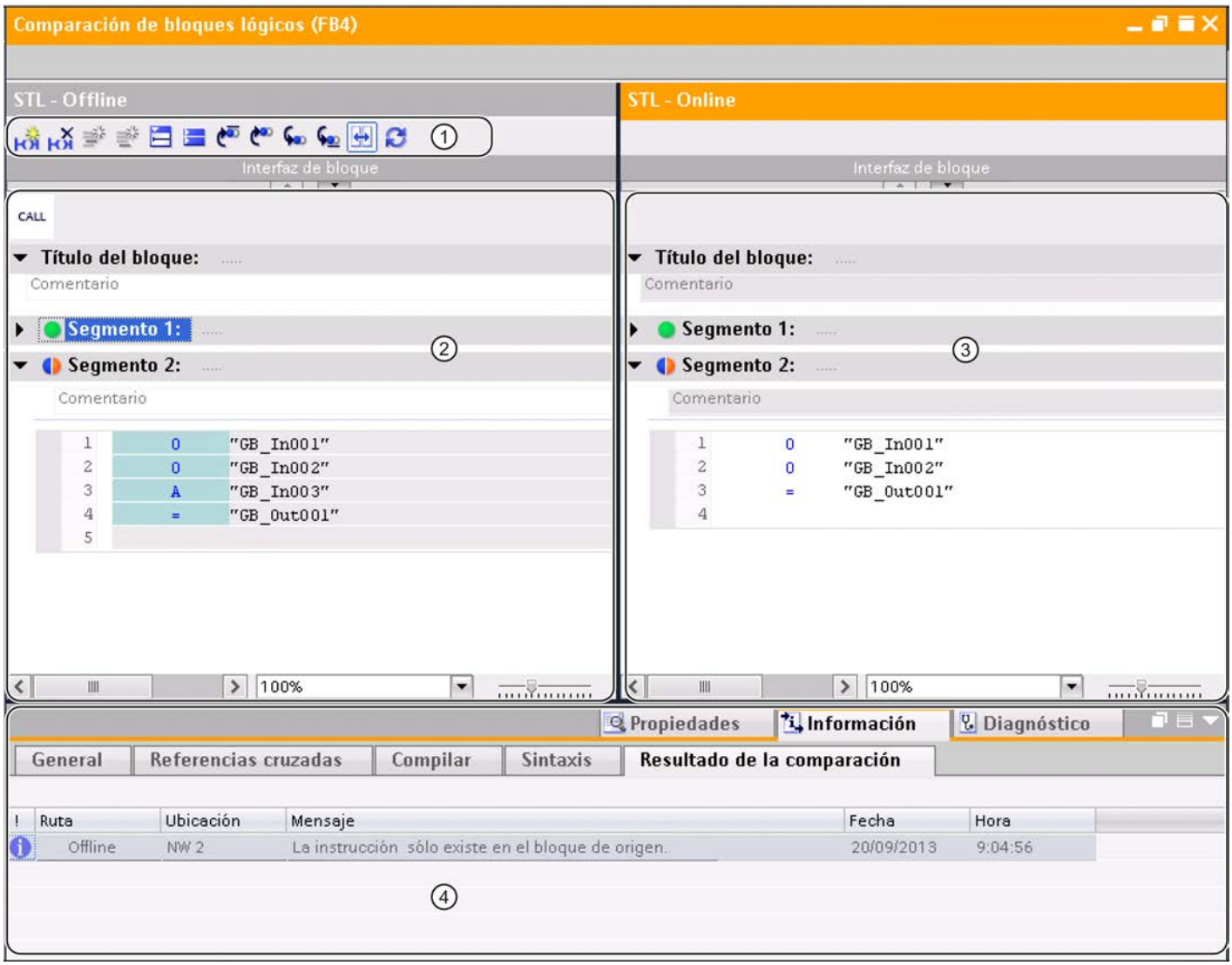
Introducción

La comparación detallada permite identificar exactamente las diferencias entre las versiones comparadas de un bloque. Las marcas siguientes permiten encontrar las diferencias rápidamente:

- Las filas que presentan diferencias se destacan con fondo gris.
- Los operandos e instrucciones diferentes se destacan con fondo verde.
- Si el número de segmentos es diferente, se insertan pseudo-segmentos para permitir una representación sincronizada de los segmentos idénticos. Estos pseudo-segmentos se destacan con fondo gris y contienen en su barra de título el texto "No se ha encontrado ningún segmento equivalente.". Los pseudo-segmentos no se pueden editar.

Estructura de la comparación detallada

La figura siguiente muestra un ejemplo para la comparación detallada online/offline para el lenguaje de programación AWL:



① Barra de herramientas de la comparación detallada para AWL

② Bloque de partida

③ Bloque comparado

④ Resultado de la comparación en la ventana de inspección

Nota

La indicación de los nombres simbólicos de la versión online del bloque solo es posible para S7-1500.

Barra de herramientas de la comparación detallada

La barra de herramientas permite acceder a las siguientes funciones:

- Funciones generales
  - Insertar segmento
  - Borrar segmento
  - Insertar fila
  - Agregar fila
  - Abrir todos los segmentos
  - Cerrar todos los segmentos
- Funciones específicas de comparación
  - Posicionar en la primera diferencia
  - Posicionar en la anterior diferencia
  - Posicionar en la siguiente diferencia
  - Posicionar en la última diferencia
  - Sincronizar desplazamiento de los editores
  - Actualizar los resultados de la comparación

Bloque de partida

El bloque de partida se muestra en la ventana izquierda. En una comparación online/offline, el bloque de partida es la versión offline del bloque.

Bloque comparado

El bloque comparado se muestra en la ventana derecha. En una comparación online/offline, el bloque comparado es la versión online del bloque.

Resultado de la comparación en la ventana de inspección

En la ficha "Información > Resultado de la comparación" de la ventana de inspección se muestra una tabla que recoge las diferencias. Haga doble clic en una fila para acceder a la diferencia correspondiente en el bloque.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Navegar en la comparación detallada (Página 3006)

Modificar bloques durante la comparación detallada (Página 3007)

Actualizar resultados de la comparación (Página 3008)

Representación del resultado de la comparación para SCL

Introducción

La comparación detallada permite identificar exactamente las diferencias entre las versiones comparadas de un bloque. Las marcas siguientes permiten encontrar las diferencias rápidamente:

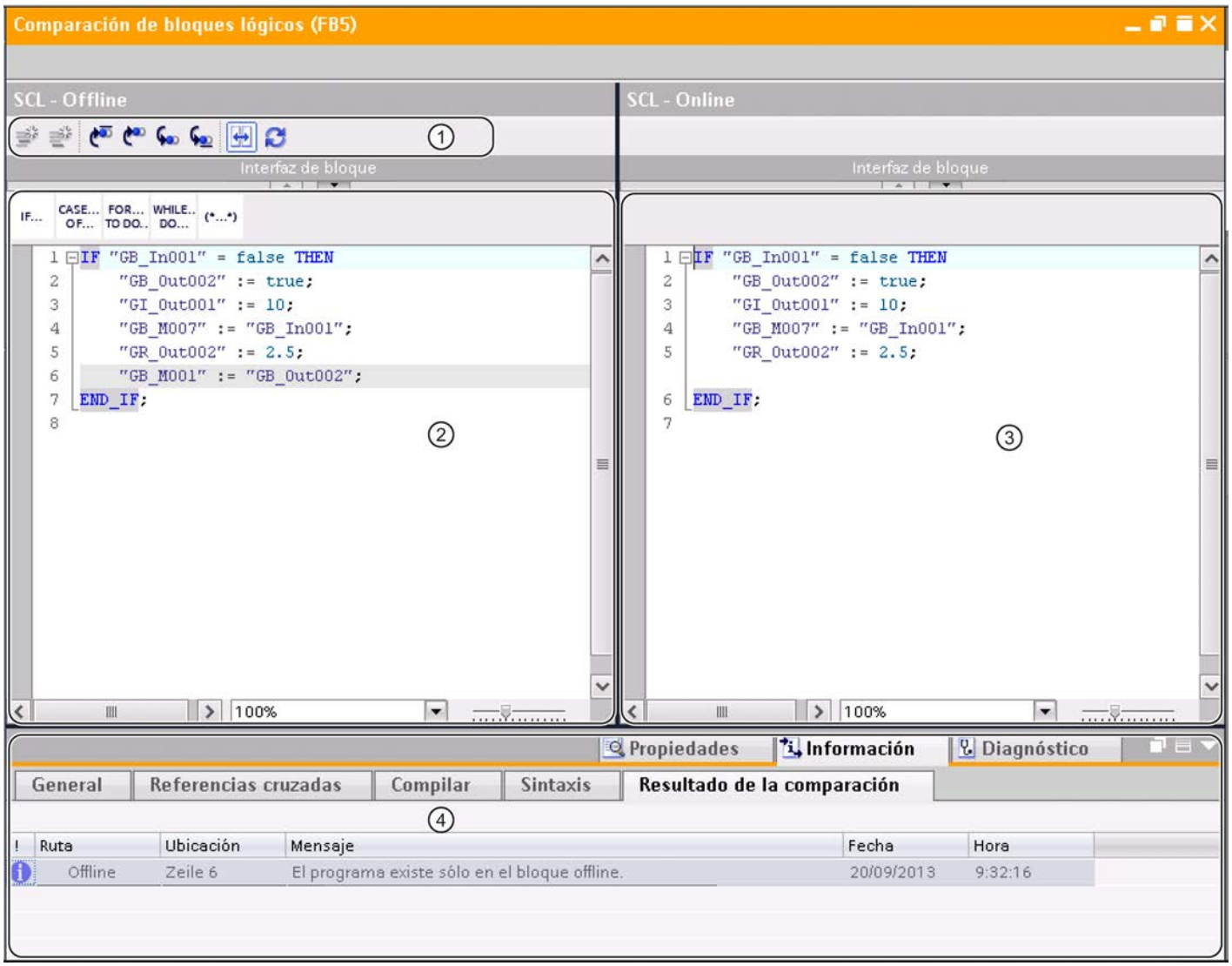
- Las filas que presentan diferencias se destacan con fondo gris.
- Los operandos e instrucciones diferentes se destacan con fondo verde.

Nota

La comparación detallada online/offline no está disponible para SCL para las familias de CPU S7-300/400 ni para S7-1200 con una versión inferior a 2.0.

Estructura de la comparación detallada

La figura siguiente muestra un ejemplo de la comparación detallada online/offline para el lenguaje de programación SCL:



① Barra de herramientas de la comparación detallada para SCL

② Bloque de partida

③ Bloque comparado

④ Resultado de la comparación en la ventana de inspección

Nota

La indicación de los nombres simbólicos de la versión online del bloque solo es posible para S7-1200 y S7-1500.

Barra de herramientas de la comparación detallada

La barra de herramientas permite acceder a las siguientes funciones:

- Funciones generales
  - Insertar fila
  - Agregar fila
- Funciones específicas de comparación
  - Posicionar en la primera diferencia
  - Posicionar en la anterior diferencia
  - Posicionar en la siguiente diferencia
  - Posicionar en la última diferencia
  - Sincronizar desplazamiento de los editores
  - Actualizar los resultados de la comparación

Bloque de partida

El bloque de partida se muestra en la ventana izquierda. En una comparación online/offline, el bloque de partida es la versión offline del bloque.

Bloque comparado

El bloque comparado se muestra en la ventana derecha. En una comparación online/offline, el bloque comparado es la versión online del bloque.

Resultado de la comparación en la ventana de inspección

En la ficha "Información > Resultado de la comparación" de la ventana de inspección se muestra una tabla que recoge las diferencias. Haga doble clic en una fila para acceder a la diferencia correspondiente en el bloque.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Navegar en la comparación detallada (Página 3006)

Modificar bloques durante la comparación detallada (Página 3007)

Actualizar resultados de la comparación (Página 3008)

Representación del resultado de la comparación para GRAPH

Introducción

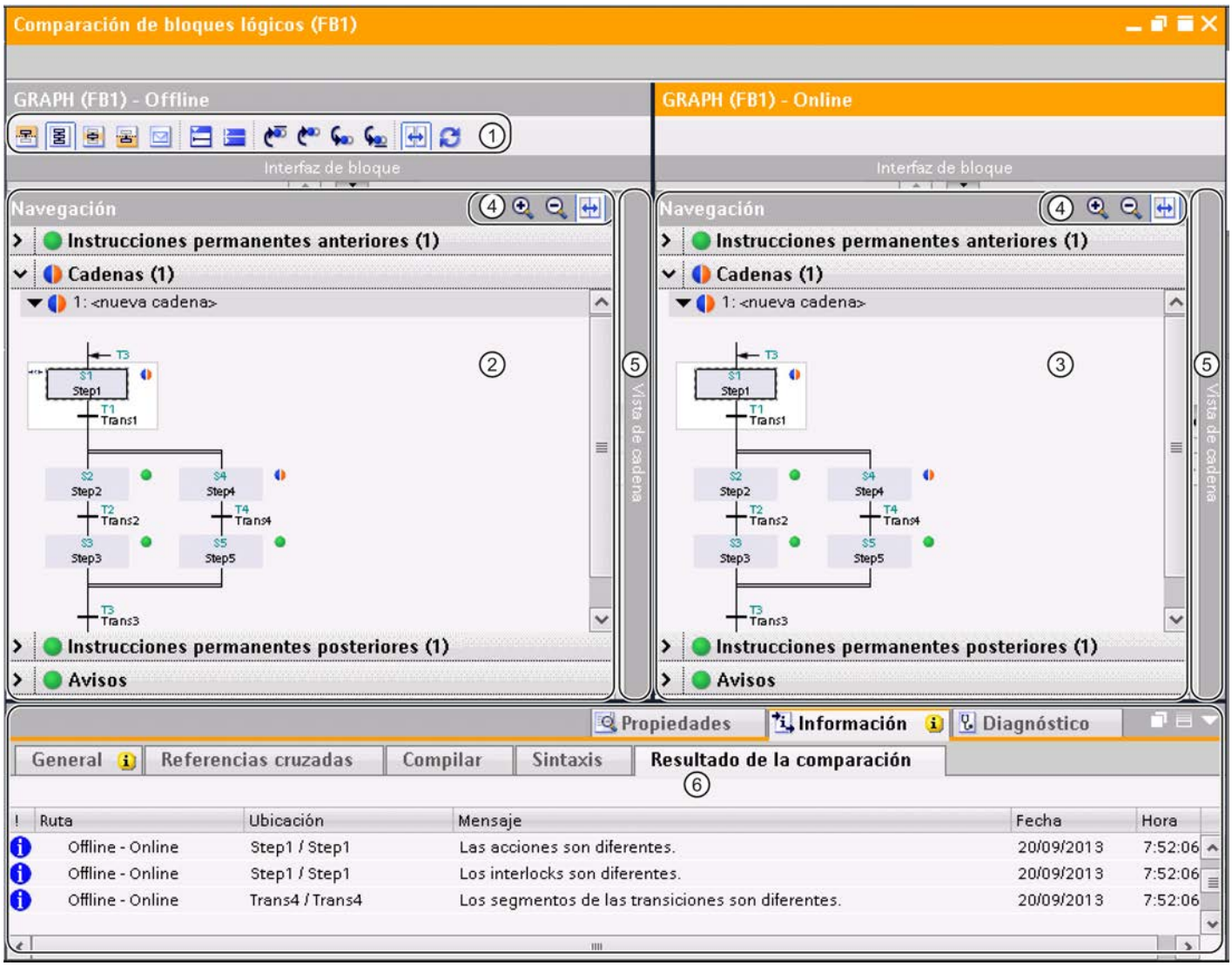
La comparación detallada permite identificar exactamente las diferencias entre las versiones comparadas de un bloque. Al iniciar una comparación detallada para un bloque GRAPH, primero se abre la navegación. El divisor de ventana permite cambiar entre la navegación y la vista configurada actualmente. En la barra de herramientas de la comparación detallada pueden seleccionarse otras vistas.

El resultado de la comparación se marca con los símbolos de comparación.

Consulte también: Resumen del editor de comparación (Página 375)

Estructura de la comparación detallada

La figura siguiente muestra un ejemplo de la vista de navegación en una comparación detallada online/offline para el lenguaje de programación GRAPH:



① Barra de herramientas de la comparación detallada para GRAPH

② Bloque de partida

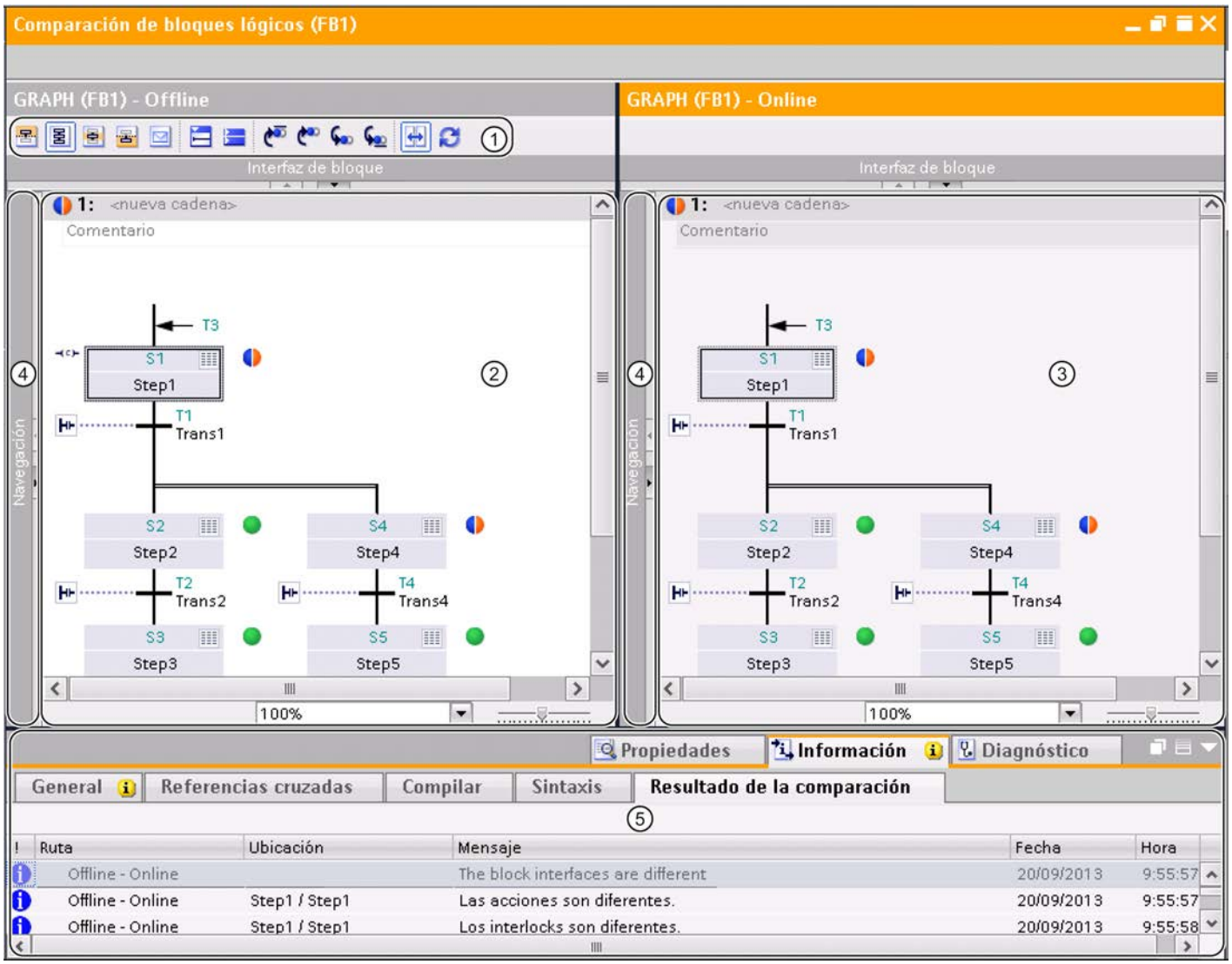
③ Bloque comparado

④ Barra de herramientas de navegación

⑤ Divisor de ventana

⑥ Resultado de la comparación en la ventana de inspección

La figura siguiente muestra un ejemplo de la vista de cadena en una comparación detallada online/offline para el lenguaje de programación GRAPH:



① Barra de herramientas de la comparación detallada para GRAPH

② Bloque de partida

③ Bloque comparado

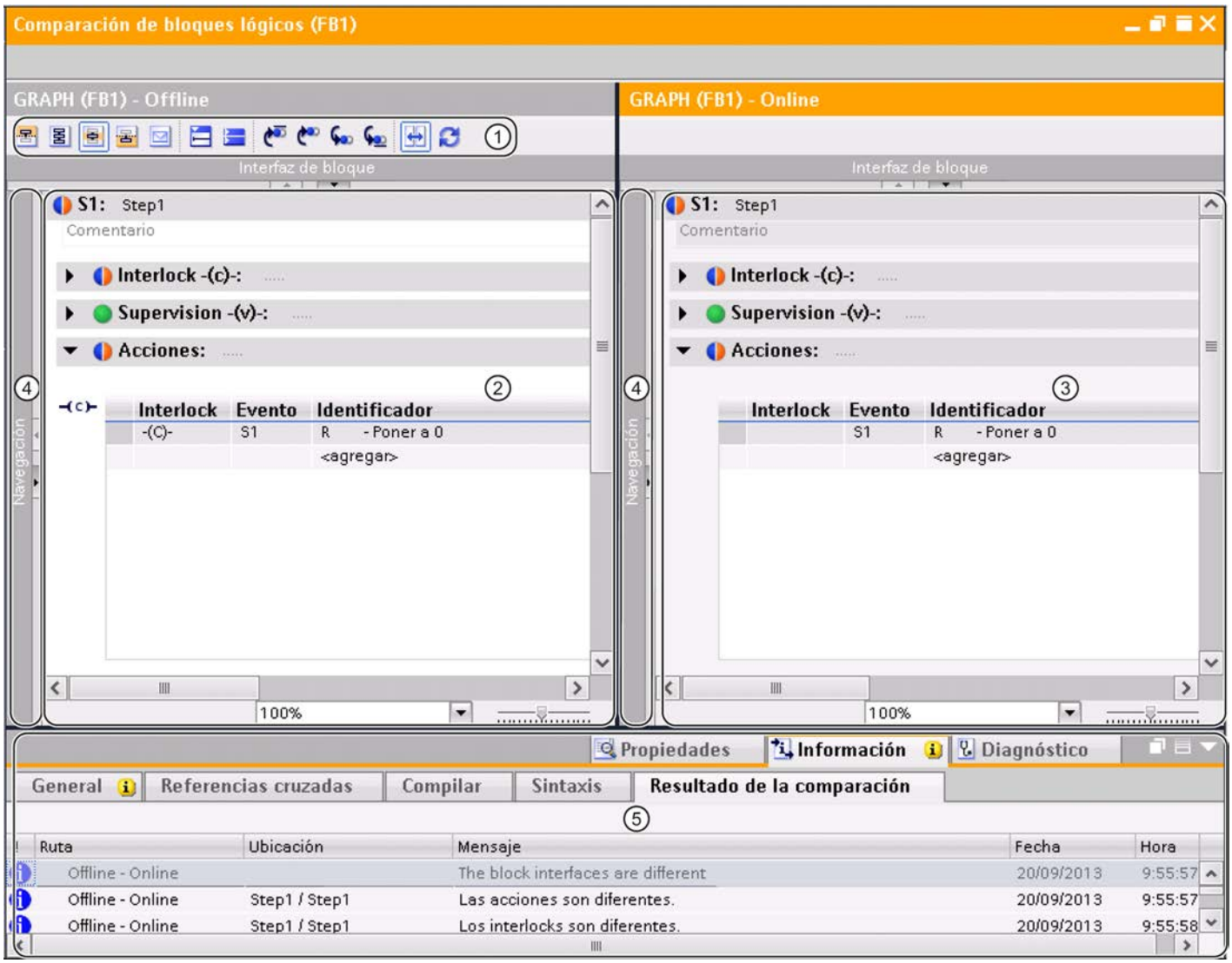
④ Divisor de ventana

⑤ Resultado de la comparación en la ventana de inspección

Nota

Si existen diferencias estructurales entre los bloques, en la vista de cadena solo se muestran los resultados de la comparación hasta la primera diferencia estructural.

La figura siguiente muestra un ejemplo de la vista de etapa individual en una comparación detallada online/offline para el lenguaje de programación GRAPH:



① Barra de herramientas de la comparación detallada para GRAPH

② Bloque de partida

③ Bloque comparado

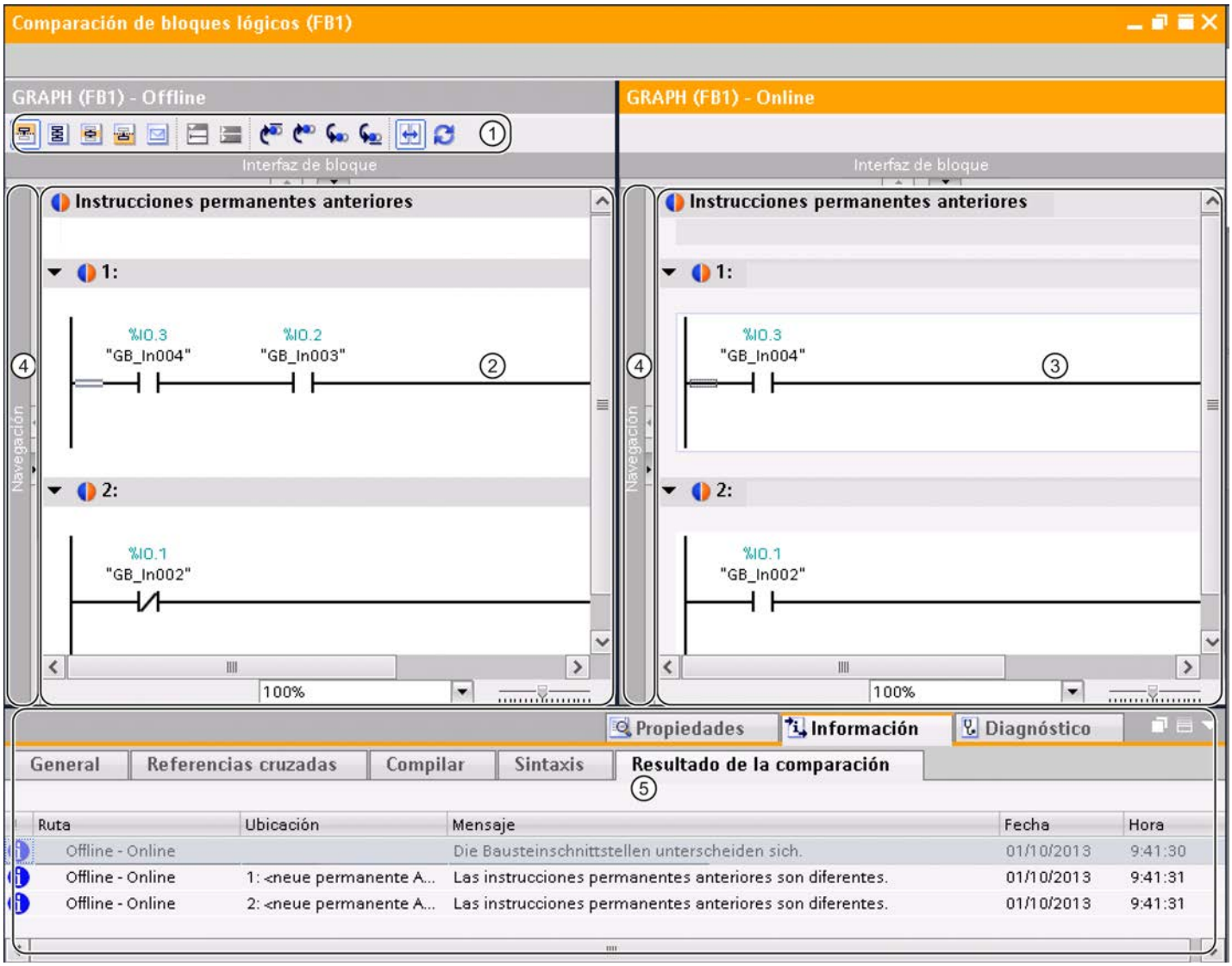
④ Divisor de ventana

⑤ Resultado de la comparación en la ventana de inspección

Nota

El resultado de la comparación se refiere al segmento completo. No se marcan diferencias dentro de los segmentos.

La figura siguiente muestra un ejemplo de la vista de instrucciones permanentes en una comparación detallada online/offline para el lenguaje de programación GRAPH:



① Barra de herramientas de la comparación detallada para GRAPH

② Bloque de partida

③ Bloque comparado

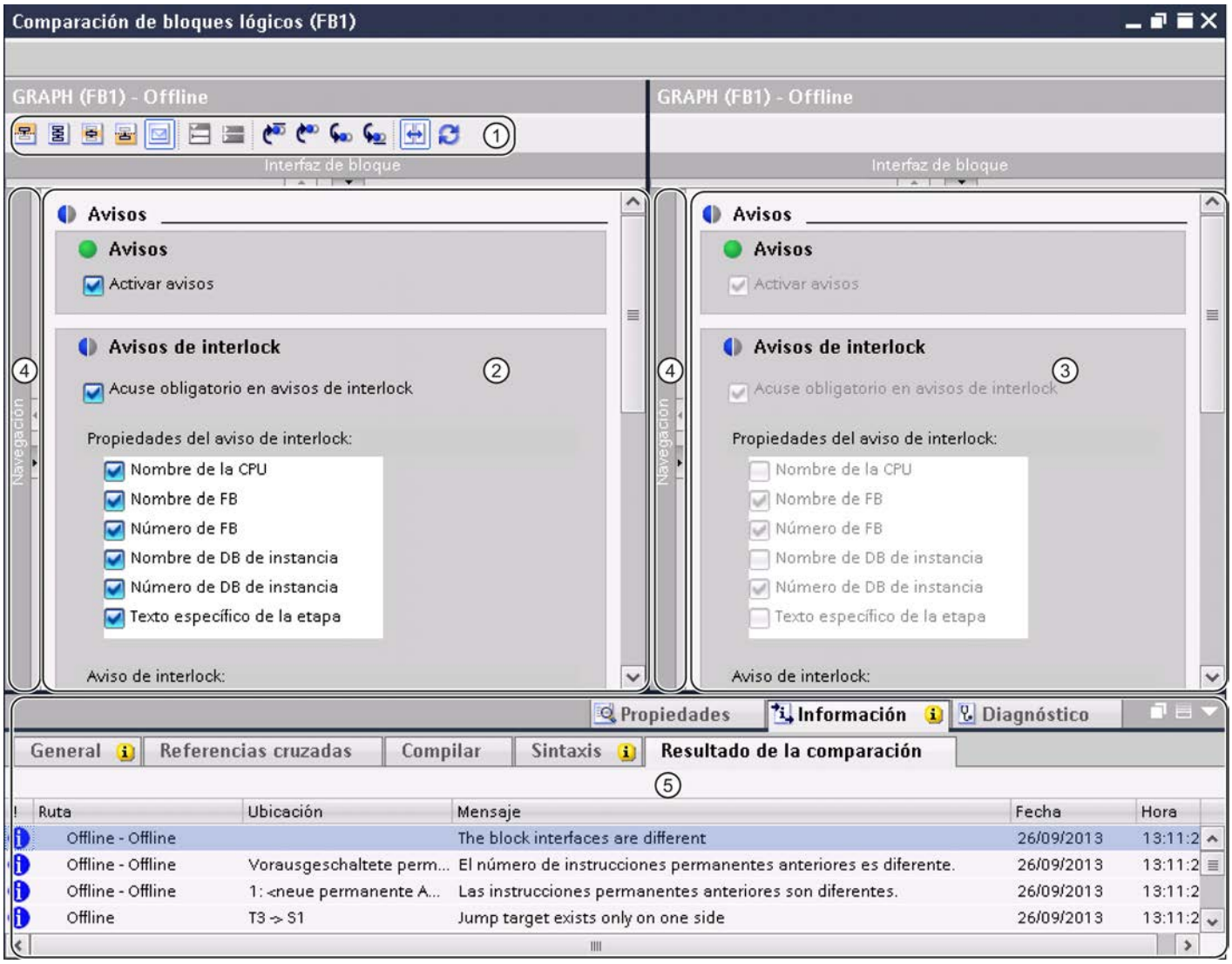
④ Divisor de ventana

⑤ Resultado de la comparación en la ventana de inspección

Nota

El resultado de la comparación se refiere al segmento completo. No se marcan diferencias dentro de los segmentos.

La figura siguiente muestra un ejemplo de la vista de avisos en una comparación detallada offline/online para el lenguaje de programación GRAPH:



① Barra de herramientas de la comparación detallada para GRAPH

② Bloque de partida

③ Bloque comparado

④ Divisor de ventana

⑤ Resultado de la comparación en la ventana de inspección

Barras de herramientas

La barra de herramientas de la comparación detallada permite acceder a las siguientes funciones:

- Funciones generales
  - Cambiar a las instrucciones permanentes anteriores
  - Cambiar a la vista de cadena
  - Cambiar a la vista de etapa individual
  - Cambiar a las instrucciones permanentes posteriores
  - Cambiar a la vista de avisos
  - Abrir todos los segmentos
  - Cerrar todos los segmentos
- Funciones específicas de comparación
  - Posicionar en la primera diferencia
  - Posicionar en la anterior diferencia
  - Posicionar en la siguiente diferencia
  - Posicionar en la última diferencia
  - Sincronizar desplazamiento de los editores
  - Actualizar los resultados de la comparación

La navegación dispone de una barra de herramientas propia con las siguientes funciones:

- Aumentar o reducir el tamaño de los elementos dentro de la navegación
- Sincronizar la navegación

Bloque de partida

El bloque de partida se muestra en la ventana izquierda. En una comparación online/offline, el bloque de partida es la versión offline del bloque.

Bloque comparado

El bloque comparado se muestra en la ventana derecha. En una comparación online/offline, el bloque comparado es la versión online del bloque.

Divisor de ventana

Puede hacer clic en el divisor de ventana para cambiar rápidamente entre la navegación y la vista actual.

Resultado de la comparación en la ventana de inspección

En la ficha "Información > Resultado de la comparación" de la ventana de inspección se muestra una tabla que recoge las diferencias. Haga doble clic en una fila para acceder a la diferencia correspondiente en el bloque.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Navegar en la comparación detallada (Página 3006)

Modificar bloques durante la comparación detallada (Página 3007)

Actualizar resultados de la comparación (Página 3008)

Navegar en la comparación detallada

Requisitos

Se ha realizado una comparación detallada.

Navegar hasta las diferencias

Para navegar hasta una diferencia entre ambos bloques, proceda del siguiente modo:

1. En la ventana de inspección, elija "Información > Resultado de la comparación" para abrir la lista de resultados de la comparación detallada.
2. Haga doble clic en una diferencia.
   La diferencia en cuestión se selecciona en ambos editores.

O bien:

1. Haga clic en uno de los siguientes botones de navegación de la barra de herramientas:
   - Posicionar en la primera diferencia
     Navega hasta la primera diferencia del bloque y la muestra en ambos editores.
   - Posicionar en la anterior diferencia
     A partir de la posición actual, navega hasta la diferencia anterior y la muestra en ambos editores.
   - Posicionar en la siguiente diferencia
     A partir de la posición actual, navega hasta la diferencia siguiente y la muestra en ambos editores.
   - Posicionar en la última diferencia
     Navega hasta la última diferencia del bloque y la muestra en ambos editores.

Activar y desactivar la sincronización del desplazamiento vertical entre los editores

Para que los segmentos que se correspondan permanezcan visibles paralelamente al desplazarse en sentido vertical, es posible sincronizar el desplazamiento de ambos editores. Esta función se puede activar o desactivar a discreción. Proceda del siguiente modo:

1. Para desactivar el desplazamiento isócrono, haga clic en el botón "Sincronizar desplazamiento de los editores" de la barra de herramientas.
2. Para volver a activar el desplazamiento isócrono, haga clic nuevamente en el botón "Sincronizar desplazamiento de los editores" de la barra de herramientas.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Representación del resultado de la comparación para KOP/FUP (Página 2989)

Modificar bloques durante la comparación detallada (Página 3007)

Actualizar resultados de la comparación (Página 3008)

Modificar bloques durante la comparación detallada

Modificar bloques offline

Los bloques offline que sirven de partida de la comparación pueden modificarse en cualquier momento. En cambio el bloque comparado está protegido contra escritura y no se puede modificar.

Modificar bloques online

Los bloques online no se pueden modificar.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Representación del resultado de la comparación para KOP/FUP (Página 2989)

Navegar en la comparación detallada (Página 3006)

Actualizar resultados de la comparación (Página 3008)

Actualizar resultados de la comparación

Tan pronto como modifique un objeto, los resultados de la comparación dejarán de ser válidos y deberán actualizarse.

Requisitos

Se ha realizado una comparación detallada.

Procedimiento

Para actualizar los resultados de la comparación, proceda del siguiente modo:

1. Haga clic en el botón "Actualizar resultados de la comparación" de la barra de herramientas.

Consulte también

Principios básicos para comparar programas de PLC (Página 2982)

Realizar una comparación online/offline (Página 374)

Realizar una comparación offline/offline (Página 374)

Utilizar el editor de comparación (Página 375)

Iniciar la comparación detallada (Página 2988)

Representación del resultado de la comparación para KOP/FUP (Página 2989)

Navegar en la comparación detallada (Página 3006)

Modificar bloques durante la comparación detallada (Página 3007)

9.1.5.3 Comparar variables PLC

Existen las siguientes posibilidades de comparar variables PLC:

- Comparación offline/offline automática en el editor de comparación
  Se comparan offline las tablas de variables PLC de los dispositivos seleccionados.
- Comparación offline/offline manual en el editor de comparación
  Se comparan offline las tablas de variables PLC seleccionadas de los dispositivos.
- Comparación detallada
  Mediante la comparación detallada se determinan las diferencias dentro de las tablas de variables PLC.

Comparación offline/offline automática en el editor de comparación

Para realizar una comparación offline/offline automática de tablas de variables PLC, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   El dispositivo comparado puede ser del mismo proyecto, de un proyecto de referencia o de la librería.
4. Abra la carpeta "Variables PLC".
   Los símbolos del área de estado y acción indican el estado de las tablas de variables PLC. Dependiendo de su estado se pueden definir determinadas acciones.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Comparación offline/offline manual en el editor de comparación

Para realizar una comparación offline/offline manual de tablas de variables PLC, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   El dispositivo comparado puede ser del mismo proyecto, de un proyecto de referencia o de la librería.
4. En el área de estado y acción haga clic en el botón para conmutar entre la comparación automática y manual.
5. Seleccione las tablas de variables PLC que desee comparar.
   Se muestra la comparación de propiedades. Los símbolos indican el estado.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Realización de una comparación detallada

Para iniciar una comparación detallada de una tabla de variables PLC, proceda del siguiente modo:

1. Ejecute una comparación offline/offline automática o manual.
2. Para una comparación offline/offline automática, seleccione en el editor de comparación la tabla de variables PLC con la que desee realizar una comparación detallada. En una comparación offline/offline manual, tenga en cuenta que deben seleccionarse dos tablas de variables PLC para poder realizar la comparación.
3. Haga clic en el botón "Iniciar comparación detallada" de la barra de herramientas.
   Se abrirá un editor de comparación propio. Todas las variables PLC disponibles de las tablas de variables PLC seleccionadas se muestran en función de los ajustes del editor de comparación. Las constantes de usuario y de sistema no se visualizan. Los símbolos indican el estado de las variables PLC. Dependiendo del estado de las variables PLC se pueden definir determinadas acciones.

Consulte también

Introducción a la comparación de programas de PLC (Página 2982)

Utilizar el editor de comparación (Página 375)

Comparar bloques (Página 2986)

Comparar tipos de datos PLC (Página 3010)

9.1.5.4 Comparar tipos de datos PLC

Existen las siguientes posibilidades de comparar tipos de datos PLC:

- Comparación online/offline (solo S7-1200/1500)
  Los tipos de datos PLC del proyecto se comparan con los tipos de datos PLC del dispositivo seleccionado.
- Comparación offline/offline automática en el editor de comparación
  Se comparan offline los tipos de datos PLC de los dispositivos seleccionados.
- Comparación offline/offline manual en el editor de comparación
  Se comparan offline los tipos de datos PLC seleccionadas de los dispositivos.
- Comparación detallada
  Mediante la comparación detallada se determinan las diferencias entre los tipos de datos PLC.

Comparación online/offline de tipos de datos PLC

Para realizar una comparación online/offline, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación online/offline.
2. Elija el comando "Comparar > offline/online" del menú contextual.
   Si hasta ahora no se ha establecido ninguna conexión online para ese dispositivo, aparecerá el cuadro de diálogo "Establecer conexión online". En ese caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Conectar".
   La conexión online se establece y se abre el editor de comparación.
3. Abra la carpeta "Tipos de datos PLC".
   Los símbolos del área de estado y acción indican el estado. Si se selecciona un objeto, las propiedades del tipo de datos PLC y del tipo de datos PLC correspondiente al dispositivo asignado se muestran en la comparación de propiedades.

Comparación offline/offline automática en el editor de comparación

Para realizar una comparación offline/offline automática de tablas de variables PLC, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   El dispositivo comparado puede ser del mismo proyecto, de un proyecto de referencia o de la librería.
4. Abra la carpeta "Tipos de datos PLC".
   Los símbolos del área de estado y acción indican el estado de las tablas de variables PLC. Dependiendo de su estado se pueden definir determinadas acciones.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Comparación offline/offline manual en el editor de comparación

Para realizar una comparación offline/offline manual de tipos de datos PLC, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto un dispositivo que permita una comparación offline/offline.
2. Elija el comando "Comparar > offline/offline" del menú contextual.
   Se abre el editor de comparación y el dispositivo seleccionado se muestra en el área izquierda.
3. Arrastre otro dispositivo mediante Drag & Drop a la superficie de arrastre del área derecha.
   El dispositivo comparado puede ser del mismo proyecto, de un proyecto de referencia o de la librería.

4. En el área de estado y acción haga clic en el botón para conmutar entre la comparación automática y manual.
5. Seleccione los tipos de datos PLC que desee comparar.
   Se muestra la comparación de propiedades. Los símbolos indican el estado.

En todo momento se pueden arrastrar otros dispositivos a las áreas de arrastre para realizar nuevas comparaciones.

Realización de una comparación detallada

Para iniciar una comparación detallada de un tipo de datos PLC, proceda del siguiente modo:

1. Efectúe una comparación offline/offline. Con las CPU de la serie S7-1200/1500 también es posible realizar una comparación online/offline.
2. Para una comparación offline/offline automática, seleccione en el editor de comparación el tipo de datos PLC con el que desee realizar una comparación detallada. En una comparación offline/offline manual, tenga en cuenta que deben seleccionarse dos tipos de datos PLC para poder realizar la comparación.
3. Haga clic en el botón "Iniciar comparación detallada" de la barra de herramientas.
   Los dos tipos de datos PLC se abren uno junto al otro para poder identificar rápidamente las diferencias.

Consulte también

Introducción a la comparación de programas de PLC (Página 2982)

Utilizar el editor de comparación (Página 375)

Comparar bloques (Página 2986)

Comparar variables PLC (Página 3009)

9.1.6 Compilar y cargar bloques

9.1.6.1 Compilar bloques

Principios básicos de la compilación de bloques

Introducción

Para que la CPU pueda procesar el programa de usuario, este se debe compilar primero. Cada vez que se modifica el programa es preciso compilarlo de nuevo.

Durante la compilación se llevan a cabo las siguientes operaciones:

- Se buscan errores de sintaxis en el programa de usuario.
- Las instrucciones no necesarias se eliminan del programa de usuario.

- Se comprueban todas las llamadas de bloque en los bloques compilados. En caso de modificaciones en las interfaces de los bloques llamados, se muestran los errores en la ficha "Compilar" de la ventana de información. Estos errores se deben corregir en primer lugar.
- En el programa de usuario, los números de bloques deben ser unívocos. Si hay varios bloques con el mismo número, durante la compilación se cambia automáticamente el número de los bloques que presentan conflictos. En los siguientes casos no se cambia el número de un bloque:
  - El bloque se ha seleccionado para la compilación individualmente o bien como parte de una selección múltiple.
  - El bloque tiene protección de know-how.
  - En las propiedades del bloque se ha ajustado una asignación de números "manual".

  Los conflictos de números que no pueden solucionarse mediante el cambio automático de número deben corregirse manualmente. Tenga en cuenta al respecto los avisos de la ventana de inspección. Los bloques con protección de know-how solo pueden corregirse manualmente si se conoce la contraseña, pues seguidamente es necesario volver a compilar los bloques.
- A continuación, el programa de usuario se compila a un código legible por la CPU.

Vías de compilación

La compilación puede iniciarse en las siguientes ventanas o editores:

- Compilar bloques en el árbol del proyecto
  Permite compilar bloques individuales, o bien compilar simultáneamente varios o todos los bloques de la carpeta "Bloques de programa".
- Compilar bloques en el editor de programación
  Permite compilar un solo bloque abierto.
- Compilar bloques en la estructura de llamadas o de dependencias
  Permite compilar bloques individuales.
  Consulte también: Estructura de llamadas (Página 3063), estructura de dependencias (Página 3070)

Opciones de compilación

Si se compilan bloques desde el árbol del proyecto, se ofrecen las siguientes opciones adicionales:

- Software (solo cambios)
  Se compilan todos los cambios de programa de los bloques seleccionados. Si se ha seleccionado una carpeta de bloques, todos los cambios de programa se compilan en los bloques contenidos en ella.
- Software (compilar todos los bloques)
  Todos los bloques se compilan por completo. Se recomienda elegir esta opción en la primera compilación y al realizar cambios mayores.
- Software (inicializar reserva de memoria)
  Todas las variables declaradas en el área de reserva de las interfaces de los bloques seleccionados se mueven al área estándar de las interfaces. La reserva de memoria se libera para otras ampliaciones de interfaz.

Nota

Esta opción solo está disponible en CPU de las series S7-1500 y S7-1200 V4.

Comprobación de coherencia

Si se modifican interfaces de bloques llamados o los tipos de datos PLC utilizados, se pueden presentar incoherencias entre los bloques invocantes y llamados o entre los tipos de datos PLC y los bloques de datos globales que utilizan estos tipos de datos PLC.

Para evitar tales incoherencias en el programa de usuario, el sistema realiza una comprobación de coherencia automática antes de cada proceso de compilación. Se comparan los sellos de tiempo y, dependiendo del resultado de la comparación, el proceso de compilación se realiza o se cancela:

- La compilación correcta del bloque invocante solo es posible si los sellos de tiempo de las interfaces de los bloques llamados son más antiguos que los del bloque invocante.
- La compilación correcta de un bloque de datos global basado en un tipo de datos PLC solo es posible si el sello de tiempo del bloque de datos global es más reciente que el del tipo de datos PLC utilizado.
- La compilación correcta de un bloque de datos instancia solo es posible si los sellos de tiempo de las interfaces del bloque de datos instancia y del bloque de función asignado son idénticos.

Si se cancela el proceso de compilación, aparece un mensaje en la ventana de inspección. Actualice primero las llamadas de bloque en los bloques respectivos, los tipos de datos PLC en los bloques de datos globales e inicie nuevamente el proceso de compilación. La comprobación de coherencia también determina los bloques con protección de know-how para los que no se realiza el proceso de compilación. También encontrará los avisos correspondientes en la ventana de inspección.

Si se inicia inmediatamente un proceso de carga en vez de uno de compilación, los bloques seleccionados se compilan automáticamente y los bloques de datos globales se actualizan implícitamente. Existen las siguientes diferencias en las familias de CPU:

- S7-1200/1500: En un proceso de carga también se cargan todos los bloques implicados, de modo que no pueden producirse incoherencias.
- S7-300/400: Solo se carga el bloque seleccionado.

Consulte también

Compilar bloques en el árbol del proyecto (Página 3015)

Compilar bloques en el editor de programación (Página 3016)

Solucionar errores de compilación (Página 3017)

Sellos de tiempo de bloques (Página 2611)

Actualizar llamadas de bloque en KOP (Página 2688)

Actualizar llamadas de bloque en FUP (Página 2732)

Actualizar las llamadas de bloque en AWL (Página 2779)

Compilar datos de proyecto (Página 363)

Compilar bloques en el árbol del proyecto

En el árbol del proyecto es posible compilar uno, varios o todos los bloques.

En las CPU de las series S7-1500 y S7-1200 V4, la compilación permite reinicializar además el formato de memoria de los bloques con reserva de memoria. Encontrará la descripción de la reserva de memoria en el capítulo "Cargar bloques (S7-1200/1500) > Cargar cambios de bloque sin reinicializar".

Requisitos

El árbol del proyecto está abierto.

Compilar uno o varios bloques en el árbol del proyecto

Para compilar uno o varios bloques en el árbol del proyecto, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del árbol del proyecto.
2. Seleccione los bloques que desea compilar.
3. Elija el comando "Compilar > Software (solo cambios)" del menú contextual.

Compilar todos los bloques en el árbol del proyecto

Para compilar todos los bloques de la carpeta "Bloques de programa" en el árbol del proyecto, proceda del siguiente modo:

1. Seleccione la carpeta "Bloques de programa" del árbol del proyecto.
2. Para la compilación se puede elegir una de las opciones siguientes:
   – Si desea compilar únicamente los cambios efectuados desde la última compilación, elija el comando "Compilar > Software (solo cambios)" del menú contextual.
   – Si desea compilar los bloques por completo, elija el comando "Compilar > Software (compilar bloques por completo)" del menú contextual.

Reinicializar el formato de memoria (S7-1500/S7-1200 V4)

Para reinicializar el formato de la memoria de bloques, proceda del siguiente modo:

1. Seleccione toda la carpeta "Bloques de programa" o bloques individuales dentro de ella.
2. En el menú contextual, elija el comando "Compilar > Software (inicializar reserva de memoria)".

Resultado

Si la comprobación de coherencia ha finalizado correctamente, se genera el código para los bloques. Se borran los bloques de datos instancia generados por el sistema y que ya no son necesarios.

El aviso que aparece en la ventana de inspección en "Información > Compilar" indica si la compilación se ha realizado correctamente.

Consulte también

Principios básicos de la compilación de bloques (Página 3012)

Compilar bloques en el editor de programación (Página 3016)

Solucionar errores de compilación (Página 3017)

Buscar errores de sintaxis en el programa (Página 2930)

Compilar bloques en el editor de programación

Nota

Recuerde que el bloque también se vuelve a compilar aunque no se hayan realizado cambios, con lo que cambiarán los sellos de tiempo del bloque.

Requisitos

El bloque que se desea compilar está abierto.

Procedimiento

Para compilar un bloque en el editor de programación, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la ventana de instrucciones del editor de programación.
2. Elija el comando "Compilar" del menú contextual.

Resultado

Se genera el código del bloque. Se borran los bloques de datos instancia generados por el sistema y que ya no son necesarios.

El aviso que aparece en la ventana de inspección en "Información > Compilar" indica si la compilación se ha realizado correctamente.

Consulte también

Principios básicos de la compilación de bloques (Página 3012)

Compilar bloques en el árbol del proyecto (Página 3015)

Solucionar errores de compilación (Página 3017)

Solucionar errores de compilación

En la ventana de inspección se indica bajo "Información > Compilar" si la compilación se ha realizado correctamente, o bien si se han detectado errores en el programa. Si ocurren errores, es preciso solucionarlos y reiniciar luego la compilación.

Procedimiento

Para solucionar errores de compilación, proceda del siguiente modo:

1. En la ventana de inspección, elija "Información > Compilar" para abrir la lista de errores.
2. Si aparece, haga clic en el signo de interrogación azul situado junto al texto de error para obtener indicaciones sobre cómo solucionar el error.
3. Haga doble clic en el error que desea corregir.
   Se selecciona el error en cuestión.
4. Corrija el error.
5. Reinicie la compilación.

Consulte también

Principios básicos de la compilación de bloques (Página 3012)

Compilar bloques en el editor de programación (Página 3016)

Compilar bloques en el árbol del proyecto (Página 3015)

9.1.6.2 Cargar bloques

Introducción a la carga de bloques

Cargar bloques en un dispositivo

Para que la CPU pueda procesar el programa de usuario, este se debe compilar primero y cargar luego en el dispositivo. Las opciones de carga disponibles son:

- Cargar bloques en el editor de programación
  Es posible cargar solo un bloque abierto en el editor de programación.
- Cargar bloques en el árbol del proyecto
  Mediante el árbol del proyecto se pueden cargar varios o todos los bloques de la carpeta de bloques.

Durante la operación de carga, toda información necesaria para reconstruir el programa (incluida la información simbólica como, p. ej., nombres y comentarios para bloques lógicos y de datos) también se carga en el idioma actual del proyecto. Por esta razón, si modifica el idioma del proyecto, es necesario realizar una nueva operación de carga.

La información simbólica no se carga en la memoria de trabajo, sino en la memoria de carga.

Una vez se han cargado los datos de un dispositivo, la información simbólica también vuelve a estar disponible en el programa, de forma que mejora la lectura del código del programa. No obstante, tenga en cuenta que no es posible reemplazar el almacenamiento de datos en un proyecto offline cargando desde y hacia un dispositivo, puesto que este proceso no permite reproducir, p. ej., las tablas de observación o el multilingüismo de proyectos.

Los datos de bloques con protección de know-how solo pueden mostrarse por completo después de realizar la carga desde un dispositivo con una contraseña correcta.

Nota

Para evitar incoherencias entre los bloques que llaman y los bloques llamados, cuando se realizan modificaciones globales (p. ej. cambios de la interfaz de un bloque) siempre se compilan y cargan todos los bloques afectados.

Nota

S7-1200 versión 1.0

Si se carga un elemento del proyecto a la CPU p. ej. un bloque de programa, un bloque de datos o la configuración hardware, la CPU realizará un arranque en frío en el siguiente cambio al estado operativo RUN. Además de borrar las entradas, inicializar las salidas y borrar la memoria no remanente, en un arranque en frío también se borran las áreas de memoria remanentes. En cada transición posterior de STOP a RUN se efectúa un arranque en caliente, y en ese caso no se borra la memoria remanente.

Nota

S7-1500

En las CPU de la serie S7-1500, la memoria de carga se encuentra en la SIMATIC Memory Card. Por lo tanto, es imprescindible insertar una SIMATIC Memory Card para el funcionamiento de la CPU.

Cargar bloques de un dispositivo

Los bloques de un dispositivo se pueden cargar en el proyecto. Esto es necesario p. ej. para modificar bloques que solo están contenidos en ese dispositivo. En este proceso existe la posibilidad de cargar en el proyecto todos los bloques existentes (bloques de organización y de función, funciones y bloques de datos) y las variables PLC globales o bien cargar los bloques individualmente.

Cargar bloques en o de una Memory Card

Las Memory Cards son tarjetas de memoria enchufables que sirven, p. ej. en una CPU de la serie S7-1200, para sustituir la memoria de carga de un dispositivo. En las CPU de la serie S7-1500, contienen la memoria de carga. Para los dispositivos de las series S7-1200 y S7-1500 solo pueden utilizarse tarjetas SD de Siemens.

Para poder utilizar la Memory Card como memoria de carga, el programa de usuario o los distintos bloques deben cargarse en una Memory Card. Los bloques también se pueden volver a cargar en el proyecto desde una Memory Card.

Nota

S7-1200

Tenga en cuenta las indicaciones siguientes al cargar en o de una Memory Card:

- Si la CPU no contiene todavía ningún programa y se inserta en ella una Memory Card vacía, el programa de la PG/el PC se cargará en la Memory Card y no en la CPU.
- Si antes de arrancar la CPU se inserta en ella una Memory Card vacía, el programa contenido en la CPU se transferirá automáticamente a la Memory Card. A continuación, se borra el programa de la CPU.
- Si se inserta una Memory Card en una CPU que también contiene un programa antes de arrancarla, se ejecutará el programa de la Memory Card y no el de la CPU. El programa de la CPU se borra.

Cargar bloques de función GRAPH

Si se carga un bloque de función GRAPH junto con su bloque de datos de instancia, el procesamiento de la cadena secuencial empieza de nuevo por la etapa inicial. Esto puede provocar problemas en la sincronización de la cadena secuencial con el proceso. Dichos problemas se evitan desactivando la cadena secuencial antes de la carga.

Cargar cambios de bloque sin reinicializar

Con frecuencia surge la necesidad de modificar o ampliar a posteriori un programa de PLC que ya se ha puesto en servicio y que funciona correctamente en una instalación. El proceso debe afectar al funcionamiento lo menos posible.

Por ello, las CPU S7-1500 ofrecen la posibilidad de ampliar las interfaces de bloques de función o datos durante el funcionamiento y cargar los bloques modificados sin necesidad de pasar la CPU a STOP y sin que los valores de las variables ya cargadas se vean afectados. De ese modo es posible realizar cambios en el programa con facilidad. El proceso controlado no se ve afectado por esta carga (cargar sin reinicializar).

Efectos de un proceso de carga en los valores de variable de un bloque de datos

Cuando se cargan bloques de datos en un dispositivo en el estado operativo STOP, la siguiente transición del dispositivo a RUN afecta a los valores actuales de las variables del siguiente modo:

- Las variables que no están marcadas como remanentes reciben sus valores de arranque definidos.
- Las variables remanentes de las CPU S7-1200 solo conservan sus valores si se cumplen las siguientes condiciones:
  - El bloque de datos se ha cargado con el comando "Cargar en dispositivo > Software (solo cambios)".
  - No se ha modificado la estructura del bloque de datos.

  De lo contrario, también las variables remanentes recibirán sus valores de arranque definidos.
- Las variables remanentes de las CPU S7-1500 solo conservan sus valores si se cumplen las siguientes condiciones:
  - El bloque de datos se ha cargado con el comando "Cargar en dispositivo > Software (solo cambios)".
  - No se ha modificado la estructura del bloque de datos o se ha modificado dentro de la reserva de memoria.

  De lo contrario, también las variables remanentes recibirán sus valores de arranque definidos.

Cargar bloques con sincronización

En el contexto de los equipos de ingeniería, puede darse el caso de que varios usuarios con varios sistemas de ingeniería trabajen en un proyecto y accedan a una CPU S7-1500 en paralelo. Para garantizar la coherencia dentro de dicho proyecto es necesario sincronizar los datos modificados antes de cargarlos para que no se sobrescriban accidentalmente.

Así pues, si durante la carga se detectan diferencias entre la gestión de datos online y offline dentro del proyecto común y estas diferencias han sido causadas por otro sistema de ingeniería, en el proceso de carga se ofrece automáticamente una sincronización de los datos que van a cargarse.

En ese caso, en el cuadro de diálogo "Sincronización" se muestran los datos que deben sincronizarse con el estado actual (comparación online/offline) y las acciones posibles.

Existen las siguientes opciones de sincronización:

Caso de aplicación	Recomendación	Sincronización
Uno o varios bloques son más nuevos en la CPU (online) que en el sistema de ingeniería (offline).	Antes de la carga, estos bloques deberían cargarse primero de la CPU al sistema de ingeniería.	La sincronización automática es posible: Los bloques del sistema de ingeniería se actualizan antes de la carga.
Uno o varios bloques se crean de nuevo y solo existen en la CPU (online).	Antes de la carga, estos bloques deberían cargarse primero de la CPU al sistema de ingeniería.	La sincronización automática es posible: Los nuevos bloques se agregan al sistema de ingeniería antes de la carga.
Se han borrado uno o varios bloques en la CPU.	Estos bloques también deberían borrarse en el sistema de ingeniería antes de la carga.	La sincronización automática no es posible. Los bloques borrados en la CPU deberían borrarse manualmente en el proyecto offline del sistema de ingeniería.
Uno o varios bloques son diferentes en la CPU y en el sistema de ingeniería. Esto ocurre cuando otro usuario ha modificado y cargado en la CPU bloques en los que usted también ha realizado correcciones.	Estos bloques con modificaciones concurrentes deben adaptarse de forma manual. En este caso, decida qué cambios desea aplicar. Si deben conservarse los bloques de la CPU, debe aceptar estos bloques antes de la carga de la CPU a su sistema de ingeniería. Si deben aceptarse los bloques modificados por usted, puede continuar con la carga sin sincronización.	La sincronización automática no es posible: Los bloques afectados de la CPU o el sistema de ingeniería deben adaptarse de forma manual. Esto implica que se sobrescribirá una de las versiones de bloque existentes (online u offline).
Se detectan diferencias en la configuración hardware de la CPU (online) y del sistema de ingeniería (offline).	Las diferencias en la configuración hardware deben resolverse de forma manual. En este caso, decida qué configuración hardware desea aplicar. Si debe conservarse la configuración hardware existente en la CPU, debe aceptarla antes de cargarla en su sistema de ingeniería. Si debe aplicarse la configuración hardware que ha modificado, puede continuar con la carga sin sincronización.	La sincronización automática no es posible: La configuración hardware debe ajustarse de forma manual. Esto implica que se sobrescribirá una de las configuraciones hardware existentes (online u offline).

El comando "Forzar la carga en dispositivo" permite cargar bloques sin sincronización, si lo desea.

Cargar bloques en un dispositivo en estado operativo "RUN"

Principios básicos de la carga de bloques en estado operativo "RUN"

Al cargar bloques modificados en un dispositivo, no siempre es necesario conmutar el dispositivo al estado operativo "STOP". Es por ello que el sistema de ingeniería verifica si el

dispositivo debe ponerse en STOP para poder realizar el proceso de carga. El resultado de esta prueba se muestra en el cuadro de diálogo "Cargar vista preliminar". Si se requiere un cambio al estado operativo "STOP", el proceso de carga solo puede continuarse cuando se haya ajustado la opción correspondiente.

Nota

Si se realiza un proceso de carga en el estado operativo "RUN", los parámetros actuales no se sobrescriben. Los parámetros actuales no se modifican hasta el siguiente cambio de estado operativo de "STOP" a "RUN".

La tabla siguiente muestra las acciones que permiten realizar a continuación un proceso de carga en el estado operativo "RUN":

Acción	Carga posible en el estado operativo "RUN"	
	S7-1200	S7-1500
Cargar bloques individuales	Sí	Sí
Cargar todos los bloques	No	No
Agregar o borrar OBs	No	Sí
Agregar o borrar DBs, FCs o FBs	Sí	Sí
Modificar interfaces de bloque para FBs	No	Solo dentro de la reserva de memoria
Modificar interfaces de bloque para FCs	Sí	Sí
Modificar la estructura de un DB o de un DB instancia	No	Solo dentro de la reserva de memoria
Modificar la configuración hardware	No	No
Modificar los ajustes de remanencia de marcas y DBs	No	No
Modificar el código de programa de un FC, FB u OB	Sí	Sí
Modificar los atributos de OBs	Sí	Sí
Agregar comentarios	Sí	Sí
Agregar áreas de entrada, salida o marcas	Sí	Sí
Modificar nombres de variable	Sí	Sí
Número máximo de bloques no rebasado que puede cargarse en el estado operativo "RUN" en el dispositivo utilizado	Sí	Sí

Consulte también

Cargar bloques en un dispositivo desde el editor de programación (Página 3023)

Cargar bloques en un dispositivo desde el árbol del proyecto (Página 3024)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar bloques en un dispositivo desde el editor de programación

Requisitos

El bloque que se desea cargar está abierto.

Procedimiento

Para cargar un bloque en un dispositivo desde el editor de programación, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la ventana de instrucciones del editor de programación.
2. Elija el comando "Cargar en dispositivo" del menú contextual.
   - Si hasta ahora no se ha establecido ninguna conexión online, aparece el cuadro de diálogo "Carga avanzada". En este caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Cargar". Existe la posibilidad de visualizar todos los dispositivos compatibles activando la opción correspondiente y haciendo clic en el comando "Iniciar búsqueda". El diálogo "Carga avanzada" también se puede abrir explícitamente desde el menú "Online".
     Consulte también: Establecer y deshacer una conexión online
   - Si se ha definido la conexión online, se compilan los datos del proyecto en caso de ser necesario y se abre el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen las acciones necesarias para la carga.
3. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".

   Nota
   Acciones

   Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

   Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

   El botón "Cargar" se activará tan pronto como sea posible la carga.
4. Haga clic en "Cargar".
   Si es necesario realizar una sincronización, el sistema muestra automáticamente el cuadro de diálogo "Sincronización". En él se muestran avisos y se proponen las acciones necesarias para la sincronización. Existe la posibilidad de ejecutar estas acciones o forzar la carga sin sincronización haciendo clic en "Forzar la carga en dispositivo". En caso de haber ejecutado las acciones propuestas, el sistema le preguntará si desea continuar con la carga. Haga clic en "Continuar carga" para que se cargue el bloque. A continuación se abre el cuadro de diálogo "Cargar resultados", que muestra el estado y las acciones después del proceso de carga.
5. Si los módulos se deben rearrancar directamente después de la carga, active la casilla de verificación "Arrancar todos".
6. Cierre el diálogo "Cargar resultados" haciendo clic en "Finalizar".

Resultado

El código del bloque se carga en el dispositivo. Si los cambios afectan a otros bloques, estos también se compilan y se cargan en el dispositivo. Los bloques que existen solo online en el dispositivo se borran. Los bloques de datos de la CPU existentes se conservan. Cargando todos los bloques afectados y borrando los bloques que no se necesitan en el dispositivo se evitan incoherencias entre los bloques del programa de usuario.

Los mensajes que aparecen en la ventana de inspección bajo "Información > General" indican si la carga se ha realizado correctamente.

Consulte también

Cargar bloques en un dispositivo desde el árbol del proyecto (Página 3024)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar bloques en un dispositivo en estado operativo "RUN" (Página 3021)

Cargar bloques en un dispositivo desde el árbol del proyecto

Desde el árbol del proyecto es posible cargar uno, varios o todos los bloques en un dispositivo.

Cargar uno o más bloques en un dispositivo desde el árbol del proyecto

Para cargar uno o varios bloques en un dispositivo desde el árbol del proyecto, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del árbol del proyecto.
2. Seleccione los bloques que desea cargar.
3. Elija el comando "Cargar en dispositivo > Software (solo cambios)" del menú contextual.
   - Si hasta ahora no se ha establecido ninguna conexión online, aparece el cuadro de diálogo "Carga avanzada". En este caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Cargar". Existe la posibilidad de visualizar todos los dispositivos compatibles activando la opción correspondiente y haciendo clic en el comando "Iniciar búsqueda". El diálogo "Carga avanzada" también se puede abrir explícitamente desde el menú "Online".
     Consulte también: Establecer y deshacer una conexión online
   - Si se ha definido la conexión online, se compilan los datos del proyecto en caso de ser necesario y se abre el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen las acciones necesarias para la carga.

4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".

Nota

Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

El botón "Cargar" se activará tan pronto como sea posible la carga.

5. Haga clic en "Cargar".
Si es necesario realizar una sincronización, el sistema muestra automáticamente el cuadro de diálogo "Sincronización". En él se muestran avisos y se proponen las acciones necesarias para la sincronización. Existe la posibilidad de ejecutar estas acciones o forzar la carga sin sincronización haciendo clic en "Forzar la carga en dispositivo". En caso de haber ejecutado las acciones propuestas, el sistema le preguntará si desea continuar con la carga. Haga clic en "Continuar carga" para que se cargue el bloque. A continuación se abre el cuadro de diálogo "Cargar resultados", que muestra el estado y las acciones después del proceso de carga.
6. Si los módulos se deben rearrancar directamente después de la carga, active la casilla de verificación "Arrancar todos".
7. Cierre el diálogo "Cargar resultados" haciendo clic en "Finalizar".

Cargar todos los bloques en un dispositivo desde el árbol del proyecto

Para cargar en un dispositivo todos los bloques de la carpeta "Bloques de programa" desde el árbol del proyecto, proceda del siguiente modo:

1. Seleccione la carpeta "Bloques de programa" del árbol del proyecto.
2. Elija el comando "Cargar en dispositivo" del menú contextual.
3. Si desea cargar únicamente los cambios efectuados desde la última carga, seleccione la opción "Software (solo cambios)". Para cargar por completo todos los bloques e inicializar todos los valores a los valores de arranque, elija la opción "Cargar programa de PLC en dispositivo e inicializar".
   - Si hasta ahora no se ha establecido ninguna conexión online, aparece el cuadro de diálogo "Carga avanzada". En este caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Cargar". Existe la posibilidad de visualizar todos los dispositivos compatibles activando la opción correspondiente y haciendo clic en el comando "Iniciar búsqueda". El cuadro de diálogo "Carga avanzada" también se puede abrir explícitamente desde el menú "Online".
     Consulte también: Establecer y deshacer una conexión online
   - Si se ha definido la conexión online, se compilan los datos del proyecto en caso de ser necesario y se abre el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen las acciones necesarias para la carga.

4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".

Nota

Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

El botón "Cargar" se activará tan pronto como sea posible la carga.

5. Haga clic en "Cargar".
Si es necesario realizar una sincronización, el sistema muestra automáticamente el cuadro de diálogo "Sincronización". En él se muestran avisos y se proponen las acciones necesarias para la sincronización. Existe la posibilidad de ejecutar estas acciones o forzar la carga sin sincronización haciendo clic en "Forzar la carga en dispositivo". En caso de haber ejecutado las acciones propuestas, el sistema le preguntará si desea continuar con la carga. Haga clic en "Continuar carga" para que se cargue el bloque. A continuación se abre el cuadro de diálogo "Cargar resultados", que muestra el estado y las acciones después del proceso de carga.
6. Si los módulos se deben rearrancar directamente después de la carga, active la casilla de verificación "Arrancar todos".
7. Cierre el diálogo "Cargar resultados" haciendo clic en "Finalizar".

Resultado

El código de los bloques se carga en el dispositivo. Si los cambios afectan a otros bloques, estos también se compilan y se cargan en el dispositivo. Los bloques que existen solo online en el dispositivo se borran. Cargando todos los bloques afectados y borrando los bloques que no se necesitan en el dispositivo se evitan incoherencias entre los bloques del programa de usuario.

Los mensajes que aparecen en la ventana de inspección bajo "Información > General" indican si la carga se ha realizado correctamente.

Consulte también

Cargar bloques en un dispositivo desde el editor de programación (Página 3023)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar bloques en un dispositivo en estado operativo "RUN" (Página 3021)

Cargar bloques de un dispositivo

En el proyecto pueden cargarse todos los bloques de un dispositivo o solo algunos de forma individual.

Nota

Tenga en cuenta las indicaciones siguientes:

- Al cargar bloques individuales, asegúrese de que no se carguen también variables u otros bloques necesarios que pueda señalar de forma no intencionada. Durante la operación de carga, las referencias vuelven a asignarse en la medida de lo posible a variables y bloques partiendo de los nombres. Una vez finalice la operación de carga, compruebe si dichas asignaciones son correctas.
- S7-1500: al cargar bloques de función GRAPH de un dispositivo al proyecto no se cargan los textos de aviso específicos de etapa para los avisos de interlock y supervisión.

Requisitos

Las versiones online y offline de los bloques que se desean cargar son diferentes o los bloques solo existen online.

Cargar todos los bloques de un dispositivo

Para cargar todos los bloques de un dispositivo, proceda del siguiente modo:

1. Establezca una conexión online con el dispositivo del que desea cargar los bloques.
   Consulte también: Establecer y deshacer una conexión online
2. En el árbol del proyecto, seleccione la carpeta del dispositivo del cual desea cargar bloques.
3. Elija el comando "Cargar de dispositivo" del menú "Online".
   Se abre el cuadro de diálogo "Vista preliminar para cargar del dispositivo". En él se muestran avisos y se proponen las acciones necesarias para la carga.
4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activa tan pronto como sea posible la carga.
5. Haga clic en el botón "Cargar de dispositivo".
   La carga se lleva a cabo.

Cargar bloques individuales de un dispositivo

Para cargar bloques de un dispositivo individualmente, proceda del siguiente modo:

1. Establezca una conexión online con el dispositivo del que desea cargar los bloques.
   Consulte también: Establecer y deshacer una conexión online
2. Seleccione en el árbol del proyecto los bloques que desea cargar desde el dispositivo.
3. Elija el comando "Cargar de dispositivo" del menú "Online".
   Se abre el cuadro de diálogo "Vista preliminar para cargar del dispositivo". En él se muestran avisos y se proponen las acciones necesarias para la carga.

4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activa tan pronto como sea posible la carga.
5. Haga clic en el botón "Cargar de dispositivo".
   La carga se lleva a cabo.

Resultado

Los bloques se cargan al proyecto desde el dispositivo. Se pueden editar normalmente, compilarse nuevamente y cargarse de nuevo en el dispositivo.

Cargar bloques en una Memory Card

Requisitos

- La Memory Card está identificada como tarjeta de programa.
- Está abierta la carpeta "Bloques de programa" de la Memory Card.

Procedimiento

Para cargar bloques en una Memory Card, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del dispositivo en el árbol del proyecto.
2. Seleccione los bloques que desea cargar en la Memory Card.
3. En el árbol del proyecto, arrastre los bloques hasta la carpeta "Bloques de programa" de la Memory Card.
   Se abre el diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen las acciones necesarias para la carga.
4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".
5. El botón "Cargar" se activará tan pronto como sea posible la carga.
6. Haga clic en el botón "Cargar".
   Si es necesario realizar una sincronización, el sistema muestra automáticamente el cuadro de diálogo "Sincronización". En él se muestran avisos y se proponen las acciones necesarias para la sincronización. Existe la posibilidad de ejecutar estas acciones o forzar la carga sin sincronización haciendo clic en "Forzar la carga en dispositivo". En caso de haber ejecutado las acciones propuestas, el sistema le preguntará si desea continuar con la carga. Haga clic en "Continuar carga" para que se cargue el bloque. A continuación se abre el cuadro de diálogo "Cargar resultados", que muestra el estado y las acciones después del proceso de carga.
7. Haga clic en el botón "Finalizar".

Resultado

El bloque se carga en la Memory Card. Si los cambios afectan a otros bloques, estos también se cargarán en la Memory Card. Los bloques existentes solo en la Memory Card se borran. Cargando todos los bloques afectados y borrando los bloques que no se necesitan en la Memory Card se evitan incoherencias entre los bloques del programa de usuario.

Los mensajes que aparecen en la ventana de inspección bajo "Información > General" indican si la carga se ha realizado correctamente.

Consulte también

Cargar bloques de una Memory Card (Página 3029)

Acceder a Memory Cards (Página 425)

Cargar bloques de una Memory Card

Solo es posible cargar todos los bloques de una Memory Card en el proyecto.

Requisitos

Se muestra la Memory Card.

Consulte también: Acceder a Memory Cards (Página 425)

Procedimiento

Para cargar bloques desde una Memory Card en el proyecto, proceda del siguiente modo:

1. En el árbol del proyecto arrastre la carpeta de la Memory Card a la carpeta del dispositivo en el proyecto.
   Se abre el cuadro de diálogo "Vista preliminar para cargar del dispositivo". En él se muestran avisos y se proponen las acciones necesarias para la carga.
2. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activa tan pronto como sea posible la carga.
3. Haga clic en el botón "Cargar de dispositivo".

Consulte también

Cargar bloques en una Memory Card (Página 3028)

Desactivar la cadena secuencial antes de cargar un bloque de datos GRAPH

La cadena secuencial puede desactivarse globalmente antes de cargar un bloque de datos de instancia de o bien durante el proceso de carga.

Desactivar la cadena secuencial globalmente

Para desactivar globalmente la cadena secuencial para cada proceso de carga de un bloque de datos de instancia, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Active la casilla de verificación "Desactivar cadena antes de cargar el DB".
   En procesos de carga futuros, la cadena secuencial se desactivará antes de cargar el bloque de datos de instancia.

Desactivar la cadena secuencial durante el proceso de carga

Para desactivar la cadena secuencial durante el proceso de carga, proceda del siguiente modo:

1. Cargue el bloque de función GRAPH en el dispositivo.
   Durante el proceso de carga se muestra el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen acciones necesarias para la carga. Si hay que cargar el bloque de datos de instancia junto con el bloque de función GRAPH, se mostrará la acción "Desactivar cadena antes de cargar el DB" en el cuadro de diálogo "Cargar vista preliminar".
2. Active la casilla de verificación "Desactivar cadena antes de cargar el DB".

Cargar ampliaciones de bloque sin reinicializar

Principios básicos para cargar ampliaciones de bloque sin reinicializar

Descripción

Para modificar con posterioridad programas de PLC que ya se han puesto en servicio y que funcionan sin problemas en una instalación, las CPU de las series S7-1200 V4 y S7-1500 permiten ampliar la interfaz de bloques de función o datos durante el funcionamiento.

Los bloques modificados pueden cargarse sin necesidad de pasar la CPU a STOP y sin afectar a los valores de las variables ya cargadas.

De ese modo es posible realizar cambios en el programa con facilidad. El proceso controlado no se ve afectado por esta carga (cargar sin reinicializar).

Modo de funcionamiento

Cada bloque de función o datos dispone de forma predeterminada de una reserva de memoria que puede utilizarse para realizar cambios de interfaz con posterioridad. En un primer momento, la reserva no se utiliza. Si compila y carga el bloque y después se da cuenta de que desea cargar cambios de interfaz con posterioridad, entonces active la reserva de memoria. Todas las variables declaradas después se cargan en la reserva de memoria. Al realizar la carga a continuación, los valores de las variables ya cargadas no se ven afectados.

Para modificar el programa más tarde, mientras la instalación no esté funcionando, se dispone además de la posibilidad de restablecer el formato de memoria de uno o varios bloques en un único paso. Todas las variables de la reserva de memoria se mueven al área de memoria habitual. La reserva de memoria se libera de nuevo para otras ampliaciones de interfaz.

Requisitos

La función "Cargar sin reinicializar" está disponible si se cumplen los siguientes requisitos:

- El proyecto está disponible en el formato "TIA Portal V12" o en una versión superior.
- Se trabaja con una CPU de la serie S7-1200 V4 o S7-1500.
- Los bloques se crearon en KOP, FUP, AWL o SCL.
- Los bloques fueron creados por el usuario, es decir, no son bloques suministrados.
- Se trata de bloques de "acceso optimizado".

Pasos fundamentales

Para ampliar la interfaz de un bloque de función o un bloque de datos y cargar a continuación el bloque sin reinicializar, siga los siguientes pasos. En los siguientes capítulos encontrará una descripción detallada de los distintos pasos:

1. Todos los bloques tienen de forma predeterminada una reserva de memoria de 100 bytes en la memoria no remanente. En caso necesario es posible adaptar el tamaño de la reserva de memoria o definir una reserva de memoria en la memoria remanente.
2. Active la reserva de memoria.
3. Amplíe la interfaz del bloque.
4. Compile el bloque.
5. Vuelva a cargar el bloque en la CPU de la forma acostumbrada.

Nota

Los siguientes bloques de datos no ofrecen la posibilidad de cargar ampliaciones de bloques sin reinicializar:

- Bloques GRAPH
- Bloques de datos ARRAY

Consulte también

Información general acerca de la carga (Página 364)

Inicializar valores de ajuste en el programa online (Página 2964)

Ajustar la reserva de memoria para ampliaciones de bloque (Página 3032)

Activar la reserva de memoria (Página 3033)

Desactivar la reserva de memoria (Página 3034)

Ampliación de una interfaz de bloque o un bloque de datos (Página 3035)

Inicialización de la reserva de memoria (Página 3036)

Ajustar la reserva de memoria para ampliaciones de bloque

Introducción

Cada bloque de función o datos dispone de forma predeterminada de una reserva de 100 bytes. Dicha reserva no se encuentra en el área de memoria remanente del bloque. De forma predeterminada, no se prevé una reserva en la memoria remanente, puesto que la memoria remanente de la CPU está limitada.

El valor predeterminado de 100 bytes que es válido para todos los bloques recién creados del proyecto, se puede modificar. Al mismo tiempo, también es posible personalizar el tamaño de la reserva de memoria para cada bloque y definir una reserva en la memoria remanente para los diferentes bloques.

Ajustar el tamaño de la reserva de memoria para bloques nuevos

Para ajustar el tamaño de la reserva de memoria de todos los bloques recién creados en el proyecto, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. Seleccione el grupo "Programación PLC > General" en la navegación local.
3. En el campo de entrada "Reserva de memoria" del grupo "Reserva de memoria para la carga sin reinicializar", introduzca el número de bytes que desea prever para ampliar la interfaz del bloque más adelante.
   Los bloques de función recién creados y los bloques de datos globales obtienen ahora una reserva de memoria del tamaño indicado. Los bloques ya existentes no se ven afectados por la modificación.

Nota

El ajuste solo es válido para los bloques que soportan la función "Cargar sin reinicializar". El resto de bloques no se ve afectado por el ajuste.

Ajustar el tamaño de la reserva de memoria para un bloque existente

Para ajustar el tamaño de la reserva de memoria para un bloque existente, proceda del siguiente modo:

1. Seleccione el bloque en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el cuadro de diálogo "Propiedades".
3. Seleccione el grupo "Cargar sin reinicializar" en la navegación local.

4. Introduzca el número de bytes deseado en el campo de entrada "Reserva de memoria".
5. Para definir una reserva en la memoria remanente, active la casilla de verificación "Activar la carga sin reinicializar para variables remanentes" e introduzca el número de bytes deseado en el campo de entrada "Reserva en la memoria remanente".

Nota

Tenga en cuenta las indicaciones siguientes:

- Si ya ha activado la reserva de memoria para el bloque actual, el tamaño de la reserva de memoria no podrá cambiarse.
- Los elementos de datos se ordenan en la memoria en función de determinadas reglas que permiten un acceso lo más rápido posible durante la ejecución del programa. Por esta razón, al añadir un elemento de datos nuevo, en algunos casos la memoria disponible se reduce en una proporción mayor al tamaño real del elemento de datos en cuestión.
  Puede suceder, por ejemplo, que el área de memoria disponible se reduzca cuatro bytes al agregar un elemento del tipo "WORD" y un elemento del tipo "BYTE", puesto que todos los elementos de datos se ajustan a un límite de WORD.

Consulte también

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)

Activar la reserva de memoria

Introducción

Cada bloque de función o datos dispone de forma predeterminada de una reserva de memoria. Sin embargo, en un primer momento la reserva no se utiliza. Si compila y carga el bloque y después se da cuenta de que desea cargar cambios de interfaz con posterioridad, entonces active la reserva de memoria. Todas las variables declaradas después se cargan en la reserva de memoria.

Requisitos

- El bloque está compilado.
- El bloque dispone de una reserva de memoria.

Procedimiento

Para activar la reserva de memoria para un bloque de función o un bloque de datos global, proceda del siguiente modo:

1. Abra el bloque.
2. Seleccione el botón "Cargar sin reinicializar" en la barra de herramientas.

Resultado

- A partir de este momento, todas las variables recién declaradas se guardarán en la reserva de memoria, de forma que puedan cargarse en el proceso en curso sin repercusiones.
- Las variables existentes ya no podrán borrarse. Tampoco será posible modificar sus propiedades, a excepción de "Valor predeterminado", "Valor de arranque" y "Comentario".
- Las constantes locales que se utilizan como límites de ARRAY, ya no se pueden modificar porque una modificación del valor de una constante repercutiría en la estructura de la memoria del bloque.

Consulte también

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)

Desactivar la reserva de memoria

Introducción

Para modificar el formato de memoria de un bloque con posterioridad, mientras la instalación no esté funcionando, puede desactivarse la reserva de memoria. Entonces todas las variables de la reserva de memoria se mueven al área de memoria habitual. La reserva de memoria permanece inalterada y se libera de nuevo para otras ampliaciones de interfaz. De ese modo vuelve a tener el tamaño definido en las propiedades del bloque.

Nota

Tenga en cuenta que es necesario compilar y cargar el bloque de nuevo después de desactivar la reserva de memoria. Una vez se ha desactivado la reserva de memoria, no es posible "Cargar sin reinicializar".

Requisitos

La reserva de memoria del bloque está activada.

Procedimiento

Para desactivar la reserva de memoria para un bloque de función o un bloque de datos global, proceda del siguiente modo:

1. Abra el bloque.
2. Seleccione el botón "Cargar sin reinicializar" en la barra de herramientas.

Resultado

Todas las variables que hasta ahora se encontraban en la reserva de memoria se mueven del área de reserva al área habitual. Por esta razón es necesario compilar y cargar el bloque de nuevo. Al cargar, los valores de las variables se reinicializan en la CPU.

Consulte también

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)

Ampliación de una interfaz de bloque o un bloque de datos

Introducción

Tras activar la reserva de memoria de un bloque de función o un bloque de datos global, pueden agregarse nuevas variables.

Requisitos

La reserva de memoria está activada.

Procedimiento

Para declarar variables adicionales, proceda del siguiente modo:

1. Seleccione una sección de declaración, p. ej. "Input", "Output", "InOut" o "Static".
2. Declare una o varias variables de la forma habitual e introduzca sus propiedades en las columnas. En la columna "Remanencia" de un FB solo es posible escoger entre los ajustes "Remanente" y "No remanente". El ajuste "Ajustar en IDB" no está disponible en la reserva de memoria.

   Nota

   No es posible añadir variables dentro de una variable de un tipo de datos estructurado. Dentro de una estructura ya existente, p. ej., no se pueden declarar elementos nuevos.

   Pero sí se pueden crear nuevas estructuras dentro de la reserva de memoria.

3. Compile los bloques modificados, p. ej. seleccionándolos en el árbol del proyecto y eligiendo el comando "Compilar > Software (solo cambios)" del menú contextual.
   Al compilar, las nuevas variables declaradas se insertan en la reserva de memoria. Si la reserva de memoria no dispone del suficiente espacio, se cancelará el proceso de compilación y aparecerá un mensaje de error.
4. Cargue los bloques modificados, p. ej. seleccionándolos en el árbol del proyecto y eligiendo el comando "Cargar > Software (solo cambios)" del menú contextual.
   Si se cargan ampliaciones de bloque, solo se inicializarán las nuevas variables insertadas con los valores de arranque definidos. Las variables que ya existen online no se vuelven a inicializar.

Consulte también

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)

Inicialización de la reserva de memoria

Introducción

Para modificar el programa con posterioridad, mientras la instalación no esté funcionando existe la posibilidad de restablecer la reserva de memoria de uno o varios bloques en un único paso. En ese caso todas las variables de la reserva de memoria se mueven al área de memoria regular. La reserva de memoria se libera de nuevo para otras ampliaciones de interfaz.

Requisitos

- La reserva de memoria del bloque está activada.
- La reserva de memoria contiene variables.

Procedimiento

Para inicializar la reserva de memoria de uno o varios bloques, proceda del siguiente modo:

1. Seleccione toda la carpeta "Bloques de programa" o bloques individuales dentro de ella.
2. En el menú contextual, elija el comando "Compilar > Software (inicializar reserva de memoria)".

Resultado

Todas las variables de los bloques seleccionados que hasta ahora se encontraban en la reserva de memoria se mueven del área de reserva al área habitual. El bloque se compila de nuevo. Las variables se reinicializan al realizar la siguiente carga. La reserva de memoria del bloque ajustada permanece inalterada y continúa estando activa.

⚠ PELIGRO

Reinicialización en la carga siguiente

Si se reinicializa una reserva de memoria, en la siguiente carga se vuelven a inicializar todas las variables que antes se encontraban en la reserva de memoria. Eso afecta tanto a las variables remanentes como a las no remanentes.

La modificación de los valores de variable con la instalación en marcha puede provocar graves daños materiales y personales. Por ese motivo, tras inicializar la reserva de memoria es imprescindible comprobar detalladamente el programa en un entorno de prueba antes de ponerlo en marcha.

Consulte también

Principios básicos para cargar ampliaciones de bloque sin reinicializar (Página 3030)

9.1.6.3 Cargar bloques

Introducción a la carga de bloques

Cargar bloques en un dispositivo

Para que la CPU pueda procesar el programa de usuario, éste se debe compilar primero y cargar luego en el dispositivo. Las opciones de carga disponibles son:

- Cargar bloques en el editor de programación
  En el editor de programas se puede cargar un bloque individual abierto.
- Cargar bloques en el árbol del proyecto
  Desde el árbol de proyectos se pueden cargar varios o todos los bloques de la carpeta de bloques.

Nota

Si carga un bloque con protección de know-how en un dispositivo, no se cargará la información de restauración. Esto significa que un bloque con know-how protegido no se podrá volver a abrir si se carga del dispositivo.

Cargar bloques desde un dispositivo

Los bloques de un dispositivo se pueden cargar en el proyecto. Esto es necesario, p. ej., para modificar bloques que solo existen en este dispositivo. Se pueden cargar en el proyecto todos los bloques existentes (bloques de organización, bloques de función, funciones, bloques de datos) y las variables PLC globales, o bien bloques individuales.

Cargar bloques en una Memory Card o desde una Memory Card

Las Memory Cards son tarjetas de memoria enchufables que sirven, por ejemplo, para ampliar la memoria de una CPU y guardar los datos en el caso de que se produzca un corte de corriente. Dependiendo de la CPU empleada se pueden utilizar diferentes tipos de tarjetas de memoria. Para mayor información sobre las tarjetas de memoria compatibles con la CPU utilizada y las respectivas posibilidades de aplicación, consulte el capítulo "Funcionamiento de la CPU S7-300/400" en "Información adicional sobre configuraciones".

Cargar bloques de función GRAPH

Si se carga un bloque de función GRAPH junto con su bloque de datos de instancia, el procesamiento de la cadena secuencial empieza de nuevo por la etapa inicial. Esto puede provocar problemas en la sincronización de la cadena secuencial con el proceso. Dichos problemas se evitan desactivando la cadena secuencial antes de la carga.

Repercusiones de una operación de carga en los valores de variables de un bloque de datos

Si los bloques de datos se cargan en un dispositivo en estado operativo STOP, la siguiente transición del dispositivo a RUN afecta a los valores actuales de las variables del siguiente modo:

- Las variables de bloques de datos no remanentes reciben sus valores de arranque definidos.
- Las variables de bloques de datos remanentes únicamente conservan sus valores si se cumplen las siguientes condiciones:
  - Se ha cargado el bloque de datos mediante "Cargar en dispositivo > Software (solo cambios)".
  - No se ha modificado la estructura del bloque de datos.

  De lo contrario, las variables de los bloques de datos remanentes también reciben sus valores de arranque definidos.

Cargar bloques en un dispositivo en modo operativo "RUN"

Fundamentos de la carga de bloques en el modo operativo "RUN"

Para cargar bloques modificados en un dispositivo, no siempre es necesario que éste cambie al modo operativo "STOP". Por eso, en un proceso de carga, el sistema de ingeniería comprueba si el dispositivo debe pararse antes de la carga. El resultado de esta comprobación se muestra en el cuadro de diálogo "Cargar vista preliminar". Si es necesario cambiar al modo "STOP", no se podrá continuar con el proceso de carga hasta que haber ajustado la opción correspondiente.

Por regla general existen las restricciones siguientes para un proceso de carga en el modo operativo "RUN":

- Dependiendo del dispositivo empleado es posible restringir el número y el tipo de los bloques que se pueden cargar en el modo "RUN".
- No es posible cargar todos los bloques con el comando "Cargar programa de PLC en dispositivo e inicializar".
- Si ha modificado la configuración hardware del dispositivo que desea utilizar como destino del proceso de carga, no será posible realizar la carga en el modo "RUN".

Consulte también

Cargar bloques en un dispositivo desde el editor de programas (Página 3039)

Cargar bloques en el dispositivo desde el árbol de proyectos (Página 3040)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar bloques en un dispositivo desde el editor de programas

Requisitos

El bloque que se desea cargar está abierto.

Procedimiento

Para cargar un bloque en un dispositivo desde el editor de programación, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la ventana de instrucciones del editor de programación.
2. Elija el comando "Cargar en dispositivo" del menú contextual.
   - Si hasta ahora no se ha establecido ninguna conexión online, aparecerá el diálogo "Carga avanzada". En este caso, configure todos los parámetros necesarios para la conexión y haga clic en "Cargar". El diálogo "Carga avanzada" también se puede abrir explícitamente desde el menú "Online".
     Consulte también: Establecer y deshacer una conexión online
   - Una vez definida la conexión online, se compilan los datos del proyecto, si es necesario, y se abre el diálogo "Cargar vista preliminar". En él se muestran mensajes y se proponen las acciones necesarias para la carga.
3. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar" se activará tan pronto como sea posible la carga.

   Nota
   Acciones

   Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

   Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

   Nota

   Para evitar incoherencias entre los bloques que llaman y los bloques llamados, al realizar modificaciones globales, p. ej. modificaciones de la interfaz del bloque, pueden cargarse siempre todos los bloques afectados. Para ello, elija la acción "Carga coherente".

4. Haga clic en "Cargar".
   El bloque se carga y aparece el diálogo "Cargar resultados". Este diálogo muestra el estado y las acciones después del proceso de carga.
5. Cierre el diálogo "Cargar resultados" haciendo clic en "Finalizar".

Resultado

En el dispositivo se carga el código del bloque. Si ha seleccionado "Carga coherente" se compilan los bloques afectados por las modificaciones y también se cargan en el dispositivo.

Los mensajes que aparecen en la ventana de inspección bajo "Información > General" indican si la carga se ha realizado correctamente.

Consulte también

Cargar bloques en el dispositivo desde el árbol de proyectos (Página 3040)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar bloques en un dispositivo en modo operativo "RUN" (Página 3038)

Cargar bloques en el dispositivo desde el árbol de proyectos

Desde el árbol del proyecto es posible cargar uno, varios o todos los bloques en un dispositivo.

Cargar uno o más bloques en un dispositivo desde el árbol del proyecto

Para cargar uno o varios bloques en un dispositivo desde el árbol del proyecto, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del árbol del proyecto.
2. Seleccione los bloques que desea cargar.
3. Elija el comando "Cargar en dispositivo > Software (solo cambios)" del menú contextual.
   - Si hasta ahora no se ha establecido ninguna conexión online, aparece el cuadro de diálogo "Carga avanzada". En este caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Cargar". El diálogo "Carga avanzada" también se puede abrir explícitamente desde el menú "Online".
     Consulte también: Establecer y deshacer una conexión online
   - Si se ha definido la conexión online, se compilan los datos del proyecto en caso de ser necesario y se abre el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen las acciones necesarias para la carga.
4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar" se activará tan pronto como sea posible la carga.

Nota

Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

Nota

Para evitar incoherencias entre los bloques que llaman y los bloques llamados, al realizar modificaciones globales, p. ej. modificaciones de la interfaz del bloque, pueden cargarse siempre todos los bloques afectados. Para ello, elija la acción "Carga coherente".

5. Haga clic en "Cargar".
   Los bloques se cargan y aparece el cuadro de diálogo "Cargar resultados". Este diálogo muestra el estado y las acciones después del proceso de carga.
6. Cierre el diálogo "Cargar resultados" haciendo clic en "Finalizar".

Cargar todos los bloques en un dispositivo desde el árbol del proyecto

Para cargar en un dispositivo todos los bloques de la carpeta "Bloques de programa" desde el árbol del proyecto, proceda del siguiente modo:

1. Seleccione la carpeta "Bloques de programa" del árbol del proyecto.
2. Elija el comando "Cargar en dispositivo" del menú contextual.
3. Si desea cargar únicamente los cambios efectuados desde la última carga, seleccione la opción "Software (solo cambios)". Para cargar por completo todos los bloques e inicializar todos los valores a los valores de arranque, elija la opción "Cargar programa de PLC en dispositivo e inicializar".
   - Si hasta ahora no se ha establecido ninguna conexión online, aparece el cuadro de diálogo "Carga avanzada". En este caso, ajuste todos los parámetros necesarios para la conexión y haga clic en "Cargar". El diálogo "Carga avanzada" también se puede abrir explícitamente desde el menú "Online".
     Consulte también: Establecer y deshacer una conexión online
   - Si se ha definido la conexión online, se compilan los datos del proyecto en caso de ser necesario y se abre el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen las acciones necesarias para la carga.
4. Verifique los avisos y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar" se activará tan pronto como sea posible la carga.

   Nota

   Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

   Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

   Nota

   Para evitar incoherencias entre los bloques que llaman y los bloques llamados, al realizar modificaciones globales, p. ej. modificaciones de la interfaz del bloque, pueden cargarse siempre todos los bloques afectados. Para ello, elija la acción "Carga coherente".

5. Haga clic en "Cargar".
   Los bloques se cargan y aparece el cuadro de diálogo "Cargar resultados". Este diálogo muestra el estado y las acciones después del proceso de carga.
6. Cierre el diálogo "Cargar resultados" haciendo clic en "Finalizar".

Resultado

El código de los bloques se carga en el dispositivo. Si los cambios afectan a otros bloques, éstos también se compilan y se cargan en el dispositivo. Los bloques que existen sólo online en el dispositivo se borran. Cargando todos los bloques afectados y borrando los bloques que no se necesitan en el dispositivo se evitan incoherencias entre los bloques del programa de usuario.

Los mensajes que aparecen en la ventana de inspección bajo "Información > General" indican si la carga se ha realizado correctamente.

Consulte también

Cargar bloques en un dispositivo desde el editor de programas (Página 3039)

Cargar los datos de proyecto en un dispositivo (Página 367)

Cargar bloques en un dispositivo en modo operativo "RUN" (Página 3038)

Cargar bloques desde un dispositivo

Se pueden cargar desde el dispositivo al proyecto todos los bloques o bien bloques sueltos.

Nota

Los bloques SCL y los bloques de función GRAPH no se pueden cargar desde un dispositivo.

Requisitos

No coincide la versión online y offline de los bloques que se van a cargar o los bloques solo existen online.

Cargar todos los bloques desde un dispositivo

Para cargar todos los bloques desde un dispositivo, proceda del siguiente modo:

1. Establezca una conexión online con el dispositivo desde el que desea cargar bloques.
   Consulte también: Establecer y deshacer una conexión online
2. En el árbol del proyecto, seleccione la carpeta del dispositivo desde la que desee cargar los bloques.
3. Seleccione en el menú "Online" el comando "Cargar de dispositivo".
   Se abre el diálogo "Vista preliminar para cargar del dispositivo". En él se muestran mensajes y se proponen las acciones necesarias para la carga.
4. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activará tan pronto como sea posible la carga.
5. Haga clic en el botón "Cargar de dispositivo".
   La carga se lleva a cabo.

Cargar bloques sueltos desde un dispositivo

Para cargar bloques sueltos desde un dispositivo, proceda del siguiente modo:

1. Establezca una conexión online con el dispositivo desde el que desea cargar bloques.
   Consulte también: Establecer y deshacer una conexión online
2. En el árbol del proyecto, seleccione los bloques que desea cargar desde el dispositivo.
3. Seleccione en el menú "Online" el comando "Cargar de dispositivo".
   La carga se lleva a cabo.

Resultado

Los bloques se cargan en el proyecto desde el dispositivo. Los puede editar de la forma habitual, compilarlos y cargarlos después en el dispositivo.

Cargar bloques en una Memory Card

Requisitos

Está abierta la carpeta "Bloques de programa" de la Memory Card.

Procedimiento

Para cargar bloques en una Memory Card, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" del dispositivo en el árbol de proyectos.
2. Seleccione los bloques que desea cargar en la Memory Card.
3. En el árbol del proyecto, arrastre los bloques hasta la carpeta "Bloques de programa" de la Memory Card.
   Se abre el diálogo "Cargar vista preliminar". En él se muestran mensajes y se proponen las acciones necesarias para la carga.
4. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar" se activará tan pronto como sea posible la carga.

   Nota

   Si lleva a cabo las acciones propuestas con la instalación en marcha, pueden producirse graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

   Antes de iniciar la ejecución de las acciones, asegúrese de que no pueden presentarse estados peligrosos.

5. Haga clic en el botón "Cargar".
   La carga se lleva a cabo. A continuación aparece el diálogo "Cargar resultados". En él se puede comprobar si la carga se ha realizado correctamente y, en caso necesario, seleccionar acciones adicionales.
6. Haga clic en el botón "Finalizar".

Resultado

Los bloques seleccionados se cargan en la Memory Card. Los mensajes que aparecen en la ventana de inspección bajo "Información > General" indican si la carga se ha realizado correctamente.

Consulte también

Cargar bloques de una Memory Card (Página 3044)

Acceder a Memory Cards (Página 425)

Cargar bloques de una Memory Card

Solo es posible cargar en el proyecto todos los bloques de una Memory Card. Recuerde que al cargar de una Memory Card no se cargará información simbólica.

Requisitos

Se muestra la Memory Card.

Consulte también: Acceso a Memory Cards (Página 425)

Procedimiento

Para cargar bloques desde una Memory Card en el proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, arrastre la carpeta de la Memory Card a la carpeta del dispositivo en el proyecto.
   Se abre el diálogo "Vista preliminar para cargar del dispositivo". En él se muestran mensajes y se proponen las acciones necesarias para la carga.
2. Verifique los mensajes y, dado el caso, active las acciones en la columna "Acción".
   El botón "Cargar de dispositivo" se activará tan pronto como sea posible la carga.
3. Haga clic en el botón "Cargar de dispositivo".

Consulte también

Cargar bloques en una Memory Card (Página 3043)

Desactivar la cadena secuencial antes de cargar un bloque de datos GRAPH

La cadena secuencial puede desactivarse globalmente antes de cargar un bloque de datos de instancia de o bien durante el proceso de carga.

Desactivar la cadena secuencial globalmente

Para desactivar globalmente la cadena secuencial para cada proceso de carga de un bloque de datos de instancia, proceda del siguiente modo:

1. Elija el comando "Configuración" del menú "Opciones".
   Se abre la ventana "Configuración" en el área de trabajo.
2. En la navegación local, seleccione el grupo "Programación PLC > GRAPH".
3. Active la casilla de verificación "Desactivar cadena antes de cargar el DB".
   En procesos de carga futuros, la cadena secuencial se desactivará antes de cargar el bloque de datos de instancia.

Desactivar la cadena secuencial durante el proceso de carga

Para desactivar la cadena secuencial durante el proceso de carga, proceda del siguiente modo:

1. Cargue el bloque de función GRAPH en el dispositivo.
   Durante el proceso de carga se muestra el cuadro de diálogo "Cargar vista preliminar". En él se muestran avisos y se proponen acciones necesarias para la carga. Si hay que cargar el bloque de datos de instancia junto con el bloque de función GRAPH, se mostrará la acción "Desactivar cadena antes de cargar el DB" en el cuadro de diálogo "Cargar vista preliminar".
2. Active la casilla de verificación "Desactivar cadena antes de cargar el DB".

9.1.7 Proteger bloques

9.1.7.1 Proteger bloques

Introducción

Puede proteger por contraseña uno o varios bloques del tipo OB, FB, FC y los bloques de datos globales contra accesos no autorizados. Tenga en cuenta las siguientes particularidades:

- No es posible proteger manualmente los bloques de datos de instancia, sino que dependen de la protección de know-how del FB asignado. Esto significa que si se genera un bloque de datos instancia para un FB con protección de know-how, el bloque de datos instancia obtiene también la protección de know-how. Esto sucede independientemente de si se crea el bloque de datos instancia de forma explícita o si se genera por una llamada de bloque.
- En los bloques de datos globales y en los bloques de datos de instancia, la protección de know-how tiene el mismo efecto que la protección contra escritura.
- Los bloques de datos ARRAY no pueden dotarse de protección de know-how.

En un bloque con protección de know-how pueden leerse únicamente los datos siguientes sin una contraseña correcta:

- Parámetros de transferencia Input, Output, InOut, Return, Static
- Título del bloque
- Comentario del bloque
- Propiedades de bloques
- Variables de bloques de datos globales sin indicación de la ubicación

Además, en un bloque con protección de know-how pueden realizarse las acciones siguientes:

- Copiar y borrar
- Llamar en un programa
- Comparación online/offline
- Cargar

En cambio, el código del bloque no se puede leer ni modificar si no se dispone de la autorización necesaria. Además, para las CPU S7-1200/1500 se puede establecer una

protección contra copia que enlace la ejecución del bloque en la CPU o la tarjeta SD con el número de serie definido.

Nota

Tenga en cuenta las indicaciones siguientes:

- S7-1200 versión 1.0 y S7-300/400 (solo bloques GRAPH y SCL): Si carga un bloque con protección de know-how en un dispositivo, no se cargará la información de restauración. Esto significa que un bloque con know-how protegido no se podrá volver a abrir si se carga del dispositivo, ni siquiera conociendo la contraseña correcta.
- En las versiones offline y online de los bloques con protección de know-how solo se comparan los datos sin protección.
- Sin la contraseña no es posible acceder al bloque.
- Si se inserta un bloque con protección de know-how a una librería, la plantilla maestra resultante obtiene también la protección de know-how.
- En los S7-1500, con la casilla de verificación "Bloque utilizable como elemento de librería con protección de know-how" de las propiedades del bloque recibirá una indicación de si el bloque puede utilizarse como elemento de librería protegido. Para ello, el bloque no puede utilizar variables de las áreas de operandos Salidas (Q), Entradas (I), Marcas (M), Temporizadores (T) o Contadores (Z) y no puede acceder a bloques de datos.
- Las referencias cruzadas a variables, marcas, entradas y salidas utilizadas en bloques con protección de know-how no se muestran ni siquiera después de introducir la contraseña correcta.
- Si modifica el número de un bloque, el componente binario cargable del bloque ya no será actual. Eso significa que el bloque debe volver a compilarse antes de cargarlo en un dispositivo, cosa que en el caso de un bloque con protección de know how solo es posible con la contraseña correcta. Este punto debe recordarse especialmente si se desea copiar un bloque con protección de know how a otro dispositivo que ya contenga un bloque con ese número.
- Transmita siempre como fichero de proyecto o de librería un proyecto que contenga bloques con protección de know-how. De ese modo, se asegurará de que no se omita la protección de know-how.
- Si desea asignar una protección de know-how a varios bloques mediante selección múltiple, la selección no puede contener ningún bloque de datos de instancia. De lo contrario no podrá activarse la protección de know-how.
- En un bloque abierto con protección de know-how no pueden modificarse los ajustes de protección de know-how.

Consulte también

Establecer y quitar la protección contra copia de un bloque (Página 3048)

Configurar la protección de know-how para bloques (Página 3049)

Abrir bloques con protección de know-how (Página 3050)

Imprimir bloques con protección de know-how (Página 3050)

Quitar la protección de know-how para los bloques (Página 3052)

Cambiar la contraseña (Página 3052)

Archivación de proyectos (Página 358)

Archivar librerías globales (Página 450)

9.1.7.2 Establecer y quitar la protección contra copia de un bloque

Para las CPU S7-1200/1500 se puede configurar una protección contra copia, de modo que la ejecución del bloque quede ligada a una CPU o una tarjeta SD determinadas. Sólo si el bloque se encuentra en el dispositivo con el número de serie definido es posible ejecutarlo.

Si se establece una protección contra copia para un bloque, es importante dotar a ese bloque también con una protección de know-how. De lo contrario, cualquier persona puede desactivar la protección contra copia.

Nota

S7-1500 y S7-1200 V2.2 y superior: Si carga un bloque con protección de copia en un dispositivo que no coincide con el número de serie determinado, se rechaza todo el proceso de carga. Eso significa que tampoco se cargan bloques sin protección contra copia.

Configurar protección contra copia

Para configurar la protección contra copia, proceda del siguiente modo:

1. Abra el bloque para el que desea configurar la protección contra copia.
2. Abra la ficha "Propiedades" de la ventana de inspección.
3. Seleccione la entrada "Protección" en la ventana de inspección, en la navegación local.
4. Seleccione la entrada "Asociar al número de serie de la CPU" o "Asociar a número de serie de la Memory Card" en el área "Protección contra copia" de la lista desplegable.
5. En una CPU S7-1500, introduzca el número de serie de la CPU o de la Memory Card. En una CPU S7-1200, puede introducir directamente el número de serie o activar la opción "Número de serie agregado al cargar en un dispositivo o una Memory Card", si el número de serie se debe insertar automáticamente en el proceso de carga.
6. En el área "Protección de know-how" se puede configurar la protección de know-how del bloque, siempre que el bloque no tenga ya protección de know-how.

Quitar protección contra copia

Para quitar la protección contra copia, proceda del siguiente modo:

1. Abra el bloque del que desee eliminar la protección contra copia.
2. Abra la ficha "Propiedades" de la ventana de inspección.
3. Seleccione la entrada "Protección" en la ventana de inspección, en la navegación local.
4. Seleccione la entrada "Sin binding" de la lista desplegable en el área "Protección contra copia".

Consulte también

Proteger bloques (Página 3045)

Configurar la protección de know-how para bloques (Página 3049)

Abrir bloques con protección de know-how (Página 3050)

Imprimir bloques con protección de know-how (Página 3050)

Quitar la protección de know-how para los bloques (Página 3052)

Cambiar la contraseña (Página 3052)

9.1.7.3 Configurar la protección de know-how para bloques

Es posible configurar una protección de know-how para los bloques en los dispositivos del proyecto.

Procedimiento

Para configurar una protección de know-how para bloques, proceda del siguiente modo:

1. Seleccione los bloques sin protección de know-how que desea proteger.
2. En el menú "Edición", elija el comando "Protección de know-how".
   Se abre el cuadro de diálogo "Protección de know-how".
3. Haga clic en "Definir".
   Se abre el cuadro de diálogo "Definir contraseña".
4. Introduzca una contraseña en el campo "Nueva".
5. Repita la contraseña en el campo "Confirmar".
6. Confirme la entrada con "Aceptar".
7. Cierre el cuadro de diálogo "Protección de know-how" haciendo clic en "Aceptar".

Resultado

Los bloques seleccionados se dotan con una protección de know-how. Los bloques con protección de know-how se marcan con un candado en el árbol del proyecto. La contraseña es válida para todos los bloques seleccionados.

Consulte también

Proteger bloques (Página 3045)

Establecer y quitar la protección contra copia de un bloque (Página 3048)

Abrir bloques con protección de know-how (Página 3050)

Imprimir bloques con protección de know-how (Página 3050)

Quitar la protección de know-how para los bloques (Página 3052)

Cambiar la contraseña (Página 3052)

9.1.7.4 Abrir bloques con protección de know-how

Sólo es posible abrir varios bloques con protección de know-how en conjunto, si se protegen con la misma contraseña.

Procedimiento

Para abrir un bloque con protección de know-how, proceda del siguiente modo:

1. Haga clic en el bloque que desea abrir.
   Se abre el cuadro de diálogo "Protección de acceso".
2. Introduzca la contraseña para el bloque con protección de know-how.
3. Confirme las entradas con "Aceptar".

Resultado

Si se introduce la contraseña correcta, se abre el bloque con protección de know-how. Sin embargo, la protección de know-how del bloque no se anula. Si el bloque se copia o se inserta a una librería, el know-how de las copias también estará protegido.

Después de abrir el bloque, el código del programa y la interfaz del bloque se pueden editar hasta que se cierre el bloque o el TIA Portal. Cuando se vuelva a abrir el bloque, será necesario volver a introducir la contraseña. Si se cierra el cuadro de diálogo "Protección de acceso" con "Cancelar", el bloque se abre pero el código del bloque no se visualiza y por consiguiente el bloque no se puede editar.

Consulte también

Proteger bloques (Página 3045)

Establecer y quitar la protección contra copia de un bloque (Página 3048)

Configurar la protección de know-how para bloques (Página 3049)

Imprimir bloques con protección de know-how (Página 3050)

Quitar la protección de know-how para los bloques (Página 3052)

Cambiar la contraseña (Página 3052)

9.1.7.5 Imprimir bloques con protección de know-how

Sólo es posible imprimir completamente los bloques con protección de know-how, si el bloque se abre con la contraseña correcta. Si se imprime un bloque cerrado o se abre con una contraseña incorrecta, sólo se imprimirán los datos del bloque que no estén protegidos.

Procedimiento

Para imprimir completamente un bloque con protección de know-how, proceda del siguiente modo:

1. Abra el bloque con protección de know-how que desea imprimir.
   Consulte también: Abrir bloques con protección de know-how (Página 3050)
2. Elija el comando "Imprimir" del menú "Proyecto".
   Se abre el cuadro de diálogo "Imprimir".
3. Seleccione la impresora en el campo "Nombre".
4. Haga clic en "Avanzadas", para editar los ajustes de Windows de la impresora.
5. Seleccione en la lista desplegable "Información del documento" el juego de datos del documento que se debe utilizar para el diseño del borde.
6. En "Impresión de objeto/área desactivada", elija si desea imprimir todo el objeto o el área completa, o bien únicamente una selección.
7. Seleccione el alcance de la impresión en "Propiedades".
   - Elija "Todos" para imprimir todo el bloque.
   - Elija "Visible" para imprimir toda la información actualmente visible en pantalla del interior del bloque.
   - Elija "Compacto" para imprimir el bloque en un formato reducido.
8. Haga clic en "Vista preliminar" para generar una vista preliminar.
   En el área de trabajo se crea una vista preliminar.
9. Haga clic en "Imprimir" para iniciar la impresión.

Consulte también

Imprimir contenidos del proyecto (Página 390)

Proteger bloques (Página 3045)

Establecer y quitar la protección contra copia de un bloque (Página 3048)

Configurar la protección de know-how para bloques (Página 3049)

Quitar la protección de know-how para los bloques (Página 3052)

Cambiar la contraseña (Página 3052)

9.1.7.6 Cambiar la contraseña

Procedimiento

Para cambiar la contraseña, proceda del siguiente modo:

1. Seleccione los bloques con protección de know-how cuya contraseña desea cambiar.

   Nota

   Si desea cambiar simultáneamente la contraseña de varios bloques, los bloques seleccionados deben tener la misma contraseña.

2. En el menú "Edición", elija el comando "Protección de know-how".
   Se abre el cuadro de diálogo "Protección de know-how".
3. Haga clic en el botón "Cambiar".
4. Introduzca la contraseña antigua en el campo "Antigua".
5. Introduzca la contraseña nueva en el campo "Nueva".
6. Repita la contraseña nueva en el campo "Confirmar".
7. Confirme la entrada con "Aceptar".
8. Cierre el cuadro de diálogo "Protección de know-how" haciendo clic en "Aceptar".

Consulte también

Proteger bloques (Página 3045)

Establecer y quitar la protección contra copia de un bloque (Página 3048)

Configurar la protección de know-how para bloques (Página 3049)

Abrir bloques con protección de know-how (Página 3050)

Imprimir bloques con protección de know-how (Página 3050)

Quitar la protección de know-how para los bloques (Página 3052)

9.1.7.7 Quitar la protección de know-how para los bloques

Procedimiento

Para quitar la protección de know-how para los bloques, proceda del siguiente modo:

1. Seleccione el bloque cuya protección de know-how desea quitar.

   Nota

   Si desea quitar simultáneamente la protección de know-how de varios bloques, los bloques seleccionados deben tener la misma contraseña.

2. En el menú "Edición", elija el comando "Protección de know-how".
   Se abre el cuadro de diálogo "Protección de know-how".

3. Desactive la casilla de verificación "Ocultar código (protección de know how)".
4. Introduzca la contraseña.
5. Confirme la entrada con "Aceptar".

Resultado

La protección de know-how del bloque se elimina.

Consulte también

Proteger bloques (Página 3045)
Establecer y quitar la protección contra copia de un bloque (Página 3048)
Configurar la protección de know-how para bloques (Página 3049)
Abrir bloques con protección de know-how (Página 3050)
Imprimir bloques con protección de know-how (Página 3050)
Cambiar la contraseña (Página 3052)

9.2 Mostrar información del programa

9.2.1 Resumen de la información del programa disponible

Información del programa

La información disponible para un programa de usuario contiene las vistas indicadas en la tabla siguiente.

Vista	Aplicación
Plano de ocupación (Página 3054)	Proporciona una vista general de los bits de los operandos de las áreas de memoria I, Q y M que ya están ocupados en el programa de usuario. Además, indica si una dirección está ocupada por un acceso desde un programa S7 o si está asignada a un módulo S7 SIMATIC.
Estructura de llamadas (Página 3063)	Muestra la jerarquía de llamada de los bloques dentro de un programa de usuario y proporciona una vista general de los bloques utilizados y sus dependencias.

Vista	Aplicación
Estructura de dependencias (Página 3070)	Muestra la lista de los bloques utilizados en el programa de usuario. En el primer nivel aparece el bloque en cuestión y debajo se relacionan sangrados los bloques que lo llaman o utilizan. Al contrario que en la estructura de llamada, aquí los bloques de datos instancia se listan por separado.
Carga de la memoria (Página 3076)	Muestra la carga de la memoria de hardware de la CPU para objetos (OB, FC, FB, DB, tipos de datos definidos por el usuario y variables PLC), para áreas de la memoria de la CPU y para los módulos de entradas y salidas existentes.

Representación simultánea de varias vistas

Es posible generar y visualizar varias vistas para uno o varios programas de usuario, con objeto de facilitar la comprobación y modificación del programa de usuario.

La visualización de varias vistas permite, p. ej.:

- Visualizar toda la información de un programa de usuario dispuesta una junto a otra
- Realizar una comparación de diferentes programas de usuario

9.2.2 Visualizar el plano de ocupación

9.2.2.1 Introducción al plano de ocupación

Información del programa en el plano de ocupación

El plano de ocupación indica si una dirección está ocupada por un acceso desde un programa S7 o si está asignada a un módulo S7 SIMATIC. Por ello, constituye un punto de partida importante a la hora de buscar errores o realizar modificaciones en el programa de usuario.

El plano de ocupación proporciona una vista general específica de la CPU que indica qué bit se utiliza en qué byte de las áreas de memoria listadas a continuación:

- Entrada (I)
- Salida (Q)
- Marca (M)
- Temporizadores (T)
- Contadores (C)
- Periferia (P)

Visualización del plano de ocupación

El plano de ocupación se visualiza en varias ventanas de trabajo, separadas por entradas, salidas y marcas.

Filtrar

La vista dentro del plano de ocupación puede filtrarse. Para ello pueden utilizarse filtros predefinidos o crearse filtros propios.

Visualización de la información de las referencias cruzadas

En el plano de ocupación es posible visualizar la información de las referencias cruzadas de las direcciones seleccionadas.

Las referencias cruzadas de una dirección seleccionada se muestran en la ventana de inspección con el comando "Información de referencia cruzada" del menú contextual. Además, con el comando "Herramientas > Referencias cruzadas" se puede abrir la lista de referencias cruzadas del objeto seleccionado.

Visualización de la tabla de variables PLC

Desde el plano de ocupación se puede abrir la tabla de variables PLC y editar en ella las propiedades de las variables utilizadas.

Seleccione para ello una dirección del plano de ocupación y elija el comando "Abrir editor" del menú contextual.

Activar la indicación de la remanencia

En el plano de ocupación es posible activar y desactivar la indicación de la remanencia de las marcas mediante el botón "Mostrar/ocultar remanencia" de la barra de herramientas.

Consulte también

Símbolos del plano de ocupación (Página 3056)

Estructura del plano de ocupación (Página 3055)

9.2.2.2 Estructura del plano de ocupación

Estructura del plano de ocupación

El plano de ocupación se muestra en varias ventanas de trabajo en función de la CPU, con los siguientes operandos.

Para CPU S7-300/400:

- Entradas
- Salidas

- Marcas
- Temporizadores
- Contadores

Para CPU S7-1200:

- Entradas
- Salidas
- Marcas

Vista de entradas, salidas, marcas, temporizadores y contadores

Se muestran todos los operandos utilizados, así como su ocupación en el programa S7.

Cada fila del plano de ocupación contiene, para cada uno de los operandos mostrados, un byte del área de memoria en el que se identifican los ocho bits correspondientes de 7 a 0 en función del acceso. A continuación se indica mediante una "barra" si el acceso corresponde a un byte (B), a una palabra (W) o a una palabra doble (D).

Aquí encontrará una explicación de los símbolos utilizados en el plano de ocupación: (Página 3056)

Consulte también

Introducción al plano de ocupación (Página 3054)

9.2.2.3 Símbolos del plano de ocupación

Significado de los símbolos del plano de ocupación

La tabla siguiente muestra el significado de los símbolos del plano de ocupación:

Símbolo	Significado
	Muestra la asignación de direcciones en estado seleccionado.
	Muestra la asignación de direcciones en estado no seleccionado.
	Indica que una dirección inicial de puntero y una dirección de variable acceden a la misma área de direccionamiento y que están seleccionadas.
	Indica que una dirección inicial de puntero y una dirección de variable acceden a la misma área de direccionamiento y que no están seleccionadas.
	Muestra la asignación de punteros en estado seleccionado.
	Muestra la asignación de punteros en estado no seleccionado.

Símbolo	Significado
	Indica que el byte está ocupado por un acceso de byte y que la variable correspondiente está seleccionada. A través del menú contextual se puede visualizar la información de las referencias cruzadas de la variable seleccionada, así como la tabla de variables PLC.
	Indica que el byte está ocupado por un acceso de byte y que la variable correspondiente no está seleccionada.
	Indica que el byte está ocupado por un acceso de palabra y que la variable correspondiente está seleccionada. A través del menú contextual se puede visualizar la información de las referencias cruzadas de la variable seleccionada, así como la tabla de variables PLC.
	Indica que el byte está ocupado por un acceso de palabra y que la variable correspondiente no está seleccionada.
	Indica que el byte está ocupado por un acceso de palabra doble y que la variable correspondiente está seleccionada. A través del menú contextual se puede visualizar la información de las referencias cruzadas de la variable seleccionada, así como la tabla de variables PLC.
	Indica que el byte está ocupado por un acceso de palabra doble y que la variable correspondiente no está seleccionada.
Color de fondo: gris	Indica que hay un acceso de byte, palabra o palabra doble y que la dirección también está ocupada por el hardware. El color de fondo gris indica que hay un acceso de memoria solapado.
Color de fondo: amarillo	Indica que la dirección no está ocupada por el hardware.
	Indica que el área de memoria está definida como marca de sistema.
	Indica que el área de memoria está definida como marca de ciclo.

Consulte también

Estructura del plano de ocupación (Página 3055)

Introducción al plano de ocupación (Página 3054)

9.2.2.4 Visualizar el plano de ocupación

Requisitos

Se ha creado un proyecto con bloques programados.

Procedimiento

Para visualizar el plano de ocupación, proceda del siguiente modo:

1. Seleccione la carpeta "Bloques de programa" o bien uno o varios de los bloques que contiene.
2. Elija el comando "Plano de ocupación" del menú "Herramientas".

Resultado

Se abrirá el plano de ocupación del programa seleccionado.

Opciones de visualización en el plano de ocupación

Tenga en cuenta las opciones de visualización configuradas para visualizar la información que desee ver en el plano de ocupación.

Consulte también

Configurar las opciones de visualización del plano de ocupación (Página 3058)

Estructura del plano de ocupación (Página 3055)

9.2.2.5 Configurar las opciones de visualización del plano de ocupación

Introducción

Para el plano de ocupación existen las siguientes opciones de visualización:

- Direcciones utilizadas:
  Si esta casilla de verificación está activada, se muestran las direcciones, la periferia y los punteros utilizados.
- Direcciones de hardware libres:
  Si esta casilla de verificación está activada, sólo se muestran las direcciones de hardware libres.

Requisitos

- Se ha creado un proyecto con bloques programados.
- El plano de ocupación está abierto.

Procedimiento

Para configurar las opciones de visualización del plano de ocupación, proceda del siguiente modo:

1. Haga clic en el botón ("Opciones de la vista") de la barra de herramientas.
   Se abren las opciones de visualización del plano de ocupación. Las opciones de visualización activadas aparecen con marcas de verificación antepuestas.
2. Para activar o desactivar una opción de visualización, haga clic en la casilla correspondiente con objeto de activar o desactivar la marca de verificación.

Resultado

Las opciones de visualización se configurarán y en el plano de ocupación se mostrará la información deseada.

9.2.2.6 Posibilidades de filtrado en el plano de ocupación

Configuración de filtros

Es posible definir ajustes de filtrado personalizados para el plano de ocupación. Las siguientes posibilidades están disponibles para definir filtros:

- Visualizar todas las direcciones utilizadas de las áreas de direccionamiento indicadas a continuación.
- Visualizar direcciones individuales definidas del área de direccionamiento seleccionada, p. ej. "0" y "200".
- Visualizar áreas completas del área de direccionamiento seleccionada, p. ej. "0 - 256".

La tabla siguiente muestra una sinopsis de todas las posibilidades de selección:

Selección en	Selección	Símbolo	Significado
Área de direccionamiento	Pueden seleccionarse todas las direcciones (I, Q, M, T, C) mostradas dependientes de la CPU, como en el ajuste predeterminado, o solo áreas de direccionamiento individuales.	La casilla de verificación está activada	En el plano de ocupación se muestran solo las áreas de direccionamiento activadas (I, Q, M, T, C).
Área de filtro	Mostrar la ocupación de todas las direcciones	*	Se muestra la ocupación de todas las direcciones utilizadas para las áreas de direccionamiento activadas (I, Q, M).

Selección en	Selección	Símbolo	Significado
	Mostrar la ocupación de las direcciones seleccionadas, p. ej. de las entradas "IB 0" e "IB 256"	0;256 Las diferentes direcciones y áreas están separadas por punto y coma.	Se muestra la ocupación de las direcciones citadas para las áreas de direccionamiento activadas (I).
	Mostrar la ocupación de las áreas seleccionadas, p. ej. de las entradas de "IB 0 hasta IB 100" y de "IB 200 hasta IB 256".	0-100;200-256 Las áreas relacionadas deben unirse con un guión.	Se muestra la ocupación de las áreas citadas para las áreas de direccionamiento activadas (I).

9.2.2.7 Definir filtros para el plano de ocupación

Requisitos

- Se ha creado un proyecto con bloques programados.
- El plano de ocupación está abierto.

Definir filtros

Para definir un filtro para el plano de ocupación, proceda del siguiente modo:

1. Haga clic en el botón ("Filtro") de la barra de herramientas.
   Se abre el diálogo "Lista de asignaciones Filtro".
2. Haga clic en el botón ("Crear nuevo filtro").
   Se crea un filtro nuevo con el nombre "Filtro_1". En el ajuste predeterminado para el filtro están activadas las casillas de verificación de todas las direcciones (p. ej., entradas, salidas, marcas, temporizadores y contadores) en función de la CPU.
3. Para cambiar el nombre del filtro, haga clic en la lista desplegable de la barra de herramientas e introduzca un nuevo nombre de filtro.
4. Desactive las casillas de verificación de las direcciones a las que no desee aplicar el filtro.
5. Introduzca una de las siguientes opciones en el área de filtro de las direcciones activadas:
   - Mostrar todas las direcciones utilizadas = "*"
   - Mostrar direcciones individuales definidas, p. ej. IB 0 e IB 25 = "0,25". Las diferentes direcciones y áreas de direccionamiento están separadas por comas o punto y coma.
   - Mostrar las áreas de direccionamiento completas, p. ej. de IB 0 a IB 256 = "0-256". Las áreas de direccionamiento completas deben unirse con un guión.
6. Confirme las entradas con "Aceptar".
   El nuevo filtro se visualiza con el nombre predeterminado en la barra de herramientas del plano de ocupación.

Borrar filtros

Para borrar un filtro, proceda del siguiente modo:

1. Haga clic en el botón ("Filtro") de la barra de herramientas.
   Se abre el cuadro de diálogo "Filtro" para el plano de ocupación.
2. Seleccione el filtro que desee borrar en la lista desplegable de la barra de herramientas.
3. Haga clic en el botón ("Borrar filtro seleccionado").
   El filtro seleccionado se borra.

Consulte también

Posibilidades de filtrado en el plano de ocupación (Página 3059)

Visualizar el plano de ocupación (Página 3057)

Introducción al plano de ocupación (Página 3054)

9.2.2.8 Filtrar el plano de ocupación

Requisitos

- Se ha creado un proyecto con bloques programados.
- El plano de ocupación está abierto.

Procedimiento

1. Haga clic en la flecha de la lista desplegable de la barra de herramientas.
   Se mostrarán los filtros disponibles.
2. Seleccione el filtro que desee utilizar.

Resultado

El plano de ocupación se filtrará según la configuración del filtro seleccionado.

Nota

La configuración del filtro se guarda al cerrar el proyecto.

9.2.2.9 Definir áreas de memoria remanentes para marcas

Introducción

En el plano de ocupación es posible definir el ancho del área de memoria remanente para marcas. El contenido de las variables direccionadas en el área de memoria remanente se conserva tras desconectar y volver a conectar la alimentación en un cambio de STOP a RUN.

La visualización de las marcas remanentes en el plano de ocupación puede activarse y desactivarse. Con la visualización correspondiente activada, las marcas remanentes se distinguen por el icono de la columna "Dirección".

Requisitos

El plano de ocupación está abierto.

Procedimiento

Para definir el ancho del área de memoria remanente para marcas, proceda del siguiente modo:

1. Haga clic en el botón "Remanencia" de la barra de herramientas.
   Se abre el cuadro de diálogo "Memoria remanente".
2. Defina el ancho del área de memoria remanente introduciendo en el campo de entrada el último byte del área contando a partir de 0. Tenga en cuenta las direcciones de variables existentes definidas como remanentes.
3. A continuación cargue los bloques modificados en el sistema de destino. Seleccione para ello la carpeta "Bloques de programa" del árbol del proyecto y elija el comando "Cargar en dispositivo" del menú contextual.

Resultado

El ancho del área de memoria remanente se definirá. Si está activada la indicación de la remanencia en el plano de ocupación, en la columna "Dirección" aparecerá el icono correspondiente para todas las variables remanentes.

9.2.2.10 Activar la indicación de la remanencia de marcas

Introducción

En el plano de ocupación se puede activar y desactivar la indicación de las marcas remanentes. Si activa la indicación de la remanencia, las marcas definidas como remanentes se identifican con un icono en la columna "Dirección".

Requisitos

El plano de ocupación está abierto.

Procedimiento

Para activar y desactivar la indicación de las marcas remanentes, proceda del siguiente modo:

1. Haga clic en el botón "Mostrar/ocultar remanencia" de la barra de herramientas.

Resultado

Si está activada la indicación de la remanencia, las variables definidas como remanentes se identificarán con un icono en la columna "Dirección" del área de marcas. Si se desactiva la indicación de la remanencia, se ocultarán los iconos de la columna "Dirección".

9.2.3 Mostrar la estructura de llamada

9.2.3.1 Introducción a la estructura de llamadas

Estructura de llamadas

La estructura de llamada describe la jerarquía de llamada de los bloques dentro de un programa S7.

Proporciona una vista general sobre:

- Bloques utilizados
- Saltos a las ubicaciones de los bloques
- Dependencias entre los bloques
- Requerimiento de datos locales de los bloques
- Estado de los bloques

Información contenida en la estructura de llamada

En la vista de la estructura de llamada aparece una lista de los bloques utilizados en el programa de usuario. El primer nivel de la estructura de llamada está resaltado en color y muestra los bloques que no son llamados por ningún otro bloque del programa. Los bloques de organización siempre se muestran en el primer nivel de la estructura de llamada. Los bloques de función, bloques de datos y funciones sólo se muestran en el primer nivel si no son llamados por ningún bloque de organización. Si un bloque llama otros bloques o funciones,

éstos se indican debajo del bloque en una lista sangrada. Las instrucciones y los bloques solo se representan en la estructura de llamadas si son llamados por un bloque.

Opciones de visualización

Para la estructura de llamada existen las siguientes opciones de visualización:

- Mostrar sólo conflictos:
  Si esta casilla de verificación está activada, en la estructura de llamada sólo se mostrarán conflictos.
- Agrupar varias llamadas:
  Si esta casilla de verificación está activada, se agruparán varias llamadas de bloque. El número de llamadas de bloque se indica en la columna "Número de llamadas". Los enlaces a los diferentes puntos de llamada se ofrecen en una lista desplegable en la columna "Detalles".

Visualización de las llamadas de bloque

Para ver las llamadas de bloque contenidas en un bloque, haga clic en la flecha antepuesta al título del bloque. Para ver la información de llamada de todos los bloques, haga clic en el símbolo "Ampliar lista" de la barra de herramientas.

Para contraer la vista general, haga clic en el símbolo "Reducir lista".

Visualización de la información de las referencias cruzadas

Para ver la información de las referencias cruzadas de un bloque en la ventana de inspección, haga clic en el bloque en cuestión con el botón derecho del ratón y seleccione la opción "Información de referencia cruzada" del menú contextual.

Para abrir la vista "Referencias cruzadas", haga clic en la opción "Referencias cruzadas" del menú contextual.

Mostrar los bloques en el editor de programación

Desde la estructura de llamada se puede abrir el editor de programación y editar allí los bloques.

Para ello seleccione el bloque deseado en la estructura de llamadas y elija el comando "Abrir editor" del menú contextual.

Mostrar los bloques borrados

Las filas correspondientes a bloques borrados se identifican con un símbolo.

Nota

Tenga en cuenta que los datos locales, si los hay, no se podrán mostrar o actualizar hasta que se haya compilado un bloque.

Consulte también

Símbolos utilizados en la estructura de llamadas (Página 3065)

9.2.3.2 Símbolos utilizados en la estructura de llamadas

Significado de los símbolos utilizados en la estructura de llamada

La tabla siguiente muestra el significado de los símbolos de la estructura de llamada:

Símbolo	Significado
	Indica un bloque de organización (OB).
	Indica un bloque de función (FB).
	Indica una función (FC).
	Indica un bloque de datos (OB).
	Indica que el bloque está declarado como multiinstancia.
	Indica que el objeto tiene una dependencia de interfaz con el objeto conectado a la izquierda.
	Indica que el bloque debe volver a compilarse.
	Indica que el bloque de datos debe volver a compilarse.
	Indica que el objeto no existe.
	Indica que la interfaz causa un conflicto de sello de tiempo.
	Indica que una variable causa un conflicto de sello de tiempo.
	Indica que el bloque no es llamado directa ni indirectamente por un OB.
	Indica que un objeto dispone de protección de know-how.
	Indica que el bloque se llama regularmente de forma recursiva.
	Indica que una declaración de variables de la interfaz tiene una dependencia recursiva: • Caso 1: FB1 llama a FB2 y éste llama a FB1. Los bloques de datos de instancia de estos FB presentan una recursión en la interfaz. • Caso 2: Un FB multiinstancia utiliza el DB de instancia de su FB padre como DB global.

9.2.3.3 Composición de la estructura de llamadas

Estructura de llamadas

La vista de la estructura de llamadas se compone de las siguientes columnas:

Columna	Contenido/significado
Estructura de llamadas	Muestra una vista general de los bloques llamados Si la opción "Agrupar varias llamadas" está activada, se agrupan varias llamadas de bloque y se muestra la columna "Número de llamadas".
Tipo de llamada (!)	Muestra el tipo de llamada, p. ej. llamada de bloque recursiva.
Dirección	Muestra la dirección absoluta del bloque. En un bloque de función también se muestra la dirección absoluta del bloque de datos instancia correspondiente.
Número de llamadas	Indica el número de llamadas múltiples de bloques.
Detalles	Muestra el segmento o interfaz del bloque que llama. Toda la información de esta columna se ofrece en forma de enlace. A través de este enlace se accede a la posición del editor de programación en la que se encuentra la llamada del bloque. Si la opción "Agrupar varias llamadas" está activada, las llamadas se agrupan y los enlaces aparecen en una lista desplegable.
Datos locales (en la ruta)	Muestra el requerimiento de datos locales clásico de toda la ruta. Los bloques con acceso optimizado requieren un mayor número de datos locales, ya que también se almacena la información de la dirección simbólica. Tenga en cuenta que los datos locales, si los hay, no se podrán mostrar o actualizar hasta que se haya compilado un bloque.
Datos locales (para bloques)	Muestra el requerimiento de datos locales del bloque. Los bloques con acceso optimizado requieren un mayor número de datos locales, ya que también se almacena la información de la dirección simbólica. Tenga en cuenta que los datos locales, si los hay, no se podrán mostrar o actualizar hasta que se haya compilado un bloque.

Consulte también

Símbolos utilizados en la estructura de llamadas (Página 3065)

Introducción a la comprobación de coherencia en la estructura de llamada (Página 3069)

9.2.3.4 Mostrar la estructura de llamadas

Requisitos

Se ha creado un proyecto con bloques programados.

Procedimiento

Para ver la estructura de llamada, proceda del siguiente modo:

1. Seleccione la carpeta "Bloques de programa" o bien uno o varios de los bloques que contiene.
2. Elija el comando "Estructura de llamadas" del menú "Herramientas".

Resultado

Se abrirá la estructura de llamadas del programa seleccionado.

Nota

Tenga en cuenta que los datos locales, si los hay, no se podrán mostrar o actualizar hasta que se haya compilado un bloque.

Consulte también

Configurar las opciones de visualización de la estructura de llamadas (Página 3068)

9.2.3.5 Configurar las opciones de visualización de la estructura de llamadas

Introducción

Para la estructura de llamada se dispone de las siguientes opciones de visualización:

- Mostrar sólo conflictos:
  Si esta casilla de verificación está activada, en la estructura de llamada sólo se mostrarán bloques con conflictos.
  Los siguientes bloques causan un conflicto:
  - Bloques que tienen un sello de tiempo del código anterior o posterior a una de sus llamadas.
  - Bloques que llaman un bloque cuya interfaz ha cambiado.
  - Bloques que utilizan una variable cuya dirección y/o tipo de datos ha cambiado.
  - Bloques que no son llamados ni directa ni indirectamente por un OB.
  - Bloques que llaman un bloque que ya no existe.
- Agrupar varias llamadas:
  Si esta opción de visualización está activada, se agruparán varias llamadas de bloque y varios accesos a bloques de datos. El número de llamadas de bloque se indica en la columna "Número de llamadas". Los enlaces a los diferentes puntos de llamada se ofrecen en una lista desplegable en la columna "Detalles".

Requisitos

- Se ha creado un proyecto con bloques programados.
- La estructura de llamada está abierta.

Procedimiento

Para configurar las opciones de visualización de la estructura de llamada, proceda del siguiente modo:

1. Haga clic en el botón ("Opciones de la vista") de la barra de herramientas.
   Se abren las opciones de visualización de la estructura de llamada. Las opciones de visualización activadas aparecen con marcas de verificación antepuestas.
2. Para activar o desactivar una opción de visualización, haga clic en la casilla correspondiente con objeto de activar o desactivar la marca de verificación.

Resultado

Las opciones de visualización se configurarán y en la estructura de llamadas se mostrará la información deseada.

9.2.3.6 Introducción a la comprobación de coherencia en la estructura de llamada

Comprobación de coherencia

Los cambios del sello de tiempo de un bloque ocurridos durante o después de la creación del programa pueden provocar conflictos de sello de tiempo, que a su vez pueden derivar en incoherencias entre los bloques invocantes y los llamados.

Utilización de la comprobación de coherencia

La función "Comprobación de coherencia" permite visualizar incoherencias en conflictos de sello de tiempo. Cuando se realiza la comprobación de coherencia, los bloques incoherentes se muestran en la estructura de llamada y se identifican con los símbolos correspondientes.

- Gran parte de los conflictos de sello de tiempo y de interfaz pueden eliminarse volviendo a compilar los bloques.
- Si no es posible depurar incoherencias mediante compilación, se puede ir al punto concreto del problema en el editor de programación a través del enlace de la columna "Detalles" y depurar las incoherencias manualmente.
- Los bloques marcados en rojo tienen que volver a compilarse.

Consulte también

Símbolos utilizados en la estructura de llamadas (Página 3065)

9.2.3.7 Comprobar la coherencia de bloques en la estructura de llamada

Requisitos

- Se ha creado un proyecto con bloques programados.
- La estructura de llamada está abierta.

Procedimiento

Para comprobar la coherencia de bloques, proceda del siguiente modo:

1. Haga clic en el botón ("Comprobación de coherencia") de la barra de herramientas.
   Se comprobará la coherencia de bloques. Los bloques que presentan incoherencias se identifican con el símbolo correspondiente.
2. Si un bloque presenta incoherencias, haga clic en la flecha antepuesta al título del bloque correspondiente en la estructura de llamada.
   Se mostrarán los bloques incoherentes. En la columna "Detalles" se listan los puntos problemáticos en forma de enlaces.
3. Para saltar al punto del bloque que se debe depurar, haga clic en el enlace correspondiente de la columna "Detalles".

4. Compruebe y corrija las incoherencias de los bloques.
5. Vuelva a compilar los bloques seleccionando los bloques deseados y eligiendo el comando "Compilar" del menú contextual.
6. Cargue los bloques corregidos en el sistema de destino haciendo clic en el comando "Cargar en dispositivo" del menú contextual.

Resultado

Se comprobará la coherencia de bloques. Se depurarán las incoherencias de los bloques. Los bloques corregidos se cargarán en el sistema de destino.

Consulte también

Símbolos utilizados en la estructura de llamadas (Página 3065)

9.2.4 Mostrar la estructura de dependencias

9.2.4.1 Introducción a la estructura de dependencias

Introducción

La estructura de dependencias muestra la dependencia de cada bloque del programa respecto a otros bloques.

Información visualizada en la estructura de dependencias

En la vista de la estructura de dependencias aparece una lista de los bloques utilizados en el programa de usuario. En el extremo izquierdo se muestra el bloque en cuestión y debajo se relacionan sangrados los bloques que lo llaman o utilizan.

La estructura de dependencias muestra además el estado de los distintos bloques mediante símbolos.

Los objetos que causan un conflicto de sello de tiempo y que pueden provocar una incoherencia en el programa se identifican mediante diferentes símbolos.

La estructura de dependencias complementa la lista de referencias cruzadas de los objetos.

Opciones de visualización

Para la estructura de dependencias se dispone de las siguientes opciones de visualización:

- Mostrar sólo conflictos:
  Si esta casilla de verificación está activada, en la estructura de dependencias sólo se mostrarán conflictos.
- Agrupar varias llamadas:
  Si esta casilla de verificación está activada, se agruparán varias llamadas de bloque. El número de llamadas de bloque se indica con un valor numérico en la columna "Estructura de dependencias". Los enlaces a los diferentes puntos de llamada se ofrecen en una lista desplegable en la columna "Detalles".

Visualización de las dependencias

Al hacer clic en la flecha situada delante del título de un bloque se visualizan los bloques en los que se llama o utiliza dicho bloque. Para mostrar las dependencias de todos los bloques,

haga clic en el símbolo "Ampliar lista" de la barra de herramientas.

Para contraer la vista general, haga clic en el símbolo "Reducir lista".

Visualización de la información de las referencias cruzadas

Para ver la información de las referencias cruzadas de un bloque en la ventana de inspección, haga clic en el bloque en cuestión con el botón derecho del ratón y seleccione el comando "Mostrar ubicación" del menú contextual.

Mostrar los bloques en el editor de programación

Desde la estructura de dependencias se puede abrir el editor de programación y editar en él los bloques. Para ello seleccione el bloque deseado en la estructura de dependencias y elija el comando "Abrir" del menú contextual.

9.2.4.2 Estructura de dependencias

Estructura de dependencias

La vista de la estructura de dependencias se compone de las siguientes columnas:

Columna	Contenido/significado
Dependencia	Muestra en un sinóptico las dependencias de cada bloque.
Tipo de llamada (!)	Muestra el tipo de llamada, p. ej. llamada de bloque recursiva.
Dirección	Muestra la dirección absoluta del bloque.

Columna	Contenido/significado
Número de llamadas	Indica el número de llamadas múltiples de bloques.
Detalles	Muestra el segmento o interfaz del bloque llamado. Toda la información de esta columna se ofrece en forma de enlace. A través de este enlace se accede a la posición del editor de programación en la que se encuentra la llamada del bloque. Si la opción "Agrupar varias llamadas" está activada, las llamadas se agrupan y los enlaces aparecen en una lista desplegable.

Consulte también

Símbolos utilizados en la estructura de dependencias (Página 3072)

9.2.4.3 Símbolos utilizados en la estructura de dependencias

Significado de los símbolos utilizados en la estructura de dependencias

La tabla siguiente muestra el significado de los símbolos de la estructura de dependencias:

Símbolo	Significado
	Indica un bloque de organización (OB).
	Indica un bloque de función (FB).
	Indica una función (FC).
	Indica un bloque de datos (OB).
	Indica que el objeto tiene una dependencia de interfaz con el objeto conectado a la izquierda.
	Indica que el bloque debe volver a compilarse.
	Indica que el bloque de datos debe volver a compilarse.
	Indica que existe una incoherencia para este objeto.
	Indica que un objeto dispone de protección de know-how.
	Indica que una declaración de variables de la interfaz tiene una dependencia recursiva: • Caso 1: FB1 llama a FB2 y éste llama a FB1. Los bloques de datos de instancia de estos FB presentan una recursión en la interfaz. • Caso 2: Un FB multiinstancia utiliza el DB de instancia de su FB padre como DB global.

9.2.4.4 Mostrar la estructura de dependencias

Requisitos

Se ha creado un proyecto con bloques programados.

Procedimiento

Para ver la estructura de dependencias, proceda del siguiente modo:

1. Seleccione la carpeta de bloques o bien uno o varios de los bloques que contiene.
2. Elija el comando "Estructura de dependencias" del menú "Herramientas".

Resultado

Se abrirá la estructura de dependencias del programa seleccionado.

Consulte también

Configurar las opciones de visualización de la estructura de dependencias (Página 3073)

9.2.4.5 Configurar las opciones de visualización de la estructura de dependencias

Introducción

Para la estructura de dependencias se dispone de las siguientes opciones de visualización:

- Mostrar sólo conflictos:
  Si esta casilla de verificación está activada, en la estructura de dependencias sólo se mostrarán conflictos.
  Los siguientes bloques causan un conflicto:
  - Bloques que tienen un sello de tiempo del código anterior o posterior a una de sus llamadas.
  - Bloques invocados por un bloque cuya interfaz ha cambiado.
  - Bloques que utilizan una variable cuya dirección y/o tipo de datos ha cambiado.
  - Bloques que no son llamados ni directa ni indirectamente por un OB.
- Agrupar varias llamadas:
  Si esta casilla de verificación está activada, se agruparán varias llamadas de bloque. El número de llamadas de bloque se indica en la columna correspondiente. Los enlaces a los diferentes puntos de llamada se ofrecen en una lista desplegable en la columna "Detalles".

Requisitos

- Se ha creado un proyecto con bloques programados.
- La estructura de dependencias está abierta.

Procedimiento

Para configurar las opciones de visualización de la estructura de dependencias, proceda del siguiente modo:

1. Haga clic en el botón ("Opciones de la vista") de la barra de herramientas.
   Se abren las opciones de visualización de la estructura de dependencias. Las opciones de visualización activadas aparecen con marcas de verificación antepuestas.
2. Para activar o desactivar una opción de visualización, haga clic en la casilla correspondiente con objeto de activar o desactivar la marca de verificación.

Resultado

Las opciones de visualización se configurarán y en la estructura de dependencias se mostrará la información deseada.

9.2.4.6 Introducción a la comprobación de coherencia en la estructura de dependencias

Comprobación de coherencia

Los cambios del sello de tiempo de un bloque ocurridos durante o después de crear el programa pueden provocar conflictos de sello de tiempo, que a su vez pueden derivar en incoherencias entre los bloques invocantes y los llamados.

Utilización de la comprobación de coherencia

La función "Comprobación de coherencia" permite visualizar incoherencias. Cuando se realiza la comprobación de coherencia, los bloques incoherentes se muestran en la estructura de dependencias y se identifican con los símbolos correspondientes.

- Gran parte de los conflictos de sello de tiempo y de interfaz pueden eliminarse volviendo a compilar los bloques.
- Si no es posible corregir las incoherencias mediante compilación, se puede ir al lugar concreto del problema en el editor de programación a través del enlace de la columna "Detalles" y corregir las incoherencias manualmente.
- Los bloques marcados en rojo tienen que volver a compilarse.

Consulte también

Estructura de dependencias (Página 3071)

Símbolos utilizados en la estructura de dependencias (Página 3072)

9.2.4.7 Comprobar la coherencia de bloques en la estructura de dependencias

Requisitos

- Se ha creado un proyecto con bloques programados.
- La estructura de dependencias está abierta.

Procedimiento

Para comprobar la coherencia de bloques, proceda del siguiente modo:

1. Haga clic en el botón ("Comprobación de coherencia") de la barra de herramientas.
   Se comprobará la coherencia de bloques. Los bloques que presentan incoherencias se marcan con el símbolo correspondiente.
2. Si un bloque presenta incoherencias, haga clic en la flecha antepuesta al título del bloque correspondiente en la estructura de dependencias.
   Se mostrarán los bloques incoherentes. En la columna "Detalles" se listan los puntos problemáticos en forma de enlaces.
3. Compruebe y corrija las incoherencias de los bloques.
4. Vuelva a compilar los bloques seleccionando los bloques deseados y eligiendo el comando "Compilar" del menú contextual.
5. Cargue los bloques corregidos en el sistema de destino haciendo clic en el comando "Cargar en dispositivo" del menú contextual.

Resultado

Se comprobará la coherencia de bloques. Se corregirán las incoherencias de los bloques. Los bloques corregidos se cargarán en el sistema de destino.

Consulte también

Símbolos utilizados en la estructura de dependencias (Página 3072)

9.2.5 Mostrar la carga de la memoria de la CPU

9.2.5.1 Introducción a la carga de la memoria

Introducción

La ficha "Carga de la memoria" muestra la carga de la memoria de hardware de la CPU configurada para:

- los objetos de programación utilizados
- la ocupación de las distintas áreas de memoria dentro de la CPU y
- las entradas y las salidas ocupadas en los módulos de entradas y salidas existentes.

Información de la ficha "Carga de la memoria"

La indicación de la carga de la memoria permite hacerse una idea de la carga de la memoria de hardware de la CPU: La visualización en esta ficha depende de la CPU utilizada. Se muestran la siguiente información:

- los objetos de programación existentes en la CPU (p. ej. OB, FC, FB, DB, tipos de datos y variables PLC),
- las áreas de memoria existentes en la CPU (memoria de carga, memoria de trabajo, que se divide a su vez según la CPU utilizada en la memoria de trabajo para código y la memoria de trabajo apra datos y memoria remanente), su tamaño máximo y su grado de ocupación por los objetos de programación utilizados,
- las entradas y salidas configurables para la CPU en los módulos disponibles (módulos de entradas/salidas, módulos de entradas digitales, módulos de salidas digitales, módulos de entradas analógicas y módulos de salidas analógicas), así como las entradas y salidas ya asignadas.

Indicación de la memoria de carga máxima disponible

Es posible seleccionar el tamaño máximo disponible de la memoria de carga en la lista desplegable de la fila "Total" de la columna "Memoria de carga".

Indicación de la memoria de trabajo máxima disponible

El tamaño máximo disponible de la memoria de trabajo se indica en la columna "Memoria de trabajo" o bien en las columnas "Memoria de trabajo para código" y "Memoria de trabajo para datos" en la fila "Total".

Indicación de la memoria remanente máxima disponible

Es posible seleccionar el tamaño máximo disponible de la memoria remanente en la fila "Total" de la columna "Memoria remanente".

Nota

Datos de remanencia

Las marcas y los bloques de datos definidos como remanentes se incluyen en el cálculo de los datos de remanencia.

Actualizar la vista de la ficha "Carga de la memoria"

Para actualizar la vista de los objetos mostrados, haga clic en el botón "Actualizar vista" de la barra de herramientas.

Ventajas de la vista de la ficha "Carga de la memoria"

En la ficha "Carga de la memoria" de la información del programa aparece una lista detallada de cada objeto utilizado y el área de memoria correspondiente ocupada.

También se indica una sobreocupación de la CPU, lo que permite impedirla.

Los bloques no compilados pueden identificarse gracias a que su tamaño se indica con interrogantes.

Consulte también

Estructura de la ficha "Carga de la memoria" (Página 3078)

Mostrar la carga de la memoria (Página 3079)

Seleccionar la memoria de carga máxima disponible (Página 3080)

9.2.5.2 Estructura de la ficha "Carga de la memoria"

Estructura de la ficha "Carga de la memoria" en la información del programa

La vista de la ficha "Carga de la memoria" se compone de las siguientes columnas:

Columna	Contenido/significado
Objetos	Muestra bajo "Detalles" una vista general de los objetos de programación existentes en la CPU y la correspondiente ocupación de la memoria.
Memoria de carga	Muestra la carga máxima de la memoria de carga de la CPU de forma porcentual y absoluta. Los valores indicados en "Total" especifican el espacio máximo disponible en la memoria de carga. Los valores indicados en "Ocupado" especifican el espacio ocupado realmente en la memoria de carga. Cuando un valor aparece en color rojo, significa que se ha rebasado la capacidad de memoria existente.
Memoria de trabajo o bien Memoria de trabajo para código y para datos	Muestra la carga máxima de la memoria de trabajo de la CPU de forma porcentual y absoluta. La memoria de trabajo depende de la CPU y se divide p. ej. en el caso de una CPU de la familia S7-400 y S7-1500 en "Memoria de trabajo para código" y "Memoria de trabajo para datos". Los valores indicados en "Total" indican el espacio máximo disponible en la memoria de trabajo. Los valores indicados en "Ocupado" indican el espacio ocupado realmente en la memoria de trabajo. Cuando un valor aparece en color rojo, significa que se ha rebasado la capacidad de memoria existente.
Memoria remanente	Muestra la carga máxima del área de memoria remanente de la CPU de forma porcentual y absoluta. Los valores indicados en "Total" indican el espacio máximo disponible en la memoria remanente. Los valores indicados en "Ocupado" indican el espacio ocupado realmente en la memoria remanente. Cuando un valor aparece en color rojo, significa que se ha rebasado la capacidad de memoria existente.

Columna	Contenido/significado
I/O	Indica las entradas y salidas disponibles en la CPU, que se muestran por tipos en las columnas indicadas a continuación. Los valores indicados en "Configurado" especifican las entradas y salidas máximas disponibles. Los valores indicados en "Ocupado" especifican las entradas y salidas ocupadas realmente.
DI / DQ / AI / AQ	Indica el número de entradas y salidas configuradas y utilizadas: DI = entradas digitales DQ = salidas digitales AI = entradas analógicas AQ = salidas analógicas Los valores indicados en "Configurado" especifican las entradas y salidas máximas disponibles. Los valores indicados en "Ocupado" especifican las entradas y salidas ocupadas realmente.

Consulte también

Mostrar la carga de la memoria (Página 3079)

Seleccionar la memoria de carga máxima disponible (Página 3080)

Introducción a la carga de la memoria (Página 3076)

9.2.5.3 Mostrar la carga de la memoria

Requisitos

Se ha creado un proyecto con bloques programados.

Procedimiento

Para visualizar la carga de la memoria para las diferentes áreas de memoria de la CPU, proceda del siguiente modo:

1. Seleccione la carpeta de bloques bajo la CPU deseada o bien uno o varios de los bloques que contiene.
2. Elija el comando "Carga de la memoria" del menú "Herramientas".

Resultado

Se mostrará la carga de la memoria de la CPU asignada.

9.2.5.4 Seleccionar la memoria de carga máxima disponible

Requisitos

Se ha creado un proyecto con bloques programados.

Procedimiento

Para seleccionar la memoria de carga máxima disponible en la carga de la memoria, proceda del siguiente modo:

1. Seleccione la carpeta de bloques bajo la CPU deseada o bien uno o varios de los bloques que contiene.
2. Elija el comando "Carga de la memoria" del menú "Herramientas".
3. En el cuadro de diálogo que aparece a continuación, haga clic en el símbolo para abrir la lista desplegable del campo "Total" de la columna "Memoria de carga".
4. Seleccione el valor para la CPU utilizada en la lista desplegable haciendo clic en él.

Resultado

En el campo "Total" se indica la carga de memoria máxima seleccionada.

Nota

Visualización de la carga de memoria máxima

Cuando en la carga de memoria un valor aparece en color rojo, significa que se ha rebasado la capacidad de memoria disponible.

En este caso, adapte la capacidad de memoria tal como se describe anteriormente.

9.3 Mostrar referencias cruzadas

9.3.1 Información general sobre las referencias cruzadas

Introducción

La lista de referencias cruzadas ofrece una vista general de la utilización de operandos y variables dentro del programa de usuario.

Utilidad de las referencias cruzadas

La lista de referencias cruzadas ofrece las siguientes ventajas:

- Al crear y modificar el programa se dispone de una vista general de los operandos, variables y llamadas de bloque utilizadas.
- Desde las referencias cruzadas es posible saltar directamente a la respectiva ubicación de los operandos y variables.
- Al comprobar el programa o buscar errores se obtiene información acerca de:
  - qué operando se procesa en qué bloque con qué instrucción
  - qué variable se utiliza en qué imagen
  - qué bloque es llamado por qué otro bloque
  - información sobre referencias cruzadas para estructuras de nivel superior o inferior.
- Como parte de la documentación del proyecto, las referencias cruzadas proporcionan una amplia vista general de todos los operandos, áreas de memoria, bloques, variables e imágenes utilizadas.

Consulte también

Estructura de la lista de referencias cruzadas (Página 3081)

Visualizar la lista de referencias cruzadas (Página 3083)

Visualizar las referencias cruzadas en la ventana de inspección (Página 3084)

9.3.2 Estructura de la lista de referencias cruzadas

Vistas de la lista de referencias cruzadas

Existen dos vistas de la lista de referencias cruzadas, que se diferencian en los objetos que aparecen en la primera columna:

- Utilizado por:
  Indicación de los objetos referenciados. Aquí se muestran las ubicaciones en las que se utiliza el objeto.
- Utilizado:
  Indicación de los objetos referenciantes. Aquí se muestran los usuarios del objeto.

Los tooltips asignados contienen información adicional sobre los distintos objetos.

Estructura de la lista de referencias cruzadas

La lista de referencias cruzadas tiene la estructura siguiente:

Columna	Contenido/significado
Objeto	Nombre del objeto que utiliza los objetos subordinados o que es utilizado por éstos.
Número	Número de utilizaciones
Ubicación	Ubicación correspondiente, p. ej. segmento
Propiedad	Propiedades especiales de los objetos referenciados, p. ej. el nombre de variables en declaraciones multiinstancia.
como	Muestra información adicional sobre el objeto, p. ej. el hecho de que un DB instancia se utiliza como plantilla o multiinstancia.
Acceso	Tipo de acceso, es decir, si se trata de un acceso de lectura (R) o escritura (W) al operando.
Dirección	Dirección del operando
Tipo	Información sobre el tipo y el lenguaje de creación del objeto
Ruta	Ruta del objeto en el árbol del proyecto

Dependiendo de los productos instalados, se muestran columnas adicionales o diferentes para las referencias cruzadas.

Configuración de la lista de referencias cruzadas

Las siguientes opciones pueden configurarse mediante los botones de la barra de herramientas de la lista de referencias cruzadas:

- Actualizar la lista de referencias cruzadas
  Actualiza la lista actual de referencias cruzadas.
- Configurar la lista de referencias cruzadas
  Haciendo clic en las casillas de verificación correspondientes, aquí se define si deben mostrarse todos los objetos utilizados, todos los no utilizados, todos los definidos o todos los no definidos. Si está activada la opción "Mostrar las no definidas", también se mostrarán las referencias a objetos ya borrados.
- Reducir entradas
  Reduce las entradas de la lista actual de referencias cruzadas cerrando los objetos subordinados.
- Ampliar entradas
  Amplía las entradas de la lista actual de referencias cruzadas abriendo los objetos subordinados.

Clasificar en la lista de referencias cruzadas

Las entradas de la columna "Objeto" y de otras columnas que varían en función de los productos instalados pueden clasificarse en orden ascendente o descendente. Haga clic para ello en el título de la columna que desee clasificar.

Consulte también

Información general sobre las referencias cruzadas (Página 3080)

Visualizar la lista de referencias cruzadas (Página 3083)

9.3.3 Visualizar la lista de referencias cruzadas

Requisitos

Se ha creado un proyecto.

Introducción

Existen diferentes posibilidades para visualizar referencias cruzadas en función de si está activada la vista del portal o la vista del proyecto, así como del objeto seleccionado en el árbol del proyecto.

En la vista del portal sólo es posible visualizar referencias cruzadas para toda la CPU, mientras que en la vista del proyecto se pueden ver, p. ej., las referencias cruzadas de los siguientes objetos:

- Carpeta "Variables de PLC"
- Carpeta "Tipos de datos PLC"
- Carpeta "Bloques de programa"
- Carpeta "Variables y conexiones"
- Variables individuales
- Tipos de datos PLC individuales
- Bloques individuales
- Objetos tecnológicos
- Tablas de observación

Mostrar referencias cruzadas

Para visualizar las referencias cruzadas, proceda del siguiente modo:

1. Seleccione la acción deseada en la vista del portal, como p. ej. "Programar PLC", y elija el comando "Mostrar referencias cruzadas" o seleccione en la vista del proyecto uno de los objetos indicados arriba y elija luego el comando "Referencias cruzadas" del menú "Herramientas".
   Se visualizará la lista de referencias cruzadas.
2. Haga clic en el botón "Utilizado por" para ver en qué ubicaciones se utilizan los objetos visualizados en la lista de referencias cruzadas.
3. Haga clic en el botón "Utilizado" para ver los usuarios de los objetos visualizados en la lista de referencias cruzadas.

4. Los botones de la barra de herramientas permiten realizar las siguientes acciones:
   - Actualizar la lista de referencias cruzadas
   - Configurar la lista de referencias cruzadas
   - Reducir entradas
   - Ampliar entradas
5. Para clasificar las entradas de las columnas "Objeto" y "Dirección" en orden ascendente o descendente, haga clic en el título de la columna correspondiente.
6. Para acceder a la ubicación del objeto en cuestión, haga clic en el enlace visualizado.

Consulte también

Información general sobre las referencias cruzadas (Página 3080)

Estructura de la lista de referencias cruzadas (Página 3081)

9.3.4 Visualizar las referencias cruzadas en la ventana de inspección

Introducción

En la ventana de inspección se muestra la información correspondiente a las referencias cruzadas de un objeto seleccionado en la ficha "Información > Referencias cruzadas". En esta ficha se muestra en qué ubicaciones y por qué otros objetos es utilizado cada objeto seleccionado.

En la ventana de inspección también se muestran en las referencias cruzadas los bloques que sólo están disponibles online.

El comando "Mostrar accesos solapados..." del menú contextual permite visualizar también accesos solapados más allá de los límites del bloque para objetos seleccionados.

Estructura

En la ventana de inspección, la información de las referencias cruzadas se muestra en tablas. Cada columna contiene información detallada determinada sobre el objeto seleccionado y su utilización. La tabla siguiente muestra la información detallada que se visualiza en la ficha "Información > Referencia cruzada":

Columna	Significado
Objeto	Nombre del objeto que utiliza los objetos subordinados o que es utilizado por éstos.
Número	Número de ubicaciones
Ubicación	Ubicación correspondiente, p. ej. segmento
Propiedad	Propiedades especiales de los objetos referenciados, p. ej. el nombre de las variables en declaraciones multiinstancia.

Columna	Significado
como	Muestra información adicional sobre el objeto, p. ej. el hecho de que un DB instancia se utiliza como plantilla o multiinstancia.
Acceso	Tipo de acceso Indica si se trata de un acceso de lectura (R) o escritura (W) al operando.
Dirección	Dirección del operando
Valor de observación	Esta columna aparece únicamente cuando está abierto el editor de programas.
Tipo	Información sobre el tipo y el lenguaje de creación del objeto
Ruta	Ruta del objeto en el árbol del proyecto

Dependiendo de los productos instalados, se muestran columnas adicionales o diferentes para las referencias cruzadas.

9.4 Comprobar el programa de usuario

9.4.1 Principios básicos del test del programa de usuario

Función

La ejecución del programa de usuario se puede ^obar en el dispositivo. Se pueden observar estados lógicos y valores de variables así como predefinir valores para variables con objeto de simular situaciones determinadas en el desarrollo del programa.

Requisitos

Hay un programa ejecutable cargado en el dispositivo.

Posibilidades de test

Existen las siguientes posibilidades de test:

- Test con el estado del programa
  El estado del programa permite observar su ejecución. Es posible visualizar los valores de los operandos y los resultados lógicos (RLO) con objeto de encontrar y solucionar los errores lógicos del programa.
- Test en modo de etapa individual (sólo S7-300/400)
  Los bloques creados en AWL o SCL pueden probarse en el modo de etapa individual. Para ello, posicione puntos de parada en el código del programa en los que se detendrá la ejecución del programa. A continuación puede seguir ejecutando el programa etapa por etapa. En de una CPU puede efectuar el test, o bien con el estado del programa o bien en modo de etapa individual. Observe que en de una CPU no se pueden aplicar simultáneamente ambas posibilidades de test.

- Test con la tabla de observación
  La tabla de observación permite observar y forzar los valores actuales de las distintas variables del programa de usuario o de una CPU. Para el test es posible predeterminar valores para las distintas variables y simular así diferentes situaciones de procesamiento. Además, existe la posibilidad de asignar valores fijos a las salidas de periferia de una CPU en estado operativo STOP, p. ej. para comprobar el cableado.
- Test con la tabla de forzado permanente
  La tabla de forzado permanente permite observar y forzar permanentemente los valores actuales de las distintas variables del programa de usuario o de una CPU. Al forzar permanentemente se sobrescriben distintas variables con valores predefinidos. De esta manera puede probar el programa de usuario y simular distintas situaciones de procesamiento.
  Antes de forzar permanentemente observe especialmente las medidas de seguridad necesarias para el forzado permanente (Página 3162).

Consulte también

Introducción al test con el estado del programa (Página 3086)

Introducción al test con la tabla de observación (Página 3115)

Introducción al test con la tabla de forzado permanente (Página 3143)

9.4.2 Test con el estado del programa

9.4.2.1 Introducción al test con el estado del programa

Función Estado del programa

Visualizando el estado del programa es posible observar su ejecución. De este modo se obtiene una vista general de los valores de los distintos operandos y de los resultados lógicos y además se puede comprobar si los componentes del sistema de automatización están siendo controlados correctamente.

La visualización del estado de ejecución del programa puede variar levemente en función de la familia de CPU utilizada.

Test con el estado del programa para S7-300/400

Si se efectúa un test con el estado del programa puede prolongarse el tiempo de ciclo de la CPU, ya que, debido a su capacidad, es posible que la lectura de todos los datos necesarios para el test difiera de la duración de las instrucciones programadas y, por tanto, no se ejecute en tiempo real.

Al ejecutar las siguientes funciones de test, para cada sesión online se visualiza una sola vez un aviso que advierte al usuario que se corre el riesgo de rebasar por exceso el tiempo de ciclo:

- Al realizar tests con condiciones de llamada
- Al realizar tests con puntos de parada

Estas funciones de test solo se pueden ejecutar después de confirmar el aviso.

Nota

En CPU más antiguas pertenecientes a las familias S7-300/400 es necesario cambiar las cualidades de funcionamiento en la configuración hardware y luego cargar ésta en el dispositivo. Existe la posibilidad de ajustar las cualidades de funcionamiento "Modo Proceso" o "Modo Test".

Restricciones de la función "Estado del programa"

La visualización de bucles puede aumentar considerablemente el tiempo de ciclo, dependiendo del número de variables que se deban visualizar y del número real de ejecuciones del bucle.

Para garantizar que la influencia que se ejerce sobre el tiempo de ciclo sea lo más reducida posible, la función "Estado del programa" presenta las restricciones siguientes:

- La visualización del estado de un bucle programado se interrumpe en el punto de retorno.

⚠ ADVERTENCIA

Test con el estado del programa

Un test con la función "Estado del programa" puede causar graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

Antes de realizar el test con la función "Estado del programa", asegúrese de que no pueden presentarse estados peligrosos.

9.4.2.2 Ajustar el entorno de llamada

Principios básicos del entorno de llamada

Función

Es posible definir el entorno de llamada para bloques y puntos de parada. Con ello se define en qué condiciones se registra el estado del programa de un bloque o se interrumpe la

ejecución del programa en un punto de parada. Puede seleccionar una de las condiciones siguientes:

- No se ha definido ninguna condición
  Esta es la opción predeterminada si no se ha seleccionado ninguna otra opción.
- Bloque de datos de instancia
  El estado del programa de un bloque de función se registra únicamente si dicho bloque se llama con el bloque de datos de instancia seleccionado.
- Ruta de llamada
  El estado del programa de un bloque se registra únicamente si dicho bloque se llama con un bloque determinado o desde una ruta determinada.

Puede definirse una condición individual para cada punto de parada. Si no se define el entorno de llamada, se registra el estado del programa de una llamada de bloque cualquiera dentro de la estructura de llamada y se interrumpe la ejecución del programa siempre en el punto de parada respectivo. Por ello, defina siempre el entorno de llamada si desea visualizar el estado del programa para una llamada determinada.

Consulte también

Ajustar el entorno de llamada del bloque (Página 3088)

Ajustar el entorno de llamada del bloque

Ajustando el entorno de llamada se puede definir cuándo debe registrarse el estado del programa de un bloque.

Encontrará instrucciones sobre el ajuste del entorno de llamada para puntos de parada en el capítulo "Ajustar el entorno de llamada para puntos de parada (Página 3091)".

Requisitos

- El bloque está abierto.

Procedimiento

Para definir el entorno de llamada, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. Haga clic en el botón "Modificar" de la paleta "Entorno de llamada".
   Se abre el cuadro de diálogo "Entorno de llamada de un bloque".
3. Seleccione la condición deseada.
   Consulte también: Principios básicos del entorno de llamada (Página 3087)
4. Confirme la selección con "Aceptar".

Consulte también

Introducción al test con el estado del programa (Página 3086)

9.4.2.3 Test en modo de etapa individual

Principios básicos del test en modo de etapa individual

Introducción

Los bloques creados en uno de los lenguajes de programación textuales AWL o SCL pueden probarse en el modo de etapa individual. Para ello, ajuste primero puntos de parada en el código del programa, establezca después una conexión online y active los puntos de parada en la CPU. Finalmente se puede ejecutar el programa hasta el primer punto de parada y desde allí se puede seguir ejecutando etapa por etapa con las siguientes funciones:

- Ejecutar
  Continuar hasta el siguiente punto de parada activo.
- Ejecutar hasta la marca
  Continuar hasta la posición del cursor.
- Saltar
  Se ejecuta la instrucción marcada actualmente.
- Entrar
  Salto a un bloque subordinado o bien llamada de un bloque subordinado.
- Salir
  Cuando se está procesando una llamada de bloque y la ejecución del programa en el bloque llamado se interrumpe en un punto de parada, existe la posibilidad de retornar al bloque que se está llamando con "Salir".

Se pueden ajustar tantos puntos de parada en el código del programa como se desee. No obstante, el número de puntos de parada que se pueden activar en la CPU depende de la CPU. A través del entorno de llamada se puede definir para los distintos puntos de parada cuándo debe interrumpirse la ejecución del programa en los puntos de parada.

Si deshace la conexión online con la CPU, se desactivan los puntos de parada.

Nota

Los puntos de parada no se guardan junto con el proyecto. Si se cierra el proyecto, se pierden todos los puntos de parada.

Consulte también

Posicionar puntos de parada (Página 3090)

Ajustar el entorno de llamada para puntos de parada (Página 3091)

Activar y desactivar puntos de parada (Página 3092)

Ejecutar test en modo de etapa individual (Página 3094)

Borrar puntos de parada (Página 3096)

Posicionar puntos de parada

Requisitos

El bloque ha sido creado en AWL o SCL.

Procedimiento

Para posicionar un punto de parada en el código del programa, proceda del siguiente modo:

1. Abra el bloque en el que desea posicionar un punto de parada.
2. Haga clic con el botón derecho del ratón en la columna gris antepuesta al número de la fila en la que desea posicionar el punto de parada.
3. Elija el comando "Posicionar punto de parada" del menú contextual.

O bien:

1. Abra el bloque en el que desea posicionar un punto de parada.
2. Haga doble clic en la columna gris antepuesta al número de la fila en que desea posicionar un punto de parada.

Resultado

En el código del programa se posiciona un punto de parada. El punto de parada se visualiza en la Task Card "Test" de la paleta "Puntos de parada" y adicionalmente en forma de círculo gris delante de la fila correspondiente en el código del programa. Para que la ejecución del programa se interrumpa en el nuevo punto de parada, éste debe activarse previamente.

Consulte también

Principios básicos del test en modo de etapa individual (Página 3089)

Ajustar el entorno de llamada para puntos de parada (Página 3091)

Activar y desactivar puntos de parada (Página 3092)

Ejecutar test en modo de etapa individual (Página 3094)

Borrar puntos de parada (Página 3096)

Navegar hasta los puntos de parada

Para navegar en el programa hasta los puntos de parada, se puede saltar de un punto de parada al siguiente o bien saltar directamente al punto del programa que contiene un punto de parada determinado.

Navegar de un punto de parada al siguiente

Para navegar en el programa de un punto de parada al siguiente, proceda del siguiente modo:

1. Abra un bloque y seleccione el editor de programas.
2. Elija el comando "Siguiente" en el menú "Online > Puntos de parada" para saltar al siguiente punto de parada situado después de la posición actual del cursor. Después del último punto de parada se salta al primero.
3. Elija el comando "Anterior" en el menú "Online > Puntos de parada" para saltar al anterior punto de parada situado antes de la posición actual del cursor. Después del primer punto de parada se salta al último.

Navegar hasta un punto de parada determinado

Para navegar en el programa hasta un punto de parada determinado, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Haga doble clic en el punto de parada hasta el que desea navegar.
   El bloque con el punto de parada se visualiza en el editor de programas y la fila con el punto de parada se resalta.

O bien:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Haga clic con la tecla derecha del ratón en el punto de parada hasta el que desea navegar.
3. Elija el comando "Ir a punto de parada" del menú contextual.
   El bloque con el punto de parada se visualiza en el editor de programas y la fila con el punto de parada se resalta.

Ajustar el entorno de llamada para puntos de parada

Ajustando el entorno de llamada se puede definir cuándo debe interrumpirse la ejecución del programa en los puntos de parada.

Requisitos

En el programa existen puntos de parada.

Procedimiento

Para ajustar el entorno de llamada para un punto de parada, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Puntos de parada" haga clic en el botón "..." situado detrás del punto de parada para el que desea ajustar el entorno de llamada.
   Se abre el cuadro de diálogo "Entorno de llamada de un bloque".
3. Seleccione la condición deseada.
   Consulte también: Principios básicos del entorno de llamada (Página 3087)
4. Confirme la selección con "Aceptar".
   Se establece el entorno de llamada para el punto de parada y el símbolo correspondiente cambia de un simple círculo gris a un círculo gris con rayas verticales delante y detrás.

Consulte también

Principios básicos del test en modo de etapa individual (Página 3089)

Posicionar puntos de parada (Página 3090)

Activar y desactivar puntos de parada (Página 3092)

Ejecutar test en modo de etapa individual (Página 3094)

Borrar puntos de parada (Página 3096)

Activar y desactivar puntos de parada

Para utilizar puntos de parada durante un test, deben activarse los puntos de parada ajustados. Con la activación los puntos de parada son instalados y activados en la CPU. Los puntos de parada pueden activarse y desactivarse tanto individual como colectivamente. Al desactivar puntos de parada, éstos ya no se tienen en cuenta en la ejecución del programa.

Nota

El número de puntos de parada que se pueden activar en una CPU depende de la CPU. Una vez alcanzado en número máximo, no se puede activar ningún punto de parada. Dado que las funciones de etapa individual "Saltar", "Entrar" y "Ejecutar hasta la marca" también son consideradas puntos de parada, tampoco se pueden utilizar. Para poder activar más puntos de parada o utilizar estas funciones de etapa individual, primero es necesario desactivar los puntos de parada activados.

Requisitos

- Se han posicionado puntos de parada.
- Existe una conexión online con la CPU.
- La CPU utilizada soporta el uso de puntos de parada.
- No se ha alcanzado el número máximo de puntos de parada activables en la CPU.

Activar puntos de parada individuales

Para activar puntos de parada individuales, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Haga clic con el botón derecho del ratón en el punto de parada que desea activar.
3. Elija el comando "Activar punto de parada" del menú contextual.
   El punto de parada seleccionado se instala y activa en la CPU y la ejecución del programa se interrumpe en el punto de parada tan pronto como se llegue al punto correspondiente del programa. El punto de parada ya no se representa con un círculo gris, sino con un círculo rojo.

Activar todos los puntos de parada

Para activar todos los puntos de parada, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Observe que ningún punto de parada esté seleccionado.
3. Haga clic en "Activar/desactivar puntos de parada" en la barra de herramientas de la paleta "Puntos de parada".
   Dependiendo de la capacidad de la CPU se instalan y activan todos los puntos de parada en la CPU y la ejecución del programa se interrumpe en todos los puntos de parada que se alcancen durante la ejecución del programa. Si se posicionan más puntos de parada de los permitidos por la CPU, únicamente se activa el número de puntos de parada permitido. Los puntos de parada activados ya no se representan con círculos grises, sino con círculos rojos.

Desactivar puntos de parada individuales

Para desactivar puntos de parada individuales, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Haga clic con el botón derecho del ratón en el punto de parada que desea activar.
3. Elija el comando "Desactivar punto de parada" del menú contextual.
   El punto de parada seleccionado se desinstala y desactiva en la CPU y la ejecución del programa ya no se interrumpe en el punto de parada. El punto de parada ya no se representa con un círculo rojo, sino con un círculo gris.

Desactivar todos los puntos de parada

Para desactivar todos los puntos de parada, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Observe que ningún punto de parada esté seleccionado.
3. Haga clic en "Activar/desactivar puntos de parada" en la barra de herramientas de la paleta "Puntos de parada".
   Todos los puntos de parada seleccionados se desinstalan y desactivan en la CPU y la ejecución del programa ya no se interrumpe en ningún punto de parada. Los puntos de parada ya no se representan con círculos rojos, sino con círculos grises.

Consulte también

Principios básicos del test en modo de etapa individual (Página 3089)

Posicionar puntos de parada (Página 3090)

Ajustar el entorno de llamada para puntos de parada (Página 3091)

Ejecutar test en modo de etapa individual (Página 3094)

Borrar puntos de parada (Página 3096)

Ejecutar test en modo de etapa individual

Si la ejecución del programa se interrumpe en un punto de parada, el programa se puede seguir ejecutando etapa por etapa.

Requisitos

- En el código del programa hay puntos de parada posicionados y activados.
- No se ha alcanzado el número máximo de puntos de parada activables en la CPU.
- El bloque está abierto.

Procedimiento

Para efectuar la ejecución del programa en modo de etapa individual, proceda del siguiente modo:

1. Establezca una conexión online con la CPU.
2. Active los puntos de parada.
   El código del programa se ejecuta hasta el punto de parada activado y luego se interrumpe, es decir, la CPU pasa al estado operativo "PARADA". Una flecha amarilla aparece superpuesta al círculo rojo del punto de parada.

3. Si desea activar las salidas para las funciones "Ejecutar" y "Ejecutar hasta la marca" cuando la CPU pase nuevamente al estado operativo "RUN", active la casilla de verificación "Desbloquear salidas en RUN" en la paleta "Puntos de parada" de la Task Card "Test".
4. Para seguir ejecutando el programa etapa por etapa, seleccione en la barra de herramientas uno de los comandos siguientes:
   - Ejecutar
     La ejecución del programa continúa. Si existen más puntos de parada, la ejecución del programa se interrumpe nuevamente en el siguiente punto de parada.
   - Ejecución hasta la marca
     El programa se sigue ejecutando hasta la posición en que se encuentra el punto de inserción. Si otro punto de parada se encuentra antes del punto de inserción, la ejecución del programa se interrumpe primero en este punto de parada.
   - Saltar
     Se ejecuta la siguiente instrucción después del punto de parada. Si la instrucción es una llamada de bloque, la llamada se procesa y se salta hasta la siguiente instrucción después de la llamada de bloque.
   - Entrar
     Se ejecuta la siguiente instrucción después del punto de parada. Si la instrucción es una llamada de bloque, se salta al bloque invocado. Allí se puede continuar el test en modo de etapa individual o bien posicionar otros puntos de parada. Al final del bloque tiene lugar un retorno a la siguiente instrucción después de la llamada de bloque.
   - Salir
     Cuando se está procesando una llamada de bloque y la ejecución del programa en el bloque llamado se interrumpe en un punto de parada, existe la posibilidad de retornar al bloque que se está llamando con "Salir". En dicho caso se ejecutan las instrucciones restantes del bloque llamado y la ejecución del programa se vuelve a detener en el bloque que se está llamando al finalizar la llamada del mismo.

Consulte también

Principios básicos del test en modo de etapa individual (Página 3089)

Posicionar puntos de parada (Página 3090)

Ajustar el entorno de llamada para puntos de parada (Página 3091)

Activar y desactivar puntos de parada (Página 3092)

Borrar puntos de parada (Página 3096)

Introducción al test con el estado del programa (Página 3086)

Finalizar el test en modo de etapa individual

Procedimiento

Para finalizar la ejecución del programa en modo de etapa individual, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. Desactive los puntos de parada de la CPU.
3. Active la casilla de verificación "Desbloquear salidas en RUN".
4. Haga clic con el botón derecho del ratón en la columna gris antepuesta a un número de fila.
5. En el menú contextual, elija el comando "Ejecutar".

Resultado

La CPU pasa a RUN y continúa la ejecución del programa.

Consulte también

Introducción al test con el estado del programa (Página 3086)

Borrar puntos de parada

Los puntos de parada pueden borrarse tanto individual como colectivamente.

Borrar puntos de parada individuales

Para borrar del programa un solo punto de parada, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Seleccione el punto de parada que desea borrar.
3. Haga clic con el botón derecho del ratón en el punto de parada seleccionado.
4. Elija el comando "Borrar punto de parada" del menú contextual.

O bien:

1. Abra el bloque del que desea borrar un punto de parada.
2. Haga doble clic en la columna gris antepuesta al número de la fila del punto de parada que desea borrar.

Borrar todos los puntos de parada

Para borrar todos los puntos de parada, proceda del siguiente modo:

1. Abra la Task Card "Test".
   En la paleta "Puntos de parada" aparecen listados todos los puntos de parada posicionados en la CPU actual.
2. Observe que ningún punto de parada esté seleccionado.
3. Haga clic en "Eliminar puntos de parada" en la barra de herramientas de la paleta "Puntos de parada".

Consulte también

Principios básicos del test en modo de etapa individual (Página 3089)

Posicionar puntos de parada (Página 3090)

Ajustar el entorno de llamada para puntos de parada (Página 3091)

Activar y desactivar puntos de parada (Página 3092)

Ejecutar test en modo de etapa individual (Página 3094)

9.4.2.4 Activar/desactivar el test con el estado del programa

Se pueden observar todos los bloques activando el estado del programa del bloque. Esta función está disponible para todos los bloques lógicos, independientemente del lenguaje de programación utilizado. Para los bloques programados en KOP y FUP se puede activar el estado del programa adicionalmente a partir de una posición determinada o para una selección determinada. El estado del programa de un bloque abierto se puede activar directamente, o bien puede abrirse un bloque desde el bloque que efectúa la llamada y visualizar el estado del programa.

Nota

Los recursos para el test con el estado del programa son limitados. Si los recursos para el test actual no son suficientes, se finalizarán los tests anteriores.

Requisitos

En el dispositivo existe un bloque idéntico.

Activar o desactivar el estado del programa directamente en el bloque

Para activar o desactivar el estado del programa directamente en un bloque, proceda del siguiente modo:

1. Abra el bloque del que desee activar el estado del programa.
2. Haga clic en el botón "Activar/desactivar observación" de la barra de herramientas.
   Si hasta ahora no se ha establecido ninguna conexión online, aparecerá el cuadro de diálogo "Establecer conexión online". Con este cuadro de diálogo podrá establecer una conexión online.
   Consulte también: Establecer y deshacer una conexión online
3. Para desactivar el estado del programa, vuelva a hacer clic en el botón Activar/desactivar observación".

Activar o desactivar el estado del programa a partir de un punto determinado de un segmento

Para activar el estado del programa para KOP y FUP a partir de un punto determinado, proceda del siguiente modo:

1. Abra el bloque del que desee activar el estado del programa.
2. Haga clic en el botón "Activar/desactivar observación" de la barra de herramientas.
3. Haga clic con el botón derecho del ratón en la variable a partir de la cual desea activar el estado del programa.
4. Elija el comando "Forzar > Observar a partir de aquí" del menú contextual.
5. Para desactivar el estado del programa, vuelva a hacer clic en el botón Activar/desactivar observación".

Activar o desactivar el estado del programa para variables seleccionadas

Para activar el estado del programa para KOP y FUP para variables seleccionadas, proceda del siguiente modo:

1. Abra el bloque del que desee activar el estado del programa.
2. Haga clic en el botón "Activar/desactivar observación" de la barra de herramientas.
3. Seleccione las variables para las que desea activar el estado del programa.
4. Elija el comando "Forzar > Observar selección" del menú contextual.
5. Para desactivar el estado del programa, vuelva a hacer clic en el botón Activar/desactivar observación".

Activar el estado del programa desde el bloque que efectúa la llamada

Para activar el estado del programa de un bloque desde el bloque que efectúa la llamada (p. ej. OB1), proceda del siguiente modo:

1. Abra el bloque que efectúa la llamada.
2. Haga clic en la llamada del bloque con el botón derecho del ratón.
3. En el menú contextual, elija el comando "Abrir y observar".
   El bloque se abre en el editor de programación. A continuación se establece la conexión online y se muestra el estado del programa.

Nota

Si el bloque que efectúa la llamada está programado en SCL y el bloque llamado es un FB, el FB se abre con "Abrir y observar". Para observar el FB, es necesario activar seguidamente el estado del programa directamente en el FB.

Resultado

Cuando active la visualización del estado del programa, se establecerá una conexión online y se visualizará el estado del programa. Si desactiva la visualización del estado del programa, es posible deshacer simultáneamente la conexión online.

Bajo la interfaz del bloque se muestra la ruta de llamada del bloque. Para las CPU de las series S7-300/400 y S7-1500 puede modificarse si es necesario la condición de llamada con el botón situado en el margen derecho.

Consulte también

Ajustar el entorno de llamada (Página 3087)

9.4.2.5 Editar bloques durante el test del programa

Si se editan bloques durante un test con el estado del programa, se interrumpirá la observación online y se podrá editar el bloque offline. Si el bloque no existe offline en el proyecto, habrá que cargarlo primero en el proyecto desde el dispositivo. Una vez finalizada la edición es necesario volver a compilar y cargar el bloque.

Procedimiento

Para editar bloques durante el test con el estado del programa, proceda del siguiente modo:

1. Edite el bloque según las exigencias requeridas.
   El test con el estado del programa se interrumpe y el bloque se conmuta a modo offline (si ya existe offline).
2. Si el bloque no existe offline, cárguelo en el proyecto desde el dispositivo.

3. Compile el bloque.
   Consulte también: Compilar bloques (Página 3012)
4. Cargue el bloque en el dispositivo.
   Consulte también: Cargar bloques (Página 3018)

Resultado

Las versiones online y offline del bloque contienen los cambios efectuados. La conexión online se puede restablecer y continuar el test con el estado del programa.

9.4.2.6 Forzar variables en el estado del programa

Durante el test con el estado del programa, las variables se pueden forzar inmediatamente y una vez a los valores siguientes:

- Forzar a 1
  Fuerza variables del tipo de datos "Bool" al valor "True".
- Forzar a 0
  Fuerza variables del tipo de datos "Bool" al valor "False".
- Forzar operando
  Es posible introducir un valor de forzado para las variables que no sean del tipo de datos "Bool".

Sin embargo, tenga en cuenta que las entradas de periferia no se pueden forzar, por ejemplo, con TagName:P.

Insertar procedimiento

Para forzar variables durante el test con el estado del programa, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la variable que desea forzar.
2. Elija uno de los comandos siguientes del menú contextual:
   – "Forzar > Forzar a 1"
   – "Forzar > Forzar a 0"
   – "Forzar > Forzar operando"
3. Si selecciona "Forzar operando", aparecerá el diálogo "Forzar operando". Introduzca el valor deseado en el campo "Valor de forzado" y confirme la entrada con "Aceptar".

9.4.2.7 Cambiar formatos de visualización en el estado del programa

Introducción

Los formatos de visualización para variables se muestran habitualmente en "enteros". En el estado del programa es posible cambiar el formato de visualización actual mediante el menú contextual. Los formatos de visualización que son posibles para una variable se ofrecen en

una lista. Sirve de ayuda, p. ej. si se requiere una representación hexadecimal, para buscar un código de error hexadecimal.

Procedimiento

Para cambiar el formato de visualización, proceda del modo siguiente:

1. Abra el bloque deseado en el editor de programación.
2. Active el estado del programa haciendo clic en el botón "Activar/desactivar observación" de la barra de herramientas.
   Si aún no se ha establecido ninguna conexión online, aparecerá el cuadro de diálogo "Establecer conexión online". Con este cuadro de diálogo podrá establecer una conexión online.
3. Seleccione las variables para las que desea activar el estado del programa.
4. Para iniciar la observación de la variable, elija el comando "Forzar > Observar selección" del menú contextual.
5. Seleccione la variable deseada en la salida de bloque correspondiente y elija el formato de visualización deseado en el menú contextual, p. ej. "Forzar > Formato de visualización > Hexadecimal".

Resultado

La variable seleccionada se muestra en formato hexadecimal.

Nota

Cambiar el formato de visualización en el estado del programa

Tenga en cuenta que para salidas no cableadas no es posible cambiar el formato de visualización, puesto que no se emite ningún valor de observación.

9.4.2.8 Particularidades del test de programas GRAPH

Principios básicos del test de programas GRAPH

Introducción

Mostrando el estado de un programa GRAPH es posible probar la lógica, la coherencia y la función de un control secuencial.

Con él se pueden localizar, por ejemplo, los siguientes errores:

- Errores de programa; por ejemplo, desfases entre las etapas y condiciones programadas y la ejecución real del proceso
- Errores de programación en el control secuencial, por ejemplo, tiempos de vigilancia mal definidos en supervisiones

Para poder probar un programa de usuario, debe existir una conexión online con la CPU.

⚠ ADVERTENCIA
Peligro durante el test en caso de instalación en marcha El test del programa con la instalación en marcha puede causar graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa. Antes de realizar el test, asegúrese de que no pueden presentarse estados peligrosos.

Funciones de test de GRAPH

Para probar controles secuenciales creados en GRAPH, se dispone de las siguientes funciones de test:

- Estado del programa para cadenas
- Estado del programa para condiciones y acciones
- Estado del programa para interlocks y supervisión
- Estado del programa para instrucciones permanentes anteriores y posteriores

Durante un test es posible forzar la cadena secuencial y también sincronizarla con el estado actual del proceso.

Modos de operación para el test de programas GRAPH

Para el test de programas GRAPH, se dispone de los tres modos de operación siguientes:

- Modo automático
  En este modo se avanza automáticamente a la siguiente etapa cuando se cumple la transición.
- Modo semiautomático
  En este modo se avanza automáticamente a la siguiente etapa cuando se cumple una de las siguientes condiciones:
  - La transición se cumple.
  - En el parámetro "T_PUSH" hay un flanco de señal ascendente.
  - Se sigue avanzando manualmente.
- Modo manual
  En este modo de operación se puede avanzar de una etapa a la siguiente manualmente o bien seleccionar una etapa concreta.

Sincronización del sistema

Si un proceso se ha llevado a otro estado en modo manual, es posible que ya no esté sincronizado con la cadena secuencial. Para volver a sincronizar el proceso y el programa, se pueden buscar puntos de sincronización y seguidamente realizar la sincronización.

Para localizar los puntos de sincronización se puede elegir uno de los dos métodos siguientes:

- Transición anterior cumplida
  Se marcan todas las etapas cuya transición anterior se ha cumplido.
- Interlock cumplido
  Se marcan todas las etapas cuyos interlocks se han cumplido.

En ambas variantes solamente se marcan las etapas para las que no se ha cumplido la transición siguiente.

Consulte también

Principios básicos del test del programa de usuario (Página 3085)

Introducción al test con el estado del programa (Página 3086)

Resumen de los ajustes de test de GRAPH (Página 3103)

Forzar cadena secuencial (Página 3105)

Acusar errores de supervisión (Página 3105)

Test en diferentes modos de operación (Página 3106)

Ejecutar sincronización del sistema (Página 3107)

Definir ajustes para el test

Resumen de los ajustes de test de GRAPH

Resumen

La siguiente tabla muestra un resumen de los ajustes para comprobar los programas GRAPH:

Ajuste	Descripción
Seguir etapa activa	La etapa que está activa en la cadena secuencial se muestra en la vista de etapa individual o en la vista de cadena.
Saltar etapas	Una etapa cuya transición anterior o siguiente es válida no se activa. El ajuste predeterminado de esta opción se aplicará a la correspondiente configuración de bloque. Esta opción no está disponible si utiliza la opción "Generar DB minimizado" para el bloque de función GRAPH.
Acuse obligatorio en errores de supervisión	Determina que los errores de supervisión deben acusarse antes de seguir ejecutando el programa. Esta opción solo está disponible si en los ajustes del bloque de función GRAPH se ha activado la propiedad de cadena "Acuse obligatorio en errores de supervisión".
Parar cadena	La cadena secuencial se detiene aunque se haya cumplido la siguiente transición.

Ajuste	Descripción
Desactivar temporizadores	Al activarse esta opción, se detienen todos tiempos de activación de etapas. Si desactiva la opción, los tiempos seguirán transcurriendo.
Procesar todos los interlocks	Se procesan todos los interlocks, incluyendo los de las etapas inactivas. Esta opción solo está disponible cuando se cumplen las dos condiciones siguientes: • En los ajustes del bloque de función GRAPH, está activada la propiedad de cadena "Procesamiento permanente de todos los interlocks en modo manual". • La cadena secuencial se encuentra en el modo de operación "Modo manual". Para ello es necesario sincronizar la cadena secuencial con el dispositivo. En la vista de cadena se pueden apreciar los interlocks que se han cumplido.
Procesar todas las transiciones	Siempre se procesan todas las transiciones. En la vista de cadena se pueden apreciar las transiciones que se han cumplido.
Activar acciones	Activa las acciones de la etapa que está activa. Esta opción está activada por defecto. Al desactivar esta opción, no se ejecuta ninguna acción más en la etapa activa.
Activar supervisiones	Activa la supervisión de la etapa que está activa. Esta opción está activada por defecto. Al desactivar esta opción, se ignoran las condiciones de la supervisión en la etapa activa.
Activar interlocks	Activa los interlocks de la etapa que está activa. Esta opción está activada por defecto. Al desactivar esta opción, se ignoran las condiciones del interlock en la etapa activa.

Consulte también

Principios básicos del test de programas GRAPH (Página 3101)

Modificar ajustes de test (Página 3104)

Modificar ajustes de test

⚠ ADVERTENCIA

Modificación de los ajustes de test

Tenga en cuenta que, al cambiar los ajustes de test, pueden producirse repercusiones no deseadas en el proceso. Antes de modificar los ajustes de test, asegúrese de que no pueden presentarse estados peligrosos.

Requisitos

Hay un bloque de función GRAPH abierto.

Procedimiento

Para modificar los ajustes de test de GRAPH, proceda del siguiente modo:

1. Inicie el test con estado de programa para el bloque de función GRAPH.
   Consulte también: Activar/desactivar el test con el estado del programa (Página 3097)
2. Abra la Task Card "Test".
3. Modifique los ajustes de la paleta "Ajustes de test".

Consulte también

Resumen de los ajustes de test de GRAPH (Página 3103)

Forzar una cadena secuencial durante un test

Forzar cadena secuencial

Durante el test con el estado de programa se puede reinicializar la cadena secuencial. De este modo, la cadena secuencial empieza de nuevo por las etapas iniciales. Además, es posible desactivar todas las etapas durante el test.

Requisitos

Se muestra el estado del programa.

Inicializar cadena secuencial

Para inicializar la cadena secuencial, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Control secuencial", haga clic en "Inicializar".
   La cadena secuencial se procesará desde las etapas iniciales.

Desactivar todas las etapas

Para desactivar todas las etapas durante el test de una cadena secuencial, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Control secuencial", haga clic en "Desactivar todos".
   Se desactivarán todas las etapas de la cadena secuencial.

Consulte también

Acusar errores de supervisión (Página 3105)

Test en diferentes modos de operación (Página 3106)

Ejecutar sincronización del sistema (Página 3107)

Principios básicos del test de programas GRAPH (Página 3101)

Introducción al test con el estado del programa (Página 3086)

Activar/desactivar el test con el estado del programa (Página 3097)

Acusar errores de supervisión

En los ajustes de test se puede especificar que los errores de supervisión deban acusarse durante el test, a fin de que el programa pueda continuar.

Requisitos

Se muestra el estado del programa.

Procedimiento

Para acusar errores de supervisión, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Control secuencial", haga clic en "Acusar -(V)-".

Consulte también

Forzar cadena secuencial (Página 3105)
Test en diferentes modos de operación (Página 3106)
Ejecutar sincronización del sistema (Página 3107)
Principios básicos del test de programas GRAPH (Página 3101)
Resumen de los ajustes de test de GRAPH (Página 3103)
Modificar ajustes de test (Página 3104)
Introducción al test con el estado del programa (Página 3086)
Activar/desactivar el test con el estado del programa (Página 3097)

Test en diferentes modos de operación

Requisitos

Se muestra el estado del programa.

Test en modo automático

Para probar una cadena secuencial en el modo automático, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Control secuencial", seleccione la opción "Modo automático".

Test en modo semiautomático

Para probar una cadena secuencial en el modo semiautomático, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Control secuencial", seleccione la opción "Modo semiautomático".
3. Si el avance no se produce automáticamente debido a que se cumple una transición o al parámetro "T_PUSH", haga clic en el botón "Ignorar transición".

Test en modo manual

Para probar una cadena secuencial en el modo manual, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. En la paleta "Control secuencial", seleccione la opción "Modo manual".
3. Avance a otra etapa:
   - Si desea activar la siguiente etapa, haga clic en el botón "Siguiente".
   - Si desea activar o desactivar una etapa cualquiera, indique el número de etapa en el campo "Seleccionar etapa manualmente" o seleccione una etapa en la cadena secuencial y haga clic en el botón "Activar" o "Desactivar".

Consulte también

Forzar cadena secuencial (Página 3105)

Acusar errores de supervisión (Página 3105)

Ejecutar sincronización del sistema (Página 3107)

Activar/desactivar el test con el estado del programa (Página 3097)

Principios básicos del test de programas GRAPH (Página 3101)

Introducción al test con el estado del programa (Página 3086)

Ejecutar sincronización del sistema

Requisitos

- Se muestra el estado del programa.
- El test se realiza en el modo de operación "Modo manual".

Procedimiento

Para ejecutar una sincronización del sistema, proceda del siguiente modo:

1. Abra la Task Card "Test".
2. Active en la paleta "Control secuencial" la casilla de verificación "Activar sincronización".
3. Elija uno de los dos métodos siguientes para encontrar puntos de sincronización:
   - "Transición anterior cumplida"
     Se marcan todas las etapas cuya transición anterior se ha cumplido.
   - "Interlock cumplido"
     Se marcan todas las etapas cuyos interlocks se han cumplido.
4. Seleccione la etapa en la que deberá comenzar la sincronización del sistema.
5. Haga clic en el botón "Activar".

Resultado

El proceso y las cadenas secuenciales se sincronizan entre sí.

Consulte también

Principios básicos del test de programas GRAPH (Página 3101)
Forzar cadena secuencial (Página 3105)
Acusar errores de supervisión (Página 3105)
Test en diferentes modos de operación (Página 3106)
Introducción al test con el estado del programa (Página 3086)
Activar/desactivar el test con el estado del programa (Página 3097)

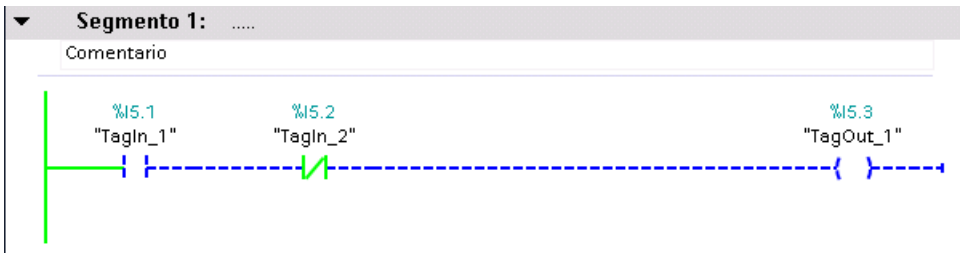
9.4.2.9 Ejemplos de visualización del estado del programa

Visualización del estado de programas KOP

Visualización en el estado del programa

La visualización del estado del programa se actualiza cíclicamente.

La figura siguiente muestra un ejemplo de la visualización del estado del programa en KOP:



Representación del estado del programa

El estado de las distintas instrucciones y líneas de un segmento se reconoce fácilmente por el color y tipo de las líneas y los símbolos. La tabla siguiente muestra cómo se representan los distintos estados:

Representación	Estado
Línea verde continua	Cumplido
Línea azul discontinua	No cumplido
Línea gris continua	Desconocido o no procesado
Negro	No cableado

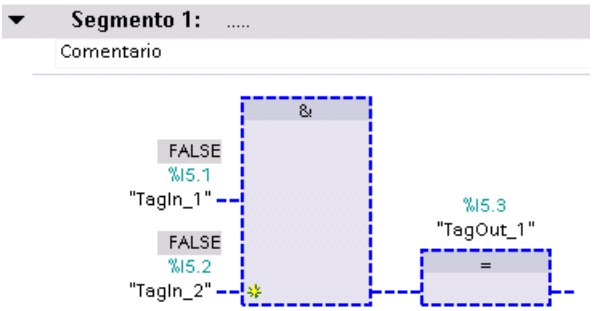
Representación	Estado
Parámetros en un marco con una saturación del 100%	El valor es actual
Parámetros en un marco con una saturación del 50 %	El valor proviene de un ciclo anterior. Este punto del programa no se ha procesado en el ciclo actual.

Visualización del estado de programas FUP

Visualización en el estado del programa

La visualización del estado del programa se actualiza cíclicamente.

La figura siguiente muestra un ejemplo de la visualización del estado del programa en FUP:



Representación del estado del programa

El estado de las distintas instrucciones y líneas de un segmento se reconoce fácilmente por el color y tipo de las líneas y los símbolos. La tabla siguiente muestra cómo se representan los distintos estados:

Representación	Estado
Línea verde continua	Se cumple
Línea azul discontinua	No se cumple
Línea gris continua	Desconocido o no procesado
Negro	No cableado
Parámetros en un marco con una saturación del 100%	El valor es actual
Parámetros en un marco con una saturación del 50 %	El valor proviene de un ciclo anterior. Este punto del programa no se ha procesado en el ciclo actual.

Los valores de los operandos se visualizan en una casilla gris sobre el nombre del operando en cuestión.

Nota

Visualización del estado del programa en el caso de salidas sin cablear

Tenga en cuenta que para las salidas no cableadas no puede emitirse ningún valor de observación.

Visualización del estado de programas AWL

Visualización en el estado del programa

La visualización del estado del programa se actualiza cíclicamente y se muestra en tablas. Las tablas se muestran directamente junto al programa AWL. Es posible leer el estado de cada línea del programa. La visualización depende de la CPU utilizada (S7-300, S7-400, S7-1200 o S7-1500).

Las tablas de visualización en el estado del programa contienen siempre la información siguiente:

- RLO
  En la columna "RLO" se indica el resultado lógico de cada línea del programa. El valor del RLO respectivo puede saberse por el color de fondo de la celda de la tabla. En este sentido, verde significa un RLO de 1 y lila un RLO de 0.
- Valor
  En la columna "Valor" se indica el valor actual del operando.
- Extra
  En la columna "Extra" aparece información adicional en función de la operación en cuestión, p. ej. bits de estado relevantes en caso de instrucciones matemáticas, valores de tiempo o contaje en caso de temporizadores y contadores o las direcciones de memoria en caso de direccionamiento indirecto.

La figura siguiente muestra un ejemplo de la visualización del estado del programa de una CPU S7-300 en AWL:

Segmento 1: .....

Comentario

			RLO	Valor	Extra
1	OPN "DB1"	%DB1	0	DB1	
2	L %DBW2	%DBW2	0	16#4D2	DB1
3	L %DBW4	%DBW4	0	16#3333	DB1
4	+I		0	14341	OS=0,OV=0,A0=0,A1=1
5	T "Tag_15"	%MW4	0	14341	
6					
7	L 3.0	3.0	0	3.0	
8	T "Tag_24"	%MD10	0	3.0	
9	L 5.0	5.0	0	5.0	
10	T "Tag_52"	%MD20	0	5.0	
11	L MD ["Tag_29"]	%MD26	0	16#40400000	P#10.0
12	LAR1 P#M16.0	P#M16.0	0	P#M16.0	
13	L MD [AR1 , P#4.0]	P#4.0	0	16#40A00000	P#M20.0
14	+R		0	8.0	OS=0,OV=0,A0=0,A1=1
15	T "Tag_17"	%MD30	0	8.0	
16					
17	OPN DB ["Tag_48"]	%MW14	0	DB1	
18	L DBW ["Tag_38"]	%MD60	0	16#3333	P#4.0
19	L W#16#7777	W#16#7777	0	16#7777	
20	XOW		0	16#4444	
21	T "Tag_47"	%MW70	0	16#4444	
22					
23	A "Tag_1"	%I0.0	1	1	
24	L S5T#55S	S5T#55S	1	S5T#55S	
25	SE "Tag_49"	%T5	1	S5T#47S600MS	
26	A "Tag_49"	%T5	1	1	S5T#47S600MS
27	= "Tag_3"	%Q0.0	1	1	
28					
29	A "Tag_55"	%I0.2	1	1	
30	CU "Tag_23"	%C1	1	C#556	
31	A "Tag_23"	%C1	1	1	C#556
32	= "Tag_3"	%Q0.0	1	1	
33					
34	A "Tag_31"	%M10.0	0	0	
35	AN M ["Tag_51"]	%MD64	0	0	P#5.1
36	O DBX ["Tag_38"]	%MD60	1	1	P#4.0
37	LAR2 P#M10.0	P#M10.0	1	P#M10.0	
38	X [AR2 , P#3.4]	P#3.4	1	0	P#M13.4
39	= Q ["Tag_51"]	%MD64	1	1	P#5.1
40					

Visualización del estado de programas SCL

Visualización en el estado del programa

La visualización del estado del programa se actualiza cíclicamente y se muestra en una tabla. La tabla se visualiza justo al lado del programa SCL y es posible leer el estado de cada línea del programa. La tabla contiene la siguiente información:

- Nombres de variables
- Valor

Puede desplazar la tabla en todo momento hacia la derecha o hacia la izquierda.

La figura siguiente muestra un ejemplo de la visualización del estado del programa en SCL:

```
1 IF "TagIn_1"                    "TagIn_1"   FALSE
2 THEN "TagIn_2" :=1;             "TagIn_2"   TRUE
3 END_IF;
4 IF "TagIn_2" = false            "TagIn_2"   FALSE
5 THEN "TagIn_3" :=1;             "TagIn_3"   TRUE
6 END_IF;
7
```

En la primera columna aparece el nombre de la variable que corresponde al valor visualizado actualmente. Si la línea contiene la instrucción "IF", "WHILE" o "REPEAT", en la misma se visualiza el resultado de la instrucción con "True" o "False". Si la línea contiene varias variables, se visualiza el valor de la primera variable. En ambos casos se visualizan en una lista adicional todas las variables de estas líneas con sus respectivos valores, tan pronto como seleccione la línea correspondiente. Si dentro del código del programa posiciona el cursor en una variable, ésta será visualizada en negrita en la lista. También se pueden visualizar explícitamente las demás variables de una línea, para ello haga clic en la flecha hacia la derecha que se encuentra antepuesta a los nombres de las variables de la línea que contiene varias variables.

Si el código de la línea no se ejecuta, el nombre de la variable se visualiza en color de fuente gris en la tabla de valores.

En la última columna aparecen los valores actuales de las variables. Si no es posible visualizar valores de una variable, la celda correspondiente recibe un fondo amarillo y aparecen adicionalmente tres signos de interrogación. En este caso active la casilla de verificación "Crear información avanzada de estado" en las propiedades del bloque y cargue nuevamente el bloque en el dispositivo. Finalmente se indican todos los valores.

Visualización del estado de programas GRAPH

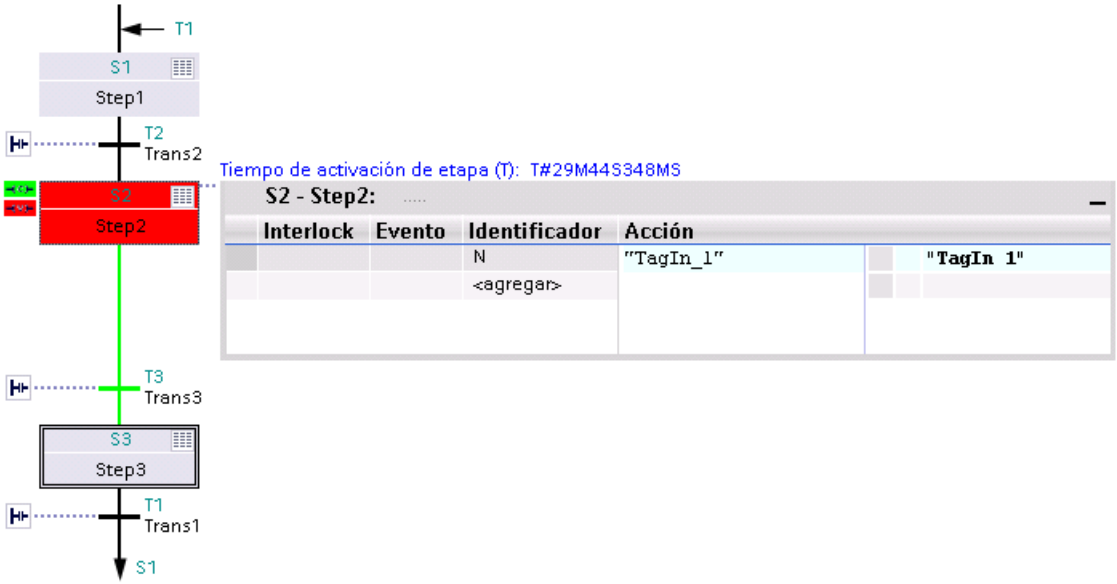
Visualización en el estado del programa

En el caso de los programas GRAPH, el estado del programa se puede visualizar en la vista de cadena y en la vista de etapa individual, así como en las instrucciones permanentes. La visualización del estado del programa de las instrucciones permanentes equivale a la visualización del estado del programa para programas KOP/FUP. La visualización del estado del programa se actualiza cíclicamente.

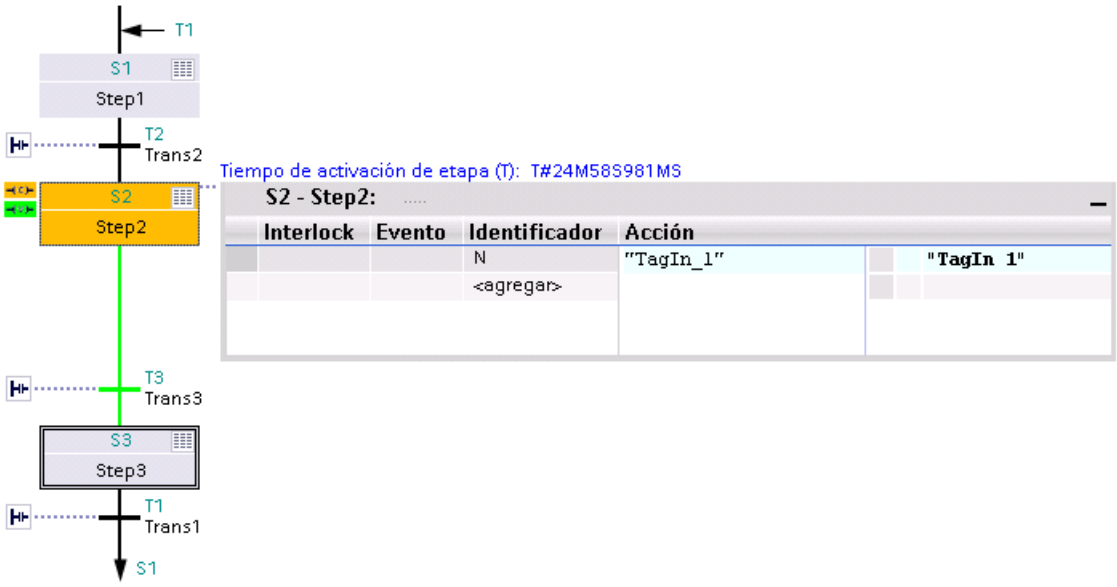
La tabla siguiente muestra cómo se representan los distintos estados:

Representación	Área	Estado
Verde	Etapa, cadena secuencial	No hay ningún fallo.
	Condición	Se cumple la transición.
Rojo	Etapa, cadena secuencial	Hay un error de supervisión.
Amarillo	Etapa, cadena secuencial	Hay un error de interlock.
Negro	Condición	No se cumple la transición.

Las siguientes figuras muestran ejemplos de visualización del estado del programa en la vista de cadena:



En la etapa 2 hay un error de supervisión. Se cumple la transición para pasar a la etapa 3.



En la etapa 2 hay un error de interlock. Se cumple la transición para pasar a la etapa 3.

La siguiente figura muestra un ejemplo de visualización del estado del programa en la vista de etapa individual para una S7-300:

No hay ningún un error de interlock en la etapa. No se cumple la siguiente transición.

9.4.3 Test con la tabla de observación

9.4.3.1 Introducción al test con la tabla de observación

Sinopsis

Dentro de la tabla de observación se dispone de las siguientes funciones:

- **Observar variables**
  Esta función permite visualizar en la programadora o el PC los valores actuales de las distintas variables de un programa de usuario o de una CPU.
- **Forzar variables**
  Esta función permite asignar valores fijos a las distintas variables de un programa de usuario o de una CPU. El forzado también es posible en el test con el estado del programa (Página 3086).
- **"Desbloquear salidas" y "Forzar inmediatamente"**
  Estas dos funciones permiten asignar valores fijos a las distintas salidas de periferia de una CPU en estado operativo STOP. Esta opción también permite comprobar el cableado.

Observar y forzar variables

Las siguientes variables pueden observarse y forzarse:

- entradas, salidas y marcas
- contenidos de bloques de datos
- periferia

Posible utilización

La tabla de observación ofrece la ventaja de que permite guardar diferentes entornos de test. De esta manera es posible reproducir tests durante la puesta en servicio o con fines de servicio y mantenimiento.

Consulte también

Crear y editar tablas de observación (Página 3119)

Estructura de la tabla de observación (Página 3116)

Modo básico/avanzado en la tabla de observación (Página 3117)

Símbolos y botones utilizados en la tabla de observación (Página 3118)

9.4.3.2 Estructura de la tabla de observación

Introducción

Una tabla de observación contiene las variables que se han definido y seleccionado y que son válidas para toda la CPU. Para cada CPU que se haya creado en el proyecto se crea automáticamente una carpeta "Tablas de observación y forzado permanente". Con el comando "Agregar nueva tabla de observación" se crea una nueva tabla de observación en esta carpeta.

Estructura de la tabla de observación

Las columnas visualizadas en la tabla de observación varían en función de si se está trabajando en el modo básico o avanzado.

En el modo avanzado se muestran además las siguientes columnas:

- Observar con disparador
- Forzar con disparador

Además, los nombres de las columnas pueden cambiar dinámicamente según la acción realizada.

Significado de las columnas

La tabla siguiente muestra el significado de las diferentes columnas en el modo básico y en el modo avanzado:

Modo	Columna	Explicación
Modo básico	i	Columna identificativa
	Nombre	Nombre de la variable insertada
	Dirección	Dirección de la variable insertada
	Formato de visualización	Formato de visualización seleccionado
	Valor de observación	Valores de las variables en función del formato de visualización seleccionado.
	Valor de forzado	Valor con el que se fuerza la variable.
		Selección de las variables que se van a forzar activando la casilla de verificación correspondiente.
	Comentario	Comentario para la documentación de las variables
En el modo avanzado se muestran también las siguientes columnas:	Observar con disparador	Indicación del modo de observación seleccionado
	Forzar con disparador	Indicación del modo de forzado seleccionado

Consulte también

Símbolos y botones utilizados en la tabla de observación (Página 3118)

9.4.3.3 Modo básico/avanzado en la tabla de observación

Diferencia entre el modo básico y el modo avanzado en la tabla de observación

Dependiendo del modo ajustado, en la tabla de observación se muestran diferentes columnas y títulos de columna que permiten realizar distintas acciones.

Encontrará una relación detallada de las diferentes columnas en Estructura de la tabla de observación (Página 3116).

Conmutar entre el modo básico y el modo avanzado

Para conmutar entre el modo básico y el avanzado existen las siguientes opciones:

- Haga clic en el botón "Ocultar o mostrar todas las columnas del modo avanzado". Haciendo clic de nuevo en el mismo botón se regresa al modo básico.
  O bien:
- Active la casilla de verificación antepuesta al comando "Modo avanzado" del menú "Online". Desactivando la casilla de verificación se regresa al modo básico.

Funciones del modo avanzado

Las siguientes funciones sólo están disponibles en el modo avanzado:

- Observar con disparador
- Forzar con disparador
- Desbloquear salidas
- Observar entradas de periferia
- Forzar salidas de periferia

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al observar entradas de periferia y forzar salidas de periferia Tenga en cuenta que al observar entradas de periferia y forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Consulte también

Definición del modo de observación y forzado (Página 3131)

9.4.3.4 Símbolos y botones utilizados en la tabla de observación

Significado de los símbolos

La tabla siguiente muestra el significado de los símbolos de la tabla de observación:

Símbolo/Botón	Significado
	Marca una tabla de la navegación del proyecto como tabla de observación.
	Muestra información en la columna de identificación.
	Inserta una fila delante de la fila seleccionada.
	Inserta una fila detrás de la fila seleccionada.
	Fuerza una sola vez y de forma inmediata las direcciones de todas las variables seleccionadas. Este comando se ejecuta una sola vez y lo más rápido posible, sin referencia a un punto de disparo determinado del programa de usuario.
	Fuerza las direcciones de todas las variables seleccionadas con referencia a un punto de disparo determinado del programa de usuario.
	Desactiva el bloqueo de salida de comandos de las salidas de periferia (BASP). Esto permite forzar las salidas de periferia en estado STOP de la CPU.
	Muestra todas las columnas del modo avanzado. Haciendo clic de nuevo en este botón, se ocultan las columnas del modo avanzado.
	Muestra todas las columnas de forzado. Haciendo clic de nuevo en el botón, se ocultan las columnas de forzado.
	Inicia la observación de las variables visibles en la tabla de observación activa. En el modo básico, para el modo de observación está predeterminada la opción "permanente". En el modo avanzado se pueden configurar puntos de disparo determinados para la observación de las variables.
	Inicia la observación de las variables visibles en la tabla de observación activa. Este comando se ejecuta inmediatamente y las variables se observan una sola vez.
	Muestra la casilla de verificación para la selección de las variables que deben forzarse.
	Indica que la variable seleccionada se ha forzado al valor "1".
	Indica que la variable seleccionada se ha forzado al valor "0".
	Indica que la dirección se utiliza de forma múltiple.
	Indica que se utiliza un valor sustitutivo. Los valores sustitutivos son valores que se transmiten al proceso (en el caso de los módulos de señales de salidas defectuosos) o que se utilizan en el programa de usuario (en caso de módulos de señales de entradas defectuosos) en lugar de un valor de proceso. Los valores sustitutivos pueden ser especificados por el usuario (p. ej. mantener el valor antiguo).
	Indica que la dirección está bloqueada porque ya está siendo forzada.

Símbolo/Botón	Significado
	Indica que la dirección no puede forzarse.
	Indica que la dirección no puede observarse.
	Indica que la dirección se está forzando permanentemente.
	Indica que parte de una dirección ha sido forzada de forma permanente.
	Indica que una dirección de periferia relacionada ha sido total o parcialmente forzada de forma permanente.
	Indica que una dirección puede no puede forzarse por completo de forma permanente. Ejemplo: Es posible forzar de forma permanente la dirección QW0:P, pero no es posible forzar la dirección QD0:P, ya que esta área de direcciones probablemente no está presente en la CPU.
	Indica que se ha producido un error de sintaxis.
	Indica que si bien la dirección está seleccionada, de momento no se fuerza todavía, p. ej.

Consulte también

Estructura de la tabla de observación (Página 3116)

9.4.3.5 Crear y editar tablas de observación

Crear una tabla de observación

Introducción

Con ayuda de la tabla de observación Ud. podrá observar y forzar variables en el programa de usuario. Después de crear una tabla de observación ésta se puede guardar, duplicar, imprimir y reutilizarse para observar y forzar variables.

Requisitos

Hay un proyecto abierto.

Procedimiento

Para crear una tabla de observación, proceda del siguiente modo:

1. Haga clic en "Vista de proyecto" de la barra de estado.
   Se abre la vista del proyecto.
2. Haga doble clic en el árbol del proyecto sobre la CPU para la que desee crear una tabla de observación.

3. Haga doble clic en primer lugar en la carpeta "Tablas de observación y forzado permanente" y a continuación en el comando "Agregar tabla de observación".
   Se agregará una nueva tabla de observación.
4. Registre en la columna "Nombre" o en la columna "Dirección" el nombre o la dirección absoluta de las variables que desea observar o forzar.
5. En caso de que quiera modificar el formato de visualización predeterminado, seleccione en la columna "Formato de visualización" el formato deseado de la lista desplegable.
6. Decida ahora si quiere observar o forzar las variables introducidas y, en caso necesario, introduzca los valores deseados para el forzado de las variables.

Abrir una tabla de observación

Requisitos

Se ha creado una tabla de observación.

Procedimiento

Para abrir una tabla de observación, proceda del siguiente modo:

1. Abra la carpeta "Tablas de observación y forzado permanente" situada bajo la CPU deseada.
2. Haga doble clic en la tabla de observación deseada de la carpeta.

Resultado

Se abre la tabla de observación seleccionada.

Copiar e insertar una tabla de observación

Requisitos

Se ha creado una tabla de observación.

Procedimiento

Para copiar una tabla de observación, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la tabla de observación que desea copiar.
2. En el menú contextual, elija el comando "Copiar".
3. En el árbol del proyecto, abra la estructura de carpetas de la CPU en la que desea insertar la tabla de observación copiada.
4. Haga clic con el botón derecho del ratón en la carpeta "Tablas de observación y forzado permanente".

5. En el menú contextual, elija el comando "Pegar".
6. Como alternativa, se puede marcar todo el contenido de la tabla de observación y arrastrarlo a otra tabla de observación mediante Drag & Drop .

Resultado

Se inserta una copia de la tabla de observación seleccionada en la carpeta "Tablas de observación y forzado permanente" de la CPU en cuestión.

Guardar una tabla de observación

Requisitos

Se ha creado una tabla de observación.

Procedimiento

Para guardar una tabla de observación, proceda del siguiente modo:

1. Seleccione en la navegación del proyecto la tabla de observación que quiere guardar.
2. En caso de que quiera modificar el nombre predeterminado de la tabla elija el comando "Cambiar nombre" del menú contextual e introduzca un nuevo nombre para la tabla.
3. En el menú "Proyecto" elija el comando "Guardar". Recuerde que con este procedimiento se guardará todo el proyecto.

Resultado

Se guardan tanto el proyecto como los contenidos de la tabla de observación.

Nota

Las tablas de observación guardadas se pueden utilizar nuevamenta para observar y forzar variables cuando vuelva a probar el programa.

9.4.3.6 Introducir variables en la tabla de observación

Información básica para la entrada de variables en la tabla de observación

Procedimiento recomendado

- Determine las variables cuyos valores quiere observar o forzar e introdúzcalas en la tabla de observación.
- Al introducir las variables en la tabla de observación tenga en cuenta que es conveniente definir estas variables primero en la tabla de variables PLC.
- Al introducir las variables vaya de "fuera" hacia "dentro". Es decir, introduzca primero las variables de las entradas en la tabla de observación. A continuación introduzca las variables que en las que influyen las entradas o que influyen en las salidas. Finalmente introduzca las variables de las salidas.

Ejemplo de cómo rellenar una tabla de observación

- Introduzca en la columna "Dirección" la dirección absoluta que debe observarse o forzarse.
- Introduzca en la columna "Nombre" el nombre simbólico de la variable.
- Seleccione en la columna "Formato de visualización" el formato de visualización deseado de la lista desplegable en caso de que no quiera utilizar el ajuste predeterminado.
- Decida ahora si quiere observar o forzar las variables introducidas. Introduzca los valores deseados, así como un comentario para el forzado en la columna correspondiente de la tabla de observación.

Cómo crear una línea de comentario

Introduciendo la cadena de caracteres "//" en la columna "Nombre" puede crear una línea de comentario si lo desea.

Comprobación de sintaxis

Al introducir variables en la tabla de observación se realiza una comprobación sintáctica antes de abandonar la celda. Las entradas erróneas se marcan en rojo.

Nota

Colocando el cursor en una celda marcada en rojo aparece información breve con indicaciones adicionales sobre el error.

Consulte también

Operandos permitidos para la tabla de observación (Página 3123)

Valores de forzado permitidos para la tabla de observación (Página 3124)

Operandos permitidos para la tabla de observación

Operandos autorizados para la tabla de observación

La siguiente tabla muestra los operandos que están autorizados para la tabla de observación:

Operando permitido	Ejemplo del tipo de datos	Ejemplo (nemónicos internacionales)
Entrada/salida/marca	BOOL	I1.0; Q1.7; M10.1 I0.0:P; Q0.0:P
Entrada/salida/marca	BYTE	IB1; QB10; MB100 IB1:P; QB1:P
Entrada/salida/marca	WORD	IW1; QW10; MW100 IW2:P; QW3:P
Entrada/salida/marca	DWORD	ID4; QD10; MD100 ID2:P; QD1:P
Temporizadores	TIMER	T1
Contadores	COUNTER	C1
Bloque de datos	BOOL	DB1.DBX1.0
Bloque de datos	BYTE	DB1.DBB1
Bloque de datos	WORD	DB1.DBW1
Bloque de datos	DWORD	DB1.DBD1

Nota

Tenga en cuenta las siguientes indicaciones para trabajar con la tabla de observación

- Por motivos de utilización interna no es posible introducir "DB0...".
- Las salidas de la periferia pueden forzarse pero no observarse.
- Las entradas de la periferia pueden observarse pero no forzarse.

ATENCIÓN

Peligro de rebasar por exceso el tiempo de ciclo al observar entradas de periferia y forzar salidas de periferia

Tenga en cuenta que al observar entradas de periferia y forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo.

En este caso, la CPU pasa al estado operativo "STOP".

Consulte también

Información básica para la entrada de variables en la tabla de observación (Página 3122)

Vista general de los tipos de datos válidos (Página 2321)

Valores de forzado permitidos para la tabla de observación

Entrada de valores de forzado en la tabla de observación

La siguiente tabla muestra los operandos que están autorizados para la entrada de valores de forzado en la tabla de observación:

Tabla 9-14 Operandos de bit

Operandos de bit posibles	Ejemplo de valores de forzado permitidos
I1.0	true
M1.7	false
Q1.0	0
Q1.1:P	1
DB1.DBX1.1	2#0
M1.6	2#1

Tabla 9-15 Operandos de byte

Operandos de byte posibles	Ejemplo de valores de forzado permitidos
IB1	2#00110011
MB12	B#16#1F
QB10	1F
QB11:P	'a'
DB1.DBB1	10

Tabla 9-16 Operandos de palabra

Operandos de palabra posibles	Ejemplo de valores de forzado permitidos
IW1	2#0011001100110011
MW12	W#16#ABCD
MW14	ABCD
QW10	B#(12, 34)
QW12:P	12345
DB1.DBW1	'ab'
MW16	S5T#9s_340ms
MW18	C#123
MW9	D#2006-12-31

Tabla 9-17 Operandos de palabra doble

Operandos de palabra doble posibles	Ejemplo de valores de forzado permitidos
ID1	2#00110011001100110011001100110011
QD10	Dw#16#abcdef10
QD12:P	ABCDEF10
DB1.DBD2	b#(12,34,56,78)
MD8	L#-12
MD12	L#12
MD16	123456789
MD20	123456789
MD24	T#12s345ms
MD28	Tod#1:2:34.567
MD32	P#e0.0

Tabla 9-18 Temporizadores

Posibles operandos del tipo "Temporizador"	Valores de forzado permitidos	Explicación
T1	0 ms	Valor de tiempo en milisegundos (ms)
T12	20 ms	Valor de tiempo en milisegundos (ms)
T14	12345 ms	Valor de tiempo en milisegundos (ms)
T16	S5t#12s340ms	Valor de tiempo 12s 340 ms

Tabla 9-19 Contadores

Posibles operandos del tipo "Contador"	Valores de forzado permitidos
C1	0
C14	20
C16	C#123

Observaciones sobre temporizadores y contadores

- Temporizadores

Nota

El forzado de un temporizador solo afecta al valor, no al estado. El temporizador T1 se puede forzar al valor "0", el resultado lógico en U T1 no cambiará.

Las secuencias "s5t" y "s5time" se pueden escribir tanto en minúscula como en mayúscula.

- Contadores

Nota

El forzado de un contador solo afecta al valor, no al estado. El contador C1 se puede forzar al valor "0", el resultado lógico en U C1 no cambiará.

Sinopsis de los formatos de visualización

Formatos de visualización de la tabla de observación

Al seleccionar un formato de visualización se define el modo en que se representará el valor de una variable.

Al introducir la dirección aparece seleccionado automáticamente un formato de visualización. Para modificarlo, puede seleccionar el formato de visualización deseado en la lista desplegable de la columna "Formatos de visualización". En la lista desplegable aparecen únicamente los formatos de visualización que son admisibles para este tipo de datos. El formato de visualización mencionado primero es el formato predeterminado.

Ejemplo

La siguiente tabla muestra los tipos de datos de 32 bits que están permitidos para todas las familias de CPU en la tabla de observación, así como sus formatos de visualización:

Tipo de datos	Formatos de visualización posibles
BOOL	Bool, Hex, BCD, Octal, Bin, Dec, Dec+/-
BYTE	Hex, BCD, Octal, Bin, Dec, Dec+/-, carácter
WORD	Hex, BCD, Octal, Bin, Dec, Dec+/-, secuencia_Dec, carácter, carácter Unicode, temporizador SIMATIC, fecha, contador
DWORD	Hex, BCD, Octal, Bin, Dec, Dec+/-, secuencia_Dec, carácter, carácter Unicode, número en coma flotante, hora del día, temporizador, puntero
SINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter
INT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, temporizador SIMATIC, contador, fecha
DINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, número en coma flotante, hora del día, temporizador, puntero
USINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter
UINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, temporizador SIMATIC, contador, fecha
UDINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, número en coma flotante, hora del día, temporizador, puntero
REAL	Número en coma flotante, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_Dec, hora del día, temporizador, puntero
DATE	Fecha, Dec, Hex, BCD, Bin
TIME_OF_DAY	Hora del día, Dec, Hex, BCD, Bin
TIME	Temporizador, Hex, BCD, Bin
DATE_AND_TIME	Fecha y hora,
TIMER	Temporizador SIMATIC, Hex, BCD, Bin
CHAR	Carácter, Hex, BCD, Octal, Bin, Dec, Dec+/-
WCHAR	Carácter_Unicode, carácter, Hex, BCD, Octal, Bin, Dec, Dec+/-
STRING	Cadena de caracteres
WSTRING	Cadena_de_caracteres_Unicode

Tipo de datos	Formatos de visualización posibles
POINTER	Puntero, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_dec, número en coma flotante, hora del día, tiempo, número de bloque
COUNTER	Contador, Hex, BCD, Bin
S5TIME	Temporizador SIMATIC, Hex, BCD, Bin

Para la familia de CPU S7-1200 son admisibles todos los tipos de datos de 32 bits, véase la tabla más arriba, así como el tipo de datos de 64 bits LREAL con los siguientes posibles formatos de visualización:

Tipo de datos	Formatos de visualización posibles
LREAL	en un proyecto creado con el TIA Portal < V12: Número en coma flotante Nota: la visualización de LREAL está limitada a 13 posiciones más el exponente.
LREAL	en un proyecto creado con el TIA Portal >= V12: Número en coma flotante, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_Dec, hora del día, temporizador, fecha y hora Nota: la visualización de LREAL está limitada a 13 posiciones más el exponente.

Para la familia de CPU S7-1500 son admisibles, además de los tipos de datos de 32 bits, los tipos de datos de 64 bits listados en la tabla con los siguientes posibles formatos de visualización:

Tipo de datos	Formatos de visualización posibles
LWORD	Hex, Octal, BCD, Bin, carácter, carácter Unicode, Dec, Dec+/-, secuencia_Dec, número en coma flotante, hora del día, temporizador, fecha y hora
LINT	Dec+/-, Dec, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, hora del día, temporizador, fecha y hora
ULINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, hora del día, temporizador, fecha y hora
LREAL	Número en coma flotante, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_Dec, hora del día, temporizador, fecha y hora
LTIME	Temporizador, Dec+/-, Dec, Hex
LTOD	Hora del día, Dec, Hex, BCD, Bin
LDT	Fecha y hora, Dec, Hex

Encontrará más información en la descripción de los tipos de datos (Página 2321) válidos.

Nota

Redondeo de números en coma flotante

En la tabla de observación los números en coma flotante se guardan en formato IEEE binario. Como no todos los números en coma flotante (real, longreal) representables en la interfaz pueden reproducirse unívocamente en formato IEEE, los números en coma flotante pueden redondearse. Si un número en coma flotante redondeado se copia en la tabla de observación y se inserta en otro campo de entrada, en determinadas circunstancias el redondeo produce una ligera diferencia.

Nota

Solo es posible el direccionamiento simbólico

Los LongDataTypes, por ejemplo LWORD o LREAL, solo se pueden direccionar simbólicamente en la tabla de observación.

Seleccionar un formato de visualización para variables

Procedimiento

Para seleccionar el formato de visualización de las variables, proceda como se indica a continuación:

1. Introduzca la dirección en la tabla de observación.
2. Haga clic en la celda deseada de la columna "Formato de visualización" y abra la lista desplegable.
   Los formatos de visualización autorizados se muestran el la lista desplegable.
3. Seleccione en la lista desplegable el formato de visualización deseado.

   Nota

   En caso de que el formato de visualización seleccionado no se pueda aplicar, se muestra automáticamente el último formato de visualización seleccionado.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Cómo crear y editar líneas de comentario

Principios básicos de las líneas de comentario

Para dar una estructura más clara a la tabla de observación, además de los comentarios referidos a filas de la columna de comentarios ahora es posible crear líneas de comentario completas.

El contenido de la línea de comentarios se guarda en la carpeta "Idiomas y recursos", ficha "Textos del proyecto" y puede traducirse a otros idiomas del proyecto.

Cómo crear líneas de comentarios

Para crear líneas de comentarios, proceda de la manera siguiente:

1. Abra la tabla de observación e introduzca las direcciones deseadas.
2. Introduzca la cadena de caracteres "//" en la columna "Nombre" para crear una línea de comentario. En el resto de las columnas no es posible crear líneas de comentario.
3. Introduzca el comentario deseado en la línea de comentario. El comentario introducido se muestra en verde.
4. Para mostrar todos los comentarios que ha introducido, seleccione en el árbol del proyecto "Idiomas y recursos" y haga doble clic en "Textos del proyecto".
5. Si trabaja con proyectos multilingües y desea traducir sus comentarios a otros idiomas, puede configurar otros idiomas de proyecto además del idioma de edición en el árbol de proyecto en "Idiomas y recursos > Textos del proyecto".

Cómo borrar líneas de comentario

Para borrar líneas de comentario, proceda de la manera siguiente:

1. Abra una tabla de observación que contenga líneas de comentario.
2. Si ya no lo necesita, borre todo el comentario, incluida la cadena de caracteres introductoria "//".
3. Como alternativa, puede borrar solo la cadena de caracteres introductoria "//". De esta forma el comentario se mantiene y se muestra en la columna "Comentario" de la tabla de observación.

Nota

Borrado de líneas de comentario

Al borrar líneas de comentario se borran también los idiomas de proyecto configurados para esos comentarios así como las traducciones que pudiera haber.

9.4.3.7 Observar variables en la tabla de observación

Introducción a la observación de variables en la tabla de observación

Introducción

En la tabla de visualización se pueden observar las variables de los módulos de entrada y de salida en la CPU dependiendo del modo de observación o forzado (Página 3131) seleccionado. Para observar variables debe existir una conexión online con la CPU.

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al observar entradas de periferia Tenga en cuenta que la observación de entradas de periferia puede ocasionar un rebase por exceso del tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Opciones para la observación de variables

Para la observación de variables existen las siguientes opciones:

- Observar inmediatamente
  Este comando inicia la observación de las variables visibles en la tabla de observación activa de inmediato y una vez.
- Observar todos
  Este comando inicia la observación de todas las variables visibles en la tabla de observación en función del modo de observación seleccionado:
  - En el modo básico está predeterminado el modo de observación "permanente".
  - En el modo avanzado se pueden definir puntos de disparo determinados para la observación de las variables.

  Nota

  Si se cambia el modo de observación al modo avanzado y otra vez al modo básico, se toma el modo de observación anteriormente ajustado también en el modo básico.

Limitaciones específicas de la CPU en la observación de variables

Existen las siguientes diferencias específicas de la CPU:

- CPU S7-300/400:
  Las CPUs de esta familia pueden observar únicamente los primeros 30 caracteres de una cadena.
- CPU S7-1200/1500:
  Las CPUs de esta familia pueden observar una cadena de hasta 254 caracteres en total.

Definición del modo de observación y forzado

Introducción

Al seleccionar el modo de observación y forzado se especifica el punto de disparo y el tiempo de observación de las variables en la tabla de observación y en la tabla de forzado permanente.

Modo de observación y forzado posible (tiempo de observación o forzado)

Existen los siguientes modos de observación y forzado:

- Permanente
- Inicio del ciclo, único
- Fin del ciclo, único
- Inicio del ciclo, permanente
- Fin del ciclo, permanente
- Transición de RUNa STOP, única
- Transición a STOP, permanente

Particularidades del modo "Permanente"

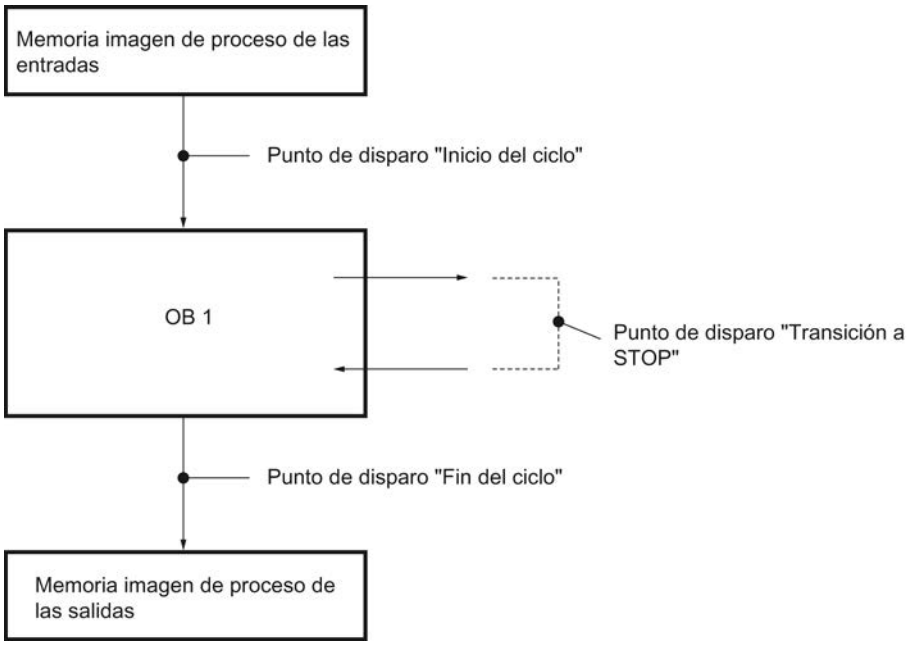
El modo "Permanente" se ejecuta de forma distinta para la observación y el forzado de variables:

- Observación: las entradas se observan al final y las salidas al principio del ciclo.
- Forzado: las entradas se fuerzan al principio y las salidas al final del ciclo.

Definición de un punto de disparo

Los puntos de disparo "Inicio del ciclo", "Fin del ciclo" y "Transición de RUN a STOP" definen el momento en que las variables se leerán de la CPU o se actualizarán en la CPU.

La siguiente figura muestra la posición de los puntos de disparo mencionados:



Posición de los puntos de disparo

De la posición de los puntos de disparo se deduce lo siguiente:

- El forzado de entradas sólo resulta efectivo al inicio del ciclo (equivale al inicio del programa de usuario OB 1), ya que de lo contrario la memoria imagen de proceso de las entradas se actualiza nuevamente después del forzado, con lo que se sobrescribe.
- El forzado de salidas sólo resulta efectivo al final del ciclo (equivale al final del programa de usuario OB 1), ya que de lo contrario el programa de usuario puede sobrescribir la memoria imagen de proceso de las salidas.
- El valor forzado se indica en la columna "Valor de observación", siempre que se observe de forma activa y que el programa de usuario no sobrescriba el valor forzado.

Forzado de variables

Al forzar variables rige para los puntos de disparo:

- Si ha seleccionado "único" como modo de forzado, aparecerá un mensaje en caso de que no se puedan forzar las variables seleccionadas.
- con el modo de forzado "permanente" no aparece ningún mensaje.

Observación respecto al comando "Forzar inmediatamente"

Los valores de las variables seleccionadas se pueden forzar inmediatamente con el comando "Online > Forzar> Forzar inmediatamente". Esta petición se realiza una sola vez y lo más rápido posible, sin referencia a un punto determinado (punto de disparo) del programa de usuario. Esta función se utiliza principalmente en el estado STOP de la CPU para forzar.

Variables "Observar todos"

Introducción

El comando "Observar todos" permite iniciar la observación de las variables visibles en la tabla de observación activa. En el modo básico de la tabla de observación está predeterminada la opción "permanente" para el modo de observación. En el modo avanzado se pueden definir puntos de disparo determinados para la observación de las variables. En ese caso, las variables se observan con respecto a los puntos de disparo definidos.

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al observar entradas de periferia Tenga en cuenta que la observación de entradas de periferia puede ocasionar un rebase por exceso del tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.

Procedimiento

Para ejecutar el comando "Observar todos", proceda del siguiente modo:

1. Introduzca las variables que desee observar y las direcciones correspondientes en la tabla de observación.
2. Cambie al modo avanzado haciendo clic en el botón "Ocultar o mostrar todas las columnas del modo avanzado".
3. Para modificar el modo de observación predeterminado de una variable, haga clic en la celda correspondiente de la columna "Observar con disparador" y seleccione el modo de observación que desee en la lista desplegable.
4. Haga clic en el botón "Observar todos" de la barra de herramientas.

Resultado

Las variables de la tabla de observación activa se observan en el modo de observación seleccionado en cada caso.

Consulte también

Símbolos y botones utilizados en la tabla de observación (Página 3118)

Introducir variables en la tabla de observación (Página 3122)

Modo básico/avanzado en la tabla de observación (Página 3117)

"Observar inmediatamente" variables

Introducción

Con el comando "Observar inmediatamente" se inicia la observación de las variables inmediatamente sin referencia a puntos de disparo determinados. Los valores de las variables se leen una sola vez y se muestran en la tabla de observación.

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al observar entradas de periferia Tenga en cuenta que la observación de entradas de periferia puede ocasionar un rebase por exceso del tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.

Procedimiento

Para ejecutar el comando "Observar inmediatamente", proceda del siguiente modo:

1. Introduzca las variables que desee observar y las direcciones correspondientes en la tabla de observación.
2. Haga clic en el botón "Observar inmediatamente" en la barra de herramientas.

Resultado

Las variables de la tabla de observación activa se observan inmediatamente y una sola vez.

Consulte también

Símbolos y botones utilizados en la tabla de observación (Página 3118)

Introducir variables en la tabla de observación (Página 3122)

Modo básico/avanzado en la tabla de observación (Página 3117)

9.4.3.8 Forzar variables en la tabla de observación

Introducción al forzado de variables

Introducción

La tabla de observación permite forzar las variables de los módulos de entrada y salida configurados en la CPU en función del modo de observación y forzado (Página 3131) seleccionado. Para poder observar las variables debe existir una conexión online con la CPU.

⚠ PELIGRO
Peligro al forzar: La modificación de las variables o direcciones durante el funcionamiento de la instalación puede provocar graves daños materiales o personales en caso de fallos de funcionamiento o del programa. Asegúrese de que no pueden darse estados peligrosos antes de ejecutar la función "Forzar".

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al forzar salidas de periferia Tenga en cuenta que al forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Opciones del forzado de variables

Para el forzado de variables existen las siguientes opciones:

- Forzar a "0"
  Este comando fuerza la dirección seleccionada al valor de forzado "0".
- Forzar a "1"
  Este comando fuerza la dirección seleccionada al valor de forzado "1".
- Forzar inmediatamente una sola vez
  Este comando fuerza "inmediatamente una sola vez" todas las direcciones activadas en la tabla de observación activa.

- Forzar con disparador
  Este comando fuerza todas las direcciones activadas en la tabla de observación activa con el modo de observación y forzado (Página 3131) seleccionado en cada caso.
  La función "Forzar con disparador" sólo está disponible en el modo avanzado. No recibirá ninguna notificación de si las direcciones activadas realmente se han forzado al valor predefinido o no. Para recibir una notificación de ese tipo, utilice la función "Forzar inmediatamente una sola vez".
- Desbloquear salidas
  Este comando desactiva el bloqueo de salida de comandos (BASP).
  Esta función solamente se puede realizar en el modo avanzado cuando la CPU está en STOP y cuando el forzado permanente (Página 3160) de las variables no está activado.
  Desactive esta función, si así lo desea, en la tabla de forzado permanente.

Nota

Al forzar, tenga en cuenta lo siguiente:

El forzado **no** puede deshacerse.

Forzar variables a "0"

Introducción

Es posible asignar valores a las variables una sola vez y forzarlas independientemente del modo de observación y forzado. En la ejecución, la petición de forzado se ejecuta lo más rápido posible, como si se tratase de un "disparador inmediato", sin referencia a un punto determinado del programa de usuario.

⚠ PELIGRO

Peligro al forzar:

La modificación de las variables o direcciones durante el funcionamiento de la instalación puede provocar graves daños materiales o personales en caso de fallos de funcionamiento o del programa.
Asegúrese de que no pueden darse estados peligrosos antes de ejecutar la función "Forzar".

ATENCIÓN

Peligro de rebasar por exceso el tiempo de ciclo al forzar salidas de periferia

Tenga en cuenta que al forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo.

En este caso, la CPU pasa al estado operativo "STOP".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.

Procedimiento

Para forzar variables a "0", proceda del siguiente modo:

1. Introduzca la dirección en la tabla de observación.
2. Elija el comando "Online > Forzar > Forzar a 0" para forzar la dirección seleccionada al valor predefinido.

Resultado

La dirección seleccionada se fuerza a "0".

Nota

Al forzar, tenga en cuenta lo siguiente:

El forzado **no** puede deshacerse.

Forzar variables a "1"

Introducción

Es posible asignar valores a las variables una sola vez y forzarlas independientemente del modo de observación y forzado. En la ejecución, la petición de forzado se ejecuta lo más rápido posible, como si se tratase de un "disparador inmediato", sin referencia a un punto determinado del programa de usuario.

⚠ PELIGRO

Peligro al forzar:

La modificación de las variables o direcciones durante el funcionamiento de la instalación puede provocar graves daños materiales o personales en caso de fallos de funcionamiento o del programa.
Asegúrese de que no pueden darse estados peligrosos antes de ejecutar la función "Forzar".

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al forzar salidas de periferia Tenga en cuenta que al forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.

Procedimiento

Para forzar variables a "1", proceda del siguiente modo:

1. Introduzca la dirección en la tabla de observación.
2. Elija el comando "Online > Forzar > Forzar a 1" para forzar la dirección seleccionada al valor predefinido.

Resultado

La dirección seleccionada se fuerza a "1".

Nota
Al forzar, tenga en cuenta lo siguiente:
El forzado **no** puede deshacerse.

"Forzar inmediatamente" variables

Introducción

Es posible asignar valores a las variables una sola vez y forzarlas de forma inmediata independientemente del modo de observación y forzado. En la ejecución, la petición de

forzado se ejecuta lo más rápido posible, como si se tratase de un "disparador inmediato", sin referencia a un punto determinado del programa de usuario.

⚠ PELIGRO
Peligro al forzar: La modificación de las variables o direcciones durante el funcionamiento de la instalación puede provocar graves daños materiales o personales en caso de fallos de funcionamiento o del programa. Asegúrese de que no pueden darse estados peligrosos antes de ejecutar la función "Forzar".

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al forzar salidas de periferia Tenga en cuenta que al forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.

Procedimiento

Para forzar variables inmediatamente, proceda del siguiente modo:

1. Introduzca la dirección y los valores de forzado en la tabla de observación.
2. Seleccione las direcciones que desee forzar activando la casilla de verificación de forzado en la columna que sigue al "Valor de forzado".
   Junto a la casilla de verificación activada aparece un triángulo amarillo que indica que la dirección está seleccionada para el forzado, pero que todavía no ha sido forzada.
3. Elija el comando "Online > Forzar > Forzar inmediatamente" para forzar inmediatamente y una sola vez con los valores predefinidos las direcciones seleccionadas.

Resultado

Las direcciones seleccionadas se fuerzan inmediatamente una sola vez.

Nota

Al forzar, tenga en cuenta lo siguiente:

El forzado **no** puede deshacerse.

"Forzar con disparador" variables

Introducción

Es posible asignar valores a las direcciones y forzarlas acorde con el modo de observación y forzado definido. En la ejecución, la petición de forzado se ejecuta de la forma definida en el modo de observación y forzado con referencia al punto del programa de usuario definido como disparador.

⚠ PELIGRO
Peligro al forzar: La modificación de las variables o direcciones durante el funcionamiento de la instalación puede provocar graves daños materiales o personales en caso de fallos de funcionamiento o del programa. Asegúrese de que no pueden darse estados peligrosos antes de ejecutar la función "Forzar".

ATENCIÓN
Peligro de rebasar por exceso el tiempo de ciclo al forzar salidas de periferia Tenga en cuenta que al forzar salidas de periferia, es posible que en la tabla de observación se rebase por exceso el tiempo de ciclo. En este caso, la CPU pasa al estado operativo "STOP".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.
- La tabla de observación debe encontrarse en modo avanzado.

Procedimiento

Para forzar variables "con disparador", proceda del siguiente modo:

1. Introduzca la dirección y los valores de forzado en la tabla de observación.
2. Seleccione las direcciones que desee forzar activando la casilla de verificación de forzado en la columna que sigue al "Valor de forzado".
   Junto a la casilla de verificación activada aparece un triángulo amarillo que indica que la dirección está seleccionada para el forzado, pero que todavía no ha sido forzada.
3. Cambie al modo avanzado con el botón "Ocultar o mostrar todas las columnas del modo avanzado" de la barra de herramientas o con el comando "Online > Modo avanzado".
   Aparecen las columnas "Observar con disparador" y "Forzar con disparador".

4. Seleccione el modo de forzado que desee utilizar en la lista desplegable de la columna "Forzar con disparador". Existen las opciones siguientes:
   - Permanente
   - Inicio del ciclo, permanente
   - Inicio del ciclo, único
   - Fin del ciclo, permanente
   - Fin del ciclo, único
   - Transición a STOP, permanente
   - Transición a STOP, única
5. Inicie el forzado con el comando "Online > Forzar > Forzar con disparador".
6. Confirme la pregunta con "Sí" si desea iniciar el forzado con disparador.

Resultado

Las variables activadas se fuerzan con el modo de observación y forzado seleccionado. El triángulo amarillo ya no aparece.

Nota

Al forzar, tenga en cuenta lo siguiente:

El forzado **no** puede deshacerse.

Desbloquear salidas

Introducción

La función "Desbloquear salidas" desactiva el bloqueo de las salidas de periferia (BASP). Esto permite forzar las salidas de la periferia cuando la CPU se encuentra en el estado operativo STOP. Esta función sólo está disponible en la tabla de observación en el "Modo avanzado".

⚠ PELIGRO

Peligro al desbloquear las salidas de la periferia:

Tenga en cuenta que el desbloqueo de las salidas de la periferia pueden causar daños materiales y personales graves.
Asegúrese de que no se pueden producir condiciones peligrosas antes de ejecutar la función "Desbloquear salidas".

Requisitos

- Se ha creado una tabla de observación.
- Existe una conexión online con la CPU.

- La CPU debe estar en el estado operativo STOP para desbloquear las salidas de la periferia.
- La tabla de observación debe encontrarse en modo avanzado.
- El forzado permanente (Página 3160) de variables no debe estar activo.

Nota

Función "Desbloquear salidas"

- Esta función es posible únicamente en el estado operativo STOP. Se finaliza cuando la CPU cambia de estado operativo y al cerrar la conexión online.
- Mientras la función está activa, no se puede forzar de forma permanente.

Procedimiento

Para desbloquera las salidas en el estado operativo STOP, proceda del siguiente modo:

1. Introduzca la dirección y los valores de forzado en la tabla de observación.
2. Seleccione las direcciones que desee forzar activando la casilla de verificación de forzado en la columna que sigue al "Valor de forzado".
   Junto a la casilla de verificación activada aparece un triángulo amarillo que indica que la dirección está seleccionada para el forzado, pero que todavía no ha sido forzada.
3. Cambie al modo avanzado con el botón "Ocultar o mostrar todas las columnas del modo avanzado" de la barra de herramientas o con el comando "Online > Modo avanzado".
   Aparecen las columnas "Observar con disparador" y "Forzar con disparador".
4. Cambie la CPU asignada al estado operativo STOP desde el panel de mando.
5. Abra el menú contextual pulsando el botón derecho del ratón y elija el comando "Desbloquear salidas".
6. Confirme la pregunta con "Sí" si desea desactivar el bloqueo de salida de comandos para las salidas de periferia.
7. Fuerce las salidas de periferia con el comando "Online > Forzar > Forzar inmediatamente".

Resultado

Las salidas de periferia se fuerzan con los valores de forzado seleccionados. El triángulo amarillo ya no aparece.

Desbloqueo de las salidas de periferia

La función "Desbloquear salidas" permanece activada hasta que:

- El comando "Desbloquear salidas" se vuelve a desactivar desde el menú contextual o con el comando "Online > Forzar > Desbloquear salidas".
- La CPU ya no está en estado operativio STOP.
- Se finaliza la conexión online.

Nota

Al forzar, tenga en cuenta lo siguiente:

El forzado **no** puede deshacerse.

9.4.4 Test con la tabla de forzado permanente

9.4.4.1 Introducción al test con la tabla de forzado permanente

Vista general

La tabla de forzado permanente permite preasignar valores fijos a distintas variables del programa de usuario

. Este procedimiento se conoce como "forzado permanente".

Dentro de la tabla de forzado permanente se dispone de las siguientes funciones:

- **Observar variables**
  Esta opción permite visualizar en la programadora o el PC los valores actuales de distintas variables de un programa de usuario o de una CPU. La observación se puede efectuar con o sin condición de disparo.
- **Forzado permanente de variables**
  Con ello puede asignar un valor fijo a variables de periferia individuales de un programa de usuario.

Observar y forzar permanentemente variables

La observación y el forzado permanente de variables siempre depende del volumen de operandos de la CPU utilizada.

Se pueden observar las siguientes variables:

- entradas, salidas y marcas
- contenidos de bloques de datos
- Entradas de periferia

Se pueden forzar permanentemente las siguientes variables:

- Entradas de periferia
- Salidas de periferia

Ejemplo

- Independientemente de la CPU utilizada en cada caso, se puede forzar permanentemente la periferia solamente, así por ejemplo: "Tag_1":P ó "QW0:P" ó "IW0:P". Recuerde que "Tag_1":P no puede ser el nombre simbólico de una marca.

Posible utilización

La tabla de forzado permanente no sólo ofrece la ventaja de poder realizar diferentes simulaciones en entorno de pruebas, sino también la de poder sobrescribir variables en la CPU con un valor fijo. Con ello se dispone de la posibilidad de intervenir, ajustando el proceso en curso.

Consulte también

Estructura de la tabla de forzado permanente (Página 3145)

Modo básico y modo avanzado en la tabla de forzado permanente (Página 3146)

Símbolos utilizados en la tabla de forzado permanente (Página 3147)

Abrir y procesar tabla de forzado permanente (Página 3148)

9.4.4.2 Medidas de seguridad para el forzado permanente de variables

Medidas de seguridad para el forzado permanente de variables

Dado que con la función de forzado permanente se interviene permanentemente en el proceso, es necesario observar las siguientes indicaciones:

PELIGRO

Evitación de daños personales y materiales

Tenga en cuenta que al ejecutar la función "Forzado permanente", una acción incorrecta puede:

- puede poner en peligro la vida o salud de personas.
- causar daños materiales en la máquina o en toda la instalación

⚠ PRECAUCIÓN

Evitación de daños personales y materiales

- Antes de iniciar la función "Forzado permanente" asegúrese de que nadie más vaya a ejecutar simultáneamente esta función en la misma CPU.
- El forzado permanente sólo puede finalizarse haciendo clic en el botón "Finalizar forzado permanente" o con el comando "Online > Forzar permanentemente > Finalizar forzado permanente". Cerrando la tabla de forzado permanente activa no se finaliza el forzado permanente.
- El "forzado permanente" no puede deshacerse.
- Infórmese sobre las diferencias entre el "forzado de variables (Página 3135)" y el "forzado permanente de variables (Página 3160)".
- Si una CPU no soporta la función de "forzado permanente", los botones relacionados con esa función no estarán disponibles.
- Si está activada la función "Desbloquear salidas de periferia" en la CPU utilizada, no se podrá efectuar el forzado permanente en esa CPU. Desactive esta función, si así lo desea, en la tabla de observación.

9.4.4.3 Estructura de la tabla de forzado permanente

Introducción

En la tabla de forzado permanente se registran las variables válidas para las CPU definidas y seleccionadas por usted, que se deben forzar permanentemente en la CPU asociada. Solamente se pueden forzar permanentemente entradas de periferia y salidas de periferia.

Para cada CPU creada en el proyecto se genera automáticamente una tabla de forzado permanente en la carpeta "Tablas de observación y de forzado permanente". Solamente se puede asociar una única tabla de forzado permanente a una CPU. Esta tabla de forzado permanente indica todas las direcciones forzadas permanentemente en la CPU asociada.

Estructura de la tabla de forzado permanente

Las columnas visualizadas en la tabla de forzado permanente varían en función de si se está trabajando en el modo básico o avanzado.

En el modo avanzado se indica además la columna "Observar con disparo"

Significado de las columnas

La tabla siguiente muestra el significado de las diferentes columnas en el modo básico y en el modo avanzado:

Modo	Columna	Explicación
Modo básico	i	Columna identificativa
	Nombre	Nombre de la variable insertada
	Dirección	Dirección de la variable insertada
	Formato de visualización	Formato de visualización seleccionado
	Valor de observación	Valores de las variables en función del formato de visualización seleccionado.
	Valor de forzado permanente	Valor con el que se fuerza la variable de forma permanente.
	F ("Forzar permanentemente")	Selección de las variables que se van a forzar permanentemente activando la casilla de verificación correspondiente.
	Comentario	Comentario para la documentación de las variables
En el modo avanzado se muestra también la siguiente columna:	Observar con disparo	Indicación del modo de observación seleccionado

Consulte también

Símbolos utilizados en la tabla de forzado permanente (Página 3147)

Modo básico y modo avanzado en la tabla de forzado permanente (Página 3146)

9.4.4.4 Modo básico y modo avanzado en la tabla de forzado permanente

Diferencias entre modo básico y modo avanzado en la tabla de forzado permanente

En el modo avanzado se muestra además en la tabla de forzado permanente la columna "Observar con disparo".

Encontrará una relación detallada de las diferentes columnas en Estructura de la tabla de forzado permanente (Página 3145).

Conmutar entre el modo básico y el modo avanzado

Para conmutar entre el modo básico y el avanzado existen las siguientes opciones:

- Haga clic en el icono "Mostrar u ocultar todas las columnas del modo avanzado". Haciendo clic de nuevo en el mismo botón se regresa al modo básico.
  O bien:
- Active la casilla de verificación antepuesta al comando "Modo avanzado" del menú "Online". Desactivando la casilla de verificación se regresa al modo básico.

Funciones del modo avanzado

Las siguientes funciones sólo están disponibles en el modo avanzado:

- Observar con disparador
- Observar entradas de periferia

9.4.4.5 Símbolos utilizados en la tabla de forzado permanente

Significado de los símbolos

La tabla siguiente muestra el significado de los símbolos en la tabla de forzado permanente:

Símbolo/Botón	Significado
	Identificador de una tabla dentro del árbol del proyecto como tabla de forzado permanente.
	Columna identificativa
	Inserta una fila delante de la fila seleccionada.
	Inserta una fila detrás de la fila seleccionada.
	Muestra todas las columnas del modo avanzado. Haciendo clic de nuevo en este botón, se ocultan de nuevo las columnas del modo avanzado.
	Inicia el forzado permanente de todas las direcciones de las variables seleccionadas. Si ya hay una operación de forzado permanente activa, ésta se sustituye sin interrupción.
	Finaliza el forzado permanente de direcciones en la tabla de forzado permanente.
	Inicia la observación de las variables visibles en la tabla de forzado permanente. En el modo básico, para la observación está predeterminada la opción "permanente". En el modo avanzado aparece además una columna y se pueden configurar puntos de disparo determinados para la observación de las variables.
	Inicia la observación de las variables visibles en la tabla de forzado permanente. Este comando se ejecuta inmediatamente y las variables se observan una sola vez.
	Muestra la casilla de verificación para la selección de las variables que deben forzarse permanentemente.
	Indica que una dirección no puede forzarse permanentemente.
	Indica que una dirección no puede forzarse permanentemente en su totalidad. Ejemplo: Si bien es posible forzar permanentemente la dirección AW0:P, no es posible sin embargo, forzar permanentemente la dirección AD0:P, Puesto que ese intervalo de direcciones no existe en la CPU.
	Indica que una dirección no puede observarse.
	Indica que una dirección se está forzando permanentemente.
	Indica que parte de una dirección se está forzando permanentemente.

Símbolo/Botón	Significado
	Indica que la dirección de periferia asociada se está forzando permanentemente.
	Indica que se ha producido un error de sintaxis.
	Indica que la dirección está seleccionada, pero que por ejemplo, aún no está siendo forzada permanentemente.

Consulte también

Estructura de la tabla de forzado permanente (Página 3145)

9.4.4.6 Abrir y procesar tabla de forzado permanente

Visualización de la tabla de forzado permanente

Introducción

Una tabla de forzado permanente no se puede crear, existe ya para cada CPU. Está asociada a dicha CPU y no se puede copiar ni duplicar.

Requisitos

Debe estar abierto un proyecto con una CPU asociada.

Visualización de una tabla de forzado permanente

La tabla de forzado permanente se muestra siempre en una CPU bajo la carpeta "Tablas de observación y de forzado permanente".

Abrir tabla de forzado permanente

Requisitos

Debe haberse creado un proyecto con una CPU asociada.

Procedimiento

Para abrir una tabla de forzado permanente, proceda del siguiente modo:

1. Abra la carpeta "Tablas de observación y de forzado permanente" situada bajo la CPU deseada.
2. Haga doble clic en la "Tabla de forzado permanente" que contiene esta carpeta.

Resultado

Se abre la tabla de forzado permanente seleccionada.

Guardar tabla de forzado permanente

Requisitos

Se ha creado un proyecto con una CPU asociada.

Procedimiento

Para guardar una tabla de forzado permanente, proceda del siguiente modo:

1. Introduzca las modificaciones deseadas en la tabla de forzado permanente.
2. Seleccione en el menú "Proyecto" el comando "Guardar" o haga clic en el icono "Guardar proyecto" en la barra de herramientas. Recuerde que con este procedimiento se guardará todo el proyecto.

Resultado

Se guardan tanto el proyecto como los contenidos de la tabla de forzado permanente.

Nota

No es posible cambiar el nombre de una tabla de forzado permanente.

9.4.4.7 Introducir variables en la tabla de forzado permanente

Información básica para la entrada de variables en la tabla de forzado permanente

Procedimiento recomendado

Determine para qué variables desea observar o forzar permanentemente valores e introdúzcalas en la tabla de forzado permanente.

Tenga en cuenta que al introducir variables en la tabla de forzado permanente, tiene sentido definir estas variables previamente en la tabla de variables PLC.

Ejemplo de cómo rellenar una tabla de forzado permanente

- Puede introducir en la columna "Dirección" las direcciones absolutas que se van a forzar permanentemente o a observar, o introduzca en la columna "Nombre" el nombre simbólico de la variable.
- Si no desea utilizar el ajuste predeterminado para el formato de visualización, seleccione el formato que desee en la lista desplegable de la columna "Formato de visualización".

- A continuación, decida si desea observar o forzar permanentemente las variables registradas. Introduzca para ello el valor de forzado permanente deseado así como un comentario en las columnas correspondientes de la tabla de forzado permanente.
- Tenga en cuenta que solamente se pueden forzar permanentemente entradas de periferia y salidas de periferia e infórmese acerca de las medidas de seguridad para el forzado de variables (Página 3162).

Cómo crear una línea de comentario

Introduciendo la cadena de caracteres "//" en la columna "Nombre" puede crear una línea de comentario si lo desea.

Comprobación de sintaxis

Cuando se introducen variables en la tabla de forzado permanente, antes de salir de la celda correspondiente se comprueba la sintaxis. Las entradas erróneas se marcan en rojo.

Nota

Colocando el cursor en una celda marcada en rojo aparece información breve con indicaciones adicionales sobre el error.

Operandos permitidos para la tabla de forzado permanente

Operandos permitidos para la tabla de forzado permanente

La tabla siguiente muestra los operandos permitidos para el forzado permanente en la tabla de forzado permanente:

Operando permitido	Ejemplo de tipo de datos	Ejemplo (nemónicos internacionales)
Entrada de periferia/salida de periferia	BOOL	I0.0:P; Q0.0:P
Entrada de periferia/salida de periferia	BYTE	IB1:P; QB1:P
Entrada de periferia/salida de periferia	WORD	IW2:P; QW3:P
Entrada de periferia/salida de periferia	DWORD	ID2:P; QD1:P

La tabla siguiente muestra los operandos permitidos para la observación en la tabla de forzado permanente:

Operando permitido	Ejemplo de tipo de datos	Ejemplo (nemónicos internacionales)
Entrada/salida/marca	BOOL	I1.0; Q1.7; M10.1 E0.0:P
Entrada/salida/marca	BYTE	IB1; QB10; MB100 IB1:P
Entrada/salida/marca	WORD	IW1; QW10; MW100 IW2:P
Entrada/salida/marca	DWORD	ID4; QD10; MD100 ID2:P
Temporizadores	TIMER	T1
Contadores	COUNTER	C1
Bloque de datos	BOOL	DB1.DBX1.0
Bloque de datos	BYTE	DB1.DBB1
Bloque de datos	WORD	DB1.DBW1
Bloque de datos	DWORD	DB1.DBD1

Nota

Por motivos de utilización interna no es posible introducir "DB0...".

Valores de forzado permanente permitidos para la tabla de forzado permanente

Entrada de valores de forzado permanente en la tabla de forzado permanente

La tabla siguiente muestra los operandos permitidos para la entrada de valores de forzado permanente en la tabla de forzado permanente:

Tabla 9-20 Operandos de bit

Operandos de bit posibles	Ejemplo de valores de forzado permanente permitidos
I1.0:P	true
I1.1:P	false
Q1.0:P	0
Q1.1:P	1
I2.0:P	2#0
I2.1:P	2#1

Tabla 9-21 Operandos de byte

Operandos de byte posibles	Ejemplo de valores de forzado permanente permitidos
IB1:P	2#00110011
IB2:P	B#16#1F
QB14:P	1F
QB10:P	'a'
IB3:P	10

Tabla 9-22 Operandos de palabra

Operandos de palabra posibles	Ejemplo de valores de forzado permanente permitidos
IW0:P	2#0011001100110011
IW2:P	W#16#ABCD
QW10:P	ABCD
QW12:P	B#(12, 34)
IW4:P	'ab'
IW6:P	12345
IW8:P	S5T#9S_340ms
IW10:P	C#123
IW12:P	D#2006-12-31

Tabla 9-23 Operandos de palabra doble

Operandos de palabra doble posibles	Ejemplo de valores de forzado permanente permitidos
ID0:P	2#00110011001100110011001100110011
ID4:P	1.2
QD10:P	1.234.e4
QD14:P	Dw#16#abcdef10
ID8:P	16#ABCDEF10
ID12:P	b#(12,34,56,78)
ID16:P	L#-12
ID20:P	L#12
ID24:P	123456789
ID28:P	123456789
ID32:P	T#12s345ms
ID36:P	Tod#14:20:40.645
ID40:P	P#e0.0

Sinopsis de los formatos de visualización

Formatos de visualización en la tabla de forzado permanente

Al seleccionar un formato de visualización se define el modo en que se representará el valor de una variable.

Al introducir la dirección aparece seleccionado automáticamente un formato de visualización. Para modificarlo, puede seleccionar el formato de visualización deseado en la lista desplegable de la columna "Formatos de visualización". En la lista desplegable aparecen únicamente los formatos de visualización que son admisibles para este tipo de datos. El formato de visualización mencionado primero es el formato predeterminado.

Ejemplo

La tabla siguiente muestra los tipos de datos de 32 bits permitidos para todas las familias de CPU en la tabla de forzado permanente así como sus posibles formatos de visualización.

Tipo de datos	Formatos de visualización posibles
BOOL	Bool, Hex, BCD, Octal, Bin, Dec, Dec+/-
BYTE	Hex, BCD, Octal, Bin, Dec, Dec+/-, carácter
WORD	Hex, BCD, Octal, Bin, Dec, Dec+/-, secuencia_Dec, carácter, temporizador SIMATIC, fecha, carácter Unicode, contador
DWORD	Hex, BCD, Octal, Bin, Dec, Dec+/-, secuencia_Dec, carácter, número en coma flotante, hora del día, temporizador, puntero, carácter Unicode
SINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter
INT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, temporizador SIMATIC, contador, fecha
DINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, número en coma flotante, hora del día, temporizador, puntero
USINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter
UINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, temporizador SIMATIC, contador, fecha
UDINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, número en coma flotante, hora del día, temporizador, puntero
REAL	Número en coma flotante, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_Dec, hora del día, temporizador, puntero
DATE	Fecha, Dec, Hex, BCD, Bin
TIME_OF_DAY	Hora del día, Dec, Hex, BCD, Bin
TIME	Temporizador, Hex, BCD, Bin
DATE_AND_TIME	Fecha y hora,
TIMER	Temporizador SIMATIC, Hex, BCD, Bin
CHAR	Carácter, Hex, BCD, Octal, Bin, Dec, Dec+/-
WCHAR	Carácter_Unicode, carácter, Hex, BCD, Octal, Bin, Dec, Dec+/-
STRING	Cadena de caracteres
WSTRING	Cadena_de_caracteres_Unicode

Tipo de datos	Formatos de visualización posibles
POINTER	Puntero, Hex, BCD, Octal, Bin, carácter, carácter Unicode, Dec, Dec+/-, secuencia_Dec, número en coma flotante, hora del día, temporizador, número
COUNTER	Contador, Hex, BCD, Bin
S5TIME	Temporizador SIMATIC, Hex, BCD, Bin

Para la familia de CPU S7-1200 son admisibles todos los tipos de datos de 32 bits, véase la tabla más arriba, así como el tipo de datos de 64 bits LREAL con los siguientes posibles formatos de visualización:

Tipo de datos	Formatos de visualización posibles
LREAL	en un proyecto creado con el TIA Portal < V12: Número en coma flotante Nota: la visualización de LREAL está limitada a 13 posiciones más el exponente.
LREAL	en un proyecto creado con el TIA Portal >= V12: Número en coma flotante, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_Dec, hora del día, temporizador, fecha y hora Nota: la visualización de LREAL está limitada a 13 posiciones más el exponente.

Para la familia de CPU S7-1500 son admisibles, además de los tipos de datos de 32 bits, los tipos de datos de 64 bits listados en la tabla con los siguientes posibles formatos de visualización:

Tipo de datos	Formatos de visualización posibles
LWORD	Hex, Octal, BCD, Bin, carácter, carácter Unicode, Dec, Dec+/-, secuencia_Dec, número en coma flotante, hora del día, temporizador, fecha y hora
LINT	Dec+/-, Dec, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, hora del día, temporizador, fecha y hora
ULINT	Dec, Dec+/-, Hex, BCD, Octal, Bin, carácter, carácter Unicode, secuencia_Dec, hora del día, temporizador, fecha y hora
LREAL	Número en coma flotante, Hex, BCD, Octal, Bin, carácter, carácter_Unicode, Dec, Dec+/-, secuencia_Dec, hora del día, temporizador, fecha y hora
LTIME	Temporizador, Dec+/-, Dec, Hex
LTOD	Dec, Hex, BCD, Bin, hora del día
LDT	Dec, Hex, fecha y hora

Encontrará más información adicionalmente en la descripción de los tipos de datos (Página 2321) válidos.

Nota

Redondeo de números en coma flotante

En la tabla de forzado permanente los números en coma flotante se guardan en formato IEEE binario. Como no todos los números en coma flotante (real, longreal) representables en la interfaz pueden reproducirse unívocamente en formato IEEE, los números en coma flotante pueden redondearse. Si un número en coma flotante redondeado se copia en la tabla de forzado permanente y se inserta en otro campo de entrada, en determinadas circunstancias el redondeo produce una ligera diferencia.

Nota

Solo es posible el direccionamiento simbólico

Los LongDataTypes, como por ejemplo LWORD o LREAL, solo se pueden direccionar simbólicamente en la tabla de forzado permanente.

Seleccionar un formato de visualización para variables

Procedimiento

Para seleccionar el formato de visualización de las variables, proceda del siguiente modo:

1. Introduzca la dirección en la tabla de forzado permanente.
2. Haga clic en la celda deseada de la columna "Formato de visualización" y abra la lista desplegable.
   En la lista desplegable se muestran los formatos de visualización permitidos.
3. Seleccione el formato de visualización deseado de la lista desplegable.

   Nota

   En caso de que el formato de visualización seleccionado no se pueda aplicar, se muestra automáticamente el último formato de visualización seleccionado.

Cómo crear y editar líneas de comentario

Principios básicos de las líneas de comentario

Para dar una estructura más clara a la tabla de forzado permanente, además de los comentarios referidos a filas de la columna de comentarios es posible crear líneas de comentario completas.

El contenido de las líneas de comentario se guarda en la carpeta "Idiomas y recursos", ficha "Textos del proyecto" y puede traducirse a otros idiomas del proyecto.

Cómo crear líneas de comentario

Para crear líneas de comentario, proceda de la manera siguiente:

1. Abra la tabla de forzado permanente e introduzca las direcciones deseadas.
2. Introduzca la cadena de caracteres "//" en la columna "Nombre" para crear una línea de comentario. En el resto de las columnas no es posible crear líneas de comentario.
3. Introduzca el comentario deseado en la línea de comentario. El comentario introducido se muestra en verde.
4. Para mostrar todos los comentarios que ha introducido, seleccione en el árbol del proyecto "Idiomas y recursos" y haga doble clic en "Textos del proyecto".
5. Si trabaja con proyectos multilingües y desea traducir sus comentarios a otros idiomas, puede configurar otros idiomas de proyecto además del idioma de edición en el árbol de proyecto en "Idiomas y recursos > Textos del proyecto".

Cómo borrar líneas de comentario

Para borrar líneas de comentario, proceda de la manera siguiente:

1. Abra una tabla de forzado permanente que contenga líneas de comentario.
2. Si ya no lo necesita, borre todo el comentario, incluida la cadena de caracteres introductoria "//".
3. Como alternativa, puede borrar solo la cadena de caracteres introductoria "//". De esta forma el comentario se mantiene y se muestra en la columna "Comentario" de la tabla de forzado permanente.

Nota

Borrado de líneas de comentario

Al borrar líneas de comentario se borran también los idiomas de proyecto configurados para esos comentarios así como las traducciones que pudiera haber.

9.4.4.8 Observar variables en la tabla de forzado permanente

Introducción a la observación de variables en la tabla de forzado permanente

Introducción

En la tabla de forzado permanente se observan las variables de los módulos de entrada y salida configurados en la CPU en función del modo de observación (Página 3157) seleccionado. Para poder observar variables debe existir una conexión online con la CPU.

Opciones para la observación de variables

Para la observación de variables existen las siguientes opciones:

- Observar todos
  Este comando inicia la observación de las variables visibles en la tabla de observación activa, en función del modo de observación seleccionado:
  - En el modo básico está predeterminado el modo de observación "permanente".
  - En el modo avanzado se pueden definir puntos de disparo determinados para la observación de variables.

  Nota

  Si el modo de observación se cambia en el modo avanzado y a continuación se regresa de nuevo al modo básico, el modo de observación ajustado antes también se aplicará en éste último.

- Observar inmediatamente
  Este comando inicia la observación de las variables visibles en la tabla de forzado permanente activa inmediatamente y una sola vez.

Restricciones específicas de la CPU en la observación de variables

Existen las siguientes diferencias específicas de CPU:

- CPU S7-300/400:
  Las CPU de esta familia pueden observar únicamente los primeros 30 caracteres de una cadena.
- CPU S7-1200:
  Las CPU de esta familia pueden observar una cadena de hasta 254 caracteres de tamaño total.

Establecimiento del modo de observación en la tabla de forzado permanente

Introducción

Seleccionando el modo de observación se define el punto de disparo y el tiempo de observación de las variables en la tabla de forzado permanente.

Modos de observación posibles (tiempo de observación)

Existen las opciones siguientes:

- Permanente
- Inicio del ciclo, único
- Fin del ciclo, único
- Inicio del ciclo, permanente
- Fin del ciclo, permanente
- Transición a STOP, única
- Transición a STOP, permanente

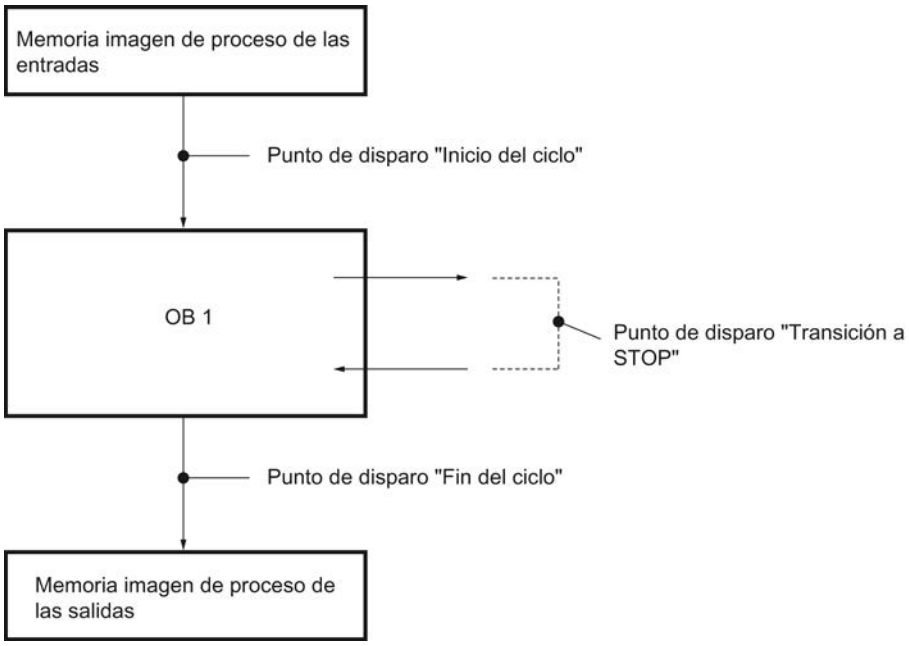
Particularidades del modo "Permanente"

El modo "Permanente" se ejecuta de la forma siguiente para la observación de variables: las entradas se observan al final y las salidas al principio del ciclo.

Definición de un punto de disparo

Los puntos de disparo "Inicio del ciclo", "Fin del ciclo" y "Transición a STOP" definen el momento en que las variables se leerán de la CPU o se actualizarán en la CPU.

La figura siguiente muestra la posición de los puntos de disparo mencionados:



Variables "Observar todos"

Introducción

El comando "Observar todos" permite iniciar la observación de las variables visibles en la tabla de forzado permanente activa. En el modo básico de la tabla de forzado permanente está predeterminada la opción "permanente" para el modo de observación. En el modo avanzado se pueden definir puntos de disparo determinados para la observación de variables. En ese caso, las variables se observan con respecto a los puntos de disparo definidos.

Requisitos

Existe una conexión online con la CPU.

Procedimiento

Para ejecutar el comando "Observar todos", proceda del siguiente modo:

1. Introduzca las variables que desee observar y las direcciones correspondientes en la tabla de forzado permanente.
2. Cambie al modo avanzado haciendo clic en el botón "Mostrar u ocultar todas las columnas del modo avanzado" de la barra de herramientas.
3. Para modificar el modo de observación predeterminado de una variable, haga clic en la celda correspondiente de la columna "Observar con disparo" y seleccione el modo de observación que desee en la lista desplegable.
4. Haga clic en el botón "Observar todos" de la barra de herramientas.

Resultado

Las variables de la tabla de forzado permanente activa se observan en el modo de observación seleccionado en cada caso.

"Observar inmediatamente" variables

Introducción

Con el comando "Observar inmediatamente" se inicia la observación de las variables inmediatamente sin referencia a puntos de disparo determinados. Los valores de las variables se leen una sola vez y se muestran en la tabla de forzado permanente.

Requisitos

Existe una conexión online con la CPU.

Procedimiento

Para ejecutar el comando "Observar inmediatamente", proceda del siguiente modo:

1. Introduzca las variables que desee observar y las direcciones correspondientes en la tabla de forzado permanente.
2. Haga clic en el botón "Observar inmediatamente" de la barra de herramientas.

Resultado

Las variables de la tabla de forzado permanente activa se observan inmediatamente y una sola vez.

9.4.4.9 Forzar permanentemente variables en la tabla de forzado permanente

Introducción al forzado permanente de variables

Introducción

La tabla de forzado permanente permite preasignar valores fijos a distintas variables del programa de usuario. Este procedimiento se conoce como forzado permanente. Solamente se pueden forzar permanentemente entradas de periferia y salidas de periferia.

Para el forzado permanente es necesario que exista una conexión online con la CPU y que la CPU utilizada soporte esta función.

Cuando se abre una tabla de forzado permanente en la carpeta "Tablas de observación y de forzado permanente" bajo una CPU, se muestran automáticamente en esa tabla de forzado permanente todos los valores forzados permanentemente en la CPU asociada, siempre que exista una conexión online con la CPU.

Posibles utilizaciones

La preasignación fija de valores definidos a variables permite preestablecer opciones determinadas y probar así las funciones programadas. El forzado permanente está disponible en el modo básico y en el modo avanzado (Página 3146).

Precaución al forzar variables permanentemente

Antes del forzado permanente es importante que se informe sobre las medidas de seguridad (Página 3162) necesarias para llevar a cabo esta operación.

> ⚠ PELIGRO
>
> **Evitación de daños personales y materiales**
>
> Tenga en cuenta que al ejecutar la función "Forzado permanente", una acción incorrecta puede:
>
> - puede poner en peligro la vida o salud de personas.
> - causar daños materiales en la máquina o en toda la instalación

Opciones para el forzado permanente de variables

Para el forzado permanente de variables existen las siguientes opciones:

- Forzar permanentemente a "0"
  Este comando fuerza la dirección seleccionada en la CPU al valor de forzado permanente "0".
- Forzar permanentemente a "1"
  Este comando fuerza la dirección seleccionada en la CPU al valor de forzado permanente "1".
- Forzar todos permanentemente
  Este comando inicia el forzado permanente de las direcciones activadas en la tabla de forzado permanente activa o reemplaza ininterrumpidamente una orden de forzado existente.
- Finalizar forzado permanente
  Este comando finaliza el forzado permanente de todas las direcciones de la tabla de forzado permanente activa.

Restricciones en el forzado permanente de variables

Observe las siguientes restricciones para el forzado permanente:

- El forzado permanente siempre depende del volumen de operandos de la CPU utilizada.
- Básicamente, sólo se pueden forzar permanentemente entradas de periferia y salidas de periferia.
- Si está activada la función "Desbloquear salidas" en la CPU utilizada, no se podrá efectuar un forzado permanente. Desactive esta función, si así lo desea, en la tabla de observación.

Particularidades del forzado permanente de variables

Tenga en cuenta que el forzado permanente de variables sobrescribe valores en la CPU y que se mantiene al finalizar la conexión online con la CPU.

- **Finalizar forzado permanente**
  La mera finalización de la conexión online no detiene la operación de forzado permanente. Para finalizar el forzado permanente, elija el comando "Online > Forzar permanentemente > Finalizar forzado permanente". Sólo entonces dejan de forzarse permanentemente las variables visibles en la tabla de forzado permanente activa.
- **Finalizar el forzado permanente de variables individuales**
  El comando "Online > Forzar permanentemente > Finalizar forzado permanente" se aplica siempre a todas las variables visibles en la tabla de forzado permanente activa. Para finalizar el forzado permanente de distintas variables, desactive la casilla de verificación del forzado permanente de las variables en cuestión en la tabla de forzado permanente activa y vuelva a iniciar el forzado permanente con el comando "Online > Forzar permanentemente > Forzar todos permanentemente".

Medidas de seguridad para el forzado permanente de variables

Medidas de seguridad para el forzado permanente de variables

Dado que con la función de forzado permanente se interviene permanentemente en el proceso, es necesario observar las siguientes indicaciones:

⚠ PELIGRO

Evitación de daños personales y materiales

Tenga en cuenta que al ejecutar la función "Forzado permanente", una acción incorrecta puede:

- puede poner en peligro la vida o salud de personas.
- causar daños materiales en la máquina o en toda la instalación

⚠ PRECAUCIÓN

Evitación de daños personales y materiales

- Antes de iniciar la función "Forzado permanente" asegúrese de que nadie más vaya a ejecutar simultáneamente esta función en la misma CPU.
- El forzado permanente sólo puede finalizarse haciendo clic en el botón "Finalizar forzado permanente" o con el comando "Online > Forzar permanentemente > Finalizar forzado permanente". Cerrando la tabla de forzado permanente activa no se finaliza el forzado permanente.
- El "forzado permanente" no puede deshacerse.
- Infórmese sobre las diferencias entre el "forzado de variables (Página 3135)" y el "forzado permanente de variables (Página 3160)".
- Si una CPU no soporta la función de "forzado permanente", los botones relacionados con esa función no estarán disponibles.
- Si está activada la función "Desbloquear salidas de periferia" en la CPU utilizada, no se podrá efectuar el forzado permanente en esa CPU. Desactive esta función, si así lo desea, en la tabla de observación.

Forzar permanentemente variables a "0"

Introducción

La función de forzado permanente permite asignar valores fijos a las distintas variables de un programa de usuario.

Precaución al forzar variables permanentemente

Antes de forzar variables permanentemente, infórmese sin falta sobre las medidas de seguridad que deben tomarse para este procedimiento (Página 3162).

⚠ PELIGRO

¡Deben evitarse daños personales y materiales

Tenga en cuenta que al ejecutar la función "Forzado permanente", una acción incorrecta puede:

- puede poner en peligro la vida o salud de personas.
- causar daños materiales en la máquina o en toda la instalación

Requisitos

- Existe una conexión online con la CPU.
- La CPU utilizada soporta el forzado permanente.
- La función "Desbloquear salidas" no está activada en la CPU en la que se debe forzar permanentemente. Desactive esta función, si así lo desea, en la tabla de observación.

Procedimiento

Para forzar permanentemente variables a "0", proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Introduzca la dirección en la tabla de forzado permanente.
3. Elija el comando "Online > Forzar permanentemente > Forzar permanentemente a 0" para forzar permanentemente la dirección seleccionada al valor predefinido.
4. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

La dirección seleccionada se fuerza permanentemente a "0". El triángulo amarillo ya no aparece. En la primera columna aparece p.ej., una "F" roja que indica que se está realizando el forzado permanente.

Finalizar el forzado permanente

Para finalizar el forzado permanente, proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Haga clic en el comando "Online > Forzar permanentemente > Finalizar forzado permanente".
3. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

Finaliza el forzado permanente de los valores seleccionados. La "F" roja que aparecía en la primera columna desaparece y el triángulo amarillo vuelve a aparecer detrás de la casilla de

verificación indicando que la dirección está seleccionada para el forzado permanente, pero que en ese momento no se está forzando.

Nota

Al forzar permanentemente, tenga en cuenta lo siguiente:

- El forzado permanente no puede deshacerse.
- Finalizando la conexión online no se finaliza el forzado permanente.
- Para finalizar el forzado permanente, es necesario que la dirección forzada de forma permanente sea visible en la tabla de forzado permanente activa.

Forzar permanentemente variables a "1"

Introducción

La función de forzado permanente permite asignar valores fijos a las distintas variables de un programa de usuario.

Precaución al forzar variables permanentemente

Antes de forzar variables permanentemente, infórmese sin falta sobre las medidas de seguridad que deben tomarse para este procedimiento (Página 3162).

⚠ PELIGRO

Evitación de daños personales y materiales

Tenga en cuenta que al ejecutar la función "Forzado permanente", una acción incorrecta puede:

- poner en peligro la vida o la salud de personas
- causar daños materiales en la máquina o en toda la instalación

Requisitos

- Existe una conexión online con la CPU.
- La CPU utilizada soporta el forzado permanente.
- La función "Desbloquear salidas" no está activada en la CPU en la que se debe forzar permanentemente. Desactive esta función, si así lo desea, en la tabla de observación.

Procedimiento

Para forzar permanentemente variables a "1", proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Introduzca la dirección en la tabla de forzado permanente.
3. Elija el comando "Online > Forzar permanentemente > Forzar permanentemente a 1" para forzar permanentemente la dirección seleccionada al valor predefinido.
4. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

La dirección seleccionada se fuerza permanentemente a "1". El triángulo amarillo ya no aparece. En la primera columna aparece p.ej., una "F" roja que indica que se está realizando el forzado permanente.

Finalizar el forzado permanente

Para finalizar el forzado permanente, proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Haga clic en el comando "Online > Forzar permanentemente > Finalizar forzado permanente".
3. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

Finaliza el forzado permanente de los valores seleccionados. La "F" roja que aparecía en la primera columna desaparece y el triángulo amarillo vuelve a aparecer detrás de la casilla de verificación indicando que la dirección está seleccionada para el forzado permanente, pero que en ese momento no se está forzando.

Nota

Al forzar permanentemente, tenga en cuenta lo siguiente:

- El forzado permanente **no** puede deshacerse.
- Finalizando la conexión online **no** se finaliza el forzado permanente.
- Para finalizar el forzado permanente, es necesario que la dirección forzada de forma permanente sea visible en la tabla de forzado permanente activa.

Forzar permanentemente variables "Forzar todos permanentemente"

Introducción

La función de forzado permanente permite asignar valores fijos a las distintas variables de un programa de usuario.

Si ya hay una operación de forzado permanente activa, ésta puede sustituirse sin interrupción con el comando "Online > Forzar permanentemente > Forzar todos permanentemente". Dejarán de forzarse permanentemente las direcciones forzadas permanentemente que no estén activadas.

Precaución al forzar variables permanentemente

Antes de forzar variables permanentemente, infórmese sin falta sobre las medidas de seguridad que deben tomarse para este procedimiento (Página 3162).

⚠ PELIGRO

¡Deben evitarse daños personales y materiales!

Tenga en cuenta que al ejecutar la función "Forzado permanente", una acción incorrecta puede:

- puede poner en peligro la vida o salud de personas.
- causar daños materiales en la máquina o en toda la instalación

Requisitos

- Existe una conexión online con la CPU.
- La CPU utilizada soporta el forzado permanente.
- La función "Desbloquear salidas" **no** está activada en la CPU en la que se debe forzar permanentemente. Desactive esta función, si así lo desea, en la tabla de observación.

Procedimiento

Para forzar variables permanentemente con el comando "Online > Forzado permanente > Forzar todos permanentemente", proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Introduzca las direcciones y los valores de forzado permanente en la tabla de forzado permanente.
3. Seleccione las direcciones que desee forzar permanentemente activando la casilla de verificación de forzado permanente de la columna después del "Valor de forzado permanente".
   Junto a la casilla de verificación activada aparece un triángulo amarillo que indica que la dirección está seleccionada para el forzado permanente, pero que todavía no está siendo forzada permanentemente.

4. Elija el comando "Online > Forzar permanentemente > Forzar todos permanentemente" para forzar permanentemente las direcciones seleccionadas con los valores predefinidos.
5. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

Las direcciones seleccionadas se fuerzan permanentemente a los valores predefinidos. El triángulo amarillo ya no aparece. En la primera columna aparece p.ej., una "F" roja que indica que se está realizando el forzado permanente.

Finalizar el forzado permanente

Para finalizar el forzado permanente, proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Haga clic en el comando "Online > Forzar permanentemente > Finalizar forzado permanente".
3. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

Finaliza el forzado permanente de las direcciones seleccionadas. La "F" roja que aparecía en la primera columna desaparece y el triángulo amarillo vuelve a aparecer detrás de la casilla de verificación indicando que la dirección está seleccionada para el forzado permanente, pero que en ese momento no se está forzando.

Nota

Al forzar permanentemente, tenga en cuenta lo siguiente:

- El forzado permanente **no** puede deshacerse.
- Finalizando la conexión online **no** se finaliza el forzado permanente.
- Para finalizar el forzado permanente, es necesario que la dirección forzada de forma permanente sea visible en la tabla de forzado permanente activa.

9.4.4.10 Finalizar forzado permanente de variables

Finalizar forzado permanente de todas las variables

Introducción

Observe las siguientes indicaciones antes de finalizar el forzado permanente de variables:

- La finalización del forzado permanente **no** puede deshacerse.
- Finalizando la conexión online **no** se finaliza el forzado permanente.
- Para finalizar el forzado permanente, es necesario que la dirección forzada de forma permanente sea visible en la tabla de forzado permanente activa.

Precaución al forzar variables permanentemente

Antes de forzar variables permanentemente, infórmese sin falta sobre las medidas de seguridad que deben tomarse para este procedimiento (Página 3162).

> ⚠ **PELIGRO**
>
> **¡Deben evitarse daños personales y materiales!**
>
> Tenga en cuenta que al salir de la función "Forzado permanente", una acción incorrecta puede:
>
> - puede poner en peligro la vida o salud de personas.
> - causar daños materiales en la máquina o en toda la instalación

Requisitos

- En una tabla de forzado permanente se fuerzan variables con carácter permanente.
- Existe una conexión online con la CPU.
- La CPU utilizada soporta el forzado permanente.
- La función "Desbloquear salidas" no está activada en la CPU en la que se debe forzar permanentemente. Desactive esta función, si así lo desea, en la tabla de observación.

Procedimiento

Para finalizar el **forzado permanente de todas las variables**, proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Elija el comando "Online > Forzar permanentemente > Finalizar forzado permanente" para finalizar el forzado permanente de las direcciones mostradas.
3. Confirme el cuadro de diálogo "Finalizar forzado permanente" con "Sí".

Resultado

Finaliza el forzado permanente de todas las variables. La "F" roja que aparecía en la primera columna desaparece y el triángulo amarillo vuelve a aparecer detrás de la casilla de verificación indicando que la dirección está anotada para el forzado permanente, pero que en ese momento no se está forzando.

Finalizar forzado permanente de variables individuales

Introducción

Observe las siguientes indicaciones antes de finalizar el forzado permanente de variables:

- La finalización del forzado permanente **no** puede deshacerse.
- Finalizando la conexión online **no** se finaliza el forzado permanente.
- Para finalizar el forzado permanente, es necesario que la dirección forzada de forma permanente sea visible en la tabla de forzado permanente activa.

Precaución al forzar variables permanentemente

Antes de forzar variables permanentemente, infórmese sin falta sobre las medidas de seguridad que deben tomarse para este procedimiento (Página 3144).

⚠ PELIGRO

¡Deben evitarse daños personales y materiales!

Tenga en cuenta que al salir de la función "Forzado permanente", una acción incorrecta puede:

- puede poner en peligro la vida o salud de personas.
- causar daños materiales en la máquina o en toda la instalación

Requisitos

- En una tabla de forzado permanente se fuerzan variables con carácter permanente.
- Existe una conexión online con la CPU.
- La CPU utilizada soporta el forzado permanente.
- La función "Desbloquear salidas" no está activada en la CPU en la que se debe forzar permanentemente. Desactive esta función, si así lo desea, en la tabla de observación.

Procedimiento

Para finalizar el **forzado permanente de variables individuales**, proceda del siguiente modo:

1. Abra la tabla de forzado permanente.
2. Desactive la casilla de verificación de las direcciones que ya no deben forzarse permanentemente.
3. Seleccione de nuevo el comando "Online > Forzar permanentemente".

Resultado

Finaliza el forzado permanente de las direcciones desactivadas. La "F" roja que aparecía en la primera columna desaparece y el triángulo amarillo vuelve a aparecer detrás de la casilla de verificación indicando que la dirección está anotada para el forzado permanente, pero que en ese momento no se está forzando.

9.5 Configurar avisos

9.5.1 Introducción a la configuración de avisos

Sinopsis

En los sistemas de automatización, los avisos permiten detectar rápidamente errores durante la ejecución del proceso, localizarlos con precisión y solucionarlos. De esta forma es posible reducir sustancialmente los tiempos de parada de una instalación.

Antes de poder visualizar avisos, es necesario configurarlos.

Es posible crear avisos dependientes de eventos y asignarles textos y atributos, así como editarlos, traducirlos y visualizarlos en visualizadores.

En la siguiente tabla se muestra una relación de los tipos de aviso con una breve descripción de su función.

Tipo de aviso	Descripción
Avisos de programa	Los avisos de programa sirven para notificar eventos síncronos y cada uno de ellos está asignado a un bloque. Se crean en el editor de programas y se editan en el editor de avisos.
Avisos de diagnóstico del sistema	Los avisos de diagnóstico del sistema son eventos de módulo que dependen de la configuración y se activan o desactivan en la configuración hardware. En el editor de avisos pueden visualizarse pero no editarse.
Avisos de diagnóstico de usuario	Los avisos de diagnóstico de usuario permiten registrar una entrada de usuario en el búfer de diagnóstico y enviar un aviso al respecto. Están asignados a una CPU. Se crean en el editor de avisos y pueden editarse aquí.

9.5.2 Asignación de números de aviso

Asignación de números

Los avisos se identifican con un número único en toda la CPU. De este modo no es necesario volver a compilar tras copiar programas enteros. Una excepción constituye la copia de bloques individuales. En este caso, es necesario volver a compilar para integrar los números de aviso modificados en el programa.

9.5.3 Componentes de un aviso

Sinopsis

El modo en que se visualice un aviso dependerá del método, del bloque de aviso utilizado y del visualizador.

La tabla siguiente muestra una relación de los posibles componentes:

Componente	Descripción
Sello de tiempo	Se genera en el sistema de automatización al producirse el evento de aviso.
Estado del aviso	Son posibles: aparecido, desaparecido, desaparecido sin acusar, desaparecido con acuse.
Valor asociado	Algunos avisos pueden acompañarse de un valor de proceso que puede ser evaluado por la instrucción de aviso utilizada.
Imagen	En caso de caída del sistema, los avisos producidos pueden visualizarse posteriormente en dispositivos HMI.
Número de aviso	Número único en la CPU que es asignado por el sistema y que identifica un aviso.
Textos de aviso	Son configurados por el usuario.

9.5.4 Bloques de aviso disponibles

S7-300/400: Vista general de los bloques de aviso

Es posible elegir entre los siguientes bloques de aviso:

- ALARM
- ALARM_8
- ALARM_8P
- NOTIFY
- ALARM_S
- ALARM_SQ
- AR_SEND (para enviar ficheros; no es posible configurar textos ni atributos de aviso)
- NOTIFY_8P
- ALARM_DQ
- ALARM_D

S7-300/400: ¿Cuándo se utilizan los bloques de aviso?

La siguiente tabla le ayudará a seleccionar el bloque de aviso apropiado para una determinada aplicación. La elección del bloque de aviso depende de los siguientes factores:

- Del número de canales disponibles en el bloque y, con ello, el número de señales que se supervisan en cada llamada de bloque.
- De la posibilidad de acusar avisos.
- De la posibilidad de incluir valores asociados.
- De los visualizadores que se utilicen.
- Del alcance de la CPU.

Bloque de aviso	Canales	Acuse	Valores asociados	Particularidades
ALARM	1	sí	hasta 10	Envía un aviso en caso de flanco entrante o saliente, respectivamente.
ALARM_8	8	sí	no	Envía un aviso en caso de flanco entrante o saliente de una o varias señales.
ALARM_8P	8	sí	hasta 10	como ALARM_8
NOTIFY	1	no	hasta 10	como ALARM
NOTIFY_8P	8	no	hasta 10	como ALARM

Bloque de aviso	Canales	Acuse	Valores asociados	Particularidades
AR_SEND	1	-	-	Sirve para enviar un fichero; no es posible configurar textos ni atributos de aviso.
ALARM_SQ	1	sí	1	Se genera un aviso con cada llamada de bloque y al cambiar la señal con respecto a la llamada anterior.
ALARM_S	1	no	1	como ALARM_SQ
ALARM_DQ	1	sí	1	como ALARM_SQ
ALARM_D	1	no	1	como ALARM_SQ

S7-1500: Vista general de los bloques de aviso

Se dispone del siguiente bloque de aviso:

- Program_Alarm

La selección de la categoría de aviso correspondiente determina si el bloque requiere o no acuse. El ajuste predeterminado es que el bloque no requiere acuse.

Bloque de aviso	Canales	Acuse	Valores asociados	Particularidades
Program_Alarm	1	posible	hasta 10	Envía un aviso con sello de tiempo en caso de flanco entrante o saliente, respectivamente.

9.5.5 Plantilla y avisos

Con ayuda de los editores de programas y avisos es posible crear una plantilla (p. ej. un FB como plantilla para DBs de instancia) o avisos (p. ej. DBs de instancia a partir de la plantilla) en varios pasos de trabajo.

El bloque de aviso puede ser un FB o un DB de instancia.

- En un FB se crea una plantilla que sirve de template para avisos. Todas las entradas que se realicen para la plantilla se aplicarán automáticamente en los avisos derivados de la misma. Si se asigna un DB de instancia al FB, se generarán automáticamente avisos según la plantilla de aviso para el DB de instancia y se asignarán números de aviso.
- En un de DB de instancia se pueden modificar los avisos generados a partir de la plantilla específicamente para cada instancia.

La diferencia visible es que en el caso de los avisos se asignan números de aviso, mientras que en el caso de las plantillas no se asignan. Las plantillas y las instancias correspondientes se disponen una bajo la otra en el editor de avisos.

Bloqueo de los datos en la plantilla

Con el editor de avisos se introducen los textos y atributos de los avisos dependientes de eventos. Para ello se puede determinar, p. ej., cómo se mostrarán los avisos en determinados visualizadores (p. ej. según la categoría). Para facilitar la creación de avisos, las plantillas se utilizan como template.

- Si se introducen datos (atributos y textos) para la plantilla, es posible determinar si se deberán inhibir o no. En caso de atributos inhibidos, junto al campo de entrada aparecerá el símbolo de un eslabón cerrado. En el caso de los atributos no inhibidos, el eslabón estará abierto.
- Los datos inhibidos en la plantilla no se pueden modificar en los avisos específicos de la instancia. Los datos sólo se visualizan.
- No obstante, si es necesario realizar modificaciones, es necesario regresar a la plantilla, inhabilitar el bloqueo y realizar los cambios. En cualquier caso, los cambios no se aplicarán a las instancias generadas antes de la modificación.

Cambio de los datos en la plantilla

Si se cambian datos a posteriori en la plantilla, estos cambios se aplicarán automáticamente a las instancias.
Excepciones: Estos datos se han modificado previamente en la instancia o se han inhibido o desbloqueado datos a posteriori en la plantilla.

Nota

Si se copian instancias a otro programa sin copiar la plantilla, la instancia no se mostrará completa. En tal caso también habrá que copiar la plantilla al nuevo programa.

Restablecimiento de datos específicos de la instancia al valor de la plantilla

Si se han sobrescrito atributos o textos en la instancia del aviso, el símbolo de la plantilla se mostrará junto al atributo. Por cada atributo puede determinarse si se volverá a utilizar el valor de la plantilla. En tal caso no se muestra ningún símbolo de la plantilla.

Consulte también

Crear y editar la plantilla (Página 3180)

Crear y editar DBs de instancia (Página 3180)

9.5.6 Parámetros formales, tipos de datos de avisos y bloques de aviso

Parámetros formales como entrada de número de aviso

Por cada aviso o grupo de avisos se requiere en el programa un parámetro formal (nombre del aviso) que se indica como parámetro IN en la vista general de variables del programa. El parámetro formal se utiliza como entrada de números de aviso y constituye la base de un aviso.

Suministro del tipo de datos adecuado al parámetro formal

Al parámetro formal debe asignársele un tipo de datos de aviso apropiado para el bloque de aviso utilizado.

Tipos de datos de avisos y sus bloques de aviso

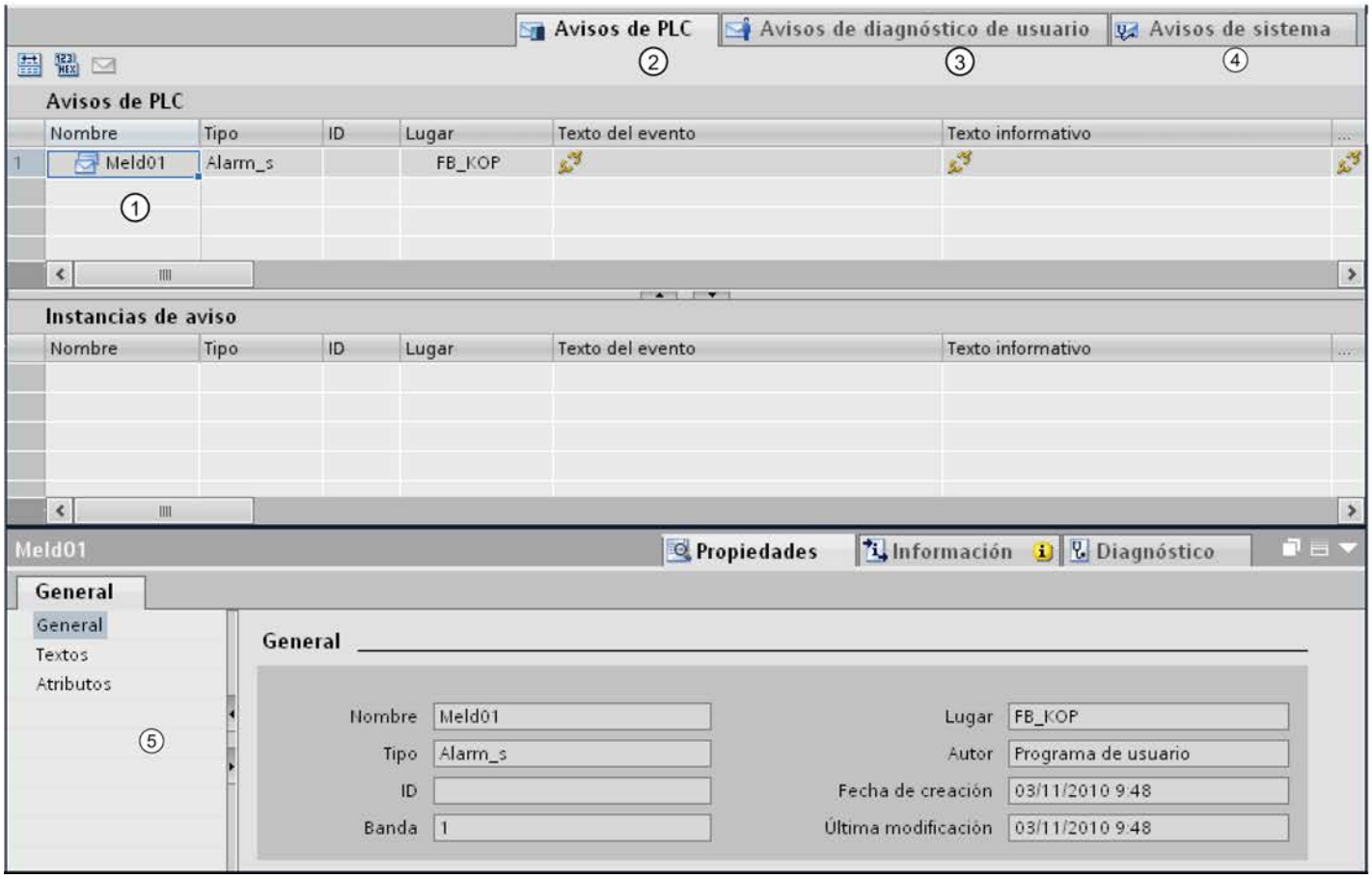
La siguiente tabla muestra los tipos de datos de avisos con sus bloques de aviso y sus propiedades. Los valores de los tipos de datos llevan el mismo nombre que los bloques de aviso (a excepción de "alarm_s") y cuentan con el prefijo "C_".

Sistema SIMATIC	Tipo de datos	Bloque de aviso	Propiedades
S7-400	C_Alarm_8	ALARM_8	8 canales, con acuse, sin valores asociados
S7-400	C_Alarm_8p	ALARM_8P	8 canales, con acuse, hasta 10 valores asociados por canal
S7-400	C_Notify	NOTIFY	1 canal, con acuse, hasta 10 valores asociados
S7-400	C_Alarm	ALARM	1 canal, con acuse, hasta 10 valores asociados
S7-300/400	C_Alarm_s	ALARM_S	1 canal, sin acuse, hasta 1 valor asociado
S7-300/400	C_Alarm_s	ALARM_SQ	1 canal, con acuse, hasta 1 valor asociado
S7-400	C_Ar_Send	AR_SEND	Sirve para enviar un fichero
S7-400	C_Notify_8p	NOTIFY_8P	8 canales, sin acuse, hasta 10 valores asociados
S7-300/400	C_Alarm_s	ALARM_DQ	1 canal, con acuse, hasta 1 valor asociado
S7-300/400	C_Alarm_s	ALARM_D	1 canal, sin acuse posible, hasta 1 valor asociado
S7-1500	Program_Alarm	Program_Alarm	1 canal, posibilidad de acuse por parametrización, hasta 10 valores asociados

9.5.7 Estructura del editor de avisos

Estructura del editor de avisos

La figura siguiente muestra a modo de ejemplo los componentes del editor de avisos:



1		Representación en forma de tabla de los avisos en el área de trabajo, consulte: Auto-Hotspot La tabla superior contiene el tipo de aviso; la tabla inferior contiene las instancias derivadas del tipo de aviso (FB) (los DB de instancia).
2		Ficha "Avisos de programa": los avisos de programa se pueden editar aquí.
3		Ficha "Avisos de sistema" Los avisos de sistema solo pueden verse, no editarse. Para editar estos avisos hay que cambiar a la vista de dispositivos (comando de menú "Ir a dispositivo") y adaptar allí los avisos de sistema en la ventana de inspección.

4		Ficha "Avisos de diagnóstico de usuario": aquí se pueden crear y editar avisos de diagnóstico de usuario. En las CPU S7-1500, la función no está disponible.
5		Auto-Hotspot

Los parámetros, textos y atributos necesarios se pueden introducir y modificar tanto en la tabla como en la ventana de inspección.

9.5.8 Crear y editar avisos

9.5.8.1 Crear avisos de programa

Requisitos

Se ha creado un bloque de función.

Procedimiento

Para crear un aviso de programa, proceda del siguiente modo:

1. En la carpeta "Bloques de programa" del árbol del proyecto, seleccione el bloque de función (FB) para el que desee crear un aviso de programa y abra este bloque haciendo doble clic.
2. Rellene la interfaz del bloque. Por cada bloque de aviso que se llame en el FB deberán declararse variables en el FB que efectúa la llamada.
   Para ello, introduzca por ejemplo las siguientes variables:
   – En el parámetro "IN", un nombre para la entrada del bloque de aviso, p. ej. "Aviso01" (para la entrada de aviso 01) y el tipo de datos.
3. En la ventana de instrucciones del FB, introduzca la llamada del bloque de aviso seleccionado, p. ej., "CALL ALARM_S", y finalice la entrada con INTRO.
   Resultado: En la sección de instrucciones del FB se muestran las variables de entrada del bloque de aviso llamado, en este caso el bloque ALARM_S.
4. Asigne a la variable "EV_ID" el nombre asignado en el paso 2 a la entrada del bloque de aviso, en este caso, "Aviso01".

   Nota

   Si en lugar de un bloque de aviso existente en la CPU se llama un FB con multiinstancias en el que también haya avisos configurados, en el bloque que efectúa la llamada también habrá que configurar los avisos del FB con multiinstancias.

5. Repita los pasos 2 a 4 para todas las llamadas de bloques de aviso en este FB.

9.5.8.2 Editar avisos de programa en el editor de avisos

Requisitos

Se ha creado un aviso de programa.

Procedimiento

Para editar avisos de programa, proceda del siguiente modo:

1. Haga doble clic en "Avisos de PLC" en el árbol del proyecto. Se abre el editor de avisos.
2. Introduzca los textos y atributos deseados en las columnas correspondientes.

Nota

Al editar tipos de avisos, se pueden bloquear textos y atributos. Para ello, haga clic en el símbolo que precede a la columna correspondiente.

Si se editan avisos en DBs de instancia en los que los textos o atributos del tipo de aviso no estén bloqueados, aparecerá un símbolo del tipo delante de la columna correspondiente. Al hacer clic en dicho símbolo, los textos o atributos modificados se restablecen al valor del tipo de aviso.

9.5.8.3 Editar avisos de programa en el editor de programación

Requisitos

Se ha creado un aviso de programa.

Procedimiento

Para editar avisos de programa, proceda del siguiente modo:

1. Seleccione la fila correspondiente en la interfaz del bloque.
2. Abra la ficha "Aviso" en la ventana de inspección y seleccione el grupo deseado.
3. Introduzca los textos y atributos deseados en los campos correspondientes.

Nota

Al editar tipos de avisos, se pueden bloquear textos y atributos. Para ello, haga clic en el símbolo junto al campo correspondiente.

Si se editan avisos en DBs de instancia en los que los textos o atributos del tipo de aviso no estén bloqueados, aparecerá un símbolo del tipo junto al campo correspondiente. Al hacer clic en dicho símbolo, los textos o atributos modificados se restablecen al valor del tipo de aviso.

9.5.8.4 Borrar avisos de programa

Procedimiento

Para borrar un aviso de programa, proceda del siguiente modo:

1. Abra el bloque que contenga el aviso que desee borrar.
2. Seleccione la fila correspondiente en la interfaz de bloque y elija el comando "Borrar" del menú contextual.

Resultado

Se borra el aviso.

9.5.8.5 Crear y editar la plantilla

Procedimiento

Para editar una plantilla, proceda del siguiente modo:

1. Seleccione el bloque de aviso deseado.
2. Introduzca en las columnas correspondientes los textos deseados o seleccione los atributos deseados.
   Si ha seleccionado un bloque de aviso de varios canales (p. ej. "ALARM_8"), podrá asignar a cada subnúmero sus propios textos y asignarles atributos parcialmente propios.
3. Si no desea que los textos o atributos se puedan cambiar en la instancia, inhíbalos en la plantilla.

9.5.8.6 Crear y editar DBs de instancia

Requisitos

Previamente se debe haber creado un FB y haber creado al menos un aviso en él.

Procedimiento

Para asignar bloques de datos de instancia (DBs) a una plantilla y editar los avisos de dichos DBs para cada instancia, proceda del siguiente modo:

1. En el árbol del proyecto, haga doble clic en "Agregar nuevo bloque"; en el cuadro de diálogo que se abrirá, haga clic en el botón "Bloque de datos (DB)" y seleccione en la lista desplegable de "Plantilla" el bloque de función (plantilla) a la que desee asignar el bloque de datos de instancia.
2. Ahora, en la ventana de inspección haga clic en la ficha "Aviso" y seleccione el grupo deseado.
   O bien:
   En el árbol del proyecto, haga clic en "Avisos de PLC" para abrir la configuración de avisos.
3. Lleve a cabo los cambios deseados del DB de instancia correspondiente.

Nota

Si las propiedades de los DBs de instancia están protegidas contra escritura, primero deberá desbloquear la plantilla (FB).

Resultado

Así finaliza la configuración de avisos del DB de instancia seleccionado.

9.5.8.7 Crear avisos de diagnóstico de usuario

Los avisos de diagnóstico de usuario están asignados a una CPU. Se crean y editan en el editor de avisos.

Procedimiento

Para crear un aviso de diagnóstico de usuario, proceda del siguiente modo:

1. Haga doble clic en "Avisos de PLC" en la navegación del proyecto para abrir el editor de avisos.
2. En el editor de avisos, seleccione la ficha "Avisos de diagnóstico de usuario".
3. Haga clic en la tabla y seleccione en el menú contextual el comando "Insertar nuevo aviso".

Resultado

Se ha creado un aviso de diagnóstico de usuario.

9.5.8.8 Editar avisos de diagnóstico de usuario

Requisitos

Se ha creado un aviso de diagnóstico de usuario.

El editor de avisos está abierto.

Procedimiento

Para editar un aviso de diagnóstico de usuario, proceda del siguiente modo:

1. Introduzca los textos y atributos deseados en las columnas correspondientes.

9.5.8.9 Borrar avisos de diagnóstico de usuario

Es posible borrar un aviso marcado. Los textos configurados para dicho aviso se borrarán y el número de aviso quedará disponible.

Procedimiento

Para borrar un aviso de diagnóstico de usuario, proceda del siguiente modo:

1. Marque la fila correspondiente en la tabla y seleccione en el menú contextual el comando "Borrar".

Resultado

El aviso se borra. Ya no aparece en la tabla.

9.5.8.10 Insertar valores asociados en los avisos

Para agregar información actualizada (p. ej., del proceso) a los avisos, es posible agregar valores asociados en cualquier punto del texto de un aviso.

Procedimiento

Para insertar un valor asociado en un aviso, proceda del siguiente modo:

SFB de alarma del S7-400 (p. ej. Alarm8)

1. Cree un bloque con la siguiente estructura:
   @<n.º del valor asociado>%<tipo de elemento><formato>@.
2. Inserte este bloque en aquel punto del texto del aviso en el que quiera que aparezca el valor asociado.

SFC de alarma del S7-300/400 (p. ej. Alarm_S)

La longitud máxima de un valor asociado en el AlarmS/SQ es de 12 bytes como máximo

1. Cree un bloque con la siguiente estructura:
   @<índice>%<tipo de elemento><formato>@.
   Índice: Índice en el valor asociado. El índice se interpreta como índice de matriz empezando por 1.
   El offset de byte se calcula con la siguiente fórmula:
   Offset = ((índice - 1) * ancho de datos (tipo de elemento))
   Utilice en lo posible solo Y,W,X, R como tipo de elemento. El tipo de elemento solo es necesario para determinar el ancho. El modo de representación se define al indicar el formato.
2. Inserte este bloque en aquel punto del texto del aviso en el que quiera que aparezca el valor asociado.

Consulte también

Estructura de los valores asociados (Página 3183)

Ejemplos de valores asociados (Página 3184)

9.5.8.11 Estructura de los valores asociados

Los valores asociados están formados por los siguientes componentes:

Tipo de elemento

Con ello se configura de forma unívoca el tipo de datos del valor asociado:

Tipo de elemento	Tipo de datos
Y	BYTE
W	WORD
X	DWORD
I	Integer
D	DINT
B	BOOL
C	CHAR
R	REAL

El tipo de elemento sólo convierte en unívoco el tipo de datos transferido desde el sistema de automatización. No se utiliza como Casting Operator.

Formato

Determine el formato de visualización del valor asociado en el visualizador. El formato irá precedido del carácter "%". Existen los siguientes formatos para los textos de aviso:

Formato	Descripción
%[i]X	Hexadecimal con i posiciones
%[i]u	Número decimal sin signo con i posiciones
%[i]d	Número decimal con signo con i posiciones
%[i]b	Número binario con i posiciones
%[i][.y]f	Número en coma flotante con signo e y posiciones tras el punto decimal e i posiciones totales
%[i]s	Cadena de caracteres (ANSI String) con i posiciones Los caracteres se imprimen hasta el primer byte 0 (00Hex).
%t#<Nombre de la librería de textos>	Acceso a la librería de textos

Aunque el formato sea demasiado pequeño, el valor se emitirá con su longitud total.

Si el formato es demasiado grande, delante del valor se visualizará un número apropiado de espacios en blanco.

Si el formato indicado es demasiado grande (en caso de números decimales o en coma flotante) y si se emite un signo, se reduce el número de espacios en una posición. Lo mismo es aplicable al punto o la coma en números en coma flotante.

Ejemplo: Valor=1234.567 Formato=%3.8f Salida=1234.567; es decir, no se emite ningún espacio aunque solo haya 7 cifras.

Nota

Tenga en cuenta que se puede introducir "[i]" de forma opcional, debiendo omitir para ello los paréntesis cuadrados.

Consulte también

Insertar valores asociados en los avisos (Página 3182)

Ejemplos de valores asociados (Página 3184)

9.5.8.12 Ejemplos de valores asociados

Ejemplos de valores asociados:

@1I%6d@: El valor del valor asociado 1 se representa como número decimal con un máximo de 6 posiciones.

@2R%6f@: El valor "5.4" p. ej. a partir del valor asociado 2 se representa como número en coma fija "5.4" (comenzando con tres espacios en blanco).

@2R%2f@: El valor "5.4" p. ej. a partir del valor asociado 2 se representa como número en coma fija "5.4" (no se corta aunque el número de posiciones sea demasiado pequeño).

@1W%t#Textbib1@: El valor asociado 1 del tipo WORD es el índice con el que se referenciará en la librería de textos Textbib1 el texto que se insertará.

Ejemplos de indicaciones de formato y valores emitidos:

Valor=255 Formato=%5X Salida="000FF"

Valor=255 Formato=%5X Salida="000FF"

Valor=123 Formato=%5u Salida=" 123"

Valor=-200 Formato=%4d Salida="-200"

Valor=19 Formato=%8b Salida="2#00010011"

Valor=1234.567 Formato=%8.3f Salida="1234.567"; es decir, no se emite ningún espacio aunque tan solo haya 7 cifras

Nota

Si se desea transferir a los bloques ALARM_S más de un valor asociado, es posible transferir una matriz con una longitud máxima de 12 bytes. Puede tratarse, p. ej., de un máximo de 12 bytes o Char, un máximo de 6 Word o Int, o bien un máximo de 3 DWord, Real o DInt.

Consulte también

Insertar valores asociados en los avisos (Página 3182)

Estructura de los valores asociados (Página 3183)

9.5.8.13 Borrar valores asociados

Es posible borrar valores asociados borrando la cadena que representa el valor asociado en el texto de aviso.

Procedimiento:

Para borrar valores asociados, proceda de la manera siguiente:

1. En el texto de aviso, busque el bloque equivalente al valor asociado que desee borrar. El bloque comienza con el carácter "@", seguido del identificador de lugar por el que se puede reconocer el valor asociado, así como un formato, y finaliza con el carácter "@".
2. Borre del texto de aviso el bloque que haya encontrado.

9.5.9 Crear y editar avisos

9.5.9.1 Crear avisos de programa

Requisitos

Se ha creado un bloque de función.

Procedimiento

Para crear un aviso de programa, proceda del siguiente modo:

1. En la carpeta "Bloques de programa" del árbol del proyecto, seleccione el bloque de función (FB) para el que desee crear un aviso de programa y abra este bloque haciendo doble clic.
2. Inserte en la ventana de instrucciones del FB la llamada para el bloque de aviso seleccionado ("Program_Alarm") y finalice la entrada con INTRO. La información necesaria se introduce automáticamente en el área "Static" de la interfaz del bloque.
   Resultado: En la sección de instrucciones del FB se muestran las variables de entrada del bloque de aviso llamado, en este caso el bloque Program_Alarm.
3. Repita los pasos 3 y 4 para todas las llamadas de bloques de aviso en este FB.

Tenga en cuenta que al crear avisos en las CPU S7-1500 es necesario indicar la categoría de aviso. Esta especifica si el aviso requiere acuse o no. Los avisos que no requieren acuse pueden tener solo fines informativos.

Según esto existen tres tipos de avisos:

- El aviso debe acusarse y tiene el estado "entrante" o "saliente".
- El aviso no tiene que ser acusado y tiene el estado "entrante" o "saliente".
- El aviso tiene solo fines informativos. Debe haber una marca en la columna "Solo información".

Nota

La configuración estándar es sin acuse y con el estado "entrante" o "saliente".

Consulte también

Crear categorías (Página 3197)

9.5.9.2 Editar avisos de programa en el editor de avisos

Requisitos

Se ha creado un aviso de programa.

Procedimiento

Para editar avisos de programa, proceda del siguiente modo:

1. Haga doble clic en "Avisos de PLC" en el árbol del proyecto. Se abre el editor de avisos.
2. Introduzca los textos y atributos deseados en las columnas correspondientes.

Nota

Al editar tipos de avisos, se pueden bloquear textos y atributos. Para ello, haga clic en el símbolo que precede a la columna correspondiente.

Si se editan avisos en DBs de instancia en los que los textos o atributos del tipo de aviso no estén bloqueados, aparecerá un símbolo del tipo delante de la columna correspondiente. Al hacer clic en dicho símbolo, los textos o atributos modificados se restablecen al valor del tipo de aviso.

9.5.9.3 Editar avisos de programa en el editor de programación

Requisitos

Se ha creado un aviso de programa.

Procedimiento

Para editar avisos de programa, proceda del siguiente modo:

1. Seleccione la fila correspondiente en la interfaz del bloque.
2. Abra la ficha "Aviso" en la ventana de inspección y seleccione el grupo deseado.
3. Introduzca los textos y atributos deseados en los campos correspondientes.

Nota

Al editar tipos de avisos, se pueden bloquear textos y atributos. Para ello, haga clic en el símbolo junto al campo correspondiente.

Si se editan avisos en DBs de instancia en los que los textos o atributos del tipo de aviso no estén bloqueados, aparecerá un símbolo del tipo junto al campo correspondiente. Al hacer clic en dicho símbolo, los textos o atributos modificados se restablecen al valor del tipo de aviso.

Bloque de aviso "Program_alarm"

Cuando se utiliza el bloque de aviso "Program_alarm", en la ventana de inspección se muestra la ficha adicional "Configuración". En los canales de entrada vacíos SD_1 a SD_10 se pueden introducir valores asociados adicionales que, por ejemplo, no aparecen en los textos.

Para ello, haga clic en el campo de selección situado junto al parámetro y seleccione el valor deseado.

9.5.9.4 Borrar avisos de programa

Procedimiento

Para borrar un aviso de programa, proceda del siguiente modo:

1. Abra el bloque que contenga el aviso que desee borrar.
2. Seleccione la fila correspondiente en la interfaz de bloque y elija el comando "Borrar" del menú contextual.

Resultado

El aviso se borra.

9.5.9.5 Crear y editar la plantilla

Procedimiento

Para editar una plantilla, proceda del siguiente modo:

1. Seleccione el bloque de aviso deseado.
2. Introduzca en las columnas correspondientes los textos deseados o seleccione los atributos deseados.
3. Si no desea que los textos o atributos se puedan cambiar en la instancia, inhíbalos en la plantilla.

9.5.9.6 Crear y editar DBs de instancia

Requisitos

Previamente se debe haber creado un FB y haber creado al menos un aviso en él.

Procedimiento

Para asignar bloques de datos de instancia (DBs) a una plantilla y editar los avisos de dichos DBs para cada instancia, proceda del siguiente modo:

1. En el árbol del proyecto, haga doble clic en "Agregar nuevo bloque"; en el cuadro de diálogo que se abrirá, haga clic en el botón "Bloque de datos (DB)" y seleccione en la lista desplegable de "Plantilla" el bloque de función (plantilla) a la que desee asignar el bloque de datos de instancia.
2. Ahora, en la ventana de inspección haga clic en la ficha "Aviso" y seleccione el grupo deseado.
   O bien:
   En el árbol del proyecto, haga clic en "Avisos de PLC" para abrir la configuración de avisos.
3. Lleve a cabo los cambios deseados del DB de instancia correspondiente.

Nota

Si las propiedades de los DBs de instancia están protegidas contra escritura, primero deberá desbloquear la plantilla (FB).

Resultado

Así finaliza la configuración de avisos del DB de instancia seleccionado.

9.5.9.7 Insertar valores asociados en los avisos

Para agregar información actualizada (p. ej. del proceso) a los avisos, es posible insertar valores asociados en diferentes puntos de un texto de aviso.

Procedimiento

Para insertar un valor asociado en un aviso, proceda del siguiente modo:

1. Abra el menú contextual en el punto deseado y seleccione el comando de menú correspondiente para insertar una palabra clave, una variable o una entrada de una lista de textos.
2. En el cuadro de diálogo que aparece, introduzca los parámetros deseados y confirme las entradas.

Nota

Los valores asociados pueden configurarse tanto en el editor de avisos como en el editor de bloques. Si trabaja en el editor de avisos, debería haber abierto el bloque para el que se configura el aviso en el editor de bloques. En caso contrario, en el campo de selección "Variable" solo se pueden interconectar o mostrar variables ya configuradas, pero no agregar o crear variables nuevas.

Resultado

El valor asociado se integra en el aviso. En caso de producirse un error de sintaxis, el texto se marca con fondo rojo y debe corregirse.

Al introducir variables y entradas en listas de textos, los campos "Dirección" y "Parámetros de aviso" se rellenan automáticamente. De todas formas, no se ven hasta que vuelve a abrirse el cuadro de diálogo. El valor asociado se agrega automáticamente en el canal de entrada SD_1 a SD_10.

Consulte también

Ejemplos de valores asociados (Página 3191)

Ajustes al introducir valores asociados (Página 3190)

9.5.9.8 Ajustes al introducir valores asociados

Los valores asociados pueden insertarse en avisos tanto en el editor de avisos como en el editor de bloques. Si trabaja en el editor de avisos, debería haber abierto el bloque para el que se configura el aviso en el editor de bloques. De este modo, en el campo de selección "Variable" se muestran los parámetros que pueden aplicarse posteriormente. Si el editor de bloques está cerrado solo se mostrarán los parámetros ya utilizados.

Introducir una palabra clave como valor asociado

En la lista desplegable se presentan las palabras clave que pueden elegirse como valor asociado.

Introducir una entrada en la lista de textos como valor asociado

Seleccione la lista de textos deseada de la lista desplegable. Seguidamente, elija la entrada deseada en la lista de textos en el campo de selección "Variable".

Introducir una variable como valor asociado

Seleccione la variable deseada en el campo de selección "Variable". El programa completa automáticamente la dirección y los parámetros de aviso. En el apartado "Formato" debe definir el tipo de visualización adecuado para la variable especificada, la longitud mínima y, dado el caso, el número de decimales.

Nota

Los proyectos creados con una versión más antigua del TIA Portal pueden contener aún la notación "antigua" (manual) en valores asociados incrustados. Esta se corresponde con la sintaxis @<número del valor asociado>%<formato>@. Encontrará información sobre el parámetro "Formato" en Estructura de valores asociados (Página 3183).

En proyectos nuevos, utilice solo los diálogos descritos anteriormente para introducir valores asociados.

Consulte también

Palabras clave para textos de aviso (Página 3192)

Insertar valores asociados en los avisos (Página 3189)

9.5.9.9 Ejemplos de valores asociados

Ejemplos de valores asociados:

Palabra clave:

<Palabra clave: CpuName>

El nombre de la CPU se introduce como valor asociado.

Variable:

<Variable: #Temperature>

La temperatura se introduce como valor asociado.

Entrada en la lista de textos:

<Lista de textos: USER_1: "tag_1">

La entrada indicada en la lista de textos se introduce como valor asociado.

Consulte también

Insertar valores asociados en los avisos (Página 3189)

9.5.9.10 Borrar valores asociados

Procedimiento:

Para borrar valores asociados, proceda de la manera siguiente:

1. Abra la ficha "Configuración" del editor de programación.
2. Seleccione el valor asociado deseado y bórrelo.

O bien:

1. Seleccione el canal de entrada deseado en el editor de programación.
2. Borre el valor asociado.

9.5.10 Textos y atributos

9.5.10.1 Palabras clave para textos de aviso

Relación de palabras clave para textos de aviso

Puede seleccionar las siguientes palabras clave en textos de aviso, textos informativos o textos adicionales, para que el sistema de ingeniería las sustituya por sus valores reales.

Nombre del PLC

Nombre del PLC configurado.

Ejemplo: <Palabra clave: CpuName> = "PLC_1"

Nombre del DB de instancia

Instancia del aviso de instancia, normalmente un bloque de datos.

Ejemplo: <Palabra clave: Instance> = "Block_3_DB"

Anidamiento de FB para el aviso

Si se utilizan bloques anidados en el proyecto, se muestra la ruta hacia el bloque de función que ha creado el bloque de datos.

Ejemplo: <Palabra clave: Path> = "myFB2\myFB1"

Nota

En el caso de los bloques no anidados, la palabra clave "Path" se sustituye por una cadena vacía.

Nombre del aviso

Nombre del aviso.

Ejemplo: <Palabra clave: Name> = "myAlarm"

Nota

Las palabras clave no se pueden introducir en listas de textos.

9.5.10.2 Introducir textos

Los textos de los avisos se pueden introducir manualmente o se pueden utilizar valores predeterminados.

Plantilla de texto de la plantilla de aviso

Todos los textos de la plantilla de aviso están disponibles como plantilla al crear los textos de aviso. Si la plantilla ya contiene un texto general, todas las instancias de dicho tipo contendrán los mismos atributos y textos. En caso necesario, sólo habrá que modificarla.

Texto informativo

El texto informativo es un texto que puede crearse adicionalmente para determinados visualizadores. En determinados grupos de dispositivos (p. ej. WinCC) puede modificarse en runtime.

Textos adicionales

Los textos adicionales son textos que pueden visualizar determinados dispositivos HMI. Para ello, haga clic en la fila correspondiente e introduzca el texto. Si desea proteger el texto contra escritura, haga clic en la opción correspondiente de la columna. Los textos pueden contener saltos de línea.

Consulte también

Inhibir textos (Página 3193)

9.5.10.3 Inhibir textos

Opción "Inhibido" en la plantilla

Los textos sólo se pueden inhibir al editar las plantillas. Los textos inhibidos están protegidos contra escritura en los avisos derivados de la plantilla. El símbolo junto al campo de entrada indica que están inhibidos.

Inhibición de textos

Para inhibir textos, proceda del siguiente modo:

1. Comience por editar las plantillas.
2. Haga clic en el símbolo a la izquierda del campo de entrada que desee inhibir.
   Resultado: El símbolo se convierte en un eslabón cerrado.

Desinhibición de textos

Para desinhibir textos, proceda del siguiente modo:

1. Comience por editar la plantilla.
2. Haga clic en el símbolo a la izquierda del campo de entrada que desee desinhibir.
   Resultado: El símbolo se convierte en un eslabón abierto.

9.5.10.4 Inhibir atributos

Inhibición de atributos en la plantilla

Los atributos sólo se pueden inhibir al editar las plantillas. Los atributos inhibidos están protegidos contra escritura en los avisos derivados de la plantilla. El símbolo situado delante del campo de entrada indica que están bloqueados.

Inhibición de atributos

Para inhibir atributos, proceda del siguiente modo:

1. Comience por editar las plantillas.
2. En la tabla, haga clic en el símbolo a la izquierda del campo de entrada que desee inhibir.
   Resultado: El símbolo se convierte en un eslabón cerrado.

Desinhibición de atributos

Para desinhibir atributos, proceda del siguiente modo:

1. Comience por editar la plantilla.
2. En la tabla, haga clic en el símbolo a la izquierda del campo de entrada que desee desinhibir.
   Resultado: El símbolo se convierte en un eslabón abierto.

9.5.11 Listas de textos para avisos

9.5.11.1 Principios básicos de las listas de textos para avisos

Es posible adaptar las listas de textos existentes (listas de textos definidas por el usuario y por el sistema) a las distintas necesidades, así como editar textos y atributos. A continuación, es posible traducir los textos al idioma o idiomas de proyecto deseados.

Si un bloque que emite avisos y hace referencia a una lista de textos es copiado a otro programa, también habrá que copiar las correspondientes listas de textos, o bien crear otra lista de textos con el mismo nombre y contenido o modificar la referencia en el texto de aviso

Nota

Los valores asociados que figuran en entradas de listas de textos no se admiten o bien no se resuelven y se muestran como texto explícito.

El capítulo "Trabajar con listas de textos" incluye información detallada sobre las listas de textos.

Consulte también

Listas de textos (Página 420)

Crear listas de textos de usuario (Página 422)

Editar listas de textos de usuario (Página 422)

Editar listas de textos de sistema (Página 423)

9.5.11.2 Editar listas de textos para avisos

Requisitos

- Se ha creado un aviso de diagnóstico definido por el usuario.

Procedimiento

Para editar las listas de textos definidas por el usuario, proceda del siguiente modo:

1. Haga doble clic en el comando "Listas de textos" bajo el dispositivo en el árbol del proyecto. Se abrirá el editor de listas de textos.
2. Seleccione en la tabla la lista de textos que desee editar.
3. Modifique los valores deseados.

Es posible modificar los siguientes valores:

- Títulos:
  Los títulos de los textos de avisos (texto del aviso, textos adicionales) se pueden configurar libremente salvo en el texto informativo.
- Textos estáticos:
  Aquí se pueden modificar textos creados anteriormente o insertar nuevos textos en una lista de textos existente. Tenga en cuenta que no está permitido introducir valores asociados en listas de textos.

Consulte también

Editar listas de textos de sistema (Página 423)

9.5.11.3 Integrar en avisos los textos de listas de textos

En un aviso se pueden integrar textos de distintas listas de textos. Los textos pueden ubicarse libremente, por lo que también se garantiza el uso en avisos en otros idiomas.

Procedimiento en el S7-300/400

Para integrar en avisos los textos de las listas de textos, proceda del siguiente modo:

1. Haga doble clic en "Avisos de PLC" en el árbol del proyecto. Se abre el editor de avisos.
2. En el lugar del aviso en el que quiere que aparezca el texto de la lista de textos, introduzca un comodín con el formato @[índice][tipo de datos]%t#[lista de textos]@.

Nota

S7-300/400: [Índice] = p. ej. 1, siendo 1 el primer valor asociado del aviso.

Procedimiento en el S7-1500

Para integrar en avisos los textos de las listas de textos, proceda del siguiente modo:

1. Haga doble clic en "Avisos de PLC" en el árbol del proyecto. Se abre el editor de avisos.
2. Vaya al lugar del aviso en el que deba aparecer el texto de la lista de textos y, en el menú contextual, seleccione la opción para insertar una lista de textos.
3. Seleccione la lista de textos deseada de la lista desplegable. Seguidamente, elija la entrada deseada de la lista de textos en el campo de selección "Variable".

9.5.11.4 Ejemplo de integración en avisos de los textos de las listas de textos

S7-300/400: Texto de aviso configurado: La presión ha aumentado @2W%t#Textliste1@.

S7-1500: texto de aviso configurado: La presión ha aumentado <lista de textos: USER_1: "tag_1">.

Lista de texto con el nombre "ListaTexto1":

Índice	alemán	inglés
1734	demasiado	too high

El valor asociado recibe el valor 1734.

Se muestra el siguiente texto de aviso (si está ajustado "español" como idioma de proyecto en el panel de operador): La presión ha aumentado demasiado.

Consulte también

Integrar en avisos los textos de listas de textos (Página 3196)

9.5.12 Categorías

9.5.12.1 Crear categorías

Las categorías se pueden configurar libremente. Es posible crearlas y editarlas en el editor de categorías. Seguidamente se puede asignar una categoría a un aviso en el editor de avisos.

El sistema contempla dos categorías, "Ackownledgement" (para avisos con acuse) y "No Ackownledgement" (para avisos sin acuse). Estas no pueden modificarse.

En total pueden crearse hasta 16 categorías. La propiedad "Con acuse" de las categorías indica si los avisos correspondientes son de acuse obligado.

Requisitos

En la navegación del proyecto se ha abierto la carpeta "Datos comunes".

Procedimiento

Para crear una categoría, proceda del siguiente modo:

1. Haga doble clic en la entrada "Categorías" de la navegación del proyecto.
   Se abrirá el editor de categorías.
2. Elija el comando "Insertar nueva categoría" del menú contextual.
3. Asigne un nombre unívoco a la nueva categoría en la columna "Nombre".
   El nombre es neutro para todos los idiomas.
4. Indique en la columna "Nombre de visualización" el nombre que se visualizará. Este nombre es traducible.
5. En la columna "Con acuse", indique si los avisos de esta categoría deben acusarse o no.

9.5.12.2 Editar categorías

En cualquier momento pueden modificarse los ajustes (nombre, nombre de visualización o acuse) de una categoría, incluso si dicha categoría ya tiene asignados avisos. Los cambios se aplicarán automáticamente en los avisos.

Copiar categorías

Para copiar categorías, proceda de la siguiente manera:

1. Marque la fila con la categoría que quiere copiar.
2. Elija el comando "Copiar" del menú contextual.
3. Elija el comando "Pegar" del menú contextual.

Resultado

La categoría copiada se almacenará con un nuevo nombre en la última posición de la tabla.

El nombre de la nueva categoría se compone de lo siguiente:

<Nombre antiguo><n.º>

N.º: número natural más bajo disponible

9.6 Uso del Team Engineering

9.6.1 Intercambio de datos con Inter Project Engineering (IPE)

9.6.1.1 Principios básicos de la ingeniería interproyecto (IPE)

Introducción a la ingeniería interproyecto (IPE)

Con la funcionalidad de ingeniería interproyecto, en lo sucesivo IPE, se pueden intercambiar los datos de control de un proyecto de origen con otros proyectos mediante un proxy de dispositivo.

Estos datos se pueden incorporar a otros proyectos, p. ej. para la visualización en HMI, y ahí ser utilizados para otras configuraciones.

Así, la ingeniería interproyecto le permite, de un modo simple, disponer de datos de control interproyecto de una programación PLC para la configuración HMI.

Con el objeto "Datos del proxy del dispositivo" se efectúa un intercambio coherente y cómodo de los datos de control entre varios proyectos sin redundancia de labores de configuración.

De esta manera, el ingeniero de configuración HMI puede trabajar en un proyecto sin necesidad de que exista configuración hardware en su proyecto. Así, los trabajos los trabajos en el proyecto PLC y en el proyecto HMI pueden realizarse paralela y simultáneamente.

Con el proxy de dispositivo se pueden intercambiar los siguientes datos de control entre los proyectos:

- Bloques de programa
- Objetos tecnológicos
- Variables PLC
- Avisos PLC

En el intercambio se transfieren de forma automática los datos siguientes:

- Interfaces del controlador
- Procesadores y módulos de comunicación configurados

Con este intercambio automático de datos se asegura que, además de los datos seleccionados por usted, también se transfieran las interfaces y los procesadores y módulos de comunicación configurados. Estos datos son necesarios para un procesamiento ulterior correcto y coherente en su proyecto de destino.

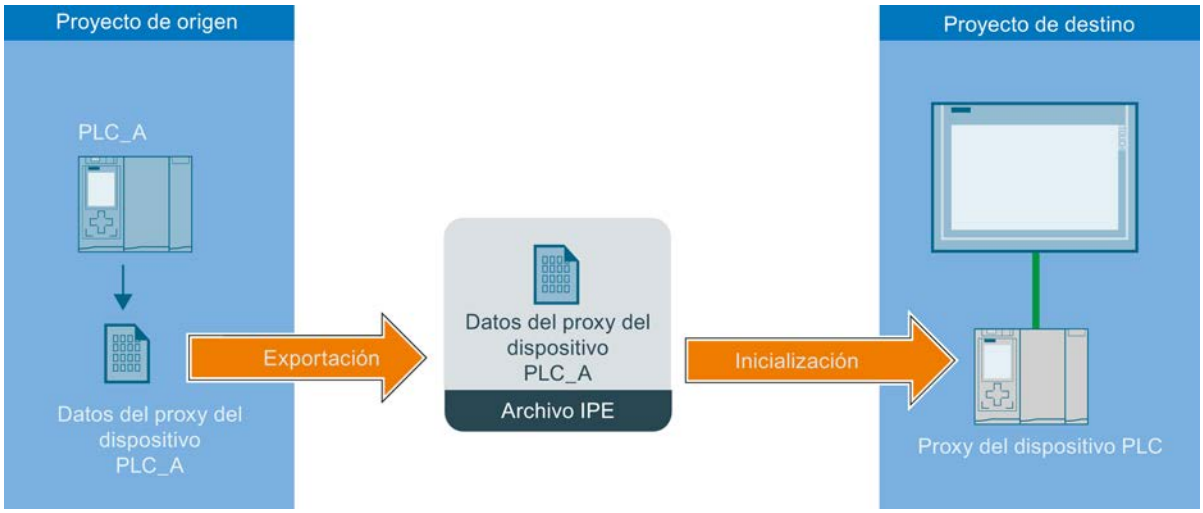
Intercambio de datos entre varios proyectos

Para el intercambio de datos entre varios proyectos con ingeniería interproyecto tiene las opciones siguientes:

- Intercambio de datos de control mediante un archivo IPE
- Intercambio de datos de control mediante un archivo de proyecto

Intercambio de datos de control mediante un archivo IPE

En la figura siguiente se muestra el intercambio de datos entre varios proyectos por medio de un archivo IPE.



Los datos de control del PLC procedentes del proyecto de origen se transfieren al PLC del proyecto de destino mediante el objeto "Datos del proxy del dispositivo".

Los datos de control que se intercambian con un archivo IPE tienen que proceder de proyectos de TIA Portal V13 o superior.

Intercambio de los datos de control mediante un archivo de proyecto

La figura siguiente muestra el intercambio de datos entre varios proyectos por medio de un archivo de proyecto.

Los datos de control de PLC procedentes del proyecto de origen se transfieren al PLC del proyecto de destino mediante un archivo de proyecto.

Es posible adoptar y reutilizar cómodamente datos de control tanto de antiguos proyectos del TIA Portal como de proyectos STEP 7 Classic que no están integrados en el TIA Portal en los proyectos TIA Portal actuales.

9.6.1.2 Requisitos para la ingeniería interproyecto (IPE)

Requisitos de software y hardware

Para aprovechar la funcionalidad de la ingeniería interproyecto deben cumplirse los requisitos mínimos de software y hardware para la instalación del TIA Portal V13 o superior.

Adicionalmente, existen los requisitos siguientes:

- El TIA Portal V13 o superior y el paquete de software "SIMATIC STEP 7 Professional" están instalados.
- Utilice una CPU de la familia S7-1200/1500 o S7-300/400.
- Todos los controladores implicados deben tener instalada la misma versión de software.

Requisitos para IPE

Para poder importar datos de control procedentes de proyectos, existen los requisitos siguientes:

- Haber creado un proyecto con configuración hardware y datos PLC que puedan transferirse a otro proyecto.
- Haber comprobado si el proyecto es coherente antes de transferir los datos de control mediante los datos del proxy del dispositivo.
- Conocer los procedimientos definidos para la ingeniería de interproyecto.

Indicaciones sobre el modo de compatibilidad

La funcionalidad para la ingeniería interproyecto no está disponible en el modo de compatibilidad.

9.6.1.3 Procedimientos al trabajar con ingeniería interproyecto (IPE)

Intercambio de datos de control entre varios proyectos

Con la funcionalidad de ingeniería interproyecto (IPE) se pueden intercambiar los datos de control existentes con un proxy de dispositivo y utilizarlos a continuación en otro proyecto. El proxy de dispositivo le permite un intercambio de datos coherente y cómodo entre varios proyectos sin redundancia de labores de configuración.

Procedimiento para el intercambio de datos de control mediante un archivo IPE

Para el intercambio de datos mediante un archivo IPE siga estos pasos:

1. Con TIA Portal V13 o superior cree un proyecto con todos los datos de control necesarios y un programa de usuario ejecutable.
2. Configure el hardware necesario para el proyecto.
3. Compile el proyecto para asegurar la coherencia de su proyecto.
4. En su proyecto, debajo de la CPU deseada, cree el objeto "Datos del proxy del dispositivo" haciendo clic en el comando "Agregar datos del proxy del dispositivo" de la carpeta "Datos del proxy del dispositivo".
   Resultado: se ha creado el objeto "Datos del proxy del dispositivo".
5. Seleccione el objeto Datos del proxy del dispositivo que desee en el árbol del proyecto, y ábralo en el editor haciendo doble clic.
6. Introduzca en el editor los datos deseados para Datos del proxy del dispositivo bajo "General" o transfiera el ajuste predeterminado. Tiene la opción de modificar el nombre y de añadir un comentario para la exportación de los datos de control.
   Seleccione en el área "Definir contenido" del editor los datos de control que desee exportar como archivo IPE y haga clic en el comando "Exportar datos del proxy del dispositivo".

7. Indique en el cuadro de diálogo siguiente el nombre y la ubicación para el archivo IPE que se ha de crear y haga clic en "Guardar". En cuanto la exportación finalice correctamente, se le mostrará un aviso.
   Resultado: los datos de control seleccionados se han exportado como archivo IPE.
8. En el proyecto de destino, cree un proxy de dispositivo para poder importar los datos de control que hay en el archivo IPE. Para ello, haga clic en el comando "Agregar dispositivo" y seleccione el proxy de dispositivo que desee en el cuadro de diálogo siguiente.
9. Haga clic en el proxy recién creado y, en el menú contextual, seleccione el comando "Inicializar proxy del dispositivo".
10. Seleccione el archivo IPE que se creó antes y que va a utilizarse para la inicialización, y confirme la selección en el cuadro de diálogo siguiente con "Aceptar". A continuación se lleva a cabo la inicialización.
    Resultado: los datos de control del proyecto de origen contenidos en el archivo IPE se transfieren al proxy de dispositivo del proyecto de destino.

En cuanto se haya realizado correctamente el intercambio de datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

Procedimiento para el intercambio de datos de control mediante un archivo de proyecto

Al intercambiar datos de control mediante un archivo de proyecto, pueden haberse predefinido datos del proxy del dispositivo en el proyecto de origen o definirse y seleccionarse desde el proyecto de destino.

Para el intercambio de datos mediante un archivo de proyecto siga estos pasos:

1. Con TIA Portal V13 o superior cree un proyecto con todos los datos de control necesarios y un programa de usuario ejecutable.
2. Configure el hardware necesario para el proyecto.
3. Compile el proyecto para asegurar la coherencia de su proyecto.
4. Abra el proyecto de TIA Portal cuyos datos de control desee importar al proyecto de destino.
5. En su proyecto, debajo de la CPU deseada, cree el objeto "Datos del proxy del dispositivo" haciendo clic en el comando "Agregar datos del proxy del dispositivo" de la carpeta "Datos del proxy del dispositivo".
   Resultado: se ha creado el objeto "Datos del proxy del dispositivo".
6. Introduzca en el editor los datos deseados para Datos del proxy del dispositivo bajo "General" o transfiera el ajuste predeterminado. Tiene la opción de modificar el nombre y de añadir un comentario para los datos del proxy del dispositivo.
7. Seleccione en el editor, en el área "Definir contenido", qué datos de control desea poner a disposición a través del objeto Datos del proxy del dispositivo.
8. Haga clic en "Guardar" para guardar las modificaciones del proyecto.
   Resultado: ha guardado correctamente los datos de control seleccionados en sus datos del proxy del dispositivo.

9. En el proyecto de destino, cree un proxy de dispositivo para poder importar los datos de control que hay en el archivo de proyecto. Para ello, haga clic en el comando "Agregar dispositivo" y seleccione el proxy de dispositivo que desee en el cuadro de diálogo siguiente.
10. Haga clic en el proxy recién creado y, en el menú contextual, seleccione el comando "Inicializar proxy del dispositivo".
11. Seleccione el archivo de proyecto creado anteriormente con el objeto Datos del proxy del dispositivo preparado y que va a utilizarse para la inicialización, y confirme la selección en el cuadro de diálogo siguiente con "Aceptar". A continuación se lleva a cabo la inicialización. **Resultado:** los datos de control del proyecto de origen contenidos en el archivo de proyecto se transfieren al proxy del proyecto de destino.

En cuanto se haya realizado correctamente el intercambio de datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

Procedimiento para actualizar los datos de control ya transferidos

Requisitos para la actualización:

Para actualizar los datos de control ya transferidos mediante el objeto "Datos del proxy del dispositivo" ya creado deben cumplirse los siguientes requisitos:

- Para la actualización se utiliza el mismo proyecto que en la anterior transferencia de los datos de control.
- Para la transferencia de datos de control se utilizan el objeto "Datos del proxy del dispositivo" ya creado.
- No ha modificado la configuración hardware ni las interfaces de comunicación. Está permitido agregar nuevos componentes de hardware.

Procedimiento para la actualización:

1. Abra el proyecto del TIA Portal cuyos datos de control desee actualizar en el proyecto de destino.
2. Haga doble clic en el objeto "Datos del proxy del dispositivo" existente y seleccione en "Contenido" los datos de control que desee para la actualización.
3. Guarde el proyecto y exporte de nuevo el archivo IPE.
4. En el TIA Portal, abra el proyecto de destino que desee.
5. Haga clic en el proxy de dispositivos existente en el proyecto de destino y, en el menú contextual, seleccione el comando "Actualizar proxy del dispositivo".
6. Seleccione el archivo de origen que deba utilizarse para la actualización y confirme la selección.
7. En el cuadro de diálogo siguiente, seleccione en "Dispositivo" la CPU utilizada anteriormente como CPU de origen, desde la que se deben volver a importar los datos de control.

8. En "Datos definidos del proxy del dispositivo", seleccione el objeto que desee y cierre la ventana con "Aceptar".
9. A continuación comienza la actualización y los datos seleccionados del proxy del dispositivo se transfieren desde el proyecto de origen al proxy de dispositivo del proyecto de destino.

Una vez finalizada correctamente la actualización de los datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

9.6.1.4 Creación de los datos del proxy del dispositivo

Creación de los datos del proxy del dispositivo

Para crear un objeto Datos del proxy del dispositivo siga los pasos siguientes:

1. Abra la carpeta "Datos del proxy del dispositivo" debajo de la CPU para la cual desea crear un objeto Datos del proxy del dispositivo.
2. Haga clic en el comando "Agregar nuevos datos del proxy del dispositivo".

Resultado

El nuevo objeto Datos del proxy del dispositivo se crea y se muestra en el árbol del proyecto.

Consulte también

Principios básicos de la ingeniería interproyecto (IPE) (Página 3198)

Requisitos para la ingeniería interproyecto (IPE) (Página 3200)

Procedimientos al trabajar con ingeniería interproyecto (IPE) (Página 3201)

Abrir datos del proxy del dispositivo (Página 3205)

Intercambio de datos de control mediante un archivo IPE (Página 3205)

9.6.1.5 Abrir datos del proxy del dispositivo

Abrir datos del proxy del dispositivo

Para abrir un objeto Datos del proxy del dispositivo, proceda del siguiente modo:

1. En la carpeta "Datos del proxy del dispositivo" del árbol del proyecto de la respectiva CPU, abra el objeto "Datos del proxy del dispositivo" que desee haciendo doble clic.

Resultado

El objeto Datos del proxy del dispositivo se abre en el editor y se muestran los datos de control que le fueron asignados.

Adopte los ajustes predeterminados que se muestran en el editor o adáptelos a sus necesidades.

Consulte también

Intercambio de datos de control mediante un archivo IPE (Página 3205)

Creación de los datos del proxy del dispositivo (Página 3204)

Principios básicos de la ingeniería interproyecto (IPE) (Página 3198)

Requisitos para la ingeniería interproyecto (IPE) (Página 3200)

Procedimientos al trabajar con ingeniería interproyecto (IPE) (Página 3201)

9.6.1.6 Intercambio de datos de control mediante un archivo IPE

Intercambio de datos de control mediante "Archivo IPE"

Para exportar los datos de control visualizados en Datos del proxy del dispositivo de la CPU de nivel superior, proceda del siguiente modo:

1. Abra el proyecto de TIA Portal cuyos datos de control desee importar al proyecto de destino.
2. Compile el proyecto para asegurar la coherencia de su proyecto.
3. En su proyecto, debajo de la CPU deseada, cree el objeto "Datos del proxy del dispositivo" haciendo clic en el comando "Agregar datos del proxy del dispositivo" de la carpeta "Datos del proxy del dispositivo".
   Resultado: se ha creado el objeto "Datos del proxy del dispositivo".
4. Seleccione el objeto Datos del proxy del dispositivo que desee en el árbol del proyecto y ábralo en el editor haciendo doble clic.
5. Introduzca en el editor los datos deseados para Datos del proxy del dispositivo bajo "General" o transfiera el ajuste predeterminado. Tiene la opción de modificar el nombre y de añadir un comentario.
6. Seleccione en el editor, en el área "Definir contenido", qué datos de control desea poner a disposición a través del objeto Datos del proxy del dispositivo.

7. Haga clic en el comando "Exportar Datos del proxy del dispositivo" para iniciar la exportación.
8. Indique en el cuadro de diálogo siguiente el nombre y la ubicación para los datos que van a exportarse y haga clic en "Guardar". En la exportación, el archivo Datos del proxy del dispositivo recibe automáticamente el formato de archivo ".ipe".

Resultado

Se ejecuta la exportación de los datos seleccionados del proxy del dispositivo y se muestra una barra de proceso para la exportación.

Al finalizar la exportación, se emite un aviso de que ha finalizado correctamente.

Los datos de control de la CPU seleccionada se almacenaron en el archivo ".ipe" guardado y pueden transferirse cómodamente por correo electrónico, p. ej. Los datos de control pueden importarse y seguirse utilizando en otros dispositivos sin más esfuerzo de configuración, p. ej., para la visualización.

Consulte también

Principios básicos de la ingeniería interproyecto (IPE) (Página 3198)

Requisitos para la ingeniería interproyecto (IPE) (Página 3200)

Procedimientos al trabajar con ingeniería interproyecto (IPE) (Página 3201)

Creación de los datos del proxy del dispositivo (Página 3204)

Abrir datos del proxy del dispositivo (Página 3205)

9.6.1.7 Intercambio de datos de control mediante un archivo de proyecto

Intercambio de datos de control mediante un archivo de proyecto

Al intercambiar datos de control mediante un archivo de proyecto, los datos del proxy del dispositivo deben estar predefinidos en el proyecto de origen. Para exportar los datos de control visualizados en los datos del proxy del dispositivo de la CPU de nivel superior como archivo de proyecto, proceda del siguiente modo:

1. Abra el proyecto de TIA Portal cuyos datos de control desee importar al proyecto de destino.
2. Compile el proyecto para asegurar la coherencia de su proyecto.
3. En su proyecto, debajo de la CPU deseada, cree el objeto "Datos del proxy del dispositivo" haciendo clic en el comando "Agregar datos del proxy del dispositivo" de la carpeta "Datos del proxy del dispositivo".
   Resultado: se ha creado el objeto "Datos del proxy del dispositivo".
4. Seleccione el objeto Datos del proxy del dispositivo que desee en el árbol del proyecto y ábralo en el editor haciendo doble clic.

5. Introduzca en el editor los datos deseados para Datos del proxy del dispositivo bajo "General" o transfiera el ajuste predeterminado. Tiene la opción de modificar el nombre y de añadir un comentario.
6. Seleccione en el editor, en el área "Definir contenido", qué datos de control desea poner a disposición a través del objeto Datos del proxy del dispositivo.
7. Haga clic en "Guardar" para guardar las modificaciones del proyecto.
   Resultado: ha guardado correctamente los datos de control seleccionados en sus datos del proxy del dispositivo.
8. Abra el proyecto de destino al cual deban transferirse los datos de control. En el proyecto de destino, cree un proxy de dispositivo para poder importar los datos de control que hay en el archivo de proyecto. Para ello, haga clic en el comando "Agregar dispositivo" y seleccione el proxy de dispositivo que desee en el cuadro de diálogo siguiente.
9. Haga clic en "Guardar" para guardar las modificaciones del proyecto.
10. Haga clic en el proxy recién creado y, en el menú contextual, seleccione el comando "Inicializar proxy del dispositivo".
11. Seleccione el archivo de proyecto creado anteriormente con el objeto Datos del proxy del dispositivo que ha preparado y que va a utilizarse para la inicialización, y confirme la selección en el cuadro de diálogo siguiente con "Aceptar".
12. En el cuadro de diálogo siguiente seleccione la CPU que desee y el objeto Datos del proxy del dispositivo preparado. Confirme el cuadro de diálogo haciendo clic en "Aceptar". A continuación se lleva a cabo la inicialización.
    Resultado: los datos de control del proyecto de origen contenidos en el archivo de proyecto se transfieren al proxy del dispositivo del proyecto de destino.

En cuanto se haya realizado correctamente el intercambio de datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

9.7 Ejemplos de programación

9.7.1 Ejemplos de programación (S7-1200, S7-1500)

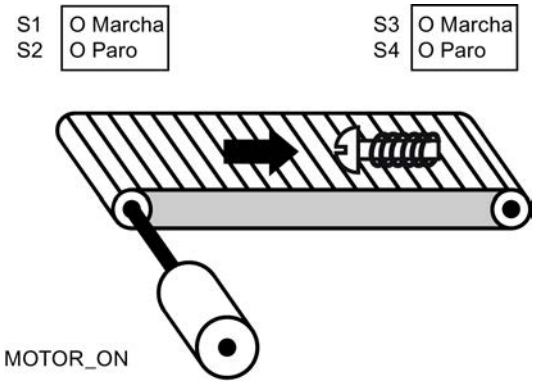
9.7.1.1 Ejemplos de programación KOP

Ejemplo de control de una cinta transportadora

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al principio de la cinta transportadora se encuentran dos pulsadores: S1 para MARCHA y S2

para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

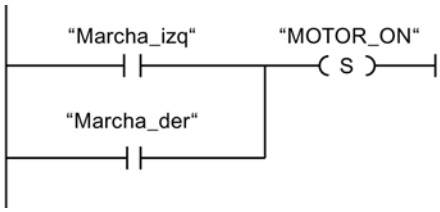
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
Marcha_izquierda (S1)	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Paro_izquierda (S2)	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora
MOTOR_ON	BOOL	Conectar el motor de la cinta transportadora

Los segmentos siguientes muestran la programación KOP para solucionar la tarea:

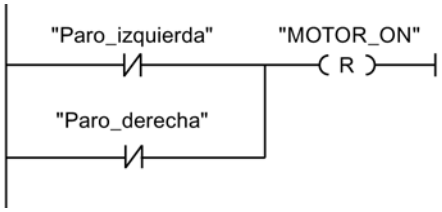
Segmento 1:

El motor de la cinta transportadora se conecta cuando se acciona el pulsador de marcha "S1" o "S3".



Segmento 2:

El motor de la cinta transportadora se desconecta cuando se acciona el pulsador de paro "S2" o "S4".



Consulte también

---(S)---: Activar salida (Página 3289)

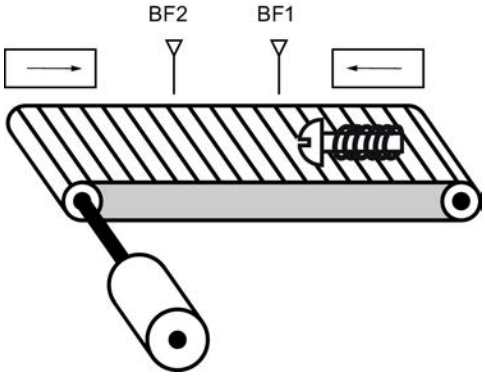
---| |---: Contacto normalmente abierto (Página 3283)

---(R)---: Desactivar salida (Página 3288)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

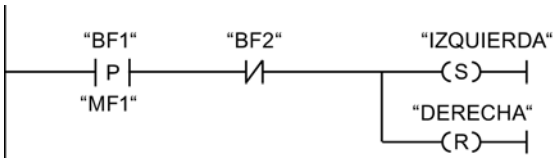
Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
DERECHA	BOOL	Indicador de desplazamiento a la derecha

Nombre	Tipo de datos	Descripción
IZQUIERDA	BOOL	Indicador de desplazamiento a la izquierda
MF1	BOOL	Marca de flancos 1
MF2	BOOL	Marca de flancos 2

Los segmentos siguientes muestran la programación KOP para solucionar la tarea:

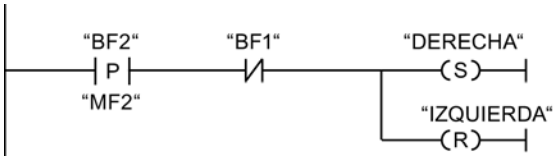
Segmento 1:

Si el estado lógico de la barrera fotoeléctrica "BF1" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF2" es "0", el objeto se desplaza hacia la izquierda sobre la cinta transportadora.



Segmento 2:

Si el estado lógico de la barrera fotoeléctrica "BF2" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora.



Consulte también

---(S)---: Activar salida (Página 3289)

---| / |---: Contacto normalmente cerrado (Página 3285)

---(R)---: Desactivar salida (Página 3288)

--|P|--: Consultar flanco de señal ascendente de un operando (Página 3295)

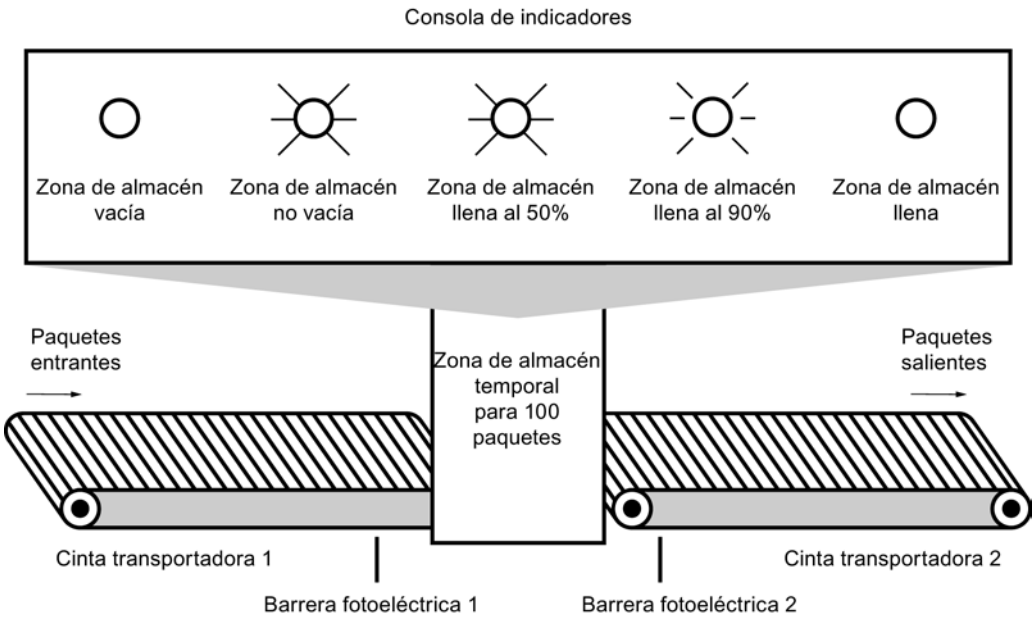
Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde esta zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben

transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
RESET	BOOL	Inicializar contador
LOAD	BOOL	Poner el contador al valor del parámetro "PV"
EXISTENCIAS	INT	Existencias en almacén al rearrancar
PAQUETES	INT	Número de paquetes en la zona de almacén (valor de contaje actual)
PAQUETES_ZA	BOOL	Se activa si el valor de contaje actual es mayor o igual al valor de la variable "EXISTENCIAS".
ALM_VACIO	BOOL	Indicador luminoso: zona de almacén vacía
ALM_NO_VACIO	BOOL	Indicador luminoso: zona de almacén no vacía
ALM_LLENO_50%	BOOL	Indicador luminoso: zona de almacén llena al 50 %

Nombre	Tipo de datos	Descripción
ALM_LLENO_90%	BOOL	Indicador luminoso: zona de almacén llena al 90 %
ALM_LLENO	BOOL	Indicador luminoso: zona de almacén llena
VOLUMEN_50	INT	Valor de comparación: 50 paquetes
VOLUMEN_90	INT	Valor de comparación: 90 paquetes
VOLUMEN_100	INT	Valor de comparación: 100 paquetes

Los segmentos siguientes muestran la programación KOP para activar los indicadores luminosos:

Segmento 1:

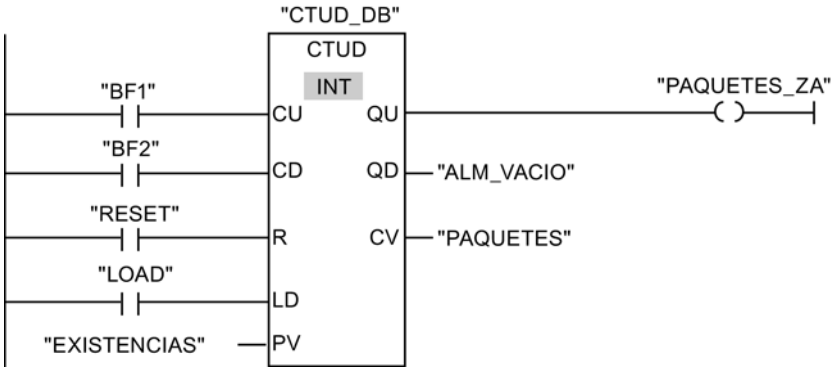
Cuando un paquete se transporta a la zona de almacén, el estado lógico de "BF1" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF1", el contador ascendente se activa y el valor de contaje actual de "PAQUETES" aumenta en uno.

Cuando un paquete se transporta de la zona de almacén a la plataforma de carga, el estado lógico de "BF2" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF2", el contador descendente se activa y el valor actual de "PAQUETES" se reduce en uno.

Si no hay paquetes en la zona de almacén ("PAQUETES" = "0"), la variable "ALM_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén vacía".

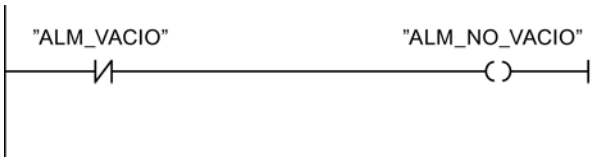
El valor de contaje actual puede ponerse a "0" si la variable "RESET" se pone al estado lógico "1".

Cuando la variable "LOAD" se pone al estado lógico "1", el valor de contaje actual adopta el valor de la variable "EXISTENCIAS". Mientras que el valor de contaje actual sea mayor o igual al valor de la variable "EXISTENCIAS", la variable "PAQUETES_ZA" devolverá el estado lógico "1".



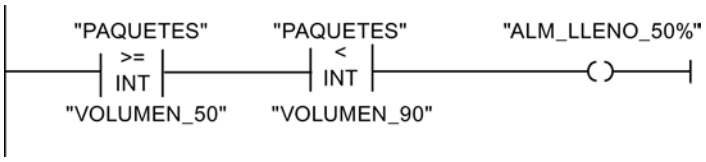
Segmento 2:

Mientras haya paquetes en la zona de almacén, la variable "ALM_NO_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén no vacía".



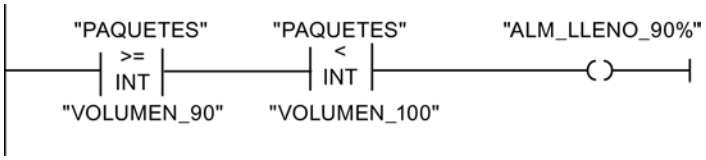
Segmento 3:

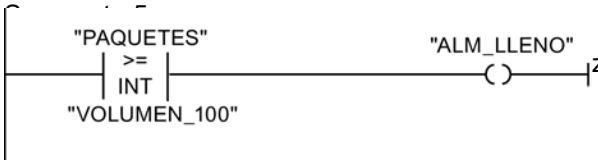
Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 50, se enciende el indicador luminoso del aviso "Zona de almacén llena al 50 %".



Segmento 4:

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 90, se enciende el indicador luminoso del aviso "Zona de almacén llena al 90 %".





zona de almacén alcanza 100, se enciende
llena".

Consulte también

---( )---: Asignación (Página 3286)

CTUD: Contador ascendente - descendente (Página 3357)

CMP >=: Mayor o igual (Página 3375)

CMP <: Menor (Página 3381)

---| |---: Contacto normalmente abierto (Página 3283)

---| / |---: Contacto normalmente cerrado (Página 3285)

Ejemplo de control de la temperatura ambiente

Controlar la temperatura ambiente

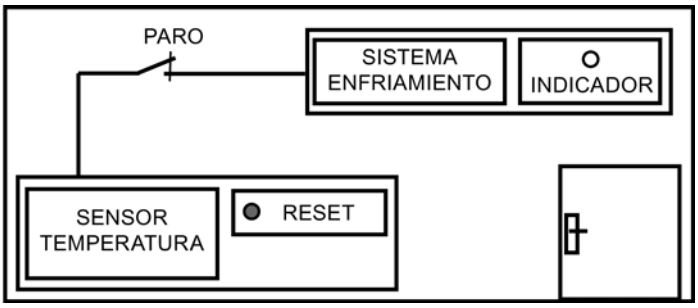
La temperatura de una cámara de enfriamiento debe mantenerse por debajo de los cero grados centígrados. Las posibles fluctuaciones de la temperatura se controlan mediante un sensor. Si la temperatura aumenta por encima de los cero grados centígrados, el sistema de

enfriamiento se conecta por un tiempo preseleccionado. Durante la fase de enfriamiento se enciende un indicador "Sistema de enfriamiento iniciado".

El sistema de enfriamiento y el indicador luminoso se apagan cuando se cumple una de las condiciones siguientes:

- El sensor indica un descenso de la temperatura por debajo de los cero grados centígrados.
- El tiempo de enfriamiento preseleccionado ha transcurrido.
- Se acciona el pulsador "PARO".

Si ha transcurrido el tiempo de enfriamiento preseleccionado y la temperatura de la cámara de enfriamiento es aún demasiado elevada, el sistema de enfriamiento se puede reiniciar mediante el pulsador "RESET".



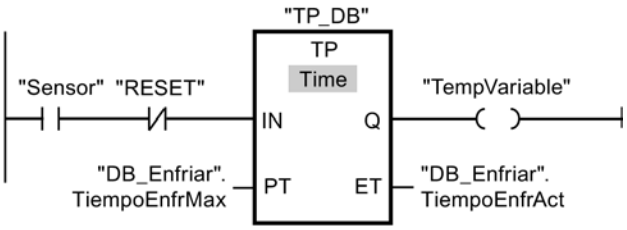
Realización

La tabla siguiente muestra la definición de las variables utilizadas:

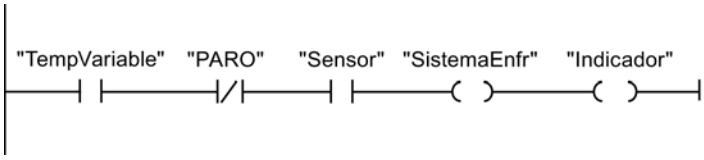
Nombre	Tipo de datos	Comentario
Sensor	BOOL	Señal del sensor de temperatura
RESET	BOOL	Reset
PARO	BOOL	El sistema de enfriamiento se desconecta.
TiempoEnfrMax	TIME	Tiempo de enfriamiento preseleccionado Esta variable está definida en el bloque de datos "DB_Enfriar".
TiempoEnfrAct	TIME	Tiempo de enfriamiento transcurrido actualmente Esta variable está definida en el bloque de datos "DB_Enfriar".
SistemaEnfr	BOOL	El sistema de enfriamiento se conecta.
Indicador	BOOL	El indicador luminoso del aviso "Sistema de enfriamiento iniciado" se enciende.
TempVariable	BOOL	Variable temporal Esta variable guarda el estado lógico del temporizador CEI TP.

El segmento siguiente muestra la programación KOP para controlar la temperatura ambiente:

Segmento 1:



Segmento 2:



Si la temperatura de la cámara de enfriamiento aumenta por encima de los cero grados centígrados, el estado lógico del operando "Sensor" cambia de "0" a "1" (flanco de señal ascendente). Con un flanco de señal ascendente en la entrada IN de la función de temporización se inicia el tiempo de enfriamiento preseleccionado y la "TempVariable" recibe el estado lógico "1". El estado lógico "1" de la "TempVariable" hace que en el segmento 2 se conecte el sistema de enfriamiento y se encienda el indicador luminoso. Las salidas "Sensor", "SistemaEnfr" e "Indicador" tienen que programarse en el segmento 2, ya que en la salida Q de la función de temporización sólo se puede programar una bobina.

Si la temperatura de la cámara de enfriamiento desciende por debajo de los cero grados centígrados, el estado lógico del sensor cambia nuevamente a "0". Con ello se apagan el sistema de enfriamiento y el indicador luminoso.

Si el sensor no indica ningún descenso de la temperatura, el sistema de enfriamiento y el indicador luminoso se apagan a más tardar al transcurrir el tiempo de enfriamiento preseleccionado. En este caso, el proceso de enfriamiento se puede reiniciar mediante el pulsador "RESET". Al pulsar y soltar el pulsador se genera otro flanco ascendente en la entrada IN, con lo que se reinicia el sistema de enfriamiento.

El sistema de enfriamiento y el indicador luminoso se pueden apagar en todo momento con el pulsador "PARO".

Consulte también

---( )---: Asignación (Página 3286)

TP: Impulso (Página 3306)

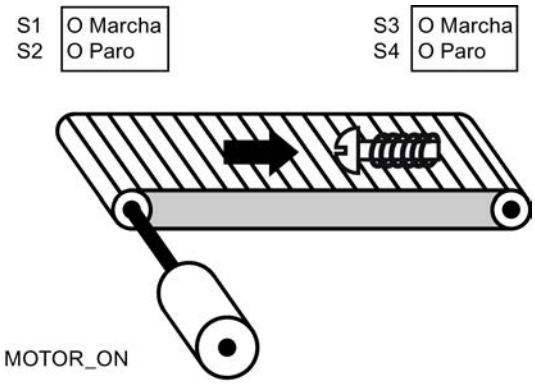
---| |---: Contacto normalmente abierto (Página 3283)

9.7.1.2 Ejemplos de programación FUP

Ejemplo de control de una cinta transportadora

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al principio de la cinta transportadora se encuentran dos pulsadores: S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

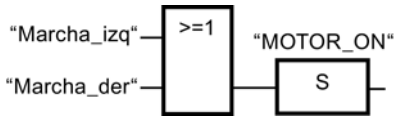
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
Marcha_izquierda (S1)	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Paro_izquierda (S2)	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora
MOTOR_ON	BOOL	Conectar el motor de la cinta transportadora

Los segmentos siguientes muestran la programación FUP para solucionar la tarea:

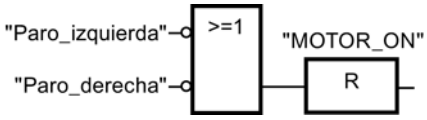
Segmento 1:

El motor de la cinta transportadora se conecta cuando se acciona el pulsador de marcha "S1" o "S3".



Segmento 2:

El motor de la cinta transportadora se desconecta cuando se acciona el pulsador de paro "S2" o "S4".



Consulte también

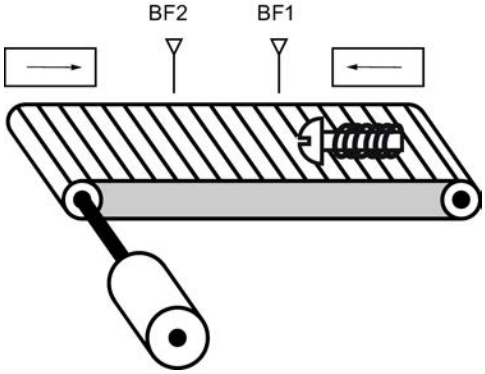
>=1: Operación lógica O (Página 3590)

R: Desactivar salida (Página 3597)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

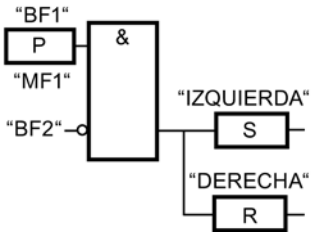
Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
DERECHA	BOOL	Indicador de desplazamiento a la derecha

Nombre	Tipo de datos	Descripción
IZQUIERDA	BOOL	Indicador de desplazamiento a la izquierda
MF1	BOOL	Marca de flancos 1
MF2	BOOL	Marca de flancos 2

Los segmentos siguientes muestran la programación FUP para solucionar la tarea:

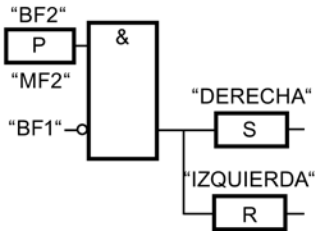
Segmento 1:

Si el estado lógico de la barrera fotoeléctrica "BF1" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF2" es "0", el objeto se desplaza hacia la izquierda sobre la cinta transportadora.



Segmento 2:

Si el estado lógico de la barrera fotoeléctrica "BF2" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora.



Consulte también

&: Operación lógica Y (Página 3588)

R: Desactivar salida (Página 3597)

P: Consultar flanco de señal ascendente de un operando (Página 3604)

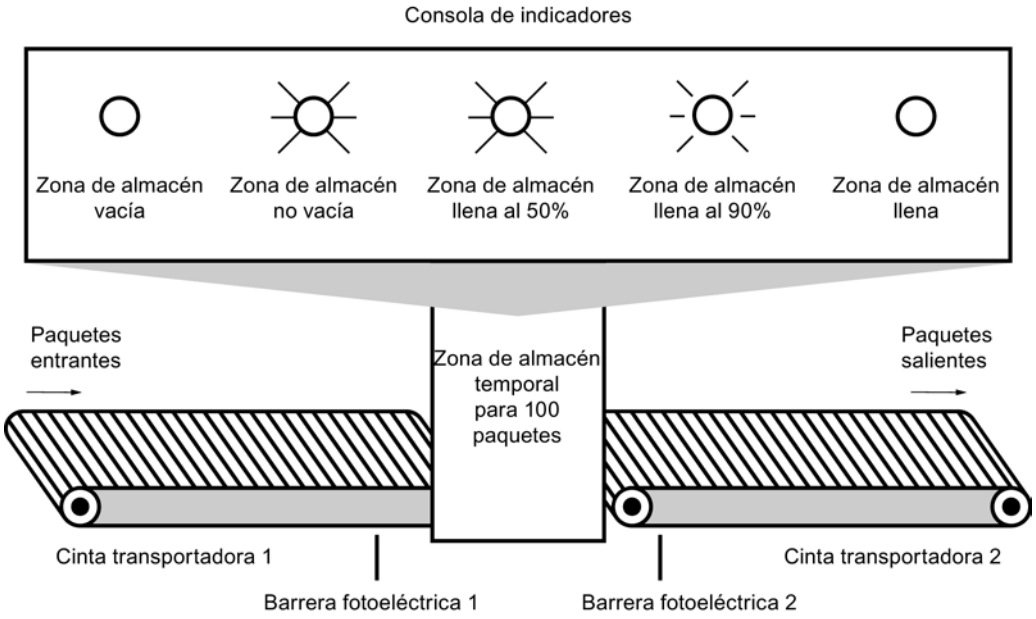
Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta transportadora 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora

2 transporta paquetes desde esta zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
RESET	BOOL	Inicializar contador
LOAD	BOOL	Poner el contador al valor del parámetro "ZW"
EXISTENCIAS	INT	Existencias en almacén al rearrancar
PAQUETES	INT	Número de paquetes en la zona de almacén (valor de contaje actual)
PAQUETES_ZA	BOOL	Se activa si el valor de contaje actual es mayor o igual al valor de la variable "EXISTENCIAS".
ALM_VACIO	BOOL	Indicador luminoso: Zona de almacén vacía

Nombre	Tipo de datos	Descripción
ALM_NO_VACIO	BOOL	Indicador luminoso: Zona de almacén no vacía
ALM_LLENO_50%	BOOL	Indicador luminoso: Zona de almacén llena al 50 %
ALM_LLENO_90%	BOOL	Indicador luminoso: Zona de almacén llena al 90 %
ALM_LLENO	BOOL	Indicador luminoso: Zona de almacén llena
VOLUMEN_50	INT	Valor de comparación: 50 paquetes
VOLUMEN_90	INT	Valor de comparación: 90 paquetes
VOLUMEN_100	INT	Valor de comparación: 100 paquetes

Los segmentos siguientes muestran la programación FUP para activar los indicadores luminosos:

Segmento 1:

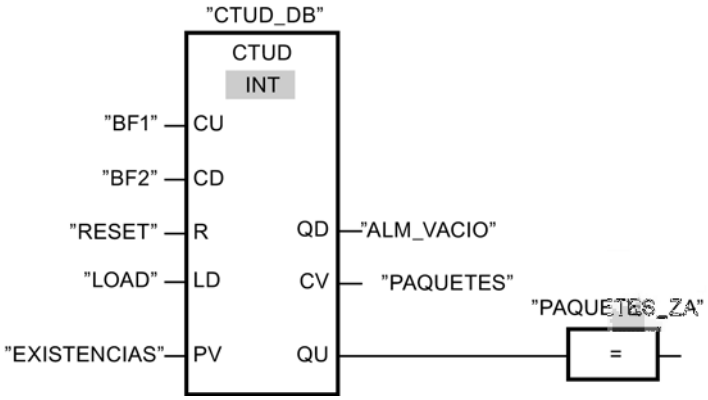
Cuando un paquete se transporta a la zona de almacén, el estado lógico de "BF1" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF1", el contador ascendente se activa y el valor de contaje actual de "PAQUETES" aumenta en uno.

Cuando un paquete se transporta de la zona de almacén a la plataforma de carga, el estado lógico de "BF2" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF2", el contador descendente se activa y el valor actual de "PAQUETES" se reduce en uno.

Si no hay paquetes en la zona de almacén ("PAQUETES" = "0"), la variable "ALM_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén vacía".

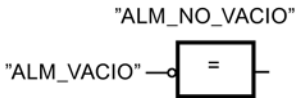
El valor de contaje actual puede ponerse a "0" si la variable "RESET" se pone al estado lógico "1".

Cuando la variable "LOAD" se pone al estado lógico "1", el valor de contaje actual adopta el valor de la variable "EXISTENCIAS". Mientras que el valor de contaje actual sea mayor o igual al valor de la variable "EXISTENCIAS", la variable "PAQUETES_ZA" devolverá el estado lógico "1".



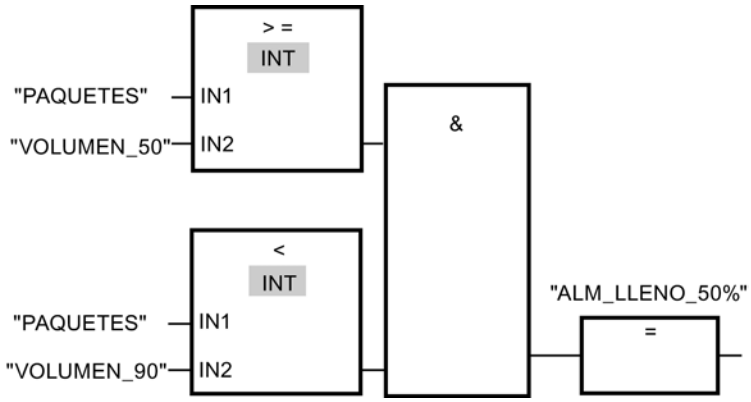
Segmento 2:

Mientras haya paquetes en la zona de almacén, la variable "ALM_NO_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén no vacía".



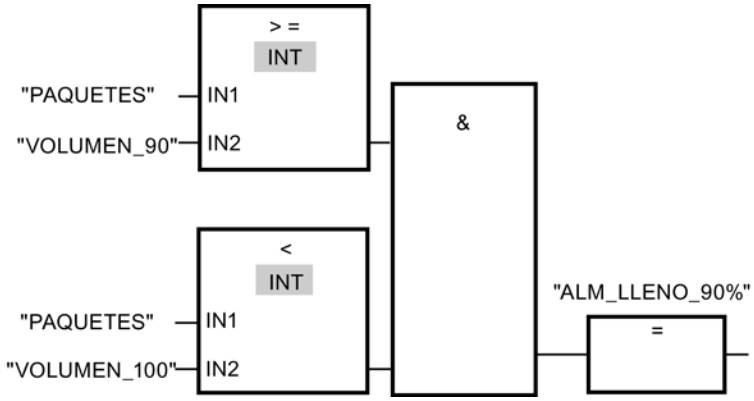
Segmento 3:

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 50, se enciende el indicador luminoso del aviso "Zona de almacén llena al 50 %".



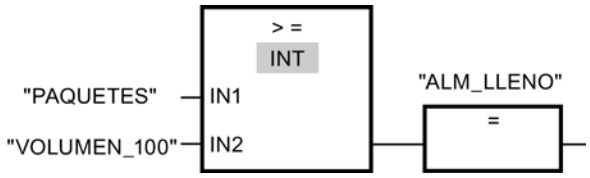
Segmento 4:

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 90, se enciende el indicador luminoso del aviso "Zona de almacén llena al 90 %".



Segmento 5:

Si el número de paquetes que se encuentran en la zona de almacén alcanza 100, se enciende el indicador luminoso del aviso "Zona de almacén llena".



Consulte también

CTUD: Contador ascendente - descendente (Página 3357)

=: Asignación (Página 3595)

CTUD: Contador ascendente - descendente (Página 3668)

CMP >=: Mayor o igual (Página 3687)

CMP <: Menor (Página 3693)

Ejemplo de control de la temperatura ambiente

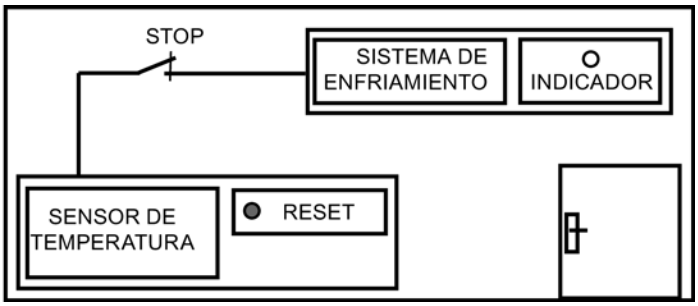
Controlar la temperatura ambiente

La temperatura de una cámara de enfriamiento debe mantenerse por debajo de los cero grados centígrados. Las posibles fluctuaciones de la temperatura se controlan mediante un sensor. Si la temperatura aumenta por encima de los cero grados centígrados, el sistema de enfriamiento se conecta por un tiempo preseleccionado. Durante la fase de enfriamiento se enciende un indicador "Sistema de enfriamiento iniciado".

El sistema de enfriamiento y el indicador luminoso se apagan cuando se cumple una de las condiciones siguientes:

- El sensor indica un descenso de la temperatura por debajo de los cero grados centígrados.
- El tiempo de enfriamiento preseleccionado ha transcurrido.
- Se acciona el pulsador "PARO".

Si ha transcurrido el tiempo de enfriamiento preseleccionado y la temperatura de la cámara de enfriamiento es aún demasiado elevada, el sistema de enfriamiento se puede reiniciar mediante el pulsador "RESET".



Realización

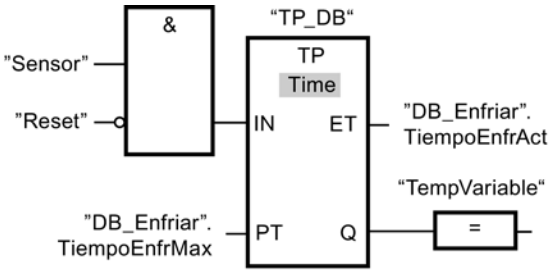
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Comentario
Sensor	BOOL	Señal del sensor de temperatura
RESET	BOOL	Reset

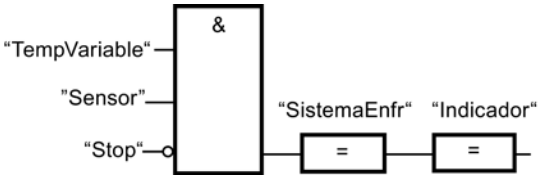
Nombre	Tipo de datos	Comentario
PARO	BOOL	El sistema de enfriamiento se desconecta.
TiempoEnfrMax	TIME	Tiempo de enfriamiento preseleccionado Esta variable está definida en el bloque de datos "DB_Enfriar".
TiempoEnfrAct	TIME	Tiempo de enfriamiento transcurrido actualmente Esta variable está definida en el bloque de datos "DB_Enfriar".
SistemaEnfr	BOOL	El sistema de enfriamiento se conecta.
Indicador	BOOL	El indicador luminoso del aviso "Sistema de enfriamiento iniciado" se enciende.
TempVariable	BOOL	Variable temporal Esta variable guarda el estado lógico del temporizador CEI TP.

El segmento siguiente muestra la programación FUP para controlar la temperatura ambiente:

Segmento 1:



Segmento 2:



Si la temperatura de la cámara de enfriamiento aumenta por encima de los cero grados centígrados, el estado lógico del operando "Sensor" cambia de "0" a "1" (flanco de señal ascendente). Con un flanco de señal ascendente en la entrada IN de la función de temporización se inicia el tiempo de enfriamiento preseleccionado y la "TempVariable" recibe el estado lógico "1". El estado lógico "1" de la "TempVariable" hace que en el segmento 2 se conecte el sistema de enfriamiento y se encienda el indicador luminoso. Las salidas "Sensor", "SistemaEnfr" e "Indicador" tienen que programarse en el segmento 2, ya que en la salida Q de la función de temporización sólo se puede programar una bobina.

Si la temperatura de la cámara de enfriamiento desciende por debajo de los cero grados centígrados, el estado lógico del sensor cambia nuevamente a "0". Con ello se apagan el sistema de enfriamiento y el indicador luminoso.

Si el sensor no indica ningún descenso de la temperatura, el sistema de enfriamiento y el indicador luminoso se apagan a más tardar al transcurrir el tiempo de enfriamiento preseleccionado. En este caso, el proceso de enfriamiento se puede reiniciar mediante el pulsador "RESET". Al pulsar y soltar el pulsador se genera otro flanco ascendente en la entrada IN, con lo que se reinicia el sistema de enfriamiento.

El sistema de enfriamiento y el indicador luminoso se pueden apagar en todo momento con el pulsador "PARO".

Consulte también

TP: Impulso (Página 3615)

=: Asignación (Página 3595)

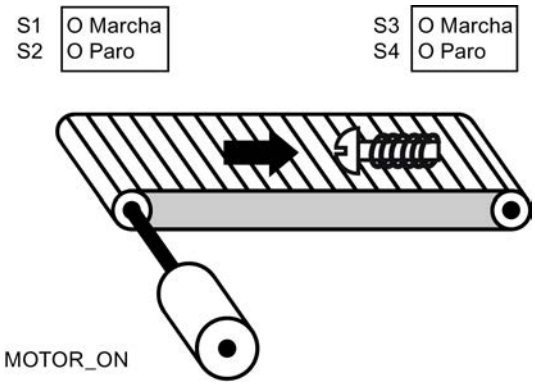
&: Operación lógica Y (Página 3588)

9.7.1.3 Ejemplos de programación AWL

Ejemplo: Operaciones lógicas con bits

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al principio de la cinta transportadora se encuentran dos pulsadores: S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Operando	Declaración	Tipo de datos	Descripción
Marcha_izquierda (S1)	Input	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Paro_izquierda (S2)	Input	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	Input	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	Input	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora
MOTOR_ON/OFF	Output	BOOL	Marcha/paro del motor de la cinta transportadora

El siguiente programa AWL muestra cómo realizar la tarea:

```
AWL                 Explicación
O #S1               // Consultar si el pulsador de marcha S1 es "1"
O #S3               // Consultar si el pulsador de marcha S3 es "1"
S #"MOTOR_ON/OFF"   // Si uno de los pulsadores de marcha (S1 o S3) devuelve el
                    estado lógico "1", se pone en marcha el motor de la cinta
                    transportadora.
O #S2               // Consultar si el pulsador de paro S2 es "1"
O #S4               // Consultar si el pulsador de paro S4 es "1"
R #"MOTOR_ON/OFF"   // Si uno de los pulsadores de paro (S2 o S4) devuelve el estado
                    lógico "1", se para el motor de la cinta transportadora.
```

Consulte también

O: Operación lógica O (Página 6161)

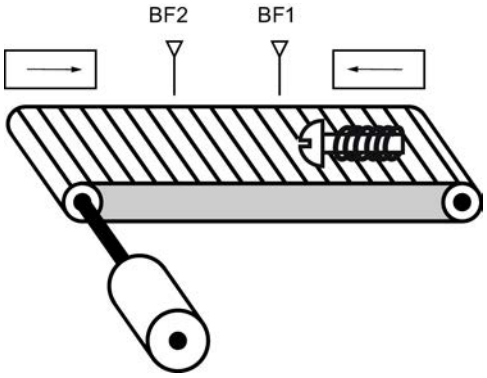
R: Desactivar (Página 6177)

S: Activar (Página 6179)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Declaración	Tipo de datos	Descripción
S1	Input	BOOL	Barrera fotoeléctrica 1
S2	Input	BOOL	Barrera fotoeléctrica 2
TM1	Input	BOOL	Marca de flancos 1
TM2	Input	BOOL	Marca de flancos 2
DERECHA	Output	BOOL	Indicador de desplazamiento a la derecha
IZQUIERDA	Output	BOOL	Indicador de desplazamiento a la izquierda

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
AWL                 Explicación
A #S1               // Consultar si la barrera fotoeléctrica "S1" es "1"
FP #TM1             // Consultar si hay flanco ascendente
AN #S2              // Consultar si la barrera fotoeléctrica "S2" es "0"
S #IZQUIERDA        // Si el estado lógico de la barrera fotoeléctrica "S1" cambia
                    de "0" a "1" (flanco ascendente) y al mismo tiempo el estado
                    lógico de la barrera fotoeléctrica "S2" es "0", el objeto se
                    desplaza hacia la izquierda sobre la cinta transportadora.
                    // Se enciende el indicador de desplazamiento a la izquierda
A #S2               // Consultar si la barrera fotoeléctrica "S2" es "1"
```

AWL	Explicación
FP #TM2	// Consultar si hay flanco ascendente
AN #S1	// Consultar si la barrera fotoeléctrica "S1" es "0"
S #DERECHA	// Si el estado lógico de la barrera fotoeléctrica "S2" cambia de "0" a "1" (flanco ascendente) y al mismo tiempo el estado lógico de "S1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora.
	// Se enciende el indicador de desplazamiento a la derecha
AN #S1	// Consultar si la barrera fotoeléctrica "S1" es "0"
AN #S2	// Consultar si la barrera fotoeléctrica "S2" es "0"
R #IZQUIERDA	// El indicador de desplazamiento hacia la izquierda se apaga cuando el estado lógico de ambas barreras fotoeléctricas es "0".
R #DERECHA	// El indicador de desplazamiento hacia la derecha se apaga cuando el estado lógico de ambas barreras fotoeléctricas es "0".

Consulte también

O: Operación lógica O (Página 6161)

R: Desactivar (Página 6177)

S: Activar (Página 6179)

A: Operación lógica Y (Página 6158)

AN: Operación lógica Y negada (Página 6159)

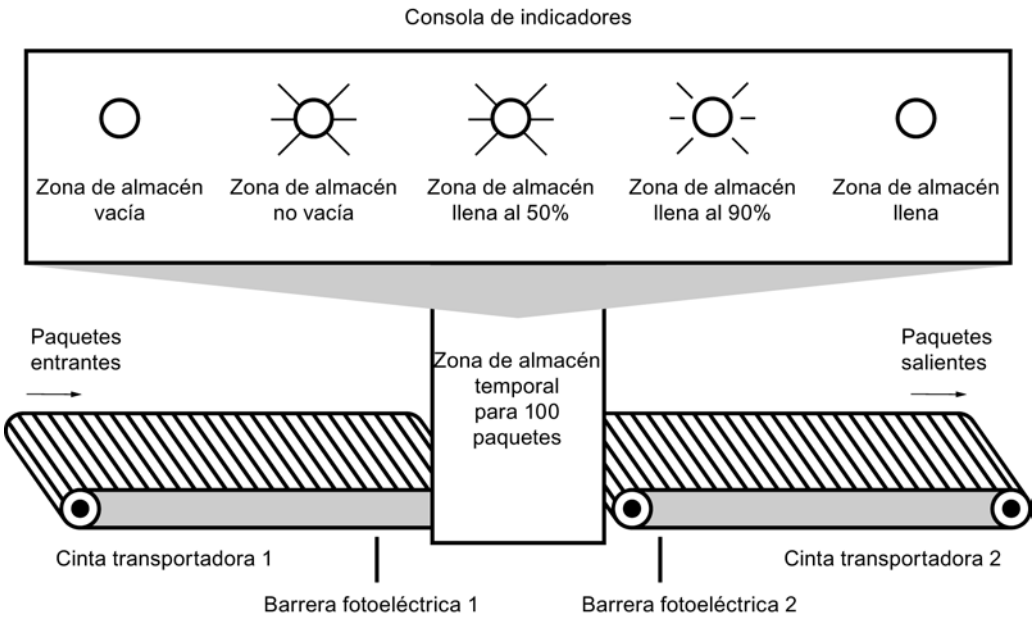
FP: Consultar flanco de señal ascendente del RLO (Página 6185)

Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta transportadora 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde la zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Dirección	Descripción
PACKAGECOUNT	COUNTER	C1	Número de paquetes en la zona de almacén (valor de contaje actual)

Nombre	Sección	Tipo de datos	Descripción
LS1	Input	BOOL	Barrera fotoeléctrica 1
LS2	Input	BOOL	Barrera fotoeléctrica 2
ALM_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén vacía
ALM_NO_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén no vacía
ALM_LLENO_50%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 50%
ALM_LLENO_90%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 90%
ALM_LLENO	Output	BOOL	Indicador luminoso: Zona de almacén llena

El siguiente programa AWL muestra cómo realizar el ejemplo:

AWL	Explicación
A #LS1	// Consultar si la barrera fotoeléctrica "LS1" es "1".
CU "PACKAGECOUNT"	// Cuando se produce un flanco ascendente en la barrera fotoeléctrica "LS1", se incrementa en uno el valor de contaje del contador "PACKAGECOUNT".
A #LS2	// Consultar si la barrera fotoeléctrica "LS2" es "1".
CD "PACKAGECOUNT"	// Cuando se produce un flanco ascendente en la barrera fotoeléctrica "LS2", se incrementa en uno el valor de contaje del contador "PACKAGECOUNT".
AN "PACKAGECOUNT"	// Consultar si el valor de contaje es "0".
= #ALM_VACIO	// Cuando el valor de contaje es "0", se enciende el indicador luminoso "Zona de almacén vacía"
A "PACKAGECOUNT"	// Consultar si el valor de contaje es "1".
= #ALM_NO_VACIO	// Cuando el valor de contaje es "0", se enciende el indicador luminoso "Zona de almacén vacía".
L 50	// Cargar el valor de comparación "50" en el acumulador 1.
L "PACKAGECOUNT"	// Desplazar el valor de comparación al acumulador 2.
	// Cargar el valor de contaje actual en el acumulador 1.
<=I	// Comparar valores
= #"ALM_LLENO_50%"	// Si el valor de contaje es mayor o igual a "50" se enciende el indicador luminoso "Zona de almacén llena al 50%"
L 90	// Desplazar el valor de contaje al acumulador 2.
	// Cargar el valor de comparación "90" en el acumulador 1.
>=I	// Comparar valores
= #"ALM_LLENO_90%"	// Si el valor de contaje es mayor o igual a "90" se enciende el indicador luminoso "Zona de almacén llena al 90%".
L "PACKAGECOUNT"	// Cargar el valor de contaje actual en el acumulador 1.
L 100	// Desplazar el valor de contaje al acumulador 2.
	// Cargar el valor de comparación "100" en el acumulador 1.
>=I	// Comparar valores
= #ALM_LLENO	// Si el valor de contaje es mayor o igual a "100" se enciende el indicador luminoso "Zona de almacén llena".

Consulte también

O: Operación lógica O (Página 6161)

R: Desactivar (Página 6177)

S: Activar (Página 6179)

A: Operación lógica Y (Página 6158)

AN: Operación lógica Y negada (Página 6159)

FP: Consultar flanco de señal ascendente del RLO (Página 6185)

=: Asignación (Página 6176)

CU: Contador ascendente (Página 6212)

CD: Contador descendente (Página 6213)

L: Cargar (Página 6251)

Ejemplo de cálculo de una ecuación

Cálculo de una ecuación

El ejemplo de programación siguiente muestra cómo obtener con tres operaciones aritméticas el mismo resultado que la siguiente ecuación:

RESULT = ((A + B) x C) /D

Realización

La tabla siguiente muestra la declaración de los operandos utilizados en la tabla de variables PLC:

Nombre	Tipo de datos	Comentario
A	INT	Primer sumando
B	INT	Segundo sumando
C	INT	Multiplicador
D	INT	Divisor
RESULT	INT	Resultado final

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
 AWL                 Explicación
L "A"                // Cargar el valor del operando "A" en el acumulador 1
L "B"                // Desplazar el valor del operando "A" al acumulador 2
                     // Cargar el valor del operando "B" en el acumulador 1
+I                   // Sumar los valores de los acumuladores 1 y 2
                     // Guardar suma en el acumulador 1
L "C"                // Desplazar la suma al acumulador 2
                     // Cargar el valor del operando "C" en el acumulador 1
*I                   // Multiplicar los valores de los acumuladores 1 y 2
                     // Guardar el producto en el acumulador 1
L "D"                // Desplazar el producto al acumulador 2
                     // Cargar el valor del operando "D" en el acumulador 1
/I                   // Dividir el valor del acumulador 2 por el valor del acumulador
                     1
                     // Guardar el resultado en el acumulador 1
T "RESULT"           // Transferir el resultado al operando "RESULT"
```

Consulte también

+I: Sumar enteros (16 bits) (Página 6220)

-I: Restar enteros (16 bits) (Página 6221)

*I: Multiplicar enteros (16 bits) (Página 6222)

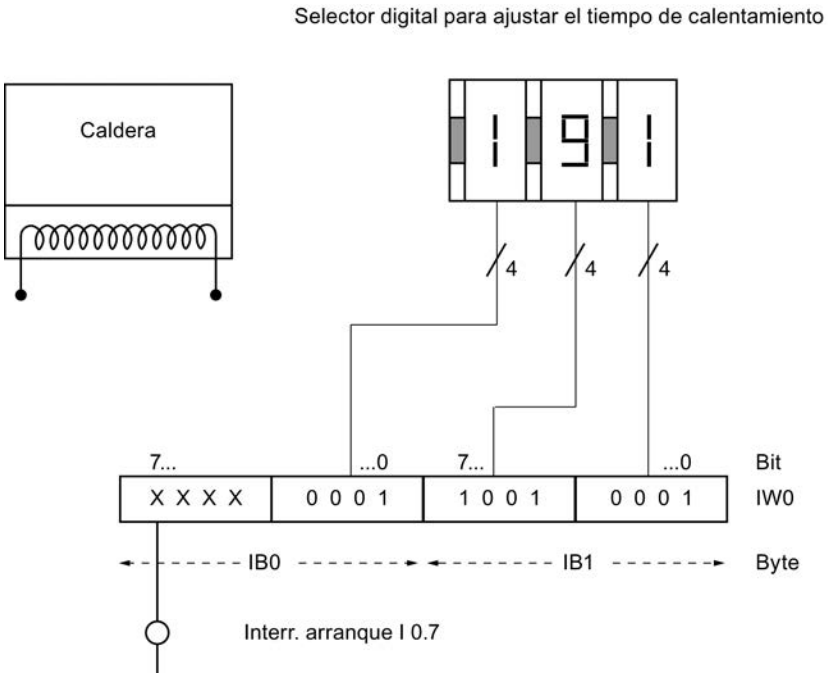
L: Cargar (Página 6251)

T: Transferir (Página 6260)

Ejemplo: Operaciones lógicas con palabras

Calentamiento de una caldera

La imagen siguiente muestra una caldera que se calienta con un interruptor de arranque. Al pulsar el interruptor de marcha, se inicia el proceso de calentamiento. El tiempo de calentamiento se establece mediante selectores digitales. El tiempo se indica en segundos en formato BCD.



Realización

La tabla siguiente muestra la declaración de los operandos utilizados en la tabla de variables PLC:

Nombre	Tipo de datos	Dirección	Comentario
DURATION	WORD	EW0	Tiempo de calentamiento en segundos • I1.0 a I1.3: Selector de unidades • I1.4 a I1.7: Selector de decenas • I0.0 a I0.3: Selector de centenas
HEATING	TIMER	T1	Temporizador que arranca con el tiempo de calentamiento preseleccionado.

La tabla siguiente muestra la declaración de los operandos utilizados en la interfaz del bloque lógico:

Nombre	Sección	Tipo de datos	Comentario
START	Input	BOOL	Interruptor de arranque
START_HEATING	Output	BOOL	Inicio del proceso de calentamiento

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
AWL                 Explicación
A "HEATING"         // Consultar si el temporizador ha arrancado
= #START_HEATING    // Iniciar proceso de calentamiento
BEC                 // Si RLO=1 finalizar la ejecución del bloque
                    // De este modo se impide que rearranque el temporizador
                    "HEATING" cuando se pulse el pulsador de arranque
L "DURATION"        // Cargar el tiempo de calentamiento en el acumulador 1
AW W#16#0FFF        // Poner los bits de entrada I0.4 a I0.7 a "0"
OW W#16#4000        // Ajustar segundos en los bits I1.2 e I1.3 del acumulador 1
A #START            // Consultar si el pulsador de arranque es "1"
SE "HEATING"        // Cuando se produzca un flanco ascendente en el interruptor de
                    arranque, iniciar el proceso de calentamiento como impulso
                    prolongado
```

Consulte también

L: Cargar (Página 6251)

SE: Arrancar temporizador como impulso prolongado (Página 6196)

A: Operación lógica Y (Página 6158)

=: Asignación (Página 6176)

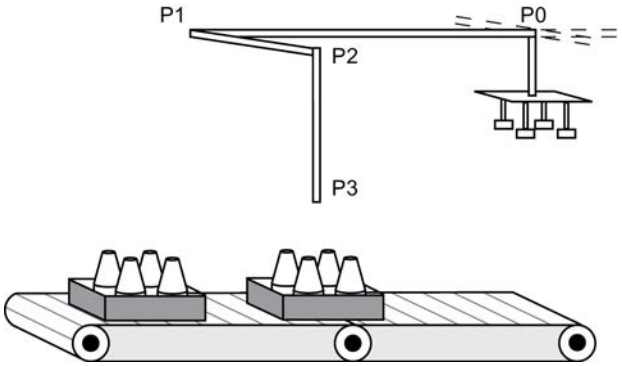
AW: Operación lógica Y por palabras (Página 6333)

OW: Operación lógica O por palabras (Página 6335)

Ejemplo de una cadena secuencial

Programación de una cadena secuencial

La imagen siguiente muestra una estación para despaletizar envases de vidrio. Los palets se transportan sobre una cinta transportadora a la estación. Cuando un palet llega con envases de vidrio a la estación, la cinta transportadora se detiene y un brazo robótico se desplaza desde su posición inicial (P0) hasta alcanzar la posición sobre el palet (P2). Si el brazo robótico se encuentra sobre el palet, se abren las garras y el brazo desciende. La posición actual del brazo y el estado de las garras son registrados por sensores. El desarrollo descrito del movimiento del brazo robótico se realiza en este ejemplo mediante una cadena secuencial. También pueden programarse los pasos sucesivos para retirar los envases y transportarlos a otra cinta transportadora.



Realización

La tabla siguiente muestra la declaración de los operandos utilizados en la tabla de variables PLC:

Nombre	Tipo de datos	Comentario
NUMBER	INT	Número de etapa
Tag_Error	BOOL	Operando que se activa si el número de la etapa es mayor que 3 o si una etapa no ha sido ejecutada.

La tabla siguiente muestra la declaración de los operandos utilizados en la interfaz del bloque lógico:

Nombre	Sección	Tipo de datos	Comentario
POS_0	Input	BOOL	Brazo en posición inicial (P0)
POS_1	Input	BOOL	Brazo en posición 1 (P1)
POS_2	Input	BOOL	Brazo en posición 2 (P2)
GRIPPER_OPEN	Input	BOOL	Garras abiertas
OUT_POS_1	Output	BOOL	Desplazar el brazo a la posición 1
OUT_POS_2	Output	BOOL	Desplazar el brazo a la posición 2
OUT_GRIPPER	Output	BOOL	Abrir las garras
OUT_POS_3	Output	BOOL	Desplazar el brazo a la posición 3

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
AWL                          Explicación
L "NUMBER"                   // Cargar el número de etapa en el acumulador 1.
JL END                       // Principio de la lista de saltos
JU POSITION_0                // Si el acumulador 1 contiene el valor "0", saltar a
                             la etiqueta "POSITION_0".
JU POSITION_1                // Si el acumulador 1 contiene el valor "1", saltar a
                             la etiqueta "POSITION_1".
JU POSITION_2                // Si el acumulador 1 contiene el valor "2", saltar a
                             la etiqueta "POSITION_2".
JU POSITION_3                // Si el acumulador 1 contiene el valor "3", saltar a
                             la etiqueta "POSITION_3".
END: JU ERROR                // Fin de la lista de saltos
                             // Si el número de etapa es mayor que 3, saltar a la
                             etiqueta "ERROR".
POSITION_0: A #POS_0         // Etiqueta "POSITION_0"
                             // Consultar si el brazo se encuentra en la posición
                             inicial (P0).
= #OUT_POS_1                 // Si se cumple la condición, activar la salida
                             "OUT_POS_1" y desplazar el brazo a la posición 1 (P1).
JCN ERROR                    // Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT                      // Si el RLO es "1", saltar a la etiqueta "NEXT".
POSITION_1: A #POS_1         // Etiqueta "POSITION_1"
                             // Consultar si el brazo se encuentra en la posición
                             1 (P1).
= #OUT_POS_2                 // Si se cumple la condición, activar la salida
                             "OUT_POS_2" y desplazar el brazo a la posición 2 (P2).
JCN ERROR                    // Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT                      // Si el RLO es "1", saltar a la etiqueta "NEXT".
POSITION_2: A #POS_2         // Etiqueta "POSITION_2"
                             // Consultar si el brazo se encuentra en la posición
                             2 (P2).
```

AWL	Explicación
= #OUT_GRIPPER	// Si se cumple la condición, activar la salida "OUT_GRIPPER" y abrir las garras.
JCN ERROR	// Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT	// Si el RLO es "1", saltar a la etiqueta "NEXT".
POSITION_3: A #POS_2	// Etiqueta "POSITION_3"
	// Consultar si el brazo se encuentra en la posición 2 (P2).
A #GRIPPER_OPEN	// Consultar si las garras están abiertas
= #OUT_POS_3	// Si se cumple la condición, activar la salida "OUT_POS_3" y desplazar el brazo a la posición 3 (P3).
JCN ERROR	// Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT	// Si el RLO es "1", saltar a la etiqueta "NEXT".
NEXT: INC 1	// Etiqueta "NEXT"
	// Incrementar el número de etapa en el acumulador 1.
T "NUMBER"	// Transferir el número de etapa al operando "NUMBER".
L 3	// Desplazar el número de etapa al acumulador 2.
	// Cargar el valor 3 en el acumulador 1.
>I	// Consultar si el número de la etapa actual es mayor que 3.
JC RESET_NUMBER	// Si el resultado de la consulta es "1", saltar a la etiqueta "RESET_NUMBER" y continuar allí el procesamiento del programa.
BEU	// Finalizar bloque
RESET_NUMBER: L 0	// Etiqueta "RESET_NUMBER"
	// Cargar el valor "0" en el acumulador 1.
T "NUMBER"	// Asignar el valor "0" al operando "NUMBER" (número de etapa).
BEU	// Finalizar bloque
ERROR: NOT	// Etiqueta "ERROR"
= "Tag_Error"	// Asignar el RLO negado al operando "Tag_Error".
BEU	// Finalizar bloque

Consulte también

A: Operación lógica Y (Página 6158)

=: Asignación (Página 6176)

L: Cargar (Página 6251)

T: Transferir (Página 6260)

? I: Comparación de enteros (Página 6215)

JU: Salto incondicional (Página 6291)

JC: Saltar si RLO = 1 (Página 6292)

JCN: Saltar si RLO = 0 (Página 6293)

JL: Definir lista de saltos (Página 6316)

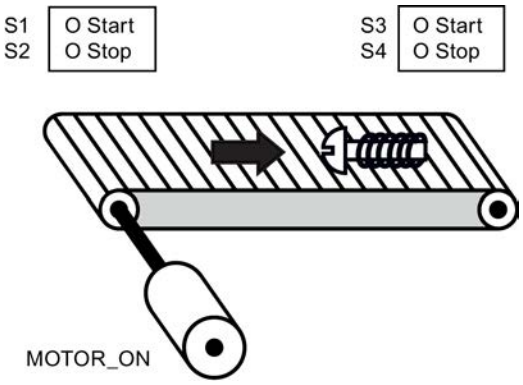
BEU: Fin de bloque incondicional (Página 6328)

9.7.1.4 Ejemplos de programación SCL

Ejemplo: Operaciones lógicas con bits

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al comienzo de la cinta transportadora se encuentran dos pulsadores, S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Operando	Declaración	Tipo de datos	Descripción
Marcha_izquierda (S1)	Input	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Marcha_izquierda (S2)	Input	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	Input	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	Input	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora

Operando	Declaración	Tipo de datos	Descripción
MOTOR_ON	Output	BOOL	Conectar el motor de la cinta transportadora
MOTOR_OFF	Output	BOOL	Desconectar el motor de la cinta transportadora

El siguiente programa SCL muestra cómo realizar la tarea:

SCL

```
IF "Marcha_izquierda" OR "Marcha_derecha" THEN
"MOTOR_ON" := 1;
END_IF;
IF "Paro_izquierda" OR "Paro_derecha" THEN
"MOTOR_OFF" := 1;
END_IF;
```

El motor de la cinta transportadora se enciende cuando se acciona el pulsador de marcha "Marcha_izquierda" o "Marcha_derecha". El motor de la cinta transportadora se apaga cuando se acciona el pulsador de paro "Paro_izquierda" o "Paro_derecha".

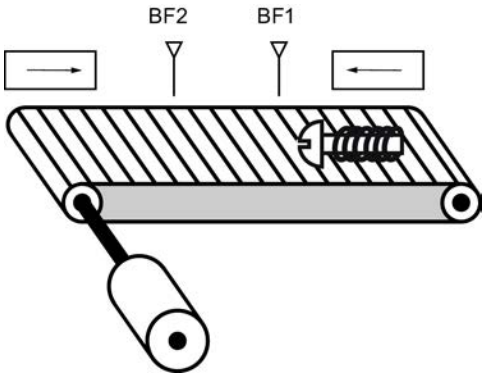
Consulte también

Expresiones lógicas (Página 2792)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Declaración	Tipo de datos	Descripción
Barrera fotoeléctrica LS1	Input	BOOL	Barrera fotoeléctrica 1
Barrera fotoeléctrica LS2	Input	BOOL	Barrera fotoeléctrica 2
DERECHA	Output	BOOL	Indicador de desplazamiento a la derecha
IZQUIERDA	Output	BOOL	Indicador de desplazamiento a la izquierda

El siguiente programa SCL muestra cómo realizar el ejemplo:

SCL

```
IF "BF1" = 1 AND NOT "BF2" = 0
THEN "IZQUIERDA" = 1;
IF "BF2" = 1 AND NOT "BF1" = 0
THEN "DERECHA" := 1;
IF "BF2" = 0
THEN "DERECHA" := 0;
IF "BF1" = 0
THEN "IZQUIERDA" := 0;
```

Si el estado lógico de la barrera fotoeléctrica "BF1"es "1" y, simultáneamente, el estado lógico de la barrera fotoeléctrica "BF2" es "0", el objeto se desplaza hacia la izquierda sobre la cinta transportadora. Si el estado lógico de la barrera fotoeléctrica "BF2"es "1" y, simultáneamente, el estado lógico de la barrera fotoeléctrica "BF1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora. Los indicadores de desplazamiento hacia la izquierda se apagan cuando el estado lógico de ambas barreras fotoeléctricas es "0".

Consulte también

Expresiones lógicas (Página 2792)

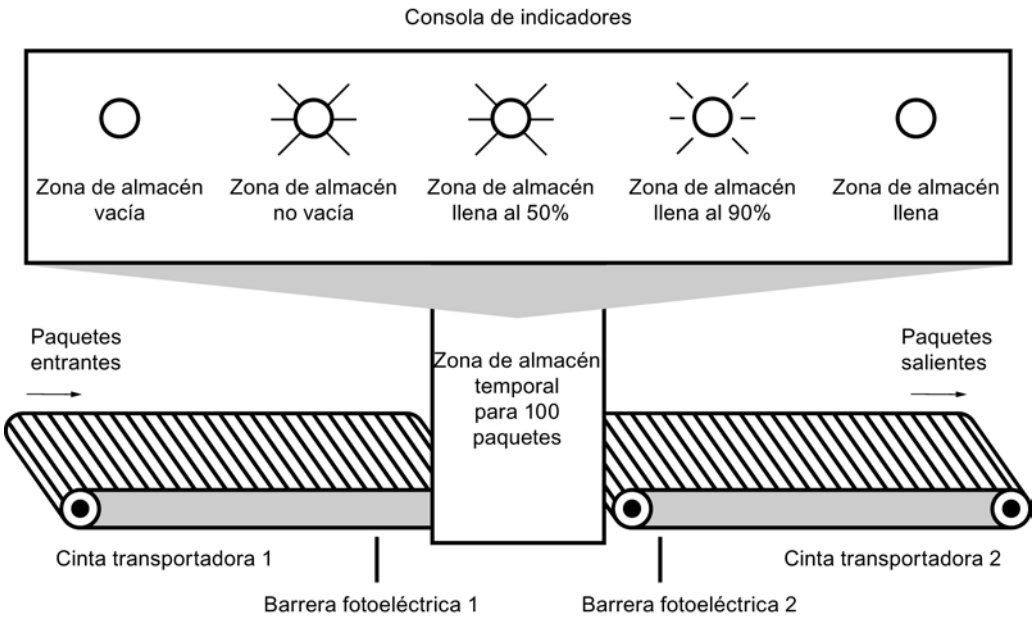
Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta transportadora 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde esta zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben

transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Declaración	Tipo de datos	Descripción
BF1	Input	BOOL	Barrera fotoeléctrica 1
BF2	Input	BOOL	Barrera fotoeléctrica 2
RESET	Input	BOOL	Inicializar contador
LOAD	Input	BOOL	Poner el contador al valor del parámetro "ZW"
EXISTENCIAS	Input	INT	Existencias en almacén al rearrancar
PAQUETES	Output	INT	Número de paquetes en la zona de almacén (valor de contaje actual)
PAQUETES_ZA	Output	BOOL	Se activa si el valor de contaje actual es mayor o igual al valor de la variable "EXISTENCIAS".
ALM_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén vacía

Nombre	Declaración	Tipo de datos	Descripción
ALM_NO_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén no vacía
ALM_LLENO_50%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 50 %
ALM_LLENO_90%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 90 %
ALM_LLENO	Output	BOOL	Indicador luminoso: Zona de almacén llena
VOLUMEN_50	Input	INT	Valor de comparación: 50 paquetes
VOLUMEN_90	Input	INT	Valor de comparación: 90 paquetes
VOLUMEN_100	Input	INT	Valor de comparación: 100 paquetes

El siguiente programa SCL muestra cómo realizar el ejemplo:

Cuando un paquete se transporta a la zona de almacén, el estado lógico de "BF1" cambia de "0" a "1" (flanco de señal ascendente). Cuando se produce un flanco ascendente en "BF1", el contador ascendente se activa y el valor de contaje actual de "PAQUETES" aumenta en uno.

Cuando un paquete se transporta de la zona de almacén a la plataforma de carga, el estado lógico de "BF2" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF2", el contador descendente se activa y el valor actual de "PAQUETES" se reduce en uno.

Si no hay paquetes en la zona de almacén ("PAQUETES" = "0"), la variable "ALM_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén vacía".

El valor de contaje actual puede ponerse a "0" si la variable "RESET" se pone al estado lógico "1".

Cuando la variable "LOAD" se pone al estado lógico "1", el valor de contaje actual adopta el valor de la variable "EXISTENCIAS". Mientras que el valor de contaje actual sea mayor o igual al valor de la variable "EXISTENCIAS", la variable "PAQUETES_ZA" devolverá el estado lógico "1".

SCL

```
"CTUD_DB".CTUD(CU := "BF1",
               CD := "BF2",
               R  := "RESET",
               LD := "LOAD",
               PV := "EXISTENCIAS",
               QU := "PAQUETES_ZA",
               QD := "ALM_VACIO",
               CV := "PAQUETES");
```

Mientras haya paquetes en la zona de almacén, la variable "ALM_NO_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén no vacía".

SCL

```
"ALM_NO_VACIO" := NOT "ALM_VACIO"
```

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 50, se enciende el indicador luminoso del aviso "Zona de almacén llena al 50 %".

SCL

```
IF "PAQUETES" >= "VOLUMEN_50" THEN "ALM_LLENO_50%" := 1;
IF "PAQUETES" <= "VOLUMEN_90" THEN "ALM_LLENO_50%" := 1;
```

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 90, se enciende el indicador luminoso del aviso "Zona de almacén llena al 90 %".

SCL

```
IF "PAQUETES" >= "VOLUMEN_90" THEN "ALM_LLENO_90%" := 1;
IF "PAQUETES" < "VOLUMEN_100" THEN "ALM_LLENO_90%" := 1;
```

Si el número de paquetes que se encuentran en la zona de almacén alcanza 100, se enciende el indicador luminoso del aviso "Zona de almacén llena".

SCL

```
IF "PAQUETES" >= "VOLUMEN_100" THEN "ALM_LLENO" := 1;
```

Consulte también

Expresiones lógicas (Página 2792)

9.7.2 Ejemplos de programación (S7-300, S7-400)

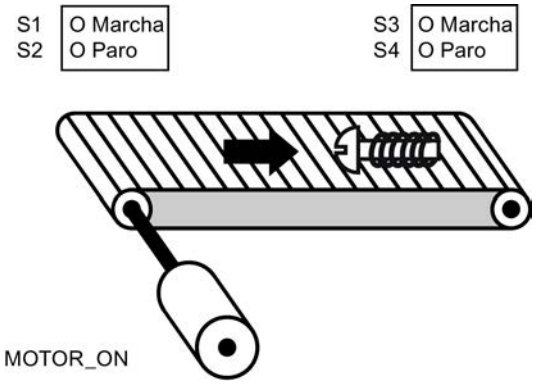
9.7.2.1 Ejemplos de programación KOP

Ejemplo de control de una cinta transportadora

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al principio de la cinta transportadora se encuentran dos pulsadores: S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para

MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

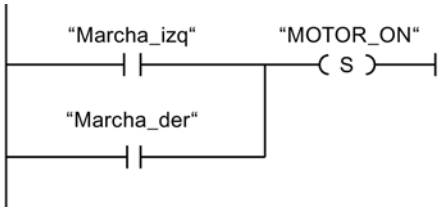
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
Marcha_izquierda (S1)	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Paro_izquierda (S2)	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora
MOTOR_ON	BOOL	Conectar el motor de la cinta transportadora

Los segmentos siguientes muestran la programación KOP para solucionar la tarea:

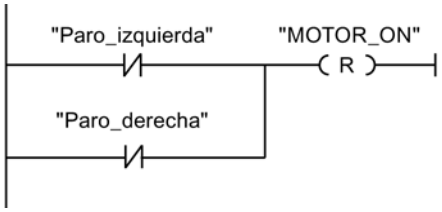
Segmento 1:

El motor de la cinta transportadora se conecta cuando se acciona el pulsador de marcha "S1" o "S3".



Segmento 2:

El motor de la cinta transportadora se desconecta cuando se acciona el pulsador de paro "S2" o "S4".



Consulte también

---(S)---: Activar salida (Página 5607)

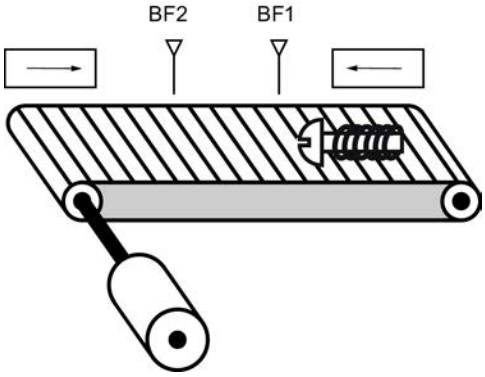
---| |---: Contacto normalmente abierto (Página 5602)

---(R)---: Desactivar salida (Página 5606)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

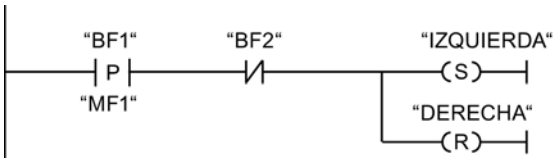
Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
DERECHA	BOOL	Indicador de desplazamiento a la derecha

Nombre	Tipo de datos	Descripción
IZQUIERDA	BOOL	Indicador de desplazamiento a la izquierda
MF1	BOOL	Marca de flancos 1
MF2	BOOL	Marca de flancos 2

Los segmentos siguientes muestran la programación KOP para solucionar la tarea:

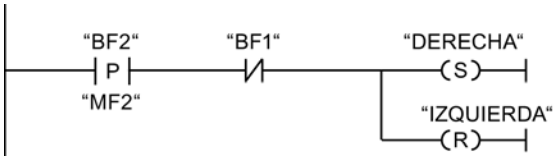
Segmento 1:

Si el estado lógico de la barrera fotoeléctrica "BF1" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF2" es "0", el objeto se desplaza hacia la izquierda sobre la cinta transportadora.



Segmento 2:

Si el estado lógico de la barrera fotoeléctrica "BF2" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora.



Consulte también

--|P|--: Consultar flanco de señal ascendente de un operando (Página 5612)

---(S)---: Activar salida (Página 5607)

---| |---: Contacto normalmente abierto (Página 5602)

---(R)---: Desactivar salida (Página 5606)

---| / |---: Contacto normalmente cerrado (Página 5603)

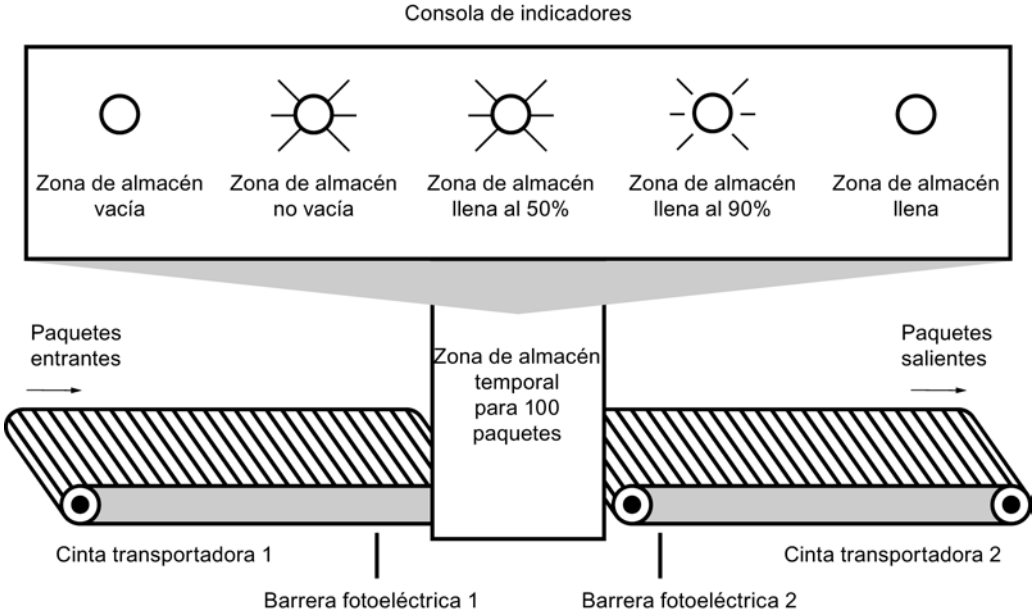
Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde esta zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera

fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
RESET	BOOL	Inicializar contador
LOAD	BOOL	Poner el contador al valor del parámetro "PV"
EXISTENCIAS	INT	Existencias en almacén al rearrancar
PAQUETES	INT	Número de paquetes en la zona de almacén (valor de contaje actual)
PAQUETES_ZA	BOOL	Se activa si el valor de contaje actual es mayor o igual al valor de la variable "EXISTENCIAS".
ALM_VACIO	BOOL	Indicador luminoso: zona de almacén vacía
ALM_NO_VACIO	BOOL	Indicador luminoso: zona de almacén no vacía

Nombre	Tipo de datos	Descripción
ALM_LLENO_50%	BOOL	Indicador luminoso: zona de almacén llena al 50 %
ALM_LLENO_90%	BOOL	Indicador luminoso: zona de almacén llena al 90 %
ALM_LLENO	BOOL	Indicador luminoso: zona de almacén llena
VOLUMEN_50	INT	Valor de comparación: 50 paquetes
VOLUMEN_90	INT	Valor de comparación: 90 paquetes
VOLUMEN_100	INT	Valor de comparación: 100 paquetes

Los segmentos siguientes muestran la programación KOP para activar los indicadores luminosos:

Segmento 1:

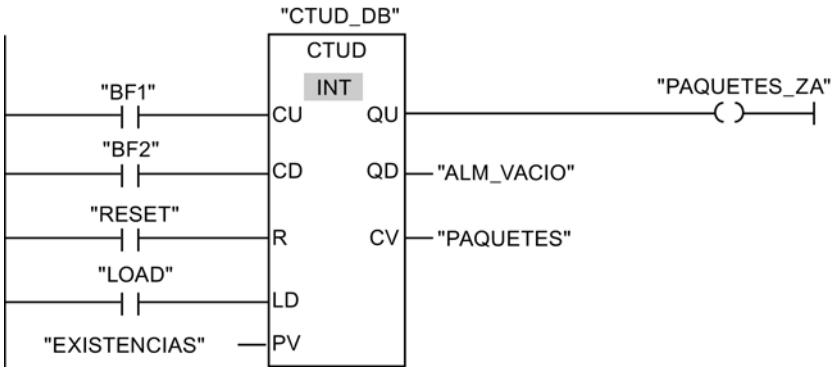
Cuando un paquete se transporta a la zona de almacén, el estado lógico de "BF1" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF1", el contador ascendente se activa y el valor de contaje actual de "PAQUETES" aumenta en uno.

Cuando un paquete se transporta de la zona de almacén a la plataforma de carga, el estado lógico de "BF2" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF2", el contador descendente se activa y el valor actual de "PAQUETES" se reduce en uno.

Si no hay paquetes en la zona de almacén ("PAQUETES" = "0"), la variable "ALM_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén vacía".

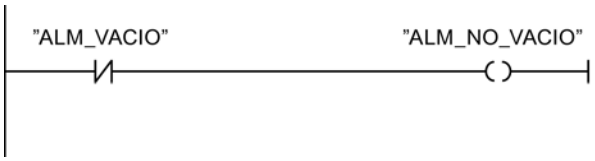
El valor de contaje actual puede ponerse a "0" si la variable "RESET" se pone al estado lógico "1".

Cuando la variable "LOAD" se pone al estado lógico "1", el valor de contaje actual adopta el valor de la variable "EXISTENCIAS". Mientras que el valor de contaje actual sea mayor o igual al valor de la variable "EXISTENCIAS", la variable "PAQUETES_ZA" devolverá el estado lógico "1".



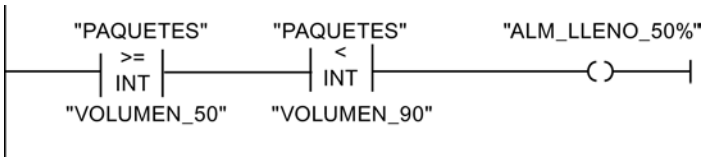
Segmento 2:

Mientras haya paquetes en la zona de almacén, la variable "ALM_NO_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén no vacía".



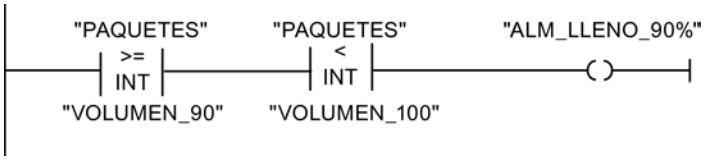
Segmento 3:

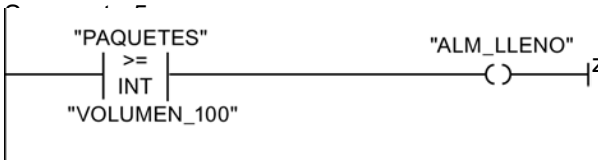
Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 50, se enciende el indicador luminoso del aviso "Zona de almacén llena al 50 %".



Segmento 4:

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 90, se enciende el indicador luminoso del aviso "Zona de almacén llena al 90 %".





zona de almacén alcanza 100, se enciende llena".

Consulte también

---( )---: Asignación (Página 5605)

CMP >=: Mayor o igual (Página 5667)

CMP <: Menor (Página 5670)

Ejemplo de cálculo de una ecuación

Cálculo de una ecuación

El ejemplo de programación siguiente muestra cómo obtener con tres operaciones aritméticas el mismo resultado que la siguiente ecuación:

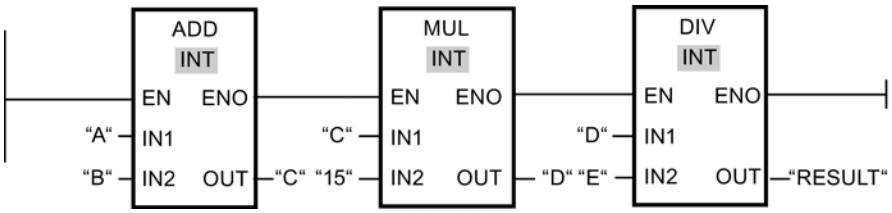
RESULT = ((A + B) x 15) / E

Aplicación

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Comentario
A	INT	Primer sumando
B	INT	Segundo sumando
C	INT	Primer resultado intermedio
15	INT	Multiplicador
D	INT	Segundo resultado intermedio
E	INT	Divisor
RESULT	INT	Resultado final

El segmento siguiente muestra la programación KOP para calcular la ecuación:



El valor del operando "A" se suma al valor del operando "B". La suma se almacena en el operando "C". El valor del operando "C" se multiplica por "15". El resultado de la multiplicación se almacena en el operando "D". A continuación, el valor almacenado en el operando "D" se divide por el valor del operando "E". El resultado final se almacena en el operando "RESULT".

Consulte también

ADD: Sumar (Página 5672)

MUL: Multiplicar (Página 5675)

DIV: Dividir (Página 5676)

Ejemplo de control de la temperatura ambiente

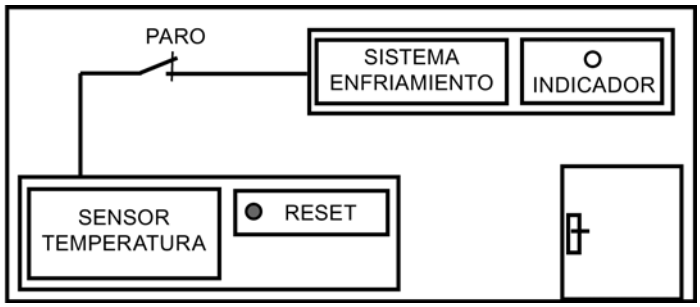
Controlar la temperatura ambiente

La temperatura de una cámara de enfriamiento debe mantenerse por debajo de los cero grados centígrados. Las posibles fluctuaciones de la temperatura se controlan mediante un sensor. Si la temperatura aumenta por encima de los cero grados centígrados, el sistema de enfriamiento se conecta por un tiempo preseleccionado. Durante la fase de enfriamiento se enciende un indicador "Sistema de enfriamiento iniciado".

El sistema de enfriamiento y el indicador luminoso se apagan cuando se cumple una de las condiciones siguientes:

- El sensor indica un descenso de la temperatura por debajo de los cero grados centígrados.
- El tiempo de enfriamiento preseleccionado ha transcurrido.
- Se acciona el pulsador "PARO".

Si ha transcurrido el tiempo de enfriamiento preseleccionado y la temperatura de la cámara de enfriamiento es aún demasiado elevada, el sistema de enfriamiento se puede reiniciar mediante el pulsador "RESET".



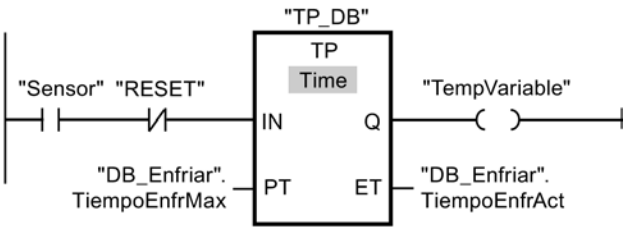
Realización

La tabla siguiente muestra la definición de las variables utilizadas:

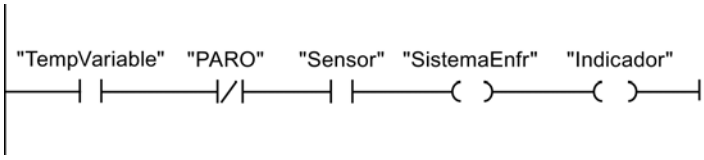
Nombre	Tipo de datos	Comentario
Sensor	BOOL	Señal del sensor de temperatura
RESET	BOOL	Reset
PARO	BOOL	El sistema de enfriamiento se desconecta.
TiempoEnfrMax	TIME	Tiempo de enfriamiento preseleccionado Esta variable está definida en el bloque de datos "DB_Enfriar".
TiempoEnfrAct	TIME	Tiempo de enfriamiento transcurrido actualmente Esta variable está definida en el bloque de datos "DB_Enfriar".
SistemaEnfr	BOOL	El sistema de enfriamiento se conecta.
Indicador	BOOL	El indicador luminoso del aviso "Sistema de enfriamiento iniciado" se enciende.
TempVariable	BOOL	Variable temporal Esta variable guarda el estado lógico del temporizador CEI TP.

El segmento siguiente muestra la programación KOP para controlar la temperatura ambiente:

Segmento 1:



Segmento 2:



Si la temperatura de la cámara de enfriamiento aumenta por encima de los cero grados centígrados, el estado lógico del operando "Sensor" cambia de "0" a "1" (flanco de señal ascendente). Con un flanco de señal ascendente en la entrada IN de la función de temporización se inicia el tiempo de enfriamiento preseleccionado y la "TempVariable" recibe el estado lógico "1". El estado lógico "1" de la "TempVariable" hace que en el segmento 2 se conecte el sistema de enfriamiento y se encienda el indicador luminoso. Las salidas "Sensor", "SistemaEnfr" e "Indicador" tienen que programarse en el segmento 2, ya que en la salida Q de la función de temporización sólo se puede programar una bobina.

Si la temperatura de la cámara de enfriamiento desciende por debajo de los cero grados centígrados, el estado lógico del sensor cambia nuevamente a "0". Con ello se apagan el sistema de enfriamiento y el indicador luminoso.

Si el sensor no indica ningún descenso de la temperatura, el sistema de enfriamiento y el indicador luminoso se apagan a más tardar al transcurrir el tiempo de enfriamiento preseleccionado. En este caso, el proceso de enfriamiento se puede reiniciar mediante el pulsador "RESET". Al pulsar y soltar el pulsador se genera otro flanco ascendente en la entrada IN, con lo que se reinicia el sistema de enfriamiento.

El sistema de enfriamiento y el indicador luminoso se pueden apagar en todo momento con el pulsador "PARO".

Consulte también

TP: Impulso (Página 5617)

---( )---: Asignación (Página 5605)

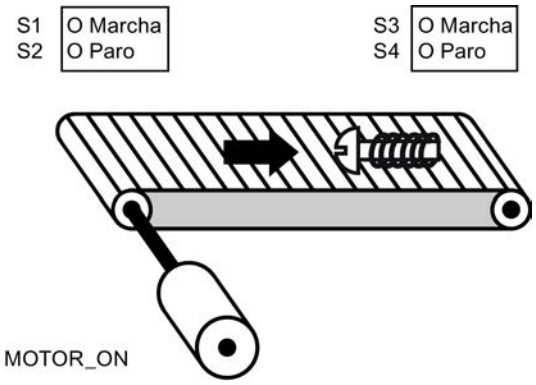
---| |---: Contacto normalmente abierto (Página 5602)

9.7.2.2 Ejemplos de programación FUP

Ejemplo de control de una cinta transportadora

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al principio de la cinta transportadora se encuentran dos pulsadores: S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

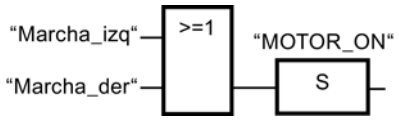
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
Marcha_izquierda (S1)	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Paro_izquierda (S2)	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora
MOTOR_ON	BOOL	Conectar el motor de la cinta transportadora

Los segmentos siguientes muestran la programación FUP para solucionar la tarea:

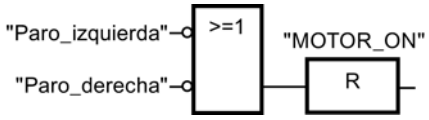
Segmento 1:

El motor de la cinta transportadora se conecta cuando se acciona el pulsador de marcha "S1" o "S3".



Segmento 2:

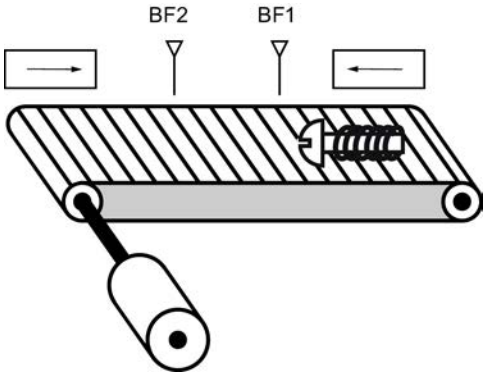
El motor de la cinta transportadora se desconecta cuando se acciona el pulsador de paro "S2" o "S4".



Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

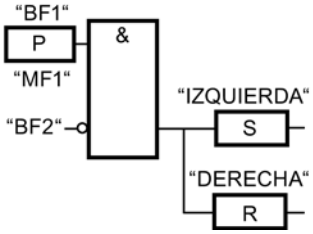
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
DERECHA	BOOL	Indicador de desplazamiento a la derecha
IZQUIERDA	BOOL	Indicador de desplazamiento a la izquierda
MF1	BOOL	Marca de flancos 1
MF2	BOOL	Marca de flancos 2

Los segmentos siguientes muestran la programación FUP para solucionar la tarea:

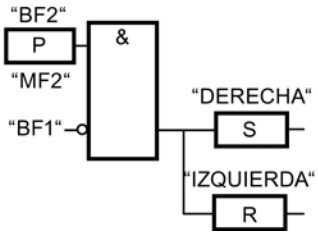
Segmento 1:

Si el estado lógico de la barrera fotoeléctrica "BF1" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF2" es "0", el objeto se desplaza hacia la izquierda sobre la cinta transportadora.



Segmento 2:

Si el estado lógico de la barrera fotoeléctrica "BF2" cambia de "0" a "1" (flanco de señal ascendente) y, simultáneamente, el estado lógico de "BF1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora.



Consulte también

Invertir RLO (Página 5825)

S: Activar salida (Página 5828)

R: Desactivar salida (Página 5827)

&: Operación lógica Y (Página 5819)

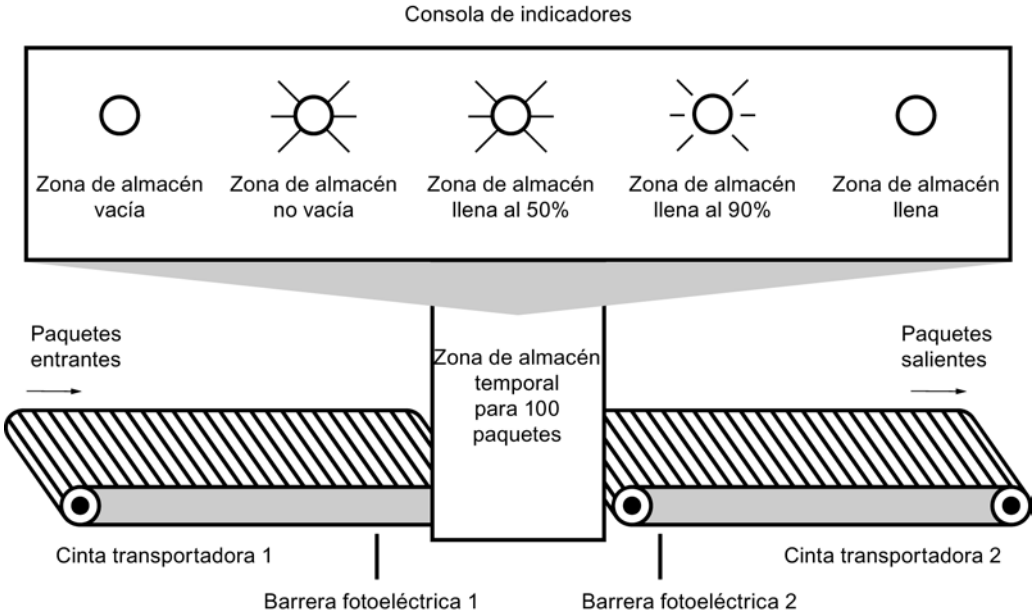
P: Consultar flanco de señal ascendente de un operando (Página 5832)

Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta transportadora 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde esta zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Descripción
BF1	BOOL	Barrera fotoeléctrica 1
BF2	BOOL	Barrera fotoeléctrica 2
RESET	BOOL	Inicializar contador
LOAD	BOOL	Poner el contador al valor del parámetro "ZW"
EXISTENCIAS	INT	Existencias en almacén al rearrancar
PAQUETES	INT	Número de paquetes en la zona de almacén (valor de contaje actual)
PAQUETES_ZA	BOOL	Se activa si el valor de contaje actual es mayor o igual al valor de la variable "EXISTENCIAS".
ALM_VACIO	BOOL	Indicador luminoso: Zona de almacén vacía
ALM_NO_VACIO	BOOL	Indicador luminoso: Zona de almacén no vacía
ALM_LLENO_50%	BOOL	Indicador luminoso: Zona de almacén llena al 50 %
ALM_LLENO_90%	BOOL	Indicador luminoso: Zona de almacén llena al 90 %

Nombre	Tipo de datos	Descripción
ALM_LLENO	BOOL	Indicador luminoso: Zona de almacén llena
VOLUMEN_50	INT	Valor de comparación: 50 paquetes
VOLUMEN_90	INT	Valor de comparación: 90 paquetes
VOLUMEN_100	INT	Valor de comparación: 100 paquetes

Los segmentos siguientes muestran la programación FUP para activar los indicadores luminosos:

Segmento 1:

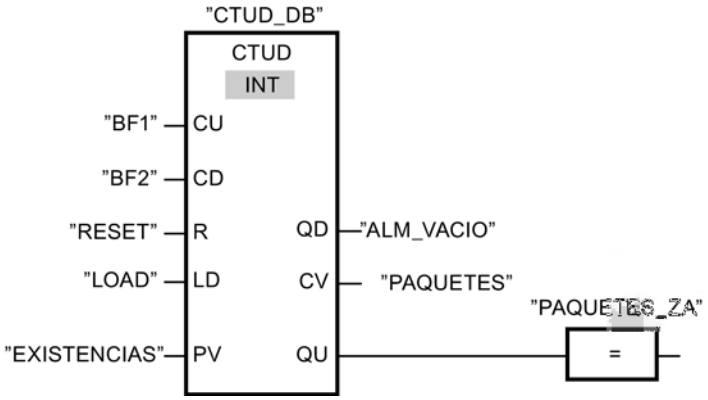
Cuando un paquete se transporta a la zona de almacén, el estado lógico de "BF1" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF1", el contador ascendente se activa y el valor de contaje actual de "PAQUETES" aumenta en uno.

Cuando un paquete se transporta de la zona de almacén a la plataforma de carga, el estado lógico de "BF2" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF2", el contador descendente se activa y el valor actual de "PAQUETES" se reduce en uno.

Si no hay paquetes en la zona de almacén ("PAQUETES" = "0"), la variable "ALM_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén vacía".

El valor de contaje actual puede ponerse a "0" si la variable "RESET" se pone al estado lógico "1".

Cuando la variable "LOAD" se pone al estado lógico "1", el valor de contaje actual adopta el valor de la variable "EXISTENCIAS". Mientras que el valor de contaje actual sea mayor o igual al valor de la variable "EXISTENCIAS", la variable "PAQUETES_ZA" devolverá el estado lógico "1".

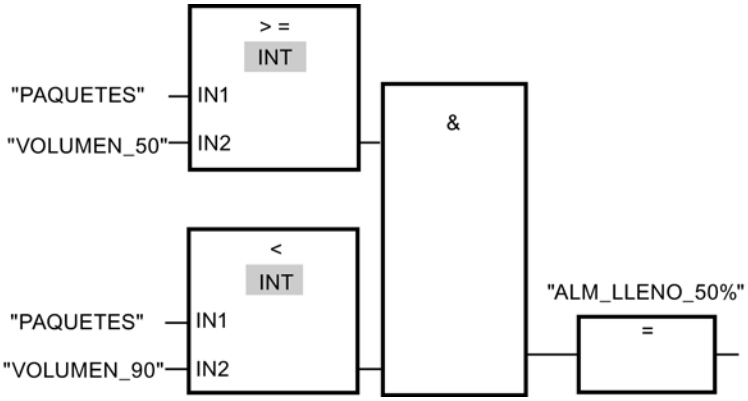


Segmento 2:

Mientras haya paquetes en la zona de almacén, la variable "ALM_NO_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén no vacía".

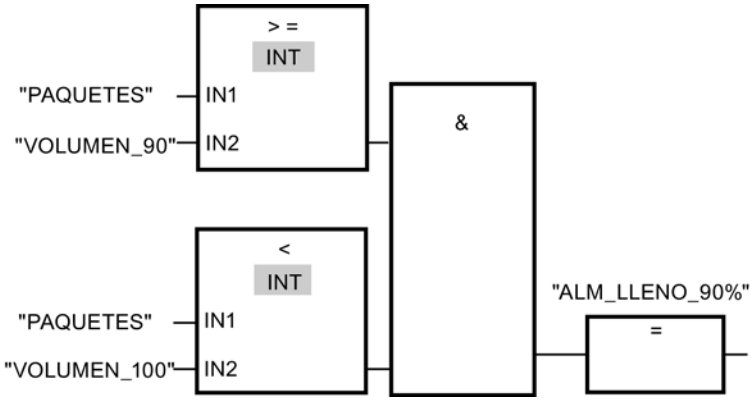
Segmento 3:

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 50, se enciende el indicador luminoso del aviso "Zona de almacén llena al 50 %".



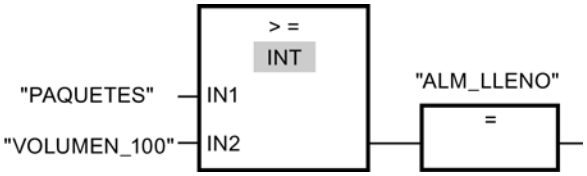
Segmento 4:

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 90, se enciende el indicador luminoso del aviso "Zona de almacén llena al 90 %".



Segmento 5:

Si el número de paquetes que se encuentran en la zona de almacén alcanza 100, se enciende el indicador luminoso del aviso "Zona de almacén llena".



Consulte también

Invertir RLO (Página 5825)

=: Asignación (Página 5826)

&: Operación lógica Y (Página 5819)

CMP >=: Mayor o igual (Página 5890)

CMP <: Menor (Página 5893)

Ejemplo de cálculo de una ecuación

Cálculo de una ecuación

El ejemplo de programación siguiente muestra cómo obtener con tres operaciones aritméticas el mismo resultado que la siguiente ecuación:

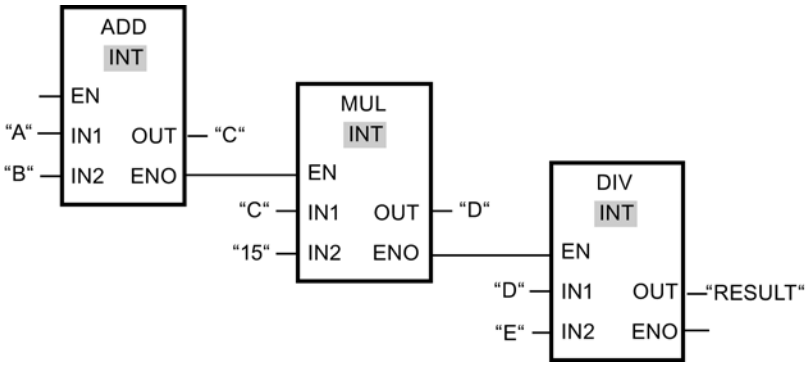
RESULT = ((A + B) x 15) / E

Aplicación

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Comentario
A	INT	Primer sumando
B	INT	Segundo sumando
C	INT	Primer resultado intermedio
15	INT	Multiplicador
D	INT	Segundo resultado intermedio
E	INT	Divisor
RESULT	INT	Resultado final

El segmento siguiente muestra la programación FUP para calcular la ecuación:



El valor del operando "A" se suma al valor del operando "B". La suma se almacena en el operando "C". El valor del operando "C" se multiplica por "15". El resultado de la multiplicación se almacena en el operando "D". A continuación, el valor almacenado en el operando "D" se divide por el valor del operando "E". El resultado final se almacena en el operando "RESULT".

Consulte también

ADD: Sumar (Página 5894)

DIV: Dividir (Página 5899)

Ejemplo de control de la temperatura ambiente

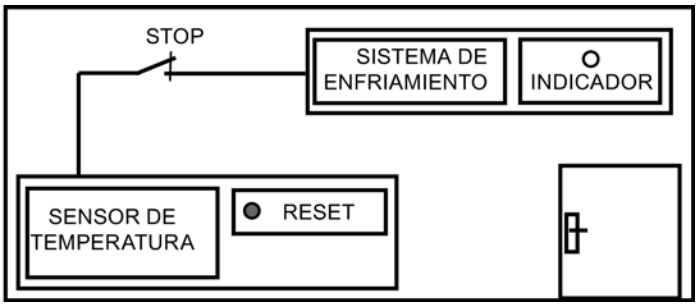
Controlar la temperatura ambiente

La temperatura de una cámara de enfriamiento debe mantenerse por debajo de los cero grados centígrados. Las posibles fluctuaciones de la temperatura se controlan mediante un sensor. Si la temperatura aumenta por encima de los cero grados centígrados, el sistema de enfriamiento se conecta por un tiempo preseleccionado. Durante la fase de enfriamiento se enciende un indicador "Sistema de enfriamiento iniciado".

El sistema de enfriamiento y el indicador luminoso se apagan cuando se cumple una de las condiciones siguientes:

- El sensor indica un descenso de la temperatura por debajo de los cero grados centígrados.
- El tiempo de enfriamiento preseleccionado ha transcurrido.
- Se acciona el pulsador "PARO".

Si ha transcurrido el tiempo de enfriamiento preseleccionado y la temperatura de la cámara de enfriamiento es aún demasiado elevada, el sistema de enfriamiento se puede reiniciar mediante el pulsador "RESET".



Realización

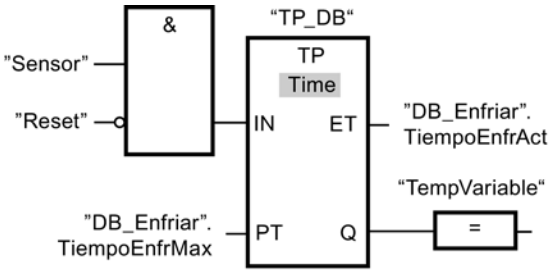
La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Comentario
Sensor	BOOL	Señal del sensor de temperatura
RESET	BOOL	Reset
PARO	BOOL	El sistema de enfriamiento se desconecta.
TiempoEnfrMax	TIME	Tiempo de enfriamiento preseleccionado Esta variable está definida en el bloque de datos "DB_Enfriar".

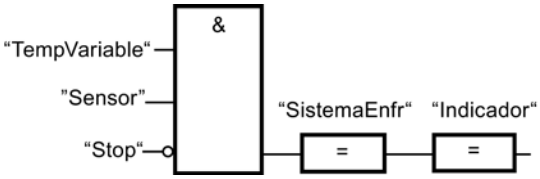
Nombre	Tipo de datos	Comentario
TiempoEnfrAct	TIME	Tiempo de enfriamiento transcurrido actualmente Esta variable está definida en el bloque de datos "DB_Enfriar".
SistemaEnfr	BOOL	El sistema de enfriamiento se conecta.
Indicador	BOOL	El indicador luminoso del aviso "Sistema de enfriamiento iniciado" se enciende.
TempVariable	BOOL	Variable temporal Esta variable guarda el estado lógico del temporizador CEI TP.

El segmento siguiente muestra la programación FUP para controlar la temperatura ambiente:

Segmento 1:



Segmento 2:



Si la temperatura de la cámara de enfriamiento aumenta por encima de los cero grados centígrados, el estado lógico del operando "Sensor" cambia de "0" a "1" (flanco de señal ascendente). Con un flanco de señal ascendente en la entrada IN de la función de temporización se inicia el tiempo de enfriamiento preseleccionado y la "TempVariable" recibe el estado lógico "1". El estado lógico "1" de la "TempVariable" hace que en el segmento 2 se conecte el sistema de enfriamiento y se encienda el indicador luminoso. Las salidas "Sensor", "SistemaEnfr" e "Indicador" tienen que programarse en el segmento 2, ya que en la salida Q de la función de temporización sólo se puede programar una bobina.

Si la temperatura de la cámara de enfriamiento desciende por debajo de los cero grados centígrados, el estado lógico del sensor cambia nuevamente a "0". Con ello se apagan el sistema de enfriamiento y el indicador luminoso.

Si el sensor no indica ningún descenso de la temperatura, el sistema de enfriamiento y el indicador luminoso se apagan a más tardar al transcurrir el tiempo de enfriamiento preseleccionado. En este caso, el proceso de enfriamiento se puede reiniciar mediante el pulsador "RESET". Al pulsar y soltar el pulsador se genera otro flanco ascendente en la entrada IN, con lo que se reinicia el sistema de enfriamiento.

El sistema de enfriamiento y el indicador luminoso se pueden apagar en todo momento con el pulsador "PARO".

Consulte también

&: Operación lógica Y (Página 5819)

=: Asignación (Página 5826)

TP: Impulso (Página 5838)

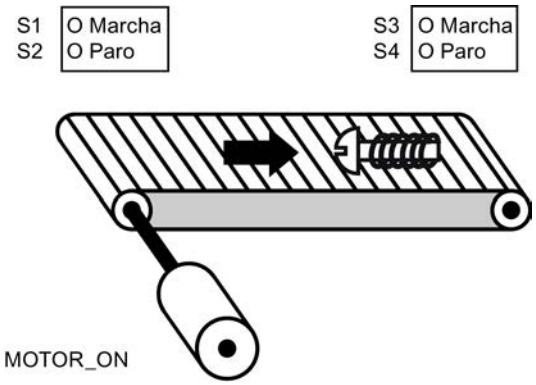
Invertir RLO (Página 5825)

9.7.2.3 Ejemplos de programación AWL

Ejemplo: Operaciones lógicas con bits

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al principio de la cinta transportadora se encuentran dos pulsadores: S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Operando	Declaración	Tipo de datos	Descripción
Marcha_izquierda (S1)	Input	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Paro_izquierda (S2)	Input	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora

Operando	Declaración	Tipo de datos	Descripción
Marcha_derecha (S3)	Input	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	Input	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora
MOTOR_ON/OFF	Output	BOOL	Marcha/paro del motor de la cinta transportadora

El siguiente programa AWL muestra cómo realizar la tarea:

```
AWL                 Explicación
O #S1               // Consultar si el pulsador de marcha S1 es "1"
O #S3               // Consultar si el pulsador de marcha S3 es "1"
S #"MOTOR_ON/OFF"   // Si uno de los pulsadores de marcha (S1 o S3) devuelve el
                    estado lógico "1", se pone en marcha el motor de la cinta
                    transportadora.
O #S2               // Consultar si el pulsador de paro S2 es "1"
O #S4               // Consultar si el pulsador de paro S4 es "1"
R #"MOTOR_ON/OFF"   // Si uno de los pulsadores de paro (S2 o S4) devuelve el estado
                    lógico "1", se para el motor de la cinta transportadora.
```

Consulte también

O: Operación lógica O (Página 6161)

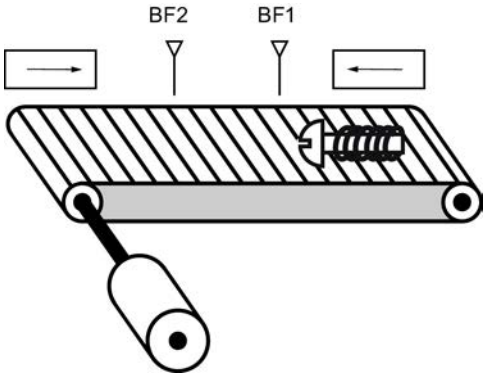
S: Activar (Página 6179)

R: Desactivar (Página 6177)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Declaración	Tipo de datos	Descripción
S1	Input	BOOL	Barrera fotoeléctrica 1
S2	Input	BOOL	Barrera fotoeléctrica 2
TM1	Input	BOOL	Marca de flancos 1
TM2	Input	BOOL	Marca de flancos 2
DERECHA	Output	BOOL	Indicador de desplazamiento a la derecha
IZQUIERDA	Output	BOOL	Indicador de desplazamiento a la izquierda

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
AWL                 Explicación
A #S1               // Consultar si la barrera fotoeléctrica "S1" es "1"
FP #TM1             // Consultar si hay flanco ascendente
AN #S2              // Consultar si la barrera fotoeléctrica "S2" es "0"
S #IZQUIERDA        // Si el estado lógico de la barrera fotoeléctrica "S1" cambia
                    de "0" a "1" (flanco ascendente) y al mismo tiempo el estado
                    lógico de la barrera fotoeléctrica "S2" es "0", el objeto se
                    desplaza hacia la izquierda sobre la cinta transportadora.
                    // Se enciende el indicador de desplazamiento a la izquierda
A #S2               // Consultar si la barrera fotoeléctrica "S2" es "1"
```

AWL	Explicación
FP #TM2	// Consultar si hay flanco ascendente
AN #S1	// Consultar si la barrera fotoeléctrica "S1" es "0"
S #DERECHA	// Si el estado lógico de la barrera fotoeléctrica "S2" cambia de "0" a "1" (flanco ascendente) y al mismo tiempo el estado lógico de "S1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora.
	// Se enciende el indicador de desplazamiento a la derecha
AN #S1	// Consultar si la barrera fotoeléctrica "S1" es "0"
AN #S2	// Consultar si la barrera fotoeléctrica "S2" es "0"
R #IZQUIERDA	// El indicador de desplazamiento hacia la izquierda se apaga cuando el estado lógico de ambas barreras fotoeléctricas es "0".
R #DERECHA	// El indicador de desplazamiento hacia la derecha se apaga cuando el estado lógico de ambas barreras fotoeléctricas es "0".

Consulte también

A: Operación lógica Y (Página 6158)

AN: Operación lógica Y negada (Página 6159)

S: Activar (Página 6179)

R: Desactivar (Página 6177)

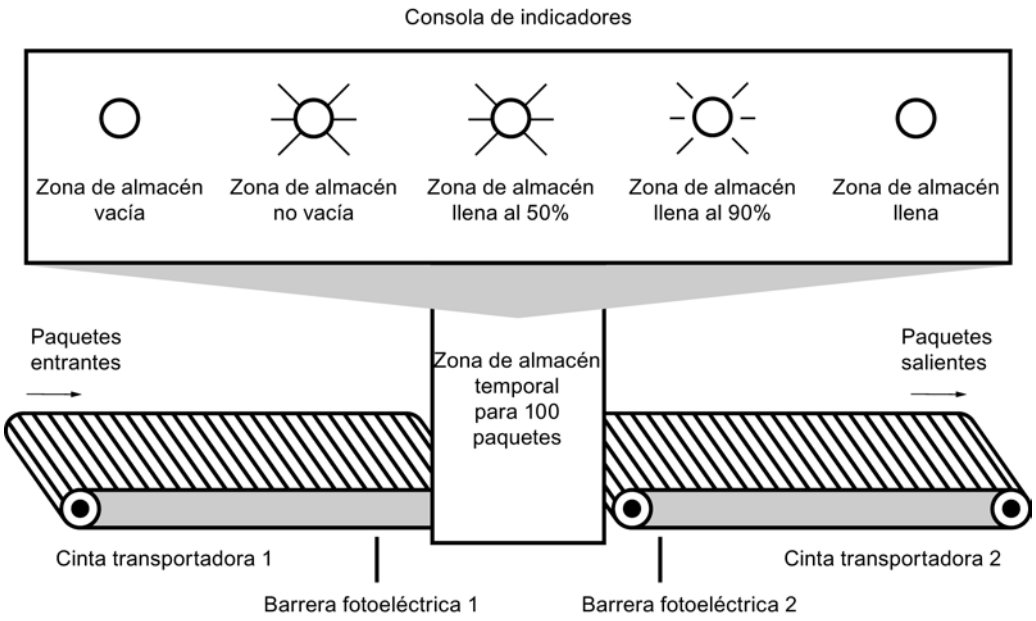
FP: Consultar flanco de señal ascendente del RLO (Página 6185)

Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta transportadora 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde la zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Tipo de datos	Dirección	Descripción
PACKAGECOUNT	COUNTER	C1	Número de paquetes en la zona de almacén (valor de contaje actual)

Nombre	Sección	Tipo de datos	Descripción
LS1	Input	BOOL	Barrera fotoeléctrica 1
LS2	Input	BOOL	Barrera fotoeléctrica 2
ALM_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén vacía
ALM_NO_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén no vacía
ALM_LLENO_50%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 50%
ALM_LLENO_90%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 90%
ALM_LLENO	Output	BOOL	Indicador luminoso: Zona de almacén llena

El siguiente programa AWL muestra cómo realizar el ejemplo:

AWL	Explicación
A #LS1	// Consultar si la barrera fotoeléctrica "LS1" es "1".
CU "PACKAGECOUNT"	// Cuando se produce un flanco ascendente en la barrera fotoeléctrica "LS1", se incrementa en uno el valor de contaje del contador "PACKAGECOUNT".
A #LS2	// Consultar si la barrera fotoeléctrica "LS2" es "1".
CD "PACKAGECOUNT"	// Cuando se produce un flanco ascendente en la barrera fotoeléctrica "LS2", se incrementa en uno el valor de contaje del contador "PACKAGECOUNT".
AN "PACKAGECOUNT"	// Consultar si el valor de contaje es "0".
= #ALM_VACIO	// Cuando el valor de contaje es "0", se enciende el indicador luminoso "Zona de almacén vacía"
A "PACKAGECOUNT"	// Consultar si el valor de contaje es "1".
= #ALM_NO_VACIO	// Cuando el valor de contaje es "0", se enciende el indicador luminoso "Zona de almacén vacía".
L 50	// Cargar el valor de comparación "50" en el acumulador 1.
L "PACKAGECOUNT"	// Desplazar el valor de comparación al acumulador 2.
	// Cargar el valor de contaje actual en el acumulador 1.
<=I	// Comparar valores
= #"ALM_LLENO_50%"	// Si el valor de contaje es mayor o igual a "50" se enciende el indicador luminoso "Zona de almacén llena al 50%"
L 90	// Desplazar el valor de contaje al acumulador 2.
	// Cargar el valor de comparación "90" en el acumulador 1.
>=I	// Comparar valores
= #"ALM_LLENO_90%"	// Si el valor de contaje es mayor o igual a "90" se enciende el indicador luminoso "Zona de almacén llena al 90%".
L "PACKAGECOUNT"	// Cargar el valor de contaje actual en el acumulador 1.
L 100	// Desplazar el valor de contaje al acumulador 2.
	// Cargar el valor de comparación "100" en el acumulador 1.
>=I	// Comparar valores
= #ALM_LLENO	// Si el valor de contaje es mayor o igual a "100" se enciende el indicador luminoso "Zona de almacén llena".

Consulte también

A: Operación lógica Y (Página 6158)

AN: Operación lógica Y negada (Página 6159)

=: Asignación (Página 6176)

CD: Contador descendente (Página 6213)

CU: Contador ascendente (Página 6212)

L: Cargar (Página 6251)

? I: Comparación de enteros (Página 6215)

Ejemplo de cálculo de una ecuación

Cálculo de una ecuación

El ejemplo de programación siguiente muestra cómo obtener con tres operaciones aritméticas el mismo resultado que la siguiente ecuación:

RESULT = ((A + B) x C) /D

Realización

La tabla siguiente muestra la declaración de los operandos utilizados en la tabla de variables PLC:

Nombre	Tipo de datos	Comentario
A	INT	Primer sumando
B	INT	Segundo sumando
C	INT	Multiplicador
D	INT	Divisor
RESULT	INT	Resultado final

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
 AWL                 Explicación
L "A"                // Cargar el valor del operando "A" en el acumulador 1
L "B"                // Desplazar el valor del operando "A" al acumulador 2
                     // Cargar el valor del operando "B" en el acumulador 1
+I                   // Sumar los valores de los acumuladores 1 y 2
                     // Guardar suma en el acumulador 1
L "C"                // Desplazar la suma al acumulador 2
                     // Cargar el valor del operando "C" en el acumulador 1
*I                   // Multiplicar los valores de los acumuladores 1 y 2
                     // Guardar el producto en el acumulador 1
L "D"                // Desplazar el producto al acumulador 2
                     // Cargar el valor del operando "D" en el acumulador 1
/I                   // Dividir el valor del acumulador 2 por el valor del acumulador
                     1
                     // Guardar el resultado en el acumulador 1
T "RESULT"           // Transferir el resultado al operando "RESULT"
```

Consulte también

L: Cargar (Página 6251)

T: Transferir (Página 6260)

+I: Sumar enteros (16 bits) (Página 6220)

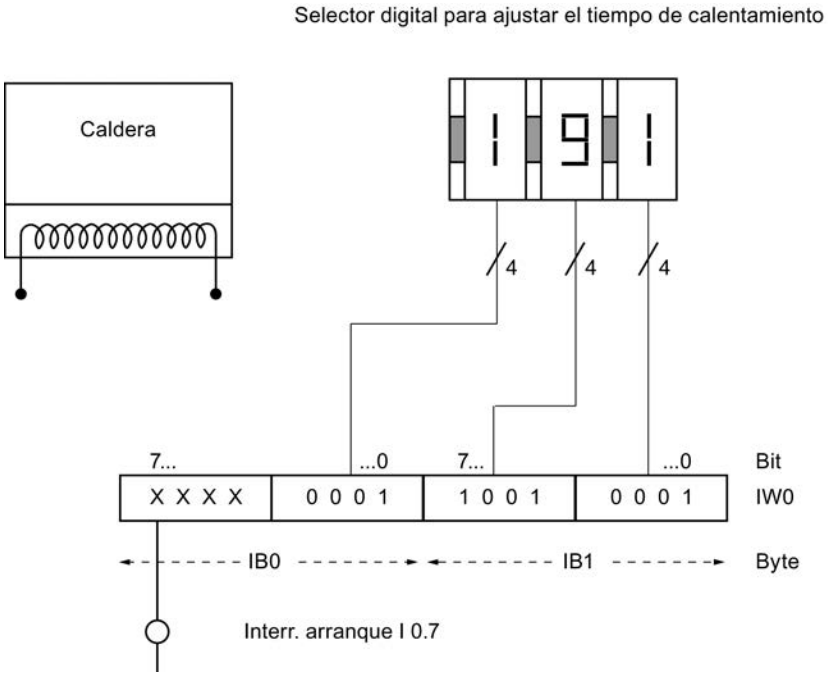
*I: Multiplicar enteros (16 bits) (Página 6222)

/I: Dividir enteros (16 bits) (Página 6223)

Ejemplo: Operaciones lógicas con palabras

Calentamiento de una caldera

La imagen siguiente muestra una caldera que se calienta con un interruptor de arranque. Al pulsar el interruptor de marcha, se inicia el proceso de calentamiento. El tiempo de calentamiento se establece mediante selectores digitales. El tiempo se indica en segundos en formato BCD.



Realización

La tabla siguiente muestra la declaración de los operandos utilizados en la tabla de variables PLC:

Nombre	Tipo de datos	Dirección	Comentario
DURATION	WORD	EW0	Tiempo de calentamiento en segundos • I1.0 a I1.3: Selector de unidades • I1.4 a I1.7: Selector de decenas • I0.0 a I0.3: Selector de centenas
HEATING	TIMER	T1	Temporizador que arranca con el tiempo de calentamiento preseleccionado.

La tabla siguiente muestra la declaración de los operandos utilizados en la interfaz del bloque lógico:

Nombre	Sección	Tipo de datos	Comentario
START	Input	BOOL	Interruptor de arranque
START_HEATING	Output	BOOL	Inicio del proceso de calentamiento

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
AWL                 Explicación
A "HEATING"         // Consultar si el temporizador ha arrancado
= #START_HEATING    // Iniciar proceso de calentamiento
BEC                 // Si RLO=1 finalizar la ejecución del bloque
                    // De este modo se impide que rearranque el temporizador
                    "HEATING" cuando se pulse el pulsador de arranque
L "DURATION"        // Cargar el tiempo de calentamiento en el acumulador 1
AW W#16#0FFF        // Poner los bits de entrada I0.4 a I0.7 a "0"
OW W#16#4000        // Ajustar segundos en los bits I1.2 e I1.3 del acumulador 1
A #START            // Consultar si el pulsador de arranque es "1"
SE "HEATING"        // Cuando se produzca un flanco ascendente en el interruptor de
                    arranque, iniciar el proceso de calentamiento como impulso
                    prolongado
```

Consulte también

L: Cargar (Página 6251)

SE: Arrancar temporizador como impulso prolongado (Página 6196)

A: Operación lógica Y (Página 6158)

=: Asignación (Página 6176)

BEC: Fin de bloque condicional (Página 6326)

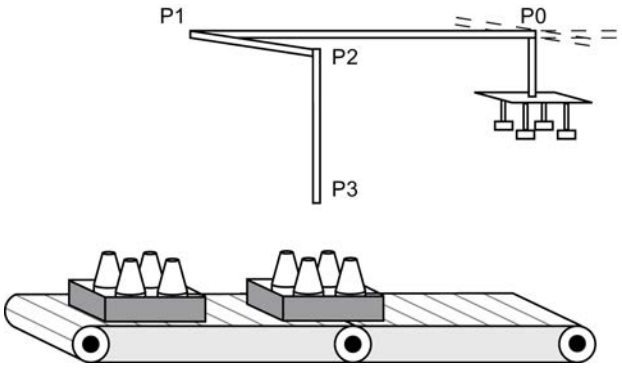
AW: Operación lógica Y por palabras (Página 6333)

OW: Operación lógica O por palabras (Página 6335)

Ejemplo de una cadena secuencial

Programación de una cadena secuencial

La imagen siguiente muestra una estación para despaletizar envases de vidrio. Los palets se transportan sobre una cinta transportadora a la estación. Cuando un palet llega con envases de vidrio a la estación, la cinta transportadora se detiene y un brazo robótico se desplaza desde su posición inicial (P0) hasta alcanzar la posición sobre el palet (P2). Si el brazo robótico se encuentra sobre el palet, se abren las garras y el brazo desciende. La posición actual del brazo y el estado de las garras son registrados por sensores. El desarrollo descrito del movimiento del brazo robótico se realiza en este ejemplo mediante una cadena secuencial. También pueden programarse los pasos sucesivos para retirar los envases y transportarlos a otra cinta transportadora.



Realización

La tabla siguiente muestra la declaración de los operandos utilizados en la tabla de variables PLC:

Nombre	Tipo de datos	Comentario
NUMBER	INT	Número de etapa
Tag_Error	BOOL	Operando que se activa si el número de la etapa es mayor que 3 o si una etapa no ha sido ejecutada.

La tabla siguiente muestra la declaración de los operandos utilizados en la interfaz del bloque lógico:

Nombre	Sección	Tipo de datos	Comentario
POS_0	Input	BOOL	Brazo en posición inicial (P0)
POS_1	Input	BOOL	Brazo en posición 1 (P1)
POS_2	Input	BOOL	Brazo en posición 2 (P2)
GRIPPER_OPEN	Input	BOOL	Garras abiertas
OUT_POS_1	Output	BOOL	Desplazar el brazo a la posición 1
OUT_POS_2	Output	BOOL	Desplazar el brazo a la posición 2
OUT_GRIPPER	Output	BOOL	Abrir las garras
OUT_POS_3	Output	BOOL	Desplazar el brazo a la posición 3

El siguiente programa AWL muestra cómo realizar el ejemplo:

```
AWL                          Explicación
L "NUMBER"                   // Cargar el número de etapa en el acumulador 1.
JL END                       // Principio de la lista de saltos
JU POSITION_0                // Si el acumulador 1 contiene el valor "0", saltar a
                             la etiqueta "POSITION_0".
JU POSITION_1                // Si el acumulador 1 contiene el valor "1", saltar a
                             la etiqueta "POSITION_1".
JU POSITION_2                // Si el acumulador 1 contiene el valor "2", saltar a
                             la etiqueta "POSITION_2".
JU POSITION_3                // Si el acumulador 1 contiene el valor "3", saltar a
                             la etiqueta "POSITION_3".
END: JU ERROR                // Fin de la lista de saltos
                             // Si el número de etapa es mayor que 3, saltar a la
                             etiqueta "ERROR".
POSITION_0: A #POS_0         // Etiqueta "POSITION_0"
                             // Consultar si el brazo se encuentra en la posición
                             inicial (P0).
= #OUT_POS_1                 // Si se cumple la condición, activar la salida
                             "OUT_POS_1" y desplazar el brazo a la posición 1 (P1).
JCN ERROR                    // Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT                      // Si el RLO es "1", saltar a la etiqueta "NEXT".
POSITION_1: A #POS_1         // Etiqueta "POSITION_1"
                             // Consultar si el brazo se encuentra en la posición
                             1 (P1).
= #OUT_POS_2                 // Si se cumple la condición, activar la salida
                             "OUT_POS_2" y desplazar el brazo a la posición 2 (P2).
JCN ERROR                    // Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT                      // Si el RLO es "1", saltar a la etiqueta "NEXT".
POSITION_2: A #POS_2         // Etiqueta "POSITION_2"
                             // Consultar si el brazo se encuentra en la posición
                             2 (P2).
```

AWL	Explicación
= #OUT_GRIPPER	// Si se cumple la condición, activar la salida "OUT_GRIPPER" y abrir las garras.
JCN ERROR	// Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT	// Si el RLO es "1", saltar a la etiqueta "NEXT".
POSITION_3: A #POS_2	// Etiqueta "POSITION_3"
	// Consultar si el brazo se encuentra en la posición 2 (P2).
A #GRIPPER_OPEN	// Consultar si las garras están abiertas
= #OUT_POS_3	// Si se cumple la condición, activar la salida "OUT_POS_3" y desplazar el brazo a la posición 3 (P3).
JCN ERROR	// Si el RLO es "0", saltar a la etiqueta "ERROR".
JC NEXT	// Si el RLO es "1", saltar a la etiqueta "NEXT".
NEXT: INC 1	// Etiqueta "NEXT"
	// Incrementar el número de etapa en el acumulador 1.
T "NUMBER"	// Transferir el número de etapa al operando "NUMBER".
L 3	// Desplazar el número de etapa al acumulador 2.
	// Cargar el valor 3 en el acumulador 1.
>I	// Consultar si el número de la etapa actual es mayor que 3.
JC RESET_NUMBER	// Si el resultado de la consulta es "1", saltar a la etiqueta "RESET_NUMBER" y continuar allí el procesamiento del programa.
BEU	// Finalizar bloque
RESET_NUMBER: L 0	// Etiqueta "RESET_NUMBER"
	// Cargar el valor "0" en el acumulador 1.
T "NUMBER"	// Asignar el valor "0" al operando "NUMBER" (número de etapa).
BEU	// Finalizar bloque
ERROR: NOT	// Etiqueta "ERROR"
= "Tag_Error"	// Asignar el RLO negado al operando "Tag_Error".
BEU	// Finalizar bloque

Consulte también

JC: Saltar si RLO = 1 (Página 6292)

JCN: Saltar si RLO = 0 (Página 6293)

JU: Salto incondicional (Página 6291)

JL: Definir lista de saltos (Página 6316)

BEU: Fin de bloque incondicional (Página 6328)

L: Cargar (Página 6251)

T: Transferir (Página 6260)

? I: Comparación de enteros (Página 6215)

A: Operación lógica Y (Página 6158)

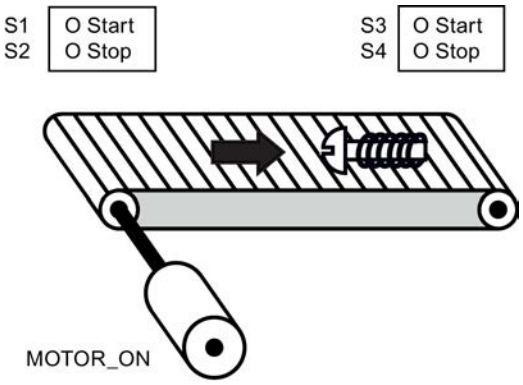
=: Asignación (Página 6176)

9.7.2.4 Ejemplos de programación SCL

Ejemplo: Operaciones lógicas con bits

Control de una cinta transportadora

La figura siguiente muestra una cinta transportadora que se pone en marcha eléctricamente. Al comienzo de la cinta transportadora se encuentran dos pulsadores, S1 para MARCHA y S2 para PARO. Al final de la cinta transportadora se encuentran otros dos pulsadores: S3 para MARCHA y S4 para PARO. La cinta transportadora puede ponerse en marcha o pararse desde cualquiera de ambos extremos.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Operando	Declaración	Tipo de datos	Descripción
Marcha_izquierda (S1)	Input	BOOL	Pulsador de marcha en el extremo izquierdo de la cinta transportadora
Marcha_izquierda (S2)	Input	BOOL	Pulsador de paro en el extremo izquierdo de la cinta transportadora
Marcha_derecha (S3)	Input	BOOL	Pulsador de marcha en el extremo derecho de la cinta transportadora
Paro_derecha (S4)	Input	BOOL	Pulsador de paro en el extremo derecho de la cinta transportadora

Operando	Declaración	Tipo de datos	Descripción
MOTOR_ON	Output	BOOL	Conectar el motor de la cinta transportadora
MOTOR_OFF	Output	BOOL	Desconectar el motor de la cinta transportadora

El siguiente programa SCL muestra cómo realizar la tarea:

SCL

```
IF "Marcha_izquierda" OR "Marcha_derecha" THEN
"MOTOR_ON" := 1;
END_IF;
IF "Paro_izquierda" OR "Paro_derecha" THEN
"MOTOR_OFF" := 1;
END_IF;
```

El motor de la cinta transportadora se enciende cuando se acciona el pulsador de marcha "Marcha_izquierda" o "Marcha_derecha". El motor de la cinta transportadora se apaga cuando se acciona el pulsador de paro "Paro_izquierda" o "Paro_derecha".

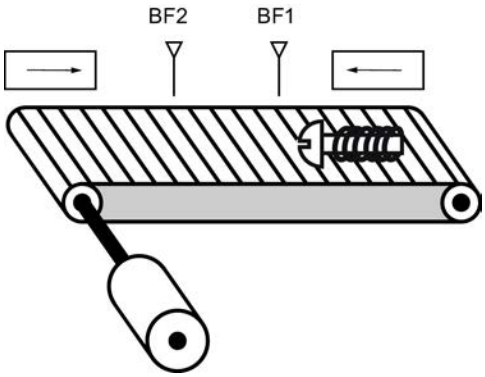
Consulte también

Expresiones lógicas (Página 2792)

Ejemplo de detección del sentido de marcha de una cinta transportadora

Detección del sentido de marcha de una cinta transportadora

La figura muestra una cinta transportadora equipada con dos barreras fotoeléctricas (BF1 y BF2). Las barreras fotoeléctricas sirven para determinar en qué sentido se desplaza un objeto sobre la cinta transportadora.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Declaración	Tipo de datos	Descripción
Barrera fotoeléctrica LS1	Input	BOOL	Barrera fotoeléctrica 1
Barrera fotoeléctrica LS2	Input	BOOL	Barrera fotoeléctrica 2
DERECHA	Output	BOOL	Indicador de desplazamiento a la derecha
IZQUIERDA	Output	BOOL	Indicador de desplazamiento a la izquierda

El siguiente programa SCL muestra cómo realizar el ejemplo:

SCL

```
IF "BF1" = 1 AND NOT "BF2" = 0
THEN "IZQUIERDA" = 1;
IF "BF2" = 1 AND NOT "BF1" = 0
THEN "DERECHA" := 1;
IF "BF2" = 0
THEN "DERECHA" := 0;
IF "BF1" = 0
THEN "IZQUIERDA" := 0;
```

Si el estado lógico de la barrera fotoeléctrica "BF1"es "1" y, simultáneamente, el estado lógico de la barrera fotoeléctrica "BF2" es "0", el objeto se desplaza hacia la izquierda sobre la cinta transportadora. Si el estado lógico de la barrera fotoeléctrica "BF2"es "1" y, simultáneamente, el estado lógico de la barrera fotoeléctrica "BF1" es "0", el objeto se desplaza hacia la derecha sobre la cinta transportadora. Los indicadores de desplazamiento hacia la izquierda se apagan cuando el estado lógico de ambas barreras fotoeléctricas es "0".

Consulte también

Expresiones lógicas (Página 2792)

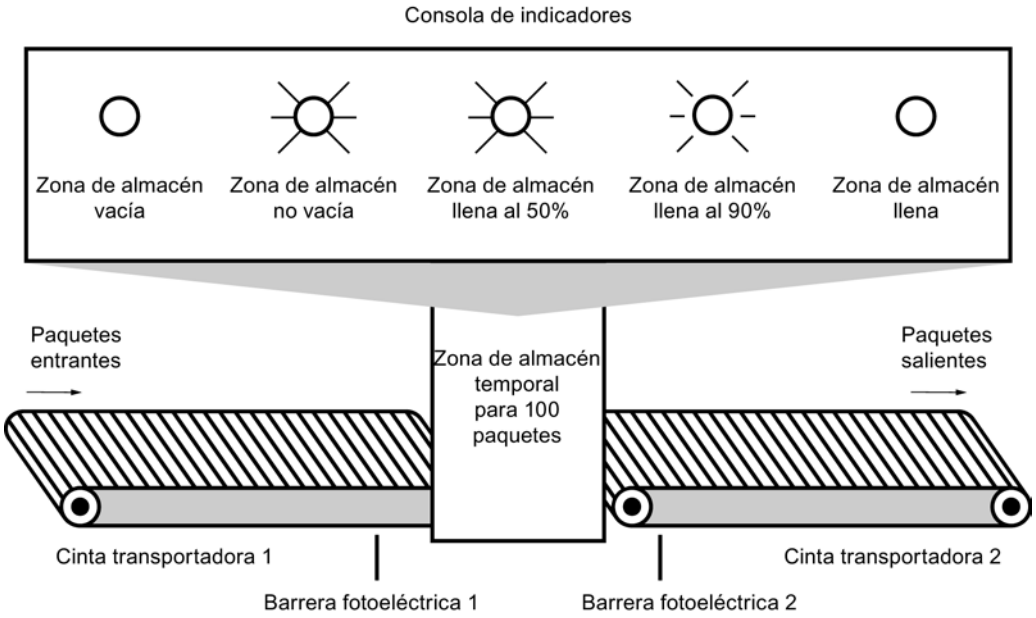
Ejemplo de registro del nivel de llenado de una zona de almacén

Detección del nivel de llenado de una zona de almacén

La figura siguiente muestra un sistema con dos cintas transportadoras y una zona de almacén temporal situada entre ambas. La cinta transportadora 1 transporta paquetes a la zona de almacén. Una barrera fotoeléctrica situada al final de la cinta transportadora 1 junto a la zona de almacén determina cuántos paquetes se transportan a dicha zona. La cinta transportadora 2 transporta paquetes desde esta zona de almacén temporal a una plataforma de carga, en la que los paquetes se cargan en camiones para ser suministrados a los clientes. Una barrera fotoeléctrica situada al final de la zona de almacén determina cuántos paquetes deben

transportarse desde allí hasta la plataforma de carga. Cinco indicadores luminosos indican el nivel de llenado de la zona de almacén temporal.

Al rearrancar las cintas transportadoras, el valor de contaje actual se pone al número de paquetes existentes en la zona de almacén.



Realización

La tabla siguiente muestra la definición de las variables utilizadas:

Nombre	Declaración	Tipo de datos	Descripción
BF1	Input	BOOL	Barrera fotoeléctrica 1
BF2	Input	BOOL	Barrera fotoeléctrica 2
RESET	Input	BOOL	Inicializar contador
LOAD	Input	BOOL	Poner el contador al valor del parámetro "ZW"
EXISTENCIAS	Input	INT	Existencias en almacén al rearrancar
PAQUETES	Output	INT	Número de paquetes en la zona de almacén (valor de contaje actual)
PAQUETES_ZA	Output	BOOL	Se activa si el valor de contaje actual es mayor o igual al valor de la variable "EXISTENCIAS".
ALM_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén vacía

Nombre	Declaración	Tipo de datos	Descripción
ALM_NO_VACIO	Output	BOOL	Indicador luminoso: Zona de almacén no vacía
ALM_LLENO_50%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 50 %
ALM_LLENO_90%	Output	BOOL	Indicador luminoso: Zona de almacén llena al 90 %
ALM_LLENO	Output	BOOL	Indicador luminoso: Zona de almacén llena
VOLUMEN_50	Input	INT	Valor de comparación: 50 paquetes
VOLUMEN_90	Input	INT	Valor de comparación: 90 paquetes
VOLUMEN_100	Input	INT	Valor de comparación: 100 paquetes

El siguiente programa SCL muestra cómo realizar el ejemplo:

Cuando un paquete se transporta a la zona de almacén, el estado lógico de "BF1" cambia de "0" a "1" (flanco de señal ascendente). Cuando se produce un flanco ascendente en "BF1", el contador ascendente se activa y el valor de contaje actual de "PAQUETES" aumenta en uno.

Cuando un paquete se transporta de la zona de almacén a la plataforma de carga, el estado lógico de "BF2" cambia de "0" a "1" (flanco ascendente). Cuando se produce un flanco ascendente en "BF2", el contador descendente se activa y el valor actual de "PAQUETES" se reduce en uno.

Si no hay paquetes en la zona de almacén ("PAQUETES" = "0"), la variable "ALM_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén vacía".

El valor de contaje actual puede ponerse a "0" si la variable "RESET" se pone al estado lógico "1".

Cuando la variable "LOAD" se pone al estado lógico "1", el valor de contaje actual adopta el valor de la variable "EXISTENCIAS". Mientras que el valor de contaje actual sea mayor o igual al valor de la variable "EXISTENCIAS", la variable "PAQUETES_ZA" devolverá el estado lógico "1".

SCL

```
"CTUD_DB".CTUD(CU := "BF1",
               CD := "BF2",
               R  := "RESET",
               LD := "LOAD",
               PV := "EXISTENCIAS",
               QU := "PAQUETES_ZA",
               QD := "ALM_VACIO",
               CV := "PAQUETES");
```

Mientras haya paquetes en la zona de almacén, la variable "ALM_NO_VACIO" se pone al estado lógico "1" y se enciende el indicador luminoso "Zona de almacén no vacía".

SCL

```
"ALM_NO_VACIO" := NOT "ALM_VACIO"
```

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 50, se enciende el indicador luminoso del aviso "Zona de almacén llena al 50 %".

SCL

```
IF "PAQUETES" >= "VOLUMEN_50" THEN "ALM_LLENO_50%" := 1;
IF "PAQUETES" <= "VOLUMEN_90" THEN "ALM_LLENO_50%" := 1;
```

Si el número de paquetes que se encuentran en la zona de almacén es mayor o igual a 90, se enciende el indicador luminoso del aviso "Zona de almacén llena al 90 %".

SCL

```
IF "PAQUETES" >= "VOLUMEN_90" THEN "ALM_LLENO_90%" := 1;
IF "PAQUETES" < "VOLUMEN_100" THEN "ALM_LLENO_90%" := 1;
```

Si el número de paquetes que se encuentran en la zona de almacén alcanza 100, se enciende el indicador luminoso del aviso "Zona de almacén llena".

SCL

```
IF "PAQUETES" >= "VOLUMEN_100" THEN "ALM_LLENO" := 1;
```

9.8 Referencias

9.8.1 Referencias (S7-1200, S7-1500)

9.8.1.1 Parámetros generales de las instrucciones

Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente

Instrucciones que funcionan asíncronamente

En las instrucciones que funcionan asíncronamente la ejecución de la función se extiende a lo largo de varias llamadas.

Identificación de la petición

Si mediante una instrucción que funciona asíncronamente se dispara una alarma de proceso o se lanza la salida de comandos de control a los esclavos DP o bien una transmisión de datos o la cancelación de una conexión no configurada y se vuelve a llamar dicha instrucción antes de que finalice la petición en curso, el comportamiento posterior de la instrucción dependerá en gran medida de si en la siguiente llamada se trata de la misma petición o no.

Parámetro REQ

El parámetro de entrada REQ (request) sirve exclusivamente para lanzar la petición:

- Si se llama la instrucción de una petición que no está activada en este momento, la petición se lanza con REQ = 1 (caso 1).
- Si hay una petición determinada en curso y se vuelve a llamar la instrucción para la misma petición antes de que finalice (p. ej. en un OB de alarma cíclica), REQ no es evaluado por la instrucción (caso 2).

Parámetro RET_VAL y BUSY

A través de los parámetros de salida RET_VAL y BUSY se indica el estado de ejecución de la petición.

Tenga en cuenta la indicación contenida en la sección: Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

- En el caso 1 (primera llamada con REQ=1) siempre que haya recursos de sistema libres y los parámetros de entrada reciban valores correctos en RET_VALW#16#7001, y se activará BUSY.
  Si los recursos de sistema necesarios están ocupados en estos momentos o si hay un error en los parámetros de entrada, se registrará el código de error correspondiente en RET_VAL y se escribe un 0 en BUSY.
- En el caso 2 (llamada intermedia) se registra W#16#7002 en RET_VAL (esto equivale a una advertencia: petición todavía en curso), y se activará BUSY.
- En la última llamada de una petición vale lo siguiente:
  - En la instrucción "DPNRM_DG (Página 4806)", si la transferencia de datos es correcta se registra en RET_VAL el número de datos transferidos en bytes como número positivo. En BUSY se escribe un "0" en este caso.
    En caso de error se registra información de error en RET_VAL. BUSY no se debe evaluar en este caso.
  - En las demás instrucciones y si la petición se ejecuta sin errores, se registra un "0" en RET_VAL y en BUSY se escribe un "0". En caso de error se registra el código de error en RET_VAL y en BUSY se escribe un "0".

Nota

Si la primera y la última llamada coinciden en el tiempo, para RET_VAL y BUSY rige lo descrito para la última llamada.

Resumen

La tabla siguiente muestra una visión esquemática de lo arriba descrito. En particular, indica los valores posibles de los parámetros de salida, si la petición no ha finalizado todavía después de una llamada de la instrucción.

Nota

En el programa es necesario evaluar los parámetros de salida relevantes depués de cada llamada.

Relación entre llamada, REQ, RET_VAL y BUSY cuando hay una petición en curso

N.º de la llamada	Tipo de llamada	REQ	RET_VAL	BUSY
1	Primera llamada	1	W#16#7001	1
			Código de error	0
2 a (n - 1)	Llamada intermedia	Irrelevante	W#16#7002	1
n	Última llamada	Irrelevante	W#16#0000, si no se han producido errores.	0
			Código de error, si se han producido errores.	0

Evaluación de errores con el parámetro de salida RET_VAL

Tipos de información de error

Una instrucción procesada le indica en el programa de usaurio si la CPU ha podido o no ejecutar la función de la instrucción.

Recibirá información de error al respecto por dos vías:

- vía el bit BR de la palabra de estado
- vía el parámetro de salida RET_VAL (return value).

Nota

Antes de evaluar los parámetros de salida de la instrucción siempre deberá procederse del siguiente modo:

- Evalúe en primer lugar el bit BR de la palabra de estado.
- Compruebe después el parámetro de salida RET_VAL.

Si el bit BR indica que la instrucción no ha sido procesada correctamente o si en RET_VAL figura un código de error general, no deberán evaluarse los parámetros de salida de la instrucción.

Información de error en el valor de retorno

Una instrucción indica con el valor "0" del bit de resultado binario (BR) de la palabra de estado que se ha producido un error al procesar la función. Algunas instrucciones ponen a disposición

un código de error adicional en un parámetro de salida llamado valor de retorno (RET_VAL). Si en el parámetro de salida RET_VAL aparece un error general (v. explicación más abajo), ello se indica con el valor "0" del bit BR de la palabra de estado.

El valor de retorno es del tipo de datos entero (INT). La relación del valor de retorno con respecto al valor "0" indica si se ha producido un error al procesar la función.

Procesamiento de la instrucción por la CPU	BR	Valor de retorno	Signo del entero
erróneo	0	menor que "0"	negativo (bit de signo es "1")
correcto	1	mayor o igual a "0"	positivo (bit de signo es "0")

Reacción a la información de error

En los códigos de error de RET_VAL se distingue entre:

- un código de error general que pueden devolver todas las instrucciones
- un código de error específico que puede devolver una instrucción según sus funciones específicas.

El programa se puede escribir de manera que reaccione a posibles errores en el procesamiento de la instrucción. De este modo pueden evitarse errores subsiguientes.

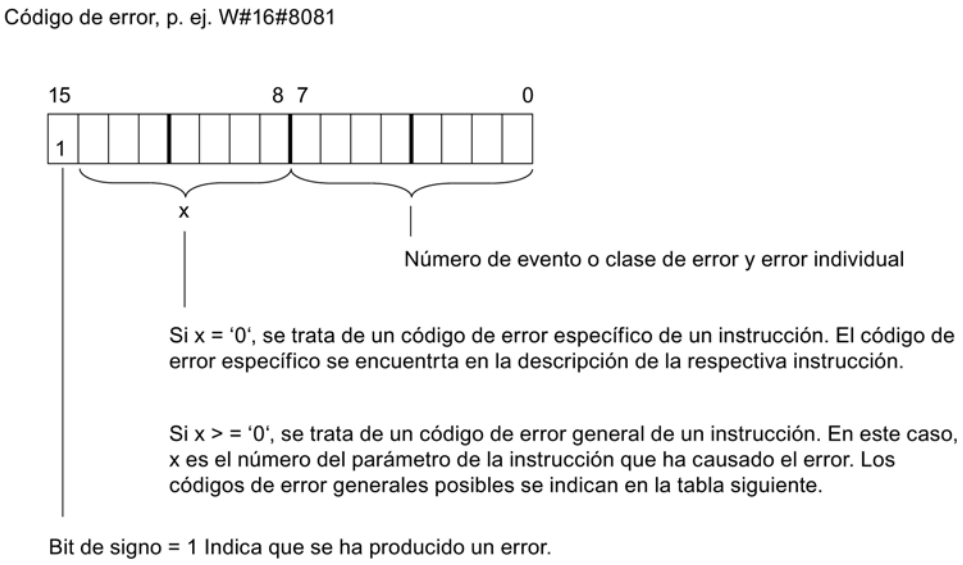
Información de error general y específica

El valor de retorno (RET_VAL) de una instrucción pone a disposición uno de los siguientes dos códigos de error:

- Un código de error general que puede referirse a cualquier instrucción.
- Un código de error específico que sólo puede referirse a la instrucción en cuestión.

Si bien el tipo de datos del parámetro de salida RET_VAL es un entero (INT), los códigos de error de la instrucción tienen valores hexadecimales. Si evalúa un valor de retorno y lo compara con los códigos de error indicados en esta documentación, visualice el código de error en formato hexadecimal.

La figura siguiente ilustra la estructura de un código de error de una función de sistema en formato hexadecimal.

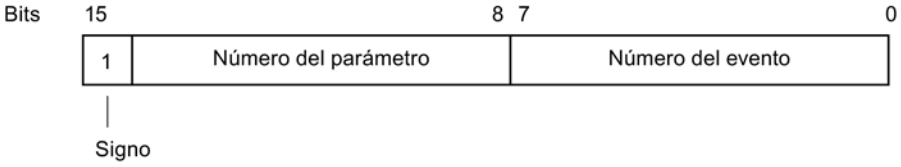


Información de error general

El código de error general indica errores que pueden aparecer en todas las instrucciones. Un código de error general está formado por los siguientes dos números:

- Un número de parámetro comprendido entre 1 y 111, donde 1 muestra el primer parámetro, 2 el segundo parámetro etc. de la instrucción llamada.
- Un número de evento comprendido entre 0 y 127. El número de evento indica un error síncrono.

A continuación se indican los códigos de errores generales así como su significado.



Nota

En caso de que se haya registrado un código de error general en RET_VAL, es posible que

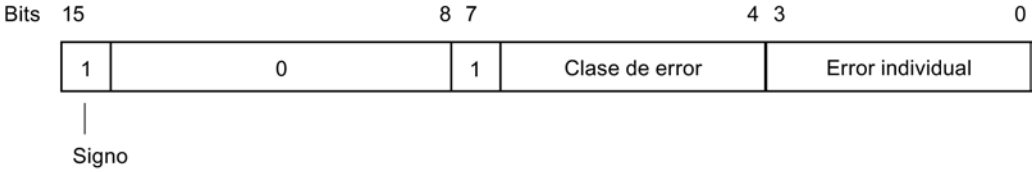
- se haya lanzado la acción correspondiente a la instrucción o que ya haya sido ejecutada por completo.
- se haya producido un error específico de la instrucción al ejecutar la acción. Debido a un error general que haya tenido lugar posteriormente, ya no ha sido posible indicar el error específico.

Información de error específica

Algunas instrucciones poseen un valor de retorno que pone a disposición un código de error específico de la instrucción. El código de error específico indica errores que sólo pueden aparecer en una única instrucción.

Un código de error específico está formado por los siguientes dos números:

- Una clase de error comprendida entre 0 y 7.
- Un error individual comprendido entre 0 y 15.



Códigos de error generales

La tabla siguiente explica los códigos de error generales de un valor de retorno. El código de error se indica en formato hexadecimal. La letra x en cada número de error sirve únicamente de comodín y representa el número del parámetro de la función de sistema que ha causado el error.

Códigos de error generales

Código de error (W#16#....)	Explicación
8x7F	Error interno Este código de error indica un error interno en el parámetro x.
8x01	Identificador de sintaxis no válido en un parámetro VARIANT
8x22	Error de longitud de área al leer un parámetro.
8x23	Error de longitud de área al escribir un parámetro. Este código de error indica que el parámetro x se encuentra íntegra o parcialmente fuera del área de operandos o que la longitud de un campo de bits de un parámetro VARIANT no es divisible por 8.
8x24	Error de área al leer un parámetro.
8x25	Error de área al escribir un parámetro. Este código de error indica que el parámetro x se encuentra en un área no válida para la función de sistema. La descripción de la función en cuestión indica las áreas que no son válidas para la función.
8x26	El parámetro contiene un número demasiado elevado de una célula de temporizador. Este código de error indica que la célula de temporizador indicada en el parámetro x no existe.
8x27	El parámetro contiene un número demasiado elevado de una célula de contador (error de número del contador). Este código de error indica que la célula de contador indicada en el parámetro x no existe.
8x28	Error de alineación al leer un parámetro.

Código de error (W#16#....)	Explicación
8x29	Error de alineación al escribir un parámetro. Este código de error indica que la referencia al parámtero x es un operando cuya dirección de bit es diferente de 0.
8x30	El parámetro se encuentra en el DB global protegido contra escritura.
8x31	El parámetro se encuentra en el DB de instancia protegido contra escritura. Este código de error indica que el parámetro x se encuentra en un bloque de datos protegido contra escritura. Si el bloque de datos ha sido abierto por la propia función de sistema, la función de sistema devolverá siempre el valor W#16#8x30.
8x32	El parámetro contiene un número de DB demasiado elevado (error de número del DB).
8x34	El parámetro contiene un número de FC demasiado elevado (error de número del FC).
8x35	El parámetro contiene un número de FB demasiado elevado (error de número del FB). Este código de error indica que el parámetro x contiene un número de bloque mayor que el número de bloque máximo admisible.
8x3A	El parámetro contiene el número de un DB que no está cargado.
8x3C	El parámetro contiene el número de un FC que no está cargado.
8x3E	El parámetro contiene el número de un FB que no está cargado.
8x42	Se ha producido un error de accdeso mientras el sistema intentaba leer un parámetro del área de periferia de las entradas.
8x43	Se ha producido un error de accdeso mientras el sistema intentaba escribir un parámetro en el área de periferia de las salidas.
8x44	Error en el enésimo (n > 1) acceso en lectura tras producirse un error.
8x45	Error en el enésimo (n > 1) acceso en escritura tras producirse un error. Este código de error indica que se deniega el acceso al parámetro deseado.

9.8.1.2 Instrucciones básicas

KOP

Operaciones lógicas con bits

---| |---: Contacto normalmente abierto

Descripción

La activación de un contacto normalmente abierto depende del estado lógico del operando correspondiente. Si el estado lógico del operando es "1", se cierra el contacto normalmente abierto y el estado lógico de la entrada se transfiere a la salida.

Si el estado lógico del operando es "0", el contacto normalmente abierto no se activa y el estado lógico de la salida de la instrucción se pone a "0".

En una conexión en serie, dos o más contactos normalmente abiertos se combinan de bit en bit mediante una operación lógica Y. La corriente fluye por una conexión en serie cuando todos los contactos están cerrados.

En una conexión en paralelo, los contactos normalmente abiertos se combinan mediante una operación lógica O. La corriente fluye por una conexión en paralelo si uno de los contactos está cerrado.

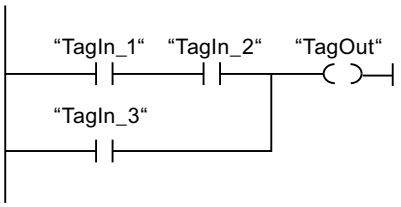
Parámetros

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" tienen el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---| / |---: Contacto normalmente cerrado

Descripción

La activación de un contacto normalmente cerrado depende del estado lógico del operando correspondiente. Si el estado lógico del operando es "1", se abre el contacto normalmente cerrado y el estado lógico de la salida de la instrucción se pone a "0".

Si el estado lógico del operando es "0", no se activa el contacto normalmente cerrado y el estado lógico de la entrada se transfiere a la salida.

En una conexión en serie, dos o más contactos normalmente cerrados se combinan de bit en bit mediante una operación lógica Y. La corriente fluye por una conexión en serie cuando todos los contactos están cerrados.

En una conexión en paralelo, los contactos normalmente cerrados se combinan mediante una operación lógica O. La corriente fluye por una conexión en paralelo si uno de los contactos está cerrado.

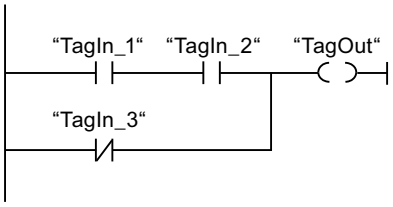
Parámetros

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" tienen el estado lógico "1".
- El operando "TagIn_3" tiene el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

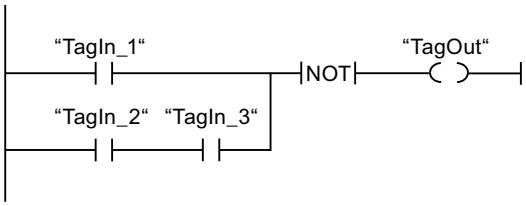
--|NOT|--: Invertir RLO

Descripción

La instrucción "Invertir RLO" invierte el estado lógico del resultado lógico (RLO). Si el estado lógico de la entrada de la instrucción es "1", la salida de la instrucción devuelve el estado lógico "0". Si el estado lógico de la entrada de la instrucción es "0", la salida devuelve el estado lógico "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El estado lógico de los operandos "TagIn_2" y "TagIn_3" es "1".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---( )---: Asignación

Descripción

La instrucción "Asignación" permite activar el bit de un operando indicado. Si el resultado lógico (RLO) en la entrada de la bobina es "1", el operando indicado adopta el estado lógico "1". Si el estado lógico de la entrada de la bobina es "0", el bit del operando indicado se pone a "0".

La instrucción no afecta al RLO. El RLO de la entrada de la bobina se transfiere directamente a la salida.

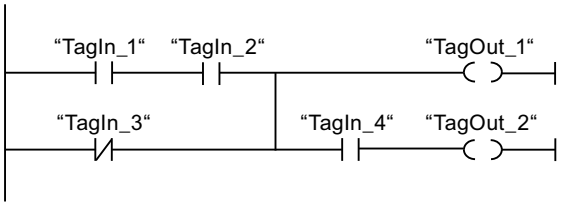
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut_1" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

El operando "TagOut_2" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1", "TagIn_2" y "TagIn_4" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0" y el del operando "TagIn_4" es "1".

Consulte también

Insertar elementos KOP (Página 2678)

Cablear operandos en instrucciones KOP (Página 2701)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

--(/)--: Negar asignación

Descripción

La instrucción "Negar asignación" invierte el resultado lógico (RLO) y lo asigna al operando indicado. Si el RLO de la entrada de la bobina es "1", se desactiva el operando. Si el RLO de la entrada de la bobina es "0", el operando adopta el estado lógico "1".

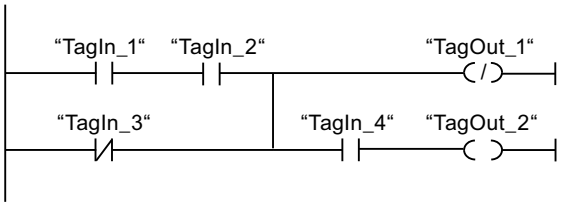
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Negar asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO negado.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut_1" se desactiva cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

Consulte también

Insertar elementos KOP (Página 2678)

Cablear operandos en instrucciones KOP (Página 2701)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

---(R)---: Desactivar salida

Descripción

La instrucción "Desactivar salida" permite poner a "0" el estado lógico de un operando indicado.

La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si fluye corriente hacia la bobina (RLO = "1"), el operando indicado se pone a "0". Si el RLO de la entrada de la bobina es "0" (no hay flujo de señales en la bobina), el estado lógico del operando indicado no cambia.

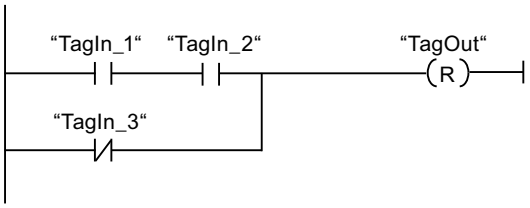
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria		
			S7-1200	S7-1500	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Operando que se desactiva si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(S)---: Activar salida

Descripción

Mediante la instrucción "Activar salida" se pone a "1" el estado lógico de un operando indicado.

La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si fluye corriente hacia la bobina (RLO = "1"), el operando indicado se pone a "1". Si el RLO de la entrada de la bobina es "0" (no hay flujo de señales en la bobina), el estado lógico del operando indicado no cambia.

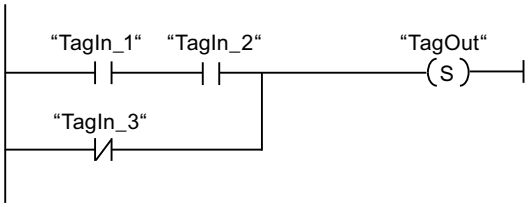
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se activa si RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

SET_BF: Activar mapa de bits

Descripción

La instrucción "Activar mapa de bits" activa varios bits a partir de una dirección específica.

El número de bits que deben activarse se determina mediante el valor del <Operando1>. El <Operando2> determina la dirección del primer bit que debe activarse. Si el valor del <Operando1> es mayor que el número de bits de un byte seleccionado, se activan los bits del byte siguiente. Los bits permanecen activados hasta que son desactivados explícitamente por otra instrucción.

La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si en la entrada de la bobina hay un RLO de "0", la instrucción no se ejecuta.

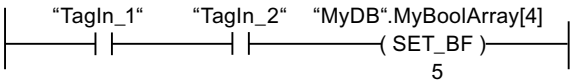
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar mapa de bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando2>	Output	BOOL	I, Q, M En un DB o un IDB, un elemento de ARRAY[..] of BOOL	Puntero al primer bit que debe activarse.
<Operando1>	Input	UINT	Constante	Número de bits que deben activarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se activan 5 bits a partir de la dirección del operando "MyDB".MyBoolArray[4].

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

RESET_BF: Desactivar mapa de bits

Descripción

La instrucción "Desactivar mapa de bits" desactiva varios bits a partir de una dirección específica.

El número de bits que se deben desactivar se determina mediante el valor del <Operando1>. El <Operando2> determina la dirección del primer bit que se debe desactivar. Si el valor del <Operando1> es mayor que el número de bits de un byte seleccionado, se desactivan los bits del byte siguiente. Los bits permanecen desactivados hasta que son activados explícitamente por otra instrucción.

La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si en la entrada de la bobina hay un RLO de "0", la instrucción no se ejecuta.

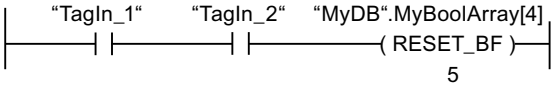
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar mapa de bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando2>	Output	BOOL	I, Q, M En un DB o un IDB, un elemento de ARRAY[..] of BOOL	Puntero al primer bit que debe desactivarse.
<Operando1>	Input	UINT	Constante	Número de bits que se deben desactivar.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se desactivan 5 bits a partir de la dirección del operando "MyDB".MyBoolArray[4].

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

SR: Flipflop de activación/desactivación

Descripción

La instrucción "Flipflop de activación/desactivación" sirve para activar o desactivar el bit de un operando indicado en función del estado lógico de las entradas S y R1. Si el estado lógico de la entrada S es "1" y el de la entrada R1 es "0", el operando indicado se pone a "1". Si el estado lógico de la entrada S es "0" y el de la entrada R1 es "1", el operando indicado se pone a "0".

La entrada R1 prevalece sobre la entrada S. Si el estado lógico de las entradas S y R1 es "1", el estado lógico del operando indicado se pone a "0".

Si el estado lógico de ambas entradas S y R1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

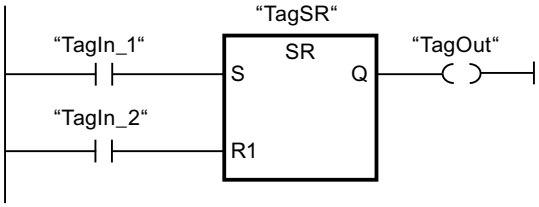
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de activación/ desactivación":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
S	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L	Activación habilitada
R1	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Desactivación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Operando que se activa o desactiva.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagSR" y "TagOut" se activan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagSR" y "TagOut" se desactivan cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

RS: Flipflop de desactivación/activación

Descripción

La instrucción "Flipflop de desactivación/activación" sirve para desactivar o activar el bit de un operando indicado en función del estado lógico de las entradas R y S1. Si el estado lógico de la entrada R es "1" y el de la entrada S1 es "0", el operando indicado se pone a "0". Si el estado lógico de la entrada R es "0" y el de la entrada S1 es "1", el operando indicado se pone a "1".

La entrada S1 prevalece sobre la entrada R. Si el estado lógico de las entradas R y S1 es "1", el estado lógico del operando indicado se pone a "1".

Si el estado lógico de ambas entradas R y S1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

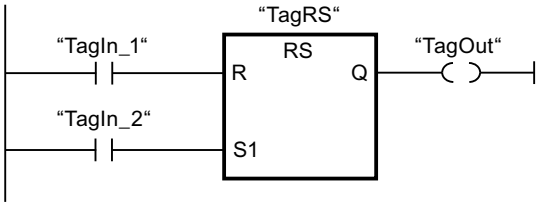
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de desactivación/activación":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
R	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L	Desactivación habilitada
S1	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Activación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Operando que se desactiva o activa.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagRS" y "TagOut" se desactivan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagRS" y "TagOut" se activan cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

--|P|--: Consultar flanco de señal ascendente de un operando

Descripción

La instrucción "Consultar flanco de señal ascendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "0" a "1". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del resultado lógico (RLO) de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

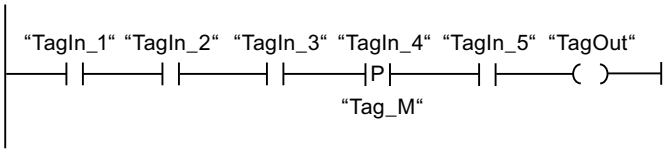
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando1>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1", "TagIn_2" y "TagIn_3" devuelven el estado lógico "1".
- En el operando "TagIn_4" hay un flanco ascendente. El estado lógico de la consulta anterior se guarda en la marca de flancos "Tag_M".
- El estado lógico del operando "TagIn_5" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

--|N|--: Consultar flanco de señal descendente de un operando

Descripción

La instrucción "Consultar flanco de señal descendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "1" a "0". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos <Operando2>. Si la

instrucción detecta un cambio del resultado lógico (RLO) de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente situado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

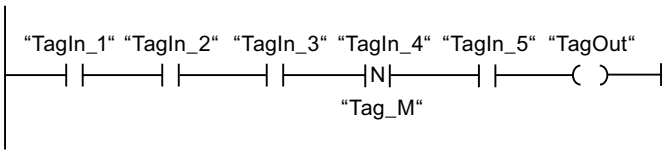
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando 1>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Señal que se debe consultar
<Operando 2>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1", "TagIn_2" y "TagIn_3" devuelven el estado lógico "1".
- En el operando "TagIn_4" hay un flanco de señal descendente. El estado lógico de la consulta anterior se guarda en la marca de flancos "Tag_M".
- El estado lógico del operando "TagIn_5" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

--(P)--: Activar operando con flanco de señal ascendente

Descripción

La instrucción "Activar operando con flanco de señal ascendente" permite activar un operando indicado (<Operando1>) cuando se detecta un cambio del resultado lógico (RLO) de "0" a "1". La instrucción compara el RLO actual con el RLO de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, el <Operando1> adopta el estado lógico "1" durante un ciclo del programa. En los demás casos, el estado lógico del operando es "0".

Indique el operando que debe activarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

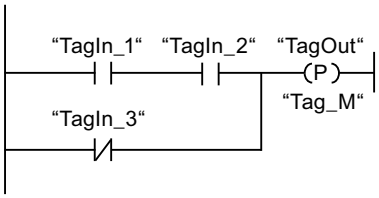
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar operando con flanco de señal ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Output	BOOL	I, Q, M, D, L	Operando que se activa si hay un flanco ascendente.
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa durante un ciclo del programa cuando el estado lógico de la entrada de la bobina cambia de "0" a "1" (flanco de señal ascendente). En todos los demás casos, el operando "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

--(N)--: Activar operando con flanco de señal descendente

Descripción

La instrucción "Activar operando con flanco de señal descendente" permite activar un operando indicado (<Operando1>) cuando se presenta un cambio del resultado lógico (RLO) de "1" a "0". La instrucción compara el RLO actual con el RLO de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, el <Operando1> adopta el estado lógico "1" durante un ciclo del programa. En los demás casos, el estado lógico del operando es "0".

Indique el operando que debe activarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

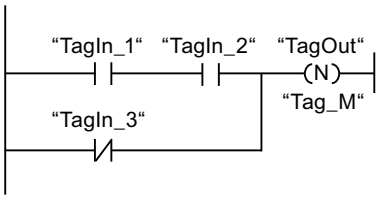
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar operando con flanco de señal descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Output	BOOL	I, Q, M, D, L	Operando que se activa si hay un flanco descendente.
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa durante un ciclo del programa cuando el estado lógico de la entrada de la bobina cambia de "1" a "0" (flanco de señal descendente). En todos los demás casos, el operando "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

P_TRIG: Consultar flanco de señal ascendente del RLO

Descripción

La instrucción "Consultar flanco de señal ascendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "0" a "1". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

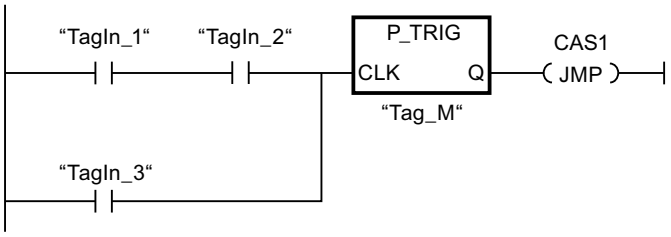
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "0" a "1", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

N_TRIG: Consultar flanco de señal descendente del RLO

Descripción

La instrucción "Consultar flanco de señal descendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "1" a "0". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

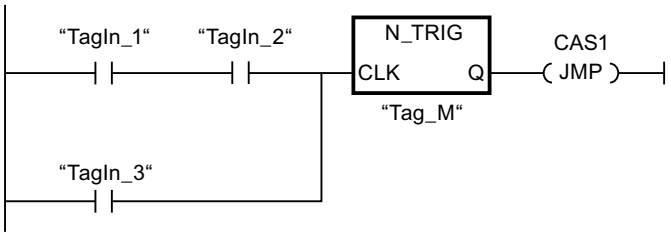
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "1" a "0", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

R_TRIG: Activar variable con flanco de señal ascendente

Descripción

La instrucción "Activar variable con flanco de señal ascendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "0" a "1". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

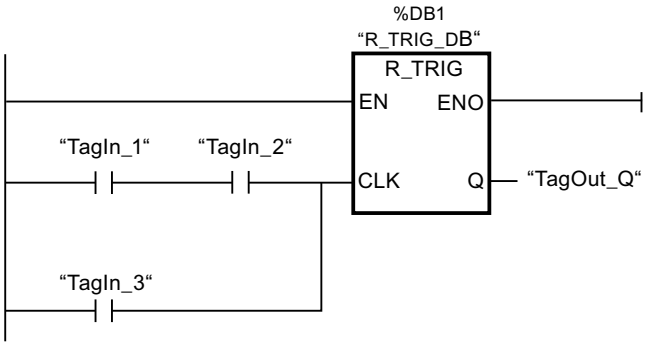
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
CLK	Input	BOOL	I, Q, M, D, L o constante	Señal entrante cuyo flanco se consulta.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la variable del DB de instancia "R_TRIG_DB" se almacena el RLO de la consulta anterior. Si en los operandos "TagIn_1" y "TagIn_2" o en el operando "TagIn_3" se detecta un cambio del estado lógico del RLO de "0" a "1", la salida "TagOut_Q" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

F_TRIG: Activar variable con flanco de señal descendente

Descripción

La instrucción "Activar variable con flanco de señal descendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "1" a "0". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

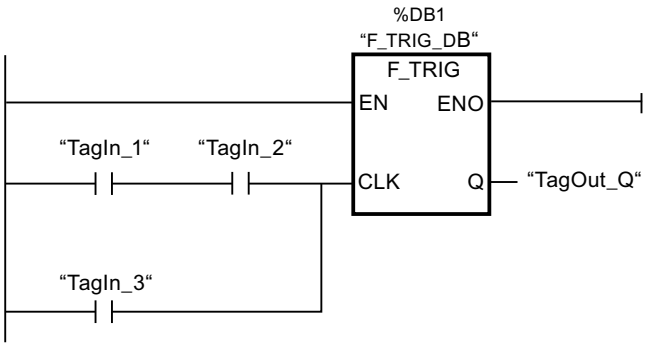
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
CLK	Input	BOOL	I, Q, M, D, L o constante	Señal entrante cuyo flanco se consulta.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la variable del DB de instancia "F_TRIG_DB" se almacena el RLO de la consulta anterior. Si en los operandos "TagIn_1" y "TagIn_2" o en el operando "TagIn_3" se detecta un cambio del estado lógico del RLO de "1" a "0", la salida "TagOut_Q" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Temporizadores

Temporizadores CEI

TP: Impulso

Descripción

La instrucción "Impulso" activa la salida Q por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. La salida Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico de la salida Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo PT y si el estado lógico de la entrada IN es "0", se desactiva la salida ET.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TP_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME, TP_LTIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME, TP_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Impulso" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

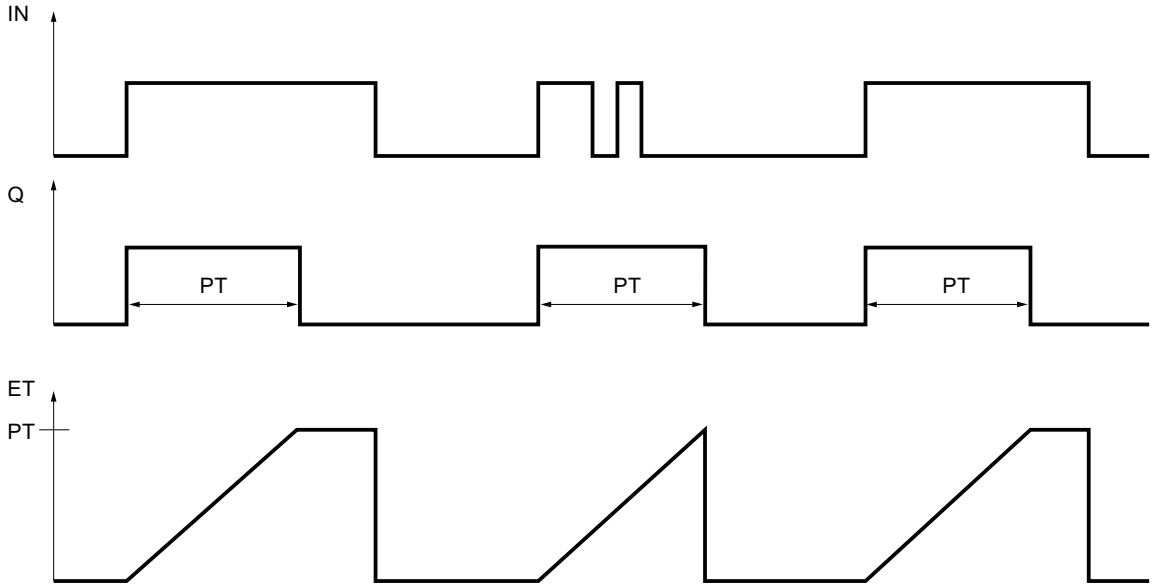
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Entrada de arranque
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Duración del impulso El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida de impulso
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

TON: Retardo al conectar

Descripción

La instrucción "Retardo al conectar" retarda la activación de la salida Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. Una vez transcurrido el tiempo PT, la salida Q devuelve el estado lógico "1". La salida Q permanece activada mientras la entrada de arranque esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva la salida Q. La función de temporización se reinicia al detectarse un nuevo flanco de señal ascendente en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. La salida ET se desactiva en cuanto el estado lógico de la entrada IN cambia a "0".

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TON_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME, TON_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Retardo al conectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

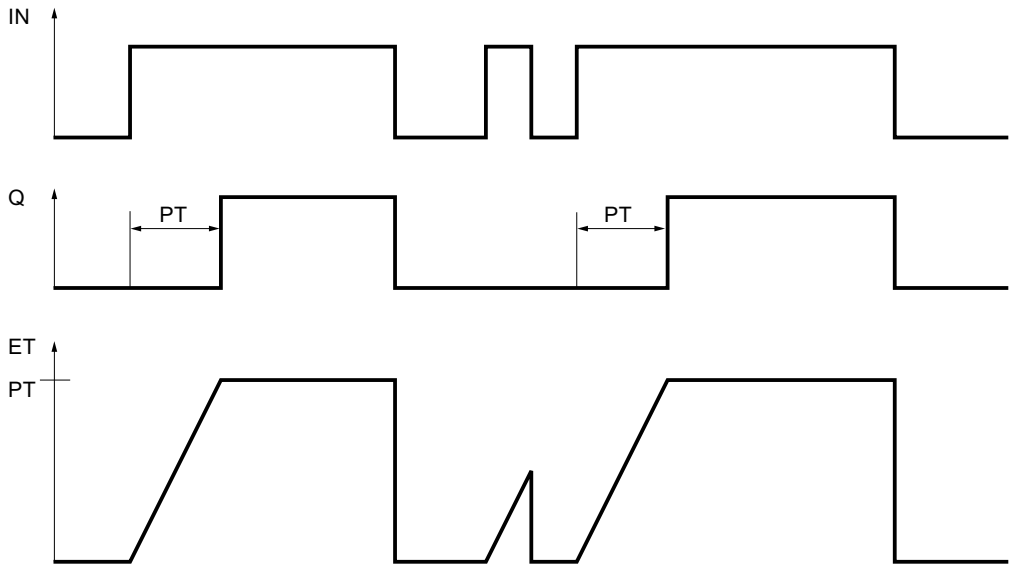
La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Entrada de arranque
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

TOF: Retardo al desconectar

Descripción

La instrucción "Retardo al desconectar" retarda la desactivación de la salida Q por el tiempo programado PT. La salida Q se activa cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico de la entrada IN cambia nuevamente a "0", el tiempo programado PT deja de contar. La salida Q permanece activada mientras transcurre el tiempo PT. Una vez transcurrido el tiempo PT se desactiva la salida Q. Si el estado lógico de la entrada IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico de la salida Q permanece en "1".

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el valor actual de la salida ET se conserva hasta que la entrada IN cambie nuevamente a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TOF_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME, TOF_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Retardo al desconectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

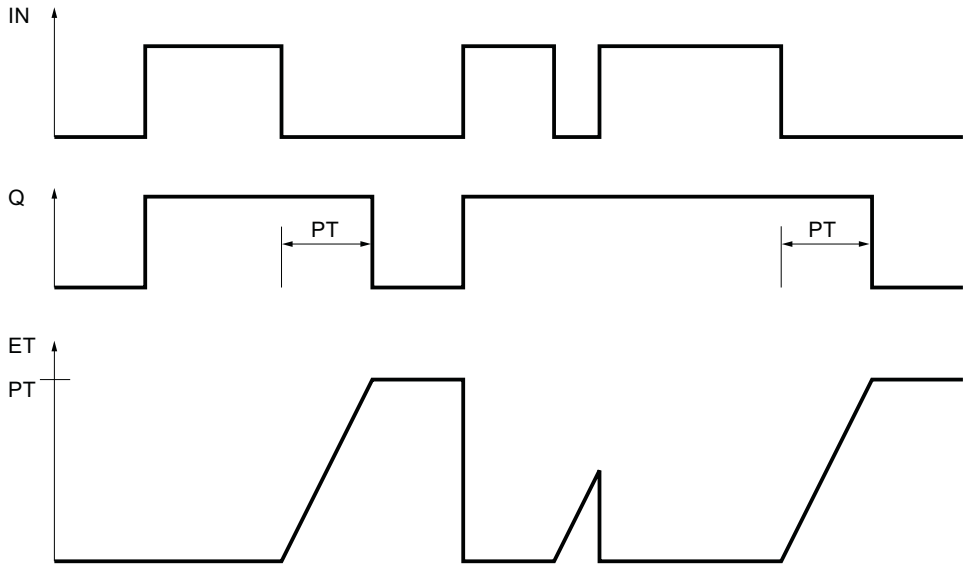
La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Entrada de arranque
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

TONR: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PT. Cuando el estado lógico de la entrada IN cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y se empieza a contar el tiempo PT. Mientras transcurre el tiempo PT se van acumulando los valores de tiempo que se leen cuando el estado lógico de la entrada IN es "1". El tiempo acumulado se deposita en la salida ET y se puede consultar allí. Una vez se ha alcanzado el tiempo PT, la salida Q devuelve el estado lógico "1". El parámetro Q permanece a "1" aunque el estado lógico del parámetro IN cambie de "1" a "0" (flanco de señal descendente).

La entrada R desactiva las salidas ET y Q independientemente del estado lógico de la entrada de arranque.

A cada llamada de la instrucción "Acumulador de tiempo" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TONR_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME, TONR_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Acumulador de tiempo" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

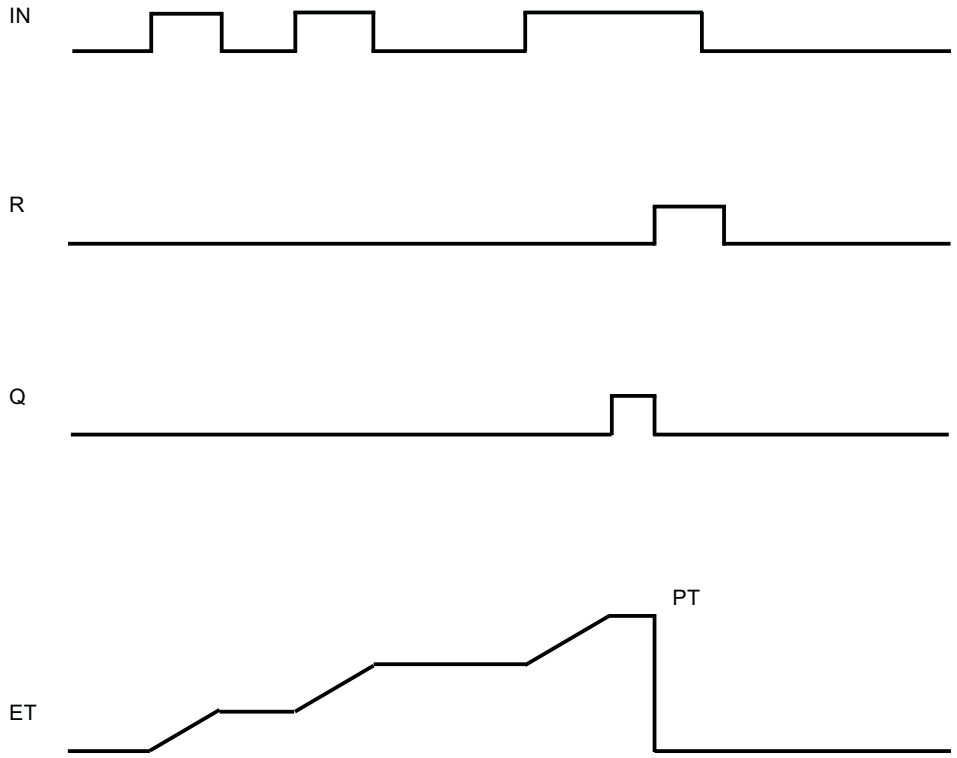
La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Entrada de arranque
R	Input	BOOL	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Entrada de reset
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Tiempo acumulado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Acumulador de tiempo":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

---(TP)---: Arrancar temporizador como impulso

Descripción

La instrucción "Arrancar temporizador como impulso" permite arrancar un temporizador CEI como impulso por un tiempo determinado. El temporizador CEI se arranca cuando el resultado lógico (RLO) cambia de "0" a "1" (flanco de señal ascendente). El temporizador CEI se ejecuta durante el tiempo indicado, independientemente de cómo evolucione el RLO. La detección de un nuevo flanco de señal ascendente no influye en el transcurso del temporizador CEI. Mientras el temporizador CEI cuenta el tiempo, una consulta de si el estado del temporizador

es "1" da como resultado "1". Una vez transcurrido el temporizador CEI, el estado del temporizador devuelve el estado lógico "0".

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles de ejecución, pues cada consulta de las salidas Q o ET actualiza la estructura del IEC_TIMER.

Para CPU S7-1200

La instrucción "Arrancar temporizador como impulso" deposita los datos en una estructura del tipo de datos IEC_TIMER o TP_TIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_LTIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Arrancar temporizador como impulso" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME, TP_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura Q del temporizador CEI. El estado del temporizador se consulta con un contacto normalmente abierto para saber si es "1" o de un contacto normalmente cerrado para saber si es "0". La consulta de Q o ET (p. ej. "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Para poder ejecutar la instrucción "Arrancar temporizador como impulso" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetros

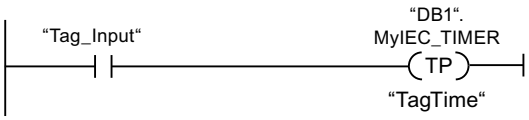
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Tiempo>	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TP_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME	D, L	Temporizador CEI que se arranca.

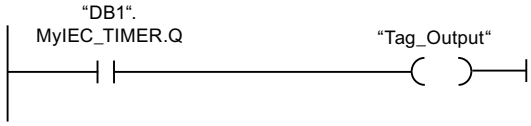
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Arrancar temporizador como impulso" se ejecuta cuando el estado lógico del operando "Tag_Input" cambia de "0" a "1". El temporizador "DB1".MyIEC_TIMER se arranca por el tiempo que está guardado en el operando "TagTime".



Mientras el temporizador "DB1".MyIEC_TIMER cuenta el tiempo, el estado del temporizador ("DB1".MyIEC_TIMER.Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa. Cuando el temporizador CEI termina de contar, el estado lógico del temporizador cambia a "0" y el operando "Tag_Output" se desactiva.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

---(TON)---: Arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión" permite arrancar un temporizador CEI como retardo a la conexión por un tiempo determinado. El temporizador CEI se arranca cuando el resultado lógico (RLO) cambia de "0" a "1" (flanco de señal ascendente). El temporizador CEI cuenta el tiempo indicado. La salida devuelve el estado lógico "1" si RLO en la entrada de la instrucción tiene el estado lógico "1". Si el RLO cambia a "0" antes de finalizar el temporizador, el temporizador CEI en marcha se desactiva. La consulta de si el estado del temporizador es "1" devuelve en este caso el estado lógico "0". Cuando se detecta el siguiente flanco de señal ascendente en la entrada de la instrucción, se inicia de nuevo el temporizador IEC.

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles de ejecución, pues cada consulta de las salidas Q o ET actualiza la estructura del IEC_TIMER.

Para CPU S7-1200

La instrucción "Arrancar temporizador como retardo a la conexión" deposita los datos en una estructura del tipo de datos IEC_TIMER o TON_TIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Arrancar temporizador como retardo a la conexión" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME, TON_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura ET del temporizador CEI. El estado del temporizador se consulta con un contacto normalmente abierto para saber si es "1" o de un contacto normalmente cerrado para saber si es "0". La consulta de Q o ET (p. ej. "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Para poder ejecutar la instrucción "Arrancar temporizador como retardo a la conexión" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetros

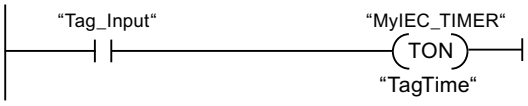
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Tiempo>	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TON_TIME	IEC_TIMER, IEC_LTIMER, TON_TIME, TON_LTIME	D, L	Temporizador CEI que se arranca.

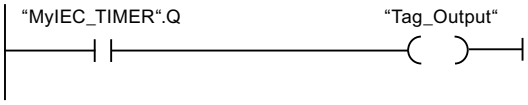
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Arrancar temporizador como retardo a la conexión" se ejecuta cuando el estado lógico del operando "Tag_Input" cambia de "0" a "1". El temporizador "MyIEC_TIMER" se arranca por el tiempo que está guardado en el operando "TagTime".



Cuando finaliza el tiempo "MyIEC_TIMER" y el operando "Tag_Input" devuelve el estado lógico "1", la consulta del estado del temporizador ("MyIEC_TIMER".Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa. Cuando el estado lógico del operando "Tag_Input" cambia a "0", la consulta del estado del temporizador devuelve el estado lógico "0" y el operando "Tag_Output" se desactiva.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

---(TOF)---: Arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Arrancar temporizador como retardo a la desconexión" permite arrancar un temporizador CEI como retardo a la desconexión por un tiempo determinado. La consulta de si el estado del temporizador es "1" devuelve el estado lógico "1" si el resultado lógico (RLO) de la entrada de la instrucción devuelve el estado lógico "0". Si el RLO cambia de "1" a "0" (flanco de señal descendente), el temporizador CEI arranca con el tiempo indicado. Mientras el temporizador CEI está activado, el estado del temporizador permanece a "1". Una vez transcurrido el tiempo y si el RLO de la entrada de la instrucción devuelve el estado lógico "0", el estado del temporizador se pone a "0". Si el RLO cambia a "1" antes de transcurrir el tiempo, el temporizador CEI activo se inicializa y el estado del temporizador permanece a "1".

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles de ejecución, pues cada consulta de las salidas Q o ET actualiza la estructura del IEC_TIMER.

Para CPU S7-1200

La instrucción "Arrancar temporizador como retardo a la desconexión" deposita los datos en una estructura del tipo de datos IEC_TIMER o TOF_TIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Arrancar temporizador como retardo a la desconexión" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME, TOF_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura ET del temporizador CEI. El estado del temporizador se consulta con un contacto normalmente abierto para saber si es "1" o de un contacto normalmente cerrado para saber si es "0". La consulta de Q o ET (p. ej. "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Para poder ejecutar la instrucción "Arrancar temporizador como retardo a la desconexión" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetros

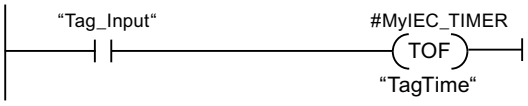
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Tiempo>	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TOF_TIME	IEC_TIMER, IEC_LTIMER, TOF_TIME, TOF_LTIME	D, L	Temporizador CEI que se arranca.

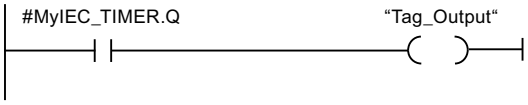
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Arrancar temporizador como retardo a la desconexión" se ejecuta cuando el estado lógico del operando "Tag_Input" cambia de "1" a "0". El temporizador #MyIEC_TIMER se inicia por el tiempo que está guardado en el operando "TagTime".



Mientras se ejecuta el temporizador #MyIEC_TIMER, la consulta del estado del temporizador (#MyIEC_TIMER.Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa. Una vez transcurrido el tiempo y si el operando "Tag_Input" devuelve el estado lógico "0", la consulta del estado del temporizador devuelve el estado lógico "0". Cuando el estado lógico del operando "Tag_Input" cambia a "1" antes de transcurrir el temporizador #MyIEC_TIMER, el temporizador se inicializa. Cuando el estado lógico del operando "Tag_Input" es "1", la consulta del estado del temporizador devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

---(TONR)---: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" permite registrar durante cuánto tiempo es "1" la señal en la entrada de la instrucción. La instrucción se lanza cuando el resultado lógico (RLO) cambia de "0" a "1" (flanco de señal ascendente). Mientras el RLO es "1", se cuenta el tiempo. Si el RLO cambia a "0", se detiene la instrucción. Si el RLO cambia nuevamente a "1", se sigue contando el tiempo. La consulta de si el estado del temporizador es "1" devuelve el estado lógico "1" si el tiempo contado supera el tiempo especificado y el RLO a la entrada de la bobina es "1".

El estado del temporizador y el tiempo transcurrido actualmente se pueden poner a "0" con la instrucción "Inicializar temporizador".

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles de ejecución, pues cada consulta de las salidas Q o ET actualiza la estructura del IEC_TIMER.

Para CPU S7-1200

La instrucción "Acumulador de tiempo" deposita los datos en una estructura del tipo de datos IEC_TIMER o TONR_TIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Acumulador de tiempo" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME. Esta estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME, TONR_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura ET del temporizador CEI. El estado del temporizador se consulta con un contacto normalmente abierto para saber si es "1" o de un contacto normalmente cerrado para saber si es "0". La consulta de Q o ET (p. ej. "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Para poder ejecutar la instrucción "Acumulador de tiempo" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetros

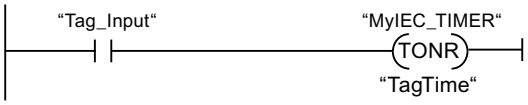
La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Tiempo>	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TONR_TIME	IEC_TIMER, IEC_LTIMER, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI que se arranca.

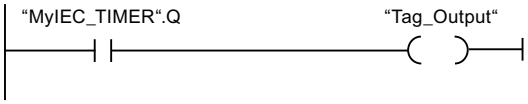
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Acumulador de tiempo" se ejecuta cuando se detecta un flanco de señal ascendente en el RLO. Mientras el operando "Tag_Input" devuelva el estado lógico "1", se cuenta el tiempo.



Si el tiempo contado supera el valor del operando "TagTime", la consulta del estado del temporizador ("MyIEC_TIMER".Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

---(RT)---: Inicializar temporizador

Descripción

La instrucción "Inicializar temporizador" permite poner a "0" un temporizador CEI. La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si fluye corriente hacia la bobina (RLO es "1"), los elementos de la estructura del temporizador en el bloque de datos indicado se ponen a "0". Si el RLO en la entrada de la instrucción es "0", el temporizador permanece invariable.

La instrucción no afecta al RLO. El RLO de la entrada de la bobina se transfiere directamente a la salida de la bobina.

A la instrucción "Inicializar temporizador" debe asignársele un temporizador CEI declarado en el programa.

Los datos de la instrucción se actualizan únicamente al efectuar una llamada de la instrucción, y no cada vez que se accede al temporizador CEI asignado. La consulta de datos es igual a la llamada de instrucción únicamente hasta la siguiente llamada de instrucción.

Parámetros

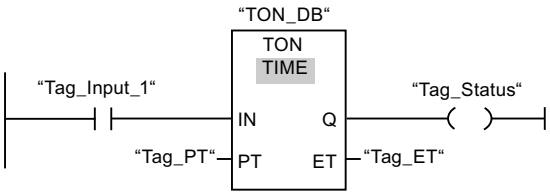
La tabla siguiente muestra los parámetros de la instrucción "Inicializar temporizador":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Temporizador CEI>	Output	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI que se inicializa.

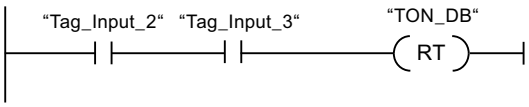
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "Tag_Input_1" cambia de "0" a "1", se ejecuta la instrucción "Retardo al conectar". El temporizador depositado en el bloque de datos de instancia "TON_DB" arranca con el tiempo especificado por el operando "Tag_PT".



Si los operandos "Tag_Input_2" y "Tag_Input_3" devuelven el estado lógico "1", la instrucción "Inicializar temporizador" se ejecuta y el temporizador depositado en el bloque de datos "TON_DB" se inicializa.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(PT)---: Cargar tiempo

Descripción

La instrucción "Cargar tiempo" permite ajustar el tiempo de un temporizador CEI. La instrucción se ejecuta en cada ciclo si el resultado lógico (RLO) de la entrada de la instrucción tiene el estado lógico "1". La instrucción escribe el tiempo indicado en la estructura del temporizador CEI indicado.

Nota

Si el temporizador CEI indicado se ejecuta mientras se ejecuta la instrucción, ésta sobrescribe el tiempo actual del temporizador CEI indicado. Esto puede modificar el estado del temporizador CEI.

A la instrucción "Cargar tiempo" debe asignársele un temporizador CEI declarado en el programa.

La actualización de los datos de la instrucción se lleva a cabo cuando se llama la instrucción y cada vez que se accede al temporizador CEI asignado. La consulta de Q o ET (p. ej. "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Parámetros

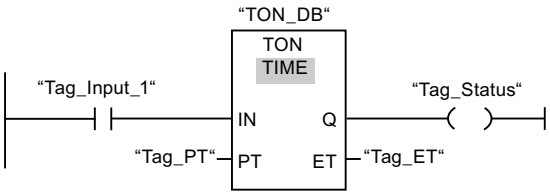
La tabla siguiente muestra los parámetros de la instrucción "Cargar tiempo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Tiempo>	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	Output	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI cuyo tiempo se ajusta.

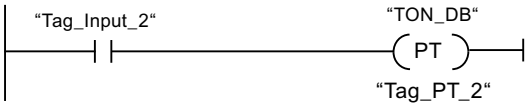
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "Tag_Input_1" cambia de "0" a "1", se ejecuta la instrucción "Retardo al conectar". El temporizador CEI depositado en el bloque de datos de instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PT".



Cuando el estado lógico del operando "Tag_Input_2" es "1", se ejecuta la instrucción "Cargar tiempo". La instrucción escribe el tiempo "Tag_PT_2" en el bloque de datos de instancia "TON_DB" y sobrescribe a la vez el valor del operando "Tag_PT" en el bloque de datos. Esto puede modificar el estado lógico del estado del temporizador en la siguiente consulta o al acceder a "MyTimer".Q o "MyTimer".ET.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Temporizadores SIMATIC

S_PULSE: Parametrizar y arrancar temporizador como impulso

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Si el estado lógico en la entrada S cambia a "0" antes de que transcurra el tiempo programado, se detiene el temporizador. En tal caso, el estado lógico de la salida Q es "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

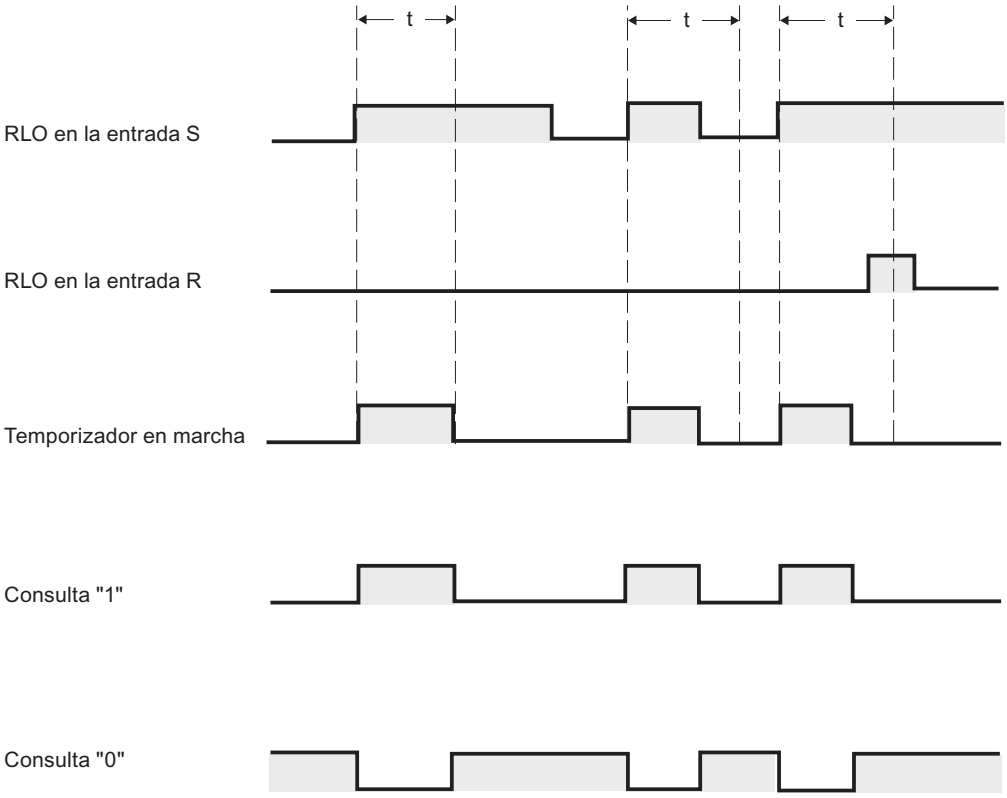
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

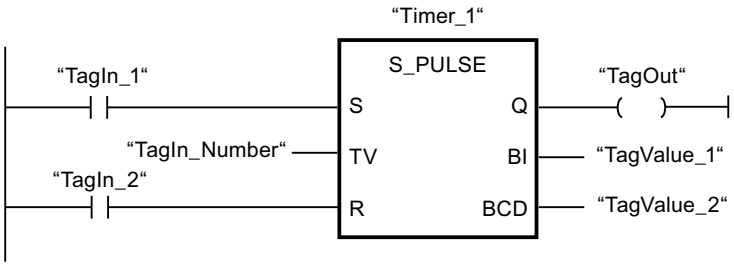
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador "Timer_1" se detiene. En ese caso, el operando "TagOut" se pone a "0".

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha y el operando "TagIn_1" tenga el estado lógico "1". Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso prolongado" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Mientras se ejecuta el temporizador, la salida Q devuelve el estado lógico "1". Una vez transcurrido el tiempo, la salida Q se pone a "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, este se reinicia con el tiempo programado en la entrada TV.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

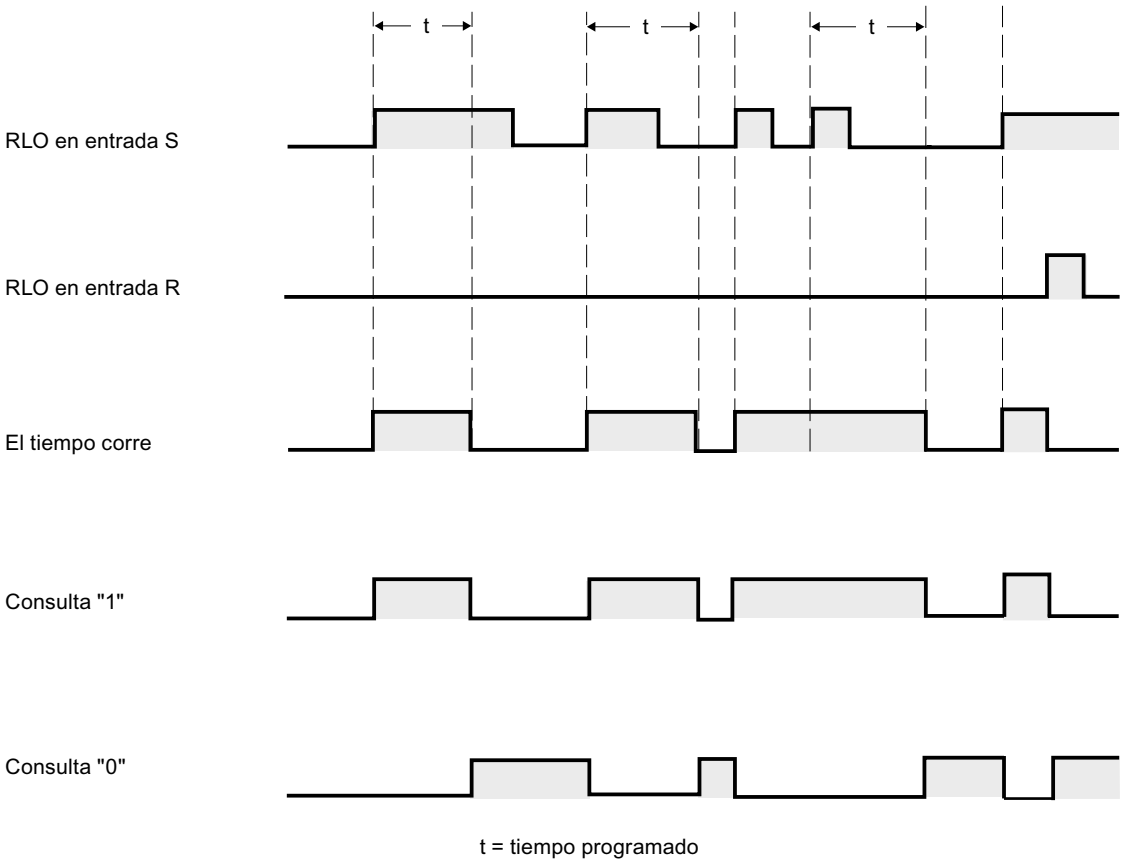
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

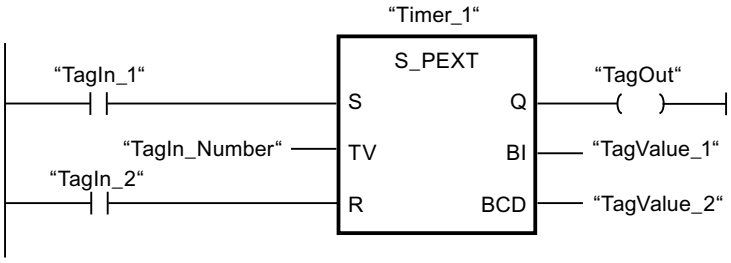
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en la entrada S. Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha. Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" se arranca un temporizador programado como retardo a la conexión al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Una vez transcurrido el tiempo debidamente y si la entrada S aún tiene el valor lógico "1", la salida Q devuelve el valor lógico "1". Si el estado de la entrada S cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En este caso, la salida Q adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia de "0" a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. En ese caso, el estado lógico de la salida Q es "0". El temporizador se reinicia cuando el estado lógico de la entrada R es "1", incluso si el temporizador no se ejecuta y el RLO en la entrada S es "1".

El temporizador de la instrucción se indica en el comodín situado sobre el cuadro. El temporizador debe estar declarado con el tipo de datos TIMER.

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

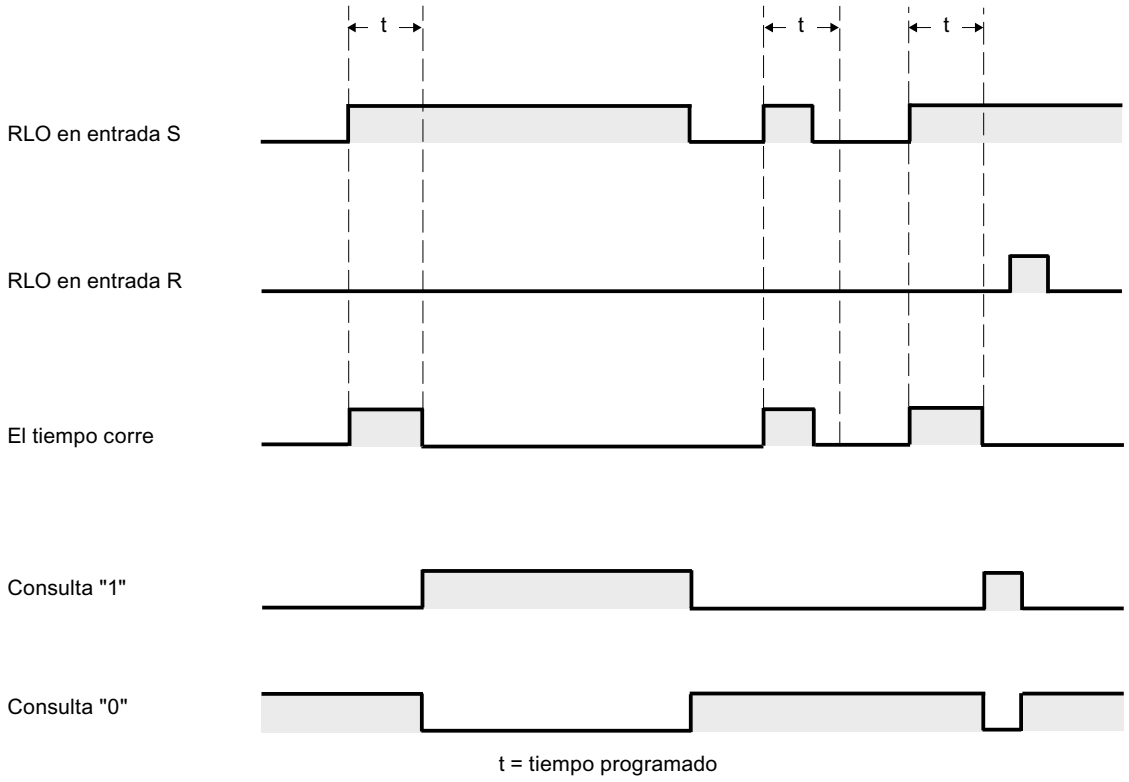
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

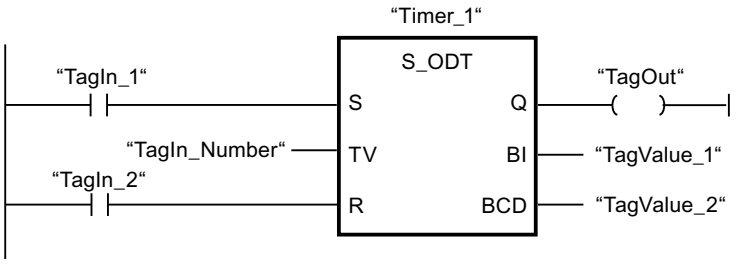
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el operando tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. El operando "TagOut" devuelve en ese caso el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Una vez transcurrido el tiempo, la salida Q devuelve el estado lógico "1" independientemente del estado lógico de la entrada S. Si el estado lógico de la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, el temporizador vuelve a arrancar con el tiempo programado (TV).

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" de la entrada R pone a "0" el valor actual de tiempo y la base de tiempo, independientemente del estado lógico de la entrada de arranque S. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

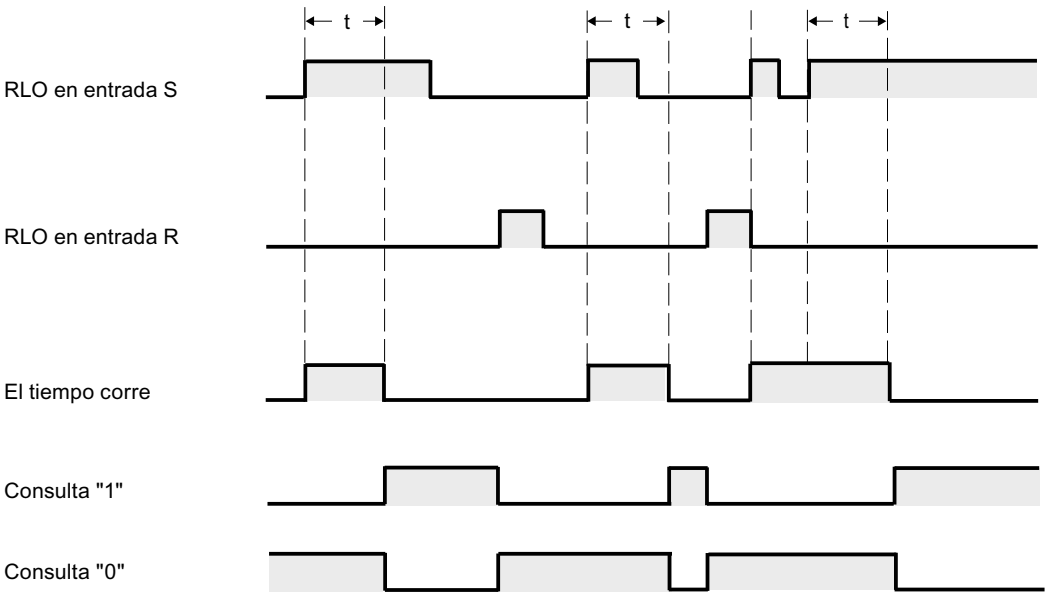
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

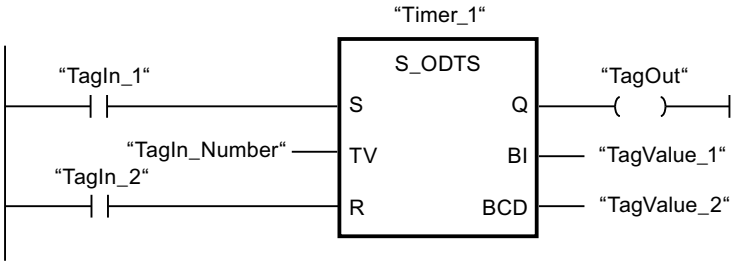
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number" aunque el estado lógico del operando "TagIn_1" cambie a "0". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente) en la entrada S. El temporizador empieza a contar con el tiempo programado (TV). Mientras el temporizador está en marcha o la entrada S devuelve el estado lógico "1", la salida Q tiene el estado lógico "1". Una vez transcurrido el tiempo y si el estado lógico en la entrada S es "0", el estado lógico de la salida Q se pone a "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras el temporizador está funcionando, este se detiene. El temporizador no vuelve a arrancar hasta que no se detecta un flanco de señal descendente en la entrada S.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" en la entrada R pone a "0" el valor de tiempo y la base de tiempo actuales. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

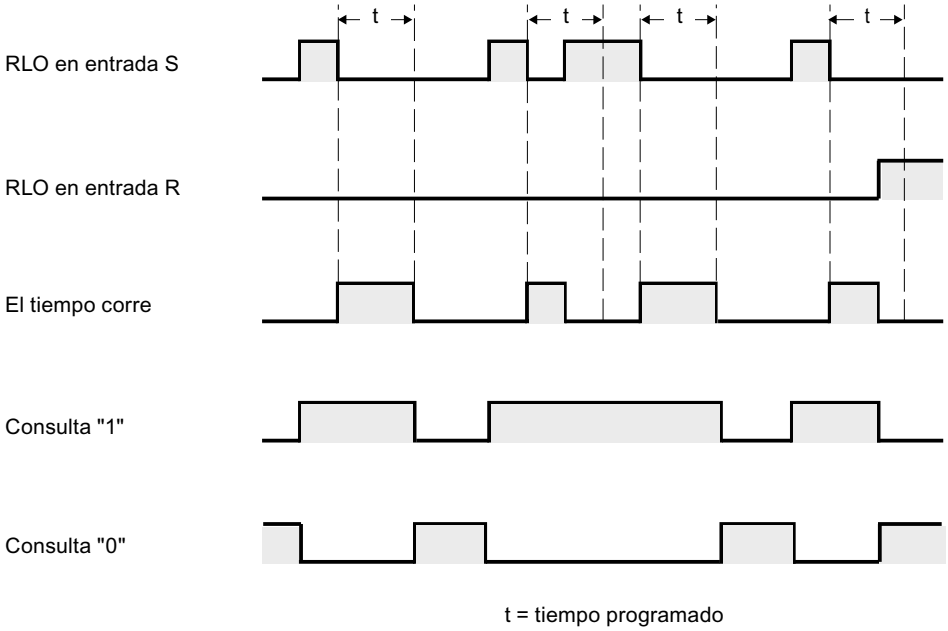
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

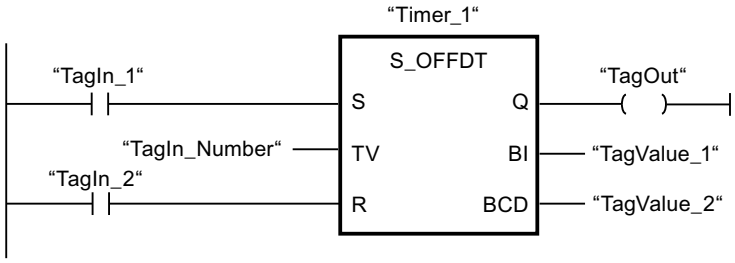
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador está en marcha o el operando "TagIn_1" tiene el estado lógico "0". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

---(SP): Arrancar temporizador como impulso

Descripción

Con la instrucción "Arrancar temporizador como impulso" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador cuenta el tiempo indicado mientras el RLO devuelva el estado lógico "1". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" antes de que haya transcurrido el valor de tiempo, el temporizador se detiene. En ese caso, una consulta de si el estado del temporizador es "1" da como resultado "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

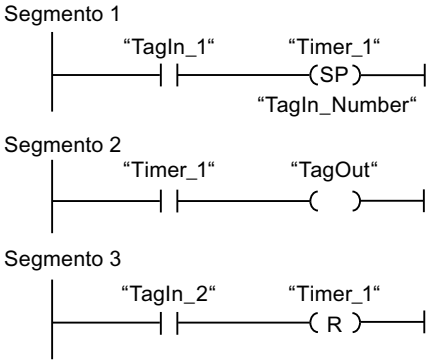
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

---(SE): Arrancar temporizador como impulso prolongado

Descripción

Con la instrucción "Arrancar temporizador como impulso prolongado" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras el temporizador está funcionando, este se arranca de nuevo con el tiempo programado. La consulta sobre si el estado es "1" da como resultado "0" si el tiempo ha transcurrido.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

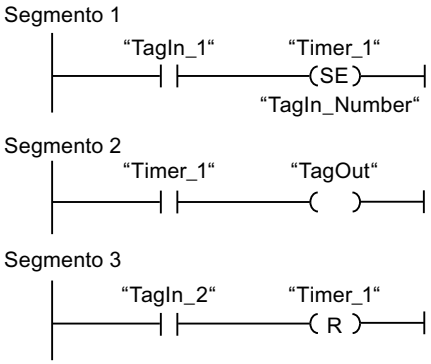
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en el RLO. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

---(SD): Arrancar temporizador como retardo a la conexión

Descripción

Con la instrucción "Arrancar temporizador como retardo a la conexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador empieza a contar con el tiempo indicado y sigue contando mientras el RLO sea "1". Una vez transcurrido el tiempo y si el RLO todavía tiene el estado lógico "1", una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En ese caso, la consulta de si el estado del temporizador es "1" devuelve el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

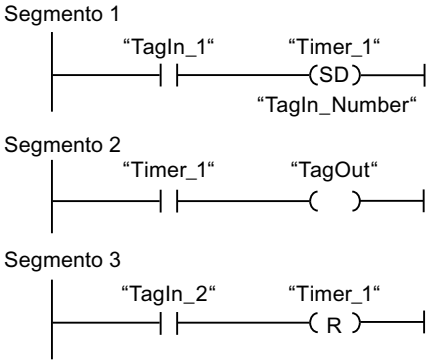
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el RLO tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

---(SS): Arrancar temporizador como retardo a la conexión con memoria

Descripción

Con la instrucción "Arrancar temporizador como retardo a la conexión con memoria" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Si ha transcurrido el tiempo, una consulta de si el estado del temporizador es "1" da como resultado "1". Una vez transcurrido el tiempo, el temporizador no puede volver a ponerse en marcha hasta que no se haya reiniciado explícitamente.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

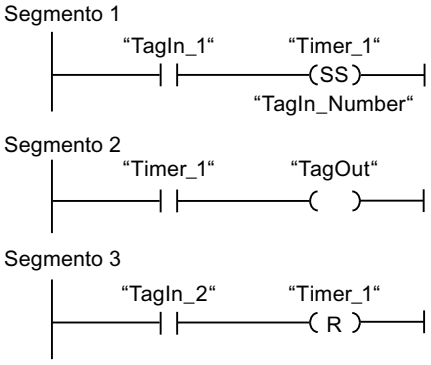
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

---(SF): Arrancar temporizador como retardo a la desconexión

Descripción

Con la instrucción "Arrancar temporizador como retardo a la desconexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente). El temporizador empieza a contar con el tiempo indicado. Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras se ejecuta el temporizador, este se inicializa. El temporizador siempre arranca de nuevo cuando el RLO cambia de "1" a "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

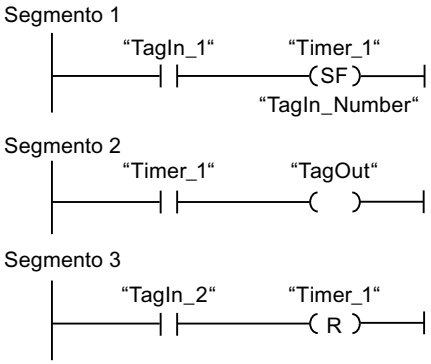
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Mientras el temporizador está en marcha, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 4093)

Contadores

Contadores CEI

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor actual de contaje de la salida CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Cuando se alcanza el valor límite superior, el estado lógico de la entrada CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. El estado lógico de la salida Q es determinado por el parámetro PV. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0".

El valor de la salida CV se pone a cero cuando el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de la entrada CU no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Para CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se declara del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<Contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTU_<Tipo de datos> o IEC_<Contador> en la sección "Static" de un bloque (p. ej. #MyIEC_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador ascendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

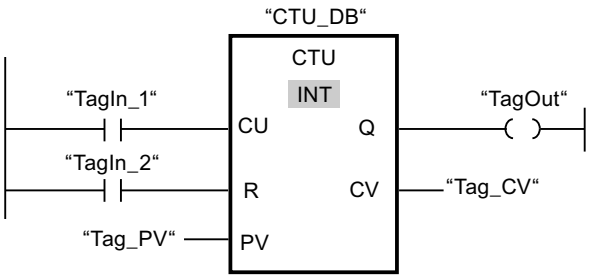
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
CU	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de reset
PV	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor actual de contaje del operando "Tag_CV" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos (INT = 32767).

El valor del parámetro PV se utiliza como límite para determinar la salida "TagOut". La salida "TagOut" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor del operando "Tag_PV". En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Remanencia de las variables locales (Página 2660)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor de la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos indicado. Cuando se alcanza el valor límite inferior, el estado lógico de la entrada CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. Si el valor actual de contaje es menor o igual a cero, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0".

El valor de la salida CV se pone al valor del parámetro PV cuando el estado lógico de la entrada LD cambia a "1". Mientras la entrada LD tenga el estado lógico "1", el estado lógico de la entrada CD no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Para CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • CTD_LINT / CTD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se declara del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<Contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTD_<Tipo de datos> o IEC_<Contador> o en la sección "Static" de un bloque (p. ej. #MyIEC_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

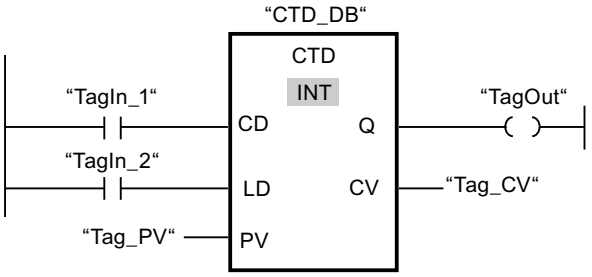
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
CD	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador descendente" y el valor de la salida "Tag_CV" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos indicado (INT = -32768).

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno y se deposita en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje de la salida CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor actual de contaje no se modifica en la salida CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se

alcanza el valor límite inferior del tipo de datos indicado, ya no se decrementa el valor de contaje.

Si el estado lógico de la entrada LD cambia a "1", el valor de contaje de la salida CV adopta el valor del parámetro PV. Mientras la entrada LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", un cambio del estado lógico de las entradas CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en la salida QU. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida QU adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QU es "0".

El estado del contador descendente se puede consultar en la salida QD. Si el valor actual de contaje es menor o igual a cero, la salida QD adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QD es "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Para CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • CTUD_LINT / CTUD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se declara del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<Contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTUD_<Tipo de datos> o IEC_<Contador> en la sección "Static" de un bloque (p. ej. #MyIEC_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador ascendente - descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
CU	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje descendente

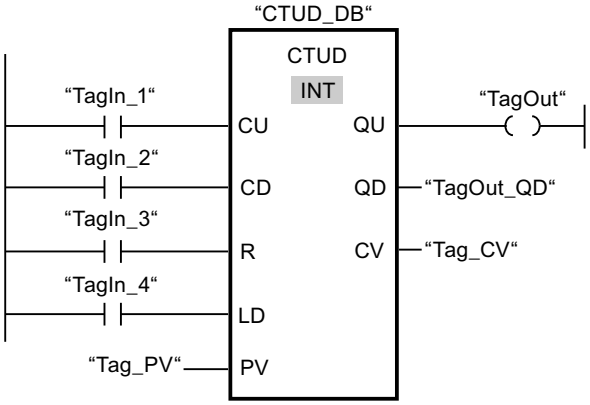
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
R	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU.
QU	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado del contador ascendente
QD	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado del contador descendente
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_1" o "TagIn_2" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Contador ascendente - descendente". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_1", el valor de contaje actual se incrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada "TagIn_2", el valor de contaje se decrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior (32767). Cuando se detecta un flanco de señal ascendente en la entrada CD, el valor de contaje se decrementa hasta alcanzar el valor límite inferior INT = -32768.

La salida "TagOut" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor de la entrada "Tag_PV". En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

La salida "TagOut_QD" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "TagOut_QD" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

Contadores SIMATIC

S_CUD: Parametrizar e incrementar/decrementar contador

Descripción

La instrucción "Parametrizar e incrementar/decrementar contador" permite incrementar y decrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje no se modifica.

El valor de contaje sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior "0", no se decrementa más el valor de contaje.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si se inicializa el contador y el RLO en las entradas CU y CD es "1", el contador solo cuenta una vez en el próximo ciclo aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU, CD y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar/decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

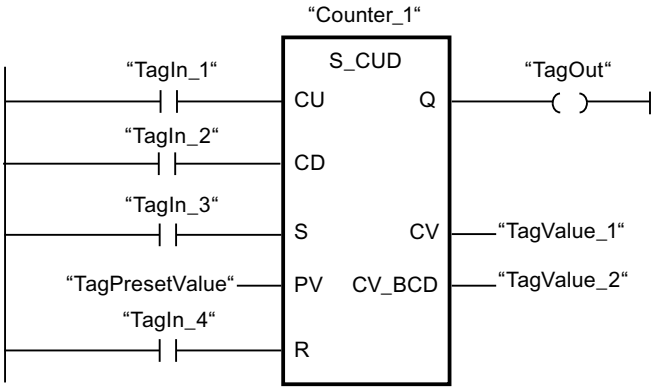
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar/decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_1" o "TagIn_2" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Parametrizar e incrementar/decrementar contador". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_1" y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_2" y el valor de contaje actual es mayor que "0", este valor se decrementará en uno.

Si el estado lógico de la entrada "TagIn_3" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_4" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

S_CU: Parametrizar e incrementar contador

Descripción

La instrucción "Parametrizar e incrementar contador" permite incrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue incrementando hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CU es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

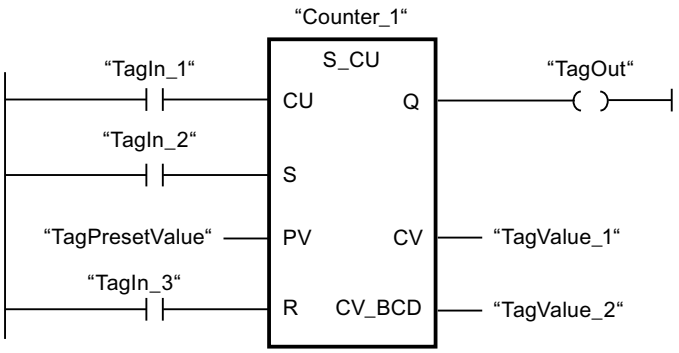
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
S	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

S_CD: Parametrizar y decrementar contador

Descripción

La instrucción "Parametrizar y decrementar contador" permite decrementar el valor de un contador. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite inferior, el valor de contaje no se decrementa al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar y decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

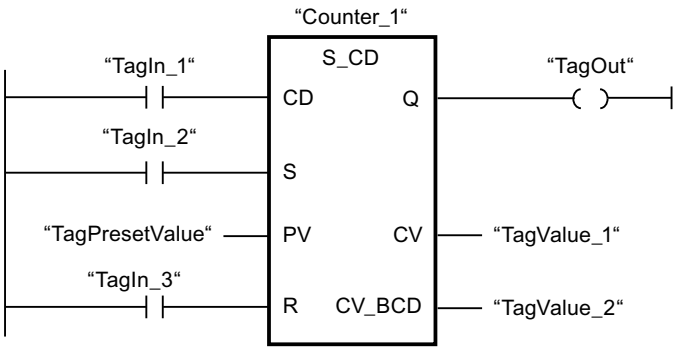
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CD	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es mayor que "0", este valor se decrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(SC): Poner contador al valor inicial

Descripción

Con la instrucción "Poner contador al valor inicial" se ajusta el valor de un contador. La instrucción se ejecuta cuando el resultado lógico (RLO) de la entrada de la instrucción cambia de "0" a "1". Si se ejecuta la instrucción, el contador se pone al valor de contaje indicado.

La instrucción "Poner contador al valor inicial" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

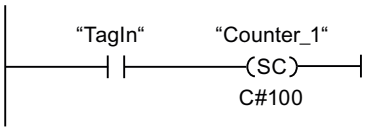
La tabla siguiente muestra los parámetros de la instrucción "Poner contador al valor inicial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Valor de contaje>	Input	WORD	I, Q, M, D, L o constante	Valor con el que se preajusta el contador codificado en BCD. (C#0 a C#999)
<Contadores>	InOut/Input	COUNTER	C	Contador que se preajusta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn" cambia de "0" a "1", el contador "Counter_1" se pone a "100".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(CU): Contador ascendente

Descripción

Con la instrucción "Contador ascendente" se puede aumentar en uno el valor del contador cuando hay un flanco ascendente en el resultado lógico (RLO). El valor de contaje puede seguir incrementándose hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

La instrucción "Contador ascendente" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

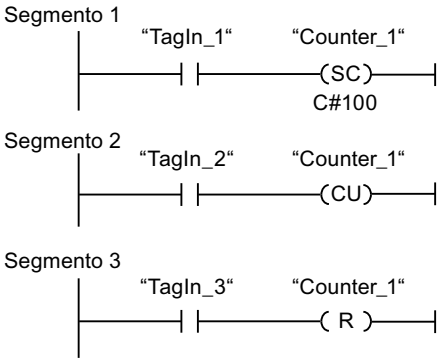
La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador cuyo valor se aumenta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se incrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(CD): Contador descendente

Descripción

Con la instrucción "Contador descendente" se puede reducir en uno el valor del contador cuando hay un flanco ascendente en el resultado lógico (RLO). El valor de contaje puede

seguir decrementándose hasta alcanzar el límite "0". Una vez alcanzado el valor límite, el valor de contaje no se modifica al detectarse un flanco de señal ascendente.

La instrucción "Contador descendente" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

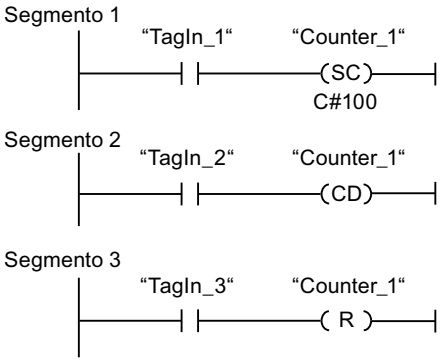
La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador cuyo valor se decrementa.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se decrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Comparación

CMP ==: Igual

Descripción

La instrucción "Igual" permite consultar si son iguales el primer (<Operando1>) y segundo (<Operando2>) valor de comparación.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'AA'	'AA'	1
'Hello World'	'HelloWorld'	0
'AA'	'aa'	0

Con la instrucción "Igual" también se pueden comparar caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Si está activada la verificación CEI, los operandos que comparar deben ser del mismo tipo de datos. Si no está activada la verificación CEI, el ancho (longitud) de los operandos debe ser igual. Si se comparan números en coma flotante, los operandos que comparar deben ser del mismo tipo de datos, independientemente del ajuste de la verificación CEI.

Nota

Comparación de números en coma flotante

Si desea comparar tipos de datos REAL o LREAL, utilice en lugar de la instrucción "CMP ==: Igual" la instrucción "IN_RANGE: Valor dentro del rango".

Nota

Comparación del tipo de datos PORT

Para poder comparar operandos del tipo de datos PORT con la instrucción "Igual", deberá seleccionar el tipo de datos WORD en la lista desplegable del cuadro de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Igual":

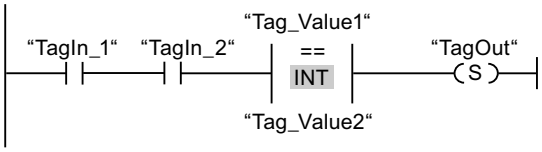
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Operando1>	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" = "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

CMP <>: Diferente

Descripción

La instrucción "Diferente" permite consultar si son diferentes el primer (<Operando1>) y segundo (<Operando2>) valor de comparación.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'AA'	'aa'	1
'Hello World'	'HelloWorld'	1
'AA'	'AA'	0

Con la instrucción "Diferente" también se pueden comparar caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Si está activada la verificación CEI, los operandos que comparar deben ser del mismo tipo de datos. Si no está activada la verificación CEI, el ancho (longitud) de los operandos debe ser

igual. Si se comparan números en coma flotante, los operandos que comparar deben ser del mismo tipo de datos, independientemente del ajuste de la verificación CEI.

Nota

Comparación del tipo de datos PORT

Para poder comparar operandos del tipo de datos PORT con la instrucción "Diferente", deberá seleccionar el tipo de datos WORD en la lista desplegable del cuadro de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Diferente":

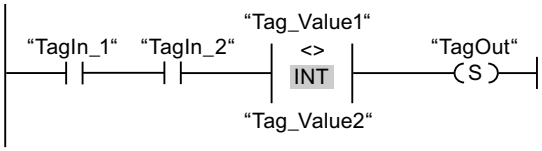
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Operando1>	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <> "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

CMP >=: Mayor o igual

Descripción

La instrucción "Mayor o igual" permite consultar si el primer valor de comparación (<Operando1>) es mayor o igual que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más larga se considera la mayor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'BB'	'AA'	1
'AAA'	'AA'	1
'Hello World'	'Hello World'	1
'Hello World'	'HelloWorld'	0
'AA'	'aa'	0
'AAA'	'a'	0

Con la instrucción "Mayor o igual" también se pueden comparar caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante del <Operando1> es mayor (más reciente) o igual al instante del <Operando2>.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor o igual":

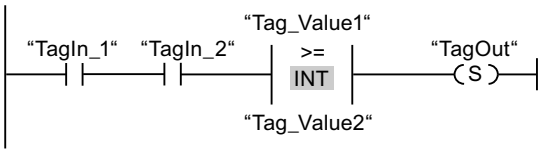
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Operando1>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" >= "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3210)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

CMP <=: Menor o igual

Descripción

La instrucción "Menor o igual" permite consultar si el primer valor de comparación (<Operando1>) es menor o igual que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más corta se considera la menor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'AA'	'aa'	1
'AAA'	'a'	1
'Hello World'	'Hello World'	1
'HelloWorld'	'Hello World'	0
'BB'	'AA'	0
'AAA'	'AA'	0

Con la instrucción "Menor o igual" también se pueden comparar caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante del <Operando1> es menor (más antiguo) o igual al instante del <Operando2>.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor o igual":

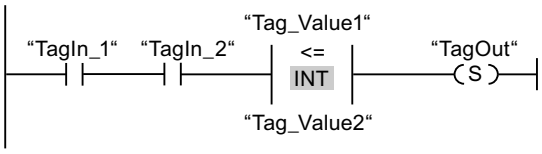
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Operando1>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <= "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

CMP >: Mayor

Descripción

La instrucción "Mayor" permite consultar si el primer valor de comparación (<Operando1>) es mayor que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más larga se considera la mayor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'BB'	'AA'	1
'AAA'	'AA'	1
'AA'	'aa'	0
'AAA'	'a'	0

Con la instrucción "Mayor" también se pueden comparar caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante del <Operando1> es mayor (más reciente) que el instante del <Operando2>.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor":

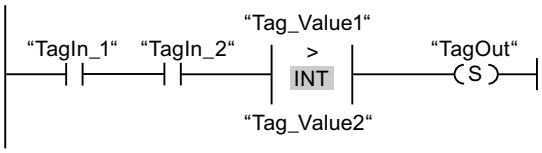
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Operando1>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" > "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

CMP <: Menor

Descripción

La instrucción "Menor" permite consultar si el primer valor de comparación (<Operando1>) es menor que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más corta se considera la menor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'AA'	'aa'	1
'AAA'	'a'	1
'BB'	'AA'	0
'AAA'	'AA'	0

Con la instrucción "Menor" también se pueden comparar caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante del <Operando1> es menor (más antiguo) que el instante del <Operando2>.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor":

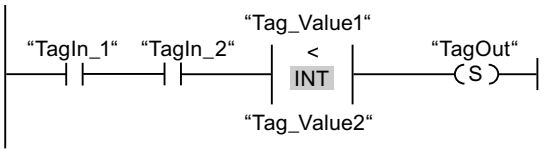
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Operando1>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" < "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3210)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

IN_RANGE: Valor dentro del rango

Descripción

La instrucción "Valor dentro del rango" permite consultar si el valor de la entrada VAL está dentro de un determinado rango de valores.

Los límites del rango de valores se determinan mediante las entradas MIN y MAX. La instrucción "Valor dentro del rango" compara el valor de la entrada VAL con los valores de las entradas MIN y MAX y transfiere el resultado a la salida del cuadro. Si el valor de la entrada VAL cumple la comparación MIN <= VAL o VAL <= MAX, la salida del cuadro devuelve el estado lógico "1". Si no se cumple la comparación, la salida del cuadro devuelve el estado lógico "0".

Si el estado lógico de la entrada del cuadro es "0", no se ejecuta la instrucción "Valor dentro del rango".

Para ejecutar la función de comparación es necesario que los valores que deben compararse sean del mismo tipo de datos y que la entrada del cuadro esté cableada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Valor dentro del rango":

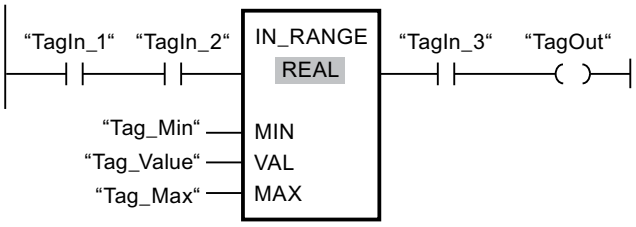
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Entrada del cuadro	Input	BOOL	I, Q, M, D, L	Resultado de la combinación lógica previa
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VAL	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Valor de comparación
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
Salida del cuadro	Output	BOOL	I, Q, M, D, L	Resultado de la comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El valor del operando "Tag_Value" está dentro del rango de valores especificado por los valores actuales de los operandos "Tag_Min" y "Tag_Max" (MIN <= VAL o VAL <= MAX).
- El operando "TagIn_3" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

OUT_RANGE: Valor fuera del rango

Descripción

La instrucción "Valor fuera del rango" permite consultar si el valor de la entrada VAL está fuera de un determinado rango de valores.

Los límites del rango de valores se determinan mediante las entradas MIN y MAX. La instrucción "Valor fuera del rango" compara el valor de la entrada VAL con los valores de las entradas MIN y MAX y transfiere el resultado a la salida del cuadro. Si el valor de la entrada VAL cumple la comparación MIN > VAL o VAL > MAX, la salida del cuadro devuelve el estado lógico "1". La salida del cuadro también devuelve el estado lógico "1" si un operando indicado del tipo de datos REAL contiene un valor no válido.

La salida del cuadro devuelve el estado lógico "0" si el valor de la entrada VAL no cumple la condición MIN > VAL o VAL > MAX .

Si el estado lógico de la entrada del cuadro es "0", no se ejecuta la instrucción "Valor fuera del rango".

Para ejecutar la función de comparación es necesario que los valores que deben compararse sean del mismo tipo de datos y que la entrada del cuadro esté cableada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Valor fuera del rango":

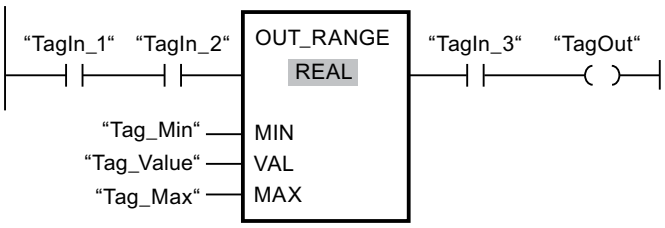
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Entrada del cuadro	Input	BOOL	I, Q, M, D, L	Resultado de la combinación lógica previa
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VAL	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Valor de comparación
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
Salida del cuadro	Output	BOOL	I, Q, M, D, L	Resultado de la comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El valor del operando "Tag_Value" está fuera del rango de valores especificado por los valores de los operandos "Tag_Min" y "Tag_Max" (MIN > VAL o VAL > MAX).
- El operando "TagIn_3" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

----I OK I----: Comprobar validez

Descripción

La instrucción "Comprobar validez" permite consultar si el valor de un operando (<Operando>) es un número en coma flotante válido. La consulta se inicia en cada ciclo del programa cuando la entrada de la instrucción tiene el estado lógico "1".

La salida de la instrucción devuelve el estado lógico "1" cuando el valor del operando es un número en coma flotante válido cuando se realiza la consulta y la entrada de la instrucción devuelve el estado lógico "1". En todos los demás casos, la salida de la instrucción "Comprobar validez" devuelve el estado lógico "0".

La instrucción "Comprobar validez" puede utilizarse en combinación con el mecanismo EN. Si el cuadro de la instrucción se conecta a una entrada de habilitación EN, ésta se activa únicamente si la consulta de la validez del valor da un resultado positivo. Esta función garantiza que una instrucción se active sólo si el valor del operando indicado es un número en coma flotante válido.

Parámetros

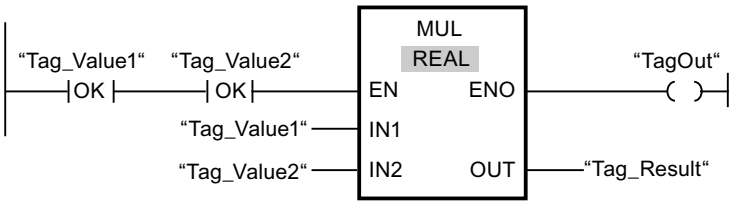
La tabla siguiente muestra los parámetros de la instrucción "Comprobar validez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L	Valor que se consulta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los valores de los operandos "Tag_Value1" y "Tag_Value2" contienen números en coma flotante válidos, se ejecuta la instrucción "Multiplicar" (MUL) y se activa la salida ENO. Cuando se ejecuta la instrucción "Multiplicar" (MUL), el valor del operando "Tag_Value1" se multiplica por el valor del operando "Tag_Value2". El resultado de la multiplicación se almacena entonces en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, las salidas ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

----I NOT_OK I----: Comprobar invalidez

Descripción

La instrucción "Comprobar invalidez" permite consultar si el valor de un operando (<Operando>) es un número en coma flotante no válido. La consulta se inicia en cada ciclo del programa cuando la entrada de la instrucción tiene el estado lógico "1".

La salida de la instrucción devuelve el estado lógico "1" cuando el valor del operando es un número en coma flotante no válido cuando se realiza la consulta y la entrada de la instrucción devuelve el estado lógico "1". En todos los demás casos, la salida de la instrucción "Comprobar invalidez" devuelve el estado lógico "0".

Parámetros

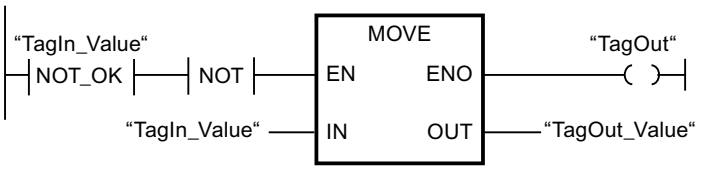
La tabla siguiente muestra los parámetros de la instrucción "Comprobar invalidez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L	Valor que se consulta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el valor del operando "TagIn_Value" no es un número en coma flotante válido, no se ejecuta la instrucción "Copiar valor" (MOVE). La salida "TagOut" se pone al estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Instrucciones VARIANT

EQ_Type: Comparar si el tipo de datos es IGUAL al tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos es IGUAL al tipo de datos de una variable" se consulta qué tipo de datos tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable (<Operando1>) es "Igual" al tipo de datos de una variable (<Operando2>) declarada en la interfaz del bloque.

El <Operando1> tiene que ser del tipo VARIANT. El <Operando2> puede ser un tipo de datos simple o un tipo de datos PLC.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El <Operando1> se indica en el comodín de operandos situado encima de la instrucción. El <Operando2> se indica en el comodín de operandos situado debajo de la instrucción.

Parámetro

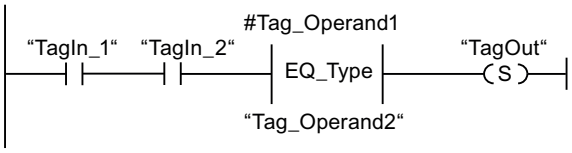
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos es IGUAL al tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	VARIANT	L	Primer operando
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es igual a "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

NE_Type: Comparar si el tipo de datos es DIFERENTE al tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos es DIFERENTE del tipo de datos de una variable" se consulta qué tipo de datos no tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable (<Operando1>) es "Diferente" al tipo de datos de una variable (<Operando2>) declarada en la interfaz del bloque.

El <Operando1> tiene que ser del tipo VARIANT. El <Operando2> puede ser un tipo de datos simple o un tipo de datos PLC.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El <Operando1> se indica en el comodín de operandos situado encima de la instrucción. El <Operando2> se indica en el comodín de operandos situado debajo de la instrucción.

Parámetro

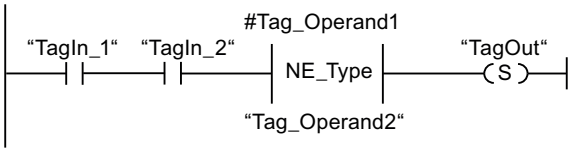
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos es DIFERENTE del tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	VARIANT	L	Primer operando
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es diferente de "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

EQ_ElemType: Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable" se consulta qué tipo de datos tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable (<Operando1>) es "Igual" al tipo de datos de una variable (<Operando2>) declarada en la interfaz del bloque.

El <Operando1> tiene que ser del tipo VARIANT. El <Operando2> puede ser un tipo de datos simple o un tipo de datos PLC.

Si el tipo de datos de la variable VARIANT (<Operando1>) es un ARRAY, se compara el tipo de datos de los elementos ARRAY.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El <Operando1> se indica en el comodín de operandos situado encima de la instrucción. El <Operando2> se indica en el comodín de operandos situado debajo de la instrucción.

Parámetro

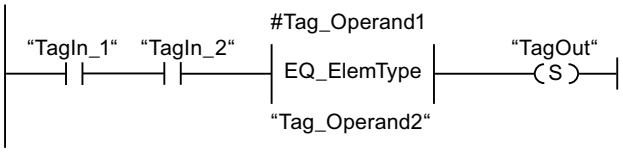
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	VARIANT	L	Primer operando
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es igual a "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

NE_ElemType: Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable" se consulta qué tipo de datos no tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable (<Operando1>) es "Diferente" al tipo de datos de una variable (<Operando2>) declarada en la interfaz del bloque.

El <Operando1> tiene que ser del tipo VARIANT. El <Operando2> puede ser un tipo de datos simple o un tipo de datos PLC.

Si el tipo de datos de la variable VARIANT (<Operando1>) es un ARRAY, se compara el tipo de datos de los elementos ARRAY.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El <Operando1> se indica en el comodín de operandos situado encima de la instrucción. El <Operando2> se indica en el comodín de operandos situado debajo de la instrucción.

Parámetro

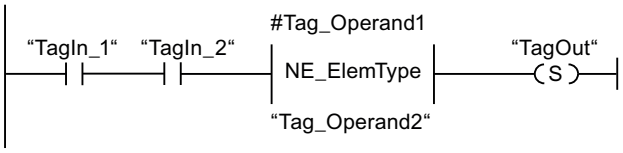
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	VARIANT	L	Primer operando
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es diferente de "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

IS_NULL: Comparar si es IGUAL A CERO

Descripción

La instrucción "Comparar si es IGUAL A CERO" permite saber si VARIANT no apunta a un objeto.

El <Operando1> tiene que ser del tipo VARIANT.

Nota

La variable VARIANT apunta a un puntero ANY

Si la variable VARIANT apunta a un puntero ANY, la instrucción devuelve siempre el resultado RLO = "0" aunque el puntero ANY tenga el valor CERO.

Parámetro

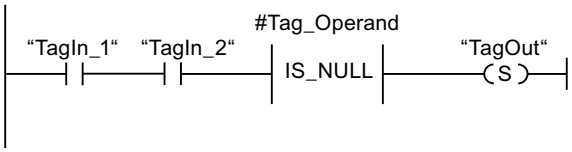
La tabla siguiente muestra los parámetros de la instrucción "Comparar si es IGUAL A CERO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	VARIANT	L	Operando que se compara si es IGUAL A CERO

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand" no apunta a ningún objeto.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

NOT_NULL: Comparar si es DIFERENTE DE CERO

Descripción

La instrucción "Comparar si es DIFERENTE DE CERO" permite saber si VARIANT apunta a un objeto.

El <Operando1> tiene que ser del tipo VARIANT.

Nota

La variable VARIANT apunta a un puntero ANY

Si la variable VARIANT apunta a un puntero ANY, la instrucción devuelve siempre el resultado RLO = "1" aunque el puntero ANY tenga el valor CERO.

Parámetro

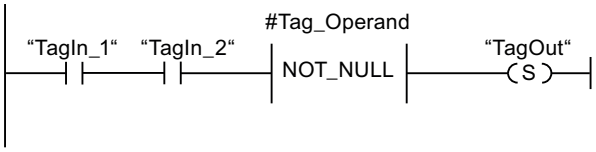
La tabla siguiente muestra los parámetros de la instrucción "Comparar si es DIFERENTE DE CERO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	VARIANT	L	Operando que se compara si es DIFERENTE DE CERO

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand" apunta a un objeto.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

IS_ARRAY: Consultar si es un ARRAY

Descripción

La instrucción "Consultar si es un ARRAY" permite saber si VARIANT apunta a una variable del tipo de datos ARRAY.

El <Operando1> tiene que ser del tipo VARIANT.

Parámetro

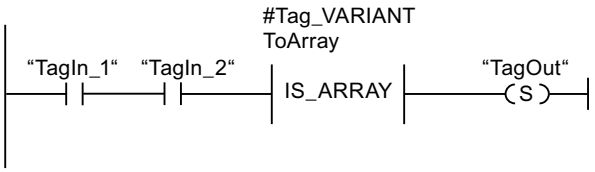
La tabla siguiente muestra los parámetros de la instrucción "Consultar si es un ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	VARIANT	L	Operando que se consulta si es un ARRAY

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_VARIANTToArray" es del tipo de datos ARRAY.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Funciones matemáticas

CALCULATE: Calcular

Descripción

La instrucción "Calcular" permite definir y ejecutar una expresión para calcular operaciones matemáticas o combinaciones lógicas complejas en función del tipo de datos seleccionado.

El tipo de datos de la instrucción se selecciona en la lista desplegable "<???>" del cuadro de la instrucción. Dependiendo del tipo de datos seleccionado se puede combinar la funcionalidad de determinadas instrucciones para ejecutar un cálculo complejo. La expresión que debe

calcularse se indica en un cuadro de diálogo que se abre con el símbolo de "calculadora" situado en el margen superior derecho del cuadro de la instrucción. La expresión puede contener los nombres de los parámetros de entrada y la sintaxis de las instrucciones. No está permitido indicar nombres o direcciones de operandos.

El cuadro de la instrucción contiene en estado básico al menos 2 entradas (IN1 e IN2). El número de entradas es ampliable. Las entradas insertadas se numeran en orden ascendente en el cuadro.

Los valores de las entradas se utilizan para ejecutar la expresión indicada. No obstante, no es necesario utilizar en la expresión todas las entradas definidas. El resultado de la instrucción se transfiere a la salida OUT.

Nota

Cuando una de las operaciones matemáticas falla en la expresión, no se transfieren resultados a la salida OUT y la salida de habilitación ENO devuelve el estado lógico "1".

Si en la expresión se utilizan entradas que no están disponibles en el cuadro, éstas se insertan automáticamente. Para ello, es indispensable que la numeración de las entradas nuevas que deben definirse en la expresión no contenga huecos. Por ejemplo, es posible no utilizar la entrada IN4 en la expresión si la entrada IN3 no está definida.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado final o intermedio de la instrucción "Calcular" está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.
- Se ha producido un error al ejecutar una de las instrucciones indicadas en la expresión.

La tabla siguiente muestra las instrucciones que pueden ejecutarse combinadas en función del tipo de datos seleccionado en la expresión de la instrucción "Calcular":

Tipo de datos	Instrucción	Sintaxis	Ejemplo
Secuencias de bits	AND: Operación lógica Y	AND	IN1 AND IN2 OR IN3
	OR: Operación lógica O	OR	
	XOR: Operación lógica O-exclusiva	XOR	
	INV: Complemento a 1	NOT	
	SWAP: Cambiar disposición [1)]	SWAP	
Enteros	ADD: Sumar	+	(IN1 + IN2) * IN3; (ABS(IN2)) * (ABS(IN1))
	SUB: Restar	-	
	MUL: Multiplicar	*	
	DIV: Dividir	/	
	MOD: Obtener resto de división	MOD	
	INV: Complemento a 1	NOT	
	NEG: Generar complemento a dos	-(in1)	
	ABS: Calcular valor absoluto	ABS( )	

Tipo de datos	Instrucción	Sintaxis	Ejemplo
Números en coma flotante	ADD: Sumar	+	((SIN(IN2) * SIN(IN2) + (SIN(IN3) * SIN(IN3)) / IN3)); (SQR(SIN(IN2)) + (SQR(COS(IN3)) / IN2))
	SUB: Restar	-	
	MUL: Multiplicar	*	
	DIV: Dividir	/	
	EXPT: Elevar a potencia	**	
	ABS: Calcular valor absoluto	ABS( )	
	SQR: Calcular cuadrado	SQR( )	
	SQRT: Calcular raíz cuadrada	SQRT( )	
	LN: Calcular logaritmo natural	LN( )	
	EXP: Calcular valor exponencial	EXP( )	
	FRAC: Determinar decimales	FRAC( )	
	SIN: Calcular valor de seno	SIN( )	
	COS: Calcular valor de coseno	COS( )	
	TAN: Calcular valor de tangente	TAN( )	
	ASIN: Calcular valor de arcoseno	ASIN( )	
	ACOS: Calcular valor de arcocoseno	ACOS( )	
	ATAN: Calcular valor de arcotangente	ATAN( )	
	NEG: Generar complemento a dos	-(in1)	
	TRUNC: Truncar a entero	TRUNC( )	
	ROUND: Redondear número	ROUND( )	
	CEIL: Redondear un número en coma flotante al siguiente entero superior	CEIL( )	
	FLOOR: Redondear un número en coma flotante al siguiente entero inferior	FLOOR( )	
[1] No es posible para el tipo de datos BYTE.			

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular":

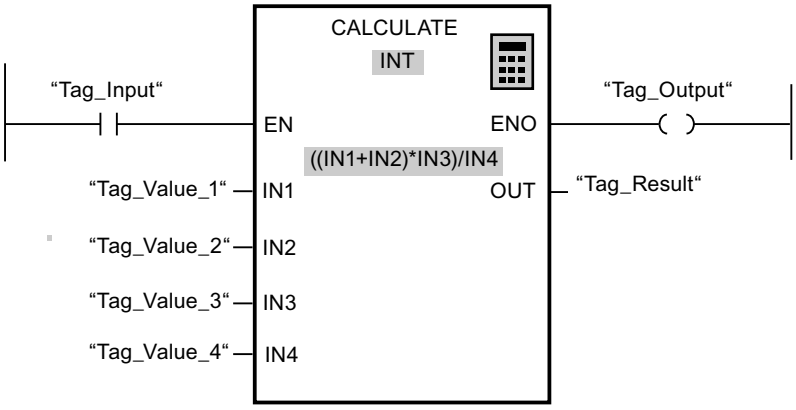
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primera entrada disponible
IN2	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segunda entrada disponible

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
INn	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Entradas insertadas adicionalmente
OUT	Output	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P	Salida a la que se transfiere el resultado final.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value_1	4
IN2	Tag_Value_2	4
IN3	Tag_Value_3	3
IN4	Tag_Value_4	2
OUT	Tag_Result	12

Cuando la entrada "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular". El valor del operando "Tag_Value_1" se suma al valor del operando "Tag_Value_2". La suma se multiplica por el valor del operando "Tag_Value_3". El producto se divide por el valor del operando "Tag_Value_4". El cociente se transfiere como resultado final al operando "Tag_Result" de la salida OUT de la instrucción. Si no ocurren errores al ejecutar las distintas instrucciones, la salida de habilitación ENO y el operando "Tag_Output" se ponen a "1".

Consulte también

Utilizar la instrucción "Calcular" (Página 2690)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ADD: Sumar

Descripción

La instrucción "Sumar" permite sumar el valor de la entrada IN1 al valor de la entrada IN2 y consultar la suma en la salida OUT (OUT:= IN1+IN2).

El cuadro de la instrucción contiene en estado estacionario mín. 2 entradas (IN1 e IN2). El número de entradas es ampliable. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se suman los valores de todos los parámetros de entrada disponibles. La suma se guarda en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer sumando
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo sumando

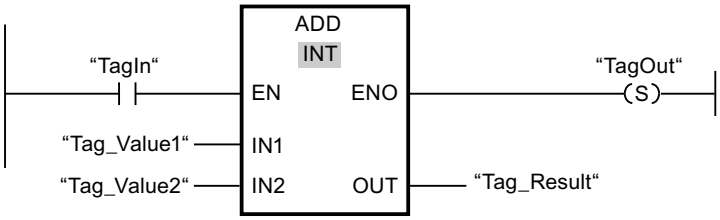
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
INn	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Valores de entrada opcionales que se suman.
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Suma

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Sumar". El valor del operando "Tag_Value1" se suma al valor del operando "Tag_Value2". El resultado de la suma se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2682)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

SUB: Restar

Descripción

La instrucción "Restar" permite restar el valor de la entrada IN2 del valor de la entrada IN1 y consultar la diferencia en la salida OUT (OUT:= IN1-IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Restar":

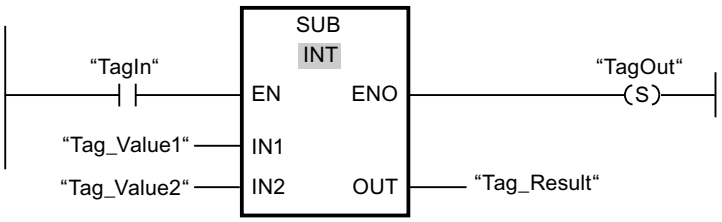
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Sustraendo
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Diferencia

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Restar". El valor del operando "Tag_Value2" se resta del valor del operando "Tag_Value1". El resultado de la resta se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar

la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2682)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

MUL: Multiplicar

Descripción

La instrucción "Multiplicar" permite multiplicar el valor de la entrada IN1 por el valor de la entrada IN2 y consultar el producto en la salida OUT (OUT:= IN1*IN2).

El número de entradas del cuadro de la instrucción se puede ampliar. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se multiplican los valores de todos los parámetros de entrada disponibles. El producto se guarda en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada EN devuelve el estado lógico "0".
- El resultado está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplicar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Multiplicador
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Multiplicando

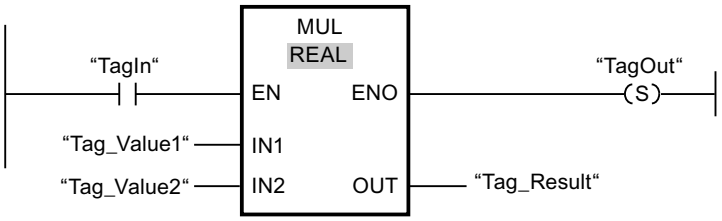
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
INn	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Valores de entrada opcionales que se pueden multiplicar.
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Producto

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar". El valor del operando "Tag_Value1" se multiplica por el valor del operando "Tag_Value2". El resultado de la multiplicación se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2682)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

DIV: Dividir

Descripción

La instrucción "Dividir" permite dividir el valor de la entrada IN1 entre el valor de la entrada IN2 y consultar el cociente en la salida OUT (OUT:= IN1/IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Dividir":

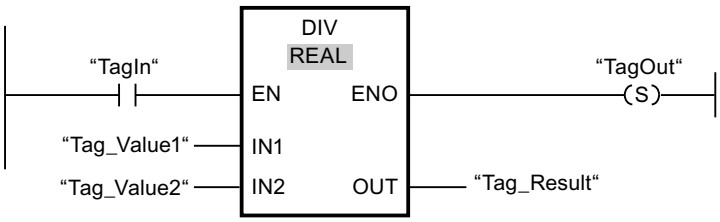
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Valor del cociente

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Dividir". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resultado de la división se almacena en el operando "Tag_Result". Si no se producen errores

al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2682)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

MOD: Obtener resto de división

Descripción

La instrucción "Obtener resto de división" permite dividir el valor de la entrada IN1 entre el valor de la entrada IN2 y consultar el resto de división en la salida OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Obtener resto de división":

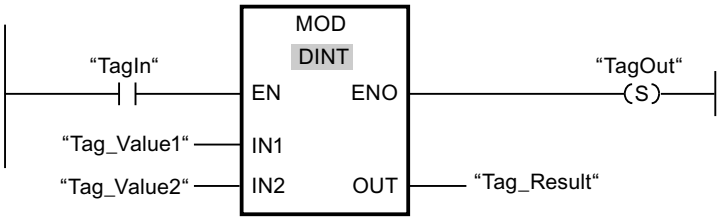
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros	I, Q, M, D, L, P	Resto de división

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Obtener resto de división". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resto de la división se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2682)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

NEG: Generar complemento a dos

Descripción

La instrucción "Generar complemento a dos" permite cambiar el signo del valor de la entrada IN y consultar el resultado en la salida OUT. Por ejemplo, si la entrada IN tiene un valor positivo, se deposita el equivalente negativo de este valor en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a dos":

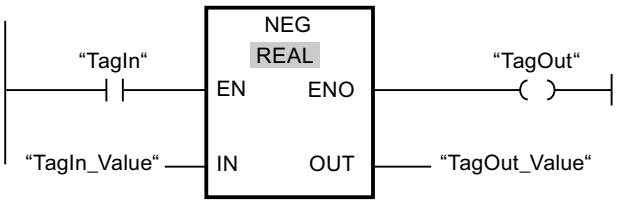
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P	Complemento a dos del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Generar complemento a dos". Se invierte el signo del valor de la entrada "TagIn_Value" y el resultado se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

INC: Incrementar

Descripción

La instrucción "Incrementar" permite cambiar el valor del operando del parámetro IN/OUT al siguiente valor superior y consultar el resultado. La ejecución de la instrucción "Incrementar" se inicia solo si el estado lógico de la entrada de habilitación EN es "1". Si no ocurre ningún rebase por exceso durante la ejecución, la salida de habilitación ENO también devuelve el estado lógico "1".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Incrementar":

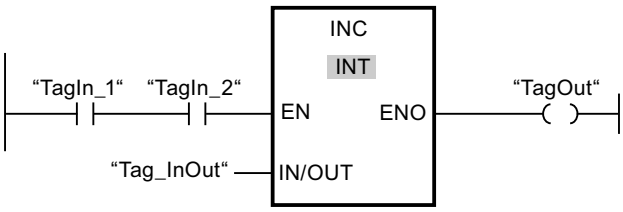
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN/OUT	InOut	Enteros	I, Q, M, D, L	Valor que se incrementa.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", el valor del operando "Tag_InOut" se incrementa en uno y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

DEC: Decrementar

Descripción

La instrucción "Decrementar" permite cambiar el valor del operando del parámetro IN/OUT al siguiente valor inferior y consultar el resultado. La ejecución de la instrucción "Decrementar" se inicia si el estado lógico de la entrada de habilitación EN es "1". Si no se rebasa por defecto el rango de valores del tipo de datos seleccionado durante la ejecución, la salida ENO devuelve asimismo el estado lógico "1".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Decrementar":

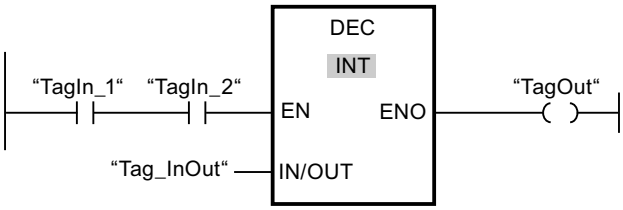
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN/OUT	InOut	Enteros	I, Q, M, D, L	Valor que se decrementa.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", el valor del operando "Tag_InOut" se decrementa en uno y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" permite calcular el valor absoluto del valor indicado en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor absoluto":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación

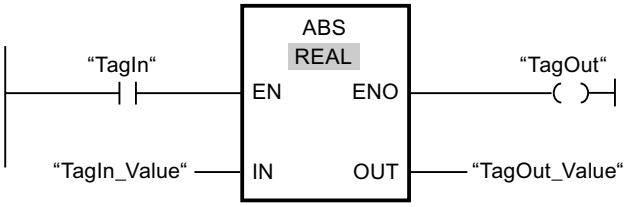
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P	Valor absoluto del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	-6.234
OUT	TagOut_Value	6.234

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor absoluto". La instrucción calcula el valor absoluto del valor de la entrada "TagIn_Value" y deposita el resultado en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas disponibles y escribe el valor menor en la salida OUT. El número de entradas se puede ampliar en el cuadro de la instrucción mediante entradas adicionales. Las entradas se numeran de forma ascendente en el cuadro.

Para la ejecución de la instrucción se deben indicar como mínimo dos valores de entrada y como máximo 100.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
INn	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Entradas insertadas adicionalmente cuyos valores se comparan

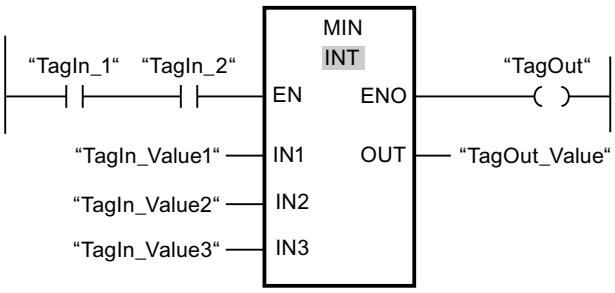
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
OUT	Output	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.					

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar mínimo". La instrucción compara los valores de los operandos indicados y copia el valor menor ("TagIn_Value1") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas disponibles y escribe el valor mayor en la salida OUT. El número de entradas se puede ampliar en el cuadro de la instrucción mediante entradas adicionales. Las entradas se numeran de forma ascendente en el cuadro.

Para la ejecución de la instrucción se deben indicar como mínimo dos valores de entrada y como máximo 100.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada

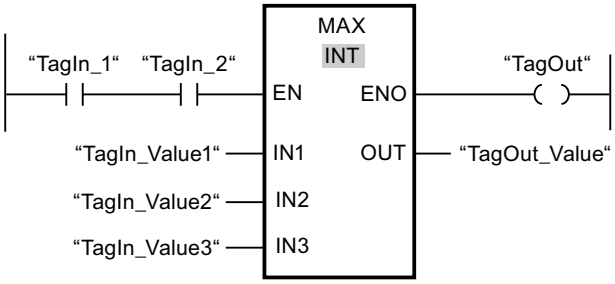
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
INn	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Entradas insertadas adicionalmente cuyos valores se comparan
OUT	Output	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.					

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar máximo". La instrucción compara los valores de los operandos indicados y copia el valor mayor ("TagIn_Value2") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor en la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no se define y la salida de habilitación ENO es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor de la entrada MN es mayor que el valor de la entrada MX.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación

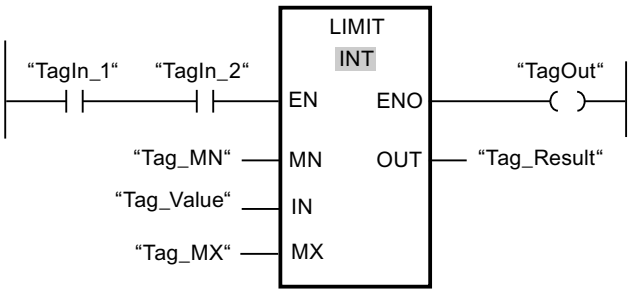
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
MN	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Límite inferior
IN	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Valor de entrada
MX	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Límite superior
OUT	Output	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.					

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se ejecuta la instrucción "Ajustar valor límite". El valor del operando "Tag_Value" se compara con los valores de los operandos "Tag_MN" y "Tag_MX". Puesto que el valor del operando "Tag_Value" es menor que el valor límite inferior, el valor del operando "Tag_MN" se copia en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" permite elevar al cuadrado el valor de la entrada IN y consultar el resultado en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular cuadrado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

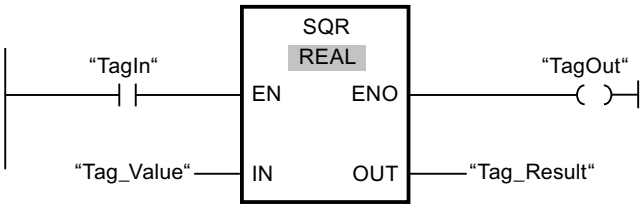
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Cuadrado del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular cuadrado". La instrucción eleva al cuadrado el valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" permite obtener la raíz cuadrada del valor de la entrada IN y consultar el resultado en la salida OUT. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido. Si el valor de la entrada IN es "0", el resultado también es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular raíz cuadrada":

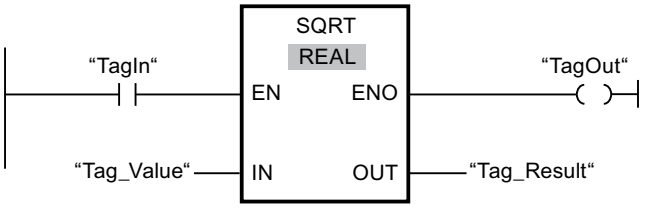
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Raíz cuadrada del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular raíz cuadrada". La instrucción extrae la raíz cuadrada del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2,718282) del valor de la entrada IN. El resultado se deposita en la salida OUT y se puede consultar allí. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular logaritmo natural":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

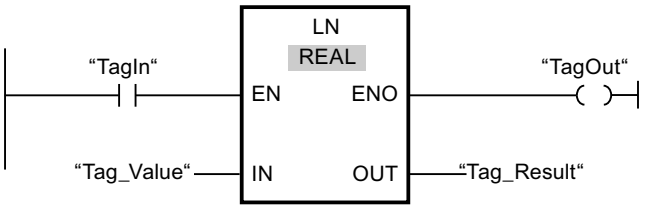
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Logaritmo natural del valor de la entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular logaritmo natural". La instrucción calcula el logaritmo natural del valor de la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2,718282) del valor indicado en la entrada IN. El resultado se emite y puede consultarse en la salida OUT (OUT = e^{IN}).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor exponencial":

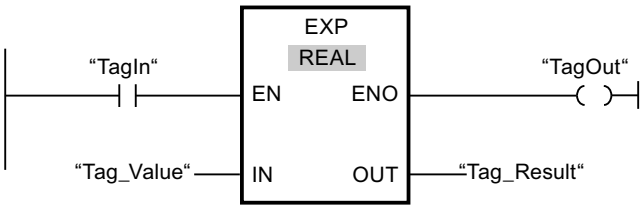
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Valor exponencial del valor de entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor exponencial". La instrucción calcula la potencia en base e del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno del ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de seno":

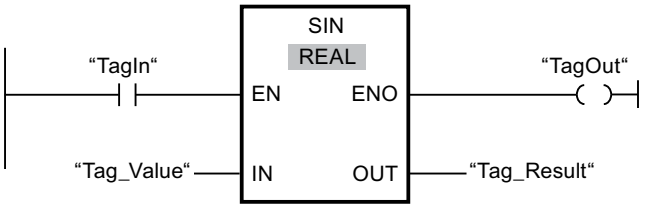
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Seno del ángulo indicado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1,570796 (π/2)
OUT	Tag_Result	1.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de seno". La instrucción calcula el seno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno del ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de coseno":

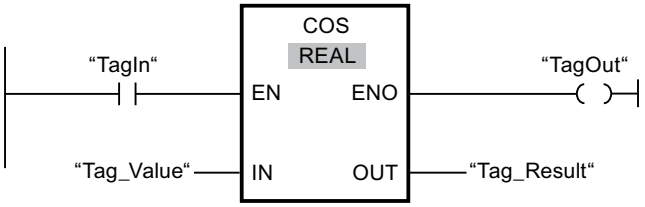
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Coseno del ángulo indicado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1,570796 (π/2)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de coseno". La instrucción calcula el coseno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente del ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de tangente":

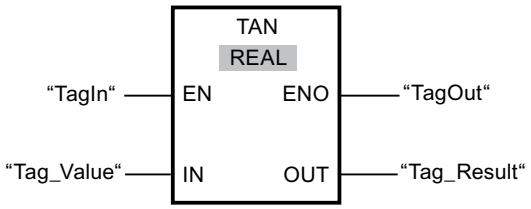
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tangente del ángulo indicado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+3,141593 (π)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de tangente". La instrucción calcula la tangente del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno indicado en la entrada IN el tamaño del ángulo que equivale a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcoseno":

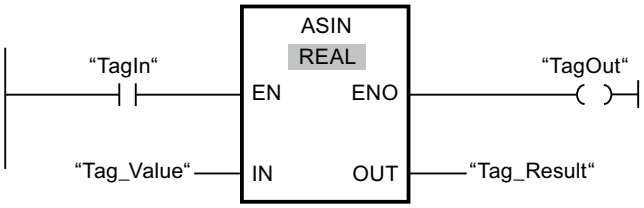
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de seno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+1,570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcoseno". La instrucción calcula el tamaño del ángulo que equivale al valor de seno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno indicado en la entrada IN el tamaño del ángulo que equivale a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre 0 y +π.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcocoseno":

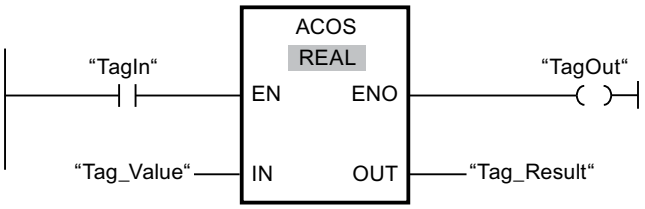
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de coseno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0
OUT	Tag_Result	+1,570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcocoseno". La instrucción calcula el tamaño del ángulo que equivale al valor de coseno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente indicado en la entrada IN el tamaño del ángulo que equivale a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcotangente":

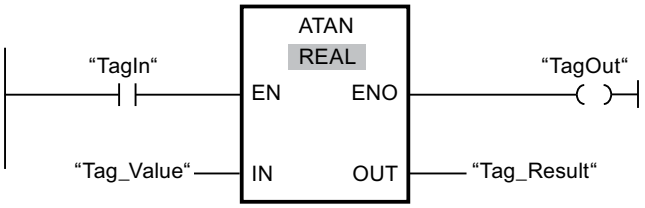
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de tangente
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+0,785398 (π/4)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcotangente". La instrucción calcula el tamaño del ángulo que equivale al valor de tangente de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

FRAC: Determinar decimales

Descripción

La instrucción "Determinar decimales" permite determinar los decimales del valor de la entrada IN. El resultado de la consulta se deposita en la salida OUT y se puede consultar allí. Si la entrada IN lleva, p. ej., el valor 123,4567, la salida OUT proporciona el valor 0,4567.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar decimales":

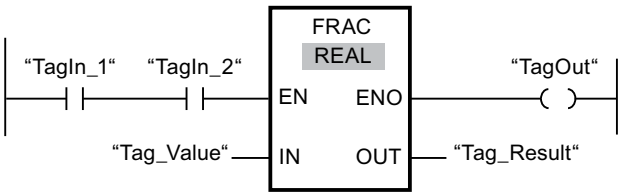
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor cuyos decimales se determinan.
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Decimales del valor de la entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	2.555
OUT	Tag_Result	0.555

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se inicia la instrucción "Determinar decimales". Los decimales del valor del operando "Tag_Value" se copian en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

EXPT: Elevar a potencia

Descripción

La instrucción "Elevar a potencia" permite elevar el valor de la entrada IN1 a la potencia del valor de la entrada IN2. El resultado de la operación se deposita en la salida OUT, donde se puede consultar (OUT = $IN1^{IN2}$).

A la entrada IN1 sólo se pueden asignar valores en coma flotante válidos. A la entrada IN2 también se pueden asignar números enteros.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Al ejecutar la instrucción se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Elevar a potencia":

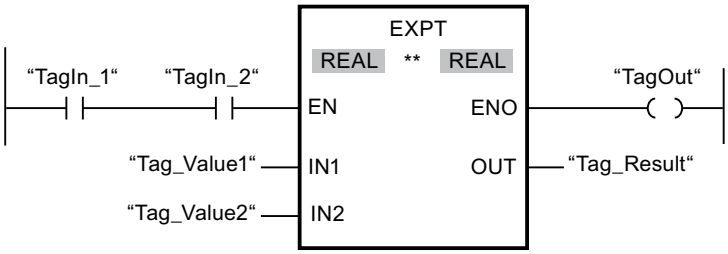
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor base
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Valor del exponente
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se inicia la instrucción "Elevar a potencia". El valor del operando "Tag_Value1" se eleva a la potencia del valor del operando "Tag_Value2". El resultado se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

Transferencia

MOVE: Copiar valor

Descripción

La instrucción "Copiar valor" transfiere el contenido del operando de la entrada IN al operando de la salida OUT1. La transferencia se efectúa siempre por orden ascendente de direcciones.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El tipo de datos del parámetro IN no puede convertirse al tipo de datos indicado en el parámetro OUT1.

La tabla siguiente muestra las posibles transferencias para la familia de CPU S7-1200:

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE ,TOD, CHAR
WORD	WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, REAL, TIME, DATE, TOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
USINT	USINT, UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
INT	INT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UINT	UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UDINT	UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LREAL
TIME	TIME	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME
DATE	DATE	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, DATE
TOD	TOD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, CHAR, carácter de una cadena[1)]
Carácter de una cadena[1)]	Carácter de una cadena	CHAR, carácter de una cadena
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_SCOUNTER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER	IEC_UCOUNTER

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
IEC_DCOUNTER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER	IEC_UDCOUNTER

La tabla siguiente muestra las posibles transferencias para la familia de CPU S7-1500:

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
BYTE	BYTE, WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE ,TOD, LTOD, CHAR
WORD	WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, S5TIME, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
DWORD	DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
LWORD	LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LREAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
USINT	USINT, UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
INT	INT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UINT	UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD. LTOD
DINT	DINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UDINT	UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
LINT	LINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
ULINT	ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LWORD, LREAL
S5TIME	S5TIME	WORD, S5TIME

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
TIME	TIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME
LTIME	LTIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTIME
DATE	DATE	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, DATE
DT	DT	DT
LDT	LDT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LDT
TOD	TOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TOD
LTOD	LTOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, LWORD, CHAR, carácter de una cadena[1)]
WCHAR	WCHAR	BYTE, WORD, DWORD, LWORD, CHAR, WCHAR, carácter de una cadena[1)]
Carácter de una cadena[1)]	Carácter de una cadena	CHAR, WCHAR, carácter de una cadena
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
COUNTER	COUNTER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	TIMER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_LTIMER	IEC_LTIMER	IEC_LTIMER
IEC_SCOUNTER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUNTER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER	IEC_UDCOUNTER
IEC_LCOUNTER	IEC_LCOUNTER	IEC_LCOUNTER
IEC_ULCOUNTER	IEC_ULCOUNTER	IEC_ULCOUNTER

[1)] La instrucción "Copiar valor" permite transferir también caracteres individuales de una cadena de caracteres a operandos del tipo de datos CHAR o WCHAR. El número del carácter que se va a transferir se indica entre corchetes al lado del nombre del operando. Si se indica

"MyString[2]", se transfiere p. ej. el segundo carácter de la cadena de caracteres "MyString". Asimismo, es posible transferir operandos del tipo de datos CHAR a caracteres individuales de una cadena de caracteres. También es posible reemplazar un carácter específico de una cadena de caracteres por el carácter de otra cadena de caracteres.

[2)] Sólo es posible transferir matrices enteras (ARRAY) si los elementos de matriz de los operandos de la entrada IN y de la salida OUT1 son del mismo tipo de datos.

Si la longitud de bits del tipo de datos de la entrada IN supera la longitud de bits del tipo de datos de la salida OUT1, se pierden los bits más significativos del valor de origen. Si la longitud de bit del tipo de datos en la entrada IN es inferior a la longitud de bits del tipo de datos en la salida OUT1, se sobrescriben con ceros los bits más significativos del valor de destino.

El cuadro de la instrucción contiene en estado inicial 1 salida (OUT1). El número de salidas es ampliable. Las salidas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se transfiere el contenido del operando de la entrada IN a todas las salidas disponibles. El cuadro de la instrucción no se puede ampliar si se transfieren tipos de datos estructurados (DTL, STRUCT, ARRAY) o caracteres de una cadena de caracteres.

Para copiar operandos del tipo de datos ARRAY pueden utilizarse también las instrucciones "Copiar área" (MOVE_BLK) y "Copiar área sin interrupciones" (UMOVE_BLK). Los operandos del tipo de datos STRING se pueden copiar con la instrucción "Desplazar cadena de caracteres" (S_MOVE).

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Copiar valor":

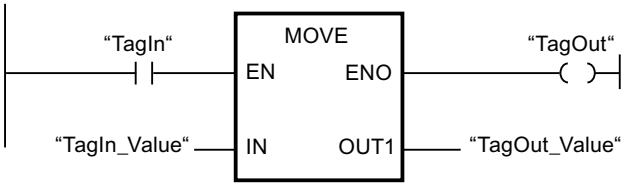
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	Secuencias de bits, enteros, números en coma flotante, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, tipos de datos IEC, tipos de datos PLC (UDT)	Secuencias de bits, enteros, números en coma flotante, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, tipos de datos IEC, tipos de datos PLC (UDT)	I, Q, M, D, L o constante	Valor de origen
OUT1	Output	Secuencias de bits, enteros, números en coma flotante, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, tipos de datos IEC, tipos de datos PLC (UDT)	Secuencias de bits, enteros, números en coma flotante, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, tipos de datos IEC, tipos de datos PLC (UDT)	I, Q, M, D, L	Operando al que se transfiere el valor de origen.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar valor". La instrucción copia los contenidos del operando "TagIn_Value" en el operando "TagOut_Value" y pone la salida "TagOut" al estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

MOVE_BLK: Copiar área (Página 3453)

UMOVE_BLK: Copiar área sin interrupciones (Página 3459)

S_MOVE: Desplazar cadena de caracteres (Página 4706)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

Deserialize: Deserializar

Descripción

Mediante la instrucción "Deserializar" se convierte la forma de representación secuencial de un tipo de datos PLC (UDT) de nuevo en un tipo de datos PLC y se rellena todo su contenido.

El área de memoria en la que se encuentra la forma de representación secuencial de un tipo de datos PLC debe ser del tipo de datos ARRAY of BYTE y haberse declarado con acceso estándar. Antes de la conversión, asegúrese de que hay suficiente espacio de memoria libre.

Esta instrucción permite convertir paso a paso a su estado inicial varias formas de representación secuenciales de tipos de datos PLC convertidos.

Nota

Para convertir de nuevo una forma de representación secuencial aislada de un tipo de datos PLC (UDT), puede utilizar directamente la instrucción "TRCV: Recibir datos a través de la conexión".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Deserializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC_ARRAY	Input	VARIANT	I, Q, D, L	Bloque de datos global en el que está guardado el flujo de datos generado
DEST_VARIABLE	InOut	VARIANT	I, Q, D, L	Variable en la que se guarda el tipo de datos PLC (UDT) reconvertido
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

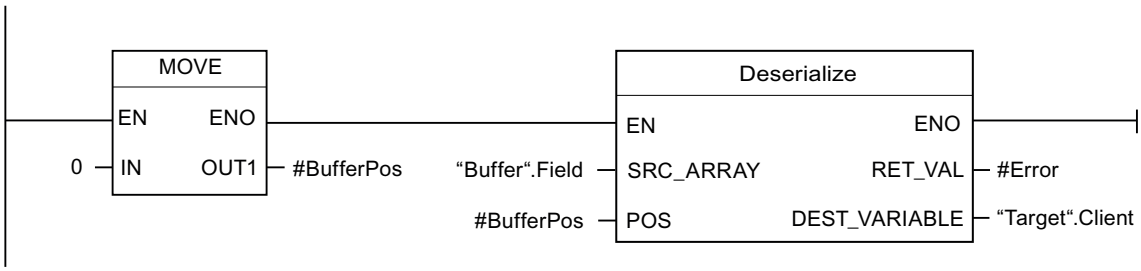
La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_ARRAY y DEST_VARIABLE se solapan.
8136	El bloque de datos del parámetro DEST_VARIABLE no es un bloque con acceso estándar.
8150	El tipo de datos VARIANT del parámetro SRC_ARRAY no contiene ningún valor.
8151	Error de creación de código en el parámetro SRC_ARRAY
8153	En el parámetro SRC_ARRAY no hay suficiente espacio de memoria libre.
8250	El tipo de datos VARIANT del parámetro DEST_VARIABLE no contiene ningún valor.
8251	Error de creación de código en el parámetro DEST_VARIABLE
8254	Tipo de datos no válido en el parámetro DEST_VARIABLE
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

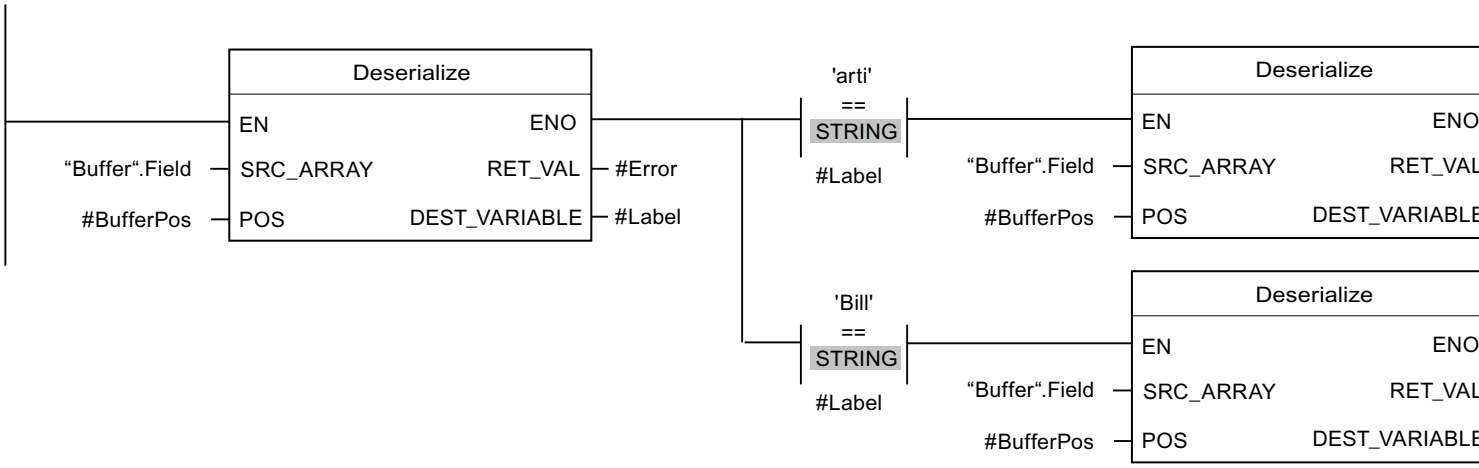
El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1:



La instrucción "Copiar valor " copia el valor "0" al operando "#BufferPos". La instrucción "Deserializar" deserializa la forma de representación secuencial de los datos de cliente del bloque de datos "Buffer" y los escribe en el bloque de datos "Target". El número de bytes que ocupan los datos de cliente convertidos se guarda en el operando "#BufferPos".

Segmento 2:



La instrucción "Deserializar" deserializa la forma de representación secuencial del separador (depositado después de los datos de cliente en la forma de representación secuencial) del bloque de datos "Buffer" y escribe los caracteres en el operando "#Label". Los caracteres se comparan en relación con "arti" y "Bill" mediante instrucciones de comparación. Si la comparación en relación con "arti" = TRUE, se trata de datos de artículo deserializados que se escriben en el bloque de datos "Target". Si la comparación en relación con "Bill" = TRUE, se trata de datos de facturas deserializados que se escriben en el bloque de datos "Target".

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Target	Client	"Client" (tipo de datos PLC)
	Article	Array[0..] of "Article" (tipo de datos PLC)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

Serialize: Serializar

Descripción

Mediante la instrucción "Serializar" se convierten varios tipos de datos PLC (UDT) a una forma de representación secuencial sin que se pierdan partes de su estructura.

La instrucción permite guardar temporalmente varios datos estructurados del programa en un búfer, ubicado preferentemente en un bloque de datos global, y enviarlos a otra CPU. El área de memoria en la que se depositen los tipos de datos PLC convertidos debe ser del tipo de datos ARRAY of BYTE y estar declarada con acceso estándar. Antes de la conversión, asegúrese de que hay suficiente espacio de memoria libre.

El operando del parámetro POS contiene la información del número de bytes que ocupan los tipos de datos PLC convertidos.

Nota

Para enviar un solo tipo de datos PLC (UDT), puede llamar directamente la instrucción "TSEND: Enviar datos a través de la conexión".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Serializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC_VARIABLE	Input	VARIANT	I, Q, D, L	Tipo de datos PLC (UDT) que se convierte a una forma de representación secuencial.
DEST_ARRAY	InOut	VARIANT	I, Q, D, L	Bloque de datos en el que se guarda el flujo de datos generado.
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

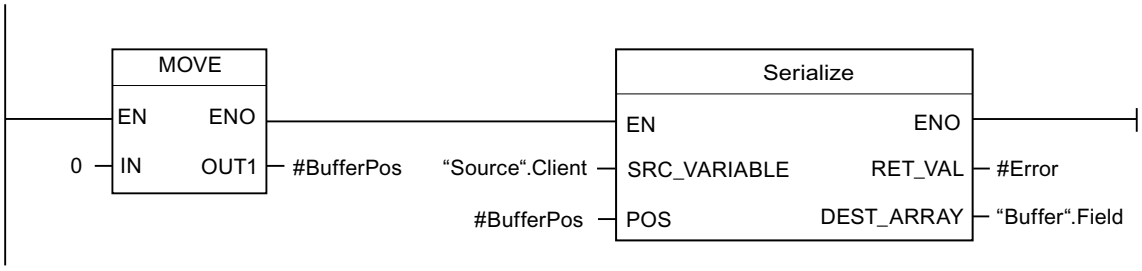
Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_VARIABLE y DEST_ARRAY se solapan.
8150	El tipo de datos VARIANT del parámetro SRC_VARIABLE no contiene ningún valor.
8152	Error de creación de código en el parámetro SRC_VARIABLE
8236	El bloque de datos del parámetro DEST_ARRAY no es un bloque con acceso estándar.
8250	El tipo de datos VARIANT del parámetro DEST_ARRAY no contiene ningún valor.
8252	Error de creación de código en el parámetro DEST_ARRAY
8253	En el parámetro DEST_ARRAY no hay suficiente espacio de memoria libre.
8254	Tipo de datos no válido en el parámetro DEST_ARRAY

Código de error* (W#16#...)	Explicación
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

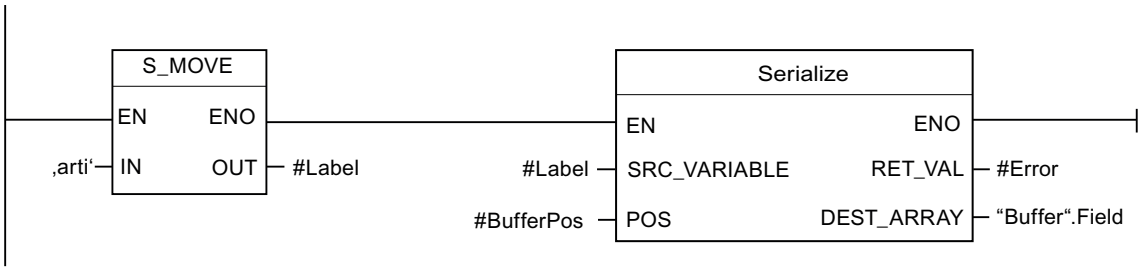
El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1:



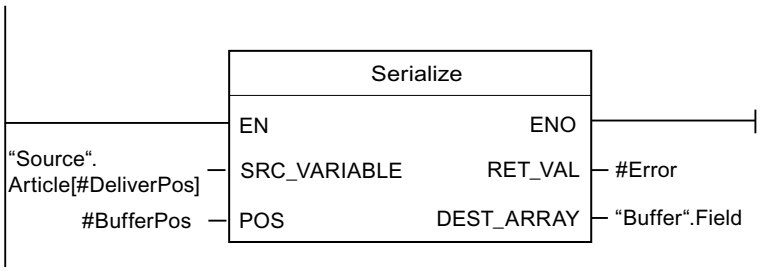
La instrucción "Copiar valor " copia el valor "0" al operando "#BufferPos". La instrucción "Serializar" serializa los datos de cliente del bloque de datos "Source" y los escribe como representación secuencial en el bloque de datos "Buffer". El número de bytes que ocupa la representación secuencial se guarda en el operando "#BufferPos".

Segmento 2:



Para facilitar la posterior deserialización de la forma de representación secuencial, ahora se inserta una especie de separador. La instrucción "Desplazar cadena de caracteres" copia los caracteres "arti" al operando "#Label". La instrucción "Serializar" escribe estos caracteres después de los datos de cliente en el bloque de datos "Buffer". El número de bytes que necesitan los caracteres se suma en el operando "#BufferPos" al número guardado anteriormente.

Segmento 3:



La instrucción "Serializar" escribe los datos de un determinado artículo, que se calcula en tiempo de ejecución, del bloque de datos "Source", y los escribe como representación secuencial después de los caracteres "arti" en el bloque de datos "Buffer".

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Source	Client	"Client" (tipo de datos PLC)
	Article	Array[0..10] of "Article" (tipo de datos PLC)
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

FieldRead: Leer campo

Descripción

La instrucción "Leer campo" permite leer un elemento determinado del campo indicado en la entrada MEMBER y transferir su contenido a la variable de la salida VALUE. El índice del elemento de campo que se lee se define en la entrada INDEX. En la entrada MEMBER se indica el primer elemento del campo que se lee.

Los tipos de datos del elemento de campo indicado en la entrada MEMBER y de la variable de la salida VALUE deben coincidir con el tipo de datos de la instrucción "Leer campo".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El elemento de campo indicado en la entrada INDEX no está definido en el campo especificado en la entrada MEMBER.
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer campo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
INDEX	Input	DINT	DINT	I, Q, M, D, L, P o constante	Índice del elemento cuyo contenido se lee.

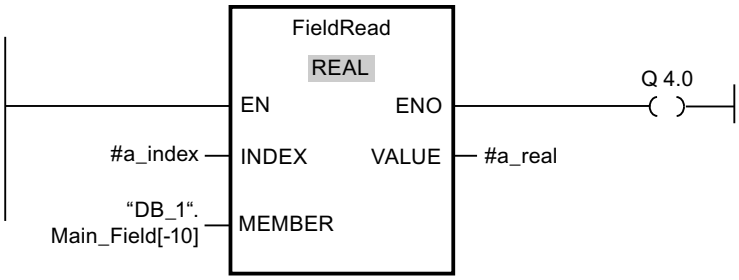
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
MEMBER	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD y CHAR como elementos de una variable ARRAY	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR como elementos de una variable ARRAY	D, L	Primer elemento del campo que se lee.
VALUE	Output	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD y CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR	I, Q, M, D, L, P	Operando al que se transfiere el contenido del elemento.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Variable	Valor
INDEX	a_index	4
MEMBER	"DB_1".Main_Field[-10]	Primer elemento del campo "Main_Field[-10..10] of REAL" en el bloque de datos "DB_1"
VALUE	a_real	Elemento con índice 4 del campo "Main_Field[-10..10] of REAL"

El elemento con índice 4 se lee del campo "Main_Field[-10...10] of REAL" y se escribe en la variable "a_real". El elemento de campo que se lee está definido por el valor de la entrada INDEX.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

FieldWrite: Escribir campo

Descripción

La instrucción "Escribir campo" transfiere el contenido de la variable de la entrada VALUE a un elemento determinado del campo especificado en la salida MEMBER. El índice del elemento que se escribe se define con el valor de la entrada INDEX. En la salida MEMBER se indica el primer elemento del campo en el que se escribe.

Los tipos de datos del elemento de campo especificado en la salida MEMBER y de la variable de la entrada VALUE deben coincidir con el tipo de datos de la instrucción "Escribir campo".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El elemento indicado en la entrada INDEX no está definido en el campo especificado en la salida MEMBER.
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir campo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
INDEX	Input	DINT	DINT	I, Q, M, D, L, P o constante	Índice del elemento en el que se escribe el contenido de VALUE.

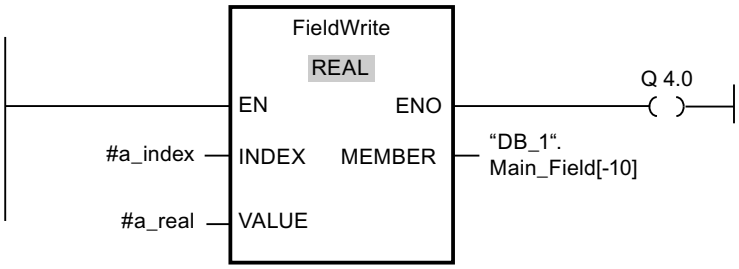
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
VALUE	Input	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD y CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR	I, Q, M, D, L, P o constante	Operando cuyo contenido se copia.
MEMBER	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD y CHAR como elementos de una variable ARRAY	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR como elementos de una variable ARRAY	D, L	Primer elemento del campo en el que se escribe el contenido de VALUE.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
INDEX	a_index	4
VALUE	a_real	10.54
MEMBER	"DB_1".Main_Field[-10]	Primer elemento del campo "Main_Field[-10..10] of REAL" en el bloque de datos "DB_1"

El valor "10.54" de la variable "a_real" se escribe en el elemento de campo con el índice 4 del campo "Main_Field[-10..10] of REAL". El índice del elemento de campo al que se transfiere el contenido de la variable "a_real" está definido por el valor de la entrada INDEX.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

MOVE_BLK: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con la entrada COUNT. El ancho del elemento de la entrada IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura ARRAY. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura ARRAY, la salida de habilitación ENO se pone a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

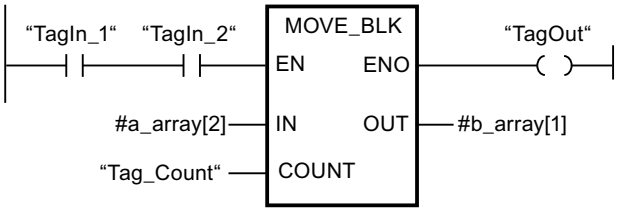
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino.
OUT [1]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.					

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Copiar área". La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

MOVE_BLK_VARIANT: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). Se pueden copiar elementos de un array en otro array del mismo tipo de datos. Los tamaños (número de elementos) de los arrays de origen y de destino pueden ser diferentes. Se pueden copiar varios elementos dentro de un array o bien copiar los elementos uno a uno.

En el momento de utilizar la instrucción para crear un bloque, aún no tiene que conocerse el array, puesto que el origen y el destino se transfieren por VARIANT.

El contaje de los parámetros SRC_INDEX y DEST_INDEX siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el puntero VARIANT (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC	Input	VARIANT (array o elemento individual)	L	El área de origen de la que se copia
COUNT	Input	UDINT	I, Q, M, D, L o constante	Número de elementos que se copian Asigne al parámetro COUNT el valor "1" si en el parámetro SRC o en el parámetro DEST no hay indicado ningún Array.
SRC_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro SRC_INDEX se calcula basado en cero. Si en el parámetro SRC hay indicado un Array, el entero del parámetro SRC_INDEX indica el primer elemento del área de origen de la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro SRC no hay indicado ningún Array o solo hay indicado un elemento individual de un Array, asigne al parámetro SRC_INDEX el valor "0".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DEST_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro DEST_INDEX se calcula basado en cero. Si en el parámetro DEST hay indicado un Array, el entero del parámetro DEST_INDEX indica el primer elemento del área de destino en la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro DEST no hay indicado ningún Array, asigne al parámetro DEST_INDEX el valor "0".
DEST	Output	VARIANT	L	Área de destino a la que se copia el contenido del área de origen.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

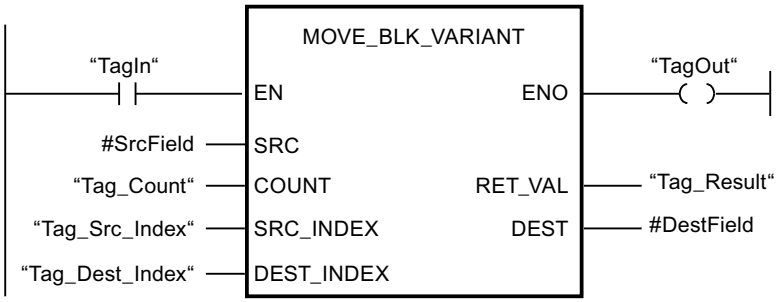
La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
8151	No es posible acceder al parámetro SRC.
8152	El operando del parámetro SRC no está tipificado.
8153	Error de creación de código en el parámetro SRC
8154	El operando del parámetro SRC es del tipo de datos BOOL.

Código de error* (W#16#...)	Explicación
8281	Valor no válido en el parámetro COUNT
8382	El valor del parámetro SRC_INDEX se encuentra fuera de los valores límite de VARIANT.
8383	El valor del parámetro SRC_INDEX se encuentra fuera del límite superior del array.
8482	El valor del parámetro DEST_INDEX se encuentra fuera de los valores límite de VARIANT.
8483	El valor del parámetro DEST_INDEX se encuentra fuera del límite superior del array.
8534	El parámetro DEST está protegido contra escritura
8551	No es posible acceder al parámetro DEST.
8552	El operando del parámetro DEST no está tipificado.
8553	Error de creación de código en el parámetro DEST
8554	El operando del parámetro DEST es del tipo de datos BOOL.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC	#SrcField	El operando local #SrcField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [0..10] of "MOVE_UDT"
COUNT	Tag_Count	2
SRC_INDEX	Tag_Src_Index	3

Parámetro	Operando	Valor
DEST_INDEX	Tag_Dest_Index	3
DEST	#DestField	El operando local #DestField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [10..20] of "MOVE_UDT"

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar área". En el área de destino se copian 2 elementos del área de origen, empezando por el cuarto elemento del array de UDT. Las copias se insertan en el Array of UDT a partir del cuarto elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VariantGet: Leer valor de una variable VARIANT (Página 3480)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

UMOVE_BLK: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia sin interrupciones los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento de la entrada IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Copiar área sin interrupciones".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura ARRAY. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura ARRAY, la salida de habilitación ENO se pone a "0".

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Parámetros

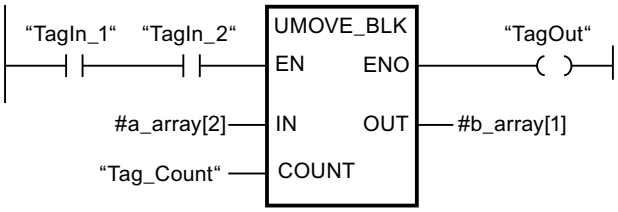
La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino.
OUT [1]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.					

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Copiar área sin interrupciones". La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Si no se producen errores al ejecutar la instrucción, la salida ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

FILL_BLK: Rellenar área

Descripción

La instrucción "Rellenar área" permite rellenar un área de memoria (área de destino) con el valor de la entrada IN. El área de destino se rellena a partir de la dirección indicada en la salida OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al

ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura ARRAY. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura ARRAY, la salida de habilitación ENO se pone a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

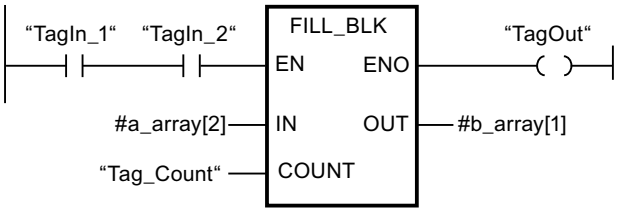
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	Dirección del área de destino a partir de la cual se rellena

[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY.
[2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Rellenar área". La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. Si no se producen errores al ejecutar la instrucción, las salidas de habilitación ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

UFILL_BLK: Rellenar área sin interrupciones

Descripción

La instrucción "Rellenar área sin interrupciones" rellena sin interrupciones un área de memoria (área de destino) con el valor de la entrada IN. El área de destino se rellena a partir de la

dirección indicada en la salida OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Rellenar área sin interrupciones".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura ARRAY. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura ARRAY, la salida de habilitación ENO se pone a "0".

La instrucción "Rellenar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área sin interrupciones":

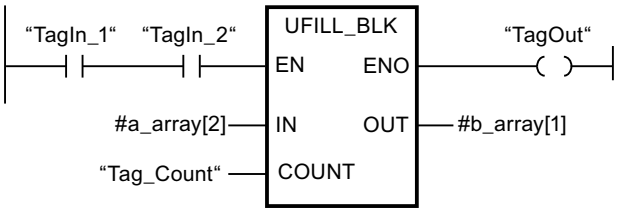
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN [1)]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino.
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de repeticiones de copia

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
OUT [2]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	Dirección del área de destino a partir de la cual se rellena.
[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY. [2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.					

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Rellenar área sin interrupciones". La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Si no se producen errores al ejecutar la instrucción, las salidas de habilitación ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

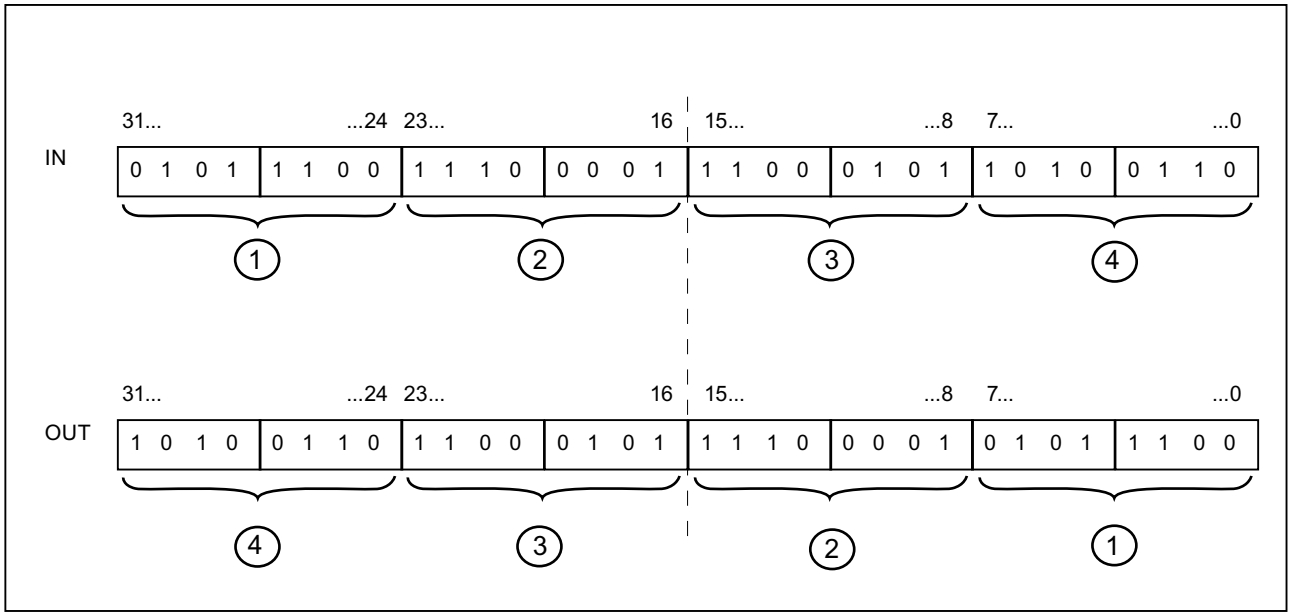
Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

SWAP: Cambiar disposición

Descripción

La instrucción "Cambiar disposición" permite cambiar la disposición de los bytes de la entrada IN y consultar el resultado en la salida OUT.

La figura siguiente muestra cómo se intercambian los bytes de un operando del tipo de datos DWORD utilizando la instrucción "Cambiar disposición":



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cambiar disposición":

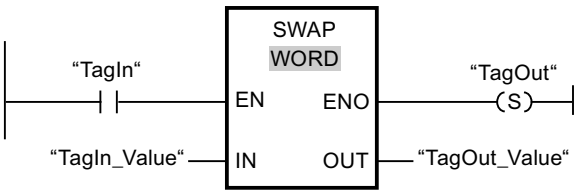
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD, DWORD	WORD, DWORD, LWORD	I, Q, M, D, L, P o constante	Operando cuyos bytes se intercambian.
OUT	Output	WORD, DWORD	WORD, DWORD, LWORD	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0000 1111 0101 0101
OUT	TagOut_Value	0101 0101 0000 1111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Cambiar disposición". La disposición de los bytes se cambia y se deposita en el operando "TagOut_Value".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Instrucciones DB de ARRAY

ReadFromArrayDB: Leer desde bloque de datos de array

Descripción

La instrucción "Leer en un bloque de datos ARRAY" permite leer datos en un DB de array y escribirlos en un área de destino.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer en un bloque de datos ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
VALUE	Output	VARIANT	L	Valor que se lee y se devuelve
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

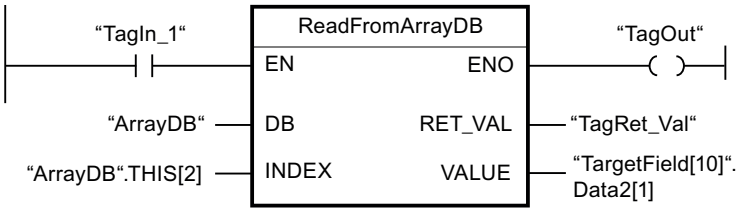
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El DB de array contiene valores no válidos
8154	El tipo de datos no está declarado correctamente
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	TargetField[10].Data2[1]	El operando "TargetField" es del tipo de datos Array [10..20] of "MOVE_UDT".

Si el operando "TagIn1" devuelve el estado lógico "1", se ejecuta la instrucción "Leer en un bloque de datos ARRAY". El tercer elemento se lee en el "ArrayDB" y se escribe en el operando "TargetField[10].Data2[1]". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

WriteToArrayDB: Escribir en bloque de datos de array

Descripción

La instrucción "Escribir en bloque de datos de array" permite escribir datos en un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en bloque de datos de array":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	L	Valor que se escribe
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

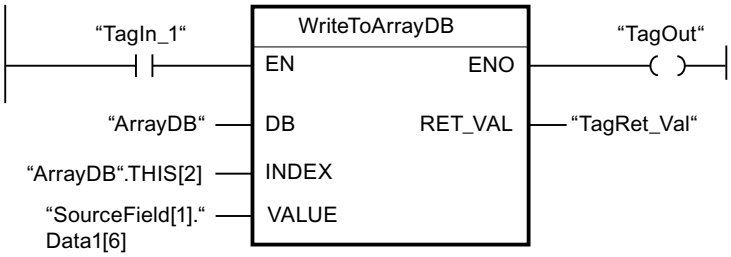
La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga

Código de error* (W#16#...)	Explicación
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetros	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceField[1].Data1[6]	El operando "SourceField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Si el operando "TagIn1" devuelve el estado lógico "1", se ejecuta la instrucción "Escribir en un bloque de datos ARRAY". Del operando "SourceField" se escribe el elemento "Data1[6]" del segundo elemento en el "ArrayDB". En el "ArrayDB" se escribe el tercer elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

ReadFromArrayDBL: Leer de bloque de datos de array de la memoria de carga

Descripción

La instrucción "Leer de bloque de datos de array de la memoria de carga" permite leer datos desde un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar sólo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción. Durante un ciclo de programa, el parámetro DONE tiene el estado lógico "1" y el valor leído en el parámetro VALUE se emite dentro de dicho ciclo. En los demás ciclos del programa, el valor del parámetro VALUE no se modifica.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer de bloque de datos de array de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la lectura del DB de array
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La lectura del DB de array todavía no ha finalizado
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
VALUE	InOut	VARIANT	D, L	Valor que se lee y se devuelve
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

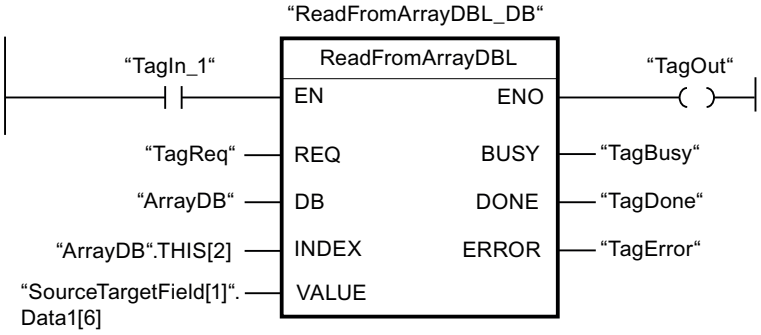
Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas

Código de error* (W#16#...)	Explicación
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Parámetro	Operando	Valor
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si el operando "TagIn1" devuelve el estado lógico "1" y en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga". El tercer elemento se lee en el "ArrayDB" y se indica en el operando "SourceTargetField[1].Data1[6]". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE ya no se modifica. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut". Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

WriteToArrayDBL: Escribir en bloque de datos de array de la memoria de carga

Descripción

La instrucción "Escribir en bloque de datos de array de la memoria de carga" permite escribir datos en un DB de array de la memoria de carga.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar sólo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción y el valor del parámetro VALUE se escribe en el bloque de datos. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se

depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en bloque de datos de array de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la escritura en el DB de array
DB	Input	DB_ANY	D	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	D, L	Valor que se escribe
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La escritura en el DB de array todavía no ha finalizado

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

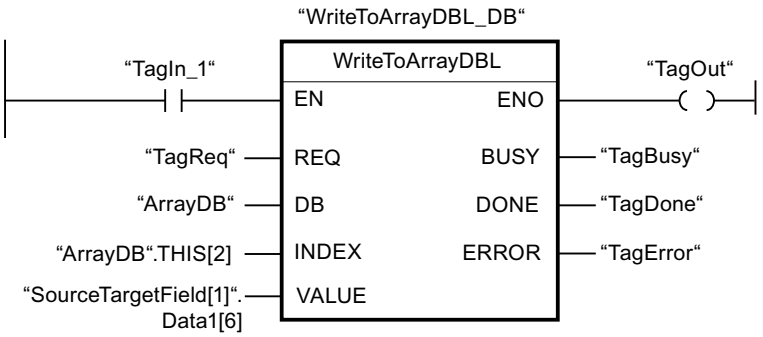
La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si el operando "TagIn1" devuelve el estado lógico "1" y en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE se escribe en el "ArrayDB" en el tercer elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut". Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

Instrucciones VARIANT

VariantGet: Leer valor de una variable VARIANT

Descripción

Con la instrucción "Leer valor de una variable VARIANT" se lee el valor de la variable a la que apunta VARIANT en el parámetro SRC y se escribe este valor en la variable del parámetro DST.

El parámetro SRC tiene que ser del tipo de datos VARIANT. En el parámetro DST se puede indicar cualquier tipo de datos excepto VARIANT.

El tipo de datos de la variable del parámetro DST debe coincidir con el tipo de datos al que apunta VARIANT.

Nota

La variable a la que señala VARIANT en el parámetro SRC, debe ser un tipo de datos PLC.

Para copiar estructuras y ARRAY también se puede utilizar la instrucción "MOVE_BLK_VARIANT: Copiar área". Encontrará más información al respecto en "Consulte también".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Los tipos de datos no coinciden. (No se transmiten valores.)

Parámetro

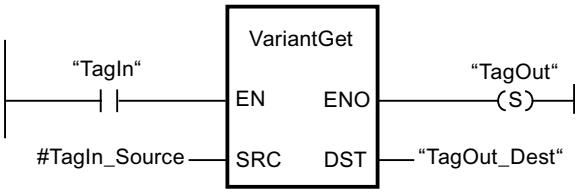
La tabla siguiente muestra los parámetros de la instrucción "Leer valor de una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC	Input	VARIANT	L	Variable que se lee
DST	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Leer valor de una variable VARIANT". El valor de la variable al que apunta VARIANT en el operando "#TagIn_Source" se lee y se escribe en el operando "TagOut_Dest".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

MOVE_BLK_VARIANT: Copiar área (Página 3455)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

VariantPut: Escribir valor en una variable VARIANT

Descripción

Mediante la instrucción "Escribir valor en una variable VARIANT" se escribe el valor de la variable del parámetro SRC en la variable del parámetro DST al que apunta VARIANT.

El parámetro DST tiene que ser del tipo de datos VARIANT. En el parámetro SRC se puede indicar cualquier tipo de datos excepto VARIANT.

El tipo de datos de la variable del parámetro SRC debe coincidir con el tipo de datos al que apunta VARIANT.

Nota

La variable a la que señala VARIANT en el parámetro DST, debe ser un tipo de datos PLC.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Los tipos de datos no coinciden. (No se transmiten valores.)

Parámetro

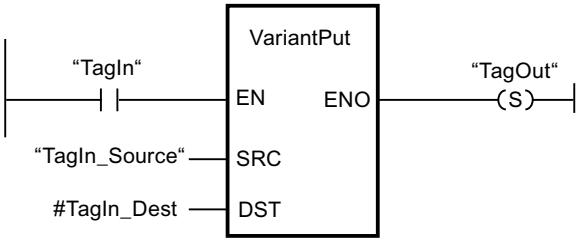
La tabla siguiente muestra los parámetros de la instrucción "Escribir valor en una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY	I, Q, M, D, L	Variable que se lee
DST	Input	VARIANT	L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Escribir valor en una variable VARIANT". El valor del operando "TagIn_Source" se escribe en la variable a la que apunta VARIANT en el operando "#TagIn_Dest".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

CountOfElements: Consultar número de elementos de ARRAY

Descripción

La instrucción "Consultar número de elementos de ARRAY" permite saber el número de elementos de ARRAY que tiene una variable a la que apunta VARIANT.

Si el ARRAY es unidimensional, el resultado se devuelve como diferencia entre el límite superior e inferior + 1. Si el ARRAY es multidimensional, el resultado se devuelve como producto de todas las dimensiones.

El parámetro IN tiene que ser del tipo de datos VARIANT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- La variable VARIANT no es un ARRAY. (El resultado es "0".)

Si VARIANT apunta a un ARRAY of BOOL, se cuentan también los elementos de relleno. (P. ej., con un ARRAY[0..1] of BOOL se devuelve 8.)

Parámetro

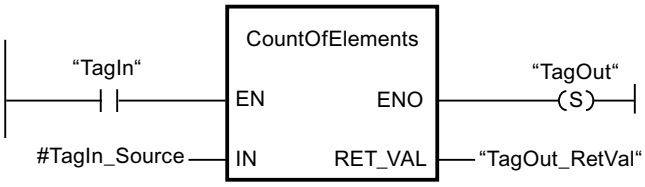
La tabla siguiente muestra los parámetros de la instrucción "Consultar número de elementos de ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	VARIANT	L	Variable que se consulta
RET_VAL	Output	UDINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Consultar número de elementos de ARRAY". El número de elementos de ARRAY de la variable a la que apunta VARIANT en el operando "#TagIn_Source" se lee y se devuelve en el operando "TagOut_RetVal".

Consulte también

Formato de Array (límites de 16 bits) (Página 2354)

Formato de Array (límites de 32 bits) (Página 2355)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Cablear operandos en instrucciones KOP (Página 2701)

Otras

BLKMOV: Copiar área

Descripción

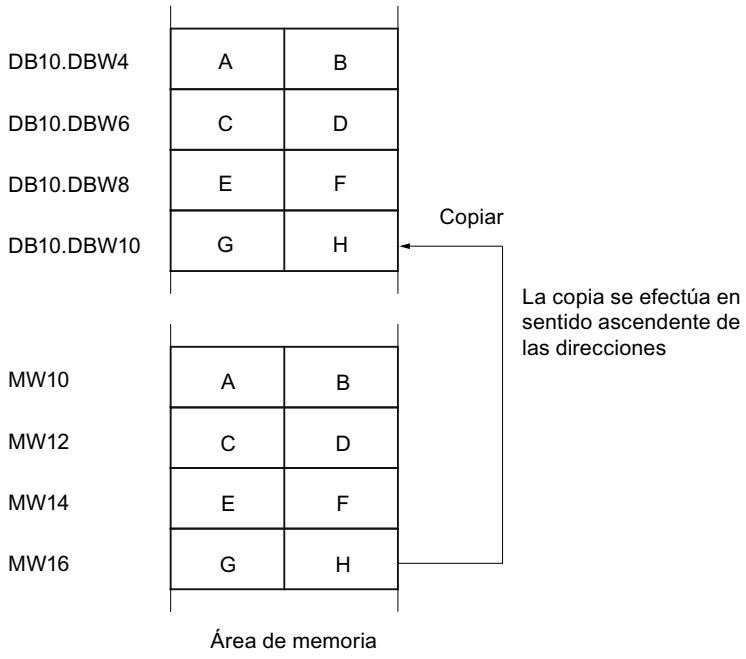
La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza

por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

No hay limitaciones en cuanto a la profundidad de anidamiento.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas

- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	VARIANT	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8092	El área de origen o destino está solo en la memoria de carga.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK , no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	VARIANT	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

FILL: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones.

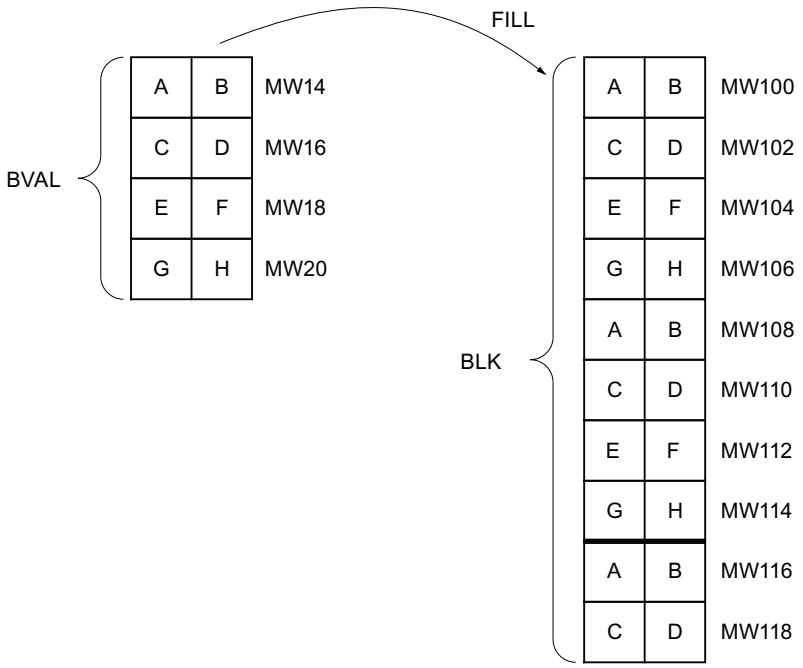
Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Para los bloques con el atributo "Acceso optimizado al bloque" puede utilizarse la instrucción "FILL_BLK: Rellenar área".

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: el contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
BVAL	Input	VARIANT	L, P	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
BLK	Output	VARIANT	L, P	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" lee el contenido del parámetro IN y lo convierte según los tipos de datos seleccionados en el cuadro de la instrucción. El valor convertido se deposita en la salida OUT.

Encontrará información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir valor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros, números en coma flotante, CHAR, BCD16, BCD32	I, Q, M, D, L, P o constante	Valor que se convierte.
OUT	Output	Secuencias de bits, enteros, números en coma flotante, CHAR, BCD16, BCD32	I, Q, M, D, L, P	Resultado de la conversión

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

En el cuadro de la instrucción no se pueden seleccionar secuencias de bits (BYTE, WORD, DWORD, LWORD). Si se especifica un operando del tipo de datos BYTE, WORD, DWORD o LWORD en un parámetro de la instrucción, el operando se interpretará conforme al tipo de

datos del parámetro de entrada o salida. El tipo de datos BYTE se interpreta entonces como USINT, WORD como UINT, DWORD como UDINT y LWORD como LINT.

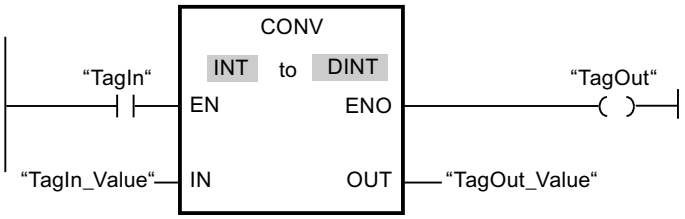
Nota

Para CPU S7-1500 rige: Pueden seleccionarse los tipos de datos DWORD y LWORD si se ha elegido REAL o LREAL como tipo de datos IN.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", el contenido del operando "TagIn_Value" se lee y se convierte en un número entero (32 bits). El resultado se deposita en el operando "TagOut_Value". Si la instrucción se ha ejecutado correctamente, la salida "TagOut" se pone a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Conversión explícita (Página 2481)

Conversión explícita de CHAR (Página 2505)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor de la entrada IN al siguiente número entero. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte a un número entero del tipo de datos DINT. Si el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear número":

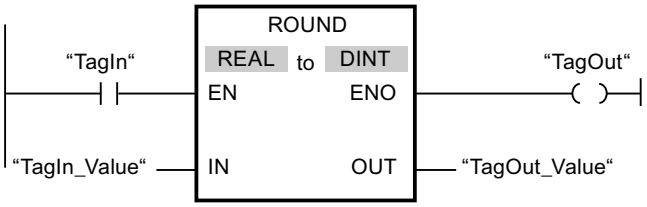
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada que se debe redondear.
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado del redondeo

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear número". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero par más próximo y se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Editar elementos KOP (Página 2694)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor de la entrada IN al siguiente número entero superior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero superior. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí. El valor de salida puede ser mayor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero superior":

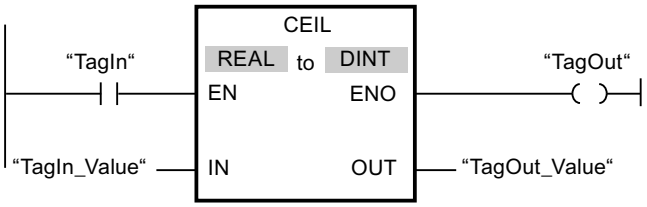
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero superior

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero superior". El número en coma flotante de la entrada "TagIn_Value" se redondea al siguiente número entero superior y se devuelve en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Editar elementos KOP (Página 2694)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor de la entrada IN al siguiente número entero inferior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero inferior. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí. El valor de salida puede ser menor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero inferior":

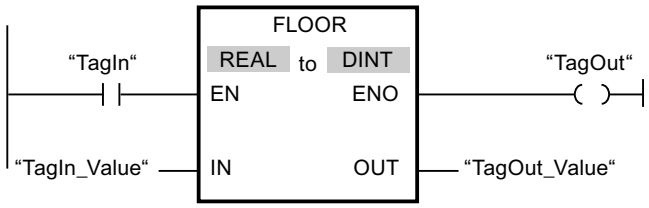
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero inferior

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero inferior". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero inferior más próximo y se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un valor entero a partir del valor de la entrada IN. El valor de la entrada IN se interpreta como número en coma flotante. La instrucción selecciona sólo la parte entera del número en coma flotante y la deposita sin decimales en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Truncar a entero":

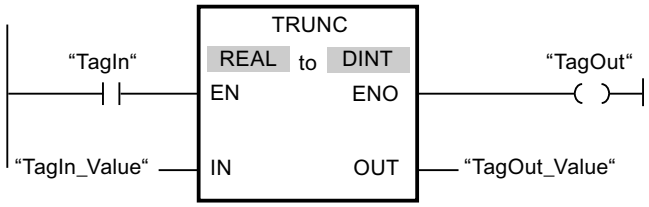
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L	Parte entera del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Truncar a entero". La parte entera del número en coma flotante de la entrada "TagIn_Value" se convierte en un número entero y se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

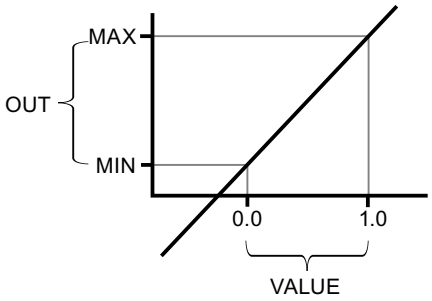
Seleccionar el tipo de datos de un elemento KOP (Página 2682)

SCALE_X: Escalar

Descripción

La instrucción "Escalar" escala el valor de la entrada VALUE mapeándolo en un determinado rango de valores. Al ejecutar la instrucción "Escalar", el número en coma flotante de la entrada VALUE se escala al rango de valores definido por los parámetros MIN y MAX. El resultado de la escala es un número entero que se deposita en la salida OUT.

La figura siguiente muestra un ejemplo de cómo pueden escalarse los valores:



La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [VALUE * (MAX – MIN)] + MIN

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada MIN es mayor o igual al valor de la entrada MAX.
- El valor de un número en coma flotante indicado está fuera del rango de los números normalizados según IEEE-754.
- Ocurre un rebase por exceso.
- El valor de la entrada VALUE es NaN (Not a number = resultado de una operación aritmética no válida).

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE	Input	Números en coma flotante	I, Q, M, D, L o constante	Valor que se escala. Si se indica una constante, esta debe declararse.
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L	Resultado de la escala

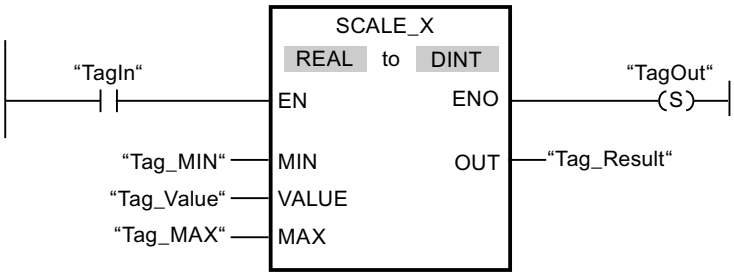
En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_MIN	10
VALUE	Tag_Value	0.5
MAX	Tag_MAX	30
OUT	Tag_Result	20

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Escalar". El valor de la entrada "Tag_Value" se escala al rango de valores definido por los valores de las entradas "Tag_MIN" y "Tag_MAX". El resultado se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

NORM_X: Normalizar (Página 3503)

Declarar constantes globales (Página 2584)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

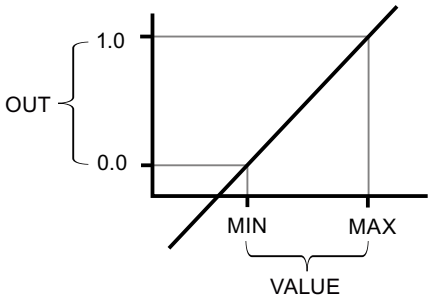
Seleccionar el tipo de datos de un elemento KOP (Página 2682)

NORM_X: Normalizar

Descripción

La instrucción "Normalizar" normaliza el valor de la variable de la entrada VALUE representándolo en una escala lineal. Los parámetros MIN y MAX sirven para definir los límites de un rango de valores que se refleja en la escala. En función de la posición del valor que se debe normalizar en este rango de valores, se calcula el resultado y se deposita como número en coma flotante en la salida OUT. Si el valor que se debe normalizar es igual al valor de la entrada MIN, la salida OUT devuelve el valor "0.0". Si el valor que se debe normalizar es igual al valor de la entrada MAX, la salida OUT devuelve el valor "1.0".

La figura siguiente muestra un ejemplo de cómo pueden normalizarse los valores:



La instrucción "Normalizar" utiliza la siguiente ecuación:

OUT = (VALUE – MIN) / (MAX – MIN)

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada MIN es mayor o igual al valor de la entrada MAX.
- El valor de un número en coma flotante indicado está fuera del rango de los números normalizados según IEEE-754.
- El valor de la entrada VALUE es NaN (resultado de una operación aritmética no válida).

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Normalizar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MIN [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Valor que se normaliza.
MAX [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
OUT	Output	Números en coma flotante	I, Q, M, D, L	Resultado de la normalización
[1)] Si se utilizan constantes en estos tres parámetros, basta con declarar una de ellas.				

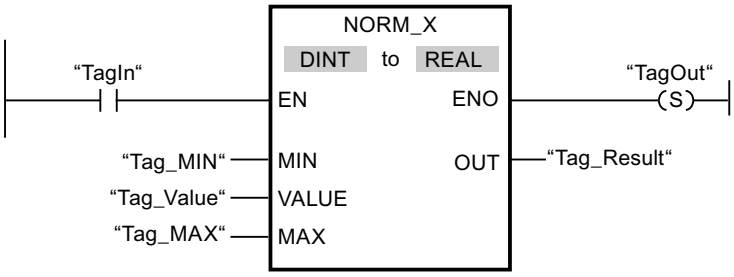
En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_MIN	10
VALUE	Tag_Value	20
MAX	Tag_MAX	30
OUT	Tag_Result	0.5

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Normalizar". El valor de la entrada "Tag_Value" se asigna al rango de valores definido por los valores de las entradas "Tag_MIN" y "Tag_MAX". El valor de la variable de la entrada "Tag_Value" se normaliza conforme al rango de valores definido. El resultado se deposita como número en coma flotante en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

SCALE_X: Escalar (Página 3500)

Declarar constantes globales (Página 2584)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Otras

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala.
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

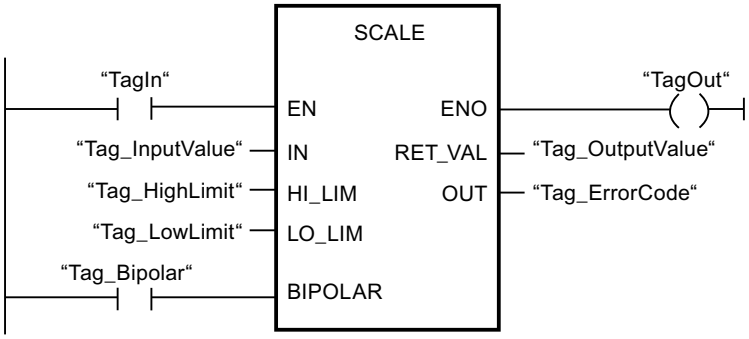
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1"
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" permite desescalar el número en coma flotante del parámetro IN en unidades físicas comprendidas entre un límite inferior y uno superior y convertirlo en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Input	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero.
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

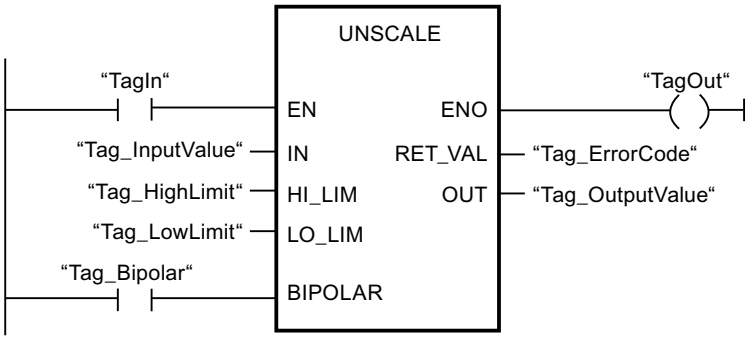
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Control del programa

---(JMP): Saltar si RLO = 1

Descripción

La instrucción "Saltar si RLO = 1" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente. El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima de la instrucción.

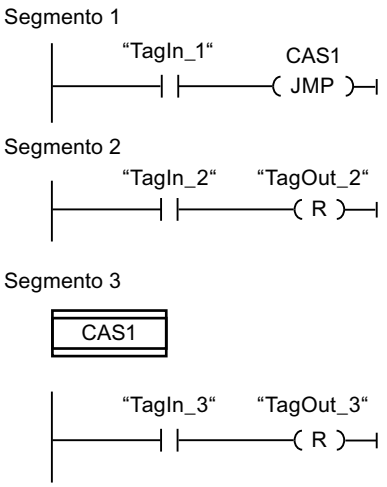
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En cada segmento no debe existir más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si no se cumple la condición en la entrada de la instrucción (RLO = 0), la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(JMPN): Saltar si RLO = 0

Descripción

La instrucción "Saltar si RLO = 0" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente si el resultado lógico de la entrada de la instrucción es "0". El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima de la instrucción.

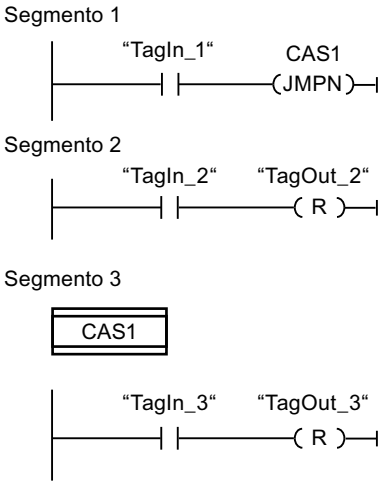
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En cada segmento no debe existir más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "0", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si el resultado lógico de la entrada de la instrucción es "1", la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "0", se ejecuta la instrucción "Saltar si RLO = 0". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones KOP (Página 2701)

LABEL: Etiqueta

Descripción

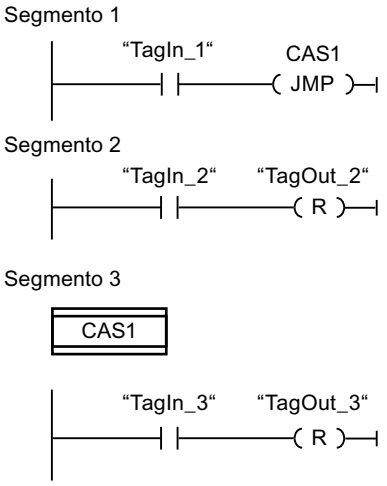
Una etiqueta sirve para marcar el segmento de destino en el que debe continuar la ejecución del programa tras un salto.

La etiqueta y la instrucción en la que se indica la etiqueta como destino del salto deben encontrarse en el mismo bloque. La denominación de una etiqueta debe ser unívoca en el bloque. Puede declarar un máximo de 32 etiquetas en caso de utilizar una CPU S7-1200 y un máximo de 256 etiquetas en caso de utilizar una CPU S7-1500.

En un segmento sólo se puede colocar una etiqueta. A toda etiqueta se puede acceder desde distintas posiciones.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

JMP_LIST: Definir lista de saltos

Descripción

La instrucción "Definir lista de saltos" permite definir varios saltos condicionados y continuar la ejecución del programa en un segmento determinado en función del valor del parámetro K.

Los saltos se definen mediante etiquetas (LABEL) que se indican en las salidas del cuadro de la instrucción. El número de salidas del cuadro de la instrucción se puede ampliar. Puede declarar un máximo de 32 salidas en caso de utilizar una CPU S7-1200 y un máximo de 99 salidas en caso de utilizar una CPU S7-1500.

La numeración de las salidas comienza por el valor "0" y continúa en orden ascendente con cada nueva salida. En las salidas de la instrucción únicamente se pueden indicar etiquetas. No está permitido indicar instrucciones u operandos.

Con el valor del parámetro K se indica el número de la salida y a la vez la etiqueta en la que debe continuarse la ejecución del programa. Si el valor del parámetro K es mayor que el número de salidas disponibles, la ejecución del programa continúa en el siguiente segmento del bloque.

La instrucción "Definir lista de saltos" se ejecuta solo si el estado lógico de la entrada de habilitación EN es "1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Definir lista de saltos":

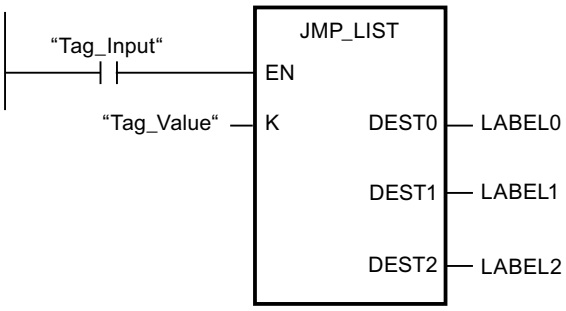
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
K	Input	UINT	I, Q, M, D, L o constante	Indica el número de la salida y a la vez el salto que se ejecuta.
DEST0	-	-	-	Primera etiqueta

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DEST1	-	-	-	Segunda etiqueta
DESTn	-	-	-	Etiquetas opcionales

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando/etiqueta	Valor
K	Tag_Value	1
Dest0	LABEL0	Salto al segmento que está marcado con la etiqueta "LABEL0".
Dest1	LABEL1	Salto al segmento que está marcado con la etiqueta "LABEL1".
Dest2	LABEL2	Salto al segmento que está marcado con la etiqueta "LABEL2".

Si el operando "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Definir lista de saltos". La ejecución del programa continúa conforme al valor del operando "Tag_Value" en el segmento que está marcado con la etiqueta "LABEL1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

SWITCH: Distribuidor de saltos

Descripción

La instrucción "Distribuidor de saltos" permite definir varios saltos de programa que se ejecutan en función del resultado de una o varias instrucciones de comparación.

El valor que se va a comparar se especifica en el parámetro K. Este valor se compara con los valores que devuelven las distintas entradas. El tipo de comparación se selecciona individualmente para cada entrada. La disponibilidad de las diferentes instrucciones de comparación depende del tipo de datos de la instrucción.

La tabla siguiente muestra las instrucciones de comparación disponibles en función del tipo de datos elegido:

Tipo de datos		Instrucción	Sintaxis
S7-1200	S7-1500		
Secuencias de bits	Secuencias de bits	Igual	==
		Diferente	<>
Enteros, números en coma flotante, TIME, DATE, TOD	Enteros, números en coma flotante, TIME, LTIME, DATE, TOD, LTOD, LDT	Igual	==
		Diferente	<>
		Mayor o igual	>=
		Menor o igual	<=
		Mayor	>
		Menor	<

El tipo de datos de la instrucción se selecciona en la lista desplegable "<???>" del cuadro de la instrucción. Si selecciona una instrucción de comparación sin que esté definido el tipo de datos de la instrucción, en la lista desplegable "<???>" sólo aparecen los tipos de datos permitidos para la instrucción de comparación seleccionada.

La ejecución de la instrucción comienza por la primera comparación y se sigue ejecutando hasta que se cumpla una condición de comparación. Cuando se cumple una condición de comparación, no se tienen en cuenta las siguientes condiciones de comparación. Si no se cumple ninguna de las condiciones de comparación indicadas, el salto se ejecuta en la salida ELSE. Si en la salida ELSE no hay definido ningún salto de programa, la ejecución del programa continúa en el segmento siguiente.

El número de salidas del cuadro de la instrucción se puede ampliar. La numeración de las salidas comienza por el valor "0" y continúa en orden ascendente con cada nueva salida. Indique etiquetas (LABEL) en las salidas de la instrucción. No está permitido indicar instrucciones u operandos en las salidas de la instrucción.

Por cada salida adicional se inserta automáticamente una entrada. El salto programado en una salida se ejecuta cuando se cumple la condición de comparación de la entrada correspondiente.

Parámetros

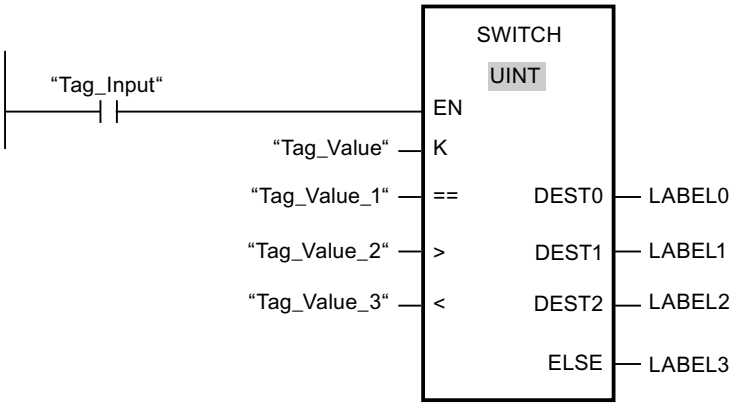
La tabla siguiente muestra los parámetros de la instrucción "Distribuidor de saltos":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
K	Input	UINT	UINT	I, Q, M, D, L o constante	Indica el valor que se compara.
<Valores de comparación>	Input	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD	Secuencias de bits, enteros, números en coma flotante, TIME, LTIME, DATE, TOD, LTOD, LDT	I, Q, M, D, L o constante	Valores de entrada con los que se compara el valor del parámetro K.
DEST0	-	-	-	-	Primera etiqueta
DEST1	-	-	-	-	Segunda etiqueta
DEST(n)	-	-	-	-	Etiquetas opcionales (n = 2 hasta 99)
ELSE	-	-	-	-	Salto de programa que se ejecuta cuando no se cumple ninguna de las condiciones de comparación.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando/etiqueta	Valor
K	Tag_Value	23
==	Tag_Value_1	20
>	Tag_Value_2	21
<	Tag_Value_3	19
Dest 0	LABEL0	Salto a la etiqueta "LABEL0" si el valor del parámetro K es igual a 20.
Dest 1	LABEL1	Salto a la etiqueta "LABEL1" si el valor del parámetro K es mayor que 21.
Dest 2	LABEL2	Salto a la etiqueta "LABEL2" si el valor del parámetro K es menor que 19.
ELSE	LABEL 3	Salto a la etiqueta "LABEL3" si no se cumple ninguna de las condiciones de comparación.

Si el operando "Tag_Input" cambia al estado lógico "1", se ejecuta la instrucción "Distribuidor de saltos". La ejecución del programa continúa en el segmento que está marcado con la etiqueta "LABEL1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

--(RET): Retroceder

Descripción

La instrucción "Retroceder" permite finalizar la ejecución de un bloque. De este modo se obtienen tres formas posibles de terminar la ejecución de un bloque:

- Sin llamada de la instrucción "Retroceder"
  Tras ejecutar el último segmento, se abandona el bloque. El ENO de la función de llamada adopta el estado lógico "1".
- Llamada de la instrucción "Retroceder" con operación lógica precedente (ver ejemplo)
  Si la conexión izquierda tiene el estado lógico "1", se abandona el bloque. El ENO de la función de llamada equivale al operando.
- Llamada de la instrucción "Retroceder" sin operación lógica precedente
  Se abandona el bloque. El ENO de la función de llamada equivale al operando.

Nota

En un segmento no puede utilizarse más de una bobina de salto ("Retroceder", "Saltar si RLO = 1", "Saltar si RLO = 0").

Si el resultado lógico (RLO) de la entrada de la instrucción "Retroceder" es "1", se finaliza la ejecución del programa en el bloque llamado actualmente y se continúa en el bloque que efectúa la llamada (p. ej. en el OB invocante) después de la función de llamada. El parámetro de la instrucción determina el estado (ENO) de la función de llamada. Puede adoptar los valores siguientes:

- RLO
- TRUE/FALSE
- <Operando>

Haga doble clic en la instrucción para ajustar los valores del parámetro y seleccione el valor correspondiente en la lista desplegable.

La tabla siguiente muestra el estado de la función de llamada cuando la instrucción "Retroceder" está programada en un segmento del bloque llamado:

RLO	Valor del parámetro	ENO de la función de llamada
1	RLO	1
	TRUE	1
	FALSE	0
	<Operando>	<Operando>
0	RLO	La ejecución del programa continúa en el siguiente segmento del bloque llamado.
	TRUE	
	FALSE	
	<Operando>	

Cuando se finaliza un OB, el sistema de niveles secuenciales selecciona otro bloque para luego iniciarlo o seguir ejecutándolo:

- Cuando se finaliza el OB de ciclo de programa, éste se reinicia.
- Cuando se finaliza un OB que ha interrumpido otro bloque (p. ej. un OB de alarma), se ejecuta el bloque interrumpido (p. ej. OB de ciclo de programa).

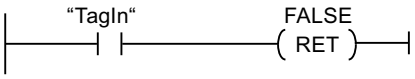
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Retroceder":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Estado de la función de llamada si RLO = 1:				
RLO	-	-		Adopta el estado lógico del RLO.
TRUE	-	-		1
FALSE	-	-		0
<Operando>	Input	BOOL	I, Q, M, D, L	Estado lógico del operando indicado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico del operando "TagIn" es "1", se ejecuta la instrucción "Retroceder". La ejecución del programa se finaliza en el bloque llamado y continúa en el bloque que efectúa la llamada. La salida de habilitación ENO de la función de llamada adopta el estado lógico "0".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones KOP (Página 2701)

Control del tiempo de ejecución

ENDIS_PW: Limitar y habilitar legitimación de la contraseña

Descripción

La instrucción "Limitar y habilitar legitimación de la contraseña" permite definir si las contraseñas configuradas para la CPU están legitimadas o no. Con ello, se pueden impedir conexiones legitimadas, aun conociendo la contraseña correcta.

Si la instrucción se ejecuta y el parámetro REQ tiene el estado lógico "0", en los parámetros de salida se muestra el estado ajustado en ese momento. Si se han realizado cambios en los parámetros de entrada, estos no se transfieren a los parámetros de salida.

Si se ejecuta la instrucción y el parámetro REQ tiene el estado lógico "1", se aplica el estado lógico de los parámetros de entrada (F_PWD, FULL_PWD, R_PWD, HMI_PWD). FALSE significa que la legitimación por contraseña no está permitida y TRUE significa que la contraseña puede utilizarse.

Es posible permitir o prohibir individualmente el bloqueo o la habilitación de las contraseñas. Por ejemplo, pueden prohibirse todas las contraseñas excepto la contraseña de seguridad positiva. Esto permite limitar las posibilidades de acceso a un reducido grupo de usuarios. Los parámetros de salida (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) siempre muestran el estado actual del uso de contraseñas, independientemente del parámetro REQ.

Las contraseñas no configuradas deben tener en la entrada el estado lógico TRUE y devolver el estado lógico TRUE en la salida. La contraseña de seguridad positiva (fail-safe) solo se puede parametrizar para una CPU F y por eso en una CPU estándar debe interconectarse siempre con el estado lógico TRUE. Si la instrucción devuelve un error, la llamada permanece sin efecto, es decir, el bloqueo anterior sigue en vigor.

Las contraseñas bloqueadas pueden volver a habilitarse en las condiciones siguientes:

- Se ha restablecido la configuración de fábrica de la CPU.
- El panel frontal de la CPU S7-1500 soporta un cuadro de diálogo que permite navegar al menú adecuado y volver a habilitar las contraseñas.
- Cuando se llama la instrucción "Limitar y habilitar legitimación de la contraseña", el parámetro de entrada de la contraseña deseada tiene el estado lógico "1".
- Poner el selector de modo a STOP. La restricción de la legitimación de contraseña se activa de nuevo en cuanto se mueve el selector nuevamente a RUN.
- La inserción de una tarjeta de memoria vacía (tarjeta de transferencia o tarjeta de programa) en una CPU S7-1200.
- La transición de POWER OFF-POWER ON desactiva la protección en la CPU S7-1200. La instrucción "Limitar y habilitar legitimación de la contraseña" debe volverse a llamar en el programa (por ejemplo en el OB de arranque).

Nota

Si la contraseña HMI no está habilitada, la instrucción "Limitar y habilitar legitimación de la contraseña" bloquea el acceso de los sistemas HMI.

Nota

Las conexiones ya existentes y legitimadas conservan sus derechos de acceso y no pueden limitarse con la instrucción "Limitar y habilitar legitimación de la contraseña".

Impedir el bloqueo no intencionado en una CPU S7-1500

Los ajustes pueden realizarse en el panel frontal de la CPU S7-1500, que guarda el último ajuste en cada caso.

Para impedir un bloqueo no intencionado, en una CPU S7-1500 es posible desactivar la protección moviendo el selector de modo a STOP. La protección se activa automáticamente después de mover el selector de modo a RUN sin que tenga que volver a llamarse la instrucción "Limitar y habilitar legitimación de la contraseña" o deban realizarse otras acciones en el panel frontal.

Impedir el bloqueo no intencionado en una CPU S7-1200

Como la CPU S7-1200 no dispone de selector de modo, la protección se desactiva con POWER OFF-POWER ON. De este modo es posible y recomendable impedir el bloqueo no intencionado con ayuda de determinadas secuencias en el programa.

Para ello, programe un control de tiempo, ya sea mediante un OB de alarma cíclica o un temporizador en el Main OB (OB 1). De este modo, tras una transición de POWER OFF-POWER ON y la consiguiente desactivación de la protección, se puede volver a llamar rápidamente la instrucción "Limitar y habilitar legitimación de la contraseña" en el OB correspondiente (por ejemplo OB 1 u OB 35). Para que la ventana temporal en la que la instrucción no está activa y en la que por tanto no existen limitaciones en la legitimación de la contraseña sea lo más pequeña posible, llame la instrucción en el OB de arranque (OB 100). Este procedimiento ofrece la mayor protección posible contra accesos no autorizados.

Si se ha producido un bloqueo no intencionado puede omitirse la llamada en el OB de arranque (por ejemplo mediante la consulta de un parámetro de entrada) y se dispondrá del tiempo configurado (por ejemplo de 10 segundos a 1 minuto) para establecer una conexión con la CPU antes de que el bloqueo vuelva a activarse.

Si no se ha previsto ningún temporizador en el código del programa y se produce un bloqueo, inserte en la CPU una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna de la CPU. Después deberá cargarse nuevamente el programa de usuario de STEP 7 en la CPU.

Procedimiento en caso de pérdida de contraseña en una CPU S7-1200

Si ha perdido la contraseña de una CPU S7-1200 protegida por contraseña, borre el programa protegido por contraseña con una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna

de la CPU. A continuación, puede transferir un nuevo programa de usuario de STEP 7 Basic a la CPU.

⚠ ADVERTENCIA

Inserción de una tarjeta de transferencia vacía

Cuando se inserta una tarjeta de transferencia en una CPU en funcionamiento, la CPU pasa a STOP. En estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización que puede provocar lesiones mortales o graves, así como daños materiales.

El contenido de la tarjeta de transferencia se encuentra en la memoria de carga interna después de su extracción. Compruebe aquí que la tarjeta no contenga ningún programa.

⚠ ADVERTENCIA

Inserción de una tarjeta de programa vacía

Cuando se inserta una tarjeta de programa en una CPU en funcionamiento, la CPU pasa a STOP. En estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización que puede provocar lesiones mortales o graves, así como daños materiales.

Asegúrese de que la tarjeta de programa esté vacía. La memoria de carga interna se copiará en la tarjeta de programa vacía. Después de retirar la tarjeta de programa previamente vacía, la memoria de carga interna quedará vacía.

Debe retirarse la tarjeta de transferencia o la tarjeta de programa antes de poner la CPU en RUN.

Efectos del uso de contraseñas en los modos de operación

La siguiente tabla muestra qué efectos tiene el uso de contraseñas en los modos de operación y en las respectivas acciones del usuario mediante la instrucción "Limitar y habilitar legitimación de la contraseña".

Acción	Protección por contraseña mediante la instrucción
Estado básico después de • Selector de modo en STOP • Restablecimiento manual de la memoria (PG, interruptor, cambio de MC (Motion Control)) • Restablecimiento de la configuración de fábrica	No activado (sin limitaciones)
Estado básico tras conexión (POWER ON)	• CPU S7-1200: El bloqueo está desactivado y la instrucción tiene que volverse a ejecutar en el programa (por ejemplo en el OB de arranque). • CPU S7-1500: Activada (si antes de la desconexión (POWER OFF) había un bloqueo activo) La posibilidad de no permitir contraseñas es remanente.
Transición del estado operativo RUN/ARRANQUE/ PARADA -> STOP (mediante finalización de la instrucción, un error o comunicación) o STOP -> ARRANQUE/RUN/PARADA	Activado Las contraseñas siguen sin poder utilizarse.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Limitar y habilitar legitimación de la contraseña":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L o constante	Si el parámetro REQ tiene el estado lógico "0", se consulta el estado lógico ajustado actualmente para las contraseñas.
F_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura incluido failsafe • F_PWD = "0": No permitir contraseña • F_PWD = "1": Permitir contraseña
FULL_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura • FULL_PWD = "0": No permitir contraseña • FULL_PWD = "1": Permitir contraseña
R_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura • R_PWD = "0": No permitir contraseña • R_PWD = "1": Permitir contraseña

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
HMI_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso a la HMI • HMI_PWD = "0": No permitir contraseña • HMI_PWD = "1": Permitir contraseña
F_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura incluido failsafe • F_PWD_ON = "0": Contraseña no permitida • F_PWD_ON = "1": Contraseña permitida
FULL_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura • FULL_PWD_ON = "0": Contraseña no permitida • FULL_PWD_ON = "1": Contraseña permitida
R_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura • R_PWD_ON = "0": Contraseña no permitida • R_PWD_ON = "1": Contraseña permitida
HMI_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso a la HMI • HMI_PWD_ON = "0": Contraseña no permitida • HMI_PWD_ON = "1": Contraseña permitida
RET_VAL	Output	WORD	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8090	No se soporta la instrucción "Limitar y habilitar legitimación de la contraseña".
80D0	La contraseña de seguridad (fail-safe) no está configurada. En las CPU estándar, el estado lógico tiene que ser TRUE.
80D1	El acceso de lectura/escritura no está configurado
80D2	El acceso de lectura no está configurado
80D3	El acceso HMI no está configurado
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del tiempo de ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar en todos los bloques independientemente de la prioridad.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar hasta 10 veces en un ciclo de programa.

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros.

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", la CPU pasa al estado operativo STOP y finaliza la ejecución del programa. El estado lógico de la salida de la instrucción no se evalúa.

Si el RLO a la entrada de la instrucción es "0", la instrucción no se ejecuta.

Parámetros

La instrucción "Finalizar programa" no tiene parámetros.

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones KOP (Página 2701)

GET_ERROR: Consultar error localmente

Descripción

La instrucción "Consultar error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si el sistema notifica errores mientras se ejecuta el bloque, en el operando de la salida ERROR se guarda información detallada acerca del primer error de ejecución ocurrido desde la última vez que se ejecutó la instrucción.

En la salida ERROR solo pueden especificarse operandos del tipo de datos de sistema "ErrorStruct". El tipo de datos de sistema "ErrorStruct" especifica la estructura exacta con la que debe guardarse la información acerca del error que ha ocurrido. Con ayuda de otras instrucciones se puede evaluar esta estructura y programar la reacción correspondiente. Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces emite información sobre el siguiente error ocurrido.

Nota

La salida ERROR se modifica solo si existe información de error. Para poner a "0" la salida después del tratamiento de errores, dispone de las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz del bloque.
- Ponga a "0" la variable antes de llamar la instrucción.
- Consulte la salida de habilitación ENO.

La salida de habilitación ENO de la instrucción "Consultar error localmente" se activa únicamente si la entrada de habilitación EN devuelve el estado lógico "1" y existe información de error. Si no se cumple alguna de estas condiciones, la ejecución ulterior del programa no se ve afectada por la instrucción "Consultar error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar error localmente" activa el tratamiento local de errores en el bloque. Si en la lógica de un bloque se ha insertado "Consultar error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	ErrorStruct	D, L	Información de error

Tipo de datos "ErrorStruct"

La tabla siguiente muestra la estructura del tipo de datos "ErrorStruct":

Componente de estructura		Tipo de datos	Descripción
ERROR_ID		WORD	ID de error
FLAGS		BYTE	Muestra si el error se ha producido durante la llamada de un bloque. 16#01: Error durante la llamada de un bloque 16#00: No hay errores durante la llamada de un bloque
REACTION		BYTE	Reacción predeterminada: 0: ignorar (error de escritura) 1: continuar con el valor sustitutivo "0" (error de lectura) 2: omitir instrucción (error de sistema)
CODE_ADDRESS		CREF	Información sobre la dirección y el tipo del bloque
	BLOCK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error: 1: OB 2: FC 3: FB
	CB_NUMBER	UINT	Número del bloque lógico
	OFFSET	UDINT	Referencia a la memoria interna
MODE		BYTE	Información sobre la dirección de un operando

Componente de estructura		Tipo de datos	Descripción
OPERAND_NUMBER		UINT	Número de operando del comando de la máquina
POINTER_NUMBER_LOCATION		UINT	(A) Puntero interno
SLOT_NUMBER_SCOPE		UINT	(B) Área de almacenamiento en la memoria interna
DATA_ADDRESS		NREF	Información sobre la dirección de un operando
	AREA	BYTE	(C) Área de memoria: L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE I: 16#81 Q: 16#82 M: 16#83 DB: 16#84, 85, 8A, 8B Infracciones de área en una variable editable directamente del tipo de datos DINT: 16#04
	DB_NUMBER	UINT	(D) Número del bloque de datos
	OFFSET	UDINT	(E) Dirección relativa del operando

Componente de estructura "ERROR_ID"

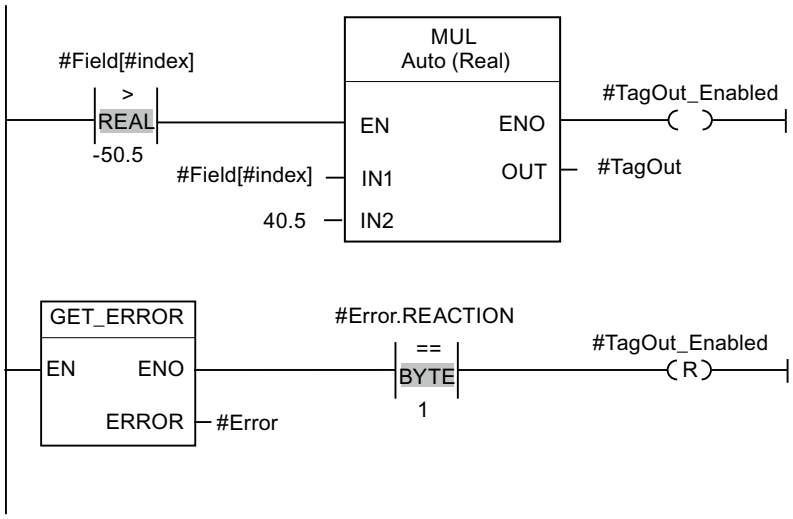
La tabla siguiente muestra los valores que pueden devolverse en el componente de estructura "ERROR_ID":

ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe

ID* (hexadecimal)	ID* (decimal)	Descripción
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
2576	9590	Error en la distribución de datos locales
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Se ha producido un error al acceder a la variable "#Field[#index]". La salida de habilitación ENO de la instrucción "Multiplicar" y el operando "#TagOut_Enabled" devuelven el estado lógico "1" a pesar del error de lectura/escritura, y la multiplicación se realiza con el valor "0.0". En este caso de error, es recomendable programar la instrucción "Consultar error localmente" después de la instrucción "Multiplicar" para interceptar el error. La información de error que devuelve la instrucción "Consultar error localmente" se evalúa mediante la instrucción de comparación "Igual". Si el componente de estructura "#Error.REACTION" tiene el valor "1", se trata de un error de lectura/escritura y la salida "#TagOut_Enabled" se pone a cero.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tratar errores de ejecución del programa (Página 2932)

Consultar y eliminar errores en el código del programa (Página 265)

GET_ERR_ID: Consultar ID de error localmente

Descripción

La instrucción "Consultar ID de error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si al ejecutar el bloque el sistema notifica errores de ejecución desde que se ejecutó la instrucción por última vez, la ID de error del primer error ocurrido se guarda en la variable de la salida ID.

En la salida ID solo pueden especificarse operandos del tipo de datos "WORD". Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces la instrucción emite la ID de error del siguiente error ocurrido.

Nota

La salida ID se modifica solo si existe información de error. Para poner a "0" la salida después del tratamiento de errores, dispone de las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz del bloque.
- Ponga a "0" la variable antes de llamar la instrucción.
- Consulte la salida de habilitación ENO.

La salida de habilitación ENO de la instrucción "Consultar ID de error localmente" se activa únicamente si la entrada de habilitación EN devuelve el estado lógico "1" y existe información de error. Si no se cumple alguna de estas condiciones, la ejecución ulterior del programa no se ve afectada por la instrucción "Consultar ID de error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar ID de error localmente" activa el tratamiento local de errores dentro del bloque. Si en la lógica de un bloque se ha insertado la instrucción "Consultar ID de error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar ID de error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Output	WORD	I, Q, M, D, L	ID de error

Parámetro ID

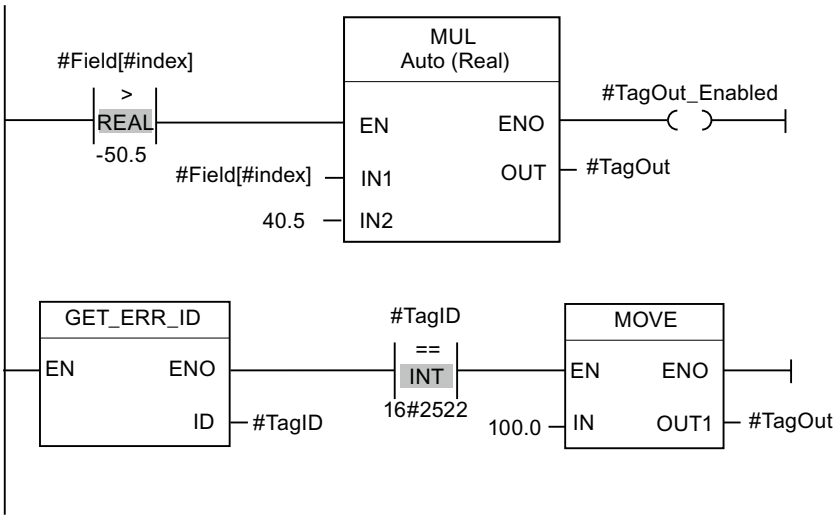
La tabla siguiente muestra los valores que pueden depositarse en el parámetro ID:

ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada

ID* (hexadecimal)	ID* (decimal)	Descripción
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Se ha producido un error al acceder a la variable "#Field[#index]". La salida de habilitación ENO de la instrucción "Multiplicar" y el operando "#TagOut_Enabled" devuelven el estado lógico "1" a pesar del error de lectura/escritura, y la multiplicación se realiza con el valor "0.0". En este caso de error, es recomendable programar la instrucción "Consultar ID de error localmente" después de la instrucción "Multiplicar" para interceptar el error. La información de error que devuelve la instrucción "Consultar ID de error localmente" se evalúa mediante la instrucción de comparación "Igual". Si el operando "#TagID" devuelve la ID 2522, se trata de un error de lectura/escritura y se escribe el valor "100.0" en la salida #TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones KOP (Página 2701)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tratar errores de ejecución del programa (Página 2932)

Consultar y eliminar errores en el código del programa (Página 265)

INIT_RD: Inicializar todos los datos remanentes

Descripción

La instrucción "Inicializar todos los datos remanentes" permite inicializar todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC al mismo tiempo. La instrucción solo puede ejecutarse dentro de un OB de arranque, puesto que su ejecución excedería el tiempo de ciclo de programa.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar todos los datos remanentes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	Si la entrada REQ devuelve el estado lógico "1", se inicializan todos los datos remanentes.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

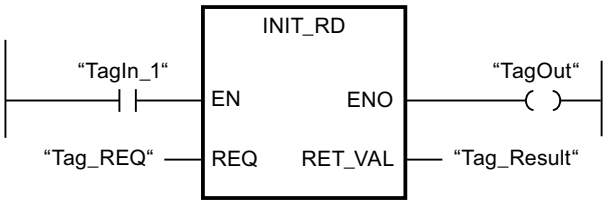
La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B5	No puede ejecutarse la instrucción porque no ha sido programada dentro de un OB de arranque.

Código de error* (W#16#...)	Explicación
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "Tag_REQ" devuelven el estado lógico "1", se ejecuta la instrucción. Se inicializan todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC. Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" permite detener la ejecución del programa durante un periodo predeterminado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible programar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

RUNTIME: Medir tiempo de ejecución

Descripción

La instrucción "Medir tiempo de ejecución" permite medir el tiempo de ejecución de todo el programa, bloques individuales o secuencias de comandos.

Si desea medir el tiempo de ejecución de todo el programa, efectúe la llamada de la instrucción "Medir tiempo de ejecución" en el OB1. El punto inicial para medir el tiempo de ejecución se determina con la primera llamada y la salida RET_VAL devuelve el tiempo de ejecución del programa tras la segunda llamada. El tiempo de ejecución medido incluye toda clase de procesos de la CPU que se hayan producido durante la ejecución del programa, como, p. ej., interrupciones a causa de eventos o comunicaciones de mayor prioridad. La instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU y escribe su valor en el parámetro de entrada/salida MEM. La instrucción calcula el tiempo de ejecución actual del programa según la frecuencia interna del contador, y la escribe en la salida RET_VAL.

Si desea medir el tiempo de ejecución de bloques o secuencias de comandos individuales, necesita tres segmentos diferentes. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" en un segmento individual dentro del programa. Al efectuar esta primera llamada de la instrucción se establece el punto inicial para medir el tiempo de ejecución. A continuación, llame el bloque de programa o la secuencia de comando que desee en el siguiente segmento.

Efectúe la llamada de la instrucción "Medir tiempo de ejecución" por segunda vez en otro segmento diferente y asigne al parámetro de entrada/salida MEM la misma memoria que en la primera llamada de la instrucción. En el tercer segmento, la instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU, calcula el tiempo de ejecución actual del bloque de programa o secuencia de comando según la frecuencia interna del contador y lo escribe en la salida RET_VAL.

La instrucción "Medir tiempo de ejecución" utiliza un contador interno de alta frecuencia para calcular el tiempo. Cuando el contador se desborda, la instrucción devuelve valores <= 0.0. Estos valores de runtime deben ignorarse.

Nota

Puesto que el orden de las instrucciones de una secuencia de comandos se modifica mediante la compilación optimizada del programa, no es posible determinar el tiempo de ejecución de una secuencia de comandos con exactitud.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Medir tiempo de ejecución":

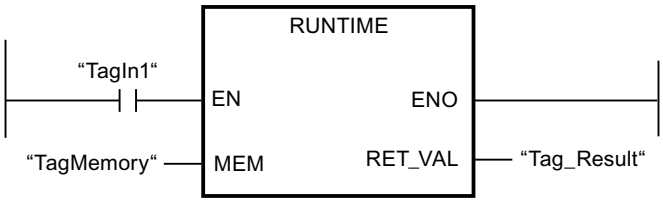
Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MEM	InOut	LREAL	M, D, L	Guarda el punto inicial de la medición del tiempo de ejecución
RET_VAL	Output	LREAL	I, Q, M, D, L	Devuelve el tiempo de ejecución medido en segundos

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

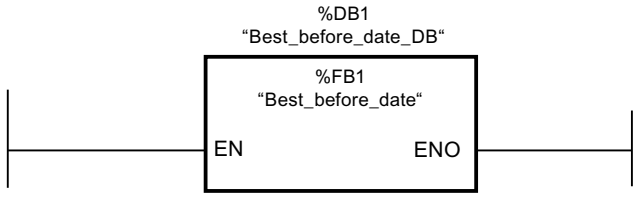
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción mediante el cálculo del tiempo de ejecución de un bloque de programa:

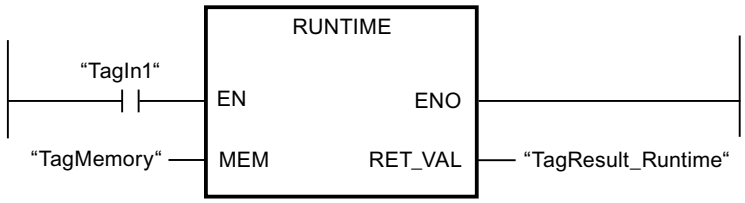
Segmento 1:



Segmento 2:



Segmento 3:



Si el operando "TagIn1" devuelve el estado lógico "1" en el segmento 1, se ejecuta la instrucción. Con la primera llamada de la instrucción se determina el punto inicial para la medición del tiempo de ejecución, y se guarda como referencia en una memora intermedia del operando "TagMemory" para la segunda llamada de la instrucción.

En el segmento 2 se llama el bloque de programa FB1 "Best_before_date".

Si el bloque de programa FB1 se ha procesado y el operando "TagIn1" devuelve el estado lógico "1", la instrucción se ejecuta en el segmento 3. La segunda llamada de la instrucción calcula el tiempo de ejecución del bloque de programa y escribe el resultado en la salida RET_VAL.

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Operaciones lógicas con palabras

AND: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una Y lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina con el bit 0 del valor de la entrada IN2 mediante una Y lógica. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits de los valores indicados.

El número de entradas del cuadro de la instrucción se puede ampliar. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se combinan mediante una Y lógica los valores de todos los parámetros de entrada disponibles. El resultado se deposita en la salida OUT.

El estado lógico del bit de resultado es "1" sólo si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

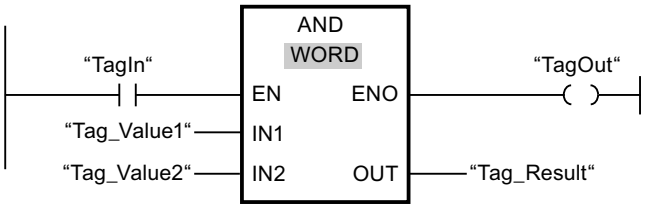
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
INn	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Otras entradas cuyos valores se combinan.
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0000 0000 0000 0101

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica Y". El valor del operando "Tag_Value1" se combina con el valor del operando "Tag_Value2" mediante una Y lógica. El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

OR: Operación lógica O

Descripción

La instrucción "Operación lógica O" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina con el bit 0 del valor de la entrada IN2 mediante una O lógica. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los bits de la variable indicada.

El número de entradas del cuadro de la instrucción se puede ampliar. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se combinan mediante una O lógica los valores de todos los parámetros de entrada disponibles. El resultado se deposita en la salida OUT.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

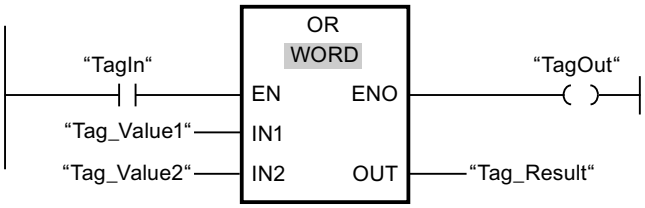
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
INn	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Otras entradas cuyos valores se combinan.
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O". El valor del operando "Tag_Value1" se combina con el valor del operando "Tag_Value2" mediante una O lógica. El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

XOR: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O exclusiva y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina con el bit 0 del valor de la entrada IN2 mediante una O exclusiva. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits del valor indicado.

El número de entradas del cuadro de la instrucción se puede ampliar. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se combinan mediante una O exclusiva los valores de todos los parámetros de entrada disponibles. El resultado se deposita en la salida OUT.

El estado lógico del bit de resultado es "1" si el estado lógico de uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "1" ó "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

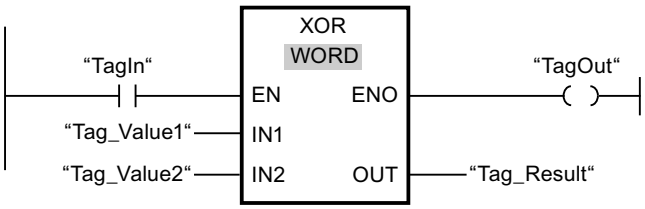
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
INn	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Otras entradas cuyos valores se combinan.
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1010

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O-exclusiva". El valor del operando "Tag_Value1" se combina con el valor del operando "Tag_Value2" mediante una O-exclusiva. El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos KOP (Página 2697)

Quitar entradas y salidas (Página 2698)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

INV: Complemento a 1

Descripción

La instrucción "Complemento a 1" permite invertir el estado lógico de los bits de la entrada IN. Al ejecutar la instrucción, el valor de la entrada IN se combina con una plantilla hexadecimal (W#16#FFFF para números de 16 bits o DW#16#FFFF FFFF para números de 32 bits) mediante una operación lógica O-exclusiva. Esto invierte el estado lógico de los distintos bits y deposita el resultado en la salida OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Complemento a 1":

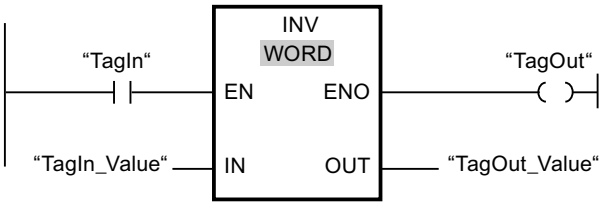
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Secuencias de bits, enteros	I, Q, M, D, L, P	Complemento a 1 del valor de la entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	W#16#000F	W#16#7E
OUT	TagOut_Value	W#16#FFF0	W#16#81

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Complemento a 1". La instrucción invierte el estado lógico de los distintos bits de la entrada "TagIn_Value" y escribe el resultado en la salida "TagOut_Value". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

DECO: Descodificar

Descripción

La instrucción "Descodificar" permite activar un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor de la entrada IN y activa el bit del valor de salida, cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor de la entrada IN es mayor que 31, se ejecuta una instrucción modulo 32.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

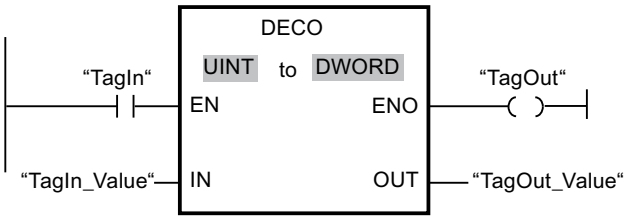
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	UINT	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida.
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

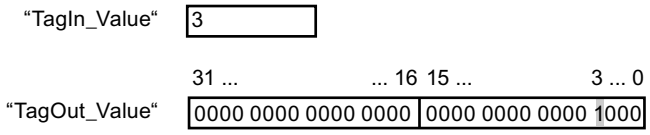
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Descodificar". La instrucción lee el número de bit "3" del valor del operando "TagIn_Value" de la entrada y activa el tercer bit del valor del operando "TagOut_Value" de la salida.

Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ENCO: Codificar

Descripción

La instrucción "Codificar" permite leer el número del bit menos significativo que está activado en el valor de entrada y depositarlo en la salida OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor de la entrada IN y escribe su número de bit en la variable de la salida OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

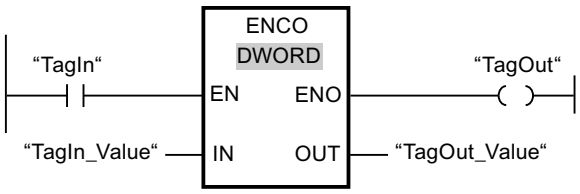
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

	31 16	15 ... 3 ... 0
“TagIn_Value“	0000 1111 0000 0101	0000 1001 0000 1000
“TagOut_Value“	3	

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Codificar". La instrucción selecciona el bit menos significativo que está activado en la entrada "TagIn_Value" y escribe el bit "3" en la variable de la salida "TagOut_Value".

Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite elegir, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copiar su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G tiene el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

Todas las variables de los parámetros deben tener el mismo tipo de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
G	Input	BOOL	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L, T, C o constante	Interruptor

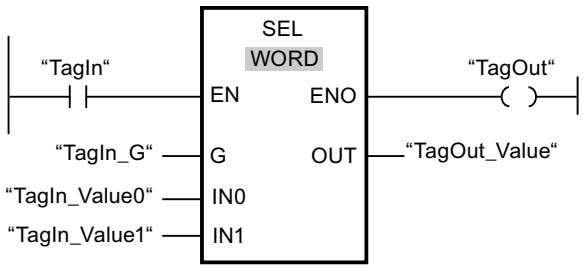
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN0	Input	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, CHAR, DATE	Secuencias de bits, enteros, números en coma flotante, temporizadores, TOD, LTOD, LDT, CHAR, DATE	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, CHAR, DATE	Secuencias de bits, enteros, números en coma flotante, temporizadores, TOD, LTOD, LDT, CHAR, DATE	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, CHAR, DATE	Secuencias de bits, enteros, números en coma flotante, temporizadores, TOD, LTOD, LDT, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C

Parámetro	Operando	Valor	
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Seleccionar". Dependiendo del estado lógico de la entrada "TagIn_G", se selecciona el valor de la entrada "TagIn_Value0" ó "TagIn_Value1" y se copia en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

MUX: Multiplexar

Descripción

La instrucción "Multiplexar" permite copiar el contenido de una entrada seleccionada en la salida OUT. El número de entradas seleccionables en el cuadro de la instrucción se puede ampliar. Se puede declarar un máximo de 32 entradas.

Las entradas se numeran automáticamente en el cuadro. La numeración comienza por IN0 y continúa en orden ascendente con cada nueva entrada. El parámetro K determina la entrada cuyo contenido se copia en la salida OUT. Si el valor del parámetro K es mayor que el número de entradas disponibles, el contenido del parámetro ELSE se copia en la salida OUT y a la salida de habilitación ENO se le asigna el estado lógico "0".

La instrucción "Multiplexar" solo se puede ejecutar si las variables de todas las entradas y de la salida OUT tienen el mismo tipo de datos. El parámetro K es una excepción, ya que para éste sólo pueden indicarse números enteros.

La salida de habilitación ENO se desactiva si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor del parámetro K es mayor que el número de entradas disponibles.
- Se producen errores al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplexar":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
K	Input	Enteros	Enteros	I, Q, M, D, L, P o constante	Indica la entrada cuyo contenido se copia. • Si K = 0 => parámetro IN0 • Si K = 1 => parámetro IN1, etc.
IN0	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN1	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
INn	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Valores de entrada opcionales

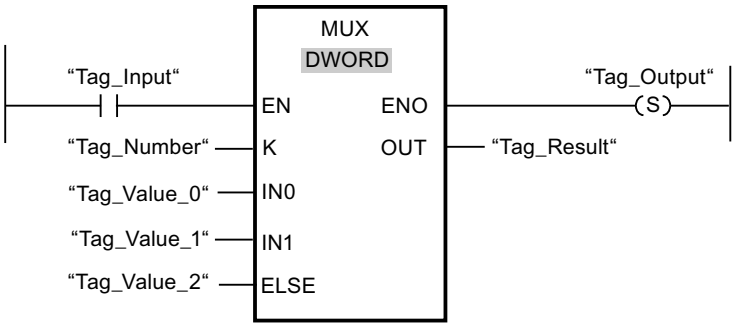
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
ELSE	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Indica el valor que se copia si K > n
OUT	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Salida en la que se copia el valor

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
K	Tag_Number	1
IN0	Tag_Value_0	DW#16#00000000
IN1	Tag_Value_1	DW#16#003E4A7D

Parámetro	Operando	Valor
ELSE	Tag_Value_2	DW#16#FFFF0000
OUT	Tag_Result	DW#16#003E4A7D

Si el operando "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplexar". De acuerdo con el valor del operando "Tag_Number", se copia el valor de la entrada "Tag_Value_1" y se asigna al operando de la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activan la salida de habilitación ENO y "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

DEMUX: Desmultiplexar

Descripción

La instrucción "Desmultiplexar" permite copiar el contenido de la entrada IN en una salida seleccionada. El número de salidas seleccionables del cuadro de la instrucción se puede ampliar. Las salidas se numeran automáticamente en el cuadro. La numeración comienza por OUT0 y continúa en orden ascendente con cada nueva salida. El parámetro K permite determinar la salida en la que se copia el contenido de la entrada IN. Las demás salidas no cambian. Si el valor del parámetro K es mayor que el número de salidas disponibles, el contenido de la entrada IN se copia en el parámetro ELSE y a la salida de habilitación ENO se le asigna el estado lógico "0".

La instrucción "Desmultiplexar" solo se puede ejecutar si las variables de la entrada IN y las de todas las salidas son del mismo tipo de datos. El parámetro K es una excepción, ya que en el mismo sólo pueden indicarse números enteros.

La salida de habilitación ENO se desactiva si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor del parámetro K es mayor que el número de salidas disponibles.
- Se producen errores al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desmultiplexar":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
K	Input	Enteros	Enteros	I, Q, M, D, L, P o constante	Indica la salida en la que se copia el valor de entrada (IN). • Si K = 0 => parámetro OUT0 • Si K = 1 => parámetro OUT1, etc.
IN	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Valor de entrada
OUT0	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Primera salida
OUT1	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Segunda salida

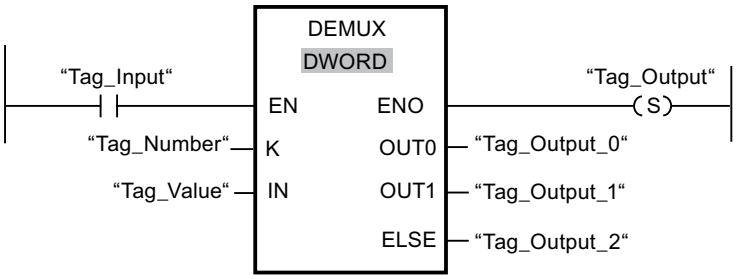
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
OUTn	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Salidas opcionales
ELSE	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Salida en la que se copia el valor de entrada (IN) con K > n.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos disponibles en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Las tablas siguientes muestran el funcionamiento de la instrucción con valores de operandos concretos:

Tabla 9-24 Valores de entrada de la instrucción "Desmultiplexar" antes de ejecutar el segmento

Parámetro	Operando	Valores	
K	Tag_Number	1	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#003E4A7D

Tabla 9-25 Valores de salida de la instrucción "Desmultiplexar" tras ejecutar el segmento

Parámetro	Operando	Valores	
OUT0	Tag_Output_0	Sin cambios	Sin cambios
OUT1	Tag_Output_1	DW#16#FFFFFFFF	Sin cambios
ELSE	Tag_Output_2	Sin cambios	DW#16#003E4A7D

Cuando la entrada "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Desmultiplexar". Según el valor del operando "Tag_Number", se copia el valor de la entrada IN en la salida correspondiente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

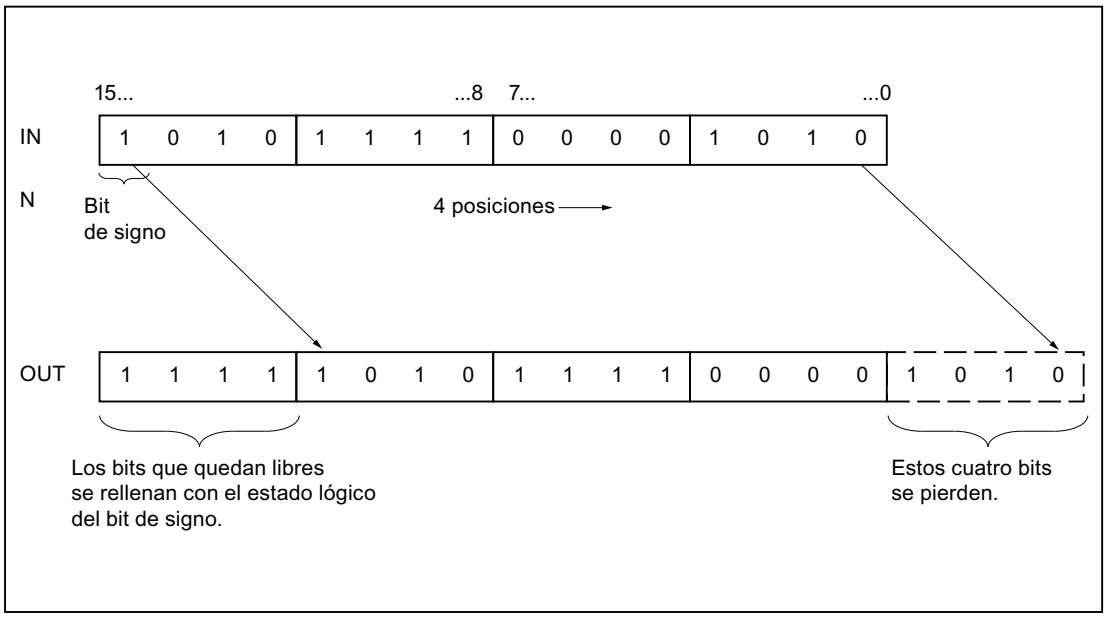
La instrucción "Desplazar a la derecha" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la derecha las posiciones de bit disponibles.

En los valores sin signo, se rellenan con ceros los bits que quedan libres en el área izquierda del operando al realizar el desplazamiento. Si el valor indicado lleva signo, las posiciones libres se ocupan con el estado lógico del bit de signo.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos Integer se desplaza cuatro posiciones de bit hacia la derecha:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

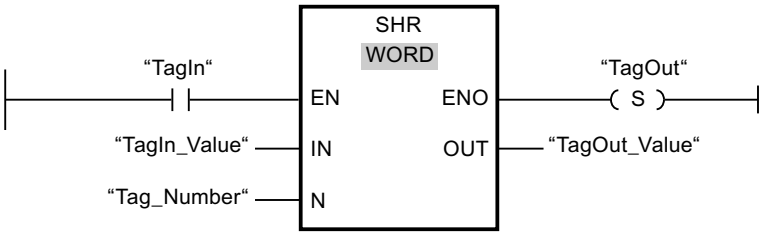
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se desplaza
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que se desplaza el valor
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la derecha". El contenido del operando "TagIn_Value" se desplaza tres posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

SHL: Desplazar a la izquierda

Descripción

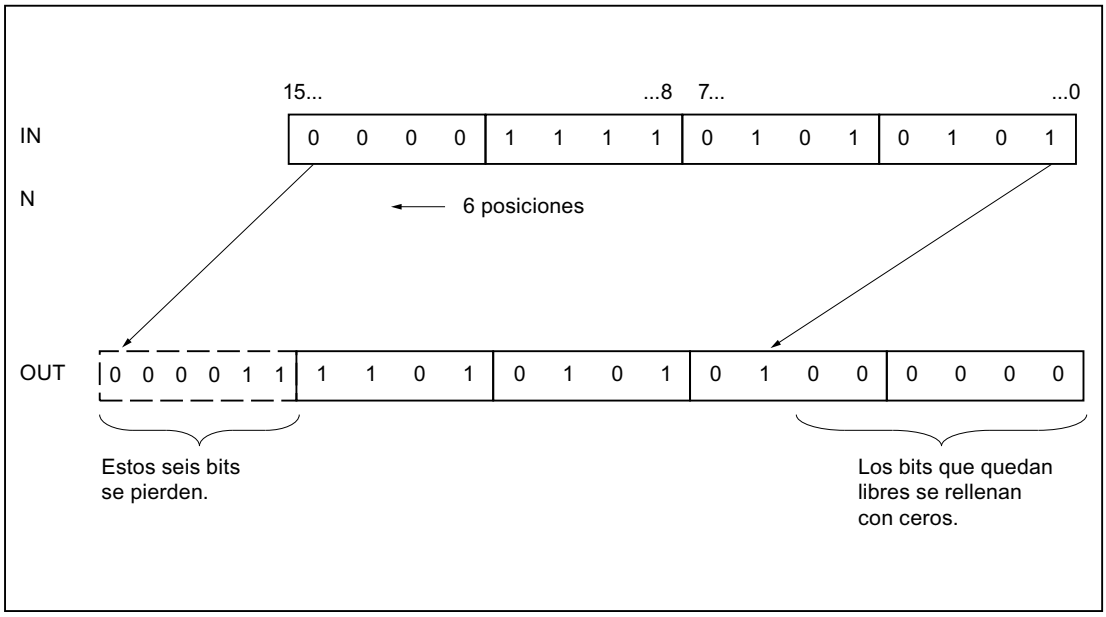
La instrucción "Desplazar a la izquierda" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la izquierda las posiciones de bit disponibles.

Los bits que quedan libres en el área derecha del operando al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

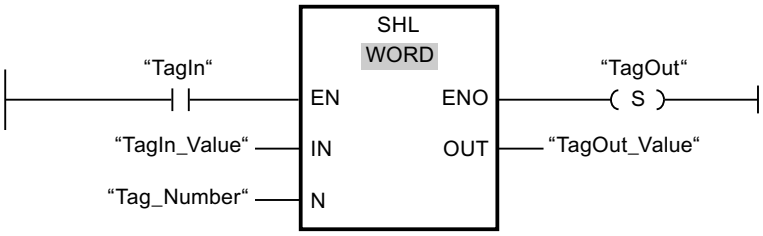
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se desplaza
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que se desplaza el valor
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la izquierda". El contenido del operando "TagIn_Value" se desplaza cuatro posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ROR: Rotar a la derecha

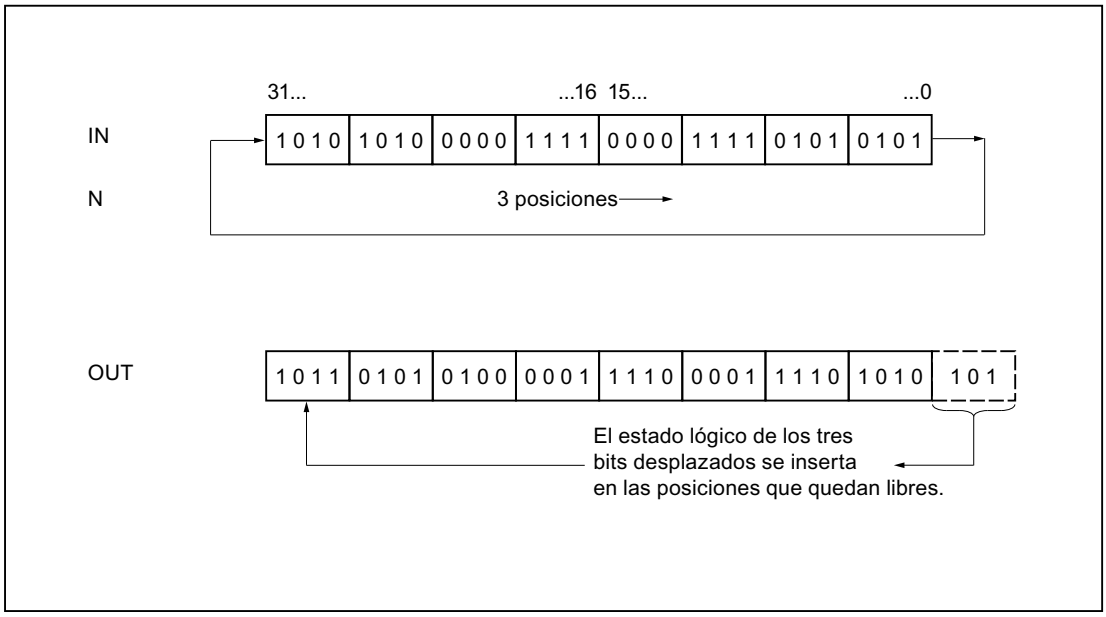
Descripción

La instrucción "Rotar a la derecha" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

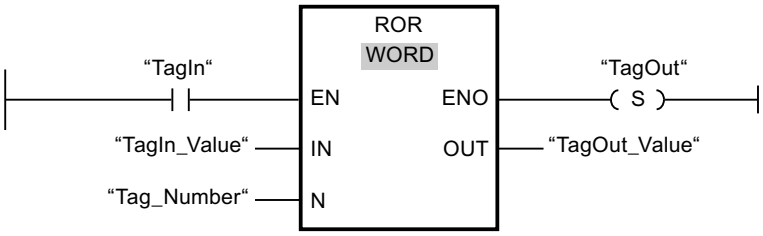
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se rota
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que rota el valor
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0000 1111 1001 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1000 0111 1100

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la derecha". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

ROL: Rotar a la izquierda

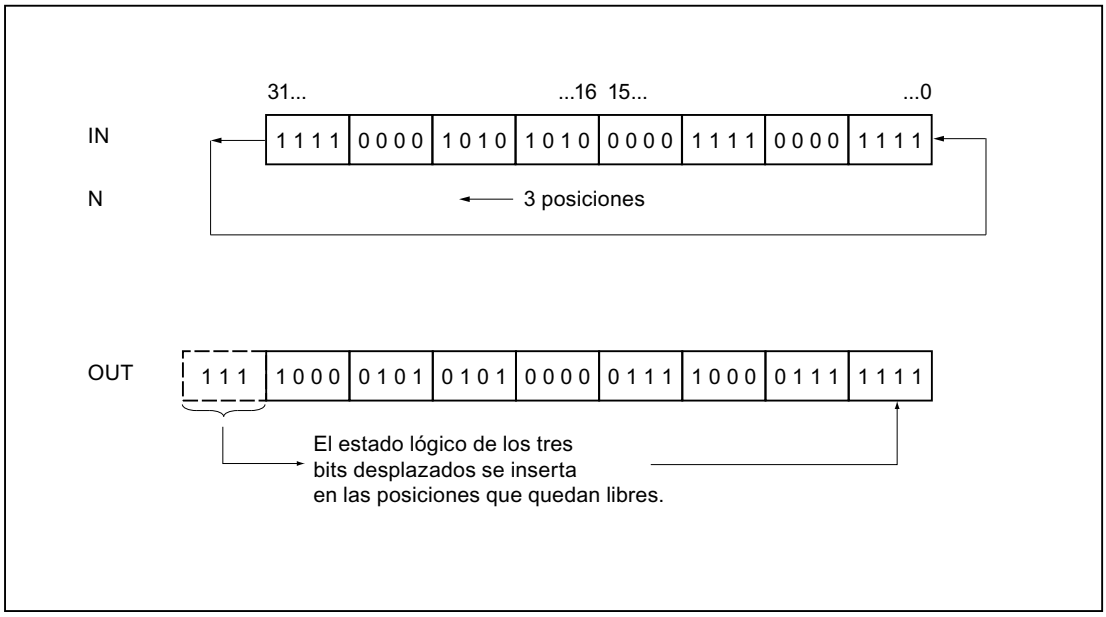
Descripción

La instrucción "Rotar a la izquierda" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

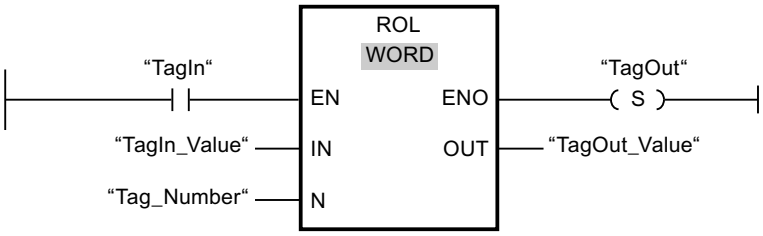
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se rota
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que rota el valor
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	1010 1000 1111 0110
N	Tag_Number	5
OUT	TagOut_Value	0001 1110 1101 0101

Si la entrada "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la izquierda". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Otras instrucciones

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite ocupar los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el

evento de éste es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L o constante	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L o constante	Bit de evento (i); el estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Paso predeterminado del circuito secuencial sucesivo

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P o constante	Valor de contaje actual del circuito secuencial sucesivo
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	Hora de sistema anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET (solo si ET < PT), OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1".
  - Los parámetros ET, OA y CA se ponen a "0".
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET (solo si ET < PT), OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
CMD	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
CMD_HIS	Static	BOOL	D, L o constante	Bit de evolución de CMD

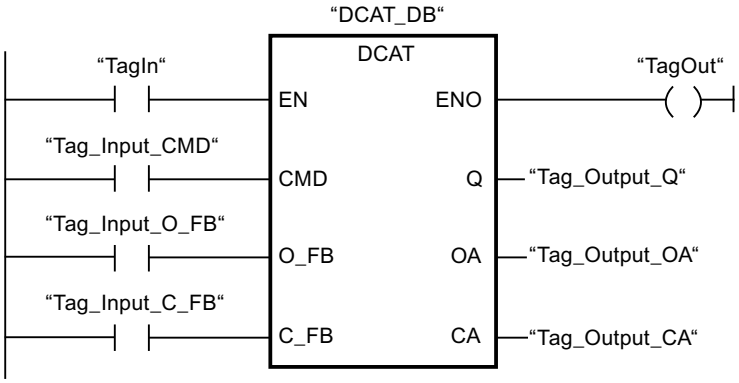
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se dispara la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

La instrucción "Alarma de tiempo para control del motor" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
O_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error.
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
O_HIS	Static	BOOL	D, L o constante	Bit de evolución "Abrir"
C_HIS	Static	BOOL	D, L o constante	Bit de evolución "Cerrar"

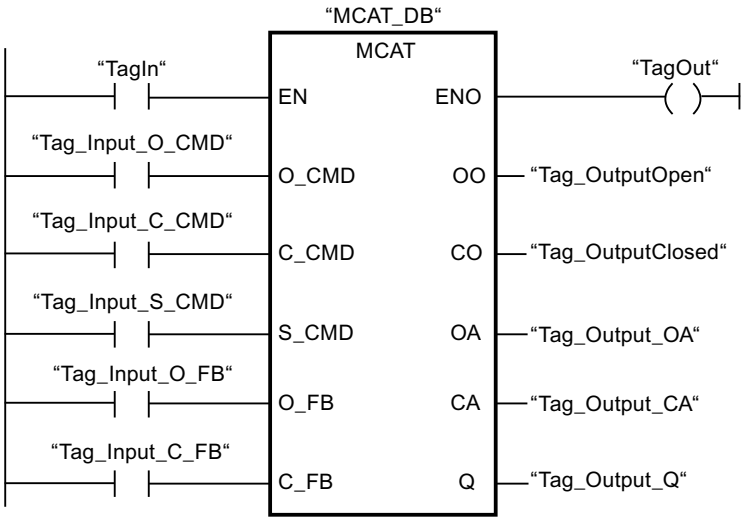
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22

Parámetro	Dirección	Valor
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una coincidencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica sólo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LAST	Static	BYTE	I, Q, M, D, L, P o constante	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor para la amplificación en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo dinámico. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse para suprimir perturbaciones o como filtro pasa bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa. En el parámetro IN también se pueden indicar constantes.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo En el parámetro SAMPLE_T también se pueden indicar constantes.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	REAL	I, Q, M, D, L	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de retardo de fase en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

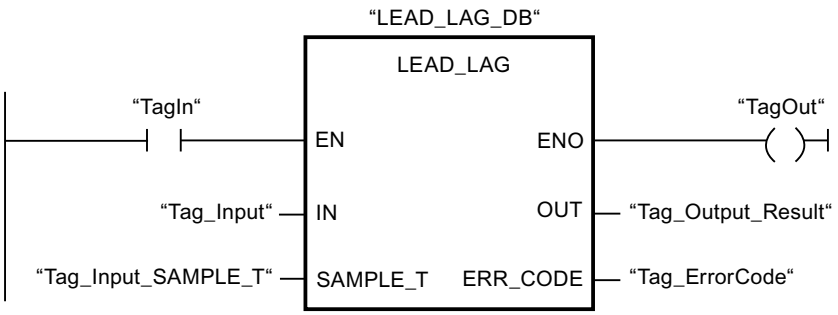
Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Los siguientes valores están almacenados en el bloque de datos de instancia "LEAD_LAG_DB" de la instrucción:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos de instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SEG: Generar patrón de bits para el display de 7 segmentos

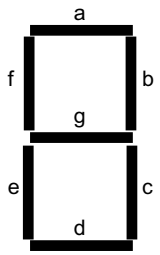
Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" permite convertir cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se devuelve en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6
0111	00000111	7
1000	01111111	8
1001	01100111	9
1010	01110111	A
1011	01111100	B
1100	00111001	C
1101	01011110	D
1110	01111001	E
1111	01110001	F

Display de siete segmentos



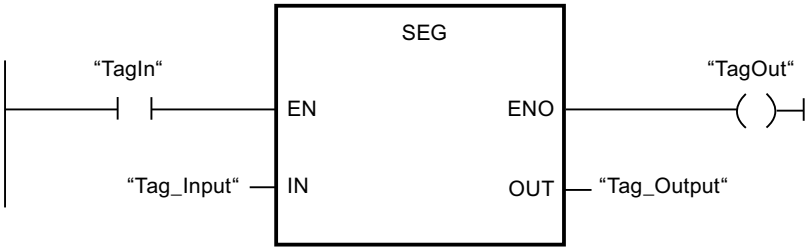
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits para el display de 7 segmentos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

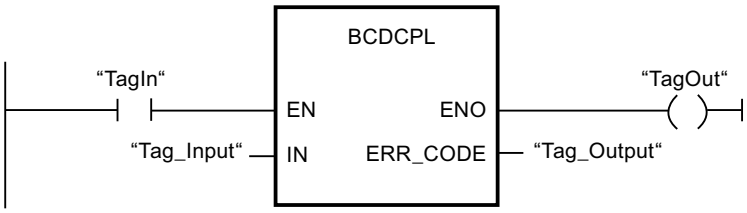
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" permite contar el número de bits de un operando que están puestos al estado lógico "1". El operando, cuyos bits se cuentan, se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

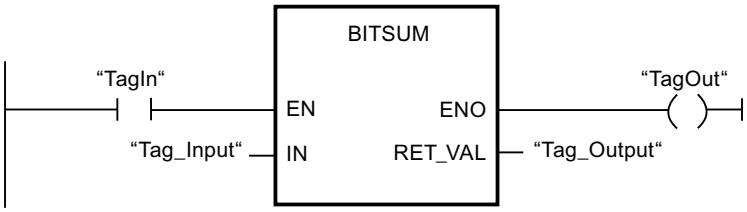
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan.
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

FUP

Operaciones lógicas con bits

&: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" permite consultar los estados lógicos de dos o más operandos indicados y evaluarlos según la tabla de verdad Y.

Si el estado lógico de todos los operandos es "1", entonces se cumple la condición y la instrucción da como resultado "1". Si el estado lógico de un operando es "0", no se cumple la condición y la instrucción da como resultado "0".

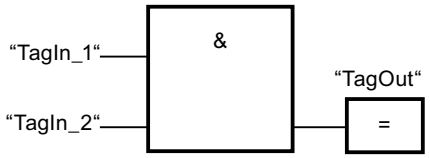
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", y se desactiva cuando el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "0".

Consulte también

Tabla de verdad Y (Página 3589)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3217)

Ejemplo de control de la temperatura ambiente (Página 3222)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entrada (Página 3594)

Tabla de verdad Y

Resultados lógicos

La tabla siguiente muestra los resultados que se obtienen combinando dos operandos mediante una Y lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Resultado lógico
1	1	1
0	1	0
1	0	0
0	0	0

Consulte también

&: Operación lógica Y (Página 3588)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

>=1: Operación lógica O

Descripción

La instrucción "Operación lógica O" permite consultar los estados lógicos de dos o más operandos indicados y evaluarlos según la tabla de verdad O.

Si el estado lógico de uno de los operandos es "1", entonces se cumple la condición y la instrucción da como resultado "1". Si el estado lógico de todos los operandos es "0", no se cumple la condición y la instrucción da como resultado "0".

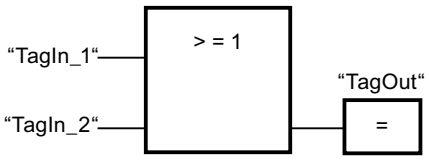
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa si el estado lógico del operando "TagIn_1" o del operando "TagIn_2" es "1".

Consulte también

Tabla de verdad O (Página 3591)

Ejemplo de control de una cinta transportadora (Página 3216)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Insertar entrada (Página 3594)

Tabla de verdad O

Resultados lógicos

La tabla siguiente muestra los resultados que se obtienen combinando dos operandos mediante una O lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Resultado lógico
1	0	1
0	1	1
1	1	1
0	0	0

Consulte también

>=1: Operación lógica O (Página 3590)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

X: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" permite consultar el resultado de la consulta del estado lógico de acuerdo con la tabla de verdad O-exclusiva.

En una instrucción "Operación lógica O-exclusiva", el estado lógico es "1" si el estado lógico de uno de los dos operandos indicados es "1". Si se consultan más de dos operandos, el resultado lógico común es "1" si un número impar de los operandos consultados devuelve el resultado "1".

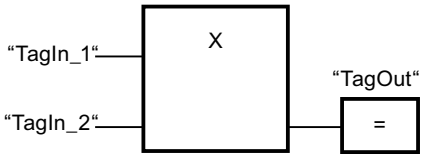
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa si el estado lógico de uno de los operandos "TagIn_1" y "TagIn_2" es "1". Si ambos operandos devuelven el estado lógico "1" ó "0", se desactiva la salida "TagOut".

Consulte también

Tabla de verdad O-exclusiva (Página 3593)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Insertar entrada (Página 3594)

Tabla de verdad O-exclusiva

Resultados lógicos

La tabla siguiente muestra los resultados que se obtienen combinando dos operandos mediante una O-exclusiva lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Resultado lógico
1	0	1
0	1	1
1	1	0
0	0	0

La tabla siguiente muestra los resultados que se obtienen combinando tres operandos mediante una O-exclusiva lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Estado lógico del tercer operando	Resultado lógico
1	0	0	1
0	1	1	0
0	1	0	1
1	0	1	0
0	0	1	1
1	1	0	0
1	1	1	1
0	0	0	0

Consulte también

X: Operación lógica O-exclusiva (Página 3591)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entrada

Descripción

La instrucción "Insertar entrada" permite ampliar el cuadro de una de las instrucciones siguientes con una entrada:

- "Operación lógica Y"
- "Operación lógica O"
- "Operación lógica O-exclusiva"

Ampliando el cuadro de una instrucción se puede consultar el estado lógico de varios operandos.

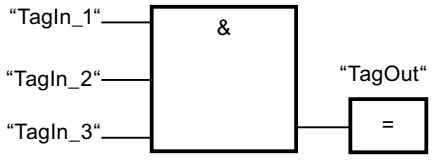
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Insertar entrada":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El cuadro de la instrucción "Operación lógica Y" ha sido ampliado con una entrada adicional, en la que se consulta el estado lógico del operando "TagIn_3". La salida "TagOut" se activa cuando los operandos "TagIn_1", "TagIn_2" y "TagIn_3" devuelven el estado lógico "1".

Consulte también

&: Operación lógica Y (Página 3588)

>=1: Operación lógica O (Página 3590)

X: Operación lógica O-exclusiva (Página 3591)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

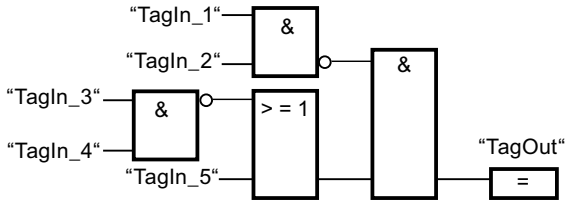
Invertir RLO

Descripción

La instrucción "Invertir RLO" invierte el estado lógico del resultado lógico (RLO).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- La entrada "TagIn_1" y/o "TagIn_2" devuelve el estado lógico "0".
- La entrada "TagIn_3" y/o "TagIn_4" devuelve el estado lógico "0" o la entrada "TagIn_5" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

=: Asignación

Descripción

La instrucción "Asignación" permite activar el bit de un operando indicado. Si el resultado lógico (RLO) de la entrada del cuadro es el estado lógico "1", el operando indicado adopta el estado lógico "1". Si el estado lógico de la entrada del cuadro es "0", el bit del operando indicado se pone a "0".

La instrucción no afecta al RLO. El RLO de la entrada del cuadro se asigna directamente a los operandos disponibles a través del cuadro de asignación.

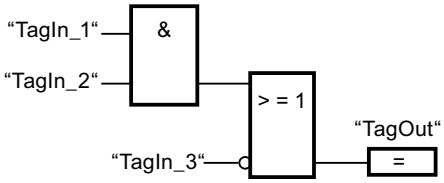
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" en la salida de la instrucción "Asignación" se activa cuando se cumple una de las condiciones siguientes:

- Las entradas "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico de la entrada "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3218)

Ejemplo de control de la temperatura ambiente (Página 3222)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

/=: Negar asignación

Descripción

La instrucción "Negar asignación" invierte el resultado lógico (RLO) y lo asigna al operando que figura sobre el cuadro. Si el RLO de la entrada del cuadro es "1", se desactiva el operando. Si el RLO de la entrada del cuadro es "0", el operando adopta el estado lógico "1".

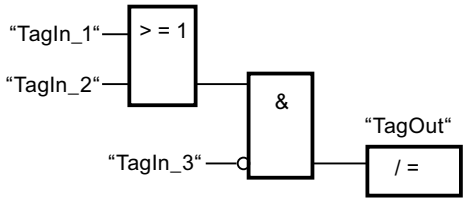
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Negar asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO negado.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" o "TagIn_2" tiene el estado lógico "1".
- El operando "TagIn_3" tiene el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

R: Desactivar salida

Descripción

La instrucción "Desactivar salida" permite poner a "0" el estado lógico de un operando indicado.

La instrucción se ejecuta sólo si el resultado lógico (RLO) de la entrada del cuadro es "1". Si el estado lógico de la entrada del cuadro es "1", el operando indicado se pone a "0". Si el RLO de la entrada del cuadro es "0", el estado lógico del operando indicado no cambia.

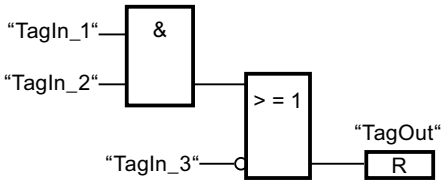
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando>	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Operando que se desactiva si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "0".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de control de una cinta transportadora (Página 3216)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3217)

Vista general de los tipos de datos válidos (Página 2321)

S: Activar salida

Descripción

La instrucción "Activar salida" permite poner a "1" el estado lógico de un operando indicado.

La instrucción se ejecuta sólo si el resultado lógico (RLO) de la entrada del cuadro es "1". Si el estado lógico de la entrada del cuadro es "1", el operando indicado se pone a "1". Si el RLO de la entrada del cuadro es "0", el estado lógico del operando indicado no cambia.

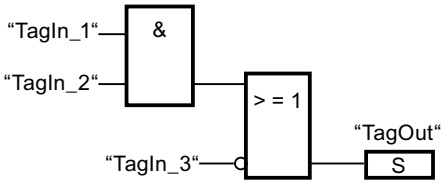
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se activa si RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "0".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

SET_BF: Activar mapa de bits

Descripción

La instrucción "Activar mapa de bits" activa varios bits a partir de una dirección específica.

El número de bits que deben activarse se determina mediante el valor de la entrada N. El <Operando> determina la dirección del primer bit que debe activarse. Si el valor de la entrada N es mayor que el número de bits de un byte seleccionado, se activan los bits del byte siguiente. Los bits permanecen activados hasta que son desactivados explícitamente por otra instrucción.

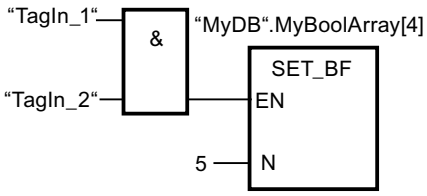
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Activar mapa de bits":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
N	Input	UINT	Constante	Constante	Número de bits que deben activarse.
<Operando>	Output	BOOL	I, Q, M En un DB o un IDB, un elemento de ARRAY[..] of BOOL	I, Q, M En un DB o un IDB, un elemento de ARRAY[..] of BOOL	Puntero al primer bit que debe activarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se activan 5 bits a partir de la dirección del operando "MyDB".MyBoolArray[4].

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

RESET_BF: Desactivar mapa de bits

Descripción

La instrucción "Desactivar mapa de bits" desactiva varios bits a partir de una dirección específica.

El número de bits que se deben desactivar se determina mediante el valor de la entrada N. El <Operando> determina la dirección del primer bit que debe desactivarse. Si el valor de la

entrada N es mayor que el número de bits de un byte seleccionado, se desactivan los bits del byte siguiente. Los bits permanecen desactivados hasta que son activados explícitamente por otra instrucción.

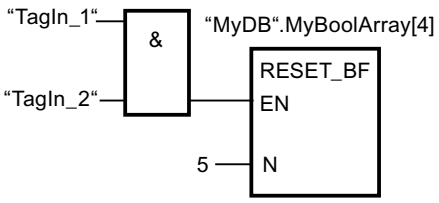
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Desactivar mapa de bits":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
N	Input	UINT	Constante	Constante	Número de bits que se deben desactivar.
<Operando>	Output	BOOL	I, Q, M En un DB o un IDB, un elemento de ARRAY[..] of BOOL	I, Q, M En un DB o un IDB, un elemento de ARRAY[..] of BOOL	Puntero al primer bit que debe desactivarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se desactivan 5 bits a partir de la dirección del operando "MyDB".MyBoolArray[4].

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SR: Flipflop de activación/desactivación

Descripción

La instrucción "Flipflop de activación/desactivación" sirve para activar o desactivar el bit de un operando indicado en función del estado lógico de las entradas S y R1. Si el estado lógico de la entrada S es "1" y el de la entrada R1 es "0", el operando indicado se pone a "1". Si el estado lógico de la entrada S es "0" y el de la entrada R1 es "1", el operando indicado se pone a "0".

La entrada R1 predomina sobre la entrada S. Si el estado lógico de las entradas S y R1 es "1", el estado lógico del operando indicado se pone a "0".

Si el estado lógico de ambas entradas S y R1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

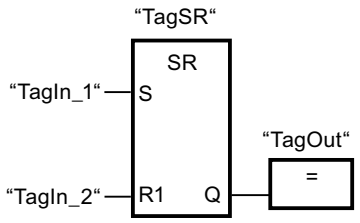
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de activación/desactivación":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
S	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Activación habilitada
R1	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Desactivación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Operando que se activa o desactiva
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagSR" y "TagOut" se activan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagSR" y "TagOut" se desactivan cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

RS: Flipflop de desactivación/activación

Descripción

La instrucción "Flipflop de desactivación/activación" sirve para desactivar o activar el bit de un operando indicado en función del estado lógico de las entradas R y S1. Si el estado lógico de la entrada R es "1" y el de la entrada S1 es "0", el operando indicado se pone a "0". Si el estado lógico de la entrada R es "0" y el de la entrada S1 es "1", el operando indicado se pone a "1".

La entrada S1 predomina sobre la entrada R. Si el estado lógico de las entradas R y S1 es "1", el estado lógico del operando indicado se pone a "1".

Si el estado lógico de ambas entradas R y S1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

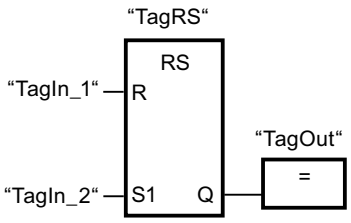
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de desactivación/activación":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
R	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Desactivación habilitada
S1	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Activación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Operando que se desactiva o activa.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagRS" y "TagOut" se desactivan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagRS" y "TagOut" se activan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

P: Consultar flanco de señal ascendente de un operando

Descripción

La instrucción "Consultar flanco de señal ascendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "0" a "1". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del resultado lógico (RLO) de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

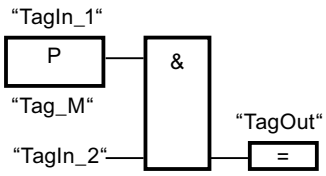
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando1>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- En la entrada "TagIn_1" hay un flanco de señal ascendente.
- El estado lógico del operando "TagIn_2" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3217)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

N: Consultar flanco de señal descendente de un operando

Descripción

La instrucción "Consultar flanco de señal descendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "1" a "0". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del resultado lógico (RLO) de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

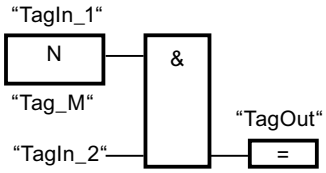
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
<Operando1>	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- En la entrada "TagIn_1" hay un flanco de señal descendente.
- El estado lógico del operando "TagIn_2" es "1".

Consulte también

Insertar elementos FUP (Página 2722)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Editar elementos FUP (Página 2738)

P=: Activar operando con flanco de señal ascendente

Descripción

La instrucción "Activar operando con flanco de señal ascendente" permite activar un operando indicado (<Operando2>) cuando se detecta un cambio del resultado lógico (RLO) de "0" a "1". La instrucción compara el RLO actual con el RLO de la consulta anterior, que está almacenado en una marca de flancos (<Operando1>). Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, el <Operando2> adopta el estado lógico "1" durante un ciclo del programa. En los demás casos, el estado lógico del operando es "0".

Indique el operando que debe activarse (<Operando2>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando1>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

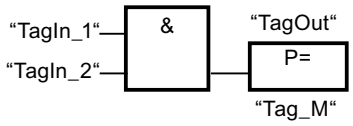
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar operando con flanco de señal ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando2>	Output	BOOL	I, Q, M, D, L	Operando que se activa si hay un flanco de señal ascendente.
<Operando1>	InOut	BOOL	I, Q, M, D, L	Marca de flancos

Ejemplo

El siguiente ejemplo muestra los parámetros de la instrucción:



La salida "TagOut" se activa durante un ciclo del programa cuando el estado lógico de la entrada del cuadro de la instrucción cambia de "0" a "1" (flanco de señal ascendente). En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

N=: Activar operando con flanco de señal descendente

Descripción

La instrucción "Activar operando con flanco de señal descendente" permite activar un operando indicado (<Operando1>) cuando se presenta un cambio del resultado lógico (RLO) de "1" a "0". La instrucción compara el RLO actual con el RLO de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, el <Operando1> adopta el estado lógico "1" durante un ciclo del programa. En los demás casos, el estado lógico del operando es "0".

Indique el operando que debe activarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

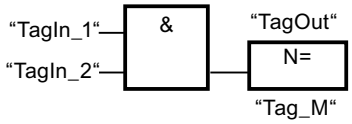
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar operando con flanco de señal descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Output	BOOL	I, Q, M, D, L	Operando que se activa si hay un flanco de señal descendente.
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa durante un ciclo del programa cuando el estado lógico de la entrada del cuadro de la instrucción cambia de "1" a "0" (flanco de señal descendente). En todos los demás casos, el operando "TagOut" devuelve el estado lógico "0".

Consulte también

Insertar elementos FUP (Página 2722)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Editar elementos FUP (Página 2738)

P_TRIG: Consultar flanco de señal ascendente del RLO

Descripción

La instrucción "Consultar flanco de señal ascendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "0" a "1". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

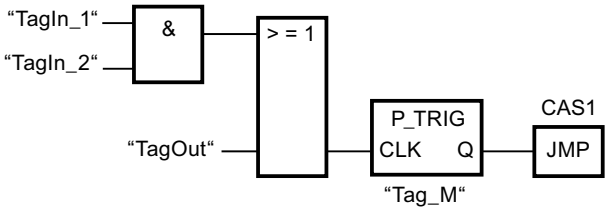
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "0" a "1", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

N_TRIG: Consultar flanco de señal descendente del RLO

Descripción

La instrucción "Consultar flanco de señal descendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "1" a "0". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

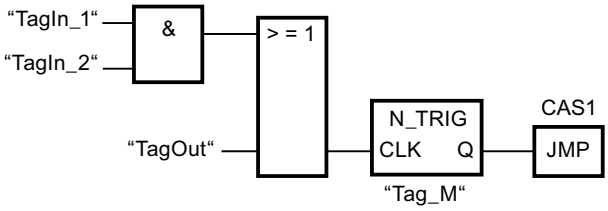
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "1" a "0", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

R_TRIG: Activar variable con flanco de señal ascendente

Descripción

La instrucción "Activar variable con flanco de señal ascendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "0" a "1". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

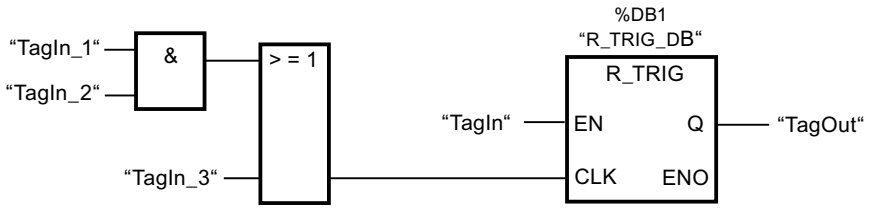
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
CLK	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L, T, C o constante	Señal entrante cuyo flanco se consulta.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la variable del DB de instancia "R_TRIG_DB" se almacena el RLO de la consulta anterior. Si en los operandos "TagIn_1" y "TagIn_2" o en el operando "TagIn_3" se detecta un cambio del estado lógico del RLO de "0" a "1", la salida "TagOut" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

F_TRIG: Activar variable con flanco de señal descendente

Descripción

La instrucción "Activar variable con flanco de señal descendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "1" a "0". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta

anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

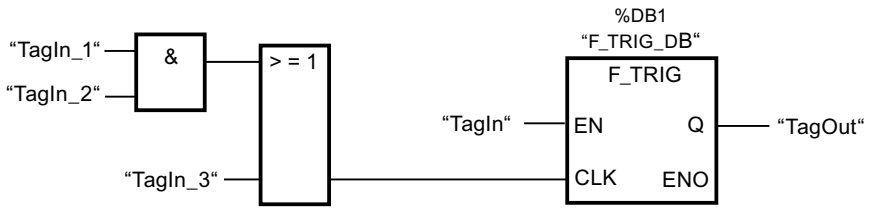
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
CLK	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L, T, C o constante	Señal entrante cuyo flanco se consulta.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la variable del DB de instancia "F_TRIG_DB" se almacena el RLO de la consulta anterior. Si en los operandos "TagIn_1" y "TagIn_2" o en el operando "TagIn_3" se detecta un cambio del estado lógico del RLO de "1" a "0", la salida "TagOut" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Temporizadores

Temporizadores CEI

TP: Impulso

Descripción

La instrucción "Impulso" permite activar la salida Q durante el tiempo PT parametrizado. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Al iniciar la instrucción, se empieza a contar el tiempo parametrizado PT. La salida Q se activa por el tiempo PT, independientemente de cómo evolucione (flanco de señal ascendente) la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico de la salida Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo parametrizado PT y si el estado lógico en la entrada IN es "0", se desactiva la salida ET.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TP_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME, TP_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Impulso" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

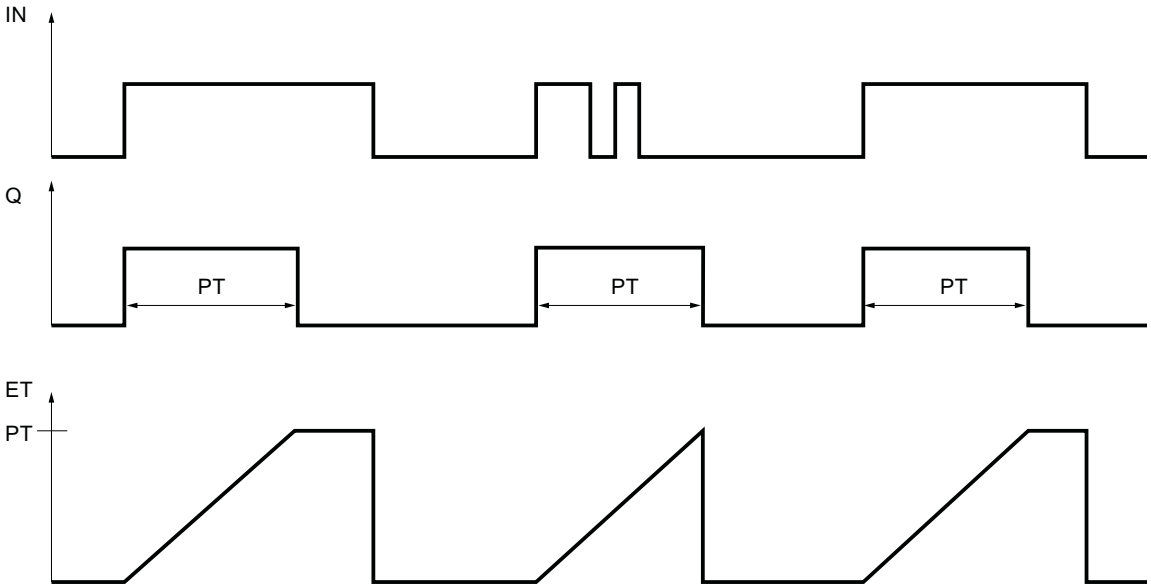
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C, P	Entrada de arranque
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Duración del impulso. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida de impulso
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de control de la temperatura ambiente (Página 3222)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

TON: Retardo al conectar

Descripción

Con la instrucción "Retardo al conectar" se puede retrasar la activación de la salida Q el tiempo parametrizado PT. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. Una vez transcurrido el tiempo PT, la salida Q devuelve el estado lógico "1". La salida Q permanece activada mientras la entrada de arranque esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva la salida Q. La función de temporización se reinicia al detectarse un nuevo flanco de señal ascendente en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. La salida ET se desactiva en cuanto el estado lógico de la entrada IN cambia a "0".

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TON_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME, TON_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Retardo al conectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

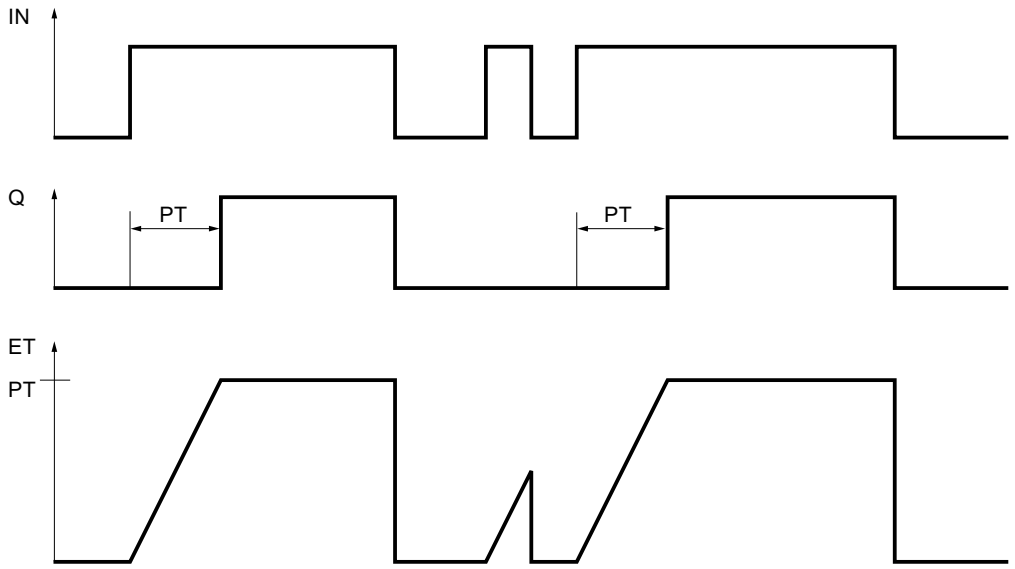
La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C, P	Entrada de arranque
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Tiempo de retardo al conectar. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

TOF: Retardo al desconectar

Descripción

Con la instrucción "Retardo al desconectar" se puede retrasar la desactivación de la salida Q el tiempo parametrizado PT. La salida Q se activa cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico de la entrada IN cambia nuevamente a "0" (flanco de señal ascendente), el tiempo parametrizado PT deja de contar. La salida Q permanece activada mientras transcurre el tiempo PT. Una vez transcurrido el tiempo PT se desactiva la salida Q. Si el estado lógico de la entrada IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico de la salida Q permanece en "1".

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el valor actual de la salida ET se conserva hasta que la entrada IN cambie nuevamente a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TOF_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME, TOF_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Retardo al desconectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

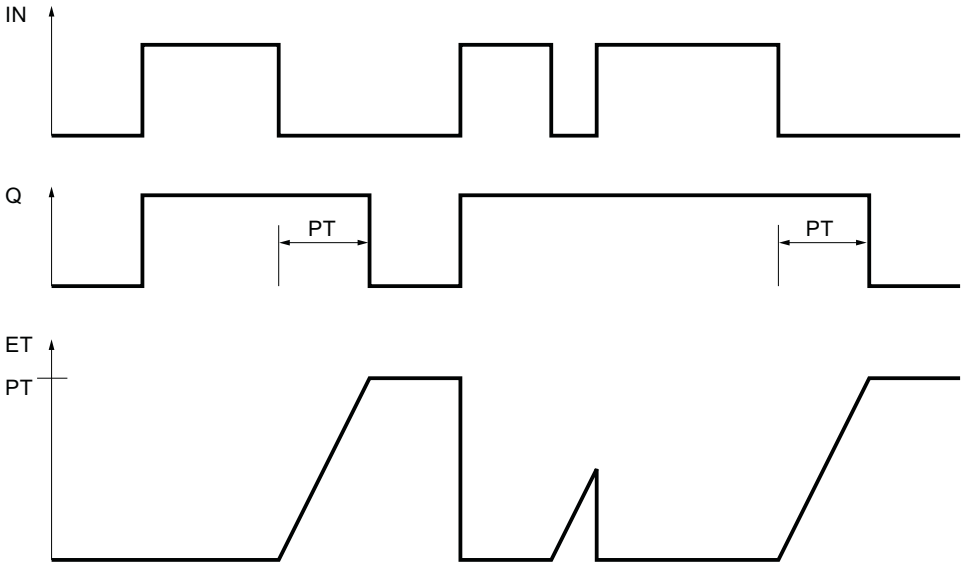
La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C, P	Entrada de arranque
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

TONR: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PT. Cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y se empieza a contar el tiempo PT parametrizado. Mientras transcurre el tiempo PT se van acumulando los valores de tiempo que se leen cuando el estado lógico de la entrada IN es "1". El tiempo acumulado se deposita en la salida ET y se puede consultar allí. Una vez alcanzado el valor de tiempo actual PT, la salida Q devuelve el estado lógico "1". La salida Q permanece a "1" aunque el estado lógico de la entrada IN cambie a "0".

La entrada R desactiva las salidas ET y Q, independientemente del estado lógico de la entrada de arranque.

A cada llamada de la instrucción "Acumulador de tiempo" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TONR_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej., #MyIEC_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME, TONR_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Acumulador de tiempo" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

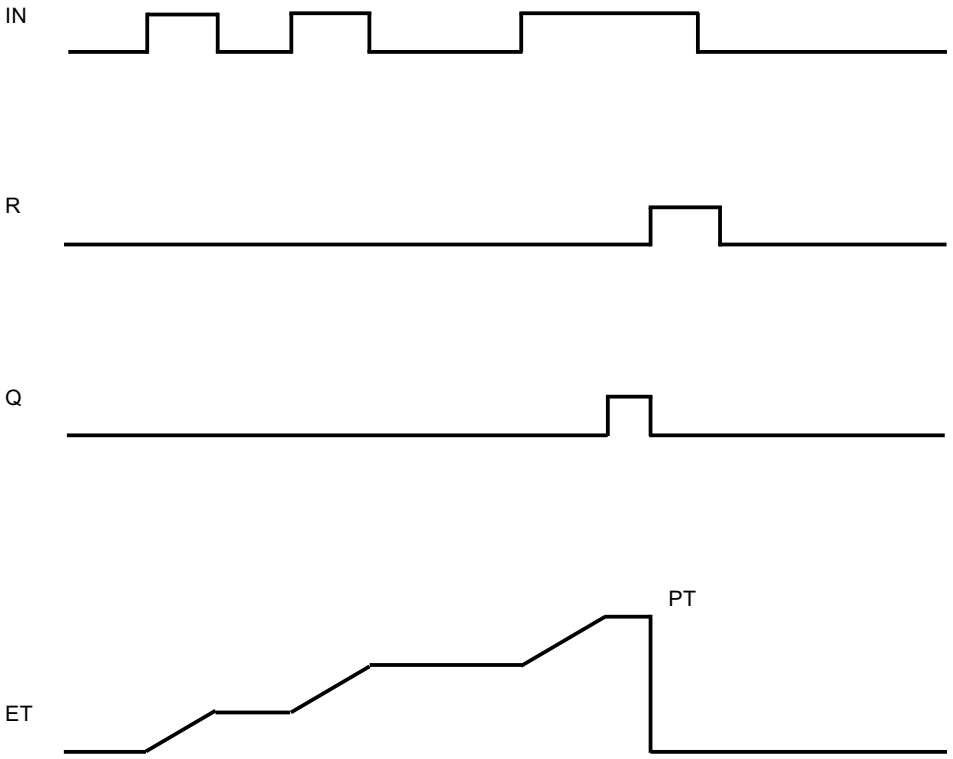
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C, P	Entrada de arranque
R	Input	BOOL	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Entrada de reset

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, P	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	I, Q, M, D, L	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Acumulador de tiempo":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

TP: Arrancar temporizador como impulso

Descripción

La instrucción "Arrancar temporizador como impulso" permite arrancar un temporizador CEI como impulso por un tiempo determinado. El temporizador CEI se inicia cuando el resultado lógico (RLO) cambia de "0" a "1" (flanco de señal ascendente). El temporizador CEI cuenta el tiempo indicado, independientemente de cómo evolucione el RLO. La detección de un nuevo flanco de señal ascendente no influye en el transcurso del temporizador CEI. Mientras el temporizador CEI cuenta el tiempo, una consulta de si el estado del temporizador es "1" da como resultado "1". Una vez transcurrido el temporizador CEI, el estado del temporizador devuelve el estado lógico "0".

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles secuenciales, pues en cada consulta de las salidas Q o ET se actualiza la estructura IEC_TIMER.

Para CPU S7-1200

La instrucción "Arrancar temporizador como impulso" deposita los datos en una estructura del tipo de datos IEC_TIMER o TP_TIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Arrancar temporizador como impulso" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME, TP_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura Q del temporizador CEI. El estado del temporizador se puede consultar con una combinación lógica binaria. La consulta de Q o ET (p. ej. "MyTimer".Q o "MyTimer".ET) actualiza la estructura IEC_TIMER.

Para poder ejecutar la instrucción "Arrancar temporizador como impulso" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetro

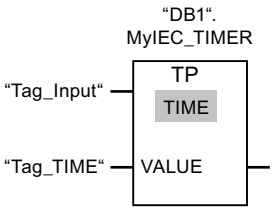
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TP_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME	D, L	Temporizador CEI que se arranca.

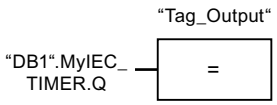
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Arrancar temporizador como impulso" se ejecuta cuando el estado lógico del operando "Tag_Input" cambia de "0" a "1". El temporizador "DB1".MyIEC_TIMER se arranca por el tiempo que está guardado en el operando "TagTime".



Mientras el temporizador "DB1".MyIEC_TIMER cuenta el tiempo, el estado del temporizador ("DB1".MyIEC_TIMER.Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa. Cuando el temporizador CEI termina de contar, el estado lógico del temporizador cambia a "0" y el operando "Tag_Output" se desactiva.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

TON: Arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión" permite arrancar un temporizador CEI como retardo a la conexión por un tiempo determinado. El temporizador CEI se inicia cuando el resultado lógico (RLO) cambia de "0" a "1" (flanco de señal ascendente). El temporizador CEI cuenta el tiempo indicado. La salida devuelve el estado lógico "1" si RLO en la entrada de la instrucción tiene el estado lógico "1". Si el RLO cambia a "0" antes de transcurrir el tiempo ajustado, el temporizador CEI en curso se reinicia. La consulta de si el estado del temporizador es "1" devuelve en este caso el estado lógico "0". Al detectarse el siguiente flanco de señal ascendente en la entrada de la instrucción arranca nuevamente el temporizador CEI.

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles secuenciales, pues en cada consulta de las salidas Q o ET se actualiza la estructura IEC_TIMER.

Para CPU S7-1200

La instrucción "Arrancar temporizador como retardo a la conexión" deposita los datos en una estructura del tipo de datos IEC_TIMER o TON_TIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Arrancar temporizador como retardo a la conexión" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME, TON_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura ET del temporizador CEI. El estado del temporizador se puede consultar con una combinación lógica binaria. La consulta de Q o ET (p. ej. "MyTimer".Q o "MyTimer".ET) actualiza la estructura IEC_TIMER.

Para poder ejecutar la instrucción "Arrancar temporizador como retardo a la conexión" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetros

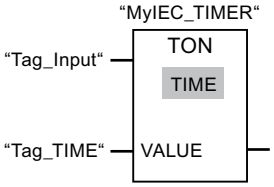
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TON_TIME	IEC_TIMER, IEC_LTIMER, TON_TIME, TON_LTIME	D, L	Temporizador CEI que se arranca.

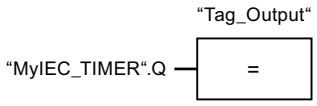
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Arrancar temporizador como retardo a la conexión" se ejecuta cuando el estado lógico del operando "Tag_Input" cambia de "0" a "1". El temporizador "MyIEC_TIMER" se arranca por el tiempo que está guardado en el operando "Tag_TIME".



Cuando finaliza el tiempo "MyIEC_TIMER" y el operando "Tag_Input" devuelve el estado lógico "1", la consulta del estado del temporizador ("MyIEC_TIMER".Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa. Cuando el estado lógico del operando "Tag_Input" cambia a "0", la consulta del estado del temporizador devuelve el estado lógico "0" y el operando "Tag_Output" se desactiva.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

TOF: Arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Arrancar temporizador como retardo a la desconexión" permite arrancar un temporizador CEI como retardo a la desconexión por un tiempo determinado. La consulta de si el estado del temporizador es "1" devuelve el estado lógico "1" si el resultado lógico (RLO) de la entrada de la instrucción devuelve el estado lógico "0". Si el RLO cambia de "1" a "0" (flanco de señal descendente), el temporizador CEI arranca con el tiempo indicado. Mientras el temporizador CEI está activado, el estado del temporizador permanece a "1". Una vez transcurrido el tiempo y si el RLO de la entrada de la instrucción devuelve el estado lógico "0", el estado lógico del temporizador se pone a "0". Si el RLO cambia a "1" antes de transcurrir el tiempo, el temporizador CEI activo se reinicia y el estado lógico del temporizador permanece a "1".

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles secuenciales, pues en cada consulta de las salidas Q o ET se actualiza la estructura IEC_TIMER.

Para CPU S7-1200

La instrucción "Arrancar temporizador como retardo a la desconexión" deposita los datos en una estructura del tipo de datos IEC_TIMER o TOF_TIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Arrancar temporizador como retardo a la desconexión" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME, TOF_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura ET del temporizador CEI. El estado del temporizador se puede consultar con una combinación lógica binaria. La consulta de Q o ET (p. ej. "MyTimer".Q o "MyTimer".ET) actualiza la estructura IEC_TIMER.

Para poder ejecutar la instrucción "Arrancar temporizador como retardo a la desconexión" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetro

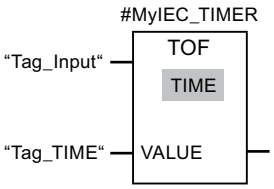
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TOF_TIME	IEC_TIMER, IEC_LTIMER, TOF_TIME, TOF_LTIME	D, L	Temporizador CEI que se arranca.

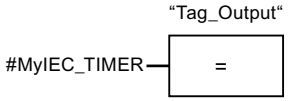
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Arrancar temporizador como retardo a la desconexión" se ejecuta cuando el estado lógico del operando "Tag_Input" cambia de "1" a "0". El temporizador #MyIEC_TIMER se inicia por el tiempo que está guardado en el operando "Tag_TIME".



Mientras se ejecuta el temporizador #MyIEC_TIMER, la consulta del estado del temporizador (#MyIEC_TIMER.Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa. Una vez transcurrido el tiempo y si el operando "Tag_Input" devuelve el estado lógico "0", la consulta del estado del temporizador devuelve el estado lógico "0". Cuando el estado lógico del operando "Tag_Input" cambia a "1" antes de transcurrir el temporizador #MyIEC_TIMER, el temporizador se inicializa. Cuando el estado lógico del operando "Tag_Input" es "1", la consulta del estado del temporizador devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

TONR: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" permite capturar durante cuánto tiempo la señal en la entrada de la instrucción es "1". La instrucción se inicia cuando el resultado lógico (RLO) cambia de "0" a "1" (flanco de señal ascendente). Mientras el RLO sea "1", se cuenta el tiempo. Si el RLO cambia a "0", se detiene la instrucción. Si el RLO cambia nuevamente a "1", se sigue contando el tiempo. La consulta de si el estado del temporizador es "1" devuelve el estado lógico "1" si el tiempo contado supera el valor del tiempo especificado y el RLO a la entrada de la bobina es "1".

El estado del temporizador y el tiempo transcurrido actualmente se pueden poner a "0" con la instrucción "Inicializar temporizador".

Nota

El arranque y la consulta del temporizador CEI pueden encontrarse en diferentes niveles secuenciales, pues en cada consulta de las salidas Q o ET se actualiza la estructura IEC_TIMER.

Para CPU S7-1200

La instrucción "Acumulador de tiempo" deposita los datos en una estructura del tipo de datos IEC_TIMER o TONR_TIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME o IEC_TIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Para CPU S7-1500

La instrucción "Acumulador de tiempo" deposita los datos en una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME. La estructura se puede declarar de la manera siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME, TONR_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede al temporizador asignado.

El estado actual del temporizador se deposita en el elemento de estructura ET del temporizador CEI. El estado del temporizador se puede consultar con una combinación lógica binaria. La consulta de Q o ET (p. ej. "MyTimer".Q o "MyTimer".ET) actualiza la estructura IEC_TIMER.

Para poder ejecutar la instrucción "Acumulador de tiempo" se requiere una operación lógica precedente. Sólo se puede colocar al final del segmento.

Parámetro

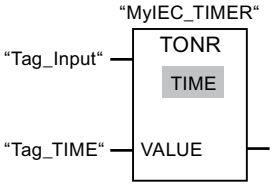
La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
VALUE	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	InOut	IEC_TIMER, TONR_TIME	IEC_TIMER, IEC_LTIMER, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI que se arranca.

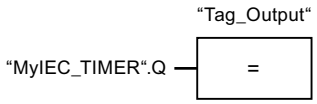
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La instrucción "Acumulador de tiempo" se ejecuta cuando se detecta un flanco de señal ascendente en el RLO. Mientras el operando "Tag_Input" devuelva el estado lógico "1", se cuenta el tiempo.



Si el tiempo contado supera el valor del operando "Tag_TIME", la consulta del estado del temporizador ("MyIEC_TIMER".Q) devuelve el estado lógico "1" y el operando "Tag_Output" se activa.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

RT: Inicializar temporizador (Página 3634)

RT: Inicializar temporizador

Descripción

La instrucción "Inicializar temporizador" permite poner a "0" un temporizador CEI. Para indicar el temporizador CEI que debe reiniciarse, introduzca en el comodín ubicado sobre la instrucción el nombre del bloque de datos que contiene la estructura del temporizador CEI.

La instrucción se ejecuta sólo si el resultado lógico (RLO) de la entrada del cuadro es "1". Al ejecutarse la función se ponen a "0" los elementos de estructura del temporizador CEI en el bloque de datos indicado. Si el RLO de la entrada del cuadro es "0", no se ejecuta la instrucción.

La instrucción no afecta al RLO. El RLO de la entrada del cuadro se transfiere directamente a la salida.

A la instrucción "Inicializar temporizador" debe asignársele un temporizador CEI declarado en el programa.

Los datos de la instrucción se actualizan únicamente al efectuar una llamada de la instrucción, y no cada vez que se accede al temporizador CEI asignado. La consulta de los datos es sólo igual desde la llamada de la instrucción hasta la siguiente llamada de la instrucción.

Parámetros

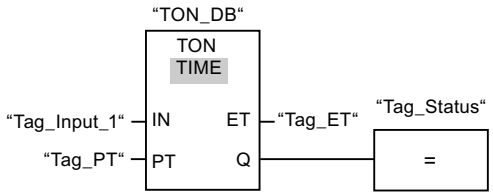
La tabla siguiente muestra los parámetros de la instrucción "Inicializar temporizador":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
<Temporizador CEI>	InOut	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI que se reinicia.

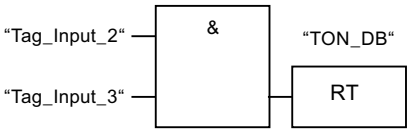
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "Tag_Input_1" cambia de "0" a "1", se ejecuta la instrucción "Retardo al conectar". El temporizador CEI depositado en el bloque de datos de instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PT".



Si los operandos "Tag_Input_2" y "Tag_Input_3" devuelven el estado lógico "1", se ejecuta la instrucción "Inicializar temporizador" y se reinicia el temporizador CEI depositado en el bloque de datos "TON_DB".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

PT: Cargar tiempo

Descripción

La instrucción "Cargar tiempo" parametriza la duración de un temporizador CEI. La instrucción se ejecuta en cada ciclo cuando el resultado lógico (RLO) de la entrada del cuadro de la instrucción tiene el estado lógico "1". La instrucción escribe el tiempo indicado en la estructura del temporizador CEI indicado.

Nota

Si el temporizador CEI indicado cuenta el tiempo mientras se ejecuta la instrucción, ésta sobrescribe el tiempo actual del temporizador CEI indicado. Esto puede modificar el estado del temporizador CEI.

A la instrucción "Cargar tiempo" debe asignársele un temporizador CEI declarado en el programa.

Los datos de la instrucción se actualizan al efectuar una llamada de la instrucción, y cada vez que se accede al temporizador CEI asignado. La consulta de Q o ET (p. ej. "MyTimer".Q o "MyTimer".ET) actualiza la estructura IEC_TIMER.

Parámetro

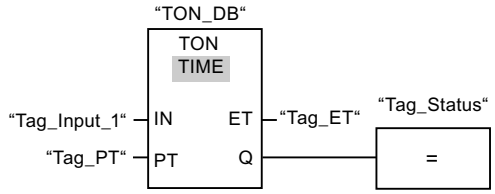
La tabla siguiente muestra los parámetros de la instrucción "Cargar tiempo":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
PT	Input	TIME	TIME, LTIME	I, Q, M, D, L o constante	Tiempo
<Temporizador CEI>	InOut	IEC_TIMER, TP_TIME, TON_TIME, TOF_TIME, TONR_TIME	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI cuyo tiempo se ajusta.

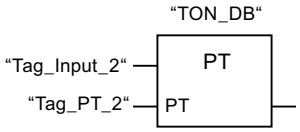
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "Tag_Input_1" cambia de "0" a "1", se ejecuta la instrucción "Retardo al conectar". El temporizador CEI depositado en el bloque de datos de instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PT".



Cuando el estado lógico del operando "Tag_Input_2" es "1", se ejecuta la instrucción "Cargar tiempo". La instrucción escribe el tiempo "Tag_PT_2" en el bloque de datos de instancia "TON_DB" y sobrescribe a la vez el valor del operando "Tag_PT" en el bloque de datos. Esto

puede hacer que pueda modificarse el estado lógico del temporizador en la siguiente consulta o bien al acceder a "MyTimer".Q o "MyTimer".ET.

Nota

"Tag_Input_2" se ejecuta como marca de impulso para poder efectuar la carga de tiempo sólo durante un ciclo del programa.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Temporizadores SIMATIC

S_PULSE: Parametrizar y arrancar temporizador como impulso

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Si el estado lógico en la entrada S cambia a "0" antes de que transcurra el tiempo programado, se detiene el temporizador. En tal caso, el estado lógico de la salida Q es "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

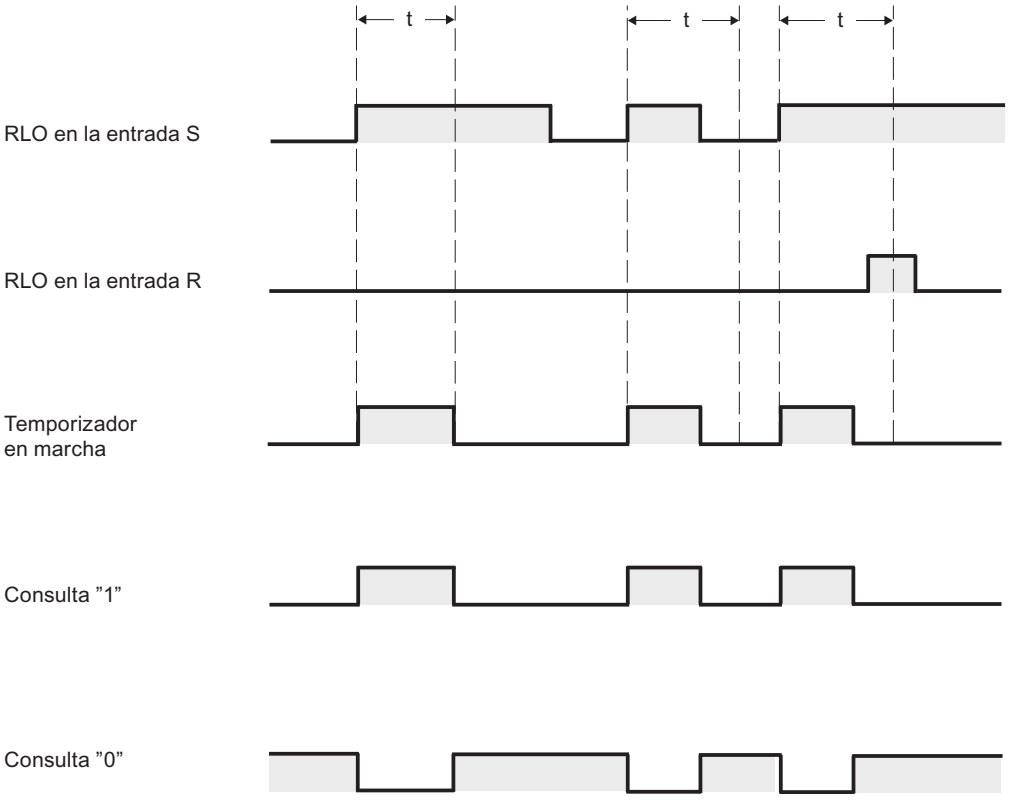
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

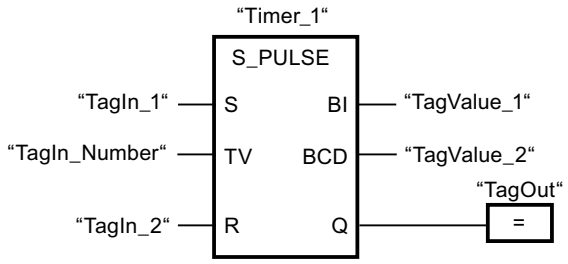
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador "Timer_1" se detiene. En ese caso, el operando "TagOut" se pone a "0".

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha y el operando "TagIn_1" tenga el estado lógico "1". Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso prolongado" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Mientras se ejecuta el temporizador, la salida Q devuelve el estado lógico "1". Una vez transcurrido el tiempo, la salida Q se pone a "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, este se reinicia con el tiempo programado en la entrada TV.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

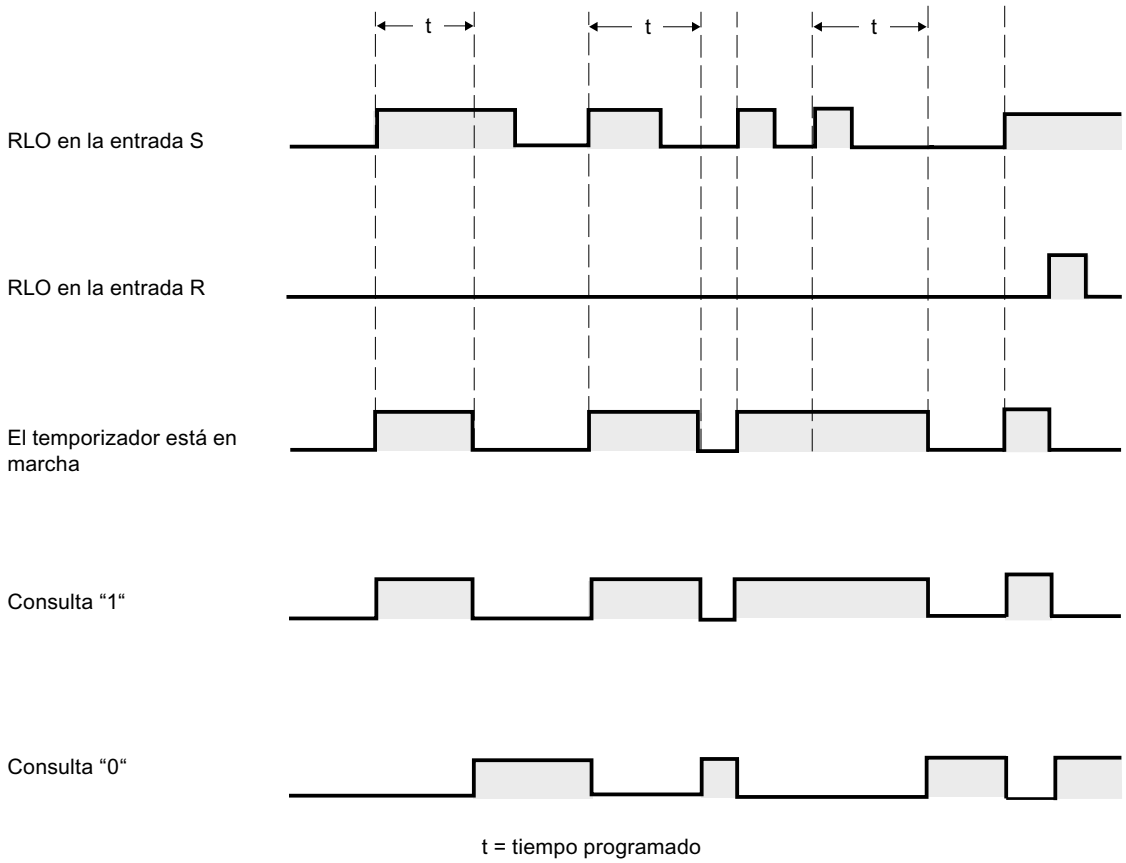
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

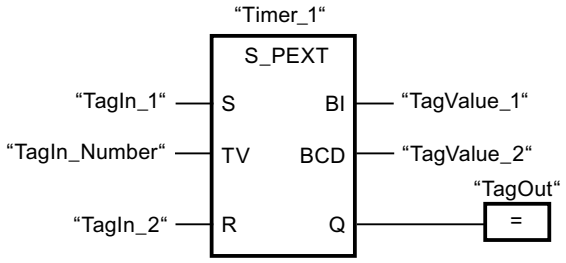
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en la entrada S. Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha. Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" se arranca un temporizador programado como retardo a la conexión al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Una vez transcurrido el tiempo debidamente y si la entrada S aún tiene el valor lógico "1", la salida Q devuelve el valor lógico "1". Si el estado de la entrada S cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En este caso, la salida Q adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia de "0" a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. En ese caso, el estado lógico de la salida Q es "0". El temporizador se reinicia cuando el estado lógico de la entrada R es "1", incluso si el temporizador no se ejecuta y el RLO en la entrada S es "1".

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

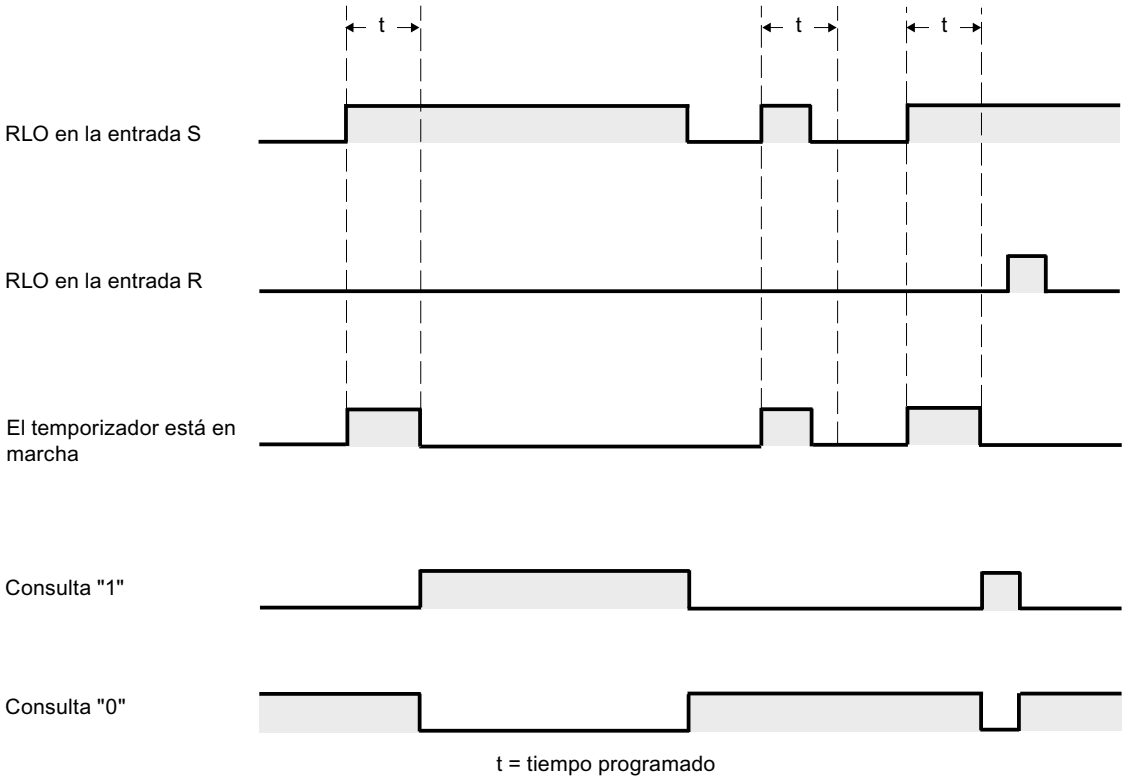
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

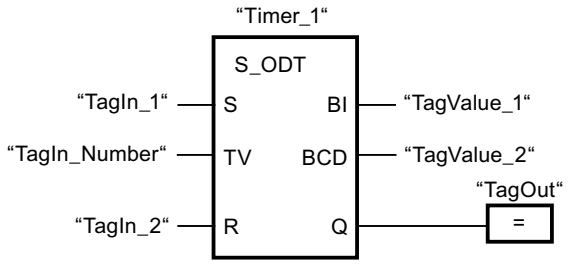
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el operando tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. El operando "TagOut" devuelve en ese caso el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Una vez transcurrido el tiempo, la salida Q devuelve el estado lógico "1" independientemente del estado lógico de la entrada S. Si el estado lógico de la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, el temporizador vuelve a arrancar con el tiempo programado (TV).

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" de la entrada R pone a "0" el valor actual de tiempo y la base de tiempo, independientemente del estado lógico de la entrada de arranque S. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

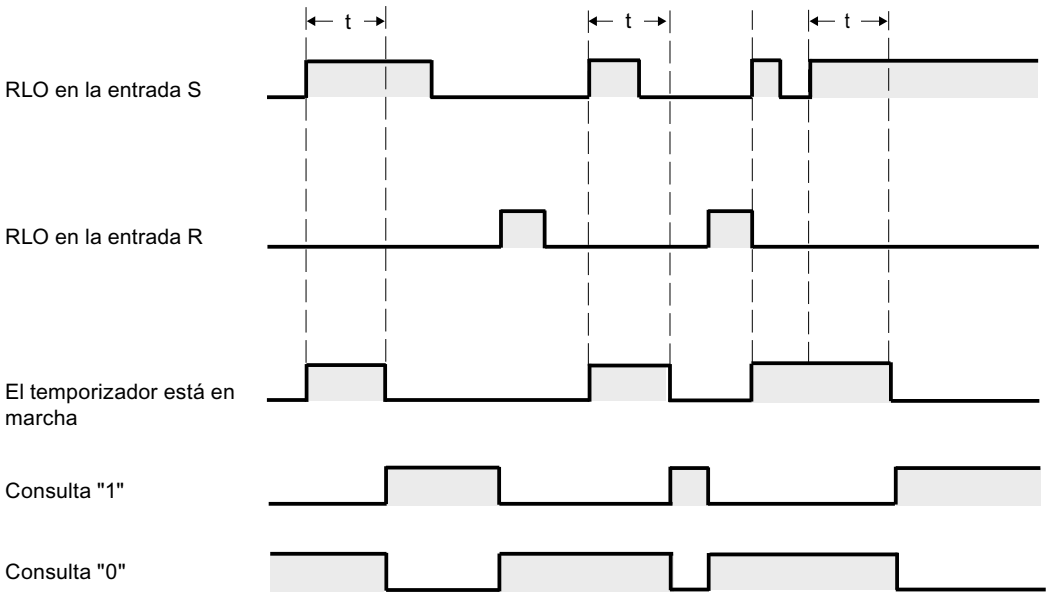
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

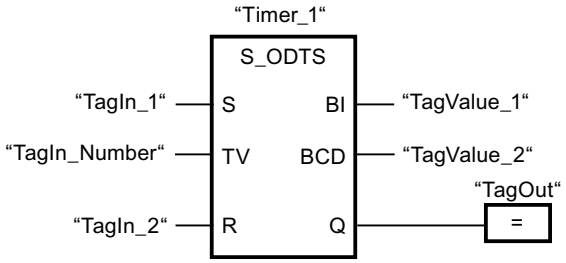
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number" aunque el estado lógico del operando "TagIn_1" cambie a "0". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente) en la entrada S. El temporizador empieza a contar con el tiempo programado (TV). Mientras el temporizador está en marcha o la entrada S devuelve el estado lógico "1", la salida Q tiene el estado lógico "1". Si el tiempo ha transcurrido y el estado lógico es "0", la salida Q adopta el valor lógico "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras el temporizador está funcionando, este se detiene. El temporizador no vuelve a arrancar hasta que no se detecta un flanco de señal descendente en la entrada S.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" en la entrada R pone a "0" el valor de tiempo y la base de tiempo actuales. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

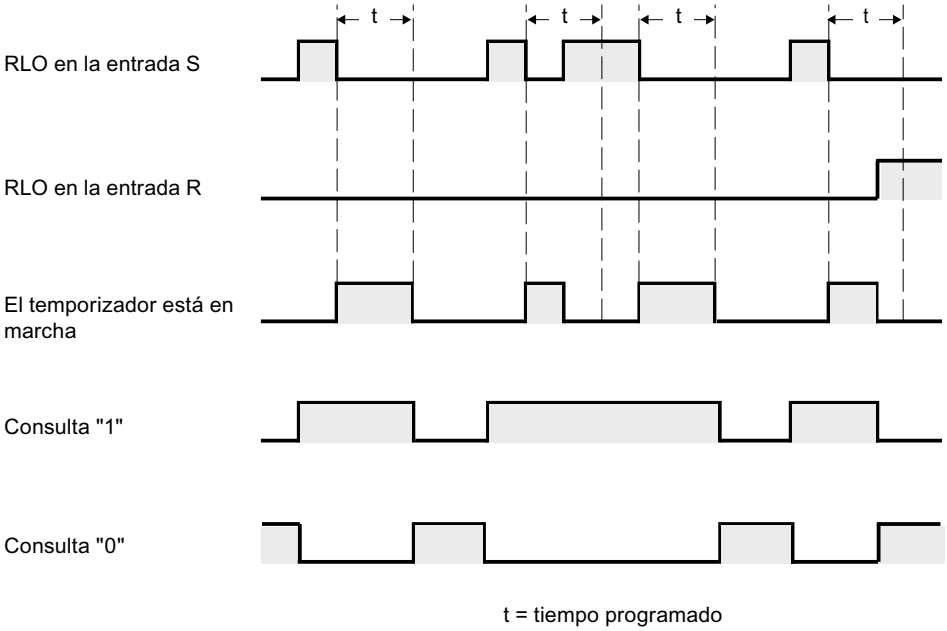
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P o constante	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

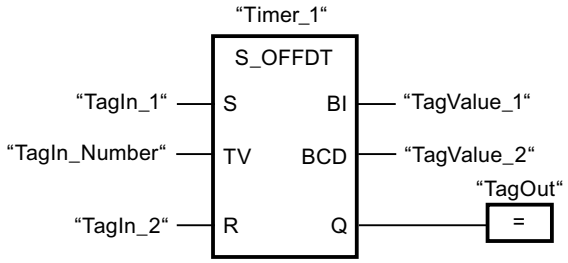
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador está en marcha o el operando "TagIn_1" tiene el estado lógico "0". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

SP: Arrancar temporizador como impulso

Descripción

La instrucción "Arrancar temporizador como impulso" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador cuenta el tiempo indicado mientras el RLO devuelva el estado lógico "1". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" antes de que haya transcurrido el valor de tiempo, el temporizador se detiene. En ese caso, una consulta de si el estado del temporizador es "1" da como resultado "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

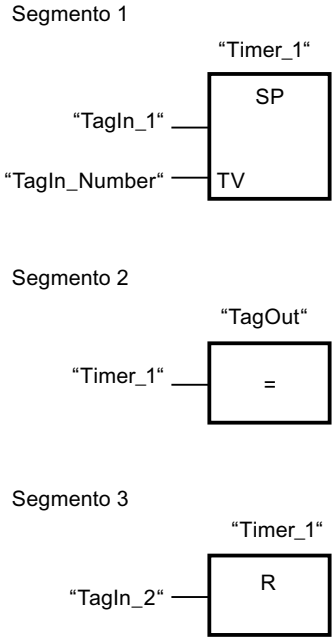
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

SE: Arrancar temporizador como impulso prolongado

Descripción

La instrucción "Arrancar temporizador como impulso prolongado" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras el temporizador está funcionando, este se arranca de nuevo con el tiempo programado. La consulta sobre si el estado es "1" da como resultado "0" si el tiempo ha transcurrido.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

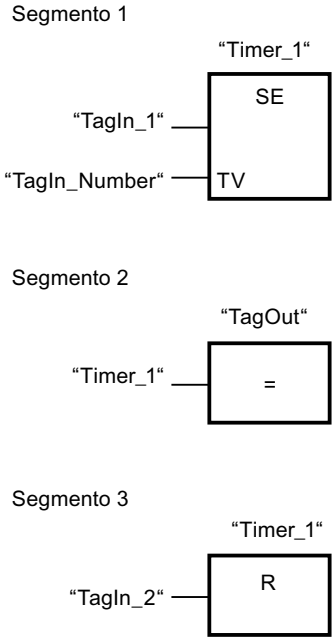
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en el RLO. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

SD: Arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador empieza a contar con el tiempo indicado y sigue contando mientras el RLO sea "1". Una vez transcurrido el tiempo y si el RLO todavía tiene el estado lógico "1", una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En ese caso, la consulta de si el estado del temporizador es "1" devuelve el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

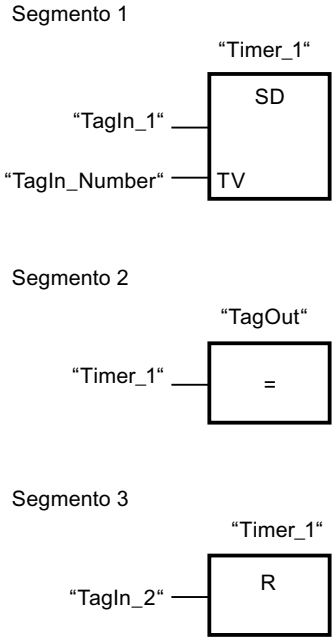
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el RLO tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

SS: Arrancar temporizador como retardo a la conexión con memoria

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Si ha transcurrido el tiempo, una consulta de si el estado del temporizador es "1" da como resultado "1". Una vez transcurrido el tiempo, el temporizador no puede volver a ponerse en marcha hasta que no se haya reiniciado explícitamente.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

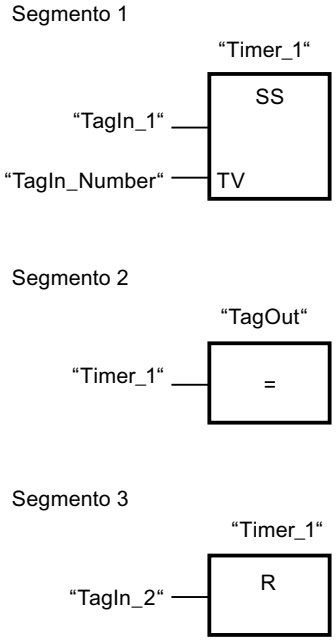
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

SF: Arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Arrancar temporizador como retardo a la desconexión" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente) en la entrada de arranque. El temporizador empieza a contar con el tiempo indicado. Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras se ejecuta el temporizador, este se inicializa. El temporizador siempre arranca de nuevo cuando el RLO cambia de "1" a "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

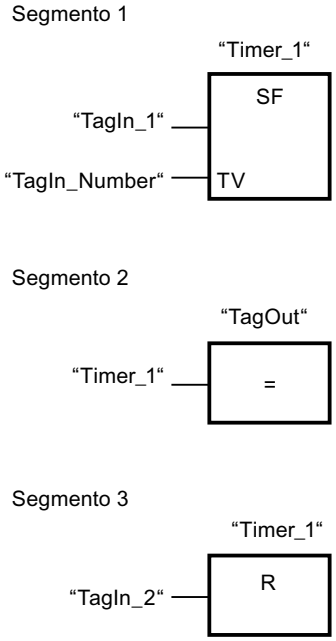
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	InOut/Input	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Mientras el temporizador está en marcha, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 4093)

Contadores

Contadores CEI

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor actual de contaje de la salida CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Cuando se alcanza el valor límite superior, el estado lógico de la entrada CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. El estado lógico de la salida Q es determinado por el parámetro PV. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0". En el parámetro PV también se puede indicar una constante.

El valor de la salida CV se pone a "0" y se guarda en una marca de flancos, cuando el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de la entrada CU no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Para CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<Contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTU_<Tipo de datos> o IEC_<Contador> en la sección "Static" de un bloque (p. ej. #MyIEC_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador ascendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

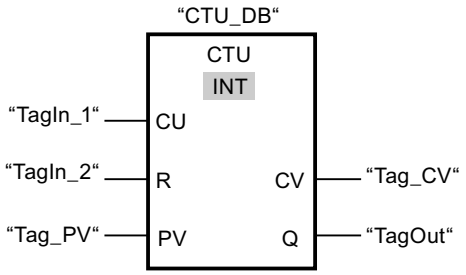
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
CU	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de reset
PV	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor actual de contaje del operando "Tag_CV" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos indicado (INT = 32767).

El valor del parámetro PV se utiliza como límite para determinar la salida "TagOut". La salida "TagOut" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor del operando "Tag_PV". En los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor de la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos indicado. Cuando se alcanza el valor límite inferior, el estado lógico de la entrada CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. Si el valor actual de contaje es menor o igual a "0", la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0". En el parámetro PV también se puede indicar una constante.

El valor de la salida CV se pone al valor del parámetro PV y se guarda en una marca de flancos, cuando el estado lógico de la entrada LD cambia de "0" a "1". Mientras la entrada LD tenga el estado lógico "1", el estado lógico de la entrada CD no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Para CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • CTD_LINT / CTD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<Contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTD_<Tipo de datos> o IEC_<Contador> o en la sección "Static" de un bloque (p. ej. #MyIEC_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

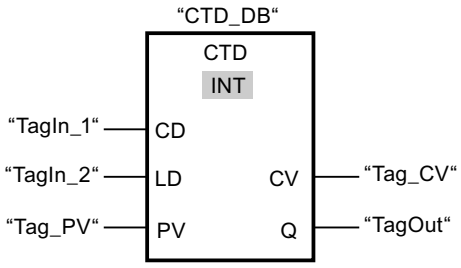
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
CD	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador descendente" y el valor de la salida "Tag_CV" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos indicado (INT = -32768).

El valor del parámetro PV se utiliza como límite para determinar la salida "TagOut". La salida "TagOut" devolverá el estado lógico "1" mientras el valor actual de contaje sea menor o igual a "0". En los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Editar elementos FUP (Página 2738)

Insertar elementos FUP (Página 2722)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno y se deposita en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor actual de contaje de la salida CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor actual de contaje no se modifica en la salida CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado, ya no se decrementa el valor de contaje.

Si el estado lógico de la entrada LD cambia a "1", el valor de contaje de la salida CV adopta el valor del parámetro PV y se guarda en una marca de flancos. Mientras la entrada LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a "0" y se guarda en una marca de flancos, cuando el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", un cambio del estado lógico de las entradas CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en la salida QU. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida QU adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QU es "0". En el parámetro PV también se puede indicar una constante.

El estado del contador descendente se puede consultar en la salida QD. Si el valor actual de contaje es menor o igual a cero, la salida QD adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QD es "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER

Para CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<Contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • CTUD_LINT / CTUD_ULINT • IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<Contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTUD_<Tipo de datos> o IEC_<Contador> en la sección "Static" de un bloque (p. ej. #MyIEC_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador ascendente - descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

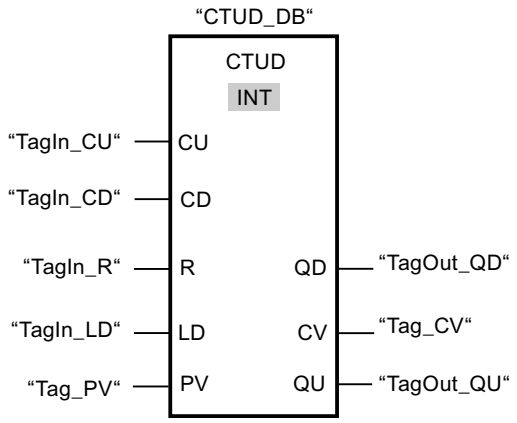
Parámetros	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
CU	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P o constante	I, Q, M, D, L, T, C, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU.
QU	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Contador ascendente de estado
QD	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Contador descendente de estado
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_CU" o "TagIn_CD" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Contador ascendente - descendente". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_CU", el valor de contaje actual se incrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada "TagIn_CD", el valor de contaje se decrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior (INT = 32767). Si se detecta un flanco de señal ascendente en la entrada CD, el valor de contaje se va decrementando hasta alcanzar el valor límite inferior de (INT = -32768).

La salida "TagOut_GU" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor de la entrada "Tag_PV". En los demás casos, la salida "TagOut_QU" devuelve el estado lógico "0".

La salida "TagOut_QD" devolverá el estado lógico "1" mientras el valor actual de contaje sea menor o igual a "0". En todos los demás casos, la salida "TagOut_QD" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3218)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Contadores SIMATIC

S_CUD: Parametrizar e incrementar/decrementar contador

Descripción

La instrucción "Parametrizar e incrementar/decrementar contador" permite incrementar y decrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje no se modifica.

El valor de contaje sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior "0", no se decrementa más el valor de contaje.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si se inicializa el contador y el RLO en las entradas CU y CD es "1", el contador solo cuenta una vez en el próximo ciclo aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU, CD y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar/decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetro

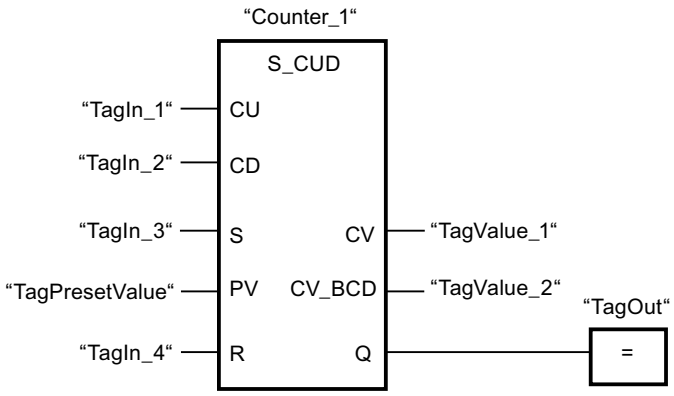
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar/ decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L, T, C	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (formato BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_1" o "TagIn_2" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Parametrizar e incrementar/decrementar contador". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_1" y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_2" y el valor de contaje actual es mayor que "0", este valor se decrementará en uno.

Si el estado lógico de la entrada "TagIn_3" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_4" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

S_CU: Parametrizar e incrementar contador

Descripción

La instrucción "Parametrizar e incrementar contador" permite incrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue incrementando hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CU es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetro

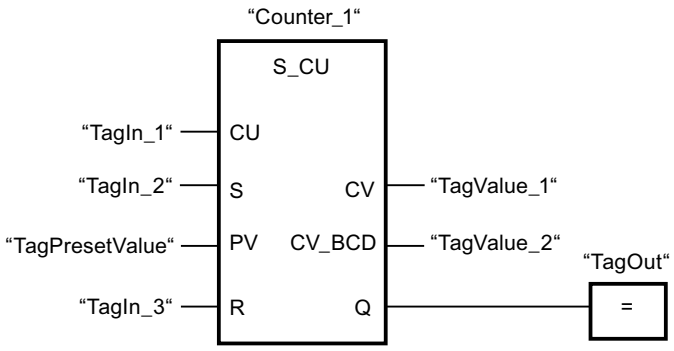
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L, T, C	Entrada de contaje ascendente
S	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (formato BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

S_CD: Parametrizar y decrementar contador

Descripción

La instrucción "Parametrizar y decrementar contador" permite decrementar el valor de un contador. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite inferior, el valor de contaje no se decrementa más al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar y decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetro

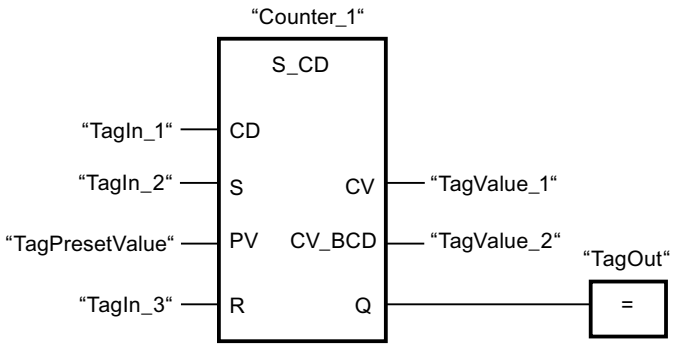
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CD	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C o constante	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L	Valor de contaje actual (formato BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es mayor que "0", este valor se decrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SC: Poner contador al valor inicial

Descripción

Con la instrucción "Poner contador al valor inicial" se ajusta el valor de un contador. La instrucción se ejecuta cuando el resultado lógico (RLO) de la entrada de arranque de la instrucción cambia de "0" a "1". Si se ejecuta la instrucción, el contador se pone al valor de contaje indicado.

La instrucción "Poner contador al valor inicial" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

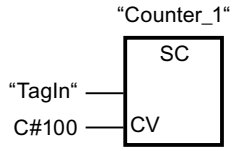
La tabla siguiente muestra los parámetros de la instrucción "Poner contador al valor inicial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	Entrada de arranque
PV	Input	WORD	I, Q, M, D, L o constante	Valor con el que se preajusta el contador codificado en BCD. (C#0 a C#999)
<Contador>	InOut/Input	COUNTER	C	Contador que se preajusta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn" cambia de "0" a "1", el contador "Counter_1" se pone a "100".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

CU: Contador ascendente

Descripción

La instrucción "Contador ascendente" permite aumentar en uno el valor del contador indicado cuando hay un flanco de señal ascendente en la entrada de arranque. El valor de contaje puede seguir incrementándose hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

La instrucción "Contador ascendente" requiere una operación lógica precedente para la evaluación de flancos y sólo se puede colocar en el margen derecho del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

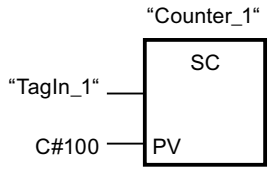
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	Entrada de arranque
<Contador>	InOut/Input	COUNTER	C	Contador cuyo valor se aumenta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

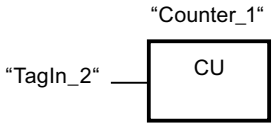
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

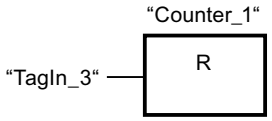
Segmento 1



Segmento 2



Segmento 3



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se incrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

CD: Contador descendente

Descripción

La instrucción "Contador descendente" permite reducir en uno el valor del contador indicado cuando hay un flanco de señal ascendente en la entrada de arranque. El valor de contaje puede seguir decrementándose hasta alcanzar el límite "0". Una vez alcanzado el valor límite, el valor de contaje no se modifica al detectarse un flanco de señal ascendente.

La instrucción "Contador descendente" requiere una operación lógica precedente para la evaluación de flancos y sólo se puede colocar en el margen derecho del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

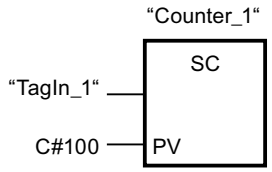
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	Entrada de arranque
<Contador>	InOut/Input	COUNTER	C	Contador cuyo valor se decrementa.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

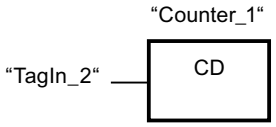
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

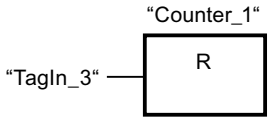
Segmento 1



Segmento 2



Segmento 3



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se decrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Comparación

CMP ==: Igual

Descripción

La instrucción "Igual" permite consultar si el valor de la entrada IN1 es igual al valor de la entrada IN2.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

IN1	IN2	RLO de la instrucción
'AA'	'AA'	1
'Hello World'	'HelloWorld'	0
'AA'	'aa'	0

La instrucción "Igual" también compara caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Si está activada la verificación CEI, los operandos que comparar deben ser del mismo tipo de datos. Si no está activada la verificación CEI, el ancho (longitud) de los operandos debe ser igual. Si se comparan números en coma flotante, los operandos que comparar deben ser del mismo tipo de datos, independientemente del ajuste de la verificación CEI.

Nota

Comparación de números en coma flotante

Si desea comparar tipos de datos REAL o LREAL, utilice en lugar de la instrucción "CMP ==: Igual" la instrucción "IN_RANGE: Valor dentro del rango".

Nota

Comparación del tipo de datos PORT

Para poder comparar operandos del tipo de datos PORT con la instrucción "Igual", deberá seleccionar el tipo de datos WORD en la lista desplegable del cuadro de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Igual":

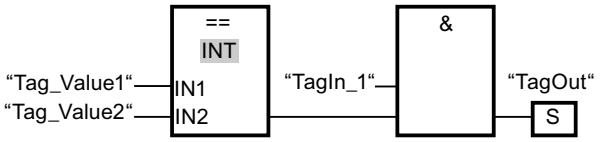
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" = "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CMP <>: Diferente

Descripción

La instrucción "Diferente" permite consultar si el valor de la entrada IN1 es distinto al valor de la entrada IN2.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

IN1	IN2	RLO de la instrucción
'AA'	'aa'	1
'Hello World'	'HelloWorld'	1
'AA'	'AA'	0

La instrucción "Diferente" también compara caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Si está activada la verificación CEI, los operandos que comparar deben ser del mismo tipo de datos. Si no está activada la verificación CEI, el ancho (longitud) de los operandos debe ser igual. Si se comparan números en coma flotante, los operandos que comparar deben ser del mismo tipo de datos, independientemente del ajuste de la verificación CEI.

Nota

Comparación del tipo de datos PORT

Para poder comparar operandos del tipo de datos PORT con la instrucción "Diferente", deberá seleccionar el tipo de datos WORD en la lista desplegable del cuadro de la instrucción.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Diferente":

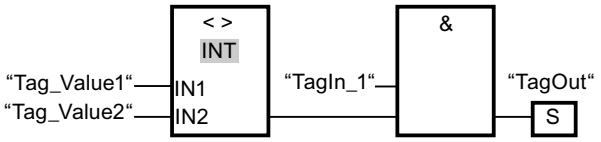
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Secuencias de bits, enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <> "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CMP >=: Mayor o igual

Descripción

La instrucción "Mayor o igual" permite consultar si el valor de la entrada IN1 es mayor o igual que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más larga se considera la mayor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

IN1	IN2	RLO de la instrucción
'BB'	'AA'	1
'AAA'	'AA'	1
'Hello World'	'Hello World'	1
'Hello World'	'HelloWorld'	0
'AA'	'aa'	0
'AAA'	'a'	0

La instrucción "Mayor o igual" también compara caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante de la entrada IN1 es mayor (más reciente) o igual que el instante de la entrada IN2.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor o igual":

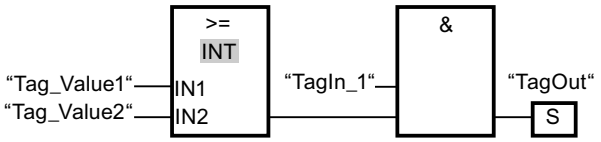
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" >= "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3218)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CMP <=: Menor o igual

Descripción

La instrucción "Menor o igual" permite consultar si el valor de la entrada IN1 es menor o igual que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más corta se considera la menor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

IN1	IN2	RLO de la instrucción
'AA'	'aa'	1
'AAA'	'a'	1
'Hello World'	'Hello World'	1
'HelloWorld'	'Hello World'	0
'BB'	'AA'	0
'AAA'	'AA'	0

La instrucción "Menor o igual" también compara caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante de la entrada IN1 es menor (más antiguo) o igual que el instante de la entrada IN2.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor o igual":

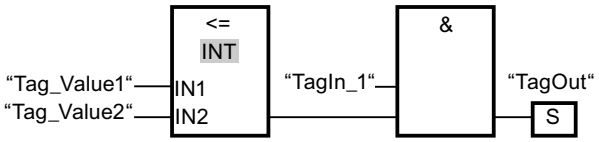
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <= "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CMP >: Mayor

Descripción

La instrucción "Mayor" permite consultar si el valor de la entrada IN1 es mayor que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más larga se considera la mayor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

IN1	IN2	RLO de la instrucción
'BB'	'AA'	1
'AAA'	'AA'	1
'AA'	'aa'	0
'AAA'	'a'	0

La instrucción "Mayor" también compara caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante de la entrada IN1 es mayor (más reciente) que el instante de la entrada IN2.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor":

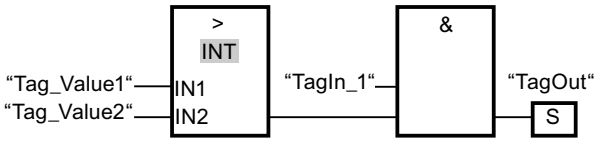
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" > "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CMP <: Menor

Descripción

La instrucción "Menor" permite consultar si el valor de la entrada IN1 es menor que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

En una comparación de cadenas de caracteres, los distintos caracteres se comparan según su codificación (p. ej. 'a' es mayor que 'A'). La comparación se realiza de izquierda a derecha. El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la más corta se considera la menor de ellas.

La tabla siguiente muestra ejemplos de comparación de cadenas de caracteres:

<Operando1>	<Operando2>	RLO de la instrucción
'AA'	'aa'	1
'AAA'	'a'	1
'BB'	'AA'	0
'AAA'	'AA'	0

La instrucción "Menor" también compara caracteres individuales de una cadena de caracteres. El número del carácter que se va a comparar se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se compara p. ej. el segundo carácter de la cadena de caracteres "MyString".

Al comparar valores de tiempo, el RLO de la instrucción es "1" si el instante de la entrada IN1 es menor (más antiguo) que el instante de la entrada IN2 .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor":

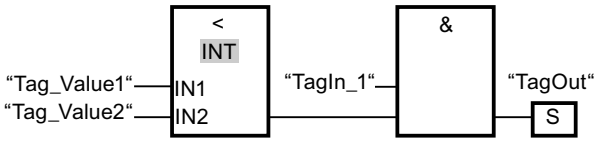
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante, cadenas de caracteres, TIME, DATE, TOD, DTL	Enteros, números en coma flotante, cadenas de caracteres, TIME, LTIME, DATE, TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" < "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3218)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

IN_RANGE: Valor dentro del rango

Descripción

La instrucción "Valor dentro del rango" permite consultar si el valor de la entrada VAL está dentro de un determinado rango de valores.

Los límites del rango de valores se determinan mediante las entradas MIN y MAX. La instrucción "Valor dentro del rango" compara el valor de la entrada VAL con los valores de las entradas MIN y MAX, y transfiere el resultado a la salida del cuadro. Si el valor de la entrada VAL cumple la comparación MIN <= VAL o VAL <= MAX la salida del cuadro devuelve el estado lógico "1". Si no se cumple la comparación, la salida del cuadro devuelve el estado lógico "0".

Para ejecutar la función de comparación es necesario que los valores que deben compararse sean del mismo tipo de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Valor dentro del rango":

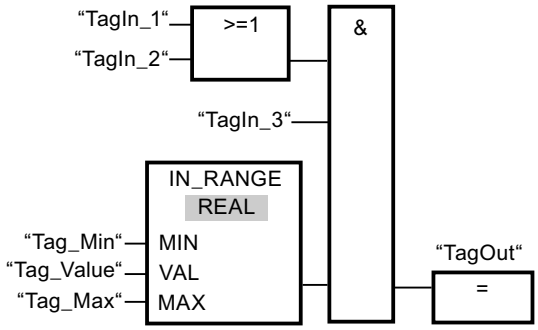
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VAL	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Valor de comparación
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
Salida del cuadro	Output	BOOL	I, Q, M, D, L	Resultado de la comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" o "TagIn_2" devuelven el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "1".
- El valor del operando "Tag_Value" está dentro del rango de valores especificado por los valores actuales de los operandos "Tag_Min" y "Tag_Max" (MIN <= VAL o VAL <= MAX).

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

OUT_RANGE: Valor fuera del rango

Descripción

La instrucción "Valor fuera del rango" permite consultar si el valor de la entrada VAL está fuera de un determinado rango de valores.

Los límites del rango de valores se determinan mediante las entradas MIN y MAX. La instrucción "Valor fuera del rango" compara el valor de la entrada VAL con los valores de las entradas MIN y MAX , y transfiere el resultado a la salida del cuadro. Si el valor de la entrada VAL cumple la comparación MIN > VAL o VAL > MAX la salida del cuadro devuelve el estado lógico "1". La salida del cuadro también devuelve el estado lógico "1" si un operando indicado del tipo de datos REAL contiene un valor no válido.

La salida del cuadro devuelve el estado lógico "0" si el valor de la entrada VAL no cumple la condición MIN > VAL o VAL > MAX .

Para ejecutar la función de comparación es necesario que los valores que deben compararse sean del mismo tipo de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Valor fuera del rango":

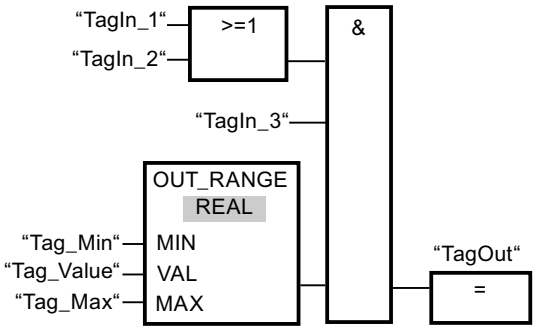
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VAL	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Valor de comparación
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
Salida del cuadro	Output	BOOL	I, Q, M, D, L	Resultado de la comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "1".
- El valor del operando "Tag_Value" está fuera del rango de valores especificado por los valores de los operandos "Tag_Min" y "Tag_Max" (MIN > VAL o VAL > MAX).

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

OK: Comprobar validez

Descripción

La instrucción "Comprobar validez" permite consultar si el valor de un operando (<Operando>) es un número en coma flotante válido. La verificación se realiza en cada ciclo del programa. Si el valor del operando es un número en coma flotante válido cuando se realiza la consulta, la salida del cuadro devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción "Comprobar validez" es "0".

La instrucción "Comprobar validez" puede utilizarse en combinación con el mecanismo EN. Si el cuadro de la instrucción se conecta a una entrada de habilitación EN, ésta se activa únicamente si la consulta de la validez del valor da un resultado positivo. Esta función garantiza que una instrucción se active sólo si el valor del operando indicado es un número en coma flotante válido.

Parámetros

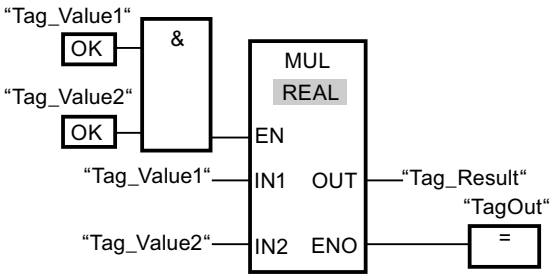
El ejemplo siguiente muestra el funcionamiento de la instrucción "Comprobar validez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L	Valor que se comprueba.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los valores de los operandos "Tag_Value1" y "Tag_Value2" contienen números en coma flotante válidos, se ejecuta la instrucción "Multiplicar" (MUL) y se activa la salida de habilitación ENO. Cuando se ejecuta la instrucción "Multiplicar" (MUL), el valor del operando "Tag_Value1" se multiplica por el valor del operando "Tag_Value2". El resultado de la multiplicación se almacena entonces en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, las salidas ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

NOT_OK: Comprobar invalidez

Descripción

La instrucción "Comprobar invalidez" permite consultar si el valor de un operando (<Operando>) es un número en coma flotante no válido. La verificación se realiza en cada ciclo del programa. Si el valor del operando es un número en coma flotante no válido cuando se realiza la consulta, la salida del cuadro devuelve el estado lógico "1". En los restantes casos, el estado lógico de la salida del cuadro es "0".

Parámetros

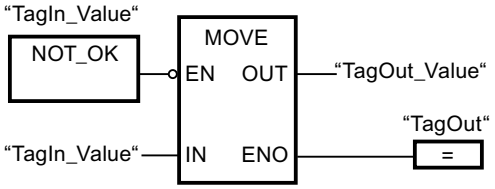
La tabla siguiente muestra los parámetros de la instrucción "Comprobar invalidez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L	Valor que se comprueba.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el valor del operando "TagIn_Value" no es un número en coma flotante válido, no se ejecuta la instrucción "Copiar valor" (MOVE). La salida "TagOut" se pone al estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Instrucciones VARIANT

EQ_Type: Comparar si el tipo de datos es IGUAL al tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos es IGUAL al tipo de datos de una variable" se consulta qué tipo de datos tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable es "Igual" al tipo de datos de una variable declarada en la interfaz del bloque.

El parámetro IN1 tiene que ser del tipo de datos VARIANT. El parámetro IN2 puede ser un tipo de datos simple o un tipo de datos PLC.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetro

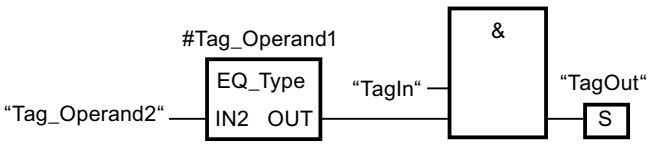
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos es IGUAL al tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	VARIANT	L	Primer operando
IN2	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es igual a "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

NE_Type: Comparar si el tipo de datos es DIFERENTE al tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos es DIFERENTE del tipo de datos de una variable" se consulta qué tipo de datos no tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable es "Diferente" del tipo de datos de una variable declarada en la interfaz del bloque.

El parámetro IN1 tiene que ser del tipo de datos VARIANT. El parámetro IN2 puede ser un tipo de datos simple o un tipo de datos PLC.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetro

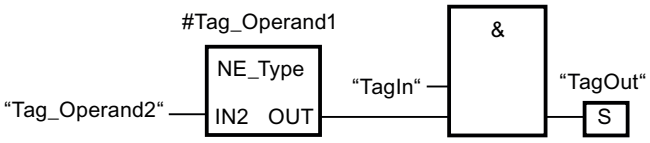
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos es DIFERENTE del tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	VARIANT	L	Primer operando
IN2	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es diferente de "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

EQ_ElemType: Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable" se consulta qué tipo de datos tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable es "Igual" al tipo de datos de una variable declarada en la interfaz del bloque.

El parámetro IN1 tiene que ser del tipo de datos VARIANT. El parámetro IN2 puede ser un tipo de datos simple o un tipo de datos PLC.

Si el tipo de datos de la variable VARIANT es un ARRAY, se compara el tipo de datos de los elementos ARRAY.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetro

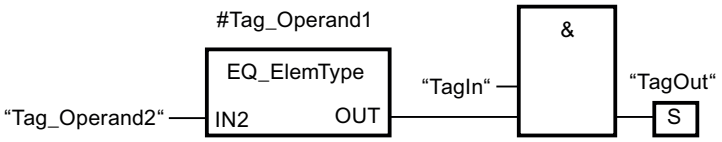
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	VARIANT	L	Primer operando
IN2	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es igual a "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

NE_ElemType: Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable

Descripción

Con la instrucción "Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable" se consulta qué tipo de datos no tiene una variable a la que apunta un VARIANT. Se compara si el tipo de datos de la variable es "Diferente" del tipo de datos de una variable declarada en la interfaz del bloque.

El parámetro IN1 tiene que ser del tipo de datos VARIANT. El parámetro IN2 puede ser un tipo de datos simple o un tipo de datos PLC.

Si el tipo de datos de la variable VARIANT es un ARRAY, se compara el tipo de datos de los elementos ARRAY.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetro

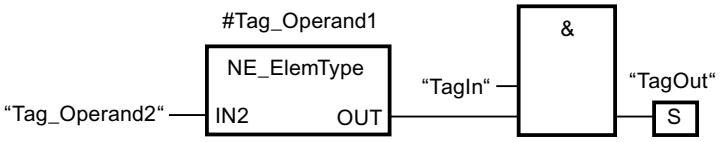
La tabla siguiente muestra los parámetros de la instrucción "Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	VARIANT	L	Primer operando
IN2	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, cadenas de caracteres, ARRAY, tipos de datos PLC	I, Q, M, D, L	Segundo operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand1" es diferente de "Tag_Operand2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

IS_NULL: Comparar si es IGUAL A CERO

Descripción

La instrucción "Comparar si es IGUAL A CERO" permite saber si VARIANT no apunta a un objeto.

El <Operando1> tiene que ser del tipo VARIANT.

Nota

La variable VARIANT apunta a un puntero ANY

Si la variable VARIANT apunta a un puntero ANY, la instrucción devuelve siempre el resultado RLO = "0" aunque el puntero ANY tenga el valor CERO.

Parámetro

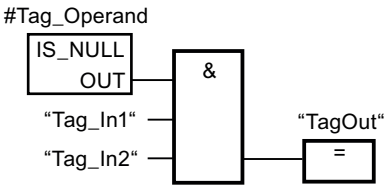
La tabla siguiente muestra los parámetros de la instrucción "Comparar si es IGUAL A CERO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	VARIANT	L	Operando que se compara si es IGUAL A CERO

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "Tag_In1" y "Tag_In2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand" no apunta a ningún objeto.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

NOT_NULL: Comparar si es DIFERENTE DE CERO

Descripción

La instrucción "Comparar si es DIFERENTE DE CERO" permite saber si VARIANT apunta a un objeto.

El <Operando1> tiene que ser del tipo VARIANT.

Nota

La variable VARIANT apunta a un puntero ANY

Si la variable VARIANT apunta a un puntero ANY, la instrucción devuelve siempre el resultado RLO = "1" aunque el puntero ANY tenga el valor CERO.

Parámetro

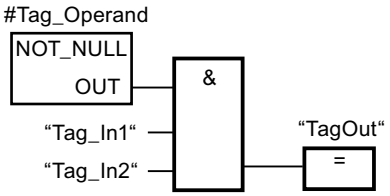
La tabla siguiente muestra los parámetros de la instrucción "Comparar si es DIFERENTE DE CERO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	VARIANT	L	Operando que se compara si es DIFERENTE DE CERO

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "Tag_In1" y "Tag_In2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_Operand" apunta a un objeto.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

IS_ARRAY: Consultar si es un ARRAY

Descripción

La instrucción "Consultar si es un ARRAY" permite saber si VARIANT apunta a una variable del tipo de datos ARRAY.

El <Operando1> tiene que ser del tipo VARIANT.

Parámetro

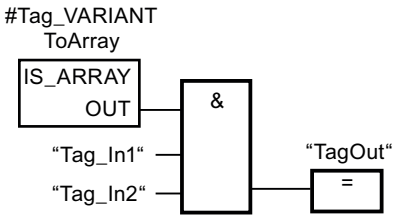
La tabla siguiente muestra los parámetros de la instrucción "Consultar si es un ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	VARIANT	L	Operando que se consulta si es un ARRAY

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "Tag_In1" y "Tag_In2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación, es decir, el operando "#Tag_VARIANTToArray" es del tipo de datos ARRAY.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Funciones matemáticas

CALCULATE: Calcular

Descripción

La instrucción "Calcular" permite definir y ejecutar una expresión para calcular operaciones matemáticas o combinaciones lógicas complejas en función del tipo de datos seleccionado.

El tipo de datos de la instrucción se selecciona en la lista desplegable "<???>" del cuadro de la instrucción. Dependiendo del tipo de datos seleccionado se puede combinar la funcionalidad de determinadas instrucciones para ejecutar un cálculo complejo. La expresión que debe calcularse se indica en un cuadro de diálogo que se abre con el símbolo de "calculadora" situado en el margen superior derecho del cuadro de la instrucción. La expresión puede contener los nombres de los parámetros de entrada y la sintaxis de las instrucciones. No está permitido indicar nombres o direcciones de operandos.

El cuadro de la instrucción contiene en estado básico al menos 2 entradas (IN1 e IN2). El número de entradas es ampliable. Las entradas insertadas se numeran en orden ascendente en el cuadro.

Los valores de las entradas se utilizan para ejecutar la expresión indicada. No obstante, no es necesario utilizar en la expresión todas las entradas definidas. El resultado de la instrucción se transfiere a la salida del cuadro OUT.

Nota

Cuando una de las operaciones matemáticas falla en la expresión, no se transfieren resultados a la salida OUT y la salida de habilitación ENO devuelve el estado lógico "1".

Si en la expresión se utilizan entradas que no están disponibles en el cuadro, éstas se insertan automáticamente. Para ello, es indispensable que la numeración de las entradas nuevas que deben definirse en la expresión no contenga huecos. Por ejemplo, es posible no utilizar la entrada IN4 en la expresión si la entrada IN3 no está definida.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado o un resultado intermedio de la instrucción "Calcular" está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.
- Se ha producido un error al ejecutar una de las instrucciones indicadas en la expresión.

La tabla siguiente muestra las instrucciones que pueden combinarse y ejecutarse en la expresión de la instrucción "Calcular" en función del tipo de datos seleccionado:

Tipo de datos	Instrucción	Sintaxis	Ejemplo
Secuencias de bits	AND: Operación lógica Y	AND	IN1 AND IN2 OR IN3
	OR: Operación lógica O	OR	
	XOR: Operación lógica O-exclusiva	XOR	
	INV: Complemento a 1	NOT	
	SWAP: Cambiar disposición [1]	SWAP	
Enteros	ADD: Sumar	+	(IN1 + IN2) * IN3; (ABS(IN2)) * (ABS(IN1))
	SUB: Restar	-	
	MUL: Multiplicar	*	
	DIV: Dividir	/	
	MOD: Obtener resto de división	MOD	
	INV: Complemento a 1	NOT	
	NEG: Generar complemento a dos	-(in1)	
	ABS: Calcular valor absoluto	ABS( )	
Números en coma flotante	ADD: Sumar	+	((SIN(IN2) * SIN(IN2) + (SIN(IN3) * SIN(IN3)) / IN3)); (SQR(SIN(IN2)) + (SQR(COS(IN3)) / IN2))
	SUB: Restar	-	
	MUL: Multiplicar	*	
	DIV: Dividir	/	
	EXPT: Elevar a potencia	**	
	ABS: Calcular valor absoluto	ABS( )	
	SQR: Calcular cuadrado	SQR( )	
	SQRT: Calcular raíz cuadrada	SQRT( )	
	LN: Calcular logaritmo natural	LN( )	
	EXP: Calcular valor exponencial	EXP( )	
	FRAC: Determinar decimales	FRAC( )	
	SIN: Calcular valor de seno	SIN( )	
	COS: Calcular valor de coseno	COS( )	
	TAN: Calcular valor de tangente	TAN( )	
	ASIN: Calcular valor de arcoseno	ASIN( )	
	ACOS: Calcular valor de arcocoseno	ACOS( )	
	ATAN: Calcular valor de arcotangente	ATAN( )	
	NEG: Generar complemento a dos	-(in1)	
	TRUNC: Truncar a entero	TRUNC( )	
	ROUND: Redondear número	ROUND( )	
	CEIL: Redondear un número en coma flotante al siguiente entero superior	CEIL( )	
	FLOOR: Redondear un número en coma flotante al siguiente entero inferior	FLOOR( )	

[1] No es posible para el tipo de datos BYTE.

Parámetros

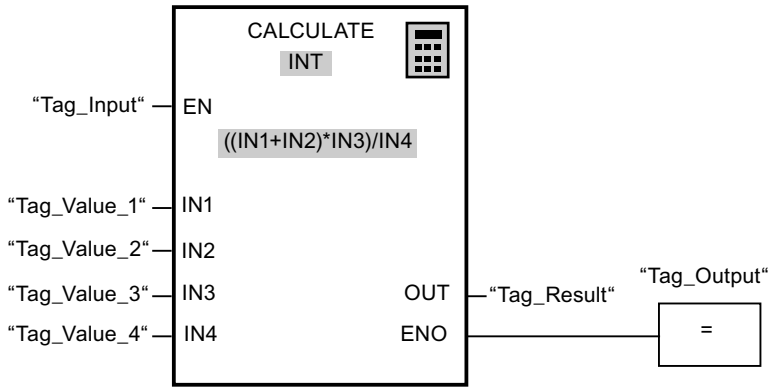
La tabla siguiente muestra los parámetros de la instrucción "Calcular":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primera entrada disponible
IN2	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segunda entrada disponible
INn	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Entradas insertadas adicionalmente
OUT	Output	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Salida a la que se transfiere el resultado final.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value_1	4
IN2	Tag_Value_2	4
IN3	Tag_Value_3	3
IN4	Tag_Value_4	2
OUT	Tag_Result	12

Cuando la entrada "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular". El valor del operando "Tag_Value_1" se suma al valor del operando "Tag_Value_2". La suma se multiplica por el valor del operando "Tag_Value_3". El producto se divide por el valor del operando "Tag_Value_4". El cociente se transfiere como resultado final al operando "Tag_Result" de la salida OUT de la instrucción. Si no ocurren errores al ejecutar las distintas instrucciones, la salida de habilitación ENO y el operando "Tag_Output" se ponen a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Principios básicos del mecanismo EN/ENO (Página 2539)

ADD: Sumar

Descripción

La instrucción "Sumar" permite sumar el valor de la entrada IN1 al valor de la entrada IN2 y consultar la suma en la salida OUT (OUT := IN1+IN2).

El cuadro de la instrucción contiene en estado estacionario mín. 2 entradas (IN1 e IN2). El número de entradas es ampliable. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se suman los valores de todos los parámetros de entrada disponibles. La suma se guarda en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar":

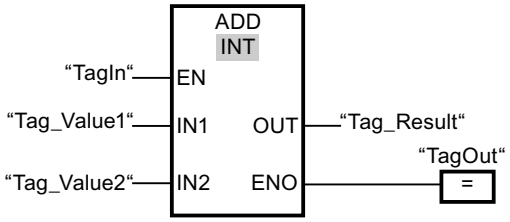
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer sumando
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo sumando
INn	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valores de entrada opcionales que se suman.
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Suma

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Sumar". El valor del operando "Tag_Value1" se suma al valor del operando "Tag_Value2". El resultado de la suma se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2726)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

SUB: Restar

Descripción

La instrucción "Restar" permite restar el valor de la entrada IN2 al valor de la entrada IN1 y consultar el resultado en la salida OUT (OUT := IN1-IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Restar":

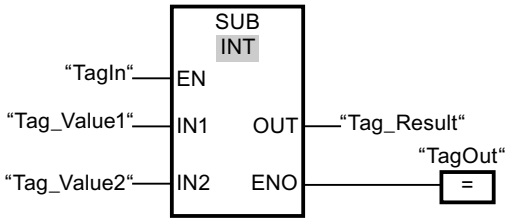
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Sustraendo
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Diferencia

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Restar". El valor del operando "Tag_Value2" se resta del valor del operando "Tag_Value1". El resultado de la resta se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2726)

MUL: Multiplicar

Descripción

La instrucción "Multiplicar" permite multiplicar el valor de la entrada IN1 con el valor de la entrada IN2 y consultar el producto en la salida OUT (OUT := IN1*IN2).

El cuadro de la instrucción contiene en estado estacionario mín. 2 entradas (IN1 e IN2). El número de entradas es ampliable. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se multiplican los valores de todos los parámetros de entrada disponibles. El producto se guarda en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplicar":

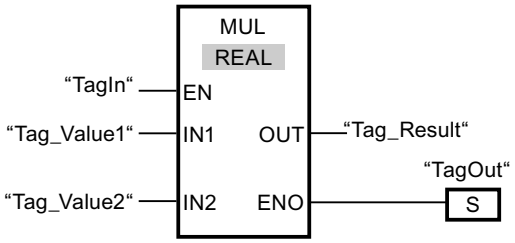
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Multiplicando
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Multiplicador
INn	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valores de entrada opcionales que se multiplican.
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Producto

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar". El valor del operando "Tag_Value1" se multiplica por el valor del operando "Tag_Value2". El resultado de la multiplicación se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2726)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

DIV: Dividir

Descripción

La instrucción "Dividir" permite dividir el valor de la entrada IN1 por el valor de la entrada IN2 y consultar el cociente en la salida OUT (OUT := IN1/IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Dividir":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Dividendo

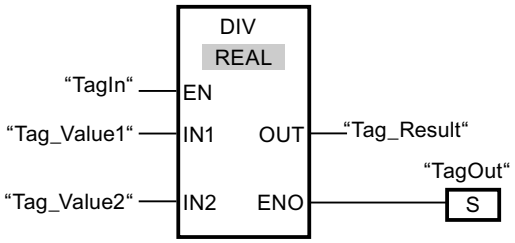
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor del cociente

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Dividir". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resultado de la división se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2726)

MOD: Obtener resto de división

Descripción

La instrucción "Obtener resto de división" permite dividir el valor de la entrada IN1 por el valor de la entrada IN2 y consultar el resto en la salida OUT.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Obtener resto de división":

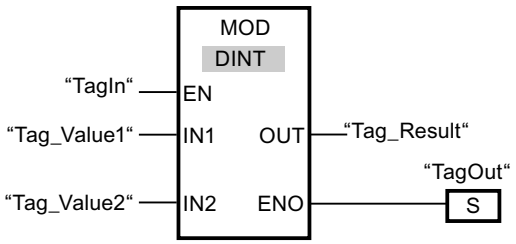
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros	I, Q, M, D, L, P	I, Q, M, D, L, P	Resto de división

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Obtener resto de división". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resto de la división se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar un tipo de datos (Página 2726)

NEG: Generar complemento a dos

Descripción

La instrucción "Generar complemento a dos" permite cambiar el signo del valor de la entrada IN y consultar el resultado en la salida OUT. Por ejemplo, si la entrada IN tiene un valor positivo, se deposita el equivalente negativo de este valor en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a dos":

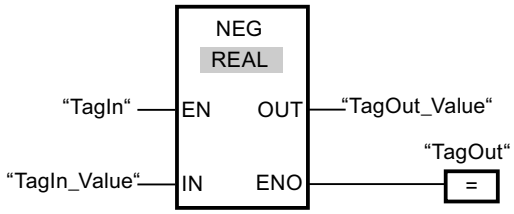
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	SINT, INT, DINT, Números en coma flotante	SINT, INT, DINT, LINT, Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	SINT, INT, DINT, Números en coma flotante	SINT, INT, DINT, LINT, Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Complemento a dos del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Generar complemento a dos". El signo del valor de la entrada "TagIn_Value" cambia y el resultado se almacena en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

INC: Incrementar

Descripción

La instrucción "Incrementar" permite cambiar el valor del operando del parámetro IN/OUT al siguiente valor superior y consultar el resultado.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Incrementar":

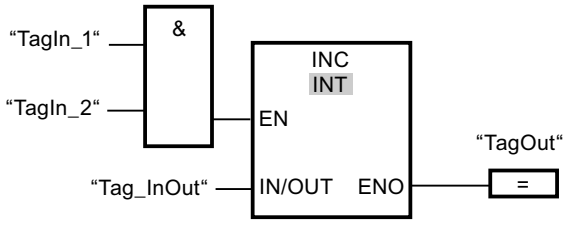
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN/OUT	InOut	Enteros	I, Q, M, D, L	I, Q, M, D, L	Valor que se incrementa.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", el valor del operando "Tag_InOut" se incrementa en uno y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

DEC: Decrementar

Descripción

La instrucción "Decrementar" permite cambiar el valor del operando del parámetro IN/OUT al siguiente valor inferior y consultar el resultado.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Decrementar":

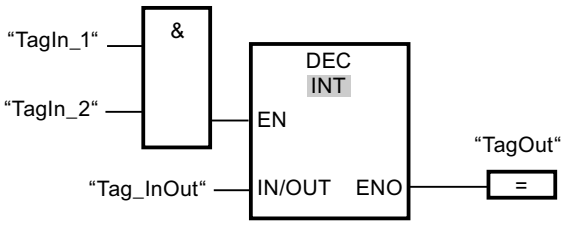
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN/OUT	InOut	Enteros	I, Q, M, D, L	I, Q, M, D, L	Valor que se decrementa.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", el valor del operando "Tag_InOut" se decrementa en uno y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" permite calcular el valor absoluto del valor indicado en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor absoluto":

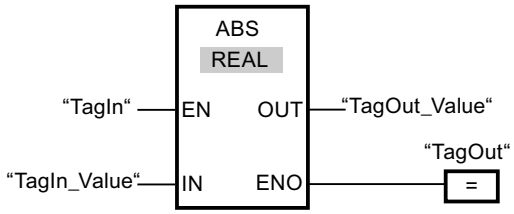
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor absoluto del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	-6.234
OUT	TagOut_Value	6.234

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor absoluto". La instrucción calcula el valor absoluto del valor de la entrada "TagIn_Value" y deposita el resultado en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas disponibles y escribe el valor menor en la salida OUT. El número de entradas se puede ampliar en el cuadro de la instrucción mediante entradas adicionales. Las entradas se numeran de forma ascendente en el cuadro.

En estado básico, la instrucción contiene mín. dos entradas (IN1 e IN2) y hasta un máx. de 100 entradas.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

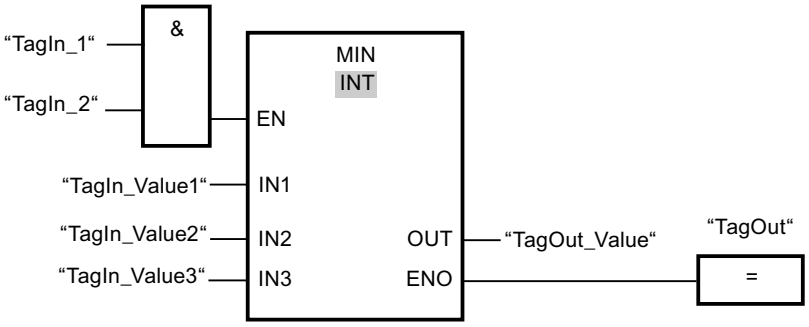
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de entrada
INn	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Entradas insertadas adicionalmente cuyos valores se comparan
OUT	Output	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.						

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar mínimo". La instrucción compara los valores de los operandos indicados y copia el valor menor ("TagIn_Value1") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas disponibles y escribe el valor mayor en la salida OUT. El número de entradas se puede ampliar en el cuadro de la instrucción mediante entradas adicionales. Las entradas se numeran de forma ascendente en el cuadro.

En estado básico, la instrucción contiene mín. dos entradas (IN1 e IN2) y hasta un máx. de 100 entradas.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de entrada

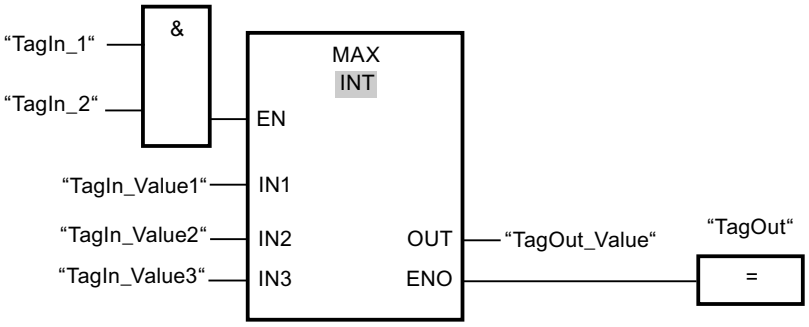
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
INn	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Entradas insertadas adicionalmente cuyos valores se comparan.
OUT	Output	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.						

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar máximo". La instrucción compara los valores de los operandos indicados y copia

el valor mayor ("TagIn_Value2") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor de la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no se define y la salida de habilitación ENO es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor de la entrada MN es mayor que el valor de la entrada MX.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación

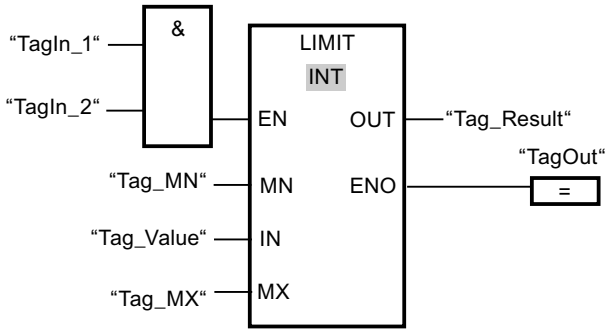
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
MN	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Límite inferior
IN	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
MX	Input	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Límite superior
OUT	Output	Enteros, números en coma flotante, TIME, TOD, DATE	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.						

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se ejecuta la instrucción "Ajustar valor límite". El valor del operando "Tag_Value" se compara con los valores de los operandos "Tag_MN" y "Tag_MX". Puesto que el valor del operando "Tag_Value" es menor que el valor límite inferior, el valor del operando "Tag_MN" se copia en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" permite elevar al cuadrado el valor de la entrada IN y consultar el resultado en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular cuadrado":

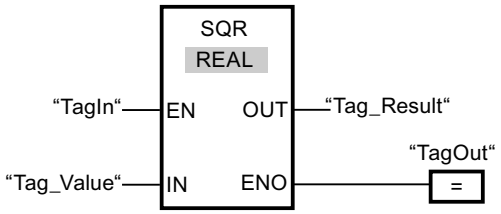
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Cuadrado del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular cuadrado". La instrucción eleva al cuadrado el valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" permite extraer la raíz cuadrada al valor de la entrada IN y consultar el resultado en la salida OUT. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido. Si el valor de la entrada IN es "0", el resultado también es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular raíz cuadrada":

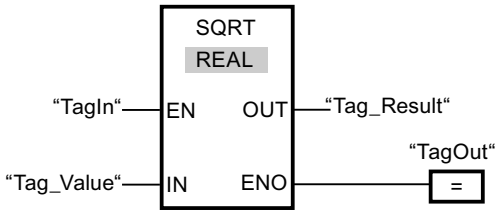
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L	I, Q, M, D, L	Raíz cuadrada del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular raíz cuadrada". La instrucción extrae la raíz cuadrada del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2,718282) del valor de la entrada IN. El resultado se deposita en la salida OUT y se puede consultar allí. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular logaritmo natural":

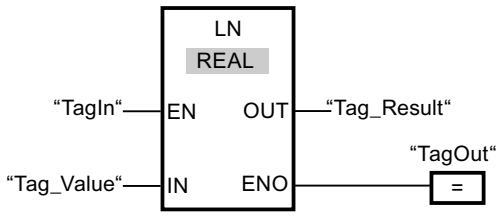
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Logaritmo natural del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular logaritmo natural". La instrucción calcula el logaritmo natural del valor de la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2,718282) del valor indicado en la entrada IN. El resultado se emite en la salida OUT y se puede consultar allí (OUT = e^{IN}).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor exponencial":

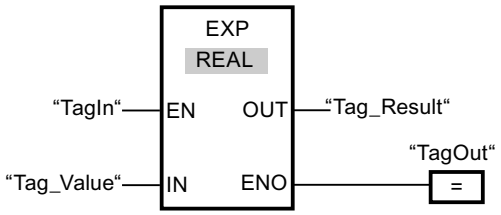
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor exponencial del valor de entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor exponencial". La instrucción calcula la potencia en base e del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno de un ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de seno":

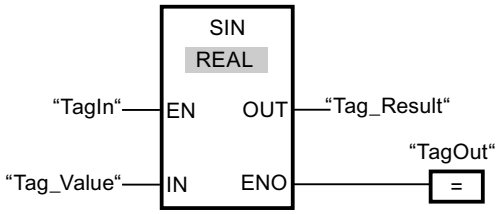
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Seno del ángulo indicado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	1.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de seno". La instrucción calcula el seno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno de un ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de coseno":

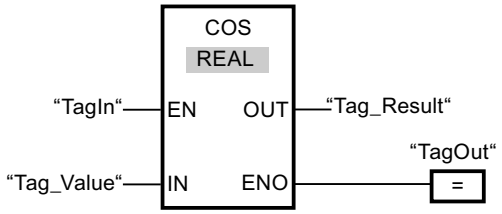
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Coseno del ángulo indicado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de coseno". La instrucción calcula el coseno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente de un ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de tangente":

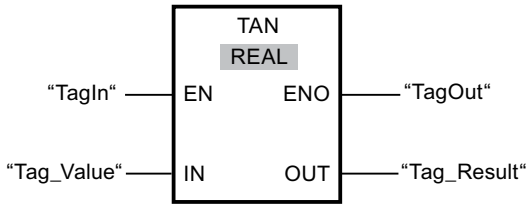
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Tangente del ángulo indicado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+3.141593 (π)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de tangente". La instrucción calcula la tangente del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno indicado en la entrada IN el tamaño del ángulo que corresponde a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcoseno":

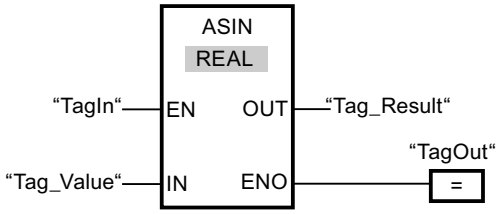
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de seno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Tamaño del ángulo en radianes

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+1.570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcoseno". La instrucción calcula el tamaño del ángulo que corresponde al valor de seno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno indicado en la entrada IN el tamaño del ángulo que corresponde a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre 0 y +π.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcocoseno":

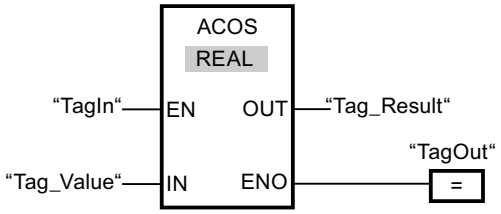
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de coseno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Tamaño del ángulo en radianes

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0
OUT	Tag_Result	+1.570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcocoseno". La instrucción calcula el tamaño del ángulo que corresponde al valor de coseno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente indicado en la entrada IN el tamaño del ángulo que corresponde a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcotangente":

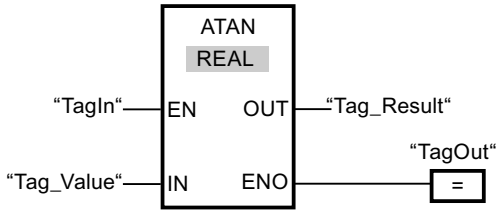
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de tangente
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Tamaño del ángulo en radianes

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+0,785398 (π/4)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcotangente". La instrucción calcula el tamaño del ángulo que corresponde al valor de tangente de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

FRAC: Determinar decimales

Descripción

La instrucción "Determinar decimales" permite determinar los decimales del valor de la entrada IN. El resultado de la consulta se deposita en la salida OUT y se puede consultar allí. Por ejemplo, si el valor de la entrada IN es 123,4567, la salida OUT devuelve el valor 0,4567.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Al ejecutar la instrucción se producen errores, p. ej. la entrada no tiene un número en coma flotante válido).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar decimales":

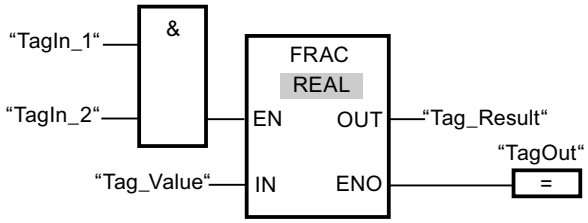
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada cuyos decimales se determinan.
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Decimales del valor de la entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	2.555
OUT	Tag_Result	0.555

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se inicia la instrucción "Determinar decimales". Los decimales del valor del operando "Tag_Value" se copian en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

EXPT: Elevar a potencia

Descripción

La instrucción "Elevar a potencia" permite elevar el valor de la entrada IN1 a la potencia del valor de la entrada IN2. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí (OUT = $IN1^{IN2}$).

A la entrada IN1 sólo se pueden asignar valores en coma flotante válidos. A la entrada IN2 también se pueden asignar números enteros.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Al ejecutar la instrucción ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Elevar a potencia":

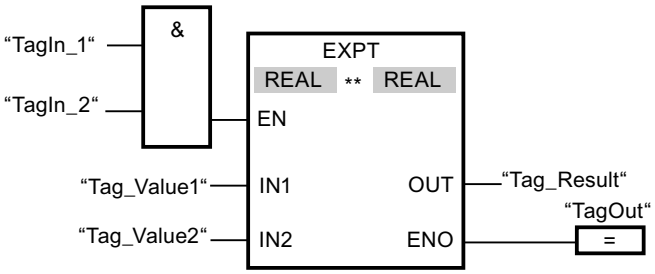
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor base
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor del exponente
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se inicia la instrucción "Elevar a potencia". El valor del operando "Tag_Value1" se eleva a la potencia del valor del operando "Tag_Value2". El resultado se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Transferencia

MOVE: Copiar valor

Descripción

La instrucción "Copiar valor" transfiere el contenido del operando de la entrada IN al operando de la salida OUT1. La transferencia se efectúa siempre por orden ascendente de direcciones.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El tipo de datos del parámetro IN no puede convertirse al tipo de datos indicado en el parámetro OUT1.

La tabla siguiente muestra las posibles transferencias para la familia de CPU S7-1200:

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE ,TOD, CHAR
WORD	WORD, DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, REAL, TIME, DATE, TOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
USINT	USINT, UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
INT	INT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UINT	UINT, UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
UDINT	UDINT	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LREAL

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
TIME	TIME	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TIME
DATE	DATE	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, DATE
TOD	TOD	BYTE, WORD, DWORD, SINT, USINT, INT, UINT, DINT, UDINT, TOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, CHAR, carácter de una cadena[1)]
Carácter de una cadena[1)]	Carácter de una cadena	CHAR, carácter de una cadena
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_SCOUNTER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUNTER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER	IEC_UDCOUNTER

La tabla siguiente muestra las posibles transferencias para la familia de CPU S7-1500:

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
BYTE	BYTE, WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE ,TOD, LTOD, CHAR
WORD	WORD, DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, S5TIME, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
DWORD	DWORD, LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, REAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
LWORD	LWORD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LREAL, TIME, LTIME, LDT, DATE, TOD, LTOD, CHAR
SINT	SINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
USINT	USINT, UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
INT	INT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UINT	UINT, UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD. LTOD
DINT	DINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
UDINT	UDINT, ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
LINT	LINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
ULINT	ULINT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME, LTIME, LDT, DATE, TOD, LTOD
REAL	REAL	DWORD, REAL
LREAL	LREAL	LWORD, LREAL
S5TIME	S5TIME	WORD, S5TIME
TIME	TIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TIME
LTIME	LTIME	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTIME
DATE	DATE	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, DATE
DT	DT	DT
LDT	LDT	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LDT
TOD	TOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, TOD
LTOD	LTOD	BYTE, WORD, DWORD, LWORD, SINT, USINT, INT, UINT, DINT, UDINT, LINT, ULINT, LTOD
DTL	DTL	DTL
CHAR	CHAR	BYTE, WORD, DWORD, LWORD, CHAR, carácter de una cadena[1)]
WCHAR	WCHAR	BYTE, WORD, DWORD, LWORD, CHAR, WCHAR, carácter de una cadena[1)]
Carácter de una cadena[1)]	Carácter de una cadena	CHAR, WCHAR, carácter de una cadena
ARRAY[2)]	ARRAY	ARRAY
STRUCT	STRUCT	STRUCT
COUNTER	COUNTER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
TIMER	TIMER, WORD, INT	WORD, DWORD, INT, UINT, DINT, UDINT
Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)
IEC_TIMER	IEC_TIMER	IEC_TIMER
IEC_LTIMER	IEC_LTIMER	IEC_LTIMER
IEC_SCOUNTER	IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUNTER	IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER	IEC_UDCOUNTER
IEC_LCOUNTER	IEC_LCOUNTER	IEC_LCOUNTER
IEC_ULCOUNTER	IEC_ULCOUNTER	IEC_ULCOUNTER

[1)] La instrucción "Copiar valor" permite transferir también caracteres individuales de una cadena de caracteres a operandos del tipo de datos CHAR o WCHAR. El número del carácter que se va a transferir se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se transfiere p. ej. el segundo carácter de la cadena de caracteres "MyString". Asimismo, es posible transferir operandos del tipo de datos CHAR a caracteres individuales de una cadena de caracteres. También es posible reemplazar un carácter específico de una cadena de caracteres por el carácter de otra cadena de caracteres.

[2)] Sólo es posible transferir matrices enteras (ARRAY) si los elementos de matriz de los operandos de la entrada IN y de la salida OUT1 son del mismo tipo de datos.

Si la longitud de bits del tipo de datos de la entrada IN supera la longitud de bits del tipo de datos de la salida OUT1, se pierden los bits más significativos del valor de origen. Si la longitud de bit del tipo de datos en la entrada IN es inferior a la longitud de bits del tipo de datos en la salida OUT1, se sobrescriben con ceros los bits más significativos del valor de destino.

El cuadro de la instrucción contiene en estado inicial 1 salida (OUT1). El número de salidas es ampliable. Las salidas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se transfiere el contenido del operando de la entrada IN a todas las salidas disponibles. El cuadro de la instrucción no se puede ampliar si se transfieren tipos de datos estructurados (DTL, STRUCT, ARRAY) o caracteres de una cadena de caracteres.

Para copiar operandos del tipo de datos ARRAY pueden utilizarse también las instrucciones "Copiar área" (MOVE_BLK) y "Copiar área sin interrupciones" (UMOVE_BLK). Los operandos del tipo de datos STRING se pueden copiar con la instrucción "Desplazar cadena de caracteres" (S_MOVE).

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Copiar valor":

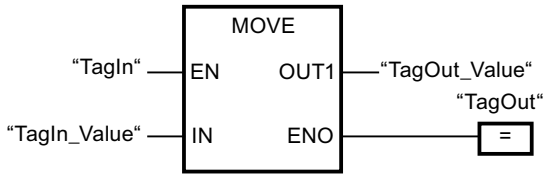
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN	Input	Secuencias de bits, enteros, números en coma flotante, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, tipos de datos IEC, tipos de datos PLC (UDT)	Secuencias de bits, enteros, números en coma flotante, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, tipos de datos IEC, tipos de datos PLC (UDT)	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Elemento con el que se sobrescribe la dirección de destino.
OUT1	Output	Secuencias de bits, enteros, números en coma flotante, DATE, TIME, TOD, DTL, CHAR, STRUCT, ARRAY, tipos de datos IEC, tipos de datos PLC (UDT)	Secuencias de bits, enteros, números en coma flotante, DATE, DT, LDT, S5TIME, TIME, LTIME, TOD, LTOD, DTL, CHAR, STRUCT, ARRAY, TIMER, COUNTER, tipos de datos IEC, tipos de datos PLC (UDT)	I, Q, M, D, L	I, Q, M, D, L	Dirección de destino

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar valor". La instrucción copia los contenidos del operando "TagIn_Value" al operando "TagOut_Value". Si no se producen errores al ejecutar la instrucción, las salidas de habilitación ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Quitar entradas y salidas de una instrucción (Página 2742)

MOVE_BLK: Copiar área (Página 3766)

UMOVE_BLK: Copiar área sin interrupciones (Página 3773)

S_MOVE: Desplazar cadena de caracteres (Página 4706)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Deserialize: Deserializar

Descripción

Con la instrucción "Deserializar", la forma de representación secuencial de un tipo de datos PLC (UDT) se vuelve a convertir en un tipo de datos PLC y se rellena todo el contenido.

El área de memoria en la que se encuentra la forma de representación secuencial de un tipo de datos PLC debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

Esta instrucción permite reconvertir paso a paso a su estado original varias formas de representación secuenciales de tipos de datos PLC convertidos.

Nota

Para reconvertir una sola forma de representación secuencial de un tipo de datos PLC (UDT) también se puede utilizar directamente la instrucción "TRCV: Recibir datos a través de la conexión".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Deserializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC_ARRAY	Input	VARIANT	I, Q, D, L	Bloque de datos global en el que se guarda el flujo de datos creado
DEST_VARIABLE	InOut	VARIANT	I, Q, D, L	Variable en la que se guarda el tipo de datos PLC (UDT) reconvertido
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

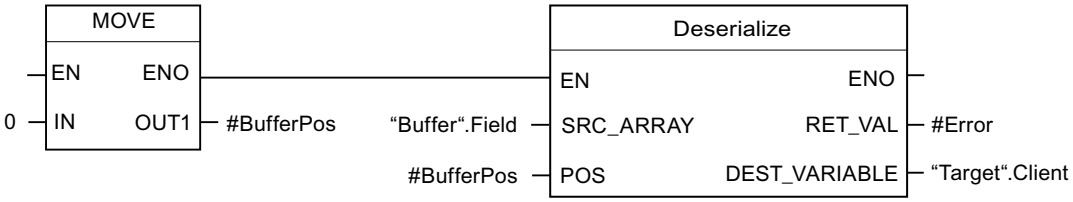
Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_ARRAY y DEST_VARIABLE se solapan.
8136	El bloque de datos del parámetro DEST_VARIABLE no es un bloque con acceso estándar.
8150	El tipo de datos VARIANT del parámetro SRC_ARRAY no contiene ningún valor.
8151	Error de creación de código en el parámetro SRC_ARRAY
8153	En el parámetro SRC_ARRAY no hay suficiente memoria disponible.
8250	El tipo de datos VARIANT del parámetro DEST_VARIABLE no contiene ningún valor.
8251	Error de creación de código en el parámetro DEST_VARIABLE

Código de error* (W#16#...)	Explicación
8254	Tipo de datos no válido en el parámetro DEST_VARIABLE
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

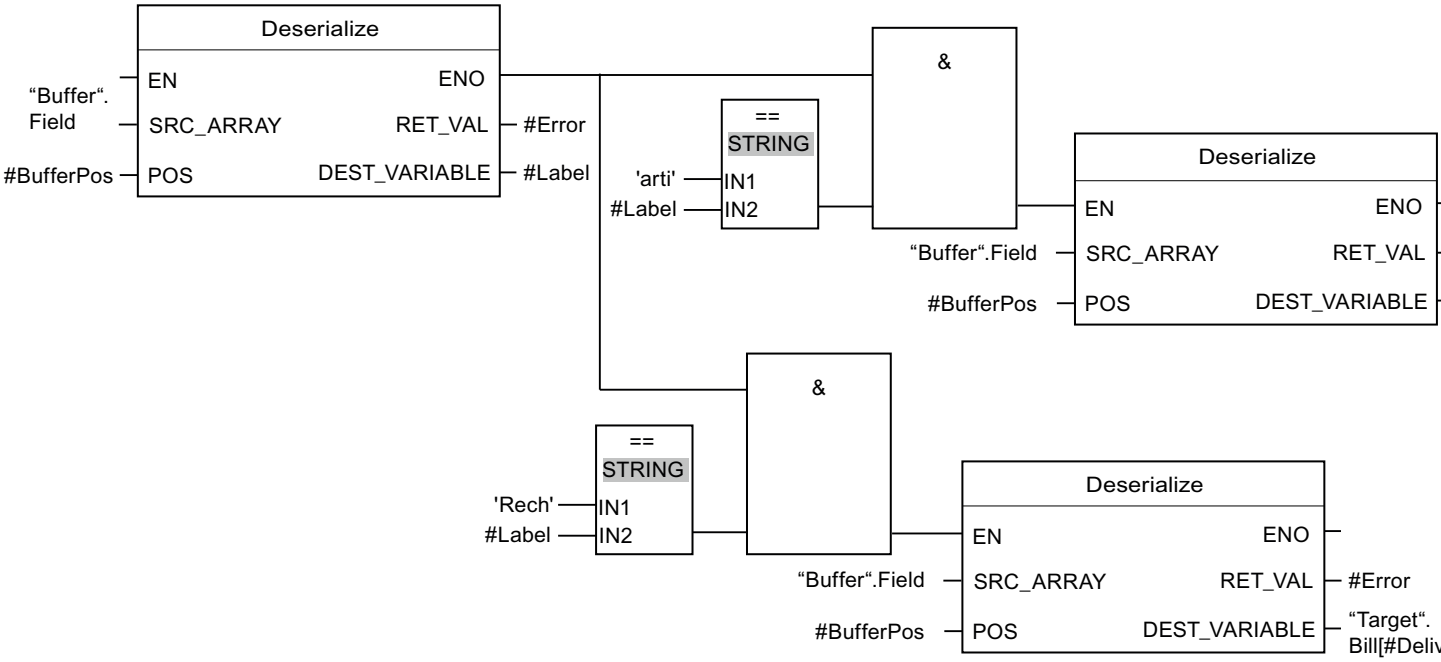
El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1:



La instrucción "Copiar valor" copia el valor "0" en el operando "#BufferPos". La instrucción "Deserializar" deserializa la forma de representación secuencial de los datos de cliente del bloque de datos "Buffer" y los escribe en el bloque de datos "Target". El número de bytes que ocupan los datos de cliente convertidos se guarda en el operando "#BufferPos".

Segmento 2:



La instrucción "Deserializar" deserializa la forma de representación secuencial del separador (depositado después de los datos de cliente en la forma de representación secuencial) del bloque de datos "Buffer" y escribe los caracteres en el operando "#Label". Los caracteres se

someten a una comparación con "arti" y "Bill" mediante instrucciones de comparación. Si la comparación en relación con "arti" = TRUE, se trata de datos de artículo que se deserializan y se escriben en el bloque de datos "Target". Si la comparación en relación con "Bill" = TRUE, se trata de datos de facturas que se deserializan y se escriben en el bloque de datos "Target".

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Target	Client	"Client" (tipo de datos PLC)
	Article	Array[0..] of "Article" (tipo de datos PLC)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Serialize: Serializar

Descripción

Mediante la instrucción "Serializar" se convierten varios tipos de datos PLC (UDT) a una forma de representación secuencial sin que se pierdan partes de su estructura.

La instrucción permite guardar temporalmente varios datos estructurados del programa en un búfer, ubicado preferentemente en un bloque de datos global, y enviarlos a otra CPU. El área de memoria en la que se depositen los tipos de datos PLC convertidos debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

El operando del parámetro POS contiene la información del número de bytes que ocupan los tipos de datos PLC convertidos.

Nota

Para enviar un solo tipo de datos PLC (UDT), se puede llamar directamente la instrucción "TSEND: Enviar datos a través de la conexión".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Serializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC_VARIABLE	Input	VARIANT	I, Q, D, L	Tipo de datos PLC (UDT), que se convierte a una forma de representación secuencial.
DEST_ARRAY	InOut	VARIANT	I, Q, D, L	Bloque de datos en el que se guarda el flujo de datos generado.
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

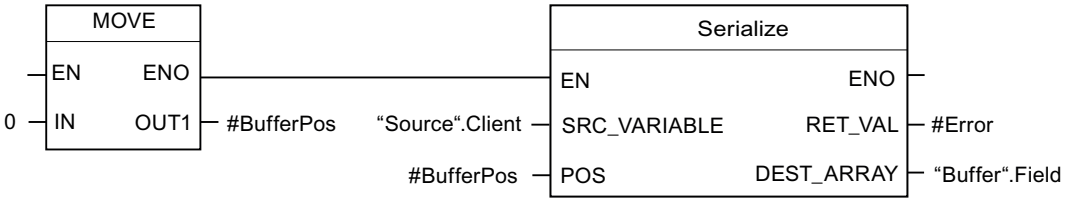
Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_VARIABLE y DEST_ARRAY se solapan.

Código de error* (W#16#...)	Explicación
8150	El tipo de datos VARIANT del parámetro SRC_VARIABLE no contiene ningún valor.
8152	Error de creación de código en el parámetro SRC_VARIABLE
8236	El bloque de datos del parámetro DEST_ARRAY no es un bloque con acceso estándar.
8250	El tipo de datos VARIANT del parámetro DEST_ARRAY no contiene ningún valor.
8252	Error de creación de código en el parámetro DEST_ARRAY
8253	En el parámetro DEST_ARRAY no hay suficiente memoria disponible.
8254	Tipo de datos no válido en el parámetro DEST_ARRAY
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

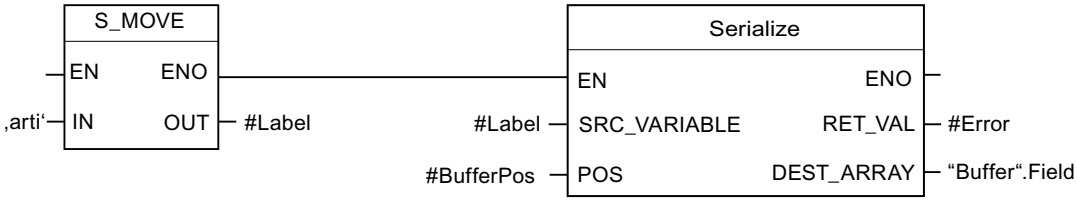
El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1:



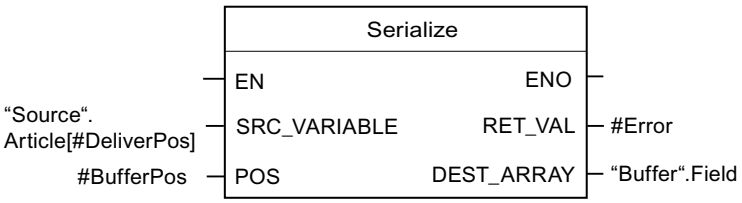
La instrucción "Copiar valor" copia el valor "0" en el operando "#BufferPos". La instrucción "Serializar" serializa los datos de cliente del bloque de datos "Source" y los escribe como representación secuencial en el bloque de datos "Buffer". El número de bytes que ocupa la representación secuencial se guarda en el operando "#BufferPos".

Segmento 2:



Para facilitar la posterior deserialización de la forma de representación secuencial, ahora se inserta una especie de separador. La instrucción "Desplazar cadena de caracteres" copia los caracteres "arti" en el operando "#Label". La instrucción "Serializar" escribe estos caracteres después de los datos de cliente en el bloque de datos "Buffer". El número de bytes que requieren los caracteres se suma en el operando "#BufferPos" al número ya guardado.

Segmento 3:



La instrucción "Serializar" serializa los datos de un determinado artículo, que se calcula en tiempo de ejecución, del bloque de datos "Source", y los escribe en forma de representación secuencial después de los caracteres "arti" en el bloque de datos "Buffer".

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Source	Client	"Client" (tipo de datos PLC)
	Article	Array[0..10] of "Article" (tipo de datos PLC)
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

FieldRead: Leer campo

Descripción

La instrucción "Leer campo" permite leer un componente determinado del campo indicado en el parámetro MEMBER y transferir su contenido a la variable del parámetro VALUE. El índice del elemento que se lee se define en el parámetro INDEX. En el parámetro MEMBER se indica el primer elemento del campo que se lee.

Los tipos de datos del componente de campo indicado en el parámetro MEMBER y de la variable del parámetro VALUE deben coincidir con el tipo de datos de la instrucción "Leer campo".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El elemento indicado en el parámetro INDEX no está definido en el campo especificado en el parámetro MEMBER.
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer campo":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
INDEX	Input	DINT	DINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Índice del elemento cuyo contenido se lee

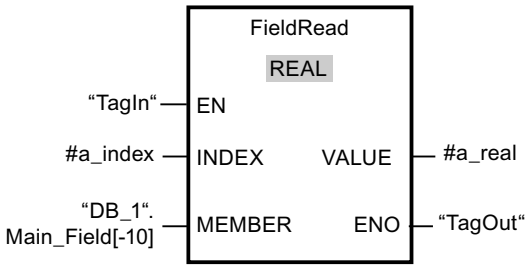
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
MEMBER	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD y CHAR como elementos de una variable ARRAY	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR como elementos de una variable ARRAY	D, L	D, L	Primer elemento del campo que se lee
VALUE	Output	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD y CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR	I, Q, M, D, L, P	I, Q, M, D, L, P	Operando al que se transfiere el contenido del elemento

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Variable	Valor
INDEX	a_index	4
MEMBER	"DB_1".Main_Field[-10]	Primer elemento del campo "Main_Field[-10..10] of REAL" en el bloque de datos "DB_1"
VALUE	a_real	Elemento con índice 4 del campo "Main_Field[-10..10] of REAL"

El elemento con índice 4 se lee del campo "Main_Field[-10...10] of REAL" y se escribe en la variable "a_real". El elemento que se lee está definido por el valor del parámetro INDEX.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

FieldWrite: Escribir campo

Descripción

La instrucción "Escribir campo" transfiere el contenido de la variable de la entrada VALUE a un elemento determinado del campo especificado en la salida MEMBER. El índice del elemento que se escribe se define con el valor de la entrada INDEX. En la salida MEMBER se indica el primer componente del campo en el que se escribe.

Los tipos de datos del componte de campo especificado en la salida MEMBER y de la variable de la entrada VALUE deben coincidir con el tipo de datos de la instrucción "Escribir campo".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El elemento indicado en la entrada INDEX no está definido en el campo especificado en la salida MEMBER.
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir campo":

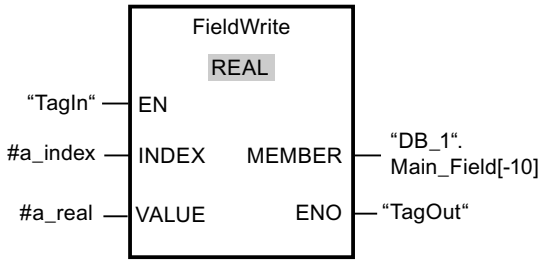
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
INDEX	Input	DINT	DINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Índice del elemento en el que se escribe el contenido de VALUE.
VALUE	Input	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD y CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Operando cuyo contenido se copia
MEMBER	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD y CHAR como elementos de una variable ARRAY	Números binarios, enteros, números en coma flotante, temporizadores, DATE, TOD, LTOD y CHAR como elementos de una variable ARRAY	D, L	D, L	Primer elemento del campo en el que se escribe el contenido de VALUE.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
INDEX	a_index	4
VALUE	a_real	10.54
MEMBER	"DB_1".Main_Field[-10]	Primer elemento del campo "Main_Field[-10..10] of REAL" en el bloque de datos "DB_1"

El valor "10.54" de la variable "a_real" se escribe en el componente de campo con el índice 4 del campo "Main_Field[-10 ... 10] of REAL". El índice del elemento de campo al que se transfiere el contenido de la variable "a_real" está definido por el valor de la entrada INDEX.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

MOVE_BLK: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento de la entrada IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura Array. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura Array, la salida de habilitación ENO se pone a "0".

Parámetros

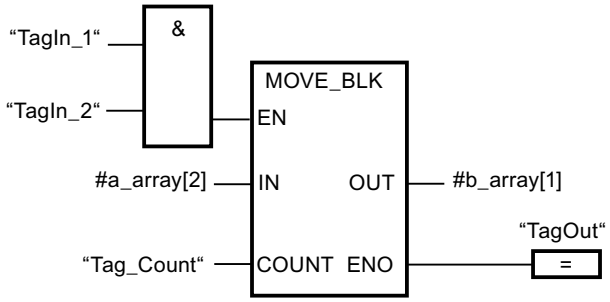
La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino
OUT [1]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.						

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Copiar área". La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

MOVE_BLK_VARIANT: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). Se pueden copiar elementos de

un array en otro array del mismo tipo de datos. Los tamaños (número de elementos) de los arrays de origen y de destino pueden ser diferentes. Se pueden copiar varios elementos dentro de un array o bien copiar los elementos uno a uno.

En el momento de utilizar la instrucción para crear un bloque, aún no tiene que conocerse el array, puesto que el origen y el destino se transfieren por VARIANT.

El contaje de los parámetros SRC_INDEX y DEST_INDEX siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el puntero VARIANT (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC	Input	VARIANT (array o elemento individual)	L	El área de origen de la que se copia
COUNT	Input	UDINT	I, Q, M, D, L o constante	Número de elementos que se copian Asigne al parámetro COUNT el valor "1" si en el parámetro SRC o en el parámetro DEST no hay indicado ningún Array.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro SRC_INDEX se calcula basado en cero. Si en el parámetro SRC hay indicado un Array, el entero del parámetro SRC_INDEX indica el primer elemento del área de origen de la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro SRC no hay indicado ningún Array o solo hay indicado un elemento individual de un Array, asigne al parámetro SRC_INDEX el valor "0".
DEST_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro DEST_INDEX se calcula basado en cero. Si en el parámetro DEST hay indicado un Array, el entero del parámetro DEST_INDEX indica el primer elemento del área de destino en la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro DEST no hay indicado ningún Array, asigne al parámetro DEST_INDEX el valor "0".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DEST	Output	VARIANT	L	Área de destino a la que se copia el contenido del área de origen.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

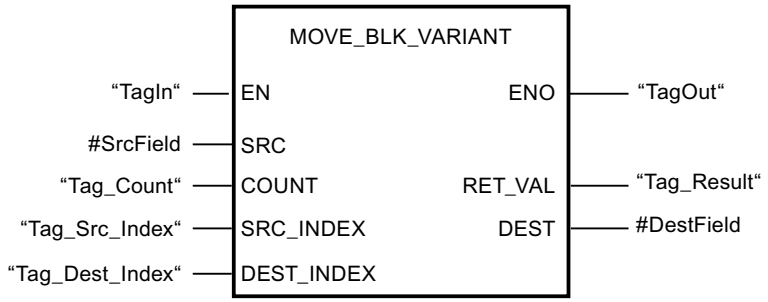
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
8151	No es posible acceder al parámetro SRC.
8152	El operando del parámetro SRC no está tipificado.
8153	Error de creación de código en el parámetro SRC
8154	El operando del parámetro SRC tiene el tipo de datos BOOL.
8281	Valor no válido en el parámetro COUNT
8382	El valor del parámetro SRC_INDEX se encuentra fuera de los valores límite de VARIANT.
8383	El valor del parámetro SRC_INDEX se encuentra fuera del límite superior del array.
8482	El valor del parámetro DEST_INDEX se encuentra fuera de los valores límite de VARIANT.
8483	El valor del parámetro DEST_INDEX se encuentra fuera del límite superior del array.
8534	El parámetro DEST está protegido contra escritura
8551	No es posible acceder al parámetro DEST.
8552	El operando del parámetro DEST no está tipificado.
8553	Error de creación de código en el parámetro DEST
8554	El operando del parámetro DEST tiene el tipo de datos BOOL.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC	#SrcField	El operando local #SrcField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [0..10] of "MOVE_UDT"
COUNT	Tag_Count	2
SRC_INDEX	Tag_Src_Index	3
DEST_INDEX	Tag_Dest_Index	3
DEST	#DestField	El operando local #DestField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [10..20] of "MOVE_UDT"

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar área". En el área de destino se copian 2 elementos del área de origen, empezando por el cuarto elemento del array de UDT. Las copias se insertan en el Array of UDT a partir del cuarto elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VariantGet: Leer valor de una variable VARIANT (Página 3793)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

UMOVE_BLK: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia sin interrupciones los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento de la entrada IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Copiar área sin interrupciones".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura Array. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura Array, la salida de habilitación ENO se pone a "0".

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

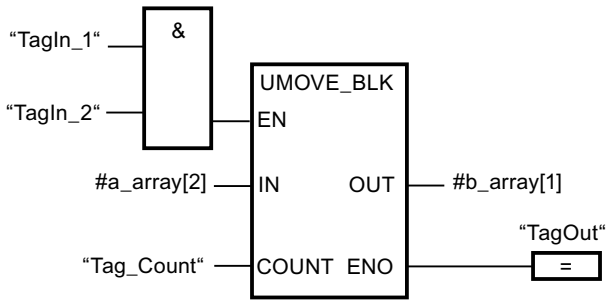
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino
OUT [1]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.						

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Copiar área sin interrupciones". La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

FILL_BLK: Rellenar área

Descripción

La instrucción "Rellenar área" permite rellenar un área de memoria (área de destino) con el valor de la entrada IN. El área de destino se rellena a partir de la dirección indicada en la salida OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura Array. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura Array, la salida de habilitación ENO se pone a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

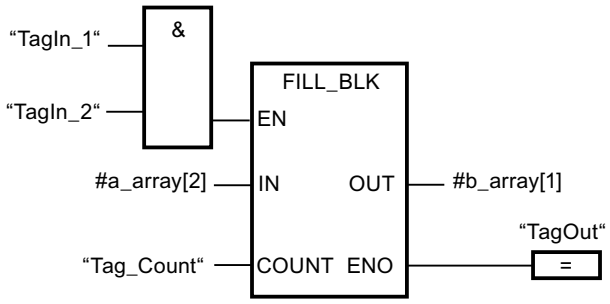
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, TOD, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	D, L	Dirección del área de destino a partir de la cual se rellena

[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura Array.
[2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Rellenar área". La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. Si no se producen errores al ejecutar la instrucción, las salidas ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

UFILL_BLK: Rellenar área sin interrupciones

Descripción

La instrucción "Rellenar área sin interrupciones" rellena sin interrupciones un área de memoria (área de destino) con el valor de la entrada IN. El área de destino se rellena a partir de la dirección indicada en la salida OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Rellenar área sin interrupciones".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Si se copia un Array of BOOL, en caso de desbordamiento la salida de habilitación ENO permanece a "1" hasta que se rebasa el límite de byte de la estructura Array. Si el valor de la entrada COUNT rebasa el límite del byte de la estructura Array, la salida de habilitación ENO se pone a "0".

La instrucción "Rellenar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área sin interrupciones":

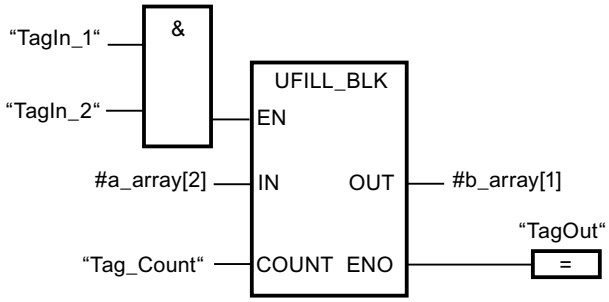
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	D, L	D, L	Dirección del área de destino a partir de la cual se rellena

[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura Array.
[2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Rellenar área sin interrupciones". La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Si no se producen errores al ejecutar la instrucción, las salidas ENO y "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

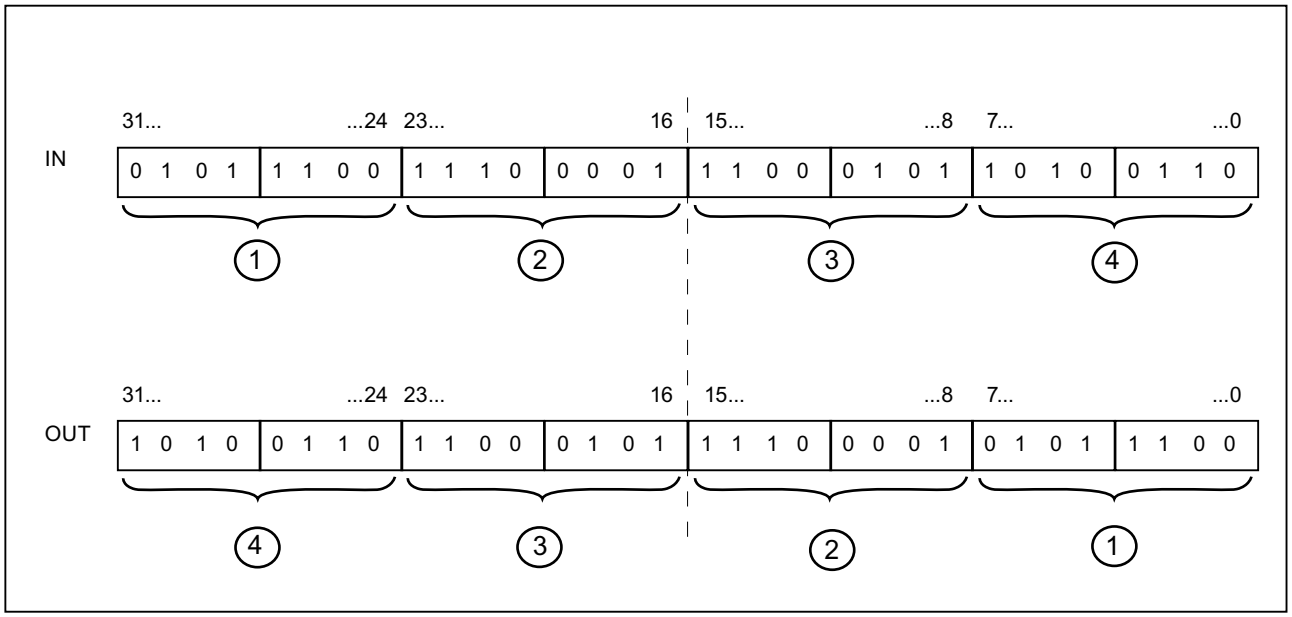
Cablear operandos en instrucciones FUP (Página 2745)

SWAP: Cambiar disposición

Descripción

La instrucción "Cambiar disposición" permite cambiar la disposición de los bytes de la entrada IN y consultar el resultado en la salida OUT.

La figura siguiente muestra cómo se cambian los bytes de un operando del tipo de datos DWORD utilizando la instrucción "Cambiar disposición":



Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Cambiar disposición":

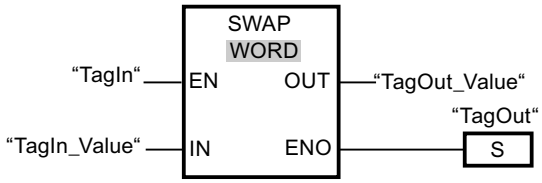
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD, DWORD	WORD, DWORD, LWORD	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Operando cuyos bytes se intercambian.
OUT	Output	WORD, DWORD	WORD, DWORD, LWORD	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0000 1111 0101 0101
OUT	TagOut_Value	0101 0101 0000 1111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Cambiar disposición". La disposición de los bytes se cambia y se deposita en el operando "TagOut_Value".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

Instrucciones DB de ARRAY

ReadFromArrayDB: Leer en un bloque de datos ARRAY

Descripción

La instrucción "Leer en un bloque de datos ARRAY" permite leer datos en un DB de array y escribirlos en un área de destino.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer en un bloque de datos ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos desde el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
VALUE	Output	VARIANT	L	Valor que se lee y se devuelve
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

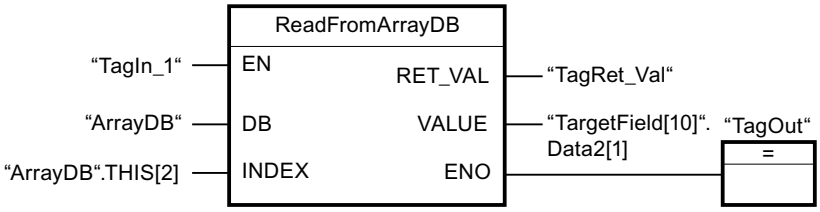
La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El DB de array contiene valores no válidos
8154	El tipo de datos no está declarado correctamente

Código de error* (W#16#...)	Explicación
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	TargetField[10].Data2[1]	El operando "TargetField" es del tipo de datos Array [10..20] of "MOVE_UDT".

Si el operando "TagIn1" devuelve el estado lógico "1", se ejecuta la instrucción "Leer en un bloque de datos ARRAY". El tercer elemento se lee en el "ArrayDB" y se escribe en el operando "TargetField[10].Data2[1]". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

WriteToArrayDB: Escribir en un bloque de datos ARRAY

Descripción

La instrucción "Escribir en un bloque de datos ARRAY" permite escribir datos en un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en bloque de datos de array":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	L	Valor que se escribe
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

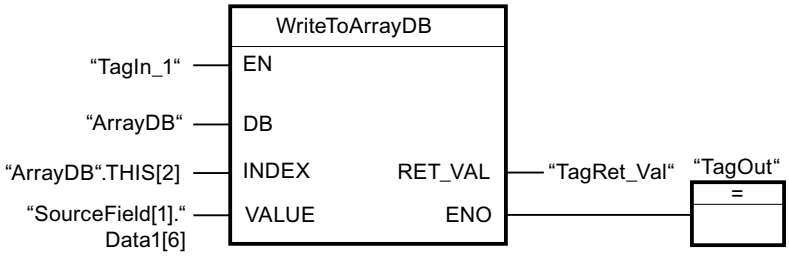
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceField[1].Data1[6]	El operando "SourceField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Si el operando "TagIn1" devuelve el estado lógico "1", se ejecuta la instrucción "Escribir en un bloque de datos ARRAY". Del operando "SourceField" se escribe el elemento "Data1[6]"

del segundo elemento en el "ArrayDB". En el "ArrayDB" se escribe el tercer elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

ReadFromArrayDBL: Leer en un bloque de datos ARRAY de la memoria de carga

Descripción

La instrucción "Leer en un bloque de datos ARRAY de la memoria de carga" permite leer datos desde un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1" y el valor leído en el parámetro VALUE se emite dentro de dicho ciclo. En los demás ciclos del programa, el valor del parámetro VALUE no se modifica.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la lectura del DB de array
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos desde el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La lectura del DB de array todavía no ha finalizado
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
VALUE	InOut	VARIANT	D, L	Valor que se lee y se devuelve
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

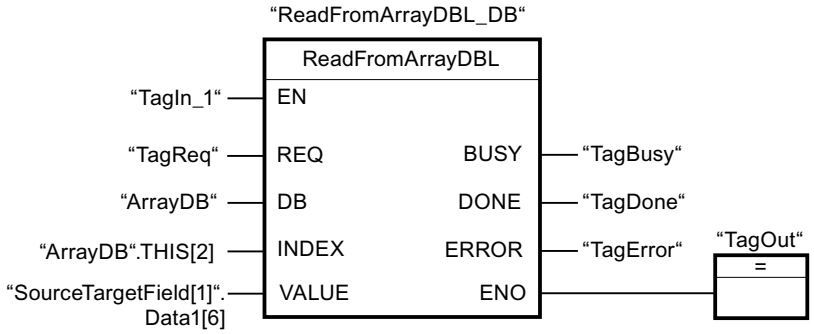
La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si el operando "TagIn1" devuelve el estado lógico "1" y en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga". El tercer elemento se lee en el "ArrayDB" y se indica en el operando "SourceTargetField[1].Data1[6]". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE ya no se modifica. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut". Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

WriteToArrayDBL: Escribir en un bloque de datos ARRAY de la memoria de carga

Descripción

La instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga" permite escribir datos en un DB de array de la memoria de carga.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un

flanco de señal descendente, finaliza la instrucción y el valor del parámetro VALUE se escribe en el bloque de datos. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Ocurre un error al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la escritura en el DB de array
DB	Input	DB_ANY	D	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	D, L	Valor que se escribe
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La escritura en el DB de array todavía no ha finalizado

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

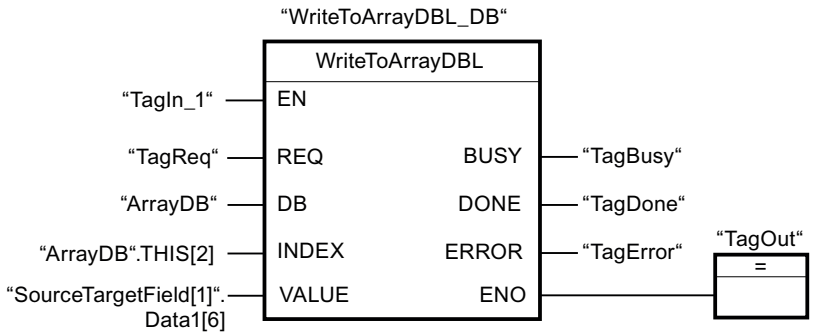
La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si el operando "TagIn1" devuelve el estado lógico "1" y en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE se escribe en el "ArrayDB" en el tercer elemento. Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut". Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Instrucciones VARIANT

VariantGet: Leer valor de una variable VARIANT

Descripción

Con la instrucción "Leer valor de una variable VARIANT" se lee el valor de la variable a la que señala VARIANT en el parámetro SRC y se escribe este valor en la variable del parámetro DST.

El parámetro SRC tiene que ser del tipo de datos VARIANT. En el parámetro DST se puede indicar cualquier tipo de datos excepto VARIANT.

El tipo de datos de la variable del parámetro DST debe coincidir con el tipo de datos al que señala VARIANT.

Nota

La variable a la que señala VARIANT en el parámetro SRC, debe ser un tipo de datos PLC.

Para copiar estructuras y ARRAY también se puede utilizar la instrucción "MOVE_BLK_VARIANT: Copiar área". Encontrará más información al respecto en "Consulte también".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Los tipos de datos no coinciden. (No se transmiten valores.)

Parámetro

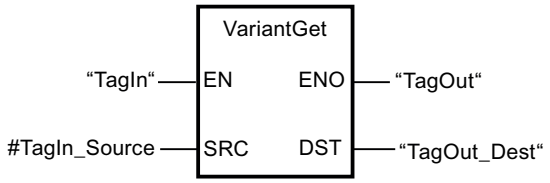
La tabla siguiente muestra los parámetros de la instrucción "Leer valor de una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC	Input	VARIANT	L	Variable que se lee
DST	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres, ARRAY	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Leer valor de una variable VARIANT". El valor de la variable al que señala VARIANT en el operando "#TagIn_Source" se lee y se escribe en el operando "TagOut_Dest".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

MOVE_BLK_VARIANT: Copiar área (Página 3768)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

VariantPut: Escribir valor en una variable VARIANT

Descripción

La instrucción "Escribir valor en una variable VARIANT" escribe el valor de la variable del parámetro SRC en la variable del parámetro DST a la que señala VARIANT.

El parámetro DST tiene que ser del tipo de datos VARIANT. En el parámetro SRC se puede indicar cualquier tipo de datos excepto VARIANT.

El tipo de datos de la variable del parámetro SRC debe coincidir con el tipo de datos al que señala VARIANT.

Nota

La variable a la que señala VARIANT en el parámetro DST, debe ser un tipo de datos PLC.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Los tipos de datos no coinciden. (No se transmiten valores.)

Parámetro

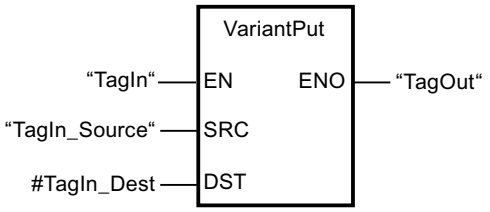
La tabla siguiente muestra los parámetros de la instrucción "Escribir valor en una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRC	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres, ARRAY	I, Q, M, D, L	Variable que se lee
DST	Input	VARIANT	L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Escribir valor en una variable VARIANT". El valor del operando "TagIn_Source" se escribe en la variable a la que señala VARIANT en el operando "#TagIn_Dest".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

CountOfElements: Consultar número de elementos de ARRAY

Descripción

La instrucción "Consultar número de elementos de ARRAY" permite saber el número de elementos de ARRAY que tiene una variable a la que apunta VARIANT.

Si el ARRAY es unidimensional, el resultado se devuelve como diferencia entre el límite superior e inferior + 1. Si el ARRAY es multidimensional, el resultado se devuelve como producto de todas las dimensiones.

El parámetro IN tiene que ser del tipo de datos VARIANT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- La variable VARIANT no es un ARRAY. (El resultado es "0".)

Si VARIANT apunta a un ARRAY of BOOL, se cuentan también los elementos de relleno. (P. ej., con un ARRAY[0..1] of BOOL se devuelve 8.)

Parámetro

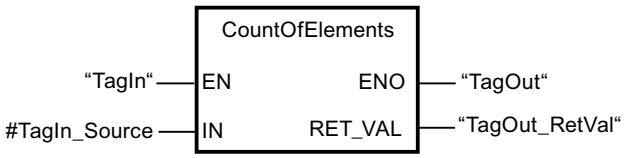
La tabla siguiente muestra los parámetros de la instrucción "Consultar número de elementos de ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	VARIANT	L	Variable que se consulta
RET_VAL	Output	UDINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Consultar número de elementos de ARRAY". El número de elementos de ARRAY de la variable a la que apunta VARIANT en el operando "#TagIn_Source" se lee y se devuelve en el operando "TagOut_RetVal".

Consulte también

Formato de Array (límites de 16 bits) (Página 2354)

Formato de Array (límites de 32 bits) (Página 2355)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

Otras

BLKMOV: Copiar área

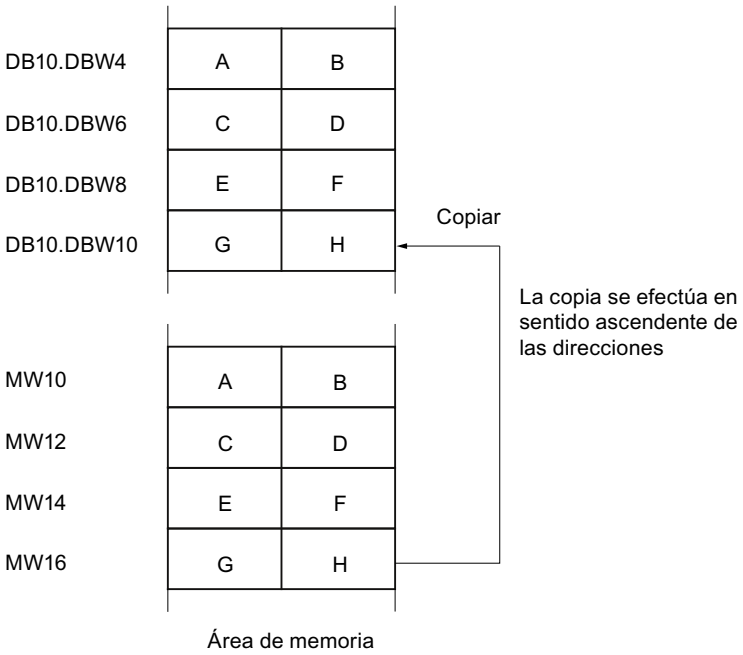
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

No hay limitaciones en cuanto a la profundidad de anidamiento.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	VARIANT	L, P	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8092	El área de origen o destino está solo en la memoria de carga.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	VARIANT	L, P	L, P	Indicación del área de memoria que se copia (área de origen).

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
RET_VAL	Output	INT	I, Q, M, D, L, P	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

FILL: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones.

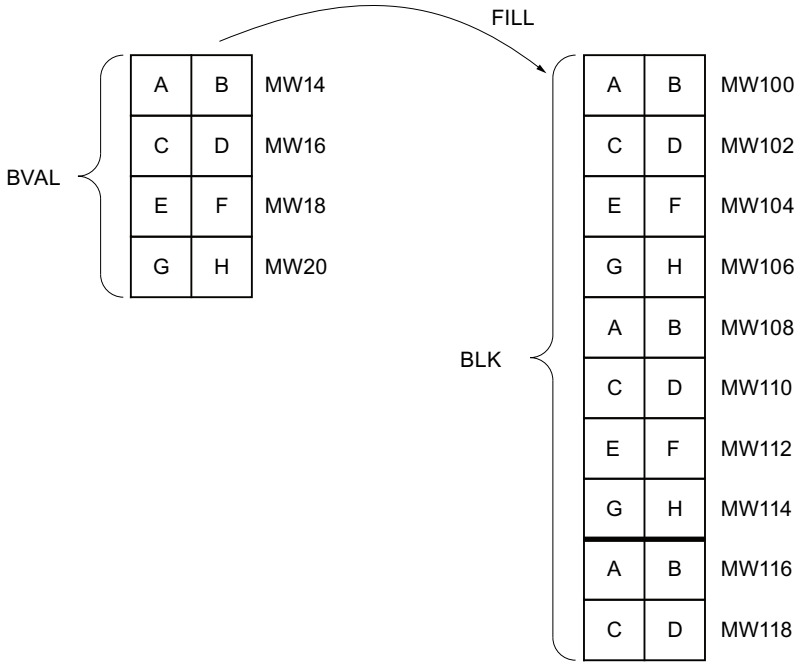
Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Para los bloques con el atributo "Acceso optimizado al bloque" puede utilizarse la instrucción "FILL_BLK: Rellenar área".

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
BVAL	Input	VARIANT	L, P	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
BLK	Output	VARIANT	L, P	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" lee el contenido del parámetro IN y lo convierte según los tipos de datos parametrizados en el cuadro de la instrucción. El valor convertido se deposita en la salida OUT.

Encontrará información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir valor":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros, números en coma flotante, CHAR, BCD16, BCD32	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor que se convierte.
OUT	Output	Secuencias de bits, enteros, números en coma flotante, CHAR, BCD16, BCD32	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado de la conversión

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

En el cuadro de la instrucción no se pueden seleccionar secuencias de bits (BYTE, WORD, DWORD, LWORD). Si se especifica un operando del tipo de datos BYTE, WORD, DWORD o LWORD en un parámetro de la instrucción, el operando se interpretará conforme al tipo de datos del parámetro de entrada o salida. El tipo de datos BYTE se interpreta entonces como USINT, WORD como UINT, DWORD como UDINT y LWORD como LINT.

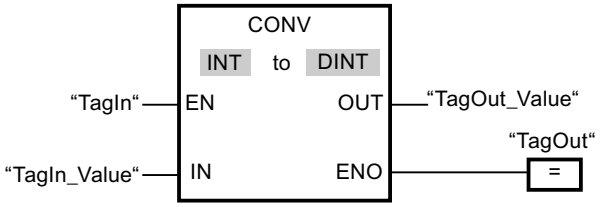
Nota

Para una CPU S7-1500 rige lo siguiente: existe la posibilidad de seleccionar los tipos de datos DWORD y LWORD si se ha seleccionado el tipo de datos IN REAL o LREAL.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", el contenido del operando "TagIn_Value" se lee y se convierte en un número entero (16 bits). El resultado se deposita en el operando "TagOut_Value". Si la instrucción se ha ejecutado correctamente, la salida "TagOut" se pone a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Conversión explícita (Página 2481)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor de la entrada IN al siguiente número entero. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte al número entero más próximo. Si el valor de entrada se encuentra entre dos números, se convierte el número par. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear número":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación

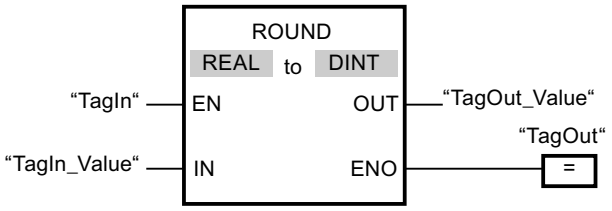
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada que se redondea.
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado del redondeo

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear número". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero par más próximo y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Conversión explícita (Página 2481)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor de la entrada IN al siguiente número entero superior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero superior. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí. El valor de salida puede ser mayor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero superior":

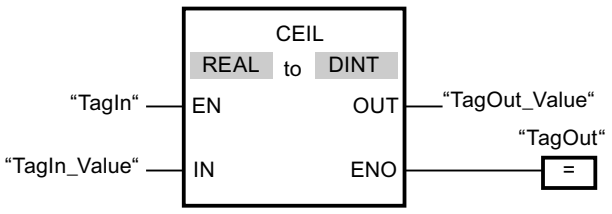
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada como número en coma flotante
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado con el siguiente entero superior

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero superior". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero superior más próximo y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Conversión explícita (Página 2481)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor de la entrada IN al siguiente número entero inferior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero inferior. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí. El valor de salida puede ser menor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero inferior":

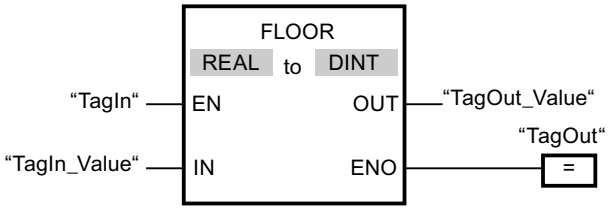
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada como número en coma flotante
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado con el siguiente entero inferior

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero inferior". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero inferior más próximo y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Conversión explícita (Página 2481)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un valor entero a partir del valor de la entrada IN. El valor de la entrada IN se interpreta como número en coma flotante. La instrucción selecciona sólo la parte entera del número en coma flotante y la deposita sin decimales en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Truncar a entero":

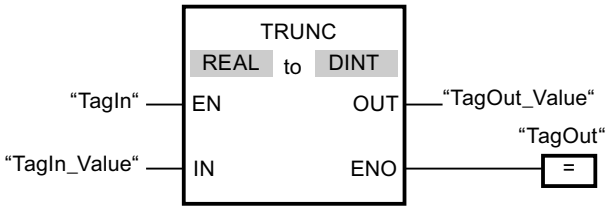
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L o constante	Valor de entrada como número en coma flotante
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L	Resultado con la parte entera del número en coma flotante

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Truncar a entero". La parte entera del número en coma flotante de la entrada "TagIn_Value" se convierte en un número entero y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Conversión explícita (Página 2481)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

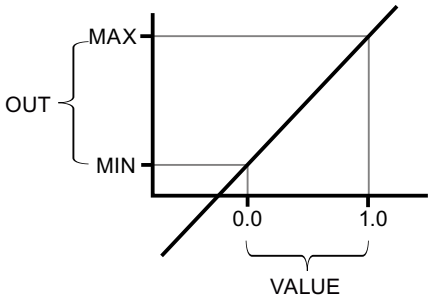
Seleccionar el tipo de datos de un elemento FUP (Página 2726)

SCALE_X: Escalar

Descripción

La instrucción "Escalar" escala el valor de la entrada VALUE mapeándolo en un determinado rango de valores. Al ejecutar la instrucción "Escalar", el número en coma flotante de la entrada VALUE se escala al rango de valores definido por los parámetros MIN y MAX. El resultado de la escala es un número entero que se deposita en la salida OUT.

La figura siguiente muestra un ejemplo de cómo pueden escalarse los valores:



La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [VALUE * (MAX – MIN)] + MIN

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada MIN es mayor o igual al valor de la entrada MAX.
- El valor de un número en coma flotante indicado está fuera del rango de los números normalizados según IEEE-754.
- Ocurre un rebase por exceso.
- El valor de la entrada VALUE es NaN (Not a number = resultado de una operación aritmética no válida).

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE	Input	Números en coma flotante	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Valor que se escala. Si se indica una constante, esta debe declararse.
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Límite superior del rango de valores
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L	I, Q, M, D, L	Resultado de la escala

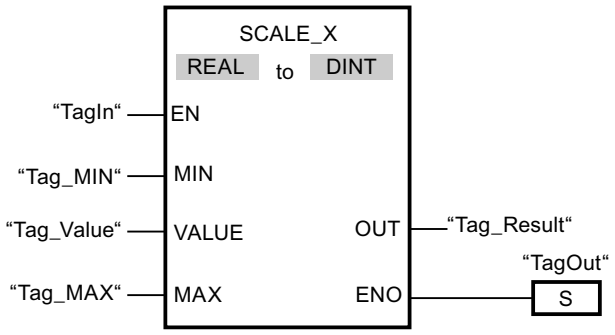
En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_MIN	10
VALUE	Tag_Value	0.5
MAX	Tag_MAX	30
OUT	Tag_Result	20

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Escalar". El valor de la entrada "Tag_Value" se escala al rango de valores definido por los valores de las entradas "Tag_MIN" y "Tag_MAX". El resultado se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

NORM_X: Normalizar (Página 3816)

Declarar constantes globales (Página 2584)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

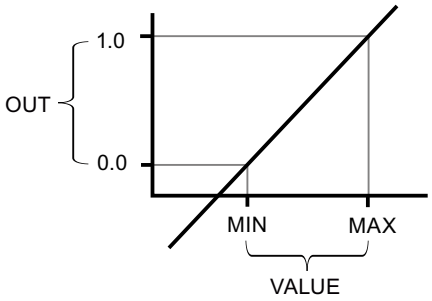
Cablear operandos en instrucciones FUP (Página 2745)

NORM_X: Normalizar

Descripción

La instrucción "Normalizar" normaliza el valor de la variable de la entrada VALUE representándolo en una escala lineal. Los parámetros MIN y MAX sirven para definir los límites del rango de valores que se refleja en la escala. En función de la posición del valor que se debe normalizar en este rango de valores, se calcula el resultado y se deposita como número en coma flotante en la salida OUT. Si el valor que se debe normalizar es igual al valor de la entrada MIN, la salida OUT devuelve el valor "0.0". Si el valor que se debe normalizar es igual al valor de la entrada MAX, la salida OUT devuelve el valor "1.0".

La figura siguiente muestra un ejemplo de cómo pueden normalizarse los valores:



La instrucción "Normalizar" utiliza la siguiente ecuación:

OUT = (VALUE – MIN) / (MAX – MIN)

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada MIN es mayor o igual al valor de la entrada MAX.
- El valor de un número en coma flotante indicado está fuera del rango de los números normalizados según IEEE-754.
- El valor de la entrada VALUE es NaN (resultado de una operación aritmética no válida).

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Normalizar":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
MIN [1]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE [1]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Valor que se normaliza.
MAX [1]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Límite superior del rango de valores
OUT	Output	Números en coma flotante	I, Q, M, D, L	I, Q, M, D, L	Resultado de la normalización
[1] Si se utilizan constantes en estos tres parámetros, basta con declarar una de ellas.					

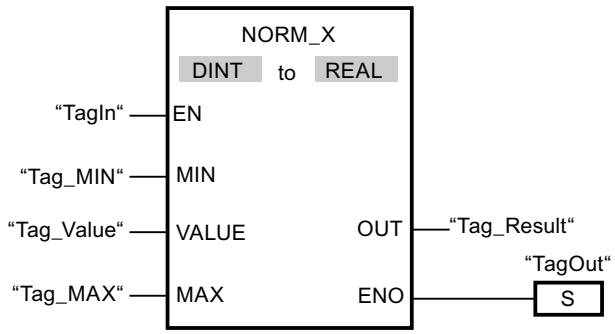
En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_MIN	10
VALUE	Tag_Value	20
MAX	Tag_MAX	30
OUT	Tag_Result	0.5

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Normalizar". El valor de la entrada "Tag_Value" se asigna al rango de valores definido por los valores de las entradas "Tag_MIN" y "Tag_MAX". El valor de la variable de la entrada "Tag_Value" se normaliza conforme al rango de valores definido. El resultado se deposita

como número en coma flotante en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

SCALE_X: Escalar (Página 3813)

Declarar constantes globales (Página 2584)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

Otras

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala inversamente proporcional al valor de entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	INT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada que se escala.
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	I, Q, M, D, L, P	Información de error

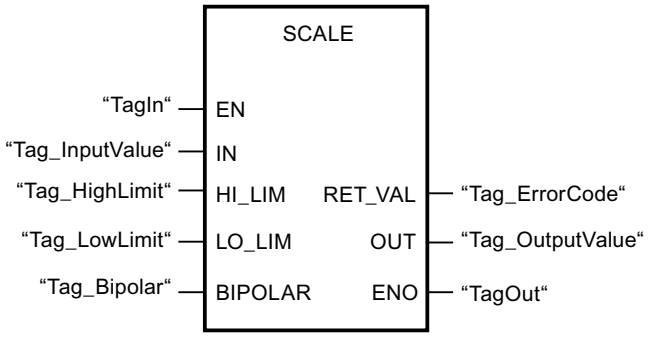
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1".
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" desescala el número en coma flotante del parámetro IN en unidades físicas entre un valor límite inferior y un valor límite superior y lo convierte en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el

valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero.
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
OUT	Output	INT	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	I, Q, M, D, L, P	Información de error

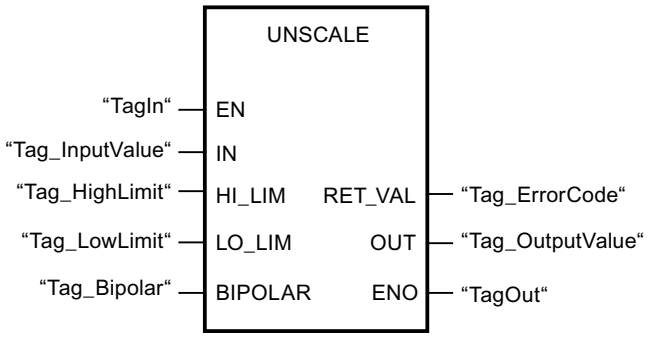
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

Control del programa

JMP: Saltar si RLO = 1

Descripción

La instrucción "Saltar si RLO = 1" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente. El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima del cuadro de la instrucción.

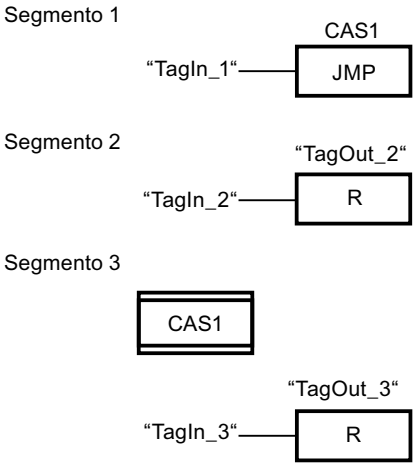
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En un segmento no se puede utilizar más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si no se cumple la condición en la entrada de la instrucción (RLO = 0), la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

JMPN: Saltar si RLO = 0

Descripción

La instrucción "Saltar si RLO = 0" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente, si el resultado lógico de la entrada de la instrucción es "0". El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima del cuadro de la instrucción.

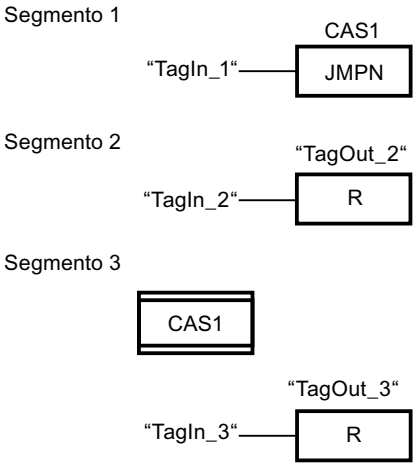
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En un segmento no se puede utilizar más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "0", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si el RLO de la entrada de la instrucción es "1", la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "0", se ejecuta la instrucción "Saltar si RLO = 0". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

LABEL: Etiqueta

Descripción

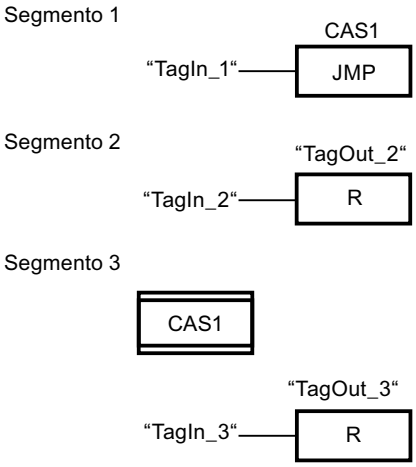
La etiqueta de salto marca un segmento de destino en el que se debe continuar la ejecución del programa tras ejecutar una instrucción de salto.

La etiqueta y la instrucción en la que se indica la etiqueta como destino del salto deben encontrarse en el mismo bloque. El nombre de una etiqueta debe ser unívoca en el bloque. Puede declarar un máximo de 32 etiquetas en caso de utilizar una CPU S7-1200 y un máximo de 256 etiquetas en caso de utilizar una CPU S7-1500.

En un segmento sólo se puede colocar una etiqueta. A toda etiqueta se puede acceder desde distintas posiciones.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

JMP_LIST: Definir lista de saltos

Descripción

La instrucción "Definir lista de saltos" permite definir varios saltos condicionados y continuar la ejecución del programa en un segmento determinado en función del valor del parámetro K.

Los saltos se definen mediante etiquetas (LABEL) que se indican en las salidas del cuadro de la instrucción. El número de salidas del cuadro de la instrucción se puede ampliar. Puede declarar un máximo de 32 salidas en caso de utilizar una CPU S7-1200 y un máximo de 99 salidas en caso de utilizar una CPU S7-1500.

La numeración de las salidas comienza por el valor "0" y continúa en orden ascendente con cada nueva salida. En las salidas de la instrucción únicamente se pueden indicar etiquetas. No está permitido indicar instrucciones u operandos.

Con el valor del parámetro K se indica el número de la salida y a la vez la etiqueta en la que debe continuarse la ejecución del programa. Si el valor del parámetro K es mayor que el número de salidas disponibles, la ejecución del programa continúa en el siguiente segmento del bloque.

La instrucción "Definir lista de saltos" se ejecuta solo si el estado lógico de la entrada de habilitación EN es "1".

Parámetro

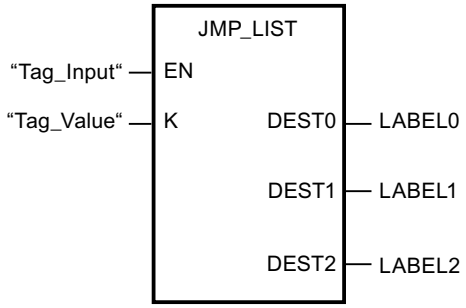
La tabla siguiente muestra los parámetros de la instrucción "Definir lista de saltos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, L, D	Entrada de habilitación
K	Input	UINT	I, Q, M, L, D o constante	Indica el número de la salida y a la vez el salto que se ejecuta.
DEST0	-	-	-	Primera etiqueta
DEST1	-	-	-	Segunda etiqueta
DESTn	-	-	-	Etiquetas opcionales

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando/Etiqueta	Valor
K	Tag_Value	1
DEST0	LABEL0	Salto al segmento que está marcado con la etiqueta "LABEL0".
DEST1	LABEL1	Salto al segmento que está marcado con la etiqueta "LABEL1".
DEST2	LABEL2	Salto al segmento que está marcado con la etiqueta "LABEL2".

Si el operando "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Definir lista de saltos". La ejecución del programa continúa conforme al valor del operando "Tag_Value" en el segmento que está marcado con la etiqueta "LABEL1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

SWITCH: Distribuidor de saltos

Descripción

La instrucción "Distribuidor de saltos" permite definir varios saltos de programa, los cuales se ejecutan en función del resultado de una o varias instrucciones de comparación.

El valor que se va a comparar se especifica en el parámetro K. Este valor se compara con los valores que devuelven las distintas entradas. El tipo de comparación se selecciona individualmente para cada entrada. La disponibilidad de las diferentes instrucciones de comparación depende del tipo de datos de la instrucción.

La tabla siguiente muestra las instrucciones de comparación disponibles en función del tipo de datos elegido:

Tipo de datos		Instrucción	Sintaxis
S7-1200	S7-1500		
Secuencias de bits	Secuencias de bits	Igual	==
		Diferente	<>
Enteros, números en coma flotante, TIME, DATE, TOD	Enteros, números en coma flotante, TIME,	Igual	==

Tipo de datos		Instrucción	Sintaxis
S7-1200	S7-1500		
	LTIME, DATE, TOD, LTOD, LDT	Diferente	<>
		Mayor o igual	>=
		Menor o igual	<=
		Mayor	>
		Menor	<

El tipo de datos de la instrucción se selecciona en la lista desplegable "<???>" del cuadro de la instrucción. Si selecciona una instrucción de comparación sin que esté definido el tipo de datos de la instrucción, en la lista desplegable "<???>" sólo aparecen los tipos de datos permitidos para la instrucción de comparación seleccionada.

La ejecución de la instrucción comienza por la primera comparación y se sigue ejecutando hasta que se cumpla una condición de comparación. Cuando se cumple una condición de comparación, no se tienen en cuenta las siguientes condiciones de comparación. Si no se cumple ninguna de las condiciones de comparación indicadas, el salto se ejecuta en la salida ELSE. Si en la salida ELSE no hay definida ninguna etiqueta, no se interrumpe la ejecución lineal del programa, sino que continúa en el segmento siguiente.

El cuadro de la instrucción contiene en estado estacionario mín. 2 salidas (DEST0 y DEST1). El número de salidas es ampliable. La numeración de las salidas comienza por el valor "0" y continúa en orden ascendente con cada nueva salida. Indique etiquetas (LABEL) en las salidas de la instrucción. No está permitido indicar instrucciones u operandos en las salidas de la instrucción.

Por cada salida adicional se inserta automáticamente una entrada. El salto programado en una salida se ejecuta cuando se cumple la condición de comparación de la entrada correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Distribuidor de saltos":

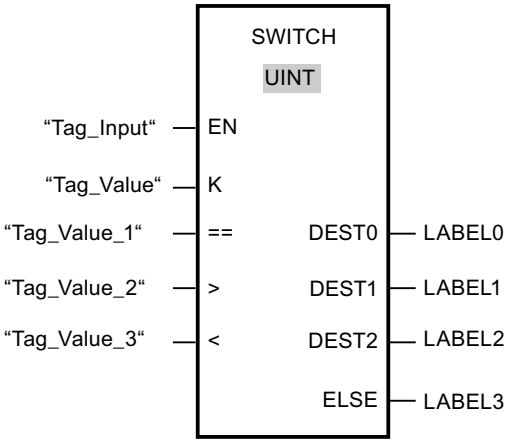
Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
EN	Input	BOOL	BOOL	I, Q, M, D, L	Entrada de habilitación
K	Input	UINT	UINT	I, Q, M, D, L o constante	Indica el valor que se compara.
<Valores de comparación>	Input	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD	Secuencias de bits, enteros, números en coma flotante, TIME, LTIME, DATE, TOD, LTOD, LDT	I, Q, M, D, L o constante	Valores de entrada con los que se compara el valor del parámetro K.

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
DEST0	-	-	-	-	Primera etiqueta
DEST1	-	-	-	-	Segunda etiqueta
DEST(n)	-	-	-	-	Etiquetas opcionales (n = 2 hasta 99)
ELSE	-	-	-	-	Salto de programa que se ejecuta cuando no se cumple ninguna de las condiciones de comparación.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando/Etiqueta	Valor
K	Tag_Value	23
==	Tag_Value_1	20
>	Tag_Value_2	21
<	Tag_Value_3	19
DEST0	LABEL0	Salto a la etiqueta "LABEL0" si el valor del parámetro K es igual a 20.
DEST1	LABEL1	Salto a la etiqueta "LABEL1" si el valor del parámetro K es mayor que 21.

Parámetro	Operando/Etiqueta	Valor
DEST2	LABEL2	Salto a la etiqueta "LABEL2" si el valor del parámetro K es menor que 19.
ELSE	LABEL 3	Salto a la etiqueta "LABEL3" si no se cumple ninguna de las condiciones de comparación.

Si el operando "Tag_Input" cambia al estado lógico "1", se ejecuta la instrucción "Distribuidor de saltos". La ejecución del programa continúa en el segmento que está marcado con la etiqueta "LABEL1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

RET: Retroceder

Descripción

La instrucción "Retroceder" permite finalizar la ejecución de un bloque. De este modo se obtienen tres formas posibles de terminar la ejecución de un bloque:

- Sin llamada de la instrucción "Retroceder"
  Tras ejecutar el último segmento, se abandona el bloque. El ENO de la función de llamada adopta el estado lógico "1".
- Llamada de la instrucción "Retroceder" con operación lógica precedente (ver ejemplo)
  Si la conexión izquierda tiene el estado lógico "1", se abandona el bloque. El ENO de la función de llamada equivale al operando.
- Llamada de la instrucción "Retroceder" sin operación lógica precedente
  Se abandona el bloque. El ENO de la función de llamada equivale al operando.

Nota

En un segmento no se puede utilizar más de una bobina de salto ("Retroceder", "Saltar si RLO = 1", "Saltar si RLO = 0").

Si el resultado lógico (RLO) de la entrada de la instrucción "Retroceder" es "1", se finaliza la ejecución del programa en el bloque llamado actualmente y se continúa en el bloque que efectúa la llamada (p. ej. en el OB que llama) después de la función de llamada. El parámetro

de la instrucción determina el estado (ENO) de la función de llamada. Puede adoptar los valores siguientes:

- RLO
- TRUE/FALSE
- <Operando>

Haga doble clic en la instrucción para ajustar los valores del parámetro y seleccione el valor correspondiente en la lista desplegable.

La tabla siguiente muestra el estado de la función de llamada cuando la función "Retroceder" está programada en un segmento del bloque llamado:

RLO	Valor del parámetro	ENO de la función de llamada
1	RLO	1
	TRUE	1
	FALSE	0
	<Operando>	<Operando>
0	RLO	La ejecución del programa continúa en el siguiente segmento del bloque llamado.
	TRUE	
	FALSE	
	<Operando>	

Cuando se finaliza un OB, el sistema de niveles secuenciales selecciona otro bloque para luego iniciarlo o seguir ejecutándolo:

- Si se ha finalizado el OB de ciclo de programa, éste se reinicia.
- Cuando se finaliza un OB que ha interrumpido otro bloque (p. ej. un OB de alarma), se ejecuta el bloque interrumpido (p. ej. OB de ciclo de programa).

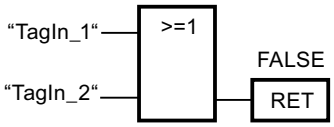
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Retroceder":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Estado de la función que efectúa llamada para RLO = 1:				
RLO	-	-	-	Se ajusta al estado lógico del RLO.
TRUE	-	-	-	1
FALSE	-	-	-	0
<Operando>	Input	BOOL	I, Q, M, D, L	Estado lógico del operando indicado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de uno de los operandos "TagIn_1" o "TagIn_2" es "1", se ejecuta la instrucción "Retroceder". La ejecución del programa se finaliza en el bloque llamado y continúa en el bloque que efectúa la llamada. La salida de habilitación ENO de la función de llamada adopta el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Control del tiempo de ejecución

ENDIS_PW: Limitar y habilitar la legitimación de contraseñas

Descripción

La instrucción "Limitar y habilitar legitimación de la contraseña" permite definir si las contraseñas configuradas para la CPU están legitimadas o no. Con ello, se pueden impedir conexiones legitimadas, aun conociendo la contraseña correcta.

Si la instrucción se ejecuta y el parámetro REQ tiene el estado lógico "0", en los parámetros de salida se muestra el estado ajustado en ese momento. Si se han realizado cambios en los parámetros de entrada, estos no se transfieren a los parámetros de salida.

Si se ejecuta la instrucción y el parámetro REQ tiene el estado lógico "1", se aplica el estado lógico de los parámetros de entrada (F_PWD, FULL_PWD, R_PWD, HMI_PWD). FALSE significa que la legitimación por contraseña no está permitida y TRUE significa que la contraseña puede utilizarse.

Es posible permitir o prohibir individualmente el bloqueo o la habilitación de las contraseñas. Por ejemplo, pueden prohibirse todas las contraseñas excepto la contraseña de seguridad positiva. Esto permite limitar las posibilidades de acceso a un reducido grupo de usuarios. Los parámetros de salida (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) siempre muestran el estado actual del uso de contraseñas, independientemente del parámetro REQ.

Las contraseñas no configuradas deben tener en la entrada el estado lógico TRUE y devolver el estado lógico TRUE en la salida. La contraseña de seguridad positiva (fail-safe) solo se puede parametrizar para una CPU F y por eso en una CPU estándar debe interconectarse siempre con el estado lógico TRUE. Si la instrucción devuelve un error, la llamada permanece sin efecto, es decir, el bloqueo anterior sigue en vigor.

Las contraseñas bloqueadas pueden volver a habilitarse en las condiciones siguientes:

- Se ha restablecido la configuración de fábrica de la CPU.
- El panel frontal de la CPU S7-1500 soporta un cuadro de diálogo que permite navegar al menú adecuado y volver a habilitar las contraseñas.
- Cuando se llama la instrucción "Limitar y habilitar legitimación de la contraseña", el parámetro de entrada de la contraseña deseada tiene el estado lógico "1".
- Ponga el selector de modo a STOP. La restricción de la legitimación de contraseña se activa de nuevo en cuanto se mueve el selector nuevamente a RUN.
- La inserción de una tarjeta de memoria vacía (tarjeta de transferencia o tarjeta de programa) en una CPU S7-1200.
- La transición de POWER OFF-POWER ON desactiva la protección en la CPU S7-1200. La instrucción "Limitar y habilitar legitimación de la contraseña" debe volverse a llamar en el programa (por ejemplo en el OB de arranque).

Nota

Si la contraseña HMI no está habilitada, la instrucción "Limitar y habilitar legitimación de la contraseña" bloquea el acceso de los sistemas HMI.

Nota

Las conexiones ya existentes y legitimadas conservan sus derechos de acceso y no pueden limitarse con la instrucción "Limitar y habilitar legitimación de la contraseña".

Impedir el bloqueo no intencionado en una CPU S7-1500

Los ajustes pueden realizarse en el panel frontal de la CPU, que guarda el último ajuste en cada caso.

Para impedir un bloqueo no intencionado, en una CPU S7-1500 es posible desactivar la protección moviendo el selector de modo a STOP. La protección se activa automáticamente después de mover el selector de modo a RUN sin que tenga que volver a llamarse la instrucción "Limitar y habilitar legitimación de la contraseña" o deban realizarse otras acciones en el panel frontal.

Impedir el bloqueo no intencionado en una CPU S7-1200

Como la CPU S7-1200 no dispone de selector de modo, la protección se desactiva con POWER OFF-POWER ON. De este modo es posible y recomendable impedir el bloqueo no intencionado con ayuda de determinadas secuencias en el programa.

Para ello, programe un control de tiempo, ya sea mediante un OB de alarma cíclica o un temporizador en el Main OB (OB 1). De este modo, tras una transición de POWER OFF-POWER ON y la consiguiente desactivación de la protección, se puede volver a llamar rápidamente la instrucción "Limitar y habilitar legitimación de la contraseña" en el OB correspondiente (por ejemplo OB 1 u OB 35). Para que la ventana temporal en la que la instrucción no está activa y en la que por tanto no existen limitaciones en la legitimación de

la contraseña sea lo más pequeña posible, llame la instrucción en el OB de arranque (OB 100). Este procedimiento ofrece la mayor protección posible contra accesos no autorizados.

Si se ha producido un bloqueo no intencionado puede omitirse la llamada en el OB de arranque (por ejemplo mediante la consulta de un parámetro de entrada) y se dispondrá del tiempo configurado (por ejemplo de 10 segundos a 1 minuto) para establecer una conexión con la CPU antes de que el bloqueo vuelva a activarse.

Si no se ha previsto ningún temporizador en el código del programa y se produce un bloqueo, inserte en la CPU una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna de la CPU. Después deberá cargarse nuevamente el programa de usuario de STEP 7 en la CPU.

Procedimiento en caso de pérdida de contraseña en una CPU S7-1200

Si ha perdido la contraseña de una CPU S7-1200 protegida por contraseña, borre el programa protegido por contraseña con una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna de la CPU. A continuación, puede transferir un nuevo programa de usuario de STEP 7 Basic a la CPU.

⚠ ADVERTENCIA

Inserción de una tarjeta de transferencia vacía

Cuando se inserta una tarjeta de transferencia en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

El contenido de la tarjeta de transferencia se encuentra en la memoria de carga interna después de su extracción. Compruebe aquí que la tarjeta no contenga ningún programa.

⚠ ADVERTENCIA

Inserción de una tarjeta de programa vacía

Cuando se inserta una tarjeta de programa en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

Asegúrese de que la tarjeta de programa esté vacía. La memoria de carga interna se copiará en la tarjeta de programa vacía. Después de retirar la tarjeta de programa previamente vacía, la memoria de carga interna quedará vacía.

Debe retirarse la tarjeta de transferencia o la tarjeta de programa antes de poner la CPU en RUN.

Efectos del uso de contraseñas en los modos de operación

La siguiente tabla muestra qué efectos tiene el uso de contraseñas en los modos de operación y en las respectivas acciones del usuario mediante la instrucción "Limitar y habilitar legitimación de la contraseña".

Acción	Protección por contraseña mediante la instrucción
Estado básico después de • Selector de modo en STOP • Restablecimiento manual de la memoria (PG, interruptor, cambio de MC (Motion Control)) • Restablecimiento de la configuración de fábrica	No activado (sin limitaciones)
Estado básico tras conexión (POWER ON)	• CPU S7-1200: El bloqueo está desactivado y la instrucción tiene que volverse a ejecutar en el programa (por ejemplo en el OB de arranque). • CPU S7-1500: Activada (si antes de la desconexión (POWER OFF) había un bloqueo activo) La posibilidad de no permitir contraseñas es remanente.
Transición del estado operativo RUN/ARRANQUE/PARADA -> STOP (mediante finalización de la instrucción, un error o comunicación) o STOP -> ARRANQUE/RUN/PARADA	Activado Las contraseñas siguen sin poder utilizarse.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Limitar y habilitar legitimación de la contraseña":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
REQ	Input	BOOL	I, Q, M, D, L o constante	Si el parámetro REQ tiene el estado lógico "0", se consulta el estado lógico ajustado actualmente para las contraseñas.
F_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura incluido failsafe • F_PWD = "0": No permitir contraseña • F_PWD = "1": Permitir contraseña
FULL_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura • FULL_PWD = "0": No permitir contraseña • FULL_PWD = "1": Permitir contraseña
R_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura • R_PWD = "0": No permitir contraseña • R_PWD = "1": Permitir contraseña

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
HMI_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso a la HMI • HMI_PWD = "0": No permitir contraseña • HMI_PWD = "1": Permitir contraseña
F_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura incluido failsafe • F_PWD_ON = "0": Contraseña no permitida • F_PWD_ON = "1": Contraseña permitida
FULL_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura • FULL_PWD_ON = "0": Contraseña no permitida • FULL_PWD_ON = "1": Contraseña permitida
R_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura • R_PWD_ON = "0": Contraseña no permitida • R_PWD_ON = "1": Contraseña permitida
HMI_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso a la HMI • HMI_PWD_ON = "0": Contraseña no permitida • HMI_PWD_ON = "1": Contraseña permitida
RET_VAL	Output	WORD	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8090	No se soporta la instrucción "Limitar y habilitar legitimación de la contraseña".
80D0	La contraseña de seguridad (fail-safe) no está configurada. En las CPU estándar, el estado lógico tiene que ser TRUE.
80D1	El acceso de lectura/escritura no está configurado
80D2	El acceso de lectura no está configurado
80D3	El acceso HMI no está configurado
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos del tratamiento de errores (Página 2932)

Principios del tratamiento de errores local (Página 2934)

Prioridades en la salida de errores (Página 2935)

Activar el tratamiento de errores local para un bloque (Página 2936)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del tiempo de ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar en todos los bloques independientemente de la prioridad.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar hasta 10 veces en un ciclo de programa.

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros.

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos del tratamiento de errores (Página 2932)

Principios del tratamiento de errores local (Página 2934)

Prioridades en la salida de errores (Página 2935)

Activar el tratamiento de errores local para un bloque (Página 2936)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", la CPU pasa al estado operativo STOP y se finaliza el procesamiento del programa. El estado lógico de la salida de la instrucción no se evalúa.

Si el RLO de la entrada de la instrucción es "0", la instrucción no se ejecuta.

Parámetro

La instrucción "Finalizar programa" no tiene parámetros.

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones FUP (Página 2745)

GET_ERROR: Consultar error localmente

Descripción

La instrucción "Consultar error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si el sistema notifica errores mientras se ejecuta el bloque, en el operando de la salida ERROR se guarda información detallada acerca del primer error de ejecución ocurrido desde la última vez que se ejecutó la instrucción.

En la salida ERROR solo pueden especificarse operandos del tipo de datos de sistema "ErrorStruct". El tipo de datos de sistema "ErrorStruct" especifica la estructura exacta con la que debe guardarse la información acerca del error que ha ocurrido. Con ayuda de otras instrucciones se puede evaluar esta estructura y programar la reacción correspondiente. Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces emite información sobre el siguiente error ocurrido.

Nota

La salida ERROR se modifica solo si existe información de error. Para poner a "0" la salida después del tratamiento de errores, dispone de las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz del bloque.
- Ponga a "0" la variable antes de llamar la instrucción.
- Consulte la salida de habilitación ENO.

La salida de habilitación ENO de la instrucción "Consultar error localmente" se activa únicamente si la entrada de habilitación EN devuelve el estado lógico "1" y existe información de error. Si no se cumple alguna de estas condiciones, la ejecución ulterior del programa no se ve afectada por la instrucción "Consultar error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar error localmente" activa el tratamiento local de errores en el bloque. Si en la lógica de un bloque se ha insertado "Consultar error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	ErrorStruct	D, L	Información de error

Tipo de datos "ErrorStruct"

La tabla siguiente muestra la estructura del tipo de datos "ErrorStruct":

Componente de estructura		Tipo de datos	Descripción
ERROR_ID		WORD	ID de error
FLAGS		BYTE	Muestra si el error se ha producido durante la llamada de un bloque. 16#01: Error durante la llamada de un bloque 16#00: No hay errores durante la llamada de un bloque
REACTION		BYTE	Reacción predeterminada: 0: ignorar (error de escritura), 1: continuar con el valor sustitutivo "0" (error de lectura), 2: omitir instrucción (error de sistema)
CODE_ADDRESS		CREF	Información sobre la dirección y el tipo del bloque
	BLOCK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error: 1: OB 2: FC 3: FB
	CB_NUMBER	UINT	Número del bloque lógico
	OFFSET	UDINT	Referencia a la memoria interna
MODE		BYTE	Información sobre la dirección de un operando

Componente de estructura		Tipo de datos	Descripción
OPERAND_NUMBER		UINT	Número de operando del comando de la máquina
POINTER_NUMBER_LOCATION		UINT	(A) Puntero interno
SLOT_NUMBER_SCOPE		UINT	(B) Área de almacenamiento en la memoria interna
DATA_ADDRESS		NREF	Información sobre la dirección de un operando
	AREA	BYTE	(C) Área de memoria: L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE I: 16#81 Q: 16#82 M: 16#83 DB: 16#84, 85, 8A, 8B Infracciones de área en una variable editable directamente del tipo de datos DINT: 16#04
	DB_NUMBER	UINT	(D) Número del bloque de datos
	OFFSET	UDINT	(E) Dirección relativa del operando

Componente de estructura "ERROR_ID"

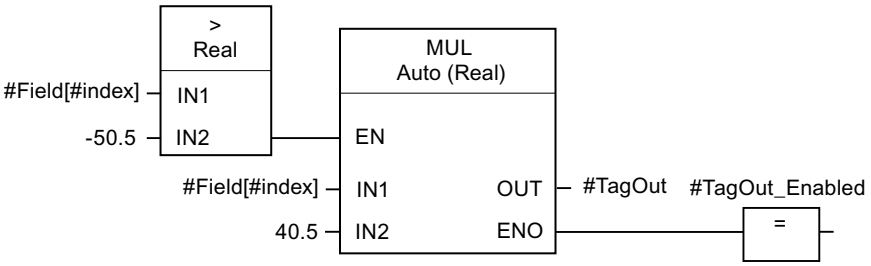
La tabla siguiente muestra los valores que pueden devolverse en el componente de estructura "ERROR_ID":

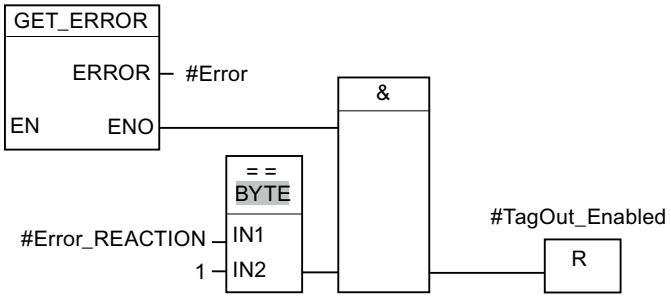
ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe

ID* (hexadecimal)	ID* (decimal)	Descripción
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:





Se ha producido un error al acceder a la variable "#Field[#index]". La salida de habilitación ENO de la instrucción "Multiplicar" y el operando "#TagOut_Enabled" devuelven el estado lógico "1" a pesar del error de lectura/escritura, y la multiplicación se realiza con el valor "0.0". En este caso de error, es recomendable programar la instrucción "Consultar error localmente" después de la instrucción "Multiplicar" para interceptar el error. La información de error que devuelve la instrucción "Consultar error localmente" se evalúa mediante la instrucción de

comparación "Igual". Si el componente de estructura "#Error.REACTION" tiene el valor "1", se trata de un error de lectura/escritura y la salida "#TagOut_Enabled" se pone a cero.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos del tratamiento de errores (Página 2932)

Principios del tratamiento de errores local (Página 2934)

Prioridades en la salida de errores (Página 2935)

Activar el tratamiento de errores local para un bloque (Página 2936)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Consultar y eliminar errores en el código del programa (Página 265)

GET_ERR_ID: Consultar ID de error localmente

Descripción

La instrucción "Consultar ID de error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si al ejecutar el bloque el sistema notifica errores de ejecución desde que se ejecutó la instrucción por última vez, la ID de error del primer error ocurrido se guarda en la variable de la salida ID.

En la salida ID solo pueden especificarse operandos del tipo de datos "WORD". Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces la instrucción emite la ID de error del siguiente error ocurrido.

Nota

La salida ID se modifica solo si existe información de error. Para poner a "0" la salida después del tratamiento de errores, dispone de las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz del bloque.
- Ponga a "0" la variable antes de llamar la instrucción.
- Consulte la salida de habilitación ENO.

La salida de habilitación ENO de la instrucción "Consultar ID de error localmente" se activa únicamente si la entrada de habilitación EN devuelve el estado lógico "1" y existe información de error. Si no se cumple alguna de estas condiciones, la ejecución ulterior del programa no se ve afectada por la instrucción "Consultar ID de error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar ID de error localmente" activa el tratamiento local de errores dentro del bloque. Si en la lógica de un bloque se ha insertado la instrucción "Consultar ID de error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar ID de error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Output	WORD	I, Q, M, D, L	ID de error

Parámetro ID

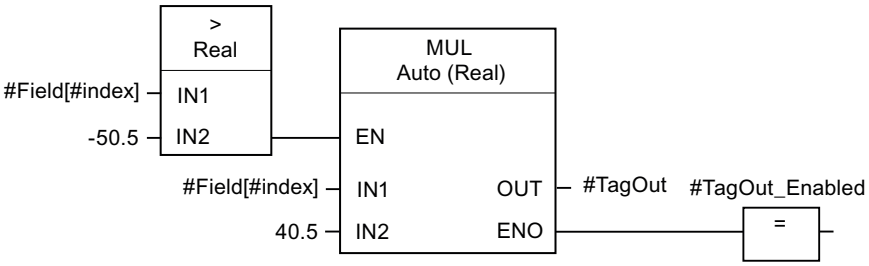
La tabla siguiente muestra los valores que pueden depositarse en el parámetro ID:

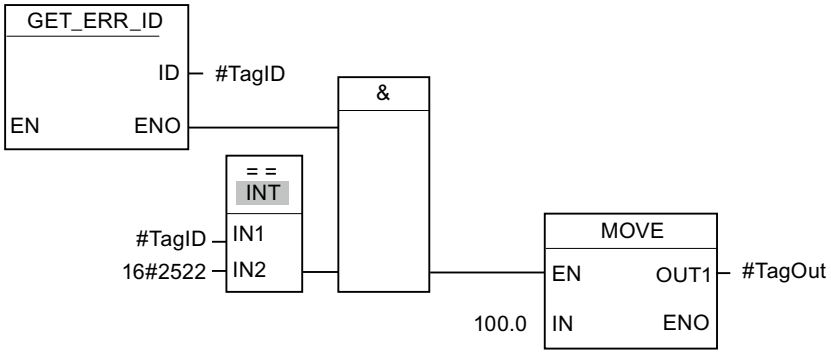
ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa

ID* (hexadecimal)	ID* (decimal)	Descripción
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:





Se ha producido un error al acceder a la variable "#Field[#index]". La salida de habilitación ENO de la instrucción "Multiplicar" y el operando "#TagOut_Enabled" devuelven el estado lógico "1" a pesar del error de lectura/escritura, y la multiplicación se realiza con el valor "0.0". En este caso de error, es recomendable programar la instrucción "Consultar ID de error localmente" después de la instrucción "Multiplicar" para interceptar el error. La información de error que devuelve la instrucción "Consultar ID de error localmente" se evalúa mediante la instrucción de comparación "Igual". Si el operando "#TagID" devuelve la ID 2522, se trata de un error de lectura/escritura y se escribe el valor "100.0" en la salida #TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Insertar elementos FUP (Página 2722)
Editar elementos FUP (Página 2738)
Cablear operandos en instrucciones FUP (Página 2745)
Principios básicos del tratamiento de errores (Página 2932)
Principios del tratamiento de errores local (Página 2934)
Prioridades en la salida de errores (Página 2935)
Activar el tratamiento de errores local para un bloque (Página 2936)
Cambiar formatos de visualización en el estado del programa (Página 3100)
Consultar y eliminar errores en el código del programa (Página 265)

INIT_RD: Inicializar todos los datos remanentes

Descripción

La instrucción "Inicializar todos los datos remanentes" permite inicializar todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC al mismo tiempo. La instrucción solo puede ejecutarse dentro de un OB de arranque, puesto que su ejecución excedería el tiempo de ciclo de programa.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar todos los datos remanentes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	Si la entrada REQ devuelve el estado lógico "1", se inicializan todos los datos remanentes.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

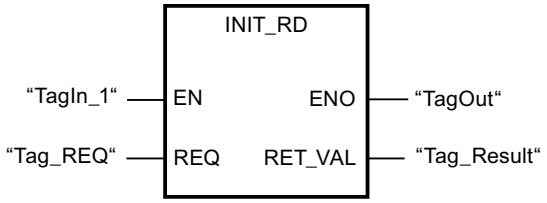
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B5	No puede ejecutarse la instrucción porque no ha sido programada dentro de un OB de arranque.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si los operandos "TagIn_1" y "Tag_REQ" devuelven el estado lógico "1", se ejecuta la instrucción. Se inicializan todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC. Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos del tratamiento de errores (Página 2932)

Principios del tratamiento de errores local (Página 2934)

Prioridades en la salida de errores (Página 2935)

Activar el tratamiento de errores local para un bloque (Página 2936)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3844)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" permite detener la ejecución del programa durante un periodo predeterminado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible programar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

RUNTIME: Medir tiempo de ejecución

Descripción

La instrucción "Medir tiempo de ejecución" permite medir el tiempo de ejecución de todo el programa, bloques individuales o secuencias de comandos.

Si desea medir el tiempo de ejecución de todo el programa, efectúe la llamada de la instrucción "Medir tiempo de ejecución" en el OB1. El punto inicial para medir el tiempo de ejecución se determina con la primera llamada y la salida RET_VAL devuelve el tiempo de ejecución del programa tras la segunda llamada. El tiempo de ejecución medido incluye toda clase de procesos de la CPU que se hayan producido durante la ejecución del programa, como, p. ej., interrupciones a causa de eventos o comunicaciones de mayor prioridad. La instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU y escribe su valor en el parámetro de entrada/salida MEM. La instrucción calcula el tiempo de ejecución actual del programa según la frecuencia interna del contador, y la escribe en la salida RET_VAL.

Si desea medir el tiempo de ejecución de bloques o secuencias de comandos individuales, necesita tres segmentos diferentes. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" en un segmento individual dentro del programa. Al efectuar esta primera llamada de la instrucción se establece el punto inicial para medir el tiempo de ejecución. A continuación, llame el bloque de programa o la secuencia de comando que desee en el siguiente segmento. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" por segunda vez en otro segmento diferente y asigne al parámetro de entrada/salida MEM la misma memoria que en la primera llamada de la instrucción. En el tercer segmento, la instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU, calcula el tiempo de ejecución actual del bloque de programa o secuencia de comando según la frecuencia interna del contador y lo escribe en la salida RET_VAL.

La instrucción "Medir tiempo de ejecución" utiliza un contador interno de alta frecuencia para calcular el tiempo. Cuando el contador se desborda, la instrucción devuelve valores <= 0.0. Estos valores de runtime deben ignorarse.

Nota

Puesto que el orden de las instrucciones de una secuencia de comandos se modifica mediante la compilación optimizada del programa, no es posible determinar el tiempo de ejecución de una secuencia de comandos con exactitud.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Medir tiempo de ejecución":

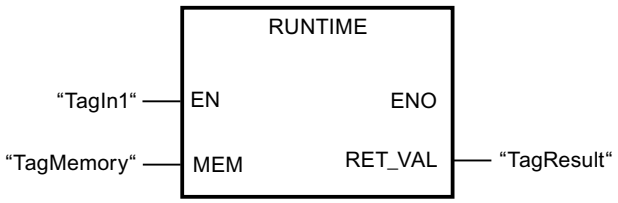
Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MEM	InOut	LREAL	M, D, L	Guarda el punto inicial de la medición del tiempo de ejecución
RET_VAL	Output	LREAL	I, Q, M, D, L	Devuelve el tiempo de ejecución medido en segundos

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

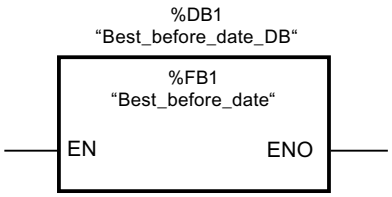
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción mediante el cálculo del tiempo de ejecución de un bloque de programa:

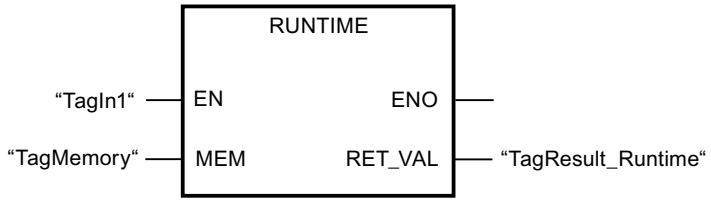
Segmento 1:



Segmento 2:



Segmento 3:



Si el operando "TagIn1" devuelve el estado lógico "1" en el segmento 1, se ejecuta la instrucción. Con la primera llamada de la instrucción se determina el punto inicial para la medición del tiempo de ejecución, y se guarda como referencia en una memora intermedia del operando "TagMemory" para la segunda llamada de la instrucción.

En el segmento 2 se llama el bloque de programa FB1 "Best_before_date".

Si el bloque de programa FB1 se ha procesado y el operando "TagIn1" devuelve el estado lógico "1", la instrucción se ejecuta en el segmento 3. La segunda llamada de la instrucción calcula el tiempo de ejecución del bloque de programa y escribe el resultado en la salida RET_VAL.

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Vista general de los tipos de datos válidos (Página 2321)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos del tratamiento de errores (Página 2932)

Principios del tratamiento de errores local (Página 2934)

Prioridades en la salida de errores (Página 2935)

Activar el tratamiento de errores local para un bloque (Página 2936)

Operaciones lógicas con palabras

AND: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una Y lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina mediante una Y lógica con el bit 0 del valor de la entrada IN2. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits de los valores indicados.

El cuadro de la instrucción contiene en estado estacionario mín. 2 entradas (IN1 e IN2). El número de entradas es ampliable. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se combinan mediante una Y lógica los valores de todos los parámetros de entrada disponibles. El resultado se emite en la salida OUT.

El estado lógico del bit de resultado es "1" sólo si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

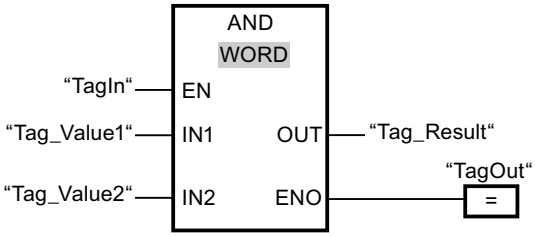
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
INn	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valores de entrada opcionales
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0000 0000 0000 0101

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica Y". El valor del operando "Tag_Value1" se combina mediante una Y lógica con el valor

del operando "Tag_Value2". El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

OR: Operación lógica O

Descripción

La instrucción "Operación lógica O" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina mediante una O lógica con el bit 0 del valor de la entrada IN2. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los bits de la variable indicada.

El cuadro de la instrucción contiene en estado estacionario mín. 2 entradas (IN1 e IN2). El número de entradas del cuadro de la instrucción se puede ampliar. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se combinan mediante una O lógica los valores de todos los parámetros de entrada disponibles. El resultado se emite en la salida OUT.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación

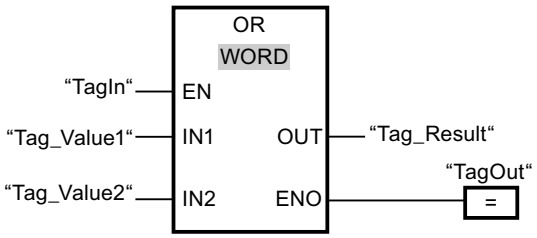
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
IN1	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
INn	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valores de entrada opcionales
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O". El valor del operando "Tag_Value1" se combina mediante una O lógica con el valor del operando "Tag_Value2". El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

XOR: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O-exclusiva lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina mediante una O-exclusiva lógica con el bit 0 del valor de la entrada IN2. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits del valor indicado.

El cuadro de la instrucción contiene en estado estacionario mín. 2 entradas (IN1 e IN2). El número de entradas del cuadro de la instrucción se puede ampliar. Las entradas insertadas se numeran en orden ascendente en el cuadro. Al ejecutar la instrucción se combinan mediante una O-exclusiva lógica los valores de todos los parámetros de entrada disponibles. El resultado se emite en la salida OUT.

El estado lógico del bit de resultado es "1" si el estado lógico de uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "1" ó "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN1	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica

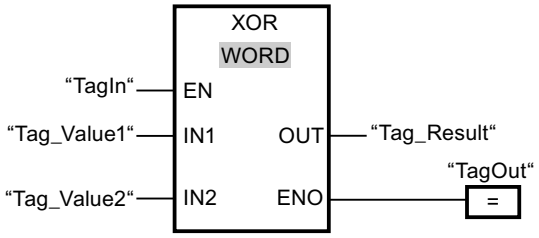
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
INn	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valores de entrada opcionales
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1010

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O-exclusiva". El valor del operando "Tag_Value1" se combina mediante una O-exclusiva lógica con el valor del operando "Tag_Value2". El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Quitar entradas y salidas de una instrucción (Página 2742)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

INV: Complemento a 1

Descripción

La instrucción "Complemento a 1" permite invertir el estado lógico de los bits de la entrada IN. Al ejecutar la instrucción, el valor de la entrada IN se combina con una operación lógica O-exclusiva utilizando una plantilla hexadecimal (W#16#FFFF para números de 16 bits o DW#16#FFFF FFFF para números de 32 bits). Esto invierte el estado lógico de los distintos bits y deposita el resultado en la salida OUT.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Complemento a 1":

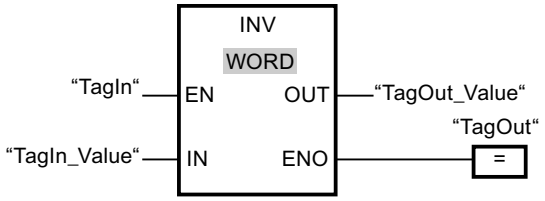
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Secuencias de bits, enteros	I, Q, M, D, L, P	I, Q, M, D, L, P	Complemento a 1 del valor de la entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	W#16#000F	W#16#7E
OUT	TagOut_Value	W#16#FFF0	W#16#81

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Complemento a 1". La instrucción invierte el estado lógico de los distintos bits de la entrada "TagIn_Value" y escribe el resultado en la salida "TagOut_Value". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

DECO: Descodificar

Descripción

La instrucción "Descodificar" permite activar un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor de la entrada IN y activa el bit del valor de salida, cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor de la entrada IN es mayor que 31, se ejecuta una instrucción modulo 32.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

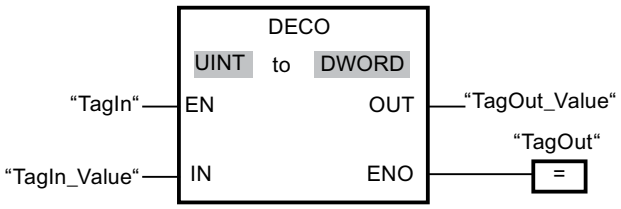
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	UINT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida.
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

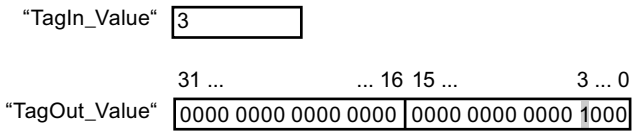
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Descodificar". La instrucción lee el número de bit "3" del valor del operando "TagIn_Value" de la entrada y activa el tercer bit del valor del operando "TagOut_Value" de la salida.

Si no ocurren errores al ejecutar la instrucción, la salida ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

ENCO: Codificar

Descripción

La instrucción "Codificar" lee el número del bit menos significativo en el valor de entrada y lo emite en la salida OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor de la entrada IN y escribe su número de bit en la variable de la salida OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

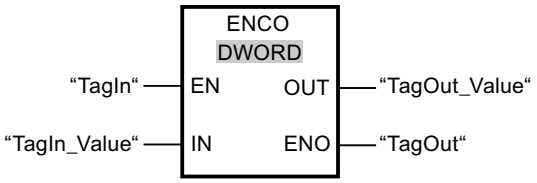
Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
EN	Input	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

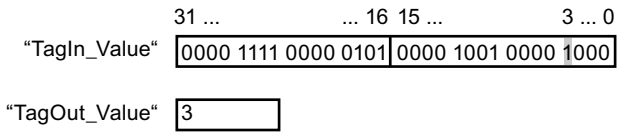
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Codificar". La instrucción selecciona la posición de bit "3" como bit menos significativo del valor de la entrada "TagIn_Value" y escribe el valor "3" en la variable de la salida "TagOut_Value".

Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Editar elementos FUP (Página 2738)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" selecciona, en función de un estado lógico del interruptor (entrada G), una de las entradas IN0 o IN1 y copia su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G tiene el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

La instrucción solo se puede ejecutar si el estado lógico de la entrada de habilitación EN es "1" y las variables de todos los parámetros son del mismo tipo de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

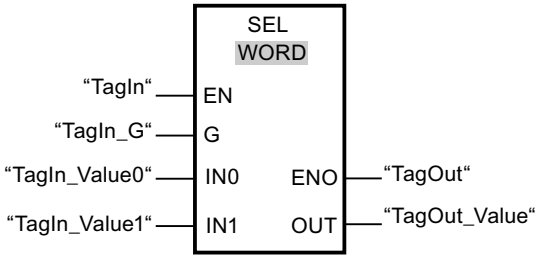
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
G	Input	BOOL	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L, T, C o constante	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, DATE, CHAR	Secuencias de bits, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, LDT, CHAR	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, DATE, CHAR	Secuencias de bits, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, LDT, CHAR	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, DATE, CHAR	Secuencias de bits, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, LDT, CHAR	I, Q, M, D, L, P	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Seleccionar". Dependiendo del estado lógico de la entrada "TagIn_G", se selecciona el valor de la entrada "TagIn_Value0" ó "TagIn_Value1" y se copia en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

MUX: Multiplexar

Descripción

La instrucción "Multiplexar" permite copiar el contenido de una entrada seleccionada en la salida OUT. El cuadro de la instrucción contiene en estado básico mín. 2 entradas (IN0 e IN1). El número de entradas seleccionables en el cuadro de la instrucción se puede ampliar. Se puede declarar un máximo de 32 entradas.

Las entradas se numeran automáticamente en el cuadro. La numeración comienza por IN0 y continúa en orden ascendente con cada nueva entrada. El parámetro K determina la entrada cuyo contenido se copia en la salida OUT. Si el valor del parámetro K es mayor que el número de entradas disponibles, el contenido del parámetro ELSE se copia en la salida OUT y a la salida de habilitación ENO se le asigna el estado lógico "0".

La instrucción "Multiplexar" solo se puede ejecutar si las variables de todas las entradas y de la salida OUT tienen el mismo tipo de datos. El parámetro K es una excepción, ya que para éste sólo pueden indicarse números enteros.

La salida de habilitación ENO se desactiva si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor del parámetro K es mayor que el número de entradas disponibles.
- Se producen errores al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplexar":

Parámetro	Declarar	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
K	Input	Enteros	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Indica la entrada cuyo contenido se copia. • Si K = 0 => parámetro IN0 • Si K = 1 => parámetro IN1, etc.

Parámetro	Declarar	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
IN0	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Primer valor de entrada
IN1	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Segundo valor de entrada
INn	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valores de entrada opcionales

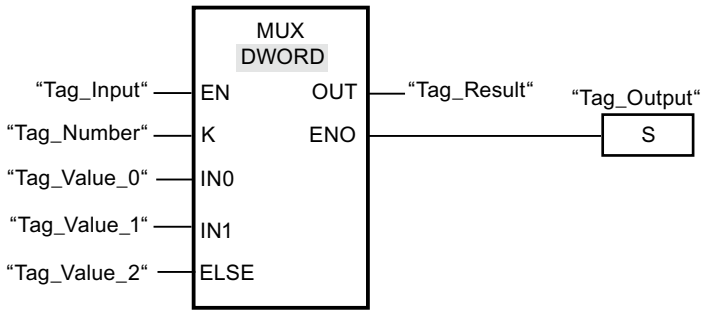
Parámetro	Declarar	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
ELSE	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Indica el valor que se copia si K > n
OUT	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Salida en la que se copia el valor

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
K	Tag_Number	1
IN0	Tag_ValueI_0	DW#16#00000000
IN1	Tag_Value_1	DW#16#003E4A7D
ELSE	Tag_Value_2	DW#16#FFFF0000
OUT	Tag_Result	DW#16#003E4A7D

Si el operando "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplexar". De acuerdo con el valor del operando "Tag_Number", se copia el valor de la entrada "Tag_Value_1" y se asigna al operando de la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activan las salidas ENO y "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

DEMUX: Desmultiplexar

Descripción

La instrucción "Desmultiplexar" permite copiar el contenido de la entrada IN en una salida seleccionada. El cuadro de la instrucción contiene en estado estacionario mín. 2 salidas (OUT0 y OUT1). El número de salidas seleccionables del cuadro de la instrucción se puede ampliar. Las salidas se numeran automáticamente en el cuadro. La numeración comienza por OUT0 y continúa en orden ascendente con cada nueva salida. El parámetro K permite determinar la salida en la que se copia el contenido de la entrada IN. Las demás salidas no cambian. Si el valor del parámetro K es mayor que el número de salidas disponibles, el contenido de la entrada IN se copia en el parámetro ELSE y a la salida de habilitación ENO se le asigna el estado lógico "0".

La instrucción "Desmultiplexar" solo se puede ejecutar si las variables de la entrada IN y las de todas las salidas son del mismo tipo de datos. El parámetro K es una excepción, ya que en el mismo sólo pueden indicarse números enteros.

La salida de habilitación ENO se desactiva si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor del parámetro K es mayor que el número de salidas disponibles.
- Se producen errores al ejecutar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desmultiplexar":

Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
K	Input	Enteros	Enteros	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Indica la salida en la que se copia el valor de entrada (IN). • Si K = 0 => parámetro OUT0 • Si K = 1 => parámetro OUT1, etc.
IN	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	I, Q, M, D, L, P o constante	Valor de entrada
OUT0	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Primera salida

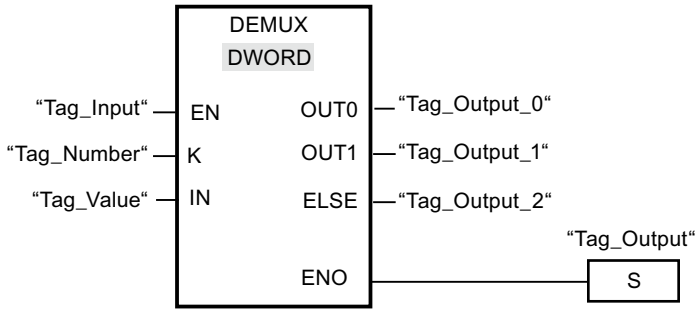
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
OUT1	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Segunda salida
OUTn	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Salidas opcionales
ELSE	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	I, Q, M, D, L, P	Salida en la que se copia el valor de entrada (IN) con K > n.

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos disponibles en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Las tablas siguientes muestran el funcionamiento de la instrucción con valores de operandos concretos:

Tabla 9-26 Valores de entrada de la instrucción "Desmultiplexar" antes de ejecutar el segmento

Parámetro	Operando	Valores	
K	Tag_Number	1	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#003E4A7D

Tabla 9-27 Valores de salida de la instrucción "Desmultiplexar" tras ejecutar el segmento

Parámetro	Operando	Valores	
OUT0	Tag_Output_0	Sin cambios	Sin cambios
OUT1	Tag_Output_1	DW#16#FFFFFFFF	Sin cambios
ELSE	Tag_Output_2	Sin cambios	DW#16#003E4A7D

Cuando la entrada "Tag_Input" devuelve el estado lógico "1", se ejecuta la instrucción "Desmultiplexar". Según el valor del operando "Tag_Number", se copia el valor de la entrada IN en la salida correspondiente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

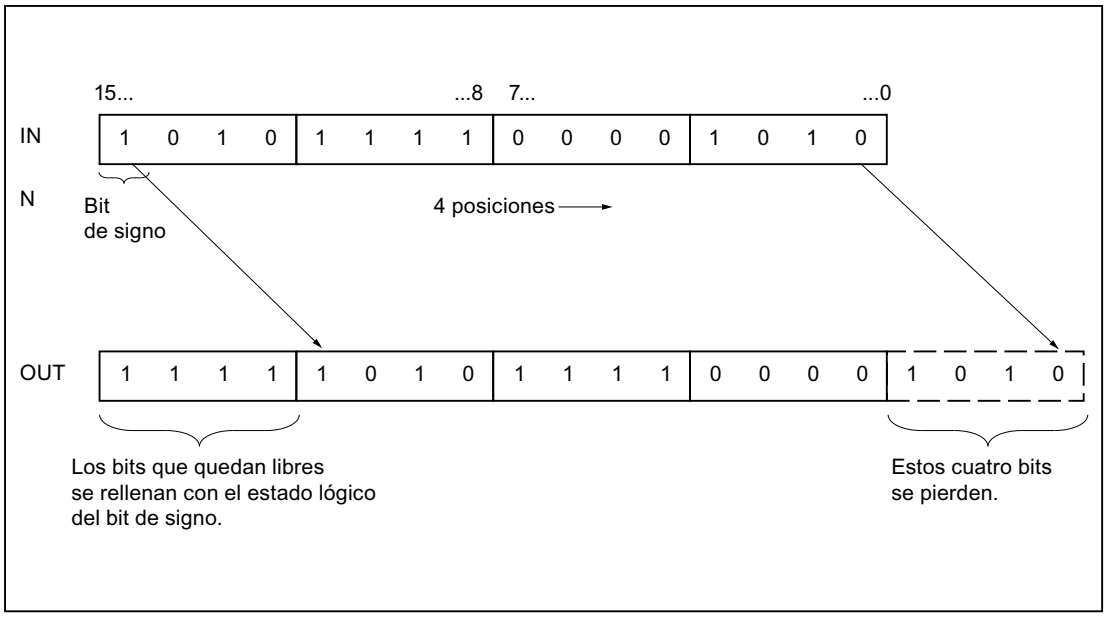
La instrucción "Desplazar a la derecha" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. La entrada N determina el número de bits que debe desplazarse el valor indicado.

Si el valor de la entrada N es "0", el valor de la entrada IN se copia sin modificaciones en el operando de la salida OUT.

Si el valor de la entrada N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la derecha las posiciones de bit disponibles.

En los valores sin signo, se rellenan con ceros los bits que quedan libres en el área izquierda del operando al realizar el desplazamiento. Si el valor indicado lleva signo, las posiciones libres se ocupan con el estado lógico del bit de signo.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos Integer se desplaza cuatro posiciones de bit hacia la derecha:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

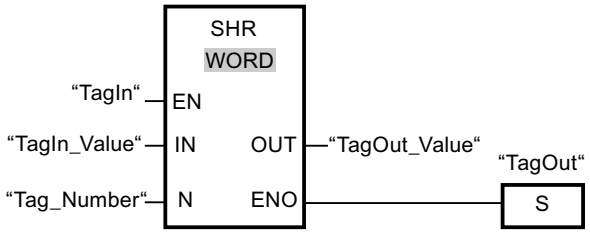
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Valor que se desplaza
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Número de posiciones de bit que se desplaza el valor
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la derecha". El contenido del operando "TagIn_Value" se desplaza tres posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al

ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

SHL: Desplazar a la izquierda

Descripción

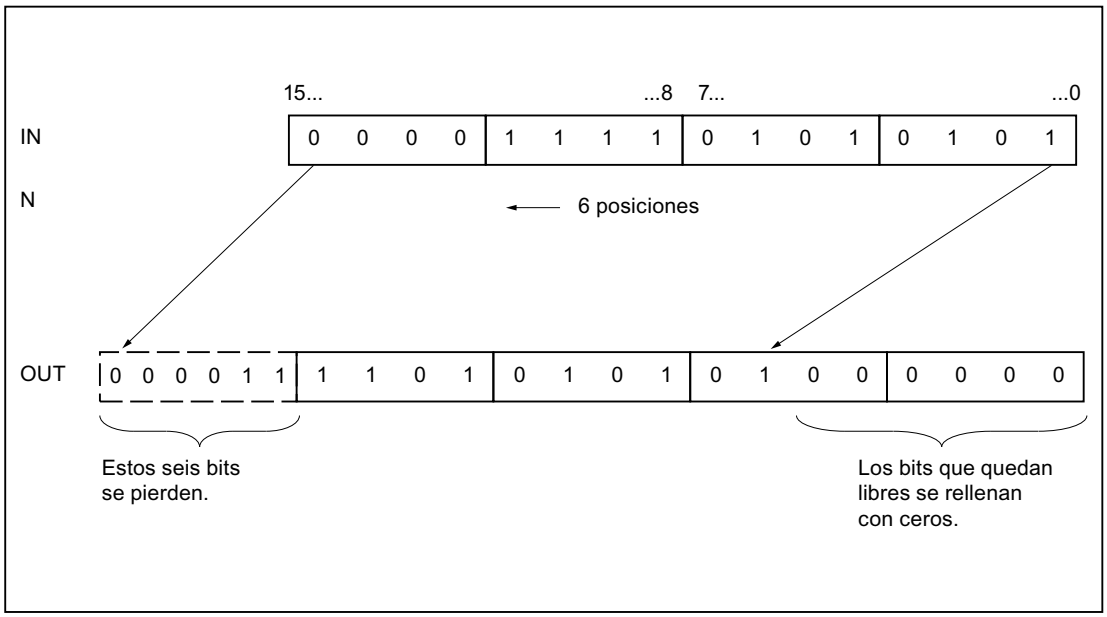
La instrucción "Desplazar a la izquierda" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. La entrada N determina el número de bits que debe desplazarse el valor indicado.

Si el valor de la entrada N es "0", el valor de la entrada IN se copia sin modificaciones en el operando de la salida OUT.

Si el valor de la entrada N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la izquierda las posiciones de bit disponibles.

Los bits que quedan libres en el área derecha del operando al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

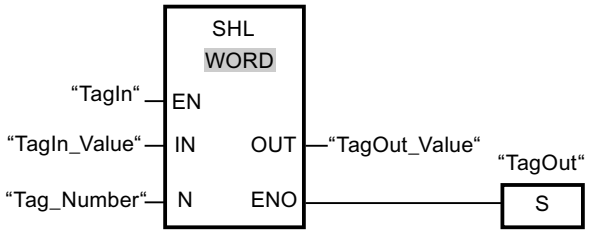
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Valor que se desplaza.
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Número de bits que se desplaza el valor.
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la izquierda". El contenido del operando "TagIn_Value" se desplaza cuatro posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

ROR: Rotar a la derecha

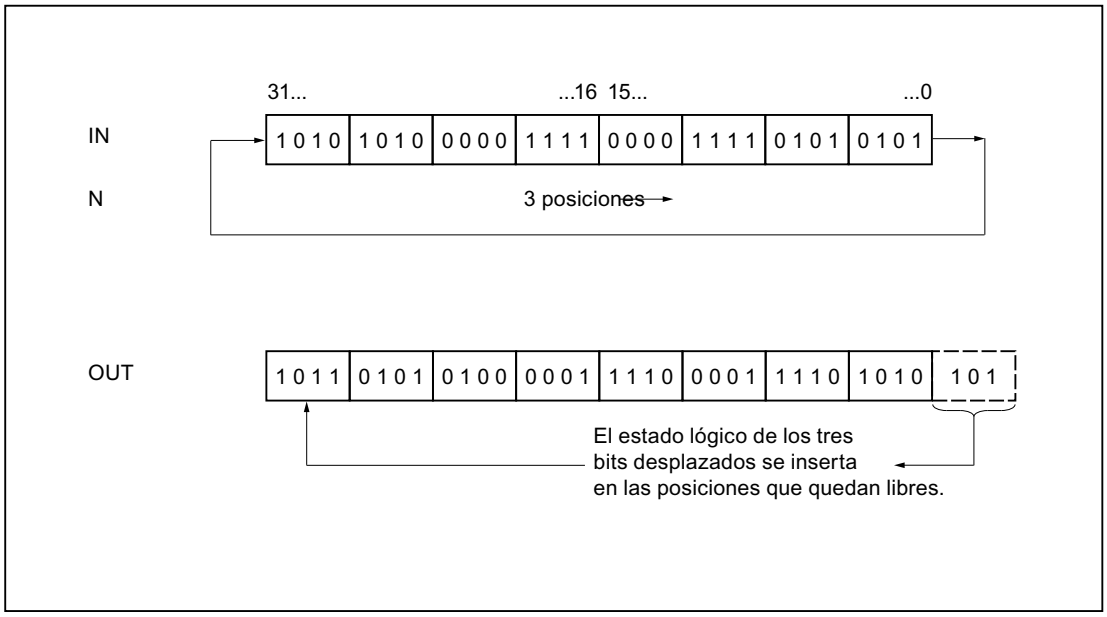
Descripción

La instrucción "Rotar a la derecha" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. La entrada N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación en el lado izquierdo se rellenan con los bits desplazados hacia fuera dispersos del lado derecho.

Si el valor de la entrada N es "0", el valor de la entrada IN se copia sin modificaciones en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

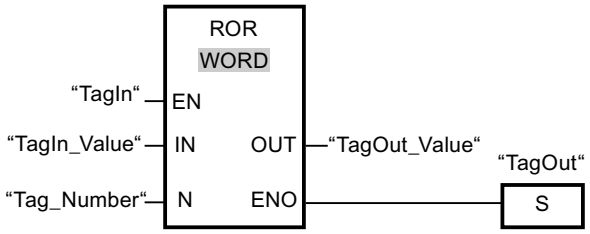
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Valor que se rota
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Número de posiciones de bit que rota el valor
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0000 1111 1001 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1000 0111 1100

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la derecha". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Cablear operandos en instrucciones FUP (Página 2745)

ROL: Rotar a la izquierda

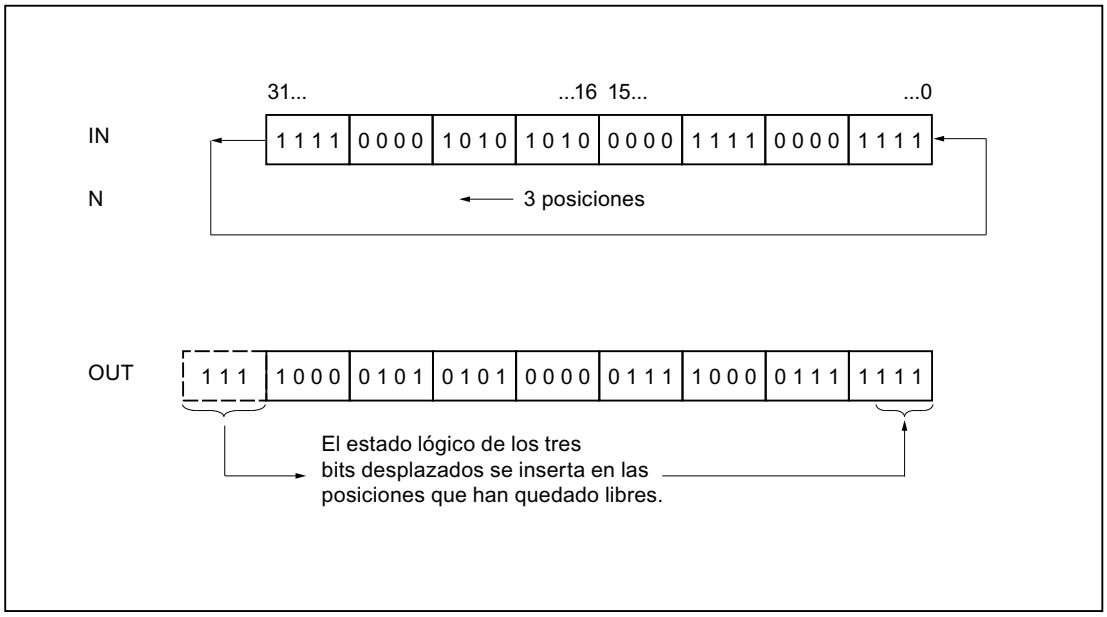
Descripción

La instrucción "Rotar a la izquierda" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. La entrada N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación en el lado derecho se rellenan con los bits desplazados hacia fuera dispersos del lado izquierdo.

Si el valor de la entrada N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

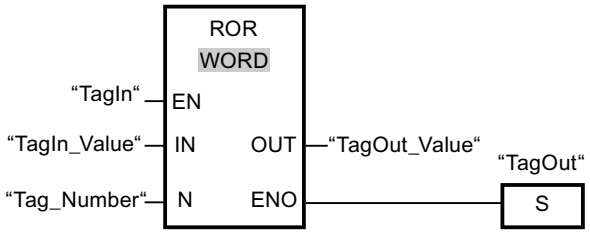
Parámetro	Declaración	Tipo de datos		Área de memoria		Descripción
		S7-1200	S7-1500	S7-1200	S7-1500	
EN	Input	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	BOOL	I, Q, M, D, L	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Valor que se rota.
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Número de bits que rota el valor.
OUT	Output	Secuencias de bits, enteros	Secuencias de bits, enteros	I, Q, M, D, L	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	1010 1000 1111 0110
N	Tag_Number	5
OUT	TagOut_Value	0001 1110 1101 0101

Si la entrada "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la izquierda". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Otras instrucciones

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite ocupar los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el

evento de éste es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L o constante	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L o constante	Bit de evento (i); el estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L o constante	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Paso predeterminado del circuito secuencial sucesivo

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P o constante	Valor de contaje actual del circuito secuencial sucesivo
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	Hora de sistema anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET (solo si ET < PT), OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1"
  - Los parámetros ET, OA y CA se ponen a "0"
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET (solo si ET < PT), OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y C_FB devuelven el estado lógico "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
CMD	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms.
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
CMD_HIS	Static	BOOL	D, L o constante	Bit de evolución de CMD

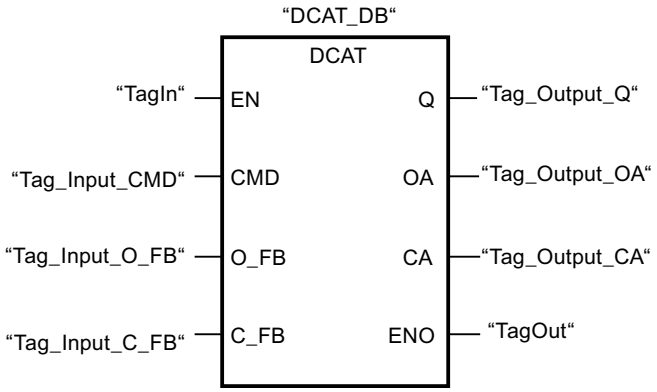
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" permite acumular el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se dispara la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se

depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control del motor" no devuelve información de error.

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado
Leyenda:																
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB															
PT	PT se pone al mismo valor que ET															
X	No aplicable															
< PT	ET < PT															

Parámetros de entrada		Parámetros de salida
>= PT	ET >= PT	
Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE		

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
O_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error.
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
O_HIS	Static	BOOL	D, L o constante	Bit de evolución "Abrir"
C_HIS	Static	BOOL	D, L o constante	Bit de evolución "Cerrar"

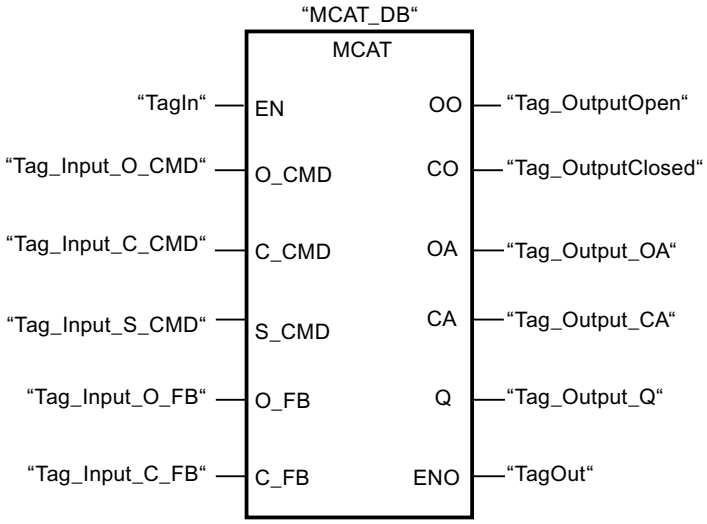
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE

Parámetro	Operando	Valor
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "MCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

Los siguientes valores están almacenados en el bloque de datos de instancia "MCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una coincidencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica sólo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
LAST	Static	BYTE	I, Q, M, D, L, P o constante	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor para la amplificación en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo dinámico. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse para suprimir perturbaciones o como filtro pasa bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

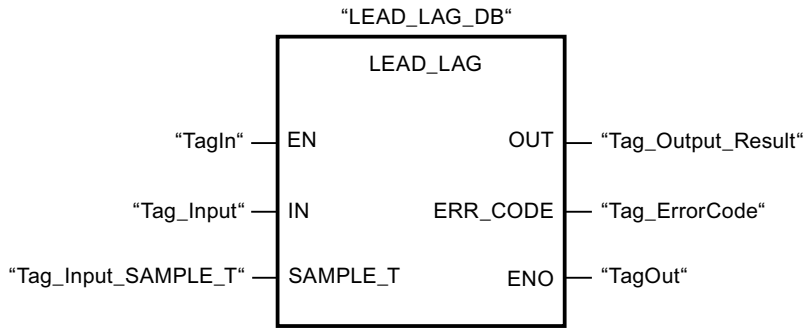
Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Los siguientes valores están almacenados en el bloque de datos de instancia "LEAD_LAG_DB" de la instrucción:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos de instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

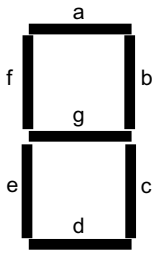
Cambiar formatos de visualización en el estado del programa (Página 3100)

SEG: Generar patrón de bits para el display de 7 segmentos

Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" permite convertir cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se deposita en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)	Display de siete segmentos
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

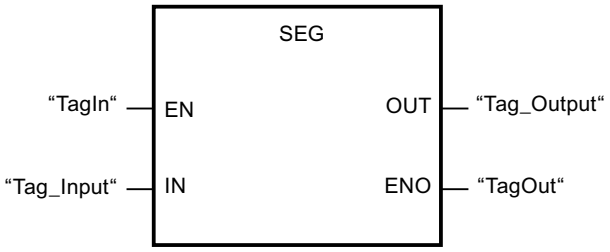
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits para el display de 7 segmentos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	00000110 01011011 01001111 01100110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

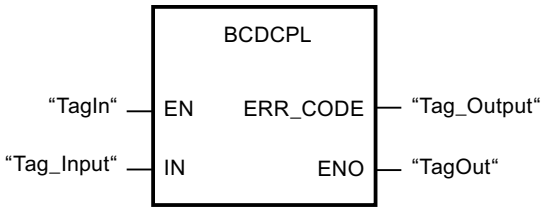
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" permite contar el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

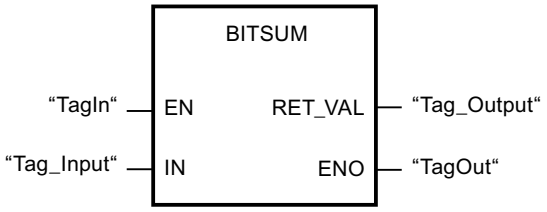
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan.
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

AWL

Operaciones lógicas con bits

R_TRIG: Activar variable con flanco de señal ascendente

Descripción

La instrucción "Activar variable con flanco de señal ascendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "0" a "1". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Activar variable con flanco de señal ascendente" se utiliza la siguiente sintaxis:

```
CALL R_TRIG, "R_TRIG_DB"
CLK := <Operando>
Q := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L o constante	Señal entrante cuyo flanco se consulta.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL R_TRIG, "R_TRIG_DB"     // Se llama la instrucción "Activar variable con
                             flanco de señal ascendente".
CLK := "TagIn"               // Se detecta un flanco de señal ascendente.
Q := "TagOut"                // Estado lógico "1" con flanco de señal ascendente.
```

En la variable del DB de instancia "R_TRIG_DB" se almacena el RLO de la consulta anterior. Si en el operando "TagIn" se detecta un cambio del estado lógico del RLO de "0" a "1", la salida "TagOut" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

F_TRIG: Activar variable con flanco de señal descendente

Descripción

La instrucción "Activar variable con flanco de señal descendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "1" a "0". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Activar variable con flanco de señal descendente" se utiliza la siguiente sintaxis:

```
CALL F_TRIG, "F_TRIG_DB"
CLK := <Operando>
Q := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L o constante	Señal entrante cuyo flanco se consulta.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL F_TRIG, "F_TRIG_DB"	// Se llama la instrucción "Activar variable con flanco de señal descendente".
CLK := "TagIn"	// Se detecta un flanco de señal descendente.
Q := "TagOut"	// Estado lógico "1" con flanco de señal descendente.

En la variable del DB de instancia "R_TRIG_DB" se almacena el RLO de la consulta anterior. Si en el operando "TagIn" se detecta un cambio del estado lógico del RLO de "1" a "0", la salida "TagOut" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Temporizadores

TP: Impulso

Descripción

La instrucción "Impulso" activa la salida Q por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. El parámetro Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico del parámetro Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo PT y si el estado lógico de la entrada IN es "0", se desactiva la salida ET.

En la lógica del programa, la instrucción "Impulso" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME o TP_LTIME en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en

el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo resetea las instancias de la instrucción "Impulso" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Impulso", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Impulso" se utiliza la siguiente sintaxis:

```
CALL TP, "Temporizador
CEI"
<Tipo de datos>
IN := <Operando>
PT := <Operando>
Q := <Operando>
ET := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Impulso":

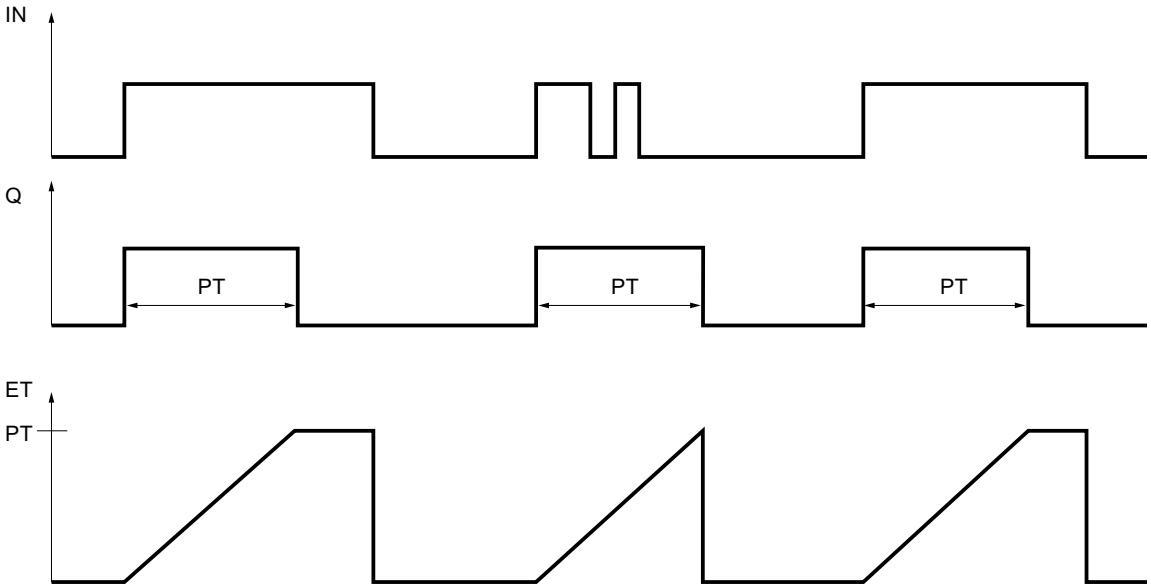
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Duración del impulso. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Salida de impulso
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Valor de tiempo actual

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Impulso" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TP, "TP_DB"	// Se llama la instrucción "Impulso". El bloque de datos "TP_DB" se asigna a la instrucción.
TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// Duración del impulso
Q := "Tag_Status"	// El operando se activa durante el periodo determinado por el operando "Tag_PresetTIME".
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TON: Retardo al conectar (CEI)

Descripción

La instrucción "Retardo al conectar" retarda la activación de la salida Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. Una vez transcurrido el tiempo PT, el parámetro Q devuelve el estado lógico "1". El parámetro Q permanecerá activado mientras la entrada de arranque IN esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva el parámetro Q. La función de temporización se reinicia al detectarse un nuevo flanco de señal ascendente en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. El parámetro ET se inicializa en cuanto el estado lógico de la entrada IN cambia a "0".

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

En la lógica del programa, la instrucción "Retardo al conectar" se debe llamar con la instrucción "Llamar bloque" (CALL).

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME o TON_LTIME en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo resetea las instancias de la instrucción "Retardo al conectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al conectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Retardo al conectar" se utiliza la siguiente sintaxis:

```
CALL TON, "Temporizador
CEI"
<Tipo de datos>
IN := <Operando>
PT := <Operando>
Q := <Operando>
ET := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

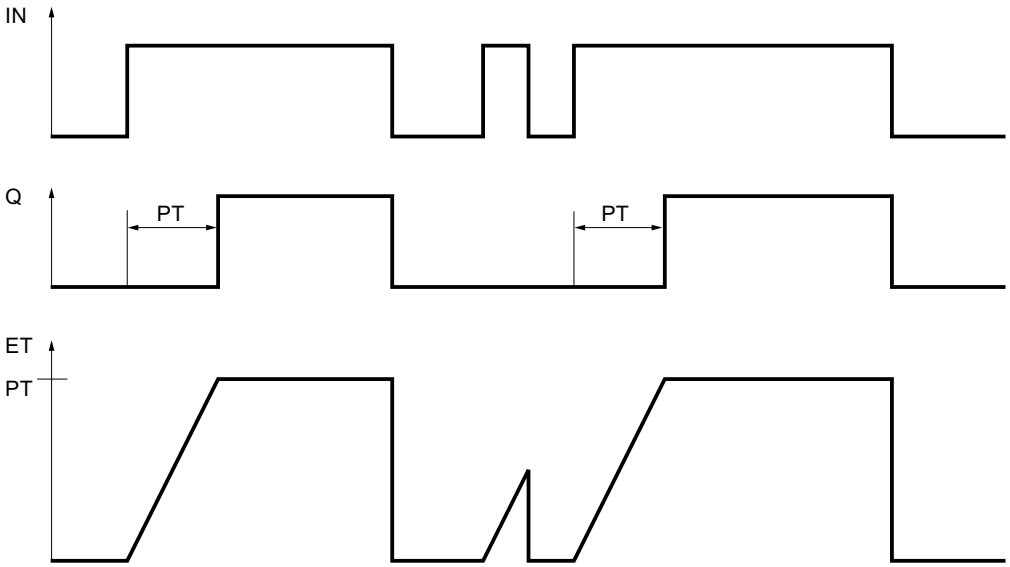
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Estado lógico que se retarda por el tiempo PT.
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Valor de tiempo actual

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Retardo al conectar" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL "TON", "TON_DB"	// Se llama la instrucción "Retardo al conectar". El bloque de datos "TON_DB" se asigna a la instrucción.
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// Indica el tiempo en el que se retarda el flanco de señal ascendente en el parámetro IN.
Q := "Tag_Status"	// El operando se activa una vez transcurrido el tiempo PT predeterminado por la variable "Tag_PresetTIME".
	// El parámetro Q permanecerá activado mientras la variable "Tag_Start" siga a "1".
	// Cuando el estado lógico de la entrada de arranque cambie de "1" a "0", se desactivará el operando en el parámetro Q .
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TOF: Retardo al desconectar

Descripción

La instrucción "Retardo al desconectar" retarda la desactivación de la salida Q por el tiempo programado PT. El parámetro Q se activa cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico del parámetro IN cambia nuevamente a "0", el tiempo programado PT deja de contar. El parámetro Q permanecerá activado mientras transcurra el tiempo PT. Una vez transcurrido el tiempo PT se inicializa el parámetro Q. Si el estado lógico del parámetro IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico del parámetro Q permanece puesto a "1".

El valor de tiempo actual se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el parámetro ET permanecerá al valor actual hasta que el parámetro IN vuelva a cambiar a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

En la lógica del programa, la instrucción "Retardo al desconectar" se debe llamar con la instrucción "Llamar bloque" (CALL).

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME o TOF_LTIME en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo inicializa las instancias de la instrucción "Retardo al desconectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al desconectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Retardo al desconectar" se utiliza la siguiente sintaxis:

```
CALL TOF, "Temporizador
CEI"
<Tipo de datos>
IN := <Operando>
PT := <Operando>
Q := <Operando>
ET := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

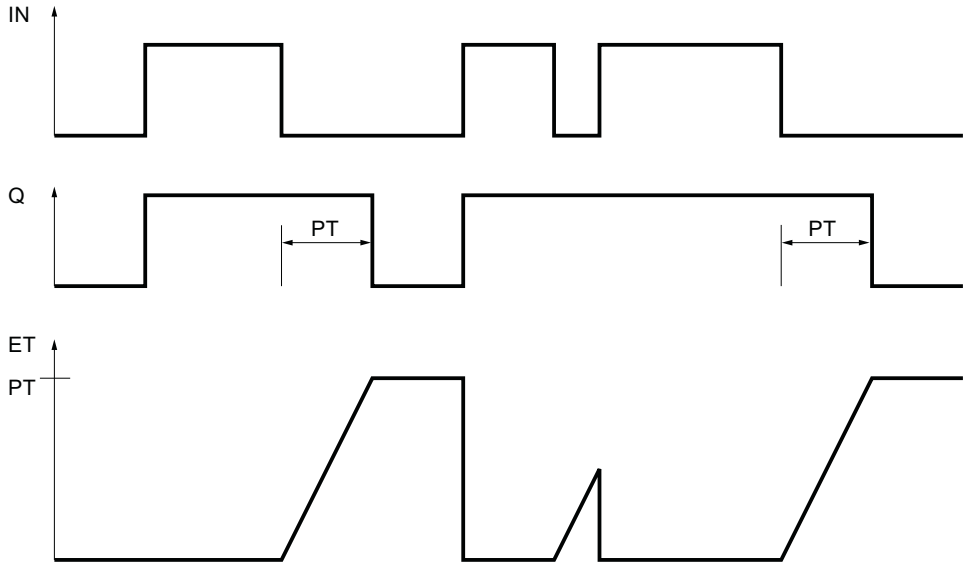
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Estado lógico que se retarda por el tiempo PT.
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Valor de tiempo actual

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Retardo al desconectar" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TOF, "TOF_DB"	// Se llama la instrucción "Retardo al desconectar". El bloque de datos "TOF_DB" se asigna a la instrucción.
TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// Indica el tiempo en el que se retarda el flanco de señal descendente en el parámetro IN.
Q := "Tag_Status"	// El operando se activa cuando la instrucción se inicia debido a un flanco de señal ascendente en el parámetro IN.
	// Si el estado lógico del parámetro IN cambia de "1" a "0", el operando "Tag_Status" permanecerá activado mientras transcurra el tiempo predeterminado por la variable "Tag_PresetTIME".
	// Una vez transcurrido el tiempo del parámetro PT, se desactiva el operando.
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TONR: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PT. Cuando el estado lógico de la entrada IN cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y se empieza a contar el tiempo PT. Mientras transcurre el tiempo PT se van acumulando los valores de tiempo que se leen cuando el estado lógico de la entrada IN es "1". El tiempo acumulado se deposita en la salida ET y se puede consultar allí. Una vez se ha alcanzado el tiempo PT, la salida Q devuelve el estado lógico "1". El parámetro Q permanece a "1" aunque el estado lógico del parámetro IN cambie de "1" a "0" (flanco de señal descendente).

La entrada R desactiva las salidas ET y Q independientemente del estado lógico de la entrada de arranque.

En la lógica del programa, la instrucción "Acumulador de tiempo" se debe llamar con la instrucción "Llamar bloque" (CALL).

A cada llamada de la instrucción "Acumulador de tiempo" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME o TONR_LTIME en la sección "Static" de un bloque (p. ej., #MyTONR_TIMER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Acumulador de tiempo" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Acumulador de tiempo" se utiliza la siguiente sintaxis:

```
CALL TONR, "Temporizador
CEI"
<Tipo de datos>
IN := <Operando>
R := <Operando>
PT := <Operando>
Q := <Operando>
ET := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

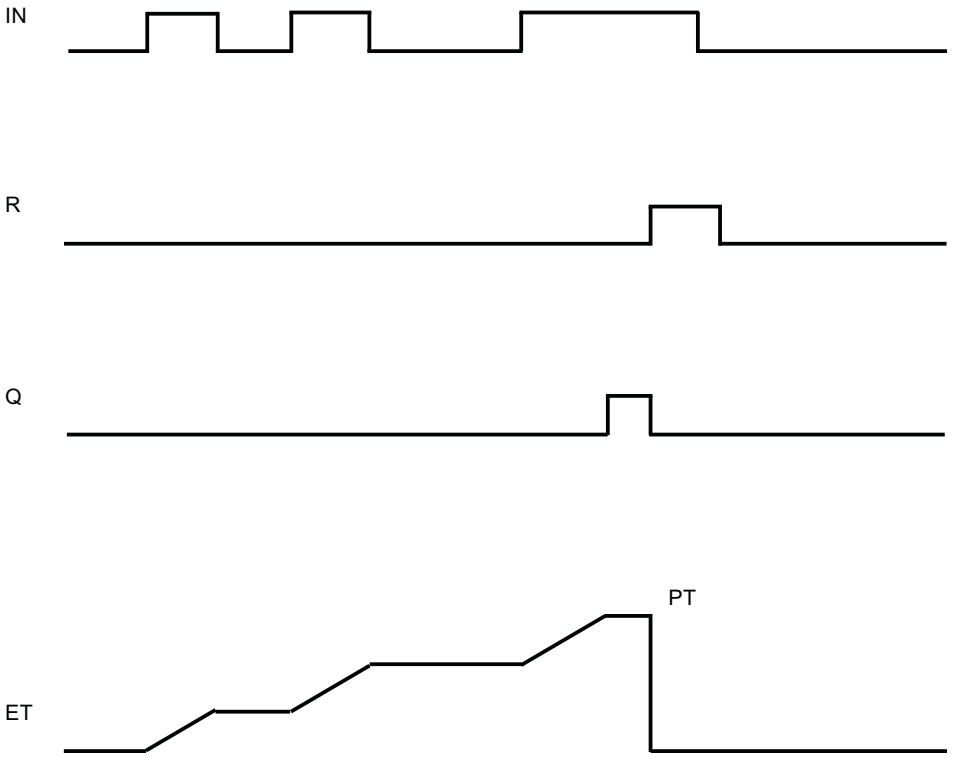
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
R	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de reset
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Tiempo acumulado

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Acumulador de tiempo":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TONR, "TONR_DB"	// Se llama la instrucción "Acumulador de tiempo". El bloque de datos "TONR_DB" se asigna a la instrucción.
TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción y se inicia el tiempo de la entrada PT. Mientras transcurre el tiempo PT se van acumulando los valores de tiempo que se leen cuando el estado lógico de la entrada IN es "1".
R := "Tag_Reset"	// La entrada R desactiva las salidas ET y Q independientemente del estado lógico de la entrada de arranque.
PT := "Tag_PresetTIME"	// Indica el tiempo durante el cual se acumularán los valores de tiempo.
Q := "Tag_Status"	// El operando se activa cuando se alcanza el tiempo PT.

AWL	Explicación
	// El parámetro Q permanece a "1" aunque el estado lógico del parámetro IN cambie de "1" a "0" (flanco de señal descendente).
ET := "Tag_ElapsedTIME"	// El tiempo acumulado se devuelve en la salida ET y se puede consultar allí.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RESET_TIMER: Inicializar temporizador

Descripción

La instrucción "Inicializar temporizador" permite poner a "0" un temporizador CEI. Los componentes de estructura del temporizador en el bloque de datos indicado se ponen a "0".

La instrucción no afecta al RLO. En el parámetro TIMER se asigna a la instrucción "Inicializar temporizador" un temporizador CEI declarado en el programa.

Los datos de la instrucción se actualizan únicamente al efectuar una llamada de la instrucción, y no cada vez que se accede al temporizador CEI asignado. La consulta de datos es igual a la llamada de instrucción únicamente hasta la siguiente llamada de instrucción.

Sintaxis

Para la instrucción "Inicializar temporizador" se utiliza la siguiente sintaxis:

```
CALL RESET_TIMER
<Tipo de datos>
TIMER := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar temporizador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TIMER	Output	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI que se inicializa

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TON, "TON_DB"	// Se llama la instrucción "Retardo al conectar". El bloque de datos instancia "TON_DB" se asigna a la instrucción.
time_type := TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando "Tag_Start", se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// El temporizador CEI depositado en el bloque de datos instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PresetTIME".
Q := "Tag_Status"	// El operando "Tag_Status" se activa una vez transcurrido el tiempo PT predeterminado por el operando "Tag_PresetTIME".
	// El parámetro Q permanecerá activado mientras el operando "Tag_Start" tenga el estado lógico "1".
	// Cuando el estado lógico de la entrada de arranque cambie de "1" a "0", se desactivará el operando en el parámetro Q.
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual
A "Tag_Input_1"	// Cuando el operando "Tag_Input_1" y
A "Tag_Input_2"	// el operando "Tag_Input_2" devuelven el estado lógico "1",
CALL RESET_TIMER	// se llama la instrucción "Inicializar temporizador".
timer_type := IEC_TIMER	// Tipo de datos del temporizador CEI
TIMER := "TON_DB"	// El temporizador CEI se inicializa

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

PRESET_TIMER: Cargar tiempo

Descripción

La instrucción "Cargar tiempo" permite ajustar el tiempo de un temporizador CEI. La instrucción se ejecuta en cada ciclo. La instrucción escribe el tiempo indicado en la estructura del temporizador CEI indicado.

A la instrucción "Cargar tiempo" debe asignársele un temporizador CEI declarado en el programa.

La actualización de los datos de la instrucción se lleva a cabo cuando se llama la instrucción y cada vez que se accede al temporizador CEI asignado. La consulta de Q o ET (p. ej., "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Nota

Si el temporizador CEI indicado se ejecuta mientras se ejecuta la instrucción, esta sobrescribe el tiempo actual del temporizador CEI indicado. Esto puede modificar el estado del temporizador CEI.

Sintaxis

Para la instrucción "Cargar tiempo" se utiliza la siguiente sintaxis:

```
CALL PRESET_TIMER
<Tipo de datos> <Tipo de
datos>
PT := <Operando>
TIMER := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar tiempo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	TIME, LTIME	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador CEI
<Temporizador CEI>	Output	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	D, L	Temporizador CEI cuyo tiempo se ajusta

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TON, "TON_DB"	// Se llama la instrucción "Retardo al conectar". El bloque de datos instancia "TON_DB" se asigna a la instrucción.
time_type := TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando "Tag_Start", se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// El temporizador CEI depositado en el bloque de datos instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PresetTIME".
Q := "Tag_Status"	// El operando "Tag_Status" se activa una vez transcurrido el tiempo PT predeterminado por el operando "Tag_PresetTIME".
	// El parámetro Q permanecerá activado mientras el operando "Tag_Start" tenga el estado lógico "1".
	// Cuando el estado lógico de la entrada de arranque cambie de "1" a "0", se desactivará el operando en el parámetro Q.
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual
A "Tag_Input_1"	// Cuando el operando "Tag_Input_1" y
A "Tag_Input_2"	// el operando "Tag_Input_2" devuelven el estado lógico "1",

AWL	Explicación
CALL PRESET_TIMER	// se llama la instrucción "Cargar tiempo".
time_type := TIME	// Tipo de datos de la instrucción
timer_type := IEC_TIMER	// Tipo de datos del temporizador CEI
PT := "Tag_PresetTIME_new"	// La instrucción escribe el tiempo "Tag_PresetTIME_new" en el bloque de datos instancia "TON_DB" y sobrescribe a la vez el valor de tiempo del operando "Tag_PresetTIME" en el bloque de datos instancia. Esto puede modificar el estado lógico del estado del temporizador en la siguiente consulta o al acceder a "MyTimer".Q o "MyTimer".ET.
TIMER := "TON_DB"	// El temporizador CEI se inicializa

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Contadores

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor de contaje actual del parámetro CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Cuando se alcanza el valor límite superior, el estado lógico del parámetro CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. El estado lógico del parámetro Q es determinado por el parámetro PV. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0".

El valor del parámetro CV se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico del parámetro CU no tendrá efecto alguno en la instrucción.

En la lógica del programa, la instrucción "Contador ascendente" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Variable local

- CTU_SINT / CTU_USINT
- CTU_INT / CTU_UINT
- CTU_DINT / CTU_UDINT
- CTU_LINT / CTU_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTU_<tipo de datos> o IEC_<contador> en la sección "Static" de un bloque (p. ej. #MyCTU_COUNTER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se declaran remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se declara remanente en la interfaz del bloque.

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro R de la instrucción. Si un bloque diferente contiene instancias de la instrucción "Contador ascendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

```
CALL CTU, "Contador CEI"
<Tipo de datos>
CU := <Operando>
R := <Operando>
PV := <Operando>
Q := <Operando>
CV := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de reset
PV	Input	Enteros	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL CTU, "CTU_DB"	// Se llama la instrucción "Contador ascendente". El bloque de datos "CTU_DB" se asigna a la instrucción.
INT	// Tipo de datos de la instrucción
CU := "Tag_StartCTU"	// Cuando el estado lógico del operando "Tag_StartCTU" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se incrementa en uno.
	// El valor de contaje sigue incrementando hasta alcanzar el valor límite superior de INT = 32767.
R := "Tag_ResetCounter"	// Cuando el estado lógico del operando "Tag_ResetCounter" cambia a "1", el operando "Tag_CounterValue" se pone a "0".
PV := "Tag_PresetValue"	// El operando determina a partir de qué valor debe activarse el operando del parámetro Q.
Q := "Tag_CounterStatus"	// El operando permanece activado mientras el valor de contaje actual sea mayor o igual al valor del parámetro PV.
CV := "Tag_CounterValue"	// Valor de contaje actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos indicado. Cuando se

alcanza el valor límite inferior, el estado lógico del parámetro CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. Si el valor de contaje actual es menor o igual a cero, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0".

El valor del parámetro CV se pone al valor del parámetro PV cuando el estado lógico del parámetro LD cambia a "1". Mientras el parámetro LD tenga el estado lógico "1", el estado lógico del parámetro CD no tendrá efecto alguno en la instrucción.

En la lógica del programa, la instrucción "Contador descendente" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Variable local

- CTD_SINT / CTD_USINT
- CTD_INT / CTD_UINT
- CTD_DINT / CTD_UDINT
- CTD_LINT / CTD_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTD_<tipo de datos> o IEC_<contador> en la sección "Static" de un bloque (p. ej. #MyCTD_COUNTER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se declaran remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se declara remanente en la interfaz del bloque.

El sistema operativo resetea las instancias de la instrucción "Contador descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro LD de la instrucción. En el parámetro PV se indica en este caso el valor inicial deseado para el parámetro CV. Si un bloque diferente contiene instancias de la instrucción "Contador descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

```
CALL CTD, "Contador CEI"
<Tipo de datos>
CD := <Operando>
LD := <Operando>
PV := <Operando>
Q := <Operando>
CV := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CD	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL CTD, "CTD_DB"	// Se llama la instrucción "Contador descendente". El bloque de datos "CTD_DB" se asigna a la instrucción.
INT	// Tipo de datos de la instrucción
CD := "Tag_StartCTD"	// Cuando el estado lógico del operando "Tag_StartCTD" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se decrementa en uno.
	// El valor de contaje del parámetro CV se sigue decrementando hasta alcanzar el valor límite inferior de INT = -32768.
LD := "Tag_LoadPV"	// Cuando el estado lógico del operando "Tag_LoadPV" cambia a "1", el operando "Tag_CounterValue" se pone al valor del operando "Tag_PresetValue".
PV := "Tag_PresetValue"	// Indica el valor al que se pone el contador cuando el estado lógico del parámetro LD es "1".
Q := "Tag_CounterStatus"	// El operando se activa si el valor de contaje actual es menor o igual a cero.
CV := "Tag_CounterValue"	// Valor de contaje actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno y se deposita en el parámetro CV. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje del parámetro CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje actual no cambia en el parámetro CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado en el parámetro CV. Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado, ya no se decrementa el valor de contaje.

Cuando el estado lógico del parámetro LD cambia a "1", el valor de contaje del parámetro CV se pone al valor del parámetro PV. Mientras el parámetro LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetro R sea "1", un cambio del estado lógico de los parámetros CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en el parámetro QU. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro QU adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QU es "0". En el parámetro PV también se puede indicar una constante.

El estado del contador descendente se puede consultar en el parámetro QD. Si el valor de contaje actual es menor o igual a cero, el parámetro QD adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QD es "0".

En la lógica del programa, la instrucción "Contador ascendente - descendente" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER

- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Variable local

- CTUD_SINT / CTUD_USINT
- CTUD_INT / CTUD_UINT
- CTUD_DINT / CTUD_UDINT
- CTUD_LINT / CTUD_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTUD_<tipo de datos> o IEC_<contador> en la sección "Static" de un bloque (p. ej. #MyCTUD_COUNTER)

Tras seleccionar el tipo de datos en la lista desplegable "???", el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol de proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se declaran remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se declara remanente en la interfaz del bloque.

El sistema operativo resetea las instancias de la instrucción "Contador ascendente - descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con los siguientes valores de parámetros:

- Si se utiliza como contador ascendente, el valor del parámetro R debe ponerse a "1".
- Si se utiliza como contador descendente, el valor del parámetro LD debe ponerse a "1". En el parámetro PV debe indicarse en este caso el valor inicial deseado del parámetro CV.

Si un bloque diferente contiene instancias de la instrucción "Contador ascendente - descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador ascendente - descendente" se utiliza la siguiente sintaxis:

```
CALL CTUD, "Contador CEI"
<Tipo de datos>
CU := <Operando>
CD := <Operando>
R := <Operando>
LD := <Operando>
PV := <Operando>
QU := <Operando>
QD := <Operando>
CV := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU.
QU	Output	BOOL	I, Q, M, D, L	Estado del contador ascendente
QD	Output	BOOL	I, Q, M, D, L	Estado del contador descendente
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	Valor de contaje actual

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL CTUD, "CTUD_DB"	// Se llama la instrucción "Contador ascendente - descendente". El bloque de datos "CTUD_DB" se asigna a la instrucción.
INT	// Tipo de datos de la instrucción
CU := "Tag_StartCTU"	// Cuando el estado lógico del operando "Tag_StartCTU" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se incrementa en uno. // Si se detecta un flanco de señal ascendente en el parámetro CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior de INT = 32767.
CD := "Tag_StartCTD"	// Cuando el estado lógico del operando "Tag_StartCTD" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se decrementa en uno. // El valor de contaje del parámetro CV se sigue decrementando hasta alcanzar el valor límite inferior de -32768.
R := "Tag_ResetCounter"	// Cuando el estado lógico del operando "Tag_ResetCounter" cambia a "1", el operando "Tag_CounterValue" se pone a "0".
LD := "Tag_LoadPV"	// Cuando el estado lógico del operando "Tag_LoadPV" cambia a "1", el operando "Tag_CounterValue" se pone al valor del operando "Tag_PresetValue".
PV := "Tag_PresetValue"	// Indica el valor al que se pone el contador cuando el estado lógico del parámetro LD es "1".
QU := "Tag_CounterStatus"	// El operando permanece activado mientras el valor de contaje actual sea mayor o igual al valor del parámetro PV.
QD := "Tag_CounterStatus"	// El operando se activa si el valor de contaje actual es menor o igual a cero.
CV := "Tag_CounterValue"	// Valor de contaje actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Funciones matemáticas

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor menor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor del parámetro OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no tienen el mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Determinar mínimo" se utiliza la siguiente sintaxis:

```
CALL MIN
<Tipo de datos>
IN1 := <Operando>
IN2 := <Operando>
IN3 := <Operando>
OUT := <Operando>
```

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN3	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Tercer valor de entrada
OUT	Output	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL MIN                     // Se llama la instrucción "Determinar mínimo".
INT                          // Tipo de datos de la instrucción
IN1 := "TagIn_Value1"        // Primer valor de entrada que se compara.
IN2 := "TagIn_Value2"        // Segundo valor de entrada que se compara.
IN3 := "TagIn_Value3"        // Tercer valor de entrada que se compara.
OUT := "Tag_Minimum"         // Valor mínimo de la comparación
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Minimum	12222

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor mayor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Determinar máximo" se utiliza la siguiente sintaxis:

```
CALL MAX
<Tipo de datos>
IN1 := <Operando>
IN2 := <Operando>
IN3 := <Operando>
OUT := <Operando>
```

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
IN3	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Tercer valor de entrada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL MAX                     // Se llama la instrucción "Determinar máximo".
INT                          // Tipo de datos de la instrucción
IN1 := "TagIn_Value1"        // Primer valor de entrada que se compara.
IN2 := "TagIn_Value2"        // Segundo valor de entrada que se compara.
IN3 := "TagIn_Value3"        // Tercer valor de entrada que se compara.
OUT := "Tag_Maximum"         // Valor máximo de la comparación
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Maximum	14444

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor de la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Para poder ejecutar la instrucción, todas las variables de las entradas deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor del parámetro MN es mayor que el valor del parámetro MX.

Sintaxis

Para la instrucción "Ajustar valor límite" se utiliza la siguiente sintaxis:

```
CALL LIMIT
<Tipo de datos>
MN := <Operando>
IN := <Operando>
MX := <Operando>
OUT := <Operando>
```

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MN	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Límite inferior
IN	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Valor de entrada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MX	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Límite superior
OUT	Output	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL LIMIT                   // Se llama la instrucción "Ajustar valor límite".
INT                          // Tipo de datos de la instrucción
MN := "Tag_LowLimit"         // Límite inferior
IN := "Tag_InputValue"       // Valor de entrada
MX := "Tag_HighLimit"        // Límite superior
OUT := "Tag_Result"          // Resultado
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

En el operando "Tag_Result" se escribe el valor del operando "Tag_LowLimit", ya que el valor del operando "Tag_InputValue" se encuentra fuera del área límite definida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Transferencia

MOVE: Copiar valor

Descripción

La instrucción "Copiar valor" transfiere el contenido del operando de la entrada IN al operando de la salida OUT. El tipo de datos de los operandos en la entrada IN y en la salida OUT debe ser idéntico.

La tabla siguiente muestra las posibles transferencias:

Origen (IN)	Destino (OUT)
BYTE	BYTE
WORD	WORD
DWORD	DWORD
LWORD	LWORD
SINT	SINT
USINT	USINT
INT	INT
UINT	UINT
DINT	DINT
UDINT	UDINT
LINT	LINT
ULINT	ULINT
REAL	REAL
LREAL	LREAL
S5TIME	S5TIME
TIME	TIME
LTIME	LTIME
DATE	DATE
DT	DT
LDT	LDT
TOD	TOD
LTOD	LTOD
DTL	DTL
CHAR	CHAR
WCHAR	WCHAR
Caracteres de una cadena[1)]	Caracteres de una cadena
ARRAY	ARRAY
STRUCT	STRUCT

Origen (IN)	Destino (OUT)
Tipo de datos PLC (UDT)	Tipo de datos PLC (UDT)
IEC_TIMER	IEC_TIMER
IEC_LTIMER	IEC_LTIMER
IEC_SCOUNTER	IEC_SCOUNTER
IEC_USCOUNTER	IEC_USCOUNTER
IEC_COUNTER	IEC_COUNTER
IEC_UCOUNTER	IEC_UCOUNTER
IEC_DCOUNTER	IEC_DCOUNTER
IEC_UDCOUNTER	IEC_UDCOUNTER
IEC_LCOUNTER	IEC_LCOUNTER
IEC_ULCOUNTER	IEC_ULCOUNTER
DB_ANY	DB_ANY

[1)] La instrucción "Copiar valor" también permite transferir caracteres individuales de una cadena a operandos del tipo de datos CHAR o WCHAR. El número del carácter que se va a transferir se indica entre corchetes al lado del nombre del operando. Si se indica "MyString[2]", se transfiere p. ej. el segundo carácter de la cadena de caracteres "MyString". Asimismo, es posible transferir operandos del tipo de datos CHAR a caracteres individuales de una cadena de caracteres. También es posible reemplazar un carácter específico de una cadena de caracteres por el carácter de otra cadena de caracteres.

Para copiar operandos del tipo de datos ARRAY pueden utilizarse también las instrucciones "Copiar área" (MOVE_BLK) y "Copiar área sin interrupciones" (UMOVE_BLK) y "Copiar área" (MOVE_BLK_VARIANT). Los operandos del tipo de datos STRING se pueden copiar con la instrucción "Desplazar cadena de caracteres" (S_MOVE).

Sintaxis

Para la instrucción "Copiar valor" se utiliza la siguiente sintaxis:

```
CALL MOVE
VARIANT
IN := <Operando>
OUT := <Operando>
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Copiar valor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres, ARRAY, tipos de datos CEI, tipos de datos PLC (UDT), DB_ANY	I, Q, M, D, L o constante	Elemento con el que se sobrescribe la dirección de destino
OUT	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres, ARRAY, tipos de datos CEI, tipo de datos PLC (UDT), DB_ANY	I, Q, M, D, L	Dirección de destino

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL MOVE                    // Se llama la instrucción "Copiar valor".
VARIANT                      // Tipo de datos de la instrucción
IN := "TagIn_LREAL"          // Se copia el contenido del operando "TagIn_LREAL".
OUT := "TagOut_LREAL"        // Y se transfiere al operando "TagOut_LREAL".
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_LREAL	1,23456789098e55
OUT	TagOut_LREAL	1,23456789098e55

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar valor". La instrucción copia los contenidos del operando "TagIn_LREAL" en el operando "TagOut_LREAL".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Deserialize: Deserializar

Descripción

Con la instrucción "Deserializar", la forma de representación secuencial de un tipo de datos PLC (UDT) se vuelve a convertir en un tipo de datos PLC y se rellena todo el contenido.

El área de memoria en la que se encuentra la forma de representación secuencial de un tipo de datos PLC debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

Esta instrucción permite reconvertir paso a paso a su estado original varias formas de representación secuenciales de tipos de datos PLC convertidos.

Nota

Para reconvertir una sola forma de representación secuencial de un tipo de datos PLC (UDT) también se puede utilizar directamente la instrucción "TRCV: Recibir datos a través de la conexión".

Sintaxis

Para la instrucción "Deserializar" se utiliza la siguiente sintaxis:

```
CALL Deserialize
SRC_ARRAY := <Operando>
RET_VAL := <Operando>
DEST_VARIABLE :=
<Operando>
POS := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Deserializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC_ARRAY	Input	VARIANT	I, Q, D, L	Bloque de datos global en el que se guarda el flujo de datos creado
DEST_VARIABLE	InOut	VARIANT	I, Q, D, L	Variable en la que se guarda el tipo de datos PLC (UDT) reconvertido
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_ARRAY y DEST_VARIABLE se solapan.
8136	El bloque de datos del parámetro DEST_VARIABLE no es un bloque con acceso estándar.
8150	El tipo de datos VARIANT del parámetro SRC_ARRAY no contiene ningún valor.
8151	Error de creación de código en el parámetro SRC_ARRAY
8153	En el parámetro SRC_ARRAY no hay suficiente memoria disponible.
8250	El tipo de datos VARIANT del parámetro DEST_VARIABLE no contiene ningún valor.
8251	Error de creación de código en el parámetro DEST_VARIABLE
8254	Tipo de datos no válido en el parámetro DEST_VARIABLE
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Significado
L 0	// El valor "0" se carga en el acumulador 1.

AWL	Significado
T #BufferPos	// El valor "0" se transfiere al operando "#BufferPos".
CALL Deserialize	// Se llama la instrucción "Deserializar".
SRC_ARRAY := "Buffer".Field	// Se deserializan los datos de cliente del bloque de datos "Buffer".
RET_VAL := #Error	// Información de error
DEST_VARIABLE := "Target".Client	// Los datos de cliente deserializados se escriben en el bloque de datos "Target".
POS := #BufferPos	// El número de bytes que ocupan los datos de cliente convertidos se guarda en el operando "#BufferPos".
AN BR JC end	// Si ocurre un error durante la ejecución de la instrucción, la ejecución del programa salta al final.
CALL Deserialize	// Se llama la instrucción "Deserializar".
SRC_ARRAY := "Buffer".Field	// Se deserializa el separador del bloque de datos "Buffer". El separador se ha insertado en forma de caracteres 'arti' o 'Bill' para poder diferenciar mejor los datos de cliente.
RET_VAL := #Error	// Información de error
DEST_VARIABLE := #Label	// Los caracteres deserializados del separador se escriben en el operando #Label.
POS := #BufferPos	// El número de bytes que ocupan los caracteres convertidos se guarda en el operando "#BufferPos".
AN BR JC end	// Si ocurre un error durante la ejecución de la instrucción, la ejecución del programa salta al final.
CALL S_COMP	// Se llama la instrucción "Comparar cadena de caracteres".
src_type := STRING	// Tipo de datos de la instrucción
relation := EQ	
IN1 := 'arti' IN2 := #Label	// Compara los caracteres 'arti' con los caracteres que se han guardado en el operando #Label.
OUT := #cmp	// Resultado de la instrucción
A #cmp JCN noarticle	// Si la comparación devuelve el estado lógico "0", significa que en el operando #Label no se encuentran los caracteres 'arti', y la ejecución del programa salta a la etiqueta "noarticle".
CALL Deserialize	// Si el resultado de la comparación = "1", se llama la instrucción "Deserializar".
SRC_ARRAY := "Buffer".Field	// Los datos de artículo se deserializan.

AWL	Significado
RET_VAL := #Error	// Información de error
DEST_VARIABLE := "Target".Article[#DeliverPos]	// Los datos de artículo deserializados se escriben en el bloque de datos "Target".
POS := #BufferPos	// El número de bytes que ocupan los datos de artículo convertidos se guarda en el operando #BufferPos.
AN BR JC end BE	// Si no se ha producido ningún error durante la ejecución, se alcanza el fin de bloque.
noarticle: NOP 0	// Si la comparación de si es 'arti' ha sido = "0", el programa se sigue ejecutando.
CALL S_COMP	// Se llama la instrucción "Comparar cadena de caracteres".
src_type := STRING	// Tipo de datos de la instrucción
relation := EQ	
IN1 := 'Bill' IN2 := #Label	// Compara los caracteres 'Bill' con los caracteres que se han guardado en el operando #Label.
OUT := #cmp	// Resultado de la instrucción
A #cmp JCN end	// Si la comparación devuelve el estado lógico "1", significa que en el operando #Label se encuentran los caracteres 'Bill', y el programa se sigue ejecutando.
CALL Deserialize	// Si el resultado de la comparación = "1", se llama la instrucción "Deserializar".
SRC_ARRAY := "Buffer".Field	// Los datos de cálculo se deserializan.
RET_VAL := #Error	// Información de error
DEST_VARIABLE := "Target".Bill[#DeliverPos]	// Los datos de cliente deserializados se escriben en el bloque de datos "Target".
POS := #BufferPos	// El número de bytes que ocupan los datos de artículo convertidos se guarda en el operando #BufferPos.
AN BR JC end BE	// Si no se ha producido ningún error durante la ejecución, se alcanza el fin de bloque.
end: CLR	// En caso de error, se salta a la etiqueta "end".
SAVE	// El RLO se guarda en el bit BR.

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Target	Client	"Client" (tipo de datos PLC)
	Article	Array[0..] of "Article" (tipo de datos PLC)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

Serialize: Serializar

Descripción

Mediante la instrucción "Serializar" se convierten varios tipos de datos PLC (UDT) a una forma de representación secuencial sin que se pierdan partes de su estructura.

La instrucción permite guardar temporalmente varios datos estructurados del programa en un búfer, ubicado preferentemente en un bloque de datos global, y enviarlos a otra CPU. El área de memoria en la que se depositen los tipos de datos PLC convertidos debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

El operando del parámetro POS contiene la información del número de bytes que ocupan los tipos de datos PLC convertidos.

Nota

Para enviar un solo tipo de datos PLC (UDT), se puede llamar directamente la instrucción "TSEND: Enviar datos a través de la conexión".

Sintaxis

Para la instrucción "Serializar" se utiliza la siguiente sintaxis:

```
CALL Serialize
SRC_VARIABLE :=
<Operando>
RET_VAL := <Operando>
DEST_ARRAY := <Operando>
POS := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Serializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC_VARIABLE	Input	VARIANT	I, Q, D, L	Tipo de datos PLC (UDT) que se convierte a una forma de representación secuencial.
DEST_ARRAY	InOut	VARIANT	I, Q, D, L	Bloque de datos en el que se guarda el flujo de datos generado.
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_VARIABLE y DEST_ARRAY se solapan.
8150	El tipo de datos VARIANT del parámetro SRC_VARIABLE no contiene ningún valor.
8152	Error de creación de código en el parámetro SRC_VARIABLE
8236	El bloque de datos del parámetro DEST_ARRAY no es un bloque con acceso estándar.
8250	El tipo de datos VARIANT del parámetro DEST_ARRAY no contiene ningún valor.
8252	Error de creación de código en el parámetro DEST_ARRAY
8253	En el parámetro DEST_ARRAY no hay suficiente memoria disponible.
8254	Tipo de datos no válido en el parámetro DEST_ARRAY
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                                  Explicación
L 0                                  // El valor "0" se carga en el acumulador 1.
T #BufferPos                         // El valor "0" se transfiere al operando
                                     "#BufferPos".

CALL Serialize                       // Se llama la instrucción "Serializar".
SRC_VARIABLE := "Source".Client      // Se serializan los datos de cliente del
                                     bloque de datos "Source".Client.
RET_VAL := #Error                    // Información de error
DEST_ARRAY := "Buffer".Field         // La representación secuencial se escribe en
                                     el bloque de datos "Buffer".Field.
POS := #BufferPos                    // El número de bytes que ocupa la
                                     representación secuencial se escribe en el
                                     operando "#BufferPos".

CALL S_MOVE                          // Se llama la instrucción "Desplazar cadena
                                     de caracteres".
src_type := STRING                   // Tipos de datos de la instrucción
dest_type := STRING
```

AWL	Explicación
IN := 'arti' OUT := #Label	// Los caracteres 'arti' se copian en el operando "#Label".
CALL Serialize	// Se llama la instrucción "Serializar".
SRC_VARIABLE := #Label	// Los caracteres se serializan.
RET_VAL := #Error	// Información de error
DEST_ARRAY := "Buffer".Field	// La representación secuencial se escribe en el bloque de datos "Buffer".Field después de los datos de cliente.
POS := #BufferPos	// El número de bytes que requieren los caracteres se suman en el operando "#BufferPos" al número ya guardado.
CALL Serialize	// Se llama la instrucción "Serializar".
SRC_VARIABLE := "Source".Article[#DeliverPos]	// Se serializan los datos de un artículo determinado del bloque de datos "Source".Article[#DeliverPos] que se calcula en tiempo de ejecución.
RET_VAL := #Error	// Información de error
DEST_ARRAY := "Buffer".Field	// La representación secuencial se escribe en el bloque de datos "Buffer".Field después de los caracteres 'arti'.
POS := #BufferPos	// Número de bytes

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Source	Client	"Client" (tipo de datos PLC)
	Article	Array[0..10] of "Article" (tipo de datos PLC)
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

MOVE_BLK: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento del parámetro IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en el parámetro IN o en el parámetro OUT.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

```
CALL MOVE_BLK
<Tipo de datos> <Tipo de datos>
IN := <Operando>
COUNT := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino
OUT [1]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.				

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL MOVE_BLK	// Se llama la instrucción "Copiar área".
value_type := INT	// Tipos de datos de la instrucción
count_type := UINT	// Tipo de datos del parámetro COUNT
IN := #a_array[2]	// Primer elemento desde el que se copia.
COUNT := "Tag_Count"	// Número de elementos que se copian al área de destino.
OUT := #b_array[1]	// Primer elemento en el que se copia.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

MOVE_BLK_VARIANT: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). Se pueden copiar elementos de un array en otro array del mismo tipo de datos. Los tamaños (número de elementos) de los arrays de origen y de destino pueden ser diferentes. Se pueden copiar varios elementos dentro de un array o bien copiar los elementos uno a uno.

En el momento de utilizar la instrucción para crear un bloque, aún no tiene que conocerse el array, puesto que el origen y el destino se transfieren por VARIANT.

El contaje de los parámetros SRC_INDEX y DEST_INDEX siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La instrucción no se ejecuta si se copian más datos de los que están disponibles.

Si el puntero VARIANT (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

```
CALL MOVE_BLK_VARIANT
SRC := <Operando>
COUNT := <Operando>
SRC_INDEX := <Operando>
DEST_INDEX := <Operando>
DEST := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC	Input	VARIANT (array o elemento individual)	L	El área de origen de la que se copia
COUNT	Input	UDINT	I, Q, M, D, L o constante	Número de elementos que se copian Asigne al parámetro COUNT el valor "1" si en el parámetro SRC o en el parámetro DEST no hay indicado ningún Array.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro SRC_INDEX se calcula basado en cero. Si en el parámetro SRC hay indicado un Array, el entero del parámetro SRC_INDEX indica el primer elemento del área de origen de la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro SRC no hay indicado ningún Array o solo hay indicado un elemento individual de un Array, asigne al parámetro SRC_INDEX el valor "0".
DEST_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro DEST_INDEX se calcula basado en cero. Si en el parámetro DEST hay indicado un Array, el entero del parámetro DEST_INDEX indica el primer elemento del área de destino en la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro DEST no hay indicado ningún Array, asigne al parámetro DEST_INDEX el valor "0".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DEST	Output	VARIANT	L	Área de destino a la que se copia el contenido del área de origen.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
8151	No es posible acceder al parámetro SRC.
8152	El operando del parámetro SRC no es de ningún tipo.
8153	Error de creación de código en el parámetro SRC
8154	El operando del parámetro SRC es del tipo de datos BOOL.
8281	Valor no válido en el parámetro COUNT
8382	El valor del parámetro SRC_INDEX se encuentra fuera de los valores límite de VARIANT.
8383	El valor del parámetro SRC_INDEX se encuentra fuera del límite superior del array.
8482	El valor del parámetro DEST_INDEX se encuentra fuera de los valores límite de VARIANT.
8483	El valor del parámetro DEST_INDEX se encuentra fuera del límite superior del array.
8534	El parámetro DEST está protegido contra escritura
8551	No es posible acceder al parámetro DEST.
8552	El operando del parámetro DEST no es de ningún tipo.
8553	Error de creación de código en el parámetro DEST
8554	El operando del parámetro DEST es del tipo de datos BOOL.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL MOVE_BLK_VARIANT	// Se llama la instrucción "Copiar área".
SRC := #SrcField	// Primer elemento que se copia
COUNT := "Tag_Count"	// Número de elementos que se copian al área de destino.
SRC_INDEX := "Tag_Src_Index"	// Primer elemento del parámetro SRC que se copia
DEST_INDEX := "Tag_Dest_Index"	// Primer elemento del parámetro DEST que se copia
DEST := #DestField	// Primer elemento del área de origen que se copia.
RET_VAL := "Tag_Result"	// Información de error

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC	#SrcField	El operando local #SrcField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [0..10] of "MOVE_UDT"
COUNT	Tag_Count	2
SRC_INDEX	Tag_Src_Index	3
DEST_INDEX	Tag_Dest_Index	3
DEST	#DestField	El operando local #DestField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [10..20] of "MOVE_UDT"

En el área de destino se copian 2 elementos del área de origen, empezando por el cuarto elemento del array de UDT. Las copias se insertan en el Array of UDT a partir del cuarto elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UMOVE_BLK: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia sin interrupciones los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento de la entrada IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Copiar área sin interrupciones".

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en el parámetro IN o en el parámetro OUT.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

```
CALL UMOVE_BLK
<Tipo de datos> <Tipo de datos>
IN := <Operando>
COUNT := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino
OUT [1]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.				

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL UMOVE_BLK               // Se llama la instrucción "Copiar área sin
                             interrupciones".
value_type := INT            // Tipos de datos de la instrucción
count_type := UINT           // Tipo de datos del parámetro COUNT
IN := #a_array[2]            // Primer elemento desde el que se copia.
COUNT := "Tag_Count"         // Número de elementos que se copian al área de
                             destino.
OUT := #b_array[1]           // Primer elemento en el que se copia.
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

FILL_BLK: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el valor de la entrada IN. El área de destino se rellena a partir de la dirección indicada en la salida OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

```
CALL FILL_BLK
```

```
<Tipo de datos> <Tipo de datos>
IN := <Operando>
COUNT := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Dirección del área de destino a partir de la cual se rellena

[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura Array.
[2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura Array.

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL FILL_BLK	// Se llama la instrucción "Rellenar área".
value_type := INT	// Tipos de datos de la instrucción
count_type := UINT	// Tipo de datos del parámetro COUNT
IN := a_array[2]	// Elemento que debe copiarse.
COUNT := "Tag_Count"	// Número de repeticiones de copia
OUT := b_array[1]	// El área de destino se rellena a partir de la dirección indicada en la salida OUT.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

UFILL_BLK: Rellenar área sin interrupciones

Descripción

La instrucción "Rellenar área sin interrupciones" rellena sin interrupciones un área de memoria (área de destino) con el valor de la entrada IN. El área de destino se rellena a partir de la dirección indicada en la salida OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Rellenar área sin interrupciones".

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

La instrucción "Rellenar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Sintaxis

Para la instrucción "Rellenar área sin interrupciones" se utiliza la siguiente sintaxis:

```
CALL UFILL_BLK
<Tipo de datos> <Tipo de datos>
IN := <Operando>
COUNT := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Dirección del área de destino a partir de la cual se rellena

[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY.
[2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL UFILL_BLK	// Se llama la instrucción "Rellenar área sin interrupciones".

AWL	Explicación
`value_type := INT`	`// Tipos de datos de la instrucción`
`count_type := UINT`	`// Tipo de datos del parámetro COUNT`
`IN := "Tag_IN"`	`// Elemento que debe copiarse`
`COUNT := "Tag_Count"`	`// Número de repeticiones de copia`
`OUT := "Tag_OUT"`	`// El área de destino se rellena a partir de la dirección indicada en la salida OUT.`

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Instrucciones ARRAY-DB

ReadFromArrayDB: Leer desde bloque de datos de array

Descripción

La instrucción "Leer en un bloque de datos ARRAY" permite leer datos en un DB de array y escribirlos en un área de destino.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Sintaxis

Para la instrucción "Leer en un bloque de datos ARRAY" se utiliza la siguiente sintaxis:

```
CALL ReadFromArrayDB
DB := <Operando>
INDEX := <Operando>
RET_VAL := <Operando>
VALUE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer desde bloque de datos de array":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos desde el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
VALUE	Output	VARIANT	L	Valor que se lee y se devuelve
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El DB de array contiene valores no válidos
8154	El tipo de datos no está declarado correctamente
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL ReadFromArrayDB	// Llamada de la instrucción
DB := "ArrayDB"	// El bloque de datos en el que se lee.
INDEX := "ArrayDB".THIS[2]	// El elemento [2] se lee en el bloque de datos.
RET_VAL := "TagRet_Val"	// Información de error
VALUE := "TargetField[10]".Data2[1]	// El valor leído se devuelve en el parámetro VALUE.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetros	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of DINT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	TargetField[10].Data2[1]	El operando "TargetField" es del tipo de datos Array [10..20] of "MOVE_UDT".

El tercer elemento se lee en el "ArrayDB" y se escribe en el operando "TargetField[10].Data2[1]".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

WriteToArrayDB: Escribir en bloque de datos de array

Descripción

La instrucción "Escribir en bloque de datos de array" permite escribir datos en un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Sintaxis

Para la instrucción "Escribir en un bloque de datos ARRAY" se utiliza la siguiente sintaxis:

```
CALL WriteToArrayDB
DB := <Operando>
INDEX := <Operando>
RET_VAL := <Operando>
VALUE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en bloque de datos de array":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	L	Valor que se escribe
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL WriteToArrayDB	// Llamada de la instrucción
DB := "ArrayDB"	// Bloque de datos en el que se escribe.
INDEX := "ArrayDB".THIS[2]	// El valor se escribe en el elemento [2] del bloque de datos.
RET_VAL := "TagRet_Val"	// Información de error
VALUE := "SourceField[1]".Data1[6]	// El valor que se escribe en el bloque de datos.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of DINT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceField[1].Data1[6]	El operando "SourceField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Del operando "SourceField" se escribe el elemento "Data1[6]" del segundo elemento en el "ArrayDB". En el "ArrayDB" se escribe el tercer elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ReadFromArrayDBL: Leer de bloque de datos de array de la memoria de carga

Descripción

La instrucción "Leer de bloque de datos de array de la memoria de carga" permite leer datos desde un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos

simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción. Durante un ciclo de programa, el parámetro DONE tiene el estado lógico "1" y el valor leído en el parámetro VALUE se emite dentro de dicho ciclo. En los demás ciclos del programa, el valor del parámetro VALUE no se modifica.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

Sintaxis

Para la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga" se utiliza la siguiente sintaxis:

```
CALL ReadFromArrayDBL,
"DB de instancia"
REQ := <Operando>
DB := <Operando>
INDEX := <Operando>
BUSY := <Operando>
DONE := <Operando>
ERROR := <Operando>
VALUE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer de bloque de datos de array de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la lectura del DB de array
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos desde el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La lectura del DB de array todavía no ha finalizado
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
VALUE	InOut	VARIANT	D, L	Valor que se lee y se devuelve
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura

Código de error* (W#16#...)	Explicación
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`CALL ReadFromArrayDBL, "ReadFromArrayDBL_DB"`	`// Llamada de la instrucción`
`REQ := "TagReg"`	`// Cuando se produce un flanco de señal ascendente, se ejecuta la instrucción.`
`DB := "ArrayDB"`	`// El bloque de datos en el que se lee.`
`INDEX := "ArrayDB".THIS[2]`	`// El elemento [2] se lee en el bloque de datos.`
`BUSY := "TagBusy"`	`// Indica si la instrucción aún está en ejecución.`
`DONE := "TagDone"`	`// Indica si la ejecución ha finalizado.`
`ERROR := "TagError"`	`// Información de error`
`VALUE := "SourceTargetField[1]".Data1[6]`	`// El valor leído se devuelve en el parámetro VALUE.`

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of DINT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Parámetro	Operando	Valor
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga". El tercer elemento se lee en el "ArrayDB" y se indica en el operando "SourceTargetField[1].Data1[6]". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE ya no se modifica. Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

WriteToArrayDBL: Escribir en bloque de datos de array de la memoria de carga

Descripción

La instrucción "Escribir en bloque de datos de array de la memoria de carga" permite escribir datos en un DB de array de la memoria de carga.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción y el valor del parámetro VALUE se escribe en el bloque de datos. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en

el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

Sintaxis

Para la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga" se utiliza la siguiente sintaxis:

```
CALL WriteToArrayDBL,
"DB de instancia"
REQ := <Operando>
DB := <Operando>
INDEX := <Operando>
VALUE := <Operando>
BUSY := <Operando>
DONE := <Operando>
ERROR := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en bloque de datos de array de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la escritura en el DB de array
DB	Input	DB_ANY	D	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	D, L	Valor que se escribe
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La escritura en el DB de array todavía no ha finalizado

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`CALL WriteToArrayDBL, "WriteToArrayDBL_DB"`	`// Llamada de la instrucción`
`REQ := "TagReg"`	`// Cuando se produce un flanco de señal ascendente, se ejecuta la instrucción.`
`DB := "ArrayDB"`	`// El bloque de datos en el que se lee.`
`INDEX := "ArrayDB".THIS[2]`	`// El elemento [2] se lee en el bloque de datos.`
`BUSY := "TagBusy"`	`// Indica si la instrucción aún está en ejecución.`
`DONE := "TagDone"`	`// Indica si la ejecución ha finalizado.`
`ERROR := "TagError"`	`// Información de error`
`VALUE := "SourceTargetField[1]".Data1[6]`	`// El valor leído se devuelve en el parámetro VALUE.`

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of DINT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE se escribe en el "ArrayDB" en el tercer elemento. Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

Otras

BLKMOV: Copiar área

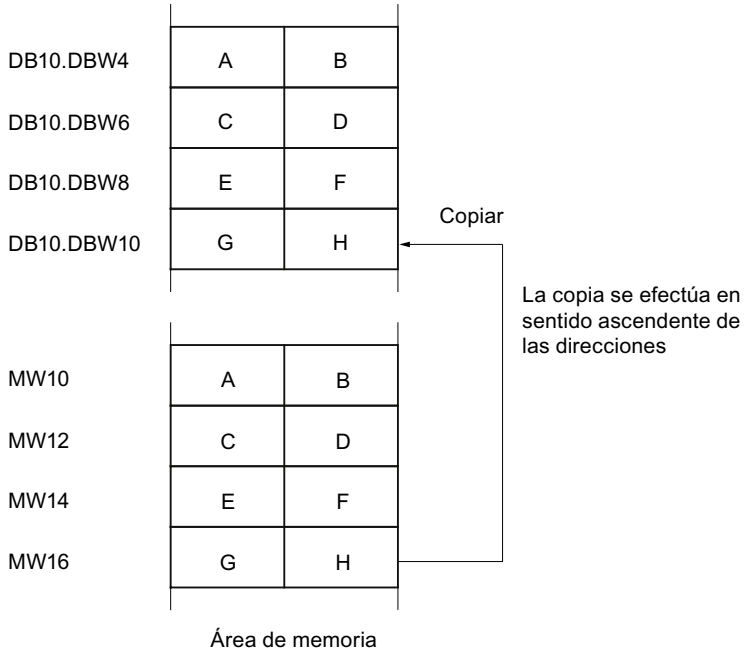
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

No hay limitaciones en cuanto a la profundidad de anidamiento.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

```
CALL BLKMOV
```

```
VARIANT
SRCBLK := <Operando>
RET_VAL := <Operando>
DSTBLK := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	VARIANT	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8092	El área de origen o destino está solo en la memoria de carga.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL BLKMOV	// Se llama la instrucción "Copiar área".
VARIANT	// Tipo de datos de la instrucción
SRCBLK := P#M100.0 BYTE 10	// Se copian 10 bytes a partir de MB100.
RET_VAL := "Tag_ErrorCode"	// Si ocurre un error durante la operación de copia, el respectivo código de error se deposita en la variable "Tag_ErrorCode".

AWL	Explicación
DSTBLK := P#DB1.DBX0.0 BYTE 10	// Los bytes copiados se transmiten en DB1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4013)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

```
CALL UBLKMOV
VARIANT
SRCBLK := <Operando>
RET_VAL := <Operando>
DSTBLK := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	VARIANT	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8091	El área de origen o de destino se encuentra en un bloque de datos no relevante para la ejecución.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL UBLKMOV	// Se llama la instrucción "Copiar área sin interrupciones".
VARIANT	// Tipo de datos de la instrucción
SRCBLK := P#M100.0 BYTE 10	// Se copian 10 bytes a partir de MB100.
RET_VAL := "Tag_ErrorCode"	// Si ocurre un error durante la operación de copia, el respectivo código de error se deposita en la variable "Tag_ErrorCode".
DSTBLK := P#M200.0 BYTE 10	// Los bytes copiados se escriben en los 10 bytes a partir de MB200.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4013)

FILL: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones.

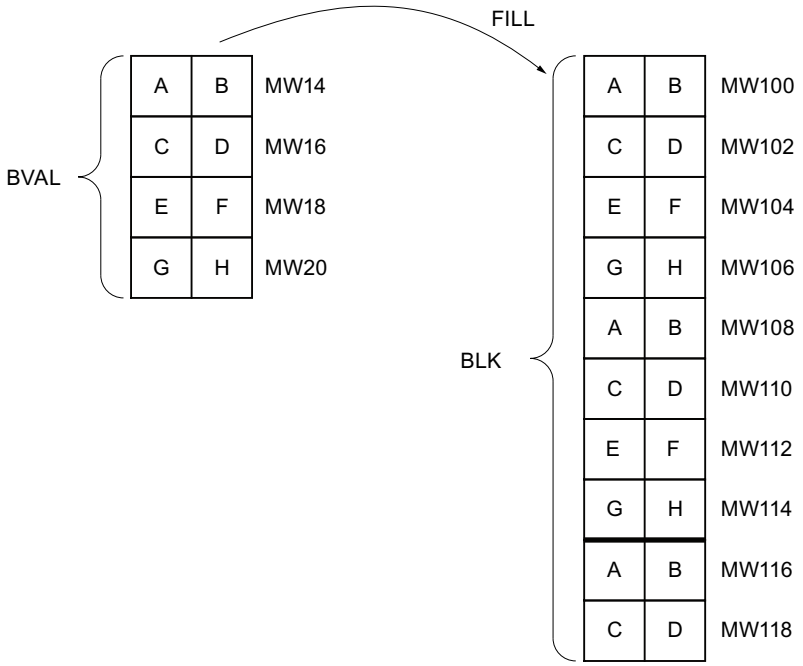
Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Para los bloques con el atributo "Acceso optimizado al bloque" puede utilizarse la instrucción "FILL_BLK: Rellenar área".

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: el contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Reglas para copiar estructuras

Al indicar una estructura a modo de parámetro de entrada, tenga en cuenta que la longitud de una estructura se adapta siempre a un número entero de bytes. Si se declara una estructura con un número impar de bytes, esta estructura necesita un byte adicional de espacio en memoria.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

```
CALL FILL
VARIANT
BVAL := <Operando>
RET_VAL := <Operando>
BLK := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BVAL	Input	VARIANT	L, P	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
BLK	Output	VARIANT	L, P	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL FILL	// Se llama la instrucción "Rellenar área".
VARIANT	// Tipo de datos de la instrucción
BVAL := P#M14.0 WORD 4	// Se copia el área de MW14 a MW20.
RET_VAL := "Tag_ErrorCode"	// Información de error
BLK := P#M100.0 WORD 10	// El área de memoria de MW100 a MW118 se rellena con el contenido de las 4 palabras contenidas en el área de memoria del parámetro BVAL.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4013)

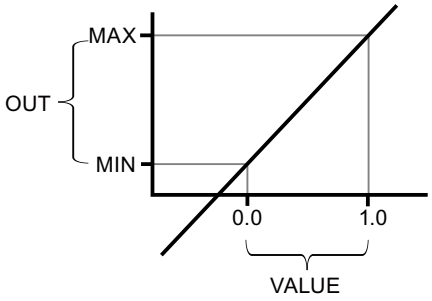
Conversión

SCALE_X: Escalar

Descripción

La instrucción "Escalar" escala el valor de la entrada VALUE mapeándolo en un determinado rango de valores. Al ejecutar la instrucción "Escalar", el número en coma flotante de la entrada VALUE se escala al rango de valores definido por los parámetros MIN y MAX. El resultado de la escala es un número entero que se deposita en la salida RET_VAL.

La figura siguiente muestra un ejemplo de cómo pueden escalarse los valores:



La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [VALUE * (MAX – MIN)] + MIN

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

```
CALL SCALE_X
<Tipo de datos> <Tipo de datos>
MIN := <Operando>
```

```
VALUE := <Operando>
MAX := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE	Input	Números en coma flotante	I, Q, M, D, L o constante	Valor que se escala. Si se indica una constante, esta debe declararse.
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
RET_VAL	Output	Enteros, números en coma flotante	I, Q, M, D, L	Resultado de la escala

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL SCALE_X                 // Se llama la instrucción "Escalar".
dest_type := INT             // Tipos de datos de la instrucción
src_type := REAL             // Tipo de datos del parámetro VALUE
MIN := "Tag_Minimum"         // Valor límite inferior
VALUE := "Tag_Value"         // Valor en coma flotante
MAX := "Tag_Maximum"         // Valor límite superior
RET_VAL := "Tag_Result"      // Resultado de la instrucción
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_Minimum	10
VALUE	Tag_Value	0.5
MAX	Tag_Maximum	30
OUT	Tag_Result	20

El valor de la entrada "Tag_Value" se escala al rango de valores definido por los valores de las entradas "Tag_Minimum" y "Tag_Maximum". El resultado se deposita en la salida "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

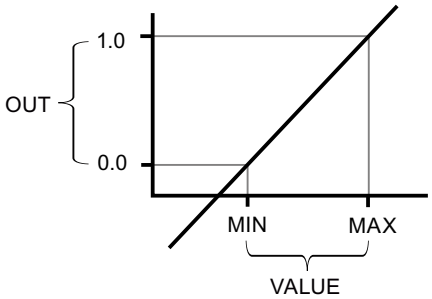
Declarar constantes globales (Página 2584)

NORM_X: Normalizar

Descripción

La instrucción "Normalizar" normaliza el valor de la variable de la entrada VALUE representándolo en una escala lineal. Los parámetros MIN y MAX sirven para definir los límites de un rango de valores que se refleja en la escala. En función de la posición del valor que se debe normalizar en este rango de valores, el resultado se calcula y se deposita como número en coma flotante en la salida RET_VAL. Si el valor que se debe normalizar es igual al valor de la entrada MIN, la salida OUT devuelve el valor "0.0". Si el valor que se debe normalizar es igual al de la entrada MAX, la salida OUT devuelve el valor "1.0".

La figura siguiente muestra un ejemplo de cómo pueden normalizarse los valores:



La instrucción "Normalizar" utiliza la siguiente ecuación:

OUT = (VALUE – MIN) / (MAX – MIN)

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Sintaxis

Para la instrucción "Normalizar" se utiliza la siguiente sintaxis:

```
CALL NORM_X
<Tipo de datos> <Tipo de datos>
MIN := <Operando>
VALUE := <Operando>
MAX := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Normalizar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L	Límite inferior del rango de valores
VALUE [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L	Valor que se normaliza

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MAX [1]	Input	Enteros, números en coma flotante	I, Q, M, D, L	Límite superior del rango de valores
RET_VAL	Output	Números en coma flotante	I, Q, M, D, L	Resultado de la normalización
[1] Si se utilizan constantes en estos tres parámetros, basta con declarar una de ellas.				

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL NORM_X                  // Se llama la instrucción "Normalizar".
src_type := INT              // Tipos de datos de la instrucción
dest_type := REAL            // Tipo de datos del parámetro RET_VAL
MIN := "Tag_Minimum"         // Valor límite inferior
VALUE := "Tag_Value"         // Valor en coma flotante
MAX := "Tag_Maximum"         // Valor límite superior
RET_VAL := "Tag_Result"      // Resultado de la instrucción
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_Minimum	10
VALUE	Tag_Value	20
MAX	Tag_Maximum	30
RET_VAL	Tag_Result	0.5

El valor de la entrada "Tag_Value" se asigna al rango de valores definido por los valores de las entradas "Tag_Minimum" y "Tag_Maximum". El valor de la variable de la entrada "Tag_Value" se normaliza conforme al rango de valores definido. El resultado se deposita como número en coma flotante en la salida "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)
Declarar constantes globales (Página 2584)

Instrucciones VARIANT

VARIANT_TO_DB_ANY: Convertir VARIANT en DB_ANY

Descripción

La instrucción "Convertir VARIANT en DB_ANY" permite leer el número del bloque de datos de un VARIANT y convertirlo al tipo de datos DB_ANY.

Requisitos

Si se cumplen los requisitos, se ejecuta la instrucción. Si no se cumplen los requisitos, se devuelve "0" como número del bloque de datos.

La variable de salida...	referencia...	Posibilidades de conversión
VARIANT	... un bloque de datos instancia de un UDT o de un SDT	Es posible convertir la variable de salida a un número del bloque de datos.
VARIANT	... un DB de array	Es posible convertir la variable de salida a un número del bloque de datos.
VARIANT	... un objeto con un tipo de datos simple	No es posible convertir la variable de salida a un número del bloque de datos, puesto que un bloque de datos nunca consta solamente de un tipo de datos simple.
VARIANT	... una estructura de un bloque de datos	No es posible convertir la variable de salida a un número del bloque de datos, puesto que solo se trata de una parte del bloque de datos.

Sintaxis

Para la instrucción "Convertir VARIANT en DB_ANY" se utiliza la siguiente sintaxis:

```
CALL VARIANT_TO_DB_ANY
IN := <Operando>
ERR := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir VARIANT en DB_ANY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	VARIANT	L	Variable que se lee
ERR	Output	INT	I, Q, M, D, L	Información de error
RET_VAL	Output	DB_ANY	I, Q, M, D, L	Resultado: número de DB

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR

La tabla siguiente muestra el significado de los valores del parámetro ERR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
252C	El tipo de datos VARIANT devuelve el valor "0"
8130	El número del bloque de datos es "0"
8131	El bloque de datos no existe
8132	El bloque de datos es demasiado corto
8134	El bloque de datos está protegido contra escritura
8150	El tipo de datos VARIANT devuelve el valor "0"
8154	El tipo de datos no está declarado correctamente
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL VARIANT_TO_DB_ANY	// Se llama la instrucción "Convertir VARIANT en DB_ANY".
IN := #InputofTypeVARIANT	// Se lee la variable #InputofTypeVARIANT.
RET_VAL := "OutputofTypeDBANY"	// El número del DB se escribe en la variable "OutputofTypeDBANY".
ERR := "Tag_Error"	// Información de error

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)
Declarar constantes globales (Página 2584)
VARIANT (Página 2364)

DB_ANY_TO_VARIANT: Convertir DB_ANY en VARIANT

Descripción

La instrucción "Convertir DB_ANY en VARIANT" permite leer el número del bloque de datos de un DB_ANY y convertirlo al tipo de datos VARIANT.

Requisitos

Si se cumplen los requisitos, se ejecuta la instrucción. Si no se cumplen los requisitos o el bloque de datos no existe, se devuelve el valor "0". Los demás accesos a este valor "0" fallan.

La variable de salida...	referencia...	Posibilidades de conversión
DB_ANY	... un bloque de datos instancia de un UDT o de un SDT	Es posible convertir la variable de salida a un tipo de datos VARIANT.
DB_ANY	... un bloque de datos de array	Es posible convertir la variable de salida a un tipo de datos VARIANT.
DB_ANY	... un bloque de datos instancia de un bloque de función o un bloque de datos global	No es posible convertir la variable de salida a un tipo de datos VARIANT.

Sintaxis

Para la instrucción "Convertir DB_ANY en VARIANT" se utiliza la siguiente sintaxis:

```
CALL DB_ANY_TO_VARIANT
IN := <Operando>
ERR := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir DB_ANY en VARIANT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DB_ANY	I, Q, M, D, L	Variable que se lee
ERR	Output	INT	I, Q, M, D, L	Información de error
RET_VAL	Output	VARIANT	L	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR

La tabla siguiente muestra el significado de los valores del parámetro ERR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8130	El número del bloque de datos es "0"
8131	El bloque de datos no existe
8132	El bloque de datos es demasiado corto
8134	El bloque de datos está protegido contra escritura
8154	El tipo de datos no está declarado correctamente
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL DB_ANY_TO_VARIANT	// Se llama la instrucción "Convertir DB_ANY en VARIANT".
IN := #InputofTypeDBANY	// Se lee la variable #InputofTypeDBANY.
RET_VAL := "OutputofTypeVARIANT"	// El número del DB se escribe en la variable "OutputofTypeVARIANT".
ERR := "Tag_Error"	// Información de error

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Declarar constantes globales (Página 2584)

VARIANT (Página 2364)

Otras

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": Se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27 648 y 27 648. En este caso, la constante "K1" tiene el valor "-27 648,0", y la constante "K2", el valor "+27 648,0".
- Estado lógico "0": Se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27 648. En este caso, la constante "K1" tiene el valor "0,0", y la constante "K2", el valor "+27 648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

```
CALL SCALE
IN := <Operando>
HI_LIM := <Operando>
LO_LIM := <Operando>
BIPOLAR := <Operando>
```

```
RET_VAL := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1".
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL SCALE	// Se llama la instrucción "Escalar".

```
AWL                          Explicación
IN := "Tag_InputValue"       // Indica el valor que se convierte y escala.
HI_LIM := "Tag_HighLimit"    // Valor límite superior
LO_LIM := "Tag_LowLimit"     // Valor límite inferior
BIPOLAR := "Tag_Bipolar"     // Indica si el valor del parámetro IN se interpreta
                             como bipolar o unipolar.
RET_VAL := "Tag_ErrorCode"   // Información de error
OUT := "Tag_OutputValue"     // Resultado de la instrucción
```

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4013)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" desescala el número en coma flotante del parámetro IN en unidades físicas entre un valor límite inferior y un valor límite superior y lo convierte en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": Se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27 648 y 27 648. En este caso, la constante "K1" tiene el valor "-27 648,0", y la constante "K2", el valor "+27 648,0".
- Estado lógico "0": Se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27 648. En este caso, la constante "K1" tiene el valor "0,0", y la constante "K2", el valor "+27 648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Sintaxis

Para la instrucción "Desescalar" se utiliza la siguiente sintaxis:

```
CALL UNSCALE
IN := <Operando>
HI_LIM := <Operando>
LO_LIM := <Operando>
BIPOLAR := <Operando>
RET_VAL := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL UNSCALE	// Se llama la instrucción "Desescalar".
IN := "Tag_InputValue"	// Indica el valor que se desescala.
HI_LIM := "Tag_HighLimit"	// Valor límite superior

```
AWL                              Explicación
LO_LIM := "Tag_LowLimit"         // Valor límite inferior
BIPOLAR := "Tag_Bipolar"         // Indica si el valor del parámetro IN se interpreta
                                 como bipolar o unipolar.
RET_VAL := "Tag_ErrorCode"       // Información de error
OUT := "Tag_OutputValue"         // Resultado de la instrucción
```

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4013)

Control del programa

Control del tiempo de ejecución

ENDIS_PW: Limitar y habilitar legitimación de la contraseña

Descripción

La instrucción "Limitar y habilitar legitimación de la contraseña" permite definir si las contraseñas configuradas para la CPU están legitimadas o no. Con ello, se pueden impedir conexiones legitimadas, aun conociendo la contraseña correcta.

Si se llama la instrucción y el parámetro REQ tiene el estado lógico "0", en los parámetros de salida se muestra el estado ajustado en ese momento. Si se han realizado cambios en los parámetros de entrada, estos no se transfieren a los parámetros de salida.

Si se llama la instrucción y el parámetro REQ tiene el estado lógico "1", se aplica el estado lógico de los parámetros de entrada (F_PWD, FULL_PWD, R_PWD, HMI_PWD). FALSE significa que la legitimación por contraseña no está permitida y TRUE significa que la contraseña puede utilizarse.

Es posible permitir o prohibir individualmente el bloqueo o la habilitación de las contraseñas. Por ejemplo, pueden prohibirse todas las contraseñas excepto la contraseña de seguridad positiva. Esto permite limitar las posibilidades de acceso a un reducido grupo de usuarios. Los parámetros de salida (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) siempre muestran el estado actual del uso de contraseñas, independientemente del parámetro REQ.

Las contraseñas no configuradas deben tener el estado lógico TRUE en la entrada y devolver el estado lógico TRUE en la salida. La contraseña de seguridad positiva (fail-safe) solo se puede parametrizar para una CPU F y por eso en una CPU estándar debe interconectarse siempre con el estado lógico TRUE. Si la instrucción devuelve un error, la llamada permanece sin efecto, es decir, el bloqueo anterior sigue en vigor.

Las contraseñas bloqueadas pueden volver a habilitarse en las condiciones siguientes:

- Se ha restablecido la configuración de fábrica de la CPU.
- El panel frontal de la CPU S7-1500 soporta un cuadro de diálogo que permite navegar al menú adecuado y volver a habilitar las contraseñas.
- Cuando se llama la instrucción "Limitar y habilitar legitimación de la contraseña", el parámetro de entrada de la contraseña deseada tiene el estado lógico "1".
- Ponga el selector de modo a STOP. La restricción de la legitimación de contraseña se activa de nuevo en cuanto se mueve el selector nuevamente a RUN.
- La inserción de una tarjeta de memoria vacía (tarjeta de transferencia o tarjeta de programa) en una CPU S7-1200.
- La transición de POWER OFF-POWER ON desactiva la protección en la CPU S7-1200. La instrucción "Limitar y habilitar legitimación de la contraseña" debe volverse a llamar en el programa (por ejemplo en el OB de arranque).

Nota

Si la contraseña HMI no está habilitada, la instrucción "Limitar y habilitar legitimación de la contraseña" bloquea el acceso de los sistemas HMI.

Nota

Las conexiones ya existentes y legitimadas conservan sus derechos de acceso y no pueden limitarse con la instrucción "Limitar y habilitar legitimación de la contraseña".

Impedir el bloqueo no intencionado en una CPU S7-1500

Los ajustes pueden realizarse en el panel frontal de la CPU, que guarda el último ajuste en cada caso.

Para impedir un bloqueo no intencionado, en una CPU S7-1500 es posible desactivar la protección moviendo el selector de modo a STOP. La protección se activa automáticamente después de mover el selector de modo a RUN sin que tenga que volver a llamarse la instrucción "Limitar y habilitar legitimación de la contraseña" o deban realizarse otras acciones en el panel frontal.

Impedir el bloqueo no intencionado en una CPU S7-1200

Como la CPU S7-1200 no dispone de selector de modo, la protección se desactiva con POWER OFF-POWER ON. De este modo es posible y recomendable impedir el bloqueo no intencionado con ayuda de determinadas secuencias en el programa.

Para ello, programe un control de tiempo, ya sea mediante un OB de alarma cíclica o un temporizador en el Main OB (OB 1). De este modo, tras una transición de POWER OFF-POWER ON y la consiguiente desactivación de la protección, se puede volver a llamar rápidamente la instrucción "Limitar y habilitar legitimación de la contraseña" en el OB correspondiente (por ejemplo OB 1 u OB 35). Para que la ventana temporal en la que la instrucción no está activa y en la que por tanto no existen limitaciones en la legitimación de la contraseña sea lo más pequeña posible, llame la instrucción en el OB de arranque (OB 100). Este procedimiento ofrece la mayor protección posible contra accesos no autorizados.

Si se ha producido un bloqueo no intencionado, puede omitirse la llamada en el OB de arranque (por ejemplo mediante consulta de un parámetro de entrada) y se dispondrá del tiempo configurado (por ejemplo de 10 segundos a 1 minuto) para establecer una conexión con la CPU antes de que el bloqueo vuelva a activarse.

Si no se ha previsto ningún temporizador en el código del programa y se produce un bloqueo, inserte en la CPU una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna de la CPU. Después deberá cargarse nuevamente el programa de usuario de STEP 7 en la CPU.

Procedimiento en caso de pérdida de contraseña en una CPU S7-1200

Si ha perdido la contraseña de una CPU S7-1200 protegida por contraseña, borre el programa protegido por contraseña con una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna

de la CPU. A continuación, puede transferir un nuevo programa de usuario de STEP 7 Basic a la CPU.

⚠ ADVERTENCIA

Inserción de una tarjeta de transferencia vacía

Cuando se inserta una tarjeta de transferencia en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

El contenido de la tarjeta de transferencia se encuentra en la memoria de carga interna después de su extracción. Compruebe aquí que la tarjeta no contenga ningún programa.

⚠ ADVERTENCIA

Inserción de una tarjeta de programa vacía

Cuando se inserta una tarjeta de programa en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

Asegúrese de que la tarjeta de programa esté vacía. La memoria de carga interna se copiará en la tarjeta de programa vacía. Después de retirar la tarjeta de programa previamente vacía, la memoria de carga interna quedará vacía.

Debe retirarse la tarjeta de transferencia o la tarjeta de programa antes de poner la CPU en RUN.

Efectos del uso de contraseñas en los modos de operación

La siguiente tabla muestra qué efectos tiene el uso de contraseñas en los modos de operación y en las respectivas acciones del usuario mediante la instrucción "Limitar y habilitar legitimación de la contraseña".

Acción	Protección por contraseña mediante la instrucción
Estado básico después de • Selector de modo en STOP • Restablecimiento manual de la memoria (PG, interruptor, cambio de MC (Motion Control)) • Restablecimiento de la configuración de fábrica	No activado (sin limitaciones)
Estado básico tras conexión (POWER ON)	• CPU S7-1200: El bloqueo está desactivado y la instrucción tiene que volverse a ejecutar en el programa (por ejemplo en el OB de arranque). • CPU S7-1500: Activada (si antes de la desconexión (POWER OFF) había un bloqueo activo) La posibilidad de no permitir contraseñas es remanente.
Transición del estado operativo RUN/ARRANQUE/PARADA -> STOP (mediante finalización de la instrucción, un error o comunicación) o STOP -> ARRANQUE/RUN/PARADA	Activado Las contraseñas siguen sin poder utilizarse.

Sintaxis

Para la instrucción "Limitar y habilitar legitimación de la contraseña" se utiliza la siguiente sintaxis:

```
CALL ENDIS_PW
REQ := <Operando>
F_PWD := <Operando>
FULL_PWD := <Operando>
R_PWD := <Operando>
HMI_PWD := <Operando>
F_PWD_ON := <Operando>
FULL_PWD_ON := <Operando>
R_PWD_ON := <Operando>
HMI_PWD_ON := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Limitar y habilitar legitimación de la contraseña":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Si el parámetro REQ tiene el estado lógico "0", se consulta el estado lógico ajustado actualmente para las contraseñas.
F_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura incluido failsafe • F_PWD = "0": No permitir contraseña • F_PWD = "1": Permitir contraseña
FULL_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura • FULL_PWD = "0": No permitir contraseña • FULL_PWD = "1": Permitir contraseña
R_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura • R_PWD = "0": No permitir contraseña • R_PWD = "1": Permitir contraseña
HMI_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso a la HMI • HMI_PWD = "0": No permitir contraseña • HMI_PWD = "1": Permitir contraseña
F_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura incluido failsafe • F_PWD_ON = "0": Contraseña no permitida • F_PWD_ON = "1": Contraseña permitida
FULL_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura • FULL_PWD_ON = "0": Contraseña no permitida • FULL_PWD_ON = "1": Contraseña permitida
R_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura • R_PWD_ON = "0": Contraseña no permitida • R_PWD_ON = "1": Contraseña permitida
HMI_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso a la HMI • HMI_PWD_ON = "0": Contraseña no permitida • HMI_PWD_ON = "1": Contraseña permitida
RET_VAL	Output	WORD	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8090	No se soporta la instrucción "Limitar y habilitar legitimación de la contraseña".

Código de error* (W#16#...)	Explicación
80D0	La contraseña de seguridad (fail-safe) no está configurada. En las CPU estándar, el estado lógico tiene que ser TRUE.
80D1	El acceso de lectura/escritura no está configurado
80D2	El acceso de lectura no está configurado
80D3	El acceso HMI no está configurado
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del tiempo de ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar en todos los bloques independientemente de la prioridad.

Si se llama la instrucción en un bloque de prioridad superior, p. ej. en una alarma de proceso o de diagnóstico o en una alarma cíclica, la instrucción no se ejecuta.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar hasta 10 veces en un ciclo de programa.

Sintaxis

Para la instrucción "Reiniciar tiempo de vigilancia del ciclo" se utiliza la siguiente sintaxis:

```
CALL RE_TRIGR
```

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Sintaxis

Para la instrucción "Finalizar programa" se utiliza la siguiente sintaxis:

```
CALL STP
```

Parámetros

La instrucción "Finalizar programa" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

GET_ERROR: Consultar error localmente

Descripción

La instrucción "Consultar error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si el sistema notifica errores mientras se ejecuta el bloque, en el operando de la salida OUT se guarda información detallada acerca del primer error de ejecución ocurrido desde la última vez que se ejecutó la instrucción.

En la salida OUT solo pueden indicarse operandos del tipo de datos de sistema "ErrorStruct". El tipo de datos de sistema "ErrorStruct" especifica la estructura exacta con la que debe guardarse la información acerca del error que ha ocurrido. Con ayuda de otras instrucciones se puede evaluar esta estructura y programar la reacción correspondiente. Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces emite información sobre el siguiente error ocurrido.

Nota

La salida OUT solo se modifica si hay una información de error. Para volver a poner la salida a "0" tras tratar un error existen las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz de bloque.
- Ponga la variable de nuevo a "0" antes de llamar la instrucción.

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar error localmente" activa el tratamiento local de errores en el bloque. Si en la lógica de un bloque se ha insertado "Consultar error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Sintaxis

Para la instrucción "Consultar error localmente" se utiliza la siguiente sintaxis:

```
CALL GET_ERROR
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	ErrorStruct	D, L	Información de error

Tipo de datos "ErrorStruct"

La tabla siguiente muestra la estructura del tipo de datos "ErrorStruct":

Componente de estructura		Tipo de datos	Descripción
ERROR_ID		WORD	ID de error
FLAGS		BYTE	Muestra si el error se ha producido durante la llamada de un bloque. 16#01: Error durante la llamada de un bloque 16#00: No hay errores durante la llamada de un bloque
REACTION		BYTE	Reacción predeterminada: 0: ignorar (error de escritura), 1: continuar con el valor sustitutivo "0" (error de lectura), 2: omitir instrucción (error de sistema)
CODE_ADDRESS		CREF	Información sobre la dirección y el tipo del bloque
	BLOCK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error: 1: OB 2: FC 3: FB
	CB_NUMBER	UINT	Número del bloque lógico
	OFFSET	UDINT	Referencia a la memoria interna
MODE		BYTE	Información sobre la dirección de un operando
OPERAND_NUMBER		UINT	Número de operando del comando de la máquina
POINTER_NUMBER_LOCATION		UINT	(A) Puntero interno
SLOT_NUMBER_SCOPE		UINT	(B) Área de almacenamiento en la memoria interna
DATA_ADDRESS		NREF	Información sobre la dirección de un operando
	AREA	BYTE	(C) Área de memoria: L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE I: 16#81 Q: 16#82 M: 16#83 DB: 16#84, 85, 8A, 8B Infracciones de área en una variable editable directamente del tipo de datos DINT: 16#04
	DB_NUMBER	UINT	(D) Número del bloque de datos
	OFFSET	UDINT	(E) Dirección relativa del operando

Componente de estructura "ERROR_ID"

La tabla siguiente muestra los valores que pueden devolverse en el componente de estructura "ERROR_ID":

ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

	Explicación
LABEL: L #Field[#index]	// Acceso a la variable #Field[#index]
L 40.5	// Cargar valor 40.5
*R	// Multiplicar el valor de la variable por el valor 40.5.
T #TagOut	// Transferir el producto al operando #TagOut.
CALL GET_ERROR	// Se llama la instrucción "Consultar error localmente".
OUT := #Error	// En caso de error se guarda información detallada en el operando #Error.
T #Error.REACTION	// Se lee el componente de estructura #Error.REACTION.
L 1	// Cargar valor 1
==I	// El valor del componente de estructura se compara con el valor 1.
JC LABEL	// Si la comparación de ambos valores es positiva, la ejecución del programa retrocede a la etiqueta de salto LABEL.

Se ha producido un error al acceder a la variable "#Field[#index]". A pesar del error de lectura y acceso, el operando "#TagOut" devuelve el estado lógico "1" y la multiplicación se realiza con el valor "0.0". Si se produce este error se recomienda programar la instrucción "Consultar error localmente" tras la multiplicación para capturar el error. La información de error suministrada por la instrucción "Consultar error localmente" se evalúa con una comparación. Si el componente de estructura "#Error.REACTION" tiene el valor "1", se trata de un error de lectura/acceso y la ejecución del programa comienza de nuevo en la etiqueta de salto LABEL.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Consultar y eliminar errores en el código del programa (Página 265)

Tratar errores de ejecución del programa (Página 2932)

GET_ERR_ID: Consultar ID de error localmente

Descripción

La instrucción "Consultar ID de error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si al ejecutar el bloque el sistema notifica errores de ejecución desde que se ejecutó la instrucción por última vez, la ID de error del primer error ocurrido se guarda en la variable de la salida RET_VAL.

En la salida RET_VAL solo pueden especificarse operandos del tipo de datos "WORD". Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces la instrucción emite la ID de error del siguiente error ocurrido.

Nota

La salida RET_VAL solo se modifica si hay una información de error. Para volver a poner la salida a "0" tras tratar un error existen las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz de bloque.
- Ponga la variable de nuevo a "0" antes de llamar la instrucción.

La salida de la instrucción "Consultar ID de error localmente" se activa únicamente si existe una información de error. Si no se cumple esta condición, el procesamiento ulterior del programa no se ve afectado por la instrucción "Consultar ID de error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar ID de error localmente" activa el tratamiento local de errores dentro del bloque. Si en la lógica de un bloque se ha insertado la instrucción "Consultar ID de error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Sintaxis

Para la instrucción "Consultar ID de error localmente" se utiliza la siguiente sintaxis:

```
CALL GET_ERR_ID
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar ID de error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	WORD	I, Q, M, D, L	ID de error

Parámetro RET_VAL

La tabla siguiente muestra los valores que pueden depositarse en el parámetro RET_VAL:

RET_VAL* (hexadecimal)	RET_VAL* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

	Explicación
L #Field[#index]	// Acceso a la variable #Field[#index]
L 40.5	// Cargar valor 40.5
*R	// Multiplicar el valor de la variable por el valor 40.5.
T #TagOut	// Transferir el producto al operando #TagOut.
CALL GET_ERR_ID	// Se llama la instrucción "Consultar ID de error localmente".
RET_VAL := #TagID	// En caso de error se guarda una ID de error en el operando #TagID.
T #TagID	// Se lee la ID de error en el operando #TagID.
L 16#2522	// Cargar valor 16#2522
==I	// El valor de la ID de error se compara con el valor 16#2522.
CALL MOVE	// Se llama la instrucción "Copiar valor".
VARIANT	// Tipo de datos de la instrucción
IN := 100.0	// El valor 100.0 se copia en el operando #TagOut.
OUT := #TagOut	// Salida de la instrucción

Se ha producido un error al acceder a la variable "#Field[#index]". A pesar del error de lectura y acceso, el operando "#TagOut" devuelve el estado lógico "1" y la multiplicación se realiza con el valor "0.0". Si se produce este error se recomienda programar la instrucción "Consultar ID de error localmente" tras la multiplicación para capturar el error. La ID de error suministrada por la instrucción "Consultar ID de error localmente" se evalúa con una comparación. Si el operando "#TagID" devuelve la ID 16#2522, se trata de un error de lectura/acceso y el valor "100.0" se copia en la salida "#TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Consultar y eliminar errores en el código del programa (Página 265)

Tratar errores de ejecución del programa (Página 2932)

INIT_RD: Inicializar todos los datos remanentes

Descripción

La instrucción "Inicializar todos los datos remanentes" permite inicializar todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC al mismo tiempo. La instrucción solo puede ejecutarse dentro de un OB de arranque, puesto que su ejecución excedería el tiempo de ciclo de programa.

Sintaxis

Para la instrucción "Inicializar todos los datos remanentes" se utiliza la siguiente sintaxis:

```
CALL INIT_RD
REQ := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar todos los datos remanentes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Si la entrada REQ devuelve el estado lógico "1", se inicializan todos los datos remanentes.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B5	No puede ejecutarse la instrucción porque no ha sido programada dentro de un OB de arranque.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`CALL INIT_RD`	`// Se llama la instrucción "Inicializar todos los datos remanentes".`
`REQ := "Tag_REQ"`	`// Si el operando "Tag_REQ" devuelve el estado lógico "1", se inicializan todos los datos remanentes.`
`RET_VAL := "Tag_RET_VAL"`	`// Código de error`

Si el operando "Tag_REQ" devuelve el estado lógico "1", se ejecuta la instrucción. Se inicializan todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4013)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" detiene la ejecución del programa durante un periodo parametrizado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible parametrizar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve la información de error.

Sintaxis

Para la instrucción "Programar retardo" se utiliza la siguiente sintaxis:

```
CALL WAIT
WT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RUNTIME: Medir tiempo de ejecución

Descripción

La instrucción "Medir tiempo de ejecución" permite medir el tiempo de ejecución de todo el programa, bloques individuales o secuencias de comandos.

Si desea medir el tiempo de ejecución de todo el programa, efectúe la llamada de la instrucción "Medir tiempo de ejecución" en el OB1. El punto inicial para medir el tiempo de ejecución se determina con la primera llamada y la salida RET_VAL devuelve el tiempo de ejecución del programa tras la segunda llamada. El tiempo de ejecución medido incluye toda clase de procesos de la CPU que se hayan producido durante la ejecución del programa, como, p. ej., interrupciones a causa de eventos o comunicaciones de mayor prioridad. La instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU y escribe su valor en el parámetro de entrada/salida MEM. La instrucción calcula el tiempo de ejecución actual del programa según la frecuencia interna del contador, y la escribe en la salida RET_VAL.

Si desea medir el tiempo de ejecución de bloques o secuencias de comandos individuales, necesita tres segmentos diferentes. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" en un segmento individual dentro del programa. Al efectuar esta primera llamada de la instrucción se establece el punto inicial para medir el tiempo de ejecución. A continuación, llame el bloque de programa o la secuencia de comando que desee en el siguiente segmento. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" por segunda vez en otro segmento diferente y asigne al parámetro de entrada/salida MEM la misma memoria que en la primera llamada de la instrucción. En el tercer segmento, la instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU, calcula el tiempo de ejecución actual del bloque de programa o secuencia de comando según la frecuencia interna del contador y lo escribe en la salida RET_VAL.

La instrucción "Medición del tiempo de ejecución" utiliza un contador interno de alta frecuencia para calcular el tiempo. Cuando el contador se desborda, la instrucción devuelve valores <= 0.0. Estos valores de runtime deben ignorarse.

Nota

Puesto que el orden de las instrucciones de una secuencia de comandos se modifica mediante la compilación optimizada del programa, no es posible determinar el tiempo de ejecución de una secuencia de comandos con exactitud.

Sintaxis

Para la instrucción "Medir tiempo de ejecución" se utiliza la siguiente sintaxis:

```
CALL RUNTIME
RET_VAL := <Operando>
MEM := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Medir tiempo de ejecución":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	LREAL	I, Q, M, D, L	Devuelve el tiempo de ejecución medido en segundos
MEM	InOut	LREAL	M, D, L	Guarda el punto inicial de la medición del tiempo de ejecución

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción mediante el cálculo del tiempo de ejecución de un bloque de programa:

```
AWL                                  Explicación
CALL RUNTIME                         // Se llama la instrucción "Medir tiempo de ejecución".
   RET_VAL := "TagResult"            // Resultado intermedio
   MEM := "TagMemory"                // Guarda el punto inicial de la medición del tiempo de ejecución.

CALL "Best_before_date",             // El bloque "Best_before_date" se llama y se procesa.
"Best_before_date_DB"

CALL RUNTIME                         // Se llama la instrucción "Medir tiempo de ejecución" por
                                     segunda vez.
   RET_VAL := "TagResult"            // Resultado de la instrucción
   MEM := "TagMemory"                // Toma el punto inicial de la medición del tiempo de ejecución
                                     como base para calcular el tiempo de ejecución.
```

Con la primera llamada de la instrucción se determina el punto inicial para la medición del tiempo de ejecución, y se guarda como referencia en una memora intermedia del operando "TagMemory" para la segunda llamada de la instrucción.

A continuación se llama el bloque de programa FB1 "Best_before_date".

Una vez se ha procesado el bloque de programa FB1, la instrucción se ejecuta por segunda vez. La segunda llamada de la instrucción calcula el tiempo de ejecución del bloque de programa y escribe el resultado en el operando "TagResult" de la salida RET_VAL.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Temporizadores (Página 3907)

Operaciones lógicas con palabras

DECO: Descodificar

Descripción

La instrucción "Descodificar" activa un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor del parámetro IN y activa el bit del parámetro OUT cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor del parámetro IN es mayor que 31, se ejecuta una operación modulo 32.

Sintaxis

Para la instrucción "Descodificar" se utiliza la siguiente sintaxis:

```
CALL DECO
UINT <Tipo de datos>
IN := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	UINT	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

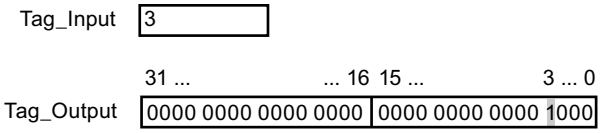
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL DECO	// Se llama la instrucción "Descodificar".
UINT WORD	// Tipos de datos de la instrucción
IN := "Tag_Input"	// Valor de entrada que predetermina el bit que debe activarse en el valor de salida.
OUT := "Tag_Output"	// Resultado

La figura siguiente muestra el funcionamiento de la instrucción con valores concretos:



La instrucción lee el número de bit "3" del valor del operando "Tag_Input" de la entrada y activa el tercer bit del valor del operando "Tag_Output" de la salida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ENCO: Codificar

Descripción

La instrucción "Codificar" lee el número del bit menos significativo que está activado en el valor de entrada y devolverlo en el parámetro OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor del parámetro IN y escribe su número de bit en el operando del parámetro OUT. Si el parámetro IN devuelve el valor DW#16#00000001 o DW#16#00000000, en el parámetro OUT se deposita el valor "0".

Sintaxis

Para la instrucción "Codificar" se utiliza la siguiente sintaxis:

```
CALL ENCO
<Tipo de datos>
IN := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

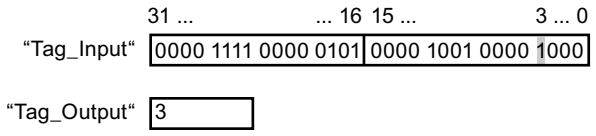
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL ENCO	// Se llama la instrucción "Codificar".
WORD	// Tipo de datos de la instrucción
IN := "Tag_Input"	// Valor de entrada cuyo bit activado menos significativo se lee.
OUT := "Tag_Output"	// Número del bit menos significativo que está activado en el valor de entrada

La figura siguiente muestra el funcionamiento de la instrucción con valores concretos:



La instrucción selecciona el bit menos significativo que está activado en la variable "Tag_Input" y escribe su bit "3" en la variable "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite elegir, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copiar su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G tiene el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

Para poder procesar la instrucción, las variables de todos los parámetros deben ser del mismo tipo de datos.

Sintaxis

Para la instrucción "Seleccionar" se utiliza la siguiente sintaxis:

```
CALL SEL
<Tipo de datos>
G := <Operando>
IN0 := <Operando>
IN1 := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
G	Input	BOOL	I, Q, M, D, L o constante	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, LTOD, DATE, DT, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Secuencias de bits, enteros, números en coma flotante, TIME, CHAR, TOD, DATE, DT, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, TIME, CHAR, TOD, DATE, DT, LDT	I, Q, M, D, L, P	Resultado

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL SEL                     // Se llama la instrucción "Seleccionar".
DWORD                        // Tipo de datos de la instrucción
G := "Tag_Input_G"           // Estado lógico del interruptor
IN0 := "Tag_Input0"          // Primer valor de entrada
IN1 := "Tag_Input1"          // Segundo valor de entrada
OUT := "Tag_Output"          // Valor de la entrada seleccionada
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor
G	Tag_Input_G	1
IN0	Tag_Input0	W#16#0000
IN1	Tag_Input1	W#16#FFFF
OUT	Tag_Output	W#16#FFFF

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Otras instrucciones

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" ocupa los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de este es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

```
CALL DRUM, "DB instancia"
RESET := <Operando>
JOG := <Operando>
DRUM_EN := <Operando>
LST_STEP := <Operando>
EVENT1 - 16 := <Operando>
OUT1 - 16 := <Operando>
Q := <Operando>
OUT_WORD := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L o constante	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L o constante	Bit de evento (i); El estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L o constante	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Primer paso predeterminado del circuito secuencial sucesivo (1 a 16)
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P o constante	Tiempo de ejecución restante para el paso actual
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	hora de sistema de la llamada anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente, la instrucción pasa del paso 1 al paso 2. Los bits de salida (OUT1 a OUT16) y la palabra de salida (OUT_WORD) se activan según la máscara configurada para el paso 2 y los valores del parámetro OUT_VAL.

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
CALL DRUM, "DRUM_DB"	// Se llama la instrucción "Materializar circuito secuencial sucesivo" y se crea el bloque de datos instancia "DRUM_DB".
RESET := "Tag_Reset"	// Entrada de inicialización
JOG := "Tag_Input_Jog"	// Si se produce un flanco de señal ascendente, la instrucción pasa al paso siguiente.
DRUM_EN := "Tag_Input_DrumEN"	// El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.

AWL	Explicación
LST_STEP := "Tag_Number_LastStep"	// Número del último paso programado
EVENT1 := "MyTag_Event_1"	// Bit de evento 1
EVENT2 := "MyTag_Event_2"	// Bit de evento 2
EVENT3 := "MyTag_Event_3"	// Bit de evento 3
EVENT4 := "MyTag_Event_4"	// Bit de evento 4
EVENT5 := "MyTag_Event_5"	// Bit de evento 5
EVENT6 := "MyTag_Event_6"	// Bit de evento 6
EVENT7 := "MyTag_Event_7"	// Bit de evento 7
EVENT8 := "MyTag_Event_8"	// Bit de evento 8
EVENT9 := "MyTag_Event_9"	// Bit de evento 9
EVENT10 := "MyTag_Event_10"	// Bit de evento 10
EVENT11 := "MyTag_Event_11"	// Bit de evento 11
EVENT12 := "MyTag_Event_12"	// Bit de evento 12
EVENT13 := "MyTag_Event_13"	// Bit de evento 13
EVENT14 := "MyTag_Event_14"	// Bit de evento 14
EVENT15 := "MyTag_Event_15"	// Bit de evento 15
EVENT16 := "MyTag_Event_16"	// Bit de evento 16
OUT1 := "MyTag_Output_1"	// Bit de salida 1
OUT2 := "MyTag_Output_2"	// Bit de salida 2
OUT3 := "MyTag_Output_3"	// Bit de salida 3
OUT4 := "MyTag_Output_4"	// Bit de salida 4
OUT5 := "MyTag_Output_5"	// Bit de salida 5
OUT6 := "MyTag_Output_6"	// Bit de salida 6
OUT7 := "MyTag_Output_7"	// Bit de salida 7
OUT8 := "MyTag_Output_8"	// Bit de salida 8
OUT9 := "MyTag_Output_9"	// Bit de salida 9
OUT10 := "MyTag_Output_10"	// Bit de salida 10
OUT11 := "MyTag_Output_11"	// Bit de salida 11
OUT12 := "MyTag_Output_12"	// Bit de salida 12
OUT13 := "MyTag_Output_13"	// Bit de salida 13
OUT14 := "MyTag_Output_14"	// Bit de salida 14
OUT15 := "MyTag_Output_15"	// Bit de salida 15
OUT16 := "MyTag_Output_16"	// Bit de salida 16
Q := "Tag_Output_Q"	// El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD := "Tag_OutputWord"	// Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida
ERR_CODE := "Tag_ErrorCode"	// Información de error

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para asignar valores a los parámetros de entrada:

Parámetro	Operando	Dirección	Valor
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DRUM_DB" de la instrucción:

Parámetro	Dirección	Valor
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE

Parámetro	Dirección	Valor
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

Los parámetros de salida tienen los valores siguientes antes de ejecutarse la instrucción:

Parámetro	Operando	Dirección	Valor
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE

Parámetro	Operando	Dirección	Valor
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Dirección	Valor
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Los siguientes valores se modifican en el bloque de datos instancia DRUM_DB de la instrucción después de su ejecución:

Parámetro	Dirección	Valor
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene los efectos siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET (solo si ET < PT), OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1".
  - Los parámetros ET, OA y CA se ponen a "0".
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET (solo si ET < PT), OA, CA y CMD_HIS se ponen a "0".

- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control discreto" se utiliza la siguiente sintaxis:

```
CALL DCAT, "DB instancia"
CMD := <Operando>
O_FB := <Operando>
C_FB := <Operando>
Q := <Operando>
OA := <Operando>
CA := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
CMD_HIS	Static	BOOL	D, L o constante	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
CALL DCAT, "DCAT_DB"	// Se llama la instrucción "Alarma de tiempo para control discreto" y se crea el bloque de datos instancia "DCAT_DB".
CMD := "Tag_Input_CMD"	// Abrir o cerrar el dispositivo
O_FB := "Tag_Input_O_FB"	// Realimentación al abrir
C_FB := "Tag_Input_C_FB"	// Realimentación al cerrar
Q := "Tag_Output_Q"	// Indica el estado del parámetro CMD

```
AWL                          Explicación
OA := "Tag_Output_OA"        // Salida de alarma al abrir
CA := "Tag_Output_CA"        // Salida de alarma al cerrar
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" permite acumular el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se dispara la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma

Parámetros de entrada								Parámetros de salida								
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado
Leyenda:																
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB															
PT	PT se pone al mismo valor que ET															
X	No aplicable															
< PT	ET < PT															
>= PT	ET >= PT															
Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE																

Sintaxis

Para la instrucción "Alarma de tiempo para control del motor" se utiliza la siguiente sintaxis:

```
CALL MCAT, "DB instancia"
O_CMD := <Operando>
C_CMD := <Operando>
S_CMD := <Operando>
O_FB := <Operando>
C_FB := <Operando>
OO := <Operando>
CO := <Operando>
OA := <Operando>
CA := <Operando>
Q := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
O_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
O_HIS	Static	BOOL	D, L o constante	Bit de evolución "Abrir"
C_HIS	Static	BOOL	D, L o constante	Bit de evolución "Cerrar"

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
CALL MCAT, "MCAT_DB"	// Se llama la instrucción "Alarma de tiempo para control del motor" y se crea el bloque de datos instancia "MCAT_DB".
O_CMD := "Tag_Input_O_CMD"	// Entrada de la orden "Abrir"
C_CMD := "Tag_Input_C_CMD"	// Entrada de la orden "Cerrar"
S_CMD := "Tag_Input_S_CMD"	// Entrada de la orden "Parar"
O_FB := "Tag_Input_O_FB"	// Realimentación al abrir
C_FB := "Tag_Input_C_FB"	// Realimentación al cerrar
OO := "Tag_OutputOpen"	// Salida "Abrir"
CO := "Tag_OutputClosed"	// Salida "Cerrar"
OA := "Tag_Output_OA"	// Salida de alarma al abrir
CA := "Tag_Output_CA"	// Salida de alarma al cerrar
Q := "Tag_Output_Q"	// Información de error

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del

parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una correspondencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Comparar bits de entrada con bits de una máscara" se utiliza la siguiente sintaxis:

```
CALL IMC, "DB instancia"
IN_BIT0 - 15 := <Operando>
CMP_STEP := <Operando>
OUT := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente se comparan todos los 16 bits de entrada con la máscara para el paso 2. El estado lógico del parámetro OUT se pone a TRUE porque los bits de entrada concuerdan con la máscara para el paso 2.

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

```
AWL                          Explicación
CALL IMC, "IMC_DB"           // Se llama la instrucción "Comparar bits de entrada
                             con bits de una máscara" y se crea el bloque de datos
                             instancia "IMC_DB".
IN_BIT0 := "Tag_Input_BIT0"  // Bit de entrada 0
IN_BIT1 := "Tag_Input_BIT1"  // Bit de entrada 1
IN_BIT2 := "Tag_Input_BIT2"  // Bit de entrada 2
IN_BIT3 := "Tag_Input_BIT3"  // Bit de entrada 3
IN_BIT4 := "Tag_Input_BIT4"  // Bit de entrada 4
IN_BIT5 := "Tag_Input_BIT5"  // Bit de entrada 5
IN_BIT6 := "Tag_Input_BIT6"  // Bit de entrada 6
IN_BIT7 := "Tag_Input_BIT7"  // Bit de entrada 7
IN_BIT8 := "Tag_Input_BIT8"  // Bit de entrada 8
IN_BIT9 := "Tag_Input_BIT9"  // Bit de entrada 9
```

```
AWL                              Explicación
IN_BIT10 := "Tag_Input_BIT10"  // Bit de entrada 10
IN_BIT11 := "Tag_Input_BIT11"  // Bit de entrada 11
IN_BIT12 := "Tag_Input_BIT12"  // Bit de entrada 12
IN_BIT13 := "Tag_Input_BIT13"  // Bit de entrada 13
IN_BIT14 := "Tag_Input_BIT14"  // Bit de entrada 14
IN_BIT15 := "Tag_Input_BIT15"  // Bit de entrada 15
CMP_STEP := "Tag_CMP_STEP"     // Número de paso de la máscara
OUT := "Tag_Output"            // Indica si se ha encontrado una correspondencia.
ERR_CODE := "Tag_ErrorCode"    // Información de error
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
CMP_STEP	Tag_CMP_STEP	B#16#02
OUT	Tag_Output	FALSE
ERR_CODE	Tag_ErrorCode	W#16#0000

En el bloque de datos de instancia IMC_DB de la instrucción están almacenados los siguientes valores para la máscara del paso 2:

Parámetro	Dirección	Valor
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE

Parámetro	Dirección	Valor
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output	TRUE
ERR_CODE	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último

paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Barrido de matriz" se utiliza la siguiente sintaxis:

```
CALL SMC, "DB instancia"
IN_BIT0 - 15 := <Operando>
OUT := <Operando>
OUT_STEP := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
LAST	Static	BYTE	I, Q, M, D, L, P o constante	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente, todos los 16 bits de entrada se comparan con la máscara para los pasos 0 a 5 hasta que se encuentre una correspondencia. Se consultan únicamente las máscaras para los pasos 0 a 2, ya que la máscara para el paso 2 se corresponde con el bit de entrada.

Nota

Los parámetros estáticos pueden inicializarse en el bloque.

```
AWL                          Explicación
CALL SMC, "SMC_DB"           // Se llama la instrucción "Barrido de matriz" y se
                             crea el bloque de datos instancia "SMC_DB".
IN_BIT0 := "Tag_Input_BIT0"  // Bit de entrada 0
IN_BIT1 := "Tag_Input_BIT1"  // Bit de entrada 1
IN_BIT2 := "Tag_Input_BIT2"  // Bit de entrada 2
IN_BIT3 := "Tag_Input_BIT3"  // Bit de entrada 3
IN_BIT4 := "Tag_Input_BIT4"  // Bit de entrada 4
IN_BIT5 := "Tag_Input_BIT5"  // Bit de entrada 5
IN_BIT6 := "Tag_Input_BIT6"  // Bit de entrada 6
IN_BIT7 := "Tag_Input_BIT7"  // Bit de entrada 7
IN_BIT8 := "Tag_Input_BIT8"  // Bit de entrada 8
IN_BIT9 := "Tag_Input_BIT9"  // Bit de entrada 9
IN_BIT10 := "Tag_Input_BIT10" // Bit de entrada 10
IN_BIT11 := "Tag_Input_BIT11" // Bit de entrada 11
IN_BIT12 := "Tag_Input_BIT12" // Bit de entrada 12
IN_BIT13 := "Tag_Input_BIT13" // Bit de entrada 13
IN_BIT14 := "Tag_Input_BIT14" // Bit de entrada 14
IN_BIT15 := "Tag_Input_BIT15" // Bit de entrada 15
OUT := "Tag_Output"          // Indica si se ha encontrado una correspondencia.
OUT_STEP := "Tag_Output_STEP" // Contiene el número de paso con la máscara
                             correspondiente.
ERR_CODE := "Tag_ErrorCode"  // Información de error
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
OUT	Tag_Output	FALSE
OUT_STEP	Tag_Output_STEP	B#16#00
ERR_CODE	Tag_ErrorCode	W#16#0000

En el bloque de datos de instancia SMC_DB de la instrucción están almacenados los siguientes valores para la máscara del paso 2:

Parámetro	Dirección	Valor
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE

Parámetro	Dirección	Valor
CMP_VAL [2,15]	DBX13.7	TRUE
LAST	DB84	B#16#05

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output	TRUE
OUT_STEP'	Tag_Output_STEP	B#16#02
ERR_CODE	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor de la ganancia en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse como supresor de perturbaciones o filtro de paso bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

```
CALL LEAD_LAG, "DB
instancia"
IN := <Operando>
SAMPLE_T := <Operando>
OUT := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
`CALL LEAD_LAG, "LEAD_LAG_DB"`	`// Se llama la instrucción "Algoritmo Lead y Lag" y se crea el bloque de datos instancia "LEAD_LAG_DB".`

AWL	Explicación
IN := "Tag_Input"	// Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T := "Tag_Input_SAMPE_T"	// Tiempo de muestreo
OUT := "Tag_Output_Result"	// Resultado de la instrucción
ERR_CODE := "Tag_ErrorCode"	// Información de error

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

En el bloque de datos instancia "LEAD_LAG_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SEG: Generar patrón de bits para el display de 7 segmentos

Descripción

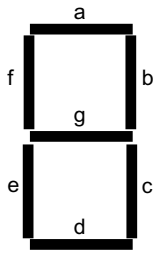
La instrucción "Generar patrón de bits para el display de 7 segmentos" convierte cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en cuatro códigos equivalentes para un display de 7 segmentos y los escribe en la palabra doble de la salida (OUT).

La instrucción "Generar patrón de bits para el display de 7 segmentos" no detecta condiciones de error.

Entre los valores de entrada hexadecimales y los patrones de bits de la salida existe la relación siguiente:

Cifra	-gfedcba	Display
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6
0111	00000111	7
1000	01111111	8
1001	01100111	9
1010	01110111	A
1011	01111100	B
1100	00111001	C
1101	01011110	D
1110	01111001	E
1111	01110001	F

Display de siete segmentos



Sintaxis

Para la instrucción "Generar patrón de bits para el display de 7 segmentos" se utiliza la siguiente sintaxis:

```
CALL SEG
IN := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	IN	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales.
OUT	OUT	DWORD	I, Q, M, D, L, P	Patrón de bits del destino con cuatro bytes.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL SEG	// Se llama la instrucción "Generar patrón de bits para el display de 7 segmentos".
IN := "Tag_Input"	// Palabra de origen
OUT := "Tag_Output"	// Patrón de bits para el display de 7 segmentos

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16#065B4F66	00000110 01011011 01001111 01100110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

Sintaxis

Para la instrucción "Generar complemento a diez" se utiliza la siguiente sintaxis:

```
CALL BCDCPL
IN := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL BCDCPL	// Se llama la instrucción "Generar complemento a diez".
IN := "Tag_Input"	// Número BCD de 7 dígitos
ERR_CODE := "Tag_Output"	// Complemento a diez del número BCD indicado

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" cuenta el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

Sintaxis

Para la instrucción "Contar número de bits activados" se utiliza la siguiente sintaxis:

```
CALL BITSUM
IN := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan.
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL BITSUM	// Se llama la instrucción "Contar número de bits activados".
IN := "Tag_Input"	// Operando cuyos bits activados se cuentan.
RET_VAL := "Tag_Output"	// Número de bits activados del operando "Tag_Input".

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones básicas

Operaciones lógicas con bits

A: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" consulta si el estado lógico de un operando binario es "1" y combinar el resultado lógicamente con Y con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es idéntico al estado lógico del operando consultado.

Una vez ejecutada la instrucción, el RLO es "1" si los dos estados lógicos combinados también son "1". Si uno de los estados lógicos combinados es "0", el RLO será "0" tras ejecutar la instrucción.

Si se aplica la instrucción varias veces consecutivas, todos los resultados de las consultas deben ser "1" para que el resultado lógico común también sea "1".

Con la instrucción "Operación lógica Y" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica Y" se utiliza la siguiente sintaxis:

```
A <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL		Esquema de conexiones de relé	
		Barra de alimentación	
A	“Tag_Input_1“	Estado lógico 1	Contacto NA
A	“Tag_Input_2“	Estado lógico 1	Contacto NA
=	“Tag_Output“	Estado lógico 1	Bobina
Indica un interruptor cerrado.			

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

AN: Operación lógica Y negada

Descripción

La instrucción "Operación lógica Y negada" consulta si el estado lógico de un operando binario es "0" y combinar el resultado lógicamente con Y con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es el estado lógico negado del operando consultado. Si el estado lógico del operando es "0", el resultado de la consulta es "1". Si el estado lógico del operando es "1", el resultado de la consulta es "0".

Una vez ejecutada la instrucción, el RLO es "1" si el resultado de la consulta y el RLO antes de ejecutar la instrucción devuelven el estado lógico "1".

Una vez ejecutada la instrucción, el RLO es "0" si el resultado de la consulta o el RLO antes de ejecutar la instrucción devuelven el estado lógico "0".

Con la instrucción "Operación lógica Y negada" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica Y negada" se utiliza la siguiente sintaxis:

```
AN <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y negada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL		Esquema de conexiones de relé	
		Barra de alimentación	
A	“Tag_Input_1“	Estado lógico 0	Contacto NA
AN	“Tag_Input_2“	Estado lógico 1	Contacto NC
=	“Tag_Output“	Estado lógico 0	Bobina

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)

O: Operación lógica O

Descripción

La instrucción "Operación lógica O" consulta si el estado lógico de un operando binario es "1" y combinar el resultado lógicamente con O con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es idéntico al estado lógico del operando consultado.

Una vez ejecutada la instrucción, el RLO es "1" si uno de los estados lógicos combinados también es "1". Si los dos estados lógicos combinados son "0", el RLO será "0" tras ejecutar la instrucción.

Si se aplica la instrucción varias veces consecutivas, basta con que uno de los resultados de las consultas sea "1" para que el resultado lógico común también sea "1".

Con la instrucción "Operación lógica O" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O" se utiliza la siguiente sintaxis:

```
O <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL	Esquema de conexiones de relé
	Barra de alimentación
O “Tag_Input_1“	Estaco lógico 1 contacto NA (“Tag_Input_1“) / Estado lógico 0 contacto NA (“Tag_Input_2“)
O “Tag_Input_2“	
= “Tag_Output“	Estado lógico 1 / Bobina
	Indica un interruptor cerrado.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ON: Operación lógica O negada

Descripción

La instrucción "Operación lógica O negada" consulta si el estado lógico de un operando binario es "0" y combinar el resultado lógicamente con O con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es el estado lógico negado del operando consultado. Si el estado lógico del operando es "0", el resultado de la consulta es "1". Si el estado lógico del operando es "1", el resultado de la consulta es "0".

Una vez ejecutada la instrucción, el RLO es "1" si el resultado de la consulta o el RLO antes de ejecutar la instrucción devuelven el estado lógico "1". Si los dos estados lógicos combinados son "0", el RLO también será "0" tras ejecutar la instrucción.

Si se aplica la instrucción varias veces consecutivas, basta con que uno de los resultados de las consultas sea "1" para que el resultado lógico común también sea "1".

Con la instrucción "Operación lógica O negada" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O negada" se utiliza la siguiente sintaxis:

```
ON <Operando>
```

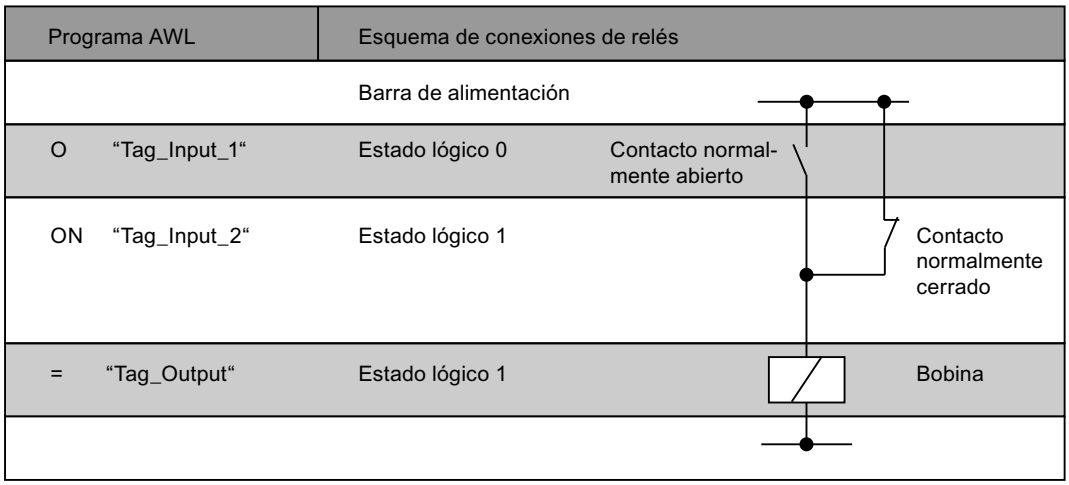
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O negada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

X: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" consulta si el estado lógico de un operando binario es "1" y combinar el resultado lógicamente con O-exclusiva con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es idéntico al estado lógico del operando consultado. Una vez ejecutada la instrucción, el RLO es "1" si los estados lógicos combinados entre sí son distintos. El RLO es "0" si ambos estados lógicos son idénticos.

Si, p. ej., el RLO antes de ejecutar la instrucción es "1" y el operando consultado devuelve "1", el RLO se pone a "0". Si el RLO es "1" y el operando devuelve el estado lógico "0", el RLO tras ejecutar la instrucción se pondrá a "1".

Si se aplica la instrucción varias veces consecutivas, basta con que uno de los resultados de las consultas sea "1" para que el RLO común también sea "1". Si todos los resultados son "1" o "0", el RLO se pone a "0".

Con la instrucción "Operación lógica O-exclusiva" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva" se utiliza la siguiente sintaxis:

```
X <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL	Esquema de conexiones de relé
	Barra de alimentación
X “Tag_Input_1“	Contacto
X “Tag_Input_2“	Contacto
= “Tag_Output“	Bobina

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

XN: Operación lógica O-exclusiva negada

Descripción

La instrucción "Operación lógica O-exclusiva negada" consulta si el estado lógico de un operando binario es "0" y combinar el resultado lógicamente con O con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es el estado lógico negado del operando consultado. Una vez ejecutada la instrucción, el RLO es "1" si los estados lógicos combinados entre sí son distintos. El RLO es "0" si ambos estados lógicos son idénticos.

Si, p. ej., el estado lógico del operando consultado es "0", el resultado de la consulta es "1". La combinación lógica con O-exclusiva del resultado de la consulta y un RLO "1" devuelve el estado lógico "0". La combinación con un RLO "0" devuelve el estado lógico "1".

Con la instrucción "Operación lógica O-exclusiva negada" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva negada" se utiliza la siguiente sintaxis:

```
XN <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva negada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL		Esquema de conexiones de relé
		Barra de alimentación
X	“Tag_Input_1“	Contacto
XN	“Tag_Input_2“	Contacto
=	“Tag_Output“	Bobina

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

O: Operación lógica O de funciones Y

Descripción

La instrucción "Operación lógica O de funciones Y" ejecuta varias operaciones lógicas Y antes de una operación lógica O. El resultado global de las operaciones lógicas Y se guarda en una memoria intermedia y se combina lógicamente con O con las demás consultas de señales.

Sintaxis

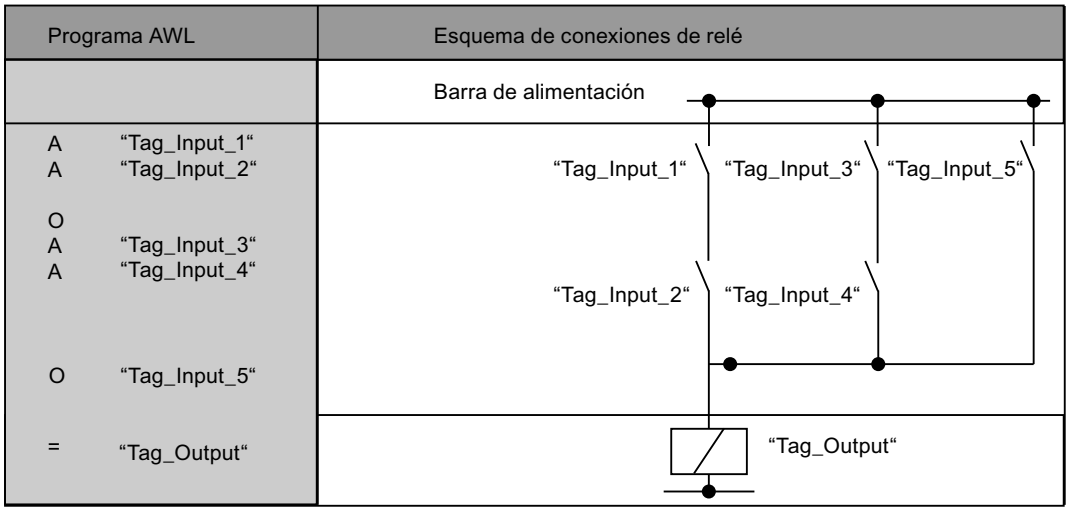
Para la instrucción "Operación lógica O de funciones Y" se utiliza la siguiente sintaxis:

O

La sintaxis de esta instrucción no requiere paréntesis.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

A(: Operación lógica Y con abrir paréntesis

Descripción

La instrucción "Operación lógica Y con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica Y.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR. El RLO guardado se combina lógicamente con Y con el resultado lógico de toda la expresión entre paréntesis. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la función Y.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

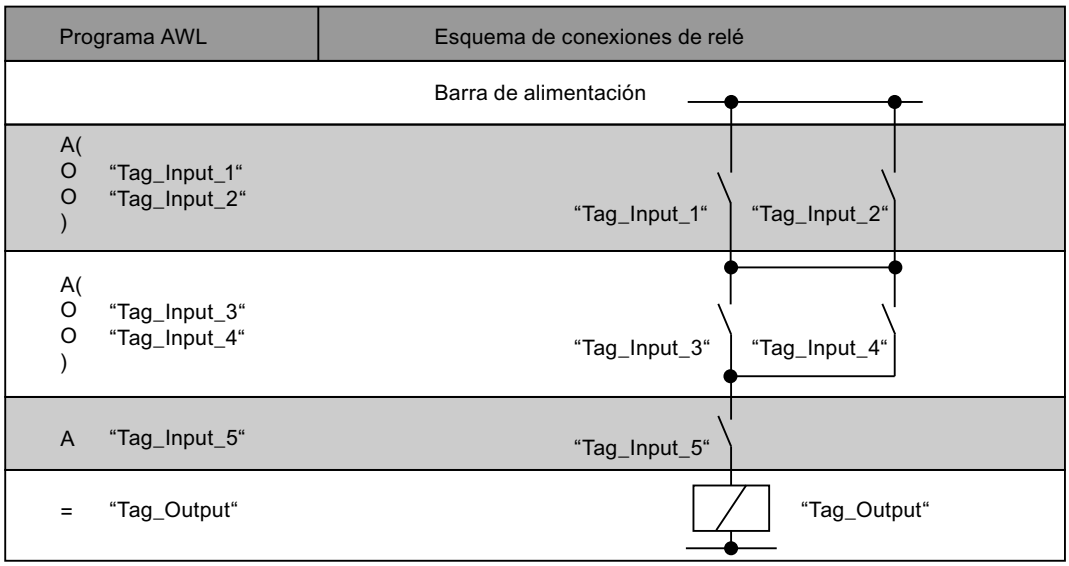
Sintaxis

Para la instrucción "Operación lógica Y con abrir paréntesis" se utiliza la siguiente sintaxis:

```
A(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

AN(: Operación lógica Y negada con abrir paréntesis

Descripción

La instrucción "Operación lógica Y negada con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica Y y negar el resultado global de la expresión entre paréntesis.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO de toda la expresión entre paréntesis se niega y se combina lógicamente con Y con el RLO guardado. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la instrucción Y.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica Y negada con abrir paréntesis" se utiliza la siguiente sintaxis:

```
AN(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
AN(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con Y el RLO de la expresión entre paréntesis y negar el resultado global.
O "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con O con el RLO actual.
O "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con O con el RLO actual.
)	// Fin de la expresión entre paréntesis

AWL	Explicación
AN "Tag_Input_3"	// Consultar si el estado lógico del operando es "0" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

O(: Operación lógica O con abrir paréntesis

Descripción

La instrucción "Operación lógica O con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO guardado se combina lógicamente con O con el RLO de toda la expresión entre paréntesis. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la instrucción O.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O con abrir paréntesis" se utiliza la siguiente sintaxis:

```
O (
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
O(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O el RLO de la expresión entre paréntesis.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ON(: Operación lógica O negada con abrir paréntesis

Descripción

La instrucción "Operación lógica O negada con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O y negar el resultado global de la expresión entre paréntesis.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO de la expresión entre paréntesis se niega y se combina lógicamente con O con el RLO guardado. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la instrucción O.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O negada con abrir paréntesis" se utiliza la siguiente sintaxis:

```
ON(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
ON(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O el RLO de la expresión entre paréntesis y negar el resultado global.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

X(: Operación lógica O-exclusiva con abrir paréntesis

Descripción

La instrucción "Operación lógica O-exclusiva con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O-exclusiva.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR. Después de procesar las instrucciones que están en la expresión entre paréntesis,

el RLO guardado se combina lógicamente con O-exclusiva con el RLO de toda la expresión entre paréntesis. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la función O-exclusiva.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva con abrir paréntesis" se utiliza la siguiente sintaxis:

```
X(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
X(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O-exclusiva el RLO de la expresión entre paréntesis.
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
X "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

XN(: Operación lógica O-exclusiva negada con abrir paréntesis

Descripción

La instrucción "Operación lógica O-exclusiva negada con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O-exclusiva y negar el resultado global de la expresión entre paréntesis.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO de la expresión entre paréntesis se niega y se combina lógicamente con O-exclusiva con el RLO guardado. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la función O-exclusiva.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva negada con abrir paréntesis" se utiliza la siguiente sintaxis:

```
XN(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
XN(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O-exclusiva el RLO de la expresión entre paréntesis y negar el resultado global.
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
XN "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

): Cerrar paréntesis

Descripción

La instrucción "Cerrar paréntesis" determina el final de una expresión entre paréntesis. Tras ejecutar la instrucción, el resultado global de la expresión entre paréntesis se combina lógicamente con la instrucción que se ha utilizado para abrir la expresión entre paréntesis.

Para abrir una expresión entre paréntesis se utilizan las instrucciones siguientes:

- A(: Operación lógica Y con abrir paréntesis
- AN(: Operación lógica Y negada con abrir paréntesis
- O(: Operación lógica O con abrir paréntesis
- ON(: Operación lógica O negada con abrir paréntesis
- X(: Operación lógica O-exclusiva con abrir paréntesis
- XN(: Operación lógica O-exclusiva negada con abrir paréntesis

Después de la instrucción "Cerrar paréntesis", la CPU pone el resultado binario BR al estado lógico que tenía antes de la expresión entre paréntesis.

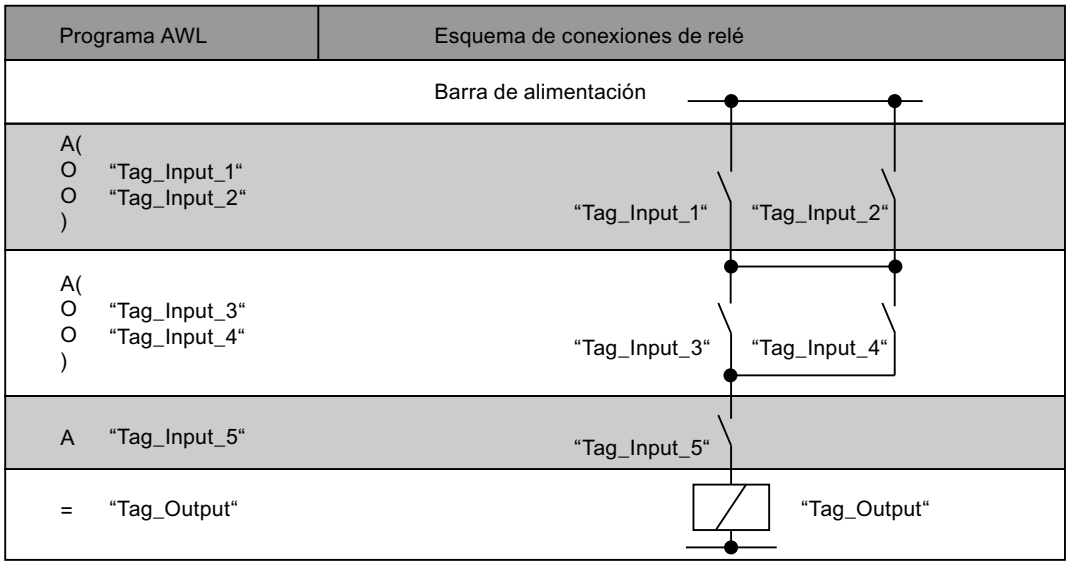
Sintaxis

Para la instrucción "Cerrar paréntesis" se utiliza la siguiente sintaxis:

```
)
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

=: Asignación

Descripción

La instrucción "Asignación" asigna a un operando concreto el estado lógico del resultado lógico (RLO) guardado en la CPU. Si el RLO tiene el estado lógico "1", se activa el operando. Si el estado lógico de la entrada es "0", el operando se pone a "0".

Sintaxis

Para la instrucción "Asignación" se utiliza la siguiente sintaxis:

```
= <Operando>
```

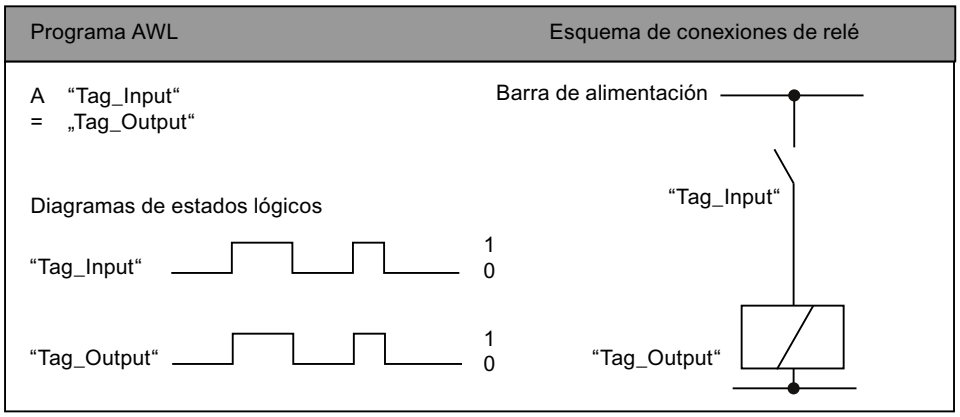
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO actual.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

R: Desactivar

Descripción

La instrucción "Desactivar" pone a "0" el estado lógico de un operando concreto.

La instrucción solo se ejecuta si el resultado lógico (RLO) actual es "1". Tras ejecutar la instrucción, el operando indicado se pone a "0". Si el RLO actual es "0", el estado lógico del operando indicado no cambia.

Sintaxis

Para la instrucción "Desactivar" se utiliza la siguiente sintaxis:

```
R <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se desactiva si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "Tag_Output"	// Poner el operando a "0" si el RLO = "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

S: Activar

Descripción

La instrucción "Activar" pone a "1" el estado lógico de un operando concreto.

La instrucción solo se ejecuta si el resultado lógico (RLO) actual es "1". Tras ejecutar la instrucción, el operando indicado se pone a "1". Si el RLO actual es "0", el estado lógico del operando indicado no cambia.

Sintaxis

Para la instrucción "Activar" se utiliza la siguiente sintaxis:

```
S <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se activa si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
S "Tag_Output"	// Poner el operando a "1" si el RLO = "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)

NOT: Invertir RLO

Descripción

La instrucción "Invertir RLO" invierte el estado lógico del resultado lógico (RLO). La instrucción "Invertir RLO" puede aplicarse en cualquier posición, incluso dentro de una operación lógica.

Sintaxis

Para la instrucción "Invertir RLO" se utiliza la siguiente sintaxis:

```
NOT
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_3"	// Consultar si el estado lógico del operando "Tag_Input_3" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
NOT	// Invertir el estado lógico del RLO
= "Tag_Output"	// Asignar el estado lógico del RLO al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Input_1	1	1	1	0	0	0	1	0
Tag_Input_2	0	1	1	0	1	1	0	1
Tag_Input_3	0	0	1	0	0	1	1	1
Tag_Output	1	1	0	1	1	1	1	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)

SET: Poner RLO a 1

Descripción

La instrucción "Poner RLO a 1" pone el resultado lógico (RLO) actual al estado lógico "1".

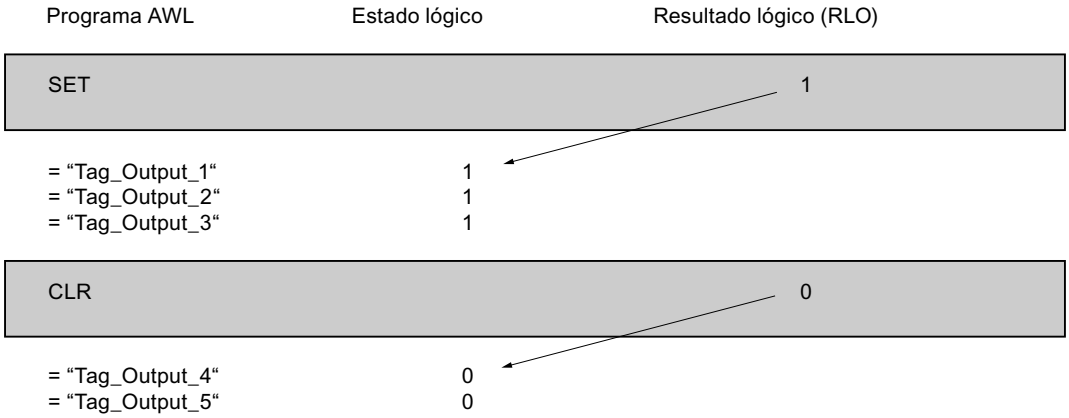
Sintaxis

Para la instrucción "Poner RLO a 1" se utiliza la siguiente sintaxis:

```
SET
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)

CLR: Poner RLO a 0

Descripción

La instrucción "Poner RLO a 0" pone el resultado lógico (RLO) actual al estado lógico "0".

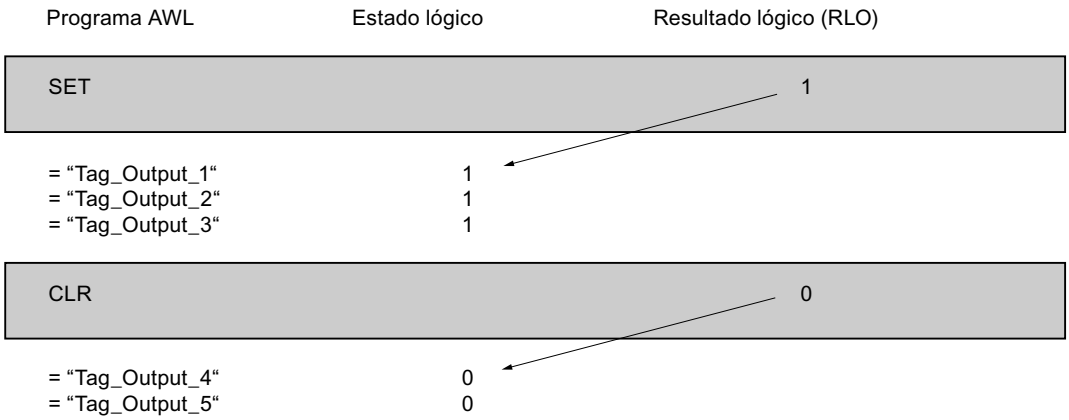
Sintaxis

Para la instrucción "Poner RLO a 0" se utiliza la siguiente sintaxis:

```
CLR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SAVE: Guardar RLO en bit BR

Descripción

La instrucción "Guardar RLO en bit BR" guarda el resultado lógico (RLO) en el resultado binario (BR). Durante la ejecución, la instrucción transfiere el estado lógico del resultado lógico actual al bit de estado BR. La instrucción trabaja independientemente de las condiciones y no influye en ningún otro bit de estado.

Nota

Tras aplicar la instrucción "Guardar RLO en bit BR", el bit BR puede ser modificado nuevamente por instrucciones posteriores del mismo bloque o de bloques de nivel inferior.

La instrucción "Guardar RLO en bit BR" también se utiliza para controlar el estado de ejecución de un bloque. Si, p. ej., la instrucción "Guardar RLO en bit BR" se aplica al final de un bloque, el bit BR se pone al estado lógico del resultado lógico actual del bloque.

El bit BR ayuda a la realización del mecanismo EN/ENO en llamadas de bloques. Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Guardar RLO en bit BR" se utiliza la siguiente sintaxis:

```
SAVE
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico del operando "Tag_Input" es "1".
SAVE	// Transferir el estado lógico del RLO al bit BR.
BEC	// Si el RLO = "1", finalizar el bloque.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

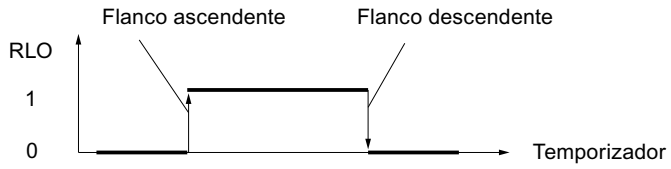
Consulta y activación de los bits de estado en AWL (Página 2560)

FN: Consultar flanco de señal descendente del RLO

Descripción

La instrucción "Consultar flanco de señal descendente del RLO" consulta un cambio del estado lógico del resultado lógico (RLO) de "1" a "0". La instrucción compara el estado lógico actual del resultado lógico (RLO) con el estado lógico de la consulta anterior, que está guardado en el operando de la instrucción. Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco descendente.

La figura siguiente muestra el cambio del RLO en caso de flanco descendente y ascendente:



El flanco descendente se consulta cada vez que se ejecuta la instrucción. Si la instrucción detecta un flanco descendente, el RLO se pone al estado lógico "1" tras evaluar el flanco. Si no se detecta ningún flanco descendente, el estado lógico del resultado lógico es "0".

Nota

El operando de la evaluación de flanco debe encontrarse en el área de memoria de marcas o en un bloque de datos.

Sintaxis

Para la instrucción "Consultar flanco de señal descendente del RLO" se utiliza la siguiente sintaxis:

```
FN <Operando>
```

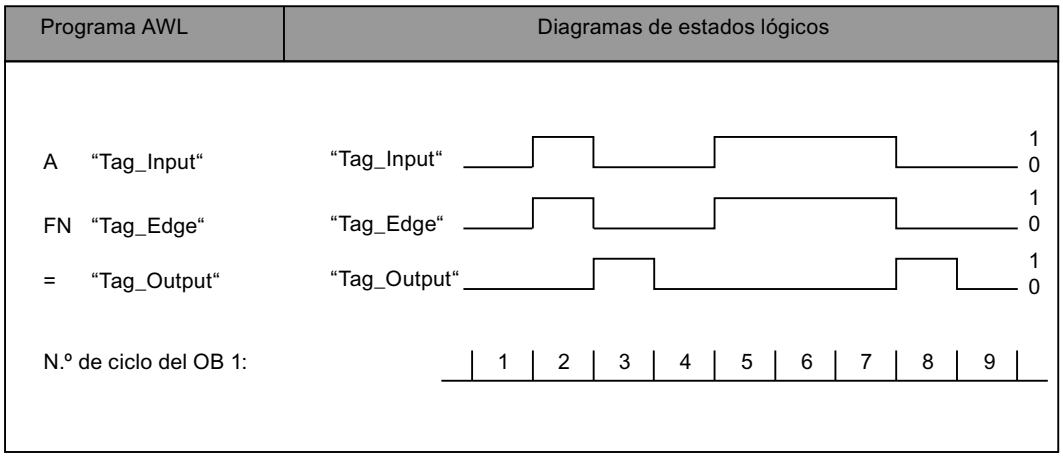
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Output	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el RLO de la última consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

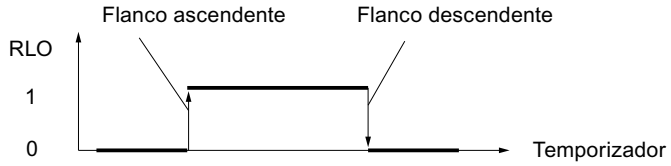
Consulta y activación de los bits de estado en AWL (Página 2560)

FP: Consultar flanco de señal ascendente del RLO

Descripción

La instrucción "Consultar flanco de señal ascendente del RLO" consulta un cambio del estado lógico del resultado lógico (RLO) de "0" a "1". La instrucción compara el estado lógico actual del resultado lógico (RLO) con el estado lógico de la consulta anterior, que está guardado en el operando de la instrucción. Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco ascendente.

La figura siguiente muestra el cambio del RLO en caso de flanco ascendente y descendente:



El flanco ascendente se consulta cada vez que se ejecuta la instrucción. Si la instrucción detecta un flanco ascendente, el RLO se pone al estado lógico "1" tras evaluar el flanco. Si no se detecta ningún flanco ascendente, el estado lógico del resultado lógico es "0".

Nota

El operando de la evaluación de flanco debe encontrarse en el área de memoria de marcas o en un bloque de datos.

Sintaxis

Para la instrucción "Consultar flanco de señal ascendente del RLO" se utiliza la siguiente sintaxis:

```
FP <operando>
```

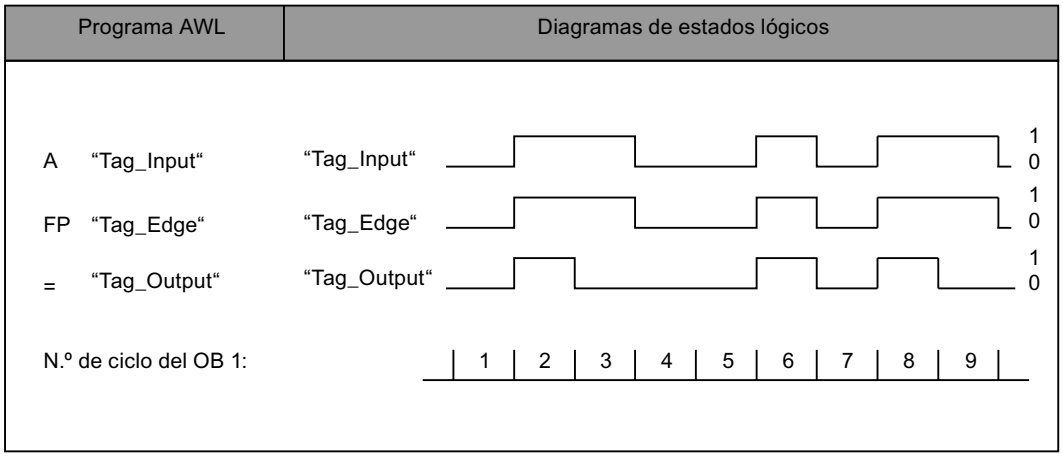
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Output	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el RLO de la última consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Temporizadores

FR: Habilitar temporizador

Descripción

La instrucción "Habilitar temporizador" provoca el rearranque de un temporizador. La instrucción se ejecuta con un flanco de señal ascendente y desactiva la marca de flancos interna para arrancar un temporizador. Al procesar una instrucción para arrancar el temporizador con un resultado lógico "1", el temporizador arranca de nuevo, aunque no haya ningún flanco de señal ascendente antes de la instrucción de arranque. Si el resultado lógico es "0" antes de una instrucción de arranque, la desactivación de la marca de flancos interna no influye en el temporizador.

Un cambio del resultado lógico a "0" antes de la instrucción "Habilitar temporizador" no tiene repercusión una vez rearrancado el temporizador.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Habilitar temporizador" se utiliza la siguiente sintaxis:

```
FR <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Habilitar temporizador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que se habilita

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_EnableInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyTimer"	// Habilitar temporizador con un flanco de señal ascendente.
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor del temporizador (Página 4093)

L: Cargar valor del temporizador

Descripción

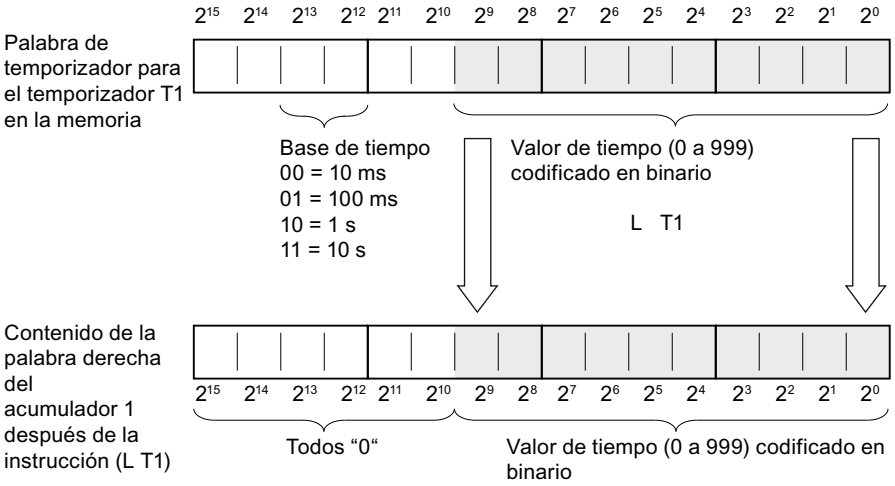
La instrucción "Cargar valor del temporizador" carga el valor de un temporizador concreto en el acumulador 1. El valor de tiempo en código binario se carga sin base de tiempo. En lugar de la base de tiempo se escriben ceros en el acumulador 1.

Después de la carga, el valor que está en el acumulador 1 equivale a un número positivo del tipo de datos INT.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

La figura siguiente muestra, a modo de ejemplo, cómo se carga en valor del temporizador en el acumulador:



Sintaxis

Para la instrucción "Cargar valor del temporizador" se utiliza la siguiente sintaxis:

```
L <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor del temporizador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador cuyo valor actual se carga

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_EnableInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyTimer"	// Habilitar temporizador con un flanco ascendente.
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LC: Cargar valor del temporizador codificado en BCD

Descripción

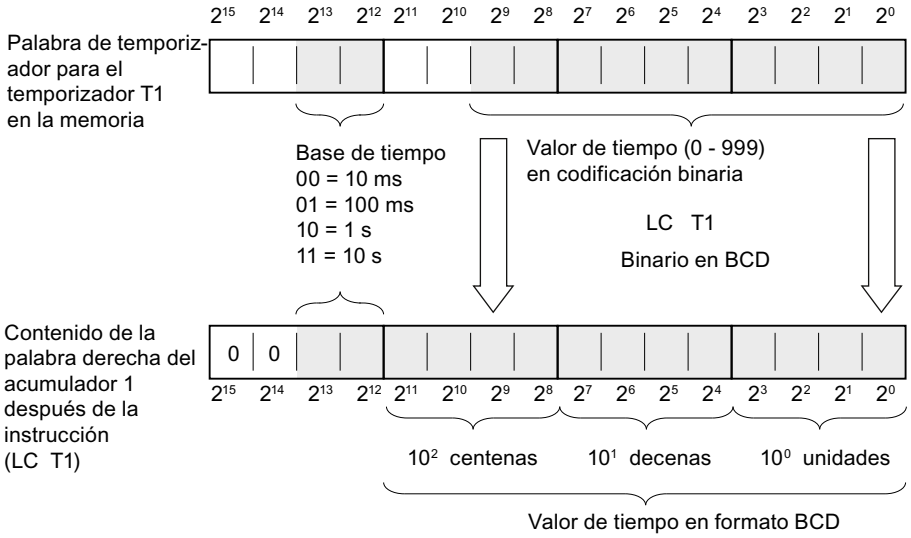
La instrucción "Cargar valor del temporizador codificado en BCD" carga el valor de un temporizador concreto en código binario en el acumulador 1 en formato BCD. Durante la carga también se transfiere la base de tiempo al acumulador 1.

Después de la carga, el valor que está en el acumulador 1 equivale a una duración en formato S5TIME. La palabra izquierda del acumulador 1 se rellena con ceros.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

La figura siguiente muestra cómo se carga en valor del temporizador en el acumulador:



Sintaxis

Para la instrucción "Cargar valor del temporizador codificado en BCD" se utiliza la siguiente sintaxis:

```
LC <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor del temporizador codificado en BCD":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador cuyo valor actual se carga codificado en BCD.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_EnableInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyTimer"	// Habilitar temporizador con un flanco ascendente.

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor del temporizador (Página 4093)

R: Inicializar temporizador

Descripción

La instrucción "Inicializar temporizador" pone a "0" un temporizador concreto. La instrucción se ejecuta si el resultado lógico (RLO) actual es "1". Mientras el RLO "1" esté pendiente antes de la instrucción, el temporizador indicado dará el valor "0". La instrucción también pone a "0" el valor del temporizador y la base de tiempo de la duración programada.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Nota

La instrucción "Inicializar temporizador" no desactiva la marca de flancos interna. Para desactivar la marca de flancos interna debe programarse la instrucción "Habilitar temporizador" o ejecutarse una instrucción para arrancar el temporizador con el estado lógico "0".

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Inicializar temporizador" se utiliza la siguiente sintaxis:

```
R <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar temporizador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que se inicializa

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor del temporizador (Página 4093)

SP: Arrancar temporizador como impulso

Descripción

La instrucción "Arrancar temporizador como impulso" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1, mientras el resultado lógico (RLO) antes de la instrucción siga siendo "1". Si el RLO cambia a "0" antes de finalizar el temporizador, este se detiene. Las consultas de si el estado del temporizador es "1" dan como resultado "1" mientras el temporizador está en marcha.

Es posible rearrancar el temporizador con un flanco de señal ascendente antes de la instrucción "Arrancar temporizador como impulso" o con la ayuda de la instrucción "Habilitar temporizador". Para ello, es imprescindible que el temporizador no esté desactivado.

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como impulso", el valor del temporizador se reduce de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido y el RLO de la operación lógica precedente devuelve el valor "0", el bit de estado OV se pone a "1". No se inicia ningún tratamiento de errores síncrono. Si el acumulador 1 tiene un valor BCD válido, el bit de estado OV se pone a "0".

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como impulso" se utiliza la siguiente sintaxis:

```
SP <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como impulso.

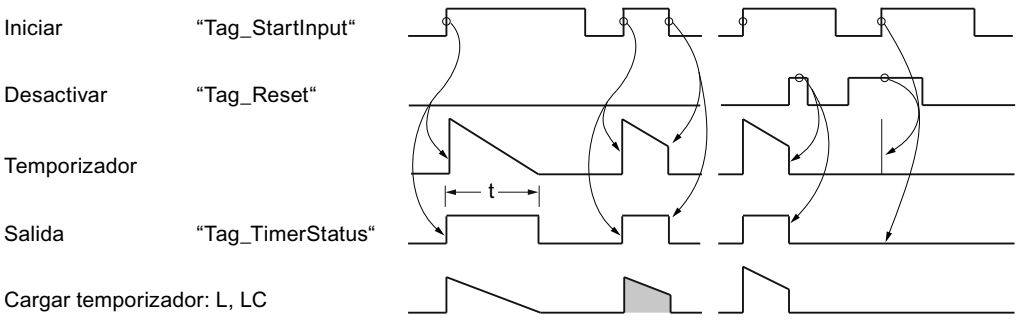
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor del temporizador (Página 4093)

SE: Arrancar temporizador como impulso prolongado

Descripción

La instrucción "Arrancar temporizador como impulso prolongado" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con el tiempo que figura en el acumulador 1, aunque el resultado lógico antes de la instrucción cambie a "0". Las consultas de si el estado del temporizador es "1" dan como resultado "1" mientras el temporizador está en marcha.

Con cada flanco de señal ascendente, la instrucción rearranca el temporizador con la duración programada, aunque este todavía no haya terminado.

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como impulso prolongado", el valor del temporizador decrementa de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido y el RLO de la operación lógica precedente devuelve el valor "0", el bit de estado OV

se pone a "1". No se inicia ningún tratamiento de errores síncrono. Si el acumulador 1 tiene un valor BCD válido, el bit de estado OV se pone a "0".

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como impulso prolongado" se utiliza la siguiente sintaxis:

```
SE <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso prolongado":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como impulso prolongado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

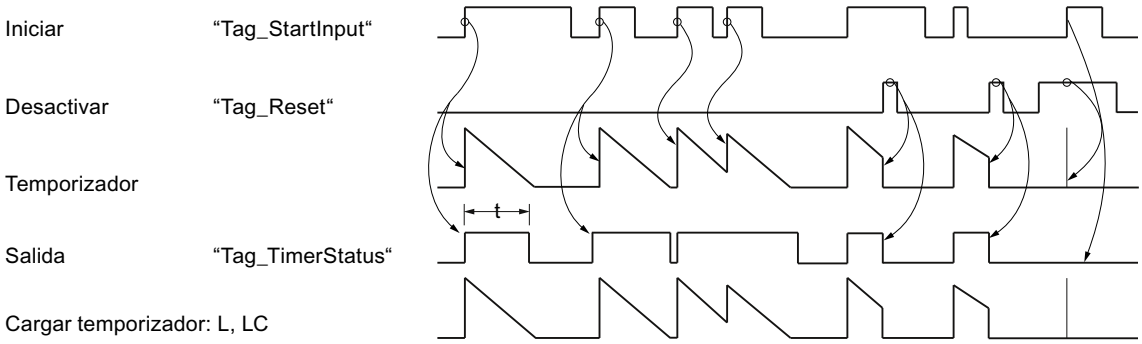
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SE "MyTimer"	// Arrancar temporizador como impulso prolongado cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)
L: Cargar valor del temporizador (Página 4093)

SD: Arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1, mientras el resultado lógico (RLO) antes de la instrucción siga siendo "1". Si el resultado lógico cambia a "0" antes de finalizar el temporizador, el temporizador se detiene.

Las consultas de si el estado del temporizador es "1" dan como resultado "1" si el temporizador ha terminado correctamente y el RLO actual antes de la instrucción es "1". Si el temporizador no ha terminado o el RLO cambia a "0", las consultas de si el estado del temporizador es "1" dan como resultado "0".

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como retardo a la conexión", el valor del temporizador se reduce de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido y el RLO de la operación lógica precedente devuelve el valor "0", el bit de estado OV se pone a "1". No se inicia ningún tratamiento de errores síncrono. Si el acumulador 1 tiene un valor BCD válido, el bit de estado OV se pone a "0".

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como retardo a la conexión" se utiliza la siguiente sintaxis:

```
SD <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como retardo a la conexión

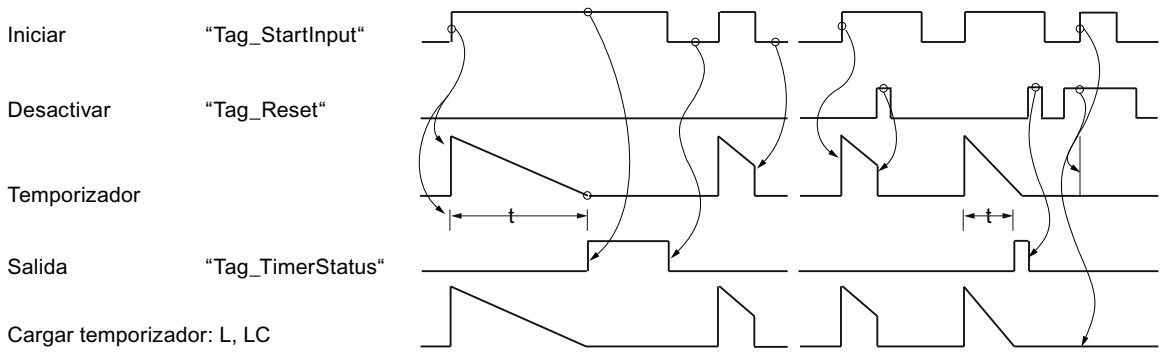
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SD "MyTimer"	// Arrancar temporizador como retardo a la conexión cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1"
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor del temporizador (Página 4093)

SS: Arrancar temporizador como retardo a la conexión con memoria

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1, aunque el resultado lógico (RLO) antes de la instrucción cambie a "0". Cuando el tiempo ha transcurrido, las consultas de si el estado del temporizador es "1" dan como resultado "1". El estado lógico del resultado lógico actual antes de la instrucción no influye en el resultado de la consulta.

Con cada flanco ascendente, la instrucción rearranca el temporizador con la duración programada, aunque este todavía no haya terminado.

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como retardo a la conexión con memoria", el valor del temporizador se reduce de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido y el RLO de la operación lógica precedente devuelve el valor "0", el bit de estado OV

se pone a "1". No se inicia ningún tratamiento de errores síncrono. Si el acumulador 1 tiene un valor BCD válido, el bit de estado OV se pone a "0".

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como retardo a la conexión con memoria" se utiliza la siguiente sintaxis:

```
SS <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión con memoria":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como retardo a la conexión con memoria.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

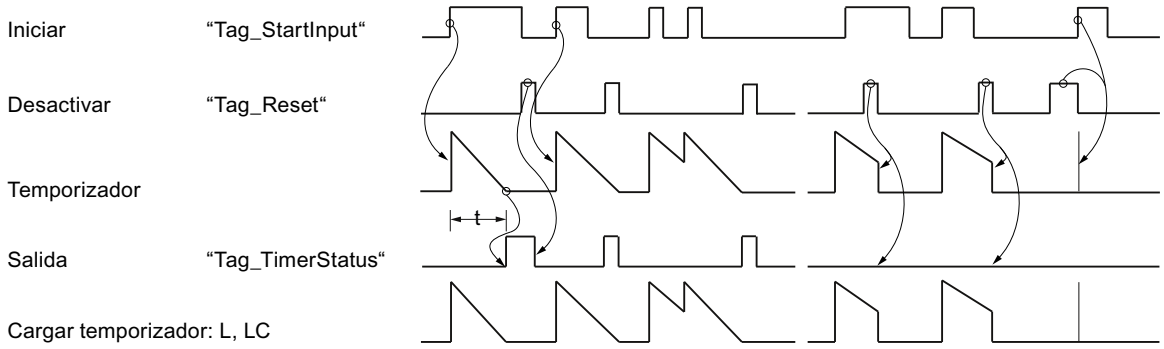
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SS "MyTimer"	// Arrancar temporizador como retardo a la conexión con memoria cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Consulta y activación de los bits de estado en AWL (Página 2560)
L: Cargar valor del temporizador (Página 4093)

SF: Arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Arrancar temporizador como retardo a la desconexión" arranca un temporizador concreto con un flanco de señal descendente. Cuando se registra un flanco de señal descendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1. Si el resultado lógico (RLO) cambia a "1" antes de finalizar el tiempo, el temporizador se inicializa. El temporizador solo arranca de nuevo cuando se registra un flanco de señal descendente al ejecutar la instrucción.

Las consultas del estado del temporizador en "1" dan como resultado "1" si el estado del resultado lógico al ejecutar la instrucción es "1" o si el temporizador programado está en marcha. Si el temporizador no cuenta o el RLO está a "0", las consultas de si el estado del temporizador es "1" dan como resultado "0".

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como retardo a la desconexión", el valor del temporizador decrementa de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido y el RLO de la operación lógica precedente devuelve el valor "0", el bit de estado OV se pone a "1". No se inicia ningún tratamiento de errores síncrono. Si el acumulador 1 tiene un valor BCD válido, el bit de estado OV se pone a "0".

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como retardo a la desconexión" se utiliza la siguiente sintaxis:

```
SF <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como retardo a la desconexión

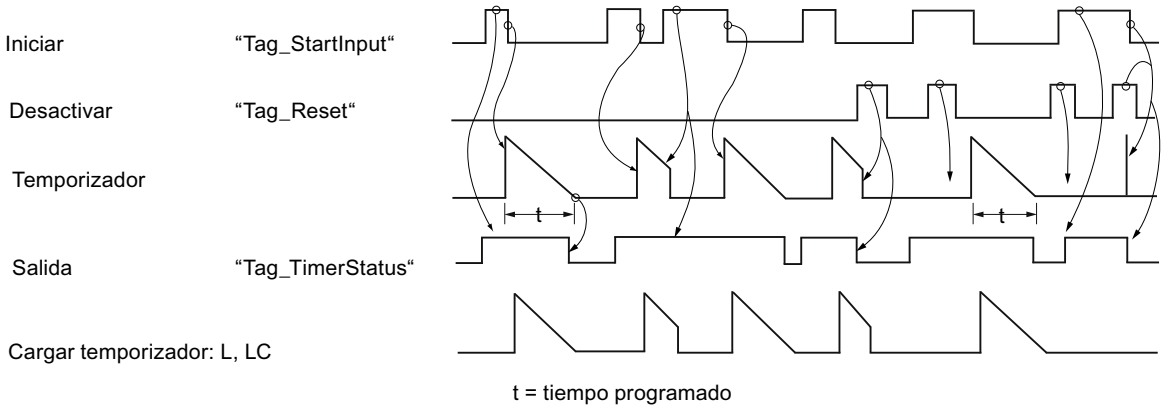
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
A "Tag_StartInput"           // Consultar si el operando es "1" y combinar el
                             resultado lógicamente con Y con el RLO actual.
L S5T#10s                    // Cargar tiempo (10 s)
SF "MyTimer"                 // Arrancar temporizador como retardo a la
                             desconexión cuando haya un flanco de señal
                             descendente en el operando "Tag_StartInput" o en el
                             operando "Tag_EnableInput" y el operando
                             "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"                // Consultar si el operando es "1" y combinar el
                             resultado lógicamente con Y con el RLO actual.
R "MyTimer"                  // Inicializar el temporizador si el RLO es "1" y
                             detener el tiempo.
A "MyTimer"                  // Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"          // Asignar el estado del temporizador al operando.
L "MyTimer"                  // Cargar el valor actual del temporizador en el
                             acumulador 1.
T "Tag_TimerValue"           // Transferir el valor actual del temporizador al
                             operando "Tag_TimerValue".
LC "MyTimer"                 // Cargar el valor actual del temporizador codificado
                             en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"       // Transferir el valor del temporizador codificado en
                             BCD al operando "Tag_TimerValue_BCD".
```

La figura siguiente muestra el diagrama de tiempo del ejemplo.



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor del temporizador (Página 4093)

Contadores

FR: Habilitar contador

Descripción

La instrucción "Habilitar contador" ejecuta las instrucciones "Inicializar contador", "Contador ascendente" y "Contador descendente" sin que exista actualmente un flanco de señal ascendente. Durante la ejecución, la instrucción pone la marca de flancos interna al estado lógico "0". La consulta anterior a la instrucción de contaje correspondiente debe seguir teniendo el estado lógico "1".

La instrucción "Habilitar contador" se ejecuta si el resultado lógico (RLO) cambia de "0" a "1" antes de la instrucción.

Para ejecutar una instrucción de contaje no es necesario habilitar un contador.

Sintaxis

Para la instrucción "Habilitar contador" se utiliza la siguiente sintaxis:

```
FR <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Habilitar contador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador que se habilita.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyCounter"	// Habilitar contador
A "Tag_Input_2"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CD "MyCounter"	// Decrementar contador
A "MyCounter"	// Consultar estado del contador
= "Tag_Output"	// Asignar el resultado de la consulta al operando.

En el ejemplo, el contador "MyCounter" se habilita cuando el estado lógico del operando "Tag_Input_1" cambia de "0" a "1". Si el operando "Tag_Input_2" tiene el estado lógico "1", el valor de contaje decrementa en uno al habilitar el contador. Si el estado lógico del operando "Tag_Input_2" es "0", el valor de contaje no cambia. El valor de contaje también decrementa en uno, independientemente de la habilitación del contador, cuando el estado lógico del operando "Tag_Input_2" cambia de "0" a "1".

La consulta del estado del contador da como resultado "1" si el estado del contador es mayor que cero. Si el estado del contador es igual a cero, el resultado de la consulta es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

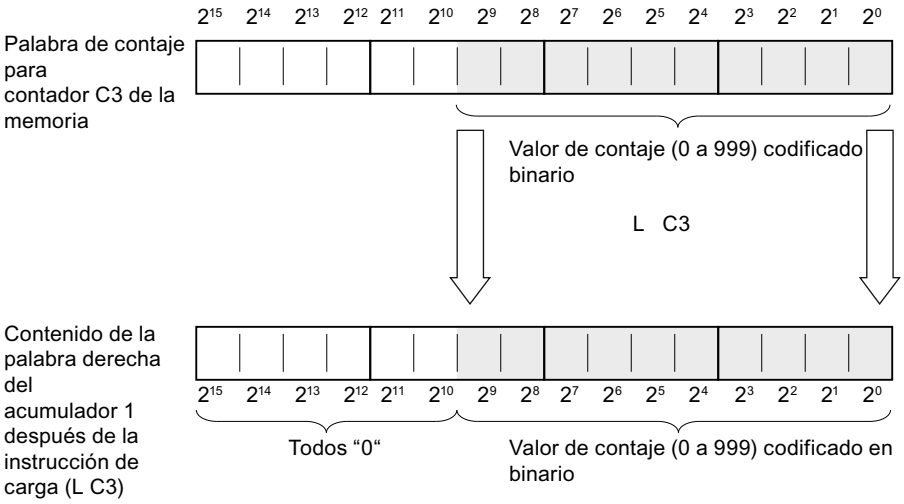
Consulta y activación de los bits de estado en AWL (Página 2560)

L: Cargar valor de contaje

Descripción

La instrucción "Cargar valor de contaje" transfiere el valor actual de un contador específico a la palabra derecha del acumulador 1 en código binario. A continuación, el valor de contaje cargado está disponible como entero de 16 bits para su posterior procesamiento.

La figura siguiente muestra el funcionamiento de la instrucción:



Sintaxis

Para la instrucción "Cargar valor de contaje" se utiliza la siguiente sintaxis:

```
L <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor de contaje":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor actual se carga.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "MyCounter"	// Cargar el valor de contaje actual en el acumulador 1 en código binario.
L "Tag_Value"	// Desplazar el valor de contaje al acumulador 2.
	// Cargar el valor del operando "Tag_Value" en el acumulador 1.
>=I	// Consulta sobre si el valor de contaje es mayor o igual al valor del operando "Tag_Value".
= "Tag_Output"	// Asignar el resultado de la consulta al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

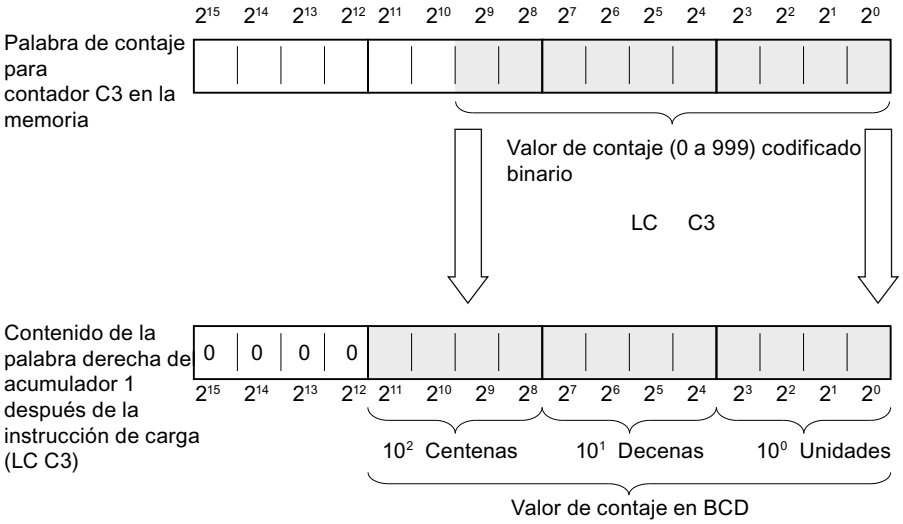
Consulta y activación de los bits de estado en AWL (Página 2560)

LC: Cargar valor de contaje codificado en BCD

Descripción

La instrucción "Cargar valor de contaje codificado en BCD" transfiere el valor actual de un contador específico a la palabra derecha del acumulador 1 en código BCD. A continuación, el valor de contaje cargado está disponible en formato BCD para su posterior procesamiento.

La figura siguiente muestra el funcionamiento de la instrucción "Cargar valor de contaje codificado en BCD":



Sintaxis

Para la instrucción "Cargar valor de contaje codificado en BCD" se utiliza la siguiente sintaxis:

```
LC <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor de contaje codificado en BCD":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor actual se carga.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LC "MyCounter"	// Cargar el valor de contaje actual en el acumulador 1 codificado en BCD.
T "Tag_Output"	// Transferir el valor de contaje actual al operando "Tag_Output" y guardarlo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

R: Poner contador a 0

Descripción

La instrucción "Poner contador a 0" pone a "0" el valor de contaje actual de un contador concreto.

La instrucción solo se ejecuta si el resultado lógico (RLO) previo a la instrucción tiene el estado lógico "1". Mientras el RLO antes de la instrucción "Poner contador a 0" esté a "1", las consultas de si el estado del contador es "1" darán como resultado "0".

Al poner un contador a 0, la marca de flancos interna para las instrucciones "Inicializar contador", "Contador ascendente" y "Contador descendente" no se desactiva.

Sintaxis

Para la instrucción "Poner contador a 0" se utiliza la siguiente sintaxis:

```
R <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Poner contador a 0":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor de contaje se pone a "0".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyCounter"	// Poner contador a 0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

S: Inicializar contador

Descripción

La instrucción "Inicializar contador" pone un contador concreto al valor que se encuentra en el acumulador 1 en el momento de la consulta.

La instrucción "Inicializar contador" se ejecuta solamente cuando el resultado lógico cambia de "0" a "1" antes de la instrucción. El flanco de señal ascendente también puede obtenerse mediante la instrucción "Habilitar contador", que pone a "0" la marca de flancos interna.

Para inicializar un contador debe haber un valor de contaje válido en el acumulador 1. Se consideran valores de contaje válidos los valores codificados en BCD con tres décadas. Sólo se admiten valores de contaje positivos, la instrucción de contaje no puede procesar valores negativos. Si el acumulador 1 cuenta con un valor BCD válido, el bit de estado OV tiene el estado lógico "0". Si no hay ningún valor BCD válido, el bit de estado OV tiene el estado lógico "1". No se inicia ningún tratamiento de errores síncrono.

Sintaxis

Para la instrucción "Inicializar contador" se utiliza la siguiente sintaxis:

```
S <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar contador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador que se inicializa.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L "Tag_CountValue"	// Cargar el valor de contaje en el acumulador 1.
S "MyCounter"	// Poner el contador al valor que se encuentra en el acumulador 1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

CU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa en uno el valor de contaje actual cuando el resultado lógico (RLO) antes de la instrucción cambie de "0" a "1". En cada flanco de señal ascendente, el valor de contaje se sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite, el RLO antes de la instrucción ya no influye en el valor de contaje.

Para ejecutar la instrucción "Contador ascendente" se requiere siempre un flanco de señal ascendente. El flanco de señal ascendente también puede obtenerse mediante la instrucción "Habilitar contador", que pone a "0" la marca de flancos interna.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

```
CU <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor de contaje se incrementa en uno con un flanco ascendente.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CU "MyCounter"	// Incrementar el valor de contaje en uno con un flanco ascendente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

CD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa en uno el valor de contaje actual cuando el resultado lógico (RLO) antes de la instrucción cambie de "0" a "1". En cada flanco de señal

ascendente, el valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite, el RLO antes de la instrucción ya no influye en el valor de contaje. No se cuenta con valores negativos.

Para ejecutar la instrucción "Contador descendente" se requiere siempre un flanco de señal ascendente. El flanco de señal ascendente también puede obtenerse mediante la instrucción "Habilitar contador", que pone a "0" la marca de flancos interna.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

```
CD <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor de contaje se decrementa en uno con un flanco ascendente.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CD "MyCounter"	// Decrementar el valor de contaje en uno con un flanco ascendente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Comparación

? I: Comparación de enteros

Descripción

La instrucción "Comparación de enteros" (16 bits) compara el contenido de la palabra derecha del acumulador 2 con el contenido de la palabra derecha del acumulador 1. Los contenidos de los acumuladores 1 y 2 se interpretan como enteros (de 16 bits). La instrucción ejecuta la comparación en función de la función de comparación indicada.

La tabla siguiente muestra las funciones de comparación disponibles y su significado:

Función de comparación	Significado
==I	Comparación de igualdad de enteros
<>I	Comparación de desigualdad de enteros
>I	Comparación de superioridad de enteros
<I	Comparación de inferioridad de enteros
>=I	Comparación de superioridad o igualdad de enteros
<=I	Comparación de inferioridad o igualdad de enteros

El resultado de la comparación influye en el resultado lógico (RLO) así como en los bits de estado CC 1 y CC 0. Si se cumple la condición de la comparación, el RLO se pone al estado lógico "1". Si no se cumple la condición de la comparación, el RLO se pone a "0".

La tabla siguiente muestra cómo las distintas funciones de comparación influyen en el RLO:

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
==I	0	1	0
<>I	1	0	1
>I	1	0	0
<I	0	0	1
>=I	1	1	0
<=I	0	1	1

La activación de los bits de estado CC 1 y CC 0 depende de la relación de los dos valores que intervienen en la comparación.

Los contenidos de los acumuladores 1 y 2 no cambian debido a la ejecución de las funciones de comparación. Tras ejecutar una función de comparación, es posible evaluar el resultado lógico con funciones de asignación y salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el valor del operando "Tag_Input_1" en el acumulador 1.
L "Tag_Input_2"	// Desplazar el contenido del acumulador 1 (Tag_Input_1) al acumulador 2.
	// Cargar el valor del operando "Tag_Input_2" en el acumulador 1.
>I	// Comparar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
= "Tag_Output"	// Asignar el resultado de la comparación al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores		
Tag_Input_1	10	10	5
Tag_Input_2	5	10	10
Tag_Output	1	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

? D: Comparación de enteros dobles

Descripción

La instrucción "Comparación de enteros dobles" (32 bits) compara el contenido del acumulador 2 (ACU 2) con el contenido del acumulador 1 (ACU 1). Los contenidos de los acumuladores 1 y 2 se interpretan como enteros dobles (de 32 bits). La instrucción ejecuta la comparación en función de la función de comparación indicada.

La tabla siguiente muestra las funciones de comparación disponibles y su significado:

Función de comparación	Significado
==D	Comparación de igualdad de enteros dobles
<>D	Comparación de desigualdad de enteros dobles
>D	Comparación de superioridad de enteros dobles
<D	Comparación de inferioridad de enteros dobles
>=D	Comparación de superioridad o igualdad de enteros dobles
<=D	Comparación de inferioridad o igualdad de enteros dobles

El resultado de la comparación influye en el resultado lógico (RLO) así como en los bits de estado CC 1 y CC 0. Si se cumple la condición de la comparación, el RLO se pone al estado lógico "1". Si no se cumple la condición de la comparación, el RLO se pone al estado lógico "0".

La tabla siguiente muestra cómo las distintas funciones de comparación influyen en el RLO:

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
==D	0	1	0
<>D	1	0	1
>D	1	0	0
<D	0	0	1
>=D	1	1	0
<=D	0	1	1

La activación de los bits de estado CC 1 y CC 0 depende de la relación de los dos valores que intervienen en la comparación.

Los contenidos de los acumuladores 1 y 2 no cambian debido a la ejecución de las funciones de comparación. Tras ejecutar una función de comparación, es posible evaluar el resultado lógico con funciones de asignación y salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el valor del operando "Tag_Input_1" en el acumulador 1.
L "Tag_Input_2"	// Desplazar el contenido del acumulador 1 (Tag_Input_1) al acumulador 2.
	// Cargar el valor del operando "Tag_Input_2" en el acumulador 1.
>D	// Comparar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
= "Tag_Output"	// Asignar el resultado de la comparación al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores		
Tag_Input_1	150 000	150 000	100 000
Tag_Input_2	100 000	150 000	150 000
Tag_Output	1	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

? R: Comparación de números en coma flotante

Descripción

La instrucción "Comparación de números en coma flotante" compara el contenido del acumulador 2 (ACU 2) con el contenido del acumulador 1 (ACU 1). Los contenidos de los acumuladores 1 y 2 se interpretan como números en coma flotante. Para ejecutar la comparación debe haber números en coma flotante válidos en los acumuladores 1 y 2. La instrucción ejecuta la comparación en función de la función de comparación indicada.

La tabla siguiente muestra las funciones de comparación disponibles y su significado:

Función de comparación	Significado
==R	Comparación de igualdad de números en coma flotante
<>R	Comparación de desigualdad de números en coma flotante
>R	Comparación de superioridad de números en coma flotante
<R	Comparación de inferioridad de números en coma flotante
>=R	Comparación de superioridad o igualdad de números en coma flotante
<=R	Comparación de inferioridad o igualdad de números en coma flotante

El resultado de la comparación influye en el resultado lógico (RLO) así como en los bits de estado CC 1 y CC 0. Si se cumple la condición de la comparación, el RLO se pone al estado lógico "1". Si no se cumple la condición de la comparación, el RLO se pone al estado lógico "0".

La tabla siguiente muestra cómo las distintas funciones de comparación influyen en el RLO:

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
==R	0	1	0
<>R	1	0	1
>R	1	0	0
<R	0	0	1
>=R	1	1	0
<=R	0	1	1

La activación de los bits de estado CC 1 y CC 0 depende de la relación de los dos valores que intervienen en la comparación.

Nota

Si se comparan números en coma flotante no válidos, la CPU pone el resultado lógico a "0" y los bits de estado CC 0, CC 1, OV y OS a "1".

Los contenidos de los acumuladores 1 y 2 no cambian debido a la ejecución de las funciones de comparación. Tras ejecutar una función de comparación, es posible evaluar el resultado lógico con funciones de asignación y salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`L "Tag_Input_1"`	`// Cargar el valor del operando "Tag_Input_1" en el acumulador 1.`
`L "Tag_Input_2"`	`// Desplazar el contenido del acumulador 1 (Tag_Input_1) al acumulador 2.`
	`// Cargar el valor del operando "Tag_Input_2" en el acumulador 1.`
`>R`	`// Comparar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".`
`= "Tag_Output"`	`// Asignar el resultado de la comparación al operando "Tag_Output".`

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores		
Tag_Input_1	10 515	10 515	5 232
Tag_Input_2	5 232	10 515	10 515
Tag_Output	1	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Funciones matemáticas

Enteros

+I: Sumar enteros (16 bits)

Descripción

La instrucción "Sumar enteros (16 bits)" suma los valores que se encuentran en las palabras derechas de los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros de 16 bits.

La instrucción guarda la suma en la palabra derecha del acumulador 1. La palabra izquierda del acumulador 1 no cambia.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la suma es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Sumar enteros (16 bits)" se utiliza la siguiente sintaxis:

```
+I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer sumando.
L "Tag_Value_2"	// Cargar el segundo sumando.
+I	// Sumar valores

AWL	Explicación
T "Tag_Result"	// Transferir la suma al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

-I: Restar enteros (16 bits)

Descripción

La instrucción "Restar enteros (16 bits)" resta el valor de la palabra derecha del acumulador 1 del valor de la palabra derecha del acumulador 2. La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros (de 16 bits).

La instrucción guarda la diferencia en la palabra derecha del acumulador 1. La palabra izquierda del acumulador 1 no cambia.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la diferencia es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Restar enteros (16 bits)" se utiliza la siguiente sintaxis:

```
-I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer valor de la resta en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.

AWL	Explicación
	// Cargar el segundo valor de la suma en el acumulador 1.
-I	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Result"	// Transferir la diferencia al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

*I: Multiplicar enteros (16 bits)

Descripción

La instrucción "Multiplicar enteros (16 bits)" multiplica los valores que se encuentran en las palabras derechas de los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros de 16 bits.

La instrucción guarda el producto en el acumulador 1 como entero doble (DINT).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el producto es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplicar enteros (16 bits)" se utiliza la siguiente sintaxis:

```
*I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el multiplicando.
L "Tag_Value_2"	// Cargar el multiplicador.

AWL	Explicación
*I	// Multiplicar valores
T "Tag_Result"	// Transferir el producto al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

/I: Dividir enteros (16 bits)

Descripción

La instrucción "Dividir enteros (16 bits)" divide el valor de la palabra derecha del acumulador 2 entre el valor de la palabra derecha del acumulador 1. La instrucción interpreta los valores de los acumuladores como enteros (de 16 bits).

La instrucción da dos resultados: el cociente y el resto de la división. El cociente es el resultado entero de la división. La instrucción lo guarda en la palabra derecha del acumulador 1. El resto de la división lo guarda en la palabra izquierda del acumulador 1. Si el dividendo es negativo, el resto de la división también lo será.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el cociente es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero tanto para el cociente como para el resto. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Dividir enteros (16 bits)" se utiliza la siguiente sintaxis:

```
/I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"              // Desplazar el dividendo al acumulador 2.
                             // Cargar el divisor en el acumulador 1.
/I                           // Dividir el valor del acumulador 2 entre el valor
                             del acumulador 1.
T "Tag_Result"               // Transferir el resultado al operando "Tag_Result".
```

Antes de la ejecución

La tabla siguiente muestra los contenidos de las palabras derechas de los acumuladores 1 y 2 antes de ejecutar la instrucción "Dividir enteros (16 bits)":

Acumulador	Valor
Acumulador 2	13
Acumulador 1	4

Después de la ejecución

La tabla siguiente muestra el contenido del acumulador 1 después de ejecutar la instrucción "Dividir enteros (16 bits)":

Acumulador 1	Valor
Bit 0 a 15 (palabra derecha)	3
Bit 16 a 31 (palabra izquierda)	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

+D: Sumar enteros dobles (32 bits)

Descripción

La instrucción "Sumar enteros dobles (32 bits)" suma los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros dobles. La instrucción guarda la suma en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la suma es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Sumar enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
+D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer sumando.
L "Tag_Value_2"	// Cargar el segundo sumando.
+D	// Sumar valores
T "Tag_Result"	// Transferir la suma al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

-D: Restar enteros dobles (32 bits)

Descripción

La instrucción "Restar enteros dobles (32 bits)" resta el valor del acumulador 1 del valor del acumulador 2. La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros dobles (de 32 bits). La instrucción guarda el resultado de la resta en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la diferencia es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Restar enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
-D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer valor de la resta en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el segundo valor de la resta en el acumulador 1.
-D	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Result"	// Transferir la diferencia al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

*D: Multiplicar enteros dobles (32 bits)

Descripción

La instrucción "Multiplicar enteros dobles (32 bits)" multiplica los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros dobles. La instrucción guarda el producto en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el producto es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplicar enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
*D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el multiplicando.
L "Tag_Value_2"	// Cargar el multiplicador.
*D	// Multiplicar valores
T "Tag_Result"	// Transferir el producto al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

/D: Dividir enteros dobles (32 bits)

Descripción

La instrucción "Dividir enteros dobles (32 bits)" divide el valor del acumulador 2 entre el valor del acumulador 1. La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros dobles (de 32 bits). La instrucción guarda el resultado entero de la división (cociente) en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el cociente es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero para el cociente. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Dividir enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
/D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"	// Desplazar el dividendo al acumulador 2.
	// Cargar el divisor en el acumulador 1.
/D	// Dividir el valor del acumulador 2 entre el valor del acumulador 1.
T "Tag_Result"	// Transferir el cociente al operando "Tag_Result".

Antes de la ejecución

La tabla siguiente muestra los contenidos de los acumuladores 1 y 2 antes de ejecutar la instrucción "Dividir enteros dobles (32 bits)":

Acumulador	Valor
Acumulador 2	13
Acumulador 1	4

Después de la ejecución

La tabla siguiente muestra el contenido del acumulador 1 después de ejecutar la instrucción "Dividir enteros dobles (32 bits)":

Acumulador	Valor
Acumulador 1	3

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

+: Sumar constantes

Descripción

La instrucción "Sumar constantes" suma el valor de una constante concreta al contenido del acumulador 1. El valor de la constante puede indicarse como entero o entero doble. La instrucción guarda el resultado en el acumulador 1. Si se suman enteros, la instrucción influye sólo en la palabra derecha del acumulador 1.

Para ejecutar la suma de un entero como operación DINT, antes de la constante hay que introducir "L#". La instrucción "Sumar constantes" utiliza números negativos.

La instrucción no influye en los bits de estado.

Sintaxis

Para la instrucción "Sumar constantes" se utiliza la siguiente sintaxis:

```
+ <constante>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar constantes":

Parámetro	Tipo de datos	Descripción
<constante>	INT, DINT	Constante que se suma.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
+ 25	// Aumentar en 25 el valor del acumulador 1.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
+ -100	// Reducir en 100 el valor del acumulador 1.
<I	// Consultar si el valor del acumulador 2 es menor que el valor del acumulador 1.
JC NEXT	// Si el resultado de la consulta es "1", saltar a la etiqueta NEXT .
	// Si el resultado de la consulta es "0", procesar la instrucción siguiente.
T "Tag_Result_2"	Transferir el contenido del acumulador 1 al operando "Tag_Result_2".
L "Tag_Value_4"	// Cargar el valor del operando "Tag_Value_4".
+ L#20	// Incrementar en 20 el contenido del acumulador 1.
	// La suma se efectúa conforme a un cálculo DINT
T "Tag_Result_3"	// Transferir el contenido del acumulador 1 al operando "Tag_Result_3".
NEXT: L "Tag_Value_5"	// Etiqueta "NEXT"
	// Cargar el valor del operando "Tag_Value_5".
+D	// Sumar los valores de los acumuladores 1 y 2.
T "Tag_Result_4"	// Transferir la suma al operando "Tag_Result_4".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

INC: Incrementar

Descripción

La instrucción "Incrementar" aumenta el valor que se encuentra en el acumulador 1 en el valor que se indica como parámetro de la instrucción. El valor indicado puede estar comprendido en un rango de valores de 0 a 255. El incremento sólo se realiza en el byte derecho del acumulador 1. No se lleva a cabo ningún traslado a los bytes que están a la izquierda.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

El contenido del acumulador 2 no cambia.

Si el valor del parámetro está fuera del rango de valores, el resultado quedará falsificado o bien se generará un mensaje de error.

Sintaxis

Para la instrucción "Incrementar" se utiliza la siguiente sintaxis:

```
INC <constante>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Incrementar":

Parámetro	Formato	Descripción
<constante>	Entero de 8 bits	Valor que se aumenta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                     Explicación
L "Tag_Value"           // Cargar el valor del operando en el acumulador 1.
INC 5                   // Aumentar en 5 el valor del acumulador 1.
T "Tag_Result"          // Transferir el contenido del acumulador 1 al
                        operando.
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

DEC: Decrementar

Descripción

La instrucción "Decrementar" reduce el valor que se encuentra en el acumulador 1 en el valor que se indica como parámetro de la instrucción. El valor indicado puede estar comprendido en un rango de valores de 0 a 255. El decremento sólo se realiza en el byte derecho del acumulador 1. No se lleva a cabo ningún traslado a los bytes que están a la izquierda.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

El contenido del acumulador 2 no cambia.

Si el valor del parámetro está fuera del rango de valores, el resultado quedará falsificado o bien se generará un mensaje de error.

Sintaxis

Para la instrucción "Decrementar" se utiliza la siguiente sintaxis:

```
DEC <constante>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Decrementar":

Parámetro	Formato	Descripción
<constante>	Entero de 8 bits	Valor que se decrementa.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
DEC 5	// Decrementar en 5 el valor del acumulador 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

MOD: Obtener resto de división

Descripción

La instrucción "Obtener resto de división" divide el valor del acumulador 2 entre el valor del acumulador 1 y guardar el resto de la división.

La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros dobles. La instrucción guarda el resto de la división en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resto de la división es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero para el resultado. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Obtener resto de división" se utiliza la siguiente sintaxis:

```
MOD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"	// Desplazar el dividendo al acumulador 2.
	// Cargar el divisor en el acumulador 1.
MOD	// Dividir el valor del acumulador 2 entre el valor del acumulador 1.
T "Tag_Result"	// Transferir el resto de la división al operando "Tag_Result".

Antes de la ejecución

La tabla siguiente muestra los contenidos de los acumuladores 1 y 2 antes de ejecutar la instrucción "Obtener resto de división":

Acumulador	Valor
Acumulador 2	13
Acumulador 1	4

Después de la ejecución

La tabla siguiente muestra el contenido del acumulador 1 después de ejecutar la instrucción "Obtener resto de división":

Acumulador	Valor
Acumulador 1	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Números en coma flotante

+R: Sumar números en coma flotante

Descripción

La instrucción "Sumar números en coma flotante" suma los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como números en coma flotante. La instrucción guarda la suma en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la suma es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Sumar números en coma flotante" se utiliza la siguiente sintaxis:

```
+R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer sumando.
L "Tag_Value_2"	// Cargar el segundo sumando.
+R	// Sumar valores
T "Tag_Result"	// Transferir la suma al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

-R: Restar números en coma flotante

Descripción

La instrucción "Restar números en coma flotante" resta el valor del acumulador 1 del valor del acumulador 2. La instrucción interpreta los valores de los acumuladores 1 y 2 como números en coma flotante. La instrucción guarda la diferencia en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la diferencia es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Restar números en coma flotante" se utiliza la siguiente sintaxis:

```
-R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer valor de la resta en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el segundo valor de la resta en el acumulador 1.
-R	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Result"	// Transferir la diferencia al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

*R: Multiplicar números en coma flotante

Descripción

La instrucción "Multiplicar números en coma flotante" multiplica los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como números en coma flotante. La instrucción guarda el producto en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el producto es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplicar números en coma flotante" se utiliza la siguiente sintaxis:

```
*R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el multiplicando.
L "Tag_Value_2"	// Cargar el multiplicador.
*R	// Multiplicar valores
T "Tag_Result"	// Transferir el producto al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

/R: Dividir números en coma flotante

Descripción

La instrucción "Dividir números en coma flotante" divide el valor del acumulador 2 entre el valor del acumulador 1. La instrucción interpreta los valores de los acumuladores 1 y 2 como números en coma flotante. La instrucción guarda el resultado entero de la división (cociente) en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el cociente es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero para el cociente. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

El contenido del acumulador 2 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Dividir números en coma flotante" se utiliza la siguiente sintaxis:

```
/R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"	// Desplazar el dividendo al acumulador 2.
	// Cargar el divisor en el acumulador 1.
/R	// Dividir el valor del acumulador 2 entre el valor del acumulador 1.
T "Tag_Result"	// Transferir el cociente al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" calcula el valor absoluto del valor que se encuentra en el acumulador 1. La instrucción interpreta el contenido del acumulador 1 como número en coma flotante y pone el signo de la mantisa a "0". El resultado se guarda en el acumulador 1.

La instrucción "Calcular valor absoluto" no influye en los bits de estado.

Sintaxis

Para la instrucción "Calcular valor absoluto" se utiliza la siguiente sintaxis:

```
ABS
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el número en coma flotante en el acumulador 1.
ABS	// Calcular valor absoluto
T "Tag_Result"	// Transferir el resultado al operando "Tag_Result".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	-1.5E+02
Tag_Result	1.5E+02

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Números en coma flotante, avanzado

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" calcula el cuadrado de un número en coma flotante que se encuentra en el acumulador 1. El resultado se guarda en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular cuadrado" se utiliza la siguiente sintaxis:

```
SQR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
SQR	// Calcular el cuadrado
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" calcula la raíz cuadrada del número en coma flotante que se encuentra en el acumulador 1. El resultado se guarda en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si en el acumulador 1 se introduce un valor inferior a cero o un número en coma flotante no válido, la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

Si en el acumulador 1 se encuentra el valor "0", la instrucción devuelve el valor "0".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular raíz cuadrada" se utiliza la siguiente sintaxis:

```
SQRT
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
SQRT	// Calcular la raíz cuadrada
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" calcula la potencia en base e (e = 2.718282) del número en coma flotante que se encuentra en el acumulador 1. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor exponencial" se utiliza la siguiente sintaxis:

```
EXP
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
EXP	// Calcular el valor exponencial
	// Guardar el resultado en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" calcula el logaritmo natural en base e (e = 2.718282) a partir del número en coma flotante que se encuentra en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si en el acumulador 1 hay un valor inferior o igual a cero, la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular logaritmo natural" se utiliza la siguiente sintaxis:

```
LN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
LN	// Calcular el logaritmo natural.
	// Guardar el resultado en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" calcula el seno del ángulo que se encuentra en el acumulador 1. El ángulo debe indicarse en radianes y encontrarse en el acumulador 1 en forma de número en coma flotante. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor de seno" se utiliza la siguiente sintaxis:

```
SIN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
SIN	// Calcular valor de seno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" calcula, a partir del valor que se encuentra en el acumulador 1, el tamaño del ángulo que corresponde a dicho valor. El valor del acumulador 1 debe encontrarse en forma de número en coma flotante en un rango de valores de "-1" a "1".

El resultado de la instrucción se devuelve en radianes y se guarda en el acumulador 1. Según sea el valor del acumulador 1, el resultado puede encontrarse en un rango de valores de $-\pi/2$ a $+\pi/2$ (π= 3,14159).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si el valor que está en el acumulador 1 rebasa por exceso el rango admisible, la instrucción escribe un número en coma flotante no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor de arcoseno" se utiliza la siguiente sintaxis:

```
ASIN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.

AWL	Explicación
ASIN	// Calcular el valor de arcoseno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" calcula el coseno del ángulo que se encuentra en el acumulador 1. El tamaño del ángulo debe indicarse en radianes y encontrarse en el acumulador 1 en forma de número en coma flotante. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor de coseno" se utiliza la siguiente sintaxis:

```
COS
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
COS	// Calcular valor de coseno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" calcula, a partir del valor que se encuentra en el acumulador 1, el tamaño del ángulo que corresponde a dicho valor. El valor del acumulador 1 debe encontrarse en forma de número en coma flotante en un rango de valores de "-1" a "1".

El resultado de la instrucción se devuelve en radianes y se guarda en el acumulador 1. Según sea el valor del acumulador 1, el resultado puede encontrarse en un rango de valores de "0" a "π" (π= 3,14159).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si el valor que está en el acumulador 1 rebasa por exceso el rango admisible, la instrucción escribe un número en coma flotante no válido en el acumulador 1 y pone los bits de estado

CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor de arcocoseno" se utiliza la siguiente sintaxis:

```
ACOS
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
ACOS	// Calcular el valor de arcocoseno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" calcula la tangente del ángulo que se encuentra en el acumulador 1. El tamaño del ángulo debe indicarse en radianes y encontrarse en el acumulador 1 en forma de número en coma flotante. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor de tangente" se utiliza la siguiente sintaxis:

```
TAN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
TAN	// Calcular valor de tangente
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" calcula, a partir del valor que se encuentra en el acumulador 1, el tamaño del ángulo que corresponde a dicho valor. El valor del acumulador 1 debe encontrarse en forma de número en coma flotante.

El resultado de la instrucción se devuelve en radianes y se guarda en el acumulador 1. Según sea el valor del acumulador 1, el resultado puede encontrarse en un rango de valores de $-\pi/2$ a $+\pi/2$ (π= 3,14159).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si el valor que está en el acumulador 1 rebasa por exceso el rango admisible, la instrucción escribe un número en coma flotante no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Calcular valor de arcotangente" se utiliza la siguiente sintaxis:

```
ATAN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
ATAN	// Calcular el valor de la arcotangente
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.

AWL	Explicación
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Carga y transferencia

Carga

L: Cargar

Descripción

La instrucción "Cargar" carga el contenido de un operando específico en el acumulador 1. El operando que debe cargarse puede estar direccionado en una de las áreas de memoria siguientes en forma de byte, palabra o palabra doble:

- Memoria imagen de proceso de las entradas y salidas (I, Q)
- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)
- Punteros
- Periferia (P)
- Temporizador (T)
- Contador (C)

El área de memoria del acumulador 1 está organizada por bytes y tiene 32 bits de ancho.

La instrucción escribe el contenido de un operando que cargar en formato de byte alineado a la derecha en el acumulador 1. Los bytes restantes del acumulador 1 se rellenan con "0".

La instrucción escribe el contenido de un operando que cargar en formato de palabra en la palabra derecha del acumulador 1. El byte con un direccionamiento más alto se transfiere al

byte derecho (bits 0 a 7) del acumulador 1. El byte con un direccionamiento más bajo se escribe a su izquierda. Los bytes restantes de la palabra izquierda del acumulador 1 se rellenan con "0".

La instrucción escribe el contenido de un operando que cargar en formato de palabra doble en los 32 bits del acumulador 1. El byte con un direccionamiento más alto se transfiere al byte derecho (bits 0 a 7) del acumulador 1. El byte con el direccionamiento más bajo se escribe en los bits 24 y 31.

La tabla siguiente muestra a modo de ejemplo cómo el contenido del acumulador 1 se transforma en formato de byte, palabra y palabra doble al cargar operandos:

Instrucción	Acumulador 1							
	31 . . .	. . . 24	23 . . .	. . . 16	15 . . .	. . . 8	7 . . .	. . . 0
Cargar un byte: L MB10	0000	0000	0000	0000	0000	0000	<MB10>	
Cargar una palabra: L MW10	0000	0000	0000	0000	<MB10>		<MB11>	
Cargar una palabra doble: L MD10	<MB10>		<MB11>		<MB12>		<MB13>	

Al cargar el operando indicado, el contenido previo del acumulador 1 se desplaza al acumulador 2. La instrucción "Cargar" transfiere todo el contenido del acumulador 1 al acumulador 2. El contenido previo del acumulador 2 se pierde.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Cargar" se utiliza la siguiente sintaxis:

```
L <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, temporizadores, REAL, DATE, TOD, CHAR	I, Q, PI, M, L, DB, DI, T, C, puntero, parámetro	Operando cuyo contenido se carga.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L STW: Cargar palabra de estado en acumulador 1

Descripción

La instrucción "Cargar palabra de estado en acumulador 1" carga la palabra de estado en el acumulador 1. Durante la ejecución, la instrucción escribe los diferentes bits de estado en la palabra derecha del acumulador 1. Los bits restantes del acumulador 1 se rellenan con "0".

La tabla siguiente muestra el contenido del acumulador después de ejecutar la instrucción "Cargar palabra de estado en acumulador 1":

	Acumulador 1									
Número de bit	31-9	8	7	6	5	4	3	2	1	0
Contenido	0	BR	CC 1	CC 0	OV	OS	0	0	RLO	0

Esta instrucción se ejecuta independientemente del estado lógico del bit de estado.

Sintaxis

Para la instrucción "Cargar palabra de estado en acumulador 1" se utiliza la siguiente sintaxis:

```
L STW
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L STW	// Cargar palabra de estado en acumulador 1.
T "Tag_STW"	// Transferir el contenido del acumulador 1 (bits de estado) al operando "Tag_STW".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LAR1: Cargar contenido del acumulador 1 en AR1

Descripción

La instrucción "Cargar contenido del acumulador 1 en AR1 " carga el contenido del acumulador 1 en el registro de direcciones 1 (AR1). El valor del acumulador 1 debe equivaler al formato de un puntero de área (POINTER).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar contenido del acumulador 1 en AR1" se utiliza la siguiente sintaxis:

```
LAR1
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L P#10.0	// Cargar el puntero (P#10.0) en el acumulador 1.
LAR1	// Cargar el contenido del acumulador 1 en el registro de direcciones 1.
L MW [AR1,P#4.0]	// Cargar MW14 en acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR1,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LAR1 <D>: Cargar palabra doble o puntero de área en AR1

Descripción

La instrucción "Cargar palabra doble o puntero de área en AR1" carga el contenido de una palabra doble o de un puntero intraárea o interárea en el registro de direcciones 1 (AR1). El contenido de la palabra doble debe equivaler al formato de un puntero de área.

El puntero y la palabra doble pueden estar direccionados en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar palabra doble o puntero de área en AR1" se utiliza la siguiente sintaxis:

```
LAR1 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar palabra doble o puntero de área en AR1":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Input	DWORD, POINTER	D, M, L	Operando cuyo contenido se carga.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#10.0	// Cargar el contenido del puntero intraárea P#10.0 en el registro de direcciones 1.
L MW [AR1,P#4.0]	// Cargar MW14 en el acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR1,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".

AWL	Explicación
LAR1 MD20	// Cargar el contenido de MD20 (MD20 = P#30.0) en el registro de direcciones 1.
A I [AR1,P#2.1]	// Consultar si el estado lógico del bit I32.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LAR1 AR2: Cargar contenido del AR2 en AR1

Descripción

La instrucción "Cargar contenido del AR2 en AR1" carga el contenido del registro de direcciones 2 (AR2) en el registro de direcciones 1 (AR1).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar contenido del AR2 en AR1" se utiliza la siguiente sintaxis:

```
LAR1 AR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR2 P#20.0	// Cargar el contenido del puntero intraárea P#20.0 en el registro de direcciones 2.
LAR1 AR2	// Copiar el contenido del registro de direcciones 2 en el registro de direcciones 1.
L MW [AR1,P#4.0]	// Cargar MW24 en el acumulador 1.

AWL	Explicación
T "Tag_2"	// Transferir el acumulador 1 a "Tag_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LAR2: Cargar contenido del acumulador 1 en AR2

Descripción

La instrucción "Cargar contenido del acumulador 1 en AR2 " carga el contenido del acumulador 1 en el registro de direcciones 2 (AR2). El valor del acumulador 1 debe equivaler al formato de un puntero de área (POINTER).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar contenido del acumulador 1 en AR2" se utiliza la siguiente sintaxis:

```
LAR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L P#10.0	// Cargar el puntero (P#10.0) en el acumulador 1.
LAR2	// Cargar el contenido del acumulador 1 en el registro de direcciones 2.
L MW [AR2,P#4.0]	// Cargar MW14 en el acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.

AWL	Explicación
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR2,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LAR2 <D>: Cargar palabra doble o puntero de área en AR2

Descripción

La instrucción "Cargar palabra doble o puntero de área en AR2" carga el contenido de una palabra doble o de un puntero intraárea o interárea en el registro de direcciones 2 (AR2). El contenido de la palabra doble debe equivaler al formato de un puntero de área (POINTER).

El puntero y la palabra doble pueden estar direccionados en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar palabra doble o puntero de área en AR2" se utiliza la siguiente sintaxis:

```
LAR2 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar palabra doble o puntero de área en AR2":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Input	DWORD, POINTER	D, M, L	Operando cuyo contenido se carga.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR2 P#10.0	// Cargar el contenido del puntero intraárea P#10.0 en el registro de direcciones 2
L MW [AR2,P#4.0]	// Cargar MW14 en el acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR2,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".
LAR2 MD20	// Cargar el contenido de MD20 (MD20 = P#30.0) en el registro de direcciones 2.
A I [AR2,P#2.1]	// Consultar si el estado lógico del bit I32.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Transferencia

T: Transferir

Descripción

La instrucción "Transferir" transfiere el contenido del acumulador 1 a un operando concreto por bytes, palabras o palabras dobles. El contenido del acumulador 1 no cambia. El operando indicado debe estar direccionado en una de las áreas de memoria siguientes:

- Memoria imagen de proceso de las entradas y salidas (I, Q)
- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)
- Periferia (PQ)

El número de bytes que se transferirán depende del formato del operando indicado. Si el operando está en formato de byte, la instrucción transfiere el contenido de los bits 0 a 7 del acumulador 1.

Si el operando indicado está en formato de palabra, la instrucción transfiere el contenido de la palabra que está alineada a la derecha en el acumulador 1. El byte derecho del acumulador 1 se transfiere al byte con un direccionamiento más alto en el operando. Al byte con un direccionamiento más bajo del operando se transfieren los bits de 8 a 15 del acumulador 1.

Si el operando indicado está en formato de palabra doble, la instrucción transfiere todo el contenido del acumulador 1. El byte derecho del acumulador 1 se transfiere al byte con un direccionamiento más alto en el operando. Al byte con un direccionamiento más bajo del operando se transfieren los bits 24 a 31 del acumulador 1.

La tabla siguiente muestra un ejemplo de cómo se transfiere el contenido del acumulador 1 por bytes, palabras o palabras dobles:

Instrucción	Acumulador 1			
Bits	31 24	23 16	15 8	7 0
Contenido del acumulador 1	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)
Transferir un byte				Byte (n)
Transferir una palabra			Byte (n)	Byte (n+1)
Transferir una palabra doble	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Transferir" se utiliza la siguiente sintaxis:

```
T <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Transferir":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Output	BYTE, WORD, DWORD, SINT, INT, DINT, USINT, UINT, UDINT, temporizadores, REAL, DATE, TOD, CHAR	I, Q, PQ, M, L, DB, DI	Operando al que se transfiere el contenido del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result_1"	// Transferir el producto (contenido del acumulador 1) al operando "Tag_Result_1".
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
T "Tag_Result_2"	// Transferir la suma (contenido del acumulador 1) al operando "Tag_Result_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

T STW: Transferir acumulador 1 a palabra de estado

Descripción

La instrucción "Transferir acumulador 1 a palabra de estado" transfiere los bits 0 a 8 del acumulador 1 a la palabra de estado.

La tabla siguiente muestra los bits del acumulador 1 con los que se sobrescriben los bits de estado después de ejecutar la instrucción:

	Acumulador 1								
Número de bit	8	7	6	5	4	3	2	1	0
Bits de estado	BR	CC 1	CC 0	OV	OS	0	0	RLO	0

Esta instrucción se ejecuta independientemente del estado lógico del bit de estado.

Sintaxis

Para la instrucción "Transferir acumulador 1 a palabra de estado" se utiliza la siguiente sintaxis:

```
T STW
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_STW"	// Cargar el valor del operando "Tag_STW" en el acumulador 1.
T STW	// Transferir los bits 0 a 8 del acumulador 1 a la palabra de estado.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

CAR: Intercambiar AR1 y AR2

Descripción

La instrucción "Intercambiar AR1 y AR2" intercambia los contenidos de los registros de direcciones 1 (AR1) y 2 (AR2).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Intercambiar AR1 y AR2" se utiliza la siguiente sintaxis:

```
CAR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#10.0	// Cargar el contenido del puntero intraárea P#10.0 en el registro de direcciones 1.
LAR2 P#20.0	// Cargar el contenido del puntero intraárea P#20.0 en el registro de direcciones 2.
CAR	// Intercambiar el contenido de los registros de direcciones.
L MD [AR1,P#2.0]	// Cargar MD22 en el acumulador 1.
T MD [AR2,P#2.0]	// Transferir MD12 al acumulador 2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TAR1: Transferir AR1 a acumulador 1

Descripción

La instrucción "Transferir AR1 a acumulador 1" transfiere el contenido del registro de direcciones 1 (AR1) al acumulador 1. El contenido del acumulador 1 se desplaza al acumulador 2.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Transferir AR1 a acumulador 1" se utiliza la siguiente sintaxis:

```
TAR1
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#15.0	// Cargar el contenido del puntero intraárea P#15.0 en el registro de direcciones 1.
TAR1	// Transferir el contenido del registro de direcciones 1 (P#15.0) al acumulador 1.
T "Tag_Pointer"	// Transferir el contenido del acumulador 1 al operando "Tag_Pointer".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TAR1 <D>: Transferir AR1 a palabra doble

Descripción

La instrucción "Transferir AR1 a palabra doble" transfiere el contenido del registro de direcciones 1 (AR1) a una palabra doble. La palabra doble debe estar direccionada en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Transferir AR1 a palabra doble" se utiliza la siguiente sintaxis:

```
TAR1 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Transferir AR1 a palabra doble":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Output	DWORD	D, M, L	Operando al que se transfiere el contenido del registro de direcciones 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
TAR1 %DBD20	// Transferir el contenido del registro de direcciones 1 a la doble palabra de datos DBD20.
TAR1 %DID30	// Transferir el contenido del registro de direcciones 1 a la doble palabra de instancia DID30.
TAR1 %LD18	// Transferir el contenido del registro de direcciones 1 a la doble palabra de datos locales LD18.
TAR1 %MD24	// Transferir el contenido del registro de direcciones 1 a la doble palabra de marcas MD24.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TAR1 AR2: Transferir AR1 a AR2

Descripción

La instrucción "Transferir AR1 a AR2" copia el contenido del registro de direcciones 1 (AR1) en el registro de direcciones 2 (AR2).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Transferir AR1 a AR2" se utiliza la siguiente sintaxis:

```
TAR1 AR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#10.4	// Cargar el contenido del puntero intraárea P#10.4 en el registro de direcciones 1.
TAR1 AR2	// Transferir el contenido del registro de direcciones 1 al registro de direcciones 2.
A I [AR2,P#3.2]	// Consultar si el estado lógico del bit I13.6 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag1"	// Poner el operando "Tag_1" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TAR2: Transferir AR2 a acumulador 1

Descripción

La instrucción "Transferir AR2 a acumulador 1" transfiere el contenido del registro de direcciones 2 (AR2) al acumulador 1. El contenido del acumulador 1 se desplaza al acumulador 2.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Transferir AR2 a acumulador 1" se utiliza la siguiente sintaxis:

```
TAR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR2 P#20.0	// Cargar el contenido del puntero intraárea al registro de direcciones 2 P#20.0.
TAR2	// Transferir el contenido del registro de direcciones 2 (P#20.0) al acumulador 1.
T "Tag_Pointer"	// Transferir el contenido del acumulador 1 al operando "Tag_Pointer".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TAR2 <D>: Transferir AR2 a palabra doble

Descripción

La instrucción "Transferir AR2 a palabra doble" transfiere el contenido del registro de direcciones 2 (AR2) a una palabra doble. La palabra doble debe estar direccionada en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Transferir AR2 a palabra doble" se utiliza la siguiente sintaxis:

```
TAR2 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Transferir AR2 a palabra doble":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Output	DWORD	D, M, L	Operando al que se transfiere el contenido del registro de direcciones 2.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
TAR2 %DBD20	// Transferir el contenido del registro de direcciones 2 a la doble palabra de datos DBD20.
TAR2 %DID30	// Transferir el contenido del registro de direcciones 2 a la doble palabra de instancia DID30.

AWL	Explicación
TAR2 %LD18	// Transferir el contenido del registro de direcciones 2 a la doble palabra de datos locales LD18.
TAR2 %MD24	// Transferir el contenido del registro de direcciones 2 a la doble palabra de marcas MD24.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Conversión

BTI: Convertir BCD en entero (16 bits)

Descripción

La instrucción "Convertir BCD en entero (16 bits)" convierte el valor que está en la palabra derecha del acumulador 1 en un entero (de 16 bits). La instrucción interpreta el valor que debe convertirse como un número decimal codificado en binario de tres dígitos (BCD).

Los bits 1 a 11 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-999" a "+999".

El signo del valor del resultado se lee del bit 15 del acumulador 1. Si el estado lógico del bit es "0", el signo es positivo. Si el estado lógico del bit es "1", el signo es negativo. Los bits 12 a 14 no se utilizan en la conversión.

Si el código BCD contiene un error BCD, la CPU activa los bits OV y OS de la palabra de estado.

El resultado de la conversión se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

Sintaxis

Para la instrucción "Convertir BCD en entero (16 bits)" se utiliza la siguiente sintaxis:

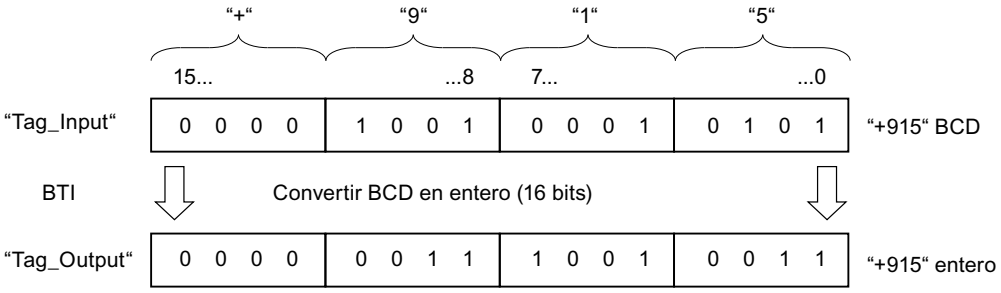
```
BTI
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar BCD en el acumulador 1.
BTI	// Convertir BCD en entero de 16 bits.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ITB: Convertir entero en BCD (16 bits)

Descripción

La instrucción "Convertir entero en BCD (16 bits)" convierte el valor que se encuentra en la palabra derecha del acumulador 1 en un número decimal en código binario de tres dígitos (BCD). La instrucción interpreta el valor que debe convertirse como un entero de 16 bits.

Los bits 0 a 11 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-999" a "+999". Si el valor que debe convertirse está fuera de este rango, los bits OV y OS se ponen al estado lógico "1". En este caso no se lleva a cabo la conversión.

El signo del valor del resultado se lee de los bits 12 a 15 del acumulador 1. Si el estado lógico de los bits es "0", el signo es positivo. Si el estado lógico de los cuatro bits es "1", el signo es negativo.

El resultado de la conversión se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

Sintaxis

Para la instrucción "Convertir entero en BCD (16 bits)" se utiliza la siguiente sintaxis:

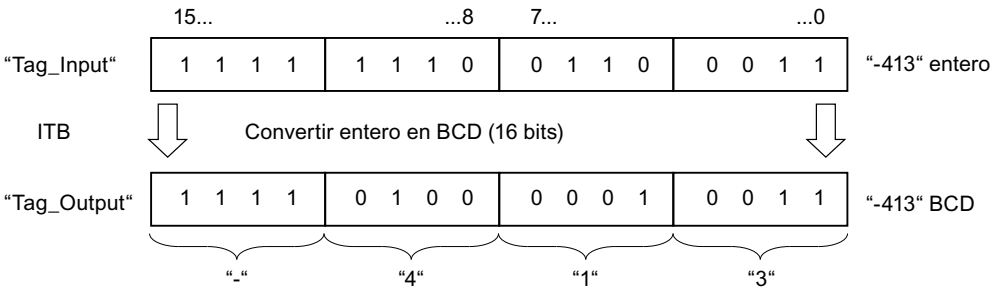
```
ITB
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero (16 bits) en el acumulador 1.
ITB	// Convertir entero en BCD.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

BTD: Convertir BCD en entero doble (32 bits)

Descripción

La instrucción "Convertir BCD en entero doble (32 bits)" convierte el valor que está en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un número decimal codificado en binario de siete dígitos (BCD).

Los bits 0 a 27 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-9999999" a "+9999999".

El signo del valor del resultado se lee del bit 31 del acumulador 1. Si el estado lógico del bit es "0", el signo es positivo. Si el estado lógico del bit es "1", el signo es negativo. Los bits 28 a 30 no se utilizan en la conversión.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Si el código BCD contiene un error BCD, la CPU activa los bits OV y OS de la palabra de estado.

Sintaxis

Para la instrucción "Convertir BCD en entero doble (32 bits)" se utiliza la siguiente sintaxis:

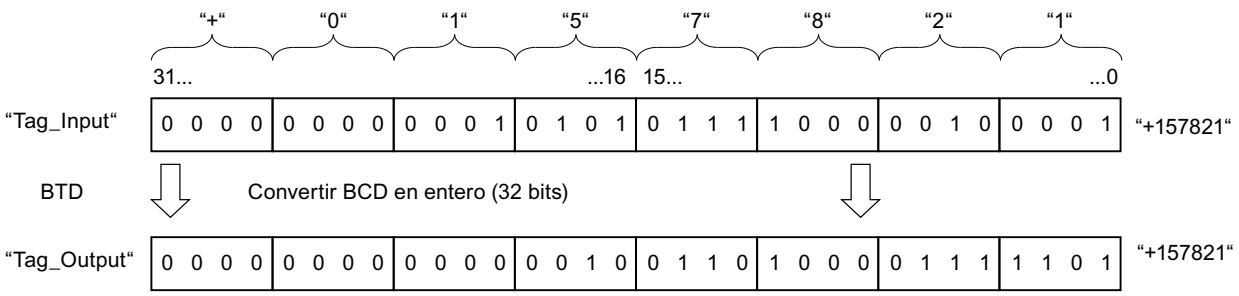
```
BTD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar BCD en el acumulador 1.
BTD	// Convertir BCD en un entero doble.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

ITD: Convertir entero (16 bits) en entero doble (32 bits)

Descripción

La instrucción "Convertir entero (16 bits) en entero doble (32 bits)" convierte el valor que se encuentra en la palabra derecha del acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un entero de 16 bits.

Los bits 0 a 15 se aplican sin cambios en la conversión. Los bits 16 a 31 del valor del resultado se rellenan con el estado lógico del bit 15.

El valor del resultado se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Convertir entero (16 bits) en entero doble (32 bits)" se utiliza la siguiente sintaxis:

```
ITD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero (16 bits) en el acumulador 1.
ITD	// Convertir entero de 16 bits en entero doble de 32 bits.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Acumulador 1							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Antes de la ejecución	XXXX	XXXX	XXXX	XXXX	1111	1111	1111	0110
Después de la ejecución	1111	1111	1111	1111	1111	1111	1111	0110
	X : el estado lógico de los bits no es relevante para la conversión							

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

DTB: Convertir entero doble (32 bits) en BCD

Descripción

La instrucción "Convertir entero doble (32 bits) en BCD" convierte el valor que está en el acumulador 1 en un número decimal en código binario de siete dígitos (BCD). La instrucción interpreta el valor que debe convertirse como un entero doble.

Los bits 0 a 27 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-9999999" a "+9999999". Si el valor que debe convertirse está fuera de este rango, los bits OV y OS se ponen al estado lógico "1". En este caso no se lleva a cabo la conversión.

El signo del valor del resultado se lee de los bits 28 a 31 del acumulador 1. Si el estado lógico de los bits es "0", el signo es positivo. Si el estado lógico de los cuatro bits es "1", el signo es negativo.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Convertir entero doble (32 bits) en BCD" se utiliza la siguiente sintaxis:

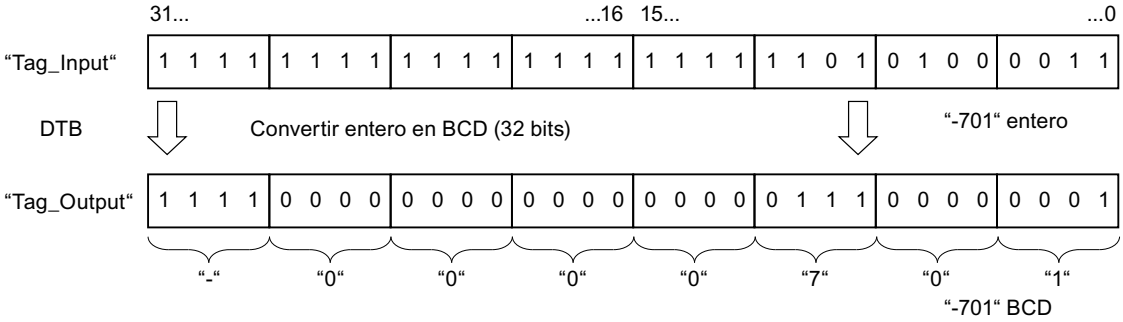
```
DTB
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero doble en el acumulador 1.
DTB	// Convertir entero doble en BCD
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

DTR: Convertir entero doble (32 bits) en número en coma flotante

Descripción

La instrucción "Convertir entero doble (32 bits) en número en coma flotante" convierte el valor que está en el acumulador 1 en un número en coma flotante. La instrucción interpreta el valor que debe convertirse como un entero doble.

Puesto que un entero doble es más exacto que un número en coma flotante, el resultado de la conversión se redondea al siguiente número representable.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Convertir entero doble (32 bits) en número en coma flotante" se utiliza la siguiente sintaxis:

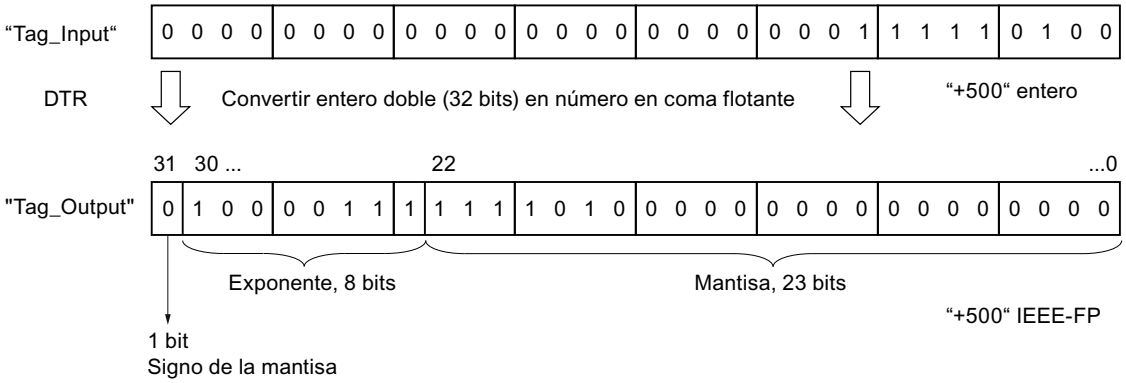
```
DTR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero doble en el acumulador 1.
DTR	// Convertir entero doble en número en coma flotante.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

INVI: Generar complemento a uno de un entero (16 bits)

Descripción

La instrucción "Generar complemento a uno de un entero (16 bits)" niega bit por bit el valor que se encuentra en la palabra derecha del acumulador 1.

Durante la ejecución, la instrucción invierte el estado lógico de cada uno de los bits de la palabra derecha del acumulador 1. Los unos son sustituidos por ceros y viceversa.

El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

Sintaxis

Para la instrucción "Generar complemento a uno de un entero (16 bits)" se utiliza la siguiente sintaxis:

```
INVI
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
INVI	// Generar un complemento a 1
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	0011	1010	1110
Tag_Output	1001	1100	0101	0001

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

INVD: Generar complemento a uno de un entero doble (32 bits)

Descripción

La instrucción "Generar complemento a uno de un entero doble (32 bits)" niega bit por bit el valor que se encuentra en el acumulador 1.

Durante la ejecución, la instrucción invierte el estado lógico de cada uno de los bits del acumulador 1. Los unos son sustituidos por ceros y viceversa.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Generar complemento a uno de un entero doble (32 bits)" se utiliza la siguiente sintaxis:

```
INVD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
INVD	// Generar un complemento a 1
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	1111	1000	1100	0110	0011	1010	1110
Tag_Output	1001	0000	0111	0011	1001	1100	0101	0001

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

NEGI: Negar entero (16 bits)

Descripción

La instrucción "Negar entero (16 bits)" invierte el signo del valor que está en la palabra derecha del acumulador 1 por una formación de complemento a dos. La instrucción interpreta el valor del acumulador 1 como un entero de 16 bits.

La ejecución de la instrucción es equivalente a una multiplicación por "-1". El resultado de la instrucción se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

La instrucción "Negar entero (16 bits)" influye en los bits de estado CC 0, CC 1, OV y OS. La tabla siguiente muestra cómo la instrucción influye en los bits de estado en función del resultado:

Resultado:	CC 1	CC 0	OV	OS
de +1 a +32 767	1	0	0	-
0	0	0	0	-
de -1 a -32 767	0	1	0	-
(-) 32 768	0	1	1	1

Sintaxis

Para la instrucción "Negar entero (16 bits)" se utiliza la siguiente sintaxis:

```
NEGI
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
NEGI	// Negar entero de 16 bits
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1101	0011	1000
Tag_Output	1010	0010	1100	1000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

NEGD: Negar entero doble (32 bits)

Descripción

La instrucción "Negar entero doble (32 bits)" invierte el signo del valor que se encuentra en el acumulador 1 por una formación de complemento a dos. La instrucción interpreta el valor del acumulador 1 como un entero doble.

La ejecución de la instrucción es equivalente a una multiplicación por "-1". El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La instrucción "Negar entero doble (32 bits)" influye en los bits de estado CC 0, CC 1, OV y OS. La tabla siguiente muestra cómo la instrucción influye en los bits de estado en función del resultado:

Resultado	CC 1	CC 0	OV	OS
de +1 a +2 147 483 647	1	0	0	-
0	0	0	0	-
de -1 a -2 147 483 647	0	1	0	-
(-) 2 147 483 648	0	1	1	1

Sintaxis

Para la instrucción "Negar entero doble (32 bits)" se utiliza la siguiente sintaxis:

```
NEGD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
NEGD	// Negar entero doble de 32 bits.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Estado	Valor							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1111	0110	0100	0101	1101	0011	1000
Tag_Output	1010	0000	1001	1011	1010	0010	1100	1000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

NEGR: Negar número en coma flotante

Descripción

La instrucción "Negar número en coma flotante" invierte el signo de la mantisa y es equivalente a una multiplicación por "-1".

La instrucción interpreta el valor del acumulador 1 como un número en coma flotante. El bit 31 del número en coma flotante contiene el signo de la mantisa.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La instrucción no influye en los bits de estado.

Sintaxis

Para la instrucción "Negar número en coma flotante" se utiliza la siguiente sintaxis:

```
NEGR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
NEGR	// Negar número en coma flotante.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Input	1.5E+02
Tag_Output	-1.5E+02

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

CAW: Invertir orden de los bytes en la palabra derecha del acumulador 1

Descripción

La instrucción "Invertir orden de los bytes en la palabra derecha del acumulador 1" invierte el orden de los dos bytes derechos en la palabra derecha del acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor B	Valor D	Valor C

El resultado de la instrucción se guarda en la palabra derecha del acumulador 1. Los bytes de la palabra izquierda del acumulador 1 no se ven influidos por la instrucción y no cambian.

Sintaxis

Para la instrucción "Invertir orden de los bytes en la palabra derecha del acumulador 1" se utiliza la siguiente sintaxis:

```
CAW
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador.
CAW	// Invertir orden de los bytes en la palabra derecha del acumulador 1.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Input	0000	0000	1111	1111
Tag_Output	1111	1111	0000	0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

CAD: Invertir orden de los bytes en todo el acumulador 1

Descripción

La instrucción "Invertir orden de los bytes en todo el acumulador 1" invierte el orden de los bytes en el acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor D	Valor C	Valor B	Valor A

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Invertir orden de los bytes en todo el acumulador 1" se utiliza la siguiente sintaxis:

```
CAD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Input"                // Cargar el valor del operando en el acumulador 1.
CAD                          // Invertir orden de los bytes del acumulador 1.
T "Tag_Output"               // Transferir el resultado al operando "Tag_Output".
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Input	1111	0000	0000	1111	0000	0000	1111	1111
Tag_Output	1111	1111	0000	0000	0000	1111	1111	0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RND: Redondear número

Descripción

La instrucción "Redondear número" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un número en coma flotante y lo redondea al siguiente entero.

Si el número en coma flotante se encuentra exactamente entre un número par y uno impar, se selecciona el número par como resultado.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Redondear número" se utiliza la siguiente sintaxis:

```
RND
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
RND	// Redondear un número en coma flotante a un entero.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	101.5	-101.5
Tag_Output	102	-102

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un número en coma flotante y devuelve como resultado su parte entera.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Truncar a entero" se utiliza la siguiente sintaxis:

```
TRUNC
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
TRUNC	// Convertir el número en coma flotante en un entero del tipo de datos DINT.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	101.5	-101.5
Tag_Output	101	-101

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RND+: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un número en coma flotante y lo redondea al siguiente entero superior. El resultado de la instrucción es un número del tipo de datos DINT, mayor o igual al número en coma flotante que se ha convertido.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero superior" se utiliza la siguiente sintaxis:

```
RND+
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
RND+	// Redondear el número en coma flotante al siguiente entero superior.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	100.5	-100.5
Tag_Output	101	-100

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RND-: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un número en coma flotante y lo redondea al siguiente entero inferior. El resultado de la instrucción es un número del tipo de datos DINT, menor o igual al número en coma flotante que se ha convertido.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero inferior" se utiliza la siguiente sintaxis:

```
RND-
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
RND-	// Redondear el número en coma flotante al siguiente entero inferior.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	100.5	-100.5
Tag_Output	100	-101

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Control del programa

Operaciones de salto

Etiqueta

Descripción

Una etiqueta sirve para identificar el punto del programa en el que debe continuar la ejecución tras un salto. El nombre de la etiqueta puede comprender un máximo de 128 letras, cifras o caracteres de subrayado.

La etiqueta y la instrucción en la que se indica la etiqueta como destino del salto deben encontrarse en el mismo bloque. El nombre de una etiqueta debe ser unívoca en el bloque. A toda etiqueta se puede acceder desde distintas posiciones. Se pueden declarar como máximo 256 etiquetas.

Es posible saltar hacia delante y hacia atrás.

Sintaxis

Para la etiqueta se utiliza la siguiente sintaxis:

<etiqueta>:

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JC MyLABEL	// Si el resultado de la consulta es "1", saltar a la etiqueta "MyLABEL" y continuar allí el procesamiento del programa
	// Si el resultado de la consulta es "0", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output".
MyLABEL: A "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Consultar si el operando "Tag_Input_4" está a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JU: Salto incondicional

Descripción

La instrucción "Salto incondicional" interrumpe la ejecución lineal del programa y continuarla en el punto marcado por la etiqueta indicada. Esta instrucción se ejecuta siempre, independientemente de las condiciones.

La instrucción "Salto incondicional" no influye en los bits de estado.

Sintaxis

Para la instrucción "Salto incondicional" se utiliza la siguiente sintaxis:

```
JU <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Salto incondicional":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JC MyLABEL_1	// Si el resultado de la consulta es "1", saltar a la etiqueta "MyLABEL_1" y continuar allí el procesamiento del programa

AWL	Explicación
	// Si el resultado de la consulta es "0", continuar con la ejecución lineal del programa.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output_1"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_1".
JU MyLABEL_2	// Saltar a la etiqueta "MyLABEL_2" y continuar allí con la ejecución del programa.
MyLABEL_1: L "Tag_Input_4"	// Etiqueta
	// Cargar el contenido del operando "Tag_Input_4" en el acumulador 1.
T "Tag_Output_2"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_2".
MyLABEL_2: A "Tag_Input_5"	// Etiqueta "MyLABEL_2"
	// Consultar si el operando "Tag_Input_5" está a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JC: Saltar si RLO = 1

Descripción

La instrucción "Saltar si RLO = 1" interrumpe la ejecución lineal del programa en función del resultado lógico y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "1". Si el RLO actual antes de la instrucción es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

La instrucción "Saltar si RLO = 1" pone el RLO a "1" tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 1" se utiliza la siguiente sintaxis:

```
JC <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Input_1"              // Cargar el primer valor de comparación.
L "Tag_Input_2"              // Cargar el segundo valor de comparación.
>I                           // Consultar si el valor del operando "Tag_Input_1" es
                             mayor que el valor del operando "Tag_Input_2".
JC MyLABEL                   // Si el RLO es "1", saltar a la etiqueta "MyLABEL" y
                             continuar allí la ejecución del programa.
                             // Si el RLO es "0", procesar la instrucción siguiente.
L "Tag_Input_3"              // Cargar el contenido del operando "Tag_Input_3" en
                             el acumulador 1.
T "Tag_Output_1"             // Cargar el contenido del acumulador 1 en el operando
                             "Tag_Output_1".
MyLABEL: L "Tag_Input_4"     // Continuar la ejecución del programa en este punto
                             en caso de haberse ejecutado un salto.
                             // Cargar el contenido del operando "Tag_Input_4" en
                             el acumulador 1.
T "Tag_Output_2"             // Transferir el contenido del acumulador 1 al
                             operando "Tag_Output_2".
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JCN: Saltar si RLO = 0

Descripción

La instrucción "Saltar si RLO = 0" interrumpe la ejecución lineal del programa en función del resultado lógico (RLO) y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "0". Si el RLO actual antes de la instrucción es "1", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

La instrucción "Saltar si RLO = 0" pone el RLO a "1" tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 0" se utiliza la siguiente sintaxis:

```
JCN <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 0":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y.
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y.
JCN MyLABEL	// Si el RLO es "0", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "1", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output".
MyLABEL: A "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Consultar si el operando "Tag_Input_4" es "1" y combinar el resultado lógicamente con Y con el RLO.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JCB: Saltar si RLO = 1 y guardar RLO

Descripción

La instrucción "Saltar si RLO = 1 y guardar RLO" interrumpe la ejecución lineal del programa en función del resultado lógico (RLO) y continuarla en el punto marcado por la etiqueta indicada. Paralelamente, el estado lógico del RLO actual se copia en el resultado binario (BR).

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "1". En este caso, la instrucción pone el resultado binario a "1".

Si el RLO actual antes de la instrucción es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente. En tal caso, la instrucción asigna el estado lógico "0" al resultado binario.

La instrucción "Saltar si RLO = 1 y guardar RLO" pone el RLO a "1" tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 1 y guardar RLO" se utiliza la siguiente sintaxis:

```
JCB <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 1 y guardar RLO":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JCB MyLABEL	// Copiar RLO actual en BR.
	// Si el RLO es "1", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "0", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output_1"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_1".
MyLABEL: L "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Cargar el contenido del operando "Tag_Input_4" en el acumulador 1.
T "Tag_Output_2"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JNB: Saltar si RLO = 0 y guardar RLO

Descripción

La instrucción "Saltar si RLO = 0 y guardar RLO" interrumpe la ejecución lineal del programa en función del resultado lógico (RLO) y continuarla en el punto marcado por la etiqueta indicada. Paralelamente, el estado del resultado lógico actual se copia en el resultado binario (BR).

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "0". En tal caso, la instrucción pone el resultado binario a "0".

Si el RLO actual antes de la instrucción es "1", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente. En tal caso, la instrucción asigna el estado lógico "1" al resultado binario.

La instrucción "Saltar si RLO = 0 y guardar RLO" pone el RLO a "1" tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 0 y guardar RLO" se utiliza la siguiente sintaxis:

```
JNB <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 0 y guardar RLO":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y.
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y.
JNB MyLABEL	// Copiar RLO actual en BR.
	// Si el RLO es "0", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "1", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output"	// Transferir el contenido del acumulador 1 al operando "Tag_Output".
MyLABEL: A "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Consultar si el operando "Tag_Input_4" es "1" y combinar el resultado lógicamente con Y con el RLO.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JBI: Saltar si BR = 1

Descripción

La instrucción "Saltar si BR = 1" interrumpe la ejecución lineal del programa en función del resultado binario y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado BR es "1". Si el bit de estado BR es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si BR = 1" se utiliza la siguiente sintaxis:

```
JBI <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si BR = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
SET                          // Poner el RLO a "1" al principio del bloque.
SAVE                         // Transferir el estado lógico del RLO al bit BR.
....                         // Cualquier programa
AN OV                        // Consultar el bit OV al final del bloque.
SAVE                         // En caso de desbordamiento, poner el bit BR a "0".
                             // Poner BR a "1" si no se ha producido un
                             desbordamiento.
JBI END                      // Si BR = "1", saltar a la etiqueta END.
                             // Si BR = "0", procesar la instrucción siguiente.
R "Tag_Output_1"             // Poner el operando "Tag_Output_1" a "0".
END: S "Tag_Output_2"        // Poner el operando "Tag_Output_2" a "1".
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JNBI: Saltar si BR = 0

Descripción

La instrucción "Saltar si BR = 0" interrumpe la ejecución lineal del programa en función del resultado binario (BR) y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado BR es "0". Si el bit de estado BR antes de la instrucción es "1", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si BR = 0" se utiliza la siguiente sintaxis:

```
JNBI <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si BR = 0":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                     Explicación
SET                     // Poner el RLO a "1" al principio del bloque.
SAVE                    // Transferir el estado lógico del RLO al bit BR.
....                    // Cualquier programa
AN OV                   // Consultar el bit OV al final del bloque.
SAVE                    // En caso de desbordamiento, poner el bit BR a "0".
```

AWL	Explicación
	// Poner BR a "1" si no se ha producido un desbordamiento.
JNBI END	// Si BR = "0", saltar a la etiqueta END.
	// Si BR = "1", procesar la instrucción siguiente.
S "Tag_Output"	// Poner el operando "Tag_Output" a "1".
END: BE	// Finalizar bloque

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JO: Saltar si OV = 1

Descripción

La instrucción "Saltar si OV = 1" interrumpe la ejecución lineal del programa en función del bit de estado OV y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado OV es "1". El bit de estado OV se activa si, p. ej., el resultado de un cálculo está fuera del rango permitido o si al comparar números en coma flotante hay un valor no válido.

Si el bit de estado OV es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si OV = 1" se utiliza la siguiente sintaxis:

```
JO <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si OV = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar valor
L "Tag_Value_2"	// Cargar valor
*I	// Multiplicar valores
JO OVER	// Si el estado lógico del bit de estado OV es "1", saltar a la etiqueta "OVER" y continuar allí la ejecución del programa.
	// Si el estado lógico del bit de estado OV es "0", procesar la instrucción siguiente.
T "Tag_Result"	// Transferir el resultado de la multiplicación al operando "Tag_Result".
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
OVER: SET	// Continuar la ejecución del programa en este punto si se cumple la condición (OV = "1").
	// Poner el RLO al estado lógico "1".
R "Tag_Output"	// Poner el operando "Tag_Output" a "0".
NEXT: A "MyTag_1"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_1" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_2"	// Consultar si el operando "MyTag_2" es "1" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO = 1, poner el operando "Tag_Output_2" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JOS: Saltar si OS = 1

Descripción

La instrucción "Saltar si OS = 1" interrumpe la ejecución lineal del programa en función del bit de estado OS y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado OS es "1". El bit de estado OS se activa siempre que un desbordamiento del rango numérico pone el bit de estado OV a "1". Al contrario que el bit de estado OV, el bit OS se mantiene activado aunque posteriormente haya un resultado que esté dentro del rango numérico permitido.

Si el bit de estado OS es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si OS = 1" se utiliza la siguiente sintaxis:

```
JOS <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si OS = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                         Explicación
L "Tag_Value_1"             // Cargar valor
L "Tag_Value_2"             // Cargar valor
*I                          // Multiplicar valores cargados.
                            // Guardar el producto en el acumulador 1.
L "Tag_Value_3"             // Cargar valor
+I                          // Sumar el valor cargado al producto.
                            // Guardar el producto en el acumulador 1.
L "Tag_Value_4"             // Cargar valor
-I                          // Restar el valor cargado de la suma calculada.
                            // Guardar el producto en el acumulador 1.
JOS OVER                    // En caso de desbordamiento en una de las tres
                            instrucciones precedentes, el bit de estado OS se pone
                            a "1".
                            // Si el estado lógico del bit de estado OS es "1" la
                            ejecución del programa continúa en la posición de la
                            etiqueta "OVER".
                            // Si el estado lógico del bit de estado OS es "0",
                            procesar la instrucción siguiente.
T "Tag_Result"              // Transferir el resultado de todo el cálculo al
                            operando "Tag_Result".
```

AWL	Explicación
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
OVER: SET	// Continuar la ejecución del programa en este punto si se cumple la condición (OV = "1").
	// Poner el RLO al estado lógico "1".
R "Tag_Output"	// Poner el operando "Tag_Output" a "0".
NEXT: A "MyTag_1"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_1" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_2"	// Consultar si el operando "MyTag_2" es "1" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO = 1, poner el operando "Tag_Output_2" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JZ: Saltar si el resultado es cero

Descripción

La instrucción "Saltar si el resultado es cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico de los bits de estado CC 0 y CC 1 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el acumulador 1 tiene el valor "0".
- Al ejecutar la instrucción "Sumar enteros (16 bits)" (+I) o "Sumar enteros dobles (32 bits)" (+D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética (aritmética de punto flotante) con desbordamiento se produce un rebasamiento inferior gradual del rango permitido.
- El contenido del acumulador 2 es igual al contenido del acumulador 1 después de ejecutar una instrucción de comparación.

- El contenido del acumulador 1 es cero después de ejecutar una operación lógica con palabras.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es "0".

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es cero" se utiliza la siguiente sintaxis:

```
JZ <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando "Tag_Value".
SRW 1	// Desplazar el contenido de la palabra derecha del acumulador 1 una posición a la derecha.
JZ ZERO	// Si el estado lógico del último bit desplazado hacia fuera es "0", saltar a la etiqueta "OVER" y continuar allí el procesamiento del programa
	// Si el estado lógico del último bit desplazado hacia fuera es "1", procesar la instrucción siguiente.
L "MyTag_1"	// Cargar el valor del operando "MyTag_1" en el acumulador 1.
INC 1	// Sumar el valor "1" al byte derecho del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
ZERO: L "MyTag_2"	// Etiqueta "ZERO"
	// Cargar el valor del operando "MyTag_2" en el acumulador 1.
INC 1	// Sumar el valor "1" al byte derecho del acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".
NEXT: A "MyTag_3"	// Etiqueta "NEXT"

AWL	Explicación
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JN: Saltar si el resultado no es cero

Descripción

La instrucción "Saltar si el resultado no es cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico de los bits de estado CC 0 y CC 1 es distinto. Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es distinto de cero.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, *I, +D, -D, *D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, *I, /I, +D, -D, *D, /D, NEGI, NEGD) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango positivo o negativo.
- El contenido del acumulador 2 es distinto al contenido del acumulador 1 después de ejecutar una instrucción de comparación.
- El contenido del acumulador 1 no es cero después de ejecutar una operación lógica con palabras.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es "1".

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado no es cero" se utiliza la siguiente sintaxis:

```
JN <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado no es cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Valu_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
XOW	// Combinar lógicamente con O-exclusiva
JN NOZERO	// Cuando el valor del acumulador 1 no es cero, se salta a la etiqueta "NOZERO" donde se continúa ejecutando el programa.
	// Cuando el valor en el acumulador 1 es igual a cero se ejecuta la siguiente instrucción.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
NOZERO: AN "MyTag_2	// Etiqueta "NOZERO"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1".
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JP: Saltar si el resultado es mayor que cero

Descripción

La instrucción "Saltar si el resultado es mayor que cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 0 es "0" y el del bit de estado CC 1 es "1". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es mayor que cero.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (*I, /I, *D, /D) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango positivo.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es mayor que el contenido del acumulador 1.
- Tras ejecutar una operación lógica con palabras, el contenido del acumulador 1 no es cero.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es "1".

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es mayor que cero" se utiliza la siguiente sintaxis:

```
JP <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es mayor que cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JP POSITIVE	// Si el valor del resultado en el acumulador 1 es positivo, saltar a la etiqueta "POSITIVE" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es negativo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
POSITIVE: AN "MyTag_2	// Etiqueta "POSITIVE"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JM: Saltar si el resultado es menor que cero

Descripción

La instrucción "Saltar si el resultado es menor que cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 0 es "1" y el del bit de estado CC 1 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es menor que cero.
- Al ejecutar una instrucción aritmética con enteros (*I, *D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D, NEGI, NEGD) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango negativo.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es menor que el contenido del acumulador 1.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es menor que cero" se utiliza la siguiente sintaxis:

```
JM <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es menor que cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JM NEGATIVE	// Si el valor del resultado en el acumulador 1 es negativo, saltar a la etiqueta "NEGATIVE" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es positivo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
NEGATIVE: AN "MyTag_2	// Etiqueta "NEGATIVE"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JPZ: Saltar si el resultado es mayor o igual a cero

Descripción

La instrucción "Saltar si el resultado es mayor o igual a cero" interrumpe la ejecución lineal del programa en función del bit de estado CC 0 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 0 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es mayor o igual que cero.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (*I, /I, *D, /D) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética (aritmética de punto flotante) con desbordamiento se produce un rebasamiento inferior gradual del rango permitido.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es mayor o igual al contenido del acumulador 1.
- Se ha ejecutado una instrucción de operación lógica por palabras.
- Se ha ejecutado una instrucción de desplazamiento.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es mayor o igual a cero" se utiliza la siguiente sintaxis:

```
JPZ <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es mayor o igual a cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JPZ REGULAR	// Si el valor del resultado en el acumulador 1 es positivo, saltar a la etiqueta "REGULAR" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es negativo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
REGULAR: AN "MyTag_2	// Etiqueta "REGULAR"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JMZ: Saltar si el resultado es menor o igual a cero

Descripción

La instrucción "Saltar si el resultado es menor o igual a cero" interrumpe la ejecución lineal del programa en función del bit de estado CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 1 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es menor o igual que cero.
- Al ejecutar una instrucción aritmética con enteros (+I, *I, +D, *D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D, NEGI, NEGD) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética (aritmética de punto flotante) con desbordamiento se produce un rebasamiento inferior gradual del rango permitido.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es menor o igual al contenido del acumulador 1.
- Tras ejecutar una operación lógica con palabras, el valor del acumulador 1 es cero.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es cero.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es menor o igual a cero" se utiliza la siguiente sintaxis:

```
JMZ <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es menor o igual a cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JMZ MyLABEL	// Si el valor del resultado en el acumulador 1 es negativo, saltar a la etiqueta "MyLABEL" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es positivo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
MyLABEL: AN "MyTag_2	// Etiqueta "MyLABEL"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si RLO="1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JUO: Saltar si el resultado no es válido

Descripción

La instrucción "Saltar si el resultado no es válido" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico de los bits de estado CC 0 y CC 1 es "1". Esto sucede cuando se cumple una de las condiciones siguientes:

- En una instrucción matemática (/I, /D, MOD) se divide entre cero.
- En la ejecución de una instrucción matemática con números en coma flotante se produce un desbordamiento y el resultado es un número en coma flotante no válido.
- En la ejecución de una instrucción de comparación con números en coma flotante se ha empleado un número en coma flotante no válido o ha dado como resultado.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado no es válido" se utiliza la siguiente sintaxis:

```
JUO <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado no es válido":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
/I	// Dividir el valor del operando "Tag_Value_1" por el valor del operando "Tag_Value_2".
JUO ERROR	// En una división por cero, saltar a la etiqueta "ERROR" y continuar allí el procesamiento del programa
	// En otro caso, procesar la instrucción siguiente.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando "Tag_Result".

AWL	Explicación
A "MyTag_1"	// Consultar si el operando "MyTag_1" es "1" y combinar el resultado lógicamente con Y.
R "MyTag_1"	// Si el RLO es "1", poner el operando "MyTag_1" a "0".
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
ERROR: AN "MyTag_1	// Etiqueta "ERROR"
	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "MyTag_1"	// Si el RLO es "1", poner el operando "MyTag_1" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

JL: Definir lista de saltos

Descripción

La instrucción "Definir lista de saltos" permite programar una lista compuesta por varias entradas de la instrucción "Salto incondicional" (JU). La lista empieza inmediatamente después de la instrucción "Definir lista de saltos" y puede contener un máximo de 255 entradas. La numeración de las funciones de salto de la lista empieza por cero. La lista de saltos debe programarse sin huecos. El final de la lista de saltos se marca con una etiqueta que se indica en la instrucción "Definir lista de saltos".

La función de salto de la lista que se ejecutará depende del valor del byte derecho del acumulador 1. Si, p. ej., en el acumulador 1 hay el valor "0", se ejecutará la primera función de salto. Si el valor del acumulador 1 es "1", se ejecutará la segunda función de salto. Si el valor del acumulador "1" es mayor que el número de entradas de la lista, la instrucción "Definir lista de saltos" remite al final de la lista.

La instrucción "Distribuidor de saltos" se ejecuta independientemente de las condiciones y no influye en los bits de estado.

Sintaxis

La instrucción "Definir lista de saltos" se programa siguiendo el siguiente esquema general:

```
L <número de salto>
JL <fin_etiqueta>
JU <etiqueta>
JU <etiqueta>
.... <etiqueta>
<fin_etiqueta>:
```

El operando "número de salto" contiene el número de la instrucción "Salto incondicional" (JU) que debe ejecutarse. Este número tiene que cargarse en el acumulador 1. Con la etiqueta "fin_etiqueta" se marca el fin de la lista de saltos.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el número de salto en el acumulador 1.
JL END	// Principio de la lista de saltos
JU MyLABEL_1	// Si el valor del acumulador 1 es "0", ejecutar y saltar a la etiqueta "MyLABEL_1".
JU MyLABEL_2	// Si el valor del acumulador 1 es "1", ejecutar y saltar a la etiqueta "MyLABEL_2".
JU MyLABEL_3	// Ejecutar con un valor "2" en el acumulador 1 y saltar a la etiqueta "MyLABEL_3".
END: L "MyTag_1"	// Fin de la lista de saltos
	// Cargar el contenido del operando "MyTag_1" en el acumulador 1.
L "MyTag_2"	// Cargar el contenido del operando "MyTag_2" en el acumulador 1.
+I	// Sumar valores
T "Tag_Output_2"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_2".
MyLABEL_1: ...	// Etiqueta "MyLABEL_1"
....	// Cualquier programa
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
MyLABEL_2: ...	// Etiqueta "MyLABEL_2"
....	// Cualquier programa
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
MyLABEL_3: ...	// Etiqueta "MyLABEL_3"
....	// Cualquier programa

AWL	Explicación
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
NEXT: ...	// Etiqueta "NEXT"

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

LOOP: Bucle

Descripción

La instrucción "Bucle" permite programar bucles.

La instrucción interpreta la palabra derecha del acumulador 1 como un entero de 16 bits sin signo en el rango de 0 a 65 535. Durante la ejecución, la instrucción decrementa primero en uno el contenido del acumulador 1. Si el valor del acumulador 1 no es cero después de decrementar, se ejecuta el salto hasta la etiqueta indicada. Si el valor es cero, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente. Por consiguiente, el valor del acumulador 1 determina el número de bucles que se ejecutarán. Dicho número debe guardarse en un contador de bucles.

La instrucción "Bucle" no influye en los bits de estado.

> ⚠ **ADVERTENCIA**
>
> Si el contador de bucles no está preasignado o lo está con el valor "0", el contador de bucles se decrementa continuamente hacia el negativo, produciéndose un bucle sin fin que puede provocar el estado operativo STOP de la CPU.

Sintaxis

La instrucción "Bucle" se programa siguiendo el siguiente esquema general:

```
L <número>
<etiqueta>:
T <contador de bucles>
... cualquier programa
```

```
L <contador de bucles>
LOOP <etiqueta>
```

El operando "número" contiene el número de bucles ejecutados. El operando "contador de bucles" contiene el número de bucles pendientes de ser ejecutados. Al final del bucle, el contenido del contador de bucles se carga en el acumulador 1 y es decrementado en uno por la instrucción. El salto hasta la etiqueta al principio del bucle se ejecuta si el valor del acumulador no es cero después de decrementar.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el número de ejecuciones de bucle pendientes en la palabra derecha del acumulador 1.
START: T "Tag_Counter"	// Principio de la lista de saltos
	// Transferir el contenido del acumulador 1 al contador de bucles.
L "MyTag_1"	// Cargar el valor del operando "MyTag_1".
L "MyTag_2"	// Cargar el valor del operando "MyTag_2".
*D	// Multiplicar valores
T "MyTag_1"	// Transferir el resultado de la multiplicación al operando "MyTag_1".
L "Tag_Counter"	// Cargar el contenido del contador de bucles en el acumulador 1.
LOOP START	// Decrementar en uno el valor del acumulador 1.
	// Si el valor es distinto de cero, saltar al principio del bucle.
	// Si el valor es cero, procesar la instrucción siguiente.
L "MyTag_2"	// Cargar el valor del operando "MyTag_2".
L 100	// Cargar el valor 100.
>I	// Comparar si el valor del operando "MyTag_2" es mayor que 200.
= "MyTag_3"	// Escribir el resultado de la comparación en el operando "MyTag_3".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Bloques de datos

OPN: Abrir bloque de datos en el registro DB

Descripción

La instrucción "Abrir bloque de datos en el registro DB" permite abrir un bloque de datos global (DB). El número del bloque de datos se transmite al registro del DB. Los siguientes comandos de DB acceden a los bloques correspondientes en función de los contenidos del registro.

La instrucción "Abrir bloque de datos en el registro DB" se ejecuta independientemente de las condiciones y no influye en el resultado lógico ni en los contenidos de los acumuladores.

Sintaxis

Para la instrucción "Abrir bloque de datos en el registro DB" se utiliza la siguiente sintaxis:

```
OPN <bloque de datos>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos en el registro DB":

Operando	Declaración	Tipo del bloque de datos	Descripción
<Bloque de datos>	-	DB	Bloque de datos que se abrirá.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos y transferir número del bloque al registro DB.
L %DBW0	// Cargar palabra de datos DBW0 del bloque de datos global abierto en acumulador 1.
T "MyTag"	// Transferir el contenido del acumulador 1 al operando "MyTag_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Bases de AWL (Página 2753)

Direcciones totalmente especificadas en AWL (Página 229)

Direcciones parcialmente especificadas en AWL (Página 230)

OPNDI: Abrir bloque de datos en el registro DI

Descripción

La instrucción "Abrir bloque de datos en el registro DI" permite abrir un bloque de datos cualquiera. El número del bloque de datos se transmite al registro del DI. Los siguientes comandos del DI acceden a los bloques correspondientes en función de los contenidos del registro.

Para el direccionamiento simbólico de un parámetro formal local de la interfaz de bloque, acceda siempre al bloque de datos cuyo número haya indicado al efectuar la llamada de bloque.

La instrucción "Abrir bloque de datos en el registro DI" se ejecuta independientemente de las condiciones y no influye en el resultado lógico ni en los contenidos de los acumuladores.

Sintaxis

Para la instrucción "Abrir bloque de datos en el registro DI" se utiliza la siguiente sintaxis:

```
OPNDI <Bloque de datos>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos en el registro DI":

Operando	Declaración	Tipo del bloque de datos	Descripción
<Bloque de datos>	-	DI	Bloque de datos que se abrirá.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPNDI "DataBlock"	// Abrir bloque de datos y transferir número del bloque de datos al registro DI.
L %DIW0	// Cargar palabra de datos DIW0 del bloque de datos abierto en acumulador 1.
T "MyTag"	// Transferir el contenido del acumulador 1 al operando "MyTag".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Bases de AWL (Página 2753)

Direcciones totalmente especificadas en AWL (Página 229)

Direcciones parcialmente especificadas en AWL (Página 230)

CDB: Intercambiar registros de bloques de datos

Descripción

La instrucción "Intercambiar registros de bloques de datos" cambia los contenidos de los registros de los bloques de datos. La instrucción se ejecuta independientemente de las condiciones y no influye en los bits de estado.

Sintaxis

Para la instrucción "Intercambiar registros de bloques de datos" se utiliza la siguiente sintaxis:

```
CDB
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global y transferir número del bloque de datos al registro DB.
OPNDI "DataBlock"	// Abrir bloque de datos y transferir número del bloque de datos al registro DI.
CDB	// Intercambiar registros de bloques de datos
	// El registro DB remite a "DataBlock" y el registro DI remite a "GlobalDataBlock".
L %DIW0	// Cargar DW0 de "GlobalDataBlock" en el acumulador 1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L DBLG: Cargar longitud de un bloque de datos global en acumulador 1

Descripción

La instrucción "Cargar longitud de un bloque de datos global en acumulador 1" carga en el acumulador 1 la longitud de un bloque de datos global abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos global mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1. La longitud del bloque de datos es equivalente al número de bytes de datos.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar longitud de un bloque de datos global en acumulador 1" se utiliza la siguiente sintaxis:

```
L DBLG
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global y transferir número del bloque de datos al registro DB.
L DBLG	// Cargar longitud de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
<D	// Comparar si la longitud del bloque de datos es menor que el valor del operando "MyTag_1".
= "MyTag_2"	// Escribir el resultado de la consulta en el operando "MyTag_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L DBNO: Cargar número de un bloque de datos global en acumulador 1

Descripción

La instrucción "Cargar número de un bloque de datos global en acumulador 1" carga en el acumulador 1 el número de un bloque de datos global abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos global mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar número de un bloque de datos global en acumulador 1" se utiliza la siguiente sintaxis:

```
L DBNO
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global y transferir número del bloque de datos al registro DB.
L DBNO	// Cargar número de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
==I	// Comparar si el número del bloque de datos es igual al valor del operando "MyTag_1".
= "MyTag_2"	// Escribir el resultado de la consulta en el operando "MyTag_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L DILG: Cargar longitud de un bloque de datos de instancia en acumulador 1

Descripción

La instrucción "Cargar longitud de un bloque de datos de instancia en acumulador 1" carga en el acumulador 1 la longitud de un bloque de datos de instancia abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos de instancia mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1. La longitud del bloque de datos es equivalente al número de bytes de datos.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar longitud de un bloque de datos de instancia en acumulador 1" se utiliza la siguiente sintaxis:

```
L DILG
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPNDI "InstanceDataBlock"	// Abrir bloque de datos y transferir número del bloque de datos al registro DI.
L DILG	// Cargar longitud de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
<I	// Comparar si la longitud del bloque de datos es menor que el valor del operando "MyTag_1".
= "MyTag_2"	// Escribir el resultado de la consulta en el operando "MyTag_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

L DINO: Cargar número de un bloque de datos de instancia en acumulador 1

Descripción

La instrucción "Cargar número de un bloque de datos de instancia en acumulador 1" carga en el acumulador 1 el número de un bloque de datos de instancia abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos de instancia mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar número de un bloque de datos de instancia en acumulador 1" se utiliza la siguiente sintaxis:

```
L DINO
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPNDI "InstanceDataBlock"	// Abrir bloque de datos y transferir número del bloque de datos al registro DI.
L DINO	// Cargar número de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
==I	// Comparar si el número del bloque de datos es igual al valor del operando "MyTag_1".
= "MyTag_2"	// Asignar el resultado de la consulta al operando "MyTag_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Bloques lógicos

BE: Fin de bloque

Descripción

La instrucción "Fin de bloque" finaliza el procesamiento del bloque actual y pasa al punto del programa en el que se llama el bloque. El procesamiento del programa continúa con la instrucción que se encuentra justo después de la llamada del bloque.

La instrucción "Fin de bloque" se ejecuta siempre, independientemente de las condiciones. Si el procesamiento de la instrucción se salta debido a una instrucción de salto, la ejecución actual del programa no finaliza, sino que continúa en el destino del salto dentro del bloque.

Sintaxis

Para la instrucción "Fin de bloque" se utiliza la siguiente sintaxis:

```
BE
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
SIN	// Calcular el valor del seno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando "Tag_Result".
BE	// Finalizar bloque

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

BEC: Fin de bloque condicional

Descripción

La instrucción "Fin de bloque condicional" finaliza el procesamiento del bloque actual en función del resultado lógico (RLO) y pasa al punto del programa en el que se llama el bloque.

Si el RLO es "1", se ejecuta la instrucción. El bloque actual se finaliza y la ejecución del programa continúa en el bloque que efectúa la llamada. El procesamiento del programa continúa con la instrucción que se encuentra justo después de la llamada del bloque.

Si el RLO es "0" al procesar la instrucción, esta no se ejecuta. En este caso, la CPU pone el RLO a "1" y procesa la instrucción siguiente.

Si el procesamiento de la instrucción se salta debido a una instrucción de salto, la ejecución actual del programa no finaliza, sino que continúa en el destino del salto dentro del bloque.

Sintaxis

Para la instrucción "Fin de bloque condicional" se utiliza la siguiente sintaxis:

```
BEC
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A "Tag_Input"	// Consultar si el operando "Tag_Input" es "1" y combinar el resultado lógicamente con Y con el RLO.
BEC	// Si la condición se cumple (RLO = "1"), finalizar el bloque.
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

BEU: Fin de bloque incondicional

Descripción

La instrucción "Fin de bloque incondicional" finaliza el procesamiento del bloque actual y pasa al punto del programa en el que se llama el bloque. El procesamiento del programa continúa con la instrucción que se encuentra justo después de la llamada del bloque.

La instrucción "Fin de bloque incondicional" se ejecuta siempre, independientemente de las condiciones, y se puede programar varias veces dentro de un bloque. Si el procesamiento de la instrucción se salta debido a una instrucción de salto, la ejecución actual del programa no finaliza, sino que continúa en el destino del salto dentro del bloque.

Sintaxis

Para la instrucción "Fin de bloque incondicional" se utiliza la siguiente sintaxis:

```
BEU
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A "Tag_Input"	// Consultar si el operando "Tag_Input" es "1" y combinar el resultado lógicamente con Y con el RLO.
JC NEXT	// Si la condición se cumple (RLO = "1"), continuar el procesamiento del programa a partir de la etiqueta "NEXT".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
BEU	// Finalizar bloque
NEXT: T "Tag_Result"	// Etiqueta "NEXT"
	// Transferir el contenido del acumulador 1 al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

CALL: Llamar bloque

Descripción

La instrucción "Llamar bloque" llama los siguientes tipos de bloque en el programa:

- Funciones
- Bloques de función

La instrucción "Llamar bloque" se ejecuta siempre, independientemente de las condiciones. Tras ejecutar la instrucción, el procesamiento del programa continúa en el bloque llamado.

Existe la posibilidad de suministrar datos a un bloque llamado. Los datos se transfieren mediante los parámetros del bloque. Los parámetros del bloque llamado se listan en el bloque que efectúa la llamada después de la instrucción de llamada. Es posible asignarles los parámetros actuales necesarios. Al ejecutar la instrucción "Llamar bloque", los datos se transfieren al bloque llamado. Con la transmisión de los datos se modifican los contenidos tanto de la palabra de estado como del registro de direcciones y del de los bloques de datos.

Los parámetros de un bloque de función que no reciben valores nuevos conservan su valor actual. Al llamar funciones deben suministrarse valores a todos los parámetros. Si un bloque llamado no tiene parámetros, no se muestra la lista de parámetros.

Si el bloque llamado requiere un bloque de datos de instancia, éste debe indicarse en la llamada, separado por una coma. El bloque de datos indicado debe haberse creado antes de la llamada.

Una vez procesado el bloque llamado, la CPU vuelve al bloque que efectúa la llamada y continúa con el procesamiento de dicho bloque después de la instrucción de llamada.

Nota

La instrucción "Llamar bloque" solo puede utilizarse si está activada la propiedad del bloque "Alimentación de parámetros a través de registros".

Sintaxis

Al llamar una función, se utiliza la siguiente sintaxis para la instrucción "Llamar bloque":

```
CALL <FC>
```

Al llamar un bloque de función, se utiliza la siguiente sintaxis para la instrucción "Llamar bloque":

```
CALL <FB>, <FB_DB>
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL "MyFunction"	// Llamar la función "MyFunction"
Input_1 := "Tag_Input_1",	// Asignar parámetros actuales
Input_2 := "Tag_Input_2"	
Output_1 := "Tag_Output_1"	
Output_2 := "Tag_Output_1"	
CALL "MyFunctionBlock", "MyFB_DB"	// Llamar el bloque de función "MyFunctionBlock"
Value_1 := "Tag_Value_1"	// Asignar parámetros actuales
Value_2 := "Tag_Value_2"	
Output := "Tag_Output"	
CALL "LIMIT"	// Llamar la instrucción "Ajustar valor límite"
MN := "Tag_LowLimit"	// Asignar parámetros actuales
IN := "Tag_InputValue"	
MX := "Tag_HighLimit"	
OUT := "Tag_Output"	
CALL "CTU", "CTU_DB"	// Llamar el contador "Contador ascendente"
CU := "Tag_StartCTU"	// Asignar parámetros actuales
R := "Tag_ResetCounter"	
PV := "Tag_PresetValue"	
Q := "Tag_CounterStatus"	
CV := "Tag_CounterValue"	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Bases de AWL (Página 2753)

Direcciones totalmente especificadas en AWL (Página 229)

Direcciones parcialmente especificadas en AWL (Página 230)

CC: Llamada condicional

Descripción

La instrucción "Llamada condicional" llama en función del resultado lógico (RLO) funciones (FC) y bloques de función (FB) que no tienen parámetros ni bloques de datos de instancia.

La instrucción solo se ejecuta si el resultado lógico (RLO) actual antes de procesar la instrucción es "1". Tras ejecutar la instrucción, el procesamiento del programa continúa en el bloque llamado. Una vez procesado el bloque llamado, la CPU vuelve al bloque que efectúa la llamada y continúa con el procesamiento de dicho bloque después de la instrucción de llamada.

Al cambiar el bloque, el bit de estado OS se pone a "0". Los bits de estado CC 0, CC 1 y OV se ven influidos en función del bloque invocante.

Nota

La instrucción "Llamada condicional" solo puede utilizarse cuando está activada la propiedad del bloque "Alimentación de parámetros a través de registros". Sin embargo, de este modo se reduce el rendimiento.

La instrucción "Llamada condicional" no modifica el contenido de los acumuladores ni los registros de direcciones.

Si el RLO actual es "0", la instrucción y, por consiguiente, la llamada de bloque no se ejecutan y el RLO se pone a "1".

Sintaxis

Al llamar una función, se utiliza la siguiente sintaxis para la instrucción "Llamada condicional":

```
CC <FC>
```

Al llamar un bloque de función, se utiliza la siguiente sintaxis para la instrucción "Llamada condicional":

```
CC <FB>
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CC "MyFunction"	// Si la condición se cumple (RLO = "1"), llamar la función "MyFunction".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CC "MyFunctionBlock"	// Si la condición se cumple (RLO = "1"), llamar el bloque de función "MyFunctionBlock".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Sinopsis de las propiedades de bloques (Página 2608)

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

UC: Llamada incondicional

Descripción

La instrucción "Llamada incondicional" llama funciones (FC) y bloques de función (FB) que no tienen parámetros ni bloques de datos de instancia.

Esta instrucción se ejecuta independientemente de las condiciones. Tras ejecutar la instrucción, el procesamiento del programa continúa en el bloque llamado. Una vez procesado el bloque llamado, la CPU vuelve al bloque que efectúa la llamada y continúa con el procesamiento de dicho bloque después de la instrucción de llamada.

Al cambiar el bloque, el bit de estado OS se pone a "0". Los bits de estado CC 0, CC 1 y OV se ven influidos en función del bloque invocante.

Nota

La instrucción "Llamada incondicional" solo puede utilizarse cuando está activada la propiedad del bloque "Alimentación de parámetros a través de registros". Sin embargo, de este modo se reduce el rendimiento.

La instrucción "Llamada incondicional" no modifica el contenido de los acumuladores ni los registros de direcciones.

Sintaxis

Al llamar una función, se utiliza la siguiente sintaxis para la instrucción "Llamada incondicional":

```
UC <FC>
```

Al llamar un bloque de función, se utiliza la siguiente sintaxis para la instrucción "Llamada incondicional":

```
UC <FB>
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
UC "MyFunction"	// Llamar la función "MyFunction".
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
UC "MyFunctionBlock"	// Llamar el bloque de función "MyFunctionBlock".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Sinopsis de las propiedades de bloques (Página 2608)

Alimentación de parámetros a través de registros en la llamada de bloques en AWL (Página 2755)

Operaciones lógicas con palabras

AW: Operación lógica Y por palabras

Descripción

La instrucción "Operación lógica Y por palabras" combina lógicamente con Y por bits el valor de la palabra derecha del acumulador 1 con el valor de la palabra derecha del acumulador 2 o una constante concreta. El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no cambia.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 15 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

La tabla siguiente muestra cómo se forma el resultado en una operación lógica Y por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	0	0	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras":

```
AW <constante>
```

En las operaciones lógicas con el valor de la palabra derecha del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras":

```
AW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y por palabras":

Parámetro	Tipo de datos	Descripción
<constante>	WORD	Valor que se combina lógicamente con Y con el valor que se encuentra en la palabra derecha del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando en la palabra derecha
                             del acumulador 1.
AW W#16#F6B5                 // Combinar lógicamente con Y el valor de la palabra
                             derecha del acumulador 1 con la constante (W#16#F6B5).
                             // El resultado se guarda en la palabra derecha del
                             acumulador 1.
T "Tag_Result_1"             // Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"              // Cargar el valor del operando en la palabra derecha
                             del acumulador 1.
L "Tag_Value_3"              // Desplazar el contenido del acumulador 1 al
                             acumulador 2.
                             // Cargar el valor del operando "Tag_Value_3" en la
                             palabra derecha del acumulador 1.
AW                           // Combinar lógicamente con Y el valor de la palabra
                             derecha del acumulador 2 ("Tag_Value_2") con el valor
                             de la palabra derecha del acumulador 1
                             ("Tag_Value_3").
                             // El resultado se guarda en la palabra derecha del
                             acumulador 1.
T "Tag_Result_2"             // Transferir el resultado al operando "Tag_Result_2".
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Value_1	0101	1001	0011	1011
Constante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	0101	0000	0011	0001
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	0100	1000	0000	0010

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Activar el bit de estado en operaciones lógicas con palabras (Página 2563)

OW: Operación lógica O por palabras

Descripción

La instrucción "Operación lógica O por palabras" combina lógicamente con O por bits el valor de la palabra derecha del acumulador 1 con el valor de la palabra derecha del acumulador 2 o una constante concreta. El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no cambia.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 15 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de los dos bits que deben combinarse lógicamente es "0", el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras":

```
OW <constante>
```

En las operaciones lógicas con el valor de la palabra derecha del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras":

```
OW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O por palabras":

Parámetro	Tipo de datos	Descripción
<constante>	WORD	Valor que se combina lógicamente con O con el valor que se encuentra en la palabra derecha del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
OW W#16#F6B5	// Combinar lógicamente con O el valor de la palabra derecha del acumulador 1 con la constante (W#16#F6B5).
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en la palabra derecha del acumulador 1.
OW	// Combinar lógicamente con O el valor de la palabra derecha del acumulador 2 ("Tag_Value_2") con el valor de la palabra derecha del acumulador 1 ("Tag_Value_3").
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Value_1	0101	1001	0011	1011
Constante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1111	1111	1011	1111
Tag_Value_2	0110	1100	0010	1010

Operando	Valor			
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1111	1110	1011	1011

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Activar el bit de estado en operaciones lógicas con palabras (Página 2563)

XOW: Operación lógica O-exclusiva por palabras

Descripción

La instrucción "Operación lógica O-exclusiva por palabras" combina lógicamente con O-exclusiva por bits el valor de la palabra derecha del acumulador 1 con el valor de la palabra derecha del acumulador 2 o una constante concreta. El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no cambia.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 15 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente es diferente. Si el estado lógico de los dos bits que deben combinarse lógicamente es el mismo, el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O-exclusiva por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	0

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras":

```
XOW <constante>
```

En las operaciones lógicas con el valor de la palabra derecha del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras":

```
XOW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva por palabras":

Parámetro	Tipo de datos	Descripción
<constante>	WORD	Valor que se combina lógicamente con O-exclusiva con el valor que se encuentra en la palabra derecha del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
XOW W#16#F6B5	// Combinar lógicamente con O-exclusiva el valor de la palabra derecha del acumulador 1 con la constante (W#16#F6B5).
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en la palabra derecha del acumulador 1.
XOW	// Combinar lógicamente con O-exclusiva el valor de la palabra derecha del acumulador 2 ("Tag_Value_2") con el valor de la palabra derecha del acumulador 1 ("Tag_Value_3").
	// El resultado se guarda en la palabra derecha del acumulador 1.

AWL	Explicación
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Value_1	0101	1001	0011	1011
Constante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1010	1111	1000	1110
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1011	0110	1011	1001

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Activar el bit de estado en operaciones lógicas con palabras (Página 2563)

AD: Operación lógica Y por palabras dobles

Descripción

La instrucción "Operación lógica Y por palabras dobles" combina lógicamente con Y por bits el contenido del acumulador 1 con el contenido del acumulador 2 o el valor de una constante concreta. El resultado se guarda en el acumulador 1.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 31 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

La tabla siguiente muestra cómo se forma el resultado en una operación lógica Y por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	0	0	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras dobles":

```
AD <constante>
```

En las operaciones lógicas con el valor del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras dobles":

```
AD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y por palabras dobles":

Parámetro	Tipo de datos	Descripción
<constante>	DWORD	Valor que se combina lógicamente con Y con el valor que está en el acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                         Explicación
L "Tag_Value_1"             // Cargar el valor del operando en el acumulador 1.
AD DW#16#39C657AC           // Combinar lógicamente con Y el valor del acumulador
                            1 con la constante (DW#16#39C657AC).
                            // Guardar el resultado en el acumulador 1.
T "Tag_Result_1"            // Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"             // Cargar el valor del operando en el acumulador 1.
L "Tag_Value_3"             // Desplazar el contenido del acumulador 1 al
                            acumulador 2.
                            // Cargar el valor del operando "Tag_Value_3" en el
                            acumulador 1.
```

AWL	Explicación
AD	// Combinar lógicamente con Y el valor del acumulador 2 ("Tag_Value_2") con el valor del acumulador 1 ("Tag_Value_3").
	// Guardar el resultado en el acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Constante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0001	1001	0100	0100	0001	0101	0010	1000
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0010	0001	0100	0010	0101	0101	0010	1010

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Activar el bit de estado en operaciones lógicas con palabras (Página 2563)

OD: Operación lógica O por palabras dobles

Descripción

La instrucción "Operación lógica O por palabras dobles" combina lógicamente con O por bits el contenido del acumulador 1 con el contenido del acumulador 2 o el valor de una constante concreta. El resultado se guarda en el acumulador 1.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 31 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de los dos bits que deben combinarse lógicamente es "0", el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras dobles":

```
OD <constante>
```

En las operaciones lógicas con el valor del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras dobles":

```
OD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O por palabras dobles":

Parámetro	Tipo de datos	Descripción
<constante>	DWORD	Valor que se combina lógicamente con O con el valor que está en el acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                         Explicación
L "Tag_Value_1"             // Cargar el valor del operando en el acumulador 1.
OD DW#16#39C657AC           // Combinar lógicamente con O el valor del acumulador
                            1 con la constante (DW#16#39C657AC).
                            // Guardar el resultado en el acumulador 1.
T "Tag_Result_1"            // Transferir el resultado al operando "Tag_Result_1".
```

AWL	Explicación
L "Tag_Value_2"	// Cargar el valor del operando en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
OD	// Combinar lógicamente con O el valor del acumulador 2 ("Tag_Value_2") con el valor del acumulador 1 ("Tag_Value_3").
	// Guardar el resultado en el acumulador 1
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Constante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0111	1111	1110	0110	1101	1111	1011	1111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0111	1101	1100	0111	0101	1111	1110	1111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Activar el bit de estado en operaciones lógicas con palabras (Página 2563)

XOD: Operación lógica O-exclusiva por palabras dobles

Descripción

La instrucción "Operación lógica O-exclusiva por palabras dobles" combina lógicamente con O-exclusiva por bits el contenido del acumulador 1 con el contenido del acumulador 2 o el valor de una constante concreta. El resultado se guarda en el acumulador 1.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 31 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente es diferente. Si el estado lógico de los dos bits que deben combinarse lógicamente es el mismo, el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O-exclusiva por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	0

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras dobles":

```
XOD <constante>
```

En las operaciones lógicas con el valor del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras dobles":

```
XOD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva por palabras dobles":

Parámetro	Tipo de datos	Descripción
<constante>	DWORD	Valor que se combina lógicamente con O-exclusiva con el valor que está en el acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
XOD DW#16#39C657AC	// Combinar lógicamente con O-exclusiva el valor del acumulador 1 con la constante (DW#16#39C657AC).
	// Guardar el resultado en el acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
XOD	// Combinar lógicamente con O-exclusiva el valor del acumulador 2 ("Tag_Value_2") con el valor del acumulador 1 ("Tag_Value_3").
	// Guardar el resultado en el acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Constante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0110	0110	1010	0010	1100	1010	1001	0111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0101	1100	1000	0101	0000	1010	1000	0111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Activar el bit de estado en operaciones lógicas con palabras (Página 2563)

Desplazamiento y rotación

Desplazamiento

SSI: Desplazar con signo por palabras

Descripción

La instrucción "Desplazar con signo por palabras" desplaza bit por bit la palabra derecha (bits de 0 a 15) del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con el estado lógico del bit 15 (bit con signo en números INT). Los bits 16 a 31 del acumulador 1 no cambian.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 15.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 15, todos los bits de la palabra derecha del acumulador 1 se rellenarán con el estado lógico del bit 15.

Aunque el número de desplazamiento indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras":

```
SSI <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras":

```
SSI
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar con signo por palabras":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SSI 6	// Desplazar los bits 0 a 15 del acumulador 1 seis posiciones a la derecha.
	// Rellenar las posiciones liberadas con el estado del bit 15.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 3	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SSI	// Desplazar los bits 0 a 15 del acumulador 1 tres posiciones a la derecha.
	// Rellenar las posiciones liberadas con el estado del bit 15.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1111	1110	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	1011	1010	0101

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SSD: Desplazar con signo por palabras dobles

Descripción

La instrucción "Desplazar con signo por palabras dobles" desplaza bit por bit todo el contenido del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con el estado lógico del bit 31 (bit con signo en números DINT).

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 31.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 31, todos los bits del acumulador 1 se rellenarán con el estado lógico del bit 31.

Aunque el número de desplazamiento indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras dobles":

```
SSD <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras dobles":

```
SSD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar con signo por palabras dobles":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SSD 7	// Desplazar los bits 0 a 31 del acumulador 1 siete posiciones a la derecha.
	// Rellenar las posiciones liberadas con el estado del bit 31.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SSD	// Desplazar los bits 0 a 31 del acumulador 1 cuatro posiciones a la derecha.
	// Rellenar las posiciones liberadas con el estado del bit 31.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	1000	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	1111	0001	1110	1100	1000	1011	1010
Tag_Value_2	0010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	0010	1000	1010	0010	1001	1011	1100

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SLW: Desplazar a la izquierda por palabras

Descripción

La instrucción "Desplazar a la izquierda por palabras" desplaza bit por bit la palabra derecha (bits 0 a 15) del acumulador 1 hacia la izquierda. Las posiciones que quedan libres al desplazar se rellenan con ceros. Los bits 16 a 31 del acumulador 1 no cambian.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 15.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 15, todos los bits de la palabra derecha del acumulador 1 se rellenarán con ceros.

Aunque el número de desplazamiento indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras":

```
SLW <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras":

```
SLW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda por palabras":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SLW 5	// Desplazar los bits 0 a 15 del acumulador 1 cinco posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SLW	// Desplazar los bits 0 a 15 del acumulador 1 cuatro posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1010	0111	0110	0000
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	1101	0010	1011	0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SRW: Desplazar a la derecha por palabras

Descripción

La instrucción "Desplazar a la derecha por palabras" desplaza bit por bit la palabra derecha (bits 0 a 15) del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con ceros. Los bits 16 a 31 del acumulador 1 no cambian.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 15.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 15, todos los bits de la palabra derecha del acumulador 1 se rellenarán con ceros.

Aunque el número de desplazamiento indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras":

```
SRW <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante la palabra derecha del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras":

```
SRW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha por palabras":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SRW 6	// Desplazar los bits 0 a 15 del acumulador 1 seis posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SRW	// Desplazar los bits 0 a 15 del acumulador 1 cuatro posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	0000	0001	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	0101	1101	0010

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SLD: Desplazar a la izquierda por palabras dobles

Descripción

La instrucción "Desplazar a la izquierda por palabras dobles" desplaza bit por bit todo el contenido del acumulador 1 hacia la izquierda. Las posiciones que quedan libres al desplazar se rellenan con ceros.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 31.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 31, todos los bits del acumulador 1 se rellenarán con ceros.

Aunque el número de desplazamiento indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras dobles":

```
SLD <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras dobles":

```
SLD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda por palabras dobles":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SLD 5	// Desplazar los bits 0 a 31 del acumulador 1 cinco posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SLD	// Desplazar los bits 0 a 31 del acumulador 1 cuatro posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1110	1100	1000	1011	1010	0111	0110	0000
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	1000	1010	0010	1001	1011	1100	1101	0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SRD: Desplazar a la derecha por palabras dobles

Descripción

La instrucción "Desplazar a la derecha por palabras dobles" desplaza por bits todo el contenido del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con ceros.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 31.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 31, todos los bits del acumulador 1 se rellenarán con ceros.

Aunque el número de desplazamiento indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras dobles":

```
SRD <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras dobles":

```
SRD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha por palabras dobles":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SRD 7	// Desplazar los bits 0 a 31 del acumulador 1 siete posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SRD	// Desplazar los bits 0 a 31 del acumulador 1 cuatro posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0000	0000	1011	1110	1100	1000	1011	1010
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	1010	1000	1010	0010	1001	1011	1100

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Rotación

RLD: Rotar a la izquierda por palabras dobles

Descripción

La instrucción "Rotar a la izquierda por palabras dobles" rota bit por bit todo el contenido del acumulador 1 hacia la izquierda. Durante la ejecución, la instrucción desplaza por bits los bits de 0 a 31 del acumulador 1 hacia la izquierda y rellena las posiciones que han quedado libres al desplazar con las posiciones de bit desplazadas.

Para indicar el número de posiciones de bit que se rotarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de rotación indicado es 32, el contenido del acumulador 1 no cambia. Si el valor es mayor que 32, el número de rotación se calcula con una división módulo entre 32. Si, p. ej., el número de rotación es 34, los bits del acumulador 1 se rotarán 2 posiciones de bit.

Aunque el número de rotación indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de rotación se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Rotar a la izquierda por palabras dobles":

```
RLD <Número de rotación>
```

Si el número de rotación se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Rotar a la izquierda por palabras dobles":

```
RLD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda por palabras dobles":

Parámetro	Formato	Descripción
<Número de rotación>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se rotarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
RLD 4	// Rotar los bits 0 a 31 del acumulador 1 cuatro posiciones a la izquierda.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 6	// Cargar el número de rotación en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de rotación al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
RLD	// Rotar los bits 0 a 31 del acumulador 1 seis posiciones a la izquierda.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	**0101**	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	0110	0100	0101	1101	0011	1011	**0101**
Tag_Value_2	**1010**	**10**00	1010	0010	1001	1011	1100	1101
Tag_Result_2	0010	1000	1010	0110	1111	0011	01**10**	**1010**

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RRD: Rotar a la derecha por palabras dobles

Descripción

La instrucción "Rotar a la derecha por palabras dobles" rota bit por bit todo el contenido del acumulador 1 hacia la derecha. Durante la ejecución, la instrucción desplaza por bits los bits 0 a 31 del acumulador 1 hacia la derecha y rellena las posiciones que han quedado libres al desplazar con las posiciones de bit desplazadas.

Para indicar el número de posiciones de bit que se rotarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta en forma de entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OV a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de rotación indicado es 32, el contenido del acumulador 1 no cambia. Si el valor es mayor que 32, el número de rotación se calcula con una división módulo entre 32. Si, p. ej., el número de rotación es 34, los bits del acumulador 1 se rotarán 2 posiciones de bit.

Aunque el número de rotación indicado sea cero, la instrucción se ejecuta. El bit de estado CC 1 se pone a "0".

Sintaxis

Si el número de rotación se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Rotar a la derecha por palabras dobles":

```
RRD <Número de rotación>
```

Si el número de rotación se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Rotar a la derecha por palabras dobles":

```
RRD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha por palabras dobles":

Parámetro	Formato	Descripción
<Número de rotación>	Enteros positivos SINT, INT, DINT, USINT, UINT, UDINT	Número de posiciones de bit que se rotarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
RRD 4	// Rotar los bits 0 a 31 del acumulador 1 cuatro posiciones a la derecha.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 6	// Cargar el número de rotación en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de rotación al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
RRD	// Rotar los bits 0 a 31 del acumulador 1 seis posiciones a la derecha.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	**1011**
Tag_Result_1	**1011**	0101	1111	0110	0100	0101	1101	0011
Tag_Value_2	1010	1000	1010	0010	1001	1011	11**00**	**1101**
Tag_Result_2	**0011**	**01**10	1010	0010	1000	1010	0110	1111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RLDA: Rotar a la izquierda vía bit de estado CC 1

Descripción

La instrucción "Rotar a la izquierda vía bit de estado CC 1" desplaza el contenido del acumulador 1 un bit a la izquierda. La posición de bit que ha quedado libre al desplazar (0) se rellena con el estado lógico del bit de estado CC 1. El bit de estado CC 1 obtiene el estado lógico del bit desplazado (31).

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0".

Sintaxis

Para la instrucción "Rotar a la izquierda vía bit de estado CC 1" se utiliza la siguiente sintaxis:

```
RLDA
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
RLDA	// Rotar los bits 0 a 31 del acumulador 1 una posición a la izquierda.
	// Rellenar el bit 0 con el estado lógico del bit de estado CC 1.
	// Escribir el bit desplazado en el bit de estado CC 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	CC 1	Valor							
Tag_Value	1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	0	1011	1110	1100	1000	1011	1010	0111	0111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

RRDA: Rotar a la derecha vía bit de estado CC 1

Descripción

La instrucción "Rotar a la derecha vía bit de estado CC 1" desplaza el contenido del acumulador 1 un bit a la derecha. La posición de bit que ha quedado libre al desplazar (31) se rellena con el estado lógico del bit de estado CC 1. El bit de estado CC 1 obtiene el estado lógico del bit desplazado (0).

Esta instrucción se ejecuta independientemente del RLO. Pone los bits de estado CC 0 y OVa "0".

Sintaxis

Para la instrucción "Rotar a la derecha vía bit de estado CC 1" se utiliza la siguiente sintaxis:

```
RRDA
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
RRDA	// Rotar los bits 0 a 31 del acumulador 1 una posición a la derecha.
	// Rellenar el bit 31 con el estado lógico del bit de estado CC 1.
	// Escribir el bit desplazado en el bit de estado CC 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	CC 1	Valor							
Tag_Value	1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	0	1010	1111	1011	0010	0010	1110	1001	1101

El bit de estado CC 1 tiene el valor "1", ya que el bit 0 del operando "Tag_Value" también tiene el valor "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Otras instrucciones

Acumulador

TAK: Intercambiar contenido de los acumuladores 1 y 2

Descripción

La instrucción "Intercambiar contenido de los acumuladores 1 y 2" intercambia el contenido del acumulador 1 con el contenido del acumulador 2.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Intercambiar contenido de los acumuladores 1 y 2" se utiliza la siguiente sintaxis:

```
TAK
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.

AWL	Explicación
JC NEXT	// Si la condición se cumple (RLO = "1"), saltar a la etiqueta "NEXT".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
TAK	// Intercambiar contenido de los acumuladores 1 y 2.
NEXT: -I	// Etiqueta "NEXT"
	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Output"	// Transferir el contenido del acumulador 1 al operando "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

PUSH: Desplazar contenido al siguiente acumulador superior

Descripción

La instrucción "Desplazar contenido al siguiente acumulador superior" desplaza el contenido del acumulador 1 al acumulador 2.

El contenido del acumulador 1 no se ve influido por la instrucción y permanece invariable una vez ejecutada la misma. El contenido del acumulador 2 se pierde.

La tabla siguiente muestra el contenido de los acumuladores 1 y 2 antes y después de ejecutar la instrucción:

Estado	Acumulador	
	1	2
Antes de la ejecución	Valor A	Valor B
Después de la ejecución	Valor A	Valor A

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Desplazar contenido al siguiente acumulador superior" se utiliza la siguiente sintaxis:

```
PUSH
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
PUSH	// Desplazar el contenido de los acumuladores 1 a 2 al siguiente acumulador superior en cada caso.
L "Tag_Value_2"	// Cargar el valor del operando en el acumulador 1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

POP: Desplazar contenido al siguiente acumulador inferior

Descripción

La instrucción "Desplazar contenido al siguiente acumulador inferior" desplaza el contenido del acumulador 2 al acumulador 1.

El contenido del acumulador 2 no se ve influido por la instrucción y permanece invariable una vez ejecutada la misma. El contenido del acumulador 1 se pierde.

La tabla siguiente muestra el contenido de los acumuladores 1 y 2 antes y después de ejecutar la instrucción:

Estado	Acumulador	
	1	2
Antes de la ejecución	Valor A	Valor B
Después de la ejecución	Valor B	Valor B

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Desplazar contenido al siguiente acumulador inferior" se utiliza la siguiente sintaxis:

```
POP
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`T "Tag_Value_1"`	`// Transferir el contenido del acumulador 1 al operando "Tag_Value_1".`
`POP`	`// Desplazar el contenido del acumulador 2 al acumulador 1 inferior.`
`T "Tag_Value_2"`	`// Transferir el contenido del acumulador 1 al operando "Tag_Value_2".`

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Registro de direcciones

+AR1: Sumar acumulador 1 a AR1

Descripción

La instrucción "Sumar acumulador 1 a AR1" suma un valor al contenido del registro de direcciones 1. El tipo de puntero que está en el registro de direcciones 1 y el área de operandos se conservan.

Para indicar el valor que debe sumarse existen las posibilidades siguientes:

- Indicación mediante una constante: la instrucción suma el valor de la constante al registro de direcciones 1. El valor de la constante debe equivaler al formato de un puntero interno del área (POINTER).
- Indicación mediante el valor de la palabra derecha del acumulador 1: la instrucción interpreta el valor de la palabra derecha del acumulador 1 como un entero de 16 bits y lo amplía a 24 bits con el signo correcto. Seguidamente, la instrucción suma el valor del acumulador 1 al registro de direcciones 1. Se admiten valores de -32 768 a +32 767.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Si el valor de la suma se indica mediante una constante, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR1":

```
+AR1 <constante>
```

Si el valor de la suma se indica mediante la palabra derecha del acumulador 1, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR1":

```
+AR1
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar acumulador 1 a AR1":

Parámetro	Tipo de datos	Descripción
<Constante>	POINTER	Valor que se suma al registro de direcciones 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
+AR1 P#10.0	// Sumar el puntero al registro de direcciones 1.
L "Tag_Value"	// Cargar el valor del operando "Tag_Value" en la palabra derecha del acumulador 1.
+AR1	// Sumar el valor del acumulador 1 al registro de direcciones 1.
TAR1 %MD24	// Transferir el contenido del registro de direcciones 1 a la palabra doble MD24.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

+AR2: Sumar acumulador 1 a AR2

Descripción

La instrucción "Sumar acumulador 1 a AR2" suma un valor al contenido del registro de direcciones 2. El tipo de puntero que está en el registro de direcciones 2 y el área de operandos se conservan.

Para indicar el valor que debe sumarse existen las posibilidades siguientes:

- Indicación mediante una constante: la instrucción suma el valor de la constante al registro de direcciones 2. El valor de la constante debe equivaler al formato de un puntero interno del área (POINTER).
- Indicación mediante el valor de la palabra derecha del acumulador 1: la instrucción interpreta el valor de la palabra derecha del acumulador 1 como un entero de 16 bits y lo amplía a 24 bits con el signo correcto. Seguidamente, la instrucción suma el valor del acumulador 1 al registro de direcciones 2. Se admiten valores de -32 768 a +32 767.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Nota

El registro de direcciones AR2 se utiiza para procesar multiinstancias. Si se programa la instrucción "Sumar acumulador 1 a AR2", primero debe "salvarse" el contenido del registro de direcciones 2 y volver a cargarlo después.

Sintaxis

Si el valor de la suma se indica mediante una constante, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR2":

```
+AR2 <constante>
```

Si el valor de la suma se indica mediante la palabra derecha del acumulador 1, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR2":

```
+AR2
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar acumulador 1 a AR2":

Parámetro	Tipo de datos	Descripción
<Constante>	POINTER	Valor que se suma al registro de direcciones 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
+AR2 P#10.0	// Sumar el puntero al registro de direcciones 2.
L "Tag_Value"	// Cargar el valor del operando "Tag_Value" en la palabra derecha del acumulador 1.
+AR2	// Sumar el valor del acumulador 1 al registro de direcciones 2.
TAR2 %MD24	// Transferir el contenido del registro de direcciones 1 a la palabra doble MD24.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

Instrucciones nulas

BLD: Refresco de imagen (instrucción nula)

Descripción

La instrucción "Refresco de imagen (instrucción nula)" no ejecuta ninguna función y no influye en los bits de estado. La instrucción sirve para reconocer secuencias de códigos en una transferencia de parámetros o en los segmentos KOP/FUP. Se genera automáticamente

cuando se visualiza un programa KOP o FUP en AWL. El valor de parámetro es el número de identificación de la instrucción y es generado por la programadora.

Sintaxis

Para la instrucción "Refresco de imagen (instrucción nula)" se utiliza la siguiente sintaxis:

```
BLD <número de identificación>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Refresco de imagen (instrucción nula)":

Parámetro	Tipo de datos	Descripción
<número de identificación>	WORD	Número de identificación de la instrucción

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

NOP 0: Instrucción nula

Descripción

La instrucción "Instrucción nula" con un parámetro 0 no ejecuta ninguna función y no influye en los bits de estado. El código de la instrucción contiene un patrón de bits con 16 ceros. La instrucción solo es importante para la programadora cuando se visualiza un programa.

Sintaxis

Para la instrucción "Instrucción nula" con un parámetro 0 se utiliza la siguiente sintaxis:

```
NOP 0
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
JC NEXT	// Si la condición se cumple (RLO = "1"), saltar a la etiqueta "NEXT".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
TAK	// Intercambiar contenido de los acumuladores 1 y 2.
NEXT: NOP 0	// Etiqueta "NEXT"

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

NOP 1: Instrucción nula

Descripción

La instrucción "Instrucción nula" con un parámetro 1 no ejecuta ninguna función y no influye en los bits de estado. El código de la instrucción contiene un patrón de bits con 16 unos. La instrucción solo es importante para la programadora cuando se visualiza un programa.

Sintaxis

Para la instrucción "Instrucción nula" con un parámetro 1 se utiliza la siguiente sintaxis:

```
NOP 1
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Consulta y activación de los bits de estado en AWL (Página 2560)

SCL

Operaciones lógicas con bits

R_TRIG: Activar variable con flanco de señal ascendente

Descripción

La instrucción "Activar variable con flanco de señal ascendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de "0" a "1". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Activar variable con flanco de señal ascendente" se utiliza la siguiente sintaxis:

```
SCL
<DB_instancia>(CLK := <Operando>,
             Q => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal ascendente":

Parámetro	Declaración	Tipo de datos	Descripción
CLK	Input	BOOL	Señal entrante cuyo flanco se consulta
Q	Output	BOOL	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"R_TRIG_DB"(CLK := "TagIn",
            Q => "TagOut");
```

En la variable del DB de instancia "R_TRIG_DB" se almacena el RLO de la consulta anterior. Si en el operando "TagIn" se detecta un cambio del estado lógico del RLO de "0" a "1", la salida "TagOut" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

F_TRIG: Activar variable con flanco de señal descendente

Descripción

La instrucción "Activar variable con flanco de señal descendente" permite activar en el DB de instancia una variable indicada cuando se produzca un cambio del resultado lógico (RLO) de

"1" a "0". La instrucción compara el RLO actual de la entrada CLK con el RLO de la consulta anterior, que está almacenado en el DB de instancia indicado. Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la variable del DB de instancia adopta el estado lógico "1" y la salida Q devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si la marca de flancos se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Activar variable con flanco de señal descendente" se utiliza la siguiente sintaxis:

SCL

```
<DB_instancia>(CLK := <Operando>,
               Q => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar variable con flanco de señal descendente":

Parámetro	Declaración	Tipo de datos	Descripción
CLK	Input	BOOL	Señal entrante cuyo flanco se consulta
Q	Output	BOOL	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"F_TRIG_DB"(CLK := "TagIn",
            Q => "TagOut");
```

En la variable del DB de instancia "F_TRIG_DB" se almacena el RLO de la consulta anterior. Si en el operando "TagIn" se detecta un cambio del estado lógico del RLO de "1" a "0", la salida "TagOut" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Temporizadores

Temporizadores CEI

TP: Impulso

Descripción

La instrucción "Impulso" permite activar el parámetro Q durante el tiempo PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. El parámetro Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico del parámetro Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Si cuando se alcanza el tiempo PT el estado lógico del parámetro IN es "0", se resetea el parámetro ET.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TP_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME o TP_LTIME en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Sintaxis

Para la instrucción "Impulso" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TP(IN  := <Operando>,
                 PT := <Operando>,
                 Q => <Operando>,
                 ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

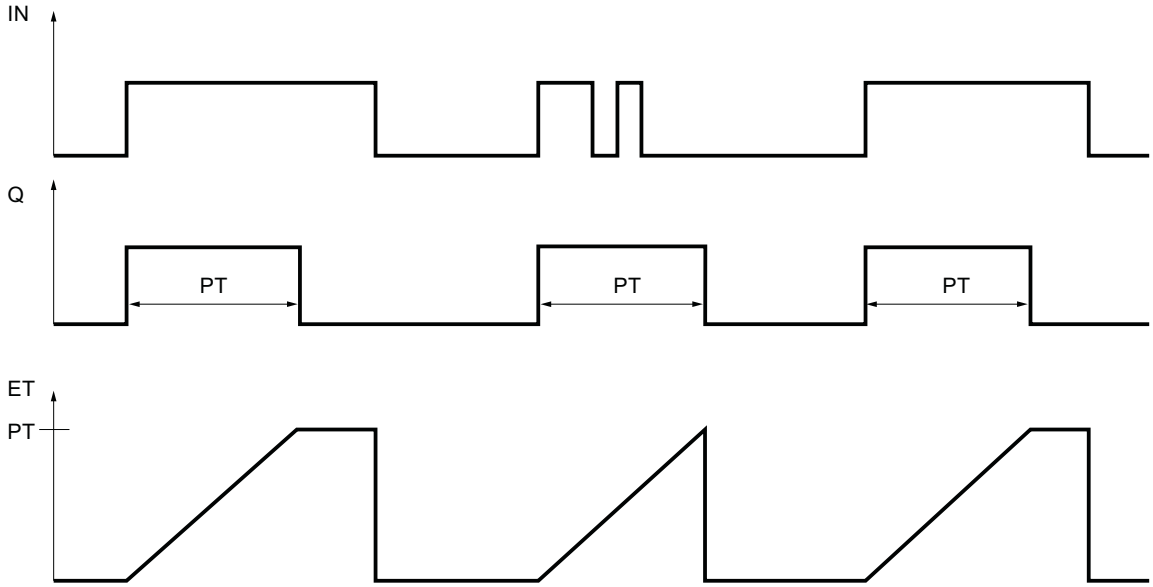
Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	BOOL	BOOL	Entrada de arranque
PT	Input	TIME	TIME, LTIME	Duración del impulso. El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
Q	Output	BOOL	BOOL	Operando que se activa por el tiempo PT.
ET	Output	TIME	TIME, LTIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"TP_DB".TP(IN := "Tag_Start",
           PT := "Tag_PresetTime",
           Q => "Tag_Status",
           ET => "Tag_ElapsedTime");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar y el operando "Tag_Status" se pone a "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TON: Retardo al conectar

Descripción

La instrucción "Retardo al conectar" permite retardar la activación del parámetro Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. Una vez transcurrido el tiempo PT, el parámetro Q devuelve el estado lógico "1". El parámetro Q permanecerá activado mientras la entrada de arranque esté puesta a "1". Cuando el estado lógico del parámetro IN cambia de "1" a "0", se resetea el parámetro Q. La función de temporización se reinicia al detectarse un nuevo flanco de señal ascendente en el parámetro IN.

El valor de tiempo actual se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. El parámetro ET se resetea en cuanto el estado lógico del parámetro IN cambia a "0".

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TON_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME o TON_LTIME en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Sintaxis

Para la instrucción "Retardo al conectar" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TON(IN  := <Operando>,
                  PT := <Operando>,
                  Q => <Operando>,
                  ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

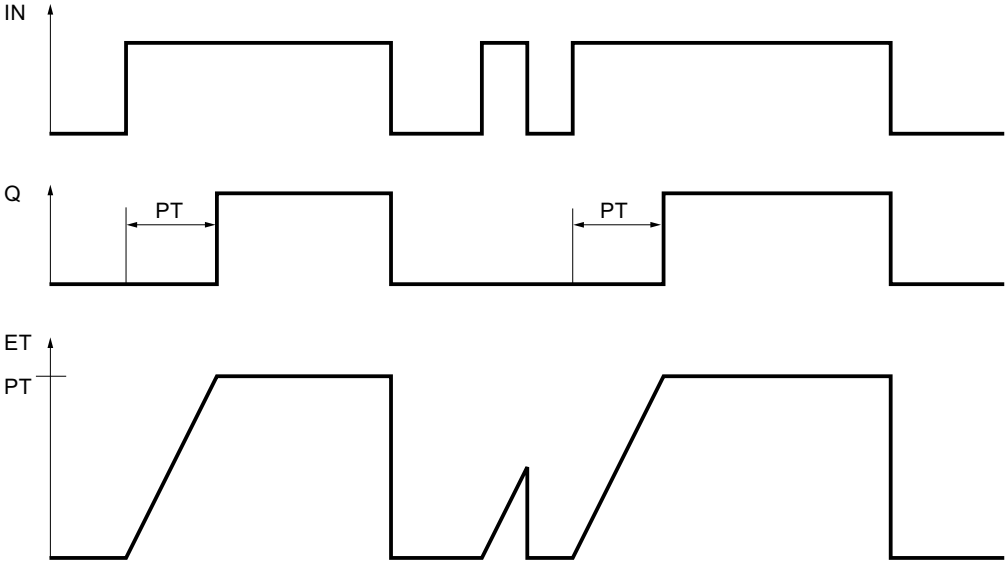
Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	BOOL	BOOL	Entrada de arranque
PT	Input	TIME	TIME, LTIME	Tiempo de retardo al conectar. El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
Q	Output	BOOL	BOOL	Operando que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar. Una vez transcurrido el tiempo, el operando "Tag_Status" se pone al estado lógico "1". El operando "Tag_Status" permanece a "1" mientras el operando "Tag_Start" tenga el estado lógico "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", se resetea el operando "Tag_Status".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TOF: Retardo al desconectar

Descripción

La instrucción "Retardo al desconectar" permite retardar el reset del parámetro Q por el tiempo programado PT. El parámetro Q se activa cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico del parámetro IN cambia nuevamente a "0", el tiempo programado PT deja de contar. El parámetro Q permanecerá activado mientras transcurra el tiempo PT. Una vez transcurrido el tiempo PT se resetea el parámetro Q. Si el estado lógico del parámetro IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico del parámetro Q permanece puesto a "1".

El valor de tiempo actual se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el valor actual del parámetro ET se conservará hasta que el parámetro INcambie nuevamente a "1". Si el parámetro IN cambia a "1" antes de transcurrir el tiempo PT, el parámetro ET adopta el valor T#0s.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TOF_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME o TOF_LTIME en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Sintaxis

Para la instrucción "Retardo al desconectar" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TOF(IN  := <Operando>,
                  PT := <Operand>,
                  Q => <Operando>,
                  ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

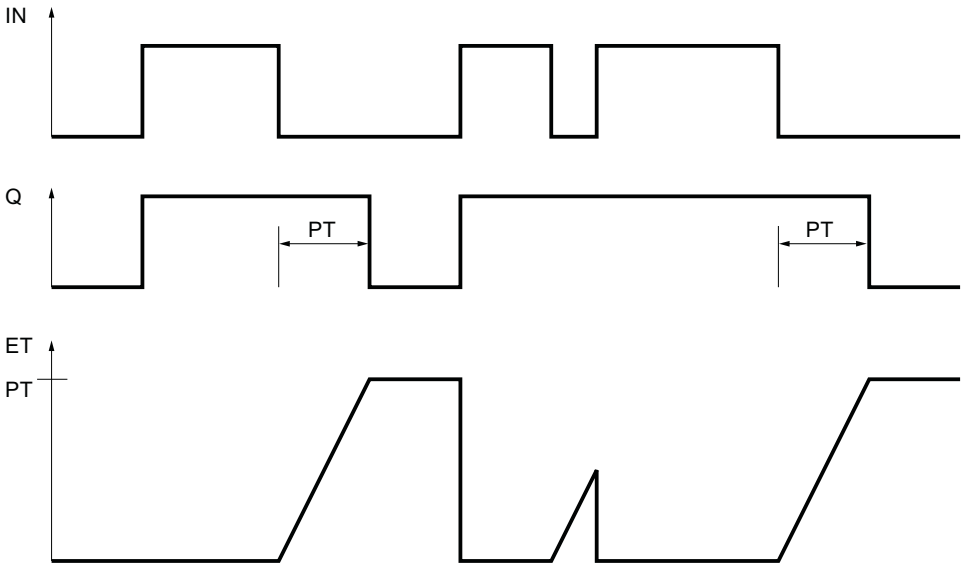
Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	BOOL	BOOL	Entrada de arranque
PT	Input	TIME	TIME, LTIME	Tiempo de retardo al desconectar. El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
Q	Output	BOOL	BOOL	Operando que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TOF_DB".TOF(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Si se produce un cambio del estado lógico del operando "Tag_Start" de "0" a "1", el operando se pone a "Tag_Status". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", el tiempo programado en el parámetro PT empieza a contar. Mientras transcurra el tiempo, permanecerá activado el operando "Tag_Status". Una vez transcurrido el tiempo, el operando "Tag_Status" se desactiva. El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TONR: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PT. Cuando el estado lógico del parámetro IN cambia a "1", se ejecuta la instrucción y se empieza a contar el tiempo PT. Mientras transcurre el tiempo PTse van acumulando los valores de tiempo que se leen cuando el estado lógico del parámetroIN es "1". El tiempo acumulado se devuelve en el parámetro ET y se puede consultar allí. Una vez se alcanza el tiempo PT, el parámetro Q devolverá el estado lógico "1". El parámetro Q permanece a "1" aunque el estado lógico del parámetro IN cambie a "0".

El parámetro R desactiva los parámetros ET y Q independientemente del estado lógico del parámetro IN.

A cada llamada de la instrucción "Acumulador de tiempo" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción.

Para CPU S7-1200

El temporizador CEI es una estructura del tipo de datos IEC_TIMER o TONR_TIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME en la sección "Static" de un bloque (p. ej., #MyTONR_TIMER)

Para CPU S7-1500

El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME o TONR_LTIME en la sección "Static" de un bloque (p. ej., #MyTONR_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la

carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Sintaxis

Para la instrucción "Acumulador de tiempo" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TONR(IN  := <Operando>,
                   R := <Operando>,
                   PT := <Operando>,
                   Q => <Operando>,
                   ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               R := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

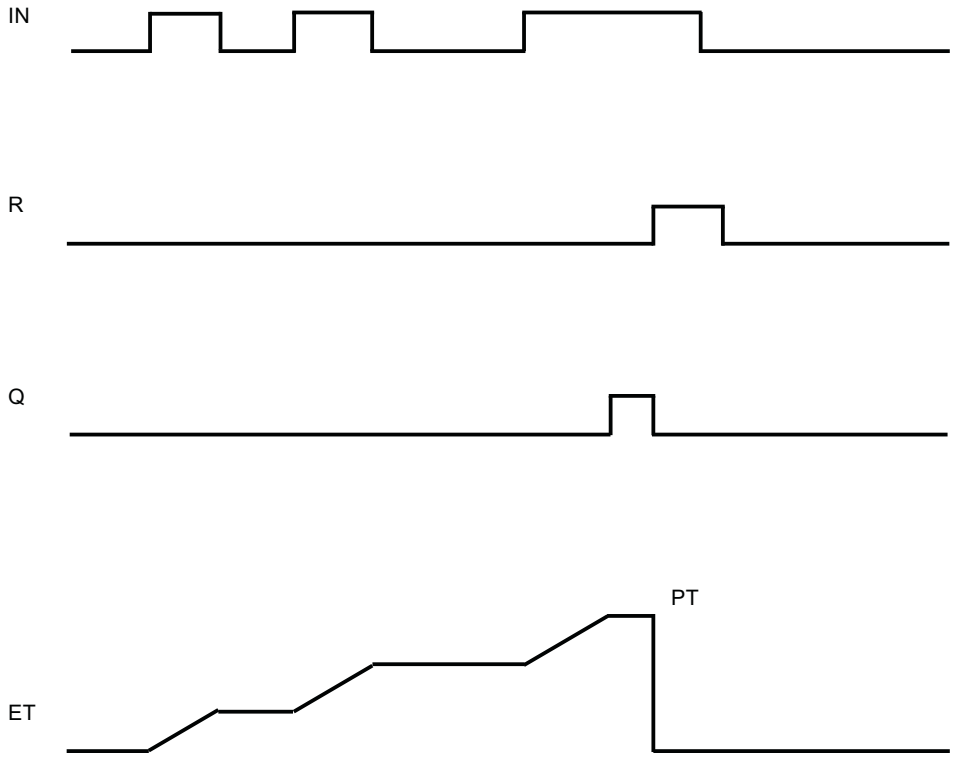
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	BOOL	BOOL	Entrada de arranque
R	Input	BOOL	BOOL	Inicializar los parámetros ET y Q
PT	Input	TIME	TIME, LTIME	Tiempo máximo de lectura del tiempo. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	BOOL	Operando que permanece activado una vez transcurrido el tiempo PT.
ET	Output	TIME	TIME, LTIME	Tiempo acumulado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Acumulador de tiempo":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TONR_DB".TONR(IN := "Tag_Start",
               R := "Tag_Reset",
               PT := "Tag_PresetTime",
               Q => "Tag_Status",
               ET => "Tag_Time");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar. Mientras transcurre el tiempo, se van acumulando los valores de tiempo que se leen cuando el estado lógico del operando "Tag_Start" es "1". El tiempo acumulado se almacena en el operando "Tag_Time". Una vez alcanzado el valor de tiempo indicado en el parámetro PT, el operando "Tag_Status" adopta el estado lógico "1". El valor actual de tiempo se guarda en el operando "Tag_Time".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

RESET_TIMER: Inicializar temporizador

Descripción

La instrucción "Inicializar temporizador" permite poner a "0" un temporizador CEI. Los componentes de estructura del temporizador en el bloque de datos indicado se ponen a "0".

La instrucción no afecta al RLO. En el parámetro TIMER se asigna a la instrucción "Inicializar temporizador" un temporizador CEI declarado en el programa.

Los datos de la instrucción se actualizan únicamente al efectuar una llamada de la instrucción, y no cada vez que se accede al temporizador CEI asignado. La consulta de datos es igual a la llamada de instrucción únicamente hasta la siguiente llamada de instrucción.

Sintaxis

Para la instrucción "Inicializar temporizador" se utiliza la siguiente sintaxis:

SCL

```
RESET_TIMER (TIMER := <Temporizador CEI>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar temporizador":

Parámetro	Declaración	Tipo de datos	Descripción
<Temporizador CEI>	Output	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	Temporizador CEI que se inicializa

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF #started = false THEN
"TON_DB".TON(IN := "Tag_Start",
```

SCL

```
            PT := "Tag_PresetTime",
            Q => "Tag_Status",
            ET => "Tag_ElapsedTime");
#started := true;
END_IF;

IF "TON_DB".ET < T#25s THEN
RESET_TIMER(TIMER := "TON_DB");
#started := false;
END_IF;
```

Si la variable #started devuelve el estado lógico "0", se ejecuta la instrucción "Retardo al conectar" cuando se produce un flanco de señal ascendente en el operando "Tag_Start". El temporizador CEI depositado en el bloque de datos instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PresetTime". El operando "Tag_Status" se activa una vez transcurrido el tiempo predeterminado por el operando "Tag_PresetTime". El parámetro Q permanecerá activado mientras el operando "Tag_Start" tenga el estado lógico "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva el operando en el parámetro Q.

Si el tiempo transcurrido del temporizador CEI "TON_DB" es inferior a 25 s, la instrucción "Inicializar temporizador" se ejecuta y el temporizador depositado en el bloque de datos instancia "TON_DB" se inicializa.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

PRESET_TIMER: Cargar tiempo

Descripción

La instrucción "Cargar tiempo" permite ajustar el tiempo de un temporizador CEI. La instrucción se ejecuta en cada ciclo si el resultado lógico (RLO) de la entrada de la instrucción tiene el estado lógico "1". La instrucción escribe el tiempo indicado en la estructura del temporizador CEI indicado.

Nota

Si el temporizador CEI indicado se ejecuta mientras se ejecuta la instrucción, esta sobrescribe el tiempo actual del temporizador CEI indicado. Esto puede modificar el estado del temporizador CEI.

A la instrucción "Cargar tiempo" debe asignársele un temporizador CEI declarado en el programa.

La actualización de los datos de la instrucción se lleva a cabo cuando se llama la instrucción y cada vez que se accede al temporizador CEI asignado. La consulta de Q o ET (p. ej. "MyTimer".Q y "MyTimer".ET) actualiza la estructura de IEC_TIMER.

Sintaxis

Para la instrucción "Cargar tiempo" se utiliza la siguiente sintaxis:

SCL

```
PRESET_TIMER (PT := <Operando>,
              TIMER := <Temporizador CEI>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar tiempo":

Parámetro	Declaración	Tipo de datos	Descripción
<Tiempo>	Input	TIME, LTIME	Tiempo que cuenta el temporizador CEI.
<Temporizador CEI>	Output	IEC_TIMER, IEC_LTIMER, TP_TIME, TP_LTIME, TON_TIME, TON_LTIME, TOF_TIME, TOF_LTIME, TONR_TIME, TONR_LTIME	Temporizador CEI cuyo tiempo se ajusta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF #started = false THEN
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
#started := true;
#preset = true
END_IF;
```

SCL

```
IF "TON_DB".ET < T#10s AND #preset = true THEN
PRESET_TIMER(PT := T#25s,
             TIMER := "TON_DB");
#preset := false;
END_IF;
```

Si la variable #started tiene el estado lógico "0" y el operando "Tag_Start" tiene un flanco de señal ascendente, se ejecuta la instrucción "Retardo al conectar". El temporizador CEI depositado en el bloque de datos instancia "TON_DB" arranca con el tiempo predeterminado por el operando "Tag_PresetTime". El operando "Tag_Status" se activa una vez transcurrido el tiempo PT predeterminado por el operando "Tag_PresetTime". El parámetro Q permanecerá activado mientras el operando "Tag_Start" tenga el estado lógico "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva el operando en el parámetro Q.

Si el tiempo transcurrido del temporizador CEI "TON_DB" es inferior a 10 s y la variable #preset tiene el estado lógico "1", se ejecuta la instrucción "Cargar tiempo". La instrucción escribe el tiempo indicado en el parámetro PT en el bloque de datos instancia "TON_DB" y sobrescribe a la vez el valor de tiempo del operando "Tag_PresetTime" en el bloque de datos instancia. Esto puede modificar el estado lógico del estado del temporizador en la siguiente consulta o en el siguiente acceso a "TON_DB".Q o "TON_DB".ET.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Temporizadores SIMATIC

S_PULSE: Parametrizar y arrancar temporizador como impulso

Descripción

La instrucción "Parametrizar y arrancar temporizador como impulso" arranca el temporizador programado en el parámetro T_NO cuando se detecta un cambio del resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en el parámetro S. El temporizador cuenta el tiempo programado TV mientras el estado lógico del parámetro S sea "1".

Si el estado lógico del parámetro S cambia a "0" antes de que transcurra el tiempo programado, el temporizador se detiene y el parámetro "Q" adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

El parámetro Q devuelve el estado lógico "1" mientras el temporizador está contando y el estado lógico del parámetro S es "1". Si el estado lógico del parámetro S cambia a "0" antes de que transcurra el tiempo programado, el parámetro Q devuelve el estado lógico "0". Si el parámetro R resetea el temporizador o si el tiempo ha transcurrido, el parámetro Q también devuelve el estado lógico "0".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como impulso" se utiliza la sintaxis siguiente:

SCL

```
S_PULSE(T_NO := <Operando>,
        S := <Operando>,
        TV := <Operando>,
        R := <Operando>,
        Q => <Operando>,
        BI => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

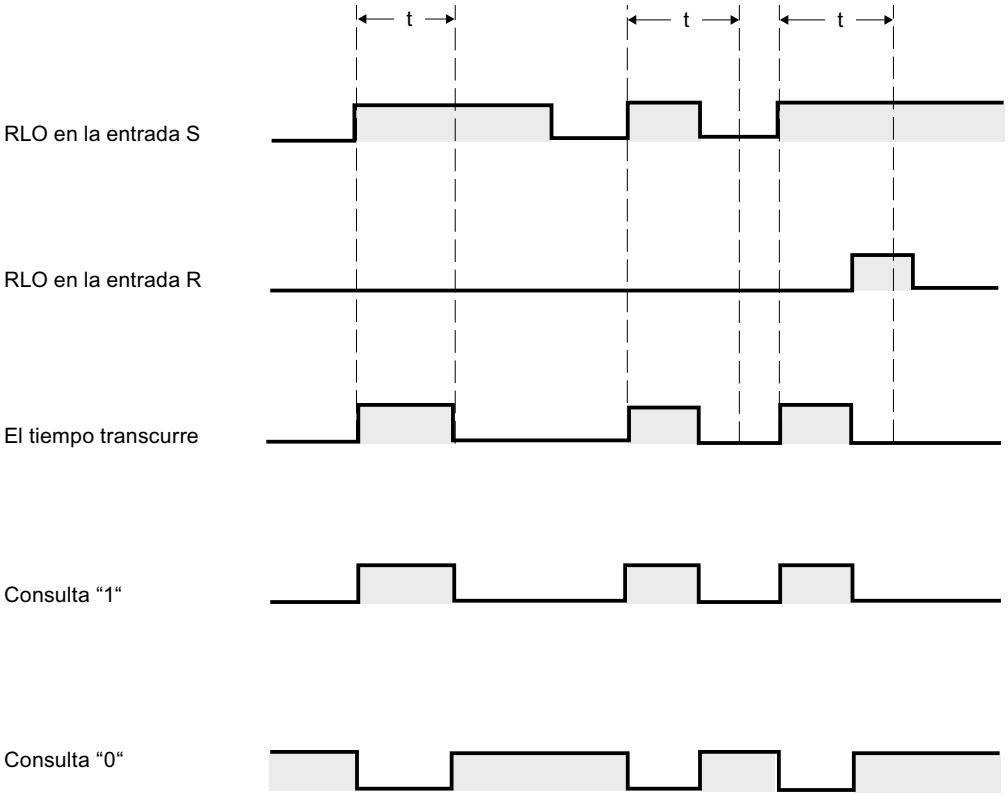
Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador

Parámetro	Declaración	Tipo de datos	Descripción
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_PULSE(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q := "Tag_Status",
                        BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "Tag_Number" y sigue contando mientras el operando "Tag_1" devuelva el estado lógico "1".

Si el estado lógico del parámetro S cambia a "0" antes de que transcurra el tiempo programado, el operando "Tag_Status" adopta el estado lógico "0". Si el parámetro R resetea el temporizador o si el tiempo ha transcurrido, el operando "Tag_Status" también devuelve el estado lógico "0".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 4093)

S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado

Descripción

La instrucción "Parametrizar y arrancar temporizador como impulso prolongado" arranca el temporizador programado cuando se detecta un flanco de señal ascendente en el parámetroS. El temporizador deja de contar el tiempo programado TV aunque el estado lógico del parámetro S cambie a "0". Mientras el temporizador está contando, el parámetro Q devuelve el estado lógico "1".

Una vez transcurrido el tiempo, el parámetro Q adopta el valor lógico "0". Si el estado lógico del parámetro S cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo programado en el parámetro TV.

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

Si el temporizador está contando y el estado lógico del parámetro R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está contando, el estado lógico "1" en el parámetro R no tiene ningún efecto.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como impulso prolongado" se utiliza la sintaxis siguiente:

SCL

```
S_PEXT(T_NO := <Operando>,
       S := <Operando>,
       TV := <Operando>,
       R := <Operando>,
       Q => <Operando>,
       BI =><Operando>)
```

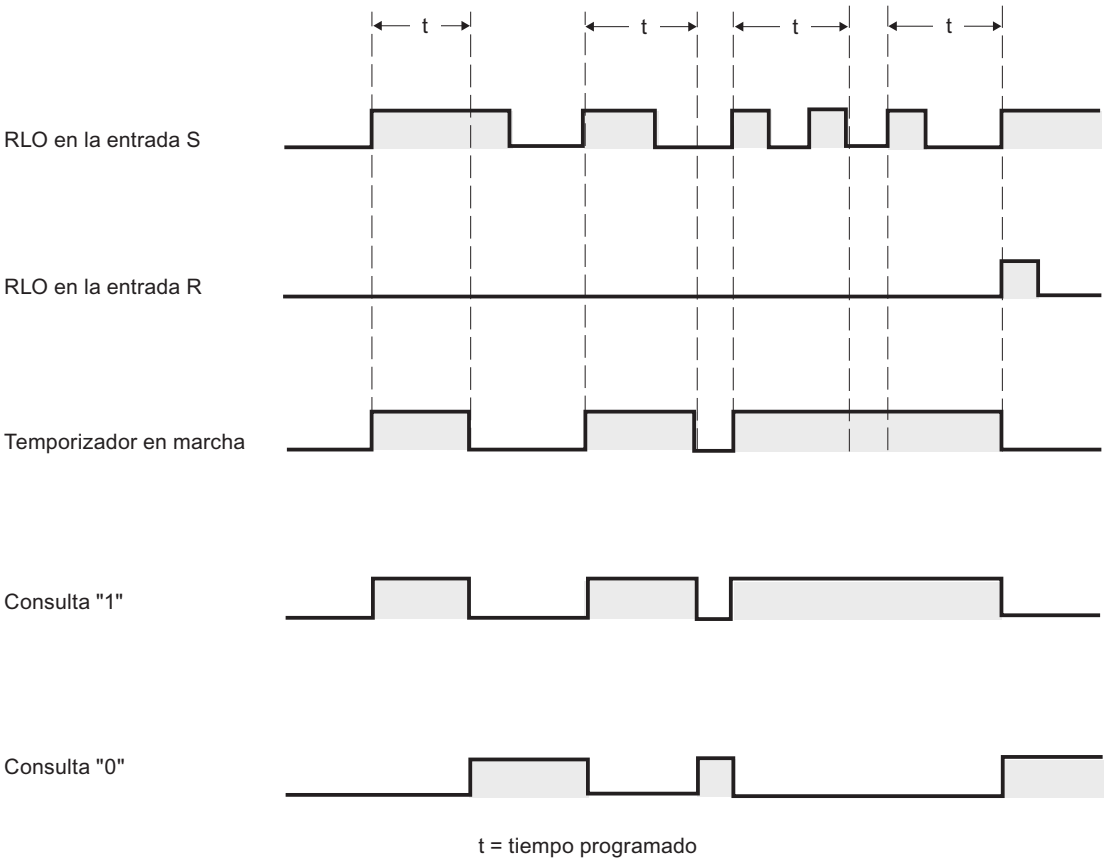
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_PEXT(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q := "Tag_Status",
                       BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. Mientras el temporizador está en marcha, el operando "Tag_Status" devuelve el estado lógico "1". Una vez transcurrido el tiempo, el operando "Tag_Status" adopta el valor lógico "0". Si el estado lógico de la entrada S cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo "Tag_Number".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 4093)

S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" arranca el temporizador programado como retardo a la conexión cuando se detecta un flanco de señal ascendente en el parámetro S. El temporizador cuenta el tiempo programado TV mientras el estado lógico del parámetro S sea "1".

Si el tiempo ha transcurrido correctamente y el parámetro S sigue presentando el estado lógico "1", el parámetro Q devuelve el estado lógico "1". Si el estado lógico del parámetro S cambia de "1" a "0" mientras el temporizador está contando, este se detiene. En este caso, la salida Q adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

Si el temporizador está contando y el estado lógico de la entrada R cambia de "0" a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. En este caso, el estado lógico del parámetro Q es "0". El temporizador se resetea si el parámetro R presenta el estado lógico "1", aunque el temporizador no esté en marcha y el resultado lógico (RLO) del parámetro S sea "1".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" se utiliza la sintaxis siguiente:

SCL

```
S_ODT(T_NO := <Operando>,
      S := <Operando>,
      TV := <Operando>,
      R := <Operando>,
      Q => <Operando>,
      BI =><Operando>)
```

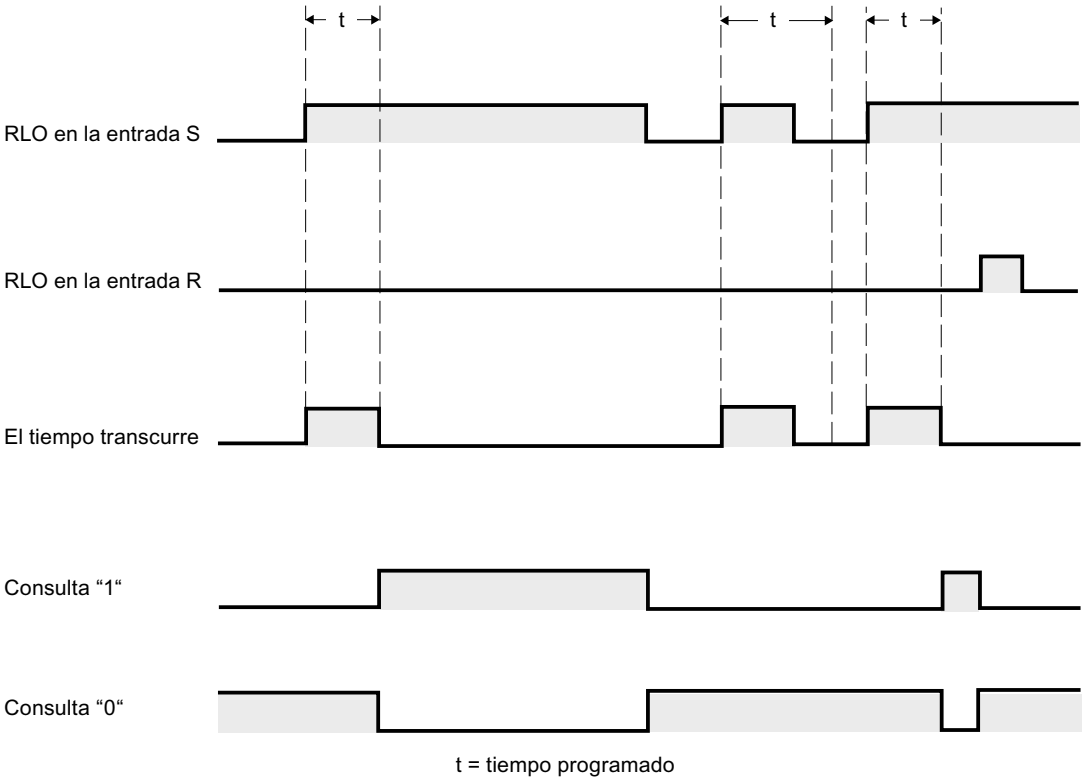
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_ODT(T_NO := "Timer_1",
                      S := "Tag_1",
                      TV := "Tag_Number",
                      R := "Tag_Reset",
                      Q := "Tag_Status",
                      BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el tiempo "Tag_Number" mientras el estado lógico del operando "Tag_1" sea "1".

Si el tiempo ha transcurrido correctamente y el operando "Tag_Status" presenta el estado lógico "1", el operando "Tag_Status" se pone a "1". Si el estado lógico del operando "Tag_1" cambia de "1" a "0" mientras el temporizador está contando, este se detiene. El operando "Tag_Status" devuelve en ese caso el estado lógico "0".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 4093)

S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria

Descripción

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" arranca el temporizador programado cuando se detecta un flanco de señal ascendente en el parámetro S. El temporizador deja de contar el tiempo programado TV aunque el estado lógico del parámetro S cambie a "0".

Una vez transcurrido el tiempo, el parámetro Q devuelve el estado lógico "1" independientemente del estado lógico del parámetro S. Si el estado lógico del parámetro S cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo programado TV.

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

El estado lógico "1" del parámetro R pone a "0" el valor actual de tiempo y la base de tiempo, independientemente del estado lógico del parámetro S. En este caso, el estado lógico del parámetro Q es "0".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" se utiliza la sintaxis siguiente:

SCL

```
S_ODTS(T_NO := <Operando>,
       S := <Operando>,
       TV := <Operando>,
       R := <Operando>,
       Q => <Operando>,
       BI =><Operando>)
```

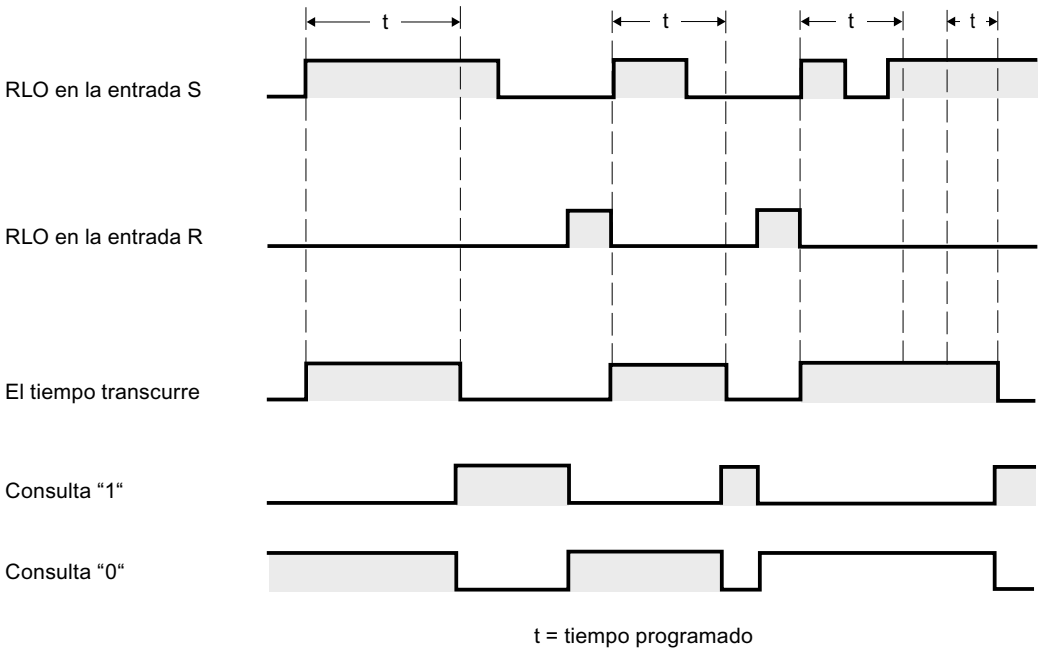
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_ODTS(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q := "Tag_Status",
                       BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el tiempo "Tag_Number".

Una vez transcurrido el tiempo, el operando "Tag_Status" devuelve el estado lógico "1" independientemente del estado lógico del operando "Tag_1". Si el estado lógico del operando "Tag_1" cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo "Tag_Number".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 4093)

S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" arranca el temporizador programado cuando se detecta un flanco de señal descendente en el parámetro S. El temporizador cuenta el tiempo programado TV. Mientras el temporizador está contando o el parámetro S devuelve el estado lógico "1", el parámetro Q presenta el estado lógico "1".

Si el tiempo ha transcurrido y el estado lógico es "0", el parámetro Q adopta el valor lógico "0". Si el estado lógico del parámetro S cambia de "0" a "1" mientras el temporizador está en marcha, este se detiene. El temporizador solo arranca de nuevo cuando se detecta un flanco de señal descendente en el parámetro S.

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

El estado lógico "1" del parámetro R pone a "0" el valor actual de tiempo y la base de tiempo. En este caso, el estado lógico del parámetro Q es "0".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" se utiliza la sintaxis siguiente:

SCL

```
S_OFFDT(T_NO := <Operando>,
        S := <Operando>,
        TV := <Operando>,
        R := <Operando>,
        Q => <Operando>,
        BI =><Operando>)
```

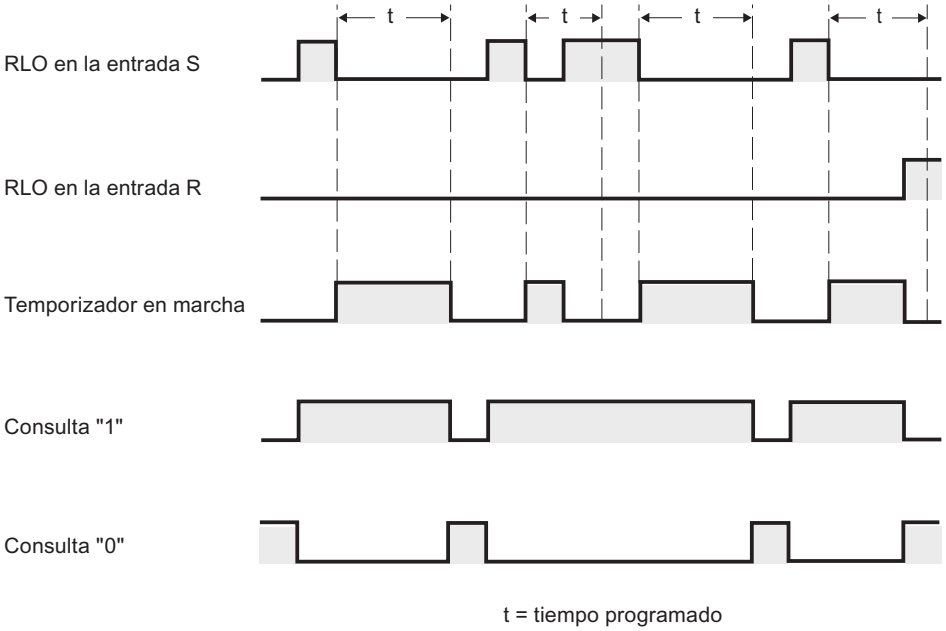
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_OFFDT(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q := "Tag_Status",
                        BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el tiempo "Tag_Number". Mientras el temporizador está contando o el operando "Tag_1" devuelve el estado lógico "1", el operando "Tag_Status" presenta el estado lógico "1".

Si el tiempo ha transcurrido y el estado lógico del operando "Tag_1" es "0", el operando "Tag_Status" adopta el valor lógico "0". Si el estado lógico del operando "Tag_1" cambia de "0" a "1" mientras el temporizador está en marcha, el temporizador se reinicia. El temporizador solo arranca de nuevo cuando se detecta un flanco descendente en el parámetro S.

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 4093)

Contadores

Contadores CEI

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor de contaje actual del parámetro CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos indicado en el parámetro CV. Cuando se alcanza el valor límite superior, el estado lógico del parámetro CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. El estado lógico del parámetro Q es determinado por el parámetro PV. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0". En el parámetro PV también se puede indicar una constante.

El valor del parámetro CV se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico del parámetro CU no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para una CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT

Para una CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTU_SINT / CTU_USINT • CTU_INT / CTU_UINT • CTU_DINT / CTU_UDINT • CTU_LINT / CTU_ULINT

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTU_<tipo de datos> en la sección "Static" de un bloque (p. ej. #MyCTU_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se declaran remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se declara remanente en la interfaz del bloque.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

Tabla 9-28 Bloque de datos del tipo de datos de sistema IEC_COUNTER (Shared DB)

SCL

```
<DB_Contador_CEI>.CTU(CU  := <Operando>,
                      R  := <Operando>,
                      PV  := <Operando>,
                      Q => <Operando>,
```

SCL

```
                    CV => <Operando>)
```

Tabla 9-29 Variable local

SCL

```
#myLocal_counter(CU  := <Operando>,
                 R  := <Operando>,
                 PV  := <Operando>,
                 Q => <Operando>,
                 CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
CU	Input	BOOL	Entrada de contaje
R	Input	BOOL	Entrada de reset
PV	Input	Enteros	Valor con el que se activa la salida Q
Q	Output	BOOL	Estado del contador
CV	Output	Enteros, CHAR, DATE	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"IEC_COUNTER_DB".CTU(CU := "Tag_Start",
                     R := "Tag_Reset",
                     PV := "Tag_PresetValue",
                     Q => "Tag_Status",
                     CV=> "Tag_CounterValue");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor de contaje actual del operando "Tag_CounterValue" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos indicado (INT = 32767).

La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea mayor o igual al valor del operando "Tag_PresetValue". En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0". El valor de contaje actual se guarda en el operando "Tag_CounterValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos indicado. Cuando se alcanza el valor límite inferior, el estado lógico del parámetro CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. Si el valor de contaje actual es menor o igual a cero, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0". En el parámetro PV también se puede indicar una constante.

El valor del parámetro CV se pone al valor del parámetro PV cuando el estado lógico del parámetro LD cambia a "1". Mientras el parámetro LD tenga el estado lógico "1", el estado lógico del parámetro CD no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para una CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT

Para una CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTD_SINT / CTD_USINT • CTD_INT / CTD_UINT • CTD_DINT / CTD_UDINT • CTD_LINT / CTD_ULINT

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTD_<tipo de datos> en la sección "Static" de un bloque (p. ej. #MyCTD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se declaran remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se declara remanente en la interfaz del bloque.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

Tabla 9-30 Bloque de datos del tipo de datos de sistema IEC_COUNTER (Shared DB)

SCL

```
<DB_Contador_CEI>.CTD(CD  := <Operando>,
                      LD : = <Operando>,
                      PV := <Operando>,
                      Q => <Operando>,
                      CV => <Operando>)
```

Tabla 9-31 Variable local

SCL

```
#myLocal_counter(CD : = <Operando>,
                 LD : = <Operando>,
                 PV  := <Operando>,
                 Q => <Operando>,
                 CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
CD	Input	BOOL	Entrada de contaje
LD	Input	BOOL	Entrada de carga
PV	Input	Enteros	Valor con el que se activa la salida Q
Q	Output	BOOL	Estado del contador
CV	Output	Enteros, CHAR, DATE	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"IEC_SCOUNTER_DB".CTD(CD := "Tag_Start",
                      LD := "Tag_Load",
                      PV := "Tag_PresetValue",
                      Q => "Tag_Status",
                      CV => "Tag_CounterValue");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", la instrucción "Contador descendente" se ejecuta y el valor del operando "Tag_CounterValue" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos indicado (-128).

El operando "Tag_Status" devuelve el estado lógico "1" mientras el valor del contador actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0". El valor de contaje actual se guarda en el operando "Tag_CounterValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1"

(flanco de señal ascendente), el valor de contaje actual del parámetro CV se incrementa en uno. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje del parámetro CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje actual no cambia en el parámetro CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado en el parámetro CV. Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado, ya no se decrementa el valor de contaje.

Cuando el estado lógico del parámetro LD cambia a "1", el valor de contaje del parámetro CV se pone al valor del parámetro PV. Mientras el parámetro LD tenga el estado lógico "1", el estado lógico de los parámetros CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetro R sea "1", un cambio del estado lógico de los parámetros CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en el parámetro QU. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro QU adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QU es "0". En el parámetro PV también se puede indicar una constante.

El estado del contador descendente se puede consultar en el parámetro QD. Si el valor de contaje actual es menor o igual a cero, el parámetro QD adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QD es "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Para una CPU S7-1200

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT

Para una CPU S7-1500

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)	Variable local
• IEC_SCOUNTER / IEC_USCOUNTER • IEC_COUNTER / IEC_UCOUNTER • IEC_DCOUNTER / IEC_UDCOUNTER • IEC_LCOUNTER / IEC_ULCOUNTER	• CTUD_SINT / CTUD_USINT • CTUD_INT / CTUD_UINT • CTUD_DINT / CTUD_UDINT • CTUD_LINT / CTUD_ULINT

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTUD_<tipo de datos> en la sección "Static" de un bloque (p. ej. #MyCTUD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se declaran remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se declara remanente en la interfaz del bloque.

Sintaxis

Para la instrucción "Contador ascendente - descendente" se utiliza la siguiente sintaxis:

Tabla 9-32 Bloque de datos del tipo de datos de sistema IEC_COUNTER (Shared DB)

SCL

```
<DB_Contador_CEI>.CTUD(CU := <Operando>,
                       CD := <Operando>,
                       R := <Operando>,
                       LD: = <Operando>,
                       PV := <Operando>,
                       QU=> <Operando>,
                       QD := <Operando>,
                       CV => <Operando>)
```

Tabla 9-33 Variable local

SCL

```
myLocal_counter(CU := <Operando>,
                CD := <Operando>,
                R := <Operando>,
```

SCL

```
LD: = <Operando>,
PV := <Operando>,
QU=> <Operando>,
QD := <Operando>,
CV=> <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
CU	Input	BOOL	Entrada de contaje ascendente
CD	Input	BOOL	Entrada de contaje descendente
R	Input	BOOL	Entrada de reset
LD	Input	BOOL	Entrada de carga
PV	Input	Enteros	Valor con el que se activa la salida QU.
QU	Output	BOOL	Estado del contador ascendente
QD	Output	BOOL	Estado del contador descendente
CV	Output	Enteros, CHAR, DATE	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"IEC_COUNTER_DB".CTUD(CU := "Tag_Start1",
                      CD := "Tag_Start2",
                      LD := "Tag_Load",
                      R := "Tag_Reset",
                      PV := "Tag_PresetValue",
                      QU => "Tag_CU_Status",
                      QD => "Tag_CD_Status",
                      CV => "Tag_CounterValue");
```

Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_Start1", el valor actual del contador se incrementa en uno y se deposita en el operando "Tag_CounterValue". Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_Start2", el valor del contador se decrementa en uno y se deposita también en el operando "Tag_CounterValue". Cuando se detecta un flanco de señal ascendente en el parámetro CU, el valor del contador se va incrementando hasta alcanzar el valor límite superior del tipo de datos indicado (INT). Cuando se detecta un flanco de señal ascendente en el parámetro CD, el valor del contador se va decrementando hasta alcanzar el valor límite inferior del tipo de datos indicado (INT).

El operando "Tag_CU_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea mayor o igual al valor del operando "Tag_PresetValue". En todos los demás casos, la salida "Tag_CU_Status" devuelve el estado lógico "0".

El operando "Tag_CD_Status" devuelve el estado lógico "1" mientras el valor del contador actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_CD_Status" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Contadores SIMATIC

S_CU: Parametrizar e incrementar contador

Descripción

La instrucción "Parametrizar e incrementar contador" permite incrementar el valor de un contador. Si el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. El valor de contaje actual se deposita en el parámetro CV. El valor de contaje sigue incrementando hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente.

Cuando el estado lógico del parámetro S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el resultado lógico (RLO) de la entrada CU es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetroR sea "1", un cambio en el estado lógico de los parámetros CU y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico del parámetro Q es "1" si el valor de contaje es mayor que cero. Si el valor de contaje es igual a cero, el parámetro Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Sintaxis

Para la instrucción "Parametrizar e incrementar contador" se utiliza la sintaxis siguiente:

SCL

```
S_CU(C_NO := <Operando>,
     CU := <Operando>,
     S := <Operando>,
     PV := <Operando>,
     R := <Operando>,
     Q=> <Operando>,
     CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetros	Declaración	Tipo de datos	Descripción
C_NO	Input	COUNTER, INT	Contadores El número de contadores depende de la CPU.
CU	Input	BOOL	Entrada de contaje ascendente
S	Input	BOOL	Entrada para preajustar el contador
PV	Input	WORD	Valor de contaje predeterminado (C#0 a C#999) en formato BCD
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del contador
CV	Output	WORD, S5TIME, DATE	Valor actual de contaje
Valor de función		WORD, S5TIME, DATE	Valor de contaje actual en formato BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_CU(C_NO := "Counter_1",
                     CU := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Cuando el estado lógico del parámetro "Tag_Start" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Si el estado lógico de la entrada "Tag_1" cambia de "0" a "1", el valor de contaje en

formato BCD se pone al valor del operando "Tag_PresetValue". El valor de contaje se pone a "0" si el operando "Tag_Reset" tiene el estado lógico "1".

El valor de contaje actual se almacena en formato hexadecimal en el operando "Tag_Value".

La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0". El valor actual de contaje se guarda en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

S_CD: Parametrizar y decrementar contador

Descripción

La instrucción "Parametrizar y decrementar contador" permite decrementar el valor de un contador. Si el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se deposita en el parámetro CV. El valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite, el valor de contaje no se decrementa más al detectarse un flanco de señal ascendente.

Cuando el estado lógico del parámetro S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el resultado lógico (RLO) del parámetro CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetroR sea "1", un cambio en el estado lógico de los parámetros CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico del parámetro Q es "1" si el valor de contaje es mayor que cero. Si el valor de contaje es igual a cero, el parámetro Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Sintaxis

Para la instrucción "Parametrizar y decrementar contador" se utiliza la sintaxis siguiente:

SCL

```
S_CD(C_NO := <Operando>,
     CD := <Operando>,
```

SCL

```
      S := <Operando>,
      PV := <Operando>,
      R := <Operando>,
      Q=> <Operando>,
      CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetros	Declaración	Tipo de datos	Descripción
C_NO	Input	COUNTER, INT	Contadores El número de contadores depende de la CPU.
CD	Input	BOOL	Entrada de contaje descendente
S	Input	BOOL	Entrada para preajustar el contador
PV	Input	WORD	Valor de contaje predeterminado (C#0 a C#999) en formato BCD
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del contador
CV	Output	WORD, S5TIME, WORD	Valor actual de contaje
Valor de función		WORD, S5TIME, DATE	Valor de contaje actual en formato BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CD := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es superior a "0", el valor de contaje se decrementa en uno. Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el valor de contaje adopta el valor del operando "Tag_PresetValue" en formato BCD. El valor de contaje se pone a "0" si el operando "Tag_Reset" tiene el estado lógico "1".

El valor de contaje actual se almacena en el operando "Tag_Value".

El operando "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea distinto de "0". El valor actual de contaje se guarda en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

S_CUD: Parametrizar e incrementar/decrementar contador

Descripción

La instrucción "Parametrizar e incrementar/decrementar contador" permite incrementar y decrementar el valor de un contador. Si el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. Si el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se deposita en el parámetro CV. Si en un ciclo del programa se detecta un flanco de señal ascendente en los parámetros CU y CD el valor de contaje permanece invariable.

El valor de contaje sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se alcanza el valor límite inferior "0", no se decrementa más el valor de contaje.

Cuando el estado lógico del parámetro S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el resultado lógico (RLO) de los parámetros CU y CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico de los parámetros CU, CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico del parámetro Q es "1" si el valor de contaje es mayor que cero. Si el valor de contaje es igual a cero, el parámetro Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Sintaxis

Para la instrucción "Parametrizar e incrementar/decrementar contador" se utiliza la sintaxis siguiente:

SCL

```
S_CUD(C_NO := <Operando>,
      CU := <Operando>,
      CD := <Operando>,
      S := <Operando>,
      PV := <Operando>,
      R := <Operando>,
      Q=> <Operando>,
      CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetros	Declaración	Tipo de datos	Descripción
C_NO	Input	COUNTER, INT	Contadores El número de contadores depende de la CPU.
CU	Input	BOOL	Entrada de contaje ascendente
CD	Input	BOOL	Entrada de contaje descendente
S	Input	BOOL	Entrada para preajustar el contador
PV	Input	WORD	Valor de contaje predeterminado (C#0 a C#999) en formato BCD
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del contador
CV	Output	WORD, S5TIME, DATE	Valor de contaje actual (hexadecimal)
Valor de función		WORD, S5TIME, DATE	Valor de contaje actual en formato BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CU := "Tag_CU",
                     CD := "Tag_CD",
```

```
SCL
                    S := "Tag_1",
                    PV := "Tag_PresetValue",
                    R := "Tag_Reset",
                    Q => "Tag_Status",
                    CV => "Tag_Value");
```

Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_CU" y el valor de contaje actual es menor que "999", el valor de contaje se incrementa en uno. Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_CD" y el valor de contaje actual es mayor que "0", el valor de contaje se decrementa en uno.

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el valor de contaje adopta el valor del operando "Tag_PresetValue" en formato BCD. El valor de contaje se pone a "0" si el operando "Tag_Reset" tiene el estado lógico "1".

El valor de contaje actual se almacena en el operando "Tag_Value".

El operando "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea distinto de "0". El valor actual de contaje se guarda en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Comparación

TypeOf: Consultar tipo de datos de una variable VARIANT

Descripción

Con la instrucción "Consultar tipo de datos de una variable VARIANT" se consulta qué tipo de datos tiene una variable a la que apunta un VARIANT. El tipo de datos de la variable se compara con el tipo de datos de una variable que se ha declarado en la interfaz de bloque para ver si es "Igual" o "Diferente".

El operando debe ser del tipo de datos VARIANT. El operando de comparación puede ser un tipo de datos elemental o un tipo de datos PLC.

La instrucción "Consultar tipo de datos de una variable VARIANT" solo se puede utilizar dentro de una instrucción IF.

Sintaxis

Para la instrucción "Consultar tipo de datos de una variable VARIANT" se utiliza la siguiente sintaxis:

SCL

```
TypeOf(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar tipo de datos de una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	VARIANT	Primer operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF TypeOf(#TagVARIANT) = TypeOf("TagBOOL") THEN
...;
END_IF;
```

Si el operando al que apunta el VARIANT "#TagVARIANT" es del tipo de datos BOOL, debe ejecutarse una orden determinada.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

TypeOfElements: Consultar tipo de datos de un elemento de una variable VARIANT

Descripción

Con la instrucción "Consultar tipo de datos de un elemento de una variable VARIANT" se consulta de qué tipo de datos es una variable a la que apunta un VARIANT. El tipo de datos de la variable se compara con el tipo de datos de una variable que se ha declarado en la interfaz de bloque para ver si es "Igual" o "Diferente".

El operando debe ser del tipo de datos VARIANT. El operando de comparación puede ser un tipo de datos simple o un tipo de datos PLC.

Si el tipo de datos de la variable VARIANT es ARRAY, se compara el tipo de datos de los elementos ARRAY.

La instrucción "Consultar tipo de datos de un elemento de una variable VARIANT" solo se puede utilizar dentro de una instrucción IF.

Sintaxis

Para la instrucción "Consultar tipo de datos de un elemento de una variable VARIANT" se utiliza la siguiente sintaxis:

SCL

```
TypeOfElements(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar tipo de datos de un elemento de una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	VARIANT	Primer operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF TypeOfElements(#Tag_VARIANT) = TypeOf("GlobalDB".Product[1]) THEN
"Tag_Result" := "GlobalDB".Product[1] * 3;
END_IF;
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
#Tag_VARIANT	1.5
"GlobalDB".Product[1]	3.5

Si la variable a la que señala VARIANT y el operando ""GlobalDB".Product[1]" son del tipo de datos REAL, el operando "GlobalDB".Product[1]" se multiplica por 3 y el resultado se escribe en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

IS_ARRAY: Consultar si es ARRAY

Descripción

La instrucción "Consultar si es ARRAY" permite saber si VARIANT apunta a una variable del tipo de datos ARRAY.

El <operando> debe ser del tipo de datos VARIANT.

La instrucción "Consultar si es ARRAY" solo se puede utilizar dentro de una instrucción IF.

Sintaxis

Para la instrucción "Consultar si es ARRAY" se utiliza la siguiente sintaxis:

SCL

```
IS_ARRAY(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar si es ARRAY":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	VARIANT	Operando para el que se consulta si es ARRAY
Valor de función		UDINT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF IS_ARRAY(#Tag_VARIANTToArray) THEN
"Tag_Result" := CountOfElements(#Tag_VARIANT);
END_IF;
```

Si la variable a la que señala VARIANT es ARRAY, se devuelve el número de elementos ARRAY.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Funciones matemáticas

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" permite calcular el valor absoluto de un valor de entrada y guardar el resultado en el operando indicado.

Sintaxis

Para la instrucción "Calcular valor absoluto" se utiliza la siguiente sintaxis:

SCL

```
ABS(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
<Expresión>	Input	SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	Valor de entrada
Valor de función		SINT, INT, DINT, números en coma flotante	SINT, INT, DINT, LINT, números en coma flotante	Valor absoluto del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := ABS("Tag_Value");
"Tag_Result2" := ABS("Tag_Value1"*"Tag_Value2");
```

El valor absoluto del valor de entrada se devuelve en el formato del valor de entrada como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	-2
Tag_Result1	2
Tag_Value1	4
Tag_Value2	-1
Tag_Result2	4

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MIN: Determinar mínimo

Descripción

Con la instrucción "Determinar mínimo" se comparan los valores de las entradas disponibles y se devuelve el menor como resultado.

En la instrucción se pueden indicar como mínimo dos y como máximo 32 entradas.

El resultado no es válido si se cumple una de las siguientes condiciones:

- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Determinar mínimo" se utiliza la siguiente sintaxis:

SCL

```
MIN(IN1 := <Operando>,
    IN2 := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN1	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Segundo valor de entrada
INn	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Entradas insertadas adicionalmente cuyos valores se comparan
Valor de función		Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Resultado de la instrucción
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := MIN(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	12222
IN2	Tag_Value2	14444

Parámetro	Operando	Valor
IN3	Tag_Value3	13333
Valor de función	Tag_Result	12222

La instrucción compara los valores de las entradas disponibles y copia el valor menor (Tag_Value1) en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MAX: Determinar máximo

Descripción

Con la instrucción "Determinar máximo" se comparan los valores de las entradas disponibles y se devuelve el mayor como resultado.

En la instrucción se pueden indicar como mínimo dos y como máximo 32 entradas.

El resultado no es válido si se cumple una de las siguientes condiciones:

- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Determinar máximo" se utiliza la siguiente sintaxis:

SCL

```
MAX(IN1 := <Operando>,
    IN2 := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN1	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Segundo valor de entrada
INn	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Entradas insertadas adicionalmente cuyos valores se comparan
Valor de función		Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Resultado de la instrucción
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := MAX(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	12 222
IN2	Tag_Value2	14 444

Parámetro	Operando	Valor
IN3	Tag_Value3	13 333
Valor de función	Tag_Result	14 444

La instrucción compara los valores de los operandos indicados y copia el valor mayor ("Tag_Value2") en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor del parámetro IN a los valores de los parámetros MN y MX. Para ello, el valor del parámetro MN no puede ser mayor que el valor del parámetro MX.

Si el valor del parámetro IN cumple la condición MN <= IN <= MX, se devuelve como resultado de la instrucción. Si no se cumple la condición y el valor de entrada IN rebasa por defecto el límite inferior MN, se devuelve como resultado el valor del parámetro MN. Si se rebasa por exceso el límite superior MX se devuelve como resultado el valor del parámetro MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no está definido.

Para poder ejecutar la instrucción, es imprescindible que los operandos de todos los parámetros sean del mismo tipo de datos.

Sintaxis

Para la instrucción "Ajustar valor límite" se utiliza la siguiente sintaxis:

SCL

```
LIMIT(MN := <Operando>,
      IN := <Operando>,
      MX := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
MN	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Límite inferior
IN	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Valor de entrada
MX	Input	Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Límite superior
Valor de función		Enteros, números en coma flotante, TIME, TOD, DATE, DTL	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT, DT, DTL	Resultado de la instrucción
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := LIMIT(MN := "Tag_Minimum",
                      IN := "Tag_Value",
                      MX := "Tag_Maximum");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_Minimum	12 000
IN	Tag_Value	8 000
MX	Tag_Maximum	16 000
Valor de función	Tag_Result	12 000

El valor del operando "Tag_Value" se compara con los valores de los operandos "Tag_Minimum" y "Tag_Maximum". Puesto que el valor del operando "Tag_Value" es menor

que el valor límite inferior, el valor del operando "Tag_Minimum" se copia en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" permite elevar al cuadrado el valor de entrada y guardar el resultado en el operando indicado.

Sintaxis

Para la instrucción "Calcular cuadrado" se utiliza la siguiente sintaxis:

SCL

```
SQR(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Cuadrado del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := SQR("Tag_Value");
"Tag_Result2" := SQR((SQR("Tag_Value1"))*"Tag_Value2);
```

El cuadrado del valor de entrada se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	2.5
Tag_Result1	6.25
Tag_Value1	6.0
Tag_Value2	2.0
Tag_Result2	5184.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" permite extraer la raíz cuadrada del valor de entrada y guardar el resultado en el operando indicado. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la instrucción devuelve un número en coma flotante no válido. Si el valor de entrada es "-0", el resultado también es "-0".

Sintaxis

Para la instrucción "Calcular raíz cuadrada" se utiliza la siguiente sintaxis:

```
SCL
SQRT(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Raíz cuadrada del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := SQRT("Tag_Value");
"Tag_Result2" := SQRT((SQR("Tag_Value1"))+"Tag_Value2");
```

La raíz cuadrada del valor de entrada se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	4.0
Tag_Result1	2.0
Tag_Value1	3.0
Tag_Value2	16.0
Tag_Result2	5.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2.718282) a partir del valor de entrada. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la instrucción devuelve un número en coma flotante no válido.

Sintaxis

Para la instrucción "Calcular logaritmo natural" se utiliza la siguiente sintaxis:

```
SCL
LN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Logaritmo natural del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := LN("Tag_Value");
"Tag_Result2" := LN("Tag_Value1"+"Tag_Value2");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	2.5
Tag_Result1	0.916
Tag_Value1	1.5
Tag_Value2	3.2
Tag_Result2	1.548

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2.718282) a partir del valor de entrada y guardar el resultado en el operando indicado.

Sintaxis

Para la instrucción "Calcular valor exponencial" se utiliza la siguiente sintaxis:

```
SCL
EXP(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Valor exponencial del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := EXP("Tag_Value");
"Tag_Result2" := EXP("Tag_Value1"/"Tag_Value2");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	20.5
Tag_Result1	799 902 200
Tag_Value1	15.5
Tag_Value2	30.2
Tag_Result2	1.671

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno del valor de entrada. El valor de entrada se ha de indicar en radianes.

Sintaxis

Para la instrucción "Calcular valor de seno" se utiliza la siguiente sintaxis:

SCL

```
SIN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada (tamaño de un ángulo en radianes)
Valor de función		Números en coma flotante	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := SIN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	+1.570796 (π/2)
Tag_Result	1.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno del valor de entrada. El valor de entrada se ha de indicar en radianes.

Sintaxis

Para la instrucción "Calcular valor de coseno" se utiliza la siguiente sintaxis:

```
SCL
COS(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada (tamaño de un ángulo en radianes)
Valor de función		Números en coma flotante	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := COS("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	+1.570796 (π/2)
Tag_Result	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente del valor de entrada. El valor de entrada se ha de indicar en radianes.

Sintaxis

Para la instrucción "Calcular valor de tangente" se utiliza la siguiente sintaxis:

```
SCL
TAN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada (tamaño de un ángulo en radianes)
Valor de función		Números en coma flotante	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := TAN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	+3.141593 (π)
Tag_Result	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno el tamaño del ángulo que equivale a este valor. Como valores de entrada solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se devuelve en radianes y puede estar comprendido entre -π/2 y +π/2.

Sintaxis

Para la instrucción "Calcular valor de arcoseno" se utiliza la siguiente sintaxis:

SCL

```
ASIN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de seno
Valor de función		Números en coma flotante	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ASIN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	1.0
Tag_Result	+1.570796 (π/2)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno el tamaño del ángulo que equivale a este valor. Como valores de entrada solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se devuelve en radianes y puede estar comprendido entre 0 y +π.

Sintaxis

Para la instrucción "Calcular valor de arcocoseno" se utiliza la siguiente sintaxis:

```
SCL
ACOS(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de coseno
Valor de función		Números en coma flotante	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := ACOS("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	0
Tag_Result	+1.570796 (π/2)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente el tamaño del ángulo que equivale a este valor. Como valores de entrada solo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado se devuelve en radianes y puede estar comprendido entre -π/2 y +π/2.

Sintaxis

Para la instrucción "Calcular valor de arcotangente" se utiliza la siguiente sintaxis:

```
SCL
ATAN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de tangente
Valor de función		Números en coma flotante	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := ATAN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	1.0
Tag_Result	+0.785398 (π/4)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

FRAC: Determinar decimales

Descripción

La instrucción "Determinar decimales" devuelve como resultado los decimales de un valor. Así, por ejemplo, el valor de entrada 123.4567 proporciona como resultado el valor 0.4567.

Sintaxis

Para la instrucción "Determinar decimales" se utiliza la siguiente sintaxis:

```
SCL
FRAC(<Expresión>)
```

SCL

```
FRAC_<tipo de datos>(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
_<tipo de datos>		Números en coma flotante Ajuste predeterminado: REAL	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Números en coma flotante	Decimales del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := FRAC("Tag_Value");
"Tag_Result2" := FRAC_LREAL("Tag_Value");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	2.555	-1.4421
Tag_Result1	0.555	-0.4421

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Transferencia

Deserialize: Deserializar

Descripción

Con la instrucción "Deserializar", la forma de representación secuencial de un tipo de datos PLC (UDT) se vuelve a convertir en un tipo de datos PLC y se rellena todo el contenido.

El área de memoria en la que se encuentra la forma de representación secuencial de un tipo de datos PLC debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

Esta instrucción permite reconvertir paso a paso a su estado original varias formas de representación secuenciales de tipos de datos PLC convertidos.

Nota

Para reconvertir una sola forma de representación secuencial de un tipo de datos PLC (UDT) también se puede utilizar directamente la instrucción "TRCV: Recibir datos a través de la conexión".

Sintaxis

Para la instrucción "Deserializar" se utiliza la siguiente sintaxis:

SCL

```
Deserialize(SRC_ARRAY: = <Operando>,
            DEST_VARIABLE := <Operando>,
            POS := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Deserializar":

Parámetro	Declaración	Tipo de datos	Descripción
SRC_ARRAY	Input	VARIANT	Bloque de datos global en el que se guarda el flujo de datos creado
DEST_VARIABLE	InOut	VARIANT	Variable en la que se guarda el tipo de datos PLC (UDT) reconvertido
POS	InOut	DINT	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
Valor de función		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_ARRAY y DEST_VARIABLE se solapan.
8136	El bloque de datos del parámetro DEST_VARIABLE no es un bloque con acceso estándar.
8150	El tipo de datos VARIANT del parámetro SRC_ARRAY no contiene ningún valor.
8151	Error de creación de código en el parámetro SRC_ARRAY
8153	En el parámetro SRC_ARRAY no hay suficiente memoria disponible.
8250	El tipo de datos VARIANT del parámetro DEST_VARIABLE no contiene ningún valor.
8251	Error de creación de código en el parámetro DEST_VARIABLE
8254	Tipo de datos no válido en el parámetro DEST_VARIABLE
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := "Target".Client,
                           POS := #BufferPos);

#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := #Label,
                           POS := #BufferPos);

IF #Label = 'arti' THEN
#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := "Target".Article[#DeliverPos],
                           POS := #BufferPos);
ELSIF #Label = 'Bill' THEN
#Tag_RetVal := Deserialize(SRC_ARRAY := "Buffer".Field,
                           DEST_VARIABLE := "Target".Bill[#DeliverPos],
                           POS := #BufferPos);
```

SCL

```
;
ELSE
;
END_IF;
```

La instrucción "Copiar valor" copia el valor "0" en el operando "#BufferPos". La instrucción "Deserializar" deserializa la forma de representación secuencial de los datos de cliente del bloque de datos "Buffer" y los escribe en el bloque de datos "Target". El número de bytes que ocupan los datos de cliente convertidos se guarda en el operando "#BufferPos".

La instrucción "Deserializar" deserializa la forma de representación secuencial del separador (depositado después de los datos de cliente en la forma de representación secuencial) del bloque de datos "Buffer" y escribe los caracteres en el operando "#Label". Los caracteres se someten a una comparación con "arti" y "Bill" mediante instrucciones de comparación. Si la comparación en relación con "arti" = TRUE, se trata de datos de artículo que se deserializan y se escriben en el bloque de datos "Target". Si la comparación en relación con "Bill" = TRUE, se trata de datos de facturas que se deserializan y se escriben en el bloque de datos "Target".

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Target	Client	"Client" (tipo de datos PLC)
	Article	Array[0..] of "Article" (tipo de datos PLC)
	Bill	Array[0..10] of INT
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

Serialize: Serializar

Descripción

Mediante la instrucción "Serializar" se convierten varios tipos de datos PLC (UDT) a una forma de representación secuencial sin que se pierdan partes de su estructura.

La instrucción permite guardar temporalmente varios datos estructurados del programa en un búfer, ubicado preferentemente en un bloque de datos global, y enviarlos a otra CPU. El área de memoria en la que se depositen los tipos de datos PLC convertidos debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

El operando del parámetro POS contiene la información del número de bytes que ocupan los tipos de datos PLC convertidos.

Nota

Para enviar un solo tipo de datos PLC (UDT), se puede llamar directamente la instrucción "TSEND: Enviar datos a través de la conexión".

Sintaxis

Para la instrucción "Serializar" se utiliza la siguiente sintaxis:

SCL

```
Serialize(SRC_VARIABLE: = <Operando>,
          DEST_ARRAY := <Operando>,
          POS := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Serializar":

Parámetro	Declaración	Tipo de datos	Descripción
SRC_VARIABLE	Input	VARIANT	Tipo de datos PLC (UDT), que se convierte a una forma de representación secuencial.
DEST_ARRAY	InOut	VARIANT	Bloque de datos en el que se guarda el flujo de datos generado.
POS	InOut	DINT	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
Valor de función		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_VARIABLE y DEST_ARRAY se solapan.
8150	El tipo de datos VARIANT del parámetro SRC_VARIABLE no contiene ningún valor.
8152	Error de creación de código en el parámetro SRC_VARIABLE
8236	El bloque de datos del parámetro DEST_ARRAY no es un bloque con acceso estándar.
8250	El tipo de datos VARIANT del parámetro DEST_ARRAY no contiene ningún valor.
8252	Error de creación de código en el parámetro DEST_ARRAY
8253	En el parámetro DEST_ARRAY no hay suficiente memoria disponible.
8254	Tipo de datos no válido en el parámetro DEST_ARRAY
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#Tag_RetVal := Serialize(SRC_VARIABLE := "Source".Client,
                         DEST_ARRAY := "Buffer".Field,
```

SCL

```
                        POS := #BufferPos);

#Label := STRING_TO_WSTRING('arti');

#Tag_RetVal := Serialize(SRC_VARIABLE := #Label,
                        DEST_ARRAY := "Buffer".Field,
                        POS := #BufferPos);

#Tag_RetVal := Serialize(SRC_VARIABLE := "Source".Article[#DeliverPos],
                        DEST_ARRAY := "Buffer".Field,
                        POS := #BufferPos);
```

La instrucción "Copiar valor" copia el valor "0" en el operando "#BufferPos". La instrucción "Serializar" serializa los datos de cliente del bloque de datos "Source" y los escribe como representación secuencial en el bloque de datos "Buffer". El número de bytes que ocupa la representación secuencial se guarda en el operando "#BufferPos".

Para facilitar la posterior deserialización de la forma de representación secuencial, ahora se inserta una especie de separador. La instrucción "Desplazar cadena de caracteres" copia los caracteres "arti" en el operando "#Label". La instrucción "Serializar" escribe estos caracteres después de los datos de cliente en el bloque de datos "Buffer". El número de bytes que requieren los caracteres se suma en el operando "#BufferPos" al número ya guardado.

La instrucción "Serializar" serializa los datos de un determinado artículo, que se calcula en tiempo de ejecución, del bloque de datos "Source", y los escribe en forma de representación secuencial después de los caracteres "arti" en el bloque de datos "Buffer".

La tabla siguiente muestra la declaración de los operandos:

Operando	Tipo de datos	Declaración
DeliverPos	INT	En la sección "Input" de la interfaz del bloque
BufferPos	DINT	En la sección "Temp" de la interfaz del bloque
Error	INT	En la sección "Temp" de la interfaz del bloque
Label	STRING[4]	En la sección "Temp" de la interfaz del bloque

La siguiente tabla muestra la declaración de los tipos de datos PLC:

Nombre de los tipos de datos PLC	Nombre	Tipo de datos
Article	Number	DINT
	Declaration	STRING
	Colli	INT
Client	Title	INT
	First name	STRING[10]
	Surname	STRING[10]

La tabla siguiente muestra la declaración de los bloques de datos:

Nombre de los bloques de datos	Nombre	Tipo de datos
Source	Client	"Client" (tipo de datos PLC)
	Article	Array[0..10] of "Article" (tipo de datos PLC)
Buffer	Field	Array[0..294] of BYTE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipos de datos PLC (Página 2367)

MOVE_BLK: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del primer elemento del área de origen define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en el parámetro IN o en el parámetro OUT.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

SCL

```
MOVE_BLK(IN := <Operando>,
         COUNT := <Operando>,
         OUT => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos		
		S7-1200	S7-1500	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de elementos que se copian del área de origen al área de destino
OUT [1]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
MOVE_BLK(IN := #a_array[2],
         COUNT := "Tag_Count",
         OUT => #b_array[1]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MOVE_BLK_VARIANT: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). Se pueden copiar elementos de un array en otro array del mismo tipo de datos. Los tamaños (número de elementos) de los arrays de origen y de destino pueden ser diferentes. Se pueden copiar varios elementos dentro de un array o bien copiar los elementos uno a uno.

En el momento de utilizar la instrucción para crear un bloque, aún no tiene que conocerse el array, puesto que el origen y el destino se transfieren por VARIANT.

El contaje de los parámetros SRC_INDEX y DEST_INDEX siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La instrucción no se ejecuta si se copian más datos de los que están disponibles.

Si el puntero VARIANT (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

SCL

```
MOVE_BLK_VARIANT(SRC := <Operando>,
                 COUNT := <Operando>,
                 SRC_INDEX := <Operando>,
                 DEST_INDEX := <Operando>,
                 DEST => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Descripción
SRC	Input	VARIANT (array o elemento individual)	El área de origen de la que se copia
COUNT	Input	UDINT	Número de elementos que se copian Asigne al parámetro COUNT el valor "1" si en el parámetro SRC o en el parámetro DEST no hay indicado ningún Array.
SRC_INDEX	Input	DINT	• El parámetro SRC_INDEX se calcula basado en cero. Si en el parámetro SRC hay indicado un Array, el entero del parámetro SRC_INDEX indica el primer elemento del área de origen de la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro SRC no hay indicado ningún Array o solo hay indicado un elemento individual de un Array, asigne al parámetro SRC_INDEX el valor "0".
DEST_INDEX	Input	DINT	• El parámetro DEST_INDEX se calcula basado en cero. Si en el parámetro DEST hay indicado un Array, el entero del parámetro DEST_INDEX indica el primer elemento del área de destino en la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro DEST no hay indicado ningún Array, asigne al parámetro DEST_INDEX el valor "0".

Parámetro	Declaración	Tipo de datos	Descripción
DEST	Output	VARIANT	Área de destino a la que se copia el contenido del área de origen.
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
8151	No es posible acceder al parámetro SRC.
8152	El operando del parámetro SRC no es de ningún tipo.
8153	Error de creación de código en el parámetro SRC
8154	El operando del parámetro SRC es del tipo de datos BOOL.
8281	Valor no válido en el parámetro COUNT
8382	El valor del parámetro SRC_INDEX se encuentra fuera de los valores límite de VARIANT.
8383	El valor del parámetro SRC_INDEX se encuentra fuera del límite superior del array.
8482	El valor del parámetro DEST_INDEX se encuentra fuera de los valores límite de VARIANT.
8483	El valor del parámetro DEST_INDEX se encuentra fuera del límite superior del array.
8534	El parámetro DEST está protegido contra escritura
8551	No es posible acceder al parámetro DEST.
8552	El operando del parámetro DEST no es de ningún tipo.
8553	Error de creación de código en el parámetro DEST
8554	El operando del parámetro DEST es del tipo de datos BOOL.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := MOVE_BLK_VARIANT(SRC := #SrcField,
                                 COUNT := "Tag_Count",
                                 SRC_INDEX := "Tag_Src_Index",
```

SCL

```
DEST_INDEX := "Tag_Dest_Index",
DEST => #DestField);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC	#SrcField	El operando local #SrcField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [0..10] of "MOVE_UDT"
COUNT	Tag_Count	2
SRC_INDEX	Tag_Src_Index	3
DEST_INDEX	Tag_Dest_Index	3
DEST	#DestField	El operando local #DestField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [10..20] of "MOVE_UDT"

En el área de destino se copian 2 elementos del área de origen, empezando por el cuarto elemento del array de UDT. Las copias se insertan en el Array of UDT a partir del cuarto elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VariantGet: Leer valor de una variable VARIANT (Página 4402)

UMOVE_BLK: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia sin interrupciones los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del primer elemento del área de origen define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Copiar área sin interrupciones".

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en el parámetro IN o en el parámetro OUT.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

SCL

```
UMOVE_BLK(IN := <Operando>,
          COUNT := <Operando>,
          OUT => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de elementos que se copian del área de origen al área de destino
OUT [1]	Output	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Primer elemento del área de destino en la que se copian los contenidos del área de origen

[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
UMOVE_BLK(IN := #a_array[2],
          COUNT := "Tag_Count",
          OUT => #b_array[1]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

FILL_BLK: Rellenar área

Descripción

La instrucción "Rellenar área" permite rellenar un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el área de origen y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

SCL

```
FILL_BLK(IN := <Operando>,
         COUNT := <Operando>,
         OUT => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, TOD, DATE, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, TIME, TOD, DATE, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Dirección del área de destino a partir de la cual se rellena

[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY.
[2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
FILL_BLK(IN := #a_array[2],
         COUNT := "Tag_Count",
         OUT => #b_array[1]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

UFILL_BLK: Rellenar área sin interrupciones

Descripción

La instrucción "Rellenar área sin interrupciones" permite rellenar un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen) sin interrupciones. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor de la entrada IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Rellenar área sin interrupciones".

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

La instrucción "Rellenar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Sintaxis

Para la instrucción "Rellenar área sin interrupciones" se utiliza la siguiente sintaxis:

SCL

```
UFILL_BLK(IN := <Operando>,
          COUNT := <Operando>,
          OUT => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área sin interrupciones":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN [1]	Input	Números binarios, enteros, números en coma flotante, TIME, DATE, CHAR, TOD	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, TIME, TOD, DATE, CHAR	Números binarios, enteros, números en coma flotante, temporizadores, DATE, CHAR, TOD, LTOD	Dirección del área de destino a partir de la cual se rellena
[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY. [2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
UFILL_BLK(IN := #a_array[2],
          COUNT := "Tag_Count",
          OUT => #b_array[1]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

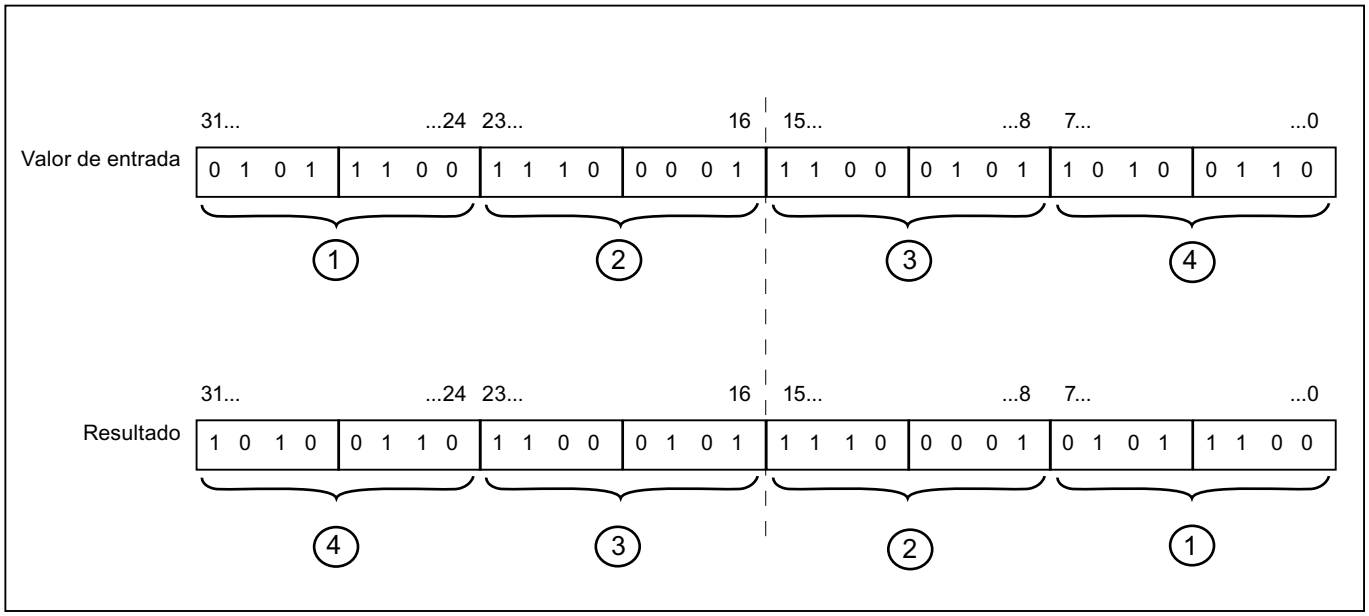
Edición de instrucciones SCL (Página 2823)

SWAP: Cambiar disposición

Descripción

La instrucción "Cambiar disposición" permite cambiar la disposición de los bytes de un valor de entrada y guardar el resultado en el operando indicado.

La figura siguiente muestra cómo se intercambian los bytes de un operando del tipo de datos DWORD utilizando la instrucción "Cambiar disposición":



Sintaxis

Para la instrucción "Cambiar disposición" se utiliza la siguiente sintaxis:

```
SCL
SWAP(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
<Expresión>	Input	WORD, DWORD	WORD, DWORD, LWORD	Valor de entrada
Valor de función		WORD, DWORD	WORD, DWORD, LWORD	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := SWAP("Tag_Value");
```

El resultado de la instrucción se devuelve como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	0000 1111 0101 0101
Tag_Result	0101 0101 0000 1111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Instrucciones ARRAY-DB

ReadFromArrayDB: Leer en un bloque de datos ARRAY

Descripción

La instrucción "Leer en un bloque de datos ARRAY" permite leer datos en un DB array y escribirlos en un área de destino.

Un DB array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Sintaxis

Para la instrucción "Leer en un bloque de datos ARRAY" se utiliza la siguiente sintaxis:

SCL

```
ReadFromArrayDB(DB := <Operando>,
                INDEX := <Operando>,
                VALUE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer en un bloque de datos ARRAY":

Parámetro	Declaración	Tipo de datos	Descripción
DB	Input	DB_ANY	Bloque de datos desde el que se lee
INDEX	Input	DINT	Elemento que se lee
VALUE	Output	VARIANT	Valor que se lee y se devuelve
Valor de función (RET_VAL)		INT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El DB array contiene valores no válidos
8154	El tipo de datos no está declarado correctamente
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TagResult" := ReadFromArrayDB(DB := "ArrayDB",
                               INDEX := "ArrayDB"."THIS"[2],
                               VALUE => "TargetField[10]".Data2[1]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB."THIS"[2]	Tercer elemento del "ArrayDB"
VALUE	TargetField[10].Data2[1]	El operando "TargetField" es del tipo de datos Array [10..20] of "MOVE_UDT".

El tercer elemento se lee en el "ArrayDB" y se escribe en el operando "TargetField[10].Data2[1]".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

WriteToArrayDB: Escribir en un bloque de datos ARRAY

Descripción

La instrucción "Escribir en un bloque de datos ARRAY" permite escribir datos en un DB array.

Un DB array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Sintaxis

Para la instrucción "Escribir en un bloque de datos ARRAY" se utiliza la siguiente sintaxis:

SCL

```
WriteToArrayDB(DB := <Operando>,
               INDEX := <Operando>,
               VALUE := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en un bloque de datos ARRAY":

Parámetro	Declaración	Tipo de datos	Descripción
DB	Input	DB_ANY	Bloque de datos en el que se escribe
INDEX	Input	DINT	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	Valor que se escribe
Valor de función (RET_VAL)		INT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TagResult" := WriteToArrayDB(DB := "ArrayDB",
                              INDEX := "ArrayDB"."THIS"[2],
                              VALUE := "SourceField[1]".Data1[6]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB."THIS"[2]	Tercer elemento del "ArrayDB"
VALUE	SourceField[1].Data1[6]	El operando "SourceField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Del operando "SourceField" se escribe el elemento "Data1[6]" del segundo elemento en el "ArrayDB". En el "ArrayDB" se escribe el tercer elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ReadFromArrayDBL: Leer en un bloque de datos ARRAY de la memoria de carga

Descripción

La instrucción "Leer en un bloque de datos ARRAY de la memoria de carga" permite leer datos desde un DB array.

Un DB array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1" y el valor leído en el parámetro VALUE se emite dentro de dicho ciclo. En los demás ciclos del programa, el valor del parámetro VALUE no se modifica.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

Sintaxis

Para la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga" se utiliza la siguiente sintaxis:

SCL

```
"DB de instancia"(REQ := <Operando>,
            DB := <Operando>,
            INDEX := <Operando>,
            BUSY => <Operando>,
```

SCL

```
DONE => <Operando>,
ERROR => <Operando>,
VALUE := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Descripción
REQ	Input	BOOL	REQ = "1": Comenzar la lectura del DB array
DB	Input	DB_ANY	Bloque de datos desde el que se lee
INDEX	Input	DINT	Elemento que se lee
BUSY	Output	BOOL	BUSY = "1": La lectura del DB array todavía no ha finalizado
DONE	Output	BOOL	DONE = "1": La instrucción se ha ejecutado correctamente
VALUE	InOut	VARIANT	Valor que se lee y se devuelve
ERROR	Output	INT	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga

Código de error* (W#16#...)	Explicación
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"ReadFromArrayDBL_DB"(REQ := "TagReg",
                      DB := "ArrayDB",
                      INDEX := "ArrayDB"."THIS"[2],
                      BUSY => "TagBusy",
                      DONE => "TagDone",
                      ERROR => "TagError",
                      VALUE := "SourceTargetField[1]".Data1[6]);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB."THIS"[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga". El tercer elemento se lee en el "ArrayDB" y se indica en el operando "SourceTargetField[1].Data1[6]". En cuanto se detecta

un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE ya no se modifica. Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

WriteToArrayDBL: Escribir en un bloque de datos ARRAY de la memoria de carga

Descripción

La instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga" permite escribir datos en un DB array de la memoria de carga.

Un DB array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción y el valor del parámetro VALUE se escribe en el bloque de datos. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

Sintaxis

Para la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga" se utiliza la siguiente sintaxis:

SCL

```
"DB de instancia"(REQ := <Operando>,
            DB := <Operando>,
            INDEX := <Operando>,
            VALUE := <Operando>,
            BUSY => <Operando>,
            DONE => <Operando>,
            ERROR => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Descripción
REQ	Input	BOOL	REQ = "1": Comenzar la escritura en el DB array
DB	Input	DB_ANY	Bloque de datos en el que se escribe
INDEX	Input	DINT	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	Valor que se escribe
BUSY	Output	BOOL	BUSY = "1": La escritura en el DB array todavía no ha finalizado
DONE	Output	BOOL	DONE = "1": La instrucción se ha ejecutado correctamente
ERROR	Output	INT	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"WriteToArrayDBL_DB"(REQ := "TagReg",
                     DB := "ArrayDB",
                     INDEX := "ArrayDB"."THIS"[2],
                     VALUE := "SourceTargetField[1]".Data1[6],
                     BUSY => "TagBusy",
                     DONE => "TagDone",
                     ERROR => "TagError");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB."THIS"[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE se escribe en el "ArrayDB" en el tercer elemento. Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

READ_DBL: Leer de un bloque de datos de la memoria de carga (Página 4997)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga (Página 4999)

Acceso de lectura/escritura

PEEK: Leer dirección de memoria

Descripción

La instrucción "Leer dirección de memoria" permite leer una dirección de memoria de un área de memoria sin indicar un tipo de datos.

Si en el parámetro AREA se indica el área 16#84 para un bloque de datos, solo será posible acceder a bloques de datos con la propiedad de bloque "Estándar".

Sintaxis

Para la instrucción "Leer dirección de memoria" se utiliza la siguiente sintaxis:

SCL

```
PEEK(AREA := <Operando>,
     DBNUMBER := <Operando>,
     BYTEOFFSET := <Operando>)
```

SCL

```
PEEK_<tipo de datos>(AREA := <Operando>,
                DBNUMBER := <Operando>,
                BYTEOFFSET := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
AREA	Input	BYTE	Pueden seleccionarse las siguientes áreas: • 16#81: Input • 16#82: Output • 16#83: Marcas • 16#84: DB • 16#1: Entrada de periferia (solo S7-1500)
DBNUMBER	Input	DINT, DB_ANY	Número del bloque de datos, si AREA = DB, de lo contrario "0"
BYTEOFFSET	Input	DINT	Dirección en la que se lee Solo se utilizan los 16 bits menos significativos.
_<tipo de datos>		Secuencias de bits valor predeterminado: BYTE	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Secuencias de bits	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Nota

Si se lee la dirección de memoria en las áreas Input, Output o Marcas, se debe aplicar el valor "0" en el parámetro "DBNUMBER", ya que de lo contrario la instrucción será errónea.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := PEEK(AREA := "Tag_Area",
                      DBNUMBER := "Tag_DBNumber",
                      BYTEOFFSET := "Tag_Byte");
```

SCL

```
"Tag_Result2" := PEEK_WORD(AREA := "Tag_Area",
                           DBNUMBER := "Tag_DBNumber",
                           BYTEOFFSET := "Tag_Byte");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
Valor de función	Tag_Result1	Valor del byte "20"
Valor de función	Tag_Result2	Valor de la palabra "20"

La instrucción lee el valor de la dirección "20" del operando "Tag_Byte" del bloque de datos "5" y devuelve el resultado como valor de la función en el operando "Tag_Result".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

PEEK_BOOL: Leer bit de memoria

Descripción

La instrucción "Leer bit de memoria" permite leer un bit memoria de un área de memoria sin indicar un tipo de datos.

Si en el parámetro AREA se indica el área 16#84 para un bloque de datos, solo será posible acceder a bloques de datos con la propiedad de bloque "Estándar".

Nota

Con la instrucción solo se puede acceder a áreas de memoria "Estándar".

Sintaxis

Para la instrucción "Leer bit de memoria" se utiliza la siguiente sintaxis:

SCL

```
PEEK_BOOL(AREA := <Operando>,
          DBNumber := <Operando>,
          BYTEEOFFSET := <Operando>,
          BITOFFSET := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
AREA	Input	BYTE	Pueden seleccionarse las siguientes áreas: • 16#81: Input • 16#82: Output • 16#83: Marcas • 16#84: DB • 16#1: Entrada de periferia (solo S7-1500)
DBNUMBER	Input	DINT, DB_ANY	Número del bloque de datos, si AREA = DB, de lo contrario "0"
BYTEOFFSET	Input	DINT	Dirección en la que se lee Solo se utilizan los 16 bits menos significativos.
BITOFFSET	Input	INT	Bit en el que se lee
Valor de función		BOOL	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Nota

Si se lee el bit de memoria en las áreas Input, Output o Marcas, se debe aplicar el valor "0" en el parámetro "DBNUMBER", ya que de lo contrario la instrucción será errónea.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := PEEK_BOOL(AREA := "Tag_Area",
                          DBNUMBER := "Tag_DBNumber",
                          BYTEOFFSET := "Tag_Byte",
                          BITOFFSET := "Tag_Bit");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
BITOFFSET	Tag_Bit	3
Valor de función	Tag_Result	3

La instrucción lee el valor del bit de memoria "3" del operando "Tag_Bit" del bloque de datos "5" del byte "20" y devuelve el resultado como valor de función en el operando "Tag_Result".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

POKE: Escribir dirección de memoria

Descripción

La instrucción "Escribir dirección de memoria" permite escribir una dirección de memoria en un área de memoria sin indicar un tipo de datos.

Si en el parámetro AREA se indica el área 16#84 para un bloque de datos, solo será posible acceder a bloques de datos con la propiedad de bloque "Estándar".

Nota

Con la instrucción solo se puede acceder a áreas de memoria "Estándar".

Sintaxis

Para la instrucción "Escribir dirección de memoria" se utiliza la siguiente sintaxis:

SCL

```
POKE(AREA := <Operando>,
     DBNUMBER := <Operando>,
     BYTEOFFSET := <Operando>,
     VALUE := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
AREA	Input	BYTE	Pueden seleccionarse las siguientes áreas: • 16#81: Input • 16#82: Output • 16#83: Marcas • 16#84: DB • 16#2: Salida de periferia (solo S7-1500)
DBNUMBER	Input	DINT, DB_ANY	Número del bloque de datos, si AREA = DB, de lo contrario "0"
BYTEOFFSET	Input	DINT	Dirección que se escribe Solo se utilizan los 16 bits menos significativos.
VALUE	Input	Secuencias de bits	Valor que se escribe

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Nota

Si se escribe la dirección de memoria en las áreas Input, Output o Marcas, se debe aplicar el valor "0" en el parámetro "DBNUMBER", ya que de lo contrario la instrucción será errónea.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
POKE(AREA := "Tag_Area",
     DBNUMBER := "Tag_DBNumber",
     BYTEOFFSET := "Tag_Byte"),
     VALUE := "Tag_Value";
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
VALUE	Tag_Value	16#11

La instrucción sobrescribe la dirección de memoria "20" en el bloque de datos "5" con el valor "16#11".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

POKE_BOOL: Escribir bit de memoria

Descripción

La instrucción "Escribir bit de memoria" permite escribir un bit de memoria en un área de memoria sin indicar un tipo de datos.

Si en el parámetro AREA se indica el área 16#84 para un bloque de datos, solo será posible acceder a bloques de datos con la propiedad de bloque "Estándar".

Nota

Con la instrucción solo se puede acceder a áreas de memoria "Estándar".

Sintaxis

Para la instrucción "Escribir bit de memoria" se utiliza la siguiente sintaxis:

SCL

```
POKE_BOOL(AREA := <Operando>,
          DBNUMBER := <Operando>,
          BYTEOFFSET := <Operando>,
          BITOFFSET := <Operando>,
          VALUE := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
AREA	Input	BYTE	Pueden seleccionarse las siguientes áreas: • 16#81: Input • 16#82: Output • 16#83: Marcas • 16#84: DB • 16#2: Salida de periferia (solo S7-1500)
DBNUMBER	Input	DINT, DB_ANY	Número del bloque de datos, si AREA = DB, de lo contrario "0"
BYTEOFFSET	Input	DINT	Dirección que se escribe Solo se utilizan los 16 bits menos significativos.
BITOFFSET	Input	INT	Bit que se escribe
VALUE	Input	BOOL	Valor que se escribe

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Nota

Si se escribe el bit de memoria en las áreas Input, Output o Marcas, se debe aplicar el valor "0" en el parámetro "DBNUMBER", ya que de lo contrario la instrucción será errónea.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
POKE_BOOL(AREA := "Tag_Area",
          DBNUMBER := "Tag_DBNumber",
          BYTEOFFSET := "Tag_Byte",
          BITOFFSET := "Tag_Bit",
          VALUE := "Tag_Value");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
AREA	Tag_Area	16#84
DBNUMBER	Tag_DBNumber	5
BYTEOFFSET	Tag_Byte	20
BITOFFSET	Tag_Bit	3
VALUE	Tag_Value	M0.0

La instrucción sobrescribe el bit de memoria "3" en el bloque de datos "5" en el byte "20" con el valor "M0.0".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

POKE_BLK: Escribir área de memoria

Descripción

La instrucción "Escribir área de memoria" permite escribir un área de memoria en un área de memoria diferente sin indicar un tipo de datos.

Si en el parámetro AREA se indica el área 16#84 para un bloque de datos, solo será posible acceder a bloques de datos con la propiedad de bloque "Estándar".

Nota

Con la instrucción solo se puede acceder a áreas de memoria "Estándar".

Sintaxis

Para la instrucción "Escribir área de memoria" se utiliza la siguiente sintaxis:

SCL

```
POKE_BLK(AREA_SRC := <Operando>,
         DBNUMBER_SRC := <Operando>,
         BYTEOFFSET_SRC := <Operando>,
         AREA_DEST := <Operando>,
         DBNUMBER_DEST := <Operando>,
         BYTEOFFSET_DEST := <Operando>,
         COUNT := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
AREA_SRC	Input	BYTE	Las siguientes áreas pueden seleccionarse en el área de memoria de origen: • 16#81: Input • 16#82: Output • 16#83: Marcas • 16#84: DB
DBNUMBER_SRC	Input	DINT, DB_ANY	Número del bloque de datos en el área de memoria de origen, si AREA = DB, de lo contrario "0"
BYTEOFFSET_SRC	Input	DINT	Dirección en el área de memoria de origen, en la que se escribe Solo se utilizan los 16 bits menos significativos.
AREA_DEST	Input	BYTE	Las siguientes áreas pueden seleccionarse en el área de memoria de destino: • 16#81: Input • 16#82: Output • 16#83: Marcas • 16#84: DB
DBNUMBER_DEST	Input	DINT, DB_ANY	Número del bloque de datos en el área de memoria de destino, si AREA = DB, de lo contrario "0"
BYTEOFFSET_DEST	Input	DINT	Dirección en el área de memoria de destino, en la que se escribe Solo se utilizan los 16 bits menos significativos.
COUNT	Input	DINT	Número de bytes que se copian

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Nota

Si se escribe la dirección de memoria en las áreas Input, Output o Marcas, se debe aplicar el valor "0" en el parámetro "DBNUMBER", ya que de lo contrario la instrucción será errónea.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
POKE_BLK(AREA_SRC := "Tag_Source_Area",
         DBNUMBER_SRC := "Tag_Source_DBNumber",
         BYTEOFFSET_SRC := "Tag_Source_Byte"),
         AREA_DEST := "Tag_Destination_Area",
         DBNUMBER_DEST := "Tag_Destination_DBNumber",
         BYTEOFFSET_DEST := "Tag_Destination_Byte",
         COUNT := "Tag_Count");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
AREA_SRC	Tag_Source_Area	16#84
DBNUMBER_SRC	Tag_Source_DBNumber	5
BYTEOFFSET_SRC	Tag_Source_Byte	20
AREA_DEST	Tag_Destination_Area	16#83
DBNUMBER_DEST	Tag_Destination_DBNumber	0
BYTEOFFSET_DEST	Tag_Destination_Byte	30
COUNT	Tag_Count	100

La instrucción escribe 100 bytes del bloque de datos "5" comenzando por la dirección "20" en el área de memoria Marcas comenzado por la dirección "30".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

READ_LITTLE: Leer datos en formato Little-Endian

Descripción

La instrucción "Leer datos en formato Little-Endian" permite leer datos en una única variable de la orden de bytes Little-Endian. En el formato Little-Endian, primero se guarda el byte con los bits menos significativos, es decir, en la dirección de memoria más baja. El área de memoria en la que se lee debe ser del tipo de datos ARRAY of BYTE.

El operando del parámetro DEST_VARIABLE debe ser de un tipo de datos elemental. El operando del parámetro POS determina la posición en la que se inicia la lectura.

Nota

Leer variable del tipo de datos VARIANT o BOOL

Para leer una variable a la que apunta un VARIANT se pueden utilizar las instrucciones "Serializar" o "Deserializar".

Para leer una variable del tipo de datos BOOL se puede utilizar un "acceso slice".

Sintaxis

Para la instrucción "Leer datos en formato Little-Endian" se utiliza la siguiente sintaxis:

SCL

```
READ_LITTLE(SRC_ARRAY := <Operando>,
            DEST_VARIABLE => <Operando>,
            POS := <Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Leer datos en formato Little-Endian":

Parámetro	Declaración	Tipo de datos	Descripción
SRC_ARRAY	Input	ARRAY of BYTE	Área de memoria en la que se lee
DEST_VARIABLE	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, LDT, TOD, LTOD, DATE, CHAR, WCHAR	Valor leído
POS	InOut	DINT	Determina la posición en la que se inicia la lectura. El parámetro POS se calcula basado en cero.
Valor de función		INT	Información de error

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#TagResult := READ_LITTLE(SRC_ARRAY := #SourceField,
                          DEST_VARIABLE => #DINTVariable,
                          POS := #TagPos);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC_ARRAY	#SourceField	ARRAY[0..3] of BYTE := 16#1A, 16#2B, 16#3C, 16#4D
DEST_VARIABLE	#DINTVariable	1295788826 16#4D3C2B1A
POS	#TagPos	0 => 4

La instrucción "Leer datos en formato Little-Endian" lee el entero 1_295_788_826 del área de memoria "#SourceField" y lo escribe en formato Little-Endian en el operando "#DINTVariable". El tipo de datos del parámetro DEST_VARIABLE indica cuántos bytes se leen. En el operando "#TagPos" se guarda el número 4.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Deserialize: Deserializar (Página 4355)

Serialize: Serializar (Página 4358)

Direccionar áreas de una variable con accesos Slice (Página 2301)

WRITE_LITTLE: Escribir datos en formato Little-Endian

Descripción

La instrucción "Escribir datos en formato Little-Endian" permite escribir los datos de una única variable de la orden de bytes Little-Endian en un área de memoria. En el formato Little-Endian, primero se guarda el byte con los bits menos significativos, es decir, en la dirección de memoria más baja. El área de memoria en la que se escribe debe ser del tipo de datos ARRAY of BYTE.

El operando del parámetro SRC_VARIABLE debe ser de un tipo de datos elemental. El operando del parámetro POS determina la posición en la que se inicia la escritura.

Nota

Escribir variable del tipo de datos VARIANT o BOOL

Para escribir una variable a la que apunta un VARIANT se pueden utilizar las instrucciones "Serializar" o "Deserializar".

Para escribir una variable del tipo de datos BOOL se puede utilizar un "acceso slice".

Sintaxis

Para la instrucción "Escribir datos en formato Little-Endian" se utiliza la siguiente sintaxis:

SCL

```
WRITE_LITTLE(SRC_VARIABLE := <Operando>,
             DEST_ARRAY := <Operando>,
             POS := <Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Escribir datos en formato Little-Endian":

Parámetro	Declaración	Tipo de datos	Descripción
SRC_VARIABLE	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, LDT, TOD, LTOD, DATE, CHAR, WCHAR	Variable cuyos datos se escriben
DEST_ARRAY	InOut	ARRAY of BYTE	Área de memoria en la que se escriben los datos
POS	InOut	DINT	Determina la posición en la que se inicia la escritura. El parámetro POS se calcula basado en cero.
Valor de función		INT	Información de error

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#TagResult := WRITE_LITTLE(SRC_VARIABLE := #DINTVariable,
                           DEST_ARRAY := #TargetField,
                           POS := #TagPos);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC_VARIABLE	#DINTVariable	1295788826 16#4D3C2B1A
DEST_ARRAY	#TargetField	ARRAY[0..10] of BYTE = 16#1A, 16#2B, 16#3C, 16#4D
POS	#TagPos	0 => 4

La instrucción "Escribir datos en formato Little-Endian" escribe el entero 1_295_788_826 del operando "#DINTVariable" en formato Little-Endian en el área de memoria "#TargetField". El tipo de datos del parámetro SRC_VARIABLE indica cuántos bytes se escriben. En el operando "#TagPos" se guarda el número 4.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Deserialize: Deserializar (Página 4355)

Serialize: Serializar (Página 4358)

Direccionar áreas de una variable con accesos Slice (Página 2301)

READ_BIG: Leer datos en formato Big-Endian

Descripción

La instrucción "Leer datos en formato Big-Endian" permite leer datos en una única variable de la orden de bytes Big-Endian. En el formato Big-Endian, primero se guarda el byte con los bits más significativos, es decir, en la dirección de memoria más baja. El área de memoria en la que se lee debe ser del tipo de datos ARRAY of BYTE.

El operando del parámetro DEST_VARIABLE debe ser un tipo de datos elemental. El operando del parámetro POS determina la posición en la que se inicia la lectura.

Nota

Leer variable del tipo de datos VARIANT o BOOL

Para leer una variable a la que apunta un VARIANT se pueden utilizar las instrucciones "Serializar" o "Deserializar".

Para leer una variable del tipo de datos BOOL se puede utilizar un "acceso slice".

Sintaxis

Para la instrucción "Leer datos en formato Big-Endian" se utiliza la siguiente sintaxis:

SCL

```
READ_BIG(SRC_ARRAY := <Operando>,
         DEST_VARIABLE => <Operando>,
         POS := <Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Leer datos en formato Big-Endian":

Parámetro	Declaración	Tipo de datos	Descripción
SRC_ARRAY	Input	ARRAY of BYTE	Área de memoria en la que se lee
DEST_VARIABLE	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, LDT, TOD, LTOD, DATE, CHAR, WCHAR	Valor leído
POS	InOut	DINT	Determina la posición en la que se inicia la lectura. El parámetro POS se calcula basado en cero.
Valor de función		INT	Información de error

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#TagResult := READ_BIG(SRC_ARRAY := #SourceField,
                       DEST_VARIABLE => #DINTVariable,
                       POS := #TagPos);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC_ARRAY	#SourceField	ARRAY[0..10] of BYTE := 16#1A, 16#2B, 16#3C, 16#4D
DEST_VARIABLE	#DINTVariable	439041101 16#1A2B3C4D
POS	#TagPos	0 => 4

La instrucción "Leer datos en formato Big-Endian" lee el entero 439_041_101 del área de memoria "#SourceField" y lo escribe en formato Big-Endian en el operando "#DINTVariable". El tipo de datos del parámetro DEST_VARIABLE indica cuántos bytes se leen. En el operando "#TagPos" se guarda el número 4.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Deserialize: Deserializar (Página 4355)

Serialize: Serializar (Página 4358)

Direccionar áreas de una variable con accesos Slice (Página 2301)

WRITE_BIG: Escribir datos en formato Big-Endian

Descripción

La instrucción "Escribir datos en formato Big-Endian" permite escribir los datos de una única variable de la orden de bytes Big-Endian en un área de memoria. En el formato Big-Endian, primero se guarda el byte con los bits más significativos, es decir, en la dirección de memoria más baja. El área de memoria en la que se escribe debe ser del tipo de datos ARRAY of BYTE.

El operando del parámetro SRC_VARIABLE debe ser un tipo de datos elemental. El operando del parámetro POS determina la posición en la que se inicia la escritura.

Nota

Escribir variable del tipo de datos VARIANT o BOOL

Para escribir una variable a la que apunta un VARIANT se pueden utilizar las instrucciones "Serializar" o "Deserializar".

Para escribir una variable del tipo de datos BOOL se puede utilizar un "acceso slice".

Sintaxis

Para la instrucción "Escribir datos en formato Big-Endian" se utiliza la siguiente sintaxis:

SCL

```
WRITE_BIG(SRC_VARIABLE := <Operando>,
          DEST_ARRAY := <Operando>,
          POS := <Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Escribir datos en formato Big-Endian":

Parámetro	Declaración	Tipo de datos	Descripción
SRC_VARIABLE	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, LDT, TOD, LTOD, DATE, CHAR, WCHAR	Variable cuyos datos se escriben
DEST_ARRAY	InOut	ARRAY of BYTE	Área de memoria en la que se escriben los datos

Parámetro	Declaración	Tipo de datos	Descripción
POS	InOut	DINT	Determina la posición en la que se inicia la escritura. El parámetro POS se calcula basado en cero.
Valor de función		INT	Información de error

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#TagResult := WRITE_BIG(SRC_VARIABLE := #DINTVariable,
                        DEST_ARRAY := #TargetField,
                        POS := #TagPos);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC_VARIABLE	#DINTVariable	439041101 16#1A2B3C4D
DEST_ARRAY	#TargetField	ARRAY[0..10] of BYTE = 16#1A, 16#2B, 16#3C, 16#4D
POS	#TagPos	0 => 4

La instrucción "Escribir datos en formato Big-Endian" escribe el entero 439_041_101 del operando "#DINTVariable" en formato Big-Endian en el área de memoria "#TargetField". El tipo de datos del parámetro SRC_VARIABLE indica cuántos bytes se escriben. En el operando "#TagPos" se guarda el número 4.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Deserialize: Deserializar (Página 4355)

Serialize: Serializar (Página 4358)

Direccionar áreas de una variable con accesos Slice (Página 2301)

Instrucciones VARIANT

VariantGet: Leer valor de una variable VARIANT

Descripción

Con la instrucción "Leer valor de una variable VARIANT" se lee el valor de la variable a la que señala VARIANT en el parámetro SRC y se escribe este valor en la variable del parámetro DST.

El parámetro SRC tiene que ser del tipo de datos VARIANT. En el parámetro DST se puede indicar cualquier tipo de datos excepto VARIANT.

El tipo de datos de la variable del parámetro DST debe coincidir con el tipo de datos al que señala VARIANT.

Nota

La variable a la que señala VARIANT en el parámetro SRC, debe ser un tipo de datos PLC.

Para copiar estructuras y ARRAY también se puede utilizar la instrucción "MOVE_BLK_VARIANT: Copiar área". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Leer valor de una variable VARIANT" se utiliza la siguiente sintaxis:

SCL

```
VariantGet(SRC: = <Operando>,
           DST => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer valor de una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Descripción
SRC	Input	VARIANT	Variable que se lee
DST	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres, ARRAY, estructuras	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
VariantGet(SRC := #TagIn_Source,
           DST => "TagOut_Dest");
```

El valor de la variable al que señala VARIANT en el operando "#TagIn_Source" se lee y se escribe en el operando "TagOut_Dest".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MOVE_BLK_VARIANT: Copiar área (Página 4363)

VariantPut: Escribir valor en una variable VARIANT

Descripción

La instrucción "Escribir valor en una variable VARIANT" escribe el valor de la variable del parámetro SRC en la variable del parámetro DST a la que señala VARIANT.

El parámetro DST tiene que ser del tipo de datos VARIANT. En el parámetro SRC se puede indicar cualquier tipo de datos excepto VARIANT.

El tipo de datos de la variable del parámetro SRC debe coincidir con el tipo de datos al que señala VARIANT.

Nota

La variable a la que señala VARIANT en el parámetro DST, debe ser un tipo de datos PLC.

Sintaxis

Para la instrucción "Escribir valor en una variable VARIANT" se utiliza la siguiente sintaxis:

SCL

```
VariantPut(SRC: = <Operando>,
           DST := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir valor en una variable VARIANT":

Parámetro	Declaración	Tipo de datos	Descripción
SRC	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres, ARRAY	Variable que se lee
DST	Input	VARIANT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
VariantPut(SRC := "TagIn_Source",
           DST => "#TagIn_Dest");
```

El valor del operando "TagIn_Source" se escribe en la variable a la que señala VARIANT en el operando "#TagIn_Dest".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

VARIANT (Página 2364)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Principios básicos de la palabra de estado (Página 2559)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

CountOfElements: Consultar número de elementos ARRAY

Descripción

Con la instrucción "Consultar número de elementos ARRAY" se consulta cuantos elementos de ARRAY tiene una variable a la que señala VARIANT.

Si se trata de un ARRAY unidimensional, como resultado se devuelve la diferencia entre los valores límite superior e inferior + 1. Si se trata de un ARRAY multidimensional, como resultado se devuelve el producto de todas las dimensiones.

El <operando> debe ser del tipo de datos VARIANT.

El resultado es "0" si la variable VARIANT no es ARRAY.

Si VARIANT apunta a un ARRAY of BOOL, se cuentan también los elementos de relleno. (P. ej. en el caso de un ARRAY[0..1] of BOOL se devuelve 8.)

Sintaxis

Para la instrucción "Consultar número de elementos ARRAY" se utiliza la siguiente sintaxis:

SCL

```
CountOfElements(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar número de elementos ARRAY":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	VARIANT	Variable que se consulta
Valor de función		UDINT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF IS_ARRAY(#Tag_VARIANTToArray) THEN
"Tag_Result" := CountOfElements(#Tag_VARIANT);
END_IF;
```

Si la variable a la que señala VARIANT es ARRAY, se devuelve el número de elementos ARRAY.

Consulte también

Formato de Array (límites de 16 bits) (Página 2354)
Formato de Array (límites de 32 bits) (Página 2355)
Vista general de los tipos de datos válidos (Página 2321)
Principios básicos del mecanismo EN/ENO (Página 2539)
Cambiar formatos de visualización en el estado del programa (Página 3100)
VARIANT (Página 2364)
Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)
Principios básicos de la palabra de estado (Página 2559)
Entrada de instrucciones SCL (Página 2804)
Edición de instrucciones SCL (Página 2823)

Otras

BLKMOV: Copiar área

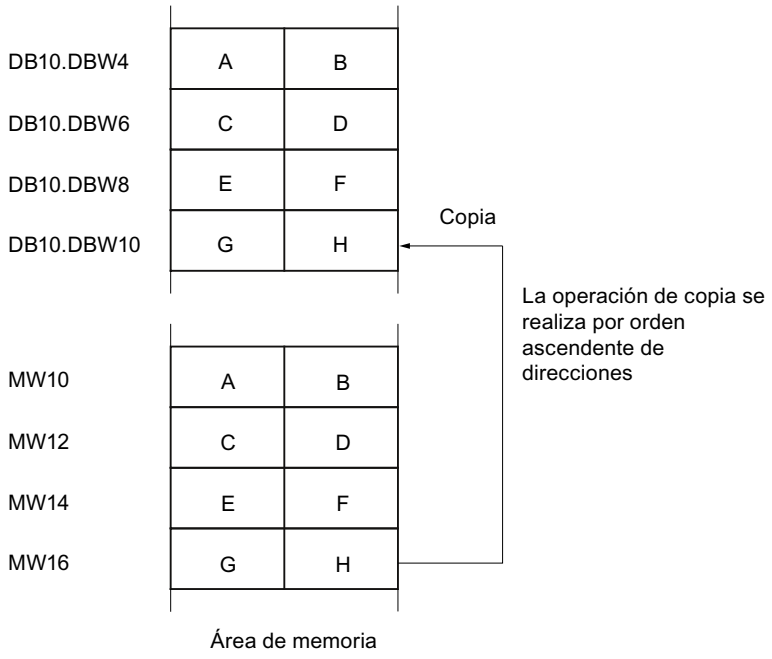
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que durante la ejecución de la instrucción "Copiar área" no se modifican los datos de origen pues, de lo contrario, no se podría garantizar la coherencia de los datos de destino.

Posibilidad de interrupción

No hay limitaciones en cuanto a la profundidad de anidamiento.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

SCL

```
BLKMOV(SRCBLK := <Operando>,
       DSTBLK => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Descripción
SRCBLK	Input	VARIANT	Indicación del área de memoria que se copia (área de origen).
DSTBLK	Output	VARIANT	Indicación del área de memoria en la que se copia (área de destino).
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8092	El área de origen o destino está solo en la memoria de carga.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK , no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

SCL

```
UBLKMOV(SRCBLK := <Operando>,
        DSTBLK => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Descripción
SRCBLK	Input	VARIANT	Indicación del área de memoria que se copia (área de origen).
DSTBLK	Output	ANY	Indicación del área de memoria en la que se copia (área de destino).
Valor de función (RET_VAL)		VARIANT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

FILL: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones.

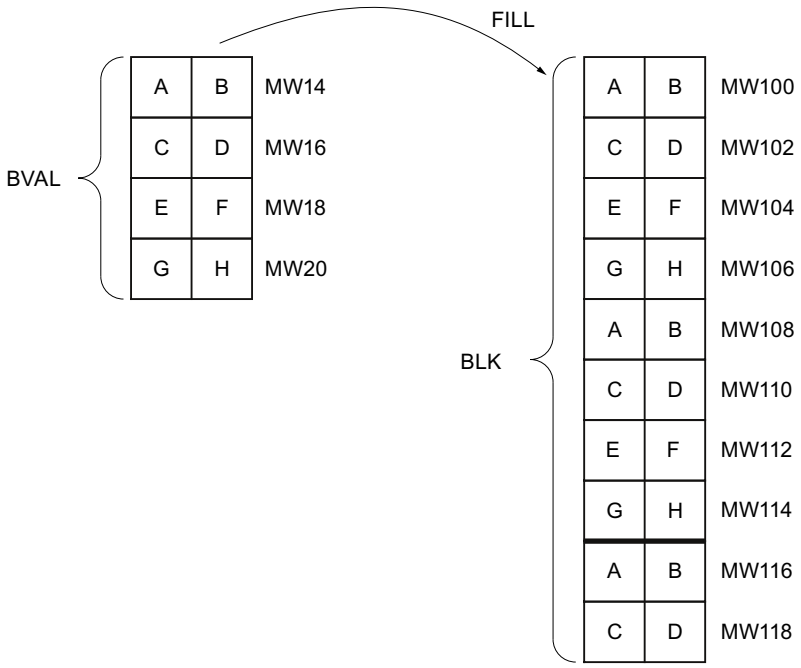
Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Para los bloques con el atributo "Acceso optimizado al bloque" puede utilizarse la instrucción "FILL_BLK: Rellenar área".

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: el contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas

- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

La instrucción "Rellenar área" no permite escribir las siguientes áreas de memoria:

- Instrucciones, bloques de datos de sistema
- Contadores
- Temporizadores
- Áreas de memoria de la periferia.

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

SCL

```
FILL(BVAL := <Operando>,
     BLK => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Descripción
BVAL	Input	VARIANT	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
BLK	Output	VARIANT	Indicación del área de memoria que se rellena con el contenido del área de origen.
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro BVAL

Al indicar una estructura a modo de parámetro de entrada, tenga en cuenta que la longitud de una estructura se adapta siempre a un número entero de bytes. Si se declara una estructura con un número impar de bytes, esta estructura necesita un byte adicional de espacio en memoria.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" permite programar conversiones explícitas. Los tipos de datos que se han de convertir se especifican en un cuadro de diálogo que se abre automáticamente al insertar la instrucción. Durante la ejecución, la instrucción lee el valor de origen y lo convierte al valor de destino indicado.

Encontrará información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

Nota

Para una CPU S7-1500 rige lo siguiente: existe la posibilidad de seleccionar los tipos de datos DWORD y LWORD si se ha seleccionado el tipo de datos IN REAL o LREAL.

Sintaxis

Para la instrucción "Convertir valor" se utiliza la siguiente sintaxis:

SCL

```
<Valor de destino> := <función de conversión> (<valor de origen>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
<Valor de origen>	Input, constante	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD, DTL, secuencias de caracteres	Secuencias de bits, enteros, números en coma flotante, TIME, LTIME, DATE, TOD, LTOD, DTL, secuencias de caracteres	Valor que se convierte.
<Función de conversión>	-	-	-	Función que especifica los tipos de datos que se han de convertir.
<Valor de destino>	Output	Secuencias de bits, enteros, números en coma flotante, TIME, DATE, TOD, DTL, secuencias de caracteres	Secuencias de bits, enteros, números en coma flotante, TIME, LTIME, DATE, TOD, LTOD, DTL, secuencias de caracteres	Resultado de la conversión

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_INT" := REAL_TO_INT("Tag_REAL");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Tipo de datos	Valor
Tag_REAL	REAL	20.56
Tag_INT	INT	21

En la conversión, el valor del operando "Tag_REAL" se redondea al siguiente entero y se deposita en el operando "Tag_INT".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Conversión de tipos de datos (Página 2373)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor de la entrada IN al siguiente número entero. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte a un número entero o a un número en coma flotante. Si el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par.

Sintaxis

Para la instrucción "Redondear número" se utiliza la siguiente sintaxis:

```
SCL
ROUND(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada que se debe redondear.
Valor de función		Enteros, números en coma flotante	Resultado del redondeo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := ROUND("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	1.50000000	-1.50000000
Tag_Result	2	-2

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Conversión de tipos de datos (Página 2373)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor al siguiente número entero superior. La instrucción interpreta el valor de entrada como número en coma flotante y lo convierte al número entero superior más próximo. El valor de la función puede ser mayor o igual al valor de entrada.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero superior" se utiliza la siguiente sintaxis:

```
SCL
CEIL(<expresión>)
CEIL_<tipo de datos>(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<expresión>	Input	Números en coma flotante	Valor de entrada
_<tipo de datos>		Enteros, números en coma flotante valor predeterminado: DINT	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Enteros, números en coma flotante	Valor de entrada redondeado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := CEIL("Tag_Value");
"Tag_Result2" := CEIL_REAL("Tag_Value);
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	0.5	-0.5
Tag_Result1	1	0
Tag_Result2	1.0	0.0

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor al siguiente número entero inferior. La instrucción interpreta el valor de entrada como número en coma flotante y lo convierte al número entero inferior más próximo. El valor de función puede ser menor o igual al valor de entrada.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero inferior" se utiliza la siguiente sintaxis:

```
SCL
FLOOR(<expresión>)
FLOOR_<tipo de datos>(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<expresión>	Input	Números en coma flotante	Valor de entrada
_<tipo de datos>		Enteros, números en coma flotante valor predeterminado: DINT	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Enteros, números en coma flotante	Valor de entrada redondeado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := FLOOR("Tag_Value");
"Tag_Result2" := FLOOR_REAL("Tag_Value");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	0.5	-0.5
Tag_Result1	0	-1
Tag_Result2	0.0	-1.0

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un valor entero no redondeado a partir del valor de entrada. La instrucción selecciona solo la parte entera del valor de entrada y la devuelve sin decimales como valor de función.

Sintaxis

Para la instrucción "Truncar a entero" se utiliza la siguiente sintaxis:

```
SCL
TRUNC(<Expresión>)
TRUNC_<tipo de datos>(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<expresión>	Input	Números en coma flotante	Valor de entrada
_<tipo de datos>		Enteros, números en coma flotante Valor predeterminado: DINT	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Enteros, números en coma flotante	Parte entera del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := TRUNC("Tag_Value1");
"Tag_Result2" := TRUNC("Tag_Value2"+"Tag_Value3");
"Tag_Result3" := TRUNC_SINT("Tag_Value4");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value1	-1.5
Tag_Result1	-1
Tag_Value2	2.1
Tag_Value3	3.2
Tag_Result2	5
Tag_Result3	2
Tag_Value4	2.4

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

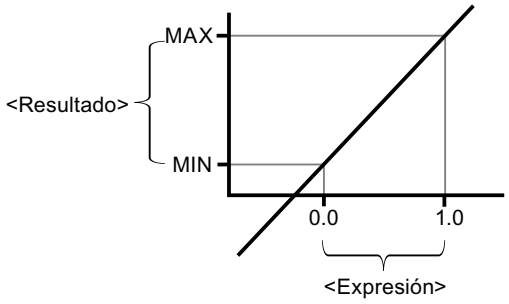
Edición de instrucciones SCL (Página 2823)

SCALE_X: Escalar

Descripción

La instrucción "Escalar" permite escalar un número en coma flotante mapeándolo en un determinado rango de valores. El rango de valores se especifica con los parámetros MIN y MAX. El resultado de la escala es un entero.

La figura siguiente muestra un ejemplo de cómo pueden escalarse los valores:



La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [VALUE * (MAX – MIN)] + MIN

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

SCL

```
SCALE_X(MIN := <Operando>,
        VALUE := <Operando>,
        MAX := <Operando>)
SCALE_X_<Tipo de datos>(MIN := <Operando>,
                        VALUE := <Operando>,
                        MAX := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
MIN	Input	Enteros, números en coma flotante	Límite inferior del rango de valores
VALUE	Input	Números en coma flotante	Valor que se escala. Si se indica una constante, esta debe declararse.
MAX	Input	Enteros, números en coma flotante	Límite superior del rango de valores
_<tipo de datos>		Enteros, números en coma flotante valor predeterminado: INT	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Enteros, números en coma flotante	Resultado de la escala

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := SCALE_X(MIN := "Tag_Value1",
                         VALUE := "Tag_Real",
                         MAX := "Tag_Value2");
"Tag_Result2" := SCALE_X_REAL(MIN := "Tag_Value1",
                              VALUE := "Tag_Real",
                              MAX := "Tag_Value2");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Real	0.5
Tag_Value1	10
Tag_Value2	30
Tag_Result1	20
Tag_Result2	20.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

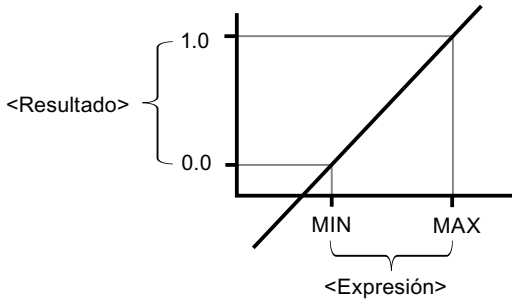
Declarar constantes globales (Página 2584)

NORM_X: Normalizar

Descripción

La instrucción "Normalizar" normaliza el valor de la variable de la entrada VALUE representándolo en una escala lineal. Los parámetros MIN y MAX sirven para definir los límites de un rango de valores que se refleja en la escala. En función de la posición del valor que se debe normalizar en este rango de valores, se calcula el resultado y se deposita como número en coma flotante en la salida OUT. Si el valor que se debe normalizar es igual al valor de la entrada MIN, la instrucción devuelve el valor "0.0" como resultado. Si el valor que se debe normalizar es igual al valor de la entrada MAX, la instrucción devuelve el valor "1.0" como resultado.

La figura siguiente muestra un ejemplo de cómo pueden normalizarse los valores:



La instrucción "Normalizar" utiliza la siguiente ecuación:

OUT = (VALUE – MIN) / (MAX – MIN)

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Sintaxis

Para la instrucción "Normalizar" se utiliza la siguiente sintaxis:

SCL

```
NORM_X(MIN := <operando>,
       VALUE := <Operando>,
       MAX := <Operando>)
NORM_X_<Tipo de datos>(MIN := <Operando>,
                  VALUE := <Operando>,
                  MAX := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
MIN [1)]	Input	Enteros, números en coma flotante	Límite inferior del rango de valores
VALUE [1)]	Input	Enteros, números en coma flotante	Valor que se normaliza.
MAX [1)]	Input	Enteros, números en coma flotante	Límite superior del rango de valores

Parámetro	Declaración	Tipo de datos	Descripción
_<tipo de datos>		Números en coma flotante valor predeterminado: REAL	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Números en coma flotante	Resultado de la normalización
[1] Si se utilizan constantes en estos tres parámetros, basta con declarar una de ellas.			

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := NORM_X(MIN := "Tag_Value1",
                        VALUE := "Tag_InputValue",
                        MAX := "Tag_Value2");
"Tag_Result2" := NORM_X_LREAL(MIN := "Tag_Value1",
                              VALUE := "Tag_InputValue",
                              MAX := "Tag_Value2");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InputValue	20
Tag_Value1	10
Tag_Value2	30
Tag_Result1	0.5
Tag_Result2	0.5

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Declarar constantes globales (Página 2584)

Instrucciones VARIANT

VARIANT_TO_DB_ANY: Convertir VARIANT en DB_ANY

Descripción

La instrucción "Convertir VARIANT en DB_ANY" permite leer el número del bloque de datos de un VARIANT y convertirlo al tipo de datos DB_ANY.

Requisitos

Si se cumplen los requisitos, se ejecuta la instrucción. Si no se cumplen los requisitos, se devuelve "0" como número del bloque de datos.

La variable de salida...	referencia...	Posibilidades de conversión
VARIANT	... un bloque de datos instancia de un UDT o de un SDT	Es posible convertir la variable de salida a un número del bloque de datos.
VARIANT	... un DB de array	Es posible convertir la variable de salida a un número del bloque de datos.
VARIANT	... un objeto con un tipo de datos simple	No es posible convertir la variable de salida a un número del bloque de datos, puesto que un bloque de datos nunca consta solamente de un tipo de datos simple.
VARIANT	... una estructura de un bloque de datos	No es posible convertir la variable de salida a un número del bloque de datos, puesto que solo se trata de una parte del bloque de datos.

Sintaxis

Para la instrucción "Convertir VARIANT en DB_ANY" se utiliza la siguiente sintaxis:

SCL

```
VARIANT_TO_DB_ANY(IN := <Operando>,
                  ERR => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir VARIANT en DB_ANY":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	VARIANT	Variable que se lee
ERR	Output	INT	Información de error
Valor de función		DB_ANY	Resultado: número de DB

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR

La tabla siguiente muestra el significado de los valores del parámetro ERR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
252C	El tipo de datos VARIANT devuelve el valor "0"
8130	El número del bloque de datos es "0"
8131	El bloque de datos no existe
8132	El bloque de datos es demasiado corto
8134	El bloque de datos está protegido contra escritura
8150	El tipo de datos VARIANT devuelve el valor "0"
8154	El tipo de datos no está declarado correctamente
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := VARIANT_TO_DB_ANY(IN := "Tag_In",
                                  ERR := "Tag_Err");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_In	El operando "Tag_In" es del tipo de datos VARIANT. Se lee el valor.
<Valor de función>	Tag_Result	Se devuelve el número del bloque

El número del bloque se lee en el operando "Tag_In" y se devuelve en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

DB_ANY_TO_VARIANT: Convertir DB_ANY en VARIANT

Descripción

La instrucción "Convertir DB_ANY en VARIANT" permite leer el número del bloque de datos de un DB_ANY y convertirlo al tipo de datos VARIANT.

Requisitos

Si se cumplen los requisitos, se ejecuta la instrucción. Si no se cumplen los requisitos o el bloque de datos no existe, se devuelve el valor "0". Los demás accesos a este valor "0" fallan.

La variable de salida...	referencia...	Posibilidades de conversión
DB_ANY	... un bloque de datos instancia de un UDT o de un SDT	Es posible convertir la variable de salida a un tipo de datos VARIANT.
DB_ANY	... un bloque de datos de array	Es posible convertir la variable de salida a un tipo de datos VARIANT.
DB_ANY	... un bloque de datos instancia de un bloque de función o un bloque de datos global	No es posible convertir la variable de salida a un tipo de datos VARIANT.

Sintaxis

Para la instrucción "Convertir DB_ANY en VARIANT" se utiliza la siguiente sintaxis:

SCL

```
DB_ANY_TO_VARIANT(IN := <Operando>,
                  ERR => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir DB_ANY en VARIANT":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	DB_ANY	Variable que se lee
ERR	Output	INT	Información de error
Valor de función		VARIANT	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR

La tabla siguiente muestra el significado de los valores del parámetro ERR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8130	El número del bloque de datos es "0"
8131	El bloque de datos no existe
8132	El bloque de datos es demasiado corto
8134	El bloque de datos está protegido contra escritura
8154	El tipo de datos no está declarado correctamente
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := DB_ANY_TO_VARIANT(IN := "Tag_In",
                                  ERR := "Tag_Err");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_In	El operando "Tag_In" es del tipo de datos DB_ANY. Se lee el valor.
<Valor de función>	Tag_Result	El número del bloque se convierte al tipo de datos VARIANT y se devuelve.

El número del bloque se lee en el operando "Tag_In" y se devuelve en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Otras

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

SCL

```
SCALE(IN := <Expresión>,
      HI_LIM := <Operando>,
      LO_LIM := <Operando>,
      BIPOLAR := <Operando>,
      OUT => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	INT	Valor de entrada que se escala.
HI_LIM	Input	REAL	Valor límite superior
LO_LIM	Input	REAL	Valor límite inferior
BIPOLAR	Input	BOOL	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	Resultado de la instrucción
Valor de función (RET_VAL)		WORD	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que 27 648 o menor que 0 (unipolar) o -27 648 (bipolar).
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
"Tag_ErrorCode" := SCALE(IN := "Tag_InputValue",
                         HI_LIM := "Tag_HighLimit"
                         LO_LIM := "Tag_LowLimit"
                         BIPOLAR := "Tag_Bipolar",
                         OUT => "Tag_Result");
```

La información de error se devuelve en el operando "Tag_ErrorCode" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" permite desescalar el número en coma flotante del parámetro IN en unidades físicas comprendidas entre un límite inferior y uno superior y convertirlo en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Sintaxis

Para la instrucción "Desescalar" se utiliza la siguiente sintaxis:

SCL

```
UNSCALE(IN := <Expresión>,
        HI_LIM := <Operando>,
        LO_LIM := <Operando>,
        BIPOLAR := <Operando>,
        OUT => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	REAL	Valor de entrada que se desescala en un valor entero.
HI_LIM	Input	REAL	Valor límite superior
LO_LIM	Input	REAL	Valor límite inferior
BIPOLAR	Input	BOOL	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	INT	Resultado de la instrucción
Valor de función (RET_VAL)		WORD	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).

Código de error (W#16#....)	Explicación
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_ErrorCode" := UNSCALE(IN := "Tag_InputValue",
                           HI_LIM := "Tag_HighLimit"
                           LO_LIM := "Tag_LowLimit"
                           BIPOLAR := "Tag_Bipolar",
                           OUT => "Tag_Result");
```

La información de error se devuelve en el operando "Tag_ErrorCode" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	22
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

Control del programa

IF: Ejecución condicional

Descripción

La instrucción "Ejecución condicional" permite ramificar el flujo del programa en función de una condición. La condición es una expresión con un valor booleano (TRUE o FALSE). Como condición se pueden indicar expresiones lógicas o expresiones de comparación.

Al ejecutar la instrucción, se evalúan las expresiones indicadas. Si el valor de una expresión es TRUE, la condición se considera cumplida; en el caso de FALSE se considera que la condición no se cumple.

Sintaxis

Dependiendo del tipo de ramificación se pueden programar las siguientes formas de la instrucción:

- Ramificación con IF:

SCL

```
IF <Condición> THEN <Instrucciones>
END_IF;
```

Si se cumple la condición, se ejecutan las instrucciones que están programadas después de THEN. Si la condición no se cumple, el procesamiento del programa continúa con la siguiente instrucción después de END_IF.

- Ramificación con IF y ELSE:

SCL

```
IF <Condición> THEN <Instrucciones1>
ELSE <Instrucciones0>
END_IF;
```

Si se cumple la condición, se ejecutan las instrucciones que están programadas después de THEN. Si no se cumple la condición, se ejecutan las instrucciones que están programadas después de ELSE. Después, el procesamiento del programa continúa con la siguiente instrucción después de END_IF.

- Ramificación con IF, ELSIF y ELSE:

SCL

```
IF <Condición1> THEN <Instrucciones1>
ELSIF <Condición2> THEN <Instrucción2>
ELSE <Instrucciones0>
END_IF;
```

Si se cumple la primera condición (<condición1>), se ejecutan las instrucciones (<instrucciones1>) después de THEN. Después de ejecutar las instrucciones, el procesamiento del programa continúa después de END_IF.

Si no se cumple la primera condición, se comprueba la segunda (<condición2>). Si se cumple la segunda condición (<condición2>), se ejecutan las instrucciones (<instrucciones2>) después de THEN. Después de ejecutar las instrucciones, el procesamiento del programa continúa después de END_IF.

Si no se cumple ninguna condición, se ejecutan las instrucciones (<instrucciones0> después de ELSE y, a continuación, continúa el procesamiento del programa después de END_IF.

En una instrucción IF se pueden anidar tantas combinaciones de ELSIF y THEN como se desee. La programación de una rama ELSE es opcional.

La sintaxis de la instrucción IF se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Condición>	BOOL	Expresión que se evalúa.
<Instrucciones>	-	Instrucciones que se ejecutan si se cumple la condición. Una excepción son las instrucciones que están programadas después de ELSE. Estas se ejecutan si no se cumple ninguna condición dentro del bucle.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF "Tag_1" = 1
THEN "Tag_Value" := 10;
ELSIF "Tag_2" = 1
THEN "Tag_Value" := 20;
ELSIF "Tag_3" = 1
THEN "Tag_Value" := 30;
ELSE "Tag_Value" := 0;
END_IF;
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_1	1	0	0	0
Tag_2	0	1	0	0
Tag_3	0	0	1	0
Tag_Value	10	20	30	0

Consulte también

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

CASE: Estructura selectiva múltiple

Descripción

La instrucción "Estructura selectiva múltiple" permite procesar una de varias secuencias de instrucciones en función del valor de una expresión numérica.

El valor de la expresión ha de ser un entero. Al ejecutar la instrucción se compara el valor de la expresión con los valores de varias constantes. Si el valor de la expresión coincide con el de una constante, se ejecutan las instrucciones que están programadas directamente después de esta constante. Las constantes pueden adoptar los siguientes valores:

- un número entero (p. ej., 5)
- un rango de números enteros (p. ej. 15 .. 20)
- una enumeración de números enteros y rangos (p. ej., 10, 11, 15 .. 20)

Sintaxis

Para la instrucción "Estructura selectiva múltiple" se utiliza la siguiente sintaxis:

SCL

```
CASE <Expresión> OF
<Constante1>: <Instrucciones1>
<Constante2>: <Instrucciones2>
<ConstanteX>: <InstruccionesX>; // X >= 3
ELSE <Instrucciones0>
END_CASE;
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Expresión>	Enteros	Valor que se compara con los valores de constantes programados.
<Constante>	Enteros	Valores de constantes que forman la condición para ejecutar una secuencia de instrucciones. Las constantes pueden adoptar los siguientes valores: • un número entero (p. ej., 5) • un rango de números enteros (p. ej. 15 .. 20) • una enumeración de números enteros y rangos (p. ej., 10, 11, 15 .. 20)
<Instrucción>	-	Instrucciones cualesquiera que se ejecutan si el valor de la expresión coincide con el valor de una constante. Una excepción son las instrucciones que están programadas después de ELSE. Estas se ejecutan cuando los valores no coinciden.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Si el valor de la expresión coincide con el valor de la primera constante (<Constante1>), se ejecutan las instrucciones (<instrucciones1>) que están programadas directamente después de la primera constante. Luego, el procesamiento del programa continúa después de END_CASE.

Si el valor de la expresión no coindice con el valor de la primera constante (<Constante1), este se compara con el valor de la constante que está programada a continuación. De esta manera la instrucción CASE se ejecuta hasta que coincidan los valores. Si el valor de la expresión no se corresponde con ninguno de los valores programados de las constantes, se ejecutan las instrucciones (<instrucciones0>) que están programadas después de ELSE. ELSE es una parte opcional de la sintaxis y se puede omitir.

La instrucción CASE también se puede anidar sustituyendo un bloque de instrucciones por CASE. END_CASE cierra la instrucción CASE.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
CASE "Tag_Value" OF
     0 :
     "Tag_1" := 1;
     1,3,5 :
     "Tag_2" :=1;
     6..10 :
     "Tag_3" := 1;
     16,17,20..25 :
     "Tag_4" := 1;
ELSE "Tag_5" := 1;
END_CASE;
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores				
Tag_Value	0	1, 3 , 5	6, 7, 8, 9, 10	16,17, 20, 21, 22, 23, 24, 25	2
Tag_1	1	-	-	-	-
Tag_2	-	1	-	-	-
Tag_3	-	-	1	-	-
Tag_4	-	-	-	1	-
Tag_5	-	-	-	-	1
1: El operando adopta el estado lógico "1". -: El estado lógico del operando permanece sin cambios.					

Consulte también

CONTINUE: Verificar condición de bucle (Página 4445)

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

EXIT: Abandonar el bucle inmediatamente (Página 4446)

FOR: Ejecutar en bucle contador

Descripción

La instrucción "Ejecutar en bucle contador" permite procesar un bucle mientras la variable de contaje se encuentre dentro del rango de valores indicado.

Los bucles también se pueden anidar. Dentro de un bucle se pueden programar otros bucles con otras variables de contaje.

La ejecución actual de un bucle se puede finalizar con la instrucción "Verificar condición de bucle" (CONTINUE). La instrucción "Abandonar el bucle inmediatamente" (EXIT) permite finalizar el procesamiento de todo el bucle. Encontrará más información al respecto en "Consulte también".

Valores límite para instrucciones FOR

Para programar instrucciones FOR "seguras" que no se ejecuten de manera infinita, tenga en cuenta las siguientes reglas y los siguientes valores límite:

Regla

```
FOR ii := principio TO fin BY paso DO
```

Si...	... entonces	Observación
principio < fin	fin < (PMAX - paso)	La variable de contaje ii cuenta en sentido ascendente
principio > fin Y paso < O	fin > (NMAX - paso)	La variable de contaje ii cuenta en sentido descendente

Valores límite

Para los tipos de datos posibles son válidos diferentes valores límite:

Tipo de datos	PMAX	NMAX
ii del tipo SINT	127	-128
ii del tipo INT	32767	-32768
ii del tipo DINT	2147483647	-2147483648
ii del tipo LINT	9223372036854775807	-9223372036854775808

Sintaxis

Para la instrucción "Ejecutar en bucle contador" se utiliza la siguiente sintaxis:

SCL

```
FOR <Variable de contaje> := <Valor inicial> TO <Valor final> BY
<Incremento> DO <Instrucciones>
END_FOR;
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ejecutar en bucle contador":

Parámetro	Tipo de datos		Descripción
	S7-1200	S7-1500	
<Variable de contaje>	SINT, INT, DINT	SINT, INT, DINT, LINT	Operando, cuyo valor se evalúa durante el procesamiento del bucle. El tipo de datos de la variable de contaje determina el tipo de datos de los parámetros restantes.
<Valor inicial>	SINT, INT, DINT	SINT, INT, DINT, LINT	Expresión, cuyo valor se asigna a la variable de contaje al comienzo del procesamiento del bucle.
<Valor final>	SINT, INT, DINT	SINT, INT, DINT, LINT	Expresión, cuyo valor determina la última ejecución del bucle. Antes de cada ejecución del bucle se comprueba qué valor tiene la variable de contaje: • Valor final no alcanzado: Se ejecutarán las instrucciones según DO • Valor final alcanzado: El bucle FOR será ejecutado una última vez • Valor final sobrepasado: El bucle FOR será finalizado No se puede modificar el valor final durante la ejecución de la instrucción.
<Incremento>	SINT, INT, DINT	SINT, INT, DINT, LINT	Expresión, en cuyo valor aumenta (incremento positivo) o disminuye (incremento negativo) la variable de contaje después de cada ejecución del bucle. La indicación del incremento es opcional. Si no se ha indicado ningún incremento, el valor de la variable de contaje aumenta en 1 después de cada ejecución del bucle. No se puede modificar el incremento durante la ejecución de la instrucción.
<Instrucciones>	-	-	Instrucciones que se ejecutan durante cada ejecución del bucle mientras el valor de la variable de control se encuentre en el rango de valores. El rango de valores se especifica mediante los valores inicial y final.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
FOR i := 2 TO 8 BY 2
    DO "a_array[i] := "Tag_Value"*"b_array[i]";
END_FOR;
```

El operando "Tag_Value" se multiplica por los elementos (2, 4, 6, 8) de las variables ARRAY "b_array". El resultado se lee en los elementos (2, 4, 6, 8) de las variables ARRAY "a_array".

Consulte también

CONTINUE: Verificar condición de bucle (Página 4445)

EXIT: Abandonar el bucle inmediatamente (Página 4446)

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

WHILE: Ejecutar cuando se cumpla la condición

Descripción

La instrucción "Ejecutar cuando se cumpla la condición" permite procesar un bucle mientras se cumpla una condición de ejecución. La condición es una expresión con un valor booleano (TRUE o FALSE). Como condición se pueden indicar expresiones lógicas o expresiones de comparación.

Al ejecutar la instrucción, se evalúan las expresiones indicadas. Si el valor de una expresión es TRUE, la condición se considera cumplida; en el caso de FALSE se considera que la condición no se cumple.

Los bucles también se pueden anidar. Dentro de un bucle se pueden programar otros bucles con otras variables de contaje.

La ejecución actual de un bucle se puede finalizar con la instrucción "Verificar condición de bucle" (CONTINUE). La instrucción "Abandonar el bucle inmediatamente" (EXIT) permite finalizar el procesamiento de todo el bucle. Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Ejecutar cuando se cumpla la condición" se utiliza la siguiente sintaxis:

SCL

```
WHILE <Condición> DO <Instrucciones>
END_WHILE;
```

La sintaxis de la instrucción WHILE se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Condición>	BOOL	Expresión que se evalúa antes de cada ejecución del bucle.
<Instrucciones>	-	Instrucciones que se ejecutan si se cumple la condición. Si no se cumple la condición, el procesamiento del programa continúa después de END_WHILE.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
WHILE
      "Tag_Value1" <> "Tag_Value2"
      DO "Tag_Result"
      := "Tag_Input";
END_WHILE;
```

Mientras difieran los valores de los operandos "Tag_Value1" y "Tag_Value2", al operando "Tag_Result" se le asignará el valor del operando "Tag_Input".

Consulte también

EXIT: Abandonar el bucle inmediatamente (Página 4446)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

CONTINUE: Verificar condición de bucle (Página 4445)

Vista general de los tipos de datos válidos (Página 2321)

REPEAT: Ejecutar cuando no se cumpla la condición

Descripción

La instrucción "Ejecutar cuando no se cumpla la condición" permite procesar un bucle mientras no se cumpla una condición de interrupción. La condición es una expresión con un valor

booleano (TRUE o FALSE). Como condición se pueden indicar expresiones lógicas o expresiones de comparación.

Al ejecutar la instrucción, se evalúan las expresiones indicadas. Si el valor de una expresión es TRUE, la condición se considera cumplida; en el caso de FALSE se considera que la condición no se cumple.

Las instrucciones se ejecutan una vez aunque se cumpla la condición de interrupción.

Los bucles también se pueden anidar. Dentro de un bucle se pueden programar otros bucles con otras variables de contaje.

La ejecución actual de un bucle se puede finalizar con la instrucción "Verificar condición de bucle" (CONTINUE). La instrucción "Abandonar el bucle inmediatamente" (EXIT) permite finalizar el procesamiento de todo el bucle. Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Ejecutar cuando no se cumpla la condición" se utiliza la siguiente sintaxis:

SCL

```
REPEAT <Instrucciones>
UNTIL <Condición> END_REPEAT;
```

La sintaxis de la instrucción REPEAT se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Instrucciones>	-	Instrucciones que se ejecutan mientras la condición programada tenga el valor FALSE. Las instrucciones se ejecutan una vez aunque se cumpla la condición de interrupción.
<Condición>	BOOL	Expresión que se evalúa después de cada ejecución del bucle. Si la expresión tiene el valor FALSE, el bucle se procesa de nuevo. Si la expresión tiene el valor TRUE, el procesamiento del programa continúa después de END_REPEAT.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
REPEAT "Tag_Result"
        := "Tag_Value";
UNTIL "Tag_Error"
END_REPEAT;
```

Mientras el valor del operando "Tag_Error" tenga el estado lógico "0", se le asignará al operando "Tag_Result" el valor del operando "Tag_Value".

Consulte también

CONTINUE: Verificar condición de bucle (Página 4445)

EXIT: Abandonar el bucle inmediatamente (Página 4446)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

CONTINUE: Verificar condición de bucle

Descripción

La instrucción "Verificar condición de bucle" permite finalizar la ejecución actual de un bucle FOR, WHILE o REPEAT.

Después de ejecutar la instrucción, se evalúan de nuevo las condiciones para continuar el bucle. La instrucción tiene efecto sobre el bucle que incluye directamente la instrucción.

Sintaxis

Para la instrucción "Verificar condición de bucle" se utiliza la siguiente sintaxis:

SCL

```
CONTINUE;
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
FOR i
     := 1 TO 15 BY 2 DO
     IF (i < 5) THEN
     CONTINUE;
END_IF;
      "DB10".Test[i] := 1;
END_FOR;
```

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Si se cumple la condición i < 5, no se procesa la siguiente asignación de valor ("DB10".Test[i] := 1). La variable de contaje (i) se incrementa en "2" y se comprueba si su valor actual se encuentra en el rango de valores programados. Si la variable de contaje se encuentra en el rango de valores, se evalúa de nuevo una condición IF.

Si no se cumple la condición i < 5, se procesa la siguiente asignación de valor ("DB10".Test[i] := 1) y se inicia una nueva ejecución del bucle. La variable de control también se aumenta con el incremento "2" y se comprueba.

Consulte también

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

EXIT: Abandonar el bucle inmediatamente (Página 4446)

Vista general de los tipos de datos válidos (Página 2321)

EXIT: Abandonar el bucle inmediatamente

Descripción

La instrucción "Abandonar el bucle inmediatamente" permite interrumpir en cualquier punto la ejecución de un bucle FOR, WHILE o REPEAT independientemente de las condiciones. El procesamiento del programa continúa después finalizar el bucle (END_FOR, END_WHILE, END_REPEAT).

La instrucción tiene efecto sobre el bucle que incluye directamente la instrucción.

Sintaxis

Para la instrucción "Abandonar el bucle inmediatamente" se utiliza la siguiente sintaxis:

SCL

```
EXIT;
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
FOR i := 15 TO 1 BY -2 DO
IF (i < 5)
THEN EXIT;
END_IF;
"DB10".Test[i] := 1;
END_FOR;
```

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Si se cumple la condición i < 5, se interrumpe el procesamiento del bucle. El procesamiento continúa después de END_FOR.

Si no se cumple la condición i < 5, se procesa la siguiente asignación de valor ("DB10".Test[i] := 1) y se inicia una nueva ejecución del bucle. La variable de contaje (i) se decrementa en "-2" y se comprueba si su valor actual se encuentra en el rango de valores programados. Si la variable de control (i) se encuentra en el rango de valores, la condición IF se evalúa de nuevo.

Consulte también

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

CONTINUE: Verificar condición de bucle (Página 4445)

Vista general de los tipos de datos válidos (Página 2321)

GOTO: Saltar

Descripción

La instrucción "Saltar" permite continuar el procesamiento del programa en un punto marcado con una etiqueta.

La etiqueta y la instrucción "Saltar" deben estar en el mismo bloque. La denominación de la etiqueta debe ser unívoca en un bloque. A toda etiqueta pueden acceder distintas instrucciones de salto.

No se permite saltar a un bucle desde "fuera", pero sí es posible saltar desde un bucle hacia "fuera".

Sintaxis

Para la instrucción "Saltar" se utiliza la siguiente sintaxis:

SCL

```
GOTO <Etiqueta>
...
<Etiqueta>: <Instrucciones>
```

La sintaxis de la instrucción GOTO se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<etiqueta>	-	Etiqueta a la que se salta
<Instrucciones>	-	Instrucciones que se ejecutan después del salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
CASE "Tag_Value" OF
1 : GOTO MyLABEL1;
2 : GOTO MyLABEL2;
3 : GOTO MyLABEL3;
ELSE GOTO MyLABEL4;
END_CASE;
MyLABEL1: "Tag_1" := 1;
MyLABEL2: "Tag_2" := 1;
MyLABEL3: "Tag_3" := 1;
MyLABEL4: "Tag_4" := 1;
```

Dependiendo del valor del operando "Tag_Value", el procesamiento del programa se continúa en el punto marcado por la etiqueta correspondiente. Por ejemplo, si el operando "Tag_Value" tiene el valor "2", el procesamiento del programa continúa a partir de la etiqueta "MyLABEL2". En este caso, la línea del programa marcada con la etiqueta "MyLABEL1" se omite.

Consulte también

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

RETURN: Abandonar el bloque

Descripción

La instrucción "Abandonar el bloque" permite finalizar el procesamiento del programa en el bloque que se está ejecutando y continuar en el bloque invocante.

En el fin del bloque se puede omitir la instrucción.

Sintaxis

Para la instrucción "Abandonar el bloque" se utiliza la siguiente sintaxis:

SCL

```
RETURN;
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
IF "Tag_Error" <>0 THEN RETURN;
END_IF;
```

Si el estado lógico del operando "Tag_Error" no es igual a cero, el procesamiento del programa se finaliza en el bloque que se está procesando.

Consulte también

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

(*...*): Insertar sección de comentario

Descripción

La instrucción "Insertar sección de comentario" permite insertar una sección de comentario. El texto encerrado entre paréntesis "(*...*)" se trata como un comentario.

Sintaxis

Para la instrucción "Insertar sección de comentario" se utiliza la siguiente sintaxis:

```
SCL
(*...*)
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
(*Esta es una sección de comentario.*)
```

Consulte también

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Control del tiempo de ejecución

ENDIS_PW: Limitar y habilitar legitimación de la contraseña

Descripción

La instrucción "Limitar y habilitar legitimación de la contraseña" permite definir si las contraseñas configuradas para la CPU están legitimadas o no. Con ello, se pueden impedir conexiones legitimadas, aun conociendo la contraseña correcta.

Si se llama la instrucción y el parámetro REQ tiene el estado lógico "0", en los parámetros de salida se muestra el estado ajustado en ese momento. Si se han realizado cambios en los parámetros de entrada, estos no se transfieren a los parámetros de salida.

Si se llama la instrucción y el parámetro REQ tiene el estado lógico "1", se aplica el estado lógico de los parámetros de entrada (F_PWD, FULL_PWD, R_PWD, HMI_PWD). FALSE significa que la legitimación por contraseña no está permitida y TRUE significa que la contraseña puede utilizarse.

Es posible permitir o prohibir individualmente el bloqueo o la habilitación de las contraseñas. Por ejemplo, pueden prohibirse todas las contraseñas excepto la contraseña de seguridad positiva. Esto permite limitar las posibilidades de acceso a un reducido grupo de usuarios. Los parámetros de salida (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) siempre muestran el estado actual del uso de contraseñas, independientemente del parámetro REQ.

Las contraseñas no configuradas deben tener el estado lógico TRUE en la entrada y devolver el estado lógico TRUE en la salida. La contraseña de seguridad positiva (fail-safe) solo se puede parametrizar para una CPU F y por eso en una CPU estándar debe interconectarse siempre con el estado lógico TRUE. Si la instrucción devuelve un error, la llamada permanece sin efecto, es decir, el bloqueo anterior sigue en vigor.

Las contraseñas bloqueadas pueden volver a habilitarse en las condiciones siguientes:

- Se ha restablecido la configuración de fábrica de la CPU.
- El panel frontal de la CPU S7-1500 soporta un cuadro de diálogo que permite navegar al menú adecuado y volver a habilitar las contraseñas.
- Cuando se llama la instrucción "Limitar y habilitar legitimación de la contraseña", el parámetro de entrada de la contraseña deseada tiene el estado lógico "1".
- Ponga el selector de modo a STOP. La restricción de la legitimación de contraseña se activa de nuevo en cuanto se mueve el selector nuevamente a RUN.
- La inserción de una tarjeta de memoria vacía (tarjeta de transferencia o tarjeta de programa) en una CPU S7-1200.
- La transición de POWER OFF-POWER ON desactiva la protección en la CPU S7-1200. La instrucción "Limitar y habilitar legitimación de la contraseña" debe volverse a llamar en el programa (por ejemplo en el OB de arranque).

Nota

Si la contraseña HMI no está habilitada, la instrucción "Limitar y habilitar legitimación de la contraseña" bloquea el acceso de los sistemas HMI.

Nota

Las conexiones ya existentes y legitimadas conservan sus derechos de acceso y no pueden limitarse con la instrucción "Limitar y habilitar legitimación de la contraseña".

Impedir el bloqueo no intencionado en una CPU S7-1500

Los ajustes pueden realizarse en el panel frontal de la CPU, que guarda el último ajuste en cada caso.

Para impedir un bloqueo no intencionado, en una CPU S7-1500 es posible desactivar la protección moviendo el selector de modo a STOP. La protección se activa automáticamente después de mover el selector de modo a RUN sin que tenga que volver a llamarse la instrucción "Limitar y habilitar legitimación de la contraseña" o deban realizarse otras acciones en el panel frontal.

Impedir el bloqueo no intencionado en una CPU S7-1200

Como la CPU S7-1200 no dispone de selector de modo, la protección se desactiva con POWER OFF-POWER ON. De este modo es posible y recomendable impedir el bloqueo no intencionado con ayuda de determinadas secuencias en el programa.

Para ello, programe un control de tiempo, ya sea mediante un OB de alarma cíclica o un temporizador en el Main OB (OB 1). De este modo, tras una transición de POWER OFF-POWER ON y la consiguiente desactivación de la protección, se puede volver a llamar rápidamente la instrucción "Limitar y habilitar legitimación de la contraseña" en el OB correspondiente (por ejemplo OB 1 u OB 35). Para que la ventana temporal en la que la instrucción no está activa y en la que por tanto no existen limitaciones en la legitimación de la contraseña sea lo más pequeña posible, llame la instrucción en el OB de arranque (OB 100). Este procedimiento ofrece la mayor protección posible contra accesos no autorizados.

Si se ha producido un bloqueo no intencionado, puede omitirse la llamada en el OB de arranque (por ejemplo mediante consulta de un parámetro de entrada) y se dispondrá del tiempo configurado (por ejemplo de 10 segundos a 1 minuto) para establecer una conexión con la CPU antes de que el bloqueo vuelva a activarse.

Si no se ha previsto ningún temporizador en el código del programa y se produce un bloqueo, inserte en la CPU una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna de la CPU. Después deberá cargarse nuevamente el programa de usuario de STEP 7 en la CPU.

Procedimiento en caso de pérdida de contraseña en una CPU S7-1200

Si ha perdido la contraseña de una CPU S7-1200 protegida por contraseña, borre el programa protegido por contraseña con una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna

de la CPU. A continuación, puede transferir un nuevo programa de usuario de STEP 7 Basic a la CPU.

⚠ ADVERTENCIA

Inserción de una tarjeta de transferencia vacía

Cuando se inserta una tarjeta de transferencia en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

El contenido de la tarjeta de transferencia se encuentra en la memoria de carga interna después de su extracción. Compruebe aquí que la tarjeta no contenga ningún programa.

⚠ ADVERTENCIA

Inserción de una tarjeta de programa vacía

Cuando se inserta una tarjeta de programa en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

Asegúrese de que la tarjeta de programa esté vacía. La memoria de carga interna se copiará en la tarjeta de programa vacía. Después de retirar la tarjeta de programa previamente vacía, la memoria de carga interna quedará vacía.

Debe retirarse la tarjeta de transferencia o la tarjeta de programa antes de poner la CPU en RUN.

Efectos del uso de contraseñas en los modos de operación

La siguiente tabla muestra qué efectos tiene el uso de contraseñas en los modos de operación y en las respectivas acciones del usuario mediante la instrucción "Limitar y habilitar legitimación de la contraseña".

Acción	Protección por contraseña mediante la instrucción
Estado básico después de • Selector de modo en STOP • Restablecimiento manual de la memoria (PG, interruptor, cambio de MC (Motion Control)) • Restablecimiento de la configuración de fábrica	No activado (sin limitaciones)
Estado básico tras conexión (POWER ON)	• CPU S7-1200: El bloqueo está desactivado y la instrucción tiene que volverse a ejecutar en el programa (por ejemplo en el OB de arranque). • CPU S7-1500: Activada (si antes de la desconexión (POWER OFF) había un bloqueo activo) La posibilidad de no permitir contraseñas es remanente.
Transición del estado operativo RUN/ARRANQUE/PARADA -> STOP (mediante finalización de la instrucción, un error o comunicación) o STOP -> ARRANQUE/RUN/PARADA	Activado Las contraseñas siguen sin poder utilizarse.

Sintaxis

Para la instrucción "Limitar y habilitar legitimación de la contraseña" se utiliza la siguiente sintaxis:

SCL

```
ENDIS_PW(REQ := <Operando>,
         F_PWD := <Operando>,
         FULL_PWD := <Operando>,
         R_PWD := <Operando>,
         HMI_PWD := <Operando>,
         F_PWD_ON => <Operando>,
         FULL_PWD_ON => <Operando>,
         R_PWD_ON => <Operando>,
         HMI_PWD_ON => <Operando>,
         RET_VAL => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Limitar y habilitar legitimación de la contraseña":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Si el parámetro REQ tiene el estado lógico "0", se consulta el estado lógico ajustado actualmente para las contraseñas.
F_PWD	Input	BOOL	I, Q, M, D, L	Acceso de lectura/escritura incluido failsafe • F_PWD = "0": No permitir contraseña • F_PWD = "1": Permitir contraseña
FULL_PWD	Input	BOOL	I, Q, M, D, L	Acceso de lectura/escritura • FULL_PWD = "0": No permitir contraseña • FULL_PWD = "1": Permitir contraseña
R_PWD	Input	BOOL	I, Q, M, D, L	Acceso de lectura • R_PWD = "0": No permitir contraseña • R_PWD = "1": Permitir contraseña
HMI_PWD	Input	BOOL	I, Q, M, D, L	Acceso a la HMI • HMI_PWD = "0": No permitir contraseña • HMI_PWD = "1": Permitir contraseña
F_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura incluido failsafe • F_PWD_ON = "0": Contraseña no permitida • F_PWD_ON = "1": Contraseña permitida
FULL_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura • FULL_PWD_ON = "0": Contraseña no permitida • FULL_PWD_ON = "1": Contraseña permitida
R_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura • R_PWD_ON = "0": Contraseña no permitida • R_PWD_ON = "1": Contraseña permitida
HMI_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso a la HMI • HMI_PWD_ON = "0": Contraseña no permitida • HMI_PWD_ON = "1": Contraseña permitida
RET_VAL	Output	WORD	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8090	No se soporta la instrucción "Limitar y habilitar legitimación de la contraseña".

Código de error* (W#16#...)	Explicación
80D0	La contraseña de seguridad (fail-safe) no está configurada. En las CPU estándar, el estado lógico tiene que ser TRUE.
80D1	El acceso de lectura/escritura no está configurado
80D2	El acceso de lectura no está configurado
80D3	El acceso HMI no está configurado
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del tiempo de ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar en todos los bloques independientemente de la prioridad.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar hasta 10 veces en un ciclo de programa.

Nota

Si se inicia la instrucción "Reiniciar tiempo de vigilancia del ciclo" dentro de un bucle LOOP más de 30 veces, la CPU pasa al estado operativo STOP debido a un error en tiempo de ejecución.

Sintaxis

Para la instrucción "Reiniciar tiempo de vigilancia del ciclo" se utiliza la siguiente sintaxis:

SCL

```
RE_TRIGR()
```

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos de una transición de RUN a STOP dependen de la configuración de la CPU.

Sintaxis

Para la instrucción "Finalizar programa" se utiliza la siguiente sintaxis:

```
SCL
STP()
```

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

GET_ERROR: Consultar error localmente

Descripción

La instrucción "Consultar error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si el sistema notifica errores mientras se ejecuta el bloque, se muestra información detallada acerca del primer error que ha ocurrido al ejecutar bloque desde la última que se ejecutó la instrucción.

La información relativa al error solo se puede guardar en operandos del tipo de datos de sistema "ErrorStruct". El tipo de datos de sistema "ErrorStruct" especifica la estructura exacta con la que debe guardarse la información acerca del error que ha ocurrido. Con ayuda de otras instrucciones se puede evaluar esta estructura y programar la reacción correspondiente.

Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces emite información sobre el siguiente error ocurrido.

Nota

El <operando> solo se modifica si hay una información de error. Para volver a poner el operando a "0" tras tratar un error existen las siguientes posibilidades:

- Declare el operando en la sección "Temp" de la interfaz de bloque.
- Ponga el operando de nuevo a "0" antes de llamar la instrucción.

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar error localmente" activa el tratamiento local de errores en el bloque. Si en la lógica de un bloque se ha insertado "Consultar error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Sintaxis

Para la instrucción "Consultar error localmente" se utiliza la siguiente sintaxis:

SCL

```
GET_ERROR(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar error localmente":

Parámetro	Tipo de datos	Descripción
<Operando>	ErrorStruct	Información sobre los errores ocurridos

Tipo de datos "ErrorStruct"

La tabla siguiente muestra la estructura del tipo de datos "ErrorStruct":

Componente de estructura	Tipo de datos	Descripción
ERROR_ID	WORD	ID de error
FLAGS	BYTE	Muestra si el error se ha producido durante la llamada de un bloque. 16#01: Error durante la llamada de un bloque 16#00: No hay errores durante la llamada de un bloque

Componente de estructura		Tipo de datos	Descripción
REACTION		BYTE	Reacción predeterminada: 0: ignorar (error de escritura), 1: continuar con el valor sustitutivo "0" (error de lectura), 2: omitir instrucción (error de sistema)
CODE_ADDRESS		CREF	Información sobre la dirección y el tipo del bloque
	BLOCK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error: 1: OB 2: FC 3: FB
	CB_NUMBER	UINT	Número del bloque lógico
	OFFSET	UDINT	Referencia a la memoria interna
MODE		BYTE	Información sobre la dirección de un operando
OPERAND_NUMBER		UINT	Número de operando del comando de la máquina
POINTER_NUMBER_LOCATION		UINT	(A) Puntero interno
SLOT_NUMBER_SCOPE		UINT	(B) Área de almacenamiento en la memoria interna
DATA_ADDRESS		NREF	Información sobre la dirección de un operando
	AREA	BYTE	(C) Área de memoria: L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE I: 16#81 Q: 16#82 M: 16#83 DB: 16#84, 85, 8A, 8B Infracciones de área en una variable editable directamente del tipo de datos DINT: 16#04
	DB_NUMBER	UINT	(D) Número del bloque de datos
	OFFSET	UDINT	(E) Dirección relativa del operando

Componente de estructura "ERROR_ID"

La tabla siguiente muestra los valores que pueden devolverse en el componente de estructura "ERROR_ID":

ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido

ID* (hexadecimal)	ID* (decimal)	Descripción
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
LABEL: #TagOut := #Field[#index] * REAL#40.5;

GET_ERROR(#Error);

IF #Error.REACTION = 1 THEN
   GOTO LABEL;
```

SCL

```
    ;
END_IF;
```

Se ha producido un error al acceder a la variable "#Field[#index]". A pesar del error de lectura y acceso, el operando "#TagOut" devuelve el estado lógico "1" y la multiplicación se realiza con el valor "0.0". Si se produce este error se recomienda programar la instrucción "Consultar error localmente" tras la multiplicación para capturar el error. La información de error suministrada por la instrucción "Consultar error localmente" se evalúa con una comparación. Si el componente de estructura "#Error.REACTION" tiene el valor "1", se trata de un error de lectura/acceso y la ejecución del programa comienza de nuevo en la etiqueta de salto LABEL.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Consultar y eliminar errores en el código del programa (Página 265)

Tratar errores de ejecución del programa (Página 2932)

GET_ERR_ID: Consultar ID de error localmente

Descripción

La instrucción "Consultar ID de error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si al ejecutar el bloque el sistema notifica errores de ejecución desde que se ejecutó la instrucción por última vez, la instrucción emite la ID del primer error ocurrido.

La ID de error solo se puede guardar en operandos del tipo de datos WORD. Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces la instrucción emite la ID de error del siguiente error ocurrido.

Nota

El <operando> solo se modifica si hay una información de error. Para volver a poner el operando a "0" tras tratar un error existen las siguientes posibilidades:

- Declare el operando en la sección "Temp" de la interfaz de bloque.
- Ponga el operando de nuevo a "0" antes de llamar la instrucción.

La salida de la instrucción "Consultar ID de error localmente" se activa únicamente si existe una información de error. Si no se cumple esta condición, el procesamiento ulterior del programa no se ve afectado por la instrucción "Consultar ID de error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar ID de error localmente" activa el tratamiento local de errores dentro del bloque. Si en la lógica de un bloque se ha insertado la instrucción "Consultar ID de error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Sintaxis

Para la instrucción "Consultar ID de error localmente" se utiliza la siguiente sintaxis:

SCL

```
GET_ERR_ID()
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar ID de error localmente":

Parámetro	Tipo de datos	Descripción
Valor de función	WORD	ID de error

ID de error

La tabla siguiente muestra los valores que pueden devolverse:

ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe

ID* (hexadecimal)	ID* (decimal)	Descripción
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
#TagOut := #Field[#index] * REAL#40.5;

#TagID := GET_ERR_ID();

IF #TagID = 16#2522 THEN
   MOVE_BLK(IN := #TagArrayIn[0],
            COUNT := 1,
            OUT => #TagArrayOut[1]);
END_IF;
```

Se ha producido un error al acceder a la variable "#Field[#index]". A pesar del error de lectura y acceso, el operando "#TagOut" devuelve el estado lógico "1" y la multiplicación se realiza con el valor "0.0". Si se produce este error se recomienda programar la instrucción "Consultar ID de error localmente" tras la multiplicación para capturar el error. La ID de error suministrada por la instrucción "Consultar ID de error localmente" se evalúa con una comparación. Si el operando "#TagID" devuelve la ID 16#2522, se trata de un error de lectura/acceso y el valor "100.0" del operando "#TagArrayIn[0]" se copia en el operando "#TagArrayOut[1]".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Consultar y eliminar errores en el código del programa (Página 265)

Tratar errores de ejecución del programa (Página 2932)

INIT_RD: Inicializar todos los datos remanentes

Descripción

La instrucción "Inicializar todos los datos remanentes" permite inicializar todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC al mismo tiempo. La instrucción solo puede ejecutarse dentro de un OB de arranque, puesto que su ejecución excedería el tiempo de ciclo de programa.

Sintaxis

Para la instrucción "Inicializar todos los datos remanentes" se utiliza la siguiente sintaxis:

SCL

```
INIT_RD(REQ := <Operando>
        RET_VAL := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar todos los datos remanentes":

Parámetro	Declaración	Tipo de datos	Descripción
REQ	Input	BOOL	Si la entrada REQ devuelve el estado lógico "1", se inicializan todos los datos remanentes.
RET_VAL	Output	INT	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B5	No puede ejecutarse la instrucción porque no ha sido programada dentro de un OB de arranque.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
INIT_RD(REQ := "Tag_REQ",
        RET_VAL := "Tag_Result");
```

Si el operando "Tag_REQ" devuelve el estado lógico "1", se ejecuta la instrucción. Se inicializan todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" permite detener la ejecución del programa durante un periodo predeterminado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible programar tiempos de retardo de hasta 32 767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve la información de error.

Sintaxis

Para la instrucción "Programar retardo" se utiliza la siguiente sintaxis:

```
SCL
WAIT(WT := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
WT	Input	INT	Tiempo de retardo en microsegundos (µs)

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

RUNTIME: Medir tiempo de ejecución

Descripción

La instrucción "Medir tiempo de ejecución" permite medir el tiempo de ejecución de todo el programa, bloques individuales o secuencias de comandos.

Si desea medir el tiempo de ejecución de todo el programa, efectúe la llamada de la instrucción "Medir tiempo de ejecución" en el OB1. El punto inicial para medir el tiempo de ejecución se determina con la primera llamada y la salida RET_VAL devuelve el tiempo de ejecución del programa tras la segunda llamada. El tiempo de ejecución medido incluye toda clase de procesos de la CPU que se hayan producido durante la ejecución del programa, como, p. ej., interrupciones a causa de eventos o comunicaciones de mayor prioridad. La instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU y escribe su valor en el parámetro de entrada/salida. La instrucción calcula el tiempo de ejecución actual del programa según la frecuencia interna del contador, y la escribe en la salida RET_VAL.

Si desea medir el tiempo de ejecución de bloques o secuencias de comandos individuales, necesita tres segmentos diferentes. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" en un segmento individual dentro del programa. Al efectuar esta primera llamada de la instrucción se establece el punto inicial para medir el tiempo de ejecución. A continuación, llame el bloque de programa o la secuencia de comando que desee en el siguiente segmento. Efectúe la llamada de la instrucción "Medir tiempo de ejecución" por segunda vez en otro segmento diferente y asigne al parámetro de entrada/salida la misma memoria que en la primera llamada de la instrucción. En el tercer segmento, la instrucción "Medir tiempo de ejecución" lee un contador interno de la CPU, calcula el tiempo de ejecución actual del bloque

de programa o secuencia de comando según la frecuencia interna del contador y lo escribe en la salida RET_VAL.

La instrucción "Medir tiempo de ejecución" utiliza un contador interno de alta frecuencia para calcular el tiempo. Cuando el contador se desborda, la instrucción devuelve valores <= 0.0. Estos valores de runtime deben ignorarse.

Nota

Puesto que el orden de las instrucciones de una secuencia de comandos se modifica mediante la compilación optimizada del programa, no es posible determinar el tiempo de ejecución de una secuencia de comandos con exactitud.

Sintaxis

Para la instrucción "Medir tiempo de ejecución" se utiliza la siguiente sintaxis:

SCL

```
RUNTIME(<Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	InOut	LREAL	Guarda el punto inicial de la medición del tiempo de ejecución
Valor de función		LREAL	Devuelve el tiempo de ejecución medido en segundos

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción mediante el cálculo del tiempo de ejecución de un bloque de programa:

SCL

```
"Tag_Result" := RUNTIME("Tag_Memory");

"Best_before_date_DB" ();

"Tag_Result" := RUNTIME("Tag_Memory");
```

Con la primera llamada de la instrucción se determina el punto inicial para la medición del tiempo de ejecución, y se guarda como referencia en una memora intermedia del operando "TagMemory" para la segunda llamada de la instrucción.

A continuación se llama el bloque de programa FB1 "Best_before_date".

Una vez se ha procesado el bloque de programa FB1, la instrucción se ejecuta por segunda vez. La segunda llamada de la instrucción calcula el tiempo de ejecución del bloque de programa y escribe el resultado en la salida "Tag_Result".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Temporizadores CEI (Página 4286)

Operaciones lógicas con palabras

DECO: Descodificar

Descripción

La instrucción "Descodificar" permite activar un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor del parámetro IN y activa el bit en el valor de salida cuya posición equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor del parámetro IN es mayor que 31, se ejecuta una instrucción modulo 32.

Sintaxis

Para la instrucción "Descodificar" se utiliza la siguiente sintaxis:

```
SCL
DECO(IN := <Expresión>)
DECO_WORD(IN := <expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	UINT	Posición del bit que se activa en el valor de salida.
_<tipo de datos>		Secuencias de bits Valor predeterminado: DWORD	Tipo de datos del valor de la función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Secuencias de bits	Valor de salida actual

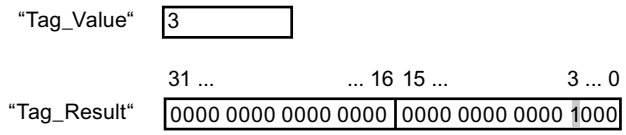
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := DECO(IN := "Tag_Value");
"Tag_Result2" := DECO_BYTE(IN := "Tag_Value2");
```

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



La instrucción lee el número "3" del valor del operando "Tag_Value" y activa el tercer bit del valor del operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ENCO: Codificar

Descripción

La instrucción "Codificar" permite leer el número del bit menos significativo que está activado en el valor de entrada y devolverlo como resultado.

Sintaxis

Para la instrucción "Codificar" se utiliza la siguiente sintaxis:

SCL

```
ENCO(IN := <Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	Secuencias de bits	Valor de entrada
Valor de función		INT	Número de bit del bit que se lee en el valor de entrada.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

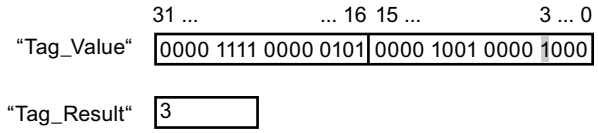
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ENCO(IN := "Tag_Value");
```

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



La instrucción lee el bit menos significativo que está activado en el operando "Tag_Value" y escribe la posición de bit "3" en el operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" selecciona, en función de un interruptor (parámetro G), uno de los parámetros IN0 o IN1 y devuelve su contenido como resultado. Si el parámetro G tiene el estado lógico "0", se copia el valor del parámetro IN0. Si el parámetro G tiene el estado lógico "1", se copia el valor del parámetro IN1 y se devuelve como valor de función.

Para poder ejecutar la instrucción, es imprescindible que las variables de todos los parámetros del tipo de datos sean de la misma clase.

Sintaxis

Para la instrucción "Seleccionar" se utiliza la siguiente sintaxis:

SCL

```
SEL(G := <Expresión>,
    IN0 := <Expresión>,
    IN1 := <Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos		
		S7-1200	S7-1500	Descripción
G	Input	BOOL	BOOL	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, DATE, CHAR, DT	Secuencias de bits, enteros, números en coma flotante, CHAR, temporizadores, DATE, TOD, LTOD, DT, LDT	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, TIME, TOD, DATE, CHAR, DT	Secuencias de bits, enteros, números en coma flotante, CHAR, temporizadores, DATE, TOD, LTOD, DT, LDT	Segundo valor de entrada
Valor de función		Secuencias de bits, enteros, números en coma flotante, TIME, TOD, DATE, CHAR, DT	Secuencias de bits, enteros, números en coma flotante, CHAR, temporizadores, DATE, TOD, LTOD, DT, LDT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := SEL(G := "Tag_Value",
                    IN0 := "Tag_0",
                    IN1 := "Tag_1");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	0	1
Tag_0	W#16#0000	W#16#4C
Tag_1	W#16#FFFF	D#16#5E
Tag_Result	W#16#0000	D#16#5E

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MUX: Multiplexar

Descripción

La instrucción "Multiplexar" copia el valor de un parámetro de entrada seleccionado y lo devuelve. Con el parámetro K se especifica el número del parámetro de entrada cuyo número se copiará. La numeración comienza por IN0 y continúa en orden ascendente con cada nueva entrada. Se puede declarar un máximo de 32 entradas.

Si el valor del parámetro K es mayor que el número de entradas y el parámetro INELSE no está activado, la salida de habilitación ENO se pone a "0".

En las entradas se permiten tipos de datos numéricos y tipos de datos de tiempo. Todas las variables parametrizadas deben ser del mismo tipo de datos.

El valor de función no es válido si se cumple una de las siguientes condiciones:

- El valor del parámetro K es mayor que el número de entradas disponibles.
- Se producen errores al ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplexar" se utiliza la siguiente sintaxis:

```
SCL
MUX(K := <Expresión>,
    IN0 := <Expresión>,
    IN1 := <Expresión>,
    INELSE := <Expresión>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplexar":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
K	Input	Enteros	Enteros	Indica el parámetro cuyo contenido se transfiere. • Si K = 0 => parámetro IN0 • Si K = 1 => parámetro IN1, etc.
IN0	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, DATE, TIME, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, LTOD, DATE, temporizadores, DT, LDT	Primer valor de entrada
IN1	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, DATE, TIME, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, LTOD, DATE, temporizadores, DT, LDT	Segundo valor de entrada
INn	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, DATE, TIME, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, LTOD, DATE, temporizadores, DT, LDT	Valores de entrada opcionales

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
INELSE	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, DATE, TIME, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, LTOD, DATE, temporizadores, DT, LDT	Indica el valor que se copia si K <> n.
Valor de función		Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, DATE, TIME, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TOD, LTOD, DATE, temporizadores, DT, LDT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := MUX(K := "Tag_Number",
                    IN0 := "Tag_1",
                    IN1 := "Tag_2",
                    INELSE := "Tag_3");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Number	1	4
Tag_1	DW#16#00000000	DW#16#00000000
Tag_2	DW#16#003E4A7D	DW#16#003E4A7D
Tag_3	DW#16#FFFF0000	DW#16#FFFF0000
Tag_Result	DW#16#003E4A7D	DW#16#FFFF0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

DEMUX: Desmultiplexar

Descripción

La instrucción "Desmultiplexar" permite transferir el valor del parámetro de entrada IN a un parámetro de salida seleccionado. La selección del parámetro de salida se efectúa en función del valor del parámetro K. El parámetro K indica el número del parámetro de salida al que se transfiere el valor del parámetro de entrada IN. Los demás parámetros de salida no cambian. La numeración comienza por OUT0 y continúa en orden ascendente con cada nueva salida. Se puede declarar un máximo de 32 parámetros de salida.

Si el valor del parámetro K es mayor que el número de parámetros de salida, la salida de habilitación ENO se pone a "0" y el valor del parámetro de entrada IN se transfiere al parámetro de salida OUTELSE.

El valor de función no es válido si se cumple una de las siguientes condiciones:

- El valor del parámetro K es mayor que el número de salidas disponibles.
- Se producen errores al ejecutar la instrucción.

Sintaxis

Para la instrucción "Desmultiplexar" se utiliza la siguiente sintaxis:

SCL

```
DEMUX(K := <Expresión>,
      IN := <Expresión>,
      OUT0 := <Operando>,
      OUT1 := <Operando>,
      OUTELSE := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desmultiplexar":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
K	Input	Enteros	Enteros	Indica la salida en la que se copia el valor de entrada (IN). • Si K = 0 => parámetro OUT0 • Si K = 1 => parámetro OUT1, etc.
IN	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	Valor de entrada
OUT0	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	Primera salida
OUT1	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	Segunda salida

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
OUTn	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	Salidas opcionales
OUTELSE	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, TIME, TOD, DATE, DT	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	Salida a la que se copia el valor de la entrada IN con K > n.

Encontrará más información sobre los tipos de datos disponibles en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
DEMUX(K := "Tag_Number",
      IN := "Tag_Value",
      OUT0 := "Tag_1",
      OUT1 := "Tag_2",
      OUTELSE := "Tag_3");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores de operandos concretos:

Valores de entrada de la instrucción "Desmultiplexar" antes de ejecutar el segmento

Parámetro	Operando	Valores	
K	Tag_Number	2	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#003E4A7D

Valores de salida de la instrucción "Desmultiplexar" tras ejecutar el segmento

Parámetro	Operando	Valores	
OUT0	Tag_1	sin cambios	sin cambios
OUT1	Tag_2	DW#16#FFFFFFFF	sin cambios
OUTELSE	Tag_3	sin cambios	DW#16#003E4A7D

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

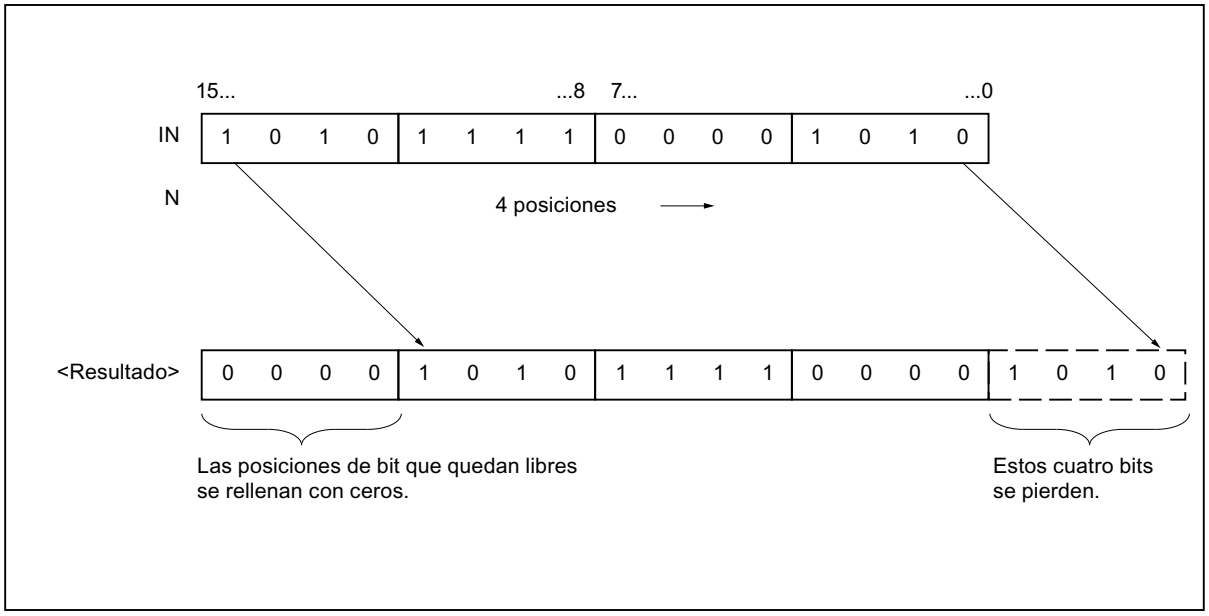
Con la instrucción "Desplazar a la derecha" se desplaza el contenido del parámetro IN de bit en bit hacia la derecha y se devuelve como valor de función. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", se devuelve como resultado el valor del parámetro IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del parámetro IN se desplaza hacia la derecha las posiciones de bit disponibles.

Los bits que quedan libres en el área izquierda del operando al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos Integer se desplaza cuatro posiciones de bit hacia la derecha:



Sintaxis

Para la instrucción "Desplazar a la derecha" se utiliza la siguiente sintaxis:

```
SCL
SHR(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	Valor que se desplaza
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de bits que se desplaza el valor (IN)
Valor de función		Secuencias de bits, enteros	Secuencias de bits, enteros	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := SHR(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	3
Valor de función	Tag_Result	0000 0111 1111 010 1

El valor del operando "Tag_Value" se desplaza tres posiciones de bit hacia la derecha. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SHL: Desplazar a la izquierda

Descripción

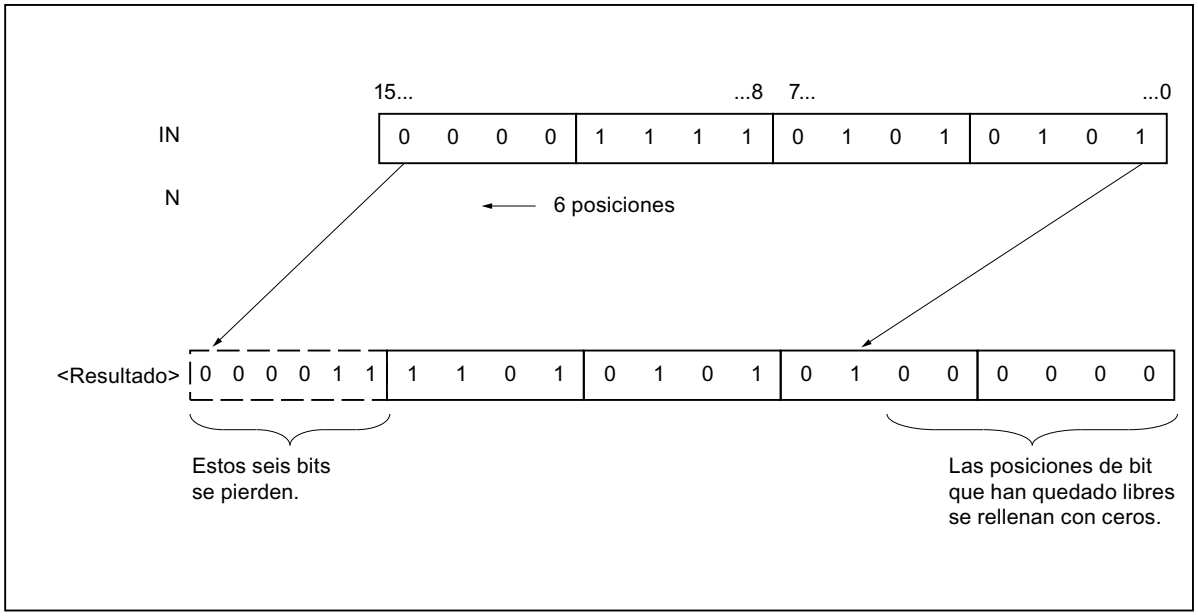
Con la instrucción "Desplazar a la izquierda" se desplaza el contenido del parámetro IN de bit en bit hacia la izquierda y se devuelve como valor de función. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", se devuelve como resultado el valor del parámetro IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del parámetro IN se desplaza hacia la izquierda las posiciones de bit disponibles.

Las posiciones que quedan libres al desplazar se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Sintaxis

Para la instrucción "Desplazar a la izquierda" se utiliza la siguiente sintaxis:

SCL

```
SHL(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	Valor que se desplaza
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de bits que se desplaza el valor (IN)
Valor de función		Secuencias de bits, enteros	Secuencias de bits, enteros	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := SHL(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	4
Valor de función	Tag_Result	1111 1010 1111 0000

El valor del operando "Tag_Value" se desplaza cuatro posiciones de bit hacia la izquierda. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ROR: Rotar a la derecha

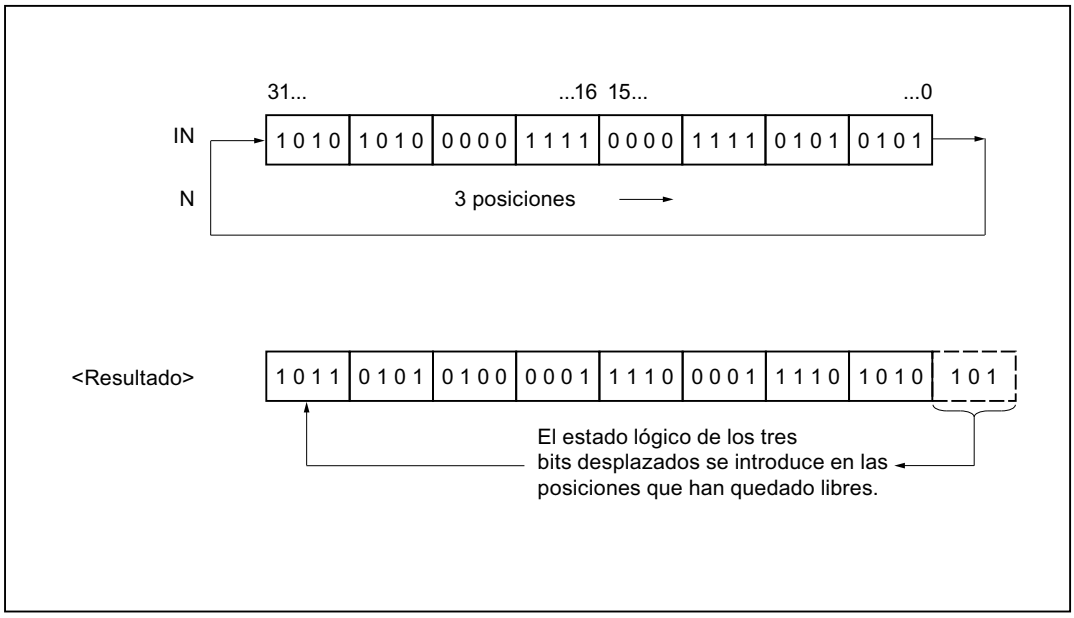
Descripción

La instrucción "Rotar a la derecha" permite rotar el contenido del parámetro IN de bit en bit hacia la derecha y asignar el resultado al operando indicado. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", se devuelve como resultado el valor de la entrada IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Sintaxis

Para la instrucción "Rotar a la derecha" se utiliza la siguiente sintaxis:

SCL

```
ROR(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	Valor que se rota
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de bits que rota el valor (IN)
Valor de función		Secuencias de bits, enteros	Secuencias de bits, enteros	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ROR(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0000 1111 1001 0101
N	Tag_Number	5
Valor de función	Tag_Result	1010 1000 0111 1100

El contenido del operando "Tag_Value" se rota cinco posiciones de bit hacia la derecha. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ROL: Rotar a la izquierda

Descripción

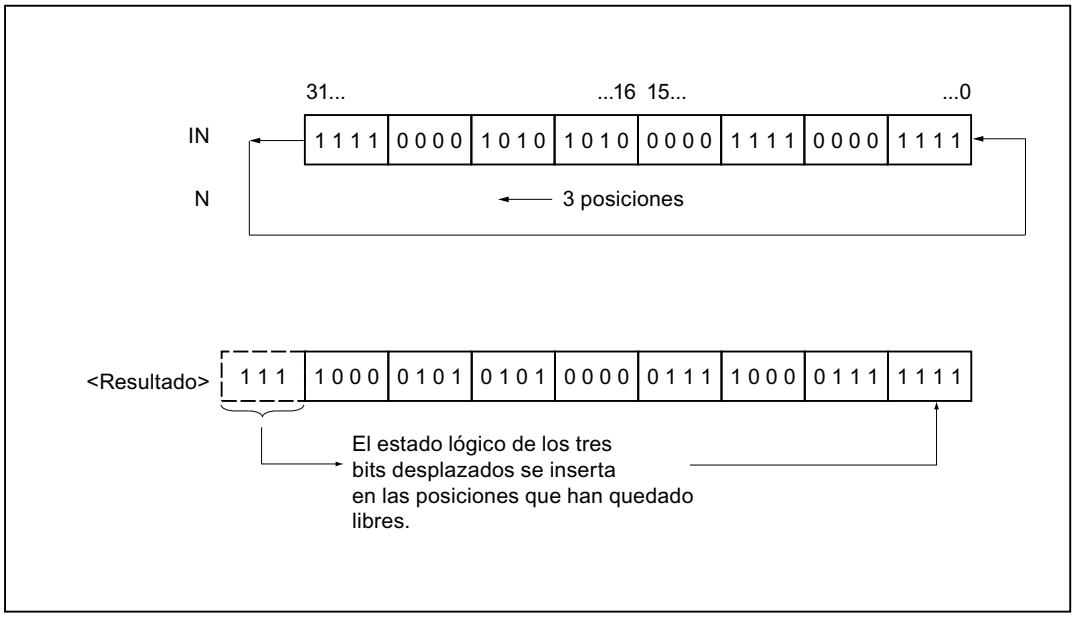
Con la instrucción "Rotar a la izquierda", el contenido del parámetro IN rota de bit en bit hacia la izquierda y se devuelve como valor de función. El parámetro N determina el número de bits

que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", se devuelve como resultado el valor de la entrada IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Sintaxis

Para la instrucción "Rotar a la izquierda" se utiliza la siguiente sintaxis:

SCL

```
ROL(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

Parámetro	Declaración	Tipo de datos		Descripción
		S7-1200	S7-1500	
IN	Input	Secuencias de bits, enteros	Secuencias de bits, enteros	Valor que se rota
N	Input	USINT, UINT, UDINT	USINT, UINT, UDINT, ULINT	Número de bits que rota el valor (IN)
Valor de función		Secuencias de bits, enteros	Secuencias de bits, enteros	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := ROL(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1010 1000 1111 0110
N	Tag_Number	5
Valor de función	Tag_Result	0001 1110 1101 0101

El contenido del operando "Tag_Value" se rota cinco posiciones de bit hacia la izquierda. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Otras instrucciones

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" ocupa los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de este es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(RESET := <Operando>,
          JOG := <Operando>,
          DRUM_EN := <Operando>,
          LST_STEP := <Operando>,
          EVENT1 - 16 := <Operando>,
          OUT1 - 16 => <Operando>,
          Q => <Operando>,
          OUT_WORD => <Operando>,
          ERR_CODE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Descripción
RESET	Input	BOOL	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	Número del último paso programado.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	Bit de evento (i); El estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	Bit de salida (j)
Q	Output	BOOL	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	Información de error

Parámetro	Declaración	Tipo de datos	Descripción
JOG_HIS	Static	BOOL	Bit de evolución del parámetro JOG
EOD	Static	BOOL	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	Paso predeterminado del circuito secuencial sucesivo
DSC	Static	BYTE	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	Paso actual del circuito secuencial sucesivo
DTBP	Static	WORD	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	Hora de sistema anterior
S_PRESET	Static	ARRAY of WORD	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET (solo si ET < PT), OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1"
  - Los parámetros ET, OA y CA se ponen a "0"
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET (solo si ET < PT), OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control discreto" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(CMD := <Operando>,
          O_FB := <Operando>,
          C_FB := <Operando>,
          Q => <Operando>,
          OA => <Operando>,
          CA => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Descripción
CMD	Input	BOOL	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	Entrada de realimentación al abrir
C_FB	Input	BOOL	Entrada de realimentación al cerrar
Q	Output	BOOL	Indica el estado del parámetro CMD
OA	Output	BOOL	Salida de alarma al abrir
CA	Output	BOOL	Salida de alarma al cerrar
ET	Static	DINT	Tiempo actual transcurrido; un ciclo de reloj = 1 ms.
PT	Static	DINT	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms.
PREV_TIME	Static	DWORD	Hora de sistema anterior
CMD_HIS	Static	BOOL	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

SCL

```
"DCAT_DB"(CMD := "Tag_Input_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          Q => "Tag_Output_Q",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se activa la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma

Parámetros de entrada								Parámetros de salida								
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Sintaxis

Para la instrucción "Alarma de tiempo para control del motor" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(O_CMD := <Operando>,
          C_CMD := <Operando>,
          S_CMD := <Operando>,
          O_FB := <Operando>,
          C_FB := <Operando>,
          OO => <Operando>,
          CO => <Operando>,
          OA => <Operando>,
          CA => <Operando>,
```

SCL

```
            Q => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Descripción
O_CMD	Input	BOOL	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	Entrada de la orden "Parar"
O_FB	Input	BOOL	Entrada de realimentación al abrir
C_FB	Input	BOOL	Entrada de realimentación al cerrar
OO	Output	BOOL	Salida "Abrir"
CO	Output	BOOL	Salida "Cerrar"
OA	Output	BOOL	Salida de alarma al abrir
CA	Output	BOOL	Salida de alarma al cerrar
Q	Output	BOOL	El estado lógico "0" indica una condición de error.
ET	Static	DINT	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	Hora de sistema anterior
O_HIS	Static	BOOL	Bit de evolución "Abrir"
C_HIS	Static	BOOL	Bit de evolución "Cerrar"

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

SCL

```
"MCAT_DB"(O_CMD := "Tag_Iput_O_CMD",
          C_CMD := "Tag_Input_C_CMD",
          S_CMD := "Tag_Input_S_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          OO => "Tag_OutputOpen",
          CO => "Tag_OutputClosed",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA",
          Q => "Tag_Output_Q");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una correspondencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en

el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Comparar bits de entrada con bits de una máscara" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(IN_BIT0 - 15 := <Operando>,
          CMP_STEP := <Operando>,
          OUT => <Operando>,
          ERR_CODE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Descripción
IN_BIT0	Input	BOOL	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	El bit de entrada 13 se compara con el bit 13 de la máscara.

Parámetro	Declaración	Tipo de datos	Descripción
IN_BIT14	Input	BOOL	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	Información de error
CMP_VAL	Static	ARRAY OF WORD	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último

paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Barrido de matriz" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(IN_BIT0 - 15 := <Operando>,
          OUT => <Operando>,
          OUT_STEP => <Operando>,
          ERR_CODE => <Operando>)
```

Sintaxis

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Descripción
IN_BIT0	Input	BOOL	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	El bit de entrada 7 se compara con el bit 7 de la máscara.

Parámetro	Declaración	Tipo de datos	Descripción
IN_BIT8	Input	BOOL	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
OUT_STEP	Output	BYTE	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	Información de error
LAST	Static	BYTE	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY OF WORD	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor de la ganancia en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse como supresor de perturbaciones o filtro de paso bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

```
SCL
<Instancia>(IN := <Operando>,
          SAMPLE_T := <Operando>,
          OUT => <Operando>,
          ERR_CODE => <Operando>)
```

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	REAL	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa. En el parámetro IN también se pueden indicar constantes.
SAMPLE_T	Input	INT	Tiempo de muestreo En el parámetro SAMPLE_T también se pueden indicar constantes.
OUT	Output	REAL	Resultado de la instrucción
ERR_CODE	Output	WORD	Información de error
LD_TIME	Static	REAL	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	Entrada anterior
PREV_OUT	Static	REAL	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

SCL

```
"LEAD_LAG_DB"(IN := "Tag_Input",
              SAMPLE_T := "Tag_Input_SAMPLE_T",
              OUT => "Tag_Output_Result",
              ERR_CODE => "Tag_ErrorCode");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

En el bloque de datos instancia "LEAD_LAG_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

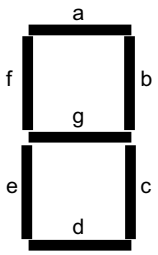
Cambiar formatos de visualización en el estado del programa (Página 3100)

SEG: Generar patrón de bits para el display de 7 segmentos

Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" convierte cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se deposita en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)	Display de siete segmentos
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	

1101	01011110	D
1110	01111001	E
1111	01110001	F

Sintaxis

Para la instrucción "Generar patrón de bits para el display de 7 segmentos" se utiliza la siguiente sintaxis:

SCL

```
SEG(IN := <Operando>,
    OUT => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	WORD	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	Patrón de bits para el display de 7 segmentos
Valor de función		VOID	Valor de función vacío

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
SEG(IN := "Tag_Input",
    OUT => "Tag_Output");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
		Hexadecimal	Binario
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW16#065B4F66	00000110 01011011 01001111 01100110 Visualización: 1234

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el operando. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

Sintaxis

Para la instrucción "Generar complemento a diez" se utiliza la siguiente sintaxis:

SCL

```
BCDCPL(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	Secuencias de bits	Número BCD de 7 dígitos
Valor de función		DWORD	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := BCDCPL("Tag_Input");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Operando	Valor*
Tag_Input	DW#16#01234567
Tag_Result	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" cuenta el número de bits de un operando que están puestos al estado lógico "1".

Sintaxis

Para la instrucción "Contar número de bits activados" se utiliza la siguiente sintaxis:

SCL

```
BITSUM(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	DWORD	Operando cuyos bits activados se cuentan
Valor de función		INT	Resultado de la instrucción

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := BITSUM("Tag_Input");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Operando	Valor*
Tag_Input	DW#16#12345678
Tag_Result	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GRAPH

Estructura GRAPH

Etapa y transición

Descripción

El componente de estructura "Etapa y transición" permite insertar simultáneamente una etapa y una transición en la cadena secuencial.

Consulte también:

Etapa (Página 4508)

Transición (Página 4508)

Consulte también

Etapas y transiciones (Página 2831)

Etapa (Página 4508)

Transición (Página 4508)

Fin de cadena (Página 4509)

Salto a etapa (Página 4509)

Rama alternativa (Página 4510)

Rama simultánea (Página 4510)

Cerrar rama (Página 4511)

Etapa

Descripción

Las etapas sirven para dividir las tareas de automatización complejas en tareas más pequeñas y claras que, seguidamente, se llevan a cabo por medio de acciones. Las diferentes etapas se organizan en cadenas secuenciales, de modo que al ejecutar el programa cada etapa se procesa en el orden especificado. Cada etapa debe tener un nombre y un número unívocos.

Para que el procesamiento se lleve a cabo realmente, la etapa debe activarse con una de las condiciones siguientes:

- La etapa se ha definido como inicial.
- Se cumple la transición de la etapa anterior.
- La etapa es llamada por una acción controlada por evento.

Una vez se han procesado todas las acciones se vuelve a desactivar la etapa.

Las etapas en las que no hay ninguna acción programada se denominan etapas vacías. Una etapa vacía tiene el mismo comportamiento que una etapa activa y la transición siguiente siempre se cumple.

Consulte también

Principios básicos de las etapas (Página 2831)

Etapa y transición (Página 4507)

Transición (Página 4508)

Fin de cadena (Página 4509)

Salto a etapa (Página 4509)

Rama alternativa (Página 4510)

Rama simultánea (Página 4510)

Cerrar rama (Página 4511)

Transición

Descripción

Las transiciones se encuentran entre una etapa y otra y contienen las condiciones para pasar de una etapa a la siguiente. Cuando se cumplen las condiciones de continuación de una transición, se activa la etapa siguiente y se procesan sus acciones. Las condiciones de una transición se programan en KOP o FUP.

Consulte también

Principios básicos de las transiciones (Página 2833)

Etapa y transición (Página 4507)

Etapa (Página 4508)

Fin de cadena (Página 4509)

Salto a etapa (Página 4509)

Rama alternativa (Página 4510)

Rama simultánea (Página 4510)

Cerrar rama (Página 4511)

Fin de cadena

Descripción

El componente de estructura "Fin de cadena" permite finalizar una cadena secuencial o una rama. En una rama simultánea, sin embargo, el fin de cadena debe estar precedido de una transición.

Nota

Si todas las ramas de una cadena secuencial se cierran con un fin de cadena, es posible volver a iniciar la cadena secuencial con el parámetro "INIT_SQ" o con el botón "Inicializar" de la Task Card "Test" en la paleta "Control secuencial".

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 4507)

Etapa (Página 4508)

Transición (Página 4508)

Salto a etapa (Página 4509)

Rama alternativa (Página 4510)

Rama simultánea (Página 4510)

Cerrar rama (Página 4511)

Salto a etapa

Descripción

Con un salto se prosigue la ejecución del programa en una etapa cualquiera dentro del bloque de función GRAPH. Los saltos pueden insertarse al final de la rama principal o de una rama alternativa, de modo que sea posible procesar la cadena secuencial cíclicamente. El salto y su destino se representan en la cadena secuencial en forma de flecha, de modo que en el destino del salto se indica la transición de origen.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 4507)

Etapa (Página 4508)

Transición (Página 4508)

Fin de cadena (Página 4509)

Rama alternativa (Página 4510)

Rama simultánea (Página 4510)

Cerrar rama (Página 4511)

Rama alternativa

Descripción

Las ramas alternativas permiten programar ramas O. Esto significa que, después de una etapa, se insertan ramas que empiezan con una transición. En función de la transición que se cumpla primero se ejecutará la rama correspondiente. Si se cumplen varias transiciones al mismo tiempo, tendrá prioridad la transición que esté más a la izquierda y se ejecutará la rama correspondiente. Las ramas alternativas finalizan con otra transición.

Es posible programar un máximo de 125 ramas alternativas en una cadena secuencial.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 4507)

Etapa (Página 4508)

Transición (Página 4508)

Fin de cadena (Página 4509)

Salto a etapa (Página 4509)

Rama simultánea (Página 4510)

Cerrar rama (Página 4511)

Rama simultánea

Descripción

Las ramas simultáneas permiten programar ramas Y. Esto significa que con una transición se activan varias etapas cuyas acciones se ejecutarán a continuación. Por ese motivo, las ramas simultáneas empiezan y terminan siempre con una etapa.

Las transiciones posteriores de las ramas simultáneas se encuentran en la rama principal, de modo que las diferentes ramas simultáneas pueden cerrarse en diferentes puntos de la rama principal. Tenga en cuenta que las ramas que se agrupan en una transición no conmutarán a la etapa siguiente, hasta que no se hayan ejecutado por completo dichas ramas.

Es posible programar un máximo de 249 ramas simultáneas en una cadena secuencial.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 4507)

Etapa (Página 4508)

Transición (Página 4508)

Fin de cadena (Página 4509)

Salto a etapa (Página 4509)

Rama alternativa (Página 4510)

Cerrar rama (Página 4511)

Cerrar rama

Descripción

El elemento "Cerrar rama" permite cerrar ramas simultáneas o alternativas y llevarlas a su rama matriz. Esto debe hacerse cuando la rama no se cierra con un salto o un fin de cadena. En una rama simultánea solo es posible insertar "Cerrar rama" después de una etapa.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 4507)

Etapa (Página 4508)

Transición (Página 4508)

Fin de cadena (Página 4509)

Salto a etapa (Página 4509)

Rama alternativa (Página 4510)

Rama simultánea (Página 4510)

Acciones GRAPH

Temporizadores

TP: Impulso

Descripción

La instrucción "Impulso" activa la salida Q por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. El parámetro Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico del parámetro Q mientras transcurra el tiempo PT.

El valor actual de tiempo se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Si cuando se alcanza el tiempo PT el estado lógico del parámetro IN es "0", se resetea el parámetro ET.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, el parámetro ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TP_TIME o TP_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej., "MyIEC_TIMER")
- Declaración como variable local del tipo TP_TIME,, TP_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede a los parámetros Q o ET.

Para poder ejecutar la instrucción "Impulso" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Impulso" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TP <Tipo de datos>, <Temporizador CEI>
     (IN := <Operando>
      PT := <Operando>
      Q => <Operando>
      ET => <Operando>
      )
```

Parámetros

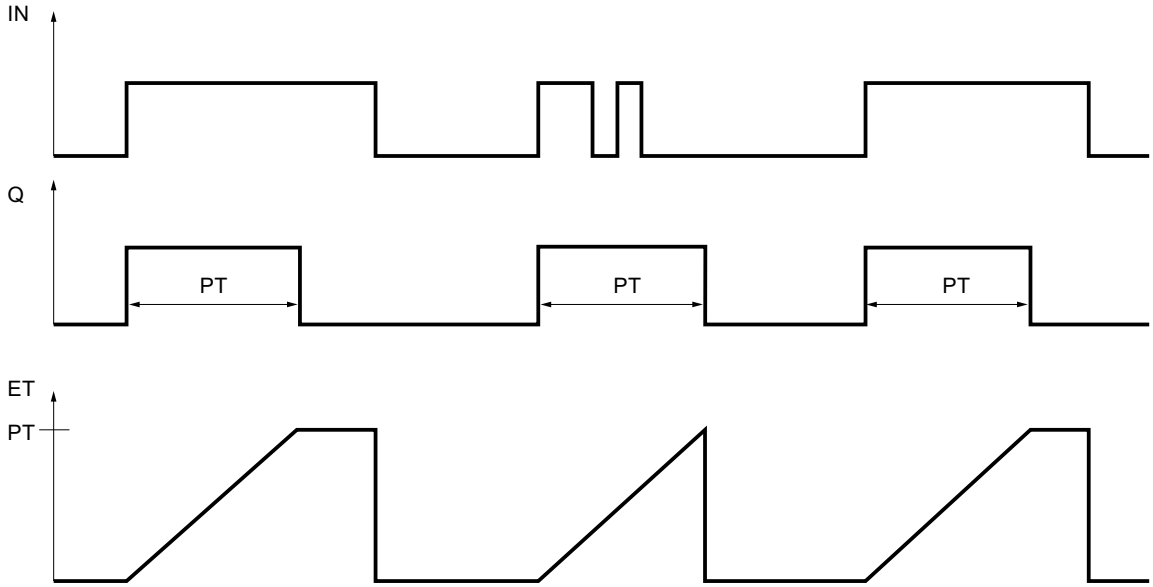
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Duración del impulso. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Salida de impulso
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TP TIME, "IEC_TP_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar y el operando "Tag_Status" se pone a "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTIME".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

TON: Retardo al conectar (CEI)

Descripción

La instrucción "Retardo al conectar" retarda la activación de la salida Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando se inicia la instrucción, se empieza a contar el tiempo programado PT. Una vez transcurrido el tiempo PT, el parámetro Q devuelve el estado lógico "1". El parámetro Q permanecerá activado mientras la entrada de arranque IN esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva el parámetro Q. La función de temporización se reinicia al detectarse un nuevo flanco de señal ascendente en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. El parámetro ET se inicializa en cuanto el estado lógico de la entrada IN cambia a "0".

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TON_TIME o TON_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej. "MyIEC_TIMER")
- Declaración como variable local del tipo TON_TIME, TON_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Retardo al conectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Retardo al conectar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TON <Tipo de datos>, <Temporizador CEI>
     (IN := <Operando>
```

GRAPH

```
PT := <Operando>
Q => <Operando>
ET => <Operando>
)
```

Parámetros

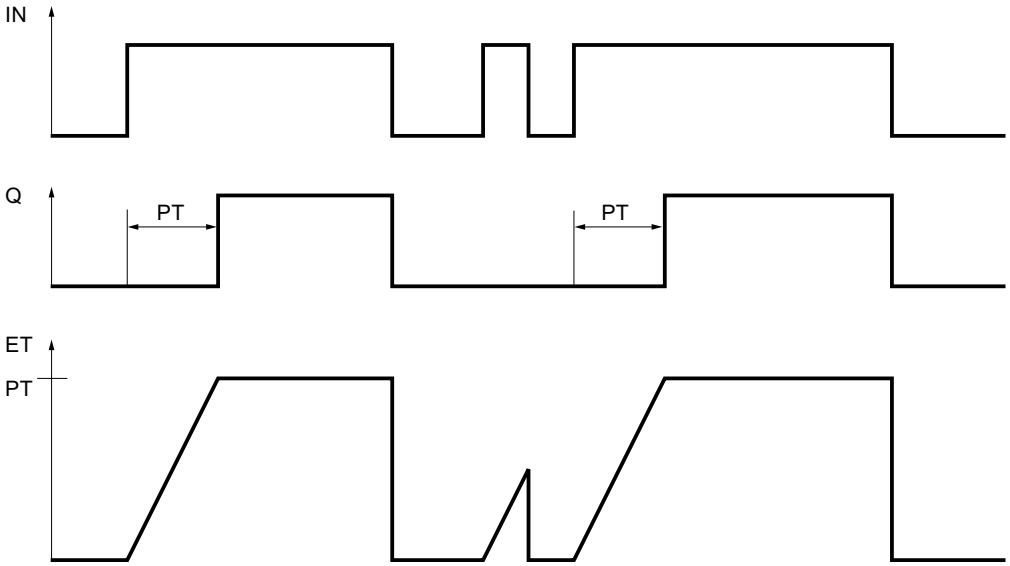
La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Estado lógico que se retarda por el tiempo PT.
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Retardo al conectar" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TON TIME, "IEC_TON_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar. Una vez transcurrido el tiempo, el operando "Tag_Status" se pone al estado lógico "1". El operando "Tag_Status" permanece a "1" mientras el operando "Tag_Start" tenga el estado lógico "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", se resetea el operando "Tag_Status".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

TOF: Retardo al desconectar

Descripción

La instrucción "Retardo al desconectar" retarda la desactivación de la salida Q por el tiempo programado PT. El parámetro Q se activa cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico del parámetro IN cambia nuevamente a "0", el tiempo programado PT deja de contar. El parámetro Q permanecerá activado mientras transcurra el tiempo PT. Una vez transcurrido el tiempo PT se inicializa el parámetro Q. Si el estado lógico del parámetro IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico del parámetro Q permanece puesto a "1".

El valor de tiempo actual se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el parámetro ET permanecerá al valor actual hasta que el parámetro IN vuelva a cambiar a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TOF_TIME o TOF_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej. "MyIEC_TIMER")
- Declaración como variable local del tipo TOF_TIME, TOF_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Retardo al desconectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Retardo al desconectar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TOF <Tipo de datos>, <Temporizador CEI>
     (IN := <Operando>
      PT := <Operando>
      Q => <Operando>
      ET => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

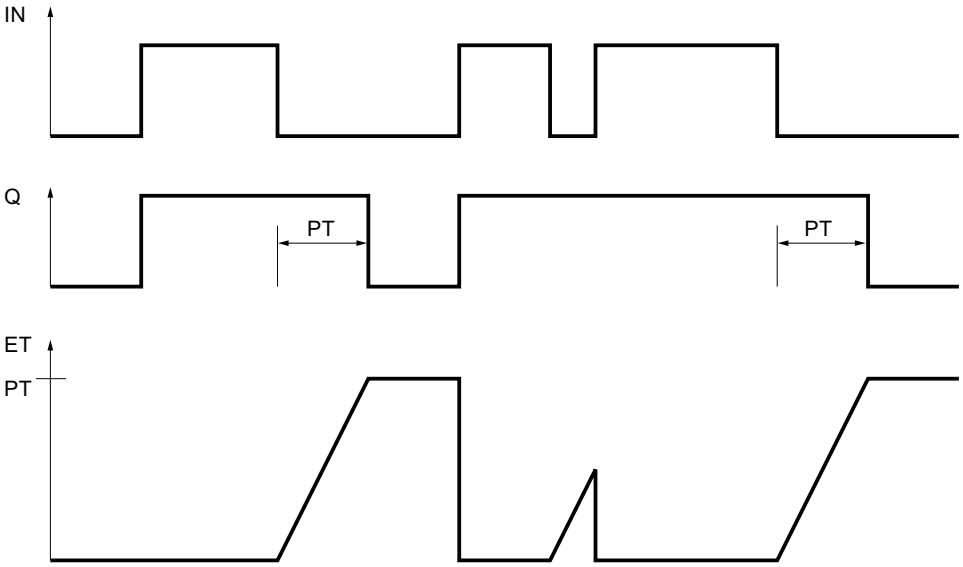
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L, P	Estado lógico que se retarda por el tiempo PT.
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Retardo al desconectar" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TOF TIME, "IEC_TOF_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Si se produce un cambio del estado lógico del operando "Tag_Start" de "0" a "1", el operando se pone a "Tag_Status". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", el tiempo programado en el parámetro PT empieza a contar. Mientras transcurra el tiempo, permanecerá activado el operando "Tag_Status". Una vez transcurrido el tiempo, el operando "Tag_Status" se desactiva. El valor actual de tiempo se guarda en el operando "Tag_ElapsedTIME".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

TONR: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PT. Cuando el estado lógico de la entrada IN cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y se empieza a contar el tiempo PT. Mientras transcurre el tiempo PT se van acumulando los valores de tiempo que se leen cuando el estado lógico de la entrada IN es "1". El tiempo acumulado se deposita en la salida ET y se puede consultar allí. Una vez se ha alcanzado el tiempo PT, la salida Q devuelve el estado lógico "1". El parámetro Q permanece a "1" aunque el estado lógico del parámetro IN cambie de "1" a "0" (flanco de señal descendente).

La entrada R desactiva las salidas ET y Q independientemente del estado lógico de la entrada de arranque.

A cada llamada de la instrucción "Acumulador de tiempo" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. El temporizador CEI es una estructura del tipo de datos IEC_TIMER, IEC_LTIMER, TONR_TIME o TONR_LTIME que se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_TIMER o IEC_LTIMER (p. ej. "MyIEC_TIMER")
- Declaración como variable local del tipo TONR_TIME, TONR_LTIME, IEC_TIMER o IEC_LTIMER en la sección "Static" de un bloque (p. ej. #MyIEC_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

Para poder ejecutar la instrucción "Acumulador de tiempo" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Acumulador de tiempo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TONR <Tipo de datos>, <Temporizador CEI>
```

GRAPH

```
(IN := <Operando>
 R := <Operando>
 PT := <Operando>
 Q => <Operando>
 ET => <Operando>
 )
```

Parámetros

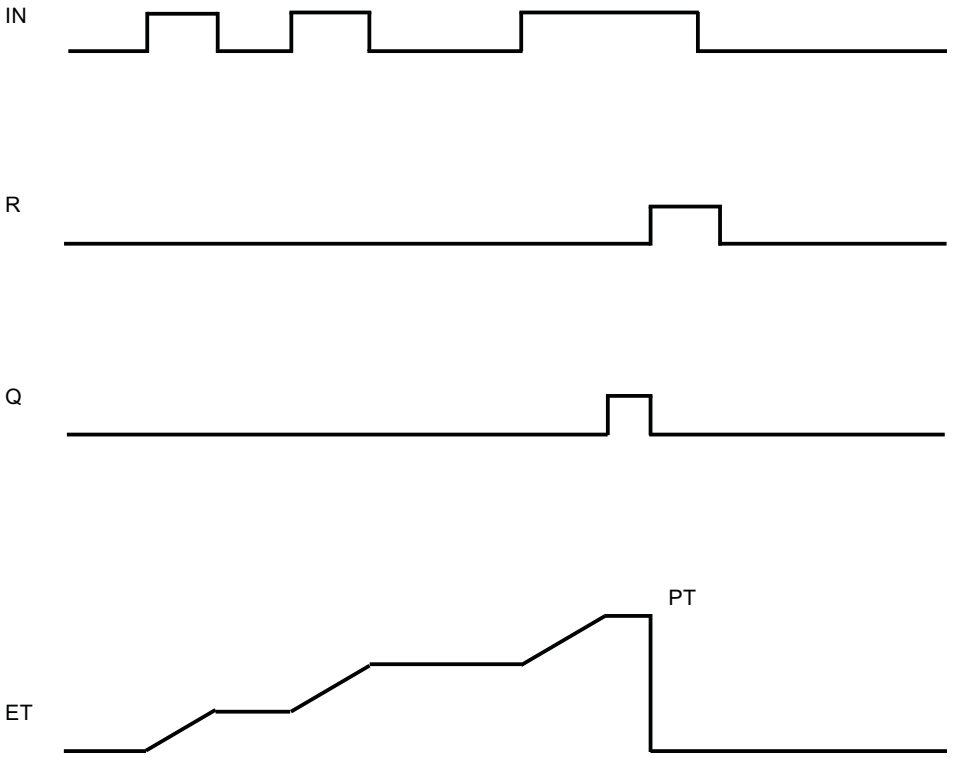
La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de arranque
R	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de reset
PT	Input	TIME, LTIME	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L, P	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME, LTIME	I, Q, M, D, L, P	Tiempo acumulado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Acumulador de tiempo":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TONR TIME, "IEC_TONR_DB"
      (IN := "Tag_Start"
       R := "Tag_Reset"
       PT := "Tag_PresetTIME"
       Q => "Tag_Status"
       ET => "Tag_ElapsedTIME"
       )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar. Mientras transcurre el tiempo, se van acumulando los valores de tiempo que se leen cuando el estado lógico del operando "Tag_Start" es "1". El tiempo acumulado se almacena en el operando "Tag_ElapsedTIME". Una vez alcanzado el valor de tiempo indicado en el parámetro PT, el operando "Tag_Status" adopta el estado lógico "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTIME".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Contadores

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor de contaje actual del parámetro CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Cuando se alcanza el valor límite superior, el estado lógico del parámetro CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. El estado lógico del parámetro Q es determinado por el parámetro PV. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0".

El valor del parámetro CV se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico del parámetro CU no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Variable local

- CTU_SINT / CTU_USINT
- CTU_INT / CTU_UINT
- CTU_DINT / CTU_UDINT
- CTU_LINT / CTU_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTU_<tipo de datos> o IEC_<contador> en la sección "Static" de un bloque (p. ej. #MyCTU_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador ascendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CTU <Tipo de datos>, <Contador CEI>
     (CU := <Operando>
      R := <Operando>
      PV := <Operando>
      Q => <Operando>
      CV => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de reset
PV	Input	Enteros	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL CTU INT, "IEC_CTU_DB"
     (CU := "Tag_Start"
      R := "Tag_ResetCOUNTER"
      PV := "Tag_PresetValue"
      Q => "Tag_Status"
      CV => "Tag_CounterValue"
      )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor de contaje actual del operando "Tag_CounterValue" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos (INT = 32767).

El valor del parámetro PV se utiliza como límite para determinar la salida "Tag_Status". La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea mayor o igual al valor del operando "Tag_PresetValue". En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos indicado. Cuando se alcanza el valor límite inferior, el estado lógico del parámetro CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. Si el valor de contaje actual es menor o igual a cero, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0".

El valor del parámetro CV se pone al valor del parámetro PV cuando el estado lógico del parámetro LD cambia a "1". Mientras el parámetro LD tenga el estado lógico "1", el estado lógico del parámetro CD no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Variable local

- CTD_SINT / CTD_USINT
- CTD_INT / CTD_UINT
- CTD_DINT / CTD_UDINT
- CTD_LINT / CTD_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER

- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTD_<tipo de datos> o IEC_<contador> en la sección "Static" de un bloque (p. ej. #MyCTD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CTD <Tipo de datos>, <Contador CEI>
     (CD := <Operando>
      LD := <Operando>
      PV := <Operando>
      Q => <Operando>
      CV => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CD	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL CTD INT, "IEC_CTD_DB"
     (CD := "Tag_Start"
      LD := "Tag_LoadPV"
      PV := "Tag_PresetValue"
      Q => "Tag_Status"
      CV => "Tag_CounterValue"
      )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", se ejecuta la instrucción "Contador descendente" y el valor de la salida "Tag_CounterValue" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos indicado (INT = -32768).

La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual del parámetro CV se incrementa en uno. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual del parámetro CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje actual no cambia en el parámetro CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado en el parámetro CV. Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado, ya no se decrementa el valor de contaje.

Cuando el estado lógico del parámetro LD cambia a "1", el valor de contaje del parámetro CV se pone al valor del parámetro PV. Mientras el parámetro LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetro R sea "1", un cambio del estado lógico de los parámetros CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en el parámetro QU. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro QU adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QU es "0".

El estado del contador descendente se puede consultar en el parámetro QD. Si el valor de contaje actual es menor o igual a cero, el parámetro QD adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QD es "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI es una estructura con uno de los tipos de datos siguientes:

Bloque de datos del tipo de datos de sistema IEC_<contador> (Shared DB)

- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Variable local

- CTUD_SINT / CTUD_USINT
- CTUD_INT / CTUD_UINT
- CTUD_DINT / CTUD_UDINT
- CTUD_LINT / CTUD_ULINT
- IEC_SCOUNTER / IEC_USCOUNTER
- IEC_COUNTER / IEC_UCOUNTER
- IEC_DCOUNTER / IEC_UDCOUNTER
- IEC_LCOUNTER / IEC_ULCOUNTER

Un contador CEI se puede declarar del modo siguiente:

- Declaración de un bloque de datos del tipo de datos de sistema IEC_<contador> (p. ej. "MyIEC_COUNTER")
- Declaración como variable local del tipo CTUD_<tipo de datos> o IEC_<contador> en la sección "Static" de un bloque (p. ej. #MyCTUD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si el contador CEI se crea en un bloque de datos propio (instancia individual), el bloque de datos de instancia se crea de forma estándar con "acceso optimizado al bloque" y las distintas variables se definen como remanentes. Encontrará más información sobre la configuración de la remanencia en un bloque de datos de instancia en "Consulte también".

Si el contador CEI se crea como variable local (multiinstancia) en un bloque de función con "acceso optimizado al bloque", esta se define como remanente en la interfaz del bloque.

Para poder ejecutar la instrucción "Contador ascendente - descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Contador ascendente - descendente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CTUD <Tipo de datos>, <Contador CEI>
     (CU := <Operando>
      CD := <Operando>
      R := <Operando>
      LD := <Operando>
      PV := <Operando>
      QU => <Operando>
      QD => <Operando>
```

GRAPH

```
      CV => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L o constante	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P o constante	Entrada de carga
PV	Input	Enteros	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU.
QU	Output	BOOL	I, Q, M, D, L	Estado del contador ascendente
QD	Output	BOOL	I, Q, M, D, L	Estado del contador descendente
CV	Output	Enteros, CHAR, DATE	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL CTUD INT, "IEC_CTUD_DB"
     (CU := "Tag_StartCTU"
      CD := "Tag_StartCTD"
      R := "Tag_ResetCOUNTER"
      LD := "Tag_LoadPV"
      PV := "Tag_PresetValue"
      QU => "Tag_CounterStatusUP"
      QD => "Tag_CounterStatusDOWN"
      CV => "Tag_CounterValue"
```

GRAPH
)

Si el estado lógico de la entrada "Tag_StartCTU" o "Tag_StartCTD" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Contador ascendente - descendente". Cuando se detecta un flanco de señal ascendente en la entrada "Tag_StartCTU", el valor de contaje actual se incrementa en uno y se deposita en la salida "Tag_CounterValue". Si se detecta un flanco de señal ascendente en la entrada "Tag_StartCTD", el valor de contaje se decrementa en uno y se deposita en la salida "Tag_CounterValue". Si se detecta un flanco de señal ascendente en la entrada CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior (32767). Cuando se detecta un flanco de señal ascendente en la entrada CD, el valor de contaje se decrementa hasta alcanzar el valor límite inferior INT = -32768.

La salida "Tag_CounterStatusUP" devuelve el estado lógico "1" mientras el valor de contaje actual sea mayor o igual al valor de la entrada "Tag_PresetValue". En todos los demás casos, la salida "Tag_CounterStatusUP" devuelve el estado lógico "0".

La salida "Tag_CounterStatusDOWN" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_CounterStatusDOWN" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Ajustar la remanencia en un bloque de datos de instancia (Página 2951)

Remanencia de las variables locales (Página 2660)

Funciones matemáticas

NEG: Generar complemento a dos

Descripción

Con la instrucción "Generar complemento a dos" se cambia el signo del valor del operando. Por ejemplo, si tiene un valor positivo, se deposita el equivalente negativo de este valor.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT (sólo válido para números enteros).

Sintaxis

Para la instrucción "Generar complemento a dos" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := NEG_<Tipo de datos>(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a dos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P	Complemento a dos del valor de entrada

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := NEG_REAL("Tag_InValue")
```

La instrucción cambia el signo del valor en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ABS: Calcular valor absoluto

Descripción

Con la instrucción "Calcular valor absoluto" se calcula el valor absoluto del valor del operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Calcular valor absoluto" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := ABS(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor absoluto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	SINT, INT, DINT, LINT, números en coma flotante	I, Q, M, D, L, P	Valor absoluto del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := ABS("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	-6.234
Tag_OutValue	6.234

La instrucción calcula el valor absoluto del valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas disponibles IN1 e IN2 y escribe el valor más pequeño en la salida OUT.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Determinar mínimo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MIN <Tipo de datos>
     (IN1 := <Operando>
      IN2 := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
OUT	Output	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MIN INT
     (IN1 := "TagIn_Value1"
      IN2 := "TagIn_Value2"
      OUT => "Tag_Minimum"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
OUT	Tag_Minimum	12222

La instrucción compara los valores de los operandos indicados y copia el valor menor ("TagIn_Value1") en la salida "Tag_Minimum".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas IN1 e IN2 y escribe el valor mayor en la salida OUT.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- La conversión implícita de los tipos de datos falla durante la ejecución de la instrucción.
- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Determinar máximo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MAX <Tipo de datos>
     (IN1 := <Operando>
```

GRAPH

```
      IN2 := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN2	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
OUT	Output	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MAX INT
     (IN1 := "TagIn_Value1"
      IN2 := "TagIn_Value2"
      OUT => "Tag_Maximum"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
OUT	Tag_Maximum	14444

La instrucción compara los valores de los operandos indicados y copia el valor mayor ("TagIn_Value2") en la salida "Tag_Maximum".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor de la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no está definido. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor del parámetro MN es mayor que el valor del parámetro MX.

Sintaxis

Para la instrucción "Ajustar valor límite" se utiliza la siguiente sintaxis:

GRAPH

```
CALL LIMIT <Tipo de datos>
     (MN := <Operando>
      IN := <Operando>
      MX := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MN	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Límite inferior
IN	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Valor de entrada
MX	Input	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Límite superior
OUT	Output	Enteros, números en coma flotante, TIME, LTIME, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P	Resultado
Los tipos de datos TOD, LTOD, DATE y LDT solo se pueden utilizar cuando la verificación CEI está desactivada.				

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL LIMIT INT
      (MN := "Tag_LowLimit"
       IN := "Tag_InputValue"
       MX := "Tag_HighLimit"
       OUT => "Tag_Result"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000

Parámetro	Operando	Valor
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

El valor del operando "Tag_InputValue" se compara con los valores de los operandos "Tag_LowLimit" y "Tag_HighLimit". Puesto que el valor del operando "Tag_InputValue" es menor que el valor límite inferior, el valor del operando "Tag_LowLimit" se copia en la salida "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

SQR: Calcular cuadrado

Descripción

Con la instrucción "Calcular cuadrado" se forma el cuadrado del valor del operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular cuadrado" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SQR(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular cuadrado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Cuadrado del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SQR("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	5.0
Tag_OutValue	25.0

La instrucción eleva al cuadrado el valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

SQRT: Calcular raíz cuadrada

Descripción

Con la instrucción "Calcular raíz cuadrada" se extrae la raíz cuadrada del valor del operando. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero se devuelve un número en coma flotante no válido. Si el valor del operando es "0", el resultado también es "0".

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando es negativo.

Sintaxis

Para la instrucción "Calcular raíz cuadrada" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SQRT(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular raíz cuadrada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L	Raíz cuadrada del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SQRT("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	25.0
Tag_OutValue	5.0

La instrucción extrae la raíz cuadrada del valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Números en coma flotante no válidos (Página 2340)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2.718282) del valor del operando. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero se devuelve un número en coma flotante no válido.

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando es negativo.

Sintaxis

Para la instrucción "Calcular logaritmo natural" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := LN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular logaritmo natural":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Logaritmo natural del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := LN("Tag_InValue")
```

La instrucción calcula el logaritmo natural del valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Números en coma flotante no válidos (Página 2340)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2,718282) del valor del operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor exponencial" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := EXP(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor exponencial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Valor exponencial del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := EXP("Tag_InValue")
```

La instrucción calcula la potencia en base e del valor del operando "Tag_InValue" e indica el resultado (e^{IN}) en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Números en coma flotante no válidos (Página 2340)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno de un ángulo. El tamaño del ángulo se indica en radianes en el operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de seno" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SIN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de seno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Seno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SIN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	+1,570796 (π/2)
Tag_OutValue	1.0

La instrucción calcula el seno del ángulo indicado en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno de un ángulo. El tamaño del ángulo se indica en radianes en el operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de coseno" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := COS(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de coseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Coseno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := COS("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	+1,570796 (π/2)
Tag_OutValue	0

La instrucción calcula el coseno del ángulo indicado en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente de un ángulo. El tamaño del ángulo se indica en radianes en el operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de tangente" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := TAN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de tangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Tangente del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := TAN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	+3.141593 (π)
Tag_OutValue	0

La instrucción calcula la tangente del ángulo indicado en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno indicado en el operando el tamaño del ángulo que corresponde a este valor. En el operando solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado puede estar comprendido entre -π/2 y +π/2.

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando está fuera del rango de valores permitido (-1 hasta +1).

Sintaxis

Para la instrucción "Calcular valor de arcoseno" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := ASIN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de seno
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ASIN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1.0
Tag_OutValue	+1,570796 (π/2)

La instrucción calcula el tamaño del ángulo que corresponde al valor de seno del operando "Tag_InValue". El resultado de la instrucción se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno indicado en el operando el tamaño del ángulo que corresponde a este valor. En el operando solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado puede estar comprendido entre 0 y +π.

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando está fuera del rango de valores permitido (-1 hasta +1).

Sintaxis

Para la instrucción "Calcular valor de arcocoseno" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := ACOS(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcocoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de coseno
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := ACOS("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	0
Tag_OutValue	+1,570796 (π/2)

La instrucción calcula el tamaño del ángulo que corresponde al valor de coseno del operando "Tag_InValue". El resultado de la instrucción se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente indicado en el operando el tamaño del ángulo que corresponde a este valor. En el operando solo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado puede estar comprendido entre -π/2 y +π/2.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de arcotangente" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ATAN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcotangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de tangente
<Resultado>	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ATAN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1.0
Tag_OutValue	+0,785398 (π/4)

La instrucción calcula el tamaño del ángulo que equivale al valor de tangente del operando "Tag_InValue". El resultado de la instrucción se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

FRAC: Determinar decimales

Descripción

La instrucción "Determinar decimales" permite determinar los decimales del valor del operando. Por ejemplo, si el valor del operando es 123,4567, el resultado devuelve el valor 0,4567.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Determinar decimales" se utiliza la siguiente sintaxis:

GRAPH

```
CALL FRAC <Tipo de datos>
     (IN := <Operando>
     RET_VAL => <Operando>
     )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar decimales":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada cuyos decimales se determinan
RET_VAL	Output	Números en coma flotante	I, Q, M, D, L, P	Decimales del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL FRAC REAL
```

GRAPH

```
(IN := "Tag_InValue"
RET_VAL => "Tag_OutValue"
)
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	2.555
Tag_OutValue	0.555

La instrucción copia los decimales del valor del operando "Tag_InValue" en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Transferencia

Deserialize: Deserializar

Descripción

Con la instrucción "Deserializar", la forma de representación secuencial de un tipo de datos PLC (UDT) se vuelve a convertir en un tipo de datos PLC y se rellena todo el contenido.

El área de memoria en la que se encuentra la forma de representación secuencial de un tipo de datos PLC debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

Esta instrucción permite reconvertir paso a paso a su estado original varias formas de representación secuenciales de tipos de datos PLC convertidos.

Nota

Para reconvertir una sola forma de representación secuencial de un tipo de datos PLC (UDT) también se puede utilizar directamente la instrucción "TRCV: Recibir datos a través de la conexión".

Sintaxis

Para la instrucción "Deserializar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL Deserialize
     (SRC_ARRAY := <Operando>
      DEST_VARIABLE := <Operando>
      POS := <Operando>
      RET_VAL => <Operando>
      )
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Deserializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC_ARRAY	Input	VARIANT	I, Q, D, L	Bloque de datos global en el que se guarda el flujo de datos creado
DEST_VARIABLE	InOut	VARIANT	I, Q, D, L	Variable en la que se guarda el tipo de datos PLC (UDT) reconvertido
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_ARRAY y DEST_VARIABLE se solapan.
8136	El bloque de datos del parámetro DEST_VARIABLE no es un bloque con acceso estándar.
8150	El tipo de datos VARIANT del parámetro SRC_ARRAY no contiene ningún valor.
8151	Error de creación de código en el parámetro SRC_ARRAY
8153	En el parámetro SRC_ARRAY no hay suficiente memoria disponible.
8250	El tipo de datos VARIANT del parámetro DEST_VARIABLE no contiene ningún valor.
8251	Error de creación de código en el parámetro DEST_VARIABLE

Código de error* (W#16#...)	Explicación
8254	Tipo de datos no válido en el parámetro DEST_VARIABLE
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL Deserialize
     (SRC_ARRAY := "Buffer".Field
      DEST_VARIABLE := "Target".Client
      POS := #BufferPos
      RET_VAL => #Error
      )
```

La instrucción "Deserializar" deserializa la forma de representación secuencial de los datos de cliente del bloque de datos "Buffer" y los escribe en el bloque de datos "Target". El número de bytes que ocupan los datos de cliente convertidos se guarda en el operando "#BufferPos".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Serialize: Serializar

Descripción

Mediante la instrucción "Serializar" se convierten varios tipos de datos PLC (UDT) a una forma de representación secuencial sin que se pierdan partes de su estructura.

La instrucción permite guardar temporalmente varios datos estructurados del programa en un búfer, ubicado preferentemente en un bloque de datos global, y enviarlos a otra CPU. El área de memoria en la que se depositen los tipos de datos PLC convertidos debe ser del tipo de datos ARRAY of BYTE y tener declarado el acceso estándar. Antes de la conversión asegúrese de que hay suficiente memoria disponible.

El operando del parámetro POS contiene la información del número de bytes que ocupan los tipos de datos PLC convertidos.

Nota

Para enviar un solo tipo de datos PLC (UDT), se puede llamar directamente la instrucción "TSEND: Enviar datos a través de la conexión".

Sintaxis

Para la instrucción "Serializar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL Serialize
     (SRC_VARIABLE := <Operando>
      DEST_ARRAY := <Operando>
      POS := <Operando>
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Serializar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC_VARIABLE	Input	VARIANT	I, Q, D, L	Tipo de datos PLC (UDT) que se convierte a una forma de representación secuencial.
DEST_ARRAY	InOut	VARIANT	I, Q, D, L	Bloque de datos en el que se guarda el flujo de datos generado.
POS	InOut	DINT	I, Q, M, D, L	Número de bytes que ocupan los tipos de datos PLC convertidos. El parámetro POS se calcula basado en cero.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B0	Las áreas de memoria de los parámetros SRC_VARIABLE y DEST_ARRAY se solapan.

Código de error* (W#16#...)	Explicación
8150	El tipo de datos VARIANT del parámetro SRC_VARIABLE no contiene ningún valor.
8152	Error de creación de código en el parámetro SRC_VARIABLE
8236	El bloque de datos del parámetro DEST_ARRAY no es un bloque con acceso estándar.
8250	El tipo de datos VARIANT del parámetro DEST_ARRAY no contiene ningún valor.
8252	Error de creación de código en el parámetro DEST_ARRAY
8253	En el parámetro DEST_ARRAY no hay suficiente memoria disponible.
8254	Tipo de datos no válido en el parámetro DEST_ARRAY
8382	El valor del parámetro POS se encuentra fuera de los valores límite del array.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL Serialize
     (SRC_VARIABLE := "Source".Client
      DEST_ARRAY := "Buffer".Field
      POS := #BufferPos
      RET_VAL => #Error
      )
```

La instrucción "Serializar" serializa los datos de cliente del bloque de datos "Source" y los escribe como representación secuencial en el bloque de datos "Buffer". El número de bytes que ocupa la representación secuencial se guarda en el operando "#BufferPos".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

MOVE_BLK: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento del parámetro IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MOVE_BLK <Tipo de datos>_<Tipo de datos>
     (IN := <Operando>
      COUNT := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1)]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT [1]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MOVE_BLK INT_UINT
     (IN := #a_array[2]
      COUNT := "Tag_Count"
      OUT => #b_array[1]
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

MOVE_BLK_VARIANT: Copiar área

Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). Se pueden copiar elementos de un array en otro array del mismo tipo de datos. Los tamaños (número de elementos) de los arrays de origen y de destino pueden ser diferentes. Se pueden copiar varios elementos dentro de un array o bien copiar los elementos uno a uno.

En el momento de utilizar la instrucción para crear un bloque, aún no tiene que conocerse el array, puesto que el origen y el destino se transfieren por VARIANT.

El contaje de los parámetros SRC_INDEX y DEST_INDEX siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

La instrucción no se ejecuta si se copian más datos de los que están disponibles.

Si el puntero VARIANT (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MOVE_BLK_VARIANT
      (SRC := <Operando>
       COUNT := <Operando>
       SRC_INDEX := <Operando>
       DEST_INDEX := <Operando>
       DEST := <Operando>
       RET_VAL => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC	Input	VARIANT (array o elemento individual)	L	El área de origen de la que se copia
COUNT	Input	UDINT	I, Q, M, D, L o constante	Número de elementos que se copian Asigne al parámetro COUNT el valor "1" si en el parámetro SRC o en el parámetro DEST no hay indicado ningún Array.
SRC_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro SRC_INDEX se calcula basado en cero. Si en el parámetro SRC hay indicado un Array, el entero del parámetro SRC_INDEX indica el primer elemento del área de origen de la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro SRC no hay indicado ningún Array o solo hay indicado un elemento individual de un Array, asigne al parámetro SRC_INDEX el valor "0".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DEST_INDEX	Input	DINT	I, Q, M, D, L o constante	• El parámetro DEST_INDEX se calcula basado en cero. Si en el parámetro DEST hay indicado un Array, el entero del parámetro DEST_INDEX indica el primer elemento del área de destino en la que debe copiarse. Esto es así independientemente de los límites declarados del Array. • Si en el parámetro DEST no hay indicado ningún Array, asigne al parámetro DEST_INDEX el valor "0".
DEST	Output	VARIANT	L	Área de destino a la que se copia el contenido del área de origen.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
8151	No es posible acceder al parámetro SRC.
8152	El operando del parámetro SRC no está tipificado.
8153	Error de creación de código en el parámetro SRC
8154	El operando del parámetro SRC tiene el tipo de datos BOOL.

Código de error* (W#16#...)	Explicación
8281	Valor no válido en el parámetro COUNT
8382	El valor del parámetro SRC_INDEX se encuentra fuera de los valores límite de VARIANT.
8383	El valor del parámetro SRC_INDEX se encuentra fuera del límite superior del array.
8482	El valor del parámetro DEST_INDEX se encuentra fuera de los valores límite de VARIANT.
8483	El valor del parámetro DEST_INDEX se encuentra fuera del límite superior del array.
8534	El parámetro DEST está protegido contra escritura
8551	No es posible acceder al parámetro DEST.
8552	El operando del parámetro DEST no está tipificado.
8553	Error de creación de código en el parámetro DEST
8554	El operando del parámetro DEST tiene el tipo de datos BOOL.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MOVE_BLK_VARIANT
     (SRC := #SrcField
      COUNT := "Tag_Count"
      SRC_INDEX := "Tag_Src_Index"
      DEST_INDEX := "Tag_Dest_Index"
      DEST := #DestField
      RET_VAL := "Tag_Result"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
SRC	#SrcField	El operando local #SrcField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [0..10] of "MOVE_UDT"
COUNT	Tag_Count	2
SRC_INDEX	Tag_Src_Index	3

Parámetro	Operando	Valor
DEST_INDEX	Tag_Dest_Index	3
DEST	#DestField	El operando local #DestField utiliza un UDT todavía desconocido en el momento de programar el bloque. (Array [10..20] of "MOVE_UDT"

En el área de destino se copian 2 elementos del área de origen, empezando por el cuarto elemento del array de UDT. Las copias se insertan en el Array of UDT a partir del cuarto elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

UMOVE_BLK: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia sin interrupciones los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). El número de elementos que se copian en el área de destino se determina con el parámetro COUNT. El ancho del elemento del parámetro IN define el ancho de los elementos que deben copiarse.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

Copiar sin interrupciones

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Copiar área sin interrupciones".

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

GRAPH

```
CALL UMOVE_BLK <Tipo de datos>_<Tipo de datos>
     (IN := <Operando>
      COUNT := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1)]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de origen que se copia
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de elementos que se copian del área de origen al área de destino
OUT [1)]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Primer elemento del área de destino en la que se copian los contenidos del área de origen
[1)] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL UMOVE_BLK INT_UINT
     (IN := #a_array[2]
      COUNT := "Tag_Count"
      OUT => #b_array[1]
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción selecciona a partir del tercer elemento tres elementos INT de la variable #a_array y copia su contenido en la variable de salida #b_array, comenzando por el segundo elemento. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

FILL_BLK: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el valor del parámetro IN. El área de destino se rellena a partir de la dirección indicada en el parámetro OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor del parámetro IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL FILL_BLK <Tipo de datos>_<Tipo de datos>
     (IN := <Operando>
      COUNT := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Dirección del área de destino a partir de la cual se rellena
[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY. [2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL FILL_BLK INT_UINT
     (IN := #a_array[2]
      COUNT := "Tag_Count"
      OUT => #b_array[1]
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

UFILL_BLK: Rellenar área sin interrupciones

Descripción

La instrucción "Rellenar área sin interrupciones" rellena sin interrupciones una área de memoria (área de destino) con el valor del parámetro IN. El área de destino se rellena a partir de la dirección indicada en el parámetro OUT. El número de repeticiones de copia se determina mediante el parámetro COUNT. Al ejecutar la instrucción, se selecciona el valor del parámetro IN y se copia en el área de destino el número de veces especificado por el valor del parámetro COUNT.

Para poder ejecutar la instrucción, el área de origen y el área de destino deben ser del mismo tipo de datos.

Nota

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Por este motivo, los tiempos de reacción a alarmas de la CPU podrían aumentar al ejecutar la instrucción "Rellenar área sin interrupciones".

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- Se copian más datos de los que están disponibles en la entrada IN o en la salida OUT.

La instrucción "Rellenar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Sintaxis

Para la instrucción "Rellenar área sin interrupciones" se utiliza la siguiente sintaxis:

GRAPH

```
CALL UFILL_BLK <Tipo de datos>_<Tipo de datos>
      (IN := <Operando>
       COUNT := <Operando>
       OUT => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN [1]	Input	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	I, Q, M, D, L, P o constante	Elemento con el que se rellena el área de destino
COUNT	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L, P o constante	Número de repeticiones de copia
OUT [2]	Output	Números binarios, enteros, números en coma flotante, temporizadores, TOD, LTOD, DATE, CHAR	D, L	Dirección del área de destino a partir de la cual se rellena
[1] Los tipos de datos indicados también pueden utilizarse como elementos de una estructura ARRAY. [2] Los tipos de datos indicados solamente pueden utilizarse como elementos de una estructura ARRAY.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL UFILL_BLK INT_UINT
     (IN := #a_array[2]
      COUNT := "Tag_Count"
      OUT => #b_array[1]
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	a_array[2]	El operando "a_array" es del tipo de datos Array [0..5] of INT. Consta de 6 elementos del tipo de datos INT.
COUNT	Tag_Count	3
OUT	b_array[1]	El operando "b_array" es del tipo de datos Array [0..6] of INT. Consta de 7 elementos del tipo de datos INT.

La instrucción copia tres veces el tercer elemento (#a_array[2]) de la variable #a_array en la variable de salida #b_array. La operación de copia no debe ser interrumpida por otras actividades del sistema operativo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

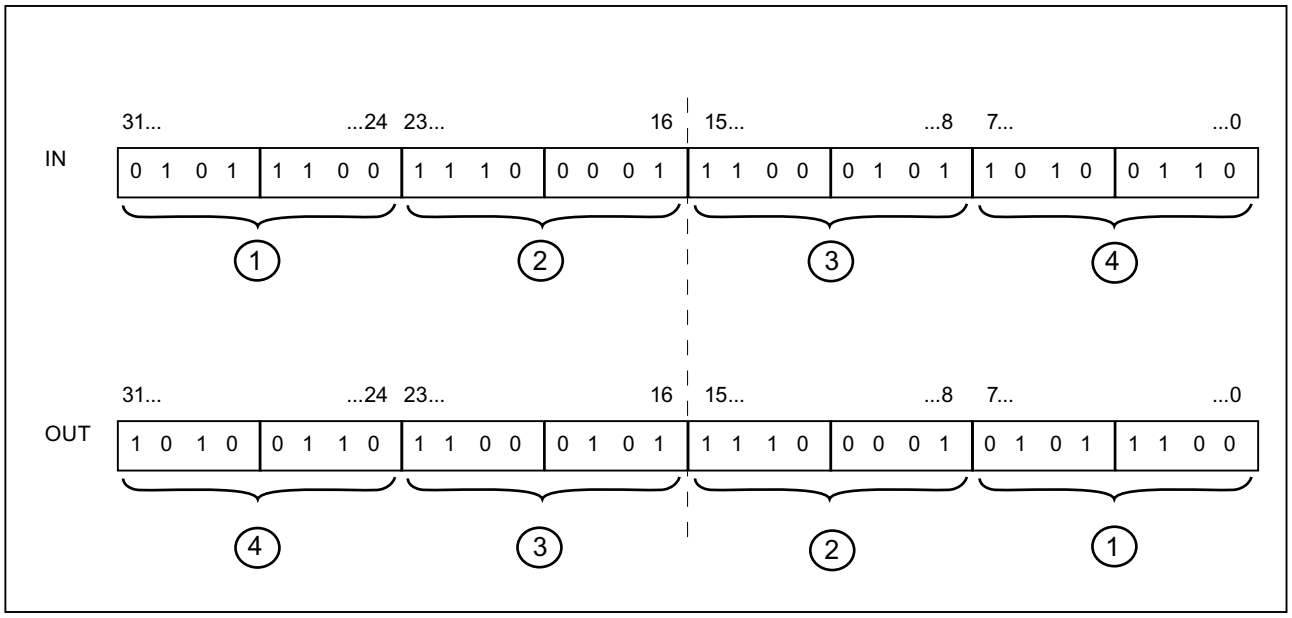
SWAP: Cambiar disposición

Descripción

Con la instrucción "Cambiar disposición" se modifica la disposición de los bytes del operando.

Dependiendo del tipo de datos utilizado, se puede cambiar el orden de los bytes en todo el acumulador 1 o solamente en la palabra derecha del acumulador 1.

La siguiente figura muestra cómo se cambian las posiciones de los bytes del operando del tipo de datos DWORD:



Disposición de los bytes en la palabra derecha del acumulador 1

Para el tipo de datos WORD se cambia la disposición de los bytes en la palabra derecha del acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor B	Valor D	Valor C

El resultado de la instrucción se guarda en la palabra derecha del acumulador 1. Los bytes de la palabra izquierda del acumulador 1 no se ven influidos por la instrucción y no cambian.

Sintaxis

Para la instrucción "Cambiar disposición" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SWAP(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cambiar disposición":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	WORD, DWORD, LWORD	I, Q, M, D, L o constante	Operando cuyas posiciones de bytes se cambian
<Resultado>	Output	WORD, DWORD, LWORD	I, Q, M, D, L	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SWAP("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_InValue	0000	1111	0000	1111
Tag_OutValue	0000	1111	1111	0000

Cambiar la disposición de los bytes en todo el acumulador 1:

Para el tipo de datos DWORD se modifica la disposición de los bytes en todo el acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor D	Valor C	Valor B	Valor A

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_InValue	1111	0000	0000	1111	0000	0000	1111	1111
Tag_OutValue	1111	1111	0000	0000	0000	1111	1111	0000

La disposición de los bytes en el operando "Tag_InValue" se guarda cambiada en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Instrucciones ARRAY-DB

ReadFromArrayDB: Leer en un bloque de datos ARRAY

Descripción

La instrucción "Leer en un bloque de datos ARRAY" permite leer datos en un DB de array y escribirlos en un área de destino.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Sintaxis

Para la instrucción "Leer en un bloque de datos ARRAY" se utiliza la siguiente sintaxis:

GRAPH

```
CALL ReadFromArrayDB
     (DB := <Operando>
      INDEX := <Operando>
      VALUE => <Operando>
```

GRAPH

```
RET_VAL => <Operando>
)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer desde bloque de datos de array":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
VALUE	Output	VARIANT	L	Valor que se lee y se devuelve
RET_VAL	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El DB de array contiene valores no válidos
8154	El tipo de datos no está declarado correctamente
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL ReadFromArrayDB
     (DB := "ArrayDB"
      INDEX := "ArrayDB".THIS[2]
      VALUE => "TargetField[10]".Data2[1]
      RET_VAL => "TagResult"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	TargetField[10].Data2[1]	El operando "TargetField" es del tipo de datos Array [10..20] of "MOVE_UDT".

El tercer elemento se lee en el "ArrayDB" y se escribe en el operando "TargetField[10].Data2[1]".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

WriteToArrayDB: Escribir en un bloque de datos ARRAY

Descripción

La instrucción "Escribir en un bloque de datos ARRAY" permite escribir datos en un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Sintaxis

Para la instrucción "Escribir en un bloque de datos ARRAY" se utiliza la siguiente sintaxis:

GRAPH

```
CALL WriteToArrayDB
     (DB := <Operando>
      INDEX := <Operando>
      VALUE := <Operando>)
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en un bloque de datos ARRAY":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	L	Valor que se escribe
RET_VAL	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código

Código de error* (W#16#...)	Explicación
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL WriteToArrayDB
     (DB := "ArrayDB"
      INDEX := "ArrayDB".THIS[2]
      VALUE := "SourceField[1]".Data1[6]
      RET_VAL => "TagResult"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceField[1].Data1[6]	El operando "SourceField" es del tipo de datos Array [0..10] of "MOVE_UDT".

Del operando "SourceField" se escribe el elemento "Data1[6]" del segundo elemento en el "ArrayDB". En el "ArrayDB" se escribe el tercer elemento.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

ReadFromArrayDBL: Leer en un bloque de datos ARRAY de la memoria de carga

Descripción

La instrucción "Leer en un bloque de datos ARRAY de la memoria de carga" permite leer datos desde un DB de array.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1" y el valor leído en el parámetro VALUE se emite dentro de dicho ciclo. En los demás ciclos del programa, el valor del parámetro VALUE no se modifica.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

Sintaxis

Para la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga" se utiliza la siguiente sintaxis:

GRAPH

```
CALL ReadFromArrayDBL, "DB de instancia"
     (REQ := <Operando>
      DB := <Operando>
      INDEX := <Operando>
      BUSY => <Operando>
      DONE => <Operando>
      ERROR => <Operando>
      VALUE := <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la lectura del DB de array
DB	Input	DB_ANY	I, Q, M, D, L	Bloque de datos en el que se lee
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento que se lee
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La lectura del DB de array todavía no ha finalizado
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
VALUE	InOut	VARIANT	D, L	Valor que se lee y se devuelve
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura

Código de error* (W#16#...)	Explicación
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL ReadFromArrayDBL, "ReadFromArrayDBL_DB"
     (REQ := "TagReg"
      DB := "ArrayDB"
      INDEX := "ArrayDB".THIS[2]
      BUSY => "TagBusy"
      DONE => "TagDone"
      ERROR => "TagError"
      VALUE := "SourceTargetField[1]".Data1[6]
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Leer en un bloque de datos ARRAY de la memoria de carga". El tercer elemento se lee en

el "ArrayDB" y se indica en el operando "SourceTargetField[1].Data1[6]". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE ya no se modifica. Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

WriteToArrayDBL: Escribir en un bloque de datos ARRAY de la memoria de carga

Descripción

La instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga" permite escribir datos en un DB de array de la memoria de carga.

Un DB de array es un bloque de datos que consta exactamente de un array de [tipo de datos]. Los elementos del array pueden ser del tipo de datos UDT o de cualquier otro tipo de datos simple. El contaje del array siempre comienza por el límite inferior "0", independientemente de la declaración posterior del array.

Si el DB de array se ha identificado con el atributo de bloque "Depositar solo en la memoria de carga", solo se deposita en la memoria de carga.

Si en el parámetro REQ se detecta un flanco de señal ascendente, se ejecuta la instrucción. El parámetro BUSY devuelve el estado lógico "1". Si en el parámetro BUSY se detecta un flanco de señal descendente, finaliza la instrucción y el valor del parámetro VALUE se escribe en el bloque de datos. Durante un ciclo del programa, el parámetro DONE tiene el estado lógico "1".

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Nota

El DB de array debe encontrarse en un bloque de datos con la propiedad de bloque "optimizado".

Sintaxis

Para la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga" se utiliza la siguiente sintaxis:

GRAPH

```
CALL WriteToArrayDBL, "DB de instancia"
     (REQ := <Operando>
      DB := <Operando>
      INDEX := <Operando>
      VALUE := <Operando>
      BUSY => <Operando>
      DONE => <Operando>
      ERROR => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = "1": Comenzar la escritura en el DB de array
DB	Input	DB_ANY	D	Bloque de datos en el que se escribe
INDEX	Input	DINT	I, Q, M, D, L, P o constante	Elemento del DB en el que se escribe
VALUE	Input	VARIANT	D, L	Valor que se escribe
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = "1": La escritura en el DB de array todavía no ha finalizado
DONE	Output	BOOL	I, Q, M, D, L	DONE = "1": La instrucción se ha ejecutado correctamente
ERROR	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro ERROR devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERROR

La tabla siguiente muestra el significado de los valores del parámetro ERROR:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B4	Los tipos de datos no coinciden
80B5	La operación de copia se ha cancelado
80C3	Demasiadas llamadas simultáneas
8132	El bloque de datos no existe o es demasiado corto
8133	El bloque de datos no se encuentra en la memoria de carga
8134	El bloque de datos está protegido contra escritura
8135	El bloque de datos no es un DB de array
8282	El valor del parámetro INDEX se encuentra fuera de los valores límite del array.
8350	El tipo de datos VARIANT no contiene ningún valor
8352	Error de creación de código
8353	Error de creación de código
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará la descripción de los códigos de error que son activados por las instrucciones "READ_DBL: Leer en un bloque de datos de la memoria de carga" y "WRIT_DBL: Escribir en un bloque de datos de la memoria de carga" en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL WriteToArrayDBL, "WriteToArrayDBL_DB"
      (REQ := "TagReg"
       DB := "ArrayDB"
       INDEX := "ArrayDB".THIS[2]
       VALUE := "SourceTargetField[1]".Data1[6]
       BUSY => "TagBusy"
       DONE => "TagDone"
       ERROR => "TagError"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
REQ	TagReq	BOOL
DB	ArrayDB	El operando "ArrayDB" es del tipo de datos Array [0..10] of INT.
INDEX	ArrayDB.THIS[2]	Tercer elemento del "ArrayDB"
VALUE	SourceTargetField[1].Data1[6]	El operando "SourceTargetField" es del tipo de datos Array [0..10] of "MOVE_UDT".
BUSY	TagBusy	BOOL
DONE	TagDone	BOOL

Si en el operando "TagReq" se registra un flanco de señal ascendente, se ejecuta la instrucción "Escribir en un bloque de datos ARRAY de la memoria de carga". En cuanto se detecta un flanco de señal descendente en el operando TagBusy, finaliza la instrucción y el valor del parámetro VALUE se escribe en el "ArrayDB" en el tercer elemento. Después de ejecutar la instrucción, el operando "TagDone" tiene el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Otras

BLKMOV: Copiar área

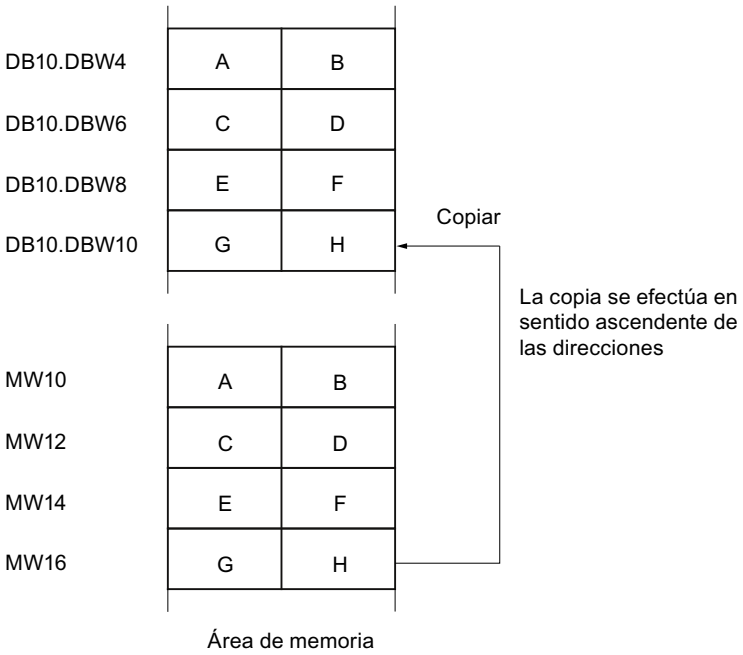
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

No hay limitaciones en cuanto a la profundidad de anidamiento.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL BLKMOV <Tipo de datos>
     (SRCBLK := <Operando>
      RET_VAL => <Operando>
      DSTBLK => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	VARIANT	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8092	El área de origen o destino está solo en la memoria de carga.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL BLKMOV VARIANT
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

La instrucción copia 10 bytes a partir de MB100 y los escribe en el DB1. Si ocurre un error durante la operación de copia, el respectivo código de error se indica en la variable "Tag_ErrorCode".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERR_ID: Consultar ID de error localmente (Página 4620)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante VARIANT.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la variable debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 16 KB como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

GRAPH

```
CALL UBLKMOV <Tipo de datos>
     (SRCBLK := <Operando>
      RET_VAL => <Operando>
      DSTBLK => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	VARIANT	L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	VARIANT	L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
8091	El área de origen o de destino se encuentra en un bloque de datos no relevante para la ejecución.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL UBLKMOV VARIANT
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

La instrucción copia 10 bytes a partir de MB100 y los escribe en el DB1. Si ocurre un error durante la operación de copia, el respectivo código de error se indica en la variable "Tag_ErrorCode".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERR_ID: Consultar ID de error localmente (Página 4620)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

FILL: Rellenar área

Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones.

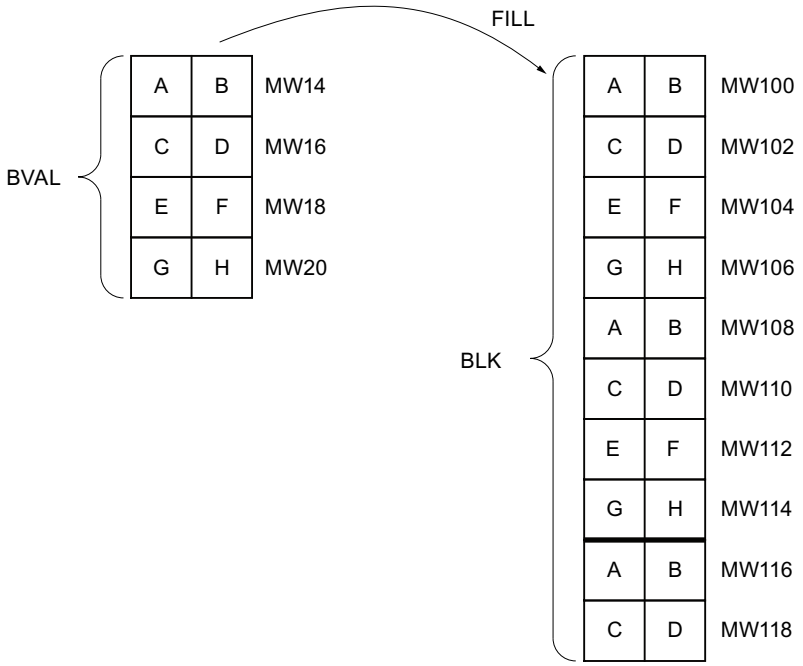
Las áreas de origen y destino se definen mediante VARIANT.

Nota

Las variables de la instrucción solo se pueden usar en los bloques de datos en los que no está ajustado el atributo "Acceso optimizado al bloque". Si la variable se ha declarado con el ajuste de remanencia "Ajustar en IDB", la variable también se puede usar "con acceso optimizado".

Para los bloques con el atributo "Acceso optimizado al bloque" puede utilizarse la instrucción "FILL_BLK: Rellenar área".

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: el contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, debe direccionarse de forma absoluta y la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Reglas para copiar estructuras

Al indicar una estructura a modo de parámetro de entrada, tenga en cuenta que la longitud de una estructura se adapta siempre a un número entero de bytes. Si se declara una estructura con un número impar de bytes, esta estructura necesita un byte adicional de espacio en memoria.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL FILL <Tipo de datos>
     (BVAL := <Operando>
      RET_VAL => <Operando>
      BLK => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BVAL	Input	VARIANT	L, P	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
BLK	Output	VARIANT	L, P	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8152	No se soportan los tipos de datos WSTRING, WCHAR y BOOL.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL FILL VARIANT
     (BVAL := P#M14.0 WORD 4
      RET_VAL => "Tag_ErrorCode"
      BLK => P#M100.0 WORD 10
      )
```

La instrucción copia el área de origen de MW14 a MW20 y rellena el área de destino de MW100 a MW118 con el contenido de las 4 palabras contenidas en el área de memoria del parámetro BVAL.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERR_ID: Consultar ID de error localmente (Página 4620)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" lee el contenido del operando y lo convierte según los tipos de datos parametrizados.

Encontrará información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Convertir valor" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ???_TO_???(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir valor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Secuencias de bits, enteros, números en coma flotante, CHAR, BCD16, BCD32	I, Q, M, D, L, P o constante	Valor que se convierte.
<Resultado>	Output	Secuencias de bits, enteros, números en coma flotante, CHAR, BCD16, BCD32	I, Q, M, D, L, P	Resultado de la conversión

En las listas desplegables "<???>" puede seleccionar los tipos de datos de la instrucción.

En el cuadro de la instrucción no se pueden seleccionar secuencias de bits (BYTE, WORD, DWORD, LWORD). Si se especifica un operando del tipo de datos BYTE, WORD, DWORD o LWORD en un parámetro de la instrucción, el operando se interpretará conforme al tipo de datos del parámetro de entrada o salida. El tipo de datos BYTE se interpreta entonces como USINT, WORD como UINT, DWORD como UDINT y LWORD como LINT.

Nota

Existe la posibilidad de seleccionar los tipos de datos DWORD y LWORD si se ha seleccionado como tipo de datos IN REAL o LREAL.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := INT_TO_DINT("Tag_InValue")
```

El contenido del operando "Tag_InValue" se lee y se convierte en un entero (32 bits). El resultado se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Conversión explícita (Página 2411)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor del operando al siguiente número entero. La instrucción interpreta el valor como número en coma flotante y lo convierte al número entero más próximo. Si en el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Redondear número" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := ROUND(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear número":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada que se redondea.
<Resultado>	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado del redondeo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := ROUND("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	2	-2

El número en coma flotante del operando "Tag_InValue" se redondea al número entero par más próximo y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor del operando al siguiente número entero superior. La instrucción interpreta el valor como número en coma flotante y lo convierte al siguiente número entero superior. El resultado puede ser mayor o igual que el valor de entrada.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero superior" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := CEIL(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero superior":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada como número en coma flotante
<Resultado>	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero superior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := CEIL("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	1	0

El número en coma flotante del operando "Tag_InValue" se redondea al siguiente número entero superior y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor del operando al siguiente número entero inferior. La instrucción interpreta el valor como número en coma flotante y lo convierte al siguiente número entero inferior. El resultado puede ser menor o igual que el valor de entrada.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero inferior" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := FLOOR(<Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero inferior":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada como número en coma flotante
<Resultado>	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero inferior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := FLOOR("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	0	-1

El número en coma flotante del operando "Tag_InValue" se redondea al siguiente número entero inferior y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un número entero a partir del valor del operando. El valor se interpreta como número en coma flotante. La instrucción selecciona solo la parte entera del número en coma flotante y la indica sin decimales como resultado.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Truncar a entero" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := TRUNC(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Truncar a entero":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Números en coma flotante	I, Q, M, D, L o constante	Valor de entrada como número en coma flotante
<Resultado>	Output	Enteros, números en coma flotante	I, Q, M, D, L	Resultado con la parte entera del número en coma flotante

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := TRUNC("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	1	-1

La parte entera del número en coma flotante del operando "Tag_InValue" se convierte en un número entero y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

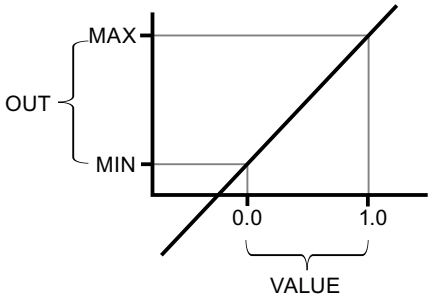
Principios básicos de GRAPH (Página 2825)

SCALE_X: Escalar

Descripción

La instrucción "Escalar" escala el valor del parámetro VALUE mapeándolo en un determinado rango de valores. Al ejecutar la instrucción, el número en coma flotante del parámetro VALUE se escala al rango de valores definido por los parámetros MIN y MAX. El resultado de la escala es un número entero que se deposita en la salida OUT.

La figura siguiente muestra un ejemplo de cómo pueden escalarse los valores:



La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [VALUE * (MAX – MIN)] + MIN

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del parámetro MIN es mayor o igual que el valor del parámetro MAX.
- El valor de un número en coma flotante indicado está fuera del rango de los números normalizados según IEEE-754.

- Ocurre un rebase por exceso.
- El valor del parámetro VALUE es NaN (Not a number = resultado de una operación aritmética no válida).

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SCALE_X <Tipo de datos>_<Tipo de datos>
     (MIN := <Operando>
      VALUE := <Operando>
      MAX := <Operando>
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE	Input	Números en coma flotante	I, Q, M, D, L o constante	Valor que se escala. Si se indica una constante, esta debe declararse.
MAX	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
RET_VAL	Output	Enteros, números en coma flotante	I, Q, M, D, L	Resultado de la escala

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SCALE_X INT_REAL
     (MIN := "Tag_Minimum"
      VALUE := "Tag_Value"
      MAX := "Tag_Maximum"
      RET_VAL => "Tag_OutputValue"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_Minimum	10
VALUE	Tag_Value	0.5
MAX	Tag_Maximum	30
RET_VAL	Tag_ReturnValue	20

El valor del parámetro "Tag_Value" se escala al rango de valores definido por los valores de los parámetros "Tag_Minimum" y "Tag_Maximum". El resultado se deposita en el parámetro "Tag_ReturnValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Declarar constantes globales (Página 2584)

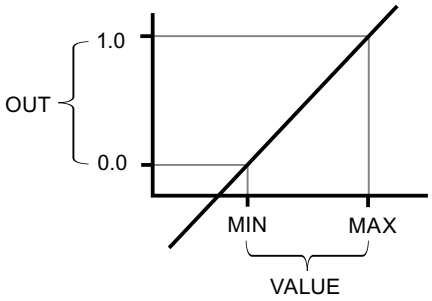
Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

NORM_X: Normalizar

Descripción

La instrucción "Normalizar" permite normalizar el valor del parámetro VALUE mapeándolo en una escala lineal. Los parámetros MIN y MAX sirven para definir los límites de un rango de valores que se refleja en la escala. En función de la posición del valor que se debe normalizar en este rango de valores, el resultado se calcula y se deposita como número en coma flotante en el parámetro OUT. Si el valor que se debe normalizar es igual al valor del parámetro MIN, el parámetro OUT devuelve el valor "0.0". Si el valor que se debe normalizar es igual al valor del parámetro MAX, el parámetro OUT devuelve el valor "1.0".

La figura siguiente muestra un ejemplo de cómo pueden normalizarse los valores:



La instrucción "Normalizar" utiliza la siguiente ecuación:

OUT = (VALUE – MIN) / (MAX – MIN)

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del parámetro MIN es mayor o igual que el valor del parámetro MAX.
- El valor de un número en coma flotante indicado está fuera del rango de los números normalizados según IEEE-754.
- Ocurre un rebase por exceso.
- El valor del parámetro VALUE es NaN (Not a number = resultado de una operación aritmética no válida).

Nota

Encontrará más información sobre la conversión de valores analógicos en el manual de producto correspondiente.

Sintaxis

Para la instrucción "Normalizar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL NORM_X <Tipo de datos>_<Tipo de datos>
     (MIN := <Operando>
      VALUE := <Operando>
      MAX := <Operando>
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Normalizar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MIN [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite inferior del rango de valores
VALUE [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Valor que se normaliza
MAX [1)]	Input	Enteros, números en coma flotante	I, Q, M, D, L o constante	Límite superior del rango de valores
RET_VAL	Output	Números en coma flotante	I, Q, M, D, L	Resultado de la normalización
[1)] Si se utilizan constantes en estos tres parámetros, basta con declarar una de ellas.				

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Encontrará más información sobre la declaración de constantes en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL NORM_X INT_REAL
     (MIN := "Tag_Minimum"
      VALUE := "Tag_Value"
      MAX := "Tag_Maximum"
      RET_VAL => "Tag_OutputValue"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MIN	Tag_Minimum	10
VALUE	Tag_Value	20
MAX	Tag_Maximum	30
RET_VAL	Tag_ReturnValue	0.5

El valor del parámetro "Tag_Value" se asigna al rango de valores definido por los valores de los parámetros "Tag_Minimum" y "Tag_Maximum". El valor de la variable del parámetro "Tag_Value" se normaliza conforme al rango de valores definido. El resultado se deposita como número en coma flotante en el parámetro "Tag_ReturnValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Declarar constantes globales (Página 2584)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Otras

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": Se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27 648 y 27 648. En este caso, la constante "K1" tiene el valor "-27 648,0", y la constante "K2", el valor "+27 648,0".
- Estado lógico "0": Se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27 648. En este caso, la constante "K1" tiene el valor "0,0", y la constante "K2", el valor "+27 648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SCALE
      (IN := <Operando>
       HI_LIM := <Operando>
       LO_LIM := <Operando>
       BIPOLAR := <Operando>
       RET_VAL => <Operando>
       OUT => <Operando>
```

GRAPH

```
)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala.
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1".
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SCALE
```

GRAPH

```
(IN := "Tag_InputValue"
 HI_LIM := "Tag_HighLimit"
 LO_LIM := "Tag_LowLimit"
 BIPOLAR := "Tag_Bipolar"
 RET_VAL => "Tag_ErrorCode"
 OUT => "Tag_OutputValue"
 )
```

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

En el operando "Tag_InputValue" se indica el valor que debe convertirse y escalarse. El valor límite superior y el inferior se definen mediante los operandos "Tag_HighLimit" y "Tag_LowLimit". Con ayuda del operando "Tag_Bipolar" = TRUE se indica que el valor del parámetro IN debe interpretarse como bipolar. En el operando "Tag_OutputValue" se indica el resultado de la instrucción.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERR_ID: Consultar ID de error localmente (Página 4620)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" permite desescalar el número en coma flotante del parámetro IN en unidades físicas comprendidas entre un límite inferior y uno superior y convertirlo en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": Se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27 648 y 27 648. En este caso, la constante "K1" tiene el valor "-27 648,0", y la constante "K2", el valor "+27 648,0".
- Estado lógico "0": Se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27 648. En este caso, la constante "K1" tiene el valor "0,0", y la constante "K2", el valor "+27 648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Sintaxis

Para la instrucción "Desescalar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL UNSCALE
     (IN := <Operando>
      HI_LIM := <Operando>
      LO_LIM := <Operando>
      BIPOLAR := <Operando>
      RET_VAL => <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero.
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL UNSCALE
      (IN := "Tag_InputValue"
       HI_LIM := "Tag_HighLimit"
```

GRAPH

```
LO_LIM := "Tag_LowLimit"
BIPOLAR := "Tag_Bipolar"
RET_VAL => "Tag_ErrorCode"
OUT => "Tag_OutputValue"
)
```

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

En el operando "Tag_InputValue" se indica el valor que debe convertirse y desescalarse. El valor límite superior y el inferior se definen mediante los operandos "Tag_HighLimit" y "Tag_LowLimit". Con ayuda del operando "Tag_Bipolar" = TRUE se indica que el valor del parámetro IN debe interpretarse como bipolar. En el operando "Tag_OutputValue" se indica el resultado de la instrucción.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERR_ID: Consultar ID de error localmente (Página 4620)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Control del programa

Control en runtime

ENDIS_PW: Limitar y habilitar legitimación de la contraseña

Descripción

La instrucción "Limitar y habilitar legitimación de la contraseña" permite definir si las contraseñas configuradas para la CPU están legitimadas o no. Con ello, se pueden impedir conexiones legitimadas, aun conociendo la contraseña correcta.

Si llama la instrucción y el parámetro REQ tiene el estado lógico "0", en los parámetros de salida se muestra el estado ajustado en ese momento. Si se han realizado cambios en los parámetros de entrada, estos no se transfieren a los parámetros de salida.

Si llama la instrucción y el parámetro REQ tiene el estado lógico "1", se aplica el estado lógico de los parámetros de entrada (F_PWD, FULL_PWD, R_PWD, HMI_PWD). FALSE significa que la legitimación por contraseña no está permitida y TRUE significa que la contraseña puede utilizarse.

Es posible permitir o prohibir individualmente el bloqueo o la habilitación de las contraseñas. Por ejemplo, pueden prohibirse todas las contraseñas excepto la contraseña de seguridad. Esto permite limitar las posibilidades de acceso a un reducido grupo de usuarios. Los parámetros de salida (F_PWD_ON, FULL_PWD_ON, R_PWD_ON, HMI_PWD_ON) siempre muestran el estado actual del uso de contraseñas, independientemente del parámetro REQ.

Las contraseñas no configuradas deben tener en la entrada el estado lógico TRUE y devolver el estado lógico TRUE en la salida. La contraseña de seguridad positiva (fail-safe) solo se puede parametrizar para una CPU F y por eso en una CPU estándar debe interconectarse siempre con el estado lógico TRUE. Si la instrucción devuelve un error, la llamada permanece sin efecto, es decir, el bloqueo anterior sigue en vigor.

Las contraseñas bloqueadas pueden volver a habilitarse en las condiciones siguientes:

- Se ha restablecido la configuración de fábrica de la CPU.
- El panel frontal de la CPU S7-1500 soporta un cuadro de diálogo que permite navegar al menú adecuado y volver a habilitar las contraseñas.
- Cuando se llama la instrucción "Limitar y habilitar legitimación de la contraseña", el parámetro de entrada de la contraseña deseada tiene el estado lógico "1".
- Ponga el selector de modo a STOP. La restricción de la legitimación de contraseña se activa de nuevo en cuanto se mueve el selector nuevamente a RUN.
- La inserción de una tarjeta de memoria vacía (tarjeta de transferencia o tarjeta de programa) en una CPU S7-1200.
- La transición de POWER OFF-POWER ON desactiva la protección en la CPU S7-1200. La instrucción "Limitar y habilitar legitimación de la contraseña" debe volverse a llamar en el programa (por ejemplo en el OB de arranque).

Nota

Si la contraseña HMI no está habilitada, la instrucción "Limitar y habilitar legitimación de la contraseña" bloquea el acceso de los sistemas HMI.

Nota

Las conexiones ya existentes y legitimadas conservan sus derechos de acceso y no pueden limitarse con la instrucción "Limitar y habilitar legitimación de la contraseña".

Impedir el bloqueo no intencionado en una CPU S7-1500

Los ajustes pueden realizarse en el panel frontal de la CPU, que guarda el último ajuste en cada caso.

Para impedir un bloqueo no intencionado, en una CPU S7-1500 es posible desactivar la protección moviendo el selector de modo a STOP. La protección se activa automáticamente después de mover el selector de modo a RUN sin que tenga que volver a llamarse la instrucción "Limitar y habilitar legitimación de la contraseña" o deban realizarse otras acciones en el panel frontal.

Impedir el bloqueo no intencionado en una CPU S7-1200

Como la CPU S7-1200 no dispone de selector de modo, la protección se desactiva con POWER OFF-POWER ON. De este modo es posible y recomendable impedir el bloqueo no intencionado con ayuda de determinadas secuencias en el programa.

Para ello, programe un control de tiempo, ya sea mediante un OB de alarma cíclica o un temporizador en el Main OB (OB 1). De este modo, tras una transición de POWER OFF-POWER ON y la consiguiente desactivación de la protección, se puede volver a llamar rápidamente la instrucción "Limitar y habilitar legitimación de la contraseña" en el OB correspondiente (por ejemplo OB 1 u OB 35). Para que la ventana temporal en la que la instrucción no está activa y en la que por tanto no existen limitaciones en la legitimación de la contraseña sea lo más pequeña posible, llame la instrucción en el OB de arranque (OB 100). Este procedimiento ofrece la mayor protección posible contra accesos no autorizados.

Si se ha producido un bloqueo no intencionado puede omitirse la llamada en el OB de arranque (por ejemplo mediante la consulta de un parámetro de entrada) y se dispondrá del tiempo configurado (por ejemplo de 10 segundos a 1 minuto) para establecer una conexión con la CPU antes de que el bloqueo vuelva a activarse.

Si no se ha previsto ningún temporizador en el código del programa y se produce un bloqueo, inserte en la CPU una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna de la CPU. Después deberá cargarse nuevamente el programa de usuario de STEP 7 en la CPU.

Procedimiento en caso de pérdida de contraseña en una CPU S7-1200

Si ha perdido la contraseña de una CPU S7-1200 protegida por contraseña, borre el programa protegido por contraseña con una tarjeta de transferencia o una tarjeta de programa vacía. La tarjeta de transferencia o la tarjeta de programa vacía borra la memoria de carga interna

de la CPU. A continuación, puede transferir un nuevo programa de usuario de STEP 7 Basic a la CPU.

⚠ ADVERTENCIA

Inserción de una tarjeta de transferencia vacía

Cuando se inserta una tarjeta de transferencia en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

El contenido de la tarjeta de transferencia se encuentra en la memoria de carga interna después de su extracción. Compruebe aquí que la tarjeta no contenga ningún programa.

⚠ ADVERTENCIA

Inserción de una tarjeta de programa vacía

Cuando se inserta una tarjeta de programa en una CPU en funcionamiento, la CPU pasa a STOP. En caso de estados operativos no seguros, los controladores pueden fallar y, con ello, provocar un funcionamiento descontrolado de los dispositivos que se controlan. El resultado es un funcionamiento impredecible del sistema de automatización, que puede provocar lesiones mortales o graves, así como daños materiales.

Asegúrese de que la tarjeta de programa esté vacía. La memoria de carga interna se copiará en la tarjeta de programa vacía. Después de retirar la tarjeta de programa previamente vacía, la memoria de carga interna quedará vacía.

Debe retirarse la tarjeta de transferencia o la tarjeta de programa antes de poner la CPU en RUN.

Efectos del uso de contraseñas en los modos de operación

La siguiente tabla muestra qué efectos tiene el uso de contraseñas en los modos de operación y en las respectivas acciones del usuario mediante la instrucción "Limitar y habilitar legitimación de la contraseña".

Acción	Protección por contraseña mediante la instrucción
Estado básico después de • Selector de modo en STOP • Restablecimiento manual de la memoria (PG, interruptor, cambio de MC (Motion Control)) • Restablecimiento de la configuración de fábrica	No activado (sin limitaciones)
Estado básico tras conexión (POWER ON)	• CPU S7-1200: El bloqueo está desactivado y la instrucción tiene que volverse a ejecutar en el programa (por ejemplo en el OB de arranque). • CPU S7-1500: Activada (si antes de la desconexión (POWER OFF) había un bloqueo activo) La posibilidad de no permitir contraseñas es remanente.
Transición del estado operativo RUN/ARRANQUE/PARADA -> STOP (mediante finalización de la instrucción, un error o comunicación) o STOP -> ARRANQUE/RUN/PARADA	Activado Las contraseñas siguen sin poder utilizarse.

Sintaxis

Para la instrucción "Limitar y habilitar legitimación de la contraseña" se utiliza la siguiente sintaxis:

GRAPH

```
CALL ENDIS_PW
     (REQ := <Operando>
      F_PWD := <Operando>
      FULL_PWD := <Operando>
      R_PWD := <Operando>
      HMI_PWD := <Operando>
      F_PWD_ON => <Operando>
      FULL_PWD_ON => <Operando>
      R_PWD_ON => <Operando>
      HMI_PWD_ON => <Operando>
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Limitar y habilitar legitimación de la contraseña":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Si el parámetro REQ tiene el estado lógico "0", se consulta el estado lógico ajustado actualmente para las contraseñas.
F_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura incluido failsafe • F_PWD = "0": No permitir contraseña • F_PWD = "1": Permitir contraseña
FULL_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura/escritura • FULL_PWD = "0": No permitir contraseña • FULL_PWD = "1": Permitir contraseña
R_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso de lectura • R_PWD = "0": No permitir contraseña • R_PWD = "1": Permitir contraseña
HMI_PWD	Input	BOOL	I, Q, M, D, L o constante	Acceso a la HMI • HMI_PWD = "0": No permitir contraseña • HMI_PWD = "1": Permitir contraseña
F_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura incluido failsafe • F_PWD_ON = "0": Contraseña no permitida • F_PWD_ON = "1": Contraseña permitida
FULL_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura/escritura • FULL_PWD_ON = "0": Contraseña no permitida • FULL_PWD_ON = "1": Contraseña permitida
R_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso de lectura • R_PWD_ON = "0": Contraseña no permitida • R_PWD_ON = "1": Contraseña permitida
HMI_PWD_ON	Output	BOOL	I, Q, M, D, L	Estado del acceso a la HMI • HMI_PWD_ON = "0": Contraseña no permitida • HMI_PWD_ON = "1": Contraseña permitida
RET_VAL	Output	WORD	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8090	No se soporta la instrucción "Limitar y habilitar legitimación de la contraseña".

Código de error* (W#16#...)	Explicación
80D0	La contraseña de seguridad (fail-safe) no está configurada. En las CPU estándar, el estado lógico tiene que ser TRUE.
80D1	El acceso de lectura/escritura no está configurado
80D2	El acceso de lectura no está configurado
80D3	El acceso HMI no está configurado
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del tiempo de ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar en todos los bloques independientemente de la prioridad.

Si se llama la instrucción en un bloque de prioridad superior, p. ej, en una alarma de proceso o de diagnóstico o en una alarma cíclica, la instrucción no se ejecuta.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" se puede llamar hasta 10 veces en un ciclo de programa.

Sintaxis

Para la instrucción "Reiniciar tiempo de vigilancia del ciclo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL RE_TRIGR ()
```

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Sintaxis

Para la instrucción "Finalizar programa" se utiliza la siguiente sintaxis:

GRAPH

```
CALL STP ()
```

Parámetros

La instrucción "Finalizar programa" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERROR: Consultar error localmente

Descripción

La instrucción "Consultar error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si el sistema notifica errores mientras se ejecuta el bloque, en el operando de la salida OUT se guarda información detallada acerca del primer error de ejecución ocurrido desde la última vez que se ejecutó la instrucción.

En la salida OUT solo pueden especificarse operandos del tipo de datos de sistema "ErrorStruct". El tipo de datos de sistema "ErrorStruct" especifica la estructura exacta con la que debe guardarse la información acerca del error que ha ocurrido. Con ayuda de otras instrucciones se puede evaluar esta estructura y programar la reacción correspondiente. Si

se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces emite información sobre el siguiente error ocurrido.

Nota

La salida OUT solo se modifica si hay una información de error. Para volver a poner la salida a "0" tras tratar un error existen las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz de bloque.
- Ponga la variable de nuevo a "0" antes de llamar la instrucción.

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar error localmente" activa el tratamiento local de errores en el bloque. Si en la lógica de un bloque se ha insertado "Consultar error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Sintaxis

Para la instrucción "Consultar error localmente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL GET_ERROR
     (OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	ErrorStruct	D, L	Información de error

Tipo de datos "ErrorStruct"

La tabla siguiente muestra la estructura del tipo de datos "ErrorStruct":

Componente de estructura		Tipo de datos	Descripción
ERROR_ID		WORD	ID de error
FLAGS		BYTE	Muestra si el error se ha producido durante la llamada de un bloque. 16#01: Error durante la llamada de un bloque 16#00: No hay errores durante la llamada de un bloque
REACTION		BYTE	Reacción predeterminada: 0: ignorar (error de escritura), 1: continuar con el valor sustitutivo "0" (error de lectura), 2: omitir instrucción (error de sistema)
CODE_ADDRESS		CREF	Información sobre la dirección y el tipo del bloque
	BLOCK_TYPE	BYTE	Tipo de bloque en el que se ha producido el error: 1: OB 2: FC 3: FB
	CB_NUMBER	UINT	Número del bloque lógico
	OFFSET	UDINT	Referencia a la memoria interna
MODE		BYTE	Información sobre la dirección de un operando
OPERAND_NUMBER		UINT	Número de operando del comando de la máquina
POINTER_NUMBER_LOCATION		UINT	(A) Puntero interno
SLOT_NUMBER_SCOPE		UINT	(B) Área de almacenamiento en la memoria interna
DATA_ADDRESS		NREF	Información sobre la dirección de un operando
	AREA	BYTE	(C) Área de memoria: L: 16#40 – 4E, 86, 87, 8E, 8F, C0 – CE I: 16#81 Q: 16#82 M: 16#83 DB: 16#84, 85, 8A, 8B Infracciones de área en una variable editable directamente del tipo de datos DINT: 16#04
	DB_NUMBER	UINT	(D) Número del bloque de datos
	OFFSET	UDINT	(E) Dirección relativa del operando

Componente de estructura "ERROR_ID"

La tabla siguiente muestra los valores que pueden devolverse en el componente de estructura "ERROR_ID":

ID* (hexadecimal)	ID* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2534	9524	Error de número de bloque FC
2535	9525	Error de número de bloque FB
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Consultar y eliminar errores en el código del programa (Página 265)

Tratar errores de ejecución del programa (Página 2932)

GET_ERR_ID: Consultar ID de error localmente

Descripción

La instrucción "Consultar ID de error localmente" consulta si se han producido errores en un bloque. Suele tratarse de un error de acceso. Si al ejecutar el bloque el sistema notifica errores de ejecución desde que se ejecutó la instrucción por última vez, la ID de error del primer error ocurrido se guarda en la variable de la salida RET_VAL.

En la salida RET_VAL solo pueden especificarse operandos del tipo de datos "WORD". Si se producen varios errores en el bloque, la instrucción soluciona el primer error ocurrido, y solo entonces la instrucción emite la ID de error del siguiente error ocurrido.

Nota

La salida RET_VAL solo se modifica si hay una información de error. Para volver a poner la salida a "0" tras tratar un error existen las siguientes posibilidades:

- Declare la variable en la sección "Temp" de la interfaz de bloque.
- Ponga la variable de nuevo a "0" antes de llamar la instrucción.

La salida de la instrucción "Consultar ID de error localmente" se activa únicamente si existe una información de error. Si no se cumple esta condición, el procesamiento ulterior del programa no se ve afectado por la instrucción "Consultar ID de error localmente".

Encontrará un ejemplo sobre cómo utilizar la instrucción en combinación con otras posibilidades de tratamiento de errores, en "Consulte también".

Nota

La instrucción "Consultar ID de error localmente" activa el tratamiento local de errores dentro del bloque. Si en la lógica de un bloque se ha insertado la instrucción "Consultar ID de error localmente", se ignoran las reacciones predeterminadas del sistema al ocurrir errores.

Sintaxis

Para la instrucción "Consultar ID de error localmente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL GET_ERR_ID
```

GRAPH

```
(RET_VAL => <Operando>
 )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar ID de error localmente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	WORD	I, Q, M, D, L	ID de error

Parámetro RET_VAL

La tabla siguiente muestra los valores que pueden depositarse en el parámetro RET_VAL:

Código de error* (hexadecimal)	Código de error* (decimal)	Descripción
0	0	Ningún error
2503	9475	Puntero no válido
2520	9504	STRING no válido
2522	9506	Error de lectura: operando fuera del rango válido
2523	9507	Error de escritura: operando fuera del rango válido
2524	9508	Error de lectura: operando no válido
2525	9509	Error de escritura: operando no válido
2528	9512	Error de lectura: alineación de datos
2529	9513	Error de escritura: alineación de datos
252C	9516	Puntero no válido
2530	9520	Error de escritura: bloque de datos
2533	9523	Puntero usado no válido
2534	9524	Error de número de bloque FC
2535	9525	Error de número de bloque FB
2538	9528	Error de acceso: el DB no existe
2539	9529	Error de acceso: se ha utilizado un DB incorrecto
253A	9530	El bloque de datos global no existe
253C	9532	Indicación errónea o la función no existe
253D	9533	La función de sistema no existe
253E	9534	Indicación errónea o el bloque de función no existe
253F	9535	El bloque de sistema no existe
2550	9552	Error de acceso: el DB no existe
2551	9553	Error de acceso: se ha utilizado un DB incorrecto
2575	9589	Error en la profundidad de anidamiento del programa
2576	9590	Error en la distribución de datos locales

Código de error* (hexadecimal)	Código de error* (decimal)	Descripción
2577	9591	La propiedad de bloque "Alimentación de parámetros a través de registros" no está activada.
25A0	9632	Error interno en TP
25A1	9633	Variable protegida contra escritura
25A2	9634	Valor numérico no válido de la variable
2942	10562	Error de lectura: entrada
2943	10563	Error de escritura: salida
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Consultar y eliminar errores en el código del programa (Página 265)

Tratar errores de ejecución del programa (Página 2932)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

INIT_RD: Inicializar todos los datos remanentes

Descripción

La instrucción "Inicializar todos los datos remanentes" permite inicializar todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC al mismo tiempo. La instrucción solo puede ejecutarse dentro de un OB de arranque, puesto que su ejecución excedería el tiempo de ciclo de programa.

Sintaxis

Para la instrucción "Inicializar todos los datos remanentes" se utiliza la siguiente sintaxis:

GRAPH

```
CALL INIT_RD
     (REQ := <Operando>
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar todos los datos remanentes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Si la entrada REQ devuelve el estado lógico "1", se inicializan todos los datos remanentes.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
80B5	No puede ejecutarse la instrucción porque no ha sido programada dentro de un OB de arranque.
Información de error general	Consulte también: "GET_ERR_ID: Consultar ID de error localmente"
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL INIT_RD
     (REQ := "Tag_REQ"
      RET_VAL => "Tag_RET_VAL"
      )
```

Si el operando "Tag_REQ" devuelve el estado lógico "1", se ejecuta la instrucción. Se inicializan todos los datos remanentes de todos los bloques de datos, marcas y temporizadores y contadores SIMATIC.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

GET_ERR_ID: Consultar ID de error localmente (Página 4620)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" detiene la ejecución del programa durante un periodo parametrizado. El periodo se indica en microsegundos en el parámetro WT.

Es posible parametrizar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve la información de error.

Sintaxis

Para la instrucción "Programar retardo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL WAIT
     (WT := <Operando>
     )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Operaciones lógicas con palabras

NOT (INV): Invertir

Descripción

La instrucción "Invertir" permite invertir el estado lógico de los bits del operando. Al ejecutar la instrucción, el valor del operando se combina con una operación lógica O-exclusiva utilizando una plantilla hexadecimal (W#16#FFFF para números de 16 bits o DW#16#FFFF FFFF para números de 32 bits). Esto invierte el estado lógico de los distintos bits y lo devuelve como resultado.

Sintaxis

Para la instrucción "Invertir" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := NOT(<Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Invertir":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Secuencias de bits, enteros	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L, P	Complemento a 1 del valor del operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"TagOut_Value" := NOT("TagIn_Value")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
TagIn_Value	W#16#000F
TagOut_Value	W#16#FFF0

La instrucción invierte el estado lógico de los distintos bits del operando "TagIn_Value" y escribe el resultado en el operando "TagOut_Value".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

DECO: Descodificar

Descripción

La instrucción "Descodificar" activa un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor del parámetro IN y activa el bit del parámetro OUT cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor del parámetro IN es mayor que 31, se ejecuta una instrucción modulo 32.

Sintaxis

Para la instrucción "Descodificar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DECO UINT_<Tipo de datos>
      (IN := <Operando>
       OUT => <Operando>
      )
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	UINT	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida.
OUT	Output	Secuencias de bits	I, Q, M, D, L, P	Valor de salida

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

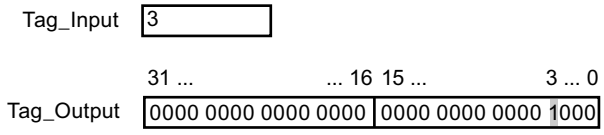
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL DECO UINT_DWORD
     (IN := "Tag_Input"
      OUT => "Tag_Output"
     )
```

La figura siguiente muestra el funcionamiento de la instrucción con valores concretos:



La instrucción lee el número de bit "3" del valor del operando "Tag_Input" de la entrada y activa el tercer bit del valor del operando "Tag_Output" de la salida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ENCO: Codificar

Descripción

La instrucción "Codificar" lee el número del bit menos significativo que está activado en el valor de entrada y devolverlo en el parámetro OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor del parámetro IN y escribe su número de bit en el operando del parámetro OUT. Si el parámetro IN devuelve el valor DW#16#00000001 o DW#16#00000000, en el parámetro OUT se deposita el valor "0".

Sintaxis

Para la instrucción "Codificar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL ENCO <Tipo de datos>
     (IN := <Operando>
      OUT => <Operando>
     )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

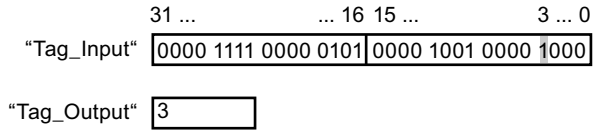
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
CALL ENCO DWORD
     (IN := "Tag_Input"
      OUT => "Tag_Output"
     )
```

La figura siguiente muestra el funcionamiento de la instrucción con valores concretos:



La instrucción selecciona el bit menos significativo que está activado en la variable "Tag_Input" y escribe su bit "3" en la variable "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite elegir, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copiar su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G tiene el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

Para poder procesar la instrucción, las variables de todos los parámetros deben ser del mismo tipo de datos.

Sintaxis

Para la instrucción "Seleccionar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SEL <Tipo de datos>
     (G := <Operando>
      IN0 := <Operando>
      IN1 := <Operando>
      OUT => <Operando>
     )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
G	Input	BOOL	I, Q, M, D, L o constante	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, LTOD, DATE, LDT	I, Q, M, D, L, P o constante	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, TIME, CHAR, TOD, DATE, LDT	I, Q, M, D, L, P o constante	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, TIME, CHAR, TOD, DATE, LDT	I, Q, M, D, L, P	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SEL CHAR
     (G := "Tag_Input_G"
      IN0 := "Tag_Input0"
```

GRAPH

```
     IN1 := "Tag_Input1"
     OUT => "Tag_Output"
    )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor
G	Tag_Input_G	1
IN0	Tag_Input0	W#16#0000
IN1	Tag_Input1	W#16#FFFF
OUT	Tag_Output	W#16#FFFF

Dependiendo del estado lógico de la entrada "Tag_Input_G", se selecciona el valor de la entrada "Tag_Input0" o "Tag_Input1" y se copia en la salida "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

MUX: Multiplexar

Descripción

La instrucción "Multiplexar" permite copiar el contenido de una entrada seleccionada en la salida RET_VAL. El número de entradas seleccionables se puede ampliar hasta un máximo de 32. La numeración comienza por IN0 y continúa en orden ascendente con cada nueva entrada. El parámetro K determina la entrada cuyo contenido se copia en la salida RET_VAL. Si el valor del parámetro K es mayor que el número de entradas disponibles, el contenido del parámetro INELSE se copia en la salida RET_VAL.

La instrucción "Multiplexar" solo se puede ejecutar si las variables de todas las entradas y de la salida RET_VAL tienen el mismo tipo de datos. El parámetro K es una excepción, ya que para éste sólo pueden indicarse números enteros.

El valor del parámetro RET_VAL no es válido si se cumple una de las siguientes condiciones:

- El valor del parámetro K es mayor que el número de entradas disponibles.
- Se producen errores al ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplexar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MUX <Tipo de datos>_<Tipo de datos>
```

GRAPH

```
(K := <Operando>
(IN0 := <Operando>
(IN1 := <Operando>
(INn := <Operando>
(INELSE := <Operando>
(RET_VAL := <Operando>
)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplexar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
K	Input	Enteros	I, Q, M, D, L o constante	Indica la entrada cuyo contenido se copia.
IN0	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L o constante	Primer valor de entrada
IN1	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L o constante	Segundo valor de entrada
INn	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L o constante	Valores de entrada opcionales
INELSE	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, LDT	I, Q, M, D, L o constante	Indica el valor que se copia si K > n.
RET_VAL	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT	I, Q, M, D, L	Salida en la que se copia el valor.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MUX SINT_SINT
      (K := "Tag_Number"
      (IN0 := "Tag_Value_1"
      (IN1 := "Tag_Value_2"
      (INELSE := "Tag_Value_3"
      (RET_VAL := "Tag_Result"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
K	Tag_Number	1
IN0	Tag_Value_1	DW#16#00000000
IN1	Tag_Value_2	DW#16#3E4A7D
INELSE	Tag_Value_3	DW#16#FFFF0000
RET_VAL	Tag_Result	DW#16#3E4A7D

De acuerdo con el valor del operando "Tag_Number", se copia el valor de la entrada "Tag_Value_1" y se indica en el operando de la salida "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

DEMUX: Desmultiplexar

Descripción

La instrucción "Desmultiplexar" permite copiar el contenido de la entrada IN en una salida seleccionada. El número de salidas seleccionables se puede ampliar hasta un máximo de 32. La numeración comienza por OUT0 y continúa en orden ascendente con cada nueva salida. El parámetro K permite determinar la salida en la que se copia el contenido de la entrada IN. Las demás salidas no cambian. Si el valor del parámetro K es mayor que el número de salidas disponibles, el contenido de la entrada IN se copia en el parámetro OUTELSE.

La instrucción "Desmultiplexar" solo se puede ejecutar si las variables de la entrada IN y las de todas las salidas tienen el mismo tipo de datos. El parámetro K es una excepción, ya que en el mismo sólo pueden indicarse números enteros.

El valor de la salida OUTELSE no es válido si se cumple una de las siguientes condiciones:

- El valor del parámetro K es mayor que el número de salidas disponibles.
- Se producen errores al ejecutar la instrucción.

Sintaxis

Para la instrucción "Desmultiplexar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DEMUX <Tipo de datos>_<Tipo de datos>
     (K := <Operando>
     (IN := <Operando>
     (OUT0 := <Operando>
     (OUT1 := <Operando>
     (OUTn := <Operando>
     (OUTELSE := <Operando>
     )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desmultiplexar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
K	Input	Enteros	I, Q, M, D, L o constante	Indica la salida en la que se copia el valor de entrada (IN).
IN	Input	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	I, Q, M, D, L o constante	Valor de entrada
OUT0	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	I, Q, M, D, L	Primera salida

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT1	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	I, Q, M, D, L	Segunda salida
OUTn	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	I, Q, M, D, L	Salidas opcionales
OUTELSE	Output	Números binarios, enteros, números en coma flotante, secuencias de caracteres, temporizadores, TOD, LTOD, DATE, DT, LDT	I, Q, M, D, L	Salida en la que se copia el valor de entrada (IN) con K > n.

Encontrará más información sobre los tipos de datos disponibles en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL DEMUX SINT_SINT
      (K := "Tag_Number"
      (IN := "Tag_Value"
      (OUT0 := "Tag_Output_1"
      (OUT1 := "Tag_Output_2"
      (OUTELSE := "Tag_Output_3"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores de operandos concretos:

Tabla 9-34 Valores de entrada de la instrucción "Desmultiplexar" antes de ejecutar el segmento:

Parámetro	Operando	Valores	
K	Tag_Number	1	4
IN	Tag_Value	DW#16#FFFFFFFF	DW#16#3E4A7D

Tabla 9-35 Valores de salida de la instrucción "Desmultiplexar" tras ejecutar el segmento:

Parámetro	Operando	Valores	
OUT0	Tag_Output_1	Sin cambios	Sin cambios
OUT1	Tag_Output_2	DW#16#FFFFFFFF	Sin cambios
OUTELSE	Tag_Output_3	Sin cambios	DW#16#3E4A7D

Según el valor del operando "Tag_Number", se copia el valor de la entrada IN en la salida correspondiente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

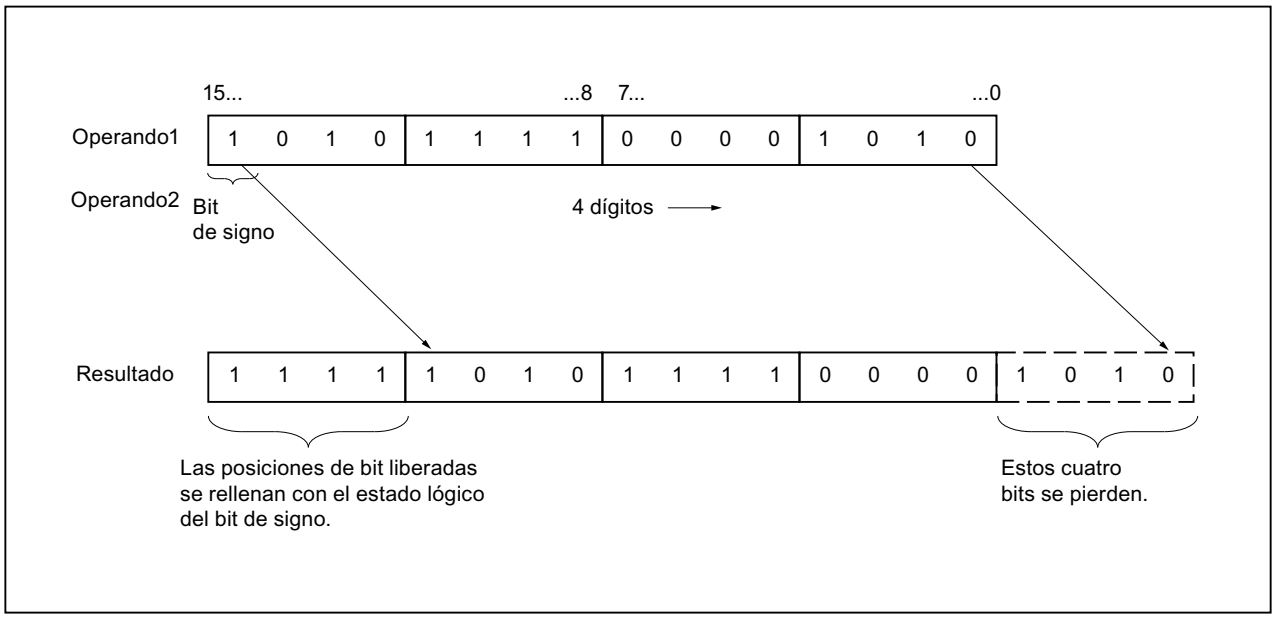
Con la instrucción "Desplazar a la derecha", el contenido del operando 1 se mueve bit a bit hacia la derecha. El operando 2 determina el número de posiciones de bit que debe desplazarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se desplaza hacia la derecha las posiciones de bit disponibles.

En los valores sin signo, se rellenan con ceros las posiciones de bits que quedan libres en el área izquierda del operando 1 al realizar el desplazamiento. Si el valor indicado lleva signo, las posiciones libres se ocupan con el estado lógico del bit de signo.

La figura siguiente muestra cómo el contenido del operando 1 del tipo de datos INT se desplaza cuatro posiciones de bit hacia la derecha:



Sintaxis

Para la instrucción "Desplazar a la derecha" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SHR_<Tipo de datos>(<Operando 1>,<Operando 2>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se desplaza
<Operando2>	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que se desplaza el valor
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SHR_INT("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	0011 1111 1010 1111
Tag_Number	3
Tag_OutValue	0000 0111 1111 0101

El contenido del operando "Tag_InValue" se desplaza tres posiciones de bit hacia la derecha. El resultado se deposita en la salida "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

SHL: Desplazar a la izquierda

Descripción

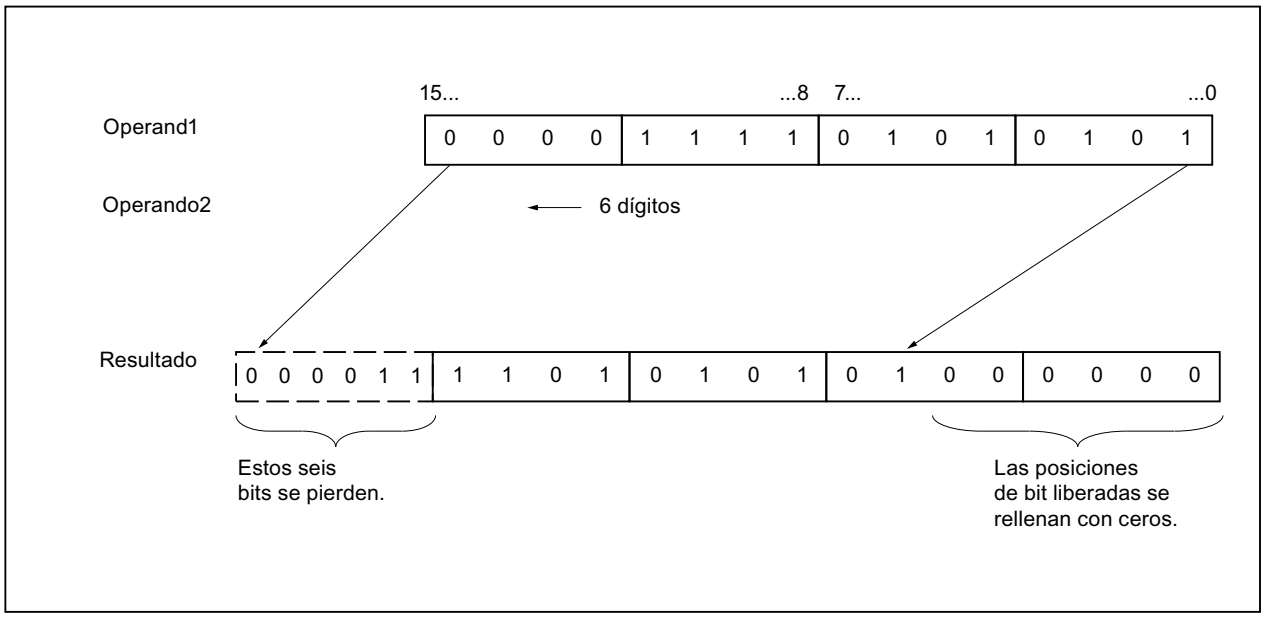
Con la instrucción "Desplazar a la izquierda", el contenido del operando 1 se mueve bit a bit hacia la izquierda. El operando 2 determina el número de posiciones de bit que debe desplazarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se desplaza hacia la izquierda las posiciones de bit disponibles.

Las posiciones de bit que quedan libres en el área derecha del operando 1 al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido del operando 1 del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Sintaxis

Para la instrucción "Desplazar a la izquierda" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SHL_<Tipo de datos>(<Operando 1>,<Operando 2>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se desplaza
<Operando2>	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que se desplaza el valor
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SHL_WORD("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	0011 1111 1010 1111
Tag_Number	4
Tag_OutValue	1111 1010 1111 0000

El contenido del operando "Tag_InValue" se desplaza cuatro posiciones de bit hacia la izquierda y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ROR: Rotar a la derecha

Descripción

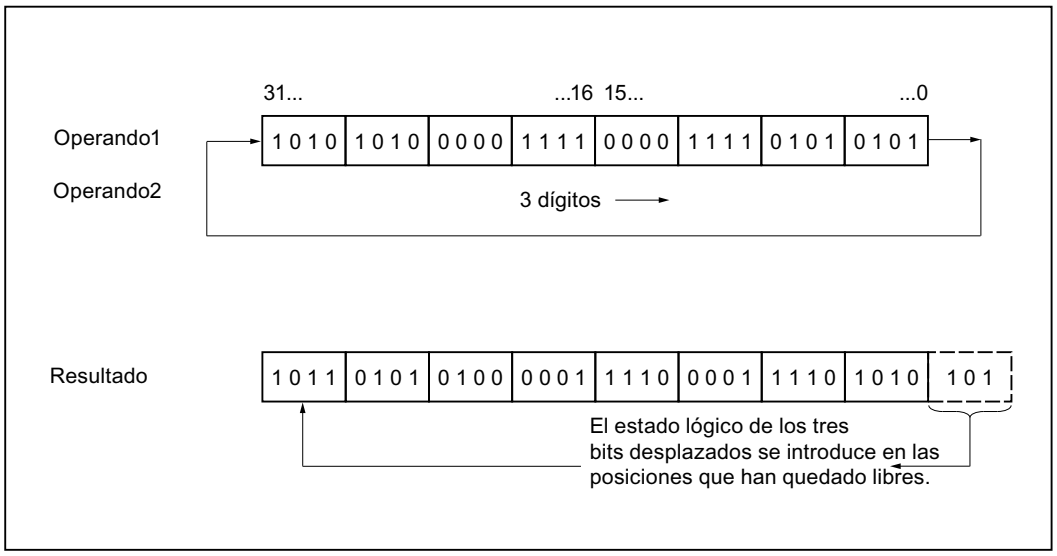
Con la instrucción "Rotar a la derecha", el contenido del operando 1 se rota bit a bit hacia la derecha. El operando 2 determina el número de posiciones de bit que debe rotarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se rota el número indicado de posiciones de bit.

Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Sintaxis

Para la instrucción "Rotar a la derecha" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ROR(<Operando 1>,<Operando 2>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se rota
<Operando2>	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que rota el valor
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ROR("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1010 1010 0000 1111 0000 1111 0101 0101
Tag_Number	5
Tag_OutValue	1010 1101 0101 0000 0111 1000 0111 1010

El contenido del operando "Tag_InValue" se rota cinco posiciones de bit hacia la derecha y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

ROL: Rotar a la izquierda

Descripción

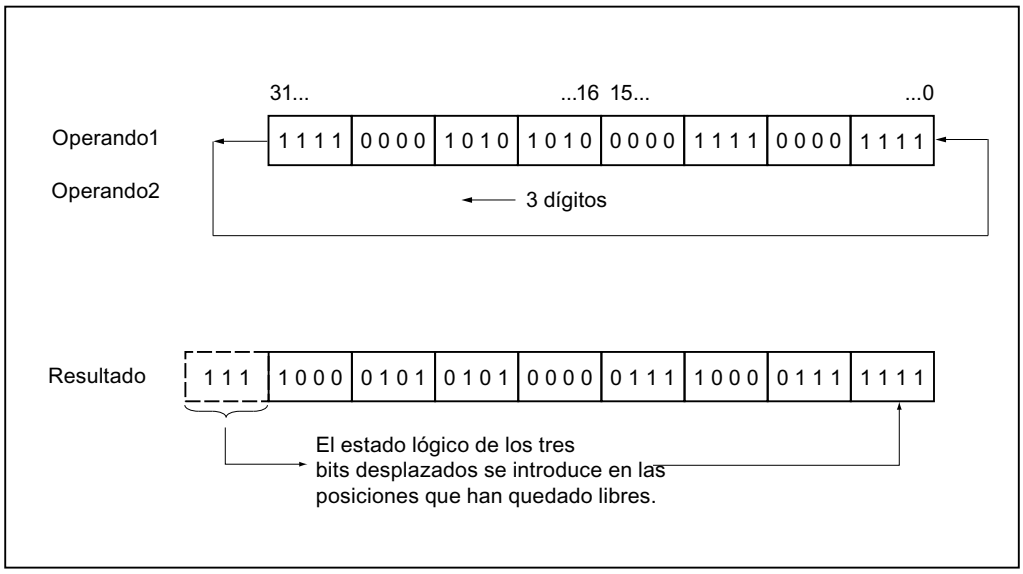
Con la instrucción "Rotar a la izquierda", el contenido del operando 1 se rota bit a bit hacia la izquierda. El operando 2 determina el número de posiciones de bit que debe rotarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se rota el número indicado de posiciones de bit.

Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Sintaxis

Para la instrucción "Rotar a la izquierda" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ROL(<Operando 1>,<Operando 2>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros	I, Q, M, D, L o constante	Valor que se rota
<Operando2>	Input	USINT, UINT, UDINT, ULINT	I, Q, M, D, L o constante	Número de posiciones de bit que rota el valor
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ROL("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1111 0000 1010 1010 0000 1111 0000 1111
Tag_Number	5
Tag_OutValue	0001 0101 0100 0001 1110 0001 1111 1110

El contenido del operando "Tag_InValue" se rota cinco posiciones de bit hacia la izquierda y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Otras instrucciones

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" ocupa los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de éste es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De

lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DRUM, <DB instancia>
     (RESET := <Operando>
      JOG := <Operando>
      DRUM_EN := <Operando>
      LST_STEP := <Operando>
      EVENT1 - 16 := <Operando>
      OUT1 - 16 => <Operando>
      Q => <Operando>
      OUT_WORD => <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L o constante	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado.
EVENT(i) 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L o constante	Bit de evento (i); El estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L o constante	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L o constante	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Primer paso predeterminado del circuito secuencial sucesivo (1 a 16)
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P o constante	Tiempo de ejecución restante para el paso actual
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	hora de sistema de la llamada anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L o constante	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente, la instrucción pasa del paso 1 al paso 2. Los bits de salida (OUT1 a OUT16) y la palabra de salida (OUT_WORD) se activan según la máscara configurada para el paso 2 y los valores del parámetro OUT_VAL.

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL DRUM, "DRUM_DB"
     (RESET := "Tag_Reset"
      JOG := "Tag_Input_Jog"
      DRUM_EN := "Tag_Input_Drum_EN"
      LST_STEP := "Tag_Number_LastStep"
      EVENTn := "MyTag_Event_n"
      OUTn => "MyTag_Output_n"
      Q => "Tag_Output_Q"
      OUT_WORD => "Tag_OutputWord"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para asignar valores a los parámetros de entrada:

Parámetro	Operando	Dirección	Valor
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_Drum_EN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DRUM_DB" de la instrucción:

Parámetro	Dirección	Valor
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE

Parámetro	Dirección	Valor
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

Los parámetros de salida tienen los valores siguientes antes de ejecutarse la instrucción:

Parámetro	Operando	Dirección	Valor
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Dirección	Valor
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Los siguientes valores están almacenados en el bloque de datos de instancia "DRUM_DB" de la instrucción:

Parámetro	Dirección	Valor
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET (solo si ET < PT), OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1"
  - Los parámetros ET, OA y CA se ponen a "0"
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET (solo si ET < PT), OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control discreto" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DCAT, <DB instancia>
      (CMD := <Operando>
       O_FB := <Operando>
       C_FB := <Operando>
       Q => <Operando>
       OA => <Operando>
       CA => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BOOL	I, Q, M, D, L o constante	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms.
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
CMD_HIS	Static	BOOL	D, L o constante	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL DCAT, "DCAT_DB"
     (CMD := "Tag_Input_CMD"
      O_FB := "Tag_Input_O_FB"
      C_FB := "Tag_Input_C_FB"
      Q => "Tag_Output_Q"
      OA => "Tag_Output_OA"
      CA => "Tag_Output_CA"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos de instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos de instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se dispara la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se asigna en este caso el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Sintaxis

Para la instrucción "Alarma de tiempo para control del motor" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MCAT, <DB instancia>
     (O_CMD := <Operando>
      C_CMD := <Operando>
      S_CMD := <Operando>
      O_FB := <Operando>
      C_FB := <Operando>
      OO => <Operando>
      CO => <Operando>
      OA => <Operando>
      CA => <Operando>
      Q => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
O_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L o constante	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L o constante	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error
ET	Static	DINT	D, L o constante	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	D, L o constante	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PREV_TIME	Static	DWORD	D, L o constante	Hora de sistema anterior
O_HIS	Static	BOOL	D, L o constante	Bit de evolución "Abrir"
C_HIS	Static	BOOL	D, L o constante	Bit de evolución "Cerrar"

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL MCAT, "MCAT_DB"
     (O_CMD := "Tag_Input_O_CMD"
      C_CMD := "Tag_Input_C_CMD"
      S_CMD := "Tag_Input_S_CMD"
      O_FB := "Tag_Input_O_FB"
      C_FB := "Tag_Input_C_FB"
      OO => "Tag_OutputOpen"
      CO => "Tag_OutputClosed"
      OA => "Tag_Output_OA"
      CA => "Tag_Output_CA"
      Q => "Tag_Output_Q"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE

Parámetro	Operando	Valor
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una correspondencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Comparar bits de entrada con bits de una máscara" se utiliza la siguiente sintaxis:

GRAPH

```
CALL IMC, <DB instancia>
     (IN_BIT0 - 15 := <Operando>
      CMP_STEP := <Operando>
      OUT => <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente se comparan todos los 16 bits de entrada con la máscara para el paso 2. El estado lógico del parámetro OUT se pone a TRUE porque los bits de entrada concuerdan con la máscara para el paso 2.

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL IMC, "IMC_DB"
     (IN_BIT0 - 15 := "Tag_Input_BITn"
      CMP_STEP := "Tag_CMP_STEP"
      OUT => "Tag_Output"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
CMP_STEP	Tag_CMP_STEP	B#16#02
OUT	Tag_Output	FALSE
ERR_CODE	Tag_ErrorCode	W#16#0000

En el bloque de datos de instancia IMC_DB de la instrucción están almacenados los siguientes valores para la máscara del paso 2:

Parámetro	Dirección	Valor
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output	TRUE
ERR_CODE	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Barrido de matriz" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SMC, <DB instancia>
     (IN_BIT0 - 15 := <Operando>
```

GRAPH

```
OUT => <Operando>
OUT_STEP => <Operando>
ERR_CODE => <Operando>
)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 14 se compara con el bit 14 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT15	Input	BOOL	I, Q, M, D, L o constante	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
LAST	Static	BYTE	I, Q, M, D, L, P o constante	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY OF WORD	I, Q, M, D, L o constante	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente, todos los 16 bits de entrada se comparan con la máscara para los pasos 0 a 5 hasta que se encuentre una correspondencia. Se consultan únicamente las

máscaras para los pasos 0 a 2, ya que la máscara para el paso 2 se corresponde con el bit de entrada.

Nota

Los parámetros estáticos pueden inicializarse en el bloque.

GRAPH

```
CALL SMC, "SMC_DB"
      (IN_BIT0 - 15 := "Tag_Input_BITn"
       OUT => "Tag_Output"
       OUT_STEP => "Tag_Output_STEP"
       ERR_CODE => "Tag_ErrorCode"
       )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
OUT	Tag_Output	FALSE
OUT_STEP	Tag_Output_STEP	B#16#00
ERR_CODE	Tag_ErrorCode	W#16#0000

En el bloque de datos de instancia SMC_DB de la instrucción están almacenados los siguientes valores para la máscara del paso 2:

Parámetro	Dirección	Valor
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE
LAST	DB84	B#16#05

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output	TRUE
OUT_STEP	Tag_Output_STEP	B#16#02
ERR_CODE	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor para la amplificación en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" sólo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse para suprimir perturbaciones o como filtro pasa bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

GRAPH

```
CALL LEAD_LAG, <DB instancia>
      (IN := <Operando>
       SAMPLE_T := <Operando>
       OUT => <Operando>
       ERR_CODE => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL LEAD_LAG, "LEAD_LAG_DB"
     (IN := "Tag_Input"
      SAMPLE_T := "Tag_Input_SAMPLE_T"
      OUT => "Tag_Output_Result"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

En el bloque de datos de instancia "LEAD_LAG_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos de instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

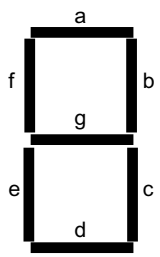
Principios básicos de GRAPH (Página 2825)

SEG: Generar patrón de bits para el display de 7 segmentos

Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" convierte cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en cuatro códigos equivalentes para un display de 7 segmentos y los escribe en la palabra doble de la salida (OUT).

Entre los valores de entrada hexadecimales y los patrones de bits de la salida existe la relación siguiente:

Cifra	-gfedcba	Display	Display de siete segmentos
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Sintaxis

Para la instrucción "Generar patrón de bits para el display de 7 segmentos" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SEG
      (IN := <Operando>
       OUT => <Operando>
       )
```

Parámetros

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits del destino con cuatro bytes

La instrucción "Generar patrón de bits para el display de 7 segmentos" no detecta condiciones de error.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SEG
      (IN := "Tag_Input"
       OUT => "Tag_Output"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

GRAPH

```
CALL BCDCPL
     (IN := <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Secuencias de bits	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL BCDCPL
```

GRAPH

```
(IN := "Tag_Input"
 ERR_CODE => "Tag_Output"
 )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" cuenta el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

Sintaxis

Para la instrucción "Contar número de bits activados" se utiliza la siguiente sintaxis:

GRAPH

```
CALL BITSUM
      (IN := <Operando>
       RET_VAL => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL BITSUM
     (IN := "Tag_Input"
      RET_VAL => "Tag_Output"
      )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Instrucciones KOP Comparación

Instrucciones KOP generales

Descripción

Para la programación en GRAPH se dispone de diferentes instrucciones KOP.

La tabla siguiente ofrece una vista general de las instrucciones y su disponibilidad:

Instrucciones KOP	Instrucción permanente	Transición / Supervisión / Interlock
General		
Insertar segmento	x	
Cuadro vacío	x	x
Abrir rama	x	x
Cerrar rama	x	x
-|: Insertar entrada	x	
Operaciones lógicas con bits		
---| |---: Contacto normalmente abierto (Página 3283)	x	x
---| / |---: Contacto normalmente cerrado (Página 3285)	x	x
--| NOT |--: Invertir RLO (Página 3286)	x	x
---( )---: Asignación (Página 3286)	x	
---(/)---: Negar asignación (Página 3287)	x	
---(R)---: Desactivar salida (Página 3288)	x	
---(S)---: Activar salida (Página 3289)	x	
SET_BF: Activar mapa de bits (Página 3290)	x	
RESET_BF: Desactivar mapa de bits (Página 3291)	x	
SR: Flipflop de activación/desactivación (Página 3292)	x	
RS: Flipflop de desactivación/activación (Página 3294)	x	
--|P|--: Consultar flanco de señal ascendente de un operando (Página 3295)	x	
--|N|--: Consultar flanco de señal descendente de un operando (Página 3296)	x	
--(P)--: Activar operando con flanco de señal ascendente (Página 3298)	x	
--(N)--: Activar operando con flanco de señal descendente (Página 3299)	x	
P_TRIG: Consultar flanco de señal ascendente del RLO (Página 3300)	x	
N_TRIG: Consultar flanco de señal descendente del RLO (Página 3302)	x	
R_TRIG: Activar variable con flanco de señal ascendente (Página 3303)	x	
F_TRIG: Activar variable con flanco de señal descendente (Página 3305)	x	
Temporizadores		
Temporizadores CEI		
TP: Impulso (Página 3306)	x	
TON: Retardo al conectar (Página 3309)	x	
TOF: Retardo al desconectar (Página 3311)	x	
TONR: Acumulador de tiempo (Página 3314)	x	
---(TP)---: Arrancar temporizador como impulso (Página 3316)	x	
---(TON)---: Arrancar temporizador como retardo a la conexión (Página 3319)	x	
---(TOF)---: Arrancar temporizador como retardo a la desconexión (Página 3321)	x	
---(TONR)---: Acumulador de tiempo (Página 3323)	x	
---(RT)---: Inicializar temporizador (Página 3325)	x	
---(PT)---: Cargar tiempo (Página 3326)	x	
Temporizadores SIMATIC (S7-1500)		
S_PULSE: Parametrizar y arrancar temporizador como impulso (Página 3328)	x	
S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado (Página 3331)	x	

Instrucciones KOP	Instrucción permanente	Transición / Supervisión / Interlock
S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión (Página 3334)	x	
S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria (Página 3337)	x	
S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión (Página 3340)	x	
Contadores		
Contadores CEI		
CTU: Contador ascendente (Página 3351)	x	
CTD: Contador descendente (Página 3354)	x	
CTUD: Contador ascendente - descendente (Página 3357)	x	
Contadores SIMATIC (S7-1500)		
S_CU : Parametrizar e incrementar contador (Página 3363)	x	
S_CD : Parametrizar y decrementar contador (Página 3365)	x	
S_CUD : Parametrizar e incrementar/decrementar contador (Página 3361)	x	
Comparación		
CMP ==: Igual (Página 3371)	x	x
CMP <>: Diferente (Página 3373)	x	x
CMP >=: Mayor o igual (Página 3375)	x	x
CMP <=: Menor o igual (Página 3377)	x	x
CMP >: Mayor (Página 3379)	x	x
CMP <: Menor (Página 3381)	x	x
IN_RANGE: Valor dentro del rango (Página 3383)	x	
OUT_RANGE: Valor fuera del rango (Página 3384)	x	
---| OK |---: Comprobar validez (Página 3386)	x	
---| NOT_OK |---: Comprobar invalidez (Página 3387)	x	
EQ_Type: Comparar si el tipo de datos es IGUAL al tipo de datos de una variable (Página 3388)	x	
NE_Type: Comparar si el tipo de datos es DIFERENTE del tipo de datos de una variable (Página 3389)	x	
EQ_ElemType: Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable (Página 3390)	x	
NE_ElemType: Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable (Página 3392)	x	
IS_NULL: Comparar si es IGUAL A CERO (Página 3393)	x	
NOT_NULL: Comparar si es DIFERENTE DE CERO (Página 3394)	x	
IS_ARRAY: Consultar si es un ARRAY (Página 3395)	x	
CMP>T: Superior al tiempo de activación de la etapa (Página 4680)		x
CMP>U: Superior al tiempo de activación ininterrumpido de la etapa (Página 4682)		x
Funciones matemáticas		
CALCULATE: Calcular (Página 3396)	x	
ADD: Sumar (Página 3400)	x	
SUB: Restar (Página 3402)	x	

Instrucciones KOP	Instrucción permanente	Transición / Supervisión / Interlock
MUL: Multiplicar (Página 3403)	x	
DIV: Dividir (Página 3405)	x	
MOD: Obtener resto de división (Página 3406)	x	
NEG: Generar complemento a dos (Página 3407)	x	
INC: Incrementar (Página 3409)	x	
DEC: Decrementar (Página 3410)	x	
ABS: Calcular valor absoluto (Página 3411)	x	
MIN: Determinar mínimo (Página 3413)	x	
MAX: Determinar máximo (Página 3415)	x	
LIMIT: Ajustar valor límite (Página 3417)	x	
SQR: Calcular cuadrado (Página 3419)	x	
SQRT: Calcular raíz cuadrada (Página 3421)	x	
LN: Calcular logaritmo natural (Página 3422)	x	
EXP: Calcular valor exponencial (Página 3423)	x	
SIN: Calcular valor de seno (Página 3425)	x	
COS: Calcular valor de coseno (Página 3426)	x	
TAN: Calcular valor de tangente (Página 3427)	x	
ASIN: Calcular valor de arcoseno (Página 3429)	x	
ACOS: Calcular valor de arcocoseno (Página 3430)	x	
ATAN: Calcular valor de arcotangente (Página 3432)	x	
FRAC: Determinar decimales (Página 3433)		
EXPT: Elevar a potencia (Página 3435)	x	
Transferencia		
MOVE: Copiar valor (Página 3436)	x	
Deserialize: Deserializar (Página 3442)	x	
Serialize: Serializar (Página 3445)	x	
FieldRead: Leer campo (Página 3449)	x	
FieldWrite: Escribir campo (Página 3451)	x	
MOVE_BLK: Copiar área (Página 3453)	x	
MOVE_BLK_VARIANT: Copiar área (Página 3455)	x	
UMOVE_BLK: Copiar área sin interrupciones (Página 3459)	x	
FILL_BLK: Rellenar área (Página 3461)	x	
UFILL_BLK: Rellenar área sin interrupciones (Página 3463)	x	
SWAP: Cambiar disposición (Página 3466)	x	
ReadFromArrayDB: Leer desde bloque de datos de array (Página 3468)	x	
WriteToArrayDB: Escribir en bloque de datos de array (Página 3470)	x	
ReadFromArrayDBL: Leer de bloque de datos de array de la memoria de carga (Página 3473)	x	
WriteToArrayDBL: Escribir en bloque de datos de array de la memoria de carga (Página 3476)	x	
VariantGet: Leer valor de una variable VARIANT (Página 3480)	x	
VariantPut: Escribir valor en una variable VARIANT (Página 3481)	x	

Instrucciones KOP	Instrucción permanente	Transición / Supervisión / Interlock
CountOfElements: Consultar número de elementos de ARRAY (Página 3483)	x	
BLKMOV: Copiar área (S7-1500) (Página 3484)	x	
UBLKMOV: Copiar área sin interrupciones (S7-1500) (Página 3487)	x	
FILL: Rellenar área (S7-1500) (Página 3490)	x	
Conversión		
CONVERT: Convertir valor (Página 3493)	x	
ROUND: Redondear número (Página 3494)	x	
CEIL: Redondear número en coma flotante al siguiente entero superior (Página 3496)	x	
FLOOR: Redondear número en coma flotante al siguiente entero inferior (Página 3497)	x	
TRUNC: Truncar a entero (Página 3499)	x	
SCALE_X: Escalar (Página 3500)	x	
NORM_X: Normalizar (Página 3503)	x	
SCALE: Escalar (S7-1500) (Página 3506)	x	
UNSCALE: Desescalar (S7-1500) (Página 3508)	x	
Control del programa		
Control en runtime		
ENDIS_PW: Limitar y habilitar legitimación de la contraseña (Página 3521)	x	
RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo (Página 3526)	x	
STP: Finalizar programa (Página 3526)	x	
GET_ERROR: Consultar error localmente (Página 3527)	x	
GET_ERR_ID: Consultar ID de error localmente (Página 3531)	x	
INIT_RD: Inicializar todos los datos remanentes (Página 3534)	x	
WAIT: Programar retardo (Página 3535)	x	
RUNTIME: Medir tiempo de ejecución (Página 3536)	x	
Operaciones lógicas con palabras		
AND: Operación lógica Y (Página 3539)	x	
OR: Operación lógica O (Página 3540)	x	
XOR: Operación lógica O-exclusiva (Página 3542)	x	
INV: Complemento a 1 (Página 3544)	x	
DECO: Descodificar (Página 3545)	x	
ENCO: Codificar (Página 3547)	x	
SEL: Seleccionar (Página 3548)	x	
MUX: Multiplexar (Página 3550)	x	
DEMUX: Desmultiplexar (Página 3553)	x	
Desplazamiento y rotación		
SHR: Desplazar a la derecha (Página 3556)	x	
SHL: Desplazar a la izquierda (Página 3558)	x	
ROR: Rotar a la derecha (Página 3560)	x	
ROL: Rotar a la izquierda (Página 3562)	x	
Otras instrucciones		
DRUM: Materializar circuito secuencial sucesivo (Página 3564)	x	

Instrucciones KOP	Instrucción permanente	Transición / Supervisión / Interlock
DCAT: Alarma de tiempo para control discreto (Página 3568)	x	
MCAT: Alarma de tiempo para control del motor (Página 3571)	x	
IMC: Comparar los bits de entrada con los bits de la máscara (Página 3575)	x	
SMC: Barrido de matriz (Página 3578)	x	
LEAD_LAG: Algoritmo Lead y Lag (Página 3580)	x	
SEG: Generar patrón de bits para el display de 7 segmentos (Página 3584)	x	
BCDCPL: Generar complemento a diez (Página 3586)	x	
BITSUM: Contar número de bits activados (Página 3587)	x	

CMP>T: Superior al tiempo de activación de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación de la etapa" se vigila la duración total de una etapa en condiciones de supervisión. De esta manera se registra el tiempo de los posibles eventos o errores.

En la condición programada se compara el tiempo de activación en curso o el último de una etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

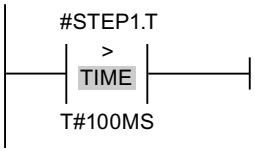
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa
<Operando2>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.T esté por debajo de 100 ms, el RLO será "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

CMP>U: Superior al tiempo de activación ininterrumpido de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación ininterrumpido de la etapa", se vigila la duración de una etapa, sin tener en cuenta las posibles averías, en condiciones de supervisión. No se registra el tiempo de posibles eventos o fallos sino que se supervisa la duración de la mera etapa.

En la condición programada se compara todo el tiempo de activación de la etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

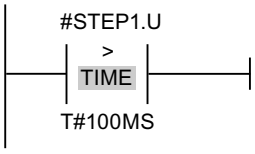
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación ininterrumpido de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa sin contar averías
<Operando2>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.U, sin tener en cuenta las posibles averías, esté por debajo de 100 ms, el RLO será igual a "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2559)

Principios básicos de GRAPH (Página 2825)

Instrucciones FUP Comparación

Instrucciones FUP generales

Descripción

Para la programación en GRAPH se dispone de diferentes instrucciones FUP.

La tabla siguiente ofrece una vista general de las instrucciones y su disponibilidad:

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
General		
Nuevo segmento	x	
Cuadro vacío	x	x
Derivación: Abrir rama	x	x
-|: Insertar entrada (Página 3594)	x	x
-o|: Invertir RLO (Página 3595)	x	x
Operaciones lógicas con bits		
&: Operación lógica Y (Página 3588)	x	x
>=1: Operación lógica O (Página 3590)	x	x
X: Operación lógica O-exclusiva (Página 3591)	x	
=: Asignación (Página 3595)	x	
/=: Negar asignación (Página 3596)	x	
R: Desactivar salida (Página 3597)	x	
S: Activar salida (Página 3598)	x	
SET_BF: Activar mapa de bits (Página 3599)	x	
RESET_BF: Desactivar mapa de bits (Página 3600)	x	
SR: Flipflop de activación/desactivación (Página 3602)	x	
RS: Flipflop de desactivación/activación (Página 3603)	x	
P: Consultar flanco de señal ascendente de un operando (Página 3604)	x	
N: Consultar flanco de señal descendente de un operando (Página 3606)	x	
P=: Activar operando con flanco de señal ascendente (Página 3607)	x	

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
N=: Activar operando con flanco de señal descendente (Página 3608)	x	
P_TRIG: Consultar flanco de señal ascendente del RLO (Página 3610)	x	
N_TRIG: Consultar flanco de señal descendente del RLO (Página 3611)	x	
R_TRIG: Activar variable con flanco de señal ascendente (Página 3612)	x	
F_TRIG: Activar variable con flanco de señal descendente (Página 3613)	x	
Temporizadores		
Temporizadores CEI		
TP: Impulso (Página 3615)	x	
TON: Retardo al conectar (Página 3617)	x	
TOF: Retardo al desconectar (Página 3620)	x	
TONR: Acumulador de tiempo (Página 3622)	x	
TP: Arrancar temporizador como impulso (Página 3625)	x	
TON: Arrancar temporizador como retardo a la conexión (Página 3627)	x	
TOF: Arrancar temporizador como retardo a la desconexión (Página 3629)	x	
TONR: Acumulador de tiempo (Página 3631)	x	
RT: Inicializar temporizador (Página 3634)	x	
PT: Cargar tiempo (Página 3635)	x	
Temporizadores SIMATIC		
S_PULSE: Parametrizar y arrancar temporizador como impulso (Página 3637)	x	
S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado (Página 3640)	x	
S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión (Página 3643)	x	
S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria (Página 3646)	x	
S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión (Página 3649)	x	
Contadores		
Contadores CEI		
CTU: Contador ascendente (Página 3662)	x	
CTD: Contador descendente (Página 3665)	x	
CTUD: Contador ascendente - descendente (Página 3668)	x	
Contadores SIMATIC		
S_CU: Parametrizar e incrementar contador (Página 3674)	x	
S_CD : Parametrizar y decrementar contador (Página 3676)	x	
S_CUD: Parametrizar e incrementar/decrementar contador (Página 3672)	x	
Comparación		
CMP ==: Igual (Página 3682)	x	x
CMP <>: Diferente (Página 3685)	x	x
CMP >=: Mayor o igual (Página 3687)	x	x
CMP <=: Menor o igual (Página 3689)	x	x
CMP >: Mayor (Página 3691)	x	x
CMP <: Menor (Página 3693)	x	x
IN_RANGE: Valor dentro del rango (Página 3695)	x	

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
OUT_RANGE: Valor fuera del rango (Página 3696)	x	
---| OK |---: Comprobar validez (Página 3698)	x	
---| NOT_OK |---: Comprobar invalidez (Página 3699)	x	
EQ_Type: Comparar si el tipo de datos es IGUAL al tipo de datos de una variable (Página 3700)	x	
NE_Type: Comparar si el tipo de datos es DIFERENTE del tipo de datos de una variable (Página 3701)	x	
EQ_ElemType: Comparar si el tipo de datos de un elemento es IGUAL al tipo de datos de una variable (Página 3702)	x	
NE_ElemType: Comparar si el tipo de datos de un elemento es DIFERENTE del tipo de datos de una variable (Página 3703)	x	
IS_NULL: Comparar si es IGUAL A CERO (Página 3704)	x	
NOT_NULL: Comparar si es DIFERENTE DE CERO (Página 3706)	x	
IS_ARRAY: Consultar si es un ARRAY (Página 3707)	x	
CMP>T: Superior al tiempo de activación de la etapa (Página 4687)		x
CMP>U: Superior al tiempo de activación ininterrumpido de la etapa (Página 4688)		x
Funciones matemáticas		
CALCULATE: Calcular (Página 3708)	x	
ADD: Sumar (Página 3711)	x	
SUB: Restar (Página 3713)	x	
MUL: Multiplicar (Página 3714)	x	
DIV: Dividir (Página 3716)	x	
MOD: Obtener resto de división (Página 3717)	x	
NEG: Generar complemento a dos (Página 3719)	x	
INC: Incrementar (Página 3720)	x	
DEC: Decrementar (Página 3722)	x	
ABS: Calcular valor absoluto (Página 3723)	x	
MIN: Determinar mínimo (Página 3724)	x	
MAX: Determinar máximo (Página 3727)	x	
LIMIT: Ajustar valor límite (Página 3729)	x	
SQR: Calcular cuadrado (Página 3731)	x	
SQRT: Calcular raíz cuadrada (Página 3733)	x	
LN: Calcular logaritmo natural (Página 3734)	x	
EXP: Calcular valor exponencial (Página 3736)	x	
SIN: Calcular valor de seno (Página 3737)	x	
COS: Calcular valor de coseno (Página 3738)	x	
TAN: Calcular valor de tangente (Página 3740)	x	
ASIN: Calcular valor de arcoseno (Página 3741)	x	
ACOS: Calcular valor de arcocoseno (Página 3743)	x	
ATAN: Calcular valor de arcotangente (Página 3744)	x	
FRAC: Determinar decimales (Página 3746)	x	
EXPT: Elevar a potencia (Página 3747)	x	

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
Transferencia		
MOVE: Copiar valor (Página 3749)	x	
Deserialize: Deserializar (Página 3755)	x	
Serialize: Serializar (Página 3759)	x	
FieldRead: Leer campo (Página 3762)	x	
FieldWrite: Escribir campo (Página 3764)	x	
MOVE_BLK: Copiar área (Página 3766)	x	
MOVE_BLK_VARIANT: Copiar área (Página 3768)	x	
UMOVE_BLK: Copiar área sin interrupciones (Página 3773)	x	
FILL_BLK: Rellenar área (Página 3775)	x	
UFILL_BLK: Rellenar área sin interrupciones (Página 3777)	x	
SWAP: Cambiar disposición (Página 3779)	x	
ReadFromArrayDB: Leer desde bloque de datos de array (Página 3781)	x	
WriteToArrayDB: Escribir en bloque de datos de array (Página 3784)	x	
ReadFromArrayDBL: Leer de bloque de datos de array de la memoria de carga (Página 3786)	x	
WriteToArrayDBL: Escribir en bloque de datos de array de la memoria de carga (Página 3789)	x	
VariantGet: Leer valor de una variable VARIANT (Página 3793)	x	
VariantPut: Escribir valor en una variable VARIANT (Página 3794)	x	
CountOfElements: Consultar número de elementos de ARRAY (Página 3796)	x	
BLKMOV: Copiar área (S7-1500) (Página 3797)	x	
UBLKMOV: Copiar área sin interrupciones (S7-1500) (Página 3800)	x	
FILL: Rellenar área (S7-1500) (Página 3803)	x	
Conversión		
CONVERT: Convertir valor (Página 3805)	x	
ROUND: Redondear número (Página 3807)	x	
CEIL: Redondear número en coma flotante al siguiente entero superior (Página 3809)	x	
FLOOR: Redondear número en coma flotante al siguiente entero inferior (Página 3810)	x	
TRUNC: Truncar a entero (Página 3812)	x	
SCALE_X: Escalar (Página 3813)	x	
NORM_X: Normalizar (Página 3816)	x	
SCALE: Escalar (S7-1500) (Página 3819)	x	
UNSCALE: Desescalar (S7-1500) (Página 3821)	x	
Control del programa		
Control en runtime		
ENDIS_PW: Limitar y habilitar legitimación de la contraseña (Página 3834)	x	
RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo (Página 3839)	x	
STP: Finalizar programa (Página 3839)	x	
GET_ERROR: Consultar error localmente (Página 3840)	x	
GET_ERR_ID: Consultar ID de error localmente (Página 3844)	x	
INIT_RD: Inicializar todos los datos remanentes (Página 3847)	x	

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
WAIT: Programar retardo (Página 3849)	x	
RUNTIME: Medir tiempo de ejecución (Página 3850)	x	
Operaciones lógicas con palabras		
AND: Operación lógica Y (Página 3852)	x	
OR: Operación lógica O (Página 3854)	x	
XOR: Operación lógica O-exclusiva (Página 3856)	x	
INV: Complemento a 1 (Página 3858)	x	
DECO: Descodificar (Página 3859)	x	
ENCO: Codificar (Página 3860)	x	
SEL: Seleccionar (Página 3862)	x	
MUX: Multiplexar (Página 3864)	x	
DEMUX: Desmultiplexar (Página 3867)	x	
Desplazamiento y rotación		
SHR: Desplazar a la derecha (Página 3871)	x	
SHL: Desplazar a la izquierda (Página 3873)	x	
ROR: Rotar a la derecha (Página 3875)	x	
ROL: Rotar a la izquierda (Página 3877)	x	
Otras instrucciones		
DRUM: Materializar circuito secuencial sucesivo (Página 3879)	x	
DCAT: Alarma de tiempo para control discreto (Página 3883)	x	
MCAT: Alarma de tiempo para control del motor (Página 3886)	x	
IMC: Comparar los bits de entrada con los bits de la máscara (Página 3891)	x	
SMC: Barrido de matriz (Página 3893)	x	
LEAD_LAG: Algoritmo Lead y Lag (Página 3896)	x	
SEG: Generar patrón de bits para el display de 7 segmentos (Página 3899)	x	
BCDCPL: Generar complemento a diez (Página 3901)	x	
BITSUM: Contar número de bits activados (Página 3903)	x	

CMP>T: Superior al tiempo de activación de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación de la etapa" se puede vigilar la duración total de una etapa en condiciones de supervisión. De esta manera se registra el tiempo de los posibles eventos o errores.

En la condición programada se compara el tiempo de activación en curso o el último de una etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

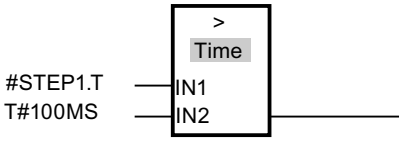
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa
<Operando2>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.T esté por debajo de 100 ms, el RLO será "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

CMP>U: Superior al tiempo de activación ininterrumpido de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación ininterrumpido de la etapa", se puede vigilar la duración de una etapa, sin tener en cuenta las posibles averías, en condiciones de supervisión. No se registra el tiempo de posibles eventos o fallos sino que se supervisa la duración de la mera etapa.

En la condición programada se compara todo el tiempo de activación de una etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

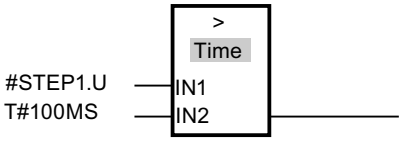
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación ininterrumpido de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa sin contar averías
<Operando2>	Input	Enteros, números en coma flotante, temporizadores, secuencias de caracteres, DATE, DT, DTL, TOD, LTOD, LDT	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.U, sin tener en cuenta las posibles averías, esté por debajo de 100 ms, el RLO será igual a "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

9.8.1.3 Instrucciones avanzadas

Fecha y hora

T_COMP: Comparar variables de tiempo

Descripción

Con esta instrucción se comparan los contenidos de dos variables de los tipos de datos "Temporizadores" o "Fecha y hora".

La instrucción soporta la comparación de los tipos de datos siguientes: DATE, TIME, LTIME, TOD (TIME_OF_DAY), LTOD (LTIME_OF_DAY), DT (DATE_AND_TIME), LDT (DATE_AND_LTIME), DTL, S5Time.

Para realizar la comparación, los tipos de datos deben coincidir en longitud y formato.

El resultado de la comparación se muestra como valor de retorno en el parámetro OUT. Para ello, el parámetro OUT se pone a "1" una vez que se ha cumplido la condición de comparación utilizada.

Existen las siguientes posibilidades de comparación:

Símbolo	Descripción
EQ	El valor de retorno tiene el estado lógico "1" si los momentos son idénticos en el parámetro IN1 y en el parámetro IN2.
NE	El valor de retorno tiene el estado lógico "1" si los momentos no son idénticos en el parámetro IN1 y en el parámetro IN2.
GE	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es mayor (más reciente) o igual al del parámetro IN2 .
LE	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es menor (más antiguo) o igual al del parámetro IN2.
GT	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es mayor (más reciente) que el del parámetro IN2.
LT	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es menor (más antiguo) que el del parámetro IN2 .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_COMP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	DATE, TIME, LTIME, TOD, LTOD, DT, LDT, DTL	I, Q, M, D, L, P o constante	Primer valor que se va a comparar
IN2	Input	DATE, TIME, LTIME, TOD, LTOD, DT, LDT, DTL	I, Q, M, D, L, P o constante	Segundo valor que se va a comparar
OUT	Output	BOOL	I, Q, M, D, L, P	Valor de retorno

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_CONV: Convertir y extraer tiempos

Descripción

Con la instrucción "T_CONV" se convierte el tipo de datos del parámetro de entrada IN en el tipo de datos de la salida OUT. Con los cuadros de instrucciones de la entrada y la salida se pueden seleccionar los formatos de datos para la conversión.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_CONV". Si en el parámetro de entrada y de salida se utiliza el mismo tipo de datos, la instrucción copia el correspondiente valor.

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	Enteros, TIME, fecha y hora*	WORD, enteros, temporizadores, fecha y hora*	I, Q, M, D, L, P o constante	Valor que se va a convertir
OUT	Return	Enteros, TIME, fecha y hora*	WORD, enteros, temporizadores, fecha y hora*	I, Q, M, D, L, P	Resultado de la conversión
* El volumen de tipos de datos soportados depende de la CPU. Consulte en el resumen de los correspondientes tipos de datos válidos los tipos de datos que soportan los módulos S7-1200 y S7-1500.					

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_ADD: Sumar tiempos

Descripción

Con esta instrucción se suma la indicación horaria del parámetro de entrada IN1 a la indicación horaria del parámetro de entrada IN2. El resultado se puede consultar en el parámetro de salida OUT. Es posible sumar los formatos siguientes:

- Suma de un tiempo a otro tiempo.
  Ejemplo: suma de un tipo de datos TIME a otro tipo de datos TIME.
- Suma de un tiempo a un momento.
  Ejemplo: suma de un tipo de datos TIME al tipo de datos DTL.

El tipo de datos para los valores del parámetro de entrada IN1 y el parámetro de salida OUT se determinan seleccionándolos en los cuadros de instrucciones de la entrada y de la salida. En el parámetro de entrada IN2 solo pueden especificarse indicaciones horarias en formato TIME (en módulos S7-1500 también LTIME).

Parámetros

Las tablas siguientes muestran los parámetros de la instrucción "T_ADD" correspondientes a las posibles conversiones:

Tabla 9-36 Suma de un tiempo a otro tiempo

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	TIME	TIME, LTIME	I, Q, M, D, L, P o constante	Primer sumando
IN2	Input	TIME	TIME, LTIME	I, Q, M, D, L, P o constante	Segundo sumando
OUT	Return	DINT, DWORD, TIME, TOD	TIME, LTIME,	I, Q, M, D, L, P	Resultado de la suma La selección de los tipos de datos depende de los tipos de datos seleccionados para los parámetros de entrada IN1 e IN2.

Tabla 9-37 Suma de un tiempo a un momento

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	DTL, TOD	DT, TOD, LTOD, LDT, DTL	I, Q, M, D, L, P o constante	Primer sumando
IN2	Input	TIME	TIME, LTIME	I, Q, M, D, L, P o constante	Segundo sumando
OUT	Return	DINT, DWORD, TIME, TOD, UDINT, DTL	DT, DTL, LDT, TOD, LTOD	I, Q, M, D, L, P	Resultado de la suma La selección de los tipos de datos depende de los tipos de datos seleccionados para los parámetros de entrada IN1 e IN2.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_SUB: Restar tiempos

Descripción

Con esta instrucción se resta la indicación horaria del parámetro de entrada IN2 de la indicación horaria del parámetro de entrada IN1. La diferencia se puede consultar en el parámetro de salida OUT. Es posible restar los formatos siguientes:

- Restar un tiempo a otro tiempo
  Ejemplo: restar un tiempo del tipo de datos TIME a otro tiempo del tipo de datos TIME. El resultado puede emitirse en una variable del tipo de datos TIME.
- Restar un tiempo a un momento
  Ejemplo: restar un tiempo del tipo de datos TIME a un momento del tipo de datos DTL. El resultado puede emitirse en una variable del tipo de datos DTL.

Los formatos de los valores del parámetro de entrada IN1y del parámetro de salida OUT se determinan seleccionando los tipos de datos para el parámetro de entrada y el parámetro de salida de la instrucción.

Parámetros

Las tablas siguientes muestran los parámetros de la instrucción "T_SUB" correspondientes a las posibles conversiones:

Tabla 9-38 Restar un tiempo a otro tiempo

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	TIME	TIME, LTIME	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	TIME	TIME, LTIME	I, Q, M, D, L, P o constante	Sustraendo
OUT	Return	DINT, DWORD, TIME, TOD, UDINT	TIME, LTIME	I, Q, M, D, L, P	Resultado de la resta

Tabla 9-39 Restar un tiempo a un momento

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	DTL, TOD	TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	TIME	TIME, LTIME	I, Q, M, D, L, P o constante	Sustraendo
OUT	Return	DTL, DINT, DWORD, TIME, TOD, UDINT	TOD, LTOD, DTL, DT, LDT	I, Q, M, D, L, P	Resultado de la resta

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_DIFF: Diferencia de tiempo

Descripción

Con esta instrucción se resta la indicación horaria del parámetro de entrada IN2 de la indicación horaria del parámetro de entrada IN1. El resultado se devuelve en el parámetro de salida OUT.

- Si la indicación horaria en el parámetro de entrada IN2es mayor que la indicación horaria en el parámetro de entrada IN1, se emite un valor negativo en el parámetro de salida OUT.
- Si el resultado de la resta se encuentra fuera del rango numérico TIME, el resultado se ajusta a "0" (0:00) y la salida de habilitación ENO = "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_DIFF":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	DTL, DATE, TOD	DTL, DATE, DT, TOD, LTOD, LDT	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	DTL, DATE, TOD	DTL, DATE, DT, TOD, LTOD, LDT	I, Q, M, D, L, P o constante	Sustraendo
OUT	Return	TIME, INT	TIME, LTIME, INT	I, Q, M, D, L, P	Diferencia de los parámetros de entrada

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_COMBINE: Combinar tiempos

Descripción

La instrucción combina el valor de una fecha con el de una indicación horaria y los convierte en una indicación horaria combinada de fecha y hora.

- En el parámetro de entrada IN1 se introduce la indicación de fecha. En el tipo de datos DATE debe emplearse un valor comprendido entre 1990-01-01 y 2089-12-31 (no se comprueba).
- En el valor de entrada IN2 se introduce la indicación horaria (tipo de datos TOD/LTOD).
- En el valor de salida OUT se emite el tipo de datos combinado para indicación de fecha y hora.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_COMBINE":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	DATE	DATE	I, Q, M, D, L, P o constante	Variable de entrada de la fecha
IN2	Input	TOD	TOD, LTOD	I, Q, M, D, L, P o constante	Variable de entrada de la hora
OUT	Return	DTL	DT, DTL, LDT	I, Q, M, D, L, P	Valor de retorno de fecha y hora

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Funciones de reloj

WR_SYS_T: Ajustar la hora

Descripción

La instrucción ajusta la fecha y hora del reloj de la CPU. La indicación de fecha y hora se especifica en el parámetro de entrada IN . El valor debe estar dentro del rango siguiente:

- Con DT: mín. DT#1990-01-01-0:0:0, máx. DT#2200-12-31-23:59:59.999
- Con LDT: mín. LDT#1970-1-1-0:0:0.000000000, máx. LDT#2200-12-31-23:59:99.999999999
- Con DTL: mín. DTL#1970-01-01-00:00:00.0, máx. DTL#2200-12-31-23:59:59.999999999

En el parámetro de salida RET_VAL se puede consultar si han ocurrido errores durante la ejecución de la instrucción.

La información de zona horaria local u horario de verano no se puede transferir con la instrucción "WR_SYS_T".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_SYS_T":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	DTL	DT, DTL, LDT	I, Q, M, D, L, P o constante *	Fecha y hora
RET_VAL	Return	INT	INT	I, Q, M, D, L, P	Estado de la instrucción
* Los tipos de datos DT y DTL no pueden utilizarse para las áreas de memoria Entrada, Salida y Marca.					

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0000	Sin errores
8080	Error en la fecha
8081	Error en la hora
8082	Mes no válido
8083	Día no válido
8084	Indicación de horas no válida
8085	Indicación de minutos no válida
8086	Indicación de segundos no válida
8087	Indicación de nanosegundos no válida
80B0	Fallo del reloj en tiempo real
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RD_SYS_T: Leer la hora

Descripción

La instrucción lee la fecha y hora actuales del reloj de la CPU.

Los datos leídos se devuelven en el parámetro de salida OUT de la instrucción. El valor emitido no contiene indicaciones respecto a la zona horaria local ni al horario de verano.

En la salida RET_VALse puede consultar si se han producido errores durante la ejecución de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_SYS_T":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
RET_VAL	Return	INT	INT	I, Q, M, D, L, P	Estado de la instrucción
OUT	Output	DTL	DT, DTL, LDT	I, Q, M, D, L, P	Fecha y hora de la CPU

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0000	Sin errores
8081	Valor de tiempo indicado en el parámetro OUT fuera del rango de valores admisible: • Con DT: mín. DT#1990-01-01-0:0:0, máx. DT#2200-12-31-23:59:59.999 • Con LDT: mín. LDT#1970-1-1-0:0:0.000000000, máx. LDT#2200-12-31-23:59:59.999999999 • Con DTL: mín. DTL#1970-01-01-00:00:00.0, máx. DTL#2200-12-31-23:59:59.999999999
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RD_LOC_T: Leer hora local

Descripción

La instrucción lee la hora local actual del reloj de la CPU y emite esta hora en la salida OUT. Para emitir la hora local se utilizan las indicaciones relativas a la zona horaria y al comienzo de los horarios de verano e invierno, que se han ajustado al configurar el reloj de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_LOC_T":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
RET_VAL	Return	INT	INT	I, Q, M, D, L, P	Estado de la instrucción
OUT	Output	DTL	DT, LDT, DTL	I, Q, M, D, L, P	Hora local

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0000	Ningún error
0001	Ningún error. La hora local se devuelve como horario de verano.
8080	No se puede leer la hora local.

Código de error* (W#16#....)	Descripción
8081	Valor de tiempo indicado en el parámetro OUT fuera del rango de valores admisible: • Con DT: mín. DT#1990-01-01-0:0:0, máx. DT#2089-12-31-23:59:59.999 • Con LDT: mín. LDT#1970-1-1-0:0:0.000000000, máx. LDT#2200-12-31-23:59:99.999999999 • Con DTL: mín. DTL#1970-01-01-00:00:00.0, máx. DTL#2200-12-31-23:59:59.999999999
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

WR_LOC_T: escribir hora local

Descripción

La instrucción "WR_LOC_T" ajusta la fecha y hora del reloj de la CPU. La indicación de fecha y hora se especifica como hora local en el parámetro de entrada LOCTIME.

El valor debe estar dentro del rango siguiente:

- Con DT: mín. DT#1990-01-01-00:00:00, máx. DT#2089-12-31-23:59:59.999
- Con DTL: mín. DTL#1970-01-01-00:00:00.0, máx. DTL#2200-12-31-23:59:59.999999999
- Con LDT: mín. LDT#1970-1-1-0:0:0.000000000, máx. LDT#2200-12-31-23:59:59.999999999

La granularidad en la indicación de la hora local y la hora del sistema es específica del producto y es como mínimo de un milisegundo. Al calcular la hora del sistema, los valores de entrada del parámetro LOCTIME inferiores a los valores soportados por la CPU se redondean.

En el parámetro de salida RET_VAL se puede consultar si han ocurrido errores durante la ejecución de la instrucción.

Uso de WR_LOC_T durante el cambio a horario de verano o a horario de invierno

- Cambio del horario de invierno al de verano
  Para lo expuesto a continuación se asume que la hora del cambio son las 02:00 de la mañana y que se adelantará una hora. Esto significa que la hora entre las 02:00:00:000000000 y las 02:59:59:999999999 no existirá.
  Si especifica tal indicación de tiempo en LOCTIME, se emitirá el código de error W#16#8089.
  DST es irrelevante.
- Cambio del horario de verano al de invierno
  Para lo expuesto a continuación se asume que la hora del cambio son las 03:00 de la mañana y que se retrasará una hora. Esto significa que la hora entre las 02:00:00:000000000 y las 02:59:59:999999999 existirá dos veces.
  Por tanto, en todas las indicaciones de tiempo para LOCTIME que estén entre las 02:00:00:000000000 y las 02:59:59:999999999, deberá definir si la indicación de tiempo se refiere al momento anterior o posterior al cambio de horario. Esto se realiza a través del parámetro DST:
  - Con DST=TRUE, la hora indicada se encuentra en la primera de estas dos horas; por tanto, todavía en el horario de verano.
  - Con DST=FALSE, la hora indicada se encuentra en la segunda de estas dos horas; por tanto, ya en el horario de invierno.

  En todas las indicaciones horarias de LOCTIME que están fuera de la hora duplicada, DST no tiene significado.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_LOC_T":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
LOCTIME	Input	DTL	DT, DTL, LDT	D, L P o constante	Hora local
DST	Input	BOOL	BOOL	I, Q, M, D, L, P, T, C o constante	**D**aylight **S**aving **T**ime Solo se evalúa durante la "hora duplicada" al cambiar al horario de invierno. • TRUE = horario de verano (primera hora) • FALSE = horario de invierno (segunda hora)
RET_VAL	Return	INT	INT	I, Q, M, D, L, P	Mensaje de error (véase "Parámetro RET_VAL")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0000	Ningún error.
8080	Valor no válido en el parámetro LOCTIME.
8081	Valor de tiempo indicado en el parámetro LOCTIME fuera del rango de valores admisible: • Con DT: mín. DT#1990-01-01-00:00:00, máx. DT#2089-12-31-23:59:59.999 • Con DTL: mín. DTL#1970-01-01-00:00:00.0, máx. DTL#2200-12-31-23:59:59.999999999 • Con LDT: mín. LDT#1970-1-1-0:0:0.000000000, máx. LDT#2200-12-31-23:59:59.999999999
8082**	Valor no válido para indicar el mes (byte 2 en formato DTL).
8083**	Valor no válido para indicar el día (byte 3 en formato DTL).
8084**	Valor no válido para indicar la hora (byte 5 en formato DTL).
8085**	Valor no válido para indicar el minuto (byte 6 en formato DTL).
8086**	Valor no válido para indicar el segundo (byte 7 en formato DTL).
8087**	Valor no válido para indicar el nanosegundo (bytes 8 a 11 en formato DTL).
8089	El valor de tiempo no existe (hora pasada al cambiar a horario de verano).
80B0	Fallo del reloj en tiempo real.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	
** Solo con indicación de hora local en el parámetro LOCTIME en formato DTL.	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

SET_TIMEZONE: Ajustar zona horaria

Descripción

La instrucción "SET_TIMEZONE" permite ajustar los parámetros para establecer la zona horaria local y cambiar al horario de verano/invierno.

Los ajustes realizados con la instrucción "SET_TIMEZONE" se corresponden con los ajustes asociados a la hora en las propiedades de la CPU. Para ejecutar la instrucción "SET_TIMEZONE" guarde los parámetros correspondientes en el tipo de datos de sistema TimeTransformationRule.

La hora local se calcula a partir de la hora del sistema con los ajustes para establecer la zona horaria y cambiar al horario de verano/invierno. La hora del sistema de la CPU es la hora UTC. Para la comunicación dentro del sistema se utiliza exclusivamente la hora del sistema.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_TIMEZONE":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, P o constante	Ajuste de los parámetros que están guardados en el parámetro TimeZone.
TimeZone	Input	Time TransformationRule (Página 4702)	D, L	En el tipo de datos de sistema TimeTransformationRule se guardan los parámetros para establecer la zona horaria local y cambiar al horario de verano/ invierno.
DONE	Output	BOOL	I, Q, M, D, L, P	Parámetros de estado: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L, P	Parámetros de estado: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No es posible iniciar una nueva petición.
ERROR	Output	BOOL	I, Q, M, D, L, P	Parámetros de estado: • 0: Sin errores • 1: Con errores
STATUS	Output	DINT, DWORD, UDINT, WORD	I, Q, M, D, L, P	En el parámetro STATUS se visualiza información de estado y error detallada. El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#....)	Descripción
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
808x	Entrada errónea en el x parámetro del tipo de datos de sistema TimeTransformationRule. Ejemplo: el código de error 8084 indica una entrada errónea en el 4.º parámetro (DaylightStartWeek). Aquí solo son válidos los valores de 1 a 5.
80C3	Error temporal de recursos: en este momento la CPU está procesando el máximo posible de llamadas de bloque simultáneas. No se puede ejecutar "SET_TIMEZONE" antes de haber finalizado como mínimo una llamada de bloque.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

TimeTransformationRule

Descripción

Con el tipo de datos de sistema TimeTransformationRule se definen los puntos temporales de cambio a horario de verano y a horario de invierno.

Para crear la estructura TimeTransformationRule se dispone de las siguientes posibilidades:

- Cree un nuevo bloque de datos y seleccione como tipo TimeTransformationRule.
- Introduzca como tipo de datos TimeTransformationRule en un bloque de datos existente o en la interfaz local de un bloque de función.

La estructura está organizada de la manera siguiente:

Nombre		Tipo de datos	Descripción
TimeTransformationRule		STRUCT	
	Bias	INT	Diferencia de tiempo entre la hora local y la hora del sistema (UTC) en minutos.
	DaylightBias	INT	Desfase entre horario de invierno y horario de verano en minutos.
	Indicación del instante de cambio a horario de verano. Las siguientes indicaciones horarias hacen referencia siempre a la hora local.		
	DaylightStartMonth	USINT	Mes del cambio a horario de verano: 1 = enero 2 = febrero 3 = marzo ... 12 = diciembre
	DaylightStartWeek	USINT	Semana del cambio a horario de verano. 1 = primera aparición del día de la semana en el mes ... 5 = última aparición del día de la semana en el mes
	DaylightStartWeekday	USINT	Día de la semana del cambio a horario de verano: 1 = domingo
	DaylightStartHour	USINT	Hora del cambio a horario de verano
	DaylightStartMinute	USINT	Minuto del cambio a horario de verano
	Indicación del instante de cambio a horario de invierno. Las siguientes indicaciones horarias hacen referencia siempre a la hora local.		
	StandardStartMonth	USINT	Mes del cambio a horario de invierno: 1 = enero 2 = febrero 3 = marzo ... 12 = diciembre

Nombre		Tipo de datos	Descripción
	StandardStartWeek	USINT	Semana del cambio a horario de invierno: 1 = primera aparición del día de la semana en el mes ... 5 = última aparición del día de la semana en el mes
	StandardStartWeekday	USINT	Día de la semana del cambio a horario de invierno: 1 = domingo
	StandardStartHour	USINT	Hora del cambio a horario de invierno
	StandardStartMinute	USINT	Minuto del cambio a horario de invierno
	TimeZoneName	STRING[80]	Nombre de la zona horaria, por ejemplo: "(GMT+01:00) Amsterdam, Berlin, Bern, Rome, Stockholm, Vienna"

SNC_RTCB: Sincronizar relojes esclavos

Definición: Sincronización de relojes esclavos

La sincronización de relojes esclavos consiste en la transmisión de la fecha y la hora del reloj maestro de un segmento de bus a todos los relojes esclavos de ese segmento de bus.

Descripción

La instrucción sincroniza todos los relojes esclavos existentes en un segmento de bus independientemente del intervalo de sincronización parametrizado. Para una correcta sincronización es imprescindible que "SNC_RTCB" " se llame en una CPU cuyo reloj de tiempo real haya sido parametrizado como reloj maestro para al menos un segmento de bus.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SNC_RTCB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No se ha producido ningún error durante la sincronización.
0001	El reloj presente no ha sido parametrizado como reloj maestro para ningún segmento de bus.

Código de error* (W#16#...)	Explicación
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

TIME_TCK: Leer hora del sistema

Descripción

Con la instrucción "TIME_TCK" se lee la hora del sistema de la CPU. La hora del sistema es un contador de tiempo que contabiliza desde 0 a un máximo de 2147483647 ms. En caso de desbordamiento de la hora del sistema, se reinicia el recuento desde "0". El incremento de tiempo y la exactitud de la hora del sistema es de 1 ms. El único factor que influye en la hora del sistema son los estados operativos de la CPU. La hora del sistema se puede usar, por ejemplo, para medir la duración de procesos obteniendo la diferencia entre los resultados de dos llamadas de "TIME_TCK". Esta instrucción no ofrece información de error.

La tabla siguiente resume los cambios que experimenta la hora del sistema en función de los estados operativos de la CPU.

Estado operativo	La hora del sistema ...
Arranque	... se actualiza de modo permanente.
RUN	
STOP	... se detiene y mantiene el valor actual.
Rearranque completo (en caliente)	... se borra y vuelve a empezar desde "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TIME_TCK":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	TIME	I, Q, M, D, L	El parámetro RET_VAL contiene la hora del sistema leída, en el rango de 0 a 2^{31} -1 ms.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RTM: Contador de horas de funcionamiento

Descripción

La instrucción permite inicializar, arrancar, parar y leer un contador de horas de funcionamiento de 32 bits de la CPU.

Tenga en cuenta que el contador de horas de funcionamiento también se puede detener o reiniciar durante la ejecución del programa de usuario, por lo que los valores guardados pueden verse alterados.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RTM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NR	Input	RTM	I, Q, M, D, L o constante	Número del contador de horas de funcionamiento La numeración comienza a partir de 0. Para conocer el número de contadores de horas de funcionamiento de su CPU, consulte los Datos técnicos.
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición: • 0: Leer (tras ello, el estado se registra en CQ y el valor actual en CV). Si el contador de horas de funcionamiento trabaja durante más de (2 elevado a 31) - 1 horas, se detiene en el último valor representable y envía el aviso de error "Desbordamiento". • 1: Arrancar (con el último valor registrado) • 2: Parar • 4: Inicializar (al valor especificado en PV) • 5: Inicializar (al valor especificado en PV) y arrancar • 6: Inicializar (al valor especificado en PV) y detener
PV	Input	DINT	I, Q, M, D, L o constante	Nuevo valor para el contador de horas de funcionamiento
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
CQ	Output	BOOL	I, Q, M, D, L	Estado del contador de horas de funcionamiento (1: en marcha)
CV	Output	DINT	I, Q, M, D, L	Valor actual del contador de horas de funcionamiento

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Sin errores
8080	Número erróneo del contador de horas de funcionamiento
8081	Se ha transmitido un valor negativo al parámetro PV.
8082	Desbordamiento del contador de horas de funcionamiento
8091	El parámetro de entrada MODE contiene un valor no permitido.
Información de error general	Consulte también: Evaluación de errores mediante GET_ERR_ID (Página 4460)

String + Char

S_MOVE: Desplazar cadena de caracteres

Descripción

La instrucción desplaza el contenido de una cadena de caracteres (W)STRING. La cadena de caracteres del parámetro de entrada IN se copia en el parámetro de salida OUT.

Las variables del parámetro de entrada IN y de la salida OUT deben ser del mismo tipo.

Para copiar variables del tipo de datos ARRAYse pueden utilizar las instrucciones "MOVE_BLK" y "UMOVE_BLK".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "S_MOVE":

Parámetros	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	STRING	STRING, WSTRING	D, L o constante	Cadena de caracteres de origen
OUT	Output	STRING	STRING, WSTRING	D, L o constante	Cadena de caracteres de destino

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

S_COMP: Comparar cadenas de caracteres

Descripción

Esta instrucción compara los contenidos de dos variables en el formato STRING y devuelve el resultado de la comparación como valor de retorno. Las variables que se deben comparar se interconectan en las entradas IN1 y IN2 . A los parámetros de entrada solo pueden asignárseles variables definidas como simbólicas.

Seleccione la condición de comparación mediante el cuadro de instrucción. Si se cumple la condición de comparación (p. ej. mayor o igual), el estado lógico del parámetro de salida OUT se pone a "1".

Pueden usarse las siguientes condiciones de comparación:

Símbolo	Descripción
EQ	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es igual que la del parámetro IN2 .
NE	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es distinta a la del parámetro IN2 .
GT (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es mayor que la del parámetro IN2 .
LT (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es menor que la del parámetro IN2 .
GE (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es mayor o igual que la del parámetro IN2 .
LE (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es menor o igual que la del parámetro IN2 .
(1) Los caracteres se comparan empezando por la izquierda según su codificación ASCII (p. ej. 'a' es mayor que 'A'). El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la cadena de caracteres más larga se considera mayor.	

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "S_COMP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING*	D, L o constante	Variable de entrada en el formato STRING
IN2	Input	STRING*	D, L o constante	Variable de entrada en el formato STRING
OUT	Output	BOOL	I, Q, M, D, L	Resultado de la comparación
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

S_CONV: Convertir cadena de caracteres

Descripción

Esta instrucción permite convertir el valor de la entrada INal formato de datos indicado en la salida OUT. El formato de salida de la conversión se determina seleccionando un tipo de datos para el parámetro de salida OUT.

Son posibles las conversiones siguientes:

- Conversión de una cadena de caracteres (STRING) a:
  - Un valor numérico (entero o número en coma flotante)
    La conversión se ejecuta para todos los caracteres de la cadena de caracteres indicada en el parámetro de entrada IN. Los caracteres permitidos son las cifras de "0" a "9", el punto decimal, así como los signos más (+) y menos (–). El primer carácter de la cadena puede ser una cifra válida o un signo. Se ignoran los espacios iniciales y las representaciones exponenciales.
  - Un carácter (CHAR)
    Al convertir una cadena de caracteres en un carácter (CHAR), el primer carácter de la cadena de caracteres se transfiere al parámetro OUT.
- Conversión de un valor numérico o de un carácter a una cadena de caracteres (STRING):
  - El formato del valor numérico que se debe convertir se determina seleccionando un tipo de datos para la entrada IN. En la salida OUTdebe especificarse una variable válida del tipo de datos STRING. La longitud de la cadena de caracteres tras la conversión depende del valor de la entrada IN.
  - El resultado de la conversión se guarda a partir del tercer byte de la cadena de caracteres. En el primer byte de la cadena de caracteres se registra la longitud máxima, y en el segundo, la longitud real de la cadena. Los valores numéricos positivos se emiten sin signo.
  - En una conversión de cero (p. ej. INT_TO_STRING(0)), la cadena de caracteres resultante tiene 6 caracteres de longitud.
  - En caso de convertir un valor numérico, los primeros caracteres de la cadena se rellenan con espacios. El número de espacios varía en función de la longitud del valor numérico.
  - En caso de convertir un carácter (CHAR), el carácter se escribe en la primera posición de la cadena de caracteres.

Nota

Notación exponencial en la conversión de números en coma flotante

No utilice notación exponencial ("e" o "E") en la conversión de números en coma flotante con la instrucción "S_CONV". Para la conversión de números en coma flotante con notación exponencial, utilice en su lugar la instrucción "STRG_VAL (Página 4709)". El parámetro FORMAT de la instrucción permite seleccionar la notación exponencial como formato de entrada.

Parámetros

Las tablas siguientes muestran los parámetros de la instrucción "S_CONV" correspondientes a las posibles conversiones:

Tabla 9-40 Parámetros cuando se convierte una cadena de caracteres en un valor numérico:

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Valor que se va a convertir
OUT	Output	CHAR, USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL	I, Q, M, D, L	Resultado de la conversión

Tabla 9-41 Parámetros cuando se convierte un valor numérico o un carácter en una cadena de caracteres:

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	CHAR, USINT, UINT, UDINT, ULINT, SINT, INT, DINT, LINT, REAL, LREAL	I, Q, M, D, L o constante	Valor que se va a convertir
OUT	Output	STRING	D, L	Resultado de la conversión

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

STRG_VAL: Convertir cadena de caracteres en valor numérico

Descripción

La instrucción "STRG_VAL" convierte una cadena de caracteres numéricos en un entero o un número en coma flotante:

- La cadena de caracteres que se va a convertir se especifica en el parámetro de entrada IN.
- El formato del valor de salida se determina seleccionando un tipo de datos para el parámetro de salida OUT.

Los caracteres permitidos para la conversión son las cifras de "0" a "9", el punto decimal, la coma decimal, las notaciones "E" y "e", así como los signos más (+) y menos (–). La conversión puede interrumpirse si se detectan caracteres no válidos.

La instrucción "STRG_VAL" no está soportada en el lenguaje de programación SCL.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "STRG_VAL":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	STRING	STRING, WSTRING	D, L o constante	Cadena de caracteres numéricos que se va a convertir
FORMAT	Input	WORD	WORD	I, Q, M, D, L, P o constante	Formato de salida de los caracteres
P	Input	UINT	UINT	I, Q, M, D, L, P o constante	Remite al primer carácter que debe convertirse (primer carácter = 1, el valor "0" o un valor > longitud del string no es válido)
OUT	Output	USINT, SINT, UINT, INT, UDINT, DINT, REAL, LREAL	USINT, SINT, UINT, INT, UDINT, DINT, ULINT, LINT, REAL, LREAL	I, Q, M, D, L, P	Resultado de la conversión

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro FORMAT

Con el parámetro FORMATse especifica cómo deben interpretarse los caracteres de una cadena de caracteres. Con la instrucción "STRG_VAL" también es posible convertir y representar valores exponenciales.

La tabla siguiente muestra los valores posibles del parámetro FORMATy su significado:

Valor (W#16#....)	Notación	Representación decimal
0000	Fracción decimal	"."
0001		","
0002	Exponencial	"."
0003		","
0004 a FFFF	Valores no válidos	

Parámetro P

La conversión comienza a partir del carácter cuya posición se ha especificado en el parámetro P. Si, por ejemplo, se ha especificado el valor "1" en el parámetro P, la conversión comenzará a partir del primer carácter de la cadena de caracteres indicada.

Ejemplo

La tabla siguiente muestra ejemplos de conversión de una cadena de caracteres en un valor numérico:

IN (STRING)	FORMAT (W#16#....)	OUT (Tipo de datos)	OUT (Valor)	Estado ENO
'123'	0000	INT/DINT	123	1
'-00456'	0000	INT/DINT	-456	1
'123.45'	0000	INT/DINT	123	1
'+2345'	0000	INT/DINT	2345	1
'00123AB'	0000	INT/DINT	123	1
'123'	0000	REAL	123.0	1
'-00456'	0001	REAL	-456.0	1
'+00456'	0001	REAL	456.0	1
'123.45'	0000	REAL	123.45	1
'123.45'	0001	REAL	12345.0	1
'123,45'	0000	REAL	12345.0	1
'123,45'	0001	REAL	123.45	1
'.00123AB'	0001	REAL	123.0	1
'1.23e-4'	0000	REAL	1.23	1
'1.23E-4'	0000	REAL	1.23	1
'1.23E-4'	0002	REAL	1.23E-4	1
'12,345.67'	0000	REAL	12345.67	1
'12,345.67'	0001	REAL	12.345	1
'3.4e39'	0002	REAL	W#16#7F800000	1
'-3.4e39'	0002	REAL	W#16#FF800000	1
'1.1754943e-38'	0002	REAL	0.0	1
'12345'	-/-	SINT	0	0
'A123'	-/-	-/-	0	0
''	-/-	-/-	0	0
'++123'	-/-	-/-	0	0
'+-123'	-/-	-/-	0	0

VAL_STRG: Convertir valor numérico en cadena de caracteres

Descripción

La instrucción "VAL_STRG" convierte un valor numérico en una cadena de caracteres.

- El valor que se va a convertir se especifica en el parámetro de entrada IN. El formato del valor numérico se determina seleccionando un tipo de datos.
- El resultado de la conversión se consulta en el parámetro de salida OUT.

Los caracteres permitidos para la conversión son las cifras de "0" a "9", el punto decimal, la coma decimal, las notaciones "E" y "e", así como los signos más (+) y menos (–). La conversión puede interrumpirse si se detectan caracteres no válidos.

La instrucción "VAL_STRG" no está soportada en el lenguaje de programación SCL.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "VAL_STRG":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN	Input	USINT, SINT, UINT, INT, UDINT, DINT, REAL, LREAL	USINT, SINT, UINT, INT, UDINT, DINT, ULINT, LINT, REAL, LREAL	I, Q, M, D, L, P o constante	Valor que se va a convertir
SIZE	Input	USINT	USINT	I, Q, M, D, L, P o constante	Número de posiciones de carácter
PREC	Input	USINT	USINT	I, Q, M, D, L, P o constante	Número de decimales
FORMAT	Input	WORD	WORD	I, Q, M, D, L, P o constante	Formato de salida de los caracteres
P	InOut	UINT	UINT	I, Q, M, D, L, P o constante	Carácter a partir del cual se escribe el resultado.
OUT	Output	STRING	STRING, WSTRING	D, L	Resultado de la conversión

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro P

El parámetro P especifica a partir de qué carácter de la cadena de caracteres se escribe el resultado. Si, por ejemplo, se ha especificado el valor "2" en el parámetro P, el valor convertido se guardará a partir del segundo carácter de la cadena de caracteres.

Parámetros SIZE y P

El parámetro SIZEdetermina en cuántos caracteres de la cadena de caracteres se debe escribir. A este efecto, se cuenta a partir del carácter especificado en el parámetro P. Si el valor emitido es más corto que la longitud especificada, el resultado se escribe justificado a la derecha en la cadena de caracteres. Las posiciones de carácter vacías se rellenan con espacios.

Parámetro FORMAT

Con el parámetro FORMATse especifica cómo debe interpretarse el valor numérico durante la conversión y escribirse en la cadena de caracteres. En el parámetro FORMATsolo pueden especificarse variables del tipo de datos USINT.

La tabla siguiente muestra los valores posibles del parámetro FORMATy su significado:

Valor (W#16#....)	Notación	Signo	Representación decimal
0000	Fracción decimal	"-"	"."
0001			","
0002	Exponencial		"."
0003			","
0004	Fracción decimal	"+" y "-"	"."
0005			","
0006	Exponencial		"."
0007			","
0008 a FFFF	Valores no válidos		

Parámetro PREC

Con el parámetro PRECse define el número de decimales al convertir números en coma flotante. Como máximo se admite una precisión de 7 cifras para valores numéricos del tipo de datos REAL. Si el valor que se va a convertir es un entero, es necesario definir con el parámetro PRECla posición en la que debe colocarse un punto decimal.

Ejemplo

La tabla siguiente muestra ejemplos de conversión de valores numéricos en una cadena de caracteres.

IN(Valor)	IN (Tipo de datos)	P	SIZE	FORMAT(W#16#....)	PREC	OUT (STRING)	Estado ENO
123	UINT	16	10	0000	0	xxxxxxx123 C	1
0	UINT	16	10	0000	2	xxxxxx0.00 C	1
12345678	UDINT	16	10	0000	3	x12345.678 C	1
12345678	UDINT	16	10	0001	3	x12345.678 C	1
123	INT	16	10	0004	0	xxxxxx+123 C	1
-123	INT	16	10	0004	0	xxxxxx-123 C	1
-0.00123	REAL	16	10	0004	4	xxx-0.0012 C	1
-0.00123	REAL	16	10	0006	4	-1.2300E-3 C	1
-Inf [1]	REAL	16	10	-/-	4	xxxxxx-INF C	0
+Inf [2]	REAL	16	10	-/-	4	xxxxxx+INF C	0
NaN [3]	REAL	16	10	-/-	4	xxxxxxxNaN C	0
12345678	UDINT	16	6	-/-	3	xxxxxxxxxx C	0

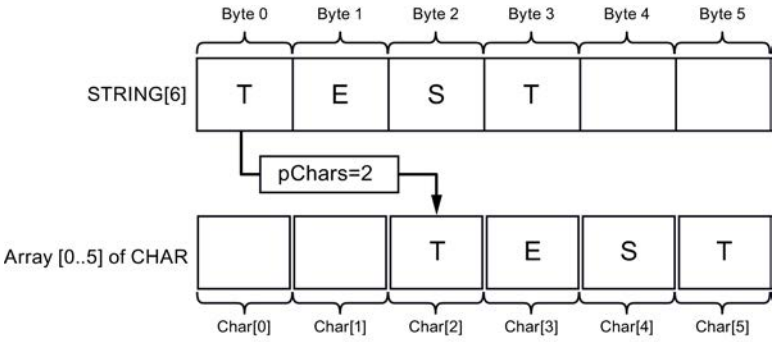
"x" representa espacios
[1]-Inf: número en coma flotante que representa un valor infinito negativo.
[2] +Inf: número en coma flotante que representa un valor infinito positivo.
[3] NaN: Valor devuelto como resultado de operaciones aritméticas no válidas.

Strg_TO_Chars: Convertir cadena de caracteres en Array of CHAR

Descripción

Con esta instrucción se copia una cadena de caracteres (W)STRING en una estructura del tipo de datos Array of CHAR o Array of BYTE. Para el proceso de copia solo son válidos caracteres ASCII.

- La cadena de caracteres se especifica en el parámetro de entrada STRG.
- Los caracteres se escriben en el parámetro CHARS en un tipo de datos Array of CHAR / BYTE .
  - El número de caracteres en la estructura Array of CHAR / BYTE debe ser por lo menos la cantidad de caracteres que se deben copiar de la cadena de caracteres STRING.
  - Si la estructura Array of CHAR / BYTE contiene menos caracteres que la cadena, los caracteres se escriben hasta alcanzar la longitud máxima de Array of CHAR / BYTE.
  - Si la cadena de caracteres contiene un carácter "$00", el proceso de copia solo se ejecuta hasta la posición correspondiente.
  - El número de caracteres copiados se visualiza en el parámetro CNT.
- A través del parámetro PCHARS se indica a partir de qué carácter se debe escribir en la estructura Array of CHAR / BYTE.
  - Ejemplo: si el proceso de escritura debe empezar a partir del tercer carácter, aplique el valor "2" en el parámetro PCHARS:



  - Por defecto ("0") se utiliza el límite inferior del índice del Array (p. ej. CHAR[0] con Array [0..5] of CHAR).

Nota

Uso de la instrucción con S7-1200 V2.0

El S7-1200 hasta la versión 2.0 solo soporta Array [0 .. n] of CHAR / BYTE. Los límites de índice negativos (p. ej. Array [-3..2] of CHAR) no son admisibles. El software no comprueba esta restricción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Strg_TO_Chars":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
STRG	Input	STRING	STRING, WSTRING	D, L o constante	Origen del proceso de copia Cadena de caracteres con una longitud máxima de 254 caracteres.
PCHARS	Input	DINT	DINT	I, Q, M, D, L, P o constante	Posición en la estructura Array of (W)CHAR / BYTE a partir de la cual se escriben los caracteres de la cadena.
CHARS	InOut	VARIANT	VARIANT	D, L	Destino del proceso de copia Los caracteres se pueden copiar en una estructura del tipo de datos Array of (W)CHAR o Array of BYTE.
CNT	Output	UINT	UINT	I, Q, M, D, L, P	Número de caracteres copiados.

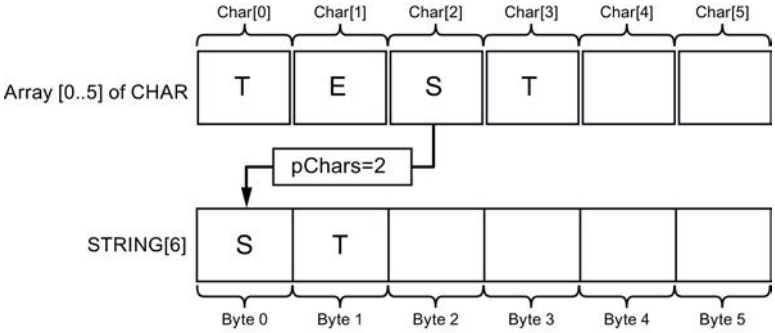
Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Chars_TO_Strg: Convertir Array of CHAR en cadena de caracteres

Descripción

Con la instrucción "Chars_TO_Strg" se copian caracteres de una estructura del tipo de datos Array of CHAR o Array of BYTE en una cadena de caracteres STRING. Para el proceso de copia solo son válidos caracteres ASCII.

- Introduzca en el parámetro de entrada CHARS los caracteres de la estructura Array of (W)CHAR / BYTE que se deben copiar en una cadena de caracteres.
- Los caracteres se escriben en el parámetro STRG en un tipo de datos STRING.
  - El número de caracteres de la cadena STRING debe ser por lo menos la cantidad de caracteres que se deben copiar del campo Array of CHAR.
  - Si la cadena tiene menos caracteres que la estructura Array of CHAR / BYTE, los caracteres se escriben hasta alcanzar la longitud máxima de la cadena de caracteres.
  - Si la estructura Array of CHAR / BYTE contiene un carácter "$00", el proceso de copia solo se ejecuta hasta la posición correspondiente.
- A través del parámetro PCHARS se indica a partir de qué posición de la estructura Array of CHAR / BYTE se deben copiar los caracteres.
  - Ejemplo: si el proceso de copia debe empezar a partir del tercer carácter de la estructura, aplique el valor "2" en el parámetro PCHARS:



  - Por defecto ("0") se utiliza el límite inferior del índice del Array (p. ej. CHAR[0] con Array [0..5] of CHAR).
  - Si en el parámetro PCHARS se indica un índice que no está contenido en el origen de la copia (p. ej. "7" con Array [0..5] of CHAR), la instrucción no se ejecuta.

Nota

Uso de la instrucción con S7-1200 V2.0

El S7-1200 hasta la versión 2.0 solo soporta Array [0 .. n] of CHAR / BYTE. Los límites de índice negativos (p. ej. Array [-3..2] of CHAR) no son admisibles. El software no comprueba esta restricción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Chars_TO_Strg":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
CHARS	Input	VARIANT	VARIANT	D, L	Origen del proceso de copia Estructura Array of CHAR / BYTE de la que se copian los caracteres.
PCHARS	Input	DINT	DINT	I, Q, M, D, L, P o constante	Posición en la estructura Array of (W)CHAR / Array of BYTE a partir de la cual se copian los caracteres.
CNT	Output	UINT	UINT	I, Q, M, D, L, P	Número de caracteres que se van a copiar. Con "0" se copian todos los caracteres.
STRG	Output	STRING	STRING, WSTRING	D, L	Destino del proceso de copia Cadena de caracteres

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

MAX_LEN: Determinar la longitud máxima de una cadena de caracteres

Descripción

Una variable del tipo de datos STRING contiene dos longitudes: la longitud máxima y la longitud actual (es decir, el número de caracteres válidos actualmente).

- La longitud máxima de la cadena de caracteres de cada variable se especifica entre corchetes en la palabra clave STRING. El número de bytes ocupados por una cadena de caracteres excede en 2 la longitud máxima.
- La longitud actual representa el número de posiciones de carácter realmente ocupadas. La longitud actual es menor o igual que la longitud máxima.

Con la instrucción "MAX_LEN" se consulta la longitud máxima de la cadena de caracteres especificada en el parámetro de entrada IN y la devuelve como valor numérico en el parámetro de salida OUT.

Si ocurren errores durante la ejecución de la instrucción, se devuelve una cadena vacía.

Nota

Leer la longitud actual

La instrucción "LEN (Página 4729)" permite leer la longitud actual de una cadena de caracteres.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MAX_LEN":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Cadena de caracteres
OUT	Return	INT	I, Q, M, D, L, P	Número máximo de caracteres

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Join: Unir varias cadenas de caracteres

Descripción

La instrucción "JOIN" une varias cadenas de caracteres y crea un array.

Para convertir varias cadenas de caracteres en una, la instrucción ofrece las funciones siguientes:

- Selección de formato
  Con el primer bit del parámetro Mode se puede seleccionar si para el array va a utilizarse el formato CSV o el FSR.
  - Con CSV (Comma Separated Values), los contenidos de las cadenas de caracteres de origen se escriben uno tras otro y separados por un separador. Ejemplo:
    `1963,Miller,John,CitynameA,Roadname`
    `1974,Jackson,Peter,CitynameB,VeryLongRoadname`
  - Con FSR (Fixed Size Records) se define un número determinado de caracteres para cada valor (cada cadena de caracteres de origen). Si se necesitan menos caracteres, estos se rellenan con el delimintador. Si se necesitan más caracteres, se recortan los textos. Ejemplo:
    `1963Miller,,,,,,,John,,,,,,CitynameA,,,,Roadname,,,,,,`
    `1974Jackson,,,,,,Peter,,,,,CitynameB,,,,VeryLongRoadna`
- Selección de separadores de las cadenas de caracteres de origen
  Con el parámetro RecSeparator puede decidir qué separador se utilizará para cada una de las cadenas de caracteres. La elección del carácter separador debería depender del contenido de las cadenas de caracteres de entrada en el parámetro SrcStruct. Si las cadenas de caracteres de entrada contienen una coma dentro de una cadena, por ejemplo, la coma no debe utilizarse como separador. El tipo de datos que se utilice para el separador deberá corresponderse con el del array de destino en el parámetro DstArray, para que los separadores puedan escribirse en el array de destino.
- Selección del separador para el final de todas las cadenas de caracteres
  Con el tercer bit del parámetro Mode se selecciona si en el array de destino (parámetro DstArray) debe escribirse un carácter adicional como separador al final de la cadena. En el parámetro EndSeparator puede indicar el carácter que se utilizará como separador. Asegúrese de usar otro carácter del indicado en el parámetro RecSeparator (separador de cadenas de caracteres individuales). De lo contrario, en caso de una reconversión realizada con la instrucción "SPLIT", obtendrá resultados indeseados si los dos separadores no pueden discernirse.

- **Selección de las cadenas de caracteres de origen**
  Las cadenas de caracteres de origen se indican en el parámetro SrcStruct. Como tipo de datos se puede utilizar Array of STRING o Array of WSTRING o cualquier estructura que contenga exclusivamente el tipo de datosSTRING o WSTRING. Esto también es válido para tipos de datos de usuario o estructuras anidadas. Mientras contengan exclusivamente el tipo de datos STRING o WSTRING, pueden utilizarse.
- **Se indica el número de cadenas de caracteres unidas**
  Si en el parámetro SrcStruct (cadenas de caracteres de origen) se utiliza un Array of STRING o Array of WSTRING (y no estructuras anidadas), mediante el parámetro Count se puede indicar el número de cadenas de caracteres de origen que se han unido para formar una cadena larga. Si en el parámetro SrcStruct se utiliza un tipo de datos distinto a Array of (W)STRING, se ignora el parámetro Count. Esto permite unir solo una parte de una array de mayor tamaño.
- **Selección del área de destino para escribir el array**
  En el parámetro DstArray se utiliza el tipo de datos Array of (W)CHAR. En este parámetro no es posible utilizar los tipos de datos STRING o WSTRING, pues en el caso de STRING la longitud estaría limitada a 256 caracteres.
- **Índice de la posición en el array (parámetro de destino DestArray)**
  La conversión comienza en esta posición y la instrucción indica mediante el parámetro Position en qué posición finalizó la conversión. Esto permite realizar llamadas consecutivas de la instrucción para rellenar el array.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "JOIN":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Mode	Input	DWORD	I, Q, M, D, L o constante	Especifica cómo se realiza la fusión en una cadena de caracteres (véase "Parámetro Mode").
RecSeparator	Input	VARIANT	I, Q, M, D, L	Separadores de las cadenas de caracteres de origen • Con CSV: Carácter que se utiliza como separador de las distintas cadenas de caracteres. • Con FSR: Carácter que se utiliza como carácter de relleno de las distintas cadenas de caracteres.
EndSeparator	Input	VARIANT	I, Q, M, D, L	Separador para el final de la conversión Separador que se escribe al final de los caracteres en caso de que para el parámetro Mode se haya puesto el bit 3 = 1.
SrcStruct	Input	VARIANT	I, Q, M, D, L	Puntero a las cadenas de caracteres de origen.
Count	Input	UDINT	I, Q, M, D, L	Número de cadenas de caracteres que se han fusionado. El parámetro Count puede utilizarse únicamente si en el parámetro SrcStruct se ha utilizado un Array of (W)STRING.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DestArray	InOut	VARIANT	I, Q, M, D, L	Área en la que se escriben los caracteres después de la conversión. Utilice en el parámetro DestArray el tipo de datos Array [0 .. x] of CHAR/WCHAR. La longitud (x) del array se selecciona en el parámetro SrcStruct según la longitud de las cadenas de caracteres de origen.
Position	InOut	UDINT	I, Q, M, D, L	Índice de la posición en la cadena de caracteres total
Ret_Val	Return	UINT	I, Q, M, D, L	Estado de la instrucción (véase la tabla "Parámetros RET_VAL")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro Mode

Bit	Valor de bit "0"	Valor de bit "1"	Descripción
0	Formato CSV (Comma Separated Values)	Formato FSR (Fixed Size Record)	Selección del formato: • Con CSV, las cadenas de caracteres de origen se separan con un separador en el array de destino. • Con FSR se escriben en el array de destino las cadenas de caracteres de origen con los caracteres de relleno definidos en el parámetro RecSeparator.
1	-	-	No relevante para la instrucción "JOIN".
2	-	-	Reservado (valor de bit irrelevante)
3	No escribir ningún separador adicional.	Al final de los caracteres leídos escribir el carácter que se definió mediante el parámetro EndSeparator.	Se selecciona si en el array (parámetro DestArray) debe escribirse un carácter adicional como separador al final de los caracteres.
4	-	-	No relevante para la instrucción "JOIN".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	Ningún error.
0001	Los caracteres adicionales se ignoran.
8190	No está soportada la selección en el parámetro Mode.
8x20	Cadenas de caracteres de origen no válidas.
8x50	VARIANT señala a un contenido vacío.
8x52	VARIANT señala a un bloque que no se ha creado con acceso optimizado.

Código de error* (W#16#...)	Explicación
8x53	VARIANT señala a una estructura de datos demasiado corta.
8x54	Tipo de datos no válido
8882	El valor en el parámetro Position está fuera de la cadena de caracteres de origen.
8x84	Se han encontrado caracteres adicionales.
80B4	Tipos de datos diferentes en los parámetros SrcStruct (origen) y DestArray (destino) o en los separadores (parámetros RecSeparator y EndSeparator).
80B5	Desbordamiento del búfer en la instrucción. En el parámetro DestArray se emiten cadenas de caracteres incompletas.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)

* Observe lo siguiente respecto a los códigos de error:

- Los códigos de error se pueden representar en el editor de programación como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".
- La "x" en la segunda posición de los códigos de error listados indica el parámetro que ha causado el error.
  Ejemplo: Código de error 8352 hex = error ocurrido en el 3.er parámetro (EndSeparator), consulte la tabla de parámetros.
- Si el error no puede atribuirse a un parámetro en particular, se devuelve un "0".
  Ejemplo: Para el separador (parámetro RecSeparator) se utiliza el tipo de datos CHAR. Para el array del parámetro DestArray se utiliza el tipo de datos WCHAR. En este caso se devuelve el código de error 80B4.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Split: Dividir un array de caracteres en varias cadenas de caracteres

Descripción

La instrucción "SPLIT" convierte un array (Array of CHAR / WCHAR) en varias cadenas de caracteres separadas (Array of STRING / WSTRING o estructura).

Para la conversión del array en varias cadenas de caracteres introduzca la información siguiente:

- Selección del array que se va a leer
  El array que se va a leer se especifica en el parámetro SrcArray. En los parámetros siguientes, asegúrese de que coincidan los tipos de datos utilizados para los parámetros de entrada y salida. Si en el parámetro SrcArray se utiliza por ejemplo un array del tipo CHAR, tanto los separadores (Rec-/EndSeparator) como la estructura del parámetro DestStruct deberán contener únicamente cadenas del tipo STRING.
- Selección de formato del array que se va a leer
  Con el primer bit del parámetro Mode se indica si el array que se va a leer tiene el formato CSV o FSR (consulte la definición en la instrucción "JOIN (Página 4718)").

- **Deslimitador utilizado para el array que se va a leer**
  - Si el array que se va a leer tiene el formato CSV, indique en el parámetro RecSeparator qué separador se ha utilizado.
  - Si el array que se va a leer tiene el formato FSR, indique en el parámetro RecSeparator qué carácter de relleno se ha utilizado.
- **Separador utilizado para el final de la cadena de caracteres total**
  En el parámetro EndSeparator se especifica el separador a partir del cual se deja de leer el array. La instrucción "SPLIT" termina en esta posición y emite las cadenas de caracteres encontradas. En este caso debe tenerse en cuenta que el separador EndSeparator se evalúa con preferencia con respecto al separador RecSeparator. Si el separador del parámetro EndSeparator se utiliza en una cadena de caracteres que se va a leer (entre dos separadores RecSeparator), dejarán de emitirse los contenidos que se encuentren a partir del separador EndSeparator.
- **Indicación de la posición a partir de la cual se lee el array**
  La conversión comienza en esta posición del array y la instrucción indica mediante el parámetro Position en qué posición finalizó la conversión. Esto permite realizar llamadas consecutivas de la instrucción para rellenar distintas cadenas de caracteres en el parámetro DestStruct.
- **Se indica el número de cadenas de caracteres leídas**
  Si en el parámetro DestStruct se utiliza un Array of STRING, puede indicarse el número de cadenas de caracteres leídas a través del parámetro Count. Solo se cuentan las cadenas de caracteres con contenido. Si en el parámetro DestStruct se utiliza un tipo de datos distinto a Array of STRING, en el parámetro Count se indica "0".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "SPLIT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Mode	Input	DWord	I, Q, M, D, L o constante	Especifica cómo se realiza la segmentación en varias cadenas de caracteres (véase "Parámetro Mode").
RecSeparator	Input	Variant	I, Q, M, D, L	Carácter de separación o de relleno • Con CSV: carácter que se ha utilizado en el array que se va a leer para identificar cada una de las cadenas de caracteres. • Con FSR: carácter que se ha utilizado como relleno en el array que se va a leer.
EndSeparator	Input	Variant	I, Q, M, D, L	Separador mediante el cual se ha definido el final de la cadena de caracteres total en el array que se va a leer.
SrcArray	Input	Variant	I, Q, M, D, L	Puntero al array que se va a leer (Array of CHAR/WCHAR)
DestStruct	InOut	Variant	I, Q, M, D, L	Estructura que contiene las cadenas de caracteres convertidas (Array of STRING / WSTRING).

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Position	InOut	UDInt	I, Q, M, D, L	Posición a partir de la cual se lee el array en el parámetro SrcArray.
Ret_Val	Return	Int	I, Q, M, D, L	Resultado de la ejecución de la instrucción / código de error (consulte la tabla "Parámetro Ret_Val")
Count	Output	UDInt	I, Q, M, D, L	Número de cadenas de caracteres que se han encontrado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro Mode

Bit	Valor de bit "0"	Valor de bit "1"	Descripción
0	Formato CSV (Comma Separated Values)	Formato FSR (Fixed Size Record)	Modo básico: Elección entre CSV o FSR
1	• Con CSV: los caracteres adicionales se escriben en la cadena de caracteres siguiente. • Con FSR: Los caracteres de relleno permanecen en la cadena.	• Con CSV: Los caracteres adicionales se eliminan. • Con FSR: Los caracteres de relleno se eliminan.	Con el bit 1 se elige cómo proceder con los caracteres adicionales: • Con CSV: – si el bit está activado, se ignoran los caracteres adicionales que no caben en la cadena de caracteres. Ejemplo: la instrucción escribe en una cadena de caracteres de longitud 16 (tipo de datos STRING[16]). El origen no contiene ningún separador después de los primeros 16 caracteres. Si está activado el bit 1, los caracteres sobrantes se ignoran y la instrucción continúa leyendo el array. – Si el bit no está activado, la instrucción se cancela y emite un mensaje de error en el parámetro Ret_Val. • Con FSR: – Si el bit está activado, los caracteres de relleno empleados no se escriben en la cadena de caracteres de destino. – Si el bit no está activado, los caracteres de relleno se escriben en la cadena de caracteres.
2	-	-	Reservado para el uso en versiones futuras.
3	El separador se conserva.	El separador se elimina.	Elección de si se elimina el separador al final de la cadena de caracteres total.
4	Conservar la longitud de las cadenas de caracteres no escritas (STRING).	Cambiar a "0" la longitud de las cadenas de caracteres no escritas (STRING).	Elección de si las cadenas no utilizadas (STRING) deben cambiar a longitud "0" en el parámetro DestStruct.

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	Ningún error.
0001	Los caracteres adicionales se ignoran.
8190	No está soportada la selección en el parámetro Mode.
8x20	Cadenas de caracteres de origen no válidas.
8x50	VARIANT señala a un contenido vacío.
8x52	VARIANT señala a un bloque que no se ha creado con acceso optimizado.
8x53	VARIANT señala a una estructura de datos demasiado corta.
8x54	Tipo de datos no válido
8882	El valor en el parámetro Position está fuera de la cadena de caracteres de origen.
8x84	Se han encontrado caracteres adicionales.
80B4	Tipos de datos diferentes en los parámetros SrcStruct (origen) y DestArray (destino) o en los separadores (parámetros RecSeparator y EndSeparator).
80B5	Desbordamiento del búfer en la instrucción. En el parámetro DestArray se emiten las cadenas de caracteres incompletas.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)

* Observe lo siguiente respecto a los códigos de error:

- Los códigos de error se pueden representar en el editor de programación como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".
- La "x" en la segunda posición de los códigos de error listados indica el parámetro que ha causado el error. Ejemplo: Código de error 8352 hex = error ocurrido en el 3.er parámetro (EndSeparator), consulte la tabla de parámetros.
- Si el error no puede atribuirse a un parámetro en particular, se devuelve un "0". Ejemplo: Para el separador (parámetro RecSeparator) se utiliza el tipo de datos CHAR. Para el array del parámetro DestArray se utiliza el tipo de datos WCHAR. En este caso se devuelve el código de error 80B4.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

ATH: Convertir cadena de caracteres ASCII en número hexadecimal

Descripción

La instrucción "ATH" convierte la cadena de caracteres ASCII especificada en el parámetro de entrada IN en un número hexadecimal. El resultado de la conversión se deposita en el parámetro de salida OUT.

- Con el puntero en el parámetro IN (ASCII) se puede remitir a los siguientes tipos de datos: STRING, Array of CHAR, Array of BYTE, WSTRING, Array of WCHAR.
- Con el puntero en el parámetro OUT (hexadecimal) se puede remitir a los siguientes tipos de datos: Secuencias de bits, enteros, STRING, Array of CHAR, Array of BYTE, WSTRING, Array of WCHAR.

El número de caracteres ASCII que se deben convertir se define con el parámetro N. Como máximo se pueden convertir 32.767 caracteres ASCII válidos. Solo pueden interpretarse los números de "0" a "9", las letras mayúsculas de "A" a "F" y las letras minúsculas de "a" a "f". Todos los demás caracteres se convierten en ceros.

Dado que un carácter ASCII requiere 8 bits y un número hexadecimal solo 4, la palabra de salida es la mitad de larga que la palabra de entrada. Después de la conversión, los caracteres ASCII se ordenan a la salida en el mismo orden en el que se han leído. Si se trata de un número impar de caracteres ASCII, el número hexadecimal correspondiente al semibyte a la derecha del último número hexadecimal convertido se rellena con ceros.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ATH":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	VARIANT	I, Q, D, L	Puntero a cadena de caracteres ASCII
N	Input	INT	I, Q, M, D, L o constante	Número de caracteres ASCII que se deben convertir
RET_VAL	Return	WORD	I, Q, M, D, L	Estado de la instrucción
OUT	Output	VARIANT	I, Q, M, D, L	Número hexadecimal

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#....)*	Descripción
0000	Sin errores
0007	Carácter no válido. Solo pueden utilizarse los siguientes caracteres ASCII: Números de "0" a "9", letras mayúsculas de "A" a "F", letras minúsculas de "a" a "f".
8101	Puntero no válido en el parámetro IN, p. ej. porque se remite a un bloque de datos no existente.
8182	El búfer de entrada es demasiado pequeño para los datos en el parámetro N.
8120	Formato no válido en el parámetro IN.
8151	Tipo de datos no soportado en el parámetro IN.
8401	Puntero no válido en el parámetro OUT, p. ej. porque se remite a un bloque de datos no existente.
8482	El búfer de salida es demasiado pequeño para los datos en el parámetro N.
8420	Formato no válido en el parámetro OUT.
8451	Tipo de datos no soportado en el parámetro OUT.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Caracteres ASCII y valores hexadecimales

La tabla siguiente muestra los caracteres ASCII y los correspondientes valores hexadecimales:

Caracteres ASCII	Valor hexadecimal codificado en ASCII	Número hexadecimal
"0"	30	0
"1"	31	1
"2"	32	2
"3"	33	3
"4"	34	4
"5"	35	5
"6"	36	6
"7"	37	7
"8"	38	8
"9"	39	9
"A"	41	A
"B"	42	B
"C"	43	C
"D"	44	D
"E"	45	E
"F"	46	F

Ejemplo

La tabla siguiente muestra ejemplos de conversión de cadenas de caracteres ASCII en un número hexadecimal:

IN	N	OUT	Estado ENO
'0123'	4	16#0123	1
'123AFx1a23'	10	16#123AF01a23	0

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

HTA: Convertir número hexadecimal en cadena de caracteres ASCII

Descripción

La instrucción "HTA" convierte el número hexadecimal especificado en la entrada IN en una cadena de caracteres ASCII. El resultado de la conversión se deposita en la dirección especificada en el parámetro OUT.

- Con el puntero en el parámetro IN (hexadecimal) se puede remitir a los siguientes tipos de datos: Secuencias de bits, enteros, STRING, Array of CHAR, Array of BYTE, WSTRING, Array of WCHAR.
- Con el puntero en el parámetro OUT (ASCII) se puede remitir a los siguientes tipos de datos: STRING, Array of CHAR, Array of BYTE , WSTRING, Array of WCHAR.

El número de bytes hexadecimales que se deben convertir se define con el parámetro N. Dado que un carácter ASCII requiere 8 bits y un número hexadecimal solo 4, el valor de salida es el doble de largo que el valor de entrada. Cada semibyte del número hexadecimal se convierte en un carácter manteniendo el orden original.

En la cadena de caracteres ASCII pueden escribirse como máximo 32.767 caracteres. El resultado de la conversión se representa con los números "0" a "9" y las letras mayúsculas "A" a "F".

Si no se puede mostrar el resultado completo de la conversión en el parámetro OUT, el resultado solo se escribe parcialmente en el parámetro.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "HTA":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	VARIANT	I, Q, M, D, L	Dirección inicial de la cifra hexadecimal
N	Input	INT	I, Q, M, D, L o constante	Número de bytes hexadecimales que se deben convertir
RET_VAL	Return	WORD	I, Q, M, D, L	Mensaje de error
OUT	Output	VARIANT	I, Q, D, L	Dirección en la que se guarda el resultado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0000	Sin errores
8101	Puntero no válido en el parámetro IN, p. ej. porque se remite a un bloque de datos no existente.
8182	El búfer de entrada es demasiado pequeño para los datos en el parámetro N.

Código de error* (W#16#....)	Descripción
8120	Formato no válido en el parámetro IN.
8151	Tipo de datos no soportado en el parámetro IN.
8401	Puntero no válido en el parámetro OUT, p. ej. porque se remite a un bloque de datos no existente.
8482	El búfer de salida es demasiado pequeño para los datos en el parámetro N.
8420	Formato no válido en el parámetro OUT.
8451	Tipo de datos no soportado en el parámetro OUT.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Caracteres ASCII y valores hexadecimales

La tabla siguiente muestra los caracteres ASCII y los correspondientes valores hexadecimales:

Número hexadecimal	Valor hexadecimal codificado en ASCII	Caracteres ASCII
0	30	"0"
1	31	"1"
2	32	"2"
3	33	"3"
4	34	"4"
5	35	"5"
6	36	"6"
7	37	"7"
8	38	"8"
9	39	"9"
A	41	"A"
B	42	"B"
C	43	"C"
D	44	"D"
E	45	"E"
F	46	"F"

Ejemplo

La tabla siguiente muestra ejemplos de conversión de números hexadecimales en cadenas de caracteres ASCII:

IN	N	OUT	Estado ENO
W#16#0123	2	'0123'	1
16#123AF01023	4	'123AF010'	0

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Otras instrucciones

LEN: Determinar la longitud de una cadena de caracteres

Descripción

Una variable del tipo de datos STRING contiene dos longitudes: la longitud máxima y la longitud actual (es decir, el número de caracteres válidos actualmente).

- La longitud máxima de la cadena de caracteres de cada variable se especifica entre corchetes en la palabra clave STRING. El número de bytes ocupados por una cadena de caracteres excede en 2 la longitud máxima.
- La longitud actual representa el número de posiciones de carácter realmente ocupadas. La longitud actual es menor o igual que la longitud máxima.

La instrucción "LEN" consulta la longitud actual de la cadena de caracteres especificada en el parámetro de entrada IN y la devuelve como valor numérico en el parámetro de salida OUT. Una cadena vacía ('') tiene la longitud cero.

Si ocurren errores durante la ejecución de la instrucción, se emite una cadena vacía.

Nota

Leer la longitud máxima

La instrucción "MAX_LEN (Página 4717)" permite leer la longitud actual de una cadena de caracteres.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LEN":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Cadena de caracteres
OUT	Return	INT, DINT, REAL, LREAL	I, Q, M, D, L	Número de caracteres válidos

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

CONCAT: Agrupar cadenas de caracteres

Descripción

La instrucción "CONCAT" une la cadena de caracteres del parámetro de entrada IN1 con la cadena de caracteres del parámetro de entrada IN2. El resultado se indica en el parámetro de salida OUT en formato STRING. Si la cadena de caracteres resultante excede la longitud de la variable especificada en el parámetro de salida OUT, la cadena se trunca a la longitud disponible.

Si ocurren errores durante la ejecución de la instrucción y es posible escribir en el parámetro de salida OUT, se emite una cadena vacía.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CONCAT":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING	D, L o constante	Cadena de caracteres
IN2	Input	STRING	D, L o constante	Cadena de caracteres
OUT	Return	STRING	D, L	Cadena de caracteres resultante

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

LEFT: Leer los caracteres izquierdos de una cadena

Descripción

La instrucción "LEFT" extrae una cadena de caracteres parcial a partir del primer carácter de la cadena en el parámetro de entrada IN. El número de caracteres que se extraen se especifica en el parámetro L. Los caracteres extraídos se emiten en el parámetro de salida OUT en formato STRING.

Si el número de caracteres que debe extraerse excede la longitud actual de la cadena de caracteres, el parámetro de salida OUT devuelve como resultado la cadena de caracteres de entrada. Si el parámetro L tiene el valor "0" o si el valor de entrada es una cadena vacía, se devuelve una cadena vacía. Si el valor del parámetro L es negativo, se devuelve una cadena vacía.

Si ocurren errores durante la ejecución de la instrucción y es posible escribir en el parámetro de salida OUT, se emite una cadena vacía.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LEFT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Cadena de caracteres
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Número de caracteres que deben extraerse
OUT	Return	STRING	D, L	Cadena de caracteres parcial extraída

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

RIGHT: Leer los caracteres derechos de una cadena

Descripción

Con la instrucción se extraen los últimos L caracteres de una cadena en el parámetro de entrada IN. El número de caracteres que se extraen se especifica en el parámetro L. Los caracteres extraídos se emiten en el parámetro de salida OUT en formato STRING.

Si el número de caracteres que debe extraerse excede la longitud actual de la cadena de caracteres, el parámetro de salida OUT devuelve como resultado la cadena de caracteres de entrada. Si el parámetro L tiene el valor "0" o si el valor de entrada es una cadena vacía, se devuelve una cadena vacía. Si el valor del parámetro L es negativo, se devuelve una cadena vacía.

Si ocurren errores durante la ejecución de la instrucción y es posible escribir en el parámetro de salida OUT, se emite una cadena vacía.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RIGHT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Cadena de caracteres
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Número de caracteres que deben extraerse
OUT	Return	STRING	D, L	Cadena de caracteres parcial extraída

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

MID: Leer los caracteres centrales de una cadena

Descripción

La instrucción extrae una parte de la cadena de caracteres del parámetro de entrada IN. Con el parámetro P se especifica la posición del primer carácter que se va a extraer. La longitud de la cadena de caracteres que se va a extraer se determina con el parámetro L. La cadena de caracteres extraída se emite en el parámetro de salida OUT.

Al ejecutar esta instrucción deben observarse las reglas siguientes:

- Si el número de caracteres que se va a extraer excede la longitud actual de la cadena de caracteres en el parámetro de entrada IN, se emite una cadena de caracteres parcial que comienza en la posición de carácter P y que continúa hasta el final de la cadena de caracteres.
- Si la posición de carácter especificada en el parámetro P se encuentra fuera de la longitud actual de la cadena de caracteres del parámetro de entrada IN, se deposita una cadena de caracteres vacía en el parámetro de salida OUT.
- Si el valor del parámetro P o L es cero o negativo, se deposita una cadena de caracteres vacía en el parámetro de salida OUT.

Si ocurren errores durante la ejecución de la instrucción y es posible escribir en el parámetro de salida OUT, se emite una cadena vacía.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MID":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Cadena de caracteres
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Longitud de la cadena de caracteres que debe extraerse
P	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Posición del primer carácter que debe extraerse (primer carácter = 1)
OUT	Return	STRING	D, L	Cadena de caracteres parcial extraída

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

DELETE: Borrar caracteres de una cadena

Descripción

La instrucción borra una parte de la cadena de caracteres del parámetro de entrada IN. La posición del primer carácter que se va a borrar se determina con el parámetro P. En el

parámetro L se especifica el número de caracteres que se van a borrar. La cadena de caracteres parcial restante se emite en el parámetro de salida OUT en formato STRING.

Al ejecutar esta instrucción deben observarse las reglas siguientes:

- Si el valor del parámetro P es igual o menor que cero, se devuelve una cadena de caracteres vacía en el parámetro de salida OUT.
- Si el valor del parámetro P excede la longitud actual de la cadena de caracteres en la entrada IN, el parámetro de salida OUTdevuelve la cadena de caracteres de entrada.
- Si el valor del parámetro L es cero, el parámetro de salida OUT devuelve la cadena de caracteres de entrada.
- Si el número de caracteres que debe borrarse en el parámetro L excede la longitud de la cadena de caracteres en el parámetro de entrada IN, se emite una cadena de caracteres vacía.
- Si el valor del parámetro L es negativo, se emite una cadena de caracteres vacía.

Si ocurren errores durante la ejecución de la instrucción y es posible escribir en el parámetro de salida OUT, se emite una cadena vacía.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DELETE":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING	D, L o constante	Cadena de caracteres
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Número de caracteres que se van a borrar
P	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Posición del primer carácter que debe borrarse
OUT	Return	STRING	D, L	Cadena de caracteres resultante

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

INSERT: Insertar caracteres en una cadena

Descripción

La instrucción inserta la cadena de caracteres del parámetro de entrada IN2 en la cadena de caracteres del parámetro de entrada IN1. Con el parámetro P se especifica la posición del carácter a partir del cual deben insertarse los caracteres. El resultado se indica en el parámetro de salida OUTen formato STRING.

Al ejecutar esta instrucción deben observarse las reglas siguientes:

- Si el valor del parámetro P excede la longitud actual de la cadena de caracteres en el parámetro de entrada IN1, la cadena de caracteres del parámetro de entrada IN2 se añade a la cadena de caracteres del parámetro de entrada IN1.
- Si el valor del parámetro P es cero, en la salida OUT se emite la cadena de caracteres del parámetro IN2 seguida por la cadena de caracteres del parámetro IN1.
- Si el valor del parámetro P es negativo, en la salida OUT se emite una cadena de caracteres vacía.
- Si la cadena de caracteres resultante excede la longitud de la variable especificada en el parámetro de salida OUT, la cadena se trunca a la longitud disponible.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "INSERT":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING	D, L o constante	Cadena de caracteres
IN2	Input	STRING	D, L o constante	Cadena de caracteres que se va a insertar
P	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Posición de inserción
OUT	Return	STRING	D, L	Cadena de caracteres resultante

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

REPLACE: Reemplazar caracteres de una cadena

Descripción

La instrucción sustituye una parte de la cadena de caracteres de la entrada IN1 por la cadena de caracteres del parámetro de entrada IN2. La posición del primer carácter que se va a sustituir se determina con el parámetro P. El número de caracteres que se van a sustituir se especifica en el parámetro L. El resultado se indica en el parámetro de salida OUT en formato STRING.

Al ejecutar esta instrucción deben observarse las reglas siguientes:

- Si el valor del parámetro P es igual o menor que cero, se devuelve una cadena de caracteres vacía en el parámetro de salida OUT.
- Si el valor del parámetro L es menor que cero, en el parámetro de salida OUT se devuelve una cadena de caracteres vacía.
- Si P es igual a uno, se sustituye la cadena de caracteres de la entrada IN1 a partir del primer carácter (inclusive).
- Si el valor del parámetro P excede la longitud actual de la cadena de caracteres en el parámetro de entrada IN1, la cadena de caracteres del parámetro de entrada IN2 se añade a la cadena de caracteres del parámetro de entrada IN1.

- Si la cadena de caracteres resultante excede la longitud de la variable especificada en el parámetro de salida OUT, la cadena se trunca a la longitud disponible.
- Si el valor del parámetro L es igual a cero, no se sustituyen caracteres, sino que se insertan. Rigen las mismas condiciones que para la instrucción INSERT. Consulte también: INSERT: Insertar caracteres en una cadena (Página 4733)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "REPLACE":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING	D, L o constante	Cadena cuyos caracteres se sustituyen.
IN2	Input	STRING	D, L o constante	Cadena cuyos caracteres se insertan.
L	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Número de caracteres que se van a sustituir
P	Input	BYTE, INT, SINT, USINT	I, Q, M, D, L o constante	Posición del primer carácter que se va a sustituir
OUT	Return	STRING	D, L	Cadena de caracteres resultante

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

FIND: Buscar caracteres en una cadena

Descripción

La instrucción busca un determinado carácter o una determinada secuencia de caracteres en la cadena de caracteres del parámetro de entrada IN1.

- El valor que se va a buscar se especifica en el parámetro de entrada IN2. La búsqueda se efectúa de izquierda a derecha.
- En el parámetro de salida OUT se emite la posición de la primera ocurrencia. Si la búsqueda es infructuosa, se deposita el valor "0" en el parámetro de salida OUT.

Si en el parámetro IN2 se indica un carácter no válido o si durante la ejecución se produce un error, se devuelve el valor "0" en el parámetro OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "FIND":

Parámetro	Declaración	Tipo de datos		Área de memoria	Descripción
		S7-1200	S7-1500		
IN1	Input	STRING	STRING, WSTRING	D, L o constante	Cadena de caracteres en la que se busca
IN2	Input	STRING, CHAR	STRING, WSTRING, CHAR, WCHAR	D, L o constante (Con (W)CHAR también I, Q, M)	Caracteres buscados
OUT	Return	INT	INT	I, Q, M, D, L	Posición de carácter

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Información del tiempo de ejecución

GetSymbolName: Leer nombre de una variable en el parámetro de entrada.

Descripción

La instrucción "GetSymbolName" permite leer el nombre de una variable interconectada mediante el parámetro de entrada de un bloque.

Si un bloque se utiliza más de una vez en el proyecto o es llamado por diferentes variables, existe la posibilidad de evaluar el nombre de la variable que realiza la llamada con la instrucción "GetSymbolName". El valor de proceso de la variable es irrelevante en este contexto.

- El parámetro de entrada de la interfaz del bloque se indica en el parámetro VARIABLE de la instrucción. No utilice en este parámetro variables PLC o de bloques de datos, sino única y exlusivamente un parámetro de la interfaz.
- Con el parámetro SIZE se puede limitar la longitud del nombre de la variable que se ha leído. Si el nombre está cortado, ello se indica con el carácter "..." (carácter Unicode 16#2026) al final del nombre. Tenga en cuenta que el propio carácter "..." tiene una longitud de 1. Si por ejemplo se introduce como longitud máxima 6 en el parámetro SIZE y se lee el nombre de variable "MyPLCTag" a través de la interfaz del bloque, se devolverá lo siguiente: "MyPLC...".
- En el parámetro OUT se devuelve el nombre leído.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GetSymbolName":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
VARIABLE	Input	PARAMETER	L	Parámetro de entrada de la interfaz del bloque
SIZE	Input	UINT	I, Q, M, D, L	Limitación de la longitud del nombre leído
OUT	Return	WSTRING	I, Q, M, D, L	Nombre de la variable léida a través de la interfaz del bloque

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GetInstanceName: Leer el nombre de la instancia del bloque

Descripción

La instrucción "GetInstanceName" lee el nombre del bloque de datos de instancia en un bloque de función.

- Con el parámetro SIZE se define cuántos caracteres del nombre de la instancia deben leerse. Si para SIZE se utiliza el valor "0", se lee todo el nombre.
- En el parámetro OUT se escribe el nombre del bloque de datos de instancia. Si el nombre del bloque de datos de instancia es más largo que la longitud máxima de WSTRING (254 caracteres), el nombre se corta.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GetInstanceName":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIZE	Input	DINT	I, Q, M, D, L o constante	Número de los primeros caracteres que deben leerse del nombre del bloque de datos de instancia.
OUT	Output	WSTRING	I, Q, M, D, L	Nombre leído del bloque de datos de instancia

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GetBlockName: Leer el nombre del bloque

Descripción

Mediante la instrucción "GetBlockName" se lee el nombre del bloque en el que se llama la instrucción.

- Con el parámetro SIZE se define cuántos caracteres del nombre del bloque deben leerse. Si para SIZE se utiliza el valor "0", se lee todo el nombre.
- En el parámetro OUT se escribe el nombre del bloque. Si el nombre del bloque es más largo que la longitud máxima de WSTRING (254 caracteres), el nombre se corta.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GetBlockName":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIZE	Input	UINT	I, Q, M, D, L o constante	Número de los primeros caracteres que deben leerse del nombre del bloque.
RET_VAL	Output	WSTRING	I, Q, M, D, L	Nombre leído del bloque de datos

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Memoria imagen de proceso

UPDAT_PI: Actualizar memoria imagen de proceso de las entradas

Descripción

Esta instrucción permite actualizar la memoria imagen de proceso del OB 1 (=memoria imagen parcial de proceso 0) de las entradas o una memoria imagen parcial de proceso de las entradas definida mediante configuración.

Si se ha configurado como método de notificación para la actualización de la memoria imagen de proceso por el sistema la notificación repetida de todos los errores de acceso a la periferia, la actualización de la memoria imagen de proceso seleccionada se lleva a cabo de modo continuo.

De lo contrario, dicha actualización solo se efectúa si la memoria imagen parcial de proceso seleccionada no es actualizada por el sistema, es decir,

- si no se ha asignado esa memoria imagen parcial de proceso a ningún OB de alarma, o
- si se ha seleccionado la memoria imagen parcial de proceso 0 y se ha desactivado (en la configuración) la actualización de la memoria imagen parcial de proceso del OB 1.

Nota

Las direcciones lógicas que se han asignado en la configuración a una memoria imagen parcial de proceso de las entradas no pertenecen ya a la memoria imagen de proceso del OB 1 de las entradas.
Una memoria imagen parcial de proceso que se actualiza con "UPDAT_PI" no debe actualizarse al mismo tiempo con la instrucción "SYNC_PI (Página 4742)".

Si se ha asignado a un OB de alarma la actualización por el sistema de la memoria imagen de proceso del OB 1 de las entradas y de la memoria imagen parcial de proceso de las entradas, la actualización se realiza independientemente de las llamadas de "UPDAT_PI".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "UPDAT_PI":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PART	Input	PIP	I, Q, M, D, L o constante	Número de la memoria imagen parcial de proceso de las entradas que se debe actualizar. Rango máximo de valores (el rango depende de la CPU): De 0 a 31 (0 significa memoria imagen de proceso del OB 1, n con 1 <= n <= 31 significa memoria imagen parcial de proceso n).
RET_VAL	Return	INT	I, Q, M, D, L, P	Información de error
FLADDR	Output	WORD	I, Q, M, D, L, P	Dirección del primer byte que provoca errores, en caso de que se haya producido un error de acceso

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	Valor no válido en el parámetro PART
8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU.

Código de error* (W#16#...)	Explicación
8092	La memoria imagen parcial de proceso es actualizada por el sistema y no puede utilizarse para el parámetro PART.
8093	La actualización de la memoria imagen parcial de proceso está teniendo lugar en otro OB.
80A0	Se ha detectado un error de acceso al acceder a la periferia.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Nota

Si se utiliza esta instrucción para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPRD_DAT (Página 4791)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

UPDAT_PO: Actualizar memoria imagen de proceso de las salidas

Descripción

Esta instrucción permite transferir a los módulos de salida los estados lógicos de la memoria imagen de proceso del OB 1 (=memoria imagen parcial de proceso 0) de las salidas o de una memoria imagen parcial de proceso de las salidas definida mediante configuración.

Si se ha definido un área de coherencia para la memoria imagen parcial de proceso seleccionada, los datos en cuestión se transfieren de modo coherente al módulo de periferia correspondiente.

Nota

Las direcciones lógicas que se han asignado mediante configuración a una memoria imagen parcial de proceso de las salidas no pertenecen ya a la memoria imagen de proceso del OB 1 de las salidas.
Las salidas que se actualizan con "UPDAT_PO" no deben actualizarse al mismo tiempo con la instrucción "SYNC_PO (Página 4744)".

Si se ha asignado a un OB de alarma la transferencia a los módulos de salida de la memoria imagen de proceso del OB 1 de las salidas y de la memoria imagen parcial de proceso de las salidas, el sistema realiza la transferencia independientemente de las llamadas de "UPDAT_PO".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "UPDAT_PO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PART	Input	PIP	I, Q, M, D, L o constante	Número de la memoria imagen parcial de proceso de las salidas que se debe transferir. Rango máximo de valores (el rango depende de la CPU): de 0 a 31. (0 significa memoria imagen de proceso del OB 1, n con 1 <= n <= 31 significa memoria imagen parcial de proceso n)
RET_VAL	Return	INT	I, Q, M, D, L, P	Información de error
FLADDR	Output	WORD	I, Q, M, D, L, P	Dirección del primer byte que provoca errores, en caso de que se haya producido un error de acceso

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	Valor no válido en el parámetro PART
8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU.
8092	La memoria imagen parcial de proceso es actualizada por el sistema y no puede utilizarse para el parámetro PART.
8093	La actualización de la memoria imagen parcial de proceso tiene lugar en otro OB.
80A0	Se ha detectado un error de acceso al acceder a la periferia.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Nota

Si se utiliza esta instrucción para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPWR_DAT (Página 4793)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

SYNC_PI: Sincronizar memoria imagen de proceso de las entradas

Descripción

La instrucción "SYNC_PI" permite actualizar de modo isócrono una memoria imagen parcial de proceso de las entradas. Con esta instrucción, un programa de usuario vinculado a un ciclo DP o a un ciclo de emisión PN puede actualizar los datos de entrada registrados de modo isócrono y coherente en una memoria imagen parcial de proceso de las entradas.

Llamada

"SYNC_PI" puede interrumpirse, y solo puede llamarse en los OBs 61, 62, 63 y 64.

Nota

La llamada de la instrucción "SYNC_PI" en los OB 61 a 64 solo está autorizada si la memoria imagen parcial de proceso en cuestión se ha asignado al OB correspondiente en la configuración hardware

Una memoria imagen parcial de proceso que se actualiza con "SYNC_PI" no puede actualizarse al mismo tiempo con la instrucción "UPDAT_PI (Página 4738)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SYNC_PI":

Parámetro	Declaración	Tipo de datos	Área de memoria	Rango de valores	Descripción
PART	Input	PIP	I, Q, M, D, L o constante	de 1 a 31	Número de la memoria imagen parcial de proceso de las entradas que se debe actualizar en modo isócrono.
RET_VAL	Return	INT	I, Q, M, D, L, P	-	Información de error
FLADDR	Output	WORD	I, Q, M, D, L, P		Dirección del primer byte que provoca errores, en caso de que se haya producido un error de acceso.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)*	Explicación
0000	No ha ocurrido ningún error.
0001	Advertencia de coherencia. La actualización de la memoria imagen parcial de proceso se ha repartido entre dos ciclos DP o PN. Sin embargo, los datos pertenecientes a un esclavo o un dispositivo IO se han transferido de modo coherente.
8090	Valor no admisible en el parámetro PART, o bien la actualización de la memoria imagen parcial de proceso de las entradas indicada no está permitida en este OB. No se ha actualizado la memoria imagen parcial de proceso de las entradas.
8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU. No se ha actualizado la memoria imagen parcial de proceso de las entradas.
8092	La memoria imagen parcial de proceso es actualizada por el sistema y no puede utilizarse para el parámetro PART.
8093	La actualización de la memoria imagen parcial de proceso tiene lugar en otro OB.
80A0	Durante la actualización se ha detectado un error de acceso. Las entradas afectadas se han ajustado a "0".
80A1	El momento de actualización es posterior a la ventana de acceso admisible. No se ha actualizado la memoria imagen parcial de proceso de las entradas. El ciclo DP o PN es demasiado corto y no garantiza suficiente tiempo para procesar la instrucción. Por ello deben incrementarse los tiempos TDP (también conocidos como T_DC), Ti y To.
80A2	Error de acceso con advertencia de coherencia Durante la actualización de la memoria imagen parcial de proceso de las entradas se ha detectado un error de acceso con advertencia de coherencia simultánea. • Los datos de las entradas erróneas no han sido leídos todavía por la periferia. En la memoria imagen parcial de proceso de las entradas las entradas afectadas se ponen a cero. • La actualización de los datos de entrada no afectados por el error de acceso se ha distribuido entre dos ciclos DP o PN.
80C1	El momento de actualización es anterior a la ventana de acceso admisible. No se ha actualizado la memoria imagen parcial de proceso de las entradas.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Nota

Si se utiliza la instrucción "SYNC_PI" para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPRD_DAT (Página 4791)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

SYNC_PO: Sincronizar memoria imagen de proceso de las salidas

Descripción

La instrucción "SYNC_PO" permite actualizar de modo isócrono una memoria imagen parcial de proceso de las salidas. Con esta instrucción, un programa de usuario vinculado a un ciclo DP o a un tiempo de ciclo de emisión PN puede transferir a la periferia de modo isócrono y coherente los datos de salida calculados de una memoria imagen parcial de proceso de las salidas.

Llamada

"SYNC_PO" puede interrumpirse, y solo puede llamarse en los OBs 61, 62, 63 y 64.

Nota

La llamada de la instrucción "SYNC_PO" en los OB 61 a 64 solo está autorizada si la memoria imagen parcial de proceso en cuestión se ha asignado al OB correspondiente en la configuración hardware

Una memoria imagen parcial de proceso que se actualiza con "SYNC_PO" no puede actualizarse al mismo tiempo con la instrucción "UPDAT_PO (Página 4740)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SYNC_PO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Rango de valores	Descripción
PART	Input	PIP	I, Q, M, D, L o constante	de 1 a 31	Número de la memoria imagen parcial de proceso de las salidas que se debe actualizar en modo isócrono.
RET_VAL	Return	INT	I, Q, M, D, P	-	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
FLADDR	Output	WORD	I, Q, M, D, L, P	-	Dirección del primer byte que provoca errores.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)*	Explicación
0000	No ha ocurrido ningún error.
0001	Advertencia de coherencia. La actualización de la memoria imagen parcial de proceso se ha repartido entre dos ciclos DP o PN. Sin embargo, los datos pertenecientes a un esclavo o un dispositivo IO se han transferido de modo coherente.
8090	Valor no admisible en el parámetro PART, o bien la actualización de la memoria imagen parcial de proceso de las salidas indicada no está permitida en este OB. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
8092	La memoria imagen parcial de proceso es actualizada por el sistema y no puede utilizarse para el parámetro PART.
8093	La actualización de la memoria imagen parcial de proceso tiene lugar en otro OB.
80A0	Durante la actualización de la memoria imagen parcial de proceso de las salidas se ha detectado un error de acceso. No se han transferido las salidas erróneas a la periferia. En la memoria imagen parcial de proceso de las salidas, estas no sufren modificación alguna.
80A1	Error de acceso con advertencia de coherencia Durante la actualización de la memoria imagen parcial de proceso de las salidas se ha detectado un error de acceso con advertencia de coherencia simultánea. • No se han transferido los datos de las salidas erróneas a la periferia. En la memoria imagen parcial de proceso de las salidas, las salidas afectadas no sufren modificación alguna. • La actualización de los datos de salida no afectados por el error de acceso se ha distribuido entre dos ciclos DP o PN.
80A2	El momento de actualización es posterior a la ventana de acceso admisible. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
80C1	El momento de actualización es anterior a la ventana de acceso admisible. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Nota

Si se utiliza la instrucción "SYNC_PO" para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPWR_DAT (Página 4793)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Periferia descentralizada

RDREC: Leer registro

Descripción

La instrucción "RDREC" permite leer el registro con número INDEX del módulo direccionado mediante ID. Puede tratarse de un módulo central o descentralizado (PROFIBUS DP o PROFINET IO).

- El parámetro ID permite seleccionar desde qué módulo (DP/PROFINET IO) se debe leer un registro. Para el parámetro ID utilice la ID de hardware del módulo.
- El parámetro INDEX permite seleccionar el registro que se debe leer en el módulo. Qué registros se pueden leer y con qué número de registro depende del módulo. Para más información al respecto consulte el manual del respectivo módulo.
- Los registros que se pueden leer mediante "RDREC" son de diferente longitud. Con MLEN se especifica el número máximo de bytes del registro que se quiere leer. Si en el parámetro MLEN se selecciona "0" como longitud, se escribirá el registro completo en el parámetro RECORD.
- Debe escogerse un área de destino RECORD con una longitud de por lo menos MLEN bytes. Si se lee el registro completo con MLEN=0, para RECORD se deberá utilizar la longitud máxima. La estructura (estructura, tipos de datos y longitud) que se utiliza en el parámetro RECORD también depende de qué registro se lea de qué módulo.
- El valor TRUE del parámetro de salida VALID indica que el registro se ha transferido correctamente al área de destino RECORD. En ese caso, el parámetro de salida LEN contiene la longitud en bytes de los datos leídos.
- Si se ha producido un error durante la transferencia del registro, se señaliza mediante el parámetro de salida ERROR. En ese caso, el parámetro de salida STATUS contiene la información de error.

Nota

La interfaz de la instrucción "RDREC" es idéntica a la del FB "RDREC" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Funcionamiento

"RDREC" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia del registro, debe llamarse "RDREC" con REQ= 1.

Mediante el parámetro de salida BUSYy los dos bytes centrales del parámetro de salida STATUS, se muestra el estado de la petición. Los dos bytes centrales de STATUScorresponden al parámetro de salida RET_VALde las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

La transferencia del registro se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RDREC":

Parámetro	Declaración	Tipo de datos*	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	REQ = 1: Realizar la transferencia del registro
ID	Input	HW_IO	I, Q, M, L o constante	ID de hardware del módulo de hardware (DP/PROFINET IO) El número se asigna automáticamente y se guarda en las propiedades del módulo o de la interfaz en la configuración hardware.
INDEX	Input	BYTE, DINT, INT, SINT, UINT, USINT, WORD	I, Q, M, D, L o constante	Número de registro
MLEN	Input	BYTE, UINT, USINT	I, Q, M, D, L o constante	Longitud máxima en bytes de la información de registro que se debe leer.
VALID	Output	BOOL	I, Q, M, D, L	El nuevo registro se ha recibido y es válido.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la lectura todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: Se ha producido un error durante la lectura.
STATUS	Output	DWORD	I, Q, M, D, L	Estado del bloque o información de error.
LEN	Output	UINT	I, Q, M, D, L	Longitud de la información del registro leída.
RECORD	InOut	VARIANT	I, Q, M, D, L	Área de destino para el registro leído.
* En AWL no se realiza conversión implícita, por lo que el volumen de tipos de datos admisibles puede estar restringido. Al programar en AWL tenga en cuenta los tipos de datos admisibles en el tooltip del parámetro.				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si utiliza "RDREC" para leer un registro en PROFINET IO, los valores negativos en los parámetros INDEX, MLEN y LENse interpretan como números enteros de 16 bits sin signo previo.

Parámetro STATUS

Para la interpretación del parámetro STATUS, véase Parámetro STATUS (Página 4755).

Consulte también

Fundamentos del acceso a bloques (Página 2258)

WRREC: Escribir registro

Descripción

La instrucción "WRREC" permite transferir el registro RECORD al componente direccionado con ID. Puede tratarse de un módulo central o de un componente descentralizado (PROFIBUS DP o PROFINET IO).

- Con LENse indica la longitud en bytes del registro que se debe transferir. Por ello debe escogerse un área de origen RECORDcon una longitud de por lo menos LENbytes.
- El valor TRUE del parámetro de salida DONEindica que el registro se ha transferido correctamente.
- Si se ha producido un error durante la transferencia del registro, se señaliza mediante el parámetro de salida ERROR. En ese caso, el parámetro de salida STATUS contiene la información de error.

Nota

La interfaz de la instrucción "WRREC" es idéntica a la del FB "WRREC" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Funcionamiento

"WRREC" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia del registro, debe llamarse "WRREC" con REQ = 1.

Mediante el parámetro de salida BUSYy los dos bytes centrales del parámetro de salida STATUS, se muestra el estado de la petición. Los dos bytes centrales de STATUScorresponden al parámetro de salida RET_VALde las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Asegúrese de asignar siempre el mismo valor al parámetro actual de RECORD en todas las llamadas a "WRREC" pertenecientes a una misma petición. Lo mismo se aplica a los parámetros actuales de LEN.

La transferencia del registro se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WRREC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	REQ = 1: Realizar la transferencia del registro
ID	Input	HW_IO	I, Q, M, L o constante	Número de identificación de los componentes de hardware (DP/PROFINET IO) El número se asigna automáticamente y se guarda en las propiedades del componente o de la interfaz en la configuración hardware.
INDEX	Input	DINT	I, Q, M, D, L o constante	Número de registro
LEN	Input	BYTE, UINT, USINT	I, Q, M, D, L o constante	(oculto) Longitud máxima en bytes del registro que se debe transferir
DONE	Output	BOOL	I, Q, M, D, L	Se ha transferido el registro
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la escritura todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: Se ha producido un error durante la escritura.
STATUS	Output	DWORD	I, Q, M, D, L	Estado del bloque o información de error Para la interpretación del parámetro STATUS, véase Parámetro STATUS (Página 4755).
RECORD	InOut	VARIANT	I, Q, M, D, L	Registro

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si utiliza "WRREC" para escribir un registro en PROFINET IO, los valores negativos de los parámetros INDEXy LEN se interpretan como números enteros de 16 bits sin signo previo.

Parámetro STATUS

Para la interpretación del parámetro STATUS, véase Parámetro STATUS (Página 4755).

GETIO: Leer memoria imagen de proceso

Descripción

Con la instrucción "GETIO" se leen de modo coherente todas las entradas de un esclavo DP normalizado/dispositivo PROFINET IO. Para ello, la instrucción "GETIO" llama la instrucción "DPRD_DAT (Página 4791)". Si no se ha producido ningún error durante la transferencia de datos, los datos leídos se registran en el área de destino del parámetro INPUTS .

El área de destino debe tener la misma longitud que se haya configurado para el componente seleccionado.

En un esclavo DP normalizado con estructura modular o con varias ID de DP, la llamada con "GETIO" solo permite acceder cada vez a los datos de un solo componente o ID de DP en la dirección inicial configurada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GETIO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	ID de hardware del esclavo DP normalizado/ dispositivo PROFINET IO.
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "DPRD_DAT (Página 4791)" en la forma DW#16#40xxxx00.
LEN	Output	INT	I, Q, M, D, L	Número de datos leídos, en bytes.
INPUTS	InOut	VARIANT	I, Q, M, D	Área de destino para los datos leídos. Debe tener la misma longitud que se haya configurado para el esclavo DP normalizado/dispositivo PROFINET IO seleccionado. Solo se admite el tipo de datos BYTE .

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Consulte también: DPRD_DAT: Leer datos coherentes de un esclavo DP normalizado (Página 4791).

SETIO: Transferir memoria imagen de proceso

Descripción

Con la instrucción "SETIO" se transfieren de modo coherente los datos del área de origen definida por el parámetro OUTPUTS al esclavo DP normalizado/dispositivo PROFINET IO direccionado y, en su caso, a la memoria imagen de proceso (concretamente cuando el área de direcciones en cuestión del esclavo DP normalizado/dispositivo PROFINET IO se ha configurado como área de coherencia en una memoria imagen de proceso). "Para ello, SETIO llama la instrucción DPWR_DAT (Página 4793)".

El área de origen debe tener la misma longitud que se haya configurado para el componente seleccionado.

En un esclavo DP normalizado/dispositivo PROFINET IO con estructura modular o con varias ID, la llamada de "SETIO" permite acceder a los datos de un solo componente o ID de DP.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SETIO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	ID de hardware del esclavo DP normalizado/dispositivo PROFINET IO.
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "DPWR_DAT (Página 4793)" en la forma DW#16#40xxxx00.
OUTPUTS	InOut	VARIANT	I, Q, M, D	Área de origen para los datos que se deben escribir. Debe tener la misma longitud que se haya configurado para el esclavo DP normalizado/dispositivo PROFINET IO seleccionado. Solo se admite el tipo de datos BYTE.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Consulte también: DPWR_DAT: Escribir datos coherentes de un esclavo DP normalizado (Página 4793).

GETIO_PART: Leer área de memoria imagen de proceso

Descripción

La instrucción "GETIO_PART" permite leer de forma coherente una parte conexa de las entradas de un módulo E/S. "Para ello, "GETIO_PART" llama la instrucción "DPRD_DAT (Página 4791)".

Mediante el parámetro de entrada ID se selecciona el módulo E/S a partir de la ID de hardware.

Con los parámetros OFFSET y LEN se define la porción que se debe leer del área de la memoria imagen de proceso. En el caso de que el módulo no cubra por completo el área de entrada delimitada mediante OFFSET y LEN, el bloque devuelve el código de error DW#16#40**80B7**00.

La longitud del área de destino debe ser superior o igual al número de bytes que se van a leer:

- Si no se ha producido ningún error durante la transferencia de datos, ERROR adquiere el valor FALSE. Los datos leídos se escriben en el área de destino definida en el parámetro INPUTS.
- Si se ha producido algún error durante la transferencia de datos, ERROR adquiere el valor TRUE. El parámetro STATUS recibe la información de error de "DPRD_DAT".
- Si el área de destino es mayor que LEN, se escriben los primeros LEN bytes del área de destino. ERROR adquiere el valor FALSE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GETIO_PART":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	ID de hardware del módulo
OFFSET	Input	INT	I, Q, M, D, L o constante	Número del primer byte que se debe leer en la memoria imagen de proceso del componente (valor más pequeño posible: 0).
LEN	Input	INT	I, Q, M, D, L o constante	Número de bytes que se van a leer.
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "DPRD_DAT" en la forma DW#16#40xxxx00, si ERROR = TRUE.
ERROR	Output	BOOL	I, Q, M, D, L	Visualización de error: ERROR = TRUE, en caso de que se produzca un error al llamar "DPRD_DAT".
INPUTS	InOut	VARIANT	I, Q, M, D	Área de destino para los datos leídos: Si el área de destino es mayor que LEN, se escriben los primeros LEN bytes del área de destino.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Véase el parámetro RET_VAL de la instrucción "DPRD_DAT (Página 4791)".

SETIO_PART: Transferir área de memoria imagen de proceso

Descripción

La instrucción "SETIO_PART" permite escribir de forma coherente los datos del área de origen delimitada por OUTPUTS en las salidas de un módulo E/S. "Para ello, SETIO_PART llama la instrucción "DPWR_DAT (Página 4793)".

A través del parámetro de entrada ID se selecciona el módulo E/S a partir de la ID de hardware. Con los parámetros OFFSET y LEN se define la porción que se debe escribir del área de la memoria imagen de proceso del componente direccionado mediante ID. En el caso de que el módulo no cubra por completo el área de salida delimitada mediante OFFSET y LEN, el bloque devuelve el código de error DW#16#40**80B7**00.

La longitud del área de origen debe ser superior o igual al número de bytes que se van a escribir:

- Si no se ha producido ningún error durante la transferencia de datos, ERROR adquiere el valor FALSE.
- Si se ha producido algún error durante la transferencia de datos, ERROR adquiere el valor TRUE, y STATUS recibe la información de error de "DPWR_DAT".
- Si el área de origen es mayor que LEN, se transfieren los primeros LEN bytes de OUTPUTS. ERROR adquiere el valor FALSE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SETIO_PART":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	ID de hardware del módulo E/S.
OFFSET	Input	INT	I, Q, M, D, L o constante	Número del primer byte que se debe escribir en la memoria imagen de proceso del componente (valor más pequeño posible: 0).
LEN	Input	INT	I, Q, M, D, L o constante	Número de bytes que se van a escribir.
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "DPWR_DAT" en la forma DW#16#40xxxx00, si ERROR = TRUE.
ERROR	Output	BOOL	I, Q, M, D, L	Visualización de error: ERROR = TRUE, en caso de que se produzca un error al llamar "DPWR_DAT".
OUTPUTS	InOut	VARIANT	I, Q, M, D	Área de origen para los datos que se deben escribir: Si el área de origen es mayor que LEN, se transmiten los primeros LEN bytes de OUTPUTS.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros STATUS y ERROR

Véase la instrucción "DPWR_DAT (Página 4793)".

RALRM: Recibir alarma

Descripción de RALRM

Descripción

La instrucción recibe una alarma, junto con la información correspondiente, desde un módulo de periferia (configuración centralizada) o desde un módulo de un esclavo DP o un dispositivo PROFINET IO, y proporciona dicha información a través de sus parámetros de salida.

Los parámetros de salida ofrecen tanto la información de arranque del OB llamado como información procedente del origen de la alarma.

La composición de la estructura de datos del área de destino AINFO equivale a la estructura de datos de PROFINET IO para una instalación central.

Llame a "RALRM" solamente desde el OB de alarma que el sistema operativo de la CPU ha arrancado a consecuencia de la alarma de la periferia que se debe analizar

Nota

Si se llama a "RALRM" en un OB cuyo evento de arranque no es una alarma de la periferia, la instrucción proporcionará menos información a través de sus salidas.
Recuerde utilizar diferentes DB de instancia al llamar a "RALRM" en diferentes OB. Si se van a evaluar los datos resultantes de una llamada de "RALRM" fuera del OB de alarma correspondiente, es necesario incluso utilizar un DB de instancia independiente por cada evento de arranque de OB.

Nota

La interfaz de la instrucción "RALRM" es idéntica a la del FB "RALRM" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Llamada de RALRM

"Puede llamar "RALRM" en tres modos de operación diferentes (parámetro MODE), que se muestran en la siguiente tabla.

MODE	RALRM ...
0	... Muestra el componente desencadenante de la alarma en el parámetro de salida ID y describe el parámetro de salida NEW con TRUE.
1	... Escribe en todos los parámetros de salida, sea cual sea el componente desencadenante de la alarma.
2	... Comprueba si el componente especificado en el parámetro de entrada F_ID ha desencadenado la alarma. • En caso negativo, NEW adquiere el valor FALSE. • En caso afirmativo, NEW adquiere el valor TRUE, y se escriben todos los demás parámetros de salida.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RALRM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	Modo de operación
F_ID	Input	HW_IO	D, L o constante	ID de hardware del módulo o submódulo El número se asigna automáticamente y se guarda en las propiedades del componente o de la interfaz en la configuración hardware.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MLEN	Input	UINT	I, Q, M, D, L o constante	Longitud máxima en bytes de la información de alarma que se va a recibir. Si MLEN = 0 se leen todos los datos especificados en el parámetro AINFO.
NEW	Output	BOOL	I, Q, M, D, L	Se ha recibido una nueva alarma.
STATUS (Página 4755)	Output	DWORD	I, Q, M, D, L	Código de error
ID	Output	HW_IO	I, Q, M, D, L	ID de hardware del módulo o submódulo del que se ha recibido una alarma.
LEN	Output	UINT	I, Q, M, D, L	Longitud de la información de alarma recibida
TINFO (Página 4759)	InOut	VARIANT	I, Q, M, D, L	Área de destino para la información de arranque del OB y la información administrativa
AINFO (Página 4773)	InOut	VARIANT	I, Q, M, D, L	Área de destino para información de encabezado e información adicional de alarma Para AINFO debe reservarse una longitud de por lo menos MLEN bytes.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se selecciona el área de destino (Página 4779)TINFO o AINFO demasiado corta, RALRM no podrá registrar toda la información.

Parámetro STATUS

Descripción

El parámetro de salida STATUS contiene información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	• B#16#00, si no hay ningún error • ID de función de DPV1-PDU: En caso de error se emite B#16#80 (en caso de error al leer un registro B#16#DE y al escribir un registro B#16#DF). Si no se utiliza ningún elemento de protocolo DPV1 se emite B#16#C0.
STATUS[2]	Error_Decode	Ubicación de la identificación de error
STATUS[3]	Error_Code_1	Identificación de error
STATUS[4]	Error_Code_2	Extensión específica de fabricante de la identificación de error

Elemento de campo STATUS[2]

STATUS[2]puede adoptar los siguientes valores:

Error_Decode (B#16#...)	Origen	Significado
De 00 a 7F	CPU	Ningún error ni advertencia
80	DPV1	Error según IEC 61158-6
De 81 a 8F	CPU	B#16#8x indica un error en el x parámetro de llamada de la instrucción.
FE, FF	DP Profile	Error específico de perfil

Elemento de campo STATUS[3]

STATUS[3]puede adoptar los siguientes valores:

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
00	00		Ningún error ni advertencia
70	00	reserved, reject	Primera llamada; no hay ninguna transferencia de registros activa
	01	reserved, reject	Primera llamada; transferencia de registro iniciada
	02	reserved, reject	Llamada intermedia; transferencia de registro ya activa
80	81		El tipo de datos de sistema del parámetro TINFO no cuadra con el entorno de llamada de la instrucción. El tipo de datos de sistema utilizado debe cuadrar con el bloque de organización en el programa de usuario (ejemplo: para un OB de alarma de retardo se requiere el tipo de datos del sistema TI_Delay).
	90	reserved, pass	Dirección lógica inicial no válida
	92	reserved, pass	Tipo no admisible con puntero VARIANT
	93	reserved, pass	El componente DP direccionado mediante ID o F_ID no está configurado.
	96		"RALRM (Página 4753)" no puede proporcionar la información de arranque de OB, la información de administración, la información de cabecera ni la información de alarma adicional. En los OB 4x, 55, 56, 57, 82 y 83 se puede leer de manera asíncrona el telegrama de diagnóstico actual del esclavo DP afectado (información de dirección de información de arranque OB) con la instrucción "DPNRM_DG (Página 4806)".
	A0	read error	Confirmación negativa al leer el módulo
	A1	write error	Confirmación negativa al escribir en el módulo
	A2	module failure	Error de protocolo DP con Layer 2 (p. ej. fallo de esclavo o fallos de bus)

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
	A3	reserved, pass	• PROFIBUS DP: Error de protocolo DP con Direct-Data-Link-Mapper o User-Interface/User • PROFINET IO: Error CM general
	A4	reserved, pass	Fallo de comunicación en PBUS+
	A5	reserved, pass	–
	A7	reserved, pass	El esclavo DP o el módulo está ocupado en otra tarea (error temporal)
	A8	version conflict	El esclavo DP o el módulo notifica versiones no concordantes
	A9	feature not supported	Función no soportada por el esclavo DP o el módulo
	AA a AF	user specific	El esclavo DP o el módulo notifica un error específico de fabricante en su aplicación. Consulte la documentación del fabricante del esclavo DP o el módulo.
	B0	invalid index	El módulo no conoce el registro No se admiten números de registro ≥ 256
	B1	write length error	Error al indicar la longitud: • Con "RALRM (Página 4753)": Error de longitud en AINFO (Página 4773) • Con "RDREC (Página 4746)": Error de longitud en MLEN • Con "WRREC (Página 4748)": Error de longitud en LEN
	B2	invalid slot	El slot configurado no está ocupado.
	B3	type conflict	El tipo de módulo real no coincide con el tipo de módulo configurado
	B4	invalid area	El esclavo DP o el módulo notifica un acceso a un área no autorizada.
	B5	state conflict	El esclavo DP o el módulo no está listo
	B6	access denied	El esclavo DP o el módulo deniega el acceso
	B7	invalid range	El esclavo DP o el módulo notifica un área no autorizada en un parámetro o un valor
	B8	invalid parameter	El esclavo DP o el módulo notifica un parámetro no autorizado
	B9	invalid type	El esclavo DP o el módulo notifica un tipo no autorizado Con "RDREC (Página 4746)": Búfer demasiado pequeño (no es posible leer porciones) Con "WRREC (Página 4748)": Búfer demasiado pequeño (no es posible escribir porciones)
	BA a BF	user specific	El esclavo DP o el módulo notifica un error específico de fabricante en el acceso. Consulte la documentación del fabricante del esclavo DP o el módulo.

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
	C0	read constrain conflict	Con "WRREC (Página 4748)": Los datos solo pueden escribirse en el estado STOP de la CPU. Nota: En consecuencia, el programa de usuario no puede escribir. Solo es posible escribir datos online con la PG/el PC. Con "RDREC (Página 4746)": El registro está cargado en el módulo, pero todavía no existen datos o los datos solo pueden leerse en el estado STOP de la CPU. Nota: Si los datos solo pueden leerse en el estado STOP de la CPU, el programa de usuario no puede evaluarlos. En tal caso solo es posible leer los datos online con la PG/el PC.
	C1	write constrain conflict	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.
	C2	resource busy	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.
	C3	resource unavailable	En este momento los recursos necesarios están ocupados.
	C4		Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe si hay fuentes de perturbación en la instalación.
	C5		El esclavo DP o el módulo no está disponible.
	C6		Transferencia del registro cancelada debido a una interrupción de la clase de prioridad
	C7		Interrupción de la petición debido a rearranque completo (en caliente) o arranque en frío del maestro DP
	C8 a CF		El esclavo DP o el módulo notifica a sus recursos un error específico de fabricante. Consulte la documentación del fabricante del esclavo DP o el módulo.
	Dx	user specific	Específico de esclavo DP. Ver descripción del esclavo DP.
81	00 a FF		Error en el primer parámetro de llamada (con "RALRM (Página 4753)": MODE)
	00		Modo de operación no admisible
82	00 a FF		Error en el segundo parámetro de llamada
:	:		:
88	00 a FF		Error en el octavo parámetro de llamada (con "RALRM (Página 4753)": TINFO (Página 4759))
	01		Identificación de sintaxis incorrecta
	23		Rebase de la cantidad o zona de destino demasiado pequeña
	24		Identificación de zona incorrecta
	32		N.º DB/DI fuera del área de usuario
	3A		El N.º DB/DI es CERO en la identificación de zona DB/DI o el DB/DI especificado no existe
89	00 a FF		Error en el noveno parámetro de llamada (con "RALRM (Página 4753)": AINFO (Página 4773))

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
	01		Identificación de sintaxis incorrecta
	23		Rebase de la cantidad o zona de destino demasiado pequeña
	24		Identificación de zona incorrecta
	32		N.º DB/DI fuera del área de usuario
	3A		El N.º DB/DI es CERO en la identificación de zona DB/DI o el DB/DI especificado no existe
8A	00 a FF		Error en el 10º parámetro de llamada
:	:		:
8F	00 a FF		Error en el 15º parámetro de llamada
FE, FF	00 a FF		Error específico de perfil

Elemento de campo STATUS[4]

STATUS[4] es transferido por el maestro DP a la CPU y a la instrucción en caso de error de DPV1. Sin errores DPV1, el valor se pone a "0" con las siguientes excepciones con "RDREC":

- STATUS[4] contiene la longitud del área de destino extraída de RECORD, en caso de que MLEN > longitud del área de destino extraída de RECORD
- STATUS[4]=MLEN en el caso de que la longitud real del registro < MLEN < longitud de la zona de destino de RECORD
- STATUS[4]=0 en el caso de que hubiese que poner STATUS[4] > 255.

En PROFINET IO, STATUS[4] contiene el valor "0".

Parámetro TINFO

Estructura de datos del área de destino TINFO

La estructura de datos del área de destino TINFO contiene la información de arranque del bloque de organización en el que se llamó a "RALRM".

El área de destino TINFO puede incluir la información de arranque con acceso estándar o acceso optimizado. El formato de la información de arranque en el área de destino TINFO debe corresponderse siempre con el de la información de arranque del respectivo bloque de organización.

- La información de arranque de un OB con acceso estándar figura siempre en los primeros 20 bytes de la sección "Temp" de la interfaz del bloque. En este caso se utiliza la estructura de datos "TI_Classic".
- La información de arranque de un OB con acceso optimizado se guarda en la sección "Input". Con estos OB se debe utilizar una estructura de datos específica del tipo de OB.

Cambiando el acceso a bloque (estándar/optimizado), también cambia la interfaz del bloque.

La siguiente tabla ofrece una lista de las estructuras de datos que se utilizan en función del bloque de organización en el parámetro TINFO.

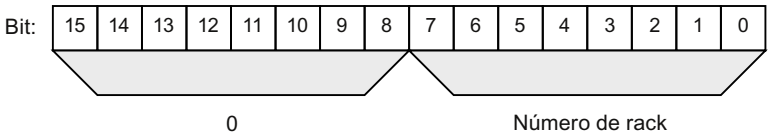
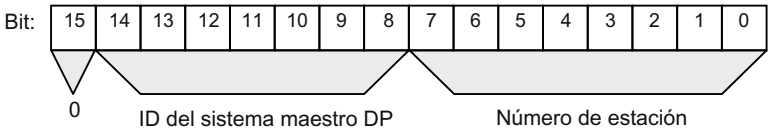
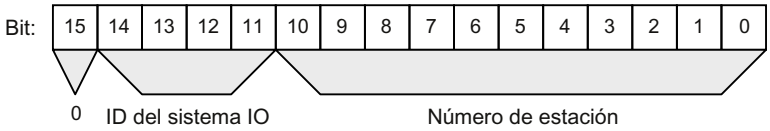
Nombre de la estructura de datos	CPU S7-1200 a partir de versión	CPU S7-1500 a partir de versión	Estructura de datos utilizada para:
Estructura de datos para bloques de organización con acceso estándar			
TI_Classic	-	V1	Bloques de organización sin acceso a bloque optimizado
Estructura de datos para bloques de organización con acceso a bloque optimizado			
TI_ProgramCycle	V2	V1	OB de ciclo (Program cycle)
TI_Startup	V2	V1	OB de arranque (Startup)
TI_Delay	V2	V1	OB de alarma de retardo (Time delay interrupt)
TI_Cyclic	V2	V1	OB de alarma cíclica (Cyclic interrupt)
TI_HWInterrupt	V2	V1	OB de alarma de proceso (Hardware interrupt)
TI_TimeError	V2	V1	OB de error de tiempo (Time error interrupt)
TI_DiagnosticInterrupt	V2	V1	OB de alarma de diagnóstico (Diagnostic error interrupt)
TI_PlugPullModule	V2	V1	OB de cambio de módulo (Pull or plug of modules)
TI_StationFailure	V2	V1	OB de fallo del rack (Rack or station failure)
TI_ProgIOAccessError	V2	V1	• OB de error de programación (Programming error) • OB de error de acceso a la periferia (IO access error)
TI_TimeOfDay	V2	V1	OB de alarma horaria (Time of day)
TI_SynchCycle	-	V1	OB de alarma de sincronismo (Synchronous Cycle)
TI_Submodule	V2	V1	• OB de alarma de estado (Status) • OB de alarma de actualización (Update) • OB para la alarma específica del fabricante o perfil (Profile)

Estructura de datos para bloques de organización con acceso estándar

La estructura de datos TI_Classic está organizada de la manera siguiente:

Parámetro	Tipo de datos	Byte	Descripción
TI_Classic			Estructura de datos para bloques de organización sin acceso a bloque optimizado
Bytes 0 a 19: información de arranque del OB en el que se llamó a "RALRM".*			
EV_CLASS	BYTE	0	Clase de evento Ejemplo OB1: • Bits 0-3: identificador del evento (1 = evento entrante) • Bits 4-7: clase de evento (1 = clase de evento 1)
EV_NUM	BYTE	1	Número de evento (en función del tipo de OB) Ejemplo OB1 (SCAN_1): • SCAN_1 = 1 en primera llamada • SCAN_1 = 3 en todas las demás llamadas
PRIORITY	BYTE	2	Clase de prioridad
NUM	BYTE	3	Número de OB

Parámetro	Tipo de datos	Byte	Descripción
TYP2_3	BYTE	4	Información adicional En los BYTES "TYP2_3" y "TYP1" se guardan diferentes informaciones en función del tipo de OB utilizado. Encontrará información al respecto en la documentación de los respectivos bloques de organización. Ejemplo (OB1): • TYP2_3: OB1_RESERVED_1 (reservado) • TYP1: OB1_RESERVED_2 (reservado)
TYP1	BYTE	5	
ZI1	WORD	6 a 7	Información adicional En "ZI1" se guardan diferentes informaciones en función del tipo de OB utilizado. Encontrará información al respecto en la documentación de los respectivos bloques de organización. Ejemplo (OB1): • ZI1: OB1_PREV_CYCLE (tiempo de ejecución del ciclo anterior en ms)
ZI2_3	DWORD	8 a 11	Información adicional En "ZI2_3" se guardan diferentes informaciones en función del tipo de OB utilizado. Encontrará información al respecto en la documentación de los respectivos bloques de organización. Ejemplo (OB1): • ZI2: OB1_MIN_CYCLE (tiempo de ciclo mínimo (ms) desde el último arranque) • ZI3: OB1_MAX_CYCLE (tiempo de ciclo máximo (ms) desde el último arranque)
OB_DATE_TIME	DATE_AND_TIME (DT)	de 12 a 19	Fecha y hora en las que se solicitó el OB.
Bytes 20 y 21: información de dirección			

Parámetro	Tipo de datos	Byte	Descripción
address	WORD	20 y 21	Información de dirección como en CPU S7-300/400: • En caso de configuración centralizada, el número de rack (0-31):  • En caso de configuración descentralizada con PROFIBUS DP: – ID del sistema maestro DP (1-31) – Número de estación (0-127).  • En caso de configuración descentralizada con PROFINET IO: – Las dos últimas posiciones de la ID del sistema PROFINET IO (0-15); para obtener la ID del sistema PROFINET IO completa, sume 100 (decimal) – Número de estación (0-2047) 
Bytes 22 a 31: Información administrativa			
slv_prfl	BYTE	22	Perfil de esclavo como en CPU S7-300/400: • En configuración centralizada: 0 (registro 0 o registro 1) • En configuración descentralizada: – bits de 0 a 3: Tipo de esclavo - 0000: DP (estructura registro 0) - 0001: DPS7 (estructura registro 0 o registro 1) - 0010: DPS7 V1 (estructura registro 0 o registro 1) - 0011: DPV1 (estructura según estándar PROFIBUS DP) - 0100 – 0111: Reservado - 1000: PROFINET IO (estructura según estándar PROFINET IO) - a partir de 1001: reservado – bits de 4 a 7: tipo de perfil (reservado)

Parámetro		Tipo de datos	Byte	Descripción
	intr_type	BYTE	23	Tipo de información de alarma como en CPU S7-300/400: • En configuración centralizada: 0 • En configuración descentralizada: – bits de 0 a 3: Tipo de información de alarma - 0000: la alarma procede de un módulo descentralizado configurado - 0001: alarma de un esclavo no DPV1/un dispositivo no IO o de un slot no configurado - 0010: alarma generada en la CPU - a partir de 0011: reservado – Bits de 4 a 7: versión de la estructura - 0000: Inicial - a partir de 0001: reservado
	flags1	BYTE	24	Flags de la interfaz de maestro PROFIBUS DP/interfaz de controlador PROFINET IO como en CPU S7-300/400: • En configuración centralizada: 0 • En configuración descentralizada: – Bit 0 = 0: Alarma procedente de una interfaz integrada (PROFINET IO o PROFIBUS DP) – Bit 0 = 1: Alarma procedente de una interfaz externa (PROFINET IO o PROFIBUS DP) – Bits de 1 a 7: reservado
	flags2	BYTE	25	Flags de la interfaz de maestro PROFIBUS DP/interfaz de controlador PROFINET IO como en CPU S7-300/400: • En configuración centralizada: 0 • En caso de configuración descentralizada con PROFIBUS DP: – Bit 0: EXT_DIAG_FLAG del telegrama de diagnóstico, o 0 si este bit no está presente en la alarma. El bit es 1 si el esclavo DP tiene un fallo. – Bits de 1 a 7: reservado • En caso de configuración descentralizada con PROFINET IO: – Bit 0: ARDiagnosisstate o 0, si no hay información en la alarma. El bit es 1 si el dispositivo IO tiene un fallo. – Bits de 1 a 7: reservado
	id	UINT	26 y 27	Información administrativa • En configuración centralizada: 0 • En caso de configuración descentralizada con PROFIBUS DP: Número de identificación de PROFIBUS como identificación unívoca del esclavo PROFIBUS DP • En caso de configuración descentralizada con PROFINET IO: Número de identificación del dispositivo PROFINET IO como identificación unívoca del dispositivo
	manufacturer	UINT	28 y 29	Identificación de fabricante (solo en caso de configuración descentralizada con PROFINET IO).
	instance	UINT	30 y 31	Número de identificación de la instancia (solo en caso de configuración descentralizada con PROFINET IO).
* La información de arranque depende del OB utilizado. La información de arranque del respectivo tipo de OB se encuentra en la interfaz o la documentación del OB.				

Estructuras de datos para bloques de organización con acceso a bloque optimizado

Las estructuras de datos para bloques de organización con acceso a bloque optimizado están organizadas de la manera siguiente:

- Bytes 0 a 3: formato de información de arranque, clase y número del OB llamado (misma organización para todas las estructuras de datos).
- Bytes 4 a 19: información de arranque optimizada (estructura en función del tipo de OB). Los datos de los bytes 4 a 19 se corresponden en organización y contenido con la interfaz del OB correspondiente.
- Bytes 20 a 31: En determinados OB, adicionalmente información administrativa y de dirección. Los datos de los bytes 20 a 31 se corresponden con los datos de los bytes 20 a 31 de la estructura de datos TI_Classic.

En las siguientes tablas se describe la organización de las estructuras de datos.

Tabla 9-42 **OB de ciclo: estructura de datos TI_ProgramCycle**

Parámetro		Tipo de datos	Byte	Descripción
TI_ProgramCycle				Estructura de datos del OB de ciclo
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 1)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	Initial_Call	BOOL	4	= TRUE en la primera llamada del OB con: • transición de STOP o PARADA a RUN • Tras recargar
	Remanence	BOOL	5	= TRUE si hay datos remanentes disponibles.

Tabla 9-43 **OB de arranque: estructura de datos TI_Startup**

Parámetro		Tipo de datos	Byte	Descripción
TI_Startup				Estructura de datos del OB de arranque
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 100)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	LostRetentive	BOOL	4	= TRUE si se han perdido los contenidos de las áreas de datos remanentes.
	LostRTC	BOOL	5	= TRUE si se ha perdido la hora del reloj en tiempo real.

Tabla 9-44 **OB de alarma de retardo: estructura de datos TI_Delay**

Parámetro		Tipo de datos	Byte	Descripción
TI_Delay				Estructura de datos del OB de alarma de retardo
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 20)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	Sign	WORD	4	Identificador de usuario: parámetro de entrada SIGN de la llamada de la instrucción "SRT_DINT (Página 4899)".

Tabla 9-45 **OB de alarma cíclica: estructura de datos TI_Cyclic**

Parámetro		Tipo de datos	Byte	Descripción
TI_Cyclic				Estructura de datos del OB de alarma cíclica
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 30)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	Initial_Call	BOOL	4	= TRUE en la primera llamada del OB • En la transición de STOP o PARADA a RUN • Tras recargar
	Event_Count	INT	6	Número de eventos de arranque rechazados desde el último arranque de este OB.

Tabla 9-46 **OB de alarma de proceso: estructura de datos TI_HWInterrupt**

Parámetro		Tipo de datos	Byte	Descripción
TI_HWInterrupt				Estructura de datos del OB de alarma de proceso
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 40)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	LADDR	HW_IO	4	Identificador de hardware del módulo que dispara la alarma de proceso.
	USI	WORD	6	Identificador para ampliaciones futuras (irrelevante para el usuario).
	IChannel	USINT	8	Número del canal que dispara la alarma de proceso.
	EventType	BYTE	9	Identificador del tipo de evento que dispara la alarma (p. ej., flanco ascendente). Este identificador se indica en la descripción del módulo correspondiente.
	address	WORD	20	Ver parámetro "address" de la estructura de datos TI_Cassic..
	slv_prfl	BYTE	22	Ver parámetro "slv_prfl" de la estructura de datos TI_Cassic.
	intr_type	BYTE	23	Ver parámetro "intr_type" de la estructura de datos TI_Cassic.
	flags1	BYTE	24	Ver parámetro "flags1" de la estructura de datos TI_Cassic.

Parámetro		Tipo de datos	Byte	Descripción
	flags2	BYTE	25	Ver parámetro "flags2" de la estructura de datos TI_Cassic.
	id	UNIT	26	Ver parámetro "id" de la estructura de datos TI_Cassic.
	manufacturer	UNIT	28	Ver parámetro "manufacturer" de la estructura de datos TI_Cassic.
	instance	UNIT	30	Ver parámetro "instance" de la estructura de datos TI_Cassic.

Tabla 9-47 **OB de error de tiempo: estructura de datos TI_TimeError**

Parámetro		Tipo de datos	Byte	Descripción
TI_TimeError				Estructura de datos del OB de error de tiempo
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 80)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	Csg_OBnr	OB_ANY	4	Número del OB que se estaba procesando cuando se produjo el error de tiempo.
	Fault_ID	BYTE	7	Código de error. Valores posibles: • B#16#01: rebase por exceso del tiempo de ciclo • B#16#02: el OB solicitado todavía está procesándose. • B#16#05: alarma horaria expirada debido a salto de hora. • B#16#06: alarma horaria expirada al reanudar RUN después de PARADA. • B#16#07: desbordamiento del búfer de solicitud de OB para la clase de prioridad actual. • B#16#08: error de tiempo en alarma de sincronismo. • B#16#09: pérdida de alarmas por sobrecarga de alarmas • B#16#0B: error de tiempo en alarma de sincronismo tecnológica
	Csg_Prio	UNIT	8	Prioridad del OB que se estaba procesando cuando se produjo el error de tiempo.

Tabla 9-48 **OB de alarma de diagnóstico: estructura de datos TI_DiagnosticInterrupt**

Parámetro		Tipo de datos	Byte	Descripción
TI_DiagnosticInterrupt				Estructura de datos del OB de alarma de diagnóstico
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 82)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	LADDR	HW_ANY	4	Identificador de hardware del objeto de hardware que ha disparado la alarma de diagnóstico.

Parámetro		Tipo de datos	Byte	Descripción
	IO_State	WORD	6	Estado del objeto de hardware: • Bit 0: bueno • Bit 1: desactivado • Bit 2: mantenimiento necesario • Bit 3: mantenimiento solicitado • Bit 4: error • Bit 5: no accesible • Bit 6: cualificado • Bit 7: no disponible
	Channel	UINT	8	Número de canal
	MultiError	BOOL	10	= TRUE si existe más de un error.
	address	WORD	20	Ver parámetro "address" de la estructura de datos TI_Cassic..
	slv_prfl	BYTE	22	Ver parámetro "slv_prfl" de la estructura de datos TI_Cassic.
	intr_type	BYTE	23	Ver parámetro "intr_type" de la estructura de datos TI_Cassic.
	flags1	BYTE	24	Ver parámetro "flags1" de la estructura de datos TI_Cassic.
	flags2	BYTE	25	Ver parámetro "flags2" de la estructura de datos TI_Cassic.
	id	UINT	26	Ver parámetro "id" de la estructura de datos TI_Cassic.
	manufacturer	UINT	28	Ver parámetro "manufacturer" de la estructura de datos TI_Cassic.
	instance	UINT	30	Ver parámetro "instance" de la estructura de datos TI_Cassic.

Tabla 9-49 **OB de cambio de módulo: estructura de datos** TI_PlugPullModule

Parámetro		Tipo de datos	Byte	Descripción
TI_PlugPullModule				Estructura de datos del OB de cambio de módulo
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 83)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	LADDR	HW_IO	4	Identificador de hardware del módulo o submódulo afectado
	Event_Class	BYTE	6	• B#16#38: (sub)módulo insertado • B#16#39: (sub)módulo desenchufado o no responde

Parámetro		Tipo de datos	Byte	Descripción
	Fault_ID	BYTE	7	Código de error El código de error indica qué evento ha provocado el arranque del OB de cambio de módulo. • Si Event_Class = B#16#38 - (sub)módulo enchufado: – B#16#54: submódulo PROFINET IO enchufado y coincide con el submódulo parametrizado – B#16#55: submódulo PROFINET IO enchufado, pero no coincide con el submódulo parametrizado – B#16#56: submódulo PROFINET IO enchufado, pero error en parametrización del módulo – B#16#57: submódulo o módulo PROFINET IO enchufado, pero con fallo o mantenimiento – B#16#58: submódulo PROFINET IO, error de acceso eliminado • Si Event_Class = B#16#39 - (sub)módulo desenchufado o no responde: – B#16#51: módulo PROFINET IO desenchufado – B#16#54: submódulo PROFINET IO desenchufado
	address	WORD	20	Ver parámetro "address" de la estructura de datos TI_Cassic..
	slv_prfl	BYTE	22	Ver parámetro "slv_prfl" de la estructura de datos TI_Cassic.
	intr_type	BYTE	23	Ver parámetro "intr_type" de la estructura de datos TI_Cassic.
	flags1	BYTE	24	Ver parámetro "flags1" de la estructura de datos TI_Cassic.
	flags2	BYTE	25	Ver parámetro "flags2" de la estructura de datos TI_Cassic.
	id	UINT	26	Ver parámetro "id" de la estructura de datos TI_Cassic.
	manufacturer	UINT	28	Ver parámetro "manufacturer" de la estructura de datos TI_Cassic.
	instance	UINT	30	Ver parámetro "instance" de la estructura de datos TI_Cassic.

Tabla 9-50 **OB de fallo de rack: estructura de datos TI_StationFailure**

Parámetro		Tipo de datos	Byte	Descripción
TI_StationFailure				Estructura de datos de OB de fallo de rack
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 86)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	LADDR	HW_Device	4	Identificador de hardware del objeto de hardware defectuoso.
	Event_Class	BYTE	6	• B#16#38: (sub)módulo insertado • B#16#39: (sub)módulo extraído o no responde

Parámetro		Tipo de datos	Byte	Descripción
	Fault_ID	BYTE	7	Código de error El código de error indica qué evento ha provocado el arranque del OB de cambio de módulo. • Si Event_Class = B#16#38 - (sub)módulo enchufado: – B#16#54: submódulo PROFINET IO enchufado y coincide con el submódulo parametrizado – B#16#55: submódulo PROFINET IO enchufado, pero no coincide con el submódulo parametrizado – B#16#56: submódulo PROFINET IO enchufado, pero error en parametrización del módulo – B#16#57: submódulo o módulo PROFINET IO enchufado, pero con fallo o mantenimiento – B#16#58: submódulo PROFINET IO, error de acceso eliminado • Si Event_Class = B#16#39 - (sub)módulo extraído o no responde: – B#16#51: módulo PROFINET IO desenchufado – B#16#54: submódulo PROFINET IO desenchufado
	address	WORD	20	Ver parámetro "address" de la estructura de datos TI_Cassic..
	slv_prfl	BYTE	22	Ver parámetro "slv_prfl" de la estructura de datos TI_Cassic.
	intr_type	BYTE	23	Ver parámetro "intr_type" de la estructura de datos TI_Cassic.
	flags1	BYTE	24	Ver parámetro "flags1" de la estructura de datos TI_Cassic.
	flags2	BYTE	25	Ver parámetro "flags2" de la estructura de datos TI_Cassic.
	id	UINT	26	Ver parámetro "id" de la estructura de datos TI_Cassic.
	manufacturer	UINT	28	Ver parámetro "manufacturer" de la estructura de datos TI_Cassic.
	instance	UINT	30	Ver parámetro "instance" de la estructura de datos TI_Cassic.

Tabla 9-51 **OB de error de programación/OB de error de acceso a la periferia: estructura de datos TI_ProgIOAccessError**

Parámetro		Tipo de datos	Byte	Descripción*
TI_ProgIOAccessError				Estructura de datos de OB de error de programación y OB de error de acceso a la periferia
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB: • = 121 para OB de error de programación • = 122 para OB de error de acceso a la periferia
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	BlockNr	UINT	4	Número del bloque en el que se ha producido el error de programación.
	Reaction	USINT	6	• 0: ignorar error • 1: reemplazar valor incorrecto • 2: saltar comando

Parámetro	Tipo de datos	Byte	Descripción*
Fault_ID	BYTE	7	Código de error • B#16#21: error de conversión BCD • B#16#22: error de longitud de área durante lectura • B#16#23: error de longitud de área durante escritura • B#16#24: error de área durante lectura • B#16#25: error de área durante escritura • B#16#26: error en el n.º del temporizador • B#16#27: error en el n.º del contador • B#16#28: acceso de lectura a un byte, palabra o palabra doble con un puntero cuya dirección de bit es diferente de cero • B#16#29: acceso de escritura a un byte, palabra o palabra doble con un puntero cuya dirección de bit es diferente de cero • B#16#:30: acceso de escritura a un DB global protegido contra escritura • B#16#31: acceso de escritura a un DB de instancia protegido contra escritura • B#16#32: error de número de DB al acceder a un DB global • B#16#33: error de número de DB al acceder a un DB de instancia • B#16#34: error de número en la llamada FC • B#16#35: error de número en la llamada FB • B#16#42: error de acceso a la periferia, lectura • B#16#43: error de acceso a la periferia, escritura • B#16#3A: acceso a un DB no cargado; el número de DB está dentro del rango admisible • B#16#3C: acceso a una FC no cargada; el número de la FC está dentro del rango admisible • B#16#3D: Acceso a una instrucción (SFC) no cargada; el número de la SFC está dentro del rango admisible • B#16#3E: acceso a un FB no cargado; el número de FB está dentro del rango admisible • B#16#3F: acceso a una SFB inexistente; el número de SFB está dentro del rango admisible
BlockType	USINT	8	Tipo de bloque en el que se ha producido el error: • OB: B#16#88 • FC: B#16#8C • FB: B#16#8E
Area	USINT	9	Área de memoria en la que se ha producido el acceso incorrecto: • Datos locales: B#16#40 a 4E, 86, 87, 8E, 8F, C0 a CE • Memoria imagen de proceso de las entradas: B#16#01 • Memoria imagen de proceso de las salidas: B#16#02 • DB de tecnología: B#16#04 • I: B#16#81 • Q: B#16#82 • M: B#16#83 • DB: B#16#84, 85, 8A, 8B

Parámetro		Tipo de datos	Byte	Descripción*
	DBNr	DB_ANY	10	N.º DB si Area = DB (DB global o DB de instancia) Solo relevante con OB de error de programación.
	Csg_OBNr	OB_ANY	12	Número de OB: • 121: OB de error de programación • 122: OB de error de acceso a la periferia
	Csg_Prio	USINT	14	Prioridad de OB
	Width	USINT	15	Tipo de acceso en el que se ha producido el error: • Bit: B#16#00 • Byte: B#16#01 • Word: B#16#02 • DWord: B#16#03 • LWord: B#16#04
* En función del OB cuya información se lee (OB de error de acceso a la periferia u OB de error de programación), solo son posibles determinados valores de salida.				

Tabla 9-52 **OB de alarma horaria: estructura de datos TI_TimeOfDay**

Parámetro		Tipo de datos	Byte	Descripción
TI_TimeOfDay				Estructura de datos del OB de alarma horaria
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 10)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	CaughtUp	BOOL	4	=TRUE, si se ha ejecutado un OB a posteriori por haberse adelantado la hora
	SecondTime	BOOL	5	=TRUE, si se ha llamado a un OB por segunda vez debido al retraso de la hora. Nota: SecondTime se activa una única vez con la hora retrasada.

Tabla 9-53 **OB de alarma de sincronismo: estructura de datos TI_SynchCycle**

Parámetro		Tipo de datos	Byte	Descripción
TI_SynchCycle				Estructura de datos de OB de alarma de sincronismo
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB (= 61)
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	Initial_Call	BOOL	4	= TRUE en la primera llamada del OB con: • transición de STOP o PARADA a RUN • Tras recargar
	IO _System	USINT	5	Número del sistema de periferia descentralizada que ha disparado la alarma

Parámetro		Tipo de datos	Byte	Descripción
	Event_Count	INT	6	• = n: número de ciclos perdidos • = -1: Se ha perdido un número desconocido de ciclos (p. ej. porque el ciclo ha cambiado).
	PIP_Input	BOOL	10	= TRUE: la memoria imagen de proceso correspondiente de las entradas es actual
	PIP_Output	BOOL	11	= TRUE: la memoria imagen de proceso correspondiente de las salidas se transfirió a tiempo a las salidas tras el último ciclo
	SyncCycleTime	LTIME	16	Tiempo de ciclo calculado

Tabla 9-54 **OB de alarma de estado/OB de alarma de actualización/OB para la alarma específica del fabricante o perfil: estructura de datos TI_Submodule**

Parámetro		Tipo de datos	Byte	Descripción
TI_Submodule				Estructura de datos del OB de alarma de estado, OB de alarma de actualización y OB para la alarma específica del fabricante o perfil
	SI_Format	USINT	0	Formato de la información de arranque: • 16#FF: ninguna • 16#FE: información de arranque optimizada
	OB_Class	USINT	1	Clase de OB • = 55 para OB de alarma de estado • = 56 para OB de alarma de actualización • = 57 para OB para la alarma específica del fabricante o perfil
	OB_Nr	UINT	2	Número de OB (1 ... 32767)
	LADDR	HW_IO	4	Dirección de hardware del componente causante de la alarma
	Slot	UINT	6	Número de slot del componente causante de la alarma
	Specifier	WORD	8	Especificador de alarma del telegrama de alarma
	address	WORD	20	Ver parámetro "address" de la estructura de datos TI_Cassic..
	slv_prfl	BYTE	22	Ver parámetro "slv_prfl" de la estructura de datos TI_Cassic.
	intr_type	BYTE	23	Ver parámetro "intr_type" de la estructura de datos TI_Cassic.
	flags1	BYTE	24	Ver parámetro "flags1" de la estructura de datos TI_Cassic.
	flags2	BYTE	25	Ver parámetro "flags2" de la estructura de datos TI_Cassic.
	id	UINT	26	Ver parámetro "id" de la estructura de datos TI_Cassic.
	manufacturer	UINT	28	Ver parámetro "manufacturer" de la estructura de datos TI_Cassic.
	instance	UINT	30	Ver parámetro "instance" de la estructura de datos TI_Cassic.

Parámetro AINFO

Estructura de datos del área de destino AINFO en las alarmas de PROFIBUS DP

Byte	Significado	
de 0 a 3	Información de encabezado y descripción exacta: ver abajo	
4 a 63	Información adicional de alarma: Datos sobre la alarma en cuestión:	
	Descentralizado:	ARRAY[0] a ARRAY[59]

Estructura de la información de encabezado en las alarmas de PROFIBUS DP

Byte	Tipo de datos	Significado		
0	BYTE	Longitud en bytes de la información de alarma recibida		
		Centralizado:	4 a 224	
		Descentralizado:	4 a 63	
1	BYTE	Centralizado:	Reservado	
		Descentralizado:	Identificador del tipo de alarma	
			1: 2: 3: 4: 5: 6: 31	Alarma de diagnóstico Alarma de proceso Alarma de extracción Alarma de inserción Alarma de estado Alarma de actualización Fallo de un aparato de ampliación, un sistema maestro DP o una estación DP
			de 32 a 126:	Alarma específica de fabricante
2	BYTE	Número de slot del componente causante de la alarma		
3	BYTE	Centralizado:	Reservado	
		Descentralizado:	Specifier	
			Bits 0 y 1:	0: No hay más información; 1: Evento entrante, slot con fallo 2: Evento saliente, slot ya sin fallo 3: Evento saliente, slot sigue con fallo
			Bit 2:	Add_Ack
			Bits de 3 a 7:	Número de secuencia

Estructura de datos del área de destino AINFO en las alarmas de PROFINET IO o de la periferia centralizada

Byte	Significado
de 0 a 25	Información de encabezado y descripción exacta: ver abajo
26 a 1431	Información adicional de alarma: Datos de diagnóstico normalizados sobre la alarma en cuestión: ARRAY[0] a ARRAY[1405] Nota: La información adicional de alarma también puede omitirse.

Estructura de la información de encabezado en las alarmas de PROFINET IO o de la periferia centralizada

Byte	Tipo de datos	Significado
0 y 1	WORD	• Bits de 0 a 7: Tipo de bloque • Bits de 8 a 15: Reservado
2 y 3	WORD	Longitud del bloque
4 y 5	WORD	Versión: • Bits de 0 a 7: low byte • Bits de 8 a 15: high byte
6 y 7	WORD	Identificador del tipo de alarma: • 1: Alarma de diagnóstico (entrante) • 2: Alarma de proceso • 3: Alarma de extracción • 4: Alarma de inserción • 5: Alarma de estado • 6: Alarma de actualización • 7: Alarma de redundancia • 8: Controlado por el Supervisor (Controlled_by_Supervisor) • 9: Habilitado por el Supervisor (Released_by_Supervisor) • 10: No se ha insertado el módulo configurado • 11: Retorno del submódulo • 12: Alarma de diagnóstico (saliente) • 13: Aviso de conexión para comunicación directa • 14: Aviso de cambio de vecindario • 15: Aviso de sincronización de ciclo (desde el bus) • 16: Aviso de sincronización de ciclo (desde el dispositivo) • 17: Aviso de componente de red • 18: Aviso de sincronización de reloj (desde el bus) • de 19 a 31: Reservado • de 32 a 127: Alarma específica de fabricante • de 128 a 65535: Reservado
8 a 11	DWORD	API (Application Process Identifier)
de 12 a 13	WORD	Número de slot del componente que dispara la alarma (rango de valores de 0 a 65535)

Byte	Tipo de datos	Significado
14 a 15	WORD	Número de slot de submódulo del componente que dispara la alarma (rango de valores de 0 a 65535)
16 a 19	DWORD	Identificación de módulo; información unívoca del origen de la alarma
20 a 23	DWORD	Identificación de submódulo; información unívoca del origen de la alarma
24 a 25	WORD	Alarm Specifier: • Bits de 0 a 10: Número de secuencia (rango de valores de 0 a 2047) • Bit 11: Diagnóstico de canal: 0: No hay ningún diagnóstico de canal 1: Hay diagnóstico de canal • Bit 12: Estado de diagnóstico específico de fabricante: 0: No hay ninguna información de estado específica de fabricante 1: Hay información de estado específica de fabricante • Bit 13: Estado del diagnóstico en el submódulo: 0: No hay información de estado, se han eliminado todos los errores 1: Hay por lo menos un diagnóstico de canal y/o una información de estado • Bit 14: Reservado • Bit 15: Application Relationship Diagnosis State: – 0: Ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico – 1: Por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

Estructura de la información adicional de alarma en las alarmas de PROFINET IO o de la periferia centralizada

La información adicional de alarma con PROFINET IO depende del especificador de formato. Puede estar compuesta por varios bloques de datos con el mismo o distinto especificador de formato. Son posibles los siguientes especificadores de formato:

- W#16#0000 a W#16#7FFF: Diagnóstico específico de fabricante

Byte	Tipo de datos	Significado
de 0 a 1	WORD	Especificador de formato para la estructura de los datos que siguen como información adicional de alarma: W#16#0000 a W#16#7FFF: Diagnóstico específico de fabricante
de 2 a n	BYTE	Ver manual del fabricante.

- W#16#8000: Diagnóstico de canal
  El diagnóstico de canal se emite en bloques de 6 bytes cada uno. La información adicional de alarma (sin especificador de formato) se emite solo para los canales que presentan fallos.

Byte	Tipo de datos	Significado	
de 0 a 1	WORD	Especificador de formato para la estructura de los datos que siguen como información adicional de alarma: W#16#8000: Diagnóstico de canal	
de 2 a 3	WORD	Número de canal del componente que dispara la alarma (rango de valores de 0 a 65535): • W#16#0000 a W#16#7FFF: Número de canal del submódulo/módulo • W#16#8000: es el representante para todo el submódulo • W#16#8001 a W#16#FFFF: Reservado	
4	BYTE	Bits de 0 a 2:	Reservado
		Bits de 3 a 4:	Tipo de error: • 0: Reservado • 1: Error entrante • 2: Error saliente • 3: Error saliente, hay más errores
		Bits de 5 a 7:	Tipo de canal: • 0: Reservado • 1: Canal de entrada • 2: Canal de salida • 3: Canal de entrada/salida

Byte	Tipo de datos	Significado
5	BYTE	Formato de datos: • B#16#00: Formato de datos libre • B#16#01: Bit • B#16#02: 2 bits • B#16#03: 4 bits • B#16#04: Byte • B#16#05: Palabra • B#16#06: Palabra doble • B#16#07: 2 palabras dobles • B#16#08 a B#16#FF: Reservado
6 a 7	WORD	Tipo de error: • W#16#0000: reservado • W#16#0001: Cortocircuito • W#16#0002: Subtensión • W#16#0003: Sobretensión • W#16#0004: Sobrecarga • W#16#0005: Sobretemperatura • W#16#0006: Rotura de hilo • W#16#0007: Rebase del límite superior • W#16#0008: Rebase del límite inferior • W#16#0009: Error • W#16#000A a W#16#000F: Reservado • W#16#0010 a W#16#001F: Específico de fabricante • W#16#0020 a W#16#00FF: Reservado • W#16#0100 a w#16#7FFF: Específico de fabricante • W#16#8000: Hay diagnóstico de dispositivo • W#16#8001 a W#16#FFFF: Reservado No todos los canales soportan todos los tipos de error. Encontrará más detalles en la descripción de los datos de diagnóstico del dispositivo correspondiente.

Nota

La parte de "Número de canal" a "Tipo de error" puede aparecer entre 0 y n veces.

- W#16#8001: MULTIPLE (se transfieren diferentes tipos de información de diagnóstico). En este caso, la información adicional de alarma se transfiere en estructura de bloque con longitud variable.

Byte	Tipo de datos	Significado
de 0 a 1	WORD	Especificador de formato para la estructura de los datos que siguen como información adicional de alarma: W#16#8001: Diagnóstico específico del fabricante y/o diagnóstico de canal
de 2 a 3	WORD	Tipo de bloque
4 a 5	WORD	Longitud del bloque
6	BYTE	Versión: high byte
7	BYTE	Versión: low byte
8 a 11	DWORD	API (solo si el low byte de la versión = 1)
de 12 a 13	WORD	Número de slot
14 a 15	WORD	Número de subslot
16 a 17	WORD	Número de canal
18 a 19	WORD	Propiedades de canal
20 a 21	WORD	Identificación de formato: • W#16#0000 a W#16#7FFF: Diagnóstico específico de fabricante • W#16#8000: Diagnóstico de canal • W#16#8002: Diagnóstico avanzado de canal • W#16#8003: Diagnóstico avanzado de canal graduado • W#16#8004 a W#16#80FF: Reservado
de 22 a n	BYTE	Datos en función del especificador de formato

Nota

La parte a partir de "Tipo de bloque" puede aparecer entre 1 y n veces.

- W#16#8002: Diagnóstico avanzado de canal

Byte	Significado
de 0 a 1	Identificación de formato W#16#8002
de 2 a 3	Número de canal
4 a 5	Propiedades de canal
6 a 7	Tipo de error
8 a 9	Valor de error adicional
10 a 13	Información adicional de error

- W#16#8003: Diagnóstico avanzado de canal graduado

Byte	Significado
de 0 a 1	Identificación de formato W#16#8003
de 2 a 3	Número de canal
4 a 5	Propiedades de canal
6 a 7	Tipo de error
8 a 9	Valor de error adicional
10 a 13	Información adicional de error
14 a 17	Qualified Channel Qualifier

- W#16#8100: Información de mantenimiento

Byte	Significado
de 0 a 1	Identificación de formato W#16#8100
de 2 a 3	Tipo de bloque
4 a 5	Longitud del bloque
6 a 7	Versión del bloque
8 a 9	Reservado
10 a 13	Estado de mantenimiento

Nota

Encontrará información más detallada sobre la estructura de la información adicional de alarma en el *Manual de programación SIMATIC PROFINET IO De PROFIBUS DP a PROFINET IO* y en la versión más reciente de la norma IEC 61158-6-10-1.

Área de destino TINFO y AINFO

Área de destino TINFO y AINFO

Dependiendo del OB en el que se llame "RALRM (Página 4753)", las áreas de destino TINFO y AINFO solo se escriben parcialmente. La tabla siguiente describe la información que se registra en cada caso.

Tipo de alarma	Clase de OB	TINFO Información de arranque del OB	TINFO Información administrativa	AINFO Información de encabezado	AINFO Información de alarma adicional	
Alarma de proceso	4x	Sí	Sí	Sí	Centralizado:	No
					Descentralizado:	Igual que la proporcionada por el esclavo PROFIBUS DP/dispositivo PROFINET IO
Alarma de estado	55	Sí	Sí	Sí	Sí	Sí
Alarma de actualización	56	Sí	Sí	Sí	Sí	Sí

Tipo de alarma	Clase de OB	TINFO Información de arranque del OB	TINFO Información administrativa	AINFO Información de encabezado	AINFO Información de alarma adicional	
Alarma específica de fabricante	57	Sí	Sí	Sí	Sí	Sí
Error de redundancia de la periferia	70	Sí	Sí	No	No	No
Alarma de diagnóstico	82	Sí	Sí	Sí	Centralizado:	Estructura según la norma PROFINET IO
					Descentralizado:	Igual que la proporcionada por el esclavo PROFIBUS DP/dispositivo PROFINET IO
Alarma de extracción/ inserción	83	Sí	Sí	Sí	Centralizado:	No
					Descentralizado:	Igual que la proporcionada por el esclavo PROFIBUS DP/dispositivo PROFINET IO
Alarma de extracción especial: Controlada por el supervisor	83	Sí	Sí	Sí	Solo PROFINET IO	
Alarma de inserción especial: Habilitada por el supervisor	83	Sí	Sí	Sí	Solo PROFINET IO	
Se ha insertado un módulo no configurado	83	Sí	Sí	Sí	Solo PROFINET IO	
Fallo de rack/ fallo de estación	86	Sí	Sí	No	No	
... Todos los demás OBs		Sí	No	No	No	

D_ACT_DP: Activar / desactivar esclavos DP

Descripción

La instrucción "D_ACT_DP" permite desactivar y volver a activar de forma selectiva esclavos DP o dispositivos PROFINET IO configurados. Además es posible averiguar, para cada esclavo DP o dispositivo PROFINET IO utilizado, si el componente en cuestión está activado o desactivado en un determinado momento.

Si se desactiva con esta instrucción una gateway del tipo IE/PB Link PN IO, se desactivarán también todos los esclavos PROFIBUS DP conectados a ella. Estas desactivaciones se notifican.

Esta instrucción no se puede usar en aparatos de campo PROFIBUS PA que estén conectados mediante DP/PA Link a un sistema maestro DP.

Nota

Mientras están activas una o varias peticiones "D_ACT_DP", no es posible cargar de la programadora a la CPU ninguna configuración modificada (en el marco de CiR).
Durante la carga de una configuración modificada de la programadora a la CPU durante el funcionamiento (CiR), la CPU rechaza la activación de una petición "D_ACT_DP".

Para procesar una petición de desactivación o activación se requieren varios pasos por el punto de control de ciclo. Por ello no es posible esperar la finalización de una petición de este tipo dentro de un bucle programado.

Funcionamiento

"D_ACT_DP" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "D_ACT_DP" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Campo de aplicación

Si en una de las CPU se configuran esclavos DP o dispositivos PROFINET IO que no existen realmente o que actualmente no se necesitan, la CPU accederá igualmente con regularidad a dichos esclavos DP o dispositivos PROFINET IO. Una vez desactivados dichos componentes, la CPU no vuelve a acceder a ellos. Con PROFIBUS DP, esto permite obtener el ciclo de bus DP más rápido posible, y evita además la aparición de los correspondientes eventos de error.

Ejemplos

En la fabricación de máquinas en serie, el fabricante dispone de una gran cantidad de opciones de máquina. Sin embargo, cada máquina suministrada contiene una combinación única de opciones seleccionadas.

Todas las opciones de máquina posibles son configuradas por el fabricante como esclavos DP o dispositivos PROFINET IO, a fin de poder crear y llevar un programa de usuario común para todas las opciones posibles. Con "D_ACT_DP" es posible desactivar en el arranque de la máquina todos los esclavos DP o dispositivos PROFINET IO que no estén presentes en la configuración real.

Algo similar sucede en el caso de las máquinas herramienta, que ofrecen una amplia gama de herramientas pero solo utilizan unas pocas en cada momento concreto. En estos casos, las herramientas se configuran como esclavos DP o dispositivos PROFINET IO. Con ayuda de "D_ACT_DP", el programa de usuario activa las herramientas requeridas en cada momento y desactiva las que no se van a utilizar hasta más adelante.

Identificación de una petición

Si se ha lanzado una petición de desactivación o activación y se vuelve a llamar "D_ACT_DP" antes de que finalice, el comportamiento posterior de la instrucción depende esencialmente de si la nueva llamada corresponde a la misma petición. Si el parámetro de entrada LADDR coincide, la llamada se considera una llamada consecutiva.

Desactivar esclavos DP o dispositivos PROFINET IO

Si se desactiva un esclavo DP o un dispositivo PROFINET IO con "D_ACT_DP", sus salidas de proceso se ajustan a los valores sustitutivos configurados, o a 0 (estado seguro). A partir de entonces, el correspondiente maestro DP o controlador PROFINET IO no volverá a llamar ese componente. Los esclavos DP o dispositivos PROFINET IO desactivados no se identifican como averiados o ausentes en los LEDs de error del maestro DP o controlador PROFINET IO ni de la CPU.

La memoria imagen de proceso de las entradas de los esclavos DP o dispositivos PROFINET IO desactivados se actualiza con 0, es decir, se actúa como si se tratase de esclavos DP o dispositivos PROFINET IO averiados.

Si en el programa se accede mediante acceso directo a los datos útiles de un esclavo DP o dispositivo PROFINET IO desactivado anteriormente, se llama el OB de error de acceso a la periferia y se guarda en el búfer de diagnóstico el correspondiente evento de arranque. Si mediante una instrucción (p. ej. "RD_REC (Página 4785)") se accede a un esclavo DP o dispositivo PROFINET IO desactivado, se recibe en RET_VAL la misma información de error que con un esclavo DP o dispositivo PROFINET IO no disponible.

La desactivación de un esclavo DP o un dispositivo PROFINET IO no ocasiona el arranque del OB de error de ejecución del programa, aunque las entradas y salidas de este pertenezcan a la memoria imagen de proceso que debe actualizarse por el sistema. Tampoco se realiza un registro en el búfer de diagnóstico.

Si una estación DP o PNIO falla después de haberla desactivado con "D_ACT_DP", el sistema operativo no reconoce el fallo.

Para PROFIBUS DP se aplica lo siguiente: Si se desea desactivar un esclavo DP que participa como emisor en la comunicación directa, se recomienda desactivar en primer lugar los receptores que comparten los datos que el emisor envía a su maestro DP. Una vez hecho esto ya se puede desactivar el emisor.

Activar esclavos DP o dispositivos PROFINET IO

Si se vuelve a activar un esclavo DP o dispositivo PROFINET IO con "D_ACT_DP", este componente es configurado y parametrizado por el correspondiente maestro DP o controlador PROFINET IO (igual que cuando se recupera una estación DP o PROFINET IO tras un fallo). La activación ha finalizado cuando el componente está ya en disposición de transferir datos útiles.

La activación de un esclavo DP o un dispositivo PROFINET IO no ocasiona el arranque del OB de error de ejecución del programa, aunque las entradas y salidas de este pertenezcan a la memoria imagen de proceso que debe actualizarse por el sistema. Tampoco se realiza un registro en el búfer de diagnóstico.

Si intenta activar con "D_ACT_DP" un esclavo DP o dispositivo PROFINET IO no accesible (p. ej. porque se ha desconectado físicamente del bus), la instrucción emite el código de error

W#16#80A7 una vez transcurrido el tiempo de parametrización configurado para la periferia descentralizada. El esclavo DP o dispositivo PROFINET IO está activado y la inaccesibilidad del esclavo DP o dispositivo PROFINET IO activado registra el aviso correspondiente en el diagnóstico del sistema.

Si el esclavo DP o dispositivo PROFINET IO vuelve a estar accesible a continuación, esto se traducirá en el comportamiento normal del sistema (p. ej. la llamada del OB configurado a tal efecto).

Nota

La activación de un esclavo DP o un dispositivo PROFINET IO puede tomar un tiempo considerable. Para cancelar una petición de activación en curso, debe iniciarse "D_ACT_DP" con el mismo valor para LADDR y MODE = 2. Debe repetirse la llamada de "D_ACT_DP" con MODE = 2 tantas veces como sea necesario hasta que se señalice, mediante RET_VAL = 0, que se ha cancelado la petición de activación.

Si se desea activar esclavos DP que participan en la comunicación directa, se recomienda activar primero los emisores y luego los receptores (oyentes).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "D_ACT_DP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Parámetro de control disparado por nivel REQ=1: Realizar activación o desactivación
MODE	Input	USINT	I, Q, M, D, L o constante	Identificador de petición Valores posibles: • 0: Conocer si el componente llamado está activado o desactivado (salida mediante el parámetro RET_VAL) • 1: Activar esclavo DP o dispositivo PROFINET IO • 2: Desactivar esclavo DP o dispositivo PROFINET IO
LADDR	Input	HW_DEVICE	I, Q, M, D, L o constante	Identificador de hardware del esclavo DP (HW_DPSlave)/dispositivos PROFINET IO (HW_Device) El número puede aplicarse en la vista de redes adoptándose de las propiedades del esclavo DP/dispositivo PROFINET IO o de la ficha "Constantes del sistema" de la tabla de variables estándar. En el caso de que allí se indiquen tanto el identificador para el diagnóstico del dispositivo como el identificador para las transiciones de estado operativo, deberá utilizar el del diagnóstico del dispositivo.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	Identificación de componente activo: • BUSY=1: La petición todavía está activa. • BUSY=0: La petición ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
0001	El esclavo DP o dispositivo PROFINET IO está activado (este código de error solo es posible con MODE = 0.)
0002	El esclavo DP o dispositivo PROFINET IO está desactivado (este código de error solo es posible con MODE = 0.)
7000	Primera llamada con REQ=0. La petición definida mediante LADDR No está activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1. Se ha lanzado la petición definida mediante LADDR; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante). La petición activada todavía está procesándose; BUSY tiene el valor "1".
8090	• No se ha configurado un módulo con la dirección especificada en LADDR. • Se está utilizando la CPU como esclavo I y se ha indicado en LADDR una dirección de este esclavo I.
8092	El proceso de desactivación en curso de un esclavo DP o un dispositivo PROFINET IO (MODE=2) no se puede interrumpir activándolo (MODE=1). Active el componente en otro momento.
8093	La dirección indicada en LADDR no tiene ningún esclavo DP o dispositivo PROFINET IO que se pueda activar o desactivar.
8094	Se ha intentado activar un dispositivo que podría ser interlocutor de un puerto de cambio de herramienta. Sin embargo, en este momento hay otro dispositivo activado en ese puerto de cambio de herramienta. El dispositivo activado permanece activado.
80A0	Error de comunicación entre la CPU y el controlador IO.
80A1	No se ha podido parametrizar el componente llamado. (Este código de error solo es posible con MODE = 1.) Nota: "D_ACT_DP" devuelve esta información de error solo si este componente vuelve a fallar durante la parametrización del esclavo o dispositivo activado. Si solo ha fallado la parametrización de un único módulo, "D_ACT_DP" devuelve la información de error W#16#0000.
80A2	El esclavo DP llamado no responde (esta información de error no existe para dispositivos PROFINET IO. En PROFINET no existe vigilancia de tiempo para este proceso.).
80A3	El maestro DP o controlador PROFINET IO afectado no soporta esta función.
80A4	La CPU no soporta esta función para maestros DP o controladores PROFINET IO externos.

Código de error* (W#16#...)	Explicación
80A6	Error de slot en el esclavo DP o dispositivo PROFINET IO; no se puede acceder a todos los datos útiles (este código de error solo es posible con MODE=1.) Nota: "D_ACT_DP" devuelve esta información de error solo si el componente vuelve a fallar después de la parametrización del componente activado y antes del final de "D_ACT_DP". Si solo hay un módulo no disponible, "D_ACT_DP" devuelve la información de error W#16#0000.
80A7	Activación de un dispositivo no accesible.
80AA	Activación con errores en el esclavo DP o dispositivo PROFINET IO: Diferencias en la configuración
80AB	Activación con errores en el esclavo DP o dispositivo PROFINET IO: Error de parametrización
80AC	Activación con errores en el esclavo DP o dispositivo PROFINET IO: mantenimiento necesario
80C1	"D_ACT_DP" se ha iniciado y se continúa con otra dirección (este código de error es posible si MODE = 1 y si MODE = 2).
80C3	• Error temporal de recursos: En este momento la CPU está procesando el máximo posible de peticiones de activación/desactivación. (Este código de error solo es posible con MODE = 1 y MODE = 2.) • La CPU está recibiendo una configuración modificada. Debido a ello, en este momento no es posible desactivar ni activar esclavos DP o dispositivos PROFINET IO.
80C5	DP: Las peticiones todavía no recogidas por el usuario se descartan en el rearranque.
80C6	PROFINET: Las peticiones todavía no recogidas por el usuario se descartan en el rearranque.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Otras

RD_REC: Leer registro de periferia

Descripción

Esta instrucción permite leer el registro con número RECNUM del componente que se ha direccionado. Para iniciar la lectura, debe ajustarse a "1" el parámetro de entrada REQ . Si la operación de lectura se ha podido ejecutar de inmediato, la instrucción devuelve el valor "0" en el parámetro de salida BUSY . Si BUSY tiene el valor "1", la operación de lectura todavía no ha finalizado.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277). Una vez efectuada correctamente la transferencia de datos, el registro leído se guarda en el área de destino delimitada mediante RECORD .

Nota

Si se lee un registro con número mayor que 1 desde una FM o un CP adquiridos antes de febrero de 1997 (en lo sucesivo denominados "módulos antiguos"), "RD_REC" se comporta de un modo diferente que con un módulo nuevo. Este caso especial se describe en la sección "Uso de FMs y CPs S7-300 antiguos con n.º de registro > 1" (ver abajo).

Si un esclavo DPV1 está configurado mediante archivo GSD (a partir de la rev. 3 de GSD) y la interfaz DP del maestro DP está ajustada como "Compatible con S7", los módulos de E/S no pueden leer en el programa de usuario los registros con "RD_REC". En este caso, el maestro DP direcciona un slot incorrecto (slot configurado + 3).

Solución: Conmutar la interfaz del maestro DP a "DPV1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_REC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de lectura
LADDR	Input	HW_IO	I, Q, M, D, L o constante	ID de hardware del módulo.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 0 a 240)
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Adicionalmente: Longitud en bytes del registro realmente transferido (valores posibles: de +1 a +240), si el área de destino es mayor que el registro transferido y no se ha producido ningún error durante la transferencia.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la lectura todavía no ha finalizado.
RECORD	Output	ANY	I, Q, M, D, L	Área de destino para el registro leído. En caso de procesamiento asíncrono de "RD_REC", debe comprobarse que los parámetros actuales de RECORD presenten la misma información de longitud en todas las llamadas. Solo se admite el tipo de datos BYTE . Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RECORD

Nota

Para asegurarse de que siempre se lea el registro completo, debe seleccionarse un área de destino con una longitud de 241 bytes. Si la transferencia de datos se realiza sin errores, la longitud real del registro se puede leer en RET_VAL .

Uso de FMs y CPs S7-300 antiguos con n.º de registro > 1

Si se desea leer un registro con número mayor que 1 con la instrucción "RD_REC" desde una FM S7-300 antigua o un CP S7-300 antiguo, debe tenerse en cuenta lo siguiente:

- Si el área de destino es mayor que la longitud real del registro deseado, no se registran datos en RECORD . En RET_VAL se escribe W#16#80B1.
- Si el área de destino es menor que la longitud real del registro deseado, la CPU leerá tantos bytes desde el principio del registro como se indique en la longitud especificada en RECORD y los registrará en RECORD . En RET_VAL se escribe "0".
- Si la indicación de longitud de RECORD es igual a la longitud real del registro deseado, la CPU lee el registro en cuestión y lo guarda en RECORD ; RET_VAL se ajusta a "0".

Parámetro RET_VAL

- Si se ha producido un error durante la ejecución de la función, el valor de respuesta contendrá un código de error.
- Si no se ha producido ningún error durante la transferencia, RET_VAL contiene el valor
  - 0 si toda el área de destino se ha llenado con datos del registro seleccionado (aunque el registro esté incompleto).
  - la longitud en bytes del registro realmente transferido (valores posibles: de +1 a +240), si el área de destino es mayor que el registro transferido.

  Nota

  Si se produce el error general W#16#8745, esto significa solo que durante la operación de escritura no ha sido posible acceder a por lo menos un byte. El registro se ha leído correctamente desde el módulo y escrito correctamente en el área de memoria de E/S.

Dentro de la información de error "verdadera" (códigos de error W#16#8xyz) de la tabla siguiente, debe distinguirse entre dos casos:

- Errores temporales (códigos de error de W#16#80A2 a 80A3, 80Cx):
  En este tipo de errores es posible que el error se elimine sin intervención del usuario, es decir, puede bastar con llamar de nuevo la instrucción (varias veces si es necesario). Ejemplo de error temporal: En este momento los recursos necesarios están ocupados (W#16#80C3).
- Errores permanentes (códigos de error W#16#809x, 80A0, 80A1, 80Bx):
  Los errores de este tipo no se pueden eliminar sin intervención del usuario. Solo tiene sentido volver a llamar la instrucción una vez que se haya eliminado el error. Ejemplo de error permanente: Indicación de longitud errónea en RECORD (W#16#80B1).

Nota

Si se transfieren con "WR_REC (Página 4790)" registros a un esclavo DPV1 o se leen registros RD_REC desde un esclavo DPV1 que funciona en modo DPV1, el maestro DP evalúa de la siguiente manera la información de error recibida desde el esclavo:

Si la información de error se encuentra en las áreas de W#16#8000 a W#16#80FF o de W#16#F000 a W#16#FFFF, el maestro DP reenvía la información de error a la instrucción. Si se encuentra fuera de estas áreas, reenvía a la instrucción el valor W#16#80A2 y suspende el esclavo.

Encontrará la descripción de la información de error procedente de los esclavos DPV1 en STATUS[3] Parámetro STATUS (Página 4755).

Parámetros RET_VAL para WR_REC y RD_REC

Código de error* (W#16#...)	Explicación	Restricción
0000	Sin errores	-
7000	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.	-
7001	Primera llamada con REQ=1: Transferencia de datos lanzada; BUSY tiene el valor 1.	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transferencia de datos ya activa; BUSY tiene el valor 1.	Periferia descentralizada
8090	Dirección indicada no válida en el parámetro ADDR.	-
8092	En la referencia ANY se ha indicado un tipo distinto a BYTE.	-
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR e IOID .	-
80A0	Acuse negativo al leer el módulo: Módulo extraído durante la operación de lectura o módulo defectuoso	Solo con "RD_REC"
80A1	Acuse negativo al escribir en el módulo: Módulo extraído durante la operación de escritura o módulo defectuoso	Solo con "WR_REC (Página 4790)"
80A2	• Error de protocolo DP con Layer 2 (p. ej. fallo de esclavo o fallos de bus) • Con ET200S no es posible leer registros en el modo DPV0.	Periferia descentralizada
80A3	Error de protocolo de DP con User-Interface/User	Periferia descentralizada
80A4	Fallo de comunicación en PBUS+	-

Código de error* (W#16#...)	Explicación	Restricción
80B0	• La instrucción no es posible para el tipo de módulo • El módulo no conoce este registro. • El número de registro 241 no es admisible. • Con "WR_REC (Página 4790)" no se permiten los registros 0 ni 1.	-
80B1	La indicación de longitud en el parámetro RECORD es errónea.	• Con "WR_REC (Página 4790)": Longitud errónea • Con "RD_REC" (posible solo si se usan FM S7-300 antiguas o CP S7-300 antiguos):Indicación > longitud registro • Con DPNRM_DG: Indicación < longitud de registro
80B2	El slot configurado no está ocupado.	-
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado	-
80B7	El esclavo DP o el módulo notifica un área no autorizada en un parámetro o un valor.	Solo con "RD_REC"
80C0	Con "WR_REC (Página 4790)": Los datos solo pueden escribirse en el estado STOP de la CPU. Nota: En consecuencia, el programa de usuario no puede escribir. Solo es posible escribir datos online con la PG/el PC. Con "RD_REC": El registro está cargado en el módulo, pero todavía no existen datos o los datos solo pueden leerse en el estado STOP de la CPU. Nota: Si los datos solo pueden leerse en el estado STOP de la CPU, el programa de usuario no puede evaluarlos. En tal caso solo es posible leer los datos online con la PG/el PC. Con "DPNRM_DG (Página 4806)": No hay datos de diagnóstico.	Con "WR_REC (Página 4790)", "RD_REC" o "DPNRM_DG (Página 4806)"
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	-
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	-
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	-
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe si hay fuentes de perturbación en la instalación.	-
80C5	La periferia descentralizada no está disponible.	Periferia descentralizada
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano)	Periferia descentralizada
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)	-
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

WR_REC: Escribir registro en periferia

Descripción

La instrucción "WR_REC" permite transferir el registro RECORD al módulo direccionado.

Los datos que se deben transferir se leen del parámetro RECORD en la primera llamada. Si la transferencia del registro dura más que una llamada, en las llamadas consecutivas de la instrucción (para la misma petición), el contenido del parámetro RECORD ya no es relevante.

Para iniciar la escritura, debe ajustarse a "1" el parámetro de entrada REQ. Si la operación de escritura se ha podido ejecutar de inmediato, la instrucción devuelve en el parámetro de salida BUSY el valor "0". Si BUSY tiene el valor "1", la operación de escritura todavía no ha finalizado.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_REC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	REQ = 1: solicitud de escritura
LADDR	Input	HW_IO	I, Q, M, D, L o constante	ID de hardware del módulo.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro
RECORD	Input	VARIANT	I, Q, M, D, L	Registro
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la escritura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Consulte también: RD_REC: Leer registro de periferia (Página 4785)

Nota

Si aparece el error general W#16#8544, esto solo significa que estaba bloqueado el acceso a por lo menos un byte del área de memoria de E/S que contiene el registro. Se ha proseguido la transferencia de datos.

DPRD_DAT: Leer datos coherentes de un esclavo DP normalizado

Descripción

La instrucción "DPRD_DAT" lee datos coherentes de un módulo E/S.

La instrucción se puede utilizar tanto para módulos del módulo central como para esclavos DP normalizados y dispositivos PROFINET IO.

"DPRD_DAT" es necesaria porque con los comandos de carga que acceden a la periferia o a la memoria imagen de proceso de las entradas solo se puede leer un máximo de cuatro bytes de forma coherente. Para leer datos coherentes también puede usarse, en caso necesario, la memoria imagen de proceso de las entradas. En la documentación de la CPU se puede comprobar si dispone de esta función. Encontrará más información sobre datos coherentes de un esclavo DP normalizado/dispositivo PROFINET IO en el apartado "Coherencia de datos (Página 5244)".

En caso necesario, la instrucción "DPRD_DAT" también se puede utilizar para un área de datos a partir de 1 byte. La longitud máxima de los datos se indica en la documentación de la CPU (p. ej. 64 bytes en una S7-1214).

- Mediante el parámetro LADDR se selecciona el módulo del esclavo DP normalizado/ dispositivo PROFINET IO. Si se produce un error de acceso se notifica el código de error W#16#8090.
- Mediante el parámetro RECORD se define el área de destino de los datos leídos:
  - El área de destino tiene que tener como mínimo la misma longitud que las entradas del módulo seleccionado. Solo se transmiten las entradas; el resto de los bytes no se tienen en cuenta. En un esclavo DP normalizado con estructura modular o con varias ID de DP, la llamada a "DPRD_DAT" solo permite acceder cada vez a los datos de un solo módulo con la ID de hardware configurada. Si se selecciona un área de destino demasiado pequeña, el parámetro RET_VAL devuelve el código de error 80B1.
  - Como tipo de datos se pueden utilizar todas las secuencias de bits y todos los enteros. También está permitida la utilización de estos tipos de datos en una estructura de datos del tipo ARRAY. No se soporta el tipo de datos STRING.
- Si no se ha producido ningún error durante la transferencia de datos, los datos leídos se registran en el área de destino definida en el parámetro RECORD .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPRD_DAT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IO	I, Q, M, L o constante	ID de hardware del módulo del que se deben leer los datos. La ID de hardware figura en la Vista de dispositivos o en las Constantes de sistema de las propiedades del módulo.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
RECORD	Output	VARIANT	I, Q, M, D, L	Área de destino para los datos de usuario leídos.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	• No se ha configurado ningún módulo para la identificación HW especificada, o bien • No se ha tenido en cuenta la limitación de longitud de los datos coherentes, o • No se ha indicado ninguna identificación HW como dirección en el parámetro LADDR .
8092	En el parámetro RECORD se ha indicado un tipo de datos diferente a (Array of) secuencias de bits o enteros.
8093	Para la identificación HW indicada en LADDR no existe ningún módulo DP o ningún dispositivo PROFINET IO del que puedan leerse datos de forma coherente. Este código de error aparece cuando el módulo direccionado a través de LADDR no dispone de entradas.
80A0	Se ha detectado un error de acceso al acceder a la periferia.
80B1	La longitud del área de destino especificada en el parámetro RECORD es más corta que la longitud de datos de usuario configurada.
80C0	Los datos todavía no se han leído.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Parámetro STATUS (Página 4755)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DPWR_DAT: Escribir datos coherentes de un esclavo DP normalizado

Descripción

La instrucción "DPWR_DAT" transfiere los datos del parámetro RECORD de forma coherente al módulo direccionado del módulo central o del esclavo DP normalizado/dispositivo PROFINET IO y, dado el caso, a la memoria imagen de proceso (cuando el área de direcciones afectada del esclavo DP normalizado se ha configurado como área de coherencia en una memoria imagen de proceso).

"DPWR_DAT" es necesaria porque con los comandos de transferencia que acceden a la periferia o a la memoria imagen de proceso de las salidas solo se puede escribir un máximo de cuatro bytes de forma coherente. Para escribir datos coherentes también puede usarse, en caso necesario, la memoria imagen de proceso de las salidas. En la documentación de la CPU se puede comprobar si dispone de esta función. Para escribir datos coherentes, no deben utilizarse al mismo tiempo ambas posibilidades: Utilice o bien "DPWR_DAT" o bien la escritura a través de la memoria imagen de proceso de las salidas. Encontrará más información sobre datos coherentes de un esclavo DP normalizado/dispositivo PROFINET IO en el apartado "Coherencia de datos (Página 5244)". Con un esclavo DP normalizado con estructura modular solo es posible acceder a un módulo del esclavo DP.

⚠ PRECAUCIÓN

Acceso a la periferia

Al utilizar "DPWR_DAT", evite acceder a áreas de la periferia que tengan asignadas memorias imagen parcial de proceso vinculadas a OB6x (alarmas de sincronismo).

En caso necesario, la instrucción "DPRD_DAT" también se puede utilizar para un área de datos a partir de 1 byte. La longitud máxima de los datos se indica en la documentación de su CPU (p. ej. 64 bytes en una S7-1214).

- Mediante el parámetro LADDR se selecciona el esclavo DP normalizado/dispositivo PROFINET IO. Si se produce un error de acceso al módulo direccionado, se visualiza el código de error 8090.
- Mediante el parámetro RECORD se define el área de origen de los datos que se deben escribir:
  - El área de origen tiene que tener como mínimo la misma longitud que las salidas del módulo seleccionado. Solo se transmiten las salidas; el resto de los bytes no se tienen en cuenta. Si el área de origen especificada en el parámetro RECORD tiene una longitud mayor que las salidas del módulo seleccionado, tan solo se transferirán datos hasta la longitud máxima de las salidas. Si el área de origen especificada en el parámetro RECORD tiene una longitud menor que las salidas del módulo configurado, se visualiza el código de error 80B1.
  - Se pueden utilizar los siguientes tipos de datos: Byte, Char, Word, LWord, DWord, Int, UInt, USInt, SInt, LInt, ULInt, DInt, UDInt. También está permitida la utilización de estos tipos de datos en una estructura de datos del tipo ARRAY o STRUCT.
  - No se soporta el tipo de datos STRING.

La transferencia de datos se realiza de forma sincrónica, es decir, con la finalización de la instrucción ha finalizado el proceso de escritura.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPWR_DAT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IO	I, Q, M, L o constante	ID de hardware del módulo en el que se deben escribir los datos. La ID de hardware figura en la Vista de dispositivos o en las Constantes de sistema de las propiedades del módulo.
RECORD	Input	VARIANT	I, Q, M, D, L	Área de origen para los datos de usuario que se deben escribir.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	• No se ha configurado ningún módulo para la identificación HW especificada, o bien • No se ha tenido en cuenta la limitación de longitud de los datos coherentes, o • No se ha indicado ninguna identificación HW en el parámetro LADDR .
8092	En el parámetro RECORD se ha indicado un tipo de datos diferente a (Array of) secuencias de bits o enteros.
8093	Para la ID de hardware indicada en LADDR no existe ningún módulo DP o dispositivo PROFINET IO en el que se puedan escribir datos de forma coherente. Este código de error aparece cuando el esclavo DP normalizado/dispositivo PROFINET IO direccionado a través de LADDR no dispone de salidas.
80A1	Se ha detectado un error de acceso al acceder a la periferia
80B1	La longitud del área de origen indicada en el parámetro RECORD es inferior a las salidas del esclavo DP normalizado/dispositivo PROFINET IO configurado.
80C1	Los datos de la petición de escritura efectuada anteriormente no han sido procesados todavía por el esclavo DP normalizado/dispositivo PROFINET IO.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Parámetro STATUS (Página 4755)

iDevice / iSlave

RCVREC: Recibir registro

Descripción

Un I-Device puede recibir un registro desde un controlador superior. La recepción se efectúa en el programa de aplicación con la instrucción "RCVREC" (receive record).

La instrucción tiene los siguientes modos de operación:

- Comprobar si el I-Device ha recibido una solicitud para la recepción de un registro.
- Preparar un registro en los parámetros de salida.
- Enviar una respuesta al controlador superior.

Para establecer cuál de estos modos de operación debe ejecutar la instrucción, se usa el parámetro de entrada MODE (ver abajo).

El I-Device debe encontrarse en el estado operativo RUN o en el estado operativo ARRANQUE.

Con MLEN se indica el máximo de bytes que se deben recibir. Por ello debe escogerse un área de destino RECORDcon una longitud de por lo menos MLENbytes.

Cuando se ha recibido un registro (MODE=1 o MODE=2), en el parámetro de salida NEW se indica que el registro se ha guardado en RECORD. Asegúrese de que RECORD tenga una longitud suficiente. El parámetro de salida LEN contiene la longitud real en bytes del registro recibido.

Para la respuesta positiva al controlador superior, se deben ajustar CODE1 y CODE2 a cero. Si se desea rechazar el registro recibido, debe enviarse con CODE1 el Error Code 1 y con CODE2 el Error Code 2 de la respuesta negativa al controlador superior.

Nota

Si el I-Device ha recibido una solicitud de recepción de un registro, es necesario haber reconocido la emisión de dicha solicitud dentro de un período determinado. Una vez reconocida la solicitud, debe haberse enviado al controlador superior una respuesta dentro del mismo período. De lo contrario, se produce en el I-Device un error de timeout, a consecuencia del cual el sistema operativo del I-Device envía una respuesta negativa al controlador superior. Para conocer dicho período, consulte los datos técnicos de la CPU.

Cuando se produce un error, el parámetro de salida STATUS contiene la información de error.

Modos de operación

El modo de operación de la instrucción "RCVREC" se define mediante el parámetro de entrada MODE. Esto se explica en la siguiente tabla:

MODE	Significado
0	Comprobar si existe una solicitud para la recepción de un registro. Si en el I-Device existe un registro procedente del controlador superior, la instrucción escribirá solo en los parámetros de salida NEW, SLOT, SUBSLOT, INDEX y LEN. Si se llama la instrucción varias veces consecutivas con MODE=0, los parámetros de salida se refieren siempre a la misma solicitud.
1	Recibir un registro para cualquier subslot del I-Device Si en el I-Device existe un registro procedente del controlador superior para cualquier subslot del I-Device, la instrucción escribe solo en los parámetros de salida y transfiere el registro al parámetro RECORD.
2	Recibir un registro para un determinado subslot del I-Device Si en el I-Device existe un registro procedente del controlador superior para un determinado subslot del I-Device, la instrucción escribe solo en los parámetros de salida y transfiere el registro al parámetro RECORD.
3	Enviar respuesta positiva al controlador superior La instrucción comprueba la solicitud de recepción de registro procedente del controlador superior, acepta el registro existente y envía un acuse positivo al controlador superior.
4	Enviar respuesta negativa al controlador superior La instrucción comprueba la solicitud de recepción de registro procedente del controlador superior, rechaza el registro existente y envía un acuse negativo al controlador superior. El motivo del rechazo puede indicarse en los parámetros de entrada CODE1 y CODE2.

Nota

Una vez llegado el registro (NEW=1), debe llamarse "RCVREC" dos veces más para garantizar que se procese por completo. Al hacerlo debe mantenerse el siguiente orden:

- Primera llamada con MODE=1 o MODE=2
- Segunda llamada con MODE=3 o MODE=4

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RCVREC":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	Modo de operación
F_ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	Subslot en el área de trasferencia del I-Device para el registro que se debe recibir (relevante solo para MODE=2). El high word debe ajustarse siempre a cero.
MLEN	Input	INT*	I, Q, M, D, L o constante	Longitud máxima en bytes del registro que se debe transferir.
CODE1	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 1 (para MODE=4)
CODE2	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 2 (para MODE=4)

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
NEW	Output	BOOL	I, Q, M, D, L	• MODE=0: Se ha recibido un registro nuevo • MODE=1 ó 2: El registro se ha transferido en RECORD
STATUS	Output	DWORD	I, Q, M, D, L	Información de error
SLOT	Output	HW_SUBMODULE	I, Q, M, D, L	Idéntico a F_ID
SUBSLOT	Output	HW_SUBMODULE	I, Q, M, D, L	Idéntico a F_ID
INDEX	Output	UINT	I, Q, M, D, L	Número del registro recibido
LEN	Output	UINT	I, Q, M, D, L	Longitud del registro recibido
RECORD	InOut	VARIANT	I, Q, M, D, L	Área de destino para el registro recibido. Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 Byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
* En el lenguaje de programación AWL utilice el tipo de datos UINT.				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Para la interpretación del parámetro STATUS, véase el capítulo: Parámetro STATUS (Página 4755)

PRVREC: Proveer registro

Descripción

Un I-Device puede recibir desde un controlador superior la solicitud de preparar un registro. La preparación se efectúa en el programa de usuario con la instrucción "PRVREC" (provide record).

La instrucción tiene los siguientes modos de operación:

- Comprobar si el I-Device ha recibido una solicitud para la preparación de un registro.
- Transferencia del registro solicitado al controlador superior.
- Enviar una respuesta al controlador superior.

Para establecer cuál de estos modos de operación debe ejecutar la instrucción, se usa el parámetro de entrada MODE (ver abajo).

El I-Device debe encontrarse en el estado operativo RUN o en el estado operativo ARRANQUE.

Con LEN se indica el máximo de bytes que debe tener el registro que se desea enviar. Por ello debe escogerse un área de destino RECORDcon una longitud de por lo menos LENbytes.

Cuando existe una solicitud de preparación de un registro (MODE=0), el parámetro de salida NEW se ajusta a TRUE.

Si se acepta la solicitud de preparación de registro, para la respuesta positiva al controlador superior debe escribirse en RECORD el registro solicitado, así como ajustar CODE1 y CODE2 a cero. Si se desea rechazar una solicitud de preparación de registro, debe enviarse con CODE1 el Error Code 1 y con CODE2 el Error Code 2 de la respuesta negativa al controlador superior.

Nota

Si el I-Device ha recibido una solicitud de preparación de un registro, es necesario haber reconocido la emisión de dicha solicitud dentro de un período determinado. Una vez reconocida la solicitud, debe haberse enviado al controlador superior una respuesta dentro del mismo período. De lo contrario, se produce en el I-Device un error de timeout, a consecuencia del cual el sistema operativo del I-Device envía una respuesta negativa al controlador superior. Para conocer dicho período, consulte los datos técnicos de la CPU.

Cuando se produce un error, el parámetro de salida STATUS contiene la información de error.

Modos de operación

El modo de operación de la instrucción "PRVREC" se define mediante el parámetro de entrada MODE. Esto se explica en la siguiente tabla:

MODE	Significado
0	Comprobar si existe una solicitud de preparación de un registro. Si en el I-Device existe una solicitud de preparación de registro procedente del controlador superior, la instrucción escribirá solo en los parámetros de salida NEW, SLOT, SUBSLOT, INDEX y RLEN. Si se llama la instrucción varias veces consecutivas con MODE=0, los parámetros de salida se refieren siempre a la misma solicitud.
1	Recibir una solicitud de preparación de registro para cualquier subslot del I-Device Si en el I-Device existe una solicitud de ese tipo procedente del controlador superior para cualquier subslot del I-Device, la instrucción escribe solo en los parámetros de salida.
2	Recibir una solicitud de preparación de registro para un determinado subslot del I-Device Si en el I-Device existe una solicitud de ese tipo procedente del controlador superior para un determinado subslot del I-Device, la instrucción escribe solo en los parámetros de salida.
3	Preparar registro y enviar respuesta positiva al controlador superior La instrucción comprueba la solicitud de preparación de registro procedente del controlador superior, pone a disposición el registro solicitado en RECORD y envía un acuse positivo al controlador superior.
4	Enviar respuesta negativa al controlador superior La instrucción comprueba la solicitud de preparación de registro procedente del controlador superior, la rechaza y envía un acuse negativo al controlador superior. El motivo del rechazo puede indicarse en los parámetros de entrada CODE1 y CODE2.

Nota

Una vez llegada la solicitud (NEW=1), debe llamarse la instrucción dos veces más para garantizar que se procese por completo. Al hacerlo debe mantenerse el siguiente orden:

- Primera llamada con MODE=1 o MODE=2
- Segunda llamada con MODE=3 o MODE=4

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PRVREC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	Modo de operación
F_ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	Subslot en el área de trasferencia del I-Device para el registro que se debe enviar (relevante solo para MODE=2). El high word debe ajustarse siempre a cero.
CODE1	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 1 (para MODE=4)
CODE2	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 2 (para MODE=4)
LEN	Input	UINT	I, Q, M, D, L o constante	Longitud máxima en bytes del registro que se debe enviar.
NEW	Output	BOOL	I, Q, M, D, L	El controlador superior ha solicitado un nuevo registro.
STATUS	Output	DWORD	I, Q, M, D, L	Información de error
SLOT	Output	HW_SUBMODULE	I, Q, M, D, L	Idéntico a F_ID
SUBSLOT	Output	HW_SUBMODULE	I, Q, M, D, L	Idéntico a F_ID
INDEX	Output	UINT	I, Q, M, D, L	Número del registro que se debe enviar
RLEN	Output	UINT	I, Q, M, D, L	Longitud del registro que se debe enviar
RECORD	InOut	VARIANT	I, Q, M, D, L	Registro preparado Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 Byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Para la interpretación del parámetro STATUS, véase el capítulo: Parámetro STATUS (Página 4755)

PROFIBUS

DPSYC_FR: Sincronizar esclavos DP / Congelar entradas

Descripción

Esta instrucción permite sincronizar uno o varios grupos de esclavos DP.

Para ello debe enviarse a los grupos en cuestión uno de los siguientes comandos de control o una combinación de ellos:

- SYNC (emisión y congelación simultáneas de estados de salida en los esclavos DP)
- UNSYNC (anula el comando de control SYNC)
- FREEZE (congelación de estados de entrada en los esclavos DP y lectura de las entradas congeladas)
- UNFREEZE (anula el comando de control FREEZE)

Antes de enviar los comandos de control mencionados, deben haberse dividido en grupos, mediante configuración, los esclavos DP. Es necesario saber qué esclavo DP está asignado a cada grupo y con qué número, así como las características de cada uno de los grupos en lo que respecta al comportamiento de SYNC/FREEZE.

Nota

Debe tenerse en cuenta que los comandos de control SYNC y FREEZE mantienen su validez incluso en caso de rearranque completo o arranque en frío.

También debe tenerse en cuenta que en cada momento solo puede estar lanzada una petición SYNC/UNSYNC y solo una petición FREEZE/UNFREEZE.

Funcionamiento

"DPSYC_FR" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "DPSYC_FR" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Identificación de una petición

Si se ha lanzado una petición SYNC/FREEZE y se ha vuelto a llamar "DPSYC_FR" antes de que finalice, el comportamiento posterior de la instrucción depende esencialmente de si la

nueva llamada corresponde a la misma petición. Si los parámetros de entrada LADDR, GROUP y MODE coinciden entre sí, la llamada se considera una llamada consecutiva.

Escribir en salidas de módulos DP

La escritura en las salidas de los módulos DP se lanza mediante

- Comandos de transferencia a la periferia DP,
- La escritura de la memoria imagen de proceso de las salidas en los módulos (mediante el sistema operativo al final del OB 1 o mediante llamada de la instrucción "UPDAT_PO (Página 4740)"),
- La llamada de la instrucción "DPWR_DAT (Página 4793)".

En el caso normal, el maestro DP transfiere los datos de salida cíclicamente (en el ciclo del bus PROFIBUS DP) a las salidas de los esclavos DP.

Si se desea transferir determinados datos de salida que pueden estar repartidos entre varios esclavos a las salidas hacia el proceso de manera exactamente simultánea, debe enviarse el comando de control SYNC al maestro DP correspondiente con la instrucción "DPSYC_FR".

¿Qué efecto produce SYNC?

Con el comando de control SYNC se ponen en modo de sincronización los esclavos DP de los grupos especificados, es decir, el maestro DP transfiere los datos de salida actuales y ordena a los esclavos DP implicados que congelen las salidas. En los siguientes telegramas de salida, los esclavos DP guardan los datos de salida en un búfer interno; el estado de las salidas permanece invariable.

Después de cada comando de control SYNC, los esclavos DP de los grupos seleccionados depositan los datos de salida de su búfer interno en las salidas al proceso.

Las salidas no se vuelven a actualizar cíclicamente hasta que el usuario no envía el comando de control UNSYNC con la instrucción "DPSYC_FR".

Nota

Si, en el momento en que se envía el comando de control, algún esclavo DP del grupo o grupos seleccionados no se encuentra conectado a la red o presenta un fallo, dichos esclavos tampoco pasan al modo SYNC. Este hecho no se notifica mediante el valor de retorno de la instrucción.

Leer entradas de módulos DP

Se leen los datos de entrada de los módulos DP

- Con comandos de carga a la periferia DP,
- Al actualizar la memoria imagen de proceso de las entradas (mediante el sistema operativo al final del OB 1 o mediante llamada de la instrucción "UPDAT_PI (Página 4738)"),
- Llamando la instrucción "DPRD_DAT (Página 4791)".

En el caso normal, el maestro DP recibe estos datos de entrada cíclicamente (en el ciclo del bus PROFIBUS DP) desde sus esclavos DP y los pone a disposición de la CPU.

Si se desea leer determinados datos de entrada que pueden estar repartidos entre varios esclavos desde el proceso de manera exactamente simultánea, debe enviarse el comando de control FREEZE al maestro DP correspondiente con la instrucción "DPSYC_FR".

¿Qué efecto produce FREEZE?

Con el comando de control FREEZE se ponen en modo de congelación los esclavos DP afectados, es decir, el maestro DP ordena a los esclavos DP afectados que congelen el estado actual de las entradas. A continuación transfiere los datos congelados al área de entrada de la CPU.

Después de cada comando de control FREEZE, los esclavos DP vuelven a congelar el estado de las entradas.

El maestro DP no vuelve a recibir cíclicamente el estado actual de las entradas hasta que el usuario no envía el comando de control UNFREEZE con la instrucción "DPSYC_FR".

Nota

Si, en el momento en que se envía el comando de control, algún esclavo DP del grupo o grupos seleccionados no se encuentra conectado a la red o presenta un fallo, dichos esclavos tampoco pasan al modo Freeze. Este hecho no se notifica mediante el valor de retorno de la instrucción.

Coherencia de datos

Debido al funcionamiento asíncrono y a la posibilidad de interrupción por clases de prioridad superiores, al utilizar la instrucción "DPSYC_FR" debe asegurarse que las memorias imagen de proceso sean coherentes con las entradas y salidas reales de la periferia.

Para garantizarlo, debe aplicarse una de las reglas de coherencia que se mencionan a continuación:

- Defina memorias imagen parcial de proceso adecuadas (solo posible con S7-400) para las salidas "SYNC" y las entradas "FREEZE". Llame la instrucción "UPDAT_PO (Página 4740)" inmediatamente antes de cada primera llamada de una petición SYNC. Llame la instrucción "UPDAT_PI (Página 4738)" inmediatamente después de cada última llamada de una petición FREEZE.
- Alternativamente: Para las salidas afectadas por una petición SYNC, y para las entradas afectadas por una petición FREEZE, utilice solo accesos directos a la periferia. Mientras haya una petición SYNC en curso, no es posible escribir en las salidas afectadas, ni tampoco leer las entradas afectadas mientras está en curso una petición FREEZE.

Uso de DPWR_DAT y DPRD_DAT

Si se utiliza la instrucción "DPWR_DAT (Página 4793)", esta debe estar finalizada antes de que se envíe una petición SYNC para las salidas en cuestión.

Si se utiliza la instrucción "DPRD_DAT (Página 4791)", esta debe estar finalizada antes de que se envíe una petición FREEZE para las entradas en cuestión.

Arranque y "DPSYC_FR"

El envío de los comandos de control SYNC y FREEZE en los OBs de arranque es responsabilidad exclusiva del usuario.

Si se desea que las salidas de uno o varios grupos empiecen a funcionar en el modo SYNC ya desde el inicio del programa de usuario, deben inicializarse esas salidas en el arranque y ejecutarse por completo la instrucción "DPSYC_FR" con el comando de control SYNC.

Si se desea que las entradas de uno o varios grupos empiecen a funcionar en el modo FREEZE ya desde el inicio del programa de usuario, debe ejecutarse por completo para esas entradas la instrucción "DPSYC_FR" con el comando de control FREEZE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPSYC_FR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Parámetro de control disparado por nivelREQ=1: Lanzar la petición SYNC/FREEZE
LADDR	Input	HW_DPMASTER	I, Q, M, D, L o constante	Identificador de hardware de la interfaz del maestro DP El número puede aplicarse en la vista de redes adoptándose de las propiedades de la interfaz del maestro DP o de la ficha "Constantes del sistema" de la tabla de variables estándar.
GROUP	Input	BYTE	I, Q, M, D, L o constante	Selección de grupo Bit 0 = 1: Grupo 1 seleccionado Bit 1 = 1: Grupo 2 seleccionado : Bit 7 = 1: Grupo 8 seleccionado Pueden seleccionarse varios grupos por petición. El valor B#16#0 no está permitido.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición (codificación según EN 50 170 Volume 2, PROFIBUS) Bit 0: Reservado (valor 0) Bit 1: Reservado (valor 0) Bit 2: • = 1: UNFREEZE se ejecuta • = 0: Sin significado Bit 3: • = 1: FREEZE se ejecuta • = 0: Sin significado Bit 4: • = 1: UNSYNC se ejecuta • = 0: Sin significado Bit 5: • = 1: SYNC se ejecuta • = 0: Sin significado Bit 6: Reservado (valor 0) Bit 7: Reservado (valor 0) Valores posibles: • Con una y solo una ID por petición: – B#16#04 (UNFREEZE) – B#16#08 (FREEZE) – B#16#10 (UNSYNC) – B#16#20 (SYNC) • Con varias ID por petición: – B#16#14 (UNSYNC, UNFREEZE) – B#16#18 (UNSYNC, FREEZE) – B#16#24 (SYNC, UNFREEZE) – B#16#28 (SYNC, FREEZE)
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Después de cada pasada del bloque debe evaluarse RET_VAL .
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La petición de SYNC/FREEZE todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Nota

Si accede a esclavos DPV1, el maestro DP puede transferir la información de error de dichos esclavos a la instrucción. Encontrará una descripción de esta información de error en STATUS[3], parámetro STATUS (Página 4755).

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
7000	Primera llamada con REQ=0. La petición definida mediante LADDR, GROUP y MODE no está activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1. La petición definida mediante LADDR, GROUP y MODE se ha lanzado; BUSY tiene el valor 1.
7002	Llamada intermedia (REQ irrelevante). La petición SYNC/FREEZE activada todavía está en curso; BUSY tiene el valor 1.
8090	El módulo seleccionado mediante LADDR no es un maestro DP.
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR (configuración o versión del maestro DP).
8094	Parámetros GROUP erróneos
8095	Parámetros MODE erróneos
80B0	El grupo seleccionado mediante GROUP no está configurado.
80B1	El grupo seleccionado mediante GROUP no está asignado a esta CPU.
80B2	La petición SYNC especificada mediante MODE no está permitida en el grupo seleccionado mediante GROUP.
80B3	La petición FREEZEespecificada mediante MODE no está autorizada en el grupo seleccionado mediante GROUP .
80C2	Falta de recursos temporal del maestro DP: En este momento el maestro DP está procesando el máximo posible de peticiones para una CPU.
80C3	Esta petición SYNC/UNSYNCno se puede activar en este momento, ya que en cada momento determinado solo puede estar lanzada una petición SYNC/UNSYNC. Compruebe el programa de usuario.
80C4	Esta petición FREEZE/UNFREEZEno se puede activar en este momento, ya que en cada momento determinado solo puede estar lanzada una petición FREEZE/UNFREEZE. Compruebe el programa de usuario.
80C5	Cortocircuito en la proximidad inmediata de la interfaz DP
80C6	Interrupción de la petición debido a expulsión de la periferia por la CPU
80C7	Interrupción de la petición debido a rearranque completo (en caliente) o arranque en frío del maestro DP
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

DPNRM_DG: Leer datos de diagnóstico de un esclavo DP

Descripción

La instrucción "DPNRM_DG" lee los datos de diagnóstico actuales de un esclavo DP de la forma establecida por EN 50 170 Volume 2, PROFIBUS.

La estructura fundamental del diagnóstico de esclavo se recoge en la siguiente tabla, y la información más detallada, en los manuales de los esclavos DP.

Byte	Significado
0	Estado de estación 1
1	Estado de estación 2
2	Estado de estación 3
3	Número de estación maestra
4	Identificación de fabricante (high byte)
5	Identificación de fabricante (low byte)
6 ...	Otros diagnósticos específicos de esclavo

Los datos leídos se introducen en el área de destino determinada por RECORD tras la correcta transferencia de los datos. El proceso de lectura se inicia ocupando el parámetro de entrada REQ con "1" durante la llamada de la instrucción "DPNRM_DG".

Funcionamiento

La lectura se realiza de forma asíncrona, es decir, que se puede prolongar a lo largo de varias llamadas. El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPNRM_DG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de lectura
LADDR	Input	HW_DPSLAVE	I, Q, M, D, L o constante	Dirección de diagnóstico configurada del esclavo DP Nota: La dirección se debe indicar en formato hexadecimal; p. ej., la dirección de diagnóstico 1022 significa: LADDR:=W#16#3FE.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	DINT, INT, LREAL, REAL	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Si no aparece ningún error, en RET_VAL se especifica la longitud de los datos realmente transferidos.
RECORD	Output	VARIANT	I, Q, M, D, L	Área de destino para los datos de diagnóstico leídos. Solo se admite el tipo de datos BYTE. La longitud mínima del registro que se debe leer o del área de destino es de 6. La longitud máxima del registro que se debe leer es de 240; en esclavos normalizados cuyo número de datos de diagnóstico normalizados sea superior a 240 bytes y tenga un máximo de 244 bytes, los primeros 240 bytes se transfieren al área de destino y el correspondiente bit de desbordamiento se activa en los datos.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La lectura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Encontrará información sobre la conversión de tipos de datos en los distintos lenguajes de programación en "Sinopsis de la conversión de tipos de datos (Página 2507)".

Parámetro RECORD

La CPU valora la longitud real de los datos de diagnóstico leídos:

Si la indicación de longitud de RECORD

- es inferior al número de datos proporcionados, los datos se rechazan y en RET_VAL se deposita la correspondiente información de error.
- es superior o igual al número de datos proporcionados, los datos se aceptan en el área de destino y en RET_VAL se registra la longitud real como valor positivo.

Nota

Hay que asegurarse de que los parámetros actuales de RECORD coincidan en todas las llamadas pertenecientes a una petición.

Una petición está claramente establecida por el parámetro de entrada LADDR.

Esclavos normalizados con datos de diagnóstico de más de 240 bytes

En esclavos normalizados con un número de datos de diagnóstico normalizados entre 241 y 244 bytes, debe tenerse en cuenta los siguiente:

Si la indicación de longitud de RECORD

- es inferior a 240 bytes, los datos se rechazan y en RET_VAL se deposita la correspondiente información de error.
- es superior o igual a 240 bytes, los primeros 240 bytes de los datos de diagnóstico normalizados se transfieren al área de destino y en los datos se activa el correspondiente bit de desbordamiento.

Parámetro RET_VAL

- Si se ha producido un error durante la ejecución de la función, el valor de respuesta contendrá un código de error,
- Si durante la transferencia no se ha producido ningún error, RET_VAL contiene la longitud de los datos leídos en bytes como número positivo.

Nota

En un esclavo DP, el número de datos leídos depende de su estado de diagnóstico.

Para evaluar la información de error del parámetro RET_VAL, véase la tabla siguiente.

Código de error (W#16#....)	Explicación	Restricción
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".	-
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".	Periferia descentralizada
8090	Dirección indicada no válida en el parámetro LADDR.	-
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR e IOID.	-
80A2	• Error de protocolo DP con Layer 2 (p. ej. fallo de esclavo o fallos de bus) • Con ET200S no es posible leer registros en el modo DPV0.	Periferia descentralizada
80A3	Error de protocolo de DP con User-Interface/User	Periferia descentralizada
80A4	Fallo de comunicación en PBUS+	Periferia descentralizada
80B0	• La instrucción no es posible para el tipo de módulo • El módulo no conoce este registro. • El número de registro 241 no es admisible. • Con "WR_REC (Página 4748)" no se permiten los registros 0 ni 1.	-
80B1	La indicación de longitud en el parámetro RECORD es errónea.	Indicación < longitud de registro
80B2	El slot configurado no está ocupado.	-
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado	-
80C0	No hay datos de diagnóstico.	-
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	-

Código de error (W#16#....)	Explicación	Restricción
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	-
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	-
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe si hay fuentes de perturbación en la instalación.	-
80C5	La periferia descentralizada no está disponible.	Periferia descentralizada
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano)	Periferia descentralizada
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)	-
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Sinopsis de la conversión de tipos de datos (Página 2507)

DP_TOPOL: Determinar topología para sistema maestro DP

Descripción

Esta instrucción permite lanzar la determinación de topología para un sistema maestro seleccionado. Al llamar esta instrucción, se accede a todos los repetidores de diagnóstico conectados a un sistema maestro DP.

Nota

La determinación de topología puede efectuarse en cada momento concreto solo para un sistema maestro DP.

La determinación de topología es imprescindible para poder visualizar con detalle la ubicación de los eventuales errores de cable. Después del montaje del sistema maestro DP y de cada modificación de su estructura física, debe efectuarse de nuevo la determinación de topología con la instrucción "DP_TOPOL".

Se considera que se ha modificado la estructura física cuando:

- Se modifican longitudes de cables
- Se añaden o eliminan estaciones o componentes con función de repetidor
- Se modifican direcciones de estaciones

Si un repetidor de diagnóstico notifica un error, "DP_TOPOL" escribe en las salidas DPR y DPRI a lo largo de la duración de un paso de "DP_TOPOL". Si varios repetidores de diagnóstico del sistema maestro DP seleccionado notifican errores, "DP_TOPOL" envía a DPR

y DPRI información acerca del primer repetidor de diagnóstico que ha notificado errores. La información de diagnóstico completa se puede leer con la instrucción "DPNRM_DG (Página 4806)" o en la PG. Si ningún repetidor de diagnóstico notifica errores, las salidas DPR y DPRI adquieren el valor NULL.

Si, después de producirse un error, se desea volver a lanzar una determinación de topología, en primer lugar debe restablecerse "DP_TOPOL". Para hacerlo, debe llamarse "DP_TOPOL" con REQ=0 y R=1.

Funcionamiento

"DP_TOPOL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la determinación de topología de bus, debe llamarse "DP_TOPOL" con REQ=1 . Para interrumpir la operación, debe llamarse "DP_TOPOL" con R=1 .

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Nota

La determinación de topología puede durar varios minutos.

Identificación de una petición

El parámetro de entrada DP_ID define una petición de manera unívoca.

Si se ha llamado "DP_TOPOL" y se vuelve a llamar la misma instrucción antes de que finalice la determinación de topología, el comportamiento posterior de la instrucción depende de si la nueva llamada corresponde a la misma petición: Si el parámetro DP_ID coincide con una petición todavía no finalizada, la llamada se interpreta como llamada consecutiva, y en RET_VALse registra el valor W#16#7002. En cambio, si se trata de una petición diferente, la CPU la rechaza.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DP_TOPOL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ=1: Lanzar determinación de topología
R	Input	BOOL	I, Q, M, D, L o constante	R=1: Interrumpir la determinación de topología
DP_ID	Input	HW_IOSYSTEM	I, Q, M, D, L o constante	ID del sistema maestro DP cuya topología se debe determinar
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La determinación de topología todavía no ha finalizado.
DPR	Output	BYTE	I, Q, M, D, L	Dirección PROFIBUS del repetidor de diagnóstico que notifica errores
DPRI	Output	BYTE	I, Q, M, D, L	Segmento de medición del repetidor de diagnóstico que notifica errores: • Bit 0 = 1: Fallos temporales en el segmento DP2 • Bit 1 = 1: Fallos permanentes en el segmento DP2 • Bit 4 = 1: Fallos temporales en el segmento DP3 • Bit 5 = 1: Fallos permanentes en el segmento DP3

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Dentro de la información de error "verdadera" (códigos de error W#16#8xyz) de la tabla siguiente, debe distinguirse entre dos casos:

- Errores temporales (códigos de error de W#16#80A2 a 80A4, 80C3, 80C5):
  En este tipo de errores es posible que el error se elimine sin intervención del usuario, es decir, puede bastar con llamar de nuevo "DP_TOPOL" (varias veces si es necesario). Ejemplo de error temporal: En este momento los recursos necesarios están ocupados (W#16#80C3).
- Errores permanentes (códigos de error de W#16#8082, 80B0, 80B2):
  Los errores de este tipo no se pueden eliminar sin intervención del usuario. Solo tiene sentido volver a llamar "DP_TOPOL" una vez que se haya eliminado el error. Ejemplo de error permanente: El maestro DP o la CPU no soportan este servicio. (W#16#80B0).

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
7000	Primera llamada con REQ=0. No se ha lanzado ninguna determinación de topología. BUSY tiene el valor "0".
7001	Primera llamada con REQ=1. Se ha lanzado la petición de determinación de topología. BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): La determinación de topología todavía está procesándose. BUSY tiene el valor "1".
7010	Se ha intentado interrumpir la determinación de topología. Sin embargo, no hay ninguna petición en curso con la DP_IDindicada. BUSY tiene el valor "0".
7011	Primera llamada con R=1. Se ha lanzado la interrupción de la determinación de topología; BUSY tiene el valor "1".
7012	Llamada intermedia: La interrupción de la determinación de topología todavía está procesándose. BUSY tiene el valor "1".

Código de error* (W#16#...)	Explicación
7013	Última llamada: Se ha interrumpido la determinación de topología. BUSY tiene el valor "0".
8082	No está configurado ningún maestro DP con la DP_ID indicada.
80A2	Error en determinación de topología; encontrará información más detallada en los parámetros de salida DPR y DPRI.
80A3	Error en determinación de topología: El tiempo de vigilancia ha finalizado (timeout).
80A4	Fallo de comunicación en PBUS+
80B0	El maestro DP o la CPU no soportan este servicio.
80B2	Error en determinación de topología: No se ha detectado ningún repetidor de diagnóstico en el sistema maestro DP seleccionado.
80C3	En este momento los recursos necesarios están ocupados. Causa posible: Ha iniciado una segunda determinación de la topología (no está permitido determinar más de una topología en un determinado instante).
80C5	El sistema maestro DP no está disponible en este momento.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

ASi

ASI_CTRL: controlar el comportamiento del maestro ASi

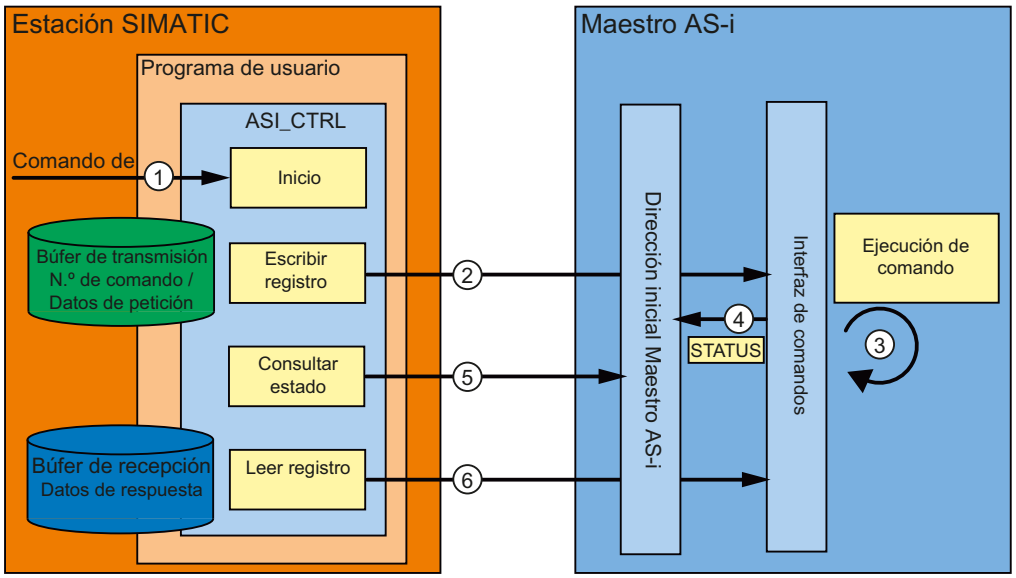
Descripción ASI_CTRL

Descripción

La instrucción "ASI_CTRL" permite controlar el comportamiento del maestro AS-i a través del programa de usuario del PLC. La instrucción procesa automáticamente el protocolo del comando y permite la parametrización de maestros AS-i SIMATIC así como la lectura de datos de información. Las posibilidades disponibles y el funcionamiento de la interfaz de comandos se describen en el manual del maestro AS-i.

Se soportan maestros AS-i con conexión centralizada y maestros AS-i descentralizados vía PROFIBUS DP. E igualmente son posibles combinaciones con PROFINET IO (ejemplo: IE/PB Link PN IO).

La figura siguiente muestra esquemáticamente la funcionalidad de la instrucción "ASI_CTRL":



① Inicio del procesamiento en el parámetro REQ.

② El programa transmite al maestro AS-i el comando deseado mediante la instrucción "RDREC".

③ El maestro AS-i ejecuta el comando.

④ El estado actual del maestro AS-i se deposita en el área de entrada de datos binarios (dirección base lógica).

⑤ La instrucción "ASI_CTRL" consulta cíclicamente los 4 bits de estado y los evalúa.

⑥ Si ha terminado la ejecución de comandos, la petición de comandos concluye mediante "RDREC". En el campo de datos de "RDREC" se encuentran, para cada comando, los datos de respuesta del mismo u otra información de estado.

Diferencias en la llamada de comandos con IE/AS-i Link y DP/AS-i Links

Existen diferencias sustanciales en el procedimiento de intercambio de comandos entre un controlador y un maestro AS-i.

- Con **IE/AS-i Link** (PROFINET) se utiliza la interfaz de registros. Los distintos comandos son llamados por diferentes números de registro mediante "Escribir registro" (instrucción "WRREC") o "Leer registro" (instrucción "RDREC").
- Los **DP/AS-i Links** (PROFIBUS) utilizan la interfaz de comandos. Todos los comandos son llamados por el número de registro 2 mediante una pareja de instrucciones "Escribir registro" (instrucción "WRREC") y "Leer registro" (instrucción "RDREC"). El tipo de comando lo define el contenido de datos al efectuarse la petición de escritura.

Cambios respecto de la instrucción "ASi_3422"

La instrucción "ASI_CTRL" es una versión revisada de la instrucción "ASi_3422 (Página 6809)" (S7-300/400) y presenta mejoras de funcionalidad y compatibilidad. En concreto, se han efectuado las modificaciones siguientes:

- Para escribir y leer registros de diagnóstico, las instrucciones "WR_REC (Página 4790)" y "RD_REC (Página 4785)" han sido sustituidas por las instrucciones "RDREC (Página 4746)" y "WRREC (Página 4748)". Sus funciones son idénticas, pero soportan la transferencia de datos vía PROFINET IO.
- El tipo de bloque de la instrucción ha cambiado de función (FC) a bloque de función (FB). "ASI_CTRL" dispone de un bloque de datos instancia y es apto para multiinstancia.
- La denominación de los parámetros formales de "ASI_CTRL" es conforme con los bloques de sistema SIMATIC. Desaparece el parámetro de entrada STARTUP. La definición del parámetro STATUS se ha basado en las instrucciones "RDREC (Página 4746)" y "WRREC (Página 4748)". Además, se han adaptado las identificaciones de estado para el parámetro DONE y para el parámetro BUSY nuevo.

Funcionamiento de la instrucción "ASI_CTRL"

La instrucción "ASI_CTRL" es un bloque de función asíncrono, de modo que la ejecución se prolonga a lo largo de varias llamadas.

- Una petición se inicia con REQ = TRUE.
- Mediante los parámetros de salida BUSY y los dos bytes centrales del parámetro de salida STATUS se muestra el estado de la petición.
- Mientras se procesa la petición, está activado el parámetro BUSY. En la primera llamada, STATUS contiene el valor 00700100_H. En todas las llamadas sucesivas pertenecientes a esta petición, contiene el valor 00700200_H. Al concluir la petición, el resultado se indica en los parámetros DONE o ERROR.
  - Si no se ha producido ningún error, se activa DONE. En peticiones con datos de respuesta del maestro AS-i, dichos datos se ponen a disposición en el búfer de recepción indicado. En este caso, además, en el parámetro STATUS se muestra la cantidad de datos proporcionados en bytes. En peticiones sin datos de respuesta, en STATUS se registra el valor 00000000_H.
  - Si durante el procesamiento de la petición se presenta un error, se activa ERROR. En ese caso, el contenido del búfer de recepción no es válido. Para definir con más precisión el error ocurrido, en el parámetro STATUS se registra un código de error.

Número de llamadas de comandos

Si se utiliza la instrucción "ASI_CTRL" para efectuar la petición de comandos, no deben emitirse simultáneamente otros comandos al mismo maestro AS-i mediante "RDREC (Página 4746)" y "WRREC (Página 4748)". Lo mismo se aplica a las llamadas múltiples de la instrucción en el mismo maestro AS-i.

La instrucción "ASI_CTRL" no se puede ejecutar con posibilidad de interrupción. Por ello, las llamadas no se pueden programar en niveles de ejecución que se interrumpan recíprocamente (p. ej. con llamada en el OB 1 y en el OB 35).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ASI_CTRL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Con REQ = TRUE se inicia una petición nueva siempre que no se esté ejecutando otra. No se realiza ninguna evaluación de flancos.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección inicial del maestro AS-i en el área de direcciones S7 (dirección base lógica). La dirección inicial se determina en la configuración de hardware al configurar el maestro.
SD	Input	VARIANT	I, Q, M, D, L	Búfer de transmisión El parámetro remite a un área de memoria en la que debe especificarse el comando (véase "Comandos AS-i (Página 4817)"). Ejemplo: P#DB101.DBX 0.0 BYTE 223
RD	Input	VARIANT	I, Q, M, D, L	Búfer de recepción Este búfer solo es relevante para comandos que devuelven datos de respuesta. El parámetro remite a un área de memoria en la que se guarda una respuesta al comando (véase "Comandos AS-i (Página 4817)"). Ejemplo: P#DB102.DBX 224.0 BYTE 221
DONE	Output	BOOL	Q, M, D, L	DONE = TRUE: petición finalizada sin errores.
BUSY	Output	BOOL	Q, M, D, L	BUSY = TRUE: petición en proceso.
ERROR	Output	BOOL	Q, M, D, L	ERROR = TRUE: petición cancelada con errores.
STATUS	Output	DWORD	M, D	Estado de la petición / Código de error Véase la descripción de "Parámetro STATUS".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Parámetros LADDR, SD y RD

Los parámetros LADDR, SD y RD no pueden modificarse en ninguna ejecución del bloque mientras se procesa una petición; deben permanecer constantes.

Parámetro STATUS

La tabla siguiente contiene las posibles indicaciones de STATUS en función de DONE y ERROR.

DONE	ERROR	STATUS	Significado
0	0	00700000_H	Primera llamada con REQ = FALSE; no hay ninguna petición activa.
0	0	00700100_H	Primera llamada con REQ = TRUE; petición lanzada.
0	0	00700200_H	Llamada sucesiva (REQ es irrelevante); se sigue procesando la petición.
1	0	00000000_H	Petición finalizada sin errores. Sin datos de respuesta.
1	0	$0000xx00_H$	Petición finalizada sin errores. Número de bytes xx de los datos de respuesta.
0	1	$C0818400_H$	Tipo de datos del operando formal RD no admisible.
0	1	$C0818500_H$	Error de comunicación con el maestro AS-i (dirección configurada erróneamente en el parámetro LADDR).
0	1	$C0838100_H$	Dirección errónea del esclavo AS-i.
0	1	$C0838200_H$	Esclavo AS-i no activado (no en LAS).
0	1	$C0838300_H$	Error en AS-Interface (es posible que se haya ajustado un valor insuficiente en el parámetro SD).
0	1	$C0838400_H$	En el estado actual del maestro AS-i este comando no está permitido.
0	1	$C0838500_H$	Existe un esclavo AS-i con dirección "0".
0	1	$C0838600_H$	El esclavo AS-i tiene datos de configuración no permitidos (E/S o códigos ID).
0	1	$C083A100_H$	El esclavo AS-i solicitado no se ha encontrado en el AS-Interface.
0	1	$C083A200_H$	Existe un esclavo AS-i con dirección "0".
0	1	$C083A300_H$	En el AS-Interface ya existe un esclavo AS-i con la dirección nueva.
0	1	$C083A400_H$	No es posible borrar la dirección del esclavo AS-i.
0	1	$C083A500_H$	No es posible establecer la dirección del esclavo AS-i.
0	1	$C083A600_H$	No es posible guardar permanentemente la dirección del esclavo AS-i.
0	1	$C083A700_H$	Error al leer el código Extended ID1.
0	1	$C083A800_H$	La dirección de destino no es plausible (p. ej., se ha utilizado una dirección de esclavo B para un esclavo estándar).
0	1	$C083B100_H$	Se ha producido un error de longitud en la transferencia de string.
0	1	$C083B200_H$	Se ha producido un error de protocolo en la transferencia de string.
0	1	$C083F800_H$	Se desconoce el número de petición o el parámetro de la petición.
0	1	$C083F900_H$	El maestro AS-i ha detectado un error EEPROM.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Comandos AS-i

Descripción

La interfaz de comandos permite intercambiar datos de parametrización e información entre el controlador y el maestro AS-i.

Con estos comandos:

- se proporciona a la especificación del maestro AS-i la funcionalidad completa del perfil de maestro M4;
- puede configurarse completamente el maestro AS-i desde el controlador.

Nota

Comandos AS-i soportados

En el manual del maestro AS-i correspondiente pueden consultarse los comandos AS-i soportados y una descripción detallada de los mismos.

Estructura general del búfer de transmisión

La tabla siguiente muestra la estructura general del búfer de transmisión para comandos y datos de peticiones. El área para el número de comando debe estar ocupada siempre. El número de bytes para los datos de la petición se encuentran en la descripción de cada comando (véase la documentación del maestro AS-i). "q" corresponde a la dirección inicial del búfer de transmisión.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Significado							
q + 0	Número del comando							
q + 1	Datos de la petición							
q + 2	Datos de la petición							
q + ...	Datos de la petición							

Estructura general del búfer de recepción

La tabla siguiente muestra la estructura general del búfer de recepción para los datos de respuesta del comando. El número de bytes para los datos de respuesta depende de cada comando. Algunos comandos no proporcionan datos de respuesta y, por tanto, necesitan la especificación de un búfer de recepción virtual que no esté rellenado con datos. "n" corresponde a la dirección inicial del búfer de recepción.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Significado							
n + 0	Número del comando (eco)							
n + 1	Datos de respuesta							

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Significado							
n + 2	Datos de respuesta							
n + ...	Datos de respuesta							

PRECAUCIÓN
Las áreas de memoria pueden sobrescribirse
Si el búfer de recepción de la instrucción "ASI_CTRL" es demasiado corto, pueden sobrescribirse las áreas de memoria colindantes. La longitud indicada en el puntero ANY del parámetro RD durante la llamada de la instrucción "ASI_CTRL" es irrelevante. La longitud necesaria del búfer de recepción debe consultarse en la descripción del comando.
Para los números de comando 39_H, 41_H, 42_H, 43_H y 44_H se aplica lo siguiente:
el búfer de recepción debe tener 221 bytes de longitud (del byte 0 al byte 220) aun cuando el comando devuelva menos datos. Dependiendo del comando, el maestro AS-i puede sobrescribir con valores cero los bytes más altos contenidos en el búfer de recepción.

Comandos AS-i

La tabla siguiente muestra una selección posible de comandos AS-i.

Nombre	Parámetro	Retorno	Codificación
Configurar_valor_de_parámetro (Set_Permanent_Parameter)	Dirección de esclavo, parámetros		00_H
Leer_valor_de_parámetro_configurado (Get_Permanent_Parameter)	Dirección de esclavo	Parámetros	01_H
Escribir_valor_de_parámetro (Write_Parameter)	Dirección de esclavo, parámetros	Eco de parámetro	02_H
Leer_valor_de_parámetro (Read_Parameter)	Dirección de esclavo	Valor del parámetro	03_H
Configurar_valores_de_parámetros_reales (Store_Actual_Parameters)			04_H
Configurar_datos_de_configuración	Dirección de esclavo, configuración		25_H
Leer_datos_de_configuración_configurados	Dirección de esclavo	Datos de configuración configurados	26_H
Configurar_datos_de_configuración_reales (Store_Actual_Configuration)			07_H
Leer_datos_de_configuración_reales	Dirección de esclavo	Datos de configuración reales	28_H
Configurar_LPS	LPS		29_H
Activar_modo_Offline	Modo		$0A_H$
Seleccionar_autoprogramación	Modo		$0B_H$
Activar_modo	Modo		$0C_H$

Nombre	Parámetro	Retorno	Codificación
Cambiar_dirección_esclavo_AS–i (Change_AS–iSlave_Address)	Dirección1, dirección2		$0D_H$
Leer_estado_esclavo_AS–i	Dirección de esclavo	Registro de errores del esclavo AS-i	$0F_H$
Leer_listas_y_flags		LDS, LAS, LPS, flags	30_H
Leer_configuración_total		Datos de configuración reales, parámetros actuales, LAS, flags	39_H
Configurar_configuración_total	Configuración total		$3A_H$
Escribir_lista_parámetros	Lista de parámetros		$3C_H$
Leer_lista_eco_parámetros		Lista de ecos de parámetros	33_H
Escribir_petición_CTT2	Dirección de esclavo string CTT2	String CTT2	44_H
Leer_identificador_versión		String de versión	14_H
Leer_ID_esclavo_AS–i	Dirección de esclavo	Código ID	17_H
Leer_Extended–ID1_esclavo_AS-i	Dirección de esclavo	Código Extended ID1	37_H
Escribir_Extended-ID1_esclavo_AS–i	Código Extended ID1		$3F_H$
Leer_Extended–ID2_esclavo_AS-i	Dirección de esclavo	Código Extended ID2	38_H
Leer_ES_esclavo_AS–i	Dirección de esclavo	Configuración E/S	18_H
Leer_lista_errores_periferia		LPF	$3E_H$
Escribir_string_parámetros_esclavo_AS–i	Dirección de esclavo, string de parámetros		40_H
Leer_string_parámetros_esclavo_AS–i	Dirección de esclavo	String de parámetros	41_H
Leer_string_ID_esclavo_AS–i	Dirección de esclavo	String ID	42_H
Leer_string_diagnóstico_esclavo_AS–i	Dirección de esclavo	String de diagnóstico	43_H
Leer_contador_errores_línea_AS–i			$4A_H$
Leer_y_borrar_contador_errores_línea_AS–i			$4B_H$
Leer_contador_errores_esclavo_AS–i	Dirección de esclavo		$4C_H$
Leer_y_borrar_contador_errores_esclavo_AS–i	Dirección de esclavo		$4D_H$
Comando adicional para el DP/ AS-i F-Link:			
Estado_AS-i/Diag_esclavos_F		Estado / Diagnóstico de todos los esclavos ASIsafe	51_H

Nota

Nueva inicialización de la interfaz de comandos del maestro AS-i

Otro comando no mencionado en la tabla es el comando 77_H. Con esta llamada se reinicializa la interfaz de comandos del maestro AS-i. Si justo en este momento el maestro AS-i indicado está ocupado con un comando, este se interrumpe.

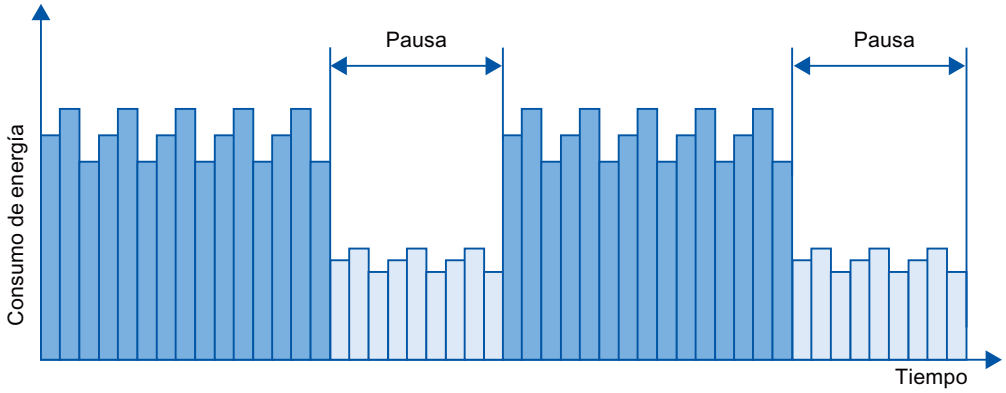
A partir de la versión V2.1.20 de DP/AS-i LINK Advanced se ofrece el comando adicional $0E_H$. Con esta llamada puede habilitarse o bloquearse la función de vigilancia de contacto a tierra de una línea.

PROFIenergy

Descripción de PROFIenergy

PROFIenergy

PROFIenergy es un perfil de gestión energética con PROFINET independiente de cualquier fabricante y cualquier dispositivo concreto. PROFIenergy permite desconectar los dispositivos de forma coordinada y centralizada para reducir el consumo eléctrico durante pausas de producción o durante interrupciones no planificadas.



Los dispositivos PROFINET o los módulos de potencia se desconectan mediante comandos especiales en el programa de usuario del controlador PROFINET IO. No se requiere hardware adicional. Los dispositivos PROFINET interpretan directamente los comandos PROFIenergy.

Controlador PROFIenergy (controlador PE)

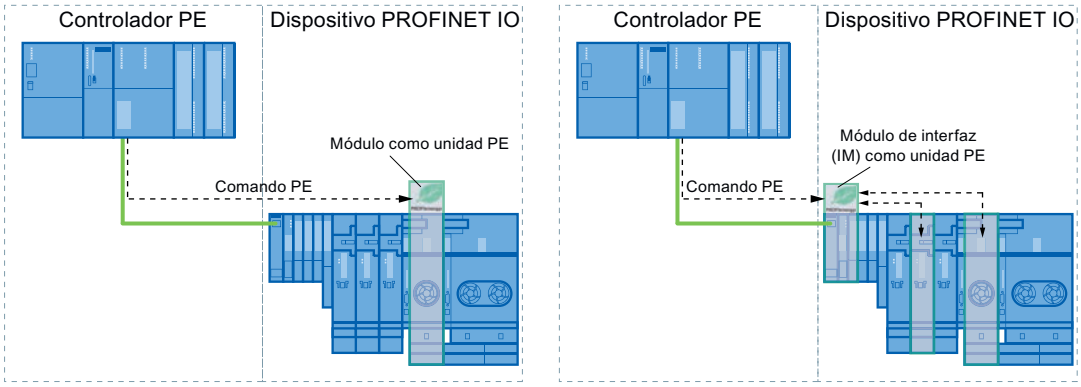
El controlador PE es un PLC que activa o desactiva el estado de reposo de los dispositivos subordinados. La desconexión o la reactivación de componentes de producción concretos o de líneas de producción enteras se realiza mediante el programa de usuario. Los comandos (p. ej. "Start_Pause" o "End_Pause") se emiten al dispositivo subordinado mediante las instrucciones correspondientes (bloques de función). Los comandos se envían con el protocolo de comunicación PROFINET.

Unidad PROFIenergy (unidad PE)

La unidad PE recibe los comandos PROFIenergy del controlador PE y los aplica de forma correspondiente (p. ej. devolviendo un valor medido o activando un modo de ahorro de energía). La aplicación de la unidad PE en un dispositivo con capacidad PROFIenergy depende del tipo de dispositivo y del fabricante.

La unidad PE se puede aplicar, por ejemplo:

- En el Proxy de un módulo: los comandos PE son válidos para el módulo direccionado, respectivamente, y dado el caso, para módulos subordinados.
- En el Proxy de una tarjeta: los comandos PE son válidos para distintos módulos dentro de la tarjeta, respectivamente.



- En un módulo sin funcionalidad Proxy dentro de la red: en este caso, los comandos PE solo son válidos para el módulo correspondiente.

Instrucciones PROFIenergy

- Instrucciones para controlador IO
  - La manera más sencilla de activar o desactivar el estado de reposo de los dispositivos PROFINET es el uso de la instrucción "PE_START_END (Página 4822)" (comandos PROFIenergy "Start_Pause" y "End_Pause"). mediante un flanco de señal ascendente y descendente en la misma.
  - La instrucción "PE_CMD (Página 4826)" transfiere todos los comandos PROFIenergy, incluidos "Start_Pause" y "End_Pause". Además, con otros comandos puede consultarse el estado actual de los dispositivos PROFINET o su comportamiento durante las pausas, por ejemplo.
  - Con la instrucción "PE_DS3_Write_ET200S (Página 4831)" se especifican los ajustes para el comportamiento de conexión de un máximo de 8 slots del ET 200S. Esta instrucción no es una instrucción PROFIenergy pero complementa las funciones PROFIenergy de un ET 200S.
- Instrucción para iDevices
  La instrucción "PE_I_DEV (Página 4856)" permite ejecutar PROFIenergy también en iDevices. Esta instrucción recibe comandos PROFIenergy en el iDevice y los transfiere al programa de usuario para ejecutarlos. Después de ejecutar el comando, el programa de usuario llama otra vez a la instrucción "PE_I_DEV (Página 4856)" para enviar acuse al controlador IO. Para responder, se dispone para cada comando de un bloque auxiliar específico que proporciona a la instrucción "PE_I_DEV (Página 4856)" los datos de respuesta.

Comandos PROFIenergy (comandos PE)

El controlador PE envía los comandos PE a la unidad PE. El comando PE puede ser un comando de control para poner una unidad PE en un determinado modo de ahorro de energía, o un comando para leer un estado o un valor medido:

- Comandos PE para el control
  PROFIenergy soporta dos comandos de control que pueden ejecutarse mediante la instrucción "PE_Start_End (Página 4822)" o mediante la instrucción "PE_CMD (Página 4826)":
  - Start_Pause: Iniciar un modo de ahorro de energía adecuado (PE Energy-saving mode)
  - End_Pause: Finalizar el modo de ahorro de energía (paso al modo PE_ready_to_operate)
- Comandos PE para leer un estado o un valor medido
  Mediante los siguientes comandos de estado, el control puede leer con la instrucción "PE_CMD (Página 4826)" determinada información de estado:
  - PE_Identify: Leer qué comandos PE soporta la unidad PE.
  - PEM_Status: Leer cuál es el modo activo de una unidad PE en el momento actual (p. ej. PE_ready_to_operate).
  - Query_Modes: Muestra una sinopsis de todos los modos de ahorro de energía soportados, incluida la información de tiempo y energía
  - Query_Measurement: Muestra los valores medidos de una unidad PE

Ejemplos de aplicación

Pueden consultarse ejemplos de aplicación de las instrucciones PROFIenergy en el artículo "PROFIenergy - Ahorrar energía con SIMATIC S7 (http://support.automation.siemens.com/WW/view/es/41986454)" del Industry Online Support.

Consulte también

Service & Support (http://support.automation.siemens.com/)

Controlador IO

PE_START_END: Iniciar y finalizar modo de ahorro de energía

Descripción

Con la instrucción "PE_START_END" se inicia y finaliza la pausa de ahorro de energía para la entidad PE especificada (p. ej. el ET 200S).

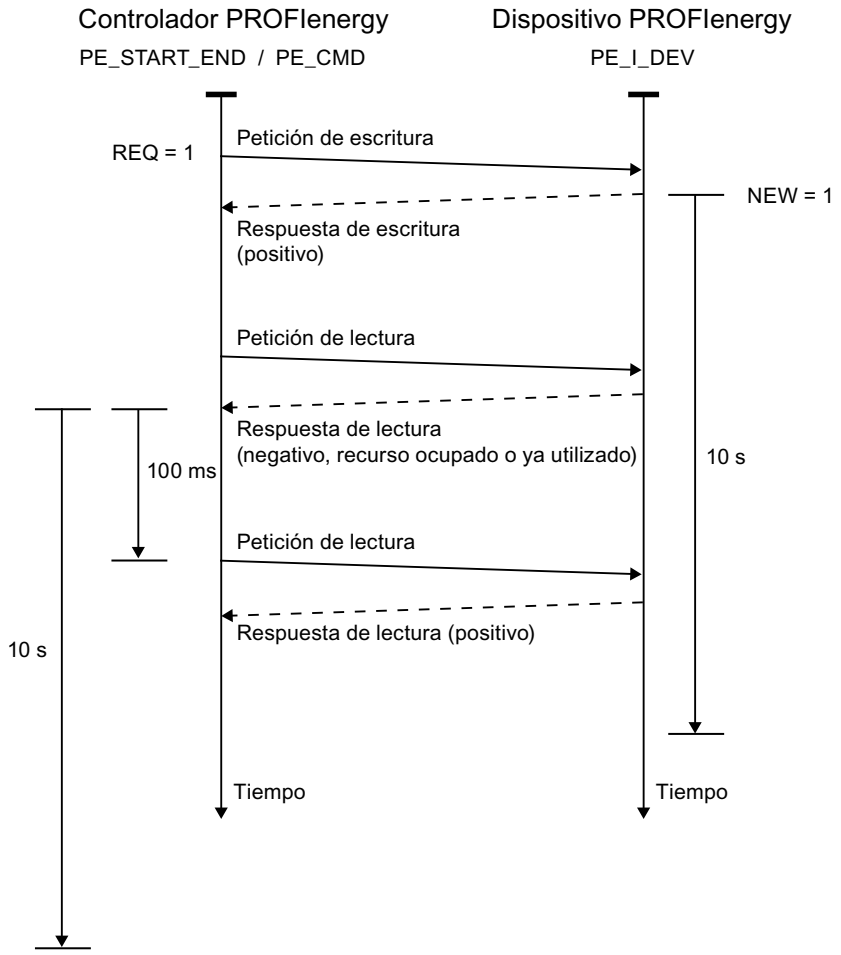
La instrucción "PE_START_END" se emplea en el controlador PE preferentemente cuando en los dispositivos PE asignados solo hay conectados aparatos de campo en los que no van a leerse datos de energía. Para leer los datos de energía también puede emplearse la instrucción "PE_CMD (Página 4826)".

Los modos de ahorro de energía se configuran en el programa de usuario del controlador PE. Después de ejecutar "PE_START_END", la unidad PE responde notificando qué modo de ahorro de energía se ha adoptado realmente y lo indica en el parámetro PE_MODE_ID.

Peticiones de escritura y lectura de la instrucción "PE_START_END"

La instrucción "PE_START_END" envía internamente mediante "WRREC (Página 4748)" un comando PROFIenergy en forma de petición de escritura a la unidad PE. A continuación, "PE_START_END" espera el acuse de la unidad PE. Para ello, cada 100 milisegundos se lee el registro de acuse con la instrucción "RDREC (Página 4746)". Mientras no se reciba acuse de la unidad PE, la petición de lectura se repite cada 100 milisegundos durante 10 segundos. Los datos de respuesta de la unidad PE también se leen con la instrucción "RDREC (Página 4746)".

La figura siguiente muestra el diagrama de ejecución de las peticiones de escritura y lectura:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_START_END":

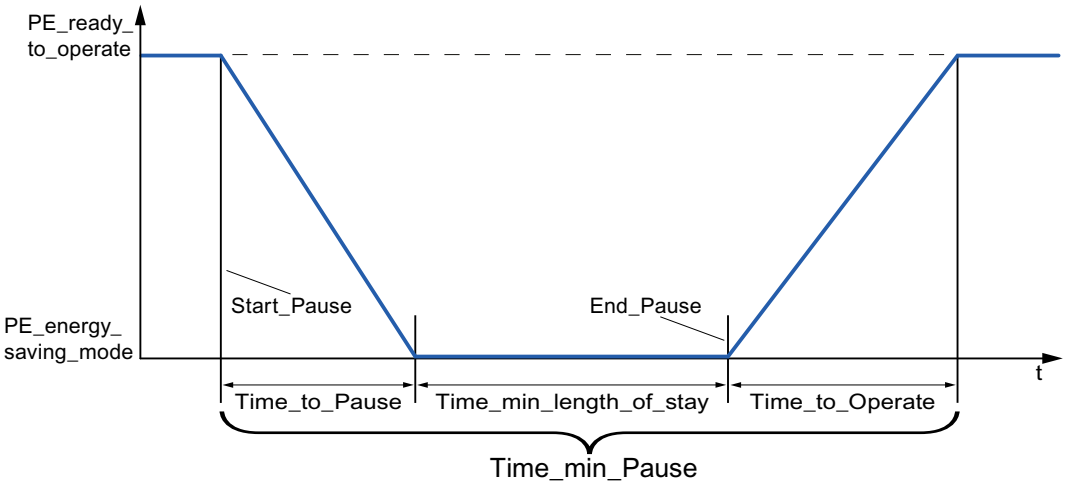
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
START	Input	BOOL	I, Q, M, D, L o constante	Transmisión del comando PE "Start_Pause" a la unidad PE con la dirección en el parámetro ID.
END	Input	BOOL	I, Q, M, D, L o constante	Transmisión del comando PE "End_Pause" a la unidad PE con la dirección en el parámetro ID.
ID	Input	HW_ SUBMODULE	I, Q, M, D, L o constante	Dirección de la entidad PE Para un dispositivo PROFINET IO utilice el ID de hardware del módulo de cabecera. El identificador de hardware figura en las constantes de sistema del controlador IO asignado. El nombre del módulo de cabecera se compone del nombre del dispositivo IO y el complemento [Head] (ejemplo: "IO_Device_1[Head]").
PAUSE_TIME	Input	TIME	I, Q, M, D, L, P o constante	Duración planificada de la pausa. • Rango: T#1MS a T#24D20H31M23S647MS • Valor de arranque: T#0MS
VALID	Output	BOOL	I, Q, M, D, L	Comando PE emitido correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Aún se está ejecutando el comando PE.
ERROR	Output	BOOL	I, Q, M, D, L	Se ha producido un error durante la ejecución. El mensaje de error se indica en el parámetro STATUS.
STATUS	Output	DWORD	I, Q, M, D, L, P	Estado de bloque o número de error (véase "Parámetro STATUS")
PE_MODE_ID	Output	BYTE	I, Q, M, D, L, P	Número de identificación del modo de ahorro de energía (nivel de ahorro de energía adoptado durante la pausa).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro PAUSE_TIME

Con el parámetro PAUSE_TIME se predefine la duración de la pausa de ahorro de energía para la unidad PE. En la unidad PE se comprueba si la pausa predefinida tiene duración suficiente y si se puede aplicar. La duración mínima de pausa (Time_min_Pause) debe ser

mayor que la suma de los tiempos que el dispositivo necesita para cambiar al modo de ahorro de energía (Time_to_Pause) y para volver al modo de operación (Time_to_Operate).



En el ET 200S se comprueba si la pausa planificada es mayor o igual que la duración mínima de pausa (PM-E_Pause_Min) depositada en el ET 200S. Esta es de 10 segundos. Si se utiliza una pausa menor, los módulos de potencia (PM-E) del ET 200S permanecen conectados.

Una vez transcurrido el tiempo de pausa no se produce ningún rearranque automático, sino que el módulo permanece en estado OFF hasta que se emite el comando "END". De este modo se evita una reconexión descoordinada, que podría provocar picos de carga indeseados.

Parámetro STATUS

En el parámetro de salida STATUS se muestra información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	Causa del error • B#16#**00**: sin error • B#16#**DE**: error al leer el registro • B#16#**DF**: error al escribir el registro • B#16#**C0**: mensaje de error de la instrucción o de las instrucciones de comunicación "RDREC (Página 4746)" y "WRREC (Página 4748)" empleadas internamente.
STATUS[2]	Error_Decode	Ubicación de la identificación de error • **80**: error DPV1 según la norma IEC 61158-6 o error específico de instrucción • **FE**: perfil DP/PNIO; error específico de PROFIenergy

Elemento de campo	Nombre	Significado
STATUS[3]	Error_Code_1	Identificación de error • Con Error_Decode = **80**: – **80**: flancos ascendentes simultáneos en los parámetros de entrada START y END. – **81**: conflicto de longitud en los parámetros CMD_PARAM y CMD_PARAM_LEN. – **82-8F**: otros mensajes de error (reservado) • Con Error_Decode = **FE**: – **01**: "Service Request ID" no válido – **02**: "Request_Reference" erróneo – **03**: "Modifier" no válido – **04**: "Data Structure Identifier RQ" no válido – **05**: "Data Structure Identifier RS" no válido – **06**: "PE energy-saving modes" no soportados – **07**: "Response" es demasiado largo (se ha rebasado la longitud máxima transferible) – **08**: "Count" no válido – **50**: no hay ningún "energy mode" adecuado disponible. – **51**: el valor de tiempo indicado no se soporta. – **52**: "PE_Mode_ID" no admisible
STATUS[4]	Error_Code_2	Extensión específica de fabricante de la identificación de error

Nota

Mensajes de error de las instrucciones RDREC y WRREC

La instrucción "PE_START_END" utiliza para la comunicación las instrucciones "WRREC (Página 4748)" y "RDREC (Página 4746)". Los mensajes de error de estas instrucciones se emiten correspondientemente en los elementos de campo STATUS[1] a STATUS[4].

Consulte el significado de los códigos de error de las instrucciones "WRREC (Página 4748)" y "RDREC (Página 4746)" en la descripción del parámetro STATUS (Página 4755) correspondiente.

Consulte también

Descripción de PROFIenergy (Página 4820)

PE_CMD: Iniciar y finalizar modo de ahorro de energía / Leer información de estado

Descripción

La instrucción "PE_CMD" se emplea en el controlador PE e inicia o finaliza una pausa de ahorro de energía en la unidad PE. Además, "PE_CMD" permite leer más información y valores medidos de energía de una unidad PE.

La instrucción puede emplearse preferentemente en controladores PE en cuyos dispositivos PE correspondientes estén conectados aparatos de campo de los que deban leerse valores medidos de energía. Si esto no fuera así, para iniciar y finalizar las pausas también puede emplearse la instrucción "PE_START_END (Página 4822)".

Transferencia de los comandos PROFIenergy (comandos PE)

La instrucción "PE_CMD" transfiere un comando PROFIenergy a una unidad PE.

La instrucción también puede emplearse si en el futuro se amplía el perfil PROFIenergy con otros comandos. Los comandos que pueden utilizarse de acuerdo con el perfil PROFIenergy actual se listan en la descripción de los parámetros CMD y CMD_MODIFIER (véase tabla "Parámetros CMD y CMD_MODIFIER").

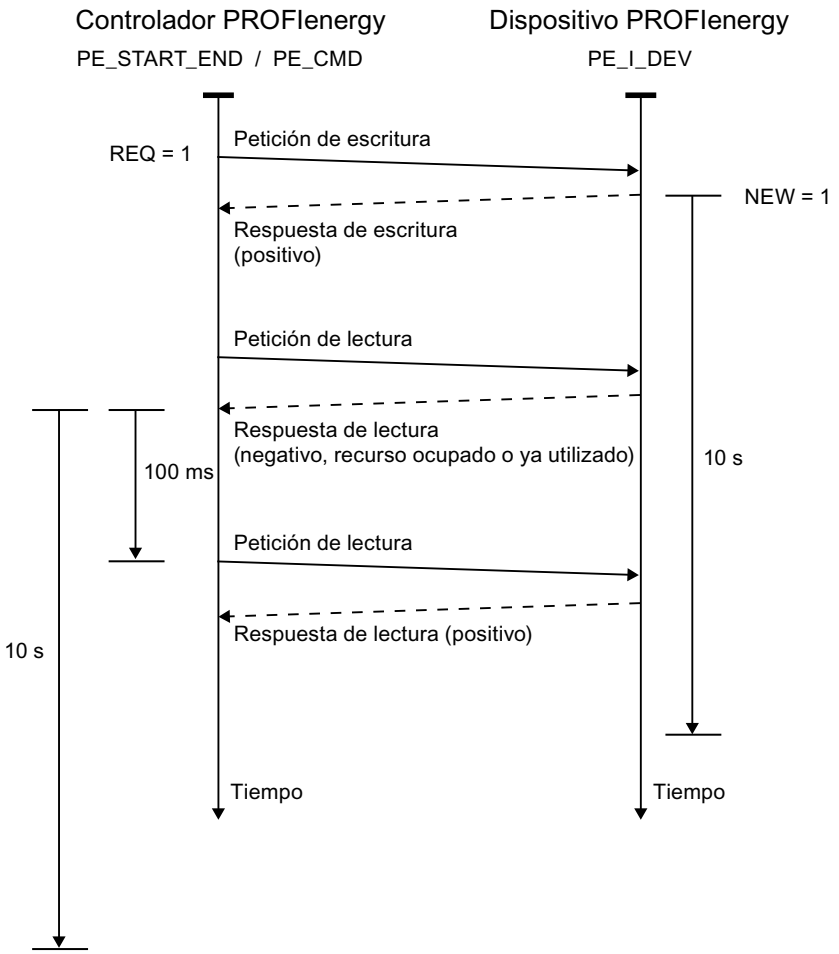
- Los diferentes comandos PE que se transfieren a la unidad PE mediante la instrucción tienen asignadas "Service_Request_IDs" definidas. Las Service_Request_IDs 01...05 y 16 se asignan en el parámetro CMD.
- Mediante el parámetro CMD_MODIFIER se especifican los dos comandos PE 04 (Query_Modes) y 16 (Query_Measurement).
- Para comandos PE concretos, en el parámetro CMD_PARA se transfieren valores adicionales (véase la descripción de los distintos comandos PE). El parámetro CMD_PARA_LEN define la longitud de datos en el parámetro CMD_PARA.

Los comandos se transfieren sin prueba de plausibilidad. Los datos de respuesta de la unidad PE se depositan en el área de datos RESPONSE_DATA direccionada por el puntero VARIANT (los contenidos de los telegramas de respuesta están en las descripciones de los correspondientes comandos PE).

Peticiones de escritura y lectura de la instrucción "PE_CMD"

La instrucción "PE_CMD" envía mediante "WRREC (Página 4748)" un comando PROFIenergy en forma de petición de escritura a la unidad PE. A continuación, "PE_CMD" espera el acuse de la unidad PE. Para ello, cada 100 milisegundos se lee el registro de acuse con la instrucción "RDREC (Página 4746)". Mientras no se reciba acuse de la unidad PE, la petición de lectura se repite cada 100 milisegundos durante 10 segundos. Los datos de respuesta de la unidad PE también se leen con la instrucción "RDREC (Página 4746)".

La figura siguiente muestra el diagrama de ejecución de las peticiones de escritura y lectura:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_CMD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Inicia la transmisión del comando PE con flanco ascendente.
ID	Input	HW_ SUBMODULE	I, Q, M, D, L o constante	Dirección de la unidad PE Para un dispositivo PROFINET IO utilice el ID de hardware del módulo de cabecera. El identificador de hardware figura en las constantes de sistema del controlador IO asignado. El nombre del módulo de cabecera se compone del nombre del dispositivo IO y el complemento [Head] (ejemplo: "IO_Device_1[Head]").

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BYTE	I, Q, M, D, L, P o constante	Service-Request-ID del comando PROFIenergy en función del perfil PROFIenergy (véase "Parámetros CMD y CMD_MODIFIER"). Si se amplía el perfil PROFIenergy son posibles otras Service-Request-IDs.
CMD_MODIFIER	Input	BYTE	I, Q, M, D, L, P o constante	Subcomando PROFIenergy (solo con CMD=3 o CMD=16, véase "Parámetros CMD y CMD_MODIFIER") Si se amplía el perfil PROFIenergy son posibles otros subcomandos.
CMD_PARA	Input	VARIANT	I, Q, M, D, L	Parámetros de los comandos PE: • Get mode: PE_mode_ID • Get measurement values: List of Measurement_Ids Se registra el Service Data Request completo.
CMD_PARA_LEN	Input	INT	I, Q, M, D, L, P o constante	Longitud real de los parámetros del comando (la instrucción comprueba si es <= a la longitud en CMD_PARA).
RESPONSE_DATA	InOut	VARIANT	I, Q, M, D, L	Información PROFIenergy Dependiendo del comando, telegrama de respuesta completo, incluido encabezado de bloque. Nota: si el búfer es demasiado pequeño solo se registra el número de bytes, como se indica en el puntero VARIANT.
VALID	Output	BOOL	I, Q, M, D, L	Comando emitido correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Aún se está ejecutando el comando.
ERROR	Output	BOOL	I, Q, M, D, L	Se ha producido un error durante la ejecución.
STATUS	Output	DWORD	I, Q, M, D, L	Estado de bloque o número de error (véase "Parámetro STATUS"):

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros CMD y CMD_MODIFIER

CMD	CMD_ MODIFIER	Comando PROFIenergy	Descripción
01	0	Start_Pause (Página 4846)	Inicia el modo de ahorro de energía o cambia a otro modo de ahorro de energía.
02	0	End_Pause (Página 4846)	Sale del modo de ahorro de energía.
03	1	Query_Modes - List energy saving Modes (Página 4847)	Muestra los modos de ahorro de energía soportados.

CMD	CMD_ MODIFIER	Comando PROFIenergy	Descripción
	2	Query_Modes - Get Mode (Página 4848)	Muestra los atributos del modo de ahorro de energía activado en el momento actual.
04	0	PEM_Status (Página 4850)	Consulta el estado del modo de ahorro de energía.
05	0	PE_Identify (Página 4852)	Lectura del número de comandos PE soportados y descripción de los mismos.
16	1	Query_Measurement - Get_Measurement_List (Página 4852)	Listado de los valores medidos soportados por la unidad PE.
	2	Query_Measurement - Get_Measurement_Values (Página 4855)	Muestra los valores medidos de la unidad PE.

Parámetro STATUS

En el parámetro de salida STATUS se muestra información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	Causa del error • B#16#00: sin error • B#16#DE: error al leer el registro • B#16#DF: error al escribir el registro • B#16#C0: mensaje de error de las instrucciones de comunicación "RDREC (Página 4746)" y "WRREC (Página 4748)" empleadas internamente.
STATUS[2]	Error_Decode	Ubicación de la identificación de error • 80: error DPV1 según la norma IEC 61158-6 o error específico de instrucción • FE: perfil DP/PNIO; error específico de PROFIenergy

Elemento de campo	Nombre	Significado
STATUS[3]	Error_Code_1	Identificación de error • Con Error_Decode = **80**: – **81**: conflicto de longitud en los parámetros CMD_PARA y CMD_PARA_LEN o bien se ha excedido la longitud máxima del registro (4095 bytes). – **82-8F**: otros mensajes de error (reservado) • Con Error_Decode = **FE**: – **01**: "Service Request ID" no válido – **02**: "Request_Reference" erróneo – **03**: "Modifier" no válido – **04**: "Data Structure Identifier RQ" no válido – **05**: "Data Structure Identifier RS" no válido – **06**: "PE energy-saving modes" no soportados – **07**: "Response" es demasiado largo (se ha rebasado la longitud máxima transferible) – **08**: "Count" no válido – **50**: no hay ningún modo de ahorro de energía (energy mode) adecuado disponible. – **51**: el valor de tiempo indicado no se soporta. – **52**: "PE_Mode_ID" no válido
STATUS[4]	Error_Code_2	Extensión específica de fabricante de la identificación de error

Nota

Mensajes de error de las instrucciones RDREC y WRREC

La instrucción "PE_CMD" utiliza para la comunicación las instrucciones "WRREC (Página 4748)" y "RDREC (Página 4746)". Los mensajes de error de estas instrucciones se emiten correspondientemente en los elementos de campo STATUS[1] a STATUS[4].

Consulte el significado de los códigos de error de las instrucciones "WRREC (Página 4748)" y "RDREC (Página 4746)" en la descripción del parámetro STATUS (Página 4755) correspondiente.

Consulte también

Descripción de PROFIenergy (Página 4820)

PE_DS3_Write_ET200S: Ajustar el modo de activación de los módulos de potencia

Descripción

La instrucción "PE_DS3_Write_ET200S" envía al ET 200S configuraciones básicas sobre el modo de activación de los módulos de potencia. La instrucción "PE_DS3_Write_ET200S"

permite definir el modo de activación de hasta 8 slots del ET 200S (p. ej. para módulos de potencia).

Nota

Esta instrucción no forma parte del perfil PROFIenergy, sino que complementa funciones específicas de SIMATIC.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_DS3_Write_ET200S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ENABLE	Input	BOOL	I, Q, M, D, L o constante	Un flanco ascendente activa la transferencia del registro. El registro debe transferirse de nuevo después de desconectar y volver a conectar la tensión.
ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	Dirección del ET 200S La dirección puede tomarse de la configuración de hardware.
SLOT_NO_X	Input	INT	I, Q, M, D, L, P o constante	Número de slot del X.º módulo de potencia conectable.
FUNC_X	Input	INT	I, Q, M, D, L, P o constante	Función del módulo en este slot. Mediante el parámetro FUNC_X se define el modo de activación del PM-E (módulo de potencia del ET 200S): • FALSE: – Con "PAUSE_START": - no influye en el PM-E - el PM-E continúa conectado – Con "PAUSE_STOP": - se reconecta el PM-E • TRUE: – Con "PAUSE_START": - se desconecta el PM-E – Con "PAUSE_STOP": - se reconecta el PM-E
BUSY	Output	BOOL	I, Q, M, D, L	La transferencia no ha finalizado todavía.
DONE	Output	BOOL	I, Q, M, D, L	La transferencia ha finalizado sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Transferencia finalizada con error.
STATUS	Output	DWORD	I, Q, M, D, L, P	Número de error (véase el parámetro STATUS de la instrucción "PE_Start_End (Página 4822)")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Consulte también

Descripción de PROFIenergy (Página 4820)

PE_WOL: Iniciar y finalizar modo de ahorro de energía mediante WakeOnLan

Descripción PE_WOL

Descripción

La instrucción "PE_WOL" envía los comandos PROFIenergy "Start_Pause" y "End_Pause" a varios dispositivos compatibles con PROFIenergy en sistemas PROFINET I/O.

La instrucción permite la coordinación de varios dispositivos PE siempre que soporten la función "Wake on LAN" a través de una conexión UDP.

La instrucción "PE_WOL" solo puede ejecutarse en una CPU con interfaz Ethernet integrada. Esta CPU debe ser capaz de cargar bloques de un tamaño de aprox. 400 kB. El bloque no se puede utilizar con sistemas PROFINET I/O si estos están conectados a través de un CP Ethernet.

La instrucción "PE_WOL" se ejecuta de forma asíncrona.

Definición: Wake on LAN

La funcionalidad Wake On LAN permite que dispositivos de procesamiento de datos reanuden el trabajo a partir de un estado prácticamente de desconexión gracias a la recepción de un paquete Ethernet especial.

Para que funcione este procedimiento, el dispositivo de procesamiento de datos debe disponer de un controlador de red equipado para poder recibir un paquete de este tipo.

Este paquete (Magic Packet™) tiene un formato especial. Contiene 15 veces la dirección MAC del adaptador de red.

Selección de los dispositivos

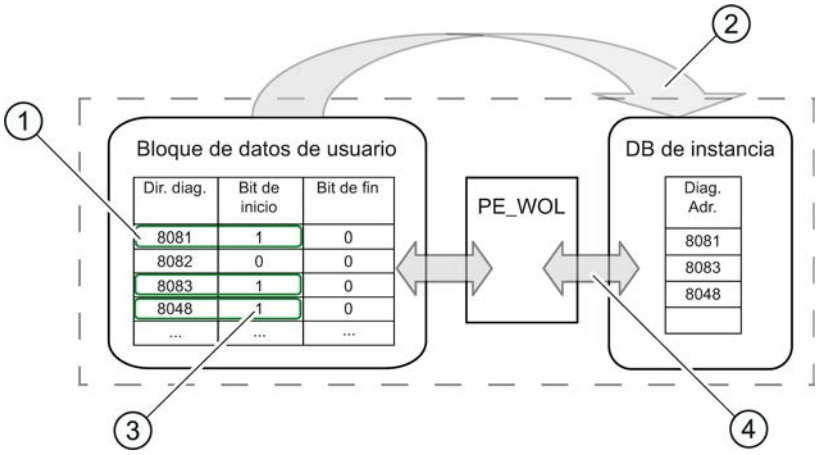
La selección de los dispositivos se realiza a través del bloque de datos de usuario en el parámetro PENERGY (tipo: “PE_PLUS"). El DB de usuario representa la base de datos para el procesamiento de varios dispositivos.

Antes de la inicialización de "PE_WOL" se debe guardar en el DB de usuario la siguiente información, como mínimo:

- ID del sistema PROFINET I/O
- Datos de la conexión que se utiliza para "Wake On LAN"
- Número de puerto que se utiliza para "Wake On LAN"
- Para cada dispositivo
  - Tiempo de pausa (PauseTime)
  - Conmutación del dispositivo al PE_SLEEP_MODE (EnableSleep)

Con el parámetro COM_RST se inicializa la instrucción "PE_WOL". Tras la inicialización se ejecutan las peticiones guardadas en el DB de usuario una tras otra.

El siguiente gráfico muestra a modo de ejemplo cómo se transmite el comando PE "Start_Pause" a varios dispositivos:



(1) Paso 1: El usuario pone a "1" el bit "CmdStartPause" de los dispositivos que se van a desconectar.

(2) Paso 2: Las direcciones de diagnóstico de los dispositivos que se van a desconectar (CmdStartPause = "1") se encadenan en la lista de espera.

(3) Paso 3: El bit "CmdStartPause" se desactiva automáticamente una vez encadenadas las peticiones.

(4) Paso 4: La instrucción "PE_WOL" procesa las peticiones en cuanto estas están encadenadas.

A través de los parámetros START y END se puede enviar un comando PROFIenergy "CmdStartPause" o "CmdEndPause" para todos los dispositivos detectados en el sistema PROFINET IO.

Con el parámetro STATUS se indica el estado del procesamiento de la petición, así como posibles errores durante su procesamiento.

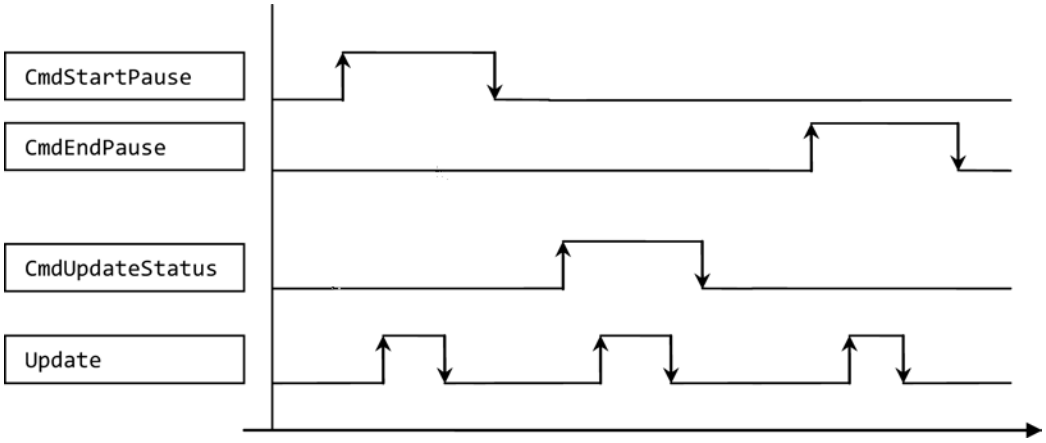
Manejo de la instrucción a través del DB de usuario

El manejo de la instrucción "PE_WOL" solo se puede realizar a través del DB de usuario. A este respecto se aplica un procedimiento básico:

1. Selección del comando que se va a ejecutar para un dispositivo:
   - START_PAUSE ("CmdStartPause" en el DB de usuario)
   - ENDE_PAUSE ("CmdEndPause" en el DB de usuario)
   - UPDATE_STATUS ("CmdUpdateStatus" en el DB de usuario)
2. Activación del bit para la actualización ("Update" en el encabezado del DB de usuario) Entre dos actualizaciones debería transcurrir como mínimo un ciclo de CPU con un "Update" = False, ya que de lo contrario no se puede garantizar la detección de flancos.

Priorización de los comandos PE

El siguiente gráfico muestra la cronología de los tres comandos posibles.



Estos se procesan uno tras otro, independientemente de si el comando previo se ha ejecutado correctamente o ha finalizado con errores.

En caso de activación simultánea de, p. ej., "CmdEndPause" y "CmdUpdateStatus" solo se ejecutará un comando. Dentro del bloque existe una priorización:

- El comando "CmdStartPause" tiene la máxima prioridad y siempre se ejecuta mientras esté seleccionado.
- El comando "CmdEndPause" tiene la segunda máxima prioridad.
- El comando "CmdUpdateStatus" tiene la mínima prioridad.

Si se activan los tres comandos a la vez, los que no se ejecuten permanecen preseleccionados. En ese caso, en el siguiente flanco ascendente se ejecutará el siguiente comando que corresponda.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_WOL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
COM_RST (Página 4836)	Input	BOOL	I, Q, M, D, L	Resetea el bloque y ejecuta una reinicialización. Mientras aquí esté ajustado True, comienza la operación de inicialización pero no se ejecuta hasta el final. La inicialización prosigue con el flanco descendente y, una vez concluida esta, se pasa al modo de operación normal.
START (Página 4837)	Input	BOOL	I, Q, M, D, L	Con un flanco ascendente se ejecuta un comando PROFIenergy "CmdStartPause" para todos los dispositivos detectados que soporten esta función.
END (Página 4838)	Input	BOOL	I, Q, M, D, L	Con un flanco ascendente se ejecuta un comando PROFIenergy "CmdEndPause" para todos los dispositivos detectados que soporten esta función.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PENERGY (Página 4838)	InOut	PE_PLUS	D	Puntero hacia el DB de usuario que contiene la base de datos para el procesamiento de varios dispositivos.
STATUS (Página 4842)	Output	DWORD	I, Q, M, D, L	Número de estado/error para el estado actual de la instrucción (ver "Parámetro STATUS").

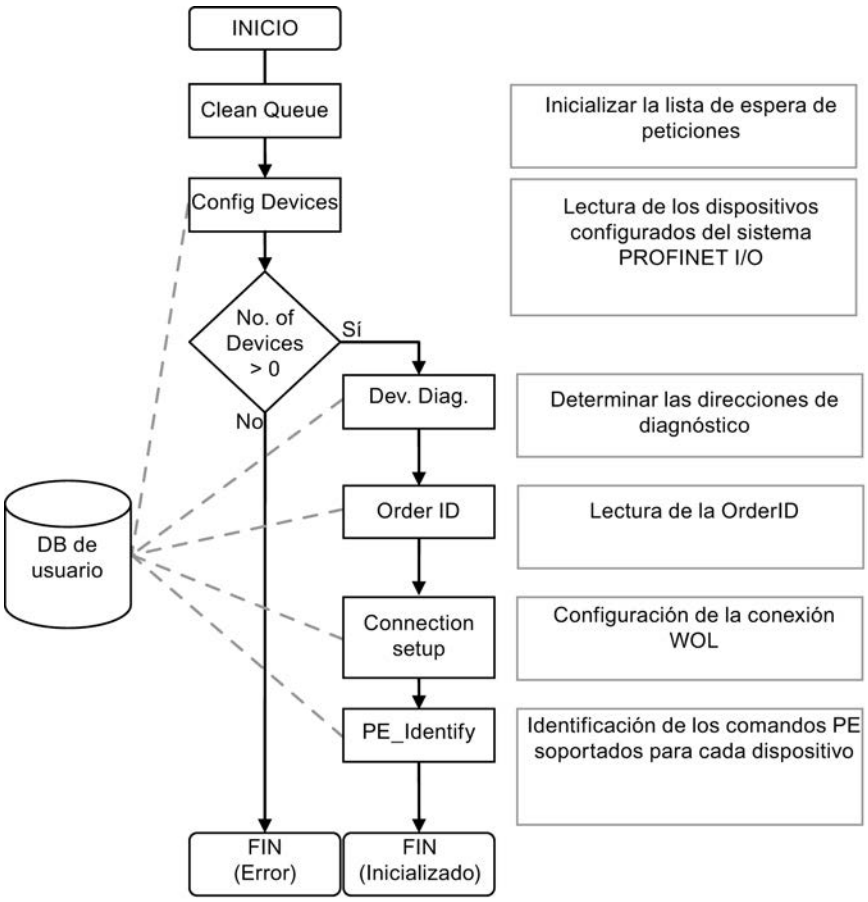
Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro COM_RST

Desarrollo de la rutina de inicialización

Con el parámetro COM_RST se arranca la inicialización de la instrucción "PE_WOL".

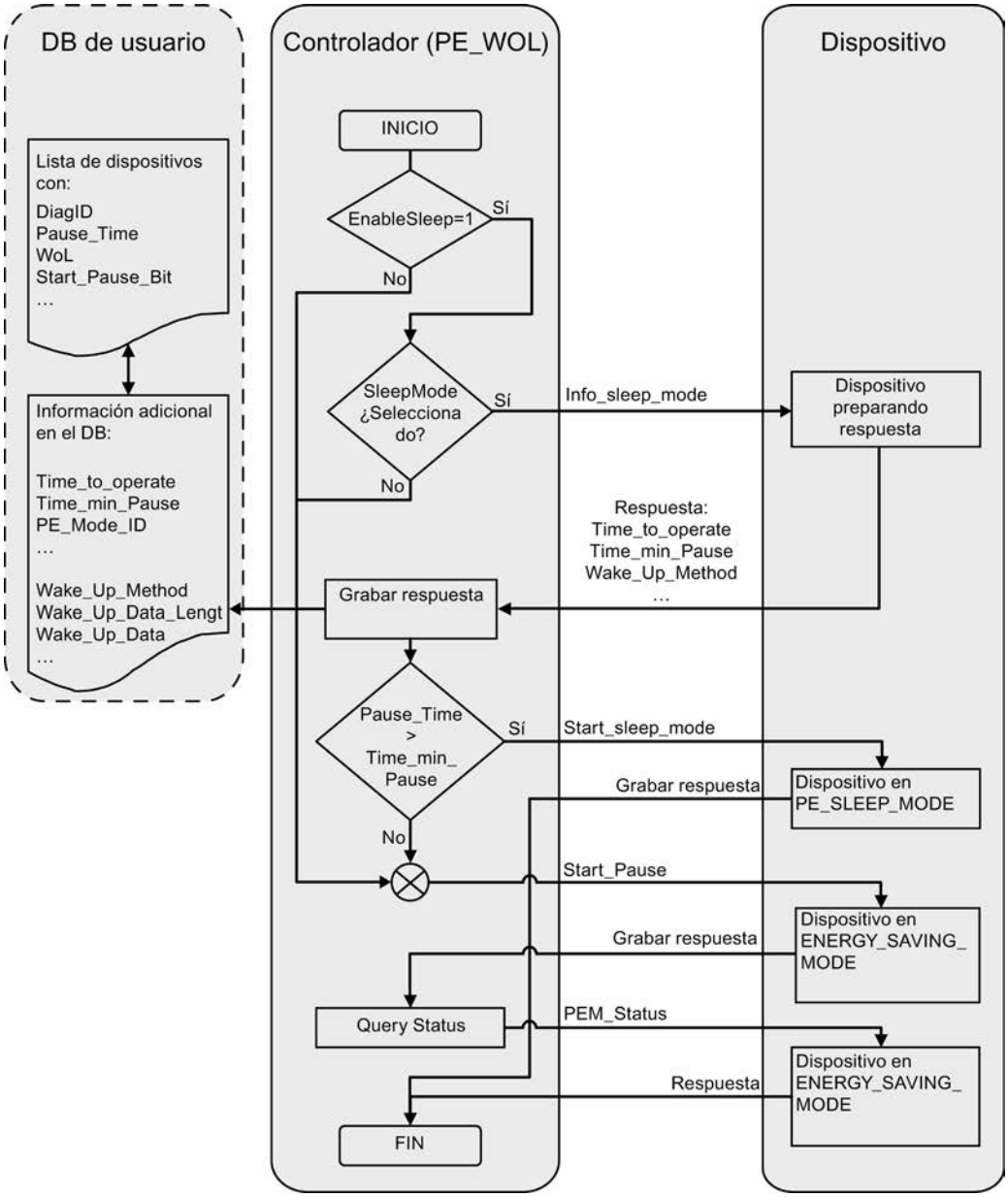
El siguiente diagrama de flujo muestra la rutina de inicialización.



Parámetro START

Desarrollo del comando CmdStartPause

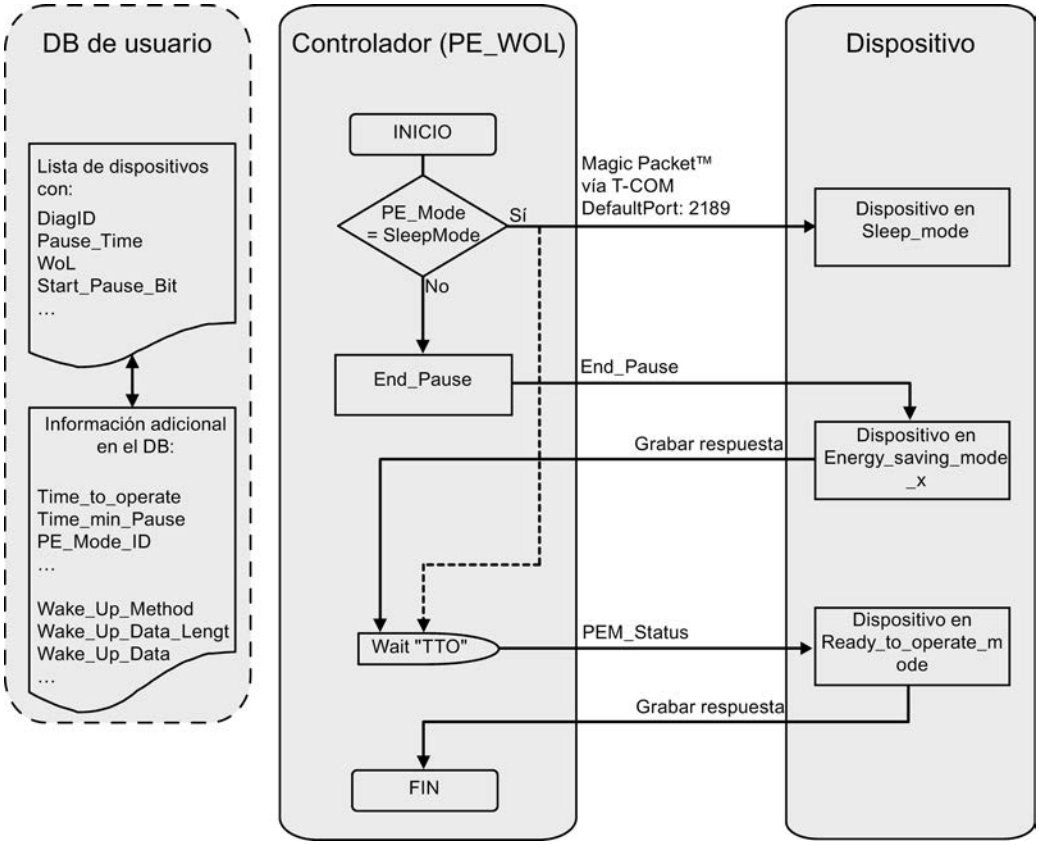
El diagrama de flujo muestra las funciones utilizadas internamente y la interacción con un dispositivo al ejecutar el comando CmdStartPause.



Parámetro END

Desarrollo del comando CmdEndPause

El diagrama de flujo muestra las funciones utilizadas internamente y la interacción con un dispositivo al ejecutar el comando CmdEndPause.



Parámetro PENERGY

Bloque de datos en el parámetro PENERGY

El DB de usuario para la instrucción PROFIenergy "PE_WOL" constituye una base de datos para el procesamiento de varios dispositivos.

El bloque de datos se divide generalmente en dos secciones. A saber:

- Sección de encabezado 110 bytes (header)
- Sección de dispositivos para 256 dispositivos como máximo de 100 bytes cada uno (Device). Estos incluyen:
  - Datos específicos de dispositivo (Device)
  - Datos específicos de PROFIenergy (PE)
  - Datos de procesamiento de petición (Task)
  - Datos de usuario (UserData)

El bloque de datos funciona con acceso optimizado.

Parámetro de conexión "Connection"

La instrucción "PE_WOL" reserva un recurso de conexión del área de "Open User Communication". Este se utiliza como conexión UDP. Para ello, deben definirse los siguientes parámetros en el bloque de datos:

- ID de conexión (parámetro "Connection.id")
  La ID de conexión es un número entero entre 1 y 32. El ajuste estándar es 31. Sirve para identificar los recursos de comunicación asignados por el firmware, como el búfer de transmisión y de recepción.
  La ID de conexión debe ser unívoca en toda la CPU.
- Número de puerto que se utiliza para la función "Wake On LAN" (parámetro "Header.PortNo")
  Número del puerto UDP a través del que se envía un paquete "Wake On LAN". Estos números de puerto forman parte de los recursos de comunicación identificados por la ID de conexión y asignados por el firmware. El ajuste estándar utilizado aquí para el puerto de 2189 no está asignado actualmente por la IANA. El número de puerto se transfiere a la configuración de conexión. Esto repercute en los parámetros "Connection.local_tsap_id[1]" y "Connection.rem_tsap_id[1]".
- ID de interfaz (parámetro "Connection.local_device_id")
  La ID de interfaz también forma parte de la descripción de la conexión. Esta ID identifica la interfaz de la CPU que se debe utilizar para esta conexión. Se admiten varios valores. No obstante, estos deben adaptarse a la CPU y la interfaz utilizadas:
  - B#16#01 con CPU ET200S o WinAC RTX con interfaz Ethernet en subslot IF1
  - B#16#02 con CPU 315(F)-2PN/DP o CPU 317(F)-2PN/DP
  - B#16#03 con CPU 319(F)-2PN/DP
  - B#16#05 con CPU 41x(F)-3PN/DP
  - B#16#06 con WinAC RTX con interfaz Ethernet en subslot IF2
  - B#16#0B con WinAC RTX con interfaz Ethernet en subslot IF3
  - B#16#0F con WinAC RTX con interfaz Ethernet en subslot IF4

Estructura del bloque de datos

El bloque de datos tiene la siguiente estructura:

Nombre	Tipo de datos	Offset	Comentario
Header	PE_HEADER	-	Información de encabezado
Update (1)	BOOL	-	Señal que indica una modificación en el área de datos. • True= indica una modificación por parte del usuario. • False= indica la aplicación de las modificaciones.
Initialized	BOOL	-	Señal que indica el final de la inicialización. • True= inicialización finalizada. • False= indica que el bloque no se ha inicializado.
LinkUp	BOOL	-	Indica la configuración correcta de la interfaz Ethernet. • True= interfaz lista para su uso. • False= interfaz aún no configurada.
LinkDown	BOOL	-	Indica una interfaz sin configurar. • True= interfaz no configurada. • False= la interfaz está configurada o se está configurando actualmente.
PROFINET_ID (1)	INT	-	ID del sistema PROFINET I/O
Reserved	ARRAY [1..37] OF BYTE	-	Reservado
LastDeviceID	INT	-	Contiene la Device-ID más alta en este sistema PROFINET I/O.
PortNo (1)	WORD	-	Número de puerto que se utiliza para la función "Wake On LAN" (valor predeterminado = 2189).
Connection	TCON_Param	-	Contiene la configuración de la conexión "Wake on LAN".
BLOCK_LENGTH	UInt	-	Longitud de la estructura (siempre B#16#40).
ID (1)	CONN_OUC	-	ID de conexión
CONNECTION_TYPE (1)	USINT	-	Tipo de conexión (UDP = B#16#13)
ACTIVE_EST (1)	BOOL	-	Establecimiento activo de la conexión (siempre pasivo con UDP)
LOCAL_DEVICE_ID (1)	USINT	-	Contiene la ID de interfaz (dependiente de la CPU).
LOCAL_TSAP_ID_LEN (1)	USINT	-	Contiene la longitud en bytes del puerto UDP propio/local.
REM_SUBNET_ID_LEN (1)	USINT	-	Sin utilizar (siempre B#16#00)
REM_STADDR_LEN (1)	USINT	-	Contiene la longitud de la dirección IP remota o B#16#00.
REM_TSAP_ID_LEN (1)	USINT	-	Contiene la longitud en bytes del puerto UDP remoto.
NEXT_STADDR_LEN (1)	USINT	-	Contiene la longitud de la dirección del router predeterminado (no relevante).
LOCAL_TSAP_ID (1)	ARRAY[1..16] OF BYTE	-	Contiene el número del puerto propio/local.
REM_SUBNET_ID (1)	ARRAY[1..6] OF USINT	-	Sin utilizar (siempre B#16#00)
REM_STADDR (1)	ARRAY[1..6] OF USINT	-	Contiene la dirección IP remota.

Nombre	Tipo de datos	Offset	Comentario
REM_TSAP_ID (1)	ARRAY[1..16] OF BYTE	-	Contiene el número del puerto UDP remoto.
NEXT_STADDR (1)	ARRAY[1..6] OF BYTE	-	Irrelevante
SPARE (1)	WORD	-	
Device	ARRAY[1..256] OF PE_DEVICE	-	Matriz de los dispositivos
Device	PE_DEV	-	Contiene datos de cada dispositivo.
DiagID	HW_IO	-	Dirección de diagnóstico del dispositivo. Es asignada por la configuración hardware.
InterfaceID	WORD	-	Dirección de diagnóstico de la interfaz del dispositivo. Es asignada por la configuración hardware.
MACAdr	ARRAY[1..6] OF BYTE	-	Contiene la dirección MAC del dispositivo.
IPAdr	ARRAY[1..4] OF BYTE	-	Contiene la dirección IP del dispositivo.
OrderID_MxLen	Byte	-	Contiene la longitud máxima de OrderID.
OrderId_ActLen	Byte		Contiene la longitud actual de OrderID.
OrderID_Data	ARRAY[1..20] OF CHAR		Contiene la OrderID del dispositivo.
PE	PE_PE	-	Datos específicos de PROFIenergy.
ModeID	BYTE	-	PE_MODE_ID conforme a especificación PROFIenergy.
Result	BYTE	-	PE ErrorCode conforme a especificación PROFIenergy.
PauseTime (1)	TIME	-	Contiene el tiempo de pausa en ms.
TimeToPause	TIME	-	Contiene el tiempo que necesita el dispositivo para pasar al modo de pausa.
TimeToOperate	TIME	-	Contiene el tiempo que necesita el dispositivo para pasar al modo de operación.
MinSleepTime	TIME	-	Contiene el tiempo mínimo del dispositivo en PE_SLEEP_MODE.
SleepToOperate	TIME	-	Contiene el tiempo que necesita el dispositivo para estar listo para el servicio desde PE_SLEEP_MODE.
StatusOperate	BOOL	-	Indica el modo de operación del dispositivo.
StatusPause	BOOL	-	Indica el modo de pausa del dispositivo.
StatusSleep	BOOL	-	Indica el PE_SLEEP_MODE del dispositivo.
StatusTransitOK	BOOL	-	Indica la transición de un estado de energía a otro.
StatusInTransit	BOOL	-	Indica una transición de estado actual.
StatusTransitNOK	BOOL	-	Indica que ha fallado el cambio de estado.
StatusError	BOOL	-	Indica un error con el dispositivo.
StatusRetryEx	BOOL	-	Indica la ejecución incorrecta de un comando. No se intenta ejecutar de nuevo este comando.
CmdStartPause (1)	BOOL	-	Posiciona un comando START_PAUSE para este dispositivo en la lista de espera.
CmdEndPause (1)	BOOL	-	Posiciona un comando END_PAUSE para este dispositivo en la lista de espera.

Nombre	Tipo de datos	Offset	Comentario
CmdUpdateStatus (1)	BOOL	-	Posiciona un comando PEM_STATUS para este dispositivo en la lista de espera.
EnableSleep (1)	BOOL	-	Permite el PE_SLEEP_MODE para este dispositivo. • True = el dispositivo debe pasar a PE_SLEEP_MODE si la pausa dura lo suficiente. • False= el dispositivo no debe pasar a PE_SLEEP_MODE.
Services	WORD	-	Indica todos los servicios PROFIenergy soportados.
UserData (2)	ARRAY[1..24] OF BYTE	-	Datos definidos por el usuario
Task	PE_TASK	-	Procesamiento de la petición
Cmd	BYTE	-	Bits internos para el procesamiento de la petición
CmdJ	BYTE	-	Bits internos para el procesamiento de la petición
TimeStart	BOOL	-	Inicia un tiempo de retardo.
TimeStarted	BOOL	-	Acaba de iniciarse un tiempo de retardo.
TimeDone	BOOL	-	Indica la finalización del tiempo de retardo.
Done	BOOL	-	Indica que la petición está finalizándose.
DelayedCmd	BOOL	-	Indica que todavía hay pendiente un comando retardado.
IsV1_0	BOOL	-	Indica que este dispositivo es un dispositivo Spec. V1.0.
IsWakeOnLAN	BOOL	-	Indica que este dispositivo se activa por "Wake On LAN".
RetryCount	BYTE	-	Contador de repeticiones de PE_COMMANDS
Duration	TIME	-	Contiene el valor de retardo en ms.
StartTime	TIME	-	Contiene el punto de inicio del tiempo de retardo.
MachineState	INT	-	Contiene el estado interno de la petición.

(1) Ajuste por parte del usuario.
(2) Uso libre para el usuario.

Parámetro STATUS

Parámetro STATUS

El valor de salida del parámetro STATUS se divide en tres áreas:

- Bits 31 a 24: MESSAGE
- Bits 23 a 16: LOCATION
- Bits 15 a 0: INFORMATION

En las siguientes tablas se explica el significado de los diferentes códigos de error de las tres áreas:

Tabla 9-55 Valores posibles de MESSAGE

Código de error (W#16#...)	Descripción
00	Ningún error.
50	Instrucción inicializada.
51	Se determina la configuración del sistema PROFINET I/O.
52	La instrucción no ha podido localizar dispositivos configurados en el sistema PROFINET I/O.
53	Se determinan las direcciones lógicas de los dispositivos configurados.
54	Se lee la información de interfaz de los dispositivos.
55	Se determinan los datos I&M (solo registro 0) de los dispositivos configurados.
56	Se configura la interfaz PROFINET para el envío de "Wake on LAN" MagicPaket™ a través de UDP.
57	Se determina la compatibilidad PROFIenergy de los dispositivos conectados.
62	Detectada ID no válida del sistema PROFINET I/O. El número causante se indica en el campo INFORMATION.
70	La instrucción se ha inicializado y procesa peticiones. El valor del campo INFORMATION indica el número de las peticiones activas actualmente.
80	La instrucción se ha desinicializado al procesar peticiones. Esto ocurre generalmente cuando vuelve a cargarse el DB de instancia o bien el DB de usuario.
FF	Se ha producido un error desconocido.

Tabla 9-56 Valores posibles de LOCATION

Código de error (W#16#...)	Descripción
00	La instrucción no se ha inicializado o bien está inactiva.
70	La instrucción está esperando peticiones.
71	La instrucción introduce una petición en la lista de peticiones.
72	La instrucción prepara el envío de una petición.
73	La instrucción envía una petición a un dispositivo.
74	La instrucción espera la respuesta del dispositivo.
75	La instrucción evalúa la respuesta del dispositivo.
76	La instrucción elimina la petición de la lista de peticiones.
FF	Se ha producido un error desconocido.

Tabla 9-57 Valores posibles de INFORMATION

Código de error (W#16#...)	Descripción
0000	No hay información adicional disponible ni peticiones activas.
0001 -00FF	Actualmente se procesan 1 – 255 peticiones.
8001	Error en el 1.er parámetro
8002	Error en el 2.º parámetro
8003	Error en el 3.er parámetro

Código de error (W#16#...)	Descripción
8004	Error en el 4.º parámetro
8005	Error en el 5.º parámetro. Este error se indica si no hay ninguna interconexión con el DB de usuario o bien hay una no válida. Causas posibles: • El DB de usuario es demasiado pequeño. • El DB de usuario está protegido contra escritura. • El DB de usuario no está disponible en la RAM. • El DB de usuario no es válido para la CPU utilizada.
8100	Se han intentado posicionar más de las 256 peticiones posibles. Se trata de un error temporal que se soluciona con la finalización de algunas peticiones. La petición que se ha intentado posicionar no ha sido aceptada y debe volver a posicionarse.
8200	Se ha intentado enviar un comando PROFIenergy no válido o no soportado (PE_COMMAND).
84xx	Se ha producido un error de comunicación. El número del dispositivo que ha provocado el error se indica en "xx".
85xx	El dispositivo xx ha notificado un error. El número del dispositivo que ha provocado el error se indica en "xx".
8600	Actualmente no se soporta el método de WakeUp solicitado.
FFFF	Se ha producido un error desconocido.

Comandos PROFIenergy

Formato de los telegramas de respuesta

Estructura del telegrama de respuesta según el perfil PROFIenergy

La tabla siguiente muestra la estructura básica del telegrama de respuesta según el perfil PROFIenergy. El telegrama de respuesta consta de una parte general (Header) y una parte específica (Service Data Response). El contenido de la parte específica del telegrama de respuesta se encuentra en la descripción del comando PROFIenergy correspondiente.

Definición de bloque	Atributos	Valor	Tipo de datos	Descripción
BlockHeader	BlockType	801_{hex}	WORD	
	BlockLength		WORD	Número de bytes sin incluir los campos BlockType y BlockLength.
	BlockVersionHigh	1_{hex}	BYTE	
	BlockVersionLow	0_{hex}	BYTE	

Definición de bloque	Atributos	Valor	Tipo de datos	Descripción
Response Header	Service_Request_ID	1de $_{hex}$ a FF $_{hex}$	BYTE	ID del comando PE ejecutado. La ID del comando PE procesado por la unidad PE se devuelve en el telegrama de respuesta: • 01: Start_Pause • 02: End_Pause • 03: Query_Modes • 04: PEM_Status • 05: PE_Identify • de 06 a 09: Reservado • 16: Query_Measurement • de 11 a CF: Reservado • de D0 a FF: Específico de fabricante
	Request_Reference	1de $_{hex}$ a FF $_{hex}$	BYTE	Número unívoco de identificación del par consulta-respuesta (lo devuelve el servidor en la respuesta).
Service Header Response	Status	1de $_{hex}$ a FF $_{hex}$	BYTE	Información acerca de si se ha ejecutado el comando PE: • 00: Reservado • 01: Finalizado • 02: Finalizado con error • 03: Datos incompletos • de 04 a CF: Reservado • de D0 a FF: En función de la Service_Request_ID
	Data_Structure_ Identifier_RS	1de $_{hex}$ a FF $_{hex}$	BYTE	• 00: Reservado • de 01 a FF: Estructura de datos en función de Service_Request_ID • 0xFF - Error
Service Data Response				En función de Service-Request-ID: • Sobre las Service-Request-IDs, consulte los parámetros CMD y CMD_MODIFIER de la instrucción "PE_CMD (Página 4826)". • El contenido específico del telegrama de respuesta se encuentra en la descripción del comando PE correspondiente (véase p. ej. el comando "Start_Pause (Página 4846)").

Comando PE "Start_Pause"

Descripción

El comando PE "Start_Pause" inicia el modo de ahorro de energía. El comando Start_Pause puede utilizarse para:

- Conmutar la unidad PE desde el estado "Listo para el funcionamiento" (PE_ready_to_operate) a un modo de ahorro de energía (PE_energy_saving_mode).
- Conmutar la unidad PE de un modo de ahorro de energía a otro.
  El consumo de energía puede aumentar o disminuir al cambiar el modo de ahorro de energía.

Llamada del comando PE "Start_Pause"

La llamada del comando "Start_Pause" con la instrucción "PE_CMD (Página 4826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	1	Llamada del comando PE "Start_Pause".
CMD_MODIFIER	0	Para el comando "Start_Pause" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	4	Longitud del parámetro CMD_PARA de 4 bytes.
CMD_PARA	VARIANT	Puntero VARIANT al valor correspondiente a "Pause_Time" (TIME).

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase la instrucción "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
PE_Mode_ID	de 1 a 255	BYTE	Número de identificación del modo de ahorro de energía
Reservado	0	BYTE	-

Comando PE "End_Pause"

Descripción

El comando PE "End_Pause" finaliza el modo de ahorro de energía en la unidad PE.

Llamada al comando PE "End_Pause"

La llamada del comando "End_Pause" con la instrucción "PE_CMD (Página 4826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	2	Llamada del comando PE "End_Pause".
CMD_MODIFIER	0	Para el comando "End_Pause" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
Time_to_operate	-	DWORD	Tiempo de espera para pasar al modo "PE_ready_to_operate"

Comando PE "Query_modes" - "List_Energy_Saving_Modes"

Descripción

Con el comando PE "Query_modes" y el subcomando (Modifier) "List_Energy_Saving_Modes" se indican todos los modos de ahorro de energía (PE_Mode_ID) soportados por la unidad PE.

El resultado de la consulta se escribe en forma de telegrama de respuesta en el bloque de datos referenciado en el parámetro RESPONSE_DATA.

Llamada del comando PE "Query_modes" - "List_Energy_Saving_Modes"

La llamada del comando "List_Energy_Saving_Modes" con la instrucción "PE_CMD (Página 4826)" se realiza con los parámetros siguientes:

Parámetro	Valor	Descripción
CMD	3	Llamada del comando PE "Query_modes".
CMD_MODIFIER	1	Especificación de la llamada del comando: seleccionar el subcomando "List_Energy_Saving_Modes" para mostrar el número y los tipos de modos de ahorro de energía soportados.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
Number_of_PE_ Mode_IDs	1	BYTE	La cantidad de modos de ahorro de energía.
PE_Mode_IDs	-	Array [...] of BYTE	Matriz con las IDs de los modos de ahorro de energía soportados. El significado de las distintas IDs depende de la unidad PE.

Comando PE "Query_modes" - "Get_Mode"

Descripción

Con el comando PE "Query_modes" y el subcomando (Modifier) "Get_Mode" se indican los atributos para el modo de ahorro de energía activado actualmente.

Llamada del comando PE "Query_modes" - "Get_Mode"

La llamada del comando con la instrucción "PE_CMD" se realiza con los siguientes parámetros:

Parámetro	Valor	Descripción
CMD	3	Llamada del comando PE "Query_modes"
CMD_MODIFIER	2	Especificación de la llamada del comando: seleccionar el subcomando "Get_Mode" para mostrar el estado del modo activado en el momento actual.
CMD_PARA_LEN	1	Longitud del parámetro CMD_PARA de 1 byte.
CMD_PARA	VARIANT	VARIANT -Puntero al valor correspondiente a PE_MODE_ID.

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
PE_Mode_ID	• 0 Modo "PE_power_off" • 1...254 Modo de ahorro de energía de la unidad PE (específico de fabricante) • 255 Modo "PE_ready_to_operate"	BYTE	ID del modo de ahorro de energía activo en el momento actual.
PE_Mode_Attributes	Bit 0: • = 0: Solo están disponibles los valores estáticos de consumo de energía y de tiempo. • = 1: Están disponibles los valores dinámicos de consumo de energía y de tiempo. Bits de 1 a 7: • Reservado	BYTE	
Time_min_Pause [1]	Diferencia de tiempo sin fecha	DWORD	Tiempo mínimo de pausa para el modo PE. El tiempo mínimo de pausa es la suma de los valores de los siguientes atributos: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay Véase la descripción de "Parámetro PAUSE_TIME" de la instrucción "PE_START_END: Iniciar y finalizar modo de ahorro de energía (Página 4822)".
Time_to_Pause [1]	Diferencia de tiempo sin fecha	DWORD	Tiempo de desconexión: Tiempo transcurrido desde la llamada del modo de ahorro de energía hasta el inicio de este (tiempo de transición de PE_ready_to_operate a PE_energy_saving_mode). El tiempo de desconexión depende de la unidad PE.
Time_to_operate [1]	Diferencia de tiempo sin fecha	DWORD	Tiempo de conexión: Duración de la transición del modo de ahorro de energía (PE_energy_saving_mode) al modo de disponibilidad (PE_ready_to_operate). La duración es calculada dinámicamente por la unidad PE en el momento de la salida.
Time_min_length_of_stay [1]		DWORD	Tiempo mínimo durante el cual el modo de ahorro de energía está activo en la unidad PE.

Atributo	Valor	Tipo de datos	Descripción
Time_max_length_ of_stay [1]		DWORD	Tiempo máximo durante el cual el modo de ahorro de energía está activo en la unidad PE.
Mode_Power_ Consumption [2]		REAL	Consumo de corriente de la unidad PE con el modo de ahorro de energía activado. Unidad: kW
Energy_ Consumption_ to_pause [2]		REAL	Consumo de energía de la unidad PE durante la transición del modo de disponibilidad (PE_ready_to_operate) al modo de ahorro de energía (PE_energy_saving_mode) Unidad: kWh
Energy_ Consumption_ to_operate [2]		REAL	Consumo de energía de la unidad PE durante la transición del modo de ahorro de energía (PE_energy_saving_mode) al modo de disponibilidad (PE_ready_to_operate) Unidad: kWh
[1] Si la duración es infinita, se devuelve el valor 0xFFFFFFFF. Si la duración es cero, se devuelve "0". [2] Si los datos de energía y consumo de corriente de la unidad PE no están definidos, se devuelve "0,0".			

Comando PE "PEM_Status"

Descripción

Con el comando "PEM_Status" se consulta el estado de un modo de ahorro de energía activado en el momento actual en una unidad PE.

Llamada al comando PE "PEM_Status"

La llamada del comando "PEM_Status" con la instrucción "PE_CMD" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	4	Llamada del comando PE "PEM_Status".
CMD_MODIFIER	0	Para el comando "PEM_Status" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
PE_Mode_ID_ Source	• 0 Modo "PE_power_off" • de 1 a 254 Modo de ahorro de energía de la unidad PE (específico de fabricante) • 255 Modo "PE_ready_to_operate"	BYTE	Modo en el que se encuentra la unidad PE antes de emitirse un comando PE.
PE_Mode_ID_ Destination	• 0 Modo "PE_power_off" • de 1 a 254 Modo de ahorro de energía de la unidad PE (específico de fabricante) • 255 Modo "PE_ready_to_operate"	BYTE	Modo en el que se encuentra la unidad PE después de ejecutarse un comando PE.
Time_to_operate	Diferencia de tiempo sin fecha.	DWORD	Tiempo de conexión: Duración de la transición del modo de ahorro de energía (PE_energy_saving_mode) al modo de disponibilidad (PE_ready_to_operate). La duración es calculada dinámicamente por la unidad PE en el momento de la salida.
Remaining_time_to_ destination	Diferencia de tiempo sin fecha.	DWORD	Tiempo restante para el paso al otro modo.
Mode_Power_ Consumption		REAL	Consumo de corriente de la unidad PE con el modo de ahorro de energía activado. Unidad: kW
Energy_ Consumption_ to_Destination		REAL	Consumo de energía para la transición PE momentánea Unidad: kWh
Energy_ Consumption_ to_operate		REAL	Consumo de energía de la unidad PE durante la transición del modo de ahorro de energía (PE_energy-saving mode) al modo de disponibilidad (PE_ready_to_operate) Unidad: kWh

Comando PE "PE_Identify"

Descripción

Con el comando PE "PE_Identify" se leen el número y la descripción de los comandos PE soportados por la unidad PE. El tipo y la cantidad de comandos que se soporten dependen de la unidad PE. Dado que el propio PE_Identify es un comando PE, en caso de respuesta positiva del dispositivo PE se muestran por lo menos tres comandos PE soportados: Start_Pause, End_Pause y PE_Identify.

Llamada al comando PE "PE_Identify"

La llamada del comando "PE_Identify" con la instrucción "PE_CMD (Página 4826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	5	Llamada del comando "PE_Identify".
CMD_MODIFIER	0	Para el comando "PE_Identify" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
Count [1]	6	BYTE	Número de comandos PROFIenergy soportados
Start_Pause	1	BYTE	Primer comando PE (Service_Request_ID) soportado
End_Pause	2	BYTE	...
Query_Modes	3	BYTE	...
PEM_Status	4	BYTE	...
PE_Identify	5	BYTE	...
Query_Measurement	16	BYTE	Último comando PE (Service_Request_ID) soportado
[1] El número de comandos soportados es específico del fabricante y depende de la unidad PE utilizada. Los valores indicados son un ejemplo de telegrama de respuesta en caso de que se soporten 6 comandos PE.			

Comando PE "Query_Measurement" - "Get_Measurement_list"

Descripción

Con el comando PE "Query_Measurement" y el subcomando (Modifier) "Get_measurement_list" se consultan los valores medidos que soporta la unidad PE. Los

valores medidos soportados se guardan en forma de lista en el bloque de datos referenciado en el parámetro RESPONSE_DATA.

Llamada del comando PE "Query_Measurement" - "Get_Measurement_list"

La llamada del comando con la instrucción "PE_CMD (Página 4826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	16	Llamada del comando "Query_Measurement"
CMD_MODIFIER	1	Especificación de la llamada del comando: seleccionar el subcomando "Get_Measurement_List" para mostrar una lista de los valores medidos soportados.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
Count	-	BYTE	Número de Measurement-IDs
reserved	-	BYTE	
...			
Measurement_ID	-	WORD	Primer Measurement_ID soportado. El Measurement_ID es específico de fabricante. Encontrará más información en el manual de la unidad PE en cuestión.
Accuracy_Domain	-	BYTE	Ver la tabla "Rangos de precisión".
Accuracy_Class	-	BYTE	Ver las tablas "Clases de precisión".
Range	-	REAL	Especifica el valor de fin de escala para el valor medido (solo con rango de precisión 1). El atributo Range utiliza la misma unidad que se haya definido con el atributo Measurement_ID (para cada Measurement_ID se utiliza solo una unidad).
...			
Measurement_ID	-	WORD	Último Measurement_ID soportado
Accuracy_Domain	-	BYTE	Consulte la tabla "Rangos de precisión".
Accuracy_Class	-	BYTE	Consulte las tablas "Clases de precisión".
Range	-	REAL	Especifica el valor de fin de escala para el valor medido (solo con rango de precisión 1). El atributo Range utiliza la misma unidad que se haya definido con el atributo Measurement_ID (para cada Measurement_ID se utiliza solo una unidad).

Rangos de precisión

Rango de precisión	Descripción
0	Reservado
1	La desviación de precisión se expresa en porcentaje del valor de final de escala. El porcentaje de desviación posible se divide en clases de precisión (ver tabla: Clases de precisión de los rangos de precisión 1 y 2).
2	La desviación de precisión se expresa en porcentaje del valor medido real. El porcentaje de desviación posible se divide en clases de precisión (ver tabla: Clases de precisión de los rangos de precisión 1 y 2).
3	La precisión de medida se define según la norma IEC 61557-12. Las clases de potencia de funcionamiento para dispositivos de medición de potencia y supervisión (PMD) sin sensores externos y las clases de potencia del sistema para PMD con sensores externos están codificadas conforme a la tabla "Clases de precisión del rango de precisión 3".
4	La precisión se expresa de acuerdo con la norma EN 50470-3, capítulo 8 (ver también la tabla: Clases de precisión del rango de precisión 4).

Clases de precisión

Tabla 9-58 Clases de precisión de los rangos de precisión 1 y 2

Clase de precisión	0	1	2	3	4	5	6	7	8
Significado	Reservado	0,01%	0,02%	0,05%	0,1%	0,2%	0,5%	1%	1,5%

Clase de precisión	9	10	11	12	13	14	15	>15
Significado	2%	2,5%	3%	5%	10%	20%	>20%	No definido

Tabla 9-59 Clases de precisión del rango de precisión 3

Clase de precisión	0	1	2	3	4	5	6	7	8
Significado	Reservado	0,02	0,05	0,1	0,2	0,5	1	1,5	2

Clase de precisión	9	10	11	12	13	14	>13
Significado	2,5	3	5	10	20	20%	No definido

Tabla 9-60 Clases de precisión del rango de precisión 4

Clase de precisión	0	1	2	3	4	5	6	>7
Significado	Reservado	0,5	1,0	1,5	2,0	2,5	3,0	No definido

Comando PE "Query_Measurement" - "Get_Measurement_values"

Descripción

Con el comando PE "Query_Measurement" y el subcomando (Modifier) "Get_measurement_values" se indican los valores medidos soportados por la unidad PE. Los valores medidos se guardan en forma de lista en el bloque de datos referenciado en el parámetro RESPONSE_DATA.

Llamada del comando PE "Query_Measurement" - "Get_Measurement_values"

La llamada del comando con la instrucción "PE_CMD (Página 4826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	16	Llamada del comando "Query_Measurement"
CMD_MODIFIER	2	Especificación de la llamada del comando: Seleccionar el comando "Get_Measurement_Values" para mostrar una lista de los valores medidos soportados.
CMD_PARA_LEN	0	En función del número de valores medidos. La longitud del parámetro se obtiene a partir del atributo count y la suma de las longitudes de los atributos de los valores medidos transferidos.
CMD_PARA	VARIANT	Puntero VARIANT hacia la estructura de datos con listado de los valores medidos que deben consultarse (véase "Parámetro CMD_PARA").

Parámetro CMD_PARA

La estructura que se indica mediante el puntero VARIANT en el parámetro CMD_PARA debe tener la estructura siguiente:

Atributo	Valor	Tipo de datos	Descripción
Count	-	BYTE	Número de valores medidos (Measurement-IDs)
reserved	0	BYTE	Sin utilizar
Measurement_ID	-	WORD	Primer valor medido consultado
...			
Measurement_ID	-	WORD	Último valor medido consultado

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 4826)"):

Atributo	Valor	Tipo de datos	Descripción
Count [1]	-	BYTE	Número de valores medidos (Measurement-IDs)
reserved	0	BYTE	Sin utilizar

Atributo	Valor	Tipo de datos	Descripción
Length_of_Structure	de 2 a 65535	WORD	Longitud de la estructura en bytes.
Measurement_Data_ Structure_ID	1 = simple value	BYTE	Define la siguiente estructura.
Measurement_ID	de 0 a 65535	WORD	ID del valor medido soportado.
Status_of_ Measurement_Value	de 1 a 3	BYTE	Estado del valor medido: • 1: Válido • 2: No soportado • 3: No válido
Transmission_Data_ Type	-	REAL	
End_of_demand	-	TOD	Sellado de tiempo opcional con el tipo de datos TimeOfDay.
...			
Length_of_Structure	-	WORD	Longitud de la estructura en bytes.
Measurement_Data_ Structure_ID	-	BYTE	Define la siguiente estructura.
Measurement_ID	-	WORD	ID del valor medido soportado.
Status_of_Measurem ent_Value	-	BYTE	Estado del valor medido: • 1: Válido • 2: No soportado • 3: No válido
Transmission_Data_ Type	-	REAL	
End_of_demand	-	TOD	Sellado de tiempo opcional con el tipo de datos TimeOfDay.
[1] Si la longitud de datos de los valores medidos consultados supera el tamaño de la PDU (Protocol Data Unit) del nivel de protocolo, los datos se transfieren de modo incompleto y solo se devuelven los valores medidos soportados.			

iDevice / iSlave

PE_I_DEV: Forzar comandos PROFIenergy en el iDevice

Descripción

La instrucción "PE_I_DEV" se utiliza para desarrollar el perfil PROFIenergy en el dispositivo IO inteligente (iDevice). Las funciones que en un dispositivo IO normal compatible con PROFIenergy, p. ej. el ET 200S, son realizadas por el firmware, en el iDevice son realizadas por la instrucción "PE_I_DEV" y los bloques auxiliares correspondientes:

- el programa de usuario del iDevice llama cíclicamente la instrucción "PE_I_DEV", que recibe todos los comandos PROFIenergy;
- la respuesta de PROFIenergy es generada por parametrización de un bloque auxiliar. La reacción en la pausa puede programarse libremente. Los datos de respuesta deben prepararse en 10 segundos, pues en caso contrario en el parámetro STATUS de la instrucción del controlador IO aparece "Stateconflict 0x80B5".

Para utilizar la instrucción no se requieren conocimientos explícitos del estándar PROFIenergy.

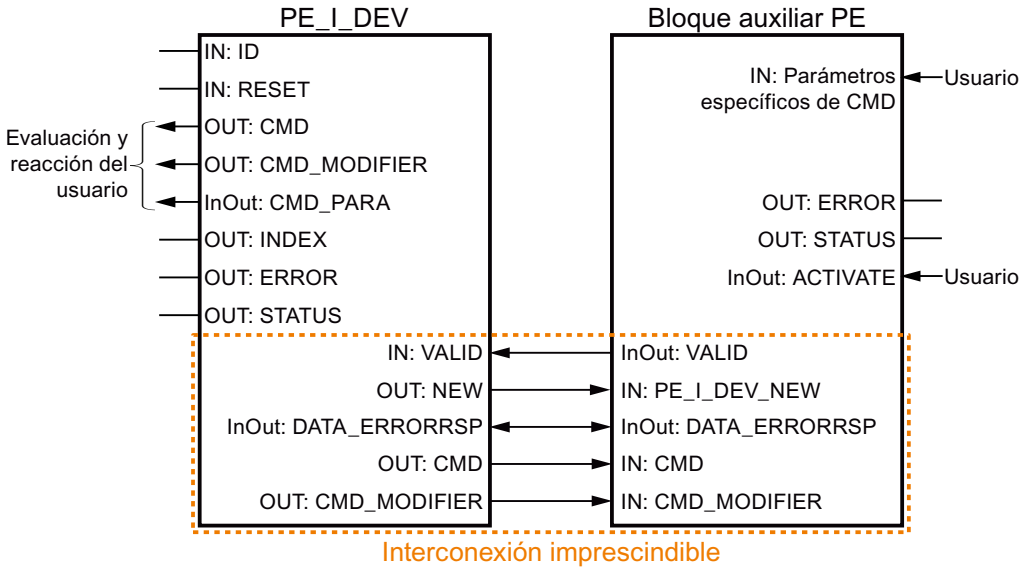
Bloques auxiliares PROFIenergy (bloques auxiliares PE)

Con los bloques auxiliares PE se genera el telegrama de respuesta. Para ello, indique los datos de respuesta (en texto explícito) en los parámetros de entrada del bloque correspondiente.

- Para cada comando PROFIenergy hay un bloque auxiliar correspondiente a una respuesta positiva:
  - comando PE "Start_Pause": PE_Start_RSP (Página 4861)
  - comando PE "End_Pause": PE_End_RSP (Página 4863)
  - comando PE "Query_modes" - "List_Energy_Saving_Modes": PE_List_Modes_RSP (Página 4864)
  - comando PE "Query_modes" - "Get_Mode": PE_Get_Mode_RSP (Página 4865)
  - comando PE "PEM_Status": PE_PEM_Status_RSP (Página 4867)
  - comando PE "PE_Identify": PE_Identify_RSP (Página 4869)
  - comando PE "Query_Measurement" - "Get_Measurement_list": PE_Measurement_List_RSP (Página 4871)
  - comando PE "Query_Measurement" - "Get_Measurement_values": PE_Measurement_Value_RSP (Página 4872)
- Con independencia del comando PROFIenergy utilizado, además hay un bloque auxiliar común para una respuesta negativa (véase PE_Error_RSP (Página 4860)).

Interconexión de los bloques auxiliares

La instrucción "PE_I_DEV" y los bloques auxiliares están adaptados mutuamente. Los parámetros se interconectan en parte de manera sencilla. El gráfico siguiente muestra qué parámetros deben interconectarse.



Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PE_I_DEV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L o constante	Restablece la instrucción.
ID	Input	HW_SUBMODULE	I, Q, M, D, L o constante	Dirección del iDevice La dirección puede tomarse de la configuración de hardware.
VALID	Input	BOOL	I, Q, M, D, L o constante	Los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
CMD_PARA	Output	VARIANT	I, Q, M, D, L	Parámetros de: • Get mode: PE_mode_ID • Get measurement values: lista de Measurement_IDs (lista de IDs de las variables que deben leerse; es posible leer una sola variable o varias a la vez) Longitud máxima: 234 bytes
DATA_ERRORRSP	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el área de datos que contiene los datos de acuse del controlador PROFIenergy. Debe coincidir con el puntero utilizado en los bloques auxiliares.
INDEX	Output	INT	I, Q, M, D, L	Número del registro PROFIenergy (0x80A0 fijo)
CMD	Output	INT	I, Q, M, D, L	Service-Request-ID del comando PROFIenergy en función del perfil PROFIenergy (véase "Parámetros CMD y CMD_MODIFIER"). Si se amplía el perfil PROFIenergy son posibles otros comandos PE (Service-Request-IDs).
CMD_MODIFIER	Output	INT	I, Q, M, D, L	Subcomando PROFIenergy: • solo con CMD=3 o CMD=16, véase "Parámetros CMD y CMD_MODIFIER"; • en todos los demás comandos: "0". Si se amplía el perfil PROFIenergy son posibles otros subcomandos.
NEW	Output	BOOL	I, Q, M, D, L	Nuevos datos disponibles para el controlador PROFIenergy.
ERROR	Output	BOOL	I, Q, M, D, L	Comando finalizado con error.
STATUS	Output	DWORD	I, Q, M, D, L	Información de error (véase "Parámetro STATUS")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros CMD y CMD_MODIFIER

CMD	CMD_ MODIFIER	Comando PROFIenergy	Descripción
01	0	Start_Pause	Inicia el modo de ahorro de energía o cambia a otro modo de ahorro de energía.
02	0	End_Pause	Sale del modo de ahorro de energía.
03	1	Query_Modes - List energy saving Modes	Muestra los modos de ahorro de energía soportados.
	2	Query_Modes - Get Mode	Muestra los atributos del modo de ahorro de energía activado en el momento actual.
04	0	PEM_Status	Consulta el estado del modo de ahorro de energía.
05	0	PE_Identify	Lectura del número de comandos PE soportados y descripción de los mismos.
16	1	Query_Measurement - Get_Measurement_List	Listado de los valores medidos soportados por la unidad PE.
	2	Query_Measurement - Get_Measurement_Values	Muestra los valores medidos de la unidad PE.

Parámetro STATUS

En el parámetro de salida STATUS se muestra información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	Causa del error • B#16#**00**: sin error • B#16#**DE**: error al leer el registro • B#16#**DF**: error al escribir el registro • B#16#**C0**: mensaje de error de la instrucción "PE_I_DEV" o de las instrucciones de comunicación "RDREC (Página 4746)" y "WRREC (Página 4748)" empleadas internamente.
STATUS[2]	Error_Decode	Ubicación de la identificación de error • **80**: error DPV1 según la norma IEC 61158-6 o error específico de instrucción
STATUS[3]	Error_Code_1	Identificación del error (con Error_Decode = **80**): • **B1**: Write length error (error en la longitud de escritura o indicación de longitud insuficiente mediante el tipo de datos VARIANT).
STATUS[4]	Error_Code_2	Con error PROFINET: salida del mensaje de error del controlador IO. Si no se ha producido ningún error PROFINET, el valor de STATUS[4] = "0".

Nota

Mensajes de error de las instrucciones RDREC y WRREC

La instrucción "PE_I_DEV" utiliza para la comunicación las instrucciones "WRREC (Página 4748)" y "RDREC (Página 4746)". Los mensajes de error de estas instrucciones se emiten correspondientemente en los elementos de campo STATUS[1] a STATUS[4].

Consulte el significado de los códigos de error de las instrucciones "WRREC (Página 4748)" y "RDREC (Página 4746)" en la descripción del parámetro STATUS (Página 4755) correspondiente.

Consulte también

Descripción de PROFIenergy (Página 4820)

Módulos auxiliares de la instrucción PE_I_DEV

PE_Error_RSP: Generación de respuesta negativa a comando

Descripción

El bloque auxiliar "PE_Error_RSP" (Response with failure) genera una respuesta negativa si el comando solicitado no es soportado con carácter general o temporalmente. La respuesta se genera con independencia del comando solicitado.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Error_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
ERROR_CODE	Input	BYTE	I, Q, M, D, L o constante	Número de error
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Start_RSP: Generar respuesta a comando para iniciar pausa

Descripción

El bloque auxiliar "PE_Start_RSP" (Iniciar pausa) genera la respuesta al comando PE Start_Pause (Página 4846). La instrucción devuelve el estado de ahorro de energía que adopta el dispositivo (parámetro PE_MODE_ID).

Si las reacciones a pausas de distinta duración son diferentes, se puede indicar en la notificación mediante el estado de ahorro de energía adoptado (PE_Mode_ID = 1 para una pausa corta, PE_Mode_ID = 2 para una pausa más larga, etc.).

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Start_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
PE_MODE_ID	Input	BYTE	I, Q, M, D, L o constante	Modo PE que adopta el proceso
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_End_RSP: Generar respuesta a comando para final de pausa

Descripción

El bloque auxiliar "PE_End_RSP" genera la respuesta al comando PE End_Pause (Página 4846). Se devuelve como respuesta el tiempo necesario para cambiar del estado operativo actual al estado operativo "Ready_To_Operate".

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_End_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
Time_to_Operate	Input	DWORD	I, Q, M, D, L o constante	Tiempo para cambiar del estado operativo actual a "Ready_To_Operate".
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_List_Modes_RSP: Generar respuesta con modos de ahorro de energía consultados

Descripción

El bloque auxiliar "PE_List_Modes_RSP" genera la respuesta al comando PE List_Energy_Saving_Modes (Página 4847). La respuesta generada contiene el número de los estados de ahorro de energía soportados y sus IDs.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_List_Modes_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
Number_of_PE_Mode_IDs	Input	BYTE	I, Q, M, D, L o constante	Número de estados de ahorro de energía soportados. Valores permitidos: de 1 a 254
PE_MODE_ID	Input	VARIANT	I, Q, M, D, L	Señala el área en la que se guardan las IDs de los estados de ahorro de energía soportados (PE_Mode_ID). Área permitida: de 1 a 254.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Get_Mode_RSP: Generar respuesta con datos de energía consultados

Descripción

El bloque auxiliar "PE_Get_Mode_RSP" genera la respuesta al comando Get_Mode (Página 4848). La respuesta contiene los tiempos y los datos de potencia o de energía de cada estado de ahorro de energía concreto.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Get_Mode_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
PE_Mode_ID	Input	BYTE	I, Q, M, D, L o constante	ID del modo de ahorro de energía activo en el momento actual.
Time_min_Pause	Input	DWORD	I, Q, M, D, L o constante	Duración mínima de pausa para este modo de ahorro de energía PE. Está formada por la suma de tres parámetros: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay
Time_to_Pause	Input	DWORD	I, Q, M, D, L o constante	Duración desde el flanco en el parámetro START (véase "PE_I_DEV (Página 4856)") hasta que se alcanza el modo de ahorro de energía PE solicitado.
Time_to_Operate	Input	DWORD	I, Q, M, D, L o constante	Tiempo de conexión máximo hasta "PE_ready_to_operate". El parámetro "Time_to_operate" puede utilizarse directamente para los cálculos correspondientes. El valor puede ser un valor máximo estático o bien puede calcularlo dinámicamente la unidad PE.
Time_min_Lenght_of_stay	Input	DWORD	I, Q, M, D, L o constante	Tiempo mínimo durante el que la unidad PE debe permanecer en este modo PE.
Time_max_Lenght_of_stay	Input	DWORD	I, Q, M, D, L o constante	Tiempo máximo durante el que la unidad PE puede permanecer en este modo PE.
Mode_Power_Consumption	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía en el modo PE actual, en [kW].
Energy_Consum_to_Pause	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía desde "PE_ready_to_operate" hasta el modo PE actual, en [kWh].
Energy_Consum_to_operate	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía desde el modo PE actual hasta "PE_ready_to_operate", en [kWh].

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_PEM_Status_RSP: Generar respuesta con estado PEM

Descripción

El bloque auxiliar "PE_PEM_Status_RSP" genera la respuesta al comando PEM_Status (Página 4850).

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_PEM_Status_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
PE_MODE_ID_ Source	Input	BYTE	I, Q, M, D, L o constante	Source y Destination para PEM_STATUS. Valores: • 0x00: PE_POWER_OFF • 0x01 – 0xFE: específico del fabricante • 0xFF: PE_READY_TO_OPERATE
PE_MODE_ID_ Destination	Input	BYTE	I, Q, M, D, L o constante	
Time_to_Operate[1]	Input	DWORD	I, Q, M, D, L o constante	Tiempo de conexión máximo hasta "PE_ready_to_operate". "Time_to_operate" puede utilizarse directamente para los cálculos correspondientes. El valor puede ser un valor máximo estático o bien puede calcularlo dinámicamente la unidad PE.
Remaining_time_ to_destination[1]	Input	DWORD	I, Q, M, D, L o constante	Opcionalmente: tiempo restante hasta el modo PE solicitado. Valor dinámico o valor máximo estático
Mode_Power_ Consumption[2]	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía en el modo PE actual, en [kW].
Energy_ Consumption_ to_Destination[2]	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía hasta el modo PE solicitado, en [kW].
Energy_ Consumption_ to_operate[2]	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía desde el modo PE actual hasta "PE_ready_to_operate", en [kWh].
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea
[1] Si la duración es ilimitada, puede indicarse el valor máximo "0xFFFFFFFF". Si la duración es "cero", puede utilizarse "0x00". [2] Si el consumo de energía no está definido, puede indicarse "0,0".				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Identify_RSP: Generar respuesta con comandos PROFIenergy soportados

Descripción

El bloque auxiliar "PE_Identify_RSP" genera la respuesta al comando PE_Identify (Página 4852). En la respuesta al comando se indican los comandos PROFIenergy que se soportan. Tenga en cuenta que PE_IDENTIFY es en sí mismo un comando PE y debe indicarse correspondientemente en la respuesta.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Identify_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
Start_Pause	Input	BOOL	I, Q, M, D, L o constante	Un parámetro para cada comando PROFIenergy: • 0: este comando PE no se soporta • 1: este comando PE se soporta
End_Pause	Input	BOOL	I, Q, M, D, L o constante	
Query_Modes	Input	BOOL	I, Q, M, D, L o constante	
PEM_Status	Input	BOOL	I, Q, M, D, L o constante	
PE_Identify	Input	BOOL	I, Q, M, D, L o constante	
Query_ Measurement	Input	BOOL	I, Q, M, D, L o constante	
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Measurement_List_RSP: Generar respuesta con lista de valores medidos soportados

Descripción

El bloque auxiliar "PE_Measurement_List_RSP" genera la respuesta al comando Get_measurement_list (Página 4852). En la respuesta se indican los valores medidos (Measurement-IDs) que se soportan.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Measurement_List_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
Count	Input	BYTE	I, Q, M, D, L o constante	Número de valores medidos soportados (Measurement-IDs)
Measurement_List	Input	VARIANT	D	Puntero hacia la la matriz con las Measurement_IDs soportadas. Consulte la estructura de la matriz según el perfil PROFIenergy en: Comando PE "Query_Measurement" - "Get_Measurement_list" (Página 4852)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Measurement_Value_RSP: Generar respuesta con valores medidos solicitados

Descripción

El bloque auxiliar "PE_Measurement_Value_RSP" genera la respuesta al comando Get_measurement_values (Página 4855). En la respuesta se devuelven los valores de los valores medidos solicitados.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Measurement_Value_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 4856)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 4856)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
Count	Input	BYTE	I, Q, M, D, L o constante	Número de valores medidos (Measurement_Values).
Measurement_Values	Input	VARIANT	D	Puntero hacia la matriz con los valores medidos (Measurement_IDs). Consulte la estructura de la matriz según el perfil PROFIenergy en Comando PE "Query_Measurement" - "Get_Measurement_values" (Página 4855)
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe activarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 4856)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 4856)". El parámetro será activado por el bloque auxiliar cuando estén disponibles los datos de respuesta para el controlador PROFIenergy y puedan enviarse.
DATA_ERRORRSP	InOut	VARIANT	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 4856)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• "0": Ningún error • "1": Con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": Ningún error • "0x80B1": Error en la indicación de VARIANT, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parametrización del módulo

Escribir y leer registros

Principio

Algunos módulos poseen un área de datos del sistema a la que solo es posible acceder desde el programa mediante escritura. Esta área contiene registros con los números de 0 a 240, aunque no todos los módulos poseen todos los registros (ver la tabla siguiente).

Además, los módulos pueden poseer también un área de datos del sistema a la que solo es posible acceder desde el programa mediante lectura. Esta área contiene registros con los números de 0 a 240, aunque no todos los módulos poseen todos los registros.

Nota

Algunos módulos poseen ambas áreas de datos del sistema. Se trata de áreas distintas desde el punto de vista físico, que solo tienen en común la división lógica en registros.

Área de datos del sistema solo accesible por escritura

La tabla siguiente muestra el formato del área de datos del sistema solo accesible por escritura. Indica el tamaño que puede ocupar cada uno de los registros y con qué instrucciones pueden escribirse.

Número de registro	Contenido	Tamaño	Con acceso de escritura mediante instrucción
0	Parámetros	-	WR_DPARM (Página 4882)
1	Parámetros	-	WR_DPARM (Página 4882)
de 2 a 127	Datos del usuario	Cada uno ≤ 240 bytes	WR_DPARM (Página 4882) WR_REC (Página 4790)
de 128 a 240	Parámetros	Cada uno ≤ 240 bytes	WR_DPARM (Página 4882) WR_REC (Página 4790)

Área de datos del sistema solo accesible por lectura

La tabla siguiente muestra el formato del área de datos del sistema solo accesible por lectura. Indica el tamaño que pueden ocupar los distintos registros y con qué instrucciones pueden leerse.

Número de registro	Contenido	Tamaño	Con acceso de lectura mediante instrucción
0	Datos de diagnóstico específicos de módulo (definidos de modo homogéneo para el sistema)	4 bytes	RD_REC (Página 4785)
1	Datos de diagnóstico específicos de canal (incl. registro 0)	de 4 a 220 bytes	RD_REC (Página 4785)
de 2 a 127	Datos del usuario	Cada uno ≤ 240 bytes	RD_REC (Página 4785)
de 128 a 240	Datos de diagnóstico	Cada uno ≤ 240 bytes	RD_REC (Página 4785)

Recursos del sistema

Si se lanzan consecutivamente varias transferencias de registro de funcionamiento asíncrono, se garantiza que todas las peticiones se ejecuten y que no se produzcan interferencias recíprocas.

En el momento en que se alcanza el límite de los recursos del sistema, se envía una notificación mediante RET_VAL. El estado de fallo temporal puede eliminarse repitiendo la petición.

El número máximo de peticiones activas "simultáneamente" para un mismo tipo de instrucción depende de la CPU.

RD_DPAR: Leer registro de un módulo

Descripción

Esta instrucción permite leer el registro con número INDEX del componente direccionado desde los datos del sistema configurados. Puede tratarse de un módulo central o de un componente descentralizado (PROFIBUS DP o PROFINET IO).

El valor TRUE del parámetro de salida VALID indica que el registro se ha transferido correctamente al área de destino RECORD. En ese caso, el parámetro de salida LEN contiene la longitud en bytes de los datos leídos.

Si se ha producido un error durante la transferencia del registro, se señaliza mediante el parámetro de salida ERROR . En ese caso, el parámetro de salida STATUS contiene la información de error.

Funcionamiento

La instrucción "RD_DPAR" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia del registro, debe llamarse "RD_DPAR" con REQ = 1.

Mediante el parámetro de salida BUSY y los bytes 2 y 3 del parámetro de salida STATUS se muestra el estado de la petición. Los bytes 2 y 3 de STATUS corresponden al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

La transferencia del registro se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_DPAR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de lectura
LADDR	Input	HW_IO	I, Q, M, D, L o constante	Número de identificación de la CPU o de la interfaz. El número se asigna automáticamente y se guarda en las propiedades de la CPU o de la interfaz en la configuración hardware.
INDEX	Input	INT	I, Q, M, D, L o constante	Número de registro
RECORD	InOut	VARIANT	I, Q, M, D, L	Área de destino para el registro leído
VALID	Output	BOOL	I, Q, M, D, L	El nuevo registro se ha recibido y es válido
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La petición todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: Se ha producido un error durante la lectura.
STATUS	Output	DWORD	I, Q, M, D, L	• Identificación de llamada (bytes 2 y 3) o código de error • Byte 1: B#16#00, si no hay ningún error. De lo contrario, ID de función de DPV1-PDU: En caso de error al leer registros B#16#DE, en caso de error al escribir registros B#16#DF. Si no se utiliza ningún elemento de protocolo DPV1: B#16#C0. • Byte 4: Extensión específica de fabricante de la identificación de error
LEN	Output	INT	I, Q, M, D, L	Longitud de la información del registro leída

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Explicación	Restricción
0000	Sin errores	-
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".	-
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".	Periferia descentralizada
8090	Dirección indicada no válida en el parámetro LADDR.	-
8092	En el parámetro RECORD se ha indicado un tipo de datos diferente a (Array of) secuencias de bits o enteros.	-
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR.	-
80A1	Acuse negativo al enviar el registro al módulo (módulo extraído durante la transferencia o defectuoso).	-
80A2	Error de protocolo DP con Layer 2, posible error de hardware/interfaz en el esclavo DP.	Periferia descentralizada
80A3	Error de protocolo DP con User-Interface/User.	Periferia descentralizada
80A4	Fallo de comunicación en el bus K.	El error se produce entre la CPU y una interfaz DP externa
80B0	Instrucción no posible con este tipo de módulo, o el módulo no conoce el registro.	-
80B1	La longitud del registro que se debe transferir es errónea.	-
80B2	El slot configurado no está ocupado.	-
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado.	-
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	-
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	-
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe si hay fuentes de perturbación en la instalación.	-
80C5	Periferia descentralizada no disponible o desactivada.	Periferia descentralizada
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano).	Periferia descentralizada
80D0	No hay entradas para el módulo.	-
80D1	El número de registro no está configurado para el módulo (se descartan los números de registro >241).	-
80D2	Módulo no parametrizable según la identificación del módulo.	-
80D5	El registro es estático.	-

Código de error* (W#16#...)	Explicación	Restricción
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)	-
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

RD_DPARM: Leer registro de los datos de sistema configurados (Página 4880)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RD_DPARA: Leer registro de un módulo de forma asíncrona

Descripción

Esta instrucción permite leer el registro con número RECNUM del módulo seleccionado desde los datos del sistema configurados. El registro leído se guarda en el área de destino definida mediante el parámetro RECORD .

Funcionamiento

La instrucción "RD_DPARA" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la lectura debe llamarse la instrucción con REQ = 1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_DPARA":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de lectura
LADDR	Input	HW_IO	I, Q, M, D, L o constante	Número de identificación de la CPU o de la interfaz. El número se asigna automáticamente y se guarda en las propiedades de la CPU o de la interfaz en la configuración hardware.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 0 a 240)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Si no se han producido errores durante la transferencia, deben distinguirse los siguientes casos: • RET_VAL contiene la longitud en bytes del registro realmente leído, en caso de que el área de destino sea mayor que el registro leído. • RET_VAL contiene "0" en caso de que la longitud del registro leído sea igual a la longitud del área de destino.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La petición todavía no ha finalizado.
RECORD	Output	VARIANT	I, Q, M, D, L	Área de destino para el registro leído.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación	Restricción
0000	Sin errores	-
7000	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.	-
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".	Periferia descentralizada
8090	La dirección base lógica indicada no es válida.	-
8092	En el parámetro RECORD se ha indicado un tipo de datos diferente a (Array of) secuencias de bits o enteros.	-
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR.	-
80A1	Acuse negativo al enviar el registro al módulo (módulo extraído durante la transferencia o defectuoso).	-
80A2	Error de protocolo DP con Layer 2, posible error de hardware/interfaz en el esclavo DP.	Periferia descentralizada
80A3	Error de protocolo DP con User-Interface/User.	Periferia descentralizada
80A4	Fallo de comunicación en PBUS+	-
80B0	Instrucción no posible con este tipo de módulo, o el módulo no conoce el registro.	-
80B1	La longitud del registro que se debe transferir es errónea. Con RD_DPARM (Página 4880): La longitud del área de destino delimitada mediante RECORD es insuficiente.	-
80B2	El slot configurado no está ocupado.	-

Código de error* (W#16#...)	Explicación	Restricción
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado.	-
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	-
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	-
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe si hay fuentes de perturbación en la instalación.	-
80C5	Periferia descentralizada no disponible o desactivada.	Periferia descentralizada
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano).	Periferia descentralizada
80D0	No hay entradas para el módulo.	-
80D1	El número de registro no está configurado para el módulo (se descartan los números de registro >241).	-
80D2	Módulo no parametrizable según la identificación del módulo.	-
80D5	El registro es estático.	-
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)	-
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RD_DPARM: Leer registro de los datos de sistema configurados

Descripción

Esta instrucción permite leer el registro con número RECNUMdel módulo direccionado desde los datos del sistema configurados. El registro leído se guarda en el área de destino delimitada mediante el parámetro RECORD.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_DPARM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si las direcciones son idénticas, debe indicarse B#16#54.
LADDR	Input	HW_IO	I, Q, M, D, L o constante	ID de hardware de la CPU o de la interfaz. El número se asigna automáticamente y se guarda en las propiedades de la CPU o de la interfaz en la configuración hardware.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 0 a 240)
RET_VAL	Return	INT	I, Q, M, D, L	Longitud en bytes del registro leído, si el registro leído cabe en el área de destino y no se ha producido ningún error durante la transferencia. Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
RECORD	Output	VARIANT	I, Q, M, D, L	Área de destino para el registro leído.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación	Restricción
0000	Sin errores	-
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".	-
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".	Periferia descentralizada
8090	La dirección base lógica indicada no es válida.	-
8092	En la referencia VARIANT del parámetro RECORD se ha indicado un tipo diferente a BYTE .	-
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR.	-
80B1	La longitud del área de destino delimitada mediante RECORD es insuficiente.	-

Código de error* (W#16#...)	Explicación	Restricción
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	-
80D0	No hay entradas para el módulo.	-
80D1	El número de registro no está configurado para el módulo (se descartan los números de registro >241).	-
80D2	Módulo no parametrizable según la identificación del módulo.	-
80D5	El registro es estático.	-
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)	-
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

WR_DPARM: Transferir registro

Descripción

La instrucción "WR_DPARM" permite transferir el registro con número RECNUM desde los datos de configuración al módulo direccionado. A este respecto es indiferente que el registro sea estático o dinámico.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_DPARM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Solicitud de escritura
LADDR	Input	HW_IO	I, Q, M, D, L o constante	ID de hardware de la CPU o de la interfaz. El número se asigna automáticamente y se guarda en las propiedades de la CPU o de la interfaz en la configuración de hardware.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la escritura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación	Restricción
0000	Sin errores	-
7000	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.	-
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".	Periferia descentralizada
8090	Dirección indicada no válida en el parámetro LADDR.	-
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR.	-
80A1	Acuse negativo al enviar el registro al módulo (módulo extraído durante la transferencia o defectuoso).	-
80A2	Error de protocolo DP con Layer 2, posible error de hardware/interfaz en el esclavo DP.	Periferia descentralizada
80A3	Error de protocolo DP con User-Interface/User.	Periferia descentralizada
80A4	Fallo de comunicación en PBUS+	-
80B0	Instrucción no posible con este tipo de módulo, o el módulo no conoce el registro.	-
80B1	La longitud del registro que se debe transferir es errónea.	-
80B2	El slot configurado no está ocupado.	-
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado.	-
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	-
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	-
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe si hay fuentes de perturbación en la instalación.	-
80C5	Periferia descentralizada no disponible o desactivada.	Periferia descentralizada
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano).	Periferia descentralizada
80D0	No hay entradas para el módulo.	-
80D1	El número de registro no está configurado para el módulo (se descartan los números de registro >241).	-
80D2	Módulo no parametrizable según la identificación del módulo.	-
80D5	El registro es estático.	-
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)	-
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Alarmas

ATTACH: Asignar OB a evento de alarma

Descripción

La instrucción "ATTACH" asigna un bloque de organización (OB) a un evento.

En el parámetro OB_NR se especifica la denominación simbólica o numérica del bloque de organización que se asigna al evento indicado en el parámetro EVENT.

Si el evento indicado en el parámetro EVENT ocurre tras haberse ejecutado correctamente la instrucción "ATTACH", se llama el bloque de organización especificado en el parámetro OB_NR y se ejecuta el programa correspondiente.

Con el parámetro ADD se determina si se deben deshacer o conservar las asignaciones existentes del bloque de organización a otros eventos. Si el parámetro ADD está puesto a "0", las asignaciones existentes se sustituyen por la asignación actual.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ATTACH":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_ATT	I, Q, M, D, L o constante	Bloque de organización (se soportan números hasta 32767).
EVENT	Input	EVENT_ATT	D, L o constante	Evento, p. ej. creación de un error de ejecución (fallo de un módulo de hardware; lectura de la ID HW (16#C0xxyyzz) y consulta en el bloque)
ADD	Input	BOOL	I, Q, M, D, L o constante	Efectos en las asignaciones existentes: • ADD=0 (valor predeterminado): este evento reemplaza todas las asignaciones de evento actuales para el OB. • ADD=1: este evento se agrega a las asignaciones de evento vigentes hasta ahora para este OB.
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0	Sin errores
8090	El OB no existe
8091	El OB es del tipo erróneo
8093	El evento no existe
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

DETACH: Deshacer asignación entre OB y evento de alarma

Descripción

La instrucción deshace en el tiempo de ejecución la asignación que existe entre un bloque de organización y uno o varios eventos.

- Si se ha seleccionado un solo evento, se deshace la asignación del OB a dicho evento. Todas las demás asignaciones existentes en ese momento se conservan. Se puede seleccionar un evento individual a través de la lista desplegable del comodín del operando en el parámetro EVENT.
- Si no se ha seleccionado ningún evento, se deshacen todas las asignaciones existentes en este momento entre el bloque de organización y los eventos.

En el parámetro OB_NRse especifica la denominación simbólica o numérica del bloque de organización cuya asignación al evento indicado en el parámetro EVENTse debe deshacer.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DETACH":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_ATT	I, Q, M, D, L o constante	Bloque de organización (se soportan números hasta 32767).
EVENT	Input	EVENT_ATT	D, L o constante	Evento
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0	Sin errores
1	No existe ninguna asignación (advertencia)
8090	El OB no existe
8091	El OB es del tipo erróneo
8093	El evento no existe
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Alarma cíclica

SET_CINT: Ajustar parámetros de alarma cíclica

Descripción

Con la instrucción se configuran los parámetros para un OB de alarma cíclica. El momento de inicio de un OB de alarma cíclica resulta del correspondiente intervalo de tiempo del OB y del desfase.

- El intervalo de tiempo de un OB es el intervalo en el que se llama el OB regularmente. Si el intervalo de tiempo p. ej. es de 100 µs, el OB se llama cada 100 µs durante la ejecución del programa.
- El desfase es el intervalo de tiempo que se aplaza la llamada de un OB de alarma cíclica. El desfase se puede utilizar para procesar bloques de organización de baja prioridad en una escala de tiempos exacta.

Si no existe el OB o no se soporta el intervalo de tiempo utilizado, se emite el correspondiente mensaje de error en el parámetro RET_VAL.

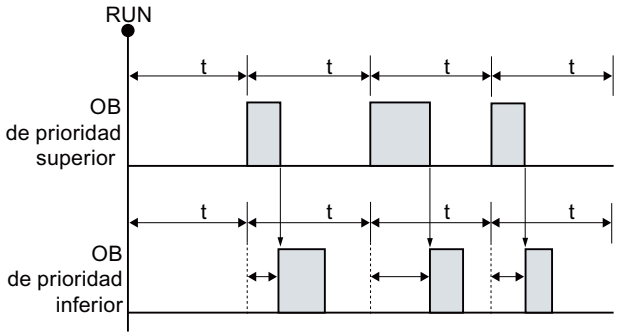
Un intervalo de tiempo de "0" en el parámetro CYCLE significa que el OB no se llama.

Funcionamiento

Si se llaman un OB de baja prioridad y un OB de alta prioridad en el mismo intervalo de tiempo, el OB de baja prioridad no se llama hasta que se haya procesado el OB de alta prioridad. El

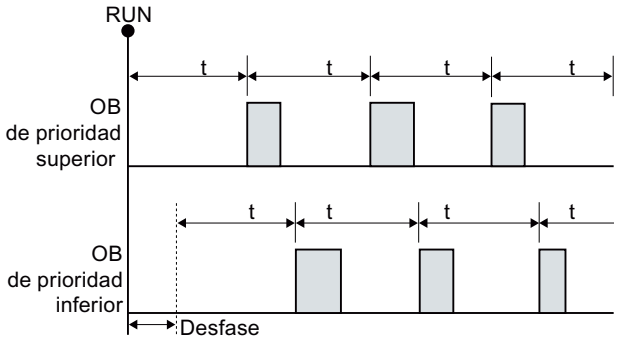
momento de llamada del OB de baja prioridad se puede retardar según la longitud del periodo de procesamiento del OB de alta prioridad.

Llamada de OB sin desfase



Si para el OB de baja prioridad se ha configurado un desfase y este es mayor que el correspondiente tiempo de procesamiento actual del OB de alta prioridad, el bloque se llama según un modelo de tiempos fijo.

Llamada de OB con desfase



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_CINT":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_CYCLIC	I, Q, M, D, L o constante	Número de OB (<32768)
CYCLE	Input	UDINT	I, Q, M, D, L o constante	Intervalo de tiempo en microsegundos
PHASE	Input	UDINT	I, Q, M, D, L o constante	Desfase
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#....)	Descripción
0	Sin error
8090	El OB no existe o el OB es del tipo incorrecto
8091	Intervalo de tiempo incorrecto
8092	Desfase incorrecto
80B2	Ningún evento asignado al OB
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

QRY_CINT: Consultar parámetros de alarma cíclica

Descripción

La instrucción consulta los parámetros actuales de un OB de alarma cíclica. El OB de alarma cíclica se identifica con el parámetro OB_NR.

Los valores de los parámetros consultados de la alarma cíclica se corresponden con los del momento de ejecución de la instrucción "QRY_CINT".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "QRY_CINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_CYCLIC (INT)	I, Q, M, D, L o constante	Número de OB (<32768) o direccionamiento simbólico mediante los nombres del OB (p. ej., OB_MyOB)
CYCLE	Output	UDINT	I, Q, M, D, L	Intervalo de tiempo en microsegundos
PHASE	Output	UDINT	I, Q, M, D, L	Desfase
STATUS	Output	WORD	I, Q, M, D, L	Estado de la alarma cíclica: • Bit 0 a bit 4: Véase el parámetro STATUS • Otros bits: Siempre "0"
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Bit	Valor	Significado
0	0	No utilizado (siempre "0").
1	0	La alarma cíclica está habilitada.
	1	La alarma cíclica está retardada.
2	0	La alarma cíclica no está activada o ha expirado.
	1	La alarma cíclica está activada.
3	0	No utilizado (siempre "0").
4	0	El OB con el número indicado no existe.
	1	El OB con el número indicado existe.
Otros bits		No utilizado (siempre "0").

Parámetro RET_VAL

Si aparece un error, el correspondiente código de error se muestra en el parámetro RET_VAL y el parámetro STATUSse pone a "0".

Código de error* (W#16#....)	Descripción
0	Sin error.
8090	El OB no existe o es del tipo incorrecto.
80B2	Ningún evento asignado al OB.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Alarma horaria

SET_TINT: Ajustar alarma horaria

Descripción

La instrucción "SET_TINT" permite ajustar la fecha y hora de inicio de los bloques de organización de la alarma horaria desde el programa de usuario sin necesidad de realizar ajustes en la configuración hardware.

- En el parámetro OB_NR se indica el número del OB de alarma horaria para el que desee ajustar la fecha y hora de inicio.
- Con los parámetros SDT y PERIOD se indica cuándo y con qué frecuencia debe llamarse el OB de alarma horaria:
  - Llamada única: indique la fecha y hora en el parámetro SDT y aplique el valor "0" en el parámetro PERIOD.
  - Llamada múltiple: indique en el parámetro SDT la fecha y hora de la primera llamada. Ajuste con el parámetro PERIOD el intervalo de tiempo en el que deben efectuarse las llamadas consecutivas del OB.

Tenga en cuenta lo siguiente al ajustar la fecha y hora de inicio:

- La fecha y hora de inicio especificadas en el parámetro SDT hacen referencia a la hora del sistema.
- Si al fijar la hora de inicio se indican segundos y milisegundos, la indicación se ignora y se sustituye por "0".
- Si desea fijar el arranque mensual de un OB de alarma horaria, solo pueden usarse para la fecha de inicio los días 1, 2, ... 28. Esta restricción evita un salto de la llamada mensual (p. ej. en meses de 30 días o en febrero).
  Como alternativa para los días 29, 30 y 31 de un mes, en el parámetro PERIOD se puede utilizar el ajuste "Final de mes" (W#16#2001).

Tras ajustar la alarma horaria con "SET_TINT", aún deberá activarla con la instrucción "ACT_TINT".

Nota

Información adicional sobre los OBs de alarma horaria

En la descripción de los bloques de organización de la respectiva CPU encontrará otras particularidades sobre la utilización de los OBs de alarma horaria:

Para el S7-1200: Auto-Hotspot

Para el S7-1500: Auto-Hotspot

Los ajustes en los parámetros SDT y PERIOD se corresponden con los ajustes asociados a la alarma horaria en las propiedades del OB de alarma horaria.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_TOD	I, Q, M, D, L o constante	Número del OB de alarma horaria • Para los OBs de alarma horaria están disponibles los números 10 a 17. • Alternativamente también se puede asignar un número de OB a partir de 123. El número de OB se visualiza en la carpeta "Bloques de programa" y en las constantes de sistema.
SDT	Input	DT	D, L o constante	Fecha y hora de inicio
PERIOD	Input	WORD	I, Q, M, D, L o constante	Intervalos de ejecución desde el punto de partida SDT : • W#16#0000 = Ejecución única • W#16#0201 = Una vez por minuto • W#16#0401 = Una vez por hora • W#16#1001 = Una vez al día • W#16#1201 = Una vez por semana • W#16#1401 = Una vez al mes • W#16#1801 = Una vez al año • W#16#2001 = A final de mes
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Error en el parámetro OB_NR (ningún OB de alarma horaria direccionado).
8091	Error en el parámetro SDT (indicación de fecha y hora no válida).
8092	Entrada errónea en el parámetro PERIOD.
80A1	El momento inicial ajustado se encuentra en el pasado. Este código de error solo aparece si PERIOD = W#16#0000.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

ACT_TINT: Activar alarma horaria (Página 4895)

SET_TINTL: Ajustar alarma horaria

Descripción

La instrucción "SET_TINTL" permite ajustar la fecha y hora de inicio de los bloques de organización de la alarma horaria desde el programa de usuario sin necesidad de realizar ajustes en la configuración hardware.

- En el parámetro OB_NR se indica el número del OB de alarma horaria para el que desee ajustar la fecha y hora de inicio.
- Con los parámetros SDT y PERIOD se indica cuándo y con qué frecuencia debe llamarse el OB de alarma horaria:
  - Llamada única: indique la fecha y hora en el parámetro SDT y aplique el valor "0" en el parámetro PERIOD.
  - Llamada múltiple: indique en el parámetro SDT la fecha y hora de la primera llamada. Ajuste con el parámetro PERIOD el intervalo de tiempo en el que deben efectuarse las llamadas consecutivas del OB.
- Con el parámetro LOCAL se determina si la indicación de la hora en el parámetro SDT hace referencia a la hora local o a la hora del sistema.
- Con el parámetro ACTIVATE se indica si los ajustes realizados se deben aplicar directamente en el bloque de organización (ACTIVATE = true) o si deben aplicarse en el bloque de organización de alarma horaria tras la llamada de "ACT_TINT (Página 4895)" (ACTIVATE = false).

Tenga en cuenta lo siguiente al ajustar la fecha y hora de inicio:

- Si al fijar la hora de inicio se indican segundos y milisegundos, la indicación se ignora y se sustituye por "0".
- Si desea fijar el arranque mensual de un OB de alarma horaria, solo pueden usarse para la fecha de inicio los días 1, 2, ... 28. Esta restricción evita un salto de la llamada mensual (p. ej. en meses de 30 días o en febrero).
  Como alternativa para los días 29, 30 y 31 de un mes, en el parámetro PERIOD se puede utilizar el ajuste "Final de mes" (W#16#2001).

Tenga en cuenta lo siguiente al utilizar la hora local:

- Cambio del horario de verano al de invierno: Si llama bloques de organización de alarma horaria con una hora de inicio dentro de la segunda hora con cambio horario de verano a invierno, utilice una alarma de retardo adicional durante la primera hora del cambio de hora.
- Cambio del horario de invierno al de verano: si para el día del cambio al horario de verano indica como hora el salto de hora, con una ejecución única (PERIOD = W#16#0000) se visualizará el código de error 16#8091.

Nota

Información adicional sobre los OBs de alarma horaria

En la descripción de los bloques de organización de la respectiva CPU encontrará otras particularidades sobre la utilización de los OBs de alarma horaria:

Para el S7-1200: Auto-Hotspot

Para el S7-1500: Auto-Hotspot

Los ajustes en los parámetros SDT y PERIOD se corresponden con los ajustes asociados a la alarma horaria en las propiedades del OB de alarma horaria.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_TINTL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_TOD	I, Q, M, D, L o constante	Número del OB de alarma horaria • Para los OBs de alarma horaria están disponibles los números 10 a 17. • Alternativamente también se puede asignar un número de OB a partir de 123. El número de OB se visualiza en la carpeta "Bloques de programa" y en las constantes de sistema.
SDT	Input	DTL	D, L o constante	Fecha y hora de inicio
LOCAL	Input	BOOL	I, Q, M, D, L o constante	• true: utilizar hora local • false: utilizar hora del sistema
PERIOD	Input	WORD	I, Q, M, D, L o constante	Intervalos de ejecución desde el punto de partida SDT: • W#16#0000 = Ejecución única • W#16#0201 = Una vez por minuto • W#16#0401 = Una vez por hora • W#16#1001 = Una vez al día • W#16#1201 = Una vez por semana • W#16#1401 = Una vez al mes • W#16#1801 = Una vez al año • W#16#2001 = A final de mes
ACTIVATE	Input	BOOL	I, Q, M, D, L o constante	• true: Ajustar y activar alarma horaria • false: Ajustar alarma horaria y activar exclusivamente al llamar "ACT_TINT (Página 4895)"
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Error en el parámetro OB_NR (ningún OB de alarma horaria direccionado).
8091	Error en el parámetro SDT (indicación de fecha y hora no válida).
8092	Entrada errónea en el parámetro PERIOD.
80A1	El momento inicial ajustado se encuentra en el pasado. Este código de error solo aparece si PERIOD = W#16#0000.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

CAN_TINT: Anular alarma horaria

Descripción

La instrucción "CAN_TINT" permite borrar la fecha y hora de inicio de un bloque de organización de alarma horaria. La alarma horaria se desactiva y el bloque de organización ya no se llama.

Para volver a utilizar la alarma horaria, primero debe ajustarse de nuevo el instante de inicio (instrucción "SET_TINTL (Página 4892)" o "SET_TINT (Página 4890)").

A continuación deberá activar nuevamente la alarma horaria:

- Si para ajustar la alarma horaria ha utilizado la instrucción "SET_TINT (Página 4890)" o "SET_TINTL (Página 4892)" con el parámetro ACTIVE=false, llame "ACT_TINT (Página 4895)".
- En el caso de la instrucción "SET_TINTL (Página 4892)" también es posible activar la alarma horaria directamente con el parámetro ACTIVE=true.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CAN_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_TOD	I, Q, M, D, L o constante	Número del OB de alarma horaria cuya fecha y hora de inicio se deben borrar.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Error en el parámetro OB_NR.
80A0	No se ha establecido ninguna fecha/hora de inicio para el OB de alarma horaria correspondiente.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

ACT_TINT: Activar alarma horaria

Descripción

La instrucción "ACT_TINT" permite activar un bloque de organización de alarma horaria desde el programa de usuario. Para la ejecución de la instrucción es imprescindible que la fecha y hora de inicio para el OB de alarma horaria hayan sido ajustadas previamente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ACT_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_TOD	I, Q, M, D, L o constante	Número del OB de alarma horaria • Para los OBs de alarma horaria están disponibles los números 10 a 17. • Alternativamente también se puede asignar un número de OB a partir de 123. El número de OB se visualiza en la carpeta "Bloques de programa" y en las constantes de sistema.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Error en el parámetro OB_NR (ningún OB de alarma horaria direccionado).
80A0	La fecha y hora de inicio no están ajustadas para el OB de alarma horaria correspondiente.
80A1	La hora activada se encuentra en el pasado. El error solo aparece si la alarma horaria debe ejecutarse una única vez.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

QRY_TINT: Consultar estado de alarma horaria

Descripción

La instrucción muestra el estado de un bloque de organización de alarma horaria en el parámetro de salida STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "QRY_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_TOD	I, Q, M, D, L o constante	Número del OB de alarma horaria cuyo estado debe consultarse. El número de OB se visualiza en la carpeta "Bloques de programa" y en las constantes de sistema.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la alarma horaria (véase a continuación)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error. Excepción: aviso de estado "0" en el bit 4 (ningún OB creado con este número).
8090	Error en el parámetro OB_NR. Causas posibles: • El valor en el parámetro OB_NR no es un número de OB soportado por la CPU (<1 o >32767). • El número de OB no direcciona ningún OB de alarma horaria.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS

Si aparece un error (véase parámetro RET_VAL), en el parámetro STATUS se muestra "0".

Bit	Valor	Significado
0	0	En RUN.
	1	Durante el arranque.
1	0	La alarma horaria está habilitada.
	1	La alarma horaria está bloqueada.
2	0	La alarma horaria no está activada o ha expirado.
	1	La alarma horaria está activada.
4	0	El OB con el número de OB indicado en el parámetro OB_NR no existe.
	1	El OB con el número de OB indicado en el parámetro OB_NR existe.

Bit	Valor	Significado
6	0	La base de la alarma horaria es la hora del sistema
	1	La base de la alarma horaria es la hora local
Otros		Siempre "0"

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Alarma de retardo

Usar alarmas de retardo

Definición

Después de llamar la instrucción "SRT_DINT (Página 4899)", el sistema operativo genera una alarma (es decir, llama el OB de alarma de retardo parametrizado) una vez transcurrido el tiempo de retardo ajustado.

Requisitos para la llamada

Para que el sistema operativo pueda llamar un OB de alarma de retardo, deben cumplirse los siguientes requisitos:

- El OB de alarma de retardo debe haber sido activado mediante la instrucción "SRT_DINT (Página 4899)".
- El OB de alarma de retardo no debe haber sido desactivado mediante configuración.
- El OB de alarma de retardo debe estar presente en la CPU.

Finalidad de las instrucciones "SRT_DINT", "CAN_DINT" y "QRY_DINT"

Con estas instrucciones, las alarmas de retardo se pueden

- Iniciar ("SRT_DINT (Página 4899)")
- Anular ("CAN_DINT (Página 4901)")
- Consultar ("QRY_DINT (Página 4901)").

Repercusiones en la alarma de retardo

La tabla siguiente muestra qué efectos producen las distintas situaciones en las alarmas de retardo.

Si ...	y ...	entonces ...
Se inicia una alarma de retardo (llamada de "SRT_DINT (Página 4899)")	La alarma de retardo ya se ha iniciado,	Se sobrescribe la alarma de retardo; se inicia de nuevo la alarma de retardo.
	El OB de alarma de retardo no está presente en el momento de la llamada,	El sistema operativo genera un error de clase de prioridad (llamada del OB 85; si el OB 85 no está presente, la CPU pasa al estado operativo STOP.)
	Se ha iniciado la alarma en un OB de arranque y el tiempo de retardo expira antes de que la CPU esté en RUN,	La llamada al OB de alarma de retardo se pospone hasta que la CPU esté en RUN.
El tiempo de retardo ha transcurrido	Un OB de alarma de retardo iniciado anteriormente todavía está procesándose,	El sistema operativo genera un error de tiempo (llamada del OB 80; si el OB 80 no está presente, la CPU pasa al estado operativo STOP.)

Comportamiento en rearranque completo (en caliente) o arranque en frío

En rearranque completo (en caliente) o arranque en frío, se borran todos los ajustes de alarma de retardo especificados en el programa de usuario mediante instrucciones.

Inicio en un OB de arranque

Una alarma de retardo puede iniciarse en un OB de arranque. Para llamar el OB de alarma de retardo deben cumplirse dos condiciones:

- El tiempo de retardo ha expirado.
- La CPU está en el estado operativo RUN.

Si el tiempo de retardo ha expirado y la CPU todavía no se encuentra en el estado operativo RUN, la llamada del OB de alarma de retardo se pospone hasta que la CPU se encuentre en dicho estado operativo. Tras ello, el OB de alarma de retardo se llama antes de la primera instrucción en el OB Main [OB 1].

SRT_DINT: Iniciar alarma de retardo

Descripción

La instrucción "SRT_DINT" inicia una alarma de retardo que llama un OB de alarma de retardo una vez transcurrido el tiempo de retardo especificado en el parámetro DTIME. El tiempo de retardo se inicia cuando se genera un flanco descendente en la entrada de habilitación EN. Mientras transcurre el tiempo de retardo, la entrada de habilitación EN debe tener el estado

lógico "0". Si se interrumpe el tiempo de retardo, no se ejecuta el OB parametrizado en el parámetro OB_NR.

Precisión

El tiempo entre la llamada de la instrucción "SRT_DINT" y el arranque del OB de alarma de retardo es como máximo un milisegundo superior al tiempo de retardo parametrizado, siempre y cuando no ocurran eventos de interrupción que retarden la llamada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SRT_DINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_DELAY (INT)	I, Q, M, D, L o constante	Número del OB que se ejecuta tras un tiempo de retardo
DTIME	Input	TIME	I, Q, M, D, L o constante	Tiempo de retardo (de 1 a 60000 ms) Para alcanzar tiempos más prolongados, es posible utilizar p. ej. un contador en un OB de alarma de retardo.
SIGN	Input	WORD	I, Q, M, D, L o constante	Identificación que aparece en la información de evento de arranque al llamar el OB de alarma de retardo.
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Ningún error
8090	Parámetro erróneo OB_NR
8091	Parámetro erróneo DTIME
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

CAN_DINT: Anular alarma de retardo

Descripción

La instrucción anula una alarma de retardo iniciada y, con ello, también la llamada del OB de alarma de retardo que debe ejecutarse una vez transcurrido el tiempo de retardo parametrizado. El número del bloque de organización cuya llamada debe anularse se especifica en el parámetro OB_NR.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CAN_DINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_DELAY (INT)	I, Q, M, D, L o constante	Número del OB cuya llamada debe anularse
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Ningún error
8090	Parámetro erróneo OB_NR
80A0	No se ha iniciado la alarma de retardo.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

QRY_DINT: Consultar estado de una alarma de retardo

Descripción

La instrucción "QRY_DINT" permite consultar el estado de una alarma de retardo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "QRY_DINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	OB_DELAY (INT)	I, Q, M, D, L o constante	Número de OB cuyo estado debe consultarse.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error. En el parámetro STATUS se muestra "0" como valor.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la alarma de retardo; véase la siguiente tabla.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Bit	Valor	Significado
0	0	Sistema operativo en RUN
	1	Sistema operativo en el arranque
1	0	La alarma de retardo está habilitada por el sistema operativo.
	1	Alarma de retardo bloqueada.
2	0	Alarma de retardo no activada o expirada.
	1	Alarma de retardo activada.
3	-	-
4	0	El OB de alarma de retardo con el número indicado no existe.
	1	El OB de alarma de retardo con el número indicado existe.
Otros bits		Siempre "0"

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Indicación errónea en el parámetro OB_NR
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Eventos de errores síncronos

Enmascarar eventos de errores sincrónicos

Introducción

Los eventos de errores sincrónicos son eventos de error de programación y acceso. Estos eventos de error se producen cuando se emplean en la programación áreas o números de operando erróneos o direcciones erróneas. **Enmascarar** estos eventos de error significa lo siguiente:

- Los eventos de error enmascarados no desencadenan ningún OB de error y no provocan ninguna reacción sustitutiva.
- La CPU "anota" en un registro de estado de eventos los errores enmascarados que se han producido.

Para enmascarar errores se debe llamar la instrucción "MSK_FLT (Página 4908)".

Desenmascarar eventos de error significa anular un enmascaramiento efectuado anteriormente y borrar el bit correspondiente en el registro de estado de eventos de la clase de prioridad actual. El enmascaramiento se anula

- Llamando la instrucción "DMSK_FLT (Página 4909)".
- cuando ha terminado la clase de prioridad actual.

Si se produce un evento de error después de ser desenmascarado, el sistema operativo inicia el correspondiente OB de error.

Los eventos de error enmascarados y detectados se pueden leer con la instrucción "READ_ERR (Página 4910)".

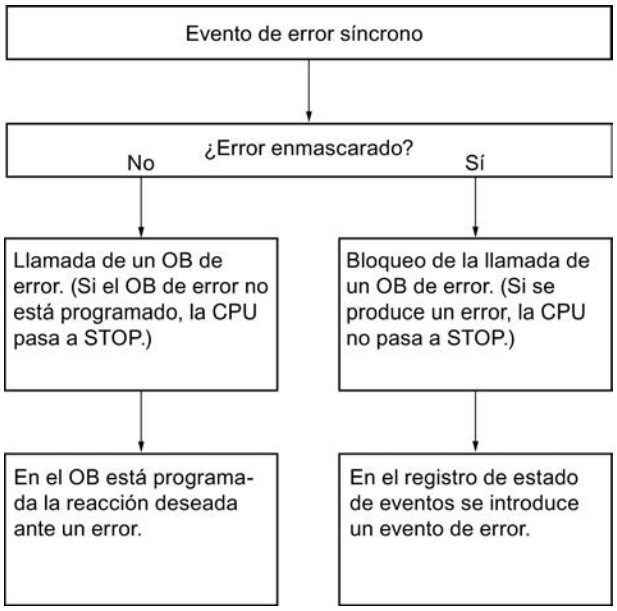
Nota

Independientemente de que estén enmascarados o desenmascarados, con el S7-1500 los eventos de error se registran en el búfer de diagnóstico y se enciende el LED de error de grupo de la CPU.

Tratamiento de errores en general

Cuando se producen errores de programación y acceso a la periferia en un programa de usuario, puede reaccionarse ante ello de diferentes maneras:

- Se puede programar un OB de error, que el sistema operativo llamará cuando se produzca un evento de error relevante.
- Se puede bloquear la llamada al OB de error por separado para cada clase de prioridad. En tal caso, la CPU no pasa a STOP cuando se produce un error relevante en la clase de prioridad bloqueada. La CPU guarda en un registro de estado de eventos los errores que se producen. Sin embargo, esta entrada en el registro no permite saber cuándo y cuántas veces se ha producido el error en cuestión.



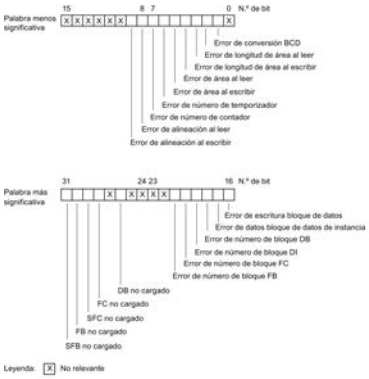
Los **eventos de errores síncronos** están asignados a un determinado patrón de bits, la **máscara de error**. Esta máscara de error se reproduce también en los parámetros de entrada y salida de las instrucciones "MSK_FLT (Página 4908)", "DMSK_FLT (Página 4909)" y "READ_ERR (Página 4910)".

En los eventos de errores síncronos se distingue entre los **errores de programación y los de acceso**, que pueden enmascararse con dos máscaras de error distintas. Encontrará las correspondientes máscaras de error en las siguientes imágenes.

Máscara de error de programación

La siguiente imagen muestra el patrón de bits de la máscara de error para errores de programación. La máscara para errores de programación se encuentra en los parámetros

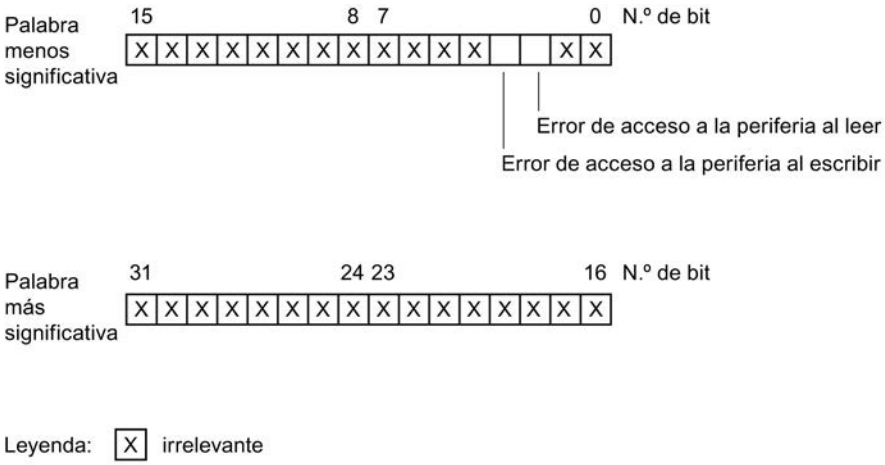
"PRGFLT_..." (ver a continuación "Error de programación, palabra menos significativa" o "Error de programación, palabra más significativa").



Los bits "x" no relevantes para los parámetros de entrada y salida de las instrucciones "MSK_FLT (Página 4908)", "DMSK_FLT (Página 4909)" y "READ_ERR (Página 4910)" se ponen a "0".

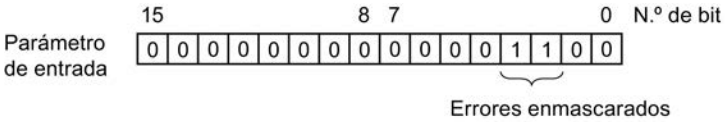
Máscara de errores de acceso

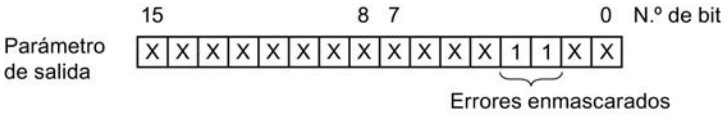
La siguiente imagen muestra el patrón de bits de la máscara de errores de acceso. La máscara de errores de acceso se encuentra en los parámetros ACCFLT_...



Ejemplo: la siguiente imagen muestra el aspecto de la palabra menos significativa de la máscara de errores de acceso con todos los errores enmascarados.

- Como parámetro de entrada para "MSK_FLT (Página 4908)"
- Como parámetro de salida para "MSK_FLT (Página 4908)"





Leyenda:
- X No relevante
- 0 No enmascarados
- 1 Enmascarados

Error de programación, palabra menos significativa

La tabla siguiente contiene los errores que están asignados a la palabra menos significativa de la máscara para errores de programación. Los distintos errores se muestran emparejados con sus posibles causas.

Error	ID de evento (W#16#...)	Causa posible del error
BCD - Error de conversión	2521	El error que se debe convertir no es un número BCD (p. ej. 5E8).
Error de longitud de área durante lectura	2522	Un operando direccionado no se encuentra por completo dentro del área de operandos posible. Ejemplo: Debe leerse MW 320, aunque el área de marca solo ocupa 256 bytes.
Error de longitud de área durante escritura	2523	Un operando direccionado no se encuentra por completo dentro del área de operandos posible. Ejemplo: Debe escribirse un valor en MW 320, aunque el área de marca solo ocupa 256 bytes.
Error de área durante lectura	2524	En un direccionamiento indirecto interárea se ha especificado una identificación de área errónea para el operando. Ejemplo: • Correcto: LAR1 P#E 12.0 L W[AR1, P#0.0] • Erróneo: LAR1 P#12.0 L W[AR1, P#0.0] Con esta operación se notifica el error de área.

Error	ID de evento (W#16#...)	Causa posible del error
Error de área durante escritura	2525	En un direccionamiento indirecto interárea se ha especificado una identificación de área errónea para el operando. Ejemplo: • Correcto: LAR1 P#E 12.0 T W[AR1, P#0.0] • Erróneo: LAR1 P#12.0 T W[AR1, P#0.0] Con esta operación se notifica el error de área.
Error de número de temporizador	2526	Se accede a un temporizador no existente. Ejemplo: • SI T [MW 0] con MW 0 = 129; se solicita iniciar el temporizador 129 aunque solo existen 128 temporizadores.
Error de número de contador	2527	Se accede a un contador no existente. • Ejemplo: ZV Z [MW 0] con MW 0 = 600; se accede al contador 600 aunque solo existen 512 contadores.
Error de alineación durante lectura	2528	Se direcciona un operando de byte, palabra o palabra doble con una dirección de bit ≠ 0. Ejemplo: • Correcto: LAR1 P#M12.0 L B[AR1, P#0.0] • Erróneo: LAR1 P#M12.4 L B[AR1, P#0.0]
Error de alineación durante escritura	2529	Se direcciona un operando de byte, palabra o palabra doble con una dirección de bit ≠ 0. Ejemplo: • Correcto: LAR1 P#M12.0 T B[AR1, P#0.0] • Erróneo: LAR1 P#M12.4 T B[AR1, P#0.0]

Error de programación, palabra más significativa

La tabla siguiente contiene los errores que están asignados a la palabra más significativa de la máscara para errores de programación. Los distintos errores se muestran emparejados con sus posibles causas.

Error	ID de evento (W#16#...)	Causa posible del error
Error de escritura bloque de datos	2530	El bloque de datos en el que se escribe está protegido contra escritura.
Error de datos bloque de datos de instancia	2531	El bloque de datos de instancia en el que se escribe está protegido contra escritura.
Error de número de bloque DB	2532	El número del bloque de datos excede el número máximo admisible.
Error de número de bloque DI	2533	El número del bloque de datos de instancia excede el número máximo admisible.
Error de números de bloque FC	2534	El número de una función (FC) invocada excede el número máximo admisible.

Error	ID de evento (W#16#...)	Causa posible del error
Error de número de bloque FB	2535	El número de un bloque de función (FB) invocado excede el número máximo admisible.
DB no cargado	253A	El bloque de datos no está cargado.
Instrucción no cargada	253C a 253F	La instrucción que se desea llamar no está cargada.

Error de acceso

La tabla siguiente contiene los errores que están asignados a la máscara de errores de acceso. Los distintos errores se muestran emparejados con sus posibles causas.

Error	ID de evento (W#16#...)	Causa posible del error
Error de acceso a la periferia al leer	2942	• La dirección del área de periferia no tiene asignado ningún módulo de señales. • El acceso a dicha área de periferia no se ha acusado dentro del tiempo de vigilancia de módulo ajustado (retardo de acuse).
Error de acceso a periferia al escribir	2943	

MSK_FLT: Enmascarar eventos de error síncrono

Descripción

Esta instrucción permite controlar la reacción de la CPU ante eventos de errores sincrónicos. Para ello se enmascaran los correspondientes eventos de errores sincrónicos (para las máscaras de error, ver Enmascarar eventos de errores sincrónicos (Página 4903)). Llamando a "MSK_FLT" se enmascaran los eventos de errores sincrónicos en la clase de prioridad actual.

Si en los parámetros de entrada se ajustan a "1" algunos bits de las máscaras de errores sincrónicos, los otros bits, ya ajustados anteriormente, mantienen el valor "1". Así se obtienen nuevas máscaras de error que pueden leerse a través de los parámetros de salida. Los eventos de errores sincrónicos enmascarados por el usuario no llaman ningún OB, sino que únicamente se consignan en un registro de estado de eventos. El registro de estado de eventos puede leerse con la instrucción "READ_ERR (Página 4910)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MSK_FLT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_SET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de programación que deben enmascararse
ACCFLT_SET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de acceso que deben enmascararse
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_MASKED	Output	DWORD	I, Q, M, D, L	Errores de programación enmascarados
ACCFLT_MASKED	Output	DWORD	I, Q, M, D, L	Errores de acceso enmascarados

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Ninguno de los errores estaba ya enmascarado.
0001	Por lo menos uno de los errores estaba ya enmascarado; pese a ello se enmascaran los otros errores.
-	Información de error general Consulte también: Consultar ID de error localmente mediante GetErrorID (Página 4460)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DMSK_FLT: Desenmascarar eventos de error síncrono

Descripción

Esta instrucción se utiliza para desenmascarar los eventos de error enmascarados con "MSK_FLT (Página 4908)". Para ello deben ajustarse a "1" en los parámetros de entrada los bits correspondientes de las máscaras de error. Llamando a "DMSK_FLT" se desenmascaran los correspondientes eventos de errores sincrónicos de la clase de prioridad actual. Al mismo tiempo se borran las entradas consultadas en el registro de estado de eventos. Las nuevas máscaras de error pueden leerse a través de los parámetros de salida.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DMSK_FLT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_RESET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de programación que deben desenmascararse
ACCFLT_RESET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de acceso que deben desenmascararse
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_MASKED	Output	DWORD	I, Q, M, D, L	Errores de programación todavía enmascarados
ACCFLT_MASKED	Output	DWORD	I, Q, M, D, L	Errores de acceso todavía enmascarados

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Se han desenmascarado todos los errores especificados.
0001	Por lo menos uno de los errores no estaba enmascarado; pese a ello se desenmascaran los otros errores.
-	Información de error general Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 4460)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

READ_ERR: Leer registro de estado de eventos

Descripción

Esta instrucción permite leer el registro de estado de eventos. El registro de estado de eventos tiene una estructura equivalente a la de las máscaras de errores de programación y acceso que pueden programarse como parámetros de entrada con "MSK_FLT (Página 4908)" y "DMSK_FLT (Página 4909)".

En los parámetros de entrada se especifican los eventos de errores sincrónicos que se desean consultar en el registro de estado de eventos. Llamando "READ_ERR" se leen (y al mismo tiempo se borran) las entradas deseadas del registro de estado de eventos.

En el registro de estado de eventos se puede averiguar cuáles de los errores sincrónicos enmascarados se han producido al menos una vez en la clase de prioridad actual. Un bit activado significa que el error sincrónico correspondiente ha aparecido por lo menos una vez.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "READ_ERR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_QUERY	Input	DWORD	I, Q, M, D, L o constante	Consulta de los errores de programación
ACCFLT_QUERY	Input	DWORD	I, Q, M, D, L o constante	Consulta de los errores de acceso
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
PRGFLT_CLR	Output	DWORD	I, Q, M, D, L	Errores de programación que se han producido
ACCFLT_CLR	Output	DWORD	I, Q, M, D, L	Errores de acceso que se han producido

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Todos los errores consultados están enmascarados.
0001	Por lo menos uno de los errores consultados no está enmascarado.
-	Información de error general Consulte también: Consultar ID de error localmente mediante GetErrorID (Página 4460)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Evento de error asíncrono

DIS_IRT: Bloquear evento de interrupción

Descripción

La instrucción "DIS_IRT" permite bloquear el procesamiento de nuevos eventos de alarma y de errores asíncronos. Bloquear significa que el sistema operativo de la CPU, al producirse un evento de interrupción,

- **ni** llama un OB de alarma o un OB de error asíncrono,
- **ni** desencadena la reacción definida en caso de que no haya ningún OB de alarma ni ningún OB de error asíncrono programado.

El bloqueo del procesamiento de eventos de alarma o de errores asíncronos es válido en todas las clases de prioridad. El bloqueo solo puede eliminarse con la instrucción "EN_IRT (Página 4914)" o con un rearranque completo (en caliente) o un arranque en frío.

El hecho de que el sistema operativo registre o no el evento de alarma o de errores asíncronos en el búfer de diagnóstico depende del parámetro de entrada MODE seleccionado.

Nota

Tenga en cuenta que al programar la instrucción "DIS_IRT" se descartan todas las alarmas que se produzcan.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DIS_IRT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	Indica qué eventos de alarma y errores asíncronos se bloquean.
OB_NR	Input	INT	I, Q, M, D, L o constante	Número de OB
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

MODE (B#16#...)	Significado
00	Se bloquean todos los eventos de alarma y errores asíncronos nuevos que se produzcan. (Los eventos de errores sincrónicos no se bloquean.) Asigne al parámetro OB_NR el valor "0". Se siguen registrando los eventos en el búfer de diagnóstico.
01	Se bloquean todos los eventos nuevos que se produzcan de la clase de alarma indicada. Para identificar la clase de alarma, haga lo siguiente: • Alarmas horarias: 10 • Alarmas de retardo: 20 • Alarmas cíclicas: 30 • Alarmas de proceso: 40 • Alarmas para DPV1: 50 • Alarma de multiprocesamiento: 60 • Alarmas de error de redundancia: 70 • Alarmas de error asíncronas: 80 Se siguen registrando los eventos en el búfer de diagnóstico.
02	Se bloquean todos los eventos nuevos que se produzcan de la alarma indicada. Para identificar la alarma se usa el número de OB. Se siguen registrando los eventos en el búfer de diagnóstico.

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	El parámetro de entrada OB_NR contiene un valor no permitido.
8091	El parámetro de entrada MODE contiene un valor no permitido.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

EN_IRT: Habilitar evento de interrupción

Descripción

La instrucción permite habilitar el procesamiento de nuevos eventos de alarma y errores asíncronos que se ha bloqueado anteriormente con la instrucción "DIS_IRT (Página 4912)". Habilitar significa que el sistema operativo de la CPU, al producirse un evento de interrupción,

- lama un OB de alarma o un OB de error asíncrono
  o
- dispara la reacción definida en caso de que no haya ningún OB de alarma ni ningún OB de error asíncrono programado.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "EN_IRT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	Indica qué eventos de alarma y errores asíncronos se habilitan (ver abajo).
OB_NR	Input	INT	I, Q, M, D, L o constante	Número de OB
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

MODE	Significado
0	Se habilitan todos los eventos de alarma y errores asíncronos nuevos que se produzcan.
1	Se habilitan todos los eventos nuevos que se produzcan de la clase de alarma indicada. Para identificar la clase de alarma, haga lo siguiente: • Alarmas horarias: 10 • Alarmas de retardo: 20 • Alarmas cíclicas: 30 • Alarmas de proceso: 40 • Alarmas para DPV1: 50 • Alarma de multiprocesamiento: 60 • Alarmas de error de redundancia: 70 • Alarmas de error asíncronas: 80
2	Se habilitan todos los eventos nuevos que se produzcan de la alarma indicada. Para identificar la alarma se usa el número de OB.

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	El parámetro de entrada OB_NR contiene un valor no permitido.
8091	El parámetro de entrada MODE contiene un valor no permitido.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

DIS_AIRT: Retardar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad

Descripción

"DIS_AIRT" retarda el procesamiento de OB de alarma cuya prioridad es superior a la del bloque de organización actual.

"DIS_AIRT" se puede llamar varias veces en un bloque de organización. El sistema operativo cuenta las llamadas de "DIS_AIRT". Cada vez que se ejecuta "DIS_AIRT", el procesamiento se retarda aún más. Para revocar un retardo es preciso ejecutar la instrucción "EN_AIRT (Página 4915)". Para revocar todos los retardos, el número de ejecuciones de "EN_AIRT (Página 4915)" debe ser igual al número de llamadas de "DIS_AIRT".

El número de retardos se especifica en el parámetro RET_VAL de la instrucción "DIS_AIRT". Si el parámetro RET_VAL tiene el valor "0", significa que no hay retardos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DIS_AIRT":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Número de retardos

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

EN_AIRT: Habilitar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad

Descripción

Cuando ocurren alarmas, "EN_AIRT" habilita el procesamiento de bloques de organización retardado por la instrucción "DIS_AIRT (Página 4915)".

Ejecutando la instrucción "EN_AIRT" se revoca el retardo de procesamiento que el sistema operativo ha registrado en una llamada de "DIS_AIRT (Página 4915)". Para revocar todos los retardos, el número de ejecuciones de "EN_AIRT" debe ser igual al número de llamadas de

"DIS_AIRT (Página 4915)". Por ejemplo, si "DIS_AIRT (Página 4915)" se ha llamado cinco veces, con lo que el procesamiento también se ha retardado cinco veces, la instrucción "EN_AIRT" deberá ejecutarse cinco veces para poder revocar los cinco retardos.

En el parámetro RET_VAL de la instrucción "EN_AIRT" se especifica el número de retardos de alarma aún no habilitados tras la ejecución de "EN_AIRT". Si el parámetro RET_VAL tiene el valor "0", significa que se han revocado todos los retardos activados por "DIS_AIRT (Página 4915)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "EN_AIRT":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Número de retardos aún programados

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Avisos

Program_Alarm: Crear aviso de programa con valores asociados

Descripción

La instrucción "Program_Alarm" vigila una señal y genera un aviso de programa en caso de que se produzca un cambio de señal en el parámetro SIG (para una definición, véase también: Introducción a la configuración de avisos (Página 3171)). Si la señal cambia de 0 a 1, se genera un aviso de programa entrante; si la señal cambia de 1 a 0, se genera un aviso de programa saliente. El aviso de programa se lanza de modo síncrono a la ejecución del programa.

A los parámetros SD_i (0 ≤ i ≤10) del aviso de programa se pueden adjuntar hasta diez valores asociados. Los valores asociados se captan en el momento del cambio de señal en el parámetro SIG y a continuación se asignan al aviso de programa. En las secciones siguientes encontrará más información sobre la estructura de los valores asociados: Insertar valores asociados en los avisos (Página 3189), Ejemplos de valores asociados (Página 3191).

Todos los avisos, tanto los entrantes como los salientes, se dotan con un sello de tiempo:

- A tal efecto, en caso de producirse un cambio de señal, se utiliza por defecto la hora de sistema actual del PLC (valor predeterminado en el parámetro TIMESTAMP).
- Si desea un sello de tiempo diferente a este, puede crearlo en el parámetro TIMESTAMP. El valor de tiempo debe indicarse siempre como hora del sistema (es decir, UTC), que es la hora que se utiliza para la sincronización horaria de toda la instalación.
- Si un aviso ha de sellarse con una hora local determinada, entonces es necesario intercalar un bloque de conversión que convierta la hora local a la hora del sistema. Solo de esta manera se puede garantizar que los sellos de tiempo se representen correctamente en el aviso.

Para utilizar de nuevo la hora de sistema actual de la CPU, ajuste el parámetro TIMESTAMP a su valor predeterminado (LDT#1970-1-1-0:0:0.0).

Llamada de la instrucción "Program_Alarm"

La instrucción "Program_Alarm" solo es accesible en un bloque de función (FB).

Una vez insertada la instrucción en el FB, en la sección "Static" de la interfaz del FB se crea una multiinstancia del tipo de datos "Program_Alarm". El nombre de la multiinstancia se puede elegir libremente en el cuadro de diálogo que aparece. El nombre de la multiinstancia es, a su vez, el nombre del aviso de programa.

A continuación, complete los parámetros de la instrucción "Program_Alarm" en función de sus necesidades (véase la tabla "Parámetros").

Configuración del aviso de programa

Si selecciona el nombre del aviso de programa en la sección "Static" o en el segmento del FB, los ajustes del aviso de programa se muestran en la ventana "Propiedades". Aquí puede seleccionar la categoría de aviso, la prioridad, etc. y editar el texto del aviso.

Los ajustes aquí efectuados también se pueden editar en el árbol del proyecto. Para ello, abra en "Avisos PLC" la pestaña "Avisos de programa". En la tabla "Tipos de aviso" se muestran todos los avisos de programa creados hasta el momento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Program_Alarm":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIG	Input	BOOL	I, Q, M, D, L, T, C o constante	La señal que debe vigilarse.
TIMESTAMP	Input	LDT	M, D, L o constante	Estos parámetros se utilizan para dotar una alarma de un sello de tiempo que procede, por ejemplo, de una señal de entrada sellada de forma descentralizada. El valor de tiempo debe indicarse siempre como hora del sistema (es decir, UTC) debido a que esta hora se utiliza para la sincronización horaria de toda la instalación. • "No interconectado" significa que la hora de sistema de la CPU se utiliza como sello de tiempo para la alarma en caso de un cambio de señal (valor predeterminado). • La transferencia de una hora de sistema implica que esta se utiliza como sello de tiempo para la alarma en caso de un cambio de señal. Nota: si una alarma ha de sellarse con una hora local determinada, será necesario intercalar un bloque de conversión que convierta la hora local a la hora del sistema. Solo de esta manera se puede garantizar que el sello de tiempo se represente correctamente en la alarma.
SD_i	Input	LREAL, REAL, DINT, INT, SINT, UDINT, UINT, USINT, DWORD, WORD, BYTE, BOOL, WSTRING, STRING, WCHAR, CHAR	I, Q, M, D, L	Valor asociado n.º i (0 ≤ i ≤ 10)
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR ERROR = TRUE significa que se ha producido un error durante el procesamiento. La posible causa del error se visualiza en el parámetro STATUS.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS Visualización de la información de error (véase "Parámetros ERROR y STATUS").

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La tabla siguiente contiene toda la información de error específica que puede devolverse a través de los parámetros ERROR y STATUS .

ERROR	STATUS*	Explicación
0	0000	Ningún error
1	0085	Aviso del tipo "Solo información".
1	8001	Información de aviso estática no válida.
1	8002	Sin información de aviso estática válida.
1	8004	Los valores asociados del aviso han alcanzado el tamaño máximo de 512 bytes.
1	8005	Aviso entrante.
1	8007	Aviso saliente para el que no hay aviso entrante previo.
1	8087	Avisos estáticos desactivados.
1	8089	Aviso demasiado largo.
1	80Ax	Valor no válido en el parámetro SD_i.
1	80C1	La CPU no puede generar avisos en este momento, ya que hay rutinas de inicialización en curso. (Este es el caso, p. ej., tras descargar en RUN). Vuelva a intentarlo más tarde.
1	80C2	Se ha enviado el número máximo de avisos permitido por unidad de tiempo. Vuelva a intentarlo más tarde.
1	80C3	Todas las instancias de aviso dinámicas están en uso. Vuelva a intentarlo más tarde.
1	80C4	Se emite un aviso, que no se puede sobrescribir. Vuelva a intentarlo más tarde.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

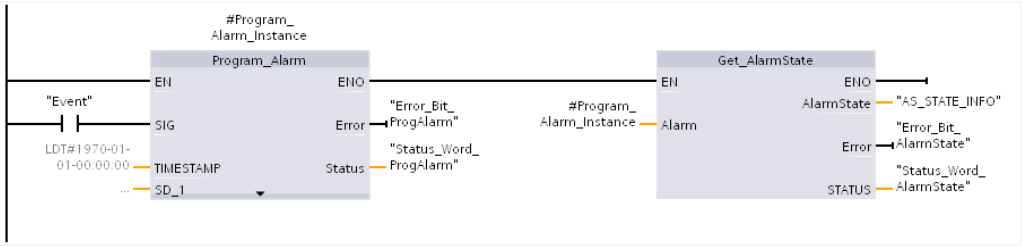
Cambiar formatos de visualización en el estado del programa (Página 3100)

Get_AlarmState: Emitir estado del aviso

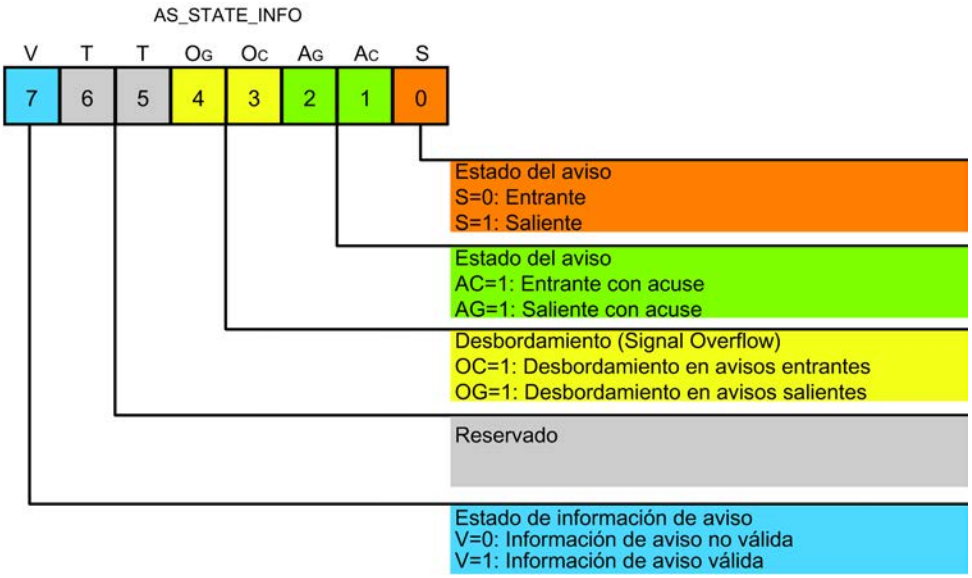
Descripción

La instrucción "Get_AlarmState" emite el estado de un aviso de programa. La emisión del estado del aviso se refiere a un aviso de programa generado mediante la instrucción "Program_Alarm (Página 4916)".

La selección del aviso de programa se efectúa mediante el parámetro de entrada Alarm. En el parámetro Alarm se indica el DB de instancia de la instrucción "Program_Alarm".



A través del parámetro de salida AlarmState se emite el estado del aviso en un byte. El significado de los diferentes bits se ilustra en la siguiente figura:



Para más información sobre los posibles estados de aviso, consulte el apartado "Crear y editar avisos (Página 3186)".

A través de los parámetros de salida ERROR y STATUS se indica el estado de ejecución de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Get_AlarmState":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Alarm	Input	ALARM_BASE	D	Instancia de la instrucción "Program_Alarm (Página 4916)".
AlarmState	Output	BYTE	I, Q, M, D, L	Estado del aviso como mapa de bits.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ningún error • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Explicación
8001	Instancia de aviso estática no válida.
8002	ID del aviso no válida.
8003	No hay avisos activos dentro de la categoría de aviso.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Gen_UsrMsg: Crear avisos de diagnóstico de usuario

Descripción

La instrucción "Gen_UsrMsg" permite generar un aviso que se introduce en el búfer de diagnóstico.

Con el parámetro Mode se elige si debe generarse un aviso entrante o saliente:

- Si Mode = 0: se genera un aviso entrante.
- Si Mode = 1: se genera un aviso saliente.
- Independientemente de si se genera un aviso entrante o saliente, el aviso tiene siempre el atributo "Solo información".

El contenido del aviso se define a través de una lista de textos:

- Con el parámetro TextListID se indica la lista de textos que se desea utilizar. Para ello, abra el cuadro de diálogo "Listas de textos" del árbol del proyecto. Expanda la columna "ID" en el cuadro de diálogo "Listas de textos". Aplique la ID en el parámetro TextListID.
- Con el parámetro TextID se selecciona la entrada de la lista de textos que se desea escribir en el búfer de diagnóstico. Para ello, seleccione en el cuadro de diálogo "Entradas de la lista de textos" una entrada en la que se debe aplicar el número que figura en las columnas "Rango de / Rango hasta" del parámetro TextID. Para ello, debe utilizarse el mismo número para la entrada de la lista de textos "Rango de" y "Rango hasta", respectivamente.
- Encontrará más información sobre las listas de textos en: Listas de textos (Página 420)

En la entrada de la lista de textos se pueden definir adicionalmente valores asociados que se pueden agregar al aviso para complementarlo:

- Los valores asociados se definen al complementar la entrada de la lista de textos con la siguiente información: @<N.º del valor asociado><Formato>@.
- Mediante el tipo de datos de sistema AssocValues se especifica qué valor asociado debe complementarse al generar el aviso.
- Encontrará más información sobre la estructura de los valores asociados en: Estructura de los valores asociados (Página 3183)

La entrada en el búfer de diagnóstico se crea de forma síncrona. La transmisión del aviso se realiza de forma asíncrona.

Si al ejecutar la instrucción se produce un error, este se visualiza a través del parámetro RET_VAL.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Gen_UsrMsg":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Mode	Input	UInt	I, Q, M, D, L o constante	Parámetro para seleccionar el estado del aviso: • 1: aviso entrante • 2: aviso saliente
TextID	Input	UInt	I, Q, M, D, L o constante	ID de la entrada de la lista de textos que debe utilizarse para el texto del aviso.
TextListID	Input	UInt	I, Q, M, D, L o constante	ID de la lista de textos que contiene la entrada de la lista de textos.
Ret_Val	Return	Int	I, Q, M, D, L	Código de error de la instrucción.
AssocValues	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el tipo de datos de sistema AssocValues con el que se definen los valores asociados.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro AssocValues

Con el tipo de datos de sistema AssocValues se define qué valores asociados deben enviarse adjuntos. Se admiten como máximo 8 valores asociados. La estructura del tipo de datos de sistema se crea introduciendo "AssocValues" como tipo de datos en un bloque de datos.

Los valores asociados se seleccionan al introducir los números de los valores asociados en los parámetros Value[x]. El valor asociado debe utilizarse en la entrada de la lista de textos seleccionada (parámetro TextID). Para ello, considere lo siguiente:

- La instrucción procesa los valores para TextID y TextListID como los valores asociados que se van a enviar. Es decir, las cifras "1" y "2" no se pueden utilizar como números para los valores asociados.
- Por tanto, utilice en el parámetro Value [0] el "3" como primer número, en el parámetro Value [1] el "4", y así sucesivamente.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0..1	Value[1]	UINT	0	Primer valor asociado del aviso.
2..3	Value[2]	UINT	0	Segundo valor asociado del aviso.
4..5	Value[3]	UINT	0	...
6..7	Value[4]	UINT	0	...
8..9	Value[5]	UINT	0	...
10..11	Value[6]	UINT	0	...
12..13	Value[7]	UINT	0	...
14..15	Value[8]	UINT	0	Octavo valor asociado del aviso.

Parámetro RET_VAL

La tabla siguiente contiene toda la información de error específica que puede emitirse a través del parámetro RET_VAL .

Código de error* (W#16#...)	Explicación
0000	Ningún error
8080	Valor del parámetro Mode no soportado.
80C1	Escasez de recursos debido a demasiadas llamadas paralelas.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Crear y editar avisos (Página 3186)

Diagnóstico

RD_SINFO: Leer información de arranque del OB actual

Descripción

La instrucción "RD_SINFO" permite leer la información de arranque

- del último OB llamado que todavía no se ha procesado completamente o
- del último OB de arranque iniciado.

Ambas informaciones de arranque carecen de sello de tiempo. Si la llamada se produce en el OB 100, el OB 101 o el OB 102, se devuelven dos informaciones de arranque idénticas.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "RD_SINFO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
TOP_SI	Output	VARIANT	D, L	Información de arranque del OB actual
START_UP_SI	Output	VARIANT	D, L	Información de arranque del último OB de arranque iniciado

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

SDT del parámetro TOP_SI

La tabla siguiente muestra los SDT posibles del parámetro TOP_SI:

Bloques de organización (OB)	Tipos de datos de sistema (SDT)	Números de los tipos de datos de sistema
Todos	SI_classic	592
	SI_none	593
ProgramCycleOB	SI_ProgramCycle	594
TimeOfDayOB	SI_TimeOfDay	595
TimeDelayOB	SI_Delay	596
CyclicOB	SI_Cyclic	597
ProcessEventOB	SI_HWInterrupt	598
ProfileEventOB StatusEventOB UpdateEventOB	SI_Submodule	601
SynchronousCycleOB	SI_SynchCycle	602
IOredundancyErrorOB	SI_IORedundancyError	604
CPUredundancyErrorOB	SI_CPURedundancyError	605
TimeErrorOB	SI_TimeError	606
DiagnosticErrorOB	SI_DiagnosticInterrupt	607
PullPlugEventOB	SI_PlugPullModule	608
PeripheralAccessErrorOB	SI_AccessError	609
RackStationFailureOB	SI_StationFailure	610
ServoOB	SI_Servo	611
IpoOB	SI_Ipo	612
StartupOB	SI_Startup	613
ProgrammingErrorOB IOaccessErrorOB	SI_ProgIOAccessError	614

SDT del parámetro START_UP_SI

La tabla siguiente muestra el significado de los SDT posibles del parámetro START_UP_SI:

Tipos de datos de sistema (SDT)	Números de los tipos de datos de sistema
SI_classic	592
SI_none	593
SI_Startup	613

Estructuras

Las siguientes tablas muestran el significado de los elementos estructurales de las diferentes estructuras:

Tabla 9-61 Estructura SI_classic

Elemento de estructura	Tipo de datos	Descripción
EV_CLASS	BYTE	• Bits de 0 a 3: ID de evento • Bits de 4 a 7: Clase de evento
EV_NUM	BYTE	Número de evento
PRIORITY	BYTE	Número del nivel de ejecución (Significado de B#16#FE: El OB no existe, está bloqueado o no puede arrancarse en el actual estado operativo.)
NUM	BYTE	Número de OB
TYP2_3	BYTE	ID de datos 2_3: Identifica la información registrada en ZI2_3
TYP1	BYTE	ID de datos 1: Identifica la información registrada en ZI1
ZI1	WORD	Información adicional 1
ZI2_3	DWORD	Información adicional 2_3

Tabla 9-62 Estructura SI_none

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)

Tabla 9-63 Estructura SI_ProgramCycle

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 1	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Initial_Call	BOOL	Para OB_Class = 1, 30, 52, 61, 65
Remanence	BOOL	Para OB_Class = 1

Tabla 9-64 Estructura SI_TimeOfDay

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 10	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
CaughtUp	BOOL	Para OB_Class = 10
SecondTime	BOOL	Para OB_Class = 10

Tabla 9-65 Estructura SI_Delay

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 20	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Sign	WORD	Para OB_Class = 20

Tabla 9-66 Estructura SI_Cyclic

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 30	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Initial_Call	BOOL	Para OB_Class = 1, 30, 52, 61, 65
Event_Count	INT	Para OB_Class = 30, 51, 52, 61, 65, 91, 92

Tabla 9-67 Estructura SI_HWInterrupt

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 40	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
LADDR	HW_IO	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
USI	WORD	Para OB_Class = 40
IChannel	USINT	Para OB_Class = 40
EventType	BYTE	Para OB_Class = 40

Tabla 9-68 Estructura SI_Submodule

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
LADDR	HW_IO	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Slot	UINT	Para OB_Class = 55, 56, 57
Specifier	WORD	Para OB_Class = 55, 56, 57

Tabla 9-69 Estructura SI_SynchCycle

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 61	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Initial_Call	BOOL	Para OB_Class = 1, 30, 52, 61, 65
PIP_Input	BOOL	Para OB_Class = 61, 91, 92
PIP_Output	BOOL	Para OB_Class = 61, 91, 92
IO_System	USINT	Para OB_Class = 61, 91, 92
Event_Count	INT	Para OB_Class = 30, 51, 52, 61, 65, 91, 92
SyncCycleTime	LTIME	Tiempo de ciclo calculado

Tabla 9-70 Estructura SI_IORedundancyError

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 70	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
LADDR	HW_ANY	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Para OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Para OB_Class = 70, 80, 83, 85, 86

Tabla 9-71 Estructura SI_CPURedundancyError

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 72	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Switch_Over	BOOL	Para OB_Class = 72

Tabla 9-72 Estructura SI_TimeError

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 80	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Fault_ID	BYTE	Para OB_Class = 70, 80, 83, 85, 86
Csg_OBnr	OB_ANY	Para OB_Class = 80
Csg_Prio	UINT	Para OB_Class = 80

Tabla 9-73 Estructura SI_DiagnosticInterrupt

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 82	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
IO_State	WORD	Para OB_Class = 82
LADDR	HW_ANY	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Channel	UINT	Para OB_Class = 82
MultiError	BOOL	Para OB_Class = 82

Tabla 9-74 Estructura SI_PlugPullModule

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 83	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
LADDR	HW_IO	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Para OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Para OB_Class = 70, 80, 83, 85, 86

Tabla 9-75 Estructura SI_AccessError

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 85	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
LADDR	HW_IO	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Para OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Para OB_Class = 70, 80, 83, 85, 86
IO_Addr	UINT	Para OB_Class = 85
IO_LEN	UINT	Para OB_Class = 85

Tabla 9-76 Estructura SI_StationFailure

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 86	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
LADDR	HW_IO	Para OB_Class = 40, 51, 55, 56, 57, 70, 82, 83, 85, 86, 91, 92
Event_Class	BYTE	Para OB_Class = 70, 83, 85, 86
Fault_ID	BYTE	Para OB_Class = 70, 80, 83, 85, 86

Tabla 9-77 Estructura SI_Servo

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 91	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Initial_Call	BOOL	Para OB_Class = 1, 30, 52, 61, 65
PIP_Input	BOOL	Para OB_Class = 61, 91, 92
PIP_Output	BOOL	Para OB_Class = 61, 91, 92
IO_System	USINT	Para OB_Class = 61, 91, 92
Event_Count	INT	Para OB_Class = 30, 51, 52, 61, 65, 91, 92
Synchronous	BOOL	

Tabla 9-78 Estructura SI_Ipo

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 92	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
Initial_Call	BOOL	Para OB_Class = 1, 30, 52, 61, 65
PIP_Input	BOOL	Para OB_Class = 61, 91, 92
PIP_Output	BOOL	Para OB_Class = 61, 91, 92
IO_System	USINT	Para OB_Class = 61, 91, 92
Event_Count	INT	Para OB_Class = 30, 51, 52, 61, 65, 91, 92
Reduction	UINT	Para OB_Class = 92

Tabla 9-79 Estructura SI_Startup

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT := 100	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)

Elemento de estructura	Tipo de datos	Descripción
LostRetentive	BOOL	Para OB_Class = 100
LostRTC	BOOL	Para OB_Class = 100

Tabla 9-80 Estructura SI_ProgIOAcessError

Elemento de estructura	Tipo de datos	Descripción
SI_Format	USINT	• 16#FF = Sin indicaciones • 16#FE = Información de arranque optimizada
OB_Class	USINT	Clase de OB para "Sin indicaciones" o "Información de arranque optimizada"
OB_Nr	UINT	N.º de OB (1 ... 32767)
BlockNr	UINT	Para OB_Class = 121, 122
Reaction	USINT	Para OB_Class = 121, 122
Fault_ID	BYTE	Para OB_Class = 121, 122
BlockType	USINT	Para OB_Class = 121, 122
Area	USINT	Para OB_Class = 121, 122
DBNr	DB_ANY	Para OB_Class = 121, 122
Csg_OBNr	OB_ANY	Para OB_Class = 121, 122
Csg_Prio	USINT	Para OB_Class = 121, 122
Width	USINT	Para OB_Class = 121, 122

Nota

El contenido de los elementos especificados en la estructura SI_classic es idéntico al de las variables temporales de un OB, siempre y cuando este se haya creado con la propiedad de bloque "Estándar".

Sin embargo, debe tenerse en cuenta que, en los distintos OBs, las variables temporales pueden tener otros nombres y tipos de datos. También debe tenerse en cuenta que la interfaz de llamada de los OBs contiene además la fecha y hora en que se solicitó el OB.

Los bits de 4 a 7 del elemento estructural EV_CLASS contienen la clase de evento. Aquí son posibles los siguientes valores:

- 1: Eventos de arranque de OBs estándar
- 2: Eventos de arranque de OBs de errores síncronos
- 3: Eventos de arranque de OBs de errores asíncronos

El elemento estructural PRIORITY devuelve la clase de prioridad correspondiente al OB actual.

Además de estos dos elementos, también es importante NUM . NUM contiene el número del OB actual o del último OB de arranque iniciado.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
8080	La información de arranque del OB actual no concuerda con el tipo de datos personalizado especificado
8081	La información de arranque del OB actual no concuerda con el tipo de datos de sistema especificado
8082	La información de arranque del último OB de arranque iniciado no concuerda con el tipo de datos personalizado especificado
8083	La información de arranque del último OB de arranque iniciado no concuerda con el tipo de datos de sistema especificado

Ejemplo

Supongamos que el último OB llamado, que todavía no se ha procesado por completo, es el OB 80, y el último OB de arranque llamado es el OB 100.

La tabla siguiente muestra la correspondencia entre los elementos estructurales del parámetro TOP_SI de la instrucción "RD_SINFO" y las correspondientes variables locales del OB 80.

TOP_SI Elemento de estructura	Tipo de datos	OB 80 - Variable local correspondiente	Tipo de datos
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

La tabla siguiente muestra la correspondencia entre los elementos estructurales del parámetro START_UP_SI de la instrucción "RD_SINFO" y las correspondientes variables locales del OB 100.

START_UP_SI Elemento de estructura	Tipo de datos	OB 100 - Variable local	Tipo de datos
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE

START_UP_SI Elemento de estructura	Tipo de datos	OB 100 - Variable local	Tipo de datos
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

RT_INFO: Leer estadísticas del tiempo de ejecución

Descripción

Con la instrucción "RT_INFO" se generan estadísticas del tiempo de ejecución de determinados bloques de organización, de la comunicación o del programa de usuario.

La información que debe devolverse se selecciona en el parámetro MODE:

- MODE 1 a 3 devuelven datos relativos al tiempo de ejecución de un bloque de organización determinado, cuyo número se ha indicado previamente en el parámetro OB.
- MODE 10 y 11 devuelven información sobre qué porcentajes del tiempo de ejecución corresponden a la comunicación y al programa de usuario.
- MODE 30 a 32 devuelven datos sobre ajustes configurados del programa de usuario.

En la transición de la CPU de arranque a RUN se vuelven a iniciar todas las mediciones.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "RT_INFO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	UINT	I, Q, M, D, L o constante	Con el parámetro MODE se selecciona la información que debe leerse (consulte la tabla "Parámetro MODE").
OB	Input	OB_ANY	I, Q, M, D, L o constante	Con el parámetro OB se elige el OB cuya información se desea leer.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error (véase "Parámetro RET_VAL")
INFO	InOut	VARIANT	I, Q, M, D, L	Puntero al área en la que deben escribirse los datos leídos. El tipo de datos necesario para INFO depende del parámetro MODE (véase la tabla "Parámetro MODE").

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

La tabla siguiente muestra la información que se devuelve para los valores del parámetro MODE.

MODE (decimal)	Descripción	Nota	Valor en el parámetro OB	Tipo de datos de INFO	Disponible a partir de la versión de CPU
1	Tiempo de ejecución de un OB determinado	La indicación del tiempo de ejecución se refiere siempre al OB indicado. No se registran procesos tales como las interrupciones por OB de mayor prioridad o por comunicación. Para medir el tiempo de ejecución de todo el programa se puede utilizar la instrucción "RUNTIME".	Número de OB	LTIME	S7-1500 V1.5
2	Tiempo de ejecución máximo de un OB determinado		Número de OB	LTIME	S7-1500 V1.5
3	Tiempo de ejecución mínimo de un OB determinado		Número de OB	LTIME	S7-1500 V1.5
10	Porcentaje de los OB de mayor prioridad en el tiempo de ejecución total Se indica el tiempo de ejecución de todos los OB utilizados en el programa de usuario con mayor prioridad que los OB de programa cíclicos (ProgramCycle). Generalmente se incluyen entre ellos todos los tipos de OB, exceptuando los OB de arranque. Los OB que pueden utilizarse y la prioridad que estos tienen dependen de la CPU y se describen en los capítulos dedicados a los principios básicos de la programación.	Los porcentajes indicados para MODE 10, 11 y 12 siempre suman 100 (= tiempo de ejecución total).	No relevante	UINT	S7-1500 V1.5
11	Porcentaje de la comunicación en el tiempo de ejecución total Se indica qué porcentaje tienen los procesos de comunicación en el tiempo de ejecución total del programa de usuario.		No relevante	UINT	S7-1500 V1.5

MODE (decimal)	Descripción	Nota	Valor en el parámetro OB	Tipo de datos de INFO	Disponible a partir de la versión de CPU
30	Se indica el límite de advertencia a partir del cual se llama un OB de error de tiempo. El límite de advertencia es el número máximo permitido de solicitudes realizadas pero no ejecutadas de un OB.		No relevante	LTIME	S7-1500 V1.5
31	Se indica el tiempo de ciclo mínimo configurado para el programa de usuario		No relevante	UINT	S7-1500 V1.5
32	Se indica la carga por comunicación máxima configurada		No relevante	UINT	S7-1500 V1.5

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0	Sin errores
8080	La selección para el parámetro MODE no se soporta.
8081	El bloque de organización seleccionado en el parámetro OB no existe en el programa de usuario.
80C3	Recursos insuficientes. Intente llamar la instrucción más tarde.
8452	Tipo de datos incorrecto en el parámetro INFO. Compruebe si se ha seleccionado el tipo de datos correcto en función del parámetro MODE.

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

LED: Leer estado del LED

Descripción

La instrucción "LED" lee el estado (p. ej. "On" u "Off") de un determinado LED de módulo.

- El parámetro LADDR direcciona la CPU o la interfaz.
- El parámetro LED selecciona el LED del módulo cuyo estado actual se va a leer con la instrucción.
- El parámetro RET_VAL indica el estado del LED seleccionado al llamar la instrucción. Dependiendo del LED seleccionado, solo puede mostrarse cierta información de estado, por ejemplo, algunos LED solo tienen un color. La información sobre qué estado es posible para un determinado LED se especifica en la documentación de hardware del módulo correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LED":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IO	I, Q, M, L o constante	Número de identificación de la CPU o de la interfaz. El número se asigna automáticamente y se guarda en las propiedades de la CPU o en la interfaz en la configuración de hardware.
LED	Input	UINT	I, Q, M, D, L o constante	Número de identificación del LED: • 1: STOP/RUN • 2: ERROR • 3: MAINT (Mantenimiento) • 4: Redundante • 5: Link (verde) • 6: Rx/Tx (amarillo)
RET_VAL	Return	INT	I, Q, M, D, L	Estado del LED

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

RET_VAL	Descripción
0 a 9	Estado del LED: • 0 = LED no existe • 1 = apagado permanentemente • 2 = color 1 (p. ej., en LED STOP/RUN: verde) encendido permanentemente • 3 = color 2 (p. ej., en LED STOP/RUN: naranja) encendido permanentemente • 4 = color 1 parpadea con 2 Hz • 5 = color 2 parpadea con 2 Hz • 6 = color 1 y 2 parpadean alternativamente con 2 Hz • 7 = LED activo, color 1 • 8 = LED activo, color 2 • 9 = LED existe, sin embargo, la información de estado no está disponible
8091	El módulo direccionado con el parámetro LADDR no existe.
8092	Con el parámetro LADDR se ha direccionado un módulo que no soporta LEDs
8093	El número de identificación indicado en el parámetro LED no está definido.
80Bx	La CPU indicada en el parámetro LADDR no soporta la instrucción "LED".

Get_IM_Data: Leer datos de identificación y mantenimiento

Descripción

La instrucción "Get_IM_Data" lee los datos de identificación y mantenimiento (datos I&M) de un dispositivo. Con el parámetro LADDR se selecciona el identificador de hardware del dispositivo cuyos datos I&M desean leerse.

Con el parámetro IM_TYPE se definen los datos que deben leerse mediante la instrucción:

- IM_TYPE = 0: datos I&M 0
  Los datos I&M 0 ofrecen información básica de un dispositivo, como la identificación de fabricante, la referencia, el número de serie o la versión de hardware y firmware. Los datos I&M 0 solo permiten un acceso de lectura. Esta información también se muestra en el TIA Portal en la vista "Online y diagnóstico" de un dispositivo.
- IM_TYPE = 11: datos I&M 1 de los datos de parametrización de la CPU
  Los datos I&M 1 contienen una descripción del funcionamiento del dispositivo así como el identificador de situación, es decir cómo se ha designado o identificado el dispositivo dentro de la instalación.
- IM_TYPE = 12: datos I&M 2 de los datos de parametrización de la CPU
  Los datos I&M 2 contienen la fecha de incorporación, es decir, la fecha en que se instaló el dispositivo en la instalación.
- IM_TYPE = 13: datos I&M 3 de los datos de parametrización de la CPU
  Los datos I&M 3 contienen información adicional sobre el dispositivo instalado. La información adicional es un texto cualquiera que puede introducirse a discreción.

Los datos I&M leídos se escriben en el área direccionada en el parámetro DATA.

El estado de procesamiento de la petición de lectura se indica a través de los parámetros de salida BUSY, DONE, ERROR y los dos bytes centrales del parámetro de salida STATUS.

Definición: datos de identificación y mantenimiento (I&M)

Los datos de identificación y mantenimiento (I&M) son datos almacenados en un módulo que ayudan a comprobar la configuración del sistema, a buscar modificaciones en el hardware de una instalación o a solucionar errores.

- Los datos de identificación (datos I) son datos invariables de un dispositivo a los que solo puede accederse para su lectura.
- Los datos de mantenimiento (datos M) son informaciones dependientes del sistema como, p. ej. lugar y fecha de montaje. Los datos de mantenimiento se generan durante la configuración y se escriben en el módulo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Get_IM_Data":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IO	I, Q, M, D, L o constante	Identificador de hardware del dispositivo. El número se asigna automáticamente y se guarda en las propiedades del dispositivo, en la configuración hardware.
IM_TYPE	Input	UINT	I, Q, M, D, L o constante	Número de datos de identificación y mantenimiento Valores posibles: • 0: Datos I&M 0 • 11: datos I&M 1 de los datos de parametrización de la CPU • 12: datos I&M 2 de los datos de parametrización de la CPU • 13: datos I&M 3 de los datos de parametrización de la CPU
DATA	InOut	VARIANT	I, Q, M, D, L	Área para el almacenamiento de los datos de identificación y mantenimiento leídos (véase el parámetro DATA).
DONE	Output	BOOL	I, Q, M, D, L	La instrucción se ha ejecutado correctamente. Los datos I&M se han transferido al parámetro DATA.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ejecución de la instrucción finalizada o no iniciada. • 1: Ejecución de la instrucción no finalizada.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ningún error. • 1: se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para la visualización debe copiarse el valor de STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro DATA

Para el almacenamiento de los datos I&M puede utilizarse una cadena de caracteres (STRING), un array (ARRAY of CHAR/BYTE) o una estructura de datos (STRUCT):

- Si en el parámetro DATA se direcciona una cadena de caracteres (tipo de datos STRING), la longitud de la cadena de caracteres se ajusta automáticamente a la longitud de los datos I&M leídos (254 caracteres como máximo).
- Si en el parámetro DATA se direcciona una estructura de datos (ARRAY of CHAR/BYTE o STRUCT), los datos I&M leídos se escriben en los diferentes componentes del tipo de datos utilizado. Si la estructura de datos direccionada es más larga que los datos leídos, se asigna el valor cero a los componentes restantes.
- Si en el parámetro DATA se crea una estructura de datos del tipo STRUCT, debe utilizarse un bloque de datos sin acceso a bloque optimizado (ver categoría "Atributos" de las propiedades del bloque).

Si en el parámetro DATA se utiliza un tipo de datos diferente a STRING, ARRAY of BYTE/CHAR o STRUCT, se emite el código de error 8093 en el parámetro STATUS. Una excepción son los datos I&M 0 (parámetro IM_TYPE = 0). Para los datos I&M 0 también puede utilizarse la estructura "IuM0_data" en el parámetro DATA:

Parámetro		Tipo de datos	Bytes	Descripción
Manufacturer_ID		UINT	2	Identificación de fabricante (p. ej., "42" para SIEMENS)
Order_ID		CHAR[20]	20	Referencia
Serial_Number		CHAR[16]	16	Número de serie
Hardware_Revision		UNIT	2	Versión de hardware
Software_Revision		STRUCT	4	Versión de firmware
	Type	CHAR	1	-
	Functional	UNIT8	1	-
	Bugfix	UNIT8	1	-
	Internal	UINT8	1	-
Revision_Counter		UINT	2	Contador de revisiones

Parámetro	Tipo de datos	Bytes	Descripción
Profile_ID	UNIT	2	Perfil
Profile_Specific_Type	UNIT	2	Clase de dispositivo
IM_Version	UNIT	2	Versión I&M
I&M_Supported	UNIT	2	Datos I&M soportados por el dispositivo (I&M 0-I&M 4)

Parámetro STATUS

Código de error* (W#16#...)	Descripción
0	Ningún error
7000	Ninguna petición en proceso.
7001	Primera llamada de la instrucción asíncrona "Get_IM_Data". La ejecución de la instrucción aún no ha finalizado (BUSY = 1, DONE = 0).
7002	Una llamada más de la instrucción asíncrona "Get_IM_Data". La ejecución de la instrucción aún no ha finalizado (BUSY = 1, DONE = 0).
807F	Se ha superado el máximo de 10 ejecuciones paralelas de las instancias: • Evite más de 10 llamadas paralelas. • Repita la llamada en otro momento ya que se trata de un error temporal.
8091	No existe el dispositivo direccionado en el parámetro LADDR.
8092	LADDR direcciona un dispositivo que no soporta la emisión de datos I&M.
8093	El tipo de datos del parámetro DATA no está soportado.
80B1	La CPU utilizada no soporta la instrucción "Get_IM_Data".
80B2	El valor del parámetro IM_TYPE no es válido o bien la CPU o el dispositivo direccionado no soporta el IM_TYPE seleccionado.
8752	El área de memoria indicada en el parámetro DATA es demasiado pequeña para guardar todos los datos I&M. Los datos I&M leídos se guardan solo hasta la longitud máxima del área de memoria indicada.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

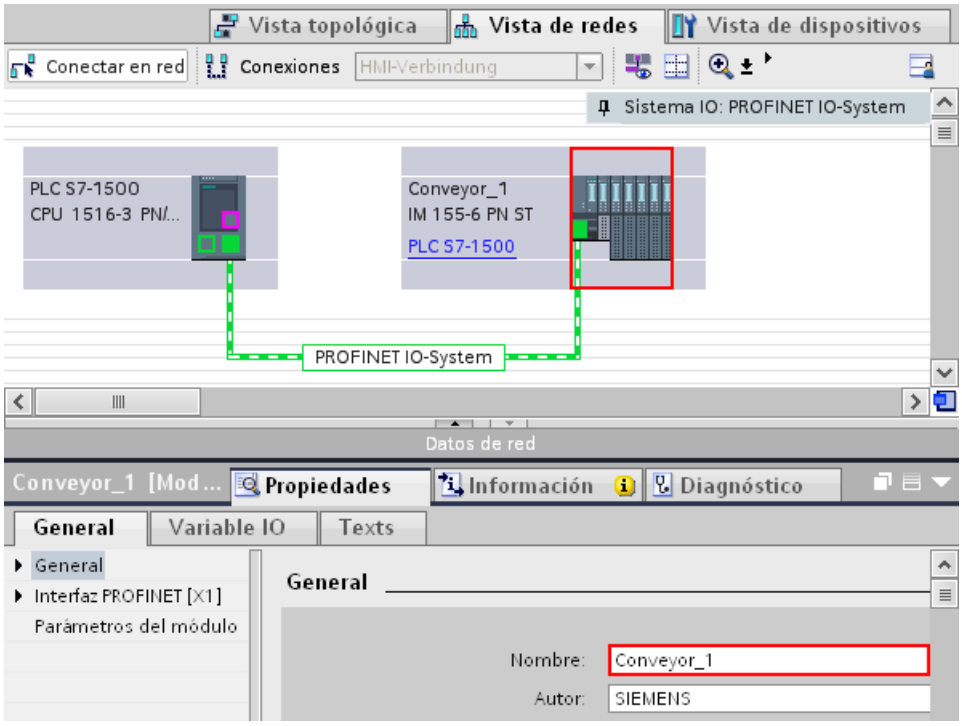
Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_NAME: leer el nombre de un módulo

Descripción

La instrucción "GET_NAME" lee el nombre de un dispositivo IO. El nombre se indica en la vista de redes y en las propiedades del dispositivo IO:



La selección del dispositivo IO se realiza a través de la ID de hardware del sistema PROFINET IO (en el parámetro LADDR) y del número de dispositivo del dispositivo IO (parámetro STATION_NR).

Una vez ejecutada la instrucción correctamente, el nombre del dispositivo IO se escribe en el área direccionada por el parámetro DATA.

El nombre leído varía en función del tipo de dispositivo IO:

- En un esclavo DP o un dispositivo IO se visualiza el nombre del módulo de cabecera.
- En un esclavo I o un I-Device se visualiza el nombre del módulo de interfaz.
- En un HMI Panel se visualiza el nombre de la estación.
- En una estación PC se visualiza el nombre del módulo de interfaz.
- En los dispositivos GSD se visualiza el nombre del Device Access Point (DAP) (nombre de la interfaz o del módulo de cabecera).

La longitud del nombre se indica en el parámetro LEN. Si el nombre supera la longitud del área indicada en el parámetro DATA, el nombre se escribe solo hasta la longitud máxima del área direccionada.

La longitud máxima del nombre es de 128 caracteres.

Nota

Lectura del nombre de la CPU (a partir de la versión 1.1)

Si aplica un "0" en los parámetros LADDR y STATION_NR, respectivamente, la instrucción muestra el nombre de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GET_NAME":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IOSYSTEM	I, Q, M, D, L o constante	ID de hardware del sistema PROFINET IO. El número puede aplicarse a partir de las constantes de sistema o las propiedades del sistema IO.
STATION_NR	Input	UINT	I, Q, M, D, L o constante	Número de dispositivo del dispositivo IO. El número de dispositivo puede aplicarse en la vista de redes tomándolo de las propiedades del dispositivo IO en "Direcciones Ethernet".
DATA	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el área en el que se escribe el nombre del módulo.
DONE	Output	BOOL	I, Q, M, D, L	La instrucción se ha ejecutado correctamente. Transferir el nombre del módulo al área del parámetro DATA.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ejecución de la instrucción finalizada. • 1: Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Sin error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
LEN	Output	DINT	I, Q, M, D, L	Longitud del nombre del dispositivo IO (número de caracteres).
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

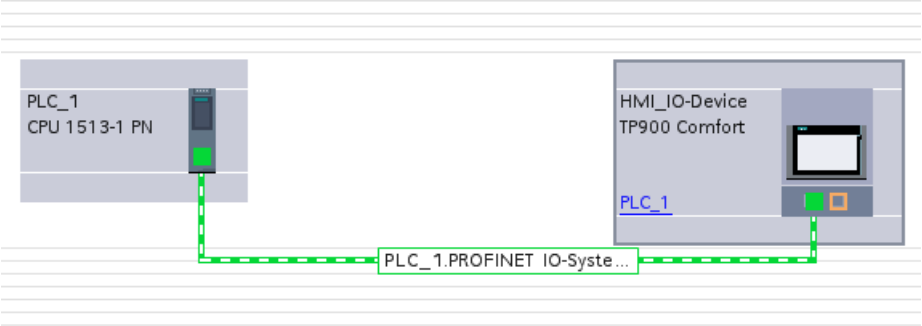
Código de error* (W#16#...)	Explicación
0	Sin error
7000	Ninguna petición en proceso
7001	Primera llamada de la instrucción asíncrona "GET_NAME". La ejecución de la instrucción aún no ha finalizado (BUSY = 1, DONE = 0).
7002	Una llamada más de la instrucción asíncrona "GET_NAME". La ejecución de la instrucción aún no ha finalizado (BUSY = 1, DONE = 0).
8090	El parámetro de entrada LADDR contiene un valor no permitido. Causas posibles: • El número indicado no es ninguna ID de hardware de un sistema PROFINET IO. • No se ha indicado ningún número en el parámetro LADDR .
8092	El valor del parámetro LADDR no direcciona ningún sistema PROFINET-IO.
8093	La instrucción no soporta el tipo de datos del parámetro DATA.
8095	El número de dispositivo (parámetro STATION_NR) no existe en el sistema PROFINET IO seleccionado.
80B1	La CPU utilizada no soporta la instrucción.
80C3	Error temporal de recursos: en este momento la CPU está procesando el máximo posible de llamadas de bloque simultáneas. No se puede ejecutar "GET_NAME" antes de haber finalizado como mínimo una llamada de bloque.
8852	El área indicada en el parámetro DATA es demasiado corta para el nombre completo del dispositivo IO. El nombre solo se escribirá hasta la longitud máxima posible. Para poder leer el nombre completo, utilice un área de datos más larga en el parámetro DATA. Para ello, el área debe abarcar como mínimo el número de caracteres del parámetro LEN.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El ejemplo siguiente muestra cómo se lee el nombre de estación de un HMI Panel.

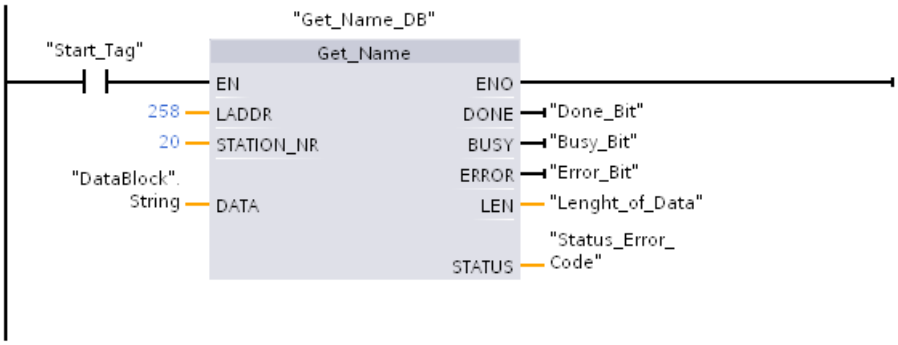
Configuración del HMI Panel:

- El HMI Panel con el nombre de estación "HMI_IO-Device" se ha creado en la vista de redes y está asignado al mismo sistema PROFINET IO que la CPU.
- Para el HMI Panel se ha activado el modo de operación "Dispositivo IO" en las propiedades de la configuración hardware y la CPU se ha asignado como controlador IO.
- En "Direcciones Ethernet" de las propiedades se ha asignado el número de dispositivo "20".



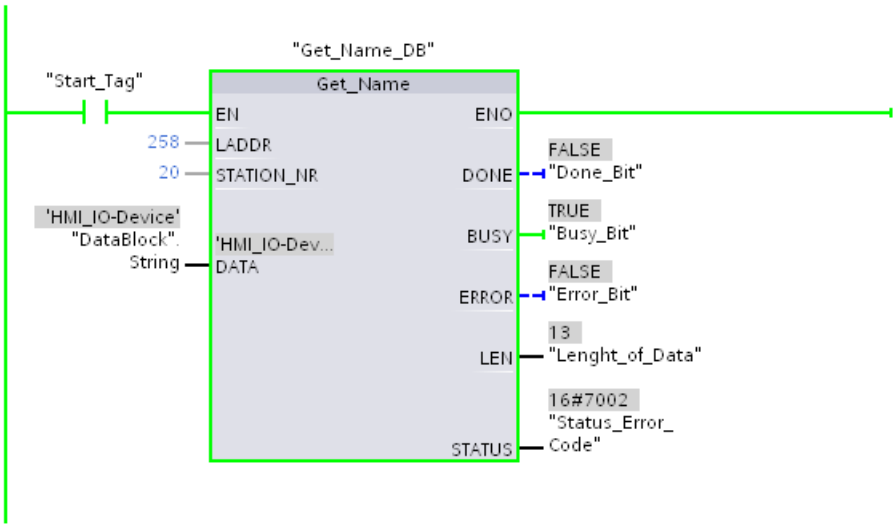
Parametrización de la instrucción "GET_NAME":

- En el parámetro LADDR se ha especificado la identificación de hardware del sistema IO (258).
- En el parámetro STATION_NR se ha especificado el número de dispositivo del HMI Panel (20).
- En el parámetro DATA se ha interconectado una variable con el tipo de datos STRING de un bloque datos.
- Para los parámetros de salida de la instrucción se han definido variables PLC (área de memoria de marcas).



Ejecución de la instrucción:

- Una vez ejecutada la instrucción, el nombre de estación del HMI Panel (HMI_IO-Device) se escribe en el parámetro DATA del bloque de datos.
- En el parámetro LEN se visualiza el número de caracteres del nombre (13).



Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GetStationInfo: leer información de un dispositivo IO

Descripción

La instrucción "GetStationInfo" permite leer información de un dispositivo PROFINET IO. La instrucción también permite leer información de un dispositivo IO que se encuentre en un sistema IO subordinado (conexión con CP/CM).

El dispositivo IO se direcciona mediante el identificador de hardware de la estación en el parámetro LADDR. El identificador de hardware se muestra en la vista "Dispositivos y redes", en las propiedades de la estación.

Con el parámetro MODE se selecciona la información que debe leerse. La instrucción soporta en TIA V12 SP1 la lectura de los parámetros de dirección IP según IPv4 (MODE = 1).

En el parámetro DATA se indica el área de datos en la que deben escribirse los datos de dirección leídos. Utilice la estructura "IF_CONF_v4" para el almacenamiento de la dirección IP.

Mediante el parámetro de control REQ se activa la lectura de los datos de dirección. Para ello debe poder accederse al dispositivo IO.

El estado de procesamiento de la petición de lectura se indica a través de los parámetros de salida BUSY, DONE, ERROR y STATUS.

Nota

Direccione el dispositivo IO exclusivamente con el identificador de hardware de la estación

La estación, el dispositivo IO y la interfaz PROFINET tienen cada uno un identificador de hardware propio. Utilice exclusivamente el identificador de hardware de la estación para la instrucción "GetStationInfo".

Si mediante el parámetro LADDR se direcciona una interfaz PROFINET, p. ej., los datos de dirección no se leen y se emite el código de error 8092.

Para leer los datos de dirección de una interfaz PROFINET integrada o de un CM/CP con la configuración centralizada, utilice la instrucción "RDREC".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GetStationInfo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control Request Activa la lectura de la información cuando REQ = "1".
LADDR	Input	HW_DEVICE	I, Q, M, D, L o constante	Identificador de hardware de la estación del dispositivo IO El número puede aplicarse en la vista de redes adoptándose de las propiedades de la estación o de la ficha "Constantes del sistema" de la tabla de variables estándar.
DETAIL	Input	HW_SUBMODULE	I, Q, M, D, L o constante	El parámetro DETAIL no se utiliza. No asigne ningún valor al parámetro.
MODE	Input	UNIT	I, Q, M, D, L o constante	Selección de los datos de dirección que se leen Asigne al parámetro MODE el valor "1" (lectura de los parámetros de dirección según IPv4).
DATA	InOut	VARIANT	D, L	Puntero hacia el área en la que se escriben los datos de dirección del dispositivo IO. Si MODE = 1, utilice la estructura "IF_CONF_v4".
DONE	Output	BOOL	I, Q, M, D, L	La instrucción se ha ejecutado correctamente. Los datos de dirección se han transferido al parámetro DATA.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ejecución de la instrucción finalizada. • 1: Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ningún error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro DATA

Utilice en el parámetro DATA la estructura "IF_CONF_v4" para el almacenamiento de los parámetros de dirección según IPv4.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	Id	UINT	30	ID de la estructura "IF_CONF_v4".
2 … 3	Length	UNIT	18	Longitud de los datos leídos en BYTE.
4 ... 5	Mode	UNIT	0	No relevante para la instrucción "GetStationInfo" (se deja en "0").
6 ... 9	InterfaceAddress	ARRAY [1..4] of BYTE	-	Dirección IP del dispositivo IO en formato IP_V4, p. ej. para 192.168.3.10: • addr[1] = 192 • addr[2] = 168 • addr[3] = 3 • addr[4] = 10

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
10 … 13	SubnetMask	ARRAY [1..4] of BYTE	-	Máscara de subred del dispositivo IO en formato IP_V4, p. ej. para 255.255.255.0: • addr[1] = 255 • addr[2] = 255 • addr[3] = 255 • addr[4] = 0
14 ... 17	DefaultRouter	ARRAY [1..4] of BYTE	-	Dirección IP del router en formato IP_V4, p. ej. para 192.168.3.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 3 • addr[4] = 1

Parámetro STATUS

Código de error* (W#16#...)	Explicación
0	Ningún error
7000	Ninguna petición en proceso.
7001	Primera llamada de la instrucción asíncrona "GetStationInfo". La ejecución de la instrucción aún no ha finalizado (BUSY = 1, DONE = 0).
7002	Una llamada más de la instrucción asíncrona "GetStationInfo". La ejecución de la instrucción aún no ha finalizado (BUSY = 1, DONE = 0).
8080	El valor del parámetro MODE no está soportado.
8090	La ID de hardware indicada en el parámetro LADDR no está configurada.
8092	El parámetro LADDR no direcciona ningún dispositivo PROFINET IO.
8093	Tipo de datos no válido en el parámetro DATA.
80A0	No puede leerse la información solicitada.
80C0	No puede accederse al dispositivo IO direccionado.
80C3	Se ha alcanzado el número máximo de llamadas simultáneas de la instrucción "GetStationInfo" (10 instancias).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

DeviceStates: Leer la información de estado de los módulos en un sistema IO

Descripción

Con la instrucción "DeviceStates" se consulta una determinada información de estado de todos los módulos de un sistema IO, es decir:

- o bien de todos los dispositivos IO de un sistema PROFINET IO,
- o bien de todos los esclavos DP de un sistema maestro DP

Se indica en forma de valor booleano en qué módulos se cumple el estado seleccionado. Por ejemplo, se puede leer qué dispositivos IO están desactivados en un sistema PROFINET IO.

Además, se indica si la información de estado que se va a leer se cumple en al menos uno de los dispositivos IO o esclavos DP.

La instrucción puede llamarse tanto en el OB cíclico como en el OB de alarma (p. ej. OB82 - alarma de diagnóstico).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DeviceStates":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IOSYSTEM	I, Q, M, L o constante	ID de hardware del sistema PROFINET IO o maestro DP (ver descripción abajo)
MODE	Input	UINT	I, Q, M, D, L o constante	Selección de la información de estado que se va a leer (ver descripción abajo)
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción (ver descripción abajo)
STATE	InOut	VARIANT	I, Q, M, D, L	Búfer para el estado del dispositivo IO o del esclavo DP (ver descripción abajo)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro LADDR

El sistema PROFINET IO o maestro DP se selecciona en el parámetro LADDR mediante la ID de hardware.

La ID de hardware se puede consultar:

- o bien en las propiedades del sistema PROFINET IO o maestro DP en la vista de redes,
- o bien en la tabla de variables PLC, en la lista de constantes del sistema con el tipo de datos HW_IOSYSTEM.

Parámetro MODE

Seleccione la información de estado que se va a leer con el parámetro MODE. Se puede leer una de las siguientes informaciones de estado para todo el sistema PROFINET IO o maestro DP:

- 1: Los dispositivos IO/esclavos DP están configurados
- 2: Los dispositivos IO/esclavos DP fallan
- 3: Los dispositivos IO/esclavos DP están desactivados
- 4: Los dispositivos IO/esclavos DP están disponibles
- 5: Los dispositivos IO/esclavos DP presentan problemas. Por ejemplo:
  - Mantenimiento necesario o recomendado
  - No accesible
  - No disponible
  - Con errores

Parámetro STATE

El parámetro STATE emite el estado del dispositivo IO/esclavo DP seleccionado con el parámetro MODE.

Si el estado seleccionado con MODE se cumple en un dispositivo IO/esclavo DP, los siguientes bits se ponen a "1" en el parámetro STATE.

- Bit 0 = 1: Indicador de grupo. El bit n de al menos un dispositivo IO/esclavo DP se ha puesto en "1".
- Bit n = 1: El estado seleccionado mediante MODE se aplica al dispositivo IO/esclavo DP.
  - En un sistema PROFINET IO el bit n equivale al número de dispositivo del dispositivo IO correspondiente (ver Propiedades de la interfaz PROFINET en la vista de dispositivos y redes).
  - En un sistema PROFIBUS DP el bit n equivale a la dirección PROFIBUS del esclavo DP (ver Propiedades del esclavo DP en la vista de dispositivos y redes).

Como tipo de datos se utiliza BOOL o Array of BOOL:

- Para emitir exclusivamente el bit del indicador de grupo de la información de estado, se puede utilizar el tipo de datos BOOL en el parámetro STATE.
- Para emitir información de estado de todos los dispositivos IO/esclavos DP, utilice Array of BOOL con la siguiente longitud:
  - Sistema PROFINET IO: 1024 bits
  - Sistema maestro DP: 128 bits

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0	Ningún error
8091	ID de hardware del parámetro LADDR no disponible. Compruebe (p. ej. en las constantes del sistema) si existe el valor de LADDR en el proyecto.
8092	LADDR no direcciona ningún sistema PROFINET IO o maestro DP.
8093	Tipo de datos no válido en el parámetro STATE.
80B1	La instrucción "DeviceStates" no está soportada por la CPU.
80B2	El parámetro seleccionado MODE no está soportado por la CPU utilizada para el sistema IO indicado en el parámetro LADDR.
8452	La información de estado completa no cabe en la variable configurada en el parámetro STATE.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo: lectura de las estaciones averiadas de un sistema maestro PROFINET IO.

Un sistema PROFINET IO contiene 4 dispositivos IO con los números de dispositivo 1, 2, 3 y 4. El dispositivo IO con el número 2 está averiado.

La instrucción "DeviceStates" se ejecuta para el sistema PROFINET IO con MODE = 2 (averiado/no averiado).

En el parámetro STATE se activan los siguientes bits:

- Bit 0 = 1: Existe una avería en al menos uno de los dispositivos IO.
- Bit 1 = 0: El dispositivo IO con el número de dispositivo 1 no está averiado.
- Bit 2 = 1: El dispositivo IO con el número de dispositivo 2 está averiado.
- Bit 3 = 0: El dispositivo IO con el número de dispositivo 3 no está averiado.
- Bit 4 = 0: El dispositivo IO con el número de dispositivo 4 no está averiado.
- Bit 5 = 0: Irrelevante
- Bit 6 = 0: Irrelevante
- ...

Ejemplo: lectura de las estaciones averiadas de un sistema maestro PROFIBUS DP

Un sistema maestro DP contiene 4 esclavos DP con las direcciones PROFIBUS 3, 4, 5 y 6. El esclavo DP con la dirección 4 está averiado.

La instrucción "DeviceStates" se ejecuta para el sistema maestro DP con MODE = 2 (averiado/no averiado).

En el parámetro STATE se activan los siguientes bits:

- Bit 0 = 1: Existe una avería en al menos uno de los esclavos DP.
- Bit 1 = 0: Irrelevante

- Bit 2 = 0: Irrelevante
- Bit 3 = 0: El esclavo DP con la dirección 3 no está averiado.
- Bit 4 = 1: El esclavo DP con la dirección 4 está averiado.
- Bit 5 = 0: El esclavo DP con la dirección 5 no está averiado.
- Bit 6 = 0: El esclavo DP con la dirección 6 no está averiado.
- Bit 7 = 0: Irrelevante
- Bit 8 = 0: Irrelevante
- ...

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

ModuleStates: Leer información de estado del submódulo de un módulo

Descripción

Con la instrucción "ModuleStates" se lee información de estado de los módulos de un dispositivo PROFINET IO o un esclavo PROFIBUS DP.

Se indica en forma de valor booleano en qué módulos se cumple el estado seleccionado. Por ejemplo, se puede leer qué módulos están desactivados en un dispositivo PROFINET IO.

Además, se indica si la información de estado que se va a leer se cumple en al menos uno de los módulos.

La instrucción puede llamarse tanto en el OB cíclico como en el OB de alarma (p. ej. OB82 - alarma de diagnóstico).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ModuleStates":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_DEVICE	I, Q, M, D, L o constante	ID de hardware de la estación (ver descripción abajo)
MODE	Input	UINT	I, Q, M, D, L o constante	Selección de la información de estado de los módulos que se va a leer (ver descripción abajo)
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción (ver descripción abajo)
STATE	InOut	VARIANT	I, Q, M, D, L	Búfer para el estado de los módulos (ver descripción abajo)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro LADDR

El dispositivo IO o el maestro DP se selecciona en el parámetro LADDR mediante la ID de hardware de la estación.

La ID de hardware se puede consultar:

- o bien en la vista de redes, en las propiedades de la estación del dispositivo IO o del esclavo DP,
- o bien en la tabla de variables PLC, en la lista de constantes del sistema con el tipo de datos HW_DEVICE (para un dispositivo IO) o con el tipo de datos HW_DPSLAVE (para un esclavo DP).

Parámetro MODE

Seleccione la información de estado que se va a leer con el parámetro MODE. Se puede leer en cada caso una de las siguientes informaciones de estado de los módulos:

- 1: Los módulos están configurados
- 2: Los módulos están averiados
- 3: Los módulos están desactivados
- 4: Los módulos están disponibles
- 5: Los módulos presentan problemas. Por ejemplo:
  - Mantenimiento necesario o recomendado
  - No accesible
  - No disponible
  - Con errores

Parámetro STATE

El parámetro STATE emite el estado seleccionado con el parámetro MODE de los módulos.

Si el estado seleccionado con MODE se cumple en un módulo, los siguientes bits se ponen a "1":

- Bit 0 = 1: Indicador de grupo. El bit n de al menos un módulo se ha puesto a "1".
- Bit n = 1: El estado seleccionado mediante MODE se cumple en el módulo del slot n-1 (ejemplo: bit 3 = slot 2).

Como tipo de datos se utiliza BOOL o Array of BOOL:

- Para emitir exclusivamente el bit del indicador de grupo de la información de estado, se puede utilizar el tipo de datos BOOL en el parámetro STATE.
- Para emitir la información de estado de todos los módulos, utilice Array of BOOL con una longitud de 128 bits.

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0	Ningún error
8091	ID de hardware del parámetro LADDR no disponible. Compruebe (p. ej. en las constantes del sistema) si existe el valor de LADDR en el proyecto.
8092	LADDR no direcciona ningún dispositivo IO o esclavo DP.
8093	Tipo de datos no válido en el parámetro STATE.
80B1	La instrucción "ModuleStates" no está soportada por la CPU.
80B2	El parámetro seleccionado MODE no está soportado por la CPU utilizada para el sistema IO o el esclavo DP indicado en el parámetro LADDR.
8452	La información de estado completa no cabe en la variable configurada en el parámetro STATE.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

Un dispositivo IO contiene 4 módulos en los slots 1 a 4. El módulo del slot 2 está averiado.

La instrucción "ModuleStates" se ejecuta para el dispositivo IO con MODE = 2 (averiado/no averiado).

En el parámetro STATE se activan los siguientes bits:

- Bit 0 = 1: Al menos uno de los módulos está averiado.
- Bit 1 = 0: Número de slot 0 (ocupado por dispositivo IO)
- Bit 2 = 0: El módulo del slot número 1 no está averiado.
- Bit 3 = 1: El módulo del slot número 2 está averiado.
- Bit 4 = 0: El módulo del slot número 3 no está averiado.
- Bit 5 = 0: El módulo del slot número 4 no está averiado.
- Bit 6 = 0: Irrelevante
- Bit 7 = 0: Irrelevante

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GEN_DIAG: Generar información de diagnóstico

Descripción

La instrucción "GEN_DIAG" genera información de diagnóstico sobre componentes de hardware de otros fabricantes para utilizarla en el diagnóstico del TIA Portal. Para poder usar

la instrucción, debe haberse instalado previamente el archivo GSD(GSDL/GSDML) suministrado por el fabricante.

La instrucción genera todos los eventos de diagnóstico (también para mantenimiento necesario).

- El parámetro LADDR permite seleccionar los componentes de hardware para los que se desea generar un evento de diagnóstico.
- El parámetro MODE permite especificar si el evento es entrante o saliente.
- El parámetro DiagEvent permite definir el evento de diagnóstico en la estructura DiagnosticDetail. La estructura se crea de forma automática en la interfaz local del bloque si se define una variable en el parámetro DiagEvent.

La información de diagnóstico se proporciona de modo síncrono. La transmisión de la información de diagnóstico y la emisión de avisos se producen de modo asíncrono.

ATENCIÓN
No se permiten mensajes de error específicos de seguridad (fail-safe) Si se define información de diagnóstico específica de seguridad en el parámetro DiagEvent, la instrucción lo comprueba y se emite el código de error 80A1.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GEN_DIAG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_ANY	I, Q, M, D, L o constante	Número de identificación del componente de hardware
MODE	Input	UINT	I, Q, M, D, L o constante	Selección de la información entrante/saliente: • 1: El evento de diagnóstico especificado es un evento entrante • 2: El evento de diagnóstico especificado es un evento saliente • 3: Todos los eventos de diagnóstico son salientes. Según esta información, no existe ningún fallo en el componente de hardware (símbolo de diagnóstico verde). El parámetro DiagEvent no se evalúa si MODE = 3.
DiagEvent	InOut	DiagnosticDetail	L	Especifica el evento de diagnóstico (véase "Parámetro DiagEvent").
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción/mensaje de error (véase "Parámetro RET_VAL").

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro DiagEvent

La estructura DiagnosticDetail es un tipo de datos de sistema para especificar el evento de diagnóstico con el siguiente formato:

Parámetro		Tipo de datos	Descripción
DiagnosticDetail		Struct	
	ChannelInfo	WORD	Propiedades del canal (0...7)
	ALID	UINT	ID local del aviso. Con la ID se identifica el aviso de forma unívoca.
	TextID	UINT	ID de un texto de aviso en una lista de textos.
	Channel Number	UINT	Número de canal específico del fabricante (0x0000 — 0x7FFF)
	Addval_0	DWORD	Comodines para información adicional. El valor/la lista de valores depende del error de conexión.
	TextID2	UINT	Textos para la reacción de la CPU (estado operativo, llamadas de OB, etc.).
	LADDR	HW_ANY	Idéntico al parámetro LADDR.
	TextListId	UINT	• 0: Sin lista de textos • ≠0: ID de la lista de textos
	Channel Direction	UINT	• 0000: Irrelevante • FFF1: Input • FFF2: Output • FFF3: Input/Output
	Addval_1	DWORD	Comodines para información adicional sobre el fallo de canal (en función del archivo GSD). En relación con los tipos de fallos de canal, véase también: norma IEC 61158 (PROFINET IO Type 10 and PROFIBUS DP Type 3).

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	Ningún error
8080	Valor del parámetro MODE no soportado.
8090	Número de identificación para el componente de hardware del parámetro LADDR no disponible.
8091	No es posible generar información de diagnóstico para el componente de hardware direccionado con el parámetro LADDR.
80A1	• Contenido de la estructura DiagnosticsDetail del parámetro DiagEvent no válida o incoherente. • Información de diagnóstico específica de seguridad (fail-safe) definida en el parámetro DiagEvent (no permitido).
80A4	Componente de hardware direccionado no accesible.
80C1	Recursos insuficientes para la ejecución paralela.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_DIAG: Leer información de diagnóstico

Descripción

La instrucción "GET_DIAG" permite leer la información de diagnóstico de un objeto de hardware. La selección del objeto de hardware se realiza con el parámetro LADDR. El parámetro MODE selecciona qué información de diagnóstico se va a leer.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GET_DIAG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	UINT	I, Q, M, D, L o constante	El parámetro MODE selecciona qué datos de diagnóstico se deben emitir.
LADDR	Input	HW_ANY (WORD)	I, Q, M, L o constante	Reconocimiento de hardware del dispositivo.
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción
CNT_DIAG	Output	UINT	I, Q, M, D, L	Número de detalles de diagnóstico emitidos
DIAG	InOut	VARIANT	I, Q, M, D, L	Consulte en la siguiente tabla la información de diagnóstico en función del modo seleccionado
DETAIL	InOut	VARIANT	I, Q, M, D, L	Detalles de diagnóstico correspondientes al modo seleccionado El parámetro está oculto (solo se utiliza con MODE = 3).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

En función del valor del parámetro MODE, se emiten datos de diagnóstico distintos a través de los parámetros de salida DIAG, CNT_DIAG y DETAIL.

MODE	Descripción	DIAG	CNT_DIAG	DETAIL
0	Salida de toda la información de diagnóstico soportada para un módulo como DWORD, en la que bit X=1 indica que el modo X se soporta.	Secuencia de bits de los modos soportados como DWORD, donde el bit X=1 indica que se soporta el modo X.	0	-
1	Salida del propio estado del objeto de hardware direccionado.	Estado de diagnóstico. Salida del estado propio del objeto de hardware direccionado conforme a la estructura DIS.	0	-
2	Salida del estado de todos los módulos subordinados del objeto de hardware direccionado.	Estado de diagnóstico. Salida conforme a la estructura DNN.	0	-
3 *	Salida del propio estado del objeto de hardware direccionado, así como información detallada sobre el diagnóstico de canal.	Estado de diagnóstico. Salida del estado propio del objeto de hardware direccionado conforme a la estructura DIS.	Cantidad de información de estado del módulo.	Información de estado del módulo conforme a la estructura DiagnosticDetail.
* No soportado por la CPU S7-1200.				

Estructura DIS

En el parámetro MODE = 1 o = 3 se emite la información de diagnóstico conforme a la estructura DIS. En tal caso indique como tipo de datos el tipo de datos del sistema "DIS" en la declaración de variables.

La tabla siguiente muestra el significado de los valores individuales de los parámetros.

Parámetro	Tipo de datos	Valor	Descripción
MaintenanceState	DWORD	Enum	
		0	Mantenimiento no requerido
		1	El módulo o el dispositivo están desactivados.
		2	-
		3	-
		4	-
		5	Mantenimiento necesario
		6	Mantenimiento solicitado
		7	Error
		8	Estado desconocido/error en módulo subordinado
		9	-
		10	Las entradas/salidas no están disponibles.

Parámetro	Tipo de datos	Valor	Descripción
ComponentStateDetail	DWORD	Matriz de bits	Estado de los submódulos del módulo: • bits de 0 a 15: Mensaje de estado del módulo • Bits 16 a 31: Mensaje de estado de la CPU
		0 a 2 (enum)	Información adicional: • Bit 0: Sin información adicional • Bit 1: No se permite ninguna aceptación
		3	Bit 3 = 1: Por lo menos un canal soporta el qualifier de diagnóstico
		4	Bit 4 = 1: Mantenimiento necesario en por lo menos un canal o un componente.
		5	Bit 5 = 1: Solicitud de mantenimiento en por lo menos un canal o un componente.
		6	Bit 6 = 1: Error en por lo menos un canal o un componente.
		7 a 10	Reservado (siempre "0")
		11 a 14	• Bit 11 = 1: Submódulo PNIO correcto • Bit 12 = 1: Módulo de reserva PNIO • Bit 13 = 1: PNIO: módulo equivocado • Bit 14 = 1: PNIO: módulo no enchufado
		15	Reservado (siempre "0")
		16 a 31	Información de estado para los módulos generada por la CPU: • Bit 16 = 1: Módulo desactivado • Bit 17 = 1: Operación CiR activa • Bit 18 = 1: Entrada no disponible • Bit 19 = 1: Salida no disponible Bit • 20 = 1: Desbordamiento del búfer de diagnóstico Bit • 21 = 1: Diagnóstico no disponible Bit • 22 - 31: Reservado (siempre 0)
OwnState	UINT	Enum	El valor del parámetro Ownstate describe el estado de mantenimiento del módulo.
		0	No hay fallos
		1	El módulo o el dispositivo están desactivados.
		2	Mantenimiento necesario
		3	Mantenimiento solicitado
		4	Error
		5	No es posible acceder al módulo o dispositivo desde la CPU (válido para módulos y dispositivos que están por debajo de una CPU).
		6	Las entradas/salidas no están disponibles.
		7	-
IOState	WORD	Matriz de bits	Estado de E/S del módulo
		0	Bit 0 = 1: Mantenimiento no requerido
		1	Bit 1 = 1: El módulo o el dispositivo están desactivados.
		2	Bit 2 = 1: Mantenimiento necesario
		3	Bit 3 = 1: Mantenimiento solicitado
		4	Bit 4 = 1: Error

Parámetro	Tipo de datos	Valor	Descripción
		5	Bit 5 = 1: No es posible acceder al módulo o dispositivo desde la CPU (válido para módulos y dispositivos que están por debajo de una CPU).
		6	Las entradas/salidas no están disponibles.
		7	Qualifier; bit 7 = 1 si los bits 0, 2 ó 3 están activos
		8 a 15	Reservado (siempre "0")
OperatingState	UINT	Enum	
		0	-
		1	En STOP/actualización del firmware
		2	En STOP/borrado total
		3	En STOP/autoarranque
		4	En STOP
		5	Borrado total
		6	En START
		7	En RUN
		8	-
		9	En PARADA
		10	-
		11	-
		12	Módulo defectuoso
		13	-
		14	Sin corriente
		15	CiR
		16	En STOP/sin ODIS
		17	En
		18	
		19	
		20	

Estructura DNN

En el parámetro MODE = 2 se emiten los detalles de la información de diagnóstico conforme a la estructura DNN. En tal caso indique como tipo de datos el tipo de datos del sistema "DNN" en la declaración de variables.

La tabla siguiente muestra el significado de los valores individuales de los parámetros.

Parámetros	Tipo de datos	Valor	Descripción
SubordinateState	UINT	Enum	Estado del módulo subordinado (véase el parámetro OwnState de la estructura DIS)
SubordinateIOState	WORD	Bitarray	Estado de las entradas y salidas del módulo subordinado (véase el parámetro IO State de la estructura DIS)
DNNmode	WORD	Bitarray	• Bit 0 = 0: Diagnóstico activado • Bit 0 = 1: Diagnóstico desactivado • Bits de 1 a 15: Reservado

Estructura DiagnosticDetail

En el parámetro MODE = 3 se emiten los detalles de la información de diagnóstico conforme a la estructura DiagnosticDetail. En tal caso indique como tipo de datos el tipo de datos del sistema "DiagnosticDetail" en la declaración de variables.

La tabla siguiente muestra el significado de los valores individuales de los parámetros.

Parámetro	Tipo de datos	Descripción
ChannelInfo	WORD	Propiedades del canal (0..7)
ALID	UInt	ID de identificación de la alarma
TextID	UNIT	ID de un texto de aviso en una lista de textos.
ChannelNumber	UINT	Número de canal específico del fabricante (0x0000 — 0x7FFF)
Addval_0	DWORD	Comodines para información adicional. El valor/la lista de valores depende del error de conexión.
TextID2	UInt	Textos para la reacción de la CPU (estado operativo, llamadas de OB, etc.).
LADDR	HW_ANY	Idéntico al parámetro LADDR.
TextListId	UInt	• 0: Sin lista de textos • ≠0: ID de la lista de textos
ChannelDirection	UInt	• 000: Específico de fabricante • 001: Input • 002: Output • 003: Input/Output • 004 - 007: Reservado
AddVal_1	DWORD	Comodines para información adicional sobre el fallo de canal (en función del archivo GSD). En relación con los tipos de fallos de canal, véase también: norma IEC 61158 (PROFINET IO Type 10 and PROFIBUS DP Type 3).

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0	Sin errores
n	El área de datos del parámetro DETAIL es demasiado pequeña. No se pueden emitir todos los detalles de los datos de diagnóstico.
8080	Valor del parámetro MODE no soportado.
8081	El tipo del parámetro DIAG no es soportado por el modo seleccionado (parámetro MODE).
8082	El tipo del parámetro DETAIL no es soportado por el modo seleccionado (parámetro MODE).
8090	LADDR no existe
80C1	Recursos insuficientes para la ejecución paralela.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Impulso

CTRL_PWM: Modulación del ancho de impulso

Descripción

La instrucción "CTRL_PWM" permite activar o desactivar por software un generador de impulsos soportado por la CPU.

- La identificación de hardware del generador de impulsos que debe controlarse mediante la instrucción se especifica en la entrada PWM.
- El generador de impulsos se activa cuando está activado el bit en la entrada ENABLE de la instrucción.
  - Si ENABLE tiene el valor TRUE, el generador genera impulsos con las propiedades definidas en la configuración de dispositivos.
  - Si se desactiva el bit en la entrada ENABLE o la CPU pasa a STOP, se desactiva el generador de impulsos y ya no se generan impulsos.

Puesto que el S7-1200 activa el generador de impulsos cuando se ejecuta la instrucción "CTRL_PWM", BUSY tiene siempre el valor FALSE en el S7-1200.

La salida de habilitación ENO se activa únicamente si la entrada de habilitación EN devuelve el estado lógico "1" y si no han ocurrido errores durante la ejecución de la instrucción.

Nota

Uso de la tabla de forzado con PWM y PTO

Las entradas y salidas digitales que se usan para PWM y PTO no se pueden forzar. Las entradas y salidas digitales asignadas mediante la configuración de dispositivos no se pueden controlar con la tabla de forzado ni con la tabla de observación.

Requisitos

Para poder ejecutar la instrucción correctamente es preciso que el generador de impulsos especificado esté activado en la configuración hardware.

Para ello, abra las Propiedades del módulo en la Vista de dispositivos. Abra el PTO/PWM deseado en "Generadores de impulsos (PTO/PWM)" y active la función "Activar este generador de impulsos" en "General".

Ajuste las Opciones de impulso en "Parametrización".

Nota

Un generador de impulsos se parametriza exclusivamente en la configuración de dispositivos y no mediante la instrucción "CTRL_PWM". Por ello, solo es posible modificar los parámetros que afectan a la CPU cuando esta está en estado operativo STOP. Una excepción es el cambio de la duración de impulso.

Cambio de la duración de impulso desde el programa de usuario

El ajuste efectuado en el cuadro de diálogo "Opciones de impulso" para definir la duración de impulso se puede modificar desde el programa de usuario.

El valor ajustado para "Duración de impulso inicial" se escribe en los bytes de salida del generador de impulsos. La dirección inicial y la dirección final se visualizan en las propiedades del generador de impulsos en "Direcciones E/S".

Para modificar la duración de impulso, escriba los valores deseados en la dirección de palabra de salida indicada en la configuración de dispositivos.

Ejemplo:

- Para "Duración de impulso inicial" se utiliza el valor 500 (decimal). La dirección inicial del PTO/PWM es "1000", la dirección final "1001".
- En ambos bytes de salida se escribe el valor binario "0000000111110100" (=500 decimal).
  - Dirección inicial (AB1000): 0000_0001 (BIN)
  - Dirección final (AB1001): 1111_0100 (BIN)

Observe que la duración de impulso varía siempre en función del Formato de duración de impulso parametrizado (centésima, milésima, ...).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CTRL_PWM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PWM	Input	HW_PWM	I, Q, M, D, L o constante	Identificación de hardware del generador de impulsos La ID de hardware se puede consultar en las propiedades del generador de impulsos en la Vista de dispositivos. Las ID de hardware de los generadores de impulsos también figuran en las constantes de sistema.
ENABLE	Input	BOOL	I, Q, M, D, L o constante	El generador de impulsos se activa con ENABLE = TRUE y se desactiva con ENABLE = FALSE.
BUSY	Output	BOOL	I, Q, M, D, L	Estado de ejecución
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción (véase a continuación).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
0	Sin errores
80A1	La identificación de hardware del generador de impulsos no es válida.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Recetas y Data Logging

Funciones de receta

RecipeExport: exportar receta

Descripción

La instrucción "RecipeExport" exporta los datos de receta de un bloque de datos en forma de archivo CSV a la Memory Card de la CPU.

La exportación se inicia a través del parámetro REQ. Durante la exportación el parámetro BUSY se pone a "1". Además, al exportar se crea el archivo CSV en la carpeta "Recipes", en el directorio principal de la Memory Card. El archivo CSV creado recibe el nombre del bloque de datos. Si ya hay un archivo CSV con el mismo nombre, dicho archivo se borrará antes de la exportación.

Tras ejecutar la instrucción, BUSY se restablece a "0" y en el parámetro DONE se indica con "1" que la instrucción ha finalizado. Si se produce algún error durante la ejecución, se indicará mediante los parámetros ERROR y STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RecipeExport":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Parámetro de control REQUEST: Activa la exportación con un flanco ascendente.
RECIPE_DB	InOut	VARIANT	D	Puntero hacia el bloque de datos de receta. Consulte la estructura del bloque de datos en: Estructura de un DB de receta (Página 4967)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: No se ejecuta la instrucción. • 1: Se ejecuta la instrucción.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ni advertencia ni error. • 1: Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado Véase tabla "Parámetro STATUS".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Explicación
0	No ha ocurrido ningún error
7000	No se está procesando ninguna petición
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8090	El nombre del archivo CSV contiene caracteres no válidos. El nombre del archivo CSV es idéntico al del bloque de datos.
8091	No puede procesarse la estructura de datos referenciada mediante RECIPE_DB.
8092	La estructura de datos del parámetro RECIPE_DB excede los 5000 bytes.
80B3	No hay espacio suficiente en la Memory Card o en la memoria de carga interna.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	Archivo CSV bloqueado temporalmente.

Código de error* (W#16#...)	Explicación
80C1	Bloque de datos bloqueado temporalmente.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RecipeImport: importar receta

Descripción

La instrucción "RecipeImport" importa los datos de receta de un archivo CSV a la Memory Card, al bloque de datos del parámetro RECIPE_DB. En este proceso se sobrescriben los valores existentes en el bloque de datos.

Para importar el archivo CSV, tenga en cuenta lo siguiente:

- El archivo CSV debe estar en el directorio "Recipes" de la Memory Card.
- El nombre del archivo CSV debe ser idéntico al nombre del bloque de datos del parámetro RECIPE_DB.
- La primera fila (encabezado) del archivo CSV contiene los nombres de los componentes de la receta (véase también: Estructura de un DB de receta (Página 4967)). Dicha primera fila se ignora durante la importación. Los nombres de los componentes de la receta del archivo CSV y del bloque de datos no se comparan durante la importación.
- El primer valor de cada fila del archivo CSV es el índice de la receta. Las recetas individuales se importan siguiendo el orden del índice. Por lo tanto, el índice del archivo CSV debe ser ascendente y no contener espacios vacíos (de lo contrario se emitirá el aviso de error 80B0 en el parámetro STATUS).
- El archivo CSV no debe contener más registros de receta de los previstos en el bloque de datos. El número máximo de registros en el bloque de datos viene determinado por los límites de la matriz.

La importación se inicia a través del parámetro REQ. Durante la importación el parámetro BUSY se pone a "1". Tras ejecutar la instrucción, BUSY se restablece a "0" y en el parámetro DONE se indica con "1" que la instrucción ha finalizado. Si se produce algún error durante la ejecución, se indicará mediante los parámetros ERROR y STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RecipeImport":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Parámetro de control REQUEST: Activa la importación con un flanco ascendente
RECIPE_DB	InOut	VARIANT	D	Puntero hacia el bloque de datos de receta. Consulte la estructura del bloque de datos en: Estructura de un DB de receta (Página 4967)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: No se ejecuta la instrucción. • 1: Se ejecuta la instrucción.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ni advertencia ni error. • 1: Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado Véase tabla "Parámetro STATUS".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Explicación
0	No ha ocurrido ningún error
7000	No se está procesando ninguna petición
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8090	El nombre de archivo contiene caracteres no válidos.
8092	No se ha encontrado ningún archivo CSV concordante para la importación. Causa posible: el nombre del archivo CSV no coincide con el nombre del DB de receta.
80C0	El archivo CSV está bloqueado temporalmente.
80C1	El bloque de datos está bloqueado temporalmente.
80B0	La numeración del índice del archivo CSV no es homogénea, no es ascendente o supera el número máximo (límite de la matriz) del bloque de datos.

Código de error* (W#16#...)	Explicación
80B1	La estructura del bloque de datos de receta y del archivo CSV no coinciden: el archivo CSV tiene demasiados campos.
80B2	La estructura del bloque de datos de receta y del archivo CSV no coinciden: el archivo CSV no tiene suficientes campos.
80D0 +n	La estructura del bloque de datos de receta y del archivo CSV no coinciden: el tipo de datos del campo n no coincide (n <= 46).
80FF	La estructura del bloque de datos de receta y del archivo CSV no coinciden: el tipo de datos del campo n no coincide (n > 46).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Estructura de un DB de receta

Introducción

A continuación se describe la estructura de un DB de receta mediante un sencillo ejemplo. El DB de receta se compone de cinco registros, de los cuales se utilizan tres. El cuarto y el quinto registro se dejan libres para extensiones posteriores. Cada registro contiene una receta, que a su vez se compone de un nombre de receta y ocho ingredientes.

product name	water	barley	wheat	hops	yeast	waterTmp	mashTmp	mashTime	QTest
Pils	10	9	3	280	39	40	30	100	0
Lager	10	9	3	150	33	50	30	120	0
BlackBeer	10	9	3	410	47	60	30	90	1
Not_used	0	0	0	0	0	0	0	0	0
Not_used	0	0	0	0	0	0	0	0	0

Estructura del bloque de datos de receta

En un bloque de datos global, los datos de receta se disponen del siguiente modo:

- La plantilla para todas las recetas es el tipo de datos PLC "Beer_Recipe" con los componentes de receta "procutname", "water", etc. con los correspondientes tipos de datos.
- En un bloque de datos global, el tipo de datos PLC se utiliza como Array [1.. 5] of "Beer_Recipe". El número máximo de recetas que puede contener el DB lo determinan los límites de la matriz (en este caso de 1 a 5).

- Los valores de los componentes de receta se añaden al bloque de datos como valores de arranque.
- A través del parámetro InOut RECIPE_DB se interconecta el DB global con la instrucción.

Recipe_DB					
		Name	Data type	Offset	Start value
1		Static			
2		Products	Array [1 .. 5] of "Beer_Recipe"	...	
3		Products[1]	"Beer_Recipe"	...	
4		Products[2]	"Beer_Recipe"	...	
5		Products[3]	"Beer_Recipe"	...	
6		productname	String[20]	...	'BlackBeer'
7		water	UInt	...	10
8		barley	UInt	...	9
9		wheat	UInt	...	3
10		hops	UInt	...	410
11		yeast	UInt	...	47
12		waterTmp	UInt	...	60
13		mashTmp	UInt	...	30
14		mashTime	UInt	...	90
15		QTest	UInt	...	1
16		Products[4]	"Beer_Recipe"	...	
17		Products[5]	"Beer_Recipe"	...	

Exportar a archivo CSV

Tras ejecutar la instrucción "RecipeExport (Página 4963)", los datos del DB se escriben en un archivo CSV con la siguiente estructura:

Recipe_DB.csv

```
index,productname,water,barley,wheat,hops,yeast,waterTmp,mashTmp,mashTime,QTest
1,”Pils”,10,9,3,280,39,40,30,100,0
2,”Lager”,10,9,3,150,33,50,30,120,0
3,”BlackBeer”,10,9,3,410,47,60,30,90,1
4,”Not_used”,0,0,0,0,0,0,0,0,0
5,”Not_used”,0,0,0,0,0,0,0,0,0
```

Representación en Excel

El archivo CSV se puede abrir para con Excel para visualizarlo de forma clara y editarlo. Si al abrirlo no se reconocen las comas como carácter de separación, utilice la función de importación de Excel para mostrar los datos de forma estructurada:

	A	B	C	D	E	F	G	H	I	J	K
1	index	product	water	barley	wheat	hops	yeast	waterTmp	mashTmp	mashTime	QTest
2	1	"Pils"	10	9	3	280	39	40	30	100	0
3	2	"Lager"	10	9	3	150	33	50	30	120	0
4	3	"BlackBeer"	10	9	3	410	47	60	30	90	1
5	4	"Not_used"	0	0	0	0	0	0	0	0	0
6	5	"Not_used"	0	0	0	0	0	0	0	0	0

Edición del archivo CSV

El archivo CSV se puede cargar y editar en el PC o la programadora a través del servidor web: Después de la edición se puede volver a cargar en la CPU el archivo modificado, La instrucción "RecipeImport (Página 4965)" permite volver a importar los datos modificados del archivo CSV al bloque de datos.

Recuerde que para ello los datos modificados también tienen que ser compatibles con el bloque de datos. Es decir:

- En la tabla no se deben realizar modificaciones en la estructura (por ejemplo, añadiendo ingredientes en una nueva columna).
- Si añade registros adicionales al archivo, al importarlo al bloque de datos debe tener en cuenta que los límites de la matriz que determinan el número máximo de registros deben coincidir como mínimo con el número de registros.
- Durante la exportación al archivo CSV se genera automáticamente un índice. Si crea registros adicionales, añádalos conforme al número de índice correlativo.
- Los valores de las celdas de la tabla deben corresponderse con los tipos de datos utilizados en el bloque de datos, tanto en formato como en longitud.
  - Ejemplo 1: Si en el bloque de datos se ha utilizado el tipo de datos INT, en la tabla solo podrán utilizarse números enteros.
  - Ejemplo 2: Si en el bloque de datos se ha utilizado el tipo de datos SINT, en la tabla solo podrán utilizarse números enteros con los valores -128 a +127.

  Si realiza cambios en la tabla, observe en general los tipos de datos y áreas de datos permitidos, tal y como se relacionan en la tabla siguiente.

Tipo de datos		Formato	Nota
Números en coma flotante	LReal	+9.9999999999999E +999	Siempre en notación exponencial
	Real	+9.999999E+99	Siempre en notación exponencial
Enteros con signo	LInt	+9999999999999999999	Rango de valores para enteros con signo: -9223372036854775808 .. +9223372036854775807
	DInt	+9999999999	Rango de valores enteros: -2147483648 a +2147483647
	Int	+99999	Rango de valores enteros: -32768 a +32767
	SInt	+999	Rango de valores enteros: -128 a +127
Enteros sin signo	ULInt	+9999999999999999999 9	Rango de valores enteros: 0 a +18446744073709551615
	UDInt	+9999999999	Rango de valores enteros: 0 a +4294967295
	UInt	+99999	Rango de valores enteros: 0 a + 65535
	USInt	+999	Rango de valores enteros: 0 a +255
Números binarios	LWord	+9999999999999999999 9	Rango de valores enteros: +00000000000000000000 a +18446744073709551615
	DWord	+9999999999	Rango de valores enteros: +0000000000 a +4294967295
	Word	+99999	Rango de valores enteros: +00000 a + 65535
	Byte	+999	Rango de valores enteros: +000 a +255
	Bool	9	Rango de valores: 0 ó 1
Fecha y hora	LTIME	hhh:mm:ss.999	Formato ISO con milisegundos
	TIME	hhh:mm:ss.999	Formato ISO con milisegundos
	S5TIME	hhh:mm:ss.999	Formato ISO con milisegundos
	LDT	YYYY-MM-DD hh:mm:ss. 999	Formato ISO con milisegundos
	DTL	YYYY-MM-DD hh:mm:ss. 999	Formato ISO con milisegundos

Tipo de datos		Formato	Nota
	DT	YYYY-MM-DD hh:mm:ss.999	Formato ISO con milisegundos
	DATE	YYYY-MM-DD	Formato ISO
	LTime_Of_Day	hh:mm:ss.999_999_999	Formato ISO
	TOD	hh:mm:ss.999	Formato ISO
Carácter	String	"abcd"	La secuencia de caracteres debe estar entre comillas.
	Char	"a"	Carácter individual entre comillas.

Data Logging

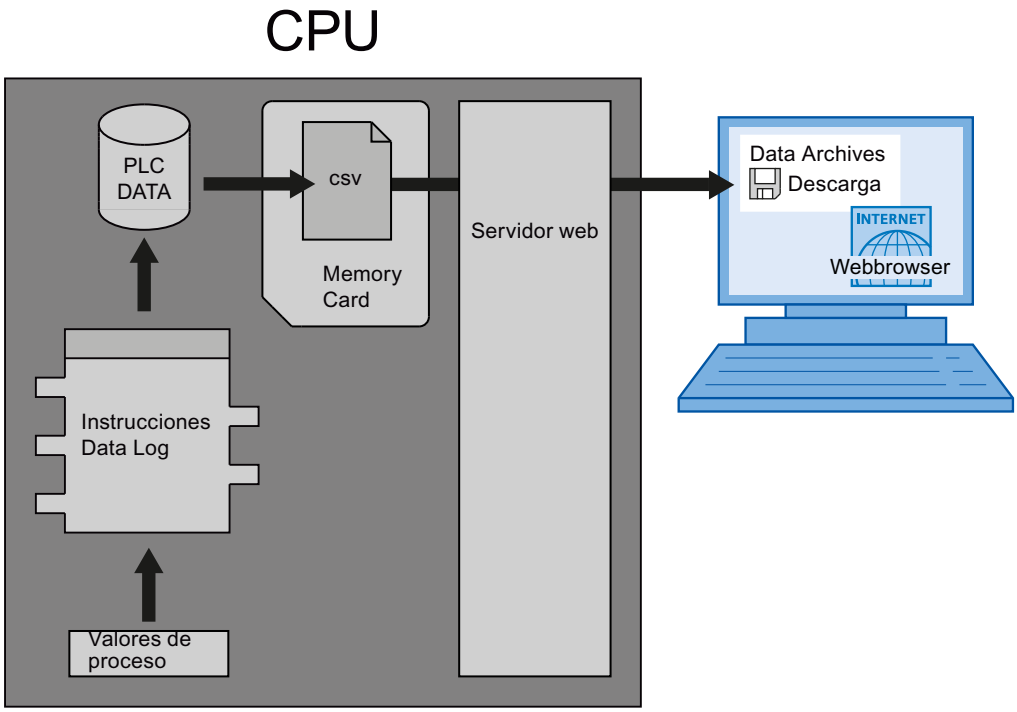
Data Logging - Übersicht

Guardar valores de proceso

Las instrucciones de Data Logging se utilizan en el programa de usuario para guardar valores de proceso en Data Logs. Los Data Logs pueden guardarse en la Memory Card (MC) o en la memoria de carga interna. Los Data Logs se guardan en formato CSV (Comma Separated Values).

Las instrucciones de Data Logging se utilizan en el programa para crear o abrir un Data Log, para escribir una entrada y para cerrar el archivo Data Log.

Se puede decidir qué valores de programa se guardan en el Data Log al crear el búfer de datos. El búfer de datos se utiliza como memoria para nuevas entradas Data Log. Los valores nuevos se deben escribir en el búfer antes de la llamada de "DataLogWrite (Página 4985)". Durante la ejecución de la instrucción "DataLogWrite (Página 4985)" los datos del búfer se escriben en un registro Data Log.



Los archivos Data Log se pueden copiar en el PC de la siguiente manera:

- Si la interfaz PROFINET está conectada con el PC, es necesario utilizar un navegador web para acceder a los Data Logs a través del servidor web. La CPU puede estar en modo "RUN" o "STOP". Si la CPU está funcionando en modo "RUN", el programa continúa mientras el servidor web efectúa una transferencia de datos.
- Si en la CPU hay una Memory Card, esta puede extraerse e introducirse en una unidad insertable estándar para tarjetas SD (Secure Digital) o tarjetas MMC (MultiMediaCard) de un PC o una programadora. Utilice el administrador de archivos para transferir los archivos Data Log de la Memory Card al PC. La CPU pasa a "STOP" al extraer la Memory Card.

Propiedades de Data Log

La escritura de los registros de un Data Log se realiza según el principio de un búfer en anillo. Los registros nuevos se van añadiendo hasta que se alcanza el número máximo de registros (parámetro RECORD). El siguiente registro sobrescribe entonces el registro "más antiguo" del Data Log.

Si quiere evitar que se sobrescriban registros, utilice la instrucción "DataLogNewFile (Página 4991)" para crear un archivo Data Log basado en el Data Log actual. Los nuevos registros se escriben entonces en el Data Log nuevo.

Crear Data Logs

La instrucción "DataLogCreate (Página 4973)" crea un archivo Data Log en el directorio ""\DataLogs" de la memoria de carga.

- El nombre asignado en el parámetro NAME es la designación del Data Log y también se utiliza como nombre de archivo para el archivo CSV. El archivo se guarda en el directorio "DataLogs".
- El parámetro de bloque DATA determina el búfer de datos para el nuevo objeto Data Log así como las columnas y tipos de datos en el Data Log. Las columnas y los tipos de datos de un registro en el Data Log se generan a partir de los elementos de la declaración de estructura o de matriz de este búfer de datos. Cada elemento de una estructura o de una matriz se corresponde con una columna en una fila del Data Log.
- Con el parámetro de bloque HEADER se puede asignar un título en el encabezado a cada columna.
- La instrucción "DataLogCreate (Página 4973)" devuelve una ID. Las demás instrucciones de Data Logging utilizan esta ID como referencia para el Data Log creado.

Abrir Data Logs

Las instrucciones "DataLogOpen" (S7-1200 y S7-1500) abren un Data Log existente en la Memory Card. Un Data Log tiene que estar abierto para escribir nuevos registros en él.

Durante la ejecución de las instrucciones "DataLogCreate (Página 4973)" y "DataLogNewFile (Página 4991)", el Data Log se abre automáticamente.

Se pueden tener abiertos simultáneamente 10 Data Logs como máximo. El Data Log que se va a abrir se puede seleccionar con la ID o con el nombre del Data Log.

- Si se indica tanto la ID como el nombre del Data Log en los parámetros ID y NAME, el Data Log se identifica con la ID. No se compara el nombre del Data Log.
- Si se selecciona el Data Log con el parámetro NAME y no se indica ninguna ID, la ID se muestra en el parámetro ID al abrir el Data Log.
- Si se selecciona el Data Log con el parámetro ID y no se indica ningún nombre, el nombre no se muestra en el parámetro NAME al abrir el Data Log.

Con el parámetro MODE se selecciona si los registros del Data Log se deben borrar al abrirlo.

Escribir en el Data Log

Para escribir un registro en un Data Log es imprescindible tener un Data Log abierto (instrucción "DataLogOpen (Página 4980)"). La instrucción "DataLogWrite (Página 4985)" escribe un registro en el Data Log.

Cerrar Data Logs

La instrucción "DataLogClose (Página 4987)" cierra un Data Log abierto. El Data Log se selecciona con el parámetro ID.

El Data Log se cierra automáticamente cuando la CPU pasa a STOP o en caso de rearranque.

Borrar Data Logs

La instrucción "DataLogDelete (Página 4989)" (S7-1500) borra un archivo Data Log existente en la Memory Card. El Data Log y los registros que contiene solo pueden borrarse si el Data Log se ha creado mediante la instrucción "DataLogCreate (Página 4973)".

Los parámetros NAME e ID permiten seleccionar el Data Log que debe borrarse. El parámetro ID es el primero que se evalúa. Si existe un Data Log con la ID correspondiente, el parámetro NAME no se evalúa. Si el parámetro ID tiene el valor "0", en el parámetro NAME debe utilizarse un valor del tipo de datos STRING.

Vaciar Data Logs

La instrucción "DataLogClear (Página 4984)" (S7-1500) borra todos los registros de un Data Log existente. No se borra el encabezado opcional del archivo CSV (véase la descripción del parámetro HEADER de la instrucción "DataLogCreate (Página 4973)").

Con la ID de parámetro se selecciona el Data Log del que se van a borrar los registros. Para borrar los registros, el Data Log tiene que estar abierto.

Archivo nuevo para Data Logs

Las instrucciones "DataLogNewFile (Página 4991)" (S7-1200) y "DataLogTypedNewFile (Página 4992)" (S7-1500) crean un Data Log nuevo con las mismas propiedades que un Data Log ya existente. De esta forma es posible obtener los contenidos de un Data Log existente.

Al ser llamada, la instrucción crea un Data Log en la Memory Card o en la memoria de carga interna con el nombre definido en el parámetro NAME. Con el parámetro ID se indica la ID del Data Log antiguo cuyas propiedades se quieren adoptar para el Data Log nuevo. Con el parámetro ID se indica a continuación la ID del Data Log nuevo.

El tamaño de archivo del Data Log nuevo se indica con el parámetro RECORDS de la instrucción.

Se puede realizar una comprobación de coherencia para "DataLogTypedNewFile (Página 4992)" (S7-1500).

DataLogCreate: Crear Data Log

Descripción

Con la instrucción "DataLogCreate", el usuario crea un Data Log.

El Data Log se guarda en la Memory Card o en la memoria de carga interna en el directorio "\DataLogs". La cantidad de datos que se puede guardar en un Data Log depende de la memoria disponible en la Memory Card o en la memoria de carga interna de la CPU utilizada.

El número máximo de registros que se pueden guardar en un Data Log se indica en el parámetro RECORDS. Una vez que se alcanza el número máximo de registros en el Data Log, se va sobrescribiendo el registro más antiguo. Para evitar que se sobrescriban los registros existentes, se puede utilizar la instrucción "DataLogNewFile (Página 4991)". Al alcanzar el número indicado en el parámetro RECORDS (valor de retorno 1 del parámetro STATUS de la instrucción "DataLogWrite (Página 4985)") se puede utilizar la instrucción para

crear un nuevo Data Log con la misma estructura. Los registros se guardan entonces en el nuevo Data Log.

El nombre del Data Log se puede indicar en el parámetro NAME. El Data Log se crea en formato CSV (Comma Separated Values). Con el parámetro HEADER se puede crear un encabezado (opcional) para el Data Log.

Tras la creación, el Data Log se abre automáticamente. Esto significa que se pueden escribir datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogCreate":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecución de la instrucción El Data Log se crea cuando se detecta un flanco ascendente en el parámetro REQ.
RECORDS	Input	UDInt	I, Q, M, L, D o constante	Número máximo de registros en Data Log Si con la instrucción "DataLogWrite (Página 4985)" se escriben más registros que los especificados en este parámetro, el registro más antiguo se sobrescribe.
FORMAT	Input	UInt	I, Q, M, L, D o constante	Formato de datos: • 0: Interno (no soportado) • 1: CSV (Comma separated values)
TIMESTAMP	Input	UInt	I, Q, M, L, D o constante	Sellado de tiempo: • 0: Sin sellado de tiempo • 1: Fecha y hora Si el sellado de tiempo está activado, se agregan automáticamente columnas adicionales en el encabezado.
NAME	Input	VARIANT	L, D	Nombre del Data Log El nombre asignado también se utiliza como nombre de archivo del archivo CSV. En la asignación del nombre se aplican las restricciones para nombres de archivos de Windows. Los siguientes caracteres no pueden utilizarse: "\", "/", ":", "*", "?", "<", ">", "|", "espacio"
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log (solo Output) La ID del Data Log se necesita para otras instrucciones de Data Logging a fin de direccionar el Data Log creado.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
HEADER	InOut	VARIANT	L, D	Encabezado del Data Log (opcional) El parámetro se oculta después de insertar la instrucción. El encabezado se escribe en la primera fila del archivo CSV.
DATA	InOut	VARIANT	L, D	Puntero hacia la estructura de datos que debe escribirse como registro al ejecutar la instrucción "DataLogWrite (Página 4985)".
DONE	Output	BOOL	I, Q, M, L, D	Parámetros de estado: • 0: el procesamiento todavía no ha finalizado • 1: procesamiento de la instrucción finalizado correctamente.
BUSY	Output	BOOL	I, Q, M, L, D	Parámetros de estado: • 0: el procesamiento de la instrucción aún no ha comenzado, ha terminado o se ha interrumpido. • 1: la instrucción se está procesando.
ERROR	Output	BOOL	I, Q, M, L, D	Parámetros de estado: • 0: Ningún error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Información de estado detallada: En el parámetro STATUS se visualiza información de estado y error detallada. El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro HEADER

El parámetro HEADER es un puntero VARIANT a un bloque de datos que define un encabezado para el archivo CSV (Header). El encabezado siempre es la primera fila en la representación de un archivo CSV.

- En la creación de un encabezado, las diferentes columnas tienen que estar separadas por una coma (S7-1200) o por un punto y coma (S7-1500).
- Como tipo de datos se puede utilizar para la designación de las columnas un STRING, Array of BYTE o Array of CHAR. Con la utilización del tipo de datos Array [...] of type se puede utilizar una cadena de caracteres más larga que con el tipo de datos STRING. Si se utiliza STRING la longitud está limitada a 254 bytes.

Si no desea crear ningún encabezado, no indique ningún valor en el parámetro HEADER.

Parámetro DATA

El parámetro DATA es un puntero VARIANT a una estructura o matriz de un bloque de datos. Un elemento de una estructura o matriz se corresponde con una columna en el Data Log con un tipo de datos determinado.

Tenga en cuenta lo siguiente en la creación del bloque de datos:

- El número de columnas debe corresponderse con el número de columnas definido en el parámetro HEADER.
- Cada elemento de la estructura o del Array tiene asignada una entrada en la columna del archivo CSV. Por tanto, si se utiliza el tipo de datos STRUCT no se pueden utilizar anidamientos de estructura (STRUCT en STRUCT).
- La estructura puede contener hasta 256 elementos. Si contiene más de 256 elementos, se emite el código de error 8C52 en el parámetro STATUS.
- Las variables del bloque de datos pueden ajustarse como remanentes o no remanentes. Sin embargo, el ajuste de remanencia tiene que ser igual para todas las variables del bloque de datos.

Parámetro STATUS (S7-1200)

Código de error* (W#16#...)	Descripción
0	Ningún error.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8070	Toda la memoria interna de la instancia está ocupada.
8090	Nombre de archivo no admisible (véase la descripción del parámetro NAME).
8093	El Data Log ya existe.
8097	La longitud del archivo sobrepasa el límite del sistema de archivos.
80B3	No hay espacio de memoria suficiente en la Memory Card.
80B4	La Memory Card tiene activada la protección contra escritura.
80C1	Demasiados Data Logs abiertos.
80C3	Recursos insuficientes. Causas posibles: • Llamada múltiple de instrucciones con distintos parámetros. • Más de 10 Data Logs abiertos simultáneamente. Dado que la instrucción "DataLogCreate" abre automáticamente el Data Log que debe crearse, primero deberá cerrar al menos un Data Log para continuar.
8453	Selección de formato no válida.
8553	Sellado de tiempo no válido.
8B51	Tipo de datos no admisible en el parámetro HEADER.
8C20	String utilizado con una longitud diferente a 254.
8C51	Tipo de datos no admisible en el parámetro DATA.
8C52	La estructura en el parámetro DATA contiene más de 256 elementos.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (S7-1500)

Código de error* (W#16#...)	Descripción
0	Ningún error.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
8070	Toda la memoria interna de la instancia está ocupada.
8090	Nombre de archivo no admisible (véase la descripción del parámetro NAME).
8091	El parámetro "NAME" no es un string.
8093	El Data Log ya existe.
8097	La longitud del archivo sobrepasa el límite del sistema de archivos.
80B3	No hay espacio de memoria suficiente en la Memory Card.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	Acceso no disponible actualmente.
80C1	Demasiados Data Logs abiertos.
80C3	Recursos insuficientes. Causas posibles: • Llamada múltiple de instrucciones con distintos parámetros. • Más de 10 Data Logs abiertos simultáneamente. Dado que la instrucción "DataLogCreate" abre automáticamente el Data Log que debe crearse, primero deberá cerrar al menos un Data Log para continuar.
8253	Valor no válido en el parámetro RECORDS.
8353	Selección de formato no válida.
8453	Sellado de tiempo no válido.
8B51	Tipo de datos no admisible en el parámetro HEADER.
8C51	Tipo de datos no admisible en el parámetro DATA.
8C52	La estructura en el parámetro DATA contiene más de 256 elementos.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente se crea un Data Log sencillo con un sellado de tiempo y tres valores de proceso.

Variables del bloque de datos global

Los valores para los parámetros de entrada del Data Log están guardados en el bloque de datos global "DataLogDB":

- DataLogName (String): la variable contiene el nombre del Data Log que también se usa como nombre del archivo CSV.
- DataLogID (DInt): en esta variable se escribe la ID del Data Log cuando se llama la instrucción.
  - La instrucción asigna automáticamente la ID.
  - La variable "DataLogID" se utiliza en otras instrucciones de DataLog para direccionar el Data Log correspondiente.
- MyHeader (String): la variable contiene el encabezado del DataLog, es decir, los títulos de columna para los valores de proceso. Si emplea un S7-1500 se utiliza un punto y coma como carácter de separación para las columnas.
- MyData (Struct): la variable contiene los tres valores de proceso que se escriben en el Data Log. Cada vez que se escribe un registro (instrucción "DataLogWrite (Página 4985)"), los valores actuales se escriben en un nuevo registro.

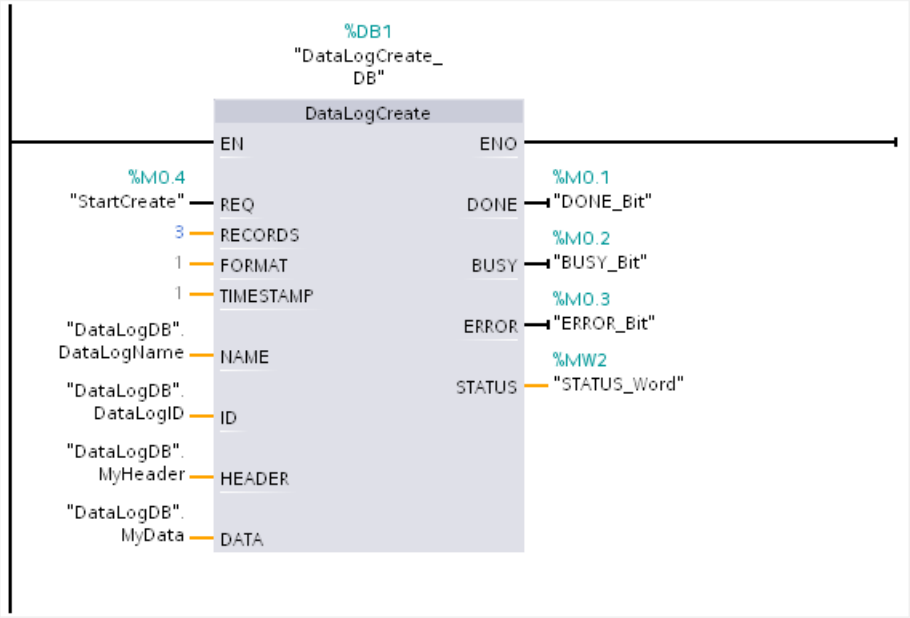
DataLogDB

		Nombre	Tipo de datos	Valor de arranque	Remanen...
1		Static			
2		DataLogName	String	'MyDataLog'	
3		DataLogID	DInt	0	
4		MyHeader	String	'Value1;Value2;Value3'	
5		MyData	Struct		
6		ProcessValue1	Int	2	
7		ProcessValue2	Int	3	
8		ProcessValue3	Int	4	

Llamada de la instrucción "DataLogCreate"

La instrucción se llama con los siguientes parámetros de entrada:

- REQ (BOOL): si REQ = "1" se crea el Data Log.
- RECORD (3): en el Data Log se pueden escribir como máximo tres registros. A continuación se sobrescribe el registro más antiguo.
- FORMAT (1): el Data Log se crea como archivo CSV.
- TIMESTAMP (1): activado. Para el Data Log se crean automáticamente dos columnas adicionales (fecha y hora). El sello de tiempo actual se escribe en el registro cada vez que se ejecuta "DataLogWrite (Página 4985)".
- NAME (VARIANT): puntero hacia la variable "DataLogName" en el bloque de datos "DataLogDB".
- ID (VARIANT): puntero hacia la variable "DataLogID" en el bloque de datos "DataLogDB" (solo Output).
- HEADER (VARIANT): puntero hacia la variable "MyHeader" en el bloque de datos "DataLogDB".
- DATA (VARIANT): puntero hacia la variable "MyData" en el bloque de datos "DataLogDB".



Lectura del Data Log a través del servidor web

A través del servidor web se puede leer el Data Log creado:

- Active el servidor web en las propiedades de la CPU. Para leer el Data Log debe estar activada la autorización "Leer archivos..." para el servidor web.
- A través del navegador de Internet se puede acceder al servidor web, introduzca para ello la dirección IP de la CPU como URL.
- En "Filebrowser" se ha creado automáticamente el directorio "\DataLogs" que contiene el Data Log.
- Si la instrucción "DataLogWrite (Página 4985)" aún no se ha ejecutado, el DataLog solo contiene la entrada "//END". Tras ejecutar por primera vez "DataLogWrite (Página 4985)" se escribe el primer registro.

	A	B	C	D	E	F
1	SeqNo	Date	Time	ProcessValue1	ProcessValue2	ProcessValue3
2	1	01.07.2013	12:33:42.917	2	3	4
3						
4						
5						

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Data Logging - Übersicht (Página 4970)

DataLogOpen: Abrir Data Log

DataLogOpen: Abrir Data Log

Descripción

La instrucción "DataLogOpen" abre un Data Log existente en la Memory Card. Un Data Log tiene que estar abierto para escribir nuevos registros en él.

Durante la ejecución de las instrucciones "DataLogCreate (Página 4973)" y "DataLogNewFile (Página 4991)", el Data Log se abre automáticamente.

Se pueden tener abiertos simultáneamente 10 Data Logs como máximo. El Data Log que se va a abrir se puede seleccionar con la ID o con el nombre del Data Log.

- Si se indica tanto la ID como el nombre del Data Log en los parámetros ID y NAME, el Data Log se identifica con la ID. No se compara el nombre del Data Log.
- Si se selecciona el Data Log con el parámetro NAME y no se indica ninguna ID, la ID se muestra en el parámetro ID al abrir el Data Log.
- Si se selecciona el Data Log con el parámetro ID y no se indica ningún nombre, el nombre no se muestra en el parámetro NAME al abrir el Data Log.

Con el parámetro MODE se selecciona si los registros del Data Log se deben borrar al abrirlo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogOpen":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Ejecución de la instrucción con flanco ascendente.
MODE	Input	UInt	I, Q, M, L, D o constante	Modo para abrir el Data Log: • MODE= "0" Los registros del Data Log se mantienen • MODE= "1" Los registros del Data Log se borran, el encabezado se mantiene
NAME	Input	VARIANT	L, D	Nombre (de archivo) del Data Log.
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log.
DONE	Output	BOOL	I, Q, M, L, D	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, L, D	Ejecución de la instrucción no finalizada.

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, L, D	• 0: Sin errores. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Parámetros de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
0	Sin errores.
2	Advertencia: El archivo Data Log ya fue abierto por esta aplicación.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8070	Toda la memoria de instancia interna está ocupada.
8090	Hay incoherencias entre la definición Data Log y los datos Data Log existentes.
8091	En el parámetro NAME se ha utilizado un tipo de datos diferente a String.
8092	El Data Log no existe.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	El archivo Data Log está bloqueado.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Data Logging - Übersicht (Página 4970)

DataLogOpen: Abrir Data Log

Descripción

La instrucción "DataLogOpen" abre un Data Log existente en la Memory Card. Un Data Log tiene que estar abierto para poder escribir nuevos registros en él.

Durante la ejecución de las instrucciones "DataLogCreate (Página 4973)" y "DataLogNewFile (Página 4991)", el Data Log se abre automáticamente.

Se pueden tener abiertos simultáneamente 10 Data Logs como máximo. El Data Log que se va a abrir se puede seleccionar con la ID o con el nombre del Data Log.

- Si se indica tanto la ID como el nombre del Data Log en los parámetros ID y NAME, el Data Log se identifica con la ID. No se compara el nombre del Data Log.
- Si se selecciona el Data Log con el parámetro NAME y no se indica ninguna ID, la ID se muestra en el parámetro ID al abrir el Data Log.
- Si se selecciona el Data Log con el parámetro ID y no se indica ningún nombre, el nombre no se muestra en el parámetro NAME al abrir el Data Log.

Con el parámetro MODE se selecciona si los registros del Data Log se deben borrar al abrirlo.

El parámetro DATA permite comprobar la coherencia entre el Data Log que se va a abrir y la definición del Data Log de la instrucción "DataLogCreate (Página 4973)". La prueba de coherencia solo se puede ejecutar si el Data Log ha sido creado con la instrucción "DataLogCreate (Página 4973)":

- Si utiliza en el parámetro DATA el mismo puntero que en el parámetro DATA de la instrucción "DataLogCreate (Página 4973)", se comprueba si los tipos de datos coinciden. Si no es el caso, el parámetro STATUS emite el código de error W#16#80A0.
- Si el Data Log que se va a abrir no se ha creado con "DataLogCreate (Página 4973)", no es posible realizar la prueba de coherencia. En ese caso, introduzca el valor "NULL" en el parámetro DATA.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogOpen":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecución de la instrucción con flanco ascendente.
MODE	Input	UInt	I, Q, M, L, D o constante	Modo para abrir el Data Log: • MODE= "0" Los registros del Data Log se mantienen • MODE= "1" Los registros del Data Log se borran, el encabezado se mantiene
NAME	Input	VARIANT	L, D	Nombre (de archivo) del Data Log.
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log.
DATA	InOut	VARIANT	L, D	Durante la prueba de coherencia: Puntero hacia el área de datos del parámetro DATA de la instrucción "DataLogCreate (Página 4973)".
DONE	Output	BOOL	I, Q, M, L, D	La instrucción se ha ejecutado correctamente.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BUSY	Output	BOOL	I, Q, M, L, D	Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, L, D	• 0: Sin errores. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
0	Sin errores.
2	Advertencia: El archivo Data Log ya fue abierto por esta aplicación.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8070	Toda la memoria de instancia interna está ocupada.
8090	Hay incoherencias entre la definición Data Log y los datos Data Log existentes.
8091	En el parámetro NAME se ha utilizado un tipo de datos diferente a String.
8092	El Data Log no existe.
80A0	Tipos de datos incoherentes. El Data Log del parámetro ID utiliza otros tipos de datos que los indicados en el parámetro DATA.
80B4	La Memory Card tiene activada la protección contra escritura.
80C1	Demasiados archivos abiertos.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Data Logging - Übersicht (Página 4970)

DataLogClear: Vaciar Data Log

Descripción

La instrucción "DataLogClear" borra todos los registros de un Data Log existente. No se borra el encabezado opcional del archivo CSV (véase la descripción del parámetro HEADER de la instrucción "DataLogCreate (Página 4973)").

Con la ID de parámetro se selecciona el Data Log del que se van a borrar los registros.

Requisitos

Para borrar los registros, el Data Log tiene que estar abierto (véase la instrucción "DataLogOpen (Página 4980)").

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogClear":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecución de la instrucción con flanco ascendente.
ID	InOut	DWORD	I, Q, M, D, L	ID de objeto del Data Log
DONE	Output	BOOL	I, Q, M, D, L	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, D, L	• 0: Sin errores. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Explicación
0000	Sin errores.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8080	El archivo Data Log seleccionado a través del parámetro ID no puede ser procesado por la instrucción "DataLogClear".
8092	El Data Log no existe.
80B0	El Data Log no está abierto.
80B4	La Memory Card tiene activada la protección contra escritura.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Data Logging - Übersicht (Página 4970)

DataLogWrite: Escribir Data Log

Descripción

La instrucción "DataLogWrite" escribe un registro en un Data Log existente. Con el parámetro ID se elige el Data Log en el que se debe escribir el registro. Para crear un nuevo registro, el Data Log tiene que estar abierto. La instrucción crea un nuevo registro con el formato que se había indicado en el parámetro DATA en la creación del Data Log.

Antes de la llamada de la instrucción "DataLogWrite" transfiera los datos a la variable que se conecta al parámetro DATA de la instrucción "DataLogCreate". Durante la ejecución de la instrucción "DataLogWrite" los datos transferidos se copian en el Data Log.

ATENCIÓN
Pérdida de datos de Data Log por interrupción de la alimentación de la CPU Si se interrumpe la alimentación durante la ejecución de la instrucción "DataLogWrite", se pierde el registro que se va a transferir.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogWrite":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecución de la instrucción con flanco ascendente.
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log
DONE	Output	BOOL	I, Q, M, L, D	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, L, D	Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, L, D	• 0: Sin errores. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS (S7-1200)

Código de error* (W#16#...)	Descripción
0	Sin errores
0001	Último registro posible creado al final del archivo. Cuando se crea un registro nuevo, se sobrescribe otro más antiguo.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8070	Toda la memoria de instancia interna está ocupada.
8092	El Data Log no existe.
80B0	El Data Log no está abierto.
80B3	No hay espacio de memoria suficiente en la Memory Card.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	El Data Log está bloqueado.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (S7-1500)

Código de error* (W#16#...)	Descripción
0	Sin errores
0001	Último registro posible creado al final del archivo. Cuando se crea un registro nuevo, se sobrescribe otro más antiguo.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0
8070	Toda la memoria de instancia interna está ocupada.
8092	El Data Log no existe.
80B0	El Data Log no está abierto.
80B3	No hay espacio de memoria suficiente en la Memory Card.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	El Data Log está bloqueado.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Data Logging - Übersicht (Página 4970)

DataLogClose: Cerrar Data Log

Descripción

La instrucción "DataLogClose" cierra un Data Log abierto. El Data Log se selecciona con el parámetro ID.

Nota

Cierre automático de Data Logs

El Data Log se cierra automáticamente cuando la CPU pasa a STOP o en caso de rearranque.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogClose":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecutar función con flanco ascendente.
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log
DONE	Output	BOOL	I, Q, M, L, D	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, L, D	Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, L, D	• 0: Ningún error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS (S7-1200)

Código de error* (W#16#...)	Descripción
0	Ningún error
1	El Data Log no está abierto
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8092	El Data Log no existe.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (S7-1500)

Código de error* (W#16#...)	Descripción
0	Ningún error
1	El Data Log no está abierto
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0
8070	Toda la memoria interna de la instancia está ocupada.
8092	El Data Log no existe.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	Acceso no disponible actualmente.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

DataLogDelete: Borrar Data Log

Descripción

La instrucción "DataLogDelete" borra un archivo Data Log existente en la Memory Card. El Data Log y los registros que contiene solo se pueden borrar si ha sido creado con las instrucciones "DataLogCreate" o "DataLogNewFile".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogDelete":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecución de la instrucción con flanco ascendente.
NAME	Input	VARIANT	L, D	Nombre del archivo Data Log
DELFILE	Input	BOOL	I, Q, M, D, L	• 0: El Data Log se conserva. • 1: El Data Log se borra. El parámetro STATUS contiene información más detallada.
ID	InOut	DWORD	I, Q, M, D, L	ID de objeto del Data Log
DONE	Output	BOOL	I, Q, M, D, L	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Borrado del Data Log aún no finalizado.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• 0: Ningún error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros NAME e ID

Los parámetros NAME e ID permiten seleccionar el Data Log que debe borrarse. El parámetro ID es el primero que se evalúa. Si existe un Data Log con la ID correspondiente, el parámetro NAME no se evalúa. Si el parámetro ID tiene el valor "0", en el parámetro NAME debe utilizarse un valor del tipo de datos STRING.

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	Ningún error.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8091	En el parámetro NAME se utiliza un tipo de datos diferente de STRING.
8092	El Data Log no existe.
80A2	Error de escritura
80B4	La Memory Card tiene activada la protección contra escritura.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

DataLogCreate: Crear Data Log (Página 4973)

DataLogNewFile: Data Log en archivo nuevo

DataLogNewFile: Data Log en archivo nuevo

Descripción

La instrucción "DataLogNewFile" crea un Data Log con las mismas propiedades que un Data Log ya existente. De esta forma es posible obtener los contenidos de un Data Log existente.

Al ser llamada, la instrucción crea un Data Log en la Memory Card o en la memoria de carga interna con el nombre definido en el parámetro NAME. Con el parámetro ID se indica la ID del Data Log antiguo cuyas propiedades se quieren adoptar para el Data Log nuevo. Con el parámetro ID se indica a continuación la ID del Data Log nuevo.

El tamaño de archivo del Data Log nuevo se indica con el parámetro RECORDS de la instrucción.

Tras la creación, el Data Log nuevo se abre automáticamente. Esto significa que se pueden escribir datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DataLogNewFile":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D o constante	Ejecución de la instrucción con flanco ascendente.
RECORDS	Input	UDInt	I, Q, M, L, D o constante	Número de registros en el Data Log nuevo.
NAME	Input	VARIANT	L, D	Nombre de archivo del Data Log nuevo.
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log • In: ID del Data Log ya existente • Out: ID del Data Log nuevo
DONE	Output	BOOL	I, Q, M, L, D	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, L, D	Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, L, D	• 0: Sin errores. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
0	Sin errores.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8070	Toda la memoria de instancia interna está ocupada.
8090	Nombre de archivo no válido.
8091	La ruta no existe.
8092	El Data Log de origen no existe.
8093	El Data Log nuevo ya existe.
8097	La longitud del archivo sobrepasa el límite del sistema de archivos.
80B3	Memoria de carga insuficiente.
80B4	La Memory Card tiene activada la protección contra escritura.
80C1	Demasiados archivos abiertos.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

DataLogNewFile: Data Log en archivo nuevo

Descripción

La instrucción "DataLogNewFile" crea un Data Log con las mismas propiedades que un Data Log ya existente. De esta forma es posible obtener los contenidos de un Data Log existente.

Al ser llamada, la instrucción crea un Data Log en la Memory Card o en la memoria de carga interna con el nombre definido en el parámetro NAME. Con el parámetro ID se indica la ID del Data Log antiguo cuyas propiedades se quieren adoptar para el Data Log nuevo. Con el parámetro ID se indica a continuación la ID del Data Log nuevo.

El tamaño de archivo del Data Log nuevo se indica con el parámetro RECORDS de la instrucción.

El parámetro DATA permite comprobar la coherencia entre el Data Log que se va a crear y la definición del Data Log de la instrucción "DataLogCreate (Página 4973)". La prueba de coherencia solo se puede ejecutar si el Data Log ha sido creado con la instrucción "DataLogCreate (Página 4973)":

- Si utiliza en el parámetro DATA el mismo puntero que en el parámetro DATA de la instrucción "DataLogCreate (Página 4973)", se comprueba si los tipos de datos coinciden. Si no es el caso, el parámetro STATUS emite el código de error W#16#80A0.
- Si el Data Log que se va a abrir no se ha creado con "DataLogCreate (Página 4973)", no es posible realizar la prueba de coherencia. En ese caso, introduzca el valor "NULL" en el parámetro DATA.

Tras la creación, el Data Log nuevo se abre automáticamente. Esto significa que se pueden escribir datos.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "DataLogNewFile":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, L, D, T, C o constante (T y C solo están disponibles en S7-1500 en KOP y FUP)	Ejecución de la instrucción con flanco ascendente.
RECORDS	Input	UDInt	I, Q, M, L, D o constante	Número de registros en el Data Log nuevo.
NAME	Input	VARIANT	L, D	Nombre de archivo del Data Log nuevo.
ID	InOut	DWORD	I, Q, M, L, D	ID de objeto del Data Log • In: ID del Data Log ya existente • Out: ID del Data Log nuevo
DATA	InOut	VARIANT	L, D	Tipo de datos para comprobación de coherencia
DONE	Output	BOOL	I, Q, M, L, D	La instrucción se ha ejecutado correctamente.
BUSY	Output	BOOL	I, Q, M, L, D	Ejecución de la instrucción no finalizada.
ERROR	Output	BOOL	I, Q, M, L, D	• 0: Ningún error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, L, D	Parámetro de estado El parámetro solo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el parámetro STATUS en un área de datos libre.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
0	Ningún error.
7000	No se está procesando ninguna petición.
7001	Inicio de la ejecución de la petición. Parámetro BUSY = 1, DONE = 0.
7002	Llamada intermedia (REQ irrelevante): la instrucción ya está activa; BUSY tiene el valor "1".
8070	Toda la memoria interna de la instancia está ocupada.
8090	• Nombre de archivo no válido. • Tipos de datos incoherentes. El Data Log del parámetro ID utiliza otros tipos de datos que los indicados en el parámetro DATA.

Código de error* (W#16#...)	Descripción
8091	La ruta no existe.
8092	El Data Log de origen no existe.
8093	El Data Log nuevo ya existe.
8097	La longitud del archivo sobrepasa el límite del sistema de archivos.
80B3	Memoria de carga insuficiente.
80B4	La Memory Card tiene activada la protección contra escritura.
80C0	Acceso no disponible actualmente.
80C1	Demasiados archivos abiertos.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Funciones de bloques de datos

CREATE_DB: Generar bloque de datos

Descripción

La instrucción "CREATE_DB" permite crear un bloque de datos en la memoria de carga y/o en la memoria de trabajo.

La instrucción "CREATE_DB" no modifica la suma de verificación del programa de usuario.

Número del bloque de datos

Al bloque de datos generado se le asigna un número del área definida en los parámetros LOW_LIMIT (límite inferior) y UP_LIMIT (límite superior). "CREATE_DB" asigna al DB el número más bajo posible del área especificada. No se pueden asignar los números de los DBs ya contenidos en el programa de usuario.

Para crear un DB con un número determinado, asigne el mismo número al límite superior y al límite inferior del área especificada. Si ya existe en la memoria de trabajo y/o en la memoria de carga un DB con el mismo número, o si el DB existe ya como variante copiada, la instrucción finaliza y se devuelve un mensaje de error en el parámetro RET_VAL.

Valores de arranque del bloque de datos

Con el parámetro SRCBLK se definen los valores de arranque del DB que va a generarse. El parámetro SRCBLK es un puntero hacia un DB o un área DB del o de la que se toman los valores de arranque. El DB direccionado en el parámetro SRCBLK ha tenido que crearse con acceso estándar (atributo "Acceso optimizado al bloque" desactivado).

- Si el área indicada en el parámetro SRCBLK es mayor que el DB generado, los valores hasta la longitud completa del DB generado se consideran valores de arranque.
- Si el área indicada en el parámetro SRCBLK es menor que el DB generado, los valores restantes se rellenan con "0".

Para garantizar la coherencia de los datos, no se debe modificar esa área de datos durante el procesamiento de "CREATE_DB" (es decir, mientras el parámetro BUSY tenga el valor TRUE).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CREATE_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de creación del DB
LOW_LIMIT	Input	UINT	I, Q, M, D, L o constante	Límite inferior del área de la cual "CREATE_DB" extrae el número que se asigna al DB (número de DB más bajo posible: 60000)
UP_LIMIT	Input	UINT	I, Q, M, D, L o constante	Límite superior del área de la cual "CREATE_DB" extrae el número que se asigna al DB (número de DB más alto posible: 60999)
COUNT	Input	UDINT	I, Q, M, D, L o constante	El valor de contaje indica el número de bytes que se desea reservar para el DB creado. El número de bytes debe ser un número par. La longitud máxima es de 65.534 bytes.
ATTRIB	Input	BYTE	I, Q, M, D, L o constante	Con los primeros 4 bits del byte en el parámetro ATTRIB se definen las propiedades del bloque de datos:
				• Bit 0 = 0: DB solo en la memoria de trabajo • Bit 0 = 1: DB solo en la memoria de carga
				• Bit 1 = 0: el DB no tiene activada la protección contra escritura • Bit 1 = 1: DB protegido contra escritura
				• Bit 2 = 0: DB remanente (solo DBs creados en la memoria de carga) • Bit 2 = 1: DB no remanente
				• Bit 3= 0: creación del DB en la memoria de carga o en la memoria de trabajo • Bit 3= 1: creación del DB en la memoria de carga y en la memoria de trabajo Por motivos de compatibilidad con STEP7 V5.x, los bits 1 y 3 deben utilizarse combinados:

Bit 0	Bit 3	Creación de DB
0	0	Solo en la memoria de trabajo
1	0	Solo en la memoria de carga
0	1	Memoria de trabajo y memoria de carga
1	1	Memoria de trabajo y memoria de carga

				• Bit 4 = 0: valores de arranque sin predefinir (los valores de entrada en el parámetro SRCBLK se ignoran). • Bit 4 = 1: predefinir valores de arranque (los valores corresponden al DB direccionado mediante el parámetro SRCBLK).
SRCBLK	Input	VARIANT	D	Puntero hacia el bloque de datos con cuyos valores se inicializará el bloque de datos que se va a crear.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la operación todavía no ha finalizado.
DB_NUM	Output	DB_DYN (UINT)	I, Q, M, D, L	Número del DB creado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Sin errores
0081	El área de destino es mayor que el área de origen. El área de origen se escribe por completo en el área de destino, y el resto de los bytes del área de destino se rellenan con 0.
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): transferencia de datos ya activa; BUSY tiene el valor "1"
8081	El área de origen es mayor que el área de destino. El área de destino se escribe por completo, el resto de bytes del área de origen se ignoran.
8092	La función "Generar bloque de datos" no puede ejecutarse en este momento, porque • está activa la función "Comprimir memoria de usuario"; • se ha alcanzado ya el número máximo de bloques admisible en la CPU
8093	En el parámetro SRCBLK no se ha indicado ningún bloque de datos o se ha indicado uno que no se encuentra en la memoria de trabajo.
8094	En el parámetro ATTRIB se ha indicado un atributo que todavía no se soporta.
80A1	Error en el número del DB: • el número es "0" • Límite inferior > límite superior
80A2	Error en la longitud del DB: • la longitud es "0" • la longitud es un número impar • la longitud es mayor que la admisible en la CPU
80A3	El bloque de datos del parámetro SRCBLK no se ha creado con acceso estándar.
80B1	No hay ningún número de DB libre.
80B2	Memoria de trabajo libre insuficiente.
80B4	La Memory Card tiene activada la protección contra escritura.
80BB	Memoria de carga libre insuficiente.
80C0	El destino está siendo procesado por otra instrucción o por una función de comunicación.
80C1	Se está borrando un DB con ese número de DB.

Código de error* (W#16#...)	Descripción
80C3	El número máximo de instrucciones "CREATE_DB" activas simultáneamente ya se ha alcanzado en este momento.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

READ_DBL: Leer de un bloque de datos de la memoria de carga

Descripción

La instrucción copia un DB o una parte de un DB que se encuentra en la memoria de carga (Micro Memory Card) en el área de datos de un DB de destino. El DB de destino tiene que ser relevante para la ejecución, es decir, no puede haberse creado con el atributo UNLINKED. El contenido de la memoria de carga no se modifica en la operación de copia.

Para garantizar la coherencia de los datos no debe modificarse el área de destino durante el procesamiento de "READ_DBL" (es decir, mientras el parámetro BUSY tenga el valor TRUE).

Para los parámetros SRCBLK y DSTBLK (bloque de origen y bloque de destino) se aplican las siguientes restricciones:

- La longitud del puntero VARIANT debe ser divisible entre 8.
- En el caso de un puntero VARIANT del tipo STRING la longitud debe ser 1.
- Los bloques de origen y de destino se deben haber creado con el mismo acceso al bloque, es decir, ambos deben utilizar el tipo de acceso "Optimizado" o "Estándar".

Nota

"READ_DBL" se procesa de forma asíncrona, por lo que no es adecuada para leer variables frecuentemente (o cíclicamente) en la memoria de carga.

Una petición iniciada siempre se finaliza. Si se alcanza el número máximo de instrucciones "READ_DBL" activas simultáneamente y en ese momento se vuelve a llamar "READ_DBL" en una clase de mayor prioridad, se devuelve el código de error W#16#80C3. Por este motivo no es útil volver a arrancar inmediatamente la petición de mayor prioridad.

Funcionamiento

La instrucción "READ_DBL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "READ_DBL" con REQ = 1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "READ_DBL":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de lectura
SRCBLK	Input	VARIANT	D	Puntero al bloque de datos de la memoria de carga de la que se lee
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la lectura todavía no ha finalizado.
DSTBLK	Output	VARIANT	D	Puntero al bloque de datos de la memoria de trabajo en la que se tiene que escribir

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Ningún error
0081	El área de destino es mayor que el área de origen. El área de origen se escribe completamente en el área de destino, el resto de bytes del área de destino no se modifica.
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".
8081	El área de origen es mayor que el área de destino. El área de destino se escribe por completo, el resto de bytes del área de origen se ignoran.
8082	El DB de destino y el DB de origen son de tipos diferentes (acceso optimizado/acceso estándar).
8093	En el parámetro DSTBLK no se ha indicado ningún bloque de datos o se ha indicado uno que no se encuentra en la memoria de trabajo.
80B1	En el parámetro SRCBLK no se ha indicado ningún bloque de datos o el bloque de datos indicado no es ningún objeto de memoria de carga.
80B4	El DB con atributo F no debe leerse.
80C3	El número máximo de instrucciones "READ_DBL" activas simultáneamente ya se ha alcanzado en este momento.
80C0	El DB de destino está siendo procesado por otra instrucción o función de comunicación.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 4460)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga

Descripción

La instrucción "WRIT_DBL" permite transferir el contenido de un DB o un área de DB desde la memoria de trabajo a un DB o un área de DB de la memoria de carga (Micro Memory Card). El DB de origen tiene que ser relevante para la ejecución, es decir, no puede haberse creado con el atributo UNLINKED.

Para garantizar la coherencia de los datos no debe modificarse el área de origen durante el procesamiento de "WRIT_DBL" (es decir, mientras el parámetro BUSY tenga el valor TRUE).

Para los parámetros SRCBLK y DSTBLK (bloque de origen y bloque de destino) se aplican las siguientes restricciones:

- En el caso de un puntero VARIANT del tipo BOOL la longitud debe ser divisible entre 8.
- En el caso de un puntero VARIANT del tipo STRING la longitud debe ser 1.
- Los bloques de origen y de destino se deben haber creado con el mismo acceso al bloque, es decir, ambos deben utilizar el tipo de acceso "Acceso optimizado al bloque" o "Estándar: compatible con S7-300/400".

La instrucción "WRIT_DBL" no modifica la suma de verificación del programa de usuario al describir un DB creado con una instrucción. Por el contrario, al describir un DB cargado, la primera escritura en este DB sí modifica la suma de verificación del programa de usuario.

Nota

"WRIT_DBL" no es adecuado para escribir variables frecuentemente (o cíclicamente) en la memoria de carga, ya que por limitaciones tecnológicas una Memory-Card solo tiene un determinado número de accesos de escritura.

Funcionamiento

La instrucción "WRIT_DBL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "WRIT_DBL" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WRIT_DBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: solicitud de escritura
SRCBLK	Input	VARIANT	D	Puntero en el bloque de datos de la memoria de trabajo de la que se lee
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la escritura todavía no ha finalizado.
DSTBLK	Output	VARIANT	D	Puntero al bloque de datos de la memoria de carga en la que se tiene que escribir

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Descripción
0000	Sin errores
0081	El área de destino es mayor que el área de origen. El área de origen se escribe completamente en el área de destino, el resto de bytes del área de destino no se modifica.
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".
8081	El área de origen es mayor que el área de destino. El área de destino se escribe por completo, el resto de bytes del área de origen se ignoran.
8082	El DB de destino y el DB de origen son de dos tipos diferentes (acceso optimizado/no optimizado).
8092	Modo de operación incorrecto: Mientras "WRIT_DBL" estaba activo, la CPU ha pasado a STOP. Cuando vuelva a pasar a RUN, se proporcionará este código de error. Vuelva a llamar "WRIT_DBL".
8093	En el parámetro SRCBLK no se ha indicado ningún bloque de datos o se ha indicado uno que no se encuentra en la memoria de trabajo.
80B1	En el parámetro DSTBLK no se ha indicado ningún bloque de datos o el bloque de datos indicado no es ningún objeto de memoria de carga.
80B4	• El DB con atributo F no debe leerse. • La Memory Card tiene activada la protección contra escritura.
80BB	No hay suficiente memoria de carga disponible.
80C3	El número máximo de instrucciones "WRIT_DBL" activas simultáneamente ya se ha alcanzado en este momento.
80C0	El DB de destino está siendo procesado por otra instrucción o función de comunicación.
Códigos de error generales	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

ATTR_DB: Leer atributos de un bloque de datos

Descripción

La instrucción "ATTR_DB" permite obtener información sobre un bloque de datos (DB) residente en la memoria de trabajo de la CPU. La instrucción determina para el DB seleccionado los atributos especificados en el parámetro ATTRIB.

No se puede leer la longitud de bloques de datos de acceso optimizado; el parámetro DB_LENGH tiene la longitud "0" en los DB de acceso optimizado.

No se pueden leer bloques de datos para Motion Control con la instrucción "ATTR_DB". Para esos bloques de datos se devuelve el código de error 80B2.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ATTR_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ = 1: petición de lectura de los atributos del bloque
DB_NUMBER	Input	DB_ANY (UINT)	I, Q, M, D, L o constante	Número del DB que se va a probar
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
DB_LENGTH	Output	UDINT	I, Q, M, D, L	Número de bytes de datos que contiene el DB seleccionado.
ATTRIB	Output	BYTE	I, Q, M, D, L	Propiedades del DB: • Bit 0* = 0: LINKED: el DB se encuentra solo en la memoria de trabajo • Bit 0* = 1: UNLINKED: el DB se encuentra solo en la memoria de carga • Bit 1 = 0: READ_ONLY: el DB no tiene activada la protección contra escritura. • Bit 1 = 1: READ_ONLY: el DB es de solo lectura. • Bit 2 = 0: RETAIN: el DB es remanente. • Bit 2 = 1: NON_RETAIN: el DB no es remanente. • Bit 3* = 0: El DB se encuentra en la memoria de carga o en la memoria de trabajo • Bit 3* = 1: El DB se encuentra en la memoria de carga y en la memoria de trabajo
* La relación entre el bit 0 y el bit 3 se explica en los parámetros de la instrucción "CREATE_DB: Generar bloque de datos (Página 4994)".				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
80A1	Error en el parámetro de entrada DB_NUMBER: el parámetro actual seleccionado • tiene el valor "0" • es mayor que el número de DB máximo posible en la CPU utilizada.
80B1	El DB con el número indicado no está cargado en la CPU.
80B2	No se puede acceder a los DB de los objetos tecnológicos Motion Control con la instrucción "ATTR_DB".
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

DELETE_DB: Borrar bloque de datos

Descripción

Con la instrucción "DELETE_DB" se borra un bloque de datos (DB) que no haya sido creado llamando la instrucción "CREATE_DB (Página 4994)" desde el programa de usuario.

Si el bloque de datos no se ha creado con "CREATE_DB", el parámetro RET_VAL devuelve el código de error W#16#80B5.

El DB que se va a borrar no debe encontrarse ni en el nivel de ejecución actual ni en otro de menor prioridad. De lo contrario, al llamar a la instrucción "DELETE_DB", la CPU inicia el OB 121. Si el OB 121 no está cargado, la CPU pasa a STOP. La instrucción "DELETE_DB" puede ser interrumpida por niveles de ejecución de mayor prioridad. Si se vuelve a llamar a la instrucción allí, esa segunda llamada se interrumpe y en RET_VAL se registra W#16#8091.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DELETE_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	REQ =1: Solicitud de borrado del DB con número en el parámetro DB_NUMBER
DB_NUMBER	Input	UINT	I, Q, M, D, L o constante	Número del DB que se va a borrar
RET_VAL	Output	INT	I, Q, M, D, L	Información de error (véase "Parámetro RET_VAL")
BUSY	Output	BOOL	I, Q, M, D, L	BUSY =1: La operación todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8091	En llamadas de "DELETE_DB" anidadas se ha superado la profundidad de anidamiento máxima de la CPU utilizada.
8092	La instrucción no puede ejecutarse en este momento, porque: • está activa la función "Comprimir memoria de usuario"; • el DB que se debe borrar se está copiando de la CPU a un proyecto offline;
80A1	Error en el parámetro de entrada DB_NUMBER: • El valor del parámetro es "0". • El valor del parámetro es mayor que el número de DB máximo posible en la CPU utilizada.
80B1	El DB con el número indicado no está cargado en la CPU.
80B4	El DB no se puede borrar porque la Memory Card de la CPU tiene activada la protección contra escritura.
80B5	El DB no se creó con "CREATE_DB" .
80BB	Memoria de carga libre insuficiente.
80C1	La función "Borrar un DB" no puede ejecutarse en este momento debido a escasez temporal de recursos.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Direccionamiento

Instrucciones para convertir direcciones

Descripción

Existen distintas posibilidades de direccionar un módulo (dirección ES, ID de hardware, slot).

Los datos de la dirección se pueden convertir con las siguientes instrucciones:

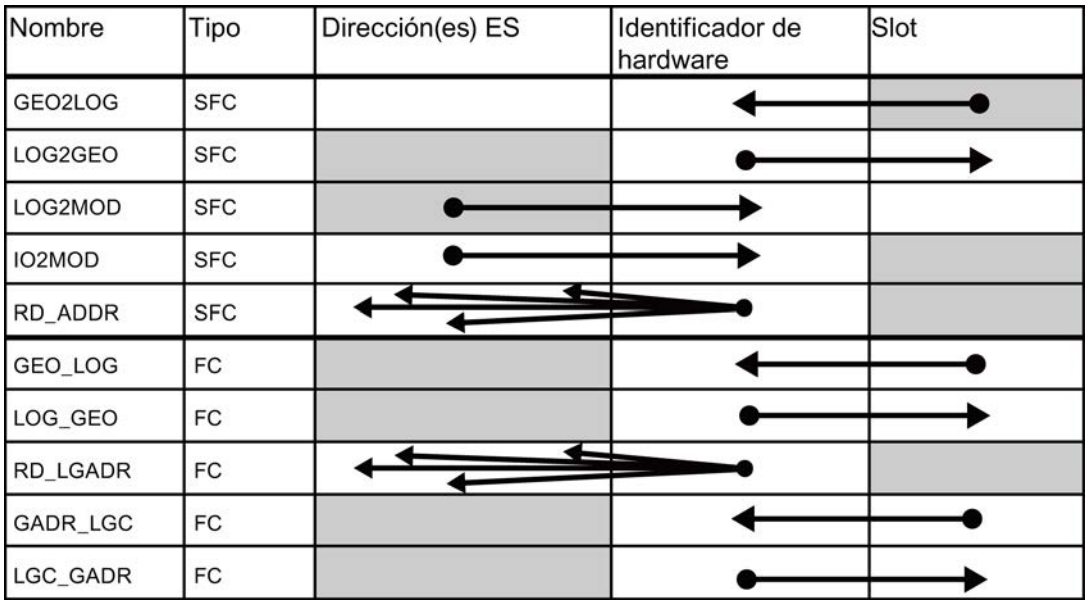
- GEO2LOG: Determinar la ID de hardware a partir del slot (Página 5004)
- LOG2GEO: Determinar el slot a partir de la ID de hardware (Página 5006)
- LOG2MOD: Determinar la ID de hardware a partir del direccionamiento de STEP 7 V5.5 SPx (Página 5007)
- IO2MOD: Determinar la ID de hardware a partir de una dirección ES (Página 5008)
- RD_ADDR: Determinar las direcciones ES a partir de la ID de hardware (Página 5009)

En los proyectos migrados se permiten también las siguientes instrucciones:

- GEO_LOG: Determinar la ID de hardware a partir del slot (Página 5012)
- LOG_GEO: Determinar el slot a partir de la ID de hardware (Página 5013)
- RD_LGADR: Determinar las direcciones ES a partir de la ID de hardware (Página 5014)
- GADR_LGC: Determinar la ID de hardware a partir del slot y del offset del área de direcciones de datos de usuario. (Página 5016)
- LGC_GADR: Determinar el slot a partir de la ID de hardware (Página 5017)

Tipo de conversión de dirección

La figura siguiente muestra qué instrucción ejecuta qué conversión.



GEO2LOG: Determinar la ID de hardware a partir del slot

Descripción

Con la instrucción "GEO2LOG" se determina la ID de hardware a partir de los datos de slot que se definen mediante el tipo de datos de sistema GEOADDR.

En función del tipo de hardware que se defina en el parámetro HWTYPE se evaluará la siguiente información del resto de los parámetros de GEOADDR:

- Con HWTYPE = 1 (sistema PROFINET IO):
  - Solo se evaluará IOSYSTEM. El resto de los parámetros de GEOADDR no se tienen en cuenta.
  - Se muestra la ID de hardware del sistema PROFINET IO.
- Con HWTYPE = 2 (dispositivo PROFINET IO):
  - Se evaluarán IOSYSTEM y STATION. El resto de los parámetros de GEOADDR no se tienen en cuenta.
  - Se muestra la ID de hardware del dispositivo PROFINET IO.
- Si HWTYPE = 3 (rack):
  - Solo se evaluarán IOSYSTEM y STATION. El resto de los parámetros de GEOADDR no se tienen en cuenta.
  - Se muestra la ID de hardware del rack.
- Con HWTYPE = 4 (módulo):
  - Se evaluarán IOSYSTEM, STATION y SLOT. El parámetro SUBSLOT de GEOADDR no se tiene en cuenta.
  - Se muestra la ID de hardware del módulo.
- Con HWTYPE = 5 (submódulo):
  - Todos los parámetros son evaluados por GEOADDR.
  - Se muestra la ID de hardware del submódulo.

El parámetro AREA del tipo de datos de sistema GEOADDR no se evaluará.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GEO2LOG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
GEOADDR	Input	VARIANT	D, L	Puntero hacia la estructura del tipo de datos de sistema GEOADDR. El tipo de datos de sistema contiene los datos de slot a partir de los cuales se determina la ID de hardware. Consulte también: Tipo de datos de sistema GEOADDR (Página 5011)
RET_VAL	Return	INT	I, Q, M, D, L	Salida de la información de error.
LADDR	Output	HW_ANY	I, Q, M, D, L	ID de hardware del bloque o del módulo. El número se asigna automáticamente y se guarda en las propiedades de la configuración hardware.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	No se ha producido ningún error.
8091	Valor no válido en GEOADDR para HWTYPE.
8094	Valor no válido en GEOADDR para IOSYSTEM.
8095	Valor no válido en GEOADDR para STATION.
8096	Valor no válido en GEOADDR para SLOT.
8097	Valor no válido en GEOADDR para SUBSLOT.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones para convertir direcciones (Página 5003)

LOG2GEO: Determinar el slot a partir de la ID de hardware

Descripción

La instrucción "LOG2GEO" determina el slot de módulo correspondiente a una ID de hardware.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LOG2GEO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_ANY	I, Q, M, D, L o constante	ID de hardware del módulo cuyo slot desea determinar. La ID de hardware se asigna automáticamente y se guarda en las propiedades del módulo en la configuración hardware y en las constantes de sistema.
RET_VAL	Output	INT	I, Q, M, D, L	Salida de la información de error.
GEOADDR	InOut	VARIANT	D	Puntero hacia el tipo de datos de sistema GEOADDR. En el tipo de datos de sistema GEOADDR se escriben los datos de slot. Consulte también: Tipo de datos de sistema GEOADDR (Página 5011)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	No se ha producido ningún error.
8090	La dirección indicada en el parámetro LADDR no es válida.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

LOG2MOD: Determinar la ID de hardware a partir del direccionamiento de STEP 7 V5.5 SPx

Descripción

La instrucción "LOG2MOD" determina la ID de hardware para un (sub)módulo E/S a partir del direccionamiento de STEP 7 5.5 SPx (dirección de datos E/S o dirección de diagnóstico).

Distintas instrucciones utilizan la ID de hardware en el parámetro de entrada LADDR para el direccionamiento. Llamando previamente "LOG2MOD" se pueden convertir los parámetros de direccionamiento de STEP 7 5.5 SPx.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LOG2MOD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificación del área de direcciones como en STEP 7 5.5 SPx: • B#16#00: El bit 15 de ADDR indica si se trata de una dirección de entrada (bit 15 = 0) o de salida (bit15 = 1). • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ)
ADDR	Input	WORD	I, Q, M, D, L o constante	Dirección lógica de los datos E/S del módulo como offset (conforme al direccionamiento en STEP 7 5.5 SPx) o dirección de diagnóstico.
RET_VAL	Return	INT	I, Q, M, D, L	Código de error de la instrucción.
HWID	Output	HW_IO	I, Q, M, D, L	ID de hardware determinada (dirección lógica) del (sub)módulo E/S.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	No se ha producido ningún error.
8093	• Ningún componente de hardware utiliza la dirección indicada. • El valor indicado en el parámetro IOID no es válido.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

IO2MOD: Determinar la ID de hardware a partir de una dirección ES

Descripción

La instrucción "IO2MOD" determina la ID de hardware del módulo a partir de una dirección E/S (I, Q, PI, PQ) de un (sub)módulo.

En el parámetro ADDR se introduce la dirección E/S. Si se indican varias direcciones E/S en este parámetro, para determinar la ID de hardware solo se evaluará la primera dirección. Si se indica la primera dirección correctamente, la longitud carece de importancia a la hora de introducir la dirección en el parámetro ADDR. Incluso si se utiliza un área de direcciones que contenga varios módulos o direcciones sin utilizar, se puede determinar la ID de hardware del primer módulo.

Si en el parámetro ADDR no se indica ninguna dirección E/S de un (sub)módulo, el parámetro RET_VAL devuelve el código de error 8090.

Nota

Entrada de la dirección E/S en SCL

En SCL no se puede programar con el identificador de acceso a la periferia "%QWx:P". Utilice en este caso el nombre simbólico de la variable o bien la dirección absoluta en la memoria imagen de proceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "IO2MOD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ADDR	Input	VARIANT	I, Q, M, D, L	Dirección E/S (I, Q, PI, PQ) dentro de un (sub)módulo. Preste atención a que no se utilicen accesos segmentados (slice) para el parámetro ADDR. Si es el caso, el parámetro LADDR devuelve valores incorrectos.
RET_VAL	Return	INT	I, Q, M, D, L	Código de error de la instrucción.
LADDR	Output	HW_IO	I, Q, M, D, L	ID de hardware determinada (dirección lógica) del (sub)módulo E/S.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	No ha ocurrido ningún error.
8090	Ningún componente de hardware utiliza la dirección E/S indicada en el parámetro ADDR.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RD_ADDR: Determinar las direcciones ES a partir de la ID de hardware

Descripción

La instrucción "RD_ADDR" determina la longitud y la dirección inicial de las entradas o salidas a partir de la ID de hardware de un (sub)módulo.

- Mediante el parámetro LADDR se selecciona el módulo de entrada o de salida a partir de la ID de hardware.
- Dependiendo de si se trata de un módulo de entrada o de salida, se utilizan los siguientes parámetros de salida:
  - En el caso de un módulo de entrada, se devuelven los valores determinados en los parámetros PIADDR y PICOUNT.
  - En el caso de un módulo de salida, se devuelven los valores determinados en los parámetros PQADDR y PQCOUNT.

- Los parámetros PIADDR y PQADDR contienen respectivamente las direcciones iniciales de las entradas y de las salidas del módulo.
- Los parámetros PICOUNT y PQCOUNT contienen respectivamente el número de bytes de las entradas y salidas (1 byte con 8 entradas/salidas, 2 bytes con 16 entradas/salidas).

Nota

Direccionamiento de módulos comprimidos

Si se direccionan módulos comprimidos (no el primer módulo del grupo de módulos comprimidos), los datos indicados difieren de la configuración hardware. Para PIADDR o PQADDR y PICOUNT o PQCOUNT se devuelve "0". RET_VAL no indica ningún error (16#0000).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_ADDR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IO	I, Q, M, D, L o constante	ID de hardware del (sub)módulo.
RET_VAL	Return	INT	I, Q, M, D, L	Código de error de la instrucción.
PIADDR	Output	UDINT	I, Q, M, D, L	Dirección inicial del módulo de entrada.
PICOUNT	Output	UINT	I, Q, M, D, L	Número de bytes de las entradas.
PQADDR	Output	UDINT	I, Q, M, D, L	Dirección inicial del módulo de salida.
PQCOUNT	Output	UINT	I, Q, M, D, L	Número de bytes de las salidas.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0	No se ha producido ningún error.
8090	La ID de hardware del módulo del parámetro LADDR no es válida.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Tipo de datos de sistema GEOADDR

Dirección geográfica

El tipo de datos de sistema GEOADDR contiene la dirección geográfica de un módulo, es decir, los datos de slot.

- Dirección geográfica con PROFINET IO
  En PROFINET IO, la dirección geográfica estaba compuesta por la ID del sistema PROFINET IO, el número de dispositivo, el número de slot y el número de submódulo (en caso de que se usara un submódulo).
- Dirección geográfica con PROFIBUS DP
  En PROFIBUS DP, la dirección geográfica está compuesta por la ID del sistema maestro DP, el número de estación y el número de slot.

Los datos de slot de los módulos figuran en la configuración hardware del módulo correspondiente.

Tipo de datos de sistema GEOADDR

La estructura GEOADDR se crea automáticamente cuando se introduce "GEOADDR" como tipo de datos en un bloque de datos.

El tipo de datos de sistema GEOADDR tiene la siguiente estructura:

Nombre de parámetro	Tipo de datos	Descripción
GEOADDR	STRUCT	
HWTYPE	UINT	Tipo de hardware: • 1: Sistema IO (PROFINET/PROFIBUS) • 2: Dispositivo IO/esclavo DP • 3: Rack • 4: Módulo • 5: Submódulo Si el tipo de hardware no es compatible con la instrucción, HWTYPE devuelve "0".
AREA	UINT	Identificación de área: • 0 = módulo central • 1 = PROFINET IO • 2 = PROFIBUS DP • 3 = AS-i
IOSYSTEM	UINT	Sistema PROFINET IO (0 = aparato central en el rack)
STATION	UINT	• Número del rack, si la identificación de área AREA = 0 (módulo central). • Número de la estación, si la identificación de área AREA > 0.
SLOT	UINT	Número de slot
SUBSLOT	UINT	Número del submódulo. Si no está disponible o no se puede insertar ningún submódulo, este parámetro tiene el valor "0".

Otras

GEO_LOG: Determinar la ID de hardware a partir del slot

Descripción

Supongamos que se conoce el slot de módulo correspondiente de un módulo de señales. La instrucción "GEO_LOG" permite determinar a partir de ello la correspondiente ID de hardware del módulo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GEO_LOG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MASTER	Input	INT	I, Q, M, D, L o constante	Identificación de área: • 0 si el slot se encuentra en una estructura centralizada. • de 1 a 32: ID de sistema maestro DP del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFIBUS • de 100 a 115: ID de sistema PROFINET IO del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFINET
STATION	Input	INT	I, Q, M, D, L o constante	• Si MASTER = 0: Número de rack • Si MASTER > 0: Número de estación del aparato de campo
SLOT	Input	INT	I, Q, M, D, L o constante	Número de slot
SUBSLOT	Input	INT	I, Q, M, D, L o constante	La instrucción no evalúa el parámetro SUBSLOT.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
LADDR	Output	HW_IO	I, Q, M, D, L	ID de hardware del módulo

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8094	No está configurada ninguna subred con la SUBNETID indicada.
8095	Valor no válido en el parámetro STATION

Código de error* (W#16#...)	Explicación
8096	Valor no válido en el parámetro SLOT
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones para convertir direcciones (Página 5003)

LOG_GEO: Determinar el slot a partir de la ID de hardware

Descripción

La instrucción "LOG_GEO" determina el slot de módulo correspondiente a una ID de hardware.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LOG_GEO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	HW_IO	I, Q, M, D, L o constante	ID de hardware del módulo cuyo slot se debe determinar.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
AREA	Output	INT	I, Q, M, D, L	Identificación de área: Indica cómo deben interpretarse los parámetros de salida restantes: • 0: Aparato central • 2: PROFIBUS DP/PROFINET IO
MASTER	Output	INT	I, Q, M, D, L	Con AREA = 0: • 0: Si el slot se encuentra en uno de los racks (aparato central). Con AREA = 2: • de 1 a 32: ID de sistema maestro DP del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFIBUS • de 100 a 115: ID de sistema PROFINET IO del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFINET

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATION	Output	INT	I, Q, M, D, L	• Con MASTER = 0: Número de rack • Con MASTER > 0: Número de estación del aparato de campo
SLOT	Output	INT	I, Q, M, D, L	Número de slot
SUBSLOT	Output	INT	I, Q, M, D, L	La instrucción no devuelve el parámetro SUBSLOT (siempre "0").
OFFSET	Output	INT	I, Q, M, D, L	La instrucción no devuelve el parámetro OFFSET (siempre "0").

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	La dirección lógica indicada no es válida
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones para convertir direcciones (Página 5003)

RD_LGADR: Determinar las direcciones ES a partir de la ID de hardware

Descripción

La instrucción "RD_LGADR" permite determinar a partir de la ID de hardware las direcciones lógicas de un módulo, de un submódulo central o de un submódulo en PNIO.

- La ID de hardware del submódulo se indica en el parámetro LADDR.
- Las direcciones se escriben en los parámetros PEADDR y PAADDR en orden ascendente.
  - En un módulo de entradas solo se escribe el parámetro PEADDR. En un módulo de salidas el parámetro PAADDR.
  - Para guardar las direcciones se debe utilizar un Array of WORD, respectivamente.
- El número de direcciones se emite por los parámetros PECOUNT (en un módulo de entradas) y PACOUNT (en un módulo de salidas).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_LGADR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ)
LADDR	Input	HW_ANY	I, Q, M, D, L o constante	ID de hardware del módulo o del submódulo.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
PEADDR	Output	ANY	I, Q, M, D, L	Campo para las direcciones de entradas de la periferia del tipo Array of WORD
PECOUNT	Output	INT	I, Q, M, D, L	Número de direcciones PI devueltas
PAADDR	Output	ANY	I, Q, M, D, L	Campo para las direcciones de salidas de la periferia del tipo Array of WORD
PACOUNT	Output	INT	I, Q, M, D, L	Número de direcciones PQ devueltas

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	La dirección lógica indicada no es válida o se ha usado un valor no admisible en el parámetro IOID
80A0	Error en el parámetro de salida PEADDR: El tipo de datos de los elementos del campo no es WORD.
80A1	Error en el parámetro de salida PAADDR: El tipo de datos de los elementos del campo no es WORD.
80A2	Error en el parámetro de salida PEADDR: El campo indicado no ha podido alojar todas las direcciones lógicas.
80A3	Error en el parámetro de salida PAADDR: El campo indicado no ha podido alojar todas las direcciones lógicas.
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones para convertir direcciones (Página 5003)

GADR_LGC: Determinar la ID de hardware a partir del slot y del offset del área de direcciones de datos de usuario.

Descripción

La instrucción "GADR_LGC" permite determinar la ID de hardware de un módulo de señales. La ID de hardware se determina a partir del slot del módulo y del offset del área de direcciones de datos de usuario del módulo.

Si se aplica la instrucción "GADR_LGC" a módulos de potencia o módulos con direcciones comprimidas (ET 200S), se devuelve la dirección de diagnóstico.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GADR_LGC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SUBNETID	Input	BYTE	I, Q, M, D, L o constante	Identificación de área: • 0: Si el slot se encuentra en el módulo central • de 1 a 32: ID de sistema maestro DP del sistema de periferia descentralizada correspondiente, si el slot se encuentra en un sistema de periferia descentralizada • de 100 a 115: ID del sistema PROFINET IO del dispositivo de campo correspondiente si el slot se encuentra en un dispositivo de campo conectado a PROFINET
RACK	Input	WORD	I, Q, M, D, L o constante	• Número del rack, si la identificación de área es 0 • Número de estación del sistema de periferia descentralizada, si la identificación de área > 0
SLOT	Input	WORD	I, Q, M, D, L o constante	Número de slot
SUBSLOT	Input	BYTE	I, Q, M, D, L o constante	Slot de submódulo (si no puede insertarse ningún módulo, debe indicarse aquí 0)
SUBADDR	Input	WORD	I, Q, M, D, L o constante	Offset en el área de direcciones de datos útiles del módulo
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
IOID	Output	BYTE	I, Q, M, D, L	El parámetro de salida IOID no se describe (siempre "0").
LADDR	Output	HW_MODULE	I, Q, M, D, L	ID de hardware del módulo

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8093	Valor no válido en el parámetro SUBNETID
8094	No está configurada ninguna subred con la SUBNETID indicada.
8095	Valor no válido en el parámetro RACK.
8096	Valor no válido en el parámetro SLOT.
8097	Valor no válido en el parámetro SUBSLOT.
8098	Valor no válido en el parámetro SUBADDR.
8099	El slot no está configurado.
809A	La subdirección correspondiente al slot seleccionado no está configurada (posible solo con periferia central para CPU e IM).
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones para convertir direcciones (Página 5003)

LGC_GADR: Determinar el slot a partir de la ID de hardware

Descripción

La instrucción "LGC_GADR" determina el slot de módulo correspondiente a una ID de hardware.

Nota

No es posible aplicar la instrucción "LGC_GADR" a un módulo con direcciones comprimidas (ET 200S).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LGC_GADR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	No se evalúa.
LADDR	Input	HW_MODULE	I, Q, M, D, L o constante	ID de hardware del módulo

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
AREA	Output	BYTE	I, Q, M, D, L	Identificación de área: Indica cómo deben interpretarse los parámetros de salida restantes: • 0: Módulo central • 2: PROFIBUS DP
RACK	Output	WORD	I, Q, M, D, L	Número de rack: • En módulo central (AREA = 0): – Número de rack • En PROFIBUS DP (AREA = 2): – Byte menos significativo: Número de estación – Byte más significativo: ID del sistema maestro DP
SLOT	Output	WORD	I, Q, M, D, L	Número de slot: • En módulo central (AREA = 0): – Número de slot • En PROFIBUS DP (AREA = 2): – Número de slot en la estación
SUBADDR	Output	WORD	I, Q, M, D, L	No se devuelve (siempre "0").

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error* (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	La dirección lógica indicada no es válida o se ha usado un valor no admisible en el parámetro IOID
8093	Esta instrucción no es válida para el módulo seleccionado mediante los parámetros IOID y LADDR .
Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones para convertir direcciones (Página 5003)

9.8.1.4 Tecnología

Motion Control

S7-1200 Motion Control

S7-1200 Motion Control V4

MC_Power

MC_Power: Habilitar, bloquear ejes V4

Descripción

La instrucción de Motion Control "MC_Power" habilita o bloquea un eje.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- No hay ningún error que impida la habilitación.

Comportamiento de relevo

El procesamiento del "MC_Power" no puede ser cancelado por ninguna petición de Motion Control.

Al bloquear el eje (parámetro de entrada "Enable" = FALSE) se cancelan todas las peticiones de Motion Control conforme al "StopMode" seleccionado en el respectivo objeto tecnológico.

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis	-	Objeto tecnológico del eje	
Enable	INPUT	BOOL	FALSE	TRUE	El eje es habilitado.
				FALSE	Todas las peticiones en curso se cancelan conforme al "StopMode" parametrizado. El eje se detiene y se bloquea.
StopMode	INPUT	INT	0	0	Parada de emergencia Si existe una demanda de bloqueo del eje, éste frena con la deceleración de parada de emergencia configurada. Cuando el eje se para, se bloquea.

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
				1	Desconexión inmediata Si existe una demanda de bloqueo del eje, ésta se bloquea sin deceleración. La salida de impulsos se detiene inmediatamente.
				2	Parada de emergencia con limitación de tirones Si existe una demanda de bloqueo del eje, éste frena con la deceleración de parada de emergencia configurada. Si la limitación de tirones está activada, se tiene en cuenta el tirón configurado. Cuando el eje se para, se bloquea.
Status	OUTPUT	BOOL	FALSE	Estado de la habilitación del eje	
				FALSE	El eje está bloqueado El eje no ejecuta ninguna petición de Motion Control y no acepta nuevas peticiones (excepción: Petición MC_Reset). El eje no está referenciado. Al bloquear, el estado cambia a FALSE en cuanto el eje se para.
				TRUE	El eje está habilitado El eje está listo para ejecutar peticiones de Motion Control. Al habilitar el eje, el estado cambia a TRUE en cuanto se produce la señal "Accionamiento listo". Si no se ha configurado la señal del accionamiento "Accionamiento listo" en la configuración del eje, el estado cambia inmediatamente a TRUE.
Busy	OUTPUT	BOOL	FALSE	TRUE	"MC_Power" está activo.
Error	OUTPUT	BOOL	FALSE	TRUE	En la instrucción de Motion Control "MC_Power" o el objeto tecnológico correspondiente se ha producido un error. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Nota

Si el eje se desconecta a causa de un error, se vuelve a habilitar automáticamente tras eliminar y acusar el error. Requisito para ello es que el parámetro de entrada "Enable" mantenga el valor TRUE durante esta operación.

Habilitación de un eje con señales del accionamiento configuradas

Para habilitar un eje, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Configure el parámetro de entrada "StopMode" con el valor deseado. Ponga el parámetro de entrada "Enable" en TRUE.
   La salida de habilitación para "Habilitar accionamiento" cambia a TRUE para habilitar la potencia del accionamiento. La CPU aguarda la señal "Accionamiento listo" del accionamiento.
   El eje queda habilitado en cuanto se emite la señal "Accionamiento listo" en la entrada de disponibilidad configurada de la CPU. El parámetro de salida "Status", así como la variable del objeto tecnológico <Nombre del eje>.StatusBits.Enable muestran el valor TRUE.

Habilitación de un eje sin señales del accionamiento configuradas

Para habilitar un eje, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Configure el parámetro de entrada "StopMode" con el valor deseado. Ponga el parámetro de entrada "Enable" en TRUE. El eje es habilitado. El parámetro de salida "Status", así como la variable del objeto tecnológico <Nombre del eje>.StatusBits.Enable muestran el valor TRUE.

Bloqueo de un eje

Para bloquear un eje puede proceder de la siguiente manera:

1. Detenga el eje.
   La variable del objeto tecnológico <Nombre del eje:>.StatusBits.StandStill permite reconocer el momento de parada del eje.
2. Cambie el parámetro de entrada "Enable" a FALSE en cuanto se produzca la parada.
3. Si los parámetros de salida "Busy" y "Status", así como la variable del objeto tecnológico <Nombre del eje>.StatusBits.Enable muestran el valor FALSE, el bloqueo del eje de habrá finalizado.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_Power: Diagrama de funcionamiento V4 (Página 5022)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

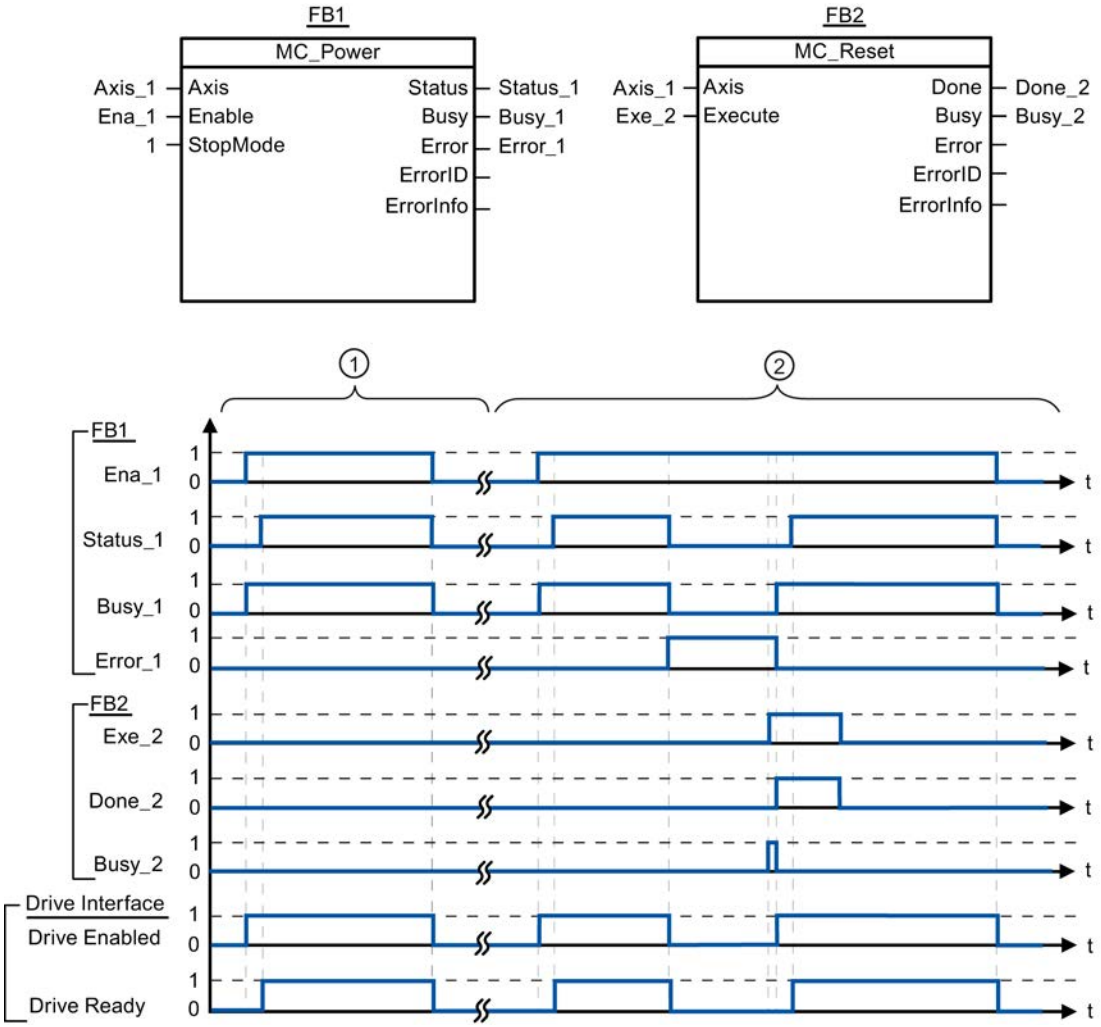
MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_Power: Diagrama de funcionamiento V4

Diagrama de funcionamiento



①	Se habilita un eje y a continuación se vuelve a bloquear. En cuanto el accionamiento devuelve la señal "Accionamiento listo" a la CPU se puede verificar el éxito de la habilitación a través de "Status_1".
②	Una vez habilitado un eje se produce un error, provocando el bloqueo del eje. El error se soluciona y se acusa con "MC_Reset". A continuación se vuelve a habilitar el eje.

Consulte también

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset

MC_Reset: Acusar errores V4

Descripción

La instrucción de Motion Control "MC_Reset" permite acusar "Errores de funcionamiento con parada del eje" y "Errores de configuración". En la "Lista de ErrorIDs y ErrorInfos", en el apartado "Ayuda", puede consultar los errores que deben ser acusados.

Es posible cargar la configuración del eje en la memoria de trabajo después de una carga en el estado operativo RUN.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- En caso de un error de configuración acusable debe haberse eliminado la causa (p. ej. la aceleración en el objeto tecnológico Eje de posicionamiento ha sido cambiada a un valor válido).

Comportamiento de relevo

La petición MC_Reset no puede ser cancelada por ninguna otra petición de Motion Control.

La nueva petición MC_Reset no cancela ninguna otra petición de Motion Control en curso.

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Restart	INPUT	BOOL	FALSE	TRUE	Carga la configuración del eje en la memoria de trabajo desde la memoria de carga. La petición solo puede ejecutarse con el eje bloqueado. Para ello, tenga en cuenta las indicaciones referentes a la Carga en la CPU (Página 10502).

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
				FALSE	Acusa errores pendientes
Done	OUTPUT	BOOL	FALSE	TRUE	El error se ha acusado.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Acusar un error con MC_Reset

Para acusar un error proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Inicie el acuse del error mediante un flanco ascendente en el parámetro de entrada "Execute".
3. El error ha sido acusado si el parámetro de entrada "Done" muestra el valor TRUE y la variable del objeto tecnológico <Nombre del eje:>.StatusBits.Error el valor FALSE.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

Cargar en la CPU (Página 10502)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_Home

MC_Home: Referenciar ejes, fijar punto de referencia V4

Descripción

La instrucción de Motion Control "MC_Home" permite contrastar la coordenada axial con la posición física real del accionamiento. Si el eje debe posicionarse de forma absoluta, es necesaria la referenciación. Pueden ejecutarse los siguientes tipos de referenciación:

- Referenciación activa (Mode = 3)
  La aproximación al punto de referencia se ejecuta automáticamente.
- Referenciación pasiva (Mode = 2)
  En la referenciación pasiva, la instrucción de Motion Control "MC_Home" no realiza ningún movimiento de referencia. El desplazamiento necesario para ello debe ser realizado por el usuario con otras instrucciones de Motion Control. El eje se referencia al detectarse el sensor del punto de referencia.
- Referenciación directa absoluta (Mode = 0)
  La posición actual del eje se fija con el valor del parámetro "Position".
- Referenciación directa relativa (Mode = 1)
  La posición actual del eje se desplaza en la cuantía del valor del parámetro "Position".

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- El eje está habilitado.
- Al iniciar con Mode = 0, 1 y 2 no puede haber ninguna petición MC_CommandTable activa.

Comportamiento de relevo

El comportamiento de relevo depende del modo seleccionado:

Mode = 0, 1

La petición MC_Home no puede ser cancelada por ninguna otra petición de Motion Control.

La petición MC_Home no cancela ninguna petición de Motion Control en curso. Las peticiones de desplazamiento con referencia de posición continúan, después de la referenciación, de acuerdo con la nueva posición de referenciación (valor en el parámetro de entrada: "Position").

Mode = 2

La petición MC_Home puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 2, 3

La nueva petición MC_Home cancela la siguiente petición de Motion Control en curso:

- Petición MC_Home Mode = 2

Las peticiones de desplazamiento con referencia de posición continúan, después de la referenciación, de acuerdo con la nueva posición de referenciación (valor en el parámetro de entrada: "Position").

Mode = 3

La petición MC_Home puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_Home cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 2, 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetro

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Position	INPUT	REAL	0.0	• Mode = 0, 2 y 3 Posición absoluta del eje una vez finalizado el proceso de referenciación • Mode = 1 Valor de corrección para la posición actual del eje Límites: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$	
Mode	INPUT	INT	0	Modo de toma de referencia	
				0	Referenciado directo (absoluto) La nueva posición del eje es el valor de posición del parámetro "Position".

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
				1	Referenciado directo (relativo) La nueva posición del eje es la posición actual del eje + el valor de posición del parámetro "Position".
				2	Referenciación pasiva Referenciación conforme a la configuración del eje. Tras la referenciación se aplica el valor del parámetro "Position" como nueva posición del eje.
				3	Referenciación activa Aproximación al punto de referencia conforme a la configuración del eje. Tras la referenciación se aplica el valor del parámetro "Position" como nueva posición del eje.
Done	OUTPUT	BOOL	FALSE	TRUE	La petición ha finalizado
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Nota

La toma de referencia de un eje se pierde cuando se dan las siguientes condiciones:

- Bloqueo del eje por la instrucción de Motion Control "MC_Power"
- Cambio entre modo automático y control manual
- Al iniciar la referenciación activa. La toma de referencia del eje vuelve a estar presente una vez concluido el proceso de referenciación.
- Tras POWER OFF -> POWER ON de la CPU
- Tras un rearranque completo de la CPU (RUN-STOP -> STOP-RUN)

Referenciación de un eje

Para referenciar el eje, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Configure los valores con los parámetros de entrada necesarios e inicie la referenciación mediante un flanco ascendente en el parámetro de entrada "Execute"
3. La referenciación concluye cuando el parámetro de salida muestra "Done" y la variable del objeto tecnológico <Nombre del eje:>.StatusBits.HomingDone el valor TRUE.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_Halt

MC_Halt: Parar ejes V4

Descripción

La instrucción de Motion Control "MC_Halt" cancela todas las operaciones de movimiento y frena el eje con la deceleración configurada hasta pararlo por completo. La posición de parada no está definida.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_Halt puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_Halt cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_SpeedAxis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Done	OUTPUT	BOOL	FALSE	TRUE	Se ha alcanzado la velocidad cero
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_Halt: Diagrama de funcionamiento V4 (Página 5031)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

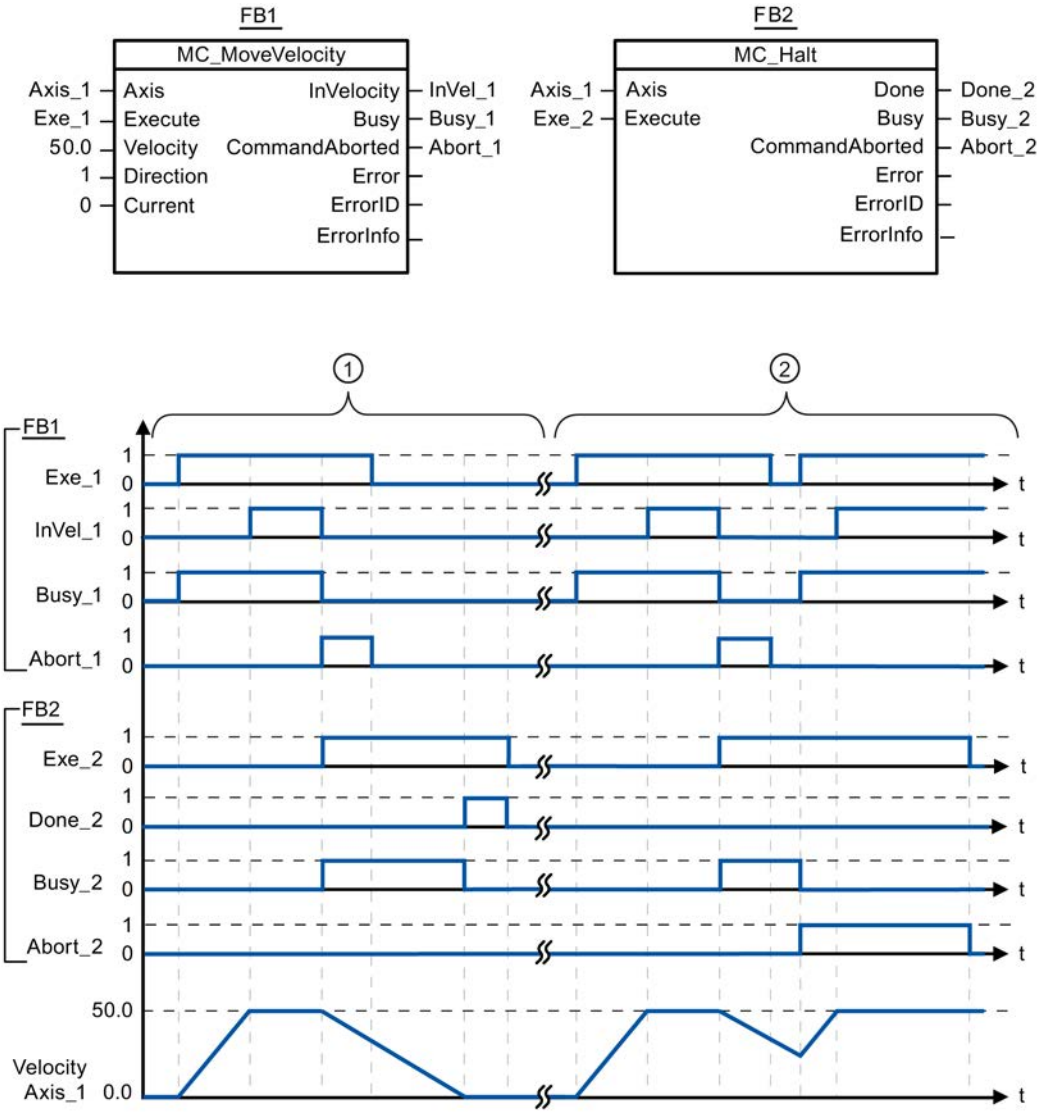
MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_Halt: Diagrama de funcionamiento V4

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 5.0

①	El eje se frena con una petición MC_Halt hasta pararse completamente. La parada completa se notifica mediante "Done_2".
②	Mientras una petición MC_Halt frena el eje, dicha petición es cancelada por otra petición de desplazamiento. La cancelación se notifica mediante "Abort_2".

Consulte también

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4

Descripción

La instrucción de Motion Control "MC_MoveAbsolute" inicia un movimiento de posicionamiento del eje respecto a una posición absoluta.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- El eje está habilitado.
- El eje está referenciado.

Comportamiento de relevo

La petición MC_MoveAbsolute puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveAbsolute cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Positioning Axis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Position	INPUT	REAL	0.0	Posición absoluta de destino Límites: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Velocidad del eje No siempre se alcanza esta velocidad, dependiendo de la aceleración y deceleración configuradas y de la posición de destino fijada. Límites: Velocidad de arranque/parada ≤ Velocity ≤ velocidad máxima	
Done	OUTPUT	BOOL	FALSE	TRUE	Posición absoluta de destino alcanzada
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_MoveAbsolute: Diagrama de funcionamiento V4 (Página 5034)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

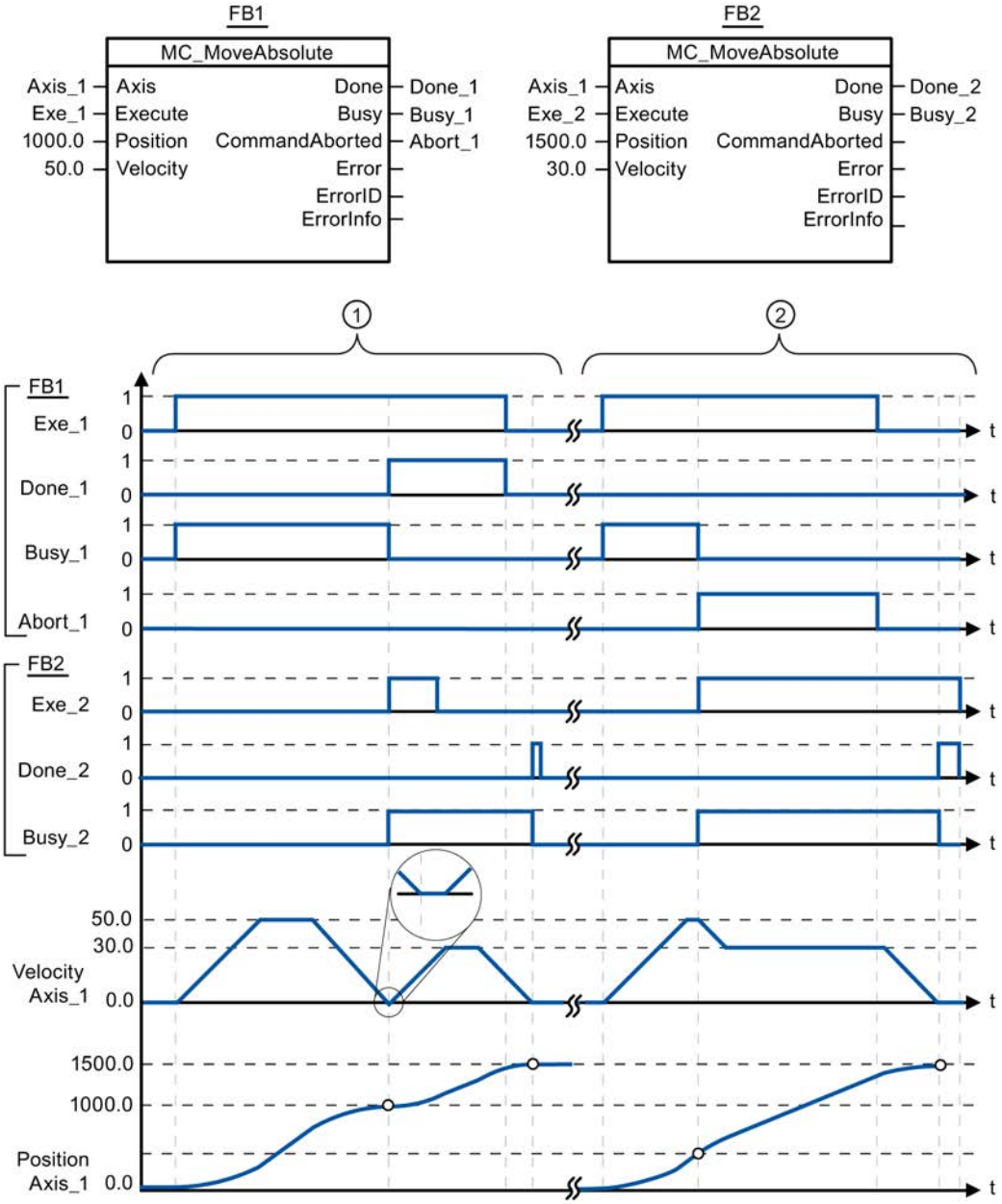
MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_MoveAbsolute: Diagrama de funcionamiento V4

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 10.0

①	Un eje es desplazado por una petición MC_MoveAbsolute a la posición absoluta 1000.0. En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_1". Con "Done_1" = TRUE se lanza otra petición MC_MoveAbsolute, con posición de destino 1500.0. Debido a los tiempos de reacción (p.ej. tiempo de ciclo del programa de usuario, ...) se produce una breve parada del eje (ver representación ampliada). En cuanto se alcanza con éxito la nueva posición de destino el sistema lo notifica mediante "Done_2".
②	Una petición MC_MoveAbsolute activa es cancelada por otra petición MC_MoveAbsolute. La cancelación se notifica mediante "Abort_1". A continuación, el eje se mueve con la nueva velocidad hasta la nueva posición de destino 1500.0. En cuanto se alcanza la nueva posición de destino el sistema lo notifica mediante "Done_2".

Consulte también

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative

MC_MoveRelative: Posicionar ejes de forma relativa V4

Descripción

La instrucción de Motion Control "MC_MoveRelative" inicia un movimiento de posicionamiento relativo respecto a la posición inicial.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_MoveRelative puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveRelative cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Positioning Axis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Distance	INPUT	REAL	0.0	Trayecto de posicionamiento Límites: $-1.0e^{12} \leq$ Distance $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Velocidad del eje No siempre se alcanza esta velocidad, dependiendo de la aceleración y deceleración configuradas y del trayecto fijado. Límites: Velocidad de arranque/parada ≤ Velocity ≤ velocidad máxima	
Done	OUTPUT	BOOL	FALSE	TRUE	Posición de destino alcanzada
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_MoveRelative: Diagrama de funcionamiento V4 (Página 5038)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

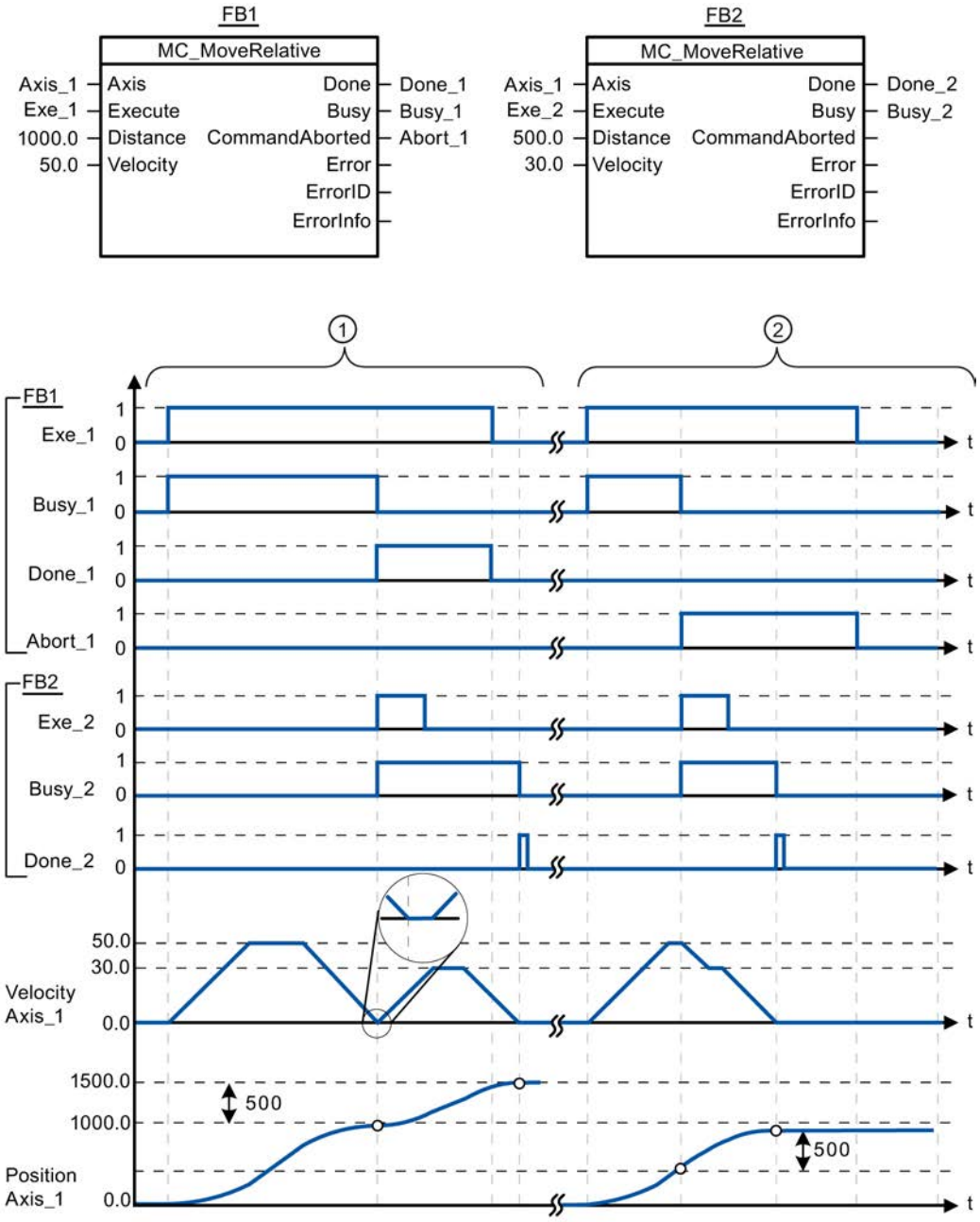
MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_MoveRelative: Diagrama de funcionamiento V4

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 10.0

①	Un eje es desplazado por una petición MC_MoveRelative en la distancia ("Distance") 1000.0. En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_1". Con "Done_1" = TRUE se lanza otra petición MC_MoveRelative con la distancia 500.0. Debido a los tiempos de reacción (p.ej. tiempo de ciclo del programa de usuario, ...) se produce una breve parada del eje (ver representación ampliada). En cuanto se alcanza con éxito la nueva posición de destino el sistema lo notifica mediante "Done_2".
②	Una petición MC_MoveRelative activa es cancelada por otra petición MC_MoveRelative. La cancelación se notifica mediante "Abort_1". A continuación, el eje se mueve con la nueva velocidad hasta recorrer el nuevo trayecto ("Distance") 500.0. En cuanto se alcanza la nueva posición de destino el sistema lo notifica mediante "Done_2".

Consulte también

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4

Descripción

La instrucción de Motion Control "MC_MoveVelocity" mueve el eje de forma constante a la velocidad preajustada.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_MoveVelocity puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveVelocity cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt

- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_SpeedAxis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Velocity	INPUT	REAL	10.0	Velocidad predeterminada para mover el eje Límites: Velocidad de inicio/parada ≤ |Velocity| ≤ velocidad máxima (Velocity = 0.0 está permitido)	
Direction	INPUT	INT	0	Preajuste de sentido	
				0	Sentido de giro según el signo del valor del parámetro "Velocity"
				1	Sentido de giro positivo (El signo del valor del parámetro "Velocity" se ignora)
				2	Sentido de giro negativo (El signo del valor del parámetro "Velocity" se ignora)
Current	INPUT	BOOL	FALSE	Mantener velocidad actual	
				FALSE	"Mantener velocidad actual" desactivado. Se utilizan los valores de los parámetros "Velocity" y "Direction".
				TRUE	"Mantener velocidad actual" activado. Los valores de los parámetros "Velocity" y "Direction" no se tienen en cuenta. En cuanto el eje vuelve a desplazarse a la velocidad actual, el parámetro "InVelocity" devuelve el valor TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	• "Current" = FALSE: La velocidad indicada en el parámetro "Velocity" ha sido alcanzada. • "Current" = TRUE: El eje se mueve a la velocidad actual en el momento de arranque.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Nota

PLCopen versión 2.0

La instrucción de Motion Control "MC_MoveVelocity" a partir de V4 es conforme con la versión 2.0 de PLCopen.

Los parámetros "InVelocity" y "Busy" muestran su estado con independencia del parámetro "Execute" hasta que la petición es relevada por otra o cancelada con un error. Tenga en cuenta también al respecto el capítulo Seguimiento de peticiones en curso (Página 10536).

Comportamiento con velocidad de consigna cero (Velocity = 0.0)

Una petición MC_MoveVelocity con "Velocity" = 0.0 cancela (al igual que una petición MC_Halt) las peticiones de movimiento activas y detiene el eje con la deceleración configurada.

Una vez alcanzada la parada se muestra en el parámetro de salida "InVelocity" el valor TRUE durante al menos un ciclo de programa.

"Durante el proceso de deceleración, Busy" indica el valor TRUE y lo cambia junto con "InVelocity" a FALSE. Si el parámetro "Execute" = TRUE, "InVelocity" y "Busy" se indican memorizados.

Cuando inicia la petición MC_MoveVelocity se activa el bit de estado "SpeedCommand" en el objeto tecnológico. El bit de estado "ConstantVelocity" con la parada del eje. Ambos bits se ajustan a la nueva situación con el inicio de una nueva petición de movimiento.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_MoveVelocity: Diagrama de funcionamiento V4 (Página 5043)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

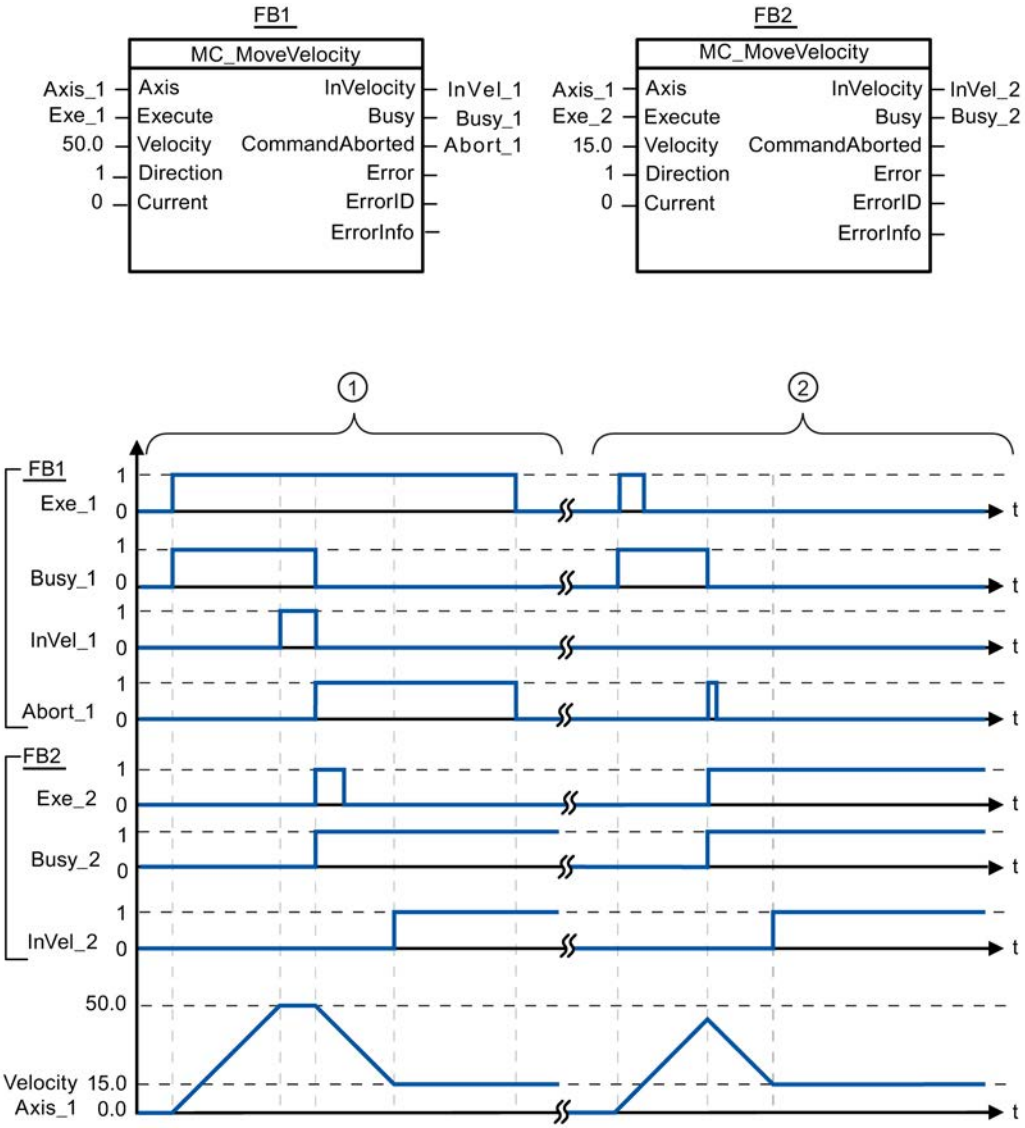
MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_MoveVelocity: Diagrama de funcionamiento V4

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 10.0

①	Una petición MC_MoveVelocity activa notifica el alcance de su velocidad consignada a través de "InVel_1". Seguidamente es cancelada por otra petición MC_MoveVelocity. La cancelación se notifica mediante "Abort_1". En cuanto se alcanza la nueva velocidad de destino 15.0 el sistema lo notifica mediante "InVel_2". A continuación el eje continúa moviéndose de forma constante a la nueva velocidad.
②	Una petición MC_MoveVelocity activa es cancelada por otra petición MC_MoveVelocity antes de alcanzar su velocidad consignada. La cancelación se notifica mediante "Abort_1". En cuanto se alcanza la nueva velocidad de destino 15.0 el sistema lo notifica mediante "InVel_2". A continuación el eje continúa moviéndose de forma constante a la nueva velocidad.

Consulte también

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog

MC_MoveJog: Mover ejes en el modo de operación Jog V4

Descripción

La instrucción de Motion Control "MC_MoveJog" mueve el eje de forma constante a la velocidad predeterminada en modo Jog. Utilice esta instrucción de Motion Control para, p.ej., fines de test y puesta en servicio.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_MoveJog puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveJog cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute

- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_SpeedAxis	-	Objeto tecnológico del eje	
JogForward	INPUT	BOOL	FALSE	Mientras el parámetro sea TRUE, el eje se moverá a la velocidad predeterminada en el parámetro "Velocity" en sentido positivo.	
JogBackward	INPUT	BOOL	FALSE	Mientras el parámetro sea TRUE, el eje se moverá a la velocidad predeterminada en el parámetro "Velocity" en sentido negativo.	
Si ambos parámetros son al mismo tiempo TRUE, el eje se detiene con la deceleración configurada. En los parámetros "Error", "ErrorID" y "ErrorInfo" se muestra un error.					
Velocity	INPUT	REAL	10.0	Preajuste de velocidad para el modo Jog. Límites: Velocidad de arranque/parada ≤ Velocity ≤ velocidad máxima	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	La velocidad indicada en el parámetro "Velocity" ha sido alcanzada.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_MoveJog: Diagrama de funcionamiento V4 (Página 5046)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

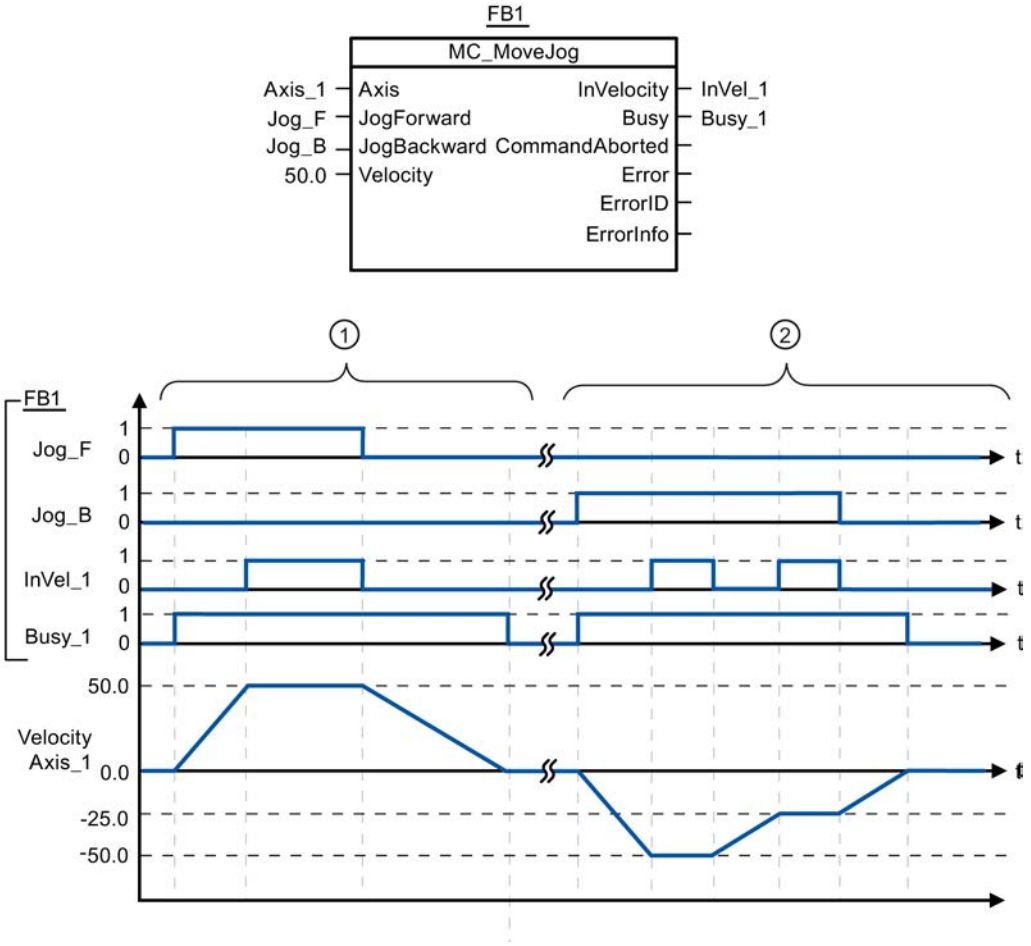
MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_MoveJog: Diagrama de funcionamiento V4

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 5.0

①	"Jog_F" permite mover el eje en sentido positivo en modo Jog. En cuanto se alcanza la velocidad de destino 50.0 el sistema lo notifica a través de "InVel_1". Tras desactivar "Jog_F" el eje frena hasta pararse del todo.
②	"Jog_B" permite mover el eje en sentido negativo en modo Jog. En cuanto se alcanza la velocidad de destino -50.0 el sistema lo notifica a través de "InVel_1". Si está activado "Jog_B", se modifica el valor del parámetro "Velocity" a 25.0. "InVel_1" se desactiva y el eje frena. En cuanto se alcanza la nueva velocidad de destino -25.0 el sistema lo notifica mediante "InVel_1". Tras desactivar "Jog_B" el eje frena hasta pararse del todo.

Consulte también

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4

Descripción

La instrucción de Motion Control "MC_CommandTable" reúne varias peticiones individuales de control de un eje en una secuencia de movimientos.

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha insertado y configurado correctamente.
- El objeto tecnológico Tabla de peticiones se ha insertado y configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_CommandTable puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_CommandTable cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La petición Motion Control en curso se cancela cuando se inicia la primera petición "Positioning Relative", "Positioning Absolute", "Velocity set point" o "Halt".

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_SpeedAxis	-	Objeto tecnológico del eje	
CommandTable	INPUT	TO_CommandTable	-	Objeto tecnológico de la tabla de peticiones	
Execute	INPUT	BOOL	FALSE	Inicio de la tabla de peticiones con flanco ascendente	
StartStep	INPUT	INT	1	Indicación que señala a partir de qué paso debe procesarse la tabla de peticiones. Límites: 1 ≤ StartStep ≤ EndStep	
EndStep	INPUT	INT	32	Indicación que señala hasta qué paso debe procesarse la tabla de peticiones. Límites: StartStep ≤ EndStep ≤ 32	
Done	OUTPUT	BOOL	FALSE	TRUE	La tabla de peticiones se ha procesado correctamente
Busy	OUTPUT	BOOL	FALSE	TRUE	La tabla de peticiones se encuentra en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La tabla de peticiones ha sido cancelada por otra petición durante el procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Durante el procesamiento de la tabla de peticiones se ha producido un error. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	
CurrentStep	OUTPUT	INT	0	Paso de la tabla de peticiones que se encuentra actualmente en preparación	
StepCode	OUTPUT	WORD	16#0000	Valor numérico/patrón de bits definido por el usuario que corresponde a la etapa actualmente en procesamiento	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_ChangeDynamic

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4

Descripción

La instrucción de Motion Control "MC_ChangeDynamic" permite modificar los ajustes siguientes del eje:

- Modificar valor del tiempo de aceleración (aceleración)
- Modificar valor del tiempo de deceleración (deceleración)
- Modificar valor del tiempo de deceleración de parada de emergencia (deceleración de parada de emergencia)
- Modificar valor del tiempo de redondeo (tirón)

El efecto del cambio se puede consultar en la descripción de las variables (Página 10573).

Requisitos

El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.

Comportamiento de relevo

Una petición MC_ChangeDynamic no puede ser cancelada por ninguna otra petición de Motion Control.

Una nueva petición MC_ChangeDynamic no cancela ninguna petición de Motion Control en curso.

Parámetro

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_SpeedAxis	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
ChangeRampUp	INPUT	BOOL	FALSE	TRUE	Modificar tiempo de posicionamiento según el parámetro de entrada "RampUpTime":
RampUpTime	INPUT	REAL	5.00	Tiempo (en segundos) que debe pasar para acelerar el eje sin limitación de tirones desde la parada a la velocidad máxima configurada. El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.Acceleration. El efecto de los cambios puede consultarse en la descripción de esta variable.	
ChangeRampDown	INPUT	BOOL	FALSE	TRUE	Tiempo de deceleración según el parámetro de entrada "RampDownTime" modificado
RampDownTime	INPUT	REAL	5.00	Tiempo (en segundos) que debe pasar para decelerar el eje sin limitación de tirones desde la velocidad máxima configurada hasta su detención El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.Deceleration . El efecto de los cambios puede consultarse en la descripción de esta variable.	
ChangeEmergency	INPUT	BOOL	FALSE	TRUE	Modificar tiempo de deceleración de parada de emergencia según el parámetro de entrada "EmergencyRampTime"
EmergencyRampTime	INPUT	REAL	2.00	Tiempo (en segundos) que debe transcurrir para decelerar el eje sin limitación de tirones en el modo de parada de emergencia desde la velocidad máxima configurada hasta la velocidad cero El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.EmergencyDeceleration . El efecto de los cambios puede consultarse en la descripción de esta variable.	
ChangeJerkTime	INPUT	BOOL	FALSE	TRUE	Modificar tiempo de redondeo según el parámetro de entrada "JerkTime"
JerkTime	INPUT	REAL	0.25	Tiempo de redondeo (en segundos) que se aplica a la rampa de aceleración y deceleración del eje El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.Jerk . El efecto de los cambios puede consultarse en la descripción de esta variable.	

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Done	OUTPUT	BOOL	FALSE	TRUE	Los valores modificados se han escrito en el DB tecnológico. En la descripción de las variables se indica cuándo surte efecto la modificación.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Nota

En los parámetros de entrada "RampUpTime", "RampDownTime", "EmergencyRampTime" und "JerkTime" se pueden indicar valores que excedan los valores límite admisibles de los parámetros resultantes: "aceleración", "deceleración", "deceleración de parada de emergencia" y "tirón".

Teniendo en cuenta las ecuaciones y límites del capítulo "Dinámica (Página 10468)", recuerde que las entradas que efectúe deben encontrarse dentro del rango permitido.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

Modificar la configuración de los valores de dinámica en el programa de usuario (Página 10474)

Modificar la configuración para la referenciación en el programa de usuario (Página 10481)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

S7-1200 Motion Control V1...3 (Página 5055)

MC_ReadParam

MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4

Descripción

La instrucción de Motion Control "MC_ReadParam" permite la lectura continuada de los datos de movimiento de un eje. El valor actual de las correspondientes variables se determina al iniciarse la petición.

Pueden leerse los siguientes datos de movimiento:

- Posición actual del eje
- Velocidad actual del eje
- Distancia actual del eje hasta la posición de destino
- Posición de destino del eje

Requisitos

El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.

Comportamiento de relevo

Una petición MC_ReadParam no puede ser cancelada por ninguna otra petición de Motion Control.

Una nueva petición MC_ReadParam no cancela ninguna petición de Motion Control en curso.

Parámetro

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Enable	INPUT	BOOL	FALSE	TRUE	Leer la variable determinada a través de "Parameter" y almacenar el valor en la dirección de destino determinada a través de "Value".
				FALSE	No actualizar datos de movimiento parametrizados
Parameter	INPUT	VARIANT (REAL)	-	Puntero VARIANT en el valor que se debe leer. Las siguientes variables son admisibles: • <Nombre de eje>.Position • <Nombre de eje>.Velocity • <Nombre de eje>.StatusPositioning.Distance • <Nombre de eje>.StatusPositioning.TargetPosition	

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Value	INOUT	VARIANT (REAL)	-	Puntero VARIANT en la variable de destino o en la dirección de destino en la que se debe escribir el valor leído.	
Valid	OUTPUT	BOOL	FALSE	TRUE	El valor leído es válido.
				FALSE	El valor leído no es válido.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

S7-1200 Motion Control V1...3 (Página 5055)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

MC_WriteParam

MC_WriteParam: Escribir variable del eje de posicionamiento V4

Descripción

La instrucción de Motion Control "MC_WriteParam" permite escribir variables del objeto tecnológico Eje de posicionamiento en el programa de usuario. A diferencia de la asignación de valores de las variables en el programa de usuario, "MC_WriteParam" también puede modificar valores de variables de solo lectura.

Las variables que se pueden escribir en las distintas condiciones, así como el momento en el que se hacen efectivos los cambios, se pueden consultar en la descripción de las variables del objeto tecnológico (Página 10573).

Requisitos

- El objeto tecnológico Eje de posicionamiento se ha configurado correctamente.
- Para escribir variables que solo son legibles en el programa de usuario, el eje debe estar bloqueado.

Comportamiento de relevo

Una petición MC_WriteParam no puede ser cancelada por ninguna otra petición de Motion Control.

Una nueva petición MC_WriteParam no cancela ninguna petición de Motion Control en curso.

Parámetro

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Parameter	INPUT	VARIANT (BOOL, INT, DINT, REAL)	-	Puntero VARIANT en la variable del objeto tecnológico (Página 10573) Eje de posicionamiento (dirección de destino) que se desea escribir.	
Value	INPUT	VARIANT (BOOL, INT, DINT, REAL)	-	Puntero VARIANT en el valor (dirección de origen) que se desea escribir.	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Done	OUTPUT	BOOL	FALSE	TRUE	Se ha escrito el valor
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10563) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10563) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

S7-1200 Motion Control V1...3 (Página 5055)

MC_Power: Habilitar, bloquear ejes V4 (Página 5019)

MC_Reset: Acusar errores V4 (Página 5023)

MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)

MC_Halt: Parar ejes V4 (Página 5028)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)

MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)

MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)

S7-1200 Motion Control V1...3

MC_Power

MC_Power: Habilitar/bloquear ejes V1...3

Descripción

La instrucción de Motion Control "MC_Power" habilita o bloquea un eje.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- No hay ningún error que impida la habilitación.

Comportamiento de relevo

El procesamiento del "MC_Power" no puede ser cancelado por ninguna petición de Motion Control.

Al bloquear el eje (parámetro de entrada "Enable" = FALSE) se cancelan todas las peticiones de Motion Control conforme al "StopMode" seleccionado en el respectivo objeto tecnológico.

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Enable	INPUT	BOOL	FALSE	TRUE	Motion Control intenta habilitar el eje.
				FALSE	Todas las peticiones en curso se cancelan conforme al "StopMode" parametrizado. El eje se detiene y se bloquea.
StopMode	INPUT	INT	0	0	Parada de emergencia Si existe una demanda de bloqueo del eje, éste frena con la deceleración de parada de emergencia configurada. Cuando el eje se para, se bloquea.
				1	Desconexión inmediata Si existe una demanda de bloqueo del eje, ésta se bloquea sin deceleración. La salida de impulsos se detiene inmediatamente.
				2	Parada de emergencia con limitación de tirones Si existe una demanda de bloqueo del eje, éste frena con la deceleración de parada de emergencia configurada. Si la limitación de tirones está activada, se tiene en cuenta el tirón configurado. Cuando el eje se para, se bloquea.
Status	OUTPUT	BOOL	FALSE	Estado de la habilitación del eje	
				FALSE	El eje está bloqueado El eje no ejecuta ninguna petición de Motion Control y no acepta nuevas peticiones (excepción: Petición MC_Reset). El eje no está referenciado. Al bloquear, el estado cambia a FALSE en cuanto el eje se para.
				TRUE	El eje está habilitado El eje está listo para ejecutar peticiones de Motion Control. Al habilitar el eje, el estado cambia a TRUE en cuanto se produce la señal "Accionamiento listo". Si no se ha configurado la señal del accionamiento "Accionamiento listo" en la configuración del eje, el estado cambia inmediatamente a TRUE.
Busy	OUTPUT	BOOL	FALSE	TRUE	"MC_Power" está activo.
Error	OUTPUT	BOOL	FALSE	TRUE	En la instrucción de Motion Control "MC_Power" o el objeto tecnológico correspondiente se ha producido un error. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Nota

Si el eje se desconecta a causa de un error, se vuelve a habilitar automáticamente tras eliminar y acusar el error. Requisito para ello es que el parámetro de entrada "Enable" mantenga el valor TRUE durante esta operación.

Habilitación de un eje con señales del accionamiento configuradas

Para habilitar un eje, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Configure el parámetro de entrada "StopMode" con el valor deseado. Ponga el parámetro de entrada "Enable" en TRUE.
   La salida de habilitación para "Habilitar accionamiento" cambia a TRUE para habilitar la potencia del accionamiento. La CPU aguarda la señal "Accionamiento listo" del accionamiento.
   El eje queda habilitado en cuanto se emite la señal "Accionamiento listo" en la entrada de disponibilidad configurada de la CPU. El parámetro de salida "Status", así como la variable del objeto tecnológico <Nombre del eje>.StatusBits.Enable muestran el valor TRUE.

Habilitación de un eje sin señales del accionamiento configuradas

Para habilitar un eje, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Configure el parámetro de entrada "StopMode" con el valor deseado. Ponga el parámetro de entrada "Enable" en TRUE. El eje es habilitado. El parámetro de salida "Status", así como la variable del objeto tecnológico <Nombre del eje>.StatusBits.Enable muestran el valor TRUE.

Bloqueo de un eje

Para bloquear un eje puede proceder de la siguiente manera:

1. Detenga el eje.
   La variable del objeto tecnológico <Nombre del eje>.StatusBits.StandStill permite reconocer el momento de parada del eje.
2. Cambie el parámetro de entrada "Enable" a FALSE en cuanto se produzca la parada.
3. Si los parámetros de salida "Busy" y "Status", así como la variable del objeto tecnológico <Nombre del eje>.StatusBits.Enable muestran el valor FALSE, el bloqueo del eje de habrá finalizado.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_Power: Diagrama de funcionamiento V1...3 (Página 5059)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

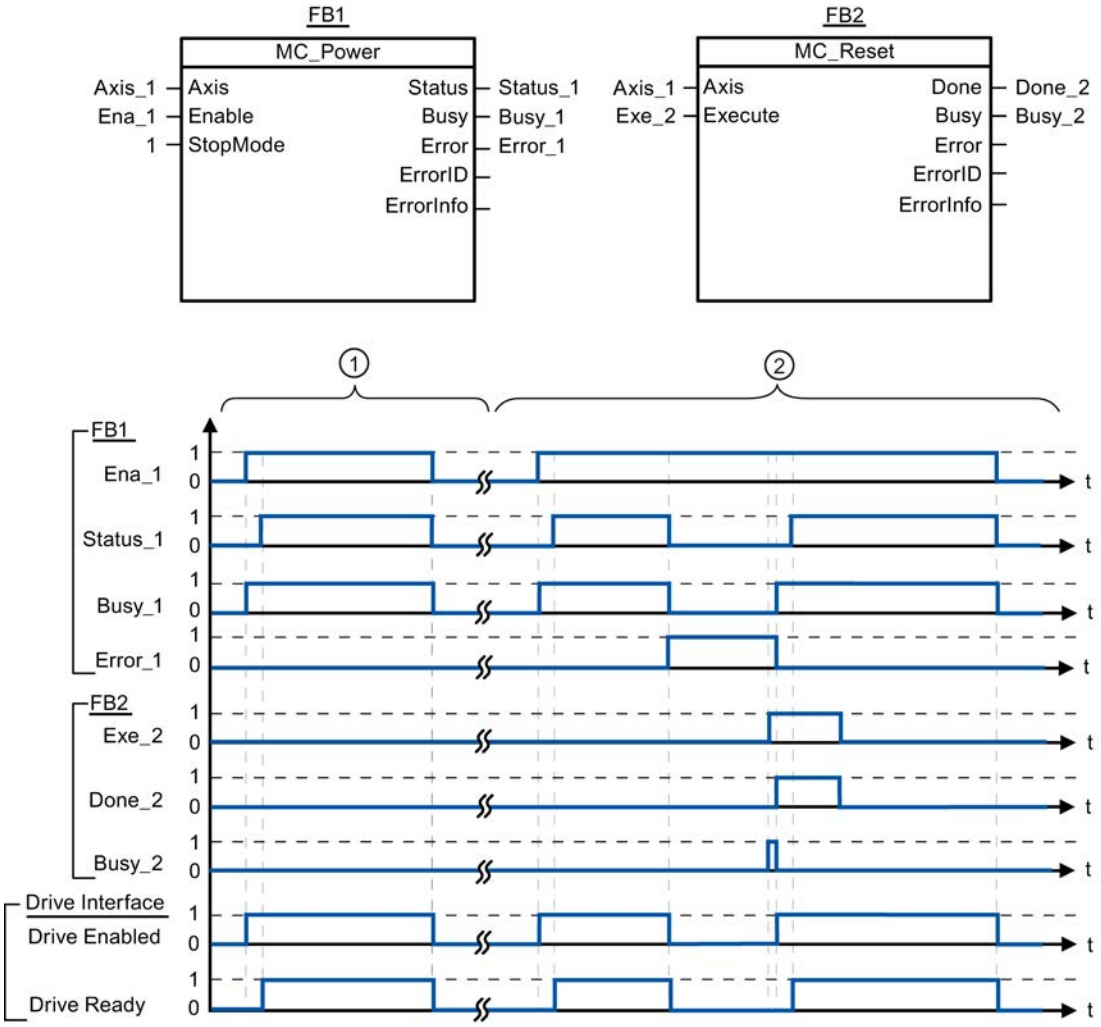
MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_Power: Diagrama de funcionamiento V1...3

Diagrama de funcionamiento



①	Se habilita un eje y a continuación se vuelve a bloquear. En cuanto el accionamiento devuelve la señal "Accionamiento listo" a la CPU se puede verificar el éxito de la habilitación a través de "Status_1".
②	Una vez habilitado un eje se produce un error, provocando el bloqueo del eje. El error se soluciona y se acusa con "MC_Reset". A continuación se vuelve a habilitar el eje.

Consulte también

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset

MC_Reset: Acusar errores V1...3

Descripción

La instrucción de Motion Control "MC_Reset" permite acusar "Errores de funcionamiento con parada del eje" y "Errores de configuración". En la "Lista de ErrorIDs y ErrorInfos", en el apartado "Ayuda", puede consultar los errores que deben ser acusados.

A partir de la versión V3.0 es posible cargar la configuración del eje en la memoria de trabajo después de una carga en el estado operativo RUN.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- En caso de un error de configuración acusable debe haberse eliminado la causa (p. ej. la aceleración en el objeto tecnológico Eje de posicionamiento ha sido cambiada a un valor válido).

Comportamiento de relevo

La petición MC_Reset no puede ser cancelada por ninguna otra petición de Motion Control.

La nueva petición MC_Reset no cancela ninguna otra petición de Motion Control en curso.

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Restart	INPUT	BOOL	FALSE	(A partir de la versión V3.0)	
				TRUE	Carga la configuración del eje en la memoria de trabajo desde la memoria de carga. La petición solo puede ejecutarse con el eje bloqueado. Para ello, tenga en cuenta las indicaciones referentes a la Carga en la CPU (Página 10502).
				FALSE	Acusa errores pendientes
Done	OUTPUT	BOOL	FALSE	TRUE	El error se ha acusado.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Acusar un error con MC_Reset

Para acusar un error proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Inicie el acuse del error mediante un flanco ascendente en el parámetro de entrada "Execute".
3. El error ha sido acusado si el parámetro de entrada "Done" muestra el valor TRUE y la variable del objeto tecnológico <Nombre del eje:>.StatusBits.Error el valor FALSE.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

Cargar en la CPU (Página 10502)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_Home

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3

Descripción

La instrucción de Motion Control "MC_Home" permite contrastar la coordenada axial con la posición física real del accionamiento. Si el eje debe posicionarse de forma absoluta, es necesaria la referenciación. Pueden ejecutarse los siguientes tipos de referenciación:

- Referenciación activa (Mode = 3)
  La aproximación al punto de referencia se ejecuta automáticamente.
- Referenciación pasiva (Mode = 2)
  En la referenciación pasiva, la instrucción de Motion Control "MC_Home" no realiza ningún movimiento de referencia. El desplazamiento necesario para ello debe ser realizado por el usuario con otras instrucciones de Motion Control. El eje se referencia al detectarse el sensor del punto de referencia.
- Referenciación directa absoluta (Mode = 0)
  La posición actual del eje se fija con el valor del parámetro "Position".
- Referenciación directa relativa (Mode = 1)
  La posición actual del eje se desplaza en la cuantía del valor del parámetro "Position".

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- El eje está habilitado.
- Al iniciar con Mode = 0, 1 y 2 no puede haber ninguna petición MC_CommandTable activa.

Comportamiento de relevo

El comportamiento de relevo depende del modo seleccionado:

Mode = 0, 1

La petición MC_Home no puede ser cancelada por ninguna otra petición de Motion Control.

La petición MC_Home no cancela ninguna petición de Motion Control en curso. Las peticiones de desplazamiento con referencia de posición continúan, después de la referenciación, de acuerdo con la nueva posición de referenciación (valor en el parámetro de entrada: "Position").

Mode = 2

La petición MC_Home puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 2, 3

La nueva petición MC_Home cancela la siguiente petición de Motion Control en curso:

- Petición MC_Home Mode = 2

Las peticiones de desplazamiento con referencia de posición continúan, después de la referenciación, de acuerdo con la nueva posición de referenciación (valor en el parámetro de entrada: "Position").

Mode = 3

La petición MC_Home puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_Home cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 2, 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetro

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Position	INPUT	REAL	0.0	• Mode = 0, 2 y 3 Posición absoluta del eje una vez finalizado el proceso de referenciación • Mode = 1 Valor de corrección para la posición actual del eje Límites: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$	
Mode	INPUT	INT	0	Modo de toma de referencia	
				0	Referenciación directa absoluta La nueva posición del eje es el valor de posición del parámetro "Position".

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
				1	Referenciación directa relativa La nueva posición del eje es la posición actual del eje + el valor de posición del parámetro "Position".
				2	Referenciación pasiva Referenciación conforme a la configuración del eje. Tras la referenciación se aplica el valor del parámetro "Position" como nueva posición del eje.
				3	Referenciación activa Aproximación al punto de referencia conforme a la configuración del eje. Tras la referenciación se aplica el valor del parámetro "Position" como nueva posición del eje.
Done	OUTPUT	BOOL	FALSE	TRUE	La petición ha finalizado
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Nota

La toma de referencia de un eje se pierde cuando se dan las siguientes condiciones:

- Bloqueo del eje por la instrucción de Motion Control "MC_Power"
- Cambio entre modo automático y control manual
- Al iniciar la referenciación activa. La toma de referencia del eje vuelve a estar presente una vez concluido el proceso de referenciación.
- Tras POWER OFF -> POWER ON de la CPU
- Tras un rearranque completo de la CPU (RUN-STOP -> STOP-RUN)

Referenciación de un eje

Para referenciar el eje, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Configure los valores con los parámetros de entrada necesarios e inicie la referenciación mediante un flanco ascendente en el parámetro de entrada "Execute"
3. La referenciación concluye cuando el parámetro de salida muestra "Done" y la variable del objeto tecnológico <Nombre del eje:>.StatusBits.HomingDone el valor TRUE.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_Halt

MC_Halt: Parar ejes V1...3

Descripción

La instrucción de Motion Control "MC_Halt" cancela todas las operaciones de movimiento y frena el eje con la deceleración configurada hasta pararlo por completo. La posición de parada no está definida.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_Halt puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_Halt cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Done	OUTPUT	BOOL	FALSE	TRUE	Se ha alcanzado la velocidad cero
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_Halt: Diagrama de funcionamiento V1...3 (Página 5068)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

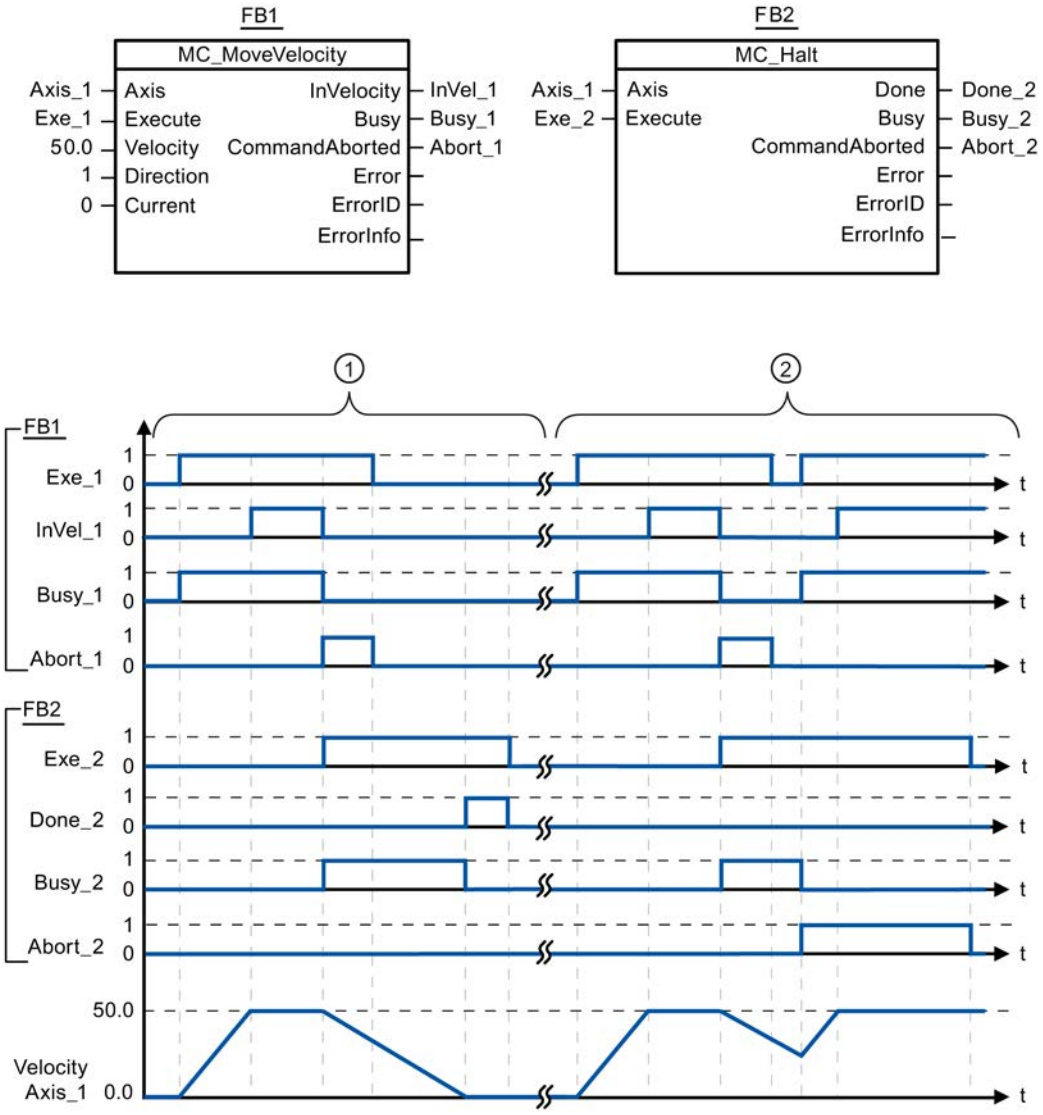
MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_Halt: Diagrama de funcionamiento V1...3

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 5.0

①	El eje se frena con una petición MC_Halt hasta pararse completamente. La parada completa se notifica mediante "Done_2".
②	Mientras una petición MC_Halt frena el eje, dicha petición es cancelada por otra petición de desplazamiento. La cancelación se notifica mediante "Abort_2".

Consulte también

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3

Descripción

La instrucción de Motion Control "MC_MoveAbsolute" inicia un movimiento de posicionamiento del eje respecto a una posición absoluta.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- El eje está habilitado.
- El eje está referenciado.

Comportamiento de relevo

La petición MC_MoveAbsolute puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveAbsolute cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Position	INPUT	REAL	0.0	Posición absoluta de destino Límites: $-1.0e^{12} \leq$ Position $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Velocidad del eje No siempre se alcanza esta velocidad, dependiendo de la aceleración y deceleración configuradas y de la posición de destino fijada. Límites: Velocidad de arranque/parada ≤ Velocity ≤ velocidad máxima	
Done	OUTPUT	BOOL	FALSE	TRUE	Posición absoluta de destino alcanzada
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_MoveAbsolute: Diagrama de funcionamiento V1...3 (Página 5071)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

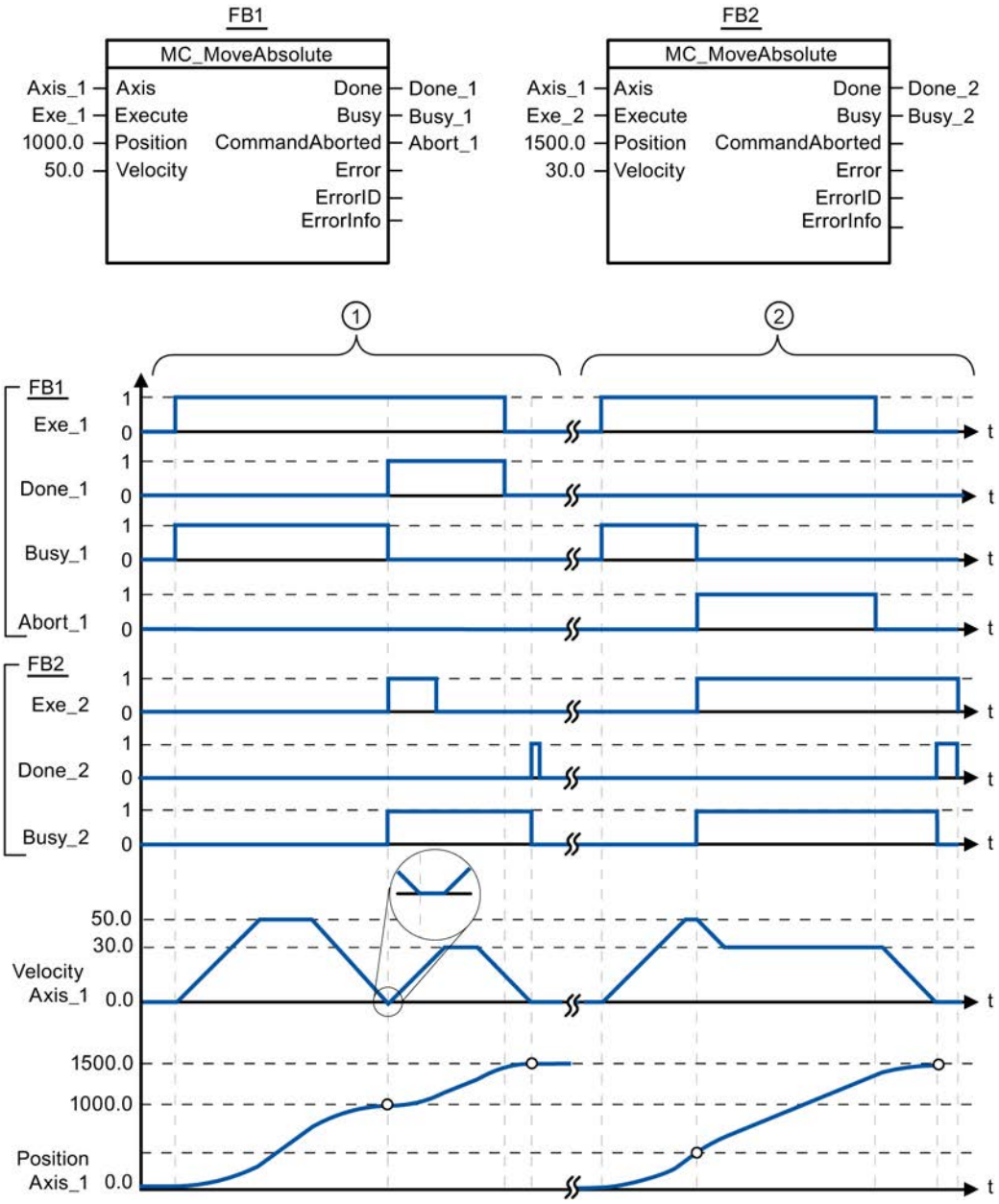
MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_MoveAbsolute: Diagrama de funcionamiento V1...3

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 10.0

①	Un eje es desplazado por una petición MC_MoveAbsolute a la posición absoluta 1000.0. En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_1". Con "Done_1" = TRUE se lanza otra petición MC_MoveAbsolute, con posición de destino 1500.0. Debido a los tiempos de reacción (p.ej. tiempo de ciclo del programa de usuario, ...) se produce una breve parada del eje (ver representación ampliada). En cuanto se alcanza con éxito la nueva posición de destino el sistema lo notifica mediante "Done_2".
②	Una petición MC_MoveAbsolute activa es cancelada por otra petición MC_MoveAbsolute. La cancelación se notifica mediante "Abort_1". A continuación, el eje se mueve con la nueva velocidad hasta la nueva posición de destino 1500.0. En cuanto se alcanza la nueva posición de destino el sistema lo notifica mediante "Done_2".

Consulte también

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative

MC_MoveRelative: Posicionar ejes de forma relativa V1...3

Descripción

La instrucción de Motion Control "MC_MoveRelative" inicia un movimiento de posicionamiento relativo respecto a la posición inicial.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_MoveRelative puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveRelative cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Distance	INPUT	REAL	0.0	Trayecto de posicionamiento Límites: $-1.0e^{12} \leq$ Distance $\leq 1.0e^{12}$	
Velocity	INPUT	REAL	10.0	Velocidad del eje No siempre se alcanza esta velocidad, dependiendo de la aceleración y deceleración configuradas y del trayecto fijado. Límites: Velocidad de arranque/parada ≤ Velocity ≤ velocidad máxima	
Done	OUTPUT	BOOL	FALSE	TRUE	Posición de destino alcanzada
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_MoveRelative: Diagrama de funcionamiento V1...3 (Página 5075)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

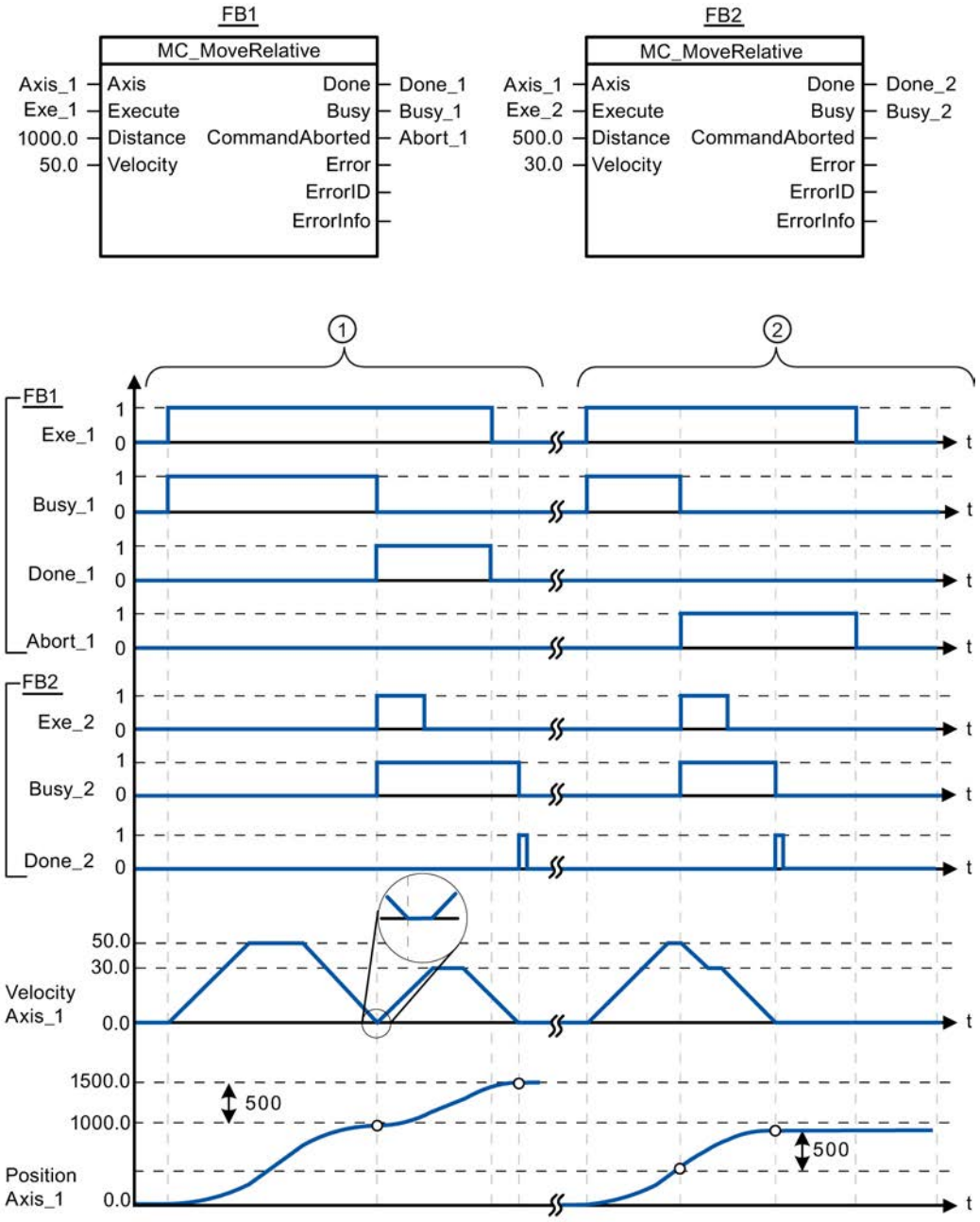
MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_MoveRelative: Diagrama de funcionamiento V1...3

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 10.0

①	Un eje es desplazado por una petición MC_MoveRelative en la distancia ("Distance") 1000.0. En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_1". Con "Done_1" = TRUE se lanza otra petición MC_MoveRelative con la distancia 500.0. Debido a los tiempos de reacción (p.ej. tiempo de ciclo del programa de usuario, ...) se produce una breve parada del eje (ver representación ampliada). En cuanto se alcanza con éxito la nueva posición de destino el sistema lo notifica mediante "Done_2".
②	Una petición MC_MoveRelative activa es cancelada por otra petición MC_MoveRelative. La cancelación se notifica mediante "Abort_1". A continuación, el eje se mueve con la nueva velocidad hasta recorrer el nuevo trayecto ("Distance") 500.0. En cuanto se alcanza la nueva posición de destino el sistema lo notifica mediante "Done_2".

Consulte también

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3

Descripción

La instrucción de Motion Control "MC_MoveVelocity" mueve el eje de forma constante a la velocidad predeterminada.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_MoveVelocity puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveVelocity cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt

- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Velocity	INPUT	REAL	10.0	Velocidad predeterminada para mover el eje Límites: Velocidad de inicio/parada ≤ |Velocity| ≤ velocidad máxima (Velocity = 0.0 está permitido)	
Direction	INPUT	INT	0	Preajuste de sentido	
				0	Sentido de giro según el signo del valor del parámetro "Velocity"
				1	Sentido de giro positivo (El signo del valor del parámetro "Velocity" se ignora)
				2	Sentido de giro negativo (El signo del valor del parámetro "Velocity" se ignora)
Current	INPUT	BOOL	FALSE	Mantener velocidad actual	
				FALSE	"Mantener velocidad actual" desactivado. Se utilizan los valores de los parámetros "Velocity" y "Direction".
				TRUE	"Mantener velocidad actual" activado. Los valores de los parámetros "Velocity" y "Direction" no se tienen en cuenta. En cuanto el eje vuelve a desplazarse a la velocidad actual, el parámetro "InVelocity" devuelve el valor TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	• "Current" = FALSE: La velocidad indicada en el parámetro "Velocity" ha sido alcanzada. • "Current" = TRUE: El eje se mueve a la velocidad actual en el momento de arranque.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Comportamiento con velocidad de consigna cero (Velocity = 0.0)

Una petición MC_MoveVelocity con "Velocity" = 0.0 cancela (al igual que una petición MC_Halt) las peticiones de movimiento activas y detiene el eje con la deceleración configurada.

Una vez alcanzada la parada se muestra en el parámetro de salida "InVelocity" el valor TRUE durante al menos un ciclo de programa.

"Durante el proceso de deceleración, Busy" indica el valor TRUE y lo cambia junto con "InVelocity" a FALSE. Si el parámetro "Execute" = TRUE, "InVelocity" y "Busy" se indican memorizados.

Cuando inicia la petición MC_MoveVelocity se activa el bit de estado "SpeedCommand" en el objeto tecnológico. El bit de estado "ConstantVelocity" con la parada del eje. Ambos bits se ajustan a la nueva situación con el inicio de una nueva petición de movimiento.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_MoveVelocity: Diagrama de funcionamiento V1...3 (Página 5079)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

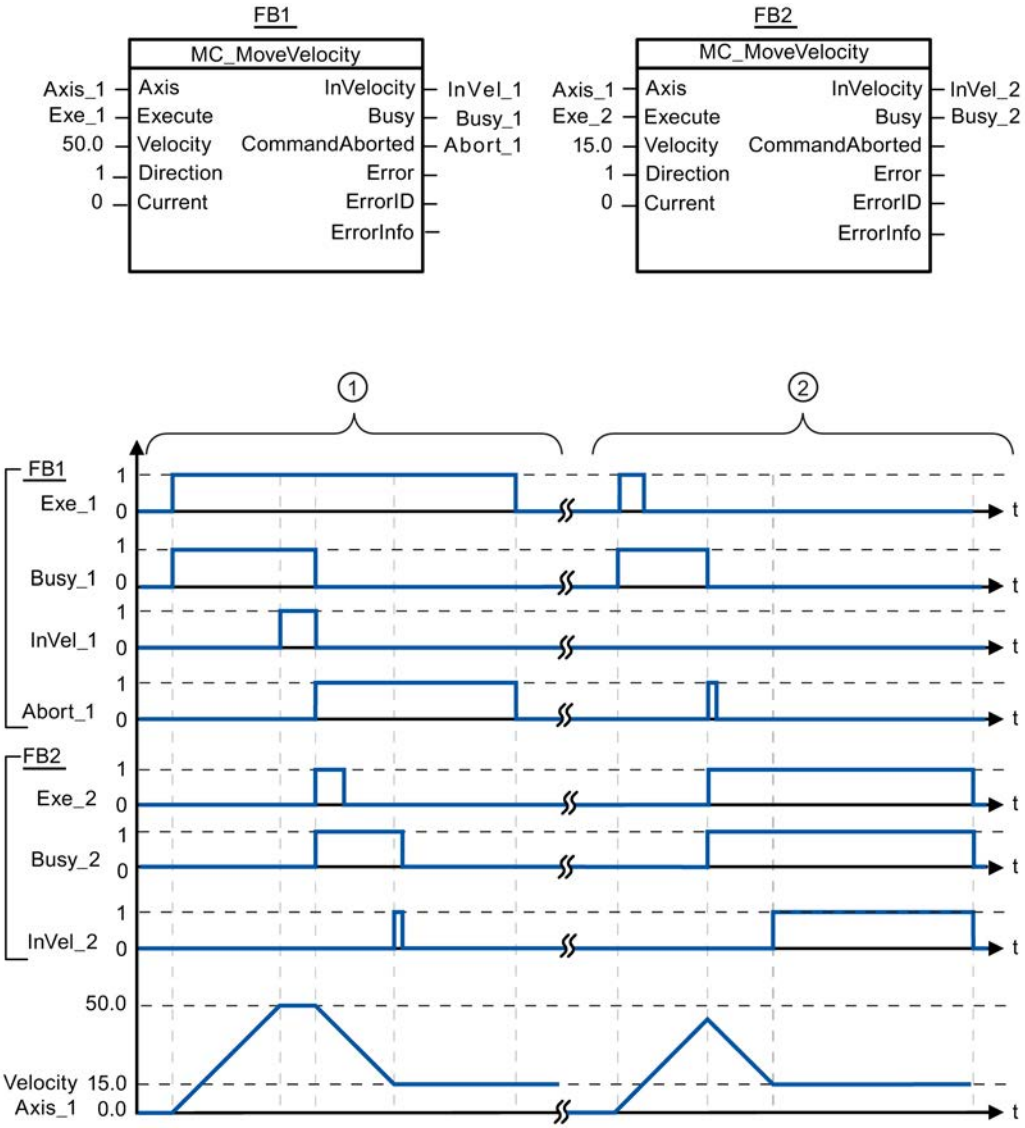
MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_MoveVelocity: Diagrama de funcionamiento V1...3

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 10.0

①	Una petición MC_MoveVelocity activa notifica el alcance de su velocidad consignada a través de "InVel_1". Seguidamente es cancelada por otra petición MC_MoveVelocity. La cancelación se notifica mediante "Abort_1". En cuanto se alcanza la nueva velocidad de destino 15.0 el sistema lo notifica mediante "InVel_2". A continuación el eje continúa moviéndose de forma constante a la nueva velocidad.
②	Una petición MC_MoveVelocity activa es cancelada por otra petición MC_MoveVelocity antes de alcanzar su velocidad consignada. La cancelación se notifica mediante "Abort_1". En cuanto se alcanza la nueva velocidad de destino 15.0 el sistema lo notifica mediante "InVel_2". A continuación el eje continúa moviéndose de forma constante a la nueva velocidad.

Consulte también

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

MC_MoveJog

MC_MoveJog: Mover ejes en el modo de operación Jog V1...3

Descripción

La instrucción de Motion Control "MC_MoveJog" mueve el eje de forma constante a la velocidad predeterminada en modo Jog. Utilice esta instrucción de Motion Control para, p.ej., fines de test y puesta en servicio.

Requisitos

- El objeto tecnológico Eje se ha configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_MoveJog puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_MoveJog cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute

- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
JogForward	INPUT	BOOL	FALSE	Mientras el parámetro sea TRUE, el eje se moverá a la velocidad predeterminada en el parámetro "Velocity" en sentido positivo.	
JogBackward	INPUT	BOOL	FALSE	Mientras el parámetro sea TRUE, el eje se moverá a la velocidad predeterminada en el parámetro "Velocity" en sentido negativo.	
Si ambos parámetros son al mismo tiempo TRUE, el eje se detiene con la deceleración configurada. En los parámetros "Error", "ErrorID" y "ErrorInfo" se muestra un error.					
Velocity	INPUT	REAL	10.0	Preajuste de velocidad para el modo Jog.	
				Límites de la versión de instrucción V1.0: Velocidad de inicio/parada ≤ |Velocity| ≤ velocidad máxima	
				Valores límite de la versión de instrucción V2.0: Velocidad de arranque/parada ≤ Velocity ≤ velocidad máxima	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	La velocidad indicada en el parámetro "Velocity" ha sido alcanzada.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_MoveJog: Diagrama de funcionamiento V1...3 (Página 5082)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

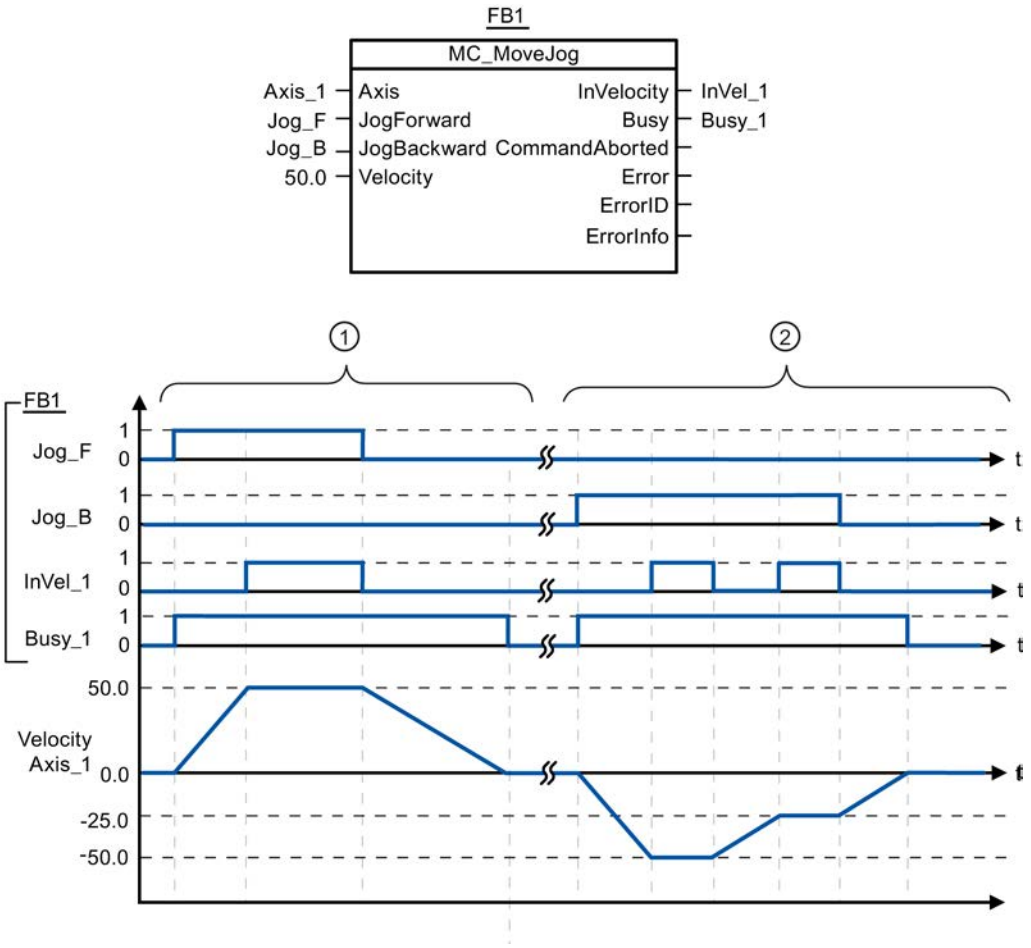
MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_MoveJog: Diagrama de funcionamiento V1...3

Diagrama de funcionamiento



Los siguientes valores han sido configurados en la ventana de configuración **Dinámica > General**:

- Aceleración: 10.0
- Deceleración: 5.0

①	"Jog_F" permite mover el eje en sentido positivo en modo Jog. En cuanto se alcanza la velocidad de destino 50.0 el sistema lo notifica a través de "InVelo_1". Tras desactivar " Jog_F" el eje frena hasta pararse del todo.
②	"Jog_B" permite mover el eje en sentido negativo en modo Jog. En cuanto se alcanza la velocidad de destino 50.0 el sistema lo notifica a través de "InVelo_1". Tras desactivar " Jog_B" el eje frena hasta pararse del todo.

Consulte también

MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3

Descripción

La instrucción de Motion Control "MC_CommandTable" reúne varias peticiones individuales de control de un eje en una secuencia de movimientos.

Requisitos

- El objeto tecnológico Eje se ha insertado y configurado correctamente en la versión V2.
- El objeto tecnológico Tabla de peticiones se ha insertado y configurado correctamente.
- El eje está habilitado.

Comportamiento de relevo

La petición MC_CommandTable puede ser cancelada por las siguientes peticiones de Motion Control:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La nueva petición MC_CommandTable cancela las siguientes peticiones de Motion Control en curso:

- Petición MC_Home Mode = 3
- Petición MC_Halt
- Petición MC_MoveAbsolute
- Petición MC_MoveRelative
- Petición MC_MoveVelocity
- Petición MC_MoveJog
- Petición MC_CommandTable

La petición Motion Control en curso se cancela cuando se inicia la primera petición "Positioning Relative", "Positioning Absolute", "Velocity set point" o "Halt".

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
CommandTable	INPUT	TO_CommandTable_1	-	Objeto tecnológico de la tabla de peticiones	
Execute	INPUT	BOOL	FALSE	Inicio de la tabla de peticiones con flanco ascendente	
StartStep	INPUT	INT	1	Indicación que señala a partir de qué paso debe procesarse la tabla de peticiones. Límites: 1 ≤ StartStep ≤ EndStep	
EndStep	INPUT	INT	32	Indicación que señala hasta qué paso debe procesarse la tabla de peticiones. Límites: StartStep ≤ EndStep ≤ 32	
Done	OUTPUT	BOOL	FALSE	TRUE	La tabla de peticiones se ha procesado correctamente
Busy	OUTPUT	BOOL	FALSE	TRUE	La tabla de peticiones se encuentra en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La tabla de peticiones ha sido cancelada por otra petición durante el procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Durante el procesamiento de la tabla de peticiones se ha producido un error. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	
CurrentStep	OUTPUT	INT	0	Paso de la tabla de peticiones que se encuentra actualmente en preparación	
StepCode	OUTPUT	WORD	16#0000	Valor numérico/patrón de bits definido por el usuario que corresponde a la etapa actualmente en procesamiento	

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3 (Página 5085)

MC_ChangeDynamic

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V2...3

Descripción

La instrucción de Motion Control "MC_ChangeDynamic" permite modificar los ajustes siguientes del eje:

- Modificar valor del tiempo de aceleración (aceleración)
- Modificar valor del tiempo de deceleración (deceleración)
- Modificar valor del tiempo de deceleración de parada de emergencia (deceleración de parada de emergencia)
- Modificar valor del tiempo de redondeo (tirón)

El efecto del cambio se puede consultar en la descripción de las variables (Página 10623).

Requisitos

- El objeto tecnológico Eje se ha insertado en la versión V2.
- El objeto tecnológico Eje se ha configurado correctamente.

Comportamiento de relevo

Una petición MC_ChangeDynamic no puede ser cancelada por ninguna otra petición de Motion Control.

Una nueva petición MC_ChangeDynamic no cancela ninguna petición de Motion Control en curso.

Parámetros

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	INPUT	TO_Axis_1	-	Objeto tecnológico del eje	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
ChangeRampUp	INPUT	BOOL	FALSE	TRUE	Modificar tiempo de posicionamiento según el parámetro de entrada "RampUpTime":
RampUpTime	INPUT	REAL	5.00	Tiempo (en segundos) que debe pasar para acelerar el eje sin limitación de tirones desde la parada a la velocidad máxima configurada. El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.Acceleration. El efecto de los cambios puede consultarse en la descripción de esta variable.	
ChangeRampDown	INPUT	BOOL	FALSE	TRUE	Tiempo de deceleración según el parámetro de entrada "RampDownTime modificado
RampDownTime	INPUT	REAL	5.00	Tiempo (en segundos) que debe pasar para decelerar el eje sin limitación de tirones desde la velocidad máxima configurada hasta su detención El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.Deceleration . El efecto de los cambios puede consultarse en la descripción de esta variable.	
ChangeEmergency	INPUT	BOOL	FALSE	TRUE	Modificar tiempo de deceleración de parada de emergencia según el parámetro de entrada "EmergencyRampTime"
EmergencyRampTime	INPUT	REAL	2.00	Tiempo (en segundos) que debe transcurrir para decelerar el eje sin limitación de tirones en el modo de parada de emergencia desde la velocidad máxima configurada hasta la velocidad cero El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.EmergencyDeceleration . El efecto de los cambios puede consultarse en la descripción de esta variable.	
ChangeJerkTime	INPUT	BOOL	FALSE	TRUE	Modificar tiempo de redondeo según el parámetro de entrada "JerkTime"
JerkTime	INPUT	REAL	0.25	Tiempo de redondeo (en segundos) que se aplica a la rampa de aceleración y deceleración del eje El cambio afecta al valor de la variable <Nombre del eje>.Config.DynamicDefaults.Jerk . El efecto de los cambios puede consultarse en la descripción de esta variable.	
Done	OUTPUT	BOOL	FALSE	TRUE	Los valores modificados se han escrito en el DB tecnológico. En la descripción de las variables se indica cuándo surte efecto la modificación.

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en los parámetros "ErrorID" y "ErrorInfo".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10609) del parámetro "Error"	
ErrorInfo	OUTPUT	WORD	16#0000	Identificador de información de error (Página 10609) del parámetro "ErrorID"	

Nota

En los parámetros de entrada "RampUpTime", "RampDownTime", "EmergencyRampTime" und "JerkTime" se pueden indicar valores que excedan los valores límite admisibles de los parámetros resultantes: "aceleración", "deceleración", "deceleración de parada de emergencia" y "tirón".

Considerando las ecuaciones y valores límite indicados en los capítulos: "Objeto tecnológico Eje" -> "Configuración del objeto tecnológico" -> "Dinámica" tenga en cuenta que las entradas que efectúe deben encontrarse dentro del rango permitido.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10526)

S7-1200 Motion Control V4 (Página 5019)

ErrorID y ErrorInfos (Página 10609)

Variables del objeto tecnológico Eje V1...3 (Página 10623)

MC_Power: Habilitar/bloquear ejes V1...3 (Página 5055)

MC_Reset: Acusar errores V1...3 (Página 5060)

MC_Home: Referenciar ejes, ajustar el punto de referencia V1...3 (Página 5062)

MC_Halt: Parar ejes V1...3 (Página 5065)

MC_MoveAbsolute: Posicionar ejes de forma absoluta V1...3 (Página 5069)

MC_MoveRelative: Posicionar ejes de forma relativa V1...3 (Página 5072)

MC_MoveVelocity: Mover ejes con preajuste de velocidad V1...3 (Página 5076)

MC_MoveJog: Mover ejes en el modo de operación Jog V1...3 (Página 5080)

MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V2...3 (Página 5083)

S7-1500 Motion Control

MC_Power

MC_Power: Habilitar y bloquear objetos tecnológicos

Descripción

La instrucción "MC_Power" de Motion Control habilita o bloquea un objeto tecnológico.

Se aplica a

- eje de posicionamiento
- eje de velocidad de giro
- encóder externo

Requisitos

- El objeto tecnológico se ha configurado correctamente.

Comportamiento de relevo

- Una petición "MC_Power" no puede ser cancelada por ninguna otra petición Motion Control.
- Una petición "MC_Power" con el parámetro "Enable" = TRUE habilita un objeto tecnológico y no cancela ninguna otra instrucción de Motion Control.
- Al bloquear el objeto tecnológico (parámetro "Enable" = FALSE) se cancelan todas las peticiones de movimiento en el objeto tecnológico correspondiente conforme al "StopMode" seleccionado. Este proceso no puede ser cancelado por el usuario.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MC_Power" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	InOut	TO_Axis	-	Objeto tecnológico	
Enable	INPUT	BOOL	FALSE	TRUE	El objeto tecnológico se habilita.
				FALSE	El objeto tecnológico se bloquea. Todas las peticiones en curso en el objeto tecnológico se cancelan conforme al "StopMode" parametrizado.

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
StopMode	INPUT	INT	0	Irrelevante para el objeto tecnológico Encóder externo. Si se bloquea un objeto tecnológico con un flanco descendente en el parámetro "Enable", el eje reduce la velocidad de acuerdo con el "StopMode" seleccionado.	
				0	Parada de emergencia Al bloquear el objeto tecnológico, el eje reduce la velocidad sin limitación de tirones con la deceleración de parada de emergencia configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Rampa de parada de emergencia", hasta que se para. Seguidamente, se retira la habilitación. (<TO>.DynamicDefaults. EmergencyDeceleration)
				1	Desconexión inmediata Al bloquear un objeto tecnológico se emite la consigna cero y se retira la habilitación. En función de la configuración, el eje se frena en el accionamiento y se para.
				2	Parada con valores dinámicos máximos Al bloquear el objeto tecnológico, el eje reduce la velocidad con la deceleración máxima configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Límites dinámicos", hasta que se para. En este caso se tiene en cuenta el tirón máximo configurado. Seguidamente, se retira la habilitación. (<TO>.DynamicLimits.MaxDeceleration; <TO>.DynamicLimits.MaxJerk)
Status	OUTPUT	BOOL	FALSE	Estado de habilitación del objeto tecnológico	
				FALSE	Bloqueado - Un eje de posicionamiento o velocidad de giro no acepta peticiones de movimiento. - El control de velocidad de giro y la regulación de posición no están activos. - No se comprueba la validez de los valores reales del objeto tecnológico.
				TRUE	Habilitado - Un eje de posicionamiento o velocidad de giro habilitado acepta peticiones de movimiento. - El control de velocidad de giro y la regulación de posición están activos. - Los valores reales del objeto tecnológico son válidos.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso.

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Error	OUTPUT	BOOL	FALSE	TRUE	En la instrucción "MC_Power" de Motion Control se ha producido un error. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Habilitación de objetos tecnológicos

Para habilitar un objeto tecnológico, ponga el parámetro "Enable" en TRUE.

Si el parámetro "Status" presenta el valor TRUE , el objeto tecnológico está habilitado.

Si, al habilitar el objeto tecnológico, hay un eje en movimiento, es decir, si existe velocidad real, el eje reduce la velocidad hasta la consigna cero con la deceleración máxima configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Límites dinámicos" (<TO>.DynamicLimits.MaxDeceleration). Esta rampa de deceleración puede relevarse con peticiones de movimiento.

Nota

Habilitación automática tras acuse de una alarma tecnológica

Si el objeto tecnológico se bloquea a causa de una alarma tecnológica, se vuelve a habilitar automáticamente tras eliminar la causa y acusar la alarma. Para ello es imprescindible que el parámetro "Enable" mantenga el valor TRUE durante esta operación.

Bloqueo de objetos tecnológicos

Para bloquear un objeto tecnológico, ponga el parámetro "Enable" en FALSE.

Si hay un eje en movimiento, reduce la velocidad conforme al "StopMode" seleccionado hasta que se para.

Si los parámetros "Busy" y "Status" presentan el valor FALSE , finaliza el bloqueo del objeto tecnológico.

Conexión del accionamiento con PROFIdrive

Al acoplar un accionamiento con PROFIdrive, la consigna, la habilitación y el estado del accionamiento se transfieren mediante el telegrama PROFIdrive.

- **Habilitar objeto tecnológico y activar accionamiento**
  Con el parámetro "Enable" = TRUE se habilita el objeto tecnológico. El accionamiento se activa de acuerdo con la norma PROFIdrive.
  Si la variable <TO>.StatusDrive.InOperation presenta el valor TRUE , el accionamiento está preparado para ejecutar consignas. El parámetro "Status" se pone al valor TRUE .
- **Bloqueo del objeto tecnológico y desactivación del accionamiento**
  Con el parámetro "Enable" = FALSE, el parámetro "Status" se pone al valor FALSE y el eje se frena de acuerdo con el "StopMode" seleccionado. El accionamiento se desactiva de acuerdo con la norma PROFIdrive.

Conexión analógica del accionamiento

La consigna se emite en una salida analógica. También es posible configurar una señal de habilitación por la salida digital (<TO>.Actor.Interface.EnableDriveOutput) y una señal de disponibilidad por la entrada digital (<TO>.Actor.Interface.DriveReadyInput).

- **Habilitar objeto tecnológico y activar accionamiento**
  Con el parámetro "Enable" = TRUE se activa la salida de habilitación ("Enable drive output"). Si el accionamiento devuelve la señal de disponibilidad por la entrada de disponibilidad ("Drive ready input"), el parámetro "Status" y la variable del objeto tecnológico <TO>.StatusDrive.InOperation se ponen a TRUE y la consigna pasa a la salida analógica.
- **Bloqueo del objeto tecnológico y desactivación del accionamiento**
  Con el parámetro "Enable" = FALSE, el parámetro "Status" se pone al valor FALSE y el eje se frena de acuerdo con el "StopMode" seleccionado. Cuando se alcanza la consigna cero, la salida de habilitación se pone a FALSE.

Información adicional

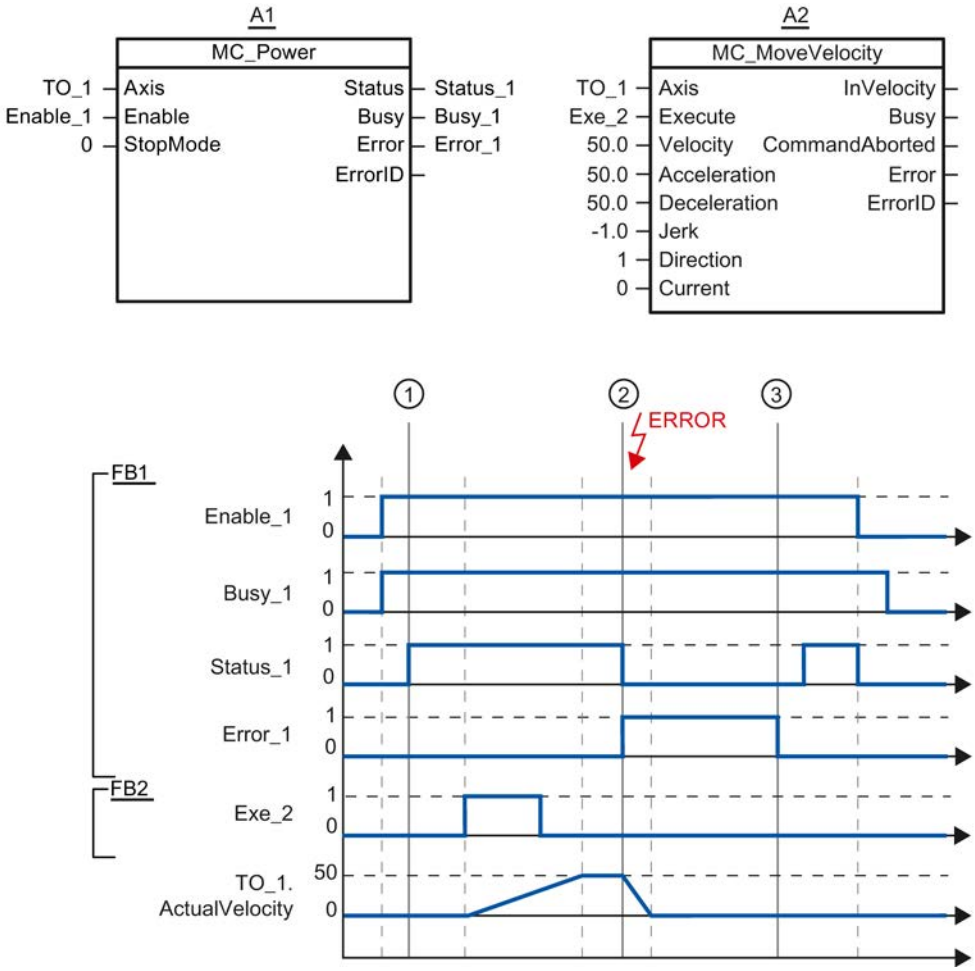
Encontrará más información sobre la habilitación y el bloqueo de objetos tecnológicos y accionamientos en el anexo Diagramas de funcionamiento MC_Power (Página 10847).

Consulte también

Identificación de error (Página 10844)

MC_Power: Diagrama de funcionamiento

Diagrama de funcionamiento: habilitación de un objeto tecnológico y ejemplo de reacción a alarma



Un objeto tecnológico se habilita cuando "Enable_1" = TRUE. La correcta habilitación se lee en "Status_1" en el instante ①. Seguidamente, el eje se mueve con una petición "MC_MoveVelocity" (A2). La curva de velocidad del eje se lee en "Velocity Axis_1".

En el instante ② se produce un error en el objeto tecnológico, que provoca el bloqueo del mismo (reacción a alarma: retirar habilitación). En función de la configuración, el eje se frena en el accionamiento y se para. Con el bloqueo del objeto tecnológico se desactiva "Status_1". Puesto que el eje no se ha bloqueado poniendo "Enable_1" = FALSE, el "StopMode" seleccionado es irrelevante. La causa del error se elimina y la alarma se acusa en el instante ③.

El objeto tecnológico se habilita de nuevo porque "Enable_1" sigue estando activado. La correcta habilitación se lee en "Status_1". Finalmente, el objeto tecnológico se bloquea poniendo "Enable_1" = FALSE.

MC_Home

MC_Home: referenciar objetos tecnológicos, ajustar el punto de referencia

Descripción

La instrucción "MC_Home" de Motion Control establece la referencia entre la posición del objeto tecnológico y la posición mecánica. El valor de posición del objeto tecnológico se asigna para ello a una marca de referencia. Dicha marca representa una posición mecánica conocida.

El proceso de referenciado se lleva a cabo de acuerdo con el modo de operación seleccionado en el parámetro "Mode" y con la configuración realizada en "Objeto tecnológico > Configuración > Parámetros avanzados > Referenciado".

Para los valores dinámicos aceleración, deceleración y tirón, se utilizan los valores predeterminados de "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica".

Se aplica a

- eje de posicionamiento
- encóder externo

La tabla siguiente muestra los modos de operación que pueden utilizarse con los correspondientes objetos tecnológicos:

Modo de operación	Eje de posicionamiento con encóder incremental	Eje de posicionamiento con encóder absoluto	Encóder externo incremental	Encóder externo absoluto
Referenciado activo ("Mode" = 4, 5)	X	-	-	-
Referenciado pasivo ("Mode" = 2, 3, 8)	X	-	X	-
Ajuste de un valor de posición ("Mode" = 0)	X	X	X	X
Desplazamiento relativo del valor real ("Mode" = 1)	X	X	X	X
Referenciar encóder absoluto ("Mode" = 6, 7)	-	X	-	X

Requisitos

- El objeto tecnológico se ha configurado correctamente.
- "Mode" = 2, 3, 4, 5, 8
  El objeto tecnológico debe estar habilitado.
- "Mode" = 0, 1, 2, 6, 7
  Los valores del encóder deben ser válidos. (<TO>.StatusSensor[n].State = 2)

Comportamiento de relevo

- Una petición "MC_Home" de referenciado pasivo se cancela con:
  - Bloqueo del objeto tecnológico poniendo "MC_Power.Enable" = FALSE
  - "Petición "MC_Home" con el parámetro "Mode" = 4, 5, 9
- Una petición "MC_Home" de referenciado pasivo no cancela ninguna otra petición de Motion Control.
- Una petición "MC_Home" de referenciado activo se cancela con:
  - Bloqueo del objeto tecnológico "MC_Power.Enable" = FALSE
  - "Petición "MC_Home" "Mode" = 4, 5
  - "Petición "MC_Halt"
  - "Petición "MC_MoveAbsolute"
  - "Petición "MC_MoveRelative"
  - "Petición "MC_MoveVelocity"
  - "Petición "MC_MoveJog"
- Una petición "MC_Home" de referenciado activo cancela las siguientes peticiones de Motion Control en curso:
  - "Petición "MC_Home" con el parámetro "Mode" = 2, 3, 4, 5
  - "Petición "MC_Halt"
  - "Petición "MC_MoveAbsolute"
  - "Petición "MC_MoveRelative"
  - "Petición "MC_MoveVelocity"
  - "Petición "MC_MoveJog"

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MC_Home" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción
Axis	InOut	TO_Axis	-	Objeto tecnológico
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Position	INPUT	LREAL	0.0	El valor indicado se utiliza de acuerdo con el "Mode" seleccionado.	
Mode	INPUT	INT	4	Modo de operación	
				0	Referenciado directo (absoluto) La posición actual del objeto tecnológico se pone al valor del parámetro "Position".
				1	Referenciado directo (relativo) La posición actual del objeto tecnológico se mueve de acuerdo con el valor del parámetro "Position".
				2	Referenciado pasivo Al detectar la marca de referencia, el valor real se pone al valor del parámetro "Position".
				3	Referenciado pasivo (parámetro "Position" inactivo) Al detectar la marca de referencia, el valor real se pone al punto de referencia configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Referenciado > Referenciado pasivo". (<TO>.Homing.HomePosition)
				4	Referenciado activo El eje de posicionamiento TO ejecuta un movimiento de referenciado según la configuración. Una vez concluido el movimiento, el eje se encuentra en el valor del parámetro "Position".
				5	Referenciado activo (parámetro "Position" inactivo) El eje de posicionamiento TO ejecuta un movimiento de referenciado según la configuración. Una vez concluido el movimiento, el eje se encuentra en el punto de referencia configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Referenciado > Referenciado activo". (<TO>.Homing.HomePosition)
				6	Reajuste del encóder absoluto (relativo) La posición actual se desplaza al valor del parámetro "Position". El offset calculado del valor absoluto se guarda de forma remanente en la CPU. (<TO>.StatusSensor[n].AbsEncoderOffset)

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
				7	Reajuste del encóder absoluto (absoluto) La posición actual se pone al valor del parámetro "Position". El offset calculado del valor absoluto se guarda de forma remanente en la CPU. (<TO>.StatusSensor[n].AbsEncoderOffset)
				8	Referenciado pasivo (sin desactivación) La función es la misma que en Mode 2 pero con la diferencia de que el estado "referenciado" no se desactiva al activar la función.
				9	Referenciado pasivo cancelado Una petición en curso de referenciado pasivo se cancela.
Done	OUTPUT	BOOL	FALSE	TRUE	La petición ha finalizado.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La petición se rechaza. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Desactivación del estado "referenciado"

El estado "referenciado" de un objeto tecnológico se desactiva cuando se dan las siguientes condiciones (<TO>.StatusWord.HomingDone):

- **Objetos tecnológicos con valores reales incrementales:**
  - Inicio de una petición "MC_Home" con "Mode" = 2, 3, 4, 5
    (El estado "referenciado" se activa de nuevo una vez concluido correctamente el proceso de referenciado.)
  - Error en el sistema del encóder o fallo del encóder
  - Reinicialización del objeto tecnológico
  - Tras POWER OFF -> POWER ON de la CPU
  - Borrado total
  - Cambio de la configuración del encóder
- **Objetos tecnológicos con valores reales absolutos:**
  - Restablecimiento de la configuración de fábrica de la CPU
  - Cambio de la configuración del encóder
  - Sustitución de la CPU

Referenciado de un objeto tecnológico con "Mode" = 1..8

Para referenciar un objeto tecnológico, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Indique la función de referenciado deseada en el parámetro "Mode".
3. Asigne valores a los parámetros necesarios e inicie el referenciado mediante un flanco ascendente en el parámetro "Execute"

Si el parámetro "Done" presenta el valor TRUE, la petición "MC_Home" ha concluido según el "Mode" seleccionado. El estado "referenciado" del objeto tecnológico se muestra en "Objeto tecnológico > Diagnóstico > Bits de estado y de error > Estado movimiento > Referenciado (<TO>.StatusWord.HomingDone).

Cancelación de un proceso de referenciado pasivo con "Mode" = 9

El objeto tecnológico no se referencia cuando "Mode" = 9. Cuando una petición "MC_Home" en curso de referenciado pasivo ("Mode" = 2, 3, 8) es relevada por otra petición "MC_Home" con "Mode" = 9, la petición en curso se cancela si el parámetro "CommandAborted" = TRUE. La petición que realiza el relevo con "Mode" = 9 notifica la ejecución correcta con el parámetro "Done" = TRUE.

Información adicional

Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10752).

Consulte también

Identificación de error (Página 10844)

MC_MoveJog

MC_MoveJog: mover ejes en modo Jog

Descripción

La instrucción "MC_MoveJog" de Motion Control mueve un eje en modo Jog.

Con los parámetros "Velocity", "Jerk", "Acceleration" y "Deceleration" se especifica el comportamiento dinámico durante la operación de movimiento.

- Eje de posicionamiento:
  En el parámetro "Velocity" se especifica una velocidad.
- Eje de velocidad de giro:
  En el parámetro "Velocity" se especifica una velocidad de giro.

Se aplica a

- eje de posicionamiento
- eje de velocidad de giro

Requisitos

- El objeto tecnológico se ha configurado correctamente.
- El objeto tecnológico está habilitado.

Comportamiento de relevo

La petición "MC_MoveJog" se cancela con:

- Bloqueo del eje poniendo "MC_Power.Enable" = FALSE
- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

El inicio de una petición "MC_MoveJog" cancela las siguientes peticiones de Motion Control en curso:

- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MC_MoveJog" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción
Axis	InOut	TO_SpeedAxis	-	Objeto tecnológico
JogForward	INPUT	BOOL	FALSE	Mientras el parámetro sea TRUE, el eje se moverá a la velocidad predeterminada en el parámetro "Velocity" en sentido positivo.
JogBackward	INPUT	BOOL	FALSE	Mientras el parámetro sea TRUE, el eje se moverá a la velocidad predeterminada en el parámetro "Velocity" en sentido negativo.
Velocity	INPUT	LREAL	100.0	Consigna de velocidad / de velocidad de giro para la operación de movimiento Valor > 0.0: se utiliza el valor indicado. Valor < 0.0: se utiliza la cantidad del valor indicado. ("Velocity" = 0.0 está permitido)
Acceleration	INPUT	LREAL	-1.0	Aceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la aceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Deceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la deceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Deceleration)

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Jerk	INPUT	LREAL	-1.0	Tirón Valor > 0.0: perfil de velocidad con aceleración constante, se utiliza el tirón indicado. Valor = 0.0: perfil de velocidad trapezoidal Valor < 0.0: se utiliza el tirón configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Jerk)	
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Se ha alcanzado la consigna de velocidad / de velocidad de giro y se mantiene.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La petición se rechaza. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Comportamiento con consigna de velocidad / de velocidad de giro cero ("Velocity" = 0.0)

Una petición "MC_MoveJog" con "Velocity" = 0.0 detiene el eje con la deceleración configurada. Al alcanzar la consigna de velocidad / de velocidad de giro cero, se muestra el valor TRUE en el parámetro "InVelocity".

En "Objeto tecnológico > Diagnóstico > Bits de estado y de error > Estado movimiento" se muestra "Velocidad constante" y "Velocidad cero" (<TO>.StatusWord.ConstantVelocity; <TO>.StatusWord.Standstill).

Desplazamiento de un eje en modo Jog

Para mover un eje en modo Jog, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Mueva el eje en sentido positivo con "JogForward" o en sentido negativo con "JogBackward".

En los parámetros "Busy", "InVelocity" y "Error" se muestra el estado de movimiento actual.

Nota

Comportamiento en caso de cambio de la corrección

Si la velocidad / velocidad de giro se ve influida por un cambio de la corrección (<TO>.Override.Velocity) durante el movimiento constante, el parámetro "InVelocity" se desactiva durante la aceleración o deceleración. Cuando se alcanza la velocidad nueva ("Velocity" x "Override" %) vuelve a activarse "InVelocity".

Información adicional

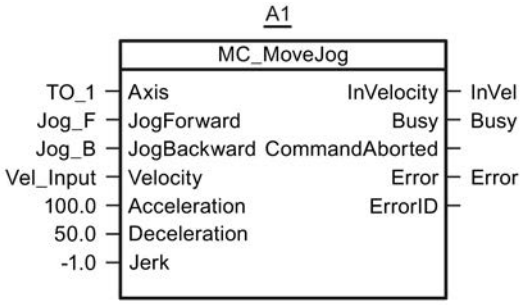
Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10752).

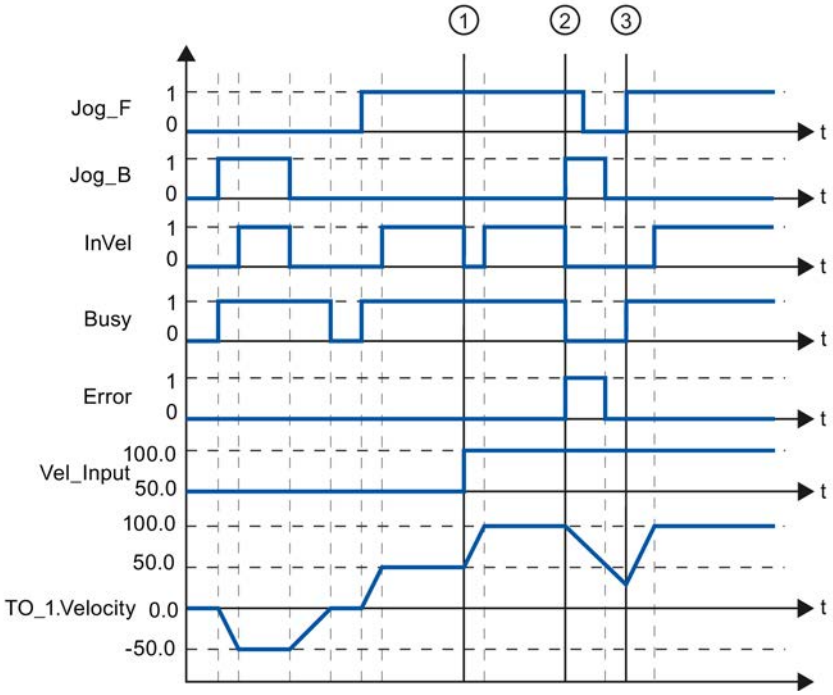
Consulte también

Identificación de error (Página 10844)

MC_MoveJog: Diagrama de funcionamiento

Diagrama de funcionamiento: desplazamiento de un eje en modo Jog





"Jog_B" permite mover el eje en sentido negativo en modo Jog. En cuanto se alcanza la consigna de velocidad -50.0, el sistema lo notifica mediante "InVel" = TRUE. Tras desactivar

"Jog_B" el eje frena hasta pararse del todo. Seguidamente, "Jog_F" mueve el eje en sentido positivo. En cuanto se alcanza la consigna de velocidad 50.0, el sistema lo notifica mediante "InVel" = TRUE.

En el punto ① y si está activado "Jog_F", la consigna de velocidad cambia a 100.0 a través de "Vel_Input". Como alternativa, es posible cambiar la velocidad consignada mediante corrección de la velocidad. "InVel" se desactiva. El eje se acelera. En cuanto se alcanza la nueva consigna de velocidad 100.0, el sistema lo notifica mediante "InVel" = TRUE.

Si "Jog_F" está activado, en el punto ② también se activa "Jog_B". Si "Jog_F" y "Jog_B" están activados, el eje se frena con la última deceleración válida. Con "Error" se visualiza un error y en la salida "ErrorID" se emite el error 16#8007 (indicación de sentido errónea).

Este error se soluciona desactivando ambas salidas "Jog_F" y "Jog_B".

Ya durante la rampa de frencado se activa "Jog_F" en el punto ③. El eje se acelera hasta la última velocidad configurada. En cuanto se alcanza la consigna de velocidad 100.0, el sistema lo notifica mediante "InVel" = TRUE.

MC_MoveVelocity

MC_MoveVelocity: mover ejes con especificación de velocidad

Descripción

La instrucción "MC_MoveVelocity" de Motion Control mueve un eje a velocidad constante.

Con los parámetros "Velocity", "Jerk", "Acceleration" y "Deceleration" se especifica el comportamiento dinámico durante la operación de movimiento.

- Eje de posicionamiento:
  En el parámetro "Velocity" se especifica una velocidad.
- Eje de velocidad de giro:
  En el parámetro "Velocity" se especifica una velocidad de giro.

Se aplica a

- eje de posicionamiento
- eje de velocidad de giro

Requisitos

- El objeto tecnológico se ha configurado correctamente.
- El objeto tecnológico está habilitado.

Comportamiento de relevo

La petición "MC_MoveVelocity" se cancela con:

- Bloqueo del eje poniendo "MC_Power.Enable" = FALSE
- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

El inicio de una petición "MC_MoveVelocity" cancela las siguientes peticiones de Motion Control en curso:

- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MC_MoveVelocity" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción
Axis	InOut	TO_SpeedAxis	-	Objeto tecnológico
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente
Velocity	INPUT	LREAL	100.0	Consigna de velocidad / de velocidad de giro para la operación de movimiento ("Velocity" = 0.0 está permitido)
Acceleration	INPUT	LREAL	-1.0	Aceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la aceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Acceleration)

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Deceleration	INPUT	LREAL	-1.0	Deceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la deceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Tirón Valor > 0.0: perfil de velocidad con aceleración constante, se utiliza el tirón indicado. Valor = 0.0: perfil de velocidad trapezoidal Valor < 0.0: se utiliza el tirón configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Jerk)	
Direction	INPUT	INT	0	Sentido de giro del eje	
				0	El signo de la velocidad indicada en el parámetro "Velocity" define el sentido de giro.
				1	Sentido de giro positivo Se utiliza el valor de "Velocity".
				2	Sentido de giro negativo Se utiliza el valor de "Velocity".
Current	INPUT	BOOL	FALSE	Mantener velocidad actual	
				FALSE	Desactivado Se tienen en cuenta los valores de los parámetros "Velocity" y "Direction".
				TRUE	Activado Los valores de los parámetros "Velocity" y "Direction" **no** se tienen en cuenta. Se mantienen la velocidad y el sentido que había al iniciar la función. En cuanto el eje vuelve a moverse a la velocidad que había al iniciar la función, el parámetro "InVelocity" proporciona el valor TRUE.
InVelocity	OUTPUT	BOOL	FALSE	TRUE	Se ha alcanzado la consigna de velocidad / de velocidad de giro y se mantiene.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Comportamiento con consigna de velocidad / de velocidad de giro cero ("Velocity" = 0.0)

Una petición "MC_MoveVelocity" con "Velocity" = 0.0 detiene el eje con la deceleración configurada. Al alcanzar la consigna de velocidad / de velocidad de giro cero, se muestra el valor TRUE en el parámetro "InVelocity".

En "Objeto tecnológico > Diagnóstico > Bits de estado y de error > Estado movimiento" se muestra "Velocidad constante" y "Velocidad cero" (<TO>.StatusWord.ConstantVelocity; <TO>.StatusWord.Standstill).

Los parámetros "InVelocity" y "Busy" presentan el valor TRUE, hasta que la petición "MC_MoveVelocity" es relevada por otra petición de Motion Control.

Desplazamiento de un eje con velocidad / velocidad de giro constante

Para mover un eje a velocidad / velocidad de giro constante, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. En el parámetro "Velocity", indique la velocidad / velocidad de giro con la que debe moverse el eje.
3. Inicie la petición "MC_MoveVelocity" con un flanco ascendente en el parámetro "Execute".

En los parámetros "Busy", "InVelocity" y "Error" se muestra el estado de movimiento actual.

Si el parámetro "InVelocity" presenta el valor TRUE, significa que se ha alcanzado la consigna de velocidad / de velocidad de giro. El eje continúa moviéndose de forma constante a esta velocidad. Los parámetros "InVelocity" y "Busy" presentan el valor TRUE, hasta que la petición "MC_MoveVelocity" es relevada por otra petición de Motion Control.

Nota

Comportamiento en caso de cambio de la corrección

Si la velocidad / velocidad de giro se ve influida por un cambio de la corrección (<TO>.Override.Velocity) durante el movimiento constante, el parámetro "InVelocity" se desactiva durante la aceleración o deceleración. Cuando se alcanza la velocidad / velocidad de giro nueva ("Velocity" x "Override" %) vuelve a activarse "InVelocity".

Información adicional

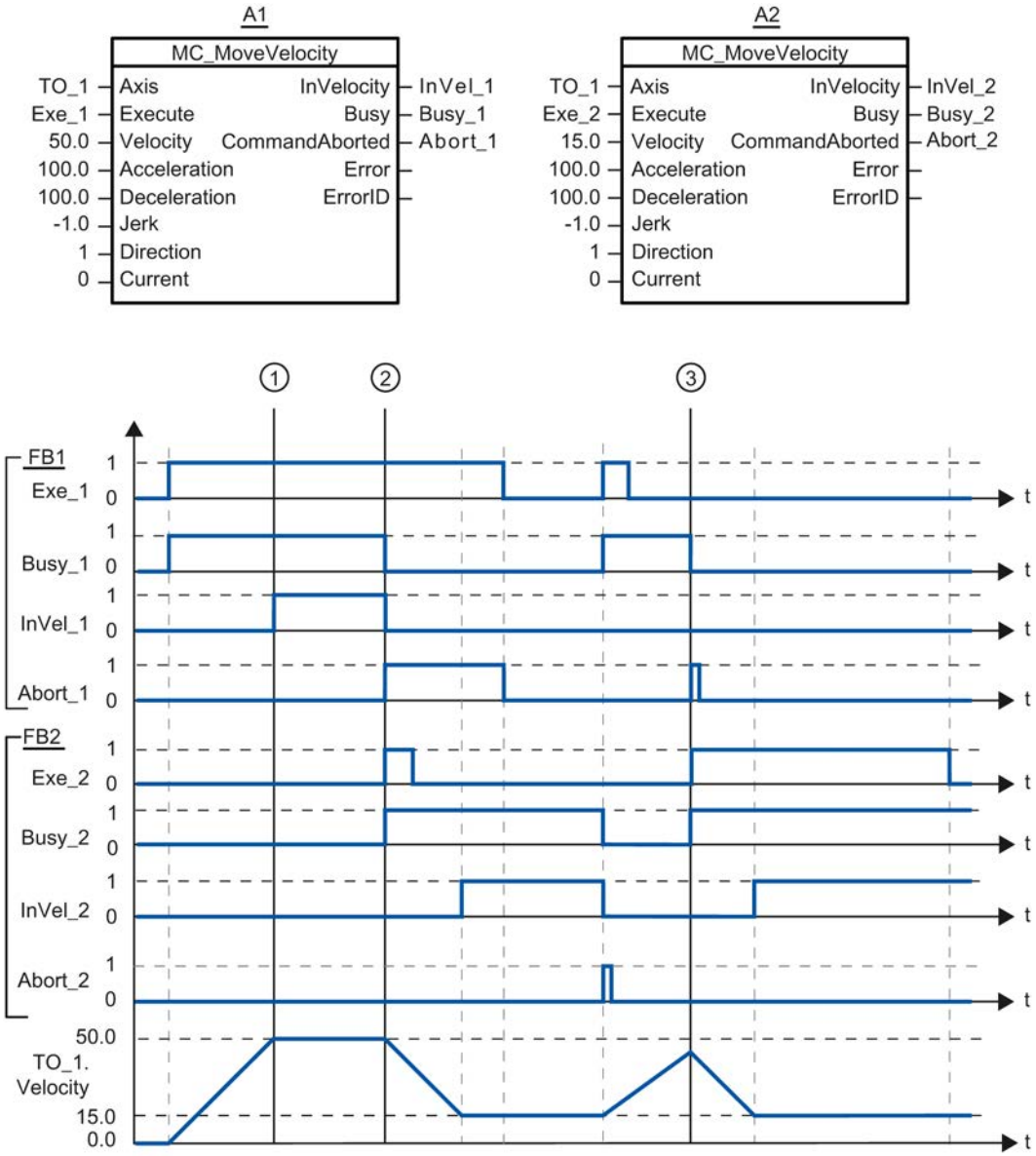
Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10752).

Consulte también

Identificación de error (Página 10844)

MC_MoveVelocity: Diagrama de funcionamiento

Diagrama de funcionamiento: desplazamiento de un eje con especificación de velocidad y comportamiento de relevo de la petición



Una petición "MC_MoveVelocity" lanzada mediante "Exe_1" (A1) acelera el eje y notifica que se ha alcanzado la consigna de velocidad 50.0 mediante "InVel_1" en el instante ①.

En el instante ② la petición es relevada por otra petición "MC_MoveVelocity" (A2). La cancelación se notifica mediante "Abort_1". En cuanto se alcanza la nueva consigna de velocidad 15.0 el sistema lo notifica mediante "InVel_2". A continuación el eje continúa moviéndose de forma constante a la velocidad 15.0.

La petición "MC_MoveVelocity" en curso (A2) es relevada por otra petición "MC_MoveVelocity" (A1). La cancelación se notifica mediante "Abort_2". El eje se acelera a la nueva consigna de velocidad 50.0. Antes de alcanzar la consigna de velocidad, la petición "MC_MoveVelocity" actual (A1) es relevada por otra petición "MC_MoveVelocity" (A2) en el instante ③. La cancelación se notifica mediante "Abort_1". En cuanto se alcanza la nueva consigna de velocidad 15.0 el sistema lo notifica mediante "InVel_2". A continuación el eje continúa moviéndose de forma constante a la velocidad 15.0.

MC_MoveRelative

MC_MoveRelative: Posicionar ejes de forma relativa

Descripción

La instrucción "MC_MoveRelative" de Motion Control mueve un eje en relación a la posición que hay al iniciar el procesamiento de la petición.

Con los parámetros "Velocity", "Jerk", "Acceleration" y "Deceleration" se especifica el comportamiento dinámico durante la operación de movimiento.

Se aplica a

- eje de posicionamiento

Requisitos

- El objeto tecnológico se ha configurado correctamente.
- El objeto tecnológico está habilitado.

Comportamiento de relevo

La petición "MC_MoveRelative" se cancela con:

- Bloqueo del eje poniendo "MC_Power.Enable" = FALSE
- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

El inicio de una petición "MC_MoveRelative" cancela las siguientes peticiones de Motion Control en curso:

- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"

- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MC_MoveRelative" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción
Axis	InOut	TO_Positioning Axis	-	Objeto tecnológico
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente
Distance	INPUT	LREAL	0.0	Trayecto de posicionamiento (negativo o positivo)
Velocity	INPUT	LREAL	-1.0	Consigna de velocidad para el posicionamiento Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la velocidad configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Aceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la aceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Deceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la deceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Deceleration)
Jerk	INPUT	LREAL	-1.0	Tirón Valor > 0.0: perfil de velocidad con aceleración constante, se utiliza el tirón indicado Valor = 0.0: perfil de velocidad trapezoidal Valor < 0.0: se utiliza el tirón configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Jerk)

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Done	OUTPUT	BOOL	FALSE	TRUE	Posición de destino alcanzada
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La petición se rechaza. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Desplazamiento de un eje en relación a la posición inicial

Para mover un eje en relación a la posición inicial, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Indique el trayecto que debe recorrerse en el parámetro "Distance".
3. Inicie la petición "MC_MoveRelative" con un flanco ascendente en el parámetro "Execute".

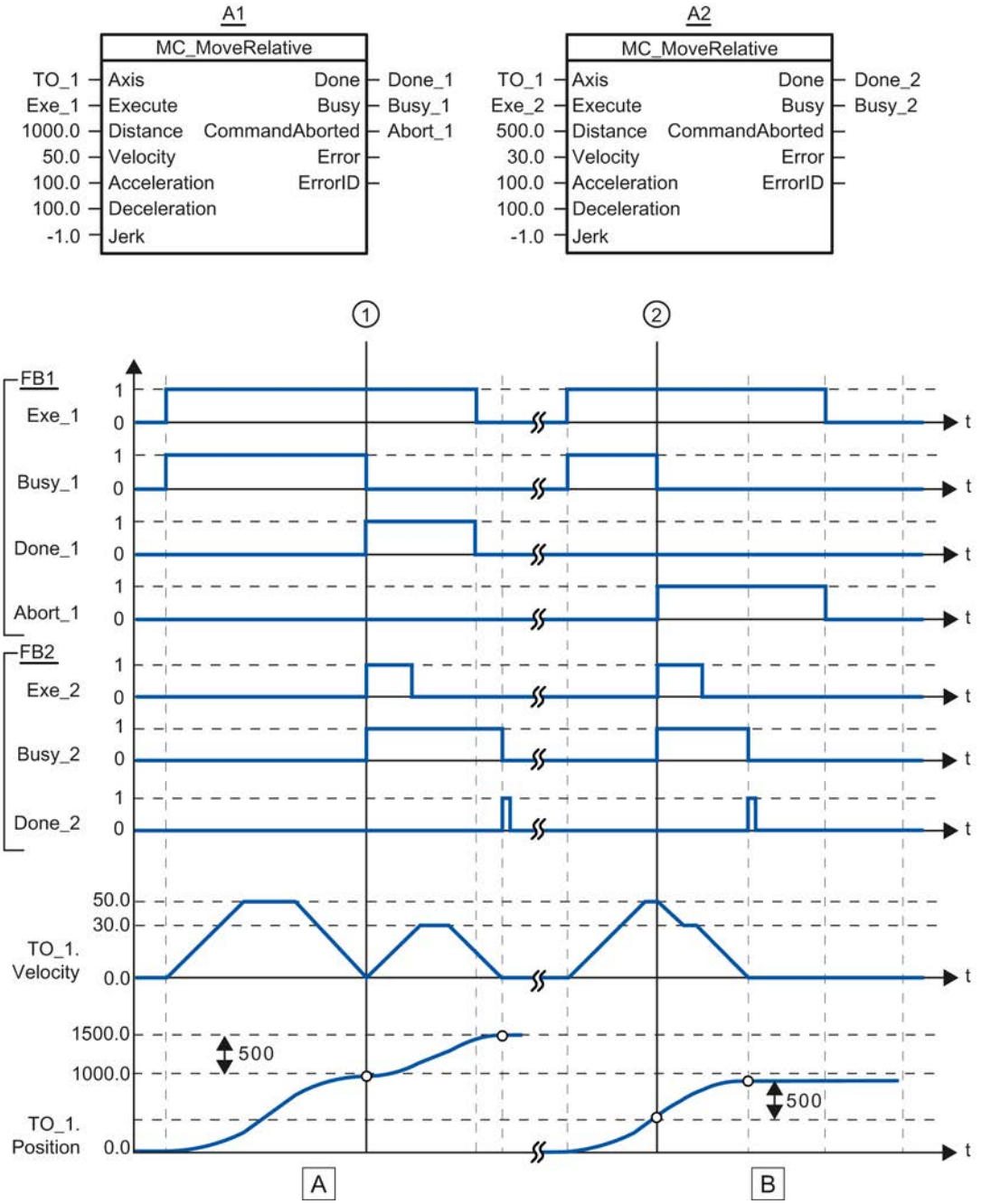
En los parámetros "Busy", "Done" y "Error" se muestra el estado de movimiento actual.

Consulte también

Identificación de error (Página 10844)

MC_MoveRelative: Diagrama de funcionamiento

Diagrama de funcionamiento: posicionamiento relativo de un eje y comportamiento de relevo de la petición



Sección A	Con una petición "MC_MoveRelative" (A1), el eje recorre el trayecto ("Distance") 1000.0 (en este caso, la posición inicial es 0.0). En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_1" en el instante ①. En este mismo instante ① se inicia otra petición "MC_MoveRelative" (A2) con el trayecto 500.0. En cuanto se alcanza con éxito la nueva posición de destino el sistema lo notifica mediante "Done_2". Puesto que previamente se había desactivado "Exe_2", "Done_2" solo está pendiente durante un ciclo.
Sección B	Una petición "MC_MoveRelative" en curso (A1) es relevada por otra petición "MC_MoveRelative" (A2). La cancelación se notifica mediante "Abort_1" en el instante ②. A continuación, el eje se mueve con la nueva velocidad hasta recorrer el trayecto ("Distance") 500.0. En cuanto se alcanza la nueva posición de destino el sistema lo notifica mediante "Done_2".

MC_MoveAbsolute

MC_MoveAbsolute: Posicionar ejes de forma absoluta

Descripción

La instrucción "MC_MoveAbsolute" de Motion Control mueve un eje hasta una posición absoluta.

Con los parámetros "Velocity", "Jerk", "Acceleration" y "Deceleration" se especifica el comportamiento dinámico durante la operación de movimiento.

Se aplica a

- eje de posicionamiento

Requisitos

- El objeto tecnológico se ha configurado correctamente.
- El objeto tecnológico está habilitado.
- El objeto tecnológico está referenciado.

Comportamiento de relevo

La petición "MC_MoveAbsolute" se cancela con:

- Bloqueo del eje poniendo "MC_Power.Enable" = FALSE
- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

El inicio de una petición "MC_MoveAbsolute" cancela las siguientes peticiones de Motion Control en curso:

- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MC_MoveAbsolute" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción
Axis	InOut	TO_Positioning Axis	-	Objeto tecnológico
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente
Position	INPUT	REAL	0.0	Posición absoluta de destino
Velocity	INPUT	LREAL	-1.0	Consigna de velocidad para el posicionamiento Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la velocidad configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Velocity)
Acceleration	INPUT	LREAL	-1.0	Aceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la aceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Acceleration)
Deceleration	INPUT	LREAL	-1.0	Deceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la deceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Deceleration)

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Jerk	INPUT	LREAL	-1.0	Tirón Valor > 0.0: perfil de velocidad con aceleración constante, se utiliza el tirón indicado Valor = 0.0: perfil de velocidad trapezoidal Valor < 0.0: se utiliza el tirón configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Jerk)	
Direction	INPUT	INT	3	Sentido de giro del eje Solo se evalúa si "Módulo" está activado. "Objeto tecnológico > Configuración > Parámetros básicos > Activar módulo"	
				1	Sentido de giro positivo
				2	Sentido de giro negativo
				3	Camino más corto
Done	OUTPUT	BOOL	FALSE	TRUE	Posición de destino alcanzada.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso.
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La petición se rechaza. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Desplazamiento de un eje hasta una posición absoluta

Para mover un eje hasta una posición absoluta, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Indique la posición de destino deseada en el parámetro "Position".
3. Inicie la petición "MC_MoveAbsolute" con un flanco ascendente en el parámetro "Execute".

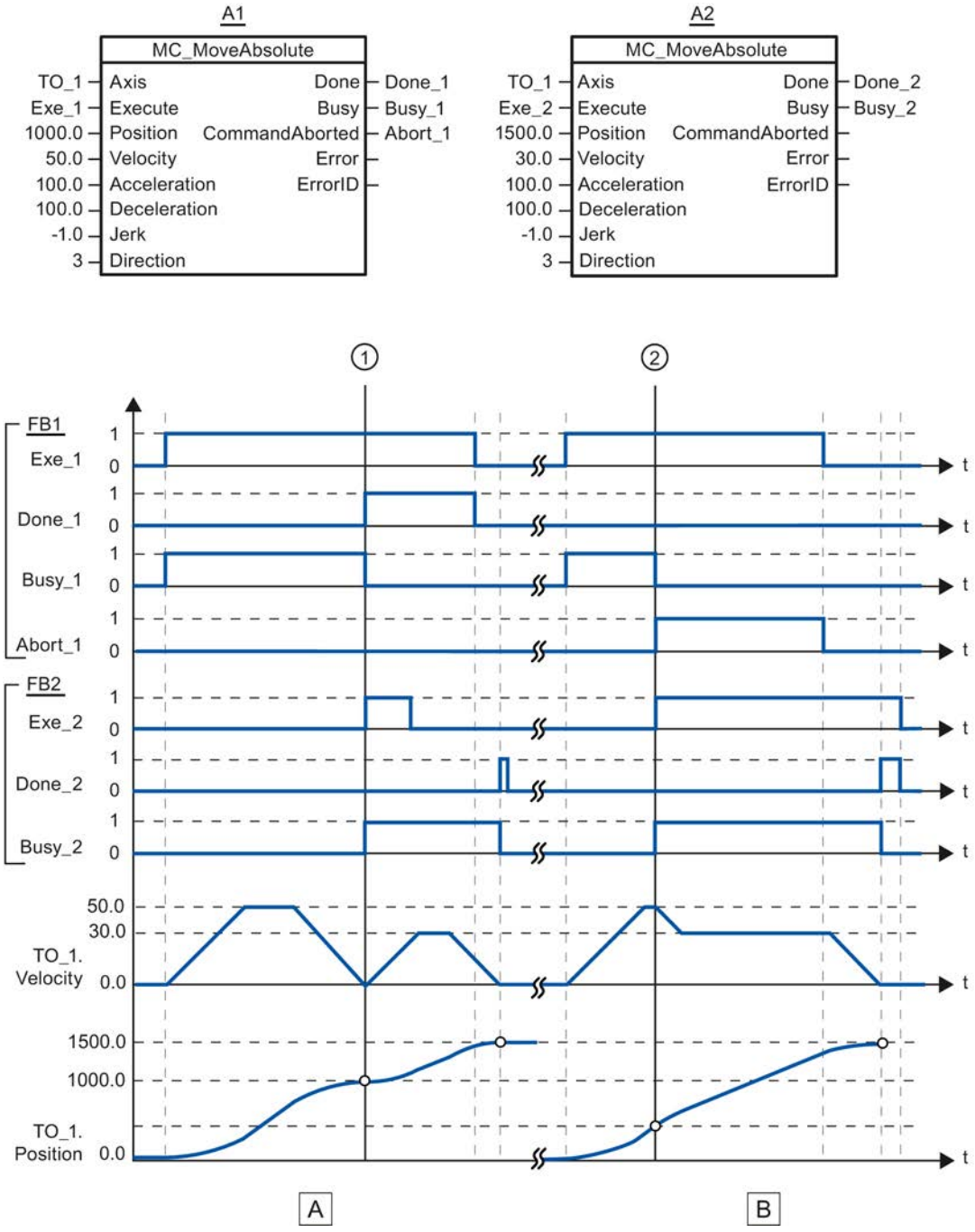
En los parámetros "Busy", "Done" y "Error" se muestra el estado de movimiento actual.

Consulte también

Identificación de error (Página 10844)

MC_MoveAbsolute: Diagrama de funcionamiento

Diagrama de funcionamiento: posicionamiento absoluto de un eje y comportamiento de relevo de la petición



Sección A	Con una petición "MC_MoveAbsolute" (A1), un eje se mueve hasta la posición absoluta 1000.0. En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_1" en el instante ①. En este mismo instante ① se inicia otra petición "MC_MoveAbsolute" (A2) con la posición de destino 1500.0. En cuanto se alcanza la posición de destino 1500.0 el sistema lo notifica mediante "Done_2". Puesto que previamente se había desactivado "Exe_2", "Done_2" solo está pendiente durante un ciclo.
Sección B	Una petición "MC_MoveAbsolute" en curso (A1) es relevada por otra petición "MC_MoveAbsolute" (A2) en el instante ②. La cancelación se notifica mediante "Abort_1". El eje se frena hasta la velocidad modificada y se mueve hasta la nueva posición de destino 1500.0. En cuanto se alcanza la nueva posición de destino el sistema lo notifica mediante "Done_2".

MC_Halt

MC_Halt: Parar ejes

Descripción

La instrucción "MC_Halt" de Motion Control frena un eje hasta que se para.

Con los parámetros "Jerk" y "Deceleration" se especifica el comportamiento dinámico durante la operación de deceleración.

Se aplica a

- eje de posicionamiento
- eje de velocidad de giro

Requisitos

- El objeto tecnológico se ha configurado correctamente.
- El objeto tecnológico está habilitado.

Comportamiento de relevo

La petición "MC_Halt" se cancela con:

- Bloqueo del eje poniendo "MC_Power.Enable" = FALSE
- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

El inicio de una petición "MC_Halt" cancela las siguientes peticiones de Motion Control en curso:

- "Petición "MC_Home" "Mode" = 4, 5
- "Petición "MC_Halt"
- "Petición "MC_MoveAbsolute"
- "Petición "MC_MoveRelative"
- "Petición "MC_MoveVelocity"
- "Petición "MC_MoveJog"

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MC_Halt" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	InOut	TO_SpeedAxis	-	Objeto tecnológico	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Deceleration	INPUT	LREAL	-1.0	Deceleración Valor > 0.0: se utiliza el valor indicado. Valor = 0.0: no admisible Valor < 0.0: se utiliza la deceleración configurada en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Deceleration)	
Jerk	INPUT	LREAL	-1.0	Tirón Valor > 0.0: perfil de velocidad con aceleración constante, se utiliza el tirón indicado Valor = 0.0: perfil de velocidad trapezoidal Valor < 0.0: se utiliza el tirón configurado en "Objeto tecnológico > Configuración > Parámetros avanzados > Preajuste de dinámica". (<TO>.DynamicDefaults.Jerk)	
Done	OUTPUT	BOOL	FALSE	TRUE	Se ha alcanzado la velocidad cero
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La petición se rechaza. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Deceleración de un eje con "MC_Halt"

Para frenar un eje hasta pararlo, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Asigne los valores deseados a los parámetros "Deceleration" y "Jerk".
3. Inicie la petición "MC_Halt" con un flanco ascendente en el parámetro "Execute".

En los parámetros "Busy", "Done" y "Error" se muestra el estado de movimiento actual. La velocidad cero del eje se muestra en "Objeto tecnológico > Diagnóstico > Bits de estado y de error > Estado movimiento > Velocidad cero" (<TO>.StatusWord.Standstill).

Información adicional

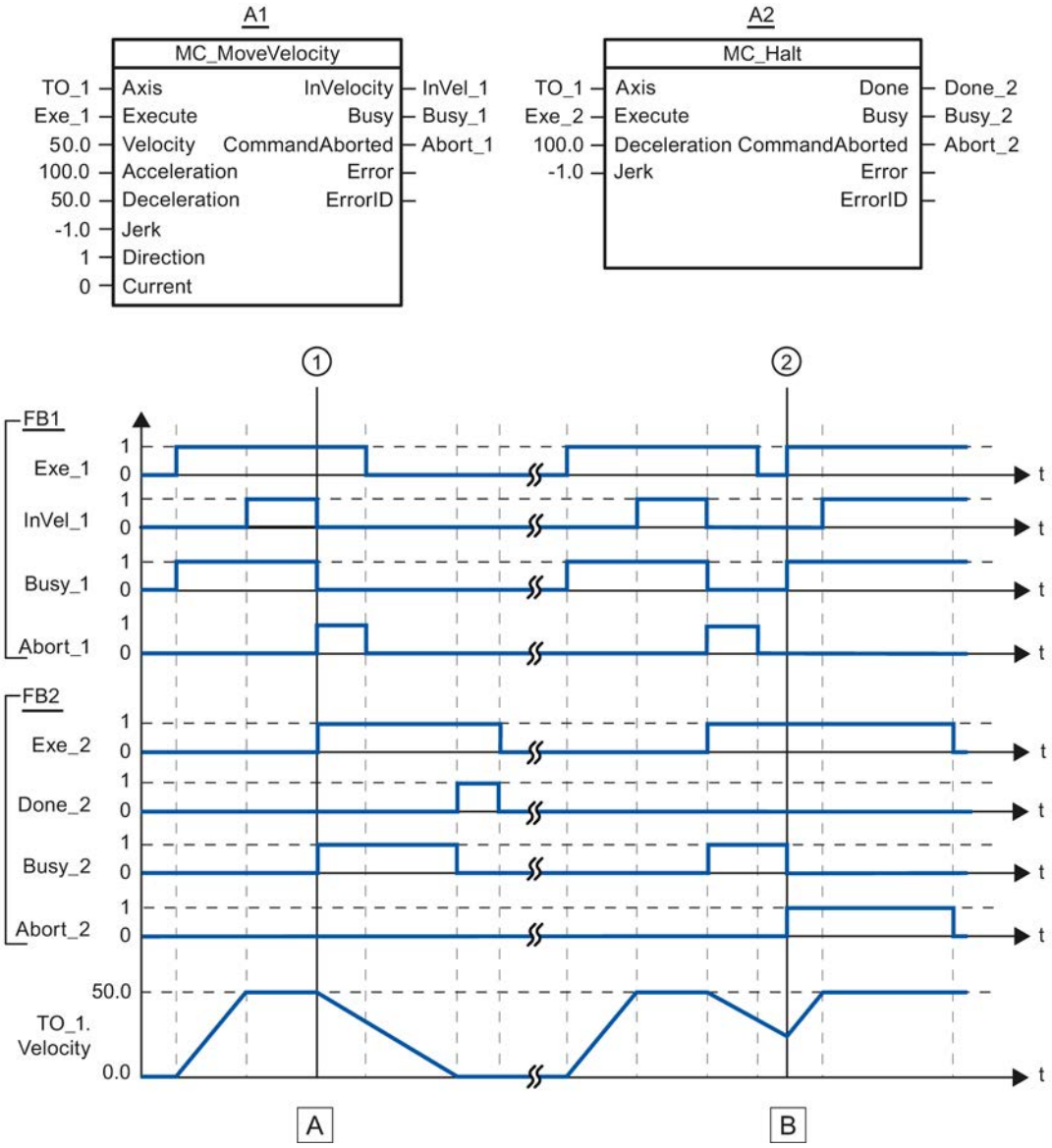
Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10752).

Consulte también

Identificación de error (Página 10844)

MC_Halt: Diagrama de funcionamiento

Diagrama de funcionamiento: parada de un eje y comportamiento de relevo de la petición



Sección A	Un eje se mueve con una petición "MC_MoveVelocity" (A1). En cuanto se alcanza la consigna de velocidad 50.0, el sistema lo notifica mediante "InVel_1". En el instante ① la petición "MC_MoveVelocity" es relevada por una petición "MC_Halt" (A2). La cancelación de la petición se notifica mediante "Abort_1". El eje se frena hasta pararse. En cuanto finaliza correctamente la petición "MC_Halt" el sistema lo notifica mediante "Done_2".
Sección B	El eje se mueve con una petición "MC_MoveVelocity" (A1). En cuanto se alcanza la consigna de velocidad 50.0, el sistema lo notifica mediante "InVel_1". Seguidamente, la petición "MC_MoveVelocity" es relevada por una petición "MC_Halt" (A2). La cancelación de la petición se notifica mediante "Abort_1". Durante la operación de deceleración, la petición "MC_Halt" es relevada por una petición "MC_MoveVelocity" (A1) en el instante ②. La cancelación de la petición se notifica mediante "Abort_2". En cuanto se alcanza la consigna de velocidad 50.0, el sistema lo notifica mediante "InVel_1". Seguidamente, el eje se mueve a velocidad constante.

MC_Reset

MC_Reset: acusar alarmas, reinicializar objetos tecnológicos

Descripción

La instrucción "MC_Reset" de Motion Control acusa todas las alarmas tecnológicas acusables del programa de usuario. Al acusar se borran también los bits "Error" y "Warning" del bloque de datos tecnológico.

La instrucción "MC_Reset" de Motion Control con "Restart" = TRUE lanza la reinicialización (restart) de objetos tecnológicos. Con la reinicialización del objeto tecnológico se aplican datos de configuración nuevos en el bloque de datos tecnológico.

Se aplica a

- eje de posicionamiento
- eje de velocidad de giro
- encóder externo

Requisitos

- Se ha eliminado la causa de error de una alarma tecnológica pendiente.
- Para una reinicialización, el objeto tecnológico debe estar bloqueado. ("MC_Power.Status" = FALSE y "MC_Power.Busy" = FALSE)

Comportamiento de relevo

- Parámetro "Restart" = FALSE:

El procesamiento de la instrucción "MC_Reset" puede ser cancelado por otras peticiones de Motion Control. La petición "MC_Reset" no cancela peticiones de Motion Control en curso.

- Parámetro "Restart" = TRUE:

El procesamiento de la instrucción "MC_Reset" con el parámetro "Restart" = TRUE no puede ser cancelado por ninguna otra petición de Motion Control.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MC_Reset" de Motion Control:

Parámetro	Declaración	Tipo de datos	Valor por defecto	Descripción	
Axis	InOut	TO_Axis	-	Objeto tecnológico	
Execute	INPUT	BOOL	FALSE	Inicio de la petición con flanco ascendente	
Restart	INPUT	BOOL	FALSE	TRUE	"Restart" Reinicialización del objeto tecnológico y acuse de alarmas tecnológicas pendientes. El objeto tecnológico se reinicializa con los valores iniciales configurados.
				FALSE	Acuse de alarmas tecnológicas pendientes.
Done	OUTPUT	BOOL	FALSE	TRUE	El error se ha acusado. Se ha ejecutado la reinicialización.
Busy	OUTPUT	BOOL	FALSE	TRUE	Petición en proceso
CommandAborted	OUTPUT	BOOL	FALSE	TRUE	La petición ha sido cancelada por otra durante su procesamiento.
Error	OUTPUT	BOOL	FALSE	TRUE	Ha ocurrido un error al ejecutar la petición. La petición se rechaza. La causa del error se puede consultar en el parámetro "ErrorID".
ErrorID	OUTPUT	WORD	16#0000	Identificador de error (Página 10844) del parámetro "Error"	

Acusar alarmas tecnológicas

Para acusar alarmas tecnológicas, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Ponga el parámetro "Restart" = FALSE.
3. Inicie el acuse del error mediante un flanco ascendente en el parámetro "Execute".

Si el parámetro "Done" presenta el valor TRUE, el error está acusado.

Nota

Acusar con "Restart" = FALSE

Ponga "Restart" = FALSE si únicamente deben acusarse las alarmas tecnológicas. Durante una reinicialización no es posible utilizar el objeto tecnológico.

Reinicialización de un objeto tecnológico

Para reinicializar un objeto tecnológico, proceda del siguiente modo:

1. Compruebe los requisitos antes mencionados.
2. Ponga el parámetro "Restart" = TRUE.
3. Ejecute la reinicialización mediante un flanco ascendente en el parámetro "Execute".

Si el parámetro "Done" presenta el valor TRUE, la reinicialización del objeto tecnológico ha concluido.

Encontrará más información sobre la reinicialización en el capítulo Reinicialización de objetos tecnológicos (Página 10771).

Consulte también

Identificación de error (Página 10844)

Contadores rápidos

CTRL_HSC: Controlar contadores rápidos

Parámetro

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	INPUT	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	OUTPUT	BOOL	I, Q, M, D, L	Salida de habilitación
HSC	INPUT	HW_HSC	I, Q, M o constante	Identificador de hardware del contador rápido (ID HW)
DIR	INPUT	BOOL	I, Q, M, D, L o constante	Habilitación del nuevo sentido de contaje (v. NEW_DIR)
CV	INPUT	BOOL	I, Q, M, D, L o constante	Habilitación del nuevo valor de contaje (v. NEW_CV)
RV	INPUT	BOOL	I, Q, M, D, L o constante	Habilitación del nuevo valor de referencia (v. NEW_RV)
PERIOD	INPUT	BOOL	I, Q, M, D, L o constante	Habilitación del nuevo periodo de medición de frecuencia (v. NEW_PERIOD)
NEW_DIR	INPUT	INT	I, Q, M, D, L o constante	Sentido de contaje que se carga si DIR = TRUE.
NEW_CV	INPUT	DINT	I, Q, M, D, L o constante	Valor del contador que se carga si CV = TRUE.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NEW_RV	INPUT	DINT	I, Q, M, D, L o constante	Valor de referencia que se carga si RV = TRUE.
NEW_PERIOD	INPUT	INT	I, Q, M, D, L o constante	Periodo de medición de frecuencia que se carga si PERIOD = TRUE.
BUSY	OUTPUT	BOOL	I, Q, M, D, L	Estado de ejecución
STATUS	OUTPUT	WORD	I, Q, M, D, L	Estado de la operación

Descripción

La instrucción "Controlar contadores rápidos" permite parametrizar y controlar los contadores rápidos que soporta la CPU, cargando para ello valores nuevos en los contadores. Para poder ejecutar la instrucción es preciso que esté activado un contador rápido que disponga de control. Por cada contador rápido no es posible ejecutar simultáneamente en el programa varias instrucciones "Controlar contadores rápidos".

La instrucción "Controlar contadores rápidos" permite cargar los siguientes valores de parámetros en un contador rápido:

- Sentido de contaje (NEW_DIR): El sentido de contaje define si un contador rápido debe contar adelante o atrás. Los siguientes valores de la entrada NEW_DIR definen el sentido de contaje: 1 = adelante, -1= atrás.
  El sentido de contaje solo se puede cambiar mediante la instrucción "Controlar contadores rápidos" si el control del sentido de contaje se ha parametrizado para el contador indicado en el programa. El sentido de contaje indicado en la entrada NEW_DIR se carga en un contador rápido si está activado el bit de la entrada DIR.
- Valor de contaje (NEW_CV): El valor de contaje es el valor inicial con el que un contador rápido comienza a contar. El valor de contaje puede estar comprendido en un rango de -2147483648 a 2147483647.
  El valor de contaje indicado en la entrada NEW_CV se carga en un contador rápido si está activado el bit de la entrada CV.
- Valor de referencia (NEW_RV): El valor der referencia puede compararse con el valor de contaje actual para poder desencadenar una alarma. El valor de referencia, igual que el valor de contaje, puede estar comprendido en un rango de -2147483648 a 2147483647.
  El valor de referencia indicado en la entrada NEW_RV se carga en un contador rápido si está activado el bit de la entrada RV.
- Periodo de medición de frecuencia (NEW_PERIOD): Los siguientes valores de la entrada NEW_PERIOD definen el periodo de medición de frecuencia: 10 = 0,01s, 100 = 0,1s, 1000 = 1s.
  Solo es posible actualizar el tiempo si se ha parametrizado la función "Medición de frecuencia" para el contador rápido indicado. El tiempo indicado en la entrada NEW_PERIOD se carga en un contador rápido si está activado el bit de la entrada PERIOD.

La instrucción "Controlar contadores rápidos" se ejecuta solo si el estado lógico de la entrada EN es "1". Mientras se procesa la operación, permanece activado el bit de la salida BUSY. Una vez ejecutada la operación por completo, se desactiva el bit de la salida BUSY.

La salida de habilitación ENO se activa únicamente si el estado lógico de la entrada de habilitación EN es "1" y no ocurren errores al ejecutar la operación.

Cuando se inserta la instrucción "Controlar contadores rápidos" se crea un bloque de datos de instancia en el que se almacenan los datos de la operación.

Parámetro STATUS

En la salida STATUS es posible consultar si han ocurrido errores al ejecutar la instrucción "Controlar contadores rápidos". En la tabla siguiente se explica el significado de los valores que se depositan en la salida STATUS:

Código de error (hexadecimal)	Descripción
0	No han ocurrido errores
80A1	Identificador de hardware no válido del contador rápido
80B1	Sentido de contaje (NEW_DIR) no válido
80B2	Valor de contaje (NEW_CV) no válido
80B3	Valor de referencia (NEW_RV) no válido
80B4	Periodo de medición de frecuencia (NEW_PERIOD) no válido
80C0	Acceso repetido al contador rápido
80D0	El contador rápido (HSC) no está activado en la configuración hardware de la CPU.

Contaje y medición (S7-1500)

High_Speed_Counter

Descripción High_Speed_Counter

Descripción

Con la instrucción High_Speed_Counter se controlan a través del programa del usuario las funciones de contaje y medida del módulo tecnológico.

Llamada

La instrucción High_Speed_Counter debe llamarse una vez por contador en el ciclo o en un programa controlado por tiempo. No se permite realizar la llamada en un programa de interrupción controlado por eventos.

Funcionamiento

Valor de contaje: el valor de contaje está disponible en el parámetro de salida CountValue. El valor de contaje se actualiza cada vez que se llama a la instrucción High_Speed_Counter.

Valor medido: el valor medido está disponible en el parámetro de salida MeasuredValue. El módulo tecnológico vuelve a leer el valor medido cada vez que se llama la instrucción High_Speed_Counter.

El valor medido y el valor de contaje están disponibles simultáneamente en la interfaz de realimentación.

Capture: el parámetro de salida CaptureStatus = TRUE indica un valor de captura (Capture) válido en el parámetro de salida CapturedValue.

- Un valor de Capture se detecta en las siguientes condiciones:
  - Una entrada digital tiene la parametrización "Capture"
  - CaptureEnable = TRUE
  - Flanco en la entrada digital con la función de Capture
- El parámetro de salida CaptureStatus se restablece si se detecta un flanco descendente en el parámetro de entrada CaptureEnable.

Sincronización: El parámetro de salida SyncStatus = TRUE indica que se ha llevado a cabo una sincronización.

- El valor de contaje se sincroniza en las siguientes condiciones:
  - Una entrada digital tiene la parametrización "Sincronización"" o el encóder incremental tiene la parametrización "Sincronización en caso de señal N"
  - SyncEnable = TRUE
  - SyncUpDirection (o SyncDownDirection) = TRUE
  - Flanco en la entrada digital con la función de sincronización o flanco ascendente de la señal N en la entrada del encóder
- El parámetro de salida SyncStatus se restablece si se detecta un flanco descendente en
  - el parámetro de entrada SyncEnable o
  - la variable estática SyncDownDirection o
  - la variable estática SyncUpDirection

Modificación de parámetros a través del programa de usuario

Para modificar los parámetros mediante el programa de usuario, proceda del siguiente modo:

1. Compruebe mediante la variable Set correspondiente si el objeto tecnológico para la modificación de parámetros está listo (variable Set = FALSE) o si aún hay una petición de modificación en curso (variable Set = TRUE).
2. Cuando el objeto tecnológico para la modificación de parámetros esté listo, modifique la variable estática correspondiente.
3. Ajuste la variable Set correspondiente para ejecutar la petición de modificación.
4. Compruebe por medio del parámetro de salida Error si se ha producido un error.
   Si no se ha producido ningún error y el objeto tecnológico ha reseteado automáticamente la variable Set, la modificación de parámetros se ha realizado correctamente.

Límite inferior o superior de contaje: mediante el programa del usuario se modifican los dos límites de la manera siguiente:

1. Comprobación de la variable estática SetUpperLimit (o SetLowerLimit) = FALSE
2. Modificación de la variable estática NewUpperLimit (o NewLowerLimit)
3. SetUpperLimit (o SetLowerLimit) = TRUE
4. Comprobación de los nuevos límites de contaje en las variables estáticas CurUpperLimit o CurLowerLimit

---

Nota

Si el nuevo límite de contaje superior es menor que el valor de contaje actual, el valor de contaje se establece en el límite de contaje inferior o en el valor de arranque en función de la parametrización. Si el nuevo límite de contaje inferior es mayor que el valor de contaje actual, el valor de contaje se establece en el límite de contaje superior o en el valor de arranque en función de la parametrización.

Valores de comparación: mediante el programa del usuario se modifican los dos valores de comparación de la manera siguiente:

1. Comprobación de la variable estática SetReferenceValue0 (o ReferenceValue1) = FALSE
2. Modificación de la variable estática NewReferenceValue0 (o NewReferenceValue1)
3. SetReferenceValue0 (o SetReferenceValue1) = TRUE
4. Comprobación del nuevo valor de comparación en la variable estática CurReferenceValue0 (o CurReferenceValue1)

Valor de contaje: Puede establecer el valor de contaje actual del programa de usuario a un nuevo valor de contaje. El valor de contaje actual se modifica de la manera siguiente:

1. Comprobación del parámetro de entrada SetCountValue = FALSE
2. Modificación de la variable estática NewCountValue
3. SetCountValue = TRUE
4. Comprobación del nuevo valor de contaje en el parámetro de salida CountValue

Valor de arranque: mediante el programa del usuario se modifica el valor de arranque de la manera siguiente:

1. Comprobación de la variable estática SetStartValue = FALSE
2. Modificación de la variable estática NewStartValue
3. SetStartValue = TRUE
4. Comprobación del nuevo valor de arranque en la variable estática CurStartValue

Acuse de eventos

El acuse de eventos notificados se realiza mediante el flanco ascendente del parámetro de entrada EventAck . EventAck debe permanecer activado hasta que el objeto tecnológico haya desactivado conjuntamente los bits de estado de los siguientes eventos del canal de contaje:

- CompResult0
- CompResult1
- ZeroStatus
- PosOverflow
- NegOverflow

Estado de las entradas digitales

El estado de las entradas digitales se obtiene a través de las variables estáticas StatusDI0, StatusDI1 o StatusDI2.

Utilizar salidas digitales mediante el programa del usuario

Mediante la instrucción High_Speed_Counter pueden ajustarse las salidas digitales

- si se ha parametrizado para "Activar salida" la opción "Utilizada por el programa de usuario".
- si se ha parametrizado para "Activar salida" la opción "Tras comando Set de la CPU hasta valor de comparación".
- si se ajusta la correspondiente variable estática ManualCtrlDQm (sobrescritura temporal).

Solo en estos casos serán efectivas las variables estáticas SetDQ0 y SetDQ1. En el primer y en el tercer caso, DQm sigue al valor de SetDQm. En el segundo caso, DQm se activa con un flanco (ascendente o descendente) de SetDQm. DQm se desactiva si el valor de contaje coincide con el valor de comparación.

Comportamiento en caso de error

Si se produce un error al llamar la instrucción o en el módulo tecnológico, se activa el parámetro de salida Error . La información de error adicional puede leerse en el parámetro de salida ErrorID (Página 5129).

Elimine la causa del error y acuse el mensaje de error activando el parámetro de entrada ErrorAck. Si ya no hay ningún error, el objeto tecnológico restablece el parámetro de salida Error . No se notifica ningún otro error hasta que se haya acusado el error anterior.

Modificar el sentido de contaje

El sentido de contaje solo puede modificarse desde el programa del usuario si se ha parametrizado como tipo de señal "Impulso (A)". En todos los demás casos, el sentido de contaje depende de las señales de entrada del módulo tecnológico. El sentido de contaje puede controlarse mediante la variable estática NewDirection:

- +1: sentido de contaje ascendente
- -1: sentido de contaje descendente

Para ejecutar la petición de modificación, es necesario ajustar la variable SetNewDirection = TRUE.

Parámetro de entrada High_Speed_Counter

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
SwGate	INPUT	BOOL	FALSE	Controlar puerta SW: • Flanco ascendente: se abre la puerta SW. • Flanco descendente: se cierra la puerta SW. SwGate habilita la puerta interna, en combinación con la puerta HW.
SetCountValue	INOUT	BOOL	FALSE	Un flanco ascendente inicia la transferencia del nuevo valor de contaje de la variable estática NewCountValue al módulo tecnológico. El valor de contaje se activa de inmediato tras la transferencia.
CaptureEnable	INPUT	BOOL	FALSE	Habilitar función de Capture Después de la habilitación, se produce un evento de Capture con el siguiente flanco parametrizado en la entrada digital correspondiente. Un flanco descendente de CaptureEnable restablece el parámetro de salida CaptureStatus. Un flanco descendente de CaptureEnable restablece asimismo la habilitación si no se produce un evento de Capture.
SyncEnable	INPUT	BOOL	FALSE	Habilitar sincronización El sentido habilitado para la sincronización se indica en las variables estáticas SyncUpDirection y SyncDownDirection. Un flanco descendente de SyncEnable restablece el parámetro de salida SyncStatus .
ErrorAck	INPUT	BOOL	FALSE	Un flanco ascendente acusa el estado de fallo notificado.
EventAck	INPUT	BOOL	FALSE	Un flanco ascendente restablece los siguientes parámetros de salida: • CompResult0 • CompResult1 • ZeroStatus • PosOverflow • NegOverflow

Parámetro de salida High_Speed_Counter

Parámetro	Declaración	Tipo de datos	Ajuste predetermin ado	Descripción
StatusHW	OUTPUT	BOOL	FALSE	Bit de estado módulo tecnológico: el módulo está parametrizado y listo para funcionar. Los datos del módulo son válidos.
StatusGate	OUTPUT	BOOL	FALSE	Bit de estado: la puerta interna está habilitada cuando el parámetro se activa.
StatusUp	OUTPUT	BOOL	FALSE	Bit de estado: el último impulso de contaje ha incrementado el contador y se encuentra como máximo 0,5 s atrás.
StatusDown	OUTPUT	BOOL	FALSE	Bit de estado: el último impulso de contaje ha provocado el decremento del contador y se encuentra como máximo 0,5 s atrás.
CompResult0	OUTPUT	BOOL	FALSE	Bit de estado: detectado evento de comparación para DQ0 Por medio del flanco ascendente del parámetro de entrada EventAck se desactiva CompResult0 .
CompResult1	OUTPUT	BOOL	FALSE	Bit de estado: detectado evento de comparación para DQ1 Por medio del flanco ascendente del parámetro de entrada EventAck se desactiva CompResult1 .
SyncStatus	OUTPUT	BOOL	FALSE	Bit de estado: se lleva a cabo la sincronización Si está ajustado el parámetro de entrada SyncEnable, el flanco parametrizado en la entrada digital correspondiente activa el bit de estado SyncStatus. SyncStatus se desactiva mediante flanco descendente en • SyncEnable (parámetro de entrada) o • SyncUpDirection (variable estática) o • SyncDownDirection (variable estática)
CaptureStatus	OUTPUT	BOOL	FALSE	Bit de estado: se ha producido un evento de Capture Si está ajustado el parámetro de entrada CaptureEnable, el flanco parametrizado en la entrada digital correspondiente activa el bit de estado CaptureStatus. El flanco descendente del parámetro de entrada CaptureEnable desactiva CaptureStatus .
ZeroStatus	OUTPUT	BOOL	FALSE	Bit de estado: CountValue ha alcanzado el valor "0" Por medio del flanco ascendente del parámetro de entrada EventAck se desactiva ZeroStatus.
PosOverflow	OUTPUT	BOOL	FALSE	Bit de estado: CountValue ha rebasado por exceso el límite de contaje superior. Por medio del flanco ascendente del parámetro de entrada EventAck se desactiva PosOverflow.
NegOverflow	OUTPUT	BOOL	FALSE	Bit de estado: CountValue ha rebasado por defecto el límite de contaje inferior. Por medio del flanco ascendente del parámetro de entrada EventAck se desactiva NegOverflow.
Error	OUTPUT	BOOL	FALSE	Se ha producido un error. Para conocer la causa del error, consulte el parámetro de salida ErrorID.

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
ErrorID	OUTPUT	WORD	0	El parámetro ErrorID (Página 5129) muestra el número del mensaje de error. ErrorID = 0000_H: No hay ningún error.
CountValue	OUTPUT	DINT	0	Valor actual de contaje
CapturedValue	OUTPUT	DINT	0	Último valor de Capture captado; CaptureStatus = TRUE si se ha producido un nuevo evento Capture.
MeasuredValue	OUTPUT	REAL	0.0	Valor medido actual de frecuencia, periodo o velocidad (dependiendo de la parametrización)

Parámetro ErrorID

Código de error (W#16#...)	Descripción
0000	No hay error
Mensajes de error del módulo tecnológico	
80A1	POWER_ERROR de la interfaz de realimentación: Tensión de alimentación L+ incorrecta
80A2	ENC_ERROR de la interfaz de realimentación: Señal de encóder incorrecta
80A3	LD_ERROR de la interfaz de realimentación: Error al cargar a través de interfaz de realimentación
Avisos de error de la instrucción High_Speed_Counter	
80B1	Sentido de contaje no válido
80B4	El nuevo límite de contaje inferior no cumple las siguientes condiciones: • Límite de contaje inferior < límite de contaje superior • Límite de contaje inferior <= valor de comparación/valor de contaje/valor de arranque
80B5	El nuevo límite de contaje superior no cumple las siguientes condiciones: • Límite de contaje inferior < límite de contaje superior • Límite de contaje superior <= valor de comparación/valor de contaje/valor de arranque
80B6	El nuevo valor de arranque no cumple la siguiente condición: • límite de contaje inferior <= valor de arranque <= límite de contaje superior
80B7	El nuevo valor de contaje no cumple la siguiente condición: • Límite de contaje inferior <= valor de contaje <= límite de contaje superior
80B8	El nuevo valor de comparación 0 no cumple la siguiente condición: • Límite de contaje inferior <= valor de comparación 0 <= límite de contaje superior • Valor de comparación 0 < valor de comparación 1
80B9	El nuevo valor de comparación 1 no cumple la siguiente condición: • Límite de contaje inferior <= valor de comparación 1 <= límite de contaje superior • Valor de comparación 0 < valor de comparación 1
80C0	La instrucción High_Speed_Counter se ha llamado varias veces con la misma instancia (DB).
80C1	La comunicación con el módulo tecnológico ha fallado (registros de lectura): coloca información de error de la instrucción interna RDREC en la variable estática AdditionalErrorID .

Código de error (W#16#...)	Descripción
80C2	La comunicación con el módulo tecnológico ha fallado (registros de escritura): coloca información de error de la instrucción interna WRREC en la variable estática AdditionalErrorID .
80C3	El acceso a los datos de entrada (interfaz de realimentación) ha fallado: coloca información de error de la instrucción interna GETIO_PART en la variable estática AdditionalErrorID.
80C4	El acceso a los datos de salida (interfaz de control) ha fallado: coloca información de error de la instrucción interna SETIO_PART en la variable estática AdditionalErrorID.
80C5	La lectura de la información de arranque actual del OB ha fallado: coloca información de error de la instrucción interna RD_SINFO en la variable estática AdditionalErrorID.
80C6	La determinación de las direcciones de E/S del módulo tecnológico ha fallado: coloca información de error de la instrucción interna RD_ADDR en la variable estática AdditionalErrorID.

Variables estáticas High_Speed_Counter

Variable		Tipo de datos	Ajuste predeterminado	Acceso	Descripción
NewCountValue		DINT	L#0	Escritura	Nuevo valor de contaje
NewReferenceValue0		DINT	L#0	Escritura	Nuevo valor de comparación 0
NewReferenceValue1		DINT	L#10	Escritura	Nuevo valor de comparación 1
NewUpperLimit		DINT	L#2147483647	Escritura	Nuevo límite de contaje superior
NewLowerLimit		DINT	L#-2147483648	Escritura	Nuevo límite de contaje inferior
NewStartValue		DINT	L#0	Escritura	Nuevo valor de arranque
CurReferenceValue0		DINT	L#0	Lectura	Valor de comparación 0 actual
CurReferenceValue1		DINT	L#10	Lectura	Valor de comparación 1 actual
CurUpperLimit		DINT	L#2147483647	Lectura	Límite de contaje superior actual
CurLowerLimit		DINT	L#-2147483648	Lectura	Límite de contaje inferior actual
CurStartValue		DINT	L#0	Lectura	Valor de arranque actual
NewDirection		INT	0	Escritura	Nuevo sentido de contaje: +1: sentido de contaje ascendente -1: sentido de contaje descendente
AdditionalErrorID		WORD	W#16#0000	Lectura	Información de error de una instrucción interna, p. ej. RDREC
UserCmdFlags		STRUCT	-		
	SetNewDirection	BOOL	FALSE	Escritura	Ajustar nuevo sentido de contaje
	SetUpperLimit	BOOL	FALSE	Escritura	Ajustar límite de contaje superior
	SetLowerLimit	BOOL	FALSE	Escritura	Ajustar límite de contaje inferior
	SetReferenceValue0	BOOL	FALSE	Escritura	Ajustar valor de comparación 0
	SetReferenceValue1	BOOL	FALSE	Escritura	Ajustar valor de comparación 1
	SetStartValue	BOOL	FALSE	Escritura	Ajustar valor de arranque
	SyncDownDirection	BOOL	TRUE	Escritura	Habilitar sincronización en sentido de contaje descendente
	SyncUpDirection	BOOL	TRUE	Escritura	Habilitar sincronización en sentido de contaje ascendente

Variable		Tipo de datos	Ajuste predeterminado	Acceso	Descripción
	SetDQ0	BOOL	FALSE	Escritura	Activar la salida digital DQ0
	SetDQ1	BOOL	FALSE	Escritura	Activar la salida digital DQ1
	ManualCtrlDQ0	BOOL	FALSE	Escritura	Habilitar ajuste de la salida digital DQ0: TRUE: • SetDQ0 activa DQ0 • Bit de control TM_CTRL_DQ0 = FALSE FALSE: • Ajuste no habilitado • Bit de control TM_CTRL_DQ0 = TRUE
	ManualCtrlDQ1	BOOL	FALSE	Escritura	Habilitar ajuste de la salida digital DQ1: TRUE: • SetDQ1 activa DQ1 • Bit de control TM_CTRL_DQ1 = FALSE FALSE: • Ajuste no habilitado • Bit de control TM_CTRL_DQ1 = TRUE
UserStatusFlags		STRUCT	-		
	StatusDI0	BOOL	FALSE	Lectura	Estado actual de entrada digital DI0
	StatusDI1	BOOL	FALSE	Lectura	Estado actual de entrada digital DI1
	StatusDI2	BOOL	FALSE	Lectura	Estado actual de entrada digital DI2
	StatusDQ0	BOOL	FALSE	Lectura	Estado actual de salida digital DQ0
	StatusDQ1	BOOL	FALSE	Lectura	Estado actual de salida digital DQ1

Control PID

PID_Compact

Novedades PID_Compact

PID_Compact V2.2

- **Uso con S7-1200**
  A partir de PID_Compact V2.2, la instrucción también se puede utilizar con funcionalidad V2 en una S7-1200 a partir de la versión de firmware 4.0.

PID_Compact V2.0

- **Comportamiento en caso de error**
  Se ha modificado a fondo el comportamiento en caso de error. PID_Compact tiene un comportamiento más tolerante a errores en el ajuste predeterminado. Este comportamiento se ajusta al copiar PID_Compact V1.X de una CPU S7-1200 a una CPU S7-1500.

ATENCIÓN
Su instalación puede sufrir daños. Si utiliza el ajuste predeterminado, PID_Compact se mantiene en modo automático al rebasar los límites del valor real. De este modo, su instalación puede sufrir daños. Configure un comportamiento en caso de error para su sistema regulado que proteja su instalación de daños.

  El parámetro Error indica si hay algún un error pendiente en ese momento. Si no persiste el error, se indica Error = FALSE. El parámetro ErrorBits indica qué errores han ocurrido. Con ErrorAck se acusan los errores y advertencias sin necesidad de reiniciar el regulador ni borrar la acción I. Al cambiar el modo de operación, ya no se borran los errores que no estén pendientes.
  El comportamiento en caso de error se configura con SetSubstituteOutput y ActivateRecoverMode.

- **Valor de salida sustitutivo**
  Puede configurarse un valor de salida sustitutivo que se emitirá en caso de error.

- **Cambio de modos de operación**
  El modo de operación se especifica en el parámetro de entrada/salida Mode y se activa mediante un flanco ascendente en ModeActivate. Se ha suprimido la variable sRet.i_Mode.

- **Capacidad multiinstancia**
  Puede llamar PID_Compact como DB multiinstancia. No se creará ningún objeto tecnológico y no dispondrá de interfaz de parametrización y puesta en servicio. Los PID_Compact se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

- **Comportamiento en arranque**
  Con RunModeByStartup = TRUE, el modo de operación especificado en Mode se inicia también en caso de flanco descendente en Reset y en caso de arranque en frío de la CPU.

- Comportamiento ENO
  ENO se ajusta en función del modo de operación.
  Si State = 0, entonces ENO = FALSE.
  Si State ≠ 0, entonces ENO = TRUE.
- Especificación de consigna durante la optimización
  La fluctuación admisible de la consigna durante la optimización se configura en la variable CancelTuningLevel.
- Rango de valores para límites de valor de salida
  Ahora ya no es necesario que el valor 0.0 se encuentre dentro de los límites de valor de salida.
- Preasignación de la acción I
  Con las variables IntegralResetMode y OverwriteInitialOutputValue se determina la preasignación de la acción I al cambiar del modo de operación "Inactivo" a "Modo automático".
- Control anticipativo
  El parámetro Disturbance permite activar el control anticipativo.
- Ajuste predeterminado de los parámetros PID
  Se han modificado los siguientes ajustes predeterminados:
  - Ponderación de la acción P (PWeighting), de 0.0 a 1.0
  - Ponderación de la acción D (DWeighting), de 0.0 a 1.0
  - Coeficiente para el retardo de la acción derivada (TdFiltRatio), de 0.0 a 0.2
- Cambio de nombre de las variables
  Se han cambiado los nombres de las variables estáticas para hacerlos más compatibles con PID_3Step.

PID_Compact V1.2

- Modo manual en el arranque de la CPU
  Si al arrancar la CPU ManualEnable = TRUE, PID_Compact arranca en modo manual. No es necesario un flanco ascendente en ManualEnable.
- Optimización inicial
  Si la CPU se apaga durante la optimización inicial, la optimización inicial arranca de nuevo al conectar la CPU.

PID_Compact V1.1

- Modo manual en el arranque de la CPU
  Al arrancar la CPU, PID_Compact únicamente pasa al modo manual si se detecta un flanco ascendente en ManualEnable. Si no se detecta un flanco ascendente, PID_Compact arranca en el último modo de operación en el que ManualEnable era FALSE.
- Comportamiento en Reset
  Un flanco ascendente en Reset hace que se desactiven los errores y las advertencias y que se borre la acción I. Si se detecta un flanco descendente en Reset, se cambia al último modo de operación activo.

- **Ajuste predeterminado del límite superior de valor real**
  El ajuste predeterminado r_Pv_Hlm se ha cambiado a 120.0.
- **Vigilancia del tiempo de muestreo**
  - No se emiten errores cuando el tiempo de muestreo actual >= 1,5 x valor medio actual o el tiempo de muestreo actual <= 0,5 x valor medio actual. En el modo automático se permite una desviación mayor del tiempo de muestreo.
  - PID_Compact es compatible con FW a partir de V2.0.
- **Acceso a variables**
  Las siguientes variables pueden utilizarse ahora en el programa de usuario.
  - i_Event_SUT
  - i_Event_TIR
  - r_Ctrl_Ioutv
- **Corrección de errores**
  PID_Compact aplica ahora impulsos correctos si el tiempo de conexión más pequeño es distinto del tiempo de desconexión más pequeño.

Compatibilidad con CPU y FW

La siguiente tabla muestra en qué CPU puede usarse cada versión de PID_Compact.

CPU	FW	PID_Compact
S7-1200	V4.x	V2.2 V1.2
S7-1200	V3.X	V1.2 V1.1
S7-1200	V2.X	V1.2 V1.1
S7-1200	V1.X	V1.0
S7-1500	V1.1	V2.2 V2.1 V2.0
S7-1500	V1.0	V2.0

PID_Compact V2

Descripción PID_Compact V2

Descripción

La instrucción PID_Compact ofrece un regulador PID con optimización integrada para actuadores de acción proporcional.

Se dispone de los modos de operación siguientes:

- Inactivo
- Optimización inicial
- Optimización fina
- Modo automático
- Modo manual
- Valor de salida sustitutivo con monitorización de errores

Los modos de operación se describen detalladamente en el parámetro State.

Algoritmo PID

PID_Compact es un regulador PIDT1 con Anti-Windup y ponderación de las acciones P y D. El algoritmo PID funciona de acuerdo con la siguiente fórmula:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
y	Valor de salida del algoritmo PID
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
T_D	Tiempo derivativo
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada $T1 = a \times T_D$)
c	Ponderación de la acción D

Diagrama de bloques PID_Compact

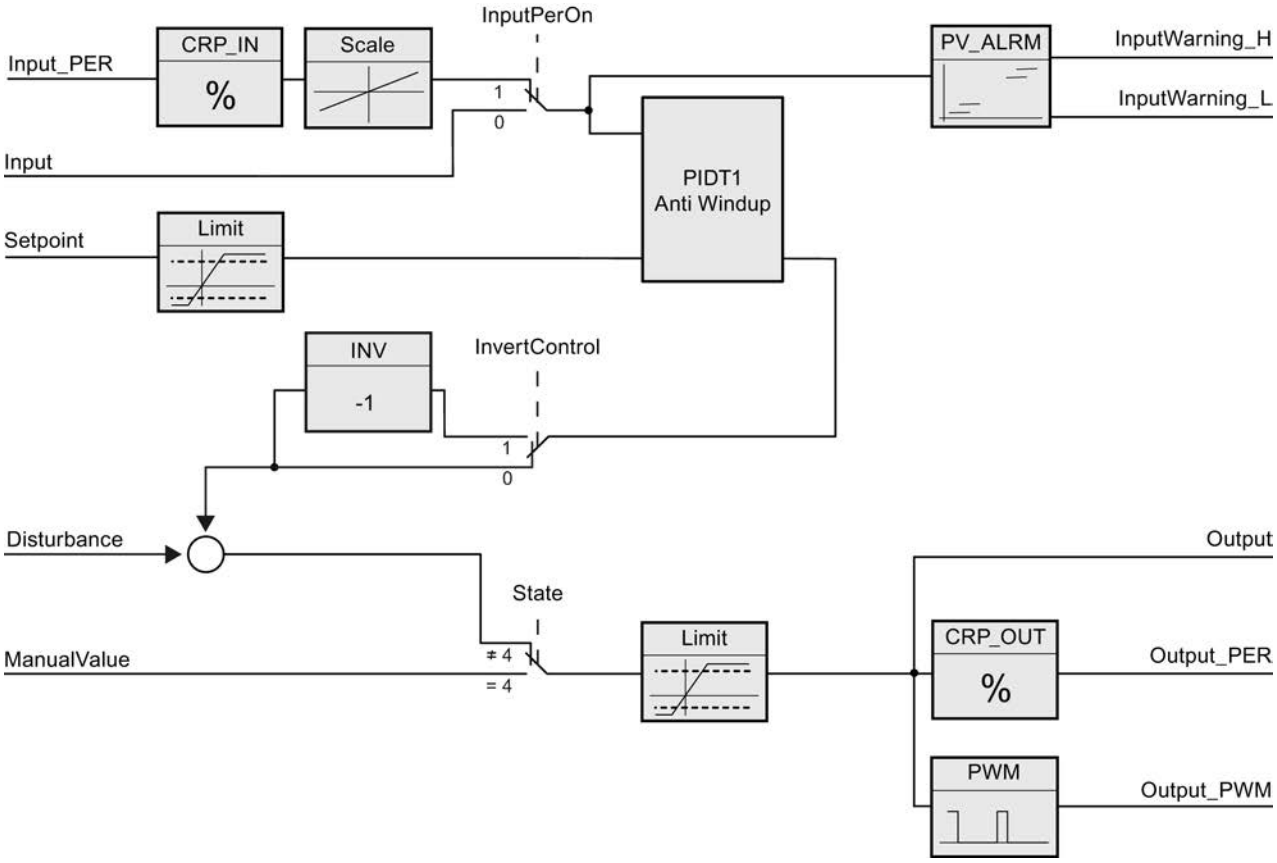
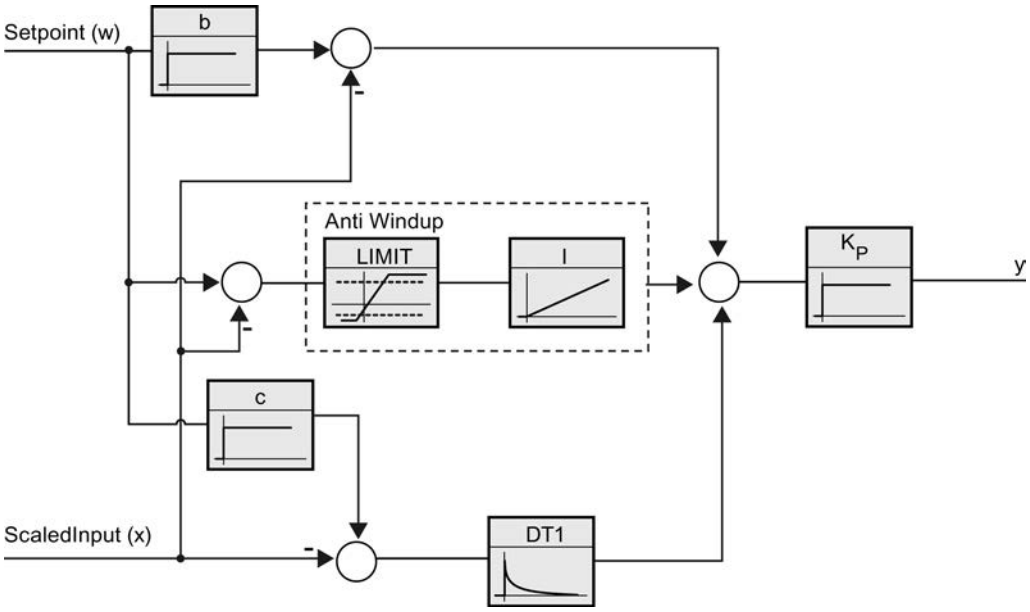


Diagrama de bloques PIDT1 con Anti-Windup



Llamada

PID_Compact se llama en una base de tiempo constante de un OB de alarma cíclica.

Si efectúa una llamada de PID_Compact como DB multiinstancia, no se creará ningún objeto tecnológico. No dispondrá de interfaz de parametrización y puesta en servicio. Los PID_Compact se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Cargar en dispositivo

Los valores actuales de variables remanentes solo se actualizan si se carga PID_Compact por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Arranque

Al arrancar la CPU, PID_Compact se inicia en el modo guardado en el parámetro de entrada/salida Mode. Para cambiar al modo de operación "Inactivo" durante el arranque, ajuste RunModeByStartup = FALSE.

Comportamiento en caso de error

En modo automático y durante la puesta en servicio, el comportamiento en caso de error depende de las variables SetSubstituteOutput y ActivateRecoverMode. En modo manual, el comportamiento no depende de SetSubstituteOutput y ActivateRecoverMode. Si ActivateRecoverMode = TRUE, el comportamiento dependerá además del error que se haya producido.

SetSubstituteOutput	ActivateRecoverMode	Ajuste del Editor de configuración > Valor de salida > Output a	Comportamiento
Irrelevante	FALSE	Cero (inactivo)	Cambio al modo de operación "Inactivo" (State = 0) El valor 0.0.0 se transfiere al elemento final de control.
FALSE	TRUE	Valor de salida actual por la duración del error	Cambio al modo de operación "Valor de salida sustitutivo con monitorización de errores" (State = 5) El valor de salida actual se transfiere al elemento final de control o actuador mientras dure el error.
TRUE	TRUE	Valor de salida sustitutivo mientras dure el error	Cambio al modo de operación "Valor de salida sustitutivo con monitorización de errores" (State = 5) El valor de SubstituteOutput se transfiere al actuador mientras dure el error.

PID_Compact utiliza ManualValue como valor de salida en el modo manual, excepto si ManualValue no es válido. Si ManualValue no es válido, se utiliza SubstituteOutput. Si ManualValue y SubstituteOutput no son válidos, se utiliza Config.OutputLowerLimit.

El parámetro Error indica si hay algún un error pendiente en ese momento. Si no persiste el error, se indica Error = FALSE. El parámetro ErrorBits muestra qué errores se han producido. ErrorBits se reinicia mediante un flanco ascendente en Reset o ErrorAck.

Funcionamiento PID_Compact V2

Monitorizar límites del valor real

En las variables Config.InputUpperLimit y Config.InputLowerLimit es posible definir un límite superior y uno inferior del valor real. Si el valor real se encuentra fuera de estos límites, ocurre un error (ErrorBits = 0001h).

En las variables Config.InputUpperWarning y Config.InputLowerWarning se definen un límite de advertencia superior y uno inferior del valor real. Si el valor real se encuentra fuera de estos límites de advertencia, se produce una advertencia (Warning = 0040h) y el parámetro de salida InputWarning_H o InputWarning_L adopta el valor TRUE.

Limitar consigna

En las variables Config.SetpointUpperLimit y Config.SetpointLowerLimit es posible definir un límite superior y uno inferior para la consigna. PID_Compact limita la consigna automáticamente a los límites del valor real. Es posible limitar la consigna a un rango más pequeño. PID_Compact verifica si dicho rango está dentro de los límites del valor real. Si la consigna está fuera de estos límites, se utilizan los límites superior e inferior como consigna y el parámetro de salida SetpointLimit_H o SetpointLimit_L adopta el valor TRUE.

La consigna se limita en todos los modos de operación.

Limitar el valor de salida

En las variables Config.OutputUpperLimit y Config.OutputLowerLimit se define un límite superior y uno inferior del valor de salida. Output, ManualValue y SubstituteOutput se limitan a estos valores. Los límites del valor de salida deben ser compatibles con el sentido de regulación.

Los valores válidos para los límites del valor de salida dependen del Output utilizado.

Output	de -100.0 a 100.0%
Output_PER	de -100.0 a 100.0%
Output_PWM	de -0.0 a 100.0%

Rige:

OutputUpperLimit > OutputLowerLimit

Valor de salida sustitutivo

PID_Compact puede emitir un valor de salida sustitutivo en caso de error que predefine en la variable SubstituteOutput. El valor de salida sustitutivo debe encontrarse dentro de los límites del valor de salida.

Vigilar la validez de las señales

Se vigila la validez de los valores de los siguientes parámetros al utilizarlos:

- Setpoint
- Input
- Input_PER
- Disturbance
- ManualValue
- SubstituteOutput
- Output
- Output_PER
- Output_PWM

Vigilancia del tiempo de muestreo PID_Compact

En el caso ideal, el tiempo de muestreo equivale al tiempo de ciclo del OB invocante. La instrucción PID_Compact mide en cada caso el tiempo que transcurre entre dos llamadas. Este es el tiempo de muestreo actual. Con cada cambio de modo de operación y en el primer arranque se calcula la media de los 10 primeros tiempos de muestreo. Si el tiempo de muestreo actual se desvía mucho de este valor medio, ocurre un error (Error = 0800h).

Durante la optimización, el error se produce si:

- valor medio nuevo >= 1,1 veces el valor medio antiguo
- valor medio nuevo <= 0,9 veces el valor medio antiguo

En el modo automático, el error se produce si:

- Valor medio nuevo >= 1,5 veces el valor medio antiguo
- Valor medio nuevo <= 0,5 veces el valor medio antiguo

Si se desactiva la monitorización del tiempo de muestreo (CycleTime.EnMonitoring = FALSE), puede llamarse PID_Compact también en el OB1. En tal caso, deberá aceptarse una regulación de menor calidad, debido a la fluctuación del tiempo de muestreo.

Tiempo de muestreo del algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de ciclo. Todas las demás funciones de PID_Compact se ejecutan con cada llamada.

Si utiliza Output_PWM, la precisión de la señal de salida se determina mediante el comportamiento del tiempo de muestreo del algoritmo PID al tiempo de ciclo del OB. El tiempo de ciclo debería ser de al menos 10 veces el tiempo de muestreo del algoritmo PID.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal. Para refrigeración o para las regulaciones de desagües puede ser necesario invertir el sentido de regulación. PID_Compact no funciona con ganancia proporcional negativa. Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida. El sentido de regulación se tiene en cuenta también durante la optimización inicial y la optimización fina.

Parámetros de entrada PID_Compact V2

Tabla 9-81

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
Setpoint	REAL	0.0	Consigna del regulador PID en modo automático
Input	REAL	0.0	Una variable del programa de usuario se utiliza como origen del valor real. Si utiliza el parámetro Input, debe cumplirse lo siguiente: Config.InputPerOn = FALSE.
Input_PER	INT	0	Una entrada analógica se utiliza como origen del valor real. Si utiliza el parámetro Input_PER, debe cumplirse lo siguiente: Config.InputPerOn = TRUE.
Disturbance	REAL	0.0	Magnitud perturbadora o control anticipativo
ManualEnable	BOOL	FALSE	• El flanco FALSE -> TRUE activa el modo de operación "Modo manual", State = 4, Mode no cambia. Mientras se cumple ManualEnable = TRUE, no es posible cambiar el modo de operación mediante un flanco ascendente en ModeActivate, ni utilizar el diálogo de puesta en servicio. • El flanco TRUE -> FALSE activa el modo de operación especificado en Mode. Se recomienda cambiar el modo de operación solo mediante ModeActivate.
ManualValue	REAL	0.0	Valor manual Este valor se utiliza en modo manual como valor de salida. Se admiten valores entre Config.OutputLowerLimit y Config.OutputUpperLimit
ErrorAck	BOOL	FALSE	• Flanco FALSE -> TRUE ErrorBits y Warning se desactivan.

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
Reset	BOOL	FALSE	Realiza un rearranque completo del regulador. • Flanco FALSE -> TRUE – Cambio al modo de operación "Inactivo" – ErrorBits y Warnings se desactivan. – Se borra la acción I (los parámetros PID se mantienen) • Mientras se cumple Reset = TRUE, PID_Compact permanece en el modo "Inactivo" (State = 0). • Flanco TRUE -> FALSE PID_Compact cambia al modo guardado en Mode.
ModeActivate	BOOL	FALSE	• Flanco FALSE -> TRUE PID_Compact cambia al modo guardado en Mode.

Parámetros de salida PID_Compact V2

Tabla 9-82

Parameter	Tipo de datos	Ajuste predeterminado	Descripción
ScaledInput	REAL	0.0	Valor real escalado
Las salidas "Output", "Output_PER" y "Output_PWM" pueden utilizarse paralelamente.			
Output	REAL	0.0	Valor de salida en el formato REAL
Output_PER	INT	0	Valor de salida analógico
Output_PWM	BOOL	FALSE	Valor de salida modulado por ancho de impulso El valor de salida se obtiene mediante tiempos de conexión y desconexión variables.
SetpointLimit_H	BOOL	FALSE	Si SetpointLimit_H = TRUE se cumple, significa que se ha alcanzado el límite superior absoluto de la consigna (Setpoint ≥ Config.SetpointUpperLimit). La consigna está limitada a Config.SetpointUpperLimit .
SetpointLimit_L	BOOL	FALSE	Si SetpointLimit_L = TRUE se cumple, significa que se ha alcanzado el límite inferior absoluto de la consigna (Setpoint ≤ Config.SetpointLowerLimit). La consigna está limitada a Config.SetpointLowerLimit .
InputWarning_H	BOOL	FALSE	Cuando InputWarning_H = TRUE, significa que se ha alcanzado o rebasado el límite superior de advertencia del valor real.
InputWarning_L	BOOL	FALSE	Cuando InputWarning_L = TRUE, significa que se ha alcanzado o rebasado por defecto el límite inferior de advertencia del valor real.

Parameter	Tipo de datos	Ajuste predeterminado	Descripción
State	INT	0	El parámetro State (Página 5152) indica el modo de operación actual del regulador PID. El modo de operación se cambia con el parámetro de entrada Mode y un flanco ascendente en ModeActivate. • State = 0: Inactivo • State = 1: optimización inicial • State = 2: Optimización fina • State = 3: Modo automático • State = 4: Modo manual • State = 5: Valor de salida sustitutivo con monitorización de errores
Error	BOOL	FALSE	Si Error = TRUE, existe al menos un mensaje de error en el ciclo actual.
ErrorBits	DWORD	DW#16#0	El parámetro ErrorBits (Página 5156) muestra qué mensajes de error existen. ErrorBits es remanente y se restablece con un flanco ascendente en Reset o ErrorAck.

Parámetros de entrada/salida PID_Compact V2

Tabla 9-83

Parameter	Tipo de datos	Ajuste predeterminado	Descripción
Mode	INT	4	Especifique en Mode el modo de operación al que debe cambiar PID_Compact. Posibilidades: • Mode = 0: Inactivo • Mode = 1: Optimización inicial • Mode = 2: Optimización fina • Mode = 3: Modo automático • Mode = 4: Modo manual El modo se activa mediante: • Flanco ascendente en ModeActivate • Flanco descendente en Reset • Flanco descendente en ManualEnable • Arranque en frío de la CPU si RunModeByStartup = TRUE Mode es remanente. Encontrará una descripción detallada de los modos en Parámetros State y Mode V2 (Página 5152).

Consulte también

Parámetros State y Mode V2 (Página 5152)

Variables estáticas de PID_Compact V2

No se deben modificar las variables que no aparecen listadas. Estas solo se utilizan internamente.

Tabla 9-84

Variable	Tipo de datos	Ajuste predeterminado	Descripción
IntegralResetMode	INT	1	La variable IntegralResetMode determina cómo se ajusta de forma predeterminada PIDCtrl.IntegralSum al cambiar del modo de operación "Inactivo" a "Modo automático". Este ajuste surte efecto solo para un ciclo. Posibilidades: • IntegralResetMode = 0: filtrar El valor de IntegralSum se ajusta de forma predeterminada de tal modo que la conmutación se realice sin discontinuidad. • IntegralResetMode = 1: borrar El valor de IntegralSum se borra. Si existe un error de regulación, se produce un salto del valor de salida. • IntegralResetMode = 2: parar El valor de IntegralSum no se modifica. Mediante el programa de usuario puede especificar un nuevo valor. • IntegralResetMode = 3: preasignar El valor de IntegralSum se preasigna automáticamente de modo que Output se calcula según el valor de OverwriteInitialOutputValue. Este ajuste es razonable, por ejemplo, para un regulador de relevo.
OverwriteInitialOutputValue	REAL	0.0	Si IntegralResetMode = 3, el valor de IntegralSum se preasigna automáticamente de modo que en el siguiente ciclo Output = OverwriteInitialOutputValue.
RunModeByStartup	BOOL	TRUE	Activar Mode después del rearranque de la CPU Si RunModeByStartup = TRUE, PID_Compact se inicia después del arranque de la CPU en el modo guardado en Mode. Si RunModeByStartup = FALSE, PID_Compact permanece en modo "Inactivo" después del arranque de la CPU.
LoadBackUp	BOOL	FALSE	Si LoadBackUp = TRUE, se carga nuevamente el último juego de parámetros PID. El juego se guardó antes de la última optimización. LoadBackUp se ajusta automáticamente de nuevo a FALSE.
PhysicalUnit	INT	0	Unidad física del valor real y de la consigna, p. ej. ºC o ºF.
PhysicalQuantity	INT	0	Unidad física del valor real y de la consigna, p. ej. temperatura.
ActivateRecoverMode	BOOL	TRUE	La Variable ActivateRecoverMode V2 (Página 5158) determina el comportamiento en caso de error.

Variable	Tipo de datos	Ajuste predeterminado	Descripción
Warning	DWORD	0	Variable Warning V2 (Página 5160) muestra las advertencias desde Reset = TRUE o ErrorAck =TRUE. Warning es remanente.
Progress	REAL	0.0	Progreso de la optimización en porcentaje (0.0 - 100.0)
CurrentSetpoint	REAL	0.0	CurrentSetpoint muestra siempre la consigna actual. Este valor se congela durante la optimización.
CancelTuningLevel	REAL	10.0	Fluctuación admisible de la consigna durante la optimización. La optimización no se cancela hasta que no se cumple: • Setpoint > CurrentSetpoint + CancelTuningLevel o bien, • Setpoint < CurrentSetpoint - CancelTuningLevel
SubstituteOutput	REAL	0.0	Valor de salida sustitutivo Si se cumplen las siguientes condiciones, se utiliza el valor de salida sustitutivo: • En el modo automático se ha producido un error. • SetSubstituteOutput = TRUE • ActivateRecoverMode = TRUE
SetSubstituteOutput	BOOL	TRUE	Si SetSubstituteOutput = TRUE y ActivateRecoverMode = TRUE, se emite el valor de salida sustitutivo configurado mientras haya un error pendiente. Si SetSubstituteOutput = FALSE y ActivateRecoverMode = TRUE, el actuador se mantiene en el valor de salida actual mientras haya un error pendiente. Si ActivateRecoverMode = FALSE, SetSubstituteOutput queda sin efecto. Si SubstituteOutput no es válido (ErrorBits = 20000h), el valor de salida sustitutivo no se puede emitir.
Config.InputPerOn	BOOL	TRUE	Si InputPerOn = TRUE, se utiliza el parámetro Input_PER. Si InputPerOn = FALSE, se utiliza el parámetro Input.
Config.InvertControl	BOOL	FALSE	Inversión del sentido de regulación Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida.

Variable	Tipo de datos	Ajuste predeterminado	Descripción
Config.InputUpperLimit	REAL	120.0	Límite superior del valor real Se vigila que Input y Input_PER respeten estos límites. En la entrada de periferia, el valor real puede encontrarse como máximo un 18% por encima del rango normalizado (margen de saturación). No se comunica ningún error más con el ajuste predeterminado debido al rebase por exceso del "Límite superior del valor real". Únicamente se detectan la rotura de hilo y el cortocircuito y PID_Compact se comporta según la reacción configurada en caso de error. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit	REAL	0.0	Límite inferior del valor real Se vigila que Input y Input_PER respeten estos límites. InputLowerLimit < InputUpperLimit
Config.InputUpperWarning	REAL	3.402822e+38	Límite superior de advertencia del valor real Si se configura InputUpperWarning a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se utiliza como límite superior de advertencia. Si se configura InputUpperWarning a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de advertencia. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning	REAL	-3.402822e+38	Límite inferior de advertencia del valor real Si se configura InputLowerWarning a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se utiliza como límite inferior de advertencia. Si se configura InputLowerWarning a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de advertencia. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit
Config.OutputUpperLimit	REAL	100.0	Límite superior del valor de salida Para más detalles ver OutputLowerLimit OutputUpperLimit > OutputLowerLimit

Variable	Tipo de datos	Ajuste predeterminado	Descripción
Config.OutputLowerLimit	REAL	0.0	Límite inferior del valor de salida Para Output o Output_PER, el rango de valores válido va de -100.0 hasta +100.0 incluido el cero. Para -100.0, Output_PER = -27648; para +100.0, Output_PER = 27648. Para Output_PWM el rango de valores válido va de 0.0 a +100.0. Los límites del valor de salida deben ser compatibles con el sentido de regulación. OutputLowerLimit < OutputUpperLimit
Config.SetpointUpperLimit	REAL	3.402822e+38	Límite superior de la consigna Si se configura SetpointUpperLimit a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se utiliza como límite superior de la consigna. Si se configura SetpointUpperLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de la consigna.
Config.SetpointLowerLimit	REAL	-3.402822e+38	Límite inferior de la consigna Si se configura SetpointLowerLimit a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se utiliza como límite inferior de la consigna. Si se configura SetpointLowerLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de la consigna.
Config.MinimumOnTime	REAL	0.0	El tiempo mínimo de conexión de la modulación de ancho de impulsos en segundos se redondea a MinimumOnTime = n×CycleTime.Value
Config.MinimumOffTime	REAL	0.0	El tiempo mínimo de desconexión de la modulación de ancho de impulsos en segundos se redondea a MinimumOffTime = n×CycleTime.Value
Config.InputScaling.UpperPointIn	REAL	27648.0	Escalado Input_PER arriba Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn, Input_PER se convierte a porcentaje.
Config.InputScaling.LowerPointIn	REAL	0.0	Escalado Input_PER abajo Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn, Input_PER se convierte a porcentaje.
Config.InputScaling.UpperPointOut	REAL	100.0	Valor real superior escalado Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn, Input_PER se convierte a porcentaje.

Variable	Tipo de datos	Ajuste predeterminado	Descripción
Config.InputScaling.LowerPointOut	REAL	0.0	Valor real inferior escalado Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn, Input_PER se convierte a porcentaje.
CycleTime.StartEstimation	BOOL	TRUE	Si CycleTime.StartEstimation = TRUE, se empieza a medir automáticamente el tiempo de ciclo. Una vez finalizada la medición, CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Si CycleTime.EnEstimation = TRUE, se calcula el tiempo de muestreo PID_Compact. Si CycleTime.EnEstimation = FALSE, el tiempo de muestreo PID_Compact no se calcula y debe configurar CycleTime.Value correctamente a mano.
CycleTime.EnMonitoring	BOOL	TRUE	Si CycleTime.EnMonitoring = FALSE no se vigila el tiempo de muestreo PID_Compact. Si no es posible ejecutar PID_Compact dentro del tiempo de muestreo, no se emite ningún error (ErrorBits=0800h) y PID_Compact no cambia al modo de operación "Inactivo".
CycleTime.Value	REAL	0.1	Tiempo de muestreo PID_Compact en segundos CycleTime.Value se determina automáticamente y equivale normalmente al tiempo de ciclo del OB invocante.
CtrlParamsBackUp.Gain	REAL	1.0	Ganancia proporcional almacenada Los valores de la estructura CtrlParamsBackUp pueden volverse a cargar con LoadBackUp = TRUE.
CtrlParamsBackUp.Ti	REAL	20.0	Tiempo de integración [s] almacenado
CtrlParamsBackUp.Td	REAL	0.0	Tiempo derivativo [s] almacenado
CtrlParamsBackUp.TdFiltRatio	REAL	0.2	Coeficiente almacenado para el retardo de la acción derivada
CtrlParamsBackUp.PWeighting	REAL	1.0	Factor de ponderación almacenado de la acción P
CtrlParamsBackUp.DWeighting	REAL	1.0	Factor de ponderación almacenado de la acción D
CtrlParamsBackUp.Cycle	REAL	1.0	Tiempo de muestreo almacenado del algoritmo PID
PIDSelfTune.SUT.CalculateParams	BOOL	FALSE	Las propiedades del sistema regulado se almacenan durante la optimización. Si SUT.CalculateParams = TRUE, se vuelven a calcular los parámetros de la optimización inicial con estas propiedades. De este modo puede cambiarse el método para calcular los parámetros sin que sea necesario repetir la optimización. SUT.CalculateParams se ajusta a FALSE después del cálculo.

Variable	Tipo de datos	Ajuste predeterminado	Descripción
PIDSelfTune.SUT.TuneRule	INT	0	Calcular los parámetros durante la optimización inicial según el método: • SUT.TuneRule = 0: PID según Chien, Hrones y Reswick • SUT.TuneRule = 1: PI según Chien, Hrones y Reswick
PIDSelfTune.SUT.State	INT	0	La variable SUT.State muestra la fase actual de la optimización inicial: • State = 0: Iniciar la optimización inicial • State = 100: Calcular desviación estándar • State = 200: Determinar punto de inflexión • State = 9900: Optimización inicial correcta • State = 1: Optimización inicial no correcta
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Con la variable RunIn puede determinar que se realice una optimización fina incluso sin optimización inicial. • RunIn = FALSE Si la optimización fina se inicia desde los modos de operación Inactivo o Manual, se inicia una optimización inicial. Si no se cumplen las condiciones para la optimización inicial, PID_Compact se comporta como si RunIn = TRUE. Si la optimización fina se inicia desde el modo automático, los parámetros PID existentes se regulan a la consigna. Solo entonces comienza la optimización fina. Si no es posible realizar una optimización inicial, PID_Compact pasa al modo desde el que se inició la optimización. • RunIn = TRUE La optimización inicial se omite. PID_Compact intenta alcanzar la consigna con el valor de salida mínimo o máximo, lo que puede ocasionar una sobreoscilación muy alta. La optimización fina se inicia entonces automáticamente. RunIn se ajusta a FALSE después de la optimización fina.
PIDSelfTune.TIR.CalculateParams	BOOL	FALSE	Las propiedades del sistema regulado se almacenan durante la optimización. Si TIR.CalculateParams = TRUE, se vuelven a calcular los parámetros de la optimización fina con estas propiedades. De este modo puede cambiarse el método para calcular los parámetros sin que sea necesario repetir la optimización. TIR.CalculateParams se ajusta a FALSE después del cálculo.

Variable	Tipo de datos	Ajuste predeterminado	Descripción
PIDSelfTune.TIR.TuneRule	INT	0	Calcular los parámetros durante la optimización fina según el método: • TIR.TuneRule = 0: PID automático • TIR.TuneRule = 1: PID rápido • TIR.TuneRule = 2: PID lento • TIR.TuneRule = 3: PID Ziegler-Nichols • TIR.TuneRule = 4: PI Ziegler-Nichols • TIR.TuneRule = 5: P Ziegler-Nichols
PIDSelfTune.TIR.State	INT	0	La variable TIR.State muestra la fase actual de la optimización fina: • State = -100 No es posible realizar la optimización fina. Primero se efectuará una optimización inicial. • State = 0: Iniciar la optimización fina • State = 200: Calcular desviación estándar • State = 300: Intentar alcanzar consigna • State = 400: Intentar alcanzar consigna con los parámetros PID actuales (si la optimización inicial ha sido correcta) • State = 500: Determinar oscilación y calcular parámetros • State = 9900: Optimización fina correcta • State = 1: Optimización fina no correcta
PIDCtrl.IntegralSum	REAL	0.0	Acción I actual
Retain.CtrlParams.Gain	REAL	1.0	Ganancia proporcional activa Para invertir el sentido de regulación utilice la variable Config.InvertControl. Los valores negativos de Gain invierten también el sentido de regulación. Se recomienda que ajuste el sentido de regulación únicamente a través de InvertControl. Si se cumplen InvertControl = TRUE y Gain < 0.0, el sentido de regulación también se invierte. Gain es remanente.
Retain.CtrlParams.Ti	REAL	20.0	• CtrlParams.Ti > 0.0: Tiempo de integración activo • CtrlParams.Ti = 0.0: La acción I está desactivada Ti es remanente.
Retain.CtrlParams.Td	REAL	0.0	• CtrlParams.Td > 0.0: Tiempo derivativo activo • CtrlParams.Td = 0.0: La acción D está desactivada Td es remanente.

Variable	Tipo de datos	Ajuste predeterminado	Descripción
Retain.CtrlParams.TdFiltRatio	REAL	0.2	Coeficiente activo para el retardo de la acción derivada El efecto de la acción D se retrasa mediante el coeficiente de retardo de la acción derivada. Retardo de la acción derivada = Tiempo derivativo x coeficiente de retardo de la acción derivada • 0.0: la acción D solo surte efecto para un ciclo y, por ello, casi no es efectiva. • 0.5: este valor se ha acreditado en la práctica para sistemas regulados con **una** constante de tiempo dominante. • > 1.0: cuanto mayor sea el coeficiente, más se retrasará el efecto de la acción D. TdFiltRatio es remanente.
Retain.CtrlParams.PWeighting	REAL	1.0	Ponderación activa de la acción P En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción P. Se recomiendan valores comprendidos entre 0.0 y 1.0. • 1.0: la acción P es totalmente efectiva cuando se cambia la consigna • 0.0: la acción P no actúa al cambiar la consigna Si se produce una modificación del valor real, la acción P es totalmente efectiva. PWeighting es remanente.
Retain.CtrlParams.DWeighting	REAL	1.0	Ponderación activa de la acción D En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción D. Se recomiendan valores comprendidos entre 0.0 y 1.0. • 1.0: la acción D es totalmente efectiva cuando se cambia la consigna. • 0.0: la acción D no actúa al cambiar la consigna Si se produce una modificación del valor real, la acción D es totalmente efectiva. DWeighting es remanente.
Retain.CtrlParams.Cycle	REAL	1.0	Tiempo de muestreo activo del algoritmo PID CtrlParams.Cycle se determina durante la optimización y se redondea a un múltiplo entero de CycleTime.Value. Cycle es remanente.

Nota

Modifique las variables que aparecen aquí en el modo de operación "Inactivo" para evitar un comportamiento erróneo del regulador PID.

Consulte también

Variable ActivateRecoverMode V2 (Página 5158)
Variable Warning V2 (Página 5160)
Cargar objetos tecnológicos en el dispositivo (Página 9940)

Modificaciones de la interfaz PID_Compact V2

La siguiente tabla muestra lo que ha cambiado en la interfaz de la instrucción PID_Compact.

PID_Compact V1	PID_Compact V2	Cambio
Input_PER	Input_PER	Tipo de datos de Word a Int
	Disturbance	Nuevo
	ErrorAck	Nuevo
	ModeActivate	Nuevo
Output_PER	Output_PER	Tipo de datos de Word a Int
Error	ErrorBits	Cambio de nombre
	Error	Nuevo
	Mode	Nuevo
sb_RunModeByStartup	RunModeByStartup	Función
	IntegralResetMode	
	OverwriteInitialOutputValue	Nuevo
	SetSubstituteOutput	Nuevo
	CancelTuningLevel	Nuevo
	SubstituteOutput	Nuevo

La siguiente tabla muestra las variables que han cambiado de nombre.

PID_Compact V1.x	PID_Compact V2
sb_GetCycleTime	CycleTime.StartEstimation
sb_EnCyclEstimation	CycleTime.EnEstimation
sb_EnCyclMonitoring	CycleTime.EnMonitoring
sb_RunModeByStartup	RunModeByStartup
si_Unit	PhysicalUnit
si_Type	PhysicalQuantity
sd_Warning	Warning
sBackUp.r_Gain	CtrlParamsBackUp.Gain
sBackUp.r_Ti	CtrlParamsBackUp.Ti
sBackUp.r_Td	CtrlParamsBackUp.Td
sBackUp.r_A	CtrlParamsBackUp.TdFiltRatio
sBackUp.r_B	CtrlParamsBackUp.PWeighting
sBackUp.r_C	CtrlParamsBackUp.DWeighting
sBackUp.r_Cycle	CtrlParamsBackUp.Cycle

PID_Compact V1.x	PID_Compact V2
sPid_Calc.r_Cycle	CycleTime.Value
sPid_Calc.b_RunIn	PIDSelfTune.TIR.RunIn
sPid_Calc.b_CalcParamSUT	PIDSelfTune.SUT.CalculateParams
sPid_Calc.b_CalcParamTIR	PIDSelfTune.TIR.CalculateParams
sPid_Calc.i_CtrlTypeSUT	PIDSelfTune.SUT.TuneRule
sPid_Calc.i_CtrlTypeTIR	PIDSelfTune.TIR.TuneRule
sPid_Calc.r_Progress	Progress
sPid_Cmpt.r_Sp_Hlm	Config.SetpointUpperLimit
sPid_Cmpt.r_Sp_Llm	Config.SetpointLowerLimit
sPid_Cmpt.r_Pv_Norm_IN_1	Config.InputScaling.LowerPointIn
sPid_Cmpt.r_Pv_Norm_IN_2	Config.InputScaling.UpperPointIn
sPid_Cmpt.r_Pv_Norm_OUT_1	Config.InputScaling.LowerPointOut
sPid_Cmpt.r_Pv_Norm_OUT_2	Config.InputScaling.UpperPointOut
sPid_Cmpt.r_Lmn_Hlm	Config.OutputUpperLimit
sPid_Cmpt.r_Lmn_Llm	Config.OutputLowerLimit
sPid_Cmpt.b_Input_PER_On	Config.InputPerOn
sPid_Cmpt.b_LoadBackUp	LoadBackUp
sPid_Cmpt.b_InvCtrl	Config.InvertControl
sPid_Cmpt.r_Lmn_Pwm_PPTm	Config.MinimumOnTime
sPid_Cmpt.r_Lmn_Pwm_PBTm	Config.MinimumOffTime
sPid_Cmpt.r_Pv_Hlm	Config.InputUpperLimit
sPid_Cmpt.r_Pv_Llm	Config.InputLowerLimit
sPid_Cmpt.r_Pv_HWrn	Config.InputUpperWarning
sPid_Cmpt.r_Pv_LWrn	Config.InputLowerWarning
sParamCalc.i_Event_SUT	PIDSelfTune.SUT.State
sParamCalc.i_Event_TIR	PIDSelfTune.TIR.State
sRet.i_Mode	sRet.i_Mode se ha suprimido. El modo de operación se cambia mediante Mode y ModeActivate.
sRet.r_Ctrl_Gain	Retain.CtrlParams.Gain
sRet.r_Ctrl_Ti	Retain.CtrlParams.Ti
sRet.r_Ctrl_Td	Retain.CtrlParams.Td
sRet.r_Ctrl_A	Retain.CtrlParams.TdFiltRatio
sRet.r_Ctrl_B	Retain.CtrlParams.PWeighting
sRet.r_Ctrl_C	Retain.CtrlParams.DWeighting
sRet.r_Ctrl_Cycle	Retain.CtrlParams.Cycle

Parámetros State y Mode V2

Interrelación de los parámetros

El parámetro State indica el modo de operación actual del regulador PID. El parámetro State no puede modificarse.

Con un flanco ascendente en ModeActivate, PID_Compact pasa al modo guardado en el parámetro de entrada/salida Mode.

Al conectar la CPU o cambiar de STOP a RUN, PID_Compact se inicia en el modo de operación guardado en Mode. Para dejar PID_Compact en el modo de operación "Inactivo", ajuste RunModeByStartup = FALSE.

Significado de los valores

State / Mode	Descripción del modo de operación
0	Inactivo En el modo de operación "Inactivo" se emite siempre el valor de salida 0.0, independientemente de Config.OutputUpperLimit y Config.OutputLowerLimit. La modulación del ancho de pulso está desactivada.
1	Optimización inicial La optimización inicial determina el comportamiento del proceso a un escalón del valor de salida y busca el punto de inflexión. Los parámetros PID óptimos se calculan a partir de la pendiente máxima y el tiempo muerto del sistema regulado. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina. Requisitos para la optimización inicial: • Modo de operación Inactivo (State = 0), Manual (State = 4) o Automático (State = 3) • ManualEnable = FALSE • Reset = FALSE • El valor real no debe aproximarse demasiado a la consigna. |Setpoint - Input| > 0.3 * | Config.InputUpperLimit - Config.InputLowerLimit| y |Setpoint - Input| > 0.5 * |Setpoint| • La consigna y el valor real se encuentran dentro de los límites configurados. Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. La consigna se congela en la variable CurrentSetpoint. La optimización no se cancela hasta que no se cumple: • Setpoint > CurrentSetpoint + CancelTuningLevel o bien, • Setpoint < CurrentSetpoint - CancelTuningLevel Antes de volver a calcular los parámetros PID, se guarda una copia de seguridad de estos y es posible reactivarlos con LoadBackUp. Una vez realizada correctamente la optimización inicial, se pasa al modo automático; si la optimización inicial no se ha realizado correctamente, el cambio del modo de operación depende de ActivateRecoverMode. Se muestra la fase de la optimización inicial con PIDSelfTune.SUT.State.

State / Mode	Descripción del modo de operación
2	Optimización fina La optimización fina genera una oscilación constante y limitada del valor real. A partir de la amplitud y frecuencia de esta oscilación se calculan de nuevo los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina. PID_Compact intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan solo una mínima influencia sobre la optimización fina. La consigna se congela en la variable CurrentSetpoint. La optimización no se cancela hasta que no se cumple: • Setpoint > CurrentSetpoint + CancelTuningLevel o bien, • Setpoint < CurrentSetpoint - CancelTuningLevel Antes de volver a calcular los parámetros PID, se guarda una copia de seguridad de estos y es posible reactivarlos con LoadBackUp. Requisitos para la optimización fina: • No se esperan perturbaciones. • La consigna y el valor real se encuentran dentro de los límites configurados. • ManualEnable = FALSE • Reset = FALSE • Modo de operación Automático (State = 3), Inactivo (State = 0) o Manual (State = 4) Al inicio, la optimización fina se desarrolla del modo siguiente: • Modo automático (State = 3) Si desea mejorar los parámetros PID existentes mediante optimización, inicie la optimización fina desde el modo automático. PID_Compact regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se cumplen las condiciones para la optimización fina. Sólo entonces comienza la optimización fina. • Modo Inactivo (State = 0) o Manual (State = 4) Si se cumplen las condiciones para optimización inicial, ésta se inicia. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Si el valor real para una optimización inicial se encuentra ya muy cerca de la consigna o si PIDSelfTune.TIR.RunIn = TRUE, se intenta alcanzar la consigna con el valor de salida máximo o mínimo. lo que puede ocasionar una sobreoscilación muy alta. Sólo entonces comienza la optimización fina. Una vez realizada correctamente la optimización fina, el regulador pasa al modo automático; si la optimización fina no se ha realizado correctamente, el cambio del modo de operación depende de ActivateRecoverMode. La fase de la optimización fina se muestra con PIDSelfTune.TIR.State.
3	Modo automático En el modo automático, PID_Compact regula el sistema regulado conforme a los parámetros predefinidos. Cuando se cumple uno de los requisitos siguientes, se cambia al modo automático: • Optimización inicial finalizada correctamente. • Optimización fina finalizada correctamente. • Cambio del parámetro de entrada/salida Mode al valor 3 y un flanco ascendente en ModeActivate. El cambio de modo automático a modo manual se realiza sin discontinuidad únicamente en el editor de puesta de servicio. En el modo automático se tiene en cuenta la variable ActivateRecoverMode.

State / Mode	Descripción del modo de operación
4	Modo manual En el modo manual se especifica un valor de salida manual en el parámetro ManualValue. Este modo puede activarse también mediante ManualEnable = TRUE. Se recomienda cambiar los modos de operación solo mediante Mode y ModeActivate. El cambio de modo manual a modo automático se efectúa sin discontinuidad. El modo manual es posible aunque haya un error pendiente.
5	Valor de salida sustitutivo con monitorización de errores El algoritmo de regulación está desactivado. La variable SetSubstituteOutput determina qué valor de salida se emite durante este modo de operación. • SetSubstituteOutput = FALSE: Último valor de salida válido • SetSubstituteOutput = TRUE: Valor de salida sustitutivo Este modo de operación no puede activarse con Mode = 5. Se activa en caso de error en lugar del modo de operación "Inactivo", cuando se cumplen todas las condiciones siguientes: • Modo automático (Mode = 3) • ActivateRecoverMode = TRUE • Se han producido uno o varios errores en los que interviene ActivateRecoverMode. En cuanto los errores dejan de estar activos, PID_Compact pasa de nuevo al modo automático.

Comportamiento ENO

Si State = 0, entonces ENO = FALSE.

Si State ≠ 0, entonces ENO = TRUE.

Cambio automático de modo de operación durante la puesta en servicio

Una vez realizada correctamente la optimización inicial o la optimización fina, se activa el modo automático. La siguiente tabla muestra cómo se modifican Mode y State durante una optimización inicial correcta.

N.º de ciclo	Mode	State	Acción
0	4	4	Ajustar Mode = 1
1	1	4	Ajustar ModeActivate = TRUE
1	4	1	El valor de State se guarda en Mode Se inicia la optimización inicial
n	4	1	Optimización inicial correcta
n	3	3	Se inicia el modo automático

En caso de error, PID_Compact cambia automáticamente el modo de operación. La siguiente tabla muestra cómo se modifican Mode y State durante una optimización inicial incorrecta.

N.º de ciclo	Mode	State	Acción
0	4	4	Ajustar Mode = 1
1	1	4	Ajustar ModeActivate = TRUE

N.º de ciclo	Mode	State	Acción
1	4	1	El valor de State se guarda en Mode Se inicia la optimización inicial
n	4	1	La optimización inicial se ha cancelado
n	4	4	Se inicia el modo manual

Si ActivateRecoverMode = TRUE, se activa el modo guardado en Mode. Al iniciar la optimización inicial o fina, PID_Compact ha guardado el valor de State en el parámetro de entrada/salida Mode. Por lo tanto, PID_Compact cambia al modo desde el que se inició la optimización.

Si ActivateRecoverMode = FALSE, se cambia al modo de operación "Inactivo".

Consulte también

Parámetros de salida PID_Compact V2 (Página 5141)

Parámetro ErrorBits V2

Si hay varios errores presentes a la vez, los valores del ErrorBits se muestran sumados binariamente. Si se muestra, p. ej., ErrorBits = 0003h, significa que están presentes simultáneamente los errores 0001h y 0002h.

PID_Compact utiliza en modo manual ManualValue como valor de salida. La excepción es Errorbits = 10000h.

ErrorBits (DW#16#...)	Descripción
0000	No hay ningún error.
0001	El parámetro "Input" se encuentra fuera de los límites del valor real. • Input > Config.InputUpperLimit o bien, • Input < Config.InputLowerLimit Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_Compact permanece en modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode.
0002	Valor no válido en el parámetro "Input_PER". Compruebe si hay un error en la entrada analógica. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_Compact emite el valor de salida sustitutivo configurado. En cuanto el error deja de estar pendiente, PID_Compact pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode.
0004	Error durante la optimización fina. No se ha podido mantener la oscilación del valor real. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact cancela la optimización y cambia al modo guardado en Mode.
0008	Error al iniciar la optimización inicial. El valor real se aproxima demasiado a la consigna. Inicie la optimización fina. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact cancela la optimización y cambia al modo guardado en Mode.

ErrorBits (DW#16#...)	Descripción
0010	Se ha cambiado la consigna durante la optimización. En la variable CancelTuningLevel puede ajustar la fluctuación admisible de la consigna. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact cancela la optimización y cambia al modo guardado en Mode.
0020	La optimización inicial no está permitida durante la optimización fina. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact permanece en el modo optimización fina.
0080	Error durante la optimización inicial. Los límites del valor de salida no están configurados correctamente. Compruebe si los límites del valor de salida están configurados correctamente y si encajan con el sentido de la regulación. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact cancela la optimización y cambia al modo guardado en Mode.
0100	Un error en la optimización fina ha provocado parámetros no válidos. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact cancela la optimización y cambia al modo guardado en Mode.
0200	Valor no válido en el parámetro "Input": El valor no tiene un formato numérico válido. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_Compact emite el valor de salida sustitutivo configurado. En cuanto el error deja de estar pendiente, PID_Compact pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode.
0400	Error al calcular el valor de salida. Compruebe los parámetros PID. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_Compact emite el valor de salida sustitutivo configurado. En cuanto el error deja de estar pendiente, PID_Compact pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode.
0800	Error de tiempo de muestreo: PID_Compact no se llama dentro del tiempo de muestreo del OB de alarma cíclica. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_Compact permanece en modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode.
1000	Valor no válido en el parámetro "Setpoint": El valor no tiene un formato numérico válido. Si antes de producirse el error estaba activo el modo automático y se cumplía ActivateRecoverMode = TRUE, PID_Compact emite el valor de salida sustitutivo configurado. En cuanto el error deja de estar pendiente, PID_Compact pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode.
10000	Valor no válido en el parámetro ManualValue. El valor no tiene un formato numérico válido. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_Compact utiliza SubstituteOutput como valor de salida. En cuanto se especifica un valor válido en ManualValue, PID_Compact lo utiliza como valor de salida.

ErrorBits (DW#16#...)	Descripción
20000	Valor no válido en la variable SubstituteOutput. El valor no tiene un formato numérico válido. PID_Compact utiliza como valor de salida el límite inferior del valor de salida. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_Compact pasa de nuevo al modo automático.
40000	Valor no válido en el parámetro Disturbance. El valor no tiene un formato numérico válido. Si antes de producirse el error estaba activo el modo automático y ActivateRecoverMode = TRUE, Disturbance se ajusta a cero. PID_Compact permanece en modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_Compact pasa al modo guardado en Mode. Si en la fase actual Disturbance no influye en el valor de salida, la optimización no se cancela.

Variable ActivateRecoverMode V2

La variable ActivateRecoverMode determina el comportamiento en caso de error. El parámetro Error indica si hay algún un error pendiente en ese momento. Si no persiste el error, se indica Error = FALSE. El parámetro ErrorBits indica qué errores han ocurrido.

Modo automático

ATENCIÓN
Su instalación puede sufrir daños. Si se cumple ActivateRecoverMode = TRUE, incluso al rebasar los límites del valor real PID_Compact permanece en el modo automático en caso de error. De este modo, su instalación puede sufrir daños. Configure un comportamiento en caso de error para su sistema regulado que proteja su instalación de daños.

ActivateRecover Mode	Descripción
FALSE	En caso de error, PID_Compact cambia al modo de operación "Inactivo". El regulador no se activa hasta que no se detecta un flanco descendente en Reset o un flanco ascendente en ModeActivate.
TRUE	Si en el modo automático se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación, ya que PID_Compact alterna entre el valor de salida calculado y el valor de salida sustitutivo cada vez que se produce un error. Compruebe en tal caso el parámetro ErrorBits y elimine la causa de error. Si se producen uno o varios de los errores siguientes, PID_Compact permanece en modo automático: • 0001h: El parámetro "Input" se encuentra fuera de los límites del valor real. • 0800h: Error en tiempo de muestreo • 40000h: Valor no válido en el parámetro Disturbance. Si se produce uno o varios de los errores siguientes, PID_Compact pasa al modo "Valor de salida sustitutivo con monitorización de errores": • 0002h: Valor no válido en el parámetro Input_PER. • 0200h: Valor no válido en el parámetro Input. • 0400h: Error al calcular el valor de salida. • 1000h: Valor no válido en el parámetro Setpoint. Si se produce el siguiente error, PID_Compact pasa al modo "Valor de salida sustitutivo con monitorización de errores" y desplaza el actuador hasta Config.OutputLowerLimit: • 20000h: Valor no válido en la variable SubstituteOutput. El valor no tiene un formato numérico válido. Este comportamiento no depende de SetSubstituteOutput. En cuanto los errores dejan de estar activos, PID_Compact pasa de nuevo al modo automático.

Optimización inicial y optimización fina

ActivateRecover Mode	Descripción
FALSE	En caso de error, PID_Compact cambia al modo de operación "Inactivo". El regulador no se activa hasta que no se detecta un flanco descendente en Reset o un flanco ascendente en ModeActivate.
TRUE	Si se produce el siguiente error, PID_Compact permanece en el modo activo: • 0020h: La optimización inicial no está permitida durante la optimización fina. Los siguientes errores se omiten: • 10000h: Valor no válido en el parámetro ManualValue. • 20000h: Valor no válido en la variable SubstituteOutput. Con todos los demás errores, PID_Compact interrumpe la optimización y cambia al modo desde el que se inició la optimización.

Modo manual

En el modo manual, ActivateRecoverMode no tiene efecto.

Variable Warning V2

Si hay varias advertencias presentes a la vez, los valores de la variable Warning se muestran sumados en modo binario. Si se muestra p. ej. la advertencia 0003h, significa que están presentes simultáneamente las advertencias 0001h y 0002h.

Warning (DW#16#....)	Descripción
0000	No hay ninguna advertencia.
0001	Durante la optimización inicial no se encontró el punto de inflexión.
0004	La consigna ha sido limitada a los límites ajustados.
0008	En el método de cálculo seleccionado no se han definido todas las propiedades necesarias del sistema regulado. En sustitución se calcularon los parámetros PID con el método TIR.TuneRule = 3.
0010	No se ha podido cambiar el modo de operación porque Reset = TRUE o ManualEnable = TRUE.
0020	El tiempo de muestreo del algoritmo PID está limitado por el tiempo de ciclo del OB invocante. Para obtener mejores resultados, utilice tiempos de ciclo más cortos para el OB.
0040	El valor real ha rebasado por exceso uno de sus límites de advertencia.
0080	Valor no válido en Mode. El modo de operación no se conmuta.
0100	El valor manual ha sido limitado a los límites de la salida del regulador.
0200	No se admiten las reglas indicadas para la optimización. No se calculan parámetros PID.
1000	El valor de salida sustitutivo no puede alcanzarse porque está fuera de los límites del valor de salida.

Las siguientes advertencias se eliminan en cuanto la causa se soluciona:

- 0001h
- 0004h
- 0008h
- 0040h
- 0100h

Todas las demás advertencias se eliminan si hay un flanco ascendente en Reset o ErrorAck.

PID_Compact V1

Descripción PID_Compact V1

Descripción

La instrucción PID_Compact ofrece un regulador PID con optimización integrada para los modos automático y manual.

Llamada

PID_Compact se llama en un periodo constante del tiempo de ciclo del OB invocante (preferentemente en un OB de alarma cíclica).

Cargar en dispositivo

Los valores actuales de variables remanentes solo se actualizan si se carga PID_Compact por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Arranque

PID_Compact arranca simultáneamente con la CPU en el último modo de operación activo. Para dejar PID_ Compact en el modo de operación "Inactivo", es necesario ajustar sb_RunModeByStartup = FALSE.

Vigilancia del tiempo de muestreo PID_Compact

En el caso ideal, el tiempo de muestreo equivale al tiempo de ciclo del OB invocante. La instrucción PID_Compact mide en cada caso el tiempo que transcurre entre dos llamadas. Este es el tiempo de muestreo actual. Con cada cambio de modo de operación y en el primer arranque se calcula la media de los 10 primeros tiempos de muestreo. Si el tiempo de muestreo actual difiere mucho de este valor medio, se produce un error (Error = 0800 hex) y PID_Compact cambia al modo de operación "Inactivo".

Durante la optimización, PID_Compact a partir de la versión 1.1 cambia al modo de operación "Inactivo" cuando se dan las siguientes condiciones:

- Valor medio nuevo >= 1,1 veces el valor medio antiguo
- Valor medio nuevo <= 0,9 veces el valor medio antiguo

PID_Compact a partir de la versión 1.1 cambia al modo de operación "Inactivo" cuando se dan las siguientes condiciones en el modo automático:

- Valor medio nuevo >= 1,5 veces el valor medio antiguo
- Valor medio nuevo <= 0,5 veces el valor medio antiguo

Durante la optimización y el modo automático PID_Compact 1.0 cambia al modo de operación "Inactivo" cuando se dan las siguientes condiciones:

- Valor medio nuevo >= 1,1 veces el valor medio antiguo
- Valor medio nuevo <= 0,9 veces el valor medio antiguo
- Tiempo de muestreo actual >= 1,5 veces el valor medio actual
- Tiempo de muestreo actual <= 0,5 veces el valor medio actual

Tiempo de muestreo del algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de ciclo. Todas las demás funciones de PID_Compact se ejecutan con cada llamada.

Algoritmo PID

PID_Compact es un regulador PIDT1 con Anti-Windup y ponderación de las acciones P y D. El valor de salida se calcula con la siguiente fórmula.

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
y	Valor de salida
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
a	Coeficiente para el retardo de la acción derivada (T1 = a x T_D)
	Tiempo derivativo
c	Ponderación de la acción D

Diagrama de bloques PID_Compact

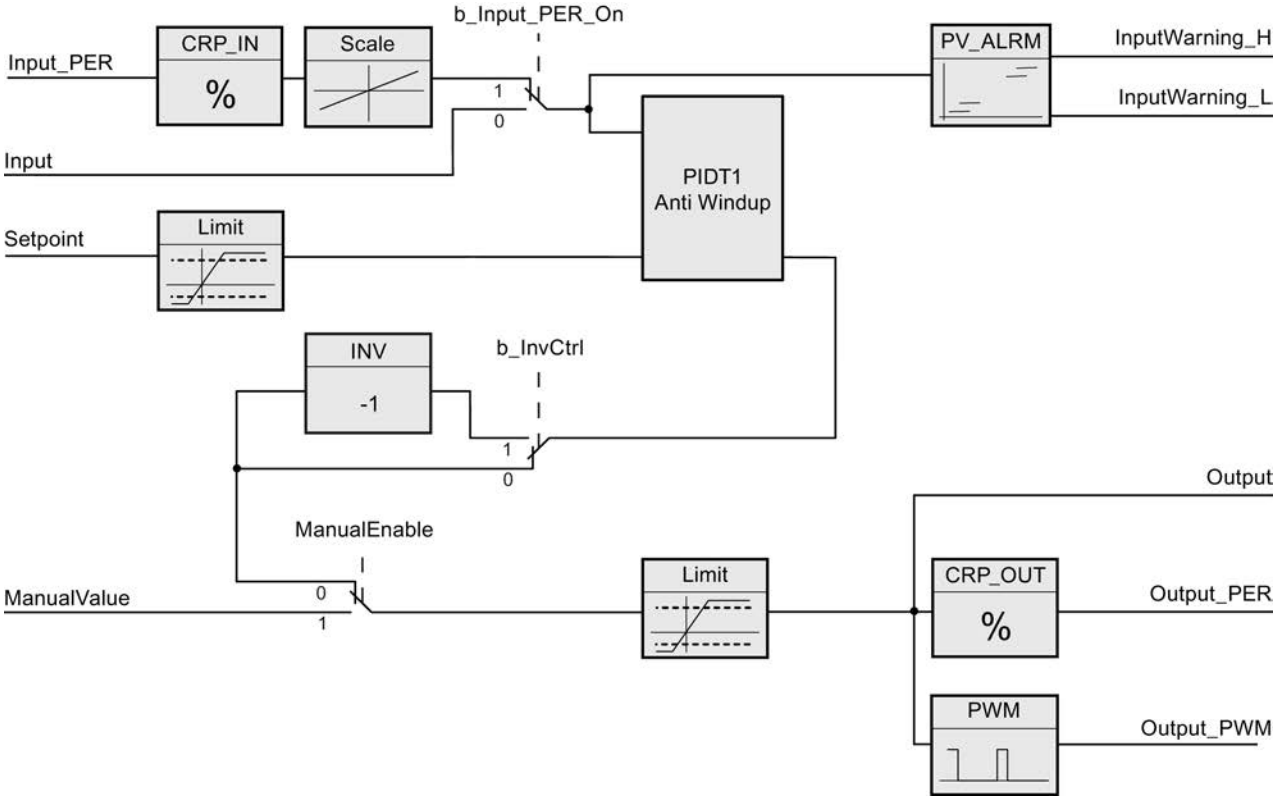
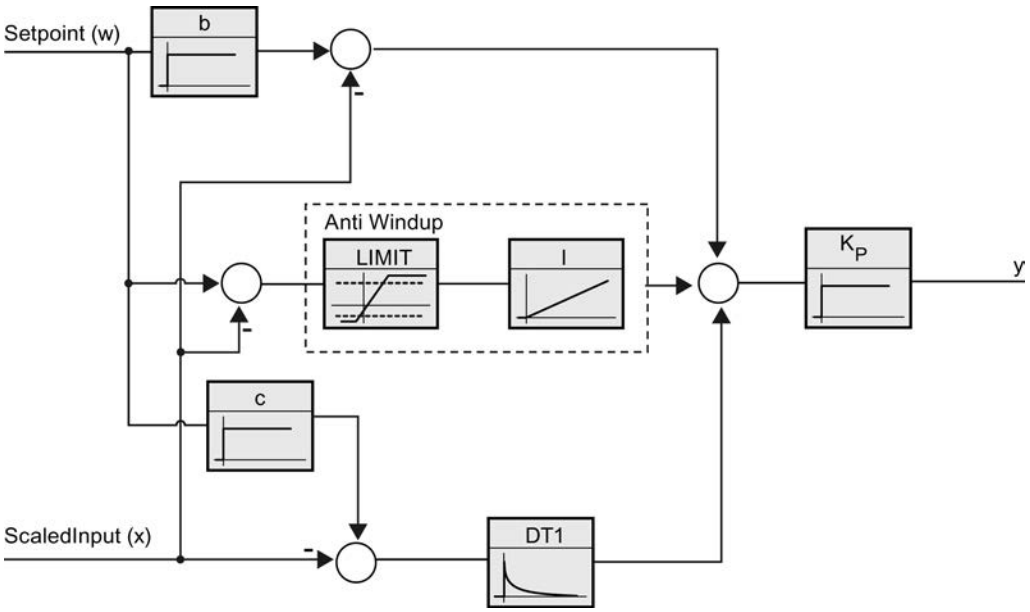


Diagrama de bloques PIDT1 con Anti-Windup



Comportamiento en caso de error

Cuando ocurren errores, estos se indican en el parámetro Error y PID_Compact cambia al modo de operación "Inactivo". Con el parámetro Reset se resetean los errores.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal. Para refrigeración o para las regulaciones de desagües puede ser necesario invertir el sentido de regulación. PID_Compact no funciona con ganancia proporcional negativa. Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida. El sentido de regulación se tiene en cuenta también durante la optimización inicial y la optimización fina.

Consulte también

Tipo de regulación (Página 9982)

Parámetros de entrada PID_Compact V1

Tabla 9-85

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
Setpoint	REAL	0.0	Consigna del regulador PID en modo automático
Input	REAL	0.0	Una variable del programa de usuario se utiliza como origen del valor real. Si se utiliza el parámetro Input, debe cumplirse lo siguiente: sPid_Cmpt.b_Input_PER_On = FALSE.
Input_PER	WORD	W#16#0	Entrada analógica como origen del valor real Si se utiliza el parámetro Input_PER, debe cumplirse lo siguiente: sPid_Cmpt.b_Input_PER_On = TRUE.
ManualEnable	BOOL	FALSE	• El flanco FALSE -> TRUE selecciona el modo de operación "Modo manual", State = 4, sRet.i_Mode no cambia. • El flanco TRUE -> FALSE selecciona el último modo de operación activo, State =sRet.i_Mode Mientras ManualEnable = TRUE un cambio de sRet.i_Mode no tendrá efecto. Sólo con el flanco TRUE -> FALSE en ManualEnable , se tiene en cuenta el cambio de sRet.i_Mode. **PID_Compact V1.2 und PID_Compact V1.0** Si al arrancar la CPU ManualEnable = TRUE, PID_Compact arranca en modo manual. No es necesario que se produzca un flanco ascendente (FALSE > TRUE) en ManualEnable. **PID_Compact V1.1** Al arrancar la CPU, PID_Compact únicamente pasa al modo manual si se detecta un flanco ascendente (FALSE->TRUE) en ManualEnable . Si no se detecta un flanco ascendente, PID_Compact arranca en el último modo de operación en el que ManualEnable era FALSE.
ManualValue	REAL	0.0	Valor manual Este valor se utiliza en el modo manual como valor de salida.
Reset	BOOL	FALSE	El parámetro Reset (Página 5174) realiza un rearranque del regulador.

Parámetros de salida PID_Compact V1

Tabla 9-86

Parameter	Tipo de datos	Ajuste predeterminado	Descripción
ScaledInput	REAL	0.0	Salida del valor real escalado
Las salidas "Output", "Output_PER" y "Output_PWM" pueden utilizarse paralelamente.			
Output	REAL	0.0	Valor de salida en el formato REAL
Output_PER	WORD	W#16#0	Valor de salida analógico
Output_PWM	BOOL	FALSE	Valor de salida modulado por ancho de impulso El valor de salida se obtiene mediante tiempos de conexión y desconexión variables.

Parameter	Tipo de datos	Ajuste predeterminado	Descripción
SetpointLimit_H	BOOL	FALSE	Cuando SetpointLimit_H = TRUE, significa que se ha alcanzado el límite superior absoluto de la consigna. En la CPU la consigna se limita al límite superior absoluto configurado para la consigna. Como límite superior de la consigna se ajusta de forma predeterminada el límite superior absoluto configurado para el valor real. Si se configura sPid_Cmpt.r_Sp_Hlm a un valor que se encuentre dentro de los límites del valor real, este valor se utilizará como límite superior de la consigna.
SetpointLimit_L	BOOL	FALSE	Cuando SetpointLimit_L = TRUE, significa que se ha alcanzado el límite inferior absoluto de la consigna. En la CPU la consigna se limita al límite inferior absoluto configurado para la consigna. Como límite inferior de la consigna se ajusta de forma predeterminada el límite inferior absoluto configurado para el valor real. Si se configura sPid_Cmpt.r_Sp_Llm a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de la consigna.
InputWarning_H	BOOL	FALSE	Cuando InputWarning_H = TRUE, significa que se ha alcanzado o rebasado el límite superior de advertencia del valor real.
InputWarning_L	BOOL	FALSE	Cuando InputWarning_L = TRUE, significa que se ha alcanzado o rebasado por defecto el límite inferior de advertencia del valor real.
State	INT	0	El parámetro State (Página 5171) indica el modo de operación actual del regulador PID. El modo de operación se cambia con la variable sRet.i_Mode. • State = 0: Inactivo • State = 1: optimización inicial • State = 2: Optimización fina • State = 3: Modo automático • State = 4: Modo manual
Error	DWORD	W#16#0	El parámetro Error (Página 5174) indica los mensajes de error. Error = 0000: No hay ningún error.

Variables estáticas PID_Compact V1

No se deben modificar las variables que no aparecen listadas. Estas solo se utilizan internamente.

Tabla 9-87

Variable	Tipo de datos	Ajuste predetermina do	Descripción
sb_GetCycleTime	BOOL	TRUE	Si sb_GetCycleTime = TRUE se empieza a medir automáticamente el tiempo de ciclo. Una vez finalizada la medición, CycleTime.StartEstimation = FALSE.
sb_EnCyclEstimation	BOOL	TRUE	Si sb_EnCyclEstimation = TRUE se calcula el tiempo de muestreo PID_Compact.

Variable	Tipo de datos	Ajuste predetermina do	Descripción
sb_EnCyclMonitoring	BOOL	TRUE	Si sb_EnCyclMonitoring = FALSE no se vigila el tiempo de muestreo PID_Compact. Si no es posible ejecutar PID_Compact dentro del tiempo de muestreo, no se genera el error 0800 y PID_Compact no cambia al modo de operación "Inactivo".
sb_RunModeByStartup	BOOL	TRUE	Activar último modo de operación después del rearranque completo de la CPU Si sb_RunModeByStartup = FALSE, el regulador permanece inactivo tras un arranque de la CPU. Si sb_RunModeByStartup = TRUE, el regulador regresa al último modo de operación activo tras el arranque de la CPU.
si_Unit	INT	0	Unidad física del valor real y de la consigna, p. ej. ºC o ºF.
si_Type	INT	0	Magnitud física del valor real y de la consigna, p. ej. temperatura
sd_Warning	DWORD	DW#16#0	La variable sd_warning (Página 5176) muestra las advertencias desde Reset o desde el último cambio del modo de operación.
sBackUp.r_Gain	REAL	1.0	Ganancia proporcional almacenada Los valores de la estructura sBackUp pueden volverse a cargar con sPid_Cmpt.b_LoadBackUp = TRUE.
sBackUp.r_Ti	REAL	20.0	Tiempo de integración [s] almacenado
sBackUp.r_Td	REAL	0.0	Tiempo derivativo [s] almacenado
sBackUp.r_A	REAL	0.0	Coeficiente almacenado para el retardo de la acción derivada
sBackUp.r_B	REAL	0.0	Factor de ponderación almacenado de la acción P
sBackUp.r_C	REAL	0.0	Factor de ponderación almacenado de la acción D
sBackUp.r_Cycle	REAL	1.0	Tiempo de muestreo almacenado del algoritmo PID
sPid_Calc.r_Cycle	REAL	0.1	Tiempo de muestreo de la instrucción PID_Compact r_Cycle se calcula automáticamente y equivale normalmente al tiempo de ciclo del OB invocante.

Variable	Tipo de datos	Ajuste predetermina do	Descripción
sPid_Calc.b_RunIn	BOOL	FALSE	• b_RunIn = FALSE Si la optimización fina se inicia desde los modos de operación Inactivo o Manual, se inicia una optimización inicial. Si no se cumplen las condiciones para la optimización inicial, PID_Compact se comporta como si b_RunIn = TRUE. Si la optimización fina se inicia desde el modo automático, los parámetros PID existentes se regulan a la consigna. Solo entonces comienza la optimización fina. Si no es posible realizar la optimización inicial, PID_Compact cambia al modo de operación "Inactivo". • b_RunIn = TRUE La optimización inicial se omite. PID_3Compact intenta alcanzar la consigna con el valor de salida mínimo o máximo, lo que puede ocasionar una sobreoscilación muy alta. La optimización fina se inicia entonces automáticamente. b_RunIn se ajusta a FALSE después de la optimización fina.
sPid_Calc.b_CalcParamSUT	BOOL	FALSE	Si b_CalcParamSUT = TRUE, se vuelven a calcular los parámetros de la optimización inicial. De este modo puede cambiarse el método para calcular los parámetros sin que sea necesario repetir la optimización. b_CalcParamSUT se ajusta a FALSE después del cálculo.
sPid_Calc.b_CalcParamTIR	BOOL	FALSE	Si b_CalcParamTIR = TRUE, se vuelven a calcular los parámetros de la optimización fina. De este modo puede cambiarse el método para calcular los parámetros sin que sea necesario repetir la optimización. b_CalcParamTIR se ajusta a FALSE después del cálculo.
sPid_Calc.i_CtrlTypeSUT	INT	0	Calcular los parámetros durante la optimización inicial según el método: • i_CtrlTypeSUT = 0: PID según Chien, Hrones y Reswick • i_CtrlTypeSUT = 1: PI según Chien, Hrones y Reswick

Variable	Tipo de datos	Ajuste predetermina do	Descripción
sPid_Calc.i_CtrlTypeTIR	INT	0	Calcular los parámetros durante la optimización fina según el método: • i_CtrlTypeTIR = 0: PID automático • i_CtrlTypeTIR = 1: PID rápido • i_CtrlTypeTIR = 2: PID lento • i_CtrlTypeTIR = 3: PID Ziegler-Nichols • i_CtrlTypeTIR = 4: PI Ziegler-Nichols • i_CtrlTypeTIR = 5: P Ziegler-Nichols
sPid_Calc.r_Progress	REAL	0.0	Progreso de la optimización en porcentaje (0,0 - 100,0)
sPid_Cmpt.r_Sp_Hlm	REAL	+3.402822e +38	Límite superior de la consigna Si se configura sPid_Cmpt.r_Sp_Hlm a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se utiliza como límite superior de la consigna. Si se configura sPid_Cmpt.r_Sp_Hlm a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de la consigna.
sPid_Cmpt.r_Sp_Llm	REAL	-3.402822e +38	Límite inferior de la consigna Si se configura sPid_Cmpt.r_Sp_Llm a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se utiliza como límite inferior de la consigna. Si se configura sPid_Cmpt.r_Sp_Llm a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de la consigna.
sPid_Cmpt.r_Pv_Norm_IN_1	REAL	0.0	Escalado Input_PER abajo Mediante los dos pares de valores r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 y r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 de la estructura sPid_Cmpt, Input_PER se convierte en porcentaje.
sPid_Cmpt.r_Pv_Norm_IN_2	REAL	27648.0	Escalado Input_PER arriba Mediante los dos pares de valores r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 y r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 de la estructura sPid_Cmpt, Input_PER se convierte en porcentaje.
sPid_Cmpt.r_Pv_Norm_OUT_1	REAL	0.0	Valor real inferior escalado Mediante los dos pares de valores r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 y r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 de la estructura sPid_Cmpt, Input_PER se convierte en porcentaje.
sPid_Cmpt.r_Pv_Norm_OUT_2	REAL	100.0	Valor real superior escalado Mediante los dos pares de valores r_Pv_Norm_OUT_1, r_Pv_Norm_IN_1 y r_Pv_Norm_OUT_2, r_Pv_Norm_IN_2 de la estructura sPid_Cmpt, Input_PER se convierte en porcentaje.
sPid_Cmpt.r_Lmn_Hlm	REAL	100.0	Límite superior del valor de salida para el parámetro de salida "Output"

Variable	Tipo de datos	Ajuste predetermina do	Descripción
sPid_Cmpt.r_Lmn_Llm	REAL	0.0	Límite inferior del valor de salida para el parámetro de salida "Output"
sPid_Cmpt.b_Input_PER_On	BOOL	TRUE	Si b_Input_PER_On = TRUE, se utiliza el parámetro Input_PER. Si b_Input_PER_On = FALSE, se utiliza el parámetro Input.
sPid_Cmpt.b_LoadBackUp	BOOL	FALSE	Activación del juego de parámetros de backup Si hubiera fallado una optimización, es posible volver a activar los parámetros PID anteriores activando este bit.
sPid_Cmpt.b_InvCtrl	BOOL	FALSE	Inversión del sentido de regulación Si b_InvCtrl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida.
sPid_Cmpt.r_Lmn_Pwm_PPTm	REAL	0.0	El tiempo mínimo de conexión de la modulación de ancho de impulsos en segundos se redondea a r_Lmn_Pwm_PPTm = r_Cycle o bien r_Lmn_Pwm_PPTm = n*r_Cycle
sPid_Cmpt.r_Lmn_Pwm_PBTm	REAL	0.0	El tiempo mínimo de desconexión de la modulación de ancho de impulsos en segundos se redondea a r_Lmn_Pwm_PBTm = r_Cycle o bien r_Lmn_Pwm_PBTm = n*r_Cycle
sPid_Cmpt.r_Pv_Hlm	REAL	120.0	Límite superior del valor real En la entrada de periferia, el valor real puede encontrarse como máximo un 18% por encima del rango normalizado (margen de saturación). No se comunica ningún error más debido al rebasamiento por exceso del "Límite superior del valor real". Únicamente se detectan la rotura de hilo y el cortocircuito y PID_Compact pasa al modo de operación "Inactivo". r_Pv_Hlm > r_Pv_Llm
sPid_Cmpt.r_Pv_Llm	REAL	0.0	Límite inferior del valor real r_Pv_Llm < r_Pv_Hlm
sPid_Cmpt.r_Pv_HWrn	REAL	+3.402822e +38	Límite superior de advertencia del valor real Si se configura r_Pv_HWrn a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se utiliza como límite superior de advertencia. Si se configura r_Pv_HWrn a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de advertencia. r_Pv_HWrn > r_Pv_LWrn r_Pv_HWrn ≤ r_Pv_Hlm

Variable	Tipo de datos	Ajuste predeterminado	Descripción
sPid_Cmpt.r_Pv_LWrn	REAL	-3.402822e+38	Límite inferior de advertencia del valor real Si se configura r_Pv_LWrn a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se utiliza como límite inferior de advertencia. Si se configura r_Pv_LWrn a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de advertencia. r_Pv_LWrn < r_Pv_HWrn r_Pv_LWrn ≥ r_Pv_LWrn
sParamCalc.i_Event_SUT	INT	0	La variable i_Event_SUT (Página 5177) muestra la fase actual de la "optimización inicial":
sParamCalc.i_Event_TIR	INT	0	La variable i_Event_TIR (Página 5177) muestra la fase actual de la "optimización fina":
sRet.i_Mode	INT	0	El modo de operación cambia controlado por flanco. El siguiente modo de operación se activa al cambiar a • i_Mode = 0: Modo de operación "Inactivo" (parada del regulador) • i_Mode = 1: Modo de operación "Optimización inicial" • i_Mode = 2: Modo de operación "Optimización fina" • i_Mode = 3: Modo de operación "Modo automático" • i_Mode = 4: Modo de operación "Modo manual" i_Mode es remanente.
sRet.r_Ctrl_Gain	REAL	1.0	Ganancia proporcional activa Gain es remanente.
sRet.r_Ctrl_Ti	REAL	20.0	• r_Ctrl_Ti > 0.0: Tiempo de integración activo • r_Ctrl_Ti = 0.0: La acción I está desactivada r_Ctrl_Ti es remanente.
sRet.r_Ctrl_Td	REAL	0.0	• r_Ctrl_Td > 0.0: Tiempo derivativo activo • r_Ctrl_Td = 0.0: La acción D está desactivada r_Ctrl_Td es remanente.
sRet.r_Ctrl_A	REAL	0.0	Coeficiente activo para el retardo de la acción derivada r_Ctrl_A es remanente.
sRet.r_Ctrl_B	REAL	0.0	Ponderación activa de la acción P r_Ctrl_B es remanente.
sRet.r_Ctrl_C	REAL	0.0	Ponderación activa de la acción D r_Ctrl_C es remanente.
sRet.r_Ctrl_Cycle	REAL	1.0	Tiempo de muestreo activo del algoritmo PID r_Ctrl_Cycle se determina durante la optimización y se redondea a un múltiplo entero de r_Cycle. r_Ctrl_Cycle es remanente.

Nota

Modifique las variables que aparecen aquí en el modo de operación "Inactivo" para evitar un comportamiento erróneo del regulador PID. El modo de operación "Inactivo" se fuerza con el valor "0" en la variable "sRet.i_Mode".

Consulte también

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Parámetros State y sRet.i_Mode V1

Interrelación de los parámetros

El parámetro State indica el modo de operación actual del regulador PID. El parámetro State no puede modificarse.

Para cambiar el modo de operación debe modificarse la variable sRet.i_Mode. Esto también se aplica cuando sRet.i_Mode ya contiene el valor para el nuevo modo de operación. A continuación, por ejemplo, puede ajustarse primero sRet.i_Mode = 0 y, después, sRet.i_Mode = 3. Si el modo de operación actual del regulador permite este cambio, State adopta el valor de sRet.i_Mode.

Cuando PID_Compact cambia automáticamente el modo de operación, se aplica lo siguiente: State != sRet.i_Mode.

Ejemplos:

- Una optimización inicial correcta
  State = 3 y sRet.i_Mode = 1
- En caso de error
  State = 0 y sRet.i_Mode mantiene el valor que tenía hasta ahora; por ejemplo, sRet.i_Mode = 3
- ManualEnalbe = TRUE
  State = 4 y sRet.i_Mode permanecen en el valor actual; por ejemplo, sRet.i_Mode = 3

Nota

Por ejemplo, es posible repetir una optimización fina correcta sin tener que finalizar el modo automático con i_Mode = 0.

Si en un ciclo ajusta sRet.i_Mode a un valor no válido, como puede ser, 9999, esto no afecta en modo alguno a State. En el siguiente ciclo se ajusta Mode = 2. Es posible crear una modificación en sRet.i_Mode sin tener que cambiar primero al modo de operación "Inactivo".

Significado de los valores

State / sRet.i_Mode	Descripción del modo de operación
0	Inactivo El regulador está apagado. Antes de realizar una optimización inicial el regulador se encuentra en el modo de operación "Inactivo". Durante el funcionamiento, el regulador PID cambia al modo de operación "Inactivo" cuando se produce un error o cuando se hace clic en el botón "Desactivar regulador" en la ventana de puesta en servicio.
1	Optimización inicial La optimización inicial determina la respuesta del proceso a un escalón del valor de salida y busca el punto de inflexión. A partir de la inclinación máxima y del tiempo muerto del sistema regulado se calculan los parámetros PID óptimos. Requisitos para la optimización inicial • El regulador se encuentra en el modo de operación inactivo o manual • ManualEnable = FALSE • El valor real no debe aproximarse demasiado a la consigna. |Setpoint - Input| > 0.3 * |sPid_Cmpt.r_Pv_Hlm - sPid_Cmpt.r_Pv_Llm| y |Setpoint - Input| > 0.5 * |Setpoint| • La consigna no puede modificarse durante la optimización inicial. Cuanto más estable sea el valor real, tanto más fácil y preciso será el cálculo de los parámetros PID. El ruido del valor real es admisible en la medida en que el aumento del valor real sea significantemente superior al propio ruido. Antes de volver a calcular los parámetros PID, se guarda una copia de seguridad de los mismos y es posible reactivarlos con sPid_Cmpt.b_LoadBackUp. Una vez realizada correctamente la optimización inicial se pasa al modo automático; si dicha optimización no se realiza correctamente, se pasa al modo de operación "Inactivo". Se muestra la fase de la optimización inicial con Variable i_Event_SUT V1 (Página 5177).

State / sRet.i_Mode	Descripción del modo de operación
2	Optimización fina La optimización fina genera una oscilación constante y limitada del valor real. Los parámetros PID se optimizan a partir de la amplitud y la frecuencia de esta oscilación. Se analizan las diferencias entre la respuesta de proceso durante la optimización inicial y la optimización fina. A partir de los resultados se vuelven a calcular todos los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial. PID_Compact intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan sólo una mínima influencia sobre la optimización fina. Antes de volver a calcular los parámetros PID, se guarda una copia de seguridad de los mismos y es posible reactivarlos con sPid_Cmpt.b_LoadBackUp. Requisitos para la optimización fina: • No se esperan perturbaciones. • La consigna y el valor real se encuentran dentro de los límites configurados. • La consigna no puede modificarse durante la optimización fina. • ManualEnable = FALSE • Modo de operación Automático (State = 3), Inactivo (State = 0) o Manual (State = 4) La optimización fina se realiza en el momento del inicio tal como se indica a continuación: • Modo automático (State = 3) Si desea mejorar los parámetros PID existentes a través de la optimización, inicie la optimización fina desde el modo automático. PID_Comact regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina. • Modo Inactivo (State = 0) o Manual (State = 4) Si se cumplen las condiciones para optimización inicial, ésta se inicia. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina. Si no es preciso realizar la optimización inicial, PID_Compact cambia al modo de operación "Inactivo". Si el valor real para una optimización inicial se encuentra ya muy cerca de la consigna o si sPid_Calc.b_RunIn = TRUE, se intenta alcanzar la consigna con el valor de salida máximo o mínimo. lo que puede ocasionar una sobreoscilación muy alta. Una vez realizada correctamente la optimización fina, el regulador cambia al modo de operación "Automático" y, si dicha optimización no se ejecuta correctamente, se pasa al modo de operación "Inactivo". La fase de la optimización fina se muestra con Variable i_Event_TIR V1 (Página 5177).
3	Modo automático En el modo automático, PID_Compact regula el sistema regulado conforme a los parámetros predefinidos. Cuando se cumple uno de los requisitos siguientes, se cambia al modo automático: • Optimización inicial finalizada correctamente. • Optimización fina finalizada correctamente. • Cambio de la variable sRet.i_Mode al valor 3. Si se conecta la CPU o se cambia de STOP a RUN, PID_Compact arranca en el último modo de operación activo. Para dejar PID_Compact en el modo de operación "Inactivo", ajuste sb_RunModeByStartup = FALSE.
4	Modo manual En el modo manual se especifica un valor de salida manual en el parámetro ManualValue. Este modo de operación se activa si sRet.i_Mode = 4 o en caso de un flanco ascendente en ManualEnable. Si ManualEnable adopta el valor TRUE, únicamente cambia State. sRet.i_Mode permanece en el valor actual. Con un flanco descendente en ManualEnable, PID_Compact regresa al modo de operación previo. El cambio al modo automático se efectúa sin discontinuidad.

Consulte también

Parámetros de salida PID_Compact V1 (Página 5164)

Optimización inicial (Página 9993)

Optimización fina (Página 9994)

Modo de operación "Modo manual" (Página 9996)

Variable i_Event_SUT V1 (Página 5177)

Variable i_Event_TIR V1 (Página 5177)

Parámetro Error V1

Si hay varios errores presentes a la vez, los valores del código de error se muestran sumados binariamente. Si se muestra, p. ej., el código de error 0003, significa que están presentes simultáneamente los errores 0001 y 0002.

Error (DW#16#...)	Descripción
0000	No hay ningún error.
0001	El parámetro "Input" se encuentra fuera de los límites del valor real. • Input > sPid_Cmpt.r_Pv_Hlm o bien, • Input < sPid_Cmpt.r_Pv_Llm Para poder desplazar de nuevo el actuador, previamente se debe eliminar el error.
0002	Hay un valor no válido en el parámetro "Input_PER". Compruebe si hay un error en la entrada analógica.
0004	Error durante la optimización fina. No se ha podido mantener la oscilación del valor real.
0008	Error al iniciar la optimización inicial. El valor real se aproxima demasiado a la consigna. Inicie la optimización fina.
0010	Se ha cambiado la consigna durante la optimización.
0020	La optimización inicial no está permitida en el modo automático ni durante la optimización fina.
0080	Los límites del valor de salida no están configurados correctamente. Compruebe si los límites del valor de salida están configurados correctamente y si encajan con el sentido de la regulación.
0100	Un error en la optimización ha provocado parámetros no válidos.
0200	Valor no válido en el parámetro "Input": El valor no tiene un formato numérico válido.
0400	Error al calcular el valor de salida. Compruebe los parámetros PID.
0800	Error de tiempo de muestreo: PID_Compact no se llama dentro del tiempo de muestreo del OB de alarma cíclica.
1000	Valor no válido en el parámetro "Setpoint": El valor no tiene un formato numérico válido.

Consulte también

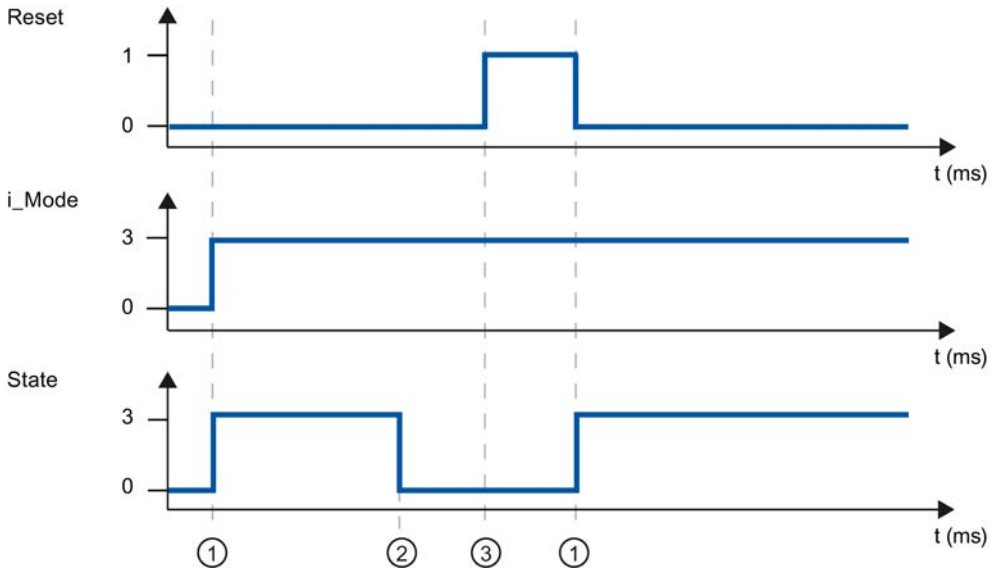
Parámetros de salida PID_Compact V1 (Página 5164)

Parámetro Reset V1

La respuesta en caso de Reset = TRUE depende de la versión de la instrucción PID_Compact.

Respuesta Reset PID_Compact a partir de V.1.1

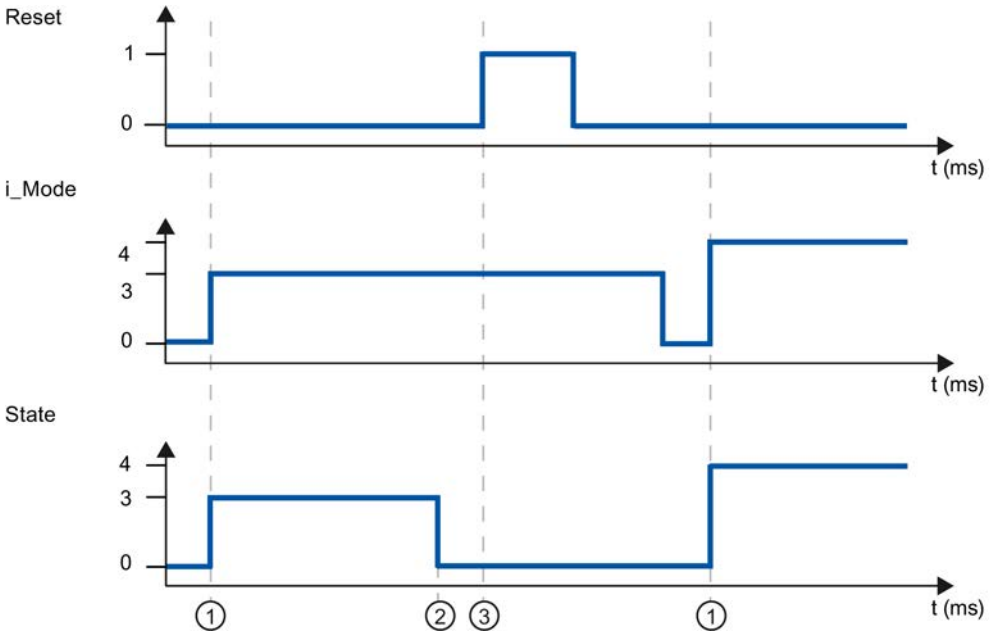
Un flanco ascendente en Reset hace que se desactiven los errores y las advertencias y que se borre la acción I. Un flanco descendente en Reset provoca un cambio al último modo de operación activo.



① Conexión
② Error
③ Reset

Respuesta Reset PID_Compact V.1.0

Un flanco ascendente en Reset hace que se desactiven los errores y las advertencias y que se borre la acción I. El regulador no vuelve a conectarse hasta que no aparece un flanco en i_Mode.



① Conexión
② Error
③ Reset

Variable sd_warning V1

Si hay varias advertencias presentes a la vez, los valores de la variable sd_warning se muestran sumados binariamente. Si se muestra p. ej. la advertencia 0003, significa que están presentes simultáneamente las advertencias 0001 y 0002.

sd_warning (DW#16#....)	Descripción
0000	No hay ninguna advertencia.
0001	Durante la optimización inicial no se encontró el punto de inflexión.
0002	Durante la optimización fina la oscilación fue más intensa.
0004	La consigna estuvo fuera de los límites ajustados.
0008	En el método de cálculo seleccionado no se han definido todas las propiedades necesarias del sistema regulado. En sustitución se calcularon los parámetros PID con el método "i_CtrlTypeTIR = 3".
0010	No se ha podido cambiar el modo de operación, pues ManualEnable = TRUE.
0020	El tiempo de muestreo del algoritmo PID está limitado por el tiempo de ciclo del OB invocante. Para obtener mejores resultados, utilice tiempos de ciclo más cortos para el OB.
0040	El valor real ha rebasado uno de sus límites de advertencia.

Las siguientes advertencias se eliminan en cuanto la causa se soluciona.

- 0004
- 0020
- 0040

Todas las demás advertencias se eliminan si hay un flanco ascendente en Reset.

Variable i_Event_SUT V1

i_Event_SUT	Nombre	Descripción
0	SUT_INIT	Iniciar optimización inicial
100	SUT_STDABW	Calcular desviación estándar
200	SUT_GET_POI	Determinar punto de inflexión
9900	SUT_IO	Optimización inicial correcta
1	SUT_NIO	Optimización inicial no correcta

Consulte también

Variables estáticas PID_Compact V1 (Página 5165)

Parámetros State y sRet.i_Mode V1 (Página 5171)

Variable i_Event_TIR V1

i_Event_TIR	Nombre	Descripción
-100	TIR_FIRST_SUT	No es posible realizar la optimización fina. Se realiza primero una optimización inicial.
0	TIR_INIT	Iniciar optimización fina
200	TIR_STDABW	Calcular desviación estándar
300	TIR_RUN_IN	Intentar alcanzar consigna
400	TIR_CTRLN	Intentar alcanzar consigna con los parámetros PID existentes (si la optimización inicial fue correcta)
500	TIR_OSZIL	Determinar oscilación y calcular parámetros
9900	TIR_IO	Optimización fina correcta
1	TIR_NIO	Optimización fina no correcta

Consulte también

Variables estáticas PID_Compact V1 (Página 5165)

Parámetros State y sRet.i_Mode V1 (Página 5171)

PID_3Step

Novedades PID_3Step

PID_3Step V2.2

- Uso con S7-1200
  A partir de PID_3Step V2.2, la instrucción también se puede utilizar con funcionalidad V2 en una S7-1200 a partir de la versión de firmware 4.0.

PID_3Step V2.0

- Comportamiento en caso de error
  Se ha modificado a fondo el comportamiento con ActivateRecoverMode = TRUE.
  PID_3Step tiene un comportamiento más tolerante a errores en el ajuste predeterminado.

ATENCIÓN
Su instalación puede sufrir daños. Si utiliza el ajuste predeterminado, PID_3Step se mantiene también en modo automático al rebasar los límites del valor real. De este modo, su instalación puede sufrir daños. Configure un comportamiento en caso de error para su sistema regulado que proteja su instalación de daños.

  Con el parámetro de entrada ErrorAck se acusan los errores y las advertencias sin necesidad de reiniciar el regulador ni borrar la acción I.
  Al cambiar el modo de operación, ya no se acusan los errores que no estén pendientes.
- Cambio de modos de operación
  El modo de operación se especifica en el parámetro de entrada/salida Mode y se activa mediante un flanco ascendente en ModeActivate. Se ha suprimido la variable Retain.Mode.
  Ya no es posible iniciar la medición del tiempo de posicionamiento con GetTransitTime.Start, sino solo con Mode = 6 y un flanco ascendente en ModeActivate.
- Capacidad multiinstancia
  Puede llamar PID_3Step como DB multiinstancia. No se creará ningún objeto tecnológico y no dispondrá de interfaz de parametrización y puesta en servicio. Los PID_3Step se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.
- Comportamiento en arranque
  Con RunModeByStartup = TRUE, el modo de operación especificado en Mode se inicia también en caso de flanco descendente en Reset y en caso de arranque en frío de la CPU.
- Comportamiento ENO
  ENO se ajusta en función del modo de operación.
  Si State = 0, entonces ENO = FALSE.
  Si State ≠ 0, entonces ENO = TRUE.

- **Modo manual**
  Los parámetros de entrada Manual_UP y Manual_DN ya no son controlados por flancos. El modo manual controlado por flancos sigue siendo posible mediante las variables ManualUpInternal y ManualDnInternal.
  En el "Modo manual sin señales de tope" (Mode = 10), se ignoran las señales de tope Actuator_H y Actuator_L aunque estén activadas.
- **Ajuste predeterminado de los parámetros PID**
  Se han modificado los siguientes ajustes predeterminados:
  – Ponderación de la acción P (PWeighting), de 0.0 a 1.0
  – Ponderación de la acción D (DWeighting), de 0.0 a 1.0
  – Coeficiente para el retardo de la acción derivada (TdFiltRatio), de 0.0 a 0.2
- **Limitación del tiempo de posicionamiento del motor**
  En la variable Config.VirtualActuatorLimit se especifica en qué % del tiempo de posicionamiento del motor se desplazará como máximo el actuador en un sentido.
- **Especificación de consigna durante la optimización**
  La fluctuación admisible de la consigna durante la optimización se configura en la variable CancelTuningLevel.
- **Control anticipativo**
  El parámetro Disturbance permite activar el control anticipativo.
- **Corrección de errores**
  Si las señales de tope no están activadas (ActuatorEndStopOn = FALSE), Actuator_H y Actuator_L ya no se tienen en cuenta para determinar ScaledFeedback.

PID_3Step V1.1

- **Modo manual en el arranque de la CPU**
  Si al arrancar la CPU ManualEnable = TRUE, PID_3Step arranca en modo manual. No es necesario un flanco ascendente en ManualEnable.
- **Comportamiento en caso de error**
  La variable ActivateRecoverMode ya no actúa en modo manual.
- **Corrección de errores**
  La variable Progress se restablece tras la optimización correcta o la medición del tiempo de posicionamiento.

Compatibilidad con CPU y FW

La siguiente tabla muestra en qué CPU puede usarse cada versión de PID_3Step.

CPU	FW	PID_3Step
S7-1200	V4.X	V2.2 V1.1
S7-1200	V3.X	V1.1 V1.0

CPU	FW	PID_3Step
S7-1200	V2.X	V1.1 V1.0
S7-1200	V1.X	-
S7-1500	V1.1	V2.2 V2.1 V2.0
S7-1500	V1.0	V2.0

PID_3Step V2

Descripción PID_3Step V2

Descripción

Con la instrucción PID_3Step se puede configurar un regulador PID con autooptimización para válvulas o elementos de control final con comportamiento integrador.

Se dispone de los modos de operación siguientes:

- Inactivo
- Optimización inicial
- Optimización fina
- Modo automático
- Modo manual
- Aproximar al valor de salida sustitutivo
- Medición del tiempo de posicionamiento
- Monitorización de errores
- Aproximar al valor de salida sustitutivo con monitorización de errores
- Modo manual sin señales de tope

Los modos de operación se describen detalladamente en el parámetro State.

Algoritmo PID

PID_3Step es un regulador PIDT1 con Anti-Windup y ponderación de la acción P y D. El algoritmo PID funciona de acuerdo con la siguiente fórmula:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
Δy	Valor de salida del algoritmo PID
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
T_D	Tiempo derivativo
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada $T1 = a \times T_D$)
c	Ponderación de la acción D

Diagrama de bloques sin realimentación de posición

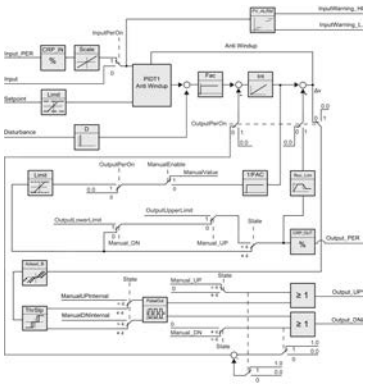


Diagrama de bloques con realimentación de posición

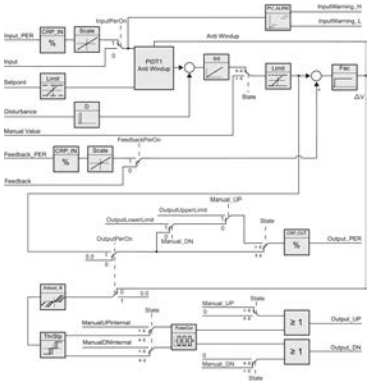
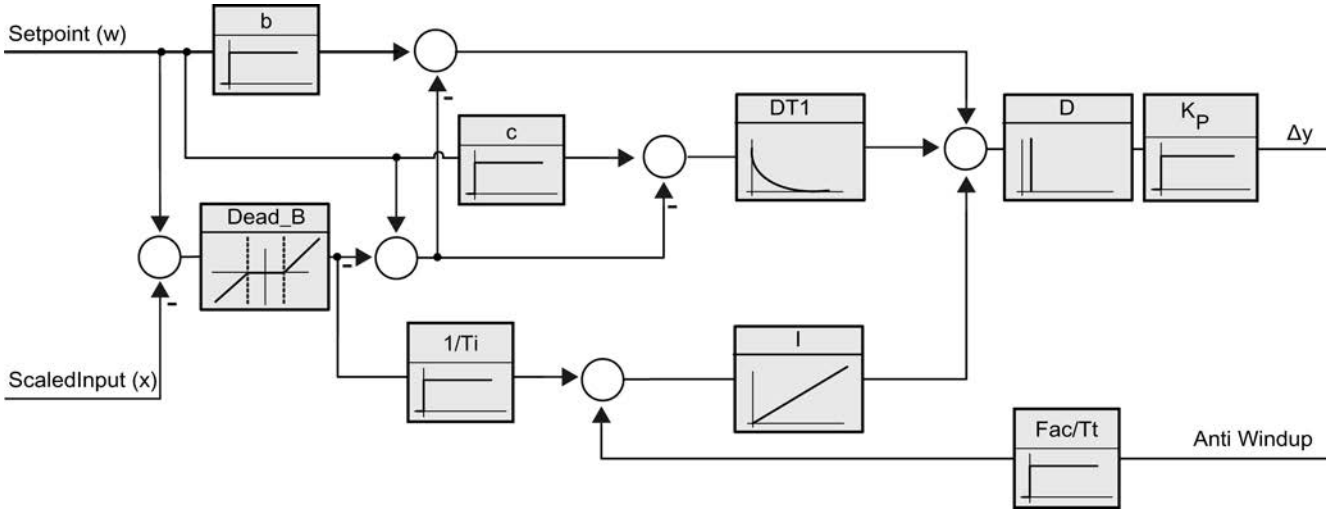


Diagrama de bloques PIDT1 con Anti-Windup



Llamada

PID_3Step se llama en una base de tiempo constante de un OB de alarma cíclica.

Si efectúa una llamada de PID_3Step como DB multiinstancia, no se creará ningún objeto tecnológico. No dispondrá de interfaz de parametrización y puesta en servicio. Los PID_3Step se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Cargar en dispositivo

Los valores actuales de variables remanentes solo se actualizan si se carga PID_3Step por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Arranque

Al arrancar la CPU, PID_3Step se inicia en el modo guardado en el parámetro de entrada/ salida Mode. Para dejar PID_3Step en el modo de operación "Inactivo", ajuste RunModeByStartup = FALSE.

Comportamiento en caso de error

En el modo automático y durante la puesta en servicio, el comportamiento en caso de error depende de las variables ErrorBehaviour y ActivateRecoverMode. En modo manual, el comportamiento no depende de ErrorBehaviour y ActivateRecoverMode. Si ActivateRecoverMode = TRUE, el comportamiento dependerá además del error que se haya producido.

ErrorBehaviour	ActivateRecoverMode	Ajuste del Editor de configuración > Ajuste del actuador > Output a	Comportamiento
FALSE	FALSE	Valor de salida actual	Cambio al modo de operación "Inactivo" (State = 0) El actuador permanece en la posición actual.
FALSE	TRUE	Valor de salida actual por la duración del error	Cambio al modo de operación "Monitorización de errores" (State = 7) El actuador permanece en la posición actual mientras dure el error.
TRUE	FALSE	Valor de salida sustitutivo	Cambio al modo de operación "Aproximar al valor de salida sustitutivo" (State = 5) El actuador se desplaza al valor de salida sustitutivo configurado. Cambio al modo de operación "Inactivo" (State = 0) El actuador permanece en la posición actual.
TRUE	TRUE	Valor de salida sustitutivo mientras dure el error	Cambio al modo de operación "Aproximar a valor de salida sustitutivo con monitorización de errores" (State = 8) El actuador se desplaza al valor de salida sustitutivo configurado. Cambio al modo de operación "Monitorización de errores" (State = 7)

PID_3Step utiliza ManualValue como valor de salida en el modo manual, excepto con los siguientes errores:

- 2000h: Valor no válido en el parámetro Feedback_PER.
- 4000h: Valor no válido en el parámetro Feedback.
- 8000h: Error en la realimentación digital de posición.

Solo es posible modificar la posición del actuador con Manual_UP y Manual_DN, no con ManualValue:

El parámetro Error indica si se ha producido un error en el ciclo actual. El parámetro ErrorBits muestra qué errores se han producido. ErrorBits se reinicia mediante un flanco ascendente en Reset o ErrorAck.

Consulte también

Parámetros State y Mode V2 (Página 5201)

Parámetro ErrorBits V2 (Página 5206)

Configuración de PID_3Step V2 (Página 9998)

Funcionamiento PID_3Step V2

Monitorizar límites del valor real

En las variables Config.InputUpperLimit y Config.InputLowerLimit es posible definir un límite superior y uno inferior del valor real. Si el valor real se encuentra fuera de estos límites, ocurre un error (ErrorBits = 0001h).

En las variables Config.InputUpperWarning y Config.InputLowerWarning se definen un límite de advertencia superior y uno inferior del valor real. Si el valor real se encuentra fuera de estos límites de advertencia, se produce una advertencia (Warning = 0040h) y el parámetro de salida InputWarning_H o InputWarning_L adopta el valor TRUE.

Limitar consigna

En las variables Config.SetpointUpperLimit y Config.SetpointLowerLimit es posible definir un límite superior y uno inferior para la consigna. PID_3Step limita la consigna automáticamente a los límites del valor real. Es posible limitar la consigna a un rango más pequeño. PID_3Step verifica si dicho rango está dentro de los límites del valor real. Si la consigna está fuera de estos límites, se utilizan los límites superior e inferior como consigna y el parámetro de salida SetpointLimit_H o SetpointLimit_L adopta el valor TRUE.

La consigna se limita en todos los modos de operación.

Limitar el valor de salida

En las variables Config.OutputUpperLimit y Config.OutputLowerLimit se define un límite superior y uno inferior del valor de salida. Los límites del valor de salida deben encontrarse dentro del "Tope inferior" o el "Tope superior".

- Tope superior: Config.FeedbackScaling.UpperPointOut
- Tope inferior: Config.FeedbackScaling.LowerPointOut

Rige:

UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut

Los valores válidos para el "Tope superior" y el "Tope inferior" dependen de:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
FALSE	FALSE	FALSE	No ajustable (0.0 %)	No ajustable (100.0 %)
FALSE	TRUE	FALSE	-100.0 % o 0.0 %	0.0 % o +100.0 %
FALSE	TRUE	TRUE	-100.0 % o 0.0 %	0.0 % o +100.0 %
TRUE	FALSE	FALSE	No ajustable (0.0 %)	No ajustable (100.0 %)
TRUE	TRUE	FALSE	-100.0 % o 0.0 %	0.0 % o +100.0 %
TRUE	TRUE	TRUE	-100.0 % o 0.0 %	0.0 % o +100.0 %

Si OutputPerOn = FALSE y FeedbackOn = FALSE, no es posible limitar el valor de salida. Output_UP y Output_DN se restablecen entonces si se cumple Actuator_H = TRUE o Actuator_L = TRUE. Si tampoco existen señales de tope, Output_UP y Output_DN se restablecen después de un tiempo de desplazamiento del Config.VirtualActuatorLimit × Retain.TransitTime/100.

El valor de salida asciende a 27648 con el 100 % y a -27648 con un -100 %. PID_3Step debe poder cerrar la válvula por completo.

Valor de salida sustitutivo

En caso de error, PID_3Step puede emitir un valor de salida sustitutivo y conducir el elemento final de control a una posición segura que se especifica en la variable SavePosition. El valor de salida sustitutivo debe encontrarse dentro de los límites del valor de salida.

Vigilar la validez de las señales

Se vigila la validez de los valores de los siguientes parámetros al utilizarlos:

- Setpoint
- Input
- Input_PER
- Input_PER
- Feedback
- Feedback_PER
- Disturbance
- ManualValue
- SavePosition
- Output_PER

Vigilar el tiempo de muestreo PID_3Step

En el caso ideal, el tiempo de muestreo equivale al tiempo de ciclo del OB invocante. La instrucción PID_3Step mide en cada caso el tiempo que transcurre entre dos llamadas. Este es el tiempo de muestreo actual. Con cada cambio de modo de operación y en el primer arranque se calcula la media de los 10 primeros tiempos de muestreo. Si el tiempo de muestreo actual se desvía mucho de este valor medio, ocurre un error (ErrorBits = 0800h).

Durante la optimización, el error se produce si:

- Valor medio nuevo >= 1,1 veces el valor medio antiguo
- Valor medio nuevo <= 0,9 veces el valor medio antiguo

En el modo automático, el error se produce si:

- Valor medio nuevo >= 1,5 veces el valor medio antiguo
- Valor medio nuevo <= 0,5 veces el valor medio antiguo

Si se desactiva la monitorización del tiempo de muestreo (CycleTime.EnMonitoring = FALSE), puede llamarse PID_3Step también en el OB1. En tal caso, deberá aceptarse una regulación de menor calidad, debido a la fluctuación del tiempo de muestreo.

Tiempo de muestreo del algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de ciclo. Todas las demás funciones de PID_3Step se ejecutan con cada llamada.

Medir el tiempo de posicionamiento del motor

El tiempo de posicionamiento del motor es el tiempo en segundos que requiere el motor para desplazar el elemento final de control del estado cerrado al estado abierto. El elemento final de control se mueve en un sentido a como máximo Config.VirtualActuatorLimit × Retain.TransitTime/100. PID_3Step necesita un tiempo de posicionamiento del motor lo más preciso posible para alcanzar un buen resultado de regulación. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento. El tiempo de posicionamiento del motor puede medirse durante la puesta en servicio. Los límites del valor de salida no se tienen en cuenta durante la medición del tiempo de posicionamiento del motor. El actuador puede desplazarse hasta el tope superior o inferior.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal. Para refrigeración o para las regulaciones de desagües puede ser necesario invertir el sentido de regulación. PID_3Step no funciona con ganancia proporcional negativa. Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida. El sentido de regulación se tiene en cuenta también durante la optimización inicial y la optimización fina.

Consulte también

Configuración de PID_3Step V1 (Página 10016)

Modificaciones de la interfaz PID_3Step V2

La siguiente tabla muestra lo que ha cambiado en la interfaz de la instrucción PID_3Step.

PID_3Step V1	PID_3Step V2	Cambio
Input_PER	Input_PER	Tipo de datos de Word a Int
Feedback_PER	Feedback_PER	Tipo de datos de Word a Int
	Disturbance	Nuevo
Manual_UP	Manual_UP	Función
Manual_DN	Manual_DN	Función
	ErrorAck	Nuevo
	ModeActivate	Nuevo
Output_PER	Output_PER	Tipo de datos de Word a Int
	ManualUPInternal	Nuevo
	ManualDNInternal	Nuevo
	CancelTuningLevel	Nuevo
	VirtualActuatorLImit	Nuevo
Config.Loadbackup	Loadbackup	Cambio de nombre
Config.TransitTime	Retain.TransitTime	Cambio de nombre y remanencia añadida
GetTransitTime.Start		Sustituido por Mode y ModeActivate
SUT.CalculateSUTParams	SUT.CalculateParams	Cambio de nombre
SUT.TuneRuleSUT	SUT.TuneRule	Cambio de nombre
TIR.CalculateTIRParams	TIR.CalculateParams	Cambio de nombre
TIR.TuneRuleTIR	TIR.TuneRule	Cambio de nombre
Retain.Mode	Mode	Función Declaración de Static a parámetros de entrada/salida

Parámetros de entrada PID_3Step V2

Tabla 9-88

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
Setpoint	REAL	0.0	Consigna del regulador PID en modo automático
Input	REAL	0.0	Una variable del programa de usuario se utiliza como origen del valor real. Si utiliza el parámetro Input, debe cumplirse lo siguiente: Config.InputPerOn = FALSE.

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
Input_PER	INT	0	Una entrada analógica se utiliza como origen del valor real. Si utiliza el parámetro Input_PER, debe cumplirse lo siguiente: Config.InputPerOn = TRUE.
Actuator_H	BOOL	FALSE	Realimentación de posicionamiento digital de la válvula para el tope superior Si Actuator_H = TRUE la posición de la válvula se encuentra en el tope superior y la válvula ya no se desplaza en ese sentido.
Actuator_L	BOOL	FALSE	Realimentación de posición digital de la válvula para el tope inferior Si Actuator_L = TRUE la posición de la válvula se encuentra en el tope inferior y la válvula ya no se desplaza en ese sentido.
Feedback	REAL	0.0	Realimentación de posición de la válvula Si utiliza el parámetro Feedback, debe cumplirse lo siguiente: Config.FeedbackPerOn = FALSE.
Feedback_PER	INT	0	Realimentación de posición analógica de una válvula Si utiliza el parámetro Feedback_PER, debe cumplirse lo siguiente: Config.FeedbackPerOn = TRUE. Feedback_PER se escala a partir de las variables: • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut
Disturbance	REAL	0.0	Magnitud perturbadora o control anticipativo
ManualEnable	BOOL	FALSE	• El flanco FALSE -> TRUE activa el modo de operación "Modo manual", State = 4, Mode no cambia. Mientras se cumple ManualEnable = TRUE, no es posible cambiar el modo de operación mediante un flanco ascendente en ModeActivate, ni utilizar el diálogo de puesta en servicio. • El flanco TRUE -> FALSE activa el modo de operación especificado en Mode. Se recomienda cambiar el modo de operación solo mediante ModeActivate.
ManualValue	REAL	0.0	En el modo manual se determina la posición absoluta de la válvula. ManualValue solo se evalúa si se utiliza Output_PER o si se dispone de una realimentación de posición.

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
Manual_UP	BOOL	FALSE	• Manual_UP = TRUE La válvula se abre aunque se utilice Output_PER o una realimentación de posición. La válvula deja de moverse cuando se alcanza el tope superior. Ver también Config.VirtualActuatorLimit • Manual_UP = FALSE Si se utiliza Output_PER o una realimentación de posición, la válvula se desplaza a ManualValue. En caso contrario, la válvula deja de moverse. Si Manual_UP y Manual_DN se ajustan simultáneamente a TRUE, la válvula no se mueve.
Manual_DN	BOOL	FALSE	• Manual_DN = TRUE La válvula se cierra aunque se utilice Output_PER o una realimentación de posición. La válvula deja de moverse cuando se alcanza el tope inferior. Ver también Config.VirtualActuatorLimit • Manual_DN = FALSE Si se utiliza Output_PER o una realimentación de posición, la válvula se desplaza a ManualValue. En caso contrario, la válvula deja de moverse.
ErrorAck	BOOL	FALSE	• Flanco FALSE -> TRUE ErrorBits y Warning se desactivan.
Reset	BOOL	FALSE	Realiza un rearranque completo del regulador. • Flanco FALSE -> TRUE – Cambio al modo de operación "Inactivo" – ErrorBits y Warning se desactivan. – Se borra la acción I (los parámetros PID se mantienen) • Mientras se cumple Reset = TRUE, PID_3Step permanece en el modo "Inactivo" (State = 0). • Flanco TRUE -> FALSE PID_3Step cambia al modo guardado en Mode.
ModeActivate	BOOL	FALSE	• Flanco FALSE -> TRUE PID_3Step cambia al modo guardado en Mode.

Parámetros de salida PID_3Step V2

Tabla 9-89

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
ScaledInput	REAL	0.0	Valor real escalado
ScaledFeedback	REAL	0.0	Realimentación de posición escalada Con actuadores sin realimentación de posición, ScaledFeedback indica la posición del actuador de modo muy impreciso. En tal caso, ScaledFeedback solo puede usarse para una estimación aproximada de la posición actual.
Output_UP	BOOL	FALSE	Valor de salida digital para abrir la válvula Si Config.OutputPerOn = FALSE, se utiliza el parámetro Output_UP.
Output_DN	BOOL	FALSE	Valor de salida digital para cerrar la válvula Si Config.OutputPerOn = FALSE, se utiliza el parámetro Output_DN.
Output_PER	INT	0	Valor de salida analógico Si Config.OutputPerOn = TRUE, se utiliza Output_PER.
SetpointLimit_H	BOOL	FALSE	Si SetpointLimit_H = TRUE se cumple, significa que se ha alcanzado el límite superior absoluto de la consigna (Setpoint ≥ Config.SetpointUpperLimit). La consigna está limitada a Config.SetpointUpperLimit .
SetpointLimit_L	BOOL	FALSE	Si SetpointLimit_L = TRUE se cumple, significa que se ha alcanzado el límite inferior absoluto de la consigna (Setpoint ≤ Config.SetpointLowerLimit). La consigna está limitada a Config.SetpointLowerLimit .
InputWarning_H	BOOL	FALSE	Cuando InputWarning_H = TRUE, significa que se ha alcanzado o rebasado el límite superior de advertencia del valor real.
InputWarning_L	BOOL	FALSE	Cuando InputWarning_L = TRUE, significa que se ha alcanzado o rebasado por defecto el límite inferior de advertencia del valor real.

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
State	INT	0	El parámetro State (Página 5201) indica el modo de operación actual del regulador PID. El modo de operación se cambia con el parámetro de entrada Mode y un flanco ascendente en ModeActivate. • State = 0: Inactivo • State = 1: Optimización inicial • State = 2: Optimización fina • State = 3: Modo automático • State = 4: Modo manual • State = 5: Aproximar al valor de salida sustitutivo • State = 6: Medición del tiempo de posicionamiento • State = 7: monitorización de errores • State = 8: Aproximar al valor de salida sustitutivo con monitorización de errores • State = 10: Modo manual sin señales de tope
Error	BOOL	FALSE	Si Error = TRUE, existe al menos un mensaje de error en el ciclo actual.
ErrorBits	DWORD	DW#16#0	El parámetro ErrorBits (Página 5206) muestra qué mensajes de error existen. ErrorBits es remanente y se restablece con un flanco ascendente en Reset o ErrorAck.

Consulte también

Parámetros State y Mode V2 (Página 5201)

Parámetro ErrorBits V2 (Página 5206)

Parámetros de entrada/salida PID_3Step V2

Tabla 9-90

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
Mode	INT	4	Especifique en Mode el modo al que debe cambiar PID_3Step. Posibilidades: • Mode = 0: Inactivo • Mode = 1: Optimización inicial • Mode = 2: Optimización fina • Mode = 3: Modo automático • Mode = 4: Modo manual • Mode = 6: Medición del tiempo de posicionamiento • Mode = 10: Modo manual sin señales de tope El modo se activa mediante: • Flanco ascendente en ModeActivate • Flanco descendente en Reset • Flanco descendente en ManualEnable • Arranque en frío de la CPU si RunModeByStartup = TRUE Mode es remanente. Encontrará una descripción detallada de los modos en Parámetros State y Mode V2 (Página 5201).

Variables estáticas PID_3Step V2

No se deben modificar las variables que no aparecen listadas. Estas solo se utilizan internamente.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
ManualUpInternal	BOOL	FALSE	En el modo manual, cada flanco ascendente abre la válvula un 5% del rango de posicionamiento total o durante el tiempo mínimo de posicionamiento del motor. ManualUpInternal solo se evalúa si no se utiliza Output_PER ni se dispone de realimentación de posición. Esta variable se utiliza en el diálogo de puesta en servicio.
ManualDnInternal	BOOL	FALSE	En el modo manual, cada flanco ascendente cierra la válvula un 5% del rango de posicionamiento total o durante el tiempo mínimo de posicionamiento del motor. ManualDnInternal solo se evalúa si no se utiliza Output_PER ni se dispone de realimentación de posición. Esta variable se utiliza en el diálogo de puesta en servicio.
ActivateRecoverMode	BOOL	TRUE	La variable ActivateRecoverMode V2 (Página 5209) determina el comportamiento en caso de error.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
RunModeByStartup	BOOL	TRUE	Activar Mode después del rearranque de la CPU Si RunModeByStartup = TRUE, PID_3Step se inicia después del arranque de la CPU en el modo guardado en Mode. Si RunModeByStartup = FALSE, PID_3Step permanece en modo "Inactivo" después del arranque de la CPU.
LoadBackUp	BOOL	FALSE	Si LoadBackUp = TRUE, se carga nuevamente el último juego de parámetros PID. El juego se guardó antes de la última optimización. LoadBackUp se ajusta automáticamente de nuevo a FALSE.
PhysicalUnit	INT	0	Unidad física del valor real y de la consigna, p. ej. ºC o ºF.
PhysicalQuantity	INT	0	Unidad física del valor real y de la consigna, p. ej. temperatura
ErrorBehaviour	BOOL	FALSE	Si ErrorBehaviour = FALSE, en caso de error la válvula permanece en la posición actual y el regulador cambia directamente al modo de operación "Inactivo" o "Monitorización de errores". Si ErrorBehaviour = TRUE, en caso de error el actuador se desplaza al valor de salida sustitutivo y solo entonces pasa al modo de operación "Inactivo" o "Monitorización de errores". Si se producen los errores siguientes, la válvula ya no se puede desplazar a un valor de salida sustitutivo configurado. • 2000h: Valor no válido en el parámetro Feedback_PER. • 4000h: Valor no válido en el parámetro Feedback. • 8000h: Error en la realimentación digital de posición. • 20000h: Valor no válido en la variable SavePosition.
Warning	DWORD	DW#16#0	La variable Warning (Página 5201) muestra las advertencias desde Reset = TRUE o ErrorAck =TRUE. Warning es remanente. Las advertencias cíclicas (por ejemplo, advertencia del valor real) se muestran mientras sigue existiendo la causa de la advertencia. Al eliminar la causa, se borran automáticamente. Las advertencias no cíclicas (como es la relativa al punto de inflexión no encontrado) se mantienen y se borran como se hace con los errores.
SavePosition	REAL	0.0	Valor de salida sustitutivo Si ErrorBehaviour = TRUE, en caso de error el elemento final de control o actuador se desplaza a una posición segura para la instalación. En cuanto se alcanza el valor de salida sustitutivo, PID_3Step cambia de modo en función del valor de ActivateRecoverMode.
CurrentSetpoint	REAL	0.0	Consigna activa actualmente. Este valor se congela al iniciarse la optimización.
CancelTuningLevel	REAL	10.0	Fluctuación admisible de la consigna durante la optimización. La optimización no se cancela hasta que no se cumple: • Setpoint > CurrentSetpoint + CancelTuningLevel o bien, • Setpoint < CurrentSetpoint - CancelTuningLevel
Progress	REAL	0.0	Progreso de la optimización en porcentaje (0.0 - 100.0)
Config.InputPerOn	BOOL	TRUE	Si InputPerOn = TRUE, se utiliza el parámetro Input_PER. Si InputPerOn = FALSE, se utiliza el parámetro Input.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Config.OutputPerOn	BOOL	FALSE	Si OutputPerOn = TRUE, se utiliza el parámetro Output_PER. Si OutputPerOn = FALSE, se utilizan los parámetros Ouput_UP y Output_DN.
Config.InvertControl	BOOL	FALSE	Inversión del sentido de regulación Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida.
Config.FeedbackOn	BOOL	FALSE	Si FeedbackOn = FALSE, se simula una realimentación de posición. Si FeedbackOn = TRUE, se activa por lo general la realimentación de posición.
Config.FeedbackPerOn	BOOL	FALSE	FeedbackPerOn solo está activo cuando FeedbackOn = TRUE. Si FeedbackPerOn = TRUE, se utiliza la entrada analógica para la realimentación de posición (parámetro Feedback_PER). Si FeedbackPerOn = FALSE, se utiliza el parámetro Feedback para la realimentación de posición.
Config.ActuatorEndStopOn	BOOL	FALSE	Si ActuatorEndStopOn = TRUE, se tiene en cuenta la realimentación de posición digital Actuator_L y Actuator_H .
Config.InputUpperLimit	REAL	120.0	Límite superior del valor real Se vigila que Input y Input_PER respeten estos límites. En la entrada de periferia, el valor real puede encontrarse como máximo un 18% por encima del rango normalizado (margen de saturación). No se comunica ningún error más debido al rebase por exceso del "Límite superior del valor real". Únicamente se detectan la rotura de hilo y el cortocircuito y PID_3Step se comporta según la reacción configurada en caso de error. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit	REAL	0.0	Límite inferior del valor real InputLowerLimit < InputUpperLimit
Config.InputUpperWarning	REAL	+3.40282 2e+38	Límite superior de advertencia del valor real Si se configura InputUpperWarning a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se utiliza como límite superior de advertencia. Si se configura InputUpperWarning a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de advertencia. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning	REAL	-3.40282 2e+38	Límite inferior de advertencia del valor real Si se configura InputLowerWarning a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se utiliza como límite inferior de advertencia. Si se configura InputLowerWarning a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de advertencia. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Config.OutputUpperLimit	REAL	100.0	Límite superior del valor de salida Para más detalles ver OutputLowerLimit
Config.OutputLowerLimit	REAL	0.0	Límite inferior del valor de salida Si OutputPerOn = TRUE o FeedbackOn = TRUE, el rango de valores válido va de -100 hasta +100 % incluyendo el cero. -100 % corresponde a Output = -27648; +100 % corresponde a Output = 27648 Si OutputPerOn = FALSE, el rango de valores válido va de 0 hasta 100 %. En 0 % la válvula está completamente cerrada y en 100 % completamente abierta.
Config.SetpointUpperLimit	REAL	+3.402822e+38	Límite superior de la consigna Si se configura SetpointUpperLimit a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se preajusta como límite superior de la consigna. Si se configura SetpointUpperLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de la consigna.
Config.SetpointLowerLimit	REAL	-3.402822e+38	Límite inferior de la consigna Si se configura SetpointLowerLimit a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se preajusta como límite inferior de la consigna. Si se configura SetpointLowerLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de la consigna.
Config.MinimumOnTime	REAL	0.0	Tiempo conexión mín. Tiempo en segundos, que debe estar conectado como mínimo el elemento final de control.
Config.MinimumOffTime	REAL	0.0	Tiempo OFF mín. Tiempo en segundos, que debe estar desconectado como mínimo el elemento final de control.
Config.VirtualActuatorLimit	REAL	150.0	Si se cumplen todas las condiciones siguientes, el actuador se desplaza como máximo durante VirtualActuatorLimit × Retain.TransitTime/100 en una dirección y se emite la advertencia 2000h: • Config.OutputPerOn = FALSE • Config.ActuatorEndStopOn = FALSE • Config.FeedbackOn = FALSE Si se cumple Config.OutputPerOn = FALSE y Config.ActuatorEndStopOn = TRUE o Config.FeedbackOn = TRUE, solo se emite la advertencia 2000h. Si se cumple Config.OutputPerOn = TRUE, no se tiene en cuenta VirtualActuatorLimit.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Config.InputScaling.UpperPointIn	REAL	27648.0	Escalado Input_PER arriba Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.InputScaling.LowerPointIn	REAL	0.0	Escalado Input_PER abajo Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.InputScaling.UpperPointOut	REAL	100.0	Valor real superior escalado Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.InputScaling.LowerPointOut	REAL	0.0	Valor real inferior escalado Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.FeedbackScaling.UpperPointIn	REAL	27648.0	Escalado Feedback_PER arriba Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
Config.FeedbackScaling.LowerPointIn	REAL	0.0	Escalado Feedback_PER abajo Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
Config.FeedbackScaling.UpperPointOut	REAL	100.0	Tope superior Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
Config.FeedbackScaling.LowerPointOut	REAL	0.0	Tope inferior Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
GetTransitTime.InvertDirection	BOOL	FALSE	Si InvertDirection = FALSE, se abre completamente la válvula para determinar el tiempo de posicionamiento, luego se cierra y se vuelve a abrir. Si InvertDirection = TRUE, se cierra completamente la válvula, luego se abre y se vuelve a cerrar.
GetTransitTime.SelectFeedback	BOOL	FALSE	Si SelectFeedback = TRUE, durante la medición del tiempo de posicionamiento se tendrá en cuenta Feedback_PER o Feedback. Si SelectFeedback = FALSE, durante la medición del tiempo de posicionamiento se tendrán en cuenta Actuator_H y Actuator_L.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
GetTransitTime.State	INT	0	Fase actual de la medición del tiempo de posicionamiento • State = 0: Inactivo • State = 1: Abrir válvula completamente • State = 2: Cerrar válvula completamente • State = 3: Desplazar válvula a la posición de destino (NewOutput) • State = 4: Tiempo de posicionamiento medido correctamente • State = 5: Medición del tiempo de posicionamiento cancelada
GetTransitTime.NewOutput	REAL	0.0	Posición de destino para la medición del tiempo de posicionamiento con realimentación de posición La posición de destino debe encontrarse dentro de los topes superior e inferior. La diferencia entre NewOutput y ScaledFeedback debe estar al menos por encima del 50% del margen de ajuste admisible.
CycleTime.StartEstimation	BOOL	TRUE	Si StartEstimation = TRUE, se inicia la medición del tiempo de muestreo PID_3Step. Una vez finalizada la medición, CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Si EnEstimation = TRUE, se calcula el tiempo de muestreo PID_3Step. Si CycleTime.EnEstimation = FALSE, el tiempo de muestreo PID_3Step no se calcula y debe configurar CycleTime.Value correctamente a mano.
CycleTime.EnMonitoring	BOOL	TRUE	Si EnMonitoring = TRUE, se vigila el tiempo de muestreo PID_3Step. Si no es posible ejecutar PID_3Step dentro del tiempo de muestreo, se emite el error 0800h y se cambia el modo de operación. El modo de operación al que se cambie depende de ActivateRecoverMode y de ErrorBehaviour. Si EnMonitoring = FALSE, el tiempo de muestreo PID_3Step no se vigila, el error 0800h no se emite y el modo de operación no se cambia.
CycleTime.Value	REAL	0.1	Tiempo de muestreo PID_3Step en segundos CycleTime.Value se determina automáticamente y equivale normalmente al tiempo de ciclo del OB invocante.
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Valor almacenado de Retain.CtrlParams.SetByUser Los valores de la estructura CtrlParamsBackUp pueden volverse a cargar con LoadBackUp = TRUE.
CtrlParamsBackUp.Gain	REAL	1.0	Ganancia proporcional almacenada
CtrlParamsBackUp.Ti	REAL	20.0	Tiempo de integración almacenado en segundos
CtrlParamsBackUp.Td	REAL	0.0	Tiempo derivativo almacenado en segundos
CtrlParamsBackUp.TdFiltRatio	REAL	0.2	Coeficiente almacenado para el retardo de la acción derivada
CtrlParamsBackUp.PWeighting	REAL	1.0	Ponderación almacenada de la acción P
CtrlParamsBackUp.DWeighting	REAL	1.0	Ponderación almacenada de la acción D
CtrlParamsBackUp.Cycle	REAL	1.0	Tiempo de muestreo almacenado del algoritmo PID en segundos

Variable	Tipo de datos	Ajuste predeter minado	Descripción
CtrlParamsBackUp.InputDead Band	REAL	0.0	Ancho de zona muerta almacenado del error de regulación
PIDSelfTune.SUT.CalculateP arams	BOOL	FALSE	Las propiedades del sistema regulado se almacenan durante la optimización. Si CalculateParams = TRUE, se vuelven a calcular los parámetros PID basándose en estas propiedades. Los parámetros PID se calculan según el método ajustado en TuneRule. CalculateParams se ajusta a FALSE después del cálculo.
PIDSelfTune.SUT.TuneRule	INT	1	Calcular los parámetros durante la optimización inicial según el método: • SUT.TuneRule = 0: PID rápido I • SUT.TuneRule = 1: PID lento I • SUT.TuneRule = 2: PID según Chien, Hrones, Reswick • SUT.TuneRule = 3: PI según Chien, Hrones, Reswick • SUT.TuneRule = 4: PID rápido II • SUT.TuneRule = 5: PID lento II
PIDSelfTune.SUT.State	INT	0	La variable SUT.State muestra la fase actual de la optimización inicial: • State = 0: Iniciar la optimización inicial • State = 50: Determinar la posición inicial sin realimentación de posición • State = 100: Calcular desviación estándar • State = 200: Determinar punto de inflexión • State = 300: Determinar el tiempo de subida • State = 9900: Optimización inicial correcta • State = 1: Optimización inicial no correcta
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Con la variable RunIn puede determinar que se realice una optimización fina incluso sin optimización inicial. • RunIn = FALSE Si la optimización fina se inicia desde los modos de operación Inactivo o Manual, se inicia una optimización inicial. Si la optimización fina se inicia desde el modo automático, los parámetros PID existentes se regulan a la consigna. Solo entonces comienza la optimización fina. Si no es posible realizar una optimización inicial, PID_3Step pasa al modo desde el que se inició la optimización. • RunIn = TRUE La optimización inicial se omite. PID_3Step intenta alcanzar la consigna con el valor de salida mínimo o máximo, lo que puede ocasionar una sobreoscilación muy alta. Solo entonces comienza la optimización fina. RunIn se ajusta a FALSE después de la optimización fina.
PIDSelfTune.TIR.CalculatePar ams	BOOL	FALSE	Las propiedades del sistema regulado se almacenan durante la optimización. Si CalculateParams = TRUE, se vuelven a calcular los parámetros PID basándose en estas propiedades. Los parámetros PID se calculan según el método ajustado en TuneRule. CalculateParams se ajusta a FALSE después del cálculo.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
PIDSelfTune.TIR.TuneRule	INT	0	Calcular los parámetros durante la optimización fina según el método: • TIR.TuneRule = 0: PID automático • TIR.TuneRule = 1: PID rápido • TIR.TuneRule = 2: PID lento • TIR.TuneRule = 3: PID Ziegler-Nichols • TIR.TuneRule = 4: PI Ziegler-Nichols • TIR.TuneRule = 5: P Ziegler-Nichols
PIDSelfTune.TIR.State	INT	0	La variable TIR.State muestra la fase actual de la optimización fina: • State = -100 No es posible realizar la optimización fina. Primero se efectuará una optimización inicial. • State = 0: Iniciar la optimización fina • State = 200: Calcular desviación estándar • State = 300: Intentar alcanzar la consigna con el valor de salida máximo o mínimo. • State = 400: Intentar alcanzar consigna con los parámetros PID actuales (si la optimización inicial ha sido correcta) • State = 500: Determinar oscilación y calcular parámetros • State = 9900: Optimización fina correcta • State = 1: Optimización fina no correcta
Retain.TransitTime	REAL	30.0	Tiempo de posicionamiento del motor en segundos Tiempo en segundos que requiere el elemento final de control para desplazar la válvula del estado cerrado al estado abierto. TransitTime es remanente.
Retain.CtrlParams.SetByUser	BOOL	FALSE	Si SetByUser = FALSE, los parámetros PID se determinan de forma automática y PID_3Step funciona con una zona muerta en el valor de salida. El ancho de la zona muerta se calcula durante la optimización a partir de una desviación estándar del valor de salida y se almacena en Retain.CtrlParams.OutputDeadBand. Si SetByUser = TRUE, los parámetros PID se introducen manualmente y PID_3 Step funciona sin zona muerta en el valor de salida.Retain.CtrlParams.OutputDeadBand = 0.0 SetByUser es remanente.
Retain.CtrlParams.Gain	REAL	1.0	Ganancia proporcional activa Para invertir el sentido de regulación utilice la variable Config.InvertControl. Los valores negativos de Gain invierten también el sentido de regulación. Se recomienda que ajuste el sentido de regulación únicamente a través de InvertControl. Si se cumplen InvertControl = TRUE y Gain < 0.0, el sentido de regulación también se invierte. Gain es remanente.
Retain.CtrlParams.Ti	REAL	20.0	• Ti > 0.0: Tiempo de integración activo en segundos • Ti = 0.0: La acción I está desactivada Ti es remanente.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Retain.CtrlParams.Td	REAL	0.0	• Td > 0.0: Tiempo derivativo activo en segundos • Td = 0.0: La acción D está desactivada Td es remanente.
Retain.CtrlParams.TdFiltRatio	REAL	0.2	Coeficiente activo para el retardo de la acción derivada El efecto de la acción D se retrasa mediante el coeficiente de retardo de la acción derivada. Retardo de la acción derivada = Tiempo derivativo x coeficiente de retardo de la acción derivada • 0.0: la acción D solo surte efecto para un ciclo y, por ello, casi no es efectiva. • 0.5: este valor se ha acreditado en la práctica para sistemas regulados con **una** constante de tiempo dominante. • > 1.0: cuanto mayor sea el coeficiente, más se retrasará el efecto de la acción D. TdFiltRatio es remanente.
Retain.CtrlParams.PWeighting	REAL	1.0	Ponderación activa de la acción P En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción P. Se recomiendan valores comprendidos entre 0.0 y 1.0. • 1.0: la acción P es totalmente efectiva cuando se cambia la consigna • 0.0: la acción P no actúa al cambiar la consigna Si se produce una modificación del valor real, la acción P es totalmente efectiva. PWeighting es remanente.
Retain.CtrlParams.DWeightin g	REAL	1.0	Ponderación activa de la acción D En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción D. Se recomiendan valores comprendidos entre 0.0 y 1.0. • 1.0: la acción D es totalmente efectiva cuando se cambia la consigna. • 0.0: la acción D no actúa al cambiar la consigna Si se produce una modificación del valor real, la acción D es totalmente efectiva. DWeighting es remanente.
Retain.CtrlParams.Cycle	REAL	1.0	Tiempo de muestreo activo del algoritmo PID en segundos que se redondea a un múltiplo entero del tiempo de ciclo del OB invocante. Cycle es remanente.
Retain.CtrlParams.InputDead Band	REAL	0.0	Ancho de zona muerta del error de regulación InputDeadBand es remanente.

Nota

Modifique las variables que aparecen aquí en el modo de operación "Inactivo" para evitar un comportamiento erróneo del regulador PID.

Consulte también

Parámetros State y Mode V2 (Página 5201)

Variable ActivateRecoverMode V2 (Página 5209)

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Parámetros State y Mode V2

Interrelación de los parámetros

El parámetro State indica el modo de operación actual del regulador PID. El parámetro State no puede modificarse.

Con un flanco ascendente en ModeActivate, PID_3Step pasa al modo guardado en el parámetro de entrada/salida Mode.

Al conectar la CPU o cambiar de STOP a RUN, PID_3Step se inicia en el modo de operación guardado en Mode. Para dejar PID_3Step en el modo de operación "Inactivo", ajuste RunModeByStartup = FALSE.

Significado de los valores

State	Descripción del modo de operación
0	Inactivo El regulador está desconectado y ya no modifica la posición de la válvula.
1	Optimización inicial La optimización inicial determina el comportamiento del proceso a un impulso del valor de salida y busca el punto de inflexión. Los parámetros PID óptimos se calculan a partir de la pendiente máxima y el tiempo muerto del sistema regulado. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina. Requisitos para la optimización inicial: • El tiempo de posicionamiento del motor está configurado o medido. • Modo de operación Inactivo (State = 0), Manual (State = 4) o Automático (State = 3) • ManualEnable = FALSE • Reset = FALSE • La consigna y el valor real se encuentran dentro de los límites configurados. Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. Esto tiene más probabilidades de suceder en los modos "Inactivo" o "Manual". La consigna se congela en la variable CurrentSetpoint. La optimización no se cancela hasta que no se cumple: • Setpoint > CurrentSetpoint + CancelTuningLevel o bien, • Setpoint < CurrentSetpoint - CancelTuningLevel Antes de volver a calcular los parámetros PID, se guarda una copia de seguridad de estos y es posible reactivarlos con LoadBackUp. Una vez realizada correctamente la optimización inicial, el regulador pasa al modo automático; si la optimización inicial no se ha realizado correctamente, el cambio del modo de operación depende de ActivateRecoverMode y ErrorBehaviour. La fase de optimización inicial se muestra con la variable SUT.State.

State	Descripción del modo de operación
2	Optimización fina La optimización fina genera una oscilación constante y limitada del valor real. A partir de la amplitud y frecuencia de esta oscilación se calculan de nuevo los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina. PID_3Step intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan solo una mínima influencia sobre la optimización fina. La consigna se congela en la variable CurrentSetpoint. La optimización no se cancela hasta que no se cumple: • Setpoint > CurrentSetpoint + CancelTuningLevel o bien, • Setpoint < CurrentSetpoint - CancelTuningLevel Antes de la optimización fina se realiza una copia de seguridad de los parámetros PID. Estos pueden reactivarse con LoadBackUp. Requisitos para la optimización fina: • El tiempo de posicionamiento del motor está configurado o medido. • La consigna y el valor real se encuentran dentro de los límites configurados. • ManualEnable = FALSE • Reset = FALSE • Modo de operación Automático (State = 3), Inactivo (State = 0) o Manual (State = 4) Al inicio, la optimización fina se desarrolla del modo siguiente: • Modo automático (State = 3) Si desea mejorar los parámetros PID existentes mediante optimización, inicie la optimización fina desde el modo automático. PID_3Step regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se cumplen las condiciones para la optimización fina. Solo entonces comienza la optimización fina. • Modo Inactivo (State = 0) o Manual (State = 4) Si se cumplen las condiciones para optimización inicial, esta se inicia. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Si PIDSelfTune.TIR.RunIn = TRUE, se omite la optimización inicial y se intenta alcanzar la consigna con el valor de salida máximo o mínimo. lo que puede ocasionar una sobreoscilación muy alta. La optimización fina comienza entonces automáticamente. Una vez realizada correctamente la optimización fina, el regulador pasa al modo automático; si la optimización fina no se ha realizado correctamente, el cambio del modo de operación depende de ActivateRecoverMode y ErrorBehaviour. La fase de optimización fina se muestra con la variable TIR.State.
3	Modo automático En el modo automático, PID_3Step regula el sistema regulado conforme a los parámetros predefinidos. Cuando se cumple uno de los requisitos siguientes, se cambia al modo automático: • Optimización inicial finalizada correctamente. • Optimización fina finalizada correctamente. • Cambio del parámetro de entrada/salida Mode al valor 3 y un flanco ascendente en ModeActivate. El cambio de modo automático a modo manual se realiza sin discontinuidad únicamente en el editor de puesta de servicio. En el modo automático se tiene en cuenta la variable ActivateRecoverMode.

State	Descripción del modo de operación
4	Modo manual En el modo manual, se especifican valores de salida manuales en los parámetros Manual_UP y Manual_DN o ManualValue. El parámetro ErrorBits describe si el actuador se puede desplazar al valor de salida en caso de error. Este modo puede activarse también mediante ManualEnable = TRUE. Se recomienda cambiar los modos de operación solo mediante Mode y ModeActivate. El cambio de modo manual a modo automático se efectúa sin discontinuidad. El modo manual es posible aunque haya un error pendiente.
5	Aproximar al valor de salida sustitutivo Este modo de operación se activa en caso de error si Errorbehaviour = TRUE y ActivateRecoverMode = FALSE.. PID_3Step desplaza el elemento final de control al valor de salida sustitutivo y cambia después al modo de operación "Inactivo".
6	Medición del tiempo de posicionamiento Se mide el tiempo que requiere el motor para abrir completamente la válvula a partir del estado cerrado. Este modo de operación se activa cuando se ajusta Mode = 6 y ModeActivate = TRUE. Si se utilizan señales de tope para la medición del tiempo de posicionamiento, la válvula se abre completamente desde la posición actual, luego se cierra completamente y se vuelve a abrir completamente. Si GetTransitTime.InvertDirection = TRUE, el comportamiento se invierte. Si se utiliza una realimentación de posición para la medición del tiempo de posicionamiento, el elemento final de control se desplaza desde la posición actual a una posición de destino. Los límites del valor de salida no se tienen en cuenta durante la medición del tiempo de posicionamiento. El actuador puede desplazarse hasta el tope superior o inferior.
7	Monitorización de errores El algoritmo de regulación está desconectado y ya no modifica la posición de la válvula. Este modo de operación se activa en caso de error en lugar del modo de operación "Inactivo". Deben haberse cumplido todas las condiciones siguientes: • Modo automático (Mode = 3) • Errorbehaviour = FALSE • ActivateRecoverMode = TRUE • Se han producido uno o varios errores en los que interviene ActivateRecoverMode (Página 5209). En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático.
8	Aproximar al valor de salida sustitutivo con monitorización de errores Este modo de operación se activa en caso de error en lugar del modo de operación "Aproximar al valor de salida sustitutivo". PID_3Step desplaza el elemento final de control o actuador al valor de salida sustitutivo y cambia después al modo de operación "Monitorización de errores". Deben haberse cumplido todas las condiciones siguientes: • Modo automático (Mode = 3) • Errorbehaviour = TRUE • ActivateRecoverMode = TRUE • Se han producido uno o varios errores en los que interviene ActivateRecoverMode (Página 5209). En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático.
10	Modo manual sin señales de tope Las señales de tope **no** se tienen en cuenta a pesar de que Config.ActuatorEndStopOn = TRUE. No se tienen en cuenta los límites del valor de salida. Por lo demás, PID_3Step se comporta del mismo modo que en el modo manual.

Comportamiento ENO

Si State = 0, entonces ENO = FALSE.

Si State ≠ 0, entonces ENO = TRUE.

Cambio automático de modo de operación durante la puesta en servicio

Una vez realizada correctamente la optimización inicial o la optimización fina, se activa el modo automático. La siguiente tabla muestra cómo se modifican Mode y State durante una optimización inicial correcta.

N.º de ciclo	Mode	State	Acción
0	4	4	Ajustar Mode = 1
1	1	4	Ajustar ModeActivate = TRUE
1	4	1	El valor de State se guarda en Mode Se inicia la optimización inicial
n	4	1	Optimización inicial correcta
n	3	3	Se inicia el modo automático

En caso de error, PID_3Step cambia automáticamente el modo de operación. La siguiente tabla muestra cómo se modifican Mode y State durante una optimización inicial incorrecta.

N.º de ciclo	Mode	State	Acción
0	4	4	Ajustar Mode = 1
1	1	4	Ajustar ModeActivate = TRUE
1	4	1	El valor de State se guarda en Mode Se inicia la optimización inicial
n	4	1	La optimización inicial se ha cancelado
n	4	4	Se inicia el modo manual

Si ActivateRecoverMode = TRUE, se activa el modo guardado en Mode. Al iniciar la medición del tiempo de posicionamiento, la optimización inicial o la optimización fina, PID_3Step ha guardado el valor de State en el parámetro de entrada/salida Mode. Por lo tanto, PID_3Step cambia al modo desde el que se inició la optimización.

Si ActivateRecoverMode = FALSE, se activa el modo "Inactivo" o "Aproximar al valor de salida sustitutivo".

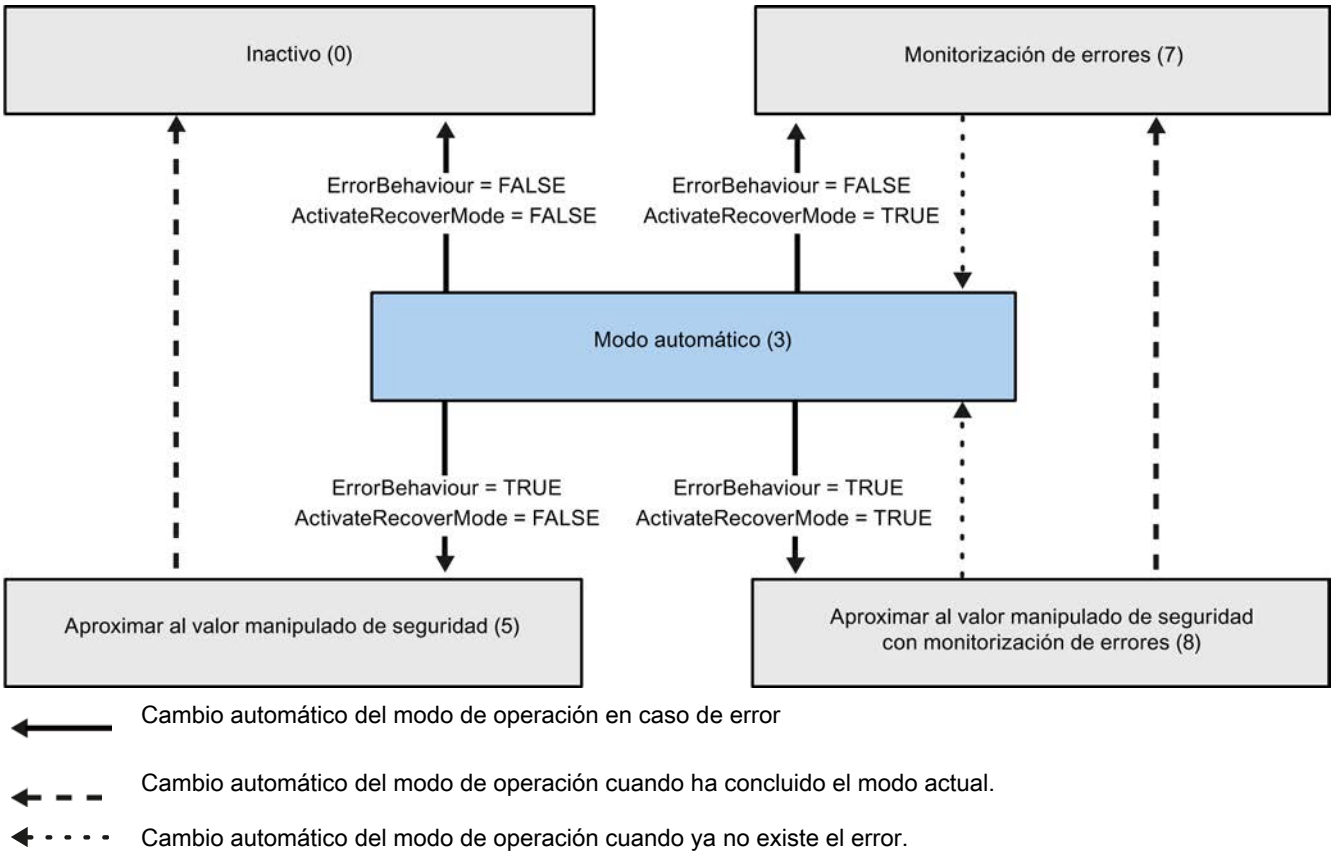
Cambio automático del modo de operación tras la medición del tiempo de posicionamiento

Si ActivateRecoverMode = TRUE, se activa el modo de operación guardado en Mode una vez realizada correctamente la medición del tiempo de posicionamiento.

Si ActivateRecoverMode = FALSE, se cambia al modo de operación "Inactivo" una vez realizada correctamente la medición del tiempo de posicionamiento.

Cambio automático del modo de operación en el modo automático

En caso de error, PID_3Step cambia automáticamente el modo de operación. El diagrama siguiente muestra la influencia de ErrorBehaviour y ActivateRecoverMode sobre este cambio del modo de operación.



Consulte también

Variable ActivateRecoverMode V2 (Página 5209)

Parámetro ErrorBits V2 (Página 5206)

Parámetro ErrorBits V2

Si hay varios errores presentes a la vez, los valores del ErrorBits se muestran sumados binariamente. Si se muestra, p. ej., ErrorBits = 0003h, significa que están presentes simultáneamente los errores 0001h y 0002h.

Si hay una realimentación de posición, PID_3Step utiliza ManualValue en el modo manual como valor de salida. La excepción es Errorbits = 10000h.

ErrorBits (DW#16#...)	Descripción
0000	No hay ningún error.
0001	El parámetro "Input" se encuentra fuera de los límites del valor real. • Input > Config.InputUpperLimit o bien, • Input < Config.InputLowerLimit Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_3Step permanece en modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
0002	Valor no válido en el parámetro "Input_PER". Compruebe si hay un error en la entrada analógica. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_3Step cambia al modo de operación "Aproximar al valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores". En cuanto el error deja de estar pendiente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
0004	Error durante la optimización fina. No se ha podido mantener la oscilación del valor real. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_3Step cancela la optimización y cambia al modo guardado en Mode.
0010	Se ha cambiado la consigna durante la optimización. En la variable CancelTuningLevel puede ajustar la fluctuación admisible de la consigna. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_3Step cancela la optimización y cambia al modo guardado en Mode.
0020	La optimización inicial no está permitida durante la optimización fina. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_3Step permanece en el modo optimización fina.
0080	Error durante la optimización inicial. Los límites del valor de salida no están configurados correctamente. Compruebe si los límites del valor de salida están configurados correctamente y si encajan con el sentido de la regulación. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_3Step cancela la optimización y cambia al modo guardado en Mode.
0100	Un error en la optimización fina ha provocado parámetros no válidos. Si antes de producirse el error se cumplía ActivateRecoverMode = TRUE, PID_3Step cancela la optimización y cambia al modo guardado en Mode.
0200	Valor no válido en el parámetro "Input": El valor no tiene un formato numérico válido. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_3Step cambia al modo de operación "Aproximar al valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores". En cuanto el error deja de estar pendiente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.

ErrorBits (DW#16#...)	Descripción
0400	Error al calcular el valor de salida. Compruebe los parámetros PID. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_3Step cambia al modo de operación "Aproximar al valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores". En cuanto el error deja de estar pendiente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
0800	Error de tiempo de muestreo: PID_3Step no se activa dentro del tiempo de muestreo del OB de alarma cíclica. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_3Step permanece en modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
1000	Valor no válido en el parámetro "Setpoint": El valor no tiene un formato numérico válido. Si antes de producirse el error estaba activo el modo automático y se cumple ActivateRecoverMode = TRUE, PID_3Step cambia al modo de operación "Aproximar al valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores". En cuanto el error deja de estar pendiente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
2000	Valor no válido en el parámetro Feedback_PER. Compruebe si hay un error en la entrada analógica. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual. En el modo manual solo es posible modificar la posición del actuador con Manual_UP y Manual_DN, no con ManualValue. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
4000	Valor no válido en el parámetro Feedback. El valor no tiene un formato numérico válido. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual. En el modo manual solo es posible modificar la posición del actuador con Manual_UP y Manual_DN, no con ManualValue. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.

ErrorBits (DW#16#...)	Descripción
8000	Error en la realimentación digital de posición. Actuator_H = TRUE y Actuator_L = TRUE. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual. El modo manual no es posible en este estado. Para poder sacar el actuador de ese estado, es necesario desactivar las "Señales de tope del actuador" (Config.ActuatorEndStopOn = FALSE) o pasar al "Modo manual sin señales de tope" (Mode = 10). Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático. Si antes de producirse el error estaba activo el modo optimización inicial, optimización fina o medición del tiempo de posicionamiento y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode.
10000	Valor no válido en el parámetro ManualValue. El valor no tiene un formato numérico válido. El actuador no puede desplazarse al valor manual y permanece en la posición actual. Especifique un valor válido en ManualValue o desplace el actuador en modo manual con Manual_UP y Manual_DN.
20000	Valor no válido en la variable SavePosition. El valor no tiene un formato numérico válido. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual.
40000	Valor no válido en el parámetro Disturbance. El valor no tiene un formato numérico válido. Si antes de producirse el error estaba activo el modo automático y ActivateRecoverMode = TRUE, Disturbance se ajusta a cero. PID_3Step permanece en modo automático. Si antes de producirse el error estaba activo el modo optimización inicial u optimización fina y se cumplía ActivateRecoverMode = TRUE, PID_3Step pasa al modo guardado en Mode. Si en la fase actual Disturbance no influye en el valor de salida, la optimización no se cancela. Durante la medición del tiempo de posicionamiento el error no tiene ninguna influencia.

Variable ActivateRecoverMode V2

La variable ActivateRecoverMode determina el comportamiento en caso de error. El parámetro Error indica si hay algún un error pendiente en ese momento. Si no persiste el error, se indica Error = FALSE. El parámetro ErrorBits indica qué errores han ocurrido.

ATENCIÓN
Su instalación puede sufrir daños. Si se cumple ActivateRecoverMode = TRUE, incluso al rebasar los límites del valor real PID_3Step permanece en el modo automático. De este modo, su instalación puede sufrir daños. Configure un comportamiento en caso de error para su sistema regulado que proteja su instalación de daños.

Modo automático

ActivateRecover Mode	Descripción
FALSE	En caso de error, PID_3Step pasa al modo de operación "Inactivo" o "Aproximar al valor de salida sustitutivo". El regulador no se activa hasta que no se detecta un flanco descendente en Reset o un flanco ascendente en ModeActivate.
TRUE	Si en el modo automático se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación, ya que PID_3Step alterna entre el valor de salida calculado y el valor de salida sustitutivo cada vez que se produce un error. Compruebe en tal caso el parámetro ErrorBits y elimine la causa de error. Si se producen uno o varios de los errores siguientes, PID_3Step permanece en modo automático: • 0001h: El parámetro "Input" se encuentra fuera de los límites del valor real. • 0800h: Error en tiempo de muestreo • 40000h: Valor no válido en el parámetro Disturbance. Si se produce uno o varios de los errores siguientes, PID_3Step pasa al modo operativo "Aproximar a valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores": • 0002h: Valor no válido en el parámetro Input_PER. • 0200h: Valor no válido en el parámetro Input. • 0400h: Error al calcular el valor de salida. • 1000h: Valor no válido en el parámetro Setpoint. Si se producen uno o varios de los siguientes errores, PID_3Step ya no puede mover el actuador: • 2000h: Valor no válido en el parámetro Feedback_PER. • 4000h: Valor no válido en el parámetro Feedback. • 8000h: Error en la realimentación digital de posición. • 20000h: Valor no válido en la variable SavePosition. El valor no tiene un formato numérico válido. Este comportamiento no depende de ErrorBehaviour. En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático.

Optimización inicial, optimización fina y medición del tiempo de posicionamiento

ActivateRecover Mode	Descripción
FALSE	En caso de error, PID_3Step pasa al modo de operación "Inactivo" o "Aproximar al valor de salida sustitutivo". El regulador no se activa hasta que no se detecta un flanco descendente en Reset o un flanco ascendente en ModeActivate. Una vez realizada correctamente la medición del tiempo de posicionamiento, el regulador pasa al modo de operación Inactivo.
TRUE	Si se produce el siguiente error, PID_3Step permanece en el modo activo: • 0020h: La optimización inicial no está permitida durante la optimización fina. Los siguientes errores se omiten: • 10000h: Valor no válido en el parámetro ManualValue. • 20000h: Valor no válido en la variable SavePosition. Con todos los demás errores, PID_3Step interrumpe la optimización y cambia al modo desde el que se inició la optimización.

Modo manual

En el modo manual, ActivateRecoverMode no tiene efecto.

Consulte también

Variables estáticas PID_3Step V2 (Página 5192)

Parámetros State y Mode V2 (Página 5201)

Variable Warning V2

Si hay varias advertencias presentes a la vez, los valores de las advertencias se muestran sumados binariamente. Si se muestra p. ej. la advertencia 0005h, significa que las advertencias 0001h y 0004h están presentes simultáneamente.

Warning (DW#16#...)	Descripción
0000	No hay advertencias pendientes.
0001	Durante la optimización inicial no se encontró el punto de inflexión.
0004	La consigna ha sido limitada a los límites ajustados.
0008	En el método de cálculo seleccionado no se han definido todas las propiedades necesarias del sistema regulado. En sustitución se calcularon los parámetros PID con el método TIR.TuneRule = 3.
0010	No se ha podido cambiar el modo de operación porque Reset = TRUE o ManualEnable = TRUE.
0020	El tiempo de muestreo del algoritmo PID está limitado por el tiempo de ciclo del OB invocante. Para obtener mejores resultados, utilice tiempos de ciclo más cortos para el OB.
0040	El valor real ha rebasado por exceso uno de sus límites de advertencia.
0080	Valor no válido en Mode. El modo de operación no se conmuta.
0100	El valor manual ha sido limitado a los límites de la salida del regulador.
0200	No se admiten las reglas indicadas para la optimización. No se calculan parámetros PID.
0400	El tiempo de posicionamiento no se puede medir porque los ajustes del elemento final de control no concuerdan con el método de medición seleccionado.
0800	En la medición del tiempo de posicionamiento, la diferencia entre la posición actual y el nuevo valor de salida es insuficiente. Esto puede ocasionar resultados erróneos. La diferencia entre el valor de salida actual y el nuevo valor de salida debe ser de al menos un 50% del rango de posicionamiento total.
1000	El valor de salida sustitutivo no puede alcanzarse porque está fuera de los límites del valor de salida.
2000	El actuador se mueve en un sentido durante un tiempo superior a Config.VirtualActuatorLimit × Retain.TransitTime. Compruebe si el actuador ha alcanzado una señal de tope.

Las siguientes advertencias se eliminan en cuanto la causa se soluciona:

- 0001h
- 0004h
- 0008h
- 0040h
- 0100h
- 2000h

Todas las demás advertencias se eliminan si hay un flanco ascendente en Reset o ErrorAck.

PID_3Step V1

Descripción PID_3Step V1

Descripción

Con la instrucción PID_3Step se puede configurar un regulador PID con autooptimización para válvulas o elementos de control final con comportamiento integrador.

Se dispone de los modos de operación siguientes:

- Inactivo
- Optimización inicial
- Optimización fina
- Modo automático
- Modo manual
- Aproximar al valor de salida sustitutivo
- Medición del tiempo de posicionamiento
- Aproximar al valor de salida sustitutivo con monitorización de errores
- Monitorización de errores

Los modos de operación se describen detalladamente en el parámetro State.

Algoritmo PID

PID_3Step es un regulador PIDT1 con Anti-Windup y ponderación de la acción P y D. El valor de salida se calcula con la siguiente fórmula.

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
y	Valor de salida
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
a	Coeficiente para el retardo de la acción derivada (T1 = a x T_D)
T_D	Tiempo derivativo
c	Ponderación de la acción D

Diagrama de bloques sin realimentación de posición

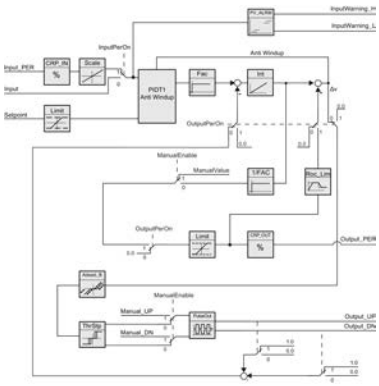


Diagrama de bloques con realimentación de posición

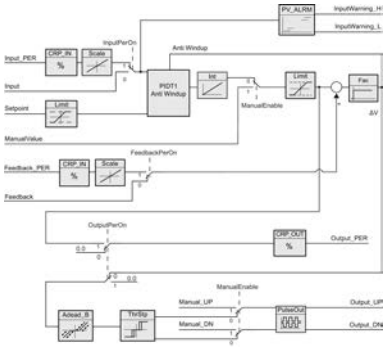
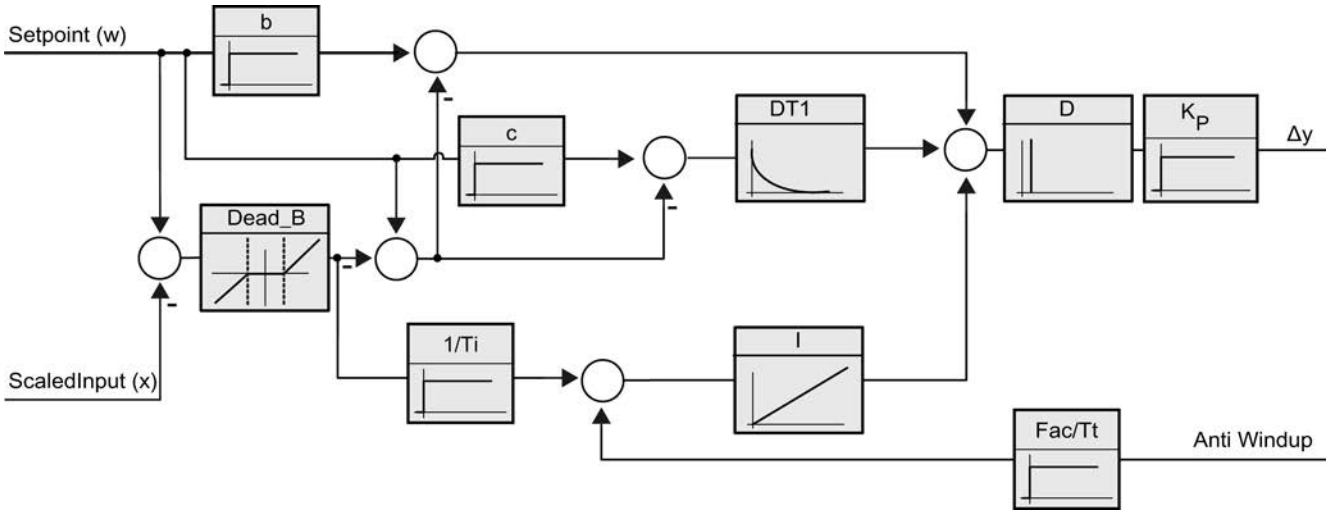


Diagrama de bloques PIDT1 con Anti-Windup



Llamada

PID_3Step se llama en un periodo constante del tiempo de ciclo del OB invocante (preferentemente en un OB de alarma cíclica).

Cargar en dispositivo

Los valores actuales de variables remanentes solo se actualizan si se carga PID_3Step por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Arranque

PID_3Step arranca simultáneamente con la CPU en el último modo de operación activo. Para dejar PID_3Step en el modo de operación "Inactivo", ajuste RunModeByStartup = FALSE.

Comportamiento en caso de error

Cuando ocurren errores, estos se indican en el parámetro Error. El comportamiento de PID_3Step se configura con las variables ErrorBehaviour y ActivateRecoverMode.

ErrorBehaviour	ActivateRecoverMode	Configuración del ajuste del actuador Poner Output a	Comportamiento
0	FALSE	Valor de salida actual	Cambio al modo de operación "Inactivo" (Mode = 0)
0	TRUE	Valor de salida actual por la duración del error	Cambio al modo de operación "Monitorización de errores" (Mode = 7)
1	FALSE	Valor de salida sustitutivo	Cambio al modo de operación "Aproximar al valor de salida sustitutivo" (Mode = 5) Cambio al modo de operación "Inactivo" (Mode = 0)
1	TRUE	Valor de salida sustitutivo mientras dure el error	Cambio al modo de operación "Aproximar a valor de salida sustitutivo con monitorización de errores" (Mode = 8) Cambio al modo de operación "Monitorización de errores" (Mode = 7)

El parámetro ErrorBits indica qué errores han ocurrido.

Consulte también

Parámetros State y Retain.Mode V1 (Página 5229)

Parámetro ErrorBits V1 (Página 5237)

Configuración de PID_3Step V1 (Página 10016)

Funcionamiento PID_3Step V1

Monitorizar límites del valor real

En las variables Config.InputUpperLimit y Config.InputLowerLimit es posible definir un límite superior y uno inferior del valor real. Si el valor real se encuentra fuera de estos límites, ocurre un error (ErrorBits = 0001hex).

En las variables Config.InputUpperWarning y Config.InputLowerWarning se definen un límite de advertencia superior y uno inferior del valor real. Si el valor real se encuentra fuera de estos límites de advertencia, se produce una advertencia (Warnings = 0040hex) y el parámetro de salida InputWarning_H o InputWarning_L adopta el valor TRUE.

Limitar consigna

En las variables Config.SetpointUpperLimit y Config.SetpointLowerLimit es posible definir un límite superior y uno inferior para la consigna. PID_3Step limita la consigna automáticamente a los límites del valor real. Es posible limitar la consigna a un rango más pequeño. PID_3Step

verifica si dicho rango está dentro de los límites del valor real. Si la consigna está fuera de estos límites, se utilizan los límites superior e inferior como consigna y el parámetro de salida SetpointLimit_H o SetpointLimit_L adopta el valor TRUE.

La consigna se limita en todos los modos de operación.

Limitar el valor de salida

En las variables Config.OutputUpperLimit y Config.OutputLowerLimit se define un límite superior y uno inferior del valor de salida. Los límites del valor de salida deben encontrarse dentro del "Tope inferior" o el "Tope superior".

- Tope superior: Config.FeedbackScaling.UpperPointOut
- Tope inferior: Config.FeedbackScaling.LowerPointOut

Rige:

UpperPointOut ≥ OutputUpperLimit > OutputLowerLimit ≥ LowerPointOut

Los valores válidos para el "Tope superior" y el "Tope inferior" dependen de:

- FeedbackOn
- FeedbackPerOn
- OutputPerOn

OutputPerOn	FeedbackOn	FeedbackPerOn	LowerPointOut	UpperPointOut
FALSE	FALSE	FALSE	No ajustable (0.0 %)	No ajustable (100.0 %)
FALSE	TRUE	FALSE	-100.0 % o 0.0 %	0.0 % o +100.0 %
FALSE	TRUE	TRUE	-100.0 % o 0.0 %	0.0 % o +100.0 %
TRUE	FALSE	FALSE	No ajustable (100.0 %)	No ajustable (100.0 %)
TRUE	TRUE	FALSE	-100.0 % o 0.0 %	0.0 % o +100.0 %
TRUE	TRUE	TRUE	-100.0 % o 0.0 %	0.0 % o +100.0 %

Si OutputPerOn = FALSE y FeedbackOn = FALSE, no es posible limitar el valor de salida. Las salidas digitales deben desactivarse en Actuator_H = TRUE o Actuator_L = TRUE o después de un tiempo de desplazamiento del 110% del tiempo de posicionamiento del motor.

El valor de salida asciende a 27648 con el 100 % y a -27648 con un -100 %. PID_3Step debe poder cerrar la válvula por completo. Por lo tanto el cero debe estar contenido en los límites del valor de salida.

Valor de salida sustitutivo

En caso de error, PID_3Step puede emitir un valor de salida sustitutivo y conducir el elemento final de control a una posición segura que se especifica en la variable SavePosition. El valor de salida sustitutivo debe encontrarse dentro de los límites del valor de salida.

Vigilar la validez de las señales

Se vigila la validez de los valores de los siguientes parámetros:

- Setpoint
- Input
- Input_PER
- Feedback
- Feedback_PER
- Output

Vigilar el tiempo de muestreo PID_3Step

En el caso ideal, el tiempo de muestreo equivale al tiempo de ciclo del OB invocante. La instrucción PID_3Step mide en cada caso el tiempo que transcurre entre dos llamadas. Este es el tiempo de muestreo actual. Con cada cambio de modo de operación y en el primer arranque se calcula la media de los 10 primeros tiempos de muestreo. Si el tiempo de muestreo actual se desvía mucho de este valor medio, ocurre un error (ErrorBits = 0800 hex).

Durante la optimización PID_3Step cambia al modo de operación "Inactivo" cuando se dan las siguientes condiciones:

- Valor medio nuevo >= 1,1 veces el valor medio antiguo
- Valor medio nuevo <= 0,9 veces el valor medio antiguo

PID_3Step cambia al modo de operación "Inactivo" cuando se dan las siguientes condiciones:

- Valor medio nuevo >= 1,5 veces el valor medio antiguo
- Valor medio nuevo <= 0,5 veces el valor medio antiguo

Tiempo de muestreo del algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de ciclo. Todas las demás funciones de PID_3Step se ejecutan con cada llamada.

Medir el tiempo de posicionamiento del motor

El tiempo de posicionamiento del motor es el tiempo en segundos que requiere el motor para desplazar el elemento final de control del estado cerrado al estado abierto. El elemento final de control se mueve en un sentido a como máximo el 110% del tiempo de posicionamiento del motor. PID_3Step necesita un tiempo de posicionamiento del motor lo más preciso posible para alcanzar un buen resultado de regulación. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento. El tiempo de posicionamiento del motor puede medirse durante la puesta en servicio. Los límites del valor de salida no se tienen en cuenta durante la medición del tiempo de posicionamiento del motor. El actuador puede desplazarse hasta el tope superior o inferior.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal. Para refrigeración o para las regulaciones de desagües puede ser necesario invertir el sentido de regulación. PID_3Step no funciona con ganancia proporcional negativa. Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida. El sentido de regulación se tiene en cuenta también durante la optimización inicial y la optimización fina.

Consulte también

Configuración de PID_3Step V1 (Página 10016)

Parámetros de entrada PID_3Step V1

Tabla 9-91

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
Setpoint	REAL	0.0	Consigna del regulador PID en modo automático
Input	REAL	0.0	Una variable del programa de usuario se utiliza como origen del valor real. Si utiliza el parámetro Input, debe cumplirse lo siguiente: Config.InputPerOn = FALSE.
Input_PER	WORD	W#16#0	Una entrada analógica se utiliza como origen del valor real. Si utiliza el parámetro Input_PER, debe cumplirse lo siguiente: Config.InputPerOn = TRUE.
Actuator_H	BOOL	FALSE	Realimentación de posicionamiento digital de la válvula para el tope superior Si Actuator_H = TRUE la posición de la válvula se encuentra en el tope superior y la válvula ya no se desplaza en ese sentido.
Actuator_L	BOOL	FALSE	Realimentación de posición digital de la válvula para el tope inferior Si Actuator_L = TRUE la posición de la válvula se encuentra en el tope inferior y la válvula ya no se desplaza en ese sentido.
Feedback	REAL	0.0	Realimentación de posición de la válvula Si utiliza el parámetro Feedback, debe cumplirse lo siguiente: Config.FeedbackPerOn = FALSE.
Feedback_PER	WORD	W#16#0	Realimentación de posición analógica de una válvula Si utiliza el parámetro Feedback_PER, debe cumplirse lo siguiente: Config.FeedbackPerOn = TRUE. Feedback_PER se escala a partir de las variables: • Config.FeedbackScaling.LowerPointIn • Config.FeedbackScaling.UpperPointIn • Config.FeedbackScaling.LowerPointOut • Config.FeedbackScaling.UpperPointOut

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
ManualEnable	BOOL	FALSE	• El flanco FALSE -> TRUE selecciona el modo de operación "Modo manual", State = 4, Retain.Mode no cambia. • El flanco TRUE -> FALSE selecciona el último modo de operación activo La operación ManualEnable = TRUE no provoca un cambio en Retain.Mode. Hasta que no se llega al flanco TRUE -> FALSE en ManualEnable , no se tiene en cuenta la modificación de Retain.Mode. PID_3Step V1.1Si al arrancar la CPU ManualEnable = TRUE, PID_3Step arranca en modo manual. No es necesario que se produzca un flanco ascendente (FALSE > TRUE) en ManualEnable. PID_3Step V1.0 Al arrancar la CPU, PID_3Step únicamente pasa al modo manual si se detecta un flanco ascendente (FALSE->TRUE) en ManualEnable . Si no se detecta un flanco ascendente, PID_3Step arranca en el último modo de operación en el que ManualEnable era FALSE.
ManualValue	REAL	0.0	En el modo manual se determina la posición absoluta de la válvula. ManualValue sólo se evalúa si se utiliza OutputPer o si se dispone de una realimentación de posición.
Manual_UP	BOOL	FALSE	En el modo manual cada flanco ascendente abre la válvula un 5 % del rango de posicionamiento total o durante el tiempo mínimo de posicionamiento del motor. Manual_UP sólo se evalúa si ni se utiliza Output_PER ni hay disponible una realimentación de posición.
Manual_DN	BOOL	FALSE	En el modo manual cada flanco ascendente cierra la válvula un 5 % del rango de posicionamiento total o durante el tiempo mínimo de posicionamiento del motor. Manual_DN sólo se evalúa si ni se utiliza Output_PER ni hay disponible una realimentación de posición.
Reset	BOOL	FALSE	Realiza un rearranque completo del regulador. • Flanco FALSE -> TRUE – Cambio al modo de operación "Inactivo" – Los valores intermedios de la regulación se desactivan (los parámetros PID se mantienen) • Flanco TRUE -> FALSE Cambio al último modo de operación activo

Parámetros de salida PID_3Step V1

Tabla 9-92

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
ScaledInput	REAL	0.0	Valor real escalado
ScaledFeedback	REAL	0.0	Realimentación de posición escalada Con actuadores sin realimentación de posición, ScaledFeedback indica la posición del actuador de modo muy impreciso. En tal caso, ScaledFeedback solo puede usarse para una estimación aproximada de la posición actual.
Output_UP	BOOL	FALSE	Valor de salida digital para abrir la válvula Si Config.OutputPerOn = FALSE, se utiliza el parámetro Output_UP.
Output_DN	BOOL	FALSE	Valor de salida digital para cerrar la válvula Si Config.OutputPerOn = FALSE, se utiliza el parámetro Output_DN.
Output_PER	WORD	W#16#0	Valor de salida analógico Si Config.OutputPerOn = TRUE, se utiliza Output_PER.
SetpointLimit_H	BOOL	FALSE	Cuando SetpointLimit_H = TRUE, significa que se ha alcanzado el límite superior absoluto de la consigna. En la CPU la consigna se limita al límite superior absoluto configurado para la consigna. Como límite superior de la consigna se ajusta de forma predeterminada el límite superior absoluto configurado para el valor real. Si se configura Config.SetpointUpperLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de la consigna.
SetpointLimit_L	BOOL	FALSE	Cuando SetpointLimit_L = TRUE, significa que se ha alcanzado el límite inferior absoluto de la consigna. En la CPU la consigna se limita al límite inferior absoluto configurado para la consigna. Como límite inferior de la consigna se ajusta de forma predeterminada el límite inferior absoluto configurado para el valor real. Si se configura Config.SetpointLowerLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de la consigna.
InputWarning_H	BOOL	FALSE	Cuando InputWarning_H = TRUE, significa que se ha alcanzado o rebasado el límite superior de advertencia del valor real.
InputWarning_L	BOOL	FALSE	Cuando InputWarning_L = TRUE, significa que se ha alcanzado o rebasado por defecto el límite inferior de advertencia del valor real.

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
State	INT	0	El parámetro State (Página 5229) indica el modo de operación actual del regulador PID. El modo de operación se cambia con la variable Retain.Mode. • State = 0: Inactivo • State = 1: Optimización inicial • State = 2: Optimización fina • State = 3: Modo automático • State = 4: Modo manual • State = 5: Aproximar al valor de salida sustitutivo • State = 6: Medición del tiempo de posicionamiento • State = 7: monitorización de errores • State = 8: Aproximar al valor de salida sustitutivo con monitorización de errores
Error	BOOL	FALSE	Si Error = TRUE, existe al menos un aviso de error.
ErrorBits	DWORD	DW#16#0	El parámetro ErrorBits (Página 5237) muestra los avisos de error.

Consulte también

Parámetros State y Retain.Mode V1 (Página 5229)

Parámetro ErrorBits V1 (Página 5237)

Variables estáticas PID_3Step V1

No se deben modificar las variables que no aparecen listadas. Estas solo se utilizan internamente.

Tabla 9-93

Variable	Tipo de datos	Ajuste predeter minado	Descripción
ActivateRecoverMode	BOOL	TRUE	La variable ActivateRecoverMode (Página 5239) determina el comportamiento en caso de error.
RunModeByStartup	BOOL	TRUE	Activar último modo de operación después del rearranque completo de la CPU Si RunModeByStartup = TRUE, el regulador regresa al último modo de operación activo tras un arranque de la CPU. Si RunModeByStartup = FALSE, el regulador permanece inactivo tras un arranque de la CPU.
PhysicalUnit	INT	0	Unidad física del valor real y de la consigna, p. ej. ºC o ºF.
PhysicalQuantity	INT	0	Unidad física del valor real y de la consigna, p. ej. temperatura

Variable	Tipo de datos	Ajuste predeter minado	Descripción
ErrorBehaviour	INT	0	Si ErrorBehaviour = 0, en caso de error la válvula permanece en la posición actual y el regulador cambia directamente al modo de operación "Inactivo" o "Monitorización de errores". Si ErrorBehaviour = 1, en caso de error el actuador se desplaza al valor de salida sustitutivo y solo entonces pasa al modo de operación "Inactivo" o "Monitorización de errores". Si se producen los errores siguientes, la válvula ya no se puede desplazar a un valor de salida sustitutivo configurado. • 2000h: Valor no válido en el parámetro Feedback_PER. • 4000h: Valor no válido en el parámetro Feedback. • 8000h: Error en la realimentación digital de posición.
Warning	DWORD	DW#16#0	La variable Warning (Página 5229) muestra las advertencias desde Reset o desde el último cambio de modo de operación. Las advertencias cíclicas (por ejemplo, advertencia del valor real) se muestran mientras sigue existiendo la causa de la advertencia. Una vez solucionada la causa, se borran automáticamente. Las advertencias no cíclicas (como es la relativa al punto de inflexión no encontrado) se mantienen y se borran como se hace con los errores.
SavePosition	REAL	0.0	Valor de salida sustitutivo Si ErrorBehaviour = 1, en caso de error el elemento final de control se desplaza a una posición segura para la instalación y solo entonces se cambia al modo de operación "Inactivo".
CurrentSetpoint	REAL	0.0	Consigna activa actualmente. Este valor se congela al iniciarse la optimización.
Progress	REAL	0.0	Progreso de la optimización en porcentaje (0.0 - 100.0)
Config.InputPerOn	BOOL	TRUE	Si InputPerOn = TRUE, se utiliza el parámetro Input_PER. Si InputPerOn = FALSE, se utiliza el parámetro Input.
Config.OutputPerOn	BOOL	FALSE	Si OutputPerOn = TRUE, se utiliza el parámetro Output_PER. Si OutputPerOn = FALSE, se utilizan los parámetros Ouput_UP y Output_DN.
Config.LoadBackUp	BOOL	FALSE	Si LoadBackUp = TRUE, se carga nuevamente el último juego de parámetros PID. El juego se guardó antes de la última optimización. LoadBackUp se ajusta automáticamente de nuevo a FALSE.
Config.InvertControl	BOOL	FALSE	Inversión del sentido de regulación Si InvertControl = TRUE, un error de regulación ascendente provocará una reducción del valor de salida.
Config.FeedbackOn	BOOL	FALSE	Si FeedbackOn = FALSE, se simula una realimentación de posición. Si FeedbackOn = TRUE, se activa por lo general la realimentación de posición.
Config.FeedbackPerOn	BOOL	FALSE	FeedbackPerOn solo está activo cuando FeedbackOn = TRUE. Si FeedbackPerOn = TRUE, se utiliza la entrada analógica para la realimentación de posición (parámetro Feedback_PER). Si FeedbackPerOn = FALSE, se utiliza el parámetro Feedback para la realimentación de posición.
Config.ActuatorEndStopOn	BOOL	FALSE	Si ActuatorEndStopOn = TRUE, se tiene en cuenta la realimentación de posición digital Actuator_L y Actuator_H .

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Config.InputUpperLimit	REAL	120.0	Límite superior del valor real En la entrada de periferia, el valor real puede encontrarse como máximo un 18% por encima del rango normalizado (margen de saturación). No se comunica ningún error más debido al rebase por exceso del "Límite superior del valor real". Únicamente se detectan la rotura de hilo y el cortocircuito y PID_3Step se comporta según la reacción configurada en caso de error. InputUpperLimit > InputLowerLimit
Config.InputLowerLimit	REAL	0.0	Límite inferior del valor real InputLowerLimit < InputUpperLimit
Config.InputUpperWarning	REAL	+3.402822e+38	Límite superior de advertencia del valor real Si se configura InputUpperWarning a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se utiliza como límite superior de advertencia. Si se configura InputUpperWarning a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de advertencia. InputUpperWarning > InputLowerWarning InputUpperWarning ≤ InputUpperLimit
Config.InputLowerWarning	REAL	-3.402822e+38	Límite inferior de advertencia del valor real Si se configura InputLowerWarning a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se utiliza como límite inferior de advertencia. Si se configura InputLowerWarning a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de advertencia. InputLowerWarning < InputUpperWarning InputLowerWarning ≥ InputLowerLimit
Config.OutputUpperLimit	REAL	100.0	Límite superior del valor de salida Para más detalles ver OutputLowerLimit
Config.OutputLowerLimit	REAL	0.0	Límite inferior del valor de salida Si OutputPerOn = TRUE o FeedbackOn = TRUE, el rango de valores válido va de -100 hasta +100 % incluyendo el cero. -100 % corresponde a Output = -27648; +100 % corresponde a Output = 27648 Si OutputPerOn = FALSE, el rango de valores válido va de 0 hasta 100 %. En 0 % la válvula está completamente cerrada y en 100 % completamente abierta.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Config.SetpointUpperLimit	REAL	+3.402822e+38	Límite superior de la consigna Si se configura SetpointUpperLimit a un valor que se encuentre fuera de los límites del valor real, el límite superior absoluto configurado para el valor real se preajusta como límite superior de la consigna. Si se configura SetpointUpperLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite superior de la consigna.
Config.SetpointLowerLimit	REAL	-3.402822e+38	Límite inferior de la consigna Si se configura SetpointLowerLimit a un valor que se encuentre fuera de los límites del valor real, el límite inferior absoluto configurado para el valor real se preajusta como límite inferior de la consigna. Si se configura SetpointLowerLimit a un valor que se encuentre dentro de los límites del valor real, este valor se utiliza como límite inferior de la consigna.
Config.MinimumOnTime	REAL	0.0	Tiempo conexión mín. Tiempo en segundos, que debe estar conectado como mínimo el elemento final de control.
Config.MinimumOffTime	REAL	0.0	Tiempo OFF mín. Tiempo en segundos, que debe estar desconectado como mínimo el elemento final de control.
Config.TransitTime	REAL	30.0	Tiempo de posicionamiento del motor Tiempo en segundos que requiere el elemento final de control para desplazar la válvula del estado cerrado al estado abierto.
Config.InputScaling.UpperPointIn	REAL	27648.0	Escalado Input_PER arriba Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.InputScaling.LowerPointIn	REAL	0.0	Escalado Input_PER abajo Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.InputScaling.UpperPointOut	REAL	100.0	Valor real superior escalado Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.InputScaling.LowerPointOut	REAL	0.0	Valor real inferior escalado Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura InputScaling, Input_PER se convierte en porcentaje.
Config.FeedbackScaling.UpperPointIn	REAL	27648.0	Escalado Feedback_PER arriba Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Config.FeedbackScaling.LowerPointIn	REAL	0.0	Escalado Feedback_PER abajo Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
Config.FeedbackScaling.UpperPointOut	REAL	100.0	Tope superior Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
Config.FeedbackScaling.LowerPointOut	REAL	0.0	Tope inferior Mediante los dos pares de valores UpperPointOut, UpperPointIn y LowerPointOut, LowerPointIn de la estructura FeedbackScaling, Feedback_PER se convierte en porcentaje.
GetTransitTime.InvertDirection	BOOL	FALSE	Si InvertDirection = FALSE, se abre completamente la válvula para determinar el tiempo de posicionamiento, luego se cierra y se vuelve a abrir. Si InvertDirection = TRUE, se cierra completamente la válvula, luego se abre y se vuelve a cerrar.
GetTransitTime.SelectFeedback	BOOL	FALSE	Si SelectFeedback = TRUE, durante la medición del tiempo de posicionamiento se tendrá en cuenta Feedback_PER o Feedback. Si SelectFeedback = FALSE, durante la medición del tiempo de posicionamiento se tendrán en cuenta Actuator_H y Actuator_L.
GetTransitTime.Start	BOOL	FALSE	Si Start = TRUE, se inicia la medición del tiempo de posicionamiento.
GetTransitTime.State	INT	0	Fase actual de la medición del tiempo de posicionamiento • State = 0: Inactivo • State = 1: Abrir válvula completamente • State = 2: Cerrar válvula completamente • State = 3: Desplazar válvula a la posición de destino (NewOutput) • State = 4: Tiempo de posicionamiento medido correctamente • State = 5: Medición del tiempo de posicionamiento cancelada
GetTransitTime.NewOutput	REAL	0.0	Posición de destino para la medición del tiempo de posicionamiento con realimentación de posición La posición de destino debe encontrarse dentro de los topes superior e inferior. La diferencia entre NewOutput y ScaledFeedback debe estar al menos por encima del 50% del margen de ajuste admisible.
CycleTime.StartEstimation	BOOL	TRUE	Si StartEstimation = TRUE, se inicia la medición del tiempo de muestreo PID_3Step. Una vez finalizada la medición, CycleTime.StartEstimation = FALSE.
CycleTime.EnEstimation	BOOL	TRUE	Si EnEstimation = TRUE, se calcula el tiempo de muestreo PID_3Step.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
CycleTime.EnMonitoring	BOOL	TRUE	Si EnMonitoring = TRUE, se vigila el tiempo de muestreo PID_3Step. Si no es posible ejecutar PID_3Step dentro del tiempo de muestreo, se emite el error 0800h y se cambia el modo de operación. El modo de operación al que se cambie depende de ActivateRecoverMode y de ErrorBehaviour. Si EnMonitoring = FALSE, el tiempo de muestreo PID_3Step no se vigila, el error 0800h no se emite y el modo de operación no se cambia.
CycleTime.Value	REAL	0.1	Tiempo de muestreo PID_3Step en segundos CycleTime.Value se determina automáticamente y equivale normalmente al tiempo de ciclo del OB invocante.
CtrlParamsBackUp.SetByUser	BOOL	FALSE	Valor almacenado de Retain.CtrlParams.SetByUser Los valores de la estructura CtrlParamsBackUp pueden volverse a cargar con Config.LoadBackUp = TRUE.
CtrlParamsBackUp.Gain	REAL	1.0	Ganancia proporcional almacenada
CtrlParamsBackUp.Ti	REAL	20.0	Tiempo de integración almacenado
CtrlParamsBackUp.Td	REAL	0.0	Tiempo derivativo almacenado
CtrlParamsBackUp.TdFiltRatio	REAL	0.0	Coeficiente almacenado para el retardo de la acción derivada
CtrlParamsBackUp.PWeighting	REAL	0.0	Ponderación almacenada de la acción P
CtrlParamsBackUp.DWeighting	REAL	0.0	Ponderación almacenada de la acción D
CtrlParamsBackUp.Cycle	REAL	1.0	Tiempo de muestreo almacenado del algoritmo PID
CtrlParamsBackUp.InputDeadBand	REAL	0.0	Ancho de zona muerta almacenado del error de regulación
PIDSelfTune.SUT.CalculateSUTParams	BOOL	FALSE	Las propiedades del sistema regulado se almacenan durante la optimización. Si CalculateSUTParams = TRUE, se vuelven a calcular los parámetros PID basándose en estas propiedades. Los parámetros PID se calculan según el método que está ajustado TuneRuleSUT. CalculateSUTParams se ajusta a FALSE después del cálculo.
PIDSelfTune.SUT.TuneRuleSUT	INT	1	Calcular los parámetros durante la optimización inicial según el método: • TuneRuleSUT = 0: PID rápido I • TuneRuleSUT = 1: PID lento I • TuneRuleSUT = 2: PID según Chien, Hrones, Reswick • TuneRuleSUT = 3: Chien, Hrones, Reswick PI • TuneRuleSUT = 4: PID rápido II • TuneRuleSUT = 5: PID lento II
PIDSelfTune.SUT.State	INT	0	La variable SUT.State muestra la fase actual de la optimización inicial:

Variable	Tipo de datos	Ajuste predeterminado	Descripción
PIDSelfTune.TIR.RunIn	BOOL	FALSE	• RunIn = FALSE Si la optimización fina se inicia desde los modos de operación Inactivo o Manual, se inicia una optimización inicial. Si la optimización fina se inicia desde el modo automático, los parámetros PID existentes se regulan a la consigna. Solo entonces comienza la optimización fina. Si no es preciso realizar la optimización inicial, PID_3Step cambia al modo de operación "Inactivo". • RunIn = TRUE La optimización inicial se omite. PID_3Step intenta alcanzar la consigna con el valor de salida mínimo o máximo, lo que puede ocasionar una sobreoscilación muy alta. Solo entonces comienza la optimización fina. RunIn se ajusta a FALSE después de la optimización fina.
PIDSelfTune.TIR.CalculateTIRParams	BOOL	FALSE	Las propiedades del sistema regulado se almacenan durante la optimización. Si CalculateTIRParams = TRUE, se vuelven a calcular los parámetros PID basándose en estas propiedades. Los parámetros PID se calculan según el método que está ajustado TuneRuleTIR. CalculateTIRParams se ajusta a FALSE después del cálculo.
PIDSelfTune.TIR.TuneRuleTIR	INT	0	Calcular los parámetros durante la optimización fina según el método: • TuneRuleTIR = 0: PID automático • TuneRuleTIR = 1: PID rápido • TuneRuleTIR = 2: PID lento • TuneRuleTIR = 3: PID Ziegler-Nichols • TuneRuleTIR = 4: PI Ziegler-Nichols • TuneRuleTIR = 5: P Ziegler-Nichols
PIDSelfTune.TIR.State	INT	0	La variable TIR.State muestra la fase actual de la "optimización fina":
Retain.Mode	INT	0	Si cambia el valor de Retain.Mode, se conmuta a otro modo de operación. El siguiente modo de operación se activa al cambiar de Mode a: • Mode = 0: Inactivo • Mode = 1: Optimización inicial • Mode = 2: Optimización fina • Mode = 3: Modo automático • Mode = 4: Modo manual • Mode = 5: Aproximar al valor de salida sustitutivo • Mode = 6: Medición del tiempo de posicionamiento • Mode = 7: Monitorización de errores • Mode = 8: Aproximar al valor de salida sustitutivo con monitorización de errores Mode es remanente.

Variable	Tipo de datos	Ajuste predeter minado	Descripción
Retain.CtrlParams.SetByUser	BOOL	FALSE	Si SetByUser = FALSE, los parámetros PID se determinan de forma automática y PID_3Step funciona con una zona muerta en el valor de salida. El ancho de la zona muerta se calcula durante la optimización a partir de una desviación estándar del valor de salida y se almacena en Retain.CtrlParams.OutputDeadBand. Si SetByUser = TRUE, los parámetros PID se introducen manualmente y PID_3 Step funciona sin zona muerta en el valor de salida. Retain.CtrlParams.OutputDeadBand = 0.0 SetByUser es remanente.
Retain.CtrlParams.Gain	REAL	1.0	Ganancia proporcional activa Gain es remanente.
Retain.CtrlParams.Ti	REAL	20.0	• Ti > 0.0: Tiempo de integración activo • Ti = 0.0: La acción I está desactivada Ti es remanente.
Retain.CtrlParams.Td	REAL	0.0	• Td > 0.0: Tiempo derivativo activo • Td = 0.0: La acción D está desactivada Td es remanente.
Retain.CtrlParams.TdFiltRatio	REAL	0.0	Coeficiente activo para el retardo de la acción derivada TdFiltRatio es remanente.
Retain.CtrlParams.PWeighting	REAL	0.0	Ponderación activa de la acción P PWeighting es remanente.
Retain.CtrlParams.DWeighting	REAL	0.0	Ponderación activa de la acción D DWeighting es remanente.
Retain.CtrlParams.Cycle	REAL	1.0	Tiempo de muestreo activo del algoritmo PID en segundos que se redondea a un múltiplo entero del tiempo de ciclo del OB invocante. Cycle es remanente.
Retain.CtrlParams.InputDeadBand	REAL	0.0	Ancho de zona muerta del error de regulación InputDeadBand es remanente.

Nota

Modifique las variables que aparecen aquí en el modo de operación "Inactivo" para evitar un comportamiento erróneo del regulador PID. El modo de operación "Inactivo" se fuerza con el valor "0" en la variable "Retain.Mode".

Consulte también

Parámetros State y Retain.Mode V1 (Página 5229)

Variable ActivateRecoverMode V1 (Página 5239)

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Parámetros State y Retain.Mode V1

Interrelación de los parámetros

El parámetro State indica el modo de operación actual del regulador PID. El parámetro State no puede modificarse.

Para cambiar el modo de operación debe modificarse la variable Retain.Mode. Esto también se aplica cuando Retain.Mode ya contiene el valor para el nuevo modo de operación. A continuación, por ejemplo, puede ajustarse primero Retain.Mode = 0 y, después, Retain.Mode = 3. Si el modo de operación actual del regulador permite este cambio, State adopta el valor de Retain.Mode.

Cuando PID_3Step cambia automáticamente el modo de operación, se aplica lo siguiente: State != Retain.Mode.

Ejemplos:

- Después de una optimización inicial correcta
  State = 3 y Retain.Mode = 1
- En caso de error
  State = 0 y Retain.Mode mantiene el valor que tenía hasta ahora; por ejemplo, Retain.Mode = 3
- ManualEnalbe = TRUE
  State = 4 y Retain.Mode mantiene el valor que tenía hasta ahora; por ejemplo, Retain.Mode = 3

Nota

Por ejemplo, es posible repetir una optimización fina correcta sin tener que finalizar el modo automático con Mode = 0.

Si en un ciclo se ajusta Retain.Mode a un valor no válido, como p. ej. 9999, esto no afecta en modo alguno a State. En el siguiente ciclo se ajusta Mode = 2. Es posible crear una modificación en Retain.Mode sin tener que cambiar primero al modo de operación "Inactivo".

Significado de los valores

State / Retain.Mode	Descripción
0	Inactivo El regulador está desconectado y ya no modifica la posición de la válvula.
1	Optimización inicial La optimización inicial determina el comportamiento del proceso a un impulso del valor de salida y busca el punto de inflexión. A partir de la inclinación máxima y del tiempo muerto del sistema regulado se calculan los parámetros PID óptimos. Requisitos para la optimización inicial: • State = 0 o State = 4 • ManualEnable = FALSE • El tiempo de posicionamiento del motor está configurado o medido. • La consigna y el valor real se encuentran dentro de los límites configurados. Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. Antes de volver a calcular los parámetros PID, se guarda una copia de seguridad de estos y es posible reactivarlos con Config.LoadBackUp. La consigna se congela en la variable CurrentSetpoint. Una vez realizada correctamente la optimización inicial se pasa al modo automático; si dicha optimización no se realiza correctamente, se pasa al modo de operación "Inactivo". La fase de optimización inicial se muestra con la variable SUT.State.

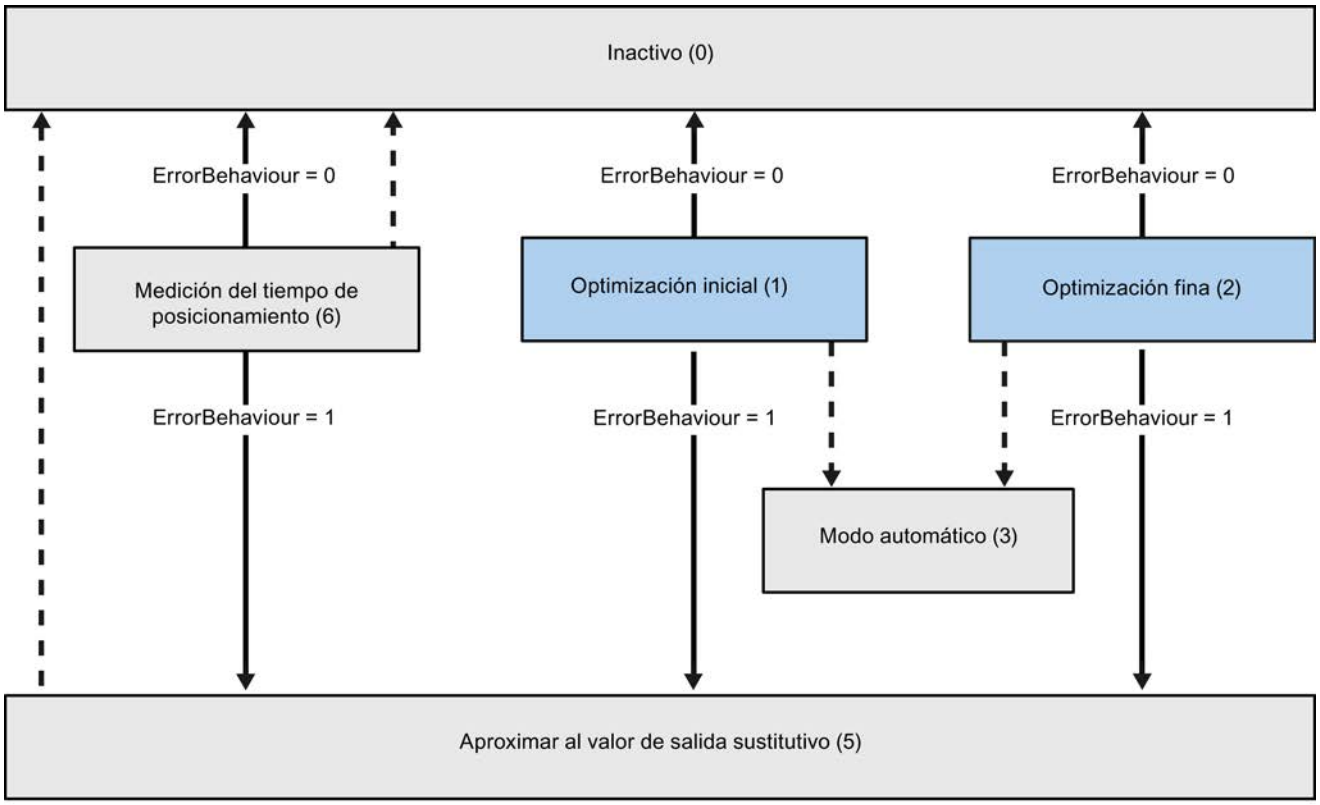
State / Retain.Mode	Descripción
2	Optimización fina La optimización fina genera una oscilación constante y limitada del valor real. A partir de la amplitud y frecuencia de esta oscilación se optimizan los parámetros PID. Se analizan las diferencias entre el comportamiento del proceso durante la optimización inicial y la optimización fina. A partir de los resultados se vuelven a calcular todos los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial. PID_3Step intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan solo una mínima influencia sobre la optimización fina. Antes de la optimización fina se realiza una copia de seguridad de los parámetros PID. Estos pueden reactivarse con Config.LoadBackUp. La consigna se congela en la variable CurrentSetpoint. Requisitos para la optimización fina: • El tiempo de posicionamiento del motor está configurado o medido. • La consigna y el valor real se encuentran dentro de los límites configurados. • ManualEnable = FALSE • Modo de operación Automático (State = 3), Inactivo (State = 0) o Manual (State = 4) Al inicio, la optimización fina se desarrolla del modo siguiente: • Modo automático (State = 3) Si desea mejorar los parámetros PID existentes mediante optimización, inicie la optimización fina desde el modo automático. PID_3Step regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se cumplen las condiciones para la optimización fina. Solo entonces comienza la optimización fina. • Modo Inactivo (State = 0) o Manual (State = 4) Se realiza primero una optimización inicial. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Si PIDSelfTune.TIR.RunIn = TRUE, se omite la optimización inicial y se intenta alcanzar la consigna con el valor de salida máximo o mínimo. lo que puede ocasionar una sobreoscilación muy alta. La optimización fina comienza entonces automáticamente. Una vez realizada correctamente la optimización fina, el regulador cambia al modo de operación "Automático" y, si dicha optimización no se ejecuta correctamente, se pasa al modo de operación "Inactivo". La fase de optimización fina se muestra con la variable TIR.State.
3	Modo automático En el modo automático, PID_3Step regula el sistema regulado conforme a los parámetros predefinidos. Cuando se cumple uno de los requisitos siguientes, se cambia al modo automático: • Optimización inicial finalizada correctamente. • Optimización fina finalizada correctamente. • Cambio de la variable Retain.Mode al valor 3. Si se conecta la CPU o se cambia de STOP a RUN, PID_3Step arranca en el último modo de operación activo. Para dejar PID_3Step en el modo de operación "Inactivo", ajuste RunModeByStartup = FALSE. En el modo automático se tiene en cuenta la variable ActivateRecoverMode.

State / Retain.Mode	Descripción
4	Modo manual En el modo manual, se especifican valores de salida manuales en los parámetros Manual_UP y Manual_DN o ManualValue. El parámetro ErrorBits describe si el actuador se puede desplazar al valor de salida en caso de error. Este modo de operación se activa si Retain.Mode = 4 o en caso de un flanco ascendente en ManualEnable. Si ManualEnable adopta el valor TRUE, únicamente cambia State. Retain.Mode permanece en el valor actual. Con un flanco descendente en ManualEnable, PID_3Step regresa al modo de operación previo. El cambio al modo automático se efectúa sin discontinuidad. **PID_3Step V1.1** En caso de error siempre se puede recurrir al modo manual. **PID_3Step V1.0** En caso de error, el modo manual depende de las variables ActivateRecoverMode.
5	Aproximar al valor de salida sustitutivo Este modo de operación se activa en caso de error o si Reset = TRUE cuando Errorbehaviour = 1 y ActivateRecoverMode = FALSE.. PID_3Step desplaza el elemento final de control al valor de salida sustitutivo y cambia después al modo de operación "Inactivo".
6	Medición del tiempo de posicionamiento Se mide el tiempo que requiere el motor para abrir completamente la válvula a partir del estado cerrado. Este modo de operación se activa si a su vez se activa GetTransitTime.Start = TRUE. Si se utilizan señales de tope para la medición del tiempo de posicionamiento, la válvula se abre completamente desde la posición actual, luego se cierra completamente y se vuelve a abrir completamente. Si GetTransitTime.InvertDirection = TRUE, el comportamiento se invierte. Si se utiliza una realimentación de posición para la medición del tiempo de posicionamiento, el elemento final de control se desplaza desde la posición actual a una posición de destino. Los límites del valor de salida no se tienen en cuenta durante la medición del tiempo de posicionamiento. El actuador puede desplazarse hasta el tope superior o inferior.

State / Retain.Mode	Descripción
7	Monitorización de errores El algoritmo de regulación está desconectado y ya no modifica la posición de la válvula. Este modo de operación se activa en caso de error en lugar del modo de operación "Inactivo". Deben haberse cumplido todas las condiciones siguientes: • Mode = 3 (modo automático) • Errorbehaviour = 0 • ActivateRecoverMode = TRUE • Se han producido uno o varios errores en los que interviene ActivateRecoverMode (Página 5239). En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático.
8	Aproximar al valor de salida sustitutivo con monitorización de errores Este modo de operación se activa en caso de error en lugar del modo de operación "Aproximar al valor de salida sustitutivo". PID_3Step desplaza el elemento final de control o actuador al valor de salida sustitutivo y cambia después al modo de operación "Monitorización de errores". Deben haberse cumplido todas las condiciones siguientes: • Mode = 3 (modo automático) • Errorbehaviour = 1 • ActivateRecoverMode = TRUE • Se han producido uno o varios errores en los que interviene ActivateRecoverMode (Página 5239). En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático.

Cambio automático de modo de operación durante la puesta en servicio

En caso de error, PID_3Step cambia automáticamente el modo de operación. El diagrama siguiente muestra la influencia de ErrorBehaviour en el cambio de modo desde los modos de medición del tiempo de posicionamiento, optimización inicial y optimización fina.

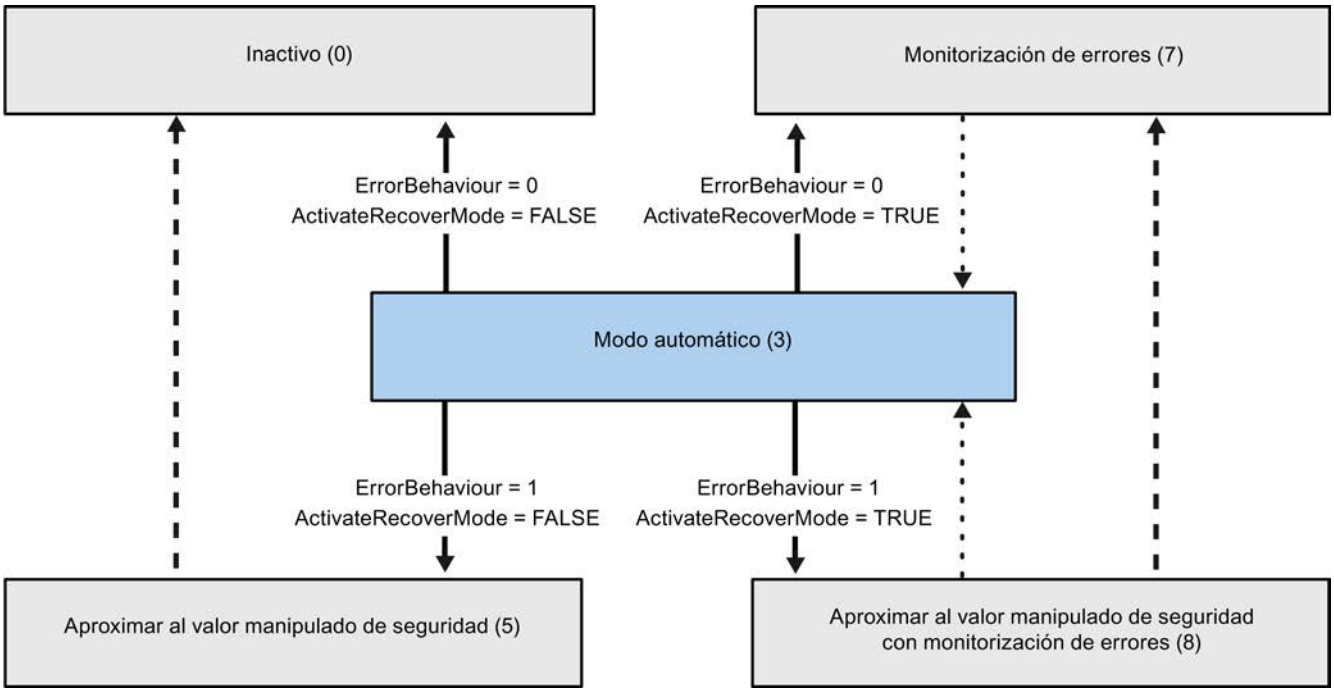


Cambio automático del modo de operación en caso de error

Cambio automático del modo de operación cuando ha concluido el modo actual.

Cambio automático del modo de operación en el modo automático (PID_3Step V1.1)

En caso de error, PID_3Step cambia automáticamente el modo de operación. El diagrama siguiente muestra la influencia de ErrorBehaviour y ActivateRecoverMode sobre este cambio del modo de operación.

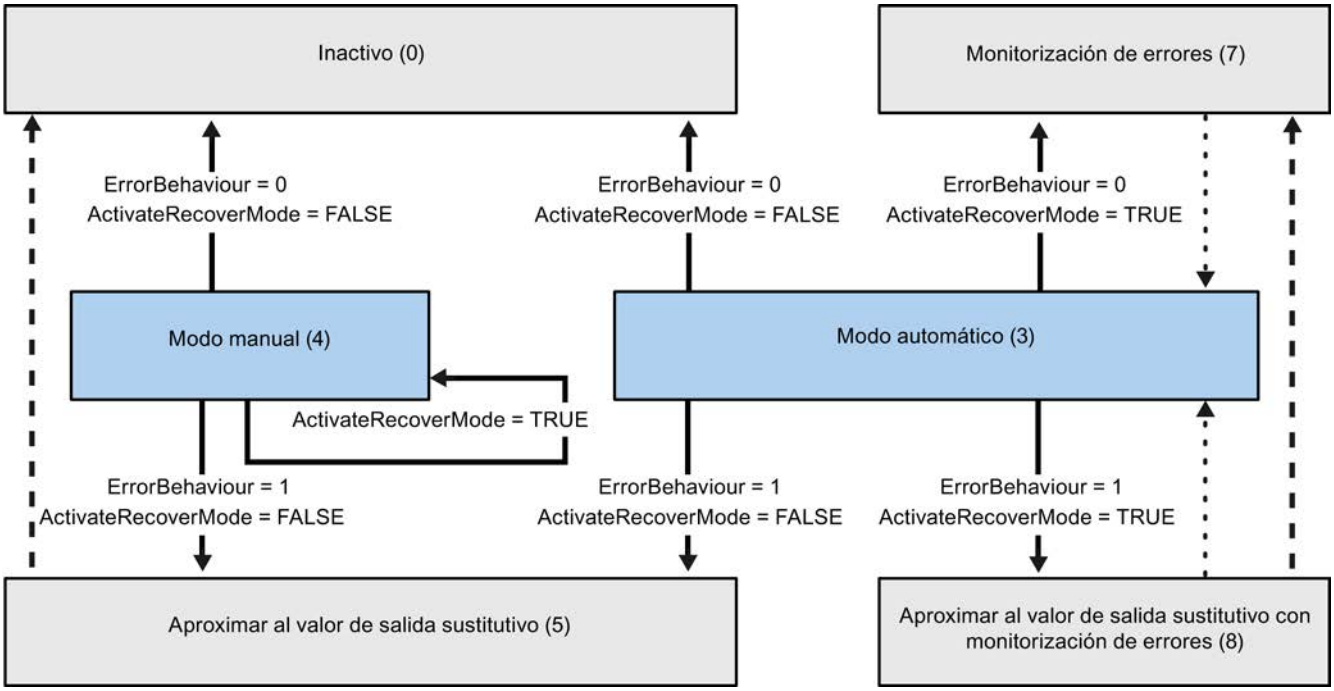


Cambio automático del modo de operación en caso de error

Cambio automático del modo de operación cuando ha concluido el modo actual.

Cambio automático del modo de operación en el modo automático y en el modo manual (PID_3Step V1.0)

En caso de error, PID_3Step cambia automáticamente el modo de operación. El diagrama siguiente muestra la influencia de ErrorBehaviour y ActivateRecoverMode sobre este cambio del modo de operación.



Cambio automático del modo de operación en caso de error

Cambio automático del modo de operación cuando ha concluido el modo actual.

Cambio automático del modo de operación cuando ya no existe el error.

Consulte también

Variable ActivateRecoverMode V1 (Página 5239)

Parámetro ErrorBits V1 (Página 5237)

Parámetro ErrorBits V1

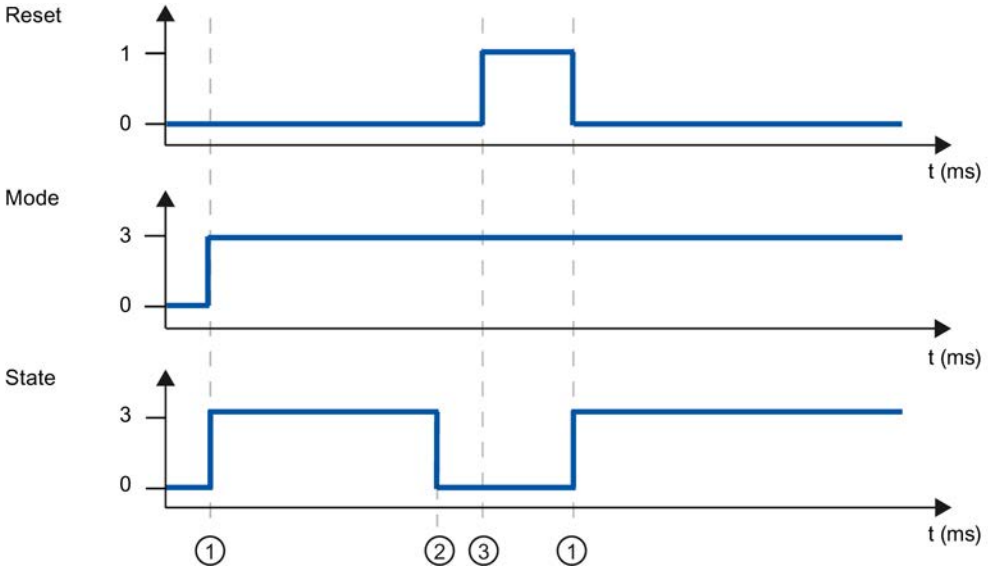
Si hay varios errores presentes a la vez, los valores del código de error se muestran sumados binariamente. Si se muestra, p. ej., el código de error 0003, significa que están presentes simultáneamente los errores 0001 y 0002.

ErrorBits (DW#16#...)	Descripción
0000	No hay ningún error.
0001	El parámetro "Input" se encuentra fuera de los límites del valor real. • Input > Config.InputUpperLimit o bien, • Input < Config.InputLowerLimit Si ActivateRecoverMode = TRUE y ErrorBehaviour = 1, el actuador se desplaza al valor de salida sustitutivo. Si ActivateRecoverMode = TRUE y ErrorBehaviour = 0, el actuador se mantiene en la posición actual. Si ActivateRecoverMode = FALSE, el actuador se mantiene en la posición actual. PID_3Step V1.1 En el modo manual, el actuador se puede desplazar. PID_3Step V1.0 El modo manual no es posible en este estado. Una vez eliminado el error, el actuador se puede volver a desplazar.
0002	Valor no válido en el parámetro "Input_PER". Compruebe si hay un error en la entrada analógica. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.
0004	Error durante la optimización fina. No se ha podido mantener la oscilación del valor real.
0020	La optimización inicial no está permitida en el modo automático ni durante la optimización fina.
0080	Error durante la optimización inicial. Los límites del valor de salida no están configurados correctamente. Compruebe si los límites del valor de salida están configurados correctamente y si encajan con el sentido de la regulación.
0100	Un error en la optimización fina ha provocado parámetros no válidos.
0200	Valor no válido en el parámetro "Input": El valor no tiene un formato numérico válido. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.
0400	Error al calcular el valor de salida. Compruebe los parámetros PID.
0800	Error de tiempo de muestreo: PID_3Step no se activa dentro del tiempo de muestreo del OB de alarma cíclica. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.
1000	Valor no válido en el parámetro "Setpoint": El valor no tiene un formato numérico válido. Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.
2000	Valor no válido en el parámetro Feedback_PER. Compruebe si hay un error en la entrada analógica. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual. El modo manual no es posible en este estado. Para poder desplazar el actuador desde este estado, es necesario desactivar la realimentación de posición (Config. FeedbackOn = FALSE). Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.

ErrorBits (DW#16#...)	Descripción
4000	Valor no válido en el parámetro Feedback. El valor no tiene un formato numérico válido. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual. El modo manual no es posible en este estado. Para poder desplazar el actuador desde este estado, es necesario desactivar la realimentación de posición (Config. FeedbackOn = FALSE). Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.
8000	Error en la realimentación digital de posición. Actuator_H = TRUE y Actuator_L = TRUE. El actuador no puede desplazarse al valor de salida sustitutivo y permanece en la posición actual. El modo manual no es posible en este estado. Para poder desplazar el actuador desde este estado, es necesario desactivar las "Señales de tope del actuador" (Config.ActuatorEndStopOn = FALSE). Si antes de aparecer el error estaba activo el modo automático, ActivateRecoverMode = TRUE y el error ya no está presente, PID_3Step pasa de nuevo al modo automático.

Parámetro Reset V1

Un flanco ascendente en Reset hace que se desactiven los errores y las advertencias y que se borre la acción I. Un flanco descendente en Reset provoca un cambio al último modo de operación activo.



① Conexión
② Error
③ Reset

Variable ActivateRecoverMode V1

La influencia de las variables ActivateRecoverMode depende de la versión de la instrucción PID_3Step.

Comportamiento en la versión 1.1

La variable ActivateRecoverMode determina en el modo automático el comportamiento en caso de error. Durante la optimización inicial, la optimización fina y la medición del tiempo de posicionamiento, ActivateRecoverMode no afecta.

ActivateRecover Mode	Descripción
FALSE	En caso de error, PID_3Step pasa al modo de operación "Inactivo" o "Aproximar al valor de salida sustitutivo". El regulador únicamente se activa tras un rearme o después de un cambio en Retain.Mode.
TRUE	Si en el modo automático se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación. Compruebe en tal caso el parámetro ErrorBits y elimine la causa de error. Si se produce uno o varios de los errores siguientes, PID_3Step pasa al modo operativo "Aproximar a valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores": • 0002h: Valor no válido en el parámetro Input_PER. • 0200h: Valor no válido en el parámetro Input. • 0800h: Error de tiempo de muestreo • 1000h: Valor no válido en el parámetro Setpoint. • 2000h: Valor no válido en el parámetro Feedback_PER. • 4000h: Valor no válido en el parámetro Feedback. • 8000h: Error en la realimentación digital de posición. Con los errores 2000h, 4000h y 8000h, PID_3Step no puede efectuar una aproximación al valor de salida sustitutivo configurado. En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático.

Comportamiento en la versión 1.0

La variable ActivateRecoverMode determina tanto en el modo automático como en el modo manual el comportamiento en caso de error. Durante la optimización inicial, la optimización fina y la medición del tiempo de posicionamiento, ActivateRecoverMode no afecta.

ActivateRecover Mode	Descripción
FALSE	En caso de error, PID_3Step pasa al modo de operación "Inactivo" o "Aproximar al valor de salida sustitutivo". El regulador únicamente se activa tras un rearme o después de un cambio en Retain.Mode.
TRUE	**Error en el modo automático** Si en el modo automático se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación. Compruebe en tal caso el parámetro ErrorBits y elimine la causa de error. Si se produce uno o varios de los errores siguientes, PID_3Step pasa al modo operativo "Aproximar a valor de salida sustitutivo con monitorización de errores" o "Monitorización de errores": • 0002h: Valor no válido en el parámetro Input_PER. • 0200h: Valor no válido en el parámetro Input. • 0800h: Error de tiempo de muestreo • 1000h: Valor no válido en el parámetro Setpoint. • 2000h: Valor no válido en el parámetro Feedback_PER. • 4000h: Valor no válido en el parámetro Feedback. • 8000h: Error en la realimentación digital de posición. Con los errores 2000h, 4000h y 8000h, PID_3Step **no** puede efectuar una aproximación al valor de salida sustitutivo configurado. En cuanto los errores dejan de estar activos, PID_3Step pasa de nuevo al modo automático. **Error en el modo manual** Si se produce uno o varios de los errores siguientes, PID_3Step permanece en el modo manual: • 0002h: Valor no válido en el parámetro Input_PER. • 0200h: Valor no válido en el parámetro Input. • 0800h: Error de tiempo de muestreo • 1000h: Valor no válido en el parámetro Setpoint. • 2000h: Valor no válido en el parámetro Feedback_PER. • 4000h: Valor no válido en el parámetro Feedback. • 8000h: Error en la realimentación digital de posición. Con los errores 2000h, 4000h y 8000h, la válvula **no** se puede desplazar a una posición apropiada.

Consulte también

Variables estáticas PID_3Step V1 (Página 5221)

Parámetros State y Retain.Mode V1 (Página 5229)

Variable Warning V1

Si hay varias advertencias presentes a la vez, los valores de las advertencias se muestran sumados binariamente. Si se muestra p. ej. la advertencia 0003, significa que están presentes simultáneamente las advertencias 0001 y 0002.

Warning (DW#16#...)	Descripción
0000	No hay advertencias pendientes.
0001	Durante la optimización inicial no se encontró el punto de inflexión.
0002	Durante la optimización fina la oscilación fue más intensa.
0004	La consigna ha sido limitada a los límites ajustados.
0008	En el método de cálculo seleccionado no se han definido todas las propiedades necesarias del sistema regulado. En sustitución se calcularon los parámetros PID con el método TuneRuleTIR = 3.
0010	No se ha podido cambiar el modo de operación, pues ManualEnable = TRUE.
0020	El tiempo de muestreo del algoritmo PID está limitado por el tiempo de ciclo del OB invocante. Para obtener mejores resultados, utilice tiempos de ciclo más cortos para el OB.
0040	El valor real ha rebasado uno de sus límites de advertencia.
0080	Valor no válido en Retain.Mode. El modo de operación no se conmuta.
0100	El valor manual ha sido limitado a los límites de la salida del regulador.
0200	La regla utilizada para la optimización no conduce a ningún resultado correcto o bien no se soporta.
0400	En la medición del tiempo de posicionamiento se ha seleccionado un método que no encaja con el elemento final de control. El tiempo de posicionamiento no se puede medir porque los ajustes del elemento final de control no concuerdan con el método de medición seleccionado.
0800	En la medición del tiempo de posicionamiento, la diferencia entre la posición actual y el nuevo valor de salida es insuficiente. Esto puede ocasionar resultados erróneos. La diferencia entre el valor de salida actual y el nuevo valo rde salida debe ser de al menos un 50% del rango de posicionamiento total.
1000	El valor de salida sustitutivo no puede alcanzarse porque está fuera de los límites del valor de salida.

Las siguientes advertencias se eliminan en cuanto la causa se soluciona:

- 0004
- 0020
- 0040
- 0100

Todas las demás advertencias se eliminan si hay un flanco ascendente en Reset.

Variable SUT.State V1

SUT.State	Nombre	Descripción
0	SUT_INIT	Iniciar la optimización inicial
50	SUT_TPDN	Determinar la posición inicial sin realimentación de posición
100	SUT_STDABW	Calcular desviación estándar
200	SUT_GET_POI	Determinar punto de inflexión

SUT.State	Nombre	Descripción
300	SUT_GET_RISETM	Determinar el tiempo de subida
9900	SUT_IO	Optimización inicial correcta
1	SUT_NIO	Optimización inicial no correcta

Variable TIR.State V1

TIR.State	Nombre	Descripción
-100	TIR_FIRST_SUT	No es posible realizar la optimización fina. Primero se efectúa una optimización inicial.
0	TIR_INIT	Iniciar la optimización fina
200	TIR_STDABW	Calcular desviación estándar
300	TIR_RUN_IN	Intentar alcanzar la consigna con el valor de salida máximo o mínimo.
400	TIR_CTRLN	Intentar alcanzar consigna con los parámetros PID existentes (si la optimización inicial fue correcta)
500	TIR_OSZIL	Determinar oscilación y calcular parámetros
9900	TIR_IO	Optimización fina correcta
1	TIR_NIO	Optimización fina no correcta

Funciones básicas PID

Diferencias respecto a CONT_C S7-300/400

La instrucción CONT_C se ha creado con acceso optimizado al bloque.

Todos los parámetros son remanentes. La remanencia no puede modificarse. Los valores actuales de variables remanentes solo se actualizan si se carga CONT_C por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Por lo demás, CONT_C se comporta en una CPU S7-1500 igual que en una CPU S7-300 o S7-400.

Descripción de CONT_C (Página 7078)

Funcionamiento de CONT_C (Página 7079)

Diagrama de bloques de CONT_C (Página 7081)

Parámetros de entrada de CONT_C (Página 7081)

Parámetros de salida de CONT_C (Página 7083)

Diferencias respecto a CONT_S S7-300/400

La instrucción CONT_S se ha creado con acceso optimizado al bloque. S

Todos los parámetros son remanentes. La remanencia no puede modificarse. Los valores actuales de variables remanentes solo se actualizan si se carga CONT_S por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Por lo demás, CONT_S se comporta en una CPU S7-1500 igual que en una CPU S7-300 o S7-400.

Descripción de CONT_S (Página 7084)

Funcionamiento de CONT_S (Página 7084)

Diagrama de bloques de CONT_S (Página 7086)

Parámetros de entrada de CONT_S (Página 7086)

Parámetros de salida de CONT_S (Página 7088)

Diferencias respecto a PULSEGEN S7-300/400

La instrucción PULSEGEN se ha creado con acceso optimizado al bloque.

Todos los parámetros son remanentes. La remanencia no puede modificarse. Los valores actuales de variables remanentes solo se actualizan si se carga PULSEGEN por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

En una CPU S7-1500, PULSEGEN se comporta igual que en una CPU S7-300, S7-400.

Descripción de PULSEGEN (Página 7088)

Funcionamiento de PULSEGEN (Página 7089)

Modos de operación de PULSEGEN (Página 7093)

Regulación de tres puntos (Página 7093)

Regulación de dos puntos (Página 7095)

Parámetros de entrada de PULSEGEN (Página 7097)

Parámetros de salida de PULSEGEN (Página 7098)

Diferencias respecto a TCONT_CP S7-300/400

La instrucción TCONT_CP se ha creado con acceso optimizado al bloque.

Todos los parámetros son remanentes. La remanencia no puede modificarse. Los valores actuales de variables remanentes solo se actualizan si se carga TCONT_CP por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Por lo demás, TCONT_CP se comporta en una CPU S7-1500 igual que en una CPU S7-300 o S7-400.

Descripción de TCONT_CP (Página 7098)

Funcionamiento de TCONT_CP (Página 7099)

Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Parámetros de entrada de TCONT_CP (Página 7111)

Parámetros de salida de TCONT_CP (Página 7112)

Parámetros de entrada/salida de TCONT_CP (Página 7113)

Variables estáticas de TCONT_CP (Página 7113)

Parámetro STATUS_H (Página 7118)

Parámetro STATUS_D (Página 7119)

Diferencias respecto a TCONT_S S7-300/400

La instrucción TCONT_S se ha creado con acceso optimizado al bloque.

Todos los parámetros son remanentes. La remanencia no puede modificarse. Los valores actuales de variables remanentes solo se actualizan si se carga TCONT_S por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9940)

Por lo demás, TCONT_S se comporta en una CPU S7-1500 igual que en una CPU S7-300 o S7-400.

Descripción de TCONT_S (Página 7120)

Funcionamiento de TCONT_S (Página 7121)

Diagrama de bloques de TCONT_S (Página 7124)

Parámetros de entrada de TCONT_S (Página 7125)

Parámetros de salida de TCONT_S (Página 7126)

Parámetros de entrada/salida de TCONT_S (Página 7126)

Variables estáticas de TCONT_S (Página 7127)

9.8.1.5 Comunicación

Comunicación S7

Coherencia de datos

Definición

El bloque de datos que no puede ser modificado por procesos concurrentes se denomina área de datos coherente. Así, un bloque de datos que supera el tamaño del área de datos coherente puede falsearse en su totalidad al transferirse. Esto significa que un bloque de datos conexos cuyo volumen supere el del área de datos coherente, puede estar compuesto en un momento dado en parte por datos nuevos, y en parte por datos coherentes antiguos.

Ejemplo

Puede surgir una incoherencia si se interrumpe una instrucción de comunicación, p. ej., mediante un OB de alarma de proceso con mayor prioridad. Si el programa de usuario de este

OB modifica ahora los datos que en parte ya había procesado la instrucción, los datos transferidos proceden:

- en parte, de antes de la ejecución de la alarma de proceso
- y, en parte, de después de ejecutar la alarma de proceso.

Esto significa que los datos son incoherentes (no concuerdan entre sí).

Aseguramiento de la coherencia de datos

Si el proceso de comunicación puede ser interrumpido por un OB de alarma, hay que asegurar que los datos se transmitan con coherencia. Asegúrese de que los datos transferidos no sean modificados directamente por el OB de alarma, sino que este modifique solo una imagen de los datos. Copie la imagen de los datos en el área de transferencia de la instrucción de comunicación antes de la siguiente transferencia de datos.

- Si en el programa de usuario existe una instrucción de comunicación que acceda a datos comunes, el acceso a esa área de datos puede coordinarse, por ejemplo, con el parámetro DONE. La coherencia de datos de las áreas de comunicación que se transmite localmente con una instrucción de comunicación puede garantizarse por tanto en el programa de usuario.
- En las instrucciones de comunicación S7 "PUT (Página 5252)"/"GET (Página 5249)", el tamaño de las áreas de datos coherentes ya debe tenerse en cuenta en la programación o configuración, puesto que en el programa de usuario del dispositivo de destino (servidor) no hay ningún bloque de comunicación que pueda sincronizar los datos de comunicación al entrar en el programa de usuario:
- Con la CPU S7-300 y C7-300 (excepción: CPU 318-2 DP) los datos de comunicación se copian en el programa de usuario con coherencia en bloques de 32 bytes en el punto de control de ciclo del sistema operativo. Para todas las áreas de datos de mayor tamaño no se garantiza la coherencia de datos. Si se requiere una coherencia de datos definida, los datos de comunicación en el programa de usuario no deben superar los 32 bytes (según la versión, máximo 8 bytes).
- Por el contrario, en las CPU S7-400 y S7-1500 los datos de comunicación no se procesan en bloques de 462 bytes en el punto de control del ciclo, sino en segmentos de tiempo fijos durante el ciclo del programa. El propio sistema garantiza la coherencia de una variable. Así, un OP o una OS, por ejemplo, puede acceder con coherencia a estas áreas de comunicación con las instrucciones "PUT (Página 5252)" / "GET (Página 5249)" o bien leer/escribir variables.

Nota

Encontrará más indicaciones sobre la coherencia de datos en la descripción de las distintas instrucciones.

Efecto en los tiempos de respuesta de alarma

Al copiar los datos, los tiempos de respuesta de alarma se prolongan mínimamente. Cuanto más datos deban transmitirse con garantía de coherencia, tanto mayor será el tiempo de reacción a alarmas de un sistema.

Parámetros comunes de las instrucciones para la comunicación S7

Clasificación

Los parámetros de las instrucciones para la comunicación S7 pueden clasificarse en las siguientes cinco categorías funcionales:

1. Los parámetros de control permiten activar una instrucción.
2. Los parámetros de direccionamiento sirven para direccionar el interlocutor remoto.
3. Los parámetros de transmisión indican las diferentes áreas de datos que deben transmitirse al interlocutor remoto.
4. Los parámetros de recepción indican las áreas de datos en las que se registran los datos recibidos del interlocutor remoto.
5. Los parámetros de estado permiten controlar si la instrucción ha terminado su cometido correctamente, o bien analizar los errores ocurridos.

Parámetros de control

El intercambio de datos no se activa hasta que en la llamada de la instrucción los parámetros de control correspondientes no tengan un valor definido (p. ej. que estén activados) o hasta que el valor haya cambiado en un modo determinado respecto a la última llamada (p. ej. flanco ascendente).

Parámetros de direccionamiento

Parámetro	Descripción
ID	Referencia a la descripción de la conexión local (predeterminada por la configuración de conexiones).
R_ID	Con el parámetro R_ID se define la correspondencia entre una instrucción de transmisión y una de recepción: El parámetro R_ID debe coincidir en las instrucciones del lado de transmisión y del lado de recepción. Ello permite la comunicación de varios pares de instrucciones a través de la misma conexión lógica. • R_ID debe indicarse en la forma DW#16#wxyzWXYZ. • Los pares de instrucciones de una conexión lógica definidos con R_ID deben ser unívocos para dicha conexión.

Nota

Parámetros de direccionamiento ID y R_ID

Los parámetros de direccionamiento ID y R_ID pueden reparametrizarse en el tiempo de ejecución. Los nuevos parámetros se hacen efectivos con cada nueva petición al finalizar la petición anterior.

Las siguientes posibilidades permiten reducir el número de DBs de instancia y, con ello, ahorrar memoria de trabajo:

1. Con IDs variables pueden utilizarse varias conexiones a través de un bloque de datos de instancia.
2. Con R_IDs variables es posible definir con una instancia varias correspondencias entre pares de instrucciones de transmisión y recepción para una petición.
3. Los casos 1 y 2 pueden combinarse entre sí.

Recuerde que los nuevos parámetros no se hacen efectivos hasta no haberse completado la petición anterior. Al activar el proceso de transmisión, el parámetro R_ID debe coincidir en la instrucción del lado de transmisión y en la del lado de recepción.

Parámetros de estado

Con los parámetros de control se controla si la instrucción ha terminado su cometido correctamente o si todavía está activa. Además, indican los errores ocurridos.

Nota

Los parámetros de estado sólo son válidos durante un ciclo, es decir, desde el primer comando que sigue a la llamada, hasta la siguiente llamada. Por ello es necesario evaluar estos parámetros cada vez que se ejecuta la instrucción.

Parámetros de transmisión y recepción

En las instrucciones para la comunicación bilateral

- el número de parámetros SD_i y RD_i utilizados en los lados de transmisión y recepción debe coincidir
- en los parámetros SD_i y RD_i conexos deben coincidir los tipos de datos en los lados de transmisión y recepción
- el número de datos que deben transferirse a través del parámetro SD_i no puede ser superior al área abarcada por el parámetro correspondiente RD_i (no es válido para "BSEND (Página 5258)" / "BRCV (Página 5261)"). Los parámetros RD_i (a excepción de "BSEND"/"BRCV") deben direccionar el mismo volumen de datos.

La infracción de estas reglas se indica con ERROR = 1 y STATUS = 4.

Nota

Alimentación de los parámetros de transmisión y recepción

Los parámetros de transmisión y recepción con el tipo de datos VARIANT deben tener alimentación siempre que se llame cualquier instrucción de comunicación. No es posible alimentar los parámetros de transmisión de las instrucciones de comunicación en el arranque y lanzar solo la petición de transmisión durante el funcionamiento cíclico.

Volumen de datos de usuario

En las instrucciones "USEND (Página 5255)", "URCV (Página 5256)", "GET (Página 5249)" y "PUT (Página 5252)", el número de datos que deben transferirse no puede superar un volumen de datos de usuario determinado. El volumen máximo de datos de usuario depende:

- de la instrucción utilizada y
- del interlocutor

El volumen mínimo garantizado de datos de usuario para una instrucción con 1-4 variables se puede consultar en la siguiente tabla:

Instrucción	Interlocutor: S7-300	Interlocutor: S7-400	Interlocutor: S7-1200	Interlocutor: S7-1500
PUT / GET	160 bytes	400 bytes	160 bytes	880 bytes
USEND / URCV	160 bytes	440 bytes	-	920 bytes
BSEND / BRCV	32768/65534 bytes	65534 bytes	-	• 65534 bytes en el acceso estándar • 65535 bytes en el acceso optimizado

Observe las limitaciones relativas al volumen de datos de usuario en los datos técnicos de la respectiva CPU.

Volumen exacto de datos de usuario

Si el volumen de datos de usuario indicado anteriormente no es suficiente, es posible determinar la longitud de datos de usuario máxima exacta en bytes del siguiente modo:

En primer lugar consulte en la tabla siguiente el tamaño válido para bloques de datos en la comunicación:

CPU propia	CPU remota	Tamaño del bloque de datos en bytes
S7-1200	cualquiera	240
S7-1500	S7-300	240
	S7-400	480

CPU propia	CPU remota	Tamaño del bloque de datos en bytes
	S7-1200	240
	S7-1500	960

Partiendo de este valor, consulte en la tabla siguiente la longitud de datos de usuario máxima posible en bytes, como suma de los parámetros de usuario. Es aplicable a longitudes pares de las áreas SD_i, RD_i, ADDR_i.

Para cada área de longitud impar la longitud de datos de usuario máxima posible se reduce en un byte.

		Número de parámetros utilizados SD_i, RD_i, ADDR_i			
Tamaño del bloque de datos	Instrucción	1	2	3	4
240 (S7-300)	PUT/GET/ USEND	160	-	-	-
240 (S7-300 vía interfaz integrada)	PUT	212	-	-	-
	GET	222	-	-	-
	USEND	212	-	-	-
240 (S7-400)	PUT	212	196	180	164
	GET	222	218	214	210
	USEND	212	-	-	-
480 (S7-400)	PUT	452	436	420	404
	GET	462	458	454	450
	USEND	452	448	444	440
240 (S7-1200)	PUT	212	196	180	164
	GET	222	218	214	210
960 (S7-1500)	PUT	932	916	900	884
	GET	942	938	934	930
	USEND	932	928	924	920

GET: Leer datos de una CPU remota

Descripción

La instrucción "GET" permite leer datos de una CPU remota.

Con un flanco ascendente en la entrada de control REQ se inicia la instrucción:

- Los punteros hacia las áreas que deben leerse (ADDR_i) se transfieren a la CPU interlocutora. La CPU interlocutora puede encontrarse en el estado operativo RUN o STOP.
- La CPU interlocutora devuelve los contenidos de los datos:
  - Si la respuesta supera la longitud máxima de datos de usuario, se indica en el parámetro STATUS con el código de error "2".
  - Los datos recibidos se copian durante la siguiente llamada de la instrucción en las áreas de recepción configuradas (RD_i).
- El término de este proceso se indica con "1" en el parámetro de estado NDR.

No es posible activar otro proceso de lectura hasta que no haya finalizado el último. En caso de problemas de acceso al leer los datos, o si la comprobación del tipo de datos devolvió un error, los errores y advertencias correspondientes se indican a través de ERROR y STATUS.

Las modificaciones en áreas de datos direccionadas en la CPU interlocutora no se registran mediante la instrucción "GET".

Requisitos para el uso de la instrucción

- En las propiedades de la CPU interlocutora se ha activado la función "Permitir acceso vía comunicación PUT/GET del interlocutor remoto" para la CPU interlocutora, en "Protección".
- Los bloques a los que se accede con la instrucción "GET" se han creado con el tipo de acceso "estándar".
- Hay que asegurarse de que las áreas definidas por los parámetros ADDR_i y SD_i se correspondan en cuanto a la cantidad, la longitud y el tipo de datos.
- El área que se debe leer (parámetro ADDR_i) no puede ser mayor que el área de almacenamiento de datos (parámetro RD_i).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	I, Q, M, D, L o constante	Parámetro de direccionamiento para indicar la conexión con la CPU interlocutora.
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: – 0000H: ni advertencia ni error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	– <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
ADDR_1	InOut	REMOTE	I, Q, M, D	Puntero hacia las áreas de la CPU interlocutora que deben leerse. Si el puntero REMOTE accede a un DB, éste siempre debe especificarse. Ejemplo: P#DB10.DBX5.0 Byte 10.
ADDR_2	InOut	REMOTE		
ADDR_3	InOut	REMOTE		
ADDR_4	InOut	REMOTE		
RD_1	InOut	VARIANT	I, Q, M, D, L	Puntero hacia las áreas de la CPU propia en las que se depositan los datos leídos.
RD_2	InOut	VARIANT		
RD_3	InOut	VARIANT		
RD_4	InOut	VARIANT		

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "GET" que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida
1	2	• Acuse negativo del interlocutor. La función no puede ejecutarse. • La respuesta de la estación remota supera la longitud de datos de usuario máxima (consulte: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)). • La protección contra acceso está activada en la CPU interlocutora. Desactive la protección contra acceso en los ajustes de la CPU.
1	4	Error en los punteros de almacenamiento de datos RD_i: • Los tipos de datos de los parámetros RD_i y ADDR_i no son compatibles entre sí. • La longitud del área RD_i es menor que la longitud de los datos que se deben leer de los parámetros ADDR_i.
1	8	Error de acceso en la CPU interlocutora.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	20	• Se ha excedido el número máximo de peticiones paralelas. • La petición se llamará en una clase de baja prioridad (primera llamada).

Nota

Coherencia de datos

Los datos se reciben con coherencia si lee la parte actualmente utilizada del área de recepción RD_i por completo antes de activar otra petición.

PUT: Escribir datos en una CPU remota

Descripción

La instrucción "PUT" permite escribir datos en una CPU remota.

Con un flanco ascendente en la entrada de control REQ se inicia la instrucción:

- Los punteros hacia las áreas que deben escribirse (ADDR_i) y los datos (SD_i) se transfieren a la CPU interlocutora. La CPU interlocutora puede encontrarse en el estado operativo RUN o STOP.
- Los datos que deben enviarse se copian desde las áreas de transmisión configuradas (SD_i). La CPU interlocutora deposita los datos transmitidos en las direcciones indicadas y devuelve un acuse de ejecución.
- Si no se ha producido ningún error, esto se indica en la siguiente llamada de la instrucción con "1" en el parámetro de estado DONE. No es posible activar otro proceso de escritura hasta que no haya finalizado el último.

En caso de problemas de acceso al escribir los datos, o si la comprobación del acuse de ejecución devolvió un error, los errores y advertencias correspondientes se indican a través de ERRORy STATUS.

Requisitos para el uso de la instrucción

- En las propiedades de la CPU interlocutora se ha activado la función "Permitir acceso vía comunicación PUT/GET del interlocutor remoto" para la CPU interlocutora, en "Protección".
- Los bloques a los que se accede con la instrucción "PUT" se han creado con el tipo de acceso "estándar".
- Hay que asegurarse de que las áreas definidas por los parámetros ADDR_i y SD_i se correspondan en cuanto a la cantidad, la longitud y el tipo de datos.
- El área que se debe escribir (parámetro ADDR_i) debe ser del mismo tamaño que el área de transmisión (parámetro SD_i).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PUT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	I, Q, M, D, L o constante	Parámetro de direccionamiento para indicar la conexión con la CPU interlocutora.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: – 0000H: ni advertencia ni error – <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	
ADDR_1	InOut	REMOTE	I, Q, M, D	Puntero hacia las áreas de la CPU interlocutora en las que debe escribirse. Si el puntero REMOTE accede a un DB, éste siempre debe especificarse. Ejemplo: P#DB10.DBX5.0 Byte 10. En la transferencia de estructuras de datos (p. ej. Struct) debe utilizarse el tipo de datos CHAR en los parámetros ADDR_i.
ADDR_2	InOut	REMOTE		
ADDR_3	InOut	REMOTE		
ADDR_4	InOut	REMOTE		
SD_1	InOut	VARIANT	I, Q, M, D, L	Puntero hacia las áreas de la CPU propia que contienen los datos que deben transmitirse. Solo están permitidos los tipos de datos BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL. En la transferencia de estructuras de datos (p. ej. Struct) debe utilizarse el tipo de datos CHAR en los parámetros SD_i.
SD_2	InOut	VARIANT		
SD_3	InOut	VARIANT		
SD_4	InOut	VARIANT		

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "PUT" que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP). • Conexión con el interlocutor todavía no establecida.
1	2	• Acuse negativo de la CPU interlocutora. La función no puede ejecutarse. • El acceso en la CPU interlocutora no se ha permitido. Active el acceso en los ajustes de la CPU.
1	4	Error en los punteros de almacenamiento de datos: • Los tipos de datos de los parámetros SD_i y ADDR_i no son compatibles entre sí. • La longitud del área SD_i es mayor que la longitud de los datos que se deben escribir de los parámetros ADDR_i. • No es posible acceder a SD_i. • Se ha excedido el volumen máximo de datos de usuario. • El número de parámetros SD_i y ADDR_i no coinciden.
1	8	Error de acceso en la CPU interlocutora (p. ej. no se ha cargado el DB o está protegido contra escritura).
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	20	• Se ha excedido el número máximo de peticiones paralelas. • La petición se llamará en una clase de baja prioridad (primera llamada).

Coherencia de datos

Con la activación de un proceso de transmisión (flanco ascendente en REQ) los datos que deben transmitirse de las áreas de transmisión SD_i se copian del programa de usuario. Se puede volver a escribir en estas áreas tras la llamada del bloque, sin que se falsifiquen los datos de transmisión actuales.

Nota

El proceso de transmisión completo finaliza sólo cuando el parámetro de estado DONE adopta el valor "1".

Otras

USEND: Envío de datos sin coordinación

Descripción

La instrucción "USEND" envía datos a una instrucción remota de interlocutor del tipo "URCV (Página 5256)". El proceso de transmisión se desarrolla sin coordinación con la instrucción del interlocutor, es decir, que la transmisión de datos se produce sin acuse por parte de la instrucción del interlocutor.

Con la activación de un proceso de transmisión (flanco ascendente en REQ), los datos que deben transmitirse de las áreas de transmisión SD_i se copian del programa de usuario. Se puede volver a escribir en estas áreas de transmisión tras la llamada de la instrucción sin que se falsifiquen los datos de transmisión actuales.

La finalización correcta del proceso de transmisión se indica con "1" en el parámetro de estado DONE .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "USEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	CONN_PRG	I, Q, M, D, L, P o constante	Parámetro de direccionamiento para indicar la conexión con la CPU interlocutora.
R_ID	Input	CONN_R_ID	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID para definir los pares de instrucciones "USEND" y "URCV". Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ni advertencia ni error. • 1: Hay un error. STATUS devuelve información detallada sobre el tipo de error.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado Véase tabla "Parámetros ERROR y STATUS".
SD_i (1≤ i ≤4)	InOut	VARIANT	I, Q, M, D	Puntero al área de transmisión n.º i. Solo se permiten los tipos de datos BOOL (no permitido: campo de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL o STRUCT. El volumen máximo de datos de usuario para los parámetros SD_i depende de la CPU interlocutora (instrucción "URCV") y del número de parámetros utilizados. Encontrará más información en: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Se ha producido un problema de comunicación. Causas posibles: • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida
1	4	• Error en los punteros de área de transmisión SD_i en cuanto a la longitud o el tipo de datos. • Se ha excedido la longitud máxima de datos de usuario.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	18	El valor del parámetro R_ID ya existe en la conexión especificada en el parámetro ID (el valor para R_ID debe ser unívoco para la conexión).
1	20	• Se ha excedido el número máximo de peticiones paralelas. • La petición se está llamando en una clase de menor prioridad (primera llamada).

URCV: Recibir datos sin coordinación

Descripción

La instrucción "URCV" recibe datos de forma asíncrona de una instrucción remota del interlocutor del tipo "USEND (Página 5255)" y los copia en las áreas de recepción configuradas.

La instrucción está lista para recibir cuando en la entrada EN_R hay un "1" lógico. Con EN_R=0 no es posible interrumpir una petición en curso.

Las áreas de datos de recepción se referencian con los parámetros RD_1, ... RD_4. Hay que asegurarse de que las áreas definidas por los parámetros RD_i/RD_1 y SD_i/SD_1 (en la respectiva instrucción del interlocutor correspondiente "USEND (Página 5255)") coincidan en número y longitud.

La finalización correcta del proceso de copia se indica con un "1" lógico en el parámetro de estado NDR . Después de que el parámetro de estado NDR haya adoptado el valor "1", hay nuevos datos de recepción en las áreas de recepción (RD_i). Una nueva llamada del bloque puede sobrescribir estos datos con nuevos datos de recepción. Para impedirlo, llame "URCV" con el valor "0" en EN_R (p. ej. en caso de ejecución cíclica del bloque) hasta haber finalizado el procesamiento de los datos de recepción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "URCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control enabled to receive señaliza disponibilidad de recepción cuando la entrada está activada.
ID	Input	CONN_PRG	I, Q, M, D, L, P o constante	Parámetro de direccionamiento para indicar la conexión con la CPU interlocutora.
R_ID	Input	CONN_R_ID	I, Q, M, D, L o constante	Parámetro de direccionamiento para definir los pares de instrucciones "USEND" y "URCV". Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: ni advertencia ni error • 1: Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	Parámetros de estado Véase tabla "Parámetros ERROR y STATUS".
RD_i (1≤ i ≤4)	InOut	VARIANT	I, Q, M, D	Puntero al área de recepción n.º i. Solo se permiten los tipos de datos BOOL(no permitido: campo de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL o STRUCT. Encontrará más información en: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (decimal)	Explicación
0	9	Advertencia: Los datos de recepción antiguos han sido sobrescritos por datos de recepción más actuales.
0	11	Advertencia: Los datos de recepción ya se están procesando con una clase de menor prioridad (puede producirse un error al copiar los datos en el área de recepción).
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Se ha producido un problema de comunicación. Causas posibles: • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida
1	4	Error en los punteros de áreas de recepción RD_i en cuanto a la longitud o el tipo de datos.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	18	El valor del parámetro R_ID ya existe en la conexión especificada en el parámetro ID (el valor para R_ID debe ser unívoco para la conexión).
1	19	La instrucción correspondiente "USEND (Página 5255)" envía datos más rápido de lo que "URCV" puede copiarlos en las áreas de recepción.
1	20	• Se ha excedido el número máximo de peticiones paralelas. • La petición se está llamando en una clase de menor prioridad (primera llamada).

BSEND: Transmitir datos en segmentos

Descripción

La instrucción "BSEND" envía datos a una instrucción remota del interlocutor del tipo "BRCV (Página 5261)". En este tipo de transferencia de datos es posible transportar una cantidad de datos mayor entre los interlocutores que con las demás instrucciones de comunicación para conexiones S7 configuradas. La cantidad de datos máxima asciende a 65534 bytes (acceso estándar) o 65535 bytes (acceso optimizado) tanto en la interfaz integrada como en el CP SIMATIC Net.

Funcionamiento

El par de instrucciones "BSEND" y "BRCV" se define con el parámetro R_ID. El parámetro R_ID debe ser idéntico en las instrucciones conexas.

El proceso de transmisión se activa después de la llamada de la instrucción y del flanco ascendente en la entrada de control REQ. "BSEND" no se ejecuta en segundo plano tras la llamada, es decir, los datos solo se pueden leer dentro del programa de usuario.

El área de datos que se va a transmitir se segmenta. Cada segmento se transmite individualmente al interlocutor. El interlocutor acusa cada segmento tras la aceptación de dicho segmento mediante "BRCV (Página 5261)". En el caso de una segmentación de los datos, debe llamarse la instrucción "BSEND" varias veces hasta que se hayan transmitido todos los segmentos.

El área de datos de los datos que deben transmitirse viene determinada por SD_1. Para garantizar la coherencia de datos, no puede volver a escribirse en la parte del área de transmisión SD_1 utilizada actualmente hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

La longitud de los datos de transmisión se define individualmente para cada petición con LEN. Si LEN = "0" se envían todos los datos direccionados por el parámetro SD_1.

Con un flanco ascendente en la entrada de control R se cancela el proceso de transmisión en curso.

Debido a la transmisión de datos asíncrona, no es posible iniciar una nueva transmisión de datos mientras los datos anteriores no hayan sido recogidos por la llamada de la instrucción del interlocutor. Cuando los datos han sido recogidos, en la instrucción del interlocutor "BRCV" se ajusta el parámetro de estado "NDR".

Nota

Migración de programas de usuario S7-400

Una CPU S7-400 interpreta el parámetro SD_1 como puntero, no como área de datos.

En el S7-1500 LEN no puede rebasar el área de SD1. En el S7-400 estaba permitido. Recomendación: Utilice como tamaño del área de datos en el parámetro SD_1 el tamaño máximo permitido para el parámetro LEN (65534 bytes con la interfaz integrada).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "BSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
R	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control reset activa la cancelación de un intercambio de datos todavía en curso con un flanco ascendente.
ID	Input	CONN_PRG	I, Q, M, D, L, P o constante	Parámetro de direccionamiento para indicar la conexión con la CPU interlocutora.
R_ID	Input	CONN_R_ID	I, Q, M, D, L o constante	Parámetro de direccionamiento para definir los pares de instrucciones "BSEND" y "BRCV (Página 5261)". Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)
SD_1	InOut	VARIANT	I, Q, M, D	Puntero al área de transmisión
LEN	InOut	WORD	I, Q, M, D, L	Longitud del bloque de datos que debe transferirse en bytes. Si LEN = "0" se envían todos los datos de SD_1.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: Ni advertencia ni error. • 1: Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado Véase tabla "Parámetros ERROR y STATUS".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de "BSEND", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Se ha producido un problema de comunicación. Causas posibles: • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida
1	2	Acuse negativo de la instrucción del interlocutor. La instrucción no puede ejecutarse.
1	3	R_ID no es conocido en la conexión de comunicación especificada por ID, o el bloque de recepción no se ha llamado nunca.
1	4	• Error en el puntero del área de transmisión SD_1 en cuanto a la longitud o el tipo de datos. • El valor de LEN es superior al área SD_1.
1	5	Se ha ejecutado la solicitud de reset.
1	6	La instrucción del interlocutor se encuentra en estado DISABLED (EN_R tiene el valor "0"). Compruebe también si los parámetros de entrada de "BRCV (Página 5261)" son coherentes con "BSEND".
1	7	La instrucción del interlocutor "BRCV (Página 5261)" no se ha vuelto a llamar desde la última transmisión de datos.
1	8	Se ha rechazado el acceso al objeto remoto en la memoria de usuario: El área de destino en el "BRCV (Página 5261)" correspondiente es demasiado pequeña. En los parámetros de salida de "BRCV (Página 5261)" se notifica ERROR = 1, STATUS = 4 o ERROR = 1, STATUS = 10.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).

ERROR	STATUS (decimal)	Explicación
1	18	R_ID ya existe en la conexión.
1	20	• Se ha excedido el número máximo de peticiones paralelas. • La petición se está llamando en una clase de menor prioridad (primera llamada).

BRCV: Recibir datos en segmentos

Descripción

La instrucción "BRCV" recibe datos de una instrucción remota del interlocutor del tipo "BSEND (Página 5258)". El parámetro R_ID debe ser idéntico en las instrucciones conexas.

La instrucción está lista para recibir tras la llamada con el valor "1" en la entrada de control EN_R (STATUS = 25). Con EN_R=0 no es posible interrumpir una petición en curso.

El área de recepción máxima viene determinada por RD_1. Los datos se reciben con coherencia si evalúa por completo la parte del área de recepción RD_1 utilizada actualmente antes de volver a llamar el bloque con el valor "1" en la entrada de control EN_R.

Después de cada segmento de datos recibido se envía un acuse a la instrucción del interlocutor. En el caso de varios segmentos, es necesario llamar la instrucción "BRCV" varias veces hasta que se hayan recibido todos los segmentos. La recepción de datos asíncrona se indica mediante STATUS = 17. La cantidad actual de datos recibidos se indica en el parámetro LEN. El parámetro RD_1 debe permanecer constante durante la operación.

La recepción sin errores de todos los segmentos de datos se indica en el parámetro de estado NDR con el valor "1". Los datos recibidos permanecen invariables hasta la siguiente llamada con EN_R=1.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "BRCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control enabled to receive señaliza disponibilidad de recepción cuando la entrada está activada.
ID	Input	CONN_PRG	I, Q, M, D, L, P o constante	Parámetro de direccionamiento para indicar la conexión con la CPU interlocutora.
R_ID	Input	CONN_R_ID	I, Q, M, D, L o constante	Parámetro de direccionamiento para definir los pares de instrucciones "BSEND (Página 5258)" y "BRCV". Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 5246)
RD_1	InOut	VARIANT	I, Q, M, D	Puntero al área de recepción.
LEN	InOut	WORD	I, Q, M, D, L	Longitud de los datos recibidos hasta el momento en bytes.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado • 0: ni advertencia ni error • 1: Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	Parámetros de estado Véase tabla "Parámetros ERROR y STATUS".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de "BRCV", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	17	Advertencia: La instrucción recibe datos de forma asíncrona. El parámetro LEN indica el número de datos recibidos hasta ahora en bytes.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Se ha producido un problema de comunicación. Causas posibles: • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida
1	2	Función no ejecutable (error de protocolo)
1	4	Error en el puntero del área de recepción RD_1 en cuanto a la longitud o el tipo de datos. El bloque de datos enviado supera la longitud del área de recepción.
1	5	Petición de reset recibida, transmisión incompleta.
1	8	Error de acceso en el respectivo "BSEND (Página 5258)": Tras enviar el último segmento de datos válido, se indica ERROR = 1 y STATUS = 4 o ERROR = 1 y STATUS = 10.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	18	El valor del parámetro R_ID ya existe en la conexión especificada en el parámetro ID (el valor para R_ID debe ser unívoco para la conexión).
1	20	• Se ha excedido el número máximo de peticiones paralelas. • La petición se está llamando en una clase de menor prioridad (primera llamada).

Open User Communication

TSEND_C: Enviar datos a través de Ethernet

TSEND_C: Enviar datos a través de Ethernet

Descripción

La instrucción "TSEND_C" configura y establece una conexión TCP o ISO on TCP. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente. Para configurar la conexión se utiliza la descripción de la conexión especificada en el parámetro CONNECT.

La instrucción se ejecuta de forma asíncrona y tiene las funciones siguientes:

- Configurar y establecer una conexión de comunicación:
  La conexión se configura y establece con CONT=1. Si la conexión se establece correctamente, el parámetro DONE se pone a "1" durante un ciclo. Si la CPU pasa al estado operativo STOP, se interrumpe una conexión existente y se elimina la conexión creada. Para volver a configurar y establecer la conexión, es preciso volver a ejecutar "TSEND_C". El número de conexiones posibles se indica en los datos técnicos de la CPU.
- Enviar datos a través de la conexión existente:
  el área de transmisión se especifica en el parámetro DATA. Éste contiene la dirección y la longitud de los datos que deben enviarse. No utilice en el parámetro DATA áreas de datos con el tipo de datos BOOL o Array of BOOL. Si utiliza únicamente valores simbólicos en el parámetro DATA, el parámetro LEN debe tener el valor "0".
- La petición de transmisión se ejecuta cuando se detecta un flanco ascendente en el parámetro REQ. En el parámetro LEN se especifica el número máximo de bytes que deben enviarse con una petición de transmisión. Al enviar datos (flanco ascendente en el parámetro REQ), el parámetro CONT deberá tener el valor "1" para establecer o mantener una conexión. Los datos por enviar no se pueden editar hasta que no se haya ejecutado por completo la petición de transmisión. Si la petición de transmisión se ejecuta correctamente, el parámetro DONE se pone a "1". No obstante, el estado lógico "1" en el parámetro DONE no confirma que el interlocutor haya leído ya los datos enviados.
- Deshacer la conexión:
  La conexión se deshace cuando el parámetro CONT se pone al valor "0" aunque no haya finalizado aún una transferencia de datos en curso. Esto no es aplicable si se utiliza una conexión ya configurada para "TSEND_C".

Ajustando el parámetro COM_RST a "1", se puede resetear el establecimiento de la conexión o una transferencia de datos en curso en cualquier momento. Con ello, la conexión existente se deshace y se establece una nueva. Si se están transfiriendo datos al reiniciarse la instrucción, podrían perderse datos.

Para volver a habilitar "TSEND_C" tras la ejecución (DONE = 1), debe llamar la instrucción una vez con REQ = 0.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TSEND_C":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Inicia la petición de transmisión cuando se detecta un flanco ascendente.
CONT	Input	BOOL	I, Q, M, D, L	Controla la conexión: • 0: Deshacer la conexión • 1: Establecer y mantener la conexión Al enviar datos (flanco ascendente en el parámetro REQ), el parámetro CONT deberá tener el valor TRUE para establecer o mantener una conexión.
LEN	Input	UINT	I, Q, M, D, L o constante	Número máximo de bytes que se envían con la petición. Si utiliza únicamente valores simbólicos en el parámetro DATA, el parámetro LEN debe tener el valor "0".
CONNECT	InOut	TCON_Param	D	Puntero hacia la descripción de la conexión Consulte también: Auto-Hotspot
DATA	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el área de transmisión que contiene la dirección y la longitud de los datos por enviar.
COM_RST	InOut	BOOL	I, Q, M, D, L	Reinicia la instrucción: • 0: Irrelevante • 1: Reinicio completo de la instrucción, con lo que se deshace la conexión existente y se establece una nueva.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No es posible iniciar una nueva petición.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TSEND_C". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	-	-	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Descripción
0	0000	Petición finalizada sin errores.
0	0001	El establecimiento de la conexión ha finalizado.
0	0003	La desconexión ha finalizado.
0	7000	No se está procesando ninguna petición.
0	7001	• Iniciar procesamiento de la petición • Establecer conexión • Esperar al interlocutor
0	7002	Datos enviados.
0	7003	Se está deshaciendo la conexión.
0	7004	Conexión establecida y vigilada, no se está procesando ninguna petición.
1	80A0	Error de grupo para los códigos de error 80A1 y 80A2.
1	80A1	• El usuario ya está utilizando la conexión o el puerto. • Error de comunicación: – La conexión indicada no se ha establecido todavía. – La conexión indicada se está finalizando. No es posible transferir a través de esta conexión. – La interfaz se está reinicializando.
1	80A2	El sistema está utilizando el puerto local o remoto.
1	80A3	Se está intentando finalizar una conexión no existente.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida, es decir, concuerda con la dirección IP del interlocutor local.
1	80A7	Error de comunicación: Se ha llamado la instrucción con COM_RST = 1 antes de finalizar la petición de transmisión.
1	80AA	Se está intentando establecer una conexión desde otro bloque con el mismo ID de conexión. Repita la petición con un nuevo flanco ascendente en el parámetro REQ.
1	80B2	El parámetro CONNECT apunta a un bloque de datos que ha sido generado con el atributo "Guardar sólo en la memoria de carga".
1	80B3	Parametrización incoherente: Error de grupo para los códigos de error 80A0 a 80A2, 80A4, 80B4 a 80B9.
1	80B4	Al establecer una conexión pasiva (active_est = FALSE) con la variante de protocolo ISO on TCP (connection_type = B#16#12) se han infringido una o ambas de las condiciones siguientes: "local_tsap_id_len >= B#16#02" y/o "local_tsap_id[1] = B#16#E0".

ERROR	STATUS* (W#16#...)	Descripción
1	80B5	Con el tipo de conexión 13 = UDP sólo se permite un establecimiento pasivo de la conexión.
1	80B6	Error de parametrización en el parámetro connection_type del bloque de datos para la descripción de la conexión.
1	80B7	Error en uno de los parámetros siguientes del bloque de datos para la descripción de la conexión: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	8085	El parámetro LEN excede el valor máximo admisible.
1	8086	El parámetro ID dentro del parámetro CONNECT está fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones.
1	8088	El valor del parámetro LEN no es compatible con el área de recepción indicada en el parámetro DATA.
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	8091	Se ha excedido la profundidad de anidamiento máxima.
1	809A	El parámetro CONNECT apunta a un campo que no concuerda con la longitud de la descripción de la conexión.
1	809B	La ID del dispositivo local indicado en la descripción de la conexión no concuerda con la CPU.
1	80C3	• Todos los recursos de conexión están ocupados. • Ya se está procesando un bloque con esta ID en otro grupo de prioridad.
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos parámetros o la conexión se está estableciendo. • Una instrucción "TDISCON" está deshaciendo la conexión configurada. • La conexión utilizada está siendo finalizada por una llamada con COM_RST= 1
1	80C6	Error de red remoto. No es posible acceder al interlocutor remoto.
1	8722	Parámetro CONNECT: El rango de origen no es válido. El rango no existe en el DB.
1	873A	Parámetro CONNECT: Imposible acceder a la descripción de la conexión (p. ej. porque el DB no está disponible).
1	877F	Parámetro CONNECT: Error interno.
1	8822	Parámetro DATA: El rango de origen no es válido, el rango no existe en el DB.
1	8824	Parámetro DATA: Error de área en el puntero VARIANT.
1	8832	Parámetro DATA: El número de DB es demasiado alto.
1	883A	Parámetro CONNECT: Imposible acceder a los datos de conexión indicados (p. ej., porque el DB no está disponible).
1	887F	Parámetro DATA: Error interno, p. ej., referencia VARIANT no admisible.
1	893A	Parámetro DATA: Imposible acceder al área de transmisión (p. ej., porque el DB no está disponible).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Nota

Mensajes de error de las instrucciones "TCON", "TSEND", "T_DIAG" y "TDISCON"

La instrucción "TSEND_C" utiliza internamente las instrucciones "TCON (Página 5298)", "TSEND (Página 5310)", "T_DIAG (Página 5331)", "T_RESET (Página 5329) y "TDISCON (Página 5305)". Los mensajes de error de estas instrucciones pueden emitirse también en el parámetro STATUS. El significado de los códigos de error está descrito en las instrucciones correspondientes. En caso de haber códigos de error idénticos de instrucciones usadas internamente con un significado distinto, es posible determinar la instrucción que ha emitido el error mediante el bloque de datos de instancia de "TSEND_C".

TSEND_C: Enviar datos a través de Ethernet

Descripción

La instrucción "TSEND_C" permite configurar y establecer una conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.

La instrucción se ejecuta de forma asíncrona y tiene las funciones siguientes:

- Configurar y establecer una conexión
- Enviar datos a través de la conexión existente
- Deshacer o inicializar la conexión

La instrucción "TSEND_C" utiliza internamente las instrucciones de comunicación "TCON", "TSEND", "T_DIAG", "T_RESET" y "TDISCON".

Configurar y establecer una conexión

La conexión de comunicación se crea y establece con CONT=1. El número de conexiones posibles se indica en los datos técnicos de la CPU. Para configurar la conexión se utiliza la descripción de la conexión especificada en el parámetro CONNECT. Existen los siguientes tipos de conexión:

- Conexiones programadas (establecimiento de la conexión con "TCON"):
  - TCP/UDP: Descripción de la conexión mediante el tipo de datos de sistema TCON_IP_v4
  - ISO on TCP: Descripción de la conexión mediante el tipo de datos del sistema TCON_IP_RFC
  - ISO: Descripción de la conexión mediante el tipo de datos de sistema TCON_ISOnative (solo con CP1543-1)
- Conexiones configuradas
  - Especificación de una conexión existente en el tipo de datos del sistema TCON_Configured.

Si la CPU pasa al estado operativo STOP, se interrumpe una conexión existente y se elimina la conexión creada. Para volver a configurar y establecer la conexión, es preciso volver a ejecutar "TSEND_C".

Enviar datos a través de la conexión existente

La petición de transmisión se ejecuta cuando se detecta un flanco ascendente en el parámetro REQ. Como se describe anteriormente, primero se establece la conexión.

El área de transmisión se especifica en el parámetro DATA. Éste contiene la dirección y la longitud de los datos que deben enviarse. No utilice en el parámetro DATA áreas de datos con el tipo de datos BOOL o Array of BOOL. En el parámetro LEN se especifica el número máximo de bytes que deben enviarse con una petición de transmisión. Si utiliza únicamente un área de transmisión de acceso optimizado en el parámetro DATA, el parámetro LEN debe tener el valor "0".

Los datos por enviar no se pueden editar hasta que no se haya ejecutado por completo la petición de transmisión.

Deshacer e inicializar la conexión

La conexión se deshace cuando el parámetro CONT se pone al valor "0" aunque no haya finalizado aún una transferencia de datos en curso. Esto no es aplicable si se utiliza una conexión configurada para "TSEND_C".

Ajustando el parámetro COM_RST a "1" se puede resetear la conexión en cualquier momento. Con ello, la conexión existente se deshace y se establece una nueva. Si se están transfiriendo datos en ese momento, podrían perderse.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TSEND_C":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	Inicia la petición de transmisión cuando se detecta un flanco ascendente.
CONT	Input	BOOL	I, Q, M, D, L	Controla la conexión: • 0: Deshacer la conexión. • 1: Establecer y mantener la conexión.
LEN	Input	UDINT	I, Q, M, D, L o constante	Parámetro opcional (oculto) Número máximo de bytes que se envían con la petición. Si utiliza un área de transmisión de acceso optimizado en el parámetro DATA, el parámetro LEN debe tener el valor "0".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CONNECT	InOut	VARIANT	D	Puntero hacia la estructura de la descripción de la conexión: • Conexión programada: – Con TCP utilice el tipo de datos del sistema TCON_IP_v4. Consulte la descripción en: Auto-Hotspot – Con ISO-on-TCP utilice el tipo de datos del sistema TCON_IP_RFC. Consulte la descripción en: Auto-Hotspot – Con ISO, utilice el tipo de datos de sistema TCON_ISOnative (solo con CP1543-1). Consulte la descripción en la instrucción "TCON (Página 5301)". • Conexión configurada: – Para conexiones existentes, utilice el tipo de datos de sistema TCON_Configured. Consulte la descripción más abajo, en "Tipo de datos de sistema para conexiones configuradas".
DATA	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el área de transmisión que contiene la dirección y la longitud de los datos por enviar.
ADDR	InOut	VARIANT	D	Parámetro opcional (oculto) Puntero a la dirección del receptor.
COM_RST	InOut	BOOL	I, Q, M, D, L	Parámetro opcional (oculto) Efectúa la inicialización de la conexión: • 0: Irrelevante • 1: Se inicializa la conexión existente. El parámetro COM_RST se restablece tras la evaluación mediante la instrucción "TSEND_C" y, por tanto, no debe interconectarse estáticamente.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición de transmisión no iniciada aún o en ejecución. • 1: Petición de transmisión ejecutada sin errores. Este estado solo se muestra durante un ciclo. El parámetro de salida DONE se activa cuando finaliza correctamente un paso intermedio del procesamiento (establecer conexión, enviar, deshacer conexión) y cuando la ejecución de "TSEND_C" finaliza correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición de transmisión no iniciada aún o ya finalizada. • 1: Petición de transmisión no finalizada aún. No es posible iniciar una nueva petición de transmisión.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Ningún error • 1: Se ha producido un error al establecer la conexión, al enviar los datos o al deshacer la conexión. El parámetro de salida ERROR puede activarse debido a un error en la instrucción "TSEND_C" o en las instrucciones de comunicación utilizadas internamente.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción (véase la descripción "Parámetros ERROR y STATUS").

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros REQ, CONT y COM_RST

El parámetro CONT controla el establecimiento de conexión de la instrucción "TSEND_C", independientemente del parámetro REQ. El comportamiento del parámetro CONT depende en parte de si se utiliza una conexión programada o configurada:

- Con CONT = "0": No se envían datos (independientemente de si se utiliza una conexión programada o configurada).
- Al cambiar de CONT = "0" a "1":
  - En el caso de una conexión programada, esta se establece con "TCON".
  - En el caso de una conexión configurada, esta se comprueba con "T_DIAG".
- Con CONT = "1":
  - Mientras no se envíe ningún dato (REQ="0"), la conexión se comprobará con "T_DIAG".
  - Si las instrucciones de comunicación internas notifican que no hay ningún punto final de la conexión, esta se restablecerá automáticamente mediante "TCON".
- Al cambiar de CONT = "1" a "0":
  - En el caso de una conexión programada, esta se deshace con "TDISCON".
  - En el caso de una conexión configurada, esta se inicializa con "T_RESET".

El parámetro COM_RST inicializa la conexión al cambiar de "0" a "1":

- Si hay una conexión establecida, esta se inicializa mediante "T_RESET" (independientemente de si se utiliza una conexión programada o configurada).
- Si no hay ninguna conexión establecida, la activación del parámetro no tiene efecto.

Los parámetros REQ y COM_RST solo se hacen efectivos si se ha puesto CONT a "1". La tabla siguiente muestra la relación entre los parámetros REQ, CONT y COM_RST:

REQ	CONT	COM_ RST	Estado de la instrucción	Descripción
Irrelevante	0	Irrelevante	No se ha ejecutado aún	Ninguna petición activa (STATUS = 7000).
Irrelevante	0	Irrelevante	Inicialización	Se está deshaciendo la conexión. La instrucción se inicializa.
Irrelevante	0 > 1	Irrelevante	Establecimiento de la conexión	La conexión se está estableciendo. Todavía no se transfieren datos.
0	1	0	Conexión establecida	La conexión se ha establecido y se vigila con la instrucción "T_DIAG".
Irrelevante	1	0 > 1	Conexión establecida	La conexión se interrumpe brevemente y se inicializa con "T_RESET".
0 > 1	1	0	Conexión establecida	La instrucción empieza a enviar.
Irrelevante	1	0 > 1	Enviando datos	La transferencia de datos se interrumpe. La conexión se inicializa.

Tipo de datos de sistema para conexiones configuradas

Para conexiones configuradas, utilice en el parámetro CONNECT la siguiente estructura para la descripción de la conexión según TCON_Configured:

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceID	HW_ANY	-	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	-	Referencia a la conexión (rango de valores: de 1 a 4095). Indique la ID de la conexión existente.
4	ConnectionType	BYTE	-	Tipo de conexión Seleccione 254 (decimal) para una conexión configurada.

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. Con el parámetro DONE puede comprobar si una petición de transmisión se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TSEND_C". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

DONE	BUSY	ERROR	Descripción
0	0	0	La instrucción aún no se ha ejecutado (no hay ningún flanco ascendente en el parámetro REQ).
0	1	0	La instrucción se ejecuta y llama las instrucciones de comunicación utilizadas internamente.

DONE	BUSY	ERROR	Descripción
1	0	0	La petición de transmisión se ha ejecutado correctamente. En el parámetro STATUS se devuelve "0000". DONE = "1" solo se muestra durante un ciclo.
0	0	1	La ejecución de la instrucción o un paso intermedio del procesamiento ha finalizado con un error. Si se produce un error como consecuencia de una instrucción de comunicación utilizada internamente, se indica el error que se ha producido en primer lugar durante el procesamiento. Este estado solo se muestra durante un ciclo.

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Descripción
0	0000	Petición de transmisión ejecutada sin errores.
0	0001	Comunicación establecida.
0	0003	Comunicación desconectada.
0	7000	No se está procesando ninguna petición de transmisión, no se ha establecido ninguna conexión.
0	7001	Primera llamada al establecer una conexión.
0	7002	Segunda llamada al establecer una conexión.
0	7003	Se está deshaciendo la conexión.
0	7004	Conexión establecida y vigilada. No hay ninguna petición de transmisión activa.
0	7005	Transmisión de datos en curso.
1	80A1	• El usuario ya está utilizando la conexión o el puerto. • Error de comunicación: – La conexión indicada no se ha establecido todavía. – La conexión indicada se está finalizando. No es posible transferir a través de esta conexión. – La interfaz se está reinicializando.
1	80A3	La instrucción "T_DIAG" subordinada ha notificado que se ha deshecho la conexión.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida o bien concuerda con la dirección IP del interlocutor local.
1	80A7	Error de comunicación: Se ha llamado la instrucción con COM_RST = 1 antes de finalizar la petición de transmisión.
1	80AA	Se está intentando establecer una conexión desde otro bloque con el mismo ID de conexión. Repita la petición con un nuevo flanco ascendente en el parámetro REQ.
1	80B4	Al establecer una conexión pasiva (active_est = FALSE) con la variante de protocolo ISO on TCP (connection_type = B#16#12) se han infringido una o ambas de las condiciones siguientes: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0
1	80B5	Con el tipo de conexión 13 = UDP sólo se permite un establecimiento pasivo de la conexión.
1	80B6	Error de parametrización en el parámetro connection_type del bloque de datos para la descripción de la conexión.
1	80B7	Error en uno de los parámetros siguientes del bloque de datos para la descripción de la conexión: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	8085	El parámetro LEN excede el valor máximo admisible.
1	8086	El parámetro ID dentro del parámetro CONNECT está fuera del rango admisible.

ERROR	STATUS* (W#16#...)	Descripción
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones.
1	8088	El valor del parámetro LEN no es compatible con el área de recepción indicada en el parámetro DATA.
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	8091	Se ha excedido la profundidad de anidamiento máxima.
1	809A	El parámetro CONNECT apunta a un campo que no concuerda con la longitud de la descripción de la conexión.
1	809B	InterfaceID no es válida. Es cero o no apunta a una interfaz de CPU local o un CP.
1	80C3	• Todos los recursos de conexión están ocupados. • Ya se está procesando un bloque con esta ID en otro grupo de prioridad.
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos parámetros o la conexión se está estableciendo. • Una instrucción "TDISCON (Página 5305)" está deshaciendo la conexión configurada. • La conexión utilizada está siendo finalizada por una llamada con COM_RST = 1
1	80C6	Error de red remoto. No es posible acceder al interlocutor remoto.
1	8722	Parámetro CONNECT: El rango de origen no es válido. El rango no existe en el DB.
1	873A	Parámetro CONNECT: Imposible acceder a la descripción de la conexión (p. ej. porque el DB no está disponible).
1	877F	Parámetro CONNECT: Error interno.
1	8822	Parámetro DATA: El rango de origen no es válido, el rango no existe en el DB.
1	8824	Parámetro DATA: Error de área en el puntero VARIANT.
1	8832	Parámetro DATA: El número de DB es demasiado alto.
1	883A	Parámetro CONNECT: Imposible acceder a los datos de conexión indicados (p. ej., porque el DB no está disponible).
1	887F	Parámetro DATA: Error interno, p. ej., referencia VARIANT no admisible.
1	893A	Parámetro DATA: Imposible acceder al área de transmisión (p. ej., porque el DB no está disponible).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Nota

Mensajes de error de las instrucciones "TCON", "TSEND", "T_DIAG", "T_RESET" y "TDISCON"

La instrucción "TSEND_C" utiliza internamente las instrucciones "TCON (Página 5301)", "TSEND (Página 5310)", "T_DIAG (Página 5331)", "T_RESET (Página 5329) y "TDISCON (Página 5305)". Los mensajes de error de estas instrucciones pueden emitirse también en el parámetro STATUS. El significado de los códigos de error está descrito en las instrucciones correspondientes. En caso de haber códigos de error idénticos de instrucciones usadas internamente con un significado distinto, es posible determinar la instrucción que ha emitido el error mediante el bloque de datos de instancia de "TSEND_C".

TRCV_C: Recibir datos a través de Ethernet

TRCV_C: Recibir datos a través de Ethernet

Descripción

La instrucción "TRCV_C" se ejecuta de forma asíncrona y tiene las funciones siguientes:

1. **Configurar y establecer una conexión:**
   "TRCV_C" configura y establece una conexión TCP o ISO on TCP. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.
   Para configurar la conexión se utiliza la descripción de la conexión especificada en el parámetro CONNECT. Para establecer la conexión, el parámetro CONT se debe poner a "1". Si la conexión se establece correctamente, el parámetro DONE se pone a "1".
   Si la CPU pasa al estado operativo STOP, se interrumpe una conexión existente y se elimina la conexión creada. Para volver a configurar y establecer la conexión, es preciso volver a ejecutar "TRCV_C".
   El número de conexiones posibles se indica en los datos técnicos de la CPU.
2. **Recibir datos a través de la conexión existente:**
   cuando el parámetro EN_R se pone a "1", se activa la recepción de datos. Al recibir datos (flanco ascendente en el parámetro EN_R), el parámetro CONT debe tener el valor TRUE para establecer o mantener una conexión.
   Los datos recibidos se escriben en un área de recepción. La longitud del área de recepción se especifica, en función de la variante de protocolo utilizada, bien sea con el parámetro LEN (si LEN <> 0), o bien mediante la indicación de longitud del parámetro DATA (si LEN = 0). Si utiliza únicamente valores simbólicos en el parámetro DATA, el parámetro LEN debe tener el valor "0".
   Una vez finalizada correctamente la recepción de datos, el estado lógico en el parámetro DONE es "1". Si ocurren errores durante la transferencia de datos, el parámetro DONE se pone a "0".
3. **Deshacer la conexión:**
   La conexión se deshace inmediatamente cuando el parámetro CONT se pone a "0".

Al activar el parámetro COM_RST se vuelve a ejecutar "TRCV_C". Con ello, la conexión existente se deshace y se establece una nueva. Si se están recibiendo datos al reiniciarse la instrucción, podrían perderse datos.

Modos de recepción de TRCV_C

La tabla siguiente muestra cómo se escriben los datos recibidos en el área de recepción.

Variante de protocolo	Disponibilidad de los datos en el área de recepción	Parámetro connection_type de la descripción de la conexión	Parámetro LEN
TCP (Modo Ad hoc)	Los datos están disponibles inmediatamente.	Valor hexadecimal: B#16#11 Valor entero: 17	0
TCP (Recepción de datos de longitud especificada)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#11 Valor entero: 17	de 1 a 8192
ISO on TCP (Transmisión de datos orientada a mensajes)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#12 Valor entero: 18	• de 1 a 1452 si se utiliza un CP. • de 1 a 8192 si no se utiliza ningún CP.

TCP (Modo Ad hoc)

El modo Ad hoc sólo está disponible en la variante de protocolo TCP. Utilice el modo Ad hoc para recibir datos de longitudes dinámicas con la instrucción "TRCV".

El modo Ad hoc se establece asignando el valor "0" al parámetro LEN. Con el modo Ad hoc se pueden utilizar todos los tipos de datos para bloques de datos de acceso estándar. Para bloques de datos de acceso optimizado solo es posible utilizar el tipo de datos ARRAY of BYTE o tipos de datos de una longitud de 8 bits (p. ej., CHAR, USINT, SINT, etc.). La longitud de datos realmente recibida se visualiza en el parámetro RCVD_LEN.

TCP (recepción de datos de longitud especificada)

La longitud para la recepción de datos se predetermina con el valor del parámetro LEN. La recepción de datos finalizará cuando se haya recibido la longitud de los datos indicados en el parámetro LEN en su totalidad. Solo entonces estarán disponibles los datos en el área de recepción (parámetro DATA). Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

ISO on TCP (Transmisión de datos orientada a mensajes)

En la variante de protocolo ISO on TCP se envían a través de una conexión bloques cerrados de mensajes que el receptor también reconoce como tales. Si se utiliza ISO on TCP, "TRCV_C" notifica la recepción de los datos en cuanto se recibe el bloque de mensajes en su totalidad. Los parámetros LEN y DATA definen el área de recepción. Si el búfer de recepción (parámetro DATA) es demasiado pequeño para los datos enviados, "TRCV_C" notifica un error. Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TRCV_C":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	Habilitación de recepción
CONT	Input	BOOL	I, Q, M, D, L	Controla la conexión: • 0: Deshacer la conexión • 1: Establecer y mantener la conexión Al recibir datos (flanco ascendente en el parámetro EN_R), el parámetro CONT debe tener el valor TRUE para establecer o mantener una conexión.
LEN	Input	UINT	I, Q, M, D, L o constante	Longitud máxima de los datos que hay que recibir. Si utiliza únicamente valores simbólicos en el parámetro DATA, el parámetro LEN debe tener el valor "0".
CONNECT	InOut	TCON_Param	D	Puntero hacia la descripción de la conexión Consulte también: Auto-Hotspot
DATA	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el área de recepción
COM_RST	InOut	BOOL	I, Q, M, D, L	Reinicia la instrucción: • 0: Irrelevante • 1: Reinicio completo de la instrucción, con lo que se deshace la conexión existente
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No se puede iniciar una nueva petición
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción
RCVD_LEN	Output	UDINT	I, Q, M, D, L	Cantidad de datos (en bytes) recibida realmente

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TRCV_C". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	-	-	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Descripción
0	0000	Petición finalizada sin errores.
0	0001	El establecimiento de la conexión ha finalizado.
0	0003	La desconexión ha finalizado.
0	7000	No se está procesando ninguna petición.
0	7001	• Iniciar procesamiento de la petición • Establecer conexión • Esperar al interlocutor
0	7002	Recibiendo datos.
0	7003	Se deshace la conexión
0	7004	• Conexión establecida y vigilada • No se está procesando ninguna petición
0	7006	Se están recibiendo datos.
1	8085	• El parámetro LEN excede el valor máximo admisible. • El valor del parámetro LEN o DATA ha sido modificado después de la primera llamada.
1	8086	El parámetro ID se encuentra fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones
1	8088	El valor del parámetro LEN no es compatible con el área de recepción indicada en el parámetro DATA.
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	8091	Se ha excedido la profundidad de anidamiento máxima.
1	809A	El parámetro CONNECT apunta a un campo que no concuerda con la longitud de la descripción de la conexión.
1	809B	La ID del dispositivo local (local_device_id) indicado en la descripción de la conexión no concuerda con la CPU.
1	80A0	Error de grupo para los códigos de error W#16#80A1 y W#16#80A2.
1	80A1	• El usuario ya está utilizando la conexión o el puerto. • Error de comunicación: – La conexión indicada no se ha establecido todavía. – La conexión indicada se está finalizando. No es posible transferir a través de esta conexión. – La interfaz se está reinicializando.

ERROR	STATUS* (W#16#...)	Descripción
1	80A2	El sistema está utilizando el puerto local o remoto.
1	80A3	• Se está intentando volver a establecer una conexión existente. • Se está intentando finalizar una conexión no existente.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida, es decir, concuerda con la dirección IP del interlocutor local.
1	80A7	Error de comunicación: Se ha llamado la instrucción con COM_RST = 1 antes de finalizar la petición de transmisión.
1	80B2	El parámetro CONNECT apunta a un bloque de datos que ha sido generado con el atributo "Guardar sólo en la memoria de carga".
1	80B3	Parametrización incoherente: Error de grupo para los códigos de error W#16#80A0 a W#16#80A2, W#16#80A4, W#16#80B4 a W#16#80B9.
1	80B4	Al establecer una conexión pasiva (active_est = FALSE) con la variante de protocolo ISO on TCP (connection_type = B#16#12) se han infringido una o ambas de las condiciones siguientes: "local_tsap_id_len >= B#16#02" y/o "local_tsap_id[1] = B#16#E0".
1	80B5	Con el tipo de conexión 13 = UDP sólo se permite un establecimiento pasivo de la conexión.
1	80B6	Error de parametrización en el parámetro connection_type del bloque de datos para la descripción de la conexión.
1	80B7	Error en uno de los parámetros siguientes del bloque de datos para la descripción de la conexión: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	80C3	• Todos los recursos de conexión están ocupados. • Ya se está procesando un bloque con esta ID en otro grupo de prioridad.
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos parámetros o la conexión se está estableciendo. • Una instrucción "TDISCON" está deshaciendo la conexión configurada. • La conexión utilizada está siendo finalizada por una llamada con COM_RST= 1
1	80C6	No es posible acceder al interlocutor remoto (error de red).
1	8722	Error en el parámetro CONNECT: Rango de origen no válido (el rango no está declarado en el bloque de datos).
1	873A	Error en el parámetro CONNECT: El acceso a la descripción de la conexión no es posible (no hay acceso al bloque de datos).
1	877F	Error en el parámetro CONNECT: Error interno
1	8922	Parámetro DATA: El rango de destino no es válido, el rango no existe en el DB.
1	8924	Parámetro DATA: Error de área en el puntero VARIANT.
1	8932	Parámetro DATA: El número de DB es demasiado alto.
1	893A	Parámetro CONNECT: Imposible acceder a los datos de conexión indicados (p. ej., porque el DB no está disponible).
1	897F	Parámetro DATA: Error interno, p. ej., referencia VARIANT no admisible.
1	8A3A	Parámetro DATA: Imposible acceder al área de datos, p. ej., porque el bloque de datos no existe.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Nota

Mensajes de error de las instrucciones "TCON", "TRCV" y "TDISCON"

La instrucción "TRV_C" utiliza internamente las instrucciones "TCON (Página 5298)", "TRCV (Página 5314)" y "TDISCON (Página 5305)". Los mensajes de error de estas instrucciones están incluidos en las descripciones correspondientes.

TRCV_C: Recibir datos a través de Ethernet

Descripción

La instrucción "TRCV_C" se ejecuta de forma asíncrona y ejecuta por orden las funciones siguientes:

- Configurar y establecer una conexión
- Recibir datos a través de la conexión existente
- Deshacer o inicializar la conexión

La instrucción "TRCV_C" utiliza internamente las instrucciones de comunicación "TCON", "TRCV", "T_DIAG", "T_RESET" y "TDISCON".

Configurar y establecer una conexión

La conexión de comunicación se crea y establece con CONT=1. El número de conexiones posibles se indica en los datos técnicos de la CPU. Para configurar la conexión se utiliza la descripción de la conexión especificada en el parámetro CONNECT. Existen los siguientes tipos de conexión:

- Conexiones programadas (establecimiento de la conexión con "TCON"):
  - TCP/UDP: Descripción de la conexión mediante el tipo de datos de sistema TCON_IP_v4
  - ISO on TCP: Descripción de la conexión mediante el tipo de datos del sistema TCON_IP_RFC
  - ISO: Descripción de la conexión mediante el tipo de datos de sistema TCON_ISOnative (solo con CP1543-1)
- Conexiones configuradas
  - Especificación de una conexión existente en el tipo de datos del sistema TCON_Configured.

Si la CPU pasa al estado operativo STOP, se interrumpe una conexión existente y se elimina la conexión creada. Para volver a configurar y establecer la conexión, es preciso volver a ejecutar "TRCV_C".

Recibir datos a través de la conexión existente

Cuando el parámetro EN_R se pone a "1", se activa la recepción de datos. Los datos recibidos se escriben en un área de recepción. La longitud del área de recepción se especifica, en función de la variante de protocolo utilizada, bien sea con el parámetro LEN (si LEN <> 0), o bien mediante la indicación de longitud del parámetro DATA (si LEN = 0). Si utiliza únicamente valores simbólicos en el parámetro DATA, para el parámetro LEN debe utilizarse el valor "0".

Modos de recepción de TRCV_C:

- TCP (Modo Ad hoc)
  El modo Ad hoc sólo está disponible en la variante de protocolo TCP. Utilice el modo Ad hoc para recibir datos de longitudes dinámicas con la instrucción "TRCV_C".
  El modo Ad hoc se establece asignando el valor "1" al parámetro ADHOC. Con el modo Ad hoc se pueden utilizar todos los tipos de datos para bloques de datos de acceso estándar. Para bloques de datos de acceso optimizado solo es posible utilizar el tipo de datos ARRAY of BYTE o tipos de datos de una longitud de 8 bits (p. ej., CHAR, USINT, SINT, etc.). La longitud de datos realmente recibida se visualiza en el parámetro RCVD_LEN.
- TCP (Recepción de datos de longitud especificada)
  Para una recepción de datos con longitud indicada, asigne el valor "0" al parámetro ADHOC. Si el modo Ad hoc está desactivado, la recepción de datos finalizará solo cuando se haya recibido la longitud de los datos indicada en el parámetro LEN en su totalidad. Solo entonces estarán disponibles los datos en el área de recepción (parámetro DATA). Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.
- ISO on TCP (Transmisión de datos orientada a mensajes)
  En la variante de protocolo ISO on TCP se envían a través de una conexión bloques cerrados de mensajes que el receptor también reconoce como tales. Los parámetros LEN y DATA definen el área de recepción. Si el búfer de recepción (parámetro DATA) es demasiado pequeño para los datos enviados, "TRCV_C" notifica un error. Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

La tabla siguiente muestra cómo se escriben los datos recibidos en el área de recepción.

Variante de protocolo	Disponibilidad de los datos en el área de recepción	Parámetro connection_type de la descripción de la conexión	Parámetro LEN
TCP (Modo Ad hoc)	Los datos están disponibles inmediatamente.	Valor hexadecimal: B#16#11 Valor entero: 17	1 hasta longitud máxima (depende de CPU)
TCP (Recepción de datos de longitud especificada)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#11 Valor entero: 17	de 1 a 8192
ISO on TCP (Transmisión de datos orientada a mensajes)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#12 Valor entero: 18	de 1 a 8192

Deshacer la conexión

La conexión se deshace cuando el parámetro CONT se pone al valor "0" aunque no haya finalizado aún una transferencia de datos en curso. Esto no es aplicable si se utiliza una conexión configurada.

Ajustando el parámetro COM_RST a "1" se puede resetear la conexión en cualquier momento. Con ello, la conexión existente se deshace y se establece una nueva. Si se están transfiriendo datos en ese momento, podrían perderse.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TRCV_C":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L, T, C o constante	Habilitación de recepción
CONT	Input	BOOL	I, Q, M, D, L	Controla la conexión: • 0: Deshacer la conexión. • 1: Establecer la conexión y mantenerla tras recibir los datos.
LEN	Input	UDINT	I, Q, M, D, L o constante	Longitud máxima de los datos que hay que recibir. Si utiliza un área de recepción de acceso optimizado en el parámetro DATA, el parámetro LEN debe tener el valor "0".
ADHOC	Input	BOOL	I, Q, M, D, L o constante	Parámetro opcional (oculto) Utilizar el modo Ad hoc para la variante de protocolo TCP.
CONNECT	InOut	VARIANT	D	Puntero hacia la descripción de la conexión • Conexión programada: – Con TCP o UDP, utilice la estructura TCON_IP_v4 Consulte la descripción en: Auto-Hotspot – Con ISO-on-TCP, utilice la estructura TCON_IP_RFC Consulte la descripción en: Auto-Hotspot – Con ISO, utilice la estructura TCON_ISOnative (solo con CP1543-1) Consulte la descripción en la instrucción "TCON (Página 5301)": "Estructura de la descripción de la conexión conforme a TCON_ISOnative" • Conexión configurada: – Para conexiones existentes, utilice el tipo de datos de sistema TCON_Configured. Consulte la descripción más abajo, en "Tipo de datos de sistema para conexiones configuradas".
DATA	InOut	VARIANT	I, Q, M, D, L	Puntero hacia el área de recepción.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ADDR	InOut	VARIANT	D	Parámetro opcional (oculto) Puntero hacia la dirección del emisor con el tipo de conexión UDP.
COM_RST	InOut	BOOL	I, Q, M, D, L	Parámetro opcional (oculto) Efectúa la inicialización de la conexión: • 0: Irrelevante • 1: Se inicializa la conexión existente. El parámetro COM_RST se restablece tras la evaluación mediante la instrucción "TRCV_C" y, por tanto, no debe interconectarse estáticamente.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Recepción no iniciada aún o en ejecución. • 1: Recepción ejecutada sin errores. Este estado solo se muestra durante un ciclo. El parámetro de salida DONE se activa cuando finaliza correctamente un paso intermedio del procesamiento (establecer conexión, recepción, deshacer conexión) y cuando la ejecución de "TRCV_C" finaliza correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Recepción no iniciada aún o ya finalizada. • 1: Recepción no finalizada todavía. No es posible iniciar una nueva petición de transmisión.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Ningún error • 1: Se ha producido un error al establecer la conexión, al recibir los datos o al deshacer la conexión. El parámetro de salida ERROR puede activarse debido a un error en la instrucción "TRCV_C" o en las instrucciones de comunicación utilizadas internamente.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción
RCVD_LEN	Output	UDINT	I, Q, M, D, L	Cantidad de datos (en bytes) recibida realmente

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros EN_R, CONT y COM_RST

El parámetro CONT controla el establecimiento de conexión de la instrucción "TRCV_C", independientemente del parámetro EN_R. El comportamiento del parámetro CONT depende en parte de si se utiliza una conexión programada o configurada:

- Con CONT = "0": No se reciben datos (independientemente de si se utiliza una conexión programada o configurada).
- Al cambiar de CONT = "0" a "1":
  - En el caso de una conexión programada, esta se establece con "TCON".
  - En el caso de una conexión configurada, esta se comprueba con "T_DIAG".
- Con CONT = "1":
  - Mientras no se reciba ningún dato (EN_R ="0"), se comprueba la conexión con "T_DIAG".
  - Si las instrucciones de comunicación internas notifican que no hay ningún punto final de la conexión, esta se restablecerá automáticamente mediante "TCON".
- Al cambiar de CONT = "1" a "0":
  - En el caso de una conexión programada, esta se deshace con "TDISCON".
  - En el caso de una conexión configurada, esta se inicializa con "T_RESET".

El parámetro COM_RST inicializa la conexión al cambiar de "0" a "1":

- Si hay una conexión establecida, esta se inicializa mediante "T_RESET" (independientemente de si se utiliza una conexión programada o configurada).
- Si no hay ninguna conexión establecida, la activación del parámetro no tiene efecto.

Los parámetros EN_R y COM_RST solo se hacen efectivos si se ha puesto CONT a "1". La tabla siguiente muestra la relación entre los parámetros EN_R, CONT y COM_RST:

EN_R	CONT	COM_RST	Estado de la instrucción	Descripción
Irrelevante	0	Irrelevante	No se ha ejecutado aún	Ninguna petición activa (STATUS = 7000).
Irrelevante	0	Irrelevante	Inicialización	Se está deshaciendo la conexión. La instrucción se inicializa.
Irrelevante	0 > 1	Irrelevante	Establecimiento de la conexión	La conexión se está estableciendo. Todavía no se transfieren datos.
0	1	0	Conexión establecida	La conexión se ha establecido y se vigila con la instrucción "T_DIAG".
Irrelevante	1	0 > 1	Conexión establecida	La conexión se interrumpe brevemente y se inicializa con "T_RESET".
0 > 1	1	0	Conexión establecida	La instrucción empieza a recibir.
Irrelevante	1	0 > 1	Recibiendo datos	La transferencia de datos se interrumpe. La conexión se inicializa.

Tipo de datos de sistema para conexiones configuradas

Para conexiones configuradas, utilice en el parámetro CONNECT la siguiente estructura para la descripción de la conexión según TCON_Configured:

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceID	HW_ANY	-	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	-	Referencia a la conexión (rango de valores: de 1 a 4095). Indique la ID de la conexión existente.
4	ConnectionType	BYTE	-	Tipo de conexión Seleccione 254 (decimal) para una conexión configurada.

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TRCV_C". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

DONE	BUSY	ERROR	Descripción
0	0	0	La instrucción aún no se ha ejecutado (no hay ningún flanco ascendente en el parámetro EN_R).
0	1	0	La instrucción se ejecuta y llama las instrucciones de comunicación utilizadas internamente.
1	0	0	La recepción se ha ejecutado correctamente. En el parámetro STATUS se devuelve "0000". DONE = "1" solo se muestra durante un ciclo.
0	0	1	La ejecución de la instrucción o un paso intermedio del procesamiento ha finalizado con un error. Si se produce un error como consecuencia de una instrucción de comunicación utilizada internamente, se indica el error que se ha producido en primer lugar durante el procesamiento. Este estado solo se muestra durante un ciclo.

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Descripción
0	0000	Petición de recepción ejecutada sin errores.
0	0001	Comunicación establecida.
0	0003	Comunicación desconectada.
0	7000	No se está procesando ninguna petición.
0	7001	Primera llamada al establecer una conexión.
0	7002	Segunda llamada al establecer una conexión.
0	7003	Se está deshaciendo la conexión.

ERROR	STATUS (W#16#...)	Descripción
0	7004	Conexión establecida y vigilada. No se está procesando ninguna petición de recepción.
0	7006	Se están recibiendo datos.
1	8085	• El parámetro LEN excede el valor máximo admisible. • El valor del parámetro LEN o DATA ha sido modificado después de la primera llamada.
1	8086	El parámetro ID se encuentra fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones
1	8088	El valor del parámetro LEN no es compatible con el área de recepción indicada en el parámetro DATA.
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	8091	Se ha excedido la profundidad de anidamiento máxima.
1	809A	El parámetro CONNECT apunta a un campo que no concuerda con la longitud de la descripción de la conexión.
1	809B	InterfaceID no es válida. Es cero o no apunta a una interfaz de CPU local o un CP.
1	80A1	• El usuario ya está utilizando la conexión o el puerto. • Error de comunicación: – La conexión indicada no se ha establecido todavía. – La conexión indicada se está finalizando. No es posible transferir a través de esta conexión. – La interfaz se está reinicializando.
1	80A3	La instrucción "T_DIAG" subordinada ha notificado que se ha deshecho la conexión.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida o bien concuerda con la dirección IP del interlocutor local.
1	80A7	Error de comunicación: Se ha llamado la instrucción con COM_RST = 1 antes de finalizar la petición de transmisión.
1	80B4	Al establecer una conexión pasiva (active_est = FALSE) con la variante de protocolo ISO on TCP (connection_type = B#16#12) se han infringido una o ambas de las condiciones siguientes: "local_tsap_id_len >= B#16#02" y/o "local_tsap_id[1] = B#16#E0".
1	80B5	Con el tipo de conexión 13 = UDP sólo se permite un establecimiento pasivo de la conexión.
1	80B6	Error de parametrización en el parámetro connection_type del bloque de datos para la descripción de la conexión.
1	80B7	Error en uno de los parámetros siguientes del bloque de datos para la descripción de la conexión: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	80C3	• Todos los recursos de conexión están ocupados. • Ya se está procesando un bloque con esta ID en otro grupo de prioridad.
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos parámetros o la conexión se está estableciendo. • Una instrucción "TDISCON" está deshaciendo la conexión configurada. • La conexión utilizada está siendo finalizada por una llamada con COM_RST = 1
1	80C6	No es posible acceder al interlocutor remoto (error de red).
1	8722	Error en el parámetro CONNECT: Rango de origen no válido (el rango no está declarado en el bloque de datos).
1	873A	Error en el parámetro CONNECT: El acceso a la descripción de la conexión no es posible (no hay acceso al bloque de datos).

ERROR	STATUS (W#16#...)	Descripción
1	877F	Error en el parámetro CONNECT: Error interno
1	8922	Parámetro DATA: El rango de destino no es válido, el rango no existe en el DB.
1	8924	Parámetro DATA: Error de área en el puntero VARIANT.
1	8932	Parámetro DATA: El número de DB es demasiado alto.
1	893A	Parámetro CONNECT: Imposible acceder a los datos de conexión indicados (p. ej., porque el DB no está disponible).
1	897F	Parámetro DATA: Error interno, p. ej., referencia VARIANT no admisible.
1	8A3A	Parámetro DATA: Imposible acceder al área de datos, p. ej., porque el bloque de datos no existe.

Nota

Mensajes de error de las instrucciones "TCON", "TRCV" y "TDISCON"

La instrucción "TRV_C" utiliza internamente las instrucciones "TCON (Página 5301)", "TRCV (Página 5314)" y "TDISCON (Página 5305)". Los mensajes de error de estas instrucciones están incluidos en las descripciones correspondientes.

Consulte también

TSEND_C: Enviar datos a través de Ethernet (Página 5267)

TMAIL_C: Transmitir un mensaje de correo electrónico

Descripción de TMAIL_C

Descripción

La instrucción "TMAIL_C" envía un mensaje de correo electrónico a través de la interfaz Ethernet del S7-1500 o S7-1200 V4.0, de un módulo de comunicaciones (CM) o de un procesador de comunicaciones (CP).

Para utilizar la instrucción, es imprescindible haber configurado previamente el hardware y que la infraestructura de la red permita establecer una conexión con el servidor de correo electrónico.

El contenido del mensaje de correo electrónico y los datos de conexión se definen mediante los siguientes parámetros:

- Las direcciones de los destinatarios se definen a través de los parámetros TO_S y CC.
- El contenido del mensaje de correo electrónico se define a través de los parámetros SUBJECT y TEXT.
- Puede definirse un archivo adjunto con el puntero VARIANT en los parámetros ATTACHMENT y ATTACHMENT_NAME.

- Los datos de conexión, al igual que el direccionamiento y la autenticación para el servidor de correo electrónico, se definen mediante el tipo de datos de sistema Tmail_v4, Tmail_v6 o Tmail_FQDN del parámetro MAIL_ADDR_PARAM.
  - Si se utiliza la interfaz de la CPU S7-1500, únicamente se emplea el tipo de datos de sistema Tmail_v4. En este caso, el envío del mensaje de correo electrónico solo es posible a través de SMTP.
  - Si se utiliza la interfaz de un CM/CP, es posible emplear todos los tipos de datos de sistema. El envío del mensaje de correo electrónico también es posible a través de SMTPS.
- El envío de un mensaje de correo electrónico se inicia con un cambio de flanco de "0" a "1" en el parámetro REQ.
- A través de los parámetros de salida "BUSY", "DONE", "ERROR" y "STATUS" se indica el estado de la petición.

No es posible enviar directamente un SMS a través de la instrucción "TMAIL_C". La posibilidad de reenviar el mensaje de correo electrónico en forma de SMS a través del servidor de correo depende del proveedor de los servicios de telecomunicaciones.

Funcionamiento de la instrucción

La instrucción "TMAIL_C" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Al llamar la instrucción "TMAIL_C" es imprescindible indicar una instancia.

La conexión con el servidor de correo electrónico se interrumpe en los siguientes casos:

- Si la CPU pasa al estado operativo STOP mientras "TMAIL_C" está activo.
- Si aparecen problemas de comunicación en el bus Industrial Ethernet.

En esos casos, el envío del mensaje de correo electrónico se cancela y este no llega al receptor. La conexión también se interrumpe tras ejecutar correctamente la instrucción y enviar el mensaje de correo electrónico.

ATENCIÓN
Cambiar programas de usuario Las partes del programa de usuario con efecto directo sobre llamadas de "TMAIL_C" solo pueden modificarse en los siguientes casos: • Si la CPU está en estado operativo "STOP". • Si no se envía ningún mensaje (REQ = 0 y BUSY = 0). Esto se refiere en especial al borrado e intercambio de bloques de programa que contienen llamadas de "TMAIL_C" o llamadas de la instancia de "TMAIL_C" En caso de incumplimiento, es posible que los recursos de conexión permanezcan ocupados. El sistema de automatización puede pasar a un estado no definido con las funciones de comunicación TCP/IP a través de Industrial Ethernet. Una vez transmitidas las modificaciones es necesario ejecutar un rearranque completo (en caliente) o un arranque en frío de la CPU.

Coherencia de datos

Los parámetros TO_S, CC, SUBJECT, TEXT, ATTACHMENT y MAIL_ADDR_PARAM son aplicados durante el funcionamiento de la instrucción "TMAIL_C", por lo que no deben modificarse hasta que haya finalizado la petición (BUSY = 0).

Autenticación SMTP

Bajo el término "autenticación" se entiende un proceso con el que se garantiza una identidad, p. ej., una consulta de contraseña.

Si se utiliza la interfaz de la CPU S7-1500, la instrucción "TMAIL_C" soporta el método de autenticación SMTP AUTH-LOGIN, requerido por la mayoría de servidores de correo. Para más información sobre el método de autenticación de su servidor de correo, consulte el manual del servidor o la página web del proveedor de servicios de Internet.

- Para utilizar el método de autenticación AUTH-LOGIN, la instrucción "TMAIL_C" requiere el nombre de usuario con el que poder iniciar sesión en el servidor de correo. Este nombre de usuario equivale al nombre de usuario con el que se ha configurado la cuenta de correo en el servidor de correo. Se transfiere a la estructura del parámetro MAIL_ADDR_PARAM a través del parámetro UserName.
  Si en la estructura del parámetro MAIL_ADDR_PARAM no se ha especificado ningún nombre de usuario, no se hace uso del método de autenticación AUTH-LOGIN. El mensaje de correo electrónico se envía entonces sin autenticación.
- Asimismo, para iniciar sesión la instrucción "TMAIL_C" requiere la contraseña correspondiente. Esta contraseña equivale a la contraseña indicada al configurar la cuenta de correo. Se transfiere a la estructura del parámetro MAIL_ADDR_PARAM a través del parámetro PassWord.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TMAIL_C":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C o constante	Parámetro de control REQUEST: Activa el envío de un mensaje de correo electrónico con un flanco ascendente.
TO_S (Página 5290)	Input	STRING	D, L o constante	Direcciones de destinatario STRING con una longitud máxima de 240 caracteres (byte). Respecto a la notación de la dirección de correo electrónico, véase el ejemplo en la descripción del parámetro.
CC (Página 5290)	Input	STRING	D, L o constante	Direcciones de destinatario CC (opcional) STRING con una longitud máxima de 240 caracteres (byte). Notación de la dirección de correo electrónico idéntica a la del parámetro TO_S. Si se asigna un string vacío, el mensaje no se envía a un destinatario CC.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SUBJECT	Input	STRING	D, L o constante	Asunto del mensaje STRING con una longitud máxima de 240 caracteres (byte).
TEXT	Input	STRING	D, L o constante	Texto del mensaje (opcional) STRING con una longitud máxima de 240 caracteres (byte). Si a este parámetro se le asigna un string vacío, el mensaje se envía sin texto.
ATTACHMENT	Input	VARIANT	D	Archivos adjuntos del mensaje (opcional) Referencia a un campo de byte/palabra/palabra doble (ArrayOfByte, ArrayOfWord o ArrayOfDWord) con una longitud máxima de 64 Kbytes. Si no se asigna ningún valor, el mensaje se envía sin archivos adjuntos.
ATTACHMENT_NAME	Input	STRING	D, L o constante	Nombre de los archivos adjuntos del mensaje (opcional) Referencia a una cadena de caracteres con una longitud máxima de 50 caracteres (byte) para definir el nombre del archivo adjunto. Si a este parámetro se le asigna un string vacío, el archivo adjunto del mensaje se envía con el nombre "attachment.bin".
MAIL_ADDR_PARAM (Página 5290)	Input	VARIANT	D	Parámetros de la conexión y dirección del servidor de correo Para definir los parámetros de conexión deben utilizarse las estructuras Tmail_v4, Tmail_v6 o Tmail_FQDN (véase la descripción del parámetro).
DONE (Página 5293)	Output	BOOL	I, Q, M, D, L	Parámetro de estado • DONE = 0: la petición aún no se ha iniciado o todavía se está ejecutando. • DONE = 1: Petición finalizada sin errores.
BUSY (Página 5293)	Output	BOOL	I, Q, M, D, L	Parámetro de estado • BUSY = 0: la ejecución de "TMAIL_C" ha finalizado. • BUSY = 1: el envío del mensaje de correo electrónico todavía no ha finalizado.
ERROR (Página 5293)	Output	BOOL	I, Q, M, D, L	Parámetro de estado • ERROR = 0: no se han producido errores. • ERROR = 1: durante la ejecución se ha producido un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS (Página 5294)	Output	WORD	I, Q, M, D, L	Parámetro de estado Valor de retorno o información de error de la instrucción "TMAIL_C" (véase la descripción del parámetro).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Parámetros opcionales

Los parámetros opcionales CC, TEXT y ATTACHMENT solo se envían con el mensaje de correo electrónico si los parámetros correspondientes contienen un string de longitud > 0.

Parámetros TO_S y CC

Descripción

Los parámetros TO_S y CC son strings que tienen, por ejemplo, el siguiente contenido:

- <wenna@mydomain.com>, <ruby@mydomain.com>
- <admin@mydomain.com>, <judy@mydomain.com>

Las siguientes reglas deben observarse a la hora de introducir los parámetros:

- Antes de cada dirección debe introducirse un espacio y un corchete en "<".
- Después de cada dirección debe introducirse un corchete en ">".
- Entre las direcciones indicadas en TO y CC debe introducirse una coma.

Por motivos relacionados con el tiempo de ejecución y el espacio en memoria, la instrucción "TMAIL_C" no realiza ninguna prueba de sintaxis de los parámetros TO_S y CC.

Parámetro MAIL_ADDR_PARAM

Descripción

En las estructuras Tmail_v4, Tmail_v6 o Tmail_FQDN del parámetro MAIL_ADDR_PARAM se define a través de qué conexión debe enviarse el mensaje de correo electrónico y se guardan tanto la dirección del servidor de correo como los datos de inicio de sesión.

En función del formato con que desee direccionar el servidor de correo, utilice en el parámetro MAIL_ADDR_PARAM la estructura:

- Tmail_v4: direccionamiento a través de la dirección IP según IPv4.
- Tmail_v6: direccionamiento a través de la dirección IP según IPv6.
- Tmail_FQDN: direccionamiento a través de un nombre de dominio totalmente calificado (FQDN).

El tipo de estructura que puede utilizarse depende de la interfaz direccionada en el parámetro InterfaceId:

- Para utilizar la instrucción "TMAIL_C" con la interfaz interna, en el parámetro MAIL_ADDR_PARAM debe emplearse la estructura Tmail_v4.
- Si se utiliza un procesador de comunicaciones (CP) o un módulo de comunicación (CM), pueden emplearse las tres posibilidades de direccionamiento (IPv4, IPv6 y FQDN).

Tabla 9-94 **Tmail_v4: direccionamiento del servidor de correo a través de la dirección IP según IPv4**

Parámetro	Tipo de datos	Descripción
Tmail_v4	Struct	
InterfaceId	LADDR	ID de hardware de la interfaz
ID	CONN_OUC	ID de conexión
ConnectionType	BYTE	Tipo de conexión. Seleccione 16#20 como tipo de conexión para IPv4.
ActiveEstablished	BOOL	Bit de estado. Se pone a "1" cuando se ha establecido la conexión.
CertIndex	BYTE	• =0: utilización de SMTP (**S**imple **M**ail **T**ransfer **P**rotocol). Para enviar el mensaje de correo electrónico a través de la interfaz de una CPU S7-1500, debe utilizarse SMTP. • ≠0: utilización de SMTPS para proteger la conexión antes de establecerla (en CP/CM). El certificado que debe utilizarse se especifica en el parámetro CertIndex (véase Árbol del proyecto" > "Ajustes de seguridad global" > "Administrador de certificados").
WatchDogTime	TIME	Vigilancia de tiempo de la ejecución. A través de este parámetro se define la máxima duración para ejecutar el proceso de transmisión. Nota: con una conexión lenta, el establecimiento de la llamada puede requerir cierto tiempo (aprox. un minuto). El tiempo necesario para el establecimiento de la conexión debe tenerse en cuenta al especificar el parámetro WATCH_DOG_TIME. Una vez ha transcurrido el tiempo predefinido, la conexión finaliza.
MailServerAddress	IP_v4	Dirección IP del servidor de correo. Según IPv4 en formato XXX.XXX.XXX.XXX (decimal). Ejemplo: 192.142.131.237.
UserName	STRING[254]	Nombre de inicio de sesión del servidor de correo
PassWord	STRING[254]	Contraseña del servidor de correo
From	EMAIL_ADDR	Dirección de remitente del mensaje de correo electrónico que se define a través de los dos parámetros STRING siguientes. Por ejemplo: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Parte local de la dirección de remitente, incluido el carácter @. Ejemplo: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (abreviado: FQDN) del servidor de correo. Ejemplo: "mymailserver.com".

Tabla 9-95 **Tmail_v6: direccionamiento del servidor de correo a través de la dirección IP según IPv6**

Parámetro	Tipo de datos	Descripción
Tmail_v6	Struct	
InterfaceId	LADDR	ID de hardware de la interfaz
ID	CONN_OUC	ID de conexión
ConnectionType	BYTE	Tipo de conexión. Seleccione 16#21 como tipo de conexión para IPv6.
ActiveEstablished	BOOL	Bit de estado. Se pone a "1" cuando se ha establecido la conexión.
CertIndex	BYTE	• =0: utilización de SMTP (Simple Mail Transfer Protocol). Para enviar el mensaje de correo electrónico a través de la interfaz de una CPU S7-1500, debe utilizarse SMTP. • ≠0: utilización de SMTPS para proteger la conexión antes de establecerla (en CP/CM). El certificado que debe utilizarse se especifica en el parámetro CertIndex (véase Árbol del proyecto" > "Ajustes de seguridad global" > "Administrador de certificados").
WatchDogTime	TIME	Vigilancia de tiempo de la ejecución. A través de este parámetro se define la máxima duración para ejecutar el proceso de transmisión. Nota: con una conexión lenta, el establecimiento de la llamada puede requerir cierto tiempo (aprox. un minuto). El tiempo necesario para el establecimiento de la conexión debe tenerse en cuenta al especificar el parámetro WATCH_DOG_TIME. Una vez ha transcurrido el tiempo predefinido, la conexión finaliza.
MailServerAddress	IP_v6	Dirección IP del servidor de correo según IPv6 en formato XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX.XXXX (hexadecimal). La dirección se divide en 8 bloques de 2 bytes cada uno (en total, 16 bytes). Ejemplo: 2001:db8:1f11:08d3:290:27ff:0370:2093
UserName	STRING[254]	Nombre de inicio de sesión del servidor de correo
PassWord	STRING[254]	Contraseña del servidor de correo
From	EMAIL_ADDR	Dirección de remitente del mensaje de correo electrónico que se define a través de los dos parámetros STRING siguientes. Por ejemplo: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Parte local de la dirección de remitente, incluido el carácter @. Ejemplo: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (abreviado: FQDN) del servidor de correo. Ejemplo: "mymailserver.com".

Tabla 9-96 **Tmail_FQDN: direccionamiento del servidor de correo a través de FQDN**

Parámetro	Tipo de datos	Descripción
Tmail_v6	Struct	
Tmail_FQDN	LADDR	ID de hardware de la interfaz
ID	CONN_OUC	ID de conexión

Parámetro	Tipo de datos	Descripción
ConnectionType	BYTE	Tipo de conexión. Seleccione 16#22 como tipo de conexión para el FQDN.
ActiveEstablished	BOOL	Bit de estado. Se pone a "1" cuando se ha establecido la conexión.
CertIndex	BYTE	• =0: utilización de SMTP (Simple Mail Transfer Protocol). Para enviar el mensaje de correo electrónico a través de la interfaz de una CPU S7-1500, debe utilizarse SMTP. • ≠0: utilización de SMTPS para proteger la conexión antes de establecerla (en CP/CM). El certificado que debe utilizarse se especifica en el parámetro CertIndex (véase Árbol del proyecto" > "Ajustes de seguridad global" > "Administrador de certificados").
WatchDogTime	TIME	Vigilancia de tiempo de la ejecución. A través de este parámetro se define la máxima duración para ejecutar el proceso de transmisión. Nota: con una conexión lenta, el establecimiento de la llamada puede requerir cierto tiempo (aprox. un minuto). El tiempo necesario para el establecimiento de la conexión debe tenerse en cuenta al especificar el parámetro WATCH_DOG_TIME. Una vez ha transcurrido el tiempo predefinido, la conexión finaliza.
MailServerAddress	STRING[254]	FQDN (Fully Qualified Domain Name) del servidor de correo. El servidor de correo se direcciona a través del nombre completo del dominio. Ejemplo: "www.mymailserver.com.".
UserName	STRING[254]	Nombre de inicio de sesión del servidor de correo
PassWord	STRING[254]	Contraseña del servidor de correo
From	Struct	Dirección de remitente del mensaje de correo electrónico que se define a través de los dos parámetros STRING siguientes. Por ejemplo: "myname@mymailserver.com".
LocalPartPlusAtSign	STRING[64]	Parte local de la dirección de remitente, incluido el carácter @. Ejemplo: "myname@".
FullQualifiedDomainName	STRING[254]	Fully Qualified Domain Name (abreviado: FQDN) del servidor de correo. Ejemplo: "mymailserver.com".

Parámetros DONE, BUSY y ERROR

Descripción

Los parámetros de salida DONE, BUSY y ERROR solo se indican durante un ciclo si el estado del parámetro de salida BUSY cambia de "1" a "0".

En la tabla siguiente se muestra la correspondencia entre DONE, BUSY y ERROR. Permite determinar en qué estado se encuentra la instrucción "TMAIL_C" actualmente o cuándo se completó la transmisión del mensaje de correo electrónico.

DONE	BUSY	ERROR	Descripción
0	1	0	La petición está en proceso.
1	0	0	La petición ha finalizado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS (Página 5294).
0	0	0	No se ha asignado ninguna (nueva) petición a la instrucción "TMAIL_C".

Parámetro STATUS

Descripción

La tabla siguiente muestra los valores de retorno en el parámetro STATUS de "TMAIL_C":

Valor de retorno STATUS* (W#16#...):	Explicación	Indicaciones
0000	La ejecución de "TMAIL_C" ha finalizado sin errores.	La finalización sin errores de "TMAIL_C" no significa que el mensaje de correo electrónico enviado sea recibido. La entrada incorrecta de las direcciones de los destinatarios no genera un error de estado de la instrucción "TMAIL_C". En tal caso no es posible garantizar el envío del mensaje de correo electrónico a otros destinatarios, aunque se hayan introducido correctamente.
7001	"TMAIL_C" está activo (BUSY = 1).	Primera llamada: petición lanzada.
7002	"TMAIL_C" está activo (BUSY = 1).	Llamada intermedia: petición ya activa.
8xxx	La ejecución de "TMAIL_C" ha finalizado con un código de error de las instrucciones de comunicación llamadas internamente.	Encontrará información detallada en las descripciones del parámetro STATUS de las instrucciones de comunicación "TCON (Página 5301)", "TDISCON (Página 5305)", "TSEND (Página 5310)" y "TRCV (Página 5314)".
8010	Error al establecer la conexión	Encontrará más información sobre la evaluación en el parámetro SFB_STATUS del bloque de datos de instancia. El significado del código de error mostrado en SFB_STATUS puede consultarse en la descripción del parámetro STATUS de la instrucción "TCON (Página 5301)".

Valor de retorno STATUS* (W#16#...):	Explicación	Indicaciones
8011	Error al enviar los datos	Encontrará más información sobre la evaluación en el parámetro SFB_STATUS del bloque de datos de instancia. El significado del código de error mostrado en SFB_STATUS puede consultarse en la descripción del parámetro STATUS de la instrucción "TSEND (Página 5310)".
8012	Error al recibir los datos	Encontrará más información sobre la evaluación en el parámetro SFB_STATUS del bloque de datos de instancia. El significado del código de error mostrado en SFB_STATUS puede consultarse en la descripción del parámetro STATUS de la instrucción "TRCV (Página 5314)".
8013	Error al establecer la conexión	Encontrará más información sobre la evaluación en el parámetro SFB_STATUS del bloque de datos de instancia. El significado del código de error mostrado en SFB_STATUS puede consultarse en la descripción del parámetro STATUS de las instrucciones "TCON (Página 5301)" y "TDISCON (Página 5305)".
8014	No es posible establecer una conexión.	Posiblemente ha introducido una dirección IP del servidor de correo incorrecta (MailServerAddress (Página 5290)) o un intervalo demasiado corto (WatchDogTime (Página 5290)) para establecer la conexión. También existe la posibilidad de que la CPU no tenga conexión con la red o que la configuración de la CPU sea incorrecta.
8015	Tipo de datos incorrecto para MAIL_ADDR_PARAM	Los tipos de datos permitidos son los tipos de datos de sistema (estructuras) Tmail_v4, Tmail_v6 o TMail_FQDN.
8016	Tipo de datos incorrecto para el parámetro ATTACHMENT	Los tipos de datos permitidos son ArrayOfByte, ArrayOfWord y ArrayOfDWord.
8017	Longitud de datos incorrecta para el parámetro ATTACHMENT	La longitud de datos permitida es <= 65534 bytes.
8401	Ningún canal disponible. Causa posible: ya existe una conexión de correo electrónico a través del CP. No es posible establecer una segunda conexión paralela.	Error específico para el CP 1543
8403	No ha podido establecerse ninguna conexión TCP/IP con el servidor de correo.	Error específico para el CP 1543
8405	El servidor ha denegado la petición de iniciar sesión.	Error específico para el CP 1543

Valor de retorno STATUS* (W#16#...):	Explicación	Indicaciones
8406	El cliente SMTP ha detectado un error de SSL interno o un problema con la estructura del certificado.	Error específico para el CP 1543
8407	Se ha denegado la petición de utilizar SSL.	Error específico para el CP 1543
8408	El cliente no ha podido determinar ninguna toma de enchufe para establecer la conexión TCP/IP con el servidor de correo.	Error específico para el CP 1543
8409	No es posible escribir a través de la conexión. Causa posible: el interlocutor ha reseteado la conexión o esta se ha interrumpido.	Error específico para el CP 1543
8410	No es posible leer a través de la conexión. Causa posible: el interlocutor ha reseteado la conexión o esta se ha interrumpido.	Error específico para el CP 1543
8411	Error al enviar el mensaje de correo electrónico. Causa: el espacio de memoria no es suficiente para llevar a cabo el proceso de transmisión	Error específico para el CP 1543
8412	El servidor DNS configurado no ha podido deshacer el nombre de dominio indicado.	Error específico para el CP 1543
8413	El nombre de dominio no ha podido deshacerse a causa de un error interno en el subsistema DNS.	Error específico para el CP 1543
8414	Se ha indicado una cadena de caracteres vacía como nombre de dominio.	Error específico para el CP 1543
8415	Se ha producido un error interno en el módulo Curl. Se ha detenido la ejecución.	Error específico para el CP 1543
8416	Se ha producido un error interno en el módulo SMTP. Se ha detenido la ejecución.	Error específico para el CP 1543
8417	Petición a SMTP a través de un canal que ya se está utilizando o de una ID de canal no válida. Se ha detenido la ejecución.	Error específico para el CP 1543
8418	Envío del mensaje de correo electrónico interrumpido. Causas posibles: rebase del tiempo de ejecución (parámetro WatchDogTime) o de la transición arranque/parada del CP.	Error específico para el CP 1543

Valor de retorno STATUS* (W#16#...):	Explicación	Indicaciones
8419	El canal se ha interrumpido y no se puede utilizar antes de que la conexión se haya cerrado.	Error específico para el CP 1543
8420	La cadena de certificados del servidor no ha podido verificarse con el certificado raíz del CP.	Error específico para el CP 1543
8421	Se ha producido un error interno. Se ha detenido la ejecución.	Error específico para el CP 1543
82xx, 84xx, o bien 85xx	El mensaje de error procede del servidor de correo y, exceptuando el "8", corresponde al número de error del protocolo SMTP. En las siguientes filas se indican algunos códigos de error posibles.	Encontrará más información sobre el código de error SMTP y otros códigos de error en el protocolo SMTP en Internet, o bien en la documentación sobre errores del servidor de correo. También es posible consultar el último mensaje de error enviado por el servidor de correo en forma de texto en el DB de instancia en el parámetro BUFFER1. En el DB de instancia, en DATEN, encontrará los últimos datos enviados por la instrucción "TMAIL_C".
8450	Acción no ejecutada: Buzón de correo no disponible/no accesible	Vuelva a intentarlo más tarde.
8451	Acción cancelada: Error local en el procesamiento	Vuelva a intentarlo más tarde.
8500	Error de sintaxis: Error no detectado. Ello incluye también el error debido a una cadena de comandos demasiado larga. Otra posible causa es que el servidor de correo electrónico no soporte el método de autenticación LOGIN.	Compruebe los parámetros de "TMAIL_C". Intente enviar un mensaje de correo electrónico sin autenticación. Sustituya para ello el contenido del parámetro UserName por un string vacío. Si no se especifica un nombre de usuario, no se hace uso del método de autenticación LOGIN.
8501	Error de sintaxis: Entrada incorrecta en un parámetro	Causa posible: dirección incorrecta en los parámetros TO_S o CC (véase también: Parámetros TO_S y CC (Página 5290)).
8502	Comando desconocido o no implementado	Compruebe las entradas realizadas, especialmente el parámetro FROM. Es posible que esté incompleto y que haya olvidado "@" o "." (véase también: Parámetros TO_S y CC (Página 5290)).
8504	Parámetro de petición no implementado	Compruebe las entradas realizadas, especialmente el parámetro FROM. Es posible que esté incompleto y que haya olvidado "@" (ver también: Parámetros TO_S y CC (Página 5290)).
8535	Autenticación SMTP incompleta	Puede que haya introducido un nombre de usuario o una contraseña incorrectas.

Valor de retorno STATUS* (W#16#...):	Explicación	Indicaciones
8550	No es posible acceder al servidor de correo. No dispone de derechos de acceso.	Posiblemente ha introducido un nombre de usuario o una contraseña incorrectos o el servidor de correo no soporta el Login. Otra posible causa del error puede ser la entrada incorrecta del nombre del dominio después de "@" o que falte un "." en los parámetros TO_S, CC y FROM (ver también: Parámetros TO_S y CC (Página 5290)).
8552	Acción cancelada: Rebase del espacio de memoria asignado	Vuelva a intentarlo más tarde.
8554	Error al realizar la transmisión.	Vuelva a intentarlo más tarde.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Otros

TCON: Establecer conexión

TCON: Establecer una conexión

Descripción

La instrucción "TCON" permite configurar y establecer una conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente. "TCON" se ejecuta de forma asíncrona.

Para configurar la conexión se utilizan los datos de conexión especificados en los parámetros CONNECT e ID. Para poder establecer la conexión debe detectarse un flanco ascendente en el parámetro REQ. Si la conexión se establece correctamente, el parámetro DONE se pone a "1".

Número de conexiones posibles

El número de conexiones posibles se indica en los datos técnicos de la CPU.

Conexión vía TCP e ISO on TCP

Ambos interlocutores llaman la instrucción "TCON" para crear y establecer la conexión. En la parametrización se especifica cuáles son el punto final activo y el punto final pasivo de la comunicación.

Si se interrumpe la conexión, p. ej. debido a una interrupción de la línea o por el interlocutor remoto, el interlocutor activo intenta restablecer la conexión. No es preciso volver a llamar

"TCON". No obstante, esto solo es aplicable si "TCON" se ha ejecutado una vez correctamente (DONE = 1).

Con la ejecución de la instrucción "TDISCON (Página 5305)" o en el estado operativo STOP de la CPU se deshace una conexión existente y se elimina la conexión configurada. Para volver a configurar y establecer la conexión es preciso ejecutar "TCON" de nuevo.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "TCON":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Inicia la petición de establecer la conexión especificada en ID cuando se detecta un flanco ascendente.
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión asignada. Rango de valores: W#16#0001 a W#16#0FFF
CONNECT	InOut	TCON_Param	D	Puntero hacia la descripción de la conexión Consulte también: Auto-Hotspot
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No se puede iniciar una nueva petición
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TCON". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	0	0	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.

BUSY	DONE	ERROR	Descripción
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	La conexión se ha establecido correctamente.
0	7000	No se está procesando ninguna petición
0	7001	Iniciar el procesamiento de la petición, establecer la conexión
0	7002	Estableciendo la conexión (REQ es irrelevante).
1	8085	La ID de conexión (parámetro ID) ya está siendo utilizada por una conexión configurada.
1	8086	El parámetro ID está fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	809A	El parámetro CONNECT apunta a un campo que no concuerda con la longitud de la descripción de la conexión.
1	809B	El elemento InterfaceId de la estructura TCON_xxx no hace referencia a ninguna ID de hardware de una interfaz de CPU o de CM/CP, o bien tiene el valor "0".
1	80A0	Error de grupo para los códigos de error W#16#80A1 y W#16#80A2
1	80A2	El sistema está utilizando el puerto local o remoto.
1	80A3	Se está intentando volver a establecer una conexión existente.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida, es decir, concuerda con la dirección IP del interlocutor local.
1	80A5	Ya se está utilizando la ID de conexión.
1	80A7	Error de comunicación: "TDISCON (Página 5305)" se ha ejecutado antes de finalizar "TCON".
1	80B2	El parámetro CONNECT apunta a un bloque de datos que ha sido generado con el atributo "Guardar sólo en la memoria de carga".
1	80B4	Al establecer una conexión pasiva vía el protocolo ISO on TCP (connection_type = B#16#12) se han infringido una o varias de las condiciones siguientes: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0 • Con local_tsap_id_len >= B#16#03, local_tsap_id[1] es un carácter ASCII. • local_tsap_id[1] es un carácter ASCII y local_tsap_id_len >= B#16#03.
1	80B5	Con el tipo de conexión 13 = UDP sólo se permite un establecimiento pasivo de la conexión.
1	80B6	Error de parametrización en el parámetro connection_type del TCON_Param de SDT.
1	80B7	Error en uno de los parámetros siguientes del bloque de datos para la descripción de la conexión: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len.
1	80B8	La descripción de la conexión de la ID del elemento estructural y la ID del parámetro del bloque son distintas.
1	80C3	Todos los recursos de conexión están ocupados.

ERROR	STATUS* (W#16#...)	Explicación
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos parámetros. • Una instrucción "TDISCON (Página 5305)" está deshaciendo la conexión configurada.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TCON: Establecer una conexión

Descripción

La instrucción "TCON" permite configurar y establecer una conexión. Una vez configurada y establecida la conexión, la CPU la mantiene y la vigila automáticamente. "TCON" se ejecuta de forma asíncrona.

Para configurar la conexión se utilizan los datos de conexión especificados en los parámetros CONNECT e ID.

- En lo posible, en el parámetro CONNECT utilice solo estructuras predefinidas tal y como se crearon en la parametrización de la conexión, en la ventana de inspección del editor de programas.
- Si el parámetro CONNECT no está interconectado con una de las estructuras listadas en la tabla de parámetros o la estructura contiene un error, el parámetro STATUS devuelve el código de error 8089.

Para poder establecer la conexión debe detectarse un flanco ascendente en el parámetro REQ. Si la conexión se establece correctamente, el parámetro DONE se pone a "1".

La instrucción ""TCON" V3.0 también puede utilizarse con la CPU S7-1200 a partir de la versión 4.0.

Número de conexiones posibles

El número de conexiones posibles se indica en los datos técnicos de la CPU.

Conexión vía TCP e ISO on TCP

Ambos interlocutores llaman la instrucción "TCON" para crear y establecer la conexión. En la parametrización se especifica cuáles son el punto final activo y el punto final pasivo de la comunicación.

Si se interrumpe la conexión, p. ej. debido a una interrupción de la línea o por el interlocutor remoto, el interlocutor activo intenta restablecer la conexión. No es preciso volver a llamar "TCON". No obstante, esto solo es aplicable si "TCON" se ha ejecutado una vez correctamente (DONE = 1).

Con la ejecución de la instrucción "TDISCON (Página 5305)" o en el estado operativo STOP de la CPU se deshace una conexión existente y se elimina la conexión configurada. Para volver a configurar y establecer la conexión es preciso ejecutar "TCON" de nuevo.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "TCON":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Inicia la petición para establecer la conexión con un flanco ascendente.
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión asignada. Rango de valores: W#16#0001 a W#16#0FFF
CONNECT	InOut	VARIANT	D	Puntero hacia la descripción de la conexión • Con TCP o UDP, utilice la estructura TCON_IP_v4 Consulte la descripción en: Auto-Hotspot • Con ISO-on-TCP, utilice la estructura TCON_IP_RFC Consulte la descripción en: Auto-Hotspot • Con ISO, utilice la estructura TCON_ISOnative (solo con CP1543-1) Consulte la descripción en: "Estructura de la descripción de la conexión conforme a TCON_ISOnative"
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No se puede iniciar una nueva petición
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TCON". La información de error se indica en el parámetro STATUS.

La instrucción "TCON" genera un mensaje de error en la versión 3.0 cuando fracasa un establecimiento activo de la conexión con un interlocutor remoto. Para realizar un nuevo establecimiento de conexión, genere un flanco ascendente en el parámetro REQ.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	0	0	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Estructura de la descripción de la conexión conforme a TCON_ISOnative

Para parametrizar las conexiones en ISO, se utiliza un DB de descripción de la conexión con una estructura conforme a TCON_ISOnative. La estructura de datos fija de TCON_ISOnative contiene los parámetros necesarios para establecer la conexión.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceId	HW_ANY	64	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	1	Referencia a esta conexión (ID unívoca en el rango de valores: de 1 a 4095).
4	ConnectionType	BYTE	16#16	Tipo de conexión: ISO
5	ActiveEstablished	BOOL	TRUE	Identificador del tipo de establecimiento de conexión: • FALSE: establecimiento pasivo de la conexión • TRUE: establecimiento activo de la conexión
8 … 13	RemoteMacAddress	ARRAY [1..6] of BYTE	-	Dirección MAC del punto final del interlocutor, p. ej. para 00-74-41-FD-AE-84: • MacAddr[1] = 00 • MacAddr[2] = 74 • MacAddr[3] = 41 • MacAddr[4] = FD • MacAddr[5] = AE • MacAddr[6] = 84
14 … 19	LocalMacAddress	ARRAY [1..6] of BYTE	-	Dirección MAC del punto final local, p. ej. para 00-74-41-FD-AE-84: • MacAddr[1] = 00 • MacAddr[2] = 74 • MacAddr[3] = 41 • MacAddr[4] = FD • MacAddr[5] = AE • MacAddr[6] = 84
20 … 53	RemoteTSelector	Struct	-	TSelector del interlocutor remoto: • Bytes 20 a 21 = TSelLength • Bytes 22 a 53: = TSel[1-32]
	TSelLength	UINT	-	Rango de valores de 0 a 32

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
	TSel	ARRAY [1..32] of BYTE	-	Rango de valores de 0 a 255 en bytes
54 .. . 87	LocalTSelector	Struct	-	TSelector del interlocutor remoto: • Bytes 20 a 21 = TSelLength • Bytes 22 a 53: = TSel[1-32]
	TSelLength	UINT		Rango de valores de 0 a 32
	TSel	ARRAY [1..32] of BYTE	-	Rango de valores de 0 a 255 en bytes
88 … 89	CrRetransmitionTime	UINT	-	Tiempo hasta que se repite el establecimiento de la conexión en segundos.
90 .. . 91	DataRetransmitionTime	UINT	100 ms	Tiempo hasta que se repite la transferencia de datos en milisegundos.
92 .. . 93	MaxRetransmitionCount	UINT	-	Número máximo de repeticiones.
94 .. . 95	InactivityTime	UINT	-	En segundos
96 .. . 97	WindowTime	UINT		En segundos

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	La conexión se ha establecido correctamente.
0	7000	No se está procesando ninguna petición
0	7001	Iniciar el procesamiento de la petición, establecer la conexión
0	7002	Estableciendo la conexión (REQ es irrelevante).
1	8085	La ID de conexión (parámetro ID) ya está siendo utilizada por una conexión configurada.
1	8086	El parámetro ID está fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones
1	8089	El parámetro CONNECT no apunta a una descripción de la conexión o bien la descripción de la conexión se ha creado manualmente.
1	809A	La estructura del parámetro CONNECT no es compatible o bien la longitud no es válida.
1	809B	El elemento InterfaceId de la estructura TCON_xxx no hace referencia a ninguna ID de hardware de una interfaz de CPU o de CM/CP, o bien tiene el valor "0".
1	80A2	El sistema está utilizando el puerto local o remoto. Localmente se reservan los siguientes puertos: 20, 21, 80, 102, 135, 161, 162, 443, 34962, 34963, 34964 y el rango de 49152 a 65535.
1	80A3	La ID está siendo utilizada por una conexión creada por el programa de usuario y que también utiliza la misma descripción de conexión en el parámetro CONNECT.
1	80A4	La dirección IP del punto final remoto de la conexión no es válida o bien concuerda con la dirección IP del interlocutor local.
1	80A7	Error de comunicación: "TDISCON (Página 5305)" se ha ejecutado antes de finalizar "TCON".

ERROR	STATUS* (W#16#...)	Explicación
1	80B4	Solo en TCON_IP_RFC: No se ha indicado el T-Selector local, el primer byte no contiene el valor 0x0E (solo para una longitud de T-Selector = 2), o bien el T-Selector local empieza por "SIMATIC-".
1	80B5	En el tipo de conexión 13 = UDP solo se permite establecer una conexión pasiva (el parámetro ActiveEstablished de la estructura TCON_IP_v4 / TCON_PARAM tiene el valor TRUE).
1	80B6	Error de parametrización en el parámetro ConnectionType del bloque de datos para la descripción de la conexión. • Solo válido para TCON_IP_v4: 0x11, 0x0B y 0x13. • Solo válido para TCON_IP_RFC: 0x0C y 0x12
1	80B7	Con TCON_IP_v4: • TCP (establecimiento activo de la conexión): el puerto remoto es "0". • TCP (establecimiento pasivo de la conexión): El puerto local es "0". • UDP: El puerto local es "0". Con TCON_IP_RFC: • Se ha indicado una longitud superior a 32 bytes para el T-Selector local (LocalTSelector) o remoto (RemoteTSelector). • Se ha introducido una longitud superior a 32 bytes para el T-Selector (local o remoto) en TSelLength. • Error de longitud de la dirección IP del interlocutor correspondiente.
1	80B8	La ID del parámetro de la descripción de la conexión local (estructura del parámetro CONNECT) y la ID del parámetro de la instrucción son distintas.
1	80C3	Todos los recursos de conexión están ocupados.
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos parámetros. • Una instrucción "TDISCON (Página 5305)" está deshaciendo la conexión configurada.
1	80C5	El interlocutor rechaza la conexión, la ha deshecho o la ha finalizado de manera activa.
1	80C6	No es posible acceder al interlocutor (error de red).
1	80C7	Rebase de tiempo de la ejecución.
1	80C8	El valor del parámetro ID ya está siendo utilizado por una conexión creada mediante el programa de usuario. La conexión utiliza la misma ID pero distintos ajustes de conexión en el parámetro CONNECT.
1	80C9	Error en la validación del interlocutor. El interlocutor que pretende establecer una conexión no se corresponde con el interlocutor definido en la estructura del parámetro CONNECT.
1	80CE	La dirección IP de la interfaz local es 0.0.0.0.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TDISCON: Desconectar comunicación

Descripción

La instrucción "TDISCON" deshace la conexión entre la CPU y un interlocutor.

Funcionamiento

"TDISCON" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. La petición para el establecimiento de la conexión se inicia llamando "TDISCON" con REQ = 1.

Tras la ejecución correcta de "TDISCON", la ID indicada en "TCON" ya no es válida, por lo que no puede utilizarse para transmitir ni para recibir.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas (véase también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277)).

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra "TDISCON" actualmente o cuándo ha finalizado el establecimiento de la conexión.

BUSY	DONE	ERROR	Descripción
1	0	0	La petición está en proceso.
0	1	0	La petición ha finalizado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
0	0	0	No se ha asignado ninguna (nueva) petición a la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TDISCON":

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
REQ	Input	BOOL	I, Q, M, D, L o constante	I, Q, M, D, L o constante	Parámetro de control REQUEST, inicia la petición para deshacer la conexión dada por ID. La petición se inicia con un flanco ascendente.
ID	Input	CONN_OUC (WORD)	D, L o constante	I, Q, M, D, L o constante	Referencia a la conexión que se va a deshacer (ID de conexión) Rango de valores: W#16#0001 a W#16#0FFF
DONE	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Parámetros de estado: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Parámetros de estado: • BUSY = 1: la petición todavía no ha finalizado. • BUSY = 0: la petición ha finalizado o aún no se ha iniciado.

Parámetro	Declaración	Tipo de datos	Área de memoria		Descripción
			S7-1200	S7-1500	
ERROR	Output	BOOL	I, Q, M, D, L	I, Q, M, D, L	Parámetros de estado: • ERROR= 0: ningún error. • ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error
STATUS	Output	WORD	I, Q, M, D, L	I, Q, M, D, L	Parámetros de estado: Información de error (véase "Parámetro ERROR y STATUS")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	Se ha podido deshacer la conexión
0	7000	No se está procesando ninguna petición.
0	7001	Inicio del procesamiento de la petición, deshaciendo conexión.
0	7002	Llamada intermedia (REQ irrelevante), deshaciendo conexión.
1	8086	El parámetro ID se encuentra en un rango de valores no válido.
1	80A3	Se está intentando deshacer una conexión inexistente o la conexión ya se ha deshecho.
1	80C4	Error de comunicación temporal: La interfaz se está reparametrizando o la conexión se está configurando.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TSEND: Enviar datos a través de la conexión

TSEND: Enviar datos a través de la conexión

Descripción

La siguiente descripción de la instrucción "TSEND" es válida para la CPU S7-1200 hasta la versión 3.0.

La instrucción "TSEND" sirve para enviar datos a través de una conexión existente. "TSEND" se ejecuta de forma asíncrona.

El área de transmisión se especifica en el parámetro DATA. Éste contiene la dirección y la longitud de los datos que deben enviarse. Para los datos que deben enviarse se pueden utilizar todos los tipos de datos exceptuando BOOL y Array of BOOL.

La petición de transmisión se ejecuta cuando se detecta un flanco ascendente en el parámetro REQ.

En el parámetro LEN se especifica el número máximo de bytes que deben enviarse con una petición de transmisión.

- En la transmisión de datos a través de TCP (protocolo de streaming), la instrucción "TSEND" no transfiere información sobre la longitud de los datos enviados a "TRCV (Página 5314)".
- En la transmisión de datos a través de ISO-on-TCP (protocolo orientado a mensajes) se transfiere la longitud de los datos enviados a "TRCV (Página 5314)". La cantidad de datos enviados a través de "TSEND" en forma de paquete debe volver a recibirse también en el lado del receptor ("TRCV (Página 5314)"):
  - si el búfer de recepción es demasiado pequeño para los datos enviados, se produce un error en el lado de receptor.
  - Si el búfer de recepción es suficientemente grande, "TRCV" devuelve DONE=1 en cuanto se ha recibido el paquete de datos.

Los datos por enviar no se pueden editar hasta que no se haya ejecutado por completo la petición de transmisión. Si la petición de transmisión se ejecuta correctamente, el parámetro DONE se pone a "1". No obstante, el estado lógico "1" en el parámetro DONE no confirma que el interlocutor haya leído ya los datos enviados.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Inicia la petición de transmisión cuando se detecta un flanco ascendente.
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión establecida con "TCON". Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	UINT	I, Q, M, D, L o constante	Número máximo de bytes que se envían con la petición.
DATA	InOut	VARIANT	I, Q, M, D	Puntero hacia el área de transmisión que contiene la dirección y la longitud de los datos por enviar. La dirección hace referencia a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No es posible iniciar una nueva petición.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros LEN y DATA

- Con LEN = 0 se envían todos los datos especificados con el parámetro DATA.
- Si el número de bytes en el parámetro LEN excede la longitud de los datos que deben enviarse y que se han definido con el parámetro DATA, se visualiza el código de error 8088 en el parámetro STATUS (véase a continuación la descripción del parámetro STATUS).
- Si a través del parámetro DATA se hace referencia a una estructura (Struct), LEN puede ser más corto que la estructura. En este caso, solo se transfieren los datos hasta la longitud del parámetro LEN.
- En los tipos de datos STRING y WSTRING se transfieren todos los datos cuando el parámetro LEN = 0. Si LEN > 0 la longitud debe abarcar como mínimo el número máximo de bytes, además de dos bytes que contienen la información de la longitud. Encontrará más información sobre la estructura de los tipos de datos en: "Vista general de los tipos de datos válidos (Página 2321)".
- El número máximo de bytes que se pueden transferir es de 65534.
- Si se utilizan variables estructuradas de DB optimizados, la dirección de la variable estructurada del parámetro DATA se deberá interconectar y el parámetro LEN se deberá poner a 0. De este modo se puede garantizar una transferencia de toda la estructura con seguridad de tipos siempre que en el lado del receptor se utilice la misma estructura.

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TSEND". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	0	0	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Nota

Debido al procesamiento asíncrono de "TSEND", los datos del área de transmisión deben mantenerse coherentes hasta que el parámetro DONE o el parámetro ERROR se ponga a "1".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Descripción
0	0000	Petición de transmisión finalizada sin errores.
0	7000	No se está procesando ninguna petición.
0	7001	Inicio del procesamiento de la petición, enviando los datos. Durante el procesamiento, el sistema operativo accede a los datos del área de transmisión DATA.
0	7002	Procesando la petición (REQ es irrelevante). Durante el procesamiento, el sistema operativo accede a los datos del área de transmisión DATA.
1	8085	• El parámetro LEN excede el valor máximo admisible (65536). • Los parámetros DATA y LEN tienen ambos el valor "0".
1	8086	El parámetro ID se encuentra fuera del área de direcciones admisible (1..0xFFF).
1	8088	El parámetro LEN es mayor que el rango indicado en DATA.
1	80A1	Error de comunicación: • La conexión indicada no se ha establecido todavía. • La conexión indicada se está finalizando. No es posible transferir a través de esta conexión. • La interfaz se está reinicializando.
1	80B3	La variante de protocolo parametrizada (parámetro ConnectionType en la descripción de la conexión) es UDP. En el caso de una conexión UDP, utilice la instrucción "TUSEND".
1	80C3	• Ya se está procesando un bloque con esta ID en otro grupo de prioridad. • Escasez interna de recursos.
1	80C4	Error de comunicación temporal: • La conexión con el interlocutor no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos ajustes de parámetros o la conexión se está estableciendo.
1	80C5	El interlocutor ha finalizado la conexión.
1	80C6	Error de red. No es posible acceder al interlocutor.
1	80C7	Rebase de tiempo de la ejecución.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TSEND: Enviar datos a través de la conexión

Descripción

La siguiente descripción de la instrucción "TSEND" es válida para la CPU S7-1500 y S7-1200 V4.0.

La instrucción "TSEND" sirve para enviar datos a través de una conexión existente. "TSEND" se ejecuta de forma asíncrona.

El área de transmisión se especifica en el parámetro DATA. Éste contiene la dirección y la longitud de los datos que deben enviarse. Para los datos que deben enviarse se pueden utilizar todos los tipos de datos exceptuando BOOL y Array of BOOL.

La petición de transmisión se ejecuta cuando se detecta un flanco ascendente en el parámetro REQ.

En el parámetro LEN se especifica el número máximo de bytes que deben enviarse con una petición de transmisión.

- En la transmisión de datos a través de TCP (protocolo de streaming), la instrucción "TSEND" no transfiere información sobre la longitud de los datos enviados a "TRCV (Página 5314)".
- En la transmisión de datos a través de ISO-on-TCP (protocolo orientado a mensajes) se transfiere la longitud de los datos enviados a "TRCV (Página 5314)". La cantidad de datos enviados a través de "TSEND" en forma de paquete debe volver a recibirse también en el lado del receptor ("TRCV (Página 5314)"):
  - si el búfer de recepción es demasiado pequeño para los datos enviados, se produce un error en el lado de receptor.
  - Si el búfer de recepción es suficientemente grande, "TRCV" devuelve DONE=1 en cuanto se ha recibido el paquete de datos.

Los datos por enviar no se pueden editar hasta que no se haya ejecutado por completo la petición de transmisión. Si la petición de transmisión se ejecuta correctamente, el parámetro DONE se pone a "1". No obstante, el estado lógico "1" en el parámetro DONE no confirma que el interlocutor haya leído ya los datos enviados.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Inicia la petición de transmisión cuando se detecta un flanco ascendente.
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión establecida con "TCON". Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	UDINT	I, Q, M, D, L o constante	Número máximo de bytes que se envían con la petición.
DATA	InOut	VARIANT	I, Q, M, D	Puntero hacia el área de transmisión que contiene la dirección y la longitud de los datos por enviar. La dirección hace referencia a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No es posible iniciar una nueva petición.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros LEN y DATA

- Con LEN = 0 se envían todos los datos especificados con el parámetro DATA.
- Si el número de bytes en el parámetro LEN excede la longitud de los datos que deben enviarse y que se han definido con el parámetro DATA, se visualiza el código de error 8088 en el parámetro STATUS (véase a continuación la descripción del parámetro STATUS).
- Si a través del parámetro DATA se hace referencia a una estructura (Struct), LEN puede ser más corto que la estructura. En este caso, solo se transfieren los datos hasta la longitud del parámetro LEN.
- En los tipos de datos STRING y WSTRING se transfieren todos los datos cuando el parámetro LEN = 0. Si LEN > 0 la longitud debe abarcar como mínimo el número máximo de bytes, además de dos bytes que contienen la información de la longitud. Encontrará más información sobre la estructura de los tipos de datos en: "Vista general de los tipos de datos válidos (Página 2321)".
- El número máximo de bytes que se pueden transferir depende del dispositivo.
- Si se utilizan variables estructuradas de DB optimizados, la dirección de la variable estructurada del parámetro DATA se deberá interconectar y el parámetro LEN se deberá poner a 0. De este modo se puede garantizar una transferencia de toda la estructura con seguridad de tipos siempre que en el lado del receptor se utilice la misma estructura.

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "TSEND". La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	0	0	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Nota

Debido al procesamiento asíncrono de "TSEND", los datos del área de transmisión deben mantenerse coherentes hasta que el parámetro DONE o el parámetro ERROR se ponga a "1".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Descripción
0	0000	Petición de transmisión finalizada sin errores.
0	7000	No se está procesando ninguna petición.
0	7001	Inicio del procesamiento de la petición, enviando los datos. Durante el procesamiento, el sistema operativo accede a los datos del área de transmisión DATA.
0	7002	Procesando la petición (REQ es irrelevante). Durante el procesamiento, el sistema operativo accede a los datos del área de transmisión DATA.
1	8085	• El parámetro LEN excede el valor máximo admisible (65536). • Los parámetros DATA y LEN tienen ambos el valor "0".
1	8086	El parámetro ID se encuentra fuera del área de direcciones admisible (1..0xFFF).
1	8088	El parámetro LEN es mayor que el rango indicado en DATA.
1	80A1	Error de comunicación: • La conexión indicada no se ha establecido todavía. • La conexión indicada se está finalizando. No es posible transferir a través de esta conexión. • La interfaz se está reinicializando.
1	80B3	La variante de protocolo parametrizada (parámetro ConnectionType en la descripción de la conexión) es UDP. En el caso de una conexión UDP, utilice la instrucción "TUSEND".
1	80C3	• Ya se está procesando un bloque con esta ID en otro grupo de prioridad. • Escasez interna de recursos.
1	80C4	Error de comunicación temporal: • La conexión con el interlocutor no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos ajustes de parámetros o la conexión se está estableciendo.
1	80C5	El interlocutor ha finalizado la conexión.
1	80C6	Error de red. No es posible acceder al interlocutor.
1	80C7	Rebase de tiempo de la ejecución.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TRCV: Recibir datos a través de la conexión

TRCV: Recibir datos a través de la conexión

Descripción

La siguiente descripción de la instrucción "TRCV" es válida para la CPU S7-1200 hasta la versión 3.0.

La instrucción "TRCV" sirve para recibir datos a través de una conexión existente. "TRCV" se ejecuta de forma asíncrona.

Cuando el parámetro EN_R se pone a "1", se activa la recepción de datos. Los datos recibidos se escriben en un área de recepción. La longitud del área de recepción se especifica, en función de la variante de protocolo utilizada, bien sea con el parámetro LEN (si LEN <> 0), o bien mediante la indicación de longitud del parámetro DATA (si LEN = 0).

Para garantizar la coherencia de los datos recibidos, no se pueden realizar modificaciones en el parámetro DATA y en el área de recepción definida mientras se estén recibiendo datos.

Tras haberse recibido los datos correctamente, el parámetro NDR se pone a "1". La cantidad de datos recibida realmente se puede consultar en el parámetro RCVD_LEN.

Modos de recepción de "TRCV"

La tabla siguiente muestra cómo se escriben los datos recibidos en el área de recepción.

Variante de protocolo	Disponibilidad de los datos en el área de recepción	Parámetro connection_type* de la descripción de la conexión	Parámetro LEN
TCP (Modo Ad hoc)	Los datos están disponibles inmediatamente.	Valor hexadecimal: B#16#11 Valor entero: 17	0
TCP (Recepción de datos de longitud especificada)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#11 Valor entero: 17	de 1 a 8192
ISO on TCP (Transmisión de datos orientada a mensajes)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#12 Valor entero: 18	• de 1 a 1452 si se utiliza un CP. • de 1 a 8192 si no se utiliza ningún CP.
* Ver "Auto-Hotspot".			

TCP (Modo Ad hoc)

El modo Ad hoc sólo está disponible en la variante de protocolo TCP. Utilice el modo Ad hoc para recibir datos de longitudes dinámicas con la instrucción "TRCV".

El modo Ad hoc se establece asignando el valor "0" al parámetro LEN. Con el modo Ad hoc se pueden utilizar todos los tipos de datos para bloques de datos de acceso estándar. Para bloques de datos de acceso optimizado solo es posible utilizar el tipo de datos ARRAY of BYTE o tipos de datos de una longitud de 8 bits (p. ej., CHAR, USINT, SINT, etc.). Si está

activado el modo Ad hoc, la recepción de datos se indica en el parámetro NDR tras recibir un byte.

TCP (recepción de datos de longitud especificada)

Para una recepción de datos de la longitud indicada, introduzca la longitud de los datos en el parámetro LEN. La recepción de datos finalizará cuando se haya recibido la longitud de los datos indicados en el parámetro LEN en su totalidad. Solo entonces estarán disponibles los datos en el área de recepción (parámetro DATA). La recepción de los datos se notifica a través del parámetro de salida NDR. Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

ISO on TCP (Transmisión de datos orientada a mensajes)

En la variante de protocolo ISO on TCP se envían a través de una conexión bloques cerrados de mensajes que el receptor también reconoce como tales. Si se utiliza ISO on TCP, "TRCV" notifica la recepción de los datos en cuanto se recibe el bloque de mensajes en su totalidad. Los parámetros LEN y DATA definen el área de recepción. Si el búfer de recepción (parámetro DATA) es demasiado pequeño para los datos enviados, "TRCV" notifica un error. Si la recepción de datos es correcta, esto se notifica a través del parámetro de salida NDR. Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "TRCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L o constante	Habilitación de recepción
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión establecida con "TCON (Página 5298)". Rango de valores: W#16#0001 (1) a W#16#0FFF (4095)
LEN	Input	UDINT	I, Q, M, D, L o constante	Longitud del área de recepción en bytes (oculta). Si utiliza únicamente un área de memoria de acceso optimizado en el parámetro DATA, el parámetro LEN debe tener el valor "0".
DATA	InOut	VARIANT	I, Q, M, D	Puntero hacia el área de recepción
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado (New Data Received): • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No se puede iniciar una nueva petición

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: Ningún error. • 1: Se ha producido un error al ejecutar la instrucción. El parámetro STATUS contiene información más detallada.
STATUS	Output	WORD	I, Q, M, D, L	Parámetros de estado: Salida de la información de estado y error.
RCVD_LEN	Output	UDINT	I, Q, M, D, L	Cantidad de datos (en bytes) recibida realmente

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros LEN, DATA y RCVD_LEN

- Si LEN = 0, los datos recibidos se guardan en el área de recepción indicada en el parámetro DATA. El número de bytes recibidos se visualiza en el parámetro RCVD_LEN.
- Si la longitud indicada en el parámetro LEN excede la longitud de los datos recibidos en el parámetro DATA, se visualiza el código de error 8088 en el parámetro STATUS (véase a continuación la descripción del parámetro STATUS).
- Si a través del parámetro DATA se hace referencia a una estructura (Struct), LEN puede ser más corto que la estructura. En este caso, solo se transfieren los datos hasta la longitud del parámetro LEN.
- Si el parámetro DATA hace referencia a un bloque de datos de acceso optimizado, se debe ajustar el parámetro LEN a "0".
- Si a través del parámetro DATA se hace referencia a un tipo de datos STRING, la longitud indicada en el parámetro LEN debe ser 0 o >=2 (LEN = no puede ser =1).
- Si a través del parámetro DATA se hace referencia a un tipo de datos WSTRING, la longitud indicada en el parámetro LEN debe ser 0 o >=5.

Parámetros BUSY, NDR y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, NDR, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro NDR permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de TRCV. La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, NDR y ERROR:

BUSY	NDR	ERROR	Descripción
1	-	-	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Nota

Debido al procesamiento asíncrono de "TRCV", los datos del área de recepción solo son coherentes si el parámetro NDR se pone a "1".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	Petición finalizada correctamente. La longitud actual de los datos recibidos se indica en el parámetro RCVD_LEN.
0	7000	El bloque no está listo para la recepción.
0	7001	El bloque está listo para la recepción, se ha activado la petición de recepción.
0	7002	Llamada intermedia, se está procesando la petición de recepción. Nota: Durante este proceso se escriben datos en el área de recepción. Entretanto el acceso al área de recepción puede devolver datos incoherentes.
1	8085	• El parámetro LEN excede el valor máximo admisible. • El valor del parámetro LEN o DATA ha sido modificado después de la primera llamada. • Los dos parámetros LEN y el parámetro DATA tienen el valor "0" o LEN tiene una longitud mayor que el máximo valor admisible (65536).
1	8086	El parámetro ID se encuentra fuera del área de direcciones admisible (1 .. 0x0FFF).
1	8088	• El área de recepción es demasiado pequeña. • El valor del parámetro LEN es mayor que el área de recepción indicada en el parámetro DATA.
1	80A1	Error de comunicación: • La conexión indicada no se ha establecido todavía. • La conexión indicada se está finalizando. La petición de recepción no es posible a través de esta conexión. • La conexión se está reinicializando.
1	80B3	La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) es UDP. En el caso de una conexión UDP, utilice la instrucción "TURCV".
1	80C3	• Ya se está procesando un bloque con esta ID en otro grupo de prioridad. • Escasez interna de recursos.
1	80C4	Error de comunicación temporal: • La conexión con el interlocutor no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos ajustes de parámetros o la conexión se está estableciendo.
1	80C5	El interlocutor remoto ha finalizado la conexión.
1	80C6	No es posible acceder al interlocutor remoto (error de red).
1	80C7	Rebase de tiempo de la ejecución.
1	80C9	La longitud del área de recepción es menor que la longitud de los datos enviados.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TRCV: Recibir datos a través de la conexión

Descripción

La siguiente descripción de la instrucción "TRCV" es válida para la CPU S7-1500 y S7-1200 V4.0.

La instrucción "TRCV" sirve para recibir datos a través de una conexión existente. "TRCV" se ejecuta de forma asíncrona.

Cuando el parámetro EN_R se pone a "1", se activa la recepción de datos. Los datos recibidos se escriben en un área de recepción. La longitud del área de recepción se especifica, en función de la variante de protocolo utilizada, bien sea con el parámetro LEN (si LEN <> 0), o bien mediante la indicación de longitud del parámetro DATA (si LEN = 0).

Para garantizar la coherencia de los datos recibidos, no se pueden realizar modificaciones en el parámetro DATA y en el área de recepción definida mientras se estén recibiendo datos.

Tras haberse recibido los datos correctamente, el parámetro NDR se pone a "1". La cantidad de datos recibida realmente se puede consultar en el parámetro RCVD_LEN.

Modos de recepción de "TRCV"

La tabla siguiente muestra cómo se escriben los datos recibidos en el área de recepción.

Variante de protocolo	Parámetro ADHOC	Disponibilidad de los datos en el área de recepción	Parámetro connection_type de la descripción de la conexión	Parámetro LEN
TCP (Modo Ad hoc)	1 (Ad hoc activado)	Los datos están disponibles inmediatamente.	Valor hexadecimal: B#16#11 Valor entero: 17	1 hasta longitud máxima (depende de CPU)
TCP (Recepción de datos de longitud especificada)	0 (Ad hoc desactivado)	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#11 Valor entero: 17	de 1 a 8192
ISO on TCP (Transmisión de datos orientada a mensajes)	-	Los datos están disponibles en cuanto se ha recibido toda la longitud de datos especificada en el parámetro LEN.	Valor hexadecimal: B#16#12 Valor entero: 18	• de 1 a 1452 si se utiliza un CP. • de 1 a 8192 si no se utiliza ningún CP.

TCP (Modo Ad hoc)

El modo Ad hoc sólo está disponible en la variante de protocolo TCP. Utilice el modo Ad hoc para recibir datos de longitudes dinámicas con la instrucción "TRCV".

El modo Ad hoc se establece asignando el valor "1" al parámetro ADHOC. Con el modo Ad hoc se pueden utilizar todos los tipos de datos para bloques de datos de acceso estándar. Para bloques de datos de acceso optimizado solo es posible utilizar el tipo de datos ARRAY of BYTE o tipos de datos de una longitud de 8 bits (p. ej., CHAR, USINT, SINT, etc.). Si está activado el modo Ad hoc, la recepción de datos se indica en el parámetro NDR tras recibir un byte.

TCP (Recepción de datos de longitud especificada)

Para una recepción de datos con longitud indicada, asigne el valor "0" al parámetro ADHOC. Si el modo Ad hoc está desactivado, la recepción de datos finalizará solo cuando se haya recibido la longitud de los datos indicada en el parámetro LEN en su totalidad. Solo entonces estarán disponibles los datos en el área de recepción (parámetro DATA). Si la recepción de datos es correcta, esto se notifica a través del parámetro de salida NDR. Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

ISO on TCP (Transmisión de datos orientada a mensajes)

En la variante de protocolo ISO on TCP se envían a través de una conexión bloques cerrados de mensajes que el receptor también reconoce como tales. Si se utiliza ISO on TCP, "TRCV" notifica la recepción de los datos en cuanto se recibe el bloque de mensajes en su totalidad. Los parámetros LEN y DATA definen el área de recepción. Si el búfer de recepción (parámetro DATA) es demasiado pequeño para los datos enviados, "TRCV" notifica un error. Si la recepción de datos es correcta, esto se notifica a través del parámetro de salida NDR. Tras la recepción, la longitud de datos realmente recibida en bytes en el parámetro RCVD_LEN coincide con la longitud de datos del parámetro LEN.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TRCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L o constante	Habilitación de recepción
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión establecida con "TCON". Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	UDINT	I, Q, M, D, L o constante	Longitud del área de recepción en bytes (oculta). Si utiliza únicamente un área de recepción de acceso optimizado en el parámetro DATA, el parámetro LEN debe tener el valor "0".
ADHOC	Input	BOOL	I, Q, M, D, L o constante	Utilizar el modo Ad hoc para la variante de protocolo TCP (oculto).
DATA	InOut	VARIANT	I, Q, M, D	Puntero hacia el área de recepción
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado (New Data Received): • 0: Petición no iniciada aún o en ejecución • 1: Recibir nuevos datos
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No se puede iniciar una nueva petición
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • 0: Ningún error • 1: Con errores

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción
RCVD_LEN	Output	UDINT	I, Q, M, D, L	Cantidad de datos (en bytes) recibida realmente

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros LEN, DATA y RCVD_LEN

- Si LEN = 0, los datos recibidos se guardan en el área de recepción indicada en el parámetro DATA. El número de bytes recibidos se visualiza en el parámetro RCVD_LEN.
- Si la longitud indicada en el parámetro LEN excede la longitud de los datos recibidos en el parámetro DATA, se visualiza el código de error 8088 en el parámetro STATUS (véase a continuación la descripción del parámetro STATUS).
- Si a través del parámetro DATA se hace referencia a una estructura (Struct), LEN puede ser más corto que la estructura. En este caso, solo se transfieren los datos hasta la longitud del parámetro LEN.
- Si el parámetro DATA hace referencia a un bloque de datos de acceso optimizado, se debe ajustar el parámetro LEN a "0". Si la longitud de los datos en los tipos de datos simples no coincide, los datos no se reciben y se visualiza el código de error 8088 en el parámetro STATUS.
- Si a través del parámetro DATA se hace referencia a un tipo de datos STRING, la longitud indicada en el parámetro LEN debe ser 0 o >=2 (LEN = no puede ser =1).
- Si a través del parámetro DATA se hace referencia a un tipo de datos WSTRING, la longitud indicada en el parámetro LEN debe ser 0 o >=5.

Parámetros BUSY, NDR y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, NDR, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro NDR permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de TRCV. La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, NDR y ERROR:

BUSY	NDR	ERROR	Descripción
1	-	-	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Nota

Debido al procesamiento asíncrono de "TRCV", los datos del área de recepción solo son coherentes si el parámetro NDR se pone a "1".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	Petición finalizada correctamente. La longitud actual de los datos recibidos se indica en el parámetro RCVD_LEN.
0	7000	El bloque no está listo para la recepción.
0	7001	El bloque está listo para la recepción, se ha activado la petición de recepción.
0	7002	Llamada intermedia, se está procesando la petición de recepción. Nota: Durante este proceso se escriben datos en el área de recepción. Entretanto el acceso al área de recepción puede devolver datos incoherentes.
1	8085	• El parámetro LEN excede el valor máximo admisible. • El valor del parámetro LEN o DATA ha sido modificado después de la primera llamada. • Los dos parámetros LEN y el parámetro DATA tienen el valor "0" o LEN tiene una longitud mayor que el máximo valor admisible (65536).
1	8086	El parámetro ID se encuentra fuera del área de valores admisible (1 .. 0x0FFF).
1	8088	• El área de recepción es demasiado pequeña. • El valor del parámetro LEN es mayor que el área de recepción indicada en el parámetro DATA.
1	80A1	Error de comunicación: • La conexión indicada no se ha establecido todavía. • La conexión indicada se está finalizando. La petición de recepción no es posible a través de esta conexión. • La conexión se está reinicializando.
1	80B3	La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) es UDP. En el caso de una conexión UDP, utilice la instrucción "TURCV".
1	80C3	• Ya se está procesando un bloque con esta ID en otro grupo de prioridad. • Escasez interna de recursos.
1	80C4	Error de comunicación temporal: • La conexión con el interlocutor no se puede establecer en estos momentos. • La interfaz está recibiendo nuevos ajustes de parámetros o la conexión se está estableciendo.
1	80C5	El interlocutor remoto ha finalizado la conexión.
1	80C6	No es posible acceder al interlocutor remoto (error de red).
1	80C7	Rebase de tiempo de la ejecución.
1	80C9	La longitud del área de recepción es menor que la longitud de los datos enviados.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

TRCV: Recibir datos a través de la conexión (Página 5314)

TCON: Establecer una conexión (Página 5301)

Estructura de la información de dirección del interlocutor remoto en UDP

Vista general

Si se utiliza una conexión UDP, la información de dirección del interlocutor remoto se guarda en el tipo de datos de sistema TADDR_Param:

- En el caso de la instrucción "TUSEND (Página 5323)", a través de TADDR_Param se transfiere la información de dirección del receptor en el parámetro ADDR.
  A partir del tipo de datos de sistema la instrucción lee la información de dirección guardada que está asociada al interlocutor remoto.
- En el caso de la instrucción "TURCV (Página 5326)", a través de TADDR_Param se recibe la dirección del remitente en el parámetro ADDR.
  La instrucción escribe la información de dirección en el tipo de datos de sistema.

Estructura de la información de dirección conforme a TADDR_Param

El tipo de datos de sistema TADDR_Param contiene la información de dirección del interlocutor remoto, la cual está compuesta por la dirección IP y el número de puerto.

El tipo de datos de sistema TADDR_Param tiene la estructura siguiente:

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
de 0 a 3	rem_ip_addr	ARRAY [1..4] of USINT	B#16#00 ...	Dirección IP del interlocutor remoto, p. ej. 192.168.002.003: • rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#02 (002) • rem_ip_addr[4] = B#16#03 (003) Consulte la dirección IP en las propiedades de la interfaz del interlocutor remoto en la vista Dispositivos y redes. Alternativamente, esta también se visualiza en las propiedades de la conexión UDP en Detalles de dirección.
de 4 a 5	rem_port_nr	UINT	B#16#00 ...	Número de puerto remoto (para valores posibles véase: Auto-Hotspot): • rem_port_nr[1] = byte alto del n.º de puerto en representación hexadecimal • rem_port_nr[2] = byte bajo del n.º de puerto en representación hexadecimal Consulte el número de puerto en las propiedades de la conexión UDP en la vista Dispositivos y redes. En Detalles de dirección se visualiza el número de puerto como valor decimal. Ejemplo: número de puerto = 2000 (decimal) / W#16#07D0 (hexadecimal) • rem_port_nr[1] = 07 (byte más significativo) • rem_port_nr[2] = D0 (byte menos significativo)
de 6 a 7	reserved	WORD	B#16#00 ...	No utilizado. Conserve el valor "0" en este parámetro.

Crear TADDR_Param en un bloque de datos

Para crear TADDR_Param, se dispone de las siguientes posibilidades:

- Cree un nuevo bloque de datos y seleccione como tipo TADDR_Param en el cuadro de diálogo "Agregar nuevo bloque".
- Abra un bloque de datos existente, cree una nueva variable e introduzca TADDR_Param en la columna Tipo de datos.

Un bloque de datos puede contener varios tipos de datos de sistema TADDR_Param.

TUSEND: Enviar datos vía Ethernet (UDP)

Descripción

La instrucción "TUSEND" envía datos a través de UDP al interlocutor remoto direccionado con el parámetro ADDR.

⚠ ADVERTENCIA

Transmisión de datos a través de UDP

En la transmisión de datos a través de UDP según RFC 768, los datos se transmiten al interlocutor remoto sin acuse y, por lo tanto, sin seguridad. Esto significa que se pueden extraviar datos sin que su pérdida se indique en el bloque.

Nota

En procesos de transmisión consecutivos a diferentes interlocutores, en las llamadas de "TUSEND" solo es necesario adaptar el parámetro ADDR. Sin embargo, se prescinde de la nueva llamada de las instrucciones "TCON (Página 5298)" y "TDISCON (Página 5305)". Para que el interlocutor remoto en cuestión pueda recibir datos, deberá tener configurado el puerto UDP direccionado.

Funcionamiento

"TUSEND" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. Para iniciar el proceso de transmisión, genere un flanco ascendente en el parámetro REQ.

El estado de la petición se indica a través de los parámetros de salida BUSY, DONE, ERROR y STATUS .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra "TUSEND" actualmente o cuándo finalizó la transmisión.

BUSY	DONE	ERROR	Descripción
TRUE	FALSE	FALSE	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Nota

Debido al funcionamiento asíncrono de "TUSEND", los datos del área de transmisión deben mantenerse coherentes hasta que el parámetro DONE o el parámetro ERROR adopten el valor TRUE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TUSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	El parámetro de control REQUEST activa la petición de transmisión con un flanco ascendente. Los datos se transmiten desde el área indicada con DATA y LEN.
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la respectiva conexión entre el programa de usuario y la capa de comunicación del sistema operativo. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión en la instrucción "TCON". Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	UINT	I, Q, M, D, L o constante	Número de bytes que deben enviarse con la petición Rango de valores: de 1 a 1472
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores. El valor solo se muestra durante un ciclo.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. No es posible iniciar otra petición. • BUSY = 0: la petición ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: durante la ejecución se ha producido un error. STATUS devuelve información detallada sobre el tipo de error
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Información de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DATA	InOut	VARIANT	I, Q, M, D	Área de transmisión, contiene la dirección y la longitud La dirección remite a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos
ADDR	InOut	VARIANT	D	Puntero hacia el tipo de datos de sistema TADDR_Param. Guarde la información de dirección del interlocutor remoto (dirección IP y número de puerto) en un bloque de datos con el tipo de datos de sistema TADDR_Param. Consulte también: Estructura de la información de dirección del interlocutor remoto en UDP (Página 5322)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	La petición de transmisión ha finalizado sin errores
0	7000	No se está procesando ninguna petición
0	7001	Inicio del procesamiento de la petición, enviando los datos. Nota: En esta fase de la ejecución, el sistema operativo accede a los datos del área de transmisión DATA.
0	7002	Llamada intermedia (REQ irrelevante), la petición se está ejecutando Nota: En esta fase de la ejecución, el sistema operativo accede a los datos del área de transmisión DATA.
1	8085	El parámetro LEN tiene el valor "0" o excede el valor máximo admisible.
1	8086	El parámetro ID se encuentra en un rango de valores no válido.
0	8088	El parámetro LEN es mayor que el área de memoria indicada en DATA.
1	8089	El parámetro ADDR no apunta a un bloque de datos con la estructura TADDR_Param.
1	80A1	Error de comunicación: • Todavía no se ha establecido la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. • Se está estableciendo la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. No es posible transmitir a través de esta conexión. • La interfaz se está reinicializando.
1	80A4	La dirección IP (en el parámetro ADDR) del punto final de la conexión no es válida o coincide con la dirección IP propia.

ERROR	STATUS* (W#16#...)	Explicación
1	80B3	• La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) no es UDP. Utilice "TSEND (Página 5310)". • Parámetro ADDR: Indicación no válida para n.º de puerto.
1	80B7	La longitud de la estructura referenciada por el parámetro ADDR no es 8 bytes.
1	80C3	• Ya se está ejecutando un bloque con esta ID en otra clase de prioridad. • Falta de recursos internos
1	80C4	Error de comunicación temporal: • En este momento no es posible establecer la conexión entre el programa de usuario y la capa de comunicación del sistema operativo. • La interfaz se está reparametrizando.
1	80C5	El interlocutor remoto ha finalizado la conexión.
1	80C6	No es posible acceder al interlocutor remoto (error de red).
1	80C7	Rebase de tiempo de la ejecución.
-	Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

TURCV: Recibir datos vía Ethernet (UDP)

Descripción

La instrucción "TURCV" recibe datos a través de UDP. Tras la finalización correcta de "TURCV" en el parámetro ADDR se pone a disposición la dirección del interlocutor remoto, es decir, del transmisor.

> ⚠ **ADVERTENCIA**
>
> **Transmisión de datos sin seguridad**
>
> En la transmisión de datos a través de UDP según RFC 768, los datos se transmiten al interlocutor remoto sin acuse y, por lo tanto, sin seguridad. Esto significa que se pueden extraviar datos sin que su pérdida se indique en el bloque.

Funcionamiento

"TURCV" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. Para iniciar la petición de recepción, llame la instrucción "TURCV" con EN_R = 1.

El estado de la petición se indica a través de los parámetros de salida BUSY, DONE, ERROR y STATUS .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 3277).

En la tabla siguiente se muestra la correspondencia entre BUSY, NDR y ERROR. Permite determinar en qué estado se encuentra TURCV actualmente o cuándo finalizó la recepción.

BUSY	NDR	ERROR	Descripción
TRUE	FALSE	FALSE	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente. Nuevos datos recibidos.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Nota

Debido al funcionamiento asíncrono de "TURCV", los datos no son coherentes en el área de recepción hasta que el parámetro NDR no adopta el valor TRUE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TURCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L o constante	Parámetro de control enabled to receive Con EN_R = 1 "TURCV" pasa a disponibilidad de recepción. La petición de recepción se está ejecutando.
ID	Input	WORD	I, Q, M, D, L o constante	Referencia a la conexión correspondiente entre el programa de usuario y la capa de comunicación del sistema operativo. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	UINT	I, Q, M, D, L o constante	Longitud del área de recepción en bytes: 0 (recomendado) o bien 1 a 1472
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • NDR = 0: la petición todavía no se ha iniciado o está en curso • NDR = 1: la petición ha finalizado correctamente
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. No es posible iniciar otra petición. • BUSY = 0: la petición ha finalizado.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Información de error
RCVD_LEN	Output	UINT	I, Q, M, D, L	Número de datos recibidos realmente en bytes.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DATA	InOut	VARIANT	I, Q, M, D, L	Área de recepción La dirección hace referencia a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos
ADDR	InOut	VARIANT	D	Puntero hacia el tipo de datos de sistema TADDR_Param. La información de dirección del interlocutor remoto (dirección IP y número de puerto) se escribe en un bloque de datos con el tipo de datos de sistema TADDR_Param. Consulte también: Estructura de la información de dirección del interlocutor remoto en UDP (Página 5322)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS* (W#16#...)	Explicación
0	0000	Se han aplicado nuevos datos. La longitud actual de los datos recibidos se indica en RCVD_LEN.
0	7000	El bloque no está preparado para la recepción
0	7001	El bloque está preparado para la recepción, la petición de recepción se ha activado
0	7002	Llamada intermedia, la petición de recepción está en proceso Nota: En esta fase de ejecución, "TURCV" escribe datos en el área de recepción. Por ello, en caso de error es posible que haya datos incoherentes en el área de recepción.
1	8085	El parámetro LEN es mayor que el valor máximo permitido, o ha modificado uno de los parámetros LEN o DATA respecto a la primera llamada
1	8086	El parámetro ID se encuentra en un rango de valores no válido
1	8088	• El área de recepción es demasiado pequeña • El valor de LEN es mayor que el área de recepción especificada por DATA
1	8089	El parámetro ADDR no apunta a un bloque de datos con la estructura TADDR_Param.
1	80A1	Error de comunicación: • Todavía no se ha establecido la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. • Se está estableciendo la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. No es posible ejecutar una petición de recepción a través de esta conexión. • La interfaz se está reparametrizando.
1	80B3	La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) no es UDP. Utilice "TRCV (Página 5314)".
1	80B7	la longitud del parámetro ADDR no equivale a 8 bytes.
1	80C3	• Ya se está ejecutando un bloque con esta ID en otra clase de prioridad. • Falta de recursos internos
1	80C4	Error de comunicación temporal: La interfaz se está reparametrizando.

ERROR	STATUS* (W#16#...)	Explicación
1	80C5	El interlocutor remoto ha finalizado la conexión.
1	80C7	Rebase de tiempo de la ejecución.
1	80C9	Con RFC1006/UDP: los datos recibidos tienen una longitud mayor de lo esperado (superan el tamaño del búfer de recepción).
-	Información de error general	Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

TCON: Establecer una conexión (Página 5298)

TDISCON: Desconectar comunicación (Página 5305)

T_RESET: Reiniciar conexión

Descripción

La instrucción "T_RESET" deshace y vuelve a establecer una conexión existente.

Los puntos finales locales de la conexión se conservan. Se generan de forma automática:

- Si se ha configurado una conexión y se ha cargado en la CPU.
- Si se ha creado una conexión a través del programa de usuario, por ejemplo llamando la instrucción "TCON (Página 5298)".

La instrucción "T_RESET" se puede ejecutar con todos los tipos de conexión (TCP, UDP, ISO on TCP, etc.). A este respecto es indistinto si se ha utilizado la interfaz local de la CPU o la interfaz de un CM/CP para la conexión.

Tras llamar la instrucción "T_RESET" a través del parámetro REQ, se deshace la conexión especificada mediante el parámetro ID y, en caso necesario, se vacía el búfer para enviar y recibir datos. Al interrumpirse la conexión, se interrumpen también las transmisiones de datos que estén en curso. Con ello se corre el peligro de que se puedan perder datos de una transmisión en curso. Después, la CPU definida como interlocutor activo intenta restablecer automáticamente la conexión interrumpida. Por ello, no es necesario llamar la instrucción "TCON (Página 5298)" para restablecer la conexión.

El estado de la petición se indica a través de los parámetros de salida DONE, BUSY y STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_RESET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Parámetro de control REQUEST, inicia la petición para deshacer la conexión dada por ID. La petición se inicia con un flanco ascendente.
ID	Input	CONN_OUC	I, Q, M, D, L o constante	Referencia a la conexión con el interlocutor pasivo que se debe deshacer. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado BUSY • 0: La petición ha finalizado. • 1: La petición todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR • 0: No se ha producido ningún error. • 1: Se ha producido un error durante la ejecución. El parámetro STATUS devuelve información detallada sobre el tipo de error
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS Información de error (véase tabla "Parámetro STATUS")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

STATUS* (W#16#...)	Explicación
0000	Sin error.
0001	La conexión no se ha establecido.
7001	Se ha iniciado la interrupción de la conexión.
7002	Se está realizando la interrupción de la conexión.
8081	La conexión indicada en el parámetro ID no se conoce.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

T_DIAG: Comprobar conexión

Descripción

La instrucción "T_DIAG" sirve para comprobar el estado de una conexión y leer información adicional sobre el punto final local de dicha conexión.

- La conexión se referencia a través del parámetro ID. Pueden leerse puntos finales de la conexión tanto configurados en el editor de conexiones como programados (p. ej., con la instrucción "TCON".
  Los puntos finales temporales de la conexión (originados, p. ej., al conectar una estación de ingeniería) no pueden diagnosticarse porque en este caso tampoco se crea una ID de conexión.
- La información leída sobre la conexión se guarda en una estructura referenciada a través del parámetro RESULT.
- El parámetro de salida STATUS indica si ha sido posible leer la información sobre la conexión. Únicamente si la instrucción "T_DIAG" ha finalizado con STATUS = W#16#0000 y ERROR = FALSE, es válida la información sobre la conexión en la estructura del parámetro RESULT.
  En caso de error, no puede evaluarse la información sobre la conexión.

Posible información sobre la conexión

Para leer la información sobre la conexión, en el parámetro RESULT pueden utilizarse dos estructuras diferentes:

- La estructura "TDiag_Status" solo contiene la principal información sobre un punto final de la conexión, como puede ser el protocolo utilizado, el estado de la conexión o la cantidad de datos enviados o recibidos.
- La estructura "TDiag_StatusExt" suministra, además de la información más relevante, el número de intentos de conexión, la causa de una posible interrupción en la conexión, etc.

El formato y los parámetros de ambas estructuras se describen más adelante (véase la tabla "Estructuras TDIAG_Status y TDIAG_StatusExt").

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_DIAG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Con un flanco positivo, inicia la instrucción para comprobar la conexión especificada en el parámetro ID.
ID	Input	CONN_OUC (WORD)	L, D o constante	Referencia a la conexión asignada. Rango de valores: W#16#0001 a W#16#0FFF

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESULT	InOut	VARIANT	D	Puntero hacia la estructura en la que se deposita la información sobre la conexión. En el parámetro RESULT pueden utilizarse las estructuras TDiag_Status o TDiag_StatusExt (para una descripción, véase la tabla "Estructuras TDIAG_Status y TDIAG_StatusExt"),
DONE	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: Instrucción no iniciada aún o en ejecución. • 1: Instrucción ejecutada sin errores.
BUSY	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: Instrucción no iniciada aún o ya finalizada. • 1: Instrucción no finalizada aún. No es posible iniciar una nueva petición.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: Ningún error. • 1: Con errores.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros BUSY, DONE y ERROR

El estado de ejecución de la instrucción "T_DIAG" se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una instrucción se ha ejecutado correctamente. El parámetro ERROR se activa si ocurren errores durante la ejecución de "T_DIAG".

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	-	-	La instrucción se está ejecutando.
0	1	0	La instrucción se ha ejecutado correctamente. Solo en este caso son válidos los datos en la estructura referenciada con RESULT.
0	0	1	La instrucción se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna instrucción nueva.

Parámetro STATUS

La tabla siguiente muestra el significado de los valores en el parámetro STATUS:

STATUS* (W#16#...)	Explicación
0000	La instrucción "T_DIAG" se ha ejecutado correctamente. Los datos pertenecientes a la estructura referenciada en el parámetro RESULT pueden evaluarse.
7000	No se está ejecutando ninguna instrucción.
7001	Se ha iniciado la ejecución de la instrucción.
7002	Se está leyendo la información sobre la conexión (parámetro REQ irrelevante).
8086	El valor del parámetro ID está fuera del rango permitido (W#16#0001 ... W#16#0FFF).
8089	El parámetro RESULT apunta a un tipo de datos no válido (solo se permiten estructuras TDIAG_Status y TDIAG_StatusExt).
80A3	El parámetro ID crea una referencia a un punto final de la conexión que no existe. Este error se produce con conexiones programadas también después de llamar la instrucción "TDISCON".
80C4	Error interno. En estos momentos no es posible acceder al punto final de la conexión.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Estructuras TDIAG_Status y TDIAG_StatusExt

La tabla siguiente describe el formato de las estructuras TDIAG_Status y TDIAG_StatusExt:

- La estructura TDIAG_StatusExt es idéntica a la estructura TDIAG_Status desde el parámetro InterfaceID hasta el parámetro ReceivedBytes.
- La estructura TDIAG_StatusExt contiene además los parámetros ConnTrials hasta LastDisconnTimeStamp.

Todos los elementos tienen un valor válido únicamente si la instrucción ha podido ejecutarse sin errores. En caso de error, el contenido de los parámetros no se modifica.

Nombre	Tipo de datos	Descripción
Los parámetros siguientes existen tanto en la estructura TDIAG_Status como en la estructura TDIAG_StatusExt:		
InterfaceID	HW_ANY	ID de interfaz (LADDR) de la CPU o del CM/CP.
ID	CONN_OUC	ID de la conexión que ha sido diagnosticada. Tras realizar la llamada con éxito, el valor de este elemento es idéntico al parámetro ID de la instrucción "T_DIAG".

Nombre	Tipo de datos	Descripción
ConnectionType	BYTE	Tipo de protocolo empleado para la conexión: • 0x01: no utilizado. • ... • 0x0B: protocolo TCP (IP_v4) • 0x0C: protocolo ISO on TCP (RFC1006) • 0x0D: protocolo TCP (DNS) • 0x0E: protocolo Dial in • 0x0F: protocolo WDC • 0x10: protocolo SMTP • 0x11: protocolo TCP • 0x12: protocolos TCP e ISO on TCP (RFC1006) • 0x13: protocolo UDP • 0x14: Reservado • 0x15: protocolo de acceso al bus PROFIBUS (FDL) • 0x16: protocolo de transporte ISO 8073 (ISO native) • ... • 0x20: protocolo SMTP o SMTPS, basado en IPv4 • 0x21: protocolo SMTP o SMTPS, basado en IPv6 • 0x22: protocolo SMTP o SMTPS, basado en FQDN (**F**ully **Q**ualified **D**omain **N**ame) • ... • 0x70: Conexión S7 • Otros: Reservado
ActiveEstablished	BOOL	• FALSE: a nivel local, el punto final pasivo de la conexión • TRUE: a nivel local, el punto final activo de la conexión
State	BYTE	Estado actual del punto final de la conexión: • 0x00: no utilizado. • 0x01: conexión finalizada. Estado temporal adoptado, p. ej., tras llamar la instrucción "T_RESET". A continuación, el sistema trata de restablecer automáticamente la conexión. • 0x02: el punto final activo de la conexión trata de establecer la conexión con el interlocutor remoto. • 0x03: el punto final pasivo de la conexión espera a que se establezca la conexión con el interlocutor remoto. • 0x04: conexión establecida. • 0x05: la conexión acaba de deshacerse. La causa de que se haya deshecho puede ser la llamada de la instrucción "T_RESET" o "T_DISCON". Otras causas posibles pueden ser un error de protocolo o una rotura de hilo. • 0x06..0xFF: no utilizado.

Nombre	Tipo de datos	Descripción
Kind	BYTE	Modo de operación del punto final de la conexión: • 0x00: no utilizado. • 0x01: conexión estática configurada y cargada después en la CPU. • 0x02: conexión dinámica configurada y cargada a continuación en la CPU (actualmente no se soporta). • 0x03: conexión programada, generada desde el programa de usuario con la instrucción "TCON". El punto final de la conexión ha vuelto a destruirse al llamar la instrucción "TDISCON" o al pasar la CPU al estado STOP. • 0x04: conexión dinámica temporal creada, p. ej., a través de la estación de ingeniería (ES) o la estación de operador (OS) (este tipo de conexión no puede diagnosticarse actualmente porque falta la ID). • 0x05..0xFF: no utilizado.
SentBytes	UDINT	Cantidad de datos enviados.
ReceivedBytes	UDINT	Cantidad de datos recibidos.
Los siguientes parámetros existen solo con la estructura TDiag_StatusExt:		
ConnTrials	UDINT	Número de intentos de conexión. Tras establecer con éxito una conexión, ConnTrials contiene el valor 0. Si el elemento difiere de 0, significa que puede haber problemas de conexión. Nota: con un punto final de la conexión pasivo, este valor nunca es mayor que 1.
ConnTrialsSuccess	UDINT	Número de intentos de conexión satisfactorios. Este elemento no se restablece nunca durante el ciclo de vida de un punto final de la conexión y, después de alcanzar el valor 0xFFFF FFFF, retorna a 0. Nota: si nunca ha habido problemas con esta conexión, este parámetro tiene el valor 1.
LastConnErrReason	UDINT	Identificación de error producida durante el último intento incorrecto de establecer una conexión (los mensajes de error son idénticos a los del parámetro LastDisconnReason): • 0x4F01: punto final remoto de la conexión no accesible (este error suele producirse durante la fase de establecimiento de la conexión). • 0x4F02: la conexión se ha deshecho a nivel local. • 0x4F03: la conexión se ha deshecho debido al interlocutor remoto. • 0x4F04: la conexión se ha deshecho debido a un error de protocolo. • 0x4F05: la conexión se ha deshecho debido a un problema de red de detección local. • 0x4F06: la conexión se ha deshecho debido a un problema de red de detección remota. • 0x4F07: la conexión se ha deshecho debido a un rebase de tiempo en el protocolo. • 0x4F08: parametrización errónea: la conexión debe establecerse con la dirección propia. • 0x4F09: la conexión se ha reiniciado temporalmente mediante la llamada de la instrucción "T_RESET". • 0x4F0A: insuficientes recursos de conexión disponibles (capacidad excedida) • 0x4F0B: Error interno: parámetros de direccionamiento incorrectos • 0x4F0C: error de comunicación interno de la CPU • 0x4F0D: error de comunicación interno del AS entre CPU y CM/CP • 0x4F0E: el puerto TCP/UDP local especificado (o el selector T RFC1006) ya se está utilizando.
LastConnErrTimeStamp	LDT	Instante del último intento de conexión incorrecto.

Nombre	Tipo de datos	Descripción
LastDisconnReason	UDINT	Identificación de error que ha provocado la última desconexión (los mensajes de error son idénticos a los del parámetro LastConnErrReason): • 0x4F01: punto final remoto de la conexión no accesible (este error suele producirse durante la fase de establecimiento de la conexión). • 0x4F02: la conexión se ha deshecho a nivel local. • 0x4F03: la conexión se ha deshecho debido al interlocutor remoto. • 0x4F04: la conexión se ha deshecho debido a un error de protocolo. • 0x4F05: la conexión se ha deshecho debido a un problema de red de detección local. • 0x4F06: la conexión se ha deshecho debido a un problema de red de detección remota. • 0x4F07: la conexión se ha deshecho debido a un rebase de tiempo en el protocolo. • 0x4F08: parametrización errónea: la conexión debe establecerse con la dirección propia. • 0x4F09: la conexión se ha reiniciado temporalmente mediante la llamada de la instrucción "T_RESET". • 0x4F0A: insuficientes recursos de conexión disponibles (capacidad excedida) • 0x4F0B: Error interno: parámetros de direccionamiento incorrectos • 0x4F0C: error de comunicación interno de la CPU • 0x4F0D: error de comunicación interno del AS entre CPU y CM/CP • 0x4F0E: el puerto TCP/UDP local especificado (o el selector T RFC1006) ya se está utilizando.
LastDisconnTimeStamp	LDT	Instante de la última desconexión.

T_CONFIG: Configurar interfaz

Descripción de T_CONFIG

Descripción

La instrucción "T_CONFIG" sirve para realizar la configuración controlada por el programa de las interfaces PROFINET integradas en la CPU o de la interfaz de un CP/CM.

La instrucción permite modificar la dirección Ethernet y el nombre de dispositivo PROFINET desde el programa de usuario. Los datos de configuración hasta ahora válidos se sobrescriben.

Los cambios siguientes se pueden efectuar con la instrucción "T_CONFIG":

- Configuración del protocolo IP
  - Dirección IP
  - Máscara de subred
  - Dirección del router
- Configuración de PROFINET
  - Asignación del nombre de dispositivo PROFINET

Los ajustes se corresponden con las opciones de configuración en "Protocolo IP" y "PROFINET" del cuadro de diálogo "Direcciones Ethernet". Este se visualiza en la vista "Dispositivos y redes" en las propiedades de la interfaz PROFINET.

⚠ ADVERTENCIA

Rearranque de la CPU tras ejecutar la instrucción "T_CONFIG"

Tras ejecutar la instrucción para modificar un parámetro IP, la CPU realiza un rearranque. La CPU pasa al estado operativo STOP, se ejecuta un rearranque en caliente y la CPU arranca de nuevo (estado operativo RUN).

Asegúrese de que el proceso controlado se encuentra en un estado operativo seguro después del rearranque de la CPU tras ejecutar la instrucción. Un funcionamiento incontrolado, provocado por ejemplo por fallos de funcionamiento o errores del programa, puede causar daños materiales y personales graves. Es posible que se pierdan datos no remanentes.

Requisitos

Para utilizar la instrucción, en la configuración hardware debe indicarse explícitamente que la asignación de los parámetros de dirección IP y del nombre de dispositivo deben efectuarse a través del programa de usuario.

- Para ello, abra las propiedades de la interfaz PROFINET en la Vista de dispositivos. Active las opciones siguientes en el cuadro de diálogo "Direcciones Ethernet":
  - Para modificar los parámetros de dirección IP con "T_CONFIG": seleccione en "Protocolo IP" el ajuste "Obtener dirección IP por otra vía".
  - Para modificar el nombre de dispositivo PROFINET con "T_CONFIG": seleccione en "PROFINET" el ajuste "Obtener nombre de dispositivos PROFINET por otra vía".
- Los datos de configuración deben guardarse en los siguientes tipos de datos de sistema y aplicarse en el parámetro CONF_DATA (Página 5340):
  - La dirección IP, la máscara de subred y la dirección del router se guardan en el tipo de datos de sistema IF_CONF_V4.
  - El nombre de dispositivo se guarda en el tipo de datos de sistema IF_CONF_NOS. Tenga en cuenta las restricciones que rigen para la asignación del nombre de dispositivo (ver parámetro CONF_DATA (Página 5340)).

Funcionamiento

La instrucción "T_CONFIG" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar el procedimiento de configuración, llame la instrucción "T_CONFIG" con REQ = 1. Solo puede haber una petición activa cada vez.

El bloque se ha disparado por flanco, lo cual significa que después de BUSY = FALSE el bloque debe volver a llamarse con REQ = FALSE para que se habilite la instancia.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_CONFIG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	La ejecución de la instrucción comienza al llamar la instrucción con REQ = 1.
INTERFACE	Input	HW_INTERFACE	I, Q, M, D, L o constante	ID de hardware de la interfaz La ID de hardware se visualiza en las propiedades de la interfaz en la vista de dispositivos y en las constantes de sistema de las variables PLC.
CONF_DATA (Página 5340)	Input	VARIANT	D, L	Puntero hacia la estructura superior que contiene los tipos de datos de sistema IF_CONF_HEADER, IF_CONF_V4 y IF_CONF_NOS (véase la descripción del parámetro CONF_DATA).
DONE (Página 5344)	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: el procesamiento todavía no ha finalizado • 1: procesamiento de la instrucción finalizado correctamente.
BUSY (Página 5344)	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: El procesamiento de la instrucción aún no ha comenzado, aún no ha terminado o aún no se ha interrumpido • 1: La instrucción se está procesando
ERROR (Página 5344)	Output	BOOL	I, Q, M, D, L	Parámetros de estado: • 0: Sin errores • 1: Con errores
STATUS (Página 5344)	Output	DWORD	I, Q, M, D, L	Información de estado detallada: En el parámetro STATUS se visualiza información de estado y error detallada a través de un código de error.
ERR_LOC (Página 5344)	Output	DWORD	I, Q, M, D, L	Ubicación del error: • 0: Error al ejecutar la instrucción o la parametrización. • > 0: Error en la estructura o el contenido de los datos de configuración en el parámetro CONF_DATA.

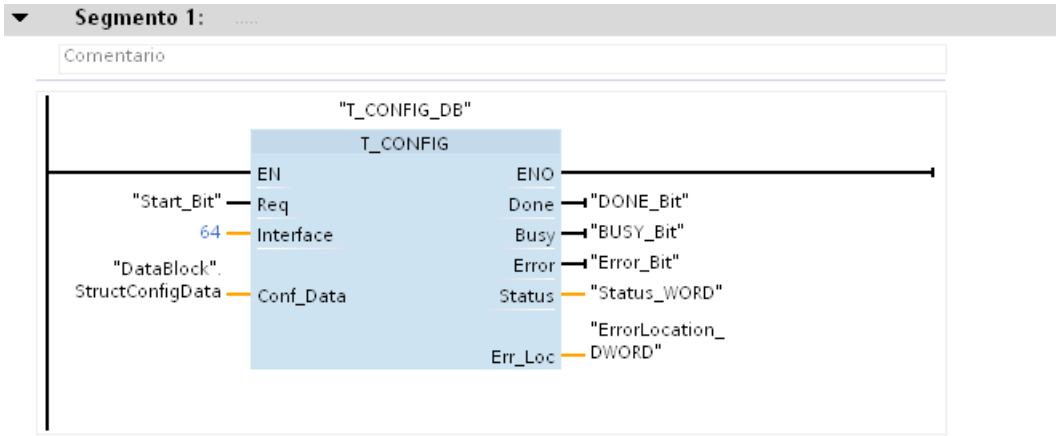
Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Ejemplo

En el ejemplo siguiente se modifica el nombre de un dispositivo IO a través de la instrucción "T_CONFIG".

Llamada de la instrucción

- La ejecución de la instrucción se inicia con REQ=1.
- En el parámetro Interface está especificada la identificación de hardware de la interfaz PROFINET.
- En el parámetro CONF_DATA está referenciada la estructura "StructConfigData". La estructura se ha creado en el bloque de datos global "DataBlock".



Parámetro CONF_DATA

La estructura "StructConfigData" en el parámetro CONF_DATA contiene la información siguiente:

- En Header (IF_CONF_HEADER).
  SubfieldCount = 1: indica que a continuación solo se utilizará una estructura adicional (nos).
- En la estructura "nos" (tipo de datos de sistema IF_CONF_NOS)
  - Lenght = 11: información sobre la longitud total de la estructura NOS (5 bytes para el nombre de dispositivo "myplc" + 6 bytes para los parámetros Id, Length y Mode)
    Nota: En lugar de indicar la longitud absoluta, también se puede utilizar el valor de arranque predeterminado (Lenght = 0) para una longitud dinámica.
  - Mode = 1: cambio permanente del nombre de dispositivo por "myplc".
  - NOS[1] ... NOS[5]: nombre de dispositivo (1 carácter / byte)

DataBlock			
	Nombre	Tipo de datos	Valor de arranque
1	Static		
2	StructConfigData	Struct	
3	Header	IF_CONF_Header	
4	FieldType	UInt	0
5	FieldId	UInt	0
6	SubfieldCount	UInt	1
7	nos	IF_CONF_NOS	
8	Id	UInt	40
9	Length	UInt	11
10	Mode	UInt	1
11	NOS	array [1..240] of Byte	
12	NOS[1]	Byte	'm'
13	NOS[2]	Byte	'y'
14	NOS[3]	Byte	'p'
15	NOS[4]	Byte	'l'
16	NOS[5]	Byte	'c'
17	NOS[6]	Byte	16#0
18	NOS[7]	Byte	16#0
19	NOS[8]	Byte	16#0
20	NOS[9]	Byte	16#0
21	NOS[10]	Byte	16#0
22	NOS[11]	Byte	16#0

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Parámetro CONF_DATA

Estructura de los datos de configuración

Los datos de configuración en el parámetro CONF_DATA se pueden guardar en un bloque de datos global o en la sección "Static" de la interfaz del bloque.

Los datos de configuración deben guardarse de acuerdo con la estructura siguiente:

Nombre	Tipo de datos	Descripción
ConfData	Struct	Estructura superior que se aplica en el parámetro "CONF_DATA".
Header	IF_CONF_HEADER	Mediante el encabezado se define el número de los siguientes tipos de datos de sistema. El tipo de datos de sistema IF_CONF_HEADER siempre debe estar incluido.
IPData	IF_CONF_V4	En este tipo de datos de sistema se guardan la dirección IP, la máscara de subred y la dirección del router. Cree IF_CONF_V4 únicamente si a través de "T_CONFIG" también desea modificar la dirección Ethernet.
NoS	IF_CONF_NOS	En este tipo de datos de sistema se guarda el nombre de dispositivo PROFINET. Cree IF_CONF_NOS únicamente si a través de "T_CONFIG" también desea modificar el nombre de dispositivo.

Los tipos de datos de sistema IF_CONF_HEADER, IF_CONF_V4 y IF_CONF_NOS se crean introduciendo el nombre del tipo de datos de sistema en la columna "Tipo de datos" del bloque

de datos o de la interfaz del bloque. Los nombres para los tipos de datos de sistema se pueden asignar libremente.

Tipo de datos de sistema IF_CONF_Header

El tipo de datos de sistema IF_CONF_Header permite indicar cuántos tipos de datos de sistema IF_CONF_V4 y IF_CONF_NOS deben utilizarse al ejecutar "T_CONFIG".

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 ... 1	FieldType	UINT	0	Tipo de campo: debe tener siempre el valor "0".
2 ... 3	FieldId	UINT	0	ID de campo: debe tener siempre el valor "0".
4 ... 5	SubfieldCount	UINT	0	Número de tipos de datos de sistema IF_CONF_V4 y IF_CONF_NOS utilizados: • 1: solo se utiliza uno de los tipos de datos de sistema. • 2: se utilizan ambos tipos de datos de sistema.

Tipo de datos de sistema IF_CONF_V4

Con el tipo de datos de sistema IF_CONF_V4 se definen la dirección IP, la máscara de subred y la dirección del router.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 ... 1	Id	UINT	30	Identificación del tipo de datos de sistema. No está permitido modificar el valor de arranque de este parámetro.
2 ... 3	Length	UINT	18	Longitud del tipo de datos de sistema IF_CONF_V4 Dado que los parámetros de IF_CONF_V4 tienen una longitud y estructura fijas, para indicar la longitud debe utilizarse el valor de arranque.
4 ... 5	Mode	UINT	0	Validez del direccionamiento: • 1: Validez permanente de los datos de configuración • 2: Validez temporal de los datos de configuración, incluido el borrado de los datos de configuración permanentes existentes
6 ... 9	InterfaceAddress	IP_V4 *	0.0.0.0	Dirección IP
10 ... 12	SubnetMask	IP_V4 *	0.0.0.0	Máscara de subred
14 ... 16	DefaultRouter	IP_V4 *	0.0.0.0	Dirección del router
* El tipo de datos IP_V4 es una estructura de 4 BYTE que contiene la dirección del parámetro correspondiente (p. ej., en el parámetro SubnetMask, la dirección de cuatro dígitos de la máscara de subred del protocolo IP).				

Subcampo IF_CONF_NOS

Con el subcampo IF_CONF_NOS se determinan los nombres de dispositivo que se deben asignar al ejecutar la instrucción "T_CONFIG".

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 ... 1	Id	UINT	40	Identificación del tipo de datos de sistema. No está permitido modificar el valor de arranque de este parámetro.
2 ... 3	Length	UINT	246	Longitud del tipo de datos de sistema IF_CONF_NOS en bytes. • Para una longitud absoluta, el valor para el parámetro Lenght resulta de: – 6 bytes para los parámetros Id, Length y Mode. – Hasta 240 bytes para el nombre de dispositivo (parámetro NOS). Ejemplo: para el nombre de dispositivo "plc1" con una longitud de 4 caracteres (= 4 bytes) resulta una longitud total de 10. • Para una longitud dinámica utilice el valor de arranque predeterminado 246 en el parámetro Lenght. Asegúrese de introducir el valor "0" tras el nombre (consulte la descripción del parámetro NOS).

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
4 ... 5	Mode	UINT	0	Validez del cambio de nombre de dispositivo: • 1: validez permanente del nombre de dispositivo. • 2: validez temporal del nombre de dispositivo.
6 ... 244	NOS	ARRAY [1...240] of Byte	0	Nombre de dispositivo (Name of Station) • ARRAY debe ocuparse desde el primer byte. Si se ocupa el primer byte con "0", se borra el nombre de estación. • La longitud mínima para el nombre es de 2 bytes. La longitud máxima para el nombre es de 240 bytes. • Si el nombre de dispositivo es más corto que el especificado en el parámetro Length, se debe introducir un byte cero (16#0 hex) después del nombre de estación (de acuerdo con IEC 61185-6-10). De lo contrario, NOS es rechazado y la instrucción "T_CONFIG" visualiza el código de error DW#16#C0809400 en el parámetro STATUS. • Si el nombre de dispositivo es más largo que el especificado en el parámetro Length, el nombre de dispositivo se escribe solo hasta la longitud especificada. Para el nombre de dispositivo rigen las siguientes restricciones: • El nombre debe introducirse en código ASCII. • Para el nombre solo está permitido utilizar letras minúsculas, cifras, guiones o puntos. – El nombre no puede empezar ni acabar con un guión. – El nombre no puede empezar con cifras. – El nombre no puede tener el formato n.n.n.n (n = 0... 999). – El nombre no puede empezar con la cadena de caracteres "port-xyz" o "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0...9). • Una parte integrante del nombre entre dos puntos puede tener máximo 63 caracteres. • No se admiten caracteres especiales como vocales acentuadas o con diéresis, paréntesis, caracteres de subrayado, barras, espacios en blanco, etc. Si se utiliza un carácter no válido, el parámetro STATUS devuelve el código de error C080_9400.

Parámetros DONE, BUSY y ERROR

Descripción

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra la instrucción actualmente o cuándo se completó la transferencia de los datos de configuración.

BUSY	DONE	ERROR	Descripción
TRUE	FALSE	FALSE	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS (Página 5344).
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Parámetros STATUS y ERR_LOC

Descripción

En los parámetros STATUS y ERR_LOC se visualizan los mensajes de estado y error de la instrucción "T_CONFIG":

- En el parámetro STATUS se visualiza la causa del error.
- El parámetro ERR_LOC indica dónde se ha producido el error. Para ello se dispone de las siguientes posibilidades:
  - 16#0000_0000: error al llamar la instrucción (p. ej. error al parametrizar la instrucción o en la comunicación con la interfaz PROFINET).
  - 16#0001_0000: error en los datos de configuración en los parámetros del tipo de datos de sistema IF_CONF_HEADER.
  - 16#0001_0001: error en los datos de configuración en los parámetros de los tipos de datos de sistema IF_CONF_V4 o IF_CONF_NOS.

La tabla siguiente muestra los valores posibles de los parámetros STATUS y ERR_LOC:

STATUS*	ERR_LOC*	Explicación
0000_0000	0000_0000	Procesamiento de la petición finalizado sin errores.
0070_0000	0000_0000	No se está procesando ninguna petición.
0070_0100	0000_0000	Inicio del procesamiento de la petición.
0070_0200	0000_0000	Llamada intermedia (REQ irrelevante).
C08x_yy00	0000_0000	Información de error general. Consulte también: GET_ERR_ID: Consultar ID de error localmente (Página 3531)
C080_8000	0000_0000	Error al llamar la instrucción: El identificador de hardware del parámetro Interface no es válido.

STATUS*	ERR_LOC*	Explicación
C080_8100	0000_0000	Error al llamar la instrucción: El identificador de hardware del parámetro Interface no direcciona ninguna interfaz PROFINET.
C080_8700	0000_0000	Error al llamar la instrucción: Longitud incorrecta del bloque de datos en el parámetro CONF_DATA.
C080_8800	0001_0000	Error en el tipo de datos de sistema IF_CONF_HEADER: El parámetro FieldType tiene un valor no admisible. Utilice el valor "0" para FieldType.
C080_8900	0001_0000	Error en el tipo de datos de sistema IF_CONF_HEADER: El parámetro FieldId tiene un valor no admisible o se ha utilizado de forma repetida. Utilice el valor "0" para FieldId.
C080_8A00	0001_0000	Error en el tipo de datos de sistema IF_CONF_HEADER: Número incorrecto en el parámetro SubfieldCount. Indique el número correcto de tipos de datos de sistema IF_CONF_V4 y IF_CONF_NOS utilizados.
C080_8B00	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: El parámetro Id tiene un valor no admisible. Utilice para IF_CONF_V4 "30" y para IF_CONF_NOS "40".
C080_8C00	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: Tipo de datos de sistema erróneo, orden erróneo o utilización repetida de un tipo de datos de sistema.
C080_8D00	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: El parámetro Length tiene un valor erróneo o no admisible.
C080_8E00	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: El parámetro Mode tiene un valor erróneo o no admisible. Solo son válidos los valores "1" (permanente) o "2" (temporal).
C080_9000	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: Los datos de configuración no se pueden aplicar. Causa posible: en la configuración hardware no se ha seleccionado el ajuste "Ajustar dirección IP en el dispositivo" o "Ajustar nombre de dispositivo PROFINET en el dispositivo".
C080_9400	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: Un valor de parámetro no está definido o no es admisible.
C080_9500	0001_0001	Error en el tipo de datos de sistema IF_CONF_V4 o IF_CONF_NOS: Los valores de dos parámetros son incoherentes.
C080_C200	0000_0000	Error al llamar la instrucción: La transferencia de los datos de configuración no puede ejecutarse. Causa posible: la interfaz PROFINET no es accesible.
C080_C300	0000_0000	Error al llamar la instrucción: Recursos insuficientes (p. ej. llamada múltiple de "T_CONFIG" con parámetros diferentes).
C080_C400	0000_0000	Error al llamar la instrucción: Error de comunicación temporal. Intente llamar la instrucción más tarde.
C080_D200	0000_0000	Error al llamar la instrucción: La llamada no es posible. La interfaz PROFINET seleccionada no soporta la instrucción.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Servidor web

WWW: Sincronizar las páginas web definidas por el usuario

Descripción

La instrucción WWW inicializa el servidor web de la CPU, o bien sincroniza las páginas web definidas por el usuario con el programa de usuario en la CPU.

Las páginas web definidas por el usuario, junto con el servidor web de la CPU, ofrecen la posibilidad de acceder con un navegador web a páginas web de libre configuración de la CPU.

Mediante instrucciones de scripts (por ej., Javascript) y de código HTML en páginas web definidas por el usuario, es posible transmitir datos a la CPU desde un navegador para su posterior procesamiento, así como visualizar en el navegador datos del área de operandos. Para la sincronización del programa de usuario con el servidor web, y también para la inicialización, es necesario llamar la instrucción WWW en el programa de usuario.

Inicialización

Las páginas web definidas por el usuario se "comprimen" en bloques de datos para que la CPU pueda procesarlas. Durante la configuración deberá generar los correspondientes bloques de datos a partir de los datos fuente (archivos HTML, imágenes, archivos Javascript...). El DB Web Control (predeterminado: DB 333), que contiene información de estado y de control, así como referencias a otros bloques de datos con páginas web encriptadas, desempeña una función particular. Los bloques de datos con páginas web codificadas se denominan DBs de fragmento.

Si los bloques de datos se han cargado en la CPU, esta no "sabe" que las páginas web definidas por el usuario que contiene están encriptadas. Por medio de la instrucción "WWW", por ejemplo, en el OB de arranque, se notifica a la CPU qué DB es el DB de Web Control. Tras esta inicialización, es posible acceder a las páginas web definidas por el usuario desde un navegador web.

Sincronización

Cuando el programa de usuario debe interactuar con las páginas web definidas por el usuario, se debe utilizar la instrucción WWW en la parte cíclica del programa.

Ejemplos de interacción entre programa de usuario y página web:

- Comprobar los datos recibidos
- Recopilar y reenviar los datos para el navegador web solicitante

En este caso, la información de estado ha de poder evaluarse en el acto, debiendo notificarse al servidor la información de control, por ejemplo, para habilitar una página web solicitada por el navegador web.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WWW":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CTRL_DB	Input	DB_WWW	D	Bloque de datos que describe las páginas web definidas por el usuario (DB Web Control)
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en Vista general de los tipos de datos válidos (Página 2321).

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error. No hay pendiente ninguna solicitud de página web que deba ser habilitada por el programa de usuario.
00xy	x: indica si ha ocurrido un error al inicializar el DB Web Control (CTRL_DB): x=0: no ha ocurrido ningún error. x=1: se han producido errores. El error está codificado en el byte "CTRL_DB.last_error" del DB Web Control, consulte la descripción del DB Web Control. y: número de peticiones pendientes. Son posibles varias peticiones (p. ej., petición "0" y "1" pendientes: y="3". y="1": petición "0" y="2": petición "1" y="4": petición "2" y="8": petición "3"
803A	El DB Web Control indicado no se encuentra en la CPU.
8081	La versión o el formato del DB Web Control son incorrectos.
80C1	No hay recursos para inicializar la aplicación web.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Procesador de comunicaciones

CM/CP SIMATIC NET

CM/CP S7-1500

Industrial Ethernet

Instrucciones para servicios FTP

FTP_CMD para servicios FTP

Panorámica de FTP_CMD

Significado

Con la instrucción FTP_CMD se pueden establecer conexiones FTP y transmitir archivos desde y a un servidor FTP.

La transferencia de datos se puede realizar vía FTP o FTPS (conexiones SSL seguras).

Nota

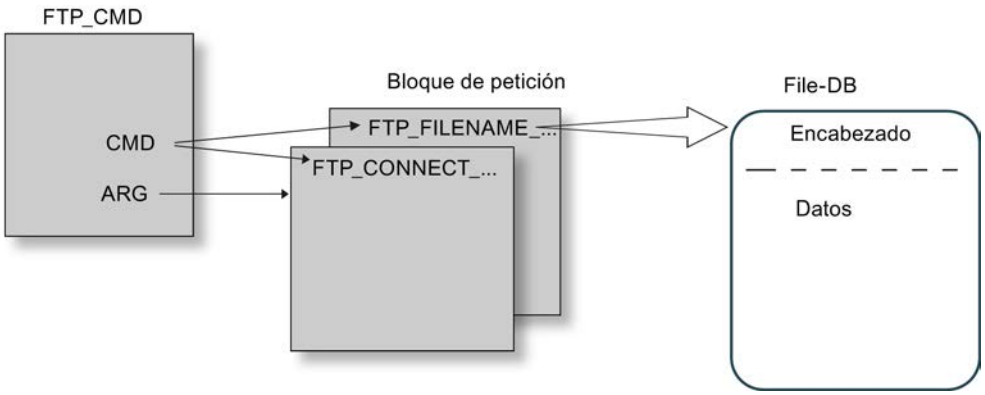
FTPS: sincronizar certificados

FTPS requiere la sincronización de los certificados entre el servidor FTP y el cliente FTP. Si el servidor FTP se configura fuera del proyecto STEP 7 del cliente FTP, habrá que importar el certificado del servidor FTP. En el administrador de certificados importe el certificado del servidor FTP como certificado de confianza.

Funcionamiento

La instrucción FTP_CMD hace referencia a un bloque de petición (ARG) en el que se especifica el comando FTP. Según sea el tipo de comando FTP (CMD), este bloque de petición utiliza diferentes estructuras de datos para la parametrización. Para dichas estructuras están disponibles en cada caso los tipos de datos adecuados (UDT).

La figura siguiente ilustra la estructura de llamada:



Bloques de petición

Para los bloques de petición se utilizan las siguientes estructuras de datos:

- Establecimiento de la conexión
  Para establecer la conexión están disponibles diferentes estructuras de datos para los siguientes tipos de acceso:
  - UDT FTP_CONNECT_IPV4: Establecimiento de la conexión con direcciones IP según IPv4
  - UDT FTP_CONNECT_IPV6: Establecimiento de la conexión con direcciones IP según IPv6
  - UDT FTP_CONNECT_NAME: Establecimiento de la conexión con nombres de servidor (DNS)
- Transferencia de datos
  Para la transferencia de datos están disponibles dos estructuras de datos distintas:
  - UDT FTP_FILENAME: Estructura de datos para el acceso a un archivo completo
  - UDT FTP_FILENAME_PART: Estructura de datos para el acceso de lectura a un área de datos

Transferencia de datos en el File-DB

Los datos se transfieren mediante bloques que contienen un encabezado para datos de petición y el área de datos de usuario. El bloque de datos se indica en el búfer de peticiones.

Requisitos en la configuración de la CPU

Utilice los ajustes siguientes para permitir el acceso FTP:

- En los datos de configuración de la CPU, en "Propiedades > General > Protección": desactive la opción "Desactivar comunicación PUT/GET".
- En todos los bloques de datos utilizados como File-DB desactive el atributo "Acceso optimizado al bloque"

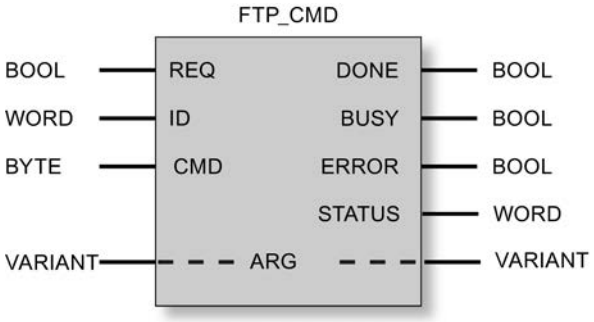
Validez

La instrucción FTP_CMD se puede utilizar en combinación con los siguientes tipos de módulos:

- CP 1543-1

Interfaz de llamada

Interfaz de llamada en representación FUP



Consulte también

Parámetros de salida e informaciones de estado FTP_CMD (Página 5358)

Parámetros de entrada FTP_CMD (Página 5350)

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Client (Página 5361)

Parámetros de entrada FTP_CMD

Explicación de los parámetros de entrada

Deben suministrarse los siguientes parámetros de entrada a la instrucción FTP_CMD:

Tabla 9-97 Parámetros formales de la instrucción FTP_CMD - parámetros de entrada

Parámetro	Declaración	Tipo de datos	Área de memoria	Significado / Observación
REQ	Input	BOOL	I, Q, M, D, L	Inicia la petición de emisión con un flanco ascendente.
ID *	INPUT	INT	1, 2 ... 64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.

Parámetro	Declaración	Tipo de datos	Área de memoria	Significado / Observación
CMD *	INPUT	BYTE	Consulte la tabla siguiente "Comandos".	Comando FTP que debe ejecutarse al llamar la instrucción. Encontrará los rangos de valores para los tipos de comandos FTP a continuación de la tabla. Nota: El comando FTP indicado aquí debe indicarse de forma idéntica en el bloque de petición (parámetro ARG). Si un comando no es compatible con el firmware del CP, se emite un mensaje de error con STATUS = $8F6B_H$.
ARG *	INPUT	VARIANT	Consulte la tabla siguiente "Comandos".	Bloque de petición Hace referencia al área de datos con los parámetros de ejecución adecuados para el comando FTP. En función del comando FTP se utilizan tipos de datos específicos (UDT). Los UDT se indican del siguiente modo. El tipo de datos ANY no está permitido para el puntero que debe especificarse aquí.

* Los valores de los parámetros de entrada "ID" y "CMD" sobrescriben el valor del parámetro de entrada "ARG".

Comandos FTP en el parámetro "CMD"

Consulte en la tabla siguiente qué significado tienen los comandos del parámetro "CMD" y qué UDTs se utilizan para abastecer a los bloques de petición.

Tabla 9-98 Tipos de comando

CMD (tipo de comando)	Bloques de petición relevantes / UDT	Significado / Manejo
0 (NOOP)	*	El FC llamado no ejecuta ninguna acción. Las indicaciones de estado se ponen como sigue para este abastecimiento de parámetros: DONE=1; ERROR=0; STATUS=0
1 (CONNECT)	FTP_CONNECT_IPV4 FTP_CONNECT_IPV6 FTP_CONNECT_NAME	Establecimiento de la conexión FTP El cliente FTP establece con este comando una conexión FTP con un servidor FTP (puerto 21). La conexión está disponible, con la ID de conexión aquí asignada, para todos los demás comandos FTP. Los datos se intercambian entonces con el servidor FTP indicado para este usuario.
2 (STORE)	FTP_FILENAME	Con esta llamada de función se transmite un bloque de datos (File-DB) del cliente FTP (CPU S7) al servidor FTP. Atención: Si el archivo (File-DB) ya existe en el servidor FTP, se sobrescribe.
3 (RETRIEVE)	FTP_FILENAME	Con esta llamada de función se transmite un archivo del servidor FTP al cliente FTP (CPU S7). Atención: Si el bloque de datos (File-DB) del cliente FTP ya contiene un archivo, este se sobrescribe.

CMD (tipo de comando)	Bloques de petición relevantes / UDT	Significado / Manejo
4 (DELETE)	FTP_FILENAME	Con esta llamada de función se borra un archivo en el servidor FTP.
5 (QUIT)	*	Con esta llamada de función se deshace la conexión FTP identificada con la ID.
6 (APPEND)	FTP_FILENAME	De forma similar a "STORE", el comando "APPEND" (añadir) guarda un archivo en el servidor FTP. Pero con "APPEND" no se sobrescribe el archivo en el servidor FTP, sino que el nuevo contenido a guardar se añade al archivo. Si el archivo no existe en el servidor FTP, se crea.
7 (RETR_PART)	FTP_FILENAME_PART	Con el comando "RETR_PART" (leer longitud parcial) se puede pedir del servidor FTP una parte de un archivo. En caso de archivos muy grandes se puede limitar así la lectura a la parte necesaria. Para esto se tiene que conocer la estructura del archivo. Indique la parte deseada del archivo con ayuda de los dos parámetros "OFFSET" y "LEN" en el FB 40.

* En los tipos de comando 0 (NOOP) y 5 (QUIT) debe indicarse un bloque de petición cualquiera (UDT). Dicho bloque no se evaluará.

Consulte también

Panorámica de FTP_CMD (Página 5348)

Bloques de petición para FTP_CMD

Significado

La instrucción FTP_CMD se abastece con un bloque de petición a través del parámetro ARG. La estructura depende del tipo de comando FTP. Al utilizar los tipos de datos especificados (UDT), la instrucción reconoce el tipo del bloque de petición. A continuación se indican los tipos de datos (UDT) correspondientes a los siguientes bloques de petición:

- Establecimiento de la conexión FTP con dirección IP según IPv4
- Establecimiento de la conexión FTP con dirección IP según IPv6
- Establecimiento de la conexión FTP con nombre de servidor
- Acceso de escritura y lectura así como otros comandos FTP
- Comando FTP RETR_PART

Bloque de petición para establecimiento de la conexión FTP con dirección IP según IPv4

Para el establecimiento de la conexión FTP con dirección IP según IPv4 se utiliza la siguiente estructura de datos.

Tabla 9-99 UDT FTP_CONNECT_IPV4

Parámetro	Tipo	Rango de valores	Significado / Observación
LADDR	WORD		Dirección inicial del módulo Al llamar una instrucción se transfiere en el parámetro LADDR la dirección inicial de módulo del CP. La dirección inicial de módulo del CP puede consultarse en la configuración del CP en: "Propiedades>Direcciones>Entradas"
ID	INT	1, 2...64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.
ConnectionType	BYTE	0	Tipo de conexión "FTP"
ActiveEstablished	BOOL	TRUE	
FTPcmd	BYTE	1	Comando FTP "CONNECT" Comando FTP que se ejecutará al llamar la instrucción. Encontrará los rangos de valores en Tabla 9-98 Tipos de comando (Página 5351) Nota: El comando FTP indicado aquí debe indicarse de forma idéntica en el parámetro de entrada CMD.
CertIndex	BYTE	0 = FTP 1 = FTPS	Elija o bien el tipo de protocolo FTP, o bien FTPS. Nota para FTPS: Si el servidor FTP se configura fuera del proyecto STEP 7 del cliente FTP, habrá que importar el certificado del servidor FTP.
UserName	STRING[32]	'usuario'	Nombre del usuario para iniciar sesión en el servidor FTP
Password	STRING[32]	'contraseña'	Contraseña para iniciar sesión en el servidor FTP
FTPserverIP_addr	IPv4	'142.11.25.135'	Dirección IP del servidor FTP

Bloque de petición para establecimiento de la conexión FTP con dirección IP según IPv6

Para el establecimiento de la conexión FTP con dirección IP según IPv6 se utiliza la siguiente estructura de datos.

Tabla 9-100 UDT FTP_CONNECT_IPV6

Parámetro	Tipo	Rango de valores	Significado / Observación
LADDR	WORD		Dirección inicial del módulo Al llamar una instrucción se transfiere en el parámetro LADDR la dirección inicial de módulo del CP. La dirección inicial de módulo del CP puede consultarse en la configuración del CP en: "Propiedades>Direcciones>Entradas"
ID	INT	1, 2...64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.
ConnectionType	BYTE	0	Tipo de conexión "FTP"
ActiveEstablished	BOOL	TRUE	
FTPcmd	BYTE	1	Comando FTP "CONNECT" Comando FTP que se ejecutará al llamar la instrucción. Encontrará los rangos de valores en Tabla 9-98 Tipos de comando (Página 5351) Nota: El comando FTP indicado aquí debe indicarse de forma idéntica en el parámetro de entrada CMD.
CertIndex	BYTE	0 = FTP 1 = FTPS	Elija o bien el tipo de protocolo FTP, o bien FTPS. Nota para FTPS: Si el servidor FTP se configura fuera del proyecto STEP 7 del cliente FTP, habrá que importar el certificado del servidor FTP.
UserName	STRING[32]	'usuario'	Nombre del usuario para iniciar sesión en el servidor FTP
Password	STRING[32]	'contraseña'	Contraseña para iniciar sesión en el servidor FTP
FTPserverIP_addr	IPv6 array [1..16] of Byte		Dirección IP del servidor FTP

Bloque de petición para establecimiento de la conexión FTP con nombre de servidor

Para el establecimiento de la conexión FTP indicando un nombre de servidor se utiliza la siguiente estructura de datos. El nombre de servidor se asigna a una dirección IP mediante DNS.

Tabla 9-101 UDT FTP_CONNECT_NAME

Parámetro	Tipo	Rango de valores	Significado / Observación
LADDR	WORD		Dirección inicial del módulo Al llamar una instrucción se transfiere en el parámetro LADDR la dirección inicial de módulo del CP. La dirección inicial de módulo del CP puede consultarse en la configuración del CP en: "Propiedades>Direcciones>Entradas"
ID	INT	1, 2...64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.
ConnectionType	BYTE	0	Tipo de conexión "FTP"
ActiveEstablished	BOOL	TRUE	
FTPcmd	BYTE	1	Comando FTP "CONNECT" Comando FTP que se ejecutará al llamar la instrucción. Encontrará los rangos de valores en Tabla 9-98 Tipos de comando (Página 5351) Nota: El comando FTP indicado aquí debe indicarse de forma idéntica en el parámetro de entrada CMD.
CertIndex	BYTE	0 = FTP 1 = FTPS	Elija o bien el tipo de protocolo FTP, o bien FTPS. Nota para FTPS: Si el servidor FTP se configura fuera del proyecto STEP 7 del cliente FTP, habrá que importar el certificado del servidor FTP.
UserName	STRING[32]	’usuario’	Nombre del usuario para iniciar sesión en el servidor FTP
Password	STRING[32]	’contraseña’	Contraseña para iniciar sesión en el servidor FTP
FTPserverName	STRING[254]		Dirección IP del servidor FTP

Bloque de petición para acceso de escritura y lectura así como otros comandos FTP

Para los comandos FTP store, retrieve, delete y append se utiliza la siguiente estructura de datos.

Tabla 9-102 UDT FTP_FILENAME

Parámetro	Tipo	Rango de valores	Significado / Observación
LADDR	WORD		Dirección inicial del módulo Al llamar una instrucción se transfiere en el parámetro LADDR la dirección inicial de módulo del CP. La dirección inicial de módulo del CP puede consultarse en la configuración del CP en: "Propiedades>Direcciones>Entradas"
ID	INT	1, 2...64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.
ConnectionType	BYTE	0	Tipo de conexión "FTP"
ActiveEstablished	BOOL	TRUE	
FTPcmd	BYTE	2, 3, 4, 6	Comando FTP "STORE / RETRIEVE / DELETE / APPEND" Comando FTP que se ejecutará al llamar la instrucción. Encontrará los rangos de valores en Tabla 9-98 Tipos de comando (Página 5351) Nota: El comando FTP indicado aquí debe indicarse de forma idéntica en el parámetro de entrada CMD.
CertIndex	BYTE	0 = FTP 1 = FTPS	Elija o bien el tipo de protocolo FTP, o bien FTPS. Nota para FTPS: Si el servidor FTP se configura fuera del proyecto STEP 7 del cliente FTP, habrá que importar el certificado del servidor FTP.
DataBlockNumber	UINT		El bloque de datos aquí indicado contiene el File-DB que se debe leer / escribir.
LenFilename	UINT	0...1000	El parámetro "LenFilename" que indica la longitud total del nombre de archivo no se evalúa. En su lugar se evalúa la longitud indicada en el string del parámetro "Filename".
Filename	ARRAY[0..3] OF STRING[220]		Nombre del archivo de origen o destino.

Bloque de petición para el comando FTP RETR_PART

Para el comando FTP RETR_PART se utiliza la siguiente estructura de datos.

Tabla 9-103 UDT FTP_FILENAME_PART

Parámetro	Tipo	Rango de valores	Significado / Observación
LADDR	WORD		Dirección inicial del módulo Al llamar una instrucción se transfiere en el parámetro LADDR la dirección inicial de módulo del CP. La dirección inicial de módulo del CP puede consultarse en la configuración del CP en: "Propiedades>Direcciones>Entradas"
ID	INT	1, 2...64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.
ConnectionType	BYTE	0	Tipo de conexión "FTP"
ActiveEstablished	BOOL	TRUE	
FTPcmd	BYTE	7	Comando FTP "RETR_PART" Comando FTP que se ejecutará al llamar la instrucción. Encontrará los rangos de valores en Tabla 9-98 Tipos de comando (Página 5351) Nota: El comando FTP indicado aquí debe indicarse de forma idéntica en el parámetro de entrada CMD.
CertIndex	BYTE	0 = FTP 1 = FTPS	Elija o bien el tipo de protocolo FTP, o bien FTPS. Nota para FTPS: Si el servidor FTP se configura fuera del proyecto STEP 7 del cliente FTP, habrá que importar el certificado del servidor FTP.
Offset	DWORD		Offset en bytes, a partir del cual se debe leer el archivo.
Length	DWORD		Longitud parcial en bytes que se debe leer a partir del valor indicado en "OFFSET". Peculiaridades: • Si se indica "DW#16#FFFFFFFF" se lee el resto disponible del archivo. Resultado OK (DONE = 1, STATUS = 0), si no se presenta ningún otro error. • Si OFFSET > longitud del archivo original: Longitud del archivo de destino (ACT_LENGTH en el File-DB): 0 bytes en la CPU. Resultado OK (DONE = 1, STATUS = 0), si no se presenta ningún otro error. • Si OFFSET + LEN > longitud del archivo original (y LEN ≠ 0xFFFFFFFF): Longitud del archivo de destino (ACT_LENGTH en el File-DB): Bytes disponibles a partir de "OFFSET". Resultado OK (DONE = 1, STATUS = 0), si no se presenta ningún otro error.
DataBlockNumber	UINT		El bloque de datos aquí indicado contiene el File-DB que se debe leer / escribir.

Parámetro	Tipo	Rango de valores	Significado / Observación
LenFilename	UINT	0...1000	El parámetro "LenFilename" que indica la longitud total del nombre de archivo no se evalúa. En su lugar se evalúa la longitud indicada en el string del parámetro "Filename".
Filename	ARRAY[0..3] OF STRING[220]		Nombre del archivo de origen o destino.

Consulte también

Panorámica de FTP_CMD (Página 5348)

Parámetros de entrada FTP_CMD (Página 5350)

Parámetros de salida e informaciones de estado FTP_CMD

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla con los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY muestra el estado de procesamiento. Con el parámetro DONE se controla si una petición se ha ejecutado correctamente. El parámetro ERROR se activa cuando se producen errores durante la ejecución de "FTP_CMD". La información se error se visualiza en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	-	-	La petición se procesa.
0	1	0	La petición se ha llevado a cabo correctamente.
0	0	1	La petición ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Evaluar indicaciones de estados

Nota

Para las entradas con la codificación 8FxxH en STATUS, tenga en cuenta también lo dicho en el manual de referencia STEP 7 Standard y funciones del sistema. Allí encontrará informaciones en el capítulo "Evaluación de fallos con el parámetro de salida RET_VAL".

Tabla 9-104 FTP_CMD: Significado del parámetro STATUS en relación con DONE y ERROR

DONE	ERROR	STATUS	Significado
0	0	0000_H	No hay ninguna petición en proceso.
1	0	0000_H	La petición ha terminado sin errores.
0	0	7001_H	La petición ha sido lanzada por primera vez (BUSY=1)
0	0	7002_H	La petición aún está procesándose (BUSY=1)
0	0	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.)
0	0	8183_H	La configuración no concuerda con los parámetros de la petición.
0	1	8401_H	Error desconocido Posibles causas: • En la conexión se detectó un timeout. • El servidor FTP canceló la conexión. Solución: • Enviar los comandos QUIT y CONNECT para volver a crear la conexión.
0	1	8402_H	Estado de la conexión incorrecto. Es posible que se haya excedido el timeout de la conexión o el servidor FTP puede haber deshecho la conexión. Solución: Enviar nuevamente los comandos QUIT y CONNECT para volver a establecer la conexión.
0	1	8403_H	Error al iniciar sesión.
0	1	8404_H	Servidor FTP no accesible.
0	1	8405_H	Error en la transferencia.
0	1	8406_H	Timeout en la operación actual
0	1	8407_H	No se ha encontrado el archivo en el servidor FTP.
0	1	8408_H	No es posible transferir.
0	1	8409_H	No se ha podido recuperar el archivo.
0	1	8410_H	Error el activar el puerto TCP para la conexión de datos.
0	1	8411_H	La indicación de offset no concuerda.
0	1	8412_H	Error al cambiar la indicación de directorio
0	1	8413_H	Error al recibir datos
0	1	8414_H	Error al enviar datos.
0	1	8415_H	El cliente ha rechazado la indicación CMD (tipo de comando).
0	1	8416_H	El servidor FTP ha cerrado la conexión.
0	1	8418_H	Error en los datos de usuario. Posibles causas: • El nombre del archivo está vacío. • La longitud de datos es "0" • etc.
0	1	8419_H	No hay recursos de socket para abrir una conexión de datos.
0	1	8420_H	No hay recursos de socket para abrir una conexión de control.
0	1	8421_H	Error al abrir el File-DB que debe leerse.
0	1	8422_H	Error al abrir el File-DB que debe escribirse.

DONE	ERROR	STATUS	Significado
0	1	8423_H	Error al establecer la conexión con el servidor FTP.
0	1	8424_H	Error interno
0	1	8425_H	Error de formato en el nombre del dominio
0	1	8426_H	Hay demasiadas peticione DNS.
0	1	8427_H	El servidor DNS indicado no ha podido asignar el nombre de dominio especificado.
0	1	8428_H	No hay ningún recurso de conexión disponible.
0	1	8429_H	ID de canal desconocida
0	1	8430_H	El File-DB es demasiado corto.
0	1	8431_H	Error al escribir en el File-DB.
0	1	8432_H	Error al leer del File-DB.
0	1	8433_H	Error al acceder al File-DB.
0	1	8434_H	Acción cancelada.
0	1	8435_H	Se reseteará el canal.
0	1	8436_H	Respuesta inesperada del servidor
0	1	8437_H	No se ha podido verificar el certificado.
0	1	8438_H	Se ha producido un error desconocido
0	1	8439_H	El comando FTP causa un error. La causa debe buscarse en el servidor FTP (comando REST).
0	1	8440_H	El servidor FTP no soporta el protocolo SSL exigido.
0	1	8446_H	Después de enviar la contraseña FTP al servidor FTP, éste devolvió un código inesperado.
0	1	8451_H	Al intentar cambiar el modo de transmisión de binario a ASCII, se notificó un error.
0	1	8455_H	Error en una petición de memorización en el CM/CP.
0	1	8460_H	Ha surgido un problema en la gestión de SSL/TLS.
0	1	8469_H	Error de interfaz No se ha podido utilizar la interfaz de salida especificada. Solución: Indique qué interfaz debe utilizarse para las conexiones salientes.
0	1	8475_H	El certificapdo SSL o el SSH md5 fingerprint no se consideran de confianza.
0	1	8476_H	El servidor FTP no ha recibido nada. En el estado actual debe contarse con un comportamiento erróneo.
0	1	8477_H	No se encontró el "Crypto engine" (módulo criptográfico) indicado.
0	1	8478_H	Se produjo un error al configurar como estándar el "Crypto engine" SSL (módulo criptográfico) seleccionado.
0	1	8480_H	Ha surgido un problema con el certificado del cliente FTP.
0	1	8481_H	No se pudo utilizar la cifra indicada.
0	1	8482_H	El servidor FTP utiliza una codificación que no está soportada.
0	1	8484_H	Se ha excedido el tamaño máximo del archivo.
0	1	8485_H	El File-DB ha sido modificado durante el envío o su estructura no es correcta.
0	1	8489_H	No se han podido enviar los datos. No hay suficiente memoria disponible para la operación en el servidor FTP.

DONE	ERROR	STATUS	Significado
0	1	8492_H	El archivo ya existe. El archivo no se sobrescribe.
0	1	8496_H	Ha surgido un problema al leer el certificado CA SSL.
0	1	8497_H	Se ha producido un error inesperado en la sesión SSH.
0	1	8498_H	No se pudo deshacer la conexión SSL.
0	1	8499_H	El socket no está listo para enviar/recibir. Espere hasta que haya señal de disponibilidad y vuélvalo a intentar.
0	1	8501_H	Error en la comprobación del certificado SSL por el servidor FTP.
0	1	8507_H	Durante la sesión FTP activa se ha producido un timeout al establecer la conexión y esperar al servidor FTP.
0	1	$8F55_H$	Bit de estado del encabezado: Locked
0	1	$8F56_H$	El bit NEW en el encabezado del File-DB no se ha desactivado
0	1	$8F6B_H$	Posibles causas: • Valor incorrecto para el parámetro CMD Se permiten valores entre 0 y 15. • Un comando del FB 40 no es compatible. Posible causa: Firmware incorrecto del CP Solución: Actualización del firmware (en el caso de CPs menos recientes, utilizar en lugar del FB 40 las funciones FC 40...FC 44.)
0	1	$8F7F_H$	Error interno p. ej., referencia ANY no permitida.

Consulte también

Panorámica de FTP_CMD (Página 5348)

Estructura y uso del File-DB - Modo cliente FTP

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Client

Funcionamiento

Para la transmisión de datos mediante FTP hay que crear bloques de datos (File-DBs) en la CPU de la estación S7. Dichos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como archivos transmisibles. Constan de los siguientes sectores:

- Sector 1: encabezado del File-DB (tiene una estructura fija con una longitud de 20 byte)
- Sector 2: datos de usuario (longitud y estructura variables)

Requisitos en la configuración de la CPU

Utilice los ajustes siguientes para permitir el acceso FTP:

- En los datos de configuración de la CPU, en "Propiedades > General > Protección": desactive la opción "Desactivar comunicación PUT/GET".
- En todos los bloques de datos utilizados como File-DB desactive el atributo "Acceso optimizado al bloque"

Encabezado del File-DB para modo cliente FTP

Observación: El encabezado del File-DB aquí descrito es prácticamente idéntico al encabezado del File-DB descrito para el modo de servidor. Las diferencias residen en los parámetros:

- WRITE_ACCESS
- FTP_REPLY_CODE

Parámetro	Tipo	Valor / Significado	Alimentación
EXIST	BOOL	El bit EXIST indica si hay datos válidos en el área de datos de usuario. El comando FTP retrieve solo procesa la petición si EXIST=1. • 0: el File-DB no contiene datos de usuario válidos ("el archivo no existe"). • 1: el File-DB contiene datos de usuario válidos ("el archivo existe").	El comando FTP dele pone EXIST=0; El comando FTP store pone EXIST=1;
LOCKED	BOOL	El bit LOCKED sirve para la protección de acceso al File-DB. • 0: se puede acceder al File-DB. • 1: el File-DB está bloqueado.	Los comandos FTP store y retr ponen LOCKED=1 durante la edición. Para una operación de escritura desde el programa de usuario rige: El programa de usuario almacenado en la CPU S7 puede activar o desactivar LOCKED para asegurar la coherencia durante un acceso de escritura. Procedimiento recomendado para el programa de usuario: 1. Comprobar el bit LOCKED; si =0 2. Poner el bit WRITEACCESS=0 3. Comprobar el bit LOCKED; si =0 4. Poner el bit LOCKED=1 5. Escribir datos 6. Poner el bit LOCKED=0

Parámetro	Tipo	Valor / Significado	Alimentación
NEW	BOOL	El bit NEW informa sobre si los datos se han modificado desde la última operación de lectura. • 0: el contenido del File-DB no ha cambiado desde la última operación de escritura. El programa de usuario de la CPU S7 ha registrado el último cambio. • 1: el programa de usuario de la CPU S7 no ha registrado la última operación de escritura.	El comando FTP store pone NEW=1 tras la edición El programa de usuario de la CPU S7 debe activar NEW=0 tras la lectura de los datos para permitir un nuevo comando retr.
WRITE_ACCESS	BOOL	0: el programa de usuario (bloques de cliente FTP) tiene derecho de escritura para los File-DBs de la CPU S7. 1: el programa de usuario (bloques de cliente FTP) no tiene derecho de escritura para los File-DBs de la CPU S7.	El bit se pone a un valor de inicialización al configurar el DB. Recomendación: A ser posible, el bit debería dejarse inalterado. En caso especiales es posible una adaptación durante el funcionamiento.
ACT_LENGTH	DINT	Longitud actual del área de datos de usuario. El contenido de este campo solo es válido si EXIST = 1.	La longitud actual se actualiza después de una operación de escritura.
MAX_LENGTH	DINT	Longitud máxima del área de datos de usuario (longitud total del DB menos 20 bytes del encabezado).	La longitud máxima se debería definir al configurar el DB. El valor puede ser modificado por el programa de usuario también durante el funcionamiento.
FTP_REPLY_CODE	INT	Cifra sin signo (16 bit) que contiene el último código Reply de FTP como valor binario. El contenido de este campo solo es válido si EXIST = 1.	Es actualizado por el cliente FTP al editar comandos FTP.
DATE_TIME	DATE_AND_TIME	Fecha y hora de la última modificación del archivo. El contenido de este campo solo es válido si EXIST = 1.	La fecha actual se actualiza después de una operación de escritura. Si se utiliza la función "Transmisión de la hora", la entrada se corresponde con la hora transmitida. Si no se utiliza la función "Transmisión de la hora", se introduce una hora relativa. La referencia es el momento de arranque del CP (el valor de inicialización es 1.1.1994 0.0 horas).

Consulte también

Bloque de datos FILE_DB_HEADER como plantilla - modo cliente FTP (Página 5364)

Bloque de datos FILE_DB_HEADER como plantilla - modo cliente FTP

Significado

Para la creación de un encabezado del File-DB está predeterminado el tipo de datos FILE_DB_HEADER.

Funcionamiento

Para la transmisión de datos mediante FTP hay que crear bloques de datos (File-DBs) en la CPU de la estación S7. Dichos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como archivos transmisibles. Constan de los siguientes sectores:

- Sector 1: encabezado del File-DB (tiene una estructura fija con una longitud de 20 byte)
- Sector 2: datos de usuario (longitud y estructura variables)

Proceda del siguiente modo:

1. Cree un bloque de datos del tipo "DB global" en la CPU de PLC en la que se crea el programa de usuario con las instrucciones FTP.
2. Seleccione la fila que va a utilizar como línea de inicio para el File-DB.
3. En la lista desplegable de la columna "Tipo de datos", seleccione un elemento de estructura del tipo FILE_DB_HEADER.
   Resultado: Se crea una estructura de datos con la estructura de encabezado necesaria para el File-DB.
4. Seleccione las propiedades del bloque de datos recién creado y desactive el atributo "Acceso optimizado al bloque".

Nota

Función "Agregar nuevo bloque" - Selección de tipo

Al crear nuevos bloques de datos, en la entada "Tipo" de la lista desplegable también se ofrece el tipo de bloque "FILE_DB_HEADER". ¡No utilice esta opción! El bloque de datos creado de esa manera solo obtiene la estructura de encabezado y no puede ampliarse con el área necesaria para depositar datos de usuario.

Bloque de datos FILE_DB_HEADER - Ejemplo y plantilla para el encabezado del File-DB

En la vista de declaración se reconoce la siguiente estructura:

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	bit08	BOOL	FALSE	reservado
+0.1	bit09	BOOL	FALSE	reservado

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.2	bit10	BOOL	FALSE	reservado
+0.3	bit11	BOOL	FALSE	reservado
+0.4	bit12	BOOL	FALSE	reservado
+0.5	bit13	BOOL	FALSE	reservado
+0.6	bit14	BOOL	FALSE	reservado
+0.7	bit15	BOOL	FALSE	reservado
+1.0	EXIST	BOOL	FALSE	si TRUE: el File-DB contiene datos válidos
+1.1	LOCKED	BOOL	FALSE	si TRUE: el File-DB está bloqueado debido a un cambio de contenido.
+1.2	NEW	BOOL	FALSE	si TRUE: el contenido del File-DB es nuevo y no puede sobrescribirse.
+1.3	WRITE_ACCESS	BOOL	FALSE	si TRUE: el servidor FTP tiene acceso de escritura.
+1.4	bit04	BOOL	FALSE	reservado
+1.5	bit05	BOOL	FALSE	reservado
+1.6	bit06	BOOL	FALSE	reservado
+1.7	bit07	BOOL	FALSE	reservado
+2.0	ACT_LENGTH	DINT	L#0	Longitud actual del contenido en bytes (sin incluir los 20 bytes del encabezado)
+6.0	MAX_LENGTH	DINT	L#0	Longitud máxima del contenido en bytes (sin incluir los 20 bytes del encabezado)
+10.0	FTP_REPLY_CODE	INT	0	Última información de respuesta del servidor FTP remoto.
+12.0	DATE_TIME	DATE_AND_TIME	DT#00-1-1-0:0:0.000	Fecha y hora del último cambio en el contenido del File-DB.
=20.0		END_STRUCT		

Diferencias en los modos de funcionamiento

Encabezado del File-DB para modo cliente FTP

El encabezado del File-DB aquí descrito es prácticamente idéntico para los modos de cliente y servidor FTP. Las diferencias residen en los parámetros

- WRITE_ACCESS
- FTP_REPLY_CODE

CM/CP S7-1200

Telecontrol

Instrucciones de Telecontrol

TC_CON: establecer un enlace a través de la red GSM

Significado

La instrucción TC_CON permite al S7-1200 establecer conexiones de los tipos siguientes a través del CP 1242-7:

- ISO-ON-TCP
  El interlocutor es un CP 1242-7.
  Las conexiones ISO-ON-TCP solo se utilizan en el modo de operación "GPRS directo".
- UDP
  El interlocutor es uno cualquiera.
- SMS
  El interlocutor es un cliente SMS.
- Conexión Telecontrol
  El interlocutor es un servidor Telecontrol o bien otra estación a la que se accede a través del servidor Telecontrol.

Un TC_CON establece una conexión. Dependiendo del modo de operación del CP 1242-7 y del protocolo utilizado, por cada CP se soportan un máximo de 3 a 5 conexiones paralelas con IDs únicas (véase más abajo). El número máximo posible de conexiones paralelas puede consultarse en los datos de rendimiento del CP.

Para describir la conexión, el parámetro CONNECT utiliza un bloque de datos (DB) con una estructura definida por el tipo de datos del sistema (SDT).

El tipo de conexión deseado se define mediante un SDT "TCON_..." específico de la conexión (véase más abajo). Para cada uno de los tipos de conexión mencionados anteriormente debe parametrizarse uno de los siguientes SDTs:

- TCON_IP_RFC para conexiones ISO-ON-TCP
- TCON_IP_V4 para conexiones UDP
- TCON_PHONE para conexiones SMS
- TCON_WDC para conexiones Telecontrol

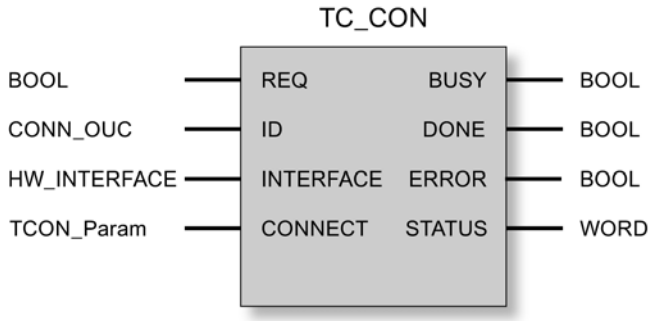
En el parámetro "ActiveEstablished" de estos SDTs se define, entre otros, si se trata de un establecimiento de conexión activo o pasivo.

Para parametrizar estos SDTs consulte TCON_...: SDTs para el establecimiento de un enlace Telecontrol (Página 5382).

El parámetro ID referencia la conexión GPRS. La ID se asigna y debe ser unívoca dentro de la CPU.

El parámetro INTERFACE referencia la interfaz GPRS del CP local deseado. Esta debe ser adoptada desde STEP 7.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales

La tabla siguiente explica los parámetros formales de la instrucción TC_CON.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
REQ	INPUT	BOOL	0, 1	Con un flanco ascendente se inicia el procesamiento de la instrucción y se inicializan los indicadores de estado. Actualización de los indicadores de estado DONE, ERROR y STATUS cuando no hay ningún flanco ascendente.
ID	INPUT	CONN_OUC (WORD)	$1...07FF_h$	Referencia a la conexión correspondiente. La ID se asigna. El valor de ID también es requerido por el tipo de datos de sistema (SDT) correspondiente al parámetro CONNECT.
INTERFACE	INPUT	HW_INTERFACE		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
CONNECT	INOUT	TCON_Param	Véase también "TCON_...: SDTs para establecer la conexión de Telecontrol"	Referencia a un bloque de datos para el establecimiento de la conexión. Los SDT del tipo TCON_IP_RFC, TCON_IP_V4, TCON_PHONE o TCON_WDC predeterminan la estructura del bloque de datos apto para la conexión correspondiente. Tenga en cuenta el parámetro "ActiveEstablished" (establecimiento activo / pasivo de la conexión) en los SDTs.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ENO	OUTPUT	BOOL	0: error 1: correcto	Salida de habilitación Si la instrucción presenta un error en el tiempo de ejecución, se pondrá ENO = 0.
BUSY	OUTPUT	BOOL	0: procesamiento de la instrucción todavía no iniciado, finalizado o cancelado 1: procesamiento de la instrucción en curso	Indicador del estado de procesamiento de la instrucción
DONE	OUTPUT	BOOL	0: - 1: procesamiento de la instrucción finalizado correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente. Consulte el significado en relación con los parámetros ERROR y STATUS en Indicadores de la instrucción.
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Indicadores de la instrucción.
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Indicadores de la instrucción.

Indicadores BUSY, DONE y ERROR

Los indicadores DONE y ERROR solo son relevantes si BUSY = 0.

BUSY	DONE	ERROR	Significado
0	0	0	Ninguna petición en proceso

Encontrará todas las demás combinaciones de los indicadores DONE y ERROR en la tabla siguiente.

En caso de llamada, la instrucción permanecerá en el estado BUSY = 1 por algunos segundos. En los siguientes casos puede durar más tiempo el estado BUSY = 1:

- En el caso de conexiones ISO-ON-TCP activas, cuando no se puede acceder al interlocutor.
- En el caso de conexiones pasivas, cuando no se recibe ningún telegrama.

Indicadores DONE, ERROR y STATUS

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

DONE	ERROR	STATUS	Significado
1	0	0000_H	Petición ejecutada correctamente
0	0	7000_H	No hay ningún procesamiento de petición activo (primera llamada de la instrucción)
0	0	7001_H	Procesamiento de petición iniciado (primera llamada de la instrucción)

DONE	ERROR	STATUS	Significado
0	0	7002_H	Procesamiento de petición en curso (nueva llamada de la instrucción si BUSY = 1)
0	1	8086_H	Valor no permitido para ID
0	1	8087_H	Número máximo de conexiones alcanzado, no es posible ninguna otra conexión
0	1	$80E3_H$	La ID ya se está utilizando en otra conexión.
0	1	$80E6_H$	Ninguna solicitud en proceso (la activación de la instrucción no se inicia)
0	1	$80E8_H$	Interlocutor remoto no accesible. Compruebe los parámetros de conexión. En el modo de operación "GPRS directo", el aviso se emite cuando el interlocutor está accesible pero no acepta las peticiones de conexión.
0	1	$80EB_H$	Solicitud rechazada por el momento (TC_CON ya se ha llamado con la misma dirección de destino.)
0	1	$80EC_H$	No se ha podido abrir el Listener Port: Compruebe los parámetros de conexión.
0	1	$80F2_H$	El CP está en el modo de operación equivocado: • Las conexiones Telecontrol solo se permiten en el modo de operación "Telecontrol". • Las conexiones ISO-ON-TCP solo se utilizan en el modo de operación "GPRS directo".
0	1	$80F3_H$	No hay ningún punto final libre en la conexión para enviar datos: • Utilice menos conexiones o • Utilice menos conexiones pasivas o • Desactive la opción NTP. Tenga en cuenta el número máximo de conexiones paralelas del CP 1242-7.
0	1	$80F4_H$	No se puede generar el punto final de la conexión: Repita la llamada. En dado caso, compruebe los parámetros de conexión.
0	1	$80F6_H$	Error de formato de un parámetro en el bloque de datos llamado (tamaño erróneo, formato incorrecto o valor no válido) Compruebe la configuración del SDT "TC_CON...".

TC_DISCON: desconectar un enlace a través de la red GSM

Significado

En un S7-1200 con CP 1242-7, la instrucción TC_DISCON provoca la desconexión de una conexión ISO-ON-TCP, UDP, SMS o de Telecontrol establecida con la instrucción TC_CON.

Encontrará información detallada sobre los tipos de conexión en la descripción de la instrucción TC_CON.

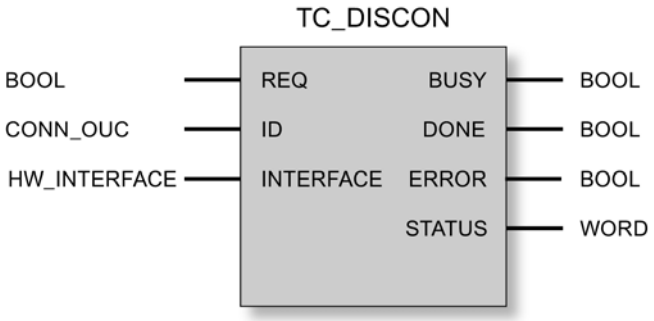
TC_DISCON desconecta solo de forma lógica la conexión con el servidor de Telecontrol. Si la conexión con el servidor de Telecontrol va a desconectarse físicamente, debe configurarse como "Conexión temporal" en STEP 7 en el grupo de parámetros "Servidor de Telecontrol".

En el nivel TCP/IP, la conexión permanece. Las estaciones temporales deshacen la conexión automáticamente después de enviar los datos.

El parámetro ID referencia la conexión GPRS. La ID debe ser unívoca dentro de la CPU e idéntica a la ID utilizada en TC_CON.

El parámetro INTERFACE referencia la interfaz GPRS del CP local deseado. El valor debe ser idéntico al utilizado para INTERFACE en TC_CON.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales

La tabla siguiente muestra los parámetros formales de la instrucción TC_DISCON.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
REQ	INPUT	BOOL	0, 1	Con un flanco ascendente se inicia el procesamiento de la instrucción y se inicializan los indicadores de estado. Actualización de los indicadores de estado DONE, ERROR y STATUS cuando no hay ningún flanco ascendente.
ID	INPUT	CONN_OUC (WORD)	1...07FF$_h$	Referencia a la conexión correspondiente
INTERFACE	INPUT	HW_INTERFACE		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
ENO	OUTPUT	BOOL	0: error 1: correcto	Salida de habilitación Si la instrucción presenta un error en el tiempo de ejecución, se pondrá ENO = 0.
BUSY	OUTPUT	BOOL	0: procesamiento de la instrucción todavía no iniciado, finalizado o cancelado 1: procesamiento de la instrucción en curso	Indicador del estado de procesamiento de la instrucción
DONE	OUTPUT	BOOL	0: - 1: procesamiento de la instrucción finalizado correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente. Consulte el significado en relación con los parámetros ERROR y STATUS en Indicadores de la instrucción.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Indicadores de la instrucción.
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Indicadores de la instrucción.

Indicadores BUSY, DONE y ERROR

Los indicadores DONE y ERROR solo son relevantes si BUSY = 0.

BUSY	DONE	ERROR	Significado
0	0	0	La instrucción todavía no se ha llamado.

Encontrará todas las demás combinaciones de los indicadores DONE y ERROR en la tabla siguiente.

Nota

En caso de llamada, la instrucción permanecerá en el estado BUSY = 1 por algunos segundos.

Indicadores DONE, ERROR y STATUS

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

DONE	ERROR	STATUS	Significado
1	0	0000_H	Petición ejecutada correctamente
0	0	7000_H	No hay ningún procesamiento de petición activo (primera llamada de la instrucción)
0	0	7001_H	Procesamiento de petición iniciado (primera llamada de la instrucción)
0	0	7002_H	Procesamiento de petición en curso (nueva llamada de la instrucción si BUSY = 1)
0	1	8086_H	Valor no permitido para ID
0	1	$80E4_H$	ID desconocida: No existe ninguna conexión e a través de TC_CON con esta ID.
0	1	$80E6_H$	Ninguna solicitud en proceso (la activación de la instrucción no se inicia)
0	1	$80F5_H$	Punto final de la conexión no válido: • Fallo en el establecimiento de la conexión a través de TC_CON o • Desconexión por parte del interlocutor remoto.
0	1	$80F6_H$	Error de formato de un parámetro en el bloque de datos llamado (tamaño erróneo, formato incorrecto o valor no válido) Compruebe la configuración del SDT "TC_CON...".

TC_SEND: enviar datos a través de la red GSM

Significado

La instrucción TC_SEND permite enviar datos a través de las conexiones programadas de los tipos siguientes:

- Conexiones ISO-ON-TCP
- Conexiones UDP
- Conexiones SMS
  El envío de SMS solo es soportado si este se estableció en la configuración STEP 7 del CP.
- Conexiones Telecontrol

Nota

Enviar SMS a varios destinatarios

Si desea enviar un SMS idéntico a varios destinatarios, deberá establecer una conexión para cada destinatario.

Encontrará más información detallada sobre los tipos de conexión en la descripción de la instrucción TC_CON.

El parámetro ID referencia la conexión GPRS. El valor de la ID debe ser equivalente al valor de ID utilizado en TC_CON.

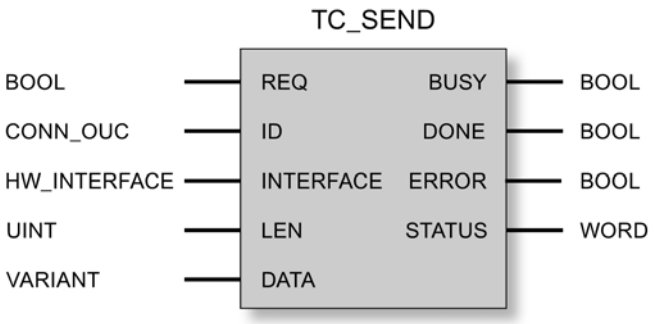
El parámetro INTERFACE referencia la interfaz GPRS del CP local deseado. El valor debe ser idéntico al utilizado para INTERFACE en TC_CON.

El número de datos que deben enviarse se indica en el parámetro LEN.

El tamaño del rango de datos indicado en DATA debe ser como mínimo tan grande como el número de bytes configurado en LEN. Los tipos de datos permitidos en el rango de datos indicado en DATA son todos excepto BOOL y ARRAY of BOOL.

La dirección de destino (interlocutor) para los datos que deben enviarse se configura en la instrucción TC_CON.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales

La tabla siguiente muestra los parámetros formales de la instrucción TC_SEND.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
REQ	INPUT	BOOL	0, 1	Con un flanco ascendente se inicia el procesamiento de la instrucción y se inicializan los indicadores de estado. Actualización de los indicadores de estado DONE, ERROR y STATUS cuando no hay ningún flanco ascendente.
ID	INPUT	CONN_OUC (WORD)	$1...07FF_h$	Referencia a la conexión correspondiente
INTERFACE	INPUT	HW_INTERFACE		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
LEN	INPUT	UINT	1...2048	Número de bytes de los datos que deben enviarse, máximo 2048. El valor debería corresponderse con el tamaño del rango de DATA.
DATA	INOUT	VARIANT		Referencia de dirección al rango de datos de la CPU que debe transferirse*
ENO	OUTPUT	BOOL	0: Error 1: correcto	Salida de habilitación Si la instrucción presenta un error en el tiempo de ejecución, se pondrá ENO = 0.
BUSY	OUTPUT	BOOL	0: procesamiento de la instrucción todavía no iniciado, finalizado o cancelado 1: procesamiento de la instrucción en curso	Indicador del estado de procesamiento de la instrucción
DONE	OUTPUT	BOOL	0: - 1: procesamiento de la instrucción finalizado correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente. ** Consulte el significado en relación con los parámetros ERROR y STATUS en Indicadores de la instrucción.
ERROR	OUTPUT	BOOL	0: - 1: Error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Indicadores de la instrucción.
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Indicadores de la instrucción.

* Consulte el apartado siguiente para las particularidades del parámetro DATA para textos SMS.

** Tras enviar un telegrama, TC_SEND pone DONE = 1. Observe el comportamiento siguiente:
el fallo de una conexión ISO-on-TCP no es detectada por el emisor hasta pasados 1 o 2 minutos. Los datos transferidos pueden haberse perdido aunque TC_SEND haya puesto DONE = 1 en el emisor.
Si se cancela una conexión ISO-on-TCP tras recibir un telegrama y antes de iniciar TC_RECV, es posible que los datos transferidos se pierdan incluso si TC_SEND ha puesto DONE = 1 en el emisor.

Configuración de textos SMS con el parámetro DATA

La instrucción envía como texto SMS aquellos datos a los que hace referencia el indicador del tipo VARIANT del parámetro DATA.

Si desde DATA se hace referencia a un operando del tipo de datos STRING para textos SMS, entonces también se transfieren los dos primeros bytes con indicaciones de longitud del string.

Una posibilidad para la representación correcta de textos SMS para enviar es la conversión del string de texto en un Array of BYTE o Array of CHAR con ayuda de la función de conversión Strg_TO_Chars. Strg_TO_Chars se vincula al parámetro EN con el parámetro de salida ENO de TC_SEND.

Para textos SMS, el CP no soporta todos los caracteres especiales, como por ejemplo la diéresis. Se aplica la especificación GSM 03.38. Pueden aparecer limitaciones adicionales debido al proveedor de red GSM.

Indicadores BUSY, DONE y ERROR

Los indicadores DONE y ERROR solo son relevantes si BUSY = 0.

BUSY	DONE	ERROR	Significado
0	0	0	Ninguna petición en proceso

Encontrará todas las demás combinaciones de los indicadores DONE y ERROR en la tabla siguiente.

Indicadores DONE, ERROR y STATUS

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

DONE	ERROR	STATUS *	Significado
1	0	0000_H	Petición ejecutada correctamente
0	0	7000_H	No hay ningún procesamiento de petición activo (primera llamada de la instrucción)
0	0	7001_H	Procesamiento de petición iniciado (primera llamada de la instrucción)
0	0	7002_H	Procesamiento de petición en curso (nueva llamada de la instrucción si BUSY = 1)
0	1	8086_H	Valor no permitido para ID
0	1	$80E0_H$	• La indicación de longitud en LEN es mayor que el rango de datos para transferir en DATA. o bien • Error interno Si se envían telegramas directamente al servidor de Telecontrol (modo de operación "Telecontrol"), asegúrese de que el tiempo de ciclo de emisión ≥ 1 segundo.
0	1	$80E1_H$	Rebase de tiempo: • Aumente el valor del "tiempo de vigilancia de conexión" en la configuración del CP 1242-7 o • Compruebe el interlocutor.
0	1	$80E4_H$	ID desconocida: Llame en primer lugar la instrucción TC_CON.

DONE	ERROR	STATUS *	Significado
0	1	$80E6_H$	Ninguna solicitud en proceso (la activación de la instrucción no se inicia)
0	1	$80E7_H$	Los datos que debían enviarse no se han transferido íntegramente: Repita la petición.
0	1	$80E8_H$	Interlocutor remoto no accesible. Compruebe los parámetros de conexión. En el modo de operación "GPRS directo", el aviso se emite cuando el interlocutor está accesible pero no acepta las peticiones de conexión.
0	1	$80E9_H$	Desconexión por parte del interlocutor remoto: Compruebe el interlocutor. En dado caso, deshaga la conexión con TC_DISCON y establézcala nuevamente con TC_CON.
0	1	$80EA_H$	Mensaje de error del interlocutor remoto: • Compruebe el interlocutor. Active la instrucción "TC_RECV" en el interlocutor. • En dado caso, deshaga la conexión con TC_DISCON y establézcala nuevamente con TC_CON.
0	1	$80EF_H$	No se ha podido enviar el SMS: • Compruebe si existe la dirección de destino (número de llamada del participante destinatario). • Compruebe si la tarjeta SIM insertada soporta el envío de SMS. • Asegúrese de que la opción "Estándar" se haya seleccionado para el acceso al bloque en el momento de crear el bloque de datos TCON_PHONE.
0	1	$80F1_H$	En envío de SMS no está activado en la configuración STEP 7 del CP: Active la opción "Permitir SMS" en la configuración del CP.
0	1	$80F4_H$	No se puede generar el punto final de la conexión: Compruebe el interlocutor.
0	1	$80F5_H$	Punto final de la conexión no válido: • Fallo en el establecimiento de la conexión a través de TC_CON. o bien • Desconexión por parte del interlocutor remoto: Llame la instrucción TC_DISCON.
0	1	$80F6_H$	Error de formato de un parámetro en el bloque de datos llamado (tamaño erróneo, formato incorrecto o valor no válido): Compruebe la configuración del SDT "TC_CON...".

* Encontrará otros estados que no aparecen aquí en los indicadores de estado de las instrucciones "RDREC" y "WRREC", bajo los dos bytes de estado centrales (STATUS[2], STATUS[3]).

TC_RECV: recibir datos a través de la red GSM

Significado

La instrucción TC_RECV permite recibir datos a través de las conexiones programadas de los tipos siguientes:

- conexiones ISO-ON-TCP
- conexiones SMS
  Para recibir SMS debe estar configurado el número de llamada del remitente en la configuración STEP 7 del CP que debe recibirse (números de llamada autorizados). El remitente debe soportar la función CLIP.
  El número de llamada del interlocutor debe introducirse en el SDT "TCON_PHONE".
  Los SMS de alarma se excluyen mediante un filtro.
- Conexiones Telecontrol

Nota

Recibir SMS de varios remitentes

Si desea recibir SMS de varios remitentes, puede escoger para ello entre dos alternativas:

- Debe configurar varias conexiones (TC_CON, TC_RECV, TC_DISCON).
  o bien
- En una única conexión configurada en el bloque de datos necesario "TCON_PHONE" en el parámetro "PhoneNumber" no puede introducir ningún número de llamada. Esto se interpretará durante la recepción de mensajes como comodín para todos los interlocutores autorizados.

Encontrará más información detallada sobre los tipos de conexión en la descripción de la instrucción TC_CON.

El parámetro ID referencia la conexión GPRS. El valor de la ID debe ser equivalente al valor de ID utilizado en TC_CON.

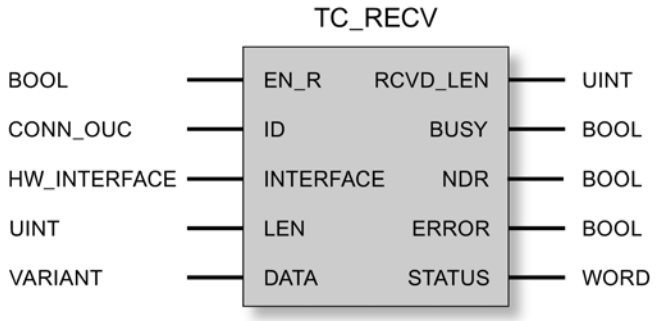
El parámetro INTERFACE referencia la interfaz GPRS del CP local deseado. El valor debe ser idéntico al utilizado para INTERFACE en TC_CON.

El número máximo de los datos de recepción se indica en el parámetro LEN.

El tamaño del rango de datos indicado en DATA debe ser como mínimo tan grande como el número de bytes configurado en LEN. Los tipos de datos permitidos en el rango de datos indicado en DATA son todos excepto BOOL y ARRAY of BOOL. Los datos recibidos se interpretan como si el interlocutor remitente hubiera utilizado los mismos tipos de datos.

El DB (tipo de datos de sistema) utilizado para la descripción de la conexión de TC_RECV se debe distinguir de un DB utilizado para TC_SEND.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales

La tabla siguiente muestra los parámetros formales de la instrucción TC_RECV.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
EN_R	INPUT	BOOL	0: recepción de datos bloqueada 1: recepción de datos permitida	Permitir/bloquear la recepción de datos. Tras ajustar 1 a 0, el bloque de programa recibe datos una vez más.
ID	INPUT	CONN_OUC (WORD)	1...07FF$_h$	Referencia a la conexión correspondiente
INTERFACE	INPUT	HW_INTERFACE		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
LEN	INPUT	UINT	1...2048	Número (mínimo) de bytes de los datos que deben recibirse, máximo 2048
DATA	INOUT	VARIANT		Referencia de dirección al rango de datos de recepción de la CPU*
ENO	OUTPUT	BOOL	0: error 1: correcto	Salida de habilitación Si la instrucción presenta un error en el tiempo de ejecución, se pondrá ENO = 0.
RCVD_LEN	OUTPUT	UINT		Número de bytes de los datos recibidos
BUSY	OUTPUT	BOOL	0: procesamiento de la instrucción todavía no iniciado, finalizado o cancelado 1: procesamiento de la instrucción en curso	Indicador del estado de procesamiento de la instrucción
DONE	OUTPUT	BOOL	0: - 1: procesamiento de la instrucción finalizado correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente. Consulte el significado en relación con los parámetros ERROR y STATUS en Indicadores de la instrucción.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Indicadores de la instrucción.
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Indicadores de la instrucción.

* Consulte el apartado siguiente para las particularidades del parámetro DATA para textos SMS.

Configuración de textos SMS con el parámetro DATA

La instrucción hace referencia al texto SMS recibido con el indicador del tipo VARIANT del parámetro DATA en el rango de datos de la CPU.

Si desde DATA se hace referencia a un operando del tipo de datos STRING para el texto SMS, entonces se interpretan los dos primeros bytes del texto SMS como indicaciones de longitud del tipo de datos STRING y no como texto SMS.

Una posibilidad para la representación correcta de textos SMS para recibir es la conversión de un Array of BYTE o Array of CHAR en un string de texto con ayuda de la función de conversión Chars_TO_Strg. Chars_TO_Strg se vincula al parámetro EN con el parámetro de salida ENO de TC_RECV.

Para textos SMS, el CP no soporta todos los caracteres especiales, como por ejemplo la diéresis. Se aplica la especificación GSM 03.38. Pueden aparecer limitaciones adicionales debido al proveedor de red GSM.

Indicadores BUSY, DONE y ERROR

Los indicadores DONE y ERROR solo son relevantes si BUSY = 0.

BUSY	DONE	ERROR	Significado
0	0	0	Ninguna petición en proceso

Encontrará todas las demás combinaciones de los indicadores DONE y ERROR en la tabla siguiente.

Indicadores DONE, ERROR y STATUS

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

DONE	ERROR	STATUS *	Significado
1	0	0000_H	Petición ejecutada correctamente
0	0	7000_H	No hay ningún procesamiento de petición activo (primera llamada de la instrucción)
0	0	7001_H	Procesamiento de petición iniciado (primera llamada de la instrucción)
0	0	7002_H	Procesamiento de petición en curso (nueva llamada de la instrucción si BUSY = 1)

DONE	ERROR	STATUS *	Significado
0	1	$80A3_H$	• Se está intentando volver a establecer una conexión existente. • Se está intentando finalizar una conexión no existente.
0	1	$80E0_H$	• El tamaño de los datos que se reciben para el rango indicado en DATA es mayor que la indicación de longitud en LEN. o bien • Error interno
0	1	8086_H	Valor no permitido para ID
0	1	$80E4_H$	ID desconocida: llame en primer lugar la instrucción TC_CON.
0	1	$80E6_H$	Ninguna solicitud en proceso (la activación de la instrucción no se inicia)
0	1	$80F5_H$	Punto final de la conexión no válido: • Fallo en el establecimiento de la conexión a través de TC_CON. o bien • Desconexión por parte del interlocutor remoto: llame la instrucción TC_DISCON.
0	1	$80F6_H$	Error de formato de un parámetro en el bloque de datos llamado (tamaño erróneo, formato incorrecto o valor no válido) Compruebe la configuración del SDT "TC_CON...".

* Encontrará otros estados que no aparecen aquí en los indicadores de estado de las instrucciones "RDREC" y "WRREC", bajo los dos bytes de estado centrales (STATUS[2], STATUS[3]).

TC_CONFIG: transferir datos de configuración al CP

Significado

La instrucción TC_CONFIG permite modificar los parámetros de un CP 1242-7 configurados en STEP 7. Los valores configurados no se sobrescriben de forma remanente. Los valores sobrescritos siguen siendo válidos hasta una nueva llamada de TC_CONFIG o hasta el siguiente arranque de la estación (arranque en frío mediante la tensión OFF → ON).

Si los datos de configuración STEP 7 del CP deben modificarse permanentemente, entonces la instrucción debe llamarse de nuevo después de cada arranque de la estación (arranque en frío) o se debe cargar un proyecto modificado en la estación.

El parámetro CONFIG hace referencia al área de memoria con los datos de configuración. Los datos de configuración se guardan en un bloque de datos (DB). La estructura del DB está predeterminada por el tipo de datos de sistema (SDT) IF_CONF.

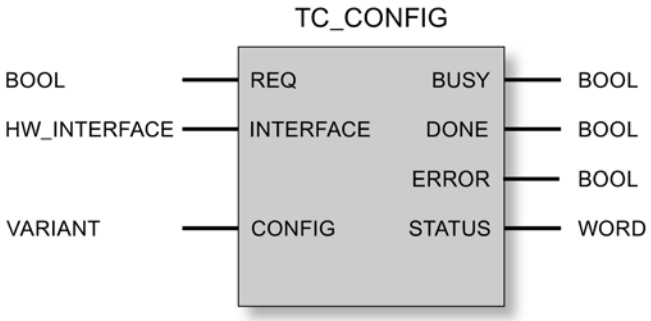
Los datos de configuración que deben modificarse en el CP se compilan en el IF_CONF como bloques "IF_CONF_..." para los diferentes parámetros, según sea necesario.

Los parámetros que no deben modificarse con la instrucción no se introducen en el IF_CONF. Estos conservan el valor configurado en STEP 7.

Encontrará información detallada sobre la parametrización de IF_CONF en el apartado IF_CONF: SDT para datos de configuración Telecontrol (Página 5386).

El parámetro INTERFACE referencia la interfaz GPRS del CP local deseado.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales

La tabla siguiente muestra los parámetros formales de la instrucción TC_CONFIG.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
REQ	INPUT	BOOL	0, 1	Con un flanco ascendente se inicia el procesamiento de la instrucción y se inicializan los indicadores de estado. Actualización de los indicadores de estado DONE, ERROR y STATUS cuando no hay ningún flanco ascendente.
INTERFACE	INPUT	HW_INTERFACE (WORD)		Referencia a la interfaz del CP 1242-7 local
CONFIG	INOUT	VARIANT	Véase también "IF_CONF: SDT para datos de configuración de Telecontrol"	Referencia al área de memoria con la compilación de los datos de configuración que deben modificarse
ENO	OUTPUT	BOOL	0: error 1: correcto	Salida de habilitación Si la instrucción presenta un error en el tiempo de ejecución, se pondrá ENO = 0.
BUSY	OUTPUT	BOOL	0: procesamiento de la instrucción todavía no iniciado, finalizado o cancelado 1: procesamiento de la instrucción en curso	Indicador del estado de procesamiento de la instrucción
DONE	OUTPUT	BOOL	0: - 1: procesamiento de la instrucción finalizado correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente. Consulte el significado en relación con los parámetros ERROR y STATUS en Indicadores de la instrucción.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Indicadores de la instrucción.
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Indicadores de la instrucción.

Indicadores BUSY, DONE y ERROR

Los indicadores DONE y ERROR solo son relevantes si BUSY = 0.

BUSY	DONE	ERROR	Significado
0	0	0	Ninguna petición en proceso

Encontrará todas las demás combinaciones de los indicadores DONE y ERROR en la tabla siguiente.

Indicadores DONE, ERROR y STATUS

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

DONE	ERROR	STATUS	Significado
1	0	0000_H	Petición ejecutada correctamente
0	0	7000_H	No hay ningún procesamiento de petición activo (primera llamada de la instrucción)
0	0	7001_H	Procesamiento de petición iniciado (primera llamada de la instrucción)
0	0	7002_H	Procesamiento de petición en curso (nueva llamada de la instrucción si BUSY = 1)
0	1	$80E6_H$	Ninguna solicitud en proceso (la activación de la instrucción no se inicia)
0	1	$80EB_H$	Solicitud rechazada temporalmente (el CP es configurado momentáneamente por STEP 7).
0	1	$80F6_H$	Error de formato de un parámetro en el bloque de datos llamado (tamaño erróneo, formato incorrecto o valor no válido) Compruebe el SDT "IF_CONF".
0	1	$80F7_H$	ID incorrecta en los bloques de parámetros de los datos de configuración: Compruebe el SDT "IF_CONF".

Otros mensajes de error

Otros mensajes de error

Los mensajes de error siguientes sólo sirven para fines de diagnóstico. Para obtener información complementaria, llame a la línea de atención de Siemens.

DONE	ERROR	STATUS	Significado
0	1	$80E0_H$	Error interno Tenga también en cuenta el posible significado en las instrucciones TC_SEND y TC_RECV.

TCON_...: SDTs para el establecimiento de un enlace Telecontrol

Tipos de datos del sistema TCON_... para la instrucción TC_CON

Para configurar el establecimiento de un enlace Telecontrol mediante la instrucción TC_CON se utiliza el parámetro CONNECT de la instrucción para describir el enlace.

La descripción del enlace está especificada por la estructura de un tipo de datos del sistema (SDT). La estructura del SDT correspondiente contiene los parámetros necesarios para establecer el enlace con un interlocutor remoto.

Para diferentes tipos de enlace que dependen del interlocutor remoto correspondiente se utilizan los siguientes SDTs:

- TCON_IP_RFC para enlaces ISO-on-TCP con estaciones IPv4 con CP 1242-7
- TCON_IP_V4 para enlaces UDP con estaciones IPv4 (solo enviar)
- TCON_PHONE para enlaces con clientes SMS
- TCON_WDC para enlaces con servidores Telecontrol o estaciones a las que se accede a través del servidor Telecontrol.

La parametrización de la descripción del enlace se realiza en un bloque de datos del mismo tipo que el SDT.

Creación de un DB del tipo TCON_...

Los tipos de datos de cada DB deben introducirse con el teclado. No se muestran en la lista de selección. No se tienen en cuenta las mayúsculas/minúsculas al introducir los tipos de datos.

Para crear un DB TCON_... proceda del siguiente modo:

1. Cree un bloque de datos del tipo "DB global" con acceso al bloque "estándar".
2. Cree un SDT en la tabla de configuración de parámetros del DB asignándole un nombre e introduciendo el tipo deseado en la celda del tipo de datos (por ejemplo, "TCON_IP_RFC").
   Se crea el SDT con sus parámetros (ver abajo).
3. Configure los parámetros que se describen a continuación para cada tipo SDT.

Los bits reservados no se muestran.

Tipo de datos de sistema TCON_IP_RFC para enlaces con estaciones IPv4

Este tipo de enlace solo es soportado en enlaces ISO-on-TCP con interlocutores con dirección IP fija. El CP debe estar configurado para el modo de operación "GPRS directo".

Tabla 9-105 Parámetros de TCON_IP_RFC

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
0 ... 1	InterfaceID	HW_ANY		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referencia al enlace GPRS local. La ID se asigna y debe ser unívoca dentro de la CPU. Aquí debe utilizarse el mismo valor que el del parámetro ID de la instrucción TC_CON.
4	ConnectionType	BYTE	W#16#0C	Variante de protocolo 12 (C_h): Enlace ISO-on-TCP
5	ActiveEstablished	BOOL		Identificación del tipo de establecimiento del enlace: • 0: establecimiento pasivo del enlace • 1: establecimiento activo del enlace
6 ... 7	-	-	-	- reservado -
8 ... 11	RemoteAddress	IP_V4		Dirección IP de los interlocutores
	ADDR	Array [1...4] of Byte		Dirección IP del interlocutor respectivo
12 ... 13	RemoteTSelector	TSelector		Selector T remoto
	TSelLen	UINT		Tamaño del selector T remoto "RemoteTSelector"
14 ... 45	TSel	Array [1...32] of Byte	Cualquiera	Selector de transporte remoto del enlace • Si "ActiveEstablished" = 1: El selector T del interlocutor local debe ser idéntico al selector T del interlocutor en el establecimiento de enlace activo (establecimiento de enlace pasivo en el interlocutor remoto). • Si "ActiveEstablished" = 0 correspondiente (establecimiento de conexión local pasivo, establecimiento de conexión remota activo)
46 ... 47	LocalTSelector	TSelector		Selector T local
	TSelLen	UINT		Longitud del selector T local "LOCAL_TSel"
48 ... 79	TSel	Array [1...32] of Byte	Cualquiera	Selector de transporte local del enlace • Si "ActiveEstablished" = 1: El selector T del interlocutor local debe ser idéntico al selector T del interlocutor en el establecimiento de enlace activo (establecimiento de enlace pasivo en el interlocutor remoto). • Si "ActiveEstablished" = 0 correspondiente (establecimiento de conexión local pasivo, establecimiento de conexión remota activo)

Tipo de datos de sistema TCON_IP_V4 para enlaces con estaciones IPv4

Este tipo de enlace solo es soportado para el envío a través de enlaces UDP a interlocutores con dirección IP fija.

Si se debe recibir, entonces ActiveEstablished = 0 debe activarse.

Tabla 9-106 Parámetros de TCON_IP_V4

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
0 ... 1	InterfaceID	HW_ANY		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referencia al enlace GPRS local. La ID se asigna y debe ser unívoca dentro de la CPU. Aquí debe utilizarse el mismo valor que el del parámetro ID de la instrucción TC_CON.
4	ConnectionType	BYTE	W#16#0B	Variante de protocolo 11 (B$_h$): Enlace UDP
5	ActiveEstablished	BOOL		Identificación del tipo de establecimiento del enlace: • 0: establecimiento pasivo del enlace Ajuste para el envío y la recepción de datos. • 1: establecimiento activo del enlace Ajuste solo para el envío de datos.
6 ... 7	-	-	-	- reservado -
8 ... 11	RemoteAddress	IP_V4		Dirección IP de los interlocutores
	ADDR	Array [1...4] of Byte		Dirección IP del interlocutor respectivo
12 ... 13	RemotePort	UINT	1...65535	Puerto IP del interlocutor Sin relevancia si ActiveEstablished = 0
14 ... 15	LocalPort	UINT	1...65535	Puerto IP local ("0" no está permitido.) Sin relevancia si ActiveEstablished = 1

Tipo de datos de sistema TCON_PHONE para enlaces SMS

Nota

Números de llamada autorizados

El requisito para que el CP acepte un SMS es la autorización del interlocutor emisor mediante su número de llamada. Estos números de llamada se configuran en STEP 7 con el CP en la lista "Números de llamada autorizados".

Texto SMS

- Se accede a los textos SMS programados para SMS que deben ser enviados a través del parámetro DATA de la instrucción TC_SEND.
- El texto de un SMS recibido se asigna al rango de direcciones de la CPU a través del parámetro DATA de la instrucción TC_RECV.

Tabla 9-107 Parámetros de TCON_PHONE

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
0 ... 1	InterfaceID	HW_ANY		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referencia al enlace GPRS local. La ID se asigna y debe ser unívoca dentro de la CPU. Aquí debe utilizarse el mismo valor que el del parámetro ID de la instrucción TC_CON.
4	ConnectionType	BYTE	W#16#0E	Variante de protocolo 14 (E$_h$): enlace SMS
5	ActiveEstablished	BOOL		Identificación del tipo de establecimiento del enlace (no relevante para el CP 1242-7): • 0: establecimiento pasivo del enlace (irrelevante en este caso) • 1: establecimiento activo del enlace
6...7	-	-	-	- reserviert -
8 ... 31	PhoneNumber	STRING[22]		Número de abonado del interlocutor Valores permitidos: signo más (+) y cifras Tenga en cuenta los caracteres concretos del prefijo de país de los números de llamada en cuestión asignado por el proveedor de red (signos "+" o ceros). Sin la entrada del parámetro PhoneNumber no se especificará ningún interlocutor y se podrán recibir SMS de todos los interlocutores autorizados. Al arrancar, tenga en cuenta lo siguiente: Si no se realiza ninguna entrada, el TC_RECV suministra primero el SMS recibido con más anterioridad.

Tipos de datos del sistema TCON_WDC para conexiones con servidores de Telecontrol o estaciones remotas

El enlace con el servidor Telecontrol que está asignado al S7-1200 o con una estación remota a la que se accede a través del servidor Telecontrol se puede configurar con TCON_WDC. Los datos de dirección del servidor Telecontrol asignado al CP se encuentran en STEP 7, en la ficha "Interfaz de Telecontrol > Modo de operación" del CP. El servidor Telecontrol o la estación remota se direcciona mediante el nombre de host o mediante la dirección IP.

En el parámetro "RemoteWdcAddress" de TCON_WDC se indica la ID de acceso del interlocutor.

Tabla 9-108 Parámetros de TCON_WDC

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
0 ... 1	InterfaceID	HW_ANY		Referencia a la interfaz del CP 1242-7 local (véase STEP 7 > Configuración CP > Interfaz Telecontrol > "Identificador HW")
2 ... 3	ID	CONN_OUC	1...07FF$_h$	Referencia al enlace GPRS local. La ID se asigna y debe ser unívoca dentro de la CPU. Aquí debe utilizarse el mismo valor que el del parámetro ID de la instrucción TC_CON.

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
4	ConnectionType	BYTE	W#16#0F	Variante de protocolo 15 (F_h): enlace Telecontrol mediante dirección IP
5	ActiveEstablished	BOOL		Identificación del tipo de establecimiento del enlace: • 0: establecimiento pasivo del enlace • 1: establecimiento activo del enlace
6 ... 7	-	-	-	- reservado -
8 ... 11	RemoteWdcAddress	DWORD		Indicación de la ID de acceso (hex). La ID de acceso depende del interlocutor. • Enlace con un CP remoto: La ID de acceso está compuesta por: – número de proyecto de STEP 7 – número de estación – slot Si la estación remota tiene varios CP GPRS y no va a definirse la ruta, deberá ponerse 0 en el último byte para el slot. La ID de acceso se encuentra en el proyecto de STEP 7, en el grupo de parámetros "Autenticación CP" del CP. • No hay conexión con el servidor de Telecontrol: ID de acceso = 0 • Solo para escribir en la imagen de proceso del CP: ID de acceso = DW#16#FEEDDADA

IF_CONF: SDT para datos de configuración Telecontrol

Establecimiento del tipo de datos del sistema IF_CONF para la instrucción TC_CONFIG

El parámetro CONFIG de la instrucción TC_CONFIG referencia el área de memoria con los datos de configuración del CP 1242-7 que deben modificarse. Los datos de configuración almacenados en un bloque de datos se describen como estructura del tipo de datos de sistema (SDT) IF_CONF.

IF_CONF está formado por un encabezado y los bloques siguientes, correspondientes a los parámetros o áreas de parámetros del CP en las propiedades de dispositivos del proyecto de STEP 7.

Los datos de configuración del CP que deben modificarse se compilan a modo de bloques IF_CONF. Los parámetros que no deben modificarse no se tienen en cuenta en la estructura de IF_CONF y se quedan tal como se han configurado en el proyecto de STEP 7.

Creación de un DB y de las estructuras IF_CONF

Los parámetros del CP los puede crear dentro del DB IF_CONF en una o más estructuras y cada una con uno o más bloques.

Los tipos de datos de cada bloque deben introducirse con el teclado. No se muestran en la lista de selección. No se tienen en cuenta las mayúsculas/minúsculas al introducir los tipos de datos.

Proceda del siguiente modo para crear IF_CONF:

1. Cree un bloque de datos del tipo "DB global" con acceso al bloque "estándar".
2. Cree una estructura en la tabla de la configuración del parámetro del DB (tipo de datos "Struct").
   Puede determinar el nombre libremente.
3. Inserte un encabezado en esta estructura asignando el nombre del encabezado e introduciéndolo en la celda del tipo de datos "IF_CONF_Header".
   Se crea el encabezado de la estructura con sus tres parámetros (ver abajo).
4. Cree un bloque para el primer parámetro que debe modificarse introduciendo el tipo de datos deseado (por ejemplo, "IF_CONF_APN") en la celda del tipo de datos.
5. Repita el último paso para todos aquellos parámetros que desee modificar con ayuda de la instrucción TC_CONFIG en el CP.
6. A continuación actualice en el encabezado el número de bloques en el parámetro "subfieldCnt".

Encabezado de IF_CONF

Tabla 9-109 IF_CONF_Header

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
0 ... 1	fieldType	UINT		Tipo de bloque: tiene que ser siempre 0.
2 ... 3	fieldId	UINT		ID de bloque: tiene que ser siempre 0.
4 ... 5	subfieldCnt	UINT		Número total de bloques incluidos en la estructura

Parámetros generales de los bloques de parámetros

Cada bloque contiene los siguientes parámetros generales:

- Id
  Este parámetro identifica el bloque correspondiente y no puede modificarse.
- Length
  Este parámetro indica el tamaño del bloque. El valor solo sirve para fines informativos. Los bloques que contienen strings y/o arrays tienen una longitud variable. Debido a los bytes ocultos, la longitud real de los bloques puede ser mayor que la suma de los parámetros mostrados.
- Mode
  Para este parámetro se admiten los valores siguientes:

Tabla 9-110 Valores de "Mode"

Valor	Significado
1	Validez permanente de los datos de configuración No tiene relevancia en el CP 1242-7
2	Validez temporal de los datos de configuración, inclusive el borrado de los datos de configuración permanentes existentes Los datos de configuración permanentes se reemplazan por los bloques de parámetros de IF_CONF.

Bloque para el área de parámetros "Acceso GPRS"

Tabla 9-111 IF_CONF_APN

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	4	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 174
Mode	UINT		Validez (1: permanente, 2: temporal)
AccesspointGPRS	STRING [98]		APN: Nombre del punto de acceso del operador de red GSM a Internet
AccesspointUser	STRING [42]		Nombre de usuario APN
AccesspointPassword	STRING [22]		Contraseña APN

Bloque para el área de parámetros "Identificación CP"

Tabla 9-112 IF_CONF_Login

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	5	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 54
Mode	UINT		Validez (1: permanente, 2: temporal)
ModemName	STRING [22]		ID de acceso El valor no es parametrizable.
ModemPassword	STRING [22]		Contraseña de Telecontrol (máx. 20 caracteres)

Bloque para el área de parámetros "Acceso al servidor Telecontrol"

Este bloque solo debe utilizarse si el servidor Telecontrol se direcciona con un nombre que puede resolverse mediante DNS. Si el servidor Telecontrol se direcciona con su dirección IP, se utilizará el bloque "IF_CONF_TCS_IP_V4".

En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Modo de operación".

Si hay varios servidores Telecontrol, utilice el bloque una vez por servidor.

Tabla 9-113 IF_CONF_TCS_Name

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	6	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 266
Mode	UINT		Validez (1: permanente, 2: temporal)
TcsName	-	-	- reservado -
	STRING [254]		Nombre del servidor Telecontrol que puede resolverse mediante DNS
RemotePort	UINT		Puerto del servidor Telecontrol
Rank	UINT		Prioridad del servidor [1, 2] 1 = servidor Telecontrol principal, 2 = servidor Telecontrol de reserva

Bloque para el área de parámetros "Acceso al servidor Telecontrol"

Este bloque solo debe utilizarse si el servidor Telecontrol se direcciona con su dirección IP. Si el servidor Telecontrol se direcciona con su nombre DNS, se utiliza el bloque "IF_CONF_TCS_Name".

En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Modo de operación".

Si hay varios servidores Telecontrol, utilice el bloque una vez por servidor.

Tabla 9-114 IF_CONF_TCS_IP_v4

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	7	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 14
Mode	UINT		Validez (1: permanente, 2: temporal)
RemoteAddress	IP_V4		Dirección IP del servidor Telecontrol
RemotePort	UINT		Puerto del servidor Telecontrol
Rank	UINT		Prioridad del servidor [1, 2] 1 = servidor Telecontrol principal, 2 = servidor Telecontrol de reserva

Bloque para el área de parámetros "Modo de operación"

En STEP 7, los datos correspondientes se encuentran en las secciones de parámetros "Modo de operación" y "Configuración del módem".

Tabla 9-115 IF_CONF_GPRS_Mode

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	8	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 10
Mode	UINT		Validez (1: permanente, 2: temporal)

Parámetro	Tipo de datos	Valor inicial	Descripción
GPRSmode	UINT		Modo de operación del CP: • 0 = Telecontrol • 1 = GPRS directo
TemporaryStation	BOOL		Bit 0: Conexión temporal Si la opción está activada, el CP establece un enlace de forma temporal para enviar datos. Una vez finalizada correctamente la transferencia de los telegramas, el CP deshace la conexión. • 1: activado (conexión temporal) • 0: desactivado (enlace permanente)
SMS_Enabled	BOOL		Bit 1: Permitir SMS La activación de esta opción permite a la estación S7 enviar SMS. • 1: activado (se permite SMS) • 0: desactivado (sin SMS)

Bloque para el parámetro "SMSC"

En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Configuración del módem".

Tabla 9-116 IF_CONF_SMS_Provider

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	10	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 28
Mode	UINT		Validez (1: permanente, 2: temporal)
SMSProvider	STRING [20]		Número de dispositivo de la central SMS (SMSC) del operador de red GSM con el que se ha cerrado el contrato de telefonía móvil para esta estación.

Bloque para el parámetro "PIN"

En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Configuración del módem".

Tabla 9-117 IF_CONF_PIN

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	11	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 16
Mode	UINT		Validez (1: permanente, 2: temporal)
Pin	STRING [8]		PIN de la tarjeta SIM insertada en el CP Este parámetro no es relevante si se ha configurado el PIN correctamente. En caso de haber configurado el PIN erróneamente, el PIN correcto puede introducirse aquí.

Bloque para tiempos de vigilancia

En STEP 7, los datos correspondientes se encuentran en las áreas de parámetros "Tiempo de espera Keep Alive" y "Modo de operación" del CP.

Tabla 9-118 IF_CONF_TC_Timeouts

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	12	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 12
Mode	UINT		Validez (1: permanente, 2: temporal)
KeepAliveTimeout	-	-	- Reservado - (no parametrizable)
SendTimeout	UINT		Tiempo de vigilancia de conexión: Tiempo de vigilancia de la conexión con el interlocutor (segundos) Relevante en los modos de operación "Telecontrol" y "GPRS directo"
RedialTimeout	UINT		Retardo de repetición de llamada: valor básico del tiempo de espera hasta el próximo intento de establecer un enlace después de fallar el presente. Cada 3 intentos se duplica el valor básico, hasta 900 s como máximo. Rango de valores: 10....600 s. Si se ha configurado un servidor Telecontrol de reserva, en el cuarto intento el CP intentará conectarse con este. Ejemplo: un valor básico de 20 da los siguientes intervalos de llamada: tres veces 20 s, tres veces 40 s, tres veces 80 s, etc. hasta un máximo de 900 s. Sin relevancia para conexiones SMS

Bloque para el área de parámetros "Autorización de alarma"

Tabla 9-119 IF_CONF_WakeupList

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	13	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 246
Mode	UINT		Validez (1: permanente, 2: temporal)
WakeupPhone [1...10]	ARRAY [1...10] of STRING [22]		Número de llamada del dispositivo autorizado como despertador El asterisco (*) al final de un número de abonado sirve de comodín para números con extensiones.

Bloque para el área de parámetros "Redes GSM preferentes"

Tabla 9-120 IF_CONF_PrefProvider

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	14	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 46

Parámetro	Tipo de datos	Valor inicial	Descripción
Mode	UINT		Validez (1: permanente, 2: temporal)
Provider [1...5]	ARRAY [1...5] of STRING [6]		Redes GSM alternativas con prioridad 1 a 5, a las que se conecta el CP de forma preferente. Es posible configurar 5 redes como máximo. La n.º 1 tiene la máxima prioridad y la n.º 5 la mínima. Entrada del Public Land Mobile Network (PLMN) del operador de red, consistente en Mobile Country Code (MCC) y Mobile Network Code (MNC). Ejemplo (red de pruebas de Siemens AG): 26276

Bloque para el área de parámetros "Configuración DNS"

Tabla 9-121 IF_CONF_DNS

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	16	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 14
Mode	UINT		Validez (1: permanente, 2: temporal)
DNS_IP [1]	IP_V4		Dirección IP del 1.er servidor Domain Name System
DNS_IP [2]	IP_V4		Dirección IP del 2.º servidor Domain Name System

Bloque para el área de parámetros "Sincronización horaria"

Tabla 9-122 IF_CONF_NTP

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	17	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 24
Mode	UINT		Validez (1: permanente, 2: temporal)
NTP_IP [1]	ARRAY [1...4] of IP_V4		Dirección IP del servidor NTP 1
...	...		(dirección IP del servidor NTP 2...3)
NTP_IP [4]	ARRAY [1...4] of IP_V4		Dirección IP del servidor NTP 4

Bloque para activar / desactivar usuarios de TeleService

SDT para activar o desactivar usuarios de TeleService que ya se han configurado en el proyecto STEP 7 del CP. En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Configuración TeleService" > "Administración de usuarios de TeleService".

Tabla 9-123 IF_CONF_GPRS_UserList

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	19	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 506
Mode	UINT		Validez (1: permanente, 2: temporal)
GPRS_User [1...10]	ARRAY [1...10] of GPRS_User		Usuarios de TeleService de n.º 1 a n.º 10 como máximo

La matriz se forma a partir de los juegos de parámetros para los usuarios de TeleService ("GPRS_User" [1...n]).

Tabla 9-124 GPRS_User [n] (parámetro para usuarios de TeleService)

Parámetro	Tipo de datos	Valor inicial	Descripción
UserName [n]	STRING [22]		Nombre de usuario de TeleService
Password [n]	STRING [22]		- El string tiene que estar vacío -
Diag_Allowed [n]	BOOL		- Reservado - (no parametrizable)
Teleserv_Allowed [n]	BOOL		Activación del usuario de TeleService • 0 = usuario desactivado • 1 = usuario activado
FW_Load_Allowed [n]	BOOL		- Reservado - (no parametrizable)

Bloque para la parametrización del acceso TeleService (Nombre DNS del servidor)

Datos de acceso del servidor de TeleService (centralita).

En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Configuración TeleService".

Si hay dos servidores de TeleService, utilice el bloque una vez por servidor.

Tabla 9-125 IF_CONF_TS_Name

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	20	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 266
Mode	UINT		Validez (1: permanente, 2: temporal)
ts_name	String [254]		Nombre del servidor de TeleService que puede resolverse mediante DNS
RemotePort	UINT		Puerto de la estación de ingeniería
Rank	UINT		Prioridad del servidor [1] o [2] 1 = servidor 1, 2 = servidor 2

Bloque para la parametrización del acceso TeleService (dirección IP del servidor)

Datos de acceso del servidor de TeleService (centralita).

En STEP 7, los datos correspondientes se encuentran en el área de parámetros "Configuración TeleService".

Si hay dos servidores de TeleService, utilice el bloque una vez por servidor.

Tabla 9-126 IF_CONF_TS_IF_V4

Parámetro	Tipo de datos	Valor inicial	Descripción
Id	UINT	21	Identificación del bloque de parámetros
Length	UINT		Tamaño del bloque de parámetros en bytes: 14
Mode	UINT		Validez (1: permanente, 2: temporal)
RemoteAddress	IP_V4		Dirección IP del servidor de TeleService
RemotePort	UINT		Puerto del servidor de TeleService
Rank	UINT		Prioridad del servidor [1] o [2] 1 = servidor 1, 2 = servidor 2

Punto a punto

Sinopsis de la comunicación Freeport

STEP 7 ofrece instrucciones avanzadas que permiten realizar la comunicación Freeport utilizando un protocolo especificado en el programa de usuario. Dichas instrucciones se dividen en dos categorías:

- Instrucciones de configuración
- Instrucciones de comunicación

Nota

Capacidad de la CPU

Las instrucciones Freeport se comunican con los módulos de comunicaciones mediante lectura o escritura de registros.
Por ello, al utilizar las instrucciones debe tenerse en cuenta la capacidad de lectura y escritura de registros de la CPU correspondiente.
Si varias instrucciones de una CPU deben leer o escribir registros de forma simultánea, es posible que el programa de usuario tenga que llamar las instrucciones con un desfase temporal.

Instrucciones de configuración

Antes de que el programa de usuario pueda iniciar la comunicación Freeport, es necesario configurar la interfaz de comunicación y los parámetros para transmitir y recibir datos.

La configuración de interfaces y datos se realiza para cada CM en la configuración de dispositivos o con las siguientes instrucciones del programa de usuario:

- Port_Config (Página 5399)
- Send_Config (Página 5401)
- Receive_Config (Página 5403)
- P3964_Config (Página 5408)

ATENCIÓN
Configuración de dispositivos <-> Instrucciones de configuración Los parámetros de la configuración de dispositivos se transfieren al CM en cada Power On de la CPU (retorno de la tensión). Los parámetros de las instrucciones de configuración se transfieren al CM de acuerdo con su programa de usuario. No existe ninguna sincronización entre los parámetros de la configuración de dispositivos y los parámetros de las instrucciones de configuración, es decir, los parámetros de las instrucciones de configuración no se aplican a la configuración de dispositivos de la CPU. El programa de usuario permite controlar qué parámetros se aplican en el CM en cada momento.

Instrucciones de comunicación

Las instrucciones de la comunicación Freeport permiten al programa de usuario enviar datos a las interfaces de comunicación y recibirlos de estas. Los CM envían datos a los dispositivos de comunicación y los reciben de ellos.

- Send_P2P (Página 5410)
- Receive_P2P (Página 5413)

Nota

Coherencia de datos

- En el caso de que los datos que vayan a enviarse deban transferirse de forma coherente, no deben modificarse desde el ascenso del flanco del parámetro REQ hasta establecer DONE con la instrucción Send_P2P .
- Si los datos recibidos deben leerse de forma coherente, solo deben evaluarse en el ciclo en el que NDR = TRUE.

Con instrucciones adicionales es posible inicializar el búfer de recepción, así como consultar y activar señales RS232 especiales.

- Receive_Reset (Página 5415)
- Signal_Get (Página 5415)
- Signal_Set (Página 5416)

Las siguientes instrucciones permiten leer y escribir funciones avanzadas, siempre que el módulo las soporte.

- Get_Features (Página 5418)
- Set_Features (Página 5418)

Todas las instrucciones Freeport funcionan de forma asíncrona. Por ello, las instrucciones deben llamarse repetidas veces hasta que el parámetro de salida DONE indique que ha finalizado la ejecución.

Utilizando una arquitectura de consultas, el programa de usuario puede determinar el estado de transmisión y recepción. Send_P2P y Receive_P2P pueden ejecutarse a la vez. Los módulos de comunicaciones respaldan los datos enviados y recibidos hasta un tamaño máximo del búfer específico del módulo.

Nota

Resolución de tiempos de bit

Para diversos parámetros se indica el número de tiempos de bits en la velocidad de transferencia configurada. Debido a la indicación del parámetro en tiempos de bit, dicho parámetro es independiente de la velocidad de transferencia. Todos los parámetros con la unidad de tiempos de bit pueden indicarse con un número máximo de 65535.

Aplicación de las instrucciones

Las instrucciones Freeport deben llamarse cíclicamente para consultar los datos recibidos o el fin de una transmisión.

En función del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).
Si la petición se ha completado con DONE = TRUE o NDR = TRUE, se ha ejecutado sin errores.

Nota

Guardar STATUS

Los parámetros DONE, NDR, ERROR y STATUS solo permanecen durante una ejecución del bloque. Por ello, para visualizar STATUS debe copiarlo en un área de datos libre.

maestro

Secuencia típica de un maestro:

1. La instrucción Send_P2P lanza una transferencia al CM.
   Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos.
2. La instrucción Send_P2P se ejecuta en ciclos consecutivos para consultar el estado del proceso de transferencia.
3. Si la instrucción Send_P2P en la salida DONE notifica que la transferencia ha finalizado, el código de usuario puede preparar la recepción de la respuesta.

4. La instrucción Receive_P2P se ejecuta repetidamente para consultar una respuesta. Una vez que el CM ha leído los datos de respuesta, la instrucción Receive_P2P copia la respuesta en la CPU y notifica en la salida NDR que se han recibido datos nuevos.
5. El programa de usuario puede procesar la respuesta.
6. Regreso al paso 1 y repetición de la secuencia.

esclavo

Secuencia típica de un esclavo:

1. El programa de usuario ejecuta la instrucción Receive_P2P en cada ciclo.
2. Una vez que el CM ha recibido una solicitud, la instrucción Receive_P2P notifica en la salida NDR que hay datos nuevos preparados y la solicitud se copia en la CPU.
3. El programa de usuario procesa la solicitud y genera una respuesta.
4. Con la instrucción Send_P2P , la respuesta se devuelve al maestro.
5. La instrucción Send_P2P debe ejecutarse repetidamente para asegurarse de que el proceso de transmisión tiene lugar realmente.
6. Regreso al paso 1 y repetición de la secuencia.

El esclavo debe procurar que Receive_P2P se llame tantas veces como sea necesario para que el maestro pueda recibir una transferencia antes de que cancele el proceso por rebasar el tiempo mientras espera una respuesta. Para cumplir esta tarea, el programa de usuario puede llamar a Receive_P2P desde un OB de ciclo cuyo tiempo de ciclo sea lo suficientemente corto como para recibir una transferencia del maestro antes de que transcurra el timeout ajustado. Si el tiempo de ciclo para el OB se ajusta de modo que se lleven a cabo dos ejecuciones dentro del timeout ajustado, el programa de usuario puede recibir todas las transferencias sin pérdidas.

Parámetros comunes para las operaciones Freeport

Tabla 9-127 Parámetros de entrada comunes de las instrucciones Freeport

Parámetro	Descripción
REQ	Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. La creación de un flanco adicional en REQ solamente puede tener lugar tras finalizar la petición (DONE o bien ERROR). Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa). Al insertar una instrucción Freeport en el programa, STEP 7 le pedirá que indique el DB de instancia (o que cree mediante STEP 7 un DB de instancia correspondiente). Utilice un DB unívoco para cada llamada de instrucción PtP.
PORT	Al configurar el módulo de comunicaciones se asigna una dirección de puerto. Con el parámetro PORT se comunica a la instrucción la asignación a un determinado módulo de comunicaciones. Tras la configuración se puede seleccionar un nombre simbólico para el puerto estándar. El valor de puerto asignado del CM es la propiedad "identificación de hardware" de la configuración de dispositivos en el S7-1500 o la "dirección de entrada" en el S7-300/400. El nombre simbólico del puerto se asigna en la tabla de símbolos.

Los parámetros de salida DONE, NDR, ERROR y STATUS de las instrucciones Freeport indican el estado de ejecución de las funciones Freeport.

Tabla 9-128 Parámetros de salida DONE, NDR, ERROR y STATUS

Parámetro	Tipo de datos	Estándar	Descripción
DONE	Bool	FALSE	Durante un ciclo se pone a TRUE para indicar que la última solicitud ha concluido correctamente; en caso contrario, FALSE.
NDR	Bool	FALSE	Durante un ciclo se pone a TRUE para indicar que se han recibido nuevos datos; en caso contrario, FALSE.
ERROR	Bool	FALSE	Durante un ciclo se pone a TRUE para indicar que la última solicitud ha concluido con error, el código de error correspondiente se encuentra en STATUS; en caso contrario, FALSE.
STATUS	Word	16#0000 o 16#7000	Estado del evento: • Si está activado el bit DONE o NDR, STATUS se ajusta a 0/16#7000 o a un código de error específico. • Si está activado el bit ERROR, se muestra en STATUS un código de error. • Si no se activa ninguno de los bits mencionados, la instrucción puede devolver resultados de estado que describen el estado actual de la función. El valor en STATUS es válido hasta la siguiente llamada de esta instrucción (con la misma dirección única de puerto).

Tabla 9-129 Parámetros de entrada/salida COM_RST

Parámetro	Tipo de datos	Estándar	Descripción
COM_RST	Bool	FALSE	Inicialización de la instrucción Con TRUE se inicializa la instrucción. Seguidamente, COM_RST se ajusta de nuevo a FALSE. Nota: En el arranque, COM_RST debe ajustarse a TRUE y, después, no deberían modificarse más los parámetros, p. ej., no asignar ningún valor al llamar la instrucción. Tras inicializar el DB de instancia, la instrucción resetea COM_RST.

Nota

Tenga en cuenta que los parámetros DONE, NDR, ERROR y STATUS solo están activados durante un ciclo.

Tabla 9-130 Códigos de error comunes

Código de error	Descripción
16#0000	No hay error
16#7000	Función inactiva
16#7001	Primera llamada tras iniciar una solicitud.
16#7002	Llamada posterior tras iniciar una solicitud.
16#8x3A	Puntero no admisible en parámetro x

Tabla 9-131 Clases de errores comunes del parámetro STATUS

Descripción de la clase	Clases de error	Descripción
Configuración del puerto	16#81Ax	Para la descripción de errores frecuentes en la configuración de interfaces
Configuración de transmisión	16#81Bx	Para la descripción de errores en la configuración de transmisión
Configuración de recepción	16#81Cx	Para la descripción de errores en la configuración de recepción
Transferir	16#81Dx	Para la descripción de errores del tiempo de ejecución al transmitir
Recibir	16#81Ex	Para la descripción de errores del tiempo de ejecución al recibir
Señales cualificadoras RS232	16#81Fx	Para la descripción de errores en relación con el procesamiento de señales

Port_Config: configurar el puerto de comunicación PtP

Descripción

La instrucción Port_Config (configuración del puerto) permite modificar parámetros como la velocidad de transferencia desde el programa durante el funcionamiento. Los datos pendientes del CM se borran al ejecutar Port_Config.

Los cambios de configuración de Port_Config no se guardan en la CPU sino en el CM. En caso de retorno de la tensión, el CM se parametriza con los datos guardados en la configuración de dispositivos.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
PROTOCOL	IN	UInt	Word	0	Protocolo • 0 = protocolo Freeport • 1 = protocolo 3964(R)

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
BAUD	IN	UInt	Word	6	Velocidad de transferencia del puerto: • 1 = 300 Bit/s • 2 = 600 Bit/s • 3 = 1200 Bit/s • 4 = 2400 Bit/s • 5 = 4800 Bit/s • 6 = 9600 Bit/s • 7 = 19200 Bit/s • 8 = 38400 Bit/s • 9 = 57600 Bit/s • 10 = 76800 Bit/s • 11 = 115200 Bit/s
PARITY	IN	UInt	Word	1	Paridad del puerto: • 1 = ninguna • 2 = paridad par • 3 = paridad impar • 4 = paridad Mark • 5 = paridad Space • 6 = cualquiera
DATABITS	IN	UInt	Word	1	Bits por carácter: • 1 = 8 bits de datos • 2 = 7 bits de datos
STOPBITS	IN	UInt	Word	1	Bits de parada: • 1 = 1 bit de parada • 2 = 2 bits de parada
FLOWCTRL	IN	UInt	Word	1	Control del flujo: • 1 = no hay control del flujo • 2 = XON/XOFF • 3 = RTS de HW siempre ON • 4 = RTS de HW con protocolo de intercambio • 5 = RTS de HW siempre ON, ignorar DTR/DSR
XONCHAR	IN	Char		16#0011	Indica el carácter que sirve de carácter XON. Suele ser un carácter DC1 (11H). Este parámetro solo se evalúa cuando está activado el control del flujo de software.
XOFFCHAR	IN	Char		16#0013	Indica el carácter que sirve de carácter XOFF. Suele ser un carácter DC3 (13H). Este parámetro solo se evalúa cuando está activado el control del flujo de software.
WAITIME	IN	UInt	Word	2000	Indica el tiempo durante el que se espera un carácter XON tras recibir un carácter XOFF o bien el tiempo durante el que se espera la señal CTS = ON tras CTS = OFF (de 0 a 65535 ms). Este parámetro solo se evalúa cuando está activado el control del flujo.

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
MODE	IN	USInt	Byte	0	Modo de operación Los modos de operación admitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) modo a cuatro hilos (punto a punto) • 2 = dúplex (RS422) modo a cuatro hilos (maestro multipunto, CM PtP (ET 200SP)) • 3 = dúplex (RS422) modo a cuatro hilos (esclavo multipunto, CM PtP (ET 200SP)) • 4 = semidúplex (RS485) modo a dos hilos
LINE_PRE	IN	USInt	Byte	0	Estado inicial de la línea de recepción Los estados iniciales admitidos son: • 0 = "sin" estado inicial • 1 = señal R(A) = 5 V, señal R(B) = 0 V (detección Break): en este estado inicial es posible una detección de rotura (Break). Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • 2 = señal R(A) = 0 V, señal R(B) = 5 V: este estado inicial corresponde al estado de reposo (ninguna transmisión activa). Con este estado inicial no es posible la detección de rotura (Break).
CABLE_BRK	IN	USInt	Byte	0	Detección Break Se admiten: • 0 = detección Break desactivada • 1 = detección Break activada
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Port_Config Con 1 se inicializa la instrucción. Seguidamente, la instrucción pone COM_RST de nuevo a 0.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Send_Config: configurar el emisor PtP

Descripción

Con la instrucción Send_Config (configuración de transmisión) puede cambiar los parámetros de transmisión durante el funcionamiento desde el programa (condiciones que marcan el

principio y fin de los datos que deben enviarse). Los datos pendientes en un CM se borran cuando se ejecuta Send_Config.

Los cambios de configuración de Send_Config no se guardan en la CPU sino en el CM. Los parámetros guardados en la configuración de dispositivos se restablecen al retornar la tensión de la CPU o del módulo de comunicaciones.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Activa el cambio de configuración con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
RTSONDLY	IN	UInt	Word	0	Número de milisegundos que debe esperarse tras la activación de RTS antes de iniciar una transferencia de datos de envío. Este parámetro solo es válido cuando está activado el control del flujo de hardware. El rango válido es de 0 a 65535 ms. El valor 0 desactiva la función.
RTSOFFDLY	IN	UInt	Word	0	Número de milisegundos que debe esperarse tras la transferencia de datos de envío antes de que se desactive RTS: este parámetro solo es válido cuando está activado el control del flujo de hardware. El rango válido es de 0 a 65535 ms. El valor 0 desactiva la función.
BREAK	IN	UInt	Word	0	Este parámetro indica que se envía un BREAK antes de iniciar cada trama durante el número indicado de tiempos de bit. El número máximo de tiempos de bit es 65535. El valor 0 desactiva la función.
IDLELINE	IN	UInt	Word	0	Este parámetro indica que la línea permanece vacía antes de iniciar cada trama durante el número indicado de tiempos de bit. El número máximo de tiempos de bit es 65535. El valor 0 desactiva la función.
USR_END	IN	STRING[2]		0	Entrada de los caracteres de fin. Se pueden configurar como máximo 2 caracteres de fin. Se envían el o los caracteres de fin, independientemente de la longitud de trama parametrizada.
APP_END	IN	STRING[5]		0	Entrada de los caracteres que deben añadirse. Se pueden añadir como máximo 5 caracteres.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Send_Config Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Receive_Config: configurar el receptor PtP

Descripción

Con la instrucción Receive_Config (configuración de recepción) puede cambiar los parámetros de recepción durante el funcionamiento desde el programa. Esta instrucción configura las condiciones que marcan el principio y fin de los datos recibidos. Los datos pendientes en un CM se borran cuando se ejecuta Receive_Config.

Los cambios de configuración de Port_Config no se guardan en la CPU sino en el CM. Los parámetros guardados en la configuración de dispositivos se restablecen al retornar la tensión de la CPU o del módulo de comunicaciones.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Activa el cambio de configuración con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
Receive_Conditions	IN	Variant	Any	-	La estructura de datos de Receive_Conditions indica las condiciones de inicio y fin en virtud de las cuales se identifica el principio y fin de una trama.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Receive_Config Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

.

Condiciones de inicio de la instrucción Receive_P2P

La instrucción Receive_P2P utiliza la configuración de dispositivos o bien la indicada por la instrucción Receive_Config para determinar el inicio y el fin de tramas de la comunicación Freeport. El inicio de la trama queda definido por las condiciones de inicio. El inicio de la trama puede especificarse mediante una o varias condiciones de inicio.
Si están activados tanto Break como Idle Line, debe cumplirse Break y, además, Idle Line. A continuación, basta una de las condiciones restantes (carácter de inicio o secuencia de inicio) para iniciar la transmisión de datos.
La condición de inicio "cualquier carácter" no puede combinarse con otras condiciones de inicio.

Estructura del tipo de datos del parámetro Receive_Conditions, parte 1 (condiciones de inicio)

Tabla 9-132 Estructura de Receive_Conditions para condiciones de inicio

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
START.START COND	IN	Word	16#0002	Indicación de la condición de inicio • 01H - detección del carácter de inicio • 02H - cualquier carácter • 04H - detección de Line Break • 08H - detección de Idle Line • 10H - detección de la secuencia de inicio 1 • 20H - detección de la secuencia de inicio 2 • 40H - detección de la secuencia de inicio 3 • 80H - detección de la secuencia de inicio 4 Las condiciones de inicio pueden combinarse sumando los valores.
START.IDLETIME	IN	Word	16#0028	Número de tiempos de bit necesarios en vacío para que se detecte un nuevo inicio de la trama (valor predeterminado: W#16#28). Solo en combinación con la condición "detección de Idle Line". 0 a FFFF
START.START CHAR	IN	Byte	16#0002	Carácter de inicio para la condición "carácter de inicio". (valor predeterminado: B#16#2)

Parámetro	Declaración	Tipo de datos	Ajuste predetermina do	Descripción
START.SEQ[1]. CTL	IN	Byte	0	Secuencia de inicio 1, desactivar/activar comparación para cada carácter: (valor predeterminado: B#16#0) Estos son los bits de activación para cada carácter de la secuencia de inicio. • 01H - carácter 1 • 02H - carácter 2 • 04H - carácter 3 • 08H - carácter 4 • 10H - carácter 5 Si se desactiva un bit para un carácter determinado, significa que en esta posición de la secuencia cada carácter representa una secuencia de inicio válida (p. ej. 1FH = se interpretan los 5 caracteres).
START.SEQ[1]. STR[1] .. START.SEQ[1]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 1, carácter de inicio (5 caracteres).
START.SEQ[2]. CTL	IN	Byte	0	Secuencia de inicio 2, desactivar/activar comparación para cada carácter. Valor predeterminado: B#16#0)
START.SEQ[2]. STR[1] .. START.SEQ[2]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 2, carácter de inicio (5 caracteres).
START.SEQ[3]. CTL	IN	Byte	0	Secuencia de inicio 3, desactivar/activar comparación para cada carácter. Valor predeterminado: B#16#0
START.SEQ[3]. STR[1] .. START.SEQ[3]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 3, carácter de inicio (5 caracteres).
START.SEQ[4]. CTL	IN	Byte	0	Secuencia de inicio 4, desactivar/activar comparación para cada carácter. Valor predeterminado: B#16#0
START.SEQ[4]. STR[1] .. START.SEQ[4]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 4, carácter de inicio (5 caracteres).

Ejemplo

Observe los siguientes datos recibidos en código hexadecimal: "68 10 aa 68 bb 10 aa 16". Encontrará las secuencias de caracteres de inicio configuradas en la tabla siguiente. Las secuencias de caracteres de inicio se evalúan una vez se ha recibido correctamente el primer carácter 68H. Tras la recepción correcta del cuarto carácter (el segundo 68H) se ha cumplido la condición de inicio 1. Si se cumplen las condiciones de inicio, empieza la evaluación de las condiciones de fin.

El procesamiento de la secuencia de caracteres inicial puede finalizarse debido a diferentes errores de paridad, framing o intervalos entre los caracteres. Dichos errores hacen que los

datos no se reciban porque no se cumple la condición de inicio (se genera un mensaje de error).

Tabla 9-133 Condiciones de inicio

Condición de inicio	Primer carácter	Primer carácter +1	Primer carácter +2	Primer carácter +3	Primer carácter +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Condiciones de fin de la instrucción Receive_P2P

El final de una trama se define por la primera aparición de una o varias condiciones de fin configuradas.

Las condiciones de fin se configuran en las propiedades de la interfaz de comunicación de la configuración de dispositivos o bien mediante la instrucción Receive_Config. Tras retornar la tensión de la CPU o del módulo de comunicaciones, los parámetros de recepción (condiciones de inicio y fin) se ponen de nuevo a los ajustes de la configuración de dispositivos. Si el programa de usuario de STEP 7 ejecuta Receive_Config, los ajustes cambian a los parámetros de Receive_Config .

Estructura del tipo de datos del parámetro Receive_Conditions, parte 2 (condiciones de fin)

Tabla 9-134 Estructura de Receive_Conditions para condiciones de fin

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
END.ENDCOND	IN	Word	0	Este parámetro indica la condición para el fin de trama: • 01H - tiempo de respuesta excedido • 02H - tiempo de espera del mensaje excedido • 04H - tiempo de retardo de caracteres • 08H - longitud máxima de trama • 10H - leer longitud del mensaje en el mensaje (N+LEN+M) • 20H - secuencia de fin • 40H - longitud fija de trama
END.FIXLEN	IN	Word	1	Longitud fija de trama: solo se utiliza si se selecciona la condición de fin "longitud fija". De 1 a 4000 bytes (en función del módulo hasta 4 kbytes)
END.MAXLEN	IN	Word	1	Longitud máxima de trama: solo se utiliza si se selecciona la condición de fin "longitud máxima". De 1 a 4000 bytes (en función del módulo hasta 4 kbytes)
END.N	IN	Word	0	Posición de byte del campo de longitud en la trama. Solo se utiliza con la condición de fin N+LEN+M. De 1 a 4000 bytes (en función del módulo hasta 4 kbytes)
END.LENGTHSIZE	IN	Word	0	Tamaño del campo de longitud (1, 2 o 4 bytes). Solo se utiliza con la condición de fin N+LEN+M.

Parámetro	Declaración	Tipo de datos	Ajuste predetermina do	Descripción
END.LENGTHM	IN	Word	0	Número de caracteres posteriores al campo de longitud que no están incluidos en el valor del mismo. Este dato solo se utiliza con la condición de fin N+LEN+M. 0 a 255 bytes
END.RCVTIME	IN	Word	200	Indique el tiempo durante el que debe esperarse el primer carácter recibido tras enviar una trama. La instrucción de recepción finalizará con un mensaje de error si dentro del tiempo indicado no se recibe ningún carácter correctamente. Este dato solo se utiliza con la condición "tiempo de respuesta excedido". (0 a 65535 ms). Nota: Este parámetro no puede utilizarse como criterio único de fin, sino solamente en combinación con al menos otra condición de fin.
END.MSGTIME	IN	Word	200	Indique el tiempo que debe esperarse a la recepción completa de la trama tras haber recibido el primer carácter. Este parámetro solo se utiliza cuando está seleccionada la condición "tiempo de espera del mensaje excedido". (0 a 65535 ms)
END.CHARGAP	IN	Word	12	Indique el número máximo de tiempos de bit entre los caracteres. Si el número de tiempos de bit entre los caracteres rebasa por exceso el valor indicado, se cumple la condición de fin. Este dato solo se utiliza con la condición "tiempo de retardo de caracteres". (0 a 65535 tiempos de bit)
END.SEQ.CTL	IN	Byte	0	Secuencia de caracteres 1, desactivar/activar comparación para cada carácter: Estos son los bits de activación para cada carácter de la secuencia de fin. El carácter 1 es el bit 0, el carácter 2 es el bit 1, ..., el carácter 5 es el bit 4. Si se desactiva un bit para un carácter concreto, significa que en esta posición de la secuencia cada carácter representa una coincidencia.
END.SEQ.STR[1] .. END.SEQ.STR[5]	IN	Char[5]	0	Secuencia de caracteres 1, carácter de inicio (5 caracteres)

Tabla 9-135 Parámetros generales de la instrucción Receive_P2P

Parámetro	Declaración	Tipo de datos	Ajuste predetermina do	Descripción
GENERAL.MBUF_SIZE	IN	Byte	255	Entrada del número de tramas que deben guardarse en el búfer de recepción del CM. Si no hay otras condiciones activas que influyan en el comportamiento del búfer de recepción (impedir sobrescritura, control del flujo de datos), tras alcanzar el límite se rechazan las nuevas tramas (1 a 255 tramas).
GENERAL.OW_PROT	IN	Byte	0	Activa la función para no sobrescribir la trama en búfer, en caso de que el CM reciba una nueva trama sin que se haya efectuado la lectura del búfer de recepción del CM. Con esto se evita una eventual pérdida de las tramas de recepción que ya están en el búfer. • 0 - no activado • 1 - activado
GENERAL.CLR_MBUF	IN	Byte	0	Activa el borrado del búfer de recepción en el arranque de la CPU. Si este parámetro está activado, el búfer de recepción se borra automáticamente en la transición de la CPU de STOP a RUN. En el búfer de recepción solo hay tramas que fueron recibidas después del arranque de la CPU. • 0 - no activado • 1 - activado

P3964_Config: Configurar el protocolo 3964(R)

Descripción

La instrucción P3964_Config (configuración del protocolo) permite modificar desde el programa y durante el funcionamiento los parámetros del protocolo 3964(R) como tiempo de retardo de caracteres, prioridad y control de bloque.

Los cambios de configuración de P3964_Config no se guardan en la CPU sino en el CM. Los parámetros guardados en la configuración de dispositivos se restablecen al retornar la tensión de la CPU o del módulo de comunicaciones.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BCC	IN	SInt	Byte	1	Activa/desactiva el uso de la comprobación de bloque • 0 = sin control de bloque • 1 = con control de bloque
Priority	IN	SInt	Byte	1	Ajuste de la prioridad • 0 = prioridad baja • 1 = prioridad alta
CharacterDelayTime	IN	UInt	Word	16#00DC	Ajuste del tiempo de retardo de caracteres (en función de la velocidad de transferencia de datos ajustada) (valor predeterminado: 220 ms) De 1 a 65535 ms
AcknDelayTime	IN	UInt	Word	16#07D0	Ajuste del tiempo de retardo de acuse (en función de la velocidad de transferencia de datos ajustada) (valor predeterminado: 2000 ms) De 1 a 65535 ms
BuildupAttempts	IN	SInt	Byte	16#0006	Ajuste del número de intentos de establecer conexión (valor predeterminado: 6 intentos de establecer conexión) 1 a 255
RepetitionAttempts	IN	SInt	Byte	16#0006	Ajuste del número de intentos de transmisión (valor predeterminado: 6 intentos de establecer conexión) 1 a 255
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción P3964_Config Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Send_P2P: transferir datos

Descripción

La instrucción Send_P2P (enviar datos punto a punto) inicia la transferencia de los datos y transfiere el contenido del búfer asignado al módulo de comunicaciones. El programa de la CPU se sigue ejecutando mientras el CM envía los datos a la velocidad de transferencia. No puede haber más de una instrucción de transmisión a la vez para cada módulo de comunicaciones. El CM emite un error si se ejecuta una segunda instrucción Send_P2P mientras el CM está enviando una trama.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BUFFER	IN	Variant	Any	0	Este parámetro señala el área de memoria del búfer de transmisión. Notas: • No se soportan datos booleanos ni campos booleanos. • Si el búfer de transmisión está en el área de memoria optimizada, la máxima longitud admisible de datos enviados es de 1024 bytes. Excepción: los arrays de Byte, Word o DWord se soportan hasta una longitud de 4096 bytes. • Si el búfer de transmisión es de tipo String o WString, el contenido del string se transmite sin la longitud actual y máxima.
LENGTH	IN	UInt	Word	0	Longitud de los datos que deben transmitirse en bytes. Con LENGTH = 0 se transfiere completamente el área de memoria direccionada en el parámetro BUFFER.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Send_P2P Con 1 se inicializa la instrucción. Seguidamente, la instrucción pone COM_RST de nuevo a 0.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente

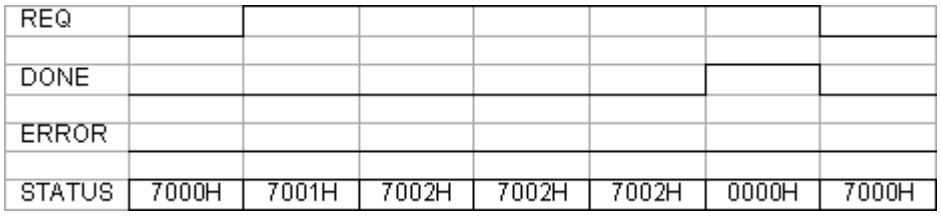
Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Parámetros

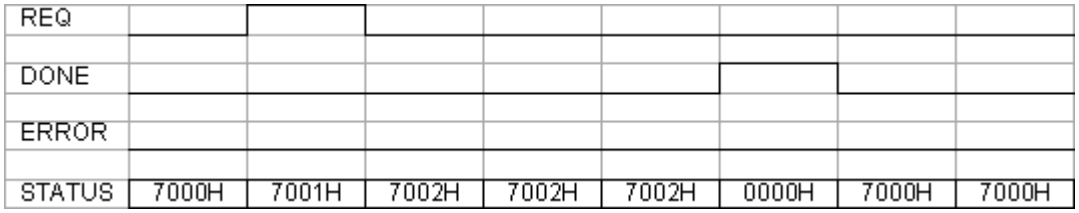
Cuando una instrucción de transmisión se está procesando, las salidas DONE y ERROR están en estado FALSE. Una vez finalizada una instrucción de transmisión, una de las salidas DONE o ERROR se pone a TRUE durante un ciclo para notificar el estado de la instrucción de transmisión. Si ERROR está en estado TRUE, puede evaluarse el código de error en la salida STATUS.

La instrucción emite el estado 16#7001 si la interfaz de comunicación acepta los datos transmitidos. Las ejecuciones posteriores de Send_P2P emiten el valor 16#7002 mientras el CM sigue enviando. Tras finalizar la instrucción de transmisión, el CM emite el estado 16#0000 para la instrucción de transmisión (siempre que no se produzca ningún error). Las ejecuciones posteriores de Send_P2P con REQ = 0 dan el estado 16#7000 (libre).

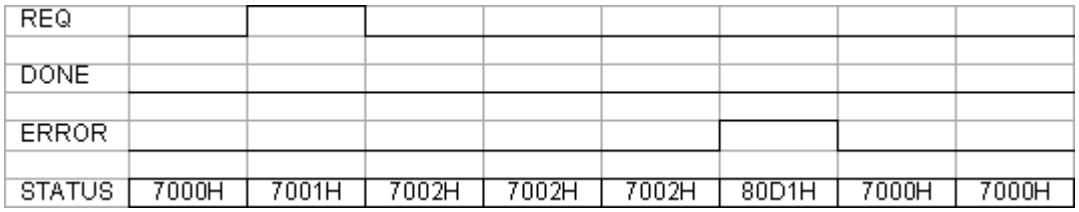
El diagrama siguiente muestra la relación entre los valores de salida y REQ. Se presupone que la instrucción se llama cíclicamente para comprobar el estado del proceso de transmisión (se representa mediante los valores de STATUS).



La figura siguiente muestra cómo los parámetros DONE y STATUS solo son válidos durante un ciclo si en la línea REQ (durante un ciclo) hay un impulso para lanzar la instrucción de transmisión.



La figura siguiente muestra la relación de los parámetros DONE, ERROR y STATUS en caso de error.



Los valores de DONE, ERROR y STATUS son válidos únicamente hasta que Send_P2P se ejecuta con el mismo DB de instancia.

Uso de los parámetros LENGTH y BUFFER en operaciones de comunicación

Interacción de los parámetros LENGTH y BUFFER para Send_P2P

El tamaño mínimo de los datos que puede enviar la instrucción Send_P2P es de un byte. El parámetro BUFFER define el tamaño de los datos que deben enviarse si durante una llamada se transfiere un "0" en el parámetro LENGTH. Para el parámetro BUFFER no es posible utilizar el tipo de datos Bool ni matrices del tipo Bool.

Tabla 9-136 Parámetro LENGTH

LENGTH	Descripción
> 0	Se transfiere el número de bytes parametrizado.
= 0	Se transfiere todo el contenido del búfer de transmisión direccionado mediante BUFFER. Si BUFFER señala a un String, se transfiere todo el contenido del String sin los bytes con la longitud máxima y real.

Tabla 9-137 Parámetro BUFFER

BUFFER	Descripción
Tipo de datos simple	En el envío, se aplica lo siguiente: El valor LENGTH debe contener el número de bytes de este tipo de datos. Ejemplo: con un valor Word, LENGTH debe ser dos. Con un valor DWord o Real, LENGTH debe ser cuatro.
Estructura	En relación con la memoria optimizada, se aplica lo siguiente: la longitud máxima permitida de BUFFER es 1024 Byte; de lo contrario, se admiten hasta 4 kbytes en función del módulo. En la transmisión se aplica lo siguiente: el valor de LENGTH puede contener un número de bytes inferior a la longitud completa de bytes de la estructura; en este caso, solo se envían los primeros bytes LENGTH de la estructura de BUFFER.

BUFFER	Descripción
Array	En relación con la memoria optimizada, se aplica lo siguiente: si el tipo de datos de la matriz es diferente de Byte, Word o DWord, la longitud máxima permitida del búfer es 1024 Byte. Si la memoria no está optimizada, pueden transferirse hasta 4 kbytes en función del módulo, independientemente de la estructura de datos. En la transmisión se aplica lo siguiente: el valor de LENGTH puede contener un número de bytes inferior a la longitud completa de bytes de la matriz, siendo siempre un múltiplo del número de bytes del elemento de datos. Ejemplo: el parámetro LENGTH de una matriz del tipo Word debe ser un múltiplo de dos y, con una matriz del tipo Real, un múltiplo de cuatro. Si se indica LENGTH, se transfiere el número de elementos de matriz. Si, por ejemplo, BUFFER contiene una matriz con 15 elementos DWord (60 Bytes en total) y se indica LENGTH = 20, se transfieren los cinco primeros elementos DWord de la matriz.
String	El parámetro LENGTH contiene el número de caracteres que deben transmitirse. Solo se transmiten los caracteres de String. Los bytes con la longitud máxima y real de String no se transmiten.

Receive_P2P: recibir datos

Descripción

La instrucción Receive_P2P (datos recibidos mediante una comunicación punto a punto) comprueba las tramas que se han recibido en el CM. Cuando hay una trama disponible, el CM la transfiere a la CPU. Un error de recepción se indica en el parámetro STATUS.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7 300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BUFFER	IN	Variant	Any	0	Este parámetro señala la dirección de inicio del búfer de recepción. Este búfer debe ser lo suficientemente grande como para recibir la longitud máxima de la trama. Nota: • No se soportan datos booleanos o campos booleanos. • Si el búfer de recepción está en el área de memoria optimizada, la máxima longitud admisible de datos recibidos es de 1024 bytes. Excepción: los arrays de Byte, Word o DWord se soportan hasta una longitud de 4096 bytes. • Si el búfer de recepción es de tipo String o WString, los datos recibidos se escriben en el contenido del string, y la longitud actual del string se ajusta de forma correspondiente.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Receive_P2P Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
LENGTH	OUT	UInt	Word	0	Longitud de la trama recibida en bytes
NDR	OUT	Bool		FALSE	TRUE durante un ciclo cuando hay datos nuevos preparados y la instrucción ha finalizado correctamente.
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Si ERROR está en estado TRUE, puede evaluarse el código de error en la salida STATUS. El valor de STATUS suministra el motivo para finalizar la operación de recepción en el CM. Este suele ser un valor positivo que indica que la operación de recepción ha sido correcta y qué criterio de fin de trama se ha detectado.
Si el valor de STATUS es negativo (se ha activado el bit más significativo del valor hexadecimal), la operación de recepción ha finalizado debido a una condición de error, como error de paridad, framing o desbordamiento.

Cada módulo de comunicaciones puede respaldar un número de tramas específico del módulo. Si hay varias tramas disponibles en el CM, la instrucción Receive_P2P emite la más antigua (FIFO).

Receive_Reset: borrar el búfer de recepción

Descripción

La instrucción Receive_Reset (resetear receptor) borra el búfer de recepción del CM.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Receive_Reset Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo significa que la última solicitud se ha ejecutado correctamente.
ERROR	OUT	Bool		FALSE	TRUE significa que la última solicitud se ha ejecutado con errores. Si esta salida es TRUE, la salida STATUS contiene los códigos de error correspondientes.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Signal_Get: leer estado

Descripción

La instrucción Signal_Get (consultar señales RS232) lee los estados actuales de las señales cualificadoras RS232 y las muestra en las salidas correspondientes de la instrucción.

Nota
Limitación

Esta función solo puede aplicarse en los CM RS232 BA y RS232 HF.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Signal_Get Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
NDR	OUT	Bool		FALSE	TRUE durante un ciclo cuando las señales cualificadoras RS232 se han leído y la instrucción ha finalizado correctamente.
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))
DTR	OUT	Bool		FALSE	Terminal de datos preparado, módulo disponible (salida)
DSR	OUT	Bool		FALSE	Terminal de datos preparado, dispositivo de comunicación preparado (entrada)
RTS	OUT	Bool		FALSE	Solicitud de transmisión, módulo preparado para enviar (salida)
CTS	OUT	Bool		FALSE	Listo para transmitir, el dispositivo de comunicación puede recibir datos (entrada)
DCD	OUT	Bool		FALSE	Señal del soporte de datos detectada, nivel de señal recibido
RING	OUT	Bool		FALSE	Indicador de llamada, aviso de una llamada entrante

Signal_Set: activar señales cualificadoras

Descripción

La instrucción Signal_Set (establecer señales RS232) permite establecer las señales de comunicación RS232.

Esta función solo es válida para el CM RS232.

Nota
Limitaciones

Esta función solo puede aplicarse en los CM RS232 BA y RS232 HF.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
SIGNAL	IN	Byte		0	Selección de la señal que debe establecerse (se admiten varias): • 01H = RTS • 02H = DTR • 04H = DSR (solo válido para el tipo de interfaz DCE)
RTS	IN	Bool		FALSE	Solicitud de transmisión, módulo preparado para enviar Definir este valor en la salida (TRUE o FALSE), valor predeterminado: FALSE
DTR	IN	Bool		FALSE	Terminal de datos preparado, módulo preparado Definir este valor en la salida (TRUE o FALSE), valor predeterminado: FALSE
DSR	IN	Bool		FALSE	Terminal de datos preparado (solo válido para el tipo de interfaz DCE), no utilizado.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Signal_Set Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

.

Get_Features: obtener funciones avanzadas

Descripción

La instrucción Get_Features (obtener funciones avanzadas) permite consultar información relacionada con la capacidad del módulo para soportar CRC y generar avisos de diagnóstico, siempre que el módulo la soporte.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	PORT	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Get_Features Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
NDR	OUT	Bool		FALSE	TRUE durante un ciclo cuando hay datos nuevos preparados y la instrucción ha finalizado correctamente
MODBUS_CRC	OUT	Bool		FALSE	Soporte de Modbus CRC
DIAG_ALARM	OUT	Bool		FALSE	Generación de avisos de diagnóstico
SUPPLY_VOLT	OUT	Bool		FALSE	Diagnóstico disponible para Falta tensión de alimentación L+
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Set_Features: activar funciones avanzadas

Descripción

La instrucción Set_Features (seleccionar funciones avanzadas) permite activar la compatibilidad con CRC y la generación de avisos de diagnóstico, siempre que el módulo la soporte.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	La instrucción para establecer funciones avanzadas se inicia con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
EN_MODBUS_CRC	IN	Bool		FALSE	Activar soporte de Modbus CRC
EN_DIAG_ALARM	IN	Bool		FALSE	Activar generación de avisos de diagnóstico
EN_SUPPLY_VOLT	IN	Bool		FALSE	Activar diagnóstico para Falta tensión de alimentación L +
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Set_Features Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante una ejecución después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5419))

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-

Código de error	Descripción	Solución
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		

Código de error	Descripción	Solución
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)

Código de error	Descripción	Solución
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Transferir datos con Freeport (Página 1027)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)

Código de error	Descripción	Solución
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.

Código de error	Descripción	Solución
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.

Código de error	Descripción	Solución
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		

Código de error	Descripción	Solución
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 [1)]	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 [1)]	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 [1)]	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)

Código de error	Descripción	Solución
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.

Código de error	Descripción	Solución
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.

Código de error	Descripción	Solución
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		

Código de error	Descripción	Solución
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.

Código de error	Descripción	Solución
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.

Código de error	Descripción	Solución
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble

Código de error	Descripción	Solución
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

1) Solo en instrucciones para CPU S7-300/400

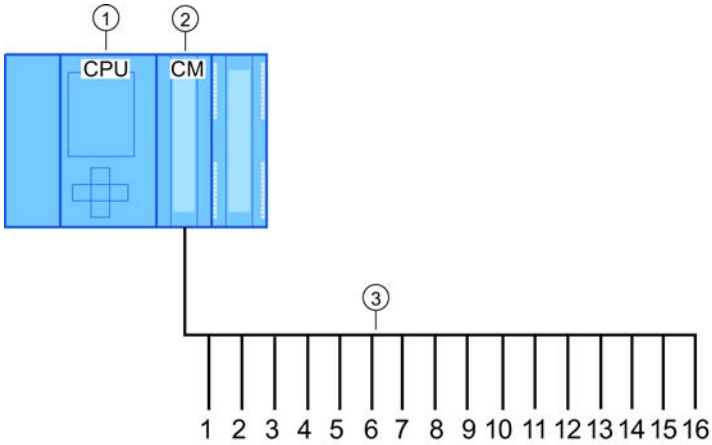
USS

Resumen de la comunicación USS

Comunicación USS

Las instrucciones USS controlan el funcionamiento de accionamientos que soportan el protocolo de la interfaz serie universal (USS). Los módulos de comunicación PtP hacen posible la comunicación con varios accionamientos a través de conexiones RS485 e instrucciones USS. Cada puerto RS485 puede operar un máximo de 16 accionamientos.

El protocolo USS utiliza una red de maestros y esclavos para la comunicación a través de un bus serie. El maestro utiliza un parámetro de dirección para enviar datos a un esclavo concreto. Un esclavo no puede enviar nunca por sí mismo sin haber recibido previamente una solicitud. La comunicación entre los diferentes esclavos no es posible. La comunicación USS funciona en modo semidúplex. La figura siguiente muestra un diagrama de red para un ejemplo de aplicación con 16 accionamientos.



① CPU
② CM
③ Accionamientos USS en una red USS

Figura 9-1 Ejemplo de interconexión con un módulo de comunicación S7-1500

Nota

Comunicación con un accionamiento mediante RS232

En principio, también es posible utilizar CM PtP RS232 BA y CM PtP RS232 HF para la comunicación con un accionamiento. Sin embargo, solo es posible conectar **un** accionamiento a un puerto RS232.

Comunicación con un accionamiento mediante RS422

En principio, también es posible utilizar la interfaz RS422 del CM PtP RS422/485 BA y CM PtP RS422/485 HF para la comunicación con un accionamiento. Sin embargo, solo es posible conectar **un** accionamiento a un puerto RS422.

Instrucciones USS en el programa

- USS_Port_Scan: la instrucción USS_Port_Scan permite la comunicación a través de un módulo de comunicaciones con hasta 16 accionamientos, mediante una red USS (debe ser llamada de forma cíclica).
  En el programa solo hay una instrucción USS_Port_Scan por puerto de comunicación PtP que controla la transferencia a todos los accionamientos.
- USS_Drive_Control: la instrucción USS_Drive_Control permite preparar los datos de USS_Port_Scan para un accionamiento y mostrar los datos recibidos.
  USS_Drive_Control configura los datos que deben enviarse y evalúa los datos recibidos por USS_Port_Scan en una solicitud anterior.
- USS_Read_Param: la instrucción USS_Read_Param permite leer parámetros de un accionamiento.
- USS_Write_Param: la instrucción USS_Write_Param permite modificar los parámetros de un accionamiento.

Requisitos para utilizar el protocolo USS

Las cuatro instrucciones USS utilizan 2 FB y 2 FC para soportar el protocolo USS. Para cada red USS se emplea un bloque de datos de instancia para USS_Port_Scan y un bloque de datos de instancia común para todas las llamadas de USS_Drive_Conrol.

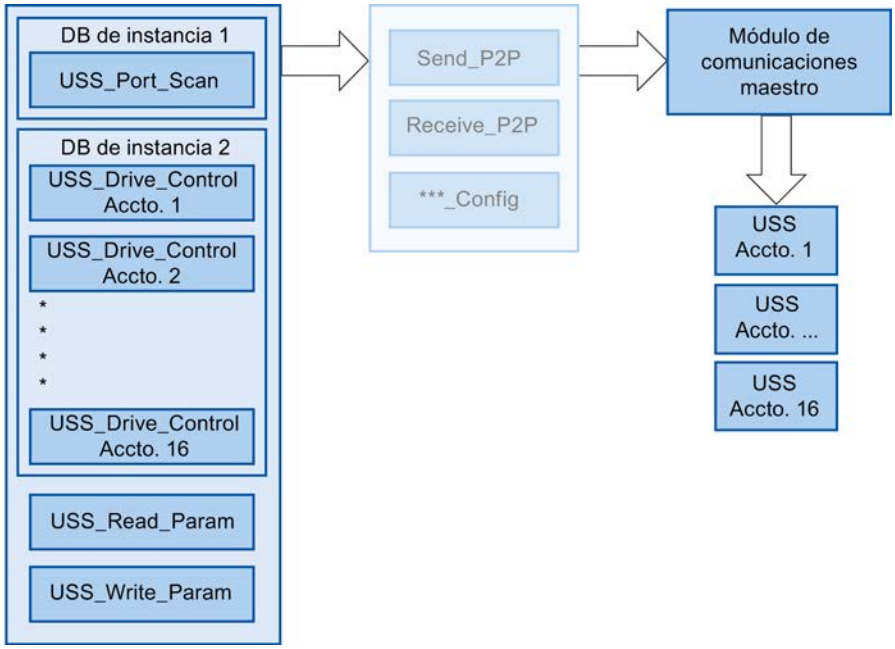


Figura 9-2 Ejecución del programa - USS

Todos los accionamientos (máx. 16) que están conectados a un puerto RS485 forman parte de la misma red USS. Todos los accionamientos que están conectados a otro puerto RS485 forman parte de otra red USS. Cada red USS se gestiona por medio de un bloque de datos de instancia único para todas las instrucciones USS_Drive_Control y otro bloque de datos de instancia para la instrucción USS_Port_Scan. Todas las instrucciones que forman parte de una red USS deben utilizar conjuntamente el bloque de datos de instancia para

USS_Drive_Conrol. Para ello, las instrucciones USS_Port_Scan, USS_Read_Param y USS_Write_Param cuentan con el parámetro USS_DB, que debe interconectarse con el parámetro (estático) USS_DB del DB de instancia de la instrucción USS_Drive_Control.

- Las instrucciones USS_Drive_Control y USS_Port_Scan son bloques de función (FB). Si se inserta la instrucción USS_Drive_Control o USS_Port_Scan en el editor de programación, deberá asignarse un DB a este FB en el cuadro de diálogo "Opciones de llamada". Si se trata de la primera instrucción USS_Drive_Control del programa para esta red USS, existe la posibilidad de aplicar la asignación estándar de DB (o cambiar el nombre en caso necesario) y el nuevo DB se creará automáticamente. Sin embargo, si no se trata de la primera instrucción USS_Drive_Control para este accionamiento, es necesario seleccionar en la lista desplegable del cuadro de diálogo "Opciones de llamada" el DB que se ha asignado previamente a esta red USS.
- Las instrucciones USS_Port_Scan y USS_Read_Param son funciones (FC). Si se insertan estas FC en el editor, no se asignará ningún DB. Si se insertan estas FC o la instrucción USS_Port_Scan en el editor, deberá asignarse a la entrada USS_DB de estas instrucciones el parámetro USS_DB del DB de instancia correspondiente de USS_Drive_Control. Haga doble clic en el campo de parámetros y pulse seguidamente el icono para visualizar los DB disponibles. Introduzca un punto "." y seleccione el parámetro USS_DB en la lista desplegable.
- La función USS_Port_Scan controla la comunicación entre la CPU y los accionamientos a través del puerto de comunicación punto a punto RS485. Cada vez que se llama esta función se procesa una comunicación con un accionamiento. El programa debe llamar a esta función con la suficiente rapidez como para que los accionamientos no notifiquen un tiempo excedido. Para garantizar un comportamiento temporal constante de la transmisión de tramas, esta instrucción debe llamarse con un OB de alarma cíclica.
- La instrucción USS_Drive_Control proporciona al programa acceso a un accionamiento concreto de la red USS. Sus entradas y salidas corresponden a los estados y las funciones de control del accionamiento. Si hay 16 accionamientos en la red, USS_Drive_Control debe llamarse 16 veces en el programa, es decir, una vez por accionamiento.
  La instrucción USS_Drive_Control solo debería llamarse desde un OB cíclico.
- Con las funciones USS_Read_Param y USS_Write_Param se leen y escriben los parámetros operativos del accionamiento. Dichos parámetros controlan el funcionamiento interno del accionamiento. Encontrará una definición de estos parámetros en el manual del accionamiento. El programa puede contener un número cualquiera de estas funciones, aunque únicamente puede estar activa una solicitud de lectura o escritura para un accionamiento. Las funciones USS_Read_Param y USS_Write_Param solo deben llamarse desde el OB de ciclo de un programa principal.

ATENCIÓN
Llamadas de las instrucciones USS Llame USS_Drive_Control, USS_Read_Param y USS_Write_Param únicamente desde un OB de ciclo del programa principal. La instrucción USS_Port_Scan puede llamarse desde cualquier OB, aunque normalmente se llama desde un OB de alarma cíclica. No utilice las instrucciones USS_Drive_Control, USS_Read_Param y USS_Write_Param en un OB con una prioridad más alta que la instrucción USS_Port_Scan correspondiente. No inserte USS_Port_Scan en el programa principal ni USS_Read_Param en un OB de alarma cíclica, por ejemplo. Si la ejecución de USS_Port_Scan es interrumpida por otra instrucción, pueden producirse errores imprevistos.

Calcular el tiempo para la comunicación con el accionamiento

La comunicación con el accionamiento se lleva a cabo de forma asíncrona al ciclo de la CPU. La CPU suele ejecutar varios ciclos antes de que finalice la comunicación con un accionamiento.

Para que la vigilancia de tiempo parametrizada durante el accionamiento no responda, las tramas de transmisión deben enviarse al accionamiento dentro de esta vigilancia de tiempo. También debe incluirse el número de intentos si se requieren varios para completar la transacción debido a errores de comunicación. De forma predeterminada, para el protocolo USS se realizan hasta 2 intentos en cada transacción.
El intervalo entre dos tramas de transmisión se calcula del siguiente modo:

(N * 5 * tiempo de ciclo + 2 * tiempo de ejecución de tramas) * número de intentos + (timeout de la trama de recepción) * (número de intentos - 1)

N	Número de accionamientos en esta red
Factor 5	Para la transmisión y recepción de tramas se requieren 5 ciclos.
Tiempo de ciclo	Tiempo de ciclo máx. de los OB de alarma cíclica en los que se llama la instrucción USS_Port_Scan.
2 * tiempo de ejecución de tramas	Una vez para cada transmisión y una para cada recepción
Tiempo de ejecución de tramas	Tiempo de ejecución de tramas = (número de caracteres por trama) * (11 Bit por carácter) / (velocidad de transferencia en Bit/s)
Número de intentos	3

Para el "Timeout de la trama de recepción" se aplican los siguientes tiempos (ms)

Bits/s	115200	57600	38400	19200	9600	4800	2400	1200
Receive_Conditions.END.RCVTIME	25	29	33	56	72	100	100	100
Receive_Conditions.END.MSGTIME	25	29	33	56	72	124	240	460

Ejemplo:
5 accionamientos

Velocidad de transferencia = 9600 bits/s
28 caracteres por trama
Tiempo de ciclo = 0,020 s
Intervalo = (5 * 5 * 0,02 s + 2 * 28 * 11 / 1200 s) * 3 = 1,69 s
En este caso, la vigilancia de tiempo del accionamiento debe ajustarse aprox. a 2 segundos.

USS_Port_Scan: comunicación vía red USS

Descripción

La instrucción USS_Port_Scan procesa la comunicación a través de una red USS. STEP 7 crea automáticamente un DB de instancia cuando se inserta la instrucción.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
PORT	IN	Port	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/ WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BAUD	IN	DInt		9600	Velocidad de transferencia de datos para la comunicación USS Se admiten: • 1200 Bit/s • 2400 Bit/s • 4800 Bit/s • 9600 Bit/s • 19200 Bit/s • 38400 Bit/s • 57600 Bit/s • 115200 Bit/s
USS_DB	INOUT	USS_BASE		–	El parámetro USS_DB debe interconectarse con el parámetro (estático) USS_DB del DB de instancia, que se crea e inicializa si se inserta una instrucción USS_Drive_Control en el programa.
COM_RST	INOUT	---	Bool	FALSE	Inicialización de la instrucción USS_Port_Scan Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
ERROR	OUT	Bool		FALSE	Si es TRUE, la salida indica que se ha producido un error y la salida STATUS es válida.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 5449)).

Solo hay una instrucción USS_Port_Scan por puerto de comunicación PtP en el programa y cada llamada de dicha instrucción controla una transferencia hacia o desde todos los accionamientos de esta red. Todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo DB de instancia.

El programa debe ejecutar la instrucción USS_Port_Scan tantas veces como sea necesario para que no se produzca ningún timeout en el accionamiento (consulte Requisitos para utilizar el protocolo USS (Página 5435) la sección "Calcular el tiempo para la comunicación con el accionamiento").

La instrucción USS_Port_Scan se llama normalmente desde un OB de alarma cíclica con el fin de impedir timeouts del accionamiento y tener disponibles las últimas actualizaciones de datos USS para llamadas de USS_Drive_Control .

Variables de bloques de datos de USS_Port_Scan

La tabla siguiente muestra las variables estáticas públicas del DB de instancia de USS_Port_Scan que pueden utilizarse en el programa.

Tabla 9-138 Variables estáticas en el DB de instancia

Variable	Tipo de datos	Estándar	Descripción
MODE	USInt	4	Modo de operación Los modos de operación admitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) modo a cuatro hilos (punto a punto) • 2 = dúplex (RS422) modo a cuatro hilos (maestro multipunto, CM PtP (ET 200SP)) • 3 = dúplex (RS422) modo a cuatro hilos (esclavo multipunto, CM PtP (ET 200SP)) • 4 = semidúplex (RS485) modo a dos hilos
LINE_PRE	USInt	2	Estado inicial de la línea de recepción Los estados iniciales admitidos son: • 0 = "sin" estado inicial • 1 = señal R(A) = 5 V, señal R(B) = 0 V (detección Break): en este estado inicial es posible una detección de rotura (Break). Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • 2 = señal R(A) = 0 V, señal R(B) = 5 V: este estado inicial corresponde al estado de reposo (ninguna transmisión activa). Con este estado inicial no es posible la detección de rotura (Break).

Variable	Tipo de datos	Estándar	Descripción
BRK_DET	USInt	0	Activar aviso de diagnóstico: • 0 - no activado • 1 - activado
RETRIES_MAX	SInt/Byte	2	Número de intentos cuando se producen errores de comunicación. Este parámetro permite establecer el número de intentos de envío de una trama de solicitud en el caso de que la trama de respuesta no se reciba dentro del tiempo ajustado.
EN_DIAG_ALARM	Bool	0	Activar aviso de diagnóstico: • 0 - no activado • 1 - activado
EN_SUPPLY_VOLT	Bool	0	Activar diagnóstico para Falta tensión de alimentación L+ • 0 - no activado • 1 - activado

USS_Drive_Control: Preparar y mostrar datos para el accionamiento

Descripción

La instrucción USS_Drive_Control prepara datos para enviarlos al accionamiento y evalúa los datos de respuesta de este. Para cada accionamiento debe utilizarse una instancia propia de la instrucción, y todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo bloque de datos de instancia. Debe introducirse el nombre del DB al insertar la primera instrucción USS_Drive_Control. Seguidamente, hay que remitir al DB que se ha creado al insertar la primera instrucción.

STEP 7 crea automáticamente el DB cuando se inserta la instrucción.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
RUN	IN	Bool		FALSE	Bit de arranque del accionamiento: si este parámetro es TRUE, la entrada hace posible el funcionamiento del accionamiento con el número de revoluciones predeterminado. Si RUN pasa a FALSE mientras el accionamiento está en marcha, el motor se desacelera hasta pararse. Este comportamiento difiere de la desconexión de la alimentación (OFF2) y del frenado del motor (OFF3).
OFF2	IN	Bool		FALSE	Bit "Decelerar hasta parar": si este parámetro es FALSE, este bit provoca la parada natural del accionamiento sin frenar.
OFF3	IN	Bool		FALSE	Bit de parada rápida: si este parámetro es FALSE, este bit provoca una parada rápida frenando el accionamiento.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
F_ACK	IN	Bool		FALSE	Bit de acuse de error: con este bit se desactiva el bit de error de un accionamiento. El bit se activa tras borrar el error, con lo que el accionamiento detecta que el error anterior no debe seguir notificándose.
DIR	IN	Bool		FALSE	Control de sentido del accionamiento: este bit se activa cuando el accionamiento debe funcionar hacia delante (cuando SPEED_SP es positivo, ver tabla "Interacción de los parámetros SPEED_SP y DIR").
DRIVE	IN	USInt	Byte	1	Dirección del accionamiento: esta entrada es la dirección del accionamiento USS. El rango válido está entre accionamiento 1 y accionamiento 16.
PZD_LEN	IN	USInt	Byte	2	Longitud de palabra: es el número de palabras de datos PZD. Los valores válidos son 2, 4, 6 u 8 palabras.
SPEED_SP	IN	Real		0.0	Consigna de velocidad: es el número de revoluciones del accionamiento proporcionalmente a la frecuencia configurada. Un valor positivo significa que el accionamiento funciona hacia delante (si DIR es true). El rango válido es de 200,00 a -200,00.
CTRL3	IN	Word		0	Palabra de control 3: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL4	IN	Word		0	Palabra de control 4: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL5	IN	Word		0	Palabra de control 5: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL6	IN	Word		0	Palabra de control 6: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL7	IN	Word		0	Palabra de control 7: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL8	IN	Word		0	Palabra de control 8: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción USS_Drive_Control Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
NDR	OUT	Bool		FALSE	Datos nuevos preparados: si este parámetro es TRUE, el bit notifica que en la salida hay preparados datos de una nueva solicitud de comunicación.
ERROR	OUT	Bool		FALSE	Se ha producido un error: Si es TRUE, indica que se ha producido un error y la salida STATUS es válida. Todas las salidas restantes se ponen a cero en caso de error. Los errores de comunicación solo se notifican en las salidas ERROR y STATUS de la instrucción USS_Port_Scan.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 5449)).

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
RUN_EN	OUT	Bool		FALSE	Servicio habilitado: este bit notifica si el accionamiento está en marcha.
D_DIR	OUT	Bool		FALSE	Sentido del accionamiento: este bit notifica si el accionamiento funciona hacia delante. • FALSE – adelante • TRUE – atrás
INHIBIT	OUT	Bool		FALSE	Accionamiento bloqueado: este bit notifica el estado del bit de bloqueo para el accionamiento. • FALSE – no bloqueado • TRUE – bloqueado
FAULT	OUT	Bool		FALSE	Error en accionamiento: este bit notifica que en el accionamiento se ha producido un error. Hay que eliminar el fallo y activar el bit F_ACK para borrar este bit.
SPEED	OUT	Real		0.0	Valor real de velocidad del accionamiento (valor escalado de STATUS 2 del accionamiento): es la velocidad del accionamiento proporcionalmente a la velocidad configurada.
STATUS1	OUT	Word		0	STATUS 1 del accionamiento Este valor contiene bits de estado fijos de un accionamiento.
STATUS3	OUT	Word		0	STATUS 3 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS4	OUT	Word		0	STATUS 4 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS5	OUT	Word		0	STATUS 5 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS6	OUT	Word		0	STATUS 6 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS7	OUT	Word		0	STATUS 7 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS8	OUT	Word		0	STATUS 8 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.

Cuando tiene lugar la primera ejecución de USS_Drive_Control, se inicializa en el DB de instancia el accionamiento indicado en la dirección USS (parámetro DRIVE). Tras la inicialización, las instrucciones USS_Port_Scan posteriores pueden iniciar la comunicación con el accionamiento en este número.

Si se cambia el número de accionamiento, la CPU debe ponerse primero a STOP y después a RUN de nuevo para que se inicialice el DB de instancia. Los parámetros de entrada se configuran en el búfer de transmisión USS y las salidas, si existen, se leen de un búfer de

respuesta "anterior" válido. USS_Drive_Control configura solo los datos que deben enviarse y evalúa los datos recibidos en una solicitud anterior.

El sentido de giro del accionamiento se controla desde la entrada D_IR (Bool) o mediante el signo (positivo o negativo) de la entrada SPEED_SP (Real). La tabla siguiente explica cómo funcionan estas entradas conjuntamente para especificar el sentido de giro del accionamiento, suponiendo que el motor gira hacia delante.

Tabla 9-139 Interacción de los parámetros SPEED_SP y DIR

SPEED_SP	DIR	Sentido de giro del accionamiento
Valor > 0	0	Atrás
Valor > 0	1	Adelante
Valor < 0	0	Adelante
Valor < 0	1	Atrás

USS_Read_Param: leer datos del accionamiento

Descripción

La instrucción USS_Read_Param lee un parámetro de un accionamiento. Todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el bloque de datos de instancia de la instrucción USS_Drive_Control . USS_Read_Param debe llamarse desde un OB de ciclo del programa principal.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
REQ	IN	Bool		–	Con un flanco ascendente en REQ se genera una nueva solicitud de lectura.
DRIVE	IN	USInt	Byte	–	Dirección del accionamiento: DRIVE es la dirección del accionamiento USS. El rango válido está entre accionamiento 1 y accionamiento 16.
PARAM	IN	UInt		–	Número de parámetro: PARAM indica el parámetro de accionamiento que se escribe. El rango de este parámetro está entre 0 y 2047. En algunos accionamientos, se puede acceder a valores de parámetro mayores que 2047 con el byte más significativo del parámetro INDEX. Encontrará más información sobre el acceso a un rango ampliado en el manual del accionamiento.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
INDEX	IN	UInt		–	Índice de parámetro: INDEX indica el índice de parámetro del accionamiento en el que debe escribirse. Es un valor de 16 bits en el que el byte menos significativo es el valor real del índice, con un rango de (0 a 255). El byte más significativo también puede ser utilizado por el accionamiento y es específico del mismo. Encontrará más información en el manual del accionamiento.
USS_DB	INOUT	USS_BASE		–	El parámetro USS_DB debe interconectarse con el parámetro (estático) USS_DB del DB de instancia, que se crea e inicializa si se inserta una instrucción USS_Drive_Control en el programa.
DONE[1]	OUT	Bool		FALSE	Si este parámetro es TRUE, en la salida VALUE está pendiente el valor previamente solicitado para el parámetro de lectura. Este bit se activa cuando la instrucción USS_Drive_Control detecta la respuesta de lectura del accionamiento. Este bit se inicializará con la siguiente llamada de USS_Read_Param .
ERROR	OUT	Bool		FALSE	ERROR = TRUE: se ha producido un error y la salida STATUS es válida. Todas las salidas restantes se ponen a cero en caso de error. Los errores de comunicación solo se notifican en las salidas ERROR y STATUS de la instrucción USS_Port_Scan.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 5449)).
VALUE	OUT	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Es el valor del parámetro que se ha leído y solo es válido si el bit DONE es true.

[1] El bit DONE indica que se han leído datos válidos del accionamiento de motor referenciado y se han suministrado a la CPU. No hace referencia al hecho de que la instrucción está preparada para leer inmediatamente otro parámetro. Una solicitud de lectura vacía debe enviarse al accionamiento del motor y también debe ser acusada por la instrucción antes de que se habilite el canal de parámetros para que sea utilizado por el accionamiento correspondiente. La llamada inmediata de USS_Read_Param o USS_Write_Param para el accionamiento de motor específico provoca el error 0x818A.

USS_Write_Param: modificar datos del accionamiento

Nota

Para instrucciones de escritura EEPROM (EEPROM en un accionamiento USS) se aplica lo siguiente:

Mantenga en un número reducido las operaciones de escritura EEPROM para garantizar una larga vida útil del EEPROM.

Descripción

La instrucción USS_Write_Param modifica un parámetro del accionamiento. Todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el bloque de datos de instancia del USS_Drive_Control.

USS_Write_Param debe llamarse desde el OB de ciclo de un programa principal.

Parámetros

Tabla 9-140 Tipos de datos de los parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
REQ	IN	Bool		–	Con un flanco ascendente en REQ se genera una nueva solicitud de escritura.
DRIVE	IN	USInt	Byte	–	Dirección del accionamiento: DRIVE es la dirección del accionamiento USS. El rango válido está entre accionamiento 1 y accionamiento 16.
PARAM	IN	UInt		–	Número de parámetro: PARAM indica el parámetro de accionamiento que se escribe. El rango de este parámetro está entre 0 y 2047. En algunos accionamientos, se puede acceder a valores de parámetro mayores que 2047 con el byte más significativo del parámetro INDEX. Encontrará más información sobre el acceso a un rango ampliado en el manual del accionamiento.
INDEX	IN	UInt		–	Índice de parámetro: INDEX indica el índice de parámetro del accionamiento en el que debe escribirse. Es un valor de 16 bits en el que el byte menos significativo es el valor real del índice, con un rango de (0 a 255). El byte más significativo también puede ser utilizado por el accionamiento y es específico del mismo. Encontrará más información en el manual del accionamiento.
EEPROM	IN	Bool		–	Guardar en el EEPROM del accionamiento: si es TRUE, la transacción de un parámetro para escribir en el accionamiento se guarda en el EEPROM del accionamiento. Si es FALSE, el valor escrito solo se guarda temporalmente y se perderá la próxima vez que se conecte el accionamiento.
VALUE	IN	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Valor del parámetro en el que debe escribirse. Debe ser válido con un flanco ascendente de REQ.
USS_DB	INOUT	USS_BASE		–	El parámetro USS_DB debe interconectarse con el parámetro (estático) USS_DB del DB de instancia, que se crea e inicializa si se inserta una instrucción USS_Drive_Control en el programa.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
DONE[1]	OUT	Bool		FALSE	Si es TRUE, la entrada VALUE se ha escrito en el accionamiento. Este bit se activa cuando la instrucción USS_Drive_Control detecta la respuesta de escritura del accionamiento. Este bit se inicializará con la siguiente llamada de USS_Write_Param.
ERROR	OUT	Bool		FALSE	Si es TRUE, se ha producido un error y la salida STATUS es válida. Todas las salidas restantes se ponen a cero en caso de error. Los errores de comunicación solo se notifican en las salidas ERROR y STATUS de la instrucción USS_Port_Scan .
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 5449)).

[1] El bit DONE indica que se han leído datos válidos del accionamiento de motor referenciado y se han suministrado a la CPU. No hace referencia al hecho de que la librería USS está preparada para leer inmediatamente otro parámetro. Una solicitud de escritura vacía debe enviarse al accionamiento del motor y también debe ser acusada por la instrucción antes de que se habilite el canal de parámetros para que sea utilizado por el accionamiento correspondiente. La llamada inmediata de USS_Read_Param o USS_Write_Param FC para el accionamiento de motor específico provoca el error 0x818A.

Información general sobre el dispositivo de accionamiento

Requisitos para la configuración del accionamiento

- Para los accionamientos debe configurar el uso de 4 palabras PKW (**P**arameter**k**ennungs**w**ert o valor de identificación de parámetro).
- Configure los accionamientos para 2, 4, 6 u 8 palabras PZD (**P**ro**z**ess**d**atenbereich o rango de datos de proceso).
- Procure que el número de palabras PZD en el accionamiento se corresponda con la entrada PZD_LEN de la instrucción USS_Drive_Control del accionamiento.
- Procure que la velocidad de transferencia de todos los accionamientos se corresponda con la entrada BAUD de la instrucción USS_Port_Scan.
- Procure que el accionamiento esté configurado para la comunicación USS.
- Asegúrese de que se haya determinado en el accionamiento que la consigna de frecuencia se proporcione a través de la interfaz USS.
- Asegúrese de que la dirección del accionamiento esté definida (rangos: 1-16). Esta dirección debe corresponderse con la entrada DRIVE en el bloque USS_Drive_Control del accionamiento.
- Asegúrese de que la red RS485 esté correctamente cerrada.

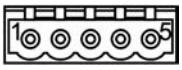
Conexión y configuración del accionamiento SINAMICS V20

Un ejemplo de aplicación para el funcionamiento de un SINAMICS V20 en una S7-1200 se puede consultar en Internet. (http://support.automation.siemens.com/WW/view/es/63696870)

Conexión del accionamiento SINAMICS V20

Ejemplo de conexión de un accionamiento SIEMENS G120(C) a una red USS. Encontrará ejemplos de conexión para otros accionamientos en el manual del correspondiente accionamiento.

La conexión de un accionamiento SINAMICS G120(C) a la red USS se realiza a través de un conector. La conexión es resistente a cortocircuitos y está aislada galvánicamente.



1	0 V potencial de referencia
2	RS485N, recepción y transmisión (-)
3	RS485N, recepción y transmisión (+)
4	Pantalla del cable
5	No utilizado

Figura 9-3 Conexión USS

ATENCIÓN
Diferentes tensiones de referencia Si se conectan entre sí dispositivos que no tienen la misma tensión de referencia, pueden producirse intensidades no deseadas en el cable de conexión. Estas intensidades no deseadas pueden provocar errores de comunicación o daños materiales en los dispositivos. Asegúrese de que todos los dispositivos que estén conectados entre sí a través de un cable de comunicación tengan el mismo conductor de referencia en el circuito o bien estén separados eléctricamente para que no se produzcan intensidades no deseadas. Asegúrese de que la pantalla esté conectada a masa o al pin 1 del conector de bus del accionamiento. Asegúrese de que el borne de cableado 2 (GND) del G120(C) esté conectado a masa.

En caso de que el maestro RS485 (p. ej., una CPU S7-1200 con módulo de comunicaciones CM1241) esté conectado a través de un conector PROFIBUS, los cables de bus deben tenderse como sigue:

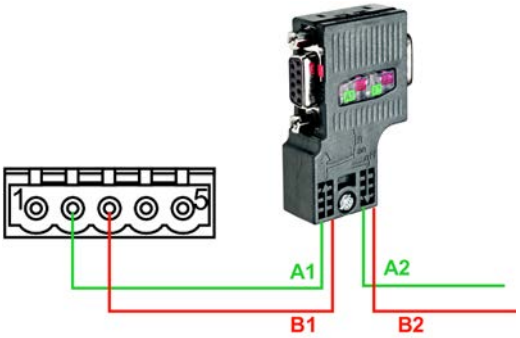


Figura 9-4 Conexión del módulo de comunicaciones

Si el maestro RS485 es un dispositivo terminador de red o si se tiene una conexión punto a punto, deben utilizarse los bornes A1 y B1 (no A2 y B2) del conector PROFIBUS, porque dichos bornes permiten los ajustes de terminación (p. ej. en el conector DP 6ES7972-0BB52-0XA0).

En caso de que el G120(C) esté configurado como dispositivo terminador de red, debe ajustar el interruptor para la resistencia terminadora de bus en "ON".

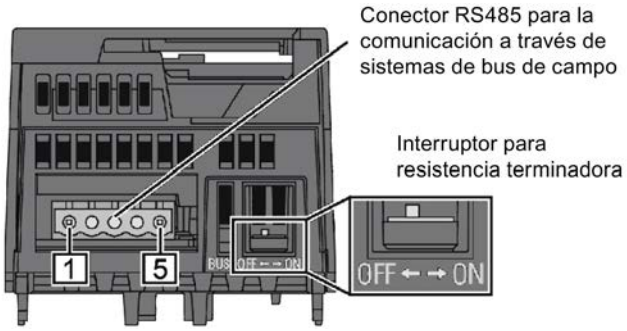


Figura 9-5 Conexión del dispositivo terminador

Configuración de un accionamiento G120(C)

Antes de conectar un accionamiento al S7-1500 o al ET 200SP, hay que asegurarse de que se dispone de los siguientes parámetros de sistema.

Paso	Instrucción	Instrucciones de servicio	
		G120 [1]	G120C [2]
1	Efectúe una puesta en marcha básica del accionamiento con el panel de operador BOP-2. El convertidor ofrece diferentes asignaciones predefinidas (macros) para sus entradas y salidas y la interfaz del bus de campo. Elija en el 9.º paso de la puesta en marcha básica (MAC PAR p15) la macro 21 para la comunicación USS. Con ello estarán también predefinidos los siguientes parámetros: • Velocidad de transferencia (p2020): 38400 bits/s • Número de PZD (p2022): 2 • Número de PKW (p2023): variable Nota: La puesta en marcha básica también se puede ejecutar con el software de puesta en marcha STARTER o con SINAMICS Startdrive.	Cap. 4.4.3	Cap. 6.4.1
2	Defina la dirección de USS del convertidor a través del interruptor de dirección de la Control Unit del G120 o del G120(C). • Área de direcciones válida: 1 … 30 Nota: La dirección USS también se puede ajustar a través del parámetro p2021 o a través de STARTER o de SINAMICS Startdrive.	Cap. 6.2.2.1	Cap. 8.4.2.1
	En los pasos siguientes, con el BOP-2 se accede directamente a parámetros seleccionando sus números y modificando sus valores.	Cap. 4.4.2	Cap. 6.4.2

Paso	Instrucción	Instrucciones de servicio	
		G120 [1]	G120C [2]
3	Adapte los siguientes parámetros de convertidor de su aplicación relativos a la comunicación: • Velocidad de transferencia (p2020), siempre que sea ≠ 38400 bits/s (Preste atención a la concordancia con el parámetro BAUD de la instrucción de comunicación USS_Port_Scan.) • Número de PZD (p2022), siempre que sea ≠ 2 (Preste atención a la concordancia con el parámetro PZD_LEN de la instrucción de comunicación USS_Drive_Control.) • Número de PKW (p2023) = 4 (Modifique a "4" el valor ajustado por defecto a través de la macro 21 a "variable" (127) (las instrucciones USS_Read_Param y USS_Write_Param lo requieren.) • Bus de campo SS Tiempo de vigilancia [ms] (p2040)	Cap. 6.2.2.2	Cap. 8.4.2.1
4	Defina el origen para la consigna de velocidad. • Selec n_cons (p1000[0]) = 6 (La consigna de velocidad se proporciona a través del bus USS.)		
5	Ajuste la magnitud de referencia para la velocidad y la frecuencia. • n_ref f_ref (p2000) = (6.00 min^{-1} hasta 210000.00 min^{-1}) (Todas las velocidades o frecuencias indicadas como relativas se refieren a esta magnitud de referencia. La magnitud de referencia corresponde al 100% o a 4000_{hex} (Palabra) o 4000 0000_{hex} (Palabra doble). En ese caso: $f_{magnitud\ ref}$ (en Hz) = $n_{magnitud\ ref}$ (en ((min^{-1}) / 60) x número de pares de polos))		
6	Transfiera los parámetros a la memoria no volátil. • Guardar par (p0971) = 1		

1) G120 (http://support.automation.siemens.com/CN/view/es/62089662)

2) G120(C) (http://support.automation.siemens.com/WW/view/es/61462568)

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-

Código de error	Descripción	Solución
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).

Código de error	Descripción	Solución
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).

Código de error	Descripción	Solución
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Recibir datos con Freeport (Página 1028)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)

Código de error	Descripción	Solución
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.

Código de error	Descripción	Solución
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		

Código de error	Descripción	Solución
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

Código de error	Descripción	Solución
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 [1)]	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 [1)]	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 [1)]	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255

Código de error	Descripción	Solución
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

1) Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.

Código de error	Descripción	Solución
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información

Código de error	Descripción	Solución
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.

Código de error	Descripción	Solución
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1)]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1)]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1)]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1)]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1)]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1)]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG

Código de error	Descripción	Solución
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.

Código de error	Descripción	Solución
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

[1] Solo en instrucciones para CPU S7-300/400

MODBUS (RTU)

Resumen de la comunicación Modbus RTU

Comunicación Modbus RTU

Modbus RTU (Remote Terminal Unit) es un protocolo estándar para la comunicación en la red y utiliza la conexión RS232 o RS422/485 para la transferencia de datos serie entre dispositivos Modbus de la red.

Modbus RTU utiliza una red de maestro/esclavos en la que toda la comunicación sale de un solo dispositivo maestro, mientras que los esclavos simplemente reaccionan a la solicitud del maestro. El maestro envía una solicitud a una dirección de esclavo y solo responde al comando el esclavo con dicha dirección de esclavo.

Caso especial: la dirección de esclavo Modbus 0 envía una trama Broadcast (difusión) a todos los esclavos (sin respuesta del esclavo).

Códigos de función Modbus

- Una CPU que se utiliza como maestro Modbus RTU puede leer y escribir datos y estados E/S en un esclavo Modbus RTU conectado por medio de una conexión de comunicación.
- Una CPU que se utiliza como esclavo Modbus RTU permite a un maestro Modbus RTU conectado por medio de una conexión de comunicación leer y escribir datos y estados E/S en la propia CPU.

Tabla 9-141 Funciones para leer datos: leer datos de programa y E/S descentralizadas

Código de función Modbus	Funciones para lectura de datos del esclavo (servidor) - direccionamiento estándar
01	Leer bits de salida: 1 hasta 2000/1992[1)] bits por solicitud
02	Leer bits de entrada: 1 hasta 2000/1992[1)] bits por solicitud
03	Leer registro de paradas: 1 hasta 125/124[1)] palabras por solicitud
04	Leer palabras de entrada: 1 hasta 125/124[1)] palabras por solicitud

1) Con direccionamiento avanzado

Tabla 9-142 Funciones para escribir datos: escribir E/S descentralizadas y modificar datos de programa

Código de función Modbus	Funciones para escritura de datos en el esclavo (servidor) - direccionamiento estándar
05	Escribir un bit de salida: 1 bit por solicitud
06	Escribir un registro de paradas: 1 palabra por solicitud
15	Escribir uno o varios bits de salida: 1 a 1960 bits por solicitud
16	Escribir uno o varios registros de paradas: 1 a 122 palabras por solicitud

- Los códigos de función Modbus 08 y 11 ofrecen posibilidades de diagnóstico para la comunicación con el dispositivo esclavo.
- La dirección de esclavo Modbus 0 envía una trama Broadcast (de difusión) a todos los esclavos (sin respuesta del esclavo; para códigos de función 5, 6, 15, 16).

Tabla 9-143 Direcciones de estación en la red Modbus

Estación		Dirección
Equipo RTU	Dirección de estación estándar	1 a 247 , así como 0 para Broadcast
	Dirección de estación avanzada	1 a 65535, así como 0 para Broadcast

Direcciones de memoria Modbus

El número de direcciones de memoria Modbus (direcciones de entrada/salida) realmente disponibles se rige por la variante de CPU y la memoria de trabajo disponible.

Instrucciones Modbus RTU en el programa

- Modbus_Comm_Load: es necesario ejecutar Modbus_Comm_Load para establecer parámetros PtP como velocidad de transferencia de datos, paridad y control del flujo de datos. Una vez se ha configurado el módulo de comunicación para el protocolo Modbus RTU, solo pueden utilizarlo la instrucción Modbus_Master o la instrucción Modbus_Slave.
- Modbus_Master: con la instrucción de maestro Modbus es posible utilizar la CPU como dispositivo maestro Modbus RTU para la comunicación con uno o varios dispositivos esclavos Modbus.
- Modbus_Slave: con la instrucción de esclavo Modbus es posible utilizar la CPU como dispositivo esclavo Modbus RTU para la comunicación con un dispositivo maestro Modbus.

Modbus_Comm_Load: configurar un módulo de comunicaciones para Modbus

Descripción

La instrucción Modbus_Comm_Load configura un módulo de comunicaciones para la comunicación a través del protocolo Modbus RTU. Si se inserta la instrucción Modbus_Comm_Load en el programa, se asigna automáticamente un bloque de datos de instancia.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	Port	Laddr	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BAUD	IN	UDInt	DWord	9600	Selección de la velocidad de transferencia de datos Los valores admitidos son: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 bit/s.
PARITY	IN	UInt	Word	1	Selección de la paridad: • 0 – ninguna • 1 – impar • 2 – par
FLOW_CTRL	IN	UInt	Word	0	Selección del control del flujo: • 0 – (estándar) no hay control del flujo • 1 – control del flujo de hardware con RTS siempre ON (no es válido para CM RS422/485) • 2 – control del flujo de hardware con RTS con protocolo de intercambio (no es válido para CM RS422/485)
RTS_ON_DLY	IN	UInt	Word	0	Selección del retardo a la conexión RTS: • 0 – no hay retardo de "RTS activo" hasta que se envía el primer carácter de la trama. • 1 a 65535 – retardo en milisegundos de "RTS activo" hasta que se envía el primer carácter de la trama (no es válido para CM RS422/485). Los retardos RTS deben aplicarse independientemente de la selección FLOW_CTRL.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
RTS_OFF_DLY	IN	UInt	Word	0	Selección del retardo a la desconexión RTS: • 0 – no hay retardo tras la transferencia del último carácter hasta "RTS inactivo" • 1 a 65535 – retardo en milisegundos desde la transferencia del último carácter hasta "RTS inactivo" (no es válido para puertos RS422/485). Los retardos RTS deben aplicarse independientemente de la selección FLOW_CTRL.
RESP_TO	IN	UInt	Word	1000	Timeout de respuesta: 5 ms a 65535 ms - tiempo en milisegundos que Modbus_Master espera una respuesta del esclavo. Si el esclavo no responde durante este período, Modbus_Master repite la solicitud o la finaliza con un error una vez que se ha enviado el número indicado de repeticiones (ver abajo, parámetro RETRIES).
MB_DB	IN/OUT	MB_BASE		-	Una referencia al bloque de datos de instancia de las instrucciones Modbus_Master o Modbus_Slave. El parámetro MB_DB debe interconectarse con el parámetro MB_DB (estático y, por tanto, no visible en la instrucción) de la instrucción Modbus_Master o Modbus_Slave .
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Modbus_Comm_Load Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	El bit DONE es TRUE durante un ciclo después de que se haya ejecutado correctamente la última solicitud.
ERROR	OUT	Bool		FALSE	El bit ERROR es TRUE durante un ciclo después de que la última solicitud haya finalizado con errores. El código de error del parámetro STATUS solo es válido en el ciclo en que ERROR = TRUE.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 5485))

Modbus_Comm_Load se ejecuta para configurar un puerto para el protocolo Modbus RTU. Una vez se ha configurado un puerto para el protocolo Modbus RTU, solo pueden utilizarlo la instrucción Modbus_Master o Modbus_Slave.

Para la configuración de cada puerto de comunicación utilizado para la comunicación Modbus debe utilizarse una ejecución de Modbus_Comm_Load. Hay que asignar un DB de instancia Modbus_Comm_Load unívoco a cada puerto utilizado. Posteriormente, solo debe volver a ejecutarse Modbus_Comm_Load cuando deban modificarse parámetros de comunicación como velocidad de transferencia de datos o paridad, así como tras restablecerse la red, en caso necesario.

Si se inserta, p. ej., Modbus_Master o Modbus_Slave en el programa, se asigna automáticamente un bloque de datos de instancia a la instrucción. Debe interconectar el parámetro MB_DB de la instrucción Modbus_Comm_Load con el parámetro MB_DB de la instrucción Modbus_Master o Modbus_Slave.

Variables de bloques de datos de Modbus_Comm_Load

La tabla siguiente muestra las variables estáticas públicas del DB de instancia de Modbus_Comm_Load que pueden utilizarse en el programa.

Tabla 9-144 Variables estáticas en el DB de instancia

Variable	Tipo de datos		Estándar	Descripción
	S7-1500	S7-300/400/WinAC		
ICHAR_GAP	Word		0	Tiempo de retardo de caracteres máximo entre caracteres. Este parámetro se indica en milisegundos y sirve para aumentar el período previsto entre los caracteres recibidos. El número correspondiente de tiempos de bit para este parámetro se suma al valor estándar Modbus de 35 tiempos de bit (3,5 tiempos de carácter).
RETRIES	Word		2	Número de intentos repetidos que lleva a cabo el maestro antes de devolver el código de error 0x80C8 para "sin respuesta".
EN_SUPPLY_VOLT	Bool		0	Activar diagnóstico para Falta tensión de alimentación L+
MODE	USInt	Byte	0	Modo de operación Los modos de operación admitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) modo a cuatro hilos (punto a punto) • 2 = dúplex (RS422) modo a cuatro hilos (maestro multipunto, CM PtP (ET 200SP)) • 3 = dúplex (RS422) modo a cuatro hilos (esclavo multipunto, CM PtP (ET 200SP)) • 4 = semidúplex (RS485) modo a dos hilos
LINE_PRE	USInt	Byte	0	Estado inicial de la línea de recepción Los estados iniciales admitidos son: • 0 = "sin" estado inicial • 1 = señal R(A) = 5 V, señal R(B) = 0 V (detección Break): en este estado inicial es posible una detección de rotura (Break). Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • 2 = señal R(A) = 0 V, señal R(B) = 5 V: este estado inicial corresponde al estado de reposo (ninguna transmisión activa). Con este estado inicial no es posible la detección de rotura (Break).
CABLE_BRK	USInt	Byte	0	Detección Break Se admiten: • 0 = detección Break desactivada • 1 = detección Break activada
EN_DIAG_ALARM	Bool		0	Activar aviso de diagnóstico: • 0 - no activado • 1 - activado

Modbus_Master: comunicarse como maestro Modbus

Descripción

La instrucción Modbus_Master se comunica en calidad de maestro Modbus a través de un puerto configurado con la instrucción Modbus_Comm_Load. Si se inserta la instrucción Modbus_Master en el programa, se asigna automáticamente un bloque de datos instancia. El parámetro MB_DB de la instrucción Modbus_Comm_Load debe interconectarse con el parámetro MB_DB (estático) de la instrucción Modbus_Master .

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
REQ	IN	Bool		FALSE	FALSE = no hay solicitud TRUE = solicitud de enviar datos al esclavo Modbus
MB_ADDR	IN	UInt	Word	-	Dirección de estación Modbus RTU: Área de direccionamiento estándar (1 a 247 y 0 para Broadcast) Área de direccionamiento ampliada (1 a 65535 y 0 para Broadcast) El valor 0 está reservado para el Broadcast (difusión) de una trama a todos los esclavos Modbus. Para el Broadcast (difusión) solo se soportan los códigos de función Modbus 05, 06, 15 y 16.
MODE	IN	USInt	Byte	0	Selección de modo: indica el tipo de solicitud (lectura, escritura o diagnóstico). Encontrará más información en la tabla de las funciones Modbus más adelante.
DATA_ADDR	IN	UDInt	DWord	0	Dirección inicial en el esclavo: indica la dirección inicial de los datos a los que debe accederse en el esclavo Modbus. Encontrará las direcciones válidas en la tabla de las funciones Modbus más adelante.
DATA_LEN	IN	UInt	Word	0	Longitud de datos: indica el número de bits o palabras a los que debe acceder esta solicitud. Encontrará las longitudes válidas en la tabla de las funciones Modbus más adelante.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Modbus_Master Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DATA_PTR	IN/OUT	Variant	Any	-	Puntero de datos: apunta a la dirección de marca o DB de los datos que deben escribirse o leerse.
DONE	OUT	Bool		FALSE	El bit DONE es TRUE durante un ciclo después de que se haya ejecutado correctamente la última solicitud.
BUSY	OUT	Bool		-	• FALSE – no hay peticiones activas para Modbus_Master • TRUE – se está procesando una petición para Modbus_Master

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
ERROR	OUT	Bool		FALSE	El bit ERROR es TRUE durante un ciclo después de que la última solicitud haya finalizado con errores. El código de error del parámetro STATUS solo es válido en el ciclo en que ERROR = TRUE.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 5485))

Variables en el bloque de datos del maestro Modbus

La tabla siguiente muestra las variables estáticas públicas del DB de instancia de Modbus_Master que pueden utilizarse en el programa.

Tabla 9-145 Variables estáticas en el DB de instancia

Variable	Tipo de datos	Estándar	Descripción
Blocked_Proc_Timeout	Real	0.5	Tiempo (en segundos) que debe esperarse una instancia bloqueada del maestro Modbus antes de que esta se elimine como ACTIVA. Esto puede suceder, por ejemplo, cuando se ha emitido una solicitud de maestro y, seguidamente, el programa deja de llamar la función del maestro antes de que la solicitud haya finalizado por completo. El valor de tiempo debe ser mayor que 0 y menor que 55 segundos, de lo contrario se produce un error. Ver también los apartados "Reglas para la comunicación del Modbus-Master" y "Llamada de la instrucción Modbus_Master con diferentes ajustes de parámetros".
Extended_Addressing	Bool	FALSE	Configura la dirección de estación del esclavo como byte único o doble. • FALSE = dirección de un byte; 0 a 247 • TRUE = dirección de dos bytes (corresponde al direccionamiento avanzado); 0 a 65535

Variable	Tipo de datos	Estándar	Descripción
Compatibility_Mode [1]	Bool	FALSE	Modo de compatibilidad con CP 341 y CP 441-2 con driver para Modbus RTU y con ET 200SP 1SI para Modbus. El valor predeterminado es 0. • FALSE = según especificación Modbus, no compatible • TRUE = compatible – Con FC1 y FC2 se aplica lo siguiente: Los datos leídos en la trama de recepción se escriben palabra por palabra en la memoria direccionada de la CPU y se sustituyen byte por byte. Si el número de bits que deben transmitirse no es múltiplo de 16, los bits no relevantes de la última palabra se ponen a cero. – Con FC15 se aplica lo siguiente: Las palabras que deben transmitirse se leen una a una en la memoria direccionada y se escriben byte por byte en la trama de transmisión. Si el número de bits que deben transmitirse no es múltiplo de 8, los bits no relevantes del último byte se leen sin cambios en la memoria direccionada y se insertan en la trama de transmisión.
MB_DB	MB_BASE	-	El parámetro MB_DB de la instrucción Modbus_Comm_Load debe conectarse con el parámetro MB_DB de la instrucción Modbus_Master.

[1] Los módulos de comunicaciones PtP se comportan según la especificación Modbus. Para que Modbus se comporte como en CP 341, CP 441-2 y ET 200SP 1SI, utilice el parámetro "Compatibility_Mode".

El programa puede escribir valores en las variables Blocked_Proc_Timeout y Extended_Addressing para controlar las operaciones del maestro Modbus.

Reglas para la comunicación del Modbus-Master

- Modbus_Comm_Load debe ejecutarse para configurar un puerto con el fin de que la instrucción Modbus_Master pueda comunicarse con dicho puerto.
- Si debe utilizarse un puerto como Modbus Master, Modbus _Slave no debe utilizar este puerto. Con este puerto pueden utilizarse una o varias instancias de Modbus_Master [1]. Sin embargo, todas las ejecuciones de Modbus_Master deben utilizar el mismo DB de instancia para el puerto.
- Las instrucciones Modbus no utilizan eventos de alarma de la comunicación para controlar el proceso de comunicación. El programa debe consultar la instrucción Modbus_Master si hay peticiones terminadas (DONE, ERROR).
- Se recomienda llamar todas las ejecuciones de Modbus_Master para un puerto concreto desde un OB de ciclo de programa. Las instrucciones de maestro Modbus solo pueden ejecutarse en cada caso en un ciclo de programa o bien en un nivel de procesamiento cíclico/controlado por tiempo. No deben procesarse en distintos niveles de procesamiento. La interrupción de preferencia de una instrucción de maestro Modbus por otra instrucción de maestro Modbus en un nivel de procesamiento con mayor prioridad provoca un funcionamiento incorrecto. Las instrucciones de maestro Modbus no deben procesarse en niveles de arranque, diagnóstico o errores de tiempo.

[1)] El concepto "instancia de maestro Modbus" significa en este caso una llamada de la instrucción Modbus_Master con la misma interconexión que una instrucción Modbus_Comm_Load y el mismo ajuste para los parámetros MB_ADDR, MODE, DATA_ADDR y DATA_LEN.

Ejemplo.

Modbus_Master se llama con MODE=0 y DATA_ADDR=10

Esta petición está activa hasta que finaliza con DONE=1 o ERROR=1 o hasta que ha transcurrido la vigilancia de tiempo parametrizada en el parámetro Blocked_Proc_Timeout . Si una vez finalizada la vigilancia de tiempo se inicia una nueva petición antes de que la petición anterior haya finalizado, la petición anterior se cancelará sin mensaje de error.

En caso de que, mientras se ejecuta esta petición, se produzca una segunda llamada de la instrucción con los mismos datos de instancia pero con ajustes distintos en los parámetros MODE y DATA_ADDR, esta segunda llamada finalizará con ERROR=1 y STATUS=8200.

Llamada de la instrucción Modbus_Master con diferentes ajustes de parámetros

Si el programa debe emplazar varias llamadas de la instrucción Modbus_Master con distintos ajustes para MB_ADDR, MODE, DATA_ADDR o DATA_LEN, debe asegurarse que solo haya una llamada activa a la vez, ya que, de lo contrario, se emitirá el mensaje de error 0x8200 (la interfaz está ocupada por una petición en curso).

Si el procesamiento de la llamada no puede completarse, se activará la vigilancia de tiempo a través del parámetro Blocked_Proc_Timeout y se cancelará la petición en curso.

Parámetro REQ

FALSE = no hay solicitud; TRUE = solicitud de enviar datos al esclavo Modbus

Habilitación de la transferencia solicitada. Con ello, el contenido del búfer se transfiere a la interfaz de comunicación punto a punto.

Por medio de los parámetros DATA_ADDR y MODE se selecciona el código de función Modbus.

DATA_ADDR (dirección inicial Modbus en el esclavo): indica la dirección inicial de los datos a los que debe accederse en el esclavo Modbus.

La instrucción Modbus_Master utiliza una entrada MODE en lugar de una entrada de código de función. La combinación entre MODE y DATA_ADDR especifica el código de función que se utiliza en la trama Modbus propiamente dicha. La tabla siguiente muestra la asignación entre el parámetro MODE, el código de función Modbus y el área de direcciones Modbus en DATA_ADDR.

Tabla 9-146 Funciones Modbus

MODE	DATA_ADDR (dirección Modbus)			DATA_LEN (longitud de los datos)			Código de función Modbus	Operación y datos		
0				Bits por solicitud			01	Leer bits de salida:		
	1	a	9999	1	a	2000/1992 [1]		0	a	9998
0				Bits por solicitud			02	Leer bits de entrada:		

MODE	DATA_ADDR (dirección Modbus)			DATA_LEN (longitud de los datos)			Código de función Modbus	Operación y datos		
	10001	a	19999	1	a	2000/1992 [1]		0	a	9998
0				Palabras por solicitud			03	Leer registro de paradas:		
	40001	a	49999	1	a	125/124 [1]		0	a	9998
	400001	a	465535	1	a	125/124 [1]		0	a	65534
0				Palabras por solicitud			04	Leer palabras de entrada:		
	30001	a	39999	1	a	125/124 [1]		0	a	9998
1				Bit por solicitud			05	Escribir un bit de salida:		
	1	a	9999	1				0	a	9998
1				1 palabra por solicitud			06	Escribir un registro de paradas:		
	40001	a	49999	1				0	a	9998
	400001	a	465535	1				0	a	65524
1				Bits por solicitud			15	Escribir varios bits de salida:		
	1	a	9999	2	a	1968/1960 [1]		0	a	9998
1				Palabras por solicitud			16	Escribir varios registros de parada:		
	40001	a	49999	2	a	123/122		0	a	9998
	400001	a	465534	2	a	123/122 [1]		0	a	65534
2 [2]				Bits por solicitud			15	Escribir uno o varios bits de salida:		
	1	a	9999	2	a	1968/1960 [1]		0	a	9998
2 [2]				Palabras por solicitud			16	Escribir uno o varios registros de paradas:		
	40001	a	49999	1	a	123		0	a	9998
	400001	a	465535	1	a	122 [1]		0	a	65534
11	Los dos operandos DATA_ADDR y DATA_LEN de Modbus_Master se ignoran en esta función.						11	Leer palabra de estado y contador de eventos de la comunicación del esclavo. La palabra de estado indica la ocupación (0 - no ocupado, 0xFFFF - ocupado). El contador de eventos se incrementa cada vez que se procesa correctamente una trama.		
80				1 palabra por solicitud			08	Comprobar el estado del esclavo mediante el código de diagnóstico de datos 0x0000 (test de bucle de control - el esclavo devuelve un eco de la solicitud)		
	-			1				-		
81				1 palabra por solicitud			08	Inicializar el contador de eventos del esclavo mediante el código de diagnóstico de datos 0x000A		

MODE	DATA_ADDR (dirección Modbus)			DATA_LEN (longitud de los datos)			Código de función Modbus	Operación y datos		
	-			1				-		
3 a 10, 12 a 79, 82 a 255	-			-				Reservado		

[1] En el direccionamiento avanzado (ver parámetro Extended_Adressing), la longitud de datos máxima es 1 byte o 1 palabra menos, según sea el tipo de datos de la función.

[2] MODE 2 permite escribir uno o varios bits de salida y uno o varios registros Holding utilizando las funciones Modbus 15 y 16. MODE 1 utiliza las funciones Modbus 5 y 6 para escribir 1 bit de salida y 1 registro Holding y las funciones Modbus 15 y 16 para escribir varios bits de salida y varios registros Holding.

Parámetro DATA_PTR

El parámetro DATA_PTR indica la dirección DB o M en la que se escribe o de la que se lee. Si se utiliza un bloque de datos, hay que crear un bloque de datos global que ponga a disposición la memoria de datos para procesos de lectura y escritura en esclavos Modbus.

Nota

S7-1500 - El bloque de datos direccionado mediante DATA_PTR debe permitir el direccionamiento directo

El bloque de datos debe permitir el direccionamiento directo (absoluto) y simbólico.

Estructuras de bloques de datos para el parámetro DATA_PTR

- Estos tipos de datos son válidos para **leer palabras** del área de direcciones Modbus (DATA_PTR) 30001 a 39999, 40001 a 49999 y 400001 a 465535, así como para **escribir palabras** del área de direcciones Modbus (parámetro DATA_PTR) 40001 a 49999 y 400001 a 465535.
  - Matriz estándar de los tipos de datos WORD, UINT o INT
  - Estructura denominada del tipo WORD, UINT o INT en la que cada elemento tiene un nombre unívoco y un tipo de datos de 16 bits.
  - Estructura compleja denominada en la que cada elemento tiene un nombre unívoco y un tipo de datos de 16 o 32 bits.
- Para **leer** y escribir bits para el área de direcciones Modbus (parámetro DATA_PTR) 00001 a 09999 y para leer bits de 10001 a 19999.
  - Campo estándar de tipos de datos booleanos.
  - Estructura booleana denominada de variables booleanas con una denominación unívoca.

- Si bien no es necesario que cada instrucción Modbus_Master tenga un área de memoria propia y separada, es recomendable. Esto se debe a que la posibilidad de que se destruyan datos es mayor cuando varias instrucciones Modbus_Master leen y escriben en la misma área de memoria.
- No es necesario que las áreas de memoria de DATA_PTR estén en el mismo bloque de datos global. Es posible crear un bloque de datos con varias áreas para procesos de lectura Modbus, un bloque de datos para procesos de escritura Modbus o un bloque de datos para cada estación esclava.

Consulte también

Modbus_Slave: comunicarse como esclavo Modbus (Página 5474)

Modbus_Slave: comunicarse como esclavo Modbus

Descripción

La instrucción Modbus_Slave permite al programa comunicarse como esclavo Modbus a través de un CM (RS422/485 o RS232). STEP 7 crea automáticamente un DB de instancia cuando se inserta la instrucción. El parámetro MB_DB de la instrucción Modbus_Comm_Load debe interconectarse con el parámetro MB_DB (estático) de la instrucción Modbus_Slave .

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
MB_ADDR	IN	UInt	Word	-	Dirección de estación del esclavo Modbus: área de direccionamiento estándar (1 a 247) área de direccionamiento ampliada (0 a 65535) Nota: 0 es la dirección Broadcast
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Modbus_Slave Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
MB_HOLD_REG	IN/OUT	Variant	Any	-	Puntero hacia el DB de registro de paradas Modbus: el registro de paradas Modbus puede ser el área de memoria de las marcas o bien un bloque de datos.
NDR	OUT	Bool		FALSE	Datos nuevos preparados: • FALSE – No hay datos nuevos • TRUE – Indica que se han escrito datos nuevos del maestro Modbus El bit NDR es TRUE durante un ciclo después de que la última solicitud haya finalizado sin errores.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
DR	OUT	Bool		FALSE	Leer datos: • FALSE – No se han leído datos • TRUE – Indica que la instrucción del maestro Modbus ha guardado datos recibidos en la zona de destino. El bit DR es TRUE durante un ciclo después de que la última solicitud haya finalizado sin errores.
ERROR	OUT	Bool		FALSE	El bit ERROR es TRUE durante un ciclo después de que la última solicitud haya finalizado con errores. Si la ejecución finaliza con un error, el código de error del parámetro STATUS solo es válido en el ciclo en que ERROR = TRUE.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 5485))

Los códigos de función de la comunicación Modbus (1, 2, 4, 5 y 15) pueden leer y escribir bits y palabras directamente en la memoria imagen de proceso de las entradas y en la memoria imagen de proceso de las salidas de la CPU. En estos códigos de función debe definirse el parámetro MB_HOLD_REG como un tipo de datos mayor que un byte. La tabla siguiente muestra la asignación de ejemplo de las direcciones Modbus a la memoria imagen de proceso de la CPU.

Tabla 9-147 Asignación de las direcciones Modbus a la memoria imagen de proceso

Funciones Modbus						S7-1500/S7-300/S7-400			
Código de función	Función	Área de datos	Área de direcciones			Área de datos	Dirección de la CPU		
01	Leer bits	Salida	0	a	9998	Memoria imagen de proceso de las salidas	Q0.0	a	Q1248.6
02	Leer bits	Entrada	0	a	9998	Memoria imagen de proceso de las entradas	I0.0	a	I1248.6
04	Leer palabras	Entrada	0	a	9998	Memoria imagen de proceso de las entradas	IW0	a	IW19996
05	Escribir bit	Salida	0	a	9998	Memoria imagen de proceso de las salidas	Q0.0	a	Q1248.6
15	Escribir bits	Salida	0	a	9998	Memoria imagen de proceso de las salidas	Q0.0	a	Q1248.6

Nota

El área de direcciones disponible puede ser también más pequeña, en función de la configuración de memoria de la CPU.

Los códigos de función de la comunicación Modbus (3, 6, 16) utilizan un registro de paradas Modbus que puede ser un área de direcciones del área de memoria de las marcas o bien un

bloque de datos. El tipo del registro de paradas se indica en el parámetro MB_HOLD_REG de la instrucción Modbus_Slave.

Nota

S7-1500 - Tipo del bloque de datos MB_HOLD_REG

Un bloque de datos con registro de paradas Modbus debe permitir el direccionamiento directo (absoluto) y simbólico.

Variables del esclavo Modbus

Esta tabla muestra las variables estáticas públicas del bloque de datos de instancia de Modbus_Slave que pueden utilizarse en el programa.

Tabla 9-148 Variables del esclavo Modbus

Variable	Tipo de datos	Estándar	Descripción
HR_Start_Offset	Word	0	Indica la dirección inicial del registro de paradas Modbus (estándar = 0)
Extended_Addressing	Bool	FALSE	Direccionamiento avanzado, configura el direccionamiento del esclavo como byte individual o doble (FALSE = dirección de byte único, TRUE = dirección de byte doble)
Request_Count	Word	0	Número de todas las solicitudes recibidas por este esclavo
Slave_Message_Count	Word	0	Número de las solicitudes recibidas para este esclavo específico
Bad_CRC_Count	Word	0	Número de solicitudes recibidas que presentan un error CRC
Broadcast_Count	Word	0	Número de solicitudes Broadcast recibidas
Exception_Count	Word	0	Errores específicos de Modbus, que se acusan en el maestro con una excepción
Success_Count	Word	0	Número de las solicitudes recibidas para este esclavo específico sin errores de protocolo
MB_DB	MB_BASE	-	El parámetro MB_DB de la instrucción Modbus_Comm_Load debe conectarse con el parámetro MB_DB de la instrucción Modbus_Master.

El programa puede escribir valores en las variables HR_Start_Offset y Extended_Addressing y controlar las operaciones del esclavo Modbus. Las demás variables pueden leerse para vigilar el estado de Modbus.

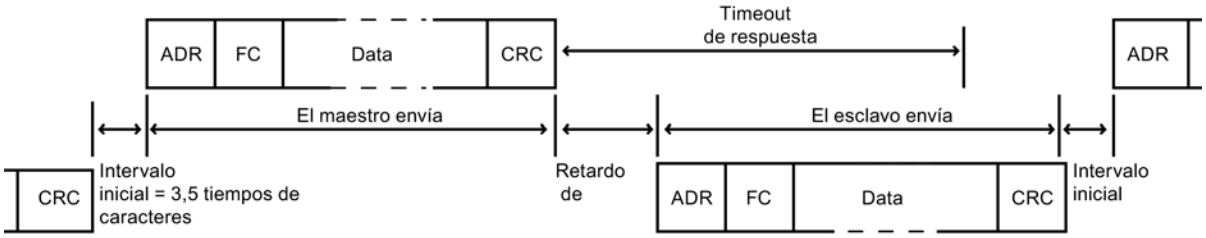
Reglas para la comunicación del esclavo Modbus

- Modbus_Comm_Load debe ejecutarse para configurar un puerto con el fin de que la instrucción Modbus_Slave pueda comunicarse a través de dicho puerto.
- Si un puerto debe reaccionar como esclavo a un maestro Modbus, dicho puerto no debe programarse con la instrucción Modbus_Master .
- Solo es posible utilizar una instancia de Modbus_Slave con un puerto concreto, de lo contrario pueden producirse comportamientos imprevistos.

- Las instrucciones Modbus no utilizan eventos de alarma de la comunicación para controlar el proceso de comunicación. El programa debe controlar el proceso de comunicación consultando la instrucción Modbus_Slave respecto de procesos de transmisión y recepción terminados.
- La instrucción Modbus_Slave debe ejecutarse regularmente y con una frecuencia que permita una respuesta temprana a solicitudes entrantes de un maestro Modbus. Se recomienda ejecutar Modbus_Slave en cada ciclo desde un OB de ciclo de programa. Modbus_Slave puede ejecutarse desde un OB de alarma cíclica, aunque no es recomendable, porque unos retardos de tiempo excesivos en el programa de interrupción pueden bloquear temporalmente la ejecución de otros programas de interrupción.

Control de tiempo de la señal Modbus

Modbus_Slave debe ejecutarse regularmente para recibir todas las solicitudes del maestro Modbus y responder de un modo adecuado. La frecuencia de ejecución de Modbus_Slave se rige por el valor de timeout para la respuesta especificado por el maestro Modbus. Esto se representa en la figura siguiente.



El período de timeout de la respuesta (RESP_TO) es el tiempo que espera un maestro Modbus al principio de una respuesta de un esclavo Modbus. Dicho período no se define en el protocolo Modbus, sino en un parámetro de la instrucción Modbus_Comm_Load. Puesto que tanto la recepción como la transmisión de una trama requieren varias llamadas (tres como mínimo) de la instrucción Modbus_Slave, debería ejecutarse Modbus_Slave un mínimo de doce veces durante el período de timeout para la respuesta del maestro Modbus para que la recepción y transmisión de datos por parte del esclavo Modbus se ejecute el doble de veces de las especificadas por el período de timeout.

HR_Start_Offset

Las direcciones del registro de paradas Modbus empiezan en 40001 o 400001. Estas direcciones equivalen a la dirección inicial del registro de paradas en la memoria del sistema de destino. De todas formas, es posible configurar la variable HR_Start_Offset para configurar una dirección inicial distinta de 40001 o 400001 para el registro de paradas Modbus.

La dirección 0 en el telegrama de recepción corresponde a la dirección inicial del registro de paradas en la memoria del sistema de destino. A través de la variable HR_Start_Offset se puede configurar otra dirección inicial como 0 para el registro de paradas de Modbus.

Así, por ejemplo, existe la posibilidad de configurar un registro de paradas que empiece en MW100 y una longitud de 100 palabras. Con HR_Start_Offset = 20, la dirección 20 del telegrama de recepción corresponde a la dirección inicial del registro de paradas en la memoria de destino (MW100). Toda dirección del telegrama de recepción por debajo de 20 y por encima de 119 provoca un error de direccionamiento.

Tabla 9-149 Ejemplo de direccionamiento del registro de paradas de Modbus si DATA_PTR señala a MW100 con una longitud de 100 palabras

HR_Start_Offset	Dirección	Mínimo	Máximo
0	Dirección Modbus (palabra)	0	99
	Dirección S7-1500	MW100	MW298
20	Dirección Modbus (palabra)	20	119
	Dirección S7-1500	MW100	MW298

HR_Start_Offset es un valor de palabra que indica la dirección inicial del registro de paradas Modbus y está guardado en el bloque de datos de instancia Modbus_Slave . Esta variable estática pública se selecciona mediante la lista desplegable de parámetros después de haber insertado Modbus_Slave en el programa.

Si, por ejemplo, se ha insertado Modbus_Slave en un segmento KOP, es posible pasar a un segmento anterior y asignar el valor HR_Start_Offset con el comando Move. El valor debe asignarse antes de ejecutar Modbus_Slave.

Introducir una variable de esclavo Modbus mediante el nombre de DB estándar:

1. Coloque el cursor en el campo de parámetros OUT1 e introduzca el carácter m.
2. Seleccione en la lista desplegable el DB de instancia que desee para la instrucción Modbus_Slave.
3. Coloque el cursor a la derecha del nombre de DB (después de las comillas) e introduzca un punto.
4. Seleccione en la lista desplegable "Modbus_Slave_DB.HR_Start_Offset".

Estructura de la trama

Extended_Addressing

A la variable Extended_Addressing se accede de un modo similar que a la referencia HR_Start_Offset, excepto por el hecho de que la variable Extended_Addressing es un valor booleano.

Para el direccionamiento de esclavo Modbus puede configurarse un byte individual (el estándar de Modbus) con Extended_Adressing = FALSE, o bien dos bytes (Extended_Adressing = TRUE). El direccionamiento avanzado sirve para direccionar más de 247 dispositivos en una sola red. Con Extended_Adressing = TRUE es posible activar un máximo de 65535 direcciones. A continuación se muestra una trama Modbus a modo de ejemplo.

Tabla 9-150 Dirección del esclavo con un byte (byte 0)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Solicitud	Dirección del esclavo	Código de función	Dirección inicial		Datos		
Respuesta válida	Dirección del esclavo	Código de función	Longitud	Datos...			
Mensaje de error	Dirección del esclavo	0xxx	Código de excepción				

Tabla 9-151 Dirección del esclavo con dos bytes (byte 0 y byte 1)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Solicitud	Dirección del esclavo		Código de función	Dirección inicial		Datos	
Respuesta válida	Dirección del esclavo		Código de función	Longitud	Datos...		
Mensaje de error	Dirección del esclavo		0xxx	Código de excepción			

Descripción de las tramas

El intercambio de datos entre maestro y esclavo comienza con la dirección del esclavo, seguida del código de función. A continuación se transfieren los datos. La estructura del campo de datos depende del código de función utilizado. Al final de la trama se transmitirá la suma de comprobación (CRC).

Código de función 1 - Esta función permite leer bits de salida individuales

Tabla 9-152 FC 1 - Leer bits de salida

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 1	Dirección inicial		Número de salidas	
Respuesta válida	Dirección del esclavo	Código de función 1	Longitud [1]	Datos de salida [3)]		
Mensaje de error	Dirección del esclavo	0x81	Código de excepción [2]	---		

[1] Longitud: si de la división de las salidas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

[3] Los datos de salida pueden abarcar varios bytes

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 1	Dirección inicial		Número de salidas	
Respuesta válida	Dirección del esclavo		Código de función 1	Longitud [1]	Datos de salida		
Mensaje de error	Dirección del esclavo		0x81	Código de excepción [2]	---		

[1] Longitud: si de la división de las salidas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

[3] Los datos de salida pueden abarcar varios bytes

Código de función 2 - Esta función permite leer bits de entrada individuales

Tabla 9-153 FC 2 - Leer bits de entrada

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 2	Dirección inicial		Número de entradas	
Respuesta válida	Dirección del esclavo	Código de función 2	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo	0x82	Código de excepción [2]	---		

[1] Longitud: si de la división de las entradas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 2	Dirección inicial		Número de entradas	
Respuesta válida	Dirección del esclavo		Código de función 2	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo		0x82	Código de excepción [2]	---		

[1] Longitud: si de la división de las entradas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

Código de función 3 - Esta función permite leer registros individuales

Tabla 9-154 FC 3 - Leer registro de paradas

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 3	Dirección inicial		Número de registros	
Respuesta válida	Dirección del esclavo	Código de función 3	Longitud [1]	Datos de registro		
Mensaje de error	Dirección del esclavo	0x83	Código de excepción [2]	---		

[1] Longitud: Número de bytes

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 3	Dirección inicial		Número de registros	
Respuesta válida	Dirección del esclavo		Código de función 3	Longitud [1]	Datos de registro		
Mensaje de error	Dirección del esclavo		0x83	Código de excepción [2]	---		

[1] Longitud: Número de bytes

[2] Código E: 01, 02, 03 o 04

Código de función 4 - Esta función permite leer registros individuales

Tabla 9-155 FC 4 - Leer palabras de entrada

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 4	Dirección inicial		Número de palabras de entrada	
Respuesta válida	Dirección del esclavo	Código de función 4	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo	0x84	Código de excepción [2]	---		

[1] Longitud: 2 * número de palabras de entrada

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 4	Dirección inicial		Número de palabras de entrada	
Respuesta válida	Dirección del esclavo		Código de función 4	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo		0x84	Código de excepción [2]	---		

[1] Longitud: 2 * número de palabras de entrada

[2] Código E: 01, 02, 03 o 04

Código de función 5- Con esta función puede activarse o borrarse un bit individual

Tabla 9-156 FC 5 - Escribir un bit de salida

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 5	Dirección inicial		Valor	
Respuesta válida	Dirección del esclavo	Código de función 5	Longitud	Valor		
Mensaje de error	Dirección del esclavo	0x85	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 5	Dirección inicial		Valor	
Respuesta válida	Dirección del esclavo		Código de función 5	Longitud	Valor		
Mensaje de error	Dirección del esclavo		0x85	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

Código de función 6 - Esta función permite escribir registros individuales

Tabla 9-157 FC 6 - Escribir registro de paradas

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 6	Dirección		Registros	
Respuesta válida	Dirección del esclavo	Código de función 6	Dirección		Registros	
Mensaje de error	Dirección del esclavo	0x86	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 6	Dirección		Registros	
Respuesta válida	Dirección del esclavo		Código de función 6	Dirección		Registros	
Mensaje de error	Dirección del esclavo		0x86	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

Código de función 8 - Esta función permite comprobar la conexión de comunicación

Tabla 9-158 FC 8 - Estado del esclavo

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 8	Código de diagnóstico		Valor de prueba	
Respuesta válida	Dirección del esclavo	Código de función 8	Código de diagnóstico		Valor de prueba	
Mensaje de error	Dirección del esclavo	0x88	Código de excepción [1]	---		

[1] Código E: 01, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 8	Código de diagnóstico		Valor de prueba	
Respuesta válida	Dirección del esclavo		Código de función 8	Código de diagnóstico		Valor de prueba	
Mensaje de error	Dirección del esclavo		0x88	Código de excepción [1]	---		

[1] Código E: 01, 03 o 04

Código de función 11 - Esta función permite leer una "Status-Word" de 2 bytes y un "Event-Counter" de 2 bytes

Tabla 9-159 FC 11 - Contador de eventos de la comunicación del esclavo

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 11	---			
Respuesta válida	Dirección del esclavo	Código de función 11	Status		Contador de eventos	
Mensaje de error	Dirección del esclavo	0x8B	Código de excepción [1]	---		

[1] Código E: 01 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 11	---			
Respuesta válida	Dirección del esclavo		Código de función 11	Status		Contador de eventos	
Mensaje de error	Dirección del esclavo		0x8B	Código de excepción [1]	---		

[1] Código E: 01 o 04

Código de función 15 - Esta función permite escribir varios bits

Tabla 9-160 FC 15 - Escribir uno o varios bits de salida

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Petición	Dirección del esclavo	Código de función 15	Dirección inicial		Número de palabras de salida		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo	Código de función 15	Dirección inicial		Número de palabras de salida		---		
Mensaje de error	Dirección del esclavo	0x8F	Código de excepción [2]	---					

[1] Contador de bytes: si de la división del número de bytes entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Petición	Dirección del esclavo		Código de función 15	Dirección inicial		Número de palabras de salida		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo		Código de función 15	Dirección inicial		Número de palabras de salida		---		
Mensaje de error	Dirección del esclavo		0x8F	Código de excepción [2]		---				

[1] Contador de bytes: si de la división del número de bytes entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

Código de función 16 - Esta función permite escribir uno o varios registros

Tabla 9-161 FC 16 - Escribir uno o varios registros de paradas

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Petición	Dirección del esclavo	Código de función 16	Dirección inicial		Número de registros		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo	Código de función 16	Dirección inicial		Número de registros		---		
Mensaje de error	Dirección del esclavo	0x90	Código de excepción [2]	---					

[1] Contador de bytes: Número de registros * 2

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Petición	Dirección del esclavo		Código de función 16	Dirección inicial		Número de registros		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo		Código de función 16	Dirección inicial		Número de registros		---		
Mensaje de error	Dirección del esclavo		0x90	Código de excepción [2]		---				

[1] Contador de bytes: Número de registros * 2

[2] Código E: 01, 02, 03 o 04

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)

Código de error	Descripción	Solución
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)

Código de error	Descripción	Solución
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		

Código de error	Descripción	Solución
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Recibir datos con Freeport (Página 1028)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)

Código de error	Descripción	Solución
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		

Código de error	Descripción	Solución
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		

Código de error	Descripción	Solución
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 [1)]	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.

Código de error	Descripción	Solución
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 [1)]	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 [1)]	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.

Código de error	Descripción	Solución
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.

Código de error	Descripción	Solución
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.

Código de error	Descripción	Solución
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información

Código de error	Descripción	Solución
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 5468) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.

Código de error	Descripción	Solución
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1)]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1)]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1)]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1)]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1)]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1)]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG

Código de error	Descripción	Solución
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.

Código de error	Descripción	Solución
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

[1] Solo en instrucciones para CPU S7-300/400

Punto a punto

PORT_CFG: Configurar parámetros de comunicación dinámicamente

Descripción

La instrucción "PORT_CFG" sirve para configurar dinámicamente los parámetros de comunicación de un puerto de comunicación punto a punto.

La configuración estática original del puerto se ajusta en la configuración de hardware. Esta configuración se puede modificar ejecutando la instrucción "PORT_CFG". Esta función también sirve, por ejemplo, para guardar en librerías los bloques creados y no tener que configurar el puerto en la configuración del hardware en caso de reutilizarlos.

"PORT_CFG" permite modificar los siguientes ajustes de los parámetros de comunicación:

- Paridad
- Velocidad de transferencia
- Número de bits por carácter
- Número de bits de parada
- Tipo y propiedades del control de flujo

Las modificaciones realizadas por la instrucción "PORT_CFG" no se guardan de forma permanente en el sistema de destino.

Los datos serie pueden transferirse vía las conexiones eléctricas RS-232 (semidúplex y dúplex) y RS-485 (semidúplex).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PORT_CFG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Activa la modificación de la configuración cuando se detecta un flanco ascendente.
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
PROTOCOL	Input	UINT	I, Q, M, D, L o constante	Protocolo de transmisión: • 0: protocolo de comunicación punto a punto • 1..n: definición futura para protocolos de transmisión específicos
BAUD	Input	UINT	I, Q, M, D, L o constante	Velocidad de transferencia del puerto: • 1: 300 baudios • 2: 600 baudios • 3: 1200 baudios • 4: 2400 baudios • 5: 4800 baudios • 6: 9600 baudios (ajuste estándar) • 7: 19200 baudios • 8: 38400 baudios • 9: 57600 baudios • 10: 76800 baudios • 11: 115200 baudios
PARITY	Input	UINT	I, Q, M, D, L o constante	Paridad del puerto: • 1: Sin paridad (ajuste estándar) • 2: Paridad par • 3: Paridad impar • 4: Selección de paridad • 5: Espacio de paridad
DATABITS	Input	UINT	I, Q, M, D, L o constante	Bits por carácter: • 1: 8 bits por carácter (ajuste estándar) • 2: 7 bits por carácter
STOPBITS	Input	UINT	I, Q, M, D, L o constante	Número de bits de parada: • 1: 1 bit de parada (ajuste estándar) • 2: 2 bits de parada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
FLOWCTRL	Input	UINT	I, Q, M, D, L o constante	Control de flujo de datos: • 1: Ninguno (ajuste estándar) • 2: XON/XOFF • 3: Control de flujo por hardware (RTS activado permanentemente) • 4: Control de flujo por hardware (RTS desactivable durante la transferencia)
XONCHAR	Input	CHAR	I, Q, M, D, L o constante	Indica el carácter que se utiliza como XON. El ajuste estándar es el carácter DC1 (11H).
XOFFCHAR	Input	CHAR	I, Q, M, D, L o constante	Indica el carácter que se utiliza como XOFF. El ajuste estándar es el carácter DC3 (13H).
WAITTIME	Input	UINT	I, Q, M, D, L o constante	Indica el tiempo de espera hasta XON o CTS tras el inicio de la transferencia. El valor especificado debe ser mayor que 0. El ajuste estándar es 2000 milisegundos.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
80A0	El protocolo especificado no es válido.
80A1	La velocidad de transferencia especificada no es válida.
80A2	La paridad especificada no es válida.
80A3	El número indicado de bits por carácter no es válido.
80A4	El número indicado de bits de parada no es válido.
80A5	El tipo indicado de control de flujo no es válido.
80A6	Valor erróneo en el parámetro WAITTIME Si el control de flujo de datos está activado, el valor del parámetro WAITTIME debe ser mayor que cero.
80A7	Valores no válidos en los parámetros XONCHAR y XOFFCHAR.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

SEND_CFG: Configurar parámetros de transmisión serie dinámicamente

Descripción

La instrucción "SEND_CFG" permite configurar dinámicamente los parámetros de transferencia serie de un puerto de comunicación punto a punto. Todos los mensajes pendientes para la transferencia se rechazan tras la ejecución de "SEND_CFG".

La configuración estática original del puerto se ajusta en la configuración de hardware. Esta configuración se puede modificar ejecutando la instrucción "SEND_CFG". Esta función también sirve, por ejemplo, para guardar en librerías los bloques creados y no tener que configurar el puerto en la configuración del hardware en caso de reutilizarlos. "SEND_CFG" permite modificar los siguientes ajustes de los parámetros de transferencia:

- Intervalo entre la activación de RTS (Request to Send) y el inicio de la transferencia
- Intervalo entre el final de la transferencia y la desactivación de RTS
- Definir tiempos de bit para pausas

Las modificaciones realizadas por la instrucción "SEND_CFG" no se guardan de forma permanente en el sistema de destino.

Los datos serie pueden transferirse vía las conexiones eléctricas RS-232 (semidúplex y dúplex) y RS-485 (semidúplex).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SEND_CFG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Activa la modificación de la configuración cuando se detecta un flanco ascendente.
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
RTSONDLY	Input	UINT	I, Q, M, D, L o constante	Tiempo que debe transcurrir desde la activación de RTS hasta el inicio de la transferencia. Los valores válidos para este parámetro son: • 0 (ajuste estándar) • de 0 a 65535 ms en incrementos de 1 ms Este parámetro no es válido para los módulos RS-485.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RTSOFFDLY	Input	UINT	I, Q, M, D, L o constante	Tiempo que debe transcurrir desde el final de la transferencia hasta la desactivación de RTS. Los valores válidos para este parámetro son: • 0 (ajuste estándar) • de 0 a 65535 ms en incrementos de 1 ms Este parámetro no es válido para los módulos RS-485.
BREAK	Input	UINT	I, Q, M, D, L o constante	Indica los tiempos de bit para una pausa que se envía al inicio del mensaje. El ajuste estándar es 12 tiempos de bit. Se admiten como máximo 25000 tiempos de bit.
IDLELINE	Input	UINT	I, Q, M, D, L o constante	Indica los tiempos de bit para Idle Line tras la pausa al inicio del mensaje. El ajuste estándar es 12 tiempos de bit. Se admiten como máximo 25000 tiempos de bit.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
80B0	No está permitido configurar una interrupción de la transferencia
80B1	El tiempo de pausa especificado excede el máximo permitido de 25000 tiempos de bit
80B2	El tiempo especificado para Idle Line excede el máximo permitido de 25000 tiempos de bit
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RCV_CFG: Configurar parámetros de recepción serie dinámicamente

Descripción

La instrucción "RCV_CFG" permite configurar dinámicamente los parámetros de recepción serie de un puerto de comunicación punto a punto. Esta instrucción permite configurar las condiciones que determinan el principio y fin de un mensaje que debe transferirse. La recepción de los mensajes que cumplen dichas condiciones se habilita mediante la instrucción "RCV_PTP (Página 5513)".

La configuración estática original del puerto se ajusta en las propiedades de la configuración de hardware. Ejecute la instrucción "RCV_CFG" en el programa con objeto de modificar la configuración. Esta función también sirve, por ejemplo, para guardar en librerías los bloques creados y no tener que configurar el puerto en la configuración del hardware en caso de reutilizarlos. Las modificaciones realizadas por la instrucción "RCV_CFG" no se guardan de forma permanente en el sistema de destino.

Todos los mensajes pendientes para la transferencia se rechazan tras la ejecución de la instrucción "RCV_CFG".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RCV_CFG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Activa la modificación de la configuración cuando se detecta un flanco ascendente.
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
CONDITIONS	Input	CONDITIONS	D, L	Estructura de datos que define las condiciones de inicio y fin de la transferencia de datos.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Tipo de datos CONDITIONS

Por medio de la estructura CONDITIONS se definen las condiciones de inicio y fin para la transferencia de mensajes. La estructura CONDITIONS está incluida en el bloque de datos de instancia de la instrucción "RCV_CFG". Por medio de la estructura CONDITIONS se

definen las condiciones de inicio y fin, el momento en el que finaliza la transferencia de un mensaje y el momento en el que empieza la siguiente.

- La condición de inicio para la transferencia de datos se define en la estructura START
- La condición de fin para la transferencia de datos se define en la estructura END

Es posible definir una o varias condiciones de inicio y fin. Si se especifican varias condiciones de inicio o fin, se enlazan con una puerta O.

La tabla siguiente muestra la estructura CONDITIONS:

Parámetro		Tipo de datos	Descripción
START		STRUCT	Condiciones de inicio
	STARTCOND	UINT	Especifica la condición de inicio (detalles más abajo). La condición de inicio se especifica en forma de valor hexadecimal de 16 bits. Los valores posibles para dicha condición son: • 1: carácter inicial • 2: cualquier carácter (ajuste estándar) • 4: Line break • 8: Idle line • 16: cadena de caracteres 1 • 32: cadena de caracteres 2 • 64: cadena de caracteres 3 • 128: cadena de caracteres 4 También es posible definir varias condiciones de inicio en el parámetro STARTCOND. Para ello se indica la suma de los valores de las diferentes condiciones. Si, por ejemplo, debe definirse la condición de inicio "Idle line" O "cadena de caracteres 1" O "cadena de caracteres 4", hay que especificar el valor "152".
	IDLETIME	UINT	Indica el tiempo de pausa máximo de la línea antes de iniciar la recepción. Los valores válidos para este parámetro son: • 40 tiempos de bit (ajuste estándar) • de 0 a 2500 tiempos de bit
	STARTCHAR	BYTE	Especifica el carácter inicial. Dicho ajuste sólo es efectivo si la condición de inicio configurada es "carácter inicial". Los valores válidos para este parámetro son: • 02 (STX): ajuste estándar • B#16#00 a B#16#FF
	SEQ[1].CTL	BYTE	Cadena de caracteres 1: control de cada carácter La posición de bit de los caracteres permite determinar qué caracteres de la cadena se consideran o ignoran. Para evaluar los caracteres hay que activar los bits correspondientes. • Bit 0: 1 carácter • Bit 1: 2 caracteres • Bit 2: 3 caracteres • Bit 3: 4 caracteres • Bit 4: 5 caracteres Un carácter se ignora cuando se desactiva el bit correspondiente.

Parámetro		Tipo de datos	Descripción
	SEQ[1].STR	CHAR[5]	Cadena de caracteres 1: carácter inicial (5 caracteres)
	SEQ[2].CTL	BYTE	Cadena de caracteres 2: ignorar/comparar el control de cada carácter
	SEQ[2].STR	CHAR[5]	Cadena de caracteres 2: carácter inicial (5 caracteres)
	SEQ[3].CTL	BYTE	Cadena de caracteres 3: ignorar/comparar el control de cada carácter
	SEQ[3].STR	CHAR[5]	Cadena de caracteres 3: carácter inicial (5 caracteres)
	SEQ[4].CTL	BYTE	Cadena de caracteres 4: ignorar/comparar el control de cada carácter
	SEQ[4].STR	CHAR[5]	Cadena de caracteres 4: carácter inicial (5 caracteres)
END		STRUCT	Condiciones de fin
	ENDCOND	UINT	Especifica la condición de fin (detalles más abajo). La condición de fin se especifica en forma de valor hexadecimal de 16 bits. Los valores posibles para dicha condición son: • 1: tiempo de respuesta excedido • 2: tiempo de mensaje excedido • 4: tiempo excedido dentro de la cadena de caracteres • 8: longitud máxima • 16: N+LEN+M; la información sobre la longitud del mensaje está integrada en él y es evaluada. • 32: cadena de caracteres 1 También es posible definir varias condiciones de fin en el parámetro ENDCOND. Para ello se indica la suma de los valores de las diferentes condiciones de fin. Si, por ejemplo, debe definirse la condición de fin "longitud máxima" O "secuencia 1", hay que especificar el valor "40".
	MAXLEN	UINT	Indica el número de caracteres máximo de un mensaje. Los valores* válidos para este parámetro son: • 1 carácter (ajuste estándar) • de 0 a 1024 caracteres Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "longitud máxima" en el parámetro ENDCOND.
	N	UINT	Offset del campo de longitud en un mensaje Los valores válidos para este parámetro son: • 0 caracteres (ajuste estándar) • de 0 a 1024 caracteres Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "N+LEN+M" en el parámetro ENDCOND.
	LENGTHSIZE	UINT	Tamaño del campo de longitud en bytes Los valores* válidos para este parámetro son: • 0 bytes (ajuste estándar) • 1 byte • 2 bytes • 4 bytes Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "N+LEN+M" en el parámetro ENDCOND.

Parámetro		Tipo de datos	Descripción
	LENGTHM	UINT	Especifica el número de caracteres de fin que siguen al campo de longitud pero que no están contenidos en la longitud del mensaje. Los valores válidos para este parámetro son: • 0 caracteres (ajuste estándar) • de 0 a 255 caracteres Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "N+LEN+M" en el parámetro ENDCOND.
	RCVTIME	UINT	Indica la duración máxima para la recepción del primer carácter de un mensaje. Los valores válidos para este parámetro son: • 200 ms (ajuste estándar) • de 0 a 65535 ms en incrementos de 1 ms Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "tiempo de respuesta excedido" en el parámetro ENDCOND.
	MSGTIME	UINT	Indica la duración máxima para la recepción de un mensaje. Los valores válidos para este parámetro son: • 200 ms (ajuste estándar) • de 0 a 65535 ms en incrementos de 1 ms Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "tiempo de mensaje excedido" en el parámetro ENDCOND.
	CHARGAP	UINT	Especifica el intervalo entre la recepción de dos caracteres consecutivos. Los valores válidos para este parámetro son: • 12 tiempos de bit (ajuste estándar) • de 0 a 2500 tiempos de bit Dicho ajuste sólo es efectivo si se ha ajustado la condición de fin "tiempo excedido dentro de la cadena de caracteres" en el parámetro ENDCOND.
	SEQ.CTL	BYTE	Cadena de caracteres: control de cada carácter La posición de bit de los caracteres permite determinar qué caracteres de la cadena se consideran o ignoran. Para evaluar los caracteres hay que activar los bits correspondientes. • Bit 0: 1 carácter • Bit 1: 2 caracteres • Bit 2: 3 caracteres • Bit 3: 4 caracteres • Bit 4: 5 caracteres Un carácter se ignora cuando se desactiva el bit correspondiente.
	SEQ.STR	CHAR[5]	Cadena de caracteres: carácter inicial (5 caracteres)
	* Estos rangos también son aplicables a los respectivos ajustes de hardware para la definición del final del mensaje.		

Condiciones de inicio para la recepción de un mensaje (parámetro STARTCOND)

El receptor detecta el inicio de un mensaje cuando se produce la condición de inicio configurada. Las condiciones siguientes pueden definirse como iniciales para la recepción de mensajes:

- Carácter inicial: el inicio de un mensaje se detecta cuando aparece un carácter determinado. Dicho carácter se guarda como primer carácter del mensaje. Todos los caracteres recibidos antes del carácter inicial se rechazan.
- Cualquier carácter: un carácter cualquiera define el inicio de un mensaje. Dicho carácter se guarda como primer carácter del mensaje.
- Line Break: el inicio de un mensaje se detecta cuando el flujo de datos recibidos se ha interrumpido durante un tiempo superior a la longitud de un carácter.
- Idle Line: el inicio de un mensaje se detecta cuando la línea de transferencia está en reposo durante un tiempo determinado (indicado en tiempos de bit) y, después, se siguen transfiriendo caracteres.
- Cadena de caracteres (secuencia): el inicio de un mensaje se detecta cuando aparece una cadena de caracteres determinada en el flujo de datos. Se admiten como máximo cuatro secuencias de cinco caracteres cada una como máximo.
  Ejemplo: Un mensaje recibido con código hexadecimal contiene los siguientes caracteres: "68 10 aa 68 bb 10 aa 16". Las cadenas de caracteres de inicio configuradas se muestran en la siguiente
  tabla. Las cadenas de caracteres de inicio son evaluadas después de que el primer carácter 68H
  ha sido recibido correctamente. Después de recibir correctamente el cuarto carácter (el segundo 68H) se cumple la condición de inicio "1". Una vez cumplidas las condiciones de inicio,
  comienza la evaluación de las condiciones de fin.
  El procesamiento de la cadena de caracteres de inicio puede cancelarse debido a distintos errores de paridad,
  framing o intervalos entre los caracteres. Estos errores hacen
  que el mensaje no se reciba por no cumplirse la condición de inicio.

Condición de inicio	Primer carácter	Primer carácter +1	Primer carácter +2	Primer carácter +3	Primer carácter +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Condiciones de fin para la recepción de un mensaje (parámetro ENDCOND)

El receptor detecta el fin de un mensaje cuando se produce la condición de fin configurada. Las condiciones siguientes pueden definirse como finales para la recepción de mensajes:

- Tiempo de respuesta excedido: La recepción de un mensaje finaliza cuando se excede la duración máxima predefinida para la recepción de un carácter. La duración máxima se define en el parámetro RCVTIME. El transcurso del tiempo definido se inicia en cuanto ha finalizado la última transferencia y la instrucción RCV_PTP habilita la recepción del mensaje. Si dentro del tiempo definido (RCVTIME) no se ha recibido ningún carácter, la instrucción RCV_PTP notifica un error.
- Tiempo de mensaje excedido: La recepción de un mensaje finaliza cuando se excede la duración máxima predefinida para la recepción de un mensaje. La duración máxima se define en el parámetro MSGTIME. El transcurso del tiempo definido se inicia en cuanto se ha recibido el primer carácter del mensaje.
- Tiempo excedido dentro de la cadena de caracteres: La recepción de un mensaje finaliza cuando el intervalo de tiempo entre la recepción de dos caracteres consecutivos es mayor que el valor del parámetro CHARGAP.
- Longitud máxima: La recepción de un mensaje finaliza cuando se excede la longitud del mensaje definida en el parámetro MAXLEN .
- Leer longitud de mensaje (N+LEN+M): La recepción de un mensaje finaliza cuando se alcanza una longitud de mensaje determinada. Dicha longitud se calcula mediante los valores de los parámetros siguientes:
  - N: posición del carácter en el mensaje a partir del que empieza el campo de longitud.
  - LENGTHSIZE: tamaño del campo de longitud en bytes
  - LENGTHM: número de caracteres de fin que siguen al campo de longitud. Dichos caracteres no se toman en consideración al evaluar la longitud del mensaje.
- Cadena de caracteres: La recepción de un mensaje finaliza cuando se recibe una cadena de caracteres determinada. Dicha cadena admite cinco caracteres como máximo. Para cada carácter de la cadena es posible definir, a través de la posición de bit, si se considera o ignora en la evaluación.

Parámetro STATUS

Código de error* (W#16#...)	Descripción
80C0	Error en la condición de inicio
80C1	• Error en la condición de fin • No se ha definido ninguna condición de fin
80C2	Alarma de recepción activada
80C3	Si se ha ajustado la condición de fin "longitud máxima", en el parámetro MAXLEN se ha introducido un valor igual a 0 o mayor que 4132.
80C4	Si se ha ajustado la condición de fin "N+LEN+M", en el parámetro N se ha introducido un valor mayor que 4131.

Código de error* (W#16#...)	Descripción
80C5	Si se ha ajustado la condición de fin "N+LEN+M", en el parámetro LENGTHSIZE se ha introducido un valor igual a 0 o no válido.
80C6	Si se ha ajustado la condición de fin "N+LEN+M", en el parámetro LENGTHM se ha introducido un valor mayor que 255.
80C7	Si se ha ajustado la condición de fin "N+LEN+M", se ha calculado una longitud de mensaje mayor que 4132.
80C8	Si se ha ajustado la condición de fin "tiempo de respuesta excedido", en el parámetro RCVTIME se ha introducido un valor igual a 0.
80C9	Si se ha ajustado la condición de fin "tiempo excedido dentro de la cadena de caracteres", en el parámetro CHARGAP se ha introducido un valor igual a 0 o mayor que 2500.
80CA	Si se ha ajustado la condición de inicio "Idle line", en el parámetro IDLETIME se ha introducido un valor igual a 0 o mayor que 2500.
80CB	A pesar de haber ajustado la condición de fin "cadena de caracteres", todos los caracteres de la cadena están marcados como irrelevantes.
80CC	A pesar de haber ajustado la condición de inicio "cadena de caracteres", todos los caracteres de la cadena están marcados como irrelevantes.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

SEND_PTP: Transmitir datos del búfer de transmisión

Descripción

La instrucción "SEND_PTP" inicia la transferencia de datos. Sin embargo, la instrucción "SEND_PTP" no ejecuta la transferencia de datos propiamente dicha. Los datos del búfer de transmisión se envían al módulo de comunicación punto a punto (CM). El CM realiza entonces la transferencia propiamente dicha.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SEND_PTP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Habilitación de la transferencia solicitada al detectarse un flanco ascendente de esa entrada de habilitación. El contenido del búfer de transmisión se envía al módulo de comunicación punto a punto (CM).
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
BUFFER	Input	VARIANT	I, Q, M, D, L o constante	Puntero hacia la dirección inicial del búfer de transmisión. Valores booleanos o Array of BOOL no soportados.
LENGTH	Input	UINT	I, Q, M, D, L o constante	Longitud del búfer de transmisión
PTRCL	Input	BOOL	I, Q, M, D, L o constante	Este parámetro selecciona el búfer para la comunicación punto a punto normal o para protocolos específicos de Siemens que están implementados en el CM conectado. FALSE = operaciones punto a punto controladas por el programa de usuario (única opción disponible)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#...)	Descripción
7000	La operación de transmisión no está activa.
7001	La operación de transmisión procesa la primera llamada.
7002	La operación de transmisión procesa las llamadas posteriores (consultas tras la primera llamada).
8080	La identificación introducida para el puerto de comunicación no es válida.
8088	La longitud del parámetro LENGHT no se corresponde con la longitud de los datos que deben transmitirse. Consulte también: Parámetros LENGHT y BUFFER.
80D0	Se ha recibido una solicitud de transmisión nueva durante una transferencia en curso.
80D1	La transferencia se ha cancelado porque la señal CTS no se ha confirmado dentro del tiempo de espera especificado.

Código de error* (W#16#...)	Descripción
80D2	La solicitud de transmisión se ha cancelado porque el interlocutor (DCE) no ha señalizado la disponibilidad de recepción (DSR).
80D3	La solicitud de transmisión se ha cancelado porque se ha sobrepasado el tamaño máximo del intervalo de espera (más de 1024 bytes).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Parámetros LENGTH y BUFFER

El volumen mínimo de datos que puede enviarse con la instrucción "PTP_SEND" es un byte. El parámetro BUFFER define el volumen de datos que deben enviarse. Para el tipo de parámetro BUFFER no se puede utilizar ni el tipo de datos BOOL ni Array of BOOL.

Parámetro LENGTH	Parámetro BUFFER	Descripción
LENGTH = 0	No utilizado	Los datos completos se envían como definidos por el parámetro BUFFER. Si LENGTH = 0, no es necesario indicar el número de bytes transferidos.
LENGTH > 0	Tipo de datos simple	El valor LENGTH debe contener el número de bytes de este tipo de datos. De lo contrario no se produce ninguna transferencia y se visualiza el error 8088.
	STRUCT	El valor LENGTH puede contener un número de bytes inferior a toda la longitud de bytes de la estructura. En este caso sólo se transfieren los primeros bytes LENGTH.
	ARRAY	El valor LENGTH puede contener un número de bytes inferior a toda la longitud de bytes del campo. En este caso sólo se transfieren los elementos de campo que caben completamente en los bytes LENGTH. El valor LENGTH debe ser múltiplo del número de bytes de los elementos de datos. De lo contrario, STATUS = 8088, ERROR = 1 y no se transfiere nada.
	STRING	Se transfiere toda la disposición de la memoria del formato de la cadena de caracteres, y también la información sobre la longitud máxima de la cadena de caracteres y su longitud actual. El valor LENGTH debe contener los bytes de longitud máxima, longitud real y los caracteres de la cadena. En el tipo de datos STRING todas las longitudes y caracteres tienen el tamaño de un byte. Si para el parámetro BUFFER se utiliza una cadena de caracteres, el valor LENGTH también debe contener dos bytes para los dos campos de longitud.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RCV_PTP: Habilitar recepción de mensajes

Descripción

La instrucción "RCV_PTP" habilita la recepción de un mensaje transmitido. Cada mensaje se debe habilitar por separado. Los datos transmitidos no están disponibles en el área de recepción hasta que el interlocutor en cuestión haya acusado el mensaje.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RCV_PTP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L o constante	Habilitación de la recepción al detectarse un flanco ascendente.
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
BUFFER	Input	VARIANT	I, Q, M, D, L o constante	Apunta a la dirección inicial del búfer de recepción. No utilice ninguna variable del tipo STRING en el búfer de recepción.
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción
LENGTH	Output	UINT	I, Q, M, D, L	Longitud del mensaje en el búfer de recepción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#....)	Descripción
80E0	La recepción del mensaje se ha finalizado porque el búfer de recepción está lleno.
80E1	La recepción del mensaje se ha finalizado debido a un error de paridad.
80E2	La recepción del mensaje se ha finalizado debido a un error de framing.
80E3	La recepción del mensaje se ha finalizado debido a un error de desbordamiento.
80E4	La recepción del mensaje se ha finalizado porque la longitud calculada (N+LEN+M) excede el tamaño del búfer de recepción.
8080	La identificación introducida para el puerto de comunicación no es válida.
8088	Con el parámetro BUFFER se ha creado una referencia al tipo de datos STRING.
0094	La recepción del mensaje se ha finalizado porque se ha alcanzado la longitud máxima de caracteres.

Código de error* (W#16#....)	Descripción
0095	La recepción del mensaje se ha finalizado debido a un tiempo excedido.
0096	La recepción del mensaje se ha finalizado debido a un tiempo excedido dentro de la cadena de caracteres.
0097	La recepción del mensaje se ha finalizado debido a un tiempo de respuesta excedido.
0098	La recepción del mensaje se ha finalizado porque no se cumple la condición de longitud "N+LEN+M".
0099	La recepción del mensaje se ha finalizado porque se ha recibido la cadena de caracteres definida como condición de fin.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

RCV_RST: Borrar búfer de recepción

Descripción

La instrucción "RCV_RST" borra el búfer de recepción de un interlocutor.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RCV_RST":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Habilitación para borrar el búfer de recepción al detectarse un flanco ascendente
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

SGN_GET: Consultar señales RS-232

Descripción

La instrucción "SGN_GET" consulta el estado actual de varias señales de un módulo de comunicación RS-232.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SGN_GET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Habilitación de consulta al detectarse un flanco ascendente
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW)
NDR	Output	BOOL	I, Q, M, D, L	Se activa para un ciclo cuando hay datos nuevos preparados y la instrucción se ha ejecutado sin errores.
DTR	Output	BOOL	I, Q, M, D, L	Terminal de datos disponible, módulo listo
DSR	Output	BOOL	I, Q, M, D, L	Registro listo, interlocutor listo
RTS	Output	BOOL	I, Q, M, D, L	Solicitud de transmisión, módulo listo para transmitir
CTS	Output	BOOL	I, Q, M, D, L	Listo para transmitir; el interlocutor puede recibir datos (reacción a RTS = módulo ON).
DCD	Output	BOOL	I, Q, M, D, L	Detección de portadora de datos, nivel de señal de recepción
RING	Output	BOOL	I, Q, M, D, L	Indicación de timbre, señalización de una llamada entrante
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#....)	Descripción
80F0	El módulo de comunicación es un módulo RS485; no hay señales disponibles.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

SGN_SET: Activar señales RS-232

Descripción

La instrucción "SGN_SET" activa el estado de las señales de salida de un módulo de comunicación RS-232.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SGN_SET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Activa el proceso cuando se detecta un flanco ascendente. Valor inicial: FALSE
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación (ID HW) Valor inicial: FALSE
SIGNAL	Input	BYTE	I, Q, M, D, L o constante	Especifica las señales que deben activarse: • 01H = activar RTS • 02H = activar DTR • 04H = activar DSR Valor inicial: FALSE
RTS	Input	BOOL	I, Q, M, D, L o constante	Solicitud de transmisión, módulo listo para transmitir Valor inicial: FALSE
DTR	Input	BOOL	I, Q, M, D, L o constante	Terminal de datos disponible, módulo listo Valor inicial: FALSE
DSR	Input	BOOL	I, Q, M, D, L o constante	Registro listo (aplicable sólo a interfaces tipo DCE) Valor inicial: FALSE

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores Valor inicial: FALSE
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Sin errores • 1: Con errores Valor inicial: FALSE
STATUS	Output	WORD	I, Q, M, D, L	Estado de la instrucción Valor inicial: 0

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#....)	Descripción
80F0	El módulo de comunicación es un módulo RS485; no hay señales disponibles.
80F1	No es posible activar señales porque está activado el control de flujo por hardware.
80F2	La señal DSR no puede activarse porque el módulo es del tipo DTE.
80F3	La señal DTR no puede activarse porque el módulo es del tipo DCE.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Encontrará más información sobre los códigos de error generales de las instrucciones de comunicación en: "Información general sobre el estado de los bloques de comunicación (Página 5517)".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Información general sobre el estado de los bloques de comunicación

Información general sobre el estado de ejecución de los bloques de comunicación

La tabla siguiente muestra la información general que se muestra en el parámetro STATUS de los bloques de comunicación:

Código de error* (W#16#....)	Descripción
8070	Todas las memorias instancia internas están en uso
8080	La identificación introducida para el puerto de comunicación no es válida

Código de error* (W#16#....)	Descripción
8081	Tiempo excedido, error del módulo, error interno
8085	Error al indicar la longitud en el parámetro LENGHT. La longitud indicada es "0" o mayor que el valor máximo admisible.
8090	Longitud del mensaje no válida, módulo no válido, mensaje no válido
8091	Tipo erróneo en el mensaje de parametrización
8092	Longitud de registro no válido en el mensaje de parametrización
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Véase también".	

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

USS

Panorámica de las instrucciones USS

Introducción

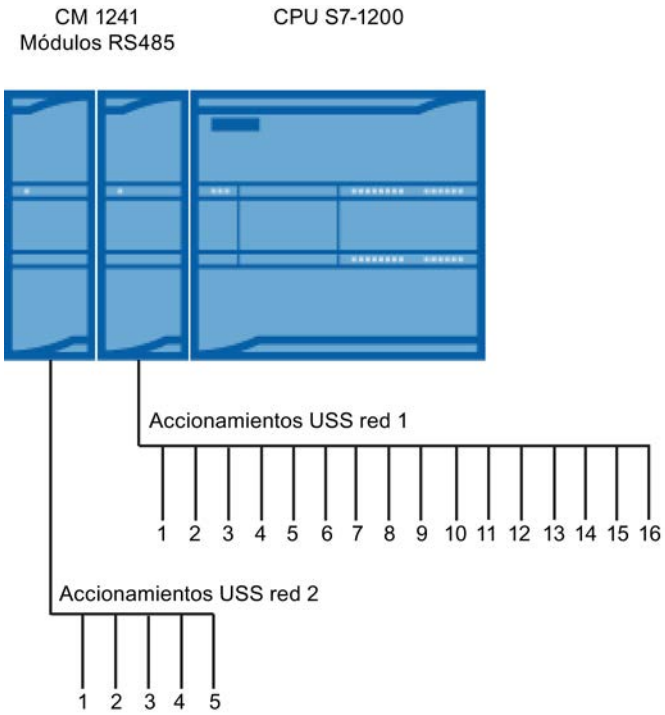
Las instrucciones USS controlan la operación de accionamientos compatibles con el protocolo de la interfaz serie universal (USS). Las instrucciones USS permiten que se comunique con varios accionamientos a través de una conexión RS-485.

Para ello necesita un módulo de comunicación RS485 CM 1241 o una tarjeta de comunicación RS485 CB 1241. En una CPU S7-1200 se pueden montar hasta tres módulos RS485 CM 1241 y una tarjeta RS485 CB 1241.

Cada puerto RS485 puede operar hasta dieciséis accionamientos.

El protocolo USS usa una red maestro/esclavo para la comunicación a través de un bus serie. El maestro emplea un parámetro de dirección para enviar un mensaje a un esclavo seleccionado. Un esclavo no puede efectuar un envío por iniciativa propia sin recibir previamente una solicitud al respecto. No es posible transmitir mensajes directamente entre los esclavos individuales. La comunicación USS funciona en modo semidúplex.

La figura siguiente muestra el ejemplo de un diagrama de red USS:



Requisitos para utilizar el protocolo USS

Requisitos generales para la configuración de accionamientos

- Para los accionamientos se debe configurar el uso de 4 palabras de identificador/valor del parámetro.
- Los accionamientos se pueden configurar para 2, 4, 6 u 8 palabras de datos de proceso.
- La cantidad de palabras de datos de proceso en el accionamiento debe corresponder a la entrada PZD_LEN de la instrucción "USS_DRIVE (Página 5524)" del accionamiento.
- La velocidad de transferencia de todos los accionamientos debe corresponder a la velocidad de transferencia en el parámetro de entrada BAUD de la instrucción "USS_PORT (Página 5523)".
- El accionamiento se debe configurar para mando a distancia.
- En la consigna de frecuencia de la conexión COM del accionamiento se debe determinar USS.
- En la dirección de accionamiento debe constar 1 a 16. Esta dirección debe corresponder a la dirección en el parámetro de entrada DRIVE de la instrucción "USS_DRIVE (Página 5524)".
- Para el control de dirección del accionamiento se debe configurar el uso de la polaridad de la consigna del accionamiento.
- La red RS485 debe disponer de una terminación correcta.

Definición: Área de identificador/valor del parámetro / área de datos de proceso

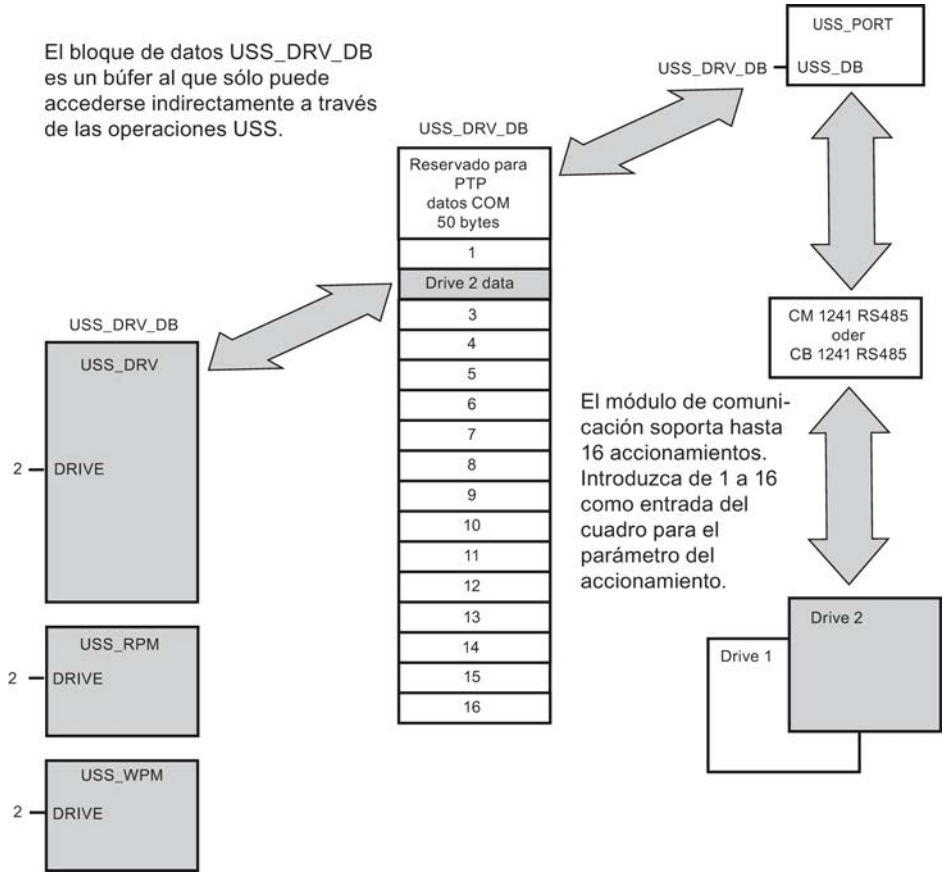
- El área de identificador/valor del parámetro hace referencia al tratamiento de la interfaz del identificador/valor del parámetro (PKW).
  La interfaz del identificador/valor del parámetro no es una interfaz física, sino que describe un mecanismo que regula el intercambio de parámetros entre dos interlocutores de comunicación. Es decir, lectura y escritura de valores de parámetros, descripciones de parámetros y textos correspondientes, así como el tratamiento de cambios de parámetros mediante mensajes espontáneos. Todas las tareas que se efectúan a través de la interfaz del identificador/valor del parámetro son principalmente tareas de manejo y observación, servicio y diagnóstico.
- El área de datos de proceso incluye las señales necesarias para la automatización:
  - Palabra(s) de mando y consigna(s) del maestro al esclavo
  - Palabra(s) de estado y valor(es) real(es) del esclavo al maestro.

Ambas áreas forman en conjunto el bloque de datos útiles. Éste se transmite como telegrama de petición del maestro al esclavo, o bien como telegrama de respuesta del esclavo al maestro.

Descripción

Cada módulo de comunicación CM 1241 RS485 soporta un máximo de 16 accionamientos. Un solo bloque de datos instancia contiene funciones temporales de memoria y búfer para todos los accionamientos de la red USS conectados con uno de los módulos de comunicación

PtP que se han instalado. Las instrucciones USS para este accionamiento tienen acceso común a la información de este bloque de datos.



- Todos los accionamientos (máx. 16) conectados a un puerto RS485 forman parte de la misma red USS. Todos los accionamientos conectados a otro puerto RS485 forman parte de otra red USS. Dado que el S7-1200 soporta hasta tres módulos CM 1241 RS485, es posible configurar un máximo de tres redes USS, cada una con un máximo de 16 accionamientos en cada red, de manera que en total se soportan 48 accionamientos USS.
- Cada red USS se administra a través de un bloque de datos unívoco (para tres redes USS con tres módulos CM 1241 RS485 se requieren tres bloques de datos). Todas las instrucciones que pertenecen a una red USS deben utilizar este bloque de datos común. Éste abarca todas las instrucciones "USS_DRIVE (Página 5524)", "USS_PORT (Página 5523)", "USS_RPM (Página 5527)" y "USS_WPM (Página 5529)" para controlar todos los accionamientos de una red USS.

- La instrucción "USS_DRIVE (Página 5524)" es un bloque de función (FB). Si se inserta la instrucción "USS_DRIVE" en el editor, en el cuadro de diálogo "Opciones de llamada" se debe asignar un DB a la instrucción.
  - Si se trata de la primera instrucción "USS_DRIVE" en este programa para esta red USS, se puede aplicar la asignación estándar de DB (o en caso necesario modificar el nombre) y se crea el nuevo DB.
  - Si, por el contrario, no es la primera instrucción "USS_DRIVE" para esta red, en la lista desplegable del cuadro de diálogo "Opciones de llamada" se debe seleccionar el correspondiente DB que ya se ha asignado previamente a esta red USS.
- Todas las instrucciones "USS_PORT (Página 5523), USS_RPM (Página 5527)" y "USS_WPM (Página 5529)" son funciones (FCs). Si estas funciones se insertan en el editor, no se asigna ningún DB. En su lugar se debe asignar la entrada USS_DB de estas instrucciones al DB correspondiente (haga doble clic en el campo del parámetro y, a continuación, en el símbolo para visualizar los DBs disponibles).
- La instrucción "USS_PORT (Página 5523)" controla la comunicación entre la CPU y los accionamientos a través del módulo de comunicación PtP. En cada llamada la instrucción establece una comunicación con un accionamiento. El programa debe llamar a esta función lo suficientemente rápido como para que el accionamiento no notifique ningún rebase de tiempo. La instrucción puede ser llamada desde el programa principal o desde un OB de alarma cualquiera.
- El bloque de función "USS_DRIVE (Página 5524)" le da al programa acceso a un accionamiento indicado en la red USS. Sus entradas y salidas corresponden a los estados y las funciones de manejo del accionamiento. Si existen 16 accionamientos en la red, "USS_DRIVE" debe ser llamado en el programa al menos 16 veces, es decir, una vez por cada accionamiento. La rapidez con la que se llama a los bloques depende de la velocidad necesaria para el control de las funciones de accionamiento.
  La instrucción "USS_DRIVE" sólo puede ser llamada desde el OB del programa principal.

  ⚠ **PRECAUCIÓN**

  Llame a "USS_DRIVE", "USS_RPM", "USS_WPM" sólo desde el OB del programa principal. La instrucción "USS_PORT" puede ser llamada desde un OB cualquiera, normalmente se llama desde un OB de alarma de retardo. Si la instrucción "USS_PORT" se interrumpe durante la ejecución, se puede producir un error inesperado.

Con las instrucciones "USS_RPM" y "USS_WPM" se leen y escriben los parámetros operativos del accionamiento. Dichos parámetros controlan el funcionamiento interno del accionamiento. En el manual del accionamiento encontrará la definición de estos parámetros.

El programa puede contener el número que se desee de estas funciones; no obstante, sólo puede haber una solicitud de lectura o escritura activa para un accionamiento. Las instrucciones "USS_RPM" y "USS_WPM" sólo pueden ser llamadas desde un OB del programa principal.

Calcular el tiempo para comunicarse con el accionamiento

La comunicación con el accionamiento es asíncrona con respecto al ciclo del S7-1200. El S7-1200 suele ejecutar varios ciclos antes de que haya finalizado la comunicación con un accionamiento.

El intervalo de "USS_PORT" es el tiempo necesario para una transacción del accionamiento. La tabla siguiente muestra los intervalos mínimos para "USS_PORT" con cada velocidad de transferencia. Si se llama la instrucción "USS_PORT" con más frecuencia de la predefinida en el intervalo "USS_PORT", el número de transacciones no se incrementa. El intervalo de timeout del accionamiento es el tiempo disponible para una transacción cuando se necesitan tres intentos para finalizar la transacción debido a errores de comunicación. Por defecto, en el protocolo USS se realizan en cada transacción hasta 2 repeticiones.

Velocidad de transferencia	Intervalo mínimo calculado para llamar USS_PORT (ms)	Timeout de intervalo para aviso de accionamiento por accionamiento (ms)
1200	790	2370
2400	405	1215
4800	212,5	638
9600	116,3	349
19200	68,2	205
38400	44,1	133
57600	36,1	109
115200	28,1	85

USS_PORT: Editar la comunicación vía red USS

Descripción

La instrucción "USS_PORT" edita la comunicación a través de la red USS. En el programa, utilice cada vez una instrucción "USS_PORT" por puerto de comunicación PtP para controlar la transferencia a o desde un accionamiento.

Todas las instrucciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo bloque de datos instancia.

Llamada

El programa debe ejecutar la instrucción "USS_PORT" lo bastante a menudo como para que no aparezca ningún rebase de tiempo en el accionamiento. Llame por tanto a la instrucción "USS_PORT" desde un OB de alarma cíclica para evitar rebases de tiempo en el accionamiento y tener disponibles las últimas actualizaciones de datos USS para llamadas de "USS_DRIVE (Página 5524)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "USS_PORT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PORT	Input	PORT	D, L o constante	Identificación del puerto de comunicación PtP Constante a la que puede señalarse en la tabla de variables estándar de la ficha "Constantes".
BAUD	Input	DINT	I, Q, M, D, L o constante	Velocidad de transferencia para la comunicación USS.
USS_DB	InOut	USS_BASE	D	Referencia al DB de instancia de la instrucción "USS_DRIVE (Página 5524)".
ERROR	Output	BOOL	I, Q, M, D, L	Si se presenta un error, ERROR se pone a TRUE. En la salida STATUS se visualiza un código de error correspondiente.
STATUS (Página 5530)	Output	WORD	I, Q, M, D, L	Valor de estado de la solicitud. Indica el resultado del ciclo o de la inicialización. Encontrará más información acerca de algunos códigos de estado en la variable "USS_Extended_Error (Página 5530)".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

USS_DRIVE: Intercambiar datos con el accionamiento

Descripción

La instrucción "USS_DRIVE" intercambia datos con el accionamiento creando avisos de solicitud y evaluando los avisos de respuesta del accionamiento. Para cada accionamiento debe utilizarse una instrucción propia, pero todas las instrucciones USS que están asignadas a una red USS y a un módulo de comunicación PtP deben utilizar el mismo bloque de datos instancia. Se debe introducir el nombre del DB al insertar la primera instrucción "USS_DRIVE". Después siga utilizando el DB que se ha creado al insertar la primera instrucción.

Cuando se produce la primera ejecución de "USS_DRIVE", se inicializa el accionamiento indicado en la dirección USS (parámetro DRIVE) en el DB de instancia. Después de la inicialización, las siguientes instrucciones "USS_PORT (Página 5523)" pueden iniciar la comunicación con el accionamiento en este número de accionamiento.

Si se modifica el número de accionamiento, el sistema de destino debe ponerse primero en STOP y después volver a RUN para que se inicialice el DB instancia. Los parámetros de entrada se configuran en el búfer de transmisión USS y las salidas, si existen, se leen desde un búfer de respuesta válido "anterior". Durante la ejecución de la instrucción "USS_DRIVE" no se realiza ninguna transferencia de datos. Tras la ejecución de "USS_PORT (Página 5523)" se comunica con los accionamientos. "USS_DRIVE" configura sólo los avisos que deben enviarse y evalúa los datos recibidos en una solicitud anterior.

Es posible controlar el sentido de giro del accionamiento con la entrada DIR (BOOL) o con el signo (positivo o negativo) de la entrada SPEED_SP (REAL). La tabla siguiente explica cómo funcionan las entradas de forma conjunta para determinar el sentido de giro del accionamiento, siempre que el motor gire hacia delante.

SPEED_SP	DIR	Sentido de giro del accionamiento
Valor > 0	0	Atrás
Valor > 0	1	Adelante
Valor < 0	0	Adelante
Valor < 0	1	Atrás

Parámetros

Amplíe el cuadro para visualizar todos los parámetros. Para ello haga clic en el área inferior del cuadro. Las conexiones de parámetros, que se representan en gris, son opcionales y no deben asignarse obligatoriamente.

La tabla siguiente muestra los parámetros de la instrucción "USS_DRIVE":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RUN	Input	BOOL	I, Q, M, D, L o constante	Bit de inicio del accionamiento: Si el parámetro tiene el valor TRUE, permite a la entrada utilizar el accionamiento con la velocidad predeterminada.
OFF2	Input	BOOL	I, Q, M, D, L o constante	Bit "Finalizar en parada": Si el parámetro tiene el valor FALSE, el bit inicia la parada del accionamiento sin frenar.
OFF3	Input	BOOL	I, Q, M, D, L o constante	Bit de parada rápida - Si el parámetro tiene el valor FALSE, el bit provoca una parada rápida frenando el accionamiento.
F_ACK	Input	BOOL	I, Q, M, D, L o constante	Bit de acuse de error - Con este bit se desactiva el bit de error de un accionamiento. Se activa después de borrar el error y así el accionamiento detecta que el error anterior no debe notificarse más.
DIR	Input	BOOL	I, Q, M, D, L o constante	Control de sentido del accionamiento - El bit se activa cuando el accionamiento debe funcionar hacia delante (si SPEED_SP es positivo).
DRIVE	Input	USINT	I, Q, M, D, L o constante	Dirección del accionamiento: esta entrada es la dirección del accionamiento USS. El rango válido está entre el accionamiento 1 y el accionamiento 16.
PZD_LEN	Input	USINT	I, Q, M, D, L o constante	Longitud de palabra - Es el número de palabras de datos PZD. Los valores válidos son 2, 4, 6 u 8 palabras. El valor predeterminado es 2.
SPEED_SP	Input	REAL	I, Q, M, D, L o constante	Consigna de velocidad - Es el porcentaje de la velocidad del accionamiento respecto a la frecuencia configurada. Un valor positivo significa que el accionamiento funciona hacia delante (si DIR tiene el valor TRUE).
CTRL3	Input	WORD	I, Q, M, D, L o constante	Palabra de control 3 - Valor que se escribe en un parámetro del accionamiento configurable por el usuario. El usuario debe configurarlo en el accionamiento. Parámetro opcional.
CTRL4	Input	WORD	I, Q, M, D, L o constante	Palabra de control 4 - Valor que se escribe en un parámetro del accionamiento configurable por el usuario. El usuario debe configurarlo en el accionamiento. Parámetro opcional.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CTRL5	Input	WORD	I, Q, M, D, L o constante	Palabra de control 5 - Valor que se escribe en un parámetro del accionamiento configurable por el usuario. El usuario debe configurarlo en el accionamiento. Parámetro opcional.
CTRL6	Input	WORD	I, Q, M, D, L o constante	Palabra de control 6 - Valor que se escribe en un parámetro del accionamiento configurable por el usuario. El usuario debe configurarlo en el accionamiento.
CTRL7	Input	WORD	I, Q, M, D, L o constante	Palabra de control 7 - Valor que se escribe en un parámetro del accionamiento configurable por el usuario. El usuario debe configurarlo en el accionamiento. Parámetro opcional.
CTRL8	Input	WORD	I, Q, M, D, L o constante	Palabra de control 8 - Valor que se escribe en un parámetro del accionamiento configurable por el usuario. El usuario debe configurarlo en el accionamiento. Parámetro opcional.
NDR	Output	BOOL	I, Q, M, D, L	Nuevos datos listos - Si el parámetro tiene el valor TRUE, el bit notifica que en la salida hay datos disponibles de una nueva solicitud de comunicación.
ERROR	Output	BOOL	I, Q, M, D, L	Con errores - Si el parámetro tiene el valor TRUE, el bit notifica que se ha producido un error y la salida STATUS es válida. Todas las demás salidas se ponen a cero en caso de error. Los errores de comunicación sólo se notifican en las salidas ERROR y STATUS de la instrucción "USS_PORT".
STATUS (Página 5530)	Output	WORD	I, Q, M, D, L	Valor de estado de la solicitud. Indica el resultado del ciclo. No es una palabra de estado emitida por el accionamiento.
RUN_EN	Output	BOOL	I, Q, M, D, L	Operación autorizada - El bit notifica si el accionamiento está funcionando.
D_DIR	Output	BOOL	I, Q, M, D, L	Sentido del accionamiento - El bit notifica si el accionamiento funciona hacia delante.
INHIBIT	Output	BOOL	I, Q, M, D, L	Accionamiento bloqueado - El bit notifica el estado del bit de bloqueo para el accionamiento.
FAULT	Output	BOOL	I, Q, M, D, L	Error de accionamiento - El bit notifica que se ha producido un error en el accionamiento. El usuario debe solucionar el fallo y activar el bit F_ACK para borrar este bit.
SPEED	Output	REAL	I, Q, M, D, L	Valor real de velocidad del accionamiento (valor escalado de la palabra de estado 2 del accionamiento) - El valor porcentual de la velocidad del accionamiento respecto a la velocidad configurada.
STATUS1	Output	WORD	I, Q, M, D, L	Palabra de estado 1 del accionamiento - El valor contiene bits de estado fijos de un accionamiento.
STATUS3	Output	WORD	I, Q, M, D, L	Palabra de estado 3 del accionamiento - El valor contiene una palabra de estado del accionamiento configurable por el usuario.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS4	Output	WORD	I, Q, M, D, L	Palabra de estado 4 del accionamiento - El valor contiene una palabra de estado del accionamiento configurable por el usuario.
STATUS5	Output	WORD	I, Q, M, D, L	Palabra de estado 5 del accionamiento - El valor contiene una palabra de estado del accionamiento configurable por el usuario.
STATUS6	Output	WORD	I, Q, M, D, L	Palabra de estado 6 del accionamiento - El valor contiene una palabra de estado del accionamiento configurable por el usuario.
STATUS7	Output	WORD	I, Q, M, D, L	Palabra de estado 7 del accionamiento - El valor contiene una palabra de estado del accionamiento configurable por el usuario.
STATUS8	Output	WORD	I, Q, M, D, L	Palabra de estado 8 del accionamiento - El valor contiene una palabra de estado del accionamiento configurable por el usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

USS_RPM: Leer parámetros del accionamiento

Descripción

La instrucción "USS_RPM" lee un parámetro del accionamiento. Todas las funciones USS que están asignadas a una red USS y a un módulo de comunicación PtP deben utilizar el mismo bloque de datos. "USS_RPM" debe ser llamado desde el OB del programa principal.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "USS_RPM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Solicitud de transmisión: si el parámetro tiene el valor TRUE, se requiere una nueva solicitud de lectura. Se ignora si la solicitud para este parámetro ya está fijada.
DRIVE	Input	USINT	I, Q, M, D, L o constante	Dirección del accionamiento: esta entrada es la dirección del accionamiento USS. El rango válido está entre el accionamiento 1 y el accionamiento 16.
PARAM	Input	UINT	I, Q, M, D, L o constante	Número de parámetro: la entrada indica qué parámetro de accionamiento se escribe. El rango de este parámetro es de 0 a 2047. Encontrará más información sobre el acceso a parámetros que excedan este rango en el manual del accionamiento.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
INDEX	Input	UINT	I, Q, M, D, L o constante	Índice de parámetro: la entrada indica en qué índice de parámetro del accionamiento se debe escribir. Es un valor de 16 bits, en el que el byte menos significativo es el valor de índice real, con un rango de 0 a 255. El accionamiento también puede utilizar el byte más significativo, que es específico del accionamiento. Encontrará más información en el manual del accionamiento.
USS_DB	InOut	USS_BASE	D	Referencia al bloque de datos de instancia que se crea e inicializa cuando se inserta una instrucción "USS_DRIVE" en el programa.
DONE	Output	BOOL	I, Q, M, D, L	Si el parámetro tiene el valor TRUE, fija en la salida VALUE el valor solicitado anteriormente para el parámetro de lectura. El bit se activa cuando la instrucción "USS_DRIVE" detecta la palabra de lectura del accionamiento. El bit se desactiva cuando: • Se solicitan los datos de respuesta a través de otra consulta "USS_RPM" o • Se ejecuta la segunda de las dos llamadas siguientes de "USS_DRIVE (Página 5524)"
ERROR	Output	BOOL	I, Q, M, D, L	Con errores - Si el parámetro tiene el valor TRUE, el bit notifica que se ha producido un error y la salida STATUS es válida. Todas las demás salidas se ponen a cero en caso de error. Los errores de comunicación sólo se notifican en las salidas ERROR y STATUS de la instrucción "USS_PORT (Página 5523)".
STATUS (Página 5530)	Output	WORD	I, Q, M, D, L	Es el valor de estado de la solicitud. Indica el resultado de la solicitud de lectura. Encontrará más información acerca de algunos códigos de estado en la variable "USS_Extended_Error (Página 5530)".
VALUE	Output	VARIANT	I, Q, M, D, L	Es el valor del parámetro que se ha leído y sólo es válido cuando el bit DONE tiene el valor TRUE.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

USS_WPM: Modificar parámetros en el accionamiento

Descripción

La instrucción "USS_WPM" modifica un parámetro en el accionamiento. Todas las funciones USS que están asignadas a una red USS y a un módulo de comunicación PtP deben utilizar el mismo bloque de datos. "USS_WPM" debe ser llamado desde el OB del programa principal.

Nota

Operaciones de escritura en EEPROM

No utilice excesivamente la operación de escritura en EEPROM. Minimice el número de operaciones de escritura en EEPROM para prolongar la vida útil de EEPROM.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "USS_WPM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Solicitud de transmisión: Si el parámetro tiene el valor TRUE, se requiere una nueva solicitud de escritura. Se ignora si la solicitud para este parámetro ya está fijada.
DRIVE	Input	USINT	I, Q, M, D, L o constante	Dirección del accionamiento: esta entrada es la dirección del accionamiento USS. El rango válido está entre el accionamiento 1 y el accionamiento 16.
PARAM	Input	UINT	I, Q, M, D, L o constante	Número de parámetro: la entrada indica qué parámetro de accionamiento se escribe. El rango de este parámetro es de 0 a 2047. Encontrará más información sobre el acceso a parámetros que excedan este rango en el manual del accionamiento.
INDEX	Input	UINT	I, Q, M, D, L o constante	Índice de parámetro: la entrada indica en qué índice de parámetro del accionamiento se debe escribir. Es un valor de 16 bits, en el que el byte menos significativo es el valor de índice real, con un rango de 0 a 255. El accionamiento también puede utilizar el byte más significativo, que es específico del accionamiento. Encontrará más información en el manual del accionamiento.
EEPROM	Input	BOOL	I, Q, M, D, L o constante	Guardar en EEPROM del accionamiento: si el parámetro tiene el valor TRUE, los valores escritos en el parámetro del accionamiento se guardan en la EEPROM del accionamiento. Si el parámetro tiene el valor FALSE, el valor escrito sólo se guarda temporalmente y se pierde en la siguiente conexión del accionamiento.
VALUE	Input	VARIANT	I, Q, M, D, L o constante	Valor del parámetro en el que se debe escribir. Debe ser válido en el cambio de estado de REQ.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
USS_DB	InOut	USS_BASE	D	Es una referencia al DB de instancia que se crea e inicializa cuando se inserta una instrucción "USS_DRIVE (Página 5524)" en el programa.
DONE	Output	BOOL	I, Q, M, D, L	Si el parámetro tiene el valor TRUE, la entrada VALUE se ha escrito en el accionamiento. El bit se activa cuando la instrucción "USS_DRIVE (Página 5524)" detecta la respuesta de escritura del accionamiento. El bit se desactiva cuando: Se solicita que el accionamiento confirme a través de otra consulta "USS_WPM" que se ha realizado la escritura, o cuando se ejecuta la segunda de las dos llamadas siguientes de "USS_DRIVE (Página 5524)".
ERROR	Output	BOOL	I, Q, M, D, L	Con errores: si el parámetro tiene el valor TRUE, se ha producido un error y se muestra en la salida STATUS. Todas las demás salidas se ponen a cero en caso de error. Los errores de comunicación sólo se notifican en las salidas ERROR y STATUS de la instrucción "USS_PORT (Página 5523)".
STATUS (Página 5530)	Output	WORD	I, Q, M, D, L	Es el valor de estado de la solicitud. Indica el resultado de la solicitud de escritura. Encontrará más información acerca de algunos códigos de estado en la variable "USS_Extended_Error (Página 5530)".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS de las instrucciones USS

Parámetro STATUS

La tabla siguiente contiene los códigos de estado de la operación USS que se indican en la salida STATUS de las instrucciones USS.

STATUS* (W#16#....)	Descripción
0000	Sin errores
8180	La longitud de la respuesta del accionamiento no se corresponde con los caracteres recibidos en el accionamiento. El número del accionamiento en el que se ha producido el error se visualiza en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
8181	El parámetro VALUE no es del tipo de datos WORD, REAL o DWORD
8182	El usuario ha introducido un valor de parámetro del tipo palabra y ha recibido la respuesta del accionamiento en el formato DWORD o REAL
8183	El usuario ha introducido un valor de parámetro del tipo DWORD o REAL y ha recibido la respuesta del accionamiento en el formato de palabra
8184	El telegrama de respuesta del accionamiento tenía una suma de verificación errónea. El número del accionamiento en el que se ha producido el error se visualiza en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.

STATUS* (W#16#....)	Descripción
8185	Dirección de accionamiento no admisible (área de direcciones válida para accionamientos: 1-16)
8186	La consigna de velocidad se encuentra fuera del rango válido (rango de consigna válido para la velocidad: de -200% a 200%)
8187	Un número de accionamiento erróneo ha respondido a la solicitud enviada. El número del accionamiento en el que se ha producido el error se visualiza en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
8188	Longitud de palabra PZD no admisible (rango admisible = 2, 4, 6 u 8 palabras)
8189	Velocidad de transferencia no admisible
818A	El canal de solicitud para el parámetro es utilizado por otra solicitud para este accionamiento
818B	El accionamiento no ha reaccionado a las solicitudes y repeticiones. El número del accionamiento en el que se ha producido el error se visualiza en la variable "USS_Extended_Error". La descripción de los errores avanzados aparece a continuación de esta tabla.
818C	El accionamiento ha emitido un error avanzado relativo a una solicitud de parámetros. La descripción de los errores avanzados aparece a continuación de esta tabla.
818D	El accionamiento ha emitido un error de acceso no permitido en una solicitud de parámetros. Encontrará más información sobre la limitación de acceso a los parámetros en el manual del accionamiento.
818E	El accionamiento no se ha inicializado: Este código de error se emite en "USS_RPM (Página 5527)" o "USS_WPM (Página 5529)" cuando la instrucción "USS_DRIVE (Página 5524)" no se ha llamado al menos una vez para este accionamiento. De esta manera se impide que la inicialización de "USS_DRIVE (Página 5524)" en el primer ciclo sobrescriba una solicitud pendiente de lectura o escritura de los parámetros, puesto que inicializa el accionamiento como entrada nueva. Para solucionar este error, llame a la instrucción "USS_DRIVE (Página 5524)" para este accionamiento.
80Ax-80Fx	Errores específicos que devuelven las instrucciones llamadas por la librería USS para la comunicación punto a punto (PtP): estos códigos de error no se modifican en la librería USS y se definen en las descripciones de las instrucciones PtP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Véase también".	

USS_Extended_Error - Códigos de error adicionales de los accionamientos USS

Los accionamientos USS soportan accesos de lectura y escritura a los parámetros internos de un accionamiento. Esta función permite controlar y configurar el accionamiento de forma descentralizada. Los accesos a los parámetros del accionamiento pueden fallar debido a errores como valores fuera de rango o solicitudes no admisibles en el modo de operación actual del accionamiento. El accionamiento crea un código de error que se indica en la variable "USS_Extended_Error" del DB de instancia de la instrucción "USS_DRIVE (Página 5524)". Este código de error sólo sirve para la última ejecución de la instrucción "USS_RPM (Página 5527)" o "USS_WPM (Página 5529)". El código de error del accionamiento se deposita en la variable "USS_Extended_Error" cuando STATUS tiene el valor hexadecimal 818C. El código de error de "USS_Extended_Error" se rige por la variante del accionamiento. Encontrará una descripción de los códigos de error avanzados para funciones de lectura y escritura de parámetros en el manual del accionamiento.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

MODBUS (RTU)

MB_COMM_LOAD: Configurar puerto en módulo PtP para Modbus-RTU

Descripción

La instrucción "MB_COMM_LOAD" configura un puerto para la comunicación mediante el protocolo Modbus RTU. Para ello se puede utilizar el hardware siguiente:

- Hasta tres módulos punto a punto (PtP) CM 1241 RS485 o CM 1241 RS232
- Además, una tarjeta de comunicación CB 1241 RS485

Después de configurar el puerto la comunicación se realiza a través de Modbus ejecutando la instrucción "MB_SLAVE" o "MB_MASTER".

Llamada

"MB_COMM_LOAD" debe ser llamado una vez para configurar el puerto para el protocolo Modbus-RTU. Tras terminar la configuración, el puerto puede ser utilizado por las instrucciones "MB_MASTER (Página 5535)" y "MB_SLAVE (Página 5543)".

"MB_COMM_LOAD" sólo debe volver a llamarse si hay que cambiar un parámetro de comunicación. Cada llamada de "MB_COMM_LOAD" borra el búfer de comunicación. Para evitar la pérdida de datos en la comunicación, no efectúe llamadas innecesarias de la instrucción.

Para configurar el puerto de cada módulo de comunicación que se usa para la comunicación Modbus, se debe utilizar una instancia de "MB_COMM_LOAD". Se debe asignar a cada puerto utilizado un bloque de datos instancia "MB_COMM_LOAD" unívoco. La CPU S7-1200 está limitada a tres módulos de comunicación.

Cuando se insertan las instrucciones "MB_MASTER (Página 5535)" o "MB_SLAVE (Página 5543)", se asigna un bloque de datos instancia. Dicho bloque se referencia indicándolo en el parámetro MB_DB de la instrucción "MB_COMM_LOAD".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MB_COMM_LOAD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Ejecución de la instrucción con flanco ascendente.
PORT	Input	PORT	I, Q, M, D, L o constante	Identificación del puerto de comunicación: Tras haber insertado el módulo de comunicación en la configuración de dispositivos, la identificación de puerto aparece en la lista desplegable de la conexión del cuadro PORT. Esta constante también se puede referenciar en la tabla de variables de la ficha "Constantes".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BAUD	Input	UDINT	I, Q, M, D, L o constante	Selección de la velocidad de transferencia: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 Todos los demás valores no son válidos.
PARITY	Input	UINT	I, Q, M, D, L o constante	Selección de la paridad: • 0 – Ninguna • 1 – Impar • 2 – Par
FLOW_CTRL	Input	UINT	I, Q, M, D, L o constante	Selección del control de flujo: • 0 – (ajuste predeterminado) Sin control de flujo • 1 – Control de flujo por hardware con RTS siempre ON (no se aplica a los puertos RS485) • 2 - Control de flujo por hardware con RTS conectado
RTS_ON_DLY	Input	UINT	I, Q, M, D, L o constante	Selección de retardo a la conexión de RTS: • 0 – (ajuste predeterminado) Sin retardo de "RTS activo" hasta que se envía el primer carácter del aviso. • de 1 a 65535 – Retardo en milisegundos de "RTS activo" hasta que se envía el primer carácter del aviso (no se aplica a los puertos RS-485). Los retardos de RTS deben aplicarse independientemente de la selección de FLOW_CTRL.
RTS_OFF_DLY	Input	UINT	I, Q, M, D, L o constante	Selección de retardo a la desconexión de RTS: • 0 – (ajuste predeterminado) Sin retardo después de la transferencia del último carácter hasta "RTS inactivo" • de 1 a 65535 – Retardo en milisegundos desde la transferencia del último carácter hasta "RTS inactivo" (no se aplica a los puertos RS-485). Los retardos de RTS deben aplicarse independientemente de la selección de FLOW_CTRL.
RESP_TO	Input	UINT	I, Q, M, D, L o constante	Timeout de respuesta: Tiempo en milisegundos que "MB_MASTER (Página 5535)" espera una respuesta del esclavo. Si el esclavo no responde en este tiempo, "MB_MASTER (Página 5535)" repite la solicitud o la finaliza con un error, siempre que se haya enviado el número de repeticiones indicado. de 5 ms a 65535 ms (valor estándar = 1000 ms).
MB_DB	Input	MB_BASE	D	Una referencia al bloque de datos instancia de las instrucciones "MB_MASTER (Página 5535)" o "MB_SLAVE (Página 5543)". Después de haber insertado "MB_SLAVE (Página 5543)" o "MB_MASTER (Página 5535)" en el programa, la identificación de DB está disponible en la lista desplegable en la conexión del cuadro MB_DB.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DONE	Output	BOOL	I, Q, M, D, L	Ejecución de la instrucción finalizada sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Error: • 0 – No se ha detectado ningún error • 1 – Indica que se ha detectado un error. En el parámetro STATUS se devuelve un código de error.
STATUS	Output	WORD	I, Q, M, D, L	Código de error de la configuración del puerto

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Código de error* (W#16#....)	Descripción
0000	Ningún error
8180	Valor no válido para la ID de puerto (dirección errónea del módulo de comunicación).
8181	Valor no válido para la velocidad de transferencia.
8182	Valor no válido para la paridad.
8183	Valor no válido para el control de flujo.
8184	Valor no válido para el tiempo excedido de la respuesta (el lapso de tiempo hasta que se comunica el tiempo excedido debe ser como mínimo 25 ms).
8185	Puntero incorrecto en el parámetro MB_DB hacia el DB de instancia de la instrucción "MB_MASTER (Página 5535)" o "MB_SLAVE (Página 5543)".
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Variables de bloques de datos MB_COMM_LOAD

La siguiente tabla muestra las variables estáticas públicas del DB de instancia de MB_COMM_LOAD que puede utilizar en su programa.

Tabla 9-162 Variables estáticas del DB de instancia

Variable	Tipo de datos	Ajuste predeterminado	Descripción
ICHAR_GAP	WORD	0	Retardo para la distancia entre caracteres. Este parámetro se indica en milisegundos y sirve para aumentar el intervalo de tiempo esperado entre los caracteres recibidos. La cantidad de tiempos de bit correspondiente para este parámetro se suma al valor estándar de Modbus de 35 tiempos de bit (3,5 tiempos de carácter).
RETRIES	WORD	2	La cantidad de intentos repetidos que ejecuta el maestro antes de que se devuelva el código de error 0x80C8 para "Sin respuesta".

Variable	Tipo de datos	Ajuste predeterminado	Descripción
MODE	USINT	0	Modo de operación Los modos de operación permitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) , modo a 4 hilos (punto a punto) • 2 = dúplex (RS422), modo a 4 hilos (maestro multipunto) • 3 = dúplex (RS422), modo a 4 hilos (esclavo multipunto) • 4 = semidúplex (RS485), modo a 2 hilos
LINE_PRE	USINT	0	Ocupación estándar de la línea de recepción Ocupaciones permitidas: • 0 = "sin" ocupación • 1 = señal R(A)=5 V, señal R(B)=0 V (detección break): Con esta ocupación no es posible la detección break. Solo se puede seleccionar con: "Dúplex (RS422), modo a 4 hilos (acoplamiento punto a punto)" y "dúplex (RS422), modo a 4 hilos (esclavo multipunto)". • 2 = señal R(A)=0 V, señal R(B)=5 V: Esta ocupación equivale al estado de reposo (no hay ninguna emisión activa). Con esta ocupación no es posible la detección de break.
CABLE-BREAK	USINT	0	Activar la detección de rotura de hilo: • 0 - no activada • 1 - activada

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

MB_MASTER: Comunicar como maestro Modbus vía puerto PtP

Descripción de MB_MASTER

Descripción

La instrucción "MB_MASTER" permite al programa comunicarse como maestro Modbus a través del puerto de un módulo punto a punto (CM) o una tarjeta de comunicación (CB). Es posible acceder a los datos de uno o varios dispositivos esclavo Modbus.

Para que la instrucción "MB_MASTER" pueda comunicarse con un puerto, se debe ejecutar previamente "MB_COMM_LOAD (Página 5532)".

Cuando se inserta la instrucción "MB_MASTER" en el programa, se crea un DB de instancia. Introduzca este DB de instancia en el parámetro de entrada MB_DB de la instrucción "MB_COMM_LOAD (Página 5532)".

Reglas para la comunicación del maestro Modbus

- Un puerto utilizado para las solicitudes del maestro Modbus no se puede usar para "MB_SLAVE".
- Un puerto se puede usar para una o varias llamadas de "MB_MASTER" si se utiliza el mismo DB de instancia.
- Las operaciones Modbus no utilizan eventos de alarma de comunicación para controlar el proceso de comunicación. El programa debe consultar la instrucción "MB_MASTER" en procedimientos de transmisión y recepción finalizados.
- Llamada de la instrucción:
  - Si resulta posible, efectúe la llamada de la instrucción "MB_MASTER" en un OB de programa cíclico. La instrucción también se puede llamar en un OB de retardo o de alarma cíclica.
  - No llame varias instrucciones "MB_MASTER" en bloques de organización con clases de prioridad distintas. En caso de ejecución anticipada de una instrucción "MB_MASTER" desde una clase de prioridad superior, la ejecución puede resultar errónea.
  - No llame instrucciones "MB_MASTER" en un OB de arranque, diagnóstico o error de tiempo.
- Tras el comienzo de una transmisión, en el parámetro EN (KOP/FUP) se debe presentar transitoriamente el valor "1" hasta que la instrucción ponga a "1" el parámetro de salida DONE o ERROR. Una nueva llamada a través del parámetro REQ mientras la instrucción se está ejecutando provoca un error. Después de ejecutar la instrucción, el bit del parámetro REQ permanece activado por el tiempo definido en el parámetro BLOCKED_PROC_TIMEOUT del DB de instancia.
- Si "MB_MASTER" envía una solicitud a un esclavo, se debe proseguir la ejecución de "MB_MASTER" hasta que la respuesta del esclavo haya llegado.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MB_MASTER":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ (Página 5539)	Input	BOOL	I, Q, M, D, L	Entrada de solicitud: • 0 – Ninguna solicitud • 1 – Solicitud de enviar datos a esclavo(s) Modbus
MB_ADDR	Input	UINT	I, Q, M, D, L o constante	Dirección de estación Modbus RTU: • Área de direccionamiento estándar: 0 a 247 • Área de direccionamiento ampliada: 0 a 65535 El valor "0" está reservado para la difusión general de un aviso a todos los esclavos Modbus. Para el broadcast sólo se soportan los códigos de función Modbus 05, 06, 15 y 16.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE (Página 5539)	Input	USINT	I, Q, M, D, L o constante	Modo de selección: Indica el tipo de solicitud: lectura, escritura o diagnóstico En la tabla de funciones Modbus encontrará más detalles.
DATA_ADDR (Página 5539)	Input	UDINT	I, Q, M, D, L o constante	Dirección inicial en el esclavo: indica la dirección inicial de los datos a los que se debe acceder en el esclavo Modbus. Las direcciones válidas se pueden encontrar en la tabla de funciones Modbus.
DATA_LEN	Input	UINT	I, Q, M, D, L o constante	Longitud de datos: indica el número de bits o palabras a los que debe acceder esta solicitud. Las longitudes válidas se pueden encontrar en la tabla de funciones Modbus.
DATA_PTR (Página 5540)	Input	VARIANT	M, D	Puntero hacia la dirección del DB o de la marca de la CPU de los datos que se deben escribir o leer. En un DB, éste se debe crear con el tipo de acceso "Estándar: compatible con S7-300/400".
DONE	Output	BOOL	I, Q, M, D, L	• 0: Transacción no finalizada • 1: Transacción finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	• 0: No se está ejecutando ninguna transacción de "MB_MASTER" • 1: Transacción de "MB_MASTER" en ejecución
ERROR	Output	BOOL	I, Q, M, D, L	• 0: Ningún error • 1: Error; el código de error se visualiza en el parámetro STATUS
STATUS	Output	WORD	I, Q, M, D, L	Condición de ejecución

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Tabla 9-163 Mensajes de error de comunicación y de error de configuración de la instrucción

Código de error* (W#16#....)	Descripción
0000	Ningún error
80C8	Timeout del esclavo. Compruebe la velocidad de transferencia, la paridad y las conexiones del esclavo.
80D1	El receptor ha solicitado un control de flujo para interrumpir una transmisión en curso, pero la transmisión no se ha vuelto a activar en el tiempo de espera. Este error también se crea en el control de flujo por hardware cuando el receptor no detecta ningún CTS en el tiempo de espera.
80D2	La solicitud de transmisión se ha cancelado, porque no se recibe ninguna señal DSR del DCE.
80E0	El aviso se ha finalizado porque el búfer de recepción está lleno.
80E1	El aviso se ha finalizado por un error de paridad.
80E2	El aviso se ha finalizado por un error de telegrama.

Código de error* (W#16#....)	Descripción
80E3	El aviso se ha finalizado por un error de desbordamiento.
80E4	El aviso se ha cancelado, porque la longitud indicada excede el tamaño del búfer total.
8180	Valor no válido para la ID de puerto.
8186	Dirección de estación Modbus no válida
8188	Valor no válido en el parámetro MODE para una llamada de difusión general.
8189	Valor no válido para la dirección de datos.
818A	Valor no válido para la longitud de datos.
818B	Puntero no válido hacia el origen o destino de datos local: tamaño incorrecto
818C	Puntero no válido en el parámetro DATA_PTR. Utilice un puntero hacia un área de marcas o un DB con el tipo de acceso "Estándar: compatible con S7-300/400".
8200	El puerto está ocupado procesando una solicitud de transmisión
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Véase también".	

Tabla 9-164 Mensajes de error del protocolo Modbus

Código de error* (W#16#....)	Código de respuesta del esclavo	Descripción
8380	-	Error CRC
8381	01	Código de función no soportado
8382	03	Error en la longitud de datos
8383	02	Error en la dirección de los datos o dirección fuera del rango válido de DATA_PTR
8384	> 03	Error en el valor de datos
8385	03	Valor del código de diagnóstico de datos no soportado (código de función 08)
8386	-	El código de función de la respuesta no concuerda con el código de función de la consulta.
8387	-	Respuesta de un esclavo equivocado
8388	-	La respuesta del esclavo a una llamada de escritura no es correcta. Los datos enviados por el esclavo no concuerdan con la consulta del maestro.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Véase también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

Parámetro REQ

Descripción

- REQ = FALSE: ninguna solicitud
- REQ = TRUE: solicitud de enviar datos a esclavo(s) Modbus

Esta entrada se puede controlar mediante un contacto controlado por nivel o flanco.

Siempre que se active esta entrada, se iniciará una máquina de estado para garantizar que otra instrucción "MB_MASTER" utilice los mismos DB de instancia y que no se puede emitir una petición hasta que la actual no se haya finalizado. Los restantes estados de entrada se registran y se guardan internamente para la petición actual hasta que se reciba respuesta o se detecte un error.

Si la misma instancia de "MB_MASTER" se vuelve a ejecutar con la entrada REQ = 1 mientras no se haya procesado la petición actual, no se realizarán más transferencias. Sin embargo, si la petición se ha completado, en cuanto "MB_MASTER" se vuelva a ejecutar con la entrada REQ = 1, se lanzará una nueva solicitud.

Parámetros DATA_ADDR y MODE

Descripción

Con el parámetro DATA_ADDR se especifica la dirección inicial para el acceso a los datos en el esclavo Modbus.

Con el parámetro MODE y la dirección Modbus se determina el código de función que se transfiere al esclavo Modbus. La tabla siguiente muestra la relación entre el parámetro MODE, el código de función y el área de direccionamiento Modbus.

MODE	Función Modbus	Longitud de datos	Operación y datos	Dirección Modbus
0	01	de 1 a 2000 1 a 1992 (1)	Leer bits de salida: 1 a (1992 ó 2000) bits por consulta	de 1 a 9999
0	02	de 1 a 2000 1 a 1992 (1)	Leer bits de entrada: 1 a (1992 ó 2000) bits por consulta	10001 a 19999
0	03	de 1 a 125 1 a 124 (1)	Leer registro de parada: 1 a (124 ó 125) WORD por consulta	40001 a 49999 ó de 400001 a 465535
0	04	de 1 a 125 1 a 124 (1)	Leer WORD de entrada: 1 a (124 ó 125) WORD por consulta	30001 a 39999
1	05	1	Escribir un bit de salida: Un bit por consulta	de 1 a 9999
1	06	1	Escribir un registro de parada: 1 WORD por consulta	40001 a 49999 ó de 400001 a 465535
1	15	de 2 a 1968 1 a 1960 (2)	Escribir varios bits de salida: 2 a (1960 ó 1968) bits por consulta	de 1 a 9999

MODE	Función Modbus	Longitud de datos	Operación y datos	Dirección Modbus
1	16	de 2 a 123 1 a 122 (2)	Escribir varios registros de parada: 2 a (122 ó 123) WORD por consulta	40001 a 49999 ó de 400001 a 465535
2	15	de 1 a 1968 1 a 1960 (2)	Escribir uno o varios bits de salida: 1 a (1960 ó 1968) bits por consulta	de 1 a 9999
2	16	de 1 a 123 1 a 122 (2)	Escribir uno o varios registros de parada: 1 a (122 ó 123) WORD por consulta	40001 a 49999 ó de 400001 a 465535
11	11	0	Lectura de la palabra de estado de comunicación del esclavo y del contador de eventos: La palabra de estado indica la ejecución de la instrucción (0: no se ejecuta; 0xFFFF: se ejecuta). El contador de eventos se incrementa en un mensaje cada vez que se efectúa correctamente una transmisión. Los parámetros DATA_ADDR y DATA_LEN de la instrucción "MB_MASTER" son ignorados en esta función.	-
80	08	1	Compruebe el estado del esclavo leyendo el código de error (0x0000): 1 WORD por consulta	-
81	08	1	Resetee el contador del esclavo mediante el código de diagnóstico 0x000A: 1 WORD por consulta	-
3 a 10, 12 a 79, 82 a 2555			Reservada	-
(1) Para el modo "Área de direccionamiento ampliada", la longitud máxima de datos se reduce en un byte o un WORD, según el tipo de datos que se utilice para la función.				

Parámetro DATA_PTR

Descripción

El parámetro DATA_PTR es un puntero a un bloque de datos o a una marca desde los que se escriben o se leen datos. Si utiliza un bloque de datos, cree un bloque de datos global con el tipo de acceso "Estándar: compatible con S7-300/400".

Estructuras del bloque de datos para el parámetro DATA_PTR

- Estos tipos de datos sirven para la **lectura de palabras** de las direcciones Modbus 30001 a 39999, 40001 a 49999 y 400001 a 465536, así como para la **escritura de palabras** en las direcciones Modbus 40001 a 49999 y 400001 a 465536.
  - Campo estándar de tipos de datos WORD, UINT o INT (véase abajo).
  - Estructura con nombres del tipo WORD, UINT o INT, en la que cada elemento tiene un nombre unívoco y un tipo de datos de 16 bits.
  - Estructura compleja con nombres, en la que cada elemento tiene un nombre unívoco y un tipo de datos de 16 ó 32 bits.
- Para la lectura y escritura de bits de las direcciones Modbus 00001 a 09999 y 10001 a 19999.
  - Campo estándar de tipos de datos booleanos.
  - Estructura booleana con nombres a partir de variables booleanas con nombres unívocos.
- No es imprescindible, pero sí recomendable, que cada instrucción "MB_MASTER" tenga un área de memoria propia separada en un bloque de datos global. El motivo es que la posibilidad de que los datos se corrompan aumenta cuando varias instrucciones "MB_MASTER" leen y escriben en la misma área de un bloque de datos global.
- No es necesario que las áreas de memoria para DATA_PTR se encuentren en el mismo bloque de datos global. Se puede crear un bloque de datos con varias áreas para lecturas Modbus, un bloque de datos para escrituras Modbus o un bloque de datos para cada estación esclava.

DB de instancia de la instrucción "MB_MASTER"

Variables estáticas del DB de instancia

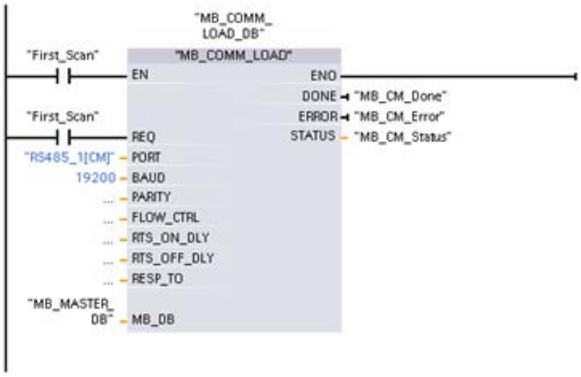
La tabla siguiente describe las variables estáticas del DB de instancia de la instrucción que se pueden utilizar en el programa de usuario.

Variable	Tipo de datos	Descripción
MB_STATE	UINT	Estado interno de la instrucción Modbus.
BLOCKED_ PROC_TIMEOUT	REAL	Intervalo de tiempo tras el final de la llamada de la instrucción y reseteo del bit ACTIVE en el DB de instancia. El búfer de tiempo se usa para evitar que la ejecución de la instrucción termine antes de haber enviado por completo una petición. El intervalo de tiempo preajustado es 500 ms.
EXTENDED_ ADDRESSING	BOOL	Configuración del direccionamiento: • 0: Área de direccionamiento estándar (1 byte) • 1: Área de direccionamiento ampliada (2 bytes) En la sección siguiente puede encontrar más información sobre la variable EXTENDED_ADDRESSING. DB de instancia de la instrucción "MB_SLAVE" (Página 5547)

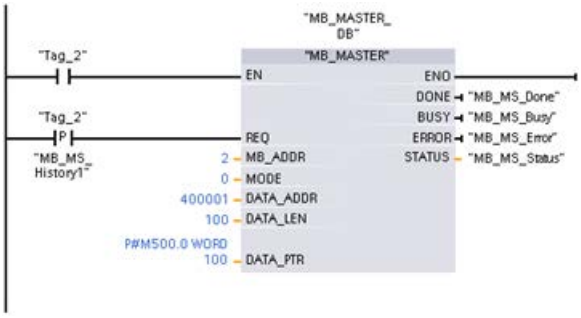
Programa de ejemplo para un maestro Modbus

Segmentos (KOP)

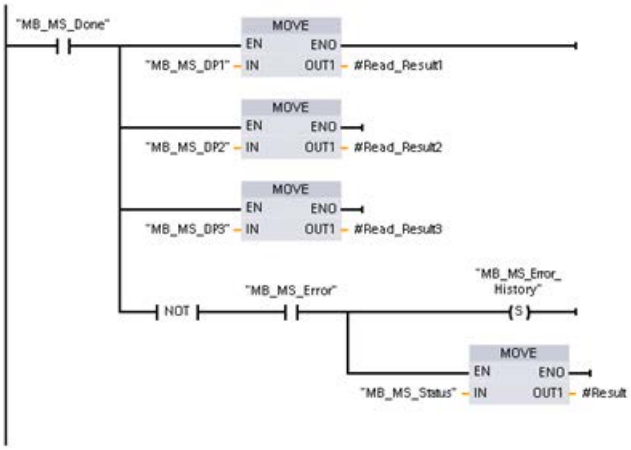
Segmento 1: Los parámetros del módulo RS-485 se deben inicializar sólo una vez durante el primer ciclo.



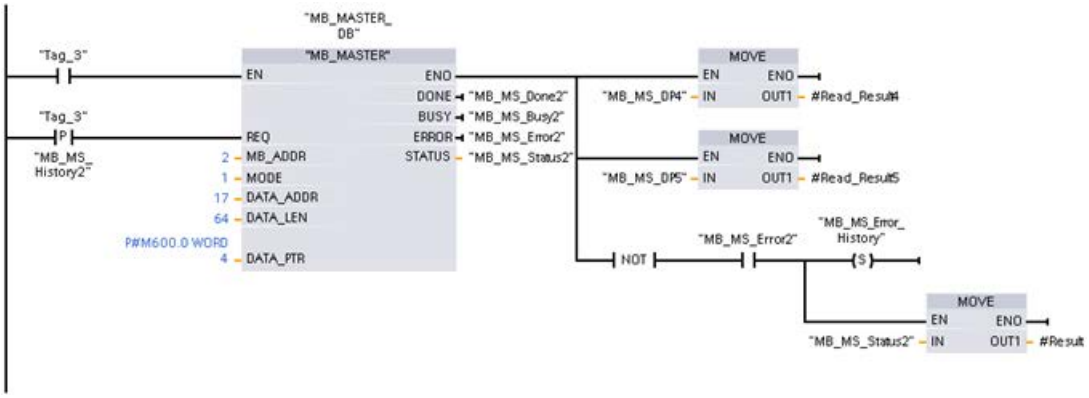
Segmento 2: Leer 100 palabras del registro de parada del esclavo.



Segmento 3: Éste es un segmento opcional que muestra los valores de las 3 primeras palabras si la operación de lectura se ha efectuado.



Segmento 4: Escribir 64 bits en la memoria imagen de proceso de las salidas, inicio en la dirección del esclavo A2.0.



MB_SLAVE: Comunicar como esclavo Modbus vía puerto PtP

Descripción de MB_SLAVE

Descripción

La instrucción "MB_SLAVE" permite al programa comunicarse como esclavo Modbus a través del puerto de un módulo punto a punto (PtP) o una tarjeta de comunicación (CB). Un maestro Modbus RTU puede enviar una solicitud y el programa responde ejecutando "MB_SLAVE".

Cuando se inserta la instrucción "MB_SLAVE" en el programa, es preciso asignar un bloque de datos instancia unívoco. Este bloque de datos de instancia se utiliza cuando se especifica en el parámetro MB_DB de la instrucción "MB_COMM_LOAD (Página 5532)".

Los códigos de función de la comunicación Modbus (códigos de función 1, 2, 4, 5 y 15) pueden leer y escribir bits y palabras directamente en la memoria imagen de proceso de las entradas y de las salidas del sistema de destino. La tabla siguiente muestra la asignación de las direcciones Modbus en la memoria imagen de proceso de la CPU.

Funciones Modbus de "MB_SLAVE"						S7-1200	
Códigos	Función	Área de datos	Área de direcciones			Área de datos	Dirección de la CPU
01	Leer bits	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7
02	Leer bits	Entrada	10001	a	18192	Memoria imagen de proceso de las entradas	I0.0 a I1023.7
04	Leer palabras	Entrada	30001	a	30512	Memoria imagen de proceso de las entradas	IW0 a IW1022
05	Escribir bit	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7
15	Escribir bits	Salida	1	a	8192	Memoria imagen de proceso de las salidas	Q0.0 a Q1023.7

Los códigos de función de la comunicación Modbus (códigos de función 3, 6, 16) utilizan un registro de parada propio. Para ello se puede utilizar una marca o un bloque de datos con el tipo de acceso "Estándar: compatible con S7-300/400".

El tipo de registro de parada se determina mediante el parámetro MB_HOLD_REG de la instrucción "MB_SLAVE". La tabla siguiente muestra la asignación del registro de parada Modbus a la dirección DB de MB_HOLD_REG en el sistema destino.

Funciones Modbus de "MB_SLAVE"				S7-1200	
Códigos	Función	Área de datos	Área de direcciones (Número WORD)	Dirección en el DB (Número BYTE)	Dirección de marca (Número BYTE)
03	Leer palabras	Registros de parada	40001 a 49999 ó	DW0 a DW19998 o bien	MW0 a límite de CPU
			400001 a 465535	DW0 a DW131068	
06	Escribir palabra	Registros de parada	40001 a 49999 ó	DW0 a DW19998 o bien	
			400001 a 465535	DW0 a DW131068	
16	Escribir palabras	Registros de parada	40001 a 49999 ó	DW0 a DW19998 o bien	
			400001 a 465535	DW0 a DW131068	

La tabla siguiente muestra las funciones de diagnóstico Modbus soportadas.

Funciones de diagnóstico Modbus de "MB_SLAVE" del S7-1200		
Códigos	Subfunción	Descripción
08	0000H	Emitir datos de consulta del test de eco: la instrucción "MB_SLAVE" devuelve a un maestro Modbus el eco de una palabra de datos recibida.
08	000AH	Borrar contador de eventos de comunicación: la instrucción "MB_SLAVE" borra el contador de eventos de comunicación que se ha utilizado para la función Modbus 11.
11	-	Consultar contador de eventos de comunicación: la instrucción "MB_SLAVE" utiliza un contador de eventos de comunicación interno para registrar el número de solicitudes correctas de lectura y escritura Modbus que se envían al esclavo Modbus. El contador no se incrementa con las funciones 8 y 11, ni con las solicitudes broadcast. Tampoco se incrementa con solicitudes que provocan errores de comunicación (p. ej., errores de paridad o CRC).

La instrucción "MB_SLAVE" soporta solicitudes de escritura broadcast del maestro Modbus mientras las solicitudes sean para acceder a direcciones válidas.

Independientemente de la validez de una solicitud, "MB_SLAVE" no emite ninguna respuesta a un maestro Modbus como resultado de una solicitud broadcast.

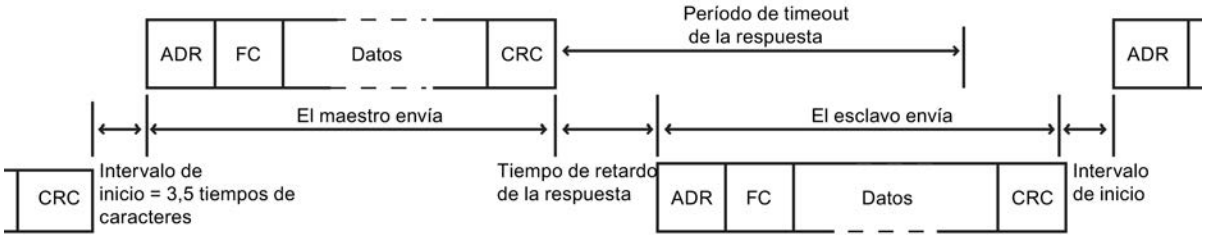
Reglas para la comunicación del esclavo Modbus

- "MB_COMM_LOAD" debe ejecutarse para configurar un puerto para que la instrucción "MB_SLAVE" pueda comunicarse con este puerto.
- Si un puerto debe responder a un maestro Modbus como esclavo, dicho puerto no puede ser utilizado por "MB_MASTER (Página 5535)". Sólo se puede utilizar una instancia de "MB_SLAVE" para un puerto determinado.

- Las instrucciones Modbus no utilizan eventos de alarma de comunicación para controlar el proceso de comunicación. El programa debe controlar el proceso de comunicación consultando la instrucción "MB_SLAVE" en los procedimientos de transmisión y recepción finalizados.
- La instrucción "MB_SLAVE" debe ejecutarse periódicamente con una frecuencia que permita una respuesta actual a las solicitudes entrantes de un maestro Modbus. Por este motivo, resulta recomendable llamar la instrucción en un OB de programa cíclico. La llamada de la instrucción "MB_SLAVE" se puede efectuar en un OB de alarma, pero no es recomendable ya que puede provocar retardos temporales considerables en la ejecución.

Frecuencia de ejecución de "MB_SLAVE"

La instrucción "MB_SLAVE" debe ejecutarse periódicamente para recibir todas las solicitudes del maestro Modbus y responder como corresponda. La frecuencia de ejecución de "MB_SLAVE" se rige por el valor del timeout de respuesta indicado en el maestro Modbus. Esto se representa en la figura que hay a continuación.



El período de timeout de respuesta es el tiempo que espera un maestro Modbus hasta el inicio de respuesta de un esclavo Modbus. Este período no se define en el protocolo Modbus, sino en un parámetro del correspondientes maestro Modbus. La frecuencia de ejecución (el tiempo entre una ejecución y la siguiente) de "MB_SLAVE" debe regirse por los parámetros correspondientes del maestro Modbus. Como mínimo, "MB_SLAVE" debe ejecutarse dos veces durante el período de timeout de respuesta del maestro Modbus.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MB_SLAVE":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MB_ADDR	Input	USINT	I, Q, M, D, L o constante	Dirección de estación del esclavo Modbus (espacio de direcciones: de 0 a 255)
MB_HOLD_REG	Input	VARIANT	D	Puntero hacia el DB del registro de parada Modbus. El DB se debe crear con el tipo de acceso "Estándar: compatible con S7-300/400".
NDR	Output	BOOL	I, Q, M, D, L	Nuevos datos listos: • 0: No hay datos nuevos • 1: Indica que el maestro Modbus ha escrito datos nuevos

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DR	Output	BOOL	I, Q, M, D, L	Leer datos: • 0: No se han leído datos • 1: Indica que los datos han sido leídos por el maestro Modbus
ERROR	Output	BOOL	I, Q, M, D, L	• 0: No se ha detectado ningún error • 1: Error; el código de error correspondiente se emite en el parámetro STATUS
STATUS	Output	WORD	I, Q, M, D, L	Código de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

STATUS* (W#16#....)	Descripción	
80C8	El timeout de respuesta indicado (véase RCVTIME o MSGTIME) es "0".	
80D1	El receptor ha solicitado un control de flujo para interrumpir una transmisión en curso, pero la transmisión no se ha vuelto a activar en el tiempo de espera. Este error también se crea en el control de flujo por hardware cuando el receptor no detecta ningún CTS en el tiempo de espera.	
80D2	La solicitud de transmisión se ha cancelado, porque no se recibe ninguna señal DSR del DCE.	
80E0	El aviso se ha finalizado porque el búfer de recepción está lleno	
80E1	El aviso se ha cancelado por un error de paridad	
80E2	El aviso se ha cancelado por un error de telegrama	
80E3	El aviso se ha cancelado por un error de desbordamiento	
80E4	El aviso se ha cancelado, porque la longitud indicada excede el tamaño del búfer total	
8180	Valor no válido para la ID de puerto.	
8186	Dirección de estación Modbus no válida	
8187	Puntero no válido hacia MB_HOLD_REG-DB	
818C	Puntero hacia un tipo de DB MB_HOLD_REG típico (se requiere un tipo de DB clásico)	
Código de respuesta enviado al maestro Modbus (B#16#...)		
8380	Sin respuesta	Error CRC
8381	01	Código de función no soportado, o bien no soportado dentro de una difusión general
8382	03	Error en la longitud de datos
8383	02	Error en la dirección de los datos o dirección fuera del rango admisible de MB_HOLD_REG
8384	03	Error en el valor de datos
8385	03	Valor del código de diagnóstico de datos no soportado (código de función 08)
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

DB de instancia de la instrucción "MB_SLAVE"

Variables estáticas del DB de instancia

La tabla siguiente describe las variables estáticas del DB de instancia de la instrucción que se pueden utilizar en el programa de usuario. El programa puede escribir valores en las variables HR_Start_Offset y Extended_Addressing y controlar las operaciones del esclavo Modbus.

Las demás variables pueden ser leídas para supervisar el estado del Modbus.

Variable	Tipo de datos	Descripción
HR_Start_Offset	WORD	Dirección inicial del registro de parada Modbus (estándar="0")
Extended_ Addressing	BOOL	Configuración del direccionamiento: • 0: Área de direccionamiento estándar (1 byte) • 1: Área de direccionamiento ampliada (2 bytes)
Request_Count	WORD	Número de todas las consultas recibidas por el esclavo
Slave_Message_ Count	WORD	Número de consultas enviadas específicamente a este esclavo
Bad_CRC_Count	WORD	Número de consultas recibidas con error CRC
Broadcast_Count	WORD	Número de consultas recibidas de difusión general
Exception_Count	WORD	Número de errores específicos del Modbus que requieren el reenvío de una excepción
Success_Count	WORD	Número de solicitudes para este esclavo específico recibidas sin error de protocolo

HR_Start_Offset

Las direcciones del registro de parada del Modbus empiezan en 40001 ó 400001. Estas direcciones corresponden a la dirección inicial del registro de parada en la memoria del sistema de destino. Por medio de la variable HR_Start_Offset se puede especificar el desplazamiento respecto a otra dirección inicial.

Ejemplo: Un registro de parada empieza en MW100 y tiene una longitud de 100 WORD. Con un desplazamiento de 20 en el parámetro HR_Start_Offset, el registro de parada empieza en la dirección 40021 en vez de en 40001. Toda dirección inferior a 40021 y superior a 400119 provoca un error de direccionamiento.

	HR_Start_Offset = 0			HR_Start_Offset = 20	
	Dirección de palabra Modbus	Dirección de byte S7-1200		Dirección de palabra Modbus	Dirección de byte S7-1200
Mínimo	40001	MW100		40021	MW100
Máximo	40099	MW198		40119	MW198

Extended_Addressing

Para el direccionamiento del esclavo Modbus se puede configurar un solo byte (área de direccionamiento estándar) o un byte doble (área de direccionamiento ampliada). El direccionamiento ampliado sirve para direccionar más de 247 dispositivos en una única red. Si opta por el direccionamiento ampliado, puede direccionar 64.000 direcciones como máximo. A continuación se muestra un telegrama de la función Modbus 1 como ejemplo.

Tabla 9-165 Dirección del esclavo con un byte (byte 0)

Función 1	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Requisito	Dirección del esclavo	Código F	Dirección inicial		Longitud de las bobinas		
Respuesta válida	Dirección del esclavo	Código F	Longitud	Datos de las bobinas			
Respuesta errónea	Dirección del esclavo	0x81	Código E				

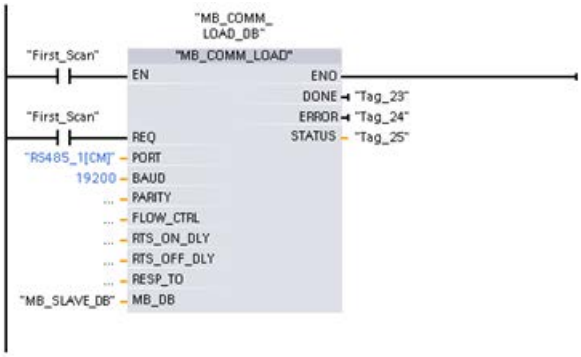
Tabla 9-166 Dirección del esclavo con dos bytes (byte 0 y byte 1)

Función 1	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Requisito	Dirección del esclavo		Código F	Dirección inicial		Longitud de las bobinas	
Respuesta válida	Dirección del esclavo		Código F	Longitud	Datos de las bobinas		
Respuesta errónea	Dirección del esclavo		0x81	Código F			

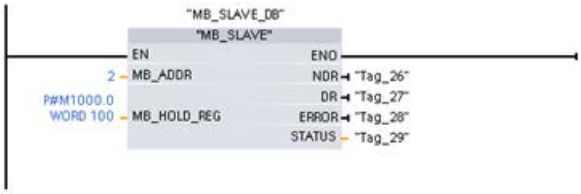
Programa de ejemplo de un esclavo Modbus

Segmentos (KOP)

Segmento 1: Inicializar los parámetros del módulo RS-485 sólo una vez durante el primer ciclo.



Segmento 2: Comprobar en cada ciclo la presencia de solicitudes del maestro Modbus. El registro de parada Modbus está configurado con 100 palabras a partir de MW1000.



MODBUS (TCP)

MODBUS (TCP)

MB_CLIENT: Comunicarse como cliente Modbus TCP vía PROFINET

Descripción MB_CLIENT

Descripción

La instrucción "MB_CLIENT" permite la comunicación como cliente Modbus TCP a través de la conexión PROFINET de la CPU S7-1200. Para utilizar esta instrucción no se requiere ningún módulo de hardware adicional. La instrucción "MB_CLIENT" permite establecer una conexión entre el cliente y el servidor, enviar peticiones y recibir respuestas y controlar la desconexión del servidor Modbus TCP.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "MB_CLIENT":

Parámetro	Declaración	Tipo de datos	Descripción
REQ (Página 5552)	Input	BOOL	Petición de comunicación con el servidor Modbus TCP El parámetro REQ se controla por nivel. Así, mientras la entrada esté activada (REQ=true), la instrucción enviará peticiones de comunicación. • Con la petición de comunicación se bloquea el DB de instancia para otros clientes. • Las modificaciones de los parámetros de entrada no se hacen efectivas hasta que no hay respuesta del servidor o hasta que no se devuelve un mensaje de error. • Si durante una petición Modbus en curso se vuelve a activar el parámetro REQ, a continuación no se ejecuta ninguna otra transferencia.
DISCONNECT (Página 5552)	Input	BOOL	Mediante este parámetro se controla el establecimiento de la conexión y la desconexión con el servidor Modbus: • 0: Establecer una conexión de comunicación con la dirección IP y número de puerto especificados. • 1: Deshacer la conexión. Durante la desconexión no se ejecuta ninguna otra función. Tras deshacer la conexión correctamente, el parámetro STATUS devuelve el valor 7003. Si el parámetro REQ está activado mientras se establece la conexión, la consulta se envía de inmediato.
CONNECT_ID	Input	UINT	ID unívoca para identificar la conexión. A cada instancia de las instrucciones "MB_CLIENT" y "MB_SERVER (Página 5557)" debe asignársele una ID de conexión unívoca.
IP_OCTET_1	Input	USINT	1. Octeto de la dirección IP* del servidor Modbus TCP.
IP_OCTET_2	Input	USINT	2. Octeto de la dirección IP* del servidor Modbus TCP.
IP_OCTET_3	Input	USINT	3. Octeto de la dirección IP* del servidor Modbus TCP.
IP_OCTET_4	Input	USINT	4. Octeto de la dirección IP* del servidor Modbus TCP.
IP_PORT	Input	UINT	Número de IP y puerto del servidor con el que el cliente establece la conexión y con el que se comunica mediante el protocolo TCP/IP (valor estándar: 502).
MB_MODE (Página 5553)	Input	USINT	Selección del modo de petición (lectura, escritura o diagnóstico).
MB_DATA_ADDR (Página 5553)	Input	UDINT	Dirección inicial de los datos a los que accede la instrucción "MB_CLIENT".
DATA_LEN	Input	UINT	Longitud de datos: Número de bits o palabras para el acceso a los datos (ver "Parámetros MB_MODE y MB_DATA_ADDR": longitud de datos).
MB_DATA_PTR (Página 5554)	InOut	VARIANT	Puntero al registro de datos Modbus: El registro es un búfer para los datos recibidos desde el servidor Modbus o que se van a enviar al servidor Modbus. El puntero debe remitir a un bloque de datos global con acceso estándar. El número de bits direccionados debe ser divisible entre 8.
DONE	Out	BOOL	El bit del parámetro de salida DONE se pone a "1" en cuanto se ha ejecutado sin errores la última petición.

Parámetro	Declaración	Tipo de datos	Descripción
BUSY	Out	BOOL	• 0: No se está ejecutando ninguna petición de "MB_CLIENT " • 1: Petición de "MB_ CLIENT " en ejecución
ERROR	Out	BOOL	• 0: Sin errores • 1: Con errores. La causa del error se indica mediante el parámetro STATUS.
STATUS (Página 5555)	Out	WORD	Código de error de la instrucción.
* Componente de 8 bits de longitud de la dirección IP IPv4 de 32 bits del servidor Modbus TCP.			

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Datos de entrada coherentes durante una llamada de "MB_CLIENT"

En cuanto un cliente Modbus ejecuta una instrucción Modbus, el estado de los parámetros de entrada se almacena internamente y luego se compara en la siguiente llamada. La comparación se utiliza para determinar si es esa petición concreta la que ha iniciado la consulta en curso. Pueden realizarse varias llamadas de "MB_CLIENT" utilizando un DB de instancia común. Los valores de los parámetros de entrada no deben modificarse mientras esté ejecutándose una instancia "MB_CLIENT". Si se modifican los parámetros de entrada durante la ejecución, no se puede comprobar mediante "MB_CLIENT" si la instancia se está ejecutando en el momento actual.

Conexiones múltiples de clientes

Un cliente Modbus TCP puede admitir varias conexiones TCP (el número máximo de conexiones depende de la CPU utilizada). El total de conexiones de una CPU, incluidos los clientes Modbus TCP y los servidores, no debe exceder el número máximo de conexiones admitido. Las conexiones Modbus TCP también pueden ser utilizadas conjuntamente por las conexiones de cliente y/o servidor.

En algunas conexiones de cliente deben respetarse las siguientes reglas:

- Cada conexión "MB_CLIENT" debe utilizar un DB de instancia unívoco.
- Para cada conexión "MB_CLIENT" debe especificarse una dirección IP unívoca del servidor.
- Cada conexión "MB_CLIENT" requiere una ID de conexión unívoca.
  Para cada DB de instancia de la instrucción debe utilizarse la correspondiente ID de conexión. Las ID de conexión y los DB de instancia se agrupan por pares y deben ser unívocos para cada conexión.
- Según la configuración del servidor, se requerirán o no números unívocos de puerto IP.

Variables estáticas de la instrucción

En la siguiente tabla se describen las variables estáticas editables del bloque de datos de instancia de la instrucción "MB_CLIENT".

Variable	Tipo de datos	Valor de arranque	Descripción
Blocked_Proc_Timeout	REAL	3.0	Tiempo en segundos que hay que esperar para que se inicialice la variable estática ACTIVE en una instancia Modbus bloqueada. Esto puede suceder, por ejemplo, cuando, habiéndose emitido una petición de cliente, se cancela la ejecución de la función de cliente antes de que se haya ejecutado por completo la petición. El tiempo de espera máximo es de 55 segundos.
MB_Transaction_ID	WORD	1	ID de transacción del protocolo Modbus TCP. El valor de arranque "1" solo debe modificarse en caso de que el servidor Modbus TCP necesite un valor diferente.
MB_Unit_ID	BYTE	255	ID de Unit del protocolo Modbus. La variable corresponde a la dirección de esclavo del protocolo Modbus RTU. Modifique este valor solo si el servidor Modbus TCP puede utilizarse como gateway y si es controlado por el programa de usuario dentro del servidor Modbus.
RCV_TIMEOUT	REAL	2.0	Intervalo en segundos durante el que la instrucción "MB_CLIENT" permanece a la espera de una respuesta del servidor.
Connected	BOOL	0	Indica si la conexión con el cliente asignado está establecida o no: 1 = conectado, 0 = no conectado.

Consulte también

MB_CLIENT ejemplo 1: Enviar varias peticiones a través de una conexión TCP (Página 5564)

MB_CLIENT ejemplo 2: Enviar varias peticiones a través de varias conexiones TCP (Página 5565)

MB_CLIENT ejemplo 3: Coordinar varias peticiones (Página 5566)

Parámetros REQ y DISCONNECT

Descripción

Si no se ejecuta ninguna instancia de la instrucción "MB_CLIENT" y si el parámetro DISCONNECT tiene el valor "0", con REQ=1 se ejecuta otra petición. Si todavía no se ha establecido ninguna conexión, esta se establecerá al ejecutarse la instrucción.

Si se vuelve a ejecutar la misma instancia de la instrucción "MB_CLIENT" (DISCONNECT=0 y REQ=1) antes de que se ejecute la petición activa, esta no se ejecutará a continuación de la petición activa. Solo podrá iniciarse una nueva petición una vez que haya finalizado la petición activa (REQ=1).

Para vigilar el estado de la ejecución puede usarse el parámetro de salida DONE. Este puede usarse para vigilar el estado de ejecución durante una ejecución secuencial de la instrucción "MB_CLIENT".

Consulte también

Descripción MB_CLIENT (Página 5549)

Parámetros MB_MODE y MB_DATA_ADDR

Descripción

La instrucción "MB_CLIENT" utiliza, en lugar de un código de función, el parámetro MB_MODE. Mediante el parámetro MB_DATA_ADDR se define la dirección de arranque Modbus de los datos a los que se desea acceder. La combinación de los parámetros MB_MODE y MB_DATA_ADDR determina el código de función que se utiliza en el mensaje Modbus actual.

La tabla siguiente muestra la relación entre el parámetro MB_MODE, la función Modbus y el área de direcciones.

MB_MODE	Función Modbus	Longitud de datos	Función y tipo de datos	MB_DATA_ADDR
0	01	de 1 a 2000	Leer bits de salida: de 1 a 2000 bits por llamada	de 1 a 9999
0	02	de 1 a 2000	Leer bits de entrada: de 1 a 2000 bits por llamada	de 10001 a 19999
0	03	de 1 a 125	Leer registro de parada: de 1 a 125 WORD por llamada	de 40001 a 49999
0	04	de 1 a 125	Leer palabras de entrada: de 1 a 125 WORD por llamada	de 30001 a 39999
1	05	1	Escribir un bit de salida: Un bit por llamada	de 1 a 9999
1	06	1	Escribir un registro de parada: 1 WORD por llamada	de 40001 a 49999
1	15	de 2 a 1968	Escribir varios bits de salida: de 2 a 1968 bits por llamada	de 1 a 9999
1	16	de 2 a 123	Escribir varios registros de parada: de 2 a 123 WORD por llamada	de 40001 a 49999
2	15	de 1 a 1968	Escribir uno o varios bits de salida: de 1 a 1968 bits por llamada	de 1 a 9999
2	16	de 1 a 123	Escribir uno o varios registros de parada: de 1 a 123 WORD por llamada	de 40001 a 49999

MB_MODE	Función Modbus	Longitud de datos	Función y tipo de datos	MB_DATA_ADDR
11	11	0	Leer la palabra de estado y el contador de eventos de la comunicación de servidor: • La palabra de estado forma el estado de ejecución (0: no en ejecución, 0xFFFF: en ejecución). • El contador de eventos se incrementa cada vez que se envía correctamente un mensaje. Los parámetros MB_DATA_ADDR y MB_DATA_LEN de la instrucción "MB_CLIENT" no se evalúan al ejecutarse esta función.	-
80	08	1	Comprobación del estado del servidor mediante el código de error 0x0000 (test de bucle de retorno: el servidor devuelve la petición): 1 WORD por llamada	-
81	08	1	Reiniciar el contador de eventos del servidor mediante el código de error 0x000A: 1 WORD por llamada	
de 3 a 10, de 12 a 79, de 82 a 255			Reservado	

Consulte también

Descripción MB_CLIENT (Página 5549)

Parámetro MB_DATA_PTR

Descripción

El parámetro MB_DATA_PTR es un puntero hacia un búfer de datos para almacenar los datos que se han leído o escrito en el servidor Modbus. Puede usarse como búfer de datos un bloque de datos global o un área de memoria (M).

Para un búfer en el área de memoria (M), utilice un puntero con el formato ANY según el patrón"P#dirección_de_bit" "Tipo de datos" "Longitud" (ejemplo: P#M1000.0 WORD 500).

El parámetro MB_DATA_PTR utiliza un búfer de comunicación:

- Para las funciones de comunicación de la instrucción "MB_CLIENT":
  - Lectura y escritura de datos de 1 bit de las direcciones de servidor Modbus de 00001 a 09999 y de 10001 a 19999.
  - Lectura de datos WORD de 16 bits de las direcciones de servidor Modbus de 30001 a 39999 y de 40001 a 49999.
  - Escritura de datos WORD de 16 bits de las direcciones del servidor Modbus de 40001 a 49999.
- Para la transferencia de datos (longitud: bit o WORD) desde o hacia el DB global o el área de memoria (M) que se ha asignado mediante el parámetro MB_DATA_PTR.

Si se utiliza en el parámetro MB_DATA_PTR un bloque de datos para el puntero al búfer, deben asignarse tipos de datos a los elementos de DB.

- Para una dirección de bit Modbus, utilice el tipo de datos de 1 bit BOOL
- Para una dirección Modbus WORD, utilice un tipo de datos de 16 bits como WORD, UINT, INT o REAL.
- Para dos direcciones Modbus WORD, utilice un tipo de datos de 32 bits (palabra doble) como DWORD, DINT o REAL.
- Mediante MB_DATA_PTR se puede acceder también a elementos de DB complejos como:
  - Matrices estándar
  - Estructuras con nombres de elemento unívocos
  - Estructuras complejas con denominación unívoca de los elementos longitudes de tipo de datos de 16 ó 32 bits.
- Las áreas de datos para el parámetro MB_DATA_PTR pueden encontrarse también en distintos bloques de datos globales (o en distintas áreas de memoria). Por ejemplo, puede utilizarse un bloque de datos para las operaciones de lectura y otro para las operaciones de escritura, o un bloque de datos independiente para cada estación "MB_CLIENT".

Consulte también

Descripción MB_CLIENT (Página 5549)

Parámetro STATUS

Parámetro STATUS (información de estado general)

STATUS* (W#16#)	Descripción
0000	Instrucción ejecutada sin errores.
0001	Conexión establecida.
0003	Desconexión realizada.
7000	Ninguna llamada activa (REQ=0).
7001	Primera llamada con REQ=1: Ejecución iniciada; BUSY tiene el valor 1.
7002	Llamada intermedia (REQ irrelevante). Ejecución ya activa; BUSY tiene el valor 1.
7003	Deshaciendo la conexión.
7004	Conexión establecida y vigilada. No se está procesando ninguna petición.
7005	Datos enviados.
7006	Datos recibidos.
80BB	Valor no válido en el parámetro ACTIVE_EST (identificador del tipo de establecimiento de conexión, véase T_CON_PARAM): • El lado servidor solo permite el establecimiento pasivo de la conexión (ACTIVE_EST = FALSE). • El lado cliente solo permite el establecimiento activo de la conexión (ACTIVE_EST = TRUE).

STATUS* (W#16#)	Descripción
8380	El frame Modbus recibido no tiene el formato correcto o la cantidad de bytes recibidos no es suficiente.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (error de protocolo)

STATUS* (W#16#)	Código de la respuesta al cliente Modbus (B#16#)	Descripción
8381	01	Este código de función no se admite.
8382	03	Error en la longitud de datos.
8383	02	Error en la dirección de los datos o acceso fuera del área de memoria de MB_DATA_PTR (Página 5554).
8384	03	Error en el valor de datos.
8385	03	Códigos de error del diagnóstico no admitidos (código de función 08).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Parámetro STATUS (error de parámetro)

Además de los errores listados en la siguiente tabla, con la instrucción "MB_CLIENT" pueden producirse también errores en las instrucciones de comunicación utilizadas por la instrucción ("TCON", "TDISCON", "TSEND" y "TRCV").

STATUS* (W#16#)	Descripción
80C8	No ha habido respuesta del servidor en el periodo definido. Compruebe la conexión con el servidor Modbus. Este error no se notifica hasta que ha no finalizado el número configurado de reintentos. Si la instrucción "MB_CLIENT" no obtiene respuesta alguna con la ID de transacción transmitida originalmente (variable MB_TRANSACTION_ID) dentro del período definido, entonces de emitirá un código de error.
8188	Valor no válido en el parámetro MB_MODE.
8189	Direccionamiento no válido de los datos en el parámetro MB_DATA_ADDR.
818A	Longitud de datos no válida en el parámetro MB_DATA_LEN.
818B	Puntero no válido en el parámetro MB_DATA_PTR. Compruebe además los valores de los parámetros MB_DATA_ADDR (Página 5553) y MB_DATA_LEN.
818C	• El puntero del parámetro MB_DATA_PTR (Página 5554) señala a un bloque de datos optimizado. Utilice un bloque de datos con acceso estándar o un área de memoria. • Rebase de tiempo en el parámetro BLOCKED_PROC_TIMEOUT (ver variables estáticas de la instrucción). Se ha rebasado el límite de 55 segundos.
818D	La ID de transacción (variable MB_TRANSACTION_ID) no coincide con la enviada originalmente (ver variables estáticas de la instrucción).
8200	• En este momento se está procesando otra petición Modbus a través del puerto. • Otra instancia de MB_CLIENT con los mismos parámetros de conexión está procesando una petición Modbus ya existente.

STATUS* (W#16#)	Descripción
8380	El bloque de transferencia de datos Modbus recibido presenta defectos de forma o no se han recibido suficientes bytes.
8386	El código de función recibido no coincide con el enviado originalmente.
8387	• La ID de conexión asignada es distinta de las de anteriores peticiones. Solo puede usarse una ID de conexión para cada DB de instancia de la instrucción "MB_CLIENT". • Este código de error se emite aunqe la ID del protocolo Modbus TCP recibido por el servidor sea distinta de "0".
8388	Se ha enviado a través del servidor Modbus una longitud de datos distinta de la solicitada. Este error solo se produce si se utilizan las funciones Modbus 15 o 16.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Nota

Códigos de error de las instrucciones de comunicación utilizadas internamente.

Además de los errores listados en las tablas, con la instrucción "MB_CLIENT" pueden producirse también errores en las instrucciones de comunicación ("TCON", "TDISCON", "TSEND" y "TRCV") utilizadas por la instrucción.

La asignación del código de error se puede realizar a través del bloque de datos de instancia de la instrucción "MB_CLIENT". En la sección Static se indican los códigos de error en la instrucción correspondiente bajo STATUS.

El significado de los códigos de error se puede consultar en la documentación de la instrucción de comunicación correspondiente.

Consulte también

Descripción MB_CLIENT (Página 5549)

Parámetro MB_HOLD_REG (Página 5561)

Cambiar formatos de visualización en el estado del programa (Página 3100)

MB_SERVER: Comunicarse como servidor Modbus TCP vía PROFINET

Descripción MB_SERVER

Descripción

La instrucción "MB_SERVER" permite la comunicación como servidor Modbus TCP a través de la conexión PROFINET de la CPU S7-1200. Para utilizar esta instrucción no se requiere ningún módulo de hardware adicional. La instrucción "MB_SERVER" permite procesar

peticiones de conexión de un cliente Modbus TCP, recibir peticiones de funciones Modbus y enviar mensajes de respuesta.

ATENCIÓN
Consigna de seguridad Tenga en cuenta que cada cliente de la red tiene acceso a los datos de las instrucciones Modbus. Si bien el bloque de datos para el registro de parada Modbus no se puede direccionar directamente con el cliente, existe la posibilidad de acceder indirectamente a los datos a través de las direcciones de offset. Para evitar que personas no autorizadas lleven a cabo procesos de lectura y escritura, es posible restringir el acceso a una dirección IP. No obstante, tenga en cuenta que la dirección habilitada también ofrece acceso a personas no autorizadas.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MB_SERVER":

Parámetro	Declaración	Tipo de datos	Descripción
DISCONNECT	Input	BOOL	La instrucción "MB_SERVER" tiene una conexión pasiva con un módulo interlocutor, es decir, el servidor reacciona a una petición de conexión TCP de cada dirección IP que la solicite. Este parámetro permite controlar cuándo se aceptará una petición de conexión: • 0: Si no hay ninguna conexión establecida, se establece una conexión pasiva. • 1: Inicialización del establecimiento de la conexión. Si la entrada esta activada, no se ejecutan otras operaciones. Tras deshacer la conexión correctamente, el parámetro STATUS devuelve el valor 7003.
CONNECT_ID	Input	UINT	Mediante este parámetro se identifica una conexión en la CPU de modo unívoco. Cada una de las instancias de las instrucciones "MB_CLIENT (Página 5549)" y "MB_SERVER" debe utilizar una ID unívoca en el parámetro CONNECT_ID.
IP_PORT	Input	UINT	Valor de arranque=502. El número del puerto IP determina qué puerto IP se vigila para peticiones de conexión del cliente Modbus. Los siguientes números de puerto TCP no deben usarse para la conexión pasiva de la instrucción "MB_SERVER": 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.
MB_HOLD_REG (Página 5561)	InOut	VARIANT	Puntero al registro de parada Modbus de la instrucción "MB_SERVER": Utilice como registro de parada un bloque de datos global con acceso estándar. El registro de parada contiene los valores a los que está autorizado a acceder un cliente Modbus mediante las funciones Modbus 3 (lectura), 6 (escritura) y 16 (lectura).
NDR	Output	BOOL	"New Data Ready": • 0: No hay datos nuevos • 1: El cliente Modbus ha escrito datos nuevos

Parámetro	Declaración	Tipo de datos	Descripción
DR	Output	BOOL	"Data Read": • 0: No se han leído datos • 1: El cliente Modbus ha leído datos
ERROR	Output	BOOL	Si se produce un error durante una llamada de la instrucción "MB_SERVER", la salida del parámetro ERROR se ajusta a TRUE. La causa detallada del error se muestra en el parámetro STATUS.
STATUS (Página 5562)	Output	WORD	Código de error de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Reproducir las direcciones Modbus en la memoria imagen de proceso

La instrucción "MB_SERVER" permite a las funciones Modbus entrantes (1, 2, 4, 5 y 15) acceder directamente en lectura y escritura a la memoria imagen de proceso de las entradas y salidas de la CPU S7-1200 (usando los tipos de datos BOOL y WORD).

Para la transferencia de datos de los códigos de función 3, 6 y 16, el registro de parada (parámetro MB_HOLD_REG) debe definirse con una longitud de más de un byte. La tabla siguiente muestra la imagen de las direcciones Modbus en la memoria imagen de proceso de la CPU.

Función Modbus						S7-1200	
Código	Función	Área de datos	Área de direcciones			Área de datos	Dirección de la CPU
01	Leer: bits	Output	1	a	8192	Memoria imagen de proceso de la salida	De Q0.0 a Q1023.7
02	Leer: bits	Input	10001	a	18192	Memoria imagen de proceso de la entrada	De I0.0 a I1023.7
04	Leer: WORD	Input	30001	a	30512	Memoria imagen de proceso de la entrada	De IW0 a IW1022
05	Escribir: bits	Output	1	a	8192	Memoria imagen de proceso de la salida	De Q0.0 a Q1023.7
15	Escribir: bits	Output	1	a	8192	Memoria imagen de proceso de la salida	De Q0.0 a Q1023.7

Los mensajes Modbus entrantes con los códigos de función 3, 6 y 16 escriben o leen desde el registro de parada de Modbus (el registro de parada debe especificarse en el parámetro MB_HOLD_REG).

Conexiones múltiples a servidor

Pueden establecerse conexiones múltiples a servidor. Gracias a ello, una sola CPU puede establecer conexiones con varios clientes Modbus TCP al mismo tiempo.

Un servidor Modbus TCP puede admitir varias conexiones TCP (el número máximo de conexiones depende de la CPU utilizada).

El total de conexiones de una CPU, incluidos los clientes Modbus TCP y los servidores, no debe exceder el número máximo de conexiones admitido.

Las conexiones Modbus TCP también pueden ser utilizadas conjuntamente por las conexiones de cliente y/o servidor.

Para las conexiones de servidor deben respetarse las siguientes reglas:

- Cada conexión "MB_SERVER" debe utilizar un DB de instancia unívoco.
- Cada conexión "MB_SERVER" debe establecerse con un número unívoco de puerto IP. Se admite una sola conexión para cada puerto.
- Cada conexión "MB_SERVER" debe utilizar una ID de conexión unívoca.
  Para cada DB de instancia de la instrucción debe utilizarse la correspondiente ID de conexión. Las ID de conexión y los DB de instancia se agrupan por pares y deben ser unívocos para cada conexión.
- Para cada conexión debe llamarse separadamente la instrucción "MB_SERVER".

Funciones de diagnóstico Modbus

La tabla siguiente contiene la descripción de las funciones de diagnóstico de Modbus.

Código	Subfunción	Descripción
08	0x0000	Prueba de eco: La instrucción "MB_SERVER" recibe una palabra de datos y la devuelve inalterada al maestro Modbus.
08	0x000A	Inicializar el contador de eventos: La instrucción "MB_SERVER" inicializa el contador de eventos de comunicación que se utiliza para la función Modbus 11.
11	-	Recoger contador de eventos de la comunicación: Durante la comunicación, la instrucción "MB_SERVER" utiliza un contador de eventos interno para registrar el número de peticiones de lectura y escritura correctas que se envían al servidor Modbus. El contador de eventos no se incrementa en las funciones 8 y 11 ni en las peticiones de difusión general. Lo mismo sucede con las peticiones que dan lugar a un error de comunicación (por ejemplo en caso de errores de paridad o de CRC). La función de difusión general no está disponible para Modbus TCP, ya que solo puede existir una conexión cliente/servidor al mismo tiempo.

Variables estáticas de la instrucción

En la tabla siguiente se describen las variables estáticas del bloque de datos de instancia de la instrucción MB_SERVER que se pueden utilizar en el programa. A la variable HR_Start_Offset se puede acceder en escritura. A las demás variables se puede acceder en lectura para vigilar el estado del Modbus.

Variable	Tipo de datos	Valor de arranque	Descripción
HR_Start_Offset	WORD	0	Asignar la dirección inicial del registro de parada de Modbus.
Request_Count	WORD	0	Número total de peticiones recibidas por el servidor.
Server_Message_Count	WORD	0	Número total de mensajes recibidos para el servidor en cuestión.

Variable	Tipo de datos	Valor de arranque	Descripción
Xmt_Rcv_Count	WORD	0	Contador para el registro del número de transferencias en las que se ha producido un error. El contador se incrementa aunque se reciba un mensaje Modbus no válido.
Exception_Count	WORD	0	Contador para el registro del número de errores específicos de Modbus que dan lugar a un error de excepción.
Success_Count	WORD	0	Contador para el registro del número de peticiones que no contienen errores en el protocolo transferido.
Connected	BOOL	0	Indica si la conexión con el cliente asignado está establecida o no: 1 = conectado, 0 = no conectado.

Ejemplo: Direccionamiento a través de variable estática HR_Start_Offset

Las direcciones del registro de parada Modbus empiezan con 40001. Estas direcciones corresponden al espacio de direcciones del área de memoria de la CPU para el registro de parada. También es posible definir la variable HR_Start_Offset de modo que el registro de parada de Modbus reciba una dirección inicial distinta de 40001.

Ejemplo: El registro de parada empieza en el MW100 y tiene una longitud de 100 WORD. Con un valor de offset en el parámetro HR_Start_Offset, la dirección inicial del registro de parada se desplaza de 40001 a 40021. Todo direccionamiento del registro de parada por debajo de la dirección 40021 y por encima de la dirección 40119 da lugar a un error.

HR_Start_Offset	Dirección	Mínimo	Máximo
0	Dirección Modbus (WORD)	40001	40099
	Dirección S7-1200	MW100	MW298
20	Dirección Modbus (WORD)	40021	40119
	Dirección S7-1200	MW100	MW298

Consulte también

MB_SERVER Ejemplo: Varias conexiones TCP (Página 5567)

Parámetro MB_HOLD_REG

Descripción

El parámetro MB_HOLD_REG es un puntero hacia un búfer de datos para almacenar los datos que se han leído o escrito en el servidor Modbus. Puede usarse como búfer de datos un bloque de datos global o un área de memoria (M).

Como puntero hacia un búfer del área de memoria (M), utilice el formato ANY según el patrón "P#dirección_del_bit" "Tipo de datos" "Longitud" (ejemplo: P#M1000.0 WORD 500).

En la siguiente tabla se muestran ejemplos de la representación de direcciones Modbus en el registro de parada de las funciones Modbus 3 (leer WORD), 6 (escribir WORD) y 16 (escribir varias WORD). El límite superior del número de direcciones del bloque de datos viene determinado por la memoria de trabajo máxima de la CPU. Si se utiliza un área de memoria,

el número máximo de direcciones viene determinado por el tamaño del área de memoria de la CPU.

Direcciones Modbus	Parámetro MB_HOLD_REG: ejemplos		
P#M100.0 WORD 5	P#DB10.DBx0.0 WORD 5	"Recipe".ingredient	
40001	MW100	DB10.DBW0	"Recipe".ingredient[1]
40002	MW102	DB10.DBW2	"Recipe".ingredient[2]
40003	MW104	DB10.DBW4	"Recipe".ingredient[3]
40004	MW106	DB10.DBW6	"Recipe".ingredient[4]
40005	MW108	DB10.DBW8	"Recipe".ingredient[5]

Consulte también

Descripción MB_SERVER (Página 5557)

Parámetro STATUS

Parámetro STATUS (información de estado general)

STATUS* (W#16#)	Descripción
0000	Instrucción ejecutada sin errores.
0001	Conexión establecida.
0003	Desconexión realizada.
7000	Ninguna llamada activa (REQ=0).
7001	Primera llamada con REQ=1: Ejecución iniciada; BUSY tiene el valor 1.
7002	Llamada intermedia (REQ irrelevante). Ejecución ya activa; BUSY tiene el valor 1.
7003	Deshaciendo la conexión.
7004	Conexión establecida y vigilada. No se está procesando ninguna petición.
7005	Datos enviados.
7006	Datos recibidos.
80BB	Valor no válido en el parámetro ACTIVE_EST (identificador del tipo de establecimiento de conexión, véase T_CON_PARAM): • El lado servidor solo permite el establecimiento pasivo de la conexión (ACTIVE_EST = FALSE). • El lado cliente solo permite el establecimiento activo de la conexión (ACTIVE_EST = TRUE).
8380	El frame Modbus recibido no tiene el formato correcto o la cantidad de bytes recibidos no es suficiente.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (error de parámetro)

STATUS* (W#16#)	Código de la respuesta al servidor Modbus (B#16#)	Descripción
8187	Sin respuesta	Puntero no válido en el parámetro MB_HOLD_REG. El área de datos es demasiado pequeña.
818C	Sin respuesta	• El puntero del parámetro MB_HOLD_REG señala a un bloque de datos optimizado. Utilice un bloque de datos con acceso estándar o un área de memoria. • Error por tiempo excedido en la ejecución (por encima de 55 segundos).
8381	01	Este código de función no se admite.
8382	03	Error en la longitud de datos
8383	02	Error en la dirección de los datos o acceso fuera del área de direcciones del registro de parada (parámetro MB_HOLD_REG (Página 5561)).
8384	03	Error en el valor de datos
8385	03	Valor del código de diagnóstico no soportado (solo en el código de función 08).
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Nota

Códigos de error de las instrucciones de comunicación utilizadas internamente.

Además de los errores listados en las tablas, con la instrucción "MB_SERVER" pueden producirse también errores en las instrucciones de comunicación ("TCON", "TDISCON", "TSEND" y "TRCV") utilizadas por la instrucción.

La asignación del código de error se puede realizar a través del bloque de datos de instancia de la instrucción "MB_SERVER". En la sección Static se indican los códigos de error en la instrucción correspondiente bajo STATUS.

El significado de los códigos de error se puede consultar en la documentación de la instrucción de comunicación correspondiente.

Consulte también

Descripción MB_SERVER (Página 5557)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Ejemplos

MB_CLIENT ejemplo 1: Enviar varias peticiones a través de una conexión TCP

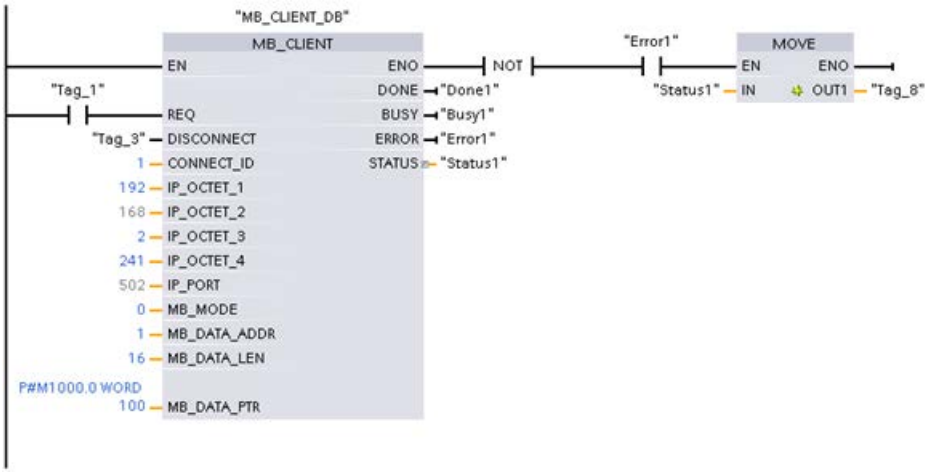
Descripción

A través de una conexión TCP pueden enviarse varias peticiones de cliente Modbus. Al hacerlo, utilice el mismo DB de instancia, la misma ID de conexión y el mismo número de puerto.

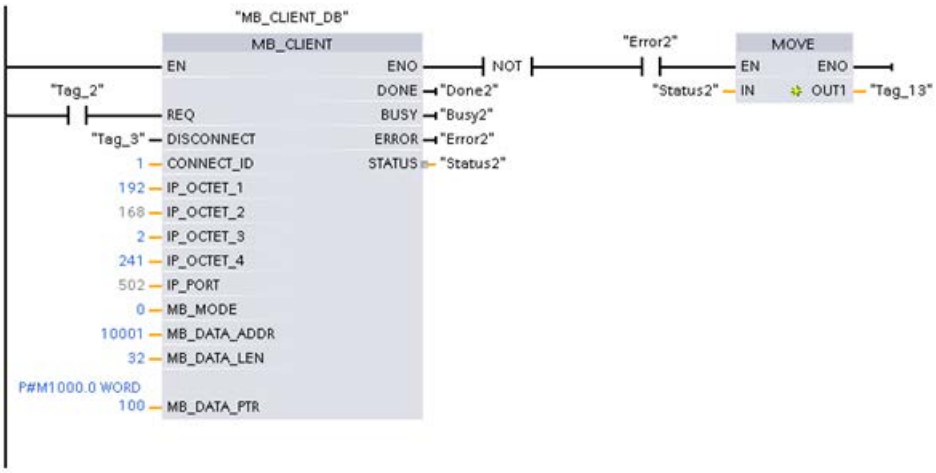
Solo puede haber un cliente activo al mismo tiempo. Una vez finalizada la ejecución de un cliente, se ejecuta el cliente siguiente. El orden de ejecución debe definirse en el programa.

En el siguiente programa de ejemplo se copia como ampliación el valor del parámetro de salida STATUS.

Segmento 1: Función Modbus 1: leer 16 bits de salida



Segmento 2: Función Modbus 2: leer 32 bits de entrada



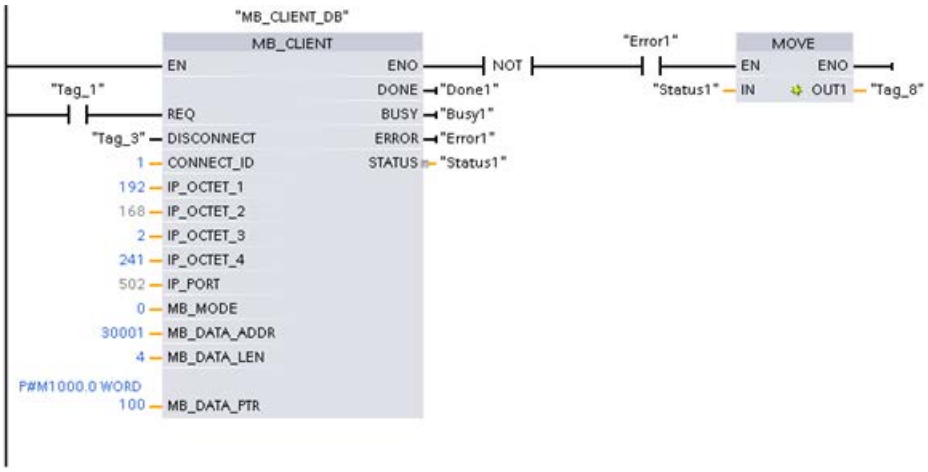
MB_CLIENT ejemplo 2: Enviar varias peticiones a través de varias conexiones TCP

Descripción

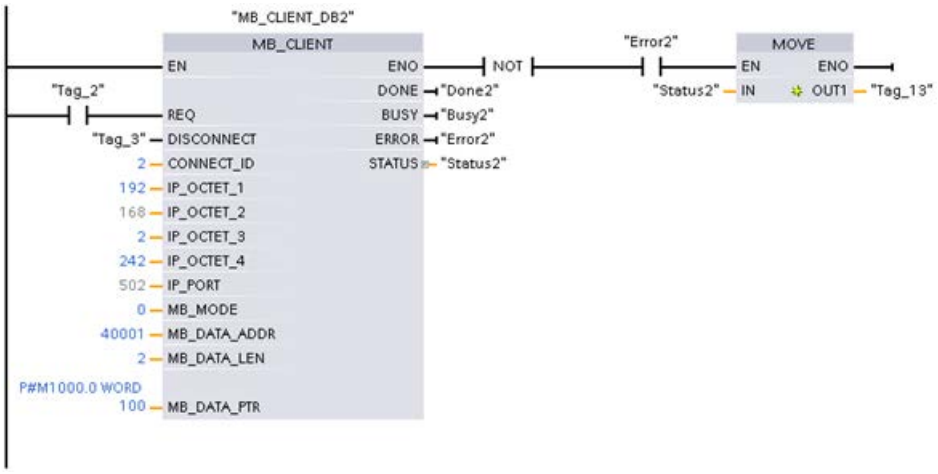
Las peticiones del cliente Modbus pueden enviarse a través de distintas conexiones TCP. Para ello, utilice en cada caso un DB de instancia diferente y una ID de conexión diferente.

Si las conexiones se realizan al mismo servidor Modbus, utilice en cada caso un número de puerto diferente. Si las conexiones se realizan cada una a un servidor Modbus distinto, puede asignar libremente los números de puerto.

Segmento 1: Función Modbus 4: leer entrada (WORD)



Segmento 2: Función Modbus 3: leer registro de parada (WORD)

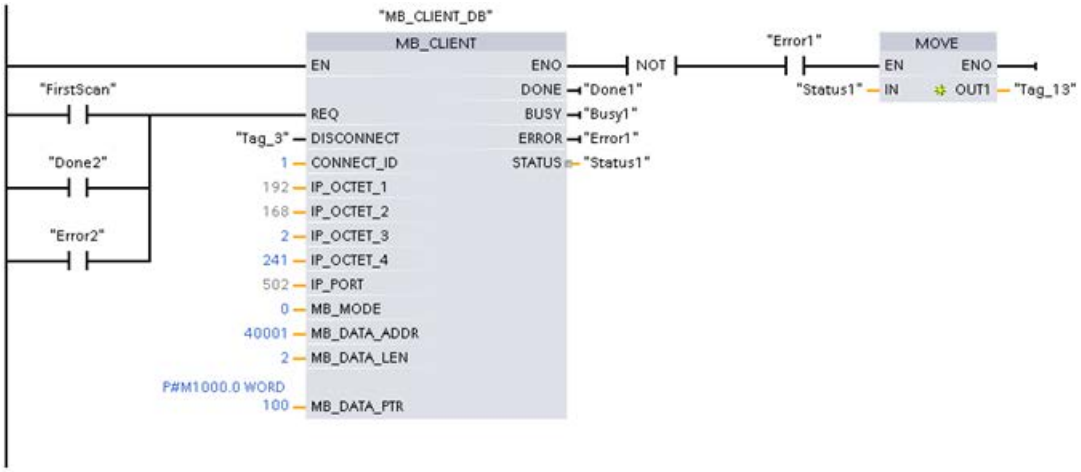


MB_CLIENT ejemplo 3: Coordinar varias peticiones

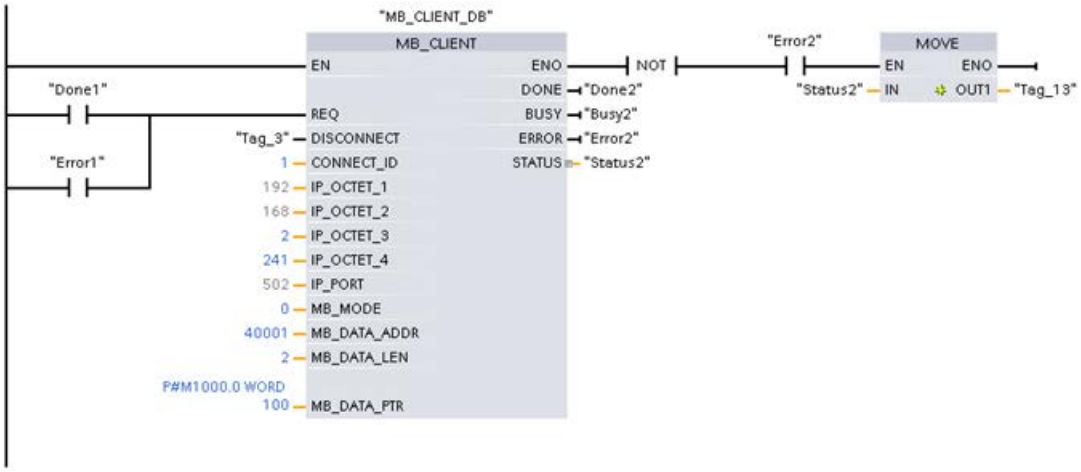
Descripción

Debe asegurarse de que se ejecuten cada una de las peticiones Modbus. La coordinación de las peticiones se controla mediante el programa. El siguiente ejemplo muestra cómo pueden utilizarse los parámetros de salida de la primera y segunda petición de cliente para coordinar la ejecución de las instrucciones.

Segmento 1: Función Modbus 3: leer registro de parada (WORD)



Segmento 2: Función Modbus 3: leer registro de parada (WORD)



MB_SERVER Ejemplo: Varias conexiones TCP

Descripción

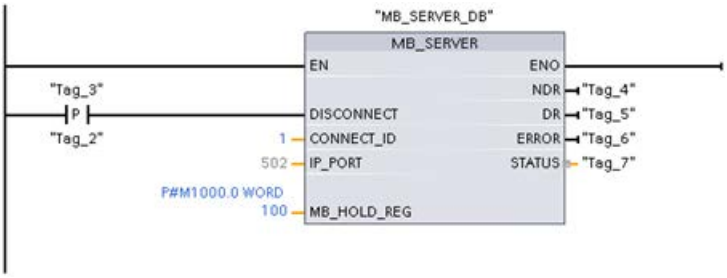
Pueden usarse varias conexiones de servidor Modbus TCP. Para ello, para cada conexión debe llamarse separadamente la instrucción "MB_SERVER".

Cada conexión requiere:

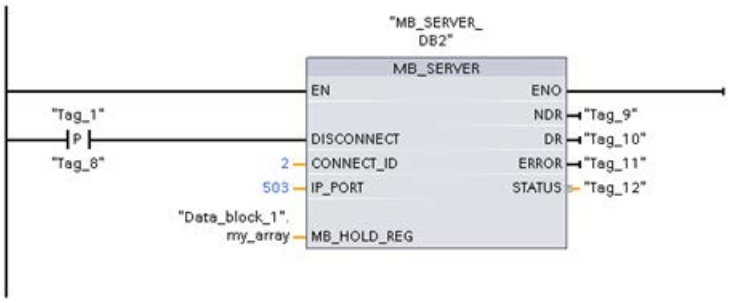
- un bloque de datos de instancia independiente de la instrucción
- una ID de conexión unívoca
- un puerto IP propio (con S7-1200 solo se permite una conexión por cada puerto IP)

Para obtener un rendimiento óptimo, ejecute "MB_SERVER" para cada conexión una vez por cada ciclo de programa.

Segmento 1: conexión #1 con el correspondiente puerto IP, ID de conexión y DB de instancia



Segmento 2: conexión #1 con el correspondiente puerto IP, ID de conexión y DB de instancia



MODBUS (TCP)

MB_CLIENT: Comunicarse como cliente Modbus TCP vía PROFINET

Descripción MB_CLIENT

Descripción

La instrucción "MB_CLIENT" permite la comunicación como cliente Modbus TCP a través de la conexión PROFINET. La instrucción "MB_CLIENT" permite establecer una conexión entre el cliente y el servidor, enviar peticiones Modbus y recibir respuestas, así como controlar la desconexión del cliente Modbus TCP.

La instrucción "MB_CLIENT" V3.0 puede utilizarse tanto para el S7-1500 como para el S7-1200 a partir de la versión 4.0. La conexión puede realizarse a través de la interfaz local de la CPU o CM/CP.

Para utilizar esta instrucción no se requiere ningún módulo de hardware adicional.

Conexiones múltiples a cliente

Un cliente Modbus TCP puede admitir varias conexiones TCP (el número máximo de conexiones depende de la CPU utilizada). El total de conexiones de una CPU, incluidos los clientes Modbus TCP y los servidores, no debe exceder el número máximo de conexiones admitido. Las conexiones Modbus TCP también pueden ser utilizadas conjuntamente por instancias de "MB_CLIENT" y/o "MB_SERVER".

En algunas conexiones de cliente deben respetarse las siguientes reglas:

- Cada conexión "MB_CLIENT" debe utilizar un DB de instancia unívoco.
- Para cada conexión "MB_CLIENT" debe especificarse una dirección IP unívoca del servidor.

- Cada conexión "MB_CLIENT" requiere una ID de conexión unívoca.
  Para cada DB de instancia de la instrucción debe utilizarse la correspondiente ID de conexión. Las ID de conexión y los DB de instancia se agrupan por pares y deben ser unívocos para cada conexión.
- Según la configuración del servidor, se requerirán o no números unívocos de puerto IP.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MB_CLIENT":

Parámetro	Declaración	Tipo de datos	Descripción
REQ (Página 5571)	Input	BOOL	Petición de comunicación con el servidor Modbus TCP El parámetro REQ se controla por nivel. Así, mientras la entrada esté activada (REQ=true), la instrucción enviará peticiones de comunicación. • Con la petición de comunicación se bloquea el DB de instancia para otros clientes. • Las modificaciones de los parámetros de entrada no se hacen efectivas hasta que no hay respuesta del servidor o hasta que no se devuelve un mensaje de error. • Si durante una petición Modbus en curso se vuelve a activar el parámetro REQ, a continuación no se ejecuta ninguna otra transferencia.
DISCONNECT (Página 5571)	Input	BOOL	Mediante este parámetro se controla el establecimiento de la conexión y la desconexión con el servidor Modbus: • 0: Establecer conexión de comunicación con el interlocutor configurado en el parámetro CONNECT (ver parámetro CONNECT). • 1: Deshacer la conexión. Durante la desconexión no se ejecuta ninguna otra función. Tras deshacer la conexión correctamente, el parámetro STATUS devuelve el valor 0003. Si el parámetro REQ está activado mientras se establece la conexión, la petición Modbus se envía de inmediato.
MB_MODE (Página 5571)	Input	USINT	Selección del modo de petición Modbus (lectura, escritura o diagnóstico).
MB_DATA_ADDR (Página 5571)	Input	UDINT	Dirección inicial de los datos a los que accede la instrucción "MB_CLIENT".
MB_DATA_LEN	Input	UINT	Longitud de datos: Número de bits o palabras para el acceso a los datos (ver "Parámetros MB_MODE y MB_DATA_ADDR": longitud de datos).
MB_DATA_PTR (Página 5574)	InOut	VARIANT	Puntero al registro de datos Modbus: El registro es un búfer para los datos recibidos desde el servidor Modbus o que se van a enviar al servidor Modbus. El puntero debe remitir a un bloque de datos global de acceso optimizado. El número de bits direccionados debe ser divisible entre 8.

Parámetro	Declaración	Tipo de datos	Descripción
CONNECT (Página 5575)	InOut	VARIANT	Puntero hacia la estructura de la descripción de la conexión Se pueden utilizar las siguientes estructuras (tipos de datos de sistema): • TCON_IP_v4: contiene todos los parámetros de direccionamiento necesarios para establecer una conexión programada. Si se utiliza TCON_IP_v4, la conexión se establece al llamar la instrucción "MB_SERVER". • TCON_Configured: contiene los parámetros de direccionamiento de una conexión configurada. En el caso de utilizar TCON_Configured se emplea una conexión existente, establecida después de que la CPU cargara la configuración hardware.
DONE	Out	BOOL	El bit del parámetro de salida DONE se pone a "1" en cuanto se ha ejecutado sin errores la última petición.
BUSY	Out	BOOL	• 0: Ninguna petición Modbus en proceso • 1: La petición Modbus se está ejecutando El parámetro de salida BUSY no se activa al establecer o al deshacer la conexión.
ERROR	Out	BOOL	• 0: Sin errores • 1: Con errores. La causa del error se indica mediante el parámetro STATUS.
STATUS (Página 5577)	Out	WORD	Información de estado detallada de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Datos de entrada coherentes durante una llamada de "MB_CLIENT"

En cuanto se ejecuta una instrucción de cliente Modbus, el estado de los parámetros de entrada se almacena internamente y luego se compara en la siguiente llamada. La comparación se utiliza para determinar si es esa llamada concreta la que ha iniciado la petición en curso. Pueden realizarse varias llamadas de "MB_CLIENT" utilizando un DB de instancia común. Los valores de los parámetros de entrada no deben modificarse mientras esté ejecutándose una instancia "MB_CLIENT". Si se modifican los parámetros de entrada durante la ejecución, no se puede comprobar mediante "MB_CLIENT" si la instancia se está ejecutando en el momento actual.

Variables estáticas de la instrucción

En la siguiente tabla se describen las variables estáticas editables del bloque de datos de instancia de la instrucción "MB_CLIENT".

Variable	Tipo de datos	Valor de arranque	Descripción
Blocked_Proc_Timeout	REAL	3.0	Tiempo en segundos que hay que esperar para que se inicialice la variable estática ACTIVE en una instancia Modbus bloqueada. Esto puede suceder, por ejemplo, cuando, habiéndose emitido una petición de cliente, se cancela la ejecución de la función de cliente antes de que se haya ejecutado por completo la petición. El tiempo de espera máximo es de 55 segundos.
MB_Transaction_ID	WORD	1	ID de transacción del protocolo Modbus TCP. El valor de arranque "1" solo debe modificarse en caso de que el servidor Modbus TCP necesite un valor diferente.
MB_Unit_ID	BYTE	255	ID de Unit del protocolo Modbus. La variable corresponde a la dirección de esclavo del protocolo Modbus RTU. Modifique este valor solo si el servidor Modbus TCP puede utilizarse como gateway y si es controlado por el programa de usuario dentro del servidor Modbus.
RCV_TIMEOUT	REAL	2.0	Intervalo en segundos durante el que la instrucción "MB_CLIENT" permanece a la espera de una respuesta del servidor.
Connected	BOOL	0	Indica si la conexión con el servidor asignado está establecida o no: 1 = conectado, 0 = no conectado.

Parámetros REQ y DISCONNECT

Descripción

Si no se ejecuta ninguna instancia de la instrucción "MB_CLIENT" y si el parámetro DISCONNECT tiene el valor "0", con REQ=1 se ejecuta otra petición. Si todavía no se ha establecido ninguna conexión, esta se establecerá al ejecutarse la instrucción.

Si se vuelve a ejecutar la misma instancia de la instrucción "MB_CLIENT" (DISCONNECT=0 y REQ=1) antes de que se ejecute la petición activa, esta no se ejecutará a continuación de la petición activa. Solo podrá iniciarse una nueva petición una vez que haya finalizado la petición activa (REQ=1).

El estado de la ejecución se emite a través de los parámetros de salida. Estos se pueden utilizar para vigilar el estado de ejecución durante la ejecución secuencial de la instrucción "MB_CLIENT".

Parámetros MB_MODE y MB_DATA_ADDR

Descripción

La instrucción "MB_CLIENT" utiliza, en lugar de un código de función, el parámetro MB_MODE. Mediante el parámetro MB_DATA_ADDR se define la dirección de arranque Modbus de los datos a los que se desea acceder.

La combinación de los parámetros MB_MODE, MB_DATA_ADDR y MB_DATA_LEN determina el código de función que se utiliza en el mensaje Modbus actual. Por ejemplo:

- Código de función 5
  - MB_MODE=1
  - MB_DATA_ADDR=1
  - MB_DATA_LEN=1
- Código de función 15
  - MB_MODE=1
  - MB_DATA_ADDR=1
  - MB_DATA_LEN=2

La siguiente tabla muestra la relación entre los parámetros de entrada de la instrucción "MB_CLIENT" y la función Modbus.

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Función Modbus	Función y tipo de datos
0	Dirección inicial: • de 1 a 9999	Longitud de datos (bits) por llamada: • de 1 a 2000	01	Leer bits de entrada: • de 1 a 2000
0	Dirección inicial: • de 10001 a 19999	Longitud de datos (bits) por llamada: • de 1 a 2000	02	Leer bits de entrada: • de 1 a 2000
0	Dirección inicial: • de 40001 a 49999 • de 400001 a 465535	Longitud de datos (WORD) por llamada: • de 1 a 125 • de 1 a 125	03	Leer registro de parada: • de 0 a 9998 • de 0 a 65534
0	Dirección inicial: • de 30001 a 39999	Longitud de datos (WORD) por llamada: • de 1 a 125	04	Leer palabras de entrada: • de 0 a 9998
1	Dirección inicial: • de 1 a 9999	Longitud de datos (bits) por llamada: • 1	05	Escribir un bit de salida: • de 0 a 9998

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Función Modbus	Función y tipo de datos
1	Dirección inicial: • de 40001 a 49999 • de 400001 a 465535	Longitud de datos (WORD) por llamada: • 1 • 1	06	Escribir un registro de parada: • de 0 a 9998 • de 0 a 65524
1	Dirección inicial: • de 1 a 9999	Longitud de datos (bits) por llamada: • de 2 a 1968	15	Escribir varios bits de salida: • de 0 a 9998
1	Dirección inicial: • de 40001 a 49999 • de 400001 a 465534	Longitud de datos (WORD) por llamada: • de 2 a 123 • de 2 a 123	16	Escribir varios registros de parada: • de 0 a 9998 • de 0 a 65534
2	Dirección inicial: • de 1 a 9999	Longitud de datos (bits) por llamada: • de 1 a 1968	15	Escribir uno o varios bits de salida: • de 0 a 9998
2	Dirección inicial: • de 40001 a 49999 • de 400001 a 465535	Longitud de datos (WORD) por llamada: • de 1 a 123 • de 1 a 123	16	Escribir uno o varios registros de parada: • de 0 a 9998 • de 0 a 65534
11	Los parámetros MB_DATA_ADDR y MB_DATA_LEN no se evalúan al ejecutar esta función.		11	Leer la palabra de estado y el contador de eventos del servidor: • La palabra de estado forma el estado de ejecución (0: no en ejecución, 0xFFFF: en ejecución). • El contador de eventos se incrementa si la petición Modbus se ha ejecutado correctamente. Si se produce un error al ejecutar una función Modbus, el servidor envía un mensaje, pero el contador de eventos no se incrementa.
80	-	Longitud de datos (WORD) por llamada: • 1	08	Comprobación del estado del servidor por el código de diagnóstico 0x0000 (test de bucle de retorno: el servidor devuelve la petición): • 1 WORD por llamada

MB_MODE	MB_DATA_ADDR	MB_DATA_LEN	Función Modbus	Función y tipo de datos
81	-	Longitud de datos (WORD) por llamada: • 1	08	Reiniciar el contador de eventos del servidor mediante el código de diagnóstico 0x000A: • 1 WORD por llamada
de 3 a 10, de 12 a 79, de 82 a 255				Reservado

Parámetro MB_DATA_PTR

Descripción

El parámetro MB_DATA_PTR es un puntero hacia un búfer de datos para almacenar los datos que se han leído o escrito en el servidor Modbus. Puede usarse como búfer de datos un bloque de datos global o un área de memoria (M).

Para un búfer en el área de memoria (M), utilice un puntero con el formato ANY según el patrón"P#dirección_de_bit" "Tipo de datos" "Longitud" (ejemplo: P#M1000.0 WORD 500).

El parámetro MB_DATA_PTR utiliza un búfer de comunicación:

El búfer de comunicación se utiliza:

- Para las funciones de comunicación de la instrucción "MB_CLIENT":
  - Lectura y escritura de datos de 1 bit de las direcciones de servidor Modbus de 00001 a 09999 y de 10001 a 19999.
  - Lectura de datos WORD de 16 bits de las direcciones del servidor Modbus de 30001 a 39999 y de 40001 a 49999.
  - Escritura de datos WORD de 16 bits de las direcciones del servidor Modbus de 40001 a 49999.
- Para la transferencia de datos (longitud: bit o WORD) desde o hacia el DB global o el área de memoria (M) que se ha asignado mediante el parámetro MB_DATA_PTR.

Si se utiliza en el parámetro MB_DATA_PTR un bloque de datos para el puntero al búfer, deben asignarse tipos de datos a los elementos de DB.

- Para una dirección de bit Modbus, utilice el tipo de datos de 1 bit BOOL
- Para una dirección Modbus WORD, utilice un tipo de datos de 16 bits como WORD, UINT, INT o REAL.
- Para dos direcciones Modbus WORD, utilice un tipo de datos de 32 bits (palabra doble) como DWORD, DINT o REAL.

- Mediante MB_DATA_PTR se puede acceder también a elementos de DB complejos como:
  - Matrices estándar
  - Estructuras con nombres de elemento unívocos
  - Estructuras complejas con denominación unívoca de los elementos longitudes de tipo de datos de 16 ó 32 bits.
- Las áreas de datos para el parámetro MB_DATA_PTR pueden encontrarse también en distintos bloques de datos globales (o en distintas áreas de memoria). Por ejemplo, puede utilizarse un bloque de datos para las operaciones de lectura y otro para las operaciones de escritura, o un bloque de datos independiente para cada estación "MB_CLIENT".

Parámetro CONNECT

Descripción de conexiones en el parámetro CONNECT

Para la instrucción "MB_CLIENT" se pueden utilizar dos descripciones de conexión distintas:

- Conexiones programadas con la estructura TCON_IP_v4
  Los parámetros de conexión se guardan en la estructura TCON_IP_v4 y la conexión no se establece hasta que no se produce la llamada de la instrucción "MB_CLIENT".
- Conexiones configuradas con la estructura TCON_Configured
  La conexión configurada ya ha sido establecida por la CPU. Con la estructura TCON_Configured se indica qué conexión existente debe usarse para la instrucción.

Cada instancia de la instrucción "MB_CLIENT" necesita una conexión unívoca. Por tanto, establezca para cada instancia de la instrucción una estructura propia TCON_IP_v4 o TCON_Configured para la descripción de la conexión.

Descripción de conexiones programadas

Para conexiones programadas, utilice en el parámetro CONNECT la siguiente estructura para la descripción de la conexión según TCON_IP_v4:

- Recuerde indicar en la estructura TCON_IP_v4 solamente conexiones del tipo TCP.
- La conexión no debe utilizar los siguientes números de puerto TCP: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceID	HW_ANY	-	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	-	Referencia a esta conexión (rango de valores: de 1 a 4095). Mediante este parámetro se identifica una conexión en la CPU de modo unívoco. Cada instancia individual de la instrucción "MB_CLIENT" debe utilizar una ID unívoca.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
4	ConnectionType	BYTE	11	Tipo de conexión Seleccione 11 (decimal) para TCP. No se permite ningún otro tipo de conexión. Si se utiliza otro tipo de conexión (p. ej. UDP), el parámetro STATUS de la instrucción emitirá el mensaje de error correspondiente.
5	ActiveEstablished	BOOL	TRUE	Identificador del tipo de establecimiento de conexión Seleccione TRUE para el establecimiento activo de la conexión.
6 … 9	RemoteAddress	ARRAY [1..4] of BYTE	-	Dirección IP del interlocutor (servidor Modbus), p. ej. para 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1
10 … 11	RemotePort	UINT	502	Número de puerto del interlocutor remoto (rango de valores: de 1 a 49151). Utilice el número de IP y de puerto del servidor con el que el cliente establece la conexión y con el que se comunica mediante el protocolo TCP/IP (valor estándar: 502).
12 … 13	LocalPort	UINT	0	Número de puerto del interlocutor local: • Números de puerto: de 1 a 49151 • Cualquier puerto: "0"

Nota

Migración de la instrucción "MB_CLIENT" versión 2.1

Los parámetros CONNECT_ID, IP_PORT y IP_OCTET_x se mapean en la versión 3.0 de la instrucción "MB_CLIENT" en la estructura TCON_IP_v4:

- El parámetro CONNECT_ID de la instrucción "MB_CLIENT" V2.1 equivale al parámetro ID de TCON_IP_v4.
- El parámetro IP_PORT de la instrucción "MB_CLIENT" V2.1 equivale al parámetro RemotePort de TCON_IP_v4.
- Los cuatro parámetros IP_OCTET_x de la instrucción "MB_CLIENT" V2.1 equivalen a la matriz del parámetro RemoteAddress de TCON_IP_v4.

Descripción de conexiones configuradas

Para conexiones programadas, utilice en el parámetro CONNECT la siguiente estructura para la descripción de la conexión según TCON_Configured..

- Recuerde indicar en la estructura TCON_Configured solamente conexiones del tipo TCP.
- La conexión no debe utilizar los siguientes números de puerto TCP: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceId	HW_ANY	-	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	-	Referencia a esta conexión (rango de valores: de 1 a 4095). Indique la ID de la conexión existente.
4	ConnectionType	BYTE	0	Tipo de conexión Seleccione 254 (decimal) para una conexión configurada.

Parámetro STATUS

Parámetro STATUS (información de estado general)

STATUS* (W#16#)	Descripción
0000	Instrucción ejecutada sin errores.
0001	Conexión establecida.
0003	Desconexión realizada.
7000	Ninguna llamada activa y ninguna conexión establecida (REQ=0).
7001	Establecimiento de conexión iniciado.
7002	Llamada intermedia. La conexión se está estableciendo.
7003	Deshaciendo la conexión.
7004	Conexión establecida y vigilada. No se está procesando ninguna petición.
7005	Enviando datos.
7006	Recibiendo datos.
80BB	Valor no válido en el parámetro ActiveEstablished (identificador del tipo de establecimiento de conexión, véase Parámetro CONNECT (Página 5575)): • El lado servidor solo permite el establecimiento pasivo de la conexión (ActiveEstablished = FALSE). • El lado cliente solo permite el establecimiento activo de la conexión (ActiveEstablished = TRUE).
8380	El frame Modbus recibido no tiene el formato correcto o la cantidad de bytes recibidos no es suficiente.
* Los códigos de estado en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (error de protocolo)

STATUS* (W#16#)	Código de error en el mensaje de error de MB_SERVER (B#16#)	Descripción
8381	01	Código de función no soportado.
8382	03	Error en la longitud de datos: • Indicación de longitud no admisible en el frame Modbus recibido. • La longitud del frame Modbus indicada en el encabezado del frame no coincide con la cantidad de bytes recibidos. • La cantidad de bytes no coincide con los bytes realmente transferidos (solo funciones 1-4). • La dirección inicial recibida no coincide con la enviada originalmente (funciones 5, 6, 15, 16). • La cantidad de palabras no coincide con las palabras realmente transferidas (funciones 15 y 16).
8383	02	Error al leer o escribir los datos o acceso fuera del área de direcciones de MB_DATA_PTR (Página 5574). El error puede aparecer tanto localmente como en la instrucción "MB_SERVER".
8384	03	Error en el valor de datos: • Error en el valor de datos para la función 5 (error en el lado servidor). • Se ha recibido otro valor de datos distinto al enviado originalmente por el cliente (funciones 5 y 6) (error local). • El Exception Code recibido no es válido.
8385	03	Código de diagnóstico no soportado (código de función 08). El error puede aparecer tanto localmente como en el lado servidor.
8386	-	El código de función recibido no coincide con el enviado originalmente.
8387	-	• La ID de conexión asignada es distinta de las de anteriores peticiones. Solo puede usarse una ID de conexión para cada DB de instancia de la instrucción "MB_CLIENT". • El código de error se emite aunque la ID del protocolo del frame Modbus TCP recibido por el servidor sea distinta de "0".
8388	15 o 16	Se ha enviado a través del servidor Modbus una longitud de datos distinta de la solicitada. Este error solo se produce si se utilizan las funciones Modbus 15 o 16.
* Los códigos de estado en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Parámetro STATUS (error de parámetro)

STATUS* (W#16#)	Descripción
80B6	Tipo de conexión no válido, solo se soportan conexiones TCP.
80C8	No ha habido respuesta del servidor en el periodo definido. Compruebe la conexión con el servidor Modbus. Este error no se notifica hasta que ha no finalizado el número configurado de reintentos. Si la instrucción "MB_CLIENT" no obtiene respuesta alguna con la ID de transacción transmitida originalmente (ver variable estática MB_TRANSACTION_ID) dentro del período definido, entonces se emitirá este código de error.
8188	Valor no válido en el parámetro MB_MODE.
8189	Direccionamiento no válido de los datos en el parámetro MB_DATA_ADDR.
818A	Longitud de datos no válida en el parámetro MB_DATA_LEN.
818B	Puntero no válido en el parámetro MB_DATA_PTR. Compruebe además los valores de los parámetros MB_DATA_ADDR (Página 5574) y MB_DATA_LEN.
818C	Rebase de tiempo en el parámetro BLOCKED_PROC_TIMEOUT o RCV_TIMEOUT (ver variables estáticas de la instrucción). Se ha rebasado el límite de 55 segundos.
818D	La ID de transacción (variable MB_TRANSACTION_ID) no coincide con la enviada originalmente (ver variables estáticas de la instrucción).
8200	• En este momento se está procesando otra petición Modbus a través del puerto. • Otra instancia de MB_CLIENT con los mismos parámetros de conexión está procesando una petición Modbus ya existente.
* Los códigos de estado en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Nota

Códigos de error de las instrucciones de comunicación utilizadas internamente

Además de los errores listados en la tabla, con la instrucción "MB_CLIENT" pueden producirse además errores en las instrucciones de comunicación ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" y "TRESET") utilizadas por la instrucción.

La asignación del código de error se puede realizar a través del bloque de datos de instancia de la instrucción "MB_CLIENT". En la sección "Static" se indican los códigos de error en la instrucción correspondiente bajo STATUS.

El significado de los códigos de error se puede consultar en la documentación de la instrucción de comunicación correspondiente.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

MB_SERVER: comunicarse como servidor Modbus TCP vía PROFINET

Descripción MB_SERVER

Descripción

La instrucción "MB_SERVER" permite la comunicación como servidor Modbus TCP a través de una conexión PROFINET. La instrucción "MB_SERVER" permite procesar peticiones de conexión de un cliente Modbus TCP, recibir y procesar peticiones Modbus y enviar mensajes de respuesta.

La instrucción "MB_SERVER" V3.0 puede utilizarse tanto para el S7-1500 como para el S7-1200 a partir de la versión 4.0. La conexión puede realizarse a través de la interfaz local de la CPU o CM/CP.

Para utilizar esta instrucción no se requiere ningún módulo de hardware adicional.

ATENCIÓN
Consigna de seguridad Tenga en cuenta que cada cliente de la red tiene acceso a los datos de las instrucciones Modbus. Si bien el bloque de datos para el registro de parada Modbus no se puede direccionar directamente con el cliente, existe la posibilidad de acceder indirectamente a los datos a través de las direcciones de offset. Para evitar que personas no autorizadas lleven a cabo procesos de lectura y escritura, es posible restringir el acceso a una dirección IP. No obstante, tenga en cuenta que la dirección habilitada también ofrece acceso a personas no autorizadas.

Conexiones múltiples a servidor

Pueden establecerse conexiones múltiples a servidor. Gracias a ello, una sola CPU puede aceptar conexiones de varios clientes Modbus TCP al mismo tiempo.

Un servidor Modbus TCP puede admitir varias conexiones TCP (el número máximo de conexiones depende de la CPU utilizada).

El total de conexiones de una CPU, incluidos los clientes Modbus TCP y los servidores, no debe exceder el número máximo de conexiones admitido.

Las conexiones Modbus TCP también pueden ser utilizadas conjuntamente por instancias de "MB_CLIENT" y/o "MB_SERVER".

Para las conexiones de servidor deben respetarse las siguientes reglas:

- Cada conexión "MB_SERVER" debe utilizar un DB de instancia unívoco.
- Cada conexión "MB_SERVER" debe utilizar una ID de conexión unívoca.
  Para cada DB de instancia de la instrucción debe utilizarse la correspondiente ID de conexión. Las ID de conexión y los DB de instancia se agrupan por pares y deben ser unívocos para cada conexión.
- Para cada conexión debe llamarse separadamente la instrucción "MB_SERVER".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MB_SERVER":

Parámetro	Declaración	Tipo de datos	Descripción
DISCONNECT	Input	BOOL	La instrucción "MB_SERVER" establece una conexión pasiva con un módulo interlocutor. El servidor reacciona a una petición de conexión de la dirección IP indicada en el SDT "TCON_IP_v4" en el parámetro CONNECT. Este parámetro permite controlar cuándo se aceptará una petición de conexión: • 0: Si no hay ninguna conexión establecida, se establece una conexión pasiva. • 1: Inicialización del establecimiento de la conexión. Si la entrada esta activada, no se ejecutan otras operaciones. Tras deshacer la conexión correctamente, el parámetro STATUS devuelve el valor 0003.
MB_HOLD_REG (Página 5584)	InOut	VARIANT	Puntero al registro de parada Modbus de la instrucción "MB_SERVER" El registro de parada contiene los valores a los que está autorizado a acceder un cliente Modbus mediante las funciones Modbus 3 (lectura), 6 (escritura) y 16 (lectura). Utilice como registro de parada un bloque de datos global de acceso optimizado o el área de memoria de marcas.
CONNECT (Página 5584)	InOut	VARIANT	Puntero hacia la estructura de la descripción de la conexión Se pueden utilizar las siguientes estructuras (SDT): • TCON_IP_v4: contiene todos los parámetros de direccionamiento necesarios para establecer una conexión programada. De forma estándar es 0.0.0.0 (cualquier dirección IP), pero también se puede indicar una dirección IP determinada de modo que el servidor solo reaccione a las peticiones de dicha dirección. Si se utiliza TCON_IP_v4, la conexión se establece al llamar la instrucción "MB_SERVER". • TCON_Configured: contiene los parámetros de direccionamiento de una conexión configurada. En caso de utilizar TCON_Configured, la conexión se establece una vez que la CPU haya cargado la configuración hardware.
NDR	Output	BOOL	"New Data Ready": • 0: No hay datos nuevos • 1: El cliente Modbus ha escrito datos nuevos
DR	Output	BOOL	"Data Read": • 0: No se han leído datos • 1: El cliente Modbus ha leído datos
ERROR	Output	BOOL	Si se produce un error durante una llamada de la instrucción "MB_SERVER", la salida del parámetro ERROR se pone a "1". La causa detallada del error se muestra en el parámetro STATUS.
STATUS (Página 5586)	Output	WORD	Información de estado detallada de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Variables estáticas de la instrucción

En la tabla siguiente se describen las variables estáticas del bloque de datos de instancia de la instrucción MB_SERVER que se pueden utilizar en el programa. A la variable HR_Start_Offset se puede acceder en escritura. A las demás variables se puede acceder en lectura para vigilar el estado del Modbus.

Variable	Tipo de datos	Valor de arranque	Descripción
HR_Start_Offset	WORD	0	Asignar la dirección inicial del registro de parada de Modbus.
Request_Count	WORD	0	Número total de peticiones recibidas por el servidor.
Server_Message_Count	WORD	0	Número total de mensajes recibidos para el servidor en cuestión.
Xmt_Rcv_Count	WORD	0	Contador para el registro del número de transferencias en las que se ha producido un error. El contador solo se incrementa cuando se recibe una petición Modbus no válida.
Exception_Count	WORD	0	Contador para registrar el número de errores específicos de Modbus que dan lugar a un mensaje de error en "MB_CLIENT".
Success_Count	WORD	0	Contador de eventos para registrar el número de peticiones que el servidor ha podido ejecutar correctamente.
Connected	BOOL	0	Indica si la conexión con el cliente asignado está establecida o no: 1 = conectado, 0 = no conectado.

Reproducir las direcciones Modbus en la memoria imagen de proceso

La instrucción "MB_SERVER" permite a las funciones Modbus entrantes (1, 2, 4, 5 y 15) acceder directamente en lectura y escritura a la memoria imagen de proceso de las entradas y salidas de la CPU (usando los tipos de datos BOOL y WORD).

Para la transferencia de datos de los códigos de función 3, 6 y 16, el registro de parada (parámetro MB_HOLD_REG) debe definirse con una longitud de más de un byte. La tabla siguiente muestra la imagen de las direcciones Modbus en la memoria imagen de proceso de la CPU.

Función Modbus						S7-1500, S7-1200 V4.0	
Código de función	Función	Área de datos	Área de direcciones			Área de datos	Dirección de la CPU
01	Leer: bits	Output	1	a	9999	Memoria imagen de proceso de las salidas	Q0.0 a Q1249.6
02	Leer: bits	Input	1	a	19999	Memoria imagen de proceso de las entradas	I0.0 a I1249.6
04	Leer: WORD	Input	1	a	39999	Memoria imagen de proceso de las entradas	De IW0 a IW19996
05	Escribir: bits	Output	1	a	9999	Memoria imagen de proceso de las salidas	Q0.0 a Q1249.6
15	Escribir: bits	Output	1	a	9999	Memoria imagen de proceso de las salidas	Q0.0 a Q1249.6

Las peticiones Modbus entrantes con los códigos de función 3, 6, 16 y 23 escriben o leen del registro de parada de Modbus (el registro de parada debe especificarse en el parámetro MB_HOLD_REG).

Ejemplo: Direccionamiento a través de variable estática HR_Start_Offset

Las direcciones del registro de parada Modbus empiezan con 40001. Estas direcciones corresponden al espacio de direcciones del área de memoria de la CPU para el registro de parada. También es posible definir la variable HR_Start_Offset de modo que el registro de parada de Modbus reciba una dirección inicial distinta de 40001.

Ejemplo: El registro de parada empieza en el MW100 y tiene una longitud de 100 WORD. Con un valor de offset en el parámetro HR_Start_Offset, la dirección inicial del registro de parada se desplaza de 40001 a 40021. Todo direccionamiento del registro de parada por debajo de la dirección 40021 y por encima de la dirección 40120 da lugar a un error.

HR_Start_Offset	Dirección	Mínimo	Máximo
0	Dirección Modbus (WORD)	40001	40100
	Dirección de la CPU	MW100	MW298
20	Dirección Modbus (WORD)	40021	40120
	Dirección de la CPU	MW100	MW298

Funciones de diagnóstico Modbus

La tabla siguiente contiene la descripción de las funciones de diagnóstico de Modbus.

Código de función	Código de diagnóstico	Descripción
08	0x0000	Prueba de eco: La instrucción "MB_SERVER" recibe una palabra de datos y la devuelve inalterada al cliente Modbus.
08	0x000A	Inicializar el contador de eventos: La instrucción "MB_SERVER" resetea los siguientes contadores de eventos: "Success_Count", "Xmt_Rcv_Count", "Exception_Count", "Server_Message_Co" y "Request_Count".
11	-	Recoger contador de eventos de la comunicación: Durante la comunicación, la instrucción "MB_SERVER" utiliza un contador de eventos interno para registrar el número de peticiones de lectura y escritura correctas que se envían al servidor Modbus. El contador de eventos no se incrementa en las funciones 8 u 11. Lo mismo se aplica a las peticiones que tengan como consecuencia un error de comunicación, por ejemplo en el caso de un error de protocolo (p. ej. el código de función en la petición Modbus recibida no es compatible).

Parámetro MB_HOLD_REG

Descripción

El parámetro MB_HOLD_REG es un puntero hacia un búfer de datos para almacenar los datos que se han leído o escrito en el servidor Modbus. Como área de memoria puede utilizarse un bloque de datos global o una marca (M).

- El límite superior del número de direcciones en el bloque de datos (D) viene determinado por la memoria de trabajo máxima de la CPU.
- El límite superior del número de marcas (M) viene determinado por el tamaño del área de memoria de la CPU.

En la siguiente tabla se muestran ejemplos del mapeo de direcciones Modbus en el registro de parada para las funciones Modbus 3 (leer WORD), 6 (escribir WORD), 16 (escribir varias WORD) y 23 (escribir y leer varias palabras).

Direcciones Modbus	Parámetro MB_HOLD_REG: ejemplos		
40001	MW100	DB10.DBW0	"Recipe".ingredient[1]
40002	MW102	DB10.DBW2	"Recipe".ingredient[2]
40003	MW104	DB10.DBW4	"Recipe".ingredient[3]
40004	MW106	DB10.DBW6	"Recipe".ingredient[4]
40005	MW108	DB10.DBW8	"Recipe".ingredient[5]

Parámetro CONNECT

Descripción de conexiones en el parámetro CONNECT

Para la instrucción "MB_SERVER" se pueden utilizar dos descripciones de conexión distintas:

- Conexiones programadas con la estructura TCON_IP_v4
  Los parámetros de conexión se guardan en la estructura TCON_IP_v4 y la conexión no se configura hasta que no se produce la llamada de la instrucción "MB_SERVER".
- Conexiones configuradas con la estructura TCON_Configured
  La conexión configurada ya ha sido establecida por la CPU. Con la estructura TCON_Configured se indica qué conexión existente debe usarse para la instrucción.

Cada instancia de la instrucción "MB_SERVER" necesita una conexión unívoca. Por tanto, establezca para cada instancia de la instrucción una estructura propia TCON_IP_v4 o TCON_Configured para la descripción de la conexión.

Descripción de conexiones programadas

Para conexiones programadas, utilice en el parámetro CONNECT la siguiente estructura para la descripción de la conexión según TCON_IP_v4..

- Recuerde indicar en la estructura TCON_IP_v4 solamente conexiones del tipo TCP.
- La conexión no debe utilizar los siguientes números de puerto TCP: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceID	HW_ANY	-	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	-	Referencia a esta conexión (rango de valores: de 1 a 4095). Mediante este parámetro se identifica una conexión en la CPU de modo unívoco. Cada instancia individual de la instrucción "MB_SERVER" debe utilizar una ID unívoca.
4	ConnectionType	BYTE	11	Tipo de conexión Seleccione 11 (decimal) para TCP. No se permite ningún otro tipo de conexión. Si se utiliza otro tipo de conexión (p. ej. UDP), el parámetro STATUS de la instrucción emitirá el mensaje de error correspondiente.
5	ActiveEstablished	BOOL	FALSE	Identificador del tipo de establecimiento de conexión Seleccione FALSE para el establecimiento pasivo de la conexión.
6 … 9	RemoteAddress	ARRAY [1..4] of BYTE	0.0.0.0	Dirección IP del interlocutor, p. ej. para 192.168.0.1: • addr[1] = 192 • addr[2] = 168 • addr[3] = 0 • addr[4] = 1 Si la instrucción "MB_SERVER" debe aceptar peticiones de conexión de cualquier interlocutor, utilice "0.0.0.0" como dirección IP.
10 … 11	RemotePort	UINT	0	Número de puerto del interlocutor remoto (rango de valores: de 1 a 49151). Si la instrucción "MB_SERVER" debe aceptar peticiones de conexión de cualquier interlocutor remoto, utilice "0" como número de puerto.
12 … 13	LocalPort	UINT	502	Número de puerto del interlocutor local (rango de valores: de 1 a 49151). El número del puerto IP determina qué puerto IP se vigila para peticiones de conexión del cliente Modbus. Los siguientes números de puerto TCP no deben usarse para la conexión pasiva de la instrucción "MB_SERVER": 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.

Nota

Migración de la instrucción "MB_SERVER" versión 2.1

Los parámetros CONNECT_ID y IP_PORT se mapean en la versión 3.0 de la instrucción "MB_SERVER" en la estructura TCON_IP_v4:

- El parámetro CONNECT_ID de la instrucción "MB_SERVER" V2.1 equivale al parámetro ID de TCON_IP_v4.
- El parámetro IP_PORT de la instrucción "MB_SERVER" V2.1 equivale al parámetro LocalPort de TCON_IP_v4.

Descripción de conexiones configuradas

Para conexiones configuradas, utilice en el parámetro CONNECT la siguiente estructura para la descripción de la conexión según TCON_Configured.

- Recuerde indicar en la estructura TCON_Configured solamente conexiones del tipo TCP.
- La conexión no debe utilizar los siguientes números de puerto TCP: 20, 21, 25, 80, 102, 123, 5001, 34962, 34963 y 34964.

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
0 … 1	InterfaceID	HW_ANY	-	Identificador de hardware de la interfaz local (rango de valores: de 0 a 65535).
2 … 3	ID	CONN_OUC	-	Referencia a esta conexión (rango de valores: de 1 a 4095). Indique la ID de la conexión existente.
4	ConnectionType	BYTE	-	Tipo de conexión Seleccione 254 (decimal) para una conexión configurada.

Parámetro STATUS

Parámetro STATUS (información de estado general)

STATUS* (W#16#)	Descripción
0000	Instrucción ejecutada sin errores.
0001	Conexión establecida.
0003	Desconexión realizada.
7000	Ninguna llamada activa (REQ=0).
7001	Primera llamada. Establecimiento de conexión iniciado.
7002	Llamada intermedia. La conexión se está estableciendo.
7003	Deshaciendo la conexión.
7005	Enviando datos.
7006	Recibiendo datos.

STATUS* (W#16#)	Descripción
80BB	Valor no válido en el parámetro ActiveEstablished (identificador del tipo de establecimiento de conexión, véase Parámetro CONNECT (Página 5584)): • El lado servidor solo permite el establecimiento pasivo de la conexión (active_established = FALSE). • El lado cliente solo permite el establecimiento activo de la conexión (active_established = TRUE).
8380	El frame Modbus recibido no tiene el formato correcto o la cantidad de bytes recibidos no es suficiente.
* Los códigos de estado en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Parámetro STATUS (error de parámetro)

STATUS* (W#16#)	Código de error en el mensaje de error de "MB_SERVER" (B#16#)	Descripción
8187	Sin respuesta	Puntero no válido en el parámetro MB_HOLD_REG. El área de datos es demasiado pequeña.
8381	01	Código de función no soportado.
8382	03	Error en la longitud de datos: • Indicación de longitud no admisible en el telegrama Modbus recibido • La longitud del frame indicada en el encabezado del frame Modbus no coincide con la cantidad de bytes realmente recibidos. • La cantidad de bytes indicada en el encabezado del frame Modbus no coincide con la cantidad de bytes realmente recibidos (funciones 15 y 16).
8383	02	Error en la dirección de los datos o acceso fuera del área de direcciones del registro de parada (parámetro MB_HOLD_REG (Página 5584)).
8384	03	Valor de datos no válido (función 5).
8385	03	Código de diagnóstico no soportado (solo con función 08).
* Los códigos de estado en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Nota

Códigos de error de las instrucciones de comunicación utilizadas internamente

Además de los errores listados en la tabla, con la instrucción "MB_SERVER" pueden producirse además errores en las instrucciones de comunicación ("TCON", "TDISCON", "TSEND", "TRCV", "T_DIAG" y "T_RESET") utilizadas por la instrucción.

La asignación del código de error se puede realizar a través del bloque de datos de instancia de la instrucción "MB_SERVER". En la sección "Static" se indican los códigos de error en la instrucción correspondiente bajo STATUS.

El significado de los códigos de error se puede consultar en la documentación de la instrucción de comunicación correspondiente.

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

GET_ERR_ID: Consultar ID de error localmente (Página 3531)

TeleService

TM_MAIL: Transmitir e-mail

Descripción de TM_MAIL

Descripción

La instrucción "TM_MAIL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Al llamar la instrucción "TM_MAIL" es imprescindible indicar una instancia. En el DB de instancia no se debe activar el atributo "Remanente". Con este atributo se garantiza que la instancia se inicialice en la transición de la CPU de STOP a RUN y que después se pueda lanzar una nueva petición para enviar un mensaje de correo electrónico.

El envío de un mensaje de correo electrónico se inicia con un cambio de flanco de "0" a "1" en el parámetro REQ. A través de los parámetros de salida "BUSY", "DONE", "ERROR", así como "STATUS" y "SFC_STATUS" se indica el estado de la petición. "SFC_STATUS" corresponde al parámetro de salida "STATUS" de los bloques de comunicación llamados.

Los parámetros de salida DONE, ERROR, STATUS y SFC_STATUS sólo se indican durante un ciclo si el estado del parámetro de salida BUSY cambia de "1" a "0". En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra la instrucción "TM_MAIL" actualmente o cuándo se completó la transmisión del mensaje de correo electrónico.

DONE	BUSY	ERROR	Descripción
0	1	0	La petición está en proceso.
1	0	0	La petición ha finalizado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se puede consultar en los parámetros STATUS y SFC_STATUS.
0	0	0	No se ha asignado ninguna (nueva) petición a la instrucción "TM_MAIL".

Si la CPU pasa al estado operativo STOP mientras "TM_MAIL" está activa, la conexión con el servidor de correo se deshace. La conexión con el servidor de correo también se pierde si

aparecen problemas de comunicación en el bus Industrial Ethernet. En esos casos, el envío del mensaje de correo electrónico se cancela y éste no llega al receptor.

ATENCIÓN
Cambiar programas de usuario Las partes del programa de usuario con efecto directo sobre llamadas de "TM_MAIL" sólo pueden modificarse en los siguientes casos: • Si la CPU está en estado operativo "STOP" • Si no se envía ningún mensaje (REQ = 0 y BUSY = 0). Esto se refiere en especial al borrado e intercambio de bloques de programa que contienen llamadas de "TM_MAIL" o llamadas de la instancia de "TM_MAIL" En caso de incumplimiento, es posible que los recursos de conexión permanezcan ocupados. El sistema de automatización puede pasar a un estado no definido con las funciones de comunicación TCP/IP a través de Industrial Ethernet. Una vez transmitidas las modificaciones es necesario ejecutar un rearranque completo (en caliente) o un arranque en frío de la CPU.

Coherencia de datos

El parámetro de entrada ADDR_MAIL_SERVER de la instrucción se aplica de nuevo cada vez que se lanza la transmisión de un mensaje de correo electrónico de la instrucción "TM_MAIL". Si se produce un cambio durante el funcionamiento, el valor "nuevo" no tendrá efecto hasta que no se vuelva a lanzar un mensaje de correo electrónico.

Por el contrario, los parámetros WATCH_DOG_TIME, TO_S, CC, FROM, SUBJECT, TEXT, ATTACHMENT, y, dado el caso, USERNAME y PASSWORD son aplicados por éste durante el funcionamiento de la instrucción "TM_MAIL", por lo que no pueden modificarse hasta que la petición no haya finalizado (BUSY = 0)

Parametrización del TS Adapter IE

En el TS Adapter IE hay que introducir los parámetros de las llamadas salientes de forma que el TS Adapter IE establezca una conexión con el servidor de marcado del proveedor de servicios de Internet.

Si para establecer la conexión se ajusta "Bajo demanda", la conexión se establecerá cuando deba enviarse un mensaje de correo electrónico.

Con una conexión analógica vía módem, el establecimiento de la llamada puede requerir cierto tiempo (aprox. un minuto). El tiempo necesario para el establecimiento de la conexión debe tenerse en cuenta al especificar el parámetro WATCH_DOG_TIME.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TM_MAIL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L o constante	Parámetro de control REQUEST: Activa el envío de un mensaje de correo electrónico con un flanco ascendente.
ID	Input	CONN_OUC (Word)	D, L o constante	Referencia a la conexión que debe establecerse. Véase la ID de parámetro de las instrucciones TCON (Página 5298), TDISCON (Página 5305), TSEND (Página 5310) y TRCV (Página 5314). Aquí se debe introducir un número que no se vaya a utilizar en ninguna otra instancia de estas instrucciones en el programa de usuario.
TO_S (Página 5592)	Input	STRING	D	Parámetro de entrada Direcciones de destinatario: STRING con una longitud máxima de 240 caracteres (véase el ejemplo de llamada).
CC (Página 5592)	Input	STRING	D	Parámetro de entrada Direcciones de destinatario CC (opcional): STRING con una longitud máxima de 240 caracteres (véase el ejemplo de llamada). Si se asigna un string vacío, el mensaje no se envía a un destinatario CC.
SUBJECT	Input	STRING	D	Parámetro de entrada Asunto del mensaje: STRING con una longitud máxima de 240 caracteres.
TEXT	Input	STRING	D	Parámetro de entrada Texto del mensaje (opcional): Referencia a un string de datos con una longitud máxima de 240 caracteres. Si a este parámetro se le asigna un string vacío, el mensaje se envía sin texto.
ATTACHMENT	Input	VARIANT	I, Q, M, D, L	Parámetro de entrada Archivos adjuntos del mensaje (opcional): Referencia a un campo de byte/palabra/palabra doble con una longitud máxima de 65534 bytes. Si no se asigna ningún valor, el mensaje se envía sin archivos adjuntos.
DONE	Output	BOOL	I, Q, M, D, L	• DONE = 0: la petición aún no se ha iniciado o todavía se está ejecutando. • DONE = 1: Petición finalizada sin errores.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: el envío del mensaje de correo electrónico todavía no ha finalizado. • BUSY = 0: la ejecución de "TM_MAIL" ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: durante la ejecución se ha producido un error. STATUS y SFC_STATUS devuelven información detallada sobre el tipo de error.
STATUS (Página 5593)	Output	WORD	I, Q, M, D, L	Parámetro de salida/estado STATUS: Valor de retorno o información de error de la instrucción "TM_MAIL".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ADDR_MAIL_SERVER	Static*	DWORD	I, Q, M, D, L	Parámetro de entrada Dirección IP del servidor de correo: Debe indicarse en forma de palabra de datos en formato HEX, p. ej.: Dirección IP = 192.168.0.200. ADDR_MAIL_SERVER = DW#16#C0A800C8, teniendo en cuenta lo siguiente: • 192 = 16#C0, • 168 =16#A8 • 0 = 16#00 y • 200 = 16#C8.
WATCH_DOG_TIME	Static*	TIME	I, Q, M, D, L	Parámetro de entrada Intervalo máx.: En el tiempo predeterminado (WATCH_DOG_TIME), la instrucción "TM_MAIL" debería establecer una conexión. Si se supera este tiempo, el bloque se finaliza con un error. El tiempo hasta que el bloque finaliza y el error se notifica puede superar el WATCH_DOG_TIME, ya que el establecimiento de la conexión también requiere tiempo. Para comenzar debe ajustarse un tiempo de 2 minutos. En la conexión telefónica vía ISDN, puede seleccionarse un tiempo notablemente inferior.
USERNAME	Static*	STRING	D	Parámetro de entrada Nombre de usuario: STRING con una longitud máxima de 180 caracteres. Para los procesos de autenticación es imprescindible un nombre de usuario.
PASSWORD	Static*	STRING	D	Parámetro de entrada Contraseña: STRING con una longitud máxima de 180 caracteres. Para los procesos de autenticación es imprescindible una contraseña.
FROM (Página 5592)	Static*	STRING	D	Parámetro de entrada Dirección de remitente: STRING con una longitud máxima de 240 caracteres (véase el ejemplo de llamada).
SFC_STATUS (Página 5593)	Static*	WORD	I, Q, M, D, L	Parámetro de salida/estado "SFC_STATUS": Información de error de los bloques de comunicación llamados.
* Los valores de los parámetros no se modifican cada vez que se llama la instrucción "TM_MAIL". Los valores se encuentran en los parámetros estáticos de la instancia y sólo se escriben una vez antes de la primera llamada de la instrucción.				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Parámetros opcionales

Los parámetros opcionales CC, TEXT y ATTACHMENT sólo se envían con el mensaje de correo electrónico si los parámetros correspondientes contienen un string de longitud > 0.

Autenticación SMTP

Bajo el término "autenticación" se entiende un proceso con el que se garantiza una identidad, como puede ser una consulta de contraseña.

La instrucción "TM_MAIL" soporta el método de autenticación SMTP AUTH-LOGIN, requerido por la mayoría de servidores de correo. Para más información sobre el método de autenticación de su servidor de correo, consulte el manual del servidor o la página web del proveedor de servicios de Internet.

Para utilizar el método de autenticación AUTH-LOGIN, la instrucción "TM_MAIL" requiere el nombre de usuario con el que poder iniciar sesión en el servidor de correo. Este nombre de usuario equivale al nombre de usuario con el que se ha configurado la cuenta de correo en el servidor de correo. Se da a conocer a través del parámetro USERNAME de la instrucción "TM_MAIL".

Asimismo, para iniciar sesión la instrucción "TM_MAIL" requiere la contraseña correspondiente. Esta contraseña equivale a la contraseña indicada al configurar la cuenta de correo. Se da a conocer a través del parámetro PASSWORD de la instrucción "TM_MAIL".

El nombre de usuario y la contraseña se transfieren sin cifrar (codificación BASE64) al servidor de correo.

Si en el DB no se indica ningún nombre de usuario, no se utiliza el método de autenticación AUTH-LOGIN. El mensaje de correo electrónico se envía entonces sin autenticación.

Parámetros TO_S, CC y FROM

Descripción

Los parámetros TO_S, CC y FROM son strings con, por ejemplo, el siguiente contenido:

- TO: <wenna@mydomain.com>, <ruby@mydomain.com>,
- CC: <admin@mydomain.com>, <judy@mydomain.com>,
- FROM: <admin@mydomain.com>

Las siguientes reglas deben observarse a la hora de introducir los parámetros:

- Deben introducirse los caracteres "TO:", "CC:" y "FROM:".
- Antes de cada dirección debe introducirse un espacio y un corchete en "<".
- Después de cada dirección debe introducirse un corchete en ">".
- Después de cada dirección que se indique en TO y CC debe introducirse una coma.
- En FROM sólo puede introducirse una dirección de correo electrónico, que no puede tener una coma al final

Por motivos relacionados con el tiempo de ejecución y el espacio en memoria, la instrucción "TM_MAIL" no realiza ninguna prueba de sintaxis de los parámetros TO_S, CC y FROM.

Parámetros STATUS y SFB_STATUS

Descripción

Los valores de retorno de la instrucción "TM_MAIL" pueden clasificarse del siguiente modo:

- W#16#0000: "TM_MAIL" ha finalizado correctamente
- W#16#7xxx: estado de "TM_MAIL"
- W#16#8xxx: se ha notificado un error en la llamada interna de un bloque de comunicación o lo ha hecho el servidor de correo electrónico.

La tabla siguiente muestra los valores de retorno de "TM_MAIL" a excepción del código de error de los bloques de comunicación llamados.

Valor de retorno STATUS* (W#16#...):	Valor de retorno SFB_STATUS (W#16#...):	Explicación	Indicaciones
0000	-	La ejecución de "TM_MAIL" ha finalizado sin errores.	La finalización sin errores de "TM_MAIL" no significa que el mensaje de correo electrónico enviado sea recibido (véase más abajo - Punto 1 de la nota)
7001		"TM_MAIL" está activo (BUSY = 1).	Primera llamada; petición lanzada
7002	7002	"TM_MAIL" está activo (BUSY = 1).	Llamada intermedia; petición ya activa
8xxx	xxxx	La ejecución de "TM_MAIL" ha finalizado con un código de error de las instrucciones de comunicación llamadas internamente.	Encontrará información detallada sobre la evaluación del parámetro SFB_STATUS en las descripciones del parámetro STATUS de las instrucciones de comunicación.
8010	xxxx	Error al establecer la conexión.	Encontrará más información sobre la evaluación del parámetro SFB_STATUS en las descripciones del parámetro STATUS de la instrucción "TCON (Página 5298)".
8011	xxxx	Error al enviar los datos.	Encontrará más información sobre la evaluación de SFB_STATUS en las descripciones del parámetro STATUS de la instrucción "TSEND (Página 5310)".
8012	xxxx	Error al recibir los datos.	Encontrará más información sobre la evaluación de SFB_STATUS en las descripciones del parámetro STATUS de la instrucción "TRCV (Página 5314)".
8013	xxxx	Error al establecer la conexión.	Encontrará más información sobre la evaluación de SFB_STATUS en las descripciones del parámetro STATUS de la instrucción "TCON (Página 5298)" y "TDISCON (Página 5305)".

Valor de retorno STATUS* (W#16#...):	Valor de retorno SFB_STATUS (W#16#...):	Explicación	Indicaciones
8014	-	No es posible establecer una conexión.	Posiblemente ha introducido una dirección IP del servidor de correo incorrecta (ADDR_MAIL_SERVER) o un intervalo demasiado reducido (WATCH_DOG_TIME) para establecer la conexión. También existe la posibilidad de que la CPU no tenga conexión con la red o que la configuración de la CPU sea incorrecta.
8016	-	Error al copiar el anexo	-
82xx, 84xx, o bien 85xx	-	El mensaje de error procede del servidor de correo y, exceptuando el "8", corresponde al número de error del protocolo SMTP. En las siguientes columnas se indican algunos códigos de error posibles:	Véase el punto 2 de la nota.
8450	-	Acción no ejecutada: Buzón de correo no disponible/no accesible.	Vuelva a intentarlo más tarde.
8451	-	Acción cancelada: Error local en el procesamiento	Vuelva a intentarlo más tarde.
8500	-	Error de sintaxis: Error no detectado. Ello incluye también el error debido a una cadena de comandos demasiado larga. Otra posible causa es que el servidor de correo electrónico no soporte el método de autenticación LOGIN.	Compruebe los parámetros de "TM_MAIL". Intente enviar un mensaje de correo electrónico sin autenticación. Sustituya para ello el parámetro USERNAME por un string vacío.
8501	-	Error de sintaxis: Parámetro o argumento incorrecto	Posiblemente ha introducido una dirección incorrecta en TO_S o CC.
8502	-	Comando desconocido o no implementado.	Compruebe las entradas realizadas, especialmente el parámetro FROM. Es posible que esté incompleto y que haya olvidado "@" o ".".
8535	-	Autenticación SMTP incompleta.	Puede que haya introducido un nombre de usuario o una contraseña incorrectas.
8550	-	No es posible acceder al servidor de correo, no tiene derechos de acceso.	Posiblemente ha introducido un nombre de usuario o una contraseña incorrectos o el servidor de correo no soporta el LOGIN. Otra posible causa del error puede ser la entrada incorrecta del nombre del dominio después de "@" en TO_S o CC.
8552	-	Acción cancelada: Rebase del espacio de memoria asignado	Vuelva a intentarlo más tarde.
8554	-	Transmisión fallida.	Vuelva a intentarlo más tarde.
* Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".			

Nota

Error de estado

1. La entrada incorrecta de las direcciones de los destinatarios no genera un error de estado de la instrucción "TM_MAIL". En tal caso no es posible garantizar el envío del mensaje de correo electrónico a otros destinatarios, aunque se hayan introducido correctamente.
2. Encontrará más información sobre el código de error SMTP y otros códigos de error en el protocolo SMTP en Internet, o bien en la documentación sobre errores del servidor de correo. También es posible ver el último mensaje de error enviado por el servidor de correo en forma de texto en el DB de instancia en el parámetro BUFFER1. En él encontrará en "Datos" los últimos datos enviados por la instrucción "TM_MAIL".

Consulte también

Cambiar formatos de visualización en el estado del programa (Página 3100)

9.8.2 Referencias (S7-300, S7-400)

9.8.2.1 Parámetros generales de las instrucciones

Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente

Instrucciones que funcionan asíncronamente

En las instrucciones que funcionan asíncronamente la ejecución de la función se extiende a lo largo de varias llamadas.

Identificación de la petición

Si mediante una instrucción que funciona asíncronamente se dispara una alarma de proceso o se lanza la salida de comandos de control a los esclavos DP o bien una transmisión de datos o la cancelación de una conexión no configurada y se vuelve a llamar dicha instrucción antes de que finalice la petición en curso, el comportamiento posterior de la instrucción dependerá en gran medida de si en la siguiente llamada se trata de la misma petición o no.
La tabla siguiente explica para cada instrucción qué parámetros de entrada definen una petición. Si los parámetros indicados ahí coinciden con una petición que no ha finalizado todavía, la llamada se considerará una llamada consecutiva.

Instrucción	La petición se identifica por
"DP_PRAL (Página 6794)"	IOID, LADDR
"DPSYC_FR (Página 6797)"	LADDR, GROUP, MODE

Instrucción	La petición se identifica por
"D_ACT_DP (Página 6764)"	LADDR
"DPNRM_DG (Página 6803)"	LADDR
"RDSYSST (Página 6963)"	SZL_ID, INDEX
"WR_PARM (Página 6882)"	IOID, LADDR, RECNUM
"WR_DPARM (Página 6883)"	IOID, LADDR, RECNUM
"PARM_MOD (Página 6877)"	IOID, LADDR
"WR_REC (Página 6774)"	IOID, LADDR, RECNUM
"RD_REC (Página 6769)"	IOID, LADDR, RECNUM
"X_SEND (Página 8269)"	DEST_ID, REQ_ID
"X_GET (Página 8274)"	DEST_ID, VAR_ADDR
"X_PUT (Página 8276)"	DEST_ID, VAR_ADDR
"X_ABORT (Página 8278)"	DEST_ID
"I_GET (Página 8258)"	IOID, LADDR, VAR_ADDR
"I_PUT (Página 8260)"	IOID, LADDR, VAR_ADDR
"I_ABORT (Página 8262)"	IOID, LADDR
"CREA_DBL (Página 7041)"	LOW_LIMIT, UP_LIMIT, COUNT, ATTRIB, SRCBLK
"READ_DBL (Página 7032)"	SRCBLK, DSTBLK
"WRIT_DBL (Página 7034)"	SRCBLK, DSTBLK
"RD_DPARA (Página 6876)"	LADDR, RECNUM
"DPSYC_FR (Página 6806)"	DP_ID
"PN_DP (Página 8268)"	-

Parámetro REQ

El parámetro de entrada REQ (request) sirve exclusivamente para lanzar la petición:

- Si se llama la instrucción de una petición que no está activada en este momento, la petición se lanza con REQ = 1 (caso 1).
- Si hay una petición determinada en curso y se vuelve a llamar la instrucción para la misma petición antes de que finalice (p. ej. en un OB de alarma cíclica), REQ no es evaluado por la instrucción (caso 2).

Parámetro RET_VAL y BUSY

A través de los parámetros de salida RET_VAL y BUSY se indica el estado de ejecución de la petición.

Tenga en cuenta la indicación contenida en la sección: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

- En el caso 1 (primera llamada con REQ=1) siempre que haya recursos de sistema libres y los parámetros de entrada reciban valores correctos se registrará en RET_VALW#16#7001 y se activará BUSY.
  Si los recursos de sistema necesarios están ocupados en estos momentos o si hay un error en los parámetros de entrada, se registrará el código de error correspondiente en RET_VAL y se escribe un 0 en BUSY.
- En el caso 2 (llamada intermedia) se registra W#16#7002 en RET_VAL (esto equivale a una advertencia: petición todavía en curso), y se activará BUSY.
- En la última llamada de una petición vale lo siguiente:
  - En el caso de las instrucciones "DPNRM_DG (Página 6803)", "X_GET (Página 8274)" y "I_GET (Página 8258)" si la transferencia de datos es correcta se registra en RET_VAL el número de datos transferidos en bytes como número positivo. En BUSY se escribe un "0" en este caso.
    En caso de error se registra información de error en RET_VAL. BUSY no se debe evaluar en este caso.
  - En el caso de "RD_REC" si la transferencia de datos es correcta, en RET_VAL se registra el tamaño del registro en bytes o 0 (véase también: RD_REC: Leer registro de periferia (Página 6769)). En BUSY se escribe un "0". En caso de error se registra el código de error en RET_VAL y en BUSY se escribe un "0".
  - En las demás instrucciones y si la petición se ejecuta sin errores, se registra un "0" en RET_VAL y en BUSY se escribe un "0". En caso de error se registra el código de error en RET_VAL y en BUSY se escribe un "0".

Nota

Si la primera y la última llamada coinciden en el tiempo, para RET_VAL y BUSY rige lo descrito para la última llamada.

Resumen

La tabla siguiente muestra una visión esquemática de lo arriba descrito. En particular, indica los valores posibles de los parámetros de salida, si la petición no ha finalizado todavía después de una llamada de la instrucción.

Nota

En el programa es necesario evaluar los parámetros de salida relevantes depués de cada llamada.

Relación entre llamada, REQ, RET_VAL y BUSY cuando hay una petición en curso

N.º de la llamada	Tipo de llamada	REQ	RET_VAL	BUSY
1	Primera llamada	1	W#16#7001	1
			Código de error	0
2 a (n - 1)	Llamada intermedia	Irrelevante	W#16#7002	1
n	Última llamada	Irrelevante	W#16#0000 (excepciones: "RD_REC (Página 6769)", si la zona de destino es mayor que el registro transferido, "DPNRM_DG (Página 6803)", "X_GET (Página 8274)" y "I_GET (Página 8258)"), si no se han producido errores.	0
			Código de error, si se han producido errores.	0

Evaluación de errores con el parámetro de salida RET_VAL

Tipos de información de error

Una instrucción procesada le indica en el programa de usaurio si la CPU ha podido o no ejecutar la función de la instrucción.

Recibirá información de error al respecto por dos vías:

- vía el bit BR de la palabra de estado
- vía el parámetro de salida RET_VAL (return value).

Nota

Antes de evaluar los parámetros de salida de la instrucción siempre deberá procederse del siguiente modo:

- Evalúe en primer lugar el bit BR de la palabra de estado.
- Compruebe después el parámetro de salida RET_VAL.

Si el bit BR indica que la instrucción no ha sido procesada correctamente o si en RET_VAL figura un código de error general, no deberán evaluarse los parámetros de salida de la instrucción.

Información de error en el valor de retorno

Una instrucción indica con el valor "0" del bit de resultado binario (BR) de la palabra de estado que se ha producido un error al procesar la función. Algunas instrucciones ponen a disposición un código de error adicional en un parámetro de salida llamado valor de retorno (RET_VAL). Si en el parámetro de salida RET_VAL aparece un error general (v. explicación más abajo), ello se indica con el valor "0" del bit BR de la palabra de estado.

El valor de retorno es del tipo de datos entero (INT). La relación del valor de retorno con respecto al valor "0" indica si se ha producido un error al procesar la función.

Procesamiento de la instrucción por la CPU	BR	Valor de retorno	Signo del entero
erróneo	0	menor que "0"	negativo (bit de signo es "1")
correcto	1	mayor o igual a "0"	positivo (bit de signo es "0")

Reacción a la información de error

En los códigos de error de RET_VAL se distingue entre:

- un código de error general que pueden devolver todas las instrucciones
- un código de error específico que puede devolver una instrucción según sus funciones específicas.

El programa se puede escribir de manera que reaccione a posibles errores en el procesamiento de la instrucción. De este modo pueden evitarse errores subsiguientes.

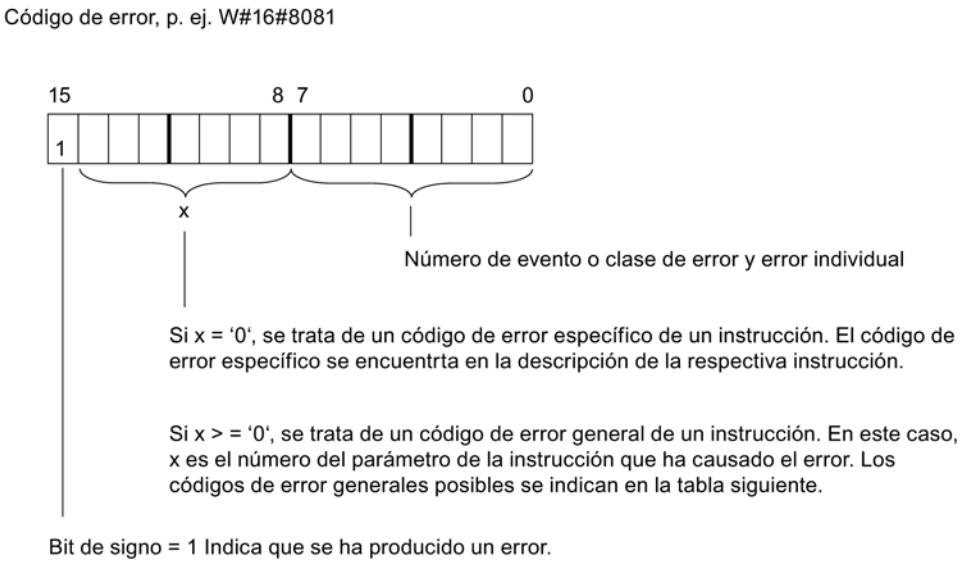
Información de error general y específica

El valor de retorno (RET_VAL) de una instrucción pone a disposición uno de los siguientes dos códigos de error:

- Un código de error general que puede referirse a cualquier instrucción.
- Un código de error específico que sólo puede referirse a la instrucción en cuestión.

Si bien el tipo de datos del parámetro de salida RET_VAL es un entero (INT), los códigos de error de la instrucción tienen valores hexadecimales. Si evalúa un valor de retorno y lo compara con los códigos de error indicados en esta documentación, visualice el código de error en formato hexadecimal.

La figura siguiente ilustra la estructura de un código de error de una función de sistema en formato hexadecimal.

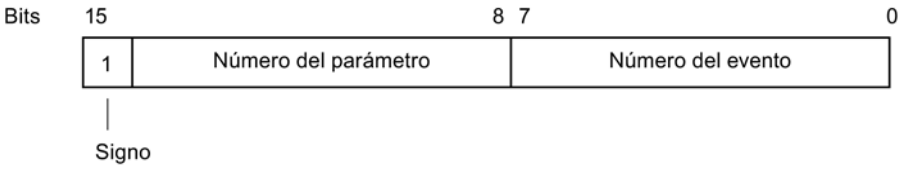


Información de error general

El código de error general indica errores que pueden aparecer en todas las instrucciones. Un código de error general está formado por los siguientes dos números:

- Un número de parámetro comprendido entre 1 y 111, donde 1 muestra el primer parámetro, 2 el segundo parámetro etc. de la instrucción llamada.
- Un número de evento comprendido entre 0 y 127. El número de evento indica un error síncrono.

A continuación se indican los códigos de errores generales así como su significado.



Nota

En caso de que se haya registrado un código de error general en RET_VAL, es posible que

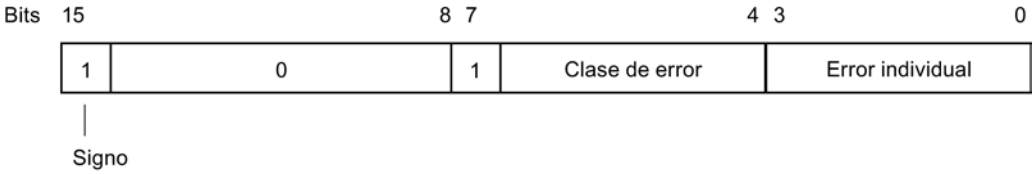
- se haya lanzado la acción correspondiente a la instrucción o que ya haya sido ejecutada por completo.
- se haya producido un error específico de la instrucción al ejecutar la acción. Debido a un error general que haya tenido lugar posteriormente, ya no ha sido posible indicar el error específico.

Información de error específica

Algunas instrucciones poseen un valor de retorno que pone a disposición un código de error específico de la instrucción. El código de error específico indica errores que sólo pueden aparecer en una única instrucción.

Un código de error específico está formado por los siguientes dos números:

- Una clase de error comprendida entre 0 y 7.
- Un error individual comprendido entre 0 y 15.



Códigos de error generales

La tabla siguiente explica los códigos de error generales de un valor de retorno. El código de error se indica en formato hexadecimal. La letra x en cada número de error sirve únicamente de comodín y representa el número del parámetro de la función de sistema que ha causado el error.

Códigos de error generales

Código de error (W#16#....)	Explicación
8x7F	Error interno Este código de error indica un error interno en el parámetro x.
8x01	Identificador de sintaxis no válido en un parámetro ANY
8x22	Error de longitud de área al leer un parámetro.
8x23	Error de longitud de área al escribir un parámetro. Este código de error indica que el parámetro x se encuentra íntegra o parcialmente fuera del área de operandos o que la longitud de un campo de bits de un parámetro ANY no es divisible por 8.
8x24	Error de área al leer un parámetro.
8x25	Error de área al escribir un parámetro. Este código de error indica que el parámetro x se encuentra en un área no válida para la función de sistema. La descripción de la función en cuestión indica las áreas que no son válidas para la función.
8x26	El parámetro contiene un número demasiado elevado de una célula de temporizador. Este código de error indica que la célula de temporizador indicada en el parámetro x no existe.
8x27	El parámetro contiene un número demasiado elevado de una célula de contador (error de número del contador). Este código de error indica que la célula de contador indicada en el parámetro x no existe.
8x28	Error de alineación al leer un parámetro.
8x29	Error de alineación al escribir un parámetro. Este código de error indica que la referencia al parámtero x es un operando cuya dirección de bit es diferente de 0.
8x30	El parámetro se encuentra en el DB global protegido contra escritura.
8x31	El parámetro se encuentra en el DB de instancia protegido contra escritura. Este código de error indica que el parámetro x se encuentra en un bloque de datos protegido contra escritura. Si el bloque de datos ha sido abierto por la propia función de sistema, la función de sistema devolverá siempre el valor W#16#8x30.
8x32	El parámetro contiene un número de DB demasiado elevado (error de número del DB).
8x34	El parámetro contiene un número de FC demasiado elevado (error de número del FC).
8x35	El parámetro contiene un número de FB demasiado elevado (error de número del FB). Este código de error indica que el parámetro x contiene un número de bloque mayor que el número de bloque máximo admisible.
8x3A	El parámetro contiene el número de un DB que no está cargado.
8x3C	El parámetro contiene el número de un FC que no está cargado.
8x3E	El parámetro contiene el número de un FB que no está cargado.
8x42	Se ha producido un error de accdeso mientras el sistema intentaba leer un parámetro del área de periferia de las entradas.

Código de error (W#16#....)	Explicación
8x43	Se ha producido un error de accdeso mientras el sistema intentaba escribir un parámetro en el área de periferia de las salidas.
8x44	Error en el enésimo (n > 1) acceso en lectura tras producirse un error.
8x45	Error en el enésimo (n > 1) acceso en escritura tras producirse un error. Este código de error indica que se deniega el acceso al parámetro deseado.

9.8.2.2 Instrucciones básicas

KOP

Operaciones lógicas con bits

---| |---: Contacto normalmente abierto

Descripción

La activación de un contacto normalmente abierto depende del estado lógico del operando correspondiente. Si el estado lógico del operando es "1", se cierra el contacto normalmente abierto y el estado lógico de la entrada se transfiere a la salida.

Si el estado lógico del operando es "0", el contacto normalmente abierto no se activa y el estado lógico de la salida de la instrucción se pone a "0".

En una conexión en serie, dos o más contactos normalmente abiertos se combinan de bit en bit mediante una operación lógica Y. La corriente fluye por una conexión en serie cuando todos los contactos están cerrados.

En una conexión en paralelo, los contactos normalmente abiertos se combinan mediante una operación lógica O. La corriente fluye por una conexión en paralelo si uno de los contactos está cerrado.

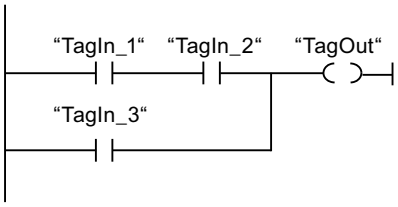
Parámetros

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" tienen el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de control de una cinta transportadora (Página 3241)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3244)

Ejemplo de control de la temperatura ambiente (Página 3248)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3243)

---| / |---: Contacto normalmente cerrado

Descripción

La activación de un contacto normalmente cerrado depende del estado lógico del operando correspondiente. Si el estado lógico del operando es "1", se abre el contacto normalmente cerrado y el estado lógico de la salida de la instrucción se pone a "0".

Si el estado lógico del operando es "0", no se activa el contacto normalmente cerrado y el estado lógico de la entrada se transfiere a la salida.

En una conexión en serie, dos o más contactos normalmente cerrados se combinan de bit en bit mediante una operación lógica Y. La corriente fluye por una conexión en serie cuando todos los contactos están cerrados.

En una conexión en paralelo, los contactos normalmente cerrados se combinan mediante una operación lógica O. La corriente fluye por una conexión en paralelo si uno de los contactos está cerrado.

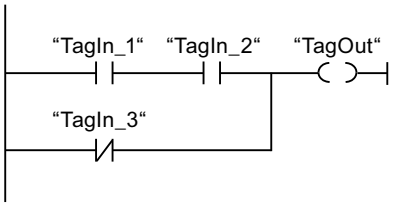
Parámetros

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" tienen el estado lógico "1".
- El operando "TagIn_3" tiene el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3243)

Ejemplo de control de la temperatura ambiente (Página 3248)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

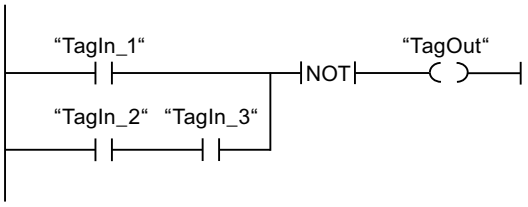
--|NOT|--: Invertir RLO

Descripción

La instrucción "Invertir RLO" invierte el estado lógico del resultado lógico (RLO). Si el estado lógico de la entrada de la instrucción es "1", la salida de la instrucción devuelve el estado lógico "0". Si el estado lógico de la entrada de la instrucción es "0", la salida devuelve el estado lógico "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El estado lógico de los operandos "TagIn_2" y "TagIn_3" es "1".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

---( )---: Asignación

Descripción

La instrucción "Asignación" permite activar el bit de un operando indicado. Si el resultado lógico (RLO) en la entrada de la bobina es "1", el operando indicado adopta el estado lógico "1". Si el estado lógico de la entrada de la bobina es "0", el bit del operando indicado se pone a "0".

La instrucción no afecta al RLO. El RLO de la entrada de la bobina se transfiere directamente a la salida.

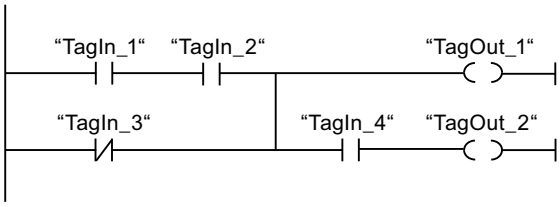
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut_1" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

El operando "TagOut_2" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1", "TagIn_2" y "TagIn_4" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0" y el del operando "TagIn_4" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3244)

Ejemplo de control de la temperatura ambiente (Página 3248)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

---(MCR<): Abrir áreas MCR (Página 5765)

---(R)---: Desactivar salida

Descripción

La instrucción "Desactivar salida" permite poner a "0" el estado lógico de un operando indicado.

La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si fluye corriente hacia la bobina (RLO = "1"), el operando indicado se pone a "0". Si el RLO de la entrada de la bobina es "0" (no hay flujo de señales en la bobina), el estado lógico del operando indicado no cambia.

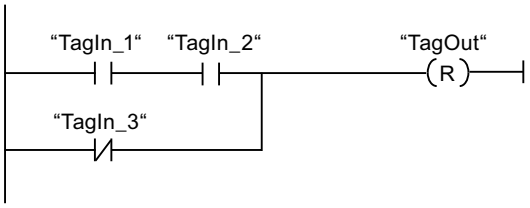
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L, T, C	Operando que se desactiva si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Insertar elementos KOP (Página 2678)
Editar elementos KOP (Página 2694)
Cablear operandos en instrucciones KOP (Página 2701)
Ejemplo de control de una cinta transportadora (Página 3241)
Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3243)
El MCR (Master Control Relay) (Página 2564)
Indicaciones importantes sobre el uso de la función MCR (Página 2565)
Activar el bit de estado en operaciones lógicas con bits (Página 2547)
Principios básicos de la palabra de estado (Página 2545)
---(MCR<): Abrir áreas MCR (Página 5765)

---(S)---: Activar salida

Descripción

Mediante la instrucción "Activar salida" se pone a "1" el estado lógico de un operando indicado.

La instrucción se ejecuta solo si el resultado lógico (RLO) de la entrada de la bobina es "1". Si fluye corriente hacia la bobina (RLO = "1"), el operando indicado se pone a "1". Si el RLO de la entrada de la bobina es "0" (no hay flujo de señales en la bobina), el estado lógico del operando indicado no cambia.

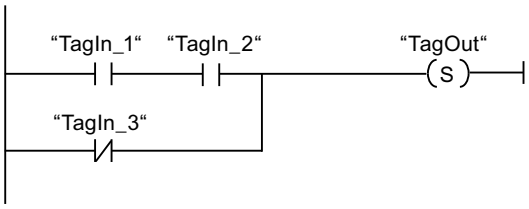
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se activa si RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de control de una cinta transportadora (Página 3241)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3243)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

---(MCR<): Abrir áreas MCR (Página 5765)

SR: Flipflop de activación/desactivación

Descripción

La instrucción "Flipflop de activación/desactivación" sirve para activar o desactivar el bit de un operando indicado en función del estado lógico de las entradas S y R1. Si el estado lógico de la entrada S es "1" y el de la entrada R1 es "0", el operando indicado se pone a "1". Si el estado lógico de la entrada S es "0" y el de la entrada R1 es "1", el operando indicado se pone a "0".

La entrada R1 prevalece sobre la entrada S. Si el estado lógico de las entradas S y R1 es "1", el estado lógico del operando indicado se pone a "0".

Si el estado lógico de ambas entradas S y R1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

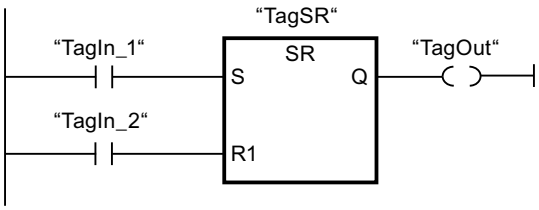
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de activación/desactivación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S	Input	BOOL	I, Q, M, D, L	Activación habilitada
R1	Input	BOOL	I, Q, M, D, L, T, C	Desactivación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	Operando que se activa o desactiva.
Q	Output	BOOL	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagSR" y "TagOut" se activan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagSR" y "TagOut" se desactivan cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

---(MCR<): Abrir áreas MCR (Página 5765)

RS: Flipflop de desactivación/activación

Descripción

La instrucción "Flipflop de desactivación/activación" sirve para desactivar o activar el bit de un operando indicado en función del estado lógico de las entradas R y S1. Si el estado lógico de la entrada R es "1" y el de la entrada S1 es "0", el operando indicado se pone a "0". Si el estado lógico de la entrada R es "0" y el de la entrada S1 es "1", el operando indicado se pone a "1".

La entrada S1 prevalece sobre la entrada R. Si el estado lógico de las entradas R y S1 es "1", el estado lógico del operando indicado se pone a "1".

Si el estado lógico de ambas entradas R y S1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

Parámetros

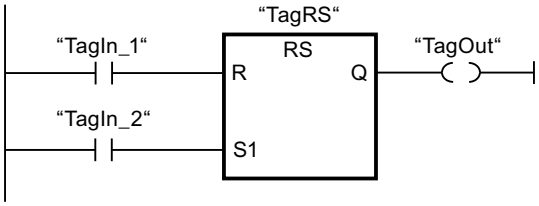
La tabla siguiente muestra los parámetros de la instrucción "Flipflop de desactivación/activación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
R	Input	BOOL	I, Q, M, D, L	Desactivación habilitada
S1	Input	BOOL	I, Q, M, D, L, T, C	Activación habilitada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	InOut	BOOL	I, Q, M, D, L	Operando que se desactiva o activa.
Q	Output	BOOL	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagRS" y "TagOut" se desactivan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagRS" y "TagOut" se activan cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

---(MCR<): Abrir áreas MCR (Página 5765)

--|P|--: Consultar flanco de señal ascendente de un operando

Descripción

La instrucción "Consultar flanco de señal ascendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "0" a "1". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del resultado lógico (RLO) de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

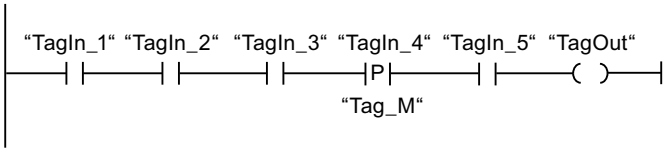
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	BOOL	I, Q, M, D, L, T, C	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1", "TagIn_2" y "TagIn_3" devuelven el estado lógico "1".
- En el operando "TagIn_4" hay un flanco ascendente. El estado lógico de la consulta anterior se guarda en la marca de flancos "Tag_M".
- El estado lógico del operando "TagIn_5" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3243)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

--|N|--: Consultar flanco de señal descendente de un operando

Descripción

La instrucción "Consultar flanco de señal descendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "1" a "0". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos <Operando2>. Si la instrucción detecta un cambio del resultado lógico (RLO) de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente situado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

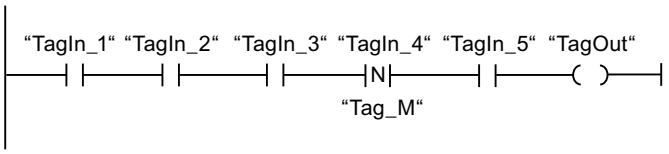
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	BOOL	I, Q, M, D, L, T, C	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1", "TagIn_2" y "TagIn_3" devuelven el estado lógico "1".
- En el operando "TagIn_4" hay un flanco de señal descendente. El estado lógico de la consulta anterior se guarda en la marca de flancos "Tag_M".
- El estado lógico del operando "TagIn_5" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

P_TRIG: Consultar flanco de señal ascendente del RLO

Descripción

La instrucción "Consultar flanco de señal ascendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "0" a "1". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una

marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

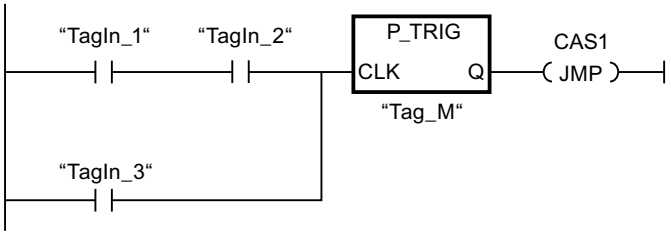
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "0" a "1", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

N_TRIG: Consultar flanco de señal descendente del RLO

Descripción

La instrucción "Consultar flanco de señal descendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "1" a "0". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

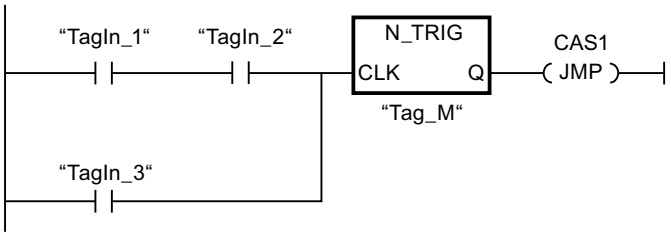
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "1" a "0", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Temporizadores

Temporizadores CEI

TP: Impulso

Descripción

La instrucción "Impulso" permite activar la salida Q por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. La salida Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico de la salida Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo PT y si el estado lógico de la entrada IN es "0", se desactiva la salida ET.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TP (p. ej., "TP_DB")
- Declaración como variable local del tipo TP en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Impulso" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Impulso", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Impulso" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

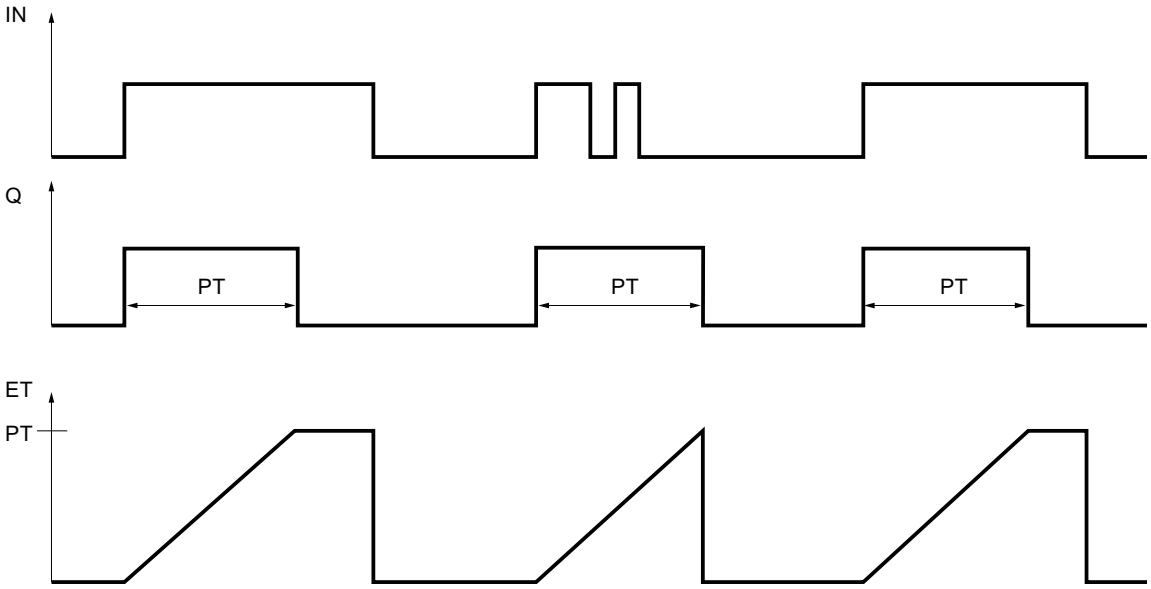
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Duración del impulso El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida de impulso
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de control de la temperatura ambiente (Página 3248)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TON: Retardo al conectar

Descripción

Con la instrucción "Retardo al conectar" se puede retardar la activación de la salida Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. Una vez transcurrido el tiempo PT, la salida Q devuelve el estado lógico "1". La salida Q permanece activada mientras la entrada de arranque esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva la salida Q. La función de temporización se reinicia al detectarse un flanco de señal ascendente nuevo en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. La salida ET se desactiva en cuanto el estado lógico de la entrada IN cambia a "0".

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TON (p. ej., "TON_DB")
- Declaración como variable local del tipo TON en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al conectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al conectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Retardo al conectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

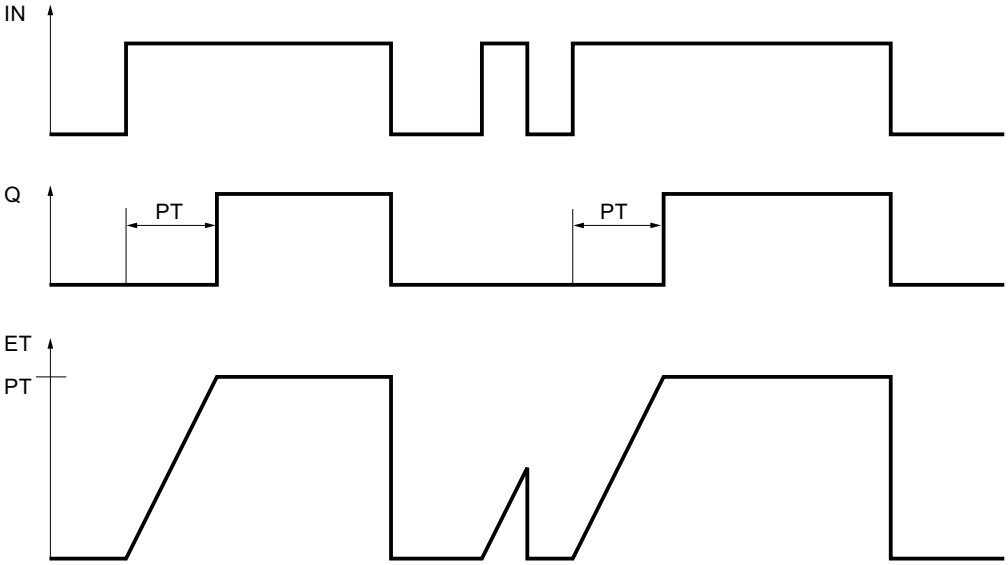
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TOF: Retardo al desconectar

Descripción

Con la instrucción "Retardo al desconectar" se puede retardar la desactivación de la salida Q por el tiempo programado PT. La salida Q se activa cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico de la

entrada IN cambia nuevamente a "0", el tiempo programado PT deja de contar. La salida Q permanece activada mientras transcurre el tiempo PT. Una vez transcurrido el tiempo PT se desactiva la salida Q. Si el estado lógico de la entrada IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico de la salida Q permanece en "1".

El valor de tiempo actual se puede consultar en la salida ET. El valor de tiempo actual empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el valor actual de la salida ET se conserva hasta que la entrada IN cambie nuevamente a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TOF (p. ej., "TOF_DB")
- Declaración como variable local del tipo TOF en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al desconectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al desconectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Retardo al desconectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

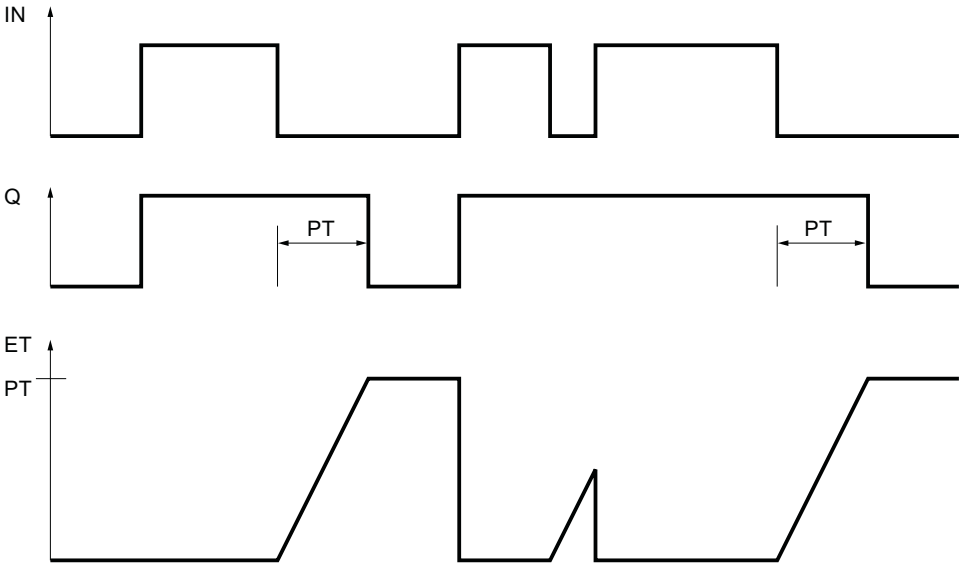
La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Temporizadores SIMATIC

S_PULSE: Parametrizar y arrancar temporizador como impulso

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Si el estado lógico en la entrada S cambia a "0" antes de que transcurra el tiempo programado, se detiene el temporizador. En tal caso, el estado lógico de la salida Q es "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

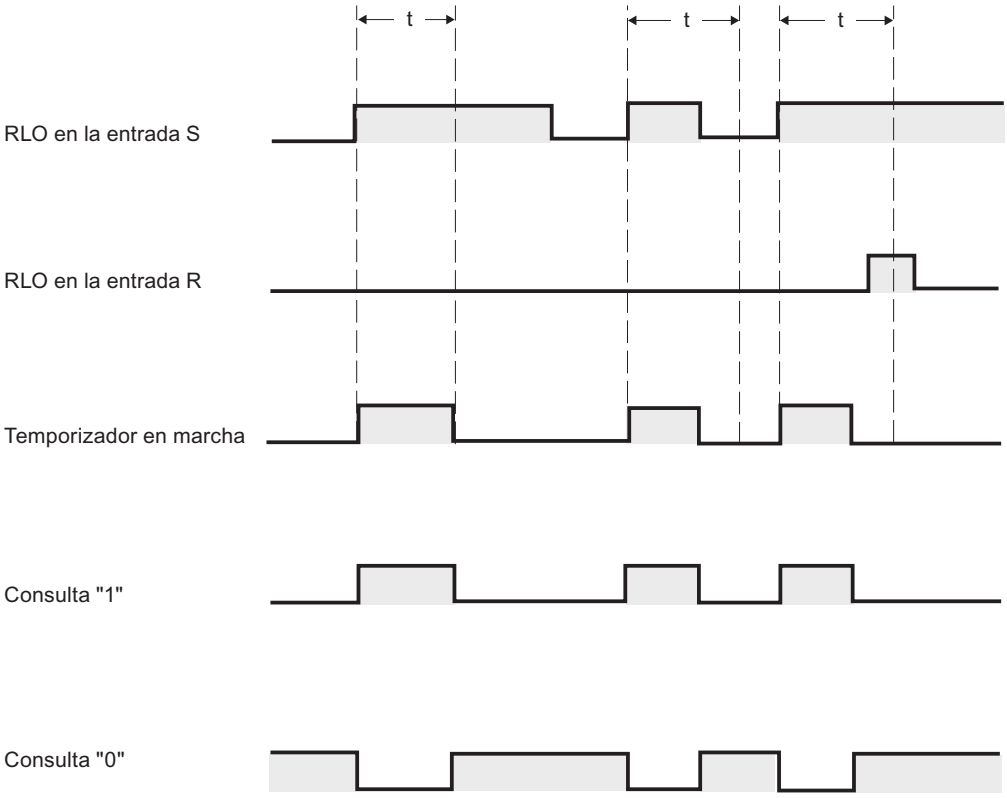
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

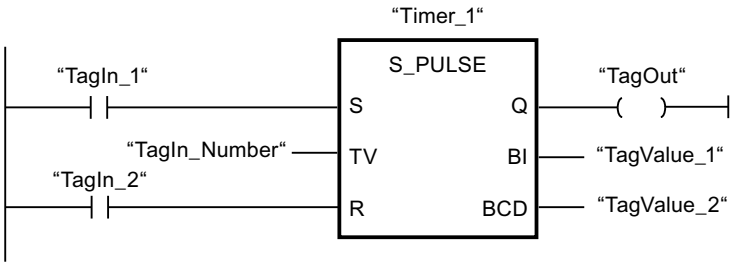
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador "Timer_1" se detiene. En ese caso, el operando "TagOut" se pone a "0".

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha y el operando "TagIn_1" tenga el estado lógico "1". Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso prolongado" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Mientras se ejecuta el temporizador, la salida Q devuelve el estado lógico "1". Una vez transcurrido el tiempo, la salida Q se pone a "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, este se reinicia con el tiempo programado en la entrada TV.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

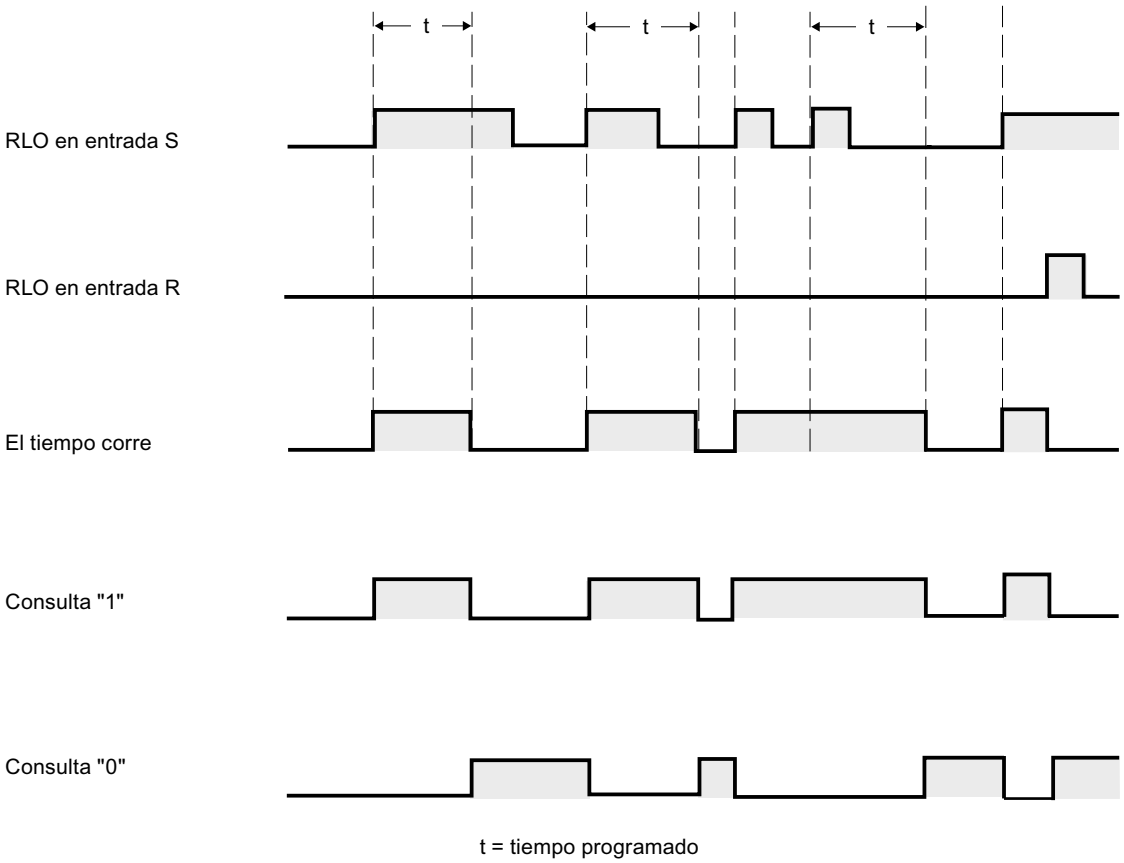
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

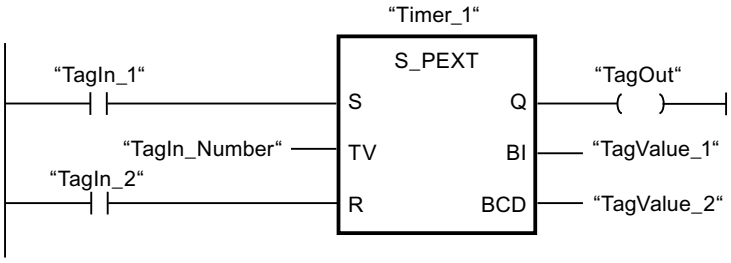
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en la entrada S. Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha. Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" se arranca un temporizador programado como retardo a la conexión al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Una vez transcurrido el tiempo debidamente y si la entrada S aún tiene el valor lógico "1", la salida Q devuelve el valor lógico "1". Si el estado de la entrada S cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En este caso, la salida Q adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia de "0" a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. En ese caso, el estado lógico de la salida Q es "0". El temporizador se reinicia cuando el estado lógico de la entrada R es "1", incluso si el temporizador no se ejecuta y el RLO en la entrada S es "1".

El temporizador de la instrucción se indica en el comodín situado sobre el cuadro. El temporizador debe estar declarado con el tipo de datos TIMER.

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

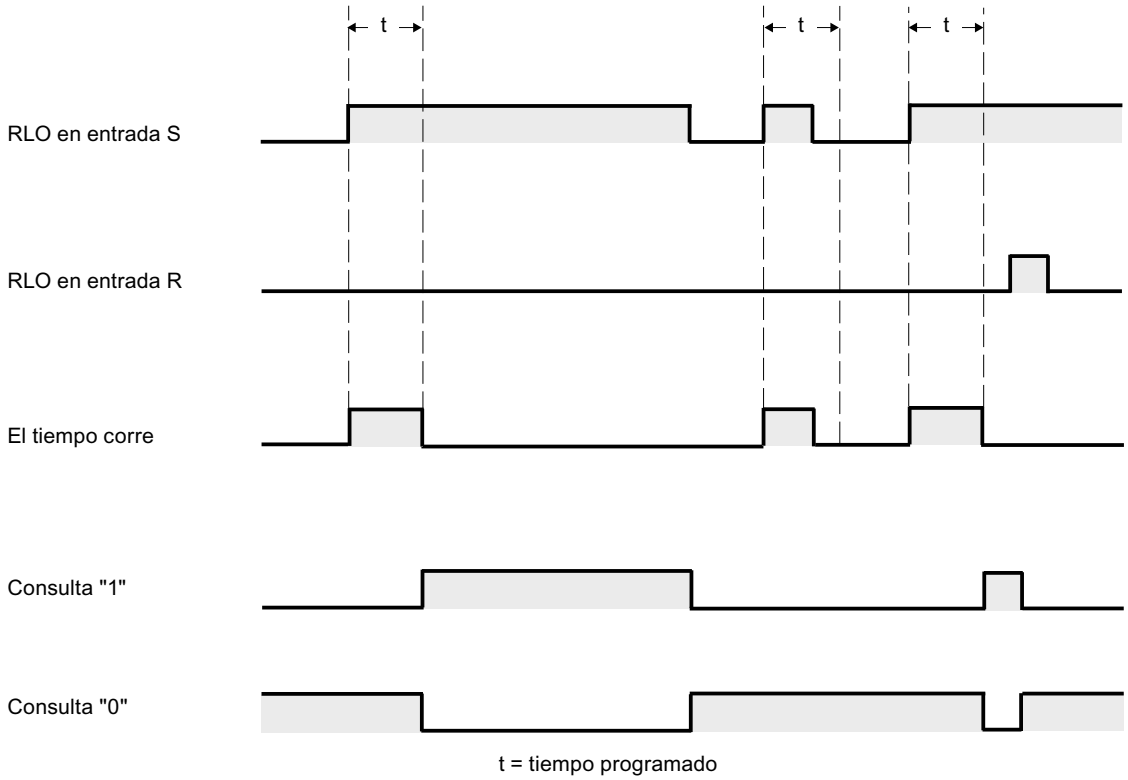
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

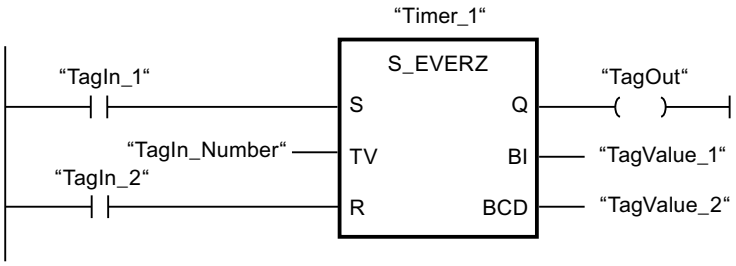
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el operando tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. El operando "TagOut" devuelve en ese caso el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Una vez transcurrido el tiempo, la salida Q devuelve el estado lógico "1" independientemente del estado lógico de la entrada S. Si el estado lógico de la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, el temporizador vuelve a arrancar con el tiempo programado (TV).

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" de la entrada R pone a "0" el valor actual de tiempo y la base de tiempo, independientemente del estado lógico de la entrada de arranque S. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

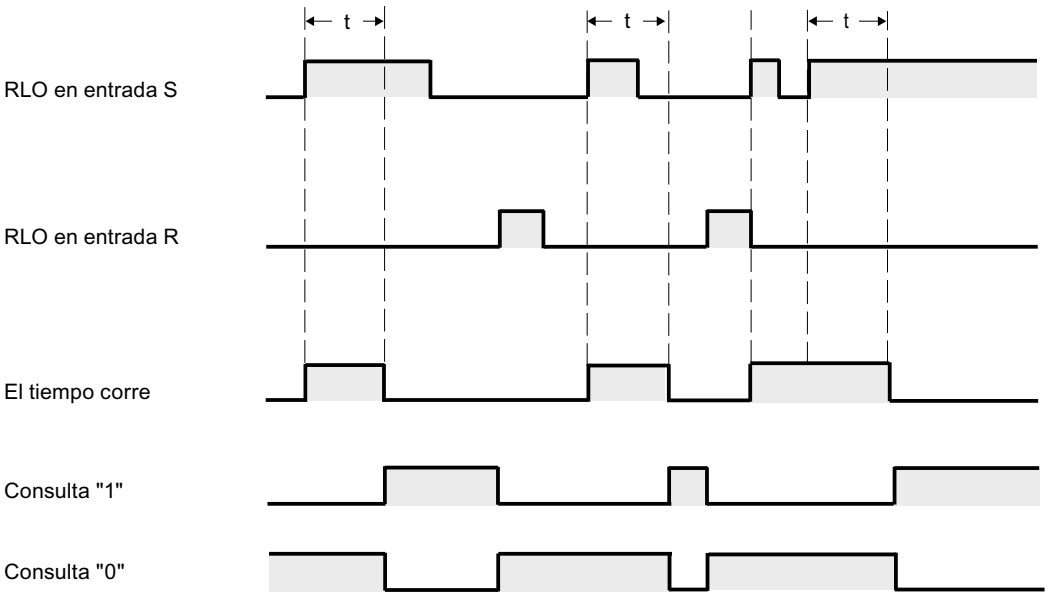
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

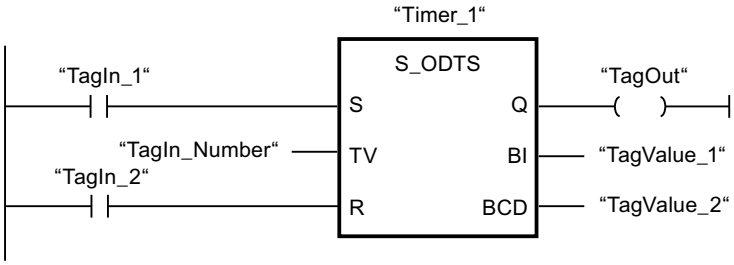
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number" aunque el estado lógico del operando "TagIn_1" cambie a "0". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente) en la entrada S. El temporizador empieza a contar con el tiempo programado (TV). Mientras el temporizador está en marcha o la entrada S devuelve el estado lógico "1", la salida Q tiene el estado lógico "1". Una vez transcurrido el tiempo y si el estado lógico en la entrada S es "0", el estado lógico de la salida Q se pone a "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras el temporizador está funcionando, este se detiene. El temporizador no vuelve a arrancar hasta que no se detecta un flanco de señal descendente en la entrada S.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" en la entrada R pone a "0" el valor de tiempo y la base de tiempo actuales. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

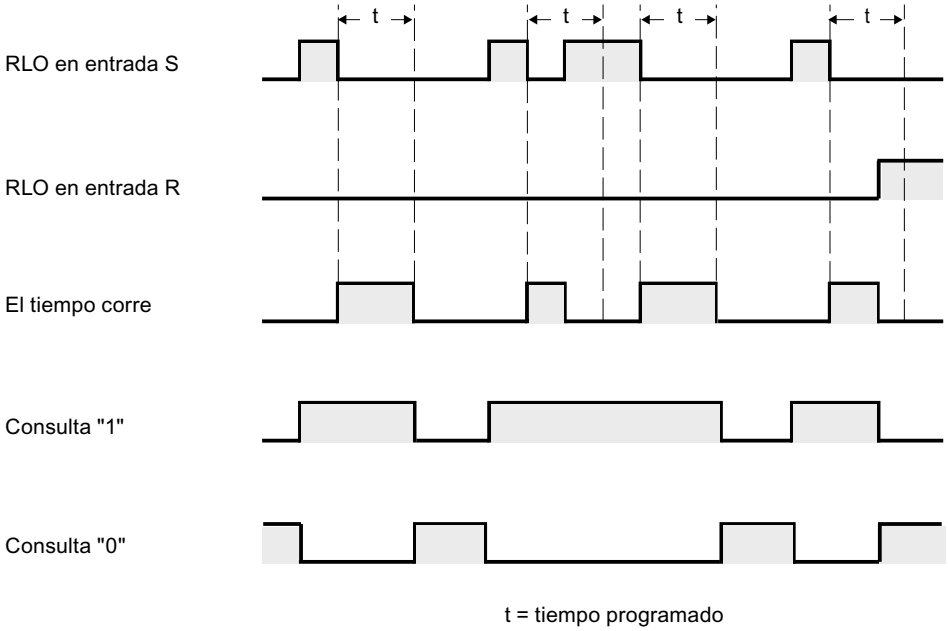
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, D, L	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Valor de tiempo predeterminado
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

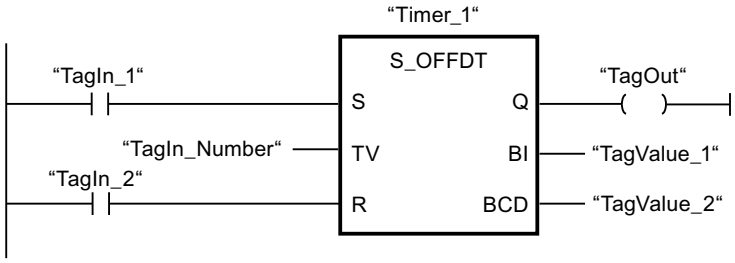
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador está en marcha o el operando "TagIn_1" tiene el estado lógico "0". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

---(SP): Arrancar temporizador como impulso

Descripción

Con la instrucción "Arrancar temporizador como impulso" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador cuenta el tiempo indicado mientras el RLO devuelva el estado lógico "1". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" antes de que haya transcurrido el valor de tiempo, el temporizador se detiene. En ese caso, una consulta de si el estado del temporizador es "1" da como resultado "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

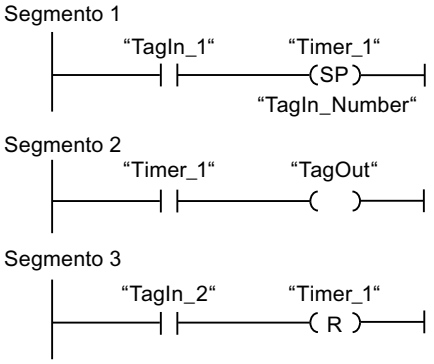
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

---(SE): Arrancar temporizador como impulso prolongado

Descripción

Con la instrucción "Arrancar temporizador como impulso prolongado" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras el temporizador está funcionando, este se arranca de nuevo con el tiempo programado. La consulta sobre si el estado es "1" da como resultado "0" si el tiempo ha transcurrido.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

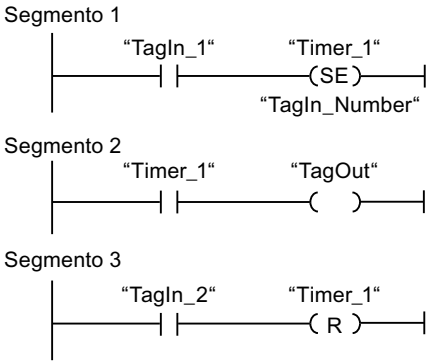
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en el RLO. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

---(SD): Arrancar temporizador como retardo a la conexión

Descripción

Con la instrucción "Arrancar temporizador como retardo a la conexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador empieza a contar con el tiempo indicado y sigue contando mientras el RLO sea "1". Una vez transcurrido el tiempo y si el RLO todavía tiene el estado lógico "1", una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En ese caso, la consulta de si el estado del temporizador es "1" devuelve el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Parámetros

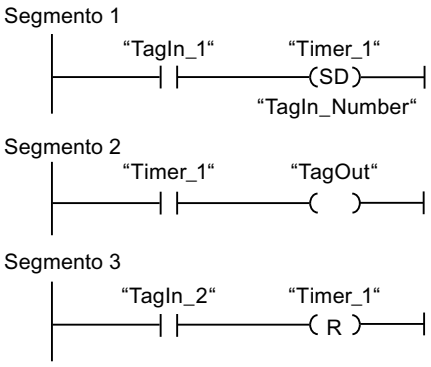
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number".

Una vez transcurrido el tiempo y si el RLO tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

---(SS): Arrancar temporizador como retardo a la conexión con memoria

Descripción

Con la instrucción "Arrancar temporizador como retardo a la conexión con memoria" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente). El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Si ha transcurrido el tiempo, una consulta de si el estado del temporizador es "1" da como resultado "1". Una vez transcurrido el tiempo, el temporizador no puede volver a ponerse en marcha hasta que no se haya reiniciado explícitamente.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

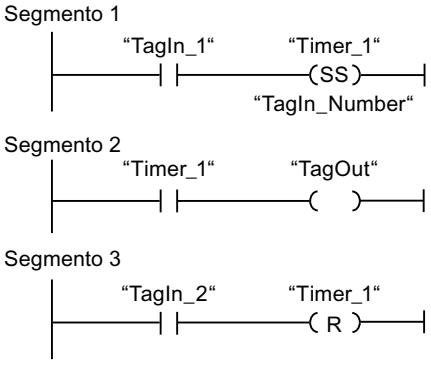
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

---(SF): Arrancar temporizador como retardo a la desconexión

Descripción

Con la instrucción "Arrancar temporizador como retardo a la desconexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente). El temporizador empieza a contar con el tiempo indicado. Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras se ejecuta el temporizador, este se inicializa. El temporizador siempre arranca de nuevo cuando el RLO cambia de "1" a "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

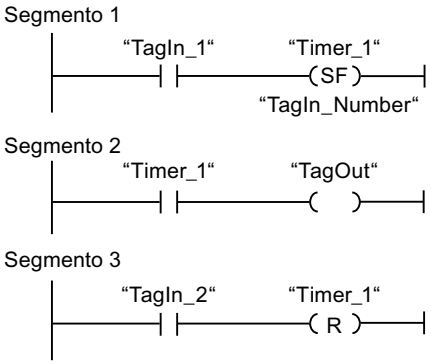
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Tiempo>	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo que cuenta el temporizador.
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Mientras el temporizador está en marcha, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

L: Cargar valor del temporizador (Página 6189)

Contadores

Contadores CEI

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor actual de contaje de la salida CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos (INT) indicado en la salida CV. Cuando se alcanza el valor límite superior, el estado lógico de la entrada CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. El estado lógico de la salida Q es determinado por el parámetro PV. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0". En el parámetro PV también se puede indicar una constante.

El valor de la salida CV se pone a cero cuando el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de la entrada CU no tendrá efecto alguno en la instrucción.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTU (p. ej., "CTU_DB")
- Declaración como variable local del tipo CTU en la sección "Static" de un bloque (p. ej., #MyCTU_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro R de la instrucción. Si un bloque diferente contiene instancias de la instrucción "Contador ascendente", estas pueden reiniciarse p. ej. inicializando el bloque de nivel superior.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Para poder ejecutar la instrucción "Contador ascendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

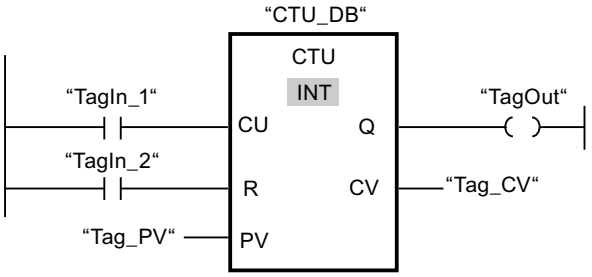
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor actual de contaje del operando "Tag_CV" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos (INT = 32767).

El valor del parámetro PV se utiliza como límite para determinar la salida "TagOut". La salida "TagOut" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor del operando "Tag_PV". En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje

se decrementa hasta alcanzar el valor límite inferior del tipo de datos (INT) indicado. Cuando se alcanza el valor límite inferior, el estado lógico de la entrada CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. Si el valor actual de contaje es menor o igual a cero, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0". En el parámetro PV también se puede indicar una constante.

El valor de la salida CV se pone al valor del parámetro PV cuando el estado lógico de la entrada LD cambia a "1". Mientras la entrada LD tenga el estado lógico "1", el estado lógico de la entrada CD no tendrá efecto alguno en la instrucción.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTD (p. ej., "CTD_DB")
- Declaración como variable local del tipo CTD en la sección "Static" de un bloque (p. ej., #MyCTD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro LD de la instrucción. En el parámetro PV se indica en este caso el valor inicial deseado para el parámetro CV.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Para poder ejecutar la instrucción "Contador descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

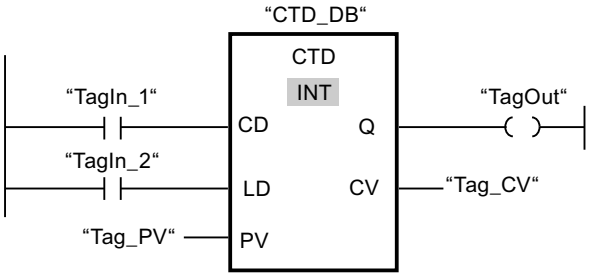
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador descendente" y el valor de la salida "Tag_CV" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos (INT = -32768).

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno y se deposita en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje de la salida CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor actual de contaje no se modifica en la salida CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos (INT) indicado en la salida CV. Una vez alcanzado el valor límite superior, el valor de contaje ya no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado (INT), ya no se decrementa el valor de contaje.

Si el estado lógico de la entrada LD cambia a "1", el valor de contaje de la salida CV adopta el valor del parámetro PV. Mientras la entrada LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", un cambio del estado lógico de las entradas CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en la salida QU. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida QU adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QU es "0".

El estado del contador descendente se puede consultar en la salida QD. Si el valor actual de contaje es menor o igual a cero, la salida QD adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QD es "0".

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTUD (p. ej., "CTUD_DB")
- Declaración como variable local del tipo CTUD en la sección "Static" de un bloque (p. ej., #MyCTUD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente - descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con los siguientes valores de parámetros:

- Si se utiliza como contador ascendente, el valor del parámetro R debe ponerse a "1".
- Si se utiliza como contador descendente, el valor del parámetro LD debe ponerse a "1". En el parámetro PV debe indicarse en este caso el valor inicial deseado del parámetro CV.

Si un bloque diferente contiene instancias de la instrucción "Contador ascendente - descendente", estas pueden reiniciarse p. ej. inicializando el bloque de nivel superior.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Para poder ejecutar la instrucción "Contador ascendente - descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

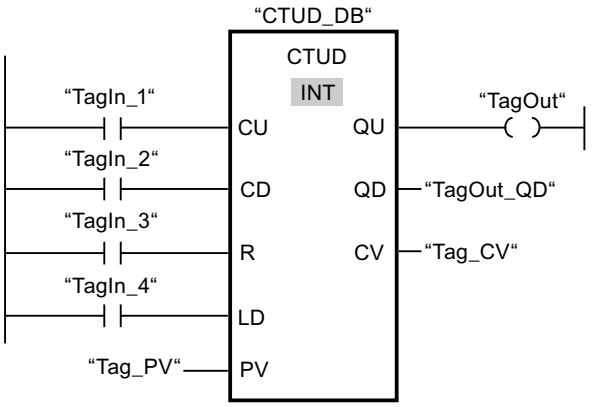
La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU
QU	Output	BOOL	I, Q, M, D, L	Estado de los contadores ascendentes
QD	Output	BOOL	I, Q, M, D, L	Estado de los contadores descendentes
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_1" o "TagIn_2" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Contador ascendente - descendente". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_1", el valor de contaje actual se incrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada "TagIn_2", el valor de contaje se decrementa en uno y se deposita

en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior (32767). Cuando se detecta un flanco de señal ascendente en la entrada CD, el valor de contaje se decrementa hasta alcanzar el valor límite inferior INT = -32768.

La salida "TagOut" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor de la entrada "Tag_PV". En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

La salida "TagOut_QD" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "TagOut_QD" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Contadores SIMATIC

S_CUD: Parametrizar e incrementar/decrementar contador

Descripción

La instrucción "Parametrizar e incrementar/decrementar contador" permite incrementar y decrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje no se modifica.

El valor de contaje sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior "0", no se decrementa más el valor de contaje.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si se inicializa el contador y el RLO en las entradas CU y CD es "1", el contador solo cuenta una vez en el próximo ciclo aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU, CD y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar/decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

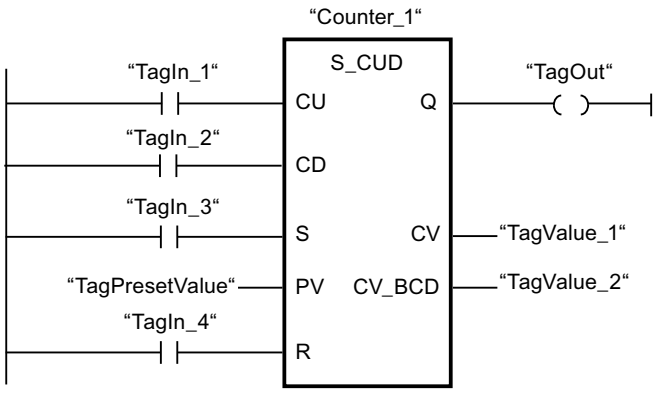
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar/decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L, T, C	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L, P	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_1" o "TagIn_2" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Parametrizar e incrementar/decrementar contador". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_1" y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_2" y el valor de contaje actual es mayor que "0", este valor se decrementará en uno.

Si el estado lógico de la entrada "TagIn_3" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_4" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

S_CU: Parametrizar e incrementar contador

Descripción

La instrucción "Parametrizar e incrementar contador" permite incrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue incrementando hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CU es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

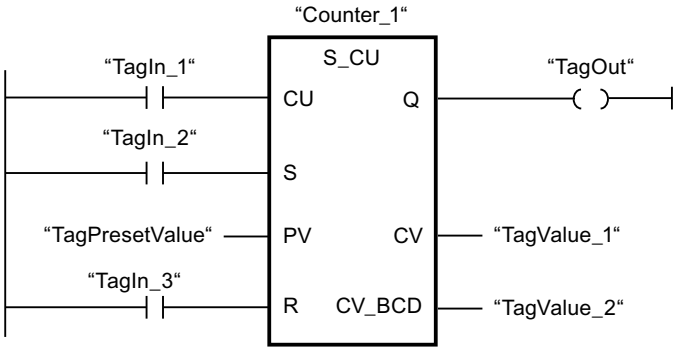
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
S	Input	BOOL	I, Q, M, D, L, T, C	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L, P	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

S_CD: Parametrizar y decrementar contador

Descripción

La instrucción "Parametrizar y decrementar contador" permite decrementar el valor de un contador. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite inferior, el valor de contaje no se decrementa más al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar y decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

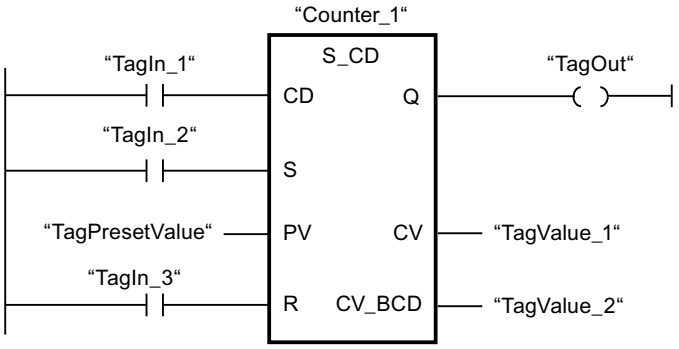
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L, P	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es mayor que "0", este valor se decrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(SC): Poner contador al valor inicial

Descripción

Con la instrucción "Poner contador al valor inicial" se ajusta el valor de un contador. La instrucción se ejecuta cuando el resultado lógico (RLO) de la entrada de la instrucción cambia de "0" a "1". Si se ejecuta la instrucción, el contador se pone al valor de contaje indicado.

La instrucción "Poner contador al valor inicial" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

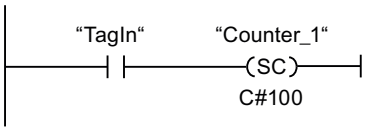
La tabla siguiente muestra los parámetros de la instrucción "Poner contador al valor inicial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Valor de contaje>	Input	WORD	I, Q, M, D, L o constante	Valor con el que se preajusta el contador codificado en BCD. (C#0 a C#999)
<Contadores>	InOut/Input	COUNTER	C	Contador que se preajusta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn" cambia de "0" a "1", el contador "Counter_1" se pone a "100".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(CU): Contador ascendente

Descripción

Con la instrucción "Contador ascendente" se puede aumentar en uno el valor del contador cuando hay un flanco ascendente en el resultado lógico (RLO). El valor de contaje puede seguir incrementándose hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

La instrucción "Contador ascendente" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

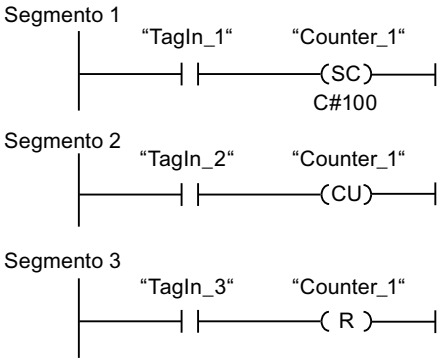
La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador cuyo valor se aumenta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se incrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

---(CD): Contador descendente

Descripción

Con la instrucción "Contador descendente" se puede reducir en uno el valor del contador cuando hay un flanco ascendente en el resultado lógico (RLO). El valor de contaje puede

seguir decrementándose hasta alcanzar el límite "0". Una vez alcanzado el valor límite, el valor de contaje no se modifica al detectarse un flanco de señal ascendente.

La instrucción "Contador descendente" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar solo en el extremo derecho del segmento.

Parámetros

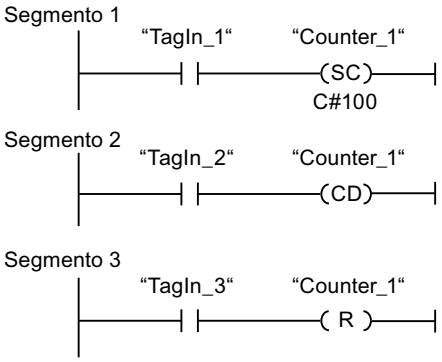
La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut/Input	COUNTER	C	Contador cuyo valor se decrementa.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se decrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Comparación

CMP ==: Igual

Descripción

La instrucción "Igual" permite consultar si son iguales el primer (<Operando1>) y segundo (<Operando2>) valor de comparación. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

Comparación de números en coma flotante

Si desea comparar tipos de datos REAL o LREAL, utilice en lugar de la instrucción "CMP ==: Igual" la instrucción "IN_RANGE: Valor dentro del rango".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Igual":

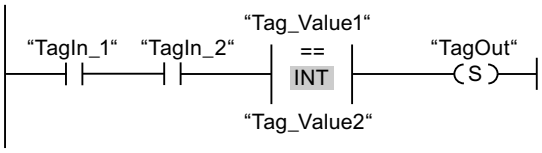
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" = "Tag_Value2".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP <>: Diferente

Descripción

La instrucción "Diferente" permite consultar si son diferentes el primer (<Operando1>) y segundo (<Operando2>) valor de comparación. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Diferente":

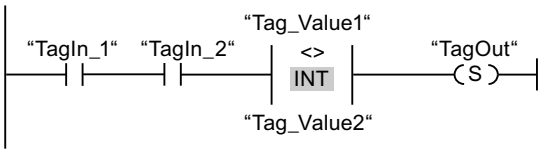
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <> "Tag_Value2".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP >=: Mayor o igual

Descripción

La instrucción "Mayor o igual" permite consultar si el primer valor de comparación (<Operando1>) es mayor o igual que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor o igual":

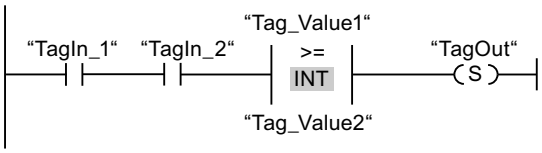
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" >= "Tag_Value2".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3244)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP <=: Menor o igual

Descripción

La instrucción "Menor o igual" permite consultar si el primer valor de comparación (<Operando1>) es menor o igual que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor o igual":

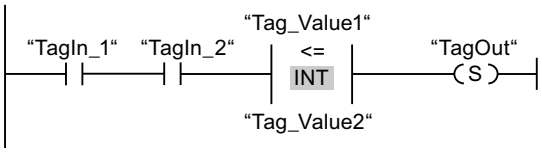
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <= "Tag_Value2".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP >: Mayor

Descripción

La instrucción "Mayor" permite consultar si el primer valor de comparación (<Operando1>) es mayor que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor":

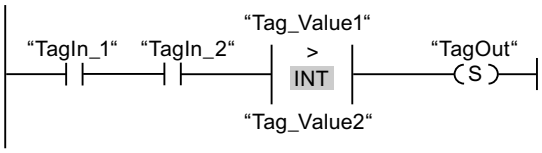
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" > "Tag_Value2".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP <: Menor

Descripción

La instrucción "Menor" permite consultar si el primer valor de comparación (<Operando1>) es menor que el segundo (<Operando2>). Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0". El RLO de la instrucción se combina lógicamente con el RLO del circuito entero como se indica a continuación:

- Con una Y lógica si la instrucción de comparación está conectada en serie.
- Con una O lógica si la instrucción de comparación está conectada en paralelo.

El primer valor de comparación (<Operando1>) se indica en el comodín de operandos situado encima de la instrucción. El segundo valor de comparación (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor":

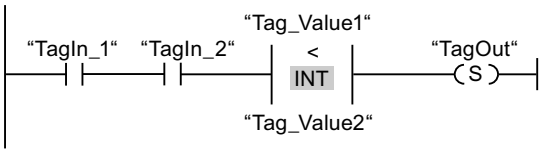
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
<Operando2>	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" < "Tag_Value2".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3244)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

Funciones matemáticas

ADD: Sumar

Descripción

La instrucción "Sumar" permite sumar el valor de la entrada IN1 al valor de la entrada IN2 y consultar la suma en la salida OUT (OUT:= IN1+IN2). Pueden indicarse como máximo 2 entradas.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar":

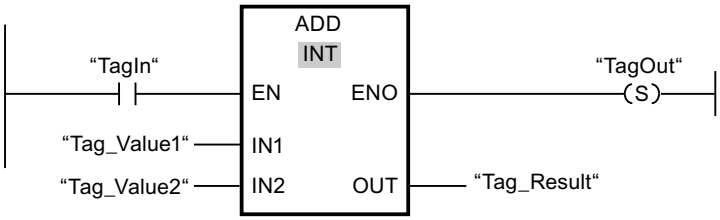
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer sumando
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo sumando
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Suma

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Sumar". El valor del operando "Tag_Value1" se suma al valor del operando "Tag_Value2". El resultado de la suma se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Ejemplo de cálculo de una ecuación (Página 3247)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SUB: Restar

Descripción

La instrucción "Restar" permite restar el valor de la entrada IN2 del valor de la entrada IN1 y consultar la diferencia en la salida OUT (OUT:= IN1-IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Restar":

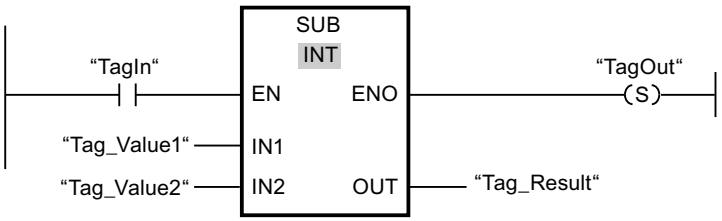
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Sustraendo
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Diferencia

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Restar". El valor del operando "Tag_Value2" se resta del valor del operando "Tag_Value1". El resultado de la resta se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MUL: Multiplicar

Descripción

La instrucción "Multiplicar" permite multiplicar el valor de la entrada IN1 por el valor de la entrada IN2 y consultar el producto en la salida OUT (OUT:= IN1*IN2). Pueden indicarse como máximo 2 entradas.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplicar":

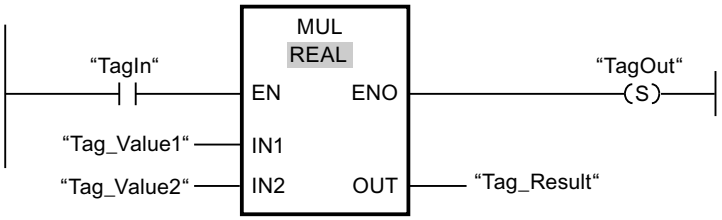
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Multiplicador
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Multiplicando
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Producto

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar". El valor del operando "Tag_Value1" se multiplica por el valor del operando "Tag_Value2". El resultado de la multiplicación se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de cálculo de una ecuación (Página 3247)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

DIV: Dividir

Descripción

La instrucción "Dividir" permite dividir el valor de la entrada IN1 entre el valor de la entrada IN2 y consultar el cociente en la salida OUT (OUT:= IN1/IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Dividir":

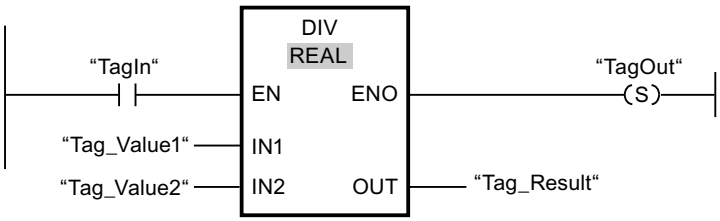
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Valor del cociente

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Dividir". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resultado de la división se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Ejemplo de cálculo de una ecuación (Página 3247)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MOD: Obtener resto de división

Descripción

La instrucción "Obtener resto de división" permite dividir el valor de la entrada IN1 entre el valor de la entrada IN2 y consultar el resto de división en la salida OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Obtener resto de división":

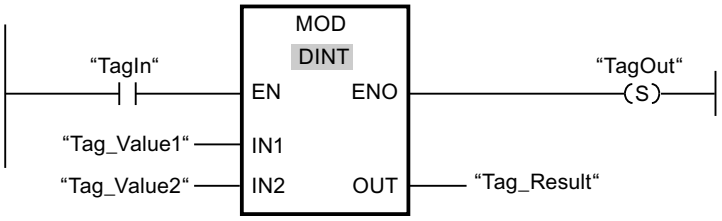
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros	I, Q, M, D, L, P	Resto de división

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Obtener resto de división". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resto de la división se almacena en el operando "Tag_Result". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

NEG: Generar complemento a dos

Descripción

La instrucción "Generar complemento a dos" permite cambiar el signo del valor de la entrada IN y consultar el resultado en la salida OUT. Por ejemplo, si la entrada IN tiene un valor positivo, se deposita el equivalente negativo de este valor en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT (sólo válido para números enteros).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a dos":

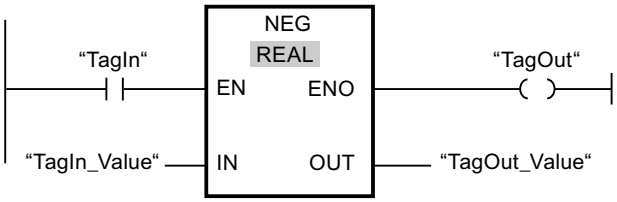
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Complemento a dos del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Generar complemento a dos". Se invierte el signo del valor de la entrada "TagIn_Value" y el resultado se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" permite calcular el valor absoluto del valor indicado en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor absoluto":

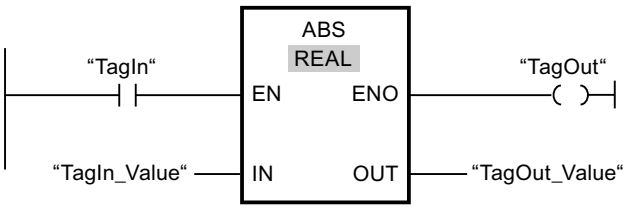
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	REAL	I, Q, M, D, L, P	Valor absoluto del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	-6.234
OUT	TagOut_Value	6.234

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor absoluto". La instrucción calcula el valor absoluto del valor de la entrada "TagIn_Value" y deposita el resultado en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor menor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

En caso de que la verificación CEI esté desactivada, en los parámetros de entrada se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, las instrucciones devolverán un resultado no definido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

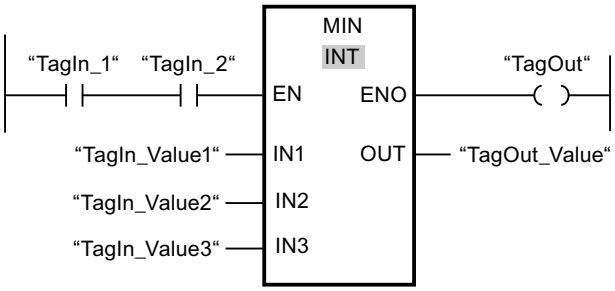
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar mínimo". La instrucción compara los valores de los operandos indicados y copia el valor menor ("TagIn_Value1") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor mayor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

En caso de que la verificación CEI esté desactivada, en los parámetros de entrada se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, las instrucciones devolverán un resultado no definido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

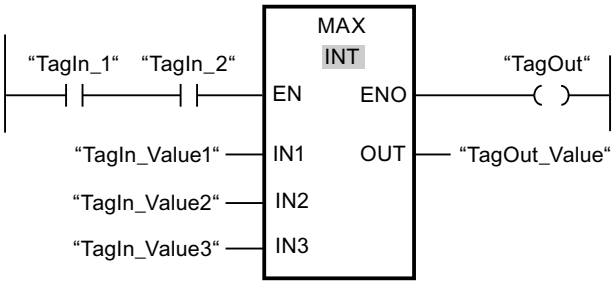
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12666
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar máximo". La instrucción compara los valores de los operandos indicados y copia el valor mayor ("TagIn_Value2") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor en la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la

salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no se define y la salida de habilitación ENO es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor de la entrada MN es mayor que el valor de la entrada MX.

Nota

En caso de que la verificación CEI esté desactivada, en los parámetros de entrada se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, las instrucciones devolverán un resultado no definido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

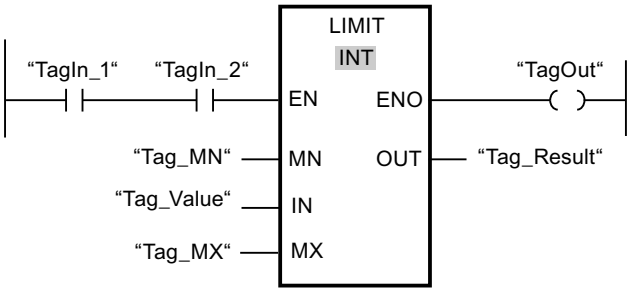
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite inferior
IN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Valor de entrada
MX	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite superior
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_MN	12000
IN	Tag_Value	8000
MX	Tag_MX	16000
OUT	Tag_Result	12000

Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se ejecuta la instrucción "Ajustar valor límite". El valor del operando "Tag_Value" se compara con los valores de los operandos "Tag_MN" y "Tag_MX". Puesto que el valor del operando "Tag_Value" es menor que el valor límite inferior, el valor del operando "Tag_MN" se copia en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" permite elevar al cuadrado el valor de la entrada IN y consultar el resultado en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

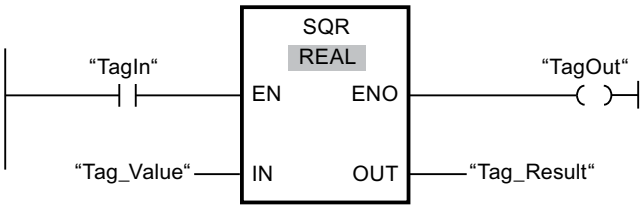
La tabla siguiente muestra los parámetros de la instrucción "Calcular cuadrado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Cuadrado del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular cuadrado". La instrucción eleva al cuadrado el valor del operando "Tag_Value" y deposita el

resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" permite obtener la raíz cuadrada del valor de la entrada IN y consultar el resultado en la salida OUT. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido. Si el valor de la entrada IN es "0", el resultado también es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

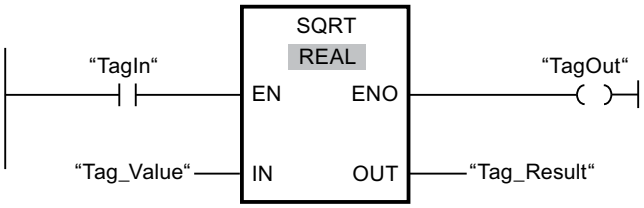
La tabla siguiente muestra los parámetros de la instrucción "Calcular raíz cuadrada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L	Raíz cuadrada del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular raíz cuadrada". La instrucción extrae la raíz cuadrada del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2,718282) del valor de la entrada IN. El resultado se deposita en la salida OUT y se puede consultar allí. La instrucción da un resultado positivo si el valor de entrada es mayor que cero.

En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

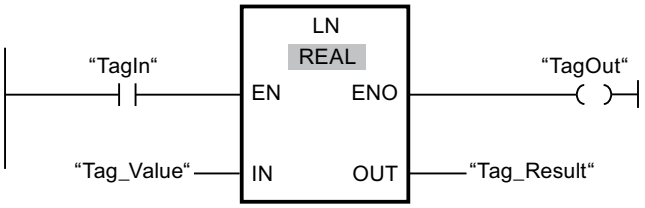
La tabla siguiente muestra los parámetros de la instrucción "Calcular logaritmo natural":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Logaritmo natural del valor de la entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular logaritmo natural". La instrucción calcula el logaritmo natural del valor de la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2,718282) del valor indicado en la entrada IN. El resultado se emite y puede consultarse en la salida OUT (OUT = e^{IN}).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

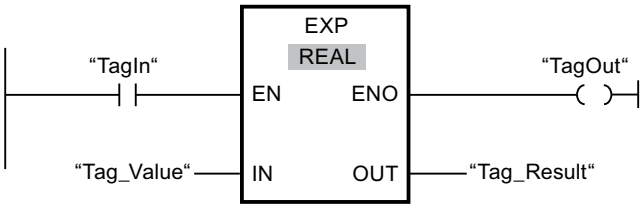
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor exponencial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Valor exponencial del valor de entrada IN

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor exponencial". La instrucción calcula la potencia en base e del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno del ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

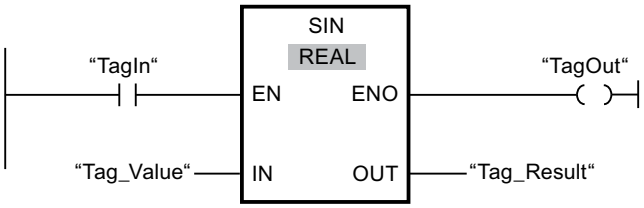
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de seno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Seno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1,570796 (π/2)
OUT	Tag_Result	1.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de seno". La instrucción calcula el seno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno del ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

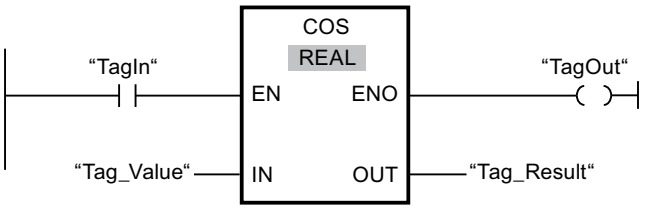
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de coseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Coseno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1,570796 (π/2)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de coseno". La instrucción calcula el coseno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente del ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

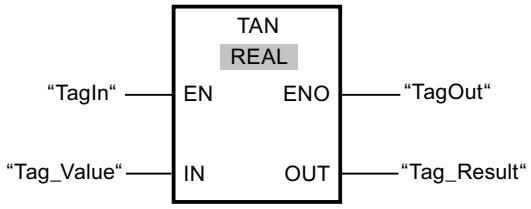
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de tangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tangente del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+3,141593 (π)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de tangente". La instrucción calcula la tangente del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno indicado en la entrada IN el tamaño del ángulo que equivale a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

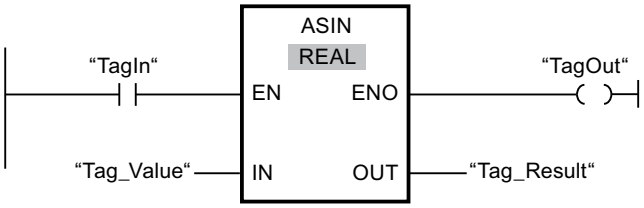
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de seno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+1,570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcoseno". La instrucción calcula el tamaño del ángulo que equivale al valor de seno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno indicado en la entrada IN el tamaño del ángulo que equivale a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre 0 y +π.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

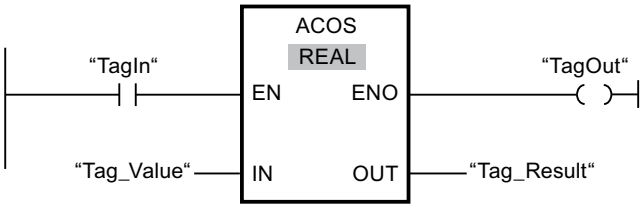
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcocoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de coseno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0
OUT	Tag_Result	+1,570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcocoseno". La instrucción calcula el tamaño del ángulo que equivale al valor de coseno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente indicado en la entrada IN el tamaño del ángulo que equivale a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

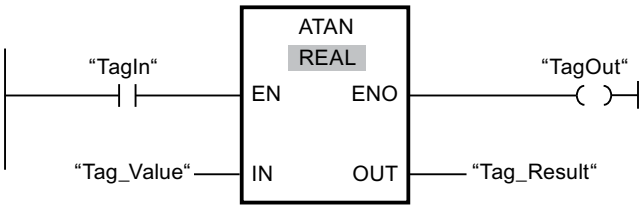
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcotangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de tangente
OUT	Output	REAL	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+0,785398 (π/4)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcotangente". La instrucción calcula el tamaño del ángulo que equivale al valor de tangente de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Principios básicos del mecanismo EN/ENO (Página 2539)
Insertar elementos KOP (Página 2678)
Seleccionar el tipo de datos de un elemento KOP (Página 2682)
Editar elementos KOP (Página 2694)
Cablear operandos en instrucciones KOP (Página 2701)
Principios básicos de la palabra de estado (Página 2545)
Activar los bits de estado en instrucciones con números enteros (Página 2551)
Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Transferencia

MOVE: Copiar valor

Descripción

La instrucción "Copiar valor" transfiere el contenido del operando de la entrada IN al operando de la salida OUT1. La transferencia se efectúa siempre por orden ascendente de direcciones.

La tabla siguiente muestra las posibles transferencias:

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD, CHAR
WORD	WORD; DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, S5TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, INT, DINT, REAL, TIME, DATE, TOD, CHAR
INT	INT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
TIME	TIME	BYTE, WORD, DWORD, INT, DINT, TIME
S5TIME	S5TIME	WORD, S5TIME
DATE	DATE	BYTE, WORD, DWORD, INT, DINT, DATE
TOD	TOD	BYTE, WORD, DWORD, INT, DINT, TOD
CHAR	CHAR	BYTE, WORD, DWORD, CHAR
COUNTER	INT, WORD, COUNTER	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	INT, WORD, TIMER	WORD, DWORD, INT, UINT, DINT, UDINT

Si la longitud de bits del tipo de datos de la entrada IN supera la longitud de bits del tipo de datos de la salida OUT1, se pierden los bits más significativos del valor de origen. Si la longitud de bits del tipo de datos en la entrada IN es inferior a la longitud de bits del tipo de datos en la salida OUT1, los bits más significativos del valor de destino se sobrescriben con ceros.

Para copiar campos y estructuras, pueden utilizarse las instrucciones "Copiar área" (BLKMOV) y "Copiar área sin interrupciones" (UBLKMOV).

Parámetros

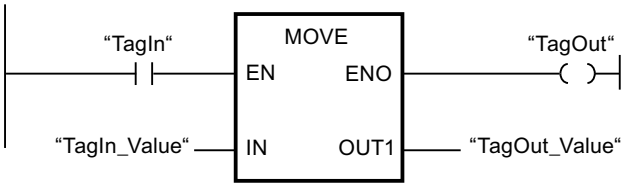
La tabla siguiente muestra los parámetros de la instrucción "Copiar valor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, DATE, TOD, CHAR, TIMER, COUNTER	I, Q, M, D, L o constante	Valor de origen
OUT1	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, DATE, TOD, CHAR, TIMER, COUNTER	I, Q, M, D, L	Dirección de destino

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar valor". La instrucción copia los contenidos del operando "TagIn_Value" en el operando "TagOut_Value" y pone la salida "TagOut" al estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

---(MCR<): Abrir áreas MCR (Página 5765)

BLKMOV: Copiar área

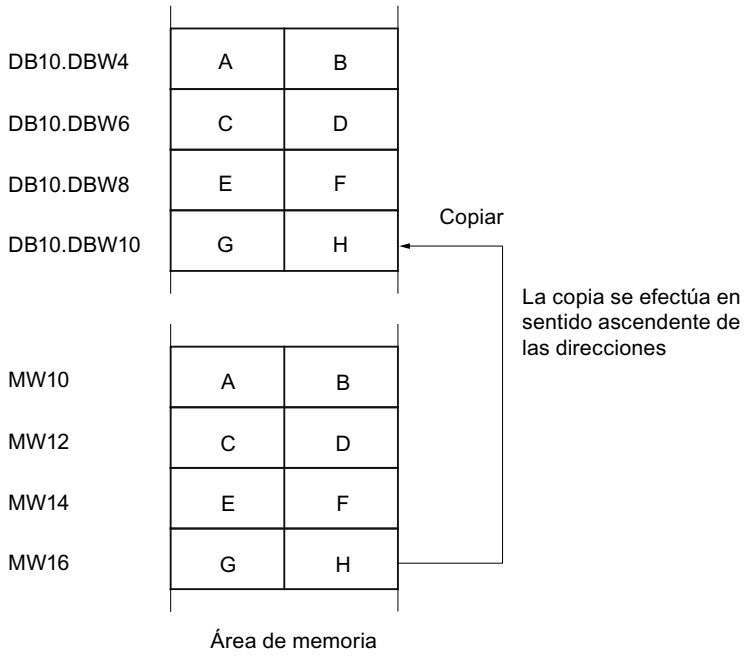
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

Mientras el área de origen no forme parte de un bloque de datos que solo figura en la memoria de carga, no hay limitaciones en cuanto a la profundidad de anidamiento.

Por el contrario, en caso de interrumpirse la ejecución de BLKMOV, en la que se copia desde un DB no relevante para la ejecución, no puede anidarse más la ejecución de la BLKMOV.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Reglas para copiar bloques de datos no relevantes para la ejecución

El área de origen también puede encontrarse en un bloque de datos no relevante para la ejecución en la memoria de carga. Los bloques de datos no relevantes para la ejecución se identifican con la palabra clave UNLINKED.

Si en la memoria de trabajo se copia un bloque de datos no relevante para la ejecución con la instrucción "Copiar área" y, al mismo tiempo, este se vuelve a cargar p. ej. desde la programadora, la ejecución de la instrucción puede retardarse varios milisegundos. Esto prolonga el ciclo del OB y puede activar la vigilancia del tiempo de ciclo.

Si un bloque de datos no relevante para la ejecución se copia con la instrucción "Copiar área" y la operación de copia se interrumpe, la instrucción no se podrá seguir ejecutando.

Si en la CPU existe la instrucción "Leer de un bloque de datos de la memoria de carga", debe utilizarse dicha instrucción para leer los bloques de datos no relevantes para la ejecución en la memoria de carga. Si se utiliza la instrucción "Copiar área" se emite el error W#16#8092.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8092	El área de origen o destino está solo en la memoria de carga.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 512 bytes como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU del S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si sólo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

FILL: Rellenar área

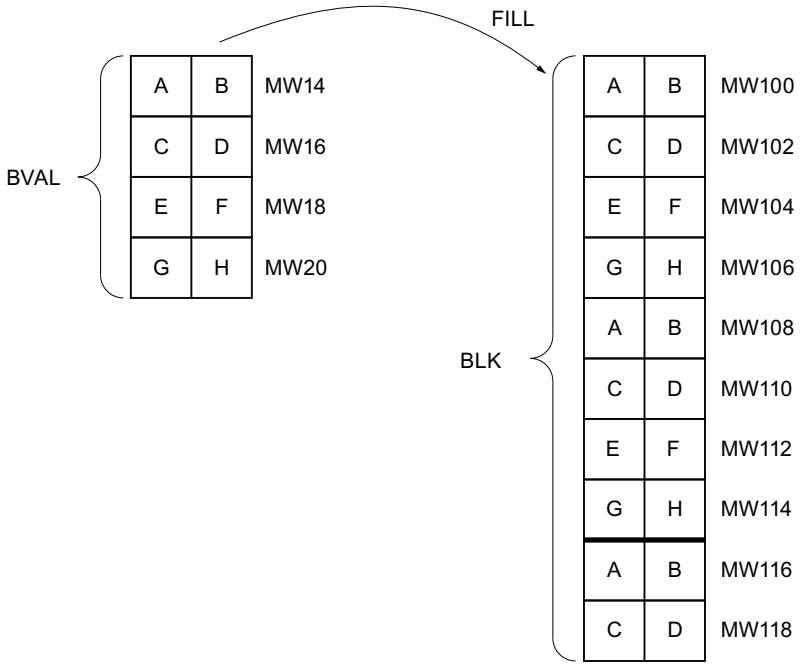
Descripción

La instrucción "Rellenar área" permite rellenar un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino las veces necesarias hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: El contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, sólo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, la longitud indicada del área debe ser divisible entre 8 ya que, de lo contrario, la instrucción no se ejecuta.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro BVAL o BLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU del S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
BVAL	Input	ANY	I, Q, M, D, L	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
BLK	Output	ANY	I, Q, M, D, L	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" lee el contenido del parámetro IN y lo convierte según los tipos de datos elegidos en el cuadro de la instrucción. El valor convertido se deposita en la salida OUT.

Encontrará más información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir valor":

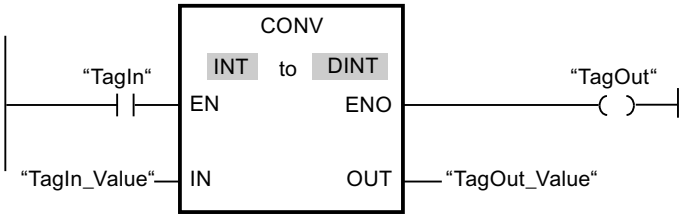
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Enteros, números en coma flotante, BCD16, BCD32	I, Q, M, D, L, P o constante	Valor que se convierte.
OUT	Output	Enteros, números en coma flotante, BCD16, BCD32	I, Q, M, D, L, P	Resultado de la conversión

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", el contenido del operando "TagIn_Value" se lee y se convierte en un número entero (32 bits). El resultado se deposita en el operando "TagOut_Value". Si la instrucción se ha ejecutado correctamente, la salida "TagOut" se pone a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Conversión explícita (Página 2522)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor de la entrada IN al siguiente número entero. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte a un número entero del tipo de datos DINT. Si el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear número":

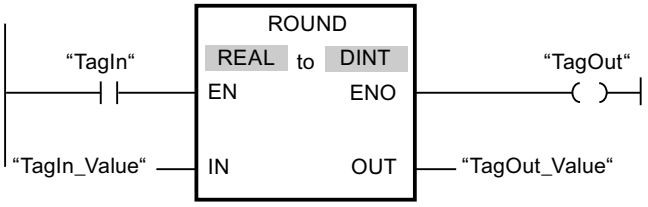
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se debe redondear.
OUT	Output	DINT	I, Q, M, D, L, P	Resultado del redondeo

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear número". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero par más próximo y se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor de la entrada IN al siguiente número entero superior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero superior. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí. El valor de salida puede ser mayor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero superior":

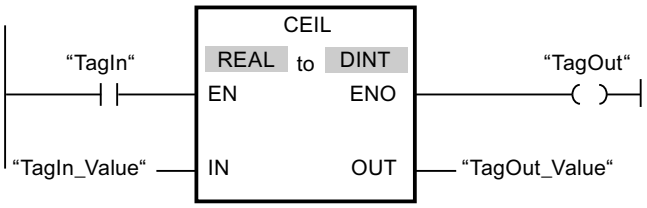
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero superior

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero superior". El número en coma flotante de la entrada "TagIn_Value" se redondea al siguiente número entero superior y se devuelve en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor de la entrada IN al siguiente número entero inferior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero inferior. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí. El valor de salida puede ser menor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero inferior":

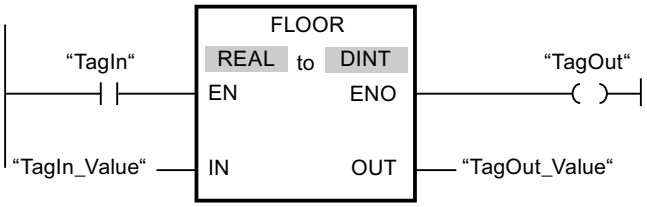
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero inferior

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero inferior". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero inferior más próximo y se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un valor entero a partir del valor de la entrada IN. El valor de la entrada IN se interpreta como número en coma flotante. La instrucción

selecciona solo la parte entera del número en coma flotante y la deposita sin decimales en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada EN devuelve el estado lógico "0".
- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Truncar a entero":

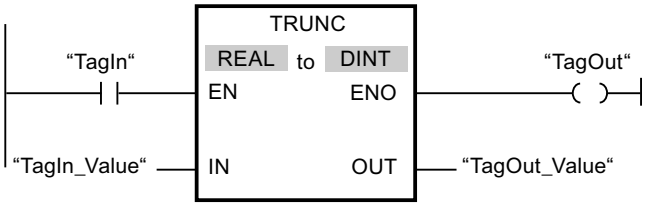
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros	I, Q, M, D, L	Parte entera del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Truncar a entero". La parte entera del número en coma flotante de la entrada "TagIn_Value" se convierte en un número entero y se deposita en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

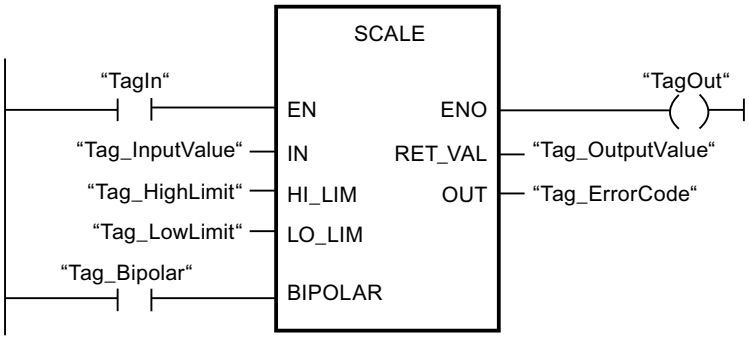
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1".
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" permite desescalar el número en coma flotante del parámetro IN en unidades físicas comprendidas entre un límite inferior y uno superior y convertirlo en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el

valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Input	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

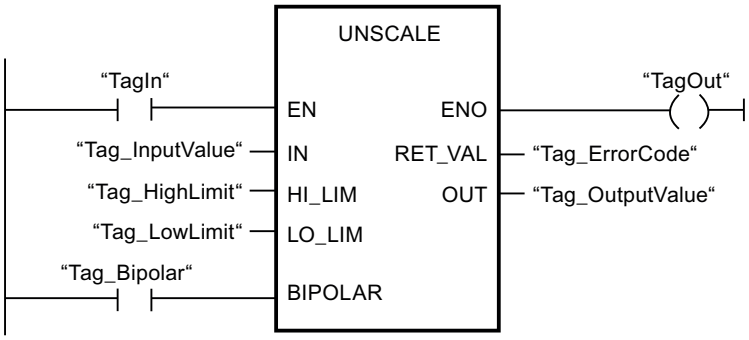
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Control del programa

---(JMP): Saltar si RLO = 1

Descripción

La instrucción "Saltar si RLO = 1" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente. El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima de la instrucción.

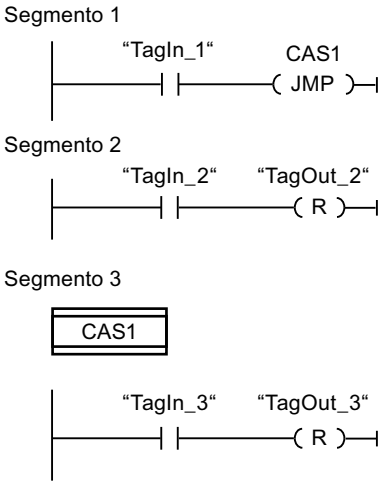
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En cada segmento no debe existir más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si no se cumple la condición en la entrada de la instrucción (RLO = 0), la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---(JMPN): Saltar si RLO = 0

Descripción

La instrucción "Saltar si RLO = 0" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente si el resultado lógico de la entrada de la instrucción es "0". El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima de la instrucción.

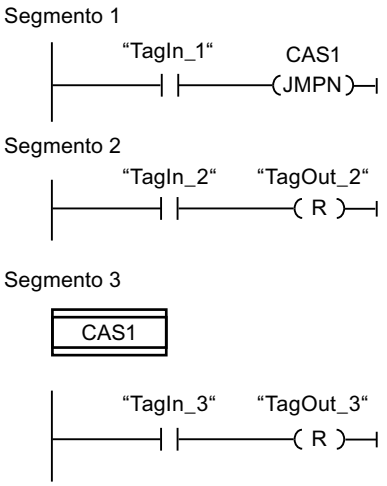
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En cada segmento no debe existir más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "0", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si el resultado lógico de la entrada de la instrucción es "1", la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "0", se ejecuta la instrucción "Saltar si RLO = 0". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

LABEL: Etiqueta

Descripción

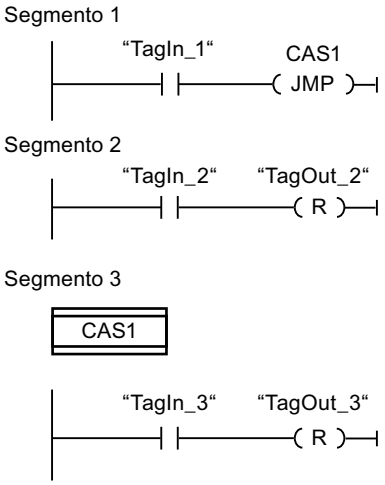
Una etiqueta sirve para marcar el segmento de destino en el que debe continuar la ejecución del programa tras un salto.

La etiqueta y la instrucción en la que se indica la etiqueta como destino del salto deben encontrarse en el mismo bloque. La denominación de una etiqueta debe ser unívoca en el bloque.

En un segmento sólo se puede colocar una etiqueta. A toda etiqueta se puede acceder desde distintas posiciones.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

--(RET): Retroceder

Descripción

La instrucción "Retroceder" permite finalizar la ejecución de un bloque.

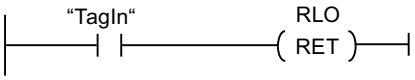
Si el resultado lógico (RLO) de la entrada de la instrucción "Retroceder" es "1", se finaliza la ejecución del programa en el bloque llamado actualmente y se continúa en el bloque que efectúa la llamada (p. ej. en el OB invocante) después de la función de llamada. El RLO determina el estado (ENO) de la función de llamada. El RLO aparece sobre la instrucción como valor de parámetro fijo. Si el RLO en la entrada de la instrucción "Retroceder" es "0", la instrucción no se ejecuta. La ejecución del programa continúa en el siguiente segmento del bloque llamado.

Si la instrucción "Retroceder" está programada en el último segmento del bloque invocado, la salida ENO pondrá la función de llamada al estado lógico actual del RLO.

Si se finaliza un bloque de organización con la instrucción "Retroceder", la CPU continúa con el programa de sistema.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico del operando "TagIn" es "1", se ejecuta la instrucción "Retroceder". La ejecución del programa se finaliza en el bloque llamado y continúa en el bloque que efectúa la llamada. La salida de habilitación ENO de la función de llamada adopta el estado lógico "1".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---(OPN): Abrir bloque de datos global

Descripción

La instrucción "Abrir bloque de datos global" permite abrir un bloque de datos (DB). El número del bloque de datos se transmite al registro del DB. Los siguientes comandos de DB acceden a los bloques correspondientes en función de los contenidos del registro.

Parámetros

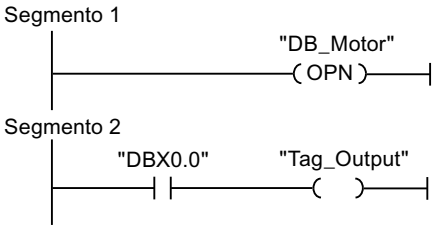
La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos global":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Bloque de datos>	Input	DB_ANY	DB	Bloque de datos que se abre.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se llama al bloque de datos "Motor_DB". El número del bloque de datos se transmite al registro del DB. En el segmento 2 se consulta el operando "DBX0.0". El estado lógico del operando "DBX0.0" se asigna al operando "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---(OPNI): Abrir bloque de datos de instancia

Descripción

La instrucción "Abrir bloque de datos de instancia" permite abrir un bloque de datos (DB). El número del bloque de datos se transmite al registro del DI. Los siguientes comandos del DI acceden a los bloques correspondientes en función de los contenidos del registro.

Parámetros

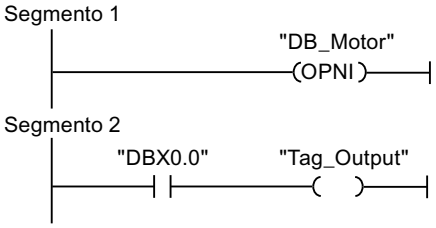
La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos de instancia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Bloque de datos>	Input	DB_ANY	DB	Bloque de datos que se abre.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se llama al bloque de datos "Motor_DB". El número del bloque de datos se transmite al registro del DI. En el segmento 2 se consulta el operando "DBX0.0". El estado lógico del operando "DBX0.0" se asigna al operando "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Control del tiempo de ejecución

COMPRESS: Comprimir la memoria de la CPU

Descripción

La instrucción "Comprimir la memoria de la CPU" permite comprimir la memoria de trabajo de la CPU o la memoria de carga RAM. Los huecos creados por haber borrado o cargado bloques, se rellenan juntando los bloques existentes.

Con la llamada de la instrucción "Comprimir la memoria de la CPU" se lanza el proceso de compresión. El proceso de compresión se reparte entre varios ciclos del programa. La vigilancia del proceso de compresión se realiza mediante los valores de los distintos parámetros. Si el parámetro BUSY de la instrucción tiene el valor "1", significa que el proceso de compresión se está ejecutando todavía. El valor "1" del parámetro DONE indica que ha finalizado el proceso de compresión. La información de error puede consultarse en el parámetro RET_VAL.

Si se está ejecutando un proceso de compresión lanzado externamente, la llamada de la instrucción "Comprimir la memoria de la CPU" origina un error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comprimir la memoria de la CPU":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
BUSY	Output	BOOL	I, Q, M, D, L	Información de si está activo el proceso de compresión.
DONE	Output	BOOL	I, Q, M, D, L	Información de si ha finalizado el proceso de compresión.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error El estado del proceso de compresión puede consultarse en los parámetros BUSY y DONE.
8091	Ya se está ejecutando un proceso de compresión lanzado externamente.
8092	La instrucción no se puede ejecutar porque se cumple una de las condiciones siguientes: • La función "Borrar bloque" está activa. • Una función de test y de puesta en servicio repercute sobre un bloque (p. ej. estado). • La función "Copiar bloques" se ha activado por un impulso externo. • La CPU H se encuentra en un proceso de acoplamiento o actualización.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU. Reiniciando el tiempo de vigilancia del ciclo es posible impedir que se produzcan errores o que la CPU pase a modo STOP.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" puede utilizarse en bloques de la clase de prioridad 1 (el OB de ciclo) así como en los bloques que se llaman desde ellos.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

Nota

En caso de utilizar la instrucción "Reiniciar tiempo de vigilancia del ciclo", debe asegurarse que no se produce un bucle sin fin en la ejecución cíclica del programa, es decir, en el OB1. De lo contrario, la CPU ya no alcanza el punto de control del ciclo. Como consecuencia no pueden ejecutarse ciertas funciones de la CPU (p. ej., la actualización de la memoria imagen de proceso).

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros.

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", la CPU pasa al estado operativo STOP y finaliza la ejecución del programa. El estado lógico de la salida de la instrucción no se evalúa.

Si el RLO a la entrada de la instrucción es "0", la instrucción no se ejecuta.

Parámetros

La instrucción "Finalizar programa" no tiene parámetros.

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" permite detener la ejecución del programa durante un periodo predeterminado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible programar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos del mecanismo EN/ENO (Página 2539)

PROTECT: Cambiar nivel de protección

Descripción

La instrucción "Cambiar nivel de protección" permite modificar el nivel de protección "1" configurado para la CPU.

El nivel de protección "1" parametrizado con la opción activada "Eliminable por contraseña" hace que sea necesario parametrizar una contraseña. Esta contraseña permite durante el funcionamiento acceder en escritura a la CPU protegida contra escritura mediante la instrucción "Cambiar nivel de protección" (p. ej. cargar).

El nivel de protección se especifica en el parámetro MODE de la instrucción. El nivel de protección actual de la CPU se visualiza en la ventana de inspección en "Propiedades > Protección".

La tabla siguiente muestra una explicación de los niveles de protección:

Nivel de protección	Explicación
1	Se permiten todas las funciones de PG. En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
2	• El programa y la configuración de la CPU no se pueden modificar. El programa contenido en la CPU se puede leer. • Se permiten las funciones de control, observación y comunicación del proceso. • Se permiten todas las funciones de información. • Si conoce la contraseña válida, podrá desactivar la protección contra escritura ajustada. • En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
3	• Aun conociendo la contraseña válida, no podrá desactivar la protección contra lectura y escritura ajustada. • Si al llamar la instrucción "Cambiar nivel de protección" existe una conexión legitimada, la llamada de la instrucción no tendrá ningún efecto en esta conexión.

Si se ha configurado el nivel de protección "2" o "3" para la CPU, la llamada de la instrucción "Cambiar nivel de protección" no tiene efecto alguno.

La efectividad del nivel de protección "2" o "3" una vez ejecutada la instrucción "Cambiar nivel de protección" puede consultarse en la lista parcial SZL con la ID de SZL W#16#0232 y el índice W#16#0004.

Nota

La instrucción "Cambiar nivel de protección" permite ajustar un nivel de protección más bajo que aquel ajustado con "Configurar hardware" de STEP 7. Supongamos que se ha configurado el nivel de protección 3 y a continuación se llama la instrucción "Cambiar nivel de protección" con el parámetro MODE = 12. Esto bloqueará la legitimación de la contraseña. Si la instrucción se llama nuevamente con el parámetro MODE = 0 o 1, el nivel de protección 3 se mantiene y será posible desactivar la protección de lectura y escritura introduciendo la contraseña válida.

Efectos de las transiciones de estado operativo en el nivel de protección

La tabla siguiente muestra qué efectos tienen las transiciones de estado operativo en el nivel de protección ajustado con la instrucción "Cambiar nivel de protección".

Acción	Efectos en el nivel de protección
• Selector de modo en "STOP" • Conexión (POWER ON) sin respaldo • Borrado total mediante el selector de modo • Cambios de configuración y/o del programa en el estado operativo "STOP" de la CPU S7-400	El nivel de protección se pone a "1". Se permiten todas las funciones de PG.
• Arranque en frío • Rearranque (en caliente)	El sistema operativo pone el nivel de protección a "1". Se permiten todas las funciones de PG. En caso necesario, llamando la instrucción "Cambiar nivel de protección" en el programa es posible ajustar el nivel de protección en "2" o "3".
Conexión (POWER ON) respaldada	No cambia
Transición de estado operativo RUN/ARRANQUE/PARADA -> STOP (debida a la llamada de la instrucción "Finalizar programa" (STP), a errores del programa sin el correspondiente OB de error o a una acción del usuario).	No cambia
S7-400: Rearranque por acción del usuario o automáticamente tras conexión (POWER ON)	No cambia

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cambiar nivel de protección":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MODE	Input	WORD	I, Q, M, D, L o constante	Nivel de protección Valores posibles: • W#16#0000: Ajuste del nivel de protección "1" • W#16#0001: Ajuste del nivel de protección "2" con legitimación de la contraseña • W#16#000C: Ajuste del nivel de protección "3" sin legitimación de la contraseña En el parámetro MODE también se pueden indicar constantes.
RET_VAL	Output	INT	I, Q, M, D, L	Si ocurre un error durante la ejecución, se emite una información de error.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
8090	Valor no admisible en el parámetro MODE.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

CiR: Controlar proceso CiR

Descripción

La instrucción "Controlar proceso CiR" permite influir en el proceso CiR en el programa de usuario como se indica a continuación:

- Bloquear el proceso CiR. En este caso, se rechazará siempre la carga de una configuración modificada desde la programadora en la CPU. El bloqueo se aplica hasta que se desbloquee con la instrucción "Controlar proceso CiR".
- Bloqueo condicional del proceso CiR estableciendo un límite superior en el parámetro FRZ_TIME para el tiempo de sincronización CiR. En este caso, sólo se permitirá cargar una configuración modificada desde la programadora en la CPU si el tiempo de sincronización CiR calculado en la CPU no excede el límite superior definido.
- Desbloquear el proceso CiR. El límite superior del tiempo de sincronización CiR se establece en el valor predeterminado de 1000 ms.
- Determinar si el proceso CiR está o no habilitado. Si el proceso CiR está habilitado o lo está condicionadamente, el límite superior actual del tiempo de sincronización CiR se indica en el parámetro A_FT de la instrucción.

Nota

Durante el tiempo de sincronización CiR, las salidas están congeladas y las entradas no se evalúan.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Controlar proceso CiR":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición Los valores admisibles del parámetro MODE son: • 0: Función de información • 1: Habilitar proceso CiR • 2: Bloquear proceso CiR • 3: Bloquear proceso CiR de forma condicionada. El límite superior del tiempo de sincronización CiR se define en FRZ_TIME.
FRZ_TIME	Input	TIME	I, Q, M, D, L o constante	Límite superior del tiempo de sincronización CiR en milisegundos. Rango de valores admisible: 60 ... 2500 ms (valor predeterminado: 1000 ms) El parámetro FRZ_TIME sólo es relevante cuando el parámetro MODE se establece en el valor "3".
RET_VAL	Output	INT	I, Q, M, D, L	Información de error Si el parámetro MODE está a "0", el parámetro RET_VAL indica si el proceso CiR está habilitado o no.
A_FT	Output	TIME	I, Q, M, D, L	Límite superior válido actualmente del tiempo de sincronización CiR

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor 1, 2 ó 3.
0001	El proceso CiR está habilitado. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor "0".

Código de error* (W#16#...)	Explicación
0002	El proceso CiR está bloqueado. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor "0".
0003	El proceso CiR está bloqueado de forma condicionada. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor "0".
8001	La CPU no está lista para un proceso CiR. Se está utilizando una CPU H en el sistema H (modo "Individual") o una CPU estándar en modo multiprocesador.
8002	El valor del parámetro MODE no es admisible.
8003	El valor del parámetro FRZ_TIME no es admisible.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

La instrucción "Controlar proceso CiR" permite programar p. ej. que no se lance ningún proceso CiR en periodos en los que se requiera la capacidad máxima de la CPU para el procesamiento.

Para ello, debe llamarse la instrucción "Controlar proceso CiR" en el programa antes de comenzar el periodo con una mayor actividad de procesos y ejecutarla con el valor "2" (Bloquear proceso CiR) en el parámetro MODE. Con ello se bloquea el proceso CiR.

Para volver a habilitar el proceso CiR tras finalizar el periodo de actividad aumentada del proceso, la instrucción "Controlar proceso CiR" debe llamarse de nuevo y ejecutarse con el valor "1" en el parámetro MODE. De manera alternativa también puede establecer el parámetro MODE en el valor "3" y bloquear así el proceso CiR.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Principios básicos del mecanismo EN/ENO (Página 2539)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Operaciones lógicas con palabras

AND: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una Y lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina con el bit 0 del valor de la entrada IN2 mediante una Y lógica. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits de los valores indicados.

El estado lógico del bit de resultado es "1" sólo si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

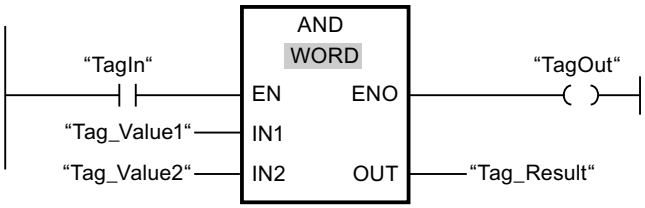
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
OUT	Output	WORD, DWORD	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0000 0000 0000 0101

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica Y". El valor del operando "Tag_Value1" se combina con el valor del operando "Tag_Value2" mediante una Y lógica. El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

OR: Operación lógica O

Descripción

La instrucción "Operación lógica O" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina con el bit 0 del valor de la entrada IN2 mediante una O lógica. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los bits de la variable indicada.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

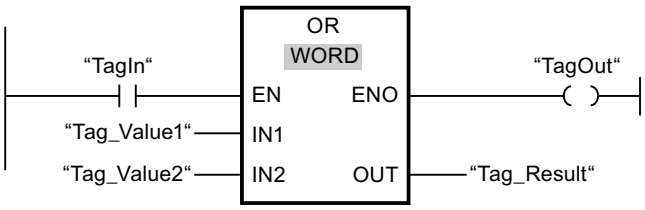
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
OUT	Output	WORD, DWORD	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O". El valor del operando "Tag_Value1" se combina con el valor del operando "Tag_Value2" mediante una O lógica. El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

XOR: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O exclusiva y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina con el bit 0 del valor de la entrada IN2 mediante una O exclusiva. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits del valor indicado.

El estado lógico del bit de resultado es "1" si el estado lógico de uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "1" ó "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

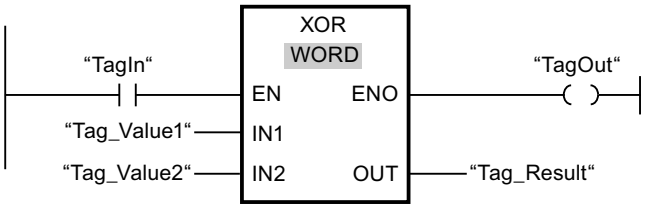
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
OUT	Output	WORD, DWORD	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	0101 0101 0101 0101
IN2	Tag_Value2	0000 0000 0000 1111
OUT	Tag_Result	0101 0101 0101 1010

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O-exclusiva". El valor del operando "Tag_Value1" se combina con el valor del operando "Tag_Value2" mediante una O-exclusiva. El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

INV: Complemento a 1

Descripción

La instrucción "Complemento a 1" permite invertir el estado lógico de los bits de la entrada IN. Al ejecutar la instrucción, el valor de la entrada IN se combina con una plantilla hexadecimal (W#16#FFFF para números de 16 bits o DW#16#FFFFFFFF para números de 32 bits) mediante una operación lógica O-exclusiva. Esto invierte el estado lógico de los distintos bits y deposita el resultado en la salida OUT.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Complemento a 1":

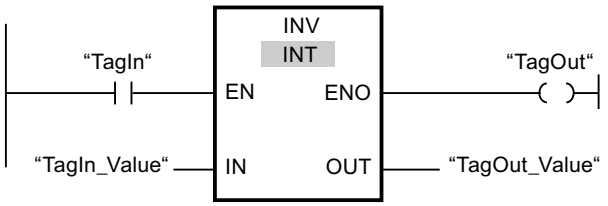
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Enteros	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros	I, Q, M, D, L, P	Complemento a 1 del valor de la entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	W#16#000F
OUT	TagOut_Value	W#16#FFF0

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Complemento a 1". La instrucción invierte el estado lógico de los distintos bits de la entrada "TagIn_Value" y escribe el resultado en la salida "TagOut_Value". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

DECO: Descodificar

Descripción

La instrucción "Descodificar" activa un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor de la entrada IN y activa el bit del valor de salida, cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor de la entrada IN es mayor que 31, se ejecuta una instrucción modulo 32.

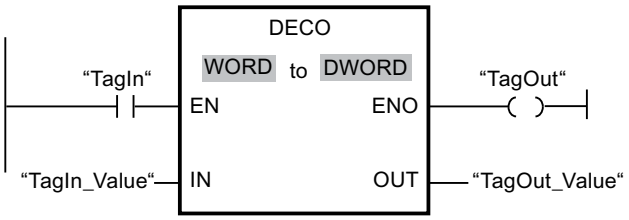
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

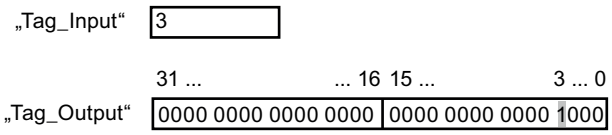
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida.
OUT	Output	DWORD	I, Q, M, D, L, P	Valor de salida

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Descodificar". La instrucción lee el número de bit "3" del valor del operando "TagIn_Value" de la entrada y activa el tercer bit del valor del operando "TagOut_Value" de la salida.

Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

ENCO: Codificar

Descripción

La instrucción "Codificar" permite leer el número del bit menos significativo que está activado en el valor de entrada y depositarlo en la salida OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor de la entrada IN y escribe su número de bit en la variable de la salida OUT.

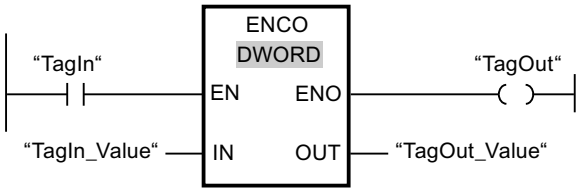
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

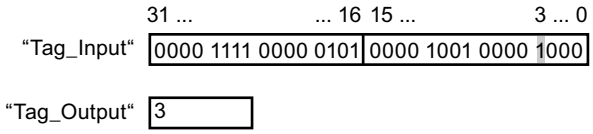
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Codificar". La instrucción selecciona el bit menos significativo que está activado en la entrada "TagIn_Value" y escribe el bit "3" en la variable de la salida "TagOut_Value".

Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite seleccionar, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copiar su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G tiene el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

Para poder procesar la instrucción, las variables de todos los parámetros deben ser del mismo tipo de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

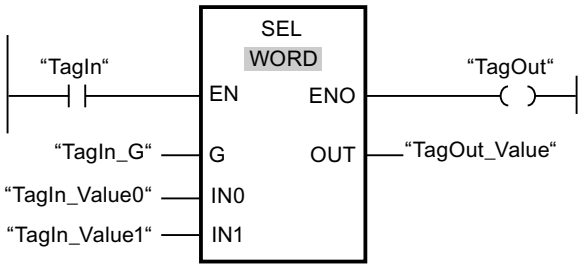
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
G	Input	BOOL	I, Q, M, D, L, T, C	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Seleccionar". Dependiendo del estado lógico de la entrada "TagIn_G", se selecciona el valor de la entrada "TagIn_Value0" ó "TagIn_Value1" y se copia en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

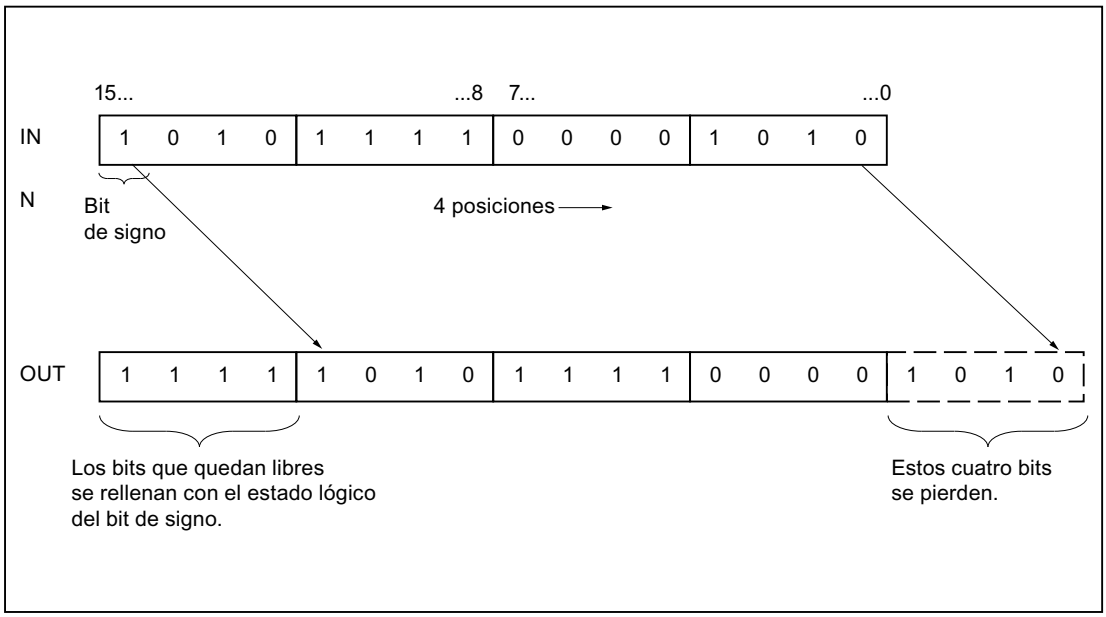
La instrucción "Desplazar a la derecha" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la derecha las posiciones de bit disponibles.

En los valores sin signo, se rellenan con ceros los bits que quedan libres en el área izquierda del operando al realizar el desplazamiento. Si el valor indicado lleva signo, las posiciones libres se ocupan con el estado lógico del bit de signo.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos INT se desplaza cuatro posiciones de bit hacia la derecha:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

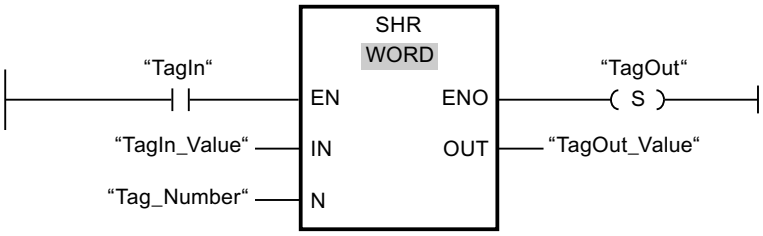
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L, P o constante	Valor que se desplaza.
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de bits que se desplaza el valor.
OUT	Output	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la derecha". El contenido del operando "TagIn_Value" se desplaza tres posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

SHL: Desplazar a la izquierda

Descripción

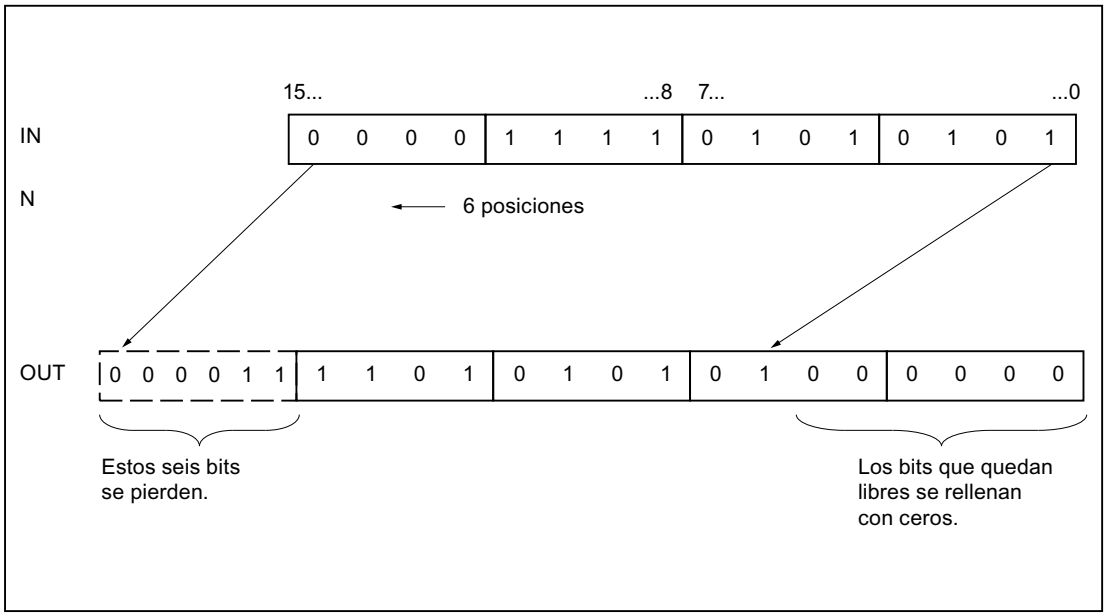
La instrucción "Desplazar a la izquierda" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la izquierda las posiciones de bit disponibles.

Los bits que quedan libres en el área derecha del operando al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

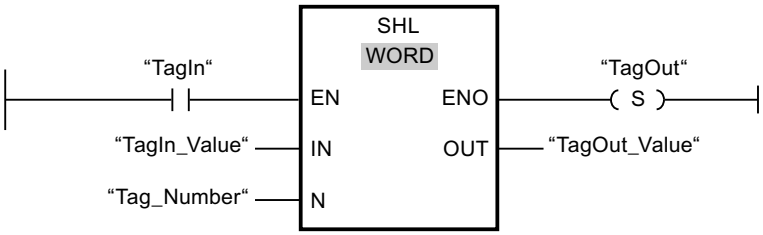
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L, P o constante	Valor que se desplaza
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que se desplaza el valor
OUT	Output	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la izquierda". El contenido del operando "TagIn_Value" se desplaza cuatro posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

ROR: Rotar a la derecha

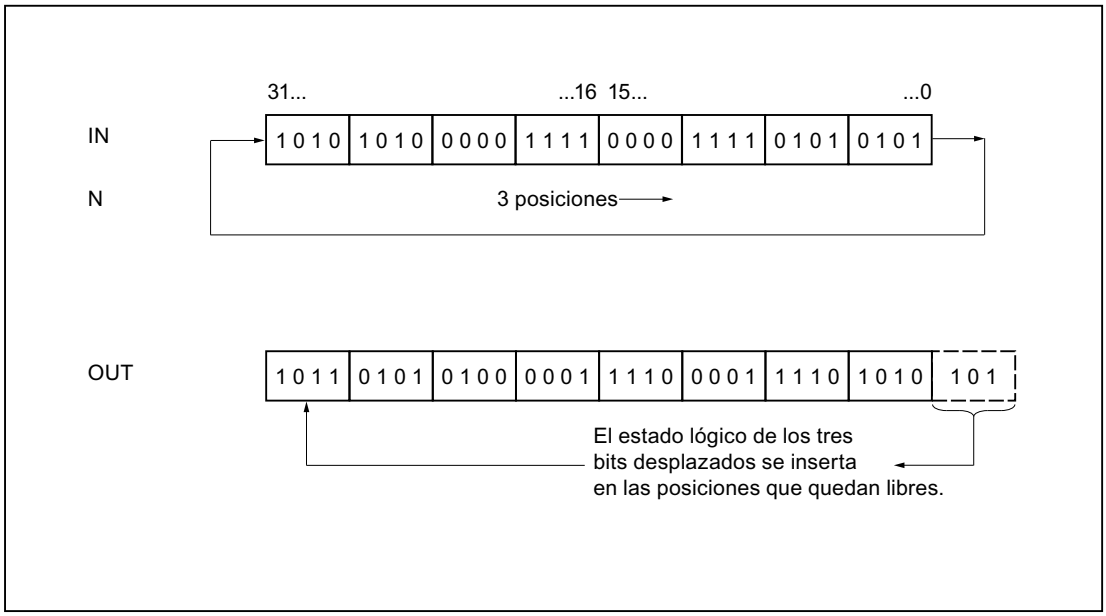
Descripción

La instrucción "Rotar a la derecha" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Parámetros

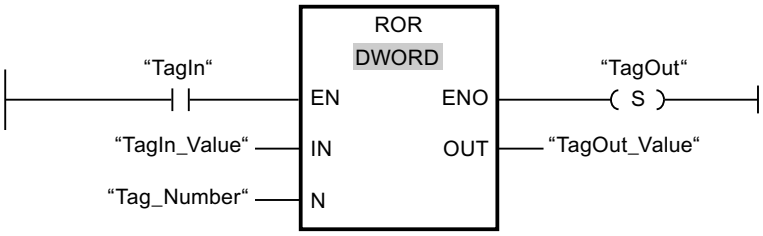
La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, DINT	I, Q, M, D, L, P o constante	Valor que se rota
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que rota el valor
OUT	Output	BYTE, WORD, DWORD, DINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	1010 1010 0000 1111 0000 1111 0101 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1101 0101 0000 0111 1000 0111 1010

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la derecha". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

ROL: Rotar a la izquierda

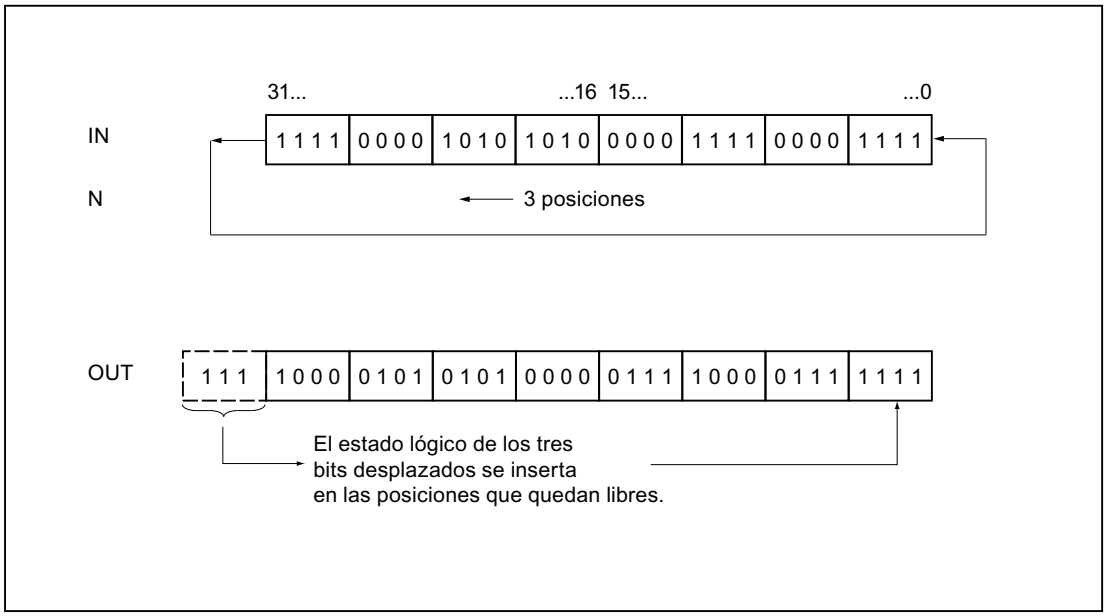
Descripción

La instrucción "Rotar a la izquierda" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Parámetros

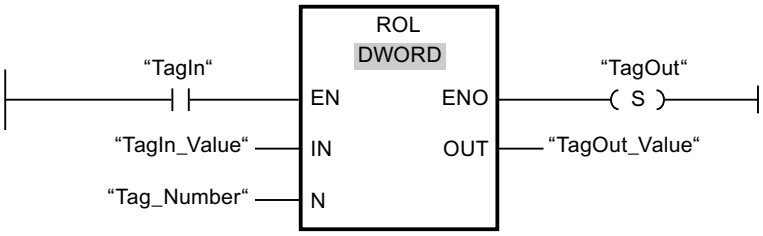
La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, DINT	I, Q, M, D, L, P o constante	Valor que se rota
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que rota el valor
OUT	Output	BYTE, WORD, DWORD, DINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	1111 0000 1010 1010 0000 1111 0000 1111
N	Tag_Number	5
OUT	TagOut_Value	0001 0101 0100 0001 1110 0001 1111 1110

Si la entrada "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la izquierda". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Seleccionar el tipo de datos de un elemento KOP (Página 2682)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Otras instrucciones

---| |---: Consultar si el bit de estado es 1

Descripción

La instrucción "Consultar si el bit de estado es 1" permite consultar el estado lógico de los siguientes bits de estado:

- BR: Resultado binario
- OV: Desbordamiento
- OS: Desbordamiento con memoria
- CC 0 y CC 1: Bits de visualización

Qué bits de estado se consultan se determina mediante la selección del símbolo correspondiente en la lista desplegable de la instrucción. La siguiente tabla muestra las posibilidades de consulta de los bits de estado:

Símbolo	Consulta	Explicación
==0	CC 0 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es igual a 0.
<>0	CC 0 = 1 y CC 1 = 0 o CC 0 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es diferente a 0.
>=0	CC 0 = 0	La instrucción consulta si el resultado de una función matemática es mayor o igual que 0.
<=0	CC 1 = 0	La instrucción consulta si el resultado de una función matemática es menor o igual que 0.
>0	CC 0 = 1 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es mayor que 0.
<0	CC 1 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es menor que 0.
UO	CC 1 = 1 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática no es válido. La causa de un resultado inválido puede ser un número en coma flotante no válido o una división entre cero.

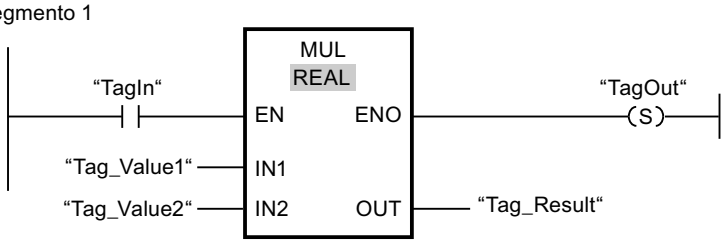
Símbolo	Consulta	Explicación
OV	OV = 1	La instrucción consulta si se ha producido un desbordamiento del rango numérico en una instrucción.
OS	OS = 1	La instrucción consulta si se ha producido un desbordamiento del rango numérico con memoria en una instrucción.
BR	BR = 1	La instrucción consulta el estado lógico del bit de estado BR.

La instrucción "Consultar si el bit de estado es 1" funciona como un contacto normalmente abierto. Si la respuesta es afirmativa, la instrucción devuelve el estado lógico "1". Si no se cumple la condición, la instrucción devuelve el estado lógico "0".

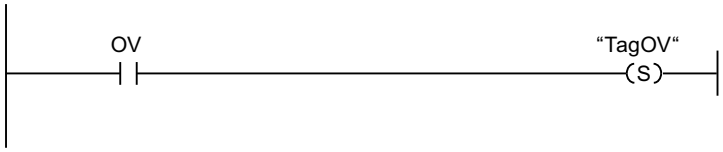
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1



Segmento 2



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar" (MUL). Si se produce un rebase por exceso durante la ejecución de la instrucción "Multiplicar" se activa el bit de estado OV. Tras la consulta del bit de estado OV, se ejecuta la instrucción "Activar salida" (S) en el segmento 2 y se activa el operando "TagOV".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

---| / |---: Consultar si el bit de estado es 0

Descripción

Con la instrucción "Consultar si el bit de estado es 0" es posible consultar el estado lógico de los siguientes bits de estado y negar el resultado de la consulta:

- BR: Resultado binario
- OV: Desbordamiento
- OS: Desbordamiento con memoria
- CC 0 y CC 1: Bits de visualización

Qué bits de estado se consultan se determina mediante la selección del símbolo correspondiente en la lista desplegable de la instrucción. La siguiente tabla muestra las posibilidades de consulta de los bits de estado:

Símbolo	Consulta	Explicación
==0	CC 0 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es igual a 0.
<>0	CC 0 = 1 y CC 1 = 0 o CC 0 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es diferente a 0.
>=0	CC 0 = 0	La instrucción consulta si el resultado de una función matemática es mayor o igual que 0.
<=0	CC 1 = 0	La instrucción consulta si el resultado de una función matemática es menor o igual que 0.
>0	CC 0 = 1 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es mayor que 0.
<0	CC 1 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es menor que 0.
UO	CC 1 = 1 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática no es válido. La causa de un resultado inválido puede ser un número en coma flotante no válido o una división entre cero.
OV	OV = 1	La instrucción consulta si en una instrucción se ha rebasado el rango numérico.

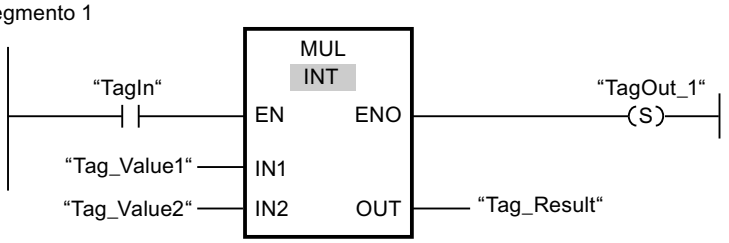
Símbolo	Consulta	Explicación
OS	OS = 1	La instrucción consulta si en una instrucción se ha producido un desbordamiento del rango numérico con memoria.
BR	BR = 1	La instrucción consulta el estado lógico del bit de estado BR.

La instrucción "Consultar si el bit de estado es 0" funciona como un contacto normalmente cerrado. Si la respuesta es afirmativa, la instrucción devuelve el estado lógico "0". Si no se cumple la condición, la instrucción devuelve el estado lógico "1".

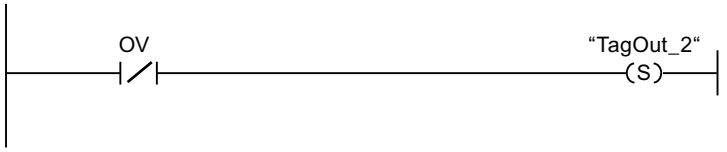
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1



Segmento 2



Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar" (MUL). Si durante la ejecución de la instrucción no se produce un rebase por exceso, se pone a "0" el bit de estado OV. Tras la consulta del bit de estado OV, se ejecuta la instrucción "Activar salida" (S) en el segmento 2 y se activa el operando "TagOut_2".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

---(CALL): Llamar bloque

Descripción

La instrucción "Llamar bloque" llama una función (FC) sin parámetros. En función de la operación lógica precedente, se puede tratar de una llamada absoluta o condicionada.

En la sección de instrucciones de una función (FC) no puede indicar como operando ningún parámetro del tipo de datos BLOCK_FC en una llamada condicional. En un bloque de función (FB) tampoco puede indiciar como operando un parámetro del tipo BLOCK_FC.

Solo se ejecutará una llamada condicional cuando el resultado lógico (RLO) sea "1". Si no se ejecuta una llamada condicionada, después de la operación de llamada el RLO es "0". Si se ejecuta la operación, ésta trabaja del modo siguiente:

- Guarda la dirección de retorno del bloque que efectúa la llamada.
- Guarda los dos registros de bloque de datos (bloque de datos y bloque de datos de instancia).
- Actualiza el área de datos locales para la FC o SFC llamada.
- Desplaza el bit MA (bit MCR activo) a la pila de bloques (pila BSTACK).

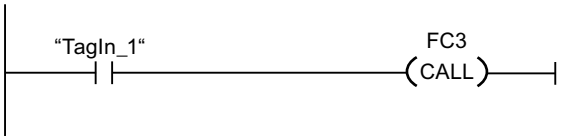
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Llamar bloque":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Bloque>	Output	BLOCK_FC	-	Función llamada

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Se llama a la función "FC3" cuando el operando "TagIn_1" tiene el estado lógico "1".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

---(SAVE): Guardar RLO en bit BR

Descripción

La instrucción "Guardar RLO en bit BR" permite guardar el resultado lógico (RLO) en el resultado binario (BR). Durante la ejecución, la instrucción transfiere el estado lógico del resultado lógico actual al bit de estado BR. La instrucción trabaja independientemente de las condiciones y no influye en ningún otro bit de estado.

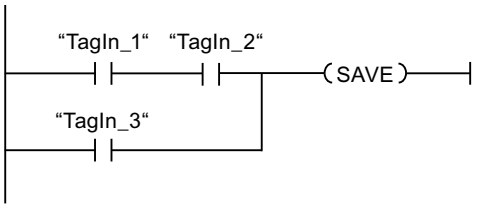
Nota

Tras aplicar la instrucción "Guardar RLO en bit BR", el bit BR puede ser modificado nuevamente por instrucciones posteriores del mismo bloque o de bloques de nivel inferior.

La instrucción "Guardar RLO en bit BR" también se utiliza para controlar el estado de ejecución de un bloque. Si, p. ej., la instrucción "Guardar RLO en bit BR" se programa al final de un bloque, el bit BR se pone al estado lógico del resultado lógico actual del bloque. Si el bloque se ejecuta sin fallos, tanto el bit BR como la salida de habilitación ENO se ponen al estado lógico "1" en el bloque llamado. El estado lógico "0" se asigna al bit BR si se han producido errores durante el procesamiento del bloque. Mediante consulta del bit BR es posible programar un tratamiento de errores.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El bit BR se pone al estado lógico "1" si se da una de las siguientes condiciones:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "1".

Consulte también

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Ejemplo del mecanismo EN/ENO en KOP (Página 2542)

---(MCR<): Abrir áreas MCR

Descripción

La instrucción "Abrir áreas MCR" guarda el resultado lógico (RLO) en la pila MCR y cambia la dependencia MCR dentro de un área MCR (MCR = Master Control Relay).

La pila MCR puede contener como máximo 8 entradas y funciona como una memoria intermedia LIFO (Last In, First Out). Cuando la pila está llena, la instrucción genera un mensaje de error (MCRF).

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

El principio y el fin de un área MCR se definen mediante las instrucciones "Activar área MCR" (MCRA) y "Desactivar área MCR" (MCRD). Dentro del área MCR definida es posible activar y desactivar la dependencia MCR para una o varias partes del programa.

La dependencia MCR se activa si el RLO es "1" antes de procesar la instrucción. Si el RLO es "0" antes de la instrucción, la dependencia MCR se desactiva. El final de la sección del programa dependiente de MCR se programa mediante la instrucción "Cerrar áreas MCR". La sección del programa que se encuentra entre las dos instrucciones se ve influenciada por el MCR. Solo están afectadas las siguientes instrucciones dependientes de MCR:

- Asignación
- Activar salida (S)
- Desactivar salida (R)
- Flipflop de desactivación/activación (RS)
- Flipflop de activación/desactivación (SR)
- Copiar valor (MOVE)

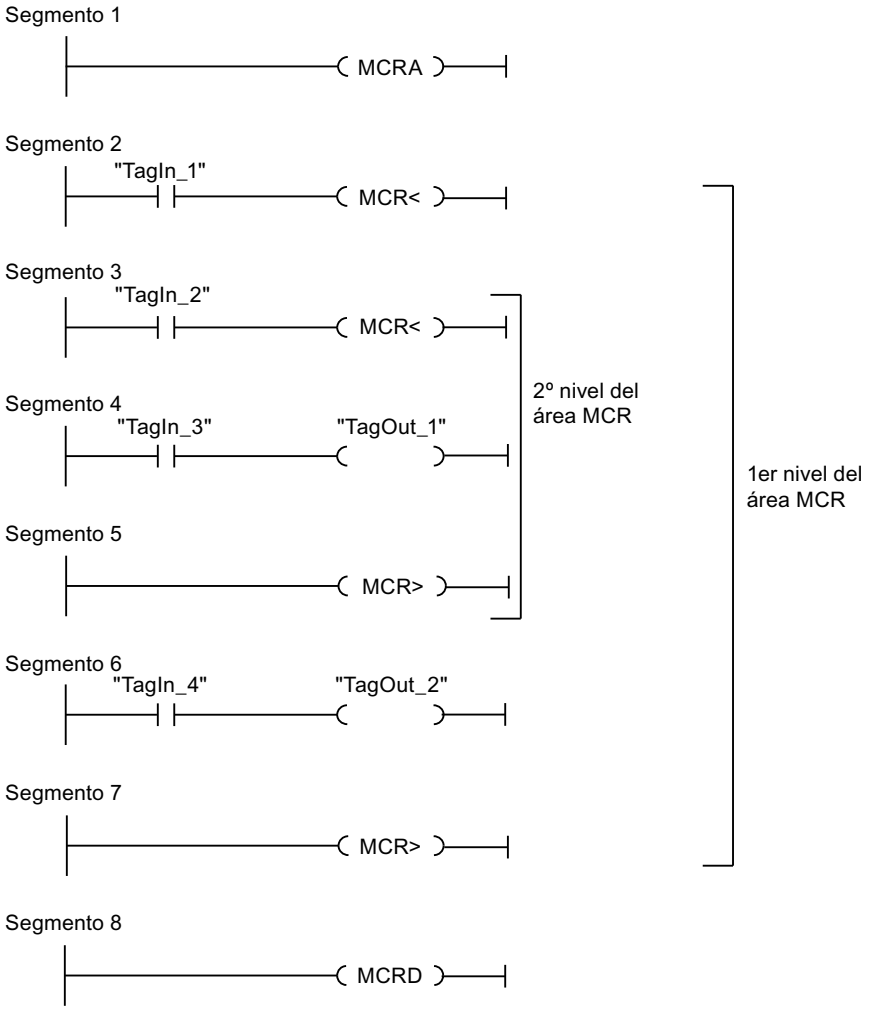
También es posible anidar la activación y desactivación de la dependencia MCR programando la instrucción "Abrir áreas MCR" en una sección del programa dependiente de MCR. Esta instrucción se puede anidar hasta 8 niveles de profundidad. La dependencia MCR de la parte del programa de nivel superior controla las dependencias MCR en todas las partes del programa subordinadas. Si la dependencia MCR en la parte del programa de nivel superior está desactivada, no es posible activarla en una parte subordinada del programa dependiente de MCR.

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

Las instrucciones "Abrir áreas MCR" y "Cerrar áreas MCR" deben programarse siempre de dos en dos. Si una de las dos instrucciones no está programada, se dispara un error de pila MCR.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se conecta la dependencia MCR en el primer nivel del área MCR. En ese caso, el estado lógico del operando "TagIn_4" está asignado al operando "TagOut_2" (segmento 6). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_2". En el segmento 7 se desconecta la dependencia MCR del área MCR.

En el segmento 3 se activa la dependencia MCR en el segundo nivel del área MCR si el operando "TagIn_2" devuelve el estado lógico "1" y el primer nivel del área MCR está activado. En ese caso, el estado lógico del operando "TagIn_3" se asigna al operando "TagOut_1". Si el operando "TagIn_2" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_1". En el segmento 5 se desactiva la dependencia MCR en el segundo nivel del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

El MCR (Master Control Relay) (Página 2564)

---(MCR>): Cerrar áreas MCR (Página 5767)

---(MCRA): Activar área MCR (Página 5769)

---(MCRD): Desactivar área MCR (Página 5770)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---(MCR>): Cerrar áreas MCR

Descripción

La instrucción "Cerrar áreas MCR" cierra una sección de programa dependiente de MCR (MCR = Master Control Relay). También borra el resultado lógico (RLO) de la última entrada de la pila MCR.

La pila MCR puede contener como máximo 8 entradas y funciona como una memoria intermedia LIFO (Last In, First Out). Cuando la pila ya está vacía, la instrucción genera un mensaje de error (MCRF).

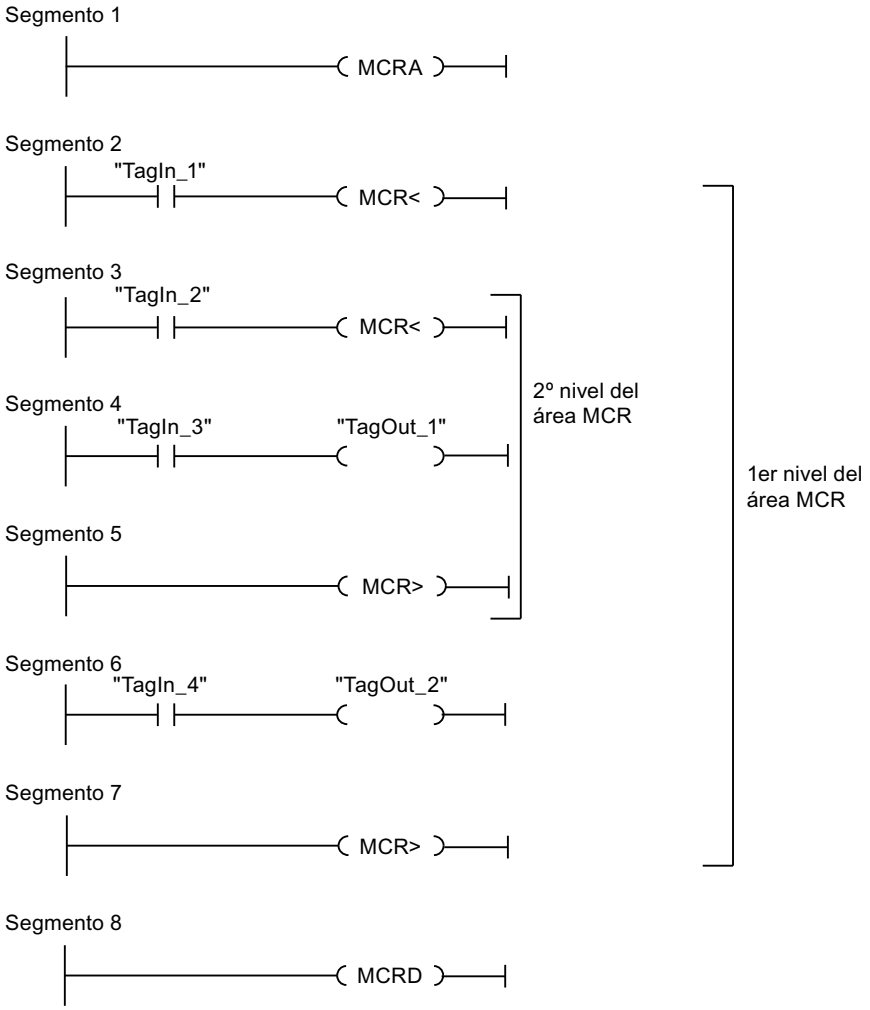
Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

La dependencia MCR se programa con la instrucción "Abrir áreas MCR". Las instrucciones "Abrir áreas MCR" y "Cerrar áreas MCR" deben programarse siempre de dos en dos. Si una de las dos instrucciones no está programada, se dispara un error de pila MCR.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se conecta la dependencia MCR en el primer nivel del área MCR. En ese caso, el estado lógico del operando "TagIn_4" está asignado al operando "TagOut_2" (segmento 6). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_2". En el segmento 7 se desconecta la dependencia MCR del área MCR.

En el segmento 3 se activa la dependencia MCR en el segundo nivel del área MCR si el operando "TagIn_2" devuelve el estado lógico "1". En ese caso, el estado lógico del operando "TagIn_3" se asigna al operando "TagOut_1". Si el operando "TagIn_2" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_1" independientemente del estado lógico del operando "TagIn_3". En el segmento 5 se desactiva la dependencia MCR en el segundo nivel del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

---(MCR<): Abrir áreas MCR (Página 5765)

---(MCRA): Activar área MCR (Página 5769)

---(MCRD): Desactivar área MCR (Página 5770)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---(MCRA): Activar área MCR

Descripción

La instrucción "Activar área MCR" define el principio de un área MCR. Dentro de dicha área es posible activar y desactivar la dependencia MCR con la ayuda de las instrucciones siguientes:

- ---(MCR<): Abrir áreas MCR
- ---(MCR>): Cerrar áreas MCR

El fin del área MCR se define con la instrucción "Desactivar área MCR" (MCRD). Las instrucciones para definir el principio y fin de un área MCR siempre deben utilizarse por parejas.

Nota

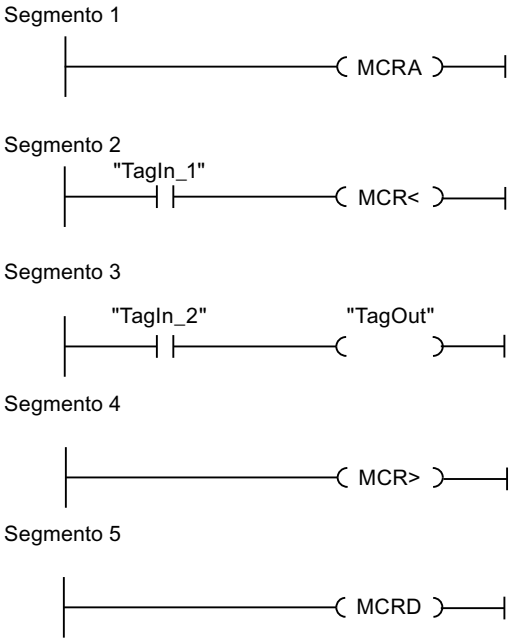
Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

La instrucción se ejecuta sin considerar ni influir en los bits de estado.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se activa la dependencia MCR. En ese caso, el estado lógico del operando "TagIn_2" está asignado al operando "TagOut" (segmento 3). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut" independientemente del estado lógico del operando "TagIn_2". En el segmento 4 se desconecta la dependencia MCR del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

---(MCR<): Abrir áreas MCR (Página 5765)

---(MCR>): Cerrar áreas MCR (Página 5767)

---(MCRD): Desactivar área MCR (Página 5770)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

---(MCRD): Desactivar área MCR

Descripción

La instrucción "Desactivar área MCR" define el final de un área MCR.

El principio de un área MCR se define con la instrucción "Activar área MCR" (MCRA). Las instrucciones para definir el principio y fin de un área MCR siempre deben utilizarse por parejas.

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

Dentro de un área MCR es posible programar la dependencia MCR con la ayuda de las instrucciones siguientes:

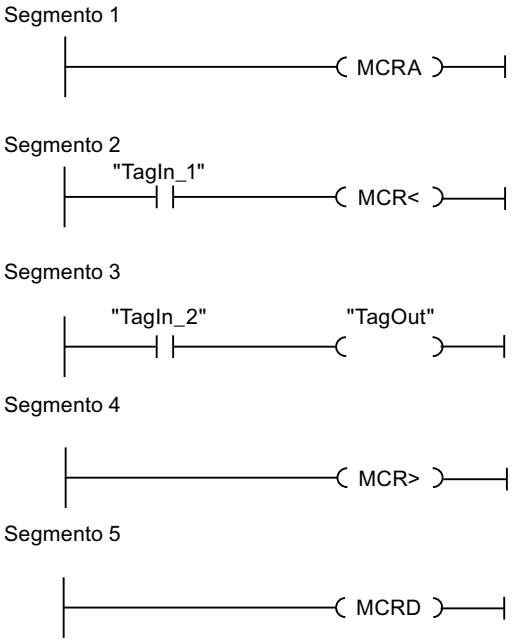
- ---(MCR<): Abrir áreas MCR
- ---(MCR>): Cerrar áreas MCR

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

La instrucción se ejecuta sin considerar ni influir en los bits de estado.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se activa la dependencia MCR. En ese caso, el estado lógico del operando "TagIn_2" está asignado al operando "TagOut" (segmento 3). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut" independientemente del estado lógico del operando "TagIn_2". En el segmento 4 se desconecta la dependencia MCR del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

---(MCR<): Abrir áreas MCR (Página 5765)

---(MCR>): Cerrar áreas MCR (Página 5767)

---(MCRA): Activar área MCR (Página 5769)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

SET: Activar área por bits

Descripción

La instrucción "Activar área por bits" establece el estado lógico de los bits en "1" en un área indicada. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben activarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta sólo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

Parámetros

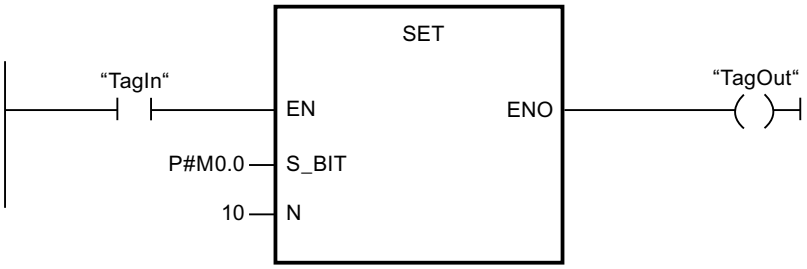
La tabla siguiente muestra los parámetros de la instrucción "Activar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del área cuyos bits se activan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben activarse

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción establece 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

SETP: Activar mapa de bits en el área de periferia

Descripción

La instrucción "Activar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "1". Paralelamente a los bits de periferia, también se activan los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe activarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no se ha conectado periferia para una parte del mapa de bits seleccionado, la instrucción intentará activar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se activa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido obtienen el valor del bit correspondiente en la memoria imagen de proceso de las salidas.

> ⚠ ADVERTENCIA
>
> **Consecuencias no deseadas**
>
> El ajuste de los bits al valor del bit correspondiente en la memoria imagen de proceso de las salidas puede ocasionar, p. ej., el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "1".

La instrucción "Activar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Activar mapa de bits en el área de periferia" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El parámetro N devuelve el valor "0"
- El bit MCR tiene el estado lógico "1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben activarse.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero hacia el primer bit que debe activarse

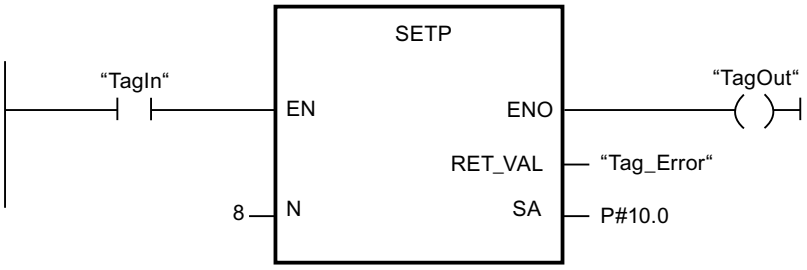
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción activa 8 bits en el área de periferia de acuerdo con las salidas Q10.0 a Q10.7.

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SETI: Activar área por bytes

Descripción

La instrucción "Activar área por bytes" establece en "1" el estado lógico de los bits en un área indicada de bytes. El punto inicial del área se define en el parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben activarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "Periferia" se accede en formato de byte, palabra

y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte. Por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas se pone a "0".

La llamada de la instrucción "Activar área por bytes" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa.
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

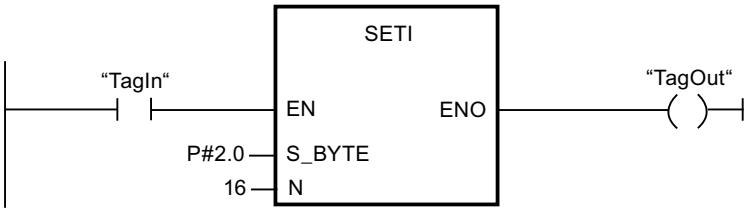
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se activan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área que debe activarse El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción activa 2 bytes (16 bits) en el área de P2.0 a P3.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

RESET: Desactivar área por bits

Descripción

La instrucción "Desactivar área por bits" establece el estado lógico de los bits en "0" en un área indicada. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa, no se ejecuta la instrucción. El número de bits que deben desactivarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta sólo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

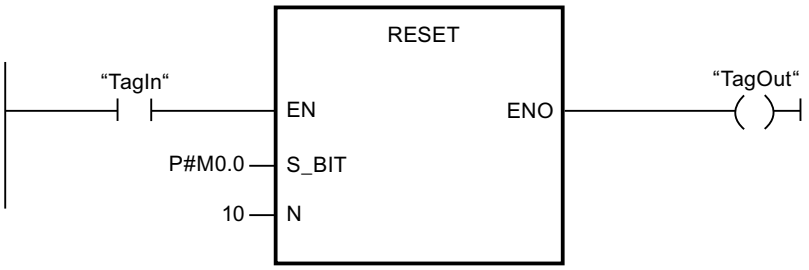
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
S_BIT	Input	POINTER	I, Q, M, D	Señala al primer bit del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben desactivarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción restablece 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

RESETP: Desactivar mapa de bits en el área de periferia

Descripción

La instrucción "Desactivar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "0". Simultáneamente con los bits de periferia se desactivan también los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe desactivarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no hay periferia enchufada para una parte del mapa de bits seleccionado, la instrucción intentará desactivar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se inicializa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas.

  ⚠ ADVERTENCIA

  El ajuste de los bits al valor de los bits correspondientes en la memoria imagen de proceso de las salidas puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "0".

La instrucción "Desactivar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Desactivar mapa de bits en el área de periferia" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El parámetro N devuelve el valor "0"
- El bit MCR tiene el estado lógico "1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben desactivarse.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero al primer bit que debe desactivarse.

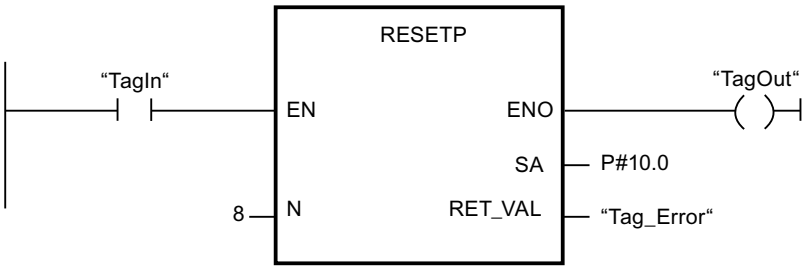
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción inicializa 8 bits en el área de periferia de acuerdo con las salidas Q10.0 a Q10.7.

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RESETI: Desactivar área por bytes

Descripción

La instrucción "Desactivar área por bytes" restablece el estado lógico de los bytes en "0" en un área indicada. El punto inicial del área se define mediante el puntero del parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben desactivarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa. Puesto que al área de memoria "Periferia" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte. Por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas se pone a "0".

La llamada de la instrucción "Desactivar área por bytes" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa.
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

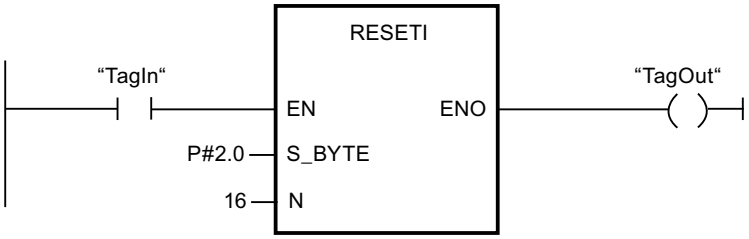
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área cuyos bits se desactivan. El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción desactiva 2 bytes (16 bits) en el área de P2.0 a P3.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

REPL_VAL: Introducir valor sustitutivo

Descripción

La instrucción "Introducir valor sustitutivo" permite introducir un valor sustitutivo en el acumulador 1 (ACU 1) desde un bloque de organización de error síncrono.

Nota

La instrucción "Introducir valor sustitutivo" sólo se puede llamar desde bloques de organización de error síncrono (OB 121, OB 122).

La instrucción "Introducir valor sustitutivo" se utiliza si ya no es posible leer ningún valor de un módulo. Tras cada acceso a dicho módulo se inicia el OB 122. Llamando la instrucción "Introducir valor sustitutivo" es posible cargar un valor sustitutivo en el acumulador 1. La ejecución del programa se prosigue entonces con este valor sustitutivo. La información necesaria para seleccionar el valor sustitutivo puede consultarse en las variables locales del OB 122.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Introducir valor sustitutivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
VAL	Input	DWORD	I, Q, M, D, L o constante	Valor sustitutivo
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#....)	Descripción
0000	Ningún error
8080	La instrucción no se ha llamado desde un OB de error síncrono.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite ocupar los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de éste es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento

y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento (i); el estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	I, Q, M, D, L, P	Paso predeterminado del circuito secuencial sucesivo
DSC	Static	BYTE	I, Q, M, D, L, P	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P	Valor de contaje actual del circuito secuencial sucesivo
DTBP	Static	WORD	I, Q, M, D, L, P	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	Hora de sistema anterior

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM_X: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite materializar un circuito secuencial sucesivo con 16 pasos como máximo. El número del primer paso se define con el parámetro DSP. El número del último paso se predetermina con el parámetro LST_STEP.

En cada paso se escriben todos los 16 bits de salida OUT0 a OUT15 y el parámetro de salida OUT_WORD (que contiene los bits de salida). A un bit de salida se asigna bien sea el bit correspondiente del campo predefinido OUT_VAL, o bien el bit de salida del paso anterior. El valor que se asigne dependerá de la ocupación de los bits de máscara en el parámetro S_MASK.

Nota

El ajuste predeterminado de los bits de máscara es 0. Para modificar el ajuste actual de uno o varios bits de máscara, realice el cambio en el bloque de datos de instancia.

La instrucción "Materializar circuito secuencial sucesivo" pasa al siguiente paso cuando se produce un flanco ascendente en el parámetro JOG. Si la instrucción ya se encuentra en el último paso y se produce un flanco ascendente en el parámetro JOG, se activan las variables Q y EOD. En este caso, el parámetro DCC se pone al valor 0. La instrucción permanece en el último paso hasta que el parámetro RESET se pone al valor "1".

Además, también se puede permitir la transición al paso siguiente en función del tiempo. Para ello se debe poner al valor "1" el parámetro DRUM_EN. La transición al paso siguiente tiene lugar si está activado el bit de evento (EVENTi) correspondiente al paso actual y si ha transcurrido el tiempo predeterminado para el paso actual. El tiempo necesario para un paso resulta de la multiplicación de la base de tiempo (DTBP por el factor de tiempo válido para el paso actual (S_PRESET).

Nota

El tiempo de ejecución restante en el paso actual (DCC) se reduce únicamente si está activado el bit de evento correspondiente (EVENTi).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si, al llamar la instrucción, el parámetro RESET tiene el valor "1", el circuito secuencial sucesivo pasará al paso cuyo número se haya asignado al parámetro DSP.

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1".

El circuito secuencial sucesivo también se puede conmutar en progresión en todo momento a través del valor del parámetro JOG.

En la primera llamada de la instrucción, el parámetro RESET se debe poner a 1.

Si el circuito secuencial se encuentra en el último paso y ha transcurrido el tiempo de ejecución predefinido para este paso, se activan los parámetros Q y EOD. La instrucción permanece en el último paso hasta que el parámetro RESET adopte el valor "1".

Esta instrucción se ejecuta sólo en los estados operativos ARRANQUE y RUN.

El sistema operativo inicializa la instrucción "Materializar circuito secuencial sucesivo" cuando se efectúa un arranque en frío. La instrucción no se inicializa en caso de un rearranque completo (en caliente). Si la instrucción "Materializar circuito secuencial sucesivo" debe inicializarse tras un rearranque completo (en caliente), es preciso llamarla en el OB 100 con el valor "1" en el parámetro RESET.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" desactiva el circuito secuencial. En la primera llamada de la instrucción, el parámetro RESET se debe poner a "1".
JOG	Input	BOOL	I, Q, M, D, L	Un flanco de señal ascendente (con respecto a la última llamada) provoca la transición al paso siguiente, siempre que el circuito secuencial no se encuentre aún en el último paso. La transición tiene lugar independientemente del valor asignado al parámetro DRUM_EN.
DRUM_EN	Input	BOOL	I, Q, M, D, L	Parámetro de control que determina si se puede realizar una transición al paso siguiente en función del tiempo. Si el valor es "1", es posible una transición en función del tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso Los valores admisibles son 1 a 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento n.º i (pertenece al paso i)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUTj, 0 ≤ j ≤ 15	Output	BOOL	I, Q, M, D, L	Bit de salida n.º j (idéntico al bit n.º j de OUT_WORD)
Q	Output	BOOL	I, Q, M, D, L	Parámetro de estado que indica si ha transcurrido el tiempo predefinido por el usuario para la ejecución del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Bits de salida resumidos en una variable
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Irrelevante para el usuario (parámetro de entrada JOG de la llamada anterior)
EOD	Static	BOOL	I, Q, M, D, L	Idéntico al parámetro de salida Q
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Número del primer paso Los valores admisibles son 1 a 16.
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Número del paso actual
DCC	Static	DWORD	I, Q, M, D, L, P	Tiempo de ejecución restante del paso actual, indicado en milisegundos (ms) (relevante sólo si DRUM_EN = 1 y el bit de evento correspondiente = 1)
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo en milisegundos (ms) válida para todos los pasos
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	(Irrelevante para el usuario: hora de sistema de la llamada anterior)
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Campo unidimensional con los factores de tiempo para cada paso. Selección recomendada de índices: [1 a 16]. En este caso, S_PRESET [x] contiene el factor de tiempo del paso x.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los valores que deben emitirse en cada paso, a no ser que se hayan inhibido vía S_MASK. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, OUT_VAL [x, y] contiene el valor asignado al bit de salida OUTy en el paso x.
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los bits de máscara para cada paso. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, S_MASK [x, y] contiene el bit de máscara para el valor "y"-ésimo que debe emitirse en el paso x. Significado de los bits de máscara: • 0: el valor correspondiente del paso anterior se asigna al respectivo bit de salida. • 1: el valor correspondiente de OUT_VAL se asigna al respectivo bit de salida.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE* (W#16#...)	Explicación
0000	Ningún error
8081	Valor no admisible en el parámetro LST_STEP
8082	Valor no admisible en DSC
8083	Valor no admisible en DSP
8084	El producto DCC = DTBP * S_PRESET[DSC] excede el valor 2**31-1 (aprox. 24,86 días).
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET, OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1".
  - Los parámetros ET, OA y CA se ponen a "0".
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET, OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
CMD	Input	BOOL	I, Q, M, D, L	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
CMD_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

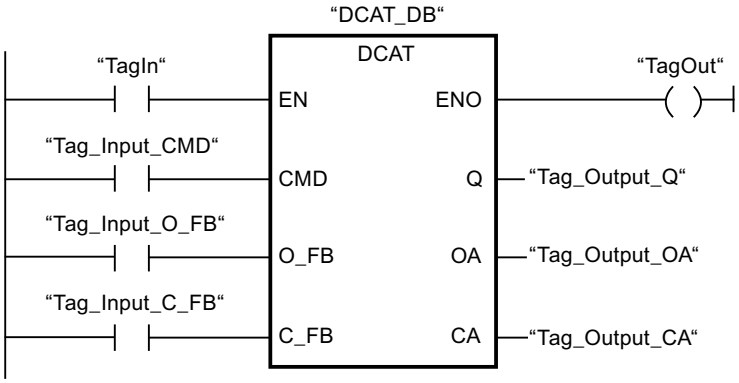
Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a

"0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se activa la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

La instrucción "Alarma de tiempo para control del motor" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
O_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error.
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
O_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Abrir"
C_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Cerrar"

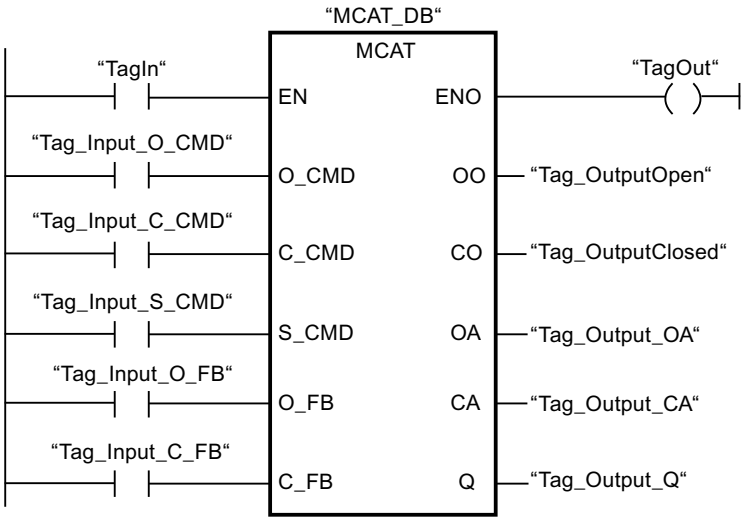
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22

Parámetro	Dirección	Valor
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una coincidencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY OF BOOL	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LAST	Static	BYTE	I, Q, M, D, L, P	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY of WORD	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor para la amplificación en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros

LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo dinámico. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse para suprimir perturbaciones o como filtro pasa bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa. En el parámetro IN también se pueden indicar constantes.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo En el parámetro SAMPLE_T también se pueden indicar constantes.
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de retardo de fase en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

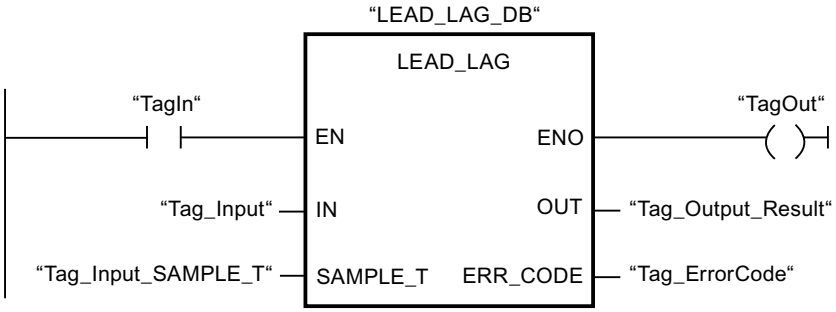
Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Los siguientes valores están almacenados en el bloque de datos de instancia "LEAD_LAG_DB" de la instrucción:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos de instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

TONR_X: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" permite acumular valores de tiempo dentro de un periodo definido por el parámetro PT. Si el estado lógico del parámetro TMR_EN cambia a "1", se ejecuta la instrucción y se empieza a contar el tiempo PT. La instrucción "Acumulador de tiempo" suma el valor del parámetro DELTA_T al del parámetro ET siempre que se cumplan las siguientes condiciones:

- El parámetro TMR_EN devuelve el estado lógico "1".
- El valor de tiempo del parámetro ET es menor que el del parámetro PT.
- El parámetro RESET devuelve el estado lógico "0".

Si el valor del parámetro TMR_EN cambia a "0", el tiempo del parámetro DELTA_T se dejará de sumar al valor de tiempo del parámetro ET.

Cuando el valor de tiempo del parámetro ET alcanza el del parámetro PT, el parámetro Q devuelve el estado lógico "1". Mientras el parámetro Q devuelva el estado lógico "1", el valor del parámetro ET no se modificará. El valor del parámetro ET se pone a "0" cuando el parámero RESET se pone a "1".

Puesto que la instrucción "Acumulador de tiempo" utiliza el tiempo de ejecución (DELTA_T) del último ciclo del bloque de organización (OB), la instrucción solo puede llamarse en bloques de organización cíclicos.

Nota

El tiempo de ciclo del bloque de organización debe desplazarse desde la información de arranque a la variable global del parámetro DELTA_T.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TMR_EN	Input	BOOL	I, Q, M, D, L	Activa el temporizador para acumular el valor de tiempo.
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
PV	Input	DINT	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo
DELTA_T	Input	INT	I, Q, M, D, L o constante	Tiempo de ciclo del OB del ciclo anterior
Q	Output	BOOL	Q, M, D, L,	Salida que se activa una vez transcurrido el tiempo PT.
ET	InOut	DINT	I, Q, M, D, L	Valor de tiempo actual del tiempo transcurrido

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

WSR: Guardar datos en el registro de desplazamiento

Descripción

La instrucción "Guardar datos en el registro de desplazamiento" permite escribir datos en un registro de desplazamiento desde el origen indicado. Todo elemento del registro se desplaza a la siguiente dirección. Los datos contenidos en la última dirección del registro de desplazamiento se pierden tras la operación de desplazamiento.

Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen (S_DATA) y se desplazan a la dirección inicial (START) del registro de desplazamiento. El parámetro LENGTH indica el número de elementos que deben desplazarse. El tipo de datos de los elementos se define en el parámetro E_TYPE. Si se indica un tipo de datos no admisible en el parámetro E_TYPE, no se procesa la instrucción.

Si el parámetro RESET está a "1" al ejecutarse la instrucción, se borra el registro. El parámetro Q se pone al estado lógico "1" si el registro de desplazamiento está ocupado con ceros o si se borra.

Parámetros

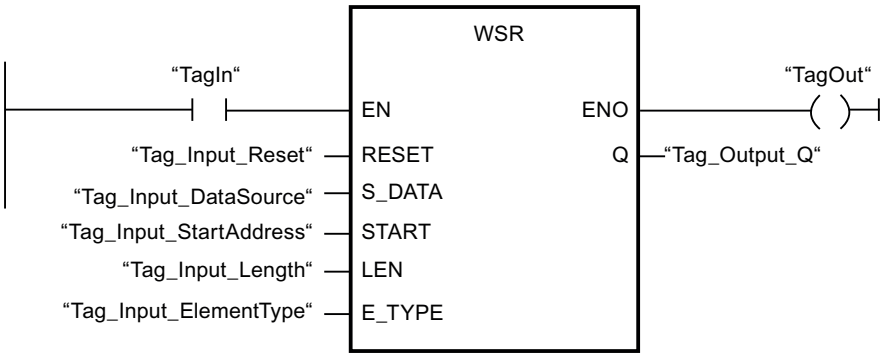
La tabla siguiente muestra los parámetros de la instrucción "Guardar datos en el registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_DATA	Input	POINTER	I, Q, M, D	Apunta al elemento de origen que se desplaza a la dirección inicial (START). En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
START	Input	POINTER	I, Q, M, D	Apunta a la dirección inicial del registro. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
LEN	Input	WORD	I, Q, M, D, L, P	Número de elementos que deben desplazarse.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de los elementos que deben desplazarse. Los siguientes tipos de datos son admisibles: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	Q, M, D, L	Indica "0" si el parámetro RESET está activado (1) o si todos los elementos que deben desplazarse tienen el valor "0".

Ejemplo

En el ejemplo siguiente se desplazan cuatro elementos del tipo de datos WORD a partir de la dirección indicada en el parámetro START. Una vez desplazado el primer elemento del registro a la siguiente dirección, la primera dirección se ocupará con los datos a los que señala el puntero en el parámetro S_DATA. El valor del último elemento del registro se pierde entonces.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

SHRB: Desplazar bit al registro de desplazamiento

Descripción

La instrucción "Desplazar bit al registro de desplazamiento" permite escribir un bit en un registro de desplazamiento desde el origen indicado (DATA). Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen y se desplazan a la dirección inicial (S_BIT) del registro de desplazamiento. Los restantes bits se desplazan una posición. El bit de la última dirección (S_BIT + N) se pierde tras el desplazamiento. Si el parámetro RESET está a "1" mientras se ejecuta la instrucción, las direcciones del registro de desplazamiento se ponen a "0" y no se desplazan.

Parámetros

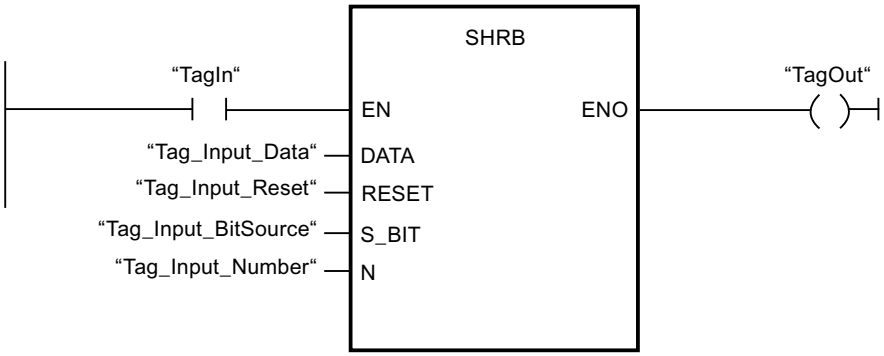
La tabla siguiente muestra los parámetros de la instrucción "Desplazar bit al registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
DATA	Input	BOOL	I, Q, M, D, L	Bit de origen
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del registro de desplazamiento. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	WORD	I, Q, M, D, L, P	Número de bits que deben desplazarse.

Ejemplo

En el ejemplo siguiente, el parámetro N se pone al valor 14 ("E" en notación hexadecimal), con lo que 14 bits se desplazan a partir de la dirección indicada en el parámetro S_BIT. Una

vez desplazados los bits, la primera dirección se ocupa con los datos del origen del parámetro DATA. El último valor de bit se pierde.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Antes de ejecutarse la instrucción, los siguientes estados lógicos están almacenados en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Después de la ejecución

Una vez ejecutada la instrucción, los siguientes estados lógicos se almacenan en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

SEG: Generar patrón de bits para el display de 7 segmentos

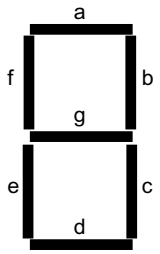
Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" permite convertir cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se devuelve en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6

Display de siete segmentos



0111	00000111	7
1000	01111111	8
1001	01100111	9
1010	01110111	A
1011	01111100	B
1100	00111001	C
1101	01011110	D
1110	01111001	E
1111	01110001	F

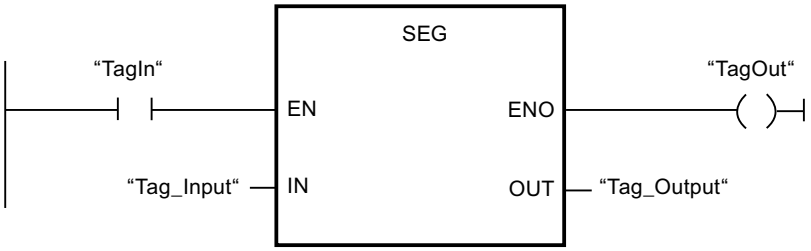
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits para el display de 7 segmentos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

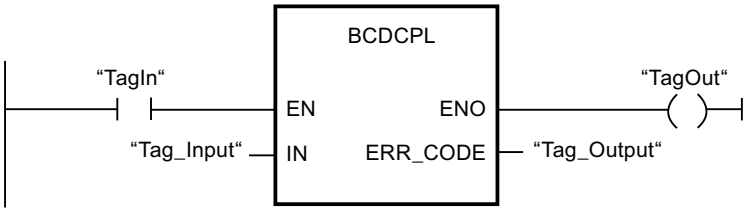
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" permite contar el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

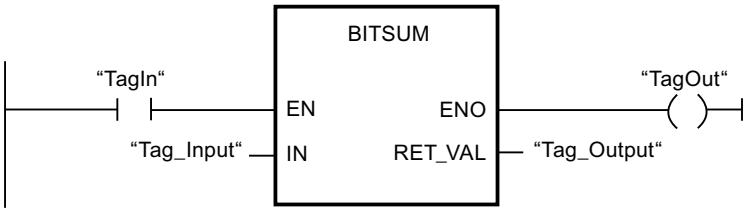
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos KOP (Página 2678)

Editar elementos KOP (Página 2694)

Cablear operandos en instrucciones KOP (Página 2701)

Cambiar formatos de visualización en el estado del programa (Página 3100)

FUP

Operaciones lógicas con bits

&: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" permite consultar los estados lógicos de dos o más operandos indicados y evaluarlos según la tabla de verdad Y.

Si el estado lógico de todos los operandos es "1", entonces se cumple la condición y la instrucción da como resultado "1". Si el estado lógico de un operando es "0", no se cumple la condición y la instrucción da como resultado "0".

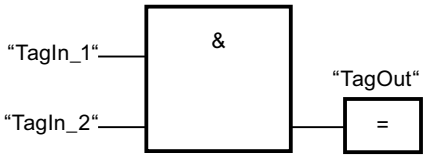
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", y se desactiva cuando el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de control de la temperatura ambiente (Página 3258)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3253)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3252)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Tabla de verdad Y (Página 5821)

Insertar entrada (Página 5824)

Tabla de verdad Y

Resultados lógicos

La tabla siguiente muestra los resultados que se obtienen combinando dos operandos mediante una Y lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Resultado lógico
1	1	1
0	1	0
1	0	0
0	0	0

Consulte también

&: Operación lógica Y (Página 5819)

>=1: Operación lógica O

Descripción

La instrucción "Operación lógica O" permite consultar los estados lógicos de dos o más operandos indicados y evaluarlos según la tabla de verdad O.

Si el estado lógico de uno de los operandos es "1", entonces se cumple la condición y la instrucción da como resultado "1". Si el estado lógico de todos los operandos es "0", no se cumple la condición y la instrucción da como resultado "0".

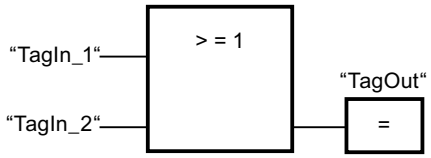
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa si el estado lógico del operando "TagIn_1" o del operando "TagIn_2" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Tabla de verdad O (Página 5822)

Insertar entrada (Página 5824)

Tabla de verdad O

Resultados lógicos

La tabla siguiente muestra los resultados que se obtienen combinando dos operandos mediante una O lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Resultado lógico
1	0	1
0	1	1
1	1	1
0	0	0

Consulte también

>=1: Operación lógica O (Página 5821)

X: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" permite consultar el resultado de la consulta del estado lógico de acuerdo con la tabla de verdad O-exclusiva.

En una instrucción "Operación lógica O-exclusiva", el estado lógico es "1" si el estado lógico de uno de los dos operandos indicados es "1". Si se consultan más de dos operandos, el resultado lógico común es "1" si un número impar de los operandos consultados devuelve el resultado "1".

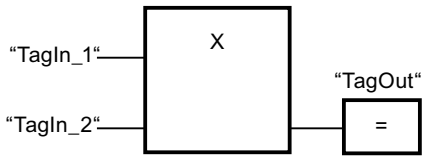
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa si el estado lógico de uno de los operandos "TagIn_1" y "TagIn_2" es "1". Si ambos operandos devuelven el estado lógico "1" ó "0", se desactiva la salida TagOut.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Tabla de verdad O-exclusiva (Página 5824)

Insertar entrada (Página 5824)

Tabla de verdad O-exclusiva

Resultados lógicos

La tabla siguiente muestra los resultados que se obtienen combinando dos operandos mediante una O-exclusiva lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Resultado lógico
1	0	1
0	1	1
1	1	0
0	0	0

La tabla siguiente muestra los resultados que se obtienen combinando tres operandos mediante una O-exclusiva lógica:

Estado lógico del primer operando	Estado lógico del segundo operando	Estado lógico del tercer operando	Resultado lógico
1	0	0	1
0	1	1	0
0	1	0	1
1	0	1	0
0	0	1	1
1	1	0	0
1	1	1	1
0	0	0	0

Consulte también

X: Operación lógica O-exclusiva (Página 5823)

Insertar entrada

Descripción

La instrucción "Insertar entrada" permite ampliar el cuadro de una de las instrucciones siguientes con una entrada:

- "Operación lógica Y"
- "Operación lógica O"
- "Operación lógica O-exclusiva"

Ampliando el cuadro de una instrucción se puede consultar el estado lógico de varios operandos.

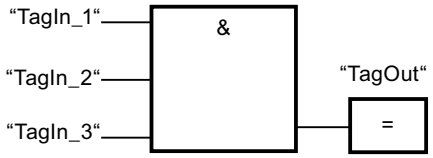
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Insertar entrada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	El operando indica el bit cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El cuadro de la instrucción "Operación lógica Y" ha sido ampliado con una entrada adicional, en la que se consulta el estado lógico del operando "TagIn_3". La salida "TagOut" se activa cuando los operandos "TagIn_1", "TagIn_2" y "TagIn_3" devuelven el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

&: Operación lógica Y (Página 5819)

>=1: Operación lógica O (Página 5821)

X: Operación lógica O-exclusiva (Página 5823)

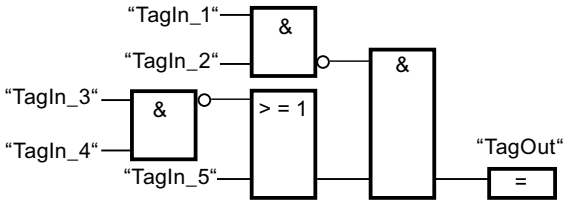
Invertir RLO

Descripción

La instrucción "Invertir RLO" invierte el estado lógico del resultado lógico (RLO).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- La entrada "TagIn_1" y/o "TagIn_2" devuelve el estado lógico "0".
- La entrada "TagIn_3" y/o "TagIn_4" devuelve el estado lógico "0" o la entrada "TagIn_5" devuelve el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Ejemplo de control de la temperatura ambiente (Página 3258)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3253)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3252)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

=: Asignación

Descripción

La instrucción "Asignación" permite activar el bit de un operando indicado. Si el resultado lógico (RLO) de la entrada del cuadro es el estado lógico "1", el operando indicado adopta el estado lógico "1". Si el estado lógico de la entrada del cuadro es "0", el bit del operando indicado se pone a "0".

La instrucción no afecta al RLO. El RLO de la entrada del cuadro se asigna directamente a los operandos disponibles a través del cuadro de asignación.

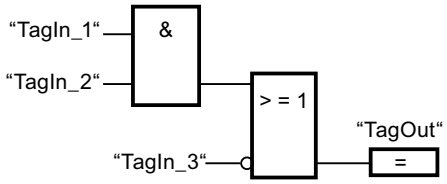
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" en la salida de la instrucción "Asignación" se activa cuando se cumple una de las condiciones siguientes:

- Las entradas "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico de la entrada "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de control de la temperatura ambiente (Página 3258)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3253)

El MCR (Master Control Relay) (Página 2564)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

MCR<: Abrir áreas MCR (Página 5985)

R: Desactivar salida

Descripción

La instrucción "Desactivar salida" permite poner a "0" el estado lógico de un operando indicado.

La instrucción se ejecuta sólo si el resultado lógico (RLO) de la entrada del cuadro es "1". Si el estado lógico de la entrada del cuadro es "1", el operando indicado se pone a "0". Si el RLO de la entrada del cuadro es "0", el estado lógico del operando indicado no cambia.

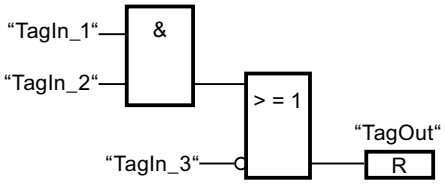
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, T, C, D, L	Operando que se desactiva si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se desactiva cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El estado lógico del operando "TagIn_3" es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3252)

El MCR (Master Control Relay) (Página 2564)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

MCR<: Abrir áreas MCR (Página 5985)

S: Activar salida

Descripción

La instrucción "Activar salida" permite poner a "1" el estado lógico de un operando indicado.

La instrucción se ejecuta sólo si el resultado lógico (RLO) de la entrada del cuadro es "1". Si el estado lógico de la entrada del cuadro es "1", el operando indicado se pone a "1". Si el RLO de la entrada del cuadro es "0", el estado lógico del operando indicado no cambia.

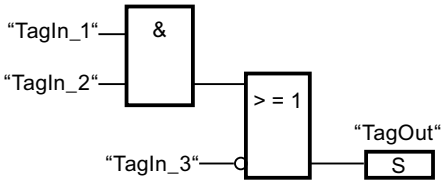
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar salida":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se activa si RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El operando "TagOut" se activa cuando se cumple una de las condiciones siguientes:

- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".
- El operando "TagIn_3" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3252)

El MCR (Master Control Relay) (Página 2564)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

MCR<: Abrir áreas MCR (Página 5985)

SR: Flipflop de activación/desactivación

Descripción

La instrucción "Flipflop de activación/desactivación" sirve para activar o desactivar el bit de un operando indicado en función del estado lógico de las entradas S y R1. Si el estado lógico de la entrada S es "1" y el de la entrada R1 es "0", el operando indicado se pone a "1". Si el estado lógico de la entrada S es "0" y el de la entrada R1 es "1", el operando indicado se pone a "0".

La entrada R1 predomina sobre la entrada S. Si el estado lógico de las entradas S y R1 es "1", el estado lógico del operando indicado se pone a "0".

Si el estado lógico de ambas entradas S y R1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

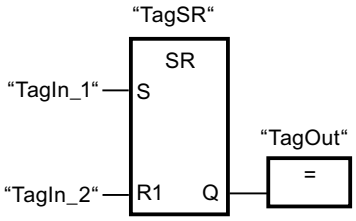
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de activación/desactivación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S	Input	BOOL	I, Q, M, D, L, T, C	Activación habilitada
R1	Input	BOOL	I, Q, M, D, L, T, C	Desactivación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	Operando que se activa o desactiva
Q	Output	BOOL	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagSR" y "TagOut" se activan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagSR" y "TagOut" se desactivan cuando se cumple una de las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

El MCR (Master Control Relay) (Página 2564)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

MCR<: Abrir áreas MCR (Página 5985)

RS: Flipflop de desactivación/activación

Descripción

La instrucción "Flipflop de desactivación/activación" sirve para desactivar o activar el bit de un operando indicado en función del estado lógico de las entradas R y S1. Si el estado lógico de la entrada R es "1" y el de la entrada S1 es "0", el operando indicado se pone a "0". Si el estado lógico de la entrada R es "0" y el de la entrada S1 es "1", el operando indicado se pone a "1".

La entrada S1 predomina sobre la entrada R. Si el estado lógico de las entradas R y S1 es "1", el estado lógico del operando indicado se pone a "1".

Si el estado lógico de ambas entradas R y S1 es "0", no se ejecuta la instrucción. En este caso, no cambia el estado lógico del operando.

El estado lógico actual del operando se transfiere a la salida Q y se puede consultar allí.

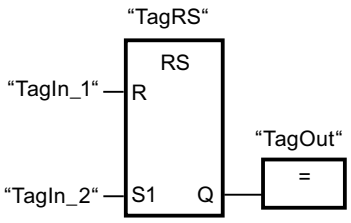
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Flipflop de desactivación/activación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
R	Input	BOOL	I, Q, M, T, C, D, L	Desactivación habilitada
S1	Input	BOOL	I, Q, M, T, C, D, L	Activación habilitada
<Operando>	InOut	BOOL	I, Q, M, D, L	Operando que se desactiva o activa
Q	Output	BOOL	I, Q, M, D, L	Estado lógico del operando

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Los operandos "TagRS" y "TagOut" se desactivan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- El operando "TagIn_2" devuelve el estado lógico "0".

Los operandos "TagRS" y "TagOut" se activan cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "0" y el operando "TagIn_2" devuelve el estado lógico "1".
- Los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

El MCR (Master Control Relay) (Página 2564)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

MCR<: Abrir áreas MCR (Página 5985)

P: Consultar flanco de señal ascendente de un operando

Descripción

La instrucción "Consultar flanco de señal ascendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "0" a "1". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del resultado lógico (RLO) de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

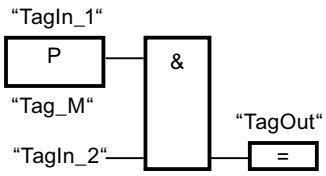
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	BOOL	I, Q, M, T, C, D, L	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- En la entrada "TagIn_1" hay un flanco de señal ascendente.
- El estado lógico del operando "TagIn_2" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3252)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

N: Consultar flanco de señal descendente de un operando

Descripción

La instrucción "Consultar flanco de señal descendente de un operando" permite detectar si el estado lógico de un operando indicado (<Operando1>) ha cambiado de "1" a "0". La instrucción compara el estado lógico actual del <Operando1> con el estado lógico de la consulta anterior, que está almacenado en una marca de flancos (<Operando2>). Si la instrucción detecta un cambio del resultado lógico (RLO) de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Indique el operando que debe consultarse (<Operando1>) en el comodín correspondiente ubicado encima de la instrucción. La marca de flancos (<Operando2>) se indica en el comodín de operandos situado debajo de la instrucción.

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

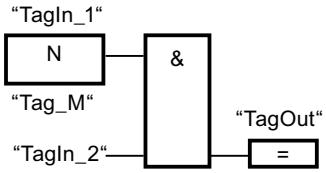
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente de un operando":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	BOOL	I, Q, M, T, C, D, L	Señal que se debe consultar
<Operando2>	InOut	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el estado lógico de la consulta anterior.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- En la entrada "TagIn_1" hay un flanco de señal descendente.
- El estado lógico del operando "TagIn_2" es "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

P_TRIG: Consultar flanco de señal ascendente del RLO

Descripción

La instrucción "Consultar flanco de señal ascendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "0" a "1". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco de señal ascendente.

Si se detecta un flanco de señal ascendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

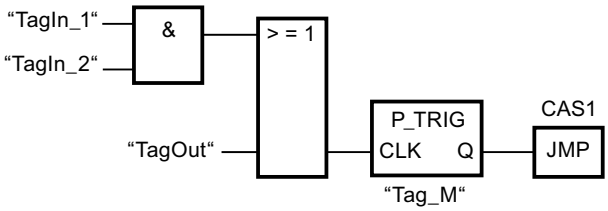
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "0" a "1", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

N_TRIG: Consultar flanco de señal descendente del RLO

Descripción

La instrucción "Consultar flanco de señal descendente del RLO" permite consultar un cambio del estado lógico del resultado lógico (RLO) de "1" a "0". La instrucción compara el estado lógico actual del RLO con el estado lógico de la consulta anterior, que está guardado en una

marca de flancos (<Operando>). Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco de señal descendente.

Si se detecta un flanco de señal descendente, la salida de la instrucción devuelve el estado lógico "1". En todos los demás casos, el estado lógico de la salida de la instrucción es "0".

Nota

La dirección de la marca de flancos no se puede utilizar más de una vez en el programa, puesto que, de lo contrario, se sobrescribe el bit de marcas. Esto influye negativamente en la evaluación de flancos, de manera que el resultado deja de ser unívoco. El área de memoria de la marca de flancos debe estar en un bloque de datos (área 'Static' de FB) o en el área de marcas.

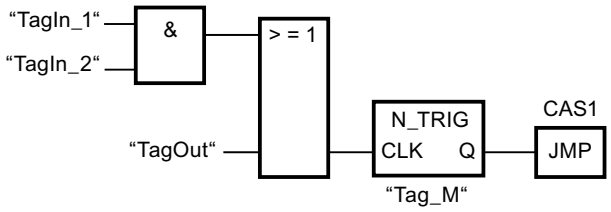
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CLK	Input	BOOL	I, Q, M, D, L	RLO actual
<Operando>	InOut	BOOL	M, D	Marca de flancos en la que se almacena el RLO de la consulta anterior.
Q	Output	BOOL	I, Q, M, D, L	Resultado de la evaluación de flancos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En la marca de flancos "Tag_M" se almacena el RLO de la consulta anterior. Cuando se detecta un cambio del estado lógico del RLO de "1" a "0", se ejecuta el salto a la etiqueta CAS1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Temporizadores

Temporizadores CEI

TP: Impulso

Descripción

La instrucción "Impulso" permite activar la salida Q por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. La salida Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico de la salida Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo PT y si el estado lógico de la entrada IN es "0", se desactiva la salida ET.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TP (p. ej., "TP_DB")
- Declaración como variable local del tipo TP en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Impulso" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque

completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Impulso", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Impulso" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

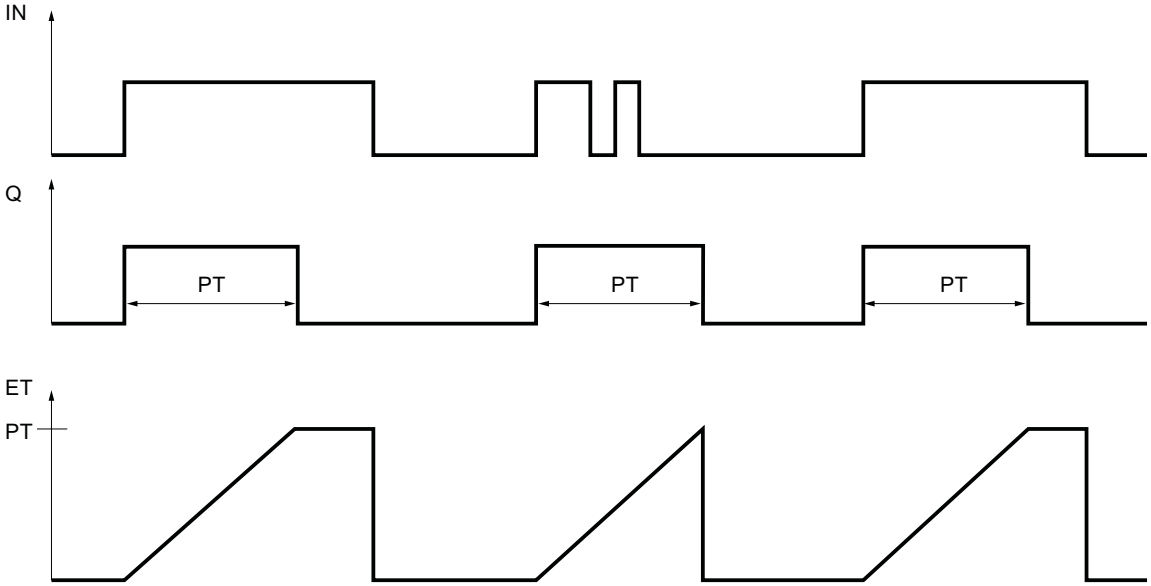
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Duración del impulso. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida de impulso
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de control de la temperatura ambiente (Página 3258)

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TON: Retardo al conectar

Descripción

Con la instrucción "Retardo al conectar" se puede retardar la activación de la salida Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. Una vez transcurrido el tiempo PT, la salida Q devuelve el estado lógico "1". La salida Q permanece activada mientras la entrada de arranque esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva la salida Q. La función de temporización se reinicia al detectarse un flanco de señal ascendente nuevo en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. La salida ET se desactiva en cuanto el estado lógico de la entrada IN cambia a "0".

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TON (p. ej., "TON_DB")
- Declaración como variable local del tipo TON en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al conectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al conectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Retardo al conectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

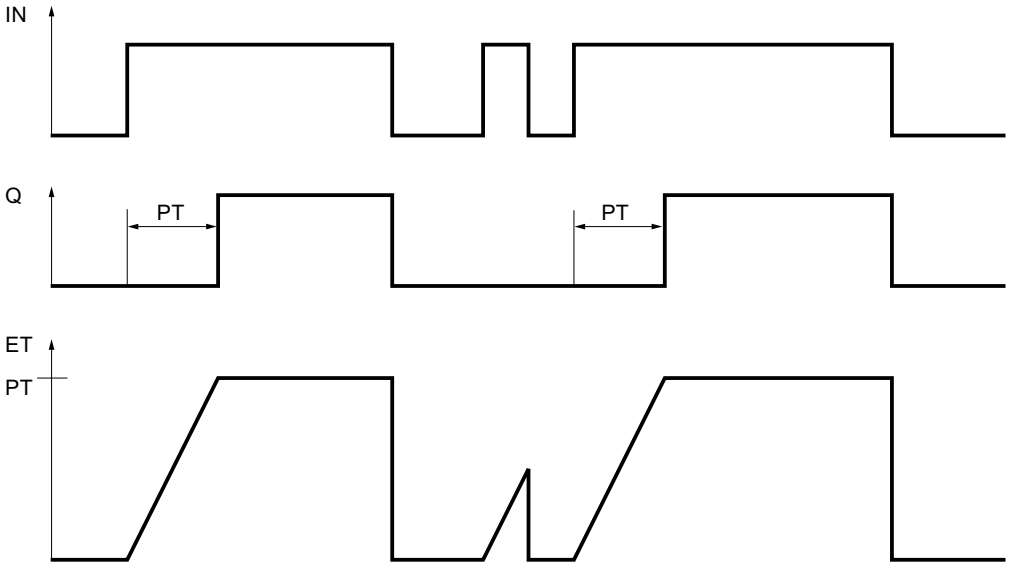
La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TOF: Retardo al desconectar

Descripción

Con la instrucción "Retardo al desconectar" se puede retardar la desactivación de la salida Q por el tiempo programado PT. La salida Q se activa cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico de la entrada IN cambia nuevamente a "0", el tiempo programado PT deja de contar. La salida Q permanece activada mientras transcurre el tiempo PT. Una vez transcurrido el tiempo PT se desactiva la salida Q. Si el estado lógico de la entrada IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico de la salida Q permanece en "1".

El valor de tiempo actual se puede consultar en la salida ET. El valor de tiempo actual empieza a partir de T#0s y termina al alcanzarse el valor de tiempo PT. Una vez transcurrido el tiempo PT, el valor actual de la salida ET se conserva hasta que la entrada IN cambie nuevamente

a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TOF (p. ej., "TOF_DB")
- Declaración como variable local del tipo TOF en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al desconectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al desconectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Retardo al desconectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

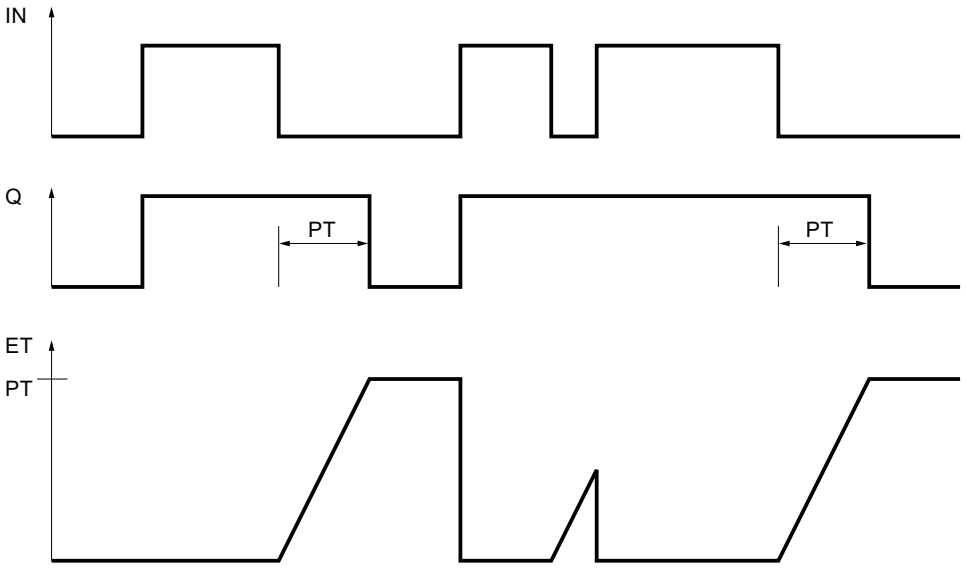
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	Salida que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Temporizadores SIMATIC

S_PULSE: Parametrizar y arrancar temporizador como impulso

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Si el estado lógico en la entrada S cambia a "0" antes de que transcurra el tiempo programado, se detiene el temporizador. En tal caso, el estado lógico de la salida Q es "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

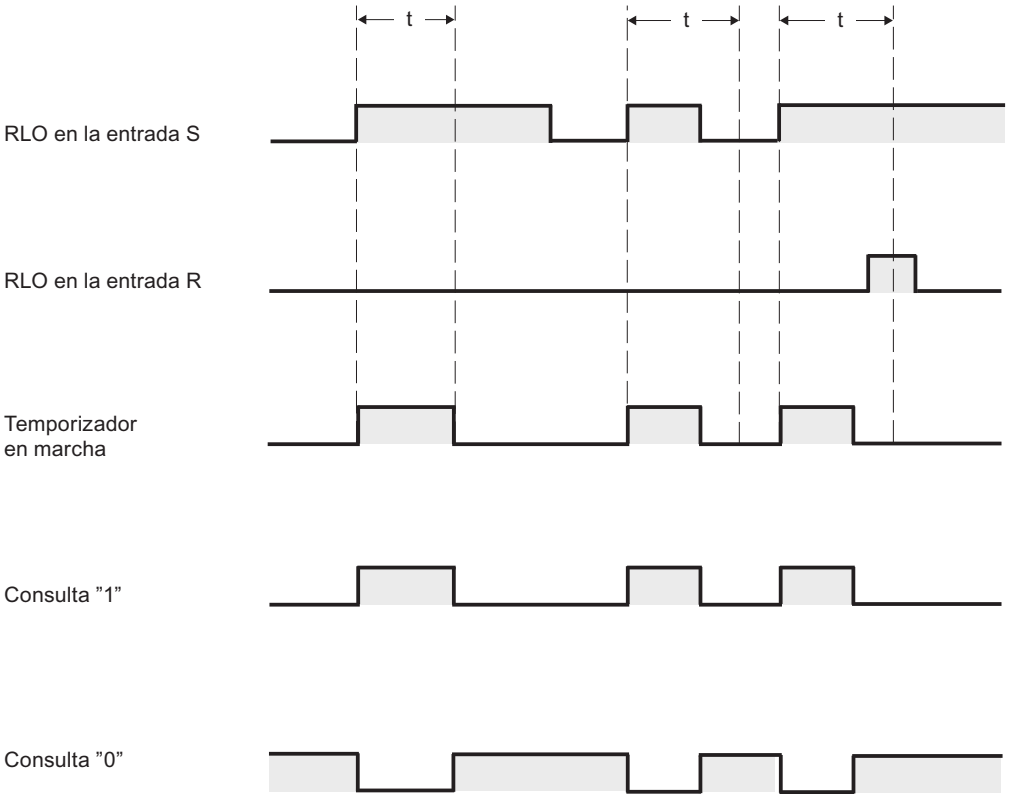
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

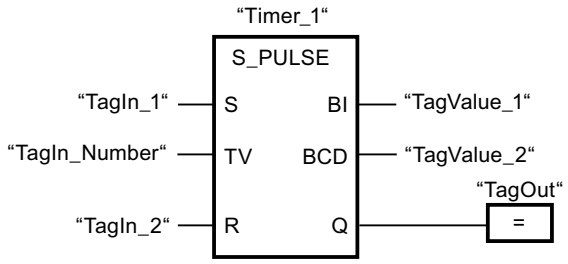
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador "Timer_1" se detiene. En ese caso, el operando "TagOut" se pone a "0".

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha y el operando "TagIn_1" tenga el estado lógico "1". Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como impulso prolongado" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Mientras se ejecuta el temporizador, la salida Q devuelve el estado lógico "1". Una vez transcurrido el tiempo, la salida Q se pone a "0". Si el estado lógico en la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, este se reinicia con el tiempo programado en la entrada TV.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

La instrucción "Parametrizar y arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

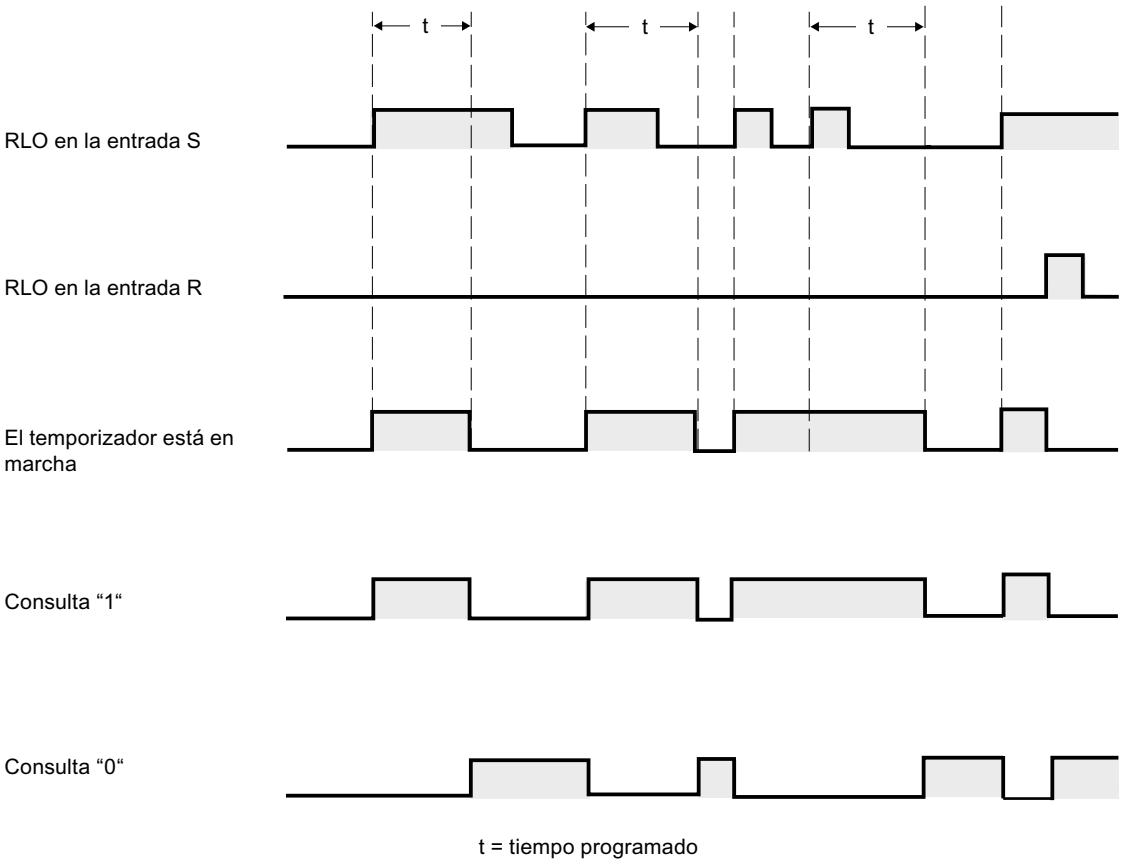
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

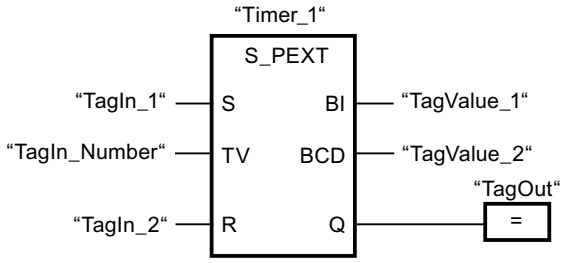
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en la entrada S. Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador esté en marcha. Una vez transcurrido el tiempo o si se inicializa el temporizador, el operando "TagOut" se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" se arranca un temporizador programado como retardo a la conexión al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV) mientras el estado lógico de la entrada S sea "1". Una vez transcurrido el tiempo debidamente y si la entrada S aún tiene el valor lógico "1", la salida Q devuelve el valor lógico "1". Si el estado de la entrada S cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En este caso, la salida Q adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

Si el temporizador está contando y el estado lógico de la entrada R cambia de "0" a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. En ese caso, el estado lógico de la salida Q es "0". El temporizador se reinicia cuando el estado lógico de la entrada R es "1", incluso si el temporizador no se ejecuta y el RLO en la entrada S es "1".

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

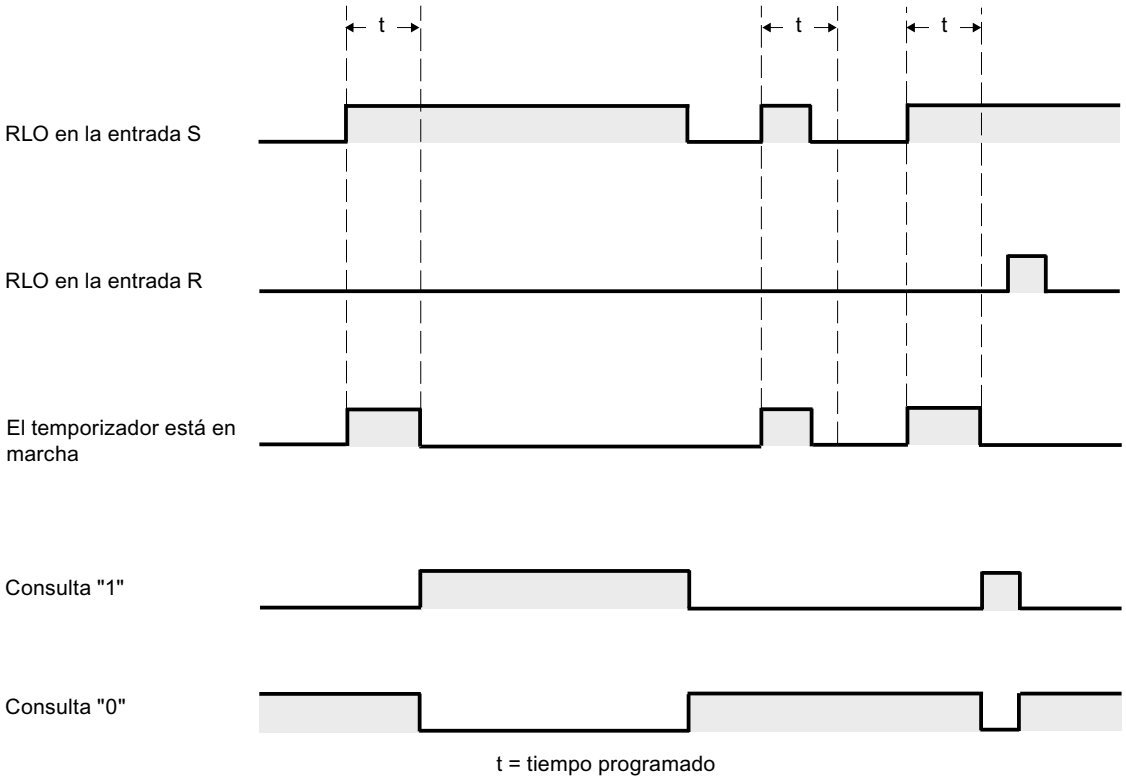
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

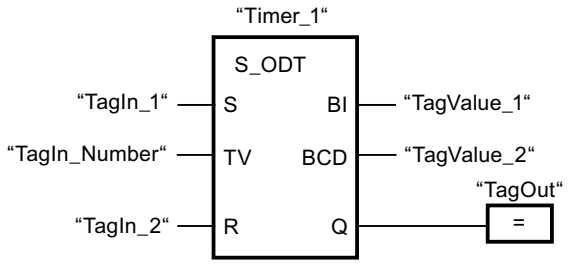
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el operando tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. El operando "TagOut" devuelve en ese caso el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada S. El temporizador cuenta el tiempo programado (TV), aunque el estado lógico de la entrada S cambie a "0". Una vez transcurrido el tiempo, la salida Q devuelve el estado lógico "1" independientemente del estado lógico de la entrada S. Si el estado lógico de la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, el temporizador vuelve a arrancar con el tiempo programado (TV).

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" de la entrada R pone a "0" el valor actual de tiempo y la base de tiempo, independientemente del estado lógico de la entrada de arranque S. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

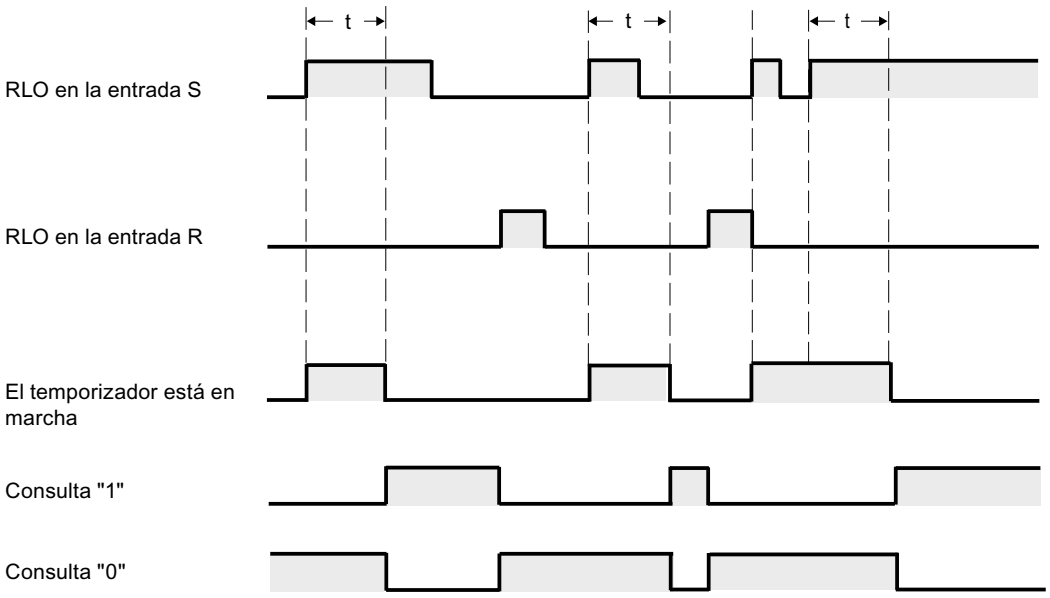
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

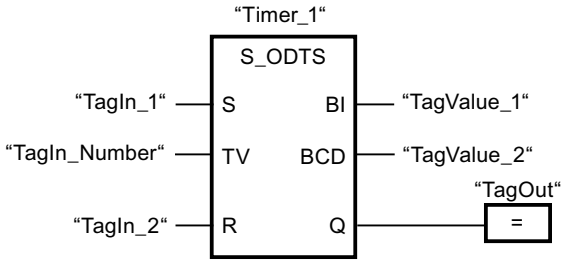
La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number" aunque el estado lógico del operando "TagIn_1" cambie a "0". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión

Descripción

Con la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" se arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente) en la entrada S. El temporizador empieza a contar con el tiempo programado (TV). Mientras el temporizador está en marcha o la entrada S devuelve el estado lógico "1", la salida Q tiene el estado lógico "1". Si el tiempo ha transcurrido y el estado lógico es "0", la salida Q adopta el valor lógico "0". Si el estado lógico de la entrada S cambia de "0" a "1" mientras se cuenta el tiempo, el temporizador se inicializa. El temporizador no vuelve a arrancar hasta que no se detecta un flanco de señal descendente en la entrada S.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor de tiempo actual se emite codificado en binario por la salida BI y codificado en BCD por la salida BCD.

El estado lógico "1" en la entrada R pone a "0" el valor de tiempo y la base de tiempo actuales. En ese caso, el estado lógico de la salida Q es "0".

La instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

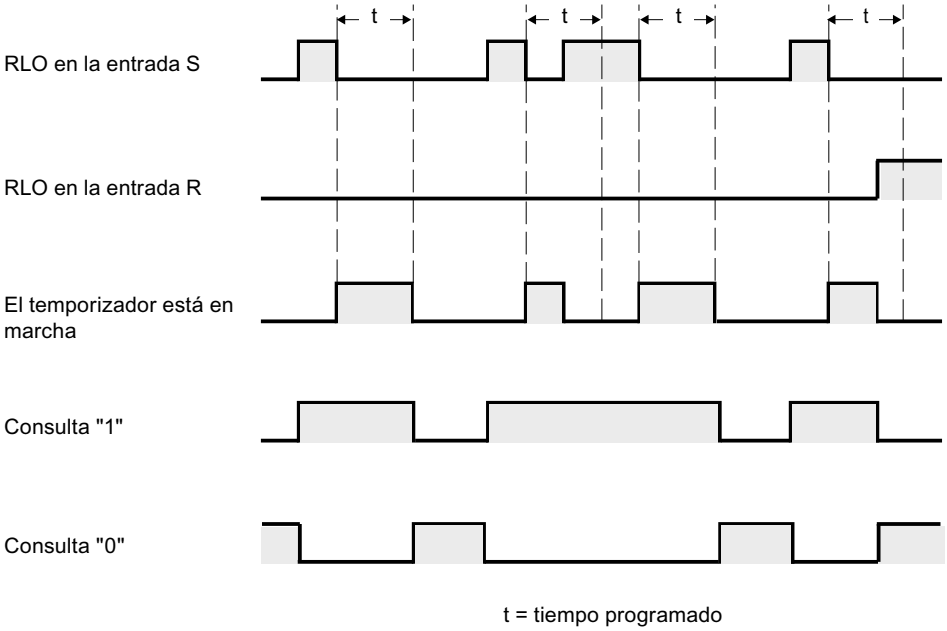
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	InOut/Input	TIMER	T	Temporizador de la instrucción El número de temporizadores depende de la CPU.
S	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
R	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de reset
BI	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en binario)
BCD	Output	WORD	I, Q, M, D, L, P	Valor de tiempo actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L, P	Estado del temporizador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

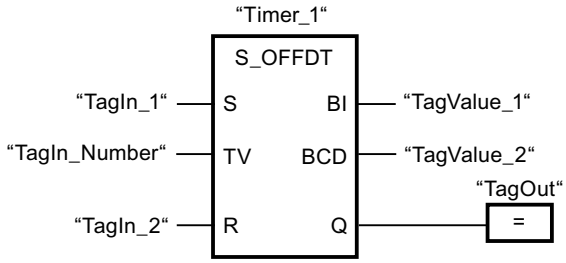
Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". El operando "TagOut" devuelve el estado lógico "1" mientras el temporizador está en marcha o el operando "TagIn_1" tiene el estado lógico "0". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

SP: Arrancar temporizador como impulso

Descripción

La instrucción "Arrancar temporizador como impulso" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador cuenta el tiempo indicado mientras el RLO devuelva el estado lógico "1". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" antes de que haya transcurrido el valor de tiempo, el temporizador se detiene. En ese caso, una consulta de si el estado del temporizador es "1" da como resultado "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

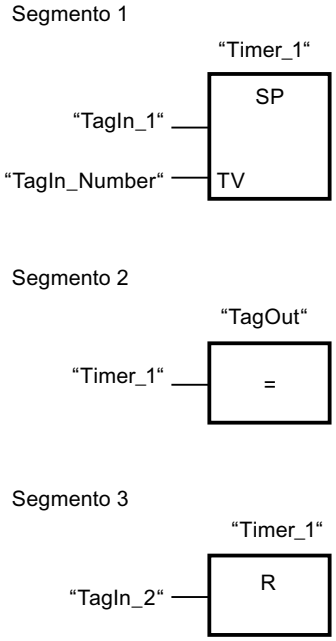
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" y sigue contando mientras el operando "TagIn_1" devuelva el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

SE: Arrancar temporizador como impulso prolongado

Descripción

La instrucción "Arrancar temporizador como impulso prolongado" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras el temporizador está funcionando, este se arranca de nuevo con el tiempo programado. La consulta sobre si el estado es "1" da como resultado "0" si el tiempo ha transcurrido.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como impulso prolongado" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

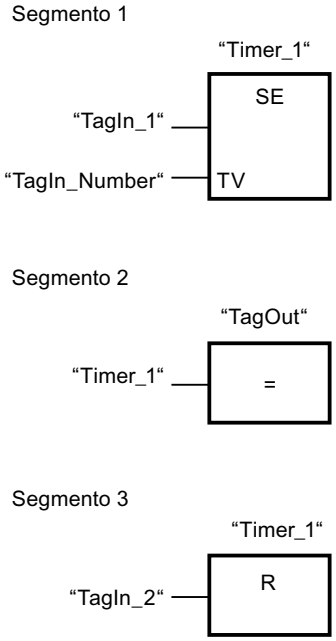
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso prolongado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "TagIn_Number" sin verse afectado por un flanco descendente en el RLO. Mientras el temporizador está en marcha, el operando "TagOut" devuelve el estado lógico "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" antes de que transcurra el tiempo, el temporizador se reinicia.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

SD: Arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador empieza a contar con el tiempo indicado y sigue contando mientras el RLO sea "1". Una vez transcurrido el tiempo y si el RLO todavía tiene el estado lógico "1", una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "1" a "0" mientras se cuenta el tiempo, el temporizador se detiene. En ese caso, la consulta de si el estado del temporizador es "1" devuelve el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

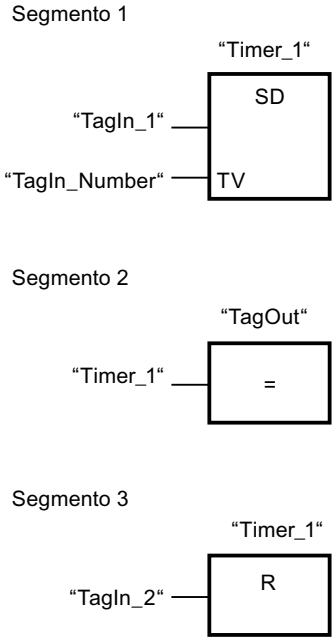
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo y si el RLO tiene el estado lógico "1", el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" antes de que transcurra el tiempo, el temporizador se detiene. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

SS: Arrancar temporizador como retardo a la conexión con memoria

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en la entrada de arranque. El temporizador cuenta el tiempo indicado, aunque el RLO cambie al estado lógico "0". Si ha transcurrido el tiempo, una consulta de si el estado del temporizador es "1" da como resultado "1". Una vez transcurrido el tiempo, el temporizador no puede volver a ponerse en marcha hasta que no se haya reiniciado explícitamente.

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

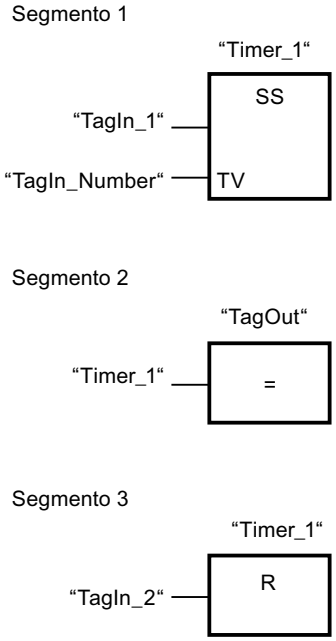
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión con memoria":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Una vez transcurrido el tiempo, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "0" a "1" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

SF: Arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Arrancar temporizador como retardo a la desconexión" arranca un temporizador programado al detectarse un cambio en el resultado lógico (RLO) de "1" a "0" (flanco de señal descendente) en la entrada de arranque. El temporizador empieza a contar con el tiempo indicado. Mientras el temporizador está en marcha, una consulta de si el estado del temporizador es "1" da como resultado "1". Si el RLO cambia de "0" a "1" mientras se ejecuta el temporizador, este se inicializa. El temporizador siempre arranca de nuevo cuando el RLO cambia de "1" a "0".

El tiempo se compone internamente de un valor de tiempo y de una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo.

La instrucción "Arrancar temporizador como retardo a la desconexión" requiere una operación lógica precedente para la evaluación de flancos y solo se puede colocar en el margen derecho del segmento.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Parámetros

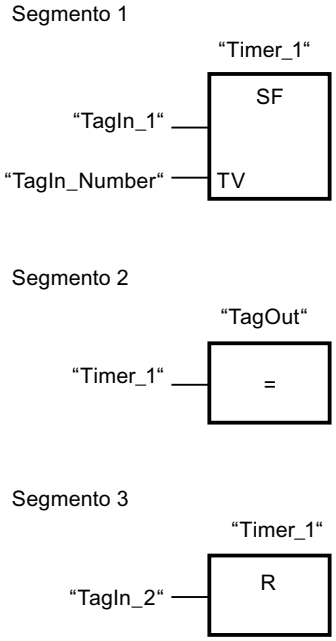
La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L, P	Entrada de arranque
TV	Input	S5TIME, WORD	I, Q, M, D, L o constante	Tiempo
<Temporizador>	Output	TIMER	T	Temporizador que arranca. El número de temporizadores depende de la CPU.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "1" a "0", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el valor de tiempo del operando "TagIn_Number". Mientras el temporizador está en marcha, el operando "TagOut" se pone a "1". Si el estado lógico del operando "TagIn_1" cambia de "1" a "0" mientras el temporizador está en marcha, se reinicia el temporizador. Si el estado lógico del operando "TagIn_2" es "1", el temporizador "Timer_1" se reinicia, es decir, que se detiene y el valor de tiempo actual se pone a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

L: Cargar valor del temporizador (Página 6189)

Contadores

Contadores CEI

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor actual de contaje de la salida CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos (INT). Cuando se alcanza el valor límite superior, el estado lógico de la entrada CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. El estado lógico de la salida Q es determinado por el parámetro PV. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0".

El valor de la salida CV se pone a cero cuando el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de la entrada CU no tendrá efecto alguno en la instrucción.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTU (p. ej., "CTU_DB")
- Declaración como variable local del tipo CTU en la sección "Static" de un bloque (p. ej., #MyCTU_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un

rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro R de la instrucción. Si un bloque diferente contiene instancias de la instrucción "Contador ascendente", estas pueden reiniciarse p. ej. inicializando el bloque de nivel superior.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Para poder ejecutar la instrucción "Contador ascendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

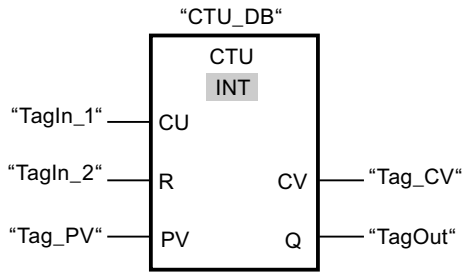
La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor actual de contaje del operando "Tag_CV" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos (INT = 32767).

El valor del parámetro PV se utiliza como límite para determinar la salida "TagOut". La salida "TagOut" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual

al valor del operando "Tag_PV". En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos (INT) indicado. Cuando se alcanza el valor límite inferior, el estado lógico de la entrada CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. Si el valor actual de contaje es menor o igual a cero, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0".

El valor de la salida CV se pone al valor del parámetro PV cuando el estado lógico de la entrada LD cambia a "1". Mientras la entrada LD tenga el estado lógico "1", el estado lógico de la entrada CD no tendrá efecto alguno en la instrucción.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTD (p. ej., "CTD_DB")
- Declaración como variable local del tipo CTD en la sección "Static" de un bloque (p. ej., #MyCTD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de

arranque con el valor "1" en el parámetro LD de la instrucción. En el parámetro PV se indica en este caso el valor inicial deseado para el parámetro CV.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Para poder ejecutar la instrucción "Contador descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetros

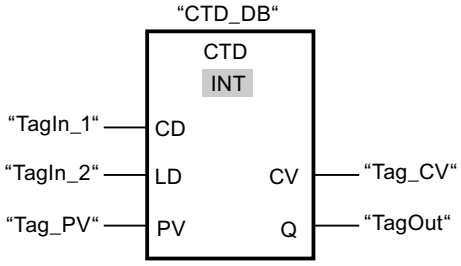
La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1", se ejecuta la instrucción "Contador descendente" y el valor de la salida "Tag_CV" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos (INT = -32768).

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "TagOut" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno y se deposita en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje de la salida CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor actual de contaje no se modifica en la salida CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado (INT). Una vez alcanzado el valor límite superior, el valor de contaje ya no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado (INT), ya no se decrementa el valor de contaje.

Si el estado lógico de la entrada LD cambia a "1", el valor de contaje de la salida CV adopta el valor del parámetro PV. Mientras la entrada LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", un cambio del estado lógico de las entradas CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en la salida QU. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida QU adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QU es "0". En el parámetro PV también se puede indicar una constante.

El estado del contador descendente se puede consultar en la salida QD. Si el valor actual de contaje es menor o igual a cero, la salida QD adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QD es "0".

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTUD (p. ej., "CTUD_DB")
- Declaración como variable local del tipo CTUD en la sección "Static" de un bloque (p. ej., #MyCTUD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en éste se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz

del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente - descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con los siguientes valores de parámetros:

- Si se utiliza como contador ascendente, el valor del parámetro R debe ponerse a "1".
- Si se utiliza como contador descendente, el valor del parámetro LD debe ponerse a "1". En el parámetro PV debe indicarse en este caso el valor inicial deseado del parámetro CV.

Si un bloque diferente contiene instancias de la instrucción "Contador ascendente - descendente", estas pueden reiniciarse p. ej. inicializando el bloque de nivel superior.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Para poder ejecutar la instrucción "Contador ascendente - descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Parámetro

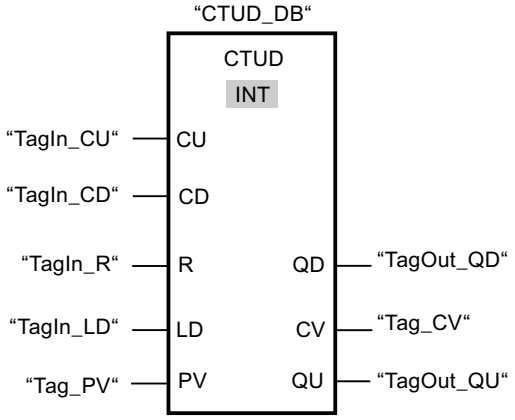
La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU.
QU	Output	BOOL	I, Q, M, D, L	Estado del contador ascendente
QD	Output	BOOL	I, Q, M, D, L	Estado del contador descendente
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_CU" o "TagIn_CD" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Contador ascendente - descendente". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_CU", el valor de contaje actual se incrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada "TagIn_CD", el valor de contaje se decrementa en uno y se deposita en la salida "Tag_CV". Si se detecta un flanco de señal ascendente en la entrada CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior (INT = 32767). Cuando se detecta un flanco de señal ascendente en la entrada CD, el valor de contaje se decrementa hasta alcanzar el valor límite inferior INT = -32768.

La salida "TagOut_QU" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor de la entrada "Tag_PV". En todos los demás casos, la salida "TagOut_QU" devuelve el estado lógico "0".

La salida "TagOut_QD" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "TagOut_QD" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Contadores SIMATIC

S_CUD: Parametrizar e incrementar/decrementar contador

Descripción

La instrucción "Parametrizar e incrementar/decrementar contador" permite incrementar y decrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje no se modifica.

El valor de contaje sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior "0", no se decrementa más el valor de contaje.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si se inicializa el contador y el RLO en las entradas CU y CD es "1", el contador solo cuenta una vez en el próximo ciclo aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU, CD y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar/decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

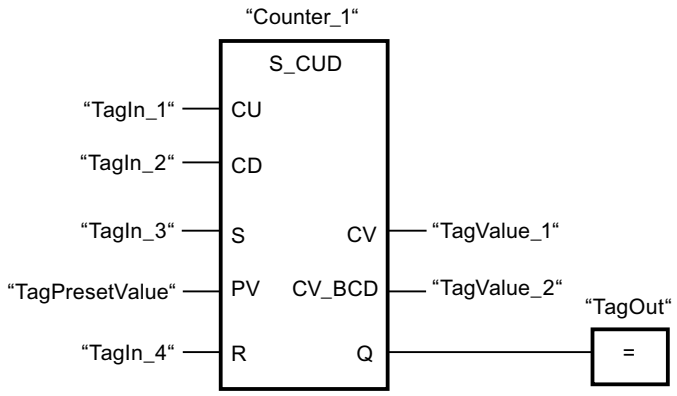
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar/ decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L, T, C	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L, T, C	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L, P	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de la entrada "TagIn_1" o "TagIn_2" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Parametrizar e incrementar/decrementar contador". Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_1" y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Cuando se detecta un flanco de señal ascendente en la entrada "TagIn_2" y el valor de contaje actual es mayor que "0", este valor se decrementará en uno.

Si el estado lógico de la entrada "TagIn_3" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_4" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

S_CU: Parametrizar e incrementar contador

Descripción

La instrucción "Parametrizar e incrementar contador" permite incrementar el valor de un contador. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue incrementando hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CU es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CU y S no tendrá ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar e incrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

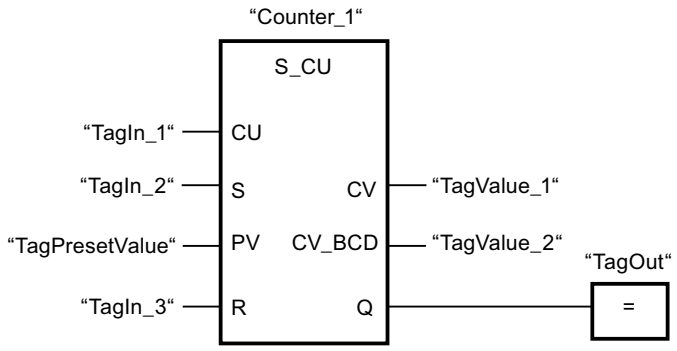
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar e incrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CU	Input	BOOL	I, Q, M, D, L, T, C	Entrada de contaje ascendente
S	Input	BOOL	I, Q, M, D, L, T, C	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L, P	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

S_CD: Parametrizar y decrementar contador

Descripción

La instrucción "Parametrizar y decrementar contador" permite decrementar el valor de un contador. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se decrementa en uno. El valor de contaje actual se emite en formato hexadecimal en la salida CV y codificado en BCD en la salida CV_BCD. El valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite inferior, el valor de contaje no se decrementa más al detectarse un flanco de señal ascendente.

Cuando el estado lógico de la entrada S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el RLO de la entrada CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de las entradas CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico en la salida Q es "1" si el valor de contaje es superior a cero. Si el valor de contaje es igual a cero, la salida Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

La instrucción "Parametrizar y decrementar contador" requiere una operación lógica precedente para la evaluación de flancos y se puede colocar dentro o en el extremo del segmento.

Parámetros

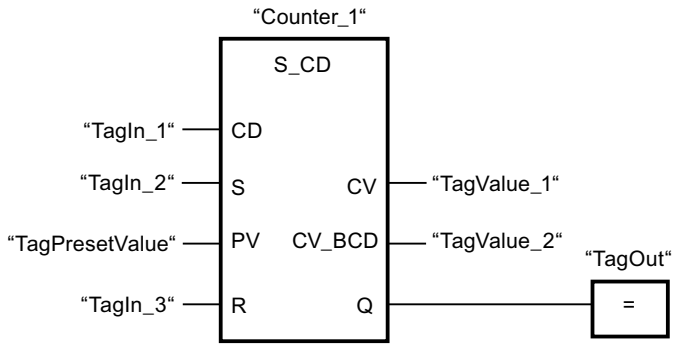
La tabla siguiente muestra los parámetros de la instrucción "Parametrizar y decrementar contador":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Contadores>	InOut	COUNTER	C	Contador de la instrucción El número de contadores depende de la CPU.
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje descendente
S	Input	BOOL	I, Q, M, D, L, T, C	Entrada para preajustar el contador
PV	Input	WORD	I, Q, M, D, L, P	Valor de contaje predeterminado (C#0 a C#999)
R	Input	BOOL	I, Q, M, D, L, T, C	Entrada de reset
CV	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (hexadecimal)
CV_BCD	Output	WORD, S5TIME, DATE	I, Q, M, D, L, P	Valor de contaje actual (codificado en BCD)
Q	Output	BOOL	I, Q, M, D, L	Estado del contador

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico de la entrada "TagIn_1" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es mayor que "0", este valor se decrementará en uno. Si el estado lógico de la entrada "TagIn_2" cambia de "0" a "1", el valor de contaje adopta el valor del operando "TagPresetValue". El valor de contaje se pone a "0" si el operando "TagIn_3" tiene el estado lógico "1".

El valor de contaje actual se memoriza en formato hexadecimal en el operando "TagValue_1" y codificado en BCD en el operando "TagValue_2".

La salida "TagOut" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SC: Poner contador al valor inicial

Descripción

Con la instrucción "Poner contador al valor inicial" se puede ajustar el valor de un contador. La instrucción se ejecuta cuando el resultado lógico (RLO) de la entrada de arranque de la instrucción cambia de "0" a "1". Si se ejecuta la instrucción, el contador se pone al valor de contaje indicado.

La instrucción "Poner contador al valor inicial" requiere una operación lógica precedente para la evaluación de flancos y sólo se puede colocar en el margen derecho del segmento.

Parámetros

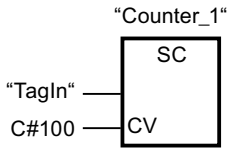
La tabla siguiente muestra los parámetros de la instrucción "Poner contador al valor inicial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	Entrada de arranque
PV	Input	WORD	I, Q, M, D, L o constante	Valor con el que se preajusta el contador en formato BCD. (C#0 a C#999)
<Contador>	InOut/Input	COUNTER	C	Contador que se preajusta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el estado lógico del operando "TagIn" cambia de "0" a "1", el contador "Counter_1" se pone a "100".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

CU: Contador ascendente

Descripción

La instrucción "Contador ascendente" permite aumentar en uno el valor del contador indicado cuando hay un flanco de señal ascendente en la entrada de arranque. El valor de contaje puede seguir incrementándose hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa al detectarse un flanco de señal ascendente.

La instrucción "Contador ascendente" requiere una operación lógica precedente para la evaluación de flancos y sólo se puede colocar en el margen derecho del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

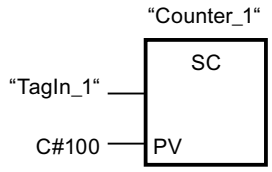
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	Entrada de arranque
<Contador>	InOut/Input	COUNTER	C	Contador cuyo valor se aumenta.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

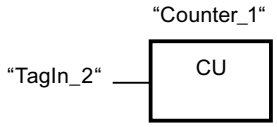
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

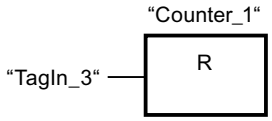
Segmento 1



Segmento 2



Segmento 3



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se incrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

CD: Contador descendente

Descripción

La instrucción "Contador descendente" permite reducir en uno el valor del contador indicado cuando hay un flanco de señal ascendente en la entrada de arranque. El valor de contaje puede seguir decrementándose hasta alcanzar el límite "0". Una vez alcanzado el valor límite, el valor de contaje no se modifica al detectarse un flanco de señal ascendente.

La instrucción "Contador descendente" requiere una operación lógica precedente para la evaluación de flancos y sólo se puede colocar en el margen derecho del segmento.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

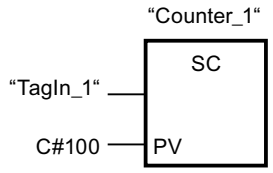
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, T, C, D, L	Entrada de arranque
<Contador>	InOut/Input	COUNTER	C	Contador cuyo valor se decrementa.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

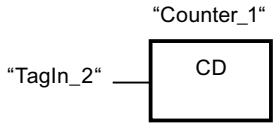
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

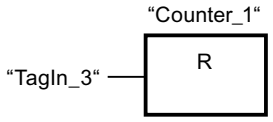
Segmento 1



Segmento 2



Segmento 3



Cuando el estado lógico del operando "TagIn_1" cambia de "0" a "1" (señal de flanco ascendente), el contador "Counter_1" se ajusta en el valor "100".

El valor del contador "Counter_1" se decrementa en uno si el estado lógico del operando "TagIn_2" cambia de "0" a "1".

Si el estado lógico del operando "TagIn_3" es "1", se pone a "0" el valor del contador "Counter_1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Comparación

CMP ==: Igual

Descripción

La instrucción "Igual" permite consultar si el valor de la entrada IN1 es igual al valor de la entrada IN2.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Nota

Comparación de números en coma flotante

Si desea comparar tipos de datos REAL o LREAL, utilice en lugar de la instrucción "CMP ==: Igual" la instrucción "IN_RANGE: Valor dentro del rango".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Igual":

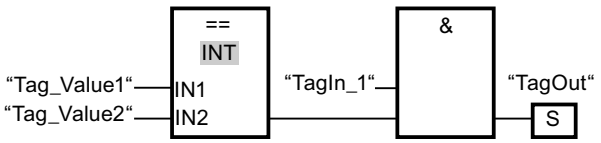
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" = "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP <>: Diferente

Descripción

La instrucción "Diferente" permite consultar si el valor de la entrada IN1 es distinto al valor de la entrada IN2.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Diferente":

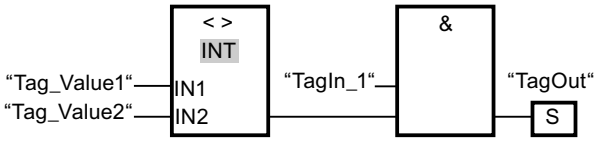
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Secuencias de bits, enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <> "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP >=: Mayor o igual

Descripción

La instrucción "Mayor o igual" permite consultar si el valor de la entrada IN1 es mayor o igual que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor o igual":

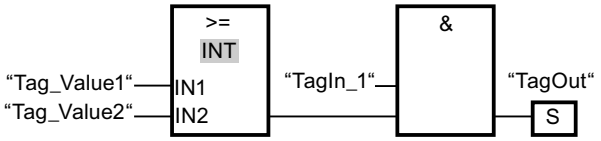
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" >= "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3253)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP <=: Menor o igual

Descripción

La instrucción "Menor o igual" permite consultar si el valor de la entrada IN1 es menor o igual que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor o igual":

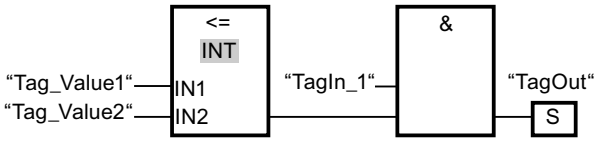
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" <= "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP >: Mayor

Descripción

La instrucción "Mayor" permite consultar si el valor de la entrada IN1 es mayor que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Mayor":

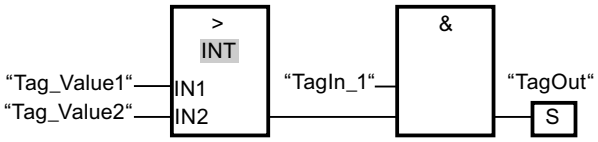
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" > "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CMP <: Menor

Descripción

La instrucción "Menor" permite consultar si el valor de la entrada IN1 es menor que el valor de la entrada IN2. Ambos valores de comparación deben ser del mismo tipo de datos.

Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Menor":

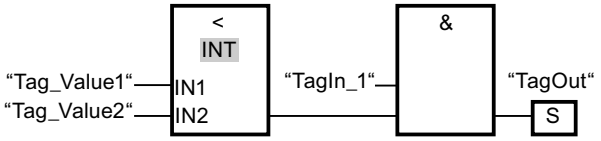
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer valor de comparación
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo valor de comparación

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La salida "TagOut" se activa cuando se cumplen las condiciones siguientes:

- El operando "TagIn_1" devuelve el estado lógico "1".
- Se cumple la condición de la instrucción de comparación cuando "Tag_Value1" < "Tag_Value2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3253)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

Funciones matemáticas

ADD: Sumar

Descripción

La instrucción "Sumar" permite sumar el valor de la entrada IN1 al valor de la entrada IN2 y consultar la suma en la salida OUT (OUT := IN1+IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar":

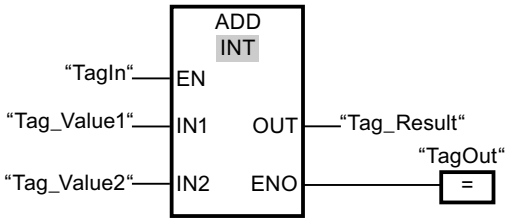
Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Primer sumando
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Segundo sumando
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Suma

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Sumar". El valor del operando "Tag_Value1" se suma al valor del operando "Tag_Value2". El resultado de la suma se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de cálculo de una ecuación (Página 3257)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SUB: Restar

Descripción

La instrucción "Restar" permite restar el valor de la entrada IN2 al valor de la entrada IN1 y consultar el resultado en la salida OUT (OUT := IN1-IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Restar":

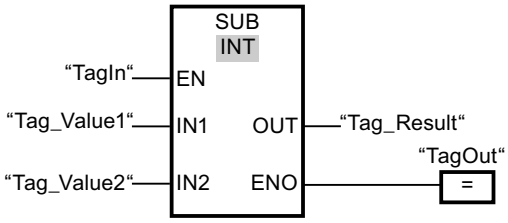
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Minuendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Sustraendo
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Diferencia

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Restar". El valor del operando "Tag_Value2" se resta del valor del operando "Tag_Value1". El resultado de la resta se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MUL: Multiplicar

Descripción

La instrucción "Multiplicar" permite multiplicar el valor de la entrada IN1 con el valor de la entrada IN2 y consultar el producto en la salida OUT (OUT := IN1*IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplicar":

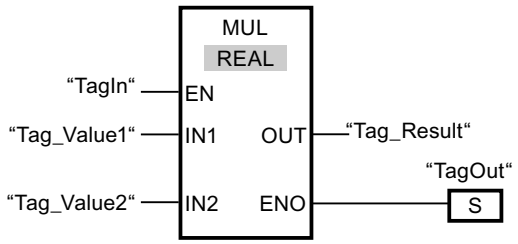
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Multiplicador
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Multiplicando
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Producto

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar". El valor del operando "Tag_Value1" se multiplica por el valor del operando "Tag_Value2". El resultado de la multiplicación se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Vista general de los tipos de datos válidos (Página 2321)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Insertar entradas y salidas adicionales en los elementos FUP (Página 2741)

Cablear operandos en instrucciones FUP (Página 2745)

DIV: Dividir

Descripción

La instrucción "Dividir" permite dividir el valor de la entrada IN1 por el valor de la entrada IN2 y consultar el cociente en la salida OUT (OUT := IN1/IN2).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT.
- Un número en coma flotante tiene un valor no válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Dividir":

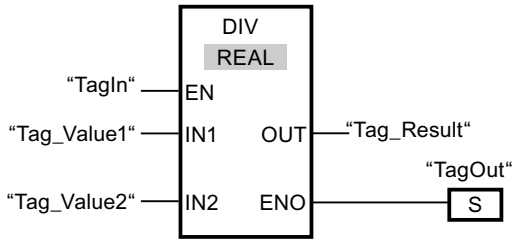
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Valor del cociente

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Dividir". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resultado de la división se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Ejemplo de cálculo de una ecuación (Página 3257)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MOD: Obtener resto de división

Descripción

La instrucción "Obtener resto de división" permite dividir el valor de la entrada IN1 por el valor de la entrada IN2 y consultar el resto en la salida OUT.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Obtener resto de división":

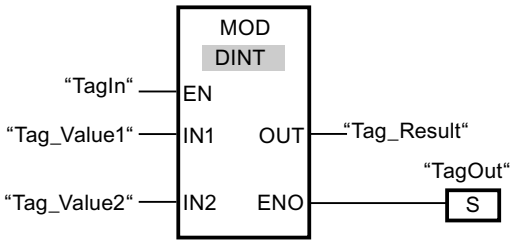
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	Enteros	I, Q, M, D, L, P o constante	Dividendo
IN2	Input	Enteros	I, Q, M, D, L, P o constante	Divisor
OUT	Output	Enteros	I, Q, M, D, L, P	Resto de división

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Obtener resto de división". El valor del operando "Tag_Value1" se divide por el valor del operando "Tag_Value2". El resto de la división se almacena en el operando "Tag_Result". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

NEG: Generar complemento a dos

Descripción

La instrucción "Generar complemento a dos" permite cambiar el signo del valor de la entrada IN y consultar el resultado en la salida OUT. Por ejemplo, si la entrada IN tiene un valor positivo, se deposita el equivalente negativo de este valor en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT (sólo válido para números enteros).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a dos":

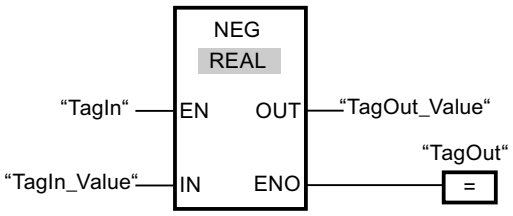
Parámetro	Declaración	Tipo de datos		Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Enteros, números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Complemento a dos del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Generar complemento a dos". Se invierte el signo del valor de la entrada "TagIn_Value" y el resultado se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" permite calcular el valor absoluto del valor indicado en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor absoluto":

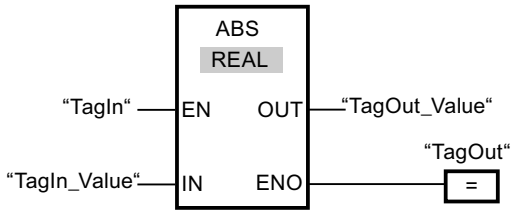
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	REAL	I, Q, M, D, L, P	Valor absoluto del valor de entrada

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	-6.234
OUT	TagOut_Value	6.234

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor absoluto". La instrucción calcula el valor absoluto del valor de la entrada "TagIn_Value" y deposita el resultado en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor menor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

En caso de que la verificación CEI esté desactivada, en los parámetros de entrada se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, las instrucciones devolverán un resultado no definido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

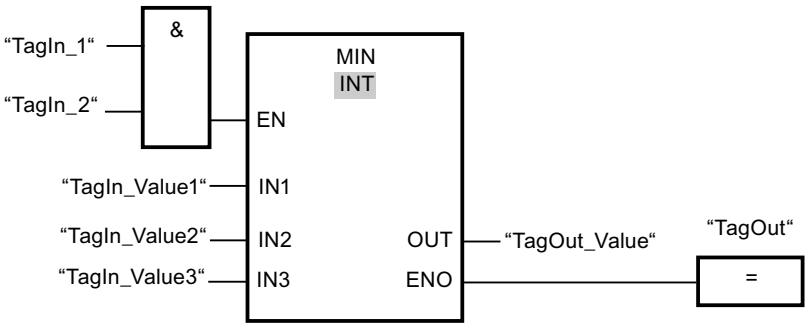
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	12222

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar mínimo". La instrucción compara los valores de los operandos indicados y copia el valor menor ("TagIn_Value1") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Quitar entradas y salidas (Página 2698)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor mayor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

En caso de que la verificación CEI esté desactivada, en los parámetros de entrada se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, las instrucciones devolverán un resultado no definido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

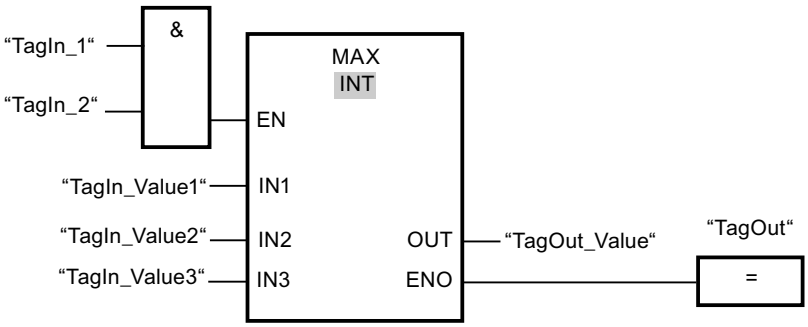
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

Si el estado lógico de los operandos "TagIn_1" y "TagIn_2" es "1", se ejecuta la instrucción "Determinar máximo". La instrucción compara los valores de los operandos indicados y copia el valor mayor ("TagIn_Value2") en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor en la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no se define y la salida de habilitación ENO es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor de la entrada MN es mayor que el valor de la entrada MX.

Nota

En caso de que la verificación CEI esté desactivada, en los parámetros de entrada se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, las instrucciones devolverán un resultado no definido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

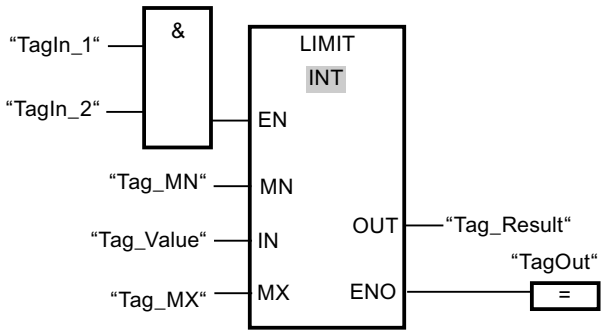
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite inferior
IN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Valor de entrada
MX	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite superior
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_MN	12000
IN	Tag_Value	8000

Parámetro	Operando	Valor
MX	Tag_MX	16000
OUT	Tag_Result	12000

Si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1", se ejecuta la instrucción "Ajustar valor límite". El valor del operando "Tag_Value" se compara con los valores de los operandos "Tag_MN" y "Tag_MX". Puesto que el valor del operando "Tag_Value" es menor que el valor límite inferior, el valor del operando "Tag_MN" se copia en la salida "Tag_Result". Si no se producen errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" permite elevar al cuadrado el valor de la entrada IN y consultar el resultado en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular cuadrado":

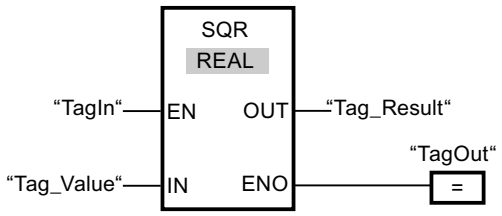
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Cuadrado del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	5.0
OUT	Tag_Result	25.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular cuadrado". La instrucción eleva al cuadrado el valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" permite extraer la raíz cuadrada al valor de la entrada IN y consultar el resultado en la salida OUT. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido. Si el valor de la entrada IN es "0", el resultado también es "0".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

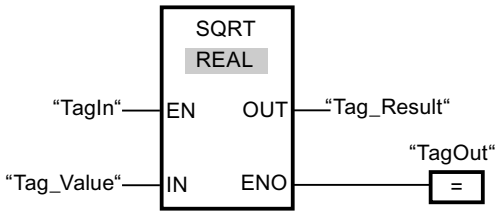
La tabla siguiente muestra los parámetros de la instrucción "Calcular raíz cuadrada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Raíz cuadrada del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	25.0
OUT	Tag_Result	5.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular raíz cuadrada". La instrucción extrae la raíz cuadrada del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2,718282) del valor de la entrada IN. El resultado se deposita en la salida OUT y se puede consultar allí. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la salida OUT devuelve un número en coma flotante no válido.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN es negativo.

Parámetros

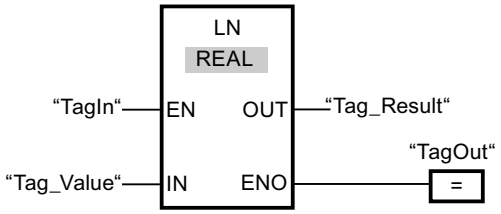
La tabla siguiente muestra los parámetros de la instrucción "Calcular logaritmo natural":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Logaritmo natural del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular logaritmo natural". La instrucción calcula el logaritmo natural del valor de la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2,718282) del valor indicado en la entrada IN. El resultado se emite en la salida OUT y se puede consultar allí (OUT = e^{IN}).

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

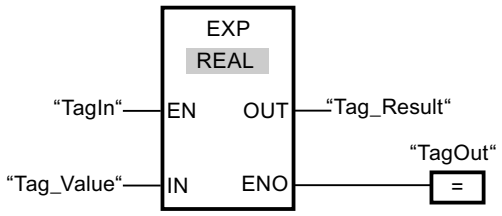
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor exponencial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Valor exponencial del valor de entrada IN

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor exponencial". La instrucción calcula la potencia en base e del valor del operando "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno de un ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

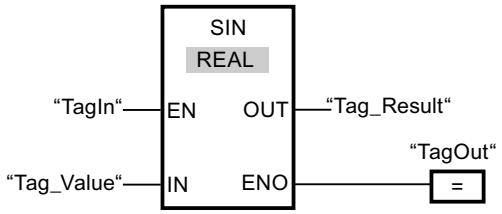
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de seno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Seno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	1.0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de seno". La instrucción calcula el seno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno de un ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

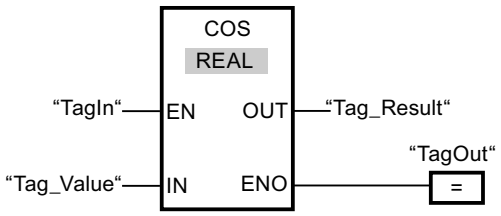
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de coseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Coseno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+1.570796 (π/2)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de coseno". La instrucción calcula el coseno del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente de un ángulo. El tamaño del ángulo se indica en radianes en la entrada IN. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

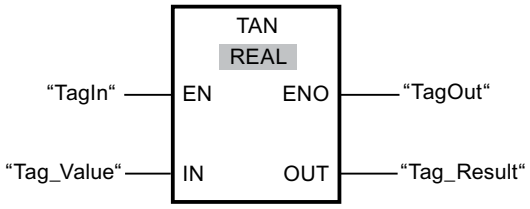
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de tangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tangente del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	+3.141593 (π)
OUT	Tag_Result	0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de tangente". La instrucción calcula la tangente del ángulo indicado en la entrada "Tag_Value" y deposita el resultado en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno indicado en la entrada IN el tamaño del ángulo que corresponde a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

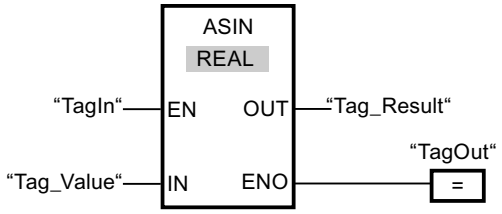
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de seno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+1.570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcoseno". La instrucción calcula el tamaño del ángulo que corresponde al valor de seno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno indicado en la entrada IN el tamaño del ángulo que corresponde a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre 0 y +π.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.
- El valor de la entrada IN está fuera del rango de valores permitido (-1 hasta +1).

Parámetros

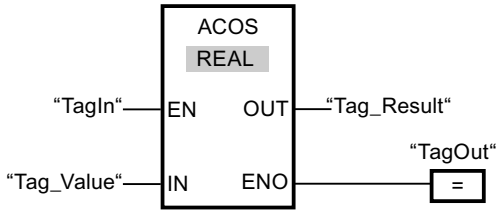
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcocoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de coseno
OUT	Output	Números en coma flotante	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0
OUT	Tag_Result	+1.570796 (π/2)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcocoseno". La instrucción calcula el tamaño del ángulo que corresponde al valor de coseno de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente indicado en la entrada IN el tamaño del ángulo que corresponde a este valor. En la entrada IN sólo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado se deposita en radianes en la salida OUT y puede estar comprendido entre -π/2 y +π/2.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El valor de la entrada IN no es un número en coma flotante válido.

Parámetros

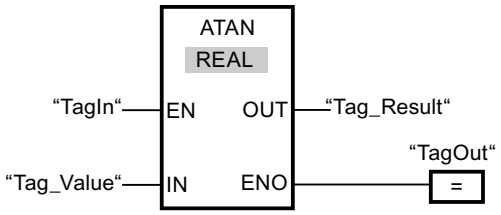
La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcotangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de tangente
OUT	Output	REAL	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1.0
OUT	Tag_Result	+0,785398 (π/4)

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Calcular valor de arcotangente". La instrucción calcula el tamaño del ángulo que corresponde al valor de tangente de la entrada "Tag_Value". El resultado de la instrucción se deposita en la salida "Tag_Result". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Principios básicos del mecanismo EN/ENO (Página 2539)
Insertar elementos FUP (Página 2722)
Seleccionar el tipo de datos de un elemento FUP (Página 2726)
Editar elementos FUP (Página 2738)
Cablear operandos en instrucciones FUP (Página 2745)
Principios básicos de la palabra de estado (Página 2545)
Activar los bits de estado en instrucciones con números enteros (Página 2551)
Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Transferencia

MOVE: Copiar valor

Descripción

La instrucción "Copiar valor" transfiere el contenido del operando de la entrada IN al operando de la salida OUT1. La transferencia se efectúa siempre por orden ascendente de direcciones.

La tabla siguiente muestra las posibles transferencias:

Origen (IN)	Destino (OUT1)	
	Con verificación CEI	Sin verificación CEI
BYTE	BYTE, WORD, DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD, CHAR
WORD	WORD; DWORD	BYTE, WORD, DWORD, INT, DINT, TIME, S5TIME, DATE, TOD, CHAR
DWORD	DWORD	BYTE, WORD, DWORD, INT, DINT, REAL, TIME, DATE, TOD, CHAR
INT	INT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
DINT	DINT	BYTE, WORD, DWORD, INT, DINT, TIME, DATE, TOD
REAL	REAL	DWORD, REAL
TIME	TIME	BYTE, WORD, DWORD, INT, DINT, TIME
S5TIME	S5TIME	WORD, S5TIME
DATE	DATE	BYTE, WORD, DWORD, INT, DINT, DATE
TOD	TOD	BYTE, WORD, DWORD, INT, DINT, TOD
CHAR	CHAR	BYTE, WORD, DWORD, CHAR
COUNTER	INT, WORD, COUNTER	WORD, DWORD, INT, UINT, DINT, UDINT
TIMER	INT, WORD, TIMER	WORD, DWORD, INT, UINT, DINT, UDINT

Si la longitud de bits del tipo de datos de la entrada IN supera la longitud de bits del tipo de datos de la salida OUT1, se pierden los bits más significativos del valor de origen. Si la longitud de bits del tipo de datos de la entrada IN es inferior a la longitud de bits del tipo de datos de la salida OUT1, se sobrescriben con ceros los bits más significativos del valor de destino.

Para copiar campos y estructuras pueden utilizarse también las instrucciones "Copiar área" (BLKMOV) y "Copiar área sin interrupciones" (UBLKMOV).

Parámetro

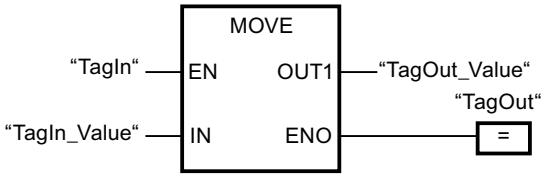
La tabla siguiente muestra los parámetros de la instrucción "Copiar valor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, DATE, TOD, CHAR, TIMER, COUNTER	I, Q, M, D, L o constante	Valor de origen
OUT1	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, DATE, TOD, CHAR, TIMER, COUNTER	I, Q, M, D, L	Dirección de destino

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
OUT1	TagOut_Value	0011 1111 1010 1111

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Copiar valor". La instrucción copia los contenidos del operando "TagIn_Value" en el operando "TagOut_Value" y pone la salida "TagOut" al estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

El MCR (Master Control Relay) (Página 2564)

MCR<: Abrir áreas MCR (Página 5985)

BLKMOV: Copiar área

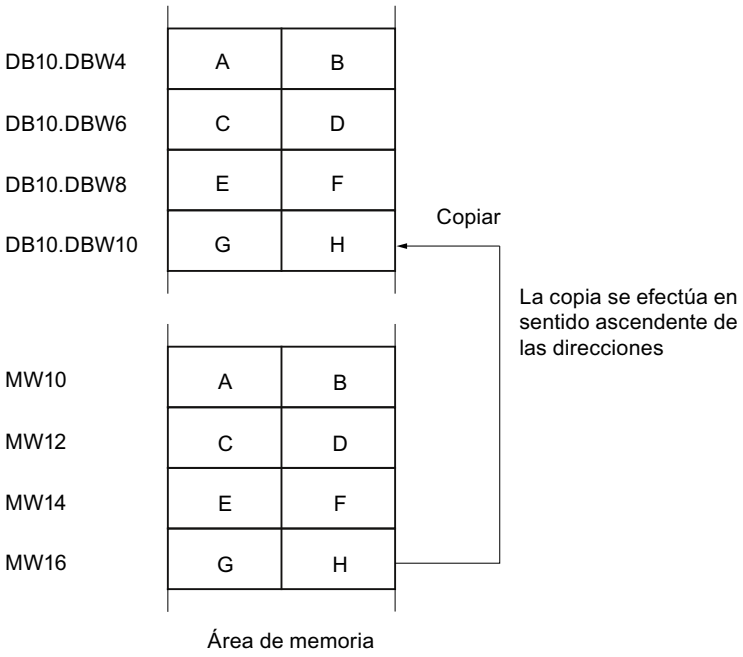
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

Mientras el área de origen no forme parte de un bloque de datos que solo figura en la memoria de carga, no hay limitaciones en cuanto a la profundidad de anidamiento.

Por el contrario, en caso de interrumpirse la ejecución de BLKMOV, en la que se copia desde un DB no relevante para la ejecución, no puede anidarse más la ejecución de la BLKMOV.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Reglas para copiar bloques de datos no relevantes para la ejecución

El área de origen también puede encontrarse en un bloque de datos no relevante para la ejecución en la memoria de carga. Los bloques de datos no relevantes para la ejecución se identifican con la palabra clave UNLINKED.

Si en la memoria de trabajo se copia un bloque de datos no relevante para la ejecución con la instrucción "Copiar área" y, al mismo tiempo, este se vuelve a cargar p. ej. desde la programadora, la ejecución de la instrucción puede retardarse varios milisegundos. Esto prolonga el ciclo del OB y puede activar la vigilancia del tiempo de ciclo.

Si un bloque de datos no relevante para la ejecución se copia con la instrucción "Copiar área" y la operación de copia se interrumpe, la instrucción no se podrá seguir ejecutando.

Si en la CPU existe la instrucción "Leer de un bloque de datos de la memoria de carga", debe utilizarse dicha instrucción para leer los bloques de datos no relevantes para la ejecución en la memoria de carga. Si se utiliza la instrucción "Copiar área" se emite el error W#16#8092.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8092	El área de origen o destino está solo en la memoria de carga.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 512 bytes como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU del S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si sólo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

FILL: Rellenar área

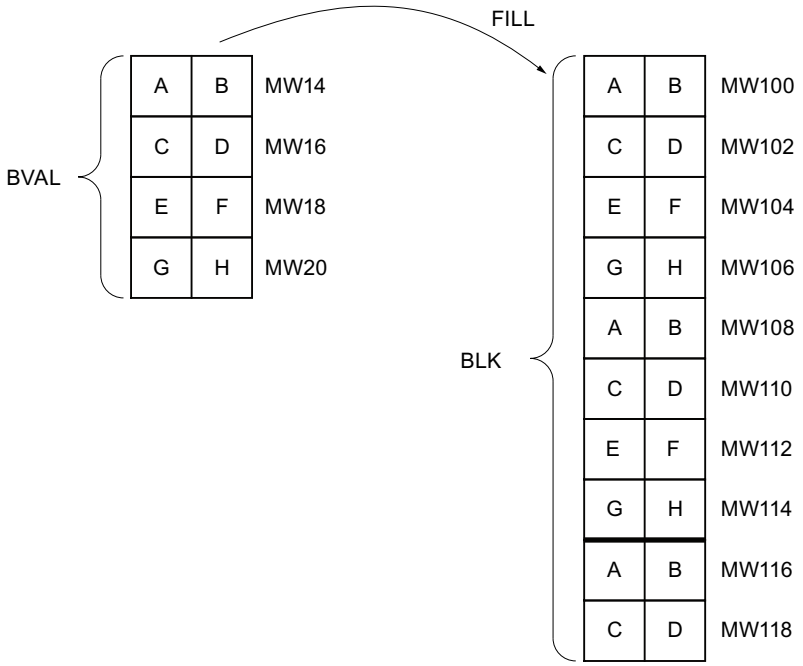
Descripción

La instrucción "Rellenar área" permite rellenar un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino las veces necesarias hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: El contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, sólo se copian los datos que caben en el área de destino.

Si el área de origen o destino realmente existente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo de datos BOOL, la longitud indicada del área debe ser divisible entre 8 ya que, de lo contrario, la instrucción no se ejecuta.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro BVAL o BLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU del S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
BVAL	Input	ANY	I, Q, M, D, L	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
BLK	Output	ANY	I, Q, M, D, L	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" lee el contenido del parámetro IN y lo convierte según los tipos de datos elegidos en el cuadro de la instrucción. El valor convertido se deposita en la salida OUT.

Encontrará más información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir valor":

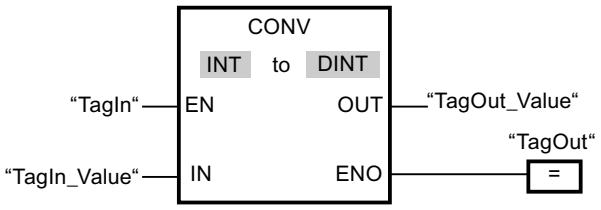
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Enteros, números en coma flotante, BCD16, BCD32	I, Q, M, D, L, P o constante	Valor que se convierte.
OUT	Output	Enteros, números en coma flotante, BCD16, BCD32	I, Q, M, D, L, P	Resultado de la conversión

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", el contenido del operando "TagIn_Value" se lee y se convierte en un número entero (16 bits). El resultado se deposita en el operando "TagOut_Value". Si la instrucción se ha ejecutado correctamente, la salida "TagOut" se pone a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Conversión explícita (Página 2522)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor de la entrada IN al siguiente número entero. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte a un número entero del tipo de datos DINT. Si el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear número":

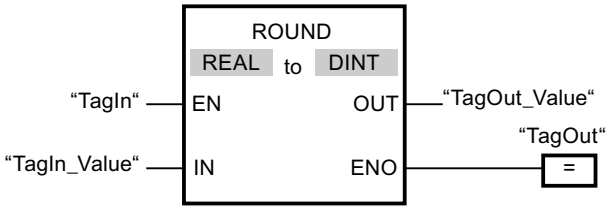
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se redondea.
OUT	Output	DINT	I, Q, M, D, L, P	Resultado del redondeo

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	2	-2

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear número". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero par más próximo y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor de la entrada IN al siguiente número entero superior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero superior. El resultado de la instrucción se emite en la salida OUT y se puede consultar allí. El valor de salida puede ser mayor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero superior":

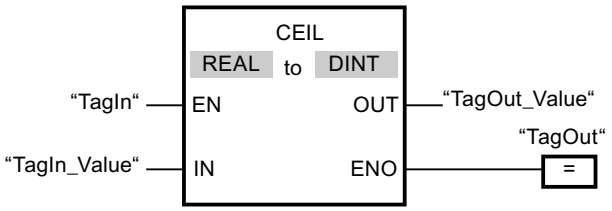
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero superior

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	1	0

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero superior". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero superior más próximo y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor de la entrada IN al siguiente número entero inferior. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte en el siguiente número entero inferior. El resultado de la instrucción se deposita en la salida OUT y se puede consultar allí. El valor de salida puede ser menor o igual al valor de entrada.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero inferior":

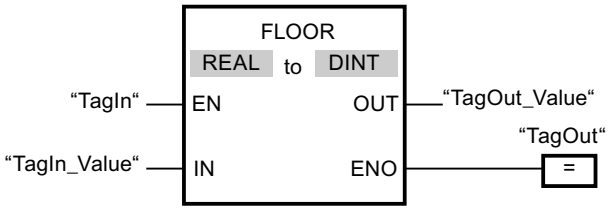
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Números en coma flotante	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros, números en coma flotante	I, Q, M, D, L, P	Resultado con el siguiente entero inferior

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	0.50000000	-0.50000000
OUT	TagOut_Value	0	-1

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Redondear un número en coma flotante al siguiente entero inferior". El número en coma flotante de la entrada "TagIn_Value" se redondea al número entero inferior más próximo y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un valor entero a partir del valor de la entrada IN. El valor de la entrada IN se interpreta como número en coma flotante. La instrucción selecciona sólo la parte entera del número en coma flotante y la deposita sin decimales en la salida OUT.

La salida de habilitación ENO devuelve el estado lógico "0" cuando se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- Durante la ejecución ocurren errores, p. ej. se produce un rebase por exceso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Truncar a entero":

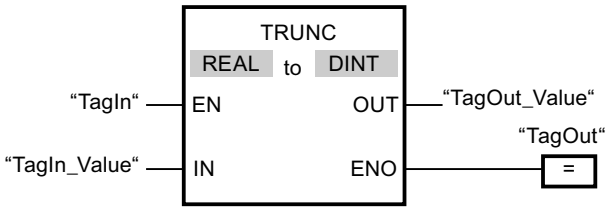
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L o constante	Valor de entrada
OUT	Output	Enteros	I, Q, M, D, L	Parte entera del valor de entrada

En las listas desplegables "<???>" del cuadro de la instrucción se pueden seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
IN	TagIn_Value	1.50000000	-1.50000000
OUT	TagOut_Value	1	-1

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Truncar a entero". La parte entera del número en coma flotante de la entrada "TagIn_Value" se convierte en un número entero y se deposita en la salida "TagOut_Value". Si no ocurren errores al ejecutar la instrucción, se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala inversamente proporcional al valor de entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

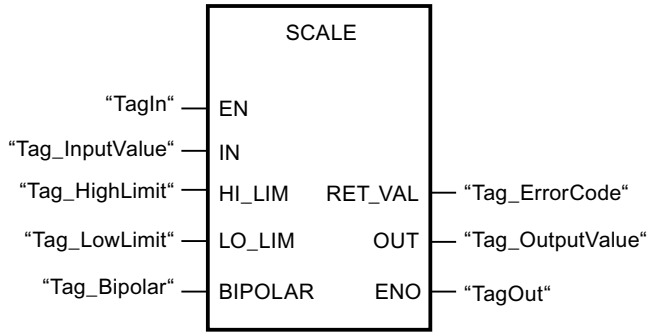
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error (W#16#....)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1".
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" desescala el número en coma flotante del parámetro IN en unidades físicas entre un valor límite inferior y un valor límite superior y lo convierte en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Input	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

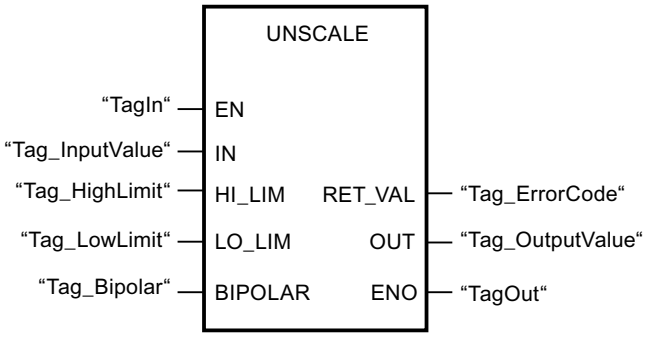
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_OutputValue	22
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Control del programa

JMP: Saltar si RLO = 1

Descripción

La instrucción "Saltar si RLO = 1" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente. El segmento de destino tiene que marcarse con una

etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima del cuadro de la instrucción.

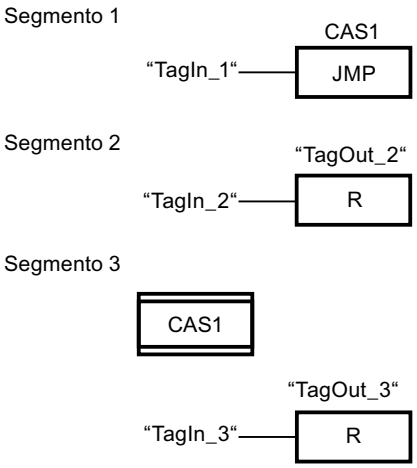
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En un segmento no se puede utilizar más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si no se cumple la condición en la entrada de la instrucción (RLO = 0), la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

JMPN: Saltar si RLO = 0

Descripción

La instrucción "Saltar si RLO = 0" permite interrumpir la ejecución lineal del programa y continuarla en un segmento diferente, si el resultado lógico de la entrada de la instrucción es

"0". El segmento de destino tiene que marcarse con una etiqueta (LABEL). El nombre de la etiqueta se indica en el comodín situado encima del cuadro de la instrucción.

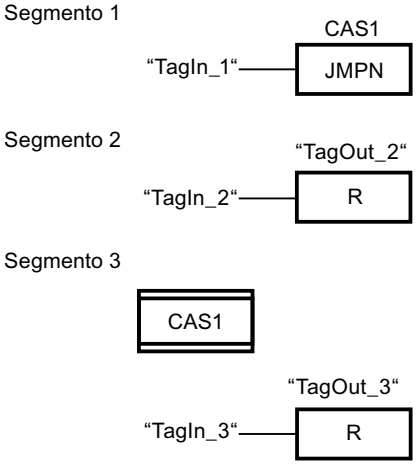
La etiqueta indicada debe encontrarse en el mismo bloque en el que se ejecuta la instrucción. Su nombre debe ser unívoco en el bloque. En un segmento no se puede utilizar más de una bobina de salto.

Si el resultado lógico (RLO) de la entrada de la instrucción es "0", se ejecuta el salto al segmento identificado por la etiqueta indicada. El salto puede realizarse hacia números de segmento superiores o inferiores.

Si el RLO de la entrada de la instrucción es "1", la ejecución del programa continúa en el segmento siguiente.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "0", se ejecuta la instrucción "Saltar si RLO = 0". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

LABEL: Etiqueta

Descripción

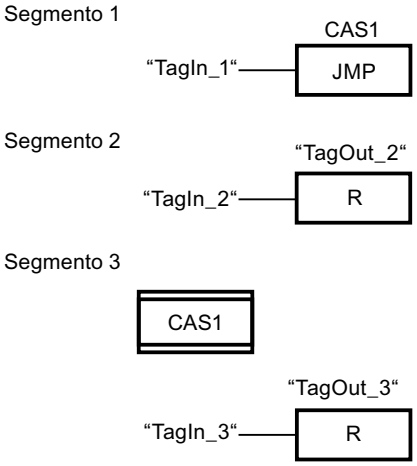
La etiqueta de salto marca un segmento de destino en el que se debe continuar la ejecución del programa tras ejecutar una instrucción de salto.

La etiqueta y la instrucción en la que se indica la etiqueta como destino del salto deben encontrarse en el mismo bloque. El nombre de una etiqueta debe ser unívoca en el bloque.

En un segmento sólo se puede colocar una etiqueta. A toda etiqueta se puede acceder desde distintas posiciones.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn_1" devuelve el estado lógico "1", se ejecuta la instrucción "Saltar si RLO = 1". Por este motivo, se interrumpe la ejecución lineal del programa y se prosigue en el segmento 3, marcado por la etiqueta CAS1. Si la entrada "TagIn_3" devuelve el estado lógico "1", se desactiva la salida "TagOut_3".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

RET: Retroceder

Descripción

La instrucción "Retroceder" permite finalizar la ejecución de un bloque.

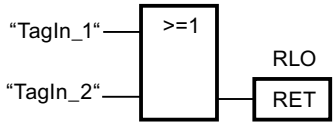
Si el resultado lógico (RLO) de la entrada de la instrucción "Retroceder" es "1", se finaliza la ejecución del programa en el bloque llamado actualmente y se continúa en el bloque que efectúa la llamada (p. ej. en el OB que llama) después de la función de llamada. El RLO determina el estado (ENO) de la función de llamada. La instrucción representa el RLO como un valor de parámetro fijo. Si el RLO en la entrada de la instrucción "Retroceder" es "0", la instrucción no se ejecuta. La ejecución del programa continúa en el siguiente segmento del bloque llamado.

Si la instrucción "Retroceder" está programada en el último segmento del bloque llamado, la salida de habilitación ENO pondrá la función de llamada al estado lógico actual del RLO.

Si se finaliza un bloque de organización con la instrucción "Retroceder", la CPU continúa con el programa de sistema.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el estado lógico de uno de los operandos "TagIn_1" o "TagIn_2" es "1", se ejecuta la instrucción "Retroceder". La ejecución del programa se finaliza en el bloque llamado y continúa en el bloque que efectúa la llamada. La salida de habilitación ENO de la función de llamada adopta el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

OPN: Abrir bloque de datos global

Descripción

La instrucción "Abrir bloque de datos global" permite abrir un bloque de datos (DB). El número del bloque de datos se transmite al registro del DB. Los siguientes comandos de DB acceden a los bloques correspondientes en función de los contenidos del registro.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos global":

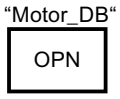
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Bloque de datos>	Input	DB_ANY	-	Bloque de datos que se abre.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

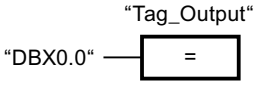
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1



Segmento 2



En el segmento 1 se llama al bloque de datos "Motor_DB". El número del bloque de datos se transmite al registro del DB. En el segmento 2 se consulta el operando "DBX0.0". El estado lógico del operando "DBX0.0" se asigna al operando "Tag_Output".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

OPNI: Abrir bloque de datos de instancia

Descripción

La instrucción "Abrir bloque de datos de instancia" permite abrir un bloque de datos (DB). El número del bloque de datos se transmite al registro del DI. Los siguientes comandos del DI acceden a los bloques correspondientes en función de los contenidos del registro.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos de instancia":

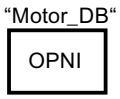
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Bloque de datos>	Input	DB_ANY	-	Bloque de datos que se abre.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

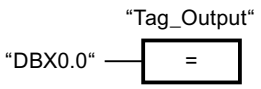
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1



Segmento 2



En el segmento 1 se llama al bloque de datos "Motor_DB". El número del bloque de datos se transmite al registro del DI. En el segmento 2 se consulta el operando "DBX0.0". El estado lógico del operando "DBX0.0" se asigna al operando "Tag_Output".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Control del tiempo de ejecución

COMPRESS: Comprimir la memoria de la CPU

Descripción

La instrucción "Comprimir la memoria de la CPU" comprime la memoria de trabajo de la CPU o la memoria de carga RAM. Los huecos creados por haber borrado o cargado bloques, se rellenan juntando los bloques existentes.

Con la llamada de la instrucción "Comprimir la memoria de la CPU" se lanza el proceso de compresión. El proceso de compresión se reparte entre varios ciclos del programa. La vigilancia del proceso de compresión se realiza mediante los valores de los distintos parámetros. Si el parámetro BUSY de la instrucción tiene el valor "1", significa que el proceso de compresión se está ejecutando todavía. El valor "1" del parámetro DONE indica que ha finalizado el proceso de compresión. La información de error puede consultarse en el parámetro RET_VAL.

Si se está ejecutando un proceso de compresión lanzado externamente, la llamada de la instrucción "Comprimir la memoria de la CPU" origina un error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comprimir la memoria de la CPU":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BUSY	Output	BOOL	I, Q, M, D, L	Información de si está activo el proceso de compresión.
DONE	Output	BOOL	I, Q, M, D, L	Información de si ha finalizado el proceso de compresión.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error El estado del proceso de compresión puede consultarse en los parámetros BUSY y DONE.
8091	Ya se está ejecutando un proceso de compresión lanzado externamente.
8092	La instrucción no se puede ejecutar porque se cumple una de las condiciones siguientes: • La función "Borrar bloque" está activa. • Una función de test y de puesta en servicio repercute sobre un bloque (p. ej. estado). • La función "Copiar bloques" se ha activado por un impulso externo. • La CPU H se encuentra en un proceso de acoplamiento o actualización.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en

la configuración de la CPU. Reiniciando el tiempo de vigilancia del ciclo se impide que se produzcan errores o que la CPU pase a modo STOP.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" puede utilizarse en bloques de la clase de prioridad 1 (el OB de ciclo), así como en los bloques que se llaman desde ellos.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

Nota

En caso de utilizar la instrucción "Reiniciar tiempo de vigilancia del ciclo", debe asegurarse que no se produce un bucle sin fin en la ejecución cíclica del programa, es decir, en el OB1. De lo contrario, la CPU ya no alcanza el punto de control del ciclo. Como consecuencia no pueden ejecutarse ciertas funciones de la CPU (p. ej., la actualización de la memoria imagen de proceso).

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros.

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Si el resultado lógico (RLO) de la entrada de la instrucción es "1", la CPU pasa al estado operativo STOP y se finaliza el procesamiento del programa. El estado lógico de la salida de la instrucción no se evalúa.

Si el RLO de la entrada de la instrucción es "0", la instrucción no se ejecuta.

Parámetro

La instrucción "Finalizar programa" no tiene parámetros.

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" permite detener la ejecución del programa durante un periodo predeterminado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible programar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

PROTECT: Cambiar nivel de protección

Descripción

La instrucción "Cambiar nivel de protección" permite modificar el nivel de protección "1" configurado para la CPU.

El nivel de protección "1" parametrizado con la opción activada "Eliminable por contraseña" hace que sea necesario parametrizar una contraseña. Esta contraseña permite durante el funcionamiento acceder en escritura a la CPU protegida contra escritura mediante la instrucción "Cambiar nivel de protección" (p. ej. cargar).

El nivel de protección se especifica en el parámetro MODE de la instrucción. El nivel de protección actual de la CPU se visualiza en la ventana de inspección en "Propiedades > Protección".

La tabla siguiente muestra una explicación de los niveles de protección:

Nivel de protección	Significado
1	Se permiten todas las funciones de PG. En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
2	• El programa y la configuración de la CPU no se pueden modificar. El programa contenido en la CPU se puede leer. • Se permiten las funciones de control, observación y comunicación del proceso. • Se permiten todas las funciones de información. • Si conoce la contraseña válida, podrá desactivar la protección contra escritura ajustada. • En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
3	• Aun conociendo la contraseña válida, no podrá desactivar la protección contra lectura y escritura ajustada. • Si al llamar la instrucción "Cambiar nivel de protección" existe una conexión legitimada, la llamada de la instrucción no tendrá ningún efecto en esta conexión.

Si se ha configurado el nivel de protección "2" o "3" para la CPU, la llamada de la instrucción "Cambiar nivel de protección" no tiene efecto alguno.

La efectividad del nivel de protección "2" o "3" una vez ejecutada la instrucción "Cambiar nivel de protección" puede consultarse en la lista parcial SZL con la ID de SZL W#16#0232 y el índice W#16#0004.

Nota

La instrucción "Cambiar nivel de protección" permite ajustar un nivel de protección más bajo que aquel ajustado con "Configurar hardware" de STEP 7. Supongamos que se ha configurado el nivel de protección 3 y a continuación se llama la instrucción "Cambiar nivel de protección" con el parámetro MODE = 12. Esto bloqueará la legitimación de la contraseña. Si la instrucción se llama nuevamente con el parámetro MODE = 0 o 1, el nivel de protección 3 se mantiene y será posible desactivar la protección de lectura y escritura introduciendo la contraseña válida.

Efectos de las transiciones de estado operativo en el nivel de protección

La tabla siguiente muestra qué efectos tienen las transiciones de estado operativo en el nivel de protección ajustado con la instrucción "Cambiar nivel de protección".

Acción	Efectos en el nivel de protección
• Selector de modo en "STOP" • Conexión (POWER ON) sin respaldo • Borrado total mediante el selector de modo • Cambios de configuración y/o del programa en el estado operativo "STOP" de la CPU S7-400	El nivel de protección se pone a "1". Se permiten todas las funciones de PG.
• Arranque en frío • Rearranque (en caliente)	El sistema operativo pone el nivel de protección a "1". Se permiten todas las funciones de PG. En caso necesario, llamando la instrucción "Cambiar nivel de protección" en el programa es posible ajustar el nivel de protección en "2" o "3".
Conexión (POWER ON) respaldada	No cambia
Transición de estado operativo RUN/ARRANQUE/PARADA -> STOP (debida a la llamada de la instrucción "Finalizar programa" (STP), a errores del programa sin el correspondiente OB de error o a una acción del usuario).	No cambia
S7-400: Rearranque por acción del usuario o automáticamente tras conexión (POWER ON)	No cambia

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Cambiar nivel de protección":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MODE	Input	WORD	I, Q, M, D, L o constante	Nivel de protección Valores posibles: • W#16#0000: Ajuste del nivel de protección "1" • W#16#0001: Ajuste del nivel de protección "2" con legitimación de la contraseña • W#16#000C: Ajuste del nivel de protección "3" sin legitimación de la contraseña En el parámetro MODE también se pueden indicar constantes.
RET_VAL	Output	INT	I, Q, M, D, L	Si ocurre un error durante la ejecución, se emite una información de error.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
8090	Valor no admisible en el parámetro MODE.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

CiR: Controlar proceso CiR

Descripción

La instrucción "Controlar proceso CiR" permite influir en el proceso CiR en el programa de usuario como se indica a continuación:

- Bloquear el proceso CiR. En este caso, se rechazará siempre la carga de una configuración modificada desde la programadora en la CPU. El bloqueo se aplica hasta que se elimina con la instrucción "Controlar proceso CiR".
- El proceso CiR se puede bloquear de forma condicionada, indicando un límite superior para el tiempo de sincronización CiR en el parámetro FRZ_TIME. En este caso, solo se permitirá cargar una configuración modificada desde la programadora en la CPU si el tiempo de sincronización CiR calculado en la CPU no excede el límite superior definido.
- Habilitar el proceso CiR. El límite superior del tiempo de sincronización CiR se establece al valor predeterminado de 1 000 ms.
- Determina si el proceso CiR está habilitado o no. Si el proceso CiR está habilitado o lo está condicionadamente, el límite superior actual del tiempo de sincronización CiR se indica en el parámetro A_FT de la instrucción.

Nota

Durante el tiempo de sincronización CiR, las salidas están congeladas y las entradas no se evalúan.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Controlar proceso CiR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición Los valores admisibles del parámetro MODE son: • 0: Función de información • 1: Habilitar proceso CiR • 2: Bloquear proceso CiR • 3: Bloquear proceso CiR de forma condicionada. El límite superior del tiempo de sincronización CiR se define en FRZ_TIME.
FRZ_TIME	Input	TIME	I, Q, M, D, L o constante	Límite superior del tiempo de sincronización CiR en milisegundos. Rango de valores admisible: 60 ... 2500 ms (valor predeterminado: 1000 ms) El parámetro FRZ_TIME solo es relevante cuando el parámetro MODE se establece en el valor "3".
RET_VAL	Output	INT	I, Q, M, D, L	Información de error Si el parámetro MODE está a "0", el parámetro RET_VAL indica si el proceso CiR está habilitado o no.
A_FT	Output	TIME	I, Q, M, D, L	Límite superior válido actualmente del tiempo de sincronización CiR

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores. Este código de error solo se puede emitir si el parámetro MODE presenta el valor "1", "2" o "3".
0001	El proceso CiR está habilitado. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".

Código de error* (W#16#...)	Explicación
0002	El proceso CiR está bloqueado. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
0003	El proceso CiR está bloqueado de forma condicionada. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
8001	La CPU no está lista para un proceso CiR. Se está utilizando una CPU H en el sistema H (modo "Individual") o una CPU estándar en modo multiprocesador.
8002	El valor del parámetro MODE no es admisible.
8003	El valor del parámetro FRZ_TIME no es admisible.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

La instrucción "Controlar proceso CiR" permite programar p. ej. que no se lance ningún proceso CiR en periodos en los que se requiera la capacidad máxima de la CPU para el procesamiento.

Para ello debe llamarse la instrucción "Controlar proceso CiR" antes del inicio del periodo con un aumento de la actividad de procesos en el programa y ejecutarla con el valor "2" (Bloquear proceso CiR) en el parámetro MODE. Con ello se bloquea el proceso CiR.

Para volver a habilitar el proceso CiR tras finalizar el periodo de actividad aumentada del proceso, la instrucción "Controlar proceso CiR" debe llamarse de nuevo y ejecutarse con el valor "1" en el parámetro MODE. Como alternativa, el parámetro MODE se puede poner al valor "3", con lo que el proceso CiR se bloquea de forma condicionada.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Operaciones lógicas con palabras

AND: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una Y lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina mediante una Y lógica con el bit 0 del valor de la entrada IN2. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits de los valores indicados.

El estado lógico del bit de resultado es "1" sólo si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

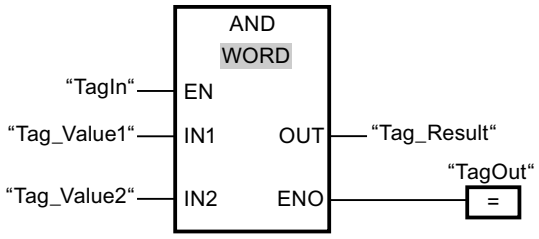
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
OUT	Output	WORD, DWORD	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	01010101 01010101
IN2	Tag_Value2	00000000 00001111
OUT	Tag_Result	00000000 00000101

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica Y". El valor del operando "Tag_Value1" se combina mediante una Y lógica con el valor del operando "Tag_Value2". El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

OR: Operación lógica O

Descripción

La instrucción "Operación lógica O" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina mediante una O lógica con el bit 0 del valor de la entrada IN2. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los bits de la variable indicada.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

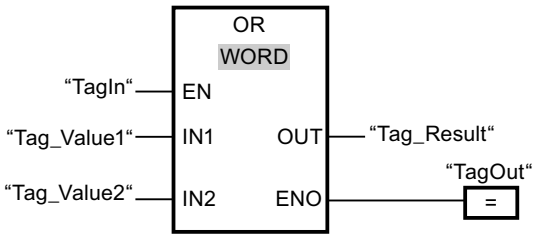
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
OUT	Output	WORD, DWORD	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	01010101 01010101
IN2	Tag_Value2	00000000 00001111
OUT	Tag_Result	01010101 01011111

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O". El valor del operando "Tag_Value1" se combina mediante una O lógica con el valor del operando "Tag_Value2". El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

XOR: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" permite combinar el valor de la entrada IN1 con el valor de la entrada IN2 de bit en bit mediante una O-exclusiva lógica y consultar el resultado en la salida OUT.

Al ejecutar la instrucción, el bit 0 del valor de la entrada IN1 se combina mediante una O-exclusiva lógica con el bit 0 del valor de la entrada IN2. El resultado se deposita en el bit 0 de la salida OUT. La misma combinación lógica se realiza para todos los demás bits del valor indicado.

El estado lógico del bit de resultado es "1" si el estado lógico de uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de ambos bits que deben combinarse lógicamente es "1" ó "0", se desactivará el bit de resultado correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

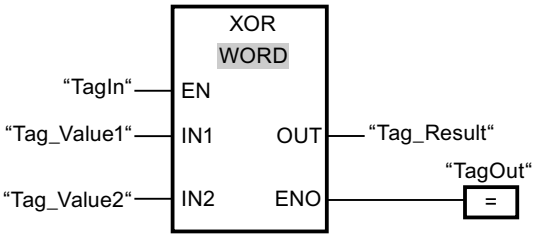
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN1	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Primer valor de la operación lógica
IN2	Input	WORD, DWORD	I, Q, M, D, L, P o constante	Segundo valor de la operación lógica
OUT	Output	WORD, DWORD	I, Q, M, D, L, P	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	01010101 01010101
IN2	Tag_Value2	00000000 00001111
OUT	Tag_Result	01010101 01011010

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Operación lógica O-exclusiva". El valor del operando "Tag_Value1" se combina mediante una O-exclusiva lógica con el valor del operando "Tag_Value2". El resultado se obtiene de bit en bit y se deposita en el operando "Tag_Result". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

INV: Complemento a 1

Descripción

La instrucción "Complemento a 1" permite invertir el estado lógico de los bits de la entrada IN. Al ejecutar la instrucción, el valor de la entrada IN se combina con una operación lógica O-exclusiva utilizando una plantilla hexadecimal (W#16#FFFF para números de 16 bits o DW#16#FFFF FFFF para números de 32 bits). Esto invierte el estado lógico de los distintos bits y deposita el resultado en la salida OUT.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Complemento a 1":

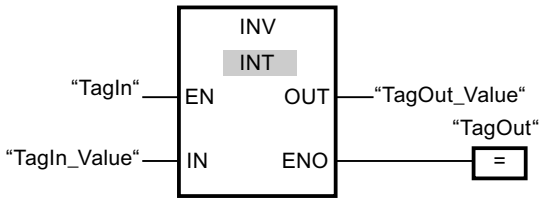
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	Enteros	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	Enteros	I, Q, M, D, L, P	Complemento a 1 del valor de la entrada IN

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	W#16#000F
OUT	TagOut_Value	W#16#FFF0

Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Complemento a 1". La instrucción invierte el estado lógico de los distintos bits de la entrada "TagIn_Value" y escribe el resultado en la salida "TagOut_Value". La salida de habilitación ENO y la salida "TagOut" adoptan el estado lógico "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

DECO: Descodificar

Descripción

La instrucción "Descodificar" permite activar un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor de la entrada IN y activa el bit del valor de salida, cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor de la entrada IN es mayor que 31, se ejecuta una instrucción modulo 32.

Parámetros

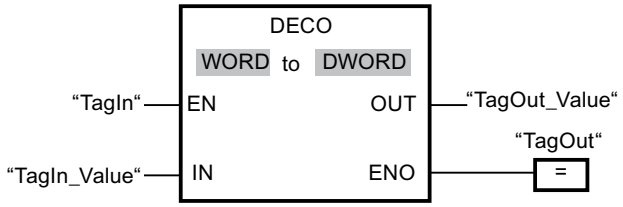
La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	WORD	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida.
OUT	Output	DWORD	I, Q, M, D, L, P	Valor de salida

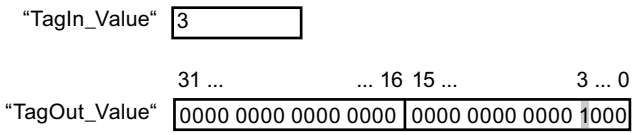
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Descodificar". La instrucción lee el número de bit "3" del valor del operando "TagIn_Value" de la entrada y activa el tercer bit del valor del operando "TagOut_Value" de la salida.

Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

ENCO: Codificar

Descripción

La instrucción "Codificar" lee el número del bit menos significativo en el valor de entrada y lo emite en la salida OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor de la entrada IN y escribe su número de bit en la variable de la salida OUT.

Parámetros

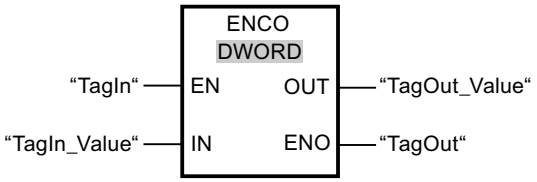
La tabla siguiente muestra los parámetros de la instrucción "Codificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

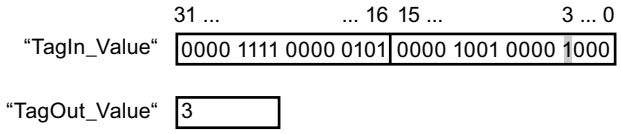
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Codificar". La instrucción selecciona el bit menos significativo activado en la entrada "TagIn_Value" y escribe el bit "3" en la variable de la salida "TagOut_Value".

Si no ocurren errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite elegir, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copia su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G tiene el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

La instrucción sólo se puede ejecutar si el estado lógico de la entrada de habilitación EN es "1" y las variables de todos los parámetros son del mismo tipo de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

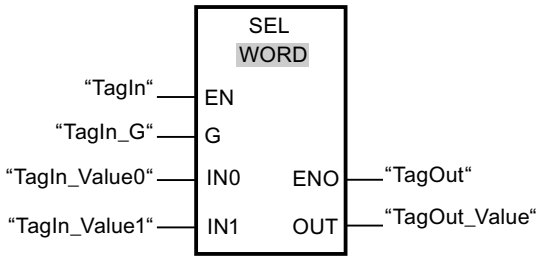
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
G	Input	BOOL	I, Q, M, D, L, T, C	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Resultado

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
G	TagIn_G	0	1
IN0	TagIn_Value0	W#16#0000	W#16#4C
IN1	TagIn_Value1	W#16#FFFF	W#16#5E
OUT	TagOut_Value	W#16#0000	W#16#5E

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Seleccionar". Dependiendo del estado lógico de la entrada "TagIn_G", se selecciona el valor de la entrada "TagIn_Value0" ó "TagIn_Value1" y se copia en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

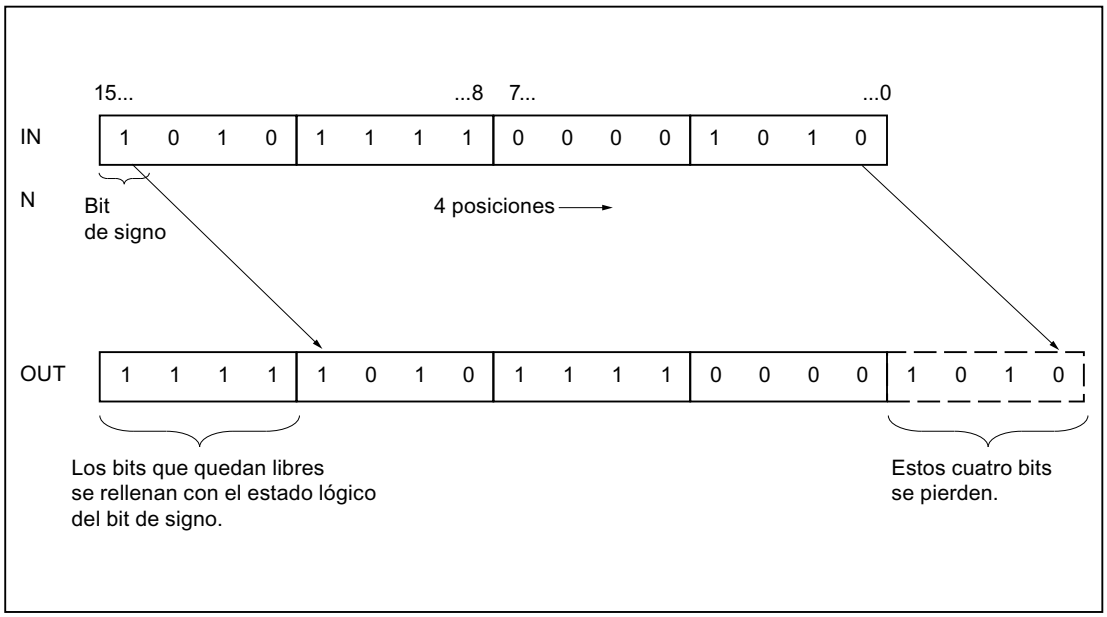
La instrucción "Desplazar a la derecha" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la derecha las posiciones de bit disponibles.

En los valores sin signo, se rellenan con ceros los bits que quedan libres en el área izquierda del operando al realizar el desplazamiento. Si el valor indicado lleva signo, las posiciones libres se ocupan con el estado lógico del bit de signo.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos INT se desplaza cuatro posiciones de bit hacia la derecha:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

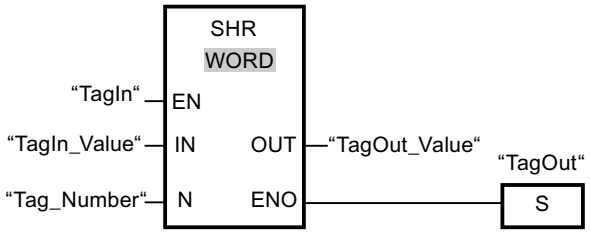
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L, P o constante	Valor que se desplaza
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que se desplaza el valor
OUT	Output	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	3
OUT	TagOut_Value	0000 0111 1111 0101

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la derecha". El contenido del operando "TagIn_Value" se desplaza tres posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

SHL: Desplazar a la izquierda

Descripción

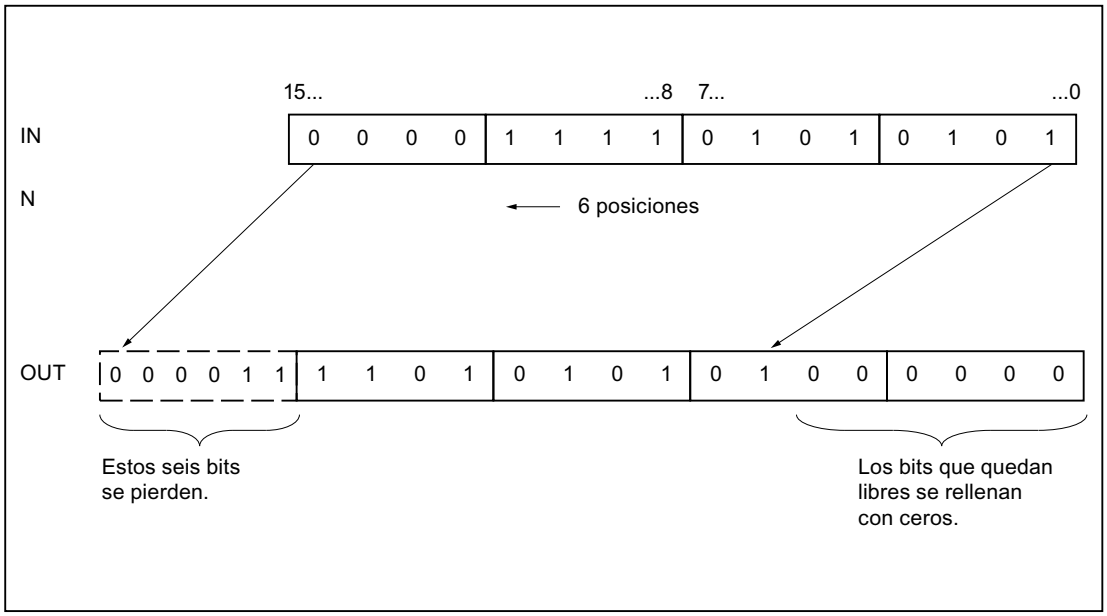
La instrucción "Desplazar a la izquierda" permite desplazar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando de la entrada IN se desplaza hacia la izquierda las posiciones de bit disponibles.

Los bits que quedan libres en el área derecha del operando al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

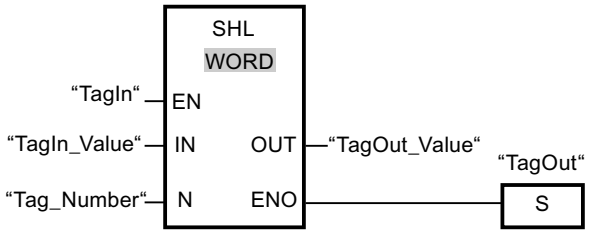
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L, P o constante	Valor que se desplaza
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que se desplaza el valor
OUT	Output	BYTE, WORD, DWORD, INT, DINT	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<???>" del cuadro de la instrucción se puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	0011 1111 1010 1111
N	Tag_Number	4
OUT	TagOut_Value	1111 1010 1111 0000

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Desplazar a la izquierda". El contenido del operando "TagIn_Value" se desplaza cuatro posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

ROR: Rotar a la derecha

Descripción

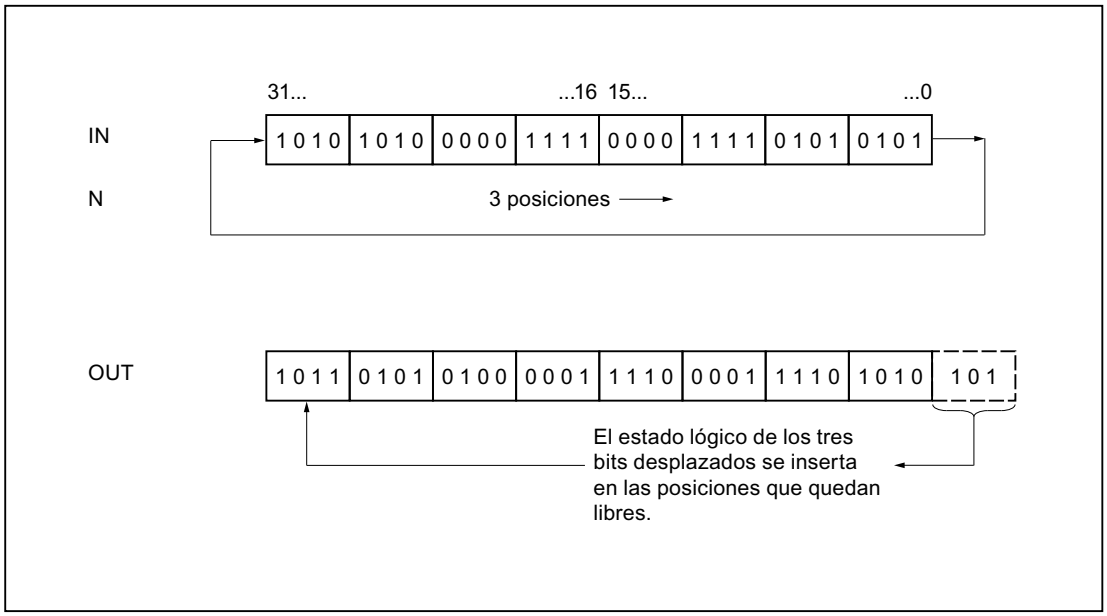
La instrucción "Rotar a la derecha" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la derecha y consultar el resultado en la salida OUT. El parámetro N

determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Parámetros

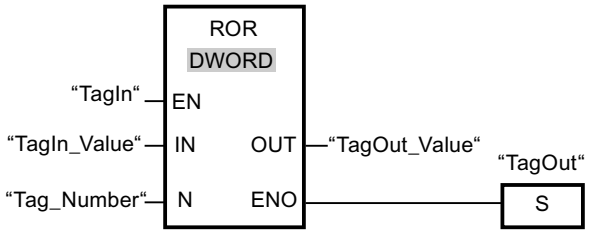
La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, DINT	I, Q, M, D, L, P o constante	Valor que se rota
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que rota el valor
OUT	Output	BYTE, WORD, DWORD, DINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	1010 1010 0000 1111 0000 1111 0101 0101
N	Tag_Number	5
OUT	TagOut_Value	1010 1101 0101 0000 0111 1000 0111 1010

Si el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la derecha". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la derecha. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

ROL: Rotar a la izquierda

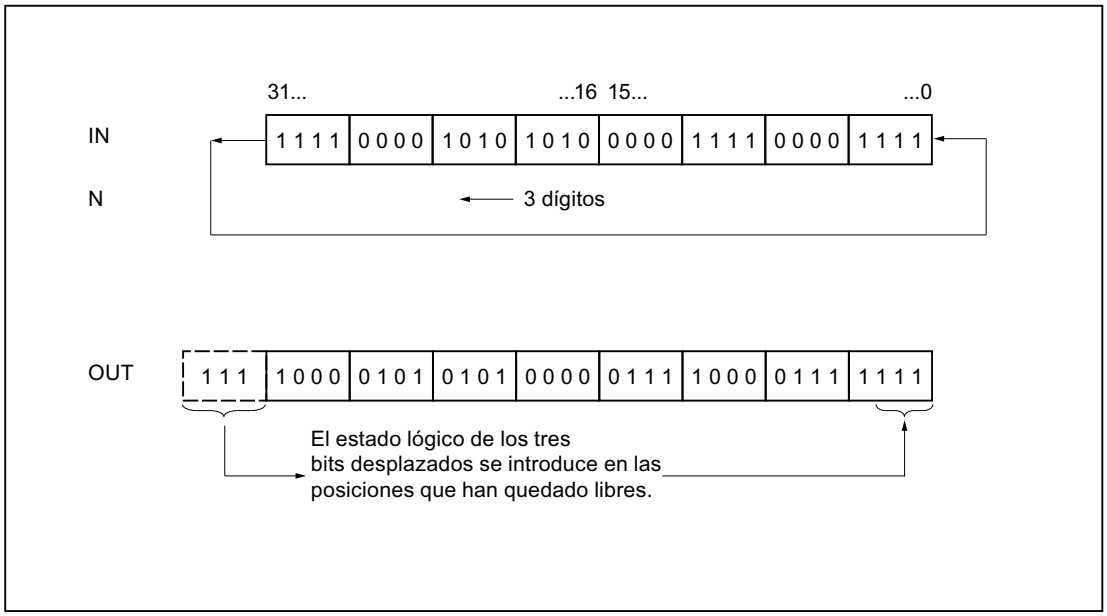
Descripción

La instrucción "Rotar a la izquierda" permite rotar el contenido del operando de la entrada IN de bit en bit hacia la izquierda y consultar el resultado en la salida OUT. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", el valor de la entrada IN se copia en el operando de la salida OUT.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Parámetros

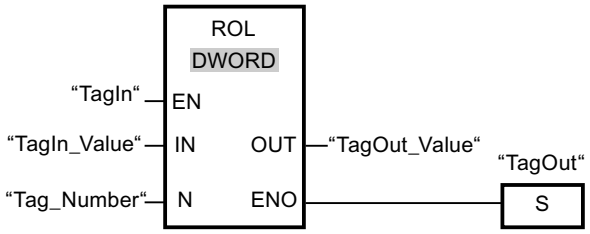
La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, T, C, D, L	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	BYTE, WORD, DWORD, DINT	I, Q, M, D, L, P o constante	Valor que se rota.
N	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de bits que rota el valor.
OUT	Output	BYTE, WORD, DWORD, DINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	TagIn_Value	1111 0000 1010 1010 0000 1111 0000 1111
N	Tag_Number	5
OUT	TagOut_Value	0001 0101 0100 0001 1110 0001 1111 1110

Si la entrada "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Rotar a la izquierda". El contenido del operando "TagIn_Value" se rota cinco posiciones de bit hacia la izquierda. El resultado se emite en la salida "TagOut_Value". Si no se producen errores al ejecutar la instrucción, la salida de habilitación ENO devuelve el estado lógico "1" y se activa la salida "TagOut".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Seleccionar el tipo de datos de un elemento FUP (Página 2726)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Otras instrucciones

---| |---: Consultar si el bit de estado es 1

Descripción

Con la instrucción "Consultar si el bit de estado es 1" es posible consultar el estado lógico de los siguientes bits de estado:

- BR: Resultado binario
- OV: rebase por exceso
- OS: desbordamiento con memoria
- CC 0 y CC 1: Bits de visualización

Qué bits de estado se consultan se determina mediante la selección del símbolo correspondiente en la lista desplegable de la instrucción. La siguiente tabla muestra las posibilidades de consulta de los bits de estado:

Símbolo	Consulta	Explicación
==0	CC 0 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es igual a 0.
<>0	CC 0 = 1 y CC 1 = 0 o CC 0 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es diferente a 0.
>=0	CC 0 = 0	La instrucción consulta si el resultado de una función matemática es mayor o igual que 0.
<=0	CC 1 = 0	La instrucción consulta si el resultado de una función matemática es menor o igual que 0.
>0	CC 0 = 1 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es mayor que 0.
<0	CC 1 = 0 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática es menor que 0.
UO	CC 1 = 1 y CC 1 = 0	La instrucción consulta si el resultado de una función matemática no es válido. La causa de un resultado inválido puede ser un número en coma flotante no válido o una división entre cero.
OV	OV = 1	La instrucción consulta si se ha producido un desbordamiento numérico.

Símbolo	Consulta	Explicación
OS	OS = 1	La instrucción consulta si se ha producido un desbordamiento numérico con memoria en una instrucción.
BR	BR = 1	La instrucción consulta el estado lógico del bit de estado BR.

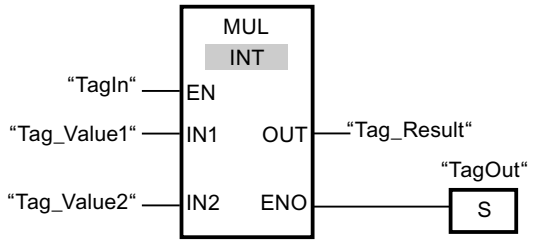
Si la respuesta es afirmativa, la instrucción devuelve el estado lógico "1". Si no se cumple la condición, la instrucción devuelve el estado lógico "0".

Se puede programar una consulta del bit de estado en "0" con ayuda de la instrucción "Invertir RLO".

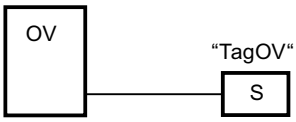
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Segmento 1



Segmento 2



Cuando el operando "TagIn" devuelve el estado lógico "1", se ejecuta la instrucción "Multiplicar" (MUL). Si se produce un desbordamiento durante la ejecución de la instrucción "Multiplicar" se activa el bit de estado OV. Tras la consulta del bit de estado OV, se ejecuta la instrucción "Activar salida" (S) en el segmento 2 y se activa el operando "TagOV".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

CALL: Llamar bloque

Descripción

La instrucción "Llamar bloque" llama una función (FC) sin parámetros. En función de la operación lógica precedente, se puede tratar de una llamada absoluta o condicionada.

En la sección de instrucciones de una función (FC) no puede indicar como operando ningún parámetro del tipo de datos BLOCK_FC en una llamada condicional. En un bloque de función (FB) tampoco puede indiciar como operando un parámetro del tipo BLOCK_FC.

Sólo se ejecutará una llamada condicional cuando el resultado lógico (RLO) sea "1". Si no se ejecuta una llamada condicionada, después de la operación de llamada el RLO es "0". Si se ejecuta la operación, ésta trabaja del modo siguiente:

- Guarda la dirección de retorno del bloque que efectúa la llamada.
- Guarda los dos registros de bloque de datos (bloque de datos y bloque de datos de instancia).
- Actualiza el área de datos locales para la FC o SFC llamada.
- Desplaza el bit MA (bit MCR activo) a la pila de bloques (pila BSTACK).

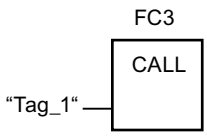
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Llamar bloque":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Bloque>	Output	BLOCK_FC	-	Función llamada

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Se llama la función "FC3" cuando el operando "Tag_1" tiene el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

El MCR (Master Control Relay) (Página 2564)

SAVE: Guardar RLO en bit BR

Descripción

La instrucción "Guardar RLO en bit BR" permite guardar el resultado lógico (RLO) en el resultado binario (BR). Durante la ejecución, la instrucción transfiere el estado lógico del resultado lógico actual al bit de estado BR. La instrucción trabaja independientemente de las condiciones y no influye en ningún otro bit de estado.

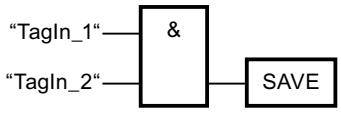
Nota

Tras aplicar la instrucción "Guardar RLO en bit BR", el bit BR puede ser modificado nuevamente por instrucciones posteriores del mismo bloque o de bloques de nivel inferior.

La instrucción "Guardar RLO en bit BR" también se utiliza para controlar el estado de ejecución de un bloque. Si, p. ej., la instrucción "Guardar RLO en bit BR" se programa al final de un bloque, el bit BR se pone al estado lógico del resultado lógico actual del bloque. Si el bloque se ejecuta sin fallos, tanto el bit BR como la salida de habilitación ENO se establecen como resultado lógico "1" en el bloque que efectúa la llamada. El estado lógico "0" se asigna al bit BR si se han producido errores durante el procesamiento del bloque. Mediante consulta del bit BR es posible programar un tratamiento de errores.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



El bit BR se ajusta en el estado lógico "1" si los operandos "TagIn_1" y "TagIn_2" devuelven el estado lógico "1".

Consulte también

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Principios básicos de la palabra de estado (Página 2545)

Ejemplo del mecanismo EN/ENO en FUP (Página 2543)

MCR<: Abrir áreas MCR

Descripción

La instrucción "Abrir áreas MCR" guarda el resultado lógico (RLO) en la pila MCR y cambia la dependencia MCR dentro de un área MCR (MCR = Master Control Relay).

La pila MCR puede contener como máximo 8 entradas y funciona como una memoria intermedia LIFO (Last In, First Out). Cuando la pila está llena, la instrucción genera un mensaje de error (MCRF).

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

El principio y el fin de un área MCR se definen mediante las instrucciones "Activar área MCR" (MCRA) y "Desactivar área MCR" (MCRD). Dentro del área MCR definida es posible activar y desactivar la dependencia MCR para una o varias partes del programa.

La dependencia MCR se activa si el RLO es "1" antes de procesar la instrucción. Si el RLO es "0" antes de la instrucción, la dependencia MCR se desactiva. El final de la sección del programa dependiente de MCR se programa mediante la instrucción "Cerrar áreas MCR". La sección del programa que se encuentra entre las dos instrucciones se ve influenciada por el MCR. Solo están afectadas las siguientes instrucciones dependientes de MCR:

- Asignación
- Activar salida (S)
- Desactivar salida (R)
- Flipflop de desactivación/activación (RS)
- Flipflop de activación/desactivación (SR)
- Copiar valor (MOVE)

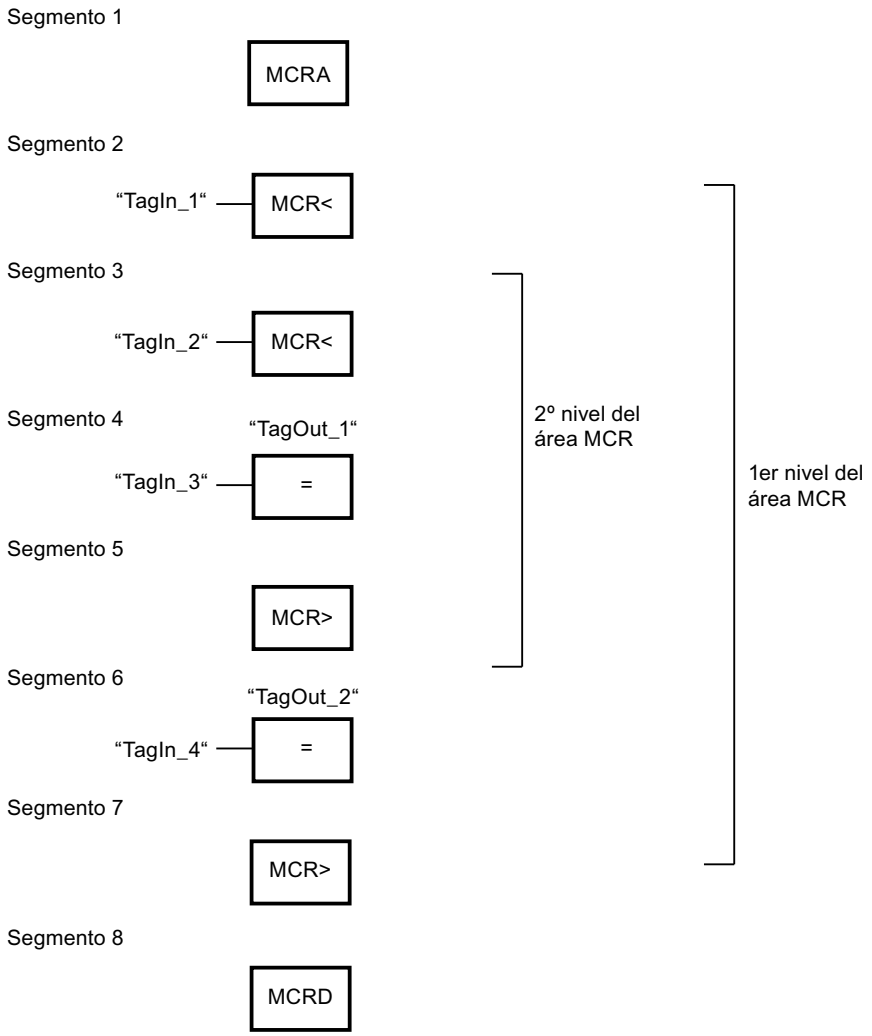
También es posible anidar la activación y desactivación de la dependencia MCR programando la instrucción "Abrir áreas MCR" en una sección del programa dependiente de MCR. Esta instrucción se puede anidar hasta 8 niveles de profundidad. La dependencia MCR de la parte del programa de nivel superior controla las dependencias MCR en todas las partes del programa subordinadas. Si la dependencia MCR en la parte del programa de nivel superior está desactivada, no es posible activarla en una parte subordinada del programa dependiente de MCR.

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

Las instrucciones "Abrir áreas MCR" y "Cerrar áreas MCR" deben programarse siempre de dos en dos. Si una de las dos instrucciones no está programada, se dispara un error de pila MCR.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se conecta la dependencia MCR en el primer nivel del área MCR. En ese caso, el estado lógico del operando "TagIn_4" está asignado al operando "TagOut_2" (segmento 6). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_2". En el segmento 7 se desconecta la dependencia MCR del área MCR.

En el segmento 3 se activa la dependencia MCR en el segundo nivel del área MCR si el operando "TagIn_2" devuelve el estado lógico "1" y el primer nivel del área MCR está activado. En ese caso, el estado lógico del operando "TagIn_3" se asigna al operando "TagOut_1". Si el operando "TagIn_2" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_1". En el segmento 5 se desactiva la dependencia MCR en el segundo nivel del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

MCR>: Cerrar áreas MCR (Página 5988)

MCRA: Activar área MCR (Página 5990)

MCRD: Desactivar área MCR (Página 5992)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

MCR>: Cerrar áreas MCR

Descripción

La instrucción "Cerrar áreas MCR" cierra una sección de programa dependiente de MCR (MCR = Master Control Relay). También borra el resultado lógico (RLO) de la última entrada de la pila MCR.

La pila MCR puede contener como máximo 8 entradas y funciona como una memoria intermedia LIFO (Last In, First Out). Cuando la pila ya está vacía, la instrucción genera un mensaje de error (MCRF).

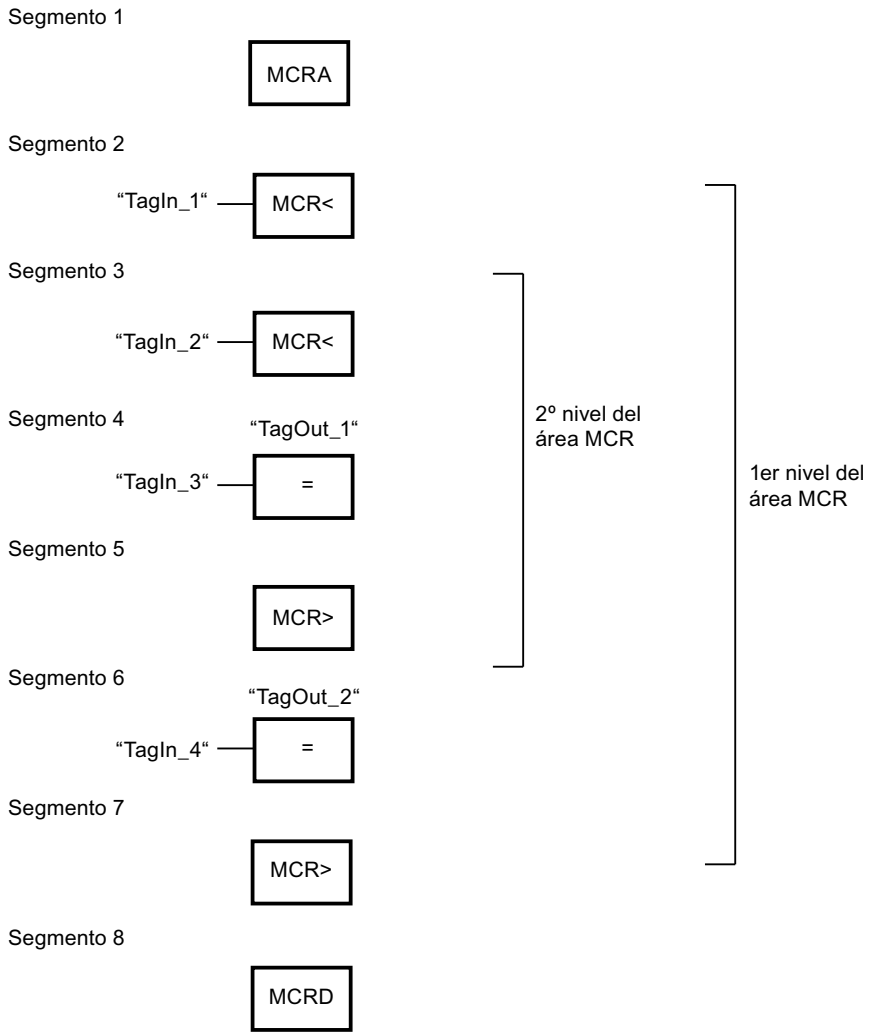
Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

La dependencia MCR se programa con la instrucción "Abrir áreas MCR". Las instrucciones "Abrir áreas MCR" y "Cerrar áreas MCR" deben programarse siempre de dos en dos. Si una de las dos instrucciones no está programada, se dispara un error de pila MCR.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se conecta la dependencia MCR en el primer nivel del área MCR. En ese caso, el estado lógico del operando "TagIn_4" está asignado al operando "TagOut_2" (segmento 6). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_2". En el segmento 7 se desconecta la dependencia MCR del área MCR.

En el segmento 3 se activa la dependencia MCR en el segundo nivel del área MCR si el operando "TagIn_2" devuelve el estado lógico "1". En ese caso, el estado lógico del operando "TagIn_3" se asigna al operando "TagOut_1". Si el operando "TagIn_2" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut_1" independientemente del estado lógico del operando "TagIn_3". En el segmento 5 se desactiva la dependencia MCR en el segundo nivel del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

MCR<: Abrir áreas MCR (Página 5985)

MCRA: Activar área MCR (Página 5990)

MCRD: Desactivar área MCR (Página 5992)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

MCRA: Activar área MCR

Descripción

La instrucción "Activar área MCR" permite definir el principio de un área MCR (MCR = Master Control Relay). Dentro de dicha área es posible activar y desactivar la dependencia MCR con la ayuda de las instrucciones siguientes:

- MCR<: Abrir áreas MCR
- MCR>: Cerrar áreas MCR

El fin del área MCR se define con la instrucción "Desactivar área MCR" (MCRD). Las instrucciones para definir el principio y fin de un área MCR siempre deben utilizarse por parejas.

Nota

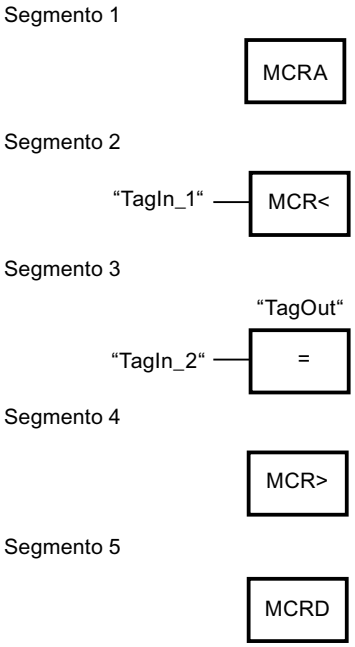
Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

La instrucción se ejecuta sin considerar ni influir en los bits de estado.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se activa la dependencia MCR. En ese caso, el estado lógico del operando "TagIn_2" está asignado al operando "TagOut" (segmento 3). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut" independientemente del estado lógico del operando "TagIn_2". En el segmento 4 se desconecta la dependencia MCR del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

MCR<: Abrir áreas MCR (Página 5985)

MCR>: Cerrar áreas MCR (Página 5988)

MCRD: Desactivar área MCR (Página 5992)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

MCRD: Desactivar área MCR

Descripción

La instrucción "Desactivar área MCR" permite definir el fin de un área MCR (MCR = Master Control Relay).

El principio de un área MCR se define con la instrucción "Activar área MCR" (MCRA). Las instrucciones para definir el principio y fin de un área MCR siempre deben utilizarse por parejas.

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

Dentro de un área MCR es posible programar la dependencia MCR con la ayuda de las instrucciones siguientes:

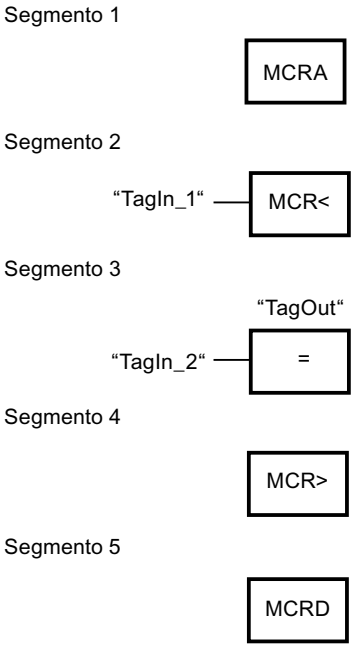
- MCR<: Abrir áreas MCR
- MCR>: Cerrar áreas MCR

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

La instrucción se ejecuta sin considerar ni influir en los bits de estado.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



En el segmento 1 se programa el inicio del área MCR. Si el operando "TagIn_1" devuelve el estado lógico "1", se activa la dependencia MCR. En ese caso, el estado lógico del operando "TagIn_2" está asignado al operando "TagOut" (segmento 3). Si el operando "TagIn_1" tiene el estado lógico "0", se desactiva la dependencia MCR y se pone a "0" el operando "TagOut". En el segmento 4 se desconecta la dependencia MCR del área MCR. En el segmento 8 se programa el final del área MCR.

Consulte también

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

MCR<: Abrir áreas MCR (Página 5985)

MCR>: Cerrar áreas MCR (Página 5988)

MCRA: Activar área MCR (Página 5990)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

SET: Activar área por bits

Descripción

La instrucción "Activar área por bits" permite poner a "1" el estado lógico de los bits en un área definida. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben activarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta sólo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar área por bits":

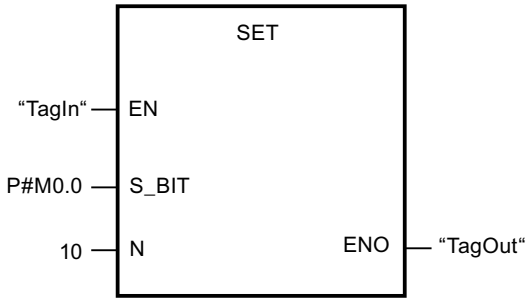
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del área cuyos bits se activan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben activarse

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", la instrucción establece 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

El MCR (Master Control Relay) (Página 2564)

SETP: Activar mapa de bits en el área de periferia

Descripción

La instrucción "Activar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "1". Paralelamente a los bits de periferia, también se activan los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe activarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no se ha conectado periferia para una parte del mapa de bits seleccionado, la instrucción intentará activar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se activa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas.

  ⚠ ADVERTENCIA

  El ajuste de los bits al valor de los bits correspondientes en la memoria imagen de proceso de las salidas puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "1".

La instrucción "Activar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Activar mapa de bits en el área de periferia" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El parámetro N devuelve el valor "0"
- El bit MCR tiene el estado lógico "1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben activarse

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero hacia el primer bit que debe activarse

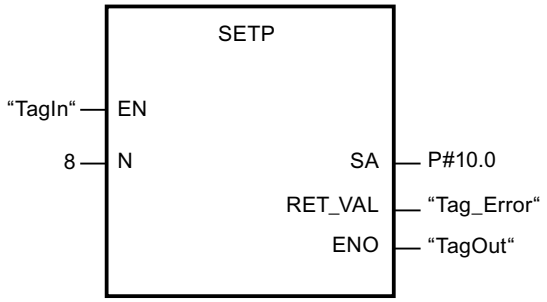
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción activa 8 bits en el área de periferia de acuerdo con las salidas Q10.0 a Q10.7.

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SETI: Activar área por bytes

Descripción

La instrucción "Activar área por bytes" permite poner a "1" el estado lógico de los bits en un área definida de bytes. El punto inicial del área se define en el parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben activarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte. Por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas (Q) se pone a "0".

La llamada de la instrucción "Activar área por bytes" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa (P).
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

Parámetros

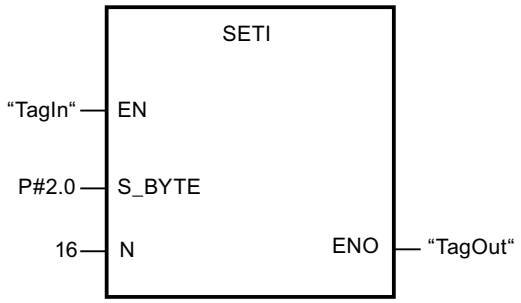
La tabla siguiente muestra los parámetros de la instrucción "Activar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se activan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área que debe activarse El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción activa 2 bytes (16 bits) en el área de P2.0 a P3.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

El MCR (Master Control Relay) (Página 2564)

RESET: Desactivar área por bits

Descripción

La instrucción "Desactivar área por bits" permite poner a "0" el estado lógico de los bits en un área definida. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben desactivarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta sólo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

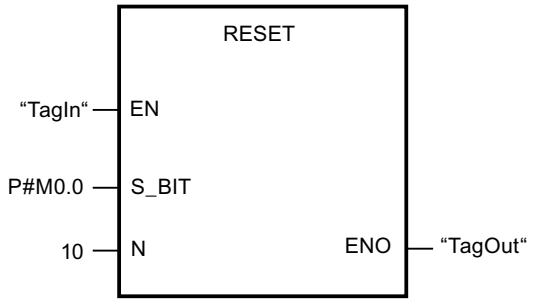
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
S_BIT	Input	POINTER	I, Q, M, D	Señala al primer bit del área cuyos bits se desactivan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben desactivarse

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Cuando el operando "TagIn" devuelve el estado lógico "1", la instrucción desactiva 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

El MCR (Master Control Relay) (Página 2564)

RESETP: Desactivar mapa de bits en el área de periferia

Descripción

La instrucción "Desactivar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "0". Simultáneamente con los bits de periferia se desactivan también los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe desactivarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no hay periferia enchufada para una parte del mapa de bits seleccionado, la instrucción intentará desactivar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se desactiva por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas.

⚠ ADVERTENCIA
El ajuste de los bits al valor de los bits correspondientes en la memoria imagen de proceso de las salidas puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "0".

La instrucción "Desactivar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Desactivar mapa de bits en el área de periferia" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El parámetro N devuelve el valor "0"
- El bit MCR tiene el estado lógico "1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben desactivarse.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero al primer bit que debe desactivarse.

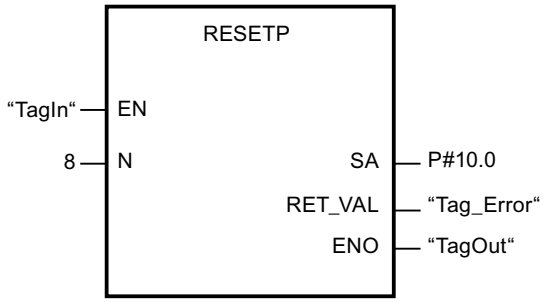
Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción desactiva 8 bits en el área de periferia de acuerdo con las salidas Q10.0 a Q10.7.

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RESETI: Desactivar área por bytes

Descripción

La instrucción "Desactivar área por bytes" permite poner a "0" el estado lógico de los bits en un área definida de bytes. El punto inicial del área se define mediante el puntero del parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben desactivarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y

palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte. Por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas (Q) se pone a "0".

La llamada de la instrucción "Desactivar área por bytes" no tiene efecto alguno y la salida de habilitación ENO se pone a "0" si se cumple una de las condiciones siguientes:

- La entrada de habilitación EN devuelve el estado lógico "0".
- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa (P).
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

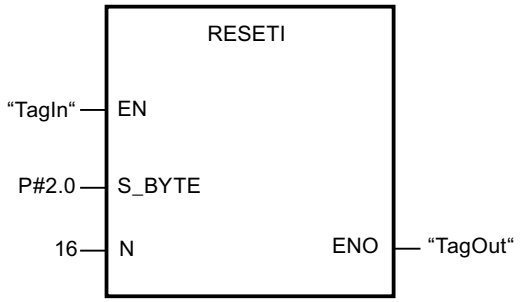
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se desactivan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área cuyos bits se desactivan. El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Si el operando "TagIn" devuelve el estado lógico "1", la instrucción desactiva 2 bytes (16 bits) en el área de P2.0 a P3.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

El MCR (Master Control Relay) (Página 2564)

REPL_VAL: Introducir valor sustitutivo

Descripción

La instrucción "Introducir valor sustitutivo" permite introducir un valor sustitutivo en el acumulador 1 (ACU 1) desde un bloque de organización de error síncrono.

Nota

La instrucción "Introducir valor sustitutivo" sólo se puede llamar desde bloques de organización de error síncrono (OB 121, OB 122).

La instrucción "Introducir valor sustitutivo" se utiliza si ya no es posible leer ningún valor de un módulo. Tras cada acceso a dicho módulo se inicia el OB 122. Llamando la instrucción "Introducir valor sustitutivo" es posible cargar un valor sustitutivo en el acumulador 1. La ejecución del programa se prosigue entonces con este valor sustitutivo. La información necesaria para seleccionar el valor sustitutivo puede consultarse en las variables locales del OB 122.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Introducir valor sustitutivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
VAL	Input	DWORD	I, Q, M, D, L o constante	Valor sustitutivo
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#....)	Descripción
0000	Ningún error
8080	La instrucción no se ha llamado desde un OB de error síncrono.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite ocupar los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de éste es verdadero y una vez transcurrido el tiempo programado para el paso actual,

o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento (i); el estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	I, Q, M, D, L, P	Paso predeterminado del circuito secuencial sucesivo

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DSC	Static	BYTE	I, Q, M, D, L, P	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P	Valor de contaje actual del circuito secuencial sucesivo
DTBP	Static	WORD	I, Q, M, D, L, P	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	Hora de sistema anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM_X: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite materializar un circuito secuencial sucesivo con 16 pasos como máximo. El número del primer paso se define con el parámetro DSP. El número del último paso se predetermina con el parámetro LST_STEP.

En cada paso se escriben todos los 16 bits de salida OUT0 a OUT15 y el parámetro de salida OUT_WORD (que contiene los bits de salida). A un bit de salida se asigna bien sea el bit correspondiente del campo predefinido OUT_VAL, o bien el bit de salida del paso anterior. El valor que se asigne dependerá de la ocupación de los bits de máscara en el parámetro S_MASK.

Nota

El ajuste predeterminado de los bits de máscara es 0. Para modificar el ajuste actual de uno o varios bits de máscara, realice el cambio en el bloque de datos de instancia.

La instrucción "Materializar circuito secuencial sucesivo" pasa al siguiente paso cuando se produce un flanco ascendente en el parámetro JOG. Si la instrucción ya se encuentra en el último paso y se produce un flanco ascendente en el parámetro JOG, se activan las variables Q y EOD. En este caso, el parámetro DCC se pone al valor 0. La instrucción permanece en el último paso hasta que el parámetro RESET se pone al valor "1".

Además, también se puede permitir la transición al paso siguiente en función del tiempo. Para ello se debe poner al valor "1" el parámetro DRUM_EN. La transición al paso siguiente tiene lugar si está activado el bit de evento (EVENTi) correspondiente al paso actual y si ha transcurrido el tiempo predeterminado para el paso actual. El tiempo necesario para un paso resulta de la multiplicación de la base de tiempo (DTBP por el factor de tiempo válido para el paso actual (S_PRESET).

Nota

El tiempo de ejecución restante en el paso actual (DCC) se reduce únicamente si está activado el bit de evento correspondiente (EVENTi).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si, al llamar la instrucción, el parámetro RESET tiene el valor "1", el circuito secuencial sucesivo pasará al paso cuyo número se haya asignado al parámetro DSP.

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1".

El circuito secuencial sucesivo también se puede conmutar en progresión en todo momento a través del valor del parámetro JOG.

En la primera llamada de la instrucción, el parámetro RESET se debe poner a 1.

Si el circuito secuencial se encuentra en el último paso y ha transcurrido el tiempo de ejecución predefinido para este paso, se activan los parámetros Q y EOD. La instrucción permanece en el último paso hasta que el parámetro RESET adopte el valor "1".

Esta instrucción se ejecuta sólo en los estados operativos ARRANQUE y RUN.

El sistema operativo inicializa la instrucción "Materializar circuito secuencial sucesivo" cuando se efectúa un arranque en frío. La instrucción no se inicializa en caso de un rearranque completo (en caliente). Si la instrucción "Materializar circuito secuencial sucesivo" debe inicializarse tras un rearranque completo (en caliente), es preciso llamarla en el OB 100 con el valor "1" en el parámetro RESET.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" desactiva el circuito secuencial. En la primera llamada de la instrucción, el parámetro RESET se debe poner a "1". En el parámetro RESET también se pueden indicar constantes.
JOG	Input	BOOL	I, Q, M, D, L	Un flanco ascendente (con respecto a la última llamada) provoca la transición al paso siguiente, siempre que el circuito secuencial no se encuentre aún en el último paso. La transición tiene lugar independientemente del valor asignado al parámetro DRUM_EN.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DRUM_EN	Input	BOOL	I, Q, M, D, L	Parámetro de control que determina si se puede realizar una transición al paso siguiente en función del tiempo. Si el valor es "1", es posible una transición en función del tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso Los valores admisibles son 1 a 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento n.º i (pertenece al paso i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	I, Q, M, D, L	Bit de salida j
Q	Output	BOOL	I, Q, M, D, L	Parámetro de estado que indica si ha transcurrido el tiempo predefinido por el usuario para la ejecución del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Bits de salida resumidos en una variable
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Irrelevante para el usuario (parámetro de entrada JOG de la llamada anterior)
EOD	Static	BOOL	I, Q, M, D, L	Idéntico al parámetro de salida Q
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Número del primer paso Los valores admisibles son 1 a 16.
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Número del paso actual
DCC	Static	DWORD	I, Q, M, D, L, P	Tiempo de ejecución restante del paso actual, indicado en milisegundos (ms) (relevante sólo si DRUM_EN = 1 y el bit de evento correspondiente = 1)
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo en milisegundos (ms) válida para todos los pasos
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	(Irrelevante para el usuario: hora de sistema de la llamada anterior)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Campo unidimensional con los factores de tiempo para cada paso. Selección recomendada de índices: [1 a 16]. En este caso, S_PRESET [x] contiene el factor de tiempo del paso x.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los valores que deben emitirse en cada paso, a no ser que se hayan inhibido vía S_MASK. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, OUT_VAL [x, y] contiene el valor asignado al bit de salida OUTy en el paso x.
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los bits de máscara para cada paso. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, S_MASK [x, y] contiene el bit de máscara para el valor "y"-ésimo que debe emitirse en el paso x. Significado de los bits de máscara: • 0: el valor correspondiente del paso anterior se asigna al respectivo bit de salida • 1: el valor correspondiente de OUT_VAL se asigna al respectivo bit de salida.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE* (W#16#...)	Explicación
0000	Ningún error
8081	Valor no admisible en el parámetro LST_STEP
8082	Valor no admisible en DSC
8083	Valor no admisible en DSP
8084	El producto DCC = DTBP * S_PRESET[DSC] excede el valor 2**31-1 (aprox. 24,86 días).
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET, OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1"
  - Los parámetros ET, OA y CA se ponen a "0"
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET, OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y C_FB devuelven el estado lógico "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
CMD	Input	BOOL	I, Q, M, D, L	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms.
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms.
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
CMD_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución de CMD

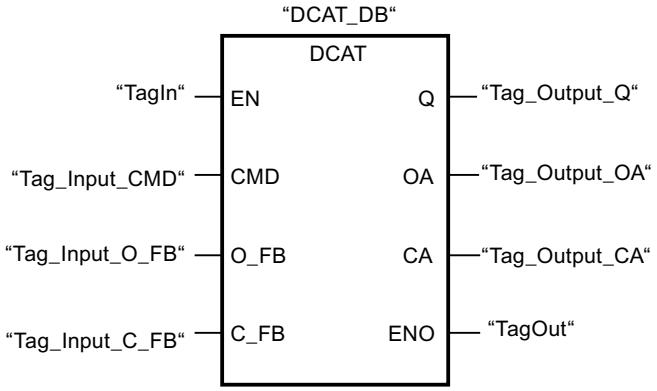
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se activa la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control del motor" no devuelve información de error.

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
O_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error.
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
O_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Abrir"
C_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Cerrar"

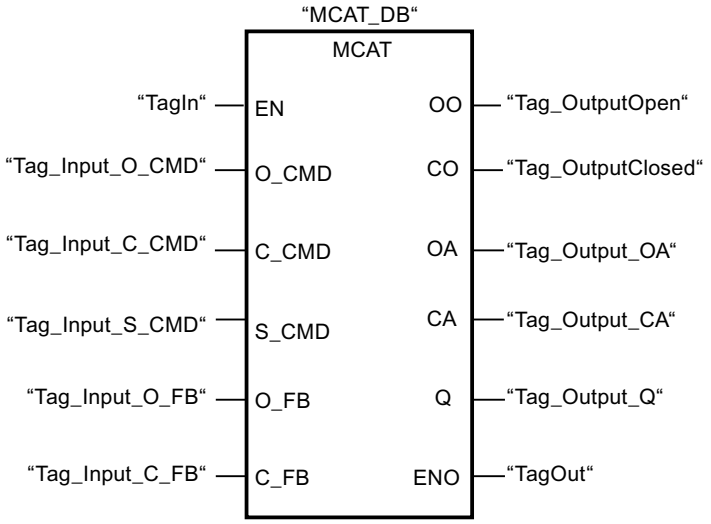
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una coincidencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada

del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica sólo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
LAST	Static	BYTE	I, Q, M, D, L, P	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY of WORD	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor para la amplificación en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo dinámico. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse para suprimir perturbaciones o como filtro pasa bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas

frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

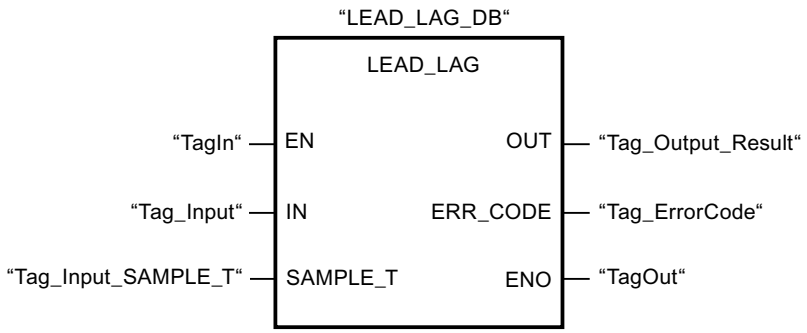
Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.



Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

Los siguientes valores están almacenados en el bloque de datos de instancia "LEAD_LAG_DB" de la instrucción:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos de instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

TONR_X: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" permite acumular valores de tiempo dentro de un periodo definido por el parámetro PT. Si el estado lógico del parámetro TMR_EN cambia a "1", se ejecuta la instrucción y se empieza a contar el tiempo PT. La instrucción "Acumulador

de tiempo" suma el valor del parámetro DELTA_T al del parámetro ET siempre que se cumplan las siguientes condiciones:

- El parámetro TMR_EN devuelve el estado lógico "1".
- El valor de tiempo del parámetro ET es menor que el del parámetro PT.
- El parámetro RESET devuelve el estado lógico "0".

Si el valor del parámetro TMR_EN cambia a "0", el tiempo del parámetro DELTA_T se dejará de sumar al valor de tiempo del parámetro ET.

Cuando el valor de tiempo del parámetro ET alcanza el del parámetro PT, el parámetro Q devuelve el estado lógico "1". Mientras el parámetro Q devuelva el estado lógico "1", el valor del parámetro ET no se modificará. El valor del parámetro ET se pone a "0" cuando el parámetro RESET se pone a "1".

Puesto que la instrucción "Acumulador de tiempo" utiliza el tiempo de ejecución (DELTA_T) del último ciclo del bloque de organización (OB), la instrucción solo puede llamarse en bloques de organización cíclicos.

Nota

El tiempo de ciclo del bloque de organización debe desplazarse desde la información de arranque a la variable global del parámetro DELTA_T.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
TMR_EN	Input	BOOL	I, Q, M, D, L	Activa el tiempo para acumular el valor de tiempo
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
PV	Input	DINT	I, Q, M, D, L, P o constante	Duración máxima del cronometraje
DELTA_T	Input	INT	I, Q, M, D, L o constante	Tiempo de ciclo del OB del ciclo anterior
Q	Output	BOOL	Q, M, D, L	Salida que se activa una vez transcurrido el tiempo PT.
ET	InOut	DINT	I, Q, M, D, L	Valor de tiempo actual del tiempo transcurrido

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

WSR: Guardar datos en el registro de desplazamiento

Descripción

La instrucción "Guardar datos en el registro de desplazamiento" permite escribir datos en un registro de desplazamiento desde el origen indicado. Todo elemento del registro se desplaza a la siguiente dirección. Los datos contenidos en la última dirección del registro de desplazamiento se pierden tras la operación de desplazamiento.

Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen (S_DATA) y se desplazan a la dirección inicial (START) del registro de desplazamiento. El parámetro LENGTH indica el número de elementos que deben desplazarse. El tipo de datos de los elementos se define en el parámetro E_TYPE. Si se indica un tipo de datos no admisible en el parámetro E_TYPE, no se procesa la instrucción.

Si el parámetro RESET está a "1" al ejecutarse la instrucción, se borra el registro. El parámetro Q se pone al estado lógico "1" si el registro de desplazamiento está ocupado con ceros o si se borra.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Guardar datos en el registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_DATA	Input	POINTER	I, Q, M, D	Apunta al elemento de origen que se desplaza a la dirección inicial (START). En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
START	Input	POINTER	I, Q, M, D	Apunta a la dirección inicial del registro. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
LEN	Input	WORD	I, Q, M, D, L, P	Número de elementos que deben desplazarse.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de los elementos que deben desplazarse. Los siguientes tipos de datos son admisibles: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	Q, M, D, L	Indica "0" si el parámetro RESET está activado (1) o si todos los elementos que deben desplazarse tienen el valor "0".

Ejemplo

En el ejemplo siguiente se desplazan cuatro elementos del tipo de datos WORD a partir de la dirección indicada en el parámetro START. Una vez desplazado el primer elemento del registro a la siguiente dirección, la primera dirección se ocupará con los datos a los que señala

el puntero en el parámetro S_DATA. El valor del último elemento del registro se pierde entonces.

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SHRB: Desplazar bit al registro de desplazamiento

Descripción

La instrucción "Desplazar bit al registro de desplazamiento" permite escribir un bit en un registro de desplazamiento desde el origen indicado (DATA). Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen y se desplazan a la dirección inicial (S_BIT) del registro de desplazamiento. Los restantes bits se desplazan una posición. El bit de la última dirección (S_BIT + N) se pierde tras el desplazamiento. Si el parámetro RESET está a "1" mientras se ejecuta la instrucción, las direcciones del registro de desplazamiento se ponen a "0" y no se desplazan.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar bit al registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
DATA	Input	BOOL	I, Q, M, D, L	Bit de origen
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_BIT	Input	POINTER	I, Q, M, D	Apunta al bit inicial del registro de desplazamiento En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	WORD	I, Q, M, D, L, P	Número de bits que deben desplazarse.

Ejemplo

En el ejemplo siguiente, el parámetro N se pone al valor 14 ("E" en notación hexadecimal), con lo que 14 bits se desplazan a partir de la dirección indicada en el parámetro S_BIT. Una

vez desplazados los bits, la primera dirección se ocupa con los datos del origen del parámetro DATA. El último valor de bit se pierde.

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Antes de ejecutarse la instrucción, los siguientes estados lógicos están almacenados en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Después de la ejecución

Una vez ejecutada la instrucción, los siguientes estados lógicos se almacenan en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

SEG: Generar patrón de bits para el display de 7 segmentos

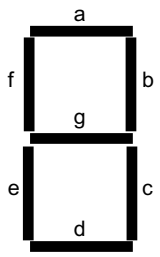
Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" permite convertir cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se deposita en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6
0111	00000111	7
1000	01111111	8
1001	01100111	9
1010	01110111	A
1011	01111100	B
1100	00111001	C
1101	01011110	D
1110	01111001	E
1111	01110001	F

Display de siete segmentos



Parámetro

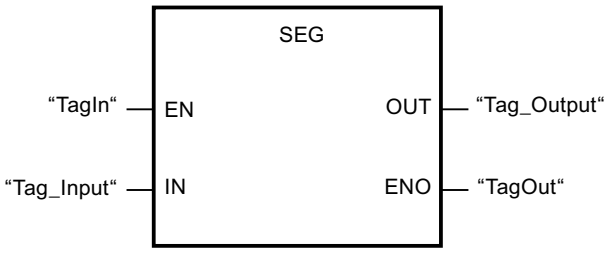
La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits para el display de 7 segmentos

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	00000110 01011011 01001111 01100110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

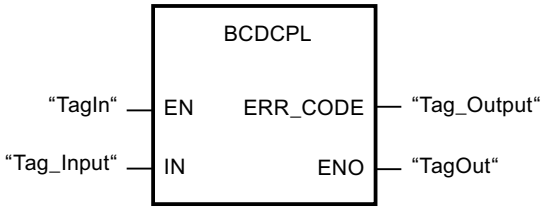
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" permite contar el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

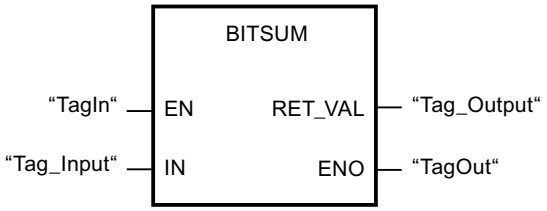
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN	Input	BOOL	I, Q, M, D, L, T, C	Entrada de habilitación
ENO	Output	BOOL	I, Q, M, D, L	Salida de habilitación
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos del mecanismo EN/ENO (Página 2539)

Insertar elementos FUP (Página 2722)

Editar elementos FUP (Página 2738)

Cablear operandos en instrucciones FUP (Página 2745)

Cambiar formatos de visualización en el estado del programa (Página 3100)

AWL

Temporizadores

TP: Impulso

Descripción

La instrucción "Impulso" activa un operando por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Al iniciar la instrucción empieza a contarse el periodo programado en el parámetro (PT). El parámetro Q se activa por el tiempo PT, independientemente de cómo evolucione la señal en el parámetro IN. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico del parámetro Q mientras transcurra el tiempo PT.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo PT y si el estado lógico de la entrada IN es "0", se desactiva la salida ET.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

En el código del programa, la instrucción "Impulso" se debe llamar con la instrucción "Llamar bloque" (CALL).

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TP (p. ej., "TP_DB")
- Declaración como variable local del tipo TP en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la

carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Impulso" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Impulso", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Impulso" se utiliza la siguiente sintaxis:

```
CALL TP, "Temporizador CEI"
TIME
IN := <Operando>
PT := := <Operando>
Q := := <Operando>
ET := <Operando>
```

Parámetros

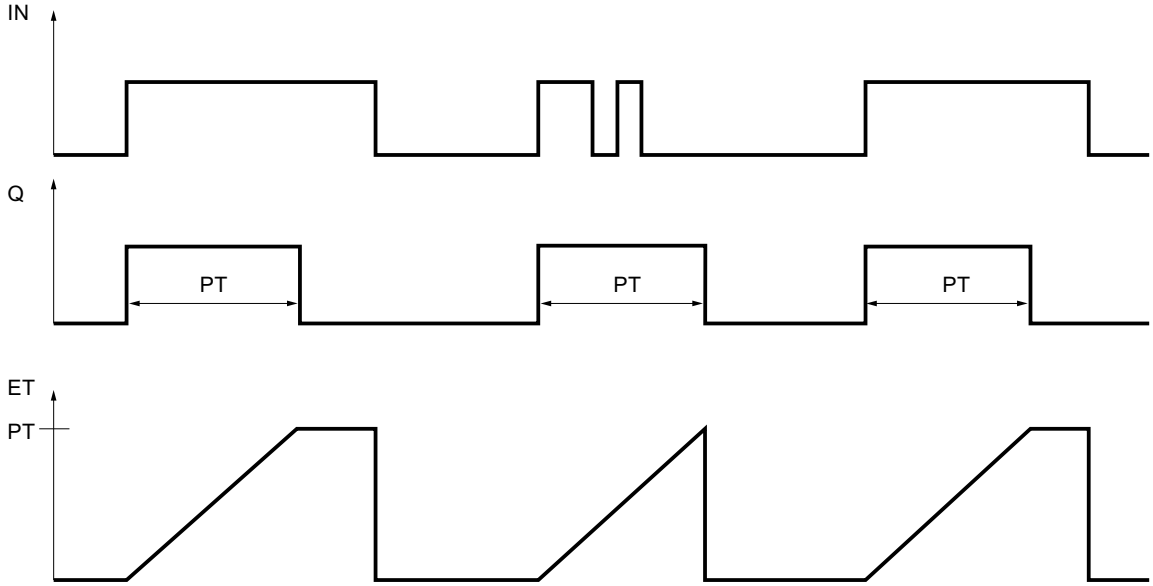
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Duración del impulso. El valor indicado en el parámetro PT debe ser positivo
Q	Output	BOOL	I, Q, M, D, L	Salida de impulso
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Impulso" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TP, "TP_DB"	// Se llama la instrucción "Impulso". El bloque de datos "TP_DB" se asigna a la instrucción.
TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// Duración del impulso
Q := "Tag_Status"	// El operando se activa durante el periodo determinado por el operando "Tag_PresetTIME".
ET := "Tag_Elapsed"	// Valor de tiempo actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

TON: Retardo al conectar

Descripción

La instrucción "Retardo al conectar" retarda un flanco de señal ascendente por el tiempo PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. Una vez transcurrido el tiempo PT, el parámetro Q lleva el estado lógico "1". El operando del parámetro Q permanecerá activado mientras la entrada de arranque IN siga a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva el parámetro Q. La función de temporización se reinicia al detectarse un flanco de señal ascendente nuevo en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez alcanzado este tiempo PT y si el estado lógico de la entrada IN es "0", se desactiva la salida ET.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

En la lógica del programa, la instrucción "Retardo al conectar" se debe llamar con la instrucción "Llamar bloque" (CALL).

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TON (p. ej., "TON_DB")
- Declaración como variable local del tipo TON en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al conectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al conectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Retardo al conectar" se utiliza la siguiente sintaxis:

```
CALL TON, "Temporizador CEI"
TIME
IN := <Operando>
PT := := <Operando>
Q := := <Operando>
ET := <Operando>
```

Parámetros

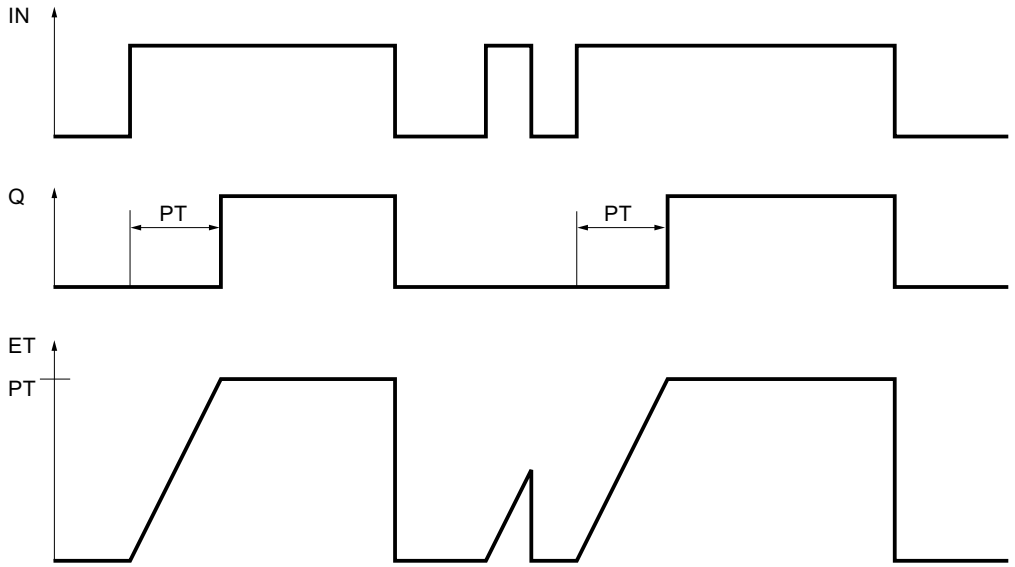
La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor indicado en el parámetro PT debe ser positivo
Q	Output	BOOL	I, Q, M, D, L	Estado lógico que se retarda por el tiempo PT
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Retardo al conectar" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TON, "TON_DB"	// Se llama la instrucción "Retardo al conectar". El bloque de datos "TON_DB" se asigna a la instrucción.
TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// Indica el tiempo en el que se retarda el flanco de señal ascendente en el parámetro IN.
Q := "Tag_Status"	// El operando se activa una vez transcurrido el tiempo PT predeterminado por la variable "Tag_PresetTIME".
	// El parámetro Q permanecerá activado mientras la variable "Tag_Start" siga a "1".
	// Cuando el estado lógico de la entrada de arranque cambie de "1" a "0", se desactivará el operando en el parámetro Q .
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

TOF: Retardo al desconectar

Descripción

La instrucción "Retardo al desconectar" retarda un flanco de señal descendente por el tiempo PT. El operando del parámetro Q se activa cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico del parámetro IN cambia nuevamente a "0", el tiempo programado PT deja de contar. El operando del parámetro Q permanece activado mientras transcurre el tiempo PT. Una vez transcurrido el tiempo PT se inicializa el parámetro Q. Si el estado lógico del parámetro IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico del parámetro Q permanece puesto a "1".

El valor actual de tiempo se puede consultar en el parámetro ET. El valor de tiempo actual empieza a partir de T#0s y termina al alcanzarse el valor de tiempo PT. Una vez transcurrido el tiempo PT, el parámetro ET permanecerá al valor actual hasta que el parámetro IN vuelva a cambiar a "1".

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

En la lógica del programa, la instrucción "Retardo al desconectar" se debe llamar con la instrucción "Llamar bloque" (CALL).

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TOF (p. ej., "TOF_DB")
- Declaración como variable local del tipo TOF en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al desconectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque

diferente contiene instancias de la instrucción "Retardo al desconectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Retardo al desconectar" se utiliza la siguiente sintaxis:

```
CALL TOF, "Temporizador CEI"
TIME
IN := <Operando>
PT := := <Operando>
Q := := <Operando>
ET := <Operando>
```

Parámetros

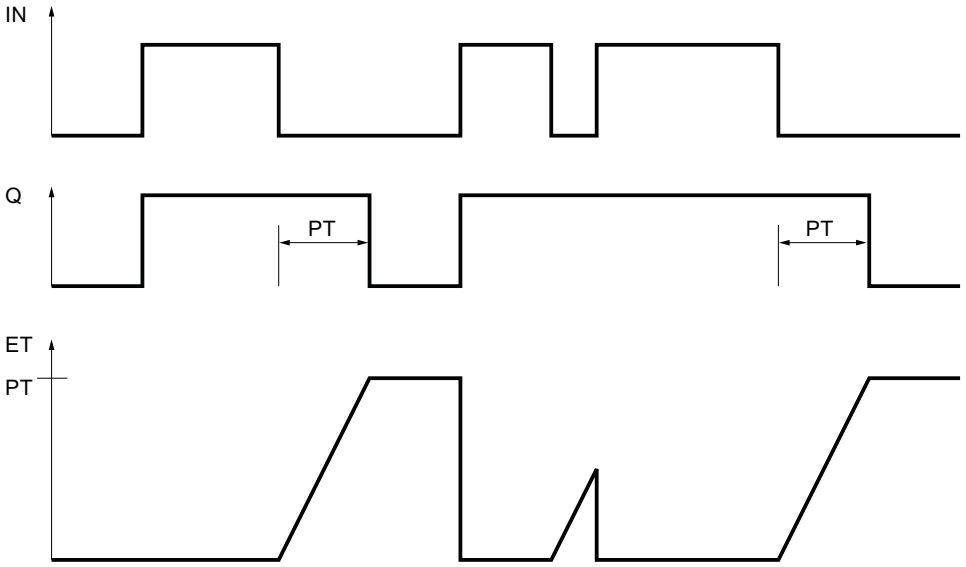
La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor indicado en el parámetro PT debe ser positivo
Q	Output	BOOL	I, Q, M, D, L	Estado lógico que se retarda por el tiempo PT
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el comportamiento de la instrucción "Retardo al desconectar" tras el arranque:



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TOF, "TOF_DB"	// Se llama la instrucción "Retardo al desconectar". El bloque de datos "TOF_DB" se asigna a la instrucción.
TIME	// Tipo de datos de la instrucción
IN := "Tag_Start"	// Cuando se produce un flanco de señal ascendente en el operando, se ejecuta la instrucción.
PT := "Tag_PresetTIME"	// Indica el tiempo en el que se retarda el flanco de señal descendente en el parámetro IN.
Q := "Tag_Status"	// El operando se activa cuando la instrucción se inicia debido a un flanco de señal ascendente en el parámetro IN.
	// Si el estado lógico del parámetro IN cambia de "1" a "0", el operando "Tag_Status" permanecerá activado mientras transcurra el tiempo predeterminado por la variable "Tag_PresetTIME".
	// Una vez transcurrido el tiempo del parámetro PT, se desactiva el operando.
ET := "Tag_ElapsedTIME"	// Valor de tiempo actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Contadores

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor de contaje del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor de contaje actual del parámetro CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos INT. Cuando se alcanza el valor límite superior, el estado lógico del parámetro CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. El estado lógico del parámetro Q es determinado por el parámetro PV. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0". En el parámetro PV también se puede indicar una constante.

El valor del parámetro CV se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico del parámetro CU no tendrá efecto alguno en la instrucción.

En la lógica del programa, la instrucción "Contador ascendente" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guardan los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTU (p. ej. "CTU_DB")
- Declaración como variable local del tipo CTU en la sección "Static" de un bloque (p. ej., #MyCTU_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se debe depositar en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la

carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro R de la instrucción. Si un bloque diferente contiene instancias de la instrucción "Contador ascendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

```
CALL CTU, "Contador CEI"
<Tipo de datos>
CU := <Operando>
R := <Operando>
PV := <Operando>
Q := <Operando>
CV := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, P	Entrada de reset
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL CTU, "CTU_DB"	// Se llama la instrucción "Contador ascendente". El bloque de datos "CTU_DB" se asigna a la instrucción.
INT	// Tipo de datos de la instrucción

AWL	Explicación
CU := "Tag_StartCTU"	// Cuando el estado lógico del operando "Tag_StartCTU" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se incrementa en uno. // El valor de contaje sigue incrementando hasta alcanzar el valor límite superior de INT = 32767.
R := "Tag_ResetCounter"	// Cuando el estado lógico del operando "Tag_ResetCounter" cambia a "1", el operando "Tag_CounterValue" se pone a "0".
PV := "Tag_PresetValue"	// El operando determina a partir de qué valor debe activarse el operando del parámetro Q.
Q := "Tag_CounterStatus"	// El operando permanece activado mientras el valor de contaje actual sea mayor o igual al valor del parámetro PV.
CV := "Tag_CounterValue"	// Valor de contaje actual

Consulte también

Procesar instrucciones AWL (Página 2782)

Añadir instrucciones AWL (Página 2767)

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor de contaje del parámetro CV. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor de contaje actual del parámetro CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos INT indicado. Cuando se alcanza el valor límite inferior, el estado lógico del parámetro CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. Si el valor de contaje actual es menor o igual a cero, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0".

El valor del parámetro CV se pone al valor del parámetro PV cuando el estado lógico del parámetro LD cambia a "1". Mientras el parámetro LD tenga el estado lógico "1", el estado lógico del parámetro CD no tendrá efecto alguno en la instrucción.

En la lógica del programa, la instrucción "Contador descendente" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTD (p. ej., "CTD_DB")
- Declaración como variable local del tipo CTD en la sección "Static" de un bloque (p. ej., #MyCTD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se debe depositar en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo resetea las instancias de la instrucción "Contador descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro LD de la instrucción. En el parámetro PV se indica en este caso el valor inicial deseado para el parámetro CV. Si un bloque diferente contiene instancias de la instrucción "Contador descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

```
CALL CTD, "Contador CEI"
<Tipo de datos>
CD := <Operando>
LD := <Operando>
PV := <Operando>
Q := <Operando>
CV := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL CTD, "CTD_DB"	// Se llama la instrucción "Contador descendente". El bloque de datos "CTD_DB" se asigna a la instrucción.
INT	// Tipo de datos de la instrucción
CD := "Tag_StartCTD"	// Cuando el estado lógico del operando "Tag_StartCTD" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se decrementa en uno.
	// El valor de contaje del parámetro CV se sigue decrementando hasta alcanzar el valor límite inferior de INT = -32768.
LD := "Tag_LoadPV"	// Cuando el estado lógico del operando "Tag_LoadPV" cambia a "1", el operando "Tag_CounterValue" se pone al valor del operando "Tag_PresetValue".
PV := "Tag_PresetValue"	// Indica el valor al que se pone el contador cuando el estado lógico del parámetro LD es "1".
Q := "Tag_CounterStatus"	// El operando se activa si el valor de contaje actual es menor o igual a cero.
CV := "Tag_CounterValue"	// Valor de contaje actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual del parámetro CV se incrementa en uno. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje del parámetro CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje actual no cambia en el parámetro CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado INT. Una vez alcanzado el valor límite superior, el valor de contaje ya no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado INT, ya no se decrementa el valor de contaje.

Cuando el estado lógico del parámetro LD cambia a "1", el valor de contaje del parámetro CV se pone al valor del parámetro PV. Mientras el parámetro LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", un cambio del estado lógico de los parámetros CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en el parámetro QU. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro QU adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QU es "0". En el parámetro PV también se puede indicar una constante.

El estado del contador descendente se puede consultar en el parámetro QD. Si el valor de contaje actual es menor o igual a cero, el parámetro QD adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QD es "0".

En la lógica del programa, la instrucción "Contador ascendente - descendente" se debe llamar con la instrucción "Llamar bloque" (CALL).

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTUD (p. ej., "CTUD_DB")
- Declaración como variable local del tipo CTUD en la sección "Static" de un bloque (p. ej., #MyCTUD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si el contador CEI se debe depositar en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la

carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo resetea las instancias de la instrucción "Contador ascendente - descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con los siguientes valores de parámetros:

- Si se utiliza como contador ascendente, el valor del parámetro R debe ponerse a "1".
- Si se utiliza como contador descendente, el valor del parámetro LD debe ponerse a "1". En el parámetro PV debe indicarse en este caso el valor inicial deseado del parámetro CV.

Si un bloque diferente contiene instancias de la instrucción "Contador ascendente - descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador ascendente - descendente" se utiliza la siguiente sintaxis:

```
CALL CTUD, "Contador CEI"
<Tipo de datos>
CU := <Operando>
CD := <Operando>
R := <Operando>
LD := <Operando>
PV := <Operando>
QU := <Operando>
QD := <Operando>
CV := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, P	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
QU	Output	BOOL	I, Q, M, D, L	Estado del contador ascendente
QD	Output	BOOL	I, Q, M, D, L	Estado del contador descendente
CV	Output	INT	I, Q, M, D, L, P	Valor de contaje actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL CTUD, "CTUD_DB"	// Se llama la instrucción "Contador ascendente - descendente". El bloque de datos "CTUD_DB" se asigna a la instrucción.
INT	// Tipo de datos de la instrucción
CU := "Tag_StartCTU"	// Cuando el estado lógico del operando "Tag_StartCTU" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se incrementa en uno. // Si se detecta un flanco ascendente en el parámetro CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior de INT = 32767.
CD := "Tag_StartCTD"	// Cuando el estado lógico del operando "Tag_StartCTD" cambia de "0" a "1", se ejecuta la instrucción y el valor actual del operando "Tag_CounterValue" se decrementa en uno. // El valor de contaje del parámetro CV se sigue decrementando hasta alcanzar el valor límite inferior de -32768.
R := "Tag_ResetCounter"	// Cuando el estado lógico del operando "Tag_ResetCounter" cambia a "1", el operando "Tag_CounterValue" se pone a "0".
LD := "Tag_LoadPV"	// Cuando el estado lógico del operando "Tag_LoadPV" cambia a "1", el operando "Tag_CounterValue" se pone al valor del operando "Tag_PresetValue".
PV := "Tag_PresetValue"	// Indica el valor al que se pone el contador cuando el estado lógico del parámetro LD es "1".
QU := "Tag_CounterStatus"	// El operando permanece activado mientras el valor de contaje actual sea mayor o igual al valor del parámetro PV.
QD := "Tag_CounterStatus"	// El operando se activa si el valor de contaje actual es menor o igual a cero.
CV := "Tag_CounterValue"	// Valor de contaje actual

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Funciones matemáticas

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor menor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor del parámetro OUT no es válido si se cumple una de las siguientes condiciones:

- Los operandos indicados no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

Aun estando desactivada la verificación IEC, en los parámetros de entrada únicamente están permitidos los tipos de datos INT, DINT y REAL. Si utiliza los tipos de datos WORD o DWORD en los parámetros de entrada, la instrucción devolverá un resultado no definido.

Sintaxis

Para la instrucción "Determinar mínimo" se utiliza la siguiente sintaxis:

```
CALL MIN
<Tipo de datos>
IN1 := <Operando>
IN2 := <Operando>
IN3 := <Operando>
OUT := <Operando>
```

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL MIN                     // Se llama la instrucción "Determinar mínimo".
INT                          // Tipo de datos de la instrucción
IN1 := "TagIn_Value1"        // Primer valor de entrada que se compara.
IN2 := "TagIn_Value2"        // Segundo valor de entrada que se compara.
IN3 := "TagIn_Value3"        // Tercer valor de entrada que se compara.
OUT := "Tag_Minimum"         // Valor mínimo de la comparación
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Minimum	12222

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas IN1, IN2 e IN3 y escribe el valor mayor en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Los operandos indicados no son del mismo tipo de datos.
- Un operando tiene un valor no válido.

Nota

Aun estando desactivada la verificación IEC, en los parámetros de entrada únicamente están permitidos los tipos de datos INT, DINT y REAL. Si utiliza los tipos de datos WORD o DWORD en los parámetros de entrada, la instrucción devolverá un resultado no definido.

Sintaxis

Para la instrucción "Determinar máximo" se utiliza la siguiente sintaxis:

```
CALL MAX
<Tipo de datos>
IN1 := <Operando>
IN2 := <Operando>
IN3 := <Operando>
OUT := <Operando>
```

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL MAX                     // Se llama la instrucción "Determinar máximo".
INT                          // Tipo de datos de la instrucción
IN1 := "TagIn_Value1"        // Primer valor de entrada que se compara.
IN2 := "TagIn_Value2"        // Segundo valor de entrada que se compara.
IN3 := "TagIn_Value3"        // Tercer valor de entrada que se compara.
OUT := "Tag_Maximum"         // Valor máximo de la comparación
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Maximum	14444

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor de la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Para poder ejecutar la instrucción, todas las variables de las entradas deben ser del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no tienen el mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor del parámetro MN es mayor que el valor del parámetro MX.

Nota

Aun estando desactivada la verificación IEC, en los parámetros de entrada únicamente están permitidos los tipos de datos INT, DINT y REAL. Si utiliza los tipos de datos WORD o DWORD en los parámetros de entrada, la instrucción devolverá un resultado no definido.

Sintaxis

Para la instrucción "Ajustar valor límite" se utiliza la siguiente sintaxis:

```
CALL LIMIT
<Tipo de datos>
MN := <Operando>
IN := <Operando>
MX := <Operando>
OUT := <Operando>
```

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite inferior
IN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Valor de entrada
MX	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite superior
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL LIMIT                   // Se llama la instrucción "Ajustar valor límite".
INT                          // Tipo de datos de la instrucción
MN := "Tag_LowLimit"         // Límite inferior
IN := "Tag_InputValue"       // Valor de entrada
MX := "Tag_HighLimit"        // Límite superior
OUT := "Tag_Result"          // Resultado
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

En el operando "Tag_Result" se escribe el valor del operando "Tag_LowLimit", ya que el valor del operando "Tag_InputValue" se encuentra fuera del área límite definida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Transferencia

BLKMOV: Copiar área

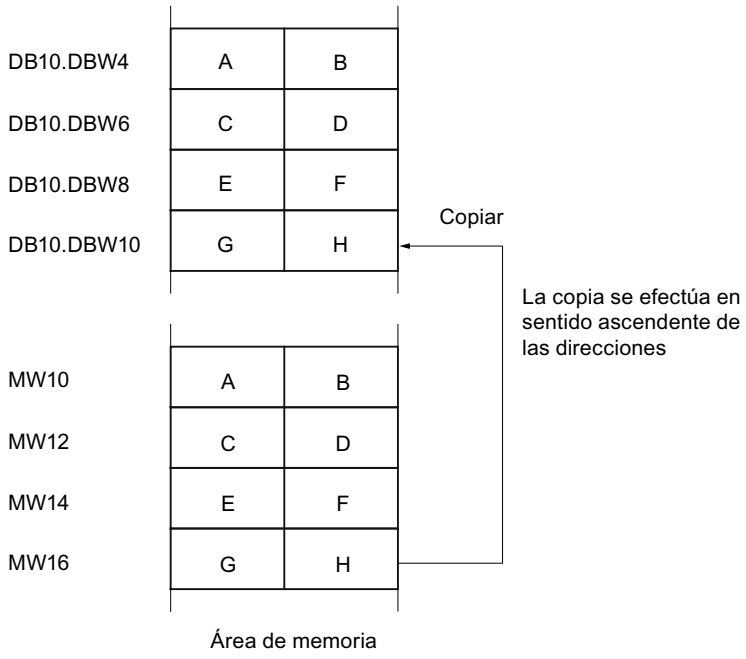
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

Mientras el área de origen no forme parte de un bloque de datos que solo figura en la memoria de carga, no hay limitaciones en cuanto a la profundidad de anidamiento.

Por el contrario, en caso de interrumpirse la ejecución de BLKMOV, en la que se copia desde un DB no relevante para la ejecución, no puede anidarse más la ejecución de la BLKMOV.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Reglas para copiar bloques de datos no relevantes para la ejecución

El área de origen también puede encontrarse en un bloque de datos no relevante para la ejecución en la memoria de carga. Los bloques de datos no relevantes para la ejecución se identifican con la palabra clave UNLINKED.

Si en la memoria de trabajo se copia un bloque de datos no relevante para la ejecución con la instrucción "Copiar área" y, al mismo tiempo, este se vuelve a cargar p. ej. desde la programadora, la ejecución de la instrucción puede retardarse varios milisegundos. Esto prolonga el ciclo del OB y puede activar la vigilancia del tiempo de ciclo.

Si un bloque de datos no relevante para la ejecución se copia con la instrucción "Copiar área" y la operación de copia se interrumpe, la instrucción no se podrá seguir ejecutando.

Si en la CPU existe la instrucción "Leer de un bloque de datos de la memoria de carga", debe utilizarse dicha instrucción para leer los bloques de datos no relevantes para la ejecución en la memoria de carga. Si se utiliza la instrucción "Copiar área" se emite el error W#16#8092.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

```
CALL BLKMOV
ANY
SRCBLK := <Operando>
RET_VAL := <Operando>
DSTBLK := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8092	El área de origen o destino está solo en la memoria de carga.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL BLKMOV	// Se llama la instrucción "Copiar área".
ANY	// Tipo de datos de la instrucción
SRCBLK := P#M100.0 BYTE 10	// Se copian 10 bytes a partir de MB100.
RET_VAL := "Tag_ErrorCode"	// Si ocurre un error durante la operación de copia, el respectivo código de error se deposita en la variable "Tag_ErrorCode".
DSTBLK := P#DB1.DBX0.0 BYTE 10	// Los bytes copiados se transmiten en DB1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK , no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 512 bytes como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de

caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

```
CALL UBLKMOV
SRCBLK := <Operando>
RET_VAL := <Operando>
DSTBLK := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL UBLKMOV	// Se llama la instrucción "Copiar área sin interrupciones".
SRCBLK := P#M100.0 BYTE 10	// Se copian 10 bytes a partir de MB100.
RET_VAL := "Tag_ErrorCode"	// Si ocurre un error durante la operación de copia, el respectivo código de error se deposita en la variable "Tag_ErrorCode".
DSTBLK := P#M200.0 BYTE 10	// Los bytes copiados se escriben en los 10 bytes a partir de MB200.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Cambiar formatos de visualización en el estado del programa (Página 3100)

FILL: Rellenar área

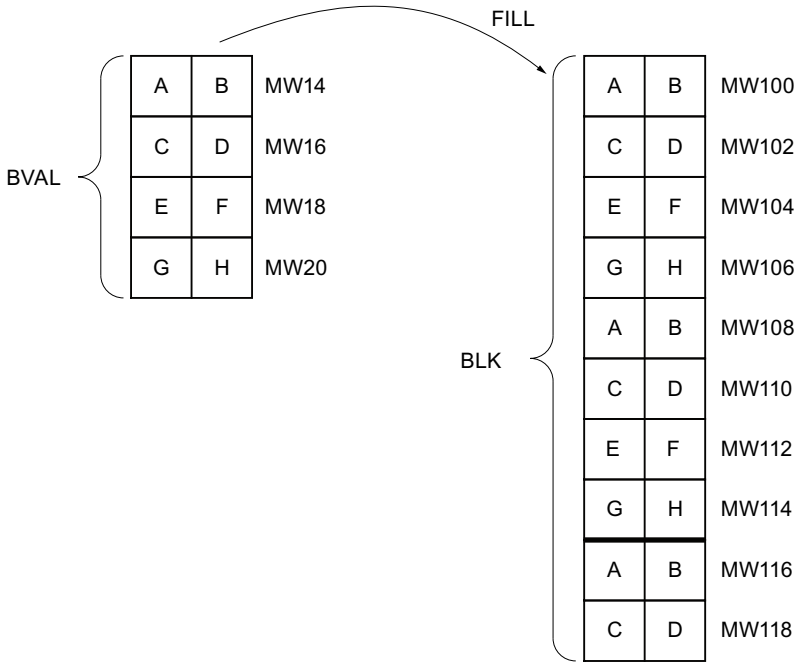
Descripción

La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: el contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que durante la ejecución de la instrucción "Rellenar área" no se modifican los datos de origen pues, de lo contrario, no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea un múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellena igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de destino o de origen disponible realmente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo BOOL, la longitud indicada debe ser divisible por 8. De lo contrario, la instrucción no se ejecutará.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro BVAL o BLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar estructuras

Al indicar una estructura a modo de parámetro de entrada, tenga en cuenta que la longitud de una estructura se adapta siempre a un número entero de bytes. Si se declara una estructura con un número impar de bytes, esta estructura necesita un byte adicional de espacio en memoria.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

```
CALL FILL
ANY
BVAL := <Operando>
RET_VAL := <Operando>
BLK := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BVAL	Input	ANY	I, Q, M, D, L	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
BLK	Output	ANY	I, Q, M, D, L	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL FILL                    // Se llama la instrucción "Rellenar área".
ANY                          // Tipo de datos de la instrucción
BVAL := P#M14.0 WORD 4       // Se copia el área de MW14 a MW20.
RET_VAL := "Tag_ErrorCode"   // Información de error
BLK := P#M100.0 WORD 10      // El área de memoria de MW100 a MW118 se rellena con
                             el contenido de las 4 palabras contenidas en el área
                             de memoria del parámetro BVAL.
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Conversión

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

```
CALL SCALE
IN := <Operando>
HI_LIM := <Operando>
LO_LIM := <Operando>
BIPOLAR := <Operando>
```

```
RET_VAL := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1"
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                         Explicación
CALL SCALE                  // Se llama la instrucción "Escalar".
```

AWL	Explicación
IN := "Tag_InputValue"	// Indica el valor que se convierte y escala.
HI_LIM := "Tag_HighLimit"	// Valor límite superior
LO_LIM := "Tag_LowLimit"	// Valor límite inferior
BIPOLAR := "Tag_Bipolar"	// Indica si el valor del parámetro IN se interpreta como bipolar o unipolar.
RET_VAL := "Tag_ErrorCode"	// Información de error
OUT := "Tag_OutputValue"	// Resultado de la instrucción

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" desescala el número en coma flotante del parámetro IN en unidades físicas entre un valor límite inferior y un valor límite superior y lo convierte en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Sintaxis

Para la instrucción "Desescalar" se utiliza la siguiente sintaxis:

```
CALL UNSCALE
IN := <Operando>
HI_LIM := <Operando>
LO_LIM := <Operando>
BIPOLAR := <Operando>
RET_VAL := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL UNSCALE                 // Se llama la instrucción "Desescalar".
IN := "Tag_InputValue"       // Indica el valor que se desescala.
HI_LIM := "Tag_HighLimit"    // Valor límite superior
LO_LIM := "Tag_LowLimit"     // Valor límite inferior
```

AWL	Explicación
BIPOLAR := "Tag_Bipolar"	// Indica si el valor del parámetro IN se interpreta como bipolar o unipolar.
RET_VAL := "Tag_ErrorCode"	// Información de error
OUT := "Tag_OutputValue"	// Resultado de la instrucción

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Control del programa

Control del tiempo de ejecución

COMPRESS: Comprimir la memoria de la CPU

Descripción

La instrucción "Comprimir la memoria de la CPU" comprime la memoria de trabajo de la CPU o la memoria de carga RAM. Los huecos creados por haber borrado o cargado bloques, se rellenan juntando los bloques existentes.

Con la llamada de la instrucción "Comprimir la memoria de la CPU" se lanza el proceso de compresión. El proceso de compresión se reparte entre varios ciclos del programa. La vigilancia del proceso de compresión se realiza mediante los valores de los distintos parámetros. Si el parámetro BUSY de la instrucción tiene el valor "1", significa que el proceso de compresión se está ejecutando todavía. El valor "1" del parámetro DONE indica que ha finalizado el proceso de compresión. La información de error puede consultarse en el parámetro RET_VAL.

Si se está ejecutando un proceso de compresión lanzado externamente, la llamada de la instrucción "Comprimir la memoria de la CPU" origina un error.

Control del proceso de compresión

Con la llamada única de la instrucción "Comprimir la memoria de la CPU" se lanza el proceso de compresión. No obstante, no puede controlar si la compresión se ha realizado satisfactoriamente.

Si desea realizar esta comprobación debe proceder de la siguiente manera:

La instrucción "Comprimir la memoria de la CPU" se debe llamar en el ciclo. Una vez realizada la llamada se debe evaluar a continuación el parámetro RET_VAL. Si este es igual a "0" se deben evaluar los parámetros BUSY y DONE. Si BUSY = "1" y DONE = "0", ello indica que el proceso de compresión aún permanece activo. En cuanto BUSY adopta valor "0" y DONE el valor "1", ello será un indicador de que el proceso de compresión ha concluido satisfactoriamente. Si a continuación se vuelve a llamar la instrucción "Comprimir la memoria de la CPU", se iniciará un nuevo proceso de compresión.

Sintaxis

Para la instrucción "Comprimir la memoria de la CPU" se utiliza la siguiente sintaxis:

```
CALL COMPRESS
RET_VAL := <Operando>
BUSY := <Operando>
DONE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comprimir la memoria de la CPU":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
BUSY	Output	BOOL	I, Q, M, D, L	Información de si está activo el proceso de compresión
DONE	Output	BOOL	I, Q, M, D, L	Información de si ha finalizado el proceso de compresión

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error El estado del proceso de compresión puede consultarse en los parámetros BUSY y DONE.
8091	Ya se está ejecutando un proceso de compresión lanzado externamente.
8092	La instrucción no se puede ejecutar porque se cumple una de las condiciones siguientes: • La función "Borrar bloque" está activa. • Una función de PG está accediendo a uno de los bloques que se deben desplazar. • Ya se está ejecutando un proceso de compresión lanzado externamente. • La CPU H se encuentra en un proceso de acoplamiento o actualización.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU. Reiniciando el tiempo de vigilancia del ciclo es posible impedir que se produzcan errores o que la CPU pase a STOP.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" puede utilizarse en bloques de la clase de prioridad 1 (los OB de ciclo), así como en los bloques que se llaman desde ellos.

Si se llama la instrucción en un bloque de prioridad superior, p. ej. en una alarma de proceso o de diagnóstico o en una alarma cíclica, la instrucción no se ejecuta.

Nota

En caso de utilizar la instrucción "Reiniciar tiempo de vigilancia del ciclo", asegúrese de que no se produce un bucle sin fin en la ejecución cíclica del programa, es decir, en el OB1. De lo contrario, la CPU ya no alcanza el punto de control del ciclo. Como consecuencia no pueden ejecutarse ciertas funciones de la CPU (p. ej., la actualización de la memoria imagen de proceso).

Sintaxis

Para la instrucción "Reiniciar tiempo de vigilancia del ciclo" se utiliza la siguiente sintaxis:

```
CALL RE_TRIGR
```

Parámetros

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Sintaxis

Para la instrucción "Finalizar programa" se utiliza la siguiente sintaxis:

```
CALL STP
```

Parámetros

La instrucción "Finalizar programa" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" detiene la ejecución del programa durante un periodo parametrizado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible parametrizar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve la información de error.

Sintaxis

Para la instrucción "Programar retardo" se utiliza la siguiente sintaxis:

```
CALL WAIT
```

```
WT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

PROTECT: Cambiar nivel de protección

Descripción

La instrucción "Cambiar nivel de protección" permite modificar el nivel de protección "1" configurado para la CPU.

El nivel de protección "1" parametrizado con la opción activada "Eliminable por contraseña" hace que sea necesario parametrizar una contraseña. Esta contraseña permite durante el funcionamiento acceder en escritura a la CPU protegida contra escritura mediante la instrucción "Cambiar nivel de protección" (p. ej. cargar).

El nivel de protección se especifica en el parámetro MODE de la instrucción. El nivel de protección actual de la CPU se visualiza en la ventana de inspección en "Propiedades > Protección".

La tabla siguiente muestra una explicación de los niveles de protección:

Nivel de protección	Significado
1	Se permiten todas las funciones de PG. En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
2	• El programa y la configuración de la CPU no se pueden modificar. El programa contenido en la CPU se puede leer. • Se permiten las funciones de control, observación y comunicación del proceso. • Se permiten todas las funciones de información. • Si conoce la contraseña válida, podrá desactivar la protección contra escritura ajustada. • En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
3	• Aun conociendo la contraseña válida, no podrá desactivar la protección contra lectura y escritura ajustada. • Si al llamar la instrucción "Cambiar nivel de protección" existe una conexión legitimada, la llamada de la instrucción no tendrá ningún efecto en esta conexión.

Si se ha configurado el nivel de protección "2" o "3" para la CPU, la llamada de la instrucción "Cambiar nivel de protección" no tiene efecto alguno.

La efectividad del nivel de protección "2" o "3" una vez ejecutada la instrucción "Cambiar nivel de protección" puede consultarse en la lista parcial SZL con la ID de SZL W#16#0232 y el índice W#16#0004.

Nota

La instrucción "Cambiar nivel de protección" permite ajustar un nivel de protección más bajo que aquel ajustado con "Configurar hardware" de STEP 7. Supongamos que se ha configurado el nivel de protección 3 y a continuación se llama la instrucción "Cambiar nivel de protección" con el parámetro MODE = 12. Esto bloqueará la legitimación de la contraseña. Si la instrucción se llama nuevamente con el parámetro MODE = 0 o 1, el nivel de protección 3 se mantiene y será posible desactivar la protección de lectura y escritura introduciendo la contraseña válida.

Efectos de las transiciones de estado operativo en el nivel de protección

La tabla siguiente muestra qué efectos tienen las transiciones de estado operativo en el nivel de protección ajustado con la instrucción "Cambiar nivel de protección".

Acción	Efectos en el nivel de protección
• Selector de modo en STOP • Conexión (POWER ON) sin respaldo • Borrado total mediante el selector de modo • Cambios de configuración y/o del programa en el estado operativo STOP de la CPU S7-400	El nivel de protección se pone a "1". Se permiten todas las funciones de PG.
• Arranque en frío • Rearranque (en caliente)	El sistema operativo pone el nivel de protección a "1". Se permiten todas las funciones de PG. En caso necesario, llamando la instrucción "Cambiar nivel de protección" en el programa es posible ajustar el nivel de protección en "2" o "3".
Conexión (POWER ON) respaldada	No cambia
Transición de estado operativo RUN/ARRANQUE/PARADA -> STOP (debida a la llamada de la instrucción "Finalizar programa" (STP), a errores del programa sin el correspondiente OB de error o a una acción del usuario).	No cambia
S7-400: Rearranque por acción del usuario o automáticamente tras conexión (POWER ON)	No cambia

Sintaxis

Para la instrucción "Cambiar nivel de protección" se utiliza la siguiente sintaxis:

```
CALL PROTECT
MODE := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cambiar nivel de protección":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	WORD	I, Q, M, D, L o constante	Nivel de protección Valores posibles: • W#16#0000: Ajuste del nivel de protección "1" • W#16#0001: Ajuste del nivel de protección "2" con legitimación de la contraseña • W#16#000C: Ajuste del nivel de protección "3" sin legitimación de la contraseña En el parámetro MODE también se pueden indicar constantes.
RET_VAL	Output	INT	I, Q, M, D, L	Si ocurre un error durante la ejecución, se emite una información de error.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
8090	Valor no admisible en el parámetro MODE.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

Por principio, la CPU debe protegerse contra accesos en el estado operativo RUN. La protección se debe suprimir únicamente si se conmuta el selector de modo a STOP. Esto se logra llamando la instrucción "Cambiar nivel de protección" en un OB de arranque y ejecutándola con el valor W#16#0001 (nivel de protección "2") en el parámetro MODE. De esta manera, el nivel de protección "2" permanece ajustado aún tras un rearranque completo (en caliente).

```
AWL                          Explicación
CALL PROTECT                 // Se llama la instrucción "Cambiar nivel de
                             protección".
MODE := "Tag_ProtectionLevel" // Ajustar el nivel de protección
RET_VAL := "Tag_ErrorCode"    // Información de error
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Cambiar formatos de visualización en el estado del programa (Página 3100)

CiR: Controlar proceso CiR

Descripción

La instrucción "Controlar proceso CiR" influye en el proceso CiR del programa de usuario como se indica a continuación:

- Bloquear el proceso CiR. En este caso, se rechazará siempre la carga de una configuración modificada desde la programadora en la CPU. El bloqueo se aplica hasta que se elimina con la instrucción "Controlar proceso CiR".
- El proceso CiR se puede bloquear de forma condicionada, indicando un límite superior para el tiempo de sincronización CiR en el parámetro FRZ_TIME. En este caso, solo se permitirá cargar una configuración modificada desde la programadora en la CPU si el tiempo de sincronización CiR calculado en la CPU no excede el límite superior definido.
- Habilitar el proceso CiR. El límite superior del tiempo de sincronización CiR se establece al valor predeterminado de 1000 ms.
- Determina si el proceso CiR está habilitado o no. Si el proceso CiR está habilitado o lo está condicionadamente, el límite superior actual del tiempo de sincronización CiR se indica en el parámetro A_FT de la instrucción.

Nota

Durante el tiempo de sincronización CiR, las salidas están congeladas y las entradas no se evalúan.

Sintaxis

Para la instrucción "Controlar proceso CiR" se utiliza la siguiente sintaxis:

```
CALL CIR
MODE := <Operando>
FRZ_TIME := <Operando>
RET_VAL := <Operando>
A_FT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Controlar proceso CiR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición Los valores admisibles del parámetro MODE son: • 0: Función de información • 1: Habilitar proceso CiR • 2: Bloquear proceso CiR • 3: Bloquear proceso CiR de forma condicionada. El límite superior del tiempo de sincronización CiR se define en FRZ_TIME.
FRZ_TIME	Input	TIME	I, Q, M, D, L o constante	Límite superior del tiempo de sincronización CiR en milisegundos. Rango de valores admisible: 60 ... 2500 ms (ajuste predeterminado: 1000 ms) El parámetro FRZ_TIME solo es relevante si el parámetro MODE está puesto al valor "3".
RET_VAL	Output	INT	I, Q, M, D, L	Información de error Si el parámetro MODE está a "0", el parámetro RET_VAL indica si el proceso CiR está habilitado o no.
A_FT	Output	TIME	I, Q, M, D, L	Límite superior válido actualmente del tiempo de sincronización CiR

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor 1, 2 ó 3.
0001	El proceso CiR está habilitado. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
0002	El proceso CiR está bloqueado. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
0003	El proceso CiR está bloqueado de forma condicionada. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".

Código de error* (W#16#...)	Explicación
8001	La CPU no está lista para un proceso CiR. Se está utilizando una CPU H en el sistema H (modo "Individual") o una CPU estándar en modo multiprocesador.
8002	El valor del parámetro MODE no es admisible.
8003	El valor del parámetro FRZ_TIME no es admisible.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

La instrucción "Controlar proceso CiR" permite programar p. ej. que no se lance ningún proceso CiR en periodos en los que se requiera la capacidad máxima de la CPU para el procesamiento.

En este caso, la instrucción "Controlar proceso CiR" debe llamarse en el programa antes de comenzar el periodo de actividad aumentada del proceso y ejecutarse con el valor "2" (bloquear proceso CiR) en el parámetro MODE. Con ello se bloquea el proceso CiR.

Para volver a habilitar el proceso CiR tras finalizar el periodo de actividad aumentada del proceso, la instrucción "Controlar proceso CiR" debe llamarse de nuevo y ejecutarse con el valor "1" en el parámetro MODE. Como alternativa, el parámetro MODE se puede poner al valor "3", con lo que el proceso CiR se bloquea de forma condicionada.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Operaciones lógicas con palabras

DECO: Descodificar

Descripción

La instrucción "Descodificar" activa un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor del parámetro IN y activa el bit del parámetro OUT cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor del parámetro IN es mayor que 31, se ejecuta una operación modulo 32.

Sintaxis

Para la instrucción "Descodificar" se utiliza la siguiente sintaxis:

```
CALL DECO
<Tipo de datos> DWORD
IN := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	WORD, INT	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida
OUT	Output	DWORD	I, Q, M, D, L, P	Valor de salida

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

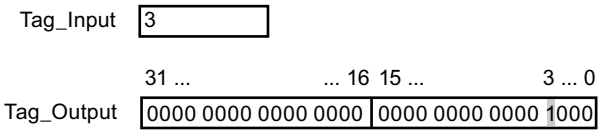
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL DECO	// Se llama la instrucción "Descodificar".
WORD DWORD	// Tipos de datos de la instrucción
IN := "Tag_Input"	// Valor de entrada que predetermina el bit que debe activarse en el valor de salida.
OUT := "Tag_Output"	// Resultado

La figura siguiente muestra el funcionamiento de la instrucción con valores concretos:



La instrucción lee el número de bit "3" del valor del operando "Tag_Input" de la entrada y activa el tercer bit del valor del operando "Tag_Output" de la salida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

ENCO: Codificar

Descripción

La instrucción "Codificar" lee el número del bit menos significativo que está activado en el valor de entrada y devolverlo en el parámetro OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor del parámetro IN y escribe su número de bit en el operando del parámetro OUT. Si el parámetro IN devuelve el valor DW#16#00000001 o DW#16#00000000, en el parámetro OUT se deposita el valor "0".

Sintaxis

Para la instrucción "Codificar" se utiliza la siguiente sintaxis:

```
CALL ENCO
DWORD
IN := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL ENCO	// Se llama la instrucción "Codificar".
DWORD	// Tipo de datos de la instrucción
IN := "Tag_Input"	// Valor de entrada cuyo bit activado menos significativo se lee.
OUT := "Tag_Output"	// Número del bit menos significativo que está activado en el valor de entrada

La figura siguiente muestra el funcionamiento de la instrucción con valores concretos:

	31 16	15 ... 3 ... 0
“Tag_Input“	0000 1111 0000 0101	0000 1001 0000 1000

“Tag_Output“	3

La instrucción selecciona el bit menos significativo que está activado en la variable "Tag_Input" y escribe su bit "3" en la variable "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite elegir, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copiar su contenido en la salida OUT. Si la entrada G tiene el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G devuelve el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

Para poder procesar la instrucción, las variables de todos los parámetros deben ser del mismo tipo de datos.

Sintaxis

Para la instrucción "Seleccionar" se utiliza la siguiente sintaxis:

```
CALL SEL
<Tipo de datos>
G := <Operando>
IN0 := <Operando>
IN1 := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
G	Input	BOOL	I, Q, M, D, L	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE, DT	I, Q, M, D, L, P	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE, DT	I, Q, M, D, L, P	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE, DT	I, Q, M, D, L, P	Resultado

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL SEL                     // Se llama la instrucción "Seleccionar".
INT                          // Tipo de datos de la instrucción
G := "Tag_Input_G"           // Estado lógico del interruptor
IN0 := "Tag_Input0"          // Primer valor de entrada
IN1 := "Tag_Input1"          // Segundo valor de entrada
OUT := "Tag_Output"          // Valor de la entrada seleccionada
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor
G	Tag_Input_G	1
IN0	Tag_Input0	W#16#0000
IN1	Tag_Input1	W#16#FFFF
OUT	Tag_Output	W#16#FFFF

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Otras instrucciones

SET: Activar área por bits

Descripción

La instrucción "Activar área por bits" establece el estado lógico de los bits en "1" en un área indicada. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben activarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta solo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

Sintaxis

Para la instrucción "Activar área por bits" se utiliza la siguiente sintaxis:

```
CALL SET
S_BIT := <Operando>
N := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del área cuyos bits se activan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben activarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL SET	// Se llama la instrucción "Activar área por bits".
S_BIT := P#M0.0	// Puntero hacia el primer bit que debe activarse (M0.0)
N := 10	// Número de bits que deben activarse

En el ejemplo, la instrucción activa 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado (Página 2547)

SETP: Activar mapa de bits en el área de periferia

Descripción

La instrucción "Activar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "1". Paralelamente a los bits de periferia, también se activan los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe activarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no se ha conectado periferia para una parte del mapa de bits seleccionado, la instrucción intentará activar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se activa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas.

ATENCIÓN
El ajuste de bits El ajuste de los bits al valor de los bits correspondientes en la memoria imagen de proceso de las salidas puede ocasionar, p. ej., el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "1".

La instrucción "Activar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Activar mapa de bits en el área de periferia" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El parámetro N devuelve el valor "0".
- El bit MCR tiene el estado lógico "0".

Sintaxis

Para la instrucción "Activar mapa de bits en el área de periferia" se utiliza la siguiente sintaxis:

```
CALL SETP
N := <Operando>
RET_VAL := <Operando>
SA := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben activarse
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero hacia el primer bit que debe activarse

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL SETP	// Se llama la instrucción "Activar mapa de bits en el área de periferia".
N := 8	// Número de bits que deben activarse
RET_VAL := "Tag_ErrorCode"	// Información de error
SA := P#10.0	// Puntero hacia el primer bit que debe activarse (Q10.0)

En el ejemplo, la instrucción activa 8 bits en el área de periferia conforme a las salidas Q10.0 a Q10.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SETI: Activar área por bytes

Descripción

La instrucción "Activar área por bytes" pone a "1" el estado lógico de los bits en un área definida de bytes. El punto inicial del área se define en el parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben activarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte; por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas (Q) se pone a "0".

La llamada de la instrucción "Activar área por bytes" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa (P).
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

Sintaxis

Para la instrucción "Activar área por bytes" se utiliza la siguiente sintaxis:

```
CALL SETI
S_BYTE := <Operando>
N := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se activan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área que debe activarse El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL SETI                    // Se llama la instrucción "Activar área por bytes".
S_BYTE := P#2.0              // Puntero hacia el primer byte del área (P2.0)
N := 16                      // Número de bits que deben activarse
```

En el ejemplo, la instrucción activa 2 bytes (16 bits) en el área de P2.0 a P3.7 (salidas periféricas).

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado (Página 2547)

RESET: Desactivar área por bits

Descripción

La instrucción "Desactivar área por bits" pone a "0" el estado lógico de los bits en un área definida. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben desactivarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta solo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

Sintaxis

Para la instrucción "Desactivar área por bits" se utiliza la siguiente sintaxis:

```
CALL RESET
S_BIT := <Operando>
N := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BIT	Input	POINTER	I, Q, M, D	Señala al primer bit del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben desactivarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL RESET	// Se llama la instrucción "Desactivar área por bits".
S_BIT := P#M0.0	// Puntero al primer bit (M0.0) que debe desactivarse.
N := 10	// Número de bits que deben desactivarse

En el ejemplo, la instrucción desactiva 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado (Página 2547)

RESETP: Desactivar mapa de bits en el área de periferia

Descripción

La instrucción "Desactivar mapa de bits en el área de periferia" pone los bits de un área de periferia al estado lógico "0". Simultáneamente con los bits de periferia se desactivan también los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe desactivarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no hay periferia enchufada para una parte del mapa de bits seleccionado, la instrucción intentará desactivar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se desactiva por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas.

ATENCIÓN
El ajuste de bits El ajuste de los bits al valor de los bits correspondientes en la memoria imagen de proceso de las salidas puede ocasionar, p. ej., el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "0".

La instrucción "Desactivar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Desactivar mapa de bits en el área de periferia" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El parámetro N devuelve el valor "0".
- El bit MCR tiene el estado lógico "0".

Sintaxis

Para la instrucción "Desactivar mapa de bits en el área de periferia" se utiliza la siguiente sintaxis:

```
CALL RESETP
N := <Operando>
RET_VAL := <Operando>
SA := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben desactivarse.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero al primer bit que debe desactivarse.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL RESETP	// Se llama la instrucción "Desactivar mapa de bits en el área de periferia".
N := 8	// Número de bits que deben desactivarse.
RET_VAL := "Tag_ErrorCode"	// Información de error
SA := P#10.0	// Puntero hacia el primer bit (Q10.0) que debe desactivarse.

En el ejemplo, la instrucción desactiva 8 bits en el área de periferia correspondientes a las salidas Q10.0 a Q10.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RESETI: Desactivar área por bytes

Descripción

La instrucción "Desactivar área por bytes" pone a "0" el estado lógico de los bits en un área definida de bytes. El punto inicial del área se define mediante el puntero del parámetro

S_BYTE. El tamaño del área se define indicando el número de bits que deben desactivarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte; por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas (Q) se pone a "0".

La llamada de la instrucción "Desactivar área por bytes" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa (P).
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

Sintaxis

Para la instrucción "Desactivar área por bytes" se utiliza la siguiente sintaxis:

```
CALL RESETI
S_BYTE := <Operando>
N := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área cuyos bits se desactivan. El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL RESETI	// Se llama la instrucción "Desactivar área por bytes".
S_BYTE := P#2.0	// Puntero hacia el primer byte del área (P2.0)
N := 16	// Número de bits que deben desactivarse.

En el ejemplo, la instrucción desactiva 2 bytes (16 bits) en el área de P2.0 a P3.7 (salidas periféricas).

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Activar el bit de estado (Página 2547)

REPL_VAL: Introducir valor sustitutivo

Descripción

La instrucción "Introducir valor sustitutivo" introduce un valor sustitutivo en el acumulador 1 (ACU 1) desde un bloque de organización de error síncrono.

Nota

La instrucción "Introducir valor sustitutivo" solo se puede llamar desde bloques de organización de error síncrono (OB 121, OB 122).

La instrucción "Introducir valor sustitutivo" se utiliza si ya no es posible leer ningún valor de un módulo. Tras cada acceso a dicho módulo se inicia el OB 122. Llamando la instrucción "Introducir valor sustitutivo" es posible cargar un valor sustitutivo en el acumulador 1. La ejecución del programa se prosigue entonces con este valor sustitutivo. La información necesaria para seleccionar el valor sustitutivo puede consultarse en las variables locales del OB 122.

Sintaxis

Para la instrucción "Introducir valor sustitutivo" se utiliza la siguiente sintaxis:

```
CALL REPL_VAL
VAL := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Introducir valor sustitutivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
VAL	Input	DWORD	I, Q, M, D, L o constante	Valor sustitutivo
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#....)	Descripción
0000	Ningún error
8080	La instrucción no se ha llamado desde un OB de error síncrono.

Código de error* (W#16#....)	Descripción
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" ocupa los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de este es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

```
CALL DRUM, "DB instancia"
RESET := <Operando>
JOG := <Operando>
DRUM_EN := <Operando>
LST_STEP := <Operando>
EVENT1 - 16 := <Operando>
OUT1 - 16 := <Operando>
Q := <Operando>
OUT_WORD := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" indica una condición de desactivación
JOG	Input	BOOL	I, Q, M, D, L	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DRUM_EN	Input	BOOL	I, Q, M, D, L	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado.
EVENT(i), 1<= i <=16	Input	BOOL	I, Q, M, D, L	Bit de evento (i); El estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	I, Q, M, D, L, P	Primer paso predeterminado del circuito secuencial sucesivo (1 a 16)
DSC	Static	BYTE	I, Q, M, D, L, P	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P	Tiempo de ejecución restante para el paso actual

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DTBP	Static	WORD	I, Q, M, D, L, P	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	hora de sistema de la llamada anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente, la instrucción pasa del paso 1 al paso 2. Los bits de salida (OUT1 a OUT16) y la palabra de salida (OUT_WORD) se activan según la máscara configurada para el paso 2 y los valores del parámetro OUT_VAL.

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
CALL DRUM, "DRUM_DB"	// Se llama la instrucción "Materializar circuito secuencial sucesivo" y se crea el bloque de datos instancia "DRUM_DB".
RESET := "Tag_Reset"	// Entrada de inicialización
JOG := "Tag_Input_Jog"	// Si se produce un flanco ascendente, la instrucción pasa al paso siguiente.
DRUM_EN := "Tag_Input_DrumEN"	// El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP := "Tag_Number_LastStep"	// Número del último paso programado
EVENT1 := "MyTag_Event_1"	// Bit de evento 1
EVENT2 := "MyTag_Event_2"	// Bit de evento 2
EVENT3 := "MyTag_Event_3"	// Bit de evento 3
EVENT4 := "MyTag_Event_4"	// Bit de evento 4
EVENT5 := "MyTag_Event_5"	// Bit de evento 5
EVENT6 := "MyTag_Event_6"	// Bit de evento 6
EVENT7 := "MyTag_Event_7"	// Bit de evento 7
EVENT8 := "MyTag_Event_8"	// Bit de evento 8
EVENT9 := "MyTag_Event_9"	// Bit de evento 9
EVENT10 := "MyTag_Event_10"	// Bit de evento 10
EVENT11 := "MyTag_Event_11"	// Bit de evento 11
EVENT12 := "MyTag_Event_12"	// Bit de evento 12
EVENT13 := "MyTag_Event_13"	// Bit de evento 13
EVENT14 := "MyTag_Event_14"	// Bit de evento 14
EVENT15 := "MyTag_Event_15"	// Bit de evento 15
EVENT16 := "MyTag_Event_16"	// Bit de evento 16
OUT1 := "MyTag_Output_1"	// Bit de salida 1
OUT2 := "MyTag_Output_2"	// Bit de salida 2
OUT3 := "MyTag_Output_3"	// Bit de salida 3
OUT4 := "MyTag_Output_4"	// Bit de salida 4
OUT5 := "MyTag_Output_5"	// Bit de salida 5
OUT6 := "MyTag_Output_6"	// Bit de salida 6
OUT7 := "MyTag_Output_7"	// Bit de salida 7
OUT8 := "MyTag_Output_8"	// Bit de salida 8
OUT9 := "MyTag_Output_9"	// Bit de salida 9
OUT10 := "MyTag_Output_10"	// Bit de salida 10
OUT11 := "MyTag_Output_11"	// Bit de salida 11
OUT12 := "MyTag_Output_12"	// Bit de salida 12
OUT13 := "MyTag_Output_13"	// Bit de salida 13
OUT14 := "MyTag_Output_14"	// Bit de salida 14
OUT15 := "MyTag_Output_15"	// Bit de salida 15
OUT16 := "MyTag_Output_16"	// Bit de salida 16
Q := "Tag_Output_Q"	// El estado lógico "1" indica que ha transcurrido el tiempo del último paso.

AWL	Explicación
OUT_WORD := "Tag_OutputWord"	// Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida
ERR_CODE := "Tag_ErrorCode"	// Información de error

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para asignar valores a los parámetros de entrada:

Parámetro	Operando	Dirección	Valor
RESET	Tag_Reset	M0.0	FALSE
JOG	Tag_Input_JOG	M0.1	FALSE
DRUM_EN	Tag_Input_DrumEN	M0.2	TRUE
LST_STEP	Tag_Number_LastStep	MB1	B#16#08
EVENT2	MyTag_Event_2	M20.0	FALSE
EVENT4	MyTag_Event_4	M20.1	FALSE
EVENT6	MyTag_Event_6	M20.2	FALSE
EVENT8	MyTag_Event_8	M20.3	FALSE
EVENT10	MyTag_Event_10	M20.4	FALSE
EVENT12	MyTag_Event_12	M20.5	FALSE
EVENT14	MyTag_Event_14	M20.6	FALSE
EVENT16	MyTag_Event_16	M20.7	FALSE

Los siguientes valores están almacenados en el bloque de datos instancia "DRUM_DB" de la instrucción:

Parámetro	Dirección	Valor
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSP	DBB13	W#16#0001
DSC	DBB14	W#16#0001
DCC	DBD16	DW#16#0000000A
DTBP	DBW20	W#16#0001
S_PRESET[1]	DBW26	W#16#0064
S_PRESET[2]	DBW28	W#16#00C8
OUT_VAL[1,0]	DBX58.0	TRUE
OUT_VAL[1,1]	DBX58.1	TRUE
OUT_VAL[1,2]	DBX58.2	TRUE
OUT_VAL[1,3]	DBX58.3	TRUE
OUT_VAL[1,4]	DBX58.4	TRUE
OUT_VAL[1,5]	DBX58.5	TRUE
OUT_VAL[1,6]	DBX58.6	TRUE

Parámetro	Dirección	Valor
OUT_VAL[1,7]	DBX58.7	TRUE
OUT_VAL[1,8]	DBX59.0	TRUE
OUT_VAL[1,9]	DBX59.1	TRUE
OUT_VAL[1,10]	DBX59.2	TRUE
OUT_VAL[1,11]	DBX59.3	TRUE
OUT_VAL[1,12]	DBX59.4	TRUE
OUT_VAL[1,13]	DBX59.5	TRUE
OUT_VAL[1,14]	DBX59.6	TRUE
OUT_VAL[1,15]	DBX59.7	TRUE
OUT_VAL[2,0]	DBX60.0	FALSE
OUT_VAL[2,1]	DBX60.1	FALSE
OUT_VAL[2,2]	DBX60.2	FALSE
OUT_VAL[2,3]	DBX60.3	FALSE
OUT_VAL[2,4]	DBX60.4	FALSE
OUT_VAL[2,5]	DBX60.5	FALSE
OUT_VAL[2,6]	DBX60.6	FALSE
OUT_VAL[2,7]	DBX60.7	FALSE
OUT_VAL[2,8]	DBX61.0	FALSE
OUT_VAL[2,9]	DBX61.1	FALSE
OUT_VAL[2,10]	DBX61.2	FALSE
OUT_VAL[2,11]	DBX61.3	FALSE
OUT_VAL[2,12]	DBX61.4	FALSE
OUT_VAL[2,13]	DBX61.5	FALSE
OUT_VAL[2,14]	DBX61.6	FALSE
OUT_VAL[2,15]	DBX61.7	FALSE
S_MASK[2,0]	DBX92.0	FALSE
S_MASK[2,1]	DBX92.1	TRUE
S_MASK[2,2]	DBX92.2	TRUE
S_MASK[2,3]	DBX92.3	TRUE
S_MASK[2,4]	DBX92.4	TRUE
S_MASK[2,5]	DBX92.5	FALSE
S_MASK[2,6]	DBX92.6	TRUE
S_MASK[2,7]	DBX92.7	TRUE
S_MASK[2,8]	DBX93.0	FALSE
S_MASK[2,9]	DBX93.1	FALSE
S_MASK[2,10]	DBX93.2	TRUE
S_MASK[2,11]	DBX93.3	TRUE
S_MASK[2,12]	DBX93.4	TRUE
S_MASK[2,13]	DBX93.5	TRUE
S_MASK[2,14]	DBX93.6	FALSE
S_MASK[2,15]	DBX93.7	TRUE

Los parámetros de salida tienen los valores siguientes antes de ejecutarse la instrucción:

Parámetro	Operando	Dirección	Valor
Q	Tag_Output_Q	M6.0	FALSE
OUTWORD	Tag_OutputWord	MW8	W#16#FFFF
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	TRUE
OUT3	MyTag_Output_3	M4.2	TRUE
OUT4	MyTag_Output_4	M4.3	TRUE
OUT5	MyTag_Output_5	M4.4	TRUE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	TRUE
OUT8	MyTag_Output_8	M4.7	TRUE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	TRUE
OUT12	MyTag_Output_12	M5.3	TRUE
OUT13	MyTag_Output_13	M5.4	TRUE
OUT14	MyTag_Output_14	M5.5	TRUE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	TRUE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Dirección	Valor
OUT1	MyTag_Output_1	M4.0	TRUE
OUT2	MyTag_Output_2	M4.1	FALSE
OUT3	MyTag_Output_3	M4.2	FALSE
OUT4	MyTag_Output_4	M4.3	FALSE
OUT5	MyTag_Output_5	M4.4	FALSE
OUT6	MyTag_Output_6	M4.5	TRUE
OUT7	MyTag_Output_7	M4.6	FALSE
OUT8	MyTag_Output_8	M4.7	FALSE
OUT9	MyTag_Output_9	M5.0	TRUE
OUT10	MyTag_Output_10	M5.1	TRUE
OUT11	MyTag_Output_11	M5.2	FALSE
OUT12	MyTag_Output_12	M5.3	FALSE
OUT13	MyTag_Output_13	M5.4	FALSE
OUT14	MyTag_Output_14	M5.5	FALSE
OUT15	MyTag_Output_15	M5.6	TRUE
OUT16	MyTag_Output_16	M5.7	FALSE
Q	Tag_Output_Q	M6.0	FALSE

Parámetro	Operando	Dirección	Valor
OUTWORD	Tag_OutputWord	MW8	W#16#4321
ERR_CODE	Tag_ErrorCode	MW10	W#16#0000

Los siguientes valores se modifican en el bloque de datos instancia DRUM_DB de la instrucción después de su ejecución:

Parámetro	Dirección	Valor
JOG_HIS	DBX12.0	FALSE
EOD	DBX12.1	FALSE
DSC	DBB14	W#16#0002
DCC	DBD16	DW#16#000000C8

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM_X: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite materializar un circuito secuencial sucesivo con 16 pasos como máximo. El número del primer paso se define con el parámetro DSP. El número del último paso se predetermina con el parámetro LST_STEP.

En cada paso se escriben todos los 16 bits de salida OUT0 a OUT15 y el parámetro de salida OUT_WORD (que contiene los bits de salida). A un bit de salida se asigna bien sea el bit correspondiente del campo predefinido OUT_VAL, o bien el bit de salida del paso anterior. El valor que se asigne dependerá de la ocupación de los bits de máscara en el parámetro S_MASK.

Nota

El ajuste predeterminado de los bits de máscara es 0. Para modificar el ajuste actual de uno o varios bits de máscara, realice el cambio en el bloque de datos instancia.

La instrucción "Materializar circuito secuencial sucesivo" pasa al siguiente paso cuando se produce un flanco de señal ascendente en el parámetro JOG. Si la instrucción ya se encuentra en el último paso y se produce un flanco de señal ascendente en el parámetro JOG, se activan las variables Q y EOD. En este caso, el parámetro DCC se pone al valor 0. La instrucción permanece en el último paso hasta que el parámetro RESET se pone al valor "1".

Además, también se puede permitir la transición al paso siguiente en función del tiempo. Para ello se debe poner al valor "1" el parámetro DRUM_EN. La transición al paso siguiente tiene lugar si está activado el bit de evento (EVENTi) correspondiente al paso actual y si ha transcurrido el tiempo predeterminado para el paso actual. El tiempo necesario para un paso resulta de la multiplicación de la base de tiempo (DTBP por el factor de tiempo válido para el paso actual (S_PRESET).

Nota

El tiempo de ejecución restante en el paso actual (DCC) se reduce únicamente si está activado el bit de evento correspondiente (EVENTi).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si, al llamar la instrucción, el parámetro RESET tiene el valor "1", el circuito secuencial sucesivo pasará al paso cuyo número se ha asignado al parámetro DSP.

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1".

El circuito secuencial sucesivo también se puede conmutar en progresión en todo momento a través del valor del parámetro JOG.

En la primera llamada de la instrucción, el parámetro RESET se debe poner a "1".

Si el circuito secuencial se encuentra en el último paso y ha transcurrido el tiempo de ejecución predefinido para este paso, se activan los parámetros Q y EOD. La instrucción permanece en el último paso hasta que el parámetro RESET adopte el valor "1".

Esta instrucción se ejecuta solo en los estados operativos ARRANQUE y RUN.

El sistema operativo inicializa la instrucción "Materializar circuito secuencial sucesivo" cuando se efectúa un arranque en frío. La instrucción no se inicializa en caso de un rearranque completo (en caliente). Si la instrucción "Materializar circuito secuencial sucesivo" debe inicializarse tras un rearranque completo (en caliente), es preciso llamarla en el OB 100 con el valor "1" en el parámetro RESET.

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

```
CALL DRUM_X, "DB instancia"
RESET := <Operando>
JOG := <Operando>
```

```
DRUM_EN := <Operando>
LST_STEP := <Operando>
EVENT1 - 16 := <Operando>
OUT0 - 15 := <Operando>
Q := <Operando>
OUT_WORD := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" desactiva el circuito secuencial. En la primera llamada de la instrucción, el parámetro RESET se debe poner a "1".
JOG	Input	BOOL	I, Q, M, D, L	Un flanco ascendente (con respecto a la última llamada) provoca la transición al paso siguiente, siempre que el circuito secuencial no se encuentre aún en el último paso. La transición tiene lugar independientemente del valor asignado al parámetro DRUM_EN.
DRUM_EN	Input	BOOL	I, Q, M, D, L	Parámetro de control que determina si se puede realizar una transición al paso siguiente en función del tiempo. Si el valor es "1", es posible una transición en función del tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso Los valores admisibles son 1 a 16.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento (i) (pertenece al paso i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	I, Q, M, D, L	Bit de salida (j) (idéntico al bit n.º j de OUT_WORD)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Q	Output	BOOL	I, Q, M, D, L	Parámetro de estado que indica si ha transcurrido el tiempo predefinido por el usuario para la ejecución del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Bits de salida resumidos en una variable
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Irrelevante para el usuario (parámetro de entrada JOG de la llamada anterior)
EOD	Static	BOOL	I, Q, M, D, L	Idéntico al parámetro de salida Q
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Número del primer paso Los valores admisibles son 1 a 16.
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Número del paso actual
DCC	Static	DWORD	I, Q, M, D, L, P	Tiempo de ejecución restante del paso actual, indicado en milisegundos (ms) (relevante solo si DRUM_EN = 1 y el bit de evento correspondiente = 1)
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo en milisegundos (ms) válida para todos los pasos
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	(Irrelevante para el usuario: hora de sistema de la llamada anterior)
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Campo unidimensional con los factores de tiempo para cada paso. Selección recomendada de índices: [1 a 16]. En este caso, S_PRESET [x] contiene el factor de tiempo del paso x.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los valores que deben emitirse en cada paso, a no ser que se hayan inhibido vía S_MASK. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, OUT_VAL [x, y] contiene el valor asignado al bit de salida OUTy en el paso x.
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los bits de máscara para cada paso. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, S_MASK [x, y] contiene el bit de máscara para el valor "y"-ésimo que debe emitirse en el paso x. Significado de los bits de máscara: • 0: el valor correspondiente del paso anterior se asigna al respectivo bit de salida • 1: el valor correspondiente de OUT_VAL se asigna al respectivo bit de salida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene los efectos siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET, OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1".
  - Los parámetros ET, OA y CA se ponen a "0".
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET, OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control discreto" se utiliza la siguiente sintaxis:

```
CALL DCAT, "DB instancia"
```

```
CMD := <Operando>
O_FB := <Operando>
C_FB := <Operando>
Q := <Operando>
OA := <Operando>
CA := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BOOL	I, Q, M, D, L	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms.
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms.
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
CMD_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

```
AWL                          Explicación
CALL DCAT, "DCAT_DB"         // Se llama la instrucción "Alarma de tiempo para
                             control discreto" y se crea el bloque de datos
                             instancia "DCAT_DB".
CMD := "Tag_Input_CMD"       // Abrir o cerrar el dispositivo
O_FB := "Tag_Input_O_FB"     // Realimentación al abrir
C_FB := "Tag_Input_C_FB"     // Realimentación al cerrar
Q := "Tag_Output_Q"          // Indica el estado del parámetro CMD
OA := "Tag_Output_OA"        // Salida de alarma al abrir
CA := "Tag_Output_CA"        // Salida de alarma al cerrar
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se activa la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto

Parámetros de entrada								Parámetros de salida								
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Sintaxis

Para la instrucción "Alarma de tiempo para control del motor" se utiliza la siguiente sintaxis:

```
CALL MCAT, "DB instancia"
O_CMD := <Operando>
C_CMD := <Operando>
S_CMD := <Operando>
O_FB := <Operando>
C_FB := <Operando>
OO := <Operando>
CO := <Operando>
OA := <Operando>
CA := <Operando>
Q := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
O_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
O_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Abrir"
C_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Cerrar"

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
CALL MCAT, "MCAT_DB"	// Se llama la instrucción "Alarma de tiempo para control del motor" y se crea el bloque de datos instancia "MCAT_DB".
O_CMD := "Tag_Input_O_CMD"	// Entrada de la orden "Abrir"
C_CMD := "Tag_Input_C_CMD"	// Entrada de la orden "Cerrar"
S_CMD := "Tag_Input_S_CMD"	// Entrada de la orden "Parar"
O_FB := "Tag_Input_O_FB"	// Realimentación al abrir
C_FB := "Tag_Input_C_FB"	// Realimentación al cerrar
OO := "Tag_OutputOpen"	// Salida "Abrir"
CO := "Tag_OutputClosed"	// Salida "Cerrar"
OA := "Tag_Output_OA"	// Salida de alarma al abrir
CA := "Tag_Output_CA"	// Salida de alarma al cerrar
Q := "Tag_Output_Q"	// Información de error

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una correspondencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Comparar bits de entrada con bits de una máscara" se utiliza la siguiente sintaxis:

```
CALL IMC, "DB instancia"
IN_BIT0 - 15 := <Operando>
CMP_STEP := <Operando>
OUT := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente se comparan todos los 16 bits de entrada con la máscara para el paso 2. El estado lógico del parámetro OUT se pone a TRUE porque los bits de entrada concuerdan con la máscara para el paso 2.

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

```
AWL                              Explicación
CALL IMC, "IMC_DB"               // Se llama la instrucción "Comparar bits de entrada
                                 con bits de una máscara" y se crea el bloque de datos
                                 instancia "IMC_DB".
IN_BIT0 := "Tag_Input_BIT0"      // Bit de entrada 0
IN_BIT1 := "Tag_Input_BIT1"      // Bit de entrada 1
IN_BIT2 := "Tag_Input_BIT2"      // Bit de entrada 2
IN_BIT3 := "Tag_Input_BIT3"      // Bit de entrada 3
IN_BIT4 := "Tag_Input_BIT4"      // Bit de entrada 4
IN_BIT5 := "Tag_Input_BIT5"      // Bit de entrada 5
IN_BIT6 := "Tag_Input_BIT6"      // Bit de entrada 6
IN_BIT7 := "Tag_Input_BIT7"      // Bit de entrada 7
IN_BIT8 := "Tag_Input_BIT8"      // Bit de entrada 8
IN_BIT9 := "Tag_Input_BIT9"      // Bit de entrada 9
IN_BIT10 := "Tag_Input_BIT10"    // Bit de entrada 10
IN_BIT11 := "Tag_Input_BIT11"    // Bit de entrada 11
IN_BIT12 := "Tag_Input_BIT12"    // Bit de entrada 12
IN_BIT13 := "Tag_Input_BIT13"    // Bit de entrada 13
IN_BIT14 := "Tag_Input_BIT14"    // Bit de entrada 14
IN_BIT15 := "Tag_Input_BIT15"    // Bit de entrada 15
CMP_STEP := "Tag_CMP_STEP"       // Número de paso de la máscara
OUT := "Tag_Output"              // Indica si se ha encontrado una correspondencia
ERR_CODE := "Tag_ErrorCode"      // Información de error
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE

Parámetro	Operando	Valor
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
CMP_STEP	Tag_CMP_STEP	B#16#02
OUT	Tag_Output	FALSE
ERR_CODE	Tag_ErrorCode	W#16#0000

En el bloque de datos instancia IMC_DB de la instrucción están almacenados los siguientes valores para la máscara del paso 2:

Parámetro	Dirección	Valor
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output	TRUE
ERR_CODE	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Barrido de matriz" se utiliza la siguiente sintaxis:

```
CALL SMC, "DB instancia"
IN_BIT0 - 15 := <Operando>
OUT := <Operando>
OUT_STEP := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
LAST	Static	BYTE	I, Q, M, D, L, P	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY of WORD	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

En el ejemplo siguiente, todos los 16 bits de entrada se comparan con la máscara para los pasos 0 a 5 hasta que se encuentre una correspondencia. Se consultan únicamente las máscaras para los pasos 0 a 2, ya que la máscara para el paso 2 se corresponde con el bit de entrada.

Nota

Los parámetros estáticos pueden inicializarse en el bloque.

AWL	Explicación
CALL SMC, "SMC_DB"	// Se llama la instrucción "Barrido de matriz" y se crea el bloque de datos instancia "SMC_DB".
IN_BIT0 := "Tag_Input_BIT0"	// Bit de entrada 0

```
AWL                              Explicación
IN_BIT1 := "Tag_Input_BIT1"    // Bit de entrada 1
IN_BIT2 := "Tag_Input_BIT2"    // Bit de entrada 2
IN_BIT3 := "Tag_Input_BIT3"    // Bit de entrada 3
IN_BIT4 := "Tag_Input_BIT4"    // Bit de entrada 4
IN_BIT5 := "Tag_Input_BIT5"    // Bit de entrada 5
IN_BIT6 := "Tag_Input_BIT6"    // Bit de entrada 6
IN_BIT7 := "Tag_Input_BIT7"    // Bit de entrada 7
IN_BIT8 := "Tag_Input_BIT8"    // Bit de entrada 8
IN_BIT9 := "Tag_Input_BIT9"    // Bit de entrada 9
IN_BIT10 := "Tag_Input_BIT10"  // Bit de entrada 10
IN_BIT11 := "Tag_Input_BIT11"  // Bit de entrada 11
IN_BIT12 := "Tag_Input_BIT12"  // Bit de entrada 12
IN_BIT13 := "Tag_Input_BIT13"  // Bit de entrada 13
IN_BIT14 := "Tag_Input_BIT14"  // Bit de entrada 14
IN_BIT15 := "Tag_Input_BIT15"  // Bit de entrada 15
OUT := "Tag_Output"            // Indica si se ha encontrado una correspondencia.
OUT_STEP := "Tag_Output_STEP"  // Contiene el número de paso con la máscara
                               correspondiente.
ERR_CODE := "Tag_ErrorCode"    // Información de error
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
IN_BIT0	Tag_Input_BIT0	TRUE
IN_BIT1	Tag_Input_BIT1	TRUE
IN_BIT2	Tag_Input_BIT2	FALSE
IN_BIT3	Tag_Input_BIT3	TRUE
IN_BIT4	Tag_Input_BIT4	TRUE
IN_BIT5	Tag_Input_BIT5	FALSE
IN_BIT6	Tag_Input_BIT6	TRUE
IN_BIT7	Tag_Input_BIT7	TRUE
IN_BIT8	Tag_Input_BIT8	FALSE
IN_BIT9	Tag_Input_BIT9	TRUE
IN_BIT10	Tag_Input_BIT10	TRUE
IN_BIT11	Tag_Input_BIT11	FALSE
IN_BIT12	Tag_Input_BIT12	TRUE
IN_BIT13	Tag_Input_BIT13	TRUE
IN_BIT14	Tag_Input_BIT14	FALSE
IN_BIT15	Tag_Input_BIT15	TRUE
OUT	Tag_Output	FALSE

Parámetro	Operando	Valor
OUT_STEP	Tag_Output_STEP	B#16#00
ERR_CODE	Tag_ErrorCode	W#16#0000

En el bloque de datos instancia SMC_DB de la instrucción están almacenados los siguientes valores para la máscara del paso 2:

Parámetro	Dirección	Valor
CMP_VAL [2,0]	DBX12.0	TRUE
CMP_VAL [2,1]	DBX12.1	TRUE
CMP_VAL [2,2]	DBX12.2	FALSE
CMP_VAL [2,3]	DBX12.3	TRUE
CMP_VAL [2,4]	DBX12.4	TRUE
CMP_VAL [2,5]	DBX12.5	FALSE
CMP_VAL [2,6]	DBX12.6	TRUE
CMP_VAL [2,7]	DBX12.7	TRUE
CMP_VAL [2,8]	DBX13.0	FALSE
CMP_VAL [2,0]	DBX13.1	TRUE
CMP_VAL [2,10]	DBX13.2	TRUE
CMP_VAL [2,11]	DBX13.3	FALSE
CMP_VAL [2,12]	DBX13.4	TRUE
CMP_VAL [2,13]	DBX13.5	TRUE
CMP_VAL [2,14]	DBX13.6	FALSE
CMP_VAL [2,15]	DBX13.7	TRUE
LAST	DB84	B#16#05

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output	TRUE
OUT_STEP	Tag_Output_STEP	B#16#02
ERR_CODE	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor de la ganancia en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse como supresor de perturbaciones o filtro de paso bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

```
CALL LEAD_LAG, "DB instancia"
IN := <Operando>
SAMPLE_T := <Operando>
OUT := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

AWL	Explicación
CALL LEAD_LAG, "LEAD_LAG_DB"	// Se llama la instrucción "Algoritmo Lead y Lag" y se crea el bloque de datos instancia "LEAD_LAG_DB".
IN := "Tag_Input"	// Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T := "Tag_Input_SAMPE_T"	// Tiempo de muestreo
OUT := "Tag_Output_Result"	// Resultado de la instrucción
ERR_CODE := "Tag_ErrorCode"	// Información de error

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_InputSampleTime	10

En el bloque de datos instancia "LEAD_LAG_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

TONR_X: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PV. Si el estado lógico del parámetro TMR_EN cambia a "1", se ejecuta la instrucción y se empieza a contar el tiempo PV. La instrucción "Acumulador de tiempo" suma el valor del parámetro DELTA_T al del parámetro ET siempre que se cumplan las siguientes condiciones:

- El parámetro TMR_EN devuelve el estado lógico "1".
- El valor de tiempo del parámetro ET es menor que el del parámetro PV.
- El parámetro RESET devuelve el estado lógico "0".

Si el valor del parámetro TMR_EN cambia a "0", el tiempo del parámetro DELTA_T se dejará de sumar al valor de tiempo del parámetro ET.

Cuando el valor de tiempo del parámetro ET alcanza el del parámetro PV, el parámetro Q devuelve el estado lógico "1". Mientras el parámetro Q devuelva el estado lógico "1", el valor del parámetro ET no se modificará. El valor del parámetro ET se pone a "0" cuando el parámetro RESET se pone a "1".

Puesto que la instrucción "Acumulador de tiempo" utiliza el tiempo de ejecución (DELTA_T) del último ciclo del bloque de organización (OB), la instrucción solo puede llamarse en bloques de organización cíclicos.

Nota

El tiempo de ciclo del bloque de organización debe desplazarse desde la información de arranque a la variable global del parámetro DELTA_T.

Sintaxis

Para la instrucción "Acumulador de tiempo" se utiliza la siguiente sintaxis:

```
CALL TONR_X
TMR_EN := <Operando>
RESET := <Operando>
PV := <Operando>
DELTA_T := <Operando>
Q := <Operando>
ET := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TMR_EN	Input	BOOL	I, Q, M, D, L	Activa el temporizador para acumular el valor de tiempo.
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
PV	Input	DINT	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo
DELTA_T	Input	INT	I, Q, M, D, L o constante	Tiempo de ciclo del OB del ciclo anterior
Q	Output	BOOL	Q, M, D, L	Salida que se activa cuando el valor del parámetro ET es mayor o igual al valor del parámetro PV.
ET	InOut	DINT	I, Q, M, D, L	Valor de tiempo actual del tiempo transcurrido

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL TONR_X	// Se llama la instrucción "Acumulador de tiempo".
TMR_EN := "Tag_TMR_EN"	// Entrada de arranque

AWL	Explicación
RESET := "Tag_ResetTimer"	// Entrada de inicialización
PV := "Tag_Input_PV"	// Duración máxima del cronometraje
DELTA_T := "Tag_DELTA_T"	// Tiempo de ciclo del OB del ciclo anterior
ET := "Tag_Input_ET"	// Tiempo acumulado
Q := "Tag_Ouput_Q"	// Indica si se ha alcanzado el tiempo del parámetro PV.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

WSR: Guardar datos en el registro de desplazamiento

Descripción

La instrucción "Guardar datos en el registro de desplazamiento" escribe datos en un registro de desplazamiento desde el origen indicado. Todo elemento del registro se desplaza a la siguiente dirección. Los datos contenidos en la última dirección del registro de desplazamiento se pierden tras la operación de desplazamiento.

Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen (S_DATA) y se desplazan a la dirección inicial (START) del registro de desplazamiento. El parámetro LEN indica el número de elementos que deben desplazarse. El tipo de datos de los elementos se define en el parámetro E_TYPE. Si se indica un tipo de datos no admisible en el parámetro E_TYPE, no se procesa la instrucción.

Si el parámetro RESET está a "1" al ejecutarse la instrucción, se borra el registro. El parámetro Q se pone al estado lógico "1" si el registro de desplazamiento está ocupado con ceros o si se borra.

Sintaxis

Para la instrucción "Guardar datos en el registro de desplazamiento" se utiliza la siguiente sintaxis:

```
CALL WSR
RESET := <Operando>
S_DATA := <Operando>
START := <Operando>
LEN := <Operando>
E_TYPE := <Operando>
Q := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Guardar datos en el registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_DATA	Input	POINTER	I, Q, M, D	Apunta al elemento de origen que se desplaza a la dirección inicial (START). En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
START	Input	POINTER	I, Q, M, D	Apunta a la dirección inicial del registro. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
LEN	Input	WORD	I, Q, M, D, L, P	Número de elementos que deben desplazarse.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de los elementos que deben desplazarse. Los siguientes tipos de datos son admisibles: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	Q, M, D, L	Indica "1" si el parámetro RESET está activado (1) o si todos los elementos que deben desplazarse tienen el valor "0".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente se desplazan cuatro elementos del tipo de datos WORD a partir de la dirección indicada en el parámetro START. Una vez desplazado el primer elemento del registro a la siguiente dirección, la primera dirección se ocupará con los datos a los que señala el puntero en el parámetro S_DATA. El valor del último elemento del registro se pierde entonces.

AWL	Explicación
CALL WSR	// Se llama la instrucción "Guardar datos en el registro de desplazamiento".
RESET := "Tag_Input_Reset"	// Entrada de inicialización
S_DATA := "Tag_Input_DataSource"	// Elemento de origen que se desplaza a la dirección inicial.
START := "Tag_Input_StartAddress"	// Dirección inicial
LEN := "Tag_Input_LEN"	// Número de elementos que deben desplazarse
E_TYPE := "Tag_Input_ElementType"	// Tipo de datos de los elementos
Q := "Tag_Output_Q"	// Estado del registro

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_SourceData	P#DB2.DBX6.0 DW6 = W#16#1234
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
LEN	Tag_Input_LEN	W#16#04
E_TYPE	Tag_Input_ElementType	W#16#04
Q	Tag_Output_Q	1

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_SourceData	P#DB2.DBX6.0 DW6 = W#16#1234
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
LEN	Tag_Input_LEN	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

SHRB: Desplazar bit al registro de desplazamiento

Descripción

La instrucción "Desplazar bit al registro de desplazamiento" escribe un bit en un registro de desplazamiento desde el origen indicado (DATA). Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen y se desplazan a la dirección inicial (S_BIT) del registro de desplazamiento. Los restantes bits se desplazan una posición. El bit de la última dirección (S_BIT + N) se pierde tras el desplazamiento. Si el parámetro RESET está a "1" mientras se ejecuta la instrucción, las direcciones del registro de desplazamiento se ponen a "0" y no se desplazan.

Sintaxis

Para la instrucción "Guardar datos en el registro de desplazamiento" se utiliza la siguiente sintaxis:

```
CALL SHRB
DATA := <Operando>
RESET := <Operando>
S_BIT := <Operando>
N := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar bit al registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DATA	Input	BOOL	I, Q, M, D, L	Bit de origen
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del registro de desplazamiento. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	WORD	I, Q, M, D, L, P	Número de bits que deben desplazarse.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro N se pone al valor 14 ("E" en notación hexadecimal), con lo que 14 bits se desplazan a partir de la dirección indicada en el parámetro S_BIT. Una vez desplazados los bits, la primera dirección se ocupa con los datos del origen del parámetro DATA. El último valor de bit se pierde.

AWL	Explicación
CALL SHRB	// Se llama la instrucción "Desplazar bit al registro de desplazamiento".
DATA := "Tag_Input_Data"	// Bit de origen que se desplaza a la dirección inicial.
RESET := "Tag_Input_Reset"	// Entrada de inicialización
S_BIT := "Tag_Input_BitSource"	// Dirección inicial
N := "Tag_Input_Number"	// Número de bits que deben desplazarse

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Antes de ejecutarse la instrucción, los siguientes estados lógicos están almacenados en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Después de la ejecución

Una vez ejecutada la instrucción, los siguientes estados lógicos se almacenan en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

SEG: Generar patrón de bits para el display de 7 segmentos

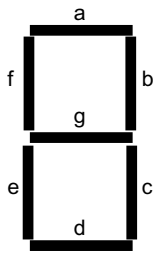
Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" convierte cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se deposita en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)
0000	00111111	0
0001	00000110	1
0010	01011011	2
0011	01001111	3
0100	01100110	4
0101	01101101	5
0110	01111101	6
0111	00000111	7

Display de siete segmentos



1000	01111111	8
1001	01100111	9
1010	01110111	A
1011	01111100	B
1100	00111001	C
1101	01011110	D
1110	01111001	E
1111	01110001	F

Sintaxis

Para la instrucción "Generar patrón de bits para el display de 7 segmentos" se utiliza la siguiente sintaxis:

```
CALL SEG
IN := <Operando>
OUT := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits para el display de 7 segmentos

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL SEG	// Se llama la instrucción "Generar patrón de bits para el display de 7 segmentos".
IN := "Tag_Input"	// Palabra de origen
OUT := "Tag_Output"	// Patrón de bits para el display de 7 segmentos

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16#065B4F66	00000110 01011011 01001111 01100110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

Sintaxis

Para la instrucción "Generar complemento a diez" se utiliza la siguiente sintaxis:

```
CALL BCDCPL
IN := <Operando>
ERR_CODE := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
CALL BCDCPL                  // Se llama la instrucción "Generar complemento a
                             diez".
IN := "Tag_Input"            // Número BCD de 7 dígitos
ERR_CODE := "Tag_Output"     // Complemento a diez del número BCD indicado
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" cuenta el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

Sintaxis

Para la instrucción "Contar número de bits activados" se utiliza la siguiente sintaxis:

```
CALL BITSUM
IN := <Operando>
RET_VAL := <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan.
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL BITSUM	// Se llama la instrucción "Contar número de bits activados".
IN := "Tag_Input"	// Operando cuyos bits activados se cuentan.
RET_VAL := "Tag_Output"	// Número de bits activados del operando "Tag_Input"

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado (Página 2547)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Instrucciones básicas

Operaciones lógicas con bits

A: Operación lógica Y

Descripción

La instrucción "Operación lógica Y" consulta si el estado lógico de un operando binario es "1" y combinar el resultado lógicamente con Y con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es idéntico al estado lógico del operando consultado.

Una vez ejecutada la instrucción, el RLO es "1" si los dos estados lógicos combinados también son "1". Si uno de los estados lógicos combinados es "0", el RLO será "0" tras ejecutar la instrucción.

Si se aplica la instrucción varias veces consecutivas, todos los resultados de las consultas deben ser "1" para que el resultado lógico común también sea "1".

Con la instrucción "Operación lógica Y" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica Y" se utiliza la siguiente sintaxis:

```
A <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL		Esquema de conexiones de relé	
		Barra de alimentación	
A	“Tag_Input_1“	Estado lógico 1	Contacto NA
A	“Tag_Input_2“	Estado lógico 1	Contacto NA
=	“Tag_Output“	Estado lógico 1	Bobina
Indica un interruptor cerrado.			

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3262)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Ejemplo de una cadena secuencial (Página 3269)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

AN: Operación lógica Y negada

Descripción

La instrucción "Operación lógica Y negada" consulta si el estado lógico de un operando binario es "0" y combinar el resultado lógicamente con Y con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es el estado lógico negado del operando consultado. Si el estado lógico del operando es "0", el resultado de la consulta es "1". Si el estado lógico del operando es "1", el resultado de la consulta es "0".

Una vez ejecutada la instrucción, el RLO es "1" si el resultado de la consulta y el RLO antes de ejecutar la instrucción devuelven el estado lógico "1".

Una vez ejecutada la instrucción, el RLO es "0" si el resultado de la consulta o el RLO antes de ejecutar la instrucción devuelven el estado lógico "0".

Con la instrucción "Operación lógica Y negada" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica Y negada" se utiliza la siguiente sintaxis:

```
AN <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y negada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL		Esquema de conexiones de relé	
		Barra de alimentación	
A	“Tag_Input_1“	Estado lógico 0	Contacto NA
AN	“Tag_Input_2“	Estado lógico 1	Contacto NC
=	“Tag_Output“	Estado lógico 0	Bobina

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3262)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

O: Operación lógica O

Descripción

La instrucción "Operación lógica O" consulta si el estado lógico de un operando binario es "1" y combinar el resultado lógicamente con O con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es idéntico al estado lógico del operando consultado.

Una vez ejecutada la instrucción, el RLO es "1" si uno de los estados lógicos combinados también es "1". Si los dos estados lógicos combinados son "0", el RLO será "0" tras ejecutar la instrucción.

Si se aplica la instrucción varias veces consecutivas, basta con que uno de los resultados de las consultas sea "1" para que el resultado lógico común también sea "1".

Con la instrucción "Operación lógica O" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O" se utiliza la siguiente sintaxis:

```
O <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL	Esquema de conexiones de relé
	Barra de alimentación
O “Tag_Input_1“	Estaco lógico 1 contacto NA (“Tag_Input_1“) / Estado lógico 0 contacto NA (“Tag_Input_2“)
O “Tag_Input_2“	
= “Tag_Output“	Estado lógico 1 / Bobina
	Indica un interruptor cerrado.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con bits (Página 3260)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

ON: Operación lógica O negada

Descripción

La instrucción "Operación lógica O negada" consulta si el estado lógico de un operando binario es "0" y combinar el resultado lógicamente con O con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es el estado lógico negado del operando consultado. Si el estado lógico del operando es "0", el resultado de la consulta es "1". Si el estado lógico del operando es "1", el resultado de la consulta es "0".

Una vez ejecutada la instrucción, el RLO es "1" si el resultado de la consulta o el RLO antes de ejecutar la instrucción devuelven el estado lógico "1". Si los dos estados lógicos combinados son "0", el RLO también será "0" tras ejecutar la instrucción.

Si se aplica la instrucción varias veces consecutivas, basta con que uno de los resultados de las consultas sea "1" para que el resultado lógico común también sea "1".

Con la instrucción "Operación lógica O negada" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O negada" se utiliza la siguiente sintaxis:

```
ON <operando>
```

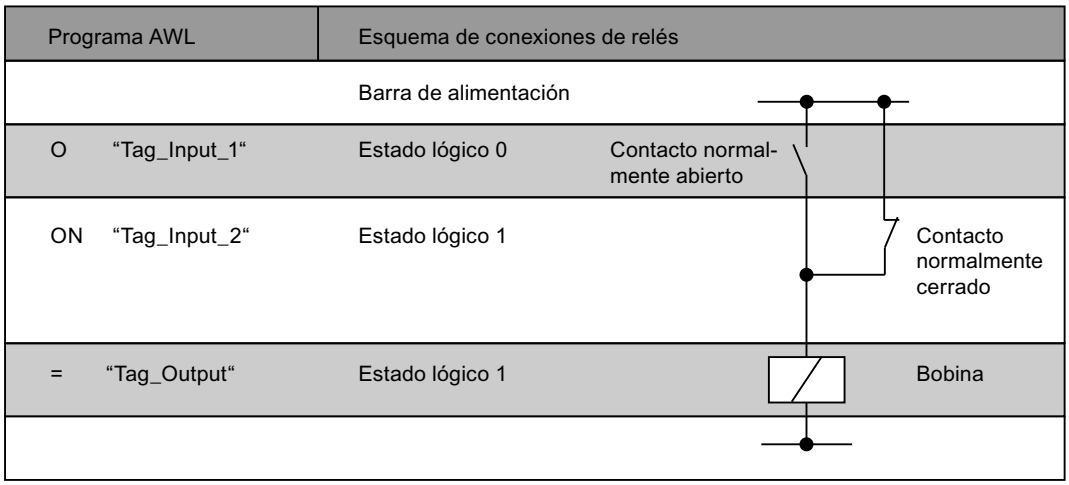
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O negada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

X: Operación lógica O-exclusiva

Descripción

La instrucción "Operación lógica O-exclusiva" consulta si el estado lógico de un operando binario es "1" y combinar el resultado lógicamente con O-exclusiva con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es idéntico al estado lógico del operando consultado. Una vez ejecutada la instrucción, el RLO es "1" si los estados lógicos combinados entre sí son distintos. El RLO es "0" si ambos estados lógicos son idénticos.

Si, p. ej., el RLO antes de ejecutar la instrucción es "1" y el operando consultado devuelve "1", el RLO se pone a "0". Si el RLO es "1" y el operando devuelve el estado lógico "0", el RLO tras ejecutar la instrucción se pondrá a "1".

Si se aplica la instrucción varias veces consecutivas, basta con que uno de los resultados de las consultas sea "1" para que el RLO común también sea "1". Si todos los resultados son "1" o "0", el RLO se pone a "0".

Con la instrucción "Operación lógica O-exclusiva" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva" se utiliza la siguiente sintaxis:

```
X <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL	Esquema de conexiones de relé
	Barra de alimentación
X “Tag_Input_1“	Contacto
X “Tag_Input_2“	Contacto
= “Tag_Output“	Bobina

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

XN: Operación lógica O-exclusiva negada

Descripción

La instrucción "Operación lógica O-exclusiva negada" consulta si el estado lógico de un operando binario es "0" y combinar el resultado lógicamente con O con el estado lógico del resultado lógico (RLO). En este caso, el resultado de la consulta es el estado lógico negado del operando consultado. Una vez ejecutada la instrucción, el RLO es "1" si los estados lógicos combinados entre sí son distintos. El RLO es "0" si ambos estados lógicos son idénticos.

Si, p. ej., el estado lógico del operando consultado es "0", el resultado de la consulta es "1". La combinación lógica con O-exclusiva del resultado de la consulta y un RLO "1" devuelve el estado lógico "0". La combinación con un RLO "0" devuelve el estado lógico "1".

Con la instrucción "Operación lógica O-exclusiva negada" también es posible consultar el estado lógico de la palabra de estado. Para ello se debe indicar el bit de estado en cuestión (==0, <>0, >0, <0, >=0, <=0, OV, OS, UO, BR) como parámetro de la instrucción.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva negada" se utiliza la siguiente sintaxis:

```
XN <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva negada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	BOOL	I, Q, M, D, L, T, C	Operando cuyo estado lógico se consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Programa AWL	Esquema de conexiones de relé
	Barra de alimentación
X "Tag_Input_1"	Contacto
XN "Tag_Input_2"	Contacto
= "Tag_Output"	Bobina

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

O: Operación lógica O de funciones Y

Descripción

La instrucción "Operación lógica O de funciones Y" ejecuta varias operaciones lógicas Y antes de una operación lógica O. El resultado global de las operaciones lógicas Y se guarda y se combina lógicamente con O con el resultado lógico (RLO) actual así como con otras posibles consultas.

Sintaxis

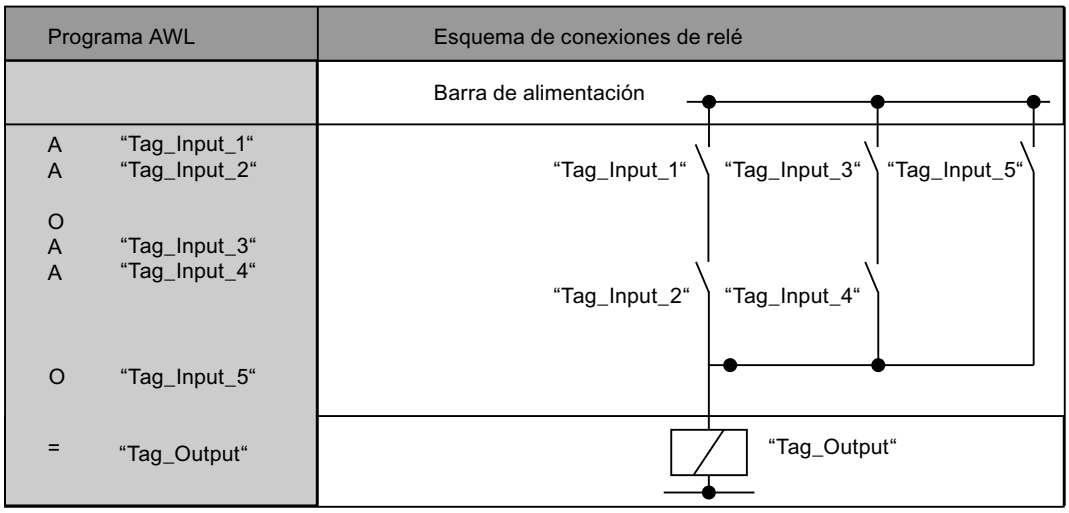
Para la instrucción "Operación lógica O de funciones Y" se utiliza la siguiente sintaxis:

```
O
```

La sintaxis de esta instrucción no requiere paréntesis.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

A(: Operación lógica Y con abrir paréntesis

Descripción

La instrucción "Operación lógica Y con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica Y.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR y el bit de estado OR. El RLO guardado se combina lógicamente con Y con el resultado lógico de toda la expresión entre paréntesis. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la función Y.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

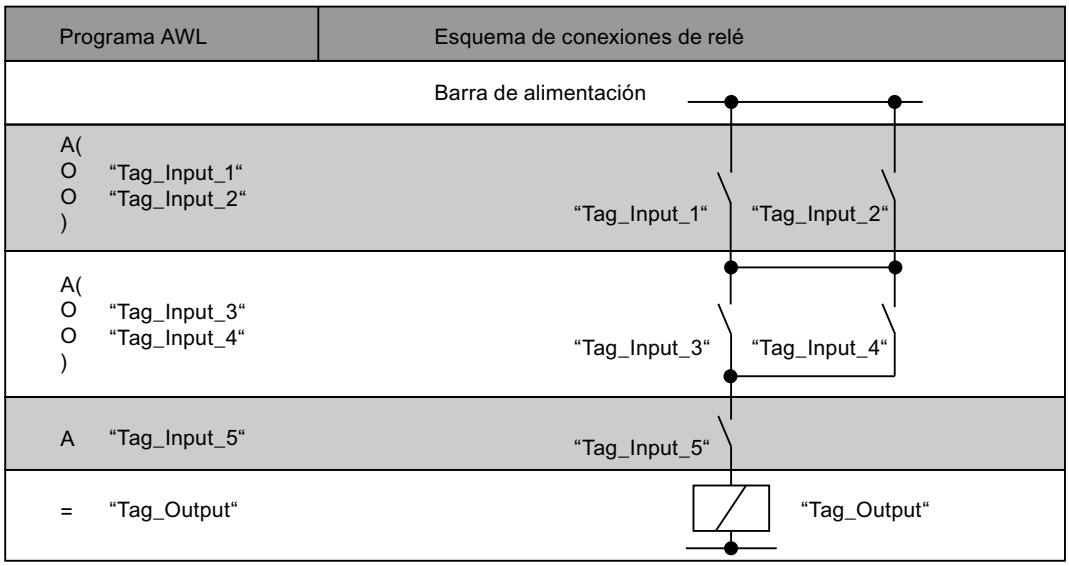
Sintaxis

Para la instrucción "Operación lógica Y con abrir paréntesis" se utiliza la siguiente sintaxis:

```
A(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

AN(: Operación lógica Y negada con abrir paréntesis

Descripción

La instrucción "Operación lógica Y negada con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica Y y negar el resultado global de la expresión entre paréntesis.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR y el bit de estado OR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO de toda la expresión entre paréntesis se niega y se combina lógicamente con Y con el RLO guardado. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la instrucción Y.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica Y negada con abrir paréntesis" se utiliza la siguiente sintaxis:

```
AN(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
AN(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con Y el RLO de la expresión entre paréntesis y negar el resultado global.
O "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con O con el RLO actual.
O "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con O con el RLO actual.
)	// Fin de la expresión entre paréntesis
AN "Tag_Input_3"	// Consultar si el estado lógico del operando es "0" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

O(: Operación lógica O con abrir paréntesis

Descripción

La instrucción "Operación lógica O con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR y el bit de estado OR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO guardado se combina lógicamente con O con el RLO de toda la expresión entre paréntesis. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la instrucción O.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O con abrir paréntesis" se utiliza la siguiente sintaxis:

```
O(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
O(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O el RLO de la expresión entre paréntesis.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.

AWL	Explicación
A "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

ON(: Operación lógica O negada con abrir paréntesis

Descripción

La instrucción "Operación lógica O negada con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O y negar el resultado global de la expresión entre paréntesis.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR y el bit de estado OR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO de la expresión entre paréntesis se niega y se combina lógicamente con O con el RLO guardado. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la instrucción O.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O negada con abrir paréntesis" se utiliza la siguiente sintaxis:

```
ON (
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
ON(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O el RLO de la expresión entre paréntesis y negar el resultado global.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

X(: Operación lógica O-exclusiva con abrir paréntesis

Descripción

La instrucción "Operación lógica O-exclusiva con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O-exclusiva.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR y el bit de estado OR. Después de procesar las instrucciones que están en la expresión entre paréntesis, el RLO guardado se combina lógicamente con O-exclusiva con el RLO de toda la expresión entre paréntesis. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la función O-exclusiva.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva con abrir paréntesis" se utiliza la siguiente sintaxis:

```
X(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
X(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O-exclusiva el RLO de la expresión entre paréntesis.
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
X "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

XN(: Operación lógica O-exclusiva negada con abrir paréntesis

Descripción

La instrucción "Operación lógica O-exclusiva negada con abrir paréntesis" procesa las instrucciones de una expresión entre paréntesis antes de una operación lógica O-exclusiva y negar el resultado global de la expresión entre paréntesis.

Al ejecutar la instrucción, la CPU guarda el resultado lógico (RLO) actual así como el resultado binario BR y el bit de estado OR. Después de procesar las instrucciones que están en la

expresión entre paréntesis, el RLO de la expresión entre paréntesis se niega y se combina lógicamente con O-exclusiva con el RLO guardado. Si a la expresión entre paréntesis le siguen otras consultas, también se combinan con la función O-exclusiva.

Las expresiones entre paréntesis se anidan programando en ellas una instrucción que también contenga una expresión entre paréntesis. La profundidad de anidamiento está limitada a 7 niveles.

Sintaxis

Para la instrucción "Operación lógica O-exclusiva negada con abrir paréntesis" se utiliza la siguiente sintaxis:

```
XN(
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
XN(	// Principio de la expresión entre paréntesis
	// Combinar lógicamente con O-exclusiva el RLO de la expresión entre paréntesis y negar el resultado global.
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
)	// Fin de la expresión entre paréntesis
XN "Tag_Input_3"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Asignar el estado lógico del RLO al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Consultar el bit de estado con instrucciones de operaciones lógicas con bits (Página 2556)

): Cerrar paréntesis

Descripción

La instrucción "Cerrar paréntesis" determina el final de una expresión entre paréntesis. Tras ejecutar la instrucción, el resultado global de la expresión entre paréntesis se combina lógicamente con la instrucción que se ha utilizado para abrir la expresión entre paréntesis.

Para abrir una expresión entre paréntesis se utilizan las instrucciones siguientes:

- A(: Operación lógica Y con abrir paréntesis
- AN(: Operación lógica Y negada con abrir paréntesis
- O(: Operación lógica O con abrir paréntesis
- ON(: Operación lógica O negada con abrir paréntesis
- X(: Operación lógica O-exclusiva con abrir paréntesis
- XN(: Operación lógica O-exclusiva negada con abrir paréntesis

Después de la instrucción "Cerrar paréntesis", la CPU pone el resultado binario BR al estado lógico que tenía antes de la expresión entre paréntesis.

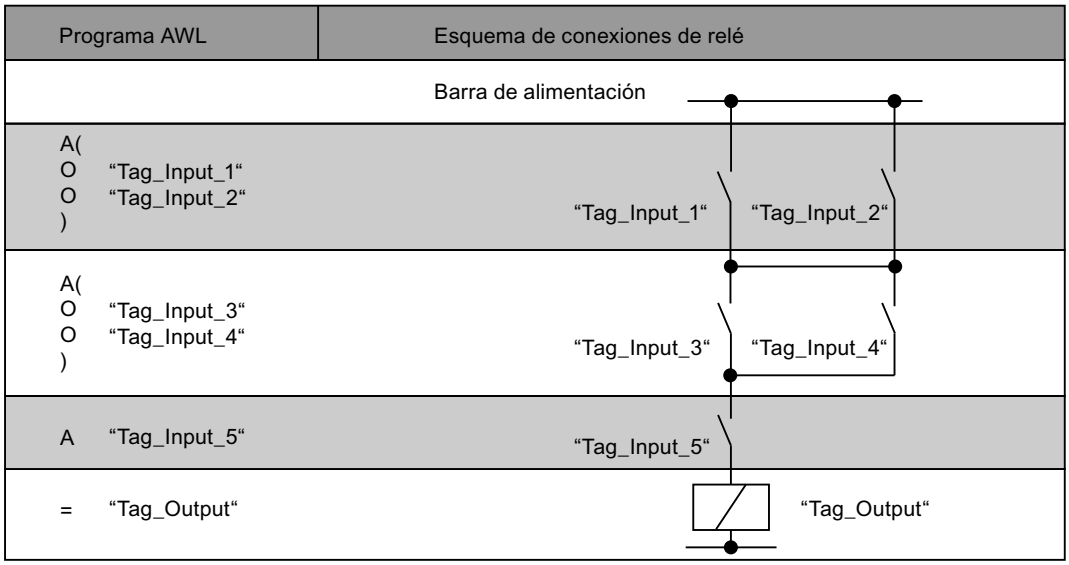
Sintaxis

Para la instrucción "Cerrar paréntesis" se utiliza la siguiente sintaxis:

```
)
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

Consultar el bit de estado en AWL (Página 2556)

=: Asignación

Descripción

La instrucción "Asignación" asigna a un operando concreto el estado lógico del resultado lógico (RLO) guardado en la CPU. Si el RLO tiene el estado lógico "1", se activa el operando. Si el estado lógico de la entrada es "0", el operando se pone a "0".

La ejecución de la instrucción "Asignación" se ve influida por el Master Control Relay (MCR). Si el bit MCR se pone a "0", el operando indicado adopta el valor "0".

Sintaxis

Para la instrucción "Asignación" se utiliza la siguiente sintaxis:

```
= <operando>
```

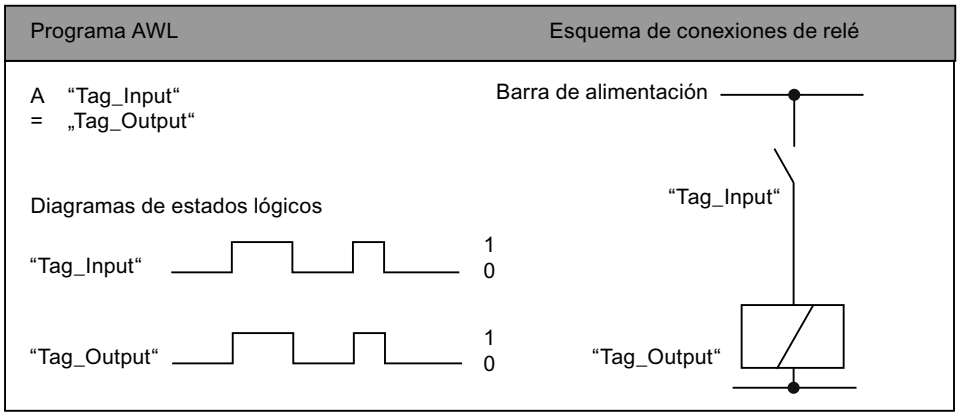
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Asignación":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando al que se asigna el RLO actual.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Ejemplo de una cadena secuencial (Página 3269)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

R: Desactivar

Descripción

La instrucción "Desactivar" pone a "0" el estado lógico de un operando concreto.

La instrucción solo se ejecuta si el resultado lógico (RLO) actual es "1". Tras ejecutar la instrucción, el operando indicado se pone a "0". Si el RLO actual es "0", el estado lógico del operando indicado no cambia.

La ejecución de la instrucción "Desactivar" se ve influida por el Master Control Relay (MCR). Si el bit MCR está puesto a "0", el estado lógico del operando indicado no cambia.

Sintaxis

Para la instrucción "Desactivar" se utiliza la siguiente sintaxis:

```
R <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se desactiva si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "Tag_Output"	// Poner el operando a "0" si el RLO = "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con bits (Página 3260)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3262)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

S: Activar

Descripción

La instrucción "Activar" pone a "1" el estado lógico de un operando concreto.

La instrucción solo se ejecuta si el resultado lógico (RLO) actual es "1". Tras ejecutar la instrucción, el operando indicado se pone a "1". Si el RLO actual es "0", el estado lógico del operando indicado no cambia.

La ejecución de la instrucción "Activar" se ve influida por el Master Control Relay (MCR). Si el bit MCR está puesto a "0", el estado lógico del operando indicado no cambia.

Sintaxis

Para la instrucción "Activar" se utiliza la siguiente sintaxis:

```
S <Operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Operando que se activa si el RLO = "1".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
S "Tag_Output"	// Poner el operando a "1" si el RLO = "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con bits (Página 3260)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3262)
El MCR (Master Control Relay) (Página 2564)
Indicaciones importantes sobre el uso de la función MCR (Página 2565)
Principios básicos de la palabra de estado (Página 2545)
Activar el bit de estado en operaciones lógicas con bits (Página 2547)

NOT: Invertir RLO

Descripción

La instrucción "Invertir RLO" invierte el estado lógico del resultado lógico (RLO). La instrucción "Invertir RLO" puede aplicarse en cualquier posición, incluso dentro de una operación lógica.

Sintaxis

Para la instrucción "Invertir RLO" se utiliza la siguiente sintaxis:

```
NOT
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_2"	// Consultar si el estado lógico del operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_3"	// Consultar si el estado lógico del operando "Tag_Input_3" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
NOT	// Invertir el estado lógico del RLO
= "Tag_Output"	// Asignar el estado lógico del RLO al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Input_1	1	1	1	0	0	0	1	0
Tag_Input_2	0	1	1	0	1	1	0	1
Tag_Input_3	0	0	1	0	0	1	1	1
Tag_Output	1	1	0	1	1	1	1	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Principios básicos de la palabra de estado (Página 2545)
Activar el bit de estado en operaciones lógicas con bits (Página 2547)

SET: Poner RLO a 1

Descripción

La instrucción "Poner RLO a 1" pone el resultado lógico (RLO) actual al estado lógico "1".

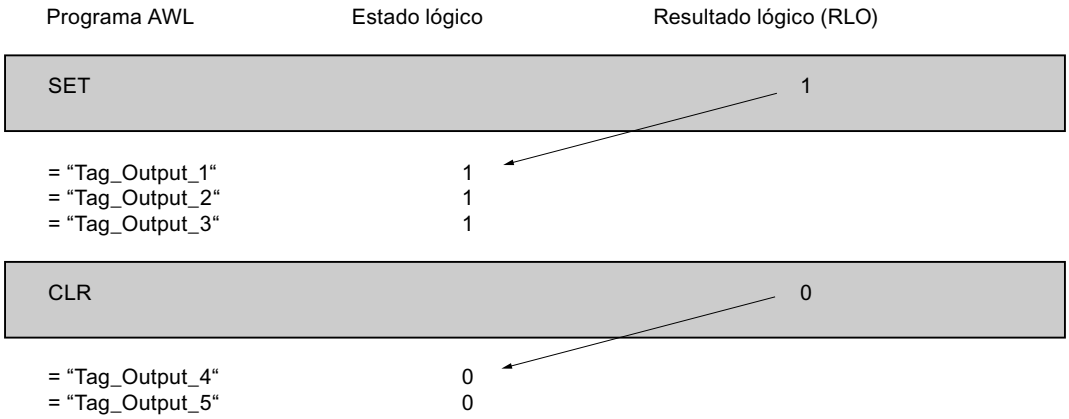
Sintaxis

Para la instrucción "Poner RLO a 1" se utiliza la siguiente sintaxis:

```
SET
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)
Principios básicos de la palabra de estado (Página 2545)
Activar el bit de estado en operaciones lógicas con bits (Página 2547)

CLR: Poner RLO a 0

Descripción

La instrucción "Poner RLO a 0" pone el resultado lógico (RLO) actual al estado lógico "0".

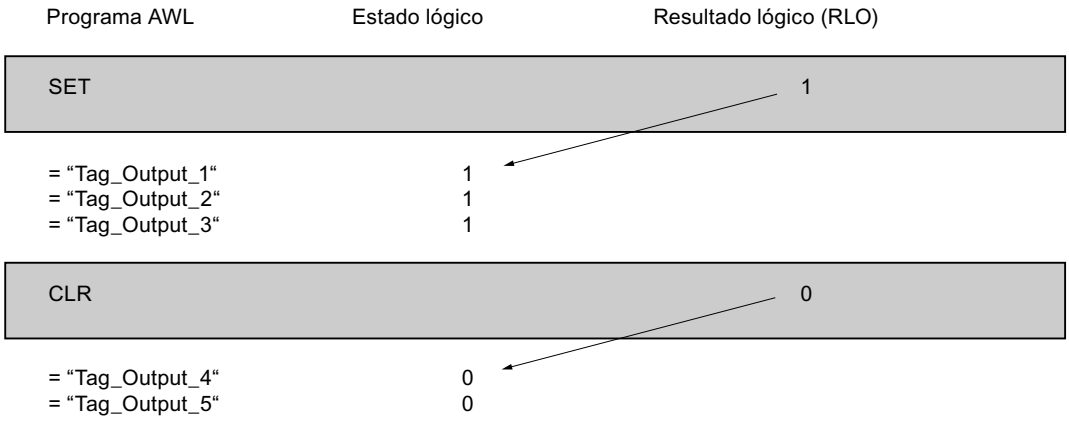
Sintaxis

Para la instrucción "Poner RLO a 0" se utiliza la siguiente sintaxis:

```
CLR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Principios básicos de la palabra de estado (Página 2545)

SAVE: Guardar RLO en bit BR

Descripción

La instrucción "Guardar RLO en bit BR" guarda el resultado lógico (RLO) en el resultado binario (BR). Durante la ejecución, la instrucción transfiere el estado lógico del resultado lógico actual

al bit de estado BR. La instrucción trabaja independientemente de las condiciones y no influye en ningún otro bit de estado.

Nota

Tras aplicar la instrucción "Guardar RLO en bit BR", el bit BR puede ser modificado nuevamente por instrucciones posteriores del mismo bloque o de bloques de nivel inferior.

La instrucción "Guardar RLO en bit BR" también se utiliza para controlar el estado de ejecución de un bloque. Si, p. ej., la instrucción "Guardar RLO en bit BR" se aplica al final de un bloque, el bit BR se pone al estado lógico del resultado lógico actual del bloque.

El bit BR ayuda a la realización del mecanismo EN/ENO en llamadas de bloques. Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Guardar RLO en bit BR" se utiliza la siguiente sintaxis:

```
SAVE
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico del operando "Tag_Input" es "1".
SAVE	// Transferir el estado lógico del RLO al bit BR.
BEC	// Si el RLO = "1", finalizar el bloque.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

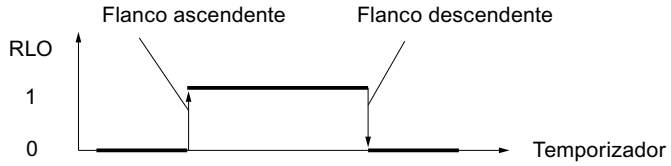
Ejemplo de la emulación del mecanismo EN/ENO en AWL (Página 2544)

FN: Consultar flanco de señal descendente del RLO

Descripción

La instrucción "Consultar flanco de señal descendente del RLO" consulta un cambio del estado lógico del resultado lógico (RLO) de "1" a "0". La instrucción compara el estado lógico actual del resultado lógico (RLO) con el estado lógico de la consulta anterior, que está guardado en el operando de la instrucción. Si la instrucción detecta un cambio del RLO de "1" a "0", significa que hay un flanco descendente.

La figura siguiente muestra el cambio del RLO en caso de flanco descendente y ascendente:



El flanco descendente se consulta cada vez que se ejecuta la instrucción. Si la instrucción detecta un flanco descendente, el RLO se pone al estado lógico "1" tras evaluar el flanco. Si no se detecta ningún flanco descendente, el estado lógico del resultado lógico es "0".

Nota

El operando de la evaluación de flanco debe encontrarse en el área de memoria de marcas o en un bloque de datos.

Sintaxis

Para la instrucción "Consultar flanco de señal descendente del RLO" se utiliza la siguiente sintaxis:

```
FN <Operando>
```

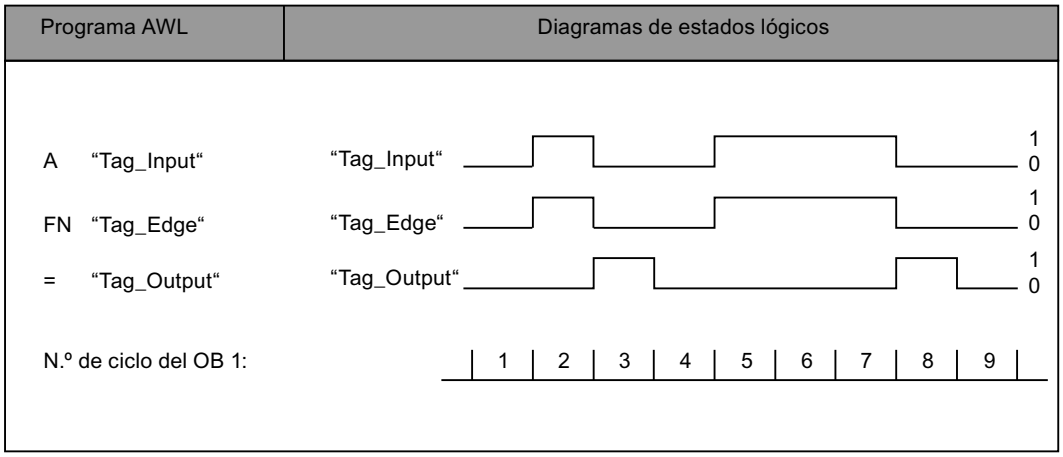
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal descendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el RLO de la última consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

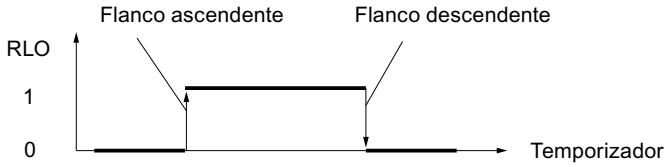
Activar el bit de estado en operaciones lógicas con bits (Página 2547)

FP: Consultar flanco de señal ascendente del RLO

Descripción

La instrucción "Consultar flanco de señal ascendente del RLO" consulta un cambio del estado lógico del resultado lógico (RLO) de "0" a "1". La instrucción compara el estado lógico actual del resultado lógico (RLO) con el estado lógico de la consulta anterior, que está guardado en el operando de la instrucción. Si la instrucción detecta un cambio del RLO de "0" a "1", significa que hay un flanco ascendente.

La figura siguiente muestra el cambio del RLO en caso de flanco ascendente y descendente:



El flanco ascendente se consulta cada vez que se ejecuta la instrucción. Si la instrucción detecta un flanco ascendente, el RLO se pone al estado lógico "1" tras evaluar el flanco. Si no se detecta ningún flanco ascendente, el estado lógico del resultado lógico es "0".

Nota

El operando de la evaluación de flanco debe encontrarse en el área de memoria de marcas o en un bloque de datos.

Sintaxis

Para la instrucción "Consultar flanco de señal ascendente del RLO" se utiliza la siguiente sintaxis:

```
FP <Operando>
```

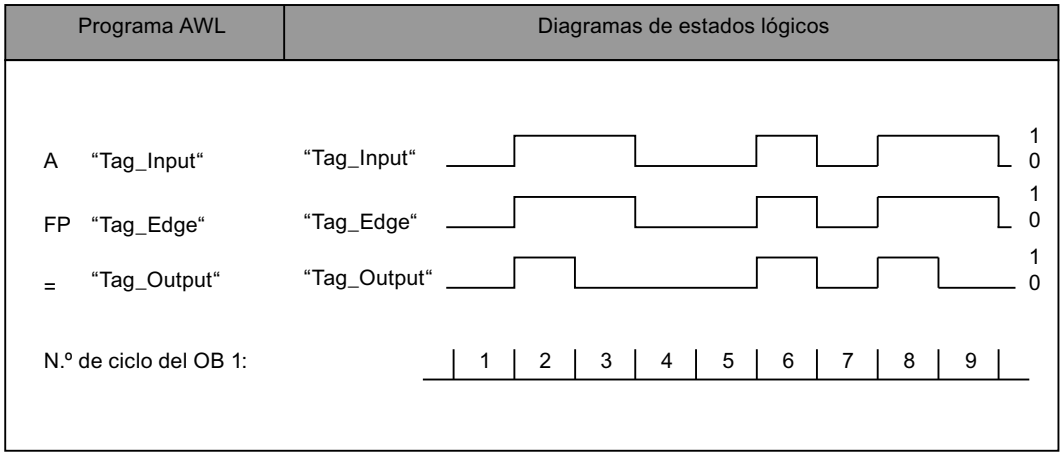
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Consultar flanco de señal ascendente del RLO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Output	BOOL	I, Q, M, D, L	Marca de flancos en la que se almacena el RLO de la última consulta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de detección del sentido de marcha de una cinta transportadora (Página 3262)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en operaciones lógicas con bits (Página 2547)

Temporizadores

FR: Habilitar temporizador

Descripción

La instrucción "Habilitar temporizador" provoca el rearranque de un temporizador. La instrucción se ejecuta con un flanco de señal ascendente y desactiva la marca de flancos interna para arrancar un temporizador. Al procesar una instrucción para arrancar el temporizador con un resultado lógico "1", el temporizador arranca de nuevo, aunque no haya ningún flanco de señal ascendente antes de la instrucción de arranque. Si el resultado lógico es "0" antes de una instrucción de arranque, la desactivación de la marca de flancos interna no influye en el temporizador.

Un cambio del resultado lógico a "0" antes de la instrucción "Habilitar temporizador" no tiene repercusión una vez rearrancado el temporizador.

En el acumulador 1 debe haber un valor de tiempo válido en formato S5TIME, aunque al procesar una función de arranque no se inicie el temporizador.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Habilitar temporizador" se utiliza la siguiente sintaxis:

```
FR <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Habilitar temporizador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que se habilita

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_EnableInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyTimer"	// Habilitar temporizador con un flanco de señal ascendente.
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

L: Cargar valor del temporizador (Página 6189)

L: Cargar valor del temporizador

Descripción

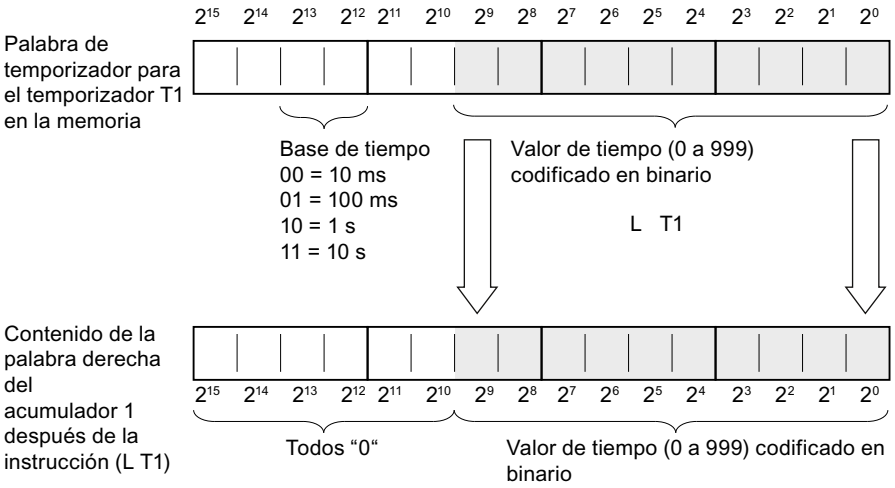
La instrucción "Cargar valor del temporizador" carga el valor de un temporizador concreto en el acumulador 1. El valor de tiempo en código binario se carga sin base de tiempo. En lugar de la base de tiempo se escriben ceros en el acumulador 1.

Después de la carga, el valor que está en el acumulador 1 equivale a un número positivo del tipo de datos INT.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

La figura siguiente muestra, a modo de ejemplo, cómo se carga en valor del temporizador en el acumulador:



Sintaxis

Para la instrucción "Cargar valor del temporizador" se utiliza la siguiente sintaxis:

```
L <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor del temporizador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador cuyo valor actual se carga

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_EnableInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyTimer"	// Habilitar temporizador con un flanco de señal ascendente.
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

LC: Cargar valor del temporizador codificado en BCD

Descripción

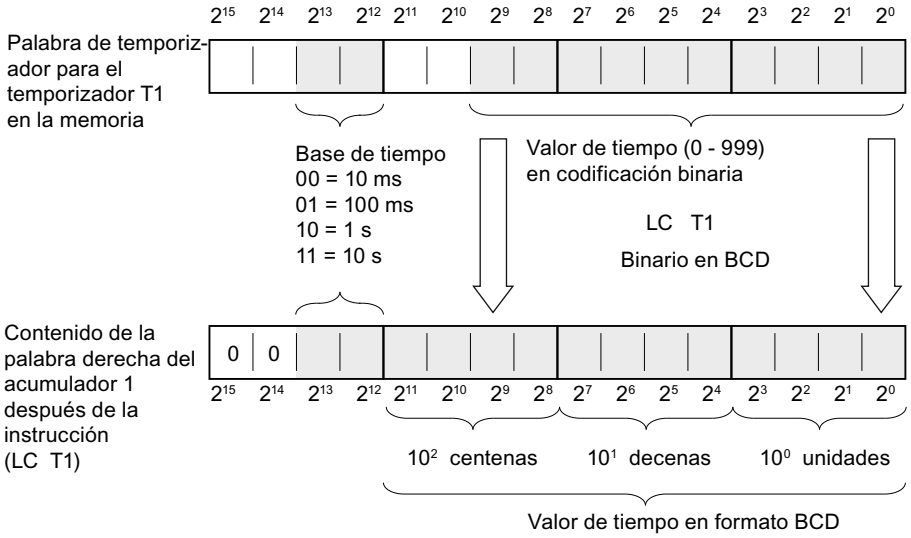
La instrucción "Cargar valor del temporizador codificado en BCD" carga el valor de un temporizador concreto en código binario en el acumulador 1 en formato BCD. Durante la carga también se transfiere la base de tiempo al acumulador 1.

Después de la carga, el valor que está en el acumulador 1 equivale a una duración en formato S5TIME. La palabra izquierda del acumulador 1 se rellena con ceros.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

La figura siguiente muestra cómo se carga en valor del temporizador en el acumulador:



Sintaxis

Para la instrucción "Cargar valor del temporizador codificado en BCD" se utiliza la siguiente sintaxis:

```
LC <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor del temporizador codificado en BCD":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador cuyo valor actual se carga codificado en BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_EnableInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyTimer"	// Habilitar temporizador con un flanco de señal ascendente.
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

R: Inicializar temporizador

Descripción

La instrucción "Inicializar temporizador" pone a "0" un temporizador concreto. La instrucción se ejecuta si el resultado lógico (RLO) actual es "1". Mientras el RLO "1" esté pendiente antes de la instrucción, el temporizador indicado dará el valor "0". La instrucción también pone a "0" el valor del temporizador y la base de tiempo de la duración programada.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Nota

La instrucción "Inicializar temporizador" no desactiva la marca de flancos interna. Para desactivar la marca de flancos interna debe programarse la instrucción "Habilitar temporizador" o ejecutarse una instrucción para arrancar el temporizador con el estado lógico "0".

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Inicializar temporizador" se utiliza la siguiente sintaxis:

```
R <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar temporizador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que se inicializa

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Si el RLO es "1", inicializar temporizador.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

SP: Arrancar temporizador como impulso

Descripción

La instrucción "Arrancar temporizador como impulso" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1, mientras el resultado lógico (RLO) antes de la instrucción siga siendo "1". Si el RLO cambia a "0" antes de finalizar el temporizador, este se detiene. Las consultas de si el estado del temporizador es "1" dan como resultado "1" mientras el temporizador está en marcha.

Es posible rearrancar el temporizador con un flanco de señal ascendente antes de la instrucción "Arrancar temporizador como impulso" o con la ayuda de la instrucción "Habilitar temporizador". Para ello, es imprescindible que el temporizador no esté desactivado.

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como impulso", el valor del temporizador se reduce de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD

válido, se ejecuta el tratamiento de errores síncronos (STOP u OB122). El bit de estado OV no se ve afectado.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como impulso" se utiliza la siguiente sintaxis:

```
SP <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como impulso

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

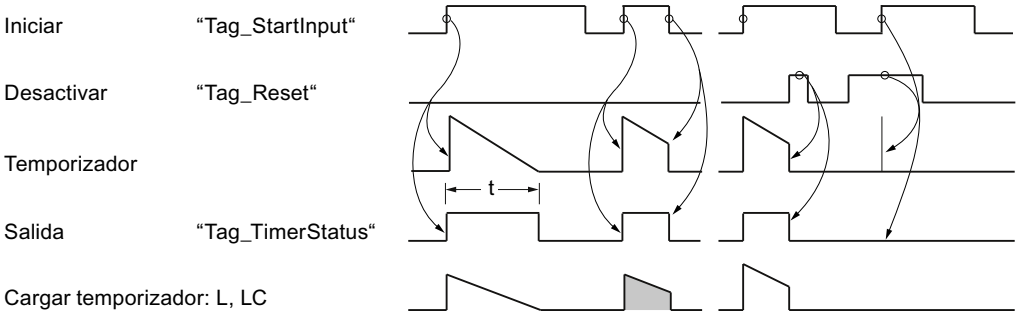
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SP "MyTimer"	// Arrancar temporizador como impulso cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.

AWL	Explicación
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

SE: Arrancar temporizador como impulso prolongado

Descripción

La instrucción "Arrancar temporizador como impulso prolongado" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con el tiempo que figura en el acumulador 1, aunque el resultado lógico antes de la instrucción

cambie a "0". Las consultas de si el estado del temporizador es "1" dan como resultado "1" mientras el temporizador está en marcha.

Con cada flanco de señal ascendente, la instrucción rearranca el temporizador con la duración programada, aunque este todavía no haya terminado.

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como impulso prolongado", el valor del temporizador decrementa de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido, se ejecuta el tratamiento de errores síncronos (STOP u OB 122). El bit de estado OV no se ve afectado.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como impulso prolongado" se utiliza la siguiente sintaxis:

```
SE <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como impulso prolongado":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como impulso prolongado

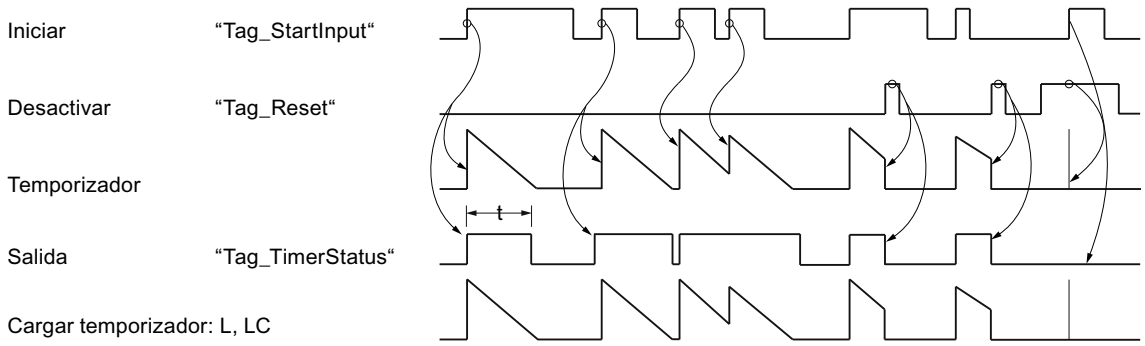
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SE "MyTimer"	// Arrancar temporizador como impulso prolongado cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

SD: Arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1, mientras el resultado lógico (RLO) antes de la instrucción siga siendo "1". Si el resultado lógico cambia a "0" antes de finalizar el temporizador, el temporizador se detiene.

Las consultas de si el estado del temporizador es "1" dan como resultado "1" si el temporizador ha terminado correctamente y el RLO actual antes de la instrucción es "1". Si el temporizador no ha terminado o el RLO cambia a "0", las consultas de si el estado del temporizador es "1" dan como resultado "0".

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como retardo a la conexión", el valor del temporizador se reduce de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido, se ejecuta el tratamiento de errores síncronos (STOP u OB 122). El bit de estado OV no se ve afectado.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como retardo a la conexión" se utiliza la siguiente sintaxis:

```
SD <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como retardo a la conexión

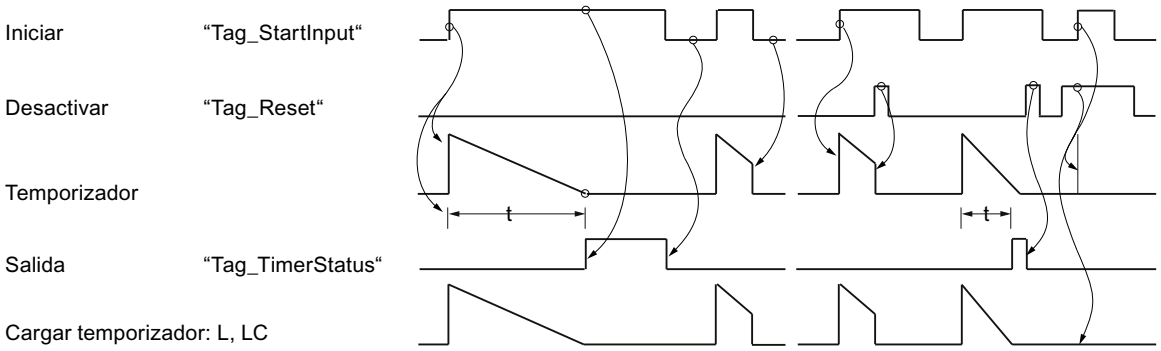
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SD "MyTimer"	// Arrancar temporizador como retardo a la conexión cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1"
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

SS: Arrancar temporizador como retardo a la conexión con memoria

Descripción

La instrucción "Arrancar temporizador como retardo a la conexión con memoria" arranca un temporizador concreto con un flanco de señal ascendente. Cuando se registra un flanco de señal ascendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1, aunque el resultado lógico (RLO) antes de la instrucción cambie a "0". Cuando el tiempo ha transcurrido, las consultas de si el estado del temporizador es "1" dan como resultado "1". El estado lógico del resultado lógico actual antes de la instrucción no influye en el resultado de la consulta.

Con cada flanco de señal ascendente, la instrucción rearranca el temporizador con la duración programada, aunque este todavía no haya terminado.

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como retardo a la conexión con memoria", el valor del temporizador se reduce de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD

válido, se ejecuta el tratamiento de errores síncronos (STOP u OB 122). El bit de estado OV no se ve afectado.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Arrancar temporizador como retardo a la conexión con memoria" se utiliza la siguiente sintaxis:

```
SS <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la conexión con memoria":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como retardo a la conexión con memoria

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

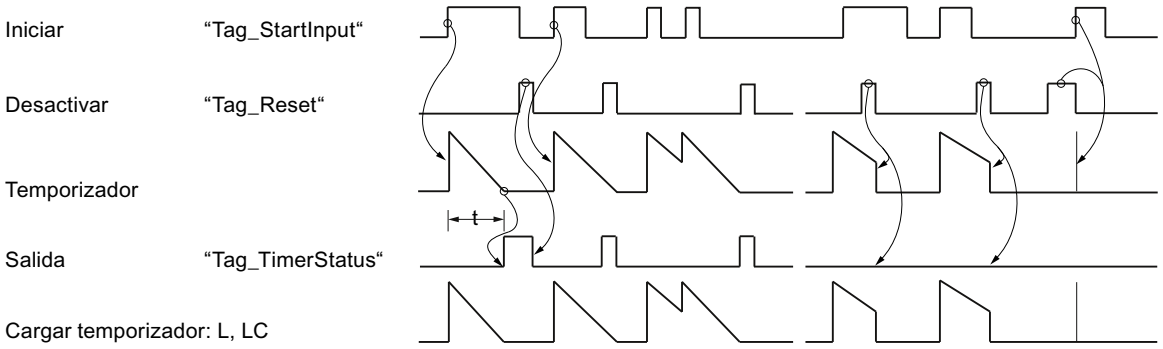
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SS "MyTimer"	// Arrancar temporizador como retardo a la conexión con memoria cuando haya un flanco de señal ascendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".

AWL	Explicación
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo:



t = tiempo programado

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

SF: Arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Arrancar temporizador como retardo a la desconexión" arranca un temporizador concreto con un flanco de señal descendente. Cuando se registra un flanco de

señal descendente, la instrucción se ejecuta y el temporizador arranca. El temporizador termina con la duración que figura en el acumulador 1. Si el resultado lógico (RLO) cambia a "1" antes de finalizar el tiempo, el temporizador se inicializa. El temporizador solo arranca de nuevo cuando se registra un flanco de señal descendente al ejecutar la instrucción.

Las consultas del estado del temporizador en "1" dan como resultado "1" si el estado del resultado lógico al ejecutar la instrucción es "1" o si el temporizador programado está en marcha. Si el temporizador no cuenta o el RLO está a "0", las consultas de si el estado del temporizador es "1" dan como resultado "0".

La duración en el acumulador 1 se compone de un valor de tiempo y una base de tiempo. Si se arranca un temporizador concreto con la instrucción "Arrancar temporizador como retardo a la desconexión", el valor del temporizador decrementa de acuerdo con la base de tiempo. El temporizador cuenta el tiempo hasta que el valor de contaje decrementa hasta cero.

En el acumulador 1 debe haber una duración válida en formato S5TIME, aunque al procesar la instrucción no arranque el temporizador. Si el acumulador 1 no tiene ningún valor BCD válido, se ejecuta el tratamiento de errores síncronos (STOP u OB 122). El bit de estado OV no se ve afectado.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Sintaxis

Para la instrucción "Arrancar temporizador como retardo a la desconexión" se utiliza la siguiente sintaxis:

```
SF <Temporizador>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Arrancar temporizador como retardo a la desconexión":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<Temporizador>	Input	TIMER	T	Temporizador que arranca como retardo a la desconexión

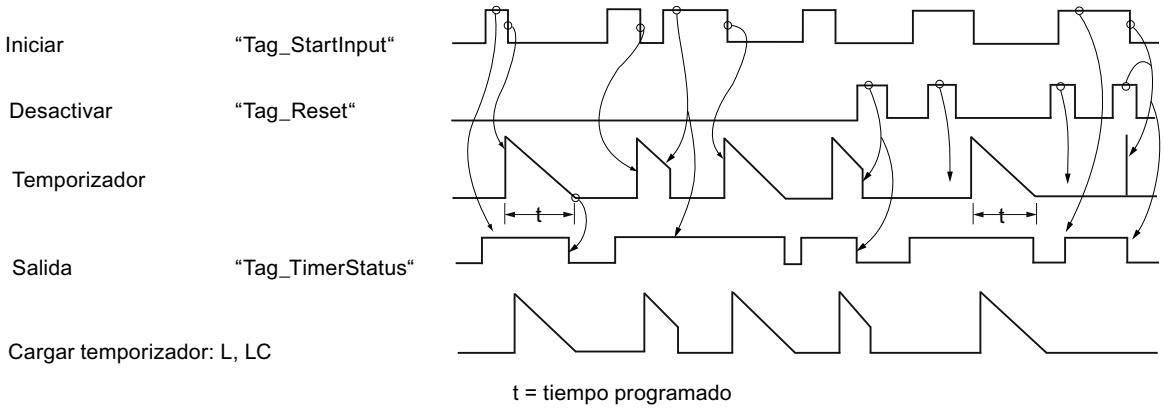
Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_StartInput"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L S5T#10s	// Cargar tiempo (10 s)
SF "MyTimer"	// Arrancar temporizador como retardo a la desconexión cuando haya un flanco de señal descendente en el operando "Tag_StartInput" o en el operando "Tag_EnableInput" y el operando "Tag_StartInput" devuelva el estado lógico "1".
A "Tag_Reset"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyTimer"	// Inicializar el temporizador si el RLO es "1" y detener el tiempo.
A "MyTimer"	// Consultar si el estado del temporizador es "1".
= "Tag_TimerStatus"	// Asignar el estado del temporizador al operando.
L "MyTimer"	// Cargar el valor actual del temporizador en el acumulador 1.
T "Tag_TimerValue"	// Transferir el valor actual del temporizador al operando "Tag_TimerValue".
LC "MyTimer"	// Cargar el valor actual del temporizador codificado en BCD en el acumulador 1.
T "Tag_TimerValue_BCD"	// Transferir el valor del temporizador codificado en BCD al operando "Tag_TimerValue_BCD".

La figura siguiente muestra el diagrama de tiempo del ejemplo.



Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

L: Cargar valor del temporizador (Página 6189)

Contadores

FR: Habilitar contador

Descripción

La instrucción "Habilitar contador" ejecuta las instrucciones "Inicializar contador", "Contador ascendente" y "Contador descendente" sin que exista actualmente un flanco de señal ascendente. Durante la ejecución, la instrucción pone la marca de flancos interna al estado lógico "0". La consulta anterior a la instrucción de contaje correspondiente debe seguir teniendo el estado lógico "1".

La instrucción "Habilitar contador" se ejecuta si el resultado lógico (RLO) cambia de "0" a "1" antes de la instrucción.

Para ejecutar una instrucción de contaje no es necesario habilitar un contador.

Sintaxis

Para la instrucción "Habilitar contador" se utiliza la siguiente sintaxis:

```
FR <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Habilitar contador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador que se habilita.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
FR "MyCounter"	// Habilitar contador
A "Tag_Input_2"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.

AWL	Explicación
CD "MyCounter"	// Decrementar contador
A "MyCounter"	// Consultar estado del contador
= "Tag_Output"	// Asignar el resultado de la consulta al operando.

En el ejemplo, el contador "MyCounter" se habilita cuando el estado lógico del operando "Tag_Input_1" cambia de "0" a "1". Si el operando "Tag_Input_2" tiene el estado lógico "1", el valor de contaje decrementa en uno al habilitar el contador. Si el estado lógico del operando "Tag_Input_2" es "0", el valor de contaje no cambia. El valor de contaje también decrementa en uno, independientemente de la habilitación del contador, cuando el estado lógico del operando "Tag_Input_2" cambia de "0" a "1".

La consulta del estado del contador da como resultado "1" si el estado del contador es mayor que cero. Si el estado del contador es igual a cero, el resultado de la consulta es "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

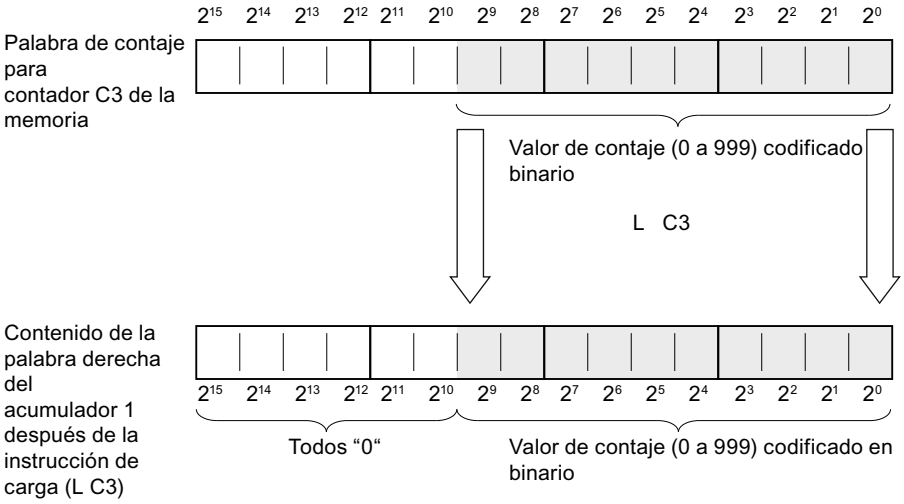
Principios básicos de la palabra de estado (Página 2545)

L: Cargar valor de contaje

Descripción

La instrucción "Cargar valor de contaje" transfiere el valor actual de un contador específico a la palabra derecha del acumulador 1 en código binario. A continuación, el valor de contaje cargado está disponible como entero de 16 bits para su posterior procesamiento.

La figura siguiente muestra el funcionamiento de la instrucción:



Sintaxis

Para la instrucción "Cargar valor de contaje" se utiliza la siguiente sintaxis:

```
L <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor de contaje":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor actual se carga.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "MyCounter"	// Cargar el valor de contaje actual en el acumulador 1 en código binario.
L "Tag_Value"	// Desplazar el valor de contaje al acumulador 2.
	// Cargar el valor del operando "Tag_Value" en el acumulador 1.
>=I	// Consulta sobre si el valor de contaje es mayor o igual al valor del operando "Tag_Value".
= "Tag_Output"	// Asignar el resultado de la consulta al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

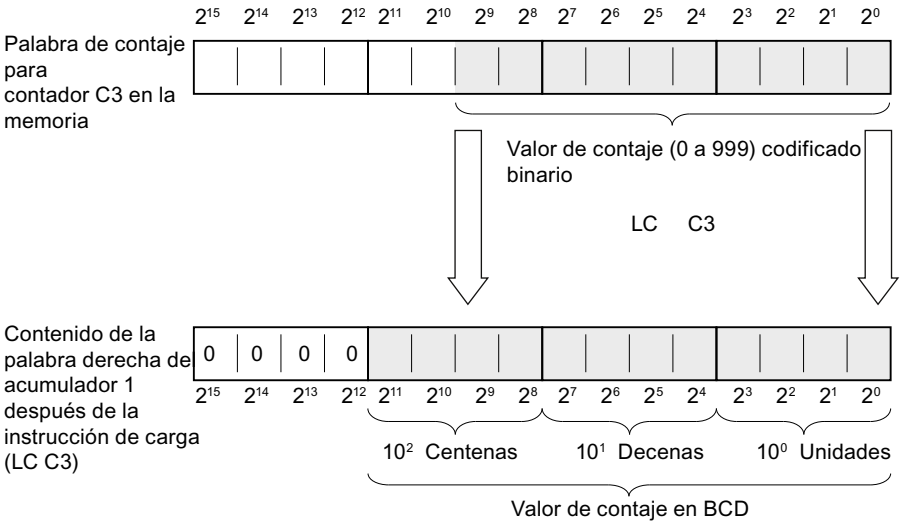
Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

LC: Cargar valor de contaje codificado en BCD

Descripción

La instrucción "Cargar valor de contaje codificado en BCD" transfiere el valor actual de un contador específico a la palabra derecha del acumulador 1 en código BCD. A continuación, el valor de contaje cargado está disponible en formato BCD para su posterior procesamiento.

La figura siguiente muestra el funcionamiento de la instrucción "Cargar valor de contaje codificado en BCD":



Sintaxis

Para la instrucción "Cargar valor de contaje codificado en BCD" se utiliza la siguiente sintaxis:

```
LC <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar valor de contaje codificado en BCD":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor actual se carga.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LC "MyCounter"	// Cargar el valor de contaje actual en el acumulador 1 codificado en BCD.
T "Tag_Output"	// Transferir el valor de contaje actual al operando "Tag_Output" y guardarlo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

R: Poner contador a 0

Descripción

La instrucción "Poner contador a 0" pone a "0" el valor de contaje actual de un contador concreto.

La instrucción solo se ejecuta si el resultado lógico (RLO) previo a la instrucción tiene el estado lógico "1". Mientras el RLO antes de la instrucción "Poner contador a 0" esté a "1", las consultas de si el estado del contador es "1" darán como resultado "0".

Al poner un contador a 0, la marca de flancos interna para las instrucciones "Inicializar contador", "Contador ascendente" y "Contador descendente" no se desactiva.

Sintaxis

Para la instrucción "Poner contador a 0" se utiliza la siguiente sintaxis:

```
R <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Poner contador a 0":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor de contaje se pone a "0".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
R "MyCounter"	// Poner contador a 0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

S: Inicializar contador

Descripción

La instrucción "Inicializar contador" pone un contador concreto al valor que se encuentra en el acumulador 1 en el momento de la consulta.

La instrucción "Inicializar contador" se ejecuta solamente cuando el resultado lógico cambia de "0" a "1" antes de la instrucción. El flanco de señal ascendente también puede obtenerse mediante la instrucción "Habilitar contador", que pone a "0" la marca de flancos interna.

Para inicializar un contador debe haber un valor de contaje válido en el acumulador 1. Se consideran valores de contaje válidos los valores codificados en BCD con tres décadas. Sólo se admiten valores de contaje positivos, la instrucción de contaje no puede procesar valores negativos. Si el acumulador 1 no tiene ningún valor BCD válido, se ejecuta el tratamiento de errores síncronos (STOP u OB122). El bit de estado OV no se ve afectado.

Sintaxis

Para la instrucción "Inicializar contador" se utiliza la siguiente sintaxis:

```
S <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Inicializar contador":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador que se inicializa.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
L "Tag_CountValue"	// Cargar el valor de contaje en el acumulador 1.
S "MyCounter"	// Poner el contador al valor que se encuentra en el acumulador 1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

CU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa en uno el valor de contaje actual cuando el resultado lógico (RLO) antes de la instrucción cambie de "0" a "1". En cada flanco de señal ascendente, el valor de contaje se sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite, el RLO antes de la instrucción ya no influye en el valor de contaje.

Para ejecutar la instrucción "Contador ascendente" se requiere siempre un flanco de señal ascendente. El flanco de señal ascendente también puede obtenerse mediante la instrucción "Habilitar contador", que pone a "0" la marca de flancos interna.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

```
CU <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor de contaje se incrementa en uno con un flanco ascendente.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CU "MyCounter"	// Incrementar el valor de contaje en uno con un flanco ascendente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

CD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa en uno el valor de contaje actual cuando el resultado lógico (RLO) antes de la instrucción cambie de "0" a "1". En cada flanco de señal ascendente, el valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una

vez alcanzado el valor límite, el RLO antes de la instrucción ya no influye en el valor de contaje. No se cuenta con valores negativos.

Para ejecutar la instrucción "Contador descendente" se requiere siempre un flanco de señal ascendente. El flanco de señal ascendente también puede obtenerse mediante la instrucción "Habilitar contador", que pone a "0" la marca de flancos interna.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

```
CD <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Operando	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	COUNTER	C	Contador cuyo valor de contaje se decrementa en uno con un flanco ascendente.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input"	// Consultar si el estado lógico es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CD "MyCounter"	// Decrementar el valor de contaje en uno con un flanco ascendente.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Activar el bit de estado en instrucciones de temporización y contaje (Página 2549)

Principios básicos de la palabra de estado (Página 2545)

Comparación

? I: Comparación de enteros

Descripción

La instrucción "Comparación de enteros" (16 bits) compara el contenido de la palabra derecha del acumulador 2 con el contenido de la palabra derecha del acumulador 1. Los contenidos de los acumuladores 1 y 2 se interpretan como enteros (de 16 bits). La instrucción ejecuta la comparación en función de la función de comparación indicada.

La tabla siguiente muestra las funciones de comparación disponibles y su significado:

Función de comparación	Significado
==I	Comparación de igualdad de enteros
<>I	Comparación de desigualdad de enteros
>I	Comparación de superioridad de enteros
<I	Comparación de inferioridad de enteros
>=I	Comparación de superioridad o igualdad de enteros
<=I	Comparación de inferioridad o igualdad de enteros

El resultado de la comparación influye en el resultado lógico (RLO) así como en los bits de estado CC 1 y CC 0. Si se cumple la condición de la comparación, el RLO se pone al estado lógico "1". Si no se cumple la condición de la comparación, el RLO se pone a "0".

La tabla siguiente muestra cómo las distintas funciones de comparación influyen en el RLO:

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
==I	0	1	0
<>I	1	0	1
>I	1	0	0
<I	0	0	1
>=I	1	1	0
<=I	0	1	1

La activación de los bits de estado CC 1 y CC 0 depende de la relación de los dos valores que intervienen en la comparación.

Los contenidos de los acumuladores 1 y 2 no cambian debido a la ejecución de las funciones de comparación. Tras ejecutar una función de comparación, es posible evaluar el resultado lógico con funciones de asignación y salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el valor del operando "Tag_Input_1" en el acumulador 1.
L "Tag_Input_2"	// Desplazar el contenido del acumulador 1 (Tag_Input_1) al acumulador 2.
	// Cargar el valor del operando "Tag_Input_2" en el acumulador 1.
>I	// Comparar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
= "Tag_Output"	// Asignar el resultado de la comparación al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores		
Tag_Input_1	10	10	5
Tag_Input_2	5	10	10
Tag_Output	1	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de una cadena secuencial (Página 3269)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

? D: Comparación de enteros dobles

Descripción

La instrucción "Comparación de enteros dobles" (32 bits) compara el contenido del acumulador 2 (ACU 2) con el contenido del acumulador 1 (ACU 1). Los contenidos de los acumuladores 1 y 2 se interpretan como enteros dobles (de 32 bits). La instrucción ejecuta la comparación en función de la función de comparación indicada.

La tabla siguiente muestra las funciones de comparación disponibles y su significado:

Función de comparación	Significado
==D	Comparación de igualdad de enteros dobles
<>D	Comparación de desigualdad de enteros dobles
>D	Comparación de superioridad de enteros dobles
<D	Comparación de inferioridad de enteros dobles
>=D	Comparación de superioridad o igualdad de enteros dobles
<=D	Comparación de inferioridad o igualdad de enteros dobles

El resultado de la comparación influye en el resultado lógico (RLO) así como en los bits de estado CC 1 y CC 0. Si se cumple la condición de la comparación, el RLO se pone al estado lógico "1". Si no se cumple la condición de la comparación, el RLO se pone al estado lógico "0".

La tabla siguiente muestra cómo las distintas funciones de comparación influyen en el RLO:

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
==D	0	1	0
<>D	1	0	1
>D	1	0	0
<D	0	0	1
>=D	1	1	0
<=D	0	1	1

La activación de los bits de estado CC 1 y CC 0 depende de la relación de los dos valores que intervienen en la comparación.

Los contenidos de los acumuladores 1 y 2 no cambian debido a la ejecución de las funciones de comparación. Tras ejecutar una función de comparación, es posible evaluar el resultado lógico con funciones de asignación y salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el valor del operando "Tag_Input_1" en el acumulador 1.
L "Tag_Input_2"	// Desplazar el contenido del acumulador 1 (Tag_Input_1) al acumulador 2.
	// Cargar el valor del operando "Tag_Input_2" en el acumulador 1.
>D	// Comparar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
= "Tag_Output"	// Asignar el resultado de la comparación al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores		
Tag_Input_1	150 000	150 000	100 000
Tag_Input_2	100 000	150 000	150 000
Tag_Output	1	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

? R: Comparación de números en coma flotante

Descripción

La instrucción "Comparación de números en coma flotante" compara el contenido del acumulador 2 (ACU 2) con el contenido del acumulador 1 (ACU 1). Los contenidos de los acumuladores 1 y 2 se interpretan como números en coma flotante. Para ejecutar la comparación debe haber números en coma flotante válidos en los acumuladores 1 y 2. La instrucción ejecuta la comparación en función de la función de comparación indicada.

La tabla siguiente muestra las funciones de comparación disponibles y su significado:

Función de comparación	Significado
==R	Comparación de igualdad de números en coma flotante
<>R	Comparación de desigualdad de números en coma flotante
>R	Comparación de superioridad de números en coma flotante
<R	Comparación de inferioridad de números en coma flotante
>=R	Comparación de superioridad o igualdad de números en coma flotante
<=R	Comparación de inferioridad o igualdad de números en coma flotante

El resultado de la comparación influye en el resultado lógico (RLO) así como en los bits de estado CC 1 y CC 0. Si se cumple la condición de la comparación, el RLO se pone al estado lógico "1". Si no se cumple la condición de la comparación, el RLO se pone al estado lógico "0".

La tabla siguiente muestra cómo las distintas funciones de comparación influyen en el RLO:

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
==R	0	1	0
<>R	1	0	1
>R	1	0	0

Función de comparación	RLO si ACU 2 > ACU 1	RLO si ACU 2 = ACU 1	RLO si ACU 2 < ACU 1
<R	0	0	1
>=R	1	1	0
<=R	0	1	1

La activación de los bits de estado CC 1 y CC 0 depende de la relación de los dos valores que intervienen en la comparación.

Nota

Si se comparan números en coma flotante no válidos, la CPU pone el resultado lógico a "0" y los bits de estado CC 0, CC 1, OV y OS a "1".

Los contenidos de los acumuladores 1 y 2 no cambian debido a la ejecución de las funciones de comparación. Tras ejecutar una función de comparación, es posible evaluar el resultado lógico con funciones de asignación y salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`L "Tag_Input_1"`	`// Cargar el valor del operando "Tag_Input_1" en el acumulador 1.`
`L "Tag_Input_2"`	`// Desplazar el contenido del acumulador 1 (Tag_Input_1) al acumulador 2.`
	`// Cargar el valor del operando "Tag_Input_2" en el acumulador 1.`
`>R`	`// Comparar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".`
`= "Tag_Output"`	`// Asignar el resultado de la comparación al operando "Tag_Output".`

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores		
Tag_Input_1	10 515	10 515	5 232
Tag_Input_2	5 232	10 515	10 515
Tag_Output	1	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones de comparación (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

Funciones matemáticas

Enteros

+I: Sumar enteros (16 bits)

Descripción

La instrucción "Sumar enteros (16 bits)" suma los valores que se encuentran en las palabras derechas de los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros de 16 bits.

La instrucción guarda la suma en la palabra derecha del acumulador 1. La palabra izquierda del acumulador 1 no cambia.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la suma es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Sumar enteros (16 bits)" se utiliza la siguiente sintaxis:

```
+I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer sumando.
L "Tag_Value_2"	// Cargar el segundo sumando.
+I	// Sumar valores
T "Tag_Result"	// Transferir la suma al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de cálculo de una ecuación (Página 3266)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

-I: Restar enteros (16 bits)

Descripción

La instrucción "Restar enteros (16 bits)" resta el valor de la palabra derecha del acumulador 1 del valor de la palabra derecha del acumulador 2. La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros (de 16 bits).

La instrucción guarda la diferencia en la palabra derecha del acumulador 1. La palabra izquierda del acumulador 1 no cambia.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la diferencia es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Restar enteros (16 bits)" se utiliza la siguiente sintaxis:

```
-I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer valor de la resta en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el segundo valor de la suma en el acumulador 1.

AWL	Explicación
-I	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Result"	// Transferir la diferencia al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

*I: Multiplicar enteros (16 bits)

Descripción

La instrucción "Multiplicar enteros (16 bits)" multiplica los valores que se encuentran en las palabras derechas de los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros de 16 bits.

La instrucción guarda el producto en el acumulador 1 como entero doble (DINT).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el producto es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplicar enteros (16 bits)" se utiliza la siguiente sintaxis:

```
*I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el multiplicando.
L "Tag_Value_2"	// Cargar el multiplicador.

AWL	Explicación
*I	// Multiplicar valores
T "Tag_Result"	// Transferir el producto al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de cálculo de una ecuación (Página 3266)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

/I: Dividir enteros (16 bits)

Descripción

La instrucción "Dividir enteros (16 bits)" divide el valor de la palabra derecha del acumulador 2 entre el valor de la palabra derecha del acumulador 1. La instrucción interpreta los valores de los acumuladores como enteros (de 16 bits).

La instrucción da dos resultados: el cociente y el resto de la división. El cociente es el resultado entero de la división. La instrucción lo guarda en la palabra derecha del acumulador 1. El resto de la división lo guarda en la palabra izquierda del acumulador 1. Si el dividendo es negativo, el resto de la división también lo será.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el cociente es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero tanto para el cociente como para el resto. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Dividir enteros (16 bits)" se utiliza la siguiente sintaxis:

```
/I
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"              // Desplazar el dividendo al acumulador 2.
                             // Cargar el divisor en el acumulador 1.
/I                           // Dividir el valor del acumulador 2 entre el valor
                             del acumulador 1.
T "Tag_Result"               // Transferir el resultado al operando "Tag_Result".
```

Antes de la ejecución

La tabla siguiente muestra los contenidos de las palabras derechas de los acumuladores 1 y 2 antes de ejecutar la instrucción "Dividir enteros (16 bits)":

Acumulador	Valor
Acumulador 2	13
Acumulador 1	4

Después de la ejecución

La tabla siguiente muestra el contenido del acumulador 1 después de ejecutar la instrucción "Dividir enteros (16 bits)":

Acumulador 1	Valor
Bit 0 a 15 (palabra derecha)	3
Bit 16 a 31 (palabra izquierda)	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de cálculo de una ecuación (Página 3266)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

+D: Sumar enteros dobles (32 bits)

Descripción

La instrucción "Sumar enteros dobles (32 bits)" suma los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros dobles. La instrucción guarda la suma en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la suma es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Sumar enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
+D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer sumando.
L "Tag_Value_2"	// Cargar el segundo sumando.
+D	// Sumar valores
T "Tag_Result"	// Transferir la suma al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

-D: Restar enteros dobles (32 bits)

Descripción

La instrucción "Restar enteros dobles (32 bits)" resta el valor del acumulador 1 del valor del acumulador 2. La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros dobles (de 32 bits). La instrucción guarda el resultado de la resta en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la diferencia es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Restar enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
-D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer valor de la resta en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el segundo valor de la resta en el acumulador 1.
-D	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Result"	// Transferir la diferencia al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

*D: Multiplicar enteros dobles (32 bits)

Descripción

La instrucción "Multiplicar enteros dobles (32 bits)" multiplica los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como enteros dobles. La instrucción guarda el producto en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el producto es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplicar enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
*D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el multiplicando.
L "Tag_Value_2"	// Cargar el multiplicador.
*D	// Multiplicar valores
T "Tag_Result"	// Transferir el producto al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

/D: Dividir enteros dobles (32 bits)

Descripción

La instrucción "Dividir enteros dobles (32 bits)" divide el valor del acumulador 2 entre el valor del acumulador 1. La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros dobles (de 32 bits). La instrucción guarda el resultado entero de la división (cociente) en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el cociente es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero para el cociente. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Dividir enteros dobles (32 bits)" se utiliza la siguiente sintaxis:

```
/D
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"	// Desplazar el dividendo al acumulador 2.
	// Cargar el divisor en el acumulador 1.
/D	// Dividir el valor del acumulador 2 entre el valor del acumulador 1.
T "Tag_Result"	// Transferir el cociente al operando "Tag_Result".

Antes de la ejecución

La tabla siguiente muestra los contenidos de los acumuladores 1 y 2 antes de ejecutar la instrucción "Dividir enteros dobles (32 bits)":

Acumulador	Valor
Acumulador 2	13
Acumulador 1	4

Después de la ejecución

La tabla siguiente muestra el contenido del acumulador 1 después de ejecutar la instrucción "Dividir enteros dobles (32 bits)":

Acumulador	Valor
Acumulador 1	3

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

+: Sumar constantes

Descripción

La instrucción "Sumar constantes" suma el valor de una constante concreta al contenido del acumulador 1. El valor de la constante puede indicarse como entero o entero doble. La instrucción guarda el resultado en el acumulador 1. Si se suman enteros, la instrucción influye sólo en la palabra derecha del acumulador 1.

Para ejecutar la suma de un entero como operación DINT, antes de la constante hay que introducir "L#". La instrucción "Sumar constantes" utiliza números negativos.

La instrucción no influye en los bits de estado.

Sintaxis

Para la instrucción "Sumar constantes" se utiliza la siguiente sintaxis:

```
+ <constante>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar constantes":

Parámetro	Tipo de datos	Descripción
<constante>	INT, DINT	Constante que se suma.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
+ 25	// Aumentar en 25 el valor del acumulador 1.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
+ -100	// Reducir en 100 el valor del acumulador 1.
<I	// Consultar si el valor del acumulador 2 es menor que el valor del acumulador 1.
JC NEXT	// Si el resultado de la consulta es "1", saltar a la etiqueta NEXT .
	// Si el resultado de la consulta es "0", procesar la instrucción siguiente.
T "Tag_Result_2"	Transferir el contenido del acumulador 1 al operando "Tag_Result_2".
L "Tag_Value_4"	// Cargar el valor del operando "Tag_Value_4".
+ L#20	// Incrementar en 20 el contenido del acumulador 1.
	// La suma se efectúa conforme a un cálculo DINT
T "Tag_Result_3"	// Transferir el contenido del acumulador 1 al operando "Tag_Result_3".
NEXT: L "Tag_Value_5"	// Etiqueta "NEXT"
	// Cargar el valor del operando "Tag_Value_5".
+D	// Sumar los valores de los acumuladores 1 y 2.
T "Tag_Result_4"	// Transferir la suma al operando "Tag_Result_4".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

INC: Incrementar

Descripción

La instrucción "Incrementar" aumenta el valor que se encuentra en el acumulador 1 en el valor que se indica como parámetro de la instrucción. El valor indicado puede estar comprendido en un rango de valores de 0 a 255. El incremento sólo se realiza en el byte derecho del acumulador 1. No se lleva a cabo ningún traslado a los bytes que están a la izquierda.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores 2, 3 y 4 no cambian.

Si el valor del parámetro está fuera del rango de valores, el resultado quedará falsificado o bien se generará un mensaje de error.

Sintaxis

Para la instrucción "Incrementar" se utiliza la siguiente sintaxis:

```
INC <constante>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Incrementar":

Parámetro	Formato	Descripción
<constante>	Entero de 8 bits	Valor que se aumenta.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
INC 5	// Aumentar en 5 el valor del acumulador 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

DEC: Decrementar

Descripción

La instrucción "Decrementar" reduce el valor que se encuentra en el acumulador 1 en el valor que se indica como parámetro de la instrucción. El valor indicado puede estar comprendido en un rango de valores de 0 a 255. El decremento sólo se realiza en el byte derecho del acumulador 1. No se lleva a cabo ningún traslado a los bytes que están a la izquierda.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores 2, 3 y 4 no cambian.

Si el valor del parámetro está fuera del rango de valores, el resultado quedará falsificado o bien se generará un mensaje de error.

Sintaxis

Para la instrucción "Decrementar" se utiliza la siguiente sintaxis:

```
DEC <constante>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Decrementar":

Parámetro	Formato	Descripción
<constante>	Entero de 8 bits	Valor que se decrementa.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
DEC 5	// Decrementar en 5 el valor del acumulador 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

MOD: Obtener resto de división

Descripción

La instrucción "Obtener resto de división" divide el valor del acumulador 2 entre el valor del acumulador 1 y guardar el resto de la división.

La instrucción interpreta los valores de los acumuladores 1 y 2 como enteros dobles. La instrucción guarda el resto de la división en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resto de la división es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero para el resultado. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Obtener resto de división" se utiliza la siguiente sintaxis:

```
MOD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"	// Desplazar el dividendo al acumulador 2.
	// Cargar el divisor en el acumulador 1.
MOD	// Dividir el valor del acumulador 2 entre el valor del acumulador 1.

AWL	Explicación
`T "Tag_Result"`	`// Transferir el resto de la división al operando "Tag_Result".`

Antes de la ejecución

La tabla siguiente muestra los contenidos de los acumuladores 1 y 2 antes de ejecutar la instrucción "Obtener resto de división":

Acumulador	Valor
Acumulador 2	13
Acumulador 1	4

Después de la ejecución

La tabla siguiente muestra el contenido del acumulador 1 después de ejecutar la instrucción "Obtener resto de división":

Acumulador	Valor
Acumulador 1	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Números en coma flotante

+R: Sumar números en coma flotante

Descripción

La instrucción "Sumar números en coma flotante" suma los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como números en coma flotante. La instrucción guarda la suma en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la suma es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Sumar números en coma flotante" se utiliza la siguiente sintaxis:

```
+R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer sumando.
L "Tag_Value_2"	// Cargar el segundo sumando.
+R	// Sumar valores
T "Tag_Result"	// Transferir la suma al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

-R: Restar números en coma flotante

Descripción

La instrucción "Restar números en coma flotante" resta el valor del acumulador 1 del valor del acumulador 2. La instrucción interpreta los valores de los acumuladores 1 y 2 como números en coma flotante. La instrucción guarda la diferencia en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si la diferencia es negativa, cero o positiva. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Restar números en coma flotante" se utiliza la siguiente sintaxis:

```
-R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el primer valor de la resta en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el segundo valor de la resta en el acumulador 1.
-R	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Result"	// Transferir la diferencia al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

*R: Multiplicar números en coma flotante

Descripción

La instrucción "Multiplicar números en coma flotante" multiplica los valores que se encuentran en los acumuladores 1 y 2. La instrucción interpreta dichos valores como números en coma flotante. La instrucción guarda el producto en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el producto es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Multiplicar números en coma flotante" se utiliza la siguiente sintaxis:

```
*R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el multiplicando.
L "Tag_Value_2"	// Cargar el multiplicador.
*R	// Multiplicar valores
T "Tag_Result"	// Transferir el producto al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

/R: Dividir números en coma flotante

Descripción

La instrucción "Dividir números en coma flotante" divide el valor del acumulador 2 entre el valor del acumulador 1. La instrucción interpreta los valores de los acumuladores 1 y 2 como números en coma flotante. La instrucción guarda el resultado entero de la división (cociente) en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el cociente es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

En caso de dividir entre cero, la instrucción da cero para el cociente. En este caso, los bits de estado CC 0, CC 1, OV y OS se ponen al estado lógico "1".

Si se introduce un número en coma flotante no válido o el carácter de infinito (+/- ∞), la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

En los controladores con dos acumuladores, el contenido del acumulador 2 no cambia después de ejecutar la instrucción.

En los controladores con cuatro acumuladores, el contenido del acumulador 3 se copia en el acumulador 2 y el contenido del acumulador 4 se copia en el acumulador 3. El contenido del acumulador 4 no cambia después de ejecutar la instrucción.

Sintaxis

Para la instrucción "Dividir números en coma flotante" se utiliza la siguiente sintaxis:

```
/R
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el dividendo en el acumulador 1.
L "Tag_Value_2"	// Desplazar el dividendo al acumulador 2.
	// Cargar el divisor en el acumulador 1.
/R	// Dividir el valor del acumulador 2 entre el valor del acumulador 1.
T "Tag_Result"	// Transferir el cociente al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" calcula el valor absoluto del valor que se encuentra en el acumulador 1. La instrucción interpreta el contenido del acumulador 1 como número en coma flotante y pone el signo de la mantisa a "0". El resultado se guarda en el acumulador 1.

La instrucción "Calcular valor absoluto" no influye en los bits de estado.

Sintaxis

Para la instrucción "Calcular valor absoluto" se utiliza la siguiente sintaxis:

```
ABS
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el número en coma flotante en el acumulador 1.
ABS	// Calcular valor absoluto
T "Tag_Result"	// Transferir el resultado al operando "Tag_Result".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	-1.5E+02
Tag_Result	1.5E+02

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Números en coma flotante, avanzado

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" calcula el cuadrado de un número en coma flotante que se encuentra en el acumulador 1. El resultado se guarda en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular cuadrado" se utiliza la siguiente sintaxis:

```
SQR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
SQR	// Calcular el cuadrado
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" calcula la raíz cuadrada del número en coma flotante que se encuentra en el acumulador 1. El resultado se guarda en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si en el acumulador 1 se introduce un valor inferior a cero o un número en coma flotante no válido, la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1".

Si en el acumulador 1 se encuentra el valor "0", la instrucción devuelve el valor "0".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular raíz cuadrada" se utiliza la siguiente sintaxis:

```
SQRT
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
SQRT	// Calcular la raíz cuadrada
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" calcula la potencia en base e (e = 2.718282) del número en coma flotante que se encuentra en el acumulador 1. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor exponencial" se utiliza la siguiente sintaxis:

```
EXP
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
EXP	// Calcular valor exponencial
	// Guardar el resultado en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" calcula el logaritmo natural en base e (e = 2.718282) a partir del número en coma flotante que se encuentra en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si en el acumulador 1 hay un valor inferior o igual a cero, la instrucción escribe un valor no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular logaritmo natural" se utiliza la siguiente sintaxis:

```
LN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
LN	// Calcular logaritmo natural
	// Guardar el resultado en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" calcula el seno del ángulo que se encuentra en el acumulador 1. El ángulo debe indicarse en radianes y encontrarse en el acumulador 1 en forma de número en coma flotante. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor de seno" se utiliza la siguiente sintaxis:

```
SIN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
SIN	// Calcular valor de seno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" calcula, a partir del valor que se encuentra en el acumulador 1, el tamaño del ángulo que corresponde a dicho valor. El valor del acumulador 1 debe encontrarse en forma de número en coma flotante en un rango de valores de "-1" a "1".

El resultado de la instrucción se devuelve en radianes y se guarda en el acumulador 1. Según sea el valor del acumulador 1, el resultado puede encontrarse en un rango de valores de $-\pi/2$ a $+\pi/2$ (π= 3,14159).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si el valor que está en el acumulador 1 rebasa por exceso el rango admisible, la instrucción escribe un número en coma flotante no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor de arcoseno" se utiliza la siguiente sintaxis:

```
ASIN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
ASIN	// Calcular valor de arcoseno // Guardar el resultado en el acumulador 1.

AWL	Explicación
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" calcula el coseno del ángulo que se encuentra en el acumulador 1. El tamaño del ángulo debe indicarse en radianes y encontrarse en el acumulador 1 en forma de número en coma flotante. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor de coseno" se utiliza la siguiente sintaxis:

```
COS
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
COS	// Calcular valor de coseno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" calcula, a partir del valor que se encuentra en el acumulador 1, el tamaño del ángulo que corresponde a dicho valor. El valor del acumulador 1 debe encontrarse en forma de número en coma flotante en un rango de valores de "-1" a "1".

El resultado de la instrucción se devuelve en radianes y se guarda en el acumulador 1. Según sea el valor del acumulador 1, el resultado puede encontrarse en un rango de valores de "0" a "π" (π= 3,14159).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si el valor que está en el acumulador 1 rebasa por exceso el rango admisible, la instrucción escribe un número en coma flotante no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor de arcocoseno" se utiliza la siguiente sintaxis:

```
ACOS
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
ACOS	// Calcular valor de arcocoseno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" calcula la tangente del ángulo que se encuentra en el acumulador 1. El tamaño del ángulo debe indicarse en radianes y encontrarse en el acumulador 1 en forma de número en coma flotante. La instrucción guarda el resultado en el acumulador 1.

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor de tangente" se utiliza la siguiente sintaxis:

```
TAN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
TAN	// Calcular valor de tangente
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" calcula, a partir del valor que se encuentra en el acumulador 1, el tamaño del ángulo que corresponde a dicho valor. El valor del acumulador 1 debe encontrarse en forma de número en coma flotante.

El resultado de la instrucción se devuelve en radianes y se guarda en el acumulador 1. Según sea el valor del acumulador 1, el resultado puede encontrarse en un rango de valores de $-\pi/2$ a $+\pi/2$ (π= 3,14159).

Tras ejecutar la instrucción, los bits de estado CC 0 y CC 1 indican si el resultado es negativo, cero o positivo. Si el resultado está fuera del rango numérico admisible, los bits de estado OV y OS se ponen a "1".

Si el valor que está en el acumulador 1 rebasa por exceso el rango admisible, la instrucción escribe un número en coma flotante no válido en el acumulador 1 y pone los bits de estado CC 0, CC 1, OV y OS a "1". Si los números en coma flotante no son válidos, la instrucción pone los bits de estado CC 0, CC 1, OV y OS también a "1".

La instrucción solo influye en el contenido del acumulador 1. Los contenidos de los acumuladores restantes no cambian.

Sintaxis

Para la instrucción "Calcular valor de arcotangente" se utiliza la siguiente sintaxis:

```
ATAN
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
ATAN	// Calcular valor de arcotangente
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

Carga y transferencia

Carga

L: Cargar

Descripción

La instrucción "Cargar" carga el contenido de un operando específico en el acumulador 1. El operando que debe cargarse puede estar direccionado en una de las áreas de memoria siguientes en forma de byte, palabra o palabra doble:

- Memoria imagen de proceso de las entradas y salidas (I, Q)
- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)
- Punteros
- Periferia (P)
- Temporizador (T)
- Contador (C)

El área de memoria del acumulador 1 está organizada por bytes y tiene 32 bits de ancho.

La instrucción escribe el contenido de un operando que cargar en formato de byte alineado a la derecha en el acumulador 1. Los bytes restantes del acumulador 1 se rellenan con "0".

La instrucción escribe el contenido de un operando que cargar en formato de palabra en la palabra derecha del acumulador 1. El byte con un direccionamiento más alto se transfiere al byte derecho (bits 0 a 7) del acumulador 1. El byte con un direccionamiento más bajo se escribe a su izquierda. Los bytes restantes de la palabra izquierda del acumulador 1 se rellenan con "0".

La instrucción escribe el contenido de un operando que cargar en formato de palabra doble en los 32 bits del acumulador 1. El byte con un direccionamiento más alto se transfiere al byte derecho (bits 0 a 7) del acumulador 1. El byte con el direccionamiento más bajo se escribe en los bits 24 y 31.

La tabla siguiente muestra a modo de ejemplo cómo el contenido del acumulador 1 se transforma en formato de byte, palabra y palabra doble al cargar operandos:

Instrucción	Acumulador 1							
	31 . . .	. . . 24	23 . . .	. . . 16	15 . . .	. . . 8	7 . . .	. . . 0
Cargar un byte: L MB10	0000	0000	0000	0000	0000	0000	<MB10>	
Cargar una palabra: L MW10	0000	0000	0000	0000	<MB10>		<MB11>	
Cargar una palabra doble: L MD10	<MB10>		<MB11>		<MB12>		<MB13>	

Al cargar el operando indicado, el contenido previo del acumulador 1 se desplaza al acumulador 2. La instrucción "Cargar" transfiere todo el contenido del acumulador 1 al acumulador 2. El contenido previo del acumulador 2 se pierde.

En los controladores con cuatro acumuladores, los contenidos de los acumuladores 3 y 4 no se ven influidos por esta instrucción.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Cargar" se utiliza la siguiente sintaxis:

```
L <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Input	Secuencias de bits, enteros, temporizadores, REAL, DATE, TOD, CHAR.	I, Q, PI, M, L, DB, DI, T, C, puntero, parámetro	Operando cuyo contenido se carga.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.

AWL	Explicación
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de registro del nivel de llenado de una zona de almacén (Página 3263)

Ejemplo de cálculo de una ecuación (Página 3266)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Ejemplo de una cadena secuencial (Página 3269)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

L STW: Cargar palabra de estado en acumulador 1

Descripción

La instrucción "Cargar palabra de estado en acumulador 1" carga la palabra de estado en el acumulador 1. Durante la ejecución, la instrucción escribe los diferentes bits de estado en la palabra derecha del acumulador 1. Los bits restantes del acumulador 1 se rellenan con "0".

La tabla siguiente muestra el contenido del acumulador después de ejecutar la instrucción "Cargar palabra de estado en acumulador 1":

	Acumulador 1									
Número de bit	31-9	8	7	6	5	4	3	2	1	0
Contenido	0	BR	CC 1	CC 0	OV	OS	OR	STA	RLO	/FC

Esta instrucción se ejecuta independientemente del estado lógico del bit de estado.

Nota

Los controladores de la serie S7-300 (excepto el S7-318) no cargan los bits de estado /FC, STA y OR en el acumulador 1. Estas posiciones se rellenan con "0".

Sintaxis

Para la instrucción "Cargar palabra de estado en acumulador 1" se utiliza la siguiente sintaxis:

```
L STW
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L STW	// Cargar palabra de estado en acumulador 1.
T "Tag_STW"	// Transferir el contenido del acumulador 1 (bits de estado) al operando "Tag_STW".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

LAR1: Cargar contenido del acumulador 1 en AR1

Descripción

La instrucción "Cargar contenido del acumulador 1 en AR1 " carga el contenido del acumulador 1 en el registro de direcciones 1 (AR1). El valor del acumulador 1 debe equivaler al formato de un puntero de área (POINTER).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar contenido del acumulador 1 en AR1" se utiliza la siguiente sintaxis:

```
LAR1
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L P#10.0	// Cargar el puntero (P#10.0) en el acumulador 1.
LAR1	// Cargar el contenido del acumulador 1 en el registro de direcciones 1.
L MW [AR1,P#4.0]	// Cargar MW14 en acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR1,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

LAR1 <D>: Cargar palabra doble o puntero de área en AR1

Descripción

La instrucción "Cargar palabra doble o puntero de área en AR1" carga el contenido de una palabra doble o de un puntero intraárea o interárea en el registro de direcciones 1 (AR1). El contenido de la palabra doble debe equivaler al formato de un puntero de área.

El puntero y la palabra doble pueden estar direccionados en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar palabra doble o puntero de área en AR1" se utiliza la siguiente sintaxis:

```
LAR1 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar palabra doble o puntero de área en AR1":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Input	DWORD, POINTER	D, M, L	Operando cuyo contenido se carga.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#10.0	// Cargar el contenido del puntero intraárea P#10.0 en el registro de direcciones 1.
L MW [AR1,P#4.0]	// Cargar MW14 en el acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR1,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".
LAR1 MD20	// Cargar el contenido de MD20 (MD20 = P#30.0) en el registro de direcciones 1.

AWL	Explicación
A I [AR1,P#2.1]	// Consultar si el estado lógico del bit I32.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

LAR1 AR2: Cargar contenido del AR2 en AR1

Descripción

La instrucción "Cargar contenido del AR2 en AR1" carga el contenido del registro de direcciones 2 (AR2) en el registro de direcciones 1 (AR1).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar contenido del AR2 en AR1" se utiliza la siguiente sintaxis:

```
LAR1 AR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR2 P#20.0	// Cargar el contenido del puntero intraárea P#20.0 en el registro de direcciones 2.
LAR1 AR2	// Copiar el contenido del registro de direcciones 2 en el registro de direcciones 1.
L MW [AR1,P#4.0]	// Cargar MW24 en el acumulador 1.
T "Tag_2"	// Transferir el acumulador 1 a "Tag_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

LAR2: Cargar contenido del acumulador 1 en AR2

Descripción

La instrucción "Cargar contenido del acumulador 1 en AR2 " carga el contenido del acumulador 1 en el registro de direcciones 2 (AR2). El valor del acumulador 1 debe equivaler al formato de un puntero de área (POINTER).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar contenido del acumulador 1 en AR2" se utiliza la siguiente sintaxis:

```
LAR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L P#10.0	// Cargar el puntero (P#10.0) en el acumulador 1.
LAR2	// Cargar el contenido del acumulador 1 en el registro de direcciones 2.
L MW [AR2,P#4.0]	// Cargar MW14 en el acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR2,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

LAR2 <D>: Cargar palabra doble o puntero de área en AR2

Descripción

La instrucción "Cargar palabra doble o puntero de área en AR2" carga el contenido de una palabra doble o de un puntero intraárea o interárea en el registro de direcciones 2 (AR2). El contenido de la palabra doble debe equivaler al formato de un puntero de área (POINTER).

El puntero y la palabra doble pueden estar direccionados en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Cargar palabra doble o puntero de área en AR2" se utiliza la siguiente sintaxis:

```
LAR2 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cargar palabra doble o puntero de área en AR2":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Input	DWORD, POINTER	D, M, L	Operando cuyo contenido se carga.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR2 P#10.0	// Cargar el contenido del puntero intraárea P#10.0 en el registro de direcciones 2.
L MW [AR2,P#4.0]	// Cargar MW14 en el acumulador 1.
L "Tag_Value"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el contenido del operando "Tag_Value" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A I [AR2,P#2.1]	// Consultar si el estado lógico del bit I12.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output"	// Si la condición se cumple (RLO = "1"), poner el operando "Tag_Output" a "1".
LAR2 MD20	// Cargar el contenido de MD20 (MD20 = P#30.0) en el registro de direcciones 2.
A I [AR2,P#2.1]	// Consultar si el estado lógico del bit I32.1 es "1" y combinar el resultado lógicamente con Y con el RLO actual.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Transferir

T: Transferir

Descripción

La instrucción "Transferir" transfiere el contenido del acumulador 1 a un operando concreto por bytes, palabras o palabras dobles. El contenido del acumulador 1 no cambia. El operando indicado debe estar direccionado en una de las áreas de memoria siguientes:

- Memoria imagen de proceso de las entradas y salidas (I, Q)
- Marcas (M)
- Datos locales temporales (L)

- Bloques de datos (DB, DI)
- Periferia (PQ)

El número de bytes que se transferirán depende del formato del operando indicado. Si el operando está en formato de byte, la instrucción transfiere el contenido de los bits 0 a 7 del acumulador 1.

Si el operando indicado está en formato de palabra, la instrucción transfiere el contenido de la palabra que está alineada a la derecha en el acumulador 1. El byte derecho del acumulador 1 se transfiere al byte con un direccionamiento más alto en el operando. Al byte con un direccionamiento más bajo del operando se transfieren los bits de 8 a 15 del acumulador 1.

Si el operando indicado está en formato de palabra doble, la instrucción transfiere todo el contenido del acumulador 1. El byte derecho del acumulador 1 se transfiere al byte con un direccionamiento más alto en el operando. Al byte con un direccionamiento más bajo del operando se transfieren los bits 24 a 31 del acumulador 1.

La tabla siguiente muestra un ejemplo de cómo se transfiere el contenido del acumulador 1 por bytes, palabras o palabras dobles:

Instrucción	Acumulador 1			
Bits	31 24	23 16	15 8	7 0
Contenido del acumulador 1	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)
Transferir un byte				Byte (n+3)
Transferir una palabra			Byte (n+2)	Byte (n+3)
Transferir una palabra doble	Byte (n)	Byte (n+1)	Byte (n+2)	Byte (n+3)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

La ejecución de la instrucción "Transferir" se ve influida por el Master Control Relay (MCR). Si el bit MCR se pone a "0", se transfiere el valor "0" al operando indicado.

Sintaxis

Para la instrucción "Transferir" se utiliza la siguiente sintaxis:

```
T <operando>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Transferir":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<operando>	Output	Secuencias de bits, enteros, temporizadores, REAL, DATE, TOD, CHAR.	I, Q, PQ, M, L, DB, DI	Operando al que se transfiere el contenido del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result_1"	// Transferir el producto (contenido del acumulador 1) al operando "Tag_Result_1".
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
T "Tag_Result_2"	// Transferir la suma (contenido del acumulador 1) al operando "Tag_Result_2".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de cálculo de una ecuación (Página 3266)

Ejemplo de una cadena secuencial (Página 3269)

El MCR (Master Control Relay) (Página 2564)

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

T STW: Transferir acumulador 1 a palabra de estado

Descripción

La instrucción "Transferir acumulador 1 a palabra de estado" transfiere los bits 0 a 8 del acumulador 1 a la palabra de estado.

La tabla siguiente muestra los bits del acumulador 1 con los que se sobrescriben los bits de estado después de ejecutar la instrucción:

	Acumulador 1								
Número de bit	8	7	6	5	4	3	2	1	0
Bits de estado	BR	CC 1	CC 0	OV	OS	OR	STA	RLO	/FC

Esta instrucción se ejecuta independientemente del estado lógico del bit de estado.

Nota

Los bits de la palabra de estado /FC, STA y OR no se cargan con la instrucción "Transferir acumulador 1 a palabra de estado". Únicamente los bits 1, 4, 5, 6, 7 y 8 se cargan en las posiciones de bit respectivas de la palabra significativamente baja del acumulador 1.

Sintaxis

Para la instrucción "Transferir acumulador 1 a palabra de estado" se utiliza la siguiente sintaxis:

```
T STW
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`L "Tag_STW"`	`// Cargar el valor del operando "Tag_STW" en el acumulador 1.`
`T STW`	`// Transferir los bits 0 a 8 del acumulador 1 a la palabra de estado.`

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

CAR: Intercambiar AR1 y AR2

Descripción

La instrucción "Intercambiar AR1 y AR2" intercambia los contenidos de los registros de direcciones 1 (AR1) y 2 (AR2).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Intercambiar AR1 y AR2" se utiliza la siguiente sintaxis:

```
CAR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#10.0	// Cargar el contenido del puntero intraárea P#10.0 en el registro de direcciones 1.
LAR2 P#20.0	// Cargar el contenido del puntero intraárea P#20.0 en el registro de direcciones 2.
CAR	// Intercambiar el contenido de los registros de direcciones.
L MD [AR1,P#2.0]	// Cargar MD22 en el acumulador 1.
T MD [AR2,P#2.0]"	// Transferir MD12 al acumulador 2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

TAR1: Transferir AR1 a acumulador 1

Descripción

La instrucción "Transferir AR1 a acumulador 1" transfiere el contenido del registro de direcciones 1 (AR1) al acumulador 1. El contenido del acumulador 1 se desplaza al acumulador 2.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Transferir AR1 a acumulador 1" se utiliza la siguiente sintaxis:

```
TAR1
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#15.0	// Cargar el contenido del puntero intraárea P#15.0 en el registro de direcciones 1.
TAR1	// Transferir el contenido del registro de direcciones 1 (P#15.0) al acumulador 1.
T "Tag_Pointer"	// Transferir el contenido del acumulador 1 al operando "Tag_Pointer".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

TAR1 <D>: Transferir AR1 a palabra doble

Descripción

La instrucción "Transferir AR1 a palabra doble" transfiere el contenido del registro de direcciones 1 (AR1) a una palabra doble. La palabra doble debe estar direccionada en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Transferir AR1 a palabra doble" se utiliza la siguiente sintaxis:

```
TAR1 <D>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Transferir AR1 a palabra doble":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Output	DWORD	D, M, L	Operando al que se transfiere el contenido del registro de direcciones 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
TAR1 %DBD20	// Transferir el contenido del registro de direcciones 1 a la doble palabra de datos DBD20.
TAR1 %DID30	// Transferir el contenido del registro de direcciones 1 a la doble palabra de instancia DID30.
TAR1 %LD18	// Transferir el contenido del registro de direcciones 1 a la doble palabra de datos locales LD18.
TAR1 %MD24	// Transferir el contenido del registro de direcciones 1 a la doble palabra de marcas MD24.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

TAR1 AR2: Transferir AR1 a AR2

Descripción

La instrucción "Transferir AR1 a AR2" copia el contenido del registro de direcciones 1 (AR1) en el registro de direcciones 2 (AR2).

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Transferir AR1 a AR2" se utiliza la siguiente sintaxis:

```
TAR1 AR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR1 P#10.4	// Cargar el contenido del puntero intraárea P#10.4 en el registro de direcciones 1.
TAR1 AR2	// Transferir el contenido del registro de direcciones 1 al registro de direcciones 2.
A I [AR2,P#3.2]	// Consultar si el estado lógico del bit I13.6 es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag1"	// Poner el operando "Tag_1" a "1".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

TAR2: Transferir AR2 a acumulador 1

Descripción

La instrucción "Transferir AR2 a acumulador 1" transfiere el contenido del registro de direcciones 2 (AR2) al acumulador 1. El contenido del acumulador 1 se desplaza al acumulador 2.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Transferir AR2 a acumulador 1" se utiliza la siguiente sintaxis:

```
TAR2
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
LAR2 P#20.0	// Cargar el contenido del puntero intraárea al registro de direcciones 2 P#20.0.
TAR2	// Transferir el contenido del registro de direcciones 2 (P#20.0) al acumulador 1.
T "Tag_Pointer"	// Transferir el contenido del acumulador 1 al operando "Tag_Pointer".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

TAR2 <D>: Transferir AR2 a palabra doble

Descripción

La instrucción "Transferir AR2 a palabra doble" transfiere el contenido del registro de direcciones 2 (AR2) a una palabra doble. La palabra doble debe estar direccionada en una de las áreas de memoria siguientes:

- Marcas (M)
- Datos locales temporales (L)
- Bloques de datos (DB, DI)

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Para la instrucción "Transferir AR2 a palabra doble" se utiliza la siguiente sintaxis:

```
TAR2 <D>
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Transferir AR2 a palabra doble":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<D>	Output	DWORD	D, M, L	Operando al que se transfiere el contenido del registro de direcciones 2.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
TAR2 %DBD20	// Transferir el contenido del registro de direcciones 2 a la doble palabra de datos DBD20.
TAR2 %DID30	// Transferir el contenido del registro de direcciones 2 a la doble palabra de instancia DID30.
TAR2 %LD18	// Transferir el contenido del registro de direcciones 2 a la doble palabra de datos locales LD18.
TAR2 %MD24	// Transferir el contenido del registro de direcciones 2 a la doble palabra de marcas MD24.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Conversión

BTI: Convertir BCD en entero (16 bits)

Descripción

La instrucción "Convertir BCD en entero (16 bits)" convierte el valor que está en la palabra derecha del acumulador 1 en un entero (de 16 bits). La instrucción interpreta el valor que debe convertirse como número decimal codificado en binario de tres dígitos (BCD).

Los bits 1 a 11 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-999" a "+999".

El signo del valor del resultado se lee del bit 15 del acumulador 1. Si el estado lógico del bit es "0", el signo es positivo. Si el estado lógico del bit es "1", el signo es negativo. Los bits 12 a 14 no se utilizan en la conversión.

Si el código BCD contiene una pseudotétrada, la CPU notifica un error de programación y llama el bloque de organización "OB121". Si el bloque de organización OB121 no existe, la CPU pasa al estado "STOP".

El resultado de la conversión se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

Sintaxis

Para la instrucción "Convertir BCD en entero (16 bits)" se utiliza la siguiente sintaxis:

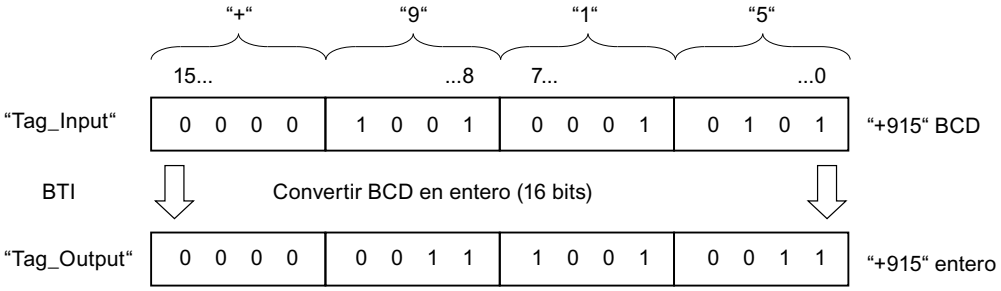
```
BTI
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar BCD en el acumulador 1.
BTI	// Convertir BCD en entero de 16 bits.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

ITB: Convertir entero en BCD (16 bits)

Descripción

La instrucción "Convertir entero en BCD (16 bits)" convierte el valor que se encuentra en la palabra derecha del acumulador 1 en un número decimal en código binario de tres dígitos (BCD). La instrucción interpreta el valor que debe convertirse como un entero de 16 bits.

Los bits 0 a 11 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-999" a "+999". Si el valor que debe convertirse está fuera de este rango, los bits OV y OS se ponen al estado lógico "1". En este caso no se lleva a cabo la conversión.

El signo del valor del resultado se lee de los bits 12 a 15 del acumulador 1. Si el estado lógico de los bits es "0", el signo es positivo. Si el estado lógico de los cuatro bits es "1", el signo es negativo.

El resultado de la conversión se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

Sintaxis

Para la instrucción "Convertir entero en BCD (16 bits)" se utiliza la siguiente sintaxis:

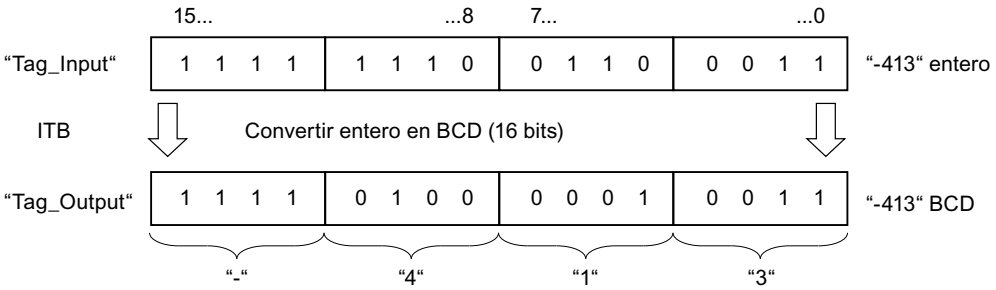
```
ITB
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero (16 bits) en el acumulador 1.
ITB	// Convertir entero en BCD.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Principios básicos de la palabra de estado (Página 2545)

BTD: Convertir BCD en entero doble (32 bits)

Descripción

La instrucción "Convertir BCD en entero doble (32 bits)" convierte el valor que está en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como número decimal codificado en binario de siete dígitos (BCD).

Los bits 0 a 27 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-9999999" a "+9999999".

El signo del valor del resultado se lee del bit 31 del acumulador 1. Si el estado lógico del bit es "0", el signo es positivo. Si el estado lógico del bit es "1", el signo es negativo. Los bits 28 a 30 no se utilizan en la conversión.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Si el código BCD contiene una pseudotétrada, la CPU notifica un error de programación y llama el bloque de organización "OB121". Si el bloque de organización OB121 no existe, la CPU pasa al estado "STOP".

Sintaxis

Para la instrucción "Convertir BCD en entero doble (32 bits)" se utiliza la siguiente sintaxis:

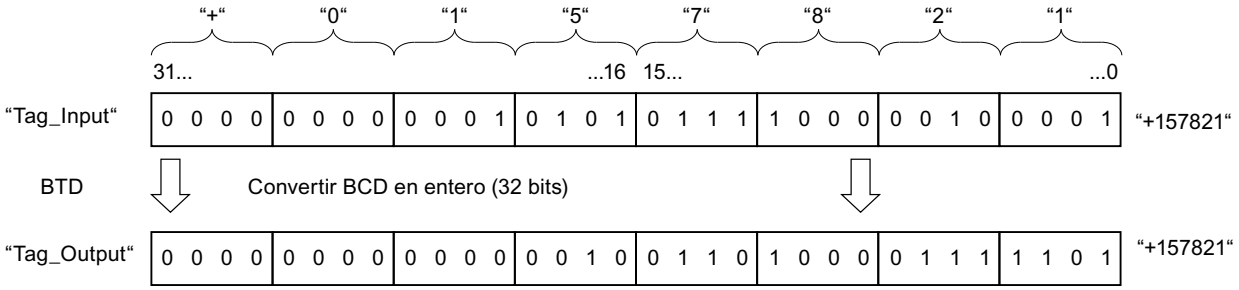
```
BTD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar BCD en el acumulador 1.
BTD	// Convertir BCD en un entero doble.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

ITD: Convertir entero (16 bits) en entero doble (32 bits)

Descripción

La instrucción "Convertir entero (16 bits) en entero doble (32 bits)" convierte el valor que se encuentra en la palabra derecha del acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como un entero de 16 bits.

Los bits 0 a 15 se aplican sin cambios en la conversión. Los bits 16 a 31 del valor del resultado se rellenan con el estado lógico del bit 15.

El valor del resultado se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Convertir entero (16 bits) en entero doble (32 bits)" se utiliza la siguiente sintaxis:

```
ITD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`L "Tag_Input"`	`// Cargar entero (16 bits) en el acumulador 1.`
`ITD`	`// Convertir entero de 16 bits en entero doble de 32 bits.`
`T "Tag_Output"`	`// Transferir el resultado al operando "Tag_Output".`

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Acumulador 1							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Antes de la ejecución	XXXX	XXXX	XXXX	XXXX	1111	1111	1111	0110
Después de la ejecución	1111	1111	1111	1111	1111	1111	1111	0110
	X : el estado lógico de los bits no es relevante para la conversión							

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

DTB: Convertir entero doble (32 bits) en BCD

Descripción

La instrucción "Convertir entero doble (32 bits) en BCD" convierte el valor que está en el acumulador 1 en un número decimal en código binario de siete dígitos (BCD). La instrucción interpreta el valor que debe convertirse como un entero doble.

Los bits 0 a 27 del acumulador 1 indican el valor que se convertirá. Se admiten valores dentro del rango "-9999999" a "+9999999". Si el valor que debe convertirse está fuera de este rango, los bits OV y OS se ponen al estado lógico "1". En este caso no se lleva a cabo la conversión.

El signo del valor del resultado se lee de los bits 28 a 31 del acumulador 1. Si el estado lógico de los bits es "0", el signo es positivo. Si el estado lógico de los cuatro bits es "1", el signo es negativo.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Convertir entero doble (32 bits) en BCD" se utiliza la siguiente sintaxis:

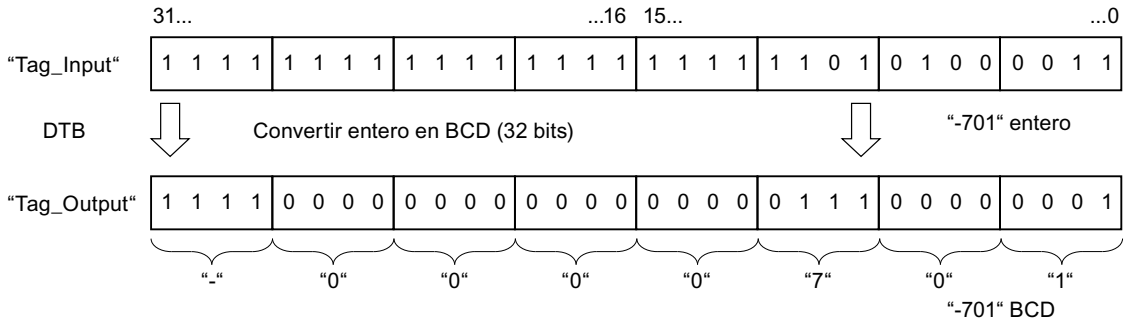
```
DTB
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero doble en el acumulador 1.
DTB	// Convertir entero doble en BCD
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

Principios básicos de la palabra de estado (Página 2545)

DTR: Convertir entero doble (32 bits) en número en coma flotante

Descripción

La instrucción "Convertir entero doble (32 bits) en número en coma flotante" convierte el valor que está en el acumulador 1 en un número en coma flotante. La instrucción interpreta el valor que debe convertirse como un entero doble.

Puesto que un entero doble es más exacto que un número en coma flotante, el resultado de la conversión se redondea al entero superior.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Convertir entero doble (32 bits) en número en coma flotante" se utiliza la siguiente sintaxis:

```
DTR
```

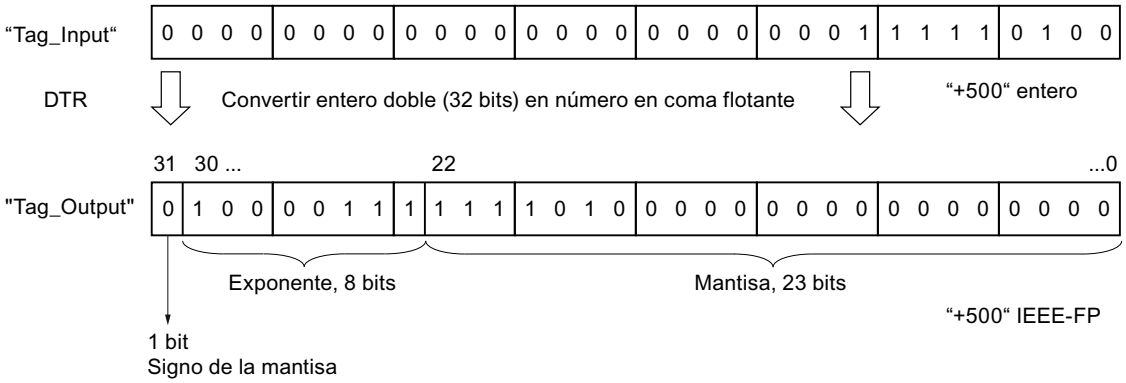
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar entero doble en el acumulador 1.

AWL	Explicación
DTR	// Convertir entero doble en número en coma flotante.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

INVI: Generar complemento a uno de un entero (16 bits)

Descripción

La instrucción "Generar complemento a uno de un entero (16 bits)" niega bit por bit el valor que se encuentra en la palabra derecha del acumulador 1.

Durante la ejecución, la instrucción invierte el estado lógico de cada uno de los bits de la palabra derecha del acumulador 1. Los unos son sustituidos por ceros y viceversa.

El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

Sintaxis

Para la instrucción "Generar complemento a uno de un entero (16 bits)" se utiliza la siguiente sintaxis:

```
INVI
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
INVI	// Generar un complemento a 1
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	0011	1010	1110
Tag_Output	1001	1100	0101	0001

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

INVD: Generar complemento a uno de un entero doble (32 bits)

Descripción

La instrucción "Generar complemento a uno de un entero doble (32 bits)" niega bit por bit el valor que se encuentra en el acumulador 1.

Durante la ejecución, la instrucción invierte el estado lógico de cada uno de los bits del acumulador 1. Los unos son sustituidos por ceros y viceversa.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Generar complemento a uno de un entero doble (32 bits)" se utiliza la siguiente sintaxis:

```
INVD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Input"                // Cargar el valor en el acumulador 1.
INVD                         // Generar un complemento a 1
T "Tag_Output"               // Transferir el resultado al operando "Tag_Output".
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0110	1111	1000	1100	0110	0011	1010	1110
Tag_Output	1001	0000	0111	0011	1001	1100	0101	0001

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

NEGI: Negar entero (16 bits)

Descripción

La instrucción "Negar entero (16 bits)" invierte el signo del valor que está en la palabra derecha del acumulador 1 por una formación de complemento a dos. La instrucción interpreta el valor del acumulador 1 como un entero de 16 bits.

La ejecución de la instrucción es equivalente a una multiplicación por "-1". El resultado de la instrucción se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no se ve influido por la instrucción.

La instrucción "Negar entero (16 bits)" influye en los bits de estado CC 0, CC 1, OV y OS. La tabla siguiente muestra cómo la instrucción influye en los bits de estado en función del resultado:

Resultado:	CC 1	CC 0	OV	OS
de +1 a +32 767	1	0	0	-
0	0	0	0	-
de -1 a -32 767	0	1	0	-
(-) 32 768	0	1	1	1

Sintaxis

Para la instrucción "Negar entero (16 bits)" se utiliza la siguiente sintaxis:

```
NEGI
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
NEGI	// Negar entero de 16 bits
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1101	0011	1000
Tag_Output	1010	0010	1100	1000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

NEGD: Negar entero doble (32 bits)

Descripción

La instrucción "Negar entero doble (32 bits)" invierte el signo del valor que se encuentra en el acumulador 1 por una formación de complemento a dos. La instrucción interpreta el valor del acumulador 1 como un entero doble.

La ejecución de la instrucción es equivalente a una multiplicación por "-1". El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La instrucción "Negar entero doble (32 bits)" influye en los bits de estado CC 0, CC 1, OV y OS. La tabla siguiente muestra cómo la instrucción influye en los bits de estado en función del resultado:

Resultado	CC 1	CC 0	OV	OS
de +1 a +2 147 483 647	1	0	0	-
0	0	0	0	-
de -1 a -2 147 483 647	0	1	0	-
(-) 2 147 483 648	0	1	1	1

Sintaxis

Para la instrucción "Negar entero doble (32 bits)" se utiliza la siguiente sintaxis:

```
NEGD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor en el acumulador 1.
NEGD	// Negar entero doble de 32 bits.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Estado	Valor							
	31 . . .	. . . .	. . . .	. . . 16	15 . . .	. . . .	. . . .	. . . 0
Tag_Input	0101	1111	0110	0100	0101	1101	0011	1000
Tag_Output	1010	0000	1001	1011	1010	0010	1100	1000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar los bits de estado en instrucciones con números enteros (Página 2551)

NEGR: Negar número en coma flotante

Descripción

La instrucción "Negar número en coma flotante" invierte el estado lógico del bit 31 en el acumulador 1. La instrucción interpreta el valor del acumulador 1 como número en coma flotante. El bit 31 del número en coma flotante contiene el signo de la mantisa.

La ejecución de la instrucción es equivalente a una multiplicación por "-1". Tras ejecutar la instrucción se invierte el signo de la mantisa.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La instrucción no influye en los bits de estado.

Sintaxis

Para la instrucción "Negar número en coma flotante" se utiliza la siguiente sintaxis:

```
NEGR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
NEGR	// Negar número en coma flotante.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Input	1.5E+02
Tag_Output	-1.5E+02

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

CAW: Invertir orden de los bytes en la palabra derecha del acumulador 1

Descripción

La instrucción "Invertir orden de los bytes en la palabra derecha del acumulador 1" invierte el orden de los dos bytes derechos en la palabra derecha del acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor B	Valor D	Valor C

El resultado de la instrucción se guarda en la palabra derecha del acumulador 1. Los bytes de la palabra izquierda del acumulador 1 no se ven influidos por la instrucción y no cambian.

Sintaxis

Para la instrucción "Invertir orden de los bytes en la palabra derecha del acumulador 1" se utiliza la siguiente sintaxis:

```
CAW
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`L "Tag_Input"`	`// Cargar el valor en el acumulador.`
`CAW`	`// Invertir orden de los bytes en la palabra derecha del acumulador 1.`
`T "Tag_Output"`	`// Transferir el resultado al operando "Tag_Output".`

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Input	0000	0000	1111	1111
Tag_Output	1111	1111	0000	0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

CAD: Invertir orden de los bytes en todo el acumulador 1

Descripción

La instrucción "Invertir orden de los bytes en todo el acumulador 1" invierte el orden de los bytes en el acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor D	Valor C	Valor B	Valor A

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

Sintaxis

Para la instrucción "Invertir orden de los bytes en todo el acumulador 1" se utiliza la siguiente sintaxis:

```
CAD
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el valor del operando en el acumulador 1.
CAD	// Invertir orden de los bytes del acumulador 1.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Input	1111	0000	0000	1111	0000	0000	1111	1111
Tag_Output	1111	1111	0000	0000	0000	1111	1111	0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

RND: Redondear número

Descripción

La instrucción "Redondear número" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como número en coma flotante y lo redondea al siguiente entero.

Si el número en coma flotante se encuentra exactamente entre un número par y uno impar, se selecciona el número par como resultado.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Redondear número" se utiliza la siguiente sintaxis:

```
RND
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
RND	// Redondear un número en coma flotante a un entero.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	101.5	-101.5
Tag_Output	102	-102

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como número en coma flotante y devuelve como resultado su parte entera.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Truncar a entero" se utiliza la siguiente sintaxis:

```
TRUNC
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
TRUNC	// Convertir el número en coma flotante en un entero del tipo de datos DINT.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	101.5	-101.5
Tag_Output	101	-101

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

RND+: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como número en coma flotante y lo redondea al siguiente entero superior. El resultado de la instrucción es un número del tipo de datos DINT, mayor o igual al número en coma flotante que se ha convertido.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero superior" se utiliza la siguiente sintaxis:

```
RND+
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
RND+	// Redondear el número en coma flotante al siguiente entero superior.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	100.5	-100.5
Tag_Output	101	-100

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

RND-: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" convierte el valor que se encuentra en el acumulador 1 en un entero doble (de 32 bits). La instrucción interpreta el valor que debe convertirse como número en coma flotante y lo redondea al siguiente entero inferior. El resultado de la instrucción es un número del tipo de datos DINT, menor o igual al número en coma flotante que se ha convertido.

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La conversión no se ejecuta y los bits de estado OV y OS se activan si se cumple una de las condiciones siguientes:

- El valor que se encuentra en el acumulador 1 no es un número en coma flotante válido.
- El resultado está fuera del rango permitido para un entero del tipo de datos DINT.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero inferior" se utiliza la siguiente sintaxis:

```
RND-
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input"	// Cargar el número en coma flotante en el acumulador 1.
RND-	// Redondear el número en coma flotante al siguiente entero inferior.
T "Tag_Output"	// Transferir el resultado al operando "Tag_Output".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores	
Tag_Input	100.5	-100.5
Tag_Output	100	-101

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar los bits de estado en instrucciones con números en coma flotante (Página 2552)

Principios básicos de la palabra de estado (Página 2545)

Control del programa

Operaciones de salto

Etiqueta

Descripción

Una etiqueta sirve para identificar el punto del programa en el que debe continuar la ejecución tras un salto. El nombre de la etiqueta puede comprender un máximo de 128 letras, cifras o caracteres de subrayado.

La etiqueta y la instrucción en la que se indica la etiqueta como destino del salto deben encontrarse en el mismo bloque. El nombre de una etiqueta debe ser unívoca en el bloque. A toda etiqueta se puede acceder desde distintas posiciones.

Es posible saltar hacia delante y hacia atrás. La amplitud máxima de salto es de -32 768 o +32 767 palabras de la lógica del programa. El número máximo real de instrucciones que pueden saltarse depende de la combinación de instrucciones dentro del programa (instrucciones de una, dos o tres palabras).

Si se utiliza la funcionalidad MCR, la etiqueta debe encontrase en el mismo rango de MCR que la instrucción de salto.

Sintaxis

Para la etiqueta se utiliza la siguiente sintaxis:

<etiqueta>:

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JC MyLABEL	// Si el resultado de la consulta es "1", saltar a la etiqueta "MyLABEL" y continuar allí el procesamiento del programa
	// Si el resultado de la consulta es "0", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output".
MyLABEL: A "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Consultar si el operando "Tag_Input_4" está a "1".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Vista general de los tipos de datos válidos (Página 2321)

JU: Salto incondicional

Descripción

La instrucción "Salto incondicional" interrumpe la ejecución lineal del programa y continuarla en el punto marcado por la etiqueta indicada. Esta instrucción se ejecuta siempre, independientemente de las condiciones.

La instrucción "Salto incondicional" no influye en los bits de estado.

Sintaxis

Para la instrucción "Salto incondicional" se utiliza la siguiente sintaxis:

```
JU <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Salto incondicional":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JC MyLABEL_1	// Si el resultado de la consulta es "1", saltar a la etiqueta "MyLABEL_1" y continuar allí el procesamiento del programa
	// Si el resultado de la consulta es "0", continuar con la ejecución lineal del programa.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output_1"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_1".
JU MyLABEL_2	// Saltar a la etiqueta "MyLABEL_2" y continuar allí con la ejecución del programa.
MyLABEL_1: L "Tag_Input_4"	// Etiqueta
	// Cargar el contenido del operando "Tag_Input_4" en el acumulador 1.

AWL	Explicación
T "Tag_Output_2"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_2".
MyLABEL_2: A "Tag_Input_5"	// Etiqueta "MyLABEL_2"
	// Consultar si el operando "Tag_Input_5" está a "1".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de una cadena secuencial (Página 3269)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

Principios básicos de la palabra de estado (Página 2545)

JC: Saltar si RLO = 1

Descripción

La instrucción "Saltar si RLO = 1" interrumpe la ejecución lineal del programa en función del resultado lógico y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "1". Si el RLO actual antes de la instrucción es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

La instrucción "Saltar si RLO = 1" pone el RLO y el bit de estado STA a "1" y los bits de estado OR y /FC a "0", tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 1" se utiliza la siguiente sintaxis:

```
JC <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JC MyLABEL	// Si el RLO es "1", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "0", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output_1"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_1".
MyLABEL: L "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Cargar el contenido del operando "Tag_Input_4" en el acumulador 1.
T "Tag_Output_2"	// Transferir el contenido del acumulador 1 al operando "Tag_Output_2".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de una cadena secuencial (Página 3269)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

Principios básicos de la palabra de estado (Página 2545)

JCN: Saltar si RLO = 0

Descripción

La instrucción "Saltar si RLO = 0" interrumpe la ejecución lineal del programa en función del resultado lógico (RLO) y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "0". Si el RLO actual antes de la instrucción es "1", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

La instrucción "Saltar si RLO = 0" pone el RLO y el bit de estado STA a "1" y los bits de estado OR y /FC a "0", tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 0" se utiliza la siguiente sintaxis:

```
JCN <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 0":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y.
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y.
JCN MyLABEL	// Si el RLO es "0", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "1", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output".
MyLABEL: A "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Consultar si el operando "Tag_Input_4" es "1" y combinar el resultado lógicamente con Y con el RLO.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de una cadena secuencial (Página 3269)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JCB: Saltar si RLO = 1 y guardar RLO

Descripción

La instrucción "Saltar si RLO = 1 y guardar RLO" interrumpe la ejecución lineal del programa en función del resultado lógico (RLO) y continuarla en el punto marcado por la etiqueta indicada. Paralelamente, el estado lógico del RLO actual se copia en el resultado binario (BR).

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "1". En este caso, la instrucción pone el resultado binario a "1".

Si el RLO actual antes de la instrucción es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente. En tal caso, la instrucción asigna el estado lógico "0" al resultado binario.

La instrucción "Saltar si RLO = 1 y guardar RLO" pone el RLO y el bit de estado STA a "1" y los bits de estado OR y /FC a "0", tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 1 y guardar RLO" se utiliza la siguiente sintaxis:

```
JCB <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 1 y guardar RLO":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Input_1"	// Cargar el primer valor de comparación.
L "Tag_Input_2"	// Cargar el segundo valor de comparación.
>I	// Consultar si el valor del operando "Tag_Input_1" es mayor que el valor del operando "Tag_Input_2".
JCB MyLABEL	// Copiar RLO actual en BR.
	// Si el RLO es "1", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "0", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output_1"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_1".

AWL	Explicación
MyLABEL: L "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto. // Cargar el contenido del operando "Tag_Input_4" en el acumulador 1.
T "Tag_Output_2"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_2".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

Vista general de los tipos de datos válidos (Página 2321)

JNB: Saltar si RLO = 0 y guardar RLO

Descripción

La instrucción "Saltar si RLO = 0 y guardar RLO" interrumpe la ejecución lineal del programa en función del resultado lógico (RLO) y continuarla en el punto marcado por la etiqueta indicada. Paralelamente, el estado del resultado lógico actual se copia en el resultado binario (BR).

El salto a la etiqueta indicada se ejecuta solo si el RLO actual es "0". En tal caso, la instrucción pone el resultado binario a "0".

Si el RLO actual antes de la instrucción es "1", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente. En tal caso, la instrucción asigna el estado lógico "1" al resultado binario.

La instrucción "Saltar si RLO = 0 y guardar RLO" pone el RLO y el bit de estado STA a "1" y los bits de estado OR y /FC a "0", tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si RLO = 0 y guardar RLO" se utiliza la siguiente sintaxis:

```
JNB <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si RLO = 0 y guardar RLO":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y.
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y.
JNB MyLABEL	// Copiar RLO actual en BR.
	// Si el RLO es "0", saltar a la etiqueta "MyLABEL" y continuar allí la ejecución del programa.
	// Si el RLO es "1", procesar la instrucción siguiente.
L "Tag_Input_3"	// Cargar el contenido del operando "Tag_Input_3" en el acumulador 1.
T "Tag_Output"	// Transferir el contenido del acumulador 1 al operando "Tag_Output".
MyLABEL: A "Tag_Input_4"	// Continuar la ejecución del programa en este punto en caso de haberse ejecutado un salto.
	// Consultar si el operando "Tag_Input_4" es "1" y combinar el resultado lógicamente con Y con el RLO.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

Vista general de los tipos de datos válidos (Página 2321)

JBI: Saltar si BR = 1

Descripción

La instrucción "Saltar si BR = 1" interrumpe la ejecución lineal del programa en función del resultado binario y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado BR es "1". Si el bit de estado BR es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

La instrucción "Saltar si BR = 1" pone el bit de estado STA a "1" y los bits de estado OR y /FC a "0", tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si BR = 1" se utiliza la siguiente sintaxis:

```
JBI <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si BR = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
SET                          // Poner el RLO a "1" al principio del bloque.
SAVE                         // Transferir el estado lógico del RLO al bit BR.
....                         // Cualquier programa
AN OV                        // Consultar el bit OV al final del bloque.
SAVE                         // En caso de desbordamiento, poner el bit BR a "0".
                             // Poner BR a "1" si no se ha producido un
                             desbordamiento.
JBI END                      // Si BR = "1", saltar a la etiqueta END.
                             // Si BR = "0", procesar la instrucción siguiente.
R "Tag_Output_1"             // Poner el operando "Tag_Output_1" a "0".
END: S "Tag_Output_2"        // Poner el operando "Tag_Output_2" a "1".
```

Consulte también

Añadir instrucciones AWL (Página 2767)
Procesar instrucciones AWL (Página 2782)
Principios básicos de la palabra de estado (Página 2545)
Activar el bit de estado (Página 2547)
Activar el bit de estado en instrucciones de control del programa (Página 2553)
Consultar el bit de estado con instrucciones de salto (Página 2557)
Vista general de los tipos de datos válidos (Página 2321)

JNBI: Saltar si BR = 0

Descripción

La instrucción "Saltar si BR = 0" interrumpe la ejecución lineal del programa en función del resultado binario (BR) y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado BR es "0". Si el bit de estado BR antes de la instrucción es "1", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

La instrucción "Saltar si BR = 0" pone el bit de estado STA a "1" y los bits de estado OR y /FC a "0", tanto si la condición se cumple como si no.

Sintaxis

Para la instrucción "Saltar si BR = 0" se utiliza la siguiente sintaxis:

```
JNBI <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si BR = 0":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
SET	// Poner el RLO a "1" al principio del bloque.
SAVE	// Transferir el estado lógico del RLO al bit BR.
....	// Cualquier programa
AN OV	// Consultar el bit OV al final del bloque.
SAVE	// En caso de desbordamiento, poner el bit BR a "0".
	// Poner BR a "1" si no se ha producido un desbordamiento.
JNBI END	// Si BR = "0", saltar a la etiqueta END.
	// Si BR = "1", procesar la instrucción siguiente.
S "Tag_Output"	// Poner el operando "Tag_Output" a "1".
END: BE	// Finalizar bloque

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JO: Saltar si OV = 1

Descripción

La instrucción "Saltar si OV = 1" interrumpe la ejecución lineal del programa en función del bit de estado OV y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado OV es "1". El bit de estado OV se activa si, p. ej., el resultado de un cálculo está fuera del rango permitido o si al comparar números en coma flotante hay un valor no válido.

Si el bit de estado OV es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si OV = 1" se utiliza la siguiente sintaxis:

```
JO <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si OV = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                         Explicación
L "Tag_Value_1"             // Cargar valor
L "Tag_Value_2"             // Cargar valor
*I                          // Multiplicar valores
```

AWL	Explicación
JO OVER	// Si el estado lógico del bit de estado OV es "1", saltar a la etiqueta "OVER" y continuar allí la ejecución del programa.
	// Si el estado lógico del bit de estado OV es "0", procesar la instrucción siguiente.
T "Tag_Result"	// Transferir el resultado de la multiplicación al operando "Tag_Result".
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
OVER: SET	// Continuar la ejecución del programa en este punto si se cumple la condición (OV = "1").
	// Poner el RLO al estado lógico "1".
R "Tag_Output"	// Poner el operando "Tag_Output" a "0".
NEXT: A "MyTag_1"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_1" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_2"	// Consultar si el operando "MyTag_2" es "1" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO = 1, poner el operando "Tag_Output_2" a "1".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JOS: Saltar si OS = 1

Descripción

La instrucción "Saltar si OS = 1" interrumpe la ejecución lineal del programa en función del bit de estado OS y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta se ejecuta solo si el bit de estado OS es "1". El bit de estado OS se activa siempre que un desbordamiento del rango numérico pone el bit de estado OV a "1". Al contrario que el bit de estado OV, el bit OS se mantiene activado aunque posteriormente haya un resultado que esté dentro del rango numérico permitido.

Si el bit de estado OS es "0", el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si OS = 1" se utiliza la siguiente sintaxis:

```
JOS <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si OS = 1":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                         Explicación
L "Tag_Value_1"             // Cargar valor
L "Tag_Value_2"             // Cargar valor
*I                          // Multiplicar valores cargados.
                            // Guardar el producto en el acumulador 1.
L "Tag_Value_3"             // Cargar valor
+I                          // Sumar el valor cargado al producto.
                            // Guardar el producto en el acumulador 1.
L "Tag_Value_4"             // Cargar valor
-I                          // Restar el valor cargado de la suma calculada.
                            // Guardar el producto en el acumulador 1.
JOS OVER                    // En caso de desbordamiento en una de las tres
                            instrucciones precedentes, el bit de estado OS se pone
                            a "1".
                            // Si el estado lógico del bit de estado OS es "1" la
                            ejecución del programa continúa en la posición de la
                            etiqueta "OVER".
                            // Si el estado lógico del bit de estado OS es "0",
                            procesar la instrucción siguiente.
T "Tag_Result"              // Transferir el resultado de todo el cálculo al
                            operando "Tag_Result".
JU NEXT                     // Saltar a la etiqueta "NEXT" y continuar allí con la
                            ejecución del programa.
OVER: SET                   // Continuar la ejecución del programa en este punto
                            si se cumple la condición (OV = "1").
                            // Poner el RLO al estado lógico "1".
R "Tag_Output"              // Poner el operando "Tag_Output" a "0".
NEXT: A "MyTag_1"           // Etiqueta "NEXT"
                            // Consultar si el operando "MyTag_1" es "1" y
                            combinar el resultado lógicamente con Y.
```

AWL	Explicación
A "MyTag_2"	// Consultar si el operando "MyTag_2" es "1" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO = 1, poner el operando "Tag_Output_2" a "1".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JZ: Saltar si el resultado es cero

Descripción

La instrucción "Saltar si el resultado es cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y continuarla en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico de los bits de estado CC 0 y CC 1 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el acumulador 1 tiene el valor "0".
- Al ejecutar la instrucción "Sumar enteros (16 bits)" (+I) o "Sumar enteros dobles (32 bits)" (+D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética (aritmética de punto flotante) con desbordamiento se produce un rebasamiento inferior gradual del rango permitido.
- El contenido del acumulador 2 es igual al contenido del acumulador 1 después de ejecutar una instrucción de comparación.
- El contenido del acumulador 1 es cero después de ejecutar una operación lógica con palabras.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es "0".

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es cero" se utiliza la siguiente sintaxis:

```
JZ <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando "Tag_Value".
SRW 1	// Desplazar el contenido de la palabra derecha del acumulador 1 una posición a la derecha.
JZ ZERO	// Si el estado lógico del último bit desplazado hacia fuera es "0", saltar a la etiqueta "OVER" y continuar allí el procesamiento del programa
	// Si el estado lógico del último bit desplazado hacia fuera es "1", procesar la instrucción siguiente.
L "MyTag_1"	// Cargar el valor del operando "MyTag_1" en el acumulador 1.
INC 1	// Sumar el valor "1" al byte derecho del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
ZERO: L "MyTag_2"	// Etiqueta "ZERO"
	// Cargar el valor del operando "MyTag_2" en el acumulador 1.
INC 1	// Sumar el valor "1" al byte derecho del acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JN: Saltar si el resultado no es cero

Descripción

La instrucción "Saltar si el resultado no es cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico de los bits de estado CC 0 y CC 1 es distinto. Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es distinto de cero.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, *I, +D, -D, *D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, *I, /I, +D, -D, *D, /D, NEGI, NEGD) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango positivo o negativo.
- El contenido del acumulador 2 es distinto al contenido del acumulador 1 después de ejecutar una instrucción de comparación.
- El contenido del acumulador 1 no es cero después de ejecutar una operación lógica con palabras.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es "1".

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado no es cero" se utiliza la siguiente sintaxis:

```
JN <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado no es cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando "Tag_Valu_1".
L "Tag_Value_2"              // Cargar el valor del operando "Tag_Value_2".
XOW                          // Combinar lógicamente con O-exclusiva
JN NOZERO                    // Cuando el valor del acumulador 1 no es cero, se
                             salta a la etiqueta "NOZERO" donde se continúa
                             ejecutando el programa.
                             // Cuando el valor en el acumulador 1 es igual a cero
                             se ejecuta la siguiente instrucción.
AN "MyTag_1"                 // Consultar si el operando "MyTag_1" es "0" y
                             combinar el resultado lógicamente con Y.
S "Tag_Output_1"             // Si el RLO es "1", poner el operando "Tag_Output_1"
                             a "1"
JU NEXT                      // Saltar a la etiqueta "NEXT" y continuar allí con la
                             ejecución del programa.
NOZERO: AN "MyTag_2          // Etiqueta "NOZERO"
                             // Consultar si el operando "MyTag_2" es "0" y
                             combinar el resultado lógicamente con Y.
S "Tag_Output_2"             // Si el RLO es "1", poner el operando "Tag_Output_2"
                             a "1".
NEXT: A "MyTag_3"            // Etiqueta "NEXT"
                             // Consultar si el operando "MyTag_3" es "1" y
                             combinar el resultado lógicamente con Y.
A "MyTag_4"                  // Consultar si el operando "MyTag_4" es "1" y
                             combinar el resultado lógicamente con Y.
```

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JP: Saltar si el resultado es mayor que cero

Descripción

La instrucción "Saltar si el resultado es mayor que cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 0 es "0" y el del bit de estado CC 1 es "1". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es mayor que cero.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (*I, /I, *D, /D) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango positivo.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es mayor que el contenido del acumulador 1.
- Tras ejecutar una operación lógica con palabras, el contenido del acumulador 1 no es cero.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es "1".

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es mayor que cero" se utiliza la siguiente sintaxis:

```
JP <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es mayor que cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JP POSITIVE	// Si el valor del resultado en el acumulador 1 es positivo, saltar a la etiqueta "POSITIVE" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es negativo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
POSITIVE: AN "MyTag_2	// Etiqueta "POSITIVE"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JM: Saltar si el resultado es menor que cero

Descripción

La instrucción "Saltar si el resultado es menor que cero" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 0 es "1" y el del bit de estado CC 1 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es menor que cero.
- Al ejecutar una instrucción aritmética con enteros (*I, *D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D, NEGI, NEGD) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango negativo.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es menor que el contenido del acumulador 1.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es menor que cero" se utiliza la siguiente sintaxis:

```
JM <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es menor que cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                     Explicación
L "Tag_Value_1"         // Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"         // Cargar el valor del operando "Tag_Value_2".
```

AWL	Explicación
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JM NEGATIVE	// Si el valor del resultado en el acumulador 1 es negativo, saltar a la etiqueta "NEGATIVE" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es positivo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
NEGATIVE: AN "MyTag_2	// Etiqueta "NEGATIVE"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JPZ: Saltar si el resultado es mayor o igual a cero

Descripción

La instrucción "Saltar si el resultado es mayor o igual a cero" interrumpe la ejecución lineal del programa en función del bit de estado CC 0 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 0 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es mayor o igual que cero.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (*I, /I, *D, /D) se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango positivo.
- Al ejecutar una instrucción aritmética (aritmética de punto flotante) con desbordamiento se produce un rebasamiento inferior gradual del rango permitido.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es mayor o igual al contenido del acumulador 1.
- Se ha ejecutado una instrucción de operación lógica por palabras.
- Se ha ejecutado una instrucción de desplazamiento.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es mayor o igual a cero" se utiliza la siguiente sintaxis:

```
JPZ <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es mayor o igual a cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JPZ REGULAR	// Si el valor del resultado en el acumulador 1 es positivo, saltar a la etiqueta "REGULAR" y continuar allí el procesamiento del programa

AWL	Explicación
	// Si el valor del resultado en el acumulador 1 es negativo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
REGULAR: AN "MyTag_2	// Etiqueta "REGULAR"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si el RLO es "1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JMZ: Saltar si el resultado es menor o igual a cero

Descripción

La instrucción "Saltar si el resultado es menor o igual a cero" interrumpe la ejecución lineal del programa en función del bit de estado CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico del bit de estado CC 1 es "0". Esto sucede cuando se cumple una de las condiciones siguientes:

- Después de ejecutar una instrucción aritmética sin desbordamiento, el valor del acumulador 1 es menor o igual que cero.
- Al ejecutar una instrucción aritmética con enteros (+I, *I, +D, *D) se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética con enteros (+I, -I, +D, -D, NEGI, NEGD) se produce un desbordamiento en el rango positivo.

- Al ejecutar una instrucción aritmética con números en coma flotante se produce un desbordamiento en el rango negativo.
- Al ejecutar una instrucción aritmética (aritmética de punto flotante) con desbordamiento se produce un rebasamiento inferior gradual del rango permitido.
- Tras ejecutar una instrucción de comparación, el contenido del acumulador 2 es menor o igual al contenido del acumulador 1.
- Tras ejecutar una operación lógica con palabras, el valor del acumulador 1 es cero.
- Tras ejecutar una instrucción de desplazamiento, el valor del último bit desplazado es cero.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado es menor o igual a cero" se utiliza la siguiente sintaxis:

```
JMZ <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado es menor o igual a cero":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"	// Cargar el valor del operando "Tag_Value_2".
-I	// Restar el valor del operando "Tag_Value_2" del valor del operando "Tag_Value_1".
JMZ MyLABEL	// Si el valor del resultado en el acumulador 1 es negativo, saltar a la etiqueta "MyLABEL" y continuar allí el procesamiento del programa
	// Si el valor del resultado en el acumulador 1 es positivo, procesar la instrucción siguiente.
AN "MyTag_1"	// Consultar si el operando "MyTag_1" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_1"	// Si el RLO es "1", poner el operando "Tag_Output_1" a "1"
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.

AWL	Explicación
MyLABEL: AN "MyTag_2	// Etiqueta "MyLABEL"
	// Consultar si el operando "MyTag_2" es "0" y combinar el resultado lógicamente con Y.
S "Tag_Output_2"	// Si RLO="1", poner el operando "Tag_Output_2" a "1"
NEXT: A "MyTag_3"	// Etiqueta "NEXT"
	// Consultar si el operando "MyTag_3" es "1" y combinar el resultado lógicamente con Y.
A "MyTag_4"	// Consultar si el operando "MyTag_4" es "1" y combinar el resultado lógicamente con Y.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JUO: Saltar si el resultado no es válido

Descripción

La instrucción "Saltar si el resultado no es válido" interrumpe la ejecución lineal del programa en función de los bits de estado CC 0 y CC 1 y la continúa en el punto marcado por la etiqueta indicada.

El salto a la etiqueta indicada se ejecuta solo si el estado lógico de los bits de estado CC 0 y CC 1 es "1". Esto sucede cuando se cumple una de las condiciones siguientes:

- En una instrucción matemática (/I, /D, MOD) se divide entre cero.
- En la ejecución de una instrucción matemática con números en coma flotante se produce un desbordamiento y el resultado es un número en coma flotante no válido.
- En la ejecución de una instrucción de comparación con números en coma flotante se ha empleado un número en coma flotante no válido o ha dado como resultado.

En los restantes casos, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente.

Sintaxis

Para la instrucción "Saltar si el resultado no es válido" se utiliza la siguiente sintaxis:

```
JUO <etiqueta>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Saltar si el resultado no es válido":

Parámetro	Descripción
<etiqueta>	Nombre simbólico del destino del salto

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando "Tag_Value_1".
L "Tag_Value_2"              // Cargar el valor del operando "Tag_Value_2".
/I                           // Dividir el valor del operando "Tag_Value_1" por el
                             valor del operando "Tag_Value_2".
JUO ERROR                    // En una división por cero, saltar a la etiqueta
                             "ERROR" y continuar allí el procesamiento del programa
                             // En otro caso, procesar la instrucción siguiente.
T "Tag_Result"               // Transferir el contenido del acumulador 1 al
                             operando "Tag_Result".
A "MyTag_1"                  // Consultar si el operando "MyTag_1" es "1" y
                             combinar el resultado lógicamente con Y.
R "MyTag_1"                  // Si el RLO es "1", poner el operando "MyTag_1" a "0".
JU NEXT                      // Saltar a la etiqueta "NEXT" y continuar allí con la
                             ejecución del programa.
ERROR: AN "MyTag_1           // Etiqueta "ERROR"
                             // Consultar si el operando "MyTag_1" es "0" y
                             combinar el resultado lógicamente con Y.
S "MyTag_1"                  // Si el RLO es "1", poner el operando "MyTag_1" a "1"
NEXT: A "MyTag_3"            // Etiqueta "NEXT"
                             // Consultar si el operando "MyTag_3" es "1" y
                             combinar el resultado lógicamente con Y.
A "MyTag_4"                  // Consultar si el operando "MyTag_4" es "1" y
                             combinar el resultado lógicamente con Y.
```

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Principios básicos de la palabra de estado (Página 2545)

Activar el bit de estado (Página 2547)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

JL: Definir lista de saltos

Descripción

La instrucción "Definir lista de saltos" permite programar una lista compuesta por varias entradas de la instrucción "Salto incondicional" (JU). La lista empieza inmediatamente después de la instrucción "Definir lista de saltos" y puede contener un máximo de 255 entradas. La numeración de las funciones de salto de la lista empieza por cero. La lista de saltos debe programarse sin huecos. El final de la lista de saltos se marca con una etiqueta que se indica en la instrucción "Definir lista de saltos".

La función de salto de la lista que se ejecutará depende del valor del byte derecho del acumulador 1. Si, p. ej., en el acumulador 1 hay el valor "0", se ejecutará la primera función de salto. Si el valor del acumulador 1 es "1", se ejecutará la segunda función de salto. Si el valor del acumulador "1" es mayor que el número de entradas de la lista, la instrucción "Definir lista de saltos" remite al final de la lista.

La instrucción "Distribuidor de saltos" se ejecuta independientemente de las condiciones y no influye en los bits de estado.

Sintaxis

La instrucción "Definir lista de saltos" se programa siguiendo el siguiente esquema general:

```
L <número de salto>
JL <fin_etiqueta>
JU <etiqueta>
JU <etiqueta>
<etiqueta>
<fin_etiqueta>:
```

El operando "número de salto" contiene el número de la instrucción "Salto incondicional" (JU) que debe ejecutarse. Este número tiene que cargarse en el acumulador 1. Con la etiqueta "fin_etiqueta" se marca el fin de la lista de saltos.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el número de salto en el acumulador 1.
JL END	// Principio de la lista de saltos
JU MyLABEL_1	// Si el valor del acumulador 1 es "0", ejecutar y saltar a la etiqueta "MyLABEL_1".
JU MyLABEL_2	// Si el valor del acumulador 1 es "1", ejecutar y saltar a la etiqueta "MyLABEL_2".
JU MyLABEL_3	// Si el valor del acumulador 2 es "1", ejecutar y saltar a la etiqueta "MyLABEL_3".
END: L "MyTag_1"	// Fin de la lista de saltos

AWL	Explicación
	// Cargar el contenido del operando "MyTag_1" en el acumulador 1.
L "MyTag_2"	// Cargar el contenido del operando "MyTag_2" en el acumulador 1.
+I	// Sumar valores
T "Tag_Output_2"	// Cargar el contenido del acumulador 1 en el operando "Tag_Output_2".
MyLABEL_1: ...	// Etiqueta "MyLABEL_1"
....	// Cualquier programa
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
MyLABEL_2: ...	// Etiqueta "MyLABEL_2"
....	// Cualquier programa
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
MyLABEL_3: ...	// Etiqueta "MyLABEL_3"
....	// Cualquier programa
JU NEXT	// Saltar a la etiqueta "NEXT" y continuar allí con la ejecución del programa.
NEXT: ...	// Etiqueta "NEXT"

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de una cadena secuencial (Página 3269)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

Principios básicos de la palabra de estado (Página 2545)

LOOP: Bucle

Descripción

La instrucción "Bucle" permite programar bucles.

La instrucción interpreta la palabra derecha del acumulador 1 como un entero de 16 bits sin signo en el rango de 0 a 65 535. Durante la ejecución, la instrucción decrementa primero en uno el contenido del acumulador 1. Si el valor del acumulador 1 no es cero después de decrementar, se ejecuta el salto hasta la etiqueta indicada. Si el valor es cero, el salto no se ejecuta y la ejecución del programa continúa con la instrucción siguiente. Por consiguiente, el valor del acumulador 1 determina el número de bucles que se ejecutarán. Dicho número debe guardarse en un contador de bucles.

La instrucción "Bucle" no influye en los bits de estado.

> ⚠ **ADVERTENCIA**
>
> Si el contador de bucles no está preasignado o lo está con el valor "0", el contador de bucles se decrementa continuamente hacia el negativo, produciéndose un bucle sin fin que puede provocar el estado operativo STOP de la CPU.

Sintaxis

La instrucción "Bucle" se programa siguiendo el siguiente esquema general:

```
L <número>
<etiqueta>:
T <contador de bucles>
... cualquier programa
L <contador de bucles>
LOOP <etiqueta>
```

El operando "número" contiene el número de bucles ejecutados. El operando "contador de bucles" contiene el número de bucles pendientes de ser ejecutados. Al final del bucle, el contenido del contador de bucles se carga en el acumulador 1 y es decrementado en uno por la instrucción. El salto hasta la etiqueta al principio del bucle se ejecuta si el valor del acumulador no es cero después de decrementar.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el número de ejecuciones de bucle pendientes en la palabra derecha del acumulador 1.
START: T "Tag_Counter"	// Principio de la lista de saltos
	// Transferir el contenido del acumulador 1 al contador de bucles.
L "MyTag_1"	// Cargar el valor del operando "MyTag_1".
L "MyTag_2"	// Cargar el valor del operando "MyTag_2".
*D	// Multiplicar valores
T "MyTag_1"	// Transferir el resultado de la multiplicación al operando "MyTag_1".
L "Tag_Counter"	// Cargar el contenido del contador de bucles en el acumulador 1.
LOOP START	// Decrementar en uno el valor del acumulador 1.
	// Si el valor es distinto de cero, saltar al principio del bucle.
	// Si el valor es cero, procesar la instrucción siguiente.

AWL	Explicación
L "MyTag_2"	// Cargar el valor del operando "MyTag_2".
L 100	// Cargar el valor 100.
>I	// Comparar si el valor del operando "MyTag_2" es mayor que 200.
= "MyTag_3"	// Escribir el resultado de la comparación en el operando "MyTag_3".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Consultar el bit de estado con instrucciones de salto (Página 2557)

Principios básicos de la palabra de estado (Página 2545)

Bloques de datos

OPN: Abrir bloque de datos global

Descripción

La instrucción "Abrir bloque de datos global" abre un bloque de datos (DB) global. El número del bloque de datos se transmite al registro del DB. Los siguientes comandos de DB acceden a los bloques correspondientes en función de los contenidos del registro.

La instrucción "Abrir bloque de datos global" se ejecuta independientemente de las condiciones y no influye ni en el resultado lógico ni en los contenidos de los acumuladores.

Sintaxis

Para la instrucción "Abrir bloque de datos global" se utiliza la siguiente sintaxis:

```
OPN <bloque de datos>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos global":

Operando	Declaración	Tipo del bloque de datos	Descripción
<Bloque de datos>	-	DB	Bloque de datos que se abrirá.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global.
L %DBW0	// Cargar palabra de datos DBW0 del bloque de datos global abierto en acumulador 1.
T "MyTag"	// Transferir el contenido del acumulador 1 al operando "MyTag_1".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

OPNDI: Abrir bloque de datos de instancia

Descripción

La instrucción "Abrir bloque de datos de instancia" abre un bloque de datos de instancia (DI). El número del bloque de datos se transmite al registro del DI. Los siguientes comandos del DI acceden a los bloques correspondientes en función de los contenidos del registro.

La instrucción "Abrir bloque de datos de instancia" se ejecuta independientemente de las condiciones y no influye ni en el resultado lógico ni en los contenidos de los acumuladores.

Sintaxis

Para la instrucción "Abrir bloque de datos de instancia" se utiliza la siguiente sintaxis:

```
OPNDI <bloque de datos>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Abrir bloque de datos de instancia":

Operando	Declaración	Tipo del bloque de datos	Descripción
<Bloque de datos>	-	DI	Bloque de datos que se abrirá.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPNDI "InstanceDataBlock"	// Abrir bloque de datos de instancia
L %DIW0	// Cargar palabra de datos DIW0 del bloque de datos de instancia abierto en acumulador 1.
T "MyTag"	// Transferir el contenido del acumulador 1 al operando "MyTag".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CDB: Intercambiar registros de bloques de datos

Descripción

La instrucción "Intercambiar registros de bloques de datos" cambia los contenidos de los registros de los bloques de datos. La instrucción se ejecuta independientemente de las condiciones y no influye en los bits de estado.

Sintaxis

Para la instrucción "Intercambiar registros de bloques de datos" se utiliza la siguiente sintaxis:

```
CDB
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global
OPNDI "InstanceDataBlock"	// Abrir bloque de datos de instancia
CDB	// Intercambiar registros de bloques de datos
	// El registro DB remite a "InstanceDataBlock" y el registro DI remite a "GlobalDataBlock".
L %DIW0	// Cargar DW0 de "GlobalDataBlock" en el acumulador 1.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

L DBLG: Cargar longitud de un bloque de datos global en acumulador 1

Descripción

La instrucción "Cargar longitud de un bloque de datos global en acumulador 1" carga en el acumulador 1 la longitud de un bloque de datos global abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos global mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1. La longitud del bloque de datos es equivalente al número de bytes de datos.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar longitud de un bloque de datos global en acumulador 1" se utiliza la siguiente sintaxis:

```
L DBLG
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global
L DBLG	// Cargar longitud de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
<D	// Comparar si la longitud del bloque de datos es menor que el valor del operando "MyTag_1".
= "MyTag_2"	// Escribir el resultado de la consulta en el operando "MyTag_2".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

L DBNO: Cargar número de un bloque de datos global en acumulador 1

Descripción

La instrucción "Cargar número de un bloque de datos global en acumulador 1" carga en el acumulador 1 el número de un bloque de datos global abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos global mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar número de un bloque de datos global en acumulador 1" se utiliza la siguiente sintaxis:

```
L DBNO
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPN "GlobalDataBlock"	// Abrir bloque de datos global
L DBNO	// Cargar número de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
==I	// Comparar si el número del bloque de datos es igual al valor del operando "MyTag_1".
= "MyTag_2"	// Escribir el resultado de la consulta en el operando "MyTag_2".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

L DILG: Cargar longitud de un bloque de datos de instancia en acumulador 1

Descripción

La instrucción "Cargar longitud de un bloque de datos de instancia en acumulador 1" carga en el acumulador 1 la longitud de un bloque de datos de instancia abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos de instancia mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1. La longitud del bloque de datos es equivalente al número de bytes de datos.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar longitud de un bloque de datos de instancia en acumulador 1" se utiliza la siguiente sintaxis:

```
L DILG
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPNDI "InstanceDataBlock"	// Abrir bloque de datos de instancia
L DILG	// Cargar longitud de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
<I	// Comparar si la longitud del bloque de datos es menor que el valor del operando "MyTag_1".
= "MyTag_2"	// Escribir el resultado de la consulta en el operando "MyTag_2".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

L DINO: Cargar número de un bloque de datos de instancia en acumulador 1

Descripción

La instrucción "Cargar número de un bloque de datos de instancia en acumulador 1" carga en el acumulador 1 el número de un bloque de datos de instancia abierto mediante el registro del bloque de datos. El contenido anterior del acumulador 1 se desplaza al acumulador 2.

Si antes de ejecutar la instrucción no se ha abierto ningún bloque de datos de instancia mediante el registro del bloque de datos, se carga el valor "0" en el acumulador 1.

La instrucción no modifica los bits de estado.

Sintaxis

Para la instrucción "Cargar número de un bloque de datos de instancia en acumulador 1" se utiliza la siguiente sintaxis:

```
L DINO
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
OPNDI "InstanceDataBlock"	// Abrir bloque de datos de instancia
L DINO	// Cargar número de un bloque de datos abierto en acumulador 1.
L "MyTag_1"	// Cargar valor de comparación
==I	// Comparar si el número del bloque de datos es igual al valor del operando "MyTag_1".
= "MyTag_2"	// Asignar el resultado de la consulta al operando "MyTag_2".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Bloques lógicos

BE: Fin de bloque

Descripción

La instrucción "Fin de bloque" finaliza el procesamiento del bloque actual y pasa al punto del programa en el que se llama el bloque. El procesamiento del programa continúa con la instrucción que se encuentra justo después de la llamada del bloque.

Al finalizar el bloque, el resultado lógico actual se transfiere al bloque que efectúa la llamada.

La instrucción "Fin de bloque" se ejecuta siempre, independientemente de las condiciones. Si el procesamiento de la instrucción se salta debido a una instrucción de salto, la ejecución actual del programa no finaliza, sino que continúa en el destino del salto dentro del bloque.

Sintaxis

Para la instrucción "Fin de bloque" se utiliza la siguiente sintaxis:

```
BE
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
SIN	// Calcular el valor del seno
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto en el acumulador 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando "Tag_Result".
BE	// Finalizar bloque

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

BEC: Fin de bloque condicional

Descripción

La instrucción "Fin de bloque condicional" finaliza el procesamiento del bloque actual en función del resultado lógico (RLO) y pasa al punto del programa en el que se llama el bloque.

Si el RLO es "1", se ejecuta la instrucción. El bloque actual se finaliza y la ejecución del programa continúa en el bloque que efectúa la llamada. El procesamiento del programa continúa con la instrucción que se encuentra justo después de la llamada del bloque.

Al finalizar el bloque, el resultado lógico actual se transfiere al bloque que efectúa la llamada. En el bloque que efectúa la llamada, la dependencia de MCR se ajusta igual que estaba antes de llamar el bloque. Los bloques de datos que estaban abiertos en el momento de llamar el bloque vuelven a abrirse.

Si el RLO es "0" al procesar la instrucción, esta no se ejecuta. En este caso, la CPU pone el RLO a "1" y procesa la instrucción siguiente.

Si el procesamiento de la instrucción se salta debido a una instrucción de salto, la ejecución actual del programa no finaliza, sino que continúa en el destino del salto dentro del bloque.

Sintaxis

Para la instrucción "Fin de bloque condicional" se utiliza la siguiente sintaxis:

```
BEC
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A "Tag_Input"	// Consultar si el operando "Tag_Input" es "1" y combinar el resultado lógicamente con Y con el RLO.
BEC	// Si la condición se cumple (RLO = "1"), finalizar el bloque.
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando "Tag_Result".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

BEU: Fin de bloque incondicional

Descripción

La instrucción "Fin de bloque incondicional" finaliza el procesamiento del bloque actual y pasa al punto del programa en el que se llama el bloque. El procesamiento del programa continúa con la instrucción que se encuentra justo después de la llamada del bloque.

Al finalizar el bloque, el resultado lógico actual se transfiere al bloque que efectúa la llamada. En el bloque que efectúa la llamada, la dependencia de MCR se ajusta igual que estaba antes de llamar el bloque. Los bloques de datos que estaban abiertos en el momento de llamar el bloque vuelven a abrirse.

La instrucción "Fin de bloque incondicional" se ejecuta siempre, independientemente de las condiciones. Al contrario que la instrucción "Fin de bloque", es posible programar la instrucción "Fin de bloque incondicional" varias veces dentro de un bloque. Si el procesamiento de la instrucción se salta debido a una instrucción de salto, la ejecución actual del programa no finaliza, sino que continúa en el destino del salto dentro del bloque.

Sintaxis

Para la instrucción "Fin de bloque incondicional" se utiliza la siguiente sintaxis:

```
BEU
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A "Tag_Input"	// Consultar si el operando "Tag_Input" es "1" y combinar el resultado lógicamente con Y con el RLO.
JC NEXT	// Si la condición se cumple (RLO = "1"), continuar el procesamiento del programa a partir de la etiqueta "NEXT".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
BEU	// Finalizar bloque
NEXT: T "Tag_Result"	// Etiqueta "NEXT"
	// Transferir el contenido del acumulador 1 al operando "Tag_Result".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo de una cadena secuencial (Página 3269)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CALL: Llamar bloque

Descripción

La instrucción "Llamar bloque" llama los siguientes tipos de bloque en el programa:

- Funciones
- Bloques de función

La instrucción "Llamar bloque" se ejecuta siempre, independientemente de las condiciones. Tras ejecutar la instrucción, el procesamiento del programa continúa en el bloque llamado.

Existe la posibilidad de suministrar datos a un bloque llamado. Los datos se transfieren mediante los parámetros del bloque. Los parámetros del bloque llamado se listan en el bloque que efectúa la llamada después de la instrucción de llamada. Es posible asignarles los parámetros actuales necesarios. Al ejecutar la instrucción "Llamar bloque", los datos se transfieren al bloque llamado. Con la transmisión de los datos se modifican los contenidos tanto de la palabra de estado como del registro de direcciones y del de los bloques de datos.

Los parámetros de un bloque de función que no reciben valores nuevos conservan su valor actual. Al llamar funciones deben suministrarse valores a todos los parámetros. Si un bloque llamado no tiene parámetros, no se muestra la lista de parámetros.

Si el bloque llamado requiere un bloque de datos de instancia, éste debe indicarse en la llamada, separado por una coma. El bloque de datos indicado debe haberse creado antes de la llamada.

Una vez procesado el bloque llamado, la CPU vuelve al bloque que efectúa la llamada y continúa con el procesamiento de dicho bloque después de la instrucción de llamada.

En el bloque que efectúa la llamada, la dependencia de MCR se ajusta igual que estaba antes de llamar el bloque. Los bloques de datos que estaban abiertos en el momento de llamar el bloque vuelven a abrirse.

Sintaxis

Al llamar una función, se utiliza la siguiente sintaxis para la instrucción "Llamar bloque":

```
CALL <FC>
```

Al llamar un bloque de función, se utiliza la siguiente sintaxis para la instrucción "Llamar bloque":

```
CALL <FB>, <FB_DB>
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
CALL "MyFunction"	// Llamar la función "MyFunction"
Input_1 := "Tag_Input_1",	// Asignar parámetros actuales
Input_2 := "Tag_Input_2"	
Output_1 := "Tag_Output_1"	
Output_2 := "Tag_Output_1"	
CALL "MyFunctionBlock", "MyFB_DB"	// Llamar el bloque de función "MyFunctionBlock"
Value_1 := "Tag_Value_1"	// Asignar parámetros actuales
Value_2 := "Tag_Value_2"	
Output := "Tag_Output"	
CALL "LIMIT"	// Llamar la instrucción "Ajustar valor límite"
MN := "Tag_LowLimit"	// Asignar parámetros actuales
IN := "Tag_InputValue"	
MX := "Tag_HighLimit"	
OUT := "Tag_Output"	
CALL "CTU", "CTU_DB"	// Llamar el contador "Contador ascendente"
CU := "Tag_StartCTU"	// Asignar parámetros actuales
R := "Tag_ResetCounter"	
PV := "Tag_PresetValue"	
Q := "Tag_CounterStatus"	
CV := "Tag_CounterValue"	

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

CC: Llamada condicional

Descripción

La instrucción "Llamada condicional" llama en función del resultado lógico (RLO) funciones (FC) y bloques de función (FB) que no tienen parámetros ni bloques de datos de instancia.

La instrucción solo se ejecuta si el resultado lógico (RLO) actual antes de procesar la instrucción es "1". Tras ejecutar la instrucción, el procesamiento del programa continúa en el bloque llamado. Una vez procesado el bloque llamado, la CPU vuelve al bloque que efectúa

la llamada y continúa con el procesamiento de dicho bloque después de la instrucción de llamada.

Al cambiar el bloque, los bits de estado OS y /FC se ponen a "0". Los bits de estado CC 0, CC 1 y OV no se ven influidos.

En el bloque que efectúa la llamada, la dependencia de MCR se ajusta igual que estaba antes de llamar el bloque. Los bloques de datos que estaban abiertos en el momento de llamar el bloque vuelven a abrirse.

La instrucción "Llamada condicional" no modifica el contenido de los acumuladores ni los registros de direcciones.

Si el RLO actual es "0", la instrucción y, por consiguiente, la llamada de bloque no se ejecutan y el RLO se pone a "1".

Sintaxis

Al llamar una función, se utiliza la siguiente sintaxis para la instrucción "Llamada condicional":

```
CC <FC>
```

Al llamar un bloque de función, se utiliza la siguiente sintaxis para la instrucción "Llamada condicional":

```
CC <FB>
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CC "MyFunction"	// Si la condición se cumple (RLO = "1"), llamar la función "MyFunction".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
CC "MyFunctionBlock"	// Si la condición se cumple (RLO = "1"), llamar el bloque de función "MyFunctionBlock".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

UC: Llamada incondicional

Descripción

La instrucción "Llamada incondicional" llama funciones (FC) y bloques de función (FB) que no tienen parámetros ni bloques de datos de instancia.

Esta instrucción se ejecuta independientemente de las condiciones. Tras ejecutar la instrucción, el procesamiento del programa continúa en el bloque llamado. Una vez procesado el bloque llamado, la CPU vuelve al bloque que efectúa la llamada y continúa con el procesamiento de dicho bloque después de la instrucción de llamada.

Al cambiar el bloque, los bits de estado OS y /FC se ponen a "0". Los bits de estado CC 0, CC 1 y OV no se ven influidos.

En el bloque que efectúa la llamada, la dependencia de MCR se ajusta igual que estaba antes de llamar el bloque. Los bloques de datos que estaban abiertos en el momento de llamar el bloque vuelven a abrirse.

La instrucción "Llamada incondicional" no modifica el contenido de los acumuladores ni los registros de direcciones.

Sintaxis

Al llamar una función, se utiliza la siguiente sintaxis para la instrucción "Llamada incondicional":

```
UC <FC>
```

Al llamar un bloque de función, se utiliza la siguiente sintaxis para la instrucción "Llamada incondicional":

```
UC <FB>
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
A "Tag_Input_1"	// Consultar si el operando "Tag_Input_1" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
UC "MyFunction"	// Llamar la función "MyFunction".

AWL	Explicación
A "Tag_Input_2"	// Consultar si el operando "Tag_Input_2" es "1" y combinar el resultado lógicamente con Y con el RLO actual.
UC "MyFunctionBlock"	// Llamar el bloque de función "MyFunctionBlock".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

Operaciones lógicas con palabras

AW: Operación lógica Y por palabras

Descripción

La instrucción "Operación lógica Y por palabras" combina lógicamente con Y por bits el valor de la palabra derecha del acumulador 1 con el valor de la palabra derecha del acumulador 2 o una constante concreta. El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no cambia.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 15 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

La tabla siguiente muestra cómo se forma el resultado en una operación lógica Y por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	0	0	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras":

```
AW <constante>
```

En las operaciones lógicas con el valor de la palabra derecha del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras":

```
AW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y por palabras":

Parámetro	Tipo de datos	Descripción
<constante>	WORD	Valor que se combina lógicamente con Y con el valor que se encuentra en la palabra derecha del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
AW W#16#F6B5	// Combinar lógicamente con Y el valor de la palabra derecha del acumulador 1 con la constante (W#16#F6B5).
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en la palabra derecha del acumulador 1.
AW	// Combinar lógicamente con Y el valor de la palabra derecha del acumulador 2 ("Tag_Value_2") con el valor de la palabra derecha del acumulador 1 ("Tag_Value_3").
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Value_1	0101	1001	0011	1011
Constante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	0101	0000	0011	0001
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	0100	1000	0000	0010

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

OW: Operación lógica O por palabras

Descripción

La instrucción "Operación lógica O por palabras" combina lógicamente con O por bits el valor de la palabra derecha del acumulador 1 con el valor de la palabra derecha del acumulador 2 o una constante concreta. El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no cambia.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 15 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de los dos bits que deben combinarse lógicamente es "0", el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras":

```
OW <constante>
```

En las operaciones lógicas con el valor de la palabra derecha del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras":

```
OW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O por palabras":

Parámetro	Tipo de datos	Descripción
<constante>	WORD	Valor que se combina lógicamente con O con el valor que se encuentra en la palabra derecha del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
OW W#16#F6B5	// Combinar lógicamente con O el valor de la palabra derecha del acumulador 1 con la constante (W#16#F6B5).
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en la palabra derecha del acumulador 1.
OW	// Combinar lógicamente con O el valor de la palabra derecha del acumulador 2 ("Tag_Value_2") con el valor de la palabra derecha del acumulador 1 ("Tag_Value_3").
	// El resultado se guarda en la palabra derecha del acumulador 1.

AWL	Explicación
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Value_1	0101	1001	0011	1011
Constante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1111	1111	1011	1111
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1111	1110	1011	1011

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Ejemplo: Operaciones lógicas con palabras (Página 3267)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

XOW: Operación lógica O-exclusiva por palabras

Descripción

La instrucción "Operación lógica O-exclusiva por palabras" combina lógicamente con O-exclusiva por bits el valor de la palabra derecha del acumulador 1 con el valor de la palabra derecha del acumulador 2 o una constante concreta. El resultado se guarda en la palabra derecha del acumulador 1. El contenido de la palabra izquierda del acumulador 1 no cambia.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 15 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente es diferente. Si el estado lógico de los dos bits que deben combinarse lógicamente es el mismo, el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O-exclusiva por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	0

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras":

```
XOW <constante>
```

En las operaciones lógicas con el valor de la palabra derecha del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras":

```
XOW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva por palabras":

Parámetro	Tipo de datos	Descripción
<constante>	WORD	Valor que se combina lógicamente con O-exclusiva con el valor que se encuentra en la palabra derecha del acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en la palabra derecha del acumulador 1.
XOW W#16#F6B5	// Combinar lógicamente con O-exclusiva el valor de la palabra derecha del acumulador 1 con la constante (W#16#F6B5).
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en la palabra derecha del acumulador 1.

AWL	Explicación
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en la palabra derecha del acumulador 1.
XOW	// Combinar lógicamente con O-exclusiva el valor de la palabra derecha del acumulador 2 ("Tag_Value_2") con el valor de la palabra derecha del acumulador 1 ("Tag_Value_3").
	// El resultado se guarda en la palabra derecha del acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_Value_1	0101	1001	0011	1011
Constante (W#16#F6B5)	1111	0110	1011	0101
Tag_Result_1	1010	1111	1000	1110
Tag_Value_2	0110	1100	0010	1010
Tag_Value_3	1101	1010	1001	0011
Tag_Result_2	1011	0110	1011	1001

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

AD: Operación lógica Y por palabras dobles

Descripción

La instrucción "Operación lógica Y por palabras dobles" combina lógicamente con Y por bits el contenido del acumulador 1 con el contenido del acumulador 2 o el valor de una constante concreta. El resultado se guarda en el acumulador 1.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 31 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de uno de los bits que deben combinarse lógicamente es "0", se desactivará el bit de resultado correspondiente.

La tabla siguiente muestra cómo se forma el resultado en una operación lógica Y por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	0	0	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras dobles":

```
AD <constante>
```

En las operaciones lógicas con el valor del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica Y por palabras dobles":

```
AD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica Y por palabras dobles":

Parámetro	Tipo de datos	Descripción
<constante>	DWORD	Valor que se combina lógicamente con Y con el valor que está en el acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando en el acumulador 1.
AD DW#16#39C657AC            // Combinar lógicamente con Y el valor del acumulador
                             1 con la constante (DW#16#39C657AC).
```

AWL	Explicación
	// Guardar el resultado en el acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
AD	// Combinar lógicamente con Y el valor del acumulador 2 ("Tag_Value_2") con el valor del acumulador 1 ("Tag_Value_3").
	// Guardar el resultado en el acumulador 1.
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Constante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0001	1001	0100	0100	0001	0101	0010	1000
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0010	0001	0100	0010	0101	0101	0010	1010

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

OD: Operación lógica O por palabras dobles

Descripción

La instrucción "Operación lógica O por palabras dobles" combina lógicamente con O por bits el contenido del acumulador 1 con el contenido del acumulador 2 o el valor de una constante concreta. El resultado se guarda en el acumulador 1.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 31 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de como mínimo uno de los dos bits que deben combinarse lógicamente también es "1". Si el estado lógico de los dos bits que deben combinarse lógicamente es "0", el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	1

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras dobles":

```
OD <constante>
```

En las operaciones lógicas con el valor del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O por palabras dobles":

```
OD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O por palabras dobles":

Parámetro	Tipo de datos	Descripción
<constante>	DWORD	Valor que se combina lógicamente con O con el valor que está en el acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
OD DW#16#39C657AC	// Combinar lógicamente con O el valor del acumulador 1 con la constante (DW#16#39C657AC).
	// Guardar el resultado en el acumulador 1.
T "Tag_Result_1"	// Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"	// Cargar el valor del operando en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
OD	// Combinar lógicamente con O el valor del acumulador 2 ("Tag_Value_2") con el valor del acumulador 1 ("Tag_Value_3").
	// Guardar el resultado en el acumulador 1
T "Tag_Result_2"	// Transferir el resultado al operando "Tag_Result_2".

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Constante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0111	1111	1110	0110	1101	1111	1011	1111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0111	1101	1100	0111	0101	1111	1110	1111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

XOD: Operación lógica O-exclusiva por palabras dobles

Descripción

La instrucción "Operación lógica O-exclusiva por palabras dobles" combina lógicamente con O-exclusiva por bits el contenido del acumulador 1 con el contenido del acumulador 2 o el valor de una constante concreta. El resultado se guarda en el acumulador 1.

La instrucción combina el bit 0 del acumulador 1 con el bit 0 del acumulador 2 o una constante y guarda el resultado en el bit 0 del acumulador 1. Los bits 1 a 31 se combinan del mismo modo.

El estado lógico del bit de resultado es "1" si el estado lógico de los dos bits que deben combinarse lógicamente es diferente. Si el estado lógico de los dos bits que deben combinarse lógicamente es el mismo, el bit de resultado correspondiente se pondrá a "0".

La tabla siguiente muestra cómo se forma el resultado en una operación lógica O-exclusiva por palabras:

Acumulador 2 / constante	0	0	1	1
Acumulador 1	0	1	0	1
Resultado	0	1	1	0

La instrucción se ejecuta independientemente de las condiciones y no influye en el resultado lógico.

La instrucción influye en los bits de estado CC 0, CC 1 y OV.

Sintaxis

En las operaciones lógicas con una constante se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras dobles":

```
XOD <constante>
```

En las operaciones lógicas con el valor del acumulador 2 se utiliza la sintaxis siguiente para la instrucción "Operación lógica O-exclusiva por palabras dobles":

```
XOD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Operación lógica O-exclusiva por palabras dobles":

Parámetro	Tipo de datos	Descripción
<constante>	DWORD	Valor que se combina lógicamente con O-exclusiva con el valor que está en el acumulador 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando en el acumulador 1.
XOD DW#16#39C657AC           // Combinar lógicamente con O-exclusiva el valor del
                             acumulador 1 con la constante (DW#16#39C657AC).
                             // Guardar el resultado en el acumulador 1.
T "Tag_Result_1"             // Transferir el resultado al operando "Tag_Result_1".
L "Tag_Value_2"              // Cargar el valor del operando en el acumulador 1.
L "Tag_Value_3"              // Desplazar el contenido del acumulador 1 al
                             acumulador 2.
                             // Cargar el valor del operando "Tag_Value_3" en el
                             acumulador 1.
XOD                          // Combinar lógicamente con O-exclusiva el valor del
                             acumulador 2 ("Tag_Value_2") con el valor del
                             acumulador 1 ("Tag_Value_3").
                             // Guardar el resultado en el acumulador 1.
T "Tag_Result_2"             // Transferir el resultado al operando "Tag_Result_2".
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Constante (DW#16#39C657AC)	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_1	0110	0110	1010	0010	1100	1010	1001	0111
Tag_Value_2	0110	0101	0100	0011	0101	1101	0010	1011
Tag_Value_3	0011	1001	1100	0110	0101	0111	1010	1100
Tag_Result_2	0101	1100	1000	0101	0000	1010	1000	0111

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en operaciones lógicas con palabras (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Desplazamiento y rotación

Desplazar

SSI: Desplazar con signo por palabras

Descripción

La instrucción "Desplazar con signo por palabras" desplaza bit por bit la palabra derecha (bits de 0 a 15) del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con el estado lógico del bit 15 (bit con signo en números INT). Los bits 16 a 31 del acumulador 1 no cambian.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 15.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 15, todos los bits de la palabra derecha del acumulador 1 se rellenarán con el estado lógico del bit 15.

Si el número de desplazamiento indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras":

```
SSI <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras":

```
SSI
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar con signo por palabras":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando en el acumulador 1.
SSI 6                        // Desplazar los bits 0 a 15 del acumulador 1 seis
                             posiciones a la derecha.
                             // Rellenar las posiciones liberadas con el estado del
                             bit 15.
T "Tag_Result_1"             // Transferir el contenido del acumulador 1 al
                             operando.
L 3                          // Cargar el número de desplazamiento en el acumulador
                             1.
L "Tag_Value_2"              // Desplazar el número de desplazamiento al byte
                             derecho del acumulador 2.
                             // Cargar el valor del operando en el acumulador 1.
SSI                          // Desplazar los bits 0 a 15 del acumulador 1 tres
                             posiciones a la derecha.
                             // Rellenar las posiciones liberadas con el estado del
                             bit 15.
T "Tag_Result_2"             // Transferir el contenido del acumulador 1 al
                             operando.
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	1001	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1111	1110	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	1011	1010	0101

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

SSD: Desplazar con signo por palabras dobles

Descripción

La instrucción "Desplazar con signo por palabras dobles" desplaza bit por bit todo el contenido del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con el estado lógico del bit 31 (bit con signo en números DINT).

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 31.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 31, todos los bits del acumulador 1 se rellenarán con el estado lógico del bit 31.

Si el número de desplazamiento indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras dobles":

```
SSD <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar con signo por palabras dobles":

```
SSD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar con signo por palabras dobles":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
AWL                          Explicación
L "Tag_Value_1"              // Cargar el valor del operando en el acumulador 1.
SSD 7                        // Desplazar los bits 0 a 31 del acumulador 1 siete
                             posiciones a la derecha.
                             // Rellenar las posiciones liberadas con el estado del
                             bit 31.
T "Tag_Result_1"             // Transferir el contenido del acumulador 1 al
                             operando.
L 4                          // Cargar el número de desplazamiento en el acumulador
                             1.
L "Tag_Value_2"              // Desplazar el número de desplazamiento al byte
                             derecho del acumulador 2.
                             // Cargar el valor del operando en el acumulador 1.
SSD                          // Desplazar los bits 0 a 31 del acumulador 1 cuatro
                             posiciones a la derecha.
                             // Rellenar las posiciones liberadas con el estado del
                             bit 31.
T "Tag_Result_2"             // Transferir el contenido del acumulador 1 al
                             operando.
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	1000	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	1111	0001	1110	1100	1000	1011	1010
Tag_Value_2	0010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	0010	1000	1010	0010	1001	1011	1100

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

SLW: Desplazar a la izquierda por palabras

Descripción

La instrucción "Desplazar a la izquierda por palabras" desplaza bit por bit la palabra derecha (bits 0 a 15) del acumulador 1 hacia la izquierda. Las posiciones que quedan libres al desplazar se rellenan con ceros. Los bits 16 a 31 del acumulador 1 no cambian.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 15.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 15, todos los bits de la palabra derecha del acumulador 1 se rellenarán con ceros.

Si el número de desplazamiento indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras":

```
SLW <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras":

```
SLW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda por palabras":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SLW 5	// Desplazar los bits 0 a 15 del acumulador 1 cinco posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SLW	// Desplazar los bits 0 a 15 del acumulador 1 cuatro posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	1010	0111	0110	0000
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	1101	0010	1011	0000

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

SRW: Desplazar a la derecha por palabras

Descripción

La instrucción "Desplazar a la derecha por palabras" desplaza bit por bit la palabra derecha (bits 0 a 15) del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con ceros. Los bits 16 a 31 del acumulador 1 no cambian.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 15.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 15, todos los bits de la palabra derecha del acumulador 1 se rellenarán con ceros.

Si el número de desplazamiento indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras":

```
SRW <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante la palabra derecha del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras":

```
SRW
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha por palabras":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SRW 6	// Desplazar los bits 0 a 15 del acumulador 1 seis posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SRW	// Desplazar los bits 0 a 15 del acumulador 1 cuatro posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0101	1111	0110	0100	0000	0001	0111	0100
Tag_Value_2	0101	1111	0110	0100	0101	1101	0010	1011
Tag_Result_2	0101	1111	0110	0100	0000	0101	1101	0010

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

SLD: Desplazar a la izquierda por palabras dobles

Descripción

La instrucción "Desplazar a la izquierda por palabras dobles" desplaza bit por bit todo el contenido del acumulador 1 hacia la izquierda. Las posiciones que quedan libres al desplazar se rellenan con ceros.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 31.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 31, todos los bits del acumulador 1 se rellenarán con ceros.

Si el número de desplazamiento indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras dobles":

```
SLD <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la izquierda por palabras dobles":

```
SLD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda por palabras dobles":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SLD 5	// Desplazar los bits 0 a 31 del acumulador 1 cinco posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SLD	// Desplazar los bits 0 a 31 del acumulador 1 cuatro posiciones a la izquierda.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1110	1100	1000	1011	1010	0111	0110	0000
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	1000	1010	0010	1001	1011	1100	1101	0000

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

SRD: Desplazar a la derecha por palabras dobles

Descripción

La instrucción "Desplazar a la derecha por palabras dobles" desplaza por bits todo el contenido del acumulador 1 hacia la derecha. Las posiciones que quedan libres al desplazar se rellenan con ceros.

Para indicar el número de posiciones de bit que se desplazarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción. El número indicado debe estar dentro de un rango de valores de 0 a 31.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de desplazamiento indicado es mayor que 31, todos los bits del acumulador 1 se rellenarán con ceros.

Si el número de desplazamiento indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de desplazamiento se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras dobles":

```
SRD <Número de desplazamiento>
```

Si el número de desplazamiento se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Desplazar a la derecha por palabras dobles":

```
SRD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha por palabras dobles":

Parámetro	Formato	Descripción
<Número de desplazamiento>	Enteros positivos	Número de posiciones de bit que se desplazarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
SRD 7	// Desplazar los bits 0 a 31 del acumulador 1 siete posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 4	// Cargar el número de desplazamiento en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de desplazamiento al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
SRD	// Desplazar los bits 0 a 31 del acumulador 1 cuatro posiciones a la derecha.
	// Rellenar las posiciones liberadas con ceros.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	0000	0000	1011	1110	1100	1000	1011	1010
Tag_Value_2	1010	1000	1010	0010	1001	1011	1100	1101
Tag_Result_2	0000	1010	1000	1010	0010	1001	1011	1100

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

Rotación

RLD: Rotar a la izquierda por palabras dobles

Descripción

La instrucción "Rotar a la izquierda por palabras dobles" rota bit por bit todo el contenido del acumulador 1 hacia la izquierda. Durante la ejecución, la instrucción desplaza por bits los bits de 0 a 31 del acumulador 1 hacia la izquierda y rellena las posiciones que han quedado libres al desplazar con las posiciones de bit desplazadas.

Para indicar el número de posiciones de bit que se rotarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de rotación indicado es 32, el contenido del acumulador 1 no cambia. Si el valor es mayor que 32, el número de rotación se calcula con una división módulo entre 32. Si, p. ej., el número de rotación es 34, los bits del acumulador 1 se rotarán 2 posiciones de bit.

Si el número de rotación indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de rotación se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Rotar a la izquierda por palabras dobles":

```
RLD <Número de rotación>
```

Si el número de rotación se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Rotar a la izquierda por palabras dobles":

```
RLD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda por palabras dobles":

Parámetro	Formato	Descripción
<Número de rotación>	Enteros positivos	Número de posiciones de bit que se rotarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
RLD 4	// Rotar los bits 0 a 31 del acumulador 1 cuatro posiciones a la izquierda.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 6	// Cargar el número de rotación en el acumulador 1.
L "Tag_Value_2"	// Desplazar el número de rotación al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
RLD	// Rotar los bits 0 a 31 del acumulador 1 seis posiciones a la izquierda.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	**0101**	1111	0110	0100	0101	1101	0011	1011
Tag_Result_1	1111	0110	0100	0101	1101	0011	1011	**0101**
Tag_Value_2	**1010**	**10**00	1010	0010	1001	1011	1100	1101
Tag_Result_2	0010	1000	1010	0110	1111	0011	01**10**	**1010**

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

RRD: Rotar a la derecha por palabras dobles

Descripción

La instrucción "Rotar a la derecha por palabras dobles" rota bit por bit todo el contenido del acumulador 1 hacia la derecha. Durante la ejecución, la instrucción desplaza por bits los bits 0 a 31 del acumulador 1 hacia la derecha y rellena las posiciones que han quedado libres al desplazar con las posiciones de bit desplazadas.

Para indicar el número de posiciones de bit que se rotarán existen las posibilidades siguientes:

- Indicación de un entero positivo como parámetro de la instrucción.
- Indicación sobre el valor del byte derecho del acumulador 2. El byte se interpreta como entero positivo.

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0" y el bit de estado CC 1 al estado lógico del último bit desplazado.

Si el número de rotación indicado es 32, el contenido del acumulador 1 no cambia. Si el valor es mayor que 32, el número de rotación se calcula con una división módulo entre 32. Si, p. ej., el número de rotación es 34, los bits del acumulador 1 se rotarán 2 posiciones de bit.

Si el número de rotación indicado es cero, la instrucción no se ejecuta. Los bits de estado no cambian.

Sintaxis

Si el número de rotación se indica como parámetro, se utiliza la sintaxis siguiente para la instrucción "Rotar a la derecha por palabras dobles":

```
RRD <Número de rotación>
```

Si el número de rotación se indica mediante el byte derecho del acumulador 2, se utiliza la sintaxis siguiente para la instrucción "Rotar a la derecha por palabras dobles":

```
RRD
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha por palabras dobles":

Parámetro	Formato	Descripción
<Número de rotación>	Enteros positivos	Número de posiciones de bit que se rotarán.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
RRD 4	// Rotar los bits 0 a 31 del acumulador 1 cuatro posiciones a la derecha.
T "Tag_Result_1"	// Transferir el contenido del acumulador 1 al operando.
L 6	// Cargar el número de rotación en el acumulador 1.

AWL	Explicación
L "Tag_Value_2"	// Desplazar el número de rotación al byte derecho del acumulador 2.
	// Cargar el valor del operando en el acumulador 1.
RRD	// Rotar los bits 0 a 31 del acumulador 1 seis posiciones a la derecha.
T "Tag_Result_2"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_Value_1	0101	1111	0110	0100	0101	1101	0011	**1011**
Tag_Result_1	**1011**	0101	1111	0110	0100	0101	1101	0011
Tag_Value_2	1010	1000	1010	0010	1001	1011	11**00**	**1101**
Tag_Result_2	**0011**	**01**10	1010	0010	1000	1010	0110	1111

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Vista general de los tipos de datos válidos (Página 2321)

RLDA: Rotar a la izquierda vía bit de estado CC 1

Descripción

La instrucción "Rotar a la izquierda vía bit de estado CC 1" desplaza 1 bit a la izquierda el contenido del acumulador 1. La posición de bit que ha quedado libre al desplazar (0) se rellena con el estado lógico del bit de estado CC 1. El bit de estado CC 1 obtiene el estado lógico del bit desplazado (31).

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0".

Sintaxis

Para la instrucción "Rotar a la izquierda vía bit de estado CC 1" se utiliza la siguiente sintaxis:

```
RLDA
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
RLDA	// Rotar los bits 0 a 31 del acumulador 1 una posición a la izquierda.
	// Rellenar el bit 1 con el estado lógico del bit de estado CC 1.
	// Escribir el bit desplazado en el bit de estado CC 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	CC 1	Valor							
Tag_Value	1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	0	1011	1110	1100	1000	1011	1010	0111	0111

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

RRDA: Rotar a la derecha vía bit de estado CC 1

Descripción

La instrucción "Rotar a la derecha vía bit de estado CC 1" desplaza el contenido del acumulador 1 1 bit a la derecha. La posición de bit que ha quedado libre al desplazar (31) se rellena con el estado lógico del bit de estado CC 1. El bit de estado CC 1 obtiene el estado lógico del bit desplazado (0).

Esta instrucción se ejecuta independientemente del RLO. Pone el bit de estado CC 0 a "0".

Sintaxis

Para la instrucción "Rotar a la derecha vía bit de estado CC 1" se utiliza la siguiente sintaxis:

```
RRDA
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value"	// Cargar el valor del operando en el acumulador 1.
RRDA	// Rotar los bits 0 a 31 del acumulador 1 una posición a la derecha.
	// Rellenar el bit 31 con el estado lógico del bit de estado CC 1.
	// Escribir el bit desplazado en el bit de estado CC 1.
T "Tag_Result"	// Transferir el contenido del acumulador 1 al operando.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	CC 1	Valor							
Tag_Value	1	0101	1111	0110	0100	0101	1101	0011	1011
Tag_Result	0	1010	1111	1011	0010	0010	1110	1001	1101

El bit de estado CC 1 tiene el valor "1", ya que el bit 0 del operando "Tag_Value" también tiene el valor "1".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de desplazamiento y rotación (Página 2555)

Principios básicos de la palabra de estado (Página 2545)

Otras instrucciones

Acumulador

TAK: Intercambiar contenido de los acumuladores 1 y 2

Descripción

La instrucción "Intercambiar contenido de los acumuladores 1 y 2" intercambia el contenido del acumulador 1 con el contenido del acumulador 2. En los controladores con cuatro acumuladores, los contenidos de los acumuladores 3 y 4 no se ven influidos por esta instrucción.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Intercambiar contenido de los acumuladores 1 y 2" se utiliza la siguiente sintaxis:

```
TAK
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
JC NEXT	// Si la condición se cumple (RLO = "1"), saltar a la etiqueta "NEXT".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
TAK	// Intercambiar contenido de los acumuladores 1 y 2.
NEXT: -I	// Etiqueta "NEXT"
	// Restar el valor del acumulador 1 del valor del acumulador 2.
T "Tag_Output"	// Transferir el contenido del acumulador 1 al operando "Tag_Output".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

PUSH: Desplazar contenido al siguiente acumulador superior

Descripción

La instrucción "Desplazar contenido al siguiente acumulador superior" desplaza los contenidos de los acumuladores 1 a 3 al siguiente acumulador superior en cada caso.

En los controladores con 2 acumuladores solo se copia el contenido del acumulador 1 en el acumulador 2.

En los controladores con 4 acumuladores se desplaza el contenido del acumulador 1 al acumulador 2, el contenido del acumulador 2 al acumulador 3 y el contenido del acumulador 3 al acumulador 4. El contenido del acumulador 4 se pierde.

El contenido del acumulador 1 no se ve influido por la instrucción y permanece invariable una vez ejecutada la misma.

La tabla siguiente muestra el contenido de los acumuladores 1 a 4 antes y después de ejecutar la instrucción:

Estado	Acumulador			
	1	2	3	4
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor A	Valor B	Valor C

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Desplazar contenido al siguiente acumulador superior" se utiliza la siguiente sintaxis:

```
PUSH
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando en el acumulador 1.
PUSH	// Desplazar el contenido de los acumuladores 1 a 3 al siguiente acumulador superior en cada caso.
L "Tag_Value_2"	// Cargar el valor del operando en el acumulador 1.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

POP: Desplazar contenido al siguiente acumulador inferior

Descripción

La instrucción "Desplazar contenido al siguiente acumulador inferior" desplaza los contenidos de los acumuladores 4 a 2 al siguiente acumulador inferior en cada caso.

En los controladores con 2 acumuladores se copia el contenido del acumulador 2 en el acumulador 1.

En los controladores con 4 acumuladores se desplaza el contenido del acumulador 4 al acumulador 3, el contenido del acumulador 3 al acumulador 2 y el contenido del acumulador 2 al acumulador 1. El contenido del acumulador 1 se pierde.

El contenido del acumulador 4 no se ve influido por la instrucción y permanece invariable una vez ejecutada la misma.

La tabla siguiente muestra el contenido de los acumuladores 1 a 4 antes y después de ejecutar la instrucción:

Estado	Acumulador			
	1	2	3	4
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor B	Valor C	Valor D	Valor D

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Desplazar contenido al siguiente acumulador inferior" se utiliza la siguiente sintaxis:

```
POP
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`T "Tag_Value_1"`	`// Transferir el contenido del acumulador 1 al operando "Tag_Value_1".`
`POP`	`// Desplazar el contenido de los acumuladores 4 a 2 al siguiente acumulador inferior en cada caso.`
`T "Tag_Value_2"`	`// Transferir el contenido del acumulador 1 al operando "Tag_Value_2".`

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

ENT: Desplazar los acumuladores 2 y 3 al siguiente acumulador superior

Descripción

La instrucción "Desplazar los acumuladores 2 y 3 al siguiente acumulador superior" desplaza el contenido del acumulador 3 al acumulador 4 y el contenido del acumulador 2 al acumulador 3. Los contenidos de los acumuladores 1 y 2 no cambian.

La tabla siguiente muestra el contenido de los acumuladores 1 a 4 antes y después de ejecutar la instrucción:

Estado	Acumulador			
	1	2	3	4
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor B	Valor B	Valor C

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Desplazar los acumuladores 2 y 3 al siguiente acumulador superior" se utiliza la siguiente sintaxis:

```
ENT
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
*R	// Multiplicar los valores de los acumuladores 1 y 2.
	// Guardar el producto (resultado parcial) en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 (resultado parcial) al acumulador 2.

AWL	Explicación
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
ENT	// Copiar el contenido del acumulador 2 (resultado parcial) en el acumulador 3.
	// Copiar el contenido del acumulador 3 en el acumulador 4.
L "Tag_Value_4"	// Desplazar el contenido del acumulador 1 ("Tag_Value_3") al acumulador 2.
	// Cargar el valor del operando "Tag_Value_4" en el acumulador 1.
+R	// Sumar los valores de los acumuladores 1 y 2.
	// Guardar la suma en el acumulador 1.
	// Transferir el valor del acumulador 3 (resultado parcial) al acumulador 2.
LEAVE	// Desplazar el contenido del acumulador 3 (resultado parcial) al acumulador 2.
	// Desplazar el contenido del acumulador 4 al acumulador 3.
/R	// Dividir los valores de los acumuladores 1 y 2.
	// Guardar el resultado en el acumulador 1.
T "Tag_Result"	// Transferir el resultado (contenido del acumulador 1) al operando "Tag_Result".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

LEAVE: Desplazar los acumuladores 3 y 4 al siguiente acumulador inferior

Descripción

La instrucción "Desplazar los acumuladores 3 y 4 al siguiente acumulador inferior" desplaza el contenido del acumulador 3 al acumulador 2 y el contenido del acumulador 4 al acumulador 3. Los contenidos de los acumuladores 1 y 4 no cambian.

Las instrucciones de cálculos aritméticos incluyen la funcionalidad de la instrucción "Desplazar los acumuladores 3 y 4 al siguiente acumulador inferior".

La tabla siguiente muestra el contenido de los acumuladores 1 a 4 antes y después de ejecutar la instrucción:

Estado	Acumulador			
	1	2	3	4
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor C	Valor D	Valor D

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Sintaxis

Para la instrucción "Desplazar los acumuladores 3 y 4 al siguiente acumulador inferior" se utiliza la siguiente sintaxis:

```
LEAVE
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
AD	// Combinar lógicamente con Y el valor del acumulador 2 ("Tag_Value_1") con el valor del acumulador 1 ("Tag_Value_2").
	// Guardar el resultado en el acumulador 1.
L "Tag_Value_3"	// Desplazar el contenido del acumulador 1 (resultado parcial) al acumulador 2.
	// Cargar el valor del operando "Tag_Value_3" en el acumulador 1.
ENT	// Copiar el contenido del acumulador 2 (resultado parcial) en el acumulador 3.
	// Copiar el contenido del acumulador 3 en el acumulador 4.
L "Tag_Value_4"	// Desplazar el contenido del acumulador 1 ("Tag_Value_3") al acumulador 2.
	// Cargar el valor del operando "Tag_Value_4" en el acumulador 1.
AD	// Combinar lógicamente con Y el valor del acumulador 2 ("Tag_Value_3") con el valor del acumulador 1 ("Tag_Value_4").
	// Guardar el resultado en el acumulador 1.
LEAVE	// Desplazar el contenido del acumulador 3 (resultado parcial) al acumulador 2.
	// Desplazar el contenido del acumulador 4 al acumulador 3.
OD	// Combinar lógicamente con O el valor del acumulador 2 con el valor del acumulador 1.
	// Guardar el resultado en el acumulador 1.

AWL	Explicación
T "Tag_Result"	// Transferir el resultado al operando "Tag_Result".

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Registro de direcciones

+AR1: Sumar acumulador 1 a AR1

Descripción

La instrucción "Sumar acumulador 1 a AR1" suma un valor al contenido del registro de direcciones 1. El tipo de puntero que está en el registro de direcciones 1 y el área de operandos se conservan.

Para indicar el valor que debe sumarse existen las posibilidades siguientes:

- Indicación mediante una constante: la instrucción suma el valor de la constante al registro de direcciones 1. El valor de la constante debe equivaler al formato de un puntero interno del área (POINTER).
- Indicación mediante el valor de la palabra derecha del acumulador 1: la instrucción interpreta el valor de la palabra derecha del acumulador 1 como un entero de 16 bits y lo amplía a 24 bits con el signo correcto. Seguidamente, la instrucción suma el valor del acumulador 1 al registro de direcciones 1. Se admiten valores de -32 768 a +32 767.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Sintaxis

Si el valor de la suma se indica mediante una constante, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR1":

```
+AR1 <constante>
```

Si el valor de la suma se indica mediante la palabra derecha del acumulador 1, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR1":

```
+AR1
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar acumulador 1 a AR1":

Parámetro	Tipo de datos	Descripción
<constante>	POINTER	Valor que se suma al registro de direcciones 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
`+AR1 P#10.0`	`// Sumar el puntero al registro de direcciones 1.`
`L "Tag_Value"`	`// Cargar el valor del operando "Tag_Value" en la palabra derecha del acumulador 1.`
`+AR1`	`// Sumar el valor del acumulador 1 al registro de direcciones 1.`
`TAR1 %MD24`	`// Transferir el contenido del registro de direcciones 1 a la palabra doble MD24.`

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

+AR2: Sumar acumulador 1 a AR2

Descripción

La instrucción "Sumar acumulador 1 a AR2" suma un valor al contenido del registro de direcciones 2. El tipo de puntero que está en el registro de direcciones 2 y el área de operandos se conservan.

Para indicar el valor que debe sumarse existen las posibilidades siguientes:

- Indicación mediante una constante: la instrucción suma el valor de la constante al registro de direcciones 2. El valor de la constante debe equivaler al formato de un puntero interno del área (POINTER).
- Indicación mediante el valor de la palabra derecha del acumulador 1: la instrucción interpreta el valor de la palabra derecha del acumulador 1 como un entero de 16 bits y lo amplía a 24 bits con el signo correcto. Seguidamente, la instrucción suma el valor del acumulador 1 al registro de direcciones 2. Se admiten valores de -32 768 a +32 767.

La CPU ejecuta la instrucción independientemente del resultado lógico y los bits de estado. La instrucción no influye ni en el resultado lógico ni en los bits de estado.

Los contenidos de los acumuladores no cambian debido a la instrucción.

Nota

El registro de direcciones AR2 se utiiza para procesar multiinstancias. Si se programa la instrucción "Sumar acumulador 1 a AR2", primero debe "salvarse" el contenido del registro de direcciones 2 y volver a cargarlo después.

Sintaxis

Si el valor de la suma se indica mediante una constante, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR2":

```
+AR2 <constante>
```

Si el valor de la suma se indica mediante la palabra derecha del acumulador 1, se utiliza la sintaxis siguiente para la instrucción "Sumar acumulador 1 a AR2":

```
+AR2
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Sumar acumulador 1 a AR2":

Parámetro	Tipo de datos	Descripción
<constante>	POINTER	Valor que se suma al registro de direcciones 1.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
+AR2 P#10.0	// Sumar el puntero al registro de direcciones 2.
L "Tag_Value"	// Cargar el valor del operando "Tag_Value" en la palabra derecha del acumulador 1.
+AR2	// Sumar el valor del acumulador 1 al registro de direcciones 2.
TAR2 %MD24	// Transferir el contenido del registro de direcciones 1 a la palabra doble MD24.

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Master Control Relay

MCR(: Abrir áreas MCR

Descripción

La instrucción "Abrir áreas MCR" activa y desactiva la dependencia MCR dentro de un área MCR en función del resultado lógico (RLO).

El principio y el fin de un área MCR se definen mediante las instrucciones "Activar área MCR" (MCRA) y "Desactivar área MCR" (MCRD). Dentro del área MCR definida es posible activar y desactivar la dependencia MCR para una o varias partes del programa.

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

La dependencia MCR se activa si el RLO es "1" antes de procesar la instrucción. Si el RLO es "0" antes de la instrucción, la dependencia MCR se desactiva. El final de la sección del programa dependiente de MCR se programa mediante la instrucción "Cerrar áreas MCR". La sección del programa que se encuentra entre las dos instrucciones se ve influenciada por el MCR. Solo se ven afectadas las instrucciones dependientes de MCR (p. ej. la instrucción "Transferir" (T)).

También es posible anidar la activación y desactivación de la dependencia MCR programando la instrucción "Abrir áreas MCR" en una sección del programa dependiente de MCR. Esta instrucción se puede anidar hasta 8 niveles de profundidad. La dependencia MCR de la parte del programa de nivel superior controla las dependencias MCR en todas las partes del programa subordinadas. Si la dependencia MCR en la parte del programa de nivel superior está desactivada, no es posible activarla en una parte subordinada del programa dependiente de MCR.

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

Las instrucciones "Abrir áreas MCR" y "Cerrar áreas MCR" deben programarse siempre de dos en dos. Si una de las dos instrucciones no está programada, se dispara un error de pila MCR.

Sintaxis

La instrucción "Abrir áreas MCR" se programa siguiendo el siguiente esquema general:

```
MCRA
MCR(
....
)MCR
```

MCRD

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
MCRA	// Principio del área MCR
A "Tag_Input_1"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
MCR(	// Principio de la dependencia MCR
	// Si el estado lógico del RLO es "1", activar la dependencia MCR.
	// Si el estado lógico del RLO es "0", desactivar la dependencia MCR.
A "Tag_Input_2"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_1"	// Si la dependencia MCR está desactivada, poner el operando a "0".
	// Si la dependencia MCR está activada, poner el operando a "1" o "0" en función del RLO.
A "Tag_Input_3"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
MCR(	// Anidamiento de la dependencia MCR
A "Tag_Input_4"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
A "Tag_Input_5"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_2"	// Si la dependencia MCR está desactivada, poner el operando a "0".
	// Si la dependencia MCR está activada, poner el operando a "1" o "0" en función del RLO.
)MCR	// Fin del anidamiento
L "Tag_InputValue_1"	// Cargar el valor del operando en el acumulador 1.
T "Tag_OutputValue_1"	// Si la dependencia MCR está desactivada, escribir el valor "0" en el operando.
	// Si la dependencia MCR está activada, transferir el contenido del acumulador 1 al operando.
)MCR	// Fin de la dependencia MCR
A "Tag_Input_6"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
MCR(	// Principio de la dependencia MCR
	// Si el estado lógico del RLO es "1", activar la dependencia MCR.
	// Si el estado lógico del RLO es "0", desactivar la dependencia MCR.
L "Tag_InputValue_2"	// Cargar el valor del operando en el acumulador 1.

AWL	Explicación
T "Tag_OutputValue_2"	// Si la dependencia MCR está desactivada, escribir el valor "0" en el operando.
	// Si la dependencia MCR está activada, transferir el contenido del acumulador 1 al operando.
)MCR	// Fin de la dependencia MCR
MCRD	// Fin del área MCR
A "Tag_Input_7"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_3"	// Poner el operando a "1" o "0" en función del RLO.

Consulte también

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

)MCR: Cerrar áreas MCR

Descripción

La instrucción "Cerrar áreas MCR" cierra una sección del programa dependiente de MCR. También borra el RLO de la última entrada de la pila MCR.

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

La dependencia MCR se programa con la instrucción "Abrir áreas MCR". Las instrucciones "Abrir áreas MCR" y "Cerrar áreas MCR" deben programarse siempre de dos en dos. Si una de las dos instrucciones no está programada, se dispara un error de pila MCR.

Sintaxis

La instrucción "Cerrar áreas MCR" se programa siguiendo el siguiente esquema general:

```
MCRA          // Principio del área MCR
MCR(          // Principio de la dependencia MCR
.....         // Cualquier programa
)MCR          // Fin de la dependencia MCR
MCRD          // Fin del área MCR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
MCRA	// Principio del área MCR
A "Tag_Input_1"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
MCR(	// Principio de la dependencia MCR
	// Si el estado lógico del RLO es "1", activar la dependencia MCR.
	// Si el estado lógico del RLO es "0", desactivar la dependencia MCR.
A "Tag_Input_2"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_1"	// Si la dependencia MCR está desactivada, poner el operando a "0".
	// Si la dependencia MCR está activada, poner el operando a "1" o "0" en función del RLO.
L "Tag_InputValue"	// Cargar el valor del operando en el acumulador 1.
T "Tag_OutputValue"	// Si la dependencia MCR está desactivada, escribir el valor "0" en el operando.
	// Si la dependencia MCR está activada, transferir el contenido del acumulador 1 al operando.
)MCR	// Fin de la dependencia MCR
MCRD	// Fin del área MCR
A "Tag_Input_3"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_2"	// Poner el operando a "1" o "0" en función del RLO.

Consulte también

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

MCRA: Activar área MCR

Descripción

La instrucción "Activar área MCR" define el principio de un área MCR. Dentro de dicha área es posible activar y desactivar la dependencia MCR con la ayuda de las instrucciones siguientes:

- MCR(: Abrir áreas MCR
-)MCR: Cerrar áreas MCR

El fin del área MCR se define con la instrucción "Desactivar área MCR" (MCRD). Las instrucciones para definir el principio y fin de un área MCR siempre deben utilizarse por parejas.

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

La instrucción se ejecuta sin considerar ni influir en los bits de estado.

Sintaxis

La instrucción "Activar área MCR" se programa siguiendo el siguiente esquema general:

```
MCRA          // Principio del área MCR
MCR(          // Principio de la dependencia MCR
.....         // Cualquier programa
)MCR          // Fin de la dependencia MCR
MCRD          // Fin del área MCR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
MCRA	// Principio del área MCR
A "Tag_Input_1"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
MCR(	// Principio de la dependencia MCR
	// Si el estado lógico del RLO es "1", activar la dependencia MCR.
	// Si el estado lógico del RLO es "0", desactivar la dependencia MCR.

AWL	Explicación
A "Tag_Input_2"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_1"	// Si la dependencia MCR está desactivada, poner el operando a "0".
	// Si la dependencia MCR está activada, poner el operando a "1" o "0" en función del RLO.
L "Tag_InputValue"	// Cargar el valor del operando en el acumulador 1.
T "Tag_OutputValue"	// Si la dependencia MCR está desactivada, escribir el valor "0" en el operando.
	// Si la dependencia MCR está activada, transferir el contenido del acumulador 1 al operando.
)MCR	// Fin de la dependencia MCR
MCRD	// Fin del área MCR
A "Tag_Input_3"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_2"	// Poner el operando a "1" o "0" en función del RLO.

Consulte también

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

MCRD: Desactivar área MCR

Descripción

La instrucción "Desactivar área MCR" define el final de un área MCR.

El principio de un área MCR se define con la instrucción "Activar área MCR" (MCRA). Las instrucciones para definir el principio y fin de un área MCR siempre deben utilizarse por parejas.

Nota

Observe la información contenida en el apartado "Indicaciones importantes sobre el uso de la funcionalidad MCR", en "Consulte también".

Dentro de un área MCR es posible programar la dependencia MCR con la ayuda de las instrucciones siguientes:

- MCR(: Abrir áreas MCR
-)MCR: Cerrar áreas MCR

Si dentro de un área MCR se llama un bloque, la dependencia MCR está desactivada en el bloque llamado. Cuando se sale de un bloque, la dependencia MCR se ajusta igual que estaba antes de llamarlo.

La instrucción se ejecuta sin considerar ni influir en los bits de estado.

Sintaxis

La instrucción "Desactivar área MCR" se programa siguiendo el siguiente esquema general:

```
MCRA          // Principio del área MCR
MCR(          // Principio de la dependencia MCR
.....         // Cualquier programa
)MCR          // Fin de la dependencia MCR
MCRD          // Fin del área MCR
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
MCRA	// Principio del área MCR
A "Tag_Input_1"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
MCR(	// Principio de la dependencia MCR
	// Si el estado lógico del RLO es "1", activar la dependencia MCR.
	// Si el estado lógico del RLO es "0", desactivar la dependencia MCR.
A "Tag_Input_2"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_1"	// Si la dependencia MCR está desactivada, poner el operando a "0".
	// Si la dependencia MCR está activada, poner el operando a "1" o "0" en función del RLO.
L "Tag_InputValue"	// Cargar el valor del operando en el acumulador 1.
T "Tag_OutputValue"	// Si la dependencia MCR está desactivada, escribir el valor "0" en el operando.
	// Si la dependencia MCR está activada, transferir el contenido del acumulador 1 al operando.
)MCR	// Fin de la dependencia MCR
MCRD	// Fin del área MCR
A "Tag_Input_3"	// Consultar si el operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
= "Tag_Output_2"	// Poner el operando a "1" o "0" en función del RLO.

Consulte también

Indicaciones importantes sobre el uso de la función MCR (Página 2565)

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

Activar el bit de estado en instrucciones de control del programa (Página 2553)

Principios básicos de la palabra de estado (Página 2545)

Instrucciones nulas

BLD: Refresco de imagen (instrucción nula)

Descripción

La instrucción "Refresco de imagen (instrucción nula)" no ejecuta ninguna función y no influye en los bits de estado. La instrucción sirve para reconocer secuencias de códigos en la transferencia de parámetros o en los segmentos KOP/FUP. Se genera automáticamente cuando se visualiza un programa KOP o FUP en AWL. El valor de parámetro es el número de identificación de la instrucción y es generado por la programadora.

Sintaxis

Para la instrucción "Refresco de imagen (instrucción nula)" se utiliza la siguiente sintaxis:

```
BLD <número de identificación>
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Refresco de imagen (instrucción nula)":

Parámetro	Tipo de datos	Descripción
<número de identificación>	WORD	Número de identificación de la instrucción

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

NOP 0: Instrucción nula

Descripción

La instrucción "Instrucción nula" con un parámetro 0 no ejecuta ninguna función y no influye en los bits de estado. El código de la instrucción contiene un patrón de bits con 16 ceros. La instrucción solo es importante para la programadora cuando se visualiza un programa.

Sintaxis

Para la instrucción "Instrucción nula" con un parámetro 0 se utiliza la siguiente sintaxis:

```
NOP 0
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

AWL	Explicación
L "Tag_Value_1"	// Cargar el valor del operando "Tag_Value_1" en el acumulador 1.
L "Tag_Value_2"	// Desplazar el contenido del acumulador 1 al acumulador 2.
	// Cargar el valor del operando "Tag_Value_2" en el acumulador 1.
>I	// Comparar si el valor del acumulador 2 es mayor que el valor del acumulador 1.
A	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual.
JC NEXT	// Si la condición se cumple (RLO = "1"), saltar a la etiqueta "NEXT".
	// Si la condición no se cumple (RLO = "0"), procesar la instrucción siguiente.
TAK	// Intercambiar contenido de los acumuladores 1 y 2.
NEXT: NOP 0	// Etiqueta "NEXT"

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

NOP 1: Instrucción nula

Descripción

La instrucción "Instrucción nula" con un parámetro 1 no ejecuta ninguna función y no influye en los bits de estado. El código de la instrucción contiene un patrón de bits con 16 unos. La instrucción solo es importante para la programadora cuando se visualiza un programa.

Sintaxis

Para la instrucción "Instrucción nula" con un parámetro 1 se utiliza la siguiente sintaxis:

```
NOP 1
```

Consulte también

Añadir instrucciones AWL (Página 2767)

Procesar instrucciones AWL (Página 2782)

SCL

Temporizadores

Temporizadores CEI

TP: Impulso

Descripción

La instrucción "Impulso" permite activar el parámetro Q durante el tiempo PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contar al iniciar la instrucción. El parámetro Q se activa por el tiempo PT, independientemente de cómo evolucione la señal en el parámetro IN. La detección de un nuevo flanco de señal ascendente tampoco influirá en el estado lógico del parámetro Q mientras transcurra el tiempo PT.

El valor actual de tiempo se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Si cuando se alcanza el tiempo PT el estado lógico del parámetro IN es "0", se resetea el parámetro ET.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guarden los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TP (p. ej. "TP_DB")
- Declaración como variable local del tipo TP en la sección "Static" de un bloque (p. ej., #MyTP_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en "Bloques de programa > Bloques de sistema" de la carpeta "Recursos de programa", en el árbol de proyecto. Encontrará más información al respecto en "Consulte también".

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede a las salidas Q o ET.

El sistema operativo resetea las instancias de la instrucción "Impulso" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Impulso", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Impulso" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TP(IN := <Operando>,
                 PT := <Operando>,
                 Q => <Operando>,
                 ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

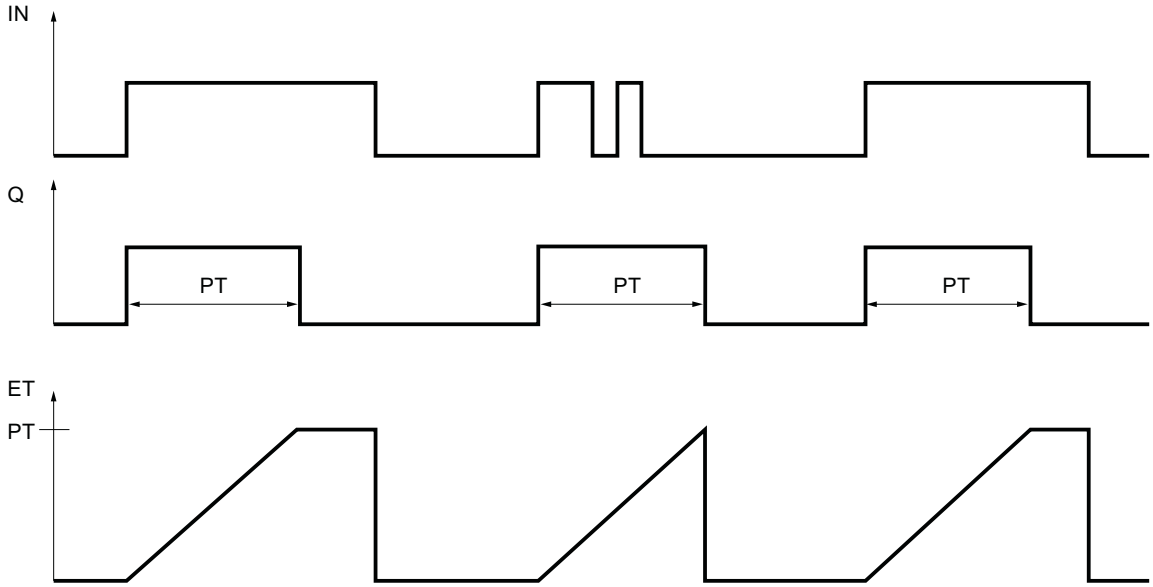
Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	BOOL	Entrada de arranque
PT	Input	TIME	Duración del impulso. El valor del parámetro PT debe ser positivo.

Parámetro	Declaración	Tipo de datos	Descripción
Q	Output	BOOL	Operando que se activa por el tiempo PT.
ET	Output	TIME	Valor actual de tiempo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TP_DB".TP(IN := "Tag_Start",
           PT := "Tag_PresetTime",
           Q => "Tag_Status",
           ET => "Tag_ElapsedTime");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar y el operando "Tag_Status" pasa a "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TON: Retardo al conectar

Descripción

La instrucción "Retardo al conectar" permite retardar la activación del parámetro Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT deja de contar al iniciar la instrucción. Una vez transcurrido el tiempo PT, el parámetro Q devuelve el estado lógico "1". El parámetro Q permanecerá activado mientras la entrada de arranque esté puesta a "1". Cuando el estado lógico del parámetro IN cambia de "1" a "0", se resetea el parámetro Q. La función de temporización se reinicia al detectarse un nuevo flanco ascendente en el parámetro IN.

El valor actual de tiempo se puede consultar en el parámetro ET. Este valor de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor del tiempo PT. El parámetro ET se resetea en cuanto el estado lógico del parámetro IN cambia a "0".

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guarden los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TON (p. ej. "TON_DB")
- Declaración como variable local del tipo TON en la sección "Static" de un bloque (p. ej., #MyTON_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en "Bloques de programa > Bloques de sistema" de la carpeta "Recursos de programa", en el árbol de proyecto. Encontrará más información al respecto en "Consulte también".

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede a las salidas Q o ET.

El sistema operativo resetea las instancias de la instrucción "Retardo al conectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al conectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Retardo al conectar" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TON(IN := <Operando>,
                  PT := <Operando>,
                  Q => <Operando>,
                  ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

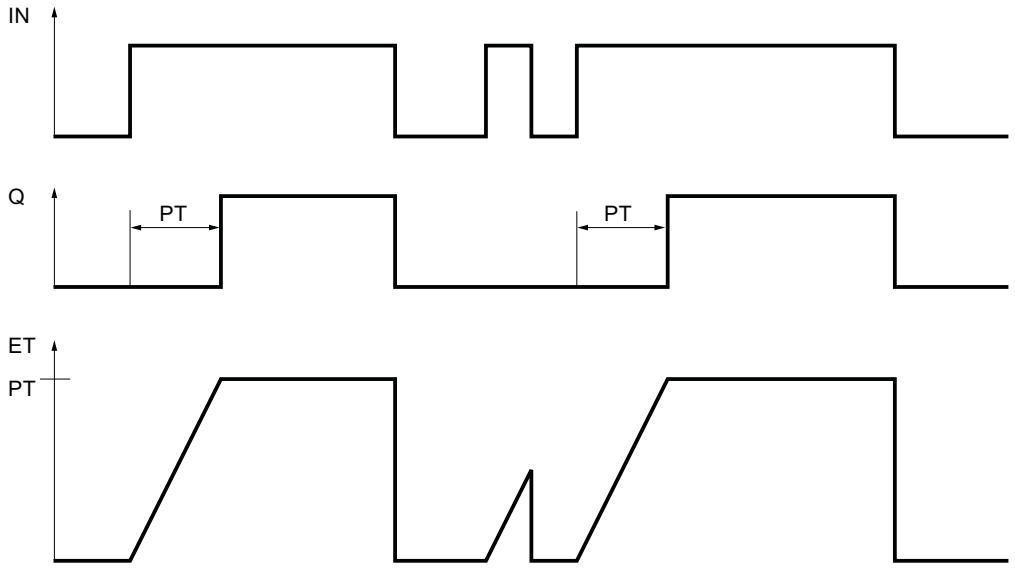
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	BOOL	Entrada de arranque
PT	Input	TIME	Duración del retardo al conectar. El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	Operando que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	Valor actual de tiempo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TON_DB".TON(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar. Una vez transcurrido el tiempo, el operando "Tag_Start" se pone a "1" si el operando "Tag_Status" tiene el estado lógico "1". Una vez transcurrido el tiempo, el operando "Tag_Status" adopta el estado lógico "1" mientras el operando "Tag_Start" tenga el estado lógico "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

TOF: Retardo al desconectar

Descripción

La instrucción "Retardo al desconectar" permite retardar el reset del parámetro Q por el tiempo programado PT. El parámetro Q se establece cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico del parámetro IN cambia nuevamente a "0", el tiempo programado PT deja de contar. El parámetro Q permanecerá activado mientras transcurra el tiempo PT. Una vez transcurrido el tiempo PT se resetea el parámetro Q. Si el estado lógico del parámetro IN cambia a "1" antes de que transcurra el tiempo PT, se resetea el tiempo. El estado lógico del parámetro Q permanece a "1".

El valor actual de tiempo se puede consultar en el parámetro ET. Este valor actual de tiempo empieza a contar a partir de T#0s y termina al alcanzarse el valor de tiempo PT. Una vez transcurrido el tiempo PT el parámetro ET permanecerá al valor actual hasta que el parámetro IN vuelva a cambiar a "1". Si el parámetro IN cambia a "1" antes de transcurrir el tiempo PT, el parámetro ET se restablece al valor T#0s.

Nota

Si en el programa no se llama el temporizador, p. ej. porque se ignora, la salida ET devuelve un valor de constante en cuanto haya transcurrido el tiempo correspondiente.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guarden los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TOF (p. ej. "TOF_DB")
- Declaración como variable local del tipo TOF en la sección "Static" de un bloque (p. ej., #MyTOF_TIMER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el temporizador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en "Bloques de programa > Bloques de sistema" de la carpeta "Recursos de programa", en el árbol de proyecto. Encontrará más información al respecto en "Consulte también".

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede a las salidas Q o ET.

El sistema operativo inicializa las instancias de la instrucción "Retardo al desconectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras

un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al desconectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Retardo al desconectar" se utiliza la siguiente sintaxis:

- Bloque de datos del tipo de datos de sistema IEC_TIMER (Shared DB):

SCL

```
<DB_Temporizador_CEI>.TOF(IN := <Operando>,
                  PT := <Operando>,
                  Q => <Operando>,
                  ET => <Operando>)
```

- Variable local:

SCL

```
#myLocal_timer(IN := <Operando>,
               PT := <Operando>,
               Q => <Operando>,
               ET => <Operando>)
```

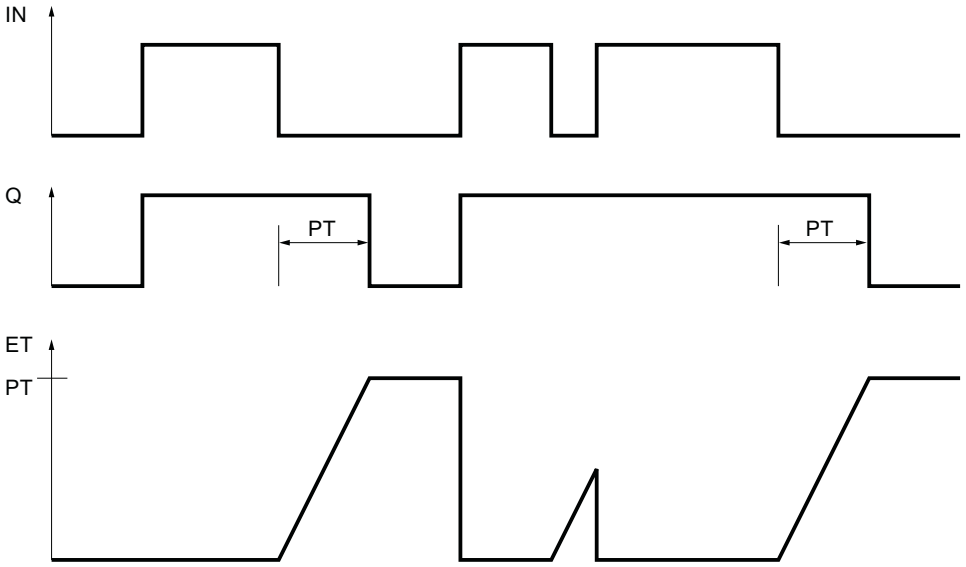
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	BOOL	Entrada de arranque
PT	Input	TIME	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	Operando que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	Valor actual de tiempo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"TOF_DB".TOF(IN := "Tag_Start",
             PT := "Tag_PresetTime",
             Q => "Tag_Status",
             ET => "Tag_ElapsedTime");
```

Si se produce un cambio del estado lógico del operando "Tag_Start" de "0" a "1", el operando se pone a "Tag_Status". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", el tiempo programado en el parámetro PT empieza a contar. Mientras transcurra el tiempo, permanecerá activado el operando "Tag_Status". Una vez transcurrido el tiempo, el operando "Tag_Status" se desactiva. El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

Temporizadores SIMATIC

S_PULSE: Parametrizar y arrancar temporizador como impulso

Descripción

La instrucción "Parametrizar y arrancar temporizador como impulso" arranca el temporizador programado en el parámetro T_NO cuando se detecta un cambio del resultado lógico (RLO) de "0" a "1" (flanco de señal ascendente) en el parámetro S. El temporizador cuenta el tiempo programado TV mientras el estado lógico del parámetro S sea "1".

Si el estado lógico del parámetro S cambia a "0" antes de que transcurra el tiempo programado, el temporizador se detiene y el parámetro "Q" adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

Si el temporizador está contando y el estado lógico de la entrada R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está en marcha, el estado lógico "1" de la entrada R no provoca ningún efecto.

El parámetro Q devuelve el estado lógico "1" mientras el temporizador está contando y el estado lógico del parámetro S es "1". Si el estado lógico del parámetro S cambia a "0" antes de que transcurra el tiempo programado, el parámetro Q devuelve el estado lógico "0". Si el parámetro R resetea el temporizador o si el tiempo ha transcurrido, el parámetro Q también devuelve el estado lógico "0".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como impulso" se utiliza la sintaxis siguiente:

SCL

```
S_PULSE(T_NO := <Operando>,
        S := <Operando>,
```

SCL

```
        TV := <Operando>,
        R := <Operando>,
        Q => <Operando>,
        BI => <Operando>)
```

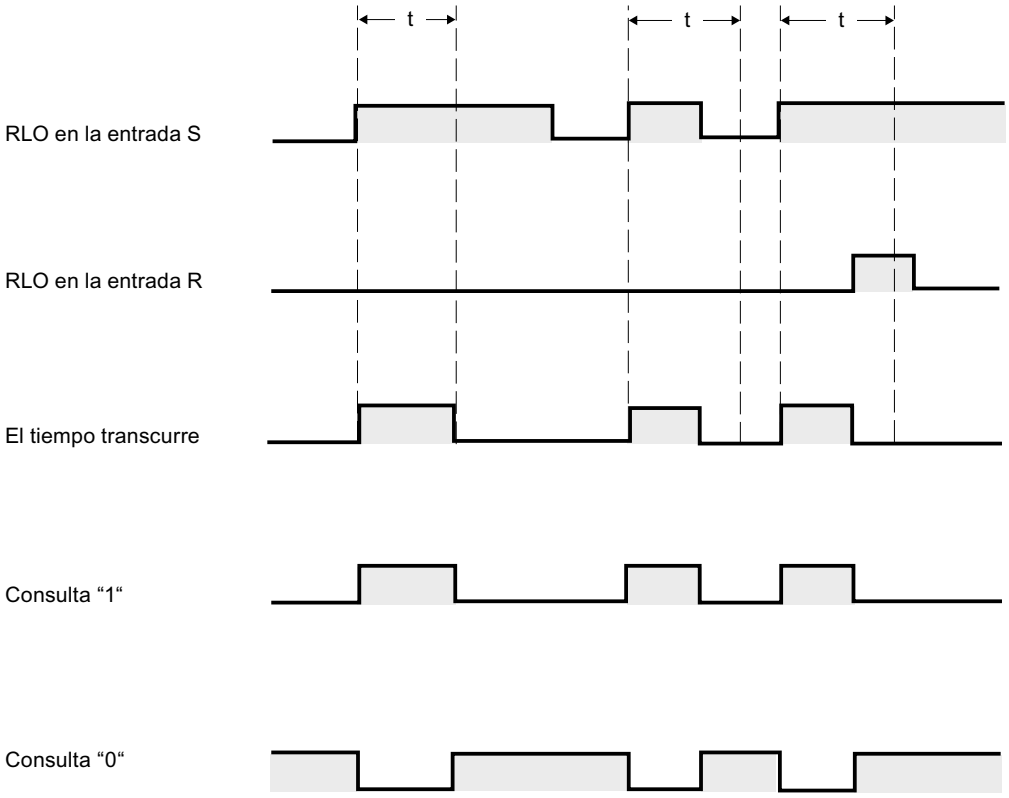
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_PULSE(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q := "Tag_Status",
                        BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador empieza a contar con el valor de tiempo del operando "Tag_Number" y sigue contando mientras el operando "Tag_1" devuelva el estado lógico "1".

Si el estado lógico del parámetro S cambia a "0" antes de que transcurra el tiempo programado, el operando "Tag_Status" adopta el estado lógico "0". Si el parámetro R resetea el

temporizador o si el tiempo ha transcurrido, el operando "Tag_Status" también devuelve el estado lógico "0".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 6189)

S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado

Descripción

La instrucción "Parametrizar y arrancar temporizador como impulso prolongado" arranca el temporizador programado cuando se detecta un flanco de señal ascendente en el parámetroS. El temporizador deja de contar el tiempo programado TV aunque el estado lógico del parámetro S cambie a "0". Mientras el temporizador está contando, el parámetro Q devuelve el estado lógico "1".

Una vez transcurrido el tiempo, el parámetro Q adopta el valor lógico "0". Si el estado lógico del parámetro S cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo programado en el parámetro TV.

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

Si el temporizador está contando y el estado lógico del parámetro R cambia a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. Si el temporizador no está contando, el estado lógico "1" en el parámetro R no tiene ningún efecto.

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como impulso prolongado" se utiliza la sintaxis siguiente:

SCL

```
S_PEXT(T_NO := <Operando>,
       S := <Operando>,
       TV := <Operando>,
       R := <Operando>,
       Q => <Operando>,
       BI =><Operando>)
```

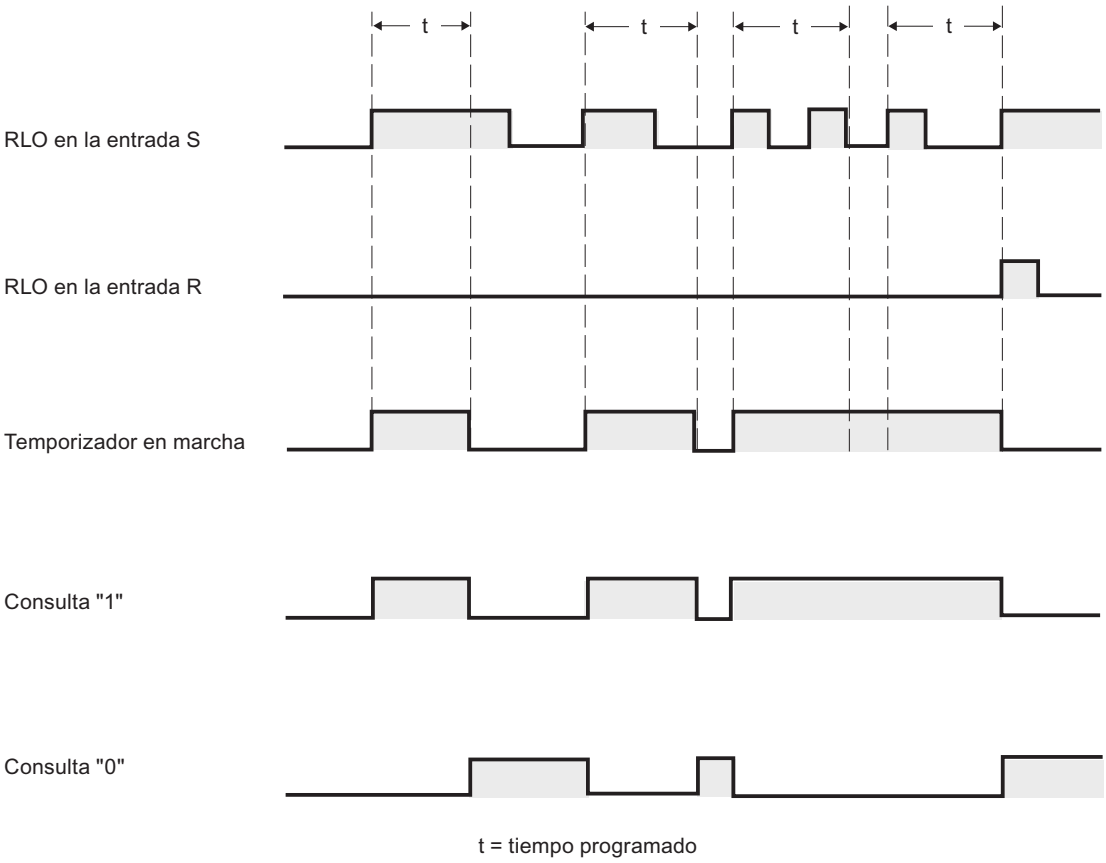
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como impulso prolongado":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_PEXT(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q := "Tag_Status",
                       BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. Mientras el temporizador está en marcha, el operando "Tag_Status" devuelve el estado lógico "1". Una vez transcurrido el tiempo, el operando "Tag_Status" adopta el valor lógico "0". Si el estado lógico de la entrada S cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo "Tag_Number".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 6189)

S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión

Descripción

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" arranca el temporizador programado como retardo a la conexión cuando se detecta un flanco de señal ascendente en el parámetro S. El temporizador cuenta el tiempo programado TV mientras el estado lógico del parámetro S sea "1".

Si el tiempo ha transcurrido correctamente y el parámetro S sigue presentando el estado lógico "1", el parámetro Q devuelve el estado lógico "1". Si el estado lógico del parámetro S cambia de "1" a "0" mientras el temporizador está contando, este se detiene. En este caso, la salida Q adopta el estado lógico "0".

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

Si el temporizador está contando y el estado lógico de la entrada R cambia de "0" a "1", el valor actual de tiempo y la base de tiempo también se ponen a cero. En este caso, el estado lógico del parámetro Q es "0". El temporizador se resetea si el parámetro R presenta el estado lógico "1", aunque el temporizador no esté en marcha y el resultado lógico (RLO) del parámetro S sea "1".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión" se utiliza la sintaxis siguiente:

SCL

```
S_ODT(T_NO := <Operando>,
      S := <Operando>,
      TV := <Operando>,
      R := <Operando>,
      Q => <Operando>,
      BI =><Operando>)
```

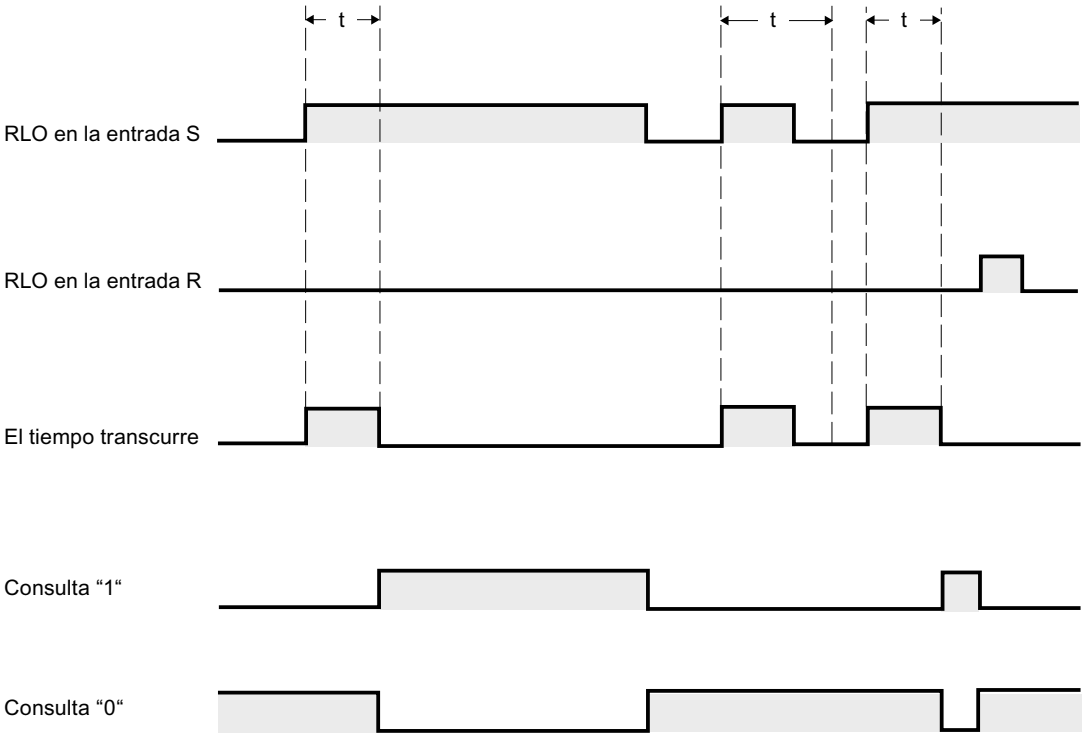
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión":



t = tiempo programado

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_ODT(T_NO := "Timer_1",
                      S := "Tag_1",
                      TV := "Tag_Number",
                      R := "Tag_Reset",
                      Q := "Tag_Status",
                      BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el tiempo "Tag_Number" mientras el estado lógico del operando "Tag_1" sea "1".

Si el tiempo ha transcurrido correctamente y el operando "Tag_Status" presenta el estado lógico "1", el operando "Tag_Status" se pone a "1". Si el estado lógico del operando "Tag_1" cambia de "1" a "0" mientras el temporizador está contando, este se detiene. El operando "Tag_Status" devuelve en ese caso el estado lógico "0".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 6189)

S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria

Descripción

La instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" arranca el temporizador programado cuando se detecta un flanco de señal ascendente en el parámetro S. El temporizador deja de contar el tiempo programado TV aunque el estado lógico del parámetro S cambie a "0".

Una vez transcurrido el tiempo, el parámetro Q devuelve el estado lógico "1" independientemente del estado lógico del parámetro S. Si el estado lógico del parámetro S cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo programado TV.

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

El estado lógico "1" del parámetro R pone a "0" el valor actual de tiempo y la base de tiempo, independientemente del estado lógico del parámetro S. En este caso, el estado lógico del parámetro Q es "0".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria" se utiliza la sintaxis siguiente:

SCL

```
S_ODTS(T_NO := <Operando>,
       S := <Operando>,
       TV := <Operando>,
       R := <Operando>,
       Q => <Operando>,
       BI =><Operando>)
```

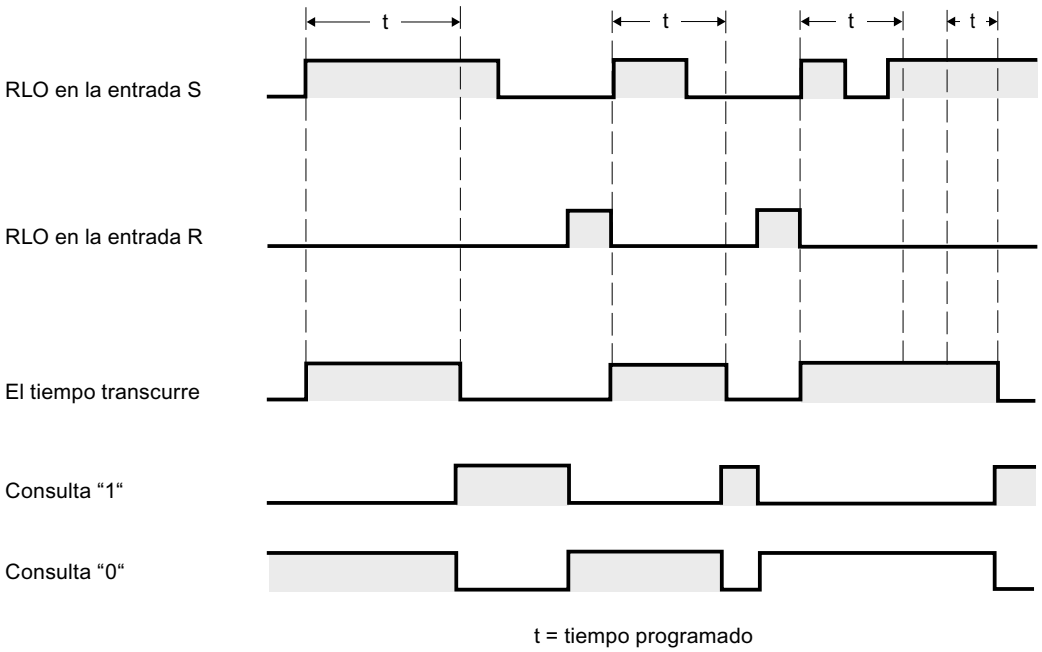
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la conexión con memoria":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_ODTS(T_NO := "Timer_1",
                       S := "Tag_1",
                       TV := "Tag_Number",
                       R := "Tag_Reset",
                       Q := "Tag_Status",
                       BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el tiempo "Tag_Number".

Una vez transcurrido el tiempo, el operando "Tag_Status" devuelve el estado lógico "1" independientemente del estado lógico del operando "Tag_1". Si el estado lógico del operando "Tag_1" cambia de "0" a "1" mientras el temporizador está contando, el temporizador se inicia de nuevo con el tiempo "Tag_Number".

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 6189)

S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión

Descripción

La instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" arranca el temporizador programado cuando se detecta un flanco de señal descendente en el parámetro S. El temporizador cuenta el tiempo programado TV. Mientras el temporizador está contando o el parámetro S devuelve el estado lógico "1", el parámetro Q presenta el estado lógico "1".

Si el tiempo ha transcurrido y el estado lógico es "0", el parámetro Q adopta el valor lógico "0". Si el estado lógico del parámetro S cambia de "0" a "1" mientras el temporizador está en marcha, este se detiene. El temporizador solo arranca de nuevo cuando se detecta un flanco de señal descendente en el parámetro S.

El tiempo se compone internamente de un valor de tiempo y una base de tiempo y se programa en el parámetro TV. Cuando se inicia la instrucción, el valor de tiempo programado se cuenta hacia atrás hasta cero. La base de tiempo indica con qué periodo de tiempo se modifica el valor de tiempo. El valor actual de tiempo se deposita en el parámetro BI.

El estado lógico "1" del parámetro R pone a "0" el valor actual de tiempo y la base de tiempo. En este caso, el estado lógico del parámetro Q es "0".

Los datos de la instrucción se actualizan con cada acceso. Por este motivo es posible que una consulta de datos al principio del ciclo devuelva valores diferentes en comparación con el fin del ciclo.

Nota

En la célula de temporización, el sistema operativo reduce el valor de tiempo en una unidad respectivamente, y en un intervalo definido por la base de tiempo, hasta que el valor de tiempo sea "0". La reducción se realiza de modo asíncrono respecto al programa de usuario. De este modo el tiempo resultante es menor que el valor de tiempo deseado, como máximo hasta un intervalo de la base de tiempo.

Encontrará un ejemplo de la estructura de una célula de temporización en: Consulte también "L: Cargar valor del temporizador".

Sintaxis

Para la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión" se utiliza la sintaxis siguiente:

SCL

```
S_OFFDT(T_NO := <Operando>,
        S := <Operando>,
        TV := <Operando>,
        R := <Operando>,
        Q => <Operando>,
        BI =><Operando>)
```

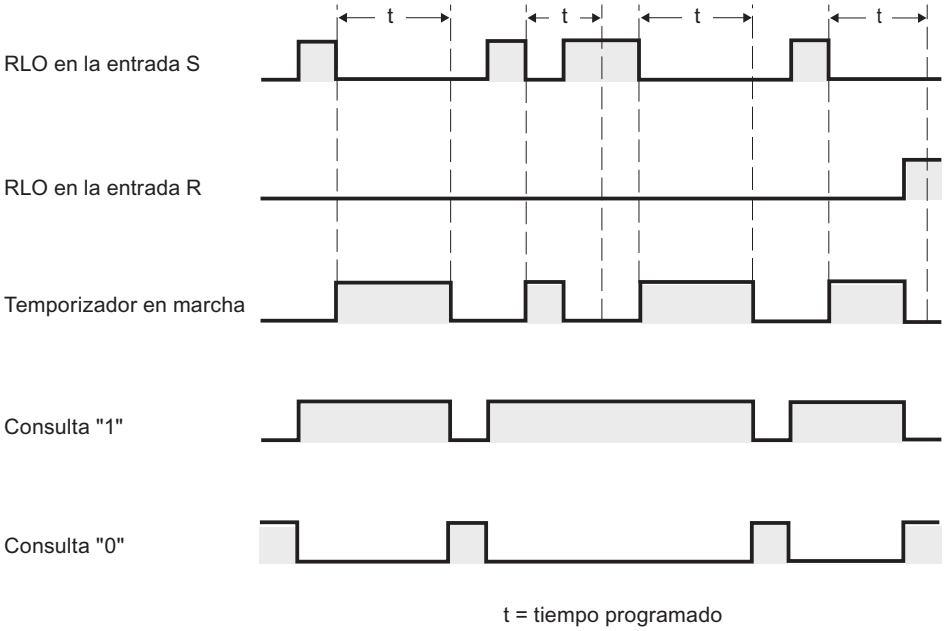
La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
T_NO	Input	TIMER, INT	Temporizador que arranca. El número de temporizadores depende de la CPU.
S	Input	BOOL	Entrada de arranque
TV	Input	S5TIME, WORD	Valor de tiempo predeterminado
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del temporizador
BI	Output	WORD	Valor actual de tiempo en código dual
Valor de función		S5TIME	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La imagen siguiente muestra el cronograma de impulsos de la instrucción "Parametrizar y arrancar temporizador como retardo a la desconexión":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_OFFDT(T_NO := "Timer_1",
                        S := "Tag_1",
                        TV := "Tag_Number",
                        R := "Tag_Reset",
                        Q := "Tag_Status",
                        BI := "Tag_Value");
```

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el temporizador "Timer_1" se pone en marcha. El temporizador cuenta el tiempo "Tag_Number". Mientras el temporizador está contando o el operando "Tag_1" devuelve el estado lógico "1", el operando "Tag_Status" presenta el estado lógico "1".

Si el tiempo ha transcurrido y el estado lógico del operando "Tag_1" es "0", el operando "Tag_Status" adopta el valor lógico "0". Si el estado lógico del operando "Tag_1" cambia de "0" a "1" mientras el temporizador está en marcha, el temporizador se reinicia. El temporizador solo arranca de nuevo cuando se detecta un flanco descendente en el parámetro S.

El valor actual de tiempo se guarda en código dual en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

L: Cargar valor del temporizador (Página 6189)

Contadores

Contadores CEI

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje del parámetro CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor actual de contaje del parámetro CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos (INT). Cuando se alcanza el valor límite superior, el estado lógico del parámetro CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. El estado lógico del parámetro Q es determinado por el parámetro PV. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0". En el parámetro PV también se puede indicar una constante.

El valor del parámetro CV se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico del parámetro CU no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTU (p. ej., "CTU_DB")
- Declaración como variable local del tipo CTU en la sección "Static" de un bloque (p. ej., #MyCTU_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en "Bloques de programa >

Bloques de sistema" de la carpeta "Recursos de programa", en el árbol de proyecto. Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro R de la instrucción. Si un bloque diferente contiene instancias de la instrucción "Contador ascendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

Tabla 9-167 Bloque de datos del tipo de datos de sistema IEC_COUNTER (Shared DB)

SCL

```
<DB_Contador_CEI>.CTU(CU := <Operando>,
                      R := <Operando>,
                      PV := <Operando>,
                      Q => <Operando>,
                      CV => <Operando>)
```

Tabla 9-168 Variable local

SCL

```
#myLocal_counter(CU := <Operando>,
                 R := <Operando>,
                 PV := <Operando>,
                 Q => <Operando>,
                 CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
CU	Input	BOOL	Entrada de contaje
R	Input	BOOL	Entrada de reset
PV	Input	INT	Valor con el que se activa la salida Q.
Q	Output	BOOL	Estado del contador
CV	Output	INT	Valor actual de contaje

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"CTU_DB".CTU(CU := "Tag_Start",
             R := "Tag_Reset",
```

SCL

```
                PV := "Tag_PresetValue",
                Q => "Tag_Status",
                CV => "Tag_CounterValue");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor actual de contaje del operando "Tag_CounterValue" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos (INT = 32767).

La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor del operando "Tag_PresetValue". En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0". El valor actual de contaje se guarda en el operando "Tag_CounterValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor del parámetro CV. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje del parámetro CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos (INT) indicado. Cuando se alcanza el valor límite inferior, el estado lógico del parámetro CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en el parámetro Q. Si el valor actual de contaje es menor o igual a cero, el parámetro Q adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro Q es "0".

El valor del parámetro CV se pone al valor del parámetro PV cuando el estado lógico del parámetro LD cambia a "1". Mientras el parámetro LD tenga el estado lógico "1", el estado lógico del parámetro CD no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTD (p. ej., "CTD_DB")
- Declaración como variable local del tipo CTD en la sección "Static" de un bloque (p. ej., #MyCTD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en "Bloques de programa > Bloques de sistema" de la carpeta "Recursos de programa", en el árbol de proyecto. Encontrará más información al respecto en "Consulte también".

El sistema operativo resetea las instancias de la instrucción "Contador descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro LD de la instrucción. En el parámetro PV se indica en este caso el valor inicial deseado para el parámetro CV. Si un bloque diferente contiene instancias de la instrucción "Contador descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

Tabla 9-169 Bloque de datos del tipo de datos de sistema IEC_COUNTER (Shared DB)

SCL

```
<DB_Contador_CEI>.CTD(CD  := <Operando>,
                      LD : = <Operando>,
                      PV : = <Operando>,
                      Q => <Operando>,
                      CV  => <Operando>)
```

Tabla 9-170 Variable local

SCL

```
#myLocal_counter(CD : = <Operando>,
                 LD : = <Operando>,
                 PV := <Operando>,
                 Q => <Operando>,
                 CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
CD	Input	BOOL	Entrada de contaje
LD	Input	BOOL	Entrada de carga
PV	Input	INT	Valor con el que se activa la salida Q.
Q	Output	BOOL	Estado del contador
CV	Output	INT	Valor actual de contaje

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"CTD_DB".CTD(CD := "Tag_Start",
             LD := "Tag_Load",
             PV := "Tag_PresetValue",
             Q => "Tag_Status",
             CV => "Tag_CounterValue");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", la instrucción "Contador descendente" se ejecuta y el valor del operando "Tag_CounterValue" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos (INT = -32768).

El operando "Tag_Status" devuelve el estado lógico "1" mientras el valor del contador actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0". El valor actual de contaje se guarda en el operando "Tag_CounterValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiinstancia (Página 2264)

CTUD: Contador ascendente - descendente

Descripción

La instrucción "Contador ascendente - descendente" incrementa y decrementa el valor de contaje del parámetro CV. Cuando el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual del parámetro CV se incrementa en uno. Cuando el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje del parámetro CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor de contaje actual no cambia en el parámetro CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado (INT). Una vez alcanzado el valor límite superior, el valor de contaje ya no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado (INT), ya no se decrementa el valor de contaje.

Cuando el estado lógico del parámetro LD cambia a "1", el valor de contaje del parámetro CV se pone al valor del parámetro PV. Mientras el parámetro LD tenga el estado lógico "1", el estado lógico de los parámetros CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", un cambio del estado lógico de los

parámetros CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en el parámetro QU. Si el valor de contaje actual es mayor o igual al valor del parámetro PV, el parámetro QU adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QU es "0". En el parámetro PV también se puede indicar una constante.

El estado del contador descendente se puede consultar en el parámetro QD. Si el valor de contaje actual es menor o igual a cero, el parámetro QD adopta el estado lógico "1". En los restantes casos, el estado lógico del parámetro QD es "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTUD (p. ej., "CTUD_DB")
- Declaración como variable local del tipo CTUD en la sección "Static" de un bloque (p. ej., #MyCTUD_COUNTER)

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en este se puede determinar si el contador CEI se deposita en un bloque de datos propio (instancia individual), o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en "Bloques de programa > Bloques de sistema" de la carpeta "Recursos de programa", en el árbol de proyecto. Encontrará más información al respecto en "Consulte también".

El sistema operativo resetea las instancias de la instrucción "Contador ascendente - descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con los siguientes valores de parámetros:

- Si se utiliza como contador ascendente, el valor del parámetro R debe ponerse a "1".
- Si se utiliza como contador descendente, el valor del parámetro LD debe ponerse a "1". En el parámetro PV debe indicarse en este caso el valor inicial deseado del parámetro CV.

Si un bloque diferente contiene instancias de la instrucción "Contador ascendente - descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Sintaxis

Para la instrucción "Contador ascendente - descendente" se utiliza la siguiente sintaxis:

Tabla 9-171 Bloque de datos del tipo de datos de sistema IEC_COUNTER (Shared DB)

SCL

```
<DB_Contador_CEI>.CTUD(CU := <Operando>,
                       CD := <Operando>,
                       R := <Operando>,
                       LD: = <Operando>,
```

SCL

```
                    PV: = <Operando>,
                    QU=> <Operando>,
                    QD: = <Operando>,
                    CV => <Operando>)
```

Tabla 9-172 Variable local

SCL

```
myLocal_counter(CU: = <Operando>,
                CD: = <Operando>,
                R: = <Operando>,
                LD: = <Operando>,
                PV := <Operando>,
                QU => <Operando>,
                QD := <Operando>,
                CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
CU	Input	BOOL	Entrada de contaje ascendente
CD	Input	BOOL	Entrada de contaje descendente
R	Input	BOOL	Entrada de reset
LD	Input	BOOL	Entrada de carga
PV	Input	INT	Valor con el que se activa la salida QU.
QU	Output	BOOL	Estado del contador ascendente
QD	Output	BOOL	Estado del contador descendente
CV	Output	INT	Valor actual de contaje

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"CTUD_DB".CTUD(CU := "Tag_Start1",
               CD := "Tag_Start2",
               LD := "Tag_Load",
               R := "Tag_Reset",
               PV := "Tag_PresetValue",
               QU => "Tag_CU_Status",
               QD => "Tag_CD_Status",
               CV => "Tag_CounterValue");
```

Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_Start1", el valor actual del contador se incrementa en uno y se deposita en el operando "Tag_CounterValue". Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_Start2", el valor del contador se decrementa en uno y se deposita también en el operando "Tag_CounterValue". Cuando se detecta un flanco de señal ascendente en el parámetro CU el valor del contador va incrementando hasta alcanzar el valor límite superior hasta alcanzar el valor límite de 32 767. Cuando se detecta un flanco de señal ascendente en el parámetro CD el valor de contaje se sigue decrementando hasta alcanzar el valor límite inferior de INT = -32 768.

El operando "Tag_CU_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea mayor o igual al valor del operando "Tag_PresetValue". En todos los demás casos, la salida "Tag_CU_Status" devuelve el estado lógico "0".

El operando "Tag_CD_Status" devuelve el estado lógico "1" mientras el valor del contador actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_CD_Status" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Llamada de bloques de función como instancia individual o multiiinstancia (Página 2264)

Contadores SIMATIC

S_CU: Parametrizar e incrementar contador

Descripción

La instrucción "Parametrizar e incrementar contador" permite incrementar el valor de un contador. Si el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. El valor de contaje actual se deposita en el parámetro CV. El valor de contaje sigue incrementando hasta alcanzar el límite de "999". Una vez alcanzado el valor límite, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente.

Cuando el estado lógico del parámetro S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el resultado lógico (RLO) de la entrada CU es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetro R sea "1", un cambio en el estado lógico de los parámetros CU y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico del parámetro Q es "1" si el valor de contaje es mayor que cero. Si el valor de contaje es igual a cero, el parámetro Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Sintaxis

Para la instrucción "Parametrizar e incrementar contador" se utiliza la sintaxis siguiente:

SCL

```
S_CU(C_NO := <Operando>,
     CU := <Operando>,
     S := <Operando>,
     PV := <Operando>,
     R := <Operando>,
     Q=> <Operando>,
     CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetros	Declaración	Tipo de datos	Descripción
C_NO	Input	COUNTER, INT	Contadores La cantidad de contadores depende de la CPU
CU	Input	BOOL	Entrada de contaje ascendente
S	Input	BOOL	Entrada para preajustar el contador
PV	Input	WORD	Valor de contaje predeterminado (C#0 a C#999) en formato BCD
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del contador
CV	Output	WORD, S5TIME, DATE	Valor actual de contaje
Valor de función		WORD, S5TIME, DATE	Valor de contaje actual en formato BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_CU(C_NO := "Counter_1",
```

SCL

```
                    CU := "Tag_Start",
                    S := "Tag_1",
                    PV := "Tag_PresetValue",
                    R := "Tag_Reset",
                    Q => "Tag_Status",
                    CV => "Tag_Value");
```

Cuando el estado lógico del parámetro "Tag_Start" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es menor que "999", este valor se incrementará en uno. Si el estado lógico de la entrada "Tag_1" cambia de "0" a "1", el valor de contaje en formato BCD se pone al valor del operando "Tag_PresetValue". El valor de contaje se pone a "0" si el operando "Tag_Reset" tiene el estado lógico "1".

El valor de contaje actual se almacena en formato hexadecimal en el operando "Tag_Value".

La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea diferente a "0". El valor actual de contaje se guarda en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

S_CD: Parametrizar y decrementar contador

Descripción

La instrucción "Parametrizar y decrementar contador" permite decrementar el valor de un contador. Si el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se deposita en el parámetro CV. El valor de contaje sigue decrementando hasta alcanzar el límite inferior "0". Una vez alcanzado el valor límite, el valor de contaje no se decrementa más al detectarse un flanco de señal ascendente.

Cuando el estado lógico del parámetro S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el resultado lógico (RLO) del parámetro CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el estado lógico del parámetro R sea "1", un cambio en el estado lógico de los parámetros CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico del parámetro Q es "1" si el valor de contaje es mayor que cero. Si el valor de contaje es igual a cero, el parámetro Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Sintaxis

Para la instrucción "Parametrizar y decrementar contador" se utiliza la sintaxis siguiente:

SCL

```
S_CD(C_NO := <Operando>,
     CD := <Operando>,
     S := <Operando>,
     PV := <Operando>,
     R := <Operando>,
     Q=> <Operando>,
     CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetros	Declaración	Tipo de datos	Descripción
C_NO	Input	COUNTER, INT	Contadores El número de contadores depende de la CPU.
CD	Input	BOOL	Entrada de contaje descendente
S	Input	BOOL	Entrada para preajustar el contador
PV	Input	WORD	Valor de contaje predeterminado (C#0 a C#999) en formato BCD
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del contador
CV	Output	WORD, S5TIME, DATE	Valor actual de contaje
Valor de función		WORD, S5TIME, DATE	Valor de contaje actual en formato BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CD := "Tag_Start",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1" (flanco de señal ascendente) y el valor de contaje actual es superior a "0", el valor de contaje se decrementa en uno. Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el valor de contaje adopta el valor del operando "Tag_PresetValue" en formato BCD. El valor de contaje se pone a "0" si el operando "Tag_Reset" tiene el estado lógico "1".

El valor de contaje actual se almacena en el operando "Tag_Value".

El operando "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea distinto de "0". El valor actual de contaje se guarda en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

S_CUD: Parametrizar e incrementar/decrementar contador

Descripción

La instrucción "Parametrizar e incrementar/decrementar contador" permite incrementar y decrementar el valor de un contador. Si el estado lógico del parámetro CU cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje actual se incrementa en uno. Si el estado lógico del parámetro CD cambia de "0" a "1" (flanco de señal ascendente), el valor de contaje se decrementa en uno. El valor de contaje actual se deposita en el parámetro CV. Si en un ciclo del programa se detecta un flanco de señal ascendente en los parámetros CU y CD el valor de contaje permanece invariable.

El valor de contaje sigue incrementando hasta alcanzar el límite superior "999". Una vez alcanzado el valor límite superior, el valor de contaje no se incrementa más aunque se detecte un flanco de señal ascendente. Cuando se alcanza el valor límite inferior "0", no se decrementa más el valor de contaje.

Cuando el estado lógico del parámetro S cambia de "0" a "1", el valor de contaje adopta el valor del parámetro PV. Si el contador se activa y el resultado lógico (RLO) de los parámetros

CU y CD es "1", el contador cuenta una única vez en el ciclo siguiente aunque no se detecte ningún cambio de flanco.

El valor de contaje se pone a cero cuando el estado lógico del parámetro R cambia a "1". Mientras el parámetro R tenga el estado lógico "1", el estado lógico de los parámetros CU, CD y S no tiene ningún efecto sobre el valor de contaje.

El estado lógico del parámetro Q es "1" si el valor de contaje es mayor que cero. Si el valor de contaje es igual a cero, el parámetro Q devuelve el estado lógico "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

Sintaxis

Para la instrucción "Parametrizar e incrementar/decrementar contador" se utiliza la sintaxis siguiente:

SCL

```
S_CUD(C_NO := <Operando>,
      CU := <Operando>,
      CD := <Operando>,
      S := <Operando>,
      PV := <Operando>,
      R := <Operando>,
      Q=> <Operando>,
      CV => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetros	Declaración	Tipo de datos	Descripción
C_NO	Input	COUNTER, INT	Contadores La cantidad de contadores depende de la CPU
CU	Input	BOOL	Entrada de contaje ascendente
CD	Input	BOOL	Entrada de contaje descendente
S	Input	BOOL	Entrada para preajustar el contador
PV	Input	WORD	Valor de contaje predeterminado (C#0 a C#999) en formato BCD
R	Input	BOOL	Entrada de reset
Q	Output	BOOL	Estado del contador

Parámetros	Declaración	Tipo de datos	Descripción
CV	Output	WORD, S5TIME, DATE	Valor de contaje actual (hexadecimal)
Valor de función		WORD, S5TIME, DATE	Valor de contaje actual en formato BCD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := S_CD(C_NO := "Counter_1",
                     CU := "Tag_CU",
                     CD := "Tag_CD",
                     S := "Tag_1",
                     PV := "Tag_PresetValue",
                     R := "Tag_Reset",
                     Q => "Tag_Status",
                     CV => "Tag_Value");
```

Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_CU" y el valor de contaje actual es menor que "999", el valor de contaje se incrementa en uno. Cuando se detecta un flanco de señal ascendente en el estado lógico del operando "Tag_CD" y el valor de contaje actual es mayor que "0", el valor de contaje se decrementa en uno.

Cuando el estado lógico del operando "Tag_1" cambia de "0" a "1", el valor de contaje adopta el valor del operando "Tag_PresetValue" en formato BCD. El valor de contaje se pone a "0" si el operando "Tag_Reset" tiene el estado lógico "1".

El valor de contaje actual se almacena en el operando "Tag_Value".

El operando "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea distinto de "0". El valor actual de contaje se guarda en el operando "Tag_Value" y se devuelve como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Funciones matemáticas

ABS: Calcular valor absoluto

Descripción

La instrucción "Calcular valor absoluto" permite calcular el valor absoluto de un valor de entrada y guardar el resultado en el operando indicado.

Sintaxis

Para la instrucción "Calcular valor absoluto" se utiliza la siguiente sintaxis:

SCL

```
ABS(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	INT, DINT, REAL	Valor de entrada
Valor de función		INT, DINT, REAL	Valor absoluto del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := ABS("Tag_Value");
"Tag_Result2" := ABS("Tag_Value1"*"Tag_Value2");
```

El valor absoluto del valor de entrada se devuelve en el formato del valor de entrada como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	-2
Tag_Result1	2
Tag_Value1	4
Tag_Value2	-1
Tag_Result2	4

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

MIN: Determinar mínimo

Descripción

Con la instrucción "Determinar mínimo" se comparan los valores de las entradas indicadas y se devuelve el menor como resultado. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

Para la ejecución de la instrucción se pueden indicar como mínimo dos y como máximo 32 entradas.

Sintaxis

Para la instrucción "Determinar mínimo" se utiliza la siguiente sintaxis:

SCL

```
MIN(IN1 := <Operando>,
    IN2 := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN1	Input	INT, DINT, REAL, TIME, DATE, DT	Primer valor de entrada
IN2	Input	INT, DINT, REAL, TIME, DATE, DT	Segundo valor de entrada
INn	Input	INT, DINT, REAL, TIME, DATE, DT	Entradas insertadas adicionalmente cuyos valores se comparan
Valor de función		INT, DINT, REAL, TIME, DATE, DT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := MIN(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	12 222
IN2	Tag_Value2	14 444
IN3	Tag_Value3	13 333
Valor de función	Tag_Result	12 222

La instrucción compara los valores de los operandos indicados y copia el valor menor ("Tag_Value1") en el operando "Tag_Result".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

MAX: Determinar máximo

Descripción

Con la instrucción "Determinar máximo" se comparan los valores de las entradas indicadas y se devuelve el mayor como resultado. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

Para la ejecución de la instrucción se pueden indicar como mínimo dos y como máximo 32 entradas.

Sintaxis

Para la instrucción "Determinar máximo" se utiliza la siguiente sintaxis:

```
SCL
MAX(IN1 := <Operando>,
    IN2 := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN1	Input	INT, DINT, REAL, TIME, DATE, DT	Primer valor de entrada
IN2	Input	INT, DINT, REAL, TIME, DATE, DT	Segundo valor de entrada
INn	Input	INT, DINT, REAL, TIME, DATE, DT	Entradas insertadas adicionalmente cuyos valores se comparan
Valor de función		INT, DINT, REAL, TIME, DATE, DT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := MAX(IN1 := "Tag_Value1",
                    IN2 := "Tag_Value2",
                    IN3 := "Tag_Value3");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	Tag_Value1	12 222
IN2	Tag_Value2	14 444
IN3	Tag_Value3	13 333
Valor de función	Tag_Result	14 444

La instrucción compara los valores de los operandos indicados y copia el valor mayor ("Tag_Value2") en el operando "Tag_Result".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor del parámetro IN a los valores de los parámetros MN y MX. Para ello, el valor del parámetro MN no puede ser mayor que el valor del parámetro MX.

Si el valor del parámetro IN cumple la condición MN <= IN <= MX, se devuelve como resultado de la instrucción. Si no se cumple la condición y el valor de entrada IN rebasa por defecto el límite inferior MN, se devuelve como resultado el valor del parámetro MN. Si se rebasa por exceso el límite superior MX se devuelve como resultado el valor del parámetro MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no está definido.

Para poder ejecutar la instrucción, es imprescindible que los operandos de todos los parámetros sean del mismo tipo de datos.

Sintaxis

Para la instrucción "Ajustar valor límite" se utiliza la siguiente sintaxis:

SCL

```
LIMIT(MN := <Operando>,
      IN := <Operando>,
      MX := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
MN	Input	INT, DINT, REAL, TIME, DATE, DT	Límite inferior
IN	Input	INT, DINT, REAL, TIME, DATE, DT	Valor de entrada
MX	Input	INT, DINT, REAL, TIME, DATE, DT	Límite superior
Valor de función		INT, DINT, REAL, TIME, DATE, DT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := LIMIT(MN := "Tag_Minimum",
                      IN := "Tag_Value",
                      MX := "Tag_Maximum");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_Minimum	12 000
IN	Tag_Value	8 000
MX	Tag_Maximum	16 000
Valor de función	Tag_Result	12 000

El valor del operando "Tag_Value" se compara con los valores de los operandos "Tag_Minimum" y "Tag_Maximum". Puesto que el valor del operando "Tag_Value" es menor que el valor límite inferior, el valor del operando "Tag_Minimum" se copia en el operando "Tag_Result".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

SQR: Calcular cuadrado

Descripción

La instrucción "Calcular cuadrado" permite elevar al cuadrado el valor de entrada y guardar el resultado en el operando indicado.

Sintaxis

Para la instrucción "Calcular cuadrado" se utiliza la siguiente sintaxis:

SCL

```
SQR(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Cuadrado del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := SQR("Tag_Value");
"Tag_Result2" := SQR((SQR("Tag_Value1"))*"Tag_Value2);
```

El cuadrado del valor de entrada se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	2.5
Tag_Result1	6.25
Tag_Value1	6.0
Tag_Value2	2.0
Tag_Result2	5184.0

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

SQRT: Calcular raíz cuadrada

Descripción

La instrucción "Calcular raíz cuadrada" permite extraer la raíz cuadrada del valor de entrada y guardar el resultado en el operando indicado. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la instrucción devuelve un número en coma flotante no válido. Si el valor de entrada es "-0", el resultado también es "-0".

Sintaxis

Para la instrucción "Calcular raíz cuadrada" se utiliza la siguiente sintaxis:

SCL

```
SQRT(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Raíz cuadrada del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := SQRT("Tag_Value");
"Tag_Result2" := SQRT((SQR("Tag_Value1"))+"Tag_Value2");
```

La raíz cuadrada del valor de entrada se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	4.0
Tag_Result1	2.0
Tag_Value1	3.0
Tag_Value2	16.0
Tag_Result2	5.0

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2.718282) a partir del valor de entrada. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero, la instrucción devuelve un número en coma flotante no válido.

Sintaxis

Para la instrucción "Calcular logaritmo natural" se utiliza la siguiente sintaxis:

```
SCL
LN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Logaritmo natural del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := LN("Tag_Value");
"Tag_Result2" := LN("Tag_Value1"+"Tag_Value2");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	2.5
Tag_Result1	0.916
Tag_Value1	1.5
Tag_Value2	3.2
Tag_Result2	1.548

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2.718282) a partir del valor de entrada y guardar el resultado en el operando indicado.

Sintaxis

Para la instrucción "Calcular valor exponencial" se utiliza la siguiente sintaxis:

```
SCL
EXP(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada
Valor de función		Números en coma flotante	Valor exponencial del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result1" := EXP("Tag_Value");
"Tag_Result2" := EXP("Tag_Value1"/"Tag_Value2");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	20.5
Tag_Result1	799 902 200
Tag_Value1	15.5
Tag_Value2	30.2
Tag_Result2	1.671

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno del valor de entrada. El valor de entrada se ha de indicar en radianes.

Sintaxis

Para la instrucción "Calcular valor de seno" se utiliza la siguiente sintaxis:

```
SCL
SIN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada (tamaño de un ángulo en radianes)
Valor de función		Números en coma flotante	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := SIN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	+1.570796 (π/2)
Tag_Result	1.0

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno del valor de entrada. El valor de entrada se ha de indicar en radianes.

Sintaxis

Para la instrucción "Calcular valor de coseno" se utiliza la siguiente sintaxis:

```
SCL
COS(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada (tamaño de un ángulo en radianes)
Valor de función		Números en coma flotante	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := COS("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	+1.570796 (π/2)
Tag_Result	0

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente del valor de entrada. El valor de entrada se ha de indicar en radianes.

Sintaxis

Para la instrucción "Calcular valor de tangente" se utiliza la siguiente sintaxis:

SCL

```
TAN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de entrada (tamaño de un ángulo en radianes)
Valor de función		Números en coma flotante	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := TAN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	+3.141593 (π)
Tag_Result	0

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno el tamaño del ángulo que equivale a este valor. Como valores de entrada solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se devuelve en radianes y puede estar comprendido entre -π/2 y +π/2.

Sintaxis

Para la instrucción "Calcular valor de arcoseno" se utiliza la siguiente sintaxis:

SCL

```
ASIN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de seno
Valor de función		Números en coma flotante	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ASIN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	1.0
Tag_Result	+1.570796 (π/2)

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno el tamaño del ángulo que equivale a este valor. Como valores de entrada solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado se devuelve en radianes y puede estar comprendido entre 0 y +π.

Sintaxis

Para la instrucción "Calcular valor de arcocoseno" se utiliza la siguiente sintaxis:

SCL

```
ACOS(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	Números en coma flotante	Valor de coseno
Valor de función		Números en coma flotante	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := ACOS("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	0
Tag_Result	+1.570796 (π/2)

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente el tamaño del ángulo que equivale a este valor. Como valores de entrada solo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado se devuelve en radianes y puede estar comprendido entre -π/2 y +π/2.

Sintaxis

Para la instrucción "Calcular valor de arcotangente" se utiliza la siguiente sintaxis:

```
SCL
ATAN(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	REAL	Valor de tangente
Valor de función		REAL	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ATAN("Tag_Value");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value	1.0
Tag_Result	+0.785398 (π/4)

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Transferencia

BLKMOV: Copiar área

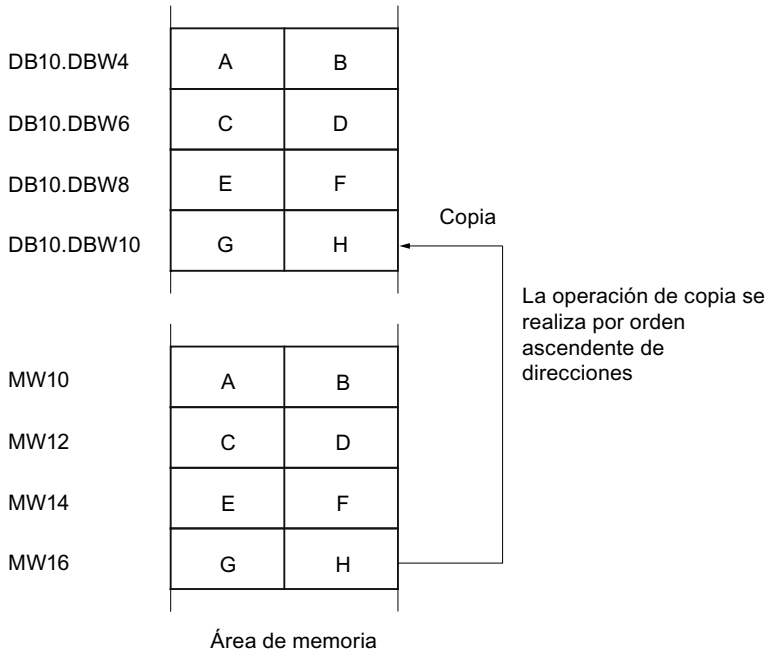
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que durante la ejecución de la instrucción "Copiar área" no se modifican los datos de origen pues, de lo contrario, no se podría garantizar la coherencia de los datos de destino.

Posibilidad de interrupción

Mientras el área de origen no forme parte de un bloque de datos que solo figura en la memoria de carga, no hay limitaciones en cuanto a la profundidad de anidamiento.

Por el contrario, en caso de interrumpirse la ejecución de BLKMOV, en la que se copia desde un DB no relevante para la ejecución, no puede anidarse más la ejecución de la BLKMOV.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Reglas para copiar bloques de datos no relevantes para la ejecución

El área de origen también puede encontrarse en un bloque de datos no relevante para la ejecución en la memoria de carga. Los bloques de datos no relevantes para la ejecución se identifican con la palabra clave UNLINKED.

Si en la memoria de trabajo se copia un bloque de datos no relevante para la ejecución con la instrucción "Copiar área" y, al mismo tiempo, este se vuelve a cargar p. ej. desde la

programadora, la ejecución de la instrucción puede retardarse varios milisegundos. Esto prolonga el ciclo del OB y puede activar la vigilancia del tiempo de ciclo.

Si un bloque de datos no relevante para la ejecución se copia con la instrucción "Copiar área" y la operación de copia se interrumpe, la instrucción no se podrá seguir ejecutando.

Si en la CPU existe la instrucción "Leer de un bloque de datos de la memoria de carga", debe utilizarse dicha instrucción para leer los bloques de datos no relevantes para la ejecución en la memoria de carga. Si se utiliza la instrucción "Copiar área" se emite el error W#16#8092.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

SCL

```
BLKMOV(SRCBLK := <Operando>,
       DSTBLK => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Descripción
SRCBLK	Input	ANY	Indicación del área de memoria que se copia (área de origen).
DSTBLK	Output	ANY	Indicación del área de memoria en la que se copia (área de destino).
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8092	El área de origen o destino está solo en la memoria de carga.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" copia los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 512 bytes como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro BVAL o BLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

SCL

```
UBLKMOV(SRCBLK := <Operando>,
        DSTBLK => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
SRCBLK	Input	ANY	Indicación del área de memoria que se copia (área de origen).
DSTBLK	Output	ANY	Indicación del área de memoria en la que se copia (área de destino).
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

FILL: Rellenar área

Descripción

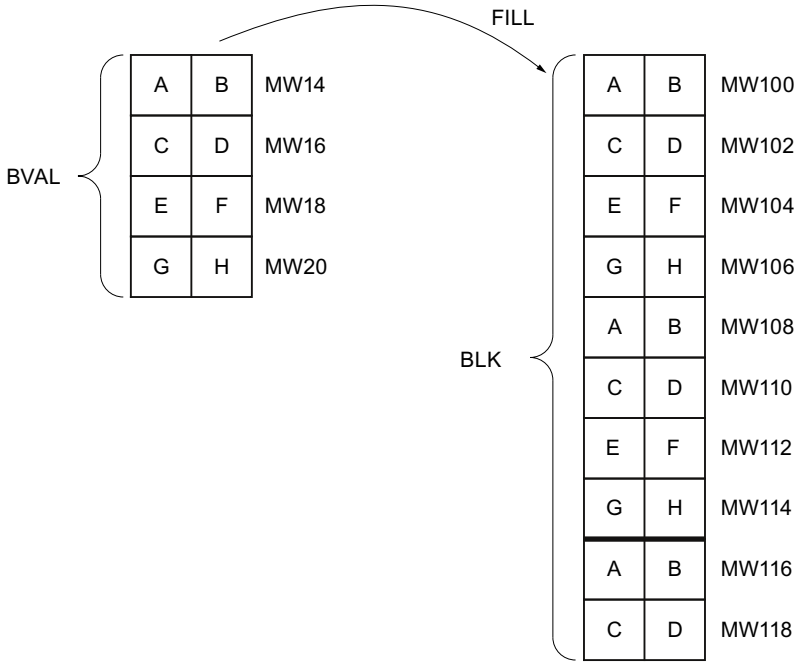
La instrucción "Rellenar área" rellena un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino hasta que el área de destino quede escrita

por completo. La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: el contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que durante la ejecución de la instrucción "Rellenar área" no se modifican los datos de origen pues, de lo contrario, no se podría garantizar la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

La instrucción "Rellenar área" no permite escribir las siguientes áreas de memoria:

- Instrucciones, bloques de datos de sistema
- Contadores
- Temporizadores
- Áreas de memoria de la periferia.

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea un múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellena igualmente hasta el último byte.

Si el área de destino que se debe preasignar es menor que el área de origen, solo se copian los datos que caben en el área de destino.

Si el área de destino o de origen disponible realmente es menor que el tamaño del área de memoria parametrizado para el área de origen o destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo BOOL, la longitud indicada debe ser divisible por 8. De lo contrario, la instrucción no se ejecutará.

Si el área de destino es del tipo de datos STRING, la instrucción escribe la cadena de caracteres completa, incluida la información de administración.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro BVAL o BLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

SCL

```
FILL(BVAL := <Operando>,
     BLK => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
BVAL	Input	ANY	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
BLK	Output	ANY	Indicación del área de memoria que se rellena con el contenido del área de origen.
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro BVAL

Al indicar una estructura a modo de parámetro de entrada, tenga en cuenta que la longitud de una estructura se adapta siempre a un número entero de bytes. Si se declara una estructura con un número impar de bytes, esta estructura necesita un byte adicional de espacio en memoria.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" permite programar conversiones explícitas. Los tipos de datos que se han de convertir se especifican en un cuadro de diálogo que se abre automáticamente al insertar la instrucción. Durante la ejecución, la instrucción lee el valor de origen y lo convierte al valor de destino indicado.

Encontrará más información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

Sintaxis

Para la instrucción "Convertir valor" se utiliza la siguiente sintaxis:

SCL

```
<Valor de destino> := <función de conversión> (<valor de origen>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Valor de origen>	Input, constante	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres	Valor que se convierte.
<Función de conversión>	-	-	Función que especifica los tipos de datos que se han de convertir.
<Valor de destino>	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, fecha y hora, secuencias de caracteres	Resultado de la conversión

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_INT" := REAL_TO_INT("Tag_REAL");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Tipo de datos	Valor
Tag_REAL	REAL	20.56
Tag_INT	INT	21

En la conversión, el valor del operando "Tag_REAL" se redondea al siguiente entero y se deposita en el operando "Tag_INT".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Conversión explícita (Página 2522)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor de la entrada IN al siguiente número entero. La instrucción interpreta el valor de la entrada IN como número en coma flotante y lo convierte a un número entero o a un número en coma flotante. Si el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par.

Sintaxis

Para la instrucción "Redondear número" se utiliza la siguiente sintaxis:

```
SCL
ROUND(<Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<Expresión>	Input	REAL	Valor de entrada que se debe redondear.
Valor de función		DINT	Resultado del redondeo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := ROUND("Tag_Value");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	1.50000000	-1.50000000
Tag_Result	2	-2

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor al siguiente número entero superior. La instrucción interpreta el valor de entrada como número en coma flotante y lo convierte al número entero superior más próximo. El valor de la función puede ser mayor o igual al valor de entrada.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero superior" se utiliza la siguiente sintaxis:

```
SCL
CEIL(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<expresión>	Input	REAL	Valor de entrada
Valor de función		DINT	Valor de entrada redondeado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := CEIL("Tag_Value");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	0,5	-0,5
Tag_Result	1	0

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor al siguiente número entero inferior. La instrucción interpreta el valor de entrada como número en coma flotante y lo convierte al número entero inferior más próximo. El valor de función puede ser menor o igual al valor de entrada.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero inferior" se utiliza la siguiente sintaxis:

```
SCL
FLOOR(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<expresión>	Input	REAL	Valor de entrada
Valor de función		DINT	Valor de entrada redondeado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := FLOOR("Tag_Value");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	0.5	-0.5
Tag_Result	0	-1

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un valor entero no redondeado a partir del valor de entrada. La instrucción selecciona solo la parte entera del valor de entrada y la devuelve sin decimales como valor de función.

Sintaxis

Para la instrucción "Truncar a entero" se utiliza la siguiente sintaxis:

```
SCL
TRUNC(<Expresión>)
TRUNC_<tipo de datos>(<expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
<expresión>	Input	REAL	Valor de entrada
_<tipo de datos>		Enteros Valor predeterminado: DINT	Tipo de datos del valor de función: • Si desea utilizar el tipo de datos predeterminado, no es necesario indicar nada. • Si desea utilizar otro tipo de datos admisible, hay que indicarlo explícitamente.
Valor de función		Enteros	Parte entera del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result1" := TRUNC("Tag_Value1");
"Tag_Result2" := TRUNC("Tag_Value2"+"Tag_Value3");
"Tag_Result3" := TRUNC_REAL("Tag_Value4");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_Value1	-1.5
Tag_Result1	-1
Tag_Value2	2.1
Tag_Value3	3.2
Tag_Result2	5
Tag_Result3	2.0
Tag_Value4	2.4

Consulte también

Operadores y su jerarquía (Página 2793)

Expresiones (Página 2786)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. El límite inferior y superior del rango en el que se escala el valor de entrada se especifican con los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

SCL

```
SCALE(IN := <Expresión>,
      HI_LIM := <Operando>,
      LO_LIM := <Operando>,
      BIPOLAR := <Operando>,
      OUT => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	INT	Valor de entrada que se escala.
HI_LIM	Input	REAL	Valor límite superior
LO_LIM	Input	REAL	Valor límite inferior

Parámetro	Declaración	Tipo de datos	Descripción
BIPOLAR	Input	BOOL	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	Resultado de la instrucción
Valor de función (RET_VAL)		WORD	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que 27 648 o menor que 0 (unipolar) o -27 648 (bipolar).
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
"Tag_ErrorCode" := SCALE(IN := "Tag_InputValue",
                         HI_LIM := "Tag_HighLimit"
                         LO_LIM := "Tag_LowLimit"
                         BIPOLAR := "Tag_Bipolar",
                         OUT => "Tag_Result");
```

La información de error de la instrucción se devuelve en el operando "Tag_ErrorCode" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	50.03978588
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" permite desescalar el número en coma flotante del parámetro IN en unidades físicas comprendidas entre un límite inferior y uno superior, y convertirlo en un número entero. El límite inferior y superior del rango en el que se desescala el valor de entrada se especifican con los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": Se presupone que el valor del parámetro IN es unipolar y que está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Sintaxis

Para la instrucción "Desescalar" se utiliza la siguiente sintaxis:

```
SCL
UNSCALE(IN := <Expresión>,
        HI_LIM := <Operando>,
        LO_LIM := <Operando>,
        BIPOLAR := <Operando>,
        OUT => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	REAL	Valor de entrada que se desescala en un valor entero.
HI_LIM	Input	REAL	Valor límite superior
LO_LIM	Input	REAL	Valor límite inferior
BIPOLAR	Input	BOOL	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	INT	Resultado de la instrucción
Valor de función (RET_VAL)		WORD	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_ErrorCode" := UNSCALE(IN := "Tag_InputValue",
                           HI_LIM := "Tag_HighLimit"
                           LO_LIM := "Tag_LowLimit"
                           BIPOLAR := "Tag_Bipolar",
                           OUT => "Tag_Result");
```

La información de error de la instrucción se devuelve en el operando "Tag_ErrorCode" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
OUT	Tag_Result	22
RET_VAL	Tag_ErrorCode	W#16#0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Control del programa

IF: Ejecución condicional

Descripción

La instrucción "Ejecución condicional" permite ramificar el flujo del programa en función de una condición. La condición es una expresión con un valor booleano (TRUE o FALSE). Como condición se pueden indicar expresiones lógicas o expresiones de comparación.

Al ejecutar la instrucción, se evalúan las expresiones indicadas. Si el valor de una expresión es TRUE, la condición se considera cumplida; en el caso de FALSE se considera que la condición no se cumple.

Sintaxis

Dependiendo del tipo de ramificación se pueden programar las siguientes formas de la instrucción:

- Ramificación con IF:

SCL

```
IF <Condición> THEN <Instrucciones>
END_IF;
```

Si se cumple la condición, se ejecutan las instrucciones que están programadas después de THEN. Si la condición no se cumple, el procesamiento del programa continúa con la siguiente instrucción después de END_IF.

- Ramificación con IF y ELSE:

SCL

```
IF <Condición> THEN <Instrucciones1>
ELSE <Instrucciones0>
END_IF;
```

Si se cumple la condición, se ejecutan las instrucciones que están programadas después de THEN. Si no se cumple la condición, se ejecutan las instrucciones que están programadas después de ELSE. Después, el procesamiento del programa continúa con la siguiente instrucción después de END_IF.

- Ramificación con IF, ELSIF y ELSE:

SCL

```
IF <Condición1> THEN <Instrucciones1>
ELSIF <Condición2> THEN <Instrucción2>
ELSE <Instrucciones0>
END_IF;
```

Si se cumple la primera condición (<condición1>), se ejecutan las instrucciones (<instrucciones1>) después de THEN. Después de ejecutar las instrucciones, el procesamiento del programa continúa después de END_IF.

Si no se cumple la primera condición, se comprueba la segunda (<condición2>). Si se cumple la segunda condición (<condición2>), se ejecutan las instrucciones (<instrucciones2>) después de THEN. Después de ejecutar las instrucciones, el procesamiento del programa continúa después de END_IF.

Si no se cumple ninguna condición, se ejecutan las instrucciones (<instrucciones0> después de ELSE y, a continuación, continúa el procesamiento del programa después de END_IF.

En una instrucción IF se pueden anidar tantas combinaciones de ELSIF y THEN como se desee. La programación de una rama ELSE es opcional.

La sintaxis de la instrucción IF se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Condición>	BOOL	Expresión que se evalúa.
<Instrucciones>	-	Instrucciones que se ejecutan si se cumple la condición. Una excepción son las instrucciones que están programadas después de ELSE. Estas se ejecutan si no se cumple ninguna condición dentro del bucle.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF "Tag_1" = 1
THEN "Tag_Value" := 10;
ELSIF "Tag_2" = 1
THEN "Tag_Value" := 20;
ELSIF "Tag_3" = 1
THEN "Tag_Value" := 30;
ELSE "Tag_Value" := 0;
END_IF;
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_1	1	0	0	0
Tag_2	0	1	0	0
Tag_3	0	0	1	0
Tag_Value	10	20	30	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

CASE: Estructura selectiva múltiple

Descripción

La instrucción "Estructura selectiva múltiple" permite procesar una de varias secuencias de instrucciones en función del valor de una expresión numérica.

El valor de la expresión ha de ser un entero. Al ejecutar la instrucción se compara el valor de la expresión con los valores de varias constantes. Si el valor de la expresión coincide con el de una constante, se ejecutan las instrucciones que están programadas directamente después de esta constante. Las constantes pueden adoptar los siguientes valores:

- un número entero (p. ej., 5)
- un rango de números enteros (p. ej. 15 .. 20)
- una enumeración de números enteros y rangos (p. ej., 10, 11, 15 .. 20)

Sintaxis

Para la instrucción "Estructura selectiva múltiple" se utiliza la siguiente sintaxis:

SCL

```
CASE <Expresión> OF
<Constante1>: <Instrucciones1>
<Constante2>: <Instrucciones2>
<ConstanteX>: <InstruccionesX>; // X >= 3
ELSE <Instrucciones0>
END_CASE;
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Expresión>	Enteros	Valor que se compara con los valores de constantes programados.
<Constante>	Enteros	Valores de constantes que forman la condición para ejecutar una secuencia de instrucciones. Las constantes pueden adoptar los siguientes valores: • un número entero (p. ej., 5) • un rango de números enteros (p. ej. 15 .. 20) • una enumeración de números enteros y rangos (p. ej., 10, 11, 15 .. 20)
<Instrucción>	-	Instrucciones cualesquiera que se ejecutan si el valor de la expresión coincide con el valor de una constante. Una excepción son las instrucciones que están programadas después de ELSE. Estas se ejecutan cuando los valores no coinciden.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Si el valor de la expresión coincide con el valor de la primera constante (<Constante1>), se ejecutan las instrucciones (<instrucciones1>) que están programadas directamente después de la primera constante. Luego, el procesamiento del programa continúa después de END_CASE.

Si el valor de la expresión no coindice con el valor de la primera constante (<Constante1), este se compara con el valor de la constante que está programada a continuación. De esta manera la instrucción CASE se ejecuta hasta que coincidan los valores. Si el valor de la expresión no se corresponde con ninguno de los valores programados de las constantes, se ejecutan las instrucciones (<instrucciones0>) que están programadas después de ELSE. ELSE es una parte opcional de la sintaxis y se puede omitir.

La instrucción CASE también se puede anidar sustituyendo un bloque de instrucciones por CASE. END_CASE cierra la instrucción CASE.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
CASE "Tag_Value" OF
     0 :
     "Tag_1" := 1;
     1,3,5 :
     "Tag_2" :=1;
     6..10 :
     "Tag_3" := 1;
     16,17,20..25 :
     "Tag_4" := 1;
ELSE "Tag_5" := 1;
END_CASE;
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valores				
Tag_Value	0	1, 3 , 5	6, 7, 8, 9, 10	16,17, 20, 21, 22, 23, 24, 25	2
Tag_1	1	-	-	-	-
Tag_2	-	1	-	-	-
Tag_3	-	-	1	-	-
Tag_4	-	-	-	1	-
Tag_5	-	-	-	-	1
1: El operando adopta el estado lógico "1". -: El estado lógico del operando permanece sin cambios.					

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

EXIT: Abandonar el bucle inmediatamente (Página 6466)

CONTINUE: Verificar condición de bucle (Página 6465)

FOR: Ejecutar en bucle contador

Descripción

La instrucción "Ejecutar en bucle contador" permite procesar un bucle mientras la variable de contaje se encuentre dentro del rango de valores indicado.

Los bucles también se pueden anidar. Dentro de un bucle se pueden programar otros bucles con otras variables de contaje.

La ejecución actual de un bucle se puede finalizar con la instrucción "Verificar condición de bucle" (CONTINUE). La instrucción "Abandonar el bucle inmediatamente" (EXIT) permite finalizar el procesamiento de todo el bucle. Encontrará más información al respecto en "Consulte también".

Valores límite para instrucciones FOR

Para programar instrucciones FOR "seguras" que no se ejecuten de manera infinita, tenga en cuenta las siguientes reglas y los siguientes valores límite:

Regla

```
FOR ii := principio TO fin BY paso DO
```

Si...	... entonces	Observación
principio < fin	fin < (PMAX - paso)	La variable de contaje ii cuenta en sentido ascendente
principio > fin Y paso < O	fin > (NMAX - paso)	La variable de contaje ii cuenta en sentido descendente

Valores límite

Para los tipos de datos posibles son válidos diferentes valores límite:

Tipo de datos	PMAX	NMAX
ii del tipo INT	32767	-32768
ii del tipo DINT	2147483647	-2147483648

Sintaxis

Para la instrucción "Ejecutar en bucle contador" se utiliza la siguiente sintaxis:

SCL

```
FOR <Variable de contaje> := <Valor inicial> TO <Valor final> BY
<Incremento> DO <Instrucciones>
END_FOR;
```

La sintaxis de la instrucción FOR se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Variable de contaje>	INT, DINT	Operando, cuyo valor se evalúa durante el procesamiento del bucle. El tipo de datos de la variable de contaje determina el tipo de datos de los parámetros restantes.
<Valor inicial>	INT, DINT	Expresión, cuyo valor se asigna a la variable de contaje al comienzo del procesamiento del bucle.
<Valor final>	INT, DINT	Expresión, cuyo valor determina la última ejecución del bucle. Antes de cada ejecución del bucle se comprueba qué valor tiene la variable de contaje: • Valor final no alcanzado: Se ejecutarán las instrucciones según DO • Valor final alcanzado: El bucle FOR será ejecutado una última vez • Valor final sobrepasado: El bucle FOR será finalizado No se puede modificar el valor final durante la ejecución de la instrucción.
<Incremento>	INT, DINT	Expresión, en cuyo valor aumenta (incremento positivo) o disminuye (incremento negativo) la variable de contaje después de cada ejecución del bucle. La indicación del incremento es opcional. Si no se ha indicado ningún incremento, el valor de la variable de contaje aumenta en 1 después de cada ejecución del bucle. No se puede modificar el incremento durante la ejecución de la instrucción.
<Instrucciones>	-	Instrucciones que se ejecutan durante cada ejecución del bucle mientras el valor de la variable de control se encuentre en el rango de valores. El rango de valores se especifica mediante los valores inicial y final.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
FOR i := 2 TO 8 BY 2
DO "a_array[i] := "Tag_Value"*"b_array[i]";
END_FOR;
```

El operando "Tag_Value" se multiplica por los elementos (2, 4, 6, 8) de las variables ARRAY "b_array". El resultado se lee en los elementos (2, 4, 6, 8) de las variables ARRAY "a_array".

Consulte también

CONTINUE: Verificar condición de bucle (Página 4445)

EXIT: Abandonar el bucle inmediatamente (Página 4446)

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

WHILE: Ejecutar cuando se cumpla la condición

Descripción

La instrucción "Ejecutar cuando se cumpla la condición" permite procesar un bucle mientras se cumpla una condición de ejecución. La condición es una expresión con un valor booleano (TRUE o FALSE). Como condición se pueden indicar expresiones lógicas o expresiones de comparación.

Al ejecutar la instrucción, se evalúan las expresiones indicadas. Si el valor de una expresión es TRUE, la condición se considera cumplida; en el caso de FALSE se considera que la condición no se cumple.

Los bucles también se pueden anidar. Dentro de un bucle se pueden programar otros bucles con otras variables de contaje.

La ejecución actual de un bucle se puede finalizar con la instrucción "Verificar condición de bucle" (CONTINUE). La instrucción "Abandonar el bucle inmediatamente" (EXIT) permite finalizar el procesamiento de todo el bucle. Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Ejecutar cuando se cumpla la condición" se utiliza la siguiente sintaxis:

SCL

```
WHILE <Condición> DO <Instrucciones>
END_WHILE;
```

La sintaxis de la instrucción WHILE se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Condición>	BOOL	Expresión que se evalúa antes de cada ejecución del bucle.
<Instrucciones>	-	Instrucciones que se ejecutan si se cumple la condición. Si no se cumple la condición, el procesamiento del programa continúa después de END_WHILE.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
WHILE
      "Tag_Value1" <> "Tag_Value2"
      DO "Tag_Result"
```

```
SCL
      := "Tag_Input";
END_WHILE;
```

Mientras difieran los valores de los operandos "Tag_Value1" y "Tag_Value2", al operando "Tag_Result" se le asignará el valor del operando "Tag_Input".

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

CONTINUE: Verificar condición de bucle (Página 6465)

EXIT: Abandonar el bucle inmediatamente (Página 6466)

REPEAT: Ejecutar cuando no se cumpla la condición

Descripción

La instrucción "Ejecutar cuando no se cumpla la condición" permite procesar un bucle mientras no se cumpla una condición de interrupción. La condición es una expresión con un valor booleano (TRUE o FALSE). Como condición se pueden indicar expresiones lógicas o expresiones de comparación.

Al ejecutar la instrucción, se evalúan las expresiones indicadas. Si el valor de una expresión es TRUE, la condición se considera cumplida; en el caso de FALSE se considera que la condición no se cumple.

Las instrucciones se ejecutan una vez aunque se cumpla la condición de interrupción.

Los bucles también se pueden anidar. Dentro de un bucle se pueden programar otros bucles con otras variables de contaje.

La ejecución actual de un bucle se puede finalizar con la instrucción "Verificar condición de bucle" (CONTINUE). La instrucción "Abandonar el bucle inmediatamente" (EXIT) permite finalizar el procesamiento de todo el bucle. Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Ejecutar cuando no se cumpla la condición" se utiliza la siguiente sintaxis:

```
SCL
REPEAT <Instrucciones>
UNTIL <Condición> END_REPEAT;
```

La sintaxis de la instrucción REPEAT se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<Instrucciones>	-	Instrucciones que se ejecutan mientras la condición programada tenga el valor FALSE. Las instrucciones se ejecutan una vez aunque se cumpla la condición de interrupción.
<Condición>	BOOL	Expresión que se evalúa después de cada ejecución del bucle. Si la expresión tiene el valor FALSE, el bucle se procesa de nuevo. Si la expresión tiene el valor TRUE, el procesamiento del programa continúa después de END_REPEAT.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
REPEAT "Tag_Result"
        := "Tag_Value";
UNTIL "Tag_Error"
END_REPEAT;
```

Mientras el valor del operando "Tag_Error" tenga el estado lógico "0", se le asignará al operando "Tag_Result" el valor del operando "Tag_Value".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

CONTINUE: Verificar condición de bucle

Descripción

La instrucción "Verificar condición de bucle" permite finalizar la ejecución actual de un bucle FOR, WHILE o REPEAT.

Después de ejecutar la instrucción, se evalúan de nuevo las condiciones para continuar el bucle. La instrucción tiene efecto sobre el bucle que incluye directamente la instrucción.

Sintaxis

Para la instrucción "Verificar condición de bucle" se utiliza la siguiente sintaxis:

SCL

```
CONTINUE;
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
FOR i
     := 1 TO 15 BY 2 DO
     IF (i < 5) THEN
     CONTINUE;
END_IF;
      "DB10".Test[i] := 1;
END_FOR;
```

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Si se cumple la condición i < 5, no se procesa la siguiente asignación de valor ("DB10".Test[i] := 1). La variable de contaje (i) se incrementa en "2" y se comprueba si su valor actual se encuentra en el rango de valores programados. Si la variable de contaje se encuentra en el rango de valores, se evalúa de nuevo una condición IF.

Si no se cumple la condición i < 5, se procesa la siguiente asignación de valor ("DB10".Test[i] := 1) y se inicia una nueva ejecución del bucle. La variable de control también se aumenta con el incremento "2" y se comprueba.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

EXIT: Abandonar el bucle inmediatamente (Página 6466)

EXIT: Abandonar el bucle inmediatamente

Descripción

La instrucción "Abandonar el bucle inmediatamente" permite interrumpir en cualquier punto la ejecución de un bucle FOR, WHILE o REPEAT independientemente de las condiciones. El procesamiento del programa continúa después finalizar el bucle (END_FOR, END_WHILE, END_REPEAT).

La instrucción tiene efecto sobre el bucle que incluye directamente la instrucción.

Sintaxis

Para la instrucción "Abandonar el bucle inmediatamente" se utiliza la siguiente sintaxis:

```
SCL
EXIT;
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
FOR i := 15 TO 1 BY -2 DO
IF (i < 5)
THEN EXIT;
END_IF;
"DB10".Test[i] := 1;
END_FOR;
```

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Si se cumple la condición i < 5, se interrumpe el procesamiento del bucle. El procesamiento continúa después de END_FOR.

Si no se cumple la condición i < 5, se procesa la siguiente asignación de valor ("DB10".Test[i] := 1) y se inicia una nueva ejecución del bucle. La variable de contaje (i) se decrementa en "-2" y se comprueba si su valor actual se encuentra en el rango de valores programados. Si la variable de control (i) se encuentra en el rango de valores, la condición IF se evalúa de nuevo.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

CONTINUE: Verificar condición de bucle (Página 6465)

GOTO: Saltar

Descripción

La instrucción "Saltar" permite continuar el procesamiento del programa en un punto marcado con una etiqueta.

La etiqueta y la instrucción "Saltar" deben estar en el mismo bloque. La denominación de la etiqueta debe ser unívoca en un bloque. A toda etiqueta pueden acceder distintas instrucciones de salto.

No se permite saltar a un bucle desde "fuera", pero sí es posible saltar desde un bucle hacia "fuera".

Sintaxis

Para la instrucción "Saltar" se utiliza la siguiente sintaxis:

SCL

```
GOTO <Etiqueta>
...
<Etiqueta>: <Instrucciones>
```

La sintaxis de la instrucción GOTO se compone de las siguientes partes:

Parámetro	Tipo de datos	Descripción
<etiqueta>	-	Etiqueta a la que se salta
<Instrucciones>	-	Instrucciones que se ejecutan después del salto.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
CASE "Tag_Value" OF
1 : GOTO MyLABEL1;
2 : GOTO MyLABEL2;
3 : GOTO MyLABEL3;
ELSE GOTO MyLABEL4;
END_CASE;
MyLABEL1: "Tag_1" := 1;
MyLABEL2: "Tag_2" := 1;
MyLABEL3: "Tag_3" := 1;
MyLABEL4: "Tag_4" := 1;
```

Dependiendo del valor del operando "Tag_Value", el procesamiento del programa se continúa en el punto marcado por la etiqueta correspondiente. Por ejemplo, si el operando "Tag_Value" tiene el valor "2", el procesamiento del programa continúa a partir de la etiqueta "MyLABEL2". En este caso, la línea del programa marcada con la etiqueta "MyLABEL1" se omite.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

RETURN: Abandonar el bloque

Descripción

La instrucción "Abandonar el bloque" permite finalizar el procesamiento del programa en el bloque que se está ejecutando y continuar en el bloque invocante.

En el fin del bloque se puede omitir la instrucción.

Sintaxis

Para la instrucción "Abandonar el bloque" se utiliza la siguiente sintaxis:

SCL

```
RETURN;
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
IF "Tag_Error" <>0 THEN RETURN;
END_IF;
```

Si el estado lógico del operando "Tag_Error" no es igual a cero, el procesamiento del programa se finaliza en el bloque que se está procesando.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

(*...*): Insertar sección de comentario

Descripción

La instrucción "Insertar sección de comentario" permite insertar una sección de comentario. El texto encerrado entre paréntesis "(*...*)" se trata como un comentario.

Sintaxis

Para la instrucción "Insertar sección de comentario" se utiliza la siguiente sintaxis:

SCL

```
(*...*)
```

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
(*Esta es una sección de comentario.*)
```

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Control del tiempo de ejecución

COMPRESS: Comprimir la memoria de la CPU

Descripción

La instrucción "Comprimir la memoria de la CPU" permite comprimir la memoria de trabajo de la CPU o la memoria de carga RAM. Los huecos creados por haber borrado o cargado bloques, se rellenan juntando los bloques existentes.

Con la llamada de la instrucción "Comprimir la memoria de la CPU" se lanza el proceso de compresión. El proceso de compresión se reparte entre varios ciclos del programa. La vigilancia del proceso de compresión se realiza mediante los valores de los distintos parámetros. Si el parámetro BUSY de la instrucción tiene el valor "1", significa que el proceso de compresión se está ejecutando todavía. El valor "1" del parámetro DONE indica que ha finalizado el proceso de compresión. La información de error se puede consultar en el parámetro RET_VAL.

Si se está ejecutando un proceso de compresión lanzado externamente, la llamada de la instrucción "Comprimir la memoria de la CPU" origina un error.

Sintaxis

Para la instrucción "Comprimir la memoria de la CPU" se utiliza la siguiente sintaxis:

```
SCL
COMPRESS(BUSY => <Operando>, DONE => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
BUSY	Output	BOOL	Información de si está activo el proceso de compresión.
DONE	Output	BOOL	Información de si ha finalizado el proceso de compresión.
Valor de función (RET_VAL)		INT	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error El estado del proceso de compresión puede consultarse en los parámetros BUSY y DONE.
8091	Ya se está ejecutando un proceso de compresión lanzado externamente.
8092	La instrucción no se puede ejecutar porque se cumple una de las condiciones siguientes: • La función "Borrar bloque" está activa. • Una función de PG está accediendo a uno de los bloques que se deben desplazar. • Ya se está ejecutando un proceso de compresión lanzado externamente. • La CPU H se encuentra en un proceso de acoplamiento o actualización.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU. Reiniciando el tiempo de vigilancia del ciclo es posible impedir que se produzcan errores o que la CPU pase a modo STOP.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" puede utilizarse en bloques de la clase de prioridad 1 (OBs de ciclo), así como en los bloques que se llaman en ellos.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

Nota

En caso de utilizar la instrucción "Reiniciar tiempo de vigilancia del ciclo", debe asegurarse de que no se produce un bucle sin fin en la ejecución cíclica del programa, es decir, en el OB1. De lo contrario, la CPU ya no alcanza el punto de control del ciclo. Como consecuencia no pueden ejecutarse ciertas funciones de la CPU (p. ej., la actualización de la memoria imagen de proceso).

Nota

Si se inicia la instrucción "Reiniciar tiempo de vigilancia del ciclo" dentro de un bucle LOOP más de 30 veces, la CPU pasa al estado operativo STOP debido a un error en tiempo de ejecución.

Sintaxis

Para la instrucción "Reiniciar tiempo de vigilancia del ciclo" se utiliza la siguiente sintaxis:

SCL

```
RE_TRIGR()
```

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" permite detener la ejecución del programa durante un periodo predeterminado. El periodo se indica en microsegundos en el parámetro WT de la instrucción.

Es posible programar tiempos de retardo de hasta 32 767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve la información de error.

Nota

En las CPUs S7-300 (a excepción de S7-318), el tiempo de retardo programado con la instrucción es un tiempo mínimo. Se prolonga con el tiempo de ejecución de las clases de prioridad que intervienen y con las cargas del sistema.

Sintaxis

Para la instrucción "Programar retardo" se utiliza la siguiente sintaxis:

SCL

```
WAIT(WT := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
WT	Input	INT	Tiempo de retardo en microsegundos (µs)

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos de una transición de RUN a STOP dependen de la configuración de la CPU.

Sintaxis

Para la instrucción "Finalizar programa" se utiliza la siguiente sintaxis:

SCL

```
STP()
```

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

PROTECT: Cambiar nivel de protección

Descripción

La instrucción "Cambiar nivel de protección" permite modificar el nivel de protección "1" configurado para la CPU.

El nivel de protección "1" parametrizado con la opción activada "Eliminable por contraseña" hace que sea necesario parametrizar una contraseña. Esta contraseña permite acceder en escritura a la CPU protegida contra escritura mediante la instrucción "Cambiar nivel de protección" (p. ej. cargar) durante el funcionamiento.

El nivel de protección se especifica en el parámetro MODE de la instrucción. El nivel de protección actual de la CPU se visualiza en la ventana de inspección en "Propiedades > Protección".

La tabla siguiente muestra una explicación de los niveles de protección:

Nivel de protección	Explicación
1	Se permiten todas las funciones de PG. En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
2	• El programa y la configuración de la CPU no se pueden modificar. El programa contenido en la CPU se puede leer. • Se permiten las funciones de control, observación y comunicación del proceso. • Se permiten todas las funciones de información. • Si conoce la contraseña válida, podrá desactivar la protección contra escritura ajustada. • En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
3	• Aun conociendo la contraseña válida, no podrá desactivar la protección contra lectura y escritura ajustada. • Si al llamar la instrucción "Cambiar nivel de protección" existe una conexión legitimada, la llamada de la instrucción no tendrá ningún efecto en esta conexión.

Si se ha configurado el nivel de protección "2" ó "3" para la CPU, la llamada de la instrucción "Cambiar nivel de protección" no tiene efecto alguno.

La efectividad del nivel de protección "2" o "3" una vez ejecutada la instrucción "Cambiar nivel de protección" puede consultarse en la lista parcial SZL con la ID de SZL W#16#0232 y el índice W#16#0004.

Nota

La instrucción "Cambiar nivel de protección" permite ajustar un nivel de protección más bajo que aquel ajustado con "Configurar hardware" de STEP 7. Supongamos que se ha configurado el nivel de protección 3 y a continuación se llama la instrucción "Cambiar nivel de protección" con el parámetro MODE = 12. Esto bloqueará la legitimación de la contraseña. Si la instrucción se llama nuevamente con el parámetro MODE = 0 o 1, el nivel de protección 3 se mantiene y será posible desactivar la protección de lectura y escritura introduciendo la contraseña válida.

Efectos de las transiciones de estado operativo en el nivel de protección

La tabla siguiente muestra qué efectos tienen las transiciones de estado operativo en el nivel de protección ajustado con la instrucción "Cambiar nivel de protección".

Acción	Efectos en el nivel de protección
• Selector de modo en "STOP" • Conexión (POWER ON) sin respaldo • Borrado total mediante el selector de modo • Cambios de configuración y/o del programa en el estado operativo "STOP" de la CPU S7-400	El nivel de protección se pone a "1". Se permiten todas las funciones de PG.
• Arranque en frío • Rearranque (en caliente)	El sistema operativo pone el nivel de protección a "1". Se permiten todas las funciones de PG. En caso necesario, llamando la instrucción "Cambiar nivel de protección" en el programa es posible ajustar el nivel de protección en "2" o "3".
Conexión (POWER ON) respaldada	No cambia
Transición de estado operativo RUN/ARRANQUE/PARADA -> STOP (debida a la llamada de la instrucción "Finalizar programa" (STP), a errores del programa sin el correspondiente OB de error o a una acción del usuario).	No cambia
S7-400: Rearranque por acción del usuario o automáticamente tras conexión (POWER ON)	No cambia

Sintaxis

Para la instrucción "Cambiar nivel de protección" se utiliza la siguiente sintaxis:

SCL

```
PROTECT(MODE := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
MODE	Input	WORD	Nivel de protección Valores posibles: • W#16#0000: Ajuste del nivel de protección "1" • W#16#0001: Ajuste del nivel de protección "2" con legitimación de la contraseña • W#16#000C: Ajuste del nivel de protección "3" sin legitimación de la contraseña
Valor de función (RET_VAL)		INT	Si ocurre un error durante la ejecución, se emite una información de error.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
8090	Valor no admisible en el parámetro MODE.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

CiR: Controlar proceso CiR

Descripción

La instrucción "Controlar proceso CiR" permite influir en el proceso CiR en el programa de usuario como se indica a continuación:

- Bloquear el proceso CiR. En este caso, se rechazará siempre la carga de una configuración modificada desde la programadora en la CPU. El bloqueo se aplica hasta que se reactive con la instrucción "Controlar proceso CiR".
- El proceso CiR se puede bloquear de forma condicionada, indicando un límite superior para el tiempo de sincronización CiR en el parámetro FRZ_TIME. En este caso, solo se permitirá cargar una configuración modificada desde la programadora en la CPU si el tiempo de sincronización CiR calculado en la CPU no excede el límite superior definido.

- Habilitar el proceso CiR. El límite superior del tiempo de sincronización CiR se establece al valor predeterminado de 1000 ms.
- Determina si el proceso CiR está habilitado o no. Si el proceso CiR está habilitado o lo está condicionadamente, el límite superior actual del tiempo de sincronización CiR se indica en el parámetro A_FT de la instrucción.

Nota

Durante el tiempo de sincronización CiR, las salidas están congeladas y las entradas no se evalúan.

Sintaxis

Para la instrucción "Controlar proceso CiR" se utiliza la siguiente sintaxis:

SCL

```
CIR(MODE := <Operando>,
    FRZ_TIME := <Operando>,
    A_FT => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
MODE	Input	BYTE	ID de la petición Los valores admisibles del parámetro MODE son: • 0: Función de información • 1: Habilitar proceso CiR • 2: Bloquear proceso CiR • 3: Bloquear proceso CiR de forma condicionada. El límite superior del tiempo de sincronización CiR se define en FRZ_TIME.
FRZ_TIME	Input	TIME	Límite superior del tiempo de sincronización CiR en milisegundos. Rango de valores admisible: 60 ... 2500 ms (ajuste predeterminado: 1000 ms) El parámetro FRZ_TIME solo es relevante si el parámetro MODE está al valor "3".
A_FT	Output	TIME	Límite superior válido actualmente del tiempo de sincronización CiR
Valor de función (RET_VAL)		INT	Información de error Si el parámetro MODE está a "0", el parámetro RET_VAL indica si el proceso CiR está habilitado o no.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor 1, 2 ó 3.
0001	El proceso CiR está habilitado. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
0002	El proceso CiR está bloqueado. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
0003	El proceso CiR está bloqueado de forma condicionada. Este código de error solo se puede emitir si el parámetro MODE está puesto al valor "0".
8001	La CPU no está lista para un proceso CiR.
8002	El valor del parámetro MODE no es admisible.
8003	El valor del parámetro FRZ_TIME no es admisible.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

La instrucción "Controlar proceso CiR" permite programar p. ej. que no se lance ningún proceso CiR en periodos en los que se requiera la capacidad máxima de la CPU para el procesamiento.

En este caso, la instrucción "Controlar proceso CiR" debe llamarse en el programa antes de comenzar el periodo de actividad aumentada del proceso y ejecutarse con el valor "2" (Bloquear proceso CiR) en el parámetro MODE. Con ello se bloquea el proceso CiR.

Para volver a habilitar el proceso CiR tras finalizar el periodo de actividad aumentada del proceso, la instrucción "Controlar proceso CiR" debe llamarse de nuevo y ejecutarse con el valor "1" en el parámetro MODE. Como alternativa, el parámetro MODE se puede poner al valor "3", con lo que el proceso CiR se bloquea de forma condicionada.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Cambiar formatos de visualización en el estado del programa (Página 3100)

Operaciones lógicas con palabras

DECO: Descodificar

Descripción

La instrucción "Descodificar" permite activar un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor del parámetro IN y activa el bit en el valor de salida cuya posición equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor del parámetro IN es mayor que 31, se ejecuta una instrucción modulo 32.

Sintaxis

Para la instrucción "Descodificar" se utiliza la siguiente sintaxis:

SCL

```
DECO(IN := <Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	WORD	Posición del bit que se activa en el valor de salida.
Valor de función		DWORD	Valor de salida actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

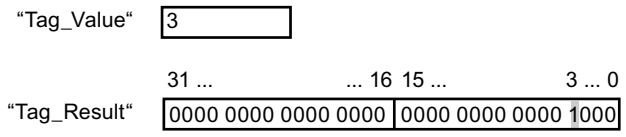
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := DECO(IN := "Tag_Value");
```

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



La instrucción lee el número "3" del valor del operando "Tag_Value" y activa el tercer bit del valor del operando "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Operadores y su jerarquía (Página 2793)

Expresiones (Página 2786)

ENCO: Codificar

Descripción

La instrucción "Codificar" permite leer el número del bit menos significativo que está activado en el valor de entrada y devolverlo como resultado.

Sintaxis

Para la instrucción "Codificar" se utiliza la siguiente sintaxis:

SCL

```
ENCO(IN := <Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	DWORD	Valor de entrada
Valor de función		INT	Número del bit que se lee en el valor de entrada.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

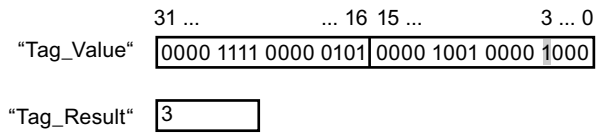
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ENCO(IN := "Tag_Value");
```

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



La instrucción lee el bit menos significativo que está activado en el operando "Tag_Value" y escribe la posición de bit "3" en el operando "Tag_Result". El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Operadores y su jerarquía (Página 2793)

Expresiones (Página 2786)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" selecciona, en función de un interruptor (parámetro G), uno de los parámetros IN0 o IN1 y devuelve su contenido como resultado. Si el parámetro G tiene el estado lógico "0", se copia el valor del parámetro IN0. Si el parámetro G tiene el estado lógico "1", se copia el valor del parámetro IN1 y se devuelve como valor de función.

Para poder ejecutar la instrucción, es imprescindible que las variables de todos los parámetros del tipo de datos sean de la misma clase.

Sintaxis

Para la instrucción "Seleccionar" se utiliza la siguiente sintaxis:

SCL

```
SEL(G := <Expresión>,
    IN0 := <Expresión>,
    IN1 := <Expresión>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
G	Input	BOOL	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE, DT	Primer valor de entrada

Parámetro	Declaración	Tipo de datos	Descripción
IN1	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE, DT	Segundo valor de entrada
Valor de función		Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE, DT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := SEL(G := "Tag_Value",
                    IN0 := "Tag_0",
                    IN1 := "Tag_1");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Value	0	1
Tag_0	W#16#0000	W#16#4C
Tag_1	W#16#FFFF	D#16#5E
Tag_Result	W#16#0000	D#16#5E

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Expresiones (Página 2786)

Operadores y su jerarquía (Página 2793)

Vista general de los tipos de datos válidos (Página 2321)

MUX: Multiplexar

Descripción

La instrucción "Multiplexar" copia el valor de un parámetro de entrada seleccionado y lo devuelve. Con el parámetro K se especifica el número del parámetro de entrada cuyo número

se copiará. La numeración comienza por IN0 y continúa en orden ascendente con cada nueva entrada. Se puede declarar un máximo de 32 parámetros de entrada.

Si el valor del parámetro K es mayor que el número de entradas y el parámetro INELSE no está activado, el valor de función de la instrucción no será válido y la salida de habilitación ENO se pone a "0".

Todas las variables parametrizadas deben ser del mismo tipo de datos. El valor de función adopta el tipo de datos más significativo.

Sintaxis

Para la instrucción "Multiplexar" se utiliza la siguiente sintaxis:

SCL

```
MUX(K := <Expresión>,
    IN0 := <Expresión>,
    IN1 := <Expresión>,
    INELSE := <Expresión>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Multiplexar":

Parámetro	Declaración	Tipo de datos	Descripción
K	Input	Enteros	Indica el parámetro cuyo contenido se transfiere. • Si K = 0 => parámetro IN0 • Si K = 1 => parámetro IN1, etc.
IN0	Input	Números binarios, enteros, números en coma flotante, temporizadores, secuencias de caracteres, TOD, DATE, DT	Primer valor de entrada
IN1	Input	Números binarios, enteros, números en coma flotante, temporizadores, secuencias de caracteres, TOD, DATE, DT	Segundo valor de entrada

Parámetro	Declaración	Tipo de datos	Descripción
INn	Input	Números binarios, enteros, números en coma flotante, temporizadores, secuencias de caracteres, TOD, DATE, DT	Valores de entrada opcionales
INELSE	Input	Números binarios, enteros, números en coma flotante, temporizadores, secuencias de caracteres, TOD, DATE, DT	Indica el valor que se copia si K <> n.
Valor de función		Números binarios, enteros, números en coma flotante, temporizadores, secuencias de caracteres, TOD, DATE, DT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := MUX(K := "Tag_Number",
                    IN0 := "Tag_1",
                    IN1 := "Tag_2",
                    INELSE := "Tag_3");
```

El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_Number	1	4
Tag_1	DW#16#00000000	DW#16#00000000
Tag_2	DW#16#003E4A7D	DW#16#003E4A7D
Tag_3	DW#16#FFFF0000	DW#16#FFFF0000
Tag_Result	DW#16#003E4A7D	DW#16#FFFF0000

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Operadores y su jerarquía (Página 2793)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

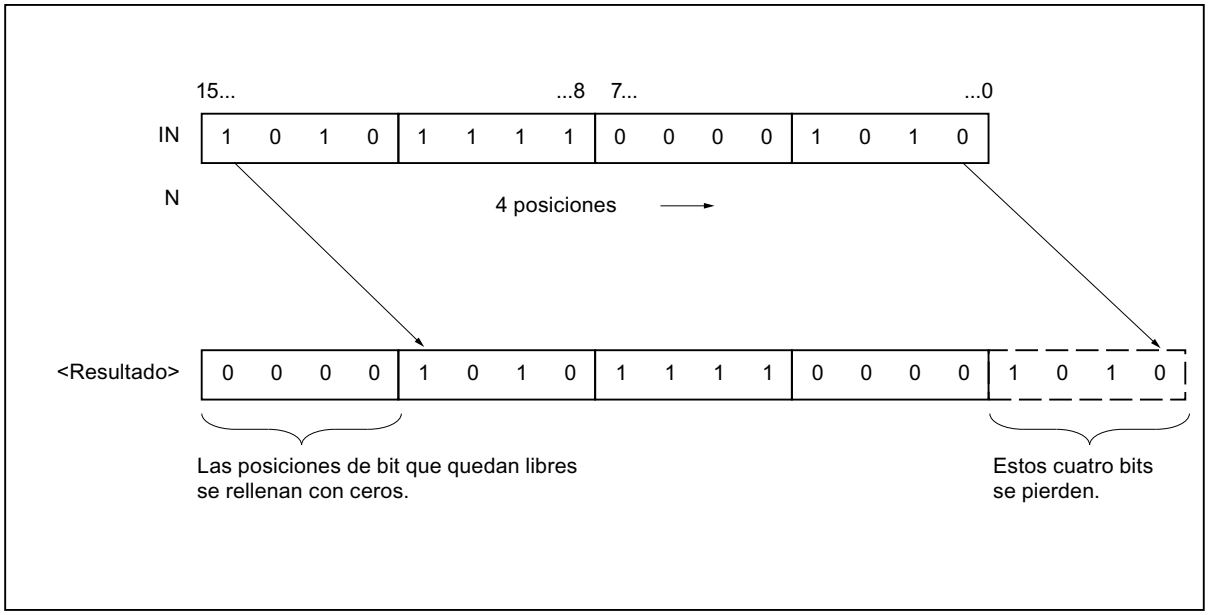
Con la instrucción "Desplazar a la derecha" se desplaza el contenido del parámetro IN de bit en bit hacia la derecha y se devuelve como valor de función. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", se devuelve como resultado el valor del parámetro IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del parámetro IN se desplaza hacia la derecha las posiciones de bit disponibles.

Los bits que quedan libres en el área izquierda del operando al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos INT se desplaza cuatro posiciones de bit a la derecha:



Sintaxis

Para la instrucción "Desplazar a la derecha" se utiliza la siguiente sintaxis:

SCL

```
SHR(IN := <Operando>,
```

```
SCL
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	Secuencias de bits	Valor que se desplaza.
N	Input	Enteros	Número de bits que se desplaza el valor (IN). El valor indicado en el parámetro N debe ser positivo.
Valor de función		Secuencias de bits	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := SHR(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	3
Valor de función	Tag_Result	0000 0111 1111 010 1

El valor del operando "Tag_Value" se desplaza tres posiciones de bit hacia la derecha. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SHL: Desplazar a la izquierda

Descripción

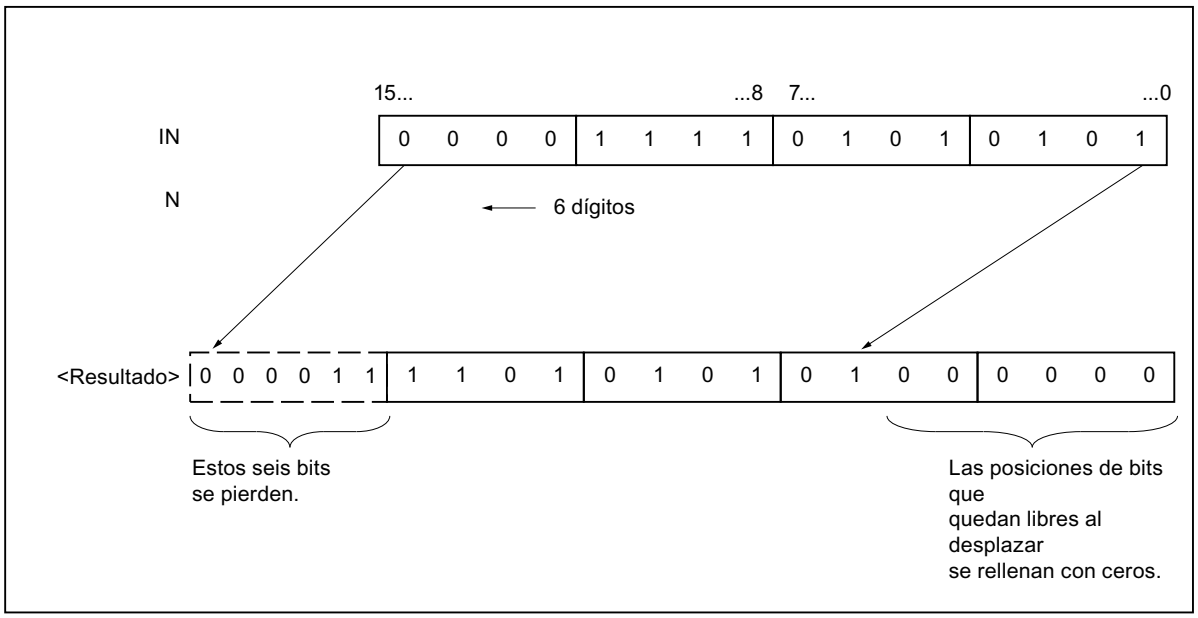
Con la instrucción "Desplazar a la izquierda" se desplaza el contenido del parámetro IN de bit en bit hacia la izquierda y se devuelve como valor de función. El parámetro N determina el número de bits que debe desplazarse el valor indicado.

Si el valor del parámetro N es "0", se devuelve como resultado el valor del parámetro IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del parámetro IN se desplaza hacia la izquierda las posiciones de bit disponibles.

Las posiciones que quedan libres al desplazar se rellenan con ceros.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Sintaxis

Para la instrucción "Desplazar a la izquierda" se utiliza la siguiente sintaxis:

SCL

```
SHL(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	Secuencias de bits	Valor que se desplaza
N	Input	Enteros	Número de bits que se desplaza el valor IN. El valor indicado en el parámetro N debe ser positivo.
Valor de función		Secuencias de bits	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := SHL(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	0011 1111 1010 1111
N	Tag_Number	4
Valor de función	Tag_Result	1111 1010 1111 0000

El valor del operando "Tag_Value" se desplaza cuatro posiciones de bit hacia la izquierda. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ROR: Rotar a la derecha

Descripción

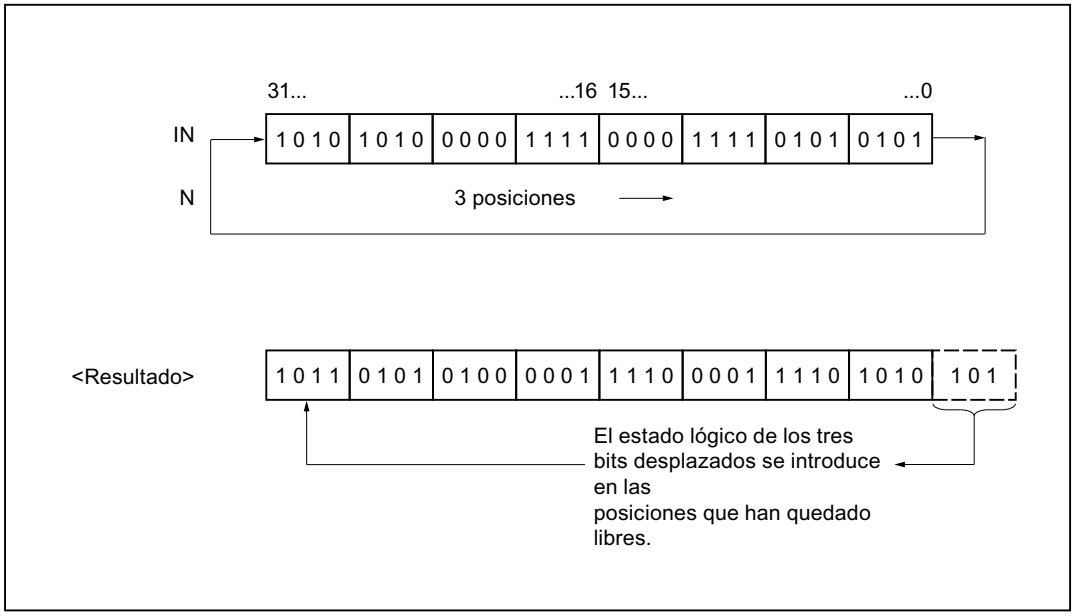
Con la instrucción "Rotar a la derecha", el contenido del parámetro IN rota de bit en bit hacia la derecha y se devuelve como valor de función. El parámetro N determina el número de bits

que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", se devuelve como resultado el valor de la entrada IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Sintaxis

Para la instrucción "Rotar a la derecha" se utiliza la siguiente sintaxis:

SCL

```
ROR(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	Secuencias de bits	Valor que se rota
N	Input	Enteros	Número de bits que rota el valor IN. El valor indicado en el parámetro N debe ser positivo.
Valor de función		Secuencias de bits	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ROR(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1010 1010 0000 1111 0000 1111 0101 0101
N	Tag_Number	5
Valor de función	Tag_Result	1010 1101 0101 0000 0111 1000 0111 1010

El contenido del operando "Tag_Value" se rota cinco posiciones de bit hacia la derecha. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

ROL: Rotar a la izquierda

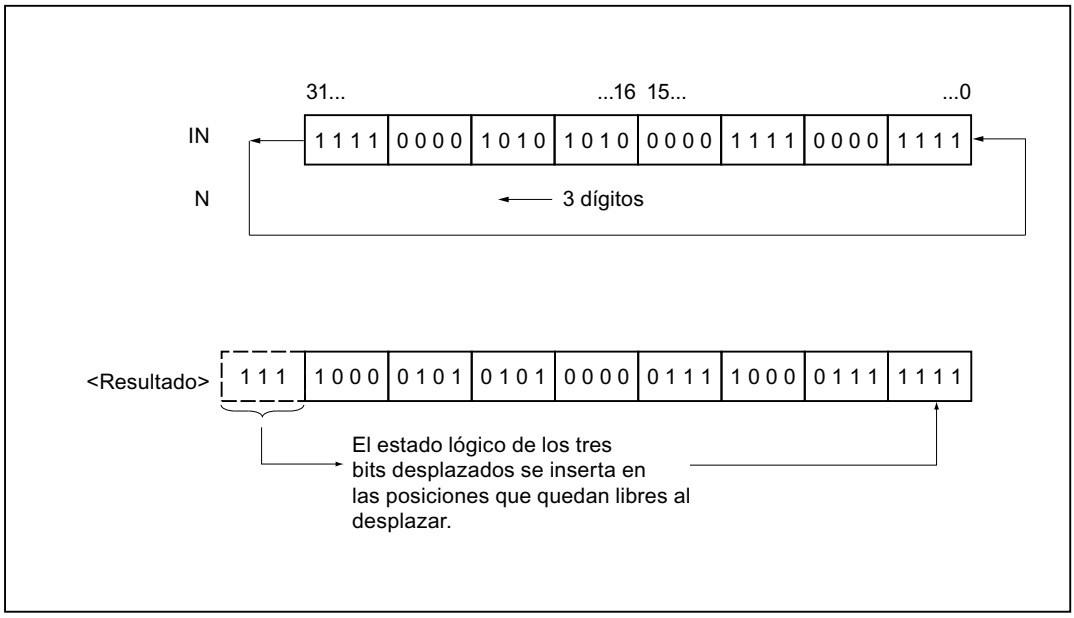
Descripción

Con la instrucción "Rotar a la izquierda", el contenido del parámetro IN rota de bit en bit hacia la izquierda y se devuelve como valor de función. El parámetro N determina el número de bits que debe rotarse el valor indicado. Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

Si el valor del parámetro N es "0", se devuelve como resultado el valor de la entrada IN.

Si el valor del parámetro N es mayor que el número de bits disponibles, el valor del operando en la entrada IN rota de todos modos el número indicado de posiciones de bit.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Sintaxis

Para la instrucción "Rotar a la izquierda" se utiliza la siguiente sintaxis:

SCL

```
ROL(IN := <Operando>,
    N := <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	Secuencias de bits	Valor que se rota
N	Input	Enteros	Número de bits que rota el valor IN. El valor indicado en el parámetro N debe ser positivo.
Valor de función		Secuencias de bits	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
"Tag_Result" := ROL(IN := "Tag_Value",
                    N := "Tag_Number");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN	Tag_Value	1111 0000 1010 1010 0000 1111 0000 1111
N	Tag_Number	5
Valor de función	Tag_Result	0001 0101 0100 0001 1110 0001 1111 1110

El contenido del operando "Tag_Value" se rota cinco posiciones de bit hacia la izquierda. El resultado de la instrucción se devuelve en el operando "Tag_Result" como valor de función.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Otras instrucciones

SET: Activar área por bits

Descripción

La instrucción "Activar área por bits" permite poner a "1" el estado lógico de los bits en un área definida. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben activarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

Sintaxis

Para la instrucción "Activar área por bits" se utiliza la siguiente sintaxis:

SCL

```
SET(S_BIT := <Operando>,
    N := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
S_BIT	Input	POINTER	Apunta al primer bit del área cuyos bits se activan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	Número de bits que deben activarse

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
SET(S_BIT := M0.0,
    N := 10);
```

En el ejemplo, la instrucción activa 10 bits en el área de M0.0 a M1.1.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SETP: Activar mapa de bits en el área de periferia

Descripción

La instrucción "Activar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "1". Paralelamente a los bits de periferia, también se activan los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe activarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no se ha conectado periferia para una parte del mapa de bits seleccionado, la instrucción intentará activar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se activa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas. Esto puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.
- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "1".

La llamada de la instrucción "Activar mapa de bits en el área de periferia" no tiene efecto alguno si el parámetro N devuelve el valor "0".

Sintaxis

Para la instrucción "Activar mapa de bits en el área de periferia" se utiliza la siguiente sintaxis:

SCL

```
SETP(N := <Operando>,
     SA => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
N	Input	INT	Número de bits que deben activarse
SA	Output	POINTER	Puntero hacia el primer bit que debe activarse
Valor de función (RET_VAL)		INT	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Error" := SETP(N := 8,
                    SA => P#10.0);
```

En el ejemplo, la instrucción activa 8 bits en el área de periferia conforme a las salidas Q10.0 a Q10.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SETI: Activar área por bytes

Descripción

La instrucción "Activar área por bytes" permite poner a "1" el estado lógico de los bits en un área definida de bytes. El punto inicial del área se define en el parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben activarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y

palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte. Por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas (Q) se pone a "0".

La llamada de la instrucción "Activar área por bytes" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa (P).
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.

Sintaxis

Para la instrucción "Activar área por bytes" se utiliza la siguiente sintaxis:

SCL

```
SETI(S_BYTE := <Operando>,
     N := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
S_BYTE	Input	POINTER	Señala al primer byte del área cuyos bits se activan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	Tamaño del área que debe activarse El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

SCL

```
SETI(S_BYTE := byte_array[100],
```

```
SCL
      N := 16);
```

En el ejemplo, la instrucción activa 2 bytes (16 bits) en el área de 100.0 a 101.7 del área por bytes.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

RESET: Desactivar área por bits

Descripción

La instrucción "Desactivar área por bits" permite poner a "0" el estado lógico de los bits en un área definida. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben desactivarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

Sintaxis

Para la instrucción "Desactivar área por bits" se utiliza la siguiente sintaxis:

```
SCL
RESET(S_BIT := <Operando>,
      N := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
S_BIT	Input	POINTER	Señala al primer bit del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	Número de bits que deben desactivarse.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
RESET(S_BIT := M0.0,
      N := 10);
```

En el ejemplo, la instrucción desactiva 10 bits en el área de M0.0 a M1.1.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

RESETP: Desactivar mapa de bits en el área de periferia

Descripción

La instrucción "Desactivar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "0". Simultáneamente con los bits de periferia se desactivan también los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe desactivarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no hay periferia enchufada para una parte del mapa de bits seleccionado, la instrucción intentará desactivar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se desactiva por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor de los bits correspondientes en la memoria imagen de proceso de las salidas.

  ⚠ ADVERTENCIA

  Ajuste de los bits

  El ajuste de los bits al valor de los bits correspondientes en la memoria imagen de proceso de las salidas puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "0".

La llamada de la instrucción "Desactivar mapa de bits en el área de periferia" no tiene efecto alguno si el parámetro N devuelve el valor "0".

Sintaxis

Para la instrucción "Desactivar mapa de bits en el área de periferia" se utiliza la siguiente sintaxis:

```
SCL
RESETP(N := <Operando>,
       SA => <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
N	Input	INT	Número de bits que deben desactivarse.
SA	Output	POINTER	Puntero al primer bit que debe desactivarse.
Valor de función (RET_VAL)		INT	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Error" := RESETP(N := 8,
                      SA => P#10.0);
```

En el ejemplo, la instrucción desactiva 8 bits en el área de periferia conforme a las salidas Q10.0 a Q10.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Evaluación de errores con el parámetro de salida RET_VAL (Página 3279)

Cambiar formatos de visualización en el estado del programa (Página 3100)

RESETI: Desactivar área por bytes

Descripción

La instrucción "Desactivar área por bytes" permite poner a "0" el estado lógico de los bits en un área definida de bytes. El punto inicial del área se define mediante el puntero del parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben desactivarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte. Por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas (Q) se pone a "0".

La llamada de la instrucción "Desactivar área por bytes" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa (P).
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.

Sintaxis

Para la instrucción "Desactivar área por bytes" se utiliza la siguiente sintaxis:

SCL

```
RESETI(S_BYTE : = <Operando>,
       N := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
S_BYTE	Input	POINTER	Señala al primer byte del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	Tamaño del área cuyos bits se desactivan. El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
RESETI(S_BYTE := byte_array[100],
       N := 16);
```

En el ejemplo, la instrucción desactiva 2 bytes (16 bits) en el área de 100.0 a 101.7 del área por bytes.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

REPL_VAL: Introducir valor sustitutivo

Descripción

La instrucción "Introducir valor sustitutivo" permite introducir un valor sustitutivo en el acumulador 1 (ACU 1) desde un bloque de organización de error síncrono.

Nota

La instrucción "Introducir valor sustitutivo" solo se puede llamar desde bloques de organización de error síncrono (OB 121, OB 122).

La instrucción "Introducir valor sustitutivo" se utiliza si ya no es posible leer ningún valor de un módulo. Tras cada acceso a dicho módulo se inicia el OB 122. Llamando la instrucción

"Introducir valor sustitutivo" es posible cargar un valor sustitutivo en el acumulador 1. La ejecución del programa se prosigue entonces con este valor sustitutivo. La información necesaria para seleccionar el valor sustitutivo puede consultarse en las variables locales del OB 122.

Sintaxis

Para la instrucción "Introducir valor sustitutivo" se utiliza la siguiente sintaxis:

```
SCL
REPL_VAL(VAL := <Operando>)
```

La sintaxis de la instrucción se compone de las siguientes partes:

Parámetro	Declaración	Tipo de datos	Descripción
VAL	Input	DWORD	Valor sustitutivo
Valor de función (RET_VAL)		INT	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#....)	Descripción
0000	Ningún error
8080	La instrucción no se ha llamado desde un OB de error síncrono.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" ocupa los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de este es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetro DCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

```
SCL
<Instancia>(RESET := <Operando>,
          JOG := <Operando>,
          DRUM_EN := <Operando>,
          LST_STEP := <Operando>,
          EVENT1 - 16 := <Operando>,
          OUT1 - 16 => <Operando>,
          Q => <Operando>,
          OUT_WORD => <Operando>,
          ERR_CODE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Descripción
RESET	Input	BOOL	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	Número del último paso programado.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	Bit de evento (i); El estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	Bit de salida (j)
Q	Output	BOOL	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	Información de error

Parámetro	Declaración	Tipo de datos	Descripción
JOG_HIS	Static	BOOL	Bit de evolución del parámetro JOG
EOD	Static	BOOL	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
DSP	Static	BYTE	Paso predeterminado del circuito secuencial sucesivo
DSC	Static	BYTE	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	Paso actual del circuito secuencial sucesivo
DTBP	Static	WORD	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	Hora de sistema anterior
S_PRESET	Static	ARRAY of WORD	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DRUM_X: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite materializar un circuito secuencial sucesivo con 16 pasos como máximo. El número del primer paso se define con el parámetro DSP. El número del último paso se predetermina con el parámetro LST_STEP.

En cada paso se escriben todos los 16 bits de salida OUT0 a OUT15 y el parámetro de salida OUT_WORD (que contiene los bits de salida). A un bit de salida se asigna bien sea el bit correspondiente del campo predefinido OUT_VAL, o bien el bit de salida del paso anterior. El valor que se asigne dependerá de la ocupación de los bits de máscara en el parámetro S_MASK.

Nota

El ajuste predeterminado de los bits de máscara es 0. Para modificar el ajuste actual de uno o varios bits de máscara, realice el cambio en el bloque de datos instancia.

La instrucción "Materializar circuito secuencial sucesivo" pasa al siguiente paso cuando se produce un flanco de señal ascendente en el parámetro JOG. Si la instrucción ya se encuentra en el último paso y se produce un flanco ascendente en el parámetro JOG, se activan las variables Q y EOD. En este caso, el parámetro DCC se pone al valor 0. La instrucción permanece en el último paso hasta que el parámetro RESET se pone al valor "1".

Además, también se puede permitir la transición al paso siguiente en función del tiempo. Para ello se debe poner al valor "1" el parámetro DRUM_EN. La transición al paso siguiente tiene lugar si está activado el bit de evento (EVENTi) correspondiente al paso actual y si ha transcurrido el tiempo predeterminado para el paso actual. El tiempo necesario para un paso resulta de la multiplicación de la base de tiempo (DTBP por el factor de tiempo válido para el paso actual (S_PRESET).

Nota

El tiempo de ejecución restante en el paso actual (DCC) se reduce únicamente si está activado el bit de evento correspondiente (EVENTi).

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si, al llamar la instrucción, el parámetro RESET tiene el valor "1", el circuito secuencial sucesivo pasará al paso cuyo número se ha asignado al parámetro DSP.

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados solo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1".

El circuito secuencial sucesivo también se puede conmutar en progresión en todo momento a través del valor del parámetro JOG.

En la primera llamada de la instrucción, el parámetro RESET se debe poner a 1.

Si el circuito secuencial se encuentra en el último paso y ha transcurrido el tiempo de ejecución predefinido para este paso, se activan los parámetros Q y EOD. La instrucción permanece en el último paso hasta que el parámetro RESET adopte el valor "1".

Esta instrucción se ejecuta solo en los estados operativos ARRANQUE y RUN.

El sistema operativo inicializa la instrucción "Materializar circuito secuencial sucesivo" cuando se efectúa un arranque en frío. La instrucción no se inicializa en caso de un rearranque completo (en caliente). Si la instrucción "Materializar circuito secuencial sucesivo" debe inicializarse tras un rearranque completo (en caliente), es preciso llamarla en el OB 100 con el valor "1" en el parámetro RESET.

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(RESET := <Operando>,
          JOG := <Operando>,
          DRUM_EN := <Operando>,
          LST_STEP := <Operando>,
          EVENT1 - 16 := <Operando>,
          OUT0 - 15 => <Operando>,
          Q => <Operando>,
          OUT_WORD => <Operando>,
          ERR_CODE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Descripción
RESET	Input	BOOL	El estado lógico "1" desactiva el circuito secuencial. En la primera llamada de la instrucción, el parámetro RESET se debe poner a "1".
JOG	Input	BOOL	Un flanco de señal ascendente (con respecto a la última llamada) provoca la transición al paso siguiente, siempre que el circuito secuencial no se encuentre aún en el último paso. La transición tiene lugar independientemente del valor asignado al parámetro DRUM_EN.
DRUM_EN	Input	BOOL	Parámetro de control que determina si se puede realizar una transición al paso siguiente en función del tiempo. Si el valor es "1", es posible una transición en función del tiempo.
LST_STEP	Input	BYTE	Número del último paso Los valores admisibles son 1 a 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	Bit de evento n.º i (pertenece al paso i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	Bit de salida n.º j (idéntico al bit n.º j de OUT_WORD)
Q	Output	BOOL	Parámetro de estado que indica si ha transcurrido el tiempo predefinido por el usuario para la ejecución del último paso.
OUT_WORD	Output	WORD	Bits de salida resumidos en una variable
ERR_CODE	Output	WORD	Información de error
JOG_HIS	Static	BOOL	Irrelevante para el usuario (parámetro de entrada JOG de la llamada anterior)
EOD	Static	BOOL	Idéntico al parámetro de salida Q
DSP	Static	BYTE	Número del primer paso Los valores admisibles son 1 a 16.
DSC	Static	BYTE	Número del paso actual
DCC	Static	DWORD	Tiempo de ejecución restante del paso actual, indicado en milisegundos (ms) (relevante solo si DRUM_EN = 1 y el bit de evento correspondiente = 1)
DTBP	Static	WORD	Base de tiempo en milisegundos (ms) válida para todos los pasos
PREV_TIME	Static	DWORD	(Irrelevante para el usuario: hora de sistema de la llamada anterior)

Parámetro	Declaración	Tipo de datos	Descripción
S_PRESET	Static	ARRAY of WORD	Campo unidimensional con los factores de tiempo para cada paso. Selección recomendada de índices: [1 a 16]. En este caso, S_PRESET [x] contiene el factor de tiempo del paso x.
OUT_VAL	Static	ARRAY of BOOL	Campo bidimensional con los valores que deben emitirse en cada paso, a no ser que se hayan inhibido vía S_MASK. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, OUT_VAL [x, y] contiene el valor asignado al bit de salida OUTy en el paso x.
S_MASK	Static	ARRAY of BOOL	Campo bidimensional con los bits de máscara para cada paso. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, S_MASK [x, y] contiene el bit de máscara para el valor "y"-ésimo que debe emitirse en el paso x. Significado de los bits de máscara: • 0: el valor correspondiente del paso anterior se asigna al respectivo bit de salida • 1: el valor correspondiente de OUT_VAL se asigna al respectivo bit de salida.

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE* (W#16#...)	Explicación
0000	Ningún error
8081	Valor no admisible en el parámetro LST_STEP
8082	Valor no admisible en DSC
8083	Valor no admisible en DSP
8084	El producto DCC = DTBP * S_PRESET[DSC] excede el valor 2**31-1 (aprox. 24,86 días).
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetro CMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET, OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1"
  - Los parámetros ET, OA y CA se ponen a "0"
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET, OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.

- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control discreto" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(CMD := <Operando>,
          O_FB := <Operando>,
          C_FB := <Operando>,
          Q => <Operando>,
          OA => <Operando>,
          CA => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Descripción
CMD	Input	BOOL	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	Entrada de realimentación al abrir
C_FB	Input	BOOL	Entrada de realimentación al cerrar
Q	Output	BOOL	Indica el estado del parámetro CMD
OA	Output	BOOL	Salida de alarma al abrir
CA	Output	BOOL	Salida de alarma al cerrar
ET	Static	DINT	Tiempo actual transcurrido; un ciclo de reloj = 1 ms.
PT	Static	DINT	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms.
PREV_TIME	Static	DWORD	Hora de sistema anterior
CMD_HIS	Static	BOOL	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

SCL

```
"DCAT_DB"(CMD := "Tag_Input_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          Q => "Tag_Output_Q",
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se activa la alarma correspondiente.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir

Parámetros de entrada								Parámetros de salida								
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT
>= PT	ET >= PT

Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida:

OO = FALSE

CO = FALSE

OA = FALSE

CA = FALSE

ET = PT

Q = TRUE

Sintaxis

Para la instrucción "Alarma de tiempo para control del motor" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(O_CMD := <Operando>,
          C_CMD := <Operando>,
          S_CMD := <Operando>,
          O_FB := <Operando>,
          C_FB := <Operando>,
          OO => <Operando>,
          CO => <Operando>,
          OA => <Operando>,
          CA => <Operando>,
          Q => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Descripción
O_CMD	Input	BOOL	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	Entrada de la orden "Parar"
O_FB	Input	BOOL	Entrada de realimentación al abrir
C_FB	Input	BOOL	Entrada de realimentación al cerrar
OO	Output	BOOL	Salida "Abrir"
CO	Output	BOOL	Salida "Cerrar"
OA	Output	BOOL	Salida de alarma al abrir
CA	Output	BOOL	Salida de alarma al cerrar
Q	Output	BOOL	El estado lógico "0" indica una condición de error.
ET	Static	DINT	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	Hora de sistema anterior
O_HIS	Static	BOOL	Bit de evolución "Abrir"
C_HIS	Static	BOOL	Bit de evolución "Cerrar"

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

SCL

```
"MCAT_DB"(O_CMD := "Tag_Iput_O_CMD",
          C_CMD := "Tag_Input_C_CMD",
          S_CMD := "Tag_Input_S_CMD",
          O_FB := "Tag_Input_O_FB",
          C_FB := "Tag_Input_C_FB",
          OO => "Tag_OutputOpen",
          CO => "Tag_OutputClosed",
```

SCL

```
          OA => "Tag_Output_OA",
          CA => "Tag_Output_CA",
          Q => "Tag_Output_Q");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una correspondencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Comparar bits de entrada con bits de una máscara" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(IN_BIT0 - 15 := <Operando>,
          CMP_STEP := <Operando>,
          OUT => <Operando>,
          ERR_CODE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Descripción
IN_BIT0	Input	BOOL	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	Número de paso de la máscara con la que se compara.
OUT	Output	BOOL	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	Información de error
CMP_VAL	Static	ARRAY OF BOOL	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Barrido de matriz" se utiliza la siguiente sintaxis:

```
SCL
<Instancia>(IN_BIT0 - 15 := <Operando>,
          OUT => <Operando>,
          OUT_STEP => <Operando>,
          ERR_CODE => <Operando>)
```

Sintaxis

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Descripción
IN_BIT0	Input	BOOL	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	El bit de entrada 15 se compara con el bit 15 de la máscara.

Parámetro	Declaración	Tipo de datos	Descripción
OUT	Output	BOOL	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
OUT_STEP	Output	BYTE	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	Información de error
LAST	Static	BYTE	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY OF WORD	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número del paso y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor de la ganancia en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" solo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse como supresor de perturbaciones o filtro de paso bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa, el cuadro de diálogo "Opciones de llamada" se abre automáticamente; en él se puede determinar si los parámetros de bloque de la instrucción se depositan en un bloque de datos propio (instancia individual) o bien como variable local (multiinstancia) en la interfaz del bloque. Si crea un bloque de datos propio, lo encontrará en el árbol del proyecto, en la carpeta "Recursos de programa", en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

SCL

```
<Instancia>(IN := <Operando>,
          SAMPLE_T := <Operando>,
          OUT => <Operando>,
          ERR_CODE => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	REAL	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa. En el parámetro IN también se pueden indicar constantes.
SAMPLE_T	Input	INT	Tiempo de muestreo En el parámetro SAMPLE_T también se pueden indicar constantes.
OUT	Output	REAL	Resultado de la instrucción
ERR_CODE	Output	WORD	Información de error
LD_TIME	Static	REAL	Tiempo de anticipación en la misma unidad que el tiempo de muestreo.
LG_TIME	Static	REAL	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).
PREV_IN	Static	REAL	Entrada anterior
PREV_OUT	Static	REAL	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Los parámetros estáticos no son visibles al efectuar la llamada de la instrucción en el programa. Estos se almacenan en la instancia de la instrucción.

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

```
SCL
"LEAD_LAG_DB"(IN := "Tag_Input",
              SAMPLE_T := "Tag_Input_SAMPLE_T",
              OUT => "Tag_Output_Result",
              ERR_CODE => "Tag_ErrorCode");
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

En el bloque de datos instancia "LEAD_LAG_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

TONR_X: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" acumula valores de tiempo dentro de un periodo definido por el parámetro PV. Si el estado lógico del parámetro TMR_EN cambia a "1", se ejecuta la instrucción y se empieza a contar el tiempo PV. La instrucción "Acumulador de tiempo" suma el valor del parámetro DELTA_T al del parámetro ET siempre que se cumplan las siguientes condiciones:

- El parámetro TMR_EN devuelve el estado lógico "1"
- El valor de tiempo del parámetro ET es menor que el del parámetro PV.
- El parámetro RESET devuelve el estado lógico "0".

Si el valor del parámetro TMR_EN cambia a "0", el tiempo del parámetro DELTA_T se dejará de sumar al valor de tiempo del parámetro ET.

Cuando el valor de tiempo del parámetro ET alcanza el del parámetro PV, el parámetro Q devuelve el estado lógico "1". Mientras el parámetro Q devuelva el estado lógico "1", el valor del parámetro ET no se modificará. El valor del parámetro ET se pone a "0" cuando el parámetro RESET se pone a "1".

Puesto que la instrucción "Acumulador de tiempo" utiliza el tiempo de ejecución (DELTA_T) del último ciclo del bloque de organización (OB), la instrucción solo puede llamarse en bloques de organización cíclicos.

Nota

El tiempo de ciclo del bloque de organización debe desplazarse desde la información de arranque a la variable global del parámetro DELTA_T.

Sintaxis

Para la instrucción "Acumulador de tiempo" se utiliza la siguiente sintaxis:

SCL

```
TONR_X(TMR_EN := <Operando>,
       RESET := <Operando>,
       PV := <Operando>,
       DELTA_T := <Operando>,
       ET := <Operando>,
       Q => <Operando>)
```

Sintaxis

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos	Descripción
TMR_EN	Input	BOOL	Activa el temporizador para acumular el valor de tiempo.
RESET	Input	BOOL	Entrada de reset
PV	Input	DINT	Tiempo máximo de lectura del tiempo
DELTA_T	Input	INT	Tiempo de ciclo del OB del ciclo anterior
ET	InOut	DINT	Valor de tiempo actual del tiempo transcurrido
Q	Output	BOOL	Salida que se activa una vez transcurrido el tiempo PV.
Valor de función		VOID	Valor de función vacío

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

WSR: Guardar datos en el registro de desplazamiento

Descripción

La instrucción "Guardar datos en el registro de desplazamiento" escribe datos en un registro de desplazamiento desde el origen indicado. Todo elemento del registro se desplaza a la siguiente dirección. Los datos contenidos en la última dirección del registro de desplazamiento se pierden tras la operación de desplazamiento.

Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen (S_DATA) y se desplazan a la dirección inicial (START) del registro de desplazamiento. El parámetro LEN indica el número de elementos que deben desplazarse. El tipo de datos de los elementos se define en el parámetro E_TYPE. Si se indica un tipo de datos no admisible en el parámetro E_TYPE, no se procesa la instrucción.

Si el parámetro RESET está a "1" al ejecutarse la instrucción, se borra el registro. El parámetro Q se pone al estado lógico "1" si el registro de desplazamiento está ocupado con ceros o si se borra.

Sintaxis

Para la instrucción "Guardar datos en el registro de desplazamiento" se utiliza la siguiente sintaxis:

SCL

```
WSR(RESET := <Operando>,
    S_DATA := <Operando>,
    START := <Operando>,
    LEN := <Operando>,
    E_TYPE := <Operando>
    Q => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Guardar datos en el registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Descripción
RESET	Input	BOOL	Entrada de reset
S_DATA	Input	POINTER	Apunta al elemento de origen que se desplaza a la dirección inicial (START). En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
START	Input	POINTER	Apunta a la dirección inicial del registro. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
LEN	Input	WORD	Número de elementos que deben desplazarse.
E_TYPE	Input	BYTE	Indica el tipo de datos de los elementos que deben desplazarse. Los siguientes tipos de datos son admisibles: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL

Parámetro	Declaración	Tipo de datos	Descripción
Q	Output	BOOL	Indica "0" si el parámetro RESET está activado (1) o si todos los elementos que deben desplazarse tienen el valor "0".
Valor de función		VOID	Valor de función vacío

Ejemplo

En el ejemplo siguiente se desplazan cuatro elementos del tipo de datos WORD a partir de la dirección indicada en el parámetro START. Una vez desplazado el primer elemento del registro a la siguiente dirección, la primera dirección se ocupará con los datos a los que señala el puntero en el parámetro S_DATA. El valor del último elemento del registro se pierde entonces.

SCL

```
WSR(RESET := "Tag_Input_Reset",
    S_DATA := "Tag_Input_DataSource",
    START := "Tag_Input_StartAddress",
    LEN := "Tag_Input_Length",
    E_TYPE := "Tag_Input_ElementType",
    Q => "Tag_Output_Q");
```

No se transfiere ningún valor de función.

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	W#16#04
Q	Tag_Output_Q	1

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

SHRB: Desplazar bit al registro de desplazamiento

Descripción

La instrucción "Desplazar bit al registro de desplazamiento" permite escribir un bit en un registro de desplazamiento desde el origen indicado (DATA). Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen y se desplazan a la dirección inicial (S_BIT) del registro de desplazamiento. Los restantes bits se desplazan una posición. El bit de la última dirección (S_BIT + N) se pierde tras el desplazamiento. Si el parámetro RESET está a "1" mientras se ejecuta la instrucción, las direcciones del registro de desplazamiento se ponen a "0" y no se desplazan.

Sintaxis

Para la instrucción "Desplazar bit al registro de desplazamiento" se utiliza la siguiente sintaxis:

SCL

```
SHRB(DATA := <Operando>,
     RESET := <Operando>,
     S_BIT := <Operando>,
     N := <Operando>)
```

La tabla siguiente muestra los parámetros de la instrucción:

Parámetro	Declaración	Tipo de datos	Descripción
DATA	Input	BOOL	Bit de origen
RESET	Input	BOOL	Entrada de reset
S_BIT	Input	POINTER	Apunta al primer bit del registro de desplazamiento. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	WORD	Número de bits que deben desplazarse.
Valor de función		VOID	Valor de función vacío

Ejemplo

En el ejemplo siguiente, el parámetro N se pone al valor 14 ("E" en notación hexadecimal), con lo que 14 bits se desplazan a partir de la dirección indicada en el parámetro S_BIT. Una vez desplazados los bits, la primera dirección se ocupa con los datos del origen del parámetro DATA. El último valor de bit se pierde.

SCL

```
SHRB(DATA := "Tag_Input_Data",
     RESET := "Tag_Input_Reset",
     S_BIT := "Tag_Input_BitSource",
     N := "Tag_Input_Number");
```

No se transfiere ningún valor de función.

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Antes de ejecutarse la instrucción, los siguientes estados lógicos están almacenados en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Después de la ejecución

Una vez ejecutada la instrucción, los siguientes estados lógicos se almacenan en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

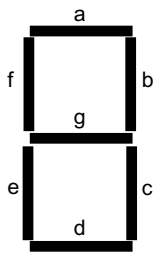
Edición de instrucciones SCL (Página 2823)

SEG: Generar patrón de bits para el display de 7 segmentos

Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" convierte cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se deposita en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)	Display de siete segmentos
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Sintaxis

Para la instrucción "Generar patrón de bits para el display de 7 segmentos" se utiliza la siguiente sintaxis:

SCL

```
SEG(IN := <Operando>,
    OUT => <Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Descripción
IN	Input	WORD	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	Patrón de bits para el display de 7 segmentos
Valor de función		VOID	Valor de función vacío

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
SEG(IN := "Tag_Input",
    OUT => "Tag_Output");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW16#065B4F66	00000110 01011011 01001111 01100110 Visualización: 1234

No se transfiere ningún valor de función.

Consulte también

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Vista general de los tipos de datos válidos (Página 2321)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el operando. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

Sintaxis

Para la instrucción "Generar complemento a diez" se utiliza la siguiente sintaxis:

```
SCL
BCDCPL(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	DWORD	Número BCD de 7 dígitos
Valor de función		DWORD	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := BCDCPL("Tag_Input");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Operando	Valor*
Tag_Input	DW#16#01234567
Tag_Result	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" cuenta el número de bits de un operando que están puestos al estado lógico "1".

Sintaxis

Para la instrucción "Contar número de bits activados" se utiliza la siguiente sintaxis:

```
SCL
BITSUM(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Descripción
<Operando>	Input	DWORD	Operando cuyos bits activados se cuentan
Valor de función		INT	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

```
SCL
"Tag_Result" := BITSUM("Tag_Input");
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Operando	Valor*
Tag_Input	DW#16#12345678
Tag_Result	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Entrada de instrucciones SCL (Página 2804)

Edición de instrucciones SCL (Página 2823)

Cambiar formatos de visualización en el estado del programa (Página 3100)

GRAPH

Estructura GRAPH

Etapa y transición

Descripción

El componente de estructura "Etapa y transición" permite insertar simultáneamente una etapa y una transición en la cadena secuencial.

Consulte también:

Etapa (Página 6536)

Transición (Página 6537)

Consulte también

Etapas y transiciones (Página 2831)

Etapa (Página 6536)

Transición (Página 6537)

Fin de cadena (Página 6538)

Salto a etapa (Página 6538)

Rama alternativa (Página 6539)

Rama simultánea (Página 6539)

Cerrar rama (Página 6540)

Etapa

Descripción

Las etapas sirven para dividir las tareas de automatización complejas en tareas más pequeñas y claras que, seguidamente, se llevan a cabo por medio de acciones. Las diferentes etapas se organizan en cadenas secuenciales, de modo que al ejecutar el programa cada etapa se procesa en el orden especificado. Cada etapa debe tener un nombre y un número unívocos.

Para que el procesamiento se lleve a cabo realmente, la etapa debe activarse con una de las condiciones siguientes:

- La etapa se ha definido como inicial.
- Se cumple la transición de la etapa anterior.
- La etapa es llamada por una acción controlada por evento.

Una vez se han procesado todas las acciones se vuelve a desactivar la etapa.

Las etapas en las que no hay ninguna acción programada se denominan etapas vacías. Una etapa vacía tiene el mismo comportamiento que una etapa activa y la transición siguiente siempre se cumple.

Consulte también

Principios básicos de las etapas (Página 2831)

Etapa y transición (Página 6536)

Transición (Página 6537)

Fin de cadena (Página 6538)

Salto a etapa (Página 6538)

Rama alternativa (Página 6539)

Rama simultánea (Página 6539)

Cerrar rama (Página 6540)

Transición

Descripción

Las transiciones se encuentran entre una etapa y otra y contienen las condiciones para pasar de una etapa a la siguiente. Cuando se cumplen las condiciones de continuación de una transición, se activa la etapa siguiente y se procesan sus acciones. Las condiciones de una transición se programan en KOP o FUP.

Consulte también

Principios básicos de las transiciones (Página 2833)

Etapa y transición (Página 6536)

Etapa (Página 6536)

Fin de cadena (Página 6538)

Salto a etapa (Página 6538)

Rama alternativa (Página 6539)

Rama simultánea (Página 6539)

Cerrar rama (Página 6540)

Fin de cadena

Descripción

El componente de estructura "Fin de cadena" permite finalizar una cadena secuencial o una rama. En una rama simultánea, sin embargo, el fin de cadena debe estar precedido de una transición.

Nota

Si todas las ramas de una cadena secuencial se cierran con un fin de cadena, es posible volver a iniciar la cadena secuencial con el parámetro "INIT_SQ" o con el botón "Inicializar" de la Task Card "Test" en la paleta "Control secuencial".

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 6536)

Etapa (Página 6536)

Transición (Página 6537)

Salto a etapa (Página 6538)

Rama alternativa (Página 6539)

Rama simultánea (Página 6539)

Cerrar rama (Página 6540)

Salto a etapa

Descripción

Con un salto se prosigue la ejecución del programa en una etapa cualquiera dentro del bloque de función GRAPH. Los saltos pueden insertarse al final de la rama principal o de una rama alternativa, de modo que sea posible procesar la cadena secuencial cíclicamente. El salto y su destino se representan en la cadena secuencial en forma de flecha, de modo que en el destino del salto se indica la transición de origen.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 6536)

Etapa (Página 6536)

Transición (Página 6537)

Fin de cadena (Página 6538)

Rama alternativa (Página 6539)

Rama simultánea (Página 6539)

Cerrar rama (Página 6540)

Rama alternativa

Descripción

Las ramas alternativas permiten programar ramas O. Esto significa que, después de una etapa, se insertan ramas que empiezan con una transición. En función de la transición que se cumpla primero se ejecutará la rama correspondiente. Si se cumplen varias transiciones al mismo tiempo, tendrá prioridad la transición que esté más a la izquierda y se ejecutará la rama correspondiente. Las ramas alternativas finalizan con otra transición.

Es posible programar un máximo de 125 ramas alternativas en una cadena secuencial.

Consulte también

Elementos de una cadena secuencial (Página 2828)

Etapa y transición (Página 6536)

Etapa (Página 6536)

Transición (Página 6537)

Fin de cadena (Página 6538)

Salto a etapa (Página 6538)

Rama simultánea (Página 6539)

Cerrar rama (Página 6540)

Rama simultánea

Descripción

Las ramas simultáneas permiten programar ramas Y. Esto significa que con una transición se activan varias etapas cuyas acciones se ejecutarán a continuación. Por ese motivo, las ramas simultáneas empiezan y terminan siempre con una etapa.

Las transiciones posteriores de las ramas simultáneas se encuentran en la rama principal, de modo que las diferentes ramas simultáneas pueden cerrarse en diferentes puntos de la rama principal. Tenga en cuenta que las ramas que se agrupan en una transición no conmutarán a la etapa siguiente, hasta que no se hayan ejecutado por completo dichas ramas.

Es posible programar un máximo de 249 ramas simultáneas en una cadena secuencial.

Consulte también

Elementos de una cadena secuencial (Página 2828)
Etapa y transición (Página 6536)
Etapa (Página 6536)
Transición (Página 6537)
Fin de cadena (Página 6538)
Salto a etapa (Página 6538)
Rama alternativa (Página 6539)
Cerrar rama (Página 6540)

Cerrar rama

Descripción

El elemento "Cerrar rama" permite cerrar ramas simultáneas o alternativas y llevarlas a su rama matriz. Esto debe hacerse cuando la rama no se cierra con un salto o un fin de cadena. En una rama simultánea solo es posible insertar "Cerrar rama" después de una etapa.

Consulte también

Elementos de una cadena secuencial (Página 2828)
Etapa y transición (Página 6536)
Etapa (Página 6536)
Transición (Página 6537)
Fin de cadena (Página 6538)
Salto a etapa (Página 6538)
Rama alternativa (Página 6539)
Rama simultánea (Página 6539)

Acciones GRAPH

Temporizadores

TP: Impulso

Descripción

La instrucción "Impulso" permite activar la salida Q por un tiempo programado. La instrucción se inicia cuando el resultado lógico (RLO) del parámetro IN cambia de "0" a "1" (flanco de

señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. El parámetro Q se activa por el tiempo PT, independientemente de cómo evolucione la señal de entrada. La detección de un nuevo flanco de señal ascendente tampoco influye en el estado lógico del parámetro Q mientras transcurra el tiempo PT.

El valor actual de tiempo se puede consultar en el parámetro ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Si cuando se alcanza el tiempo PT el estado lógico del parámetro IN es "0", se resetea el parámetro ET.

Nota

Si en el programa no se llama al temporizador porque, p. ej., este es ignorado, el parámetro ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Impulso" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TP (p. ej., "TP_DB")

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La actualización de los datos de la instrucción se lleva a cabo tanto cuando se llama la instrucción como cuando se accede a los parámetros Q y ET.

El sistema operativo desactiva las instancias de la instrucción "Impulso" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Impulso", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Impulso" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Impulso" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TP <Tipo de datos>, <Temporizador CEI>
     (IN := <Operando>
      PT := <Operando>
      Q => <Operando>
      ET => <Operando>
      )
```

Parámetros

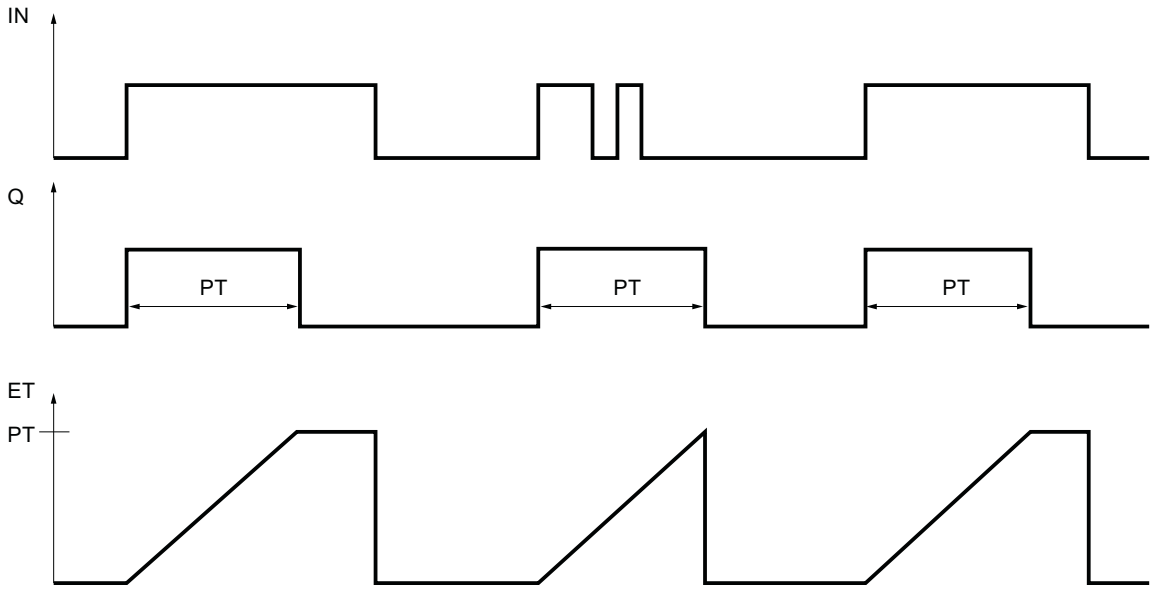
La tabla siguiente muestra los parámetros de la instrucción "Impulso":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Duración del impulso El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida de impulso
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Impulso":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TP TIME, "IEC_TP_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
```

GRAPH

```
Q => "Tag_Status"
ET => "Tag_ElapsedTIME"
)
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar y el operando "Tag_Status" se pone a "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTIME".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

TON: Retardo al conectar

Descripción

Con la instrucción "Retardo al conectar" se puede retardar la activación de la salida Q por el tiempo programado PT. La instrucción se inicia cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). El tiempo programado PT empieza a contarse al iniciar la instrucción. Una vez transcurrido el tiempo PT, la salida Q devuelve el estado lógico "1". La salida Q permanece activada mientras la entrada de arranque IN esté puesta a "1". Cuando el estado lógico de la entrada de arranque cambia de "1" a "0", se desactiva la salida Q. La función de temporización se reinicia al detectarse un flanco de señal ascendente nuevo en la entrada de arranque.

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. La salida ET se desactiva en cuanto el estado lógico de la entrada IN cambia a "0".

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Retardo al conectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TON (p. ej., "TON_DB")

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al conectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al conectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Retardo al conectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Retardo al conectar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TON <Tipo de datos>, <Temporizador CEI>
      (IN := <Operando>
       PT := <Operando>
       Q => <Operando>
       ET => <Operando>
       )
```

Parámetros

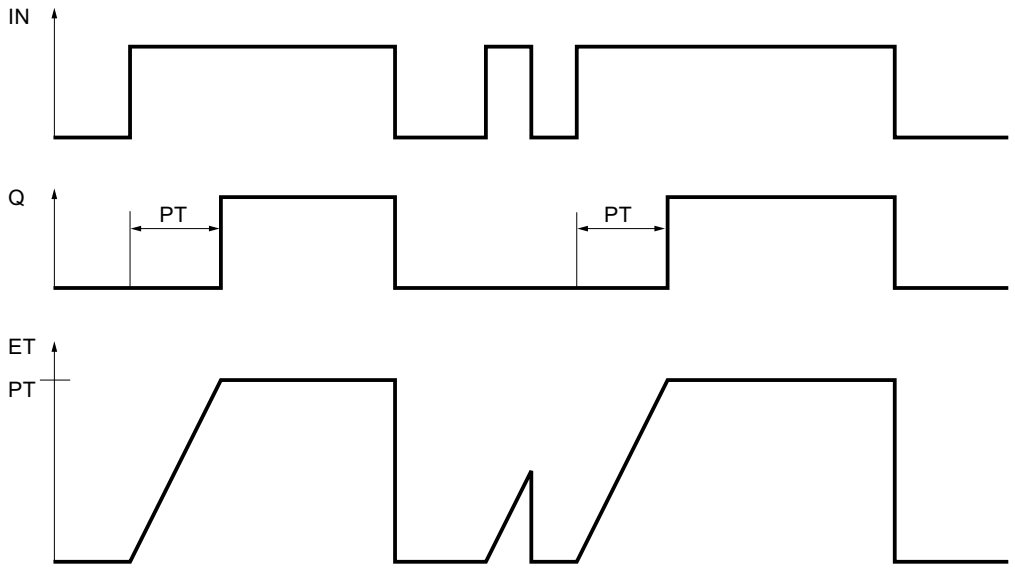
La tabla siguiente muestra los parámetros de la instrucción "Retardo al conectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al conectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida que se activa una vez transcurrido el tiempo PT.
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al conectar":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TON TIME, "IEC_TON_DB"
      (IN := "Tag_Start"
       PT := "Tag_PresetTIME"
       Q => "Tag_Status"
       ET => "Tag_ElapsedTIME"
       )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", el tiempo programado en el parámetro PT empieza a contar. Una vez transcurrido el tiempo, el operando "Tag_Status" se pone al estado lógico "1". El operando "Tag_Status" permanece a "1" mientras el operando "Tag_Start" tenga el estado lógico "1". El valor actual de tiempo se guarda en el operando "Tag_ElapsedTime". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", se resetea el operando "Tag_Status".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

TOF: Retardo al desconectar

Descripción

Con la instrucción "Retardo al desconectar" se puede retardar la desactivación de la salida Q por el tiempo programado PT. La salida Q se activa cuando el resultado lógico (RLO) de la entrada IN cambia de "0" a "1" (flanco de señal ascendente). Cuando el estado lógico de la entrada IN cambia nuevamente a "0", el tiempo programado PT deja de contar. La salida Q permanece activada mientras transcurre el tiempo PT. Una vez transcurrido el tiempo PT se desactiva la salida Q. Si el estado lógico de la entrada IN cambia a "1" antes de que transcurra el tiempo PT, se inicializa el temporizador. El estado lógico de la salida Q permanece en "1".

El valor de tiempo actual se puede consultar en la salida ET. Este valor de tiempo empieza a partir de T#0s y termina al alcanzarse el valor del tiempo PT. Una vez transcurrido el tiempo PT, el valor actual de la salida ET se conserva hasta que la entrada IN cambie nuevamente a "1". Si el estado lógico de la entrada IN cambia a "1" antes de transcurrir el tiempo PT, la salida ET adopta el valor T#0s.

Nota

Si en el programa no se llama el temporizador porque, p. ej., este es ignorado, la salida ET devuelve un valor de constante en cuanto ha transcurrido este tiempo.

A cada llamada de la instrucción "Retardo al desconectar" debe asignársele un temporizador CEI, en el que se guardan los datos de la instrucción. Un temporizador CEI se puede declarar como se indica a continuación:

- Declaración de un bloque de datos del tipo TOF (p. ej., "TOF_DB")

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Los datos de la instrucción se actualizan tanto al efectuar una llamada de la instrucción como cada vez que se accede a las salidas Q o ET.

El sistema operativo desactiva las instancias de la instrucción "Retardo al desconectar" cuando se realiza un arranque en frío. Si alguna instancia de la instrucción debe estar inicializada tras un rearranque completo (en caliente), las instancias que se deben inicializar se deben llamar en un OB de arranque con el valor "0" en el parámetro PT. Si un bloque diferente contiene instancias de la instrucción "Retardo al desconectar", estas pueden inicializarse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Retardo al desconectar" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Retardo al desconectar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TOF <Tipo de datos>, <Temporizador CEI>
```

GRAPH

```
(IN := <Operando>
 PT := <Operando>
 Q => <Operando>
 ET => <Operando>
 )
```

Parámetros

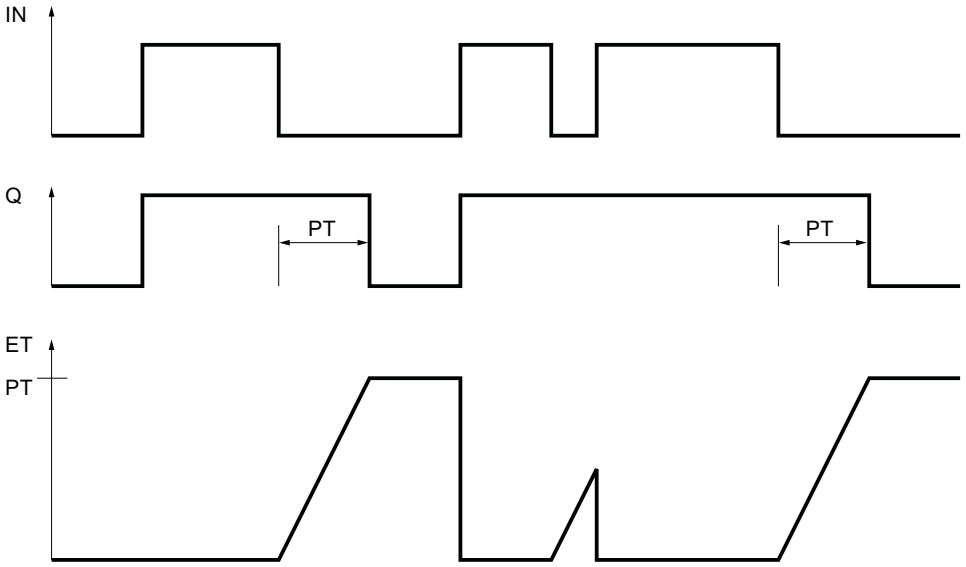
La tabla siguiente muestra los parámetros de la instrucción "Retardo al desconectar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	BOOL	I, Q, M, D, L	Entrada de arranque
PT	Input	TIME	I, Q, M, D, L, P o constante	Tiempo del retardo al desconectar El valor del parámetro PT debe ser positivo.
Q	Output	BOOL	I, Q, M, D, L	Salida que se desactiva una vez transcurrido el tiempo PT.
ET	Output	TIME	I, Q, M, D, L	Valor de tiempo actual

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Cronograma de impulsos

La figura siguiente muestra el cronograma de impulsos de la instrucción "Retardo al desconectar":



Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL TOF TIME, "IEC_TOF_DB"
     (IN := "Tag_Start"
      PT := "Tag_PresetTIME"
      Q => "Tag_Status"
      ET => "Tag_ElapsedTIME"
      )
```

Si se produce un cambio del estado lógico del operando "Tag_Start" de "0" a "1", el operando se pone a "Tag_Status". Cuando el estado lógico del operando "Tag_Start" cambia de "1" a "0", el tiempo programado en el parámetro PT empieza a contar. Mientras transcurra el tiempo, permanecerá activado el operando "Tag_Status". Una vez transcurrido el tiempo, el operando "Tag_Status" se desactiva. El valor actual de tiempo se guarda en el operando "Tag_ElapsedTIME".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Contadores

CTU: Contador ascendente

Descripción

La instrucción "Contador ascendente" incrementa el valor de la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se incrementa en uno. La primera vez que se ejecuta la instrucción, el valor actual de contaje de la salida CV está a cero. El valor de contaje se incrementa cada vez que se detecta un flanco de señal ascendente, hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Cuando se alcanza el valor límite superior, el estado lógico de la entrada CU deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. El estado lógico de la salida Q es determinado por el parámetro PV. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0".

El valor de la salida CV se pone a cero cuando el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", el estado lógico de la entrada CU no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente" debe asignársele un contador CEI en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTU (p. ej., "CTU_DB")

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con el valor "1" en el parámetro R de la instrucción. Si un bloque diferente contiene instancias de la instrucción "Contador ascendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Contador ascendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Contador ascendente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CTU <Tipo de datos>, <Contador CEI>
     (CU := <Operando>
      R := <Operando>
      PV := <Operando>
      Q => <Operando>
      CV => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q.
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor actual de contaje

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL CTU INT,"IEC_CTU_DB"
     (CU := "Tag_Start"
      R := "Tag_ResetCOUNTER"
      PV := "Tag_PresetValue"
      Q => "Tag_Status"
      CV => "Tag_CounterValue"
      )
```

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", se ejecuta la instrucción "Contador ascendente" y el valor actual de contaje del operando "Tag_CounterValue" se incrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá incrementando hasta alcanzar el valor límite superior del tipo de datos (INT = 32767).

El valor del parámetro PV se utiliza como límite para determinar la salida "Tag_Status". La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor del operando "Tag_PresetValue". En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

CTD: Contador descendente

Descripción

La instrucción "Contador descendente" decrementa el valor en la salida CV. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción y el valor actual de contaje de la salida CV se decrementa en uno. Al ejecutar la instrucción por primera vez, el valor de contaje del parámetro CV se pone al valor del parámetro PV. Cada vez que se detecta un flanco de señal ascendente, el valor de contaje se decrementa hasta alcanzar el valor límite inferior del tipo de datos indicado. Cuando se alcanza el valor límite inferior, el estado lógico de la entrada CD deja de tener efecto en la instrucción.

El estado del contador se puede consultar en la salida Q. Si el valor actual de contaje es menor o igual a cero, la salida Q adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida Q es "0".

El valor de la salida CV se pone al valor del parámetro PV cuando el estado lógico de la entrada LD cambia a "1". Mientras la entrada LD tenga el estado lógico "1", el estado lógico de la entrada CD no tendrá efecto alguno en la instrucción.

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador descendente" debe asignársele un contador CEI en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTD (p. ej., "CTD_DB")

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias de la instrucción tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de

arranque con el valor "1" en el parámetro LD de la instrucción. En el parámetro PV se indica en este caso el valor inicial deseado para el parámetro CV.

Para poder ejecutar la instrucción "Contador descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Contador descendente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CTD <Tipo de datos>, <Contador CEI>
     (CD := <Operando>
      LD := <Operando>
      PV := <Operando>
      Q => <Operando>
      CV => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje
LD	Input	BOOL	I, Q, M, D, L, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida Q
Q	Output	BOOL	I, Q, M, D, L	Estado del contador
CV	Output	INT	I, Q, M, D, L, P	Valor actual de contaje

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL CTD INT, "IEC_CTD_DB"
     (CD := "Tag_Start"
      LD := "Tag_LoadPV"
      PV := "Tag_PresetValue"
      Q => "Tag_Status"
      CV => "Tag_CounterValue"
```

GRAPH
)

Cuando el estado lógico del operando "Tag_Start" cambia de "0" a "1", se ejecuta la instrucción "Contador descendente" y el valor de la salida "Tag_CounterValue" se decrementa en uno. Con cada flanco de señal ascendente posterior, el valor de contaje irá decrementando hasta alcanzar el valor límite inferior del tipo de datos indicado (INT = -32768).

La salida "Tag_Status" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_Status" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

CTUD: Contador ascendente - descendente

Descripción

Con la instrucción "Contador ascendente - descendente" se incrementa y decrementa el valor en la salida CV. Cuando el estado lógico de la entrada CU cambia de "0" a "1" (flanco de señal ascendente), el valor actual de contaje de la salida CV aumenta en uno. Cuando el estado lógico de la entrada CD cambia de "0" a "1" (flanco de señal ascendente), el valor actual de contaje de la salida CV se decrementa en uno. Si en un ciclo del programa se detecta un flanco de señal ascendente en las entradas CU y CD, el valor actual de contaje no se modifica en la salida CV.

El valor de contaje se puede seguir incrementando hasta alcanzar el valor límite superior del tipo de datos indicado en la salida CV. Una vez alcanzado el valor límite superior, el valor de contaje ya no se incrementa al detectarse un flanco de señal ascendente. Cuando se alcanza el valor límite inferior del tipo de datos indicado, ya no se decrementa el valor de contaje.

Si el estado lógico de la entrada LD cambia a "1", el valor de contaje de la salida CV adopta el valor del parámetro PV. Mientras la entrada LD tenga el estado lógico "1", el estado lógico de las entradas CU y CD no tendrá efecto alguno en la instrucción.

El valor de contaje se pone a cero si el estado lógico de la entrada R cambia a "1". Mientras la entrada R tenga el estado lógico "1", un cambio del estado lógico de las entradas CU, CD y LD no tendrá efecto alguno en la instrucción "Contador ascendente - descendente".

El estado del contador ascendente se puede consultar en la salida QU. Si el valor actual de contaje es mayor o igual al valor del parámetro PV, la salida QU adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QU es "0".

El estado del contador descendente se puede consultar en la salida QD. Si el valor actual de contaje es menor o igual a cero, la salida QD adopta el estado lógico "1". En todos los demás casos, el estado lógico de la salida QD es "0".

Nota

Utilice el contador en una sola posición del programa para evitar fallos de contaje.

A cada llamada de la instrucción "Contador ascendente - descendente" debe asignársele un contador CEI, en el que se guarden los datos de la instrucción. Un contador CEI se declara del siguiente modo:

- Declaración de un bloque de datos del tipo CTUD (p. ej., "CTUD_DB")

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

El sistema operativo inicializa las instancias de la instrucción "Contador ascendente - descendente" cuando se realiza un arranque en frío. Si deben inicializarse instancias tras un rearranque completo (en caliente), las instancias por inicializar deben llamarse en un OB de arranque con los siguientes valores de parámetros:

- Si se utiliza como contador ascendente, el valor del parámetro R debe ponerse a "1".
- Si se utiliza como contador descendente, el valor del parámetro LD debe ponerse a "1". En el parámetro PV debe indicarse en este caso el valor inicial deseado del parámetro CV.

Si un bloque diferente contiene instancias de la instrucción "Contador ascendente - descendente", estas pueden resetearse p. ej. inicializando el bloque de nivel superior.

Para poder ejecutar la instrucción "Contador ascendente - descendente" se requiere una operación lógica precedente. Se puede colocar dentro o al final del segmento.

Sintaxis

Para la instrucción "Contador ascendente - descendente" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CTUD <Tipo de datos>, <Contador CEI>
      (CU := <Operando>
       CD := <Operando>
       R := <Operando>
       LD := <Operando>
       PV := <Operando>
       QU => <Operando>
       QD => <Operando>
       CV => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contador ascendente - descendente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CU	Input	BOOL	I, Q, M, D, L	Entrada de contaje ascendente
CD	Input	BOOL	I, Q, M, D, L	Entrada de contaje descendente
R	Input	BOOL	I, Q, M, D, L, T, C, P	Entrada de reset
LD	Input	BOOL	I, Q, M, D, L, P	Entrada de carga
PV	Input	INT	I, Q, M, D, L, P o constante	Valor con el que se activa la salida QU
QU	Output	BOOL	I, Q, M, D, L	Estado de los contadores ascendentes
QD	Output	BOOL	I, Q, M, D, L	Estado de los contadores descendentes
CV	Output	INT	I, Q, M, D, L, P	Valor actual de contaje

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL CTUD INT, "IEC_CTUD_DB"
     (CU := "Tag_StartCTU"
      CD := "Tag_StartCTD"
      R := "Tag_ResetCOUNTER"
      LD := "Tag_LoadPV"
      PV := "Tag_PresetValue"
      QU => "Tag_CounterStatusUP"
      QD => "Tag_CounterStatusDOWN"
      CV => "Tag_CounterValue"
      )
```

Si el estado lógico de la entrada "Tag_StartCTU" o "Tag_StartCTD" cambia de "0" a "1" (flanco de señal ascendente), se ejecuta la instrucción "Contador ascendente - descendente". Cuando se detecta un flanco de señal ascendente en la entrada "Tag_StartCTU", el valor de contaje actual se incrementa en uno y se deposita en la salida "Tag_CounterValue". Si se detecta un flanco de señal ascendente en la entrada "Tag_StartCTD", el valor de contaje se decrementa

en uno y se deposita en la salida "Tag_CounterValue". Si se detecta un flanco de señal ascendente en la entrada CU, el valor de contaje se sigue incrementando hasta alcanzar el valor límite superior (32767). Cuando se detecta un flanco de señal ascendente en la entrada CD, el valor de contaje se decrementa hasta alcanzar el valor límite inferior INT = -32768.

La salida "Tag_CounterStatusUP" devuelve el estado lógico "1" mientras el valor actual de contaje sea mayor o igual al valor de la entrada "Tag_PresetValue". En todos los demás casos, la salida "Tag_CounterStatusUP" devuelve el estado lógico "0".

La salida "Tag_CounterStatusDOWN" devuelve el estado lógico "1" mientras el valor de contaje actual sea menor o igual a cero. En todos los demás casos, la salida "Tag_CounterStatusDOWN" devuelve el estado lógico "0".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Funciones matemáticas

NEG: Generar complemento a dos

Descripción

Con la instrucción "Generar complemento a dos" se cambia el signo del valor del operando. Por ejemplo, si tiene un valor positivo, se deposita el equivalente negativo de este valor.

El valor de la salida OUT no es válido si se cumple la siguiente condición:

- El resultado de la instrucción está fuera del rango permitido para el tipo de datos indicado en la salida OUT (sólo válido para números enteros).

Sintaxis

Para la instrucción "Generar complemento a dos" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := NEG_<Tipo de datos>(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a dos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	INT, DINT, REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	INT, DINT, REAL	I, Q, M, D, L, P	Complemento a dos del valor de entrada

En la lista desplegable "<???>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := NEG_REAL("Tag_InValue")
```

La instrucción cambia el signo del valor en el operando "TagIn_Value" e indica el resultado en el operando "TagOut_Value".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ABS: Calcular valor absoluto

Descripción

Con la instrucción "Calcular valor absoluto" se calcula el valor absoluto del valor del operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- Un número en coma flotante tiene un valor no válido.

Sintaxis

Para la instrucción "Calcular valor absoluto" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ABS(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor absoluto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	REAL	I, Q, M, D, L, P	Valor absoluto del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ABS("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
TagIn_Value	-6.234
TagOut_Value	6.234

La instrucción calcula el valor absoluto del valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

MIN: Determinar mínimo

Descripción

La instrucción "Determinar mínimo" compara los valores de las entradas IN1, IN2 y IN3 y escribe el valor más pequeño en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

En caso de que la verificación CEI esté desactivada, se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, la instrucción devolverá un resultado no definido.

Sintaxis

Para la instrucción "Determinar mínimo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MIN <Tipo de datos>
     (IN1 := <Operando>
      IN2 := <Operando>
      IN3 := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar mínimo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MIN INT
     (IN1 := "TagIn_Value1"
      IN2 := "TagIn_Value2"
      IN3 := "TagIn_Value3"
      OUT => "Tag_Minimum"
```

GRAPH

```
)
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12222
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	Tag_Minimum	12222

La instrucción compara los valores de los operandos indicados y copia el valor menor ("TagIn_Value1") en la salida "Tag_Minimum".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

MAX: Determinar máximo

Descripción

La instrucción "Determinar máximo" compara los valores de las entradas IN1, IN2 y IN3 y escribe el valor más grande en la salida OUT. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no son del mismo tipo de datos.
- Un número en coma flotante tiene un valor no válido.

Nota

En caso de que la verificación CEI esté desactivada, se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, la instrucción devolverá un resultado no definido.

Sintaxis

Para la instrucción "Determinar máximo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MAX <Tipo de datos>
```

GRAPH

```
(IN1 := <Operando>
 IN2 := <Operando>
 IN3 := <Operando>
 OUT => <Operando>
 )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Determinar máximo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	INT, DINT, REAL	I, Q, M, D, L, P	Primer valor de entrada
IN2	Input	INT, DINT, REAL	I, Q, M, D, L, P	Segundo valor de entrada
IN3	Input	INT, DINT, REAL	I, Q, M, D, L, P	Tercer valor de entrada
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL MAX INT
      (IN1 := "TagIn_Value1"
       IN2 := "TagIn_Value2"
       IN3 := "TagIn_Value3"
       OUT => "Tag_Maximum"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
IN1	TagIn_Value1	12666
IN2	TagIn_Value2	14444
IN3	TagIn_Value3	13333
OUT	TagOut_Value	14444

La instrucción compara los valores de los operandos indicados y copia el valor mayor ("TagIn_Value2") en la salida "Tag_Maximum".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

LIMIT: Ajustar valor límite

Descripción

La instrucción "Ajustar valor límite" limita el valor en la entrada IN a los valores de las entradas MN y MX. Si el valor de la entrada IN cumple la condición MN <= IN <= MX, se copia en la salida OUT. Si no se cumple la condición y el valor de entrada IN es menor que el límite inferior MN, la salida OUT adopta el valor de la entrada MN. Si el límite superior MX se rebasa por exceso, la salida OUT adopta el valor de la entrada MX.

Si el valor de la entrada MN es mayor que el de la entrada MX, el resultado no está definido. Para poder ejecutar la instrucción, es imprescindible que las variables de todas las entradas sean del mismo tipo de datos.

El valor de la salida OUT no es válido si se cumple una de las siguientes condiciones:

- Las variables indicadas no son del mismo tipo de datos.
- Un operando tiene un valor no válido.
- El valor de la entrada MN es mayor que el valor de la entrada MX.

Nota

En caso de que la verificación CEI esté desactivada, se permiten únicamente los tipos de datos INT, DINT y REAL. Si utiliza los parámetros de entrada de los tipos WORD o DWORD, la instrucción devolverá un resultado no definido.

Sintaxis

Para la instrucción "Ajustar valor límite" se utiliza la siguiente sintaxis:

GRAPH

```
CALL LIMIT <Tipo de datos>
      (MN := <Operando>
       IN := <Operando>
       MX := <Operando>
       OUT => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Ajustar valor límite":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite inferior
IN	Input	INT, DINT, REAL	I, Q, M, D, L, P	Valor de entrada
MX	Input	INT, DINT, REAL	I, Q, M, D, L, P	Límite superior
OUT	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL LIMIT INT
      (MN := "Tag_LowLimit"
       IN := "Tag_InputValue"
       MX := "Tag_HighLimit"
       OUT => "Tag_Result"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor
MN	Tag_LowLimit	12000
IN	Tag_InputValue	8000
MX	Tag_HighLimit	16000
OUT	Tag_Result	12000

El valor del operando "Tag_InputValue" se compara con los valores de los operandos "Tag_LowLimit" y "Tag_HighLimit". Puesto que el valor del operando "Tag_InputValue" es menor que el valor límite inferior, el valor del operando "Tag_LowLimit" se copia en la salida "Tag_Result".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SQR: Calcular cuadrado

Descripción

Con la instrucción "Calcular cuadrado" se forma el cuadrado del valor del operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular cuadrado" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SQR(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular cuadrado":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	REAL	I, Q, M, D, L, P	Cuadrado del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SQR("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	5.0
Tag_OutValue	25.0

La instrucción eleva al cuadrado el valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SQRT: Calcular raíz cuadrada

Descripción

Con la instrucción "Calcular raíz cuadrada" se extrae la raíz cuadrada del valor del operando. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero se devuelve un número en coma flotante no válido. Si el valor del operando es "0", el resultado también es "0".

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando es negativo.

Sintaxis

Para la instrucción "Calcular raíz cuadrada" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SQRT(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular raíz cuadrada":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	REAL	I, Q, M, D, L	Raíz cuadrada del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SQRT("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	25.0
Tag_OutValue	5.0

La instrucción extrae la raíz cuadrada del valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

LN: Calcular logaritmo natural

Descripción

La instrucción "Calcular logaritmo natural" permite calcular el logaritmo natural en base e (e = 2.718282) del valor del operando. La instrucción da un resultado positivo si el valor de entrada es mayor que cero. En los valores de entrada menores que cero se devuelve un número en coma flotante no válido.

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando es negativo.

Sintaxis

Para la instrucción "Calcular logaritmo natural" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := LN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular logaritmo natural":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	REAL	I, Q, M, D, L, P	Logaritmo natural del valor de la entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := LN("Tag_InValue")
```

La instrucción calcula el logaritmo natural del valor del operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

EXP: Calcular valor exponencial

Descripción

La instrucción "Calcular valor exponencial" permite calcular la potencia en base e (e = 2,718282) del valor del operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor exponencial" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := EXP(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor exponencial":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	REAL	I, Q, M, D, L, P	Valor exponencial del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := EXP("Tag_InValue")
```

La instrucción calcula la potencia en base e del valor del operando "Tag_InValue" e indica el resultado (e^{IN}) en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SIN: Calcular valor de seno

Descripción

La instrucción "Calcular valor de seno" permite calcular el seno del ángulo. El tamaño del ángulo se indica en radianes en el operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de seno" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := SIN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de seno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
<Resultado>	Output	REAL	I, Q, M, D, L, P	Seno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SIN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	+1,570796 (π/2)
Tag_OutValue	1.0

La instrucción calcula el seno del ángulo indicado en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

COS: Calcular valor de coseno

Descripción

La instrucción "Calcular valor de coseno" permite calcular el coseno del ángulo. El tamaño del ángulo se indica en radianes en el operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de coseno" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := COS(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de coseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
<Resultado>	Output	REAL	I, Q, M, D, L, P	Coseno del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := COS("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	+1,570796 (π/2)
Tag_OutValue	0

La instrucción calcula el coseno del ángulo indicado en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

TAN: Calcular valor de tangente

Descripción

La instrucción "Calcular valor de tangente" permite calcular la tangente del ángulo. El tamaño del ángulo se indica en radianes en el operando.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de tangente" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := TAN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de tangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Tamaño del ángulo en radianes
<Resultado>	Output	REAL	I, Q, M, D, L, P	Tangente del ángulo indicado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := TAN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	+3,141593 (π)
Tag_OutValue	0

La instrucción calcula la tangente del ángulo indicado en el operando "Tag_InValue" e indica el resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ASIN: Calcular valor de arcoseno

Descripción

La instrucción "Calcular valor de arcoseno" permite calcular a partir del valor de seno indicado en el operando el tamaño del ángulo que corresponde a este valor. En el operando solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado puede estar comprendido entre -π/2 y +π/2.

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando está fuera del rango de valores permitido (-1 hasta +1).

Sintaxis

Para la instrucción "Calcular valor de arcoseno" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ASIN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de seno
<Resultado>	Output	REAL	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ASIN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1.0
Tag_OutValue	+1,570796 (π/2)

La instrucción calcula el tamaño del ángulo que corresponde al valor de seno del operando "Tag_InValue". El resultado de la instrucción se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ACOS: Calcular valor de arcocoseno

Descripción

La instrucción "Calcular valor de arcocoseno" permite calcular a partir del valor de coseno indicado en el operando el tamaño del ángulo que corresponde a este valor. En el operando solo se pueden indicar números en coma flotante válidos, comprendidos en un rango de valores entre -1 y +1. El tamaño del ángulo calculado puede estar comprendido entre 0 y +π.

El valor del resultado no es válido si se cumple una de las siguientes condiciones:

- El valor del operando no es un número en coma flotante válido.
- El valor del operando está fuera del rango de valores permitido (-1 hasta +1).

Sintaxis

Para la instrucción "Calcular valor de arcocoseno" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ACOS(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcocoseno":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de coseno
<Resultado>	Output	REAL	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := ACOS("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	0
Tag_OutValue	+1,570796 (π/2)

La instrucción calcula el tamaño del ángulo que corresponde al valor de coseno del operando "Tag_InValue". El resultado de la instrucción se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ATAN: Calcular valor de arcotangente

Descripción

La instrucción "Calcular valor de arcotangente" permite calcular a partir del valor de tangente indicado en el operando el tamaño del ángulo que corresponde a este valor. En el operando solo se pueden indicar números en coma flotante válidos. El tamaño del ángulo calculado puede estar comprendido entre -π/2 y +π/2.

El valor del resultado no es válido si se cumple la siguiente condición:

- El valor del operando no es un número en coma flotante válido.

Sintaxis

Para la instrucción "Calcular valor de arcotangente" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := ATAN(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Calcular valor de arcotangente":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de tangente
<Resultado>	Output	REAL	I, Q, M, D, L, P	Tamaño del ángulo en radianes

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ATAN("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1.0
Tag_OutValue	+0,785398 (π/4)

La instrucción calcula el tamaño del ángulo que equivale al valor de tangente del operando "Tag_InValue". El resultado de la instrucción se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Transferencia

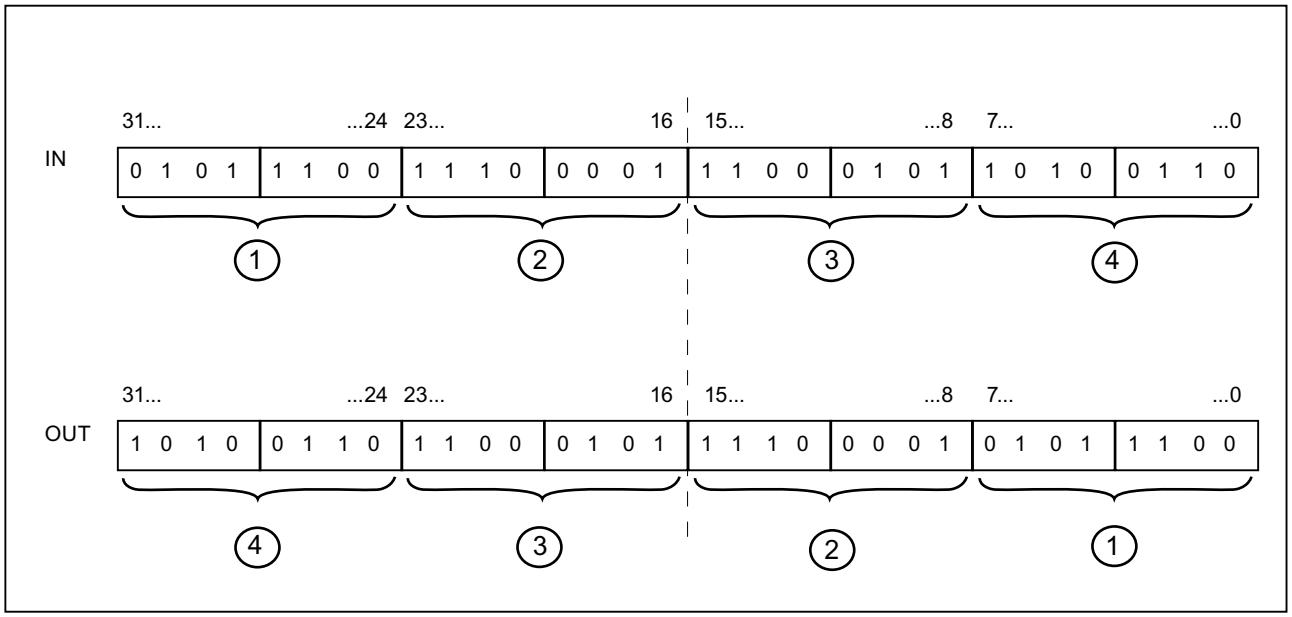
SWAP: Cambiar disposición

Descripción

Con la instrucción "Cambiar disposición" se modifica la disposición de los bytes del operando.

Dependiendo del tipo de datos utilizado, se puede cambiar el orden de los bytes en todo el acumulador 1 o solamente en la palabra derecha del acumulador 1.

La siguiente figura muestra cómo se cambian las posiciones de los bytes del operando del tipo de datos DWORD:



Disposición de los bytes en la palabra derecha del acumulador 1

Para el tipo de datos WORD se cambia la disposición de los bytes en la palabra derecha del acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor A	Valor B	Valor D	Valor C

El resultado de la instrucción se guarda en la palabra derecha del acumulador 1. Los bytes de la palabra izquierda del acumulador 1 no se ven influidos por la instrucción y no cambian.

Sintaxis

Para la instrucción "Cambiar disposición" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SWAP(<Operando>)
```

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Cambiar disposición":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	WORD, DWORD	I, Q, M, D, L o constante	Operando cuyas posiciones de bytes se cambian
<Resultado>	Output	WORD, DWORD	I, Q, M, D, L	Resultado

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SWAP("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor			
Tag_InValue	0000	1111	0000	1111
Tag_OutValue	0000	1111	1111	0000

Cambiar la disposición de los bytes en todo el acumulador 1:

Para el tipo de datos DWORD se modifica la disposición de los bytes en todo el acumulador 1.

La tabla siguiente muestra el contenido del acumulador 1 antes y después de ejecutar la instrucción:

Estado	Bytes del acumulador 1			
Antes de la ejecución	Valor A	Valor B	Valor C	Valor D
Después de la ejecución	Valor D	Valor C	Valor B	Valor A

El resultado de la instrucción se guarda en el acumulador 1. El contenido del acumulador 2 no cambia.

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor							
Tag_InValue	1111	0000	0000	1111	0000	0000	1111	1111
Tag_OutValue	1111	1111	0000	0000	0000	1111	1111	0000

La disposición de los bytes en el operando "Tag_InValue" se guarda cambiada en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

BLKMOV: Copiar área

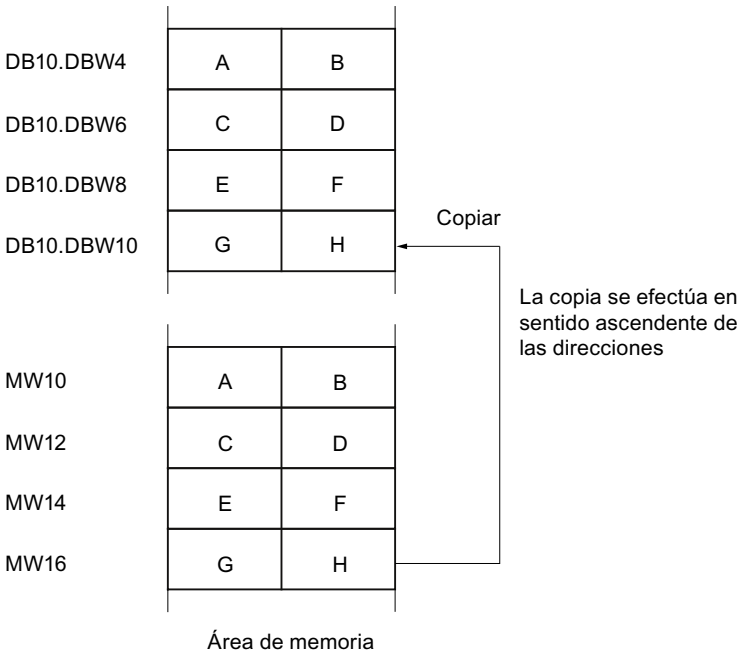
Descripción

La instrucción "Copiar área" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Copiar área" no se modifican los datos de origen. De lo contrario no está garantizada la coherencia de los datos de destino.

Posibilidad de interrupción

Mientras el área de origen no forme parte de un bloque de datos que solo figura en la memoria de carga, no hay limitaciones en cuanto a la profundidad de anidamiento.

Por el contrario, en caso de interrumpirse la ejecución de BLKMOV, en la que se copia desde un DB no relevante para la ejecución, no puede anidarse más la ejecución de la BLKMOV.

Áreas de memoria

La instrucción "Copiar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Si las áreas de origen y destino son de diferente longitud, se copia solo hasta la longitud del área más pequeña.

Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPU S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPU V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPU S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si solo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima también se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados.

Si se desea copiar información sobre la longitud máxima y real de una cadena de caracteres, las áreas de los parámetros SRCBLK y DSTBLK deben indicarse en bytes.

Reglas para copiar bloques de datos no relevantes para la ejecución

El área de origen también puede encontrarse en un bloque de datos no relevante para la ejecución en la memoria de carga. Los bloques de datos no relevantes para la ejecución se identifican con la palabra clave UNLINKED.

Si en la memoria de trabajo se copia un bloque de datos no relevante para la ejecución con la instrucción "Copiar área" y, al mismo tiempo, este se vuelve a cargar p. ej. desde la programadora, la ejecución de la instrucción puede retardarse varios milisegundos. Esto prolonga el ciclo del OB y puede activar la vigilancia del tiempo de ciclo.

Si un bloque de datos no relevante para la ejecución se copia con la instrucción "Copiar área" y la operación de copia se interrumpe, la instrucción no se podrá seguir ejecutando.

Si en la CPU existe la instrucción "Leer de un bloque de datos de la memoria de carga", debe utilizarse dicha instrucción para leer los bloques de datos no relevantes para la ejecución en la memoria de carga. Si se utiliza la instrucción "Copiar área" se emite el error W#16#8092.

Sintaxis

Para la instrucción "Copiar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL BLKMOV ANY
     (SRCBLK := <Operando>
      RET_VAL => <Operando>
      DSTBLK => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando ocurre un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8092	El área de origen o destino está solo en la memoria de carga.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL BLKMOV ANY
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

La instrucción copia 10 bytes a partir de MB100 y los escribe en el DB1. Si ocurre un error durante la operación de copia, el respectivo código de error se indica en la variable "Tag_ErrorCode".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

UBLKMOV: Copiar área sin interrupciones

Descripción

La instrucción "Copiar área sin interrupciones" permite copiar los contenidos de un área de memoria (área de origen) en un área de memoria diferente (área de destino). La operación de copia se realiza por orden ascendente de direcciones. Las áreas de origen y destino se definen mediante punteros ANY.

La operación de copia no debe ser interrumpida por otras actividades del sistema operativo. Ello puede aumentar el tiempo de reacción a alarmas de la CPU durante la ejecución de la instrucción "Copiar área sin interrupciones".

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

Áreas de memoria

La instrucción "Copiar área sin interrupciones" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas

Reglas generales para copiar

Al ejecutar la instrucción "Copiar área sin interrupciones" no pueden solaparse las áreas de origen y destino. Si el área de origen es más pequeña que el área de destino, el área de origen se copiará por completo en el área de destino. Los restantes bytes del área de destino permanecen inalterados.

Si el área de destino es más pequeña que el área de origen, se escribirá en toda el área de destino. Los restantes bytes del área de origen se ignoran.

Si un área de origen o destino definida como parámetro formal es menor que un área de origen o destino indicada en los parámetros SRCBLK o DSTBLK, no se transferirán datos.

Al copiar un área del tipo de datos BOOL, la longitud indicada del área debe ser un múltiplo de 8, pues de lo contrario no se ejecutará la instrucción.

La instrucción "Copiar área sin interrupciones" permite copiar 512 bytes como máximo. A este respecto deben tenerse en cuenta las limitaciones específicas de la CPU.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro SRCBLK o DSTBLK, el comportamiento dependerá del tipo de CPU:

- Para las CPUs S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPUs V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPUs S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Reglas para copiar cadenas de caracteres

La instrucción "Copiar área sin interrupciones" permite copiar asimismo áreas de origen y destino del tipo de datos STRING. Si sólo el área de origen es del tipo de datos STRING, se copiarán los caracteres realmente contenidos en la cadena de caracteres. La información sobre la longitud real y máxima no se escribe en el área de destino. Si tanto el área de origen como el área de destino son del tipo de datos STRING, la longitud actual de la cadena de caracteres en el área de destino se pone al número de caracteres realmente copiados. Si se copian áreas del tipo de datos STRING, debe indicarse "1" como longitud del área.

Sintaxis

Para la instrucción "Copiar área sin interrupciones" se utiliza la siguiente sintaxis:

GRAPH

```
CALL UBLKMOV ANY
     (SRCBLK := <Operando>
      RET_VAL => <Operando>
      DSTBLK => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Copiar área sin interrupciones":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRCBLK	Input	ANY	I, Q, M, D, L, P	Indicación del área de memoria que se copia (área de origen).
RET_VAL	Output	INT	I, Q, M, D, L, P	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
DSTBLK	Output	ANY	I, Q, M, D, L, P	Indicación del área de memoria en la que se copia (área de destino).

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8091	El área de origen se encuentra en un bloque de datos no relevante para la ejecución.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL UBLKMOV ANY
     (SRCBLK := P#M100.0 BYTE 10
      RET_VAL => "Tag_ErrorCode"
      DSTBLK => P#DB1.DBX0.0 BYTE 10
      )
```

La instrucción copia 10 bytes a partir de MB100 y los escribe en el DB1. Si ocurre un error durante la operación de copia, el respectivo código de error se indica en la variable "Tag_ErrorCode".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

FILL: Rellenar área

Descripción

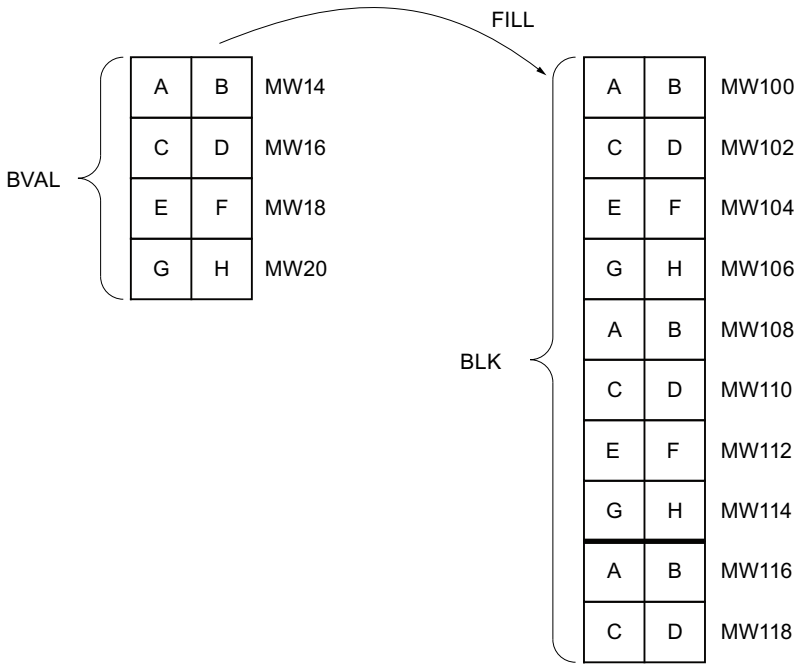
La instrucción "Rellenar área" permite rellenar un área de memoria (área de destino) con el contenido de un área de memoria diferente (área de origen). La instrucción "Rellenar área" copia el contenido del área de origen en el área de destino tantas veces hasta que el área de destino quede escrita por completo. La operación de copia se realiza por orden ascendente de direcciones.

Las áreas de origen y destino se definen mediante punteros ANY.

Nota

En esta instrucción solo se pueden utilizar tipos de datos que sean válidos en una CPU de la serie S7-300/400.

La figura siguiente muestra el principio de la operación de copia:



Ejemplo: El contenido del área MW100 a MW118 debe inicializarse con el contenido de las palabras de marcas MW14 a MW20.

Coherencia de los datos de origen y destino

Tenga en cuenta que al ejecutar la instrucción "Rellenar área" no se modifican los datos de origen, pues de lo contrario ya no está garantizada la coherencia de los datos de destino.

Áreas de memoria

La instrucción "Rellenar área" permite copiar las siguientes áreas de memoria:

- Áreas de un bloque de datos
- Marcas
- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Bloques de datos no relevantes para la ejecución

Reglas generales para copiar

Las áreas de origen y destino no deben solaparse. Aunque el área de destino que se debe inicializar no sea múltiplo entero de la longitud del parámetro de entrada BVAL, el área de destino se rellenará igualmente hasta el último byte.

Si el área de destino que se debe inicializar es menor que el área de origen, sólo se copian los datos que caben en el área de destino.

Si el área de destino o de origen disponible realmente es menor que el tamaño del área de memoria parametrizado para el área de origen o de destino (parámetros BVAL, BLK), los datos no se transfieren.

Si el puntero ANY (origen o destino) es del tipo BOOL, la longitud indicada debe ser divisible por 8. De lo contrario, la instrucción no se ejecutará.

Si el área de destino es del tipo de datos STRING, la instrucción describe la cadena de caracteres completa, incluida la información de administración.

Nota

Si el área de origen o destino realmente existente es menor que el área de origen o destino indicada en el parámetro BVAL o BLK, el comportamiento dependerá del tipo de CPU:

- Para las CPUs S7-300 rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#837F. Este código de error debe evaluarse en el programa de usuario.
- Para las CPUs V4 del S7-400 (V4, CPU H V4 y CPU H V4.5.) rige lo siguiente:
  No se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.
- Para las restantes CPUs S7-400 rige:
  Se transfieren datos. En el parámetro RET_VAL se devuelve el código de error W#16#8122 o W#16#8323. Este código de error debe evaluarse en el programa de usuario.

Sintaxis

Para la instrucción "Rellenar área" se utiliza la siguiente sintaxis:

GRAPH

```
CALL FILL ANY
     (BVAL := <Operando>
      RET_VAL => <Operando>
      BLK => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rellenar área":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BVAL	Input	ANY	I, Q, M, D, L	Indicación del área de memoria (área de origen) con cuyo contenido se rellena el área de destino en el parámetro BLK.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
BLK	Output	ANY	I, Q, M, D, L	Indicación del área de memoria que se rellena con el contenido del área de origen.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL FILL ANY
```

GRAPH

```
(BVAL := P#M14.0 WORD 4
 RET_VAL => "Tag_ErrorCode"
 BLK => P#M100.0 WORD 10
 )
```

La instrucción copia el área de origen de MW14 a MW20 y rellena el área de destino de MW100 a MW118 con el contenido de las 4 palabras contenidas en el área de memoria del parámetro BVAL.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Conversión

CONVERT: Convertir valor

Descripción

La instrucción "Convertir valor" lee el contenido del operando y lo convierte según los tipos de datos parametrizados.

Encontrará información relacionada con las conversiones posibles en el apartado "Conversión explícita" de "Consulte también".

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Convertir valor" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ???_TO_???(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Convertir valor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Enteros, números en coma flotante, BCD16, BCD32	I, Q, M, D, L, P o constante	Valor que se convierte
<Resultado>	Output	Enteros, números en coma flotante, BCD16, BCD32	I, Q, M, D, L, P	Resultado de la conversión

En las listas desplegables "<???>" puede seleccionar los tipos de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := INT_TO_DINT("Tag_InValue")
```

El contenido del operando "Tag_InValue" se lee y se convierte en un entero (32 bits). El resultado se deposita en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ROUND: Redondear número

Descripción

La instrucción "Redondear número" permite redondear el valor del operando al siguiente número entero. La instrucción interpreta el valor como número en coma flotante y lo convierte a un número entero del tipo de datos DINT. Si el valor de entrada se encuentra entre un número par y uno impar, se selecciona el número par.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Redondear número" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ROUND(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear número":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se redondea
<Resultado>	Output	DINT	I, Q, M, D, L, P	Resultado del redondeo

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ROUND("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	2	-2

El número en coma flotante del operando "Tag_InValue" se redondea al siguiente número entero par y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

CEIL: Redondear un número en coma flotante al siguiente entero superior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero superior" permite redondear el valor del operando al siguiente número entero superior. La instrucción interpreta el valor como número en coma flotante y lo convierte al siguiente número entero superior. El resultado de la instrucción puede ser mayor o igual al valor de entrada.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero superior" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := CEIL(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero superior":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado con el siguiente entero superior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := CEIL("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	1	0

El número en coma flotante del operando "Tag_InValue" se redondea al siguiente número entero superior y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

FLOOR: Redondear un número en coma flotante al siguiente entero inferior

Descripción

La instrucción "Redondear un número en coma flotante al siguiente entero inferior" permite redondear el valor del operando al siguiente número entero inferior. La instrucción interpreta el valor como número en coma flotante y lo convierte al siguiente número entero inferior. El resultado de la instrucción puede ser menor o igual al valor de entrada.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Redondear un número en coma flotante al siguiente entero inferior" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := FLOOR(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Redondear un número en coma flotante al siguiente entero inferior":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	INT, DINT, REAL	I, Q, M, D, L, P	Resultado con el siguiente entero inferior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH
```
"Tag_OutValue" := FLOOR("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	0.50000000	-0.50000000
Tag_OutValue	0	-1

El número en coma flotante del operando "Tag_InValue" se redondea al siguiente número entero inferior y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

TRUNC: Truncar a entero

Descripción

La instrucción "Truncar a entero" permite generar un número entero a partir del valor del operando. El valor se interpreta como número en coma flotante. La instrucción selecciona solo la parte entera del número en coma flotante y la indica sin decimales como resultado.

El valor del resultado no es válido si se cumple la siguiente condición:

- Durante la ejecución se producen errores, p. ej. se produce un rebase por exceso.

Sintaxis

Para la instrucción "Truncar a entero" se utiliza la siguiente sintaxis:

GRAPH
```
<Resultado> := TRUNC(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Truncar a entero":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	REAL	I, Q, M, D, L o constante	Valor de entrada
<Resultado>	Output	INT, DINT	I, Q, M, D, L	Parte entera del valor de entrada

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := TRUNC("Tag_InValue")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor	
Tag_InValue	1.50000000	-1.50000000
Tag_OutValue	1	-1

La parte entera del número en coma flotante del operando "Tag_InValue" se convierte en un número entero y se indica en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SCALE: Escalar

Descripción

La instrucción "Escalar" convierte el entero del parámetro IN en un número en coma flotante que se escala en unidades físicas entre un límite inferior y uno superior. Los límites inferior y superior del rango de valores en el que se escala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Escalar" utiliza la siguiente ecuación:

OUT = [((FLOAT (IN) – K1)/(K2–K1)) * (HI_LIM–LO_LIM)] + LO_LIM

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el de la constante "K2", el resultado de la instrucción se pone al valor del límite superior (HI_LIM) y se emite un error.

Si el valor del parámetro IN es menor que el de la constante "K1", el resultado de la instrucción se pone al valor del límite inferior (LO_LIM) y se emite un error.

Si el límite inferior indicado es mayor que el límite superior (LO_LIM > HI_LIM), el resultado se escala de forma inversamente proporcional al valor de entrada.

Sintaxis

Para la instrucción "Escalar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SCALE
     (IN := <Operando>
      HI_LIM := <Operando>
      LO_LIM := <Operando>
      BIPOLAR := <Operando>
      RET_VAL => <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Escalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	INT	I, Q, M, D, L, P o constante	Valor de entrada que se escala
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el de la constante "K2" o menor que el de la constante "K1".
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SCALE
      (IN := "Tag_InputValue"
       HI_LIM := "Tag_HighLimit"
       LO_LIM := "Tag_LowLimit"
       BIPOLAR := "Tag_Bipolar"
       RET_VAL => "Tag_ErrorCode"
       OUT => "Tag_OutputValue"
       )
```

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0

Parámetro	Operando	Valor
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	22
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	50.03978588

En el operando "Tag_InputValue" se indica el valor que debe convertirse y escalarse. El valor límite superior y el inferior se definen mediante los operandos "Tag_HighLimit" y "Tag_LowLimit". Con ayuda del operando "Tag_Bipolar" = TRUE se indica que el valor del parámetro IN debe interpretarse como bipolar. En el operando "Tag_OutputValue" se indica el resultado de la instrucción.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

UNSCALE: Desescalar

Descripción

La instrucción "Desescalar" permite desescalar el número en coma flotante del parámetro IN en unidades físicas comprendidas entre un límite inferior y uno superior y convertirlo en un número entero. Los límites inferior y superior del rango de valores en el que se desescala el valor de entrada se definen mediante los parámetros LO_LIM y HI_LIM. El resultado de la instrucción se devuelve en el parámetro OUT.

La instrucción "Desescalar" utiliza la siguiente ecuación:

OUT = [((IN–LO_LIM)/(HI_LIM–LO_LIM)) * (K2–K1)] + K1

El estado lógico del parámetro BIPOLAR determina los valores de las constantes "K1" y "K2". El parámetro BIPOLAR puede tener los siguientes estados lógicos:

- Estado lógico "1": se presupone que el valor del parámetro IN es bipolar y está comprendido en un rango de valores entre -27648 y 27648. En este caso, la constante "K1" tiene el valor "-27648,0" y la constante "K2", el valor "+27648,0".
- Estado lógico "0": se presupone que el valor del parámetro IN es unipolar y está comprendido en un rango de valores entre 0 y 27648. En este caso, la constante "K1" tiene el valor "0,0" y la constante "K2", el valor "+27648,0".

Si el valor del parámetro IN es mayor que el del límite superior (HI_LIM), el resultado de la instrucción se pone al valor de la constante "K2" y se emite un error.

Si el valor del parámetro IN es menor que el de la constante del límite inferior (LO_LIM), el resultado de la instrucción se pone al valor de la constante "K1" y se emite un error.

Sintaxis

Para la instrucción "Desescalar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL UNSCALE
      (IN := <Operando>
       HI_LIM := <Operando>
       LO_LIM := <Operando>
       BIPOLAR := <Operando>
       RET_VAL => <Operando>
       OUT => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desescalar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada que se desescala en un valor entero
HI_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite superior
LO_LIM	Input	REAL	I, Q, M, D, L, P o constante	Valor límite inferior
BIPOLAR	Input	BOOL	I, Q, M, D, L o constante	Indica si el valor del parámetro IN se interpreta como bipolar o unipolar. El parámetro puede adoptar los valores siguientes: 1: Bipolar 0: Unipolar

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	INT	I, Q, M, D, L, P	Resultado de la instrucción
RET_VAL	Output	WORD	I, Q, M, D, L, P	Información de error

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0008	El valor del parámetro IN es mayor que el del límite superior (HI_LIM) o menor que el del límite inferior (LO_LIM).
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL UNSCALE
     (IN := "Tag_InputValue"
      HI_LIM := "Tag_HighLimit"
      LO_LIM := "Tag_LowLimit"
      BIPOLAR := "Tag_Bipolar"
      RET_VAL => "Tag_ErrorCode"
      OUT => "Tag_OutputValue"
      )
```

La tabla siguiente muestra los valores de los distintos operandos antes de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	0.0

La tabla siguiente muestra los valores de los distintos operandos después de ejecutar la instrucción:

Parámetro	Operando	Valor
IN	Tag_InputValue	50.03978588
HI_LIM	Tag_HighLimit	100.0
LO_LIM	Tag_LowLimit	0.0
BIPOLAR	Tag_Bipolar	1
RET_VAL	Tag_ErrorCode	W#16#0000
OUT	Tag_OutputValue	22

En el operando "Tag_InputValue" se indica el valor que debe convertirse y desescalarse. El valor límite superior y el inferior se definen mediante los operandos "Tag_HighLimit" y "Tag_LowLimit". Con ayuda del operando "Tag_Bipolar" = TRUE se indica que el valor del parámetro IN debe interpretarse como bipolar. En el operando "Tag_OutputValue" se indica el resultado de la instrucción.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Control del programa

Control en runtime

COMPRESS: Comprimir la memoria de la CPU

Descripción

La instrucción "Comprimir la memoria de la CPU" permite comprimir la memoria de trabajo de la CPU o la memoria de carga RAM. Los huecos creados por haber borrado o cargado bloques, se rellenan juntando los bloques existentes.

Con la llamada de la instrucción "Comprimir la memoria de la CPU" se lanza el proceso de compresión. El proceso de compresión se reparte entre varios ciclos del programa. La vigilancia del proceso de compresión se realiza mediante los valores de los distintos parámetros. Si el parámetro BUSY tiene el valor "1", significa que el proceso de compresión se está ejecutando todavía. El valor "1" del parámetro DONE indica que ha finalizado el proceso de compresión. La información de error puede consultarse en el parámetro RET_VAL.

Si se está ejecutando un proceso de compresión lanzado externamente, la llamada de la instrucción "Comprimir la memoria de la CPU" origina un error.

Sintaxis

Para la instrucción "Comprimir la memoria de la CPU" se utiliza la siguiente sintaxis:

GRAPH

```
CALL COMPRESS
     (RET_VAL => <Operando>
      BUSY => <Operando>
      DONE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comprimir la memoria de la CPU":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Output	INT	I, Q, M, D, L	Información de error: El parámetro RET_VAL devuelve un código de error cuando se produce un error durante la ejecución de la instrucción.
BUSY	Output	BOOL	I, Q, M, D, L	Información de si está activo el proceso de compresión
DONE	Output	BOOL	I, Q, M, D, L	Información de si ha finalizado el proceso de compresión

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error El estado del proceso de compresión puede consultarse en los parámetros BUSY y DONE.
8091	Ya se está ejecutando un proceso de compresión lanzado externamente.
8092	La instrucción no se puede ejecutar porque se cumple una de las condiciones siguientes: • La función "Borrar bloque" está activa. • Una función de test y de puesta en servicio repercute sobre un bloque (p. ej. estado). • La función "Copiar bloques" se ha activado por un impulso externo. • La CPU H se encuentra en un proceso de acoplamiento o actualización.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo

Descripción

La instrucción "Reiniciar tiempo de vigilancia del ciclo" permite reiniciar la vigilancia del ciclo de la CPU. El tiempo de vigilancia del ciclo se reinicia entonces con la duración ajustada en la configuración de la CPU. Reiniciando el tiempo de vigilancia del ciclo es posible impedir que se produzcan errores o que la CPU pase a modo STOP.

La instrucción "Reiniciar tiempo de vigilancia del ciclo" puede utilizarse en bloques de la clase de prioridad 1 (el OB de ciclo) así como en los bloques que se llaman desde ellos.

Si la instrucción se llama desde un bloque de prioridad superior, p. ej. en una alarma de proceso, diagnóstico o en una alarma cíclica, no se ejecuta la instrucción y la salida de habilitación ENO se pone al estado lógico "0".

Nota

En caso de utilizar la instrucción "Reiniciar tiempo de vigilancia del ciclo", debe asegurarse que no se produce un bucle sin fin en la ejecución cíclica del programa, es decir, en el OB1. De lo contrario, la CPU ya no alcanza el punto de control del ciclo. Como consecuencia no pueden ejecutarse ciertas funciones de la CPU (p. ej., la actualización de la memoria imagen de proceso).

Sintaxis

Para la instrucción "Reiniciar tiempo de vigilancia del ciclo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL RE_TRIGR()
```

Parámetro

La instrucción "Reiniciar tiempo de vigilancia del ciclo" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

STP: Finalizar programa

Descripción

La instrucción "Finalizar programa" conmuta la CPU al estado operativo STOP, con lo que se finaliza la ejecución del programa. Los efectos al conmutar de RUN a STOP dependen de la configuración de la CPU.

Sintaxis

Para la instrucción "Finalizar programa" se utiliza la siguiente sintaxis:

GRAPH

```
CALL STP()
```

Parámetros

La instrucción "Finalizar programa" no tiene parámetros y no devuelve información de error.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

WAIT: Programar retardo

Descripción

La instrucción "Programar retardo" detiene la ejecución del programa durante un periodo parametrizado. El periodo se indica en microsegundos en el parámetro WT.

Es posible programar tiempos de retardo de hasta 32767 microsegundos (µs). El tiempo de retardo mínimo depende de la CPU utilizada y equivale al tiempo de ejecución de la instrucción "Programar retardo".

Los eventos de prioridad superior pueden interrumpir la ejecución de la instrucción.

La instrucción "Programar retardo" no devuelve información de error.

Sintaxis

Para la instrucción "Programar retardo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL WAIT
     (WT := <Operando>
```

GRAPH
)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Programar retardo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
WT	Input	INT	I, Q, M, D, L, P o constante	Tiempo de retardo en microsegundos (µs)

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

PROTECT: Cambiar nivel de protección

Descripción

La instrucción "Cambiar nivel de protección" permite modificar el nivel de protección "1" configurado para la CPU.

El nivel de protección "1" parametrizado con la opción activada "Eliminable por contraseña" hace que sea necesario parametrizar una contraseña. Esta contraseña permite durante el funcionamiento acceder en escritura a la CPU protegida contra escritura mediante la instrucción "Cambiar nivel de protección" (p. ej. cargar).

El nivel de protección se especifica en el parámetro MODE de la instrucción. El nivel de protección actual de la CPU se visualiza en la ventana de inspección en "Propiedades > Protección".

La tabla siguiente muestra una explicación de los niveles de protección:

Nivel de protección	Explicación
1	Se permiten todas las funciones de PG. En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
2	• El programa y la configuración de la CPU no se pueden modificar. El programa contenido en la CPU se puede leer. • Se permiten las funciones de control, observación y comunicación del proceso. • Se permiten todas las funciones de información. • Si conoce la contraseña válida, podrá desactivar la protección contra escritura ajustada. • En el caso que exista un bloqueo de la legitimación de la contraseña, este se elimina llamando la instrucción "Cambiar nivel de protección".
3	• Aun conociendo la contraseña válida, no podrá desactivar la protección contra lectura y escritura ajustada. • Si al llamar la instrucción "Cambiar nivel de protección" existe una conexión legitimada, la llamada de la instrucción no tendrá ningún efecto en esta conexión.

Si se ha configurado el nivel de protección "2" o "3" para la CPU, la llamada de la instrucción "Cambiar nivel de protección" no tiene efecto alguno.

La efectividad del nivel de protección "2" o "3" una vez ejecutada la instrucción "Cambiar nivel de protección" puede consultarse en la lista parcial SZL con la ID de SZL W#16#0232 y el índice W#16#0004.

Nota

La instrucción "Cambiar nivel de protección" permite ajustar un nivel de protección más bajo que aquel ajustado con "Configurar hardware" de STEP 7. Supongamos que se ha configurado el nivel de protección 3 y a continuación se llama la instrucción "Cambiar nivel de protección" con el parámetro MODE = 12. Esto bloqueará la legitimación de la contraseña. Si la instrucción se llama nuevamente con el parámetro MODE = 0 o 1, el nivel de protección 3 se mantiene y será posible desactivar la protección de lectura y escritura introduciendo la contraseña válida.

Efectos de las transiciones de estado operativo en el nivel de protección

La tabla siguiente muestra qué efectos tienen las transiciones de estado operativo en el nivel de protección ajustado con la instrucción "Cambiar nivel de protección".

Acción	Efectos en el nivel de protección
• Selector de modo en "STOP" • Conexión (POWER ON) sin respaldo • Borrado total mediante el selector de modo • Cambios de configuración y/o del programa en el estado operativo "STOP" de la CPU S7-400	El nivel de protección se pone a "1". Se permiten todas las funciones de PG.
• Arranque en frío • Rearranque (en caliente)	El sistema operativo pone el nivel de protección a "1". Se permiten todas las funciones de PG. En caso necesario, llamando la instrucción "Cambiar nivel de protección" en el programa es posible ajustar el nivel de protección en "2" o "3".
Conexión (POWER ON) respaldada	No cambia
Transición de estado operativo RUN/ARRANQUE/PARADA -> STOP (debida a la llamada de la instrucción "Finalizar programa" (STP), a errores del programa sin OB de error asociado o a una acción del usuario).	No cambia
S7-400: Rearranque por acción del usuario o automáticamente tras conexión (POWER ON)	No cambia

Sintaxis

Para la instrucción "Cambiar nivel de protección" se utiliza la siguiente sintaxis:

GRAPH

```
CALL PROTECT
      (MODE := <Operando>
       RET_VAL => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Cambiar nivel de protección":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	WORD	I, Q, M, D, L o constante	Nivel de protección Valores posibles: • W#16#0000: Ajuste del nivel de protección "1" • W#16#0001: Ajuste del nivel de protección "2" con legitimación de la contraseña • W#16#000C: Ajuste del nivel de protección "3" sin legitimación de la contraseña En el parámetro MODE también se pueden indicar constantes.
RET_VAL	Output	INT	I, Q, M, D, L	Si ocurre un error durante la ejecución, se emite una información de error.

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
8090	Valor no admisible en el parámetro MODE.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

CiR: Controlar proceso CiR

Descripción

La instrucción "Controlar proceso CiR" permite influir en el proceso CiR en el programa de usuario como se indica a continuación:

- Bloquear el proceso CiR. En este caso, se rechazará siempre la carga de una configuración modificada desde la programadora en la CPU. El bloqueo se aplica hasta que se desbloquee con la instrucción "Controlar proceso CiR".
- Bloqueo condicional del proceso CiR estableciendo un límite superior en el parámetro FRZ_TIME para el tiempo de sincronización CiR. En este caso, sólo se permitirá cargar una configuración modificada desde la programadora en la CPU si el tiempo de sincronización CiR calculado en la CPU no excede el límite superior definido.
- Desbloquear el proceso CiR. El límite superior del tiempo de sincronización CiR se establece en el valor predeterminado de 1000 ms.
- Determinar si el proceso CiR está o no habilitado. Si el proceso CiR está habilitado o lo está condicionadamente, el límite superior actual del tiempo de sincronización CiR se indica en el parámetro A_FT de la instrucción.

Nota

Durante el tiempo de sincronización CiR, las salidas están congeladas y las entradas no se evalúan.

Sintaxis

Para la instrucción "Controlar proceso CiR" se utiliza la siguiente sintaxis:

GRAPH

```
CALL CIR
      (MODE := <Operando>
       FRZ_TIME := <Operando>
       RET_VAL => <Operando>
       A_FT => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Controlar proceso CiR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición Los valores admisibles del parámetro MODE son: • 0: Función de información • 1: Habilitar proceso CiR • 2: Bloquear proceso CiR • 3: Bloquear proceso CiR de forma condicionada. El límite superior del tiempo de sincronización CiR se define en FRZ_TIME.
FRZ_TIME	Input	TIME	I, Q, M, D, L o constante	Límite superior del tiempo de sincronización CiR en milisegundos. Rango de valores admisible: 60 ... 2500 ms (valor predeterminado: 1000 ms) El parámetro FRZ_TIME sólo es relevante cuando el parámetro MODE se establece en el valor "3".
RET_VAL	Output	INT	I, Q, M, D, L	Información de error Si el parámetro MODE está a "0", el parámetro RET_VAL indica si el proceso CiR está habilitado o no.
A_FT	Output	TIME	I, Q, M, D, L	Límite superior válido actualmente del tiempo de sincronización CiR

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor 1, 2 ó 3.
0001	El proceso CiR está habilitado. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor "0".
0002	El proceso CiR está bloqueado. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor "0".

Código de error* (W#16#...)	Explicación
0003	El proceso CiR está bloqueado de forma condicionada. Este código de error sólo se puede emitir si el parámetro MODE está puesto al valor "0".
8001	La CPU no está lista para un proceso CiR. Se está utilizando una CPU H en el sistema H (modo "Individual") o una CPU estándar en modo multiprocesador.
8002	El valor del parámetro MODE no es admisible.
8003	El valor del parámetro FRZ_TIME no es admisible.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

La instrucción "Controlar proceso CiR" permite programar p. ej. que no se lance ningún proceso CiR en periodos en los que se requiera la capacidad máxima de la CPU para el procesamiento.

Para ello, debe llamarse la instrucción "Controlar proceso CiR" en el programa antes de comenzar el periodo con una mayor actividad de procesos y ejecutarla con el valor "2" (Bloquear proceso CiR) en el parámetro MODE. Con ello se bloquea el proceso CiR.

Para volver a habilitar el proceso CiR tras finalizar el periodo de actividad aumentada del proceso, la instrucción "Controlar proceso CiR" debe llamarse de nuevo y ejecutarse con el valor "1" en el parámetro MODE. De manera alternativa también puede establecer el parámetro MODE en el valor "3" y bloquear así el proceso CiR.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Operaciones lógicas con palabras

NOT (INV): Invertir

Descripción

La instrucción "Invertir" permite invertir el estado lógico de los bits del operando. Al ejecutar la instrucción, el valor del operando se combina con una operación lógica O-exclusiva utilizando una plantilla hexadecimal (W#16#FFFF para números de 16 bits o DW#16#FFFF FFFF para números de 32 bits). Esto invierte el estado lógico de los distintos bits y lo devuelve como resultado.

Sintaxis

Para la instrucción "Invertir" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := NOT(<Operando>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Invertir":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando>	Input	Secuencias de bits, enteros	I, Q, M, D, L, P o constante	Valor de entrada
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L, P	Complemento a 1 del valor del operando

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"TagOut_Value" := NOT("TagIn_Value")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
TagIn_Value	W#16#000F
TagOut_Value	W#16#FFF0

La instrucción invierte el estado lógico de los distintos bits del operando "TagIn_Value" y escribe el resultado en el operando "TagOut_Value".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

DECO: Descodificar

Descripción

La instrucción "Descodificar" activa un bit predeterminado por el valor de entrada en el valor de salida.

La instrucción "Descodificar" lee el valor de la entrada IN y activa el bit del parámetro OUT cuya posición de bit equivale al valor leído. Los demás bits del valor de salida se rellenan con ceros. Si el valor de la entrada IN es mayor que 31, se ejecuta una instrucción modulo 32.

Sintaxis

Para la instrucción "Descodificar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DECO <Tipo de datos>_DWORD
      (IN := <Operando>
       OUT => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Descodificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	WORD	I, Q, M, D, L, P o constante	Posición del bit que se activa en el valor de salida.
OUT	Output	DWORD	I, Q, M, D, L, P	Valor de salida

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

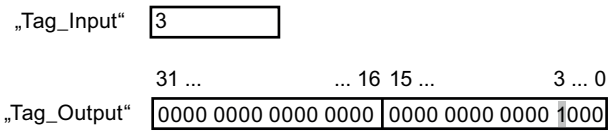
Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL DECO INT_DWORD
      (IN := "Tag_Input"
       OUT => "Tag_Output"
       )
```

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:



La instrucción lee el número de bit "3" del valor del operando "Tag_Input" de la entrada y activa el tercer bit del valor del operando "Tag_Output" de la salida.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ENCO: Codificar

Descripción

La instrucción "Codificar" permite leer el número del bit menos significativo que está activado en el valor de entrada y depositarlo en la salida OUT.

La instrucción "Codificar" selecciona el bit menos significativo del valor de la entrada IN y escribe su número de bit en la variable de la salida OUT. Si la entrada IN devuelve el valor DW#16#00000001 o DW#16#00000000, se deposita el valor "0" en el parámetro OUT.

Sintaxis

Para la instrucción "Codificar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL ENCO DWORD
     (IN := <Operando>
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Codificar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Valor de entrada
OUT	Output	INT	I, Q, M, D, L, P	Valor de salida

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL ENCO DWORD
      (IN := "Tag_Input"
       OUT => "Tag_Output"
       )
```

La figura siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

	31 16	15 ... 3 ... 0
"Tag_Input"	0000 1111 0000 0101	0000 1001 0000 1000

"Tag_Output"	3

La instrucción selecciona el bit menos significativo que está activado en la entrada "TagInput" y escribe el bit "3" en la variable de la salida "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SEL: Seleccionar

Descripción

La instrucción "Seleccionar" permite seleccionar, en función de un interruptor (entrada G), una de las entradas IN0 o IN1 y copiar su contenido en la salida OUT. Si la entrada G devuelve el estado lógico "0", se copia el valor de la entrada IN0. Si la entrada G devuelve el estado lógico "1", se copia el valor de la entrada IN1 en la salida OUT.

Para poder procesar la instrucción, las variables de todos los parámetros deben ser del mismo tipo de datos.

Sintaxis

Para la instrucción "Seleccionar" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SEL <Tipo de datos>
      (G := <Operando>
       IN0 := <Operando>
       IN1 := <Operando>
```

GRAPH

```
      OUT => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Seleccionar":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
G	Input	BOOL	I, Q, M, D, L, T, C	Interruptor
IN0	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Primer valor de entrada
IN1	Input	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Segundo valor de entrada
OUT	Output	Secuencias de bits, enteros, números en coma flotante, temporizadores, CHAR, TOD, DATE	I, Q, M, D, L, P	Resultado

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SEL INT
      (G := "Tag_Input_G"
       IN0 := "Tag_Input0"
       IN1 := "Tag_Input1"
       OUT => "Tag_Output"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
G	Tag_Input_G	0	1
IN0	Tag_Input0	W#16#0000	W#16#4C
IN1	Tag_Input1	W#16#FFFF	W#16#5E
OUT	Tag_Output	W#16#0000	W#16#5E

Dependiendo del estado lógico de la entrada "Tag_Input_G", se selecciona el valor de la entrada "Tag_Input0" o "Tag_Input1" y se copia en la salida "Tag_Output".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Desplazamiento y rotación

SHR: Desplazar a la derecha

Descripción

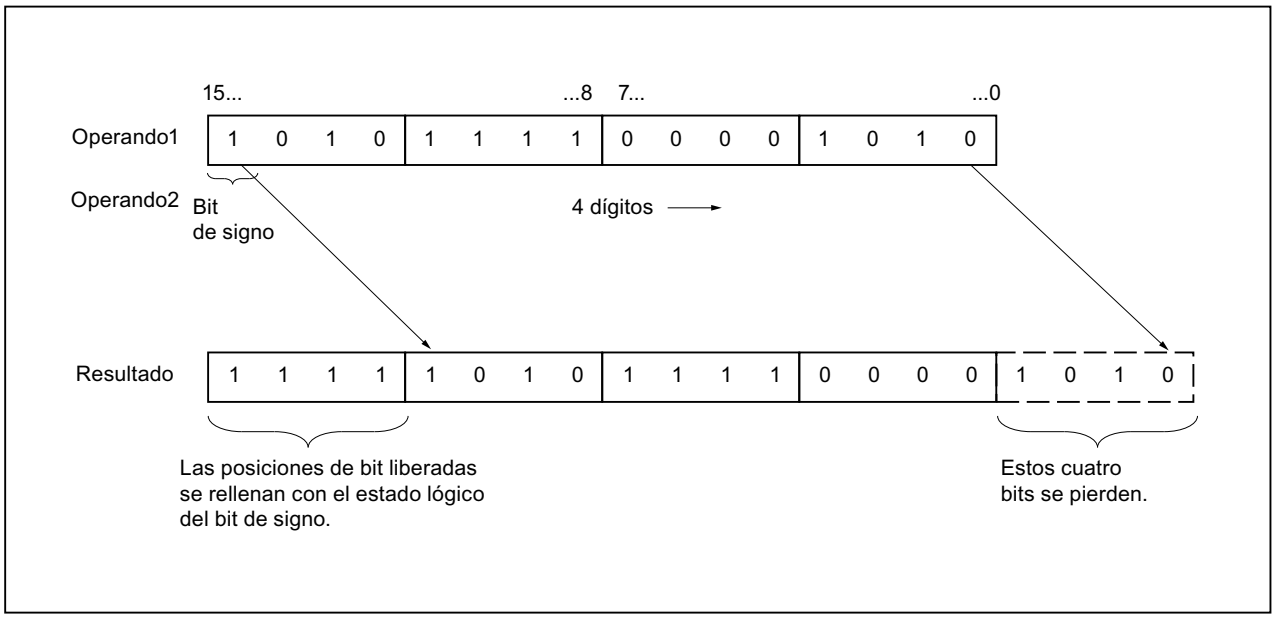
Con la instrucción "Desplazar a la derecha", el contenido del operando 1 se mueve bit a bit hacia la derecha. El operando 2 determina el número de posiciones de bit que debe desplazarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se desplaza hacia la derecha las posiciones de bit disponibles.

En los valores sin signo, se rellenan con ceros las posiciones de bits que quedan libres en el área izquierda del operando 1 al realizar el desplazamiento. Si el valor indicado lleva signo, las posiciones libres se ocupan con el estado lógico del bit de signo.

La figura siguiente muestra cómo el contenido del operando 1 del tipo de datos INT se desplaza cuatro posiciones de bit hacia la derecha:



Sintaxis

Para la instrucción "Desplazar a la derecha" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SHR_<Tipo de datos>(<Operando 1>,<Operando 2>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la derecha":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros	I, Q, M, D, L, P o constante	Valor que se desplaza
<Operando2>	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que se desplaza el valor
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SHR_INT("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	0011 1111 1010 1111
Tag_Number	3
Tag_OutValue	0000 0111 1111 0101

El contenido del operando "Tag_InValue" se desplaza tres posiciones de bit hacia la derecha y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SHL: Desplazar a la izquierda

Descripción

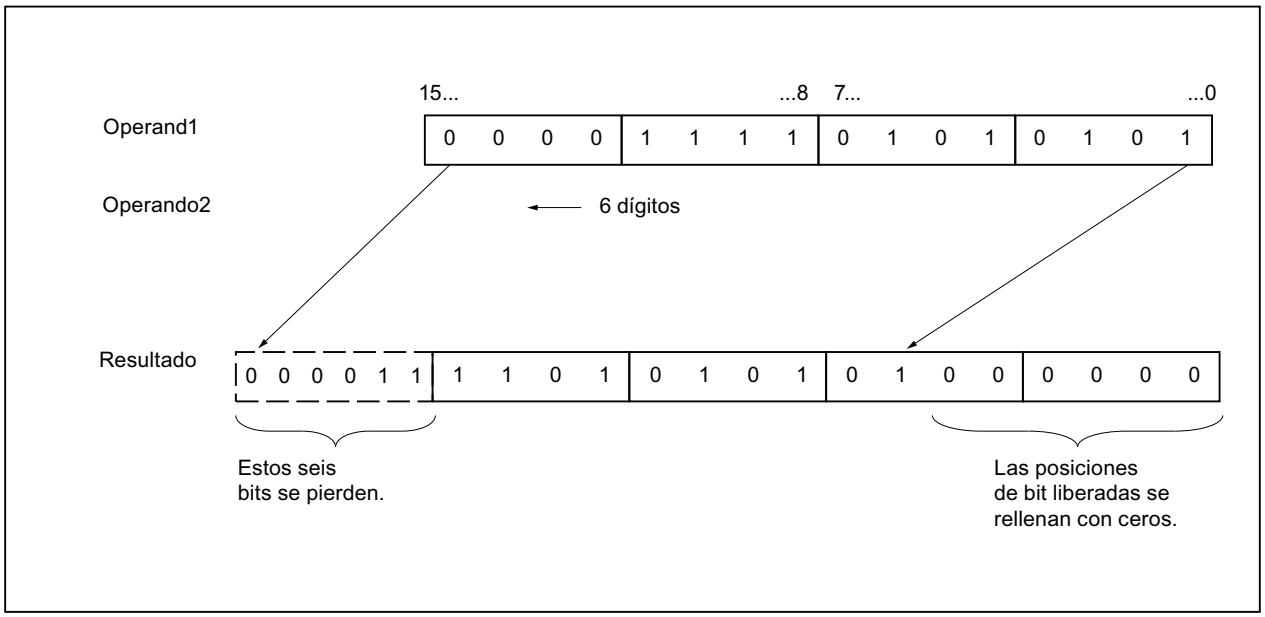
Con la instrucción "Desplazar a la izquierda", el contenido del operando 1 se mueve bit a bit hacia la izquierda. El operando 2 determina el número de posiciones de bit que debe desplazarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se desplaza hacia la izquierda las posiciones de bit disponibles.

Las posiciones de bit que quedan libres en el área derecha del operando 1 al realizar el desplazamiento se rellenan con ceros.

La figura siguiente muestra cómo el contenido del operando 1 del tipo de datos WORD se desplaza seis posiciones de bit hacia la izquierda:



Sintaxis

Para la instrucción "Desplazar a la izquierda" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := SHL_<Tipo de datos>(<Operando 1>,<Operando 2>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar a la izquierda":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Secuencias de bits, enteros	I, Q, M, D, L, P o constante	Valor que se desplaza
<Operando2>	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que se desplaza el valor
<Resultado>	Output	Secuencias de bits, enteros	I, Q, M, D, L	Resultado de la instrucción

En la lista desplegable "<Tipo de datos>" puede seleccionar el tipo de datos de la instrucción.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := SHL_WORD("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	0011 1111 1010 1111
Tag_Number	4
Tag_OutValue	1111 1010 1111 0000

El contenido del operando "Tag_InValue" se desplaza cuatro posiciones de bit hacia la izquierda y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ROR: Rotar a la derecha

Descripción

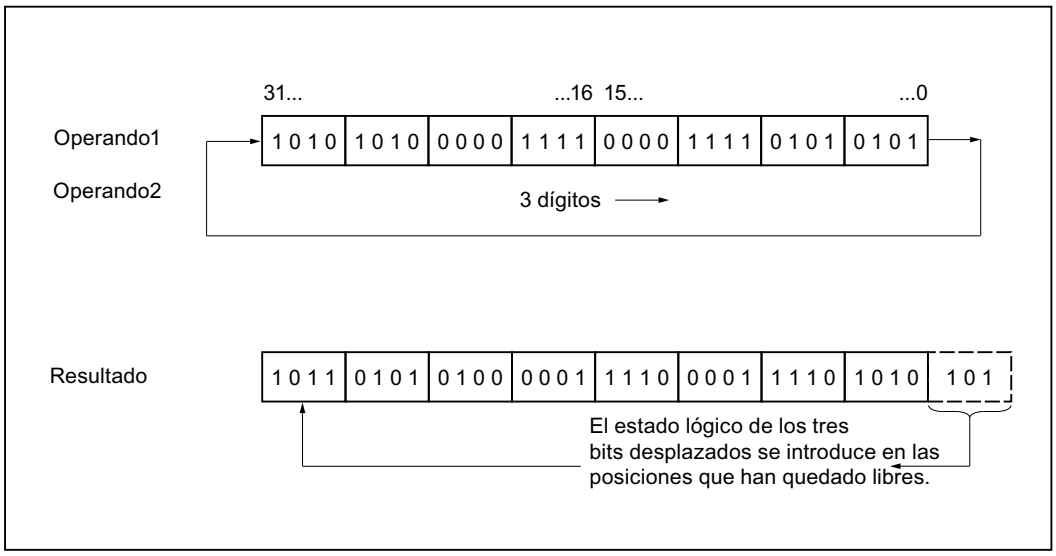
Con la instrucción "Rotar a la derecha", el contenido del operando 1 se rota bit a bit hacia la derecha. El operando 2 determina el número de posiciones de bit que debe rotarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se rota el número indicado de posiciones de bit.

Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la derecha:



Sintaxis

Para la instrucción "Rotar a la derecha" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ROR(<Operando 1>,<Operando 2>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la derecha":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	BYTE, WORD, DWORD, DINT	I, Q, M, D, L, P o constante	Valor que se rota
<Operando2>	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que rota el valor
<Resultado>	Output	BYTE, WORD, DWORD, DINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ROR("Tag_InValue", "Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1010 1010 0000 1111 0000 1111 0101 0101
Tag_Number	5
Tag_OutValue	1010 1101 0101 0000 0111 1000 0111 1010

El contenido del operando "Tag_InValue" se rota cinco posiciones de bit hacia la derecha y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

ROL: Rotar a la izquierda

Descripción

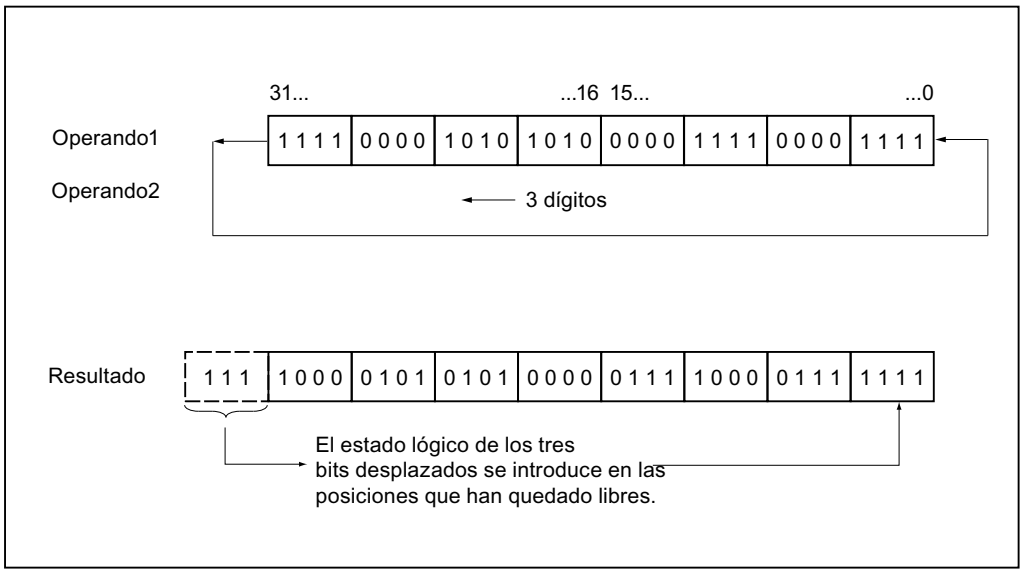
Con la instrucción "Rotar a la izquierda", el contenido del operando 1 se rota bit a bit hacia la izquierda. El operando 2 determina el número de posiciones de bit que debe rotarse el valor indicado.

Cuando el valor del operando 2 es "0", se copia en valor del operando 1 en el resultado.

Si el valor del operando 2 es mayor que el número de posiciones de bit disponibles, el valor del operando 1 se rota el número indicado de posiciones de bit.

Los bits que quedan libres al realizar la rotación se rellenan con los bits desplazados hacia fuera.

La figura siguiente muestra cómo el contenido de un operando del tipo de datos DWORD rota tres posiciones hacia la izquierda:



Sintaxis

Para la instrucción "Rotar a la izquierda" se utiliza la siguiente sintaxis:

GRAPH

```
<Resultado> := ROL(<Operando 1>,<Operando 2>)
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Rotar a la izquierda":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	BYTE, WORD, DWORD, DINT	I, Q, M, D, L, P o constante	Valor que se rota
<Operando2>	Input	BYTE, WORD, INT	I, Q, M, D, L, P o constante	Número de posiciones de bit que rota el valor
<Resultado>	Output	BYTE, WORD, DWORD, DINT	I, Q, M, D, L	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
"Tag_OutValue" := ROL("Tag_InValue","Tag_Number")
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Operando	Valor
Tag_InValue	1111 0000 1010 1010 0000 1111 0000 1111
Tag_Number	5
Tag_OutValue	0001 0101 0100 0001 1110 0001 1111 1110

El contenido del operando "Tag_InValue" se rota cinco posiciones de bit hacia la izquierda y se indica como resultado en el operando "Tag_OutValue".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Otras instrucciones

SET: Activar área por bits

Descripción

La instrucción "Activar área por bits" establece el estado lógico de los bits en "1" en un área indicada. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa (P), no se ejecuta la instrucción. El número de bits que deben activarse en el área indicada se determina en el parámetro N de la instrucción. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta sólo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

Sintaxis

Para la instrucción "Activar área por bits" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SET
     (S_BIT := <Operando>
      N := <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del área cuyos bits se activan En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben activarse

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SET
      (S_BIT := P#M0.0
       N := 10
       )
```

La instrucción activa 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SETP: Activar mapa de bits en el área de periferia

Descripción

La instrucción "Activar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "1". Paralelamente a los bits de periferia, también se activan los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA sirven para definir el mapa de bits que debe activarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no se ha conectado periferia para una parte del mapa de bits seleccionado, la instrucción intentará activar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se activa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor del bit correspondiente en la memoria imagen de proceso de las salidas.

  ⚠ ADVERTENCIA

  Consecuencias no deseadas

  El ajuste de los bits al valor del bit correspondiente en la memoria imagen de proceso de las salidas puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se ponen al estado lógico "1".

La instrucción "Activar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Activar mapa de bits en el área de periferia" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El parámetro N devuelve el valor "0"
- El bit MCR tiene el estado lógico "1".

Sintaxis

Para la instrucción "Activar mapa de bits en el área de periferia" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SETP
      (N := <Operando>
       RET_VAL => <Operando>
       SA => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben activarse.
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero hacia el primer bit que debe activarse

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SETP
     (N := 8
      RET_VAL => "Tag_Error"
      SA => P#10.0
      )
```

La instrucción activa 8 bits en el área de periferia conforme a las salidas Q10.0 a Q10.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SETI: Activar área por bytes

Descripción

La instrucción "Activar área por bytes" establece en "1" el estado lógico de los bits en un área indicada de bytes. El punto inicial del área se define en el parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben activarse en el parámetro N.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa (P). Puesto que al área de memoria "P" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte; por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas se pone a "0".

La llamada de la instrucción "Activar área por bytes" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa.
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.
- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

Sintaxis

Para la instrucción "Activar área por bytes" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SETI
     (S_BYTE := <Operando>
      N := <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Activar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se activan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área que debe activarse El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SETI
     (S_BYTE := P#2.0
      N := 16
      )
```

La instrucción activa 2 bytes (16 bits) en el área de P2.0 a P3.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

RESET: Desactivar área por bits

Descripción

La instrucción "Desactivar área por bits" establece el estado lógico de los bits en "0" en un área indicada. El punto inicial del área se define mediante el puntero del parámetro S_BIT. Si el puntero señala al área de memoria de la periferia externa, no se ejecuta la instrucción. El número de bits que deben desactivarse en el área indicada se determina en el parámetro N. Si el parámetro N tiene el valor "0", la llamada de la instrucción no tendrá efecto alguno.

La instrucción se ejecuta sólo si el bit MCR está a "1". Si el bit MCR tiene el estado lógico "0", no se modifican los bits en el área indicada.

Sintaxis

Para la instrucción "Desactivar área por bits" se utiliza la siguiente sintaxis:

GRAPH

```
CALL RESET
     (S_BIT := <Operando>
      N := <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bits":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BIT	Input	POINTER	I, Q, M, D	Señala al primer bit del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Número de bits que deben desactivarse

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL RESET
     (S_BIT := P#M0.0
      N := 10
      )
```

La instrucción desactiva 10 bits en el área de M0.0 a M1.2.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

RESETP: Desactivar mapa de bits en el área de periferia

Descripción

La instrucción "Desactivar mapa de bits en el área de periferia" permite poner los bits de un área de periferia al estado lógico "0". Simultáneamente con los bits de periferia se desactivan también los bits correspondientes en la memoria imagen de proceso de las salidas.

Los parámetros N y SA de la instrucción sirven para definir el mapa de bits que debe desactivarse. Los bits del mapa deben encontrarse en aquella parte del área de periferia que tiene asignada una memoria imagen de proceso.

Si no hay periferia enchufada para una parte del mapa de bits seleccionado, la instrucción intentará desactivar el mapa de bits completo. En este caso se devuelve una información de error en el parámetro RET_VAL.

El área de periferia se inicializa por bytes. Si el mapa de bits definido a través de los parámetros N y SA no empieza ni termina en un límite de byte, la llamada de la instrucción tendrá el efecto siguiente:

- Los bits contenidos en el primer o último byte que debe transferirse y que no pertenecen al mapa de bits definido, obtienen el valor del bit correspondiente en la memoria imagen de proceso de las salidas.

  ⚠ ADVERTENCIA

  Consecuencias no deseadas

  El ajuste de los bits al valor del bit correspondiente en la memoria imagen de proceso de las salidas puede ocasionar p. ej. el arranque accidental de motores o la desconexión de sistemas de refrigeración.

- Los bits que pertenecen al mapa de bits definido se restablecen al estado lógico "0".

La instrucción "Desactivar mapa de bits en el área de periferia" debe llamarse en un área MCR.

La llamada de la instrucción "Desactivar mapa de bits en el área de periferia" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El parámetro N devuelve el valor "0"
- El bit MCR tiene el estado lógico "1".

Sintaxis

Para la instrucción "Desactivar mapa de bits en el área de periferia" se utiliza la siguiente sintaxis:

GRAPH

```
CALL RESETP
     (N := <Operando>
      RET_VAL => <Operando>
      SA => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar mapa de bits en el área de periferia":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
N	Input	INT	I, Q, M, D, L o constante	Número de bits que deben desactivarse
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
SA	Output	POINTER	P	Puntero al primer bit que debe desactivarse

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#...)	Explicación
0000	Ningún error
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL RESETP
     (N := 8
      RET_VAL => "Tag_Error"
      SA => P#10.0
      )
```

La instrucción desactiva 8 bits en el área de periferia conforme a las salidas Q10.0 a Q10.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

RESETI: Desactivar área por bytes

Descripción

La instrucción "Desactivar área por bytes" restablece el estado lógico de los bits a "0" en un área de bytes indicada. El punto inicial del área se define mediante el puntero del parámetro S_BYTE. El tamaño del área se define indicando el número de bits que deben desactivarse en el parámetro N de la instrucción.

Nota

El valor del parámetro N debe ser un múltiplo de 8 (p. ej. 8, 16, 24, etc.).

El puntero indicado en el parámetro S_BYTE debe señalar al área de memoria de la periferia externa. Puesto que al área de memoria "P" se accede en formato de byte, palabra y palabra doble, el puntero del parámetro S_BYTE debe señalar a un operando equivalente a un valor límite de byte; por tanto, el número de bit del puntero debe ser "0".

Nota

El estado lógico de los bits correspondientes en la memoria imagen de proceso de las salidas se pone a "0".

La llamada de la instrucción "Desactivar área por bytes" no tiene efecto alguno si se cumple una de las condiciones siguientes:

- El puntero del parámetro S_BYTE no señala al área de memoria de la periferia externa.
- El puntero del parámetro S_BYTE señala a un operando que no equivale a ningún valor límite de byte.

- El valor del parámetro N no es un múltiplo de 8.
- El bit MCR tiene el estado lógico "0".

Sintaxis

Para la instrucción "Desactivar área por bytes" se utiliza la siguiente sintaxis:

GRAPH

```
CALL RESETI
      (S_BYTE := <Operando>
       N := <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desactivar área por bytes":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
S_BYTE	Input	POINTER	P o constante	Señala al primer byte del área cuyos bits se desactivan. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	INT	I, Q, M, D, L, P o constante	Tamaño del área cuyos bits se desactivan. El área se indica mediante el número de bits como múltiplo de 8 (p. ej. 8, 16, 24, etc.).

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL RESETI
```

GRAPH

```
(S_BYTE := P#2.0
 N := 16
 )
```

La instrucción desactiva 2 bytes (16 bits) en el área de P2.0 a P3.7.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

REPL_VAL: Introducir valor sustitutivo

Descripción

La instrucción "Introducir valor sustitutivo" permite introducir un valor sustitutivo en el acumulador 1 (ACU 1) desde un bloque de organización de error síncrono.

Nota

La instrucción "Introducir valor sustitutivo" sólo se puede llamar desde bloques de organización de error síncrono (OB 121, OB 122).

La instrucción "Introducir valor sustitutivo" se utiliza si ya no es posible leer ningún valor de un módulo. Tras cada acceso a dicho módulo se inicia el OB 122. Llamando la instrucción "Introducir valor sustitutivo" es posible cargar un valor sustitutivo en el acumulador 1. La ejecución del programa se prosigue entonces con este valor sustitutivo. La información necesaria para seleccionar el valor sustitutivo puede consultarse en las variables locales del OB 122.

Sintaxis

Para la instrucción "Introducir valor sustitutivo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL REPL_VAL
     (VAL := <Operando>
      RET_VAL => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Introducir valor sustitutivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
VAL	Input	DWORD	I, Q, M, D, L o constante	Valor sustitutivo
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro RET_VAL

La tabla siguiente muestra el significado de los valores del parámetro RET_VAL:

Código de error* (W#16#....)	Descripción
0000	Ningún error
8080	La instrucción no se ha llamado desde un OB de error síncrono.
8xyy	Encontrará más información de error en "Consulte también".
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

DRUM: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite ocupar los bits de salida programados (OUT1 a OUT16) y la palabra de salida (OUT_WORD) con los valores programados del parámetro OUT_VAL del paso respectivo. El paso en cuestión debe cumplir las condiciones de la máscara de habilitación programada en el parámetro S_MASK mientras la instrucción permanece en dicho paso. La instrucción pasa al paso siguiente cuando el evento de éste es verdadero y una vez transcurrido el tiempo programado para el paso actual, o bien cuando el valor del parámetro JOG cambia de "0" a "1". La instrucción se desactiva cuando el estado lógico del parámetro RESET cambia a "1". Con ello, el paso actual se iguala al paso predeterminado (DSP).

El tiempo de permanencia en un paso se calcula multiplicando la base de tiempo predeterminada (DTBP) por el valor de contaje predeterminado (S_PRESET) para cada paso. Al comienzo de un nuevo paso, este valor calculado se carga en el parámetroDCC, que contiene el tiempo restante para el paso actual. Por ejemplo, si el valor del parámetro DTBP

es "2" y el valor predeterminado para el primer paso es "100" (100 ms), el parámetro DCC devolverá el valor "200" (200 ms).

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1". Los bits de evento se inicializan con el estado lógico "1".

Si el circuito secuencial sucesivo se encuentra en el último paso programado (LST_STEP) y ha transcurrido el tiempo para dicho paso, el estado lógico del parámetro Q se pone a "1". De lo contrario, se pone a "0". Si está activado el parámetro Q, la instrucción permanece en el paso hasta la desactivación.

En la máscara configurable (S_MASK) es posible seleccionar los distintos bits de la palabra de salida (OUT_WORD) y activar o desactivar los bits de salida (OUT1 a OUT16) mediante los valores de salida (OUT_VAL). Si un bit de la máscara configurable tiene el estado lógico "1", el valor OUT_VAL activa o desactiva el bit respectivo. Si el estado lógico de un bit de la máscara configurable es "0", no cambia el bit respectivo. Todos los bits de la máscara configurable de todos los 16 pasos se inicializan con el estado lógico "1".

El bit de salida del parámetro OUT1 equivale al bit menos significativo de la palabra de salida (OUT_WORD). El bit de salida del parámetro OUT16 equivale al bit más significativo de la palabra de salida (OUT_WORD).

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DRUM, <DB instancia>
     (RESET := <Operando>
      JOG := <Operando>
      DRUM_EN := <Operando>
      LST_STEP := <Operando>
      EVENT1 - 16 := <Operando>
      OUT1 - 16 => <Operando>
      Q => <Operando>
      OUT_WORD => <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" indica una condición de desactivación.
JOG	Input	BOOL	I, Q, M, D, L	Si el estado lógico cambia de "0" a "1", la instrucción pasa al siguiente paso.
DRUM_EN	Input	BOOL	I, Q, M, D, L	El estado lógico "1" permite que el circuito secuencial sucesivo siga contando según el evento y los criterios de tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso programado.
EVENT(i), 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento (i); El estado lógico inicial es "1".
OUT(j), 1 ≤ j ≤ 16	Output	BOOL	I, Q, M, D, L	Bit de salida (j)
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Dirección de palabra en la que el circuito secuencial sucesivo escribe los valores de salida.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución del parámetro JOG
EOD	Static	BOOL	I, Q, M, D, L	El estado lógico "1" indica que ha transcurrido el tiempo del último paso.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DSP	Static	BYTE	I, Q, M, D, L, P	Paso predeterminado del circuito secuencial sucesivo
DSC	Static	BYTE	I, Q, M, D, L, P	Paso actual del circuito secuencial sucesivo
DCC	Static	DWORD	I, Q, M, D, L, P	Paso actual del circuito secuencial sucesivo
DTBP	Static	WORD	I, Q, M, D, L, P	Base de tiempo predeterminada del circuito secuencial sucesivo
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	Hora de sistema anterior
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L	Valor de contaje predeterminado para cada paso [1 a 16]; 1 ciclo de reloj = 1 ms.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Valores de salida para cada paso [1 a 16, 0 a 15].
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Máscara configurable para cada paso [1 a 16, 0 a 15]. Los estados lógicos iniciales son "1".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE*	Explicación
W#16#0000	Ningún error
W#16#000B	El valor del parámetro LST_STEP es menor que 1 o mayor que 16.
W#16#000C	El valor del parámetro DSC es menor que 1 o mayor que el valor del parámetro LST_STEP.
W#16#000D	El valor del parámetro DSP es menor que 1 o mayor que el valor de LST_STEP.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

DRUM_X: Materializar circuito secuencial sucesivo

Descripción

La instrucción "Materializar circuito secuencial sucesivo" permite materializar un circuito secuencial sucesivo con 16 pasos como máximo. El número del primer paso se define con el parámetro DSP. El número del último paso se predetermina con el parámetroLST_STEP.

En cada paso se escriben todos los 16 bits de salida OUT0 a OUT15 y el parámetro de salida OUT_WORD (que contiene los bits de salida). A un bit de salida se asigna bien sea el bit correspondiente del campo predefinido OUT_VAL, o bien el bit de salida del paso anterior. El valor que se asigne dependerá de la ocupación de los bits de máscara en el parámetro S_MASK.

Nota

El ajuste predeterminado de los bits de máscara es 0. Para modificar el ajuste actual de uno o varios bits de máscara, realice el cambio en el bloque de datos de instancia.

La instrucción "Materializar circuito secuencial sucesivo" pasa al siguiente paso cuando se produce un flanco ascendente en el parámetro JOG. Si la instrucción ya se encuentra en el último paso y se produce un flanco ascendente en el parámetro JOG, se activan las variablesQ y EOD. En este caso, el parámetro DCC se pone al valor 0. La instrucción permanece en el último paso hasta que el parámetro RESET se pone al valor "1".

Además, también se puede permitir la transición al paso siguiente en función del tiempo. Para ello se debe poner al valor "1" el parámetro DRUM_EN. La transición al paso siguiente tiene lugar si está activado el bit de evento (EVENTi) correspondiente al paso actual y si ha transcurrido el tiempo predeterminado para el paso actual. El tiempo necesario para un paso resulta de la multiplicación de la base de tiempo (DTBP por el factor de tiempo válido para el paso actual (S_PRESET).

Nota

El tiempo de ejecución restante en el paso actual (DCC) se reduce únicamente si está activado el bit de evento correspondiente (EVENTi).

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Si, al llamar la instrucción, el parámetro RESET tiene el valor "1", el circuito secuencial sucesivo pasará al paso cuyo número se ha asignado al parámetroDSP.

Un paso puede programarse con un valor de tiempo, con un evento o con ambos. Los pasos programados con un bit de evento y el valor de tiempo "0" pasan al siguiente paso en cuanto el estado lógico del bit de evento es "1". Los pasos programados sólo con un valor de tiempo empiezan a contar el tiempo inmediatamente. Los pasos programados con un bit de evento y un valor de tiempo mayor que "0" empiezan a contar el tiempo en cuanto el estado lógico del bit de evento es "1".

El circuito secuencial sucesivo también se puede conmutar en progresión en todo momento a través del valor del parámetro JOG.

En la primera llamada de la instrucción, el parámetro RESET se debe poner a 1.

Si el circuito secuencial se encuentra en el último paso y ha transcurrido el tiempo de ejecución predefinido para este paso, se activan los parámetros Q y EOD. La instrucción permanece en el último paso hasta que el parámetro RESET adopte el valor "1".

Esta instrucción se ejecuta sólo en los estados operativos ARRANQUE y RUN.

El sistema operativo inicializa la instrucción "Materializar circuito secuencial sucesivo" cuando se efectúa un arranque en frío. La instrucción no se inicializa en caso de un rearranque completo (en caliente). Si la instrucción "Materializar circuito secuencial sucesivo" debe inicializarse tras un rearranque completo (en caliente), es preciso llamarla en el OB 100 con el valor "1" en el parámetro RESET.

Sintaxis

Para la instrucción "Materializar circuito secuencial sucesivo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DRUM_X, <DB instancia>
     (RESET := <Operando>
      JOG := <Operando>
      DRUM_EN := <Operando>
      LST_STEP := <Operando>
      EVENT1 - 16 := <Operando>
      OUT0 - 15 => <Operando>
      Q => <Operando>
      OUT_WORD => <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Materializar circuito secuencial sucesivo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	El estado lógico "1" desactiva el circuito secuencial. En la primera llamada de la instrucción, el parámetro RESET se debe poner a "1".
JOG	Input	BOOL	I, Q, M, D, L	Un flanco de señal ascendente (con respecto a la última llamada) provoca la transición al paso siguiente, siempre que el circuito secuencial no se encuentre aún en el último paso. La transición tiene lugar independientemente del valor asignado al parámetro DRUM_EN.
DRUM_EN	Input	BOOL	I, Q, M, D, L	Parámetro de control que determina si se puede realizar una transición al paso siguiente en función del tiempo. Si el valor es "1", es posible una transición en función del tiempo.
LST_STEP	Input	BYTE	I, Q, M, D, L o constante	Número del último paso Los valores admisibles son 1 a 16.
EVENTi, 1 ≤ i ≤ 16	Input	BOOL	I, Q, M, D, L	Bit de evento n.º i (pertenece al paso i)
OUTj, 0 ≤ j ≤ 15	Output	BOOL	I, Q, M, D, L	Bit de salida n.º j (idéntico al bit n.º j de OUT_WORD)
Q	Output	BOOL	I, Q, M, D, L	Parámetro de estado que indica si ha transcurrido el tiempo predefinido por el usuario para la ejecución del último paso.
OUT_WORD	Output	WORD	I, Q, M, D, L, P	Bits de salida resumidos en una variable
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
JOG_HIS	Static	BOOL	I, Q, M, D, L	Irrelevante para el usuario (parámetro de entrada JOG de la llamada anterior)
EOD	Static	BOOL	I, Q, M, D, L	Idéntico al parámetro de salida Q

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DSP	Static	BYTE	I, Q, M, D, L, P o constante	Número del primer paso Los valores admisibles son 1 a 16.
DSC	Static	BYTE	I, Q, M, D, L, P o constante	Número del paso actual
DCC	Static	DWORD	I, Q, M, D, L, P	Tiempo de ejecución restante del paso actual, indicado en milisegundos (ms) (relevante sólo si DRUM_EN = 1 y el bit de evento correspondiente = 1)
DTBP	Static	WORD	I, Q, M, D, L, P o constante	Base de tiempo en milisegundos (ms) válida para todos los pasos
PREV_TIME	Static	DWORD	I, Q, M, D, L o constante	(Irrelevante para el usuario: hora de sistema de la llamada anterior)
S_PRESET	Static	ARRAY of WORD	I, Q, M, D, L o constante	Campo unidimensional con los factores de tiempo para cada paso. Selección recomendada de índices: [1 a 16]. En este caso, S_PRESET [x] contiene el factor de tiempo del paso x.
OUT_VAL	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los valores que deben emitirse en cada paso, a no ser que se hayan inhibido vía S_MASK. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, OUT_VAL [x, y] contiene el valor asignado al bit de salida OUTy en el paso x.
S_MASK	Static	ARRAY of BOOL	I, Q, M, D, L	Campo bidimensional con los bits de máscara para cada paso. Selección recomendada de índices: [1 a 16, 0 a 15]. En este caso, S_MASK [x, y] contiene el bit de máscara para el valor "y"-ésimo que debe emitirse en el paso x. Significado de los bits de máscara: • 0: el valor correspondiente del paso anterior se asigna al respectivo bit de salida. • 1: el valor correspondiente de OUT_VAL se asigna al respectivo bit de salida.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

ERR_CODE* (W#16#...)	Explicación
0000	Ningún error
8081	Valor no admisible en el parámetro LST_STEP
8082	Valor no admisible en DSC
8083	Valor no admisible en DSP
8084	El producto DCC = DTBP * S_PRESET[DSC] excede el valor 2**31-1 (aprox. 24,86 días).
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

DCAT: Alarma de tiempo para control discreto

Descripción

La instrucción "Alarma de tiempo para control discreto" acumula el tiempo desde el instante en el que el parámetroCMD da la orden de abrir o cerrar. El tiempo se acumula hasta excederse el tiempo predeterminado (PT) o hasta recibir la notificación de que el dispositivo se ha cerrado o abierto dentro del tiempo prescrito (O_FB o C_FB). Si se excede el tiempo predeterminado antes de recibirse la notificación de que se ha abierto o cerrado el dispositivo, se activa la alarma correspondiente. Si el estado lógico de la entrada de la orden cambia antes del tiempo predeterminado, se reinicia el tiempo.

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

La instrucción "Alarma de tiempo para control discreto" tiene las reacciones siguientes en las condiciones de entrada:

- Si el estado lógico del parámetro CMD cambia de "0" a "1", esto tiene los efectos siguientes en los estados lógicos de los parámetros Q, CMD_HIS, ET, OA y CA:
  - Los parámetros Q y CMD_HIS se ponen a "1".
  - Los parámetros ET, OA y CA se ponen a "0".
- Si el estado lógico del parámetro CMD cambia de "1" a "0", los parámetros Q, ET, OA, CA y CMD_HIS se ponen a "0".
- Si los parámetros CMD y CMD_HIS tienen el estado lógico "1" y el parámetro O_FB se pone a "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro OA se pone a "1". Si el valor del parámetro ET no excede el del parámetro PT, el estado lógico del parámetro OA se pone a "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los estados lógicos de los parámetros CMD, CMD_HIS y O_FB están a "1" y el parámetro C_FB devuelve el valor "0", el estado lógico del parámetro OA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro O_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y C_FB devuelven "0", la diferencia de tiempo (ms) desde la última ejecución de la instrucción se suma al valor del parámetro ET. Si el valor del parámetro ET excede el del parámetro PT, el estado lógico del parámetro CA se pone a "1". Si no se excede el valor del parámetro PT, el parámetro CA devuelve el estado lógico "0". El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros CMD, CMD_HIS y O_FB devuelven el estado lógico "0" y el parámetro C_FB está a "1", el parámetro CA se pone a "0". El valor del parámetro ET se pone al del parámetro PT. Si el estado lógico del parámetro C_FB cambia a "0", se activa la alarma en la siguiente ejecución de la instrucción. El valor del parámetro CMD_HIS se pone al del parámetro CMD.
- Si los parámetros O_FB y C_FB devuelven simultáneamente el estado lógico "1", los estados lógicos de ambas salidas de alarma se ponen a "1".

La instrucción "Alarma de tiempo para control discreto" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control discreto" se utiliza la siguiente sintaxis:

GRAPH

```
CALL DCAT, <DB instancia>
     (CMD := <Operando>
      O_FB := <Operando>
      C_FB := <Operando>
      Q => <Operando>
      OA => <Operando>
      CA => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control discreto":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BOOL	I, Q, M, D, L	El estado lógico "0" da la orden de "Cerrar". El estado lógico "1" da la orden de "Abrir".
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
Q	Output	BOOL	I, Q, M, D, L	Indica el estado del parámetro CMD
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
CMD_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución de CMD

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente, el parámetro CMD cambia de "0" a "1". Una vez ejecutada la instrucción, el parámetro Q se pone a "1" y las dos salidas de alarma OA y CA se ponen a "0". El parámetro CMD_HIS del bloque de datos de instancia se pone a "1" y el parámetro ET se pone a "0".

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL DCAT, "DCAT_DB"
      (CMD := "Tag_Input_CMD"
       O_FB := "Tag_Input_O_FB"
       C_FB := "Tag_Input_C_FB"
       Q => "Tag_Output_Q"
       OA => "Tag_Output_OA"
       CA => "Tag_Output_CA"
```

GRAPH
)

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
CMD	Tag_Input_CMD	TRUE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
Q	Tag_Output_Q	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

Los siguientes valores están almacenados en el bloque de datos de instancia "DCAT_DB" de la instrucción:

Parámetro	Dirección	Valor
ET	DBD4	L#12
PT	DBD8	L#222
CMD_HIS	DBX16.0	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
Q	Tag_Output_Q	TRUE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE

En el bloque de datos de instancia "DCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
CMD_HIS	DBX16.0	TRUE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

MCAT: Alarma de tiempo para control del motor

Descripción

La instrucción "Alarma de tiempo para control del motor" acumula el tiempo desde el instante en el que se activa una de las entradas de comando (abrir o cerrar). El tiempo se acumula hasta que se excede el tiempo predeterminado o hasta que la entrada de realimentación correspondiente indica que el dispositivo ha ejecutado la operación solicitada dentro del tiempo prescrito. Si se excede el tiempo predeterminado antes de recibir la realimentación, se activa la alarma correspondiente.

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Ejecución de la instrucción "Alarma de tiempo para control del motor"

La tabla siguiente muestra las reacciones de la instrucción "Alarma de tiempo para control del motor" a las distintas condiciones de entrada:

Parámetros de entrada								Parámetros de salida								
ET	O_HIS	C_HIS	O_CMD	C_CMD	S_CMD	O_FB	C_FB	OO	CO	OA	CA	ET	O_HIS	C_HIS	Q	Estado
X	1	1	X	X	X	X	X	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	X	1	1	0	0	1	1	PT	0	0	0	Alarma
X	X	X	X	X	1	X	X	0	0	0	0	X	0	0	1	Stop
X	X	X	1	1	X	X	X	0	0	0	0	X	0	0	1	Stop
X	0	X	1	0	0	X	X	1	0	0	0	0	1	0	1	Empezar a abrir
<PT	1	0	X	0	0	0	X	1	0	0	0	INC	1	0	1	Abrir
X	1	0	X	0	0	1	0	0	0	0	0	PT	1	0	1	Abierto
>= PT	1	0	X	0	0	0	X	0	0	1	0	PT	1	0	0	Abrir alarma
X	X	0	0	1	0	X	X	0	1	0	0	0	0	1	1	Empezar a cerrar
< PT	0	1	0	X	0	X	0	0	1	0	0	INC	0	1	1	Cerrar
X	0	1	0	X	0	0	1	0	0	0	0	PT	0	1	1	Cerrado
>= PT	0	1	0	X	0	X	0	0	0	0	1	PT	0	1	0	Cerrar alarma
X	0	0	0	0	0	X	X	0	0	0	0	X	0	0	1	Parado

Leyenda:	
INC	Sumar a ET la diferencia de tiempo (en ms) desde la última ejecución del FB
PT	PT se pone al mismo valor que ET
X	No aplicable
< PT	ET < PT

Parámetros de entrada		Parámetros de salida
>= PT	ET >= PT	
Si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", se ponen inmediatamente al estado lógico "0". En este caso se aplica la última fila de la tabla mencionada más arriba (X). Puesto que por este motivo no es posible comprobar si los parámetros de entrada O_HIS y C_HIS tienen el estado lógico "1", en este caso se asigna el siguiente valor a los parámetros de salida: OO = FALSE CO = FALSE OA = FALSE CA = FALSE ET = PT Q = TRUE		

La instrucción "Alarma de tiempo para control del motor" no devuelve información de error.

Sintaxis

Para la instrucción "Alarma de tiempo para control del motor" se utiliza la siguiente sintaxis:

GRAPH

```
CALL MCAT, <DB instancia>
     (O_CMD := <Operando>
      C_CMD := <Operando>
      S_CMD := <Operando>
      O_FB := <Operando>
      C_FB := <Operando>
      OO => <Operando>
      CO => <Operando>
      OA => <Operando>
      CA => <Operando>
      Q => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Alarma de tiempo para control del motor":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
O_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Abrir"
C_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Cerrar"
S_CMD	Input	BOOL	I, Q, M, D, L	Entrada de la orden "Parar"
O_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al abrir

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
C_FB	Input	BOOL	I, Q, M, D, L	Entrada de realimentación al cerrar
OO	Output	BOOL	I, Q, M, D, L	Salida "Abrir"
CO	Output	BOOL	I, Q, M, D, L	Salida "Cerrar"
OA	Output	BOOL	I, Q, M, D, L	Salida de alarma al abrir
CA	Output	BOOL	I, Q, M, D, L	Salida de alarma al cerrar
Q	Output	BOOL	I, Q, M, D, L	El estado lógico "0" indica una condición de error.
ET	Static	DINT	I, Q, M, D, L	Tiempo actual transcurrido; un ciclo de reloj = 1 ms
PT	Static	DINT	I, Q, M, D, L	Valor de tiempo predeterminado; un ciclo de reloj = 1 ms
PREV_TIME	Static	DWORD	I, Q, M, D, L	Hora de sistema anterior
O_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Abrir"
C_HIS	Static	BOOL	I, Q, M, D, L	Bit de evolución "Cerrar"

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL MCAT, "MCAT_DB"
     (O_CMD := "Tag_Input_O_CMD"
      C_CMD := "Tag_Input_C_CMD"
      S_CMD := "Tag_Input_S_CMD"
      O_FB := "Tag_Input_O_FB"
      C_FB := "Tag_Input_C_FB"
      OO => "Tag_OutputOpen"
      CO => "Tag_OutputClosed"
      OA => "Tag_Output_OA"
      CA => "Tag_Output_CA"
      Q => "Tag_Output_Q"
```

GRAPH
)

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
O_CMD	Tag_Input_O_CMD	TRUE
C_CMD	Tag_Input_C_CMD	FALSE
S_CMD	Tag_Input_S_CMD	FALSE
O_FB	Tag_Input_O_FB	FALSE
C_FB	Tag_Input_C_FB	FALSE
OO	Tag_OutputOpen	FALSE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	FALSE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#2
PT	DBD8	L#22
O_HIS	DBX16.0	TRUE
C_HIS	DBX16.1	FALSE

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OO	Tag_OutputOpen	TRUE
CO	Tag_OutputClosed	FALSE
OA	Tag_Output_OA	FALSE
CA	Tag_Output_CA	FALSE
Q	Tag_Output_Q	TRUE

En el bloque de datos de instancia "MCAT_DB" de la instrucción están almacenados los siguientes valores:

Parámetro	Dirección	Valor
ET	DBD4	L#0
O_HIS	DBX16.0	TRUE
CMD_HIS	DBX16.1	FALSE

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

IMC: Comparar bits de entrada con bits de una máscara

Descripción

La instrucción "Comparar bits de entrada con bits de una máscara" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con el bit correspondiente de una máscara. Es posible programar como máximo 16 pasos con máscaras. El valor del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número del paso. El número del paso de la máscara utilizada para la comparación se define en el parámetro CMP_STEP. De esta misma manera se comparan todos los valores programados. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE.

Si se encuentra una coincidencia al realizar la comparación, el estado lógico del parámetro OUT se pone a "1". De lo contrario, el parámetro OUT se pone a "0".

Si el valor del parámetro CMP_STEP es mayor que 15, no se ejecuta la instrucción. En el parámetro ERR_CODE se emite un mensaje de error.

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Comparar bits de entrada con bits de una máscara" se utiliza la siguiente sintaxis:

GRAPH

```
CALL IMC, <DB instancia>
     (IN_BIT0 - 15 := <Operando>
      CMP_STEP := <Operando>
      OUT => <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Comparar bits de entrada con bits de una máscara":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
CMP_STEP	Input	BYTE	I, Q, M, D, L, P o constante	Número de paso de la máscara con la que se compara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una coincidencia. El estado lógico "0" indica que no se ha encontrado ninguna coincidencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
CMP_VAL	Static	ARRAY OF BOOL	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número de la etapa y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000A	El valor del parámetro CMP_STEP es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SMC: Barrido de matriz

Descripción

La instrucción "Barrido de matriz" compara el estado lógico de hasta 16 bits de entrada programados (IN_BIT0 a IN_BIT15) con los bits correspondientes de las máscaras de comparación de cada paso. La ejecución empieza por el paso 1 y continúa hasta el último paso programado (LAST) o hasta que se encuentre una correspondencia. El bit de entrada del parámetro IN_BIT0 se compara con el valor de la máscara CMP_VAL[x,0], siendo "x" el número de paso. De esta misma manera se comparan todos los valores programados. Si se encuentra una correspondencia, el estado lógico del parámetro OUT se pone a "1" y el número de paso se escribe con la máscara correspondiente en el parámetro OUT_STEP. Los bits de entrada no programados o los bits no programados de la máscara tienen el estado lógico predeterminado FALSE. Si varios pasos tienen una máscara equivalente, se indica solo la

primera correspondencia encontrada en el parámetro OUT_STEP. Si no se encuentra ninguna correspondencia, el estado lógico del parámetro OUT se pone a "0". En este caso, el valor del parámetro OUT_STEP excede en "1" el valor del parámetro LAST.

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Barrido de matriz" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SMC, <DB instancia>
     (IN_BIT0 - 15 := <Operando>
      OUT => <Operando>
      OUT_STEP => <Operando>
      ERR_CODE => <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Barrido de matriz":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT0	Input	BOOL	I, Q, M, D, L	El bit de entrada 0 se compara con el bit 0 de la máscara.
IN_BIT1	Input	BOOL	I, Q, M, D, L	El bit de entrada 1 se compara con el bit 1 de la máscara.
IN_BIT2	Input	BOOL	I, Q, M, D, L	El bit de entrada 2 se compara con el bit 2 de la máscara.
IN_BIT3	Input	BOOL	I, Q, M, D, L	El bit de entrada 3 se compara con el bit 3 de la máscara.
IN_BIT4	Input	BOOL	I, Q, M, D, L	El bit de entrada 4 se compara con el bit 4 de la máscara.
IN_BIT5	Input	BOOL	I, Q, M, D, L	El bit de entrada 5 se compara con el bit 5 de la máscara.
IN_BIT6	Input	BOOL	I, Q, M, D, L	El bit de entrada 6 se compara con el bit 6 de la máscara.
IN_BIT7	Input	BOOL	I, Q, M, D, L	El bit de entrada 7 se compara con el bit 7 de la máscara.
IN_BIT8	Input	BOOL	I, Q, M, D, L	El bit de entrada 8 se compara con el bit 8 de la máscara.
IN_BIT9	Input	BOOL	I, Q, M, D, L	El bit de entrada 9 se compara con el bit 9 de la máscara.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_BIT10	Input	BOOL	I, Q, M, D, L	El bit de entrada 10 se compara con el bit 10 de la máscara.
IN_BIT11	Input	BOOL	I, Q, M, D, L	El bit de entrada 11 se compara con el bit 11 de la máscara.
IN_BIT12	Input	BOOL	I, Q, M, D, L	El bit de entrada 12 se compara con el bit 12 de la máscara.
IN_BIT13	Input	BOOL	I, Q, M, D, L	El bit de entrada 13 se compara con el bit 13 de la máscara.
IN_BIT14	Input	BOOL	I, Q, M, D, L	El bit de entrada 14 se compara con el bit 14 de la máscara.
IN_BIT15	Input	BOOL	I, Q, M, D, L	El bit de entrada 15 se compara con el bit 15 de la máscara.
OUT	Output	BOOL	I, Q, M, D, L	El estado lógico "1" indica que se ha encontrado una correspondencia. El estado lógico "0" indica que no se ha encontrado ninguna correspondencia.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
OUT_STEP	Output	BYTE	I, Q, M, D, L, P	Contiene el número de paso con la máscara correspondiente o el número de paso que excede en "1" el valor del parámetro LAST, si no se ha encontrado ninguna correspondencia.
LAST	Static	BYTE	I, Q, M, D, L, P	Indica el número del último paso en que se tiene que buscar una máscara correspondiente.
CMP_VAL	Static	ARRAY of WORD	I, Q, M, D, L	Máscaras de comparación [0 a 15, 0 a 15]: El primer número del índice es el número de la etapa y, el segundo, el número de bit de la máscara.

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
000E	El valor del parámetro LAST es mayor que 15.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

LEAD_LAG: Algoritmo Lead y Lag

Descripción

La instrucción "Algoritmo Lead y Lag" procesa señales con una variable analógica. El valor para la amplificación en el parámetro GAIN debe ser mayor que cero. El resultado de la instrucción "Algoritmo Lead y Lag" se calcula con la siguiente ecuación:

$$OUT = \left[\frac{LG_TIME}{LG_TIME + SAMPLE_T}\right] PREV_OUT + GAIN \left[\frac{LD_TIME + SAMPLE_T}{LG_TIME + SAMPLE_T}\right] IN - GAIN \left[\frac{LD_TIME}{LG_TIME + SAMPLE_T}\right] * PREV_IN$$

La instrucción "Algoritmo Lead y Lag" sólo devuelve resultados relevantes cuando se ejecuta en ciclos de programa fijos. Deben indicarse las mismas unidades en los parámetros LD_TIME, LG_TIME y SAMPLE_T. Si LG_TIME > 4 + SAMPLE_T, la instrucción se aproxima a la siguiente función:

OUT = GAIN * ((1 + LD_TIME * s) / (1 + LG_TIME * s)) * IN

Si el valor del parámetro GAIN es menor o igual a cero, no se realiza el cálculo y el parámetro ERR_CODE devuelve una información de error.

La instrucción "Algoritmo Lead y Lag" puede utilizarse junto con bucles para la compensación en el control anticipativo. La instrucción comprende dos operaciones. La operación "Lead" desplaza la fase de la salida OUT, anticipándose la salida a la entrada. En cambio, la operación "Lag" desplaza la salida, de manera que la salida sigue a la entrada. Puesto que la operación "Lag" equivale a una integración, puede utilizarse para suprimir perturbaciones o como filtro pasa bajo. La operación "Lead" equivale a una acción diferencial, por lo que puede utilizarse como un filtro de paso alto. La combinación de ambas operaciones (Lead y Lag) hace que la fase de salida siga a la entrada a bajas frecuencias y se le anticipe a altas frecuencias. Por este motivo, la instrucción "Algoritmo Lead y Lag" puede utilizarse como si fuera un filtro de paso de banda.

Al insertar la instrucción en el programa se abre automáticamente el cuadro de diálogo "Opciones de llamada". El bloque de datos creado se encuentra en la carpeta "Recursos del

programa" del árbol del proyecto, en "Bloques de programa > Bloques de sistema". Encontrará más información al respecto en "Consulte también".

Sintaxis

Para la instrucción "Algoritmo Lead y Lag" se utiliza la siguiente sintaxis:

GRAPH

```
CALL LEAD_LAG, <DB instancia>
      (IN := <Operando>
       SAMPLE_T := <Operando>
       OUT => <Operando>
       ERR_CODE => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Algoritmo Lead y Lag":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	REAL	I, Q, M, D, L, P o constante	Valor de entrada del tiempo de muestreo actual (tiempo de ciclo) que se procesa.
SAMPLE_T	Input	INT	I, Q, M, D, L, P o constante	Tiempo de muestreo
OUT	Output	REAL	I, Q, M, D, L, P	Resultado de la instrucción
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Información de error
LD_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de anticipación en la misma unidad que el tiempo de muestreo
LG_TIME	Static	REAL	I, Q, M, D, L, P o constante	Tiempo de seguimiento en la misma unidad que el tiempo de muestreo.
GAIN	Static	REAL	I, Q, M, D, L, P o constante	Ganancia en % / % (relación del cambio en la salida con respecto al cambio en la entrada como estado constante).

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PREV_IN	Static	REAL	I, Q, M, D, L, P o constante	Entrada anterior
PREV_OUT	Static	REAL	I, Q, M, D, L, P o constante	Salida anterior

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Parámetro ERR_CODE

La tabla siguiente muestra el significado de los valores del parámetro ERR_CODE:

Código de error* (W#16#...)	Explicación
0000	Ningún error
0009	El valor del parámetro GAIN es menor o igual a cero.
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".	

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

Nota

Los parámetros estáticos pueden inicializarse en el bloque de datos.

GRAPH

```
CALL LEAD_LAG, "LEAD_LAG_DB"
     (IN := "Tag_Input"
      SAMPLE_T := "Tag_Input_SAMPLE_T"
      OUT => "Tag_Output_Result"
      ERR_CODE => "Tag_ErrorCode"
      )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada:

Parámetro	Operando	Valor
IN	Tag_Input	2.0
SAMPLE_T	Tag_Input_SAMPLE_T	10

Los siguientes valores están almacenados en el bloque de datos de instancia "LEAD_LAG_DB" de la instrucción:

Parámetro	Dirección	Valor
LD_TIME	DBD12	2.0
LG_TIME	DBD16	2.0
GAIN	DBD20	1.0
PREV_IN	DBD24	6.0
PREV_OUT	DBD28	6.0

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
OUT	Tag_Output_Result	2.0

Los siguientes valores se almacenan en el bloque de datos de instancia LEAD_LAD_DB de la instrucción:

Parámetro	Operando	Valor
PREV_IN	DBD24	2.0
PREV_OUT	DBD28	2.0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

TONR_X: Acumulador de tiempo

Descripción

La instrucción "Acumulador de tiempo" permite acumular valores de tiempo dentro de un periodo definido por el parámetro PT. Si el estado lógico del parámetro TMR_EN cambia a "1", se ejecuta la instrucción y se empieza a contar el tiempo PT. La instrucción "Acumulador de tiempo" suma el valor del parámetro DELTA_T al del parámetro ET siempre que se cumplan las siguientes condiciones:

- El parámetro TMR_EN devuelve el estado lógico "1".
- El valor de tiempo del parámetro ET es menor que el del parámetro PT.
- El parámetro RESET devuelve el estado lógico "0".

Si el valor del parámetro TMR_EN cambia a "0", el tiempo del parámetro DELTA_T se dejará de sumar al valor de tiempo del parámetro ET.

Cuando el valor de tiempo del parámetro ET alcanza el del parámetro PT, el parámetro Q devuelve el estado lógico "1". Mientras el parámetro Q devuelva el estado lógico "1", el valor

del parámetro ET no se modificará. El valor del parámetro ET se pone a "0" cuando el parámetro RESET se pone a "1".

Puesto que la instrucción "Acumulador de tiempo" utiliza el tiempo de ejecución (DELTA_T) del último ciclo del bloque de organización (OB), la instrucción solo puede llamarse en bloques de organización cíclicos.

Nota

El tiempo de ciclo del bloque de organización debe desplazarse desde la información de arranque a la variable global del parámetro DELTA_T.

Sintaxis

Para la instrucción "Acumulador de tiempo" se utiliza la siguiente sintaxis:

GRAPH

```
CALL TONR_X
     (TMR_EN := <Operando>
      RESET := <Operando>
      PV := <Operando>
      DELTA_T := <Operando>
      Q => <Operando>
      ET := <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Acumulador de tiempo":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TMR_EN	Input	BOOL	I, Q, M, D, L	Activa el tiempo para acumular el valor de tiempo
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
PV	Input	DINT	I, Q, M, D, L, P o constante	Tiempo máximo de lectura del tiempo
DELTA_T	Input	INT	I, Q, M, D, L o constante	Tiempo de ciclo del OB del ciclo anterior
Q	Output	BOOL	Q, M, D, L,	Salida que se activa una vez transcurrido el tiempo PT
ET	InOut	DINT	I, Q, M, D, L	Valor de tiempo actual del tiempo transcurrido

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

WSR: Guardar datos en el registro de desplazamiento

Descripción

La instrucción "Guardar datos en el registro de desplazamiento" permite escribir datos en un registro de desplazamiento desde el origen indicado. Todo elemento del registro se desplaza a la siguiente dirección. Los datos contenidos en la última dirección del registro de desplazamiento se pierden tras la operación de desplazamiento.

Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen (S_DATA) y se desplazan a la dirección inicial (START) del registro de desplazamiento. El parámetro LENGTH indica el número de elementos que deben desplazarse. El tipo de datos de los elementos se define en el parámetro E_TYPE. Si se indica un tipo de datos no admisible en el parámetro E_TYPE, no se procesa la instrucción.

Si el parámetro RESET está a "1" al ejecutarse la instrucción, se borra el registro. El parámetro Q se pone al estado lógico "1" si el registro de desplazamiento está ocupado con ceros o si se borra.

Sintaxis

Para la instrucción "Guardar datos en el registro de desplazamiento" se utiliza la siguiente sintaxis:

GRAPH

```
CALL WSR
      (RESET := <Operando>
       S_DATA := <Operando>
       START := <Operando>
       LEN := <Operando>
       E_TYPE := <Operando>
       Q => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Guardar datos en el registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_DATA	Input	POINTER	I, Q, M, D	Apunta al elemento de origen que se desplaza a la dirección inicial (START). En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
START	Input	POINTER	I, Q, M, D	Apunta a la dirección inicial del registro. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
LEN	Input	WORD	I, Q, M, D, L, P	Número de elementos que deben desplazarse.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de los elementos que deben desplazarse. Los siguientes tipos de datos son admisibles: • W#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
Q	Output	BOOL	Q, M, D, L,	Indica "0" si el parámetro RESET está activado (1) o si todos los elementos que deben desplazarse tienen el valor "0".

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

En el ejemplo siguiente se desplazan cuatro elementos del tipo de datos WORD a partir de la dirección indicada en el parámetro START. Una vez desplazado el primer elemento del registro a la siguiente dirección, la primera dirección se ocupará con los datos a los que señala el puntero en el parámetro S_DATA. El valor del último elemento del registro se pierde entonces.

GRAPH

```
CALL WSR
      (RESET := "Tag_Input_Reset"
       S_DATA := "Tag_Input_DataSource"
       START := "Tag_Input_StartAddress"
       LEN := "Tag_Input_Length"
       E_TYPE := "Tag_Input_ElementType"
       Q => "Tag_Output_Q"
       )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1111 DW2 = W#16#2222 DW4 = W#16#3333 DW6 = W#16#4444
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Después de la ejecución

Una vez ejecutada la instrucción, los valores siguientes se escriben en los parámetros de salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
S_DATA	Tag_Input_DataSource	P#DB2.DBX6.0 DW0 = W#16#1234 DW2 = W#16#1111 DW4 = W#16#2222 DW6 = W#16#3333
START	Tag_Input_StartAddress	P#DB1.DBX0.0 DW6 = W#16#1234
LEN	Tag_Input_Length	W#16#04
E_TYPE	Tag_Input_ElementType	B#16#04
Q	Tag_Output_Q	1

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

SHRB: Desplazar bit al registro de desplazamiento

Descripción

La instrucción "Desplazar bit al registro de desplazamiento" permite escribir un bit en un registro de desplazamiento desde el origen indicado (DATA). Cada vez que la instrucción se ejecuta con el valor "0" en el parámetro RESET, se leen datos nuevos del origen y se desplazan a la dirección inicial (S_BIT) del registro de desplazamiento. Los restantes bits se

desplazan una posición. El bit de la última dirección (S_BIT + N) se pierde tras el desplazamiento. Si el parámetro RESET está a "1" mientras se ejecuta la instrucción, las direcciones del registro de desplazamiento se ponen a "0" y no se desplazan.

Sintaxis

Para la instrucción "Desplazar bit al registro de desplazamiento" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SHRB
     (DATA := <Operando>
      RESET := <Operando>
      S_BIT := <Operando>
      N := <Operando>
      )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Desplazar bit al registro de desplazamiento":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DATA	Input	BOOL	I, Q, M, D, L	Bit de origen
RESET	Input	BOOL	I, Q, M, D, L	Entrada de reset
S_BIT	Input	POINTER	I, Q, M, D	Apunta al primer bit del registro de desplazamiento. En el direccionamiento indirecto por registro interárea, el puntero debe indicarse en formato de palabra doble.
N	Input	WORD	I, Q, M, D, L, P	Número de bits que deben desplazarse.

Ejemplo

En el ejemplo siguiente, el parámetro N se pone al valor 14 ("E" en notación hexadecimal), con lo que 14 bits se desplazan a partir de la dirección indicada en el parámetro S_BIT. Una vez desplazados los bits, la primera dirección se ocupa con los datos del origen del parámetro DATA. El último valor de bit se pierde.

GRAPH

```
CALL SHRB
```

GRAPH

```
(DATA := "Tag_Input_Data"
 RESET := "Tag_Input_Reset"
 S_BIT := "Tag_Input_BitSource"
 N := "Tag_Input_Number"
 )
```

Las tablas siguientes muestran el funcionamiento de la instrucción con valores concretos.

Antes de la ejecución

En este ejemplo se utilizan los siguientes valores para los parámetros de entrada y salida:

Parámetro	Operando	Valor
RESET	Tag_Input_Reset	0
DATA	Tag_Input_Data	0
S_BIT	Tag_Input_BitSource	P#DB1.DBX1.4
N	Tag_Input_Number	B#16#0E

Antes de ejecutarse la instrucción, los siguientes estados lógicos están almacenados en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	0	1	1	1	0	1	0

2.7	...					...	2.0
0	0	1	1	1	0	1	0

3.7	...					...	3.0
0	0	1	1	1	0	1	0

Después de la ejecución

Una vez ejecutada la instrucción, los siguientes estados lógicos se almacenan en los 14 bits del registro:

1.7	...		S_BIT			...	1.0
0	1	1	0	1	0	1	0

2.7	...					...	2.0
0	1	1	1	0	1	0	0

3.7	...					...	3.0
0	0	1	1	1	0	0	0

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

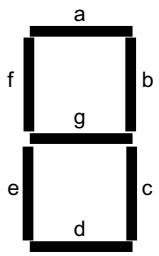
Principios básicos de GRAPH (Página 2825)

SEG: Generar patrón de bits para el display de 7 segmentos

Descripción

La instrucción "Generar patrón de bits para el display de 7 segmentos" permite convertir cada una de las cuatro cifras hexadecimales de la palabra de origen indicada (IN) en un patrón de bits equivalente para un display de 7 segmentos. El resultado de la instrucción se devuelve en la palabra doble del parámetro OUT.

Entre las cifras hexadecimales y la ocupación de los 7 segmentos (a, b, c, d, e, f, g) existe la siguiente relación:

Cifra de entrada (binaria)	Ocupación de los segmentos - g f e d c b a	Display (hexadecimal)	Display de siete segmentos
0000	00111111	0	
0001	00000110	1	
0010	01011011	2	
0011	01001111	3	
0100	01100110	4	
0101	01101101	5	
0110	01111101	6	
0111	00000111	7	
1000	01111111	8	
1001	01100111	9	
1010	01110111	A	
1011	01111100	B	
1100	00111001	C	
1101	01011110	D	
1110	01111001	E	
1111	01110001	F	

Sintaxis

Para la instrucción "Generar patrón de bits para el display de 7 segmentos" se utiliza la siguiente sintaxis:

GRAPH

```
CALL SEG
```

GRAPH

```
(IN := <Operando>
 OUT => <Operando>
 )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar patrón de bits para el display de 7 segmentos":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	WORD	I, Q, M, D, L, P o constante	Palabra de origen con cuatro cifras hexadecimales
OUT	Output	DWORD	I, Q, M, D, L, P	Patrón de bits para el display de 7 segmentos

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL SEG
      (IN := "Tag_Input"
       OUT => "Tag_Output"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores de operandos concretos:

Parámetro	Operando	Valor	
Hexadecimal	Binario		
IN	Tag_Input	W#16#1234	0001 0010 0011 0100
OUT	Tag_Output	DW#16065B4F66	000 00110 0101 1011 0100 1111 0110 0110 Visualización: 1234

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

BCDCPL: Generar complemento a diez

Descripción

La instrucción "Generar complemento a diez" crea el complemento a diez de un número BCD de siete dígitos que se indica en el parámetro IN. La instrucción realiza el cálculo con la siguiente fórmula matemática:

10000000 (como BCD)

– Valor BCD de 7 dígitos

Complemento a diez (como BCD)

Sintaxis

Para la instrucción "Generar complemento a diez" se utiliza la siguiente sintaxis:

GRAPH

```
CALL BCDCPL
      (IN := <Operando>
       ERR_CODE => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Generar complemento a diez":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Número BCD de 7 dígitos
ERR_CODE	Output	DWORD	I, Q, M, D, L, P	Resultado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Consulte también".

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL BCDCPL
      (IN := "Tag_Input"
       ERR_CODE => "Tag_Output"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#01234567
ERR_CODE	Tag_Output	DW#16#08765433
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

BITSUM: Contar número de bits activados

Descripción

La instrucción "Contar número de bits activados" permite contar el número de bits de un operando que están puestos al estado lógico "1". El operando cuyos bits se cuentan se especifica en el parámetro IN. El resultado de la instrucción se devuelve en el parámetro RET_VAL.

Sintaxis

Para la instrucción "Contar número de bits activados" se utiliza la siguiente sintaxis:

GRAPH

```
CALL BITSUM
      (IN := <Operando>
       RET_VAL => <Operando>
       )
```

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Contar número de bits activados":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DWORD	I, Q, M, D, L, P o constante	Operando cuyos bits activados se cuentan
RET_VAL	Output	INT	I, Q, M, D, L, P	Número de bits activados

Ejemplo

El siguiente ejemplo muestra el funcionamiento de la instrucción:

GRAPH

```
CALL BITSUM
      (IN := "Tag_Input"
       RET_VAL => "Tag_Output"
       )
```

La tabla siguiente muestra el funcionamiento de la instrucción con valores concretos:

Parámetro	Operando	Valor*
IN	Tag_Input	DW#16#12345678
RET_VAL	Tag_Output	W#16#000D (13 bits)
*Los códigos de error en el editor de programas se pueden representar como valores enteros o hexadecimales. Encontrará más información sobre el cambio de los formatos de visualización en "Consulte también".		

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

Principios básicos de la palabra de estado (Página 2545)

Principios básicos de GRAPH (Página 2825)

Instrucciones KOP Comparación

Instrucciones KOP generales

Descripción

Para la programación en GRAPH se dispone de diferentes instrucciones KOP.

La tabla siguiente ofrece una vista general de las instrucciones y su disponibilidad:

Instrucciones KOP	Instrucción permanente	Transición/ Supervisión/ Interlock
General		
Insertar segmento	x	
Cuadro vacío	x	x
Abrir rama	x	x
Cerrar rama	x	x
Operaciones lógicas con bits		
---| |---: Contacto normalmente abierto (Página 5602)	x	x

Instrucciones KOP	Instrucción permanente	Transición/ Supervisión/ Interlock
---| / |---: Contacto normalmente cerrado (Página 5603)	x	x
--| NOT |--: Invertir RLO (Página 5604)	x	x
---( )---: Asignación (Página 5605)	x	
---(R)---: Desactivar salida (Página 5606)	x	
---(S)---: Activar salida (Página 5607)	x	
SR: Flipflop de activación/desactivación (Página 5609)	x	
RS: Flipflop de desactivación/activación (Página 5610)	x	
--|P|--: Consultar flanco de señal ascendente de un operando (Página 5612)	x	
--|N|--: Consultar flanco de señal descendente de un operando (Página 5613)	x	
P_TRIG: Consultar flanco de señal ascendente del RLO (Página 5614)	x	
N_TRIG: Consultar flanco de señal descendente del RLO (Página 5616)	x	
Temporizadores		
Temporizadores CEI		
TP: Impulso (Página 5617)	x	
TON: Retardo al conectar (Página 5619)	x	
TOF: Retardo al desconectar (Página 5621)	x	
Temporizadores SIMATIC		
S_PULSE: Parametrizar y arrancar temporizador como impulso (Página 5624)	x	
S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado (Página 5627)	x	
S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión (Página 5630)	x	
S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria (Página 5633)	x	
S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión (Página 5636)	x	
Contadores		
Contadores CEI		
CTU: Contador ascendente (Página 5647)	x	
CTD: Contador descendente (Página 5649)	x	
CTUD: Contador ascendente - descendente (Página 5651)	x	
Contadores SIMATIC		
S_CU : Parametrizar e incrementar contador (Página 5656)	x	
S_CD : Parametrizar y decrementar contador (Página 5658)	x	
S_CUD : Parametrizar e incrementar/decrementar contador (Página 5654)	x	
Comparación		
CMP ==: Igual (Página 5664)	x	x
CMP <>: Diferente (Página 5665)	x	x
CMP >=: Mayor o igual (Página 5667)	x	x
CMP <=: Menor o igual (Página 5668)	x	x
CMP >: Mayor (Página 5669)	x	x
CMP <: Menor (Página 5670)	x	x
CMP>T: Superior al tiempo de activación de la etapa (Página 6676)		x
CMP>U: Superior al tiempo de activación ininterrumpido de la etapa (Página 6676)		x

Instrucciones KOP	Instrucción permanente	Transición/ Supervisión/ Interlock
Funciones matemáticas		
ADD: Sumar (Página 5672)	x	
SUB: Restar (Página 5673)	x	
MUL: Multiplicar (Página 5675)	x	
DIV: Dividir (Página 5676)	x	
MOD: Obtener resto de división (Página 5678)	x	
NEG: Generar complemento a dos (Página 5679)	x	
ABS: Calcular valor absoluto (Página 5680)	x	
MIN: Determinar mínimo (Página 5682)	x	
MAX: Determinar máximo (Página 5684)	x	
LIMIT: Ajustar valor límite (Página 5685)	x	
SQR: Calcular cuadrado (Página 5688)	x	
SQRT: Calcular raíz cuadrada (Página 5689)	x	
LN: Calcular logaritmo natural (Página 5690)	x	
EXP: Calcular valor exponencial (Página 5692)	x	
SIN: Calcular valor de seno (Página 5693)	x	
COS: Calcular valor de coseno (Página 5694)	x	
TAN: Calcular valor de tangente (Página 5696)	x	
ASIN: Calcular valor de arcoseno (Página 5697)	x	
ACOS: Calcular valor de arcocoseno (Página 5699)	x	
ATAN: Calcular valor de arcotangente (Página 6435)	x	
Transferencia		
MOVE: Copiar valor (Página 5702)	x	
BLKMOV: Copiar área (Página 5704)	x	
UBLKMOV: Copiar área sin interrupciones (Página 5708)	x	
FILL: Rellenar área (Página 5710)	x	
Conversión		
CONVERT: Convertir valor (Página 5713)	x	
ROUND: Redondear número (Página 5715)	x	
CEIL: Redondear número en coma flotante al siguiente entero superior (Página 5716)	x	
FLOOR: Redondear número en coma flotante al siguiente entero inferior (Página 5718)	x	
TRUNC: Truncar a entero (Página 5719)	x	
SCALE: Escalar (Página 5721)	x	
UNSCALE: Desescalar (Página 5723)	x	
Control del programa		
Control en runtime		
RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo (Página 5733)	x	
STP: Finalizar programa (Página 5733)	x	
WAIT: Programar retardo (Página 5734)	x	
PROTECT: Cambiar nivel de protección (Página 5735)	x	
COMPRESS: Comprimir memoria CPU (S7-400) (Página 5731)	x	

Instrucciones KOP	Instrucción permanente	Transición/ Supervisión/ Interlock
CiR: Controlar proceso CiR (S7-400) (Página 5737)	x	
Operaciones lógicas con palabras		
AND: Operación lógica Y (Página 5740)	x	
OR: Operación lógica O (Página 5741)	x	
XOR: Operación lógica O-exclusiva (Página 5743)	x	
INV: Complemento a 1 (Página 5744)	x	
DECO: Descodificar (Página 5746)	x	
ENCO: Codificar (Página 5747)	x	
SEL: Seleccionar (Página 5748)	x	
Desplazamiento y rotación		
SHR: Desplazar a la derecha (Página 5750)	x	
SHL: Desplazar a la izquierda (Página 5752)	x	
ROR: Rotar a la derecha (Página 5754)	x	
ROL: Rotar a la izquierda (Página 5756)	x	
Otras instrucciones		
SET: Activar área por bits (Página 5772)	x	
SETP: Activar mapa de bits en el área de periferia (S7-400) (Página 5773)	x	
SETI: Activar área por bytes (Página 5775)	x	
RESET: Desactivar área por bits (Página 5777)	x	
RESETP: Desactivar mapa de bits en el área de periferia (S7-400) (Página 5779)	x	
RESETI: Desactivar área por bytes (Página 5781)	x	
REPL_VAL: Introducir valor sustitutivo (Página 5783)	x	
DRUM: Materializar circuito secuencial sucesivo (Página 5784)	x	
DRUM_X: Materializar circuito secuencial sucesivo (Página 5788)	x	
DCAT: Alarma de tiempo para control discreto (Página 5792)	x	
MCAT: Alarma de tiempo para control del motor (Página 5795)	x	
IMC: Comparar los bits de entrada con los bits de la máscara (Página 5799)	x	
SMC: Barrido de matriz (Página 5802)	x	
LEAD_LAG: Algoritmo Lead y Lag (Página 5804)	x	
TONR_X: Acumulador de tiempo (Página 5808)	x	
WSR: Guardar datos en el registro de desplazamiento (Página 5809)	x	
SHRB: Desplazar bit al registro de desplazamiento (Página 5813)	x	
SEG: Generar patrón de bits para el display de 7 segmentos (Página 5815)	x	
BCDCPL: Generar complemento a diez (Página 5817)	x	
BITSUM: Contar número de bits activados (Página 5818)	x	

CMP>T: Superior al tiempo de activación de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación de la etapa" se vigila la duración total de una etapa en condiciones de supervisión. De esta manera se registra el tiempo de los posibles eventos o errores.

En la condición programada se compara el tiempo de activación en curso o el último de una etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

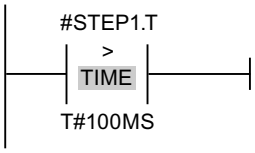
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa
<Operando2>	Input	DINT, TIME, S5TIME	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.T esté por debajo de 100 ms, el RLO será "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

CMP>U: Superior al tiempo de activación ininterrumpido de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación ininterrumpido de la etapa", se vigila la duración de una etapa, sin tener en cuenta las posibles averías, en condiciones de supervisión. No se registra el tiempo de posibles eventos o fallos sino que se supervisa la duración de la mera etapa.

En la condición programada se compara todo el tiempo de activación de la etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la

comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

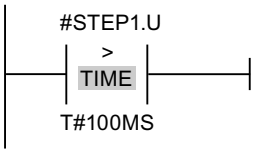
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación ininterrumpido de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa sin contar averías
<Operando2>	Input	DINT, TIME, S5TIME	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.U, sin tener en cuenta las posibles averías, esté por debajo de 100 ms, el RLO será igual a "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

Instrucciones FUP Comparación

Instrucciones FUP generales

Descripción

Para la programación en GRAPH se dispone de diferentes instrucciones FUP.

La tabla siguiente ofrece una vista general de las instrucciones y su disponibilidad:

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
General		
Nuevo segmento: Insertar segmento	x	
Cuadro vacío	x	x
Derivación: Abrir rama	x	x

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
-|: Insertar entrada (Página 5824)	x	x
-o|: Invertir RLO (Página 5825)	x	x
Operaciones lógicas con bits		
&: Operación lógica Y (Página 5819)	x	x
>=1: Operación lógica O (Página 5821)	x	x
x: Operación lógica O-exclusiva (Página 5823)	x	
= : Asignación (Página 5826)	x	
-(R): Desactivar salida (Página 5827)	x	
-(S): Activar salida (Página 5828)	x	
SR: Flipflop de activación/desactivación (Página 5829)	x	
RS: Flipflop de desactivación/activación (Página 5831)	x	
-|P|-: Consultar flanco de señal ascendente de un operando (Página 5832)	x	
-|N|-: Consultar flanco de señal descendente de un operando (Página 5834)	x	
P_TRIG: Consultar flanco de señal ascendente del RLO (Página 5835)	x	
N_TRIG: Consultar flanco de señal descendente del RLO (Página 5836)	x	
Temporizadores		
Temporizadores CEI		
TP: Impulso (Página 5838)	x	
TON: Retardo al conectar (Página 5840)	x	
TOF: Retardo al desconectar (Página 5842)	x	
Temporizadores SIMATIC		
S_PULSE: Parametrizar y arrancar temporizador como impulso (Página 5845)	x	
S_PEXT: Parametrizar y arrancar temporizador como impulso prolongado (Página 5848)	x	
S_ODT: Parametrizar y arrancar temporizador como retardo a la conexión (Página 5851)	x	
S_ODTS: Parametrizar y arrancar temporizador como retardo a la conexión con memoria (Página 5854)	x	
S_OFFDT: Parametrizar y arrancar temporizador como retardo a la desconexión (Página 5857)	x	
Contadores		
Contadores CEI		
CTU: Contador ascendente (Página 5870)	x	
CTD: Contador descendente (Página 5872)	x	
CTUD: Contador ascendente - descendente (Página 5874)	x	
Contadores SIMATIC		
S_CU : Parametrizar e incrementar contador (Página 5879)	x	
S_CD : Parametrizar y decrementar contador (Página 5881)	x	
S_CUD : Parametrizar e incrementar/decrementar contador (Página 5877)	x	
Comparación		
CMP ==: Igual (Página 5887)	x	x
CMP <>: Diferente (Página 5889)	x	x
CMP >=: Mayor o igual (Página 5890)	x	x
CMP <=: Menor o igual (Página 5891)	x	x

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
CMP >: Mayor (Página 5892)	x	x
CMP <: Menor (Página 5893)	x	x
CMP>T: Superior al tiempo de activación de la etapa (Página 6681)		x
CMP>U: Superior al tiempo de activación ininterrumpido de la etapa (Página 6682)		x
Funciones matemáticas		
ADD: Sumar (Página 5894)	x	
SUB: Restar (Página 5896)	x	
MUL: Multiplicar (Página 5897)	x	
DIV: Dividir (Página 5899)	x	
MOD: Obtener resto de división (Página 5900)	x	
NEG: Generar complemento a dos (Página 5902)	x	
ABS: Calcular valor absoluto (Página 5903)	x	
MIN: Determinar mínimo (Página 5904)	x	
MAX: Determinar máximo (Página 5906)	x	
LIMIT: Ajustar valor límite (Página 5908)	x	
SQR: Calcular cuadrado (Página 5910)	x	
SQRT: Calcular raíz cuadrada (Página 5912)	x	
LN: Calcular logaritmo natural (Página 5913)	x	
EXP: Calcular valor exponencial (Página 5915)	x	
SIN: Calcular valor de seno (Página 5916)	x	
COS: Calcular valor de coseno (Página 5917)	x	
TAN: Calcular valor de tangente (Página 5919)	x	
ASIN: Calcular valor de arcoseno (Página 5920)	x	
ACOS: Calcular valor de arcocoseno (Página 5922)	x	
ATAN: Calcular valor de arcotangente (Página 5923)	x	
Transferencia		
MOVE: Copiar valor (Página 5925)	x	
BLKMOV: Copiar área (Página 5927)	x	
UBLKMOV: Copiar área sin interrupciones (Página 5931)	x	
FILL: Rellenar área (Página 5933)	x	
Conversión		
CONVERT: Convertir valor (Página 5936)	x	
ROUND: Redondear número (Página 5938)	x	
CEIL: Redondear número en coma flotante al siguiente entero superior (Página 5939)	x	
FLOOR: Redondear número en coma flotante al siguiente entero inferior (Página 5941)	x	
TRUNC: Truncar a entero (Página 5942)	x	
SCALE: Escalar (Página 5944)	x	
UNSCALE: Desescalar (Página 5946)	x	
Control del programa		
Control en runtime		
RE_TRIGR: Reiniciar tiempo de vigilancia del ciclo (Página 5955)	x	

Instrucción FUP	Operación permanente	Transición/ Supervisión/ Interlock
STP: Finalizar programa (Página 5956)	x	
PROTECT: Cambiar nivel de protección (Página 5957)	x	
WAIT: Programar retardo (Página 5957)	x	
COMPRESS: Comprimir memoria CPU (S7-400) (Página 5954)	x	
CiR: Controlar proceso CiR (S7-400) (Página 5960)	x	
Operaciones lógicas con palabras		
AND: Operación lógica Y (Página 5963)	x	
OR: Operación lógica O (Página 5964)	x	
XOR: Operación lógica O-exclusiva (Página 5966)	x	
INV: Complemento a 1 (Página 5967)	x	
DECO: Descodificar (Página 5969)	x	
ENCO: Codificar (Página 5970)	x	
SEL: Seleccionar (Página 5971)	x	
Desplazamiento y rotación		
SHR: Desplazar a la derecha (Página 5973)	x	
SHL: Desplazar a la izquierda (Página 5975)	x	
ROR: Rotar a la derecha (Página 5977)	x	
ROL: Rotar a la izquierda (Página 5979)	x	
Otras instrucciones		
SET: Activar área por bits (Página 5993)	x	
SETP: Activar mapa de bits en el área de periferia (S7-400) (Página 5995)	x	
SETI: Activar área por bytes (Página 5997)	x	
RESET: Desactivar área por bits (Página 5999)	x	
RESETP: Desactivar mapa de bits en el área de periferia (S7-400) (Página 6000)	x	
RESETI: Desactivar área por bytes (Página 6002)	x	
REPL_VAL: Introducir valor sustitutivo (Página 6004)	x	
DRUM: Materializar circuito secuencial sucesivo (Página 6005)	x	
DRUM_X: Materializar circuito secuencial sucesivo (Página 6009)	x	
DCAT: Alarma de tiempo para control discreto (Página 6013)	x	
MCAT: Alarma de tiempo para control del motor (Página 6016)	x	
IMC: Comparar los bits de entrada con los bits de la máscara (Página 6020)	x	
SMC: Barrido de matriz (Página 6022)	x	
LEAD_LAG: Algoritmo Lead y Lag (Página 6025)	x	
TONR_X: Acumulador de tiempo (Página 6028)	x	
WSR: Guardar datos en el registro de desplazamiento (Página 6030)	x	
SHRB: Desplazar bit al registro de desplazamiento (Página 6036)	x	
SEG: Generar patrón de bits para el display de 7 segmentos (Página 6040)	x	
BCDCPL: Generar complemento a diez (Página 6041)	x	
BITSUM: Contar número de bits activados (Página 6043)	x	

Consulte también

JMP: Saltar si RLO = 1 (Página 5948)

JMPN: Saltar si RLO = 0 (Página 5949)

LABEL: Etiqueta (Página 5950)

CMP>T: Superior al tiempo de activación de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación de la etapa" se vigila la duración total de una etapa en condiciones de supervisión. De esta manera se registra el tiempo de los posibles eventos o errores.

En la condición programada se compara el tiempo de activación en curso o el último de una etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

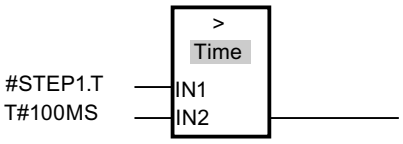
Parámetro

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa
<Operando2>	Input	DINT, TIME, S5TIME	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.T esté por debajo de 100 ms, el RLO será "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

CMP>U: Superior al tiempo de activación ininterrumpido de la etapa

Descripción

Con la instrucción "Superior al tiempo de activación ininterrumpido de la etapa", se vigila la duración de una etapa, sin tener en cuenta las posibles averías, en condiciones de supervisión. No se registra el tiempo de posibles eventos o fallos sino que se supervisa la duración de la mera etapa.

En la condición programada se compara todo el tiempo de activación de una etapa (operando1) con un tiempo definido (operando2) en milisegundos. Si se cumple la condición de la comparación, la instrucción devuelve el resultado lógico (RLO) "1". Si la condición de la comparación no se cumple, la instrucción devuelve el RLO "0".

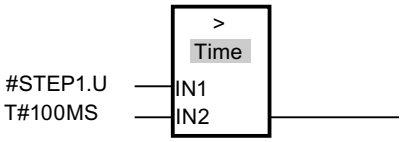
Parámetros

La tabla siguiente muestra los parámetros de la instrucción "Superior al tiempo de activación ininterrumpido de la etapa":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
<Operando1>	Input	Enteros	I, Q, M, D, L o constante	Tiempo de activación actual o último de la etapa sin contar averías
<Operando2>	Input	DINT, TIME, S5TIME	I, Q, M, D, L o constante	Tiempo con el que se compara

Ejemplo

El siguiente ejemplo muestra la instrucción en el segmento:



Mientras el tiempo de activación de #STEP1.U, sin tener en cuenta las posibles averías, esté por debajo de 100 ms, el RLO será igual a "0". En cuanto el tiempo de activación sobrepasa 100 ms, el RLO cambia a "1".

9.8.2.3 Instrucciones avanzadas

Fecha y hora

T_COMP: Comparar variables de tiempo

Descripción

Esta instrucción compara el contenido de dos variables del tipo de datos DT y devuelve el resultado de la comparación como valor de retorno en el parámetro OUT. El parámetro OUT se pone a "1" una vez que se ha cumplido la condición de comparación utilizada.

Existen las siguientes posibilidades de comparación:

Símbolo	Descripción
EQ	El valor de retorno tiene el estado lógico "1" si los momentos son idénticos en el parámetro IN1 y en el parámetro IN2.
NE	El valor de retorno tiene el estado lógico "1" si los momentos no son idénticos en el parámetro IN1 y en el parámetro IN2.
GE	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es mayor (más reciente) o igual al del parámetro IN2 .
LE	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es menor (más antiguo) o igual al del parámetro IN2.
GT	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es mayor (más reciente) que el del parámetro IN2.
LT	El valor de retorno tiene el estado lógico "1" si el momento del parámetro IN1 es menor (más antiguo) que el del parámetro IN2 .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_COMP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	DT	D, L o constante	Valor que se va a comparar
IN2	Input	DT	D, L o constante	Valor que se va a comparar
OUT	Output	BOOL	I, Q, M, D, L	Valor de retorno

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_CONV: Convertir y extraer tiempos

Descripción

La instrucción "T_CONV" convierte los tipos de datos DT (DATE_AND_TIME), S5TIME y TIME en los tipos de datos DATE, INT, TOD (TIME_OF_DAY), TIME o S5TIME.

Parámetros

El tipo de conversión se define seleccionando los tipos de datos en los cuadros de instrucción de la entrada y la salida de la instrucción. El resultado de la conversión puede consultarse en el parámetro OUT.

La tabla siguiente muestra los parámetros de la instrucción "T_CONV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DT, S5TIME, TIME	I, Q, M, D, L o constante	Valor que se va a convertir. El formato de datos de la variable del parámetro IN debe coincidir con el tipo de datos seleccionado en el cuadro de instrucción.
OUT	Return	DATE, INT, TOD, TIME, S5TIME	I, Q, M, D, L	Resultado de la conversión. El formato de datos de la variable del parámetro OUT debe coincidir con el tipo de datos seleccionado en el cuadro de instrucción.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Ejemplos

Tabla 9-173 Conversión del formato de datos TIME al formato S5TIME

Parámetro	Declaración	Tipo de datos	Descripción
IN	Output	TIME	Variable de entrada en el formato de datos TIME.
OUT	Input	S5TIME	La función convierte el formato de datos TIME al formato S5TIME. Al convertir se efectúa un redondeo. Si el parámetro de entrada es mayor que el formato representable de S5TIME (mayor que TIME#02:46:30.000), se devuelve como resultado S5TIME#999.3 y el resultado binario BR se ajusta a "0".

Tabla 9-174 Extracción del día de la semana desde el formato DATE_AND_TIME

Parámetro	Declaración	Tipo de datos	Descripción
IN	Output	DT	Variable de entrada en el formato de datos DATE_AND_TIME.
OUT	Input	INT	La función extrae el día de la semana desde el formato DATE_AND_TIME (DT). El día de la semana se devuelve en el formato de datos INTEGER: 1: Domingo 2: Lunes 3: Martes 4: Miércoles 5: Jueves 6: Viernes 7: Sábado

T_ADD: Sumar tiempos

Descripción

Con la instrucción "T_ADD" se suma a un momento (formato DT) una duración (formato TIME) y se devuelve como resultado un nuevo momento (formato DT).

El momento DT del parámetro IN1 debe encontrarse en el rango entre DT#1990-01-01-00:00:00.000 y DT#2089-12-31-23:59:59.999. Esta instrucción no efectúa ninguna comprobación de entrada. Si el resultado de la suma no se encuentra dentro del rango indicado arriba, se limita el resultado al valor correspondiente y el resultado binario BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_ADD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	DT	D, L o constante	Fecha y hora en el formato DT (DATE_AND_TIME).
IN2	Input	TIME	I, Q, M, D, L o constante	Duración que se debe sumar en el formato TIME.
OUT	Return	DT	D, L	Resultado de la suma en el formato DT (DATE_AND_TIME).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_SUB: Restar tiempos

Descripción

Con la instrucción "T_SUB" se resta de un momento (formato DT) una duración (formato TIME) y se devuelve como resultado un nuevo momento (formato DT).

El momento DT del parámetro IN1 debe encontrarse en el rango entre DT#1990-01-01-00:00:00.000 y DT#2089-12-31-23:59:59.999. Esta instrucción no efectúa ninguna comprobación de entrada. Si el resultado de la resta no se encuentra dentro del rango indicado arriba, se limita el resultado al valor correspondiente y el resultado binario BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_SUB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	DT	D, L o constante	Fecha y hora en el formato DT (DATE_AND_TIME).
IN2	Input	TIME	I, Q, M, D, L o constante	Duración que se debe restar en el formato TIME.
OUT	Return	DT	D, L	Resultado de la resta en el formato DT (DATE_AND_TIME).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_DIFF: Diferencia de tiempo

Descripción

Con la instrucción "T_DIFF" se calcula la diferencia entre dos momentos restando dos momentos con el formato DT. El momento en la entrada IN2 se resta del momento en la entrada IN1. El resultado de la resta se devuelve en la salida OUT en el formato TIME :

- Si el primer momento en la entrada es IN1 mayor (más reciente) que el segundo momento en la entrada IN2, el resultado en la salida OUT es positivo.
- Si el primer momento en la entrada es IN1 menor (más antiguo) que el segundo punto en la entrada IN2, el resultado en la salida OUT es negativo.

Si el resultado de la resta se encuentra fuera del rango numérico de TIME, se limita el resultado al valor correspondiente y el resultado binario BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_DIFF":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	DT	D, L o constante	Minuendo
IN2	Input	DT	D, L o constante	Sustraendo
OUT	Return	TIME	I, Q, M, D, L	Diferencia en el formato TIME.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

T_COMBINE: Combinar tiempos

Descripción

La instrucción "T_COMBINE" combina los formatos de datos DATE y TIME_OF_DAY (TOD) y los convierte al formato DATE_AND_TIME (DT). La instrucción no notifica errores.

- El valor de entrada DATE debe encontrarse entre DATE#1990-01-01 y DATE#2089-12-31 (no se comprueba).
- En el valor de entrada IN2 se utiliza el tipo de datos TIME_OF_DAY .
- En el valor de salida OUT se devuelve el formato de datos DATE_AND_TIME.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "T_COMBINE":

odParameter	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	DATE	D, L o constante	Variable de entrada en el formato DATE
IN2	Input	TOD	D, L o constante	Variable de entrada en el formato TOD
OUT	Return	DT	D, L	Valor de retorno en el formato DT

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Funciones de reloj

WR_SYS_T: Ajustar la hora

Descripción

Al llamar "WR_SYS_T" se ajusta la hora y la fecha del reloj de la CPU. A partir de ese momento, el reloj avanza desde la hora y fecha ajustadas. La información de fecha y hora se introduce en el formato de datos DT en la entrada IN de la instrucción. Para convertir los formatos DATE y TOD al formato de datos requerido DT se puede utilizar la instrucción "T_COMBINE (Página 6687)".

Si el reloj es un reloj maestro, al llamar la función, la CPU inicia adicionalmente la sincronización de la hora. Los intervalos de sincronización se definen en la configuración. La información de zona horaria local u horario de verano no se puede transferir con la instrucción "WR_SYS_T".

En la salida RET_VAL se puede consultar si se han producido errores durante la ejecución de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_SYS_T":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	DT	D, L	Fecha y hora
RET_VAL	Return	INT	I, Q, M, D, L	Estado de la instrucción

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#....)	Descripción
0000	Sin error
8081	Año no válido
8082	Mes no válido
8083	Día no válido
8084	Indicación de horas no válida
8085	Indicación de minutos no válida
8086	Indicación de segundos no válida
8087	Indicación de nanosegundos no válida
80B0	Fallo del reloj en tiempo real

Ejemplo

La fecha y la hora deben introducirse con el tipo de datos DT . Ejemplo para el 15 de enero de 2010, 10:30 h y 30 s: DT#2010-01-15-10:30:30. La hora solo se puede introducir con granularidad de segundos. El día de la semana se calcula a partir de la fecha usando la instrucción "WR_SYS_T".

RD_SYS_T: Leer la hora

Descripción

La instrucción "RD_SYS_T" permite leer la fecha y hora actuales del reloj de la CPU. Los datos leídos se devuelven en formato DT en la salida OUT de la instrucción. El valor emitido no contiene indicaciones respecto a la zona horaria local ni al horario de verano. En la salida RET_VAL se puede consultar si se han producido errores durante la ejecución de la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_SYS_T":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
OUT	Output	DT	D,L	Fecha y hora de la CPU

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#....)	Descripción
0000	Sin error
8222	El resultado se encuentra fuera del rango de valores admisible.
8223	El resultado no puede guardarse en el tipo de datos especificado.

TIME_TCK: Leer hora del sistema

Descripción

Con la instrucción "TIME_TCK" se lee la hora del sistema de la CPU. La hora del sistema es un contador de tiempo que contabiliza desde 0 a un máximo de 2147483647 ms. En caso de desbordamiento de la hora del sistema, se reinicia el recuento desde "0". El incremento de tiempo y la exactitud de la hora del sistema es de 1 ms. El único factor que influye en la hora del sistema son los estados operativos de la CPU. La hora del sistema se puede usar, por ejemplo, para medir la duración de procesos obteniendo la diferencia entre los resultados de dos llamadas de "TIME_TCK". Esta instrucción no ofrece información de error.

La tabla siguiente resume los cambios que experimenta la hora del sistema en función de los estados operativos de la CPU.

Estado operativo	La hora del sistema ...
Arranque	... se actualiza de modo permanente.
RUN	
STOP	... se detiene y mantiene el valor actual.
Rearranque (no con S7-300)	... continúa avanzando a partir del valor que se guardó al pasar al estado STOP.
Rearranque completo (en caliente)	... se borra y vuelve a empezar desde "0".
Arranque en frío	

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TIME_TCK":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	TIME	I, Q, M, D, L	El parámetro RET_VAL contiene la hora del sistema leída, en el rango de 0 a 2^{31} -1 ms.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

SET_CLKS: Ajustar la hora e inicializar el estado del reloj

Descripción

Con la instrucción "SET_CLKS" se ajusta la hora de la CPU y se inicializa el estado del reloj.

Nota

Active "SET_CLKS" solo si en su CPU no se sincroniza la hora. De lo contrario, en cada sincronización se adoptará el estado del reloj maestro. Al hacerlo, el valor predefinido con "SET_CLKS" se sobrescribe.

Nota

El estado actual de hora de la CPU se puede averiguar leyendo la ID de SZL W#16#0132 Index W#16#0008 (Página 6994) con la instrucción "RDSYSST (Página 6963)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_CLKS":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	Modo de operación: • B#16#01: Ajustar la hora • B#16#02: Inicializar el estado del reloj • B#16#03: Ajustar la hora e inicializar el estado del reloj
PDT	Input	DT	D, L	Hora predefinida
CORR	Input	INT	I, Q, M, D, L o constante	Valor de corrección (en incrementos de 0,5 h) Valores posibles: de –24 a +26

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SUMMER	Input	BOOL	I, Q, M, D, L	Identificación del horario de verano/ invierno: • 0 = Horario de invierno • 1 = Horario de verano
ANN_1	Input	BOOL	I, Q, M, D, L	Hora de aviso: 1: A la próxima hora en punto se cambiará de horario de verano a horario de invierno o viceversa.
RET_VAL	Return	INT	I, Q, M, D, L	Código de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

Con el parámetro MODE puede definirse si se desea cambiar solo la hora, solo el estado de la hora o ambos valores. Esto se explica en la siguiente tabla:

MODE	Significado
B#16#01	Ajustar la hora Esta llamada equivale a la llamada de la instrucción "SET_CLK". Los parámetros de entrada CORR, SUMMER y ANN_1 no se evalúan.
B#16#02	Inicializar el estado del reloj El parámetro de entrada PDT no se evalúa. A partir de los restantes parámetros de entrada se forman los siguientes elementos del estado del reloj: • Valor de corrección, incluido el signo • Hora de aviso • Indicador del horario de verano/invierno La resolución de hora se ajusta de modo que coincida con la resolución hora de la CPU. El bit de pérdida de sincronización del estado del reloj adquiere el valor FALSE. La hora no se modifica.
B#16#03	Ajustar la hora e inicializar el estado del reloj

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Ningún error
8080	MODE fuera del rango de valores admisible
8081	CORR fuera del rango de valores admisible (solo para MODE = B#16#02 o para MODE = B#16#03)
8082	PDT fuera del rango de valores admisible: Fecha y/u hora no admisibles
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

SNC_RTCB: Sincronizar relojes esclavos

Definición: Sincronizar relojes esclavos

Se entiende por sincronización de relojes esclavos la transferencia de la fecha y la hora desde el reloj maestro de un segmento de bus (p. ej. bus K de S7-400, MPI, bus de fondo de S7) a todos los relojes esclavos de ese mismo segmento de bus.

Descripción

Para garantizar que la hora coincida en todos los módulos de la red, el programa de sistema sincroniza los relojes esclavos a intervalos regulares parametrizables.

Es posible realizar una sincronización, independientemente del intervalo de sincronización parametrizado, llamando la instrucción "SNC_RTCB". En este punto, para todos los relojes esclavos presentes en un segmento de bus, la fecha y la hora se transfieren del reloj maestro al reloj esclavo.

El requisito para una sincronización correcta es que "SNC_RTCB" se ejecute en una CPU cuyo reloj en tiempo real haya sido parametrizado como reloj maestro para al menos un segmento de bus.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SNC_RTCB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No se ha producido ningún error durante la sincronización.
0001	El reloj presente no ha sido parametrizado como reloj maestro para ningún segmento de bus.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Librerías estándar adicionales

Manejo del contador de horas de funcionamiento

Contador de horas de funcionamiento

Introducción

Las CPUs disponen de un número específico de contadores de horas de funcionamiento (ver datos técnicos de las CPUs).

- Si la CPU está provista de contadores de horas de funcionamiento de 16 bits, se pueden usar las instrucciones "SET_RTM (Página 6695)", "CTRL_RTM (Página 6696)" y "READ_RTM (Página 6697)" para inicializar, arrancar, parar y leer los contadores.
- Si la CPU está provista de contadores de horas de funcionamiento de 32 bits, se pueden usar las instrucciones "RTM (Página 6694)" para inicializar, arrancar, parar y leer los contadores.

Nota

Las instrucciones "SET_RTM (Página 6695)", "CTRL_RTM (Página 6696)" y "READ_RTM (Página 6697)" también se pueden utilizar con los contadores de horas de funcionamiento de 32 bits. Sin embargo, en ese caso los contadores de horas de funcionamiento se comportan como si fueran de 16 bits (rango de valores de 0 a 32767 horas).

Consulte también: Registro del extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#000B (Página 6996).

Aplicación

Los contadores de horas de funcionamiento pueden utilizarse para las aplicaciones más diversas:

- Calcular el tiempo de funcionamiento de la CPU
- Calcular el tiempo de funcionamiento de recursos controlados.

Propiedades del contador de horas de funcionamiento

Al arrancarse, el contador de horas de funcionamiento siempre empieza a contar a partir del último valor registrado. Si se desea usar otro valor inicial, debe especificarse explícitamente ("SET_RTM (Página 6695)" o "RTM (Página 6694)" con MODE=4). Cuando la CPU pasa al estado STOP o el usuario para el contador de horas de funcionamiento, la CPU "toma nota" del valor actual del contador. Al efectuar un rearranque completo (en caliente) o un arranque en frío de la CPU, es necesario volver a arrancar el contador de horas de funcionamiento ("CTRL_RTM (Página 6696)" o "RTM (Página 6694)" con MODE=1).

Después de actualizar el sistema operativo, o después de restablecer los valores de fábrica de la CPU, los contadores de horas de funcionamiento se inicializan al valor "0".

Rango de valores

- CPU con contador de horas de funcionamiento de 16 bits: de 0 a 32 767 horas
- CPU con contador de horas de funcionamiento de 32 bits: de 0 a (2 elevado a 31) -1 hora = 2 147 483 647 horas

RTM: Contador de horas de funcionamiento

Descripción

Con la instrucción "RTM" se puede inicializar, arrancar, parar y leer un contador de horas de funcionamiento de 32 bits de la CPU.

Si se desea leer todos los contadores de horas de funcionamiento de 32 bits de la CPU, debe usarse la instrucción "RDSYSST (Página 6963)" con los parámetros SZL_ID=W#16#0132 y:

- INDEX=W#16#000B (para los contadores de horas de funcionamiento de 0 a 7) o bien
- INDEX=W#16#000C (para los contadores de horas de funcionamiento de 8 a 15).

Consulte también: Registro del extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#000B (Página 6996).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RTM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NR	Input	BYTE	I, Q, M, D, L o constante	Número del contador de horas de funcionamiento La numeración empieza por 0. Para conocer el número de contadores de horas de funcionamiento de su CPU, consulte los Datos técnicos.
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición: • 0: Leer (tras ello, el estado se registra en CQ y el valor actual en CV). Si el contador de horas de funcionamiento trabaja durante más de (2 elevado a 31) - 1 horas, se detiene en el último valor representable y envía el aviso de error "Desbordamiento". • 1: Arrancar (con el último valor registrado) • 2: Parar • 4: Inicializar (al valor especificado en PV) • 5: Inicializar (al valor especificado en PV) y arrancar • 6: Inicializar (al valor especificado en PV) y parar
PV	Input	DINT	I, Q, M, D, L o constante	Nuevo valor para el contador de horas de funcionamiento

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
CQ	Output	BOOL	I, Q, M, D, L	Estado del contador de horas de funcionamiento (1: en marcha)
CV	Output	DINT	I, Q, M, D, L	Valor actual del contador de horas de funcionamiento

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Compatibilidad con programas desarrollados para una CPU con contadores de horas de funcionamiento de 16 bits

Las instrucciones "SET_RTM (Página 6695)", "CTRL_RTM (Página 6696)" y "READ_RTM (Página 6697)" también se pueden utilizar con los contadores de horas de funcionamiento de 32 bits. Sin embargo, en ese caso los contadores de horas de funcionamiento de 32 bits se comportan como si fueran de 16 bits (rango de valores de 0 a 32767 horas).

En el extracto de lista parcial con la ID SZL-ID W#16#0132 y el índice W#16#0008 , los contadores de horas de funcionamiento de 32 bits de 0 a 7 aparecen como contadores de horas de funcionamiento de 16 bits. Esto permite seguir utilizando los programas que han sido desarrollados para una CPU con contadores de horas de funcionamiento de 16 bits, y que usan el extracto de lista parcial con la ID SZL-ID W#16#0132 y el índice W#16#0008 .

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Sin error
8080	Número erróneo del contador de horas de funcionamiento
8081	Se ha transmitido un valor negativo al parámetro PV.
8082	Desbordamiento del contador de horas de funcionamiento
8091	El parámetro de entrada MODE contiene un valor no permitido.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

SET_RTM: Inicializar contador de horas de funcionamiento

Descripción

Esta instrucción permite ajustar un contador de horas de funcionamiento de la CPU a un valor predefinido. Se puede configurar un número de contadores de horas de funcionamiento específico para la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_RTM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NR	Input	BYTE	I, Q, M, D, L o constante	La entrada NR contiene el número del contador de horas de funcionamiento que se desea ajustar. La numeración empieza por "0". Para conocer el número de contadores de horas de funcionamiento de su CPU, consulte los Datos técnicos.
PV	Input	INT	I, Q, M, D, L o constante	La entrada PV contiene el ajuste para el contador de horas de funcionamiento.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Sin error
8080	Número erróneo del contador de horas de funcionamiento
8081	Se ha transmitido un valor negativo al parámetro PV.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

CTRL_RTM: Arrancar/parar el contador de horas de funcionamiento

Descripción

Esta instrucción permite arrancar o parar un contador de horas de funcionamiento.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CTRL_RTM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NR	Input	BYTE	I, Q, M, D, L o constante	La entrada NR contiene el número del contador de horas de funcionamiento que se desea arrancar o parar. La numeración empieza por "0". Para conocer el número de contadores de horas de funcionamiento de su CPU, consulte los Datos técnicos.
S	Input	BOOL	I, Q, M, D, L	La entrada S arranca o para el contador de horas de funcionamiento. • Para parar el contador de horas de funcionamiento debe ajustarse el estado lógico a "0". • Para arrancar el contador de horas de funcionamiento debe ajustarse el estado lógico a "1".
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Ningún error
8080	Número erróneo del contador de horas de funcionamiento
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

READ_RTM: Leer contador de horas de funcionamiento

Descripción

Esta instrucción permite leer un contador de horas de funcionamiento. La instrucción devuelve como datos de salida el número actual de horas de funcionamiento y el estado del contador, es decir, "parado" o "contando".

Si el contador de horas de funcionamiento trabaja durante más de 32767 horas, se parará al llegar a 32767 y enviará el aviso de error "Desbordamiento".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "READ_RTM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NR	Input	BYTE	I, Q, M, D, L, P o constante	La entrada NR contiene el número del contador de horas de funcionamiento que se desea arrancar o parar. La numeración empieza por "0". Para conocer el número de contadores de horas de funcionamiento de su CPU, consulte los Datos técnicos.
RET_VAL	Return	INT	I, Q, M, D, L, P	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
CQ	Output	BOOL	I, Q, M, D, L	La salida CQ indica si el contador de horas de funcionamiento está en marcha o detenido. El estado lógico "0" indica que el contador de horas de funcionamiento está parado. El estado lógico "1" indica que el contador de horas de funcionamiento está en marcha.
CV	Output	INT	I, Q, M, D, L, P	La salida CV indica el valor actual del contador de horas de funcionamiento.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Sin errores
8080	Número erróneo del contador de horas de funcionamiento
8081	Desbordamiento del contador de horas de funcionamiento
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Hora local

LOC_TIME: Calcular hora local

Descripción

Esta instrucción lee la hora y el estado de la hora de la CPU, y calcula a partir de ellos la hora local. Por eso solo tiene sentido utilizarla en CPUs con estado de la hora. Esta instrucción puede llamarse en OBs de cualquier clase de prioridad.

Funcionamiento

"LOC_TIME" utiliza internamente la instrucción "RDSYSST (Página 6963)" para leer la hora del módulo, el valor de corrección con signo y el identificador de horario de verano del estado actual de la hora de la CPU. Si se produce un error de "RDSYSST (Página 6963)", se finaliza el bloque. Las indicaciones de error se envían a RET_VAL .

El valor de corrección es el número de medias horas entre la hora base y la hora local. Además se identifica como horario de verano o de invierno.

Si la CPU no posee un estado de hora válido, "LOC_TIME" se finaliza con un error.

Una vez leído correctamente el estado de la hora, la diferencia horaria obtenida a partir del valor de corrección se suma a la hora del módulo y se devuelve en la salida LT .

La salida SUMMER obtiene el valor del bit de horario de verano/invierno a partir del estado de la hora.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LOC_TIME":

Parámetro	Declaración	Tipo de datos	Descripción
RET_VAL	Return	INT	Código de error
LT	Output	DATE_AND_TIME	Hora local (Local Time)
SUMMER	Output	BOOL	Identificador de horario de verano • 0: Invierno • 1: Verano

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros RET_VAL, LT y SUMMER

La tabla siguiente muestra los valores de salida y los errores de "LOC_TIME":

RET_VAL	LT	SUMMER	Descripción
0	Hora local	TRUE/FALSE	Instrucción ejecutada sin errores
1	Hora local	TRUE/FALSE	Sin errores, pero con salto de fecha

8xyy	DT#90-01-01-0:0:0	FALSE	Códigos de error generales de "RDSYSST (Página 6963)"
8091	DT#90-01-01-0:0:0	FALSE	Estado de la hora no presente

BT_LT: Calcular hora local a partir de la hora patrón

Descripción

Esta instrucción calcula la hora local a partir de una hora patrón establecida en la entrada.

La hora patrón introducida en la entrada BT se convierte a la hora local con ayuda de los datos almacenados en un bloque de datos (DB) y se devuelve en la salida LT .

El bloque de datos (DB) contiene el número de unidades de 30 minutos en las que se diferencian la hora patrón y la hora local, así como la diferencia entre el horario de verano y el de invierno, igualmente en unidades de 30 minutos.

Si al efectuar el cálculo se produce un desbordamiento de fecha, esto se señaliza mediante un valor de retorno especial.

La instrucción "BT_LT" puede llamarse en OBs de cualquier clase de prioridad.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "BT_LT":

Parámetro	Declaración	Tipo de datos	Descripción
BT	Input	DATE_AND_TIME	Hora patrón
WS_DAT	Input	BLOCK_DB	Información sobre la zona horaria y sobre la alternancia entre horario de verano y de invierno (DB de reglas); En el parámetro WS_DAT debe usarse un puntero a un bloque de datos del tipo "WS_RULES (Página 6714)".
RET_VAL	Return	INT	Código de error
LT	Output	DATE_AND_TIME	Hora local

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Los parámetros del tipo de datos BLOCK_DB solo se pueden reenviar a llamadas FB y no a llamadas FC.

Parámetros RET_VAL y LT

La tabla siguiente muestra los valores de salida de "BT_LT":

RET_VAL	LT	Descripción
0	Hora local	Instrucción ejecutada sin errores
1	Hora local	Sin errores, pero con salto de fecha
8082	DT#90-01-01-0:0:0	Datos no válidos en el bloque de datos de reglas

LT_BT: Calcular hora patrón a partir de la hora local

Descripción

La instrucción calcula la hora patrón a partir de la hora local establecida en la entrada.

La hora local introducida en la entrada LT se convierte a la hora patrón con ayuda de los datos almacenados en un bloque de datos (DB) y se devuelve en la salida BT.

El bloque de datos (DB) contiene el número de unidades de 30 minutos en las que se diferencian la hora patrón y la hora local, así como la diferencia entre el horario de verano y el de invierno, igualmente en unidades de 30 minutos. Si al efectuar el cálculo se produce un desbordamiento de fecha, esto se señaliza mediante un valor de retorno especial.

Peculiaridades del cambio entre horario de verano e invierno:

- Al pasar del horario de invierno al de verano, se adelanta en una hora la hora local. Esto significa que la hora situada entre ambos horarios no se contabiliza.
  Para acceder a un instante LT situado dentro de esa hora omitida, la instrucción "LT_BT" "piensa" en términos de horario de verano. Esto se notifica con el valor de devolución 4 ó 5.
- Al pasar del horario de verano al de invierno, se atrasa en una hora la hora local. Esto significa que hay una hora que se contabiliza dos veces ("doble hora"). (En CE(S)T se utiliza en ese caso la denominación 2A y 2B).
  En consecuencia, no es posible ubicar de manera unívoca con referencia a la hora patrón un momento LT situado dentro de esa hora.
  "LT_BT" recibe como parámetro de entrada un LT y debe decidir, antes de la conversión a BT, si el valor se encuentra dentro del verano o el invierno. Si LT se encuentra dentro de la doble hora, LT se interpreta como horario de invierno. Esto se notifica con el valor de devolución 2 ó 3.

La instrucción "LT_BT" puede llamarse en OBs de cualquier clase de prioridad.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LT_BT":

Parámetro	Declaración	Tipo de datos	Descripción
LT	Input	DATE_AND_TIME	Hora local
WS_DAT	Input	BLOCK_DB	Información sobre la zona horaria y sobre la alternancia entre horario de verano y de invierno (DB de reglas); En el parámetro WS_DAT debe usarse un puntero a un bloque de datos del tipo "WS_RULES (Página 6714)".
RET_VAL	Return	INT	Código de error
BT	Output	DATE_AND_TIME	Hora patrón

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Los parámetros del tipo de datos BLOCK_DB solo se pueden reenviar a llamadas FB y no a llamadas FC.

Parámetros RET_VAL y BT

La tabla siguiente muestra los valores de salida de "LT_BT":

RET_VAL	BT	Descripción
0	Hora patrón	Instrucción ejecutada sin errores
1	Hora patrón	Sin errores, pero con salto de fecha
2	Hora patrón	La LT de la entrada se encuentra dentro de la "doble hora".
3	Hora patrón	Como 2, y salto de fecha adicional
4	Hora patrón	La LT de la entrada se encuentra dentro de la "hora prohibida".
5	Hora patrón	Como 4, y salto de fecha adicional
8082	DT#90-01-01-0:0:0	Datos no válidos en el bloque de datos de reglas

S_LTINT: Alarma horaria hora local

Descripción

Esta instrucción ajusta la alarma horaria deseada en el momento especificado. Dicho momento se especifica en hora local.

La hora local introducida en la entrada LT se convierte a la hora patrón con ayuda de la regla almacenada en un bloque de datos (DB).

El bloque de datos (DB) contiene el número de unidades de 30 minutos en las que se diferencian la hora patrón y la hora local, así como la diferencia entre el horario de verano y el de invierno, igualmente en unidades de 30 minutos. Con la hora patrón calculada se

parametriza y activa el OB de alarma horaria. Si al efectuar el cálculo se produce un desbordamiento de fecha, esto se señaliza mediante un valor de retorno especial.

Peculiaridades del cambio entre horario de verano e invierno:

- Al pasar del horario de invierno al de verano, se adelanta en una hora la hora local. Esto significa que la hora situada entre ambos horarios no se contabiliza.
  Para acceder a un instante LT dentro de esa hora omitida, la instrucción "S_LTINT" calcula en términos de horario de verano. Esto se notifica con el valor de devolución (RET_VAL) 4 ó 5.
- Al pasar del horario de verano al de invierno, se atrasa en una hora la hora local. Esto significa que hay una hora que se contabiliza dos veces ("doble hora"). (En CE(S)T se utiliza en ese caso la denominación 2A y 2B).
  En consecuencia, no es posible ubicar de manera unívoca con referencia a la hora patrón un momento LT situado dentro de esa hora. "S_LTINT" recibe como parámetro de entrada un LT y debe decidir, antes de la conversión a BT, si el valor se encuentra dentro del verano o el invierno. Si LT se encuentra dentro de la doble hora, LT se interpreta como horario de invierno. Esto se notifica con el valor de devolución (RET_VAL) 2 ó 3.

La instrucción "S_LTINT" puede llamarse en OBs de cualquier clase de prioridad.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "S_LTINT":

Parámetro	Declaración	Tipo de datos	Descripción
OB_NR	Input	INT	Número del OB que se debe iniciar (adm. 10 – 17)
SDT	Input	DATE_AND_TIME	Hora de inicio y hora en horario local Consulte también: "SET_TINT (Página 6887)"
PERIOD	Input	WORD	Período transcurrido desde el punto de salida SDT: • W#16#0000 = Una vez • W#16#0201 = Una vez por minuto • W#16#0401 = Una vez por hora • W#16#1001 = Una vez al día • W#16#1201 = Una vez por semana • W#16#1401 = Una vez al mes • W#16#1801 = Una vez al año • W#16#2001 = A final de mes
WS_DAT	Input	BLOCK_DB	Información sobre la zona horaria y sobre la alternancia entre horario de verano y de invierno (DB de reglas); En el parámetro WS_DAT debe usarse un puntero a un bloque de datos del tipo "WS_RULES (Página 6714)".
RET_VAL	Return	INT	Código de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Los parámetros del tipo de datos BLOCK_DB solo se pueden reenviar a llamadas FB y no a llamadas FC.

Parámetro RET_VAL

RET_VAL	Descripción
0	Instrucción ejecutada sin errores
1	Sin errores, pero con salto de fecha
2	La LT de la entrada se encontraba dentro de la "doble hora".
3	Como 2, y salto de fecha adicional
4	La LT de la entrada se encuentra dentro de la "hora prohibida".
5	Como 4, y salto de fecha adicional
8082	Datos no válidos en el bloque de datos de reglas
8090	Parámetro erróneo OB_NR
8091	Parámetro erróneo SDT
8092	Parámetro erróneo PERIOD
80A1	El momento inicial ajustado se encuentra en el pasado.
80A2	El OB no está cargado
80A3	El OB no se puede iniciar

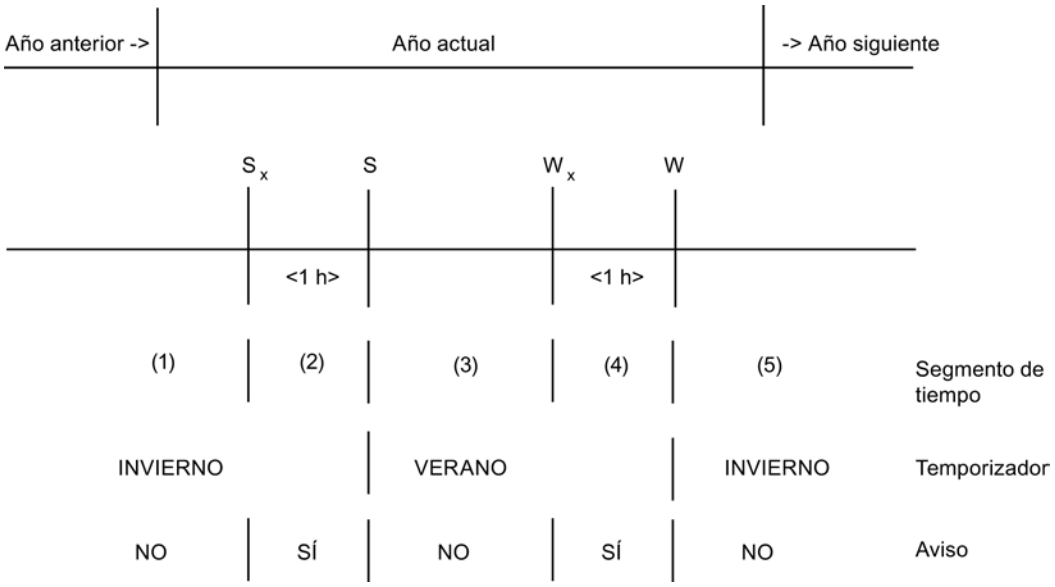
SET_SW: Ajustar horario de invierno/verano sin estado de reloj

Descripción

Esta instrucción permite alternar entre el horario de verano y el de invierno en las CPUs que no disponen de estado de la hora. Para este fin, la instrucción "SET_SW" modifica el ajuste de la hora de la CPU de acuerdo con el momento actual y las reglas de alternancia fijadas en el DB de reglas.

"SET_SW" se llama en el bloque de organización OB1 y en un OB de alarma horaria una sola vez con la misma instancia. En el OB de alarma horaria no se interconectan los parámetros de la instrucción "SET_SW". En la entrada WS_DAT se especifica el bloque de datos (DB) que contiene las reglas para el inicio y el fin del horario de verano.

La figura siguiente muestra el cronograma en el que se basa el funcionamiento de "SET_SW":



Se definen cuatro tiempos de cambio (V_P, V, I_P, I). Su posición exacta se deriva de las reglas almacenadas en el DB de reglas. Las horas de aviso señalizan el cambio de horario una hora antes de que se produzca.

Con flanco positivo en la entrada REQ se inicializa y activa el cambio de horario de verano/invierno. La alternancia definida mantiene su validez mientras se cumpla REQ = 1.

Para el cambio de horario, "SET_SW" parametriza y activa el OB de alarma horaria indicado con el momento de cambio más cercano en el modo "único". Tras ello, este OB de alarma horaria asume el control. Cuando llega el momento, modifica la hora de la CPU conforme al momento de ejecución actual y se parametriza/activa de nuevo a sí mismo usando en cada caso el próximo momento de cambio.

Si la hora actual se encuentra en el apartado (3), el OB de alarma horaria se parametriza con la hora de inicio I_P (aviso de horario de invierno).

Con flanco negativo en la entrada REQ se desactiva el cambio de horario de verano/invierno.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_SW":

Parámetros	Declaración	Tipo de datos	Descripción
REQ	Input	BOOL	Control del cambio
WS_DAT	Input	BLOCK_DB	Información sobre la zona horaria y sobre la alternancia entre horario de verano y de invierno (DB de reglas); En el parámetro WS_DAT debe usarse un puntero a un bloque de datos del tipo "WS_RULES (Página 6714)".
OB_NR	Input	INT	Número de la alarma horaria que se debe utilizar
STATUS	Output	INT	Código de error
ISSUMMER	Output	BOOL	1 = Horario de verano
ANN_1	Output	BOOL	1 = Aviso de cambio

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Los parámetros del tipo de datos BLOCK_DB solo se pueden reenviar a llamadas FB y no a llamadas FC.

Parámetro STATUS

STATUS	Significado
0	Ningún error
8xyy	Códigos de error generales de "RDSYSST (Página 6963)"
8082	Datos no válidos en WS_DAT
8083	El número de mes para el cambio al horario de invierno es mayor que el número de mes para el cambio al horario de verano, y la hora de aviso de invierno es anterior al verano, o el número de mes para el cambio al horario de invierno es menor que el número de mes para el cambio al horario de verano, y la hora de aviso de verano es anterior al invierno.
8090	OB_NR no admisible. (Solo se admiten las alarmas horarias: OB10 – OB17, según cada CPU)
80A2	OB_NR no admisible: El OB no está cargado en la CPU
80A3	No se puede iniciar el OB. Consulte también: "QRY_TINT (Página 6889)"

Campo de aplicación

"SET_SW" puede usarse en las CPUs que no disponen de estado de la hora. Las CPUs con estado de la hora utilizan "SET_SW_S (Página 6707)".

OBs invocantes

"SET_SW" debe llamarse con la misma instancia en los siguientes bloques de organización (OBs):

- **OB1 (programa cíclico):**
  "SET_SW" evalúa la entrada REQ. Con flanco ascendente en REQ ajusta el momento inicial de la alarma horaria indicada, en función del momento actual, a la más cercana de las 4 horas de cambio posibles, y activa la alarma horaria con el identificador 'único'. La alarma horaria se dispara cuando se alcanza dicho momento.
- **OB1x (alarma horaria):**
  El OB1x se ha activado al alcanzarse un momento de cambio de hora. "SET_SW" ajusta la hora de la CPU en función de la hora actual (horario de verano), se parametriza al próximo momento de cambio de hora y lo activa. En el apartado (5) se ajusta al aviso de horario de verano (S_A) del año siguiente.

"SET_SW" no debe instanciarse varias veces.

Las siguientes instrucciones permiten consultar, parametrizar, activar y borrar la alarma horaria: "QRY_TINT (Página 6889)", "SET_TINT (Página 6887)", "ACT_TINT (Página 6889)", "CAN_TINT (Página 6888)"

Nota

El cambio de hora de la CPU sin medidas adicionales, p. ej. a través de la programadora, con "SET_SW" (REQ = 1) activo, solo se autoriza si la nueva hora ajustada no está situada más allá de uno de los momentos de cambio de hora.

Si al ajustar el reloj se salta uno de los momentos de cambio de hora, antes del ajuste debe desactivarse la alternancia entre horario de verano e invierno con REQ = 0 y, una vez realizada la modificación, volver a activarse con REQ = 1.

Por supuesto, al ajustar el reloj de la CPU en verano debe tenerse en cuenta también la hora del horario de verano.

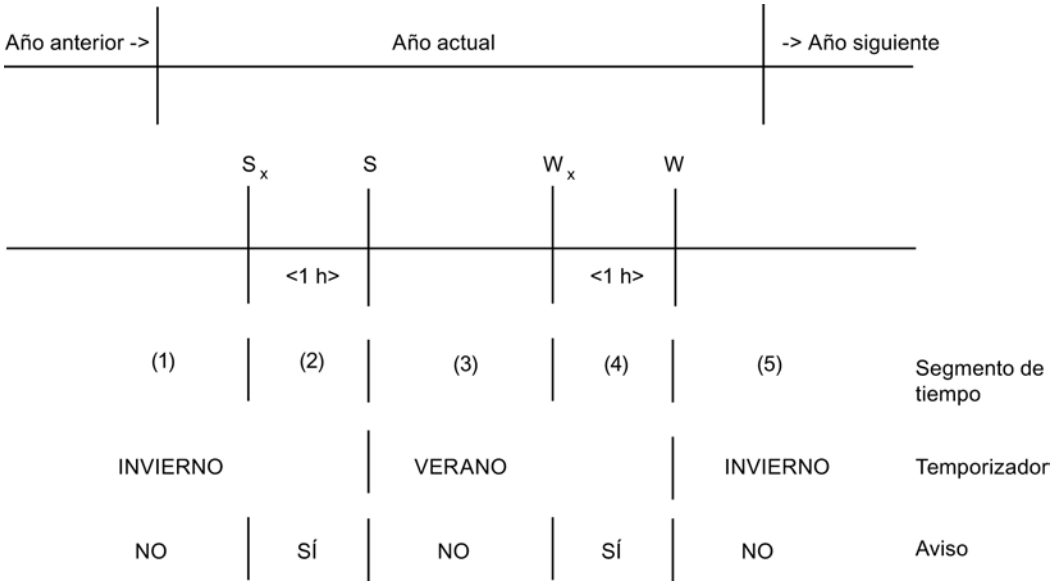
SET_SW_S: Ajustar horario de verano/invierno sin estado de reloj

Descripción

Esta instrucción permite alternar entre el horario de verano y el de invierno en las CPUs que disponen de estado de la hora. Para este fin, la instrucción "SET_SW_S" modifica el estado de la hora de acuerdo con el momento actual y las reglas de alternancia fijadas en el DB de reglas.

"SET_SW_S" se llama en el OB1 y en un OB de alarma horaria una sola vez con la misma instancia. En el OB de alarma horaria no se interconectan los parámetros de la instrucción "SET_SW_S". En la entrada WS_DAT se especifica el bloque de datos (DB) que contiene las reglas para el inicio y el fin del horario de verano.

La figura siguiente muestra el cronograma en el que se basa el funcionamiento de "SET_SW_S":



Se definen cuatro tiempos de cambio (V_P, V, I_P, I). Su posición exacta se deriva de las reglas almacenadas en el DB de reglas. Las horas de aviso señalizan el cambio de horario 1 hora antes de que se produzca

Con flanco positivo en la entrada REQ se inicializa y activa el cambio de horario de verano/invierno. La alternancia definida mantiene su validez mientras se cumpla REQ = 1.

Para el cambio de horario, "SET_SW_S" parametriza y activa el OB de alarma horaria indicado con el momento de cambio más cercano en el modo "único". Tras ello, este OB de alarma horaria asume el control. Cuando llega el momento, modifica la hora de la CPU conforme al momento de ejecución actual y se parametriza/activa de nuevo a sí mismo usando en cada caso el próximo momento de cambio.

Ejemplo: Si la hora actual se encuentra en el rango (3), el OB de alarma horaria se parametriza con la hora de inicio WA (aviso de horario de invierno).

Con flanco negativo en la entrada REQ se desactiva el cambio de horario de verano/invierno.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_SW_S":

Parámetro	Declaración	Tipo de datos	Descripción
REQ	Input	BOOL	Control del cambio
WS_DAT	Input	BLOCK_DB	Información sobre la zona horaria y sobre la alternancia entre horario de verano y de invierno (DB de reglas); En el parámetro WS_DAT debe usarse un puntero a un bloque de datos del tipo "WS_RULES (Página 6714)".
OB_NR	Input	INT	Número de la alarma horaria que se debe utilizar
STATUS	Output	INT	Código de error
ISSUMMER	Output	BOOL	Es verano
ANN_1	Output	BOOL	Aviso de cambio

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Los parámetros del tipo de datos BLOCK_DB solo se pueden reenviar a llamadas FB y no a llamadas FC.

Parámetro STATUS

STATUS	Significado
0	Ningún error
8xyy	Códigos de error generales de "RDSYSST (Página 6963)"
8082	Datos no válidos en WS_DAT
8083	El número de mes para el cambio al horario de invierno es mayor que el número de mes para el cambio al horario de verano, y la hora de aviso de invierno es anterior al verano, o el número de mes para el cambio al horario de invierno es menor que el número de mes para el cambio al horario de verano, y la hora de aviso de verano es anterior al invierno.
8090	OB_NR no admisible. (Solo se admiten las alarmas horarias: OB10 – OB17, según cada CPU)

STATUS	Significado
8091	Estado de la hora no válido ("status_valid" en el estado de la hora tenía el valor FALSE)
80A2	OB_NR no admisible: El OB no está cargado en la CPU
80A3	No se puede iniciar el OB. (ver "QRY_TINT (Página 6889)")

Campo de aplicación

"SET_SW_S" puede usarse en las CPUs que disponen de estado de la hora.

OBs invocantes

"SET_SW_S" debe llamarse con la misma instancia en los siguientes bloques de organización (OBs):

- **OB1 (programa cíclico):**
  "SET_SW_S" evalúa la entrada REQ. Con flanco ascendente en REQ ajusta el momento inicial de la alarma horaria indicada, en función del momento actual, a la más cercana de las 4 horas de cambio posibles, y activa la alarma horaria con el identificador 'único'. Así, la alarma horaria se dispara cuando se alcanza dicho momento.
- **OB1x (alarma horaria):**
  El OB1x se ha activado al alcanzarse un momento de cambio de hora. "SET_SW_S" ajusta la hora y el estado de hora de la CPU en función de la hora actual, se parametriza al próximo momento de cambio de hora y lo activa. En el rango (5) se ajusta al aviso de horario de verano del año siguiente.

"SET_SW_S" no debe instanciarse varias veces.

Las siguientes instrucciones permiten consultar, parametrizar, activar y borrar la alarma horaria: "QRY_TINT (Página 6889)", "SET_TINT (Página 6887)", "ACT_TINT (Página 6889)", "CAN_TINT (Página 6888)"

Nota

El cambio de hora de la CPU sin medidas adicionales, p. ej. a través de la programadora, con "SET_SW_S" (REQ = 1) activo, solo se autoriza si la nueva hora ajustada no está situada más allá de uno de los momentos de cambio de hora.

Si al ajustar el reloj se salta uno de los momentos de cambio de hora, antes del ajuste debe desactivarse la alternancia entre horario de verano e invierno con REQ = 0 y, una vez realizada la modificación, volver a activarse con REQ = 1 .

Al ajustar el reloj de la CPU y el estado de la hora es necesario activar el identificador de horario de verano o el identificador de hora de aviso, dependiendo de si se desea ajustar el horario de verano o el horario de invierno.

TIMESTMP: Transferir avisos con sello de tiempo

Descripción

Esta instrucción transfiere avisos con sello de tiempo de un IM153-2 a su DB de instancia. Éste suministra los datos para el posterior procesamiento a cargo del programa de usuario.

- IM153-2
  IM153-2 detecta los cambios de las señales binarias, les asigna un sello de tiempo y guarda esta información en registros con un máximo de 20 avisos. En el caso de determinados eventos, emite avisos especiales. Cuando un registro está listo para ser recogido, IM153-2 dispara una alarma de proceso.
- Llamada en el OB de alarma de proceso
  "TIMESTMP" lee la información de arranque del OB de alarma de proceso y guarda la información relativa al sello de tiempo para su posterior procesamiento en un búfer circular. "TIMESTMP" puede almacenar los datos de 17 alarmas de proceso como máximo.
- Llamada en el OB cíclico:
  En el procesamiento cíclico, "TIMESTMP" lee el registro que se debe recoger con ayuda de la instrucción "RD_REC (Página 6769)" (Leer registro) y lo guarda en el búfer de avisos MESSAGE. Si deben recogerse varios registros, se lee primero el más antiguo. Si el bit TIMECONV está activado en la entrada, "TIMESTMP" convierte todos los sellos de tiempo del formato ISP al formato DATE_AND_TIME (DT).
  Si el parámetro BUFRDY = TRUE está activado, el programa de usuario puede continuar procesando los avisos del búfer de avisos y transferirlos p. ej. a un receptor (equipo de manejo y visualización, impresora). El número de mensajes existentes se visualiza en el parámetro MSG_QTY. Después del procesamiento, el programa de usuario debe desactivar el parámetro BUFRDY con objeto de liberar el búfer de avisos para otros avisos. El parámetro BUFNOTREAD muestra el número de alarmas de proceso recibidas para las que "TIMESTMP" no ha leído aún registros del IM153-2 . Con BUFNOTREAD = 15 , todos los registros del IM153-2 están llenos. Entonces el IM153-2 no detecta otros cambios de señal. Cualquier otro registro que se genere en caso de un cambio de señal binaria por el IM 153-2 se pierde. El parámetro MSGLOST (pérdida de avisos) se pone a "1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TIMESTMP":

Parámetros	Declaración	Tipo de datos	Descripción
LADDR	Input	INT	Dirección lógica (dirección de diagnóstico) del IM153-2
LADDR2	Input	INT	Dirección lógica (dirección de diagnóstico) del slot 2 del IM153-2
TIMECONV	Input	BOOL	1=convertir el sello de tiempo ISP al formato S7-DT
MSG_QTY	Output	INT	Número de avisos válidos en el búfer de avisos
BUFNOTREAD	Output	INT	Número de registros del IM153-2con alarmas pero no leídos todavía
READERR	Output	BOOL	1=Error al leer el IM153-2
READSTATUS	Output	INT	Valor de retorno de la instrucción "RD_REC (Página 6769)"
BUFRDY	InOut	BOOL	1=Búfer de avisos listo

Parámetros	Declaración	Tipo de datos	Descripción
MSGLOST	InOut	BOOL	1=Avisos perdidos
MESSAGE	Static	ARRAY[1..20] OF STRUCT	Búfer de avisos para un máximo de 20 avisos del IM153-2
SLOT_NO	./.	BYTE	Número de slot/2=Aviso especial
CH_NO	./.	BYTE	Número de canal/ID del aviso especial
SIGNAL	./.	BYTE	Estado lógico/característica del aviso especial
TIME1	./.	DWORD	Segundos a partir de 1900,0 (ISP)/año, mes, día, hora (DT)
TIME2	./.	DWORD	Fracciones de segundo (ISP)/minuto, segundo, milisegundo, día de la semana (DT)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Configuración hardware

La instrucción "TIMESTMP" puede utilizarse solo en configuraciones que soporten la transmisión de la hora y el sellado de tiempo de señales binarias. A saber:

- CPUs
  - CPU 4xx con CP443-5 para la transmisión de telegramas horarios (a partir de 6GK7443-5DX02-0XE0), o
  - CPU 4xx con transmisión de telegramas horarios (p. ej. CPU416-2 DP a partir de la versión de firmware 3.0)
- Periferia descentralizada
  - Módulo de interfaz IM153-2 apto para redundancia con sellado de tiempo (a partir de 6ES7153-2AA02 versión de firmware V1.2.3 (versión 10))
    O bien:
    IM153-2 FO apto para redundancia con interfaz FO integrada (a partir de 6ES7153-2AB01 a partir de la versión de firmware V1.2.3 (versión 9))
  - Entrada digital SM321 (6ES7321-7BH00-0AB0) con aislamiento galvánico 16 DI, 24 V DC, alarma de proceso, diagnóstico
    O bien:
    Entrada digital SM321 (6ES7321-7TH00-0AB0) con aislamiento galvánico 16 DI, 24 V DC, NAMUR, con funciones de control de procesos
    El sello de tiempo sólo se debe utilizar para señales binarias seleccionadas e importantes para la aplicación. Debido a la carga del PROFIBUS-DP y del IM153-2, se recomienda repartir estas señales, en caso necesario, entre varios IM153-2 o entre varias estaciones.

Configuración de software

- OBs invocantes
  "TIMESTMP" debe llamarse con la misma instancia en los siguientes bloques de organización (OBs), teniendo en cuenta que solo es posible transferir y recibir valores de los parámetros en el programa cíclico:
  - OB1 (programa cíclico) o, como alternativa, un OB de alarma cíclica (OB30 a OB38)
  - OB86 (fallo del rack)
  - OB100 (arranque)
    Si se utilizan otros OBs de arranque (OB 101, OB 102), "TIMESTMP" debe llamarse también en ellos.
  - OB40 o, si es posible seleccionarlo en la configuración de hardware, también otro OB de alarma de proceso (OB41 a OB47)
- Direccionamiento
  - Utilice para cada IM153-2 una instancia independiente de "TIMESTMP". Introduzca en la entrada LADDR la dirección lógica ajustada en la configuración hardware (dirección de diagnóstico) del IM153-2.
  - Si el maestro DP se utiliza en el modo de operación "Compatible con S7", introduzca en la entrada LADDR2 el mismo valor que para LADDR. Si se utiliza el modo de operación "DPV1", introduzca en LADDR2 la dirección de diagnóstico del slot 2 del IM153-2.
  - Si las direcciones se modifican durante el funcionamiento, "TIMESTMP" borra los datos de alarma de proceso almacenados hasta entonces.

Tratamiento de errores

Si al leer un registro "TIMESTMP" ha detectado errores no eliminables, "TIMESTMP" ajusta el parámetro READERR a TRUE. Para el análisis posterior se dispone del valor de retorno de la instrucción WR_USMSG en el parámetro READSTATUS . Dado que "TIMESTMP" lee también otro registro en el ciclo siguiente, READERR y READSTATUS solo son válidos para un ciclo. Por ello debe incluirse en el programa de usuario la correspondiente evaluación.

Si se leen registros mientras el maestro DP está fuera de servicio, "TIMESTMP" emite el aviso READERR = TRUE y READSTATUS =80B2h ("El slot configurado no está ocupado.")

Comportamiento de arranque IM153-2

En el arranque/rearranque/arranque en frío, IM153-2 vuelve a emitir las alarmas de proceso para los registros que antes del arranque estaban ocupados pero todavía no se han recogido.

IM153-2 registra los siguientes avisos en el primer registro libre:

- Aviso especial "Inicio datos de arranque"
- Cambios de señal ocurridos inmediatamente antes de un STOP de la CPU ("entrante" o "saliente" según el flanco parametrizado)
- Estado lógico actual de las señales binarias que deben obtener un sello de tiempo ("entrante" o "saliente" según el flanco parametrizado)
- Aviso especial "Fin datos de arranque"

Comportamiento de arranque de TIMESTMP

Si "TIMESTMP" se llama en un OB de arranque (OB100, OB101, OB102), "TIMESTMP" borra todos los datos de alarmas de proceso almacenados y restablece BUFRDY . Si se detectan registros con alarmas pero todavía no leídos, "TIMESTMP" ajusta MSGLOST (pérdida de aviso) a TRUE. El programa de usuario debe volver a restablecer MSGLOST .

Comportamiento de "TIMESTMP" en el OB86 (fallo de módulo)

Si se llama "TIMESTMP" en el OB86 y el evento procede del correspondiente IM153-2, "TIMESTMP" reacciona igual que en el arranque. Esto es lo que sucede p. ej. en caso de fallo y recuperación del IM153-2 y recuperación del maestro DP.

Redundancia

En los sistemas H con dos IM153-2, existe redundancia en cuanto al sellado de tiempo en los siguientes casos:

- La comunicación entre los dos IM153-2 se efectúa mediante bus K.
- La sincronización de los IM153-2 activos y pasivos ha finalizado correctamente.

Durante una conmutación entre IM153-2 activos y pasivos, se ha interrumpido el sellado de tiempo. Con el aviso especial "Conmutación con redundancia inicio/fin" se muestra el período de interrupción.

Normalmente, el IM153-2 activo notifica al pasivo el estado actual de la periferia. Si esta comunicación no funciona, se emite el aviso especial "Pérdida de información de redundancia entrante". En cuanto se puede recuperar la comunicación entre los IM153-2 activo y pasivo, se emite el aviso especial "Pérdida de información de redundancia saliente". No es posible compensar a posteriori la pérdida, es decir: al pasar el IM153-2 al estado de fallo, se produce una pérdida de aviso.

Estructura del búfer de avisos

El búfer de avisos contiene un registro del IM153-2. La interpretación del sello de tiempo depende del ajuste del parámetro TIMECONV .

Formatos de tiempo de "TIMESTMP"

Si la entrada es TIMECONV = TRUE , todos los sellos de tiempo se guardan en el tipo de datos DATE_AND_TIME (formato BCD):

La tabla siguiente muestra el formato del tipo de datos DATE_AND_TIME:

Bytes	Contenido	Área
0	Año	90 … 89 Corresponde a 1990 ... 2089
1	Mes	01 ... 12
2	Día	01 ... 31
3	Hora	00 ... 23

Bytes	Contenido	Área
4	Minuto	00 ... 59
5	Segundo	00 ... 59
6	2 MSD (most significant decade) de ms	00 ... 99
7 (4 MSB)	LSD (least significant decade) de ms	0 ... 9
7 (4 LSB)	Día de la semana	1 ... 7 (1 = domingo)

Si la entrada es TIMECONV = FALSE , todos los sellos de tiempo se guardan en el formato ISP (2 palabras dobles):

La tabla siguiente muestra la distribución de los bytes:

Bytes	Contenido	Área
0 … 3	Segundos a partir de 1.1.1900 0:00:00,000 h	Corresponde a 1.1.1900 ... 6.2.2036
4 … 7	Fracciones de segundo en múltiplos de $1/2^{32}$ s	0 … <1

La conversión horaria es válida dentro del rango de fechas comunes a los formatos horarios, es decir, desde el 1-1-1990 hasta el 6-2-2036 (inclusive).

WS_RULES: DB de regla para convertir la hora base en hora local

Bloque de datos del tipo "WS_RULES"

Un bloque de datos del tipo "WS_RULES" tiene la siguiente estructura:

Nombre		Tipo de datos	Valor inicial	Descripción
// Conversión hora base<->Hora local y "Ajustar alarma según hora local"				
B2L		Struct		
	S	INT	2	// Diferencia hora base->hora local [30 m] en invierno, admisible -24 ... +24.
	T	INT	2	// Diferencia horario invierno y verano [30 m], admisible: 2
// Regla para: Horario invierno -> Horario verano. Valor predeterminado: Último domingo de marzo, 2:00 h				
W2S		Struct		// en HORARIO INVIERNO
	M	BYTE	B#16#3	// Mes del cambio
	W	BYTE	B#16#9	// n aparición del día de la semana (1=primera, 2=segunda,..,9=última)
	D	BYTE	B#16#1	// Día de la semana (domingo = 1)
	H	BYTE	B#16#2	// Hora
// Regla para: Horario verano -> Horario invierno. Valor predeterminado: Último domingo de octubre, 3:00 h				
S2W				// en HORARIO VERANO
	M	BYTE	B#16#10	// Mes del cambio
	W	BYTE	B#16#9	// n aparición del día de la semana (9 = última)

Nombre		Tipo de datos	Valor inicial	Descripción
	D	BYTE	B#16#1	// Día de la semana (domingo = 1)
	H	BYTE	B#16#3	// Hora

Nota

Crear bloque de datos del tipo "WS_RULES"

Un bloque de datos del tipo "WS_RULES" se crea seleccionando como tipo "WS_RULES" al crear un DB nuevo.

Nota

Todos los parámetros que tienen el formato BYTE se interpretan como valores BCD (decimal en código binario). Para el horario de invierno debe indicarse W2S y para el de verano S2W. En la UE está prescrita desde el año 2002 la fijación de los momentos de cambio entre horario de verano e invierno mediante una regla.

String + Char

S_COMP: Comparar cadenas de caracteres

Descripción

Esta instrucción compara los contenidos de dos variables en el formato STRING y devuelve el resultado de la comparación como valor de retorno. Las variables que se deben comparar se interconectan en las entradas IN1 y IN2 . A los parámetros de entrada solo pueden asignárseles variables definidas como simbólicas.

Seleccione la condición de comparación mediante el cuadro de instrucción. Si se cumple la condición de comparación (p. ej. mayor o igual), el estado lógico del parámetro de salida OUT se ajusta a "1".

Pueden usarse las siguientes condiciones de comparación:

Símbolo	Descripción
EQ	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es igual que la del parámetro IN2 .
NE	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es distinta a la del parámetro IN2 .
GT (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es mayor que la del parámetro IN2 .
LT (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es menor que la del parámetro IN2 .
GE (1)	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es mayor o igual que la del parámetro IN2 .

Símbolo	Descripción
LE [1]	El valor de retorno tiene el estado lógico "1" si la cadena de caracteres del parámetro IN1 es menor o igual que la del parámetro IN2 .
[1] Los caracteres se comparan empezando por la izquierda según su codificación ASCII (p. ej. 'a' es mayor que 'A'). El primer carácter diferente es el decisivo para el resultado de la comparación. Si la parte izquierda de la cadena de caracteres más larga es idéntica a la cadena de caracteres más corta, la cadena de caracteres más larga se considera mayor.	

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "S_COMP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING*	D, L	Variable de entrada en el formato STRING
IN2	Input	STRING*	D, L	Variable de entrada en el formato STRING
OUT	Output	BOOL	I, Q, M, D, L	Resultado de la comparación
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

S_CONV: Convertir cadena de caracteres

Descripción

Esta instrucción permite convertir el valor de la entrada IN al formato de datos indicado en la salida OUT . Mediante esta instrucción se convierten cadenas de caracteres en números enteros o en coma flotante, así como números enteros o en coma flotante en cadenas de caracteres.

Convertir una cadena de caracteres en números enteros o en coma flotante

Son posibles las siguientes conversiones de cadena de caracteres en números enteros o en coma flotante:

- Convertir una cadena de caracteres en una variable en el formato INT. El primer carácter de la cadena de caracteres puede ser un signo o una cifra, mientras que el resto deben ser necesariamente cifras. Si la longitud de la cadena de caracteres es igual a cero o mayor que 6, o la cadena contiene caracteres no autorizados, la conversión no se realiza y el bit BR se ajusta a "0". Si el resultado de la conversión se encuentra fuera del rango numérico INT, se limita el resultado al valor correspondiente y el resultado binario BR se ajusta a "0".
- Convertir una cadena de caracteres en una variable en el formato DINT. El primer carácter de la cadena de caracteres puede ser un signo o una cifra, mientras que el resto deben ser necesariamente cifras. Si la longitud de la cadena de caracteres es igual a cero o mayor que 11, o la cadena contiene caracteres no autorizados, la conversión no se realiza y el bit BR se ajusta a "0". Si el resultado de la conversión se encuentra fuera del rango numérico DINT, se limita el resultado al valor correspondiente y el resultado binario BR se ajusta a "0".
- Convertir una cadena de caracteres en una variable en el formato REAL. La cadena de caracteres debe tener el siguiente formato: ±v.nnnnnnnE±xx
  - ± = signo
  - v = 1 posición antes de la coma
  - n = 7 posiciones después de la coma
  - x = 2 posiciones de exponente

  Si la longitud de la cadena de caracteres es menor que 14, o la cadena de caracteres tiene un formato distinto al indicado arriba, la conversión no se realiza y el bit BR se ajusta a "0". Si el resultado de la conversión se encuentra fuera del rango numérico REAL, se limita el resultado al valor correspondiente y el resultado binario BR se ajusta a "0".

Convertir un número entero o en coma flotante en una cadena de caracteres

Son posibles las siguientes conversiones de números enteros o en coma flotante en cadenas de caracteres:

- Convertir una variable en el formato INT en una cadena de caracteres. La cadena de caracteres se representa precedida de signo. Si la variable indicada en el parámetro de retorno es demasiado corta, la conversión no se realiza y el bit BR se ajusta a "0". Al parámetro de salida solo puede asignársele una variable definida como simbólica.
- Convertir una variable en el formato DINT en una cadena de caracteres. La cadena de caracteres se representa precedida de signo. Si la variable indicada en el parámetro de retorno es demasiado corta, la conversión no se realiza y el bit BR se ajusta a "0". Al parámetro de salida solo puede asignársele una variable definida como simbólica.
- Convertir una variable en el formato REAL en una cadena de caracteres. La cadena de caracteres se representa con 14 posiciones: ±v.nnnnnnnE±xx
  - ± = signo
  - v = 1 posición antes de la coma
  - n = 7 posiciones después de la coma
  - x = 2 posiciones de exponente

  Si la variable indicada en el parámetro de retorno es demasiado corta o el parámetro IN no contiene un número en coma flotante válido, la conversión no se realiza y el bit BR se ajusta a "0". Al parámetro de salida solo puede asignársele una variable definida como simbólica.

Parámetros

Las tablas siguientes muestran los parámetros de la instrucción "S_CONV":

Tabla 9-175 Parámetros para convertir una cadena de caracteres en números enteros o en coma flotante

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING*	D, L	Valor que se va a convertir
OUT	Output	INT, DINT, REAL	I, Q, M, D, L	Resultado de la conversión.
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Tabla 9-176 Parámetros para convertir un número entero o en coma flotante en una cadena de caracteres

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	INT, DINT, REAL	I, Q, M, D, L o constante	Valor que se va a convertir
OUT	Output	STRING*	D, L	Resultado de la conversión
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

ATH: Convertir cadena de caracteres ASCII en número hexadecimal

Descripción

Esta instrucción permite convertir la cadena de caracteres ASCII indicada en la entrada IN en un número hexadecimal. El resultado de la conversión se deposita en la dirección especificada en el parámetro OUT .

- El número de caracteres ASCII que se deben convertir se define con el parámetro N.
- Como máximo se pueden convertir 65 535 caracteres ASCII válidos.
- Solo pueden interpretarse las cifras de 0 a 9 y las letras mayúsculas de A a F. Todos los demás caracteres se convierten a ceros.

Dado que un carácter ASCII requiere 8 bits y un número hexadecimal solo 4, la palabra de salida es la mitad de larga que la palabra de entrada. Después de la conversión, los caracteres ASCII se ordenan a la salida en el mismo orden en el que se han leído. Si se trata de un número impar de caracteres ASCII, el número hexadecimal correspondiente al semibyte a la derecha del último número hexadecimal convertido se rellena con ceros.

Cuando se detecta un carácter ASCII no válido, se convierte como "0" y en el parámetro RET_VAL se deposita un mensaje de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ATH":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	POINTER	I, Q, M, D, L	Puntero a la dirección inicial de una cadena de caracteres ASCII
N	Input	WORD	I, Q, M, D, L, P o constante	Número de caracteres ASCII que se deben convertir
RET_VAL	Output	WORD	I, Q, M, D, L, P	Estado de la instrucción
OUT	Output	POINTER	I, Q, M, D, L	Puntero a la dirección del resultado
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#....)	Descripción
0000	Sin error
0007	Carácter no válido

Ejemplo

Con estado lógico de la entrada I 0.0 = "1" (activado), se ejecuta la instrucción. El parámetro de entrada N = 5 indica que se deben convertir cinco caracteres ASCII. Los caracteres ASCII se guardan en el bloque de datos 1, que empieza en la dirección indicada por el puntero IN: DB1.DBX10.0. La cadena de caracteres se devuelve en la dirección indicada por el puntero OUT: Inicio en DB2.DBX0.0 (bloque de datos 2). Al tratarse en este caso de un número impar de caracteres ASCII, la última cifra hexadecimal contiene solo ceros en el semibyte derecho, lo que da lugar al valor hexadecimal 0xC0.

Si la instrucción se ha ejecutado sin errores, RET_VAL se ajusta al valor W#16#0000.

La tabla siguiente muestra ejemplos de conversión de cadenas de caracteres ASCII en números hexadecimales.

IN	N	OUT	Estado BR
'0123'	4	16#0123	1
'123AFx1a23'	10	16#123AF01023	0

Tabla 9-177 La tabla siguiente muestra los caracteres ASCII y los correspondientes valores hexadecimales.

Caracteres ASCII	Valor hexadecimal codificado en ASCII	Número hexadecimal
"0"	30	0
"1"	31	1
"2"	32	2
"3"	33	3
"4"	34	4
"5"	35	5
"6"	36	6
"7"	37	7
"8"	38	8
"9"	39	9
"A"	41	A
"B"	42	B
"C"	43	C
D	44	D
E	45	E
F	46	F

HTA: Convertir número hexadecimal en cadena de caracteres ASCII

Descripción

Esta instrucción permite convertir el número hexadecimal indicado en la entrada IN en una cadena de caracteres ASCII. El resultado de la conversión se deposita en la dirección especificada en el parámetro OUT .

El número de bytes hexadecimales que se deben convertir se define con el parámetro N. Dado que un carácter ASCII requiere 8 bits y un número hexadecimal solo 4, el valor de salida es el doble de largo que el valor de entrada. Cada semibyte del número hexadecimal se convierte en un carácter manteniendo el orden original.

En la cadena de caracteres ASCII pueden escribirse como máximo 65635 caracteres. El resultado de la conversión se representa con las cifras de 0 a 9 y las letras mayúsculas de A a F.

La instrucción no detecta condiciones de error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "HTA":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	POINTER*	I, Q, M, D	Dirección inicial de la cifra hexadecimal
N	Input	WORD	I, Q, M, D, L, P o constante	Número de bytes hexadecimales que se deben convertir
OUT	Output	POINTER*	I, Q, M, D, L	Dirección en la que se guarda el resultado.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Ejemplo

Con estado lógico de la entrada I 0.0 = "1" (activado), se ejecuta la instrucción. El parámetro de entrada N = 3 indica que se deben convertir tres caracteres hexadecimales. Los bytes hexadecimales se guardan en el bloque de datos 1, que empieza en la dirección indicada por el puntero IN : DB1.DBX10.0. La cadena de caracteres se devuelve en la dirección indicada por el puntero OUT : Inicio en DB2.DBX0.0 (bloque de datos 2).

La tabla siguiente muestra ejemplos de conversión de números hexadecimales en cadenas de caracteres ASCII:

IN	N	OUT	Estado BR
W#16#0123	2	'0123'	1
16#123AF01023	4	'123AF010'	0

La tabla siguiente muestra los caracteres ASCII y los correspondientes valores hexadecimales:

Número hexadecimal	Valor hexadecimal codificado en ASCII	Caracteres ASCII
0	30	"0"
1	31	"1"
2	32	"2"
3	33	"3"
4	34	"4"
5	35	"5"
6	36	"6"
7	37	"7"
8	38	"8"
9	39	"9"
A	41	"A"
B	42	"B"
C	43	"C"
D	44	"D"
E	45	"E"
F	46	"F"

Otras instrucciones

LEN: Determinar la longitud de una cadena de caracteres

Descripción

Esta instrucción devuelve la longitud de una cadena de caracteres (número de caracteres válidos) como valor de retorno.

Una variable STRING contiene dos longitudes:

- La longitud máxima (se especifica entre corchetes en la definición de la variable)
- La longitud actual (número de caracteres válidos en el momento actual)

La longitud actual es menor o igual a la longitud máxima. El número de bytes ocupados por una cadena de caracteres excede en 2 la longitud máxima. Una cadena vacía (") tiene la longitud cero. La longitud máxima es de 254. La instrucción no notifica ningún error.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LEN":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING*	D, L* *	Variable de entrada en el formato STRING; al parámetro de entrada solo puede asignársele una variable definida como simbólica.
OUT	Return	INT	I, Q, M, D, L	Número actual de caracteres
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos). ** En SCL: también es posible utilizar constantes.				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

CONCAT: Concatenar

Descripción

Esta instrucción agrupa dos variables STRING en una sola cadena de caracteres. Si la cadena de caracteres resultante es más larga que la variable definida en el parámetro de salida, se limita la cadena resultante a la longitud máxima ajustada y el bit BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CONCAT": A los parámetros solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING*	D, L	Variable de entrada en el formato STRING
IN2	Input	STRING*	D, L	Variable de entrada en el formato STRING
OUT	Return	STRING*	D, L	Cadena concatenada
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

LEFT: Leer los caracteres izquierdos de una cadena

Descripción

Esta instrucción devuelve los primeros L caracteres de una cadena de caracteres.

- Si L es mayor que la longitud actual de la variable STRING, se devuelve el valor de entrada.
- Si L = 0 y si el valor de entrada es una cadena vacía, se devuelve una cadena vacía.
- Si L es negativo, se devuelve una cadena vacía y el bit BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LEFT". Al parámetro IN y al valor de retorno RET_VAL solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING*	D, L	Variable de entrada en el formato STRING
L	Input	INT	I, Q, M, D, L o constante	Longitud de la cadena de caracteres izquierda
OUT	Return	STRING*	D, L	Variable de salida en el formato STRING
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

RIGHT: Leer los caracteres derechos de una cadena

Descripción

Esta instrucción devuelve los últimos L caracteres de una cadena de caracteres.

- Si L es mayor que la longitud actual de la variable STRING, se devuelve el valor de entrada.
- Si L es negativo, se devuelve una cadena vacía y el resultado binario BR se ajusta a "0".
- Si L = 0 y si el valor de entrada es una cadena vacía, se devuelve una cadena vacía.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RIGHT". Al parámetro IN y al valor de retorno RET_VAL solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING*	D, L	Variable de entrada en el formato STRING
L	Input	INT	I, Q, M, D, L o constante	Longitud de la cadena de caracteres derecha
OUT	Return	STRING*	D, L	Variable de salida en el formato STRING
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

MID: Leer los caracteres centrales de una cadena

Descripción

Esta instrucción devuelve la parte central de una cadena de caracteres (L caracteres a partir del carácter P incluido).

- Si la suma de L y (P-1) supera la longitud actual de la variable STRING, se devuelve una cadena de caracteres a partir del carácter P hasta el final del valor de entrada.
- En todos los demás casos (si P se encuentra fuera de la longitud actual, P y/o L es igual a cero o negativo), se devuelve una cadena vacía y el bit BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MID". Al parámetro IN y al valor de retorno RET_VAL solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING*	D, L	Variable de entrada en el formato STRING
L	Input	INT	I, Q, M, D, L o constante	Longitud de la cadena de caracteres central
P	Input	INT	I, Q, M, D, L o constante	Posición del primer carácter
OUT	Return	STRING*	D, L	Variable de salida en el formato STRING
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

DELETE: Borrar caracteres de una cadena

Descripción

Esta instrucción borra en una cadena de caracteres L caracteres a partir del carácter P incluido).

- Si L y/o P es igual a cero o P es mayor que la longitud actual de la cadena de caracteres de entrada, se devuelve la cadena de caracteres de entrada.
- Si la suma de L y P es mayor que la cadena de caracteres de entrada, se borra hasta el final de la cadena de caracteres.
- Si L y/o P son negativos, se devuelve una cadena vacía y el bit BR se ajusta a "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DELETE". Al parámetro de entrada IN y al parámetro de salida RET_VAL solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN	Input	STRING*	D, L	Variable STRING en la que se efectúa el borrado
L	Input	INT	I, Q, M, D, L o constante	Número de caracteres que se van a borrar
P	Input	INT	I, Q, M, D, L o constante	Posición del primer carácter que se va a borrar
OUT	Return	STRING*	D, L	Cadena de caracteres resultante
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

INSERT: Insertar caracteres en una cadena

Descripción

Esta instrucción inserta la cadena de caracteres del parámetro IN2 en la cadena de caracteres del parámetro IN1 después del carácter P .

- Si P es igual a cero, la segunda cadena de caracteres se inserta antes de la primera.
- Si P es mayor que la longitud actual de la primera cadena de caracteres, la segunda cadena de caracteres se inserta después de la primera.
- Si P es negativo, se devuelve una cadena vacía y el bit BR se ajusta a "0".

El bit BR también se ajusta a "0" si la cadena de caracteres resultante es más larga que la variable especificada en el parámetro de salida; en este caso, la cadena de caracteres resultante se limita a la longitud máxima ajustada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "INSERT". A los parámetros de entrada IN1 y IN2 y al parámetro de salida solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING*	D, L	Variable STRING en la que se efectúa la inserción
IN2	Input	STRING*	D, L	Variable STRING que se va a insertar
P	Input	INT	I, Q, M, D, L o constante	Posición de inserción
OUT	Return	STRING*	D, L	Cadena de caracteres resultante
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

REPLACE: Reemplazar caracteres de una cadena

Descripción

Esta instrucción sustituye L caracteres de la primera cadena de caracteres (IN1) a partir del carácter P (incluido) por la totalidad de la segunda cadena de caracteres (IN2).

- Si L es igual a cero y P distinto de cero, se devuelve la primera cadena de caracteres.
- Si L es igual a cero y P igual a cero, se coloca la segunda cadena de caracteres delante de la primera.
- Si L es distinto de cero y P igual a cero o uno, se sustituyen los caracteres a partir del primer carácter.

- Si P está fuera de la primera cadena de caracteres, la segunda cadena de caracteres se coloca después de la primera.
- Si L y/o P son negativos, se devuelve una cadena vacía y el bit BR se ajusta a "0". El bit BR también se ajusta a "0" si la cadena de caracteres resultante es más larga que la variable especificada en el parámetro de salida; en este caso, la cadena de caracteres resultante se limita a la longitud máxima ajustada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "REPLACE". A los parámetros de entrada IN1 y IN2 y al parámetro de salida solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING*	D, L	Variable STRING en la que se efectúa la sustitución
IN2	Input	STRING*	D, L	Variable STRING que se va a reemplazar
L	Input	INT	I, Q, M, D, L o constante	Número de caracteres que se van a reemplazar
P	Input	INT	I, Q, M, D, L o constante	Posición del primer carácter reemplazado
OUT	Return	STRING*	D, L	Cadena de caracteres resultante
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

FIND: Buscar caracteres en una cadena

Descripción

Esta instrucción devuelve la posición de la segunda cadena de caracteres (IN2) dentro de la primera cadena de caracteres (IN1). La búsqueda se inicia por la izquierda. Se notifica la primera aparición de la cadena de caracteres.

Si la segunda cadena de caracteres no existe dentro de la primera, se devuelve cero. La instrucción no notifica errores.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "FIND". A los parámetros de entrada IN1 y IN2 solo pueden asignárseles variables definidas como simbólicas.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN1	Input	STRING*	D, L	Variable STRING en la que se efectúa la búsqueda
IN2	Input	STRING*	D, L	Variable STRING que se va a buscar
OUT	Return	INT	I, Q, M, D, L	Posición de la cadena de caracteres encontrada
* Defina la longitud máxima de la cadena de caracteres si en la declaración de la interfaz se utiliza el tipo de datos STRING para una variable temporal (encontrará más información en la descripción del tipo de datos).				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Memoria imagen de proceso

UPDAT_PI: Actualizar memoria imagen de proceso de las entradas

Descripción

Esta instrucción permite actualizar la memoria imagen de proceso del OB 1 (=memoria imagen parcial de proceso 0) de las entradas o una memoria imagen parcial de proceso de las entradas definida mediante configuración.

Si se ha configurado como método de notificación para la actualización de la memoria imagen de proceso por el sistema la notificación repetida de todos los errores de acceso a la periferia, la actualización de la memoria imagen de proceso seleccionada se lleva a cabo de modo continuo.

De lo contrario, dicha actualización solo se efectúa cuando la memoria imagen parcial de proceso seleccionada no es actualizada por el sistema, es decir,

- si no se ha asignado esa memoria imagen parcial de proceso a ningún OB de alarma, o
- si se ha seleccionado la memoria imagen parcial de proceso 0 y se ha desactivado (en la configuración) la actualización de la memoria imagen parcial de proceso del OB 1.

Nota

Las direcciones lógicas que se han asignado en la configuración a una memoria imagen parcial de proceso de las entradas, no pertenecen ya a la memoria imagen de proceso del OB 1 de las entradas.
Una memoria imagen parcial de proceso que se actualiza con "UPDAT_PI" no debe actualizarse al mismo tiempo con la instrucción "SYNC_PI (Página 6732)".

Si se ha asignado a un OB de alarma la actualización por el sistema de la memoria imagen de proceso del OB 1 de las entradas y de la memoria imagen parcial de proceso de las entradas, la actualización se realiza independientemente de las llamadas de "UPDAT_PI".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "UPDAT_PI":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PART	Input	BYTE	I, Q, M, D, L o constante	Número de la memoria imagen parcial de proceso de las entradas que se debe actualizar. Rango máximo de valores (el rango depende de la CPU): De 0 a 15 (0 significa memoria imagen de proceso del OB 1, n para 1 < n < 15 significa memoria imagen parcial de proceso n).
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
FLADDR	Output	WORD	I, Q, M, D, L	Dirección del primer byte que provoca errores, en caso de que se haya producido un error de acceso

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	Valor no válido en el parámetro PART
8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU.
8092	La memoria imagen parcial de proceso es actualizada por el sistema con un OB, y el usuario no ha configurado para ello la notificación repetida de todos los errores de acceso a la periferia. No se ha realizado ninguna actualización con "UPDAT_PI".
80A0	Se ha detectado un error de acceso al acceder a la periferia.
8xyy	Información de error general, ver Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Nota

Si se utiliza esta instrucción para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPRD_DAT (Página 6776)".

UPDAT_PO: Actualizar memoria imagen de proceso de las salidas

Descripción

Esta instrucción permite transferir a los módulos de salida los estados lógicos de la memoria imagen de proceso del OB 1 (=memoria imagen parcial de proceso 0) de las salidas o de una memoria imagen parcial de proceso de las salidas definida mediante configuración.

Si se ha definido un área de coherencia para la memoria imagen parcial de proceso seleccionada, los datos en cuestión se transfieren de modo coherente al módulo de periferia correspondiente.

Nota

Las direcciones lógicas que se han asignado mediante configuración a una memoria imagen parcial de proceso de las salidas, no pertenecen ya a la memoria imagen de proceso del OB 1 de las salidas.
Las salidas que se actualizan con "UPDAT_PO" no deben actualizarse al mismo tiempo con la instrucción "SYNC_PO (Página 6734)".

Si se ha asignado a un OB de alarma la transferencia a los módulos de salida de la memoria imagen de proceso del OB 1 de las salidas y de la memoria imagen parcial de proceso de las salidas, el sistema realiza la transferencia independientemente de las llamadas de "UPDAT_PO".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "UPDAT_PO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PART	Input	BYTE	I, Q, M, D, L o constante	Número de la memoria imagen parcial de proceso de las salidas que se debe transferir. Rango máximo de valores (el rango depende de la CPU): de 0 a 15. (0 significa memoria imagen de proceso del OB 1, n para 1 < n < 15 significa memoria imagen parcial de proceso n).
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
FLADDR	Output	WORD	I, Q, M, D, L	Dirección del primer byte que provoca errores, en caso de que se haya producido un error de acceso

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	Valor no válido en el parámetro PART
8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU.
80A0	Se ha detectado un error de acceso al acceder a la periferia.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Nota

Si se utiliza esta instrucción para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPWR_DAT (Página 6778)".

SYNC_PI: Sincronizar memoria imagen de proceso de las entradas

Descripción

La instrucción "SYNC_PI" permite actualizar de modo isócrono una memoria imagen parcial de proceso de las entradas. Con esta instrucción, un programa de usuario vinculado a un ciclo DP o a un ciclo de emisión PN puede actualizar los datos de entrada registrados de modo isócrono y coherente en una memoria imagen parcial de proceso de las entradas.

Llamada

"SYNC_PI" puede interrumpirse, y solo puede llamarse en los OBs 61, 62, 63 y 64.

Nota

La llamada de la instrucción "SYNC_PI" en los OBs 61 a 64 solo está autorizada si en la configuración hardware se ha asignado la memoria imagen parcial de proceso en cuestión al OB correspondiente.

Una memoria imagen parcial de proceso que se actualiza con "SYNC_PI" no puede actualizarse al mismo tiempo con la instrucción "UPDAT_PI (Página 6729)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SYNC_PI":

Parámetro	Declaración	Tipo de datos	Área de memoria	Rango de valores	Descripción
PART	Input	BYTE	I, Q, M, D, L o constante	de 1 a 30	Número de la memoria imagen parcial de proceso de las entradas que se debe actualizar en modo isócrono.
RET_VAL	Return	INT	I, Q, M, D, L	-	Información de error
FLADDR	Output	WORD	I, Q, M, D, L	-	Dirección del primer byte que provoca errores, en caso de que se haya producido un error de acceso.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Clase de evento código de error	Explicación
W#16#0001	Advertencia de coherencia. La actualización de la memoria imagen parcial de proceso se ha repartido entre dos ciclos DP o PN. Sin embargo, los datos pertenecientes a un esclavo o un dispositivo IO se han transferido de modo coherente.
W#16#8090	Valor no admisible en el parámetro PART, o bien la actualización de la memoria imagen parcial de proceso de las entradas indicada no está permitida en este OB. No se ha actualizado la memoria imagen parcial de proceso de las entradas.
W#16#8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU. No se ha actualizado la memoria imagen parcial de proceso de las entradas.
W#16#80A0	Durante la actualización se ha detectado un error de acceso. Las entradas afectadas se han ajustado a "0".
W#16#80A1	El momento de actualización es posterior a la ventana de acceso admisible. No se ha actualizado la memoria imagen parcial de proceso de las entradas. El ciclo DP o PN es demasiado corto y no garantiza suficiente tiempo para procesar la instrucción. Por ello deben incrementarse los tiempos TDP (también conocidos como T_DC), Ti y To.
W#16#80A2	Error de acceso con advertencia de coherencia Durante la actualización de la memoria imagen parcial de proceso de las entradas se ha detectado un error de acceso con advertencia de coherencia simultánea. • Los datos de las entradas erróneas no han sido leídos todavía por la periferia. En la memoria imagen parcial de proceso de las entradas las entradas afectadas se ponen a cero. • La actualización de los datos de entrada no afectados por el error de acceso se ha distribuido entre dos ciclos DP o PN.
W#16#80C1	El momento de actualización es anterior a la ventana de acceso admisible. No se ha actualizado la memoria imagen parcial de proceso de las entradas.
W#16#8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Nota

Si se utiliza la instrucción "SYNC_PI" para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPRD_DAT (Página 6776)".

SYNC_PO: Sincronizar memoria imagen de proceso de las salidas

Descripción

La instrucción "SYNC_PO" permite actualizar de modo isócrono una memoria imagen parcial de proceso de las salidas. Con esta instrucción, un programa de usuario vinculado a un ciclo DP o a un ciclo de emisión PN puede transferir a la periferia de modo isócrono y coherente los datos de salida calculados de una memoria imagen parcial de proceso de las salidas.

Llamada

"SYNC_PO" puede interrumpirse, y solo puede llamarse en los OBs 61, 62, 63 y 64.

Nota

La llamada de la instrucción "SYNC_PO" en los OBs 61 a 64 solo está autorizada si en la configuración hardware se ha asignado la memoria imagen parcial de proceso en cuestión al OB correspondiente.

Una memoria imagen parcial de proceso que se actualiza con "SYNC_PO" no puede actualizarse al mismo tiempo con la instrucción "UPDAT_PO (Página 6731)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SYNC_PO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Rango de valores	Descripción
PART	Input	BYTE	I, Q, M, D, L o constante	de 1 a 30	Número de la memoria imagen parcial de proceso de las salidas que se debe actualizar en modo isócrono.
RET_VAL	Return	INT	I, Q, M, D, L	-	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
FLADDR	Output	WORD	I, Q, M, D, L	-	Dirección del primer byte que provoca errores.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Clase de evento código de error	Explicación
W#16#0001	Advertencia de coherencia. La actualización de la memoria imagen parcial de proceso se ha repartido entre dos ciclos DP o PN. Sin embargo, los datos pertenecientes a un esclavo o un dispositivo IO se han transferido de modo coherente.
W#16#8090	Valor no admisible en el parámetro PART, o bien la actualización de la memoria imagen parcial de proceso de las salidas indicada no está permitida en este OB. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
W#16#8091	La memoria imagen parcial de proceso especificada todavía no está definida o no se encuentra dentro del área de memoria imagen de proceso admisible de la CPU. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
W#16#80A0	Durante la actualización de la memoria imagen parcial de proceso de las salidas se ha detectado un error de acceso. No se han transferido las salidas erróneas a la periferia. En la memoria imagen parcial de proceso de las salidas, éstas no sufren modificación alguna.
W#16#80A1	Error de acceso con advertencia de coherencia Durante la actualización de la memoria imagen parcial de proceso de las salidas se ha detectado un error de acceso con advertencia de coherencia simultánea. • No se han transferido los datos de las salidas erróneas a la periferia. En la memoria imagen parcial de proceso de las salidas, las salidas afectadas no sufren modificación alguna. • La actualización de los datos de salida no afectados por el error de acceso se ha distribuido entre dos ciclos DP o PN.
W#16#80A2	El momento de actualización es posterior a la ventana de acceso admisible. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
W#16#80C1	El momento de actualización es anterior a la ventana de acceso admisible. No se han transferido las salidas a la periferia. La memoria imagen parcial de proceso de las salidas no se modifica.
W#16#8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Nota

Si se utiliza la instrucción "SYNC_PO" para memorias imagen parcial de proceso de esclavos DP normalizados para los que se hayan definido áreas de coherencia de más de 32 bytes, también son posibles los códigos de error de la instrucción "DPWR_DAT (Página 6778)".

Periferia descentralizada

RDREC: Leer registro

Descripción

Esta instrucción permite leer el registro con número INDEX del componente direccionado con ID . Puede tratarse de un módulo central o de un componente descentralizado (PROFIBUS DP o PROFINET IO).

Con MLEN se indica el máximo de bytes que se deben leer. Por ello debe escogerse un área de destino RECORD con una longitud de por lo menos MLEN bytes.

El valor TRUE del parámetro de salida VALID indica que el registro se ha transferido correctamente al área de destino RECORD . En ese caso, el parámetro de salida LEN contiene la longitud en bytes de los datos leídos.

Si se ha producido un error durante la transferencia del registro, se señaliza mediante el parámetro de salida ERROR . En ese caso, el parámetro de salida STATUS contiene la información de error.

Nota

Si un esclavo DPV1 está configurado mediante archivo GSD (a partir de la rev. 3 de GSD) y la interfaz DP del maestro DP está ajustada como "Compatible con S7", los módulos de E/S no pueden leer en el programa de usuario los registros con "RDREC". En este caso, el maestro DP direcciona un slot incorrecto (slot configurado + 3).

Solución: Conmutar la interfaz del maestro DP a "DPV1".

Nota

La interfaz de la instrucción "RDREC" es idéntica a la del FB "RDREC" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Funcionamiento

"RDREC" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia del registro, debe llamarse RDREC con REQ = 1.

Mediante el parámetro de salida BUSY y los dos bytes centrales del parámetro de salida STATUS , se muestra el estado de la petición. Los dos bytes centrales de STATUS corresponden al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

La transferencia del registro se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RDREC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Realizar la transferencia del registro
ID	Input	DWORD	I, Q, M, D, L o constante	Dirección lógica del esclavo DP o componente PROFINET IO (módulo) • Para un módulo de salida, debe activarse el bit 15 (ejemplo para la dirección 5: ID:=DW#16#8005). • Con un módulo mixto debe indicarse la dirección más baja de las dos.
INDEX	Input	INT	I, Q, M, D, L o constante	Número de registro
MLEN	Input	INT	I, Q, M, D, L o constante	Longitud máxima en bytes de la información de registro que se debe leer
VALID	Output	BOOL	I, Q, M, D, L	El nuevo registro se ha recibido y es válido
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La lectura todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: Se ha producido un error durante la lectura.
STATUS	Output	DWORD	I, Q, M, D, L	Estado del bloque o información de error Para la interpretación del parámetro STATUS , ver el parámetro STATUS (Página 6748).
LEN	Output	INT	I, Q, M, D, L	Longitud de la información del registro leída
RECORD	InOut	ANY	I, Q, M, D, L	Área de destino para el registro leído Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se utiliza la instrucción para leer un registro con PROFINET IO, los valores negativos de los parámetros INDEX, MLEN y LEN se interpretan como número entero de 16 bits sin signo.

WRREC: Escribir registro

Descripción

Esta instrucción permite transferir el registro RECORD al componente direccionado con ID. Puede tratarse de un módulo central o de un componente descentralizado (PROFIBUS DP o PROFINET IO).

Con LEN (oculto) se indica la longitud en bytes del registro que se debe transferir. Por ello debe escogerse un área de origen RECORD con una longitud de por lo menos LEN bytes.

El valor TRUE del parámetro de salida DONE indica que el registro se ha transferido correctamente.

Si se ha producido un error durante la transferencia del registro, se señaliza mediante el parámetro de salida ERROR . En ese caso, el parámetro de salida STATUS contiene la información de error.

Nota

Si un esclavo DPV1 está configurado mediante archivo GSD (a partir de la rev. 3 de GSD) y la interfaz DP del maestro DP está ajustada como "Compatible con S7", no se pueden escribir en el programa de usuario registros con "WRREC" en los módulos de E/S. En este caso, el maestro DP direcciona un slot incorrecto (slot configurado + 3).

Solución: Conmutar la interfaz del maestro DP a "DPV1".

Nota

La interfaz de la instrucción "WRREC" es idéntica a la de la instrucción "WRREC" definida en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Funcionamiento

"WRREC" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia del registro, debe llamarse "WRREC" con REQ = 1.

Mediante el parámetro de salida BUSY y los dos bytes centrales del parámetro de salida STATUS , se muestra el estado de la petición. Los dos bytes centrales de STATUS corresponden al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Asegúrese de asignar siempre el mismo valor al parámetro actual de RECORD en todas las llamadas de "WRREC" pertenecientes a una misma petición. Lo mismo se aplica a los parámetros actuales de LEN.

La transferencia del registro se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WRREC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Realizar la transferencia del registro
ID	Input	DWORD	I, Q, M, D, L o constante	Dirección lógica del esclavo DP o componente PROFINET IO (módulo) Para un módulo de salida, debe activarse el bit 15 (ejemplo para la dirección 5: ID:=DW#16#8005). Con un módulo mixto debe indicarse la dirección más baja de las dos.
INDEX	Input	INT	I, Q, M, D, L o constante	Número de registro
LEN	Input	INT	I, Q, M, D, L o constante	(oculto) Longitud máxima en bytes del registro que se debe transferir
DONE	Output	BOOL	I, Q, M, D, L	Se ha transferido el registro
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La escritura todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: Se ha producido un error durante la escritura.
STATUS	Output	DWORD	I, Q, M, D, L	Estado del bloque o información de error Para la interpretación del parámetro STATUS , ver el parámetro STATUS (Página 6748).
RECORD	InOut	ANY	I, Q, M, D, L	Registro Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se utiliza la instrucción para escribir un registro con PROFINET IO, los valores negativos de los parámetros INDEX y LEN se interpretan como número entero de 16 bits sin signo.

GETIO: Leer memoria imagen de proceso

Descripción

Con esta instrucción se leen de modo coherente todas las entradas de un esclavo DP normalizado o un dispositivo PROFINET IO. Para ello, esta instrucción llama la instrucción "DPRD_DAT (Página 6776)". Si no se ha producido ningún error durante la transferencia de datos, los datos leídos se registran en el área de destino delimitada mediante INPUTS .

El área de destino debe tener la misma longitud que se haya configurado para el componente seleccionado.

En un esclavo DP normalizado con estructura modular o con varias IDs de DP, la llamada de "GETIO" solo permite acceder cada vez a los datos de un solo componente o ID de DP en la dirección configurada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GETIO":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	DWORD	I, Q, M, D, L o constante	• low word: Dirección lógica del esclavo DP o componente PROFINET IO (módulo) • high word: Irrelevante
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "DPRD_DAT (Página 6776)" en la forma DW#16#40xxxx00
LEN	Output	INT	I, Q, M, D, L	Número de datos leídos, en bytes
INPUTS	InOut	ANY	I, Q, M, D	Área de destino para los datos leídos. Debe tener la misma longitud que se haya configurado para el esclavo DP o componente PROFINET-IO seleccionado. Solo se admite el tipo de datos BYTE .

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Consulte también: DPRD_DAT: Leer datos coherentes de un esclavo DP normalizado (Página 6776).

SETIO: Transferir memoria imagen de proceso

Descripción

Con la instrucción "SETIO" se transfieren de modo coherente los datos del área de origen delimitada por OUTPUTS al esclavo DP normalizado o dispositivo PROFINET IO y, en su caso, a la memoria imagen de proceso (concretamente si se ha configurado el área de direcciones en cuestión del esclavo DP normalizado como área de coherencia en una memoria imagen de proceso). "Para ello, SETIO llama la instrucción DPWR_DAT (Página 6778)".

El área de origen debe tener la misma longitud que se haya configurado para el componente seleccionado.

En un esclavo DP normalizado con estructura modular o con varias IDs de DP, la llamada de "SETIO" permite acceder a los datos de un solo componente o ID de DP.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SETIO":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	DWORD	I, Q, M, D, L o constante	• low word: Dirección lógica del esclavo DP o componente PROFINET IO (módulo) • high word: Irrelevante
LEN	Input	INT	I, Q, M, D, L	Irrelevante
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "DPWR_DAT (Página 6778)" en la forma DW#16#40xxxx00
OUTPUTS	InOut	ANY	I, Q, M, D	Área de origen para los datos que se deben escribir. Debe tener la misma longitud que se haya configurado para el esclavo DP o componente PROFINET-IO seleccionado. Solo se admite el tipo de datos BYTE.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Consulte también: DPWR_DAT: Escribir datos coherentes de un esclavo DP normalizado (Página 6778).

GETIO_PART: Leer área de memoria imagen de proceso

Descripción

Con la instrucción "GETIO_PART" se lee de modo coherente una parte del área de la memoria imagen de proceso perteneciente a un esclavo DP normalizado o a un dispositivo PROFINET IO. "Para ello, GETIO_PART llama la instrucción "UBLKMOV".

Nota

Al OB en el que se llame "GETIO_PART" debe asignársele una memoria imagen parcial de proceso de las entradas. Además, antes de la llamada de "GETIO_PART" es necesario incluir el correspondiente esclavo DP normalizado o dispositivo PROFINET IO en esa memoria imagen parcial de proceso de las entradas. Si la CPU no es capaz de trabajar con memorias imagen parcial de proceso o se desea llamar "GETIO_PART" en el OB 1, antes de la llamada de "GETIO_PART" debe incluirse el correspondiente esclavo DP normalizado o dispositivo PROFINET IO en la memoria imagen de proceso de las entradas.

Con los parámetros OFFSET y LEN se define la porción que se debe leer del área de la memoria imagen de proceso del componente direccionado mediante ID.

- Si no se ha producido ningún error durante la transferencia de datos, ERROR adquiere el valor FALSE, y los datos leídos se registran en el área de destino delimitada mediante INPUTS.
- Si se ha producido algún error durante la transferencia de datos, ERROR adquiere el valor TRUE, y STATUS recibe la información de error de "UBLKMOV".
- Si el área de destino (parámetro INPUTS) es menor que LEN, se transfieren todos los bytes que INPUTS pueda contener. ERROR adquiere el valor FALSE. Si el área de destino es mayor que LEN, se escriben los primeros LEN bytes del área de destino. ERROR adquiere el valor FALSE.

Nota

"GETIO_PART" no comprueba en la memoria imagen de proceso de las entradas los límites entre los datos que pertenecen a los distintos componentes PROFIBUS-DP o PROFINET-IO. Por eso el usuario debe asegurarse de que el área de memoria imagen de proceso definida mediante OFFSET y LEN pertenezca a un componente. La lectura de varios componentes al mismo tiempo puede no ser compatible con sistemas futuros y dificultar la migrabilidad a sistemas de otros fabricantes.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GETIO_PART":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	DWORD	I, Q, M, D, L o constante	• low word: Dirección lógica del esclavo DP o componente PROFINET IO (módulo) • high word: Irrelevante
OFFSET	Input	INT	I, Q, M, D, L o constante	Número del primer byte que se debe leer en la memoria imagen de proceso del componente (valor más pequeño posible: 0)
LEN	Input	INT	I, Q, M, D o constante	Número de caracteres que se van a leer
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "UBLKMOV" en la forma DW#16#40xxxx00, si ERROR = TRUE.
ERROR	Output	BOOL	I, Q, M, D, L	Visualización de error: ERROR = TRUE, en caso de que se produzca un error al llamar "UBLKMOV".
INPUTS	InOut	ANY	I, Q, M, D	Área de destino para los datos leídos: • Si el área de destino es menor que LEN, se transfieren todos los bytes que INPUTS pueda contener. ERROR adquiere el valor FALSE. • Si el área de destino es mayor que LEN, se escriben los primeros LEN bytes del área de destino. ERROR adquiere el valor FALSE.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros STATUS y ERROR

Véase la instrucción "UBLKMOV".

SETIO_PART: Transferir área de memoria imagen de proceso

Descripción

Con la instrucción "SETIO_PART" se transfieren de modo coherente los datos del área de origen delimitada por OUTPUTS a una parte del área de memoria imagen de proceso

perteneciente a un esclavo DP normalizado o a un dispositivo PROFINET IO. "Para ello, SETIO_PART llama la instrucción "UBLKMOV".

Nota

Debe asignarse una memoria imagen parcial de proceso de las salidas al OB en el que se llama "SETIO_PART". Además, antes de la llamada de "SETIO_PART" es necesario incluir el correspondiente esclavo DP normalizado o dispositivo PROFINET IO en esa memoria imagen parcial de proceso de las entradas. Si la CPU no es capaz de trabajar con memorias imagen parcial de proceso o se desea llamar "SETIO_PART" en el OB 1, antes de la llamada a "SETIO_PART" debe incluirse el correspondiente esclavo DP normalizado o dispositivo PROFINET IO en la memoria imagen de proceso de las salidas.

Con los parámetros OFFSET y LEN se define la porción que se debe escribir del área de la memoria imagen de proceso del componente direccionado mediante ID.

- Si no se ha producido ningún error durante la transferencia de datos, ERROR adquiere el valor FALSE.
- Si se ha producido algún error durante la transferencia de datos, ERROR adquiere el valor TRUE, y STATUS recibe la información de error de "UBLKMOV".
- Si el área de origen (parámetro OUTPUTS) es menor que LEN, se transfieren todos los bytes que OUTPUTS contiene. ERROR adquiere el valor FALSE. Si el área de origen es mayor que LEN, se transfieren los primerosLEN bytes de OUTPUTS. ERROR adquiere el valor FALSE.

Nota

"SETIO_PART" no comprueba en la memoria imagen de proceso de las salidas los límites entre los datos que pertenecen a los distintos componentes PROFIBUS-DP o PROFINET-IO. Por eso el usuario debe asegurarse de que el área de memoria imagen de proceso definida mediante OFFSET y LEN pertenezca a un componente. La escritura de varios componentes al mismo tiempo puede no ser compatible con sistemas futuros y dificultar la migrabilidad a sistemas de otros fabricantes.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SETIO_PART":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
ID	Input	DWORD	I, Q, M, D, L o constante	• low word: Dirección lógica del esclavo DP o componente PROFINET IO (módulo) • high word: Irrelevante
OFFSET	Input	INT	I, Q, M, D, L o constante	Número del primer byte que se debe escribir en la memoria imagen de proceso del componente (valor más pequeño posible: 0)
LEN	Input	INT	I, Q, M, D, L o constante	Número de caracteres que se van a escribir

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	DWORD	I, Q, M, D, L	Contiene la información de error de "UBLKMOV" en la forma DW#16#40xxxx00, si ERROR = TRUE.
ERROR	Output	BOOL	I, Q, M, D, L	Visualización de error: ERROR = TRUE, en caso de que se produzca un error al llamar UBLKMOV.
OUTPUTS	InOut	ANY	I, Q, M, D	Área de origen para los datos que se deben escribir: • Si el área de origen es menor que LEN, se transfieren todos los bytes que OUTPUTS contiene. ERROR adquiere el valor FALSE. • Si el área de origen es mayor que LEN, se transfieren los primerosLEN bytes de OUTPUTS. ERROR adquiere el valor FALSE.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros STATUS y ERROR

Véase la instrucción "UBLKMOV".

RALRM: Recibir alarma

Descripción de RALRM

Descripción

La instrucción recibe una alarma, junto con la información correspondiente, desde un módulo de periferia (configuración centralizada) o desde un componente de un esclavo DP o un dispositivo PROFINET IO, y proporciona dicha información a través de sus parámetros de salida.

Los parámetros de salida ofrecen tanto la información de arranque del OB llamado como información procedente del origen de la alarma.

Esta instrucción debe llamarse solo dentro del OB de alarma que el sistema operativo de la CPU ha iniciado debido a la alarma procedente de la periferia que se debe investigar.

Nota

Si se llama la instrucción en un OB cuyo evento de arranque no es una alarma procedente de la periferia, la instrucción ofrecerá al usuario menos información en sus salidas.

Si llama la instrucción en distintos OBS, asegúrese de utilizar distintos DBs de instancia. Si se van a evaluar los datos resultantes de una llamada de "RALRM" fuera del OB de alarma correspondiente, es necesario incluso utilizar un DB de instancia independiente por cada evento de arranque de OB.

Nota

La interfaz de la instrucción "RALRM" es idéntica a la del FB "RALRM" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Llamada de "RALRM"

Esta instrucción se puede llamar en tres modos de operación (MODE) distintos, que se describen en la tabla siguiente.

MODE	RALRM ...
0	... Muestra el componente que ha disparado la alarma en el parámetro de salida ID y en el parámetro de salida NEW escribe TRUE.
1	... Escribe en todos los parámetros de salida, sea cual sea el componente desencadenante de la alarma.
2	... Comprueba si el componente especificado en el parámetro de entrada F_ID ha desencadenado la alarma. • En caso negativo, NEW adquiere el valor FALSE. • En caso afirmativo, NEW adquiere el valor TRUE, y se escriben todos los demás parámetros de salida.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RALRM":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	Modo de operación
F_ID	Input	DWORD	I, Q, M, D, L o constante	Dirección lógica inicial del componente (módulo) del que se deben recibir alarmas
MLEN	Input	INT	I, Q, M, D, L o constante	Longitud máxima en bytes de la información de alarmas que se desea recibir
NEW	Output	BOOL	I, Q, M, D, L	Se ha recibido una nueva alarma.

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS (Página 6748)	Output	DWORD	I, Q, M, D, L	Código de error de la instrucción o del maestro DP
ID	Output	DWORD	I, Q, M, D, L	Dirección lógica inicial del componente (módulo) del que se ha recibido una alarma El bit 15 contiene la identificación de E/S: 0 si es una dirección de entrada y 1 si es de salida.
LEN	Output	INT	I, Q, M, D, L	Longitud de la información de alarma recibida
TINFO (Página 6752)	InOut	ANY	I, Q, M, D, L	(task information) Área de destino para la información de arranque del OB y la información administrativa Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro TINFO siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
AINFO (Página 6755)	InOut	ANY	I, Q, M, D, L	(alarm information) Área de destino para información de encabezado e información adicional de alarma Para AINFO debe reservarse una longitud de por lo menos MLEN bytes. Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro AINFO siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se escoge un área de destino TINFO o AINFO demasiado corta, la instrucción no puede registrar toda la información.

Véase también: Área de destino TINFO y AINFO (Página 6763).

Parámetro STATUS

Estructura del parámetro STATUS

El parámetro de salida STATUS contiene información de error. Si se interpreta como ARRAY[1...4] OF BYTE , la información de error presenta la siguiente estructura:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	• B#16#00, si no hay ningún error • ID de función de DPV1-PDU: En caso de error, se devuelve B#16#80 (al leer registro: B#16#DE, al escribir registro: B#16#DF). Si no se utiliza ningún elemento de protocolo de DPV1: B#16#C0.
STATUS[2]	Error_Decode	Ubicación de la identificación de error
STATUS[3]	Error_Code_1	Identificación de error
STATUS[4]	Error_Code_2	Extensión específica de fabricante de la identificación de error

Elemento de campo STATUS[2]

STATUS[2] puede adoptar los siguientes valores:

Error_Decode (B#16#...)	Origen	Significado
00 a 7F	CPU	Ningún error ni advertencia
80	DPV1	Error según IEC 61158-6
81 a 8F	CPU	B#16#8x indica un error en el x parámetro de llamada de la instrucción.
FE, FF	DP Profile	Error específico de perfil

Elemento de campo STATUS[3]

STATUS[3] puede adoptar los siguientes valores:

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
00	00		Ningún error ni advertencia
70	00	reserved, reject	Primera llamada; no hay ninguna transferencia de registros activa
	01	reserved, reject	Primera llamada; transferencia de registro iniciada
	02	reserved, reject	Llamada intermedia; transferencia de registro ya activa

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
80	90	reserved, pass	Dirección lógica inicial no válida
	92	reserved, pass	Tipo no admisible en puntero ANY
	93	reserved, pass	El componente DP direccionado mediante ID o F_ID no está configurado.
	95		Error al leer la información adicional de alarma (con periferia central o con periferia descentralizada a través de interfaz DP externa, este error se emite como "Error de grupo".) Nota: Al acoplar y sincronizar puede suceder que la información adicional de alarma no esté disponible temporalmente.
	96		La CPU maestra ha pasado a STOP. En ese momento se estaba procesando un OB. La instrucción "RALRM (Página 6745)" no puede devolver la información de arranque del OB, la información administrativa, la información de encabezado ni la información adicional de alarma. La información de arranque del OB se puede leer con la instrucción "RD_SINFO (Página 6961)". Además, para los OBs 4x, 55, 56, 57, 82 y 83 es posible leer en modo asincrónico con la instrucción "DPNRM_DG (Página 6803)" el telegrama de diagnóstico actual del esclavo DP en cuestión (información de dirección extraída de la información de arranque del OB).
	A0	read error	Confirmación negativa al leer el módulo
	A1	write error	Confirmación negativa al escribir en el módulo
	A2	module failure	Error de protocolo DP con Layer 2 (p. ej. fallo de esclavo o fallos de bus)
	A3	reserved, pass	• PROFIBUS DP: Error de protocolo DP con Direct-Data-Link-Mapper o User-Interface/ User • PROFINET IO: Error CM general
	A4	reserved, pass	Fallo de comunicación en el bus K
	A5	reserved, pass	–
	A7	reserved, pass	El esclavo DP o el módulo está ocupado en otra tarea (error temporal)
	A8	version conflict	El esclavo DP o el módulo notifica versiones no concordantes
	A9	feature not supported	Función no soportada por el esclavo DP o el módulo
	AA a AF	user specific	El esclavo DP o el módulo notifica un error específico de fabricante en su aplicación. Consulte la documentación del fabricante del esclavo DP o el módulo.

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
	B0	invalid index	El módulo no conoce el registro No se admiten números de registro ≥ 256
	B1	write length error	La indicación de longitud en el parámetro RECORD es errónea; • Con "RALRM (Página 6745)": Error de longitud en AINFO, • Con "RDREC (Página 6735)" y "WRREC (Página 6738)": Error de longitud en MLEN
	B2	invalid slot	El slot configurado no está ocupado.
	B3	type conflict	El tipo de módulo real no coincide con el tipo de módulo configurado
	B4	invalid area	El esclavo DP o el módulo notifica un acceso a un área no autorizada.
	B5	state conflict	El esclavo DP o el módulo no está listo
	B6	access denied	El esclavo DP o el módulo deniega el acceso
	B7	invalid range	El esclavo DP o el módulo notifica un área no autorizada en un parámetro o un valor
	B8	invalid parameter	El esclavo DP o el módulo notifica un parámetro no autorizado
	B9	invalid type	El esclavo DP o el módulo notifica un tipo no autorizado • Con "RDREC (Página 6735)": Búfer demasiado pequeño (no es posible leer porciones) • Con "WRREC (Página 6738)": Búfer demasiado pequeño (no es posible escribir porciones)
	BA a BF	user specific	El esclavo DP o el módulo notifica un error específico de fabricante en el acceso. Consulte la documentación del fabricante del esclavo DP o el módulo. Nota sobre el valor B#16#BA: con PROFINET en el sistema H se aplica lo siguiente: si se rechaza una petición de registro con el valor de retorno W#16#80BA, debe repetirse la petición.
	C0	read constrain conflict	Con "WRREC (Página 6738)": Los datos solo pueden escribirse en el estado STOP de la CPU. Nota: En consecuencia, el programa de usuario no puede escribir. Solo es posible escribir datos online con la PG/el PC. Con "RDREC (Página 6735)": El registro está cargado en el módulo, pero todavía no existen datos o los datos solo pueden leerse en el estado STOP de la CPU. Nota: Si los datos solo pueden leerse en el estado STOP de la CPU, el programa de usuario no puede evaluarlos. En tal caso solo es posible leer los datos online con la PG/el PC.

Error_Decode (B#16#...)	Error_Code_1 (B#16#...)	Explicación según DPV1	Significado
	C1	write constrain conflict	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.
	C2	resource busy	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.
	C3	resource unavailable	En este momento los recursos necesarios están ocupados.
	C4		Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce con frecuencia, compruebe posibles fallos de origen eléctrico en la estructura.
	C5		El esclavo DP o el módulo no está disponible.
	C6		Transferencia del registro cancelada debido a una interrupción de la clase de prioridad
	C7		Interrupción de la petición debido a rearranque completo (en caliente) o arranque en frío del maestro DP
	C8 bis CF		El esclavo DP o el módulo notifica a sus recursos un error específico de fabricante. Consulte la documentación del fabricante del esclavo DP o el módulo.
	Dx	user specific	Específico de esclavo DP. Ver descripción del esclavo DP.
8x (x = 1, ... 9, A, B, C, D, E, F)	00 a FF		Error en el parámetro de llamada y (y = 1, ...15) • Error_Code_1 = 00: Modo de operación no admisible • Para todos los demás valores de Error_Code_1, consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
FE, FF	00 bis FF		Error específico de perfil

Elemento de campo STATUS[4]

STATUS[4] es transferido por el maestro DP a la CPU y a la instrucción en caso de error de DPV1. Sin error de DPV1, el valor se ajusta a 0 con las siguientes excepciones para las instrucciones "RDREC (Página 6735)" y WRREC (Página 6738)":

- STATUS[4] contiene la longitud del área de destino extraída de RECORD, en caso de que LEN > longitud del área de destino extraída de RECORD
- STATUS[4]=LEN, si la longitud real del registro < LEN < longitud del área de destino extraída de RECORD
- STATUS[4]=0, si se debiera ajustar STATUS[4] > 255

Con PROFINET IO STATUS[4] contiene el valor 0.

Parámetro TINFO

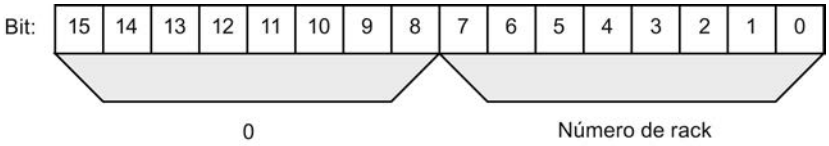
Estructura de datos del área de destino TINFO

Byte	Significado
de 0 a 19	Información de arranque del OB en el que se ha llamado actualmente "RALRM"
20 y 21	Dirección y descripción exacta: ver abajo
de 22 a 31	Información administrativa y descripción exacta: ver abajo

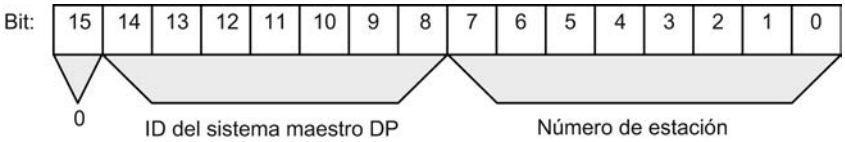
Estructura de la dirección (bytes 20 y 21)

La dirección contiene:

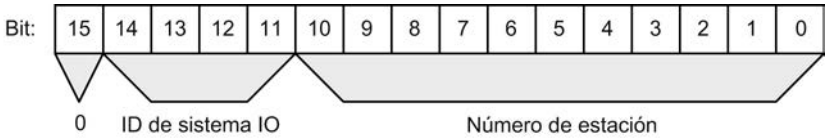
- En caso de configuración centralizada, el número de rack (0-31).



- En caso de estructura descentralizada con PROFIBUS DP
  - La ID del sistema maestro DP (1-32)
  - El número de estación (0-127).



- En caso de estructura descentralizada con PROFINET IO
  - Las dos últimas posiciones de la ID del sistema PROFINET IO (0-15); para obtener la ID del sistema PROFINET IO completa, sume 100 (decimal)
  - El número de estación (0-2047).



Estructura de la información administrativa, bytes 20 a 25

Número de byte para TINFO	Tipo de datos	Significado				
20	BYTE	Centralizado:	0			
		Descentralizado:	PROFIBUS DP: La ID del sistema maestro DP (valores posibles: de 1 a 32)			
			PROFINET IO: ver arriba			
21	BYTE	Centralizado:	Número de rack (valores posibles: de 0 a 31)			
		Descentralizado:	Número de la estación DP (valores posibles: de 0 a 127)			
			PROFINET IO: ver arriba			
22	BYTE	Centralizado:	• 0: Registro 0 o registro 1			
		Descentralizado:	bits de 0 a 3:	Tipo de esclavo	0000: 0001: 0010: 0011: 0100 – 0111: 1000: a partir de 1001:	DP (estructura registro 0) DPS7 (estructura registro 0 o registro 1) DPS7 V1 (estructura registro 0 o registro 1) DPV1 (estructura según estándar PROFIBUS DP) reservado PROFINET IO (estructura según estándar PROFINET IO) reservado
			• bits de 4 a 7:	Tipo de perfil		Reservado
23	BYTE	Centralizado:	• 0			

Número de byte para TINFO	Tipo de datos	Significado				
		Descentralizado:	• bits de 0 a 3:	Tipo de información de alarma	0000:	Transparente, como siempre con PROFINET IO (la alarma procede de un módulo descentralizado configurado)
					0001:	Representante (alarma de un esclavo no DPV1/un dispositivo no IO o de un slot no configurado)
					0010:	Generada (alarma generada en la CPU)
					a partir de 0011:	Reservado
			• bits de 4 a 7:	Versión de la estructura	0000: a partir de 0001:	Inicial Reservado
24	BYTE	Centralizado:	• 0			
		Descentralizado:	Flags de la interfaz de maestro PROFIBUS DP/interfaz de controlador PROFINET IO			
			Bit 0 = 0:	Alarma procedente de una interfaz integrada (PROFINET IO o PROFIBUS DP)		
			Bit 0 = 1:	Alarma procedente de una interfaz externa (PROFINET IO o PROFIBUS DP)		
			Bits de 1 a 7:	Reservado		
25	BYTE	Centralizado:	• 0			
		Descentralizado:	Flags de la interfaz de esclavo PROFIBUS DP			
			• Bit 0:	EXT_DIAG_FLAG del telegrama de diagnóstico, o 0 si este bit no está presente en la alarma. El bit es 1 si el esclavo DP tiene un fallo.		
			• Bits de 1 a 7:	Reservado		
			Flags de la interfaz del controlador PROFINET IO			
			• Bit 0:	ARDiagnosisstate o 0, si no hay información en la alarma. El bit es 1 si el dispositivo IO tiene un fallo.		
			• Bits de 1 a 7:	Reservado		

Estructura de la información administrativa de los bytes 26 a 27 con PROFIBUS y con configuración centralizada

Número de byte para TINFO	Tipo de datos	Significado	
26 y 27	WORD	Centralizado:	0
	WORD	Descentralizado:	Número de identificación de PROFIBUS como identificación unívoca del esclavo PROFIBUS DP
28 y 29	WORD	0	(Pueden omitirse los bytes 28 y 29)
30 y 31	WORD	0	(Pueden omitirse los bytes 30 y 31)

Estructura de la información administrativa, bytes 26 a 31 con PROFINET IO

Número de byte para TINFO	Tipo de datos	Significado	
26 y 27	WORD	Descentralizado:	Número de identificación del dispositivo PROFINET IO como identificación unívoca del dispositivo
28 y 29	WORD	Descentralizado:	Identificación de fabricante
30 y 31	WORD	Descentralizado:	Número de identificación de la instancia

Parámetro AINFO

Estructura de datos del área de destino AINFO con alarmas de PROFIBUS DP o periferia centralizada

Encontrará los datos para PROFINET IO más abajo.

Byte	Significado	
de 0 a 3	Información de encabezado y descripción exacta: ver abajo	
de 4 a 199	Información adicional de alarma: Datos sobre la alarma en cuestión:	
	Centralizado:	ARRAY[0] a ARRAY[195]
	Descentralizado:	ARRAY[0] a ARRAY[59]

Estructura de la información de encabezado con alarmas de PROFIBUS DP o periferia centralizada

Byte	Tipo de datos	Significado	
0	BYTE	Longitud en bytes de la información de alarma recibida	
		Centralizado:	de 4 a 224
		Descentralizado:	de 4 a 63
1	BYTE	Centralizado:	Reservado

Byte	Tipo de datos	Significado		
		Descentralizado:	Identificador del tipo de alarma	
			1: 2: 3: 4: 5: 6: 31	Alarma de diagnóstico Alarma de proceso Alarma de extracción Alarma de inserción Alarma de estado Alarma de actualización Fallo de un aparato de ampliación, un sistema maestro DP o una estación DP
			de 32 a 126:	Alarma específica de fabricante
2	BYTE	Número de slot del componente causante de la alarma		
3	BYTE	Centralizado:	Reservado	
		Descentralizado:	Specifier	
			Bits 0 y 1:	0: No hay más información; 1: Evento entrante, slot con fallo 2: Evento saliente, slot ya sin fallo 3: Evento entrante, slot sigue con fallo
			Bit 2:	Add_Ack
			Bits de 3 a 7:	Número de secuencia

Estructura de datos del área de destino AINFO con alarmas de PROFINET IO

Byte	Significado
de 0 a 25	Información de encabezado y descripción exacta: ver abajo
de 26 a 1431	Información adicional de alarma: Datos de diagnóstico normalizados sobre la alarma en cuestión: ARRAY[0] a ARRAY[1405] Nota: La información adicional de alarma también puede omitirse.

Estructura de la información de encabezado con alarmas de PROFINET IO

Byte	Tipo de datos	Significado
0 y 1	WORD	• Bits de 0 a 7: Tipo de bloque • Bits de 8 a 15: Reservado
2 y 3	WORD	Longitud del bloque
4 y 5	WORD	Versión: • Bits de 0 a 7: low byte • Bits de 8 a 15: high byte

Byte	Tipo de datos	Significado
6 y 7	WORD	Identificador del tipo de alarma: • 1: Alarma de diagnóstico (entrante) • 2: Alarma de proceso • 3: Alarma de extracción • 4: Alarma de inserción • 5: Alarma de estado • 6: Alarma de actualización • 7: Alarma de redundancia • 8: Controlado por el Supervisor (Controlled_by_Supervisor) • 9: Habilitado por el Supervisor (Released_by_Supervisor) • 10: No se ha insertado el módulo configurado • 11: Retorno del submódulo • 12: Alarma de diagnóstico (saliente) • 13: Aviso de conexión para comunicación directa • 14: Aviso de cambio de vecindario • 15: Aviso de sincronización de ciclo (desde el bus) • 16: Aviso de sincronización de ciclo (desde el dispositivo) • 17: Aviso de componente de red • 18: Aviso de sincronización de reloj (desde el bus) • de 19 a 31: Reservado • de 32 a 127: Alarma específica de fabricante • de 128 a 65535: Reservado
de 8 a 11	DWORD	API (Application Process Identifier)
de 12 a 13	WORD	Número de slot del componente que dispara la alarma (rango de valores de 0 a 65535)
de 14 a 15	WORD	Número de slot de submódulo del componente que dispara la alarma (rango de valores de 0 a 65535)
de 16 a 19	DWORD	Identificación de módulo; información unívoca del origen de la alarma

Byte	Tipo de datos	Significado
de 20 a 23	DWORD	Identificación de submódulo; información unívoca del origen de la alarma
de 24 a 25	WORD	Alarm Specifier: • Bits de 0 a 10: Número de secuencia (rango de valores de 0 a 2047) • Bit 11: Diagnóstico de canal: 0: No hay ningún diagnóstico de canal 1: Hay diagnóstico de canal • Bit 12: Estado de diagnóstico específico de fabricante: 0: No hay ninguna información de estado específica de fabricante 1: Hay información de estado específica de fabricante • Bit 13: Estado del diagnóstico en el submódulo: 0: No hay información de estado, se han eliminado todos los errores 1: Hay por lo menos un diagnóstico de canal y/o una información de estado • Bit 14: Reservado • Bit 15: Application Relationship Diagnosis State: – 0: Ninguno de los módulos configurados dentro de esta AR notifica un diagnóstico – 1: Por lo menos uno de los módulos configurados dentro de esta AR notifica un diagnóstico

Estructura de la información adicional de alarma con alarmas de PROFINET IO

La información adicional de alarma con PROFINET IO depende del especificador de formato. Puede estar compuesta por varios bloques de datos con el mismo o distinto especificador de formato. Son posibles los siguientes especificadores de formato:

- W#16#0000 a W#16#7FFF: Diagnóstico específico de fabricante

Byte	Tipo de datos	Significado
de 0 a 1	WORD	Identificación de formato para la estructura de los datos que siguen como información adicional de alarma: W#16#0000 a W#16#7FFF: Diagnóstico específico de fabricante
de 2 a n	BYTE	Ver manual del fabricante.

- W#16#8000: Diagnóstico de canal
  El diagnóstico de canal se emite en bloques de 6 bytes cada uno. La información adicional de alarma (sin especificador de formato) se emite solo para los canales que presentan fallos.

Byte	Tipo de datos	Significado	
de 0 a 1	WORD	Identificación de formato para la estructura de los datos que siguen como información adicional de alarma: W#16#8000: Diagnóstico de canal	
de 2 a 3	WORD	Número de canal del componente que dispara la alarma (rango de valores de 0 a 65535): • W#16#0000 a W#16#7FFF: Número de canal del submódulo/módulo • W#16#8000: es el representante para todo el submódulo • W#16#8001 a W#16#FFFF: Reservado	
4	BYTE	Bits de 0 a 2:	Reservado
		Bits de 3 a 4:	Tipo de error: • 0: Reservado • 1: Error entrante • 2: Error saliente • 3: Error saliente, hay más errores
		Bits de 5 a 7:	Tipo de canal: • 0: Reservado • 1: Canal de entrada • 2: Canal de salida • 3: Canal de entrada/salida

Byte	Tipo de datos	Significado
5	BYTE	Formato de datos: • B#16#00: Formato de datos libre • B#16#01: Bit • B#16#02: 2 bits • B#16#03: 4 bits • B#16#04: Byte • B#16#05: Palabra • B#16#06: Palabra doble • B#16#07: 2 palabras dobles • B#16#08 a B#16#FF: Reservado
de 6 a 7	WORD	Tipo de error: • W#16#0000: reservado • W#16#0001: Cortocircuito • W#16#0002: Subtensión • W#16#0003: Sobretensión • W#16#0004: Sobrecarga • W#16#0005: Sobretemperatura • W#16#0006: Rotura de hilo • W#16#0007: Rebase del límite superior • W#16#0008: Rebase del límite inferior • W#16#0009: Error • W#16#000A a W#16#000F: Reservado • W#16#0010 a W#16#001F: Específico de fabricante • W#16#0020 a W#16#00FF: Reservado • W#16#0100 a W#16#7FFF: Específico de fabricante • W#16#8000: Hay diagnóstico de dispositivo • W#16#8001 a W#16#FFFF: Reservado No todos los canales soportan todos los tipos de error. Encontrará más detalles en la descripción de los datos de diagnóstico del dispositivo correspondiente.

Nota

La parte de "Número de canal" a "Tipo de error" puede aparecer entre 0 y n veces.

W#16#8001

W#16#8001: MULTIPLE (se transfieren distintos tipos de información de diagnóstico)

En este caso, la información adicional de alarma se transfiere en estructura de bloque con longitud variable.

Byte	Tipo de datos	Significado
de 0 a 1	WORD	Especificador de formato para la estructura de los datos que siguen como información adicional de alarma: W#16#8001: Diagnóstico específico de fabricante y/o diagnóstico de canal
de 2 a 3	WORD	Tipo de bloque
de 4 a 5	WORD	Longitud del bloque
6	BYTE	Versión: high byte
7	BYTE	Versión: low byte
de 8 a 11	DWORD	API (solo si el low byte de la versión = 1)
de 12 a 13	WORD	Número de slot
de 14 a 15	WORD	Número de subslot
de 16 a 17	WORD	Número de canal
de 18 a 19	WORD	Propiedades de canal
de 20 a 21	WORD	Identificación de formato: • W#16#0000 a W#16#7FFF: Diagnóstico específico de fabricante • W#16#8000: Diagnóstico de canal • W#16#8002: Diagnóstico avanzado de canal • W#16#8003: Diagnóstico avanzado de canal graduado • W#16#8004 a W#16#80FF: Reservado
de 22 a n	BYTE	Datos en función del especificador de formato

Nota

La parte a partir de "Tipo de bloque" puede aparecer entre 1 y n veces.

W#16#8002

W#16#8002: Diagnóstico avanzado de canal

Byte	Significado
de 0 a 1	Identificación de formato W#16#8002
de 2 a 3	Número de canal
de 4 a 5	Propiedades de canal
de 6 a 7	Tipo de error
de 8 a 9	Valor adicional de error
de 10 a 13	Información adicional de error

W#16#8003

W#16#8003: Diagnóstico avanzado de canal graduado

Byte	Significado
de 0 a 1	Identificación de formato W#16#8003
de 2 a 3	Número de canal
de 4 a 5	Propiedades de canal
de 6 a 7	Tipo de error
de 8 a 9	Valor de error adicional
de 10 a 13	Información adicional de error
de 14 a 17	Qualified Channel Qualifier

W#16#8100

W#16#8100: Información de mantenimiento

Byte	Significado
de 0 a 1	Identificación de formato W#16#8100
de 2 a 3	Tipo de bloque
de 4 a 5	Longitud del bloque
de 6 a 7	Versión del bloque
de 8 a 9	Reservado
de 10 a 13	Estado de mantenimiento

Nota

Encontrará información más detallada sobre la estructura de la información adicional de alarma en el *Manual de programación SIMATIC PROFINET IO De PROFIBUS DP a PROFINET IO* y en la versión más reciente de la norma IEC 61158-6-10-1.

Área de destino TINFO y AINFO

Área de destino TINFO y AINFO

En función del OB en el que se llame la instrucción "RALRM", se escribe solo parcialmente en las áreas de destino TINFO y AINFO. La tabla siguiente describe la información que se registra en cada caso.

Tipo de alarma	OB	TINFO Información de estado de OB	TINFO Información administrativa	AINFO Información de encabezado	AINFO Información de alarma adicional	
Alarma de proceso	4x	Sí	Sí	Sí	Centralizado:	No
					Descentralizado:	Igual que la proporcionada por el esclavo PROFIBUS DP/dispositivo PROFINET IO
Alarma de estado	55	Sí	Sí	Sí	Sí	Sí
Alarma de actualización	56	Sí	Sí	Sí	Sí	Sí
Alarma específica de fabricante	57	Sí	Sí	Sí	Sí	Sí
Error de redundancia de la periferia	70	Sí	Sí	No	No	No
Alarma de diagnóstico	82	Sí	Sí	Sí	Centralizado:	Registro 1
					Descentralizado:	Igual que la proporcionada por el esclavo PROFIBUS DP/dispositivo PROFINET IO
Alarma de extracción/ inserción	83	Sí	Sí	Sí	Centralizado:	No
					Descentralizado:	Igual que la proporcionada por el esclavo PROFIBUS DP/dispositivo PROFINET IO
Alarma de extracción especial: Controlada por el supervisor	83	Sí	Sí	Sí	Solo PROFINET IO	
Alarma de inserción especial: Habilitada por el supervisor	83	Sí	Sí	Sí	Solo PROFINET IO	
Se ha insertado un módulo no configurado	83	Sí	Sí	Sí	Solo PROFINET IO	
Fallo de rack/ fallo de estación	86	Sí	Sí	No	No	
... Todos los demás OBs		Sí	No	No	No	

D_ACT_DP: Activar / desactivar esclavos DP

Descripción

La instrucción "D_ACT_DP" permite desactivar y volver a activar de forma selectiva esclavos DP o dispositivos PROFINET IO configurados. Además es posible averiguar, para cada esclavo DP o dispositivo PROFINET IO utilizado, si el componente en cuestión está activado o desactivado en un determinado momento.

Si se desactiva con esta instrucción una gateway del tipo IE/PB Link PN IO, se desactivarán también todos los esclavos PROFIBUS DP conectados a ella. Estas desactivaciones se notifican.

Esta instrucción no se puede usar en aparatos de campo PROFIBUS PA que estén conectados mediante DP/PA Link a un sistema maestro DP.

Nota

Mientras estén activas una o varias peticiones "D_ACT_DP", no es posible cargar de la programadora a la CPU ninguna configuración modificada (en el marco de CiR).
Durante la carga de una configuración modificada de la programadora a la CPU durante el funcionamiento (CiR), la CPU rechaza la activación de las peticiones "D_ACT_DP".

Para procesar una petición de desactivación o activación se requieren varios pasos por el punto de control de ciclo. Por ello no es posible esperar la finalización de una petición de este tipo dentro de un bucle programado.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "D_ACT_DP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control disparado por nivel REQ=1: Realizar activación o desactivación
MODE	Input	BYTE	I, Q, M, D, L o constante	Identificador de petición Valores posibles: • 0: Conocer si el componente llamado está activado o desactivado • 1: Activar esclavo DP o dispositivo PROFINET IO • 2: Desactivar esclavo DP o dispositivo PROFINET IO • 3: Activar esclavo DP o dispositivo PROFINET IO y llamar el OB 86 una vez cambiado el estado de activación • 4: Desactivar esclavo DP o dispositivo PROFINET IO y llamar el OB 86 una vez cambiado el estado de activación

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	WORD	I, Q, M, D, L o constante	Cualquier dirección lógica del esclavo DP o el dispositivo PROFINET IO
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	Identificación de componente activo: • BUSY=1: La petición todavía está activa. • BUSY=0: La petición ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
0001	El esclavo DP o dispositivo PROFINET IO está activado (este código de error solo es posible con MODE = 0.)
0002	El esclavo DP o dispositivo PROFINET IO está desactivado (este código de error solo es posible con MODE = 0.)
7000	Primera llamada con REQ=0. La petición definida mediante LADDR No está activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1. Se ha lanzado la petición definida mediante LADDR; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante). La petición activada todavía está procesándose; BUSY tiene el valor "1".
8090	• No se ha configurado un módulo con la dirección especificada en LADDR. • Se está utilizando la CPU como esclavo I y se ha indicado en LADDR una dirección de este esclavo I.
8092	El proceso de desactivación en curso de un esclavo DP o un dispositivo PROFINET IO (MODE=2) no se puede interrumpir activándolo (MODE=1). Active el componente en otro momento.
8093	No hay ningún esclavo DP o dispositivo PROFINET IO asignado a la dirección indicada en LADDR (no hay configuración), o el parámetro MODE es desconocido.
8094	Se ha intentado activar un dispositivo que podría ser interlocutor de un puerto de cambio de herramienta. Sin embargo, en este momento hay otro dispositivo activado en ese puerto de cambio de herramienta. El dispositivo activado permanece activado.
80A1	No se ha podido parametrizar el componente llamado. (Este código de error solo es posible con MODE = 1.) Nota: Esta información de error devuelve "D_ACT_DP" solo si este componente vuelve a fallar durante la parametrización del esclavo o dispositivo activado. Si solo ha fallado la parametrización de un único módulo, "D_ACT_DP" devuelve la información de error W#16#0000.
80A2	El esclavo DP llamado no responde (esta información de error no existe para dispositivos PROFINET IO. En PROFINET no existe vigilancia de tiempo para este proceso.).
80A3	El maestro DP o controlador PROFINET IO afectado no soporta esta función.

Código de error (W#16#...)	Explicación
80A4	La CPU no soporta esta función para maestros DP o controladores PROFINET IO externos.
80A6	Error de slot en el esclavo DP o dispositivo PROFINET IO; no se puede acceder a todos los datos útiles (este código de error solo es posible con MODE=1.) Nota: "D_ACT_DP" devuelve esta información de error solo si el componente vuelve a fallar después de la parametrización del componente activado y antes del final de "D_ACT_DP". Si solo hay un módulo no disponible, "D_ACT_DP" devuelve la información de error W#16#0000.
80C1	"D_ACT_DP" se ha iniciado y se continúa con otra dirección lógica (este código de error es posible con MODE=1 y MODE=2).
80C3	• Error temporal de recursos: En este momento la CPU está procesando el máximo posible de peticiones de activación/desactivación. (Este código de error solo es posible con MODE = 1 y MODE = 2.) • La CPU está recibiendo una configuración modificada. Debido a ello, en este momento no es posible desactivar ni activar esclavos DP o dispositivos PROFINET IO.
80C5	DP: Las peticiones todavía no recogidas por el usuario se descartan en el rearranque.
80C6	PROFINET: Las peticiones todavía no recogidas por el usuario se descartan en el rearranque.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Campo de aplicación

Si en una de las CPU se configuran esclavos DP o dispositivos PROFINET IO que no existen realmente o que actualmente no se necesitan, pese a ello la CPU accederá regularmente a dichos esclavos DP o dispositivos PROFINET IO . Una vez desactivados dichos componentes, la CPU no vuelve a acceder a ellos. Con PROFIBUS DP, esto permite obtener el ciclo de bus DP más rápido posible, y evita además la aparición de los correspondientes eventos de error.

Ejemplos

En la fabricación de máquinas en serie, el fabricante dispone de una gran cantidad de opciones de máquina. Sin embargo, cada máquina suministrada contiene una combinación única de opciones seleccionadas.

Todas las opciones de máquina posibles son configuradas por el fabricante como esclavos DP o dispositivos PROFINET IO, a fin de poder crear y llevar un programa de usuario común para todas las opciones posibles. Con "D_ACT_DP" es posible desactivar en el arranque de la máquina todos los esclavos DP o dispositivos PROFINET IO que no estén presentes en la configuración real.

Algo similar sucede en el caso de las máquinas herramienta, que ofrecen una amplia gama de herramientas pero solo utilizan unas pocas en cada momento concreto. En estos casos, las herramientas se configuran como esclavos DP o dispositivos PROFINET IO. Con ayuda de "D_ACT_DP", el programa de usuario activa las herramientas requeridas en cada momento y desactiva las que no se van a utilizar hasta más adelante.

Funcionamiento

"D_ACT_DP" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "D_ACT_DP" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Identificación de una petición

Si se ha lanzado una petición de desactivación o activación y se vuelve a llamar "D_ACT_DP" antes de que finalice, el comportamiento posterior de la instrucción depende esencialmente de si la nueva llamada corresponde a la misma petición. Si el parámetro de entrada LADDR coincide, la llamada se considera una llamada consecutiva.

Desactivar esclavos DP o dispositivos PROFINET IO

Si se desactiva un esclavo DP o un dispositivo PROFINET IO con "D_ACT_DP", sus salidas de proceso se ajustan a los valores sustitutivos configurados, o a 0 (estado seguro). A partir de entonces, el correspondiente maestro DP o controlador PROFINET IO no volverá a llamar ese componente. Los esclavos DP o dispositivos PROFINET IO desactivados no se identifican como averiados o ausentes en los LEDs de error del maestro DP o controlador PROFINET IO ni de la CPU.

La memoria imagen de proceso de las entradas de los esclavos DP o dispositivos PROFINET IO desactivados se actualiza con 0, es decir, se actúa como si se tratase de esclavos DP o dispositivos PROFINET IO averiados.

Si en el programa se accede mediante acceso directo a los datos útiles de un esclavo DP o dispositivo PROFINET IO desactivado anteriormente, se llama el OB de error de acceso a la periferia (OB 122) y se guarda en el búfer de diagnóstico el correspondiente evento de arranque. Si mediante una instrucción (p. ej. "RD_REC (Página 6769)") se accede a un esclavo DP o dispositivo PROFINET IO desactivado, se recibe en RET_VAL la misma información de error que con un esclavo DP o dispositivo PROFINET IO no disponible.

La desactivación de un esclavo DP o un dispositivo PROFINET IO no ocasiona el arranque del OB de error de ejecución del programa (OB 85), aunque las entradas y salidas de éste pertenezcan a la memoria imagen de proceso que debe actualizarse por el sistema. Tampoco se realiza un registro en el búfer de diagnóstico.

Para que la desactivación de un esclavo DP o dispositivo PROFINET IO dé lugar al arranque del OB de fallo del rack (OB 86) y a un registro en el búfer de diagnóstico, debe ajustarse el parámetro MODE :

- MODE = 2: No se arranca el OB 86 ni se realiza el registro en el búfer de diagnóstico
- MODE = 4: Se arranca el OB 86 y se realiza el registro en el búfer de diagnóstico

Si una estación DP o PNIO falla después de haberla desactivado con "D_ACT_DP", el sistema operativo no reconoce el fallo. En consecuencia, no se arranca el OB 86 ni se realiza un registro en el búfer de diagnóstico.

Para PROFIBUS DP se aplica lo siguiente: Si se desea desactivar un esclavo DP que participa como emisor en la comunicación directa, se recomienda desactivar en primer lugar los

receptores que comparten los datos que el emisor envía a su maestro DP. Una vez hecho esto ya se puede desactivar el emisor.

Activar esclavos DP o dispositivos PROFINET IO

Si se vuelve a activar un esclavo DP o dispositivo PROFINET IO con "D_ACT_DP", este componente es configurado y parametrizado por el correspondiente maestro DP o controlador PROFINET IO (igual que cuando se recupera una estación DP o PROFINET IO tras un fallo). La activación ha finalizado cuando el componente está ya en disposición de transferir datos útiles.

La activación de un esclavo DP o un dispositivo PROFINET IO no ocasiona el arranque del OB de error de ejecución del programa (OB 85), aunque las entradas y salidas de éste pertenezcan a la memoria imagen de proceso que debe actualizarse por el sistema. Tampoco se realiza un registro en el búfer de diagnóstico.

Para que la desactivación de un esclavo DP o dispositivo PROFINET IO dé lugar al arranque del OB de fallo del rack (OB 86) y a un registro en el búfer de diagnóstico, debe ajustarse el parámetro MODE :

- MODE = 1: No se arranca el OB 86 ni se realiza el registro en el búfer de diagnóstico
- MODE = 3: Se arranca el OB 86 y se realiza el registro en el búfer de diagnóstico

Si se intenta activar con "D_ACT_DP" un esclavo DP desactivado que ha sido desconectado físicamente del bus DP, la instrucción envía al cabo de aproximadamente un minuto el código de error W#16#80A2, y el esclavo DP permanece desactivado. Si en un momento posterior el esclavo vuelve a estar conectado con el bus DP, es necesario activarlo de nuevo mediante "D_ACT_DP".

Si se intenta activar un dispositivo PROFINET IO que está desconectado físicamente del bus PN, "D_ACT_DP" permanece activa. A diferencia de lo que sucede con los esclavos DP, no se produce una cancelación automática una vez transcurrido un tiempo determinado. La petición en curso debe interrumpirse manualmente.

Nota

La activación de un esclavo DP o un dispositivo PROFINET IO puede tomar un tiempo considerable. Para cancelar una petición de activación en curso, debe iniciarse "D_ACT_DP" con el mismo valor para LADDR y MODE = 2. Debe repetirse la llamada de "D_ACT_DP" con MODE = 2 tantas veces como sea necesario hasta que se señalice, mediante RET_VAL = 0, que se ha cancelado la petición de activación.

Si se desea activar esclavos DP que participan en la comunicación directa, se recomienda activar primero los emisores y luego los receptores (oyentes).

Arranque de la CPU

El sistema operativo de la CPU muestra distintos comportamientos en los distintos tipos de arranque en relación con los esclavos DP o dispositivos PROFINET IO:

- En los tipos de arranque "arranque en frío" y "rearranque completo (en caliente)", los esclavos o dispositivos desactivados se activan de nuevo automáticamente.
  - Con S7-400: La activación de los esclavos DP o dispositivos PROFINET IO puede durar bastante tiempo. La CPU arranca a pesar ello, y se producen errores de acceso a la periferia hasta finalizar la activación. La reacción de la CPU depende de la parametrización (llamada del OB 85 en caso de errores de acceso a la periferia) y del programa del OB 85. No es posible suprimir estos errores de acceso a la periferia.
  - Con S7-300: La activación de los esclavos DP o dispositivos PROFINET IO puede durar bastante tiempo. En este caso, la CPU espera hasta que los esclavos DP o dispositivos PROFINET IO hayan sido activados. El tiempo de espera máximo para la activación de los esclavos DP o dispositivos PROFINET IO es de aprox. 1 minuto. Dentro de ese tiempo no se producen errores de acceso a la periferia.
- En el tipo de arranque "Rearranque", el estado de activación de los esclavos o dispositivos no se modifica: Los esclavos o dispositivos activados siguen activados, y los esclavos o dispositivos desactivados siguen desactivados.

Después del arranque, la CPU intenta entrar en contacto cíclicamente con todos los esclavos o dispositivos configurados y no desactivados, que no están presentes o no son accesibles.

Nota

No se soporta la llamada de "D_ACT_DP" en los OBs de arranque.

Otras instrucciones

RD_REC: Leer registro de periferia

Descripción

Esta instrucción permite leer el registro con número RECNUM del componente que se ha direccionado. Para iniciar la lectura, debe ajustarse a "1" el parámetro de entrada REQ . Si la operación de lectura se ha podido ejecutar de inmediato, la instrucción devuelve en el parámetro de salida BUSY el valor "0". Si BUSY tiene el valor "1", la operación de lectura todavía no ha finalizado.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595). Una vez efectuada correctamente la transferencia de datos, el registro leído se guarda en el área de destino delimitada mediante RECORD .

Nota

Si se lee un registro con número mayor que 1 desde una FM o un CP adquiridos antes de febrero de 1997 (en lo sucesivo denominados "módulos antiguos"), "RD_REC" se comporta de un modo diferente que con un módulo nuevo. Este caso especial se describe en la sección "Uso de FMs y CPs S7-300 antiguos con n.º de registro > 1" (ver abajo).

Si un esclavo DPV1 está configurado mediante archivo GSD (a partir de la rev. 3 de GSD) y la interfaz DP del maestro DP está ajustada como "Compatible con S7", los módulos de E/S no pueden leer en el programa de usuario los registros con "RD_REC". En este caso, el maestro DP direcciona un slot incorrecto (slot configurado + 3).

Solución: Conmutar la interfaz del maestro DP a "DPV1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_REC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: solicitud de lectura
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección lógica del módulo. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 0 a 240)
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Adicionalmente: Longitud en bytes del registro realmente transferido (valores posibles: de +1 a +240), si el área de destino es mayor que el registro transferido y no se ha producido ningún error durante la transferencia.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La lectura todavía no ha finalizado.
RECORD	Output	ANY	I, Q, M, D, L	Área de destino para el registro leído. En caso de procesamiento asíncrono de "RD_REC", debe comprobarse que los parámetros actuales de RECORD presenten la misma información de longitud en todas las llamadas. Solo se admite el tipo de datos BYTE . Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RECORD

Nota

Para asegurarse de que siempre se lea el registro completo, debe seleccionarse un área de destino con una longitud de 241 bytes. Si la transferencia de datos se realiza sin errores, la longitud real del registro se puede leer en RET_VAL .

Uso de FMs y CPs S7-300 antiguos con n.º de registro > 1

Si se desea leer un registro con número mayor que 1 con la instrucción "RD_REC" desde una FM S7-300 antigua o un CP S7-300 antiguo, debe tenerse en cuenta lo siguiente:

- Si el área de destino es mayor que la longitud real del registro deseado, no se registran datos en RECORD . En RET_VAL se escribe W#16#80B1.
- Si el área de destino es menor que la longitud real del registro deseado, la CPU leerá tantos bytes desde el principio del registro como se indique en la longitud especificada en RECORD y los registrará en RECORD . En RET_VAL se escribe un "0".
- Si la indicación de longitud de RECORD es igual a la longitud real del registro deseado, la CPU lee el registro en cuestión y lo guarda en RECORD ; RET_VAL se ajusta a "0".

Parámetro RET_VAL

- Si se ha producido un error durante la ejecución de la función, el valor de respuesta contendrá un código de error.
- Si no se ha producido ningún error durante la transferencia, RET_VAL adquiere el valor
  - 0 si se ha llenado el área de destino completa con datos del registro seleccionado (aunque el registro esté incompleto).
  - la longitud en bytes del registro realmente transferido (valores posibles: de +1 a +240), si el área de destino es mayor que el registro transferido.

Nota

Si se produce el error general W#16#8745, esto significa solo que durante la operación de escritura no ha sido posible acceder a por lo menos un byte. El registro se ha leído correctamente desde el módulo y escrito correctamente en el área de memoria de E/S.

Dentro de la información de error "verdadera" (códigos de error W#16#8xyz) de la tabla siguiente, debe distinguirse entre dos casos:

- Errores temporales (códigos de error de W#16#80A2 a 80A4, 80Cx):
  En este tipo de errores es posible que el error se elimine sin intervención del usuario, es decir, puede bastar con llamar de nuevo la instrucción (varias veces si es necesario). Ejemplo de error temporal: En este momento los recursos necesarios están ocupados (W#16#80C3).
- Errores permanentes (códigos de error W#16#809x, 80A0, 80A1, 80Bx):
  Los errores de este tipo no se pueden eliminar sin intervención del usuario. Solo tiene sentido volver a llamar la instrucción una vez que se haya eliminado el error. Ejemplo de error permanente: Indicación de longitud errónea en RECORD (W#16#80B1).

Nota

Si se transfieren con "WR_REC (Página 6774)" registros a un esclavo DPV1 o se leen registros RD_REC desde un esclavo DPV1 que funciona en modo DPV1, el maestro DP evalúa de la siguiente manera la información de error recibida desde el esclavo:

Si la información de error se encuentra en las áreas de W#16#8000 a W#16#80FF o de W#16#F000 a W#16#FFFF, el maestro DP reenvía la información de error a la instrucción. Si se encuentra fuera de estas áreas, reenvía a la instrucción el valor W#16#80A2 y suspende el esclavo.

Encontrará la descripción de la información de error procedente de los esclavos DPV1 en STATUS[3] Parámetro STATUS (Página 6748).

Parámetros RET_VAL para WR_REC y RD_REC

Código de error (W#16#...)	Explicación	Restricción
0000	Sin error	-
7000	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.	-
7001	Primera llamada con REQ=1: Transferencia de datos lanzada; BUSY tiene el valor 1.	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transferencia de datos ya activa; BUSY tiene el valor 1.	Periferia descentralizada
8090	La dirección base lógica indicada no es válida: No existe ninguna asignación en el SDB1/SDB2x, o no se trata de una dirección base.	-
8092	En la referencia ANY se ha indicado un tipo distinto a BYTE.	Solo con S7-400
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR y IOID . (Se soportan los módulos S7-300 con S7-300, módulos S7-400 con S7-400, módulos S7-DP con S7-300 y S7-400.)	-
80A0	Acuse negativo al leer el módulo: Módulo extraído durante la operación de lectura o módulo defectuoso	Solo con "RD_REC"
80A1	Acuse negativo al escribir en el módulo: Módulo extraído durante la operación de escritura o módulo defectuoso	Solo con "WR_REC (Página 6774)"
80A2	• Error de protocolo DP con Layer 2 (p. ej. fallo de esclavo o fallos de bus) • Con ET200S no es posible leer registros en el modo DPV0.	Periferia descentralizada
80A3	Error de protocolo de DP con User-Interface/User	Periferia descentralizada
80A4	Fallo de comunicación en el bus K	El error se produce entre la CPU y una interfaz DP externa
80B0	• La instrucción no es posible para el tipo de módulo • El módulo no conoce este registro. • El número de registro 241 no es admisible. • Con "WR_REC (Página 6774)" no se permiten los registros 0 ni 1.	-
80B1	La indicación de longitud en el parámetro RECORD es errónea.	• Con "WR_REC (Página 6774)": Longitud errónea • Con "RD_REC" (posible solo si se usan FMs S7-300 antiguas o CPs S7-300 antiguos):Indicación > longitud registro • Con DPNRM_DG: Indicación < longitud de registro
80B2	El slot configurado no está ocupado.	-
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado en el SDB1	-
80B7	El esclavo DP o el módulo notifica un área no autorizada en un parámetro o un valor.	Solo con "RD_REC"

Código de error (W#16#...)	Explicación	Restricción
80C0	Con "WR_REC (Página 6774)": Los datos solo pueden escribirse en el estado STOP de la CPU. Nota: En consecuencia, el programa de usuario no puede escribir. Solo es posible escribir datos online con la PG/el PC. Con "RD_REC": El registro está cargado en el módulo, pero todavía no existen datos o los datos solo pueden leerse en el estado STOP de la CPU. Nota: Si los datos solo pueden leerse en el estado STOP de la CPU, el programa de usuario no puede evaluarlos. En tal caso solo es posible leer los datos online con la PG/el PC. Con "DPNRM_DG (Página 6803)": No hay datos de diagnóstico.	Con "WR_REC (Página 6774)", "RD_REC" o "DPNRM_DG (Página 6803)"
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	-
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	-
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	-
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe posibles fallos de origen eléctrico en la configuración.	-
80C5	La periferia descentralizada no está disponible.	Periferia descentralizada
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano)	Periferia descentralizada
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)	-

WR_REC: Escribir registro en periferia

Descripción

La instrucción "WR_REC" permite transferir el registro RECORD al módulo direccionado.

Para iniciar la escritura, debe ajustarse a "1" el parámetro de entrada REQ. Si la operación de escritura se ha podido ejecutar de inmediato, la instrucción devuelve en el parámetro de salida BUSY el valor "0". Si BUSY tiene el valor "1", la operación de escritura todavía no ha finalizado.

Nota

Si un esclavo DPV1 está configurado mediante archivo GSD (a partir de la rev. 3 de GSD) y la interfaz DP del maestro DP está ajustada como "Compatible con S7", no se pueden escribir en el programa de usuario registros con "WR_REC" en los módulos de E/S. En este caso, el maestro DP direcciona un slot incorrecto (slot configurado + 3).

Solución: Conmutar la interfaz del maestro DP a "DPV1".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_REC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: solicitud de escritura
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección lógica del módulo. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 2 a 240)
RECORD	Input	ANY	I, Q, M, D, L	Registro. Solo se admite el tipo de datos BYTE. Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
RET_VAL	Output	Return	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY= 1: La escritura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RECORD

Los datos que se deben transferir se leen del parámetro RECORD en la primera llamada. Si la transferencia del registro dura más que una llamada, en las llamadas consecutivas de la instrucción (para la misma petición), el contenido del parámetro RECORD ya no es relevante.

Parámetro RET_VAL

Consulte también: RD_REC: Leer registro de periferia (Página 6769)

Nota

Si aparece el error general W#16#8544, esto solo significa que estaba bloqueado el acceso a por lo menos un byte del área de memoria de E/S que contiene el registro. Se ha proseguido la transferencia de datos.

DPRD_DAT: Leer datos coherentes de un esclavo DP normalizado

Descripción

Con la instrucción "DPRD_DAT" se leen datos coherentes de un esclavo DP normalizado o un dispositivo PROFINET IO, teniendo en cuenta lo siguiente en cuanto a la longitud máxima:

- La longitud máxima puede consultarse en la documentación de la CPU.
- Si no se ha producido ningún error durante la transferencia de datos, los datos leídos se registran en el área de destino delimitada mediante RECORD.

El área de destino debe tener la misma longitud que se haya configurado para el módulo seleccionado. En un esclavo DP normalizado con estructura modular o con varias IDs de DP, la llamada a "DPRD_DAT" solo permite acceder cada vez a los datos de un solo módulo o ID de DP en la dirección configurada.

Nota

Una CPU de las series S7-300/400 soporta hasta 64 bytes de datos coherentes. Para las áreas de datos coherentes cuyo tamaño supere los 4 bytes es necesario utilizar la instrucción "DPRD_DAT". En caso de necesidad, esta instrucción también se puede utilizar para un área de datos a partir de 1 byte. Si se produce un error de acceso, se notifica el código de error W#16#8090.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPRD_DAT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección inicial configurada del área I del módulo del que hay que leer. Nota: La dirección se debe indicar en formato hexadecimal; p. ej. la dirección inicial 100 significa: LADDR:=W#16#64.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
RECORD	Output	ANY	I, Q, M, D, L	Área de destino para los datos útiles leídos. Debe tener la misma longitud que la configurada para el módulo seleccionado. Solo se admite el tipo de datos BYTE. Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Nota

Si accede a esclavos DPV1, el maestro DP puede transferir la información de error de dichos esclavos a la instrucción. Encontrará una descripción de esta información de error en STATUS[3] Parámetro STATUS (Página 6748).

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	• No hay ningún módulo configurado para la dirección base lógica especificada, o • No se ha tenido en cuenta la limitación en toda la longitud de los datos coherentes, o • En el parámetro LADDR no se ha indicado la dirección de inicio en formato hexadecimal.
8092	En la referencia ANY se ha indicado un tipo distinto a BYTE.
8093	Para la dirección lógica indicada en LADDR no existe ningún módulo DP o dispositivo PROFINET IO del que se puedan leer datos de modo coherente.
80A0	Se ha detectado un error de acceso al acceder a la periferia

Código de error (W#16#...)	Explicación
80B0	Fallo de esclavo en interfaz DP externa
80B1	La longitud del área de destino especificada es distinta a la longitud de datos útiles configurada.
80B2	Fallo del sistema con interfaz DP externa
80B3	Fallo del sistema con interfaz DP externa
80C0	Los datos todavía no se han leído del módulo
80C2	Fallo del sistema con interfaz DP externa
80Fx	Fallo del sistema con interfaz DP externa
87xy	Fallo del sistema con interfaz DP externa
808x	Fallo del sistema con interfaz DP externa
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Campo de aplicación

"DPRD_DAT" es necesaria porque con los comandos de carga que acceden a la periferia o a la memoria imagen de proceso de las entradas sólo se puede leer un máximo de cuatro bytes de forma coherente.

Nota

Para leer datos coherentes también puede usarse, en caso necesario, la memoria imagen de proceso de las entradas.

- Para saber si la CPU S7-300 posee esta función, consulte la documentación correspondiente.
- Todas las CPUs S7-400 poseen esta función.

⚠ PRECAUCIÓN

Acceso a la periferia

Al utilizar "DPRD_DAT", evite acceder a áreas de la periferia que tengan asignadas memorias imagen parcial de proceso vinculadas a OB6x (alarmas de sincronismo).

Coherencia de datos

Consulte también: Sección Coherencia de datos (Página 7668).

DPWR_DAT: Escribir datos coherentes de un esclavo DP normalizado

Descripción

Con esta instrucción se transfieren de modo coherente los datos de "RECORD " al esclavo DP normalizado o dispositivo PROFINET IO y, en su caso, a la memoria imagen de proceso

(concretamente si se ha configurado el área de direcciones en cuestión del esclavo DP normalizado como área de coherencia en una memoria imagen de proceso).

Para la longitud máxima de los datos que se deben transferir, se aplica lo siguiente: La longitud máxima puede consultarse en la documentación de la CPU. La transferencia de datos se realiza de forma sincrónica, es decir, con la finalización de la instrucción ha finalizado el proceso de escritura.

El área de origen debe tener la misma longitud que se haya configurado para el módulo seleccionado. Con un esclavo DP normalizado con estructura modular solo es posible acceder a un módulo del esclavo DP.

Nota

Una CPU de las series S7-1200/1500 soporta hasta 64 bytes de datos coherentes. Para las áreas de datos coherentes cuyo tamaño supere los 4 bytes es necesario escribir los datos en un módulo o un dispositivo utilizando la instrucción "DPWR_DAT". En caso de necesidad, esta instrucción también se puede utilizar para un área de datos a partir de 1 byte. Si se produce un error de acceso, se notifica el código de error W#16#8090.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPWR_DAT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección inicial configurada del área IPS del módulo en el que hay que escribir. Nota: La dirección se debe indicar en formato hexadecimal; p. ej. la dirección inicial 100 significa: LADDR:=W#16#64.
RECORD	Input	ANY	I, Q, M, D, L	Área de origen para los datos útiles que se deben escribir. Debe tener la misma longitud que la configurada para el módulo seleccionado. Solo se admite el tipo de datos BYTE . Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Nota

Si accede a esclavos DPV1, el maestro DP puede transferir la información de error de dichos esclavos a la instrucción. Encontrará una descripción de esta información de error en STATUS[3] Parámetro STATUS (Página 6748).

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
808x	Fallo del sistema con interfaz DP externa
8090	• No hay ningún módulo configurado para la dirección base lógica especificada, o • No se ha tenido en cuenta la limitación en toda la longitud de los datos coherentes, o • En el parámetro LADDR no se ha indicado la dirección de inicio en formato hexadecimal.
8092	En la referencia ANY se ha indicado un tipo distinto a BYTE.
8093	Para la dirección lógica indicada en LADDR no existe ningún módulo DP ni dispositivo PROFINET IO en el que se puedan escribir datos de modo coherente.
80A1	Se ha detectado un error de acceso al acceder a la periferia
80B0	Fallo de esclavo en interfaz DP externa
80B1	La longitud del área de origen especificada es distinta a la longitud de datos útiles configurada.
80B2	Fallo del sistema con interfaz DP externa
80B3	Fallo del sistema con interfaz DP externa
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo no han sido procesados todavía por el módulo.
80C2	Fallo del sistema con interfaz DP externa
80Fx	Fallo del sistema con interfaz DP externa
85xy	Fallo del sistema con interfaz DP externa
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Campo de aplicación

"DPWR_DAT" es necesario porque con los comandos de transferencia que acceden a la periferia o a la memoria imagen de proceso de las salidas solo se pueden escribir como máximo cuatro bytes coherentes.

Nota

Para escribir datos coherentes también puede usarse, en caso necesario, la memoria imagen de proceso de las salidas.
Para saber si la CPU S7-300 posee esta función, consulte la documentación correspondiente.

- Todas las CPUs S7-400 poseen esta función.
- Para escribir datos coherentes, no deben utilizarse al mismo tiempo ambas posibilidades: Utilice o bien "DPWR_DAT" o bien la escritura a través de la memoria imagen de proceso de las salidas.

⚠ PRECAUCIÓN

Acceso a la periferia

Al utilizar "DPWR_DAT", evite acceder a áreas de la periferia que tengan asignadas memorias imagen parcial de proceso vinculadas a OB6x (alarmas de sincronismo).

Coherencia de datos

Consulte también: Coherencia de datos (Página 7668).

iDevice / iSlave

RCVREC: Recibir registro

Descripción

Un I-Device puede recibir un registro desde un controlador superior. La recepción se efectúa en el programa de aplicación con la instrucción "RCVREC" (receive record).

La instrucción tiene los siguientes modos de operación:

- Comprobar si el I-Device ha recibido una solicitud para la recepción de un registro.
- Preparar un registro en los parámetros de salida.
- Enviar una respuesta al controlador superior.

Para establecer cuál de estos modos de operación debe ejecutar la instrucción, se usa el parámetro de entrada MODE (ver abajo).

El I-Device debe encontrarse en el estado operativo RUN o en el estado operativo ARRANQUE.

Con MLEN se indica el máximo de bytes que se deben recibir. Por ello debe escogerse un área de destino RECORDcon una longitud de por lo menos MLENbytes.

Cuando se ha recibido un registro (MODE=1 o MODE=2), en el parámetro de salida NEW se indica que el registro se ha guardado en RECORD. Asegúrese de que RECORD tenga una longitud suficiente. El parámetro de salida LEN contiene la longitud real en bytes del registro recibido.

Para la respuesta positiva al controlador superior, se deben ajustar CODE1 y CODE2 a cero. Si se desea rechazar el registro recibido, debe enviarse con CODE1 el Error Code 1 y con CODE2 el Error Code 2 de la respuesta negativa al controlador superior.

Nota

Si el I-Device ha recibido una solicitud de recepción de un registro, es necesario haber reconocido la emisión de dicha solicitud dentro de un período determinado. Una vez reconocida la solicitud, debe haberse enviado al controlador superior una respuesta dentro del mismo período. De lo contrario, se produce en el I-Device un error de timeout, a consecuencia del cual el sistema operativo del I-Device envía una respuesta negativa al controlador superior. Para conocer dicho período, consulte los datos técnicos de la CPU.

Cuando se produce un error, el parámetro de salida STATUS contiene la información de error.

Modos de operación

El modo de operación de la instrucción "RCVREC" se define mediante el parámetro de entrada MODE. Esto se explica en la siguiente tabla:

MODE	Significado
0	Comprobar si existe una solicitud para la recepción de un registro. Si en el I-Device existe un registro procedente del controlador superior, la instrucción escribirá solo en los parámetros de salida NEW, SLOT, INDEX y LEN. Si se llama la instrucción varias veces consecutivas con MODE=0, los parámetros de salida se refieren siempre a la misma solicitud.
1	Recibir un registro para cualquier subslot del I-Device Si en el I-Device existe un registro procedente del controlador superior para cualquier subslot del I-Device, la instrucción escribe solo en los parámetros de salida y transfiere el registro al parámetro RECORD.
2	Recibir un registro para un determinado subslot del I-Device Si en el I-Device existe un registro procedente del controlador superior para un determinado subslot del I-Device, la instrucción escribe solo en los parámetros de salida y transfiere el registro al parámetro RECORD.
3	Enviar respuesta positiva al controlador superior La instrucción comprueba la solicitud de recepción de registro procedente del controlador superior, acepta el registro existente y envía un acuse positivo al controlador superior.
4	Enviar respuesta negativa al controlador superior La instrucción comprueba la solicitud de recepción de registro procedente del controlador superior, rechaza el registro existente y envía un acuse negativo al controlador superior. El motivo del rechazo puede indicarse en los parámetros de entrada CODE1 y CODE2.

Nota

Una vez llegado el registro (NEW=1), debe llamarse "RCVREC" dos veces más para garantizar que se procese por completo. Al hacerlo debe mantenerse el siguiente orden:

- Primera llamada con MODE=1 o MODE=2
- Segunda llamada con MODE=3 o MODE=4

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RCVREC":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	Modo de operación
F_ID	Input	DWORD	I, Q, M, D, L o constante	Subslot en el área de trasferencia del I-Device para el registro que se debe recibir (relevante solo para MODE=2). El high word debe ajustarse siempre a cero.
MLEN	Input	INT	I, Q, M, D, L o constante	Longitud máxima en bytes del registro que se debe transferir.
CODE1	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 1 (para MODE=4)
CODE2	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 2 (para MODE=4)
NEW	Output	BOOL	I, Q, M, D, L	• MODE=0: Se ha recibido un registro nuevo • MODE=1 ó 2: El registro se ha transferido en RECORD
STATUS	Output	DWORD	I, Q, M, D, L	Información de error
SLOT	Output	INT	I, Q, M, D, L	Idéntico a F_ID
SUBSLOT	Output	INT	I, Q, M, D, L	Idéntico a F_ID
INDEX	Output	INT	I, Q, M, D, L	Número del registro recibido
LEN	Output	INT	I, Q, M, D, L	Longitud del registro recibido
RECORD	InOut	ANY	I, Q, M, D, L	Área de destino para el registro recibido. Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 Byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Para la interpretación del parámetro STATUS, véase el capítulo: Parámetro STATUS (Página 6748)

PRVREC: Proveer registro

Descripción

Un I-Device puede recibir desde un controlador superior la solicitud de preparar un registro. La preparación se efectúa en el programa de usuario con la instrucción "PRVREC" (provide record).

La instrucción tiene los siguientes modos de operación:

- Comprobar si el I-Device ha recibido una solicitud para la preparación de un registro.
- Transferencia del registro solicitado al controlador superior.
- Enviar una respuesta al controlador superior.

Para establecer cuál de estos modos de operación debe ejecutar la instrucción, se usa el parámetro de entrada MODE (ver abajo).

El I-Device debe encontrarse en el estado operativo RUN o en el estado operativo ARRANQUE.

Con LEN se indica el máximo de bytes que debe tener el registro que se desea enviar. Por ello debe escogerse un área de destino RECORDcon una longitud de por lo menos LENbytes.

Cuando existe una solicitud de preparación de un registro (MODE=0), el parámetro de salida NEW se ajusta a TRUE.

Si se acepta la solicitud de preparación de registro, para la respuesta positiva al controlador superior debe escribirse en RECORD el registro solicitado, así como ajustar CODE1 y CODE2 a cero. Si se desea rechazar una solicitud de preparación de registro, debe enviarse con CODE1 el Error Code 1 y con CODE2 el Error Code 2 de la respuesta negativa al controlador superior.

Nota

Si el I-Device ha recibido una solicitud de preparación de un registro, es necesario haber reconocido la emisión de dicha solicitud dentro de un período determinado. Una vez reconocida la solicitud, debe haberse enviado al controlador superior una respuesta dentro del mismo período. De lo contrario, se produce en el I-Device un error de timeout, a consecuencia del cual el sistema operativo del I-Device envía una respuesta negativa al controlador superior. Para conocer dicho período, consulte los datos técnicos de la CPU.

Cuando se produce un error, el parámetro de salida STATUS contiene la información de error.

Modos de operación

El modo de operación de la instrucción "PRVREC" se define mediante el parámetro de entrada MODE. Esto se explica en la siguiente tabla:

MODE	Significado
0	Comprobar si existe una solicitud de preparación de un registro. Si en el I-Device existe una solicitud de preparación de registro procedente del controlador superior, la instrucción escribirá solo en los parámetros de salida NEW, SLOT, INDEX y RLEN. Si se llama la instrucción varias veces consecutivas con MODE=0, los parámetros de salida se refieren siempre a la misma solicitud.
1	Recibir una solicitud de preparación de registro para cualquier subslot del I-Device Si en el I-Device existe una solicitud de ese tipo procedente del controlador superior para cualquier subslot del I-Device, la instrucción escribe solo en los parámetros de salida.
2	Recibir una solicitud de preparación de registro para un determinado subslot del I-Device Si en el I-Device existe una solicitud de ese tipo procedente del controlador superior para un determinado subslot del I-Device, la instrucción escribe solo en los parámetros de salida.
3	Preparar registro y enviar respuesta positiva al controlador superior La instrucción comprueba la solicitud de preparación de registro procedente del controlador superior, pone a disposición el registro solicitado en RECORD y envía un acuse positivo al controlador superior.
4	Enviar respuesta negativa al controlador superior La instrucción comprueba la solicitud de preparación de registro procedente del controlador superior, la rechaza y envía un acuse negativo al controlador superior. El motivo del rechazo puede indicarse en los parámetros de entrada CODE1 y CODE2.

Nota

Una vez llegada la solicitud (NEW=1), debe llamarse la instrucción dos veces más para garantizar que se procese por completo. Al hacerlo debe mantenerse el siguiente orden:

- Primera llamada con MODE=1 o MODE=2
- Segunda llamada con MODE=3 o MODE=4

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PRVREC":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	Modo de operación
F_ID	Input	DWORD	I, Q, M, D, L o constante	Subslot en el área de trasferencia del I-Device para el registro que se debe enviar (relevante solo para MODE=2). El high word debe ajustarse siempre a cero.
CODE1	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 1 (para MODE=4)
CODE2	Input	BYTE	I, Q, M, D, L o constante	Cero (para MODE=3) o Error Code 2 (para MODE=4)

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
LEN	Input	INT	I, Q, M, D, L o constante	Longitud máxima en bytes del registro que se debe enviar.
NEW	Output	BOOL	I, Q, M, D, L	El controlador superior ha solicitado un nuevo registro.
STATUS	Output	DWORD	I, Q, M, D, L	Información de error
SLOT	Output	INT	I, Q, M, D, L	Idéntico a F_ID
SUBSLOT	Output	INT	I, Q, M, D, L	Idéntico a F_ID
INDEX	Output	INT	I, Q, M, D, L	Número del registro que se debe enviar
RLEN	Output	INT	I, Q, M, D, L	Longitud del registro que se debe enviar
RECORD	InOut	ANY	I, Q, M, D, L	Registro preparado Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 Byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Para la interpretación del parámetro STATUS, véase el capítulo: Parámetro STATUS (Página 6748)

SALRM: Enviar alarma

Descripción de SALRM

Descripción

Esta instrucción permite enviar desde un programa de usuario de un esclavo inteligente una alarma de un slot situado en el área de transferencia (slot virtual) al maestro DP correspondiente. Esto da lugar al inicio del OB correspondiente en el maestro DP.

Puede asignarse a la alarma información adicional específica. Con esta instrucción puede leerse en el maestro DP la información adicional completa.

La instrucción solo puede utilizarse en el modo compatible con S7.

- DP: El maestro integra el esclavo I mediante el archivo GSD.
- Compatible con S7: El esclavo I se vincula con el maestro mediante configuración.

Nota

La interfaz de la instrucción "SALRM" es idéntica a la del FB "SALRM" definido en la norma "PROFIBUS Guideline PROFIBUS Communication and Proxy Function Blocks according to IEC 61131-3".

Funcionamiento

La instrucción "SALRM" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia de alarma, debe llamarse la instrucción con REQ = 1.

La operación de envío permanece activa hasta que el maestro DP confirma o interrumpe el procesamiento de la alarma.

Mediante el parámetro de salida BUSY y los bytes 2 y 3 del parámetro de salida STATUS se muestra el estado de la petición. Los bytes 2 y 3 de STATUS corresponden al parámetro de salida RET_VAL de las instrucciones asíncronas.

Véase también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

La transferencia de alarma se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE .

Identificación de la petición

Si se ha lanzado el envío de una alarma al maestro DP con la instrucción "SALRM" y se vuelve a llamar esta instrucción antes de que finalice la petición en curso, el comportamiento posterior de la instrucción depende de si la nueva llamada corresponde a la misma petición.

Si los parámetros ID y ATYPE coinciden con una petición todavía no finalizada, la llamada se considerará llamada consecutiva.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SALRM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Realizar la transferencia de la alarma
ID	Input	DWORD	I, Q, M, D, L o constante	Cualquier dirección lógica del área de transferencia al maestro DP (slots virtuales) (desde el punto de vista del esclavo DP) excepto la dirección de diagnóstico de la estación y la dirección lógica del slot 2. La información relevante se encuentra en la palabra menos significativa. La palabra más significativa se ajusta a cero. El bit 15 contiene la identificación E/S: 0 si es una dirección de entrada y 1 si es de salida.
ATYPE (Página 6791)	Input	INT	I, Q, M, D, L o constante	Alarm Type Identificación del tipo de alarma. Valores posibles: • 1: Alarma de diagnóstico • 2: Alarma de proceso
ASPEC (Página 6791)	Input	INT	I, Q, M, D, L o constante	Alarm Specifier: • 0: No hay más información • 1: Evento entrante, slot con fallo • 2: Evento saliente, slot ya sin fallo • 3: Evento entrante, slot sigue con fallo
LEN (Página 6792)	Input	INT	I, Q, M, D, L o constante	Longitud en bytes de la información adicional de alarma que se debe enviar Mayor valor: 16
DONE	Output	BOOL	I, Q, M, D, L	DONE=1: Se ha transferido la alarma
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La transferencia de la alarma todavía no ha finalizado.
ERROR (Página 6793)	Output	BOOL	I, Q, M, D, L	ERROR = 1: Ha ocurrido un error.
STATUS (Página 6793)	Output	DWORD	I, Q, M, D, L	Información de error
AINFO (Página 6794)	InOut	ANY	I, Q, M, D, L	Alarm Info Área de origen de la información adicional de alarma Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro AINFO siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Repercusión de la llamada a "SALRM" en la información de estado de módulo y el LED de error de grupo (SF)

Como cualquier CPU, el esclavo inteligente guarda las propiedades de sus slots en la información de estado de módulo.

Véase también: SZL-ID W#16#xy91: información de estado de módulo (Página 7002).

Si, mediante "SALRM", se envía una alarma de diagnóstico, el sistema operativo del esclavo inteligente influye en la información de estado de módulo a nivel local de esclavo y en el LED SF, en función del bit 0 del byte 0 de AINFO (este bit se envía a la información de estado con el valor "Módulo defectuoso".). Sin embargo, en el esclavo I no se efectúa ningún registro en el búfer de diagnóstico, y no se inicia ningún OB de alarma de diagnóstico.

Coherencia de la información de estado de módulo entre el maestro DP y el esclavo I

A continuación se contemplan diversos escenarios y sus repercusiones en la información de estado del módulo:

- Recuperación de estación (provoca el inicio del OB 86 en el maestro DP y en el esclavo I)
  Afecta a la información de estado de módulo tanto en el esclavo I como en el maestro S7 (se anula "Módulo defectuoso"). Si, desde el punto de vista del usuario, después de la recuperación de la estación se producen fallos en el esclavo I, estos fallos deben comunicarse al maestro DP llamando "SALRM".
- Transición de STOP a RUN del maestro DP (provoca el inicio del OB 82 en el esclavo I)
  La información de estado del módulo en el esclavo I no se modifica. El maestro DP anula el bit "Fallo del módulo" en la correspondiente información de estado de módulo.
  Para garantizar la coherencia de la información de estado de módulo entre el maestro DP y el esclavo I en el modo compatible con S7, se debe reaccionar del modo siguiente en el esclavo I:
  - Por cada slot virtual sin errores, debe enviarse al maestro DP una alarma de diagnóstico saliente mediante "SALRM".
  - Para cada slot virtual con errores, debe enviarse al maestro DP una alarma de diagnóstico entrante mediante "SALRM".
- Transición de STOP a RUN del esclavo I (provoca el inicio del OB 82 en el maestro DP)
  La información de estado de módulo en el maestro DP permanece invariable, y la del esclavo I se anula (se anula "Módulo defectuoso".).
  Para garantizar la coherencia de la información de estado de módulo entre el maestro DP y el esclavo I, se debe reaccionar del modo siguiente en el esclavo I:
  - Por cada slot virtual sin errores, debe enviarse al maestro DP una alarma de diagnóstico saliente mediante "SALRM".
  - Para cada slot virtual con errores, debe enviarse al maestro DP una alarma de diagnóstico entrante mediante "SALRM".

Nota

Dado que "SALRM" funciona de modo asíncrono, las llamadas a "SALRM" en los OBs de arranque no se pueden finalizar, es decir, deben finalizarse en el programa cíclico.

Todas las diferencias mencionadas entre la información de estado de módulo en el maestro y el esclavo I solo pueden producirse en los slots que tengan asignadas alarmas de diagnóstico mediante "SALRM". Esto, a su vez. significa que las soluciones descritas solo son necesarias para dicho tipo de slots.

Parámetro ATYPE

Parámetro ATYPE

La tabla siguiente muestra, para cada uno de los valores admisibles de ATYPE , qué OB se inicia en el respectivo maestro DP y en qué modo DP se admite la correspondiente alarma.

ATYPE	Significado en la norma DPV1	OB correspondiente en el maestro DP S7	Modo DP	
			DP	Compatible con S7
1	Alarma de diagnóstico	OB de alarma de diagnóstico (OB82)	–	Sí
2	Alarma de proceso	OBs de alarma de proceso (OBs de 40 a 47)	–	Sí

Nota

La posibilidad de uso de los tipos de alarma puede limitarse desde el maestro DP además de lo indicado en la tabla anterior.

Dependencia del tipo de alarma respecto al estado operativo de un maestro S7

En un esclavo en modo compatible con S7 (operación en un maestro S7), solo pueden enviarse alarmas de proceso y de diagnóstico cuando el correspondiente maestro DP se encuentra en el estado operativo RUN (DP: Operate). Si el maestro DP se encuentra en el estado operativo STOP (DP: Clear), la alarma se retiene y "SALRM" devuelve la información de error W#16#80C8. La tarea de volver a enviar la alarma en un momento posterior corresponde exclusivamente al usuario.

Parámetros ASPEC

Parámetro ASPEC

Este parámetro muestra el estado de diagnóstico del slot virtual conforme a la norma. Por ello solo es posible asignar a ASPEC un valor distinto de cero en caso de enviar una alarma de diagnóstico.

Dado que en la información adicional de alarma de S7 de la alarma de diagnóstico (registro 0) existe una información entrante/saliente (ver Datos de diagnóstico>Byte principal 0 bit 0), se debe asignar el valor siguiente al bit 0 (fallo de módulo) del byte 0 de la información adicional de alarma:

ASPEC	Bit "Módulo defectuoso" en AINFO (Página 6794)
0	0
1	1

ASPEC	Bit "Módulo defectuoso" en AINFO (Página 6794)
2	0
3	1

Parámetro LEN

Parámetro LEN

En LEN se indica la longitud en bytes de la información adicional de alarma que se debe transferir. El rango de valores máximo admisible es de 0 a 16.

En la tabla siguiente se indica, para todos los tipos de alarma posibles, qué valores son admisibles para LEN en los distintos modos de un esclavo inteligente.

Tipo de alarma	DP	Compatible con S7
Alarma de diagnóstico	–	de 4 a 16
Alarma de proceso	–	4

En la tabla siguiente se describe qué reacción muestra "SALRM" cuando se asigna a LEN un valor distinto a la longitud de AINFO (Página 6794)BYTE.

Valor de LEN	Comportamiento de "SALRM"
<= Indicación de longitud de AINFO	"SALRM" envía una alarma al maestro DP. Se transfieren a la información adicional de alarma tantos bytes como se haya indicado en LEN.
Fuera del rango de valores admisible, p. ej. < 0 o > 16	"SALRM" no envía ninguna alarma. Información de error: W#16#80B1, STATUS[4]=B#16#FF
> Indicación de longitud de AINFO	"SALRM" envía una alarma al maestro DP. Se transfieren a la información adicional de alarma tantos bytes como se haya indicado en la indicación de longitud de AINFO. Información de error: W#16#00B1, STATUS[4]=Indicación de longitud de AINFO

Parámetros STATUS y ERROR

Parámetros STATUS y ERROR

El parámetro de salida STATUScontiene información de error. Si se interpreta como ARRAY[1 ... 4] OF BYTE, la información de error presenta la siguiente estructura:

Elemento de campo	Significado
STATUS[1]	• B#16#00: Sin error • B#16#C0: Error detectado por el esclavo I
STATUS[2], STATUS[3]	Equivale al parámetro de salida RET_VAL de las instrucciones
STATUS[4]	B#16#00 a excepción de algunos conflictos de longitud entre LEN y la longitud de AINFO (Página 6794). Estas excepciones se recogen en la tabla siguiente.

En la tabla siguiente se indican todas las informaciones de error específicas de "SALRM".

ERROR	STATUS[2,3] (W#16# ...)	Explicación
0	0000	La petición se ha ejecutado sin errores. Si LEN es < longitud de AINFO, solo se han transferido LEN bytes de la información adicional de alarma.
0	00B1	LEN > longitud de AINFO. La petición se ha ejecutado. La información adicional de alarma especificada en AINFO se ha transferido. STATUS[4] contiene la longitud de AINFO.
0	7000	Primera llamada con REQ=0 (ejecución en vacío). No se ha enviado ninguna alarma. BUSY tiene el valor "0".
0	7001	Primera llamada con REQ=1. Se ha lanzado la petición.BUSY tiene el valor "1".
0	7002	Llamada intermedia (REQ irrelevante). La alarma enviada no ha sido confirmada todavía por el maestro DP. BUSY tiene el valor "1".
1	8090	La dirección indicada en ID se encuentra fuera del área de direcciones admisible o no se ha configurado.
1	8091	• Se ha bloqueado la alarma mediante configuración. • La alarma no es admisible para este tipo de esclavo.
1	8092	Tipo de datos no admisible en AINFO (solo se admiten BYTE y BLOCK-DB)
1	8093	ID pertenece a un slot virtual desde el que no es posible efectuar solicitudes de alarma.
1	80B0	ASPEC • No corresponde al bit 0 del byte 0 de AINFO • Debe tener el valor 0 con el tipo de alarma utilizado • Se encuentra fuera del rango de valores admisible
1	80B1	LEN se encuentra fuera del rango de valores admisible. STATUS[4] contiene B#16#FF.
1	80B5	La llamada a "SALRM" en el maestro DP no es admisible.

ERROR	STATUS[2,3] (W#16# ...)	Explicación
1	80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.
1	80C5	En este momento la periferia descentralizada no está disponible (p. ej. fallo de estación)
1	80C8	La función no está autorizada en el estado operativo actual del maestro DP (el maestro DP es un maestro S7 y se encuentra en el estado operativo STOP).

Parámetro AINFO

Parámetro AINFO

El parámetro AINFO es el área de origen de la información adicional de alarma. Desde el punto de vista del esclavo inteligente, esta área se puede rellenar con cualquier valor. Sin embargo, si se utiliza un maestro DP de la gama S7, la información adicional de alarma enviada con la alarma debe ajustarse a las convenciones de S7.

Si se envía una **alarma de diagnóstico** (ATYPE (Página 6791) =1), el usuario es responsable de asignar un valor adecuado al registro 0 y, en su caso, al registro 1.

La tabla siguiente ofrece una propuesta de asignación conforme a S7. Se ha ajustado ya anticipadamente el bit "Módulo defectuoso" (ver arriba). La propuesta coincide con la asignación predeterminada (que se realiza después de POWER ON, después de una transición STOP-RUN del esclavo inteligente o después de la recuperación de una estación), excepto en el bit mencionado.

Número de registro	Asignación
0	B#16#01, 0B, 00, 00
1	Con modo compatible con S7: Registro 0 + 12 bytes con cero

Información adicional

Consulte también: Formato de los datos de diagnóstico (Página 6912).

PROFIBUS

DP_PRAL: Disparar alarma de proceso del esclavo DP normalizado

Descripción

Esta instrucción permite disparar una alarma de proceso desde el programa de usuario de un esclavo inteligente en el correspondiente maestro DP. Esto da lugar al inicio del OB 40 en el maestro DP.

Con el parámetro de entrada AL_INFO puede identificarse la causa de la alarma de proceso deseada. Esta identificación de alarma se transfiere al maestro DP y puede evaluarse en el OB 40 (variable OB40_POINT_ADDR).

La alarma de proceso solicitada se define de modo unívoco mediante los parámetros de entrada IOID y LADDR . Por cada área de direcciones configurada de la memoria de transferencia, se puede disparar una y solo una alarma de programa en un momento cualquiera.

Funcionamiento

"DP_PRAL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la solicitud de alarma de proceso, debe llamarse "DP_PRAL" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

La petición finaliza cuando ha finalizado el procesamiento del OB 40 en el maestro DP.

Nota

Si se utiliza el esclavo DP como esclavo normalizado, la petición finaliza cuando el telegrama de diagnóstico ha sido recogido por el maestro DP.

Identificación de una petición

Los parámetros de entrada IOID y LADDR definen una petición de manera unívoca.

Si se ha llamado la instrucción "DP_PRAL" en un esclavo DP y se ha vuelto a llamar la instrucción antes de que el maestro DP haya confirmado la alarma de proceso solicitada, el comportamiento posterior de la instrucción depende esencialmente de si la nueva llamada corresponde a la misma petición. Si los parámetros IOID y LADDR coinciden con una petición todavía no finalizada, la llamada se interpreta como llamada consecutiva independientemente del valor del parámetro AL_INFO , y en RET_VAL se registra el valor W#16#7002 .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DP_PRAL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ=1: Disparar alarma de proceso en el maestro DP correspondiente
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificación del área de direcciones en la memoria de transferencia (desde el punto de vista del esclavo DP): • B#16#00: El bit15 de LADDR indica si se trata de una dirección de entrada (bit15=0) o de salida (bit15=1). • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección de inicio del área de direcciones en la memoria de transferencia (desde el punto de vista del esclavo DP). Si se trata de un área perteneciente a un módulo mixto, debe indicarse la menor de las dos direcciones.
AL_INFO	Input	DWORD	I, Q, M, D, L o constante	Identificación de alarma Esta se asigna al OB 40 que debe iniciarse en el maestro DP correspondiente (variable OB40_POINT_ADDR). Si se utiliza un esclavo inteligente con un maestro externo, debe evaluarse el telegrama de diagnóstico en el maestro.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La alarma de proceso disparada no ha sido confirmada todavía por el maestro DP.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
7000	Primera llamada con REQ=0. No hay ninguna solicitud de alarma de proceso activa; BUSY tiene el valor "0".

Código de error (W#16#...)	Explicación
7001	Primera llamada con REQ=1. Se ha enviado una solicitud de alarma al maestro DP; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): La alarma disparada no ha sido confirmada todavía por el maestro DP; BUSY tiene el valor "1".
8090	La dirección de inicio del área de direcciones en la memoria de transferencia es errónea
8091	Alarma bloqueada mediante configuración.
8093	Mediante el par de parámetros IOID y LADDR se accede a un módulo desde el que no es posible solicitar alarmas de proceso.
80B5	La llamada en el maestro DP no es admisible.
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.
80C5	En este momento la periferia descentralizada no está disponible (p. ej. fallo de estación).
80C8	La función no está autorizada en el actual estado operativo del maestro DP.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

DPSYC_FR: Sincronizar esclavos DP / Congelar entradas

Descripción

Esta instrucción permite sincronizar uno o varios grupos de esclavos DP.

Para ello debe enviarse a los grupos en cuestión uno de los siguientes comandos de control o una combinación de ellos:

- SYNC (emisión y congelación simultáneas de estados de salida en los esclavos DP)
- UNSYNC (anula el comando de control SYNC)
- FREEZE (congelación de estados de entrada en los esclavos DP y lectura de las entradas congeladas)
- UNFREEZE (anula el comando de control FREEZE)

Antes de enviar los comandos de control mencionados, deben haberse dividido en grupos, mediante configuración, los esclavos DP. Es necesario saber qué esclavo DP está asignado a cada grupo y con qué número, así como las características de cada uno de los grupos en lo que respecta al comportamiento de SYNC/FREEZE.

Nota

Debe tenerse en cuenta que los comandos de control SYNC y FREEZE mantienen su validez incluso en caso de rearranque completo o arranque en frío.

También debe tenerse en cuenta que en cada momento solo puede estar lanzada una petición SYNC/UNSYNC y solo una petición FREEZE/UNFREEZE.

Funcionamiento

"DPSYC_FR" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "DPSYC_FR" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Identificación de una petición

Si se ha lanzado una petición SYNC/FREEZE y se ha vuelto a llamar "DPSYC_FR" antes de que finalice, el comportamiento posterior de la instrucción depende esencialmente de si la nueva llamada corresponde a la misma petición. Si los parámetros de entrada LADDR, GROUP y MODE coinciden entre sí, la llamada se considera una llamada consecutiva.

Escribir en salidas de módulos DP

La escritura en las salidas de los módulos DP se lanza mediante

- Comandos de transferencia a la periferia DP,
- La escritura de la memoria imagen de proceso de las salidas en los módulos (mediante el sistema operativo al final del OB 1 o mediante llamada de la instrucción "UPDAT_PO (Página 6731)"),
- La llamada de la instrucción "DPWR_DAT (Página 6778)".

En el caso normal, el maestro DP transfiere los datos de salida cíclicamente (en el ciclo del bus PROFIBUS DP) a las salidas de los esclavos DP.

Si se desea transferir a las salidas al proceso de manera exactamente simultánea determinados datos de salida que pueden estar repartidos entre varios esclavos, debe enviarse al maestro DP correspondiente el comando de control SYNC con ayuda de la instrucción "DPSYC_FR".

¿Qué efecto produce SYNC?

Con el comando de control SYNC se ponen en modo de sincronización los esclavos DP de los grupos especificados, es decir, el maestro DP transfiere los datos de salida actuales y ordena a los esclavos DP implicados que congelen las salidas. En los siguientes telegramas de salida, los esclavos DP guardan los datos de salida en un búfer interno; el estado de las salidas permanece invariable.

Después de cada comando de control SYNC, los esclavos DP de los grupos seleccionados depositan los datos de salida de su búfer interno en las salidas al proceso.

Las salidas no se vuelven a actualizar cíclicamente hasta que el usuario envía el comando de control UNSYNC con ayuda de la instrucción "DPSYC_FR" .

Nota

Si, en el momento en que se envía el comando de control, algún esclavo DP del grupo o grupos seleccionados no se encuentra conectado a la red o presenta un fallo, dichos esclavos tampoco pasan al modo SYNC. Este hecho no se notifica mediante el valor de retorno de la instrucción.

Leer entradas de módulos DP

Se leen los datos de entrada de los módulos DP

- Con comandos de carga a la periferia DP,
- Al actualizar la memoria imagen de proceso de las entradas (mediante el sistema operativo al final del OB 1 o mediante llamada de la instrucción "UPDAT_PI (Página 6729)"),
- Llamando la instrucción "DPRD_DAT (Página 6776)".

En el caso normal, el maestro DP recibe estos datos de entrada cíclicamente (en el ciclo del bus PROFIBUS DP) desde sus esclavos DP y los pone a disposición de la CPU.

Si se desea leer desde el proceso de manera exactamente simultánea determinados datos de entrada que pueden estar repartidos entre varios esclavos, debe enviarse al maestro DP correspondiente el comando de control FREEZE con ayuda de la instrucción "DPSYC_FR".

¿Qué efecto produce FREEZE?

Con el comando de control FREEZE se ponen en modo de congelación los esclavos DP afectados, es decir, el maestro DP ordena a los esclavos DP afectados que congelen el estado actual de las entradas. A continuación transfiere los datos congelados al área de entrada de la CPU.

Después de cada comando de control FREEZE, los esclavos DP vuelven a congelar el estado de las entradas.

El maestro DP no vuelve a recibir cíclicamente el estado actual de las entradas hasta que el usuario envía el comando de control UNFREEZE con ayuda de la instrucción "DPSYC_FR".

Nota

Si, en el momento en que se envía el comando de control, algún esclavo DP del grupo o grupos seleccionados no se encuentra conectado a la red o presenta un fallo, dichos esclavos tampoco pasan al modo Freeze. Este hecho no se notifica mediante el valor de retorno de la instrucción.

Coherencia de datos

Debido al funcionamiento asíncrono y a la posibilidad de interrupción por clases de prioridad superiores, al utilizar la instrucción "DPSYC_FR" debe asegurarse que las memorias imagen de proceso sean coherentes con las entradas y salidas reales de la periferia.

Para garantizarlo, debe aplicarse una de las reglas de coherencia que se mencionan a continuación:

- Defina memorias imagen parcial de proceso adecuadas (solo posible con S7-400) para las salidas "SYNC" y las entradas "FREEZE". Llame la instrucción "UPDAT_PO (Página 6731)" inmediatamente antes de cada primera llamada de una petición SYNC. Llame la instrucción "UPDAT_PI (Página 6729)" inmediatamente después de cada última llamada de una petición FREEZE.
- Alternativamente: Para las salidas afectadas por una petición SYNC, y para las entradas afectadas por una petición FREEZE, utilice solo accesos directos a la periferia. Mientras haya una petición SYNC en curso, no es posible escribir en las salidas afectadas, ni tampoco leer las entradas afectadas mientras está en curso una petición FREEZE.

Uso de DPWR_DAT y DPRD_DAT

Si se utiliza la instrucción "DPWR_DAT (Página 6778)", ésta debe estar finalizada antes de que se envíe una petición SYNC para las salidas en cuestión.

Si se utiliza la instrucción "DPRD_DAT (Página 6776)", ésta debe estar finalizada antes de que se envíe una petición FREEZE para las entradas en cuestión.

Arranque y "DPSYC_FR"

El envío de los comandos de control SYNC y FREEZE en los OBs de arranque es responsabilidad exclusiva del usuario.

Si se desea que las salidas de uno o varios grupos empiecen a funcionar en el modo SYNC ya desde el inicio del programa de usuario, deben inicializarse esas salidas en el arranque y ejecutarse por completo la instrucción "DPSYC_FR" con el comando de control SYNC.

Si se desea que las entradas de uno o varios grupos empiecen a funcionar en el modo FREEZE ya desde el inicio del programa de usuario, debe ejecutarse por completo para esas entradas la instrucción "DPSYC_FR" con el comando de control FREEZE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPSYC_FR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control disparado por nivelREQ=1: Lanzar la petición SYNC/FREEZE
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección lógica del maestro DP

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
GROUP	Input	BYTE	I, Q, M, D, L o constante	Selección de grupo Bit 0 = 1: Grupo 1 seleccionado Bit 1 = 1: Grupo 2 seleccionado : Bit 7 = 1: Grupo 8 seleccionado Pueden seleccionarse varios grupos por petición. El valor B#16#0 no está permitido.
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición (codificación según EN 50 170 Volume 2, PROFIBUS) Bit 0: Reservado (valor 0) Bit 1: Reservado (valor 0) Bit 2: • = 1: UNFREEZE se ejecuta • = 0: Sin significado Bit 3: • = 1: FREEZE se ejecuta • = 0: Sin significado Bit 4: • = 1: UNSYNC se ejecuta • = 0: Sin significado Bit 5: • = 1: SYNC se ejecuta • = 0: Sin significado Bit 6: Reservado (valor 0) Bit 7: Reservado (valor 0) Valores posibles: • Con una y solo una ID por petición: – B#16#04 (UNFREEZE) – B#16#08 (FREEZE) – B#16#10 (UNSYNC) – B#16#20 (SYNC) • Con varias IDs por petición: – B#16#14 (UNSYNC, UNFREEZE) – B#16#18 (UNSYNC, FREEZE) – B#16#24 (SYNC, UNFREEZE) – B#16#28 (SYNC, FREEZE)
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Después de cada pasada del bloque debe evaluarse RET_VAL .
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La petición de SYNC/FREEZE todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Nota

Si accede a esclavos DPV1, el maestro DP puede transferir la información de error de dichos esclavos a la instrucción. Encontrará una descripción de esta información de error en STATUS[3], parámetro STATUS (Página 6748).

Código de error (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
7000	Primera llamada con REQ=0. La petición definida mediante LADDR, GROUP y MODE no está activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1. La petición definida mediante LADDR, GROUP y MODE se ha lanzado; BUSY tiene el valor 1.
7002	Llamada intermedia (REQ irrelevante). La petición SYNC/FREEZE activada todavía está en curso; BUSY tiene el valor 1.
8090	El módulo seleccionado mediante LADDR no es un maestro DP.
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR (configuración o versión del maestro DP).
8094	Parámetros GROUP erróneos
8095	Parámetros MODE erróneos
80B0	El grupo seleccionado mediante GROUP no está configurado.
80B1	El grupo seleccionado mediante GROUP no está asignado a esta CPU.
80B2	La petición SYNC especificada mediante MODE no está permitida en el grupo seleccionado mediante GROUP.
80B3	La petición FREEZEespecificada mediante MODE no está autorizada en el grupo seleccionado mediante GROUP .
80C2	Falta de recursos temporal del maestro DP: En este momento el maestro DP está procesando el máximo posible de peticiones para una CPU.
80C3	Esta petición SYNC/UNSYNCno se puede activar en este momento, ya que en cada momento determinado solo puede estar lanzada una petición SYNC/UNSYNC. Compruebe el programa de usuario.
80C4	Esta petición FREEZE/UNFREEZEno se puede activar en este momento, ya que en cada momento determinado solo puede estar lanzada una petición FREEZE/UNFREEZE. Compruebe el programa de usuario.
80C5	Cortocircuito en la proximidad inmediata de la interfaz DP
80C6	Interrupción de la petición debido a expulsión de la periferia por la CPU
80C7	Interrupción de la petición debido a rearranque completo (en caliente) o arranque en frío del maestro DP
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

DPNRM_DG: Leer datos de diagnóstico de un esclavo DP

Descripción

La instrucción "DPNRM_DG" lee los datos de diagnóstico actuales de un esclavo DP de la forma establecida por EN 50 170 Volume 2, PROFIBUS. Los datos leídos se introducen en el área de destino determinada por RECORD tras la correcta transferencia de los datos.

La estructura fundamental del diagnóstico de esclavo se recoge en la siguiente tabla, y la información más detallada, en los manuales de los esclavos DP.

Byte	Significado
0	Estado de estación 1
1	Estado de estación 2
2	Estado de estación 3
3	Número de estación maestra
4	Identificación de fabricante (high byte)
5	Identificación de fabricante (low byte)
6 ...	Otros diagnósticos específicos de esclavo

Funcionamiento

La lectura se realiza de forma asíncrona, es decir, que se puede prolongar a lo largo de varias llamadas. El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Llamada

Para iniciar la lectura, debe ajustarse a "1" el parámetro de entrada REQ al llamar la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DPNRM_DG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: solicitud de lectura
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección de diagnóstico configurada del esclavo DP Nota: La dirección se debe indicar en formato hexadecimal; p. ej. la dirección de diagnóstico 1022 significa: LADDR:=W#16#3FE.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Si no aparece ningún error, en RET_VAL se especifica la longitud de los datos realmente transferidos.
RECORD	Output	ANY	I, Q, M, D, L	Área de destino para los datos de diagnóstico leídos. Solo se admite el tipo de datos BYTE . La longitud mínima del registro que se debe leer o del área de destino es de 6. La longitud máxima del registro que se debe leer es de 240; en esclavos normalizados cuyo número de datos de diagnóstico normalizados sea superior a 240 bytes y tenga un máximo de 244 bytes, los primeros 240 bytes se transfieren al área de destino y el correspondiente bit de desbordamiento se activa en los datos. Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La lectura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Encontrará información sobre la conversión de tipos de datos en los distintos lenguajes de programación en "Sinopsis de la conversión de tipos de datos (Página 2373)".

Parámetro RECORD

La CPU valora la longitud real de los datos de diagnóstico leídos:

Si la indicación de longitud de RECORD

- es inferior al número de datos proporcionados, los datos se rechazan y en RET_VAL se deposita la correspondiente información de error.
- es superior o igual al número de datos proporcionados, los datos se aceptan en el área de destino y en RET_VAL se deposita la longitud real como valor positivo.

Nota

Hay que asegurarse de que los parámetros actuales de RECORD coincidan en todas las llamadas pertenecientes a una petición.

Una petición está claramente establecida por el parámetro de entrada LADDR.

Parámetro RET_VAL

- Si se ha producido un error durante la ejecución de la función, el valor de respuesta contendrá un código de error,
- Si durante la transferencia no se ha producido ningún error, RET_VAL contiene la longitud de los datos leídos en bytes como número positivo.

Nota

En un esclavo DP, el número de datos leídos depende de su estado de diagnóstico.

Para evaluar la información de error del parámetro RET_VAL , véase Evaluación de errores con el parámetro de salida RET_VAL (Página 5598). Allí encontrará también la información general de error de las instrucciones. La información específica de error "DPNRM_DG" es una parte de la información de error de "RD_REC (Página 6769)".

Esclavos normalizados con datos de diagnóstico de más de 240 bytes

En esclavos normalizados con un número de datos de diagnóstico normalizados entre 241 y 244 bytes, debe tenerse en cuenta los siguiente:

Si la indicación de longitud de RECORD

- es inferior a 240 bytes, los datos se rechazan y en RET_VAL se deposita la correspondiente información de error.
- es superior o igual a 240 bytes, los primeros 240 bytes de los datos de diagnóstico normalizados se transfieren al área de destino y en los datos se activa el correspondiente bit de desbordamiento.

Recursos del sistema con S7-400

En S7-400, al llamar "DPNRM_DG" para una petición no procesada en ese momento, se asignan los recursos (espacio de memoria) de la CPU. Es posible llamar "DPNRM_DG" sucesivamente a intervalos breves para varios esclavos DP, siempre que no se rebase la cantidad máxima de peticiones "DPNRM_DG" "simultáneas" activas que la CPU es capaz de procesar.

Si existen varias peticiones activas "simultáneas", se garantiza que todas las peticiones se ejecuten y que no se produzcan interferencias recíprocas.

En el momento en que se alcanza el límite de los recursos del sistema, se envía una notificación mediante RET_VAL. En tal caso debe repetirse la petición.

Consulte también

Sinopsis de la conversión de tipos de datos (Página 2373)

DP_TOPOL: Determinar topología para sistema maestro DP

Descripción

Esta instrucción permite lanzar la determinación de topología para un sistema maestro seleccionado. Al llamar esta instrucción, se accede a todos los repetidores de diagnóstico conectados a un sistema maestro DP.

Nota

La determinación de topología puede efectuarse en cada momento concreto solo para un sistema maestro DP.

La determinación de topología es imprescindible para poder visualizar con detalle la ubicación de los eventuales errores de cable. Después del montaje del sistema maestro DP y de cada modificación de su estructura física, debe efectuarse de nuevo la determinación de topología con la instrucción "DP_TOPOL".

Se considera que se ha modificado la estructura física cuando:

- Se modifican longitudes de cables
- Se añaden o eliminan estaciones o componentes con función de repetidor
- Se modifican direcciones de estaciones

Si un repetidor de diagnóstico notifica un error, "DP_TOPOL" escribe en las salidas DPR y DPRI a lo largo de la duración de un paso de "DP_TOPOL". Si varios repetidores de diagnóstico del sistema maestro DP seleccionado notifican errores, "DP_TOPOL" envía a DPR y DPRI información acerca del primer repetidor de diagnóstico que ha notificado errores. La información de diagnóstico completa se puede leer con la instrucción "DPNRM_DG (Página 6803)" o en la PG. Si ningún repetidor de diagnóstico notifica errores, las salidas DPR y DPRI adquieren el valor NULL.

Si, después de producirse un error, se desea volver a lanzar una determinación de topología, en primer lugar debe restablecerse "DP_TOPOL". Para hacerlo, debe llamarse "DP_TOPOL" con REQ=0 y R=1.

Funcionamiento

"DP_TOPOL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la determinación de topología de bus, debe llamarse "DP_TOPOL" con REQ=1 . Para interrumpir la operación, debe llamarse "DP_TOPOL" con R=1 .

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Nota

La determinación de topología puede durar varios minutos.

Identificación de una petición

El parámetro de entrada DP_ID define una petición de manera unívoca.

Si se ha llamado "DP_TOPOL" y se vuelve a llamar la misma instrucción antes de que finalice la determinación de topología, el comportamiento posterior de la instrucción depende de si la nueva llamada corresponde a la misma petición: Si el parámetro DP_ID coincide con una petición todavía no finalizada, la llamada se interpreta como llamada consecutiva, y en RET_VALse registra el valor W#16#7002. En cambio, si se trata de una petición diferente, la CPU la rechaza.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DP_TOPOL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ=1: Lanzar determinación de topología
R	Input	BOOL	I, Q, M, D, L	R=1: Interrumpir la determinación de topología
DP_ID	Input	INT	I, Q, M, D, L o constante	ID del sistema maestro DP cuya topología se debe determinar
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La determinación de topología todavía no ha finalizado.
DPR	Output	BYTE	I, Q, M, D, L	Dirección PROFIBUS del repetidor de diagnóstico que notifica errores
DPRI	Output	BYTE	I, Q, M, D, L	Segmento de medición del repetidor de diagnóstico que notifica errores: • Bit 0 = 1: Fallos temporales en el segmento DP2 • Bit 1 = 1: Fallos permanentes en el segmento DP2 • Bit 4 = 1: Fallos temporales en el segmento DP3 • Bit 5 = 1: Fallos permanentes en el segmento DP3

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Dentro de la información de error "verdadera" (códigos de error W#16#8xyz) de la tabla siguiente, debe distinguirse entre dos casos:

- Errores temporales (códigos de error de W#16#80A2 a 80A4, 80C3, 80C5):
  En este tipo de errores es posible que el error se elimine sin intervención del usuario, es decir, puede bastar con llamar de nuevo "DP_TOPOL" (varias veces si es necesario). Ejemplo de error temporal: En este momento los recursos necesarios están ocupados (W#16#80C3).
- Errores permanentes (códigos de error de W#16#8082, 80B0, 80B2):
  Los errores de este tipo no se pueden eliminar sin intervención del usuario. Solo tiene sentido volver a llamar "DP_TOPOL" una vez que se haya eliminado el error. Ejemplo de error permanente: El maestro DP o la CPU no soportan este servicio. (W#16#80B0).

Código de error (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
7000	Primera llamada con REQ=0. No se ha lanzado ninguna determinación de topología. BUSY tiene el valor "0".
7001	Primera llamada con REQ=1. Se ha lanzado la petición de determinación de topología. BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): La determinación de topología todavía está procesándose. BUSY tiene el valor "1".
7010	Se ha intentado interrumpir la determinación de topología. Sin embargo, no hay ninguna petición en curso con la DP_IDindicada. BUSY tiene el valor "0".
7011	Primera llamada con R=1. Se ha lanzado la interrupción de la determinación de topología; BUSY tiene el valor "1".
7012	Llamada intermedia: La interrupción de la determinación de topología todavía está procesándose. BUSY tiene el valor "1".
7013	Última llamada: Se ha interrumpido la determinación de topología. BUSY tiene el valor "0".
8082	No está configurado ningún maestro DP con la DP_ID indicada.
80A2	Error en determinación de topología; encontrará información más detallada en los parámetros de salida DPR y DPRI.
80A3	Error en determinación de topología: El tiempo de vigilancia ha finalizado (timeout).
80A4	Fallo de comunicación en el bus K
80B0	El maestro DP o la CPU no soportan este servicio.
80B2	Error en determinación de topología: No se ha detectado ningún repetidor de diagnóstico en el sistema maestro DP seleccionado.
80C3	En este momento los recursos necesarios están ocupados. Causa posible: Ha iniciado una segunda determinación de la topología (no está permitido determinar más de una topología en un determinado instante).
80C5	El sistema maestro DP no está disponible en este momento.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

ASi

ASi_3422: Controlar comportamiento del maestro ASi

Descripción

La instrucción "ASi_3422" se requiere para utilizar el modo de operación "Modo de funcionamiento avanzado" con el CP 342-2. El modo de funcionamiento avanzado permite controlar por completo el comportamiento del maestro a través del programa de usuario. El acceso a las entradas y salidas se realiza, como es habitual, en el modo de funcionamiento estándar del CP 342-2.

Las llamadas a comandos enviadas al CP342-2 se realizan desde el programa de usuario, a través de la instrucción. Para ello es necesario especificar una llamada de comando en un búfer de transmisión e iniciar la petición.

La instrucción "ASi_3422" transfiere la llamada de comando al CP342-2. Al finalizar la petición, se transfiere el estado de la petición y se ponen a disposición los posibles datos de respuesta en un búfer de recepción.

Llamada

La instrucción debe llamarse cíclicamente para cada CP342-2 existente. Por cada CP342-2 sólo puede ejecutarse una petición simultáneamente. Una petición en curso no puede interrumpirse y no está sometida a vigilancia de tiempo por la instrucción. La instrucción "ASi_3422" no se puede ejecutar con posibilidad de interrupción. Por ello las llamadas no se pueden programar en niveles de ejecución que se interrumpan recíprocamente (p. ej. con llamada en el OB 1 y en el OB 35).

Defina el procesamiento del comando en el programa de usuario del modo siguiente:

1. En caso de rearranque del programa de usuario S7, debe llamarse "ASi_3422" una sola vez con el valor de parámetro STARTUP = TRUE.
2. En el programa de usuario, especifique la llamada del comando en un búfer de transmisión. Este búfer de transmisión se transfiere con el parámetro de llamada SEND.
   Encontrará más información sobre la interfaz de comandos del CP y los comandos del esclavo AS-i en la documentación de hardware del procesador de comunicaciones.
3. Según el tipo de comando, se requiere un búfer de respuesta. Este búfer de respuesta se transfiere con el parámetro de llamada RECV. Con esta interfaz no se requiere el búfer de respuesta para información de estado.
4. Active la petición mediante el parámetro ACT=1.
5. A continuación, consulte los parámetros DONE, ERROR y STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ASi_3422":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
ACT	Input	BOOL	I, Q, M, D, L	La ejecución del comando por la función se dispara por nivel, es decir, mientras ACT = 1, se inicia la ejecución del comando, en caso de que no haya una llamada en ejecución.
STARTUP	Input	BOOL	I, Q, M, D, L	Un arranque de la CPU se notifica a la instrucción mediante STARTUP = 1. Después de la primera ejecución de la función, el usuario debe desactivar STARTUP.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección inicial de un módulo La dirección inicial de un módulo, en módulos de señal, se determina de acuerdo con los datos sobre asignación de dirección orientada a slot.
SEND	Input	ANY	I, Q, M, D, L	Búfer de transmisión El parámetro remite a un área de memoria en la que el usuario debe especificar el comando, p. ej.: P#DB20.DBX 20.0 Byte 10.
RECV	Input	ANY	I, Q, M, D, L	Búfer de recepción Este búfer solo es relevante para comandos que devuelven datos de respuesta. El parámetro remite a un área de memoria en la que se guarda una respuesta al comando. La longitud del área de datos parametrizada aquí es irrelevante, p. ej.: P#DB30.DBX 20.0 Byte 1
DONE	Output	BOOL	Q, M, D, L	Con DONE = 1 se señaliza "Petición finalizada sin error".
ERROR	Output	BOOL	Q, M, D, L	Con ERROR = 1 se señaliza "Petición finalizada con error".
STATUS	InOut	DWORD	M, D	1. Palabra: Estado de la petición/código de error (ver la tabla siguiente); con "Petición finalizada con error" se genera un código de error para obtener una descripción más detallada del error. 2. Palabra: Es utilizada por la instrucción para fines internos y no puede modificarse. Para llamadas de instrucción en diferentes direcciones lógicas (LADDR) deben asignarse diferentes palabras dobles para el parámetro STATUS.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro LADDR

La dirección de inicio de esta área de memoria se determina mediante el slot del CP 342-2.

El CP 342-2 ocupa 16 bytes de entrada y 16 bytes de salida en el área de direcciones E/S del autómata programable S7 (área analógica del AS). De los 16 bytes del área de direcciones del CP 342-2 se asignan 31 x 4 bits para datos del esclavo AS-i. Los 4 bits restantes están reservados para aplicaciones posteriores.

Parámetros DONE, ERRORy STATUS

Si durante el procesamiento de la función se produce un error, además de la información ya mencionada en ERROR y STATUS se adjunta un "0" en el bit BR. La consulta del bit BR se realiza de modo distinto en los programas de usuario KOP y AWL:

- KOP: Consulta a través del parámetro de salida ENO
- AWL: Consulta directa del bit BR

La tabla siguiente contiene las posibles indicaciones de la 1ª palabra de STATUS en función de DONE y ERROR.

DONE	ERROR	STATUS	Significado
0	0	8181	La petición está en curso.
1	0	0000	Petición finalizada sin error.
0	1	8090	Dirección no válida en el parámetro LADDR.
0	1	80A0	Acuse negativo al leer el módulo.
0	1	80A1	Acuse negativo al escribir en el módulo.
0	1	80B0	El módulo no conoce este registro.
0	1	80B1	La longitud del registro indicada es errónea.
0	1	80C0	No se puede leer el registro.
0	1	80C1	El registro indicado está siendo procesado en este momento.
0	1	80C2	Hay un atasco de peticiones.
0	1	80C3	Recursos (memoria) ocupados.
0	1	80C4	Error de comunicación
0	1	8182	Significado: Identificación después de rearranque con STARTUP=TRUE (sin errores).
0	1	8184	Tipo de datos del operando formal RECV, no admisible.
0	1	8381	Dirección del esclavo errónea
0	1	8382	Esclavo no activado (no en LAS).
0	1	8383	Error en AS-Interface
0	1	8384	Comando (en el estado del CP) no admisible.
0	1	8385	El esclavo 0 existe.
0	1	83A1	No se ha encontrado en la AS-Interface el esclavo cuya dirección se desea cambiar.
0	1	83A2	El esclavo 0 existe.
0	1	83A3	El esclavo con la nueva dirección ya existe en la AS-Interface.
0	1	83A4	No es posible borrar la dirección del esclavo.
0	1	83A5	No es posible ajustar la dirección del esclavo.
0	1	83A6	No es posible guardar de modo permanente la dirección del esclavo.
0	1	83F8	Número de petición desconocido.
0	1	83F9	Error de EEPROM
0	1	8F22	Error de longitud de área al leer un parámetro.

DONE	ERROR	STATUS	Significado
		8F23	Error de longitud de área al escribir un parámetro. Este código de error indica que un parámetro se encuentra total o parcialmente fuera del área del operando o que la longitud de un mapa de bits en un parámetro ANY no es divisible por 8.
0	1	8F24	Error de área al leer un parámetro.
		8F25	Error de área al escribir un parámetro. Este código de error indica que un parámetro se encuentra en un área que no es admisible para una función del sistema.
0	1	8F28	Error de alineación al leer un parámetro.
		8F29	Error de alineación al escribir un parámetro. Este código de error indica que la referencia a un parámetro es un operando cuya dirección de bit es distinta de "0".
0	1	8F30	El parámetro se encuentra en el DB global protegido contra escritura.
		8F31	El parámetro se encuentra en el DB de instancia protegido contra escritura. Este código de error indica que un parámetro se encuentra en un bloque de datos protegido contra escritura.
0	1	8F32	El parámetro contiene un número de DB demasiado alto
0	1	8F3A	El parámetro contiene el número de un DB que no está cargado.
0	1	8F42	Se ha producido un error de acceso mientras el sistema intentaba leer un parámetro del área de periferia de las entradas.
		8F43	Se ha producido un error de acceso mientras el sistema intentaba escribir un parámetro en el área de periferia de las salidas.
0	1	8F44	Este código de error indica que se ha denegado el acceso de lectura a un parámetro.
		8F45	Este código de error indica que se ha denegado el acceso de escritura a un parámetro.
0	1	8F7F	Error interno

ASI_CTRL: controlar comportamiento del maestro ASi

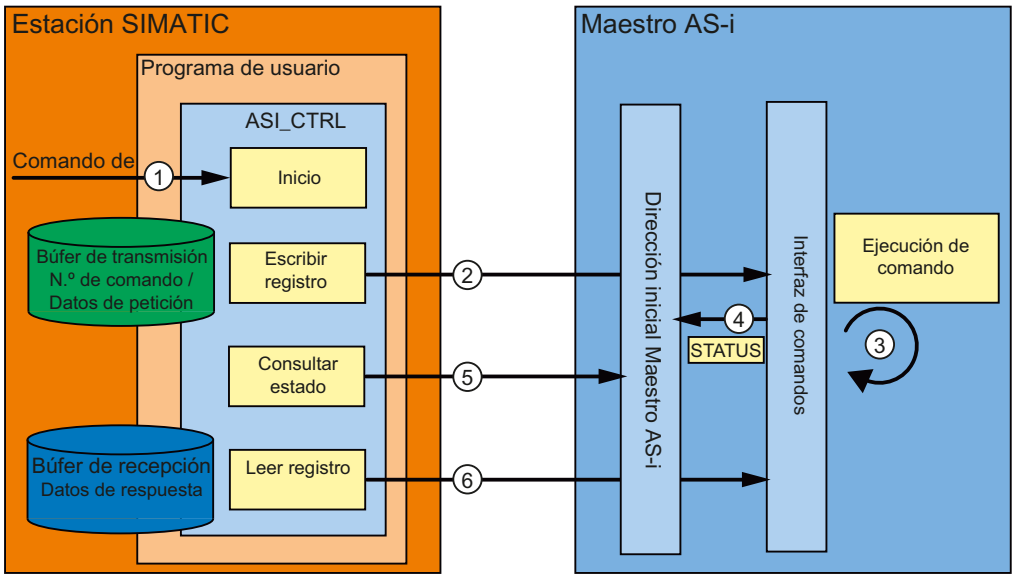
Descripción ASI_CTRL

Descripción

La instrucción "ASI_CTRL" permite controlar el comportamiento del maestro AS-i a través del programa de usuario del PLC. La instrucción procesa automáticamente el protocolo del comando y permite la parametrización de maestros AS-i SIMATIC así como la lectura de datos de información. Las posibilidades disponibles y el funcionamiento de la interfaz de comandos se describen en el manual del maestro AS-i.

Se soportan maestros AS-i con conexión centralizada y maestros AS-i descentralizados vía PROFIBUS DP. También son posibles combinaciones con PROFINET IO (ejemplo: IE/PB Link PN IO).

La figura siguiente muestra esquemáticamente la funcionalidad de la instrucción "ASI_CTRL":



① Inicio del procesamiento en el parámetro REQ.

② El programa transmite al maestro AS-i el comando deseado mediante la instrucción "RDREC".

③ El maestro AS-i ejecuta el comando.

④ El estado actual del maestro AS-i se deposita en el área de entrada de datos binarios (dirección base lógica).

⑤ La instrucción "ASI_CTRL" consulta cíclicamente los 4 bits de estado y los evalúa.

⑥ Si ha terminado la ejecución de comandos, la petición de comandos concluye mediante "RDREC". En el campo de datos de "RDREC" se encuentran, para cada comando, los datos de respuesta del mismo u otra información de estado.

Diferencias en la llamada de comandos con IE/AS-i Link y CPs o DP/AS-i Links

Existen diferencias sustanciales en el procedimiento de intercambio de comandos entre un controlador y un maestro AS-i.

- Con **IE/AS-i Link** (PROFINET) se utiliza la interfaz de registros. Los distintos comandos son llamados por diferentes números de registro mediante "Escribir registro" (instrucción "WRREC") o "Leer registro" (instrucción "RDREC").
- Los **CPs S7-300** y **DP/AS-i Links** (PROFIBUS) utilizan la interfaz de comandos. Todos los comandos son llamados por el número de registro 2 mediante una pareja de instrucciones "Escribir registro" (instrucción "WRREC") y "Leer registro" (instrucción "RDREC"). El tipo de comando lo define el contenido de datos al efectuarse la petición de escritura.

Cambios respecto de la instrucción "ASi_3422"

La instrucción "ASI_CTRL" es una versión revisada de la instrucción "ASi_3422 (Página 6809)" y presenta mejoras de funcionalidad y compatibilidad. En concreto, se han efectuado las modificaciones siguientes:

- Para escribir y leer registros de diagnóstico, las instrucciones "WR_REC (Página 6774)" y "RD_REC (Página 6769)" han sido sustituidas por las instrucciones "RDREC (Página 6735)" y "WRREC (Página 6738)". Sus funciones son idénticas, pero soportan la transferencia de datos vía PROFINET IO.
- El tipo de bloque de la instrucción ha cambiado de función (FC) a bloque de función (FB). "ASI_CTRL" dispone de un bloque de datos instancia y es apto para multiinstancia.
- La denominación de los parámetros formales de "ASI_CTRL" es conforme con los bloques de sistema SIMATIC. Desaparece el parámetro de entrada STARTUP. La definición del parámetro STATUS se ha basado en las instrucciones "RDREC (Página 6735)" y "WRREC (Página 6738)". Además, se han adaptado las identificaciones de estado para el parámetro DONE y para el parámetro BUSY nuevo.

Funcionamiento de la instrucción "ASI_CTRL"

La instrucción "ASI_CTRL" es un bloque de función asíncrono, de modo que la ejecución se prolonga a lo largo de varias llamadas.

- Una petición se inicia con REQ = TRUE.
- Mediante los parámetros de salida BUSY y los dos bytes centrales del parámetro de salida STATUS se muestra el estado de la petición.
- Mientras se procesa la petición, está activado el parámetro BUSY. En la primera llamada, STATUS contiene el valor 00700100_H. En todas las llamadas sucesivas pertenecientes a esta petición, contiene el valor 00700200_H. Al concluir la petición, el resultado se indica en los parámetros DONE o ERROR.
  - Si no se ha producido ningún error, se activa DONE. En peticiones con datos de respuesta del maestro AS-i, dichos datos se ponen a disposición en el búfer de recepción indicado. En este caso, además, en el parámetro STATUS se muestra la cantidad de datos proporcionados en bytes. En peticiones sin datos de respuesta, en STATUS se registra el valor 00000000_H.
  - Si durante el procesamiento de la petición se presenta un error, se activa ERROR. En ese caso, el contenido del búfer de recepción no es válido. Para definir con más precisión el error ocurrido, en el parámetro STATUS se registra un código de error.

Número de llamadas de comandos

Si se utiliza la instrucción "ASI_CTRL" para efectuar la petición de comandos, no deben emitirse simultáneamente otros comandos al mismo maestro AS-i mediante "RDREC (Página 6735)" y "WRREC (Página 6738)". Lo mismo se aplica a las llamadas múltiples de la instrucción en el mismo maestro AS-i.

La instrucción "ASI_CTRL" no se puede ejecutar con posibilidad de interrupción. Por ello, las llamadas no se pueden programar en niveles de ejecución que se interrumpan recíprocamente (p. ej. con llamada en el OB 1 y en el OB 35).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ASI_CTRL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Con REQ = TRUE se inicia una petición nueva siempre que no se esté ejecutando otra. No se realiza ninguna evaluación de flancos.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección inicial del maestro AS-i en el área de direcciones S7 (dirección base lógica). La dirección inicial se determina en la configuración de hardware al configurar el maestro.
SD	Input	ANY	I, Q, M, D, L	Búfer de transmisión El parámetro remite a un área de memoria en la que debe especificarse el comando (véase "Comandos AS-i (Página 6816)"). Ejemplo: P#DB101.DBX 0.0 BYTE 223
RD	Input	ANY	I, Q, M, D, L	Búfer de recepción Este búfer solo es relevante para comandos que devuelven datos de respuesta. El parámetro remite a un área de memoria en la que se guarda una respuesta al comando (véase "Comandos AS-i (Página 6816)"). Ejemplo: P#DB102.DBX 224.0 BYTE 221
DONE	Output	BOOL	Q, M, D, L	DONE = TRUE: petición finalizada sin errores.
BUSY	Output	BOOL	Q, M, D, L	BUSY = TRUE: petición en proceso.
ERROR	Output	BOOL	Q, M, D, L	ERROR = TRUE: petición cancelada con errores.
STATUS	Output	DWORD	M, D	Estado de la petición / Código de error Véase la descripción de "Parámetro STATUS".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Parámetros LADDR, SD y RD

Los parámetros LADDR, SD y RD no pueden modificarse en ninguna ejecución del bloque mientras se procesa una petición; deben permanecer constantes.

Parámetro STATUS

La tabla siguiente contiene las posibles indicaciones de STATUS en función de DONE y ERROR.

DONE	ERROR	STATUS	Significado
0	0	00700000_H	Primera llamada con REQ = FALSE; no hay ninguna petición activa.
0	0	00700100_H	Primera llamada con REQ = TRUE; petición lanzada.
0	0	00700200_H	Llamada sucesiva (REQ es irrelevante); se sigue procesando la petición.
1	0	00000000_H	Petición finalizada sin errores. Sin datos de respuesta.
1	0	$0000xx00_H$	Petición finalizada sin errores. Número de bytes xx de los datos de respuesta.
0	1	$C0818400_H$	Tipo de datos del operando formal RD no admisible.
0	1	$C0818500_H$	Error de comunicación con el maestro AS-i (dirección configurada erróneamente en el parámetro LADDR).
0	1	$C0838100_H$	Dirección errónea del esclavo AS-i.
0	1	$C0838200_H$	Esclavo AS-i no activado (no en LAS).
0	1	$C0838300_H$	Error en AS-Interface (es posible que se haya ajustado un valor insuficiente en el parámetro SD).
0	1	$C0838400_H$	En el estado actual del maestro AS-i este comando no está permitido.
0	1	$C0838500_H$	Existe un esclavo AS-i con dirección "0".
0	1	$C0838600_H$	El esclavo AS-i tiene datos de configuración no permitidos (E/S o códigos ID).
0	1	$C083A100_H$	El esclavo AS-i al que se desea acceder no se encuentra en el AS-Interface.
0	1	$C083A200_H$	Existe un esclavo AS-i con dirección "0".
0	1	$C083A300_H$	En el AS-Interface ya existe un esclavo AS-i con la dirección nueva.
0	1	$C083A400_H$	No es posible borrar la dirección del esclavo AS-i.
0	1	$C083A500_H$	No es posible establecer la dirección del esclavo AS-i.
0	1	$C083A600_H$	No es posible guardar permanentemente la dirección del esclavo AS-i.
0	1	$C083A700_H$	Error al leer el código Extended ID1.
0	1	$C083A800_H$	La dirección de destino no es plausible (p. ej., se utiliza una dirección de esclavo B para un esclavo estándar).
0	1	$C083B100_H$	Se ha producido un error de longitud en la transferencia de string.
0	1	$C083B200_H$	Se ha producido un error de protocolo en la transferencia de string.
0	1	$C083F800_H$	Se desconoce el número de petición o el parámetro de la petición.
0	1	$C083F900_H$	El maestro AS-i ha detectado un error EEPROM.

Comandos AS-i

Descripción

La interfaz de comandos permite intercambiar datos de parametrización e información entre el controlador y el maestro AS-i.

Con estos comandos:

- se proporciona a la especificación del maestro AS-i la funcionalidad completa del perfil de maestro M4;
- puede configurarse completamente el maestro AS-i desde el controlador.

Nota

Comandos AS-i soportados

En el manual del maestro AS-i correspondiente pueden consultarse los comandos AS-i soportados y una descripción detallada de los mismos.

Estructura general del búfer de transmisión

La tabla siguiente muestra la estructura general del búfer de transmisión para comandos y datos de peticiones. El área para el número de comando debe estar ocupada siempre. El número de bytes para los datos de la petición se encuentran en la descripción de cada comando (véase la documentación del maestro AS-i). "q" corresponde a la dirección inicial del búfer de transmisión.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Significado							
q + 0	Número del comando							
q + 1	Datos de la petición							
q + 2	Datos de la petición							
q + ...	Datos de la petición							

Estructura general del búfer de recepción

La tabla siguiente muestra la estructura general del búfer de recepción para los datos de respuesta del comando. El número de bytes para los datos de respuesta depende de cada comando. Algunos comandos no proporcionan datos de respuesta y, por tanto, necesitan la especificación de un búfer de recepción virtual que no esté rellenado con datos. "n" corresponde a la dirección inicial del búfer de recepción.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
	Significado							
n + 0	Número del comando (eco)							
n + 1	Datos de respuesta							
n + 2	Datos de respuesta							
n + ...	Datos de respuesta							

PRECAUCIÓN
Las áreas de memoria pueden sobrescribirse Si el búfer de recepción de la instrucción "ASI_CTRL" es demasiado corto, pueden sobrescribirse las áreas de memoria colindantes. La longitud indicada en el puntero ANY del parámetro RD durante la llamada de la instrucción "ASI_CTRL" es irrelevante. La longitud necesaria del búfer de recepción debe consultarse en la descripción del comando. Para los números de comando 39_H, 41_H, 42_H, 43_H y 44_H se aplica lo siguiente: el búfer de recepción debe tener 221 bytes de longitud (del byte 0 al byte 220) aun cuando el comando devuelva menos datos. Dependiendo del comando, el maestro AS-i puede sobrescribir con valores cero los bytes más altos contenidos en el búfer de recepción.

Comandos AS-i

La tabla siguiente muestra una selección posible de comandos AS-i.

Nombre	Parámetro	Retorno	Codificación
Configurar_valor_de_parámetro (Set_Permanent_Parameter)	Dirección de esclavo, parámetros		00_H
Leer_valor_de_parámetro_configurado (Get_Permanent_Parameter)	Dirección de esclavo	Parámetros	01_H
Escribir_valor_de_parámetro (Write_Parameter)	Dirección de esclavo, parámetros	Eco de parámetro	02_H
Leer_valor_de_parámetro (Read_Parameter)	Dirección de esclavo	Valor del parámetro	03_H
Configurar_valores_de_parámetros_reales (Store_Actual_Parameters)			04_H
Configurar_datos_de_configuración	Dirección de esclavo, configuración		25_H
Leer_datos_de_configuración_configurados	Dirección de esclavo	Datos de configuración configurados	26_H
Configurar_datos_de_configuración_reales (Store_Actual_Configuration)			07_H
Leer_datos_de_configuración_reales	Dirección de esclavo	Datos de configuración reales	28_H
Configurar_LPS	LPS		29_H
Activar_modo_Offline	Modo		$0A_H$
Seleccionar_autoprogramación	Modo		$0B_H$
Activar_modo	Modo		$0C_H$
Cambiar_dirección_esclavo_AS–i (Change_AS–iSlave_Address)	Dirección1, dirección2		$0D_H$
Leer_estado_esclavo_AS–i	Dirección de esclavo	Registro de errores del esclavo AS-i	$0F_H$
Leer_listas_y_flags		LDS, LAS, LPS, flags	30_H

Nombre	Parámetro	Retorno	Codificación
Leer_configuración_total		Datos de configuración reales, parámetros actuales, LAS, flags	39 $_H$
Configurar_configuración_total	Configuración total		3A $_H$
Escribir_lista_parámetros	Lista de parámetros		3C $_H$
Leer_lista_eco_parámetros		Lista de ecos de parámetros	33 $_H$
Escribir_petición_CTT2	Dirección de esclavo string CTT2	String CTT2	44 $_H$
Leer_identificador_versión		String de versión	14 $_H$
Leer_ID_esclavo_AS–i	Dirección de esclavo	Código ID	17 $_H$
Leer_Extended–ID1_esclavo_AS-i	Dirección de esclavo	Código Extended ID1	37 $_H$
Escribir_Extended-ID1_esclavo_AS–i	Código Extended ID1		3F $_H$
Leer_Extended–ID2_esclavo_AS-i	Dirección de esclavo	Código Extended ID2	38 $_H$
Leer_ES_esclavo_AS–i	Dirección de esclavo	Configuración E/S	18 $_H$
Leer_lista_errores_periferia		LPF	3E $_H$
Escribir_string_parámetros_esclavo_AS–i	Dirección de esclavo, string de parámetros		40 $_H$
Leer_string_parámetros_esclavo_AS–i	Dirección de esclavo	String de parámetros	41 $_H$
Leer_string_ID_esclavo_AS–i	Dirección de esclavo	String ID	42 $_H$
Leer_string_diagnóstico_esclavo_AS–i	Dirección de esclavo	String de diagnóstico	43 $_H$
Leer_contador_errores_línea_AS–i			4A $_H$
Leer_y_borrar_contador_errores_línea_AS–i			4B $_H$
Leer_contador_errores_esclavo_AS–i	Dirección de esclavo		4C $_H$
Leer_y_borrar_contador_errores_esclavo_AS–i	Dirección de esclavo		4D $_H$
Comando adicional para el DP/ AS-i F-Link:			
Estado_AS-i/Diag_esclavos_F		Estado / Diagnóstico de todos los esclavos ASIsafe	51 $_H$

Nota

Nueva inicialización de la interfaz de comandos del maestro AS-i

Otro comando no mencionado en la tabla es el comando 77 $_H$. Con esta llamada se reinicializa la interfaz de comandos del maestro AS-i. Si justo en este momento el maestro AS-i indicado está ocupado con un comando, este se interrumpe.

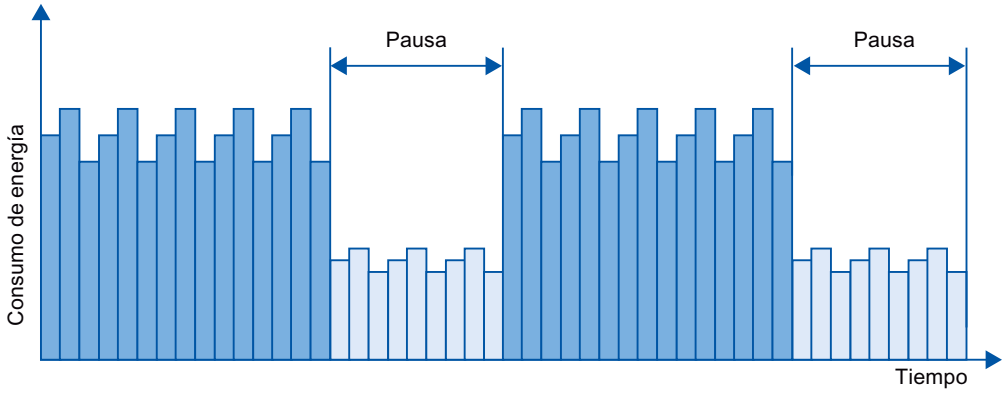
A partir de la versión V2.1.20 de DP/AS-i LINK Advanced se ofrece el comando adicional 0E $_H$. Con esta llamada puede habilitarse o bloquearse la función de vigilancia de contacto a tierra de una línea.

PROFIenergy

Descripción de PROFIenergy

PROFIenergy

PROFIenergy es un perfil de gestión energética con PROFINET independiente de cualquier fabricante y cualquier dispositivo concreto. PROFIenergy permite desconectar los dispositivos de forma coordinada y centralizada para reducir el consumo eléctrico durante pausas de producción o durante interrupciones no planificadas.



Los dispositivos PROFINET o los módulos de potencia se desconectan mediante comandos especiales en el programa de usuario del controlador PROFINET IO. No se requiere hardware adicional. Los dispositivos PROFINET interpretan directamente los comandos PROFIenergy.

Controlador PROFIenergy (controlador PE)

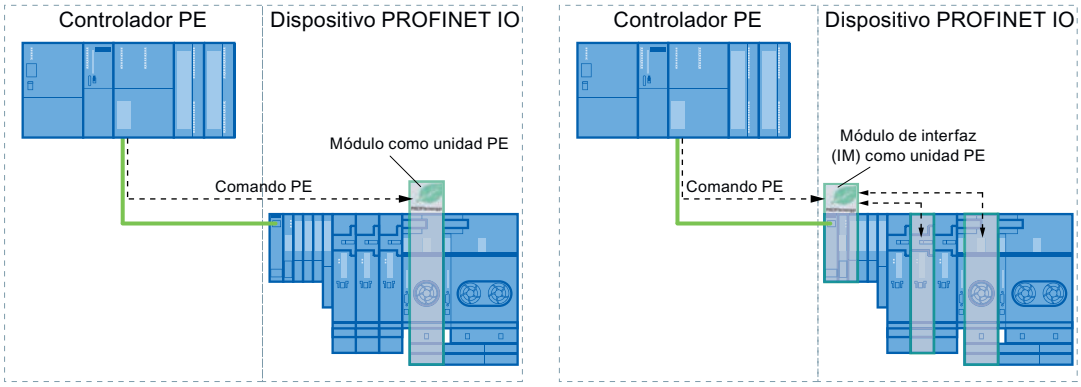
El controlador PE es un PLC que activa o desactiva el estado de reposo de los dispositivos subordinados. La desconexión o la reactivación de componentes de producción concretos o de líneas de producción enteras se realiza mediante el programa de usuario. Los comandos (p. ej. "Start_Pause" o "End_Pause") se emiten al dispositivo subordinado mediante las instrucciones correspondientes (bloques de función). Los comandos se envían con el protocolo de comunicación PROFINET.

Unidad PROFIenergy (unidad PE)

La unidad PE recibe los comandos PROFIenergy del controlador PE y los aplica de forma correspondiente (p. ej. devolviendo un valor medido o activando un modo de ahorro de energía). La aplicación de la unidad PE en un dispositivo apto para PROFIenergy depende del tipo de dispositivo y del fabricante.

Le unidad PE se puede aplicar, por ejemplo, del siguiente modo:

- En el proxy de un módulo: los comandos PE son válidos para el módulo direccionado, respectivamente, y dado el caso, para módulos subordinados.
- En el proxy de una tarjeta: los comandos PE son válidos para distintos módulos dentro de la tarjeta, respectivamente.



- En un módulo sin funcionalidad Proxy dentro de la red: en este caso, los comandos PE solo son válidos para el módulo correspondiente.

Instrucciones PROFIenergy

- Instrucciones para controlador IO
  - La manera más sencilla de activar o desactivar el estado de reposo de los dispositivos PROFINET es con la instrucción "PE_START_END (Página 6822)" (comandos PROFIenergy "Start_Pause" y "End_Pause"). mediante un flanco de señal ascendente y descendente en la misma.
  - La instrucción "PE_CMD (Página 6826)" transfiere todos los comandos PROFIenergy, incluidos "Start_Pause" y "End_Pause". Además, con otros comandos puede consultarse el estado actual de los dispositivos PROFINET o su comportamiento durante las pausas, por ejemplo.
  - Con la instrucción "PE_DS3_Write_ET200S (Página 6831)" se especifican los ajustes para el comportamiento de conexión de un máximo de 8 slots del ET 200S. Esta instrucción no es una instrucción PROFIenergy pero complementa las funciones PROFIenergy de un ET 200S.
- Instrucción para iDevices
  La instrucción "PE_I_DEV (Página 6855)" permite ejecutar PROFIenergy también en iDevices. Esta instrucción recibe comandos PROFIenergy en el iDevice y los transfiere al programa de usuario para ejecutarlos. Después de ejecutar el comando, el programa de usuario llama otra vez a la instrucción "PE_I_DEV (Página 6855)" para enviar acuse al controlador IO. Para responder, se dispone para cada comando de un bloque auxiliar específico que proporciona a la instrucción "PE_I_DEV (Página 6855)" los datos de respuesta.

Comandos PROFIenergy (comandos PE)

El controlador PE envía los comandos PE a la unidad PE. El comando PE puede ser un comando de control para poner una unidad PE en un determinado modo de ahorro de energía, o un comando para leer un estado o un valor medido:

- Comandos PE para el control
  PROFIenergy soporta dos comandos de control que pueden ejecutarse mediante la instrucción "PE_Start_End (Página 6822)" o mediante la instrucción "PE_CMD (Página 6826)":
  - Start_Pause: Iniciar un modo de ahorro de energía adecuado (PE Energy-saving mode)
  - End_Pause: Finalizar el modo de ahorro de energía (paso al modo PE_ready_to_operate)
- Comandos PE para leer un estado o un valor medido
  Mediante los siguientes comandos de estado, el control puede leer con la instrucción "PE_CMD (Página 6826)" determinada información de estado:
  - PE_Identify: determina qué comandos PE soporta la unidad PE.
  - PEM_Status: determina cuál es el modo activo de una unidad PE en el momento actual (p. ej. PE_ready_to_operate).
  - Query_Modes: Muestra una sinopsis de todos los modos de ahorro de energía soportados, incluida la información de tiempo y energía
  - Query_Measurement: muestra los valores medidos de una unidad PE

Ejemplos de aplicación

Pueden consultarse ejemplos de aplicación de las instrucciones PROFIenergy en el artículo "PROFIenergy - Ahorrar energía con SIMATIC S7 (http://support.automation.siemens.com/WW/view/es/41986454)" del Industry Online Support.

Consulte también

Service & Support (http://support.automation.siemens.com/)

Controlador IO

PE_START_END: Iniciar y finalizar modo de ahorro de energía

Descripción

Con la instrucción "PE_START_END" se inicia y finaliza la pausa de ahorro de energía para la unidad PE especificada (p. ej. el ET 200S).

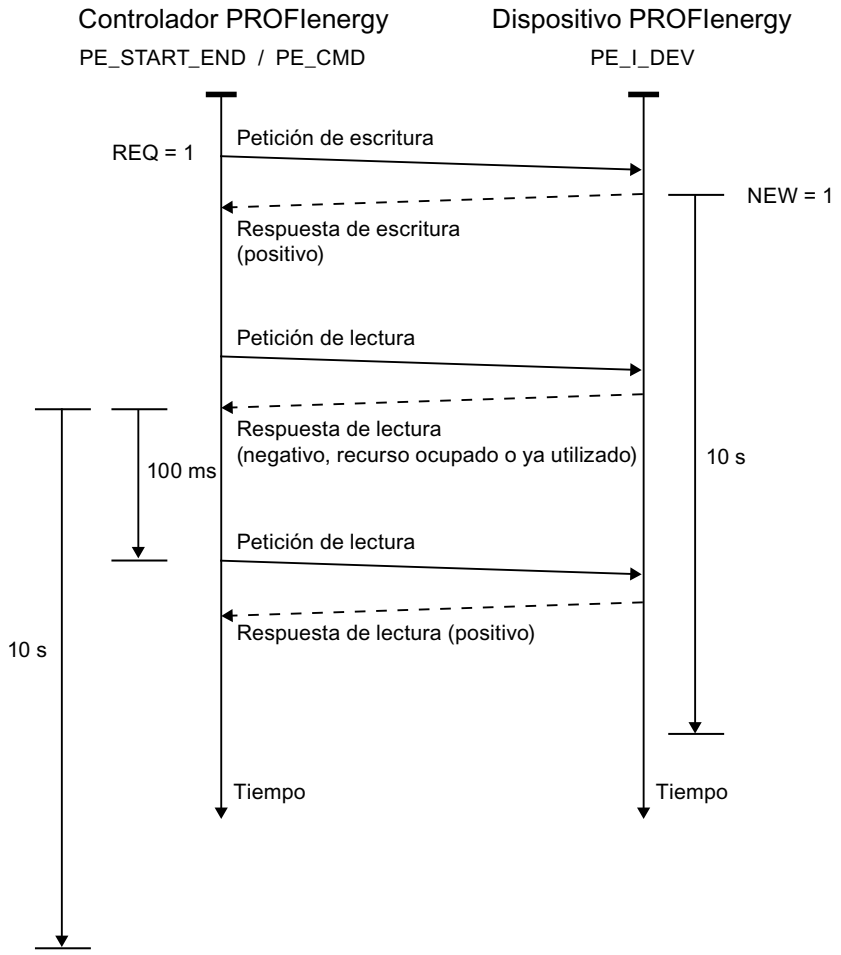
La instrucción "PE_START_END" se emplea en el controlador PE preferentemente cuando en los dispositivos PE asignados solo hay conectados aparatos de campo en los que no van a leerse datos de energía. Para leer los datos de energía también puede emplearse la instrucción "PE_CMD (Página 6826)".

Los modos de ahorro de energía se configuran en el programa de usuario del controlador PE. Después de ejecutar "PE_START_END", la unidad PE responde notificando qué modo de ahorro de energía se ha adoptado realmente y lo indica en el parámetro PE_MODE_ID.

Peticiones de escritura y lectura de la instrucción "PE_START_END"

La instrucción "PE_START_END" envía internamente mediante "WRREC (Página 6738)" un comando PROFIenergy en forma de petición de escritura a la unidad PE. A continuación, "PE_START_END" espera el acuse de la unidad PE. Para ello, cada 100 milisegundos se lee el registro de acuse con la instrucción "RDREC (Página 6735)". Mientras no se reciba acuse de la entidad PE, la petición de lectura se repetirá cada 100 milisegundos durante 10 segundos. Los datos de respuesta de la unidad PE también se leen con la instrucción "RDREC (Página 6735)".

La figura siguiente muestra el diagrama de ejecución de las peticiones de escritura y lectura:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_START_END":

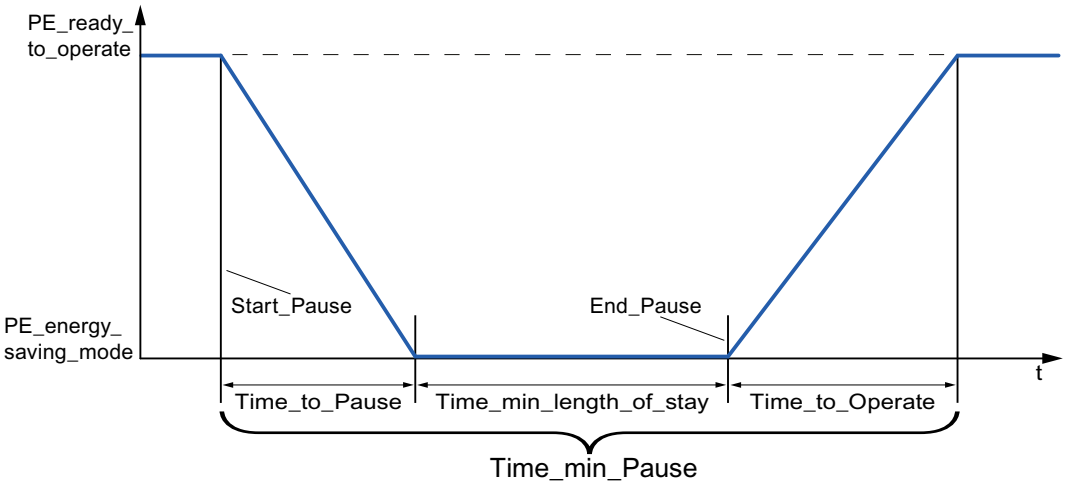
Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
START	Input	BOOL	I, Q, M, D, L	Transmisión del comando PE "Start_Pause" a la unidad PE con la dirección en el parámetro ID.
END	Input	BOOL	I, Q, M, D, L	Transmisión del comando PE "End_Pause" a la unidad PE con la dirección en el parámetro ID.
ID	Input	DWORD	I, Q, M, D, L o constante	Dirección de la entidad PE (p. ej. ET 200S). La dirección puede tomarse de la configuración de hardware.
PAUSE_TIME	Input	TIME	I, Q, M, D, L o constante	Duración planificada de la pausa. • Rango: T#1MS a T#24D20H31M23S647MS • Valor de arranque: T#0MS
VALID	Output	BOOL	I, Q, M, D, L	Comando PE emitido correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Aún se está ejecutando el comando PE.
ERROR	Output	BOOL	I, Q, M, D, L	Se ha producido un error durante la ejecución. El mensaje de error se indica en el parámetro STATUS.
STATUS	Output	DWORD	I, Q, M, D, L	Estado de bloque o número de error (véase "Parámetro STATUS")
PE_MODE_ID	Output	BYTE	I, Q, M, D, L	Número de identificación del modo de ahorro de energía (nivel de ahorro de energía adoptado durante la pausa).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro PAUSE_TIME

Con el parámetro PAUSE_TIME se predefine la duración de la pausa de ahorro de energía para la unidad PE. En la unidad PE se comprueba si la pausa predefinida tiene duración suficiente y si se puede aplicar. La duración mínima de pausa (Time_min_Pause) debe ser

mayor que la suma de los tiempos que el dispositivo necesita para cambiar al modo de ahorro de energía (Time_to_Pause) y para volver al modo de operación (Time_to_Operate).



En el ET 200S se comprueba si la pausa planificada es mayor o igual a la duración mínima de pausa (PM-E_Pause_Min) depositada en el ET 200S. Esta es de 10 segundos. Si se utiliza una pausa menor, los módulos de potencia (PM-E) del ET 200S permanecen conectados.

Una vez transcurrido el tiempo de pausa no se produce ningún rearranque automático, sino que el módulo permanece en estado OFF hasta que se emite el comando "END". De este modo se evita una reconexión descoordinada, que podría provocar picos de carga indeseados.

Parámetro STATUS

En el parámetro de salida STATUS se muestra información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	Causa del error • B#16#**00**: sin error • B#16#**DE**: error al leer el registro • B#16#**DF**: error al escribir el registro • B#16#**C0**: mensaje de error de la instrucción o de las instrucciones de comunicación "RDREC (Página 6735)" y "WRREC (Página 6738)" empleadas internamente.
STATUS[2]	Error_Decode	Ubicación de la identificación de error • **80**: error DPV1 según la norma IEC 61158-6 o error específico de instrucción • **FE**: perfil DP/PNIO; error específico de PROFIenergy

Elemento de campo	Nombre	Significado
STATUS[3]	Error_Code_1	Identificación de error • Con Error_Decode = **80**: – **80**: flancos ascendentes simultáneos en los parámetros de entrada START y END. – **81**: conflicto de longitud en los parámetros CMD_PARAM y CMD_PARAM_LEN. – **82-8F**: otros mensajes de error (reservado) • Con Error_Decode = **FE**: – **01**: "Service Request ID" no válido – **02**: "Request_Reference" erróneo – **03**: "Modifier" no válido – **04**: "Data Structure Identifier RQ" no válido – **05**: "Data Structure Identifier RS" no válido – **06**: "PE energy-saving modes" no soportados – **07**: "Response" es demasiado largo (se ha rebasado la longitud máxima transferible) – **08**: "Count" no válido – **50**: no hay ningún "energy mode" adecuado disponible. – **51**: el valor de tiempo indicado no se soporta. – **52**: "PE_Mode_ID" no admisible
STATUS[4]	Error_Code_2	Extensión específica de fabricante de la identificación de error

Nota

Mensajes de error de las instrucciones RDREC y WRREC

La instrucción "PE_START_END" utiliza para la comunicación las instrucciones "WRREC (Página 6738)" y "RDREC (Página 6735)". Los mensajes de error de estas instrucciones se emiten correspondientemente en los elementos de campo STATUS[1] a STATUS[4].

Consulte el significado de los códigos de error de las instrucciones "WRREC (Página 6738)" y "RDREC (Página 6735)" en la descripción del parámetro STATUS (Página 6748) correspondiente.

PE_CMD: Iniciar y finalizar modo de ahorro de energía / Leer información de estado

Descripción

La instrucción "PE_CMD" se emplea en el controlador PE e inicia o finaliza una pausa de ahorro de energía en la unidad PE. Además, "PE_CMD" permite leer más información y valores medidos de energía de una unidad PE.

La instrucción puede emplearse preferentemente en controladores PE en cuyos dispositivos PE correspondientes estén conectados aparatos de campo de los que deban leerse valores medidos de energía. Si esto no fuera así, para iniciar y finalizar las pausas también puede emplearse la instrucción "PE_START_END (Página 6822)".

Transferencia de los comandos PROFIenergy (comandos PE)

La instrucción "PE_CMD" transfiere un comando PROFIenergy a una unidad PE.

La instrucción también puede emplearse si en el futuro se amplía el perfil PROFIenergy con otros comandos. Los comandos que pueden utilizarse de acuerdo con el perfil PROFIenergy actual se listan en la descripción de los parámetros CMD y CMD_MODIFIER (véase tabla "Parámetros CMD y CMD_MODIFIER").

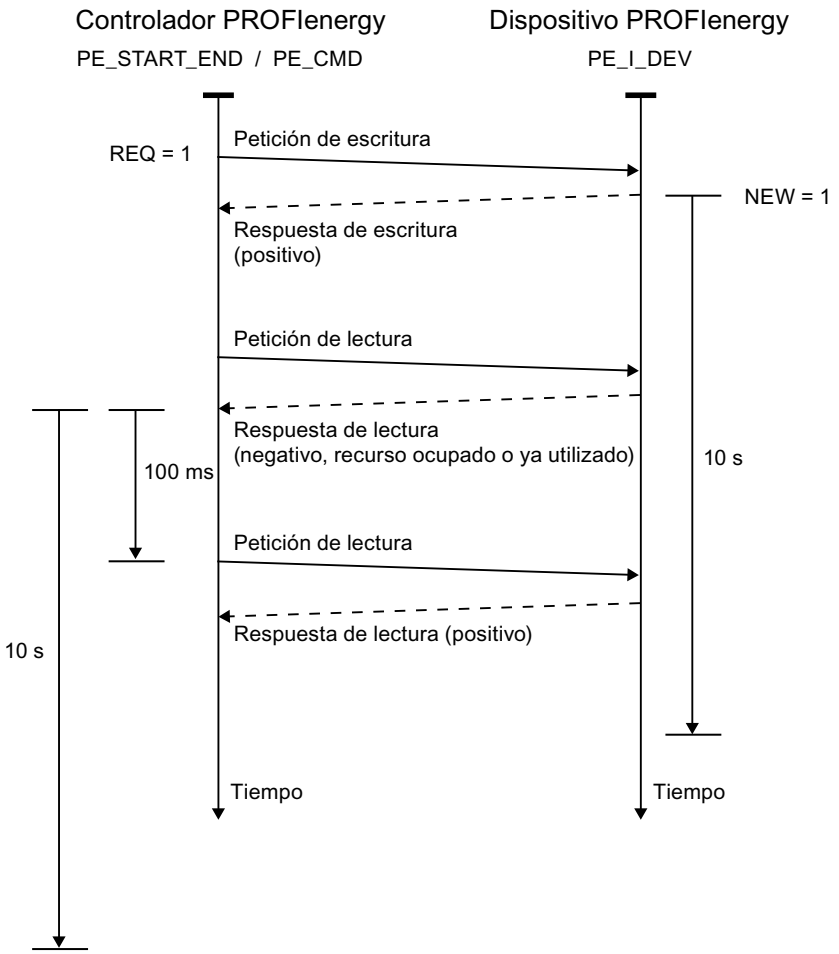
- Los diferentes comandos PE que se transfieren a la unidad PE mediante la instrucción tienen asignadas "Service_Request_IDs" definidas. Las Service_Request_IDs 01...05 y 16 se asignan en el parámetro CMD.
- Mediante el parámetro CMD_MODIFIER se especifican los dos comandos PE 04 (Query_Modes) y 16 (Query_Measurement).
- Para comandos PE concretos, en el parámetro CMD_PARA se transfieren valores adicionales (véase la descripción de los distintos comandos PE). El parámetro CMD_PARA_LEN define la longitud de datos en el parámetro CMD_PARA.

Los comandos se transfieren sin prueba de plausibilidad. Los datos de respuesta de la entidad PE se depositan en el área de datos RESPONSE_DATA direccionada por el puntero ANY (los contenidos de los telegramas de respuesta figuran en las descripciones del respectivo comando PE).

Peticiones de escritura y lectura de la instrucción "PE_CMD"

La instrucción "PE_CMD" envía mediante "WRREC (Página 6738)" un comando PROFIenergy en forma de petición de escritura a la unidad PE. A continuación, "PE_CMD" espera el acuse de la unidad PE. Para ello, cada 100 milisegundos se lee el registro de acuse con la instrucción "RDREC (Página 6735)". Mientras no se reciba acuse de la unidad PE, la petición de lectura se repite cada 100 milisegundos durante 10 segundos. Los datos de respuesta de la unidad PE también se leen con la instrucción "RDREC (Página 6735)".

La figura siguiente muestra el diagrama de ejecución de las peticiones de escritura y lectura:



Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_CMD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Inicia la transmisión del comando PE con flanco ascendente.
ID	Input	DWORD	I, Q, M, D, L o constante	Dirección de la unidad PE Para los dispositivos PROFINET IO, la dirección puede tomarse de la configuración hardware.
CMD	Input	BYTE	I, Q, M, D, L, P o constante	Service-Request-ID del comando PROFIenergy en función del perfil PROFIenergy (véase "Parámetros CMD y CMD_MODIFIER"). Si se amplía el perfil PROFIenergy son posibles otras Service-Request-IDs.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD_MODIFIER	Input	BYTE	I, Q, M, D, L, P o constante	Subcomando PROFIenergy (solo con CMD=3 o CMD=16, véase "Parámetros CMD y CMD_MODIFIER") Si se amplía el perfil PROFIenergy son posibles otros subcomandos.
CMD_PARA	Input	ANY	I, Q, M, D, L	Parámetros de los comandos PE: • Get mode: PE_mode_ID • Get measurement values: List of Measurement_Ids Se registra el Service Data Request completo.
CMD_PARA_LEN	Input	INT	I, Q, M, D, L, P o constante	Longitud real de los parámetros del comando (la instrucción comprueba si es <= a la longitud en CMD_PARA).
RESPONSE_DATA	InOut	ANY	I, Q, M, D, L	Información PROFIenergy Dependiendo del comando, telegrama de respuesta completo, incluido encabezado de bloque. Nota: si el búfer es demasiado pequeño solo se registra el número de bytes, como se indica en el puntero ANY.
VALID	Output	BOOL	I, Q, M, D, L	Comando emitido correctamente.
BUSY	Output	BOOL	I, Q, M, D, L	Aún se está ejecutando el comando.
ERROR	Output	BOOL	I, Q, M, D, L	Se ha producido un error durante la ejecución.
STATUS	Output	DWORD	I, Q, M, D, L	Estado de bloque o número de error (véase "Parámetro STATUS"):

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros CMD y CMD_MODIFIER

CMD	CMD_ MODIFIER	Comando PROFIenergy	Descripción
01	0	Start_Pause (Página 6844)	Inicia el modo de ahorro de energía o cambia a otro modo de ahorro de energía.
02	0	End_Pause (Página 6845)	Sale del modo de ahorro de energía.
03	1	Query_Modes - List energy saving Modes (Página 6846)	Muestra los modos de ahorro de energía soportados.
	2	Query_Modes - Get Mode (Página 6846)	Muestra los atributos del modo de ahorro de energía activado en el momento actual.
04	0	PEM_Status (Página 6848)	Consulta el estado del modo de ahorro de energía.
05	0	PE_Identify (Página 6850)	Lectura del número de comandos PE soportados y descripción de los mismos.

CMD	CMD_ MODIFIER	Comando PROFIenergy	Descripción
16	1	Query_Measurement - Get_Measurement_List (Página 6851)	Listado de los valores medidos soportados por la unidad PE.
	2	Query_Measurement - Get_Measurement_Values (Página 6853)	Muestra los valores medidos de la unidad PE.

Parámetro STATUS

En el parámetro de salida STATUS se muestra información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	Causa del error • B#16#00: sin error • B#16#DE: error al leer el registro • B#16#DF: error al escribir el registro • B#16#C0: mensaje de error de las instrucciones de comunicación "RDREC (Página 6735)" y "WRREC (Página 6738)" empleadas internamente.
STATUS[2]	Error_Decode	Ubicación de la identificación de error • 80: error DPV1 según la norma IEC 61158-6 o error específico de instrucción • FE: perfil DP/PNIO; error específico de PROFIenergy
STATUS[3]	Error_Code_1	Identificación de error • Con Error_Decode = 80: – 81: conflicto de longitud en los parámetros CMD_PARA y CMD_PARA_LEN o bien se ha excedido la longitud máxima del registro (4095 bytes). – 82-8F: otros mensajes de error (reservado) • Con Error_Decode = FE: – 01: "Service Request ID" no válido – 02: "Request_Reference" erróneo – 03: "Modifier" no válido – 04: "Data Structure Identifier RQ" no válido – 05: "Data Structure Identifier RS" no válido – 06: "PE energy-saving modes" no soportados – 07: "Response" es demasiado largo (se ha rebasado la longitud máxima transferible) – 08: "Count" no válido – 50: no hay ningún modo de ahorro de energía (energy mode) adecuado disponible. – 51: el valor de tiempo indicado no se soporta. – 52: "PE_Mode_ID" no válido
STATUS[4]	Error_Code_2	Extensión específica de fabricante de la identificación de error

Nota

Mensajes de error de las instrucciones RDREC y WRREC

Las instrucciones "PE_CMD" utilizan para la comunicación las instrucciones "WRREC (Página 6738)" y "RDREC (Página 6735)". Los mensajes de error de estas instrucciones se emiten correspondientemente en los elementos de campo STATUS[1] a STATUS[4].

Consulte el significado de los códigos de error de las instrucciones "WRREC (Página 6738)" y "RDREC (Página 6735)" en la descripción del parámetro STATUS (Página 6748) correspondiente.

PE_DS3_Write_ET200S: Ajustar el modo de activación de los módulos de potencia

Descripción

La instrucción "PE_DS3_Write_ET200S" envía al ET 200S configuraciones básicas sobre el modo de activación de los módulos de potencia. La instrucción "PE_DS3_Write_ET200S" permite definir el modo de activación de hasta 8 slots del ET 200S (p. ej. para módulos de potencia).

Nota

Esta instrucción no forma parte del perfil PROFIenergy, sino que complementa funciones específicas de SIMATIC.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_DS3_Write_ET200S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ENABLE	Input	BOOL	I, Q, M, D, L	Un flanco ascendente activa la transferencia del registro. El registro debe transferirse de nuevo después de desconectar y volver a conectar la tensión.
ID	Input	DWORD	I, Q, M, D, L o constante	Dirección del ET 200S La dirección puede tomarse de la configuración de hardware.
SLOT_NO_X	Input	INT	I, Q, M, D, L, P o constante	Número de slot del X.º módulo de potencia conectable.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
FUNC_X	Input	INT	I, Q, M, D, L, P o constante	Función del módulo en este slot. Mediante el parámetro FUNC_X se define el modo de activación del PM-E (módulo de potencia del ET 200S): • FALSE: – Con "PAUSE_START": - no influye en el PM-E - el PM-E continúa conectado – Con "PAUSE_STOP": - se reconecta el PM-E • TRUE: – Con "PAUSE_START": - se desconecta el PM-E – Con "PAUSE_STOP": - se reconecta el PM-E
BUSY	Output	BOOL	I, Q, M, D, L	La transferencia no ha finalizado todavía.
DONE	Output	BOOL	I, Q, M, D, L	La transferencia ha finalizado sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Transferencia finalizada con error.
STATUS	Output	DWORD	I, Q, M, D, L, P	Número de error (véase el parámetro STATUS de la instrucción "PE_Start_End (Página 6822)")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_WOL: Iniciar y finalizar modo de ahorro de energía mediante WakeOnLan

Descripción PE_WOL

Descripción

La instrucción "PE_WOL" envía los comandos PROFIenergy "Start_Pause" y "End_Pause" a varios dispositivos compatibles con PROFIenergy en sistemas PROFINET I/O.

La instrucción permite la coordinación de varios dispositivos PE siempre que soporten la función "Wake on LAN" a través de una conexión UDP.

La instrucción "PE_WOL" solo puede ejecutarse en una CPU con interfaz Ethernet integrada. Esta CPU debe ser capaz de cargar bloques de un tamaño de aprox. 40 kB. El bloque no se puede utilizar con sistemas PROFINET I/O si estos están conectados a través de un CP Ethernet.

La instrucción "PE_WOL" se ejecuta de forma asíncrona.

Definición: Wake on LAN

La funcionalidad Wake On LAN permite que dispositivos de procesamiento de datos reanuden el trabajo a partir de un estado prácticamente de desconexión gracias a la recepción de un paquete Ethernet especial.

Para que funcione este procedimiento, el dispositivo de procesamiento de datos debe disponer de un controlador de red equipado para poder recibir un paquete de este tipo.

Este paquete (Magic Packet™) tiene un formato especial. Contiene 15 veces la dirección MAC del adaptador de red.

Selección de los dispositivos

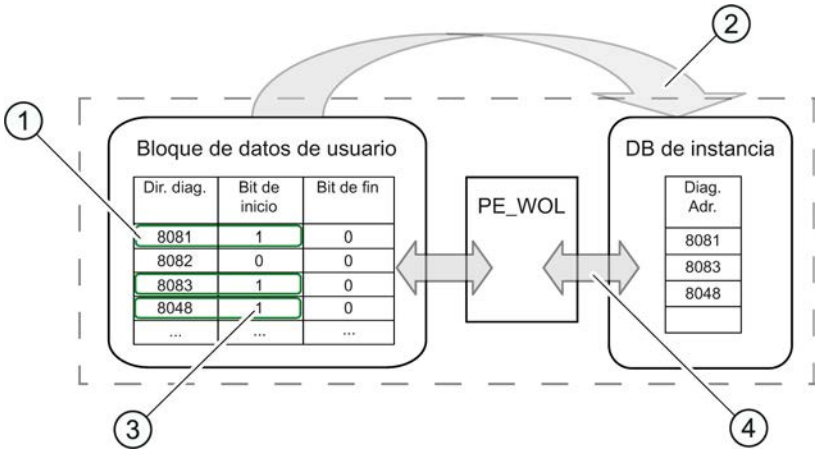
La selección de los dispositivos se realiza a través del bloque de datos de usuario en el parámetro PENERGY. El DB de usuario representa la base de datos para el procesamiento de varios dispositivos.

Antes de la inicialización de "PE_WOL" se debe guardar en el DB de usuario la siguiente información, como mínimo:

- ID del sistema PROFINET I/O
- Datos de la conexión que se utiliza para "Wake On LAN"
- Número de puerto que se utiliza para "Wake On LAN"
- Para cada dispositivo
  - Tiempo de pausa (PauseTime)
  - Conmutación del dispositivo al PE_SLEEP_MODE (EnableSleep)

Con el parámetro COM_RST se inicializa la instrucción "PE_WOL". Tras la inicialización se ejecutan las peticiones guardadas en el DB de usuario una tras otra.

El siguiente gráfico muestra a modo de ejemplo cómo se transmite el comando PE "Start_Pause" a varios dispositivos:



(1) Paso 1: El usuario pone a "1" el bit "CmdStartPause" de los dispositivos que se van a desconectar.

(2) Paso 2: Las direcciones de diagnóstico de los dispositivos que se van a desconectar (CmdStartPause = "1") se encadenan en la lista de espera.

(3) Paso 3: El bit "CmdStartPause" se desactiva automáticamente una vez encadenadas las peticiones.

(4) Paso 4: La instrucción "PE_WOL" procesa las peticiones en cuanto estas están encadenadas.

A través de los parámetros START y END se puede enviar un comando PROFIenergy "CmdStartPause" o "CmdEndPause" para todos los dispositivos detectados en el sistema PROFINET IO.

Con el parámetro STATUS se indica el estado del procesamiento de la petición, así como posibles errores durante su procesamiento.

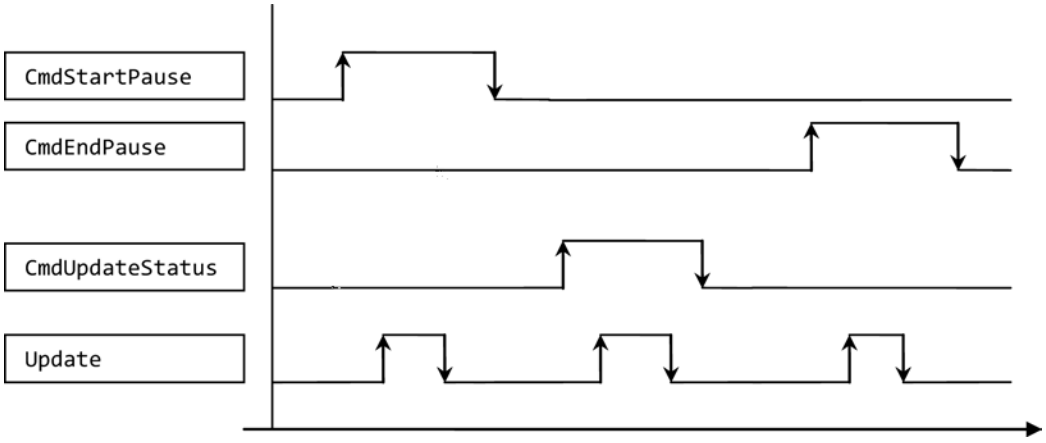
Manejo de la instrucción a través del DB de usuario

El manejo de la instrucción "PE_WOL" solo se puede realizar a través del DB de usuario. A este respecto se aplica un procedimiento básico:

1. Selección del comando que se va a ejecutar para un dispositivo:
   - START_PAUSE ("CmdStartPause" en el DB de usuario)
   - ENDE_PAUSE ("CmdEndPause" en el DB de usuario)
   - UPDATE_STATUS ("CmdUpdateStatus" en el DB de usuario)
2. Activación del bit para la actualización ("Update" en el encabezado del DB de usuario) Entre dos actualizaciones debería transcurrir como mínimo un ciclo de CPU con un "Update" = False, ya que de lo contrario no se puede garantizar la detección de flancos.

Priorización de los comandos PE

El siguiente gráfico muestra la cronología de los tres comandos posibles.



Estos se procesan uno tras otro, independientemente de si el comando previo se ha ejecutado correctamente o ha finalizado con errores.

En caso de activación simultánea de, p. ej., "CmdEndPause" y "CmdUpdateStatus" solo se ejecutará un comando. Dentro del bloque existe una priorización:

- El comando "CmdStartPause" tiene la máxima prioridad y siempre se ejecuta mientras esté seleccionado.
- El comando "CmdEndPause" tiene la segunda máxima prioridad.
- El comando "CmdUpdateStatus" tiene la mínima prioridad.

Si se activan los tres comandos a la vez, los que no se ejecuten permanecen preseleccionados. En ese caso, en el siguiente flanco ascendente se ejecutará el siguiente comando que corresponda.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PE_WOL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
COM_RST (Página 6835)	Input	BOOL	I, Q, M, D, L	Resetea el bloque y ejecuta una reinicialización. Mientras aquí esté ajustado True, comienza la operación de inicialización pero no se ejecuta hasta el final. La inicialización prosigue con el flanco descendente y, una vez concluida esta, se pasa al modo de operación normal.
START (Página 6837)	Input	BOOL	I, Q, M, D, L	Con un flanco ascendente se ejecuta un comando PROFIenergy "CmdStartPause" para todos los dispositivos detectados que soporten esta función.
END (Página 6838)	Input	BOOL	I, Q, M, D, L	Con un flanco ascendente se ejecuta un comando PROFIenergy "CmdEndPause" para todos los dispositivos detectados que soporten esta función.
PENERGY (Página 6838)	InOut	ANY	D	Puntero hacia el DB de usuario que contiene la base de datos para el procesamiento de varios dispositivos.
STATUS (Página 6842)	Output	DWORD	I, Q, M, D, L	Número de estado/error para el estado actual de la instrucción (ver "Parámetro STATUS").

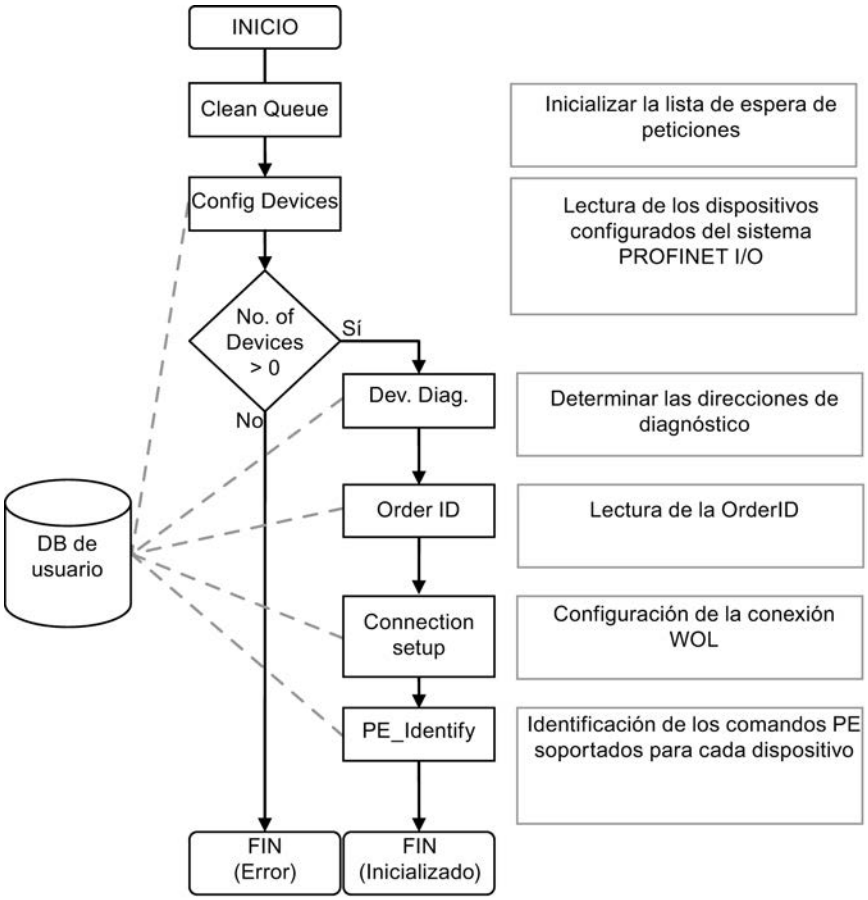
Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro COM_RST

Desarrollo de la rutina de inicialización

Con el parámetro COM_RST se arranca la inicialización de la instrucción "PE_WOL".

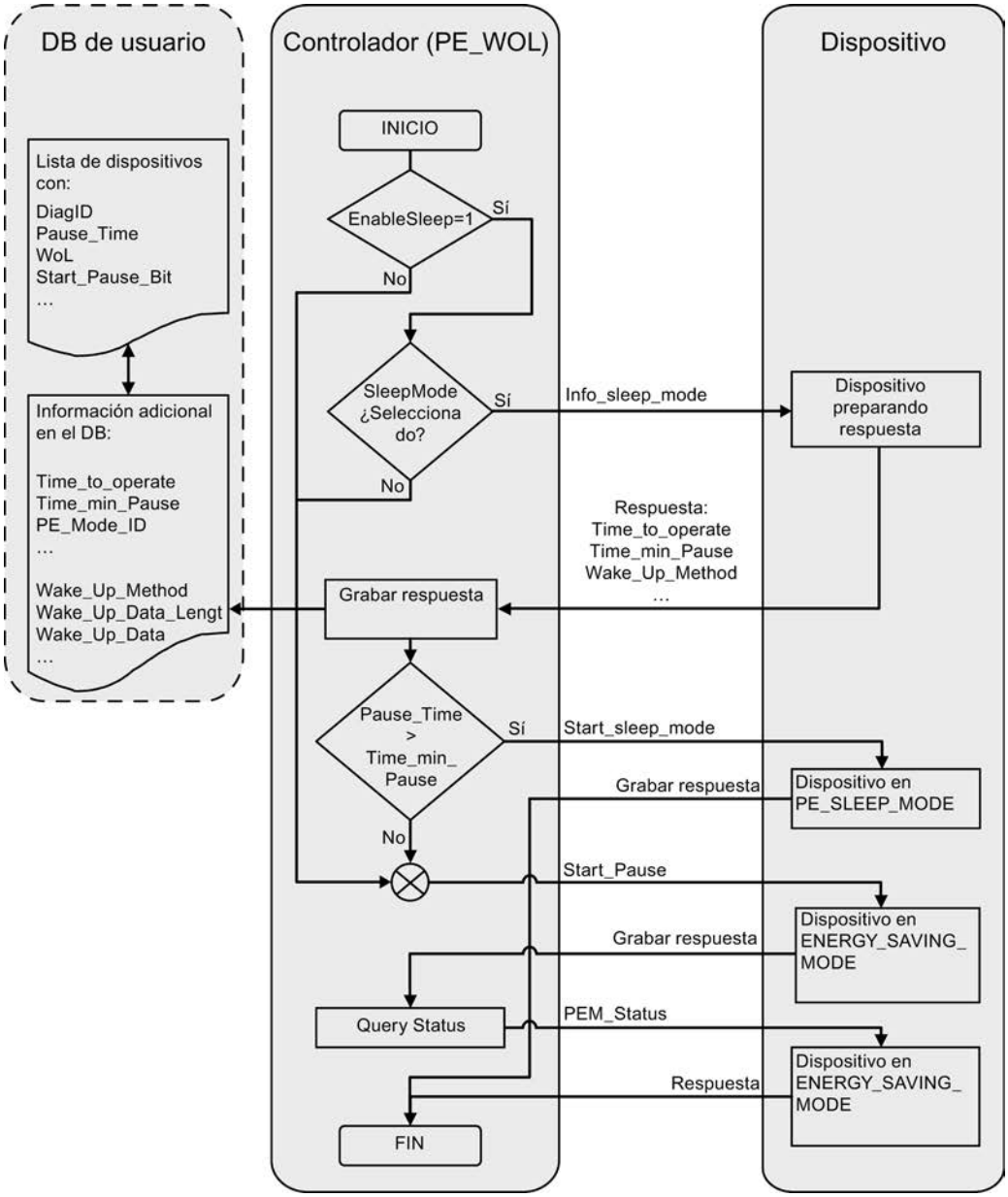
El siguiente diagrama de flujo muestra la rutina de inicialización.


INICIO
Clean Queue
Inicializar la lista de espera de peticiones
Config Devices
Lectura de los dispositivos configurados del sistema PROFINET I/O
No. of Devices > 0
Sí
No
Dev. Diag.
Determinar las direcciones de diagnóstico
Order ID
Lectura de la OrderID
DB de usuario
Connection setup
Configuración de la conexión WOL
PE_Identify
Identificación de los comandos PE soportados para cada dispositivo
FIN (Error)
FIN (Inicializado)

Parámetro START

Desarrollo del comando CmdStartPause

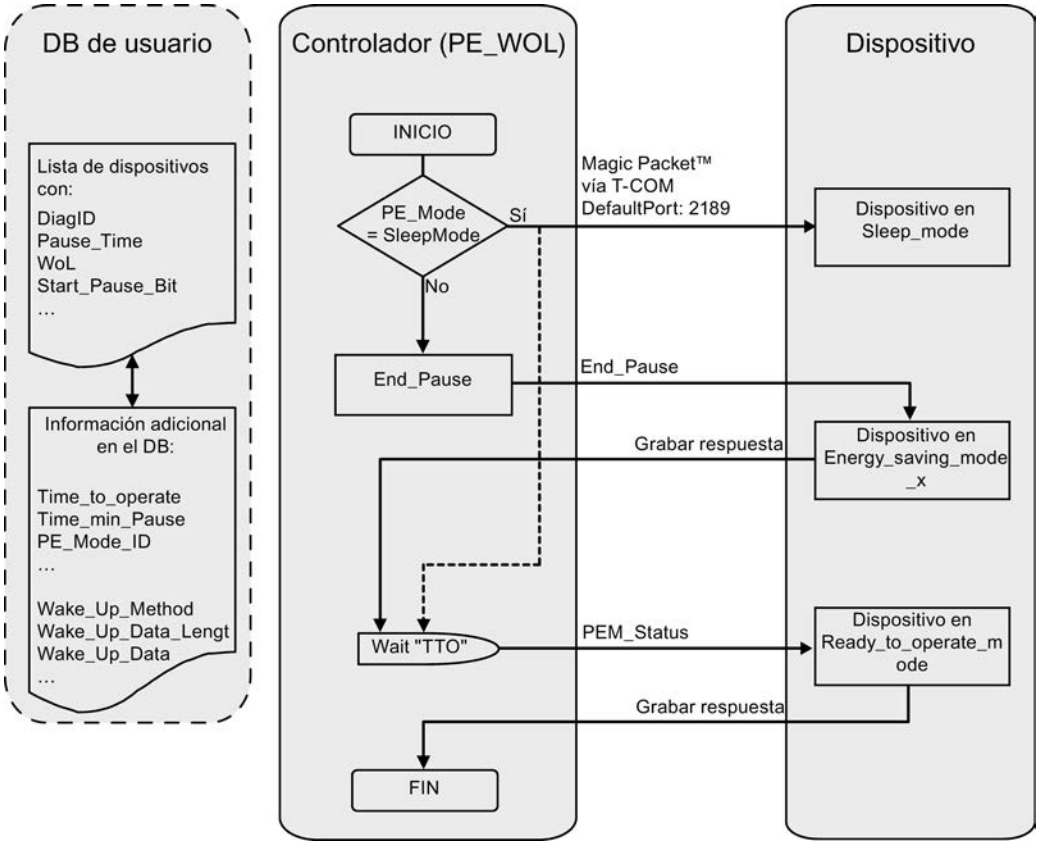
El diagrama de flujo muestra las funciones utilizadas internamente y la interacción con un dispositivo al ejecutar el comando CmdStartPause.



Parámetro END

Desarrollo del comando CmdEndPause

El diagrama de flujo muestra las funciones utilizadas internamente y la interacción con un dispositivo al ejecutar el comando CmdEndPause.



Parámetro PENERGY

Bloque de datos en el parámetro PENERGY

El DB de usuario para la instrucción PROFIenergy "PE_WOL" constituye una base de datos para el procesamiento de varios dispositivos.

El bloque de datos se divide generalmente en dos secciones. A saber:

- Sección de encabezado 110 bytes (header)
- Sección de dispositivos para 256 dispositivos como máximo de 100 bytes cada uno (Device). Estos incluyen:
  - Datos específicos de dispositivo (Device)
  - Datos específicos de PROFIenergy (PE)
  - Datos de procesamiento de petición (Task)
  - Datos de usuario (UserData)

Las diferentes áreas tienen un tamaño determinado que es constante. En total suman un tamaño constante de 25.746 bytes para el bloque de datos.

Parámetro de conexión "Connection"

La instrucción "PE_WOL" reserva un recurso de conexión del área de "Open User Communication". Este se utiliza como conexión UDP. Para ello, deben definirse los siguientes parámetros en el bloque de datos:

- ID de conexión (parámetro "Connection.id")
  La ID de conexión es un número entero entre 1 y 32. El ajuste estándar es 31. Sirve para identificar los recursos de comunicación asignados por el firmware, como el búfer de transmisión y de recepción.
  La ID de conexión debe ser unívoca en toda la CPU.
- Número de puerto que se utiliza para la función "Wake On LAN" (parámetro "Header.PortNo")
  Número del puerto UDP a través del que se envía un paquete "Wake On LAN". Estos números de puerto forman parte de los recursos de comunicación identificados por la ID de conexión y asignados por el firmware. El ajuste estándar utilizado aquí para el puerto de 2189 no está asignado actualmente por la IANA. El número de puerto se transfiere a la configuración de conexión. Esto repercute en los parámetros "Connection.local_tsap_id[1]" y "Connection.rem_tsap_id[1]".
- ID de interfaz (parámetro "Connection.local_device_id")
  La ID de interfaz también forma parte de la descripción de la conexión. Esta ID identifica la interfaz de la CPU que se debe utilizar para esta conexión. Se admiten varios valores. No obstante, estos deben adaptarse a la CPU y la interfaz utilizadas:
  - B#16#01 con CPU ET200S o WinAC RTX con interfaz Ethernet en subslot IF1
  - B#16#02 con CPU 315(F)-2PN/DP o CPU 317(F)-2PN/DP
  - B#16#03 con CPU 319(F)-2PN/DP
  - B#16#05 con CPU 41x(F)-3PN/DP
  - B#16#06 con WinAC RTX con interfaz Ethernet en subslot IF2
  - B#16#0B con WinAC RTX con interfaz Ethernet en subslot IF3
  - B#16#0F con WinAC RTX con interfaz Ethernet en subslot IF4

Estructura del bloque de datos

El bloque de datos tiene la siguiente estructura:

Nombre	Tipo de datos	Offset	Comentario
Header	STRUCT	-	Información de encabezado
Update (1)	BOOL	0.0	Señal que indica una modificación en el área de datos. • True = indica una modificación por parte del usuario. • False = indica la aplicación de las modificaciones.
Initialized	BOOL	0.1	Señal que indica el final de la inicialización. • True = inicialización finalizada. • False = indica que el bloque no se ha inicializado.
LinkUp	BOOL	0.2	Indica la configuración correcta de la interfaz Ethernet. • True = interfaz lista para su uso. • False = interfaz aún no configurada.
LinkDown	BOOL	0.3	Indica una interfaz sin configurar. • True = la interfaz no está configurada. • False = la interfaz está configurada o se está configurando actualmente.
PROFINET_ID (1)	INT	2.0	ID del sistema PROFINET I/O
Reserved	ARRAY [1..37] OF BYTE	4.0	Reservado
LastDeviceID	INT	42.0	Contiene la ID de dispositivo más alta en este sistema PROFINET I/O.
PortNo (1)	INT	44.0	Número de puerto que se utiliza para la función "Wake On LAN" (valor predeterminado = 2189).
Connection	STRUCT	-	Contiene la configuración de la conexión "Wake on LAN".
block_length	WORD	46.0	Longitud de la estructura (siempre B#16#40).
id (1)	WORD	48.0	ID de conexión
connection_type (1)	BYTE	50.0	Tipo de conexión (UDP = B#16#13)
active_est (1)	BOOL	51.0	Establecimiento activo de la conexión (siempre pasivo con UDP)
local_device_id (1)	BYTE	52.0	Contiene la ID de interfaz (dependiente de la CPU).
local_tsap_id_len (1)	BYTE	53.0	Contiene la longitud en bytes del puerto UDP propio/local.
rem_subnet_id_len (1)	BYTE	54.0	Sin utilizar (siempre B#16#00)
rem_staddr_len (1)	BYTE	55.0	Contiene la longitud de la dirección IP remota o B#16#00.
rem_tsap_id_len (1)	BYTE	56.0	Contiene la longitud en bytes del puerto UDP remoto.
next_staddr_len (1)	BYTE	57.0	Contiene la longitud de la dirección del router predeterminado (no relevante).
local_tsap_id (1)	ARRAY[1..8] OF INT	58.0	Contiene el número del puerto propio/local.
rem_subnet_id (1)	ARRAY[1..6] OF BYTE	74.0	Sin utilizar (siempre B#16#00)
rem_staddr (1)	ARRAY[1..6] OF BYTE	80.0	Contiene la dirección IP remota.
rem_tsap_id (1)	ARRAY[1..8] OF INT	86.0	Contiene el número del puerto UDP remoto.

Nombre	Tipo de datos	Offset	Comentario
next_staddr (1)	ARRAY[1..6] OF BYTE	102.0	Irrelevante
spare (1)	WORD	108.0	
Device	ARRAY[1..256] OF STRUCT	110.0	Matriz de los dispositivos
Device	STRUCT	110.0	Contiene datos de cada dispositivo.
DiagID	WORD	110.0	Dirección de diagnóstico del dispositivo. Es asignada por la configuración hardware.
InterfaceID	WORD	112.0	Dirección de diagnóstico de la interfaz del dispositivo. Es asignada por la configuración hardware.
MACAdr	ARRAY[1..6] OF BYTE	114.0	Contiene la dirección MAC del dispositivo.
IPAdr	ARRAY[1..4] OF BYTE	120.0	Contiene la dirección IP del dispositivo.
OrderID	STRUCT	124.0	Contiene la OrderID del dispositivo.
MxLen	BYTE		Contiene la longitud máxima de la matriz.
ActLen	BYTE		Contiene la cantidad actual de datos en la matriz.
Data	ARRAY[1..20] CHAR		Contiene los datos.
PE	STRUCT	146.0	Datos específicos de PROFIenergy.
ModeID	BYTE	146.0	PE_MODE_ID conforme a especificación PROFIenergy.
Result	BYTE	147.0	PEErrorCode conforme a especificación PROFIenergy.
PauseTime (1)	TIME	148.0	Contiene el tiempo de pausa en ms.
TimeToPause	TIME	152.0	Contiene el tiempo que necesita el dispositivo para pasar al modo de pausa.
TimeToOperate	TIME	156.0	Contiene el tiempo que necesita el dispositivo para pasar al modo de operación.
MinSleepTime	TIME	160.0	Contiene el tiempo mínimo del dispositivo en PE_SLEEP_MODE.
SleepToOperate	TIME	164.0	Contiene el tiempo que necesita el dispositivo para estar listo para el servicio desde PE_SLEEP_MODE.
StatusOperate	BOOL	168.0	Indica el modo de operación del dispositivo.
StatusPause	BOOL	168.1	Indica el modo de pausa del dispositivo.
StatusSleep	BOOL	168.2	Indica el PE_SLEEP_MODE del dispositivo.
StatusTransitOK	BOOL	168.3	Indica la transición de un estado de energía a otro.
StatusInTransit	BOOL	168.4	Indica una transición de estado actual.
StatusTransitNOK	BOOL	168.5	Indica que ha fallado el cambio de estado.
StatusError	BOOL	168.6	Indica un error con el dispositivo.
StatusRetryEx	BOOL	168.7	Indica la ejecución incorrecta de un comando. No se intenta ejecutar de nuevo este comando.
CmdStartPause (1)	BOOL	169.0	Posiciona un comando START_PAUSE para este dispositivo en la lista de espera.
CmdEndPause (1)	BOOL	169.1	Posiciona un comando END_PAUSE para este dispositivo en la lista de espera.
CmdUpdateStatus (1)	BOOL	169.2	Posiciona un comando PEM_STATUS para este dispositivo en la lista de espera.

Nombre	Tipo de datos	Offset	Comentario
EnableSleep [1]	BOOL	169.3	Permite el PE_SLEEP_MODE para este dispositivo. • True = el dispositivo debe pasar a PE_SLEEP_MODE si la pausa dura lo suficiente. • False = el dispositivo no debe pasar a PE_SLEEP_MODE.
Services	WORD	170.0	Indica todos los servicios PROFIenergy soportados.
UserData [2]	ARRAY[1..24] OF BYTE	172.0	Datos definidos por el usuario
Task	STRUCT	196.0	Procesamiento de la petición
Cmd	BYTE	196.0	Bits internos para el procesamiento de la petición
CmdJ	BYTE	197.0	Bits internos para el procesamiento de la petición
TimeStart	BOOL	198.0	Inicia un tiempo de retardo.
TimeStarted	BOOL	198.1	Acaba de iniciarse un tiempo de retardo.
TimeDone	BOOL	198.2	Indica la finalización del tiempo de retardo.
Done	BOOL	198.3	Indica que la petición está finalizándose.
DelayedCmd	BOOL	198.4	Indica que todavía hay pendiente un comando retardado.
IsV1_0	BOOL	198.5	Indica que este dispositivo es un dispositivo Spec. V1.0.
IsWakeOnLAN	BOOL	198.6	Indica que este dispositivo se activa por "Wake On LAN".
RetryCount	BYTE	199.0	Contador de repeticiones de PE_COMMANDS
Duration	TIME	200.0	Contiene el valor de retardo en ms.
StartTime	TIME	204.0	Contiene el punto de inicio del tiempo de retardo.
MachineState	INT	208.0	Contiene el estado interno de la petición.

[1] Ajuste por parte del usuario.
[2] Uso libre para el usuario.

Parámetro STATUS

Parámetro STATUS

El valor de salida del parámetro STATUS se divide en tres áreas:

- Bits 31 a 24: MESSAGE
- Bits 23 a 16: LOCATION
- Bits 15 a 0: INFORMATION

En las siguientes tablas se explica el significado de los diferentes códigos de error de las tres áreas:

Tabla 9-178 Valores posibles de MESSAGE

Código de error (W#16#...)	Descripción
00	Ningún error.
50	Instrucción inicializada.

Código de error (W#16#...)	Descripción
51	Se determina la configuración del sistema PROFINET I/O.
52	La instrucción no ha podido localizar dispositivos configurados en el sistema PROFINET I/O.
53	Se determinan las direcciones lógicas de los dispositivos configurados.
54	Se lee la información de interfaz de los dispositivos.
55	Se determinan los datos I&M (solo registro 0) de los dispositivos configurados.
56	Se configura la interfaz PROFINET para el envío de "Wake on LAN" MagicPaket™ a través de UDP.
57	Se determina la compatibilidad PROFIenergy de los dispositivos conectados.
62	Detectada ID no válida del sistema PROFINET I/O. El número causante se indica en el campo INFORMATION.
70	La instrucción se ha inicializado y procesa peticiones. El valor del campo INFORMATION indica el número de las peticiones activas actualmente.
80	La instrucción se ha desinicializado al procesar peticiones. Esto ocurre generalmente cuando vuelve a cargarse el DB de instancia o bien el DB de usuario.
FF	Se ha producido un error desconocido.

Tabla 9-179 Valores posibles de LOCATION

Código de error (W#16#...)	Descripción
00	La instrucción no se ha inicializado o bien está inactiva.
70	La instrucción está esperando peticiones.
71	La instrucción introduce una petición en la lista de peticiones.
72	La instrucción prepara el envío de una petición.
73	La instrucción envía una petición a un dispositivo.
74	La instrucción espera la respuesta del dispositivo.
75	La instrucción evalúa la respuesta del dispositivo.
76	La instrucción elimina la petición de la lista de peticiones.
FF	Se ha producido un error desconocido.

Tabla 9-180 Valores posibles de INFORMATION

Código de error (W#16#...)	Descripción
0000	No hay información adicional disponible ni peticiones activas.
0001 -00FF	Actualmente se procesan 1 – 255 peticiones.
8001	Error en el 1.er parámetro
8002	Error en el 2.º parámetro
8003	Error en el 3.er parámetro
8004	Error en el 4.º parámetro

Código de error (W#16#...)	Descripción
8005	Error en el 5.º parámetro. Este error se indica si no hay ninguna interconexión con el DB de usuario o bien hay una no válida. Causas posibles: • El DB de usuario es demasiado pequeño. • El DB de usuario está protegido contra escritura. • El DB de usuario no está disponible en la RAM. • El DB de usuario no es válido para la CPU utilizada.
8100	Se han intentado posicionar más de las 256 peticiones posibles. Se trata de un error temporal que se soluciona con la finalización de algunas peticiones. La petición que se ha intentado posicionar no ha sido aceptada y debe volver a posicionarse.
8200	Se ha intentado enviar un comando PROFIenergy no válido o no soportado (PE_COMMAND).
84xx	Se ha producido un error de comunicación. El número del dispositivo que ha provocado el error se indica en "xx".
85xx	El dispositivo xx ha notificado un error. El número del dispositivo que ha provocado el error se indica en "xx".
8600	Actualmente no se soporta el método de WakeUp solicitado.
FFFF	Se ha producido un error desconocido.

Comandos PROFIenergy

Comando PE "Start_Pause"

Descripción

El comando PE "Start_Pause" inicia el modo de ahorro de energía. El comando Start_Pause puede utilizarse para:

- Conmutar la unidad PE desde el estado "Listo para el funcionamiento" (PE_ready_to_operate) a un modo de ahorro de energía (PE_energy_saving_mode).
- Conmutar la unidad PE de un modo de ahorro de energía a otro.
  El consumo de energía puede aumentar o disminuir al cambiar el modo de ahorro de energía.

Llamada del comando PE "Start_Pause"

La llamada del comando "Start_Pause" con la instrucción "PE_CMD (Página 6826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	1	Llamada del comando PE "Start_Pause".
CMD_MODIFIER	0	Para el comando "Start_Pause" no hay más especificaciones respecto a la llamada.

Parámetros	Valor	Descripción
CMD_PARA_LEN	4	Longitud del parámetro CMD_PARA de 4 bytes.
CMD_PARA	ANY	Puntero ANY al valor correspondiente a "Pause_Time" (TIME).

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase instrucción "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
PE_Mode_ID	de 1 a 255	BYTE	Número de identificación del modo de ahorro de energía
Reservado	0	BYTE	-

Comando PE "End_Pause"

Descripción

El comando PE "End_Pause" finaliza el modo de ahorro de energía en la unidad PE.

Llamada al comando PE "End_Pause"

La llamada del comando "End_Pause" con la instrucción "PE_CMD (Página 6826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	2	Llamada del comando PE "End_Pause".
CMD_MODIFIER	0	Para el comando "End_Pause" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
Time_to_operate	-	DWORD	Tiempo esperado para pasar al modo "PE_ready_to_operate".

Comando PE "Query_modes" - "List_Energy_Saving_Modes"

Descripción

Con el comando PE "Query_modes" y el subcomando (Modifier) "List_Energy_Saving_Modes" se indican todos los modos de ahorro de energía (PE_Mode_ID) soportados por la unidad PE.

El resultado de la consulta se escribe en forma de telegrama de respuesta en el bloque de datos referenciado en el parámetro RESPONSE_DATA.

Llamada del comando PE "Query_modes" - "List_Energy_Saving_Modes"

La llamada del comando "List_Energy_Saving_Modes" con la instrucción "PE_CMD (Página 6826)" se realiza con los parámetros siguientes:

Parámetro	Valor	Descripción
CMD	3	Llamada del comando PE "Query_modes".
CMD_MODIFIER	1	Especificación de la llamada del comando: seleccionar el subcomando "List_Energy_Saving_Modes" para mostrar el número y los tipos de modos de ahorro de energía soportados.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
Number_of_PE_ Mode_IDs	1	BYTE	El número de modos de ahorro de energía.
PE_Mode_IDs	-	Array [...] of BYTE	Matriz con las IDs de los modos de ahorro de energía soportados. El significado de las distintas IDs depende de la unidad PE.

Comando PE "Query_modes" - "Get_Mode"

Descripción

Con el comando PE "Query_modes" y el subcomando (Modifier) "Get_Mode" se indican los atributos para el modo de ahorro de energía activado actualmente.

Llamada del comando PE "Query_modes" - "Get_Mode"

La llamada del comando con la instrucción "PE_CMD" se realiza con los siguientes parámetros:

Parámetro	Valor	Descripción
CMD	3	Llamada del comando PE "Query_modes"
CMD_MODIFIER	2	Especificación de la llamada del comando: seleccionar el subcomando "Get_Mode" para mostrar el estado del modo activado en el momento actual.
CMD_PARA_LEN	1	Longitud del parámetro CMD_PARA de 1 byte.
CMD_PARA	ANY	ANY -Puntero al valor correspondiente a PE_MODE_ID.

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
PE_Mode_ID	• 0 Modo "PE_power_off" • 1...254 Modo de ahorro de energía de la unidad PE (específico de fabricante) • 255 Modo "PE_ready_to_operate"	BYTE	ID del modo de ahorro de energía activo en el momento actual.
PE_Mode_Attributes	Bit 0: • = 0: Solo están disponibles los valores estáticos de consumo de energía y de tiempo. • = 1: Están disponibles los valores dinámicos de consumo de energía y de tiempo. Bits de 1 a 7: • Reservado	BYTE	
Time_min_Pause [1]	Diferencia de tiempo sin fecha	DWORD	Tiempo mínimo de pausa para el modo PE. El tiempo mínimo de pausa es la suma de los valores de los siguientes atributos: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay Véase la descripción de "Parámetro PAUSE_TIME" de la instrucción "PE_START_END: Iniciar y finalizar modo de ahorro de energía (Página 6822)".

Atributo	Valor	Tipo de datos	Descripción
Time_to_Pause [1]	Diferencia de tiempo sin fecha	DWORD	Tiempo de desconexión: Tiempo transcurrido desde la llamada del modo de ahorro de energía hasta el inicio de este (tiempo de transición de PE_ready_to_operate a PE_energy_saving_mode). El tiempo de desconexión depende de la unidad PE.
Time_to_operate [1]	Diferencia de tiempo sin fecha	DWORD	Tiempo de conexión: Duración de la transición del modo de ahorro de energía (PE_energy_saving_mode) al modo de disponibilidad (PE_ready_to_operate). La duración es calculada dinámicamente por la unidad PE en el momento de la salida.
Time_min_length_ of_stay [1]		DWORD	Tiempo mínimo durante el cual el modo de ahorro de energía está activo en la unidad PE.
Time_max_length_ of_stay [1]		DWORD	Tiempo máximo durante el cual el modo de ahorro de energía está activo en la unidad PE.
Mode_Power_ Consumption [2]		REAL	Consumo de corriente de la unidad PE con el modo de ahorro de energía activado. Unidad: kW
Energy_ Consumption_ to_pause [2]		REAL	Consumo de energía de la unidad PE durante la transición del modo de disponibilidad (PE_ready_to_operate) al modo de ahorro de energía (PE_energy_saving_mode) Unidad: kWh
Energy_ Consumption_ to_operate [2]		REAL	Consumo de energía de la unidad PE durante la transición del modo de ahorro de energía (PE_energy_saving_mode) al modo de disponibilidad (PE_ready_to_operate) Unidad: kWh
[1] Si la duración es infinita, se devuelve el valor 0xFFFFFFFF. Si la duración es cero, se devuelve "0". [2] Si los datos de energía y consumo de corriente de la unidad PE no están definidos, se devuelve "0,0".			

Comando PE "PEM_Status"

Descripción

Con el comando "PEM_Status" se consulta el estado de un modo de ahorro de energía activado en el momento actual en una unidad PE.

Llamada al comando PE "PEM_Status"

La llamada del comando "PEM_Status" con la instrucción "PE_CMD" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	4	Llamada del comando PE "PEM_Status".
CMD_MODIFIER	0	Para el comando "PEM_Status" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
PE_Mode_ID_ Source	• 0 Modo "PE_power_off" • de 1 a 254 Modo de ahorro de energía de la unidad PE (específico de fabricante) • 255 Modo "PE_ready_to_operate"	BYTE	Modo en el que se encuentra la unidad PE antes de emitirse un comando PE.
PE_Mode_ID_ Destination	• 0 Modo "PE_power_off" • de 1 a 254 Modo de ahorro de energía de la unidad PE (específico de fabricante) • 255 Modo "PE_ready_to_operate"	BYTE	Modo en el que se encuentra la unidad PE después de ejecutarse un comando PE.
Time_to_operate	Diferencia de tiempo sin fecha.	DWORD	Tiempo de conexión: Duración de la transición del modo de ahorro de energía (PE_energy_saving_mode) al modo de disponibilidad (PE_ready_to_operate). La duración es calculada dinámicamente por la unidad PE en el momento de la salida.
Remaining_time_to_ destination	Diferencia de tiempo sin fecha.	DWORD	Tiempo restante para el paso al otro modo.
Mode_Power_ Consumption		REAL	Consumo de corriente de la unidad PE con el modo de ahorro de energía activado. Unidad: kW

Atributo	Valor	Tipo de datos	Descripción
Energy_ Consumption_ to_Destination		REAL	Consumo de energía para la transición PE momentánea Unidad: kWh
Energy_ Consumption_ to_operate		REAL	Consumo de energía de la unidad PE durante la transición del modo de ahorro de energía (PE_energy-saving mode) al modo de disponibilidad (PE_ready_to_operate) Unidad: kWh

Comando PE "PE_Identify"

Descripción

Con el comando PE "PE_Identify" se leen el número y la descripción de los comandos PE soportados por la unidad PE. El tipo y la cantidad de comandos que se soporten dependen de la unidad PE. Dado que el propio PE_Identify es un comando PE, en caso de respuesta positiva del dispositivo PE se muestran por lo menos tres comandos PE soportados: Start_Pause, End_Pause y PE_Identify.

Llamada al comando PE "PE_Identify"

La llamada del comando "PE_Identify" con la instrucción "PE_CMD (Página 6826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	5	Llamada del comando "PE_Identify".
CMD_MODIFIER	0	Para el comando "PE_Identify" no hay más especificaciones respecto a la llamada.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
Count [1]	6	BYTE	Número de comandos PROFIenergy soportados
Start_Pause	1	BYTE	Primer comando PE (Service_Request_ID) soportado
End_Pause	2	BYTE	...
Query_Modes	3	BYTE	...
PEM_Status	4	BYTE	...

Atributo	Valor	Tipo de datos	Descripción
PE_Identify	5	BYTE	...
Query_Measurement	16	BYTE	Último comando PE (Service_Request_ID) soportado
[1] El número de comandos soportados es específico del fabricante y depende de la unidad PE utilizada. Los valores indicados son un ejemplo de telegrama de respuesta en caso de que se soporten 6 comandos PE.			

Comando PE "Query_Measurement" - "Get_Measurement_list"

Descripción

Con el comando PE "Query_Measurement" y el subcomando (Modifier) "Get_measurement_list" se consultan los valores medidos que soporta la unidad PE. Los valores medidos soportados se guardan en forma de lista en el bloque de datos referenciado en el parámetro RESPONSE_DATA.

Llamada del comando PE "Query_Measurement" - "Get_Measurement_list"

La llamada del comando con la instrucción "PE_CMD (Página 6826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	16	Llamada del comando "Query_Measurement"
CMD_MODIFIER	1	Especificación de la llamada del comando: seleccionar el subcomando "Get_Measurement_List" para mostrar una lista de los valores medidos soportados.
CMD_PARA_LEN	0	Longitud del parámetro CMD_PARA de 0 bytes.
CMD_PARA	irrelevante	-

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
Count	-	BYTE	Número de Measurement-IDs
reserved	-	BYTE	
...			
Measurement_ID	-	WORD	Primer Measurement_ID soportado. El Measurement_ID es específico de fabricante. Encontrará más información en el manual de la unidad PE en cuestión.
Accuracy_Domain	-	BYTE	Ver la tabla "Rangos de precisión".
Accuracy_Class	-	BYTE	Ver las tablas "Clases de precisión".

Atributo	Valor	Tipo de datos	Descripción
Range	-	REAL	Especifica el valor de fin de escala para el valor medido (solo con rango de precisión 1). El atributo Range utiliza la misma unidad que se haya definido con el atributo Measurement_ID (para cada Measurement_ID se utiliza solo una unidad).
...			
Measurement_ID	-	WORD	Último Measurement_ID soportado
Accuracy_Domain	-	BYTE	Ver la tabla "Rangos de precisión".
Accuracy_Class	-	BYTE	Ver las tablas "Clases de precisión".
Range	-	REAL	Especifica el valor de fin de escala para el valor medido (solo con rango de precisión 1). El atributo Range utiliza la misma unidad que se haya definido con el atributo Measurement_ID (para cada Measurement_ID se utiliza solo una unidad).

Rangos de precisión

Rango de precisión	Descripción
0	Reservado
1	La desviación de precisión se expresa en porcentaje del valor de final de escala. El porcentaje de desviación posible se divide en clases de precisión (ver tabla: Clases de precisión de los rangos de precisión 1 y 2).
2	La desviación de precisión se expresa en porcentaje del valor medido real. El porcentaje de desviación posible se divide en clases de precisión (ver tabla: Clases de precisión de los rangos de precisión 1 y 2).
3	La precisión de medida se define según la norma IEC 61557-12. Las clases de potencia de funcionamiento para dispositivos de medición de potencia y supervisión (PMD) sin sensores externos y las clases de potencia del sistema para PMD con sensores externos están codificadas conforme a la tabla "Clases de precisión del rango de precisión 3".
4	La precisión se expresa de acuerdo con la norma EN 50470-3, capítulo 8 (ver también la tabla: Clases de precisión del rango de precisión 4).

Clases de precisión

Tabla 9-181 Clases de precisión de los rangos de precisión 1 y 2

Clase de precisión	0	1	2	3	4	5	6	7	8
Significado	Reservado	0,01%	0,02%	0,05%	0,1%	0,2%	0,5%	1%	1,5%

Clase de precisión	9	10	11	12	13	14	15	>15
Significado	2%	2,5%	3%	5%	10%	20%	>20%	No definido

Tabla 9-182 Clases de precisión del rango de precisión 3

Clase de precisión	0	1	2	3	4	5	6	7	8
Significado	Reservado	0,02	0,05	0,1	0,2	0,5	1	1,5	2

Clase de precisión	9	10	11	12	13	14	>13
Significado	2,5	3	5	10	20	20%	No definido

Tabla 9-183 Clases de precisión del rango de precisión 4

Clase de precisión	0	1	2	3	4	5	6	>7
Significado	Reservado	0,5	1,0	1,5	2,0	2,5	3,0	No definido

Comando PE "Query_Measurement" - "Get_Measurement_values"

Descripción

Con el comando PE "Query_Measurement" y el subcomando (Modifier) "Get_measurement_values" se indican los valores medidos soportados por la unidad PE. Los valores medidos se guardan en forma de lista en el bloque de datos referenciado en el parámetro RESPONSE_DATA.

Llamada del comando PE "Query_Measurement" - "Get_Measurement_values"

La llamada del comando con la instrucción "PE_CMD (Página 6826)" se realiza con los parámetros siguientes:

Parámetros	Valor	Descripción
CMD	16	Llamada del comando "Query_Measurement"
CMD_MODIFIER	2	Especificación de la llamada del comando: Seleccionar el comando "Get_Measurement_Values" para mostrar una lista de los valores medidos soportados.
CMD_PARA_LEN	0	En función del número de valores medidos. La longitud del parámetro se obtiene a partir del atributo count y la suma de las longitudes de los atributos de los valores medidos transferidos.
CMD_PARA	ANY	Puntero ANY hacia la estructura de datos con listado de los valores medidos que deben consultarse (véase "Parámetro CMD_PARA").

Parámetro CMD_PARA

La estructura que se indica mediante el puntero ANY en el parámetro CMD_PARA debe tener la estructura siguiente:

Atributo	Valor	Tipo de datos	Descripción
Count	-	BYTE	Número de valores medidos (Measurement-IDs)
reserved	0	BYTE	Sin utilizar
Measurement_ID	-	WORD	Primer valor medido consultado
...			
Measurement_ID	-	WORD	Último valor medido consultado

Telegrama de respuesta (Service Data Response)

Los datos siguientes del telegrama de respuesta de la unidad PE se escriben en el bloque de datos referenciado en el parámetro RESPONSE_DATA (véase "PE_CMD (Página 6826)"):

Atributo	Valor	Tipo de datos	Descripción
Count [1]	-	BYTE	Número de valores medidos (Measurement-IDs)
reserved	0	BYTE	Sin utilizar
Length_of_Structure	de 2 a 65535	WORD	Longitud de la estructura en bytes.
Measurement_Data_ Structure_ID	1 = simple value	BYTE	Define la siguiente estructura.
Measurement_ID	de 0 a 65535	WORD	ID del valor medido soportado.
Status_of_ Measurement_Value	de 1 a 3	BYTE	Estado del valor medido: • 1: Válido • 2: No soportado • 3: No válido
Transmission_Data_ Type	-	REAL	
End_of_demand	-	TOD	Sellado de tiempo opcional con el tipo de datos TimeOfDay.
...			
Length_of_Structure	-	WORD	Longitud de la estructura en bytes.
Measurement_Data_ Structure_ID	-	BYTE	Define la siguiente estructura.
Measurement_ID	-	WORD	ID del valor medido soportado.
Status_of_Measurem ent_Value	-	BYTE	Estado del valor medido: • 1: Válido • 2: No soportado • 3: No válido
Transmission_Data_ Type	-	REAL	
End_of_demand	-	TOD	Sellado de tiempo opcional con el tipo de datos TimeOfDay.
[1] Si la longitud de datos de los valores medidos consultados supera el tamaño de la PDU (Protocol Data Unit) del nivel de protocolo, los datos se transfieren de modo incompleto y solo se devuelven los valores medidos soportados.			

iDevice / iSlave

PE_I_DEV: Forzar comandos PROFIenergy en el iDevice

Descripción

La instrucción "PE_I_DEV" se utiliza para desarrollar el perfil PROFIenergy en el dispositivo IO inteligente (iDevice). Las funciones que en un dispositivo IO normal compatible con PROFIenergy, p. ej. el ET 200S, son realizadas por el firmware, en el iDevice son realizadas por la instrucción "PE_I_DEV" y los bloques auxiliares correspondientes:

- el programa de usuario del iDevice llama cíclicamente la instrucción "PE_I_DEV", que recibe todos los comandos PROFIenergy;
- la respuesta de PROFIenergy es generada por parametrización de un bloque auxiliar. La reacción en la pausa puede programarse libremente. Los datos de respuesta deben prepararse en 10 segundos, pues en caso contrario en el parámetro STATUS de la instrucción del controlador IO aparece "Stateconflict 0x80B5".

Para utilizar la instrucción no se requieren conocimientos explícitos del estándar PROFIenergy.

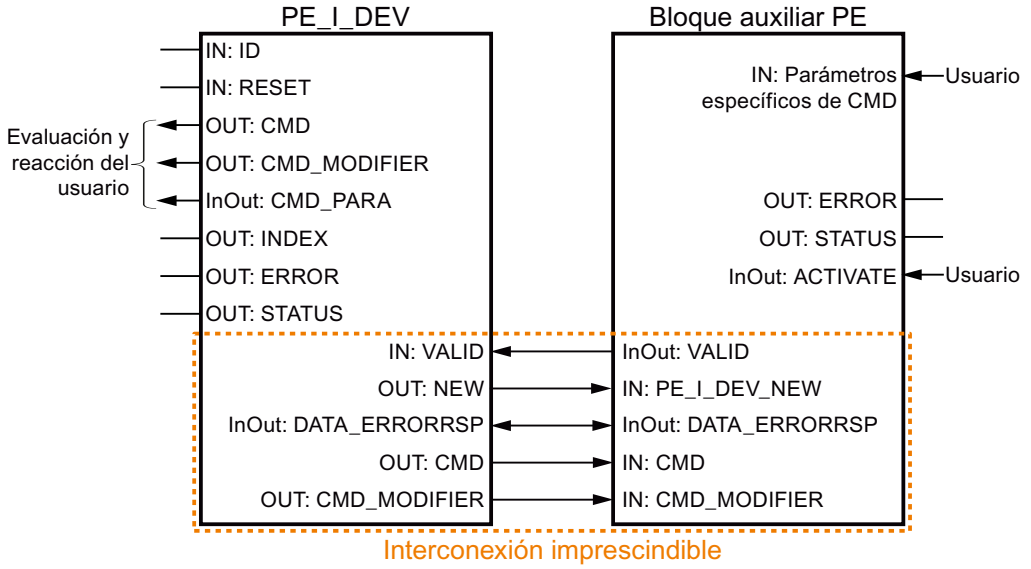
Bloques auxiliares PROFIenergy (bloques auxiliares PE)

Con los bloques auxiliares PE se genera el telegrama de respuesta. Para ello, indique los datos de respuesta (en texto explícito) en los parámetros de entrada del bloque correspondiente.

- Para cada comando PROFIenergy hay un bloque auxiliar correspondiente a una respuesta positiva:
  - comando PE "Start_Pause": PE_Start_RSP (Página 6860)
  - comando PE "End_Pause": PE_End_RSP (Página 6861)
  - comando PE "Query_modes" - "List_Energy_Saving_Modes": PE_List_Modes_RSP (Página 6862)
  - comando PE "Query_modes" - "Get_Mode": PE_Get_Mode_RSP (Página 6864)
  - comando PE "PEM_Status": PE_PEM_Status_RSP (Página 6866)
  - comando PE "PE_Identify": PE_Identify_RSP (Página 6867)
  - comando PE "Query_Measurement" - "Get_Measurement_list": PE_Measurement_List_RSP (Página 6869)
  - comando PE "Query_Measurement" - "Get_Measurement_values": PE_Measurement_Value_RSP (Página 6870)
- Con independencia del comando PROFIenergy utilizado, además hay un bloque auxiliar común para una respuesta negativa (véase PE_Error_RSP (Página 6858)).

Interconexión de los bloques auxiliares

La instrucción "PE_I_DEV" y los bloques auxiliares están adaptados mutuamente. Los parámetros se interconectan en parte de manera sencilla. El gráfico siguiente muestra qué parámetros deben interconectarse.



Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PE_I_DEV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RESET	Input	BOOL	I, Q, M, D, L	Resetea la instrucción.
ID	Input	DWORD	I, Q, M, D, L o constante	Dirección del iDevice La dirección puede tomarse de la configuración de hardware.
VALID	Input	BOOL	I, Q, M, D, L	Los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
CMD_PARA	Output	ANY	I, Q, M, D, L	Parámetros de: • Get mode: PE_mode_ID • Get measurement values: lista de Measurement_IDs (lista de IDs de las variables que deben leerse; es posible leer una sola variable o varias a la vez) Longitud máxima: 234 bytes
DATA_ERRORRSP	InOut	ANY	I, Q, M, D, L	Puntero al área de datos que contiene los datos de acuse del controlador PROFIenergy. Debe coincidir con el puntero utilizado en los bloques auxiliares.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
INDEX	Output	INT	I, Q, M, D, L	Número del registro PROFIenergy (0x80A0 fijo)
CMD	Output	INT	I, Q, M, D, L	Service-Request-ID del comando PROFIenergy en función del perfil PROFIenergy (véase "Parámetros CMD y CMD_MODIFIER"). Si se amplía el perfil PROFIenergy son posibles otros comandos PE (Service-Request-IDs).
CMD_ MODIFIER	Output	INT	I, Q, M, D, L	Subcomando PROFIenergy: • solo con CMD=3 o CMD=16, véase "Parámetros CMD y CMD_MODIFIER"; • en todos los demás comandos: "0". Si se amplía el perfil PROFIenergy son posibles otros subcomandos.
NEW	Output	BOOL	I, Q, M, D, L	Nuevos datos disponibles para el controlador PROFIenergy.
ERROR	Output	BOOL	I, Q, M, D, L	Comando finalizado con error.
STATUS	Output	DWORD	I, Q, M, D, L	Información de error (véase "Parámetro STATUS")

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros CMD y CMD_MODIFIER

CMD	CMD_ MODIFIER	Comando PROFIenergy	Descripción
01	0	Start_Pause	Inicia el modo de ahorro de energía o cambia a otro modo de ahorro de energía.
02	0	End_Pause	Sale del modo de ahorro de energía.
03	1	Query_Modes - List energy saving Modes	Muestra los modos de ahorro de energía soportados.
	2	Query_Modes - Get Mode	Muestra los atributos del modo de ahorro de energía activado en el momento actual.
04	0	PEM_Status	Consulta el estado del modo de ahorro de energía.
05	0	PE_Identify	Lectura del número de comandos PE soportados y descripción de los mismos.
16	1	Query_Measurement - Get_Measurement_List	Listado de los valores medidos soportados por la unidad PE.
	2	Query_Measurement - Get_Measurement_Values	Muestra los valores medidos de la unidad PE.

Parámetro STATUS

En el parámetro de salida STATUS se muestra información de error. Si se interpreta como ARRAY[1...4] of BYTE, la información de error tiene la estructura siguiente:

Elemento de campo	Nombre	Significado
STATUS[1]	Function_Num	Causa del error • B#16#00: sin error • B#16#DE: error al leer el registro • B#16#DF: error al escribir el registro • B#16#C0: mensaje de error de la instrucción "PE_I_DEV" o de las instrucciones de comunicación "RDREC (Página 6735)" y "WRREC (Página 6738)" empleadas internamente.
STATUS[2]	Error_Decode	Ubicación de la identificación de error • 80: error DPV1 según la norma IEC 61158-6 o error específico de instrucción
STATUS[3]	Error_Code_1	Identificación del error (con Error_Decode = 80): • B1: Write length error (error en la longitud de escritura o indicación de longitud insuficiente mediante el tipo de datos ANY).
STATUS[4]	Error_Code_2	Con error PROFINET: salida del mensaje de error del controlador IO. Si no se ha producido ningún error PROFINET, el valor de STATUS[4] = "0".

Nota

Mensajes de error de las instrucciones RDREC y WRREC

La instrucción "PE_I_DEV" utiliza para la comunicación las instrucciones "WRREC (Página 6738)" y "RDREC (Página 6735)". Los mensajes de error de estas instrucciones se emiten correspondientemente en los elementos de campo STATUS[1] a STATUS[4].

Consulte el significado de los códigos de error de las instrucciones "WRREC (Página 6738)" y "RDREC (Página 6735)" en la descripción del parámetro STATUS (Página 6748) correspondiente.

Módulos auxiliares de la instrucción PE_I_DEV

PE_Error_RSP: Generación de respuesta negativa a comando

Descripción

El bloque auxiliar "PE_Error_RSP" (Response with failure) genera una respuesta negativa si el comando solicitado no es soportado con carácter general o temporalmente. La respuesta se genera con independencia del comando solicitado.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Error_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
ERROR_CODE	Input	BYTE	I, Q, M, D, L o constante	Número de error
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Start_RSP: Generar respuesta a comando para iniciar pausa

Descripción

El bloque auxiliar "PE_Start_RSP" (Iniciar pausa) genera la respuesta al comando PE Start_Pause (Página 6844). La instrucción devuelve el estado de ahorro de energía que adopta el dispositivo (parámetro PE_MODE_ID).

Si las reacciones a pausas de distinta duración son diferentes, se puede indicar en la notificación mediante el estado de ahorro de energía adoptado (PE_Mode_ID = 1 para una pausa corta, PE_Mode_ID = 2 para una pausa más larga, etc.).

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Start_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
PE_MODE_ID	Input	BYTE	I, Q, M, D, L o constante	Modo PE que adopta el proceso
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_End_RSP: Generar respuesta a comando para final de pausa

Descripción

El bloque auxiliar "PE_End_RSP" genera la respuesta al comando PE End_Pause (Página 6845). Se devuelve como respuesta el tiempo necesario para cambiar del estado operativo actual al estado operativo "Ready_To_Operate".

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_End_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
Time_to_Operate	Input	DWORD	I, Q, M, D, L o constante	Tiempo para cambiar del estado operativo actual a "Ready_To_Operate".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_List_Modes_RSP: Generar respuesta con modos de ahorro de energía consultados

Descripción

El bloque auxiliar "PE_List_Modes_RSP" genera la respuesta al comando PE List_Energy_Saving_Modes (Página 6846). La respuesta generada contiene el número de los estados de ahorro de energía soportados y sus IDs.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_List_Modes_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
Number_of_PE_ Mode_IDs	Input	BYTE	I, Q, M, D, L o constante	Número de estados de ahorro de energía soportados. Valores permitidos: de 1 a 254
PE_MODE_ID	Input	ANY	I, Q, M, D, L	Señala el área en la que se guardan las IDs de los estados de ahorro de energía soportados (PE_Mode_ID). Área permitida: de 1 a 254.
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Get_Mode_RSP: Generar respuesta con datos de energía consultados

Descripción

El bloque auxiliar "PE_Get_Mode_RSP" genera la respuesta al comando Get_Mode (Página 6846). La respuesta contiene los tiempos y los datos de potencia o de energía de cada estado de ahorro de energía concreto.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Get_Mode_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
PE_Mode_ID	Input	BYTE	I, Q, M, D, L o constante	ID del modo de ahorro de energía activo en el momento actual.
Time_min_Pause	Input	DWORD	I, Q, M, D, L o constante	Duración mínima de pausa para este modo de ahorro de energía PE. Está formada por la suma de tres parámetros: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
Time_to_Pause	Input	DWORD	I, Q, M, D, L o constante	Duración desde el flanco en el parámetro START (véase "PE_I_DEV (Página 6855)") hasta que se alcanza el modo de ahorro de energía PE solicitado.
Time_to_Operate	Input	DWORD	I, Q, M, D, L o constante	Tiempo de conexión máximo hasta "PE_ready_to_operate". El parámetro "Time_to_operate" puede utilizarse directamente para los cálculos correspondientes. El valor puede ser un valor máximo estático o bien puede calcularlo dinámicamente la unidad PE.
Time_min_Lenght_ of_stay	Input	DWORD	I, Q, M, D, L o constante	Tiempo mínimo durante el que la unidad PE debe permanecer en este modo PE.
Time_max_Lenght_o f_stay	Input	DWORD	I, Q, M, D, L o constante	Tiempo máximo durante el que la unidad PE puede permanecer en este modo PE.
Mode_Power_ Consumption	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía en el modo PE actual, en [kW].
Energy_Consum_ to_Pause	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía desde "PE_ready_to_operate" hasta el modo PE actual, en [kWh].
Energy_Consum_ to_operate	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía desde el modo PE actual hasta "PE_ready_to_operate", en [kWh].
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_PEM_Status_RSP: Generar respuesta con estado PEM

Descripción

El bloque auxiliar "PE_PEM_Status_RSP" genera la respuesta al comando PEM_Status (Página 6848).

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_PEM_Status_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
PE_MODE_ID_ Source	Input	BYTE	I, Q, M, D, L o constante	Source y Destination para PEM_STATUS. Valores: • 0x00: PE_POWER_OFF • 0x01 – 0xFE: específico del fabricante • 0xFF: PE_READY_TO_OPERATE
PE_MODE_ID_ Destination	Input	BYTE	I, Q, M, D, L o constante	
Time_to_Operate[1]	Input	DWORD	I, Q, M, D, L o constante	Tiempo de conexión máximo hasta "PE_ready_to_operate". "Time_to_operate" puede utilizarse directamente para los cálculos correspondientes. El valor puede ser un valor máximo estático o bien puede calcularlo dinámicamente la unidad PE.
Remaining_time_ to_destination[1]	Input	DWORD	I, Q, M, D, L o constante	Opcionalmente: tiempo restante hasta el modo PE solicitado. Valor dinámico o valor máximo estático
Mode_Power_ Consumption[2]	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía en el modo PE actual, en [kW].
Energy_ Consumption_ to_Destination[2]	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía hasta el modo PE solicitado, en [kW].
Energy_ Consumption_ to_operate[2]	Input	DWORD	I, Q, M, D, L o constante	Consumo de energía desde el modo PE actual hasta "PE_ready_to_operate", en [kWh].

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea
[1] Si la duración es ilimitada, puede indicarse el valor máximo "0xFFFFFFFF". Si la duración es "cero", puede utilizarse "0x00". [2] Si el consumo de energía no está definido, puede indicarse "0,0".				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Identify_RSP: Generar respuesta con comandos PROFIenergy soportados

Descripción

El bloque auxiliar "PE_Identify_RSP" genera la respuesta al comando PE_Identify (Página 6850). En la respuesta al comando se indican los comandos PROFIenergy que se soportan. Tenga en cuenta que PE_IDENTIFY es en sí mismo un comando PE y debe indicarse correspondientemente en la respuesta.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Identify_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
Start_Pause	Input	BOOL	I, Q, M, D, L o constante	Un parámetro para cada comando PROFIenergy: • 0: este comando PE no se soporta • 1: este comando PE se soporta
End_Pause	Input	BOOL	I, Q, M, D, L o constante	
Query_Modes	Input	BOOL	I, Q, M, D, L o constante	
PEM_Status	Input	BOOL	I, Q, M, D, L o constante	
PE_Identify	Input	BOOL	I, Q, M, D, L o constante	
Query_ Measurement	Input	BOOL	I, Q, M, D, L o constante	
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Measurement_List_RSP: Generar respuesta con lista de valores medidos soportados

Descripción

El bloque auxiliar "PE_Measurement_List_RSP" genera la respuesta al comando Get_measurement_list (Página 6851). En la respuesta se indican los valores medidos (Measurement-IDs) que se soportan.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Measurement_List_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
Count	Input	BYTE	I, Q, M, D, L o constante	Número de valores medidos soportados (Measurement-IDs)
Measurement_List	Input	ANY	D	Puntero hacia la la matriz con las Measurement_IDs soportadas. Consulte la estructura de la matriz según el perfil PROFIenergy en: Comando PE "Query_Measurement" - "Get_Measurement_list" (Página 6851)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

PE_Measurement_Value_RSP: Generar respuesta con valores medidos solicitados

Descripción

El bloque auxiliar "PE_Measurement_Value_RSP" genera la respuesta al comando Get_measurement_values (Página 6853). En la respuesta se devuelven los valores de los valores medidos solicitados.

Parámetros

La tabla siguiente muestra los parámetros del bloque auxiliar "PE_Measurement_Value_RSP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PE_I_DEV_NEW	Input	BOOL	I, Q, M, D, L o constante	El parámetro debe estar interconectado con el parámetro de salida NEW de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar se ejecuta solo si el parámetro tiene el valor "1".
CMD	Input	INT	I, Q, M, D, L o constante	Service-Request-ID del comando PROFIenergy El parámetro debe estar interconectado con el parámetro de salida CMD de la instrucción "PE_I_DEV (Página 6855)".
CMD_MODIFIER	Input	INT	I, Q, M, D, L o constante	Subcomando PROFIenergy (la evaluación solo tiene lugar si CMD=3 o CMD=16). El parámetro debe estar interconectado con el parámetro de salida CMD_MODIFIER de la instrucción "PE_I_DEV".
Count	Input	BYTE	I, Q, M, D, L o constante	Número de valores medidos (Measurement_Values).
Measurement_Values	Input	ANY	D	Puntero hacia la matriz con los valores medidos (Measurement_IDs). Consulte la estructura de la matriz según el perfil PROFIenergy en Comando PE "Query_Measurement" - "Get_Measurement_values" (Página 6853)
ACTIVATE	InOut	BOOL	I, Q, M, D, L	Con un flanco ascendente en la entrada ACTIVATE, esta instrucción copia los parámetros de entrada en el área de datos DATA_ERROR_RSP. Después la instrucción restablece el parámetro. El parámetro debe fijarse en un tiempo de 10 segundos después de que en el parámetro NEW de la instrucción "PE_I_DEV (Página 6855)" se haya detectado un flanco ascendente.
VALID	InOut	BOOL	I, Q, M, D, L	El parámetro debe estar interconectado con la entrada VALID de la instrucción "PE_I_DEV (Página 6855)". El bloque auxiliar activa el parámetro cuando los datos de respuesta del controlador PROFIenergy están preparados y pueden enviarse.
DATA_ERRORRSP	InOut	ANY	D	Puntero hacia el área de datos en la que se guardan los datos de respuesta. El parámetro es idéntico al puntero de DATA_ERRORRSP de la instrucción "PE_I_DEV (Página 6855)". El área de datos direccionada contiene el telegrama PROFIenergy completo. Longitud mínima: 244 bytes

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	• "0": ningún error • "1": con errores
STATUS	Output	WORD	I, Q, M, D, L	• "0": ningún error • "0x80B1": error en la indicación de ANY, p. ej. área errónea

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parametrización del módulo

Escribir y leer registros

Principio

Algunos módulos poseen un área de datos del sistema a la que solo es posible acceder desde el programa mediante escritura. Esta área contiene registros con los números de 0 a 240, aunque no todos los módulos poseen todos los registros (ver la tabla siguiente).

Además, los módulos pueden poseer también un área de datos del sistema a la que solo es posible acceder desde el programa mediante lectura. Esta área contiene registros con los números de 0 a 240, aunque no todos los módulos poseen todos los registros.

Nota

Algunos módulos poseen ambas áreas de datos del sistema. Se trata de áreas distintas desde el punto de vista físico, que solo tienen en común la división lógica en registros.

Área de datos del sistema solo accesible por escritura

La tabla siguiente muestra el formato del área de datos del sistema solo accesible por escritura. Indica el tamaño que puede ocupar cada uno de los registros y con qué instrucciones pueden escribirse.

Número de registro	Contenido	Tamaño	Restricción	Con acceso de escritura mediante instrucción
0	Parámetros	Con S7-300: de 2 a 14 bytes	Con acceso de escritura solo con S7-400	WR_DPARM (Página 6883) PARM_MOD (Página 6877)
1	Parámetros	Con S7-300: de 2 a 14 bytes (DS0 y DS1 ocupan juntos exactamente 16 bytes)	-	WR_PARM (Página 6882) WR_DPARM (Página 6883) PARM_MOD (Página 6877)
de 2 a 127	Datos del usuario	Cada uno ≤ 240 bytes	-	WR_PARM (Página 6882) WR_DPARM (Página 6883) PARM_MOD (Página 6877) WR_REC (Página 6774)
de 128 a 240	Parámetros	Cada uno ≤ 240 bytes	-	WR_PARM (Página 6882) WR_DPARM (Página 6883) PARM_MOD (Página 6877) WR_REC (Página 6774)

Área de datos del sistema solo accesible por lectura

La tabla siguiente muestra el formato del área de datos del sistema solo accesible por lectura. Indica el tamaño que pueden ocupar los distintos registros y con qué instrucciones pueden leerse.

Número de registro	Contenido	Tamaño	Con acceso de lectura mediante instrucción
0	Datos de diagnóstico específicos de módulo (definidos de modo homogéneo para el sistema)	4 bytes	RDSYSST (Página 6963) (SZL_ID00B1H) RD_REC (Página 6769)
1	Datos de diagnóstico específicos de canal (incl. registro 0)	• Con S7-300: 16 bytes • Con S7-400: de 4 a 220 bytes	RDSYSST (Página 6963) (SZL_ID00B2H y 00B3H) RD_REC (Página 6769)
de 2 a 127	Datos del usuario	Cada uno ≤ 240 bytes	RD_REC (Página 6769)
de 128 a 240	Datos de diagnóstico	Cada uno ≤ 240 bytes	RD_REC (Página 6769)

Recursos del sistema

Si se lanzan consecutivamente varias transferencias de registro de funcionamiento asíncrono, se garantiza que todas las peticiones se ejecuten y que no se produzcan interferencias recíprocas.

En el momento en que se alcanza el límite de los recursos del sistema, se envía una notificación mediante RET_VAL. El estado de fallo temporal puede eliminarse repitiendo la petición.

El número máximo de peticiones activas "simultáneamente" para un mismo tipo de instrucción depende de la CPU.

RD_DPAR: Leer registro de un módulo

Descripción

Esta instrucción permite leer el registro con número INDEX del componente direccionado desde los datos del sistema configurados. Puede tratarse de un módulo central o de un componente descentralizado (PROFIBUS DP o PROFINET IO).

El valor TRUE del parámetro de salida VALID indica que el registro se ha transferido correctamente al área de destino RECORD. En ese caso, el parámetro de salida LEN contiene la longitud en bytes de los datos leídos.

Si se ha producido un error durante la transferencia del registro, se señaliza mediante el parámetro de salida ERROR . En ese caso, el parámetro de salida STATUS contiene la información de error.

Funcionamiento

La instrucción "RD_DPAR" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la transferencia del registro, debe llamarse "RD_DPAR" con REQ = 1.

Mediante el parámetro de salida BUSY y los bytes 2 y 3 del parámetro de salida STATUS se muestra el estado de la petición. Los bytes 2 y 3 de STATUS corresponden al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

La transferencia del registro se puede dar por finalizada cuando el parámetro de salida BUSY ha adoptado el valor FALSE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_DPAR":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Solicitud de lectura
LADDR	Input	WORD	I, Q, M, D, L o constante	Cualquier dirección lógica del módulo En el bit 15 se especifica si se trata de una dirección de entrada (bit 15 = 0) o de salida (bit 15 = 1).
INDEX	Input	INT	I, Q, M, D, L o constante	Número de registro
VALID	Output	BOOL	I, Q, M, D, L	El nuevo registro se ha recibido y es válido
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La petición todavía no ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: Se ha producido un error durante la lectura.
STATUS	Output	DWORD	I, Q, M, D, L	• Identificación de llamada (bytes 2 y 3) o código de error • Byte 1: B#16#00, si no hay ningún error. De lo contrario, ID de función de DPV1-PDU: En caso de error al leer registros B#16#DE, en caso de error al escribir registros B#16#DF. Si no se utiliza ningún elemento de protocolo DPV1: B#16#C0. • Byte 4: Extensión específica de fabricante de la identificación de error

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
LEN	Output	INT	I, Q, M, D, L	Longitud de la información del registro leída
RECORD	InOut	ANY	I, Q, M, D, L	Área de destino para el registro leído Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

Consulte también: PARM_MOD: Transferir registros de un módulo (Página 6877).

RD_DPARA: Leer registro de un módulo de forma asíncrona

Descripción

Esta instrucción permite leer el registro con número RECNUM del módulo seleccionado desde los datos del sistema configurados. El registro leído se guarda en el área de destino delimitada mediante el parámetro RECORD .

Funcionamiento

La instrucción "RD_DPARA" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse la instrucción con REQ = 1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_DPARA":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: solicitud de lectura
LADDR	Input	WORD	I, Q, M, D, L o constante	Cualquier dirección lógica del módulo. Si se trata de una dirección de salida, debe estar activado el bit más significativo.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 0 a 240)
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. Si no se han producido errores durante la transferencia, deben distinguirse los siguientes casos: • RET_VAL contiene la longitud en bytes del registro realmente leído, en caso de que el área de destino sea mayor que el registro leído. • RET_VAL contiene "0" en caso de que la longitud del registro leído sea igual a la longitud del área de destino.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La petición todavía no ha finalizado.
RECORD	Output	ANY	I, Q, M, D, L	Área de destino para el registro leído. Solo se admite el tipo de datos BYTE . Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Consulte también: PARM_MOD: Transferir registros de un módulo (Página 6877).

PARM_MOD: Transferir registros de un módulo

Descripción

Con la instrucción "PARM_MOD" se transfieren al módulo todos los registros que se hayan configurado. A este respecto es indiferente que los registros sean estáticos o dinámicos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PARM_MOD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ= 1: Solicitud de escritura
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección base lógica del módulo. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La escritura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Información de error

Dentro de la información de error "verdadera" (códigos de error W#16#8xyz), debe distinguirse entre dos casos:

- Errores temporales (códigos de error de W#16#80A2 a 80A4, 80Cx):
  En este tipo de errores es posible que el error se elimine sin intervención del usuario, es decir, puede bastar con llamar de nuevo la instrucción (varias veces si es necesario).
  Ejemplo de error temporal: En este momento los recursos necesarios están ocupados (W#16#80C3).
- Errores permanentes (códigos de error W#16#809x, 80A1, 80Bx, 80Dx):
  Los errores de este tipo no se pueden eliminar sin intervención del usuario. Solo tiene sentido volver a llamar la instrucción una vez que se haya eliminado el error.
  Ejemplo de error permanente: Longitud errónea del registro que se debe transferir (W#16#80B1).

Nota

Si, usando una de las instrucciones WR_PARM, WR_DPARM o PARM_MOD, se transfieren registros a un esclavo DPV1 o este funciona en modo DPV1, el maestro DP evalúa de la siguiente manera la información de error recibida desde el esclavo:

Si la información de error se encuentra en las áreas de W#16#8000 a W#16#80FF o de W#16#F000 a W#16#FFFF, el maestro DP reenvía la información de error a la instrucción. Si se encuentra fuera de estas áreas, reenvía a la instrucción el valor W#16#80A2 y suspende el esclavo.

Indicaciones para las instrucciones "RD_DPARM (Página 6881)", "WR_PARM (Página 6882)", "WR_DPARM (Página 6883)" y "PARM_MOD"

Parámetro RET_VAL

Código de error (W#16#...)	Explicación	Restricción
0000	Sin error	-
7000	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.	-
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".	Periferia descentralizada
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".	Periferia descentralizada
8090	La dirección base lógica indicada no es válida: No existe ninguna asignación en el SDB1/SDB2x, o no se trata de una dirección base.	-
8092	En la referencia ANY se ha indicado un tipo distinto a BYTE .	Solo con S7-400 para RD_DPARM (Página 6881) y WR_PARM (Página 6882)
8093	Esta instrucción no es válida para el módulo seleccionado con LADDR y IOID. (Se soportan los módulos S7-300 con S7-300, módulos S7-400 con S7-400, módulos S7-DP con S7-300 y S7-400.)	-
80A1	Acuse negativo al enviar el registro al módulo (módulo extraído durante la transferencia o defectuoso)	1)

Código de error (W#16#...)	Explicación	Restricción
80A2	Error de protocolo DP con Layer 2, posible error de hardware/interfaz en el esclavo DP	Periferia descentralizada 1)
80A3	Error de protocolo DP con User-Interface/User	Periferia descentralizada 1)
80A4	Fallo de comunicación en el bus K	El error se produce entre la CPU y una interfaz DP externa 1)
80B0	Instrucción no posible con este tipo de módulo, o el módulo no conoce el registro.	1)
80B1	La longitud del registro que se debe transferir es errónea. Con RD_DPARM (Página 6881): La longitud del área de destino delimitada mediante RECORD es insuficiente.	-
80B2	El slot configurado no está ocupado.	1)
80B3	El tipo de módulo real no coincide con el tipo de módulo configurado en el SDB1	1)
80C1	Los datos de la petición de escritura efectuada anteriormente en el módulo para el mismo registro no han sido procesados todavía por el módulo.	1)
80C2	En este momento el módulo está procesando el máximo posible de peticiones para una CPU.	1)
80C3	En este momento los recursos necesarios (memoria, etc.) están ocupados.	
80C4	Error temporal interno. No se ha podido ejecutar la petición. Repita la petición. Si este error se produce a menudo, compruebe posibles fallos de origen eléctrico en la configuración.	1)
80C5	Periferia descentralizada no disponible o desactivada.	Periferia descentralizada 1)
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano)	Periferia descentralizada 1)
80D0	En el SDB correspondiente no hay ninguna entrada para el módulo.	-
80D1	Este número de registro no está configurado en el correspondiente SDB para el módulo (se descartan los números >241).	-
80D2	Módulo no parametrizable según la identificación del módulo.	-
80D3	No se puede acceder al SDB porque no está presente.	-
80D4	Error de estructura del SDB: El puntero interno del SDB señala fuera del SDB	Solo con S7-300
80D5	El registro es estático.	Solo con "WR_PARM (Página 6882)"
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)	-
1) no existe con "RD_DPARM"		

RD_DPARM: Leer registro de los datos de sistema configurados

Descripción

Esta instrucción permite leer el registro con número RECNUMdel módulo direccionado desde los datos del sistema configurados. El registro leído se guarda en el área de destino delimitada mediante el parámetro RECORD.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_DPARM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección base lógica del módulo. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro (valores admisibles: de 0 a 240)
RET_VAL	Return	INT	I, Q, M, D, L	Longitud en bytes del registro leído, si el registro leído cabe en el área de destino y no se ha producido ningún error durante la transferencia. Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
RECORD	Output	ANY	I, Q, M, D, L	Área de destino para el registro leído. Solo se admite el tipo de datos BYTE .

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Consulte también: PARM_MOD: Transferir registros de un módulo (Página 6877).

WR_PARM: Escribir registro de un módulo

Descripción

Esta instrucción permite transferir el registro RECORD al módulo direccionado. Los parámetros que se transfieren al módulo no sobrescriben los parámetros creados mediante configuración.

Requisitos

El registro que se va a transferir no debe ser estático. Para saber qué registros de un módulo son estáticos, consulte la descripción del módulo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_PARM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Solicitud de escritura
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección base lógica del módulo. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro
RECORD	Input	ANY	I, Q, M, D, L	Registro Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RECORD siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La escritura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RECORD

Los datos que se deben transferir se leen del parámetro RECORD en la primera llamada. Si la transferencia del registro dura más que una llamada, en las llamadas consecutivas de la instrucción (para la misma petición), el contenido del parámetro RECORD ya no es relevante.

Parámetro RET_VAL

Consulte también: PARM_MOD: Transferir registros de un módulo (Página 6877).

Nota

Solo con S7-400

Si aparece el error general W#16#8544, esto solo significa que estaba bloqueado el acceso a por lo menos un byte del área de memoria de E/S que contiene el registro. Se ha proseguido la transferencia de datos.

WR_DPARM: Transferir registro

Descripción

Esta instrucción permite transferir el registro con número RECNUM desde los datos de configuración al módulo direccionado. A este respecto es indiferente que el registro sea estático o dinámico.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_DPARM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Solicitud de escritura
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección base lógica del módulo. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RECNUM	Input	BYTE	I, Q, M, D, L o constante	Número de registro

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La escritura todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Consulte también: PARM_MOD: Transferir registros de un módulo (Página 6877).

Alarmas

Alarma horaria

Usar las instrucciones SET_TINT, CAN_TINT, ACT_TINT, QRY_TINT

Definición

Una alarma horaria es el desencadenante de la llamada controlada por reloj de un OB de alarma horaria.

Requisitos para la llamada

Para que el sistema operativo pueda llamar un OB de alarma horaria, deben cumplirse los siguientes requisitos:

- El OB de alarma horaria debe estar parametrizado (fecha y hora de inicio, ejecución)
  - Mediante configuración o
  - con la instrucción "SET_TINT (Página 6887)" en el programa de usuario.
- El OB de alarma horaria debe estar activado
  - Mediante configuración o
  - con la instrucción "ACT_TINT (Página 6889)" en el programa de usuario.
- El OB de alarma horaria no debe haber sido desactivado mediante configuración.

- El OB de alarma horaria debe estar presente en la CPU.
- Si el ajuste se realiza mediante la instrucción "ACT_TINT (Página 6889)", la fecha y hora de inicio, en caso de ejecución **única**, no deben haber transcurrido; en caso de ejecución **periódica**, el OB de alarma horaria debe llamarse durante el próximo periodo transcurrido (momento de arranque + múltiplo de la duración del periodo).

Nota

La alarma horaria puede parametrizarse con la herramienta de configuración y activarse en el programa de usuario ("ACT_TINT (Página 6889)").

Finalidad de las instrucciones "SET_TINT", "CAN_TINT", "ACT_TINT" y "QRY_TINT"

Con estas instrucciones, las alarmas horarias se pueden

- Ajustar ("SET_TINT (Página 6887)")
- Anular ("CAN_TINT (Página 6888)")
- Activar ("ACT_TINT (Página 6889)")
- Consultar ("QRY_TINT (Página 6889)")

Repercusiones en la alarma horaria

La tabla siguiente muestra qué efectos producen las distintas situaciones en las alarmas horarias.

Si ...	entonces ...
se cambia el ajuste de una alarma horaria (Llamada de "SET_TINT (Página 6887)"),	Se anula automáticamente la alarma horaria en curso.
Se anula la alarma horaria (Llamada de "CAN_TINT (Página 6888)"),	Se borran la fecha y hora de inicio. Por ello la alarma horaria debe reajustarse antes de poder activarla.
El OB de alarma horaria no está presente en el momento de la llamada,	Se genera automáticamente un error de clase de prioridad, es decir, el sistema operativo llama el OB 85. Si el OB 85 no está presente, la CPU pasa al estado operativo STOP.
Se sincroniza el reloj o se adelanta la hora	Si al adelantar la hora se salta la fecha u hora de inicio: • El sistema operativo llama el OB 80.[1] • A continuación del OB 80 se llaman todos los OBs de alarma horaria (una sola vez, sea cual sea el número de periodos saltados), en caso de que no hayan sido manipulados en el OB 80.[2] Si el OB 80 no está presente, la CPU pasa al estado operativo STOP.

Si ...	entonces ...
Se sincroniza el reloj o se atrasa la hora	CPUs S7-400 y CPU 318: Si se han llamado OBs de alarma horaria en el periodo atrasado, no se llaman de nuevo al volver a recorrerse ese tiempo. CPUs S7-300: Se ejecutan todos los OBs de alarma horaria afectados.
[1] En la información de evento de arranque del OB 80 se indica de forma cifrada qué OBs de alarma horaria no se han podido llamar debido al adelanto de la hora. La hora indicada en la información de evento de arranque equivale al tiempo adelantado. [2] La hora indicada en la información de evento de arranque del OB de alarma horaria recuperado corresponde a la hora de inicio de la primera alarma horaria saltada.	

Comportamiento en rearranque completo (en caliente) o arranque en frío

En caso de rearranque completo (en caliente) o arranque en frío, se borran todos los ajustes de alarma horaria especificados en el programa de usuario mediante instrucciones. A partir de entonces solo son efectivos los parámetros ajustados por configuración.

Ejecución del OB de alarma horaria

La tabla siguiente muestra los distintos efectos del parámetro "Ejecución". Este parámetro se ajusta mediante configuración o con la instrucción "SET_TINT (Página 6887)" (parámetro de entrada PERIOD).

Ejecución del OB de alarma horaria	Reacción
Ninguna (Ajustable solo mediante configuración)	El OB de alarma horaria no se ejecuta aunque esté presente en la CPU. En el programa de usuario es posible reparametrizar, es decir, ajustar la alarma horaria ("SET_TINT (Página 6887)").
Una sola vez	Después de la llamada al OB de alarma horaria, la alarma queda anulada y puede reajustarse y activarse.
Periódicamente (Cada minuto, cada hora, cada día, cada semana, cada mes, cada año)	Si, en el momento de la activación, la fecha y hora de inicio ya han transcurrido, el OB de alarma horaria interrumpe el programa cíclico en el momento "Fecha/hora de inicio + múltiplo de la duración ajustada del periodo". Puede suceder, aunque es muy infrecuente, que el OB de alarma horaria todavía esté procesándose cuando se produce la siguiente llamada. Consecuencia: • Error de tiempo (el sistema operativo llama el OB 80; si el OB 80 no está presente, la CPU pasa al estado operativo STOP) • Se corrige el OB de alarma horaria.

SET_TINT: Ajustar alarma horaria

Descripción

Esta instrucción permite ajustar la fecha y hora de arranque de los bloques de organización de alarma horaria. Si al fijar la hora de inicio se indican segundos y milisegundos, la indicación se ignora y se sustituye por "0".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB que se inicia en el momento SDT + múltiplo de PERIOD (OB 10 a OB 17).
SDT	Input	DT	D, L	Si al fijar la fecha y hora de arranque se indican segundos y milisegundos, la indicación se ignora y se sustituye por 0. Si se desea fijar el arranque mensual de un OB de alarma horaria, solo pueden usarse para la fecha de inicio los días 1, 2, ... 28.
PERIOD	Input	WORD	I, Q, M, D, L o constante	Período transcurrido desde el punto de salida SDT : • W#16#0000 = Una vez • W#16#0201 = Una vez por minuto • W#16#0401 = Una vez por hora • W#16#1001 = Una vez al día • W#16#1201 = Una vez por semana • W#16#1401 = Una vez al mes • W#16#1801 = Una vez al año • W#16#2001 = A final de mes
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
8091	Parámetro erróneo SDT

Código de error (W#16#...)	Descripción
8092	Parámetro erróneo PERIOD
80A1	El momento inicial ajustado se encuentra en el pasado. (Este código de error solo aparece con PERIOD = W#16#0000.)
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

CAN_TINT: Anular alarma horaria

Descripción

La instrucción permite borrar la fecha y hora de inicio de un bloque de organización de alarma horaria. La alarma horaria se desactiva y el bloque de organización ya no se llama.

Para volver a utilizar la alarma horaria, vuelva a ajustar primero el momento de inicio (instrucción "SET_TINT (Página 6887)") y, después, active la alarma horaria (instrucción "ACT_TINT (Página 6889)").

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CAN_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB cuya fecha y hora de inicio se deben borrar (OB 10 a OB 17).
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
80A0	No se ha establecido ninguna fecha/hora de inicio para el OB de alarma horaria correspondiente.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

ACT_TINT: Activar alarma horaria

Descripción

La instrucción permite activar un bloque de organización de alarma horaria.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ACT_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB que se debe activar (OB 10 a OB 17).
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
80A0	No se ha ajustado ninguna fecha/hora de inicio para el OB de alarma horaria en cuestión.
80A1	El momento activado se encuentra en el pasado, y el error solo se produce si Ejecución=una sola vez.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

QRY_TINT: Consultar estado de alarma horaria

Descripción

La instrucción muestra el estado de un bloque de organización de alarma horaria en el parámetro de salida STATUS .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "QRY_TINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB cuyo estado se consulta (OB 10 a OB 17).
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la alarma horaria; véase la siguiente tabla.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Nota

Si RET_VAL tiene un valor diferente de cero, no se puede evaluar STATUS.

Bit	Valor	Significado
0	0	En RUN.
	1	Durante el arranque.
1	0	La alarma horaria está habilitada.
	1	La alarma horaria está bloqueada mediante "DIS_IRT (Página 6906)".
2	0	La alarma horaria no está activada o ha expirado.
	1	La alarma horaria está activada.
4	0	El OB con el número de OB indicado en el parámetro OB_NR no existe.
	1	El OB con el número de OB indicado en el parámetro OB_NR existe.
Otros		Siempre "0"

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Alarma de retardo

Usar alarmas de retardo

Definición

Después de llamar la instrucción "SRT_DINT (Página 6892)", el sistema operativo genera una alarma (es decir, llama el OB de alarma de retardo parametrizado) una vez transcurrido el tiempo de retardo ajustado.

Requisitos para la llamada

Para que el sistema operativo pueda llamar un OB de alarma de retardo, deben cumplirse los siguientes requisitos:

- El OB de alarma de retardo debe haber sido activado mediante la instrucción "SRT_DINT (Página 6892)".
- El OB de alarma de retardo no debe haber sido desactivado mediante configuración.
- El OB de alarma de retardo debe estar presente en la CPU.

Finalidad de las instrucciones "SRT_DINT", "CAN_DINT" y "QRY_DINT"

Con estas instrucciones, las alarmas de retardo se pueden

- Iniciar ("SRT_DINT (Página 6892)")
- Anular ("CAN_DINT (Página 6893)")
- Consultar ("QRY_DINT (Página 6894)").

Repercusiones en la alarma de retardo

La tabla siguiente muestra qué efectos producen las distintas situaciones en las alarmas de retardo.

Si ...	y ...	entonces ...
Se inicia una alarma de retardo (llamada de "SRT_DINT (Página 6892)")	La alarma de retardo ya se ha iniciado,	Se sobrescribe la alarma de retardo; se inicia de nuevo la alarma de retardo.
	El OB de alarma de retardo no está presente en el momento de la llamada,	El sistema operativo genera un error de clase de prioridad (llamada del OB 85; si el OB 85 no está presente, la CPU pasa al estado operativo STOP.)

Si ...	y ...	entonces ...
	Se ha iniciado la alarma en un OB de arranque y el tiempo de retardo expira antes de que la CPU esté en RUN,	La llamada al OB de alarma de retardo se pospone hasta que la CPU esté en RUN.
El tiempo de retardo ha transcurrido	Un OB de alarma de retardo iniciado anteriormente todavía está procesándose,	El sistema operativo genera un error de tiempo (llamada del OB 80; si el OB 80 no está presente, la CPU pasa al estado operativo STOP.)

Comportamiento en rearranque completo (en caliente) o arranque en frío

En rearranque completo (en caliente) o arranque en frío, se borran todos los ajustes de alarma de retardo especificados en el programa de usuario mediante instrucciones.

Inicio en un OB de arranque

Una alarma de retardo puede iniciarse en un OB de arranque. Para llamar el OB de alarma de retardo deben cumplirse dos condiciones:

- El tiempo de retardo ha expirado.
- La CPU está en el estado operativo RUN.

Si el tiempo de retardo ha expirado y la CPU todavía no se encuentra en el estado operativo RUN, la llamada del OB de alarma de retardo se pospone hasta que la CPU se encuentre en dicho estado operativo. Tras ello, el OB de alarma de retardo se llama antes de la primera instrucción en el OB Main [OB 1].

SRT_DINT: Iniciar alarma de retardo

Descripción

Esta instrucción permite iniciar una alarma de retardo que, una vez transcurrido el retardo parametrizado (parámetro DTIME), llama un OB de alarma de retardo.

En el parámetro SIGN se puede indicar un identificador específico de usuario para señalizar el inicio de la alarma de retardo. Los valores de DTIME y SIGN vuelven a aparecer en la información de evento de arranque del OB indicado cuando este se ejecuta.

El tiempo transcurrido entre la llamada de la instrucción "SRT_DINT" y el inicio del OB de alarma de retardo es como máximo **un milisegundo** menor que el tiempo parametrizado, siempre que la llamada no se retrase debido a eventos de interrupción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SRT_DINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB que se inicia una vez transcurrido un tiempo de retardo (OB 20 a OB 23).
DTIME	Input	TIME	I, Q, M, D, L o constante	Valor de tiempo del retardo (de 1 a 60000 ms) Pueden implementarse tiempos mayores, p. ej. utilizando un contador en un OB de alarma de retardo.
SIGN	Input	WORD	I, Q, M, D, L o constante	Identificación que aparece en la información de evento de arranque al llamar el OB de alarma de retardo
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
8091	Parámetro erróneo DTIME
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

CAN_DINT: Anular alarma de retardo

Descripción

Esta instrucción permite anular una alarma de retardo iniciada (ver "SRT_DINT (Página 6892)"). En tal caso, no se llama el OB de alarma de retardo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CAN_DINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB que se debe anular (OB 20 a OB 23).
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
80A0	No se ha iniciado la alarma de retardo.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

QRY_DINT: Consultar estado de una alarma de retardo

Descripción

Esta instrucción permite consultar el estado de una alarma de retardo. Las alarmas de retardo se administran con los bloques de organización OB 20 a OB 23.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "QRY_DINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
OB_NR	Input	INT	I, Q, M, D, L o constante	Número del OB cuyo estado se desea conocer (OB 20 a OB 23).
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro actual de RET_VAL contendrá un código de error.
STATUS	Output	WORD	I, Q, M, D, L	Estado de la alarma de retardo; véase la siguiente tabla.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Nota

Si RET_VAL tiene un valor diferente de cero, no se puede evaluar STATUS.

Bit	Valor	Significado
0	0	En RUN.
	1	Durante el arranque.
1	0	La alarma de retardo está habilitada.
	1	La alarma de retardo está bloqueada mediante "DIS_IRT (Página 6906)".
2	0	La alarma de retardo no está activada o ya ha caducado.
	1	La alarma de retardo está activada.
4	0	El OB con el número de OB indicado en el parámetro OB_NR no existe.
	1	Está cargado un OB con el número de OB especificado en el parámetro OB_NR.
Otros		Siempre "0"

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	Parámetro erróneo OB_NR
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Eventos de errores sincrónicos

Enmascarar eventos de errores sincrónicos

Introducción

Los eventos de errores sincrónicos son eventos de error de programación y acceso. Estos eventos de error se producen cuando se emplean en la programación áreas o números de

operando erróneos o direcciones erróneas. **Enmascarar** estos eventos de error significa lo siguiente:

- Los eventos de error enmascarados no desencadenan ningún OB de error y no provocan ninguna reacción sustitutiva.
- La CPU "anota" en un registro de estado de eventos los errores enmascarados que se han producido.

Para enmascarar errores se debe llamar la instrucción "MSK_FLT (Página 6902)".

Desenmascarar eventos de error significa anular un enmascaramiento efectuado anteriormente y borrar el bit correspondiente en el registro de estado de eventos de la clase de prioridad actual. El enmascaramiento se anula

- Llamando la instrucción "DMSK_FLT (Página 6903)".
- Cuando ha finalizado la clase de prioridad actual (solo con S7-400).

Si se produce un evento de error después de ser desenmascarado, el sistema operativo inicia el correspondiente OB de error. Para la reacción a errores de programación, debe programarse el OB 121, y para la reacción a errores de acceso, el OB 122.

Los eventos de error enmascarados y detectados se pueden leer con la instrucción "READ_ERR (Página 6903)".

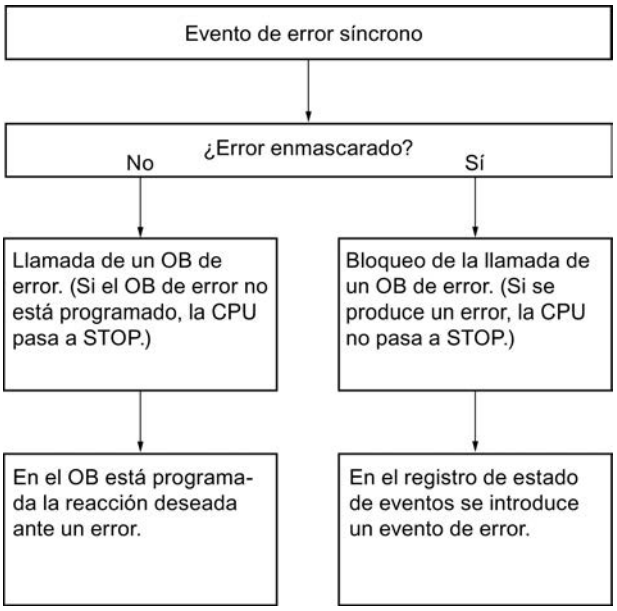
Nota

Independientemente de que estén enmascarados o desenmascarados, con S7-300 (exceptuando la CPU 318) los eventos de error se registran en el búfer de diagnóstico y se enciende el LED de error de grupo de la CPU.

Tratamiento de errores en general

Cuando se producen errores de programación o acceso en un programa de usuario, puede reaccionarse ante ello de diferentes maneras:

- Se puede programar un OB de error, que el sistema operativo llamará cuando se produzca un evento de error relevante.
- Se puede bloquear la llamada al OB de error por separado para cada clase de prioridad. En tal caso, la CPU no pasa a STOP cuando se produce un error relevante en la clase de prioridad bloqueada. La CPU guarda en un registro de estado de eventos los errores que se producen. Sin embargo, esta entrada en el registro no permite saber cuándo y cuántas veces se ha producido el error en cuestión.



Máscara de error

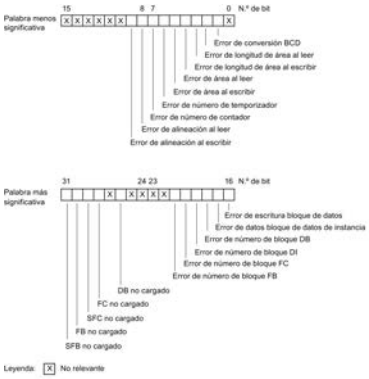
Los eventos de errores sincrónicos están asignados a un determinado patrón de bits, la **máscara de error**. Esta máscara de error se reproduce también en los parámetros de entrada y salida de las instrucciones "MSK_FLT (Página 6902)", "DMSK_FLT (Página 6903)" y "READ_ERR (Página 6903)".

En los eventos de errores sincrónicos se distingue entre los errores de programación y los de acceso, que pueden enmascararse con dos máscaras de error distintas. Encontrará las correspondientes máscaras de error en las siguientes imágenes.

Máscara de error de programación

La siguiente imagen muestra el patrón de bits de la máscara de error para errores de programación. La máscara para errores de programación se encuentra en los parámetros

"PRGFLT_..." (ver a continuación "Error de programación, palabra menos significativa" o "Error de programación, palabra más significativa").



Bits no relevantes

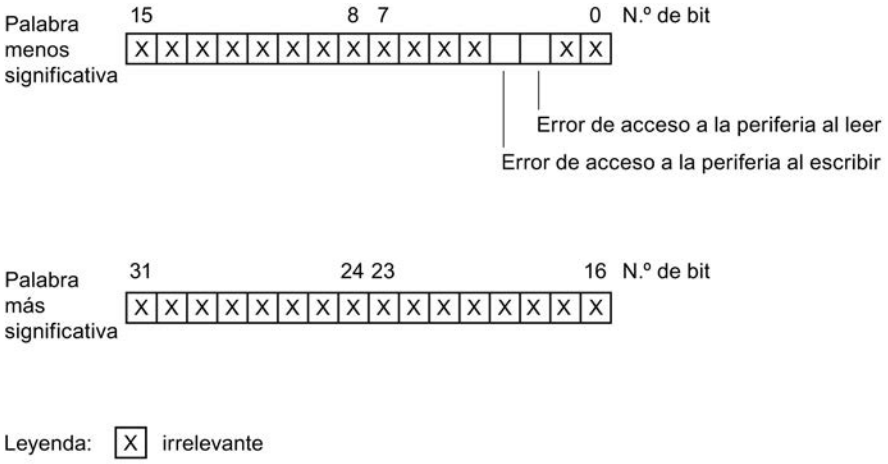
En la imagen superior, **x** significa para los ...

• ... Parámetros de entrada	para "MSK_FLT (Página 6902)", "DMSK_FLT (Página 6903)", "READ_ERR (Página 6903)"	= **"0"**
• ... Parámetros de salida	para "MSK_FLT (Página 6902)", "DMSK_FLT (Página 6903)"	= **"1"** para S7-300 = **"0"** para S7-400
	para "READ_ERR (Página 6903)"	= **"0"**

Máscara de error de acceso para todas las CPUs

La siguiente imagen muestra el patrón de bits de la máscara para errores de acceso (para todas las CPUs). La máscara de errores de acceso se encuentra en los parámetros ACCFLT_...

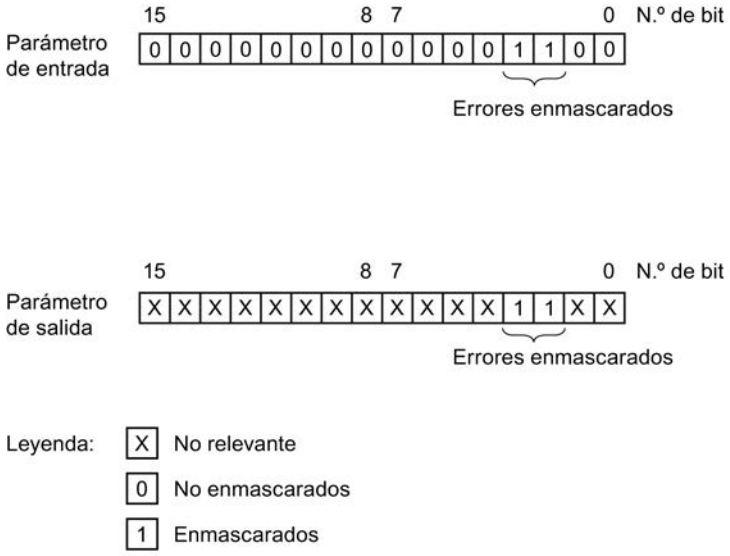
En las tablas siguientes encontrará una explicación de los errores de acceso (véanse a continuación).



Ejemplo

La siguiente imagen muestra el aspecto de la palabra menos significativa de la máscara para errores de acceso con todos los errores enmascarados (para todas las CPUs).

- Como parámetro de entrada para "MSK_FLT (Página 6902)"
- Como parámetro de salida para "MSK_FLT (Página 6902)"



Error de programación, palabra menos significativa

La tabla siguiente contiene los errores que están asignados a la palabra menos significativa de la máscara para errores de programación. Los distintos errores se muestran emparejados con sus posibles causas.

Posibles causas de errores de programación, palabra menos significativa

Error	ID de evento (W#16#...)	El error se produce ...
BCD - Error de conversión	2521	... Cuando el error que se debe convertir no es un número BCD (p. ej. 5E8).
Error de longitud de área durante lectura	2522	... Cuando un operando direccionado no se encuentra por completo dentro del área de operandos posible. Ejemplo: Debe leerse MW 320, aunque el área de marca solo ocupa 256 bytes.
Error de longitud de área durante escritura	2523	... Cuando un operando direccionado no se encuentra por completo dentro del área de operandos posible. Ejemplo: Debe escribirse un valor en MW 320, aunque el área de marca solo ocupa 256 bytes.

Error	ID de evento (W#16#...)	El error se produce ...
Error de área durante lectura	2524	... Cuando, en un direccionamiento indirecto, abarcando varias áreas, se especifica una identificación de área errónea para el operando. Ejemplo: Correcto: LAR1 P#E 12.0 L W[AR1, P#0.0] Erróneo: LAR1 P#12.0 L W[AR1, P#0.0] Con esta operación se notifica el error de área.
Error de área durante escritura	2525	... Cuando, en un direccionamiento indirecto, abarcando varias áreas, se especifica una identificación de área errónea para el operando. Ejemplo: Correcto: LAR1 P#E 12.0 T W[AR1, P#0.0] Erróneo: LAR1 P#12.0 T W[AR1, P#0.0] Con esta operación se notifica el error de área.
Error de número de temporizador	2526	... Cuando se accede a un temporizador no existente. Ejemplo:SI T [MW 0] con MW 0 = 129; se solicita iniciar el temporizador 129 aunque solo existen 128 temporizadores.
Error de número de contador	2527	... Cuando se accede a un contador no existente. Ejemplo: ZV Z [MW 0] con MW 0 = 600; se accede al contador 600, a pesar de que solo existen 512 contadores (CPU 416-1).
Error de alineación durante lectura	2528	... Cuando se direcciona un operando de byte, palabra o palabra doble con una dirección de bit ≠ 0. Ejemplo: Correcto: LAR1 P#M12.0 L B[AR1, P#0.0] Erróneo: LAR1 P#M12.4 L B[AR1, P#0.0]
Error de alineación durante escritura	2529	... Cuando se direcciona un operando de byte, palabra o palabra doble con una dirección de bit ≠ 0. Ejemplo: Correcto: LAR1 P#M12.0 T B[AR1, P#0.0] Erróneo: LAR1 P#M12.4 T B[AR1, P#0.0]

Error de programación, palabra más significativa

La tabla siguiente contiene los errores que están asignados a la palabra más significativa de la máscara para errores de programación. Los distintos errores se muestran emparejados con sus posibles causas.

Posibles causas de errores de programación, palabra más significativa

Error	ID de evento (W#16#...)	El error se produce ...
Error de escritura bloque de datos	2530	... Cuando el bloque de datos en el que se escribe está protegido contra escritura.
Error de datos bloque de datos de instancia	2531	... Cuando el bloque de datos de instancia en el que se escribe está protegido contra escritura.
Error de número de bloque DB	2532	... Cuando se abre un bloque de datos cuyo número es mayor que el número máximo admisible.
Error de número de bloque DI	2533	... Cuando se abre un bloque de datos de instancia cuyo número es mayor que el número máximo admisible.
Error de números de bloque FC	2534	... Cuando se llama una función cuyo número es mayor que el número máximo admisible.
Error de número de bloque FB	2535	... Cuando se llama un bloque de función cuyo número es mayor que el número máximo admisible.
DB no cargado	253A	... Cuando el bloque de datos que se desea abrir no está cargado.
Instrucción no cargada	253C a 253F	... Cuando la instrucción que se desea llamar no está cargada.

Error de acceso

La tabla siguiente contiene todos los errores que están asignados a la máscara de errores para errores de acceso (para todas las CPUs). Los distintos errores se muestran emparejados con sus posibles causas.

Error	ID de evento (W#16#...)	El error se produce ...
Error de acceso a la periferia al leer	2942	... Cuando la dirección del área de periferia no tiene asignado ningún módulo de señal. O bien ... Cuando el acceso a dicha área de periferia no se acusa dentro del tiempo de vigilancia de módulo ajustado (retardo de acuse).
Error de acceso a periferia al escribir	2943	... Cuando la dirección del área de periferia no tiene asignado ningún módulo de señal. o ... Cuando el acceso a dicha área de periferia no se acusa dentro del tiempo de vigilancia de módulo ajustado (retardo de acuse).

MSK_FLT: Enmascarar eventos de error síncrono

Descripción

Esta instrucción permite controlar la reacción de la CPU ante eventos de errores sincrónicos. Para ello se enmascaran los correspondientes eventos de errores sincrónicos (para las máscaras de error, ver Enmascarar eventos de errores sincrónicos (Página 6895)). Llamando a "MSK_FLT" se enmascaran los eventos de errores sincrónicos en la clase de prioridad actual.

Si en los parámetros de entrada se ajustan a "1" algunos bits de las máscaras de errores sincrónicos, los otros bits, ya ajustados anteriormente, mantienen el valor "1". Así se obtienen nuevas máscaras de error que pueden leerse a través de los parámetros de salida. Los eventos de errores sincrónicos enmascarados por el usuario no llaman ningún OB, sino que únicamente se consignan en un registro de estado de eventos. El registro de estado de eventos puede leerse con la instrucción "READ_ERR (Página 6903)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MSK_FLT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_SET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de programación que deben enmascararse
ACCFLT_SET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de acceso que deben enmascararse
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
PRGFLT_MASKED	Output	DWORD	I, Q, M, D, L	Errores de programación enmascarados
ACCFLT_MASKED	Output	DWORD	I, Q, M, D, L	Errores de acceso enmascarados

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	Ninguno de los errores estaba ya enmascarado.
0001	Por lo menos uno de los errores estaba ya enmascarado; pese a ello se enmascaran los otros errores.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

DMSK_FLT: Desenmascarar eventos de error síncrono

Descripción

Esta instrucción se utiliza para desenmascarar los eventos de error enmascarados con "MSK_FLT (Página 6902)". Para ello deben ajustarse a "1" en los parámetros de entrada los bits correspondientes de las máscaras de error. Llamando a "DMSK_FLT" se desenmascaran los correspondientes eventos de errores sincrónicos de la clase de prioridad actual. Al mismo tiempo se borran las entradas consultadas en el registro de estado de eventos. Las nuevas máscaras de error pueden leerse a través de los parámetros de salida.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DMSK_FLT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_ RESET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de programación que deben desenmascararse
ACCFLT_ RESET_MASK	Input	DWORD	I, Q, M, D, L o constante	Errores de acceso que deben desenmascararse
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
PRGFLT_ MASKED	Output	DWORD	I, Q, M, D, L	Errores de programación todavía enmascarados
ACCFLT_ MASKED	Output	DWORD	I, Q, M, D, L	Errores de acceso todavía enmascarados

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	Se han desenmascarado todos los errores especificados.
0001	Por lo menos uno de los errores no estaba enmascarado; pese a ello se desenmascaran los otros errores.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

READ_ERR: Leer registro de estado de eventos

Descripción

Esta instrucción permite leer el registro de estado de eventos. El registro de estado de eventos tiene una estructura equivalente a la de las máscaras de errores de programación y acceso

que pueden programarse como parámetros de entrada con "MSK_FLT (Página 6902)" y "DMSK_FLT (Página 6903)".

En los parámetros de entrada se especifican los eventos de errores sincrónicos que se desean consultar en el registro de estado de eventos. Llamando "READ_ERR" se leen (y al mismo tiempo se borran) las entradas deseadas del registro de estado de eventos.

En el registro de estado de eventos se puede averiguar cuáles de los errores sincrónicos enmascarados se han producido al menos una vez en la clase de prioridad actual. Un bit activado significa que el error sincrónico correspondiente ha aparecido por lo menos una vez.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "READ_ERR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PRGFLT_QUERY	Input	DWORD	I, Q, M, D, L o constante	Consulta de los errores de programación
ACCFLT_QUERY	Input	DWORD	I, Q, M, D, L o constante	Consulta de los errores de acceso
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
PRGFLT_CLR	Output	DWORD	I, Q, M, D, L	Errores de programación que se han producido
ACCFLT_CLR	Output	DWORD	I, Q, M, D, L	Errores de acceso que se han producido

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	Todos los errores consultados están enmascarados.
0001	Por lo menos uno de los errores consultados no está enmascarado.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Eventos de errores asíncronos

Retardar y bloquear alarmas y eventos de errores asíncronos

Finalidad de las instrucciones "DIS_IRT", "EN_IRT", "DIS_AIRT", "EN_AIRT"

Estas instrucciones se utilizan para las siguientes acciones referentes al procesamiento de alarmas y eventos de errores asíncronos:

- Bloquear su ejecución en todos los ciclos siguientes de la CPU con "DIS_IRT (Página 6906)", o
- Retardar clases de prioridad de mayor prioridad hasta el final del OB con "DIS_AIRT (Página 6906)"
- Volver a habilitar la ejecución con "EN_IRT (Página 6908)" o "EN_AIRT (Página 6910)".

El procesamiento de alarmas y eventos de errores asíncronos se programa en el programa de usuario. Para ello deben programarse los correspondientes OBs.

Ventaja de las instrucciones "DIS_AIRT" y "EN_AIRT"

Retardando eventos de alarma y de errores asíncronos de mayor prioridad con la instrucción "DIS_AIRT (Página 6909)" y habilitándolos con la instrucción "EN_AIRT (Página 6910)" se obtienen las siguientes ventajas para el usuario:

La CPU registra el número de retardos de alarma. Un retardo de eventos de alarma o de errores asíncronos programado por el usuario no se puede eliminar llamando FC estándar si en dichos FC estándar también se bloquean y vuelven a habilitar eventos de alarma y de errores asíncronos.

Clases de alarma

Las alarmas se dividen en varias clases. La tabla siguiente contiene todas las clases de alarma y los correspondientes OBs.

Clase de alarma	OB
Alarmas horarias	De OB 10 a OB 17
Alarmas de retardo	De OB 20 a OB 23
Alarmas cíclicas	De OB 30 a OB 38
Alarmas de proceso	De OB 40 a OB 47
Alarmas para DPV1	OB 55 a OB 57
Alarma de multiprocesamiento	OB 60
Alarmas de error de redundancia	OB 70, OB 72

Clase de alarma	OB
Alarmas de error asíncronas	OB 80 a OB 87 (ver abajo)
Alarmas de error sincrónicas	OB 121, OB 122 (Para enmascarar o desenmascarar el procesamiento de las alarmas de error sincrónicas, utilice las instrucciones "MSK_FLT (Página 6902)", "DMSK_FLT (Página 6903)" y "READ_ERR (Página 6903)")

Eventos de errores asíncronos

La tabla siguiente contiene todos los eventos de error asíncronos ante los que se puede reaccionar en el programa de usuario mediante un OB adecuado.

Eventos de errores asíncronos	OB
Error de tiempo (p. ej. rebase del tiempo de ciclo)	OB 80
Error de fuente de alimentación (p. ej. error de batería)	OB 81
Alarma de diagnóstico (p. ej. fusible defectuoso en un módulo de señal)	OB 82
Alarma de extracción/inserción	OB 83
Error de hardware en CPU (p. ej. error de interfaz)	OB 84
Error de ejecución del programa	OB 85
Fallo del rack	OB 86
Error de comunicación	OB 87

DIS_IRT: Bloquear evento de interrupción

Descripción

Esta instrucción permite bloquear el procesamiento de nuevos eventos de alarma y de errores sincrónicos. Bloquear significa que el sistema operativo de la CPU, al producirse un evento de interrupción,

- ni llama un OB de alarma o un OB de error asíncrono,
- ni desencadena la reacción definida en caso de que no haya ningún OB de alarma ni ningún OB de error asíncrono programado.

El bloqueo del procesamiento de eventos de alarma o de errores asíncronos es válido en todas las clases de prioridad. El bloqueo solo puede eliminarse con la instrucción "EN_IRT (Página 6908)" o con un rearranque completo (en caliente) o un arranque en frío.

El hecho de que el sistema operativo registre o no el evento de alarma o de errores asíncronos en el búfer de diagnóstico depende del parámetro de entrada MODE seleccionado.

Nota

Tenga en cuenta que al programar la instrucción "DIS_IRT" se descartan todas las alarmas que se produzcan.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DIS_IRT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	Indica qué eventos de alarma y errores asíncronos se bloquean.
OB_NR	Input	INT	I, Q, M, D, L o constante	Número de OB
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

MODE (B#16#...)	Significado
00	Se bloquean todos los eventos de alarma y errores asíncronos nuevos que se produzcan. (Los eventos de errores sincrónicos no se bloquean.) Asigne al parámetro OB_NR el valor "0". Se siguen registrando los eventos en el búfer de diagnóstico.
01	Se bloquean todos los eventos nuevos que se produzcan de la clase de alarma indicada. Para identificar la clase de alarma, haga lo siguiente: • Alarmas horarias: 10 • Alarmas de retardo: 20 • Alarmas cíclicas: 30 • Alarmas de proceso: 40 • Alarmas para DPV1: 50 • Alarma de multiprocesamiento: 60 • Alarmas de error de redundancia: 70 • Alarmas de error asíncronas: 80 Se siguen registrando los eventos en el búfer de diagnóstico.
02	Se bloquean todos los eventos nuevos que se produzcan de la alarma indicada. Para identificar la alarma se usa el número de OB. Se siguen registrando los eventos en el búfer de diagnóstico.
80	Todos los eventos de alarma y errores asíncronos nuevos que se producen se bloquean y no se registran en el búfer de diagnóstico. Asigne al parámetro OB_NR el valor "0". El sistema operativo registra el evento W#16#5380 en el búfer de diagnóstico.
81	Todos los eventos nuevos que se producen de la clase de alarma indicada se bloquean y no se registran ya en el búfer de diagnóstico. El sistema operativo registra el evento W#16#5380 en el búfer de diagnóstico.
82	Todos los eventos nuevos que se producen de la alarma indicada se bloquean y tampoco se registran ya en el búfer de diagnóstico. El sistema operativo registra el evento W#16#5380 en el búfer de diagnóstico.

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	El parámetro de entrada OB_NR contiene un valor no permitido.
8091	El parámetro de entrada MODE contiene un valor no permitido.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

EN_IRT: Habilitar evento de interrupción

Descripción

La instrucción permite habilitar el procesamiento de nuevos eventos de alarma y errores asíncronos que se ha bloqueado anteriormente con la instrucción "DIS_IRT (Página 6906)". Habilitar significa que el sistema operativo de la CPU, al producirse un evento de interrupción,

- lama un OB de alarma o un OB de error asíncrono
  o
- dispara la reacción definida en caso de que no haya ningún OB de alarma ni ningún OB de error asíncrono programado.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "EN_IRT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	BYTE	I, Q, M, D, L o constante	Indica qué eventos de alarma y errores asíncronos se habilitan (ver abajo).
OB_NR	Input	INT	I, Q, M, D, L o constante	Número de OB
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

MODE	Significado
0	Se habilitan todos los eventos de alarma y errores asíncronos nuevos que se produzcan.
1	Se habilitan todos los eventos nuevos que se produzcan de la clase de alarma indicada. Para identificar la clase de alarma, haga lo siguiente: • Alarmas horarias: 10 • Alarmas de retardo: 20 • Alarmas cíclicas: 30 • Alarmas de proceso: 40 • Alarmas para DPV1: 50 • Alarma de multiprocesamiento: 60 • Alarmas de error de redundancia: 70 • Alarmas de error asíncronas: 80
2	Se habilitan todos los eventos nuevos que se produzcan de la alarma indicada. Para identificar la alarma se usa el número de OB.

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	No ha ocurrido ningún error.
8090	El parámetro de entrada OB_NR contiene un valor no permitido.
8091	El parámetro de entrada MODE contiene un valor no permitido.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

DIS_AIRT: Retardar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad

Descripción

Esta instrucción permite retardar el procesamiento de OBs de alarma y OBs de errores asíncronos cuya prioridad sea mayor que la del OB actual. La instrucción "DIS_AIRT" puede llamarse varias veces en un OB. El sistema operativo cuenta las llamadas a la instrucción. El retardo de procesamiento se aplica hasta que, usando la instrucción "EN_AIRT (Página 6910)", se eliminan todos los retardos de procesamiento de OBs de alarma y OBs de errores asíncronos definidos con "DIS_AIRT", **o** acaba de procesarse el OB actual.

Los eventos de alarma o de errores asíncronos pendientes se procesan en cuanto se elimina el retardo de procesamiento con la instrucción "EN_AIRT (Página 6910)" o finaliza el procesamiento del OB actual.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DIS_AIRT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Número de retardos (= número de llamadas de DIS_AIRT)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

La tabla siguiente contiene el valor de retorno de "DIS_AIRT", que se devuelve a través del parámetro RET_VAL .

Valor de retorno	Descripción
n	"Una vez finalizada la instrucción, n" muestra el número de retardos de procesamiento, es decir, de llamadas de "DIS_AIRT" (el procesamiento de alarmas vuelve a habilitarse cuando n = 0; ver "EN_AIRT (Página 6910)").

EN_AIRT: Habilitar tratamiento de eventos de alarma y errores asíncronos de mayor prioridad

Descripción

La instrucción permite habilitar el procesamiento de eventos de alarma y errores asíncronos de mayor prioridad que se ha retardado anteriormente con la instrucción "DIS_IRT (Página 6909)". Para ello deben finalizarse uno por uno todos los retardos de procesamiento con la instrucción "EN_AIRT".

Ejemplo

Si p. ej. se han retardado alarmas 5 veces con 5 llamadas de "DIS_AIRT (Página 6909)", para eliminar todos esos retardos deben usarse 5 llamadas a "EN_AIRT".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "EN_AIRT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Número de retardos todavía programados después de la ejecución de "EN_AIRT" o aviso de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Para saber cómo evaluar la información de error del parámetro RET_VAL , consulte el capítulo Evaluación de errores con el parámetro de salida RET_VAL (Página 5598). En ese capítulo encontrará también la información general de error de las instrucciones. La tabla siguiente contiene el valor de retorno específico de "EN_AIRT", o la información de error que puede devolverse a través del parámetro RET_VAL .

Valor de retorno y error	Descripción
n	"Una vez finalizada la instrucción, n" muestra el número de retardos de procesamiento, es decir, de llamadas de "EN_AIRT" (el procesamiento de alarmas vuelve a habilitarse cuando n = 0).
W#16#8080	Aunque el procesamiento de la alarma ya se ha habilitado anteriormente, se ha llamado "EN_AIRT".

Evento de multiprocesamiento

MP_ALM: Alarma de multiprocesamiento

Descripción

En el modo de multiprocesamiento, la llamada a esta instrucción dispara la alarma de multiprocesamiento. Esto da lugar al inicio sincronizado del OB 60 en todas las CPUs correspondientes. En el modo monoprocesador y en el funcionamiento en rack segmentado, el OB 60 solo se inicia en la CPU en la que se ha llamado la instrucción.

Con el parámetro de entrada JOB puede identificarse la causa de la alarma de multiprocesamiento deseada. Esta identificación de petición se transfiere a todas las CPUs correspondientes y puede evaluarse en el OB de multiprocesamiento (OB 60).

Esta instrucción puede llamarse en cualquier punto del programa. Sin embargo, la llamada solo tiene sentido en el estado operativo RUN, por lo que en el estado operativo ARRANQUE está bloqueada. Esto se notifica mediante un valor de función.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "MP_ALM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
JOB	Input	BYTE	I, Q, M, D, L o constante	Identificador de petición Valores posibles: de 1 a 15
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	El parámetro de entrada JOBcontiene un valor no permitido.
80A0	El procesamiento del OB 60 de la alarma de multiprocesamiento anterior todavía no ha finalizado en la CPU propia o en otra CPU.
80A1	Estado operativo erróneo (ARRANQUE en lugar de RUN)
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Avisos

Formato de los datos de diagnóstico

Registros 0 y 1 de los datos del sistema

Los datos de diagnóstico de un módulo se encuentran en los registros 0 y 1 del área de datos del sistema (ver Escribir y leer registros (Página 6872)):

- El registro 0 contiene 4 bytes de datos de diagnóstico, que describen el estado actual de un módulo de señal.
- El registro 1 contiene
  - los 4 bytes de datos de dispositivo que también se encuentran en el registro 0, y
  - los datos de diagnóstico específicos de módulo.

Formato y contenido de los datos de diagnóstico

A continuación se describen el formato y el contenido de los distintos bytes de los datos de diagnóstico. Se aplica la siguiente regla general: Cuando se produce un error, el bit correspondiente se ajusta a "1".

Byte	Bit	Significado	Observación	Registro
0	0	Fallo de módulo		0 y 1
	1	Error interno		
	2	Error externo		
	3	Existe error de canal		
	4	Falta tensión auxiliar externa		
	5	Falta conector frontal		
	6	Falta parametrización		
	7	Parámetros erróneos en el módulo		
1	0 a 3	Clase de módulo	0101: Módulo analógico 0000: CPU 1000: Módulo de función 1100: CP 1111: Módulo digital 0011: Esclavo DP normalizado 1011: Esclavo I 0100: IM	0 y 1
	4	Existe información de canal		
	5	Existe información de usuario		
	6	Alarma de diagnóstico de sustituto		
	7	Mantenimiento necesario (solo con PROFINET IO)		
2	0	Módulo de memoria erróneo o ausente		0 y 1
	1	Fallo de comunicación		
	2	Estado operativo	0: RUN 1: STOP	
	3	La vigilancia del tiempo de ciclo se ha disparado		
	4	Fallo de la tensión de alimentación interna del módulo		
	5	Batería vacía		
	6	Fallo de todo el respaldo		
	7	Mantenimiento solicitado (solo con PROFINET IO)		
3	0	Fallo de aparato de ampliación		0 y 1
	1	Fallo de procesador		
	2	Error de EPROM		
	3	Error de RAM		
	4	Error de ADU/DAU		
	5	Fallo de fusible		

Byte	Bit	Significado	Observación	Registro
	6	Alarma de proceso perdida		
	7	Reservado		
4	0 a 6	Tipo de canal	B#16#70: Entrada digital B#16#72: Salida digital B#16#71: Entrada analógica B#16#73: Salida analógica B#16#74: FM-POS B#16#75: FM-REG B#16#76: FM-ZAEHL B#16#77: FM-TECHNO B#16#78: FM-NCU B#16#79: a B#16#7D: reservado B#16#7E: US300 B#16#7F: reservado	1
	7	¿Existe otro tipo de canal?	0: No 1: Sí	
5	0 a 7	Número de bits de diagnóstico que un módulo devuelve por canal.	El número de bits de diagnóstico por canal se redondea a los límites de byte.	1
6	0 a 7	Número de canales del mismo tipo en un módulo	Si en un módulo existen diferentes tipos de canales, para cada tipo de canal se repite la estructura a partir del byte 4 en el registro 1.	1
7	0	Error de canal, canal 0/grupo de canales 0	Primer byte del vector de error de canal (la longitud del vector de error de canal se define en función del número de canal y se redondea a los límites de byte.)	1
	1	Error de canal, canal 1/grupo de canales 1		
	2	Error de canal, canal 2/grupo de canales 2		
	3	Error de canal, canal 3/grupo de canales 3		
	4	Error de canal, canal 4/grupo de canales 4		
	5	Error de canal, canal 5/grupo de canales 5		
	6	Error de canal, canal 6/grupo de canales 6		
	7	Error de canal, canal 7/grupo de canales 7		
...	-	Errores específicos de canal (ver Formato de los datos de diagnóstico específicos de canal (Página 6914)):		1

Formato de los datos de diagnóstico específicos de canal

Errores específicos de canal

A partir del byte que se encuentra justo a continuación del vector de error de canal, se muestran los errores específicos de canal para cada canal del módulo. A continuación se

muestra el formato del diagnóstico específico de canal para los distintos tipos de canal. Para la asignación de bits se aplica la siguiente regla:

- 1 = error
- 0 = sin error

Canal de entrada analógica

Byte de diagnóstico para un canal de entrada analógica

Bit	Significado	Observación
0	Error de configuración/ parametrización	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x50
1	Error síncrono (error modo común)	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x51
2	Cortocircuito P	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x52
3	Cortocircuito M	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x53
4	Rotura de hilo	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x54
5	Error en canal de referencia	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x55
6	Rebase por defecto del rango de medición	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x56
7	Rebase por exceso del rango de medición	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x57

Canal de salida analógica

Byte de diagnóstico para un canal de salida analógica

Bit	Significado	Observación
0	Error de configuración/ parametrización	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x60
1	Error síncrono (error modo común)	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x61
2	Cortocircuito P	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x62
3	Cortocircuito M	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x63
4	Rotura de hilo	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x64
5	0	Reservado
6	Falta tensión de carga	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x66
7	0	Reservado

Canal de entrada digital

Byte de diagnóstico para un canal de entrada digital

Bit	Significado	Observación
0	Error de configuración/ parametrización	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x70
1	Defecto a masa	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x71
2	Cortocircuito P (sensor)	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x72
3	Cortocircuito M	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x73
4	Rotura de hilo	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x74
5	Falta alimentación del sensor	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x75
6	0	Reservado
7	0	Reservado

Canal de salida digital

Byte de diagnóstico para un canal de salida digital

Bit	Significado	Observación
0	Error de configuración/ parametrización	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x80
1	Defecto a masa	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x81
2	Cortocircuito P	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x82
3	Cortocircuito M	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x83
4	Rotura de hilo	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x84
5	Actuación fusible	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x85
6	Falta tensión de carga	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x86
7	Sobretemperatura	Notificable mediante WR_USMSG (Página 6947) y EVENTN = W#16#8x87

Instrucciones para generar avisos de PLC con DB de instancia

Introducción a la creación de avisos de PLC con instrucciones

Instrucciones para generar avisos de bloque

Para generar un aviso de PLC, se debe llamar en el programa una de las siguientes instrucciones:

- "NOTIFY (Página 6928)"
- "NOTIFY_8P (Página 6920)"
- "ALARM (Página 6930)"
- "ALARM_8 (Página 6923)"
- "ALARM_8P (Página 6925)"

Estas instrucciones tienen las siguientes características:

- Con "NOTIFY (Página 6928)" y "NOTIFY_8P (Página 6920)", todo cambio de señal 0 -> 1 ó 1 > 0 detectado en la llamada da lugar al envío de un aviso.
- En el caso de "ALARM (Página 6930)", "ALARM_8 (Página 6923)" y "ALARM_8P (Página 6925)", con el ajuste predeterminado del procedimiento de aviso (aviso disparado por acuse, inactivo), todo cambio de señal detectado en la llamada da lugar también a un aviso.
  Sin embargo, si el aviso disparado por acuse está activado, no todos los cambios de señal detectados dan lugar a un aviso (más detalles abajo).
- Después de la ejecución del bloque, los valores asociados (entradas SD_i) están captados por completo y asignados al aviso (ver "Parámetros de envío y recepción" en Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669).)
  En lo referente a la coherencia respecto a clases de prioridad de mayor prioridad, se aplica a los valores asociados lo siguiente: Todo valor asociado SD_i es coherente en sí mismo.
- Con los parámetros de estado DONE, ERROR y STATUS se vigila el estado de procesamiento del bloque (ver "Parámetros de estado" en Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)).

Nota

Los parámetros ID y EV_ID solo se evalúan en la primera llamada del bloque (los parámetros actuales o los valores predefinidos de la instancia).

Registros de visualizadores

Para que las instrucciones para generar avisos de PLC envíen un aviso al detectar un cambio de señal, es necesario que haya por lo menos un visualizador registrado para avisos de PLC. Si no hay ninguno registrado, el parámetro STATUS devuelve el valor "1".

Detección de cambio de señal

Por cada instancia de una instrucción de aviso, está disponible una memoria de avisos con dos 2 espacios de memoria.
Inicialmente, esta memoria de avisos está vacía. En cuanto la instrucción detecta un cambio de señal en la entrada SIG o en una de las entradas SIG_1, ... SIG_8, se registra el cambio en el primer espacio de memoria. Este espacio está ocupado hasta que se envía el aviso correspondiente.
El siguiente cambio de señal detectado en la entrada SIG o en una de las entradas SIG_1, ... SIG_8 se registra en el segundo espacio de memoria. Si el primer espacio de memoria todavía está ocupado y se producen nuevos cambios de señal, se sobrescribe siempre el segundo espacio de memoria.
Esta pérdida de avisos se notifica mediante los parámetros de salida ERROR y STATUS (ERROR=0, STATUS=11). Además, los dispositivos de visualización registrados reciben una notificación al respecto en cuanto puede enviarse el siguiente aviso.
Si el primer espacio de memoria queda libre, el segundo espacio de memoria se transfiere al primero. De este modo, el segundo espacio de memoria vuelve a quedar libre.

Notificación controlada por acuse

Para reducir el volumen de avisos en la instalación, pueden usarse las instrucciones "ALARM (Página 6930)", "ALARM_8 (Página 6923)" y "ALARM_8P (Página 6925)", que soportan el procedimiento de notificación controlada por acuse.

En este procedimiento, después de generarse un aviso entrante (cambio de señal de 0 a 1) para una señal, no se vuelven a generar avisos para la misma señal hasta que el usuario haya acusado el primero en un visualizador. El siguiente aviso que se muestra en el visualizador después del acuse es un aviso saliente (cambio de señal de 1 a 0). A continuación se inicia de nuevo el ciclo de aviso con un aviso entrante (cambio de señal de 0 a 1), que debe acusarse. De este modo se pueden controlar a través del visualizador los avisos provocados por cambio de señal (excepto el aviso saliente).

El procedimiento de notificación (notificación controlada por acuse activa o inactiva) para las instrucciones "ALARM (Página 6930)", "ALARM_8 (Página 6923)" y "ALARM_8P (Página 6925)" se define para toda la CPU en la configuración. De modo predeterminado, la notificación controlada por acuse está desactivada.

Para que la evaluación de avisos sea uniforme dentro de una instalación, en caso de utilizarse el procedimiento de notificación controlada por acuse, debe comprobarse que todos los visualizadores soporten este procedimiento.

Nota

(Sobre el uso de visualizadores que no soportan la notificación controlada por acuse.)

Si se ha activado en la CPU la notificación controlada por acuse, la CPU envía los avisos solo a aquellos visualizadores que soportan este procedimiento. Por lo tanto, si no hay ningún visualizador que soporte la notificación controlada por acuse, la CPU no envía ningún aviso. Esto se notifica una sola vez con ERROR=1 y STATUS=1.

Acuse de aviso con las instrucciones "ALARM", "ALARM_8" y "ALARM_8P"

Se utiliza el método de acuse centralizado, es decir: cuando se ha acusado un aviso en un visualizador, esta información de acuse se envía en primer lugar a la CPU que ha generado el aviso. Desde allí se distribuye la información de acuse a todos los visualizadores registrados al efecto.

Siempre se acusa una señal, y no un aviso individual. Si p. ej se han notificado varios flancos ascendentes de una señal y se acusa el evento entrante, todos los eventos entrantes anteriores con el mismo número de aviso se consideran acusados.

Indicador de acuse

Las instrucciones "NOTIFY (Página 6928)" y "NOTIFY_8P (Página 6920)" no poseen indicador de acuse. Con "ALARM (Página 6930)" se puede conocer el estado de acuse a través de los parámetros de salida ACK_UP y ACK_DN, con "ALARM_8P (Página 6925)" y "ALARM_8 (Página 6923)", a través del parámetro de salida ACK_STATE. Estas salidas se actualizan en el momento de la llamada del bloque, en caso de que el parámetro de control EN_R tenga el valor "1".

Bloquear y habilitar avisos mediante instrucción o dispositivo de visualización

Puede ser útil suprimir avisos, p. ej. cuando se realizan modificaciones estructurales de la instalación. Por ello existe la posibilidad de bloquear y volver a habilitar avisos desde el dispositivo de visualización o desde el programa. El bloqueo/habilitación es válido para todos los dispositivos que se han registrado para el aviso en cuestión. Permanece vigente hasta que se vuelve a habilitar el aviso correspondiente.

Si se han bloqueado avisos, esto se notifica a través de los parámetros de salida ERROR y STATUS (ERROR = 1, STATUS = 21).

Espacio requerido en memoria de trabajo por las instrucciones para generar avisos de bloque

Las instrucciones para generar avisos de PLC requieren, para funcionar sin problemas, un búfer de datos de comunicación dependiente de los valores asociados en la memoria de trabajo de la CPU (área de código). El espacio ocupado en la memoria se indica en la tabla siguiente.

Instrucción	Espacio en bytes requerido en la memoria de trabajo de la CPU
NOTIFY	200 + 2 * longitud de los valores asociados indicados en la primera llamada en SD_1,...SD_10
NOTIFY_8P	200 + 2 * longitud de los valores asociados indicados en la primera llamada en SD_1,...SD_10
ALARM	200 + 2 * longitud de los valores asociados indicados en la primera llamada en SD_1,...SD_10
ALARM_8	100
ALARM_8P	200 + 2 * longitud de los valores asociados indicados en la primera llamada en SD_1,...SD_10
AR_SEND	54

Número de datos transmisibles

El número de datos transmisibles a través de los valores asociados SD_i de las instrucciones "NOTIFY (Página 6928)", "NOTIFY_8P (Página 6920)", "ALARM (Página 6930)" y "ALARM_8P (Página 6925)" no debe superar una determinada longitud máxima. Esta longitud de datos máxima se calcula de la siguiente manera:

longmax =

mín (pdu_local, pdu_remota) - dif - 4 * número de parámetros SD_i utilizados

Donde:

- mín (pdu_local, pdu_remota) es el valor mínimo de los números pdu_local y pdu_remota
- pdu_local es la longitud máxima de los bloques de datos de la CPU propia (ver datos técnicos de la CPU utilizada)
- pdu_remota es la longitud máxima de los bloques de datos de los visualizadores
- dif = 48 si está activa la notificación controlada por acuse; 44 si está inactiva

Ejemplo

Supongamos que una CPU 414-2 envía avisos a HMI a través de Industrial Ethernet. La notificación controlada por acuse está inactiva.

Se utilizan los valores asociados SD_1, SD_2 y SD_3.

pdu_local = 480 bytes, pdu_remota = 480 bytes,

Número de parámetros SD_i utilizados: 3

Se aplica lo siguiente:

longmax = min (480, 480) - 44 - 4 * 3 = 480 - 44 - 12 = 424

Por lo tanto, la longitud de datos máxima transmisible por instrucción es de 424 bytes.

NOTIFY_8P: Notificar hasta ocho cambios de señal

Descripción

La instrucción "NOTIFY_8P" es la ampliación de "NOTIFY (Página 6928)" a ocho señales.

Se genera un aviso cuando se ha detectado un cambio de señal en por lo menos una señal. En la primera llamada de "NOTIFY_8P" se genera siempre un aviso. Para las ocho señales existe un número de aviso común, que se divide en ocho avisos parciales en el visualizador.

Por cada instancia de "NOTIFY_8P" está disponible una memoria de avisos con 2 espacios de memoria. Para más detalles sobre el almacenamiento intermedio de cambios de señal, ver la sección Introducción a la creación de avisos de PLC con instrucciones (Página 6917).

Nota

A pesar de la pérdida de aviso, se notifican al visualizador los dos últimos cambios de cada señal.

ADVERTENCIA

Llamada de "NOTIFY_8P"

Antes de llamar "NOTIFY_8P" en un sistema de automatización, debe comprobarse que todos los visualizadores soporten ese bloque. Si no se tiene en cuenta esto, se interrumpirá la comunicación entre el sistema de automatización y los visualizadores conectados. En consecuencia, no se podrá acceder a la instalación con los visualizadores conectados.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "NOTIFY_8P":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
SIG_i, 1 ≤ i ≤ 8	Input	BOOL	I, Q, M, D, L	Señal con el número ordinal i que se debe vigilar
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE ID solo se evalúa en la primera llamada.
EV_ID	Input	C_NOTIFY_8P	I, Q, M, D, L	Número de aviso (no permitido: 0) EV_ID solo se evalúa en la primera llamada. Tras ello, en cada llamada de "NOTIFY_8P" con el DB de instancia correspondiente, se utiliza el mismo número de aviso que en la primera llamada. El número de aviso se asigna automáticamente. Con ello se garantiza la coherencia de los números de aviso. El número de aviso debe ser unívoco dentro del programa de usuario.
SEVERITY	Input	WORD	I, Q, M, D, L o constante	Peso del evento Valores posibles: De 0 a 127 (el valor 0 representa el peso máximo); valor predeterminado: 64 Este parámetro no es relevante para el procesamiento del aviso.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: Ha finalizado la generación del aviso.
ERROR	Output	BOOL	I, Q, M, D, L	El parámetro de estado ERROR: ERROR=TRUE significa que se ha producido un error durante el procesamiento. Para más detalles, véase el parámetro STATUS.

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Ver una información de error
SD_i, 1 ≤ i ≤ 10	InOut	ANY	I, Q, M, D, T, C	Valor asociado con número ordinal i Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME **Nota:** Si el puntero ANY accede a un DB, este siempre debe especificarse. (p. ej.: P# DB10.DBX5.0 byte 10)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de los errores específicos de "NOTIFY_8P", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Pérdida de aviso: El cambio de señal precedente o el aviso precedente no se ha podido enviar y se sustituye por el aviso actual.
0	22	• Error en el puntero a los valores asociados SD_i: – Referente a la longitud de datos o al tipo de datos – No se puede acceder a los valores asociados en la memoria del usuario, p. ej. debido a que se ha borrado un DB o a un error de longitud de área – El aviso activado se envía sin valores asociados o solo con el número de valores asociados posible en el momento. • El parámetro actual seleccionado de SEVERITY se encuentra por encima del rango admisible. El aviso activado se envía con SEVERITY=127.
0	25	La comunicación se ha iniciado. El aviso se está procesando.
1	1	Problemas de comunicación: La conexión se ha interrumpido o no hay registro
1	4	En la primera llamada • la EV_ID indicada se encuentra fuera del rango admisible • existe un error formal del puntero ANY SD_i • se ha rebasado el área de memoria máxima transmisible para la CPU por cada "NOTIFY_8P"
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "NOTIFY_8P" • No se ha indicado ningún DB de instancia, sino un DB global
1	18	EV_ID ya ha sido utilizado por una de las instrucciones "NOTIFY (Página 6928)", "NOTIFY_8P", "ALARM (Página 6930)", "ALARM_8 (Página 6923)" o "ALARM_8P (Página 6925)".
1	20	La memoria de trabajo disponible no es suficiente.
1	21	El aviso con la EV_ID indicada está bloqueado.

Consulte también

Crear avisos de programa (Página 3178)

ALARM_8: Generar avisos PLC sin valores asociados para ocho señales

Descripción

A excepción de la ausencia de los valores asociados SD_1, ..., SD_10, esta instrucción es idéntica a NOTIFY_8P (Página 6920).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_8":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	Parámetro de control enabled to receive, con el cual, al llamar el bloque, la salida ACK_STATE se actualiza (EN_R=1) o no se actualiza (EN_R=0).
SIG_i, 1≤i ≤8	Input	BOOL	I, Q, M, D, L	Señal con el número ordinal i que se debe vigilar
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE ID solo se evalúa en la primera llamada.
EV_ID	Input	C_ALARM_8	I, Q, M, D, L	Número de aviso (no permitido: 0) EV_ID solo se evalúa en la primera llamada. Tras ello, en cada llamada de "ALARM_8" con el DB de instancia correspondiente, se utiliza el mismo número de aviso que en la primera llamada. El número de aviso se asigna automáticamente. Con ello se garantiza la coherencia de los números de aviso. El número de aviso debe ser unívoco dentro del programa de usuario.
SEVERITY	Input	WORD	I, Q, M, D, L o constante	Peso del evento Valores posibles: De 0 a 127 (el valor 0 significa el peso máximo) Este parámetro no es relevante para el procesamiento del aviso.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: Ha finalizado la generación del aviso.
ERROR	Output	BOOL	I, Q, M, D, L	El parámetro de estado ERROR: ERROR=TRUE significa que se ha producido un error durante el procesamiento. Para más detalles, véase el parámetro STATUS.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Ver una información de error
ACK_STATE	Output	WORD	I, Q, M, D, L	Mapa de bits con el estado de acuse actual de los ocho avisos (1: Evento acusado, 0: Evento no acusado): • Los bits de 0 a 7 se reproducen en el evento entrante en SIG_1 a SIG_8 • Los bits de 8 a 15 se reproducen en el evento saliente en SIG_1 a SIG_8 • Estado de inicialización: W#16#FFFF, es decir, se han acusado todos los eventos entrantes y salientes

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "ALARM_8" que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Pérdida de aviso: El cambio de señal precedente o el aviso precedente no se ha podido enviar y se sustituye por el aviso actual.
0	22	El parámetro actual seleccionado de SEVERITY se encuentra por encima del rango admisible. El aviso activado se envía con SEVERITY=127.
0	25	La comunicación se ha iniciado. El aviso se está procesando.
1	1	Problemas de comunicación: La conexión se ha interrumpido o no hay registro Si está activada la notificación controlada por acuse: Indicación temporal, si no existen visualizadores que soporten la notificación controlada por acuse
1	4	En la primera llamada, la EV_ID indicada se encuentra fuera del rango admisible.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "ALARM_8" • No se ha indicado ningún DB de instancia, sino un DB global
1	18	EV_ID ya ha sido utilizado por una de las instrucciones "NOTIFY (Página 6928)", "NOTIFY_8P (Página 6920)", "ALARM (Página 6930)", "ALARM_8" o "ALARM_8P (Página 6925)".
1	20	La memoria de trabajo disponible no es suficiente.
1	21	El aviso con la EV_ID indicada está bloqueado.

Nota

Después de la primera llamada, todos los bits de la salida ACK_STATE están activados, y los valores anteriores de las entradas SIG_i, 1 ≤ i ≤ 8 se ponen a 0.

Consulte también

Crear avisos de programa (Página 3178)

ALARM_8P: Generar avisos PLC con valores asociados para ocho señales

Descripción

Esta instrucción es la ampliación de "ALARM (Página 6930)" a ocho señales.

Si no está activada la notificación controlada por acuse, se genera siempre un aviso cuando se detecta un cambio de señal en por lo menos una señal (excepción: En la primera llamada se genera siempre un aviso). Para las ocho señales existe un número de aviso común, que se divide en ocho avisos parciales en el visualizador. Puede acusarse por separado cada uno de los avisos parciales, o bien varios avisos parciales al mismo tiempo.

Mediante el parámetro de salida ACK_STATE puede seguir procesándose el estado de acuse de los distintos avisos en el programa. Si se bloquea o habilita un aviso de una instrucción "ALARM_8P", esto afecta siempre a toda la instrucción "ALARM_8P". No es posible bloquear o habilitar señales individualmente.

Por cada instancia de "ALARM_8P" está disponible una memoria de avisos con dos espacios de memoria. Para más detalles sobre el almacenamiento intermedio de cambios de señal, ver la sección Introducción a la creación de avisos de PLC con instrucciones (Página 6917).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_8P":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	Parámetro de control enabled to receive, con el cual, al llamar el bloque, la salida ACK_STATE se actualiza (EN_R=1) o no se actualiza (EN_R=0).
SIG_i, 1≤i ≤8	Input	BOOL	I, Q, M, D, L	Señal con el número ordinal i que se debe vigilar
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE ID solo se evalúa en la primera llamada.
EV_ID	Input	C_ALARM_8P	I, Q, M, D, L	Número de aviso (no permitido: 0); EV_ID solo se evalúa en la primera llamada. Tras ello, en cada llamada de "ALARM_8P" con el DB de instancia correspondiente, se utiliza el mismo número de aviso que en la primera llamada. El número de aviso se asigna automáticamente. Con ello se garantiza la coherencia de los números de aviso. El número de aviso debe ser unívoco dentro del programa de usuario.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SEVERITY	Input	WORD	I, Q, M, D, L o constante	Peso del evento Valores posibles: De 0 a 127 (el valor 0 significa el peso máximo) Este parámetro no es relevante para el procesamiento del aviso.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: Ha finalizado la generación del aviso.
ERROR	Output	BOOL	I, Q, M, D, L	El parámetro de estado ERROR ERROR=TRUE significa que se ha producido un error durante el procesamiento. Para más detalles, véase el parámetro STATUS.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Ver una información de error
ACK_STATE	Output	WORD	I, Q, M, D, L	Mapa de bits con el estado de acuse actual de los ocho avisos (1: Evento acusado, 0: Evento no acusado): • Los bits de 0 a 7 se reproducen en el evento entrante en SIG_1 a SIG_7 • Los bits de 8 a 15 se reproducen en el evento saliente en SIG_1 a SIG_7 • Estado de inicialización: W#16#FFFF, es decir, se han acusado todos los eventos entrantes y salientes
SD_j, 1≤ j ≤10	InOut	ANY	I, Q, M, D, T, C	Valor asociado con número ordinal j Los valores asociados son válidos para todos los avisos. Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (por ej.: P# DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "ALARM_8P" que puede indicarse a través de los parámetros ERRORy STATUS.

ERROR	STATUS (decimal)	Explicación
0	11	Pérdida de aviso: El cambio de señal precedente o el aviso precedente no se ha podido enviar y se sustituye por el aviso actual.
0	22	• Error en el puntero a los valores asociados SD_i: – Referente a la longitud de datos o al tipo de datos – No se puede acceder a los valores asociados en la memoria del usuario, p. ej. debido a que se ha borrado un DB o a un error de longitud de área – El aviso activado se envía sin valores asociados. • El parámetro actual seleccionado de SEVERITY se encuentra por encima del rango admisible. El aviso activado se envía con SEVERITY=127 .
0	25	La comunicación se ha iniciado. El aviso se está procesando.
1	1	Problemas de comunicación: La conexión se ha interrumpido o no hay registro. Si está activada la notificación controlada por acuse: Indicación temporal, si no existen visualizadores que soporten la notificación controlada por acuse.
1	4	En la primera llamada: • La EV_ID indicada se encuentra fuera del rango admisible. • Existe un error formal del puntero ANY SD_i . • Se ha rebasado el área de memoria máxima transmisible para la CPU por cada "ALARM_8P".
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	Al llamar la instrucción • se ha indicado un DB de instancia que no pertenece a "ALARM_8P". • no se ha indicado ningún DB de instancia, sino un DB global
1	18	EV_ID ya ha sido utilizado por una de las instrucciones "NOTIFY (Página 6928)", "NOTIFY_8P (Página 6920)", "ALARM (Página 6930)", "ALARM_8 (Página 6923)" o "ALARM_8P".
1	20	La memoria de trabajo disponible no es suficiente.
1	21	El aviso con la EV_ID indicada está bloqueado.

Nota

Después de la primera llamada, todos los bits de la salida ACK_STATEestán activados, y se adoptan los valores anteriores "0" de las entradas SIG_i, 1< i < 8.

Consulte también

Crear avisos de programa (Página 3178)

NOTIFY: Notificar un cambio de señal

Descripción

La instrucción vigila una señal. Tanto con flanco ascendente (evento entrante) como con flanco descendente (aviso saliente), genera un aviso al cual pueden adjuntarse hasta diez valores asociados. El aviso se envía a todos los dispositivos registrados al efecto. En la primera llamada se envía un aviso con el estado de señal actual.

Los valores asociados se captan en el momento de la evaluación de flancos y se asignan al aviso.

Por cada instancia de la instrucción está disponible una memoria de avisos con 2 espacios de memoria. Para más detalles sobre el almacenamiento intermedio de cambios de señal, ver Introducción a la creación de avisos de PLC con instrucciones (Página 6917).

La instrucción "NOTIFY" se ajusta a la norma IEC 1131-5.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "NOTIFY":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
SIG	Input	BOOL	I, Q, M, D, L	La señal que debe vigilarse
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE ID solo se evalúa en la primera llamada.
EV_ID	Input	C_NOTIFY	I, Q, M, D, L	Número de aviso (no permitido: 0) EV_ID solo se evalúa en la primera llamada. Tras ello, en cada llamada de "NOTIFY" con el DB de instancia correspondiente, se utiliza el mismo número de aviso que en la primera llamada. El número de aviso se asigna automáticamente. Con ello se garantiza la coherencia de los números de aviso. El número de aviso debe ser unívoco dentro del programa de usuario.
SEVERITY	Input	WORD	I, Q, M, D, L o constante	Peso del evento Valores posibles: De 0 a 127 (el valor 0 significa el peso máximo) Este parámetro no es relevante para el procesamiento del aviso.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: Ha finalizado la generación del aviso.
ERROR	Output	BOOL	I, Q, M, D, L	El parámetro de estado ERROR ERROR=TRUE significa que se ha producido un error durante el procesamiento. Para más detalles, véase el parámetro STATUS.

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Ver una información de error
SD_i, 1≤i ≤10	InOut	ANY	I, Q, M, D, T, C	Valor asociado con número ordinal i Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Nota:** Si el puntero ANY accede a un DB, este siempre debe especificarse. (p. ej.: P# DB10.DBX5.0 byte 10)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La tabla siguiente contiene toda la información de error específica que puede devolverse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Pérdida de aviso: El cambio de señal precedente o el aviso precedente no se ha podido enviar y se sustituye por el aviso actual.
0	22	• Error en el puntero a los valores asociados SD_i: – Referente a la longitud de datos o al tipo de datos – No se puede acceder a los valores asociados en la memoria del usuario, p. ej. debido a que se ha borrado un DB o a un error de longitud de área – El aviso activado se envía sin valores asociados o solo con el número de valores asociados posible en el momento. • El parámetro actual seleccionado de SEVERITY se encuentra por encima del rango admisible. El aviso activado se envía con SEVERITY=127 .
0	25	La comunicación se ha iniciado. El aviso se está procesando.
1	1	Problemas de comunicación: La conexión se ha interrumpido o no hay registro
1	4	En la primera llamada • la EV_ID indicada se encuentra fuera del rango admisible • existe un error formal del puntero ANY SD_i • se ha rebasado el área de memoria máxima transmisible para la CPU por cada "NOTIFY"
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	Al llamar la instrucción • se ha indicado un DB de instancia que no pertenece a "NOTIFY". • no se ha indicado ningún DB de instancia, sino un DB global
1	18	EV_ID ya ha sido utilizado por una de las instrucciones "NOTIFY", "NOTIFY_8P (Página 6920)", "ALARM (Página 6930)", "ALARM_8 (Página 6923)" o "ALARM_8P (Página 6925)".
1	20	La memoria de trabajo disponible no es suficiente.
1	21	El aviso con la EV_ID indicada está bloqueado.

Consulte también

Crear avisos de programa (Página 3178)

ALARM: Generar avisos PLC con indicación de acuse

Descripción

La instrucción vigila una señal.

- Modo estándar (es decir, la notificación controlada por acuse está desactivada): Tanto con flanco ascendente (evento entrante) como con flanco descendente (aviso saliente), el bloque genera un aviso al cual pueden adjuntarse hasta diez valores asociados.
- Notificación controlada por acuse activada: Una vez generado el aviso entrante, el bloque no genera más avisos para la señal hasta que el evento entrante ha sido acusado en un visualizador.
  Consulte también: Introducción a la creación de avisos de PLC con instrucciones (Página 6917).

El aviso se envía a todos los dispositivos registrados al efecto.

En la primera llamada se envía un aviso con el estado de señal actual.

La salida ACK_UP se anula cuando existe un flanco ascendente. Se activa cuando llega el acuse del evento entrante desde un visualizador registrado.

De modo análogo, para la salida ACK_DN se aplica lo siguiente: Se anula cuando existe un flanco descendente. Se activa cuando llega el acuse del evento saliente desde un visualizador registrado. Una vez llegado el acuse desde un visualizador registrado, la información de acuse se transfiere a todos los dispositivos registrados al efecto.

Por cada instancia de la instrucción "ALARM" está disponible una memoria de avisos con dos espacios de memoria. Para más detalles sobre el almacenamiento intermedio de cambios de señal, ver la sección "Detección de cambio de señal" en Introducción a la creación de avisos de PLC con instrucciones (Página 6917).

La instrucción "ALARM" se ajusta a la norma IEC 1131-5.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	Parámetro de control enabled to receive con el cual, al llamar el bloque, las salidas ACK_UP y ACK_DN se actualizan (EN_R=1) o no se actualizan (EN_R=0). Con EN_R=0, los parámetros de salida ACK_UP y ACK_DN permanecen invariables.
SIG	Input	BOOL	I, Q, M, D, L	La señal que debe vigilarse
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE ID solo se evalúa en la primera llamada.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EV_ID	Input	C_ALARM	I, Q, M, D, L	Número de aviso (no permitido: 0) EV_ID solo se evalúa en la primera llamada. Tras ello, en cada llamada de "ALARM" con el DB de instancia correspondiente, se utiliza el mismo número de aviso que en la primera llamada. El número de aviso se asigna automáticamente. Con ello se garantiza la coherencia de los números de aviso. El número de aviso debe ser unívoco dentro del programa de usuario.
SEVERITY	Input	WORD	I, Q, M, D, L o constante	Peso del evento Valores posibles: De 0 a 127 (el valor 0 significa el peso máximo) Este parámetro no es relevante para el procesamiento del aviso.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: Ha finalizado la generación del aviso.
ERROR	Output	BOOL	I, Q, M, D, L	El parámetro de estado ERROR ERROR=TRUE significa que se ha producido un error durante el procesamiento. Para más detalles, véase el parámetro STATUS.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Ver una información de error
ACK_DN	Output	BOOL	I, Q, M, D, L	El evento saliente ha sido acusado en un visualizador. Estado de inicialización: 1
ACK_UP	Output	BOOL	I, Q, M, D, L	El evento entrante ha sido acusado en un visualizador. Estado de inicialización: 1
SD_i, 1≤i ≤10	InOut	ANY	I, Q, M, D, T, C	Valor asociado con número ordinal i Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Nota:** Si el puntero ANY accede a un DB, este siempre debe especificarse. (p. ej.: P# DB10.DBX5.0 byte 10)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "ALARM" que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Pérdida de aviso: El cambio de señal precedente o el aviso precedente no se ha podido enviar y se sustituye por el aviso actual.
0	22	• Error en el puntero a los valores asociados SD_i: – Referente a la longitud de datos o al tipo de datos – No se puede acceder a los valores asociados en la memoria del usuario, p. ej. debido a que se ha borrado un DB o a un error de longitud de área – El aviso activado se envía sin valores asociados. • El parámetro actual seleccionado de SEVERITY se encuentra por encima del rango admisible. El aviso activado se envía con SEVERITY=127.
0	25	La comunicación se ha iniciado. El aviso se está procesando.
1	1	Problemas de comunicación: La conexión se ha interrumpido o no hay registro Si está activada la notificación controlada por acuse: Indicación temporal, si no existen visualizadores que soporten la notificación controlada por acuse
1	4	En la primera llamada • la EV_ID indicada se encuentra fuera del rango admisible • existe un error formal del puntero ANY SD_i • Se ha rebasado el área de memoria máxima transmisible para la CPU por cada ALARM
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a ALARM • No se ha indicado ningún DB de instancia, sino un DB global
1	18	EV_ID ya ha sido utilizado por una de las instrucciones "NOTIFY (Página 6928)", "NOTIFY_8P" (Página 6920)", "ALARM", "ALARM_8 (Página 6923)" o "ALARM_8P (Página 6925)".
1	20	La memoria de trabajo disponible no es suficiente.
1	21	El aviso con la EV_ID indicada está bloqueado.

Nota

Después de la primera llamada, las salidas ACK_UP y ACK_DN tienen el valor "1", y para la entrada SIG se adopta el valor anterior "0".

Consulte también

Crear avisos de programa (Página 3178)

Instrucciones para generar avisos de PLC sin DB de instancia

Comportamiento en arranque de las instrucciones para generar avisos de PLC

Comportamiento en rearranque completo (en caliente)

En el rearranque completo (en caliente), los DBs de instancia de las instrucciones para generar avisos de PLC se ponen en estado no inicializado. Los parámetros actuales guardados en los DBs de instancia permanecen invariables. Con la siguiente llamada de bloque, los parámetros ID y EV_ID se evalúan de nuevo.

Comportamiento en arranque en frío

En el arranque en frío, los contenidos de los DBs de instancia de las instrucciones para generar avisos de PLC se ajustan a los valores iniciales.

Comportamiento en rearranque

En el rearranque, las instrucciones para generar avisos de PLC se comportan como bloques de función del usuario con posibilidad de continuación. Continúan ejecutándose a partir del punto de interrupción.

Comportamiento tras un borrado total

El borrado total siempre implica la interrupción de todas las conexiones, de modo que ya no hay ningún dispositivo registrado para avisos. Se borra el programa de usuario. Si se ha insertado una tarjeta FLASH, las partes del programa relevantes para la ejecución se cargan de nuevo en la CPU desde la tarjeta, y la CPU efectúa un rearranque completo (en caliente) o un arranque en frío (implícitamente siempre se trata de un arranque en frío, ya que tras el borrado total todos los datos del usuario quedan inicializados).

Comportamiento en fallo de las instrucciones para generar avisos de PLC

Interrupción de la conexión

Las conexiones asignadas a las instancias se vigilan para detectar su posible interrupción. En caso de interrupción de la conexión, el dispositivo afectado se borra de la lista interna de la CPU de los dispositivos registrados para avisos de PLC. Se borran los avisos todavía pendientes para ese dispositivo.

Si después de una interrupción de conexión existen todavía otros dispositivos registrados, éstos siguen recibiendo avisos. Las instrucciones no dejarán de enviar avisos hasta que estén interrumpidas las conexiones con todos los dispositivos registrados. Tras ello se realiza una notificación a través de los parámetros de salida ERROR y STATUS (ERROR = 1, STATUS = 1).

Interfaz de error con el programa de usuario

Si al procesarse una instrucción para generar avisos de PLC se produce un error, el parámetro de salida ERROR se ajusta a "1", y se registra en el parámetro de salida STATUS la correspondiente identificación de error. Esta información de error se puede evaluar en el programa.

Ejemplos de posibles errores:

- La transmisión no es posible debido a falta de recursos
- Error al acceder a una de las señales que se deben vigilar

Introducción a la creación de avisos de PLC con instrucciones

Instrucciones para generar avisos de PLC

Pueden generarse avisos de PLC con las siguientes instrucciones:

- "ALARM_SQ (Página 6939)"
- "ALARM_S (Página 6936)"
- "ALARM_DQ (Página 6943)"
- "ALARM_D (Página 6941)"

Estas instrucciones tienen las siguientes características:

- Los avisos de PLC enviados de "ALARM_SQ (Página 6939)" y "ALARM_DQ (Página 6943)" con el estado de señal "1" pueden acusarse desde un visualizador registrado. Los avisos de PLC de "ALARM_S" (Página 6936) y "ALARM_D (Página 6941)" siempre se acusan de modo implícito.
- Los avisos de PLC no se generan al detectarse un cambio de flanco, sino al producirse cualquier llamada.
- Después de la ejecución del bloque, el valor asociado SD está captado por completo y asignado al aviso de PLC.
  En lo referente a la coherencia respecto a clases de prioridad de mayor prioridad, se aplica al valor asociado lo siguiente: Son coherentes
  - los tipos de datos simples (bit, byte, palabra y palabra doble)
  - una matriz del tipo de datos Byte hasta una longitud máxima específica de CPU

ALARM_SC

Con ayuda de la instrucción "ALARM_SC (Página 6945)" es posible determinar

- el estado de acuse del último "aviso entrante" y el estado de señal en la última llamada de "ALARM_SQ (Página 6939)" / "ALARM_DQ (Página 6943)", o bien
- el estado de señal en la última llamada de "ALARM_S (Página 6936)" / "ALARM_D (Página 6941)".

Registro de visualizadores

Para que las instrucciones para generar avisos de PLC envíen un aviso de PLC al producirse una llamada, es necesario que haya por lo menos un visualizador registrado para avisos de PLC.

Respaldo de avisos

Para evitar en lo posible que se pierdan avisos incluso en caso de sobrecarga del sistema de comunicación, las instrucciones "ALARM_SQ (Página 6939)", "ALARM_S (Página 6936)", "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)" pueden almacenar en memoria intermedia dos avisos cada una.

Si, a pesar de ello, se pierden avisos de PLC, esto se notifica mediante RET_VAL. Con el siguiente aviso de PLC que pueda enviarse, los visualizadores registrados recibirán una notificación al respecto.

Acuse de aviso con "ALARM_SQ" y "ALARM_DQ"

Cuando se acusa en un visualizador un "aviso entrante", esta información de acuse se envía en primer lugar a la CPU que ha provocado el aviso. Desde allí se distribuye la información de acuse a todos los visualizadores registrados al efecto.

Bloquear y habilitar avisos

Los avisos de PLC generados con "ALARM_SQ (Página 6939)", "ALARM_S (Página 6936)", "ALARM_DQ (Página 6943)" o "ALARM_D (Página 6941)", no se pueden bloquear y a continuación volver a habilitar.

Modificaciones de un programa que contiene llamadas a "ALARM_SQ" / "ALARM_S"

Nota

Al cargar un bloque ya existente en la CPU con llamadas a "ALARM_SQ (Página 6939)" / "ALARM_S (Página 6936)", es posible que el bloque anterior haya enviado un aviso de PLC entrante, pero el nuevo bloque no envíe el correspondiente aviso de PLC saliente. En este caso, el aviso permanece en la memoria interna de avisos de la CPU. Esta situación puede producirse también cuando se borran bloques con llamadas de "ALARM_SQ (Página 6939)" / "ALARM_S (Página 6936)".

Para borrar de la memoria interna de avisos este tipo de avisos de PLC, debe ponerse la CPU en el estado operativo STOP y a continuación efectuar un rearranque completo (en caliente) o un arranque en frío.

Modificaciones de un programa que contiene llamadas a "ALARM_DQ" / "ALARM_D"

Aunque el programa contenga llamadas a "ALARM_DQ (Página 6943)" y/o "ALARM_D (Página 6941)", las mencionadas modificaciones del programa pueden provocar que queden avisos de PLC en la memoria interna de avisos, ocupando de modo permanente recursos del sistema.

A diferencia de los recursos del sistema asignados mediante llamadas de "ALARM_SQ (Página 6939)" / "ALARM_S (Página 6936)", los recursos del sistema asignados mediante llamadas de "ALARM_DQ (Página 6943)" / "ALARM_D (Página 6941)" pueden liberarse sin necesidad de poner la CPU en el estado operativo STOP. Para esto debe usarse la instrucción "DEL_SI (Página 6959)". Antes de liberar recursos del sistema asignados de modo dinámico "DEL_SI", puede ser útil leer con la instrucción "READ_SI" la información acerca de este tipo de recursos del sistema en la CPU en el momento actual (ver instrucción "READ_SI (Página 6956)").

Número de datos transmisibles

El número de datos transmisibles a través del valor asociado SD de las instrucciones "ALARM_S (Página 6936)", "ALARM_SQ (Página 6939)", "ALARM_D (Página 6941)" y "ALARM_DQ (Página 6943)" no debe superar una determinada longitud máxima. Esta longitud de datos máxima se calcula de la siguiente manera:

longmax = min (pdu_local, pdu_remota) - 48

Donde:

- pdu_local es la longitud máxima de los bloques de datos de la CPU propia (SZL_ID W#16#0131, INDEX 1, variable pdu)
- pdu_remota es la longitud máxima de los bloques de datos de los visualizadores

Ejemplo

Supongamos que una CPU 414-2 envía avisos a una PG 760 (a través de MPI).

pdu_local = 480 bytes, pdu_remota = 480 bytes,

Se aplica lo siguiente:

longmax = min (480, 480) - 48 = 480 - 48 = 432

Por lo tanto, la longitud de datos máxima transmisible por instrucción es de 432 bytes.

ALARM_S: Generar aviso de alarma

Descripción

La instrucción genera para cada llamada un aviso al que puede adjuntarse un valor asociado. El aviso se envía a todos los dispositivos registrados al efecto. De este modo, la instrucción pone a disposición del usuario un sencillo mecanismo de notificación. Debe tenerse en cuenta que la instrucción solo se llama cuando el valor de la señal SIG que dispara el aviso está invertido con respecto a la última llamada. De lo contrario, se notifica a través de RET_VAL, y no se envía ningún aviso. En la primera llamada de la instrucción, hay que asegurarse de

que en la entrada SIG esté ajustado el valor "1". De lo contrario, se recibe a través de RET_VAL una información de error y no se envía ningún aviso.

Nota

En los programas nuevos que se vayan a crear, se recomienda usar siempre la instrucción "ALARM_D (Página 6941)" (si la CPU la soporta), ya que ofrece mejores posibilidades de administración de los recursos del sistema.

Asignación de recursos del sistema

Al generar un aviso, el sistema operativo asigna un recurso del sistema para toda la duración de un ciclo de señal.

El ciclo de señal se prolonga desde la llamada con SIG=1 hasta la siguiente llamada con SIG=0. Si dentro del ciclo de señal se sobrecarga o se borra el bloque que genera el aviso, el correspondiente recurso del sistema queda ocupado hasta el siguiente rearranque completo (en caliente).

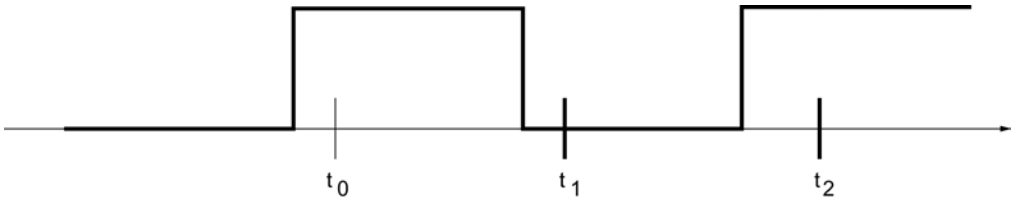
Acuse de avisos

Los avisos que se han enviado con "ALARM_S" siempre están acusados de modo implícito. El estado de señal en la última llamada de "ALARM_S se puede determinar con ayuda de "ALARM_SC (Página 6945)".

Almacenamiento intermedio de estados de señal

La instrucción "ALARM_S" ocupa recursos del sistema. En ellos se almacenan temporalmente, entre otras cosas, los dos últimos estados de señal, incluido el sello de tiempo y el valor asociado. Si se produce una llamada a "ALARM_S" en un momento en que los estados de señal de las dos últimas llamadas "válidas" no han sido enviados todavía (desbordamiento de señal), se descartan el estado de señal actual y el último estado de señal y se registra en la memoria intermedia una identificación de desbordamiento. En el primer momento en que sea posible, se envía la penúltima señal junto con el identificador de desbordamiento.

Ejemplo:



Supongamos que t_0, t_1 y t_2 son los momentos de llamada de "ALARM_S". Si los estados de señal de t_0 y t_1 todavía no han sido enviados en el momento t_2, se descartan los estados de señal de t_1 y t_2, y en el estado de señal de t_0 se activa el identificador de desbordamiento.

Desbordamiento de instancia

Si el número de llamadas de "ALARM_S" es mayor que el número máximo de recursos del sistema de la CPU, puede producirse una escasez de recursos (desbordamiento de instancia). Esto se notifica mediante información de error en RET_VAL, así como en los visualizadores registrados.

El número máximo de llamadas de "ALARM_S" depende de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIG	Input	BOOL	I, Q, M, D, L	La señal que dispara el aviso
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE
EV_ID	Input	C_ALARM_S	I, Q, M, D, L	Número de aviso (no permitido: "0")
SD	Input	ANY	I, Q, M, D, T, C	Valor asociado Longitud máxima: 12 bytes Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
0001	• El valor asociado supera la longitud máxima admisible. • No es posible acceder a la memoria de usuario local (p. ej. acceso a un DB borrado) El aviso se envía. • El valor asociado señala un valor en el área de datos local. El aviso se envía. (solo S7-400)
0002	Advertencia: Se ha asignado el último espacio de memoria libre para acuse de avisos. (solo S7-400)
8081	La EV_ID indicada se encuentra fuera del rango admisible.
8082	Se pierde el aviso, ya que la CPU no tiene más recursos libres para generar avisos de PLC mediante instrucciones.
8083	Se pierde el aviso, ya que ese cambio de señal ya existe, pero todavía no se ha podido enviar (desbordamiento de señal).
8084	En la llamada actual y anterior a "ALARM_S", la señal que dispara el aviso SIG tiene el mismo valor.
8085	No existe registro para la EV_ID indicada.
8086	Ya se está procesando en una clase de prioridad menor una llamada para la EV_ID indicada.
8087	En la primera llamada de "ALARM_S", la señal que dispara el aviso tenía el valor "0".

Código de error (W#16#...)	Explicación
8088	La EV_ID indicada ya está ocupada por otro recurso del sistema ("ALARM_S", ALARM_DQ (Página 6943) , ALARM_D (Página 6941)).
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Consulte también

Crear avisos de programa (Página 3178)

ALARM_SQ: Generar aviso de alarma con acuse

Descripción

La instrucción genera para cada llamada un aviso al que puede adjuntarse un valor asociado. El aviso se envía a todos los dispositivos registrados al efecto. "De este modo, ALARM_SQ" pone a disposición del usuario un sencillo mecanismo de notificación. Recuerde que "ALARM_SQ" solo debe llamarse cuando el valor de la señal SIG que dispara el aviso está invertido con respecto a la última llamada. De lo contrario, se notifica a través de RET_VAL, y no se envía ningún aviso. En la primera llamada de "ALARM_SQ", debe asegurarse de que en la entrada SIG esté ajustado el valor "1". De lo contrario, se recibe a través de RET_VAL una información de error y no se envía ningún aviso.

Nota

En los programas nuevos que se vayan a crear, se recomienda usar siempre la instrucción "ALARM_DQ (Página 6943)" (si la CPU la soporta), ya que ofrece mejores posibilidades de administración de los recursos del sistema.

Asignación de recursos del sistema

Al generar un aviso con "ALARM_SQ", el sistema operativo asigna un recurso del sistema para la duración de un ciclo de señal.

Con "ALARM_SQ, el ciclo de señal se prolonga desde la llamada con SIG=1 hasta la siguiente llamada con SIG=0. A ese intervalo debe sumarse, en su caso, el tiempo transcurrido hasta el acuse de la señal entrante desde uno de los visualizadores registrados.

Si dentro del ciclo de señal se sobrecarga o se borra el bloque que genera el aviso, el correspondiente recurso del sistema queda ocupado hasta el siguiente rearranque completo (en caliente).

Acuse de avisos

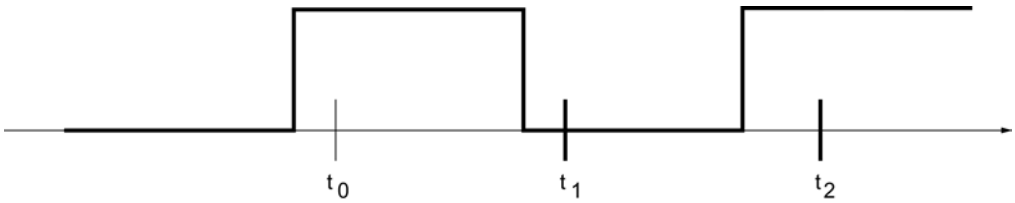
Los avisos enviados por "ALARM_SQ" pueden acusarse con el estado de señal "1" en un visualizador registrado. El estado de acuse del último "aviso entrante" y el estado de señal en

la última llamada de "ALARM_SQ pueden consultarse con ayuda de la instrucción "ALARM_SC (Página 6945)".

Almacenamiento intermedio de estados de señal

"ALARM_SQ" ocupa recursos del sistema. En ellos se almacenan temporalmente, entre otras cosas, los dos últimos estados de señal, incluido el sello de tiempo y el valor asociado. Si se produce una llamada a "ALARM_SQ" en un momento en que los estados de señal de las dos últimas llamadas "válidas" no han sido enviados todavía (desbordamiento de señal), se descartan el estado de señal actual y el último estado de señal y se registra en la memoria intermedia una identificación de desbordamiento. En el primer momento en que sea posible, se envía la penúltima señal junto con el identificador de desbordamiento.

Ejemplo:



Supongamos que t_0, t_1 y t_2 son los momentos de llamada de "ALARM_SQ". Si los estados de señal de t_0 y t_1 todavía no han sido enviados en el momento t_2, se descartan los estados de señal de t_1 y t_2, y en el estado de señal de t_0 se activa el identificador de desbordamiento.

Desbordamiento de instancia

Si el número de llamadas de "ALARM_SQ" es mayor que el número máximo de recursos del sistema de la CPU, puede producirse una escasez de recursos (desbordamiento de instancia). Esto se notifica mediante información de error en RET_VAL, así como en los visualizadores registrados.

El número máximo de llamadas de "ALARM_SQ" depende de la CPU.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_SQ":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIG	Input	BOOL	I, Q, M, D, L	La señal que dispara el aviso
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE
EV_ID	Input	C_ALARM_S	I, Q, M, D, L	Número de aviso (no permitido: 0)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SD	Input	ANY	I, Q, M, D, T, C	Valor asociado Longitud máxima: 12 bytes Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
0001	• El valor asociado supera la longitud máxima admisible. • No es posible acceder a la memoria de usuario local (p. ej. acceso a un DB borrado) El aviso se envía. • El valor asociado señala un valor en el área de datos local. El aviso se envía. (solo S7-400)
0002	Advertencia: Se ha asignado el último espacio de memoria libre para acuse de avisos. (solo S7-400)
8081	La EV_ID indicada se encuentra fuera del rango admisible.
8082	Se pierde el aviso, ya que la CPU no tiene más recursos libres para generar avisos de PLC mediante instrucciones.
8083	Se pierde el aviso, ya que ese cambio de señal ya existe, pero todavía no se ha podido enviar (desbordamiento de señal).
8084	En la llamada actual y anterior a "ALARM_SQ", la señal que dispara el aviso SIGtiene el mismo valor.
8085	No existe registro para la EV_ID indicada.
8086	Ya se está procesando en una clase de prioridad menor una llamada para la EV_ID indicada.
8087	En la primera llamada de "ALARM_SQ", la señal que dispara el aviso tenía el valor "0".
8088	La EV_ID indicada ya está ocupada por otro recurso del sistema (ALARM_S (Página 6936), ALARM_DQ (Página 6943), ALARM_D (Página 6941)).
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Consulte también

Crear avisos de programa (Página 3178)

ALARM_D: Generar siempre avisos PLC acusados

Descripción

La instrucción genera para cada llamada un aviso de PLC al que puede adjuntarse un valor asociado. En esto coincide con la instrucción "ALARM_S (Página 6936)".

Al generar un aviso con "ALARM_D", el sistema operativo asigna un recurso del sistema para la duración de un ciclo de señal.

Con "ALARM_D, el ciclo de señal se prolonga desde la llamada con SIG=1 hasta la siguiente llamada con SIG=0. Si dentro del ciclo de señal se sobrecarga o se borra el bloque que genera el aviso, el correspondiente recurso del sistema queda ocupado hasta el siguiente rearranque completo (en caliente).

La diferencia funcional entre "ALARM_D" y "ALARM_S (Página 6936)" consiste en que el usuario puede administrar esos recursos del sistema ocupados.

- Con ayuda de READ_SI puede leerse información acerca de los recursos del sistema.
- Con DEL_SI pueden liberarse recursos del sistema ocupados. Esto es especialmente útil cuando los recursos del sistema están ocupados de modo permanente. Un recurso del sistema permanece ocupado p. ej. hasta el próximo rearranque completo (en caliente) cuando, al efectuar una modificación del programa, se borra una llamada de FB que contiene llamadas de "ALARM_D". Si, al efectuar una modificación del programa, se carga de nuevo un FB con llamadas de "ALARM_D", puede suceder que "ALARM_D" no genere más avisos de PLC.

La instrucción "ALARM_D" tiene un parámetro más que la instrucción "ALARM_S (Página 6936)": se trata de la entrada CMP_ID. Con ella se asignan los avisos de PLC generados con la instrucción "ALARM_D" a áreas lógicas, p. ej. instalaciones parciales. Si se llama "ALARM_D" en un FB, es conveniente asignar a "CMP_ID" el número del DB de instancia correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_D":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIG	Input	BOOL	I, Q, M, D, L	La señal que dispara el aviso
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE
EV_ID	Input	C_ALARM_S	I, Q, M, D, L	Número de aviso (no permitido: 0)
CMP_ID	Input	DWORD	I, Q, M, D, L o constante	component identifier (no autorizado: 0) Identificador del sistema parcial al que está asignado el aviso en cuestión Valores recomendados: • low word: de 1 a 65535 • high word: 0 Atendiendo a esta recomendación, se evitan posibles conflictos con paquetes de programación creados por SIEMENS.
SD	Input	ANY	I, Q, M, D, T, C	Valor asociado Longitud máxima: 12 bytes Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
0001	• El valor asociado supera la longitud máxima admisible, o • No es posible acceder a la memoria de usuario local (p. ej. acceso a un DB borrado) El aviso se envía. • El valor asociado señala un valor en el área de datos local. El aviso se envía. (solo S7-400)
0002	Advertencia: Se ha asignado el último espacio de memoria libre para acuse de avisos. (solo S7-400)
8081	La EV_ID indicada se encuentra fuera del rango admisible.
8082	Se pierde el aviso, ya que la CPU no tiene más recursos libres para generar avisos de PLC mediante instrucciones.
8083	Se pierde el aviso, ya que ese cambio de señal ya existe, pero todavía no se ha podido enviar (desbordamiento de señal).
8084	En la llamada actual y anterior a "ALARM_D", la señal que dispara el aviso SIGtiene el mismo valor.
8085	No existe registro para la EV_ID indicada.
8086	Ya se está procesando en una clase de prioridad menor una llamada para la EV_ID indicada.
8087	En la primera llamada de "ALARM_D", la señal que dispara el aviso tenía el valor "0".
8088	La EV_ID indicada ya está ocupada por otro recurso del sistema ("ALARM_SQ (Página 6939)", "ALARM_S (Página 6936)", "ALARM_DQ" (Página 6943)).
8089	Se ha asignado a CMP_ID el valor "0".
808A	CMP_ID no concuerda con EV_ID
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Consulte también

Crear avisos de programa (Página 3178)

ALARM_DQ: Generar avisos PLC acusables

Descripción

La instrucción "ALARM_DQ" genera para cada llamada un aviso al que puede adjuntarse un valor asociado. En esto coincide con la instrucción "ALARM_SQ (Página 6939)".

Al generar un aviso con "ALARM_DQ", el sistema operativo asigna un recurso del sistema para toda la duración de un ciclo de señal.

Con "ALARM_DQ, el ciclo de señal se prolonga desde la llamada con SIG=1 hasta la siguiente llamada con SIG=0. A ese intervalo debe sumarse, en su caso, el tiempo transcurrido hasta el acuse de la señal entrante desde uno de los visualizadores registrados.

Si dentro del ciclo de señal se sobrecarga o se borra el bloque que genera el aviso, el correspondiente recurso del sistema queda ocupado hasta el siguiente rearranque completo (en caliente).

La diferencia de funcionalidad de "ALARM_DQ" respecto a "ALARM_SQ (Página 6939)" consiste en que es posible administrar esos recursos del sistema ocupados.

- Con ayuda de READ_SI puede leerse información acerca de los recursos del sistema.
- Con DEL_SI pueden liberarse recursos del sistema ocupados. Esto es especialmente útil cuando los recursos del sistema están ocupados de modo permanente. Un recurso del sistema permanece ocupado p. ej. hasta el siguiente rearranque completo (en caliente) cuando, al efectuar una modificación del programa, se borra una llamada de FB que contiene llamadas de "ALARM_DQ". Si, al efectuar una modificación del programa, se carga de nuevo un FB con llamadas de "ALARM_DQ", puede suceder que "ALARM_DQ" no genere más avisos de PLC.

La instrucción "ALARM_DQ" tiene un parámetro más que las instrucciones "ALARM_SQ (Página 6939)": se trata de la entrada CMP_ID. Con ella se asignan los avisos de PLC generados con las instrucciones "ALARM_DQ" a áreas lógicas, p. ej. instalaciones parciales. Si se llama "ALARM_DQ" en un FB, es conveniente asignar a "CMP_ID" el número del DB de instancia correspondiente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_DQ":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SIG	Input	BOOL	I, Q, M, D, L	La señal que dispara el aviso
ID	Input	WORD	I, Q, M, D, L o constante	Canal de datos para avisos: W#16#EEEE
EV_ID	Input	C_ALARM_S	I, Q, M, D, L	Número de aviso (no permitido: 0)
CMP_ID	Input	DWORD	I, Q, M, D, L o constante	component identifier (no autorizado: 0); Identificador del sistema parcial al que está asignado el aviso en cuestión Valores recomendados: • low word: de 1 a 65535 • high word: 0 Atendiendo a esta recomendación, se evitan posibles conflictos con paquetes de programación creados por SIEMENS.
SD	Input	ANY	I, Q, M, D, T, C	Valor asociado Longitud máxima: 12 bytes Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME
RET_VAL	Return	INT	I, Q, M, D, L	Mostrar

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
0001	• El valor asociado supera la longitud máxima admisible, o • No es posible acceder a la memoria de usuario local (p. ej. acceso a un DB borrado) El aviso se envía. • El valor asociado señala un valor en el área de datos local. El aviso se envía. (solo S7-400)
0002	Advertencia: Se ha asignado el último espacio de memoria libre para acuse de avisos. (solo S7-400)
8081	La EV_ID indicada se encuentra fuera del rango admisible.
8082	Se pierde el aviso, ya que la CPU no tiene más recursos libres para generar avisos de PLC mediante instrucciones.
8083	Se pierde el aviso, ya que ese cambio de señal ya existe, pero todavía no se ha podido enviar (desbordamiento de señal).
8084	En la llamada actual y anterior a "ALARM_DQ", la señal que dispara el aviso SIG tiene el mismo valor.
8085	No existe registro para la EV_ID indicada.
8086	Ya se está procesando en una clase de prioridad menor una llamada para la EV_ID indicada.
8087	En la primera llamada de "ALARM_DQ", la señal que dispara el aviso tenía el valor "0".
8088	La EV_ID indicada ya está ocupada por otro recurso del sistema (ALARM_SQ (Página 6939), ALARM_S (Página 6936), ALARM_D (Página 6941)).
8089	Se ha asignado a CMP_ID el valor "0".
808A	CMP_ID no concuerda con EV_ID
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Consulte también

Crear avisos de programa (Página 3178)

ALARM_SC: Determinar el estado de acuse del último aviso entrante ALARM_SQ

Descripción

Con ayuda de esta instrucción es posible determinar

- el estado de acuse del último aviso entrante de "ALARM_SQ (Página 6939)" / "ALARM_DQ (Página 6943)" y el estado de la señal que ha disparado el aviso en la última llamada de "ALARM_SQ (Página 6939)" / "ALARM_DQ (Página 6943)", o bien
- el estado de la señal que ha disparado el aviso en la última llamada de "ALARM_S (Página 6936)" / "ALARM_D (Página 6941)"

. Si se han asignado los números de aviso con ayuda de la configuración de avisos, el aviso o la señal están referenciados de manera unívoca a través del número de aviso especificado por el usuario.

La instrucción accede a la memoria ocupada temporalmente de las instrucciones "ALARM_SQ (Página 6939)", "ALARM_S (Página 6936)", "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ALARM_SC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EV_ID	Input	C_ALARM_S	I, Q, M, D, L	Número de aviso para el que se desea conocer el estado de señal en la última llamada o el estado de acuse del último aviso entrante (solo con ALARM_SQ y con ALARM_DQ)
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
STATE	Output	BOOL	I, Q, M, D, L	Estado de la señal que ha disparado el aviso en la última llamada
Q_STATE	Output	BOOL	I, Q, M, D, L	• si el parámetro EV_ID indicado corresponde a una llamada de "ALARM_S" / "ALARM_D": "1" • si el parámetro EV_ID indicado corresponde a una llamada de "ALARM_SQ" / "ALARM_DQ": Estado de acuse del último aviso entrante: – 0: no acusado – 1: acusado

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8081	La EV_ID indicada se encuentra fuera del rango admisible.
8082	En este momento no hay ningún espacio de memoria asignado para esta EV_ID (posible causa: La señal correspondiente todavía no ha tenido nunca el estado "1", o ya ha vuelto a adoptar el estado "0").
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Consulte también

Crear avisos de programa (Página 3178)

WR_USMSG: Escribir evento de diagnóstico de usuario en el búfer de diagnóstico

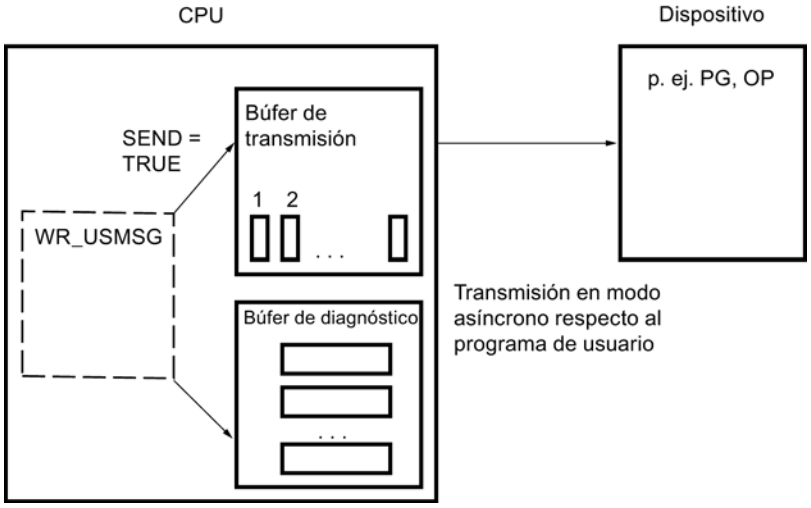
Descripción

Esta instrucción permite escribir un evento de diagnóstico del usuario en el búfer de diagnóstico. Además es posible enviar el correspondiente aviso de diagnóstico del usuario a todos los dispositivos registrados a tal efecto (asignando el parámetro de entrada SEND = TRUE). Cuando se produce un error, el parámetro de salida RET_VAL contiene la información de error.

Enviar aviso de diagnóstico del usuario

Con la instrucción, un evento de diagnóstico de usuario se registra en el búfer de diagnóstico. Además es posible enviar el correspondiente aviso de diagnóstico del usuario también a todos los dispositivos registrados al efecto (asignando el parámetro de entrada SEND = TRUE). En ese caso, el aviso de diagnóstico del usuario se escribe en el búfer de transmisión y desde allí se envía automáticamente a los dispositivos registrados al efecto.

Se puede comprobar si en el momento actual es posible enviar avisos de diagnóstico del usuario. Para ello se llama la instrucción "RDSYSST (Página 6963)" con los parámetros SZL_ID = W#16#0132 y INDEX = W#16#0005. La cuarta palabra del registro que se devuelve indica si en el momento actual la transmisión es posible (1) o no (0).



Búfer de transmisión lleno

Para introducir un aviso de diagnóstico del usuario en el búfer de transmisión es necesario que éste no se encuentre lleno. La cantidad de entradas posibles en el búfer de transmisión depende del tipo de CPU.

Cuando el búfer de transmisión está lleno, entonces

- se registra pese a ello el evento de diagnóstico en el búfer de diagnóstico
- se indica en el parámetro RET_VAL que el búfer de transmisión está lleno. (RET_VAL = W#16#8092).

Dispositivo no registrado

Si se debe enviar un aviso de diagnóstico de usuario (SEND = TRUE) y no hay ningún dispositivo registrado, entonces

- se registra pese a ello el evento de diagnóstico de usuario en el búfer de diagnóstico.
- se indica en el parámetro RET_VAL que no hay ningún dispositivo registrado (W#16#0091 o W#16#8091. El valor W#16#8091 aparece en versiones antiguas de la CPU).

Estructura de una entrada

Las entradas en el búfer de diagnóstico tienen la siguiente estructura interna:

Byte	Contenido
1 y 2	ID de evento
3	Clase de prioridad
4	Número de OB
5 y 6	Reservado
7 y 8	Información adicional 1
9, 10, 11 y 12	Información adicional 2
de 13 a 20	Sello de tiempo

ID de evento

Cada evento tiene asignada una ID de evento.

Información adicional

La información adicional contiene datos adicionales sobre el evento. El contenido de la información adicional puede variar según el evento en cuestión. Si se genera un evento de diagnóstico de usuario, se puede definir de modo personalizado el contenido de estas entradas.

Cuando se envía un aviso de diagnóstico de usuario, se puede integrar la información adicional en el texto del aviso (específico de ID de evento) en forma de valores asociados.

Sello de tiempo

El sello de tiempo es del tipo Date_and_Time.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WR_USMSG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SEND	Input	BOOL	I, Q, M, D, L	Habilitar el envío del aviso de diagnóstico de usuario a todos los dispositivos registrados
EVENTN	Input	WORD	I, Q, M, D, L o constante	ID de evento. El usuario asigna la ID de evento. La asignación predeterminada no se realiza mediante el servidor de avisos.
INFO1	Input	ANY	I, Q, M, D, L	Información adicional: 1 palabra de longitud
INFO2	Input	ANY	I, Q, M, D, L	Información adicional: 2 palabras de longitud
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro SEND

Con SEND = TRUE se envía el aviso de diagnóstico del usuario a todos los dispositivos registrados al efecto. La transmisión solo se realiza si hay por lo menos un dispositivo registrado y el búfer de transmisión no está lleno. La transmisión se realiza de modo asíncrono respecto al programa de usuario.

Parámetro EVENTN

El parámetro EVENTN contiene la ID de evento del evento de usuario. Pueden registrarse IDs de evento con el formato W#16#8xyz , W#16#9xyz, W#16#Axyz, W#16#Bxyz.

Las IDs con formato W#16#8xyz y W#16#9xyz pertenecen a eventos predefinidos, y las IDs con formato W#16#Axyz y W#16#Bxyz , a eventos personalizables.

Un evento entrante se identifica mediante x = 1, y un evento saliente con x = 0. Para eventos de clase A y B se aplica lo siguiente: yz es el número, en formato hexadecimal, asignado al aviso en cuestión en la configuración de avisos.

Parámetro INFO1

El parámetro INFO1 contiene una información de una palabra de longitud. Para INFO1 se admiten los siguientes tipos de datos:

- WORD
- INT
- ARRAY [0...1] OF CHAR

Se puede integrar el parámetro INFO1 como valor asociado en el texto del aviso para así añadir al aviso información actual.

Parámetro INFO2

El parámetro INFO2 contiene una información de dos palabras de longitud. Para INFO2 se admiten los siguientes tipos de datos:

- DWORD
- DINT
- REAL
- TIME
- ARRAY [0...3] OF CHAR

Se puede integrar el parámetro INFO2 como valor asociado en el texto del aviso para así añadir al aviso información actual.

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Ningún error
0091	No hay ningún dispositivo registrado (se registra el evento de diagnóstico en el búfer de diagnóstico)
8083	Tipo de datos INFO1 no admisible
8084	Tipo de datos INFO2 no admisible
8085	EVENTN no admisible
8086	Longitud de INFO1 no admisible
8087	Longitud de INFO2 no admisible
8091	(Este código de error aparece solo en versiones antiguas de la CPU.) No hay ningún dispositivo registrado (se registra el evento de diagnóstico en el búfer de diagnóstico)
8092	En este momento no es posible la transmisión: búfer de transmisión lleno (se registra el evento de diagnóstico en el búfer de diagnóstico)
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

AR_SEND: Enviar datos de fichero

Descripción

Esta instrucción envía datos de fichero a sistemas de control y observación registrados al efecto. Estos notifican a la CPU el número de fichero relevante en el telegrama de registro. Los datos de fichero pueden tener un tamaño de hasta 65534 bytes, en función de la memoria de trabajo de la CPU y del área de operandos utilizada. El formato de los datos de fichero debe tener en cuenta las especificaciones del sistema de control y observación utilizado.

El proceso de transmisión se activa después de la llamada del bloque y de un flanco ascendente en la entrada de control REQ. La dirección inicial de los datos de fichero que se deben enviar se define mediante SD_1, y la longitud del bloque de datos, mediante LEN. El envío de los datos se efectúa de modo asíncrono respecto al procesamiento del programa de usuario. La finalización correcta del proceso de transmisión se indica con "1" en el parámetro

de estado DONE. Con un flanco ascendente en la entrada de control R se cancela el proceso de transmisión en curso.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "AR_SEND":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control request
R	Input	BOOL	I, Q, M, D, L	Parámetro de control reset: Interrupción de la petición actual
ID	Input	WORD	I, Q, M, D o constante	Canal de datos para avisos: W#16#EEEE ID solo se evalúa en la primera llamada.
AR_ID	Input	C_AR_SEND	I, Q, M, D, L o constante	Número de fichero (no autorizado: 0); AR_ID solo se evalúa en la primera llamada. Tras ello, en cada llamada de "AR_SEND" con el DB de instancia correspondiente, se utiliza el mismo número de fichero que en la primera llamada. El número de fichero se asigna automáticamente. Con ello se garantiza la coherencia de los números de fichero.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: Transmisión finalizada
ERROR	Output	BOOL	I, Q, M, D, L	El parámetro de estado ERROR: ERROR=TRUE significa que se ha producido un error durante el procesamiento. Para más detalles, véase el parámetro STATUS.
STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS: Ver una información de error
SD_1	InOut	ANY	I, Q, M, D, T, C	Puntero a datos de fichero. La indicación de longitud no se evalúa. Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. Los datos de fichero deben tener un formato adecuado para el sistema de destino. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (por ej.: P# DB10.DBX5.0 byte 10).
LEN	InOut	WORD	I, Q, M, D, L	Longitud en bytes del bloque de datos que debe transferirse

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La tabla siguiente contiene toda la información de error específica que puede devolverse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación
1	2	Acuse negativo, la función no puede ejecutarse
1	3	No existe registro para la AR_ID indicada.
1	4	• Error en el puntero a datos de fichero SD_1 referente a la longitud o al tipo de datos • En la primera llamada, la AR_ID indicada se encuentra fuera del rango admisible.
1	5	Se ha ejecutado la solicitud de reset
1	7	La petición RESETes irrelevante, ya que la función en curso ha finalizado o no ha sido activada (bloque en estado erróneo)
1	10	No es posible acceder a la memoria de usuario local (p. ej. acceso a un DB borrado)
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a AR_SEND • No se ha indicado ningún DB de instancia, sino un DB global
1	18	AR_ID ya ha sido utilizada por un AR_SEND.
1	20	La memoria de trabajo disponible no es suficiente.

Coherencia de datos

Para garantizar la coherencia de datos, no puede volver a escribirse en la parte del área de transmisión SD_1 utilizada actualmente hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

Otras

EN_MSG: Habilitar avisos PLC

Descripción

Esta instrucción permite volver a habilitar avisos de PLC bloqueados. Es indiferente que el bloqueo se haya realizado desde un visualizador o con "DIS_MSG (Página 6954)".

Los avisos que se deben habilitar se indican mediante los parámetros de entrada MODEy MESGN. El requisito para el inicio correcto de una habilitación con esta instrucción es que en el momento actual no esté activo ningún otro bloqueo de la misma instrucción.

Para iniciar la habilitación, debe ajustarse a "1" el parámetro de entrada REQ.

Funcionamiento

La habilitación se efectúa de modo asíncrono, es decir, puede prolongarse durante varias llamadas de la instrucción (ver también Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)):

- En la primera llamada (REQ =1), la instrucción comprueba los parámetros de entrada e intenta asignar los recursos de sistema necesarios. Si la comprobación da resultado positivo, se efectúa el registro en RET_VAL W#16#7001, se activa BUSY, y se lanza la habilitación.
  De lo contrario, se registra en RET_VAL la correspondiente información de error, y la petición finaliza. En tal caso, BUSY no debe evaluarse.
- Si hay llamadas intermedias, se efectúa el registro en RET_VAL W#16#7002 (la CPU todavía procesa la petición), y se activa BUSY. Las llamadas intermedias no afectan a la petición en curso.
- En la última llamada, se efectúa el registro en RET_VALW#16#0000 si no se han detectado errores. En ese caso, BUSY se ajusta a 0. En caso de error, se registra la información de error en RET_VAL , y BUSY no debe evaluarse.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "EN_MSG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Lanzar habilitación
MODE	Input	BYTE	I, Q, M, D, L o constante	Parámetros para seleccionar los avisos que deben habilitarse: ver la tabla siguiente;
MESGN	Input	DWORD	I, Q, M, D, L o constante	Número de aviso Relevante solo para MODE = 6. Esto permite habilitar un aviso individual.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error (ver abajo);
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La habilitación todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

La tabla siguiente recoge los valores admisibles del parámetro de entrada MODE .

Valor	Significado
0	Todos los avisos de PLC de la CPU generados con instrucciones
1	Todos los avisos de PLC de la CPU generados con instrucciones, es decir, todos los avisos de "NOTIFY (Página 6928)", "NOTIFY_8P (Página 6920)", "ALARM (Página 6930)", "ALARM_8P (Página 6925)" y "ALARM_8 (Página 6923)"
6	Alarma individual de la clase "Avisos PLC generados con instrucciones"

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	La habilitación ha finalizado sin errores.
7000	Primera llamada con REQ =0: No se ha activado la habilitación.
7001	Primera llamada con REQ =1: Se ha lanzado la habilitación.
7002	Llamada intermedia: La habilitación ya está activa.
8081	Error al acceder a un parámetro
8082	MODE tiene un valor no válido.
8083	El número de aviso se encuentra fuera del rango de valores admisible.
8084	No hay ningún registro para el/los aviso(s) especificado(s) mediante MODE y en su caso mediante MESGN .
80C3	La habilitación del/los aviso(s) especificado(s) mediante MODE y en su caso mediante MESGN no se puede lanzar en este momento, porque ya está activa otra habilitación de "EN_MSG ".
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

DIS_MSG: Bloquear avisos PLC

Descripción

Esta instrucción permite bloquear avisos de PLC generados con instrucciones. Los avisos que se deben bloquear se definen mediante los parámetros de entrada MODEy MESGN. El requisito para un bloqueo correcto con la instrucción "DIS_MSG" es que en el momento actual no esté activo ningún otro bloqueo con "DIS_MSG".

Los avisos que al llamar "DIS_MSG" ya están preparados para ser enviados, pero todavía se encuentran en un búfer interno, no se ven afectados por el bloqueo, y por lo tanto se transmiten.

El bloqueo de avisos se notifica en las salidas ERROR y STATUS de las instrucciones "NOTIFY (Página 6928)", "NOTIFY_8P (Página 6925)", "ALARM (Página 6930)", "ALARM_8P (Página 6925)" y "ALARM_8 (Página 6923)".

Para iniciar el proceso, al llamar "DIS_MSG" debe ajustarse a "1"el parámetro de entrada REQ.

Funcionamiento

El bloqueo se efectúa de modo asíncrono, es decir, puede prolongarse durante varias llamadas (ver también Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)):

- En la primera llamada (REQ =1), "DIS_MSG" comprueba los parámetros de entrada e intenta asignar los recursos de sistema necesarios. Si la comprobación da resultado positivo, se efectúa el registro en RET_VALW#16#7001, se activa BUSY y se lanza el bloqueo.
  De lo contrario, se registra en RET_VAL la correspondiente información de error, y la petición finaliza. En tal caso, BUSY no debe evaluarse.
- Si hay llamadas intermedias, se efectúa el registro en RET_VALW#16#7002 (la CPU todavía procesa la petición), y se activa BUSY. Las llamadas intermedias no afectan a la petición en curso.
- En la última llamada, se efectúa el registro en RET_VALW#16#0000, si no se han detectado errores. En este caso, BUSY se ajusta a "0". En caso de error, se registra la información de error en RET_VAL, y BUSY no debe evaluarse.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DIS_MSG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Lanzar bloqueo
MODE	Input	BYTE	I, Q, M, D, L o constante	Parámetros para seleccionar los avisos que deben bloquearse: ver la tabla siguiente
MESGN	Input	DWORD	I, Q, M, D, L o constante	Número de aviso Relevante solo para MODE= 5, 6, 7. Esto permite bloquear un aviso individual.
RET_VAL	Return	INT	I, Q, M, D, L	Código de error (ver abajo)
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: El bloqueo todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro MODE

La tabla siguiente recoge los valores admisibles del parámetro de entrada MODE .

Valor	Significado
0	Todos los avisos de PLC de la CPU generados con instrucciones
1	Todos los avisos de PLC de la CPU generados con instrucciones, es decir, todos los avisos de "NOTIFY (Página 6928)", "NOTIFY_8P (Página 6920)", "ALARM (Página 6930)", "ALARM_8P (Página 6925)" y "ALARM_8 (Página 6923)"
6	Alarma individual de la clase "Avisos PLC generados con instrucciones"

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	El bloqueo ha finalizado sin errores.
7000	Primera llamada con REQ =0: No se ha activado el bloqueo.
7001	Primera llamada con REQ =1: Se ha lanzado el bloqueo.
7002	Llamada intermedia: El bloqueo ya está activo.
8081	Error al acceder a un parámetro
8082	MODE tiene un valor no válido.
8083	El número de aviso se encuentra fuera del rango de valores admisible.
8084	No hay ningún registro para el/los aviso(s) especificado(s) mediante MODE y en su caso mediante MESGN .
80C3	El bloqueo del/los aviso(s) especificado(s) mediante MODE y en su caso mediante MESGN no se puede lanzar en este momento, porque ya está activo otro bloqueo de "DIS_MSG".
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

READ_SI: Leer recursos del sistema asignados dinámicamente

Creación de recursos del sistema asignados dinámicamente al generar avisos con las instrucciones "ALARM_DQ" y "ALARM_D"

Al crear avisos con "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)", el sistema operativo asigna temporalmente espacio de almacenamiento en la memoria del sistema.

Si p. ej. se borra un FB presente en la CPU mediante llamadas de "ALARM_DQ (Página 6943)" o "ALARM_D (Página 6941)", puede suceder que los correspondientes recursos del sistema queden ocupados de modo permanente. Si se vuelve a cargar el FB con llamadas de "ALARM_DQ (Página 6943)" o "ALARM_D (Página 6941)", puede suceder que las instrucciones "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)" no se procesen debidamente.

Descripción

Con la instrucción "READ_SI" se pueden leer los recursos del sistema utilizados actualmente que se han ocupado al generar avisos con "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)". Para esto se emplean los valores EV_ID y CMP_ID. Dichos valores se envían a la instrucción "READ_SI" en el parámetro SI_ID .

La instrucción "READ_SI" tiene 4 modos de operación posibles, que se enumeran en la tabla siguiente. El modo de operación deseado se ajusta con el parámetro MODE.

MODE	¿Qué recursos del sistema ocupados por "ALARM_DQ" / "ALARM_D" se leen?
1	Todos (la llamada de "READ_SI" se efectúa con SI_ID:=0)
2	El recurso del sistema que se ha ocupado en la llamada de "ALARM_DQ" / "ALARM_D" con EV_ID:=ev_id (la llamada de READ_SI se efectúa con SI_ID:=ev_id)

MODE	¿Qué recursos del sistema ocupados por "ALARM_DQ" / "ALARM_D" se leen?
3	Todos los recursos del sistema que se han ocupado en la llamada de "ALARM_DQ" / "ALARM_D" con CMP_ID:=cmp_id (la llamada de READ_SI se efectúa con SI_ID:=cmp_id)
0	Otros recursos del sistema que no se han podido leer en la llamada anterior con MODE=1 o MODE=3 , por haberse elegido un campo de destino SYS_INST demasiado pequeño

Funcionamiento

Si al llamar READ_SI con MODE=1 o MODE=3 se ha escogido un área de destino SYS_INST lo suficientemente grande, dicha área contendrá después de la llamada el contenido de todos los recursos del sistema seleccionados mediante el parámetro MODE que estén ocupados en el momento actual.

Si en el momento actual hay una gran cantidad de recursos del sistema ocupados, el tiempo de ejecución será prolongado. Si la CPU tiene mucha carga de trabajo, puede ser que se rebase el tiempo máximo de vigilancia de ciclo configurado.

Este problema de tiempo de ejecución puede resolverse de la siguiente manera: Escoja un área de destino SYS_INST relativamente pequeña. Si la instrucción no puede guardar en SYS_INST todos los recursos del sistema que se deben leer, se recibirá una notificación a través de RET_VAL=W#16#0001. En tal caso, debe llamarse READ_SI, con MODE=0 y la misma SI_ID que en la llamada anterior, tantas veces como sea necesario para que RET_VAL adquiera el valor W#16#0000.

Nota

Dado que el sistema operativo no coordina las llamadas de READ_SIcorrespondientes a una petición de lectura, todas las llamadas de READ_SIdeben efectuarse en la misma clase de prioridad.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "READ_SI":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	ID de la petición Valores permitidos: • 1: Leer todos los recursos del sistema • 2: Leer el recurso del sistema que se ha ocupado en la llamada de "ALARM_DQ" / "ALARM_D" con EV_ID = ev_id • 3: Leer los recursos del sistema que se han ocupado en la llamada de "ALARM_DQ" / "ALARM_D" con CMP_ID = cmp_id • 0: Llamada consecutiva
SI_ID	Input	DWORD	I, Q, M, D, L o constante	Identificador para el/los recurso(s) del sistema que se debe(n) leer Valores permitidos: • 0, si MODE=1 • Número de aviso ev_id, si MODE=2 • Identificador cmp_id del sistema parcial, si MODE=3
RET_VAL	Return	INT	I, Q, M, D, L	Valor de retorno (información de error o estado de la petición)
N_SI	Output	INT	I, Q, M, D, L	Número de recursos del sistema que se muestran en SYS_INST
SYS_INST	Output	ANY	D	Área de destino para los recursos del sistema leídos

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
0001	No se han podido leer todos los recursos del sistema, porque se ha elegido un área de destino SYS_INST demasiado pequeña.
8081	(solo con MODE=2 ó 3) Se ha asignado a SI_ID el valor 0.
8082	(solo con MODE=1) Se ha asignado a SI_ID un valor distinto de 0.
8083	(solo con MODE=0) Se ha asignado a SI_ID un valor distinto al de la llamada anterior con MODE=1 ó 3.
8084	Se ha asignado a MODE un valor no admisible.
8085	READ_SI ya está procesándose en otro OB.
8086	El área de destino SYS_INST es demasiado pequeña para un recurso del sistema.

Código de error (W#16#...)	Explicación
8087 o 8092	El área de destino SYS_INST no se encuentra en un DB, o hay un error en el puntero ANY.
8xyy	Información de error general Consulte también: ALARM_D: Generar siempre avisos PLC acusados (Página 6941)

Parámetro Área de destino SYS_INST: Estructura

El área de destino para los recursos del sistema ocupados leídos debe encontrarse dentro de un DB. Es recomendable definir el área de destino como un campo de estructuras, cada una de ellas con el siguiente formato:

Elemento de estructura	Tipo de datos	Descripción
SFC_NO	WORD	N.º de la instrucción que ha asignado el recurso del sistema
LEN	BYTE	Longitud en bytes de la estructura, incl. SFC_NO y LEN: B#16#0C
SIG_STAT	BOOL	Estado de señal
ACK_STAT	BOOL	Estado de acuse del evento entrante (flanco ascendente)
EV_ID	DWORD	Número de aviso
CMP_ID	DWORD	Identificador del sistema parcial

DEL_SI: Borrar recursos del sistema asignados dinámicamente

Creación de recursos del sistema asignados dinámicamente al generar avisos con las instrucciones "ALARM_DQ" y "ALARM_D"

Al crear avisos con "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)", el sistema operativo asigna temporalmente espacio de almacenamiento en la memoria del sistema.

Si p. ej. se borra un FB presente en la CPU mediante llamadas de "ALARM_DQ (Página 6943)" o "ALARM_D (Página 6941)", puede suceder que los correspondientes recursos del sistema queden ocupados de modo permanente. Si se vuelve a cargar el FB con llamadas de "ALARM_DQ (Página 6943)" / "ALARM_D (Página 6941)", puede suceder que las instrucciones "ALARM_DQ (Página 6943)" y "ALARM_D (Página 6941)" no se procesen debidamente.

Descripción

La instrucción "DEL_SI" permite borrar recursos del sistema utilizados en el momento actual.

"DEL_SI" tiene tres modos de operación posibles, que se enumeran en la tabla siguiente. El modo de operación deseado se ajusta con el parámetro MODE .

MODE	¿Qué recursos del sistema ocupados por "ALARM_DQ" / "ALARM_D" se borran?
1	Todos (la llamada de "DEL_SI" se efectúa con SI_ID:=0)
2	El recurso del sistema que se ha ocupado en la llamada de "ALARM_DQ" (Página 6943) / ALARM_D (Página 6941) con EV_ID:=ev_id (la llamada de DEL_SI se efectúa con SI_ID:=ev_id)
3	Todos los recursos del sistema que se han ocupado en la llamada de "ALARM_DQ" (Página 6943) / ALARM_D (Página 6941) con CMP_ID:=cmp_id (la llamada de DEL_SI se efectúa con SI_ID:=cmp_id)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DEL_SI":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MODE	Input	INT	I, Q, M, D, L o constante	ID de la petición Valores permitidos: • 1: Borrar todos los recursos del sistema • 2: Borrar el recurso del sistema que se ha ocupado en la llamada de "ALARM_DQ" (Página 6943) / "ALARM_D (Página 6941)" con EV_ID = ev_id • 3: Borrar los recursos del sistema que se han ocupado en la llamada de "ALARM_DQ" (Página 6943) / "ALARM_D (Página 6941)" con CMP_ID = cmp_id
SI_ID	Input	DWORD	I, Q, M, D, L o constante	Identificador para el/los recurso(s) del sistema que se debe(n) borrar Valores permitidos: • 0, si MODE=1 • Número de aviso ev_id, si MODE=2 • Identificador cmp_id del sistema parcial, si MODE=3
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8081	(solo con MODE=2 ó 3) Se ha asignado a SI_ID el valor "0".
8082	(solo con MODE=1) Se ha asignado a SI_ID un valor distinto de "0".
8084	Se ha asignado a MODE un valor no admisible.
8085	"DEL_SI" ya se está procesando en este momento.

Código de error (W#16#...)	Explicación
8086	No se han podido borrar todos los recursos del sistema seleccionados, ya que por lo menos uno de ellos estaba procesándose en el momento de la llamada de "DEL_SI".
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Diagnóstico

RD_SINFO: Leer información de arranque del OB actual

Descripción

Esta instrucción permite leer la información de arranque:

- Del último OB llamado que todavía no se ha procesado completamente
- Del último OB de arranque iniciado

Ambas informaciones de arranque carecen de sello de tiempo. Si la llamada se produce en el OB 100, el OB 101 o el OB 102, se devuelven dos informaciones de arranque idénticas.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_SINFO".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
TOP_SI	Output	STRUCT	D, L	Información de arranque del OB actual
START_UP_SI	Output	STRUCT	D, L	Información de arranque del último OB de arranque iniciado

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

La instrucción no devuelve información de error específica, sino únicamente general. La información de error general y su evaluación se describen en el parámetro de salida "RET_VAL".

Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598).

Parámetros TOP_SI y START_UP_SI

Los parámetros de salida TOP_SI y START_UP_SI son dos estructuras con formato idéntico. El formato de ambas se muestra en la tabla siguiente.

Elemento de estructura	Tipo de datos	Descripción
EV_CLASS	BYTE	• Bits de 0 a 3: ID de evento • Bits de 4 a 7: Clase de evento
EV_NUM	BYTE	Número de evento
PRIORITY	BYTE	Número del nivel de ejecución (significado de B#16#FE: El OB no existe, está bloqueado o no puede arrancarse en el actual estado operativo.)
NUM	BYTE	Número de OB
TYP2_3	BYTE	ID de datos 2_3: identifica la información registrada en ZI2_3
TYP1	BYTE	ID de datos 1: identifica la información registrada en ZI1
ZI1	WORD	Información adicional 1
ZI2_3	DWORD	Información adicional 2_3

Nota

Los elementos estructurales indicados en la tabla anterior coinciden exactamente en su contenido con las variables temporales de un OB.

Sin embargo, debe tenerse en cuenta que, en los distintos OBs, las variables temporales pueden tener otros nombres y tipos de datos. También debe tenerse en cuenta que la interfaz de llamada de los OBs contiene además la fecha y hora en que se solicitó el OB.

Los bits de 4 a 7 del elemento estructural EV_CLASS contienen la clase de evento. Aquí son posibles los siguientes valores:

- 1: Eventos de arranque de OBs estándar
- 2: Eventos de arranque de OBs de errores síncronos
- 3: Eventos de arranque de OBs de errores asíncronos

El elemento estructural PRIORITY devuelve la clase de prioridad correspondiente al OB actual.

Además de estos dos elementos, también es importante NUM . NUM contiene el número del OB actual o del último OB de arranque iniciado.

Ejemplo

Supongamos que el último OB llamado, que todavía no se ha procesado por completo, es el OB 80, y el último OB de arranque llamado es el OB 100.

La tabla siguiente muestra la correspondencia entre los elementos estructurales del parámetro TOP_SI de la instrucción "RD_SINFO" y las correspondientes variables locales del OB 80.

TOP_SI Elemento de estructura	Tipo de datos	OB 80 - Variable local correspondiente	Tipo de datos
EV_CLASS	BYTE	OB80_EV_CLASS	BYTE
EV_NUM	BYTE	OB80_FLT_ID	BYTE
PRIORITY	BYTE	OB80_PRIORITY	BYTE
NUM	BYTE	OB80_OB_NUMBR	BYTE
TYP2_3	BYTE	OB80_RESERVED_1	BYTE
TYP1	BYTE	OB80_RESERVED_2	BYTE
ZI1	WORD	OB80_ERROR_INFO	WORD
ZI2_3	DWORD	OB80_ERR_EV_CLASS	BYTE
		OB80_ERR_EV_NUM	BYTE
		OB80_OB_PRIORITY	BYTE
		OB80_OB_NUM	BYTE

La tabla siguiente muestra la correspondencia entre los elementos estructurales del parámetro START_UP_SI de la instrucción "RD_SINFO" y las correspondientes variables locales del OB 100.

START_UP_SI Elemento de estructura	Tipo de datos	OB 100 - Variable local	Tipo de datos
EV_CLASS	BYTE	OB100_EV_CLASS	BYTE
EV_NUM	BYTE	OB100_STRTUP	BYTE
PRIORITY	BYTE	OB100_PRIORITY	BYTE
NUM	BYTE	OB100_OB_NUMBR	BYTE
TYP2_3	BYTE	OB100_RESERVED_1	BYTE
TYP1	BYTE	OB100_RESERVED_2	BYTE
ZI1	WORD	OB100_STOP	WORD
ZI2_3	DWORD	OB100_STRT_INFO	DWORD

RDSYSST: Leer lista de estado del sistema

Descripción

Esta instrucción permite leer una lista parcial de las listas de estado del sistema (SZL) o un extracto de lista parcial de SZL.

Para iniciar la lectura, al llamar "RDSYSST" debe ajustarse a "1" el parámetro de entrada REQ . Si la operación de lectura se ha podido ejecutar de inmediato, la instrucción devuelve

el valor "0" en el parámetro de salida BUSY. Si BUSY tiene el valor "1", la operación de lectura todavía no ha finalizado.

Nota

Si se llama la instrucción "RDSYSST" en el OB de alarma de diagnóstico con la SZL-IDW#16#00B1, W#16#00B2 o W#16#00B3 y se accede al módulo que ha provocado la alarma de diagnóstico, la lectura se efectúa de inmediato.

Con "RDSYSST" se transfieren siempre registros enteros.

Recursos del sistema

Si se lanzan varias operaciones de lectura con poca diferencia de tiempo entre sí (las peticiones con los parámetros SZL_IDW#16#00B4, W#16#4C91, W#16#4092, W#16#4292, W#16#4692 y, en su caso, W#16#00B1 y W#16#00B3), el sistema operativo garantiza la ejecución de todas las peticiones sin que se produzcan interferencias entre ellas. En el momento en que se alcanza el límite de los recursos del sistema, se envía una notificación mediante RET_VAL. El estado de fallo temporal puede eliminarse repitiendo la petición.

El número máximo de peticiones activas "simultáneamente" para la instrucción depende de la CPU.

Nota

En las CPU S7-400 puede transferirse un máximo de 432 bytes al área de destino con la instrucción "RDSYSST".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RDSYSST":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ=1: Lanzar el procesamiento
SZL_ID	Input	WORD	I, Q, M, D, L o constante	SZL-ID de la lista parcial o del extracto de lista parcial
INDEX	Input	WORD	I, Q, M, D, L o constante	Tipo o número de un objeto en una lista parcial
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el parámetro RET_VAL contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	TRUE: La lectura todavía no ha finalizado.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SZL_HEADER	Output	STRUCT	D, L	Véase más abajo
DR	Output	ANY	I, Q, M, L, D	Área de destino para la lista parcial SZL leída o el extracto de lista parcial SZL leído: • Si solo se ha leído la información de encabezado de una lista parcial SZL, no se debe evaluar DR, sino únicamente SZL_HEADER. • De lo contrario, el producto de LENTHDR y N_DR indica cuántos bytes se han registrado en DR.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro SZL_HEADER

El parámetro SZL_HEADER es una estructura que se define del siguiente modo:

```
SZL_HEADER:   STRUCT
    LENTHDR:  WORD
    N_DR:     WORD
END_STRUCT
```

LENTHDR es la longitud de un registro de la lista parcial SZL o del extracto de lista parcial SZL.

- Si solo se ha leído la información de encabezado de una lista parcial SZL, N_DR contiene el número de registros relevantes presentes.
- De lo contrario, N_DR contiene el número de registros transferidos al área de destino.

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Ningún error
0081	La longitud del campo de resultado es insuficiente (a pesar de ello se enviarán todos los registros posibles. La cantidad puede leerse en el SZL-Header.)
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".
8081	La longitud del campo de resultado es insuficiente (no hay espacio para un registro)
8082	La SZL_ID es errónea o se desconoce en la CPU o en la instrucción.
8083	INDEX erróneo o no permitido
8085	En este momento la información no está disponible por razones atribuibles al sistema, p. ej. debido a falta de recursos.
8086	No se puede leer el registro debido a un fallo del sistema (bus, módulos, sistema operativo).

Código de error (W#16#...)	Explicación
8087	No se puede leer el registro porque el módulo no existe o no está acusado.
8088	No se puede leer el registro porque la ID de módulo real difiere de la ID de módulo configurada.
8089	No se puede leer el registro porque el módulo no es apto para diagnóstico o no soporta el registro.
80A2	Error de protocolo de DP (error Layer 2) (error temporal)
80A3	Error de protocolo de DP con User-Interface/User (error temporal)
80A4	Fallo de comunicación en el bus K (el error se produce entre la CPU y la interfaz DP externa) (error temporal)
80C5	Periferia descentralizada no disponible (error temporal)
80C6	La transferencia de registros se ha cancelado por una interrupción de la clase de prioridad (rearranque o segundo plano)
80D2	No se puede leer el registro porque el módulo no es apto para diagnóstico.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Parámetro SZL_ID

Nota

Listas parciales legibles de las CPUs S7

- Encontrará las listas parciales para S7-300 legibles mediante "RDSYSST" en la correspondiente lista de operación.
- Encontrará las listas parciales S7-400 legibles mediante "RDSYSST" en la correspondiente lista de operación.

SZL_ID (W#16#...)	Lista parcial	INDEX (W#16#...)
ID de módulo		
0111 (Página 6975)	Un registro de identificación	
	Identificación del módulo	0001
	Identificación del hardware base	0006
	Identificación del firmware base	0007
Características de la CPU		
0012 (Página 6976)	Todas las características	Irrelevante
0112 (Página 6976)	Características de un grupo	
	Unidad de procesamiento MC7	0000
	Sistema horario	0100
	Comportamiento del sistema	0200
	Descripción de lenguaje MC7	0300
	Disponibilidad de la instrucción "C_DIAG (Página 7027)"	0400

SZL_ID (W#16#...)	Lista parcial	INDEX (W#16#...)
0F12 (Página 6976)	Solo información de encabezado de la lista parcial SZL	Irrelevante
Áreas de memoria del usuario		
0113 (Página 6978)	Un registro para el área de memoria especificada	
	Memoria de trabajo	0001
Áreas del sistema		
0014 (Página 6979)	Registros de todas las áreas del sistema	Irrelevante
0F14 (Página 6979)	Solo información de encabezado de la lista parcial SZL	Irrelevante
Tipos de bloque		
0015 (Página 6981)	Registros de todos los tipos de bloque	Irrelevante
Identificación de un componente		
001C (Página 6981)	Identificación de todos los componentes	Irrelevante
011C (Página 6981)	Identificación de un componente	
	Nombre del sistema de automatización	0001
	Nombre del módulo	0002
	Identificador de la instalación del módulo	0003
	Entrada de derechos de autor	0004
	Número de serie del módulo	0005
	Nombre de tipo de módulo	0007
	Número de serie de la Memory Card	0008
	Fabricante y perfil de un módulo de CPU	0009
	Identificador de la situación de un módulo	000B
0F1C (Página 6981)	Solo información de encabezado de la lista parcial SZL	Irrelevante
Estado de alarma		
0222 (Página 6987)	Registro para la alarma especificada	Número de OB
Correspondencia entre memorias imagen parcial de proceso y CPUs		
0025 (Página 6990)	Correspondencia entre todas las memorias imagen parcial de proceso y OBs	Irrelevante
0125 (Página 6990)	Correspondencia entre una memoria imagen parcial de proceso y el OB correspondiente	N.º de memoria imagen parcial de proceso

SZL_ID (W#16#...)	Lista parcial	INDEX (W#16#...)
0225 (Página 6990)	Correspondencia entre un OB y las memorias imagen parcial de proceso correspondientes	N.º de OB
0F25 (Página 6990)	Solo información de encabezado de la lista parcial SZL	Irrelevante
Datos de estado de comunicación		
0132 (Página 6993)	Datos de estado para un componente de comunicación	
	Diagnóstico	0005
	Sistema horario	0008
0232 (Página 6993)	Datos de estado para un componente de comunicación	
	Nivel de protección de la CPU y posiciones del interruptor de manejo	0004
Estado de los LEDs del módulo (no se puede leer en todas las CPUs)		
0174 (Página 6999)	Estado de un LED	Identificador de LED
Información del sistema maestro DP		
0090 (Página 7001)	Información acerca de todos los sistemas maestros DP conocidos para la CPU	0000
0190 (Página 7001)	Información acerca de un sistema maestro DP	ID del sistema maestro DP
0F90 (Página 7001)	Solo información de encabezado de la lista parcial SZL	0000
Información de estado del módulo (se devuelven como máximo 27 registros.)		
0091 (Página 7002)	Información de estado del módulo de todos los módulos/ submódulos insertados	Irrelevante
0191 (Página 7002)	Información de estado de todos los módulos/racks no desactivados con ID de módulo errónea	Irrelevante
0291 (Página 7002)	Información de estado de módulo de todos los módulos en estado de fallo y no desactivados	Irrelevante
0391 (Página 7002)	Información de estado de módulo de todos los módulos no disponibles	Irrelevante
0591 (Página 7002)	Información de estado de módulo de todos los submódulos del módulo host	Irrelevante

SZL_ID (W#16#...)	Lista parcial	INDEX (W#16#...)
0991 (Página 7002)	Información de estado de módulo de un sistema maestro DP	ID del sistema maestro DP
0C91 (Página 7002)	Información de estado de un módulo en configuración centralizada o conectado a una interfaz DP integrada o a una interfaz PROFINET (integrada o externa)	Dirección base lógica
4C91 (Página 7002)	Información de estado de un módulo conectado a una interfaz DP externa	Dirección base lógica
0D91 (Página 7002)	Información de estado de módulo de todos los módulos del rack o la estación especificados (DP o PROFINET)	ID del rack o del sistema maestro DP y número de estación, o número de estación seguido de las dos últimas posiciones de la ID del subsistema PNIO
0E91 (Página 7002)	Información de estado de todos los módulos asignados	Irrelevante
0F91 (Página 7002)	Solo información de encabezado de la lista parcial SZL	Irrelevante
Información de estado del rack/de la estación		
0092 (Página 7007)	Consigna de estado de los racks en configuración centralizada o de las estaciones de un sistema maestro DP.	0/ID del sistema maestro DP
4092 (Página 7007)	Consigna de estado de las estaciones de un sistema maestro DP conectado a través de una interfaz DP externa.	ID del sistema maestro DP
0192 (Página 7007)	Estado de activación de las estaciones de un sistema maestro DP conectado a través de una interfaz DP integrada.	ID del sistema maestro DP
0292 (Página 7007)	Estado real de los racks en configuración centralizada o de las estaciones de un sistema maestro DP.	0/ID del sistema maestro DP
4292 (Página 7007)	Estado real de las estaciones de un sistema maestro DP conectado a través de una interfaz DP externa.	ID del sistema maestro DP
0392 (Página 7007)	Estado del respaldo por batería de un rack de una CPU, cuando falla por lo menos una batería.	0
0492 (Página 7007)	Estado de todo el respaldo por batería de todos los racks de una CPU.	0
0592 (Página 7007)	Estado de la alimentación de 24 V de todos los racks de una CPU.	0
0692 (Página 7007)	Estado de diagnóstico de los aparatos de ampliación en configuración centralizada o de las estaciones de un sistema maestro DP conectado a través de una interfaz DP integrada.	0/ID del sistema maestro DP

SZL_ID (W#16#...)	Lista parcial	INDEX (W#16#...)
4692 (Página 7007)	Estado de diagnóstico de las estaciones de un sistema maestro DP conectado a través de una interfaz DP externa.	ID del sistema maestro DP
Información de estado del rack/de la estación		
0094 (Página 7010)	Estado teórico de los racks en configuración centralizada o de las estaciones de un sistema de controlador IO.	0/ID del subsistema PNIO
0194 (Página 7010)	Estado de activación de una estación de un sistema de controlador IO configurado y desactivado.	ID del subsistema PNIO
0294 (Página 7010)	Estado real de los racks en configuración centralizada o de las estaciones de un sistema de controlador IO.	0/ID del subsistema PNIO
0694 (Página 7010)	Estado de diagnóstico de los aparatos de ampliación en configuración centralizada o de las estaciones de un sistema de controlador IO.	0/ID del subsistema PNIO
0794 (Página 7010)	Estado de mantenimiento del rack central o de las estaciones de un sistema de controlador IO.	0/ID del subsistema PNIO
0F94 (Página 7010)	Solo información de encabezado	-
Información avanzada del sistema maestro DP		
0195 (Página 7012)	Información avanzada acerca de un sistema maestro DP	ID del sistema maestro DP
0F95 (Página 7012)	Solo información de encabezado de la lista parcial SZL	0000
Información de estado del módulo con PROFINET IO y PROFIBUS DP		
0696 (Página 7013)	Información de estado de todos los submódulos de un módulo especificado (solo con PROFINET IO conectado a una interfaz integrada).	Dirección con identificación E/S
0C96 (Página 7013)	Información de estado de un módulo o submódulo central o conectado a una interfaz PROFIBUS DP integrada o a una interfaz PROFINET (integrada o externa)	Dirección inicial con identificación E/S
Búfer de diagnóstico (se devuelven como máximo 21 registros)		
00A0 (Página 7019)	Todas las entradas transmisibles en el estado operativo actual	Irrelevante
01A0 (Página 7019)	Las entradas más recientes (el número se indica mediante el índice)	Número
0FA0 (Página 7019)	Solo información de encabezado de la lista parcial SZL	Irrelevante
Datos de diagnóstico en módulos		

SZL_ID (W#16#...)	Lista parcial	INDEX (W#16#...)
00B1 (Página 7020)	Los cuatro primeros bytes de diagnóstico de un módulo (DS0)	Dirección base lógica
00B2 (Página 7021)	Todos los datos de diagnóstico de un módulo (220 bytes, DS1) (ningún módulo DP)	Rack, slot
00B3 (Página 7022)	Todos los datos de diagnóstico de un módulo (220 bytes, DS1)	Dirección base lógica
00B4 (Página 7022)	Datos de diagnóstico de un esclavo DP	Dirección de diagnóstico configurada

Lista de estado del sistema

Sinopsis de la lista de estado del sistema (SZL)

Definición: Lista de estado del sistema

La lista de estado del sistema describe el estado actual de un sistema de automatización. El contenido de la SZL (lista de estado del sistema) puede leerse, aunque no modificarse, mediante funciones de información. Las listas parciales son listas virtuales, es decir, el sistema operativo de los módulos centrales las confecciona solo a petición del usuario.

Solo es posible leer una lista parcial al mismo tiempo con "RDSYSST (Página 6963)". Aquí se describen las listas parciales de la lista de estado del sistema que pueden leerse en el programa de usuario mediante "RDSYSST (Página 6963)" y cuya información hace referencia a

- CPUs o
- a los módulos cuyas listas parciales no son específicas de módulo (p. ej. las IDs de SZL W#16#00B1, W#16#00B2, W#16#00B3).

Encontrará las listas parciales específicas de módulo, p. ej. para CPs y FMs, en las respectivas descripciones de módulo.

Contenido

La lista de estado del sistema contiene información acerca de:

- Datos del sistema
- Información de estado del módulo en la CPU
- Datos de diagnóstico en módulos
- Búfer de diagnóstico

Datos del sistema

Los datos del sistema son datos característicos de una CPU, fijos o parametrizables. Describen las siguientes características de rendimiento:

- Configuración de la CPU
- Estado de las clases de prioridad
- Comunicación.

Información de estado del módulo

La información de estado del módulo describe el estado actual de los componentes que se vigilan mediante el diagnóstico del sistema.

Datos de diagnóstico en módulos

Los módulos aptos para diagnóstico asignados a una CPU poseen datos de diagnóstico que se almacenan en los propios módulos.

Búfer de diagnóstico

El búfer de diagnóstico contiene entradas de diagnóstico en el orden en que se generan.

Estructura de una lista parcial de SZL

Principios básicos

Se puede leer una lista parcial o un extracto de lista parcial con "RDSYSST (Página 6963)". Para ello se define con los parámetros SZL_ID (ID de lista de estado del sistema) e INDEX lo que se desea leer.

Estructura

Una lista parcial se compone de

- un encabezado y
- los registros.

Encabezado

El encabezado de una lista parcial se compone de:

- SZL-ID
- INDEX
- La longitud en bytes de un registro de la lista
- El número de registros que contiene la lista parcial.

Índice

Para determinadas listas parciales o extractos de listas parciales, es necesario especificar un identificador de tipo de objeto o un número de objeto. Para ello se utiliza el índice. Si el índice no se requiere para una información, su contenido es irrelevante.

Registros

Los registros de una lista parcial tienen una longitud determinada. Esta longitud depende de qué información esté almacenada en la lista parcial. El modo en que las palabras de datos están asignadas en un registro depende también de la lista parcial.

ID de SZL

SZL-ID (ID de lista de estado del sistema)

Cada lista parcial perteneciente a la lista de estado del sistema posee un número. Las listas parciales pueden visualizarse de modo completo o parcial. Los extractos de lista parcial posibles están definidos de modo fijo y se identifican a su vez mediante un número. La SZL-ID se compone del número de la lista parcial y el número del extracto de lista parcial y de la clase de módulo.

Estructura

La SZL-ID tiene una longitud de una palabra. Se compone de los siguientes elementos:

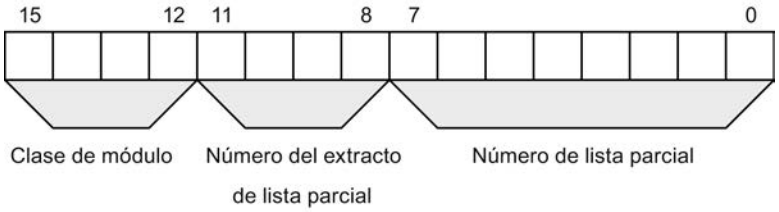


Figura 9-6 Estructura de la SZL-ID

Clase de módulo

Ejemplos de clases de módulos:

Clase de módulo	Cifrado correspondiente (binario)
CPU	0000
IM	0100
FM	1000
CP	1100

Número del extracto de lista parcial

Los números de extracto de lista parcial y su significado dependen de la lista parcial en cuestión. Con el número de extracto de lista parcial se indica qué porción de una lista parcial se desea leer.

Número de lista parcial

Con el número de lista parcial se indica qué lista parcial de la lista de estado del sistema se desea leer.

Listas parciales de SZL posibles

Subset

En un módulo siempre está disponible solo una porción de todas las listas parciales posibles. Según el módulo están disponibles unas listas parciales u otras.

Listas parciales de SZL posibles

La tabla siguiente recoge todas las listas parciales posibles con el número que les corresponde dentro de la SZL-ID (ID de lista de estado del sistema).

Lista parcial	SZL-ID
ID de módulo	W#16#xy11 (Página 6975)
Características de la CPU	W#16#xy12 (Página 6976)
Áreas de memoria del usuario	W#16#xy13 (Página 6978)
Áreas del sistema	W#16#xy14 (Página 6979)
Tipos de bloque	W#16#xy15 (Página 6981)
Identificación de un componente	W#16#xy1C (Página 6981)
Estado de alarma	W#16#xy22 (Página 6987)
Correspondencia entre memorias imagen parcial de proceso y OBs	W#16#xy25 (Página 6990)
Comunicación: Datos de estado	W#16#xy32 (Página 6993)
Estado de los LEDs de módulo	W#16#xy74 (Página 6999)
Información del sistema maestro DP	W#16#xy90 (Página 7001)
Información de estado del módulo	W#16#xy91 (Página 7002)

Lista parcial	SZL-ID
Información de estado del rack/de la estación	W#16#xy92 (Página 7007)
Información de estado del rack/de la estación	W#16#xy94 (Página 7010)
Información avanzada del sistema maestro DP	W#16#xy95 (Página 7012)
Información de estado del módulo con PROFINET IO y PROFIBUS DP	W#16#xy96 (Página 7013)
Búfer de diagnóstico	W#16#xyA0 (Página 7019)
Información de diagnóstico del módulo (DS 0)	W#16#00B1 (Página 7020)
Información de diagnóstico del módulo (DS 1), dirección física	W#16#00B2 (Página 7021)
Información de diagnóstico del módulo (DS 1), dirección lógica	W#16#00B3 (Página 7022)
Datos de diagnóstico de un esclavo DP	W#16#00B4 (Página 7022)

ID de SZL W#16#xy11: ID de módulo

Finalidad

La identificación de este módulo se obtiene a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy11.

Encabezado

El encabezado de SZL con la SZL-IDW#16#xy11 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0111: Un solo registro de identificación
INDEX	Número de un determinado registro W#16#0001: Identificación del módulo W#16#0006: Identificación del hardware base W#16#0007: Identificación del firmware base
LENTHDR	W#16#001C: Un registro tiene una longitud de 14 palabras (28 bytes)
N_DR	Número de registros

Registro

Un registro de la SZL con la SZL-IDW#16#xy11 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	Número de un registro de identificación
MRPC	20 bytes	Machine-Readable Product Code Con INDEX W#16#0007: Reservado Con INDEXW#16#0001 y W#16#0006: Referencia del módulo; cadena de 19 caracteres y un espacio en blanco (20H); p. ej. para CPU 314: "6ES7 314-0AE01-0AB0"
ModuleType	1 palabra	Reservado
OutMod1	1 palabra	Con INDEXW#16#0001: Versión del módulo Con INDEXW#16#0006 y W#16#0007: "V" y la primera cifra del identificador de versión
OutMod2	1 palabra	Con INDEXW#16#0001: Reservado Con INDEXW#16#0006y W#16#0007: Resto de cifras del identificador de versión

ID de SZL W#16#xy12: características de la CPU

Finalidad

Los módulos del tipo CPU poseen una serie de características con una clara dependencia del hardware. Cada una de estas características tiene asignado un identificador. Las características del módulo se pueden conocer a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy12.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy12 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial: W#16#0012: Todas las características W#16#0112: Características de un grupo; el grupo se indica en el parámetro INDEX. W#16#0F12: Solo información de encabezado de la lista parcial SZL
INDEX	Grupo W#16#0000: Unidad de procesamiento MC7 W#16#0100: Sistema horario W#16#0200: Comportamiento del sistema W#16#0300: Descripción de lenguaje MC7 de la CPU W#16#0400: Disponibilidad de la instrucción "C_DIAG (Página 7027)"
LENTHDR	W#16#0002: Un registro tiene una longitud de una palabra (2 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#xy12 tiene una longitud de una palabra. Para cada característica existente hay almacenado un identificador de característica. Un identificador de característica tiene una longitud de una palabra.

Nota

Se visualizan todos los registros relevantes para la CPU en uso. Se muestran uno tras otro sin espacios intermedios.

Identificador de característica

La tabla siguiente contiene todos los identificadores de características.

Identificador	Significado
W#16#0000 - 00FF	Unidad de procesamiento MC7 (grupo con índice 0000)
W#16#0001	Procesamiento MC7 generador de código
W#16#0002	Intérprete MC7
W#16#0100 - 01FF	Sistema horario (grupo con índice 0100)
W#16#0101	Resolución de 1 ms
W#16#0102	Resolución de 10 ms
W#16#0103	Sin reloj en tiempo real
W#16#0104	Formato de hora BCD
W#16#0105	Conjunto de la funcionalidad horaria (ajustar hora, activar y leer estado de hora, sincronización horaria: Reloj esclavo y reloj maestro)
W#16#0106	OB_RT está disponible
W#16#0200 - 02FF	Comportamiento del sistema (grupo con índice 0200)
W#16#0201	Apto para multiprocesamiento
W#16#0202	Arranque en frío, rearranque completo (en caliente) y rearranque disponibles
W#16#0203	Arranque en frío y rearranque completo (en caliente) disponibles
W#16#0204	Rearranque completo (en caliente) y rearranque disponibles
W#16#0205	Solo rearranque completo (en caliente) disponible
W#16#0206	Es posible reconfigurar en RUN para la periferia descentralizada mediante módulos de reserva configurados.
W#16#0208	Con atención a la funcionalidad de Motion Control
W#16#0300 - 03FF	Descripción de lenguaje MC7 de la CPU (grupo con índice 0300)
W#16#0301	Ocupado
W#16#0302	Todos los comandos con números en coma fija de 32 bits
W#16#0303	Todos los comandos con números en coma flotante
W#16#0304	sin, asin, cos, acos, tan, atan, sqr, sqrt, ln, exp
W#16#0305	Acu3/Acu4 con los comandos correspondientes (ENT, PUSH, POP, LEAVE)
W#16#0306	Comandos de Master Control Relay
W#16#0307	Registro de direcciones 1 presente con los comandos correspondientes

Identificador	Significado
W#16#0308	Registro de direcciones 2 presente con los comandos correspondientes
W#16#0309	Comandos para el direccionamiento de varias áreas
W#16#030A	Comandos para el direccionamiento interno de un área
W#16#030B	Todos los comandos de direccionamiento indirecto mediante M
W#16#030C	Todos los comandos de direccionamiento indirecto mediante DB
W#16#030D	Todos los comandos de direccionamiento indirecto mediante DI
W#16#030E	Todos los comandos de direccionamiento indirecto mediante LD
W#16#030F	Todos los comandos para la transferencia de parámetros en instrucciones
W#16#0310	Comandos de marcas de flanco mediante E
W#16#0311	Comandos de marcas de flanco mediante A
W#16#0312	Comandos de marcas de flanco mediante M
W#16#0313	Comandos de marcas de flanco mediante DB
W#16#0314	Comandos de marcas de flanco mediante DI
W#16#0315	Comandos de marcas de flanco mediante LD
W#16#0316	Evaluación dinámica del bit ERAB
W#16#0317	Área de datos local dinámica con los correspondientes comandos
W#16#0318	Ocupado
W#16#0319	Ocupado
W#16#0401	Ocupado
W#16#0402	Reservado
W#16#0403	"C_DIAG (Página 7027)" está disponible

ID de SZL W#16#xy13: áreas de memoria del usuario

Finalidad

La información sobre las áreas de memoria de este módulo se puede obtener a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy13.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy13 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0113: Registro para un área de memoria Para indicar el área de memoria debe usarse el parámetro INDEX.
INDEX	Indicar un área de memoria (solo con SZL-IDW#16#0113) W#16#0001: Memoria de trabajo
LENTHDR	W#16#0024: Un registro tiene una longitud de 18 palabras (36 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#xy13 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	Índice de un área de memoria W#16#0001: Memoria de trabajo
Code	1 palabra	Tipo de memoria: • W#16#0001: Memoria volátil (RAM) • W#16#0002: Memoria no volátil (FEPROM) • W#16#0003: Memoria mixta (RAM y FEPROM)
Size	2 palabras	Tamaño total de la memoria seleccionada (suma de Area1 y Area2)
Mode	1 palabra	Modo lógico de la memoria • Bit 0: Área de memoria volátil • Bit 1: Área de memoria no volátil • Bit 2: Área de memoria mixta Para memoria de trabajo: • Bit 3: Código y datos separados • Bit 4: Código y datos juntos
Granularity	1 palabra	Siempre ocupado con "0"
Area1	2 palabras	Tamaño en bytes del área de memoria volátil
Reserved1	2 palabras	Tamaño del área de memoria volátil ocupada
Block1	2 palabras	Mayor bloque libre en el área de memoria volátil Si es "0": No hay información o la información no se puede recuperar.
Area2	2 palabras	Tamaño en bytes del área de memoria no volátil
Reserved2	2 palabras	Tamaño del área de memoria no volátil ocupada
Block2	2 palabras	Mayor bloque libre en el área de memoria no volátil Si es "0": No hay información o la información no se puede recuperar.

ID de SZL W#16#xy14: áreas del sistema

Finalidad

La información sobre las áreas del sistema de este módulo se puede obtener a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy14.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy14 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0014: Todas las áreas del sistema de un módulo W#16#0F14: Solo información de encabezado de la lista parcial SZL
INDEX	Irrelevante
LENTHDR	W#16#0008: Un registro tiene una longitud de 4 palabras (8 bytes)
N_DR	Número de registros El número de registros debe parametrizarse por lo menos con 9. Si se elige un área de destino demasiado pequeña, "RDSYSST (Página 6963)" no devuelve ningún registro.

Registro

Un registro de la lista parcial con la SZL-IDW#16#xy14 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	Índice del área del sistema • W#16#0001: PAE (cantidad en bytes) • W#16#0002: PAA (cantidad en bytes) • W#16#0003: Marca (cantidad en bits) Nota: Este índice solo es devuelto por las CPUs en las que la cantidad de marcas puede representarse en una palabra. Si la CPU usada no devuelve este valor, debe evaluarse el índice W#16#0008 . • W#16#0004: Temporizadores (cantidad) • W#16#0005: Contadores (cantidad) • W#16#0006: Número de bytes en el área de direcciones lógica • W#16#0007: Datos locales (toda el área de datos local de la CPU, en bytes) Nota: Este índice solo es devuelto por las CPUs en las que la longitud de toda el área de datos local puede representarse en una palabra. Si la CPU usada no devuelve este valor, debe evaluarse el índice W#16#0009. • W#16#0008: Marcas (cantidad en bytes) • W#16#0009: Datos locales (toda el área de datos local de la CPU, en KB)
code	1 palabra	Tipo de memoria • W#16#0001: Memoria volátil (RAM) • W#16#0002: Memoria no volátil (FEPROM) • W#16#0003: Memoria mixta (RAM y FEPROM)
number	1 palabra	Número de elementos del área del sistema
retentive	1 palabra	Número de elementos remanentes

ID de SZL W#16#xy15: Tipos de bloque

Finalidad

Los tipos de bloque presentes en un módulo se pueden conocer a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy15.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy15 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0015: Registros de todos los tipos de bloque de un módulo
INDEX	Irrelevante
LENTHDR	W#16#000A: Un registro tiene una longitud de 5 palabras (10 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#xy15 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	Número de tipo de bloque • W#16#0800: OB • W#16#0A00: DB • W#16#0B00: SDB • W#16#0C00: FC • W#16#0E00: FB
MaxNumber	1 palabra	Número máximo de bloques del tipo • con OBs: Número máximo posible de OBs de una CPU • con DBs: Número máximo posible de DBs, incluido el DB0 • con SDBs: Número máximo posible de SDBs, incluido el SDB2 • con FCs y FBs: Número máximo posible de bloques cargables
MaxLenght	1 palabra	Tamaño total máximo en KB del objeto de carga
MaxWorkMem	2 palabras	Longitud máxima en bytes de la porción de memoria de trabajo en un bloque

ID de SZL W#16#xy1C: identificación de un componente

Finalidad

La CPU o el sistema de automatización se pueden identificar a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy1C.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy1C tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial
	• W#16#001C: Identificación de todos los componentes • W#16#011C: Identificación de un componente • W#16#0F1C: Solo información de encabezado de la lista parcial SZL
INDEX	• Identificación del componente para el extracto de lista parcial con la SZL-IDW#16#011C y W#16#031C – W#16#0001: Nombre del sistema de automatización – W#16#0002: Nombre del módulo – W#16#0003: Identificador de la instalación del módulo – W#16#0004: Entrada de derechos de autor – W#16#0005: Número de serie del módulo – W#16#0007: Nombre del tipo de módulo – W#16#0008: Número de serie de la Memory Card En los módulos que no permiten insertar una Memory Card, no se devuelve ningún registro. – W#16#0009: Fabricante y perfil de un módulo de CPU – W#16#000A: Identificador OEM de un módulo (solo S7-300) – W#16#000B: Identificador de la situación de un módulo – W#16#000C: número de serie del módulo Sync 1 – W#16#000D: número de serie del módulo Sync 2 • Número de rack correspondiente al extracto de lista parcial con la SZL-IDW#16#021C (byte 0: Número de rack, byte 1: B#16#00)
LENTHDR	W#16#0022: Un registro tiene una longitud de 17 palabras (34 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#xy1C tiene la siguiente estructura:

INDEX = W#16#0001

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0001 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#01
name	12 palabras	Nombre del sistema de automatización (máximo 24 caracteres; si se utilizan menos, se rellena el resto con B#16#00)
res	4 palabras	Reservado

INDEX = W#16#0002

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0002 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#02
name	12 palabras	Nombre del módulo (máximo 24 caracteres; si se utilizan menos, se rellena el resto con B#16#00)
res	4 palabras	Reservado

INDEX = W#16#0003

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0003 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#03
tag	16 palabras	Identificador de la instalación del módulo (máximo 32 caracteres; si se utilizan menos, se rellena el resto con B#16#00)

INDEX = W#16#0004

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0004 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#04
copyright	13 palabras	Cadena de caracteres constante "Original Siemens Equipment"
res	3 palabras	Reservado

INDEX = W#16#0005

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0005 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#05
serialn	12 palabras	Número de serie del módulo (cadena de un máximo de 24 caracteres. Si se utilizan menos, se rellena el resto con B#16#00). Nota: El número de serie es unívoco en todo el mundo para componentes SIMATIC. Está vinculado al hardware de la CPU, es decir, se mantiene aun en caso de que se actualice el firmware.
res	4 palabras	Reservado

INDEX = W#16#0007

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0007 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#07
cputypname	16 palabras	Nombre del tipo de módulo (cadena de un máximo de 32 caracteres. Si se utilizan menos, se rellena el resto con B#16#00).

INDEX = W#16#0008

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0008 • Con los extractos de lista parcial W#16#011C, W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#08
sn_mc/mmc	16 palabras	Número de serie de la Memory Card/Micro Memory Card (cadena de un máximo de 32 caracteres). Si se utilizan menos, se rellena el resto con B#16#00). • Para números de serie Siemens: Número de serie sin complementos • Para números de serie de producto (PSN) de una Micro Memory Card de S7: "MMC" más número de serie (PSN) • Para números de serie de una Memory Card de S7: "MC" más número de serie Si no hay ninguna Memory Card insertada, la cadena de caracteres finaliza justo después de "MMC" o "MC".

INDEX = W#16#0009

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#0009 • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#09
manufacturer_id	1 palabra	Ver PROFIBUS Profile Guidelines Part 1, Identification & Maintenance Functions
profile_id	1 palabra	Ver PROFIBUS Profile Guidelines Part 1, Identification & Maintenance Functions
profile_specific_typ	1 palabra	Ver PROFIBUS Profile Guidelines Part 1, Identification & Maintenance Functions
res	13 palabras	Reservado

INDEX = W#16#000A

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#000A • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#0A
oem_copyright_string	13 palabras	Identificador de copyright OEM (cadena de un máximo de 20 caracteres. Si se utilizan menos, se rellena el resto con B#16#00).
oem_id	1 palabra	ID de OEM. Asignada por Siemens.
oem_add_id	2 palabras	Identificador adicional de OEM. Definible por el usuario.

INDEX = W#16#000B

Nombre	Longitud	Significado
index	1 palabra	• Con una CPU estándar y el extracto de lista parcial W#16#011C: Identificador del componente: W#16#000B • Con los extractos de lista parcial W#16#021C y W#16#031C: byte 0: – Bits de 0 a 2: Número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 byte 1: Identificador del componente: B#16#0B
loc_id	16 palabras	Identificador de situación (cadena de un máximo de 32 caracteres. Si se utilizan menos, se rellena el resto con B#16#00).

INDEX = W#16#000C

Nombre	Longitud	Significado
index	1 palabra	• Byte0: – Bits de 0 a 2: número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 • Byte 1: identificador del componente: B#16#0C
mlfb	5 palabras	Del carácter 10.º al 19.º de la referencia del módulo Sync 1
fill	1 palabra	Cuatro espacios (ASCII)
es	1 palabra	Versión (ASCII), p. ej. "01"
fill	1 byte	Dos espacios (ASCII)
vendor_sn	17 bytes	Número de serie

Nota

Si no existe el módulo Sync 1 o no se puede identificar, el registro se rellena según la variable "index" con B#16#00.

INDEX = W#16#000D

Nombre	Longitud	Significado
index	1 palabra	• Byte0: – Bits de 0 a 2: número de rack; – Bit 3: 0 = CPU de reserva, 1 = CPU maestra; – Bits de 4 a 7: 1111 • Byte 1: identificador del componente: B#16#0D
mlfb	5 palabras	Del carácter 10.º al 19.º de la referencia del módulo Sync 2
fill	1 palabra	Cuatro espacios (ASCII)
es	1 palabra	Versión (ASCII), p. ej. "01"
fill	1 byte	Dos espacios (ASCII)
vendor_sn	17 bytes	Número de serie

Nota

Si no existe el módulo Sync 2 o no se puede identificar, el registro se rellena según la variable "index" con B#16#00.

ID de SZL W#16#xy22: estado de alarma

Finalidad

El estado actual de procesamiento de alarma y generación de alarmas se puede conocer a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy22.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy22 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0222 Registro para la alarma especificada. Debe indicarse la alarma (n.º de OB) en el parámetro INDEX.
INDEX	N.º de OB o clase de alarma (para SZL-IDW#16#0222) W#16#0000: Ciclo libre W#16#000A: Alarma horaria W#16#0014: Alarma de retardo W#16#001E: Alarma cíclica W#16#0028: Alarma de proceso W#16#0032: Alarma DP W#16#003C: Alarma de sincronismo W#16#0050: Alarmas de error asíncronas W#16#005A: Segundo plano W#16#0064: Arranque W#16#0078: Alarmas de error sincrónicas
LENTHDR	W#16#001C: Un registro tiene una longitud de 14 palabras (28 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#xy22 tiene la siguiente estructura:

Nombre	Longitud	Significado
Info	10 palabras	Información de arranque del OB correspondiente, con las siguientes excepciones: • En el OB 1 se obtienen el tiempo de ciclo actual mínimo (en los bytes 8 y 9) y máximo (en los bytes 10 y 11) (base de tiempo: ms, recuento de bytes a partir de 0). • Mientras está activa una petición para una alarma de retardo, los bytes de 8 a 11 (recuento de bytes a partir de 0) contienen el tiempo restante en ms respecto al retardo parametrizado. • En el OB 80 se pueden leer el tiempo de ciclo configurado mínimo (en los bytes 8 y 9) y máximo (en los bytes 10 y 11) (base de tiempo: ms, recuento de bytes a partir de 0). • Para alarmas de error sin información actual • En las alarmas, la información de estado contiene la parametrización actual de origen de la alarma. • En los errores sincrónicos, se registra como clase de prioridad B#16#7F si los OBs todavía no se han procesado; de lo contrario se registra la clase de prioridad de la última llamada. • Si un OB tiene varios eventos de arranque y estos no se han producido todavía en el momento de recabar la información, se devuelve como número de evento W#16#xyzz con: – x: Clase de evento – zz: Número menor definido del grupo – y: Sin definir. De lo contrario, se utiliza el número del último evento de arranque que se ha producido.
AlarmStatus1	1 palabra	Identificadores de procesamiento: • Bit 0: Estado del evento de alarma fijado mediante parametrización: – = 0: Habilitado – = 1: Bloqueado • Bit 1: Estado del evento de alarma fijado mediante "DIS_IRT (Página 6906)": – = 0: No bloqueado – = 1: Bloqueado • Bit 2 = 1: El origen de la alarma está activo (en el caso de las alarmas de tiempo, la petición de generación está vigente (OB de alarma horaria iniciado), OB de alarma de retardo iniciado, OB de alarma cíclica configurado) • Bit 4: El OB de alarma – = 0: no está cargado – = 1: está cargado • Bit 5: Estado del OB de alarma fijado mediante TIS: – = 1: Bloqueado • Bit 6: Entrada en el búfer de diagnóstico – = 1: Bloqueado

Nombre	Longitud	Significado
AlarmStatus2	1 palabra	Reacción si el OB no está cargado o está bloqueado • Bit 0 = 1: Bloquear origen de alarma • Bit 1 = 1: Generar error de evento de alarma • Bit 2 = 1: La CPU pasa al estado operativo STOP • Bit 3 = 1: La alarma solo se descarta
AlarmStatus3	2 palabras	Descartar mediante funciones TIS: N.º de bit x activado significa: El número de evento que excede en x el menor número de evento del OB en cuestión está descartado mediante función TIS.

ID de SZL W#16#xy25: correspondencia entre memorias imagen parcial de proceso y OBs

Finalidad

La correspondencia entre memorias imagen parcial de proceso y OBs se puede conocer a través de la lista parcial con la SZL-ID (ID de lista de estado del sistema) W#16#xy25.

La lista parcial informa acerca de

- Memorias imagen parcial de proceso que se han asignado a OBs individuales para su actualización por el sistema
- Memorias imagen parcial de proceso que se han asignado a OBs de alarma de sincronismo individuales (OBs 61 a 64) En este caso, la actualización de memorias imagen parcial de proceso se realiza llamando las instrucciones "SYNC_PI (Página 6732)" y "SYNC_PO (Página 6734)".
  La correspondencia entre los sistemas maestros DP y los OBs de alarma de sincronismo se puede conocer a través de SZL W#16#xy95 (Página 7012).

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy25 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial • W#16#0025: Correspondencia entre todas las memorias imagen parcial de proceso y OBs dentro de la CPU • W#16#0125: Correspondencia entre una memoria imagen parcial de proceso y el correspondiente OB El número de memoria imagen parcial de proceso se especifica en el parámetro INDEX. • W#16#0225: Correspondencia entre un OB y las correspondientes memorias imagen parcial de proceso El número de OB se especifica en el parámetro INDEX. Nota: Solo es posible asignar varias memorias imagen parcial de proceso a los OBs de alarma de sincronismo (OBs 61 a 64). • W#16#0F25: Solo información de encabezado de la lista parcial SZL
INDEX	• Para SZL-ID W#16#0025: Irrelevante • Para SZL-ID W#16#0125: N.º de memoria imagen parcial de proceso • Para SZL-ID W#16#0225: N.º de OB • Para SZL-ID W#16#0F25: Irrelevante
LENTHDR	W#16#0004: Un registro tiene una longitud de 2 palabras (4 bytes)
N_DR	Número de registros

Registro

Un registro del extracto de lista parcial con la SZL-IDW#16#xy25 tiene la siguiente estructura:

Nombre	Longitud	Significado
PPI*_nr	1 bytes	N.º de memoria imagen parcial de proceso
PPI*_use	1 bytes	Tipo de correspondencia entre memoria imagen parcial de proceso y OB: • Bit 0 = 1: La memoria imagen parcial de proceso de entrada está asignada al OB especificado para su actualización por el sistema. • Bit 1 = 1: La memoria imagen parcial de proceso de salida está asignada al OB especificado para su actualización por el sistema. • Bit 2 = 1: La memoria imagen parcial de proceso de entrada está asignada al OB de alarma de sincronismo especificado y puede actualizarse llamando "SYNC_PI (Página 6732)" en ese OB. • Bit 3 = 1: La memoria imagen parcial de proceso de salida está asignada al OB de alarma de sincronismo especificado y puede actualizarse llamando "SYNC_PO (Página 6734)" en ese OB. • Bits de 4 a 7: 0
ob_nr	1 bytes	N.º de OB
res	1 bytes	Reservado
*PPI: Parial Process Image		

Extractos de lista parcial

- Extracto de lista parcial con SZL-ID = W#16#0025:
  Se devuelven en orden ascendente los registros de todas las memorias imagen parcial de proceso que se han asignado a un OB en la configuración. Para las memorias imagen parcial de proceso no asignadas a ningún OB, ob_nr tiene valor cero. No se devuelve ningún registro para memorias imagen parcial de proceso.
- Extracto de lista parcial con SZL-ID = W#16#0125:
  Si en la configuración se ha asignado un OB a la memoria imagen parcial de proceso a la que se accede, se devuelve un registro. Si no se ha asignado ningún OB, no se devuelve ningún registro.

  Nota

  La memoria imagen parcial de proceso 0 tiene asignado de modo predeterminado el OB 1. En consecuencia, al solicitar información acerca de la memoria imagen parcial de proceso 0, siempre se obtiene un registro.

- Extracto de lista parcial con SZL-ID = W#16#0225:
  Para cada memoria imagen parcial de proceso asignada al OB al que se accede, se devuelve un registro. Si en la configuración no se ha asignado al OB al que se accede ninguna memoria imagen parcial de proceso, no se devuelve ningún registro.

  Nota

  Los OBs de alarma de sincronismo pueden tener asignadas varias memorias imagen parcial de proceso. En ese caso se devuelven varios registros.

- Extracto de lista parcial SZL-ID = W#16#0F25:
  Se devuelve el número máximo posible de registros.

Ejemplos de significado de los registros

Parámetros de llamada de "RDSYSST (Página 6963)"	Variables devueltas	Explicación
SZL_ID = W#16#0125, INDEX = W#16#0008	PPI*_nr = B#16#08, PPI*_use = B#16#03, ob_nr = B#16#15	Se devuelve un registro. Las memorias imagen de proceso de entrada y salida 8 están asignadas al OB 21 para su actualización por el sistema.
SZL_ID = W#16#0125, INDEX = W#16#0009	–	No se devuelve ningún registro. Por lo tanto: La memoria imagen parcial de proceso 9 no está asignada a ningún OB.
SZL_ID = W#16#0225, INDEX = W#16#003D	PPI*_nr = B#16#0A, PPI*_use = B#16#C0, ob_nr = B#16#3D PPI*_nr = B#16#10, PPI*_use = B#16#C0, ob_nr = B#16#3D	Se devuelven dos registros. El OB 61 tiene asignadas las memorias imagen de proceso de entrada y salida 10 y 16. Estas pueden actualizarse en el OB 61 llamando "SYNC_PI (Página 6732)" y "SYNC_PO (Página 6734)".

Parámetros de llamada de "RDSYSST (Página 6963)"	Variables devueltas	Explicación
SZL_ID = W#16#0225, INDEX = W#16#0001	PPI*_nr = B#16#00, PPI*_use = B#16#03, ob_nr = B#16#01	Se devuelve un registro. El OB 1 tiene asignadas las memorias imagen de proceso de entrada y salida 0. Estas son actualizadas por el sistema.
*PPI: Parial Process Image		

ID de SZL W#16#xy32: Datos de estado de comunicación

ID de SZL W#16#xy32: Datos de estado de comunicación

Finalidad

Los datos de estado de comunicación del módulo se pueden obtener a través de la lista parcial con la SZL-IDW#16#xy32.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy32 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial • W#16#0132: Datos de estado de un componente de comunicación (siempre solo un registro). Para indicar el componente de comunicación debe usarse el parámetro INDEX. • W#16#0232: Datos de estado para un componente de comunicación. Para indicar el componente de comunicación debe usarse el parámetro INDEX.
INDEX	Componente de comunicación • Para SZL-IDW#16#0132: – W#16#0005 Diagnóstico – W#16#0008 Sistema horario – W#16#000B Sistema horario – W#16#000C Sistema horario • Para SZL-IDW#16#0232: W#16#0004 Nivel de protección de la CPU, posiciones del interruptor de manejo e identificadores de versión/sumas de verificación
LENTHDR	W#16#0028: Un registro tiene una longitud de 20 palabras (40 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#0132 tiene siempre una longitud de 20 palabras. Los registros están ocupados de diferentes maneras. La ocupación depende del parámetro INDEX, es decir, de a qué componente de comunicación corresponda el registro.

Registro del extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#0005

Contenido

El extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#0005 contiene información acerca del estado de diagnóstico del módulo.

Registro

Un registro del extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#0005 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	W#16#0005: Diagnóstico
ext	1 palabra	Funcionalidad avanzada • 0: No • 1: Sí
send	1 palabra	Transmisión automática • 0: No • 1: Sí
possible	1 palabra	La transmisión de avisos de diagnóstico del usuario es posible en este momento • 0: No • 1: Sí
res	16 palabras	Reservado

Registro del extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#0008

Contenido

El extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#0008 contiene información acerca del estado del sistema horario del módulo.

Registro

Un registro del extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#0008 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	W#16#0008: Estado del sistema horario
cycle	1 palabra	Reservado
corr	1 palabra	Factor de corrección de la hora
clock 0	1 palabra	Contador de horas de funcionamiento 0: Tiempo en horas
clock 1	1 palabra	Contador de horas de funcionamiento 1: Tiempo en horas
clock 2	1 palabra	Contador de horas de funcionamiento 2: Tiempo en horas

Nombre	Longitud	Significado
clock 3	1 palabra	Contador de horas de funcionamiento 3: Tiempo en horas
clock 4	1 palabra	Contador de horas de funcionamiento 4: Tiempo en horas
clock 5	1 palabra	Contador de horas de funcionamiento 5: Tiempo en horas
clock 6	1 palabra	Contador de horas de funcionamiento 6: Tiempo en horas
clock 7	1 palabra	Contador de horas de funcionamiento 7: Tiempo en horas
time	4 palabras	Fecha y hora actuales (formato: date_and_time)
rtm _0	1 bytes	Runtime meter Bit x: Contador de horas de funcionamiento x, 0 ≤ x ≤ 7 (bit = 1: Contador de horas de funcionamiento en marcha)
res	1 bytes	Reservado
rtmOV _0	1 bytes	Runtime meter Overflow Bit x: Desbordamiento del contador de horas de funcionamiento x, 0 ≤ x ≤ 7 (bit = 1: Desbordamiento)
res	1 bytes	Reservado
status	1 palabra	Estado de la hora (para la asignación de bits, ver abajo)
res	3 bytes	Reservado
status_valid	1 bytes	Validez de la variable status: B#16#01: status válida

Estado de la hora (status)

Bit	Valor predeterminado	Descripción
15	0	Signo del valor de corrección (0: positivo, 1: negativo)
de 14 a 10	00000	Valor de corrección Este parámetro permite corregir la hora del módulo obtenida mediante telegrama, p. ej. para ajustarla a la hora local: `Hora local = Hora del módulo ± Valor de corrección * 0,5 h` Esta corrección tiene en cuenta la zona horaria y la diferencia entre el horario de verano y el horario de invierno
9	0	Reservado
8	0	Reservado
7	0	Hora de aviso Este parámetro indica si en el próximo cambio de hora se produce el cambio de horario de verano a horario de invierno o al revés. (0: no se produce, 1: se produce)
6	0	Indicador de horario de verano/invierno Este parámetro indica si la hora local calculada mediante el valor de corrección corresponde al horario de verano o al de invierno. (0: horario de invierno, 1: horario de verano)
5	0	Este parámetro no se utiliza en S7.
de 4 a 3	00	Resolución de hora Este parámetro refleja la exactitud de la hora transferida. (00: 0,001 s, 01: 0,01 s, 10: 0,1 s, 11: 1 s)

Bit	Valor predeterminado	Descripción
2	0	Este parámetro no se utiliza en S7.
1	0	Este parámetro no se utiliza en S7.
0	0	Fallo de sincronización Este parámetro indica si la hora transferida en el telegrama está sincronizada por un reloj maestro externo, p. ej. SICLOCK. (0: la sincronización no se ha producido, 1: la sincronización se ha producido) **Nota:** Solo tiene sentido evaluar este bit en una CPU si existe sincronización horaria externa continua.

Registro del extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#000B

Contenido

El extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#000B contiene información acerca del estado de los contadores de horas de funcionamiento de 32 bits de 0 a 7 del módulo.

Nota

En el extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#0008, estos contadores de horas de funcionamiento aparecen como contadores de horas de funcionamiento de 16 bits.
Esto permite seguir utilizando los programas que han sido desarrollados para una CPU con contadores de horas de funcionamiento de 16 bits, y que usan el extracto de lista parcial con la SZL-IDW#16#0132 y el IndexW#16#0008.

Registro

Un registro del extracto de lista parcial con la SZL-IDW#16#0132 y el índice W#16#000B tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	W#16#000B: Estado del sistema horario
rtm _0	1 bytes	Runtime meter Bit x: Estado del contador de horas de funcionamiento x , 0 ≤ x ≤ 7 (bit = 1: Contador de horas de funcionamiento en marcha)
res	1 bytes	Reservado
rtmOV _0	1 bytes	Runtime meter overflow Bit x: Desbordamiento del contador de horas de funcionamiento x, 0 ≤ x ≤ 7 (bit = 1: Desbordamiento)
res	1 bytes	Reservado

Nombre	Longitud	Significado
clock 0	2 palabras	Contador de horas de funcionamiento 0: Tiempo en horas
clock 1	2 palabras	Contador de horas de funcionamiento 1: Tiempo en horas
clock 2	2 palabras	Contador de horas de funcionamiento 2: Tiempo en horas
clock 3	2 palabras	Contador de horas de funcionamiento 3: Tiempo en horas
clock 4	2 palabras	Contador de horas de funcionamiento 4: Tiempo en horas
clock 5	2 palabras	Contador de horas de funcionamiento 5: Tiempo en horas
clock 6	2 palabras	Contador de horas de funcionamiento 6: Tiempo en horas
clock 7	2 palabras	Contador de horas de funcionamiento 7: Tiempo en horas
res	1 palabra	Reservado

Registro del extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#000C

Contenido

El extracto de lista parcial con la SZL-IDW#16#0132 y el INDEXW#16#000C contiene información acerca del estado de los contadores de horas de funcionamiento de 32 bits de 8 a 15 del módulo.

Registro

Un registro del extracto de lista parcial con la SZL-IDW#16#0132 y el INDEXW#16#000C tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	W#16#000C: Estado del sistema horario
rtm_0	1 bytes	Runtime meter Bit x: Estado del contador de horas de funcionamiento (8+x), 0 ≤ x ≤ 7 (bit = 1: Contador de horas de funcionamiento en marcha)
res	1 bytes	Reservado
rtmOV _0	1 bytes	Runtime meter overflow Bit x: Desbordamiento del contador de horas de funcionamiento (8+x), 0 ≤ x ≤ 7 (bit = 1: Desbordamiento)
res	1 bytes	Reservado
clock 8	2 palabras	Contador de horas de funcionamiento 8: Tiempo en horas
clock 9	2 palabras	Contador de horas de funcionamiento 9: Tiempo en horas

Nombre	Longitud	Significado
clock 10	2 palabras	Contador de horas de funcionamiento 10: Tiempo en horas
clock 11	2 palabras	Contador de horas de funcionamiento 11: Tiempo en horas
clock 12	2 palabras	Contador de horas de funcionamiento 12: Tiempo en horas
clock 13	2 palabras	Contador de horas de funcionamiento 13: Tiempo en horas
clock 14	2 palabras	Contador de horas de funcionamiento 14: Tiempo en horas
clock 15	2 palabras	Contador de horas de funcionamiento 15: Tiempo en horas
res	1 palabra	Reservado

Registro del extracto de lista parcial con la SZL-ID W#16#0232 Índice W#16#0004

Contenido

El extracto de lista parcial con la SZL-IDW#16#0232 y el Index W#16#0004 contiene información acerca del nivel de protección de la CPU, las posiciones del selector de modo y las sumas de verificación de la configuración de hardware y del programa de usuario.

Registro

Un registro del extracto de lista parcial con la SZL-IDW#16#0232 y el Index W#16#0004 tiene la siguiente estructura:

Nombre	Longitud	Significado
Index	1 palabra	• Byte 1: B#16#04: Nivel de protección de la CPU, posiciones del selector de modo y sumas de verificación • Byte 0: CPU estándar: B#16#00 Bit 3: 0 = CPU de reserva, 1 = CPU maestra Bits de 4 a 7: 1111
ProtLevel_ModeSwitch	1 palabra	Nivel de protección ajustado mediante selector de modo (1, 2, 3)
ProtLevel_Par	1 palabra	Nivel de protección parametrizado (0, 1, 2, 3; 0: No hay ninguna contraseña asignada, nivel de protección no válido).
ProtLevel_Valid	1 palabra	Nivel de protección válido de la CPU
ModeSwitch	1 palabra	Posición del selector de modo (1:RUN, 2:RUN-P, 3:STOP, 4:MRES, 0:no definida o desconocida)
Startup_ModeSwitch	1 palabra	Posición del selector de tipo de arranque (1:CRST, 2:WRST, 0:no definida, no existente o desconocida)
res	1 palabra	Reservado
ID_Val_CheckSum	1 palabra	Identificador de la validez de las cuatro sumas de verificación siguientes (0: no válido)

Nombre	Longitud	Significado
ID_CheckSum1_HWConf	1 palabra	Suma de verificación 1 de la configuración de hardware (formato Intel): Operación lógica O-exclusiva con las longitudes de todos los bloques de datos del sistema
ID_CheckSum2_HWConf	1 palabra	Suma de verificación 2 de la configuración de hardware (formato Intel): Operación lógica O-exclusiva con las sumas de verificación de todos los bloques de datos del sistema
ID_CheckSum1_UserProg	1 palabra	Suma de verificación 1 del programa de usuario (formato Intel): Operación lógica O-exclusiva con las longitudes de los siguientes bloques: OBs, DBs, FBs, FCs
ID_CheckSum2_UserProg	1 palabra	Suma de verificación 2 del programa de usuario (formato Intel): Operación lógica O-exclusiva con las sumas de verificación de los siguientes bloques: OBs, DBs, FBs, FCs
res	2 palabras	Reservado
sfc_req	1 palabra	Petición de nivel de protección 2 o 3 realizada por la instrucción "PROTECT" (1: petición realizada)
sfc_act	1 palabra	Activación del nivel de protección 2 o 3 realizada por la instrucción "PROTECT" (1: activación realizada)
res	4 palabras	Reservado

SZL-ID W#16#xy74: estado de los LEDs de módulo

Finalidad

El estado de los LEDs de módulo se puede conocer a través de la lista parcial con la SZL-ID W#16#xy74 en las CPU estándar (si existe).

Encabezado

El encabezado de la lista parcial con la SZL-ID W#16#xy74 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0174 Estado de un LED. Para seleccionar el LED debe usarse el parámetro INDEX.
INDEX	Identificador de LED (relevante solo para SZL-IDW#16#0174) • W#16#0001: SF (error de grupo) • W#16#0002: INTF (error interno) • W#16#0003: EXTF (error externo) • W#16#0004: RUN • W#16#0005: STOP • W#16#0006: FRCE (forzar permanentemente) • W#16#0007: CRST (rearranque completo) • W#16#0008: BAF (error de batería/sobrecarga, cortocircuito de la tensión de batería en el bus) • W#16#0009: USR (personalizado) • W#16#000A: USR1 (personalizado) • W#16#000B: BUS1F (error de bus interfaz 1) • W#16#000C: BUS2F (error de bus interfaz 2) o BUS5F (error de bus interfaz 5) en las CPUs 414-3 PN/DP, 416-3 PN/DP y 416F-3 PN/DP • W#16#0012: IFM1F (error de interfaz, módulo de interfaz 1) • W#16#0013: IFM2F (error de interfaz, módulo de interfaz 2) • W#16#0014: BUS3F (error de bus interfaz 3) • W#16#0015: MAINT (mantenimiento solicitado) • W#16#0016: DC 24 V • W#16#0017: BUS5F (error de bus interfaz 5) • W#16#0080: IF (fallo de inicialización) • W#16#0081: UF (fallo de usuario) • W#16#0082: MF (fallo de monitorización) • W#16#0083: CF (fallo de comunicación) • W#16#0084: TF (fallo de tarea) • W#16#00EC: APPL_STATE_RED • W#16#00ED: APPL_STATE_GREEN
LENTHDR	W#16#0004: Un registro tiene una longitud de 2 palabras (4 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-ID W#16#xy74 tiene la siguiente estructura:

Nombre	Longitud	Significado
CPU_LED_ID	1 palabra	• Byte 0: – CPU estándar: B#16#00 – Bit 3: 0=CPU de reserva, 1=CPU maestra – Bits de 4 a 7: 1111 • Byte 1: Identificador de LED
LED_on	1 byte	Estado del LED: • 0: Off • 1: On
LED_flash	1 byte	Estado de parpadeo del LED: • 0: Sin parpadeo • 1: Parpadeo normal (2 Hz) • 2: Parpadeo lento (0,5 Hz)

SZL-ID W#16#xy90: información del sistema maestro DP

Finalidad

La información de estado de todos los sistemas maestros DP conocidos por la CPU se puede obtener a través de la lista parcial con la SZL-IDW#16#xy90.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy90 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial W#16#0090: Información acerca de todos los sistemas maestros DP conocidos por la CPU W#16#0190: Información acerca de un sistema maestro DP W#16#0F90: Solo información de encabezado de lista parcial
INDEX	• Para el extracto de lista parcial con la SZL-IDW#16#0190: – byte menos significativo: B#16#00 – byte más significativo: ID del sistema maestro DP • Para los extractos de lista parcial con las SZL-IDW#16#0090 y W#16#0F90: – W#16#0000
LENTHDR	W#16#000E: Un registro tiene una longitud de 7 palabras (14 bytes)
N_DR	Número de registros Para el extracto de lista parcial con la SZL-IDW#16#0190: de 0 a 1 Para el extracto de lista parcial con la SZL-IDW#16#0090: • Con una CPU estándar: de 0 a 14

Registro

Un registro de la lista parcial con la IDW#16#xy90 tiene la siguiente estructura:

Nombre	Longitud	Significado	
dp_m_id	1 bytes	ID del sistema maestro DP	
dp_m_rack	1 bytes	Número de rack del maestro DP • Con una CPU estándar: 0	
dp_m_slot	1 bytes	Slot del maestro DP o slot de la CPU (con interfaz DP integrada)	
dp_m_SubModule	1 bytes	• Con interfaz DP integrada: Número de interfaz del maestro DP: – 1: X2 – 2: X1 – 3: IF1 – 4: IF2 • Con interfaz DP externa: 0	
logadr	1 palabra	Dirección lógica inicial del maestro DP	
res	1 palabra	Reservado	
res	1 palabra	Reservado	
dp_m_state	1 bytes	Otras características del sistema maestro DP	
		Bit 0:	Modo DP • 0: Compatible con S7 • 1: DPV1
		Bit 1:	Ciclo DP • 0: No equidistante • 1: Equidistante
		Bits de 2 a 6:	• Reservado
		Bit 7:	Tipo de maestro DP • 0: Maestro DP integrado • 1: Maestro DP externo
res	3 bytes	Reservado	

SZL-ID W#16#xy91: información de estado de módulo

Finalidad

La información de estado de todos los módulos asignados a la CPU se puede obtener a través de la lista parcial con la SZL-ID W#16#xy91.

Nota

No es posible evaluar la SZL-ID W#16#xy91 para un módulo con direcciones comprimidas (ET 200S).

Encabezado

El encabezado de la lista parcial con la SZL-ID W#16#xy91 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial • W#16#0091 Información de estado de todos los módulos/submódulos insertados y configurados de la CPU (solo S7-400) • W#16#0191 Información de estado de todos los módulos/racks no desactivados con ID de módulo errónea (solo S7-400) • W#16#0291 Información de estado de todos los módulos en estado de fallo y no desactivados (solo S7-400) • W#16#0391 Información de estado de todos los módulos no disponibles (solo S7-400) • W#16#0591 Información de estado de módulo de todos los submódulos del módulo host • W#16#0991 Información de estado de módulo de un sistema maestro DP • W#16#0A91 Información de estado de todos los subsistemas DP y sistemas maestros DP (solo S7-300 sin CPU 318-2 DP), así como sistemas PROFINET IO • W#16#0C91 Información de estado de un módulo en configuración centralizada o conectado a una interfaz DP integrada o a una interfaz PROFINET a través de la dirección base lógica • W#16#4C91 Información de estado de un módulo conectado a una interfaz DP externa a través de la dirección base lógica Si se usan más de 4 interfaces DP externas, puede producirse erróneamente un RET_VAL W#16#80A4. • W#16#0D91 Información de estado de todos los módulos del rack o la estación indicados (DP o PROFINET) • W#16#0E91 Información de estado de todos los módulos asignados • W#16#0F91 Solo información de encabezado de lista parcial SZL
INDEX	• Para el extracto de lista parcial con la SZL-ID W#16#0C91: – S7-400: Bits de 0 a 14: Dirección base lógica del módulo, Bit 15: 0 = entrada, 1 = salida – S7-300: Bits de 0 a 14: Cualquier dirección lógica del módulo, Bit 15: 0 = entrada, 1 = salida • Para el extracto de lista parcial con la SZL-ID W#16#4C91 (solo S7-400): – Bits de 0 a 14: Dirección base lógica del módulo, – Bit 15: 0 = entrada/1 = salida • Para el extracto de lista parcial con las IDs de SZL W#16#0091, W#16#0191,W#16#0291, W#16#0391, W#16#0491, W#16#0591, W#16#0A91, W#16#0E91, W#16#0F91: INDEX es irrelevante, todos los módulos (en rack y en periferia descentralizada) • Para el extracto de lista parcial con la SZL-ID W#16#0991 (solo S7-400): W#16#xx00: todos los módulos de un sistema maestro DP (xx contiene la ID de sistema maestro DP) • Para el extracto de lista parcial con la SZL-ID W#16#0D91 – W#16#00xx: todos los módulos y submódulos de un rack (xx contiene el número del rack) – W#16#xxyy: todos los módulos de una estación DP o todos los dispositivos IO de una estación PROFINET IO (PROFIBUS DP: xx contiene la ID del sistema maestro DP, yy el número de estación; PROFINET IO: bits de 0 a 10: número de estación, bits de 11 a 14: las dos últimas posiciones de la ID del subsistema PNIO, bit 15: 1 (ver abajo tercera figura con adr1)

Contenido	Significado
LENTHDR	W#16#0010: Un registro tiene una longitud de 8 palabras (16 bytes)
N_DR	Número de registros; en función del producto, el número de registros transferidos a la instrucción "RDSYSST (Página 6963)" puede ser menor

Con W#16#0091, W#16#0191 y W#16#0F91 se devuelven 2 registros adicionales por rack:

- Un registro para la fuente de alimentación, si la hay y está configurada, y
- un registro para el rack.

El orden de los registros en caso de configuración centralizada es el siguiente: PS, slot 1, slot 2,..., slot 18, rack.

Un registro de la lista parcial con la ID W#16#xy91 tiene la siguiente estructura:

Nombre	Longitud	Significado
adr1	1 palabra	• Con configuración centralizada: Número de rack • En caso de configuración descentralizada con PROFIBUS DP: ID del sistema maestro DP, número de estación • En caso de configuración descentralizada con PROFINET IO: Bit 15 = 1 (identificador de PROFINET IO), las dos últimas posiciones de la ID del sistema PROFINET IO, número de estación Nota: Toda interfaz PROFINET tiene la consideración de "submódulo en configuración centralizada", independientemente de su uso para PROFINET IO.
adr2	1 palabra	• En caso de configuración centralizada o descentralizada con PROFIBUS DP: Número de slot y número de slot del submódulo • En caso de configuración descentralizada con PROFINET IO: Número de slot Nota: Toda interfaz PROFINET tiene la consideración de "submódulo en configuración centralizada", independientemente de su uso para PROFINET IO.
LogicalAddr	1 palabra	Primera dirección lógica E/S asignada (dirección base)
SetType	1 palabra	PROFINET IO: Tipo teórico (ver abajo); de lo contrario, reservado
ActualType	1 palabra	PROFINET IO: Tipo real (ver abajo); de lo contrario, reservado
res	1 palabra	00xx = N.º de CPU 1-4 (solo S7-400) Con PROFINET IO: • SZL-ID=W#16#0C91: Número de submódulos realmente existentes (sin contar el submódulo 0) • SZL-ID=W#16#0D91: Número de submódulos (sin contar el submódulo 0) • SZL-ID=W#16#4C91: Número de submódulos realmente existentes (sin contar el submódulo 0) • SZL-ID=W#16#4D91: Número de submódulos realmente existentes (sin contar el submódulo 0)

Nombre	Longitud	Significado
IOStat	1 palabra	Estado E/S • Bit 0 = 1: Fallo del módulo (detectado mediante alarma de diagnóstico) • Bit 1 = 1: Módulo existente • Bit 2 = 1: Módulo no disponible • Bit 3 = 1: Módulo desactivado • Bit 4 = 1: Fallo de la estación (solo slot de representante) • Bit 5 = 1: Para este módulo o estación hay un proceso CiR activo actualmente, o todavía no finalizado • Bit 6 = 1: Reservado para S7-400 • Bit 7 = 1: Módulo en el segmento de bus local • Bits de 8 a 15: Identificador de datos para dirección lógica (entrada: B#16#B4, salida: B#16#B5, interfaz DP integrada: B#16#FE, interfaz DP externa: B#16#FF)
AreaID_ModuleWidth	1 palabra	Identificación de área/ancho de módulo • bits de 0 a 2: Ancho de módulo • Bit 3: Reservado • bits de 4 a 6: Identificación de área – 0 = S7-400 – 1 = S7-300 – 2 = Área ET – 3 = Área P – 4 = Área Q – 5 = Área IM3 – 6 = Área IM4 • Bit 7: Reservado

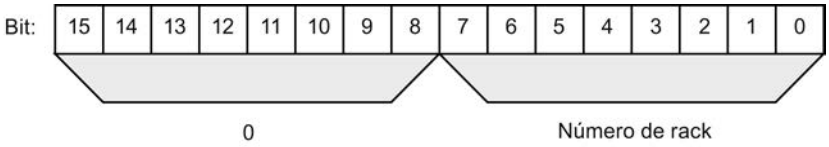
Con determinados módulos se muestran en el registro los siguientes valores:

Nombre	PS (solo S7-400)	CPU	CPU IFM (S7-300)	Rack (solo S7-400)
adr1	Número de rack	Información estándar según se ha descrito anteriormente	Información estándar según se ha descrito anteriormente	Número de rack
adr2	W#16#01FF	W#16#0200 o W#16#0200 a W#16#1800	W#16#0200	W#16#00FF
LogicalAddr	W#16#0000	W#16#7FFF	W#16#007C	W#16#0000
SetType	Información estándar según se ha descrito anteriormente	W#16#00C0 o W#16#0081 o W#16#0082	W#16#00C0	Información estándar según se ha descrito anteriormente
IOStat	W#16#0000	Información estándar según se ha descrito anteriormente	Información estándar según se ha descrito anteriormente	W#16#0000
AreaID_Module Width	W#16#0000	W#16#0011 o W#16#0001 o W#16#0002	W#16#0011	W#16#0000

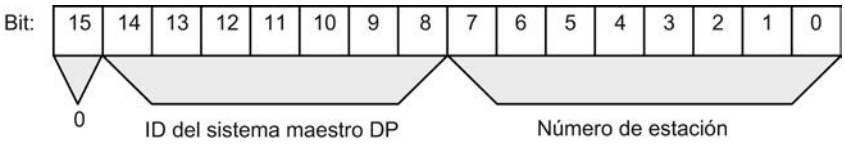
Parámetro adr1

El parámetro adr1 contiene:

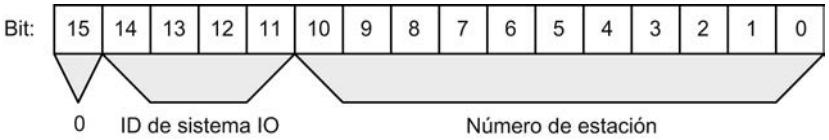
- En caso de configuración centralizada, el número de rack (0-31).



- En caso de estructura descentralizada con PROFIBUS DP
  - La ID del sistema maestro DP (1-32)
  - El número de estación (0-127).



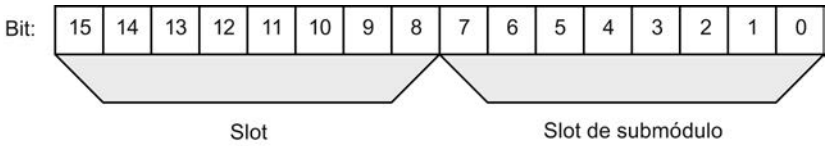
- En caso de estructura descentralizada con PROFINET IO
  - Bit de identificación para PROFINET IO (bit 15)
  - Las dos últimas posiciones de la ID del sistema PROFINET IO (0-15); para obtener la ID del sistema PROFINET IO completa, sume 100 (decimal)
  - El número de estación (0-2047).



Parámetro adr2

El parámetro adr2 contiene:

- En caso de configuración centralizada o descentralizada con PROFIBUS DP, el número de slot y el número de slot de submódulo



- En caso de configuración descentralizada con PROFINET IO, el número de slot.

Parámetros Tipo teórico (SetType) y Tipo real (ActualType) para PROFINET IO

ID de tipo (W#16#...)	Significado
8100	Se especifica en Tipo teórico y en Tipo real si no es posible la comprobación de tipo.
8101	Se especifica en Tipo teórico si es posible la comprobación de tipo.
8101	Se especifica como Tipo real si Tipo teórico=Tipo real.
8102	Se especifica como Tipo real si Tipo teórico<>Tipo real.

SZL-ID W#16#xy92: Información de estado de rack o estación

Finalidad

A través de la lista parcial con la SZL-IDW#16#xy92 se puede obtener información acerca de la configuración teórica y real de los racks con estructura centralizada y estaciones de un sistema maestro DP.

Leer la SZL mediante "RDSYSST" con una CPU S7-400

Si se lee la lista parcial con "RDSYSST (Página 6963)", es imprescindible que los parámetros SZL_ID y INDEX de "RDSYSST (Página 6963)" sean compatibles entre sí.

SZL_ID	INDEX
W#16#0092 o W#16#0192 o W#16#0292 o W#16#0392 o W#16#0492 o W#16#0592 o W#16#0692	ID de un sistema maestro DP conectado a través de una interfaz DP **integrada**.
:	:
W#16#4092 o W#16#4292 o W#16#4692	ID de un sistema maestro DP conectado a través de una interfaz DP **externa**.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy92 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial: • W#16#0092: Consigna de estado de los racks en configuración centralizada o de las estaciones de un sistema maestro DP conectado a través de una interfaz DP integrada. • W#16#4092: Consigna de estado de las estaciones de un sistema maestro DP conectado a través de una interfaz DP externa. • W#16#0192: Estado de activación de las estaciones de un sistema maestro DP conectado a través de una interfaz DP integrada • W#16#0292: Estado real de los racks en configuración centralizada o de las estaciones de un sistema maestro DP conectado a través de una interfaz DP integrada. • W#16#0392: Estado del respaldo por batería de un rack de una CPU, cuando falla por lo menos una batería. • W#16#0492: Estado de todo el respaldo por batería de todos los racks de una CPU • W#16#0592: Estado de la alimentación de 24 V de todos los racks de una CPU. • W#16#4292: Estado real de las estaciones de un sistema maestro DP conectado a través de una interfaz DP externa • W#16#0692: Estado de diagnóstico de los aparatos de ampliación en configuración centralizada o de las estaciones de un sistema maestro DP conectado a través de una interfaz DP integrada • W#16#4692: Estado de diagnóstico de las estaciones de un sistema maestro DP conectado a través de una interfaz DP externa
INDEX	0/ID del sistema maestro DP
LENTHDR	W#16#0010: El registro tiene una longitud de ocho palabras (16 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la IDW#16#xy92 tiene la siguiente estructura:

Contenido	Longitud	Significado
status_0 a status_15	16 bytes	Estado de rack/estado de estación, estado del respaldo o estado de activación (el estado de activación solo es relevante para módulos DP). • W#16#0092: – Bit=0: Rack/estación no configurados – Bit=1: Rack/estación configurados • W#16#4092: – Bit=0: Estación no configurada – Bit=1: Estación configurada • W#16#0192: – Bit=0: Estación no configurada o configurada y activada – Bit=1: Estación configurada y desactivada • W#16#0292: – Bit=0: Fallo del rack/de la estación, o rack/estación desactivados o no configurados – Bit=1: Rack/estación existentes, activados y sin fallo • W#16#4292: – Bit=0: Fallo de la estación, o estación desactivada o no configurada – Bit=1: Estación existente, activada y sin fallo • W#16#0692: – Bit=0: Todos los módulos del aparato de ampliación/de una estación están presentes, disponibles y sin fallo, y la estación está activada – Bit=1: Por lo menos un módulo del aparato de ampliación/de una estación no funciona correctamente, o la estación está desactivada • W#16#4692: – Bit=0: Todos los módulos de una estación están presentes, disponibles y sin fallo, y la estación está activada – Bit=1: Por lo menos un módulo de una estación no funciona correctamente, o la estación está desactivada
status_0	1 bytes	Bit 0: Dispositivo central (INDEX = 0) o estación 1 (INDEX <> 0) Bit 1: 1. Aparato de ampliación o estación 2 : : Bit 7: 7. Aparato de ampliación o estación 8
status_1	1 bytes	Bit 0: 8. Aparato de ampliación o estación 9 : : Bit 7: 15. Aparato de ampliación o estación 16

Contenido	Longitud	Significado
status_2	1 bytes	Bit 0: 16. Aparato de ampliación o estación 17 : : Bit 5: 21. Aparato de ampliación o estación 22 Bit 6: 0 o estación 23 Bit 7: 0 o estación 24
status_3	1 bytes	Bit 0: 0 o estación 25 : : Bit 5: 0 o estación 30 Bit 6: Aparato de ampliación en el rango SIMATIC-S5 o estación 31 Bit 7: 0 o estación 32
status_4	1 bytes	Bit 0: 0 o estación 33 : : Bit 7: 0 o estación 40
: :		
status_15	1 bytes	Bit 0: 0 o estación 121 : : Bit 7: 0 o estación 128

SZL-ID W#16#0x94: información de estado de rack o estación

Finalidad

A través de la lista parcial con la SZL-IDW#16#0x94 se puede obtener información acerca de la configuración teórica y real de los racks en configuración centralizada y estaciones de un sistema maestro PROFIBUS DP/un sistema de controlador PROFINET IO.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#0y94 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial: • W#16#0094: Consigna de estado de los racks en configuración centralizada o de las estaciones de un sistema de controlador IO (bit de estado = 1: Rack/estación configurados) • W#16#0194: Estado de activación de una estación de un sistema de controlador IO configurada y desactivada (bit de estado = 1) • W#16#0294: Estado real de los racks en configuración descentralizada o de las estaciones de un sistema de controlador IO (bit de estado = 1: Rack/estación existentes, activados y sin fallo) • W#16#0694: Estado de diagnóstico de los aparatos de ampliación en configuración descentralizada o de las estaciones de un sistema de controlador IO (bit de estado = 1: Por lo menos un módulo del rack o estación está en fallo o desactivado) • W#16#0794: Estado de diagnóstico / mantenimiento del rack central o de las estaciones de un sistema de controlador IO (bit de estado = 0: No hay averías y no se requiere mantenimiento, bit de estado = 1: El rack / la estación presenta una avería y / o se necesita mantenimiento y / o se solicita mantenimiento) • W#16#0F94: Solo información de encabezado
INDEX	• 0: Módulo central • 1-32: Módulo descentralizado conectado a PROFIBUS DP • 100-115: Módulo descentralizado conectado a PROFINET IO
LENTHDR	Longitud del registro posterior
N_DR	Número de registros

Registro

Un registro de la lista parcial con la IDW#16#0y94 tiene la siguiente estructura:

Contenido	Longitud	Significado
index	1 palabra	• 0: Módulo central • 1-32: Módulo descentralizado conectado a PROFIBUS DP • 100-115: Módulo descentralizado conectado a PROFINET IO
status_0	BOOL	Información de grupo • 1: Por lo menos uno de los bits de estado posteriores tiene el valor 1 • 0: Todos los bits de estado posteriores tienen el valor 0
status_1	BOOL	Estado estación 1
status_2	BOOL	Estado estación 2
..		
status_2047	BOOL	Estado estación 2047

Un bit de estado de racks/estaciones/dispositivos no configurados contiene el valor "0".

Nota

Diferencia importante respecto a la SZL-IDW#16#xy92 anterior

Hay un desplazamiento de un bit en relación a la SZL-IDW#16#xy92 anterior, ya que el bit status_0 sirve como información de grupo.

SZL-ID W#16#xy95: información avanzada del sistema maestro DP/sistema PROFINET IO

Finalidad

La información de estado avanzada de todos los sistemas maestros DP/ sistemas PROFINET IO conocidos por la CPU se puede obtener a través de la lista parcial con la SZL-IDW#16#xy95. En relación con la lista parcial con la SZL-IDW#16#xy90 se ha introducido información sobre los sistemas PROFINET IO y datos adicionales acerca del modo isócrono de un sistema maestro DP.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy95 tiene la siguiente estructura:

Contenido	Significado	
SZL-ID	La SZL-ID del extracto de lista parcial	
	W#16#0195:	Información avanzada acerca de un sistema maestro DP/sistema PROFINET IO
	W#16#0F95:	Solo información de encabezado de la lista parcial SZL
INDEX	• Para el extracto de lista parcial con la SZL-IDW#16#0195: – byte menos significativo: B#16#00 – byte más significativo: ID de sistema maestro DP/ID de sistema PROFINET IO • Para el extracto de lista parcial con la SZL-IDW#16#0F95: – W#16#0000	
LENTHDR	W#16#0028:	Un registro tiene una longitud de 20 palabras (40 bytes)
N_DR	Número de registros: Para el extracto de lista parcial con la SZL-IDW#16#0195: de 0 a 1	

Registro

Un registro de la lista parcial con la IDW#16#xy95 tiene la siguiente estructura:

Nombre	Longitud	Significado
dp_m_id	1 bytes	ID de sistema maestro DP/ID de sistema PROFINET IO
dp_m_rack	1 bytes	Número de rack del maestro DP/controlador IO • Con una CPU estándar: 0

Nombre	Longitud	Significado	
dp_m_slot	1 bytes	Slot del maestro DP/controlador IO o slot de la CPU (con interfaz DP integrada)	
dp_m_SubModule	1 bytes	• Con interfaz integrada: Número de interfaz del maestro DP/controlador IO: – 1: X2 – 2: X1 – 3: IF1 – 4: IF2 • Con interfaz externa: 0	
LogicalAddr	2 bytes	Dirección lógica inicial del maestro DP/controlador IO	
res	2 bytes	Reservado	
res	2 bytes	Reservado	
dp_m_state	1 bytes	• Otras características del sistema maestro DP/sistema PROFINET IO	
		Bit 0:	Modo DP (solo con PROFIBUS DP) • 0: Compatible con S7 • 1: DPV1
		Bit 1:	Ciclo DP o PN • 0: No equidistante • 1: Equidistante
		Bits de 2 a 6:	• Reservado
		Bit 7:	Tipo de maestro DP/controlador IO • 0: Maestro DP integrado/controlador IO integrado • 1: Maestro DP externo/controlador IO externo
dp_address	1 bytes	Número de dispositivo DP (dirección PROFIBUS)	
res	2 bytes	Reservado	
SyncCyclInter_OB	1 bytes	OB de alarma de sincronismo asignado (relevante solo si el ciclo DP o PN es equidistante)	
res	1 bytes	Reservado	
baudrate	4 bytes	Velocidad de transferencia del sistema maestro DP o del sistema PNIO en formato Hex	
IsoDPCycle	4 bytes	Duración en µs del ciclo DP o PN equidistante	
res	16 bytes	Reservado	

SZL-ID W#16#xy96: información de estado de módulo con PROFINET IO y PROFIBUS DP

Finalidad

La información de estado de todos los módulos asignados a la CPU se puede obtener a través de la lista parcial con la SZL-IDW#16#xy96.

Para complementar SZL-IDW#16#xy91 (Página 7002), la lista parcial con la SZL-IDW#16#xy96 ofrece datos de estado adicionales de módulos y submódulos.

Contiene información específica de PROFINET IO e información sobre módulos PROFIBUS DP y módulos centrales.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xy96 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial • W#16#0696 Información de estado de todos los submódulos de un módulo especificado (con PROFIBUS DP y módulos centrales, el nivel de submódulo no existe). • W#16#0C96 Información de estado de un módulo/un submódulo central o conectado a una interfaz PROFIBUS DP/PROFINET a través de la dirección inicial.
INDEX	• Bits de 0 a 14: Dirección del módulo • Bit 15: 0 = entrada, 1 = salida
LENTHDR	Longitud del registro posterior
N_DR	Número de registros

Registro

Un registro de la lista parcial con la IDW#16#xy96 tiene la siguiente estructura:

Contenido	Longitud	Significado
LogicalAddr	1 palabra	• Bits de 0 a 14: Dirección del módulo • Bit 15: 0 = entrada, 1 = salida
System	1 palabra	ID de módulo central/ ID de sistema maestro DP/ ID de sistema PROFINET IO: • 0: Módulo central • 1-32: Módulo descentralizado conectado a PROFIBUS DP • 100-115: Módulo descentralizado conectado a PROFINET IO
API*	2 palabras	Perfil de aplicación configurado para un dispositivo PROFINET descentralizado. Los perfiles son definiciones específicas de sector o de tecnología, que van más allá de la norma PROFINET. En el perfil 0, los datos concuerdan completamente con las definiciones de la norma PROFINET.
Station	1 palabra	Número de rack/número de estación/número de dispositivo
Slot	1 palabra	Número de slot
Subslot	1 palabra	Slot de submódulo (si no puede insertarse ningún módulo, debe indicarse aquí "0")
Offset	1 palabra	Offset en el área de direcciones de datos útiles del módulo correspondiente
SetType	7 palabras	Tipo teórico En PROFINET IO, el tipo teórico tiene estructura jerárquica

Palabra	PROFINET IO	PROFIBUS DP
1:	Número de fabricante o ID de perfil (p. ej.: W#16#FF00 para PROFIBUS)	0000
2:	Dispositivo	0000
3:	Número consecutivo o índice de perfil	0000
4:	1. Palabra de la palabra doble ID módulo	ID de tipo

Contenido	Longitud	Significado		
		5:	2. Palabra de la palabra doble ID módulo	0000
		6:	1. Palabra de la palabra doble ID submódulo	0000
		7:	2. Palabra de la palabra doble ID submódulo	0000
SetType_UnEqual_ActualType	1 palabra	Identificador teórico/real • Bit 0 = 0: Teórico coincide con real • Bit 0 = 1: Teórico no coincide con real • Bits de 1 a 15: Reservado		
res	1 palabra	Reservado		
IOStat	1 palabra	Estado E/S • Bit 0 = 1: Fallo del módulo (detectado mediante alarma de diagnóstico) • Bit 1 = 1: El módulo existe Bit 2 = 1: Módulo no disponible • Bit 3 = 1: El módulo está desactivado Bit 4 = 1: Fallo de la estación (solo slot de representante) • Bit 5 = 1: M7: El módulo puede ser host para submódulos S7: Para este módulo o estación hay un proceso CiR activo actualmente, o todavía no finalizado • Bit 6 = 1: Reservado para S7-400 • Bit 7 = 1: Módulo en el segmento de bus local (solo con S7-300) • Bit 8 = 1: Módulo, mantenimiento necesario ("verde") • Bit 9 = 1: Módulo, solicitud de mantenimiento ("amarillo") • Bits de 10 a 15: Reservado		
AreaID_ModuleWidth	1 palabra	Identificación de área/ancho de módulo • bits de 0 a 2: Ancho de módulo • Bit 3: Reservado • bits de 4 a 6: Identificación de área – 0 = S7-400 – 1 = S7-300 – 2 = PROFINET IO (descentralizado) – 3 = área P – 4 = área Q – 5 = área IM3 – 6 = área IM4 • Bit 7: Reservado • Bit 7: Reservado		
res	5 palabras	Reservado		
* Application Process Instance				

Lista parcial con la SZL-IDW#16#0696 para módulos conectados al PROFIBUS DP

Se emite el aviso de error "No existe nivel de submódulo".

SZL-ID W#16#xy9C: Información del intercambiador de herramientas (PROFINET IO)

Finalidad

Puede obtenerse información sobre los intercambiadores de herramientas configurados y sus herramientas a través de la lista parcial con la SZL-ID W#16#xy9C.

Los intercambiadores de herramientas son dispositivos IO que gestionan herramientas. Una herramienta consta de uno o más dispositivos IO. Cada herramienta se asigna de forma unívoca a un puerto de un intercambiador de herramientas.

En un cambio de herramientas en primer lugar se desactivan todos los dispositivos IO que pertenecen a la herramienta activa en esos momentos y, a continuación, se activan los dispositivos IO pertenecientes a la nueva herramienta. La desactivación y activación se realizan con la instrucción "D_ACT_DP (Página 6764)".

Encabezado

El encabezado de la lista parcial con la SZL-ID W#16#xy9C tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La ID SZL del extracto de lista parcial • W#16#009C Información sobre todos los intercambiadores de herramientas y sus herramientas en un sistema PROFINET IO • W#16#019C Información sobre todos los intercambiadores de herramientas de un sistema PROFINET IO • W#16#029C Información sobre un intercambiador de herramientas y sus herramientas • W#16#039C Información sobre una herramienta y sus dispositivos IO • W#16#0F9C Solo información de encabezado de la lista parcial SZL
INDEX	Para el extracto de lista parcial con la ID SZL • W#16#009C: ID del sistema PROFINET IO • ID del sistema W#16#019C: PROFINET IO • W#16#029C: dirección lógica del intercambiador de herramientas • W#16#039C: dirección lógica de un dispositivo IO cualquiera de la herramienta • ID del sistema W#16#0F9C: PROFINET IO
LENTHDR	W#16#000C (longitud del siguiente registro en bytes: 12)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-ID W#16#xy9C tiene la siguiente estructura:

Contenido	Longitud	Significado
StationW	1 WORD	Número de estación (del registro, puede ser intercambiador de herramientas o estación de una herramienta)
LogAdrW	1 WORD	• Bits de 0 a 14: Dirección del módulo • Bit 15: 0 = entrada, 1 = salida
StationWZK	1 WORD	Número de estación del cabezal de la herramienta (el cabezal de la herramienta es el dispositivo IO de una herramienta que se ha conectado directamente al puerto del intercambiador de herramientas).
StationWZW	1 WORD	Número de estación del intercambiador de herramientas
SlotWZW	1 WORD	Slot del intercambiador de herramientas
SubslotWZW	1 WORD	Slot de submódulo del intercambiador de herramientas

Nota

Si un intercambiador de herramientas posee varios puertos en los que se administran herramientas, para cada puerto se suministra un registro.

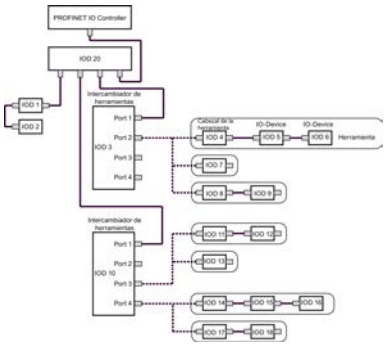
Ejemplo

El uso de las diferentes listas parciales individuales se representa una vez más a partir del siguiente ejemplo.

En la línea de un sistema PROFINET IO hay dos intercambiadores de herramientas (IOD 3 y IOD 10), cuyas herramientas tienen la siguiente estructura:

- Intercambiador de herramientas IOD 3 (IO-Device 3) con las tres herramientas siguientes en el puerto 2 del intercambiador:
  - Herramienta 1 (IOD 4, IOD 5 y IOD 6)
  - Herramienta 2 (IOD 7)
  - Herramienta 3 (IOD 8 y IOD 9)
- Intercambiador de herramientas IOD 10 con dos puertos de intercambiador de herramientas, de los que cada uno posee dos herramientas.
  - Puerto 3 del intercambiador de herramientas: Herramienta 1 (IOD 11 y IOD 12), herramienta 2 (IOD 13)
  - Puerto 4 del intercambiador de herramientas: Herramienta 1 (IOD 14, IOD 15 y IOD 16), herramienta 2 (IOD 17 y IOD 18)

De ello resulta la estructura siguiente:



Las listas parciales de la SZL ofrecen registros para los siguientes IO-Devices:

- Extracto de listas parciales con SZL-ID W#16#009C (índice: ID del sistema PROFINET IO): proporciona 17 registros para los siguientes IO-Devices:
  - Intercambiador de herramientas: IOD 3, IOD 10 (para el puerto 3) e IOD 10 (para el puerto 4)
  - Herramientas: IOD 4, IOD 5, IOD 6, IOD 7, IOD 8, IOD 9, IOD 11, IOD 12, IOD 13, IOD 14, IOD 15, IOD 16, IOD 17 y IOD 18
- Extracto de listas parciales con SZL-ID W#16#019C (índice: ID del sistema PROFINET IO): proporciona 3 registros para los siguientes IO-Devices:
  - Intercambiador de herramientas: IOD 3, IOD 10 (para el puerto 3) y IOD 10 (para el puerto 4)
  - Herramientas: Ninguno
- Extracto de listas parciales con SZL-ID W#16#029C (índice: Dirección del IOD 3): proporciona 3 registros para los siguientes IO-Devices:
  - Intercambiador de herramientas: IOD 3
  - Herramientas: IOD 4, IOD 5, IOD 6, IOD 7, IOD 8 y IOD 9
- Extracto de listas parciales con SZL-ID W#16#029C (índice: Dirección del IOD 10): proporciona 10 registros para los siguientes IO-Devices:
  - Intercambiador de herramientas: IOD 10 (para el puerto 3) y IOD 10 (para el puerto 4)
  - Herramientas: IOD 11, IOD 12, IOD 13, IOD 14, IOD 15, IOD 16, IOD 17 y IOD 18
- Extracto de listas parciales con SZL-ID W#16#039C (índice: Dirección lógica del IOD 4): proporciona 3 registros para los siguientes IO-Devices: (esto se aplica de forma análoga para la dirección lógica del IOD 5 y IOD 6)
  - Intercambiador de herramientas: Ninguno
  - Herramientas: IOD 4, IOD 5 y IOD 6
- Extracto de listas parciales con SZL-ID W#16#039C (índice: Dirección lógica del IOD 13): proporciona un registro para los siguientes IO-Devices:
  - Intercambiador de herramientas: Ninguno
  - Herramientas: IOD 13

SZL-ID W#16#xyA0: búfer de diagnóstico

Finalidad

Las entradas del búfer de diagnóstico del módulo se pueden conocer a través de la lista parcial con la SZL-IDW#16#xyA0.

Nota

Las CPUs S7-300 proporcionan como máximo tantos registros como la cantidad de entradas de búfer de diagnóstico mostradas en el modo de operación RUN (valor predeterminado: 10). 10). Las CPUs S7-400 proporcionan como máximo 21 registros.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#xyA0 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	La SZL-ID del extracto de lista parcial: • W#16#00A0: Todas las entradas transmisibles en el estado operativo actual • W#16#01A0: Las entradas más recientes, cuyo número se indica mediante el parámetro INDEX. Si el número de avisos del búfer de diagnóstico es menor que el número máximo de avisos configurado por el usuario, en algunos casos la instrucción "RDSYSST (Página 6963)" puede devolver valores no válidos con este extracto de lista parcial. Por ello debe evitarse un POWER OFF sin respaldo. • W#16#0FA0: Solo información de encabezado de la lista parcial SZL
INDEX	Solo para SZL-IDW#16#01A0: Número de las entradas más recientes
LENTHDR	W#16#0014: Un registro tiene una longitud de 10 palabras (20 bytes)
N_DR	Número de registros

Registro

Un registro de la lista parcial con la SZL-IDW#16#xyA0 tiene la siguiente estructura:

Nombre	Longitud en palabras	Significado
ID	1 palabra	ID de evento
info	5 palabras	Información sobre el evento o sus efectos
time	4 palabras	Sello de tiempo del evento

Búfer de diagnóstico

Encontrará más información sobre eventos del búfer de diagnóstico en la ayuda contextual para el evento visualizado.

SZL-ID W#16#00B1: información de diagnóstico de rack

Finalidad

Los 4 primeros bytes de diagnóstico de un módulo apto para diagnóstico se pueden conocer a través de la lista parcial con la SZL-IDW#16#00B1.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#00B1 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	W#16#00B1
INDEX	• bits de 0 a 14: Dirección base lógica • Bit 15: 0 para entrada, 1 para salida
LENTHDR	W#16#0004: Un registro tiene una longitud de 2 palabras (4 bytes)
N_DR	1

Registro

Un registro de la lista parcial con la SZL-IDW#16#00B1 tiene la siguiente estructura:

Nombre	Longitud	Significado
byte0	1 bytes	• Bit 0: Fallo de módulo/OK (identificador de fallo de grupo) • Bit 1: Error interno • Bit 2: Error externo • Bit 3: Existe error de canal • Bit 4: Falta tensión auxiliar externa • Bit 5: Falta conector frontal • Bit 6: Módulo no parametrizado • Bit 7: Parámetro erróneo en módulo
byte1	1 bytes	• Bit 0 a bit 3: Clase de módulo (CPU, FM, CP, IM, SM, ...) • Bit 4: Existe información de canal • Bit 5: Existe información de usuario • Bit 6: Alarma de diagnóstico de sustituto • Bit 7: Mantenimiento necesario (solo con PROFINET IO)

Nombre	Longitud	Significado
byte2	1 bytes	• Bit 0: Módulo de usuario erróneo o no existe • Bit 1: Fallo de comunicación • Bit 2: Estado operativo RUN/STOP (0 = RUN, 1 = STOP) • Bit 3: La vigilancia de tiempo ha reaccionado (watchdog) • Bit 4: Fallo de la tensión de alimentación interna del módulo • Bit 5: Batería vacía (BFS) • Bit 6: Fallo de todo el respaldo • Bit 7: Mantenimiento solicitado (solo con PROFINET IO)
byte3	1 bytes	• Bit 0: Fallo de aparato de ampliación (detectado por IM) • Bit 1: Fallo de procesador • Bit 2: Error de EPROM • Bit 3: Error de RAM • Bit 4: Error de ADU/DAU • Bit 5: Fallo de fusible • Bit 6: Alarma de proceso perdida • Bit 7: Reserva (inicializado con 0)

SZL-ID W#16#00B2: registro de diagnóstico 1 a través de dirección física

Finalidad

El registro de diagnóstico 1 de un módulo insertado central (es decir, excluyendo DP y submódulos) puede obtenerse a través de la lista parcial con la SZL-IDW#16#00B2. Para especificar el módulo debe indicarse el número de rack y de slot.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#00B2 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	W#16#00B2
INDEX	W#16#xxyy: • xx contiene el número del rack • yy contiene el número del slot
LENTHDR	La longitud del registro depende del módulo
N_DR	1

Registro

El tamaño de un registro de la lista parcial con la SZL-IDW#16#00B2, así como el contenido del registro, dependen del módulo que se utilice. Encontrará más información al respecto en el correspondiente manual del módulo.

SZL-ID W#16#00B3: datos de diagnóstico de módulo a través de la dirección base lógica

Finalidad

Todos los datos de diagnóstico de un módulo se pueden obtener a través de la lista parcial con la SZL-IDW#16#00B3. Esta información también es posible para DP y submódulos. El módulo se selecciona a través de su dirección base lógica.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#00B3 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	W#16#00B3
INDEX	• bits de 0 a 14: Dirección base lógica • Bit 15: 0 para entrada, 1 para salida
LENTHDR	Longitud de un registro (en función del módulo)
N_DR	1

Registro

El tamaño de un registro de la lista parcial con la SZL-IDW#16#00B3, así como su contenido, dependen del módulo que se utilice. Encontrará más información al respecto en el correspondiente manual del módulo.

SZL-ID W#16#00B4: datos de diagnóstico de un esclavo DP

Finalidad

Los datos de diagnóstico de un esclavo DP se pueden obtener a través de la lista parcial con la SZL-IDW#16#00B4. El formato de estos datos de diagnóstico se especifica en EN50 170 Volume 2, PROFIBUS. El módulo se selecciona a través de su dirección de diagnóstico configurada.

Encabezado

El encabezado de la lista parcial con la SZL-IDW#16#00B4 tiene la siguiente estructura:

Contenido	Significado
SZL-ID	W#16#00B4
INDEX	Dirección de diagnóstico configurada del esclavo DP
LENTHDR	Longitud de un registro. La longitud máxima es de 240 bytes; en los esclavos normalizados en los que el número de datos de diagnóstico normalizados es superior a 240 bytes, con un máximo de 244, se leen los 240 primeros bytes y se adjunta a los datos el correspondiente bit de desbordamiento.
N_DR	1

Registro

Un registro de la lista parcial con la SZL-IDW#16#00B4 tiene la siguiente estructura:

Nombre	Longitud	Significado
status1	1 bytes	Estado de estación 1
status2	1 bytes	Estado de estación 2
status3	1 bytes	Estado de estación 3
Station_No	1 bytes	Número de estación maestra
Vendor_ID_highByte	1 bytes	Identificación de fabricante (high byte)
Vendor_ID_lowByte	1 bytes	Identificación de fabricante (low byte)
....	....	Otros diagnósticos específicos de esclavo

OB_RT: Determinar tiempo de ejecución del programa OB

Descripción

Esta instrucción permite conocer el tiempo de ejecución de OBs individuales en distintos periodos.

Nota

Esta instrucción devuelve los últimos valores de tiempo registrados para el OB deseado, sin importar que este se encuentre cargado o no en el momento actual. Los datos no se anulan mediante borrado ni sobrecarga, sino solo mediante un rearranque completo (en caliente).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "OB_RT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Significado
OB_NR	Input	INT	I, Q, M, D, L	OB cuyos últimos tiempos conocidos se desean consultar. Son números válidos de OB todos los OBs implementados en la CPU, excepto OB 121 y OB 122. El procesamiento de errores sincrónicos se incluye en el tiempo de procesamiento del OB causante del error. Si se indican los OBs 121 y 122 o un OB no cargado en la CPU, se devuelve un aviso de error. Si OB_NR=0, se transfieren los datos del OB en cuyo contexto se haya llamado la instrucción. Si se llama la instrucción "OB_RT" en los OBs 121 ó 122 con OB_NR=0, se devuelven los temporizadores del OB causante de la alarma, junto con los temporizadores del OB 12x.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error. De lo contrario, RET_VAL contiene el número del OB para el que se han consultado dichos datos.
PRIO	Output	INT	I, Q, M, D, L	En PRIO se devuelve la clase de prioridad del OB consultado.
LAST_RT	Output	DINT	I, Q, M, D, L	Tiempo de ejecución en µs del último procesamiento finalizado del OB indicado. Si el OB cuyos tiempos de ejecución se desean conocer está procesándose actualmente, se aplica lo siguiente: En la primera llamada de la instrucción "OB_RT" durante el procesamiento actual del OB deseado se indica en LAST_RT el tiempo de ejecución del último procesamiento finalizado del OB. Con cada nueva llamada de la instrucción "OB_RT" durante el procesamiento actual del OB deseado, en LAST_RT • se indica DW#16#FFFF FFFF, en caso de que en el OB deseado ya se haya efectuado una llamada de "OB_RT" OB_NR=0, • el tiempo de ejecución del último procesamiento finalizado del OB, en caso de que en el OB deseado no se haya efectuado ninguna llamada de "OB_RT" con OB_NR=0. Nota: Los tiempos de interrupción por OBs de mayor prioridad no se indican en LAST_RT . Las prestaciones específicas de OB del sistema operativo (p. ej. la creación y preparación de información de arranque de OB, la actualización de la memoria imagen de proceso, la actualización de la memoria imagen parcial de proceso) se indican en LAST_RT .

Parámetro	Declaración	Tipo de datos	Área de memoria	Significado
LAST_ET	Output	DINT	I, Q, M, D, L	Intervalo en µs entre la solicitud de OB y el fin del procesamiento del OB indicado, para el último procesamiento finalizado del OB indicado. Si el OB cuyos tiempos de ejecución se desean conocer está procesándose actualmente, rige lo siguiente: En la primera llamada de la instrucción "OB_RT" durante el procesamiento actual del OB deseado se indica en LAST_ET el intervalo entre la última solicitud de OB completamente procesada y el fin del procesamiento del OB indicado. Con cada nueva llamada de la instrucción "OB_RT" durante el procesamiento actual del OB deseado, en LAST_ET • se indica DW#16#FFFF FFFF , en caso de que en el OB deseado ya se haya efectuado una llamada de "OB_RT" OB_NR=0, • el intervalo entre la última solicitud de OB completamente procesada y el fin del procesamiento del OB deseado, en caso de que en el OB deseado no se haya efectuado ninguna llamada de "OB_RT" con OB_NR=0. Nota: Los tiempos de interrupción por OBs de mayor prioridad se indican en LAST_ET .
CUR_T	Output	DINT	I, Q, M, D, L	Momento de la solicitud del OB indicado que está procesándose actualmente, en forma de tiempo relativo en µs. Si el OB indicado no está procesándose actualmente, CUR_T contiene el valor "0". Nota: La hora del sistema es un contador que contabiliza desde "0" a "2 147 483 647" µs. En caso de desbordamiento, el contador empieza de nuevo desde "0".
CUR_RT	Output	DINT	I, Q, M, D, L	Tiempo de ejecución transcurrido en µs del procesamiento actual del OB indicado. CUR_RT es "0" si el OB (todavía) no está procesándose. Finalizado el procesamiento, el tiempo de ejecución se transfiere a LAST_RT , y CUR_RT se ajusta a 0. Nota: Los tiempos de interrupción por OBs de mayor prioridad no se indican en CUR_RT . Las prestaciones específicas de OB del sistema operativo suministradas hasta el momento de la llamada de la instrucción "OB_RT" se indican en CUR_RT .

Parámetro	Declaración	Tipo de datos	Área de memoria	Significado
CUR_ET	Output	DINT	I, Q, M, D, L	El intervalo en µs transcurrido desde la solicitud del OB indicado que está procesándose actualmente. CUR_ET es "0" si el OB indicado no está procesándose actualmente. Finalizado el procesamiento, el tiempo de ejecución se transfiere a LAST_ET , y CUR_ET se ajusta a "0". Nota: Los tiempos de interrupción por OBs de mayor prioridad se indican en CUR_ET .
NEXT_ET	Output	DINT	I, Q, M, D, L	Si hay más procesamientos del OB indicado pendientes antes de que finalice la solicitud actual, en NEXT_ET se indica el intervalo en µs entre el momento actual y el momento de la próxima solicitud. NEXT_ET es "0" si no existe para el OB en cuestión ningún otro evento de arranque aparte del evento de inicio en procesamiento o pendiente de procesamiento. WinLC RTX y las CPUs S7-400 no utilizan este parámetro. En estos sistemas, NEXT_ET tiene el valor "FFFF FFFF". Nota: Los tiempos de interrupción por OBs de mayor prioridad se indican en NEXT_ET .

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Los tiempos incluyen también los tiempos de ejecución de posibles procesamientos anidados de alarmas de errores sincrónicos (OB 121, OB 122).

Nota

Si se indica en OB_NR un número de OB que está presente en los datos de la CPU, pero el OB correspondiente todavía no ha sido llamado por el sistema operativo o no ha sido cargado en la CPU por el usuario, RET_VAL contiene el número de OB indicado, y PRIO la prioridad configurada (o en su caso predeterminada) del OB indicado. Los parámetros de tiempo de ejecución (CUR_RT, CUR_ET, LAST_RT, LAST_ET, NEXT_ET) devuelven el valor inicial DW#16#FFFF FFFF. En los siguientes tipos de arranque o cambios de estado operativo, los parámetros de tiempo de ejecución se ajustan a su valor inicial: Rearranque completo, arranque en frío, rearranque.

Parámetro RET_VAL

Clase de evento código de error	Explicación
de 1 a 102	Número del OB al que se transfiere la información
W#16#8080	El parámetro OB_NR contiene un valor no admisible.
W#16#8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

C_DIAG: Determinar estado actual de la conexión

Descripción

Esta instrucción permite conocer el estado actual de todas las conexiones S7 y todas las conexiones S7 de alta disponibilidad (o sus conexiones parciales).

Evaluando adecuadamente estos datos de conexión pueden detectarse el fallo de las conexiones S7 y las conexiones S7 de alta disponibilidad y, en su caso, notificarlo a un sistema de manejo y visualización. Las conexiones vigiladas pueden existir tanto entre sistemas de automatización como entre un sistema de automatización y un sistema de manejo y visualización.

Nota

El cambio de estado operativo RUN -> STOP -> RUN de una CPU no influye en el estado de las conexiones configuradas. Excepción: Cuando una estación H pasa del estado redundante al estado STOP, en el caso de las conexiones de alta disponibilidad se deshacen las conexiones parciales que terminan en la CPU de reserva.

En cambio, después de un fallo de red, se vuelven a establecer todas las conexiones configuradas, con lo que cambia el estado de conexión.

Así, al llamarse por primera vez la instrucción en el arranque o después de este, la información de conexión varía en función de si el último estado operativo de la CPU fue STOP o POWER OFF.

Funcionamiento

C_DIAG" es una instrucción asíncrona, de modo que la ejecución puede prolongarse a lo largo de varias llamadas.

Para iniciar la petición, debe llamarse "C_DIAG" con REQ=1.

Si la petición se ha podido ejecutar de inmediato, la instrucción devuelve en el parámetro de salida BUSY el valor "0". Si BUSY tiene el valor "1", la petición todavía está procesándose.

Llamada

Para detectar fallos de conexiones S7 y conexiones S7 de alta disponibilidad debe llamarse "C_DIAG" en un OB de alarma cíclica que sea iniciado por el sistema operativo p. ej. cada 10 segundos.

Dado que en circunstancias normales el estado de las conexiones no suele variar, para estas llamadas cíclicas es recomendable no copiar los datos de conexión al programa de usuario hasta que hayan cambiado respecto a la última llamada (llamada con MODE=B#16#02, ver abajo).

La instrucción "C_DIAG" tiene cuatro modos de operación posibles, que se enumeran en la tabla siguiente.

MODE (B#16#...)	"C_DIAG" copia datos de conexión en el programa de usuario	"C_DIAG" transfiere información de acuse al sistema operativo
00	No	SÍ
01	SÍ	SÍ
02	• Sí, en caso de que los datos de conexión hayan cambiado • No, en caso de que los datos de conexión no hayan cambiado	SÍ
03	SÍ	No

Al transferir la información de acuse al sistema operativo, se acusan las modificaciones de estado de los datos de conexión producidas desde la última llamada de "C_DIAG" (con MODE=B#16#00, 01 o 02).

Nota

Si se llama "C_DIAG" en un OB de alarma cíclica en el modo de operación "Copia condicionada" (MODE=B#16#02), debe comprobarse que después de un arranque en frío de la CPU no haya valores iniciales en el área de destino. Para asegurarlo, debe llamarse una vez "C_DIAG" en el OB 102 con el modo de operación "Copia incondicional con acuse" (MODE=B#16#01).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "C_DIAG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control request to activate REQ=1: Se lanza la petición, si todavía no se ha iniciado.
MODE	Input	BYTE	I, Q, M, D, L o constante	ID de la petición Valores posibles: • B#16#00: "C_DIAG" no copia datos de conexión, sino que se limita a transferir al sistema operativo una información de acuse. • B#16#01: "C_DIAG" copia los datos de conexión al programa de usuario, sea cual sea su estado de modificación, y transfiere al sistema operativo una información de acuse. • B#16#02: Si los datos de conexión han cambiado, "C_DIAG" los copia en el programa de usuario. Si no han cambiado, no se realiza la copia. "C_DIAG" transfiere al sistema operativo una información de acuse en ambos casos. • B#16#03: "C_DIAG" copia los datos de conexión en el programa de usuario, sea cual sea su estado de modificación. No transfiere al sistema operativo ninguna información de acuse.
RET_VAL	Return	INT	I, Q, M, D, L	Valor de retorno (código de error o estado de la petición)
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La petición todavía no ha finalizado.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
N_CON	Output	INT	I, Q, M, D, L	Índice de la última estructura en CON_ARR, en la que .DIS_PCON o .DIS_CON tienen el valor TRUE . Así, en el programa de usuario solo es necesario comprobar los primeros N_CON elementos de CON_ARR . Nota: La primera estructura del campo CON_ARR tiene el índice "1".
CON_ARR	Output	ANY	I, Q, M, D, L	Área de destino para los datos de conexión leídos. Solo se admite el tipo de datos BYTE . Cada conexión tiene asignada una estructura. Debe escogerse un área de destino lo bastante grande para albergar todas las estructuras incluso en caso de alcanzarse el máximo de conexiones posibles en la CPU. Téngalo en cuenta sobre todo después de cambiar una CPU, siempre que la CPU nueva tenga más recursos que la anterior.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro CON_ARR

El área de destino para los datos de conexión leídos es un campo de estructuras. Cada conexión tiene asignada una estructura.

No es necesario que el campo esté ocupado desde las posiciones iniciales con valores válidos, y también puede haber entradas no válidas entre dos valores válidos.

Las conexiones no están ordenadas en función de referencias de conexión.

Nota

Al copiar los datos de conexión desde el sistema operativo al área de destino especificada, se garantiza la coherencia de los datos de una conexión.

Formato de una estructura

Parámetro	Tipo de datos	Descripción
CON_ID	WORD	Referencia de conexión que se ha asignado a esa conexión en la configuración W#16#FFFF: Identificador no válido, es decir, conexión no configurada. Si adicionalmente CON_ARR[i].DIS_PCON o CON_ARR[i].DIS_CON están activadas (ver abajo), la conexión ha sido reconfigurada o borrada desde la última llamada de "C_DIAG".
STAT_CON	BYTE	Estado actual de la conexión S7 o de la conexión S7 de alta disponibilidad Valores posibles: • B#16#00: Conexión S7 no establecida • B#16#10: Conexión S7 de alta disponibilidad no establecida • B#16#01: La conexión S7 está estableciéndose en este momento • B#16#11: La conexión S7 de alta disponibilidad está estableciéndose en este momento • B#16#02: La conexión S7 está establecida • B#16#12: La conexión S7 de alta disponibilidad está establecida, pero sin alta disponibilidad • B#16#13: La conexión S7 de alta disponibilidad está establecida, con alta disponibilidad
PROD_CON	BYTE	Número de conexión parcial de la conexión productiva Valores posibles: 0, 1, 2, 3
STBY_CON	BYTE	Número de conexión parcial de la conexión de reserva (standby) (B#16#FF: No hay conexión de reserva) Valores posibles: 0, 1, 2, 3 Nota: Solo una conexión S7 de alta disponibilidad puede poseer una conexión de reserva.
DIS_PCON	BOOL	Las transiciones W#16#12 -> W#16#13 y W#16#13 -> W#16#12 de CON_ARR[i].STAT_CON desde la última llamada de instrucción ajustan CON_ARR[i].DIS_PCON a "1". Cualquier otro cambio de estado de la conexión i no provoca alteraciones en CON_ARR[i].DIS_PCON . Nota: • Con MODE=B#16#01 y 02 , al copiar los datos de conexión en el área de destino se desactiva el bit del sistema operativo que corresponde a DIS_PCON. • Con MODE=B#16#03 , el bit del sistema operativo que corresponde a DIS_PCON permanece invariable.
DIS_CON	BOOL	Toda modificación de CON_ARR[i].STAT_CON desde la última llamada de "C_DIAG", a excepción de las transiciones W#16#12 -> W#16#13 y W#16#13 -> W#16#12 , ajusta CON_ARR[i].DIS_CON a 1. Nota: • Con MODE=B#16#01 y 02 , al copiar los datos de conexión en el área de destino se desactiva el bit del sistema operativo que corresponde a "DIS_CON". • Con MODE=B#16#03 , el bit del sistema operativo que corresponde a "DIS_CON" permanece invariable.
RES0	BYTE	Reservado (B#16#00)
RES1	BYTE	Reservado (B#16#00)

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	• MODE=B#16#00, 01 o 02: Sin cambio del estado de conexión (elemento de estructura STAT_CON) desde la última llamada. La petición se ha ejecutado sin errores. • MODE=B#16#03: La copia se ha ejecutado sin errores.
0001	• MODE=B#16#00, 01 o 02: Cambio del estado de conexión (elemento de estructura STAT_CON) en por lo menos una conexión desde la última llamada. La petición se ha ejecutado sin errores. • MODE=B#16#03: RET_VAL W#16#0001 no es posible:
7000	Primera llamada con REQ=0. La petición definida mediante MODE no se procesa; BUSY tiene el valor "0",
7001	Primera llamada con REQ=1. Se ha lanzado la petición definida mediante MODE . BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante). La petición activada todavía está en curso. BUSY tiene el valor "1".
8080	El parámetro MODE contiene un valor no admisible.
8081	El parámetro CON_ARR contiene un tipo de datos no admisible.
8082	El parámetro CON_ARR contiene una indicación de longitud demasiado pequeña. La instrucción no copia datos en el área de destino.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Funciones de bloques de datos

READ_DBL: Leer de un bloque de datos de la memoria de carga

Descripción

La instrucción copia un DB o una parte de un DB que se encuentra en la memoria de carga (Micro Memory Card) en el área de datos de un DB de destino. El DB de destino tiene que ser relevante para la ejecución, es decir, no puede haberse creado con el atributo UNLINKED. El contenido de la memoria de carga no se modifica en la operación de copia.

Para garantizar la coherencia de los datos no debe modificarse el área de destino durante el procesamiento de "READ_DBL" (es decir, mientras el parámetro BUSY tenga el valor TRUE).

Para los parámetros SRCBLK y DSTBLK se aplican las siguientes restricciones:

- En el caso de un puntero ANY del tipo BOOL la longitud debe ser divisible entre 8.
- En el caso de un puntero ANY del tipo STRING la longitud debe ser 1.

La longitud del DB de origen puede consultarse, en caso necesario, con la instrucción "TEST_DB (Página 7044)".

Nota

READ_DBL se procesa de modo asíncrono, lo que la hace inadecuada para leer variables repetidamente (es decir, de modo cíclico) en la memoria de carga.

Una petición iniciada siempre se finaliza. Si se alcanza el número máximo de instrucciones "READ_DBL" activas simultáneamente y en ese momento se vuelve a llamar "READ_DBL" en una clase de mayor prioridad, se devuelve el código de error W#16#80C3. No es recomendable, por lo tanto, volver a iniciar inmediatamente la petición de mayor prioridad.

Funcionamiento

La instrucción "READ_DBL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "READ_DBL" con REQ =1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "READ_DBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: solicitud de lectura
SRCBLK	Input	ANY	D	Puntero al bloque de datos de la memoria de carga de la que se lee
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: la lectura todavía no ha finalizado.
DSTBLK	Output	ANY	D	Puntero al bloque de datos de la memoria de trabajo en la que se tiene que escribir

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	Sin error
0081	El área de destino es mayor que el área de origen. El área de origen se escribe completamente en el área de destino, el resto de bytes del área de destino no se modifica.
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".
8081	El área de origen es mayor que el área de destino. El área de destino se escribe por completo, el resto de bytes del área de origen se ignoran.
8093	En el parámetro DSTBLK no se ha indicado ningún bloque de datos o se ha indicado uno que no se encuentra en la memoria de trabajo.
80B1	Con el parámetro SRCBLK no se ha indicado ningún bloque de datos o el bloque de datos indicado no es un objeto de la memoria de carga (p. ej. un DB creado mediante "CREAT_DB (Página 7038)").
80B4	El DB con atributo F no debe leerse.
80C0	El DB de destino está siendo procesado por otra instrucción o función de comunicación.
80C3	El número máximo de instrucciones "READ_DBL" activas simultáneamente ya se ha alcanzado en este momento.
8xyy	Códigos de error generales Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

WRIT_DBL: Escribir en un bloque de datos de la memoria de carga

Descripción

Esta instrucción permite transferir el contenido de un DB o un área de DB desde la memoria de trabajo a un DB o un área de DB de la memoria de carga (Micro Memory Card). El DB de origen tiene que ser relevante para la ejecución, es decir, no puede haberse creado con el atributo UNLINKED. Sin embargo, sí puede haber sido creado mediante la instrucción "CREAT_DB".

Para garantizar la coherencia de los datos no debe modificarse el área de origen durante el procesamiento de "WRIT_DBL" (es decir, mientras el parámetro BUSY tenga el valor TRUE).

Para los parámetros SRCBLK y DSTBLK se aplican las siguientes restricciones:

- En el caso de un puntero ANY del tipo BOOL la longitud debe ser divisible entre 8.
- En el caso de un puntero ANY del tipo STRING la longitud debe ser 1.

La longitud del DB de destino puede consultarse, en caso necesario, con la instrucción "TEST_DB (Página 7044)".

"WRIT_DBL" no modifica la suma de verificación del programa de usuario si se escribe en un DB creado mediante instrucción. Por el contrario, al describir un DB cargado, la primera escritura en este DB sí modifica la suma de verificación del programa de usuario.

Nota

"WRIT_DBL" no es adecuada para escribir variables repetidamente (es decir, de modo cíclico) en la memoria de carga, ya que, por motivos técnicos, las Micro Memory Cards solo admiten un número determinado de accesos de escritura.

Funcionamiento

La instrucción "WRIT_DBL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "WRIT_DBL" con REQ =1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WRIT_DBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: solicitud de escritura
SRCBLK	Input	ANY	D	Puntero en el bloque de datos de la memoria de trabajo de la que se lee
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La escritura todavía no ha finalizado.
DSTBLK	Output	ANY	D	Puntero al bloque de datos de la memoria de carga en la que se tiene que escribir.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	Sin error
0081	El área de destino es mayor que el área de origen. El área de origen se escribe completamente en el área de destino, el resto de bytes del área de destino no se modifica.

Código de error (W#16#...)	Descripción
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: transmisión de datos iniciada. BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): Transmisión de datos activa; BUSY tiene el valor "1".
8081	El área de origen es mayor que el área de destino. El área de destino se escribe por completo, el resto de bytes del área de origen se ignoran.
8092	Modo de operación incorrecto: Mientras "WRIT_DBL" estaba activo, la CPU ha pasado a STOP. Cuando vuelva a pasar a RUN, se proporcionará este código de error. Vuelva a llamar "WRIT_DBL".
8093	En el parámetro SRCBLK no se ha indicado ningún bloque de datos o se ha indicado uno que no se encuentra en la memoria de trabajo.
80B1	Con el parámetro DSTBLK no se ha indicado ningún bloque de datos o el bloque de datos indicado no es un objeto de la memoria de carga (p. ej. un DB creado mediante "CREAT_DB (Página 7038)").
80B4	El DB con atributo F no debe modificarse
80C0	El destino está siendo procesado por otra instrucción o función de comunicación. Ejemplo: Se carga un DB de la CPU en la programadora. El contenido de este DB se quiere modificar con "WRIT_DBL".
80C3	El número máximo de instrucciones "WRIT_DBL" activas simultáneamente ya se ha alcanzado en este momento.
8xyy	Códigos de error generales Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

DEL_DB: Borrar bloque de datos

Descripción

Esta instrucción permite borrar un bloque de datos existente en la memoria de trabajo y, en su caso, en la memoria de carga de la CPU. El DB que se va borrar no debe estar abierto en el nivel de ejecución actual ni en otro de menor prioridad. Es decir, no debe estar registrado en ninguno de los dos registros de DB ni en la pila B.

De lo contrario, al llamar la instrucción, la CPU inicia el OB 121. Si el OB 121 no está cargado, la CPU pasa a STOP; con S7-300 (excepción: CPU 318) el DB se borra sin llamar el OB 121.

Nota

Borrar DBs de instancia con "DEL_DB" no es recomendable y provoca siempre errores de programa. Evite siempre borrar DBs de instancia con "DEL_DB".

La tabla siguiente explica cuándo puede borrarse un DB con "DEL_DB".

Si el DB ...	"DEL_DB" ...
se creó llamando la instrucción "CREAT_DB",	permite borrarlo.
se transfirió a la CPU y no se creó con el atributo "Depositar sólo en la memoria de carga"	permite borrarlo.
se creó con el atributo "Depositar sólo en la memoria de carga"	• puede borrarse en la S7-300 • no puede borrarse en la S7-400

Si el DB ...	"DEL_DB" ...
se creó llamando la instrucción "CREA_DBL"	puede borrarse
está almacenado en la Flashcard,	no permite borrarlo.

Posibilidad de interrupción

La instrucción puede ser interrumpida por niveles de ejecución de mayor prioridad. Si se vuelve a llamar la instrucción allí, esa segunda llamada se interrumpe, y en RET_VAL se registra W#16#8091.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DEL_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DB_NUMBER	Input	WORD	I, Q, M, D, L o constante	Número del DB que se va a borrar
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8091	En llamadas de "DEL_DB" anidadas entre sí se ha superado la profundidad de anidamiento máxima de la CPU utilizada.
8092	La función "Borrar un DB" no puede ejecutarse en este momento, porque • está activa la función "Comprimir memoria de usuario" • el DB que se debe borrar se está copiando de la CPU a un proyecto offline. • la CPU con el software WinAC ha detectado un error en el sistema operativo del ordenador en el que está instalado WinAC.
80A1	Error en el parámetro de entrada DB_NUMBER: El parámetro actual seleccionado • tiene el valor "0" • es mayor que el número de DB máximo posible en la CPU utilizada.
80B1	El DB con el número indicado no está presente en la memoria de trabajo de la CPU.
80B3	El DB se encuentra en la Flashcard.
80B4	No se ha podido borrar el DB. Causas posibles: • Pertenece a un programa F. • Es un DB de instancia de un bloque de la comunicación S7 (solo con S7-400). • Es un DB de tecnología.

Código de error (W#16#...)	Explicación
80C1	La función "Borrar un DB" no puede ejecutarse en este momento debido a escasez temporal de recursos.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

CREAT_DB: Generar bloque de datos

Descripción

Esta instrucción permite crear en el programa de usuario un bloque carente de valores preasignados. En su lugar, contiene datos casuales. Esta instrucción crea un bloque de datos con un número perteneciente a un área especificada y con un tamaño definido. La instrucción asigna al DB el número más bajo posible del área especificada. Para crear un DB con un número determinado, debe asignarse el mismo número al límite superior y al límite inferior del área definida. No se pueden asignar los números de los DBs ya contenidos en el programa de usuario. La longitud del DB debe indicarse con un número par.

Posibilidad de interrupción

La instrucción puede ser interrumpida por OBs de mayor prioridad. Si en un OB de mayor prioridad se llama a su vez otra instrucción "CREAT_DB", se rechaza esta llamada con el código de error W#16#8091.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CREAT_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LOW_LIMIT	Input	WORD	I, Q, M, D, L o constante	El límite inferior es el número más bajo del rango de los números que se pueden asignar al bloque de datos.
UP_LIMIT	Input	WORD	I, Q, M, D, L o constante	El límite superior es el número más alto del rango de los números que se pueden asignar al bloque de datos.
COUNT	Input	WORD	I, Q, M, D, L o constante	El valor de contaje indica el número de bytes de datos que se desea reservar para el bloque de datos. Debe especificarse un número par de bytes (máximo 65534).
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
DB_NUMBER	Output	WORD	I, Q, M, D, L	El número de bloque de datos es el número del bloque de datos creado. En caso de error (bit 15 de RET_VAL activado) en DB_NUMBER se registra el valor "0".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Sin error
8091	Se ha llamado CREAT_DB con anidamiento.
8092	La función "Generar bloque de datos" no puede ejecutarse en este momento, porque • está activa la función "Comprimir memoria de usuario" • ya se ha alcanzado el número máximo posible de DBs cargados en la CPU • la CPU H se encuentra en una operación de acoplar y sincronizar. • la CPU con el software WinAC ha detectado un error en el sistema operativo del ordenador en el que está instalado WinAC. • el borrado anterior todavía no ha finalizado.
80A1	Error en el número del DB: • el número es "0" • el número supera el número de DBs específico de la CPU • Límite inferior > límite superior
80A2	Error en la longitud del DB: • la longitud es "0" • se ha indicado la longitud con un número impar • la longitud es mayor que la admisible en la CPU
80B1	No hay ningún número de DB libre.
80B2	No hay suficiente espacio de almacenamiento libre.
80B3	No hay suficiente espacio de almacenamiento libre conexo. (Realizar compresión)
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

CREA_DB: Generar bloque de datos / remanente

Descripción

Esta instrucción permite crear en el programa de usuario un bloque carente de valores preasignados. En su lugar, contiene datos casuales. Esta instrucción crea un bloque de datos con un número perteneciente a un área especificada y con un tamaño definido. La instrucción asigna al DB el número más bajo posible del área especificada. Para crear un DB con un número determinado, debe asignarse el mismo número al límite superior y al límite inferior del área definida. No se pueden asignar los números de los DBs ya contenidos en el programa de usuario. La longitud del DB debe indicarse con un número par.

En función de la selección del parámetro ATTRIB , el DB generado tiene la propiedad RETAIN o NON_RETAIN:

- RETAIN (= remanente) significa que el DB se genera en la parte remanente de la memoria de trabajo, es decir, los valores actuales del DB se conservan en cada transición Power OFF-Power ON y en cada rearranque completo (en caliente).
- NON_RETAIN (= no remanente) significa que el DB se genera en la parte no remanente de la memoria de trabajo, es decir, los valores actuales del DB no están definidos tras cada transición Power OFF-Power ON y en cada rearranque completo (en caliente).

En el caso de que no se distinga entre la memoria de trabajo remanente y no remanente, el parámetro ATTRIB se ignora, es decir, los valores del DB se conservan después de cada transición Power OFF-Power ON y después de cada rearranque completo (en caliente).

Posibilidad de interrupción

La instrucción "CREA_DB" puede ser interrumpida por OBs de mayor prioridad. Si en un OB de mayor prioridad se llama a su vez otra instrucción "CREA_DB", se rechaza esta llamada con el código de error W#16#8091 .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CREA_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LOW_LIMIT	Input	WORD	I, Q, M, D, L o constante	El límite inferior es el número más bajo del rango de los números que se pueden asignar al bloque de datos.
UP_LIMIT	Input	WORD	I, Q, M, D, L o constante	El límite superior es el número más alto del rango de los números que se pueden asignar al bloque de datos.
COUNT	Input	WORD	I, Q, M, D, L o constante	El valor de contaje indica el número de bytes de datos que se desea reservar para el bloque de datos. Debe especificarse un número par de bytes (máximo 65534).
ATTRIB	Input	BYTE	I, Q, M, D, L o constante	Atributos del DB: • B#16#00: RETAIN • B#16#04: NON_RETAIN
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
DB_NUMBER	Output	WORD	I, Q, M, D, L	El número de bloque de datos es el número del bloque de datos creado. En caso de error (bit 15 de RET_VAL activado) en DB_NUMBER se registra el valor "0".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	Sin error
8091	Se ha llamado CREA_DB con anidamiento.
8092	La función "Generar bloque de datos" no puede ejecutarse en este momento, porque • está activa la función "Comprimir memoria de usuario" • la CPU con el software WinAC ha detectado un error en el sistema operativo del ordenador en el que está instalado WinAC.
8094	Valor no admisible en ATTRIB
80A1	Error en el número del DB: • el número es "0" • el número supera el número de DBs específico de la CPU • Límite inferior > límite superior
80A2	Error en la longitud del DB: • La longitud es "0" • se ha indicado la longitud con un número impar • la longitud es mayor que la admisible en la CPU
80B1	No hay ningún número de DB libre.
80B2	No hay suficiente espacio de almacenamiento libre.
80B3	No hay suficiente espacio de almacenamiento libre conexo. (Realizar compresión)
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

CREA_DBL: Generar bloque de datos en la memoria de carga

Descripción

Esta instrucción permite crear un bloque de datos nuevo en la memoria de carga (Micro Memory Card). Esta instrucción crea un bloque de datos con un número perteneciente a un área especificada y con un tamaño definido. La instrucción asigna al DB el número más bajo posible del área especificada. Para crear un DB con un número determinado, debe asignarse el mismo número al límite superior y al límite inferior del área definida. No se pueden asignar los números de los DBs ya contenidos en el programa de usuario. Si ya existe en la memoria de trabajo y/o en la memoria de carga un DB con el mismo número, o el DB existe ya como variante copiada, la instrucción finaliza y se devuelve una información de error.

Nota

En caso necesario, puede consultarse con la instrucción "TEST_DB (Página 7044)" si ya existe un DB con el mismo número.

El DB se rellena con el contenido del área de datos a la que apunta el parámetro SRCBLK(source block). Esta área de datos debe ser un DB o un área de un DB. Para

garantizar la coherencia de los datos, no se debe modificar esa área de datos durante el procesamiento de "CREA_DBL" (es decir, mientras el parámetro BUSY tenga el valor TRUE).

Un DB con el atributo READ_ONLY solo se puede crear e inicializar mediante "CREA_DBL".

"CREA_DBL" no modifica la suma de verificación del programa de usuario.

Funcionamiento

La instrucción "CREA_DBL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "CREA_DBL" con REQ=1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CREA_DBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción	
REQ	Input	BOOL	I, Q, M, D, L	REQ = 1: Solicitud de creación del DB	
LOW_LIMIT	Input	WORD	I, Q, M, D, L	Límite inferior del área de la cual "CREA_DBL" extrae el número que se asigna al DB	
UP_LIMIT	Input	WORD	I, Q, M, D, L	Límite superior del área de la cual "CREA_DBL" extrae el número que se asigna al DB	
COUNT	Input	WORD	I, Q, M, D, L	El valor de contaje indica el número de bytes de datos que se desea reservar para el DB. Debe especificarse un número par de bytes.	
ATTRIB	Input	BYTE	I, Q, M, D, L	Propiedades del DB:	
				Bit 0 = 1:	UNLINKED: El DB se encuentra solo en la memoria de carga.
				Bit 1 = 1:	READ_ONLY: El DB está protegido contra escritura.
				Bit 2 = 1:	NON_RETAIN: El DB no es remanente.
				Bits de 3 a 7:	Reservado
SRCBLK	Input	ANY	D	Puntero al bloque de datos con cuyos valores se inicializará el bloque de datos que se va a crear	
RET_VAL	Return	INT	I, Q, M, D, L	Información de error	
BUSY	Output	BOOL	I, Q, M, D, L	BUSY = 1: La operación todavía no ha finalizado.	
DB_NUM	Output	WORD	I, Q, M, D, L	Número del DB creado	

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Descripción
0000	Sin error
0081	El área de destino es mayor que el área de origen. El área de origen se escribe por completo en el área de destino, y el resto de los bytes del área de destino se rellenan con 0.
7000	Primera llamada con REQ=0: Ninguna transmisión de datos activa; BUSY tiene el valor "0".
7001	Primera llamada con REQ=1: Transmisión de datos iniciada; BUSY tiene el valor "1".
7002	Llamada intermedia (REQ irrelevante): transferencia de datos ya activa; BUSY tiene el valor "1"
8081	El área de origen es mayor que el área de destino. El área de destino se escribe por completo, el resto de bytes del área de origen se ignoran.
8091	Se ha llamado "CREA_DBL" con anidamiento.
8092	La función "Generar bloque de datos" no puede ejecutarse en este momento, porque • está activa la función "Comprimir memoria de usuario"; • se ha alcanzado ya el número máximo de bloques admisible en la CPU
8093	En el parámetro SRCBLK no se ha indicado ningún bloque de datos o se ha indicado uno que no se encuentra en la memoria de trabajo.
8094	En el parámetro ATTRIB se ha indicado un atributo todavía no soportado.
80A1	Error en el número del DB: • el número es 0 o no está dentro del rango válido de la CPU (CPU15xx: 60001 ... 60999) • el número supera el número de DBs específico de la CPU • Límite inferior > límite superior
80A2	Error en la longitud del DB: • La longitud es "0" • La longitud es un número impar • La longitud es mayor que la admisible en la CPU
80B1	No hay ningún número de DB libre
80B2	Memoria de trabajo insuficiente
80BB	Memoria de carga insuficiente
80C0	El destino está siendo procesado por otra instrucción o por una función de comunicación.
80C3	El número máximo de instrucciones "CREA_DBL" activas simultáneamente ya se ha alcanzado en este momento.
8xyy	Códigos de error generales, p. ej.: • El DB de origen no existe o existe solo como variante copiada • El área de origen no existe en el DB Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

TEST_DB: Testear bloque de datos

Descripción

Con S7-300, la instrucción devuelve información sobre un bloque de datos que se encuentra en la memoria de trabajo o la memoria de carga de la CPU y, con S7-400, devuelve información sobre un bloque de datos que se encuentra en la memoria de trabajo de la CPU. Esta instrucción calcula el número de bytes de datos correspondientes al DB seleccionado y comprueba si está protegido contra escritura.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TEST_DB":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DB_NUMBER	Input	WORD	I, Q, M, D, L o constante	Número del DB que se va a probar
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
DB_LENGTH	Output	WORD	I, Q, M, D, L	Número de bytes de datos que contiene el DB seleccionado.
WRITE_PROT	Output	BOOL	I, Q, M, D, L	Información sobre el identificador de protección contra escritura del DB seleccionado ("1" significa protegido contra escritura).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
80A1	Error en el parámetro de entrada DB_NUMBER: El parámetro actual seleccionado • tiene el valor "0" • es mayor que el número de DB máximo posible en la CPU utilizada.
80B1	El DB con el número indicado no está cargado en la CPU.
80B2	El DB ha sido creado con la palabra clave UNLINKED.
8xyy	Información de error general, Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Funciones de tabla

ATT: Introducir valor en tabla

Descripción

Esta instrucción complementa el parámetro DATA como siguiente entrada de una tabla e incrementa en uno el número de entradas. La tabla está compuesta de palabras. Esta instrucción permite crear entradas de tabla que son utilizadas por "FIFO (Página 7046)" y "LIFO (Página 7050)".

- La primera entrada de una tabla "FIFO (Página 7046)" o "LIFO (Página 7050)" indica la longitud máxima de la tabla.
- La segunda entrada de una tabla indica el número de entradas que contiene la tabla.
- La tercera entrada de la tabla contiene la primera palabra de datos.

Nota

Al crear la tabla deben inicializarse las dos primeras entradas.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ATT":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
DATA	Input	WORD	I, Q, M, D, L, P o constante	Datos que deben introducirse en la tabla.
TABLE	Input	*Puntero	I, Q, M, D	Apunta a la dirección inicial de la tabla "FIFO" o "LIFO".
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

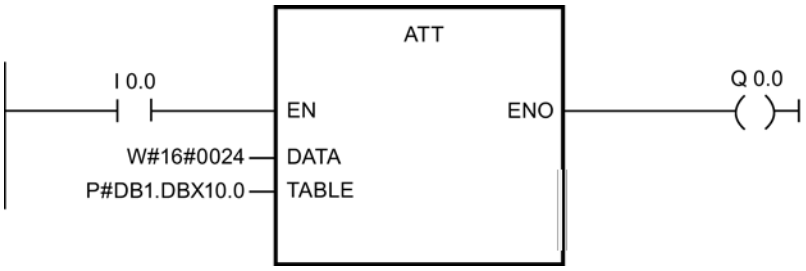
Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Si el número de entradas es igual o menor que la longitud de la tabla, los datos no se introducen en la tabla y el estado lógico de "BR" se ajusta a "0".

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "ATT" se ejecuta. En el ejemplo, DATA se introduce en la tabla como quinta entrada, y el número de entrada se incrementa de 4 a 5.

Si la instrucción "ATT" se ejecuta sin errores, los estados lógicos de "ENO" y Q 0.0 se ajustan a "1".



Antes de la ejecución:

TABLE (longitud de tabla)	DBW10 = W#16#0006
Número de entradas	DBW12 = W#16#0004 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0000 DBW24 = W#16#0000

Después de la ejecución:

TABLE (longitud de tabla)	DBW10 = W#16#0006
Número de entradas	DBW12 = W#16#0005 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000

FIFO: Emitir primer valor de la tabla

Descripción

Esta instrucción devuelve la entrada más antigua de la tabla "FIFO" como valor de función. El número se entrada se reduce en 1. Si todavía existen entradas en la tabla, se desplazan hacia abajo. La tabla "FIFO" está compuesta de palabras. Con "ATT (Página 7045)" se pueden introducir valores en la tabla "FIFO".

- La primera entrada de una tabla indica la longitud máxima de la tabla.
- La segunda entrada de una tabla indica el número de entradas que contiene la tabla.
- La tercera entrada de la tabla contiene la primera palabra de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "FIFO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TABLE	Input	*Puntero	I, Q, M, D	Apunta a la dirección inicial de la tabla "FIFO".
RET_VAL	Return	WORD	I, Q, M, D, L, P	La entrada más antigua de la tabla "FIFO".
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

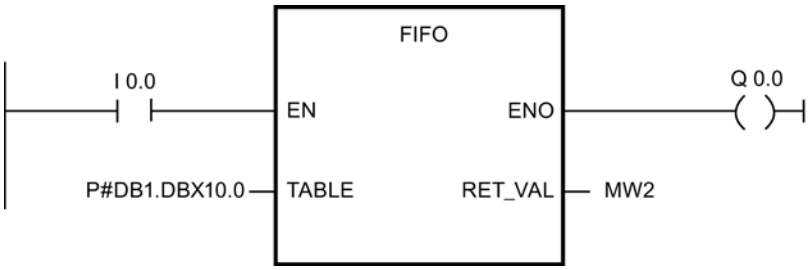
Parámetro RET_VAL

Si la tabla "FIFO" está vacía (número de entradas = 0), RET_VAL no se modifica y el estado lógico de "BR/ENO" se ajusta a "0".

Ejemplo (KOP)

Con estado lógico de la entrada I 0.0 = "1" (activado), se ejecuta "FIFO". En el ejemplo, se emite la entrada más antigua de la tabla como valor de función (MW 2.0). El número de entradas se reduce de 5 a 4, y las entradas restantes de la tabla se desplazan hacia abajo.

Si "FIFO" se ejecuta sin errores, los estados lógicos de "BR/ENO" y Q 0.0 se ajustan a "1".



Antes de la ejecución:

TABLE (longitud de tabla)	DBW10 = W#16#0006
Número de entradas	DBW12 = W#16#0005 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0000

Después de la ejecución:

TABLE (longitud de tabla)	DBW10 = W#16#0006
Número de entradas	DBW12 = W#16#0004 DBW14 = W#16#0029 DBW16 = W#16#0090 DBW18 = W#16#0002 DBW20 = W#16#0024 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0012

TBL_FIND: Buscar valor en tabla

Descripción

Esta instrucción permite buscar expresiones en la memoria. La instrucción "TBL_FIND" efectúa una comparación (parámetro CMD) entre la expresión de origen (parámetro PATRN) y las entradas de la tabla de origen (parámetro SRC). Se busca la primera entrada de la tabla, a partir de la entrada indicada en el parámetro INDX, que cumpla los criterios de comparación. El número de la entrada se guarda en el parámetro INDX. Si no se encuentra ninguna coincidencia, el parámetro INDXapunta más allá del final de la tabla y se desactiva la salida de la instrucción "TBL_FIND".

- Si CMD = 1, la instrucción busca "TBL_FIND" el primer valor que coincida con el valor indicado en el parámetro PATRN.
- Si CMD = 2, la instrucción busca "TBL_FIND" el primer valor que no coincida con el valor indicado en el parámetro PATRN .
- La primera entrada de la tabla indica la longitud máxima de la tabla.
- La segunda entrada de la tabla contiene el primer valor de la tabla.

Nota

Debe inicializarse la primera entrada (longitud) de la tabla.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TBL_FIND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC	Input	*Puntero	I, Q, M, D	Apunta a la dirección inicial de la tabla.
PATRN	Input	*Puntero	I, Q, M, D	Apunta a la expresión que se desea buscar.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BYTE	I, Q, M, D, L, P	Indica el comando: • B#16#01 = igual • B#16#02 = diferente
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla: B#16#02 = BYTE B#16#04= WORD B#16#05=INT B#16#06=DWORD B#16#07=DINT B#16#08=REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000 si la operación se ha llevado a cabo sin errores.
INDX	InOut	WORD	I, Q, M, D, L	Índice de la tabla, que contiene la siguiente información: • Entrada: Número de la entrada por la que se debe iniciar la búsqueda. • Salida: Número de la entrada que coincide con el patrón especificado.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Si se cumple una de las siguientes condiciones, los valores de la tabla no se modifican. El estado lógico de "BR/ENO" se ajusta a "0" y el valor de retorno se ajusta como corresponde:

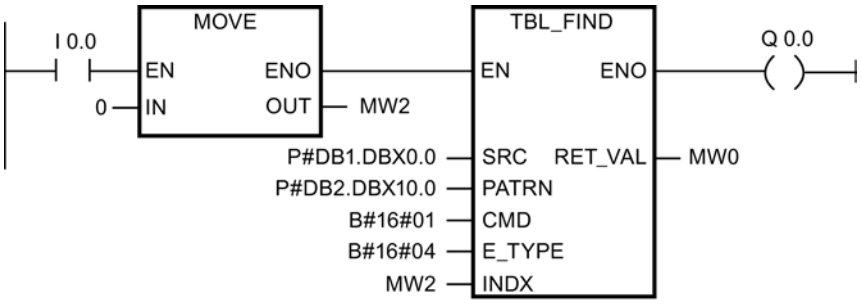
RET_VAL	Explicación
W#16#0007	El valor del parámetro INDX es mayor que el valor de longitud de la tabla.
W#16#0008	No se ha encontrado ninguna coincidencia.
W#16#0009	El parámetro E_TYPE y/o CMD es/son válido(s).

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "TBL_FIND" se ejecuta. En el ejemplo, los datos de la tabla se guardan en palabras que empiezan en la dirección a la que apunta el parámetro SRC , porque el parámetro E_TYPE es igual a "4". Estas palabras se comparan con la expresión "5555", que se guarda en la dirección indicada por el parámetro PATRN . Dado que CMD = 1, se busca el primer valor del parámetro SRC que coincide con la expresión. El parámetro INDX apunta a la dirección en la que se desea iniciar la búsqueda. Una vez procesada la operación, el parámetro INDX indica el número de la entrada de la tabla en la que se ha encontrado una coincidencia con la expresión.

Si la instrucción "TBL_FIND" se ejecuta sin errores, los estados lógicos de ENO y Q 0.0 se ajustan a "1", y RET_VAL se ajusta al valor W#16#0000.

La instrucción "MOVE" se utiliza únicamente para restablecer el valor de "MW2" en cada ciclo.



Antes de la ejecución:

SRC (longitud de tabla)	DBW0 = W#16#0004 DBW2 = W#16#1111 DBW4 = W#16#3333 DBW6 = W#16#5555 DBW8 = W#16#7777	La primera entrada de la tabla indica la longitud de la operación de búsqueda. El valor 0004 indica que se examinarán las cuatro próximas entradas.
INDX	MW2 = W#16#0000	INDX se transfiere de DW0 (antes del procesamiento) a DW6 (después del procesamiento), porque la expresión de DW6 es igual a la expresión de 5555 de PATRN .
PATRN	DBW10 = W#16#5555	

Después de la ejecución:

INDX	MW2 = W#16#0003	

LIFO: Emitir último valor de la tabla

Descripción

Esta instrucción devuelve la entrada más reciente de la tabla "LIFO" como valor de función. El número se entrada se reduce en 1. La tabla "LIFO" está compuesta de palabras. Con la instrucción "ATT (Página 7045)" se pueden introducir valores en la tabla "LIFO".

- La primera entrada de la tabla indica el número máximo admisible de entradas de la tabla (longitud de la tabla).
- La segunda entrada de una tabla indica el número de entradas que contiene la tabla.
- La tercera entrada de la tabla contiene la primera palabra de datos.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LIFO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TABLE	Input	*Puntero	I, Q, M, D	Apunta a la dirección inicial de la tabla "LIFO".
RET_VAL	Return	WORD	I, Q, M, D, L, P	La entrada más reciente de la tabla "LIFO".
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

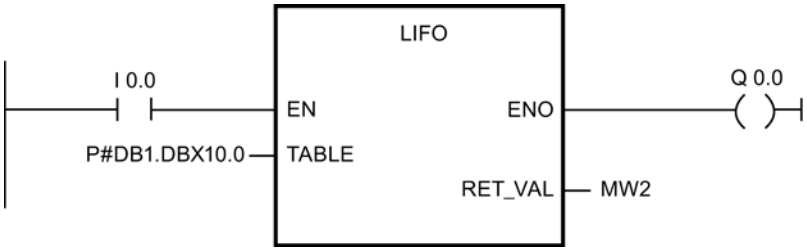
Parámetro RET_VAL

Si la tabla "LIFO" está vacía (número de entradas = 0), RET_VAL no se modifica y el estado lógico de "BR/ENO" se ajusta a "0".

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "LIFO" se ejecuta. En el ejemplo, se emite la entrada más reciente de la tabla "LIFO" como valor de función (MW 2.0). El número de entradas se reduce de 5 a 4.

Si "LIFO" se ejecuta sin errores, los estados lógicos de "ENO" y Q 0.0 se ajustan a "1".



Antes de la ejecución:

TABLE (longitud de tabla)	DBW10 = W#16#0006
Número de entradas	DBW12 = W#16#0005 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0000

Después de la ejecución:

TABLE (longitud de tabla)	DBW10 = W#16#0006
Número de entradas	DBW12 = W#16#0004 DBW14 = W#16#0012 DBW16 = W#16#0029 DBW18 = W#16#0090 DBW20 = W#16#0002 DBW22 = W#16#0024 DBW24 = W#16#0000
RET_VAL	MW2 = W#16#0024

TBL: Ejecutar operación de tabla

Descripción

Esta instrucción ejecuta la operación indicada (parámetro CMD) con la tabla de origen y guarda el resultado en la misma entrada de la tabla.

- La primera entrada de la tabla indica el número máximo admisible de entradas de la tabla (longitud de la tabla).
- La segunda entrada de la tabla contiene el primer valor de la tabla.
- Si el parámetro E_TYPE se ajusta a REAL , el valor de CMD no es válido para el complemento a 1.

Nota

Al crear la tabla debe inicializarse la primera entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC	Input	*Puntero	I, Q, M, D	Apunta a la dirección inicial de la tabla.
CMD	Input	BYTE	I, Q, M, D, L, P	Indica la operación que se desea ejecutar. Son válidas las siguientes operaciones: B#16#03 = Complemento a 1 B#16#04 = Borrar B#16#05 = Negar B#16#06 = Raíz cuadrada

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla. Son posibles los siguientes tipos de datos: B#16#04 = WORD B#16#05 = INT B#16#06 = DWORD B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000 si la operación se ha llevado a cabo sin errores.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

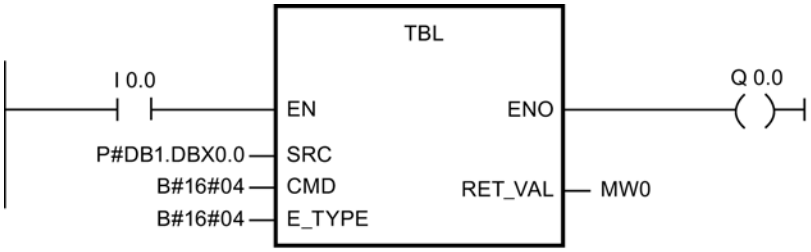
Parámetro RET_VAL

Si el parámetro CMD o el parámetro E_TYPE no son válidos y/o el parámetro CMD y el parámetro E_TYPE no coinciden, los valores de la tabla no se modificarán. El estado lógico de BR/ENO se ajusta a "0" y RET_VAL se ajusta al valor W#16#0008 :

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "TBL" se ejecuta. En el ejemplo, el parámetro SRC apunta a la dirección, dentro del bloque de datos, que la operación debe procesar. Dado que el parámetro E_TYPE = 4, los datos de la tabla se guardan en las palabras que empiezan en la dirección indicada por el parámetro SRC. Dado que CMD = 4 (borrar), cuando se ejecuta la instrucción TBL se borran (se ajustan a "0") todas las palabras de la tabla. Dado que en la primera entrada de la tabla se indica una longitud de "5", se borran las cinco primeras direcciones de la tabla.

Si la instrucción "TBL" se ejecuta sin errores, los estados lógicos de ENO y Q 0.0 se ajustan a "1", y RET_VAL se ajusta al valor W#16#0000.



Antes de la ejecución:

SRC (longitud de tabla)	DBW0 = W#16#0005 DBW2 = W#16#2000 DBW4 = W#16#3000 DBW6 = W#16#4000 DBW8 = W#16#5000 DBW10 = W#16#6000

Después de la ejecución:

SRC (longitud de tabla)	DBW0 = W#16#0005 DBW2 = W#16#0000 DBW4 = W#16#0000 DBW6 = W#16#0000 DBW8 = W#16#0000 DBW10 = W#16#0000

TBL_WRD: Copiar valor de la tabla

Descripción

La instrucción "TBL_WRD" (Copiar valor de la tabla) copia de la tabla SRC en la salida DEST la entrada que se indica en el parámetro INDX. Al hacerlo, el valor de INDX se incrementa, si es inferior a la longitud máxima que se indica en la primera palabra de la tabla (SRC[0]). Si el parámetro INDX indica la última entrada de la tabla, el bit de la salida se ajusta a "0" después de la ejecución.

- La primera entrada de la tabla indica el número máximo admisible de entradas de la tabla (longitud de la tabla).
- La segunda entrada de la tabla contiene el primer valor de la tabla.

Nota

Al crear la tabla debe inicializarse la primera entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TBL_WRD":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC	Input	*Pointer	I, Q, M, D	Apunta a la dirección inicial de la tabla.
DEST	Input	*Pointer	I, Q, M, D	Apunta a la dirección inicial del elemento de destino.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla. Son posibles los siguientes tipos de datos: • B#16#04 = WORD • B#16#05 = INT • B#16#06 = DWORD • B#16#07 = DINT • B#16#08 = REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000si la operación se ha llevado a cabo sin errores.
Q	Output	BOOL	Q, M, D, L	Indica el valor "0" si, al llamar "TBL_WRD", la variable INDXcontiene el último valor de la tabla.
INDX	InOut	WORD	I, Q, M, L	Número de la entrada que se desea copiar.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

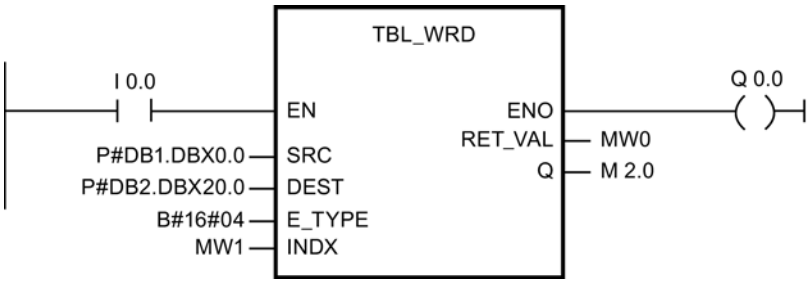
Si se cumple una de las siguientes condiciones, "TBL_WRD" no se ejecuta. El estado lógico de ENO se ajusta a "0" y el valor de retorno se ajusta como corresponde:

RET_VAL	Explicación
W#16#0007	El índice es igual a "0".
W#16#0008	E_TYPE no es válido.
W#16#0009	El índice se encuentra fuera del rango de la tabla.

Ejemplo

Con estado lógico de la entrada I 0.0 = "1" (activado), se ejecuta "TBL_WRD". Dado que en el ejemplo E_TYPE = 4, en la entrada indicada por el parámetro DEST se copia una palabra de datos que se guarda en la entrada de la tabla a la que apunta el parámetro SRC. El valor de INDX apunta a la entrada de la tabla que se desea copiar. Una vez ejecutada correctamente "TBL_WRD", el valor de INDX se incrementa automáticamente en una entrada. Si, al llamar TBL_WRD, el parámetro INDX no contiene la última entrada de la tabla, la salida del parámetro ENO se ajusta a "1" después de la ejecución.

Si "TBL_WRD" se ejecuta sin errores, el estado lógico de ENO se ajusta a "1" y RET_VAL se ajusta al valor W#16#0000.



Antes de la ejecución:

SRC (longitud de tabla)	DBW0 = W#16#0004 DBW2 = W#16#2000 DBW4 = W#16#3000 DBW6 = W#16#4000 DBW8 = W#16#5000
INDX	MW1 = W#16#0001
DEST	DBW20 = W#16#0000

Antes de la ejecución:

INDX	MW1 = W#16#0002
DEST	DBW20 = W#16#2000

WRD_TBL: Combinar un valor lógicamente con un elemento de la tabla y guardarlo

Descripción

Esta instrucción ejecuta la operación indicada (parámetro CMD) con el elemento de origen y la entrada de la tabla en el offset que se indica en el parámetro INDX. La operación solo se ejecuta mientras el parámetro INDX sea menor que la longitud de la tabla, que se guarda en la primera palabra de la tabla.

- La primera entrada de la tabla indica el número máximo admisible de entradas de la tabla (longitud de la tabla).
- La segunda entrada de la tabla contiene el primer valor de la tabla.
- Si el parámetro E_TYPE es del tipo de datos REAL, el parámetro CMD solo puede ser "Transferir".

Nota

Al crear la tabla debe inicializarse la primera entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "WRD_TBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SRC	Input	*Puntero	I, Q, M, D	Apunta a los datos de origen
TABLE	Input	*Puntero	I, Q, M, D	Apunta al inicio de la tabla.
CMD	Input	BYTE	I, Q, M, D, L, P	Apunta a la operación que se desea ejecutar. Son posibles las siguientes operaciones: B#16#0E = Transferir B#16#07 = Operación lógica Y B#16#08 = Operación lógica O B#16#09 = Operación lógica O-exclusiva
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla. Son posibles los siguientes tipos de datos: B#16#04 = WORD B#16#05 = INT B#16#06 = DWORD B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000 si la operación se ha llevado a cabo sin errores
Q	Output	BOOL	Q, M, D, L	Indica el valor "0" si la variable del parámetro INDX contiene el número de la última entrada de la tabla.
INDX	InOut	WORD	I, Q, M, D, L	Número de la entrada con la que se desea ejecutar la operación.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

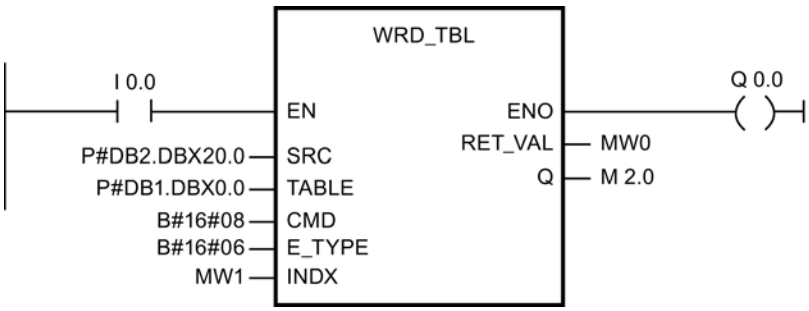
Si se cumple una de las siguientes condiciones, la instrucción "WRD_TBL" no se ejecuta. El estado lógico de ENO se ajusta a "0" y el valor de retorno se ajusta como corresponde:

RET_VAL	Explicación
W#16#0007	El índice es igual a 0.
W#16#0008	El parámetro CMD o el parámetro E_TYPE no son válidos o el parámetro CMD y el parámetro E_TYPE no coinciden.
W#16#0009	El índice se encuentra fuera del rango de la tabla.

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "WRD_TBL" se ejecuta. Dado que, en el ejemplo, el parámetro E_TYPE es igual a "6", la palabra doble de los datos de la tabla se guarda en la dirección a la que apunta el parámetro TABLE . La primera palabra de la tabla indica que la tabla contiene tres palabras dobles. El valor de INDX apunta a la entrada de la tabla que se desea procesar. Dado que CMD = 8, se ejecuta una operación lógica O con el valor al que apunta el parámetro INDX . Dado que INDX = 2, la segunda palabra (66665544) se combina mediante operación lógica O con el valor al que apunta el parámetro SRC (11111111). Una vez ejecutada la operación, el resultado de la operación lógica O (77775555) se escribe en la tabla, y el valor de INDX se incrementa automáticamente en una entrada. Si el parámetro INDX apunta a la última entrada de una tabla, el bit de la salida "Q" se ajusta a "0" después de la ejecución de la operación. En el ejemplo, el valor de INDX no contiene la última entrada de la tabla, por lo que la salida "Q" se ajusta a "1" después de la ejecución.

Si la instrucción "WRD_TBL" se ejecuta sin errores, los estados lógicos de ENO y Q 0.0 se ajustan a "1", y RET_VAL se ajusta al valor W#16#0000.



Antes de la ejecución:

SRC	DBD20 = DW#16#11111111
TABLE (longitud de tabla)	DBW0 = W#16#0003 DBD2 = DW#16#99998877 DBD6 = DW#16#66665544 DBD10 = DW#16#33332222
INDX	MW1 = W#16#0002

Después de la ejecución:

TABLE (longitud de tabla)	DBW0 = W#16#0003 DBD2 = DW#16#99998877 DBD6 = DW#16#77775555 DBD10 = DW#16#33332222
INDX	MW1 = W#16#0003

DEV: Calcular desviación estándar

Descripción

La instrucción calcula la desviación estándar de un grupo de valores guardados en una tabla (instrucción "TBL (Página 7052)"). El resultado se guarda en OUT . La desviación estándar se calcula con la siguiente fórmula:

$$\text{Desviación estándar} = \sqrt{\frac{(N * SumaCu) - Suma^2}{N * (N - 1)}}$$

Donde:

- Suma = suma de los valores de la instrucción "TBL (Página 7052)"
- N= número de valores de la instrucción "TBL (Página 7052)"
- SumaCu = suma de todos los valores de la instrucción "TBL (Página 7052)" al cuadrado

Todos los cálculos utilizan valores en coma flotante IEEE, y en caso necesario la instrucción "DEV" convierte automáticamente los tipos de datos.

- La primera entrada de la tabla indica el número de entradas (longitud de la tabla).
- La segunda entrada de la tabla contiene el primer valor de la tabla.
- El tamaño de las entradas de la tabla y el valor calculado (parámetro OUT) se definen en el parámetro E_TYPE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "DEV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TBL	Input	*Puntero	I, Q, M, D	**Apunta a la dirección inicial de la tabla con los valores.
OUT	Input	*Puntero	I, Q, M, D	**Apunta a la dirección que contiene el valor calculado para la desviación estándar.
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla. Son válidos los siguientes tipos de datos: B#16#05 = INT B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000 si la operación se ha llevado a cabo sin errores.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas ** Los bloques de datos de origen y destino deben ser idénticos				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

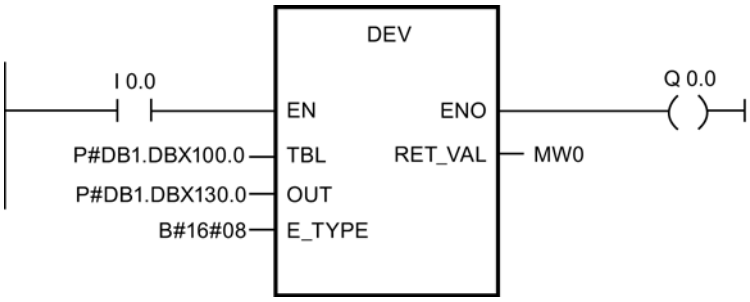
Si se cumple una de las siguientes condiciones, la instrucción "DEV" no se ejecuta. El estado lógico de BR/ENO se ajusta a "0" y el valor de retorno se ajusta como corresponde:

RET_VAL	Explicación
W#16#0001	Se ha indicado un área de memoria no válida para un parámetro de "DEV".
W#16#0002	El parámetro E_TYPE no es válido.
W#16#0004	La longitud de la tabla es igual a cero.

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "DEV" se ejecuta. En el ejemplo hay cinco valores de la tabla. Así lo indica la primera palabra de la tabla. Los valores de la tabla son del tipo de datos REAL. Así lo indica el parámetro E_TYPE.

Si la instrucción "DEV" se ejecuta sin errores, los estados lógicos de ENO y Q 0.0 se ajustan a "1", y RET_VAL se ajusta a "W#16#0000".



Antes de la ejecución:

TBL (longitud de tabla)	DBW100 = W#16#0005 DBD102 = 2.0 DBD106 = 4.0 DBD110 = 8.0 DBD114 = 16.0 DBD118 = 32.0
OUT	DBD130 = 0.0

Después de la ejecución:

OUT	DBD130 = 12.19836055

CDT: Tablas de datos correlativas

Descripción

Esta instrucción compara un valor de entrada (parámetro IN) con una tabla ya existente con valores de entrada (parámetro IN_TBL) y busca el primer valor mayor o igual que el valor de entrada. Tras ello, usando el índice del valor encontrado, se copia el valor en el valor de salida correspondiente (parámetro OUT) en la tabla de valores de salida (parámetro OUT_TBL).

- Los valores de la tabla de entrada deben estar consignados en orden ascendente. Es decir: el valor más bajo se encuentra en la primera entrada de la tabla, y el mayor en la última.
- El tamaño del valor de entrada, los valores de la tabla y los valores de salida se definen en el parámetro E_TYPE.
- La primera entrada de la tabla indica el número de entradas (longitud de la tabla).
- La segunda entrada de la tabla contiene el primer valor de la tabla.
- El número de entradas debe ser idéntico en las dos tablas y mayor que cero.

Nota

Al crear las tablas debe inicializarse en cada una la primera entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "CDT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IN_TBL	Input	*Puntero	I, Q, M, D	Apunta al inicio de la tabla de entrada.
OUT_TBL	Input	*Puntero	I, Q, M, D	Apunta al inicio de la tabla de salida.
IN	Input	*Puntero	I, Q, M, D	Apunta a la tabla de entrada.
OUT	Input	*Puntero	I, Q, M, D	Apunta a la tabla de salida.
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla. Son válidos los siguientes tipos de datos: B#16#05 = INT B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000 si la operación se ha llevado a cabo sin errores.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

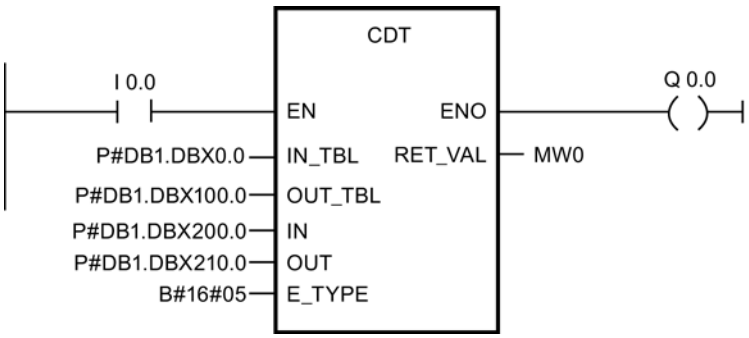
Si se cumple una de las siguientes condiciones, CDT no se ejecuta. El estado lógico de BR/ ENO se ajusta a "0" y el valor de retorno se ajusta como corresponde:

RET_VAL	Explicación
W#16#0001	Se ha indicado un área de memoria no válida para un parámetro de CDT .
W#16#0002	El parámetro E_TYPE no es válido.
W#16#0003	Las longitudes de las tablas de entrada y de salida no coinciden.
W#16#0004	La tabla tiene una longitud igual a cero.
W#16#0007	El parámetro IN_TBL no contiene ningún valor mayor o igual que el valor de entrada.

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "CDT" se ejecuta. En el ejemplo, los parámetros IN_TBL y OUT_TBL contienen cada uno cinco entradas de tabla. Así lo indica la primera palabra de cada una de las tablas. El tipo de datos de los valores de la tabla es INTEGER. Así lo indica el parámetro E_TYPE. El valor del parámetro IN es 22. El valor del parámetro IN_TBL, mayor o igual que "22", es "64", a lo que corresponde el índice "5". El valor correspondiente de OUT_TBL es "25". Por eso el valor "25" se escribe en el parámetro OUT.

Si la instrucción "CDT" se ejecuta sin errores, los estados lógicos de ENO y Q 0.0 se ajustan a "1", y RET_VAL se ajusta a W#16#0000 .



Antes de la ejecución:

IN_TBL (longitud de tabla)	DBW0 = W#16#0005 DBW2 = 2 DBW4 = 4 DBW6 = 8 DBW8 = 16 DBW10 = 64
OUT_TBL (longitud de tabla)	DBW100 = W#16#0005 DBW102 = 5 DBW104 = 10 DBW106 = 15 DBW108 = 20 DBW110 = 25
IN	DBW200 = 22
OUT	DBW210 = 0

Después de la ejecución:

OUT DBW210 = 25

TBL_TBL: Enlazar tablas

Descripción

Esta instrucción ejecuta el comando indicado en el parámetro CMD con las correspondientes entradas de las dos tablas de origen (parámetros TBL1 y TBL2) y escribe el resultado en las correspondientes entradas de la tabla de destino (DEST_TBL).

- Los tipos de datos INT, DINT y REAL solo son válidos para operaciones aritméticas.
- La primera entrada de la tabla indica el número de entradas (longitud de la tabla).
- El número de entradas debe ser en todas las tablas igual o mayor que cero.

Nota

Al crear las tablas debe inicializarse en cada una la primera entrada.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TBL_TBL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
TBL1	Input	*Puntero	I, Q, M, D	Apunta al inicio de la primera tabla de origen.
TBL2	Input	*Puntero	I, Q, M, D	Apunta al inicio de la segunda tabla de origen.
DEST_TBL	Input	*Puntero	I, Q, M, D	Apunta al inicio de la tabla de destino.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CMD	Input	BYTE	I, Q, M, D, L, P	Indica el comando que se va a ejecutar: Son válidos los siguientes comandos: B#16#07 = Operación lógica Y B#16#08 = Operación lógica O B#16#09 = Operación lógica O-exclusiva B#16#0a = Sumar B#16#0b = Restar B#16#0c = Multiplicar B#16#0d = Dividir
E_TYPE	Input	BYTE	I, Q, M, D, L, P	Indica el tipo de datos de las entradas de la tabla. Son válidos los siguientes tipos de datos: B#16#04 = WORD B#16#05 = INT B#16#06 = DWORD B#16#07 = DINT B#16#08 = REAL
RET_VAL	Return	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000 si la operación se ha llevado a cabo sin errores.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

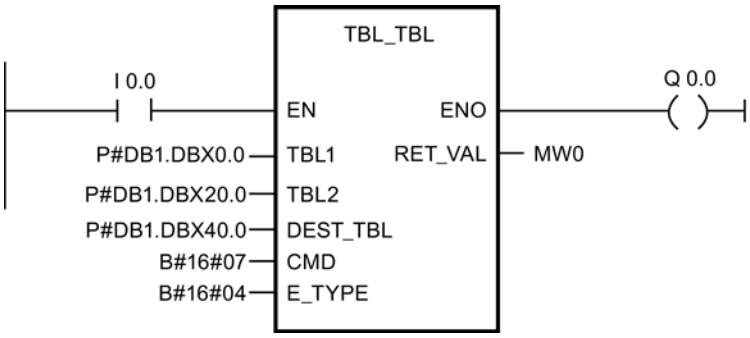
Si se cumple una de las siguientes condiciones, la instrucción "TBL_TBL" no se ejecuta. El estado lógico de BR/ENO se ajusta a "0" y el valor de retorno se ajusta como corresponde:

RET_VAL	Explicación
W#16#0001	Se ha indicado un área de memoria no válida para un parámetro de la instrucción "TBL_TBL".
W#16#0002	El parámetro E_TYPE no es válido.
W#16#0003	Las longitudes de las tablas de entrada y salida no coinciden.
W#16#0004	La tabla tiene una longitud igual a cero.
W#16#0005	Los parámetros E_TYPE y CMD no coinciden.
W#16#0006	El parámetro CMD no es válido

Ejemplo (KOP)

Con estado lógico en la entrada I 0.0 = 1 (activado), la instrucción "TBL_TBL" se ejecuta. En el ejemplo, todas las tablas contienen tres entradas cada una. Así lo indica la primera palabra de cada una de las tablas. El tipo de datos de los valores de la tabla es WORD. Así lo indica E_TYPE. CMD emite el comando con el que deben enlazarse TBL1 y TBL2 .

Si la instrucción "TBL_TBL" se ejecuta sin errores, los estados lógicos de ENOy Q 0.0 se ajustan a "1", y RET_VAL se ajusta al valor W#16#0000.



Antes de la ejecución:

TBL1 (longitud de tabla)	DBW0 = W#16#0003 DBW2 = W#16#00FF DBW4 = W#16#FF00 DBW6 = W#16#FFFF
TBL2 (longitud de tabla)	DBW20 = W#16#0003 DBW22 = W#16#1111 DBW24 = W#16#2222 DBW26 = W#16#3333
DEST_TBL (longitud de tabla)	DBW40 = W#16#0003 DBW42 = W#16#0000 DBW44 = W#16#0000 DBW46 = W#16#0000

Después de la ejecución:

DST_TBL (longitud de tabla)	DBW40 = W#16#0003 DBW42 = W#16#0011 DBW44 = W#16#2200 DBW46 = W#16#3333

PACK: Recoger/distribuir datos de tabla

Descripción

Esta instrucción transfiere datos que se encuentran entre un conjunto cualquiera de direcciones y una tabla. El sentido de la transferencia se define en el parámetro DIR . Cada operación de la instrucción "PACK" procesa como máximo cinco bloques con datos de (P_DATA1 a P_DATA5). Si el parámetro DIR contiene el valor "hacia", la instrucción "PACK" transfiere datos de las direcciones a la tabla indicada. Si el parámetro DIRcontiene el valor "desde", los datos se distribuyen desde la tabla a las direcciones indicadas.

Normas para la transferencia de datos "hacia" una tabla:

- Los bits individuales (BOOL) se transfieren al próximo bit disponible de la tabla.
- Los tipos de datos de 8 bits se transfieren al próximo byte disponible de la tabla. Cuando se escribe un byte en la tabla, los bits no utilizados del byte anterior se rellenan con ceros.
- Los tipos de datos de 16 y 32 bits se transfieren a la próxima palabra disponible de la tabla. Cuando se escribe una palabra en la tabla, los bits no utilizados de la palabra anterior se rellenan con ceros.

Normas para la transferencia de datos "desde" una tabla:

- No debe omitirse sección alguna de la tabla.
- Todos los datos indicados del tipo de datos BOOL se transfieren desde la tabla.
- Los tipos de datos de 8 bits se transfieren desde el primer byte disponible de la tabla. Por eso, el byte que se transfiere desde la tabla no contiene bits no utilizados del byte anterior de la tabla.
- Los tipos de datos de 16 y 32 bits se transfieren desde la primera palabra disponible de la tabla. En consecuencia, la palabra que se transfiere desde la tabla no contiene bits no utilizados de la palabra anterior de la tabla.

Tipos de datos válidos para el puntero ANY y soportados por "PACK": BOOL, WORD, INT, BYTE, DINT, REAL, CHAR, DWORD.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PACK":

Parámetros	Declaración	Tipo de datos	Área de memoria	Descripción
TABLE	Input	*Puntero	I, Q, M, D	Apunta al inicio de la tabla.
P_DATA1	Input	ANY	I, Q, M, D	Apunta al inicio del bloque cuyos datos se desean transferir.
P_DATA2	Input	ANY	I, Q, M, D	Apunta al inicio del bloque cuyos datos se desean transferir.
P_DATA3	Input	ANY	I, Q, M, D	Apunta al inicio del bloque cuyos datos se desean transferir.
P_DATA4	Input	ANY	I, Q, M, D	Apunta al inicio del bloque cuyos datos se desean transferir.
P_DATA5	Input	ANY	I, Q, M, D	Apunta al inicio del bloque cuyos datos se desean transferir.
ERR_CODE	Output	WORD	I, Q, M, D, L, P	Emite el valor W#16#0000si la operación se ha llevado a cabo sin errores.
DIR	Static	BOOL	I, Q, M, D, L	Sentido de la transferencia. Son posibles los siguientes estados lógicos: 0=hacia, 1=desde.
* Puntero en formato de palabra doble para el direccionamiento indirecto por registro abarcando varias áreas				

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Si se cumple una de las siguientes condiciones de error, la instrucción "PACK" no se ejecuta. El estado lógico de BR/ENO se ajusta a "0" y el ERR_CODE se ajusta como corresponde:

ERR_CODE	Explicación
W#16#0001	Se ha indicado un área de memoria no válida para un parámetro de la instrucción "PACK".
W#16#0002	La instrucción "E_TYPE" no es válida.

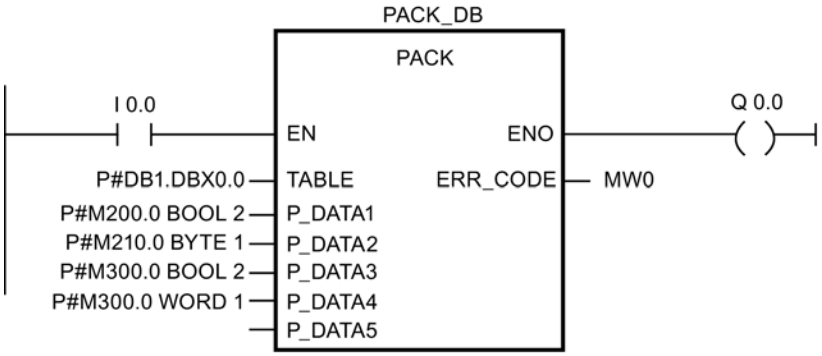
Ejemplo (KOP)

Con estado lógico de la entrada I 0.0 = 1 (activado), se ejecuta "PACK". En el ejemplo se transfieren cuatro bloques con datos "hacia" la tabla.

Si la instrucción "PACK" se ejecuta sin errores, los estados lógicos de ENOy O 0.0 se ajustan a "1", y ERR_CODE se ajusta a "W#16#0000".

Nota

Los parámetros estáticos pueden inicializarse con ayuda del editor de bloques de datos.



Antes de la ejecución:

TABLE	DBB0 = B#16#00 DBB1 = B#16#00 DBB2 = B#16#00 DBB3 = B#16#00 DBB4 = B#16#00 DBB5 = B#16#00
P_DATA1	M 200.0 = TRUE M 200.1 = TRUE
P_DATA2	MB210 = B#16#FF
P_DATA3	M 300.0 = TRUE M 300.1 = TRUE

P_DATA4	MW330 = W#16#FFFF
DB de instancia PACK_DB	
DIR	DBX58.0 = FALSE

Después de la ejecución:

TABLE	DBB0 = B#16#03 DBB1 = B#16#FF DBB2 = B#16#03 DBB3 = B#16#00 DBB4 = B#16#FF DBB5 = B#16#FF

Direccionamiento

GEO_LOG: Determinar la dirección inicial de un módulo

Descripción

Supongamos que se conoce el slot de módulo correspondiente al canal de un módulo de señal. La instrucción "GEO_LOG" permite determinar a partir de él la correspondiente dirección inicial del módulo, es decir, la dirección I o Q más baja.

Si se aplica la instrucción "GEO_LOG" a módulos de potencia o módulos con direcciones comprimidas (ET 200S), se devuelve la dirección de diagnóstico.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GEO_LOG":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MASTER	Input	INT	I, Q, M, D, L o constante	Identificación de área: • 0 si el slot se encuentra en una configuración centralizada: Racks de 0 a 3 (con S7-300) o de 0 a 21 (con S7-400) • de 1 a 32: ID de sistema maestro DP del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFIBUS • de 100 a 115: ID de sistema PROFINET IO del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFINET
STATION	Input	INT	I, Q, M, D, L o constante	• Número del rack, si la identificación de área = 0 • Número de estación del aparato de campo, si la identificación de área > 0
SLOT	Input	INT	I, Q, M, D, L o constante	Número de slot
SUBSLOT	Input	INT	I, Q, M, D, L o constante	Slot de submódulo (si no puede insertarse ningún módulo, debe indicarse aquí 0)
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
LADDR	Output	WORD	I, Q, M, D, L	Dirección inicial del módulo En el bit 15 de LADDR se especifica si se trata de una dirección de entrada (bit 15 = 0) o de salida (bit 15 = 1).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8094	No está configurada ninguna subred con la SUBNETID indicada.
8095	Valor no válido en el parámetro STATION
8096	Valor no válido en el parámetro SLOT
8097	Valor no válido en el parámetro SUBSLOT
8099	El slot no está configurado.
809A	La dirección de submódulo para el slot seleccionado no está configurada.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

LOG_GEO: Determinar el slot que corresponde a una dirección lógica

Descripción

Esta instrucción permite determinar el slot correspondiente a una dirección lógica, así como el offset del área de direcciones de datos útiles del módulo.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LOG_GEO":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
LADDR	Input	WORD	I, Q, M, D, L o constante	Cualquier dirección lógica del módulo En el bit 15 se especifica si se trata de una dirección de entrada (bit 15 = 0) o de salida (bit 15 = 1).
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
AREA	Output	INT	I, Q, M, D, L	Identificación de área: Indica cómo deben interpretarse los parámetros de salida restantes.
MASTER	Output	INT	I, Q, M, D, L	Identificación de área: • 0 si el slot se encuentra en una configuración centralizada: Racks de 0 a 3 (con S7-300) o de 0 a 21 (con S7-400) • de 1 a 32: ID de sistema maestro DP del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFIBUS • de 100 a 115: ID de sistema PROFINET IO del aparato de campo correspondiente, si el slot se encuentra en un aparato de campo conectado a PROFINET
STATION	Output	INT	I, Q, M, D, L	N.º del rack, si: • Identificación de área = 0 • Número de estación del aparato de campo, si la identificación de área > 0
SLOT	Output	INT	I, Q, M, D, L	Número de slot
SUBSLOT	Output	INT	I, Q, M, D, L	Número de submódulo
OFFSET	Output	INT	I, Q, M, D, L	Offset en el área de direcciones de datos útiles del módulo correspondiente

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro AREA

Valor de AREA	Sistema	Significado de MASTER, STATION, SLOT, SUBSLOT y OFFSET
0	S7-400	MASTER: 0 STATION: número de rack SLOT: número de slot SUBSLOT: 0 OFFSET: diferencia entre la dirección lógica y la dirección base lógica
1	S7-300	MASTER: 0 STATION: número de rack SLOT: número de slot SUBSLOT: 0 OFFSET: diferencia entre la dirección lógica y la dirección base lógica
2	PROFIBUS DP	MASTER: ID del sistema maestro DP STATION: número de estación SLOT: número de slot en la estación SUBSLOT: 0 OFFSET: offset en el área de direcciones de datos útiles del módulo correspondiente
	PROFINET IO	MASTER: ID de sistema de PROFINET IO STATION: número de estación SLOT: número de slot en la estación SUBSLOT: número de submódulo OFFSET: Offset en el área de direcciones de datos útiles del módulo correspondiente
3	Área S5-P	MASTER: 0 STATION: número de rack SLOT: número de slot de la cápsula de adaptación SUBSLOT: 0 OFFSET: dirección en el área S5-x
4	Área S5-Q	MASTER: 0 STATION: número de rack SLOT: número de slot de la cápsula de adaptación SUBSLOT: 0 OFFSET: dirección en el área S5-x
5	Área S5-IM3	MASTER: 0 STATION: número de rack SLOT: número de slot de la cápsula de adaptación OFFSET: Dirección en el área S5-x
6	Área S5-IM4	MASTER: 0 STATION: número de rack SLOT: número de slot de la cápsula de adaptación SUBSLOT: 0 OFFSET: dirección en el área S5-x

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	La dirección lógica indicada no es válida
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

RD_LGADR: Determinar todas las direcciones lógicas de un módulo

Descripción

Se toma como punto de partida una dirección lógica de un módulo, de un submódulo central o de un submódulo en PNIO. La instrucción permite determinar todas las direcciones lógicas de ese módulo o submódulo. Es necesario que esté configurada previamente la asignación de direcciones lógicas a módulos o submódulos. La instrucción "RD_LGADR" introduce las direcciones lógicas determinadas en los parámetros PEADDR o PAADDR en orden ascendente.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RD_LGADR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#00: El bit 15 de LADDR indica si se trata de una dirección de entrada (bit 15=0) o de salida (bit 15=1). • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ)
LADDR	Input	WORD	I, Q, M, D, L o constante	Una dirección lógica
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
PEADDR	Output	ANY	I, Q, M, D, L	Campo para las direcciones PI; los elementos del campo deben ser del tipo de datos WORD.
PECOUNT	Output	INT	I, Q, M, D, L	Número de direcciones PI devueltas
PAADDR	Output	ANY	I, Q, M, D, L	Campo para las direcciones PQ; los elementos del campo deben ser del tipo de datos WORD.
PACOUNT	Output	INT	I, Q, M, D, L	Número de direcciones PQ devueltas

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	La dirección lógica indicada no es válida o se ha usado un valor no admisible en el parámetro IOID
80A0	Error en el parámetro de salida PEADDR: El tipo de datos de los elementos del campo no es WORD. (Este código de error solo existe con S7-400 y en la CPU 318.)
80A1	Error en el parámetro de salida PAADDR: El tipo de datos de los elementos del campo no es WORD. (Este código de error solo existe con S7-400 y en la CPU 318.)
80A2	Error en el parámetro de salida PEADDR: El campo indicado no ha podido alojar todas las direcciones lógicas.
80A3	Error en el parámetro de salida PAADDR: El campo indicado no ha podido alojar todas las direcciones lógicas.
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

GADR_LGC: Determinar la dirección base lógica de un módulo

Descripción

Supongamos que se conocen el slot y el offset del área de direcciones de datos útiles correspondientes al canal de un módulo de señal. La instrucción "GADR_LGC" permite determinar a partir de ellos la correspondiente dirección base lógica base del módulo, es decir, la dirección I o Q más baja.

Si se aplica la instrucción "GADR_LGC" a módulos de potencia o módulos con direcciones comprimidas (ET 200S), se devuelve la dirección de diagnóstico.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GADR_LGC":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SUBNETID	Input	BYTE	I, Q, M, D, L o constante	Identificación de área: • 0 si el slot se encuentra en uno de los racks 0 (aparato central) o de 1 a 21 (aparatos de ampliación) • ID de sistema maestro DP del sistema de periferia descentralizada correspondiente, si el slot se encuentra en un sistema de periferia descentralizada
RACK	Input	WORD	I, Q, M, D, L o constante	• Número del rack, si la identificación de área es 0 • Número de estación del sistema de periferia descentralizada, si la identificación de área > 0

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SLOT	Input	WORD	I, Q, M, D, L o constante	Número de slot
SUBSLOT	Input	BYTE	I, Q, M, D, L o constante	Slot de submódulo (si no puede insertarse ningún módulo, debe indicarse aquí 0)
SUBADDR	Input	WORD	I, Q, M, D, L o constante	Offset en el área de direcciones de datos útiles del módulo
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
IOID	Output	BYTE	I, Q, M, D, L	Identificador del área de direcciones: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) En los módulos mixtos, la instrucción devuelve la identificación de área de la dirección más baja. Si las direcciones son iguales, la instrucción devuelve la identificación B#16#54.
LADDR	Output	WORD	I, Q, M, D, L	Dirección base lógica del módulo

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8093	Valor no admisible en el parámetro SUBNETID (la instrucción "GADR_LGC" no es admisible para PROFINET IO).
8094	No está configurada ninguna subred con la SUBNETID indicada.
8095	Valor no válido en el parámetro RACK
8096	Valor no válido en el parámetro SLOT
8097	Valor no válido en el parámetro SUBSLOT
8098	Valor no válido en el parámetro SUBADDR
8099	El slot no está configurado.
809A	La subdirección correspondiente al slot seleccionado no está configurada (posible solo con periferia central para CPU e IM)
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

LGC_GADR: Determinar el slot que corresponde a una dirección lógica

Descripción

Esta instrucción permite determinar el slot correspondiente a una dirección lógica, así como el offset del área de direcciones de datos útiles del módulo.

Nota

No es posible aplicar la instrucción "LGC_GADR" a un módulo con direcciones comprimidas (ET 200S).

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "LGC_GADR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
IOID	Input	BYTE	I, Q, M, D, L o constante	Identificador del área de direcciones: • B#16#00: El bit15 de LADDR indica si se trata de una dirección de entrada (bit15=0) o de salida (bit15=1). • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si se trata de las mismas direcciones, debe indicarse B#16#54 .
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección lógica Con un módulo mixto debe indicarse la dirección más baja de las dos.
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
AREA	Output	BYTE	I, Q, M, D, L	Identificación de área: Indica cómo deben interpretarse los parámetros de salida restantes.
RACK	Output	WORD	I, Q, M, D, L	Número de rack
SLOT	Output	WORD	I, Q, M, D, L	Número de slot
SUBADDR	Output	WORD	I, Q, M, D, L	Offset en el área de direcciones de datos útiles del módulo correspondiente.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se utiliza un sistema de automatización S7-400H en modo redundante y al llamar la instrucción "LGC_GADR" se indica en el parámetro LADDR la dirección lógica de un módulo de un esclavo DP conectado, se devuelve en el byte más significativo del parámetro RACK la ID de sistema maestro DP del canal activo. Si no hay ningún canal activo, se devuelve la ID del correspondiente sistema maestro DP de la CPU maestra.

Parámetro AREA

El parámetro AREA indica cómo deben interpretarse los parámetros de salida RACK, SLOT y SUBADDR . La tabla siguiente explica esa dependencia.

Valor de AREA	Sistema	Significado de RACK, SLOT y SUBADDR
0	S7-400	RACK : Número de rack SLOT : Número de slot SUBADDR : Diferencia entre la dirección lógica y la dirección base lógica
1	S7-300	RACK : Número de rack SLOT : Número de slot SUBADDR : Diferencia entre la dirección lógica y la dirección base lógica
2	DP	RACK : (byte menos significativo): Número de estación RACK : (byte más significativo): ID del sistema maestro DP SLOT : Número de slot en la estación SUBADDR : Offset en el área de direcciones de datos útiles del módulo correspondiente
3	Área S5-P	RACK : Número de rack SLOT : Número de slot de la cápsula de adaptación SUBADDR : Dirección en el área S5-x
4	Área S5-Q	RACK : Número de rack SLOT : Número de slot de la cápsula de adaptación SUBADDR : Dirección en el área S5-x
5	Área S5-IM3	RACK : Número de rack SLOT : Número de slot de la cápsula de adaptación SUBADDR : Dirección en el área S5-x
6	Área S5-IM4	RACK : Número de rack SLOT : Número de slot de la cápsula de adaptación SUBADDR : Dirección en el área S5-x

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8090	La dirección lógica indicada no es válida o se ha usado un valor no admisible en el parámetro IOID

Código de error (W#16#...)	Explicación
8093	Esta instrucción no es válida para el módulo seleccionado mediante los parámetros IOID y LADDR .
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Otras funciones

iSlave

SET_ADDR: Establecer dirección de red como iSlave propio

Descripción

Esta instrucción permite modificar la dirección MAC de una interfaz DP de iSlave. El direccionamiento se realiza mediante la dirección de diagnóstico de la interfaz DP. Desde el punto de vista del maestro DP, el esclavo DP con la dirección antigua se desconecta del bus y se conecta otro esclavo DP con la dirección nueva.

- Si la dirección ya está ocupada por una estación en la red DP, los dos esclavos DP se desconectan de la red.
- Si la dirección se utiliza también como dirección MPI para la conexión a la PG, el acceso al dispositivo puede quedar bloqueado. Para saber si los dispositivos están accesibles, puede consultarse el cuadro de diálogo "Dispositivos accesibles".

La dirección de red no se escribe en el SDB (bloque de datos del sistema) ni se carga en la memoria de trabajo. Cuando la CPU pasa a STOP, o en caso de arranque en caliente, la dirección de red se mantiene. Después de un reset o un arranque en frío, se utiliza la dirección configurada originalmente.

Esta instrucción no puede usarse para PROFINET IO, ya que las direcciones de Ethernet pueden definirse como disjuntas en todo el mundo. Si se utiliza con PROFINET IO, se devuelve el código de error 16#809B.

Funcionamiento

La instrucción "SET_ADDR" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición, debe llamarse "READ_DBL" con REQ =1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SET_ADDR":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ=1: Llamada de la función
LADDR	Input	WORD	I, Q, M, D, L o constante	Dirección base lógica
ADDR	Input	BYTE	I, Q, M, D, L o constante	Nueva dirección del dispositivo
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La petición todavía no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
700x	La petición todavía no ha finalizado. Parámetro BUSY=1
8091	La nueva dirección de estación (ADDR) no es válida
8092	La estación no es una dirección de esclavo propia del esclavo I
8093	LADDR no es válido o no es una interfaz
809B	No se puede ejecutar la instrucción (p. ej. la interfaz no es un esclavo DP o no está activa).
80C3	Recursos insuficientes (p. ej. varias llamadas de instrucciones con diferentes parámetros).
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

9.8.2.4 Tecnología

Control PID

Funciones básicas de PID

CONT_C

Descripción de CONT_C

La instrucción CONT_C permite regular procesos técnicos con magnitudes continuas de entrada y salida en los sistemas de automatización SIMATIC S7. Mediante parametrización se pueden activar o desactivar funciones parciales del regulador PID para adaptarlo al sistema regulado. Además de las funciones de las ramas de valor real y de consigna, la instrucción

realiza un regulador PID terminado con salida continua de magnitud manipulada y posibilidad de controlar manualmente la variable manipulada.

Aplicación

El regulador se puede utilizar como regulador PID de valor fijo de manera individual o también en regulaciones de varios lazos como regulador en cascada, mixto o de relación. El modo de funcionamiento se basa en el algoritmo de regulación PID del regulador por muestreo con señal de salida analógica, complementada dado el caso con una etapa de formación de impulsos para generar señales de salida con modulación de ancho de impulsos para regulaciones de dos o tres puntos con elementos finales de control proporcionales.

Llamada

La instrucción CONT_C dispone de una rutina de inicialización, que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado. El integrador se ajusta al valor de inicialización I_ITVAL durante la inicialización. Todas las salidas de señal restantes se ponen a cero. Una vez ejecutada la rutina de inicialización, debe activarse COM_RST = FALSE.

El cálculo de los valores en los bloques de regulación sólo se realiza correctamente si el bloque se llama en intervalos regulares. Por ello, los bloques de regulación deben llamarse en un OB de alarma cíclica (OB 30 a OB 38). El tiempo de muestreo se indica en el parámetro CYCLE.

Si se efectúa una llamada a la instrucción CONT_C como DB multiinstancia, no se crea ningún objeto tecnológico. No se dispone de interfaz de parametrización y puesta en servicio. Los CONT_C se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Información de error

El bloque no evalúa la palabra de notificación de error RET_VAL.

Consulte también

Diferencias respecto a CONT_C S7-300/400 (Página 5242)

Funcionamiento de CONT_C

Rama de consigna

La consigna se indica en la entrada SP_INT en formato de coma flotante.

Rama de valor real

El valor real se puede leer en formato de coma flotante o en formato de periferia. La función CRP_IN transforma el valor de periferia PV_PER en un formato en coma flotante de -100 ... +100 %, según la siguiente fórmula:

Salida de CRP_IN = PV_PER * 100 / 27648

La función PV_NORM normaliza la salida de CRP_IN según la siguiente norma:

Salida de PV_NORM = (salida de CRP_IN) *PV_FAC + PV_OFF

PV_FAC tiene preasignado el valor 1 y PV_OFF el valor 0.

Cálculo del error de regulación

La diferencia entre la consigna y el valor real es el error de regulación. Para suprimir una pequeña oscilación permanente provocada por la cuantificación de las magnitudes manipuladas (p. ej. en caso de una modulación de ancho de impulsos con PULSEGEN), el error de regulación se deriva a través de una zona muerta (DEADBAND). Con DEADB_W = 0 la zona muerta está desactivada.

Algoritmo PID

El algoritmo PID trabaja en el algoritmo de posición. Las acciones proporcional, integral (INT) y derivativa (DIF) están conectadas en paralelo y pueden conectarse y desconectarse individualmente. De esta forma pueden parametrizarse reguladores P, PI, PD y PID. Pero también son posibles reguladores I.

Procesamiento de valores en manual

Es posible conmutar entre los modos manual y automático. En el modo manual, la magnitud manipulada se corrige según un valor manual.

El integrador (INT) se pone internamente a LMN - LMN_P - DISV y el diferenciador (DIF) se pone a 0 y se compensa internamente. La conmutación al modo automático se efectúa así con suavidad.

Procesamiento de valores manipulados

La variable manipulada se limita con la función LMNLIMIT a valores especificables. El rebase de los límites por la magnitud de entrada se señaliza mediante bits de aviso.

La función LMN_NORM normaliza la salida de LMNLIMIT según la siguiente regla:

LMN = (salida de LMNLIMIT) * LMN_FAC + LMN_OFF

LMN_FAC tiene preasignado el valor 1 y LMN_OFF el valor 0.

La variable manipulada también está disponible en formato de periferia. La función CRP_OUT convierte el valor en coma flotante LMN en un valor de periferia según la siguiente regla:

LMN_PER = LMN * 27648 / 100

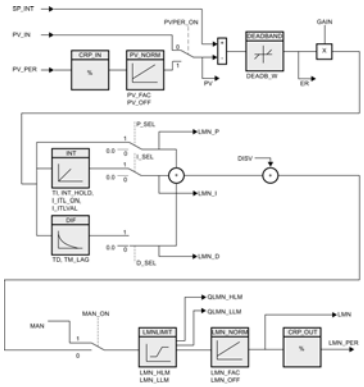
Control anticipativo

En la entrada DISV se puede aplicar una magnitud perturbadora de manera aditiva.

Consulte también

Diferencias respecto a CONT_C S7-300/400 (Página 5242)

Diagrama de bloques de CONT_C



Consulte también

Diferencias respecto a CONT_C S7-300/400 (Página 5242)

Parámetros de entrada de CONT_C

Tabla 9-184

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	BOOL	FALSE	La instrucción tiene una rutina de inicialización, que se ejecuta cuando está activada la entrada "Rearranque completo".
MAN_ON	BOOL	TRUE	Si está activada la entrada "Conectar modo manual", el lazo de regulación está interrumpido. Como variable manipulada se especifica un valor manual.
PVPER_ON	BOOL	FALSE	Si el valor real se debe leer de la periferia, la entrada PV_PER debe estar interconectada con la periferia, y la entrada "Activar valor real de periferia" debe estar activada.
P_SEL	BOOL	TRUE	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción P está conectada si está activada la entrada "Activar acción P".
I_SEL	BOOL	TRUE	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción I está conectada si está activada la entrada "Activar acción I".
INT_HOLD	BOOL	FALSE	La salida del integrador se puede congelar. Para ello debe estar activada la entrada "Congelar acción I".
I_ITL_ON	BOOL	FALSE	La salida del integrador se puede aplicar a la entrada I_ITLVAL. Para ello, la entrada "Inicializar acción I" debe estar activada.
D_SEL	BOOL	FALSE	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción D está conectada si está activada la entrada "Activar acción D".

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
CYCLE	TIME	T#1s	El tiempo entre las llamadas del bloque debe ser constante. La entrada "Tiempo de muestreo" indica el tiempo entre las llamadas del bloque. CYCLE >= 1ms
SP_INT	REAL	0.0	La entrada "Consigna interna" sirve para preseleccionar una consigna. Se admiten los valores de -100 a 100 % o una magnitud física 1).
PV_IN	REAL	0.0	En la entrada "Valor real de la entrada" se puede parametrizar un valor de puesta en servicio, o se puede conectar un valor real externo en coma flotante. Se admiten los valores de -100 a 100 % o una magnitud física 1).
PV_PER	WORD	W#16#0000	El valor real en formato de periferia se conecta al regulador a través de la entrada "Valor real de la periferia".
MAN	REAL	0.0	La entrada "Valor manual" sirve para establecer un valor manual mediante una función de manejo/visualización. Se admiten los valores de -100 a 100 % o una magnitud física 2).
GAIN	REAL	2.0	La entrada "Ganancia proporcional" indica la ganancia del regulador.
TI	TIME	T#20s	La entrada "Tiempo de integración" determina el comportamiento temporal del integrador. TI >= CYCLE
TD	TIME	T#10s	La entrada "Tiempo de acción derivada" determina el comportamiento temporal del diferenciador. TD >= CYCLE
TM_LAG	TIME	T#2s	Tiempo de retardo de la acción D El algoritmo de la acción D contiene un retardo que se puede parametrizar en la entrada "Tiempo de retardo de la acción D". TM_LAG >= CYCLE/2
DEADB_W	REAL	0.0	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. DEADB_W >= 0.0 (%) o una magnitud física 1)
LMN_HLM	REAL	100.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior de la variable manipulada" indica el límite superior. Se admiten valores reales desde LMN_LLM o una magnitud física 2).
LMN_LLM	REAL	0.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior de la variable manipulada" indica el límite inferior. Se admiten valores reales hasta LMN_HLM o una magnitud física 2).
PV_FAC	REAL	1.0	La entrada "Factor de valor real" se multiplica por el valor real. La entrada sirve para adaptar el rango del valor real.
PV_OFF	REAL	0.0	La entrada "Offset de valor real" se suma al valor real. La entrada sirve para adaptar el rango del valor real.
LMN_FAC	REAL	1.0	La entrada "Factor de la variable manipulada" se multiplica por la variable manipulado. La entrada sirve para adaptar el rango de la variable manipulada.
LMN_OFF	REAL	0.0	La entrada "Offset de la variable manipulada" se suma a la variable manipulada. La entrada sirve para adaptar el rango de la variable manipulada.

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
I_ITLVAL	REAL	0.0	La salida del integrador se puede aplicar en la entrada I_ITL_ON. En la entrada "Valor de inicialización para la acción I" está el valor de inicialización. Se admiten valores desde -100 hasta 100.0 (%) o una magnitud física 2).
DISV	REAL	0.0	Para realizar un control anticipativo, la magnitud perturbadora se conecta a la entrada "Variable perturbadora". Se admiten valores desde -100 hasta 100.0 (%) o una magnitud física 2).

1) Parámetros en la rama de consigna, rama de valor real, con la misma unidad

2) Parámetros en la rama de variable manipulada con la misma unidad

Consulte también

Diferencias respecto a CONT_C S7-300/400 (Página 5242)

Parámetros de salida de CONT_C

Tabla 9-185

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
LMN	REAL	0.0	En la salida "Variable manipulada" se indica la variable manipulada realmente efectiva en formato de coma flotante.
LMN_PER	WORD	W#16#0000	La variable manipulada en formato de periferia se conecta al regulador en la salida "Variable manipulada de la periferia".
QLMN_HLM	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida "Se ha alcanzado el límite superior de la variable manipulada" indica que se ha alcanzado el límite superior.
QLMN_LLM	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida "Se ha alcanzado el límite inferior de la variable manipulada" indica que se ha alcanzado el límite inferior.
LMN_P	REAL	0.0	La salida "Acción P" contiene la acción proporcional de la magnitud manipulada.
LMN_I	REAL	0.0	La salida "Acción I" contiene la acción integral de la magnitud manipulada.
LMN_D	REAL	0.0	La salida "Acción D" contiene la acción derivativa de la magnitud manipulada.
PV	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.

Consulte también

Diferencias respecto a CONT_C S7-300/400 (Página 5242)

CONT_S

Descripción de CONT_S

La instrucción CONT_S permite regular procesos técnicos con señales de salida binarias de la variable manipulada para elementos finales de control integrales en los sistemas de automatización SIMATIC S7. Mediante parametrización es posible activar o desactivar funciones parciales del regulador paso a paso PI para adaptarlo al sistema regulado. Junto a las funciones de la rama de valor real, la instrucción realiza un regulador PI terminado con salida binaria de variable manipulada y posibilidades de influir manualmente sobre la variable manipulada. El regulador paso a paso opera sin realimentación de posición.

Aplicación

El regulador puede utilizarse como regulador PI de valor fijo, ya sea individualmente o en lazos de regulación subordinados en regulaciones en cascada, mixtas o de relación, pero no como regulador piloto. El modo de funcionamiento se basa en el algoritmo de regulación PI del regulador por muestreo y se completa con los elementos funcionales que generan la señal de salida binaria a partir de la señal manipulada analógica.

Llamada

La instrucción CONT_S dispone de una rutina de inicialización, que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado. Todas las salidas de señal se ponen a cero. Una vez ejecutada la rutina de inicialización, debe activarse COM_RST = FALSE.

El cálculo de los valores en los bloques de regulación sólo se realiza correctamente si el bloque se llama en intervalos regulares. Por ello, los bloques de regulación deben llamarse en un OB de alarma cíclica (OB 30 a OB 38). El tiempo de muestreo se indica en el parámetro CYCLE.

Si se efectúa una llamada a la instrucción CONT_S como DB multiinstancia, no se crea ningún objeto tecnológico. No se dispone de interfaz de parametrización y puesta en servicio. Los CONT_S se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Información de error

El bloque no evalúa la palabra de notificación de error RET_VAL.

Consulte también

Diferencias respecto a CONT_S S7-300/400 (Página 5242)

Funcionamiento de CONT_S

Rama de consigna

La consigna se indica en la entrada SP_INT en formato de coma flotante.

Rama de valor real

El valor real se puede leer en formato de coma flotante o en formato de periferia. La función CRP_IN transforma el valor de periferia PV_PER en un formato en coma flotante de -100 ... +100 %, según la siguiente fórmula:

Salida de CRP_IN = PV_PER * 100 / 27648

La función PV_NORM normaliza la salida de CRP_IN según la siguiente norma:

Salida de PV_NORM = (salida de CRP_IN) * PV_FAC + PV_OFF

PV_FAC tiene preasignado el valor 1 y PV_OFF el valor 0.

Cálculo de error de regulación

La diferencia entre la consigna y el valor real constituye el error de regulación. Para suprimir una pequeña oscilación permanente provocada por la cuantificación de las magnitudes manipuladas (resolución limitada de la variable manipulada por parte de la servoválvula), el error de regulación se conduce a través de una zona muerta (DEADBAND). Con DEADB_W = 0 la zona muerta está desactivada.

Algoritmo paso a paso PI

La instrucción opera sin realimentación de posición. La acción I del algoritmo PI y la realimentación de posición supuesta se calculan en un integrador (INT) y se comparan como valor de realimentación con la acción P restante. La diferencia pasa a un elemento de tres puntos (THREE_ST) y a un generador de impulsos (PULSEOUT), que genera los impulsos para la servoválvula. Mediante una adaptación del umbral de respuesta del elemento de tres puntos se reduce la frecuencia de conmutación del regulador.

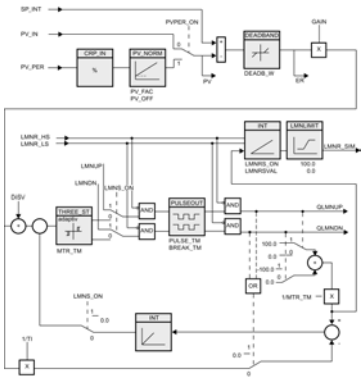
Control anticipativo

En la entrada DISV se puede aplicar una magnitud perturbadora de manera aditiva.

Consulte también

Diferencias respecto a CONT_S S7-300/400 (Página 5242)

Diagrama de bloques de CONT_S



Consulte también

Diferencias respecto a CONT_S S7-300/400 (Página 5242)

Parámetros de entrada de CONT_S

Tabla 9-186

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	BOOL	FALSE	La instrucción tiene una rutina de inicialización, que se ejecuta cuando está activada la entrada "Rearranque completo".
LMNR_HS	BOOL	FALSE	La señal "Servoválvula en el tope superior" se aplica en la entrada "Señal de tope superior de la realimentación de posición". LMNR_HS=TRUE significa: La servoválvula se encuentra en el tope superior.
LMNR_LS	BOOL	FALSE	La señal "Servoválvula en el tope inferior" se aplica en la entrada "Señal de tope inferior de la realimentación de posición". LMNR_LS=TRUE significa: La servoválvula se encuentra en el tope inferior.
LMNS_ON	BOOL	FALSE	En la entrada "Activar el modo manual de las señales de variable manipulada" el procesamiento de las señales manipuladas pasa a manual.
LMNUP	BOOL	FALSE	En el modo manual de las señales de variable manipulada se controla la señal de salida QLMNUP en la entrada "Señal alta de variable manipulada".
LMNDN	BOOL	FALSE	En el modo manual de las señales de variable manipulada se controla la señal de salida QLMNDN en la entrada "Señal baja de variable manipulada".
PVPER_ON	BOOL	FALSE	Si se debe leer el valor real de la periferia, la entrada PV_PER debe estar conectada a la periferia, y la entrada "Activar valor real de la periferia" debe estar activada.
CYCLE	TIME	T#1s	El tiempo entre las llamadas del bloque debe ser constante. La entrada "Tiempo de muestreo" indica el tiempo entre las llamadas del bloque. CYCLE >= 1ms
SP_INT	REAL	0.0	La entrada "Consigna interna" sirve para preseleccionar una consigna. Se admiten los valores de -100 a 100 % o una magnitud física [1].

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
PV_IN	REAL	0.0	En la entrada "Valor real de la entrada" se puede parametrizar un valor de puesta en servicio, o se puede conectar un valor real externo en coma flotante. Se admiten los valores de -100 a 100 % o una magnitud física [1].
PV_PER	WORD	W#16#0000	El valor real en formato de periferia se conecta al regulador a través de la entrada "Valor real de la periferia".
GAIN	REAL	2.0	La entrada "Ganancia proporcional" indica la ganancia del regulador.
TI	TIME	T#20s	La entrada "Tiempo de integración" determina el comportamiento temporal del integrador. TI >= CYCLE
DEADB_W	REAL	1.0	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. Se admiten los valores de 0 a 100 % o una magnitud física [1].
PV_FAC	REAL	1.0	La entrada "Factor de valor real" se multiplica por el valor real. La entrada sirve para adaptar el rango del valor real.
PV_OFF	REAL	0.0	La entrada "Offset de valor real" se suma al valor real. La entrada sirve para adaptar el rango del valor real.
PULSE_TM	TIME	T#3s	En el parámetro "Duración mínima de impulso" se puede parametrizar una duración de impulso mínima. PULSE_TM >= CYCLE
BREAK_TM	TIME	T#3s	En el parámetro "Duración mínima de pausa" se puede parametrizar una longitud de pausa mínima. BREAK_TM >= CYCLE
MTR_TM	TIME	T#30s	En el parámetro "Tiempo de posicionamiento del motor" se introduce el tiempo de operación de la servoválvula de tope a tope. MTR_TM >= CYCLE
DISV	REAL	0.0	Para realizar un control anticipativo, la magnitud perturbadora se conecta a la entrada "Variable perturbadora". Se admiten los valores de -100 a 100 % o una magnitud física [2].

[1] Parámetros en la rama de consigna, rama de valor real, con la misma unidad

[2] Parámetros en la rama de variable manipulada con la misma unidad

Consulte también

Diferencias respecto a CONT_S S7-300/400 (Página 5242)

Parámetros de salida de CONT_S

Tabla 9-187

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
QLMNUP	BOOL	FALSE	Si la salida "Señal alta de variable manipulada" está activada, la servoválvula debe abrirse.
QLMNDN	BOOL	FALSE	Si está activada la salida "Señal baja de variable manipulada", la servoválvula debe cerrarse.
PV	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.

Consulte también

Diferencias respecto a CONT_S S7-300/400 (Página 5242)

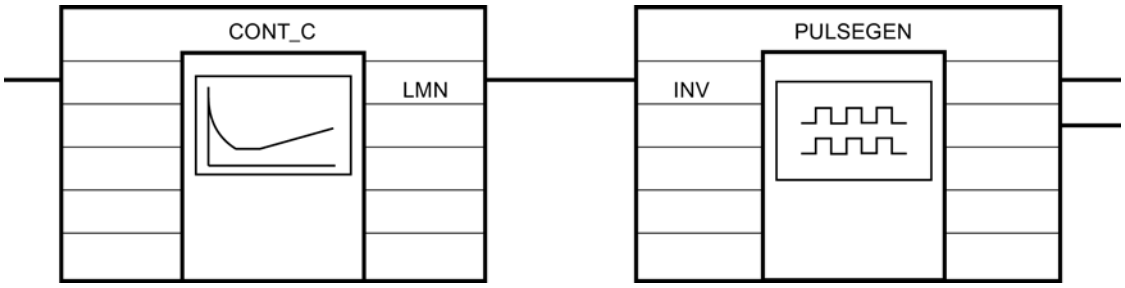
PULSEGEN

Descripción de PULSEGEN

La instrucción PULSEGEN permite establecer un regulador PID con salida de impulsos para actuadores proporcionales. PULSEGEN transforma la magnitud de entrada INV (= LMN del regulador PID) mediante modulación de ancho de impulsos en una secuencia de impulsos con duración de periodo constante, que corresponde al tiempo de ciclo con el que se actualiza la magnitud de entrada.

Aplicación

La instrucción PULSEGEN permite generar reguladores PID de dos o tres puntos con modulación de ancho de impulsos. La función se aplica casi siempre en combinación con el regulador continuo CONT_C.



Llamada

La instrucción PULSEGEN dispone de una rutina de inicialización, que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado. Todas las salidas de señal se ponen a cero. Una vez ejecutada la rutina de inicialización, debe activarse COM_RST = FALSE.

El cálculo de los valores en los bloques de regulación sólo se realiza correctamente si el bloque se llama en intervalos regulares. Por ello, los bloques de regulación deben llamarse en un OB de alarma cíclica (OB 30 a OB 38). El tiempo de muestreo se indica en el parámetro CYCLE.

Comportamiento en caso de error

El bloque no evalúa la palabra de notificación de error RET_VAL.

Consulte también

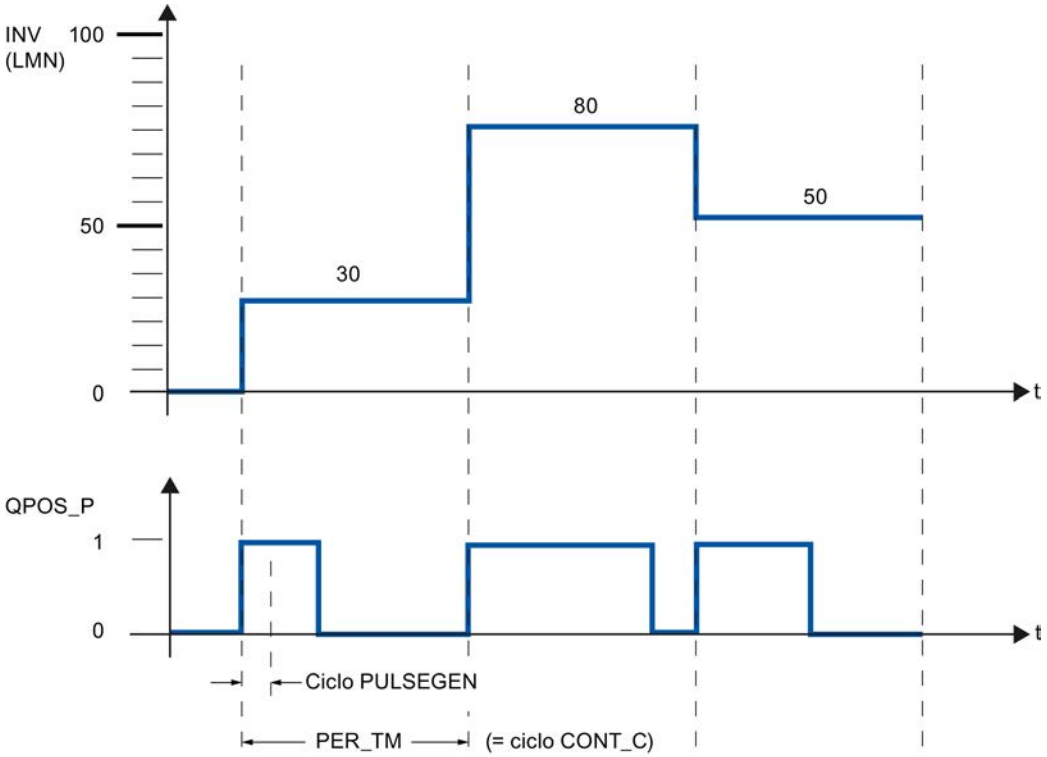
Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

Funcionamiento de PULSEGEN

Modulación de ancho de impulsos

La duración de un impulso por duración de período es proporcional a la magnitud de entrada. El ciclo parametrizado con PER_TM no es idéntico al ciclo de ejecución de la instrucción PULSEGEN. Es más, un ciclo PER_TM está compuesto por varios ciclos de ejecución de la

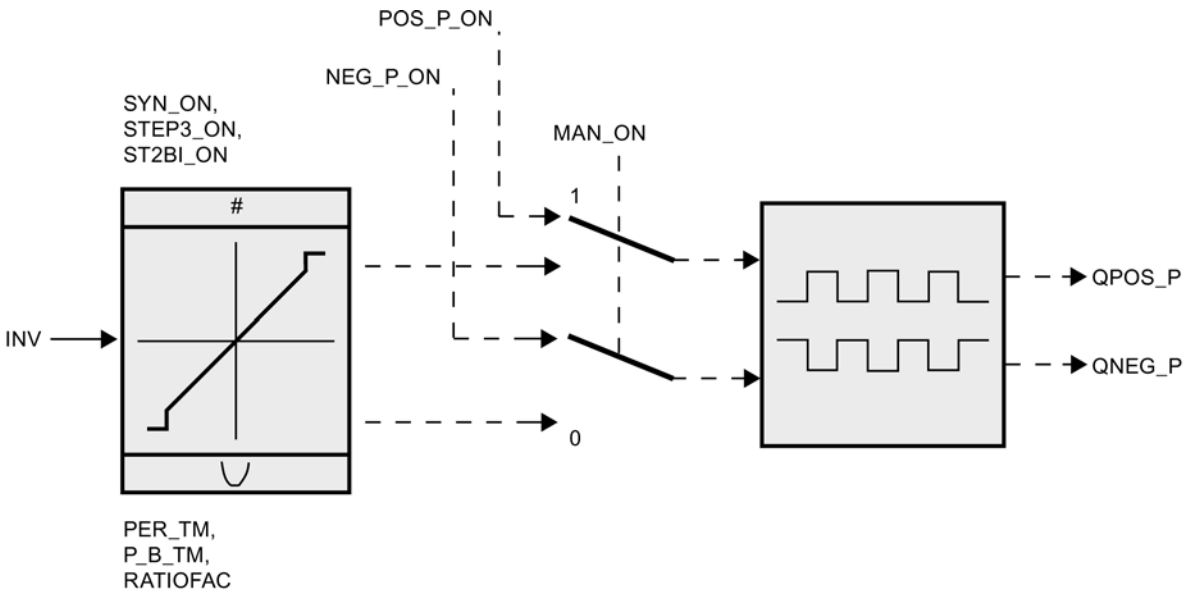
instrucción PULSEGEN, y la cantidad de llamadas de PULSEGEN por ciclo PER_TM representa una medida para la precisión del ancho de impulso.



Así, una magnitud de entrada del 30 % y 10 llamadas de PULSEGEN por PER_TM significan:

- "uno" en la salida QPOS_P para las tres primeras llamadas de PULSEGEN (30% de 10 llamadas)
- "cero" en la salida QPOS_P para otras siete llamadas de PULSEGEN (70% de 10 llamadas)

Diagrama de bloques



Precisión de variable manipulada

En este ejemplo, con una "relación de muestreo" de 1:10 (llamadas de CONT_C respecto a llamadas de PULSEGEN) la precisión de la variable manipulada se reduce al 10%, es decir, los valores de entrada predeterminados INV sólo se pueden representar en intervalos del 10% en una duración de impulso en la salida QPOS_P.

Del mismo modo aumenta la precisión con la cantidad de llamadas de PULSEGEN por cada llamada de CONT_C.

Si por ejemplo PULSEGEN se llama con una frecuencia 100 veces mayor que CONT_C, se consigue una resolución del 1% del rango de la variable manipulada.

Nota

La desmultiplicación de la frecuencia de llamada debe ser programada por el usuario.

Sincronización automática

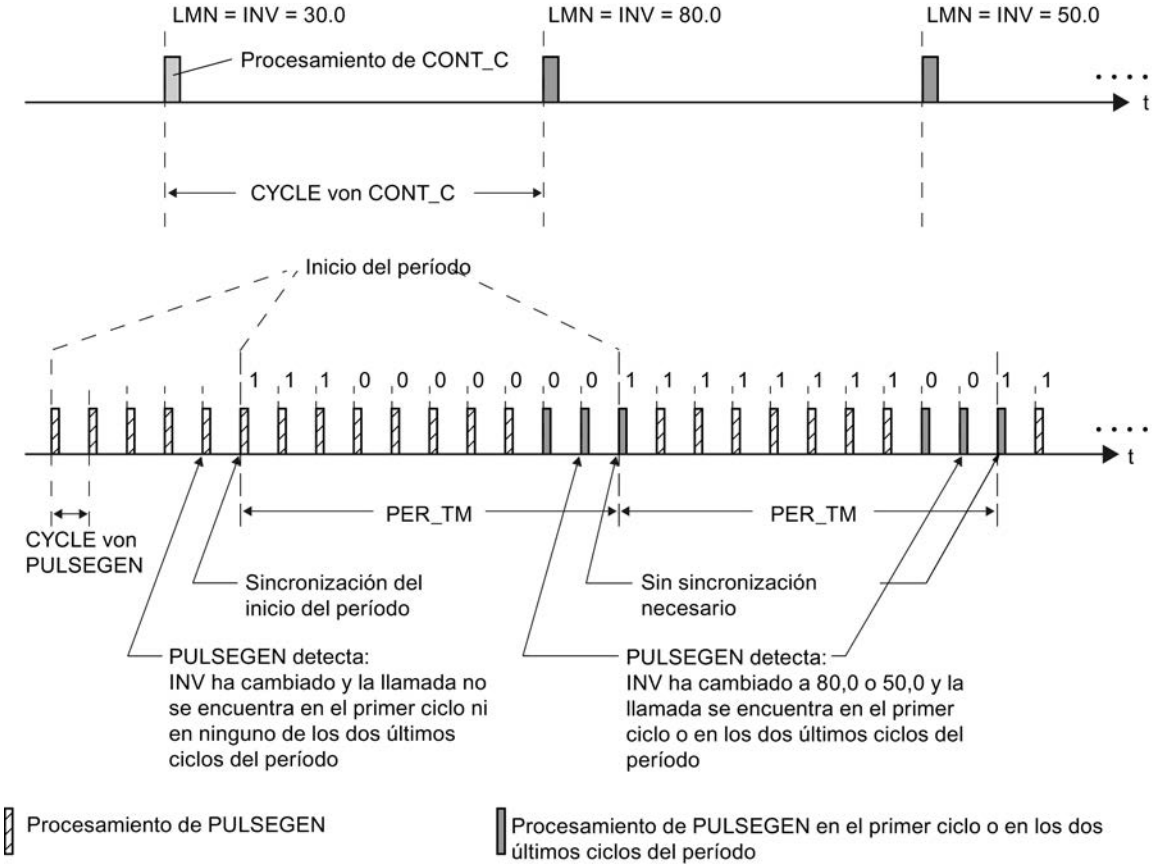
Existe la posibilidad de sincronizar automáticamente la salida de impulsos con la instrucción que actualiza la magnitud de entrada INV (p. ej. CONT_C). De este modo se garantiza que una magnitud de entrada cambiante también se pueda emitir como impulso con la mayor rapidez posible.

El formador de impulsos evalúa siempre la magnitud de entrada INV en el intervalo correspondiente a la duración de período PER_TM y transforma el valor en una señal de impulso con la correspondiente duración.

Pero como INV se calcula casi siempre en un nivel de alarma cíclica más lento, el formador de impulsos debería iniciar la transformación del valor discreto en una señal de impulso con la máxima rapidez posible tras la actualización de INV.

Para ello, el bloque puede sincronizar él mismo el inicio del período según el siguiente procedimiento:

Si INV ha cambiado y la llamada de bloque no se encuentra en el primer ciclo de llamada o en los dos últimos ciclos de un período, se realiza una sincronización. La duración de impulso se calcula de nuevo, y en el siguiente ciclo se inicia la emisión con un nuevo periodo.



La sincronización automática se desactiva con SYN_ON = FALSE.

Nota

Con el inicio del periodo nuevo, el valor antiguo de INV (es decir, de LMN) se representa en la señal de impulso de manera más o menos precisa una vez concluida la sincronización.

Consulte también

Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

Modos de operación de PULSEGEN

Modos de operación

Dependiendo de la parametrización del formador de impulsos, los reguladores PID se pueden configurar con comportamiento de tres puntos o con salida de dos puntos bipolar o unipolar. La siguiente tabla contiene los ajustes de las combinaciones de conmutadores para los modos de operación posibles.

Modo de operación	MAN_ON	STEP3_ON	ST2BI_ON
Regulación de tres puntos	FALSE	TRUE	cualquiera
Regulación de dos puntos con rango de valores manipulados bipolar (-100 % ... 100 %)	FALSE	FALSE	TRUE
Regulación de dos puntos con rango de valores manipulados unipolar (0 % ... 100 %)	FALSE	FALSE	FALSE
Modo manual	TRUE	cualquiera	cualquiera

Modo manual en la regulación de dos o de tres puntos

En el modo manual (MAN_ON = TRUE) las salidas binarias del regulador de tres puntos o del regulador de dos puntos pueden activarse independientemente de INV a través de las señales POS_P_ONy NEG_P_ON.

Regulación	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
Regulación de tres puntos	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
Regulación de dos puntos	FALSE	cualquiera	FALSE	TRUE
	TRUE	cualquiera	TRUE	FALSE

Consulte también

Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

Regulación de tres puntos

Regulación de tres puntos

En el modo de operación "Regulación de tres puntos" se pueden generar tres estados de la señal de salida. Para ello se asignan los valores de estado de las señales binarias de salida

QPOS_P y QNEG_P a los correspondientes estados operativos del actuador. La tabla muestra el ejemplo de una regulación de temperatura:

Señales de salida	Calentar	OFF	Enfriar
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

A partir de la magnitud de entrada se calcula la duración del impulso mediante una característica. La forma de esta característica está definida por la duración mínima de impulso o de pausa y por el factor de relación. El valor normal del factor de relación es 1.

Los puntos escalonados de las características son originados por la duración mínima de impulso o de pausa.

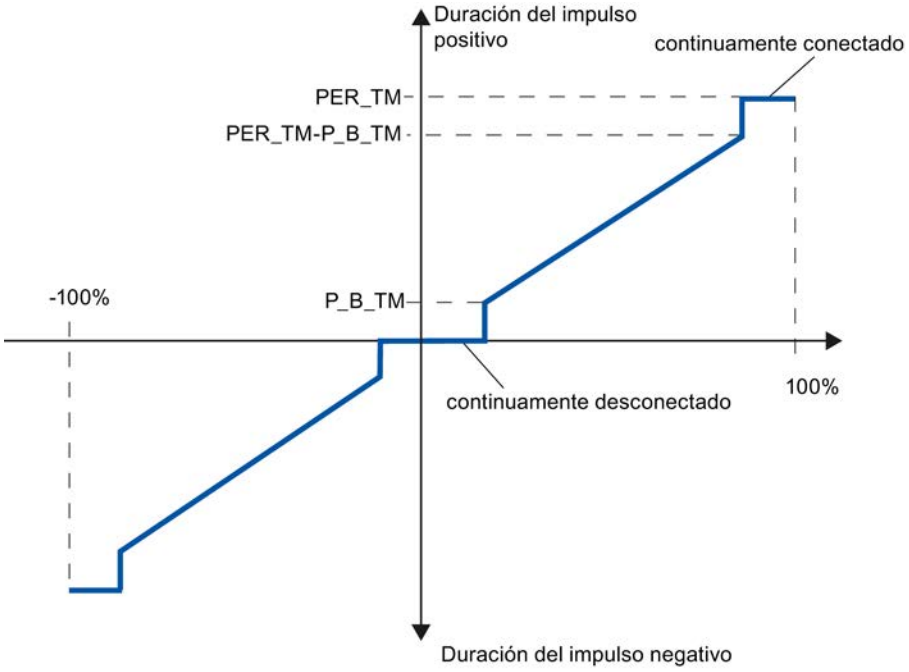
Duración mínima de impulso o de pausa

La correcta parametrización de la duración mínima de impulso o de pausa P_B_TM evita tiempos de conexión o desconexión breves, que merman la vida útil de los elementos de contacto y de los elementos de control final. Se suprimen valores absolutos pequeños de la magnitud de entrada LMN que generarían una duración de impulso inferior a P_B_TM. Los valores de entrada grandes que generarían una duración de impulso superior a PER_TM - P_B_TM se ajustan a 100 % o a -100 %.

La duración de los impulsos positivos o negativos se calcula multiplicando la magnitud de entrada (en %) por la duración del período:

Duración de impulso = INV / 100 * PER_TM

La siguiente figura muestra una característica simétrica del regulador de tres puntos (factor de relación = 1)



Regulación de tres puntos asimétrica

A través del factor de relación RATIOFAC puede modificarse la relación de la duración de impulsos positivos respecto a los negativos. Durante un lazo de regulación de temperatura, ello permite tener en cuenta p. ej. constantes temporales de proceso distintas para calentar y refrigerar.

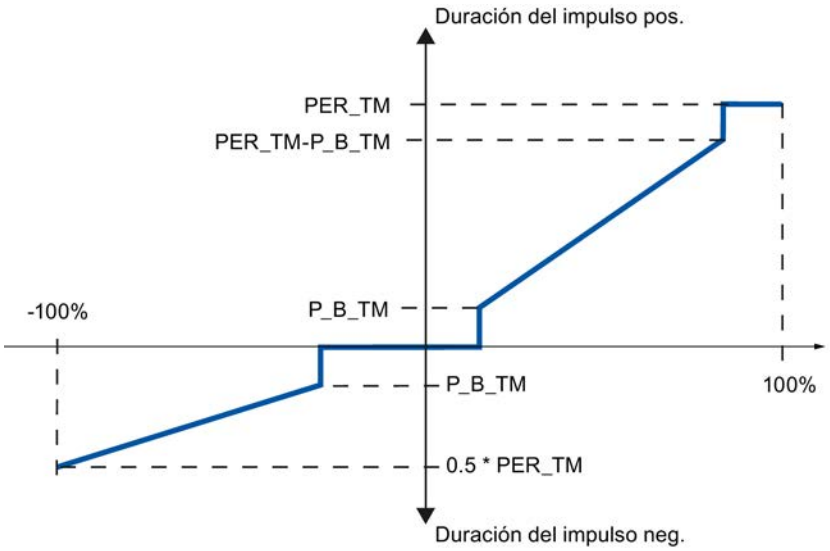
Factor de relación < 1

La duración de impulso obtenida de la multiplicación de la magnitud de entrada por la duración del período en la salida de impulsos negativa se multiplica por el factor de relación.

Duración de impulso positiva = INV /100 * PER_TM

Duración de impulso negativa = INV / 100 * PER_TM * RATIOFAC

La siguiente figura muestra una característica asimétrica del regulador de tres puntos (factor de relación = 0.5)



Factor de relación > 1

La duración de impulso obtenida de la multiplicación de la magnitud de entrada por la duración del período en la salida de impulsos positiva se divide por el factor de relación.

Duración de impulso positiva = INV / 100 * PER_TM / RATIOFAC

Duración de impulso negativa = INV / 100 * PER_TM

Consulte también

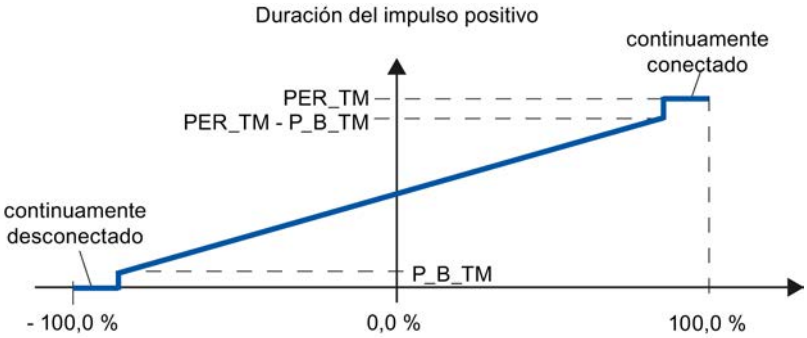
Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

Regulación de dos puntos

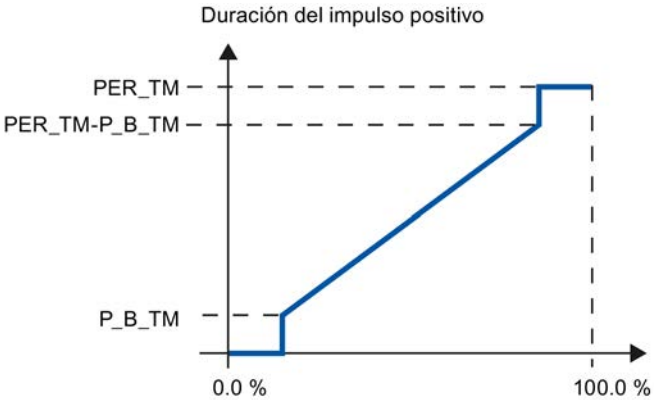
En la regulación de dos puntos sólo se conecta la salida de impulsos positiva QPOS_P de PULSEGEN al actuador de conexión/desconexión en cuestión. Según cuál sea el rango

utilizado de la variable manipulada, el regulador de dos puntos tiene un rango bipolar o unipolar.

Regulación de dos puntos con rango de variable manipulada bipolar (-100% ... 100%)



Regulación de dos puntos con rango de variable manipulada unipolar (0%...100%)



En QNEG_P está disponible la señal de salida negada, si la interconexión del regulador de dos puntos en el lazo de regulación exige una señal binaria invertida lógicamente para los impulsos de la señal manipulada.

Impulso	Actuador ON	Actuador OFF
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

Consulte también

Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

Parámetros de entrada de PULSEGEN

Los valores de los parámetros de entrada no se limitan en el bloque; no tiene lugar una comprobación de los parámetros.

Tabla 9-188

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
INV	REAL	0.0	En el parámetro de entrada "Variable de entrada" se aplica una variable manipulada analógica. Se admiten valores entre -100 y 100 %.
PER_TM	TIME	T#1s	En el parámetro "Duración de período" se introduce la duración de período constante de la modulación de ancho de impulsos. La duración corresponde al tiempo de muestreo del regulador. La relación entre el tiempo de muestreo del formador de impulsos respecto al tiempo de muestreo del regulador determina la precisión de la modulación de ancho de impulsos. PER_TM >=20*CYCLE
P_B_TM	TIME	T#50ms	En el parámetro "Duración mínima de impulso" o "Duración mínima de pausa" se puede parametrizar una duración de pausa o de impulso mínima. P_B_TM >= CYCLE
RATIOFAC	REAL	1.0	Con el parámetro de entrada "Factor de relación" puede modificarse la relación de la duración de impulsos negativos a impulsos positivos. De ese modo, en un lazo de regulación de temperatura se pueden compensar constantes temporales distintas para calentar y refrigerar (p. ej. proceso con calefacción eléctrica y refrigeración por agua). Se admiten valores comprendidos entre 0.1 y 10.0.
STEP3_ON	BOOL	TRUE	En el parámetro de entrada "Activar la regulación de tres puntos" se activa el correspondiente modo de operación. En la regulación de tres puntos trabajan ambas señales de salida.
ST2BI_ON	BOOL	FALSE	En el parámetro de entrada "Activar la regulación de dos puntos para el rango de valores manipulados bipolar" puede elegirse entre los modos de operación "Regulación de dos puntos para rango de valores manipulados bipolar" y "Regulación de dos puntos para el rango de valores manipulados unipolar". Para ello es necesario que STEP3_ON = FALSE.
MAN_ON	BOOL	FALSE	La activación del parámetro de entrada "Activar modo manual" permite ajustar las señales de salida manualmente.
POS_P_ON	BOOL	FALSE	En el modo manual Regulación de tres puntos se puede controlar la señal de salida QPOS_P en el parámetro de entrada "Activar impulso positivo". En modo manual Regulación de dos puntos QNEG_P siempre está invertido respecto a QPOS_P.
NEG_P_ON	BOOL	FALSE	En el modo manual Regulación de tres puntos se puede controlar la señal de salida QNEG_P en el parámetro de entrada "Activar impulso negativo". En modo manual Regulación de dos puntos QNEG_P siempre está invertido respecto a QPOS_P.
SYN_ON	BOOL	TRUE	La activación del parámetro de entrada "Activar sincronización" permite sincronizar automáticamente la salida de impulsos con el bloque que actualiza la magnitud de entrada INV. De ese modo se garantiza que una magnitud de entrada cambiante también se pueda emitir como impulso con la mayor rapidez posible.

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	BOOL	FALSE	La instrucción tiene una rutina de inicialización que se ejecuta cuando está activada la entrada de rearranque completo.
CYCLE	TIME	T#10ms	El tiempo entre las llamadas del bloque debe ser constante. La entrada "Tiempo de muestreo" indica el tiempo entre las llamadas del bloque. CYCLE >= 1ms

Consulte también

Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

Parámetros de salida de PULSEGEN

Tabla 9-189

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
QPOS_P	BOOL	FALSE	El parámetro de salida "Señal de salida del impulso positivo" está activado cuando debe emitirse un impulso. En la regulación de tres puntos es el impulso positivo. En la regulación de dos puntos, QNEG_P se ajusta siempre invertido respecto a QPOS_P.
QNEG_P	BOOL	FALSE	El parámetro de salida "Señal de salida de impulso negativo" está activado cuando debe emitirse un impulso. En la regulación de tres puntos es el impulso negativo. Durante la regulación de dos puntos, QNEG_P se ajusta siempre invertido respecto a QPOS_P.

Consulte también

Diferencias respecto a PULSEGEN S7-300/400 (Página 5243)

TCONT_CP

Descripción de TCONT_CP

La instrucción TCONT_CP sirve para regular procesos de temperatura con control continuo o por impulsos. El funcionamiento se basa en el algoritmo de regulación PID, que está dotado de funciones adicionales para procesos de temperatura. Para mejorar el comportamiento de regulación en procesos de temperatura, el bloque dispone de una zona de regulación y de una reducción de la acción P en caso de que haya escalones en la consigna.

La instrucción puede ajustar por sí misma los parámetros PI/PID mediante la optimización del regulador.

Aplicación

A cada regulador le corresponde un solo actuador, es decir, con un solo regulador se podrá o bien calentar o bien enfriar. Si se va a utilizar para un proceso de refrigeración, hay que parametrizar GAIN con un valor negativo. La inversión que es parametrizada de esta manera provoca que p. ej. en caso de aumento de la temperatura, aumente la variable manipulada LMN, y con ello la potencia de refrigeración.

Llamada

La instrucción TCONT_CP debe llamarse de forma equidistante. Utilice para ello un nivel de alarmas cíclicas (p. ej. OB35 en S7-300).

La instrucción TCONT_CP tiene una rutina de inicialización, que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado. El integrador se ajusta al valor de inicialización I_ITVALdurante la inicialización. Todas las salidas de señal restantes se ponen a cero. El bloque vuelve a ajustarse a FALSE después de ejecutar la rutina de inicialización COM_RST. Si desea realizar una inicialización con un rearranque completo de la CPU, llame el bloque en el OB 100 con COM_RST = TRUE.

Si se efectúa una llamada a la instrucción TCONT_CP como DB multiinstancia, no se crea ningún objeto tecnológico. No se dispone de interfaz de parametrización y puesta en servicio. Los TCONT_CP se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Consulte también

Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Funcionamiento de TCONT_CP

Rama de consigna

La consigna se introduce en la entrada SP_INT en formato de coma flotante, ya sea físico o porcentual. La consigna y el valor real deben indicarse con la misma unidad en la formación del error de regulación.

Selección del valor real (PVPER_ON)

El valor real se puede leer en formato de coma flotante o en formato de periferia en función de PVPER_ON.

PVPER_ON	Entrada del valor real
TRUE	El valor real se lee a través de la periferia analógica (PEW xxx) en la entrada PV_PER.
FALSE	El valor real se lee en formato de coma flotante en la entrada PV_IN.

Conversión del formato del valor real CRP_IN (PER_MODE)

La función CRP_IN convierte el valor de periferia PV_PER dependiendo del conmutador PER_MODE en un valor en coma flotante según la siguiente regla:

PER_MODE	Salida de CRP_IN	Tipo de entrada analógica	Unidad
0	PV_PER * 0.1	Termopares; PT100/ NI100; Estándar	°C;°F
1	PV_PER * 0.01	PT100/NI100; Climatización;	°C;°F
2	PV_PER * 100/27648	Tensión/intensidad	%

Normalización del valor real PV_NORM (PF_FAC, PV_OFFS)

La función PV_NORM calcula la salida de CRP_IN según la siguiente norma:

"Salida de PV_NORM" = "salida de CRP_IN" * PV_FAC + PV_OFFS

Se puede utilizar para los siguientes propósitos:

- Adaptación del valor real con PV_FAC como factor del valor real y PV_OFFS como offset del valor real
- Normalización de la temperatura en porcentajes
  Desea indicar la consigna en porcentajes y ahora debe convertir el valor de temperatura medido en un valor porcentual.
- Normalización de los valores porcentuales a temperaturas
  Desea indicar la consigna en la magnitud física Temperatura y ahora debe convertir el valor de tensión/intensidad medido en una temperatura.

Cálculo de los parámetros:

- PV_FAC = rango de PV_NORM/ rango de CRP_IN;
- PV_OFFS = UG(PV_NORM) - PV_FAC * UG(CRP_IN);
  con UG: límite inferior

Con los valores predeterminados (PV_FAC = 1.0 y PV_OFFS = 0.0) la normalización está desactivada. El valor real con efectos reales se emite en la salida PV.

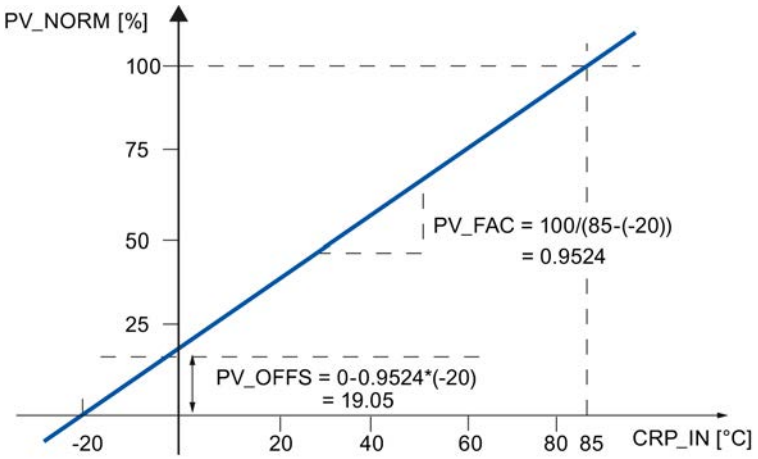
Nota

Durante la regulación de impulsos, el valor real se debe transferir al bloque en una llamada de impulso rápida (motivo: filtración de valores medios). De lo contrario puede disminuir la calidad de la regulación.

Ejemplo de normalización del valor real

Para preseleccionar la consigna en porcentajes con un rango de temperatura de -20 a 85 °C en CRP_IN, normalice el rango de temperatura a valores porcentuales.

En la siguiente figura encontrará un ejemplo para la adaptación de un rango de temperatura de entre -20 y 85 °C a valores de 0 a 100 %:



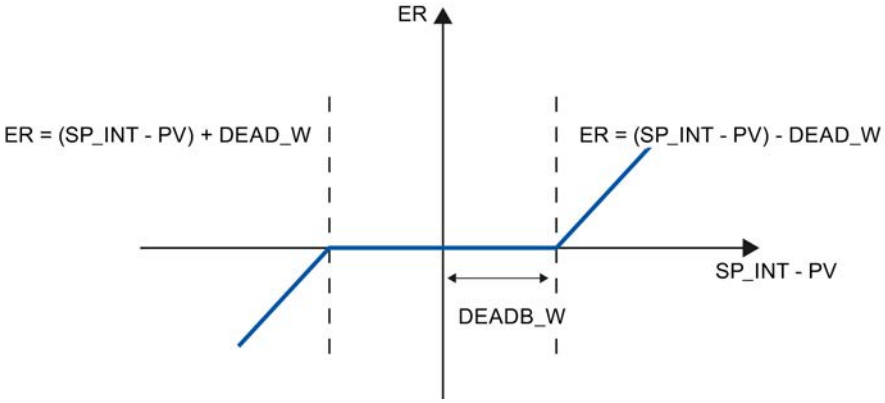
Cálculo de error de regulación

La diferencia entre la consigna y el valor real constituye el error de regulación antes de la zona muerta.

La consigna y el valor real deben indicarse con la misma unidad.

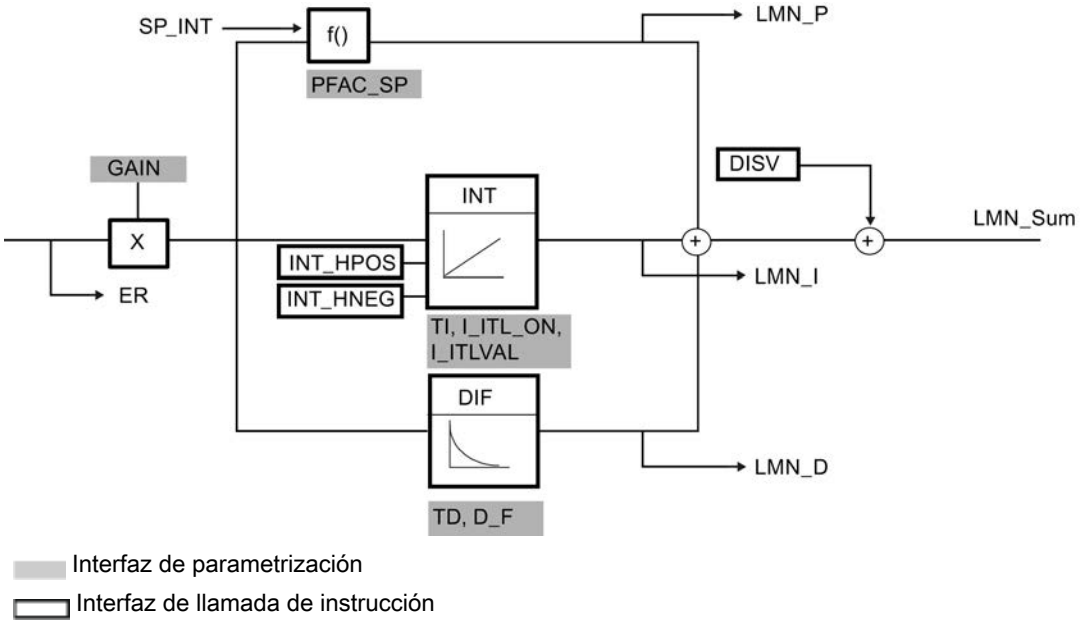
Zona muerta (DEADB_W)

Para suprimir una pequeña oscilación permanente debida a la cuantificación de la magnitud manipulada (p. ej. en una modulación de ancho de impulsos con PULSEGEN), el error de regulación se conduce por una zona muerta (DEADBAND). Con DEADB_W = 0.0 la zona muerta está desactivada. El error de regulación efectivo se muestra en el parámetro ER.



Algoritmo PID

En la siguiente figura se representa el diagrama de bloques del algoritmo PID.



Algoritmo PID (GAIN, TI, TD, D_F)

El algoritmo PID trabaja en el algoritmo de posición. Las acciones proporcional, integral (INT) y derivativa (DIF) están conectadas en paralelo y pueden activarse y desactivarse por separado. De este modo se pueden parametrizar reguladores P, PI, PD y PID.

La optimización del regulador es compatible con los reguladores PI y PID. La inversión del regulador se efectúa mediante un parámetro GAIN negativo (regulador de refrigeración).

Si pone TI y TD a 0.0, obtendrá un regulador P puro en el punto de operación.

La respuesta indicial en el rango de tiempo es:

$$LMN_Sum(t) = GAIN*ER(0)(1+\frac{1}{TI} * t+D_F * e^{\frac{-t}{TD/D_F}}$$

Donde:

LMN_Sum(t) es la magnitud manipulada en modo automático del regulador

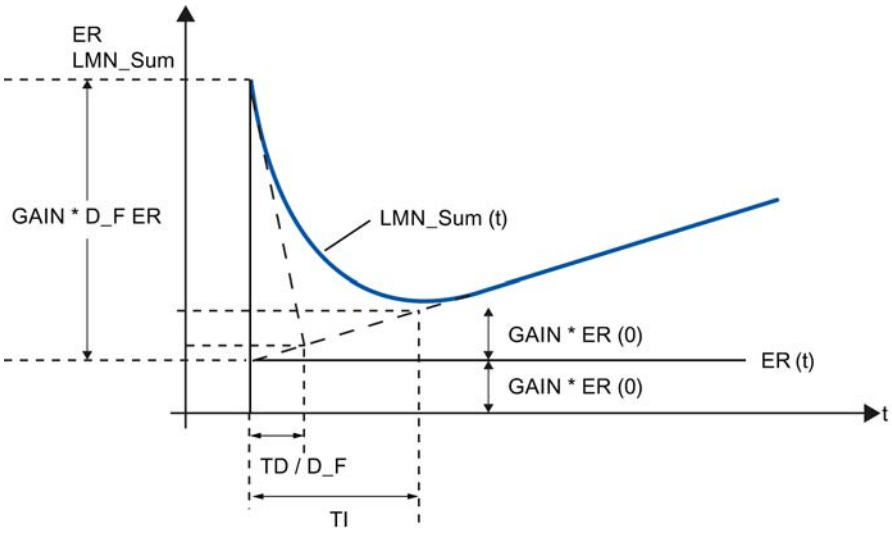
ER (0) es la amplitud del escalón del error de regulación normalizado

GAIN es la ganancia del regulador

TI es el tiempo de acción integral

TD es el tiempo de la acción derivada

D_F es el factor de acción derivativa



Integrador (TI, I_ITL_ON, I_ITLVAL)

En modo Manual, se corrige según: LMN_I = LMN - LMN_P - DISV.

Si se limita la variable manipulada, la acción I se detiene. En caso de un error de regulación que mueva la acción I en el sentido del margen de corrección interno, la acción I se habilitará de nuevo.

Con las siguientes medidas se producen más modificaciones de la acción I:

- Desconexión de la acción I del regulador con TI = 0.0
- Atenuación de la acción P en caso de modificaciones de la consigna
- Zona de regulación
- Los límites de la variable manipulada se pueden modificar online

Atenuación de la acción P en caso de modificaciones de la consigna (PFAC_SP)

Para evitar sobreoscilaciones, es posible atenuar la acción P a través del parámetro "Ganancia proporcional al cambiar la consigna" (PFAC_SP). Mediante PFAC_SP se puede elegir continuamente la intensidad - entre 0.0 y 1.0 - con la que debe actuar la acción P en caso de modificaciones de la consigna:

- PFAC_SP = 1.0: la acción P actúa plenamente al cambiar la consigna
- PFAC_SP = 0.0: la acción P no actúa en caso de modificaciones de la consigna

La atenuación de la acción P se logra mediante una compensación en la acción I.

Diferenciador (TD, D_F)

- Desconexión de la acción D del regulador con TD = 0.0.
- Cuando la acción D está conectada, debe cumplirse la siguiente ecuación:
  TD = 0.5 * CYCLE * D_F

Parametrización de un regulador P o PD con punto de operación

En la interfaz de parametrización, desconecte la acción I (TI = 0.0) y, si es necesario, la acción D (TD = 0.0). Además, efectúe la siguiente parametrización:

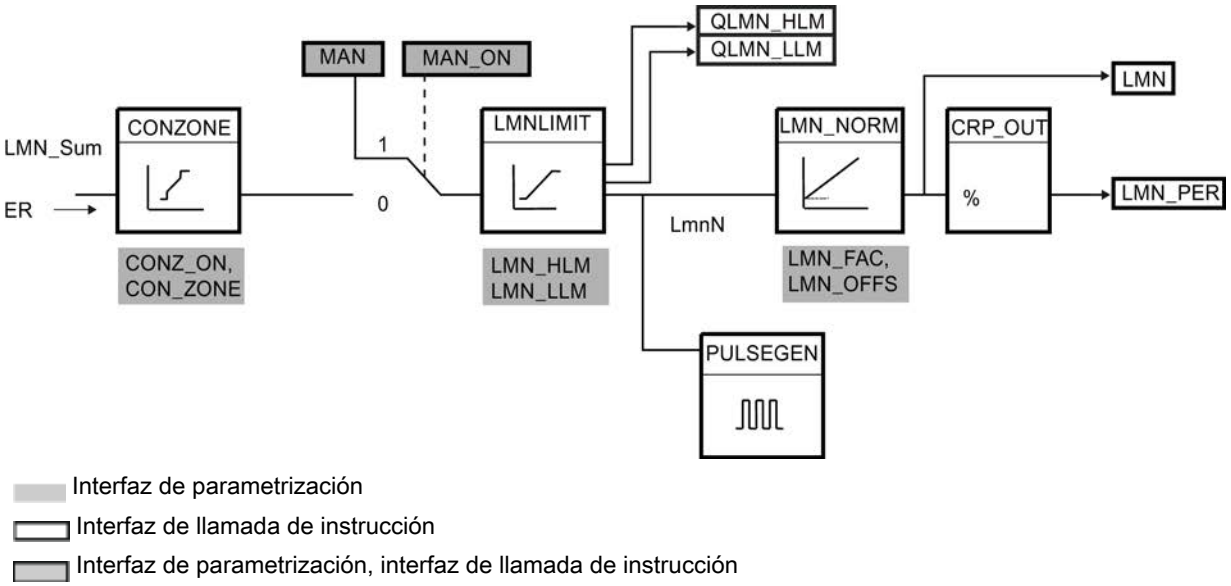
- I_ITL_ON = TRUE
- I_ITLVAL = punto de operación;

Control anticipativo (DISV)

En la entrada DISV se puede aplicar una magnitud perturbadora de forma aditiva.

Cálculo de la variable manipulada

En la siguiente figura se representa el diagrama de bloques del cálculo de la variable manipulada:



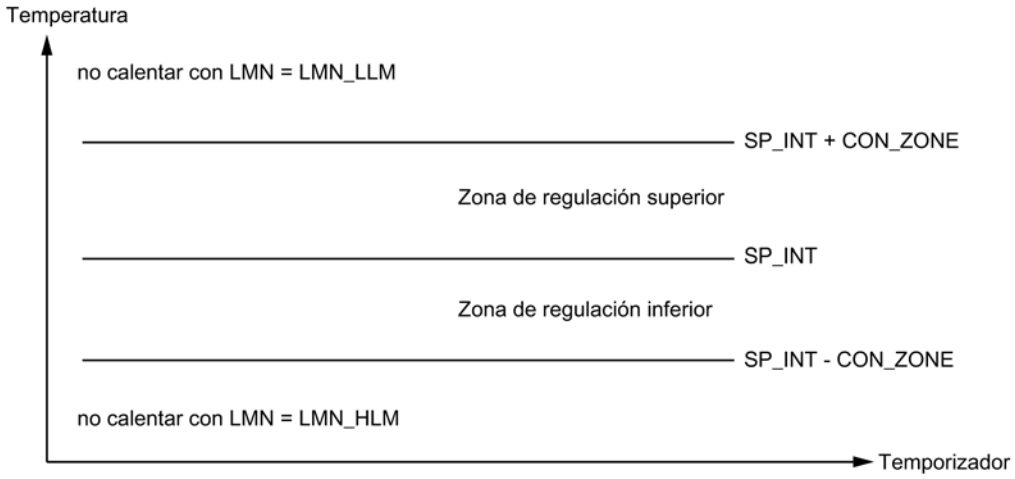
Zona de regulación (CONZ_ON, CON_ZONE)

Si CONZ_ON = TRUE, el regulador trabaja con una zona de regulación. Esto significa que el regulador se controla según el siguiente algoritmo:

- Si el valor real PV rebasa la consigna SP_INT en más de CON_ZONE, se emite el valor LMN_LLM como magnitud manipulada.
- Si el valor real PV se encuentra por debajo de la consigna SP_INT en más de CON_ZONE, se emite el valor LMN_HLM.
- Si el valor real PV se mueve dentro de la zona de regulación (CON_ZONE), la variable manipulada adopta el valor del algoritmo PID LMN_Sum.

Nota

El cambio de la magnitud manipulada de LMN_LLM o LMN_HLM a LMN_Sum se realiza manteniendo una histéresis del 20 % de la zona de regulación.



Nota

Antes de activar a mano la zona de regulación, deberá asegurarse de no ajustar un ancho de zona de regulación demasiado pequeño. Si el ancho de la zona de regulación ajustado fuera insuficiente, se producirán oscilaciones en las curvas de la magnitud manipulada y del valor real.

Ventaja de la zona de regulación

Al entrar en la zona de regulación, la acción D activada provocará una reducción muy rápida de la magnitud manipulada. Por lo tanto, se recomienda utilizar la zona de regulación sólo cuando la acción D esté conectada. Sin zona de regulación, básicamente sólo reduciría la magnitud manipulada la acción P en reducción. La zona de regulación provoca un régimen transitorio rápido sin sobreoscilaciones ni posteriores infraoscilaciones en caso de que la magnitud manipulada mínima o máxima emitida esté a mucha distancia de la magnitud manipulada necesaria para el nuevo punto de operación.

Procesamiento de valores en manual (MAN_ON, MAN)

Es posible conmutar entre modo Manual y modo Automático. En el modo Manual, la magnitud manipulada se corrige según el valor que se ajuste manualmente.

El integrador (INT) se pone internamente a LMN - LMN_P - DISV y el diferenciador (DIF) a 0 y se compensa internamente. La conmutación al modo Automático se efectúa así sin discontinuidad.

Nota

Durante una optimización, el parámetro MAN_ON no tiene efecto alguno.

Limitación de la variable manipulada LMNLIMIT (LMN_HLM, LMN_LLM)

La variable manipulada se limita con la función LMNLIMIT a los límites de la variable manipulada LMN_HLM y LMN_LLM. Los bits de aviso QLMN_HLM y QLMN_LLM indican que se han alcanzado los límites.

Si se limita la variable manipulada, la acción I se detiene. En caso de un error de regulación que mueva la acción I hacia el margen de corrección interno, se volverá a habilitar la acción.

Modificación online de los límites de la variable manipulada

Si se reduce el rango de la variable manipulada y la nueva variable manipulada se encuentra fuera de los límites, se desplaza la acción I, y por tanto, la variable manipulada.

La variable manipulada se reduce en la diferencia entre el límite anterior y el nuevo límite de la variable manipulada. Si la variable manipulada no estaba limitada antes de la modificación, se ajustará exactamente al nuevo límite (en el ejemplo aquí descrito utiliza el límite superior).

Normalización de la variable manipulada LMN_NORM (LMN_FAC, LMN_OFFS)

La función LMN_NORM normaliza la variable manipulada según la siguiente regla:

LMN = LmnN * LMN_FAC + LMN_OFFS

Se puede utilizar para los siguientes propósitos:

- Adaptación de la variable manipulada con LMN_FAC como factor de la variable manipulada y LMN_OFFS como offset de la variable manipulada

La variable manipulada también está disponible en formato de periferia. La función CRP_OUT convierte el valor en coma flotante LMN en un valor de periferia según la siguiente regla:

LMN_PER = LMN * 27648/100

Con los valores predeterminados (LMN_FAC = 1.0 y LMN_OFFS = 0.0) la normalización está desactivada. La variable manipulada que actúa verdaderamente se obtiene en la salida LMN.

Guardar parámetros del regulador SAVE_PAR

Si los parámetros actuales del regulador se consideran válidos, pueden guardarse antes de modificarse manualmente en parámetros de estructura propios previstos para tal fin en el DB

de instancia de la instrucción TCONT_CP. En caso de optimización del regulador, los parámetros guardados se sobrescriben con los valores válidos antes de la optimización.

PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON y CONZONE se escriben en la estructura PAR_SAVE.

Cargar de nuevo parámetros de regulador guardados UNDO_PAR

Esta función permite volver a activar los últimos parámetros guardados del regulador (sólo en modo Manual).

Cambio entre parámetros PI y PID LOAD_PID (PID_ON)

Tras una optimización, los parámetros PI y PID se depositan en las estructuras PI_CON y PID_CON. Con LOAD_PID en función de PID_ON se pueden escribir los parámetros PI o PID en modo manual en los parámetros activos del regulador.

Parámetros PID PID_ON = TRUE	Parámetros PI PID_ON = FALSE
• GAIN = PID_CON.GAIN • TI = PID_CON.TI • TD = PID_CON.TD	• GAIN = PI_CON.GAIN • TI = PI_CON.TI

Nota

Los parámetros del regulador sólo se rescriben con UNDO_PAR o LOAD_PID si la ganancia del regulador no es igual a cero:

Con LOAD_PID se copian los parámetros sólo si GAIN <> 0 (ya sea del registro de parámetros PI o PID). Con ello se tiene en cuenta el caso de que no se haya realizado ninguna optimización o de que falte algún parámetro PID. Si PID_ON = TRUE y PID.GAIN = FALSE, PID_ON se pone en FALSE y los parámetros PI se copian.

- D_F, PFAC_SP se preajustan con la optimización. Posteriormente pueden ser modificados por el usuario. LOAD_PID no modifica estos parámetros.
- La zona de regulación se vuelve a calcular siempre con LOAD_PID (CON_ZONE = 250/GAIN), incluso si CONZ_ON = FALSE.

Consulte también

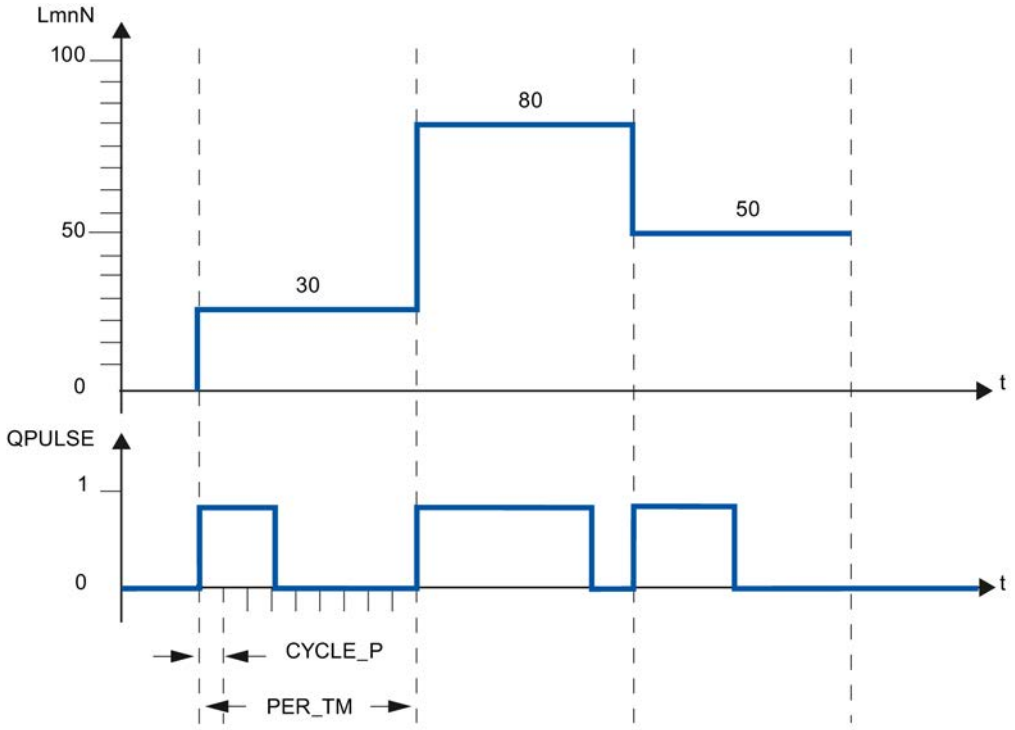
Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Funcionamiento del generador de impulsos

La función PULSEGEN cambia la variable manipulada analógica LmnN mediante una modulación de ancho de impulsos en una secuencia de impulsos con la duración de período PER_TM. PULSEGEN se activa con PULSE_ON = TRUE y se procesa en el ciclo CYCLE_P.



Una variable manipulada LmnN = 30 % y 10 llamadas de PULSEGEN por PER_TM suponen:

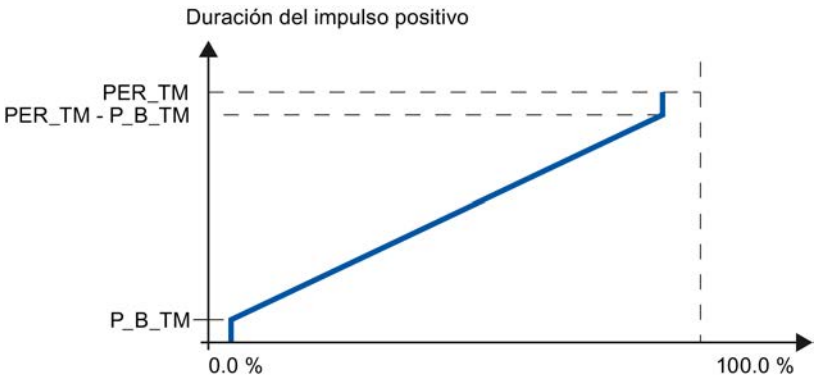
- TRUE en la salida QPULSE para las tres primeras llamadas de PULSEGEN (30 % de 10 llamadas)
- FALSE en la salida QPULSE para las otras siete llamadas de PULSEGEN (70 % de 10 llamadas)

La duración de un impulso por período es proporcional a la magnitud manipulada y se calcula del modo siguiente:

Duración de impulso = PER_TM * LmnN /100

Debido a la supresión de la duración mínima de impulso o pausa, se forman niveles escalonados en el área inicial o final de la característica de conversión.

La figura siguiente representa la regulación a dos puntos con rango unipolar de la variable manipulada
(de 0 % a 100 %):



Duración mínima del impulso o duración mínima de pausa (P_B_TM)

Los tiempos de conexión o desconexión breves merman la vida útil de los elementos finales de control y los dispositivos de control. Esto puede evitarse parametrizando una duración mínima de impulso y pausa P_B_TM.

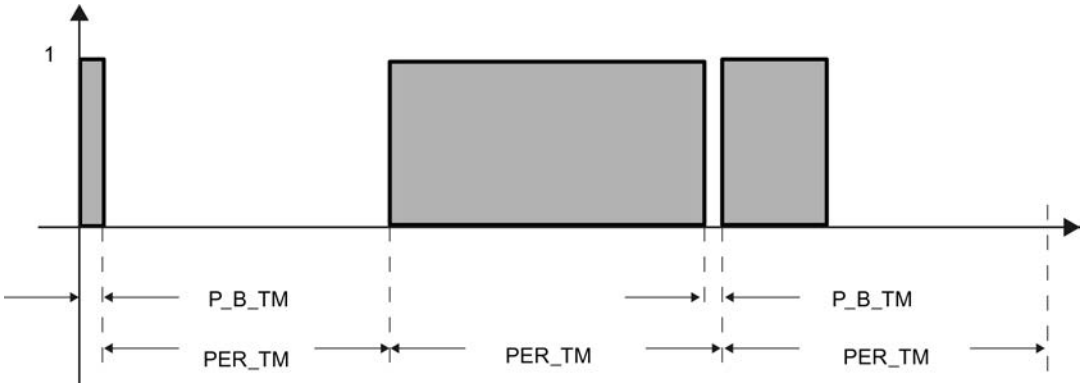
Se suprimen valores absolutos pequeños de la magnitud de entrada LmnN que pudieran generar una duración de impulso menor que P_B_TM.

Los valores de entrada grandes, que generarían una duración de impulso mayor que PER_TM - P_B_TM, se ajustan a 100 %. De este modo se reduce la dinámica de la formación de impulsos.

Para la duración mínima de impulso o pausa se recomiendan valores de ajuste P_B_TM ≤ 0,1 * PER_TM.

Los niveles escalonados en las características que aparecen en la figura anterior son originados por la duración mínima de impulso o por la duración mínima de pausa.

En la siguiente figura se representa el comportamiento de conmutación de la salida de impulso:



Precisión de la formación de impulsos

Cuanto menor sea el ancho de la base de impulso CYCLE_P en comparación con el tiempo de período PER_TM, más exacta será la modulación de ancho de impulsos. Para una regulación con una precisión suficiente sirve la relación siguiente:

CYCLE_P ≤ PER_TM/50

Entonces se cambia la variable manipulada con una resolución de ≤ 2 % en impulsos.

Nota

Si se llama el regulador en el ciclo del generador de impulsos hay que tener en cuenta lo siguiente:

Al llamar el regulador en el ciclo del generador de impulsos se toma el promedio del valor real. Como consecuencia, es posible que en la salida PV haya valores distintos a la entrada PV_IN o PV_PER. Si debe corregirse la consigna, hay que guardar el valor real del parámetro de entrada PV_IN en los tiempos de llamada del procesamiento global del regulador ((QC_ACT = TRUE)). En las llamadas intermedias del generador de impulsos, los parámetros de entrada PV_IN y SP_INT se suministran con el valor real guardado.

Consulte también

Descripción de TCONT_CP (Página 7098)

Funcionamiento de TCONT_CP (Página 7099)

Diagrama de bloques TCONT_CP (Página 7110)

Parámetros de entrada de TCONT_CP (Página 7111)

Parámetros de salida de TCONT_CP (Página 7112)

Parámetros de entrada/salida de TCONT_CP (Página 7113)

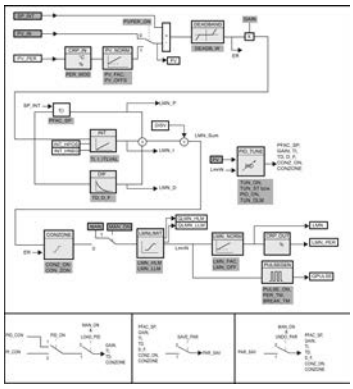
Variables estáticas de TCONT_CP (Página 7113)

Parámetro STATUS_H (Página 7118)

Parámetro STATUS_D (Página 7119)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Diagrama de bloques TCONT_CP



Consulte también

Descripción de TCONT_CP (Página 7098)
Funcionamiento de TCONT_CP (Página 7099)
Funcionamiento del generador de impulsos (Página 7108)
Parámetros de entrada de TCONT_CP (Página 7111)
Parámetros de salida de TCONT_CP (Página 7112)
Parámetros de entrada/salida de TCONT_CP (Página 7113)
Variables estáticas de TCONT_CP (Página 7113)
Parámetro STATUS_H (Página 7118)
Parámetro STATUS_D (Página 7119)
Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Parámetros de entrada de TCONT_CP

Tabla 9-190

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PV_IN	0.0	REAL	0.0	En la entrada "Valor real de la entrada" se puede parametrizar un valor de puesta en marcha, o se puede conectar un valor real externo en coma flotante. Los valores válidos dependen de los sensores utilizados.
PV_PER	4.0	INT	0	El valor real en formato de periferia se conecta al regulador a través de la entrada "Valor real de la periferia".
DISV	6.0	REAL	0.0	Para realizar un control anticipativo, la magnitud perturbadora se conecta a la entrada "Variable perturbadora".
INT_HPOS	10.0	BOOL	FALSE	La salida del integrador se puede bloquear en sentido positivo. Para ello, la entrada INT_HPOS debe estar ajustada a TRUE. En caso de regulación en cascada, se conecta INT_HPOS del regulador piloto a QLMN_HLM del regulador secuencial.
INT_HNEG	10.1	BOOL	FALSE	La salida del integrador se puede bloquear en sentido negativo. Para ello, la entrada INT_HNEG debe estar ajustada a TRUE. En caso de regulación en cascada, se conecta INT_HNEG del regulador piloto a QLMN_LLM del regulador secuencial.
SELECT	12.0	INT	0	Si el formador de impulsos está conectado, existen varias posibilidades de llamar el algoritmo PID y el formador de impulsos: • SELECT = 0: el regulador se llama en un nivel de alarmas cíclicas rápido y se procesan el algoritmo PID y el formador de impulsos. • SELECT = 1: el regulador se llama en el OB1 y sólo se procesa el algoritmo PID. • SELECT = 2: el regulador se llama en un nivel de alarmas cíclicas rápido y sólo se procesa el formador de impulsos. • SELECT = 3: el regulador se llama en un nivel de alarmas cíclicas lento y sólo se procesa el algoritmo PID.

Consulte también

Funcionamiento del generador de impulsos (Página 7108)
Diagrama de bloques TCONT_CP (Página 7110)
Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Parámetros de salida de TCONT_CP

Tabla 9-191

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PV	14.0	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente. Los valores válidos dependen de los sensores utilizados.
LMN	18.0	REAL	0.0	En la salida "Variable manipulada" se indica la variable manipulada realmente efectiva en formato de coma flotante.
LMN_PER	22.0	INT	0	La variable manipulada en formato de periferia se conecta al regulador en la salida "Variable manipulada de la periferia".
QPULSE	24.0	BOOL	FALSE	La variable manipulada se emite con modulación de ancho de impulsos por la salida QPULSE.
QLMN_HLM	24.1	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida QLMN_HLM notifica que se ha alcanzado el límite superior.
QLMN_LLM	24.2	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida QLMN_LLM notifica que se ha alcanzado el límite inferior.
QC_ACT	24.3	BOOL	TRUE	El parámetro indica si se va a procesar la acción del regulador continuo en la siguiente llamada de bloque (sólo relevante si SELECT tiene el valor 0 ó 1).

Consulte también

Funcionamiento del generador de impulsos (Página 7108)
Diagrama de bloques TCONT_CP (Página 7110)
Parámetro STATUS_H (Página 7118)
Parámetro STATUS_D (Página 7119)
Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Parámetros de entrada/salida de TCONT_CP

Tabla 9-192

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
CYCLE	26.0	REAL	0.1 s	Valor predeterminado del tiempo de muestreo para el algoritmo PID. El optimizador calcula el tiempo de muestreo en la fase 1 y lo deposita en CYCLE. CYCLE > 0.001 s
CYCLE_P	30.0	REAL	0.02 s	En esta entrada se indica el tiempo de muestreo para la acción del formador de impulsos. La instrucción TCONT_CP calcula el tiempo de muestreo en la fase 1 y lo deposita en CYCLE_P. CYCLE_P > 0.001 s
SP_INT	34.0	REAL	0.0	La entrada "Consigna interna" sirve para preseleccionar una consigna. Los valores válidos dependen de los sensores utilizados.
MAN	38.0	REAL	0.0	La entrada "Valor manual" sirve para predeterminar un valor manual. En modo automático, se corrige según la variable manipulada.
COM_RST	42.0	BOOL	FALSE	El bloque tiene una rutina de inicialización, que se ejecuta cuando se activa la entrada COM_RST.
MAN_ON	42.1	BOOL	TRUE	Si está activada la entrada "Conectar modo manual", el lazo de regulación está interrumpido. Como variable manipulada se especifica el valor manual MAN.

Consulte también

Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Variables estáticas de TCONT_CP

Tabla 9-193

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
DEADB_W	44.0	REAL	0.0	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. Los valores válidos dependen de los sensores utilizados.
I_ITLVAL	48.0	REAL	0.0	La salida del integrador se puede aplicar en la entrada I_ITL_ON. En la entrada "Valor de inicialización para la acción I" está el valor de inicialización. Durante el rearranque completo COM_RST = TRUE, la acción I se ajusta al valor de inicialización. Se admiten valores entre -100 y 100 %.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMN_HLM	52.0	REAL	100.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior de la variable manipulada" indica el límite superior. LMN_HLM > LMN_LLM
LMN_LLM	56.0	REAL	0.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior de la variable manipulada" indica el límite inferior. LMN_LLM < LMN_HLM
PV_FAC	60.0	REAL	1.0	La entrada "Factor de valor real" se multiplica por "Valor real de la periferia". La entrada sirve para adaptar el rango del valor real.
PV_OFFS	64.0	REAL	0.0	La entrada "Offset de valor real" se suma a "Valor real de periferia". La entrada sirve para adaptar el rango del valor real.
LMN_FAC	68.0	REAL	1.0	La entrada "Factor de la variable manipulada" se multiplica por la variable manipulada. La entrada sirve para adaptar el rango de la variable manipulada.
LMN_OFFS	72.0	REAL	0.0	La entrada "Offset de la variable manipulada" se suma a la variable manipulada. La entrada sirve para adaptar el rango de la variable manipulada.
PER_TM	76.0	REAL	1.0 s	En el parámetro PER_TM se introduce la duración del periodo de la modulación de ancho de impulsos. La relación entre duración de periodo y tiempo de muestreo del formador de impulsos determina la precisión de la modulación de ancho de impulsos. PER_TM ≥ CYCLE
P_B_TM	80.0	REAL	0.02 s	En el parámetro "Duración mínima de impulso" o "Duración mínima de pausa" se puede parametrizar una duración de pausa o de impulso mínima. P_B_TM se limita internamente a > CYCLE_P.
TUN_DLMN	84.0	REAL	20.0	La excitación del proceso para la optimización del regulador se realiza mediante un salto de la variable manipulada de TUN_DLMN. Se admiten valores entre -100 y 100 %.
PER_MODE	88.0	INT	0	En este conmutador se puede indicar el tipo del módulo E/S. El valor real de la entrada PV_PER se normaliza así en la salida PV de la siguiente manera. • PER_MODE = 0: termopares; PT100/NI100; Estándar PV_PER * 0.1 Unidad: °C, °F • PER_MODE = 1: PT100/NI100; Climatización PV_PER * 0.01 Unidad: °C, °F • PER_MODE = 2: Intensidad/tensión PV_PER * 100/27648 Unidad: %
PVPER_ON	90.0	BOOL	FALSE	Si se debe leer el valor real de la periferia, la entrada PV_PER debe estar interconectada con la periferia, y la entrada "Activar valor real de periferia" debe estar activada.
I_ITL_ON	90.1	BOOL	FALSE	La salida del integrador se puede aplicar a la entrada I_ITLVAL. Para ello debe estar activada la entrada "Inicializar acción I".
PULSE_ON	90.2	BOOL	FALSE	Con PULSE_ON = TRUE se activa el formador de impulsos.
TUN_KEEP	90.3	BOOL	FALSE	No se pasa al modo automático hasta que TUN_KEEP sea FALSE.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
ER	92.0	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente. Los valores válidos dependen de los sensores utilizados.
LMN_P	96.0	REAL	0.0	La salida "Acción P" contiene la acción proporcional de la magnitud manipulada.
LMN_I	100.0	REAL	0.0	La salida "Acción I" contiene la acción integral de la magnitud manipulada.
LMN_D	104.0	REAL	0.0	La salida "Acción D" contiene la acción derivativa de la magnitud manipulada.
PHASE	108.0	INT	0	En la salida PHASE se indica la fase de ejecución actual de la optimización del regulador. • PHASE = 0: no hay modo de optimización; modo automático o manual • PHASE = 1: disponibilidad para la optimización; comprobar parámetros, esperar excitación, medir los tiempos de muestreo • PHASE = 2: optimización propiamente dicha: Búsqueda de punto de inversión con variable manipulada constante. Entrada del tiempo de muestreo en el DB de instancia. • PHASE = 3: cálculo de los parámetros del proceso. Guardar los parámetros del regulador válidos antes de la optimización. • PHASE = 4: diseño del regulador • PHASE = 5: corrección del regulador a la nueva variable manipulada • PHASE = 7: verificación del tipo de sistema
STATUS_H	110.0	INT	0	STATUS_H indica un valor de diagnóstico a través de la búsqueda del punto de inversión en el proceso de calentamiento.
STATUS_D	112.0	INT	0	STATUS_D indica un valor de diagnóstico a través del diseño del regulador en el proceso de calentamiento.
QTUN_RUN	114.0	BOOL	0	La optimización se ha iniciado mediante la aplicación de la magnitud manipulada de optimización y en estos momentos se encuentra en la fase 2 (búsqueda del punto de inversión).
PI_CON	116.0	STRUCT		Parámetros PI del regulador
GAIN	+0.0	REAL	0.0	Ganancia PI del regulador %/unidad física
TI	+4.0	REAL	0.0 s	Tiempo de integración PI [s]
PID_CON	124.0	STRUCT		Parámetros PID del regulador
GAIN	+0.0	REAL	0.0	Ganancia PID del regulador
TI	+4.0	REAL	0.0s	Tiempo de integración PID [s]
TD	+8.0	REAL	0.0s	Tiempo de acción derivada PID [s]
PAR_SAVE	136.0	STRUCT		En esta estructura se guardan los parámetros PID.
PFAC_SP	+0.0	REAL	1.0	Ganancia proporcional al cambiar la consigna Se admiten valores comprendidos entre 0.0 y 1.0.
GAIN	+4.0	REAL	0.0	Ganancia del regulador %/unidad física
TI	+8.0	REAL	40.0 s	Tiempo de integración [s]
TD	+12.0	REAL	10.0 s	Tiempo de acción derivada [s]

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
D_F	+16.0	REAL	5.0	Factor de acción derivada Se admiten valores comprendidos entre 5.0 y 10.0.
CON_ZONE	+20.0	REAL	100.0	Ancho de zona de regulación Si el error de regulación es mayor que el ancho de la zona de regulación, se emite el límite superior de la variable manipulada como variable manipulada. Si el error de regulación es menor que el ancho de la zona de regulación negativa, se emite el límite inferior de la variable manipulada como variable manipulada. CON_ZONE ≥ 0.0
CONZ_ON	+24.0	BOOL	FALSE	Activar zona de regulación
PFAC_SP	162.0	REAL	1.0	PFAC_SP indica la acción P efectiva en caso de modificación de la consigna. Se ajusta entre 0 y 1. • 1: La acción P también está completamente operativa en caso de modificaciones de la consigna. • 0: La acción P no está operativa en caso de modificaciones de la consigna. Se admiten valores comprendidos entre 0.0 y 1.0.
GAIN	166.0	REAL	2.0	La entrada "Ganancia proporcional" indica la ganancia del regulador. El sentido de actuación del regulador se invierte mediante el signo negativo de GAIN. %/unidad física
TI	170.0	REAL	40.0 s	La entrada "Tiempo de integración" (tiempo de acción integral) determina el comportamiento temporal del integrador.
TD	174.0	REAL	10.0 s	La entrada "Tiempo de acción derivada" (tiempo de anticipación) determina el comportamiento temporal del diferenciador.
D_F	178.0	REAL	5.0	El factor de acción derivada determina el retardo de la acción D. D_F = tiempo de acción derivada/"Retardo de la acción D" Se admiten valores comprendidos entre 5.0 y 10.0.
CON_ZONE	182.0	REAL	100.0	Si el error de regulación es mayor que el ancho de la zona de regulación, se emite el límite superior de la variable manipulada como variable manipulada. Si el error de regulación es menor que el ancho de la zona de regulación negativa, se emite el límite inferior de la variable manipulada como variable manipulada. Los valores válidos dependen de los sensores utilizados.
CONZ_ON	186.0	BOOL	FALSE	Con CONZ_ON =TRUE se puede activar la zona de regulación.
TUN_ON	186.1	BOOL	FALSE	Si TUN_ON=TRUE se toma el promedio de la variable manipulada hasta que se aplica la excitación de la variable manipulada TUN_DLMNmediante un salto de consigna o TUN_ST=TRUE.
TUN_ST	186.2	BOOL	FALSE	Si la consigna debe permanecer constante durante la optimización del regulador en el punto de operación, se aplica un salto de la variable manipulada de TUN_DLMN mediante TUN_ST=1.
UNDO_PAR	186.3	BOOL	FALSE	Carga los parámetros del regulador PFAC_SP, GAIN, TI, TD, D_F CONZ_ON y CON_ZONE desde la estructura de datos PAR_SAVE (sólo en modo manual).
SAVE_PAR	186.4	BOOL	FALSE	Guarda los parámetros del regulador PFAC_SP, GAIN, TI, TD, D_F, CONZ_ON y CON_ZONE en la estructura de datos PAR_SAVE.
LOAD_PID	186.5	BOOL	FALSE	Carga los parámetros del regulador GAIN, TI, TD en función de PID_ON desde la estructura de datos PI_CON o bien PID_CON (sólo en modo manual)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PID_ON	186.6	BOOL	TRUE	En la entrada PID_ON se puede determinar si el regulador optimizado debe trabajar como regulador PI o PID. • Regulador PID: PID_ON = TRUE • Regulador PI: PID_ON = FALSE Puede ocurrir que en algunos tipos de proceso sólo se diseñe un regulador PI a pesar de PID_ON = TRUE.
GAIN_P	188.0	REAL	0.0	Ganancia del proceso identificada. En el tipo de sistema I, se tiende a estimar un valor demasiado bajo para GAIN_P.
TU	192.0	REAL	0.0	Tiempo de retardo identificado del proceso. TU ≥ 3*CYCLE
TA	196.0	REAL	0.0	Tiempo de compensación identificado del proceso. En el tipo de sistema I, se tiende a estimar un valor demasiado bajo para TA.
KIG	200.0	REAL	0.0	Subida máxima del valor real con una excitación de la variable manipulable de 0 a 100 % [1/s] GAIN_P = 0.01 * KIG * TA
N_PTN	204.0	REAL	0.0	El parámetro indica el orden del proceso. También se pueden utilizar "valores no enteros". Se admiten valores comprendidos entre 1.01 y 10.0.
TM_LAG_P	208.0	REAL	0.0	Constante temporal de un modelo PTN (valores significativos sólo para N_PTN >= 2).
T_P_INF	212.0	REAL	0.0	Tiempo desde la excitación del proceso hasta el punto de inversión.
P_INF	216.0	REAL	0.0	Modificación del valor real desde la excitación del proceso hasta el punto de inversión. Los valores válidos dependen de los sensores utilizados.
LMN0	220.0	REAL	0.0	Variable manipulada al principio de la optimización Se calcula en la fase 1 (valor medio). Se admiten valores entre 0 y 100 %.
PV0	224.0	REAL	0.0	Valor real al principio de la optimización
PVDT0	228.0	REAL	0.0	Subida del valor real al iniciar la optimización [1/s] Con adaptación de signo.
PVDT	232.0	REAL	0.0	Subida momentánea del valor real [1/s] Con adaptación de signo.
PVDT_MAX	236.0	REAL	0.0	Modificación máx. del valor real por segundo [1/s] Derivación máxima del valor real en el punto de inversión (con adaptación de signo, siempre > 0); se utiliza para calcular TU y KIG.
NOI_PVDT	240.0	REAL	0.0	Nivel de ruido en PVDT_MAX en % Cuanto mayor sea el nivel de ruido, menos precisos (de efecto más suave) serán los parámetros del regulador.
NOISE_PV	244.0	REAL	0.0	Ruido absoluto en el valor real Diferencia entre el valor real máximo y mínimo en la fase 1.
FIL_CYC	248.0	INT	1	Número de ciclos del filtro de valores medios Se calcula un valor medio del valor real a partir de FIL_CYC ciclos. En caso necesario, FIL_CYC se aumenta automáticamente entre 1 y máx. 1024.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
POI_CMAX	250.0	INT	2	Número máx. de ciclos tras el punto de inversión Este tiempo se utiliza para encontrar otro punto de inversión (es decir, uno mejor) en caso de ruidos de medición. Sólo entonces concluye la optimización.
POI_CYCL	252.0	INT	0	Número de ciclos tras el punto de inversión

Consulte también

Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Parámetro STATUS_H

STATUS_H	Descripción	Solución
0	Parámetros del regulador predeterminados, ningún parámetro o ninguno nuevo	
10000	Optimización finalizada + parámetros adecuados encontrados para el regulador	
2xxxx	Optimización finalizada + parámetros del regulador inseguros	
2xx2x	Punto de inversión no alcanzado (sólo en caso de excitación mediante salto de consigna)	Si el regulador oscila, atenuar los parámetros del regulador o repetir el ensayo con una diferencia menor de la variable manipulada TUN_DLMN.
2x1xx	Error de estimación (TU < 3*CYCLE)	Reducir CYCLE y repetir el ensayo. Caso excepcional proceso PT1 puro: No repetir el ensayo, atenuar los parámetros del regulador si procede.
2x3xx	Error de estimación TU demasiado grande	Repetir el ensayo con mejores condiciones.
21xxx	Error de estimación N_PTN < 1	Repetir el ensayo con mejores condiciones.
22xxx	Error de estimación N_PTN > 10	Repetir el ensayo con mejores condiciones.
3xxxx	Optimización cancelada en fase 1 debido a parametrización errónea:	

STATUS_H	Descripción	Solución
30002	Diferencia efectiva de la variable manipulada < 5 %	Corregir la diferencia de la variable manipulada TUN_DLMN.
30005	Los tiempos de muestreo CYCLE y CYCLE_P difieren en más del 5% de los valores medidos.	Compare CYCLE y CYCLE_P con el tiempo de ciclo del nivel de alarmas cíclicas y considere los posibles distribuidores de llamada. Compruebe el grado de utilización de la CPU. Una CPU con una carga excesiva conlleva tiempos de muestreo excesivos, que no concuerdan con CYCLE ni CYCLE_P.

Nota

Si se cancela la optimización en la fase 1 ó 2, entonces STATUS_H = 0. Sin embargo, STATUS_D sigue indicando el estado del último cálculo del regulador.

Cuanto mayor sea el valor de STATUS_D, mayor será el orden del proceso regulado, mayor la relación TU/TA y más suaves los parámetros del regulador.

Consulte también

Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

Parámetro STATUS_D

STATUS_D	Descripción
0	No se han calculado parámetros del regulador
110	N_PTN <= 1.5 Tipo de sistema I rápido
121	N_PTN > 1.5 Tipo de sistema I
200	N_PTN > 1.9 Tipo de sistema II (rango de transición)
310	N_PTN >= 2.1 Tipo de sistema III rápido
320	N_PTN > 2.6 Tipo de sistema III
111, 122, 201, 311, 321	Los parámetros se han corregido en la fase 7.

Nota

Cuanto mayor sea el valor de STATUS_D, mayor será el orden del proceso regulado, mayor la relación TU/TA y más suaves los parámetros del regulador.

Consulte también

Funcionamiento del generador de impulsos (Página 7108)

Diagrama de bloques TCONT_CP (Página 7110)

Diferencias respecto a TCONT_CP S7-300/400 (Página 5243)

TCONT_S

Descripción de TCONT_S

La instrucción TCONT_S permite regular lazos de temperatura técnicos con señales de salida binarias de la variable manipulada para actuadores integrantes en los sistemas de automatización SIMATIC S7. El funcionamiento se basa en el algoritmo de regulación PI del regulador de muestreo. El regulador opera sin realimentación de posición.

Aplicación

El regulador también se puede utilizar en una cascada de reguladores como posicionador subordinado. La entrada de la consigna SP_INT predetermina la posición del actuador. En este caso, tanto la entrada de valor real como el parámetro TI tienen que estar a 0 (tiempo integral). Un ejemplo de aplicación sería una regulación de la temperatura con regulación de la potencia calorífica a través de un control pulso/pausa y con regulación de la potencia frigorífica a través de una válvula de mariposa. Para cerrar completamente la válvula, la magnitud manipulada debe pasar a ser negativa (ER*GAIN).

Llamada

La instrucción TCONT_S debe llamarse de forma equidistante. Utilice para ello un nivel de alarmas cíclicas (p. ej. OB35 en S7-300). El tiempo de muestreo se predetermina en el parámetro CYCLE.

Si se efectúa una llamada a la instrucción TCONT_S como DB multiinstancia, no se crea ningún objeto tecnológico. No se dispone de interfaz de parametrización y puesta en servicio. Los TCONT_S se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Tiempo de muestreo CYCLE

El tiempo de muestreo CYCLE debe coincidir con la diferencia de tiempo entre dos llamadas (tiempo de ciclo del OB de alarma cíclica observando los factores de ciclo).

El tiempo de muestreo del regulador no debería exceder más del 10 % del tiempo integral determinado para el regulador (TI). Sin embargo, normalmente se debe ajustar un tiempo de muestreo mucho menor para garantizar la precisión exigida del regulador paso a paso.

Precisión exigida G	MTR_TM	CYCLE = MTR_TM*G	Comentario
0,5 %	10 s	0,05 s	El tiempo de muestreo viene determinado por la precisión exigida del regulador paso a paso.

Arranque

La instrucción TCONT_S incluye una rutina de inicialización, que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado. El bloque vuelve a ajustarse a FALSE después de ejecutar la rutina de inicialización COM_RST. Todas las salidas se ajustan a sus respectivos valores iniciales. Si desea realizar una inicialización con un rearranque completo de la CPU, llame la instrucción en el OB 100 con COM_RST = TRUE.

Consulte también

Diagrama de bloques de TCONT_S (Página 7124)

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Funcionamiento de TCONT_S

Rama de consigna

La consigna se introduce en la entrada SP_INT en formato de coma flotante, ya sea físico o porcentual. La consigna y el valor real deben tener la misma unidad en la formación del error de regulación.

Selección del valor real (PVPER_ON)

El valor real se puede leer en formato de coma flotante o en formato de periferia en función de PVPER_ON.

PVPER_ON	Introducción del valor real
TRUE	El valor real se lee a través de la periferia analógica (PEW xxx) en la entrada PV_PER.
FALSE	El valor real se lee en formato de coma flotante en la entrada PV_IN.

Conversión del formato del valor real CRP_IN (PER_MODE)

La función CRP_IN convierte el valor de periferia PV_PER dependiendo del conmutador PER_MODE en un valor en coma flotante según la siguiente regla:

PER_MODE	Salida de CRP_IN	Tipo de entrada analógica	Unidad
0	PV_PER * 0.1	Termopares; PT100/ NI100; Estándar	°C;°F
1	PV_PER * 0.01	PT100/NI100; Climatización;	°C;°F
2	PV_PER * 100/27648	Tensión/intensidad	%

Normalización del valor real PV_NORM (PF_FAC, PV_OFFS)

La función PV_NORM calcula la salida de CRP_IN según la siguiente norma:

"Salida de PV_NORM" = "salida de CRP_IN" * PV_FAC + PV_OFFS

Se puede utilizar para los siguientes propósitos:

- Adaptación del valor real con PV_FAC como factor del valor real y PV_OFFS como offset del valor real
- Normalización de la temperatura en porcentajes
  Desea indicar la consigna en porcentajes y ahora debe convertir el valor de temperatura medido en un valor porcentual.
- Normalización de los valores porcentuales a temperaturas
  Desea indicar la consigna en la magnitud física Temperatura y ahora hay que convertir el valor de tensión/intensidad medido en una temperatura.

Cálculo de los parámetros:

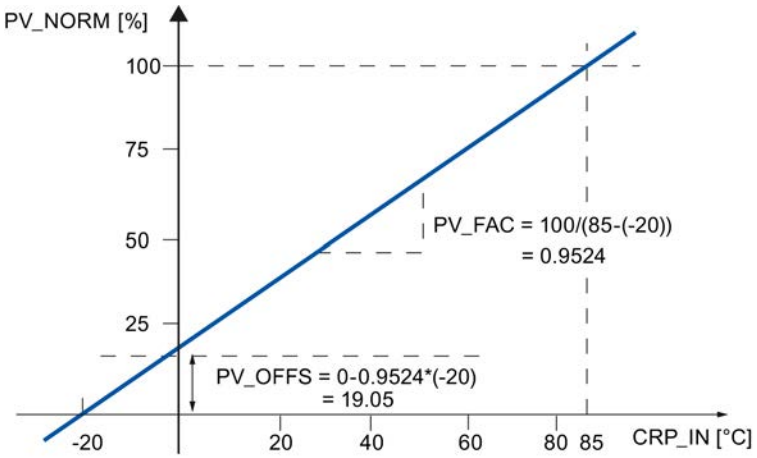
- PV_FAC = rango de PV_NORM/ rango de CRP_IN;
- PV_OFFS = LI(PV_NORM) - PV_FAC * LI(CRP_IN);
  siendo LI: límite inferior

Con los valores predeterminados (PV_FAC = 1.0 y PV_OFFS = 0.0) la normalización está desactivada. El valor real con efectos reales se emite en la salida PV.

Ejemplo de normalización del valor real

Para preseleccionar la consigna en porcentajes con un rango de temperatura de -20 a 85 °C en CRP_IN, normalice el rango de temperatura a valores porcentuales.

En la siguiente figura encontrará un ejemplo para la adaptación de un rango de temperatura de entre -20 y 85 °C a valores de 0 a 100 %:



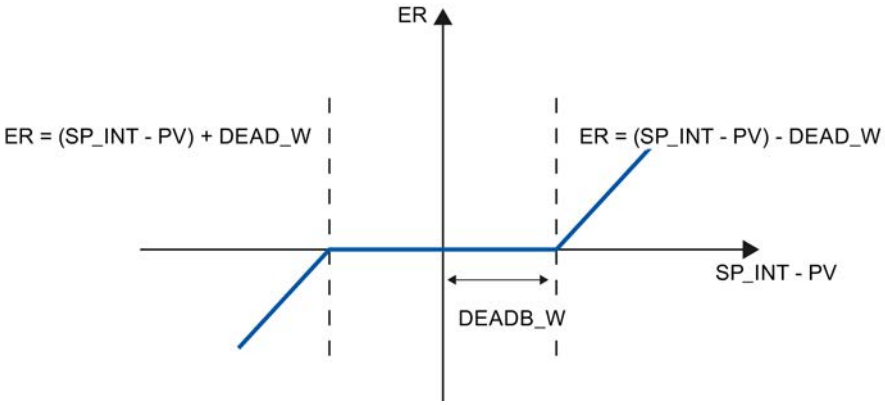
Cálculo de error de regulación

La diferencia entre la consigna y el valor real constituye el error de regulación antes de la zona muerta.

La consigna y el valor real deben estar indicados con la misma unidad.

Zona muerta (DEADB_W)

Para suprimir una pequeña oscilación permanente debida a la cuantificación de la magnitud manipulada (p. ej. en una modulación de ancho de impulsos con PULSEGEN), el error de regulación se conduce por una zona muerta (DEADBAND). Con DEADB_W = 0.0 la zona muerta está desactivada.



Algoritmo del regulador paso a paso PI

La instrucción TCONT_S opera sin realimentación de posición. La acción I del algoritmo PI y la realimentación de posición estimada se calculan en un integrador (INT) y se comparan como valor de realimentación con la acción P restante. La diferencia pasa a un elemento de tres puntos (THREE_ST) y a un generador de impulsos (PULSEOUT), que genera los impulsos para la servoválvula. Mediante una adaptación del umbral de respuesta del elemento de tres niveles se reduce la frecuencia de conmutación del regulador.

Atenuación de la acción P en caso de modificaciones de la consigna

Para evitar sobreoscilaciones, es posible atenuar la acción P a través del parámetro "Ganancia proporcional al cambiar la consigna" (PFAC_SP). Mediante PFAC_SP se puede elegir entonces continuamente la intensidad - entre 0.0 y 1.0 - con la que debe actuar la acción P en caso de modificaciones de la consigna:

- PFAC_SP = 1.0: la acción P actúa plenamente al cambiar la consigna
- PFAC_SP = 0.0: sin acción P en caso de modificación de la consigna

Un valor de PFAC_SP < 1.0 puede reducir las sobreoscilaciones como en el caso de un regulador continuo si el tiempo de funcionamiento del motor MTR_TM es corto con respecto al tiempo de compensación TA, y la relación es TU/TA < 0.2. Si MTR_TM alcanza el 20 % de TA, sólo se puede lograr una pequeña mejora.

Control anticipativo

En la entrada DISV se puede aplicar una magnitud perturbadora de manera aditiva.

Procesamiento de valores en manual (LMNS_ON, LMNUP, LMNDN)

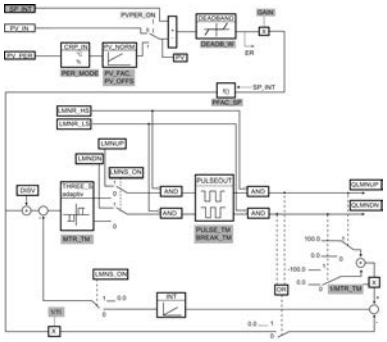
LMNS_ON permite conmutar entre los modos manual y automático. En el modo manual, el elemento final de control se detiene y el integrador (INT) se pone a 0 de forma interna. Mediante LMNUP y LMNDN se puede mover el elemento final de control ARRIBA y ABAJO. La conmutación al modo Automático se efectúa con discontinuidad. El error de regulación existente conduce, a través de GAIN, a una modificación en forma de salto de la magnitud manipulada interna. Mediante el elemento final de control de acción integral sólo se obtiene un control del proceso en forma de rampa.

Consulte también

Diagrama de bloques de TCONT_S (Página 7124)

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Diagrama de bloques de TCONT_S



Interfaz de parametrización

Interfaz de llamada de instrucción

Interfaz de parametrización, interfaz de llamada de instrucción

Consulte también

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Descripción de TCONT_S (Página 7120)

Funcionamiento de TCONT_S (Página 7121)

Parámetros de entrada de TCONT_S (Página 7125)

Parámetros de salida de TCONT_S (Página 7126)

Parámetros de entrada/salida de TCONT_S (Página 7126)

Variables estáticas de TCONT_S (Página 7127)

Parámetros de entrada de TCONT_S

Tabla 9-194

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
CYCLE	0.0	REAL	0.1 s	En esta entrada se indica el tiempo de muestreo para el regulador. CYCLE ≥ 0.001
SP_INT	4.0	REAL	0.0	La entrada "Consigna interna" sirve para preseleccionar una consigna. Los valores válidos dependen de los sensores utilizados.
PV_IN	8.0	REAL	0.0	En la entrada "Valor real de la entrada" se puede parametrizar un valor de puesta en marcha, o se puede conectar un valor real externo en coma flotante. Los valores válidos dependen de los sensores utilizados.
PV_PER	12.0	INT	0	El valor real en formato de periferia se conecta al regulador a través de la entrada "Valor real de la periferia".
DISV	14.0	REAL	0.0	Para realizar un control anticipativo, la magnitud perturbadora se conecta a la entrada "Variable perturbadora".
LMNR_HS	18.0	BOOL	FALSE	La señal "Servoválvula en el tope superior" se aplica en la entrada "Señal de tope superior de la realimentación de posición". • LMNR_HS=TRUE: la servoválvula se encuentra en el tope superior.
LMNR_LS	18.1	BOOL	FALSE	La señal "Servoválvula en el tope inferior" se aplica en la entrada "Señal de tope inferior de la realimentación de posición". • LMNR_LS=TRUE: la servoválvula se encuentra en el tope inferior.
LMNS_ON	18.2	BOOL	TRUE	En la entrada "Activar el modo manual de las señales de la variable manipulada" el procesamiento de las señales manipuladas pasa a manual.
LMNUP	18.3	BOOL	FALSE	En el modo manual de las señales de la variable manipulada se controla la señal de salida QLMNUP en la entrada "Señal alta de variable manipulada".
LMNDN	18.4	BOOL	FALSE	En el modo manual de las señales de la variable manipulada se controla la señal de salida QLMNDN en la entrada "Señal baja de variable manipulada".

Consulte también

Diagrama de bloques de TCONT_S (Página 7124)

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Parámetros de salida de TCONT_S

Tabla 9-195

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QLMNUP	20.0	BOOL	FALSE	Si la salida "Señal alta de variable manipulada" está activada, la válvula de control debe abrirse.
QLMNDN	20.1	BOOL	FALSE	Si está activada la salida "Señal baja de la variable manipulada", la válvula de control debe cerrarse.
PV	22.0	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	26.0	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.

Consulte también

Diagrama de bloques de TCONT_S (Página 7124)

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Parámetros de entrada/salida de TCONT_S

Tabla 9-196

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	30.0	BOOL	FALSE	El bloque tiene una rutina de inicialización, que se ejecuta cuando está activada la entrada COM_RST.

Consulte también

Diagrama de bloques de TCONT_S (Página 7124)

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Variables estáticas de TCONT_S

Tabla 9-197

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PV_FAC	32.0	REAL	1.0	La entrada "Factor de valor real" se multiplica por el valor real. La entrada sirve para adaptar el rango del valor real.
PV_OFFS	36.0	REAL	0.0	La entrada "Offset de valor real" se suma al valor real. La entrada sirve para adaptar el rango del valor real. Los valores válidos dependen de los sensores utilizados.
DEADB_W	40.0	REAL	0.0	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. DEADB_W ≥ 0.0
PFAC_SP	44.4	REAL	1.0	PFAC_SP indica la acción P efectiva en caso de modificación de la consigna. • 1: La acción P está completamente operativa en caso de modificaciones de la consigna. • 0: La acción P no está operativa en caso de modificaciones de la consigna. Se admiten valores comprendidos entre 0.0 y 1.0.
GAIN	48.0	REAL	2.0	La entrada "Ganancia proporcional" indica la ganancia del regulador. El sentido de actuación del regulador se invierte mediante el signo negativo de GAIN. %/unidad física
TI	52.0	REAL	40.0 s	La entrada "Tiempo de integración" (tiempo de acción integral) determina el comportamiento temporal del integrador.
MTR_TM	56.0	REAL	30 s	En el parámetro "Tiempo de posicionamiento del motor" se introduce el tiempo de operación de la servoválvula de un tope al otro. MTR_TM ≥ CYCLE
PULSE_TM	60.0	REAL	0.0 s	En el parámetro "Duración mínima de impulso" se puede parametrizar una duración de impulso mínima.
BREAK_TM	64.0	REAL	0.0 s	En el parámetro "Duración mínima de pausa" se puede parametrizar una longitud de pausa mínima.
PER_MODE	68.0	INT	0	En este conmutador se puede indicar el tipo del módulo E/S. El valor real de la entrada PV_PER se normaliza así en la salida PV de la siguiente manera. • PER_MODE = 0: termopares; PT100/NI100; Estándar PV_PER * 0.1 Unidad: °C, °F • PER_MODE = 1: PT100/NI100; Climatización PV_PER * 0.01 Unidad: °C, °F • PER_MODE = 2: Intensidad/tensión PV_PER * 100/27648 Unidad: %
PVPER_ON	70.0	BOOL	FALSE	Si se debe leer el valor real de la periferia, la entrada PV_PER debe estar interconectada con la periferia, y la entrada "Activar valor real de periferia" debe estar activada.

Consulte también

Diagrama de bloques de TCONT_S (Página 7124)

Diferencias respecto a TCONT_S S7-300/400 (Página 5244)

Funciones de sistema integradas

CONT_C_SF

CONT_C_SF

La instrucción CONT_C_SF está integrada en las CPU compactas. No es necesario transferir la instrucción al cargar en la CPU. Las funciones corresponden a la instrucción CONT_C.

Consulte también

Descripción de CONT_C (Página 7078)

Funcionamiento de CONT_C (Página 7079)

Diagrama de bloques de CONT_C (Página 7081)

Parámetros de entrada de CONT_C (Página 7081)

Parámetros de salida de CONT_C (Página 7083)

CONT_S_SF

CONT_S_SF

La instrucción CONT_S_SF está integrada en las CPU compactas. No es necesario transmitir la instrucción al cargar en la CPU. Las funciones corresponden a la instrucción CONT_S.

Consulte también

Descripción de CONT_S (Página 7084)

Funcionamiento de CONT_S (Página 7084)

Diagrama de bloques de CONT_S (Página 7086)

Parámetros de entrada de CONT_S (Página 7086)

Parámetros de salida de CONT_S (Página 7088)

PULSEGEN_SF

PULSEGEN_SF

La instrucción PULSEGEN_SF está integrada en las CPU compactas. No es necesario transmitir la instrucción al cargar en la CPU. Las funciones corresponden a la instrucción PULSEGEN.

Consulte también

Descripción de PULSEGEN (Página 7088)

Funcionamiento de PULSEGEN (Página 7089)

Modos de operación de PULSEGEN (Página 7093)

Regulación de tres puntos (Página 7093)

Regulación de dos puntos (Página 7095)

Parámetros de entrada de PULSEGEN (Página 7097)

Parámetros de salida de PULSEGEN (Página 7098)

PID Self-Tuner

TUN_EC

Descripción de TUN_EC

La instrucción TUN_EC permite ajustar un regulador PID continuo. Según el regulador, TUN_EC se activa con una de las siguientes instrucciones:

- Reguladores de impulsos FM 355 C, FM 355 S y reguladores de impulsos FM 455 C o FM 455 S
  - PID_FM
- Funciones básicas PID
  - CONT_C Con o sin PULSEGEN
- Standard PID Control (paquete opcional PID Professional)
  - PID_CP
- Modular PID Control (paquete opcional PID Professional)
  - PID
  - LMNGEN_C

Llamada

TUN_EC se llama en una base de tiempo constante del tiempo de ciclo del OB invocante (preferentemente en un OB de alarma cíclica).

Con ADAPT1ST = TRUE se ejecuta la rutina de inicialización de la instrucción TUN_EC. Se llevan a cabo los siguientes ajustes predeterminados:

- Se conmuta al regulador PI. Para ello se transfieren al regulador los parámetros de PI_CON. Si los parámetros PI_CON.GAIN y PI_CON.TI tienen ambos el valor 0.0, se les asignarán los valores predeterminados PI_CON.GAIN = 1.5 y PI_CON.TI = 3600.0 s.
- PROCESS.GAIN = 999.0
- RATIOFAC = 1.0
- STATUS_H = 0; STATUS_D = 0; STATUS_C = 0

Nota

A menos que desee hacerlo expresamente, en caso de rearranque completo (en caliente) de la CPU **no** llame TUN_ES como es habitual con otras instrucciones, pues de lo contrario se perderán todos los parámetros de proceso.

Si se efectúa una llamada de la instrucción TUN_EC como DB multiinstancia, no se crea ningún objeto tecnológico. No existe interfaz de puesta en servicio. TUN_EC debe ponerse en servicio mediante una tabla de observación.

Tiempo de muestreo CYCLE

El tiempo de muestreo se indica en el parámetro CYCLE. El tiempo de muestreo no debe ser superior al 10% del tiempo de integración calculado del regulador. Debe coincidir con la diferencia de tiempo entre dos llamadas (tiempo de ciclo del OB de alarma cíclica considerando los factores de ciclo).

Para identificar de modo fiable los parámetros de proceso, debe muestrearse por lo menos 10 ó 20 veces la constante de tiempo más pequeña del proceso (el tiempo de retardo).

Si se usan los módulos de regulación FM355/455, el tiempo de muestreo de TUN_EC debe ser aproximadamente igual al tiempo de ciclo resultante de los reguladores en el FM (Ajuste de parámetros > Parámetros del módulo).

Comportamiento en caso de error

El bloque no evalúa la palabra de notificación de error RET_VAL.

Consulte también

Funcionamiento de TUN_EC (Página 7133)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de salida de TUN_EC (Página 7140)

Parámetros de entrada/salida de TUN_EC (Página 7142)

Parámetro TUN_EC.PHASE (Página 7144)

Parámetro TUN_EC.STATUS_H (Página 7146)

Parámetro TUN_EC.STATUS_C (Página 7147)

Parámetro TUN_EC.STATUS_D (Página 7148)

Campo de aplicación de TUN_EC

El Self-Tuner resulta especialmente conveniente para

- Regulaciones de temperatura, pero también para
- Regulaciones de nivel de llenado y
- Regulaciones de caudal

Dentro de las regulaciones de caudal se distingue si debe regularse la servoválvula o si tras ésta se inicia un proceso sujeto a retardo. Si se trata de una regulación puramente de la servoválvula, es Self-Tuner no puede utilizarse.

Requisitos del proceso

El proceso debe cumplir las siguientes condiciones:

- **Comportamiento transitorio asintótico, estable y sujeto a retardo**
  Después de un escalón de la magnitud manipulada, el valor real debe pasar a un estado estacionario. Con ello se excluyen procesos que presentan un comportamiento oscilante incluso sin regulación, así como procesos regulados sin compensación (integrador en el proceso regulado).
- **Tiempos de retardo no demasiado amplios**
  El campo de utilización puede especificarse a partir de la relación entre el tiempo de retardo t_u y el tiempo de compensación t_a. El tiempo de retardo puede incluir además un tiempo muerto.
  El primer ajuste o adaptación está orientado preferentemente para el rango $t_u < 1/10\ t_a$, que abarca la mayoría de lazos de regulación de temperatura.
  Para el rango de tiempos de retardo especialmente elevados $1/10\ t_a < t_u < 1/3\ t_a$ suele ser posible diseñar un regulador PI adecuado.
- **Comportamiento suficientemente lineal durante un rango de trabajo suficientemente grande**
  Esto significa que, tanto para la identificación como para el modo de regulación normal, los efectos no lineales dentro del rango de trabajo especificado pueden ignorarse. Sin embargo, es posible volver a identificar el proceso cambiando el punto de operación si el proceso de adaptación vuelve a ejecutarse en un entorno reducido del nuevo punto de operación, y si durante la adaptación no se pasa por la no linealidad.
  Si se conocen determinadas no linealidades estáticas (p. ej. características de válvulas), es aconsejable compensarlas previamente con una línea poligonal para linealizar el comportamiento del proceso.
- **Perturbaciones reducidas en procesos de temperatura**
  Las perturbaciones como la transmisión térmica a las zonas adyacentes, o el aumento o descenso térmico por el cambio de estado de grupos constructivos no pueden afectar en exceso a todo el lazo de regulación de temperatura. Puede ser necesaria una adaptación en el punto de operación. Por ejemplo, en la optimización de zonas de un extrusor puede ser necesario calentar simultáneamente todas las zonas.

- **Calidad de las señales de medición**
  Las señales de medición deben tener una calidad suficiente, es decir, que la relación entre la señal parásita y la señal útil sea lo suficientemente reducida.
- **Ganancias del proceso no demasiado grandes**
  Las ganancias del proceso no pueden ser demasiado elevadas. No es necesaria una normalización de los valores reales. Por lo tanto, la ganancia del proceso K se ve afectada por unidades físicas, p. ej.:
  K =ΔPV/ΔLMN, [K]= °C/ %
  El diseño definitivo del regulador se basa en un cálculo de la ganancia del proceso K, y en principio puede compensar cualquier valor de K. Pero en la fase de aprendizaje K todavía no se conoce, y en combinaciones extremas de ganancia e intensidades de escalón no es posible evitar sobreoscilaciones transitorias. Reduciendo el parámetro LHLM_TUN en el proceso de calentamiento o aumentando LLLM_TUN en el de refrigeración también se reduce el rebase transitorio.

Requisitos de procesos con servoválvulas de acción integral

En procesos con servoválvulas de acción integral existen otros requisitos adicionales a los anteriores:

El tiempo de funcionamiento del motor de la servoválvula debe ser menor al necesario para encontrar un punto de inversión tras un escalón de la variable manipulada (véase la figura Comportamiento del proceso). Cuando no es así, normalmente se trata de una regulación de caudal en la que sólo actúa la servoválvula como acción dominante del proceso. En ese caso no se recomienda el uso del Self-Tuner. El regulador PI paso a paso se puede regular manualmente siguiendo la siguiente regla general:

GAIN = 1, TI = tiempo de posicionamiento del motor

Consulte también

Funcionamiento de TUN_EC (Página 7133)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de salida de TUN_EC (Página 7140)

Parámetros de entrada/salida de TUN_EC (Página 7142)

Parámetro TUN_EC.PHASE (Página 7144)

Parámetro TUN_EC.STATUS_H (Página 7146)

Parámetro TUN_EC.STATUS_C (Página 7147)

Parámetro TUN_EC.STATUS_D (Página 7148)

Funcionamiento de TUN_EC

Para controlar u optimizar el regulador, TUN_EC dispone de las siguientes funciones:

- Modo manual
  El Self-Tuner fuerza el regulador al modo manual y le transfiere sus parámetros en manual.
- Conmutación manual/automático con especificación predictiva de la variable manipulada
  Tras una optimización de regulador correcta, al desactivar el modo manual el Self-Tuner puede conmutar el regulador al modo automático aplicando una variable manipulada, de modo que se produzca una regulación rápida a la consigna. Este tipo de conmutación es discontinua. También es posible ajustar una conmutación sin discontinuidad en el Self-Tuner.
- Optimización del regulador
  Para optimizar los parámetros del regulador, primero es necesario identificar el proceso. A partir de las características del proceso se diseña un regulador, y a partir de dicho diseño se calculan los nuevos parámetros de regulador, que se activan entonces para el regulador:
  - Primer ajuste
    El primer ajuste sólo puede producirse con un salto de consigna positivo. Generalmente se aplica en la primera puesta en marcha o en caso de un cambio severo del comportamiento del proceso.
  - Optimización fina
    Si cambia el punto de operación o el comportamiento del proceso, el regulador puede optimizarse finamente online durante un salto positivo de la consigna.
  - Optimización de refrigeración
    Para regulaciones que operan con dos actuadores de acción contrapuesta (p. ej. regulación de temperatura: actuador de calentamiento y de refrigeración), en un salto negativo de la variable manipulada el Self-Tuner determina la relación de las ganancias del proceso bajo la influencia del segundo actuador (para seguir en el ejemplo de la regulación de temperatura: bajo la influencia del actuador de refrigeración).

  Optimización de la respuesta de regulación
  El regulador está diseñado básicamente para una respuesta óptima ante perturbaciones. Los "rigurosos" parámetros resultantes provocarían rebases transitorios de entre el 10 % y el 20 % en la intensidad de los saltos de la consigna.
  Una primera solución se basa en la zona de regulación. Para optimizar el comportamiento de excitación con saltos elevados de la consigna, en la optimización el Self-Tuner calcula un ancho de zona de regulación. El regulador está activo dentro de la zona. Fuera de la zona, el Self-Tuner controla el regulador con la magnitud manipulada máxima o mínima del regulador.
  Otra solución se basa en la descomposición estructural. Las acciones P o D se incluyen en el lazo de realimentación. De ese modo, el cambio de la consigna sólo afecta a la acción I del regulador.
  Si no fuese posible una descomposición estructural para el regulador, o si la zona de regulación falla, para optimizar el comportamiento de regulación puede activarse la siguiente función:
- Conmutación entre estructuras:
  Desconexión transitoria de la acción I o especificación predictiva de la variable manipulada con grandes saltos positivos de la consigna. Esta función no es efectiva con zona de regulación activada.

- **Guardar los parámetros de regulador**
  Si los parámetros actuales del regulador se consideran apropiados, pueden guardarse antes de modificarse manualmente en parámetros propios previstos para tal fin en el DB de instancia del Self-Tuner. En una optimización del regulador, los parámetros guardados se sobrescriben con los valores válidos antes de la optimización.
- **Recargar los parámetros guardados del regulador**
  Esta función permite volver a activar los últimos parámetros guardados del regulador.
- **Modificar los parámetros del regulador**
  Para ajustar manualmente los parámetros del regulador, éstos se pueden especificar en el DB de instancia del Self-Tuner mediante parámetros especiales y transferirse al regulador con esta función.

Activación de las funciones

Función	Bit de inicio	Bit de habilitación	Condición de inicio: PHASE	Condición de inicio: Salto de consigna en SP
Modo manual	—	MAN_ON	0 a 7	—
Conmutación manual/ automático con especificación predictiva de la variable manipulada	—	PRED_ON	Transición 7 -> 4	—
Primer ajuste	ADAPT1ST	—	0, 4	pos.: > MIN_STEP
Primer ajuste u optimización fina	ADAPT_ON	—	4	pos.: > MIN_STEP
Optimización de refrigeración	COOLID_ON	—	4	—
Zona de regulación	—	CONZ_ON	4	
Conmutación entre estructuras	—	STRUC_ON	4	pos.: > MIN_STEP, neg.: > MIN_STEP × RATIO_FAC
Guardar parámetros del regulador	SAVE_PAR	—	2 a 7	—
Recargar los parámetros del regulador	UNDO_PAR	—	4	—
Cambiar los parámetros del regulador	LOAD_PAR	—	4	—

Nota

Los bits de inicio se desactivan una vez que la función se ha iniciado correctamente y los de habilitación permanecen activados.

Consulte también

Descripción de TUN_EC (Página 7129)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de salida de TUN_EC (Página 7140)

Parámetros de entrada/salida de TUN_EC (Página 7142)

Parámetro TUN_EC.PHASE (Página 7144)

Parámetro TUN_EC.STATUS_H (Página 7146)

Parámetro TUN_EC.STATUS_C (Página 7147)

Parámetro TUN_EC.STATUS_D (Página 7148)

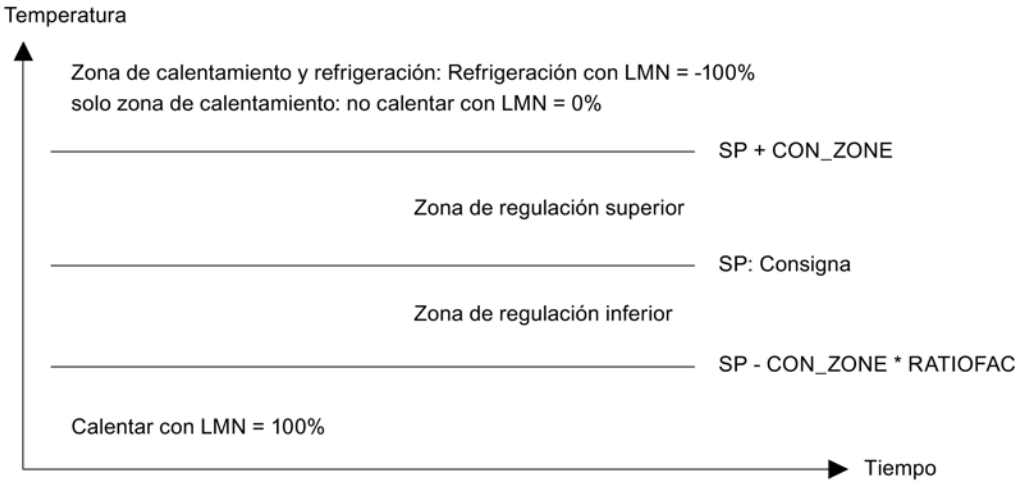
Zona de regulación y conmutación entre estructuras

Zona de regulación

El Self-Tuner trabaja con una zona de regulación. Esto significa que el regulador se controla según el siguiente algoritmo:

- Si el error de regulación es mayor que CON_ZONE, se efectúa el calentamiento a máxima potencia.
- Si el error de regulación es menor que (-CON_ZONE) × RATIOFAC, se efectúa la refrigeración a máxima potencia.

En la optimización inicial o fina, el Self-Tuner calcula una zona de regulación CON_ZONE que se activará coincidiendo con el tipo de proceso adecuado: CONZ_ON = TRUE. En el modo de regulación es posible modificar la zona de regulación o desactivarla completamente (con CONZ_ON = FALSE).



Con la zona de regulación activada (CONZ_ON = TRUE), la conmutación entre estructuras STRUC_ON carece de efecto.

Nota

Cuando se diseña el regulador especialmente para procesos con grandes tiempos de retardo, no se determina ninguna zona de regulación cuando después de la adaptación se cumple STATUS_D = 3.

Antes de activar a mano la zona de regulación, asegúrese de no ajustar un ancho de zona de regulación demasiado pequeño. Si el ancho de la zona de regulación ajustado fuera insuficiente, se producirán oscilaciones en las curvas de la variable manipulada y el recorrido del valor real.

Descomposición estructural

La descomposición estructural es una función del regulador. En el modo de regulación con descomposición estructural, las acciones P y D se ubican en el lazo de realimentación, es decir, cualquier modificación de la consigna afectará únicamente a la acción I del regulador. En caso de escalones de la consigna, la variable manipulada no registra un escalón, sino que discurre en forma de rampa. Esto no altera la respuesta a perturbaciones.

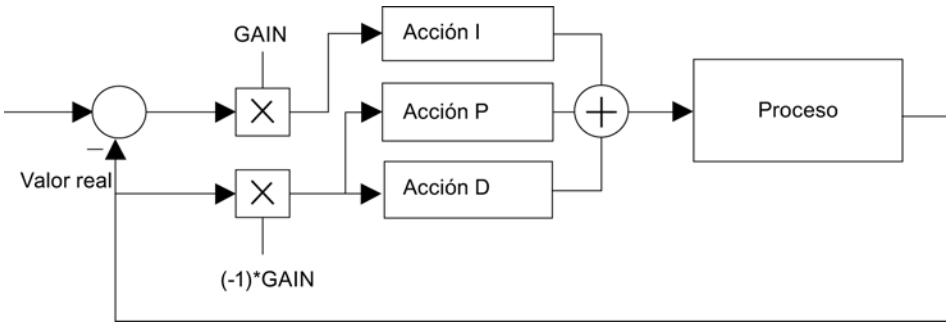


Figura 9-7 Modo de regulación con descomposición estructural

En el modo de regulación sin descomposición estructural, las acciones P y D se ubican, al igual que la acción I, en el error de regulación. En este modo de operación puede seleccionarse una conmutación entre estructuras (STRUC_ON = TRUE).

El modo de regulación con descomposición estructural es posible en los siguientes paquetes de regulación:

- FM 355/455
- Standard PID Control
- Modular PID Control

Nota

En el regulador simple PID Control de STEP 7-base y en la CPU 314 IFM solo es posible el modo de regulación sin descomposición estructural. La conmutación entre estructuras solo debe activarse si no existen otras alternativas.

Conmutación entre estructuras con acciones P y D en el error de regulación

En la configuración predeterminada, la conmutación entre estructuras está desactivada. Solo debe activarse si no existen otras alternativas, como p. ej. la zona de regulación o la descomposición estructural.

Con la zona de regulación activada (CONZ_ON = TRUE), la conmutación entre estructuras STRUC_ON carece de efecto.

Para activar la conmutación entre estructuras, ajuste STRUC_ON = TRUE. En función de la ganancia total del regulador (producto de la ganancia del proceso por la ganancia del regulador), el bloque elige automáticamente entre dos acciones de control:

- Acción I desactivada temporalmente, regulación P(D):
  Con un escalón de consigna positivo ≥ MIN_STEP la acción I del regulador se desactiva temporalmente y la ganancia se aumenta ligeramente, de modo que se utiliza un regulador P(D) puro. Cerca de la consigna se vuelve a activar la acción I y se vuelve a reducir la ganancia (PHASE = 5).
- Especificación predictiva temporal de la variable manipulada:
  Los procesos con un tiempo de retardo elevado no son adecuados para una regulación P(D). Por ello, tras un escalón positivo de consigna ≥ MIN_STEP se emite de forma constante la variable manipulada que se requiere estacionaria para la nueva consigna. Cerca de la consigna se vuelve a cambiar sin discontinuidad al modo de regulador PI o PID. Con ello se obtiene un comportamiento lento pero sin sobreoscilaciones transitorias (PHASE = 6).

Nota

Si, con escalones de consigna positivos (p. ej. en fases de calentamiento), la conmutación entre estructuras no da buenos resultados (comportamiento transitorio muy lenta), puede desactivarla con STRUC_ON = FALSE, teniendo en cuenta que pueden producirse pequeños rebases transitorios.

Modos de operación de la estructura del regulador	CONZ_ON	PFDB_SEL, DFDB_SEL [1)]	STRUC_ON
Activar zona de regulación	TRUE	cualquiera	cualquiera
Activar la descomposición estructural	cualquiera	TRUE	FALSE
Activar la conmutación entre estructuras	FALSE	FALSE	TRUE

[1)] Parámetros de regulador PFDB_SEL, DFDB_SEL: Con CONT_C, estos parámetros no están disponibles. Esto corresponde al ajuste PFDB_SEL = DFDB_SEL = FALSE (sin descomposición estructural).

Nota

Indicaciones para la selección de los modos de operación:

- El Self-Tuner calcula una zona de regulación y la activa automáticamente en caso de que la relación de las dos constantes de tiempo del proceso sea f = TM_LAG1 / TM_LAG2 > 10.
- En caso de escalones dentro de la zona de regulación, se aplica la descomposición estructural que se encuentra activada por defecto (a condición de que el regulador lo permita).
- Si desea corregir lo más rápidamente posible los pequeños escalones de consigna dentro de la zona de regulación y no tiene inconveniente en que se produzcan pequeñas sobreoscilaciones transitorias, desactive la descomposición estructural.
  Ejemplo de números: Zona de regulación 20 °C; se producen sobreoscilaciones transitorias de aprox. 10% a 20% de la altura del escalón (< 20 °C), es decir, por debajo de entre 2 y 4 °C.
- Aunque el proceso no admita zonas de regulación o estas se hayan desactivado, la descomposición estructural impide que se produzcan sobreoscilaciones transitorias para cualquier escalón de consigna. Sin embargo, los tiempos de subida son algo más largos que en caso de existir zona de regulación.
- Si el proceso no admite zonas de regulación y el regulador no permite la descomposición estructural, active la conmutación entre estructuras. La conmutación entre estructuras no debe estar activada al mismo tiempo que la descomposición estructural.

Parámetros de entrada de TUN_EC

Tabla 9-198

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP	0.0	REAL	0.0	En la entrada SP se introduce la consigna. Ésta se interconecta con el regulador a través de SP_OUT. Recuerde que la consigna debe estar comprendida dentro de los límites de consigna del regulador.
PV	4.0	REAL	0.0	PV se interconecta con el valor real del regulador.
LMN	8.0	REAL	0.0	LMN se conecta con la variable manipulada del regulador. Se admiten valores entre -100 y 100 %.
MAN	12.0	REAL	0.0	En la entrada MAN se introduce el valor manual. Si MAN_ON = TRUE, el valor manual MAN se transmite a la variable manipulada del regulador. Con MAN_ON = TRUE y MAN = 0.0 la calefacción/refrigeración están desconectadas. Si MAN_ON = FALSE el regulador especifica la variable manipulada. Se admiten valores entre -100 y 100 %.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MIN_STEP	16.0	REAL	10.0	En la entrada MIN_STEP se puede definir la altura mínima del escalón para la consigna a partir de la cual se puede realizar una primera optimización o adaptación. Con la conmutación entre estructuras activada (STRUC_ON = TRUE), para que se active es necesario que el escalón de la consigna sea mayor que MIN_STEP. Recomendación: MIN_STEP ≥ 10 % del rango de consigna/valor real requerido
LHLM_TUN	20.0	REAL	80.0	En la entrada LHLM_TUN se introduce el salto de la variable manipulada para el primer ajuste. En posteriores optimizaciones finas se calcula automáticamente el escalón de la variable manipulada. Se admiten valores entre 0 y 100 %.
LLLM_TUN	24.0	REAL	-20.0	En la entrada LLM_TUN se introduce el salto de la variable manipulada para la optimización de refrigeración. Se admiten valores entre -100 y 0 %.
NORM_FAC	28.0	REAL	1.0	En reguladores con ganancia normalizada (Standard PID Control) se aplica un factor de normalización. Éste se calcula a partir de los límites de rango del valor real del regulador y se aplica en la entrada NORM_FAC.
MAN_ON	32.0	BOOL	TRUE	Si MAN_ON = TRUE, la variable manipulada del regulador se especifica a través de la entrada MAN. Si MAN_ON = FALSE el regulador especifica la variable manipulada.
PID_ON	32.1	BOOL	TRUE	En la entrada PID_ON se puede determinar si el regulador optimizado debe trabajar como regulador PI o PID. Regulador PID: PID_ON = TRUE Regulador PI: PID_ON = FALSE Puede ocurrir que en algunos tipos de proceso se diseñe un regulador PI a pesar de PID_ON = TRUE.
STRUC_ON	32.2	BOOL	FALSE	Si STRUC_ON = TRUE, en caso de escalones de la consigna, una conmutación de la estructura (de regulación PI(D) a P(D)) garantiza que se eviten ampliamente sobreoscilaciones transitorias. La entrada STRUC_ON no debe ajustarse = TRUE si en el regulador las acciones P y D están incluidas en el lazo de realimentación.
WRITE_DIS	32.3	BOOL	FALSE	En combinación con el FM Regulación, el Self-Tuner debe saber si el bloque driver está dispuesto a enviar datos al FM. Ése es el caso si LOAD_OP = LOAD_PAR = FALSE. El resultado de la operación lógica O entre LOAD_OP y LOAD_PAR se aplica en WRITE_DIS.
PRED_ON	32.4	BOOL	TRUE	Al cambiar al modo automático se puede elegir si se desea conmutar sin discontinuidad desde el último valor manual o con discontinuidad con una variable manipulada calculada por el Self-Tuner. Si PRED_ON = TRUE (ajuste predeterminado), se conmuta con discontinuidad. La conmutación con discontinuidad no es efectiva hasta después de la optimización, y hace posible, especialmente en el caso de la regulación PI, un transitorio más rápido hasta la consigna deseada.
CYCLE	34.0	TIME	100 ms	En la entrada CYCLE hay que parametrizar el tiempo de muestreo. CYCLE debe coincidir con el tiempo de muestreo del regulador ajustado y con la diferencia de tiempo entre dos llamadas (tiempo de ciclo del OB de alarma cíclica considerando los factores de ciclo). CYCLE > 1 ms

Consulte también

Descripción de TUN_EC (Página 7129)
Funcionamiento de TUN_EC (Página 7133)
Parámetros de salida de TUN_EC (Página 7140)
Parámetros de entrada/salida de TUN_EC (Página 7142)
Parámetro TUN_EC.PHASE (Página 7144)
Parámetro TUN_EC.STATUS_H (Página 7146)
Parámetro TUN_EC.STATUS_C (Página 7147)
Parámetro TUN_EC.STATUS_D (Página 7148)

Parámetros de salida de TUN_EC

Tabla 9-199

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MAN_OUT	38.0	REAL	0.0	MAN_OUT se interconecta con el valor de entrada manual del regulador. Si MAN_ON = TRUE, el Self-Tuner emite en la salida MAN_OUT el valor predeterminado en la entrada MAN.
SP_OUT	42.0	REAL	0.0	SP_OUT se conecta con la entrada de la consigna del regulador. El Self-Tuner escribe la consigna especificada en la entrada SP en la salida SP_OUT. En el modo manual MAN_ON = TRUE la consigna se corrige según el valor real: SP_OUT = PV.
GAIN	46.0	REAL	1.5	GAIN se conecta con la ganancia proporcional del regulador (ganancia del regulador).
TI	50.0	TIME	1 h	TI se conecta con el tiempo integral (tiempo de acción integral) del regulador.
TD	54.0	TIME	0 ms	TD se conecta con el tiempo de acción derivada (tiempo de anticipación) del regulador.
TM_LAG	58.0	TIME	1 s	TM_LAG se conecta con el tiempo de retardo del diferenciador del regulador.
RATIOFAC	62.0	REAL	1.0	RATIOFAC se conecta con el factor de proporción calentar/refrigerar del regulador o con la instrucción PULSEGEN. El parámetro sólo se utiliza para refrigeración.
PHASE	66.0	INT	0	Parámetro TUN_EC.PHASE (Página 7144) muestra el progreso actual de la optimización del regulador. En la optimización de Standard PID Control, la salida PHASE se conecta con la entrada PHASE de la instrucción PID_CP.
STATUS_H	68.0	INT	0	Parámetro TUN_EC.STATUS_H (Página 7146) muestra resultados de la optimización a través de la búsqueda del punto de inversión en el proceso de calentamiento.
STATUS_D	70.0	INT	0	Parámetro TUN_EC.STATUS_D (Página 7148) muestra resultados de la optimización a través del diseño del regulador en el proceso de calentamiento.
STATUS_C	72.0	INT	0	Parámetro TUN_EC.STATUS_C (Página 7147) muestra resultados de la optimización a través de la búsqueda del punto de inversión en el proceso de refrigeración.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QMAN_ON	74.0	BOOL	FALSE	QMAN_ON se conecta con la conmutación manual/automático del regulador.
QI_SEL	74.1	BOOL	TRUE	QI_SEL se conecta con el activador/desactivador de la acción I del regulador I_SEL. En el FM355/455 debe conmutarse en función de QI_SEL TI a cero: Si QI_SEL = FALSE, entonces TI = 0.
QD_SEL	74.2	BOOL	FALSE	QD_SEL se conecta con el activador/desactivador de la acción D del regulador D_SEL. En el FM355/455 debe conmutarse en función de QD_SEL TD a cero: Si QD_SEL = FALSE, entonces TD = 0.
QWRITE	74.3	BOOL	FALSE	QWRITE se conecta con las entradas LOAD_PAR y LOAD_OP de la instrucción PID_FM del FM355/455. Con ello se cargan los registros del canal regulador en el FM.
PROCESS	76.0	STRUCT		La estructura PROCESS contiene los parámetros del proceso.
GAIN	+0.0	REAL	999.0	En la salida PROCESS.GAIN se muestra la ganancia del proceso.
TM_LAG1	+4.0	REAL	0.0	En la salida PROCESS.TM_LAG1 se muestra la constante de tiempo del proceso lenta.
TM_LAG2	+8.0	REAL	0.0	En la salida PROCESS.TM_LAG2 se muestra la constante de tiempo del proceso rápida.
PV00	+12.0	REAL	0.0	En la salida PROCESS.PV00 se muestra el valor de equilibrio del proceso. El valor de equilibrio es el valor real estacionario con la variable manipulada = cero. Equivale a la temperatura ambiente con la instalación fría.
KIG	+16.0	REAL	0.0	En la salida PROCESS.KIG se muestra la relación de rampa máxima del valor real en el calentamiento.
KIG_C	+20.0	REAL	0.0	En la salida PROCESS.KIG_C se muestra la relación de reducción máxima del valor real en la refrigeración.
PI_CON	100.0	STRUCT		La estructura PI_CON contiene los parámetros del regulador PI.
GAIN	+0.0	REAL	1.5	PI_CON.GAIN muestra la ganancia proporcional del regulador PI. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
TI	+4.0	REAL	3600.0	PI_CON.TI muestra el tiempo integral del regulador PI. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
PID_CON	108.0	STRUCT		La estructura PID_CON contiene los parámetros del regulador PID.
GAIN	+0.0	REAL	1.5	PID_CON.GAIN muestra la ganancia proporcional del regulador PID. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
TI	+4.0	REAL	3600.0	PID_CON.TI muestra el tiempo integral del regulador PID. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
TD	+8.0	REAL	0.0	PID_CON.TD muestra el tiempo de acción derivada del regulador PID. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
PI_CON_OLD	120.0	STRUCT		La estructura PI_CON_OLD contiene los parámetros antiguos del regulador PI.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
GAIN	+0.0	REAL	1.5	En la salida PI_CON_OLD.GAIN se muestra la ganancia proporcional antigua del regulador PI.
TI	+4.0	REAL	3600.0	En la salida PI_CON_OLD.TI se muestra el tiempo integral antiguo del regulador PI.
PID_CON_OLD	128.0	STRUCT		La estructura PID_CON_OLD contiene los parámetros antiguos del regulador PID.
GAIN	+0.0	REAL	0.0	En la salida PID_CON_OLD.GAIN se muestra la ganancia proporcional antigua del regulador PID.
TI	+4.0	REAL	0.0	En la salida PID_CON_OLD.TI se muestra el tiempo integral antiguo del regulador PID.
TD	+8.0	REAL	0.0	En la salida PID_CON_OLD.TD se muestra el tiempo de acción derivada antiguo del regulador PID.
TU	140.0	REAL	0.0	En la salida TU se muestra el tiempo de retardo determinado.
TA	144.0	REAL	0.0	En la salida TA se muestra el tiempo de compensación determinado.

Consulte también

Descripción de TUN_EC (Página 7129)

Funcionamiento de TUN_EC (Página 7133)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de entrada/salida de TUN_EC (Página 7142)

Parámetros de entrada/salida de TUN_EC

Tabla 9-200

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
ADAPT1ST	148.0	BOOL	FALSE	En la primera optimización del proceso o si no se desea utilizar las conclusiones de las optimizaciones anteriores, como la intensidad del salto de variable manipulada o las constantes del tiempo del filtro, hay que realizar un primer ajuste. Para ello hay que activar ADAPT1ST = TRUE, entonces el Self-Tuner realiza una marcha de inicialización y ajusta los parámetros predeterminados descritos en la marcha de inicialización. El Self-Tuner vuelve a desactivar ADAPT1ST a FALSE. No es conveniente poner ADAPT1ST en TRUE en un rearranque completo (en caliente) de la CPU en el OB 100.
ADAPT_ON	148.1	BOOL	FALSE	Para lanzar una reoptimización, active ADAPT_ON = TRUE. Con un salto de consigna posterior superior a MIN_STEP, se pone en marcha la identificación (PHASE = 2). El Self-Tuner vuelve a poner ADAPT_ON en FALSE.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
STEADY	148.2	BOOL	FALSE	Una vez que el valor real ha alcanzado un estado estacionario, la optimización se puede finalizar anticipadamente con STEADY, aunque el Self-Tuner todavía se encuentre en PHASE 3. También es posible volver a realizar el diseño del regulador en estado estacionario con STEADY, una vez que el Self-Tuner haya pasado a PHASE 4.
COOLID_ON	148.3	BOOL	FALSE	Para lanzar una optimización de refrigeración, ajuste COOLID_ON = TRUE. La variable manipulada se pone a LLLM_TUN y COOLID_ON se resetea a FALSE. Si hay varias zonas de calentamiento, primero deben haberse completado todos los primeros ajustes o adaptaciones antes de iniciar la optimización de refrigeración.
UNDO_PAR	148.4	BOOL	FALSE	Los parámetros de regulador antiguos se transfieren sobre los actuales y se activan de inmediato en función de PID_ON. PI_CON = PI_CON_OLD PID_CON = PID_CON_OLD La función para deshacer cambios sólo es posible en el modo automático (PHASE 4).
SAVE_PAR	148.5	BOOL	FALSE	Los parámetros de regulador actuales se transfieren sobre los antiguos. PI_CON_OLD = PI_CON PID_CON_OLD = PID_CON
LOAD_PAR	148.6	BOOL	FALSE	Los parámetros modificados PI_CON y PID_CON se transmiten al regulador y se activan.
CONZ_ON	148.7	BOOL	FALSE	Si CONZ_ON = TRUE, el regulador trabaja con zona de regulación. En una optimización de regulador, el Self-Tuner decide si debe regularse con o sin zona de regulación y determina una zona de regulación óptima. Tras la optimización en PHASE 4 la zona de regulación se puede desactivar con CONZ_ON = FALSE o reparametrizarse.
CON_ZONE	150.0	REAL	0.0	CON_ZONE indica la amplitud de la zona de regulación. En una optimización de regulador, el Self-Tuner decide si debe regularse con o sin zona de regulación y determina una zona de regulación óptima. Tras la optimización en PHASE 4 la zona de regulación se puede desactivar con CONZ_ON = FALSE o reparametrizarse.

Consulte también

Descripción de TUN_EC (Página 7129)

Funcionamiento de TUN_EC (Página 7133)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de salida de TUN_EC (Página 7140)

Parámetro TUN_EC.PHASE (Página 7144)

Parámetro TUN_EC.STATUS_H (Página 7146)

Parámetro TUN_EC.STATUS_C (Página 7147)

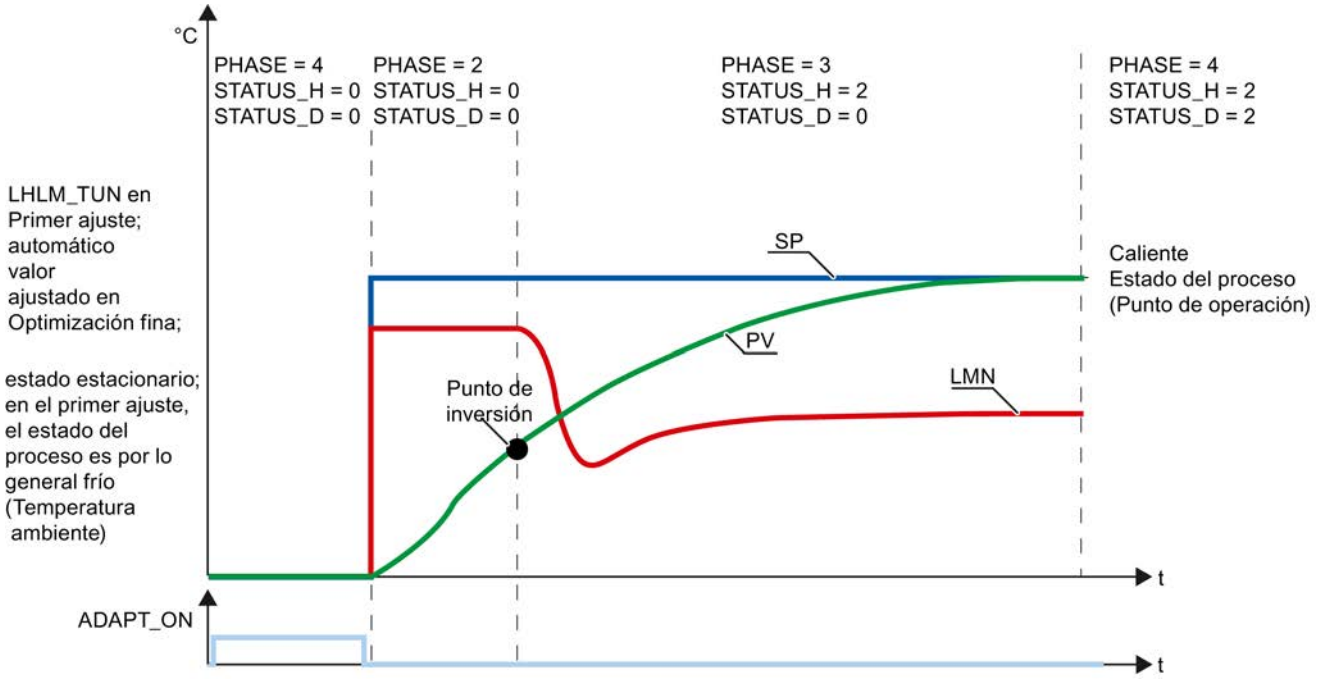
Parámetro TUN_EC.STATUS_D (Página 7148)

Parámetro TUN_EC.PHASE

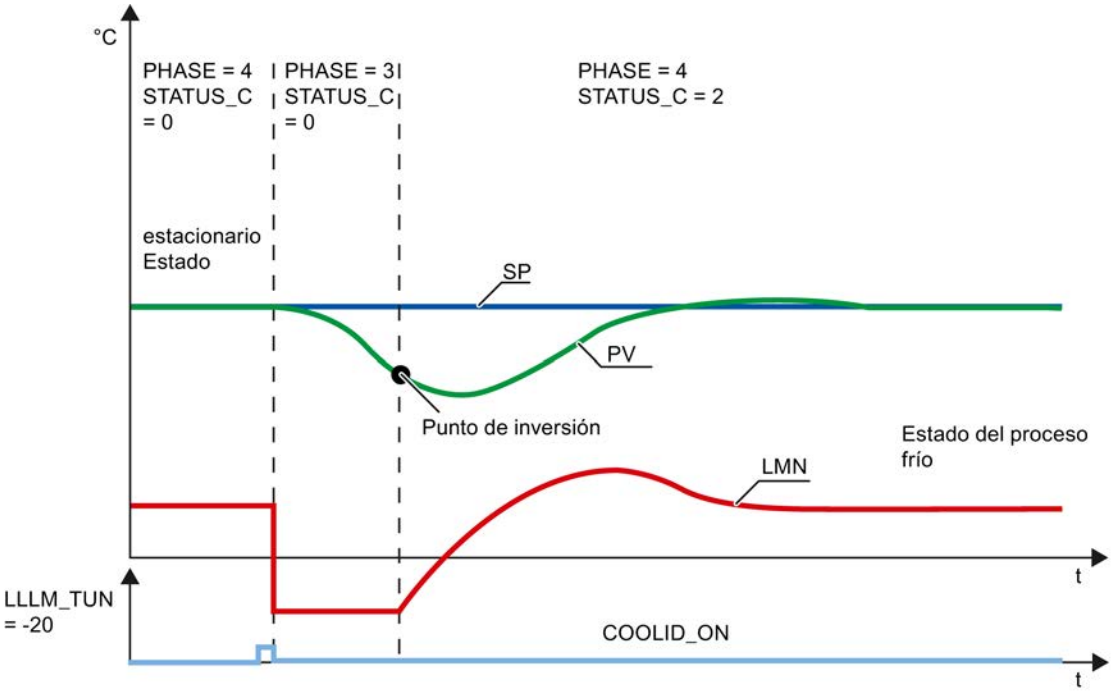
PHASE	Modo de operación	Descripción
0	Modo automático: Modo de regulación con parámetros predeterminados	Al crear un DB de instancia para el Self-Tuner, el parámetro PHASE tiene preasignado el valor cero. Dado que MAN_ON tiene preasignado el estado TRUE, el Self-Tuner cambiará a PHASE 7 (modo manual) tras la primera ejecución. Desde el modo manual se puede iniciar directamente el primer ajuste (ADAPT_ON = TRUE) o cambiar al modo automático. En la primera conmutación al modo automático MAN_ON = FALSE, PI_CON adopta los parámetros. Los valores predeterminados para GAIN y TI son: GAIN = 1.5, TI = 1h.
2	Optimización del regulador: Búsqueda del punto de inversión con variable manipulada constante	En cuanto se especifica un salto de consigna ≥ MIN_STEP en sentido positivo hacia el punto de operación del estado caliente del proceso, en el ajuste inicial a MAN_OUT se le preasigna LHLM_TUN y en la optimización fina a MAN_OUT se le preasigna el valor calculado por el Self-Tuner. Además se activa QMAN_ON = TRUE. Ambos valores se transmiten al regulador PID. El regulador PID se controla entonces en modo manual. MIN_STEP debería ser superior al 10 % del rango de trabajo de consigna y real.
3	Optimización del regulador: Regulación con un regulador PI aproximativo	Si se detecta el punto de inversión de la respuesta al salto (STATUS_H = 2) o si el valor real ha alcanzado el 70 % de la intensidad del salto de la consigna (STATUS_H = 3), se diseña un regulador PI de ajuste cuidado. La ganancia del regulador GAIN se limita al rango entre 0.2 y 20.0. El regulador trabaja a partir de ese momento como regulador PI e intenta regular el proceso a un estado estacionario. El Self-Tuner guarda los nuevos parámetros de regulador PI en GAIN, TI y PI_CON. Los valores anteriores del regulador PI se guardan en PI_CON_OLD. Si el tiempo necesario para alcanzar el estado estacionario es muy prolongado (respuesta transitoria muy lenta en lazos de regulación de temperatura), ya puede iniciarse el diseño del regulador con los datos actuales al aproximarse al estado estacionario activando STEADY = TRUE. También es posible volver a iniciar el diseño del regulador con los datos actuales posteriormente en la fase 3 ó 4 activando STEADY = TRUE. Con frecuencia, así se mejora un poco más el diseño del regulador. El hecho de que se produzca un rebase transitorio o de que no se encuentre un punto de inversión, en un primer ajuste puede deberse a que se ha seleccionado una intensidad de salto de la variable manipulada excesiva LHLM_TUN, y no tiene que derivar necesariamente en un mal ajuste del regulador. En el siguiente primer ajuste LHLM_TUN debe seleccionarse aprox. un 20 % más pequeño. Si el bloque ha detectado el estado estacionario o si el tiempo 30 × TI (TI: tiempo de acción integral del regulador PI ajustado en PHASE = 3) ha transcurrido desde el salto de la variable manipulada, se realiza un diseño mejorado del regulador, y se cambia a PHASE = 4. El Self-Tuner guarda los nuevos parámetros de regulador PI/PID determinados en GAIN, TI, TD y PI_CON o PID_CON. Los valores anteriores del regulador PID se guardan en PID_CON_OLD. A ser posible se diseñan un regulador PID y un regulador PI. Si PID_ON = TRUE, se activa el regulador PID, de lo contrario el regulador PI. En PHASE 4 se puede cambiar en todo momento entre los dos reguladores sin discontinuidad. La ganancia del regulador GAIN se limita de tal manera que la ganancia del lazo abierto (producto resultante de la ganancia del regulador por la del proceso) sea inferior a 80.0.
4	Modo automático: Funcionamiento normal del regulador con/sin zona de regulación	En esta fase el regulador regula con sus parámetros optimizados. Desde este estado se puede iniciar la optimización de refrigeración, una optimización fina o un primer ajuste.

PHASE	Modo de operación	Descripción
5	Modo automático: Conmutación entre estructuras con salto de consigna: Acción I desactivada, regulación P(D)	Con un salto de consigna positivo ≥ MIN_STEP la acción I del regulador se desactiva temporalmente y la ganancia se aumenta ligeramente, de modo que se utiliza un regulador P(D) puro. Cerca de la consigna se vuelve a activar la acción I y se vuelve a reducir la ganancia
6	Modo automático: Conmutación entre estructuras con salto de consigna: Especificación predictiva de la variable manipulada	Los procesos con un tiempo de retardo elevado no son adecuados para una regulación P(D). Por ello, tras un salto positivo de la consigna ≥ MIN_STEP, la magnitud manipulada necesaria estacionaria para la nueva consigna se emite de forma constante. Cerca de la consigna se vuelve a cambiar sin discontinuidad al modo de regulador PI o PID. Con ello se obtiene un comportamiento lento pero sin rebases transitorios.
7	Modo manual	Si la entrada MAN_ON se pone a TRUE, la salida QMAN_ON se pone a TRUE y la salida MAN_OUT a MAN. El regulador PID cambia así al modo manual. El modo manual tiene prioridad frente a todos los modos restantes. Cualquier primer ajuste, optimización fina u optimización de refrigeración se interrumpen. Con la desactivación del modo manual (MAN_ON = FALSE), el regulador pasa al modo automático (PHASE = 4 en el proceso del Self-Tuner) y continúa regulando con los parámetros disponibles.

Fases de la optimización de calefacción en el primer ajuste



Fases de una optimización de refrigeración



Consulte también

Descripción de TUN_EC (Página 7129)

Funcionamiento de TUN_EC (Página 7133)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de entrada/salida de TUN_EC (Página 7142)

Parámetro TUN_EC.STATUS_H (Página 7146)

Parámetro TUN_EC.STATUS_C (Página 7147)

Parámetro TUN_EC.STATUS_D (Página 7148)

Parámetro TUN_EC.STATUS_H

STATUS_H	Descripción	Solución
0	Inicialización	
1	Punto de inversión demasiado bajo En la mayoría de los casos, la optimización suele conducir a parámetros buenos de regulador	Para acercarse más aún al caso ideal, aumente LHLM_TUN o reduzca la intensidad del salto de la consigna.
2	Caso ideal: Punto de inversión encontrado	

STATUS_H	Descripción	Solución
3	Punto de inversión demasiado alto Se incluye un factor de seguridad en el cálculo; la optimización suele arrojar en la mayoría de los casos parámetros válidos, pero cautos.	Para acercarse más aún al caso ideal, reduzca LHLM_TUN o aumente la intensidad del salto de la consigna.
11, 12	El tiempo de retardo no se ha podido registrar correctamente. Se sigue trabajando con un valor estimado, que puede provocar parámetros no óptimos del regulador. Si el proceso presenta un **comportamiento VZ1** puro (sin tiempo de retardo), con frecuencia el Self-Tuner consigue parámetros de regulador buenos. Dado que para el cálculo posterior se estima un tiempo de retardo generoso, es posible que STATUS_D no lleve a 1 (regulador para comportamiento cercano a VZ1) y que la zona de regulación se desactive, a pesar de que una zona de regulación puede ser conveniente en procesos VZ1.	Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real. Si no está satisfecho con los parámetros del regulador, como mínimo puede utilizar los valores de PROCESS.GAIN y PROCESS.TM_LAG1 para un modelo VZ1 y un diseño de regulador propio según el libro guía.
13	El tiempo de retardo no se ha podido registrar correctamente. Se sigue trabajando con un valor estimado, que puede provocar parámetros no óptimos del regulador.	Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

Consulte también

Descripción de TUN_EC (Página 7129)

Funcionamiento de TUN_EC (Página 7133)

Parámetros de entrada de TUN_EC (Página 7138)

Parámetros de entrada/salida de TUN_EC (Página 7142)

Parámetro TUN_EC.PHASE (Página 7144)

Parámetro TUN_EC.STATUS_C (Página 7147)

Parámetro TUN_EC.STATUS_D (Página 7148)

Parámetro TUN_EC.STATUS_C

STATUS_C	Descripción	Solución
0	Inicialización	
1	Punto de inversión no encontrado En la mayoría de los casos, la optimización suele conducir a parámetros buenos de regulador.	Para aproximarse más aún al caso ideal, aumente LLLM_TUN (p. ej. de -20.0 a -10.0)
2	Caso ideal: Punto de inversión de refrigeración encontrado	
11, 12, 13	Mala estimación Al factor de proporción se le asigna el valor de preasignación: RATIOFAC = 1.0	Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

Consulte también

Descripción de TUN_EC (Página 7129)
Funcionamiento de TUN_EC (Página 7133)
Parámetros de entrada de TUN_EC (Página 7138)
Parámetros de entrada/salida de TUN_EC (Página 7142)
Parámetro TUN_EC.PHASE (Página 7144)
Parámetro TUN_EC.STATUS_H (Página 7146)
Parámetro TUN_EC.STATUS_D (Página 7148)

Parámetro TUN_EC.STATUS_D

STATUS_D	Descripción	Solución
0	Inicialización	
1	El proceso tiene un comportamiento muy próximo a PT1;	Quizás se pueda renunciar a la acción D del regulador.
2	Reguladores PI y PID posibles, comportamiento VZ2	
3	Diseño de regulador PID separado para tiempos de retardo muy elevados, tiempos muertos u orden de proceso elevada En este caso, la zona de regulación se desactiva.	
4	Mala estimación Sin diseño de regulador en reoptimización. En el primer ajuste se continúa trabajando con el regulador PI aproximado de PHASE 3	Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

Consulte también

Descripción de TUN_EC (Página 7129)
Funcionamiento de TUN_EC (Página 7133)
Parámetros de entrada de TUN_EC (Página 7138)
Parámetros de entrada/salida de TUN_EC (Página 7142)
Parámetro TUN_EC.PHASE (Página 7144)
Parámetro TUN_EC.STATUS_H (Página 7146)
Parámetro TUN_EC.STATUS_C (Página 7147)

TUN_ES

Descripción de TUN_ES

La instrucción TUN_ES permite ajustar un regulador PID paso a paso. Según el regulador, TUN_ES se activa con una de las siguientes instrucciones:

- FM 355 S o FM 455 S
  - PID_FM
- Funciones básicas PID
  - CONT_S
  - TCONT_S
- Standard PID Control (paquete opcional PID Professional)
  - PID_ES
- Modular PID Control (paquete opcional PID Professional)
  - PID
  - LMNGEN_S

Llamada

TUN_ES se llama en una base de tiempo constante del tiempo de ciclo del OB invocante (preferentemente en un OB de alarma cíclica).

Con ADAPT1ST = TRUE se ejecuta la rutina de inicialización de la instrucción TUN_ES. Se llevan a cabo los siguientes ajustes predeterminados:

- Se conmuta al regulador PI. Para ello se transfieren al regulador los parámetros de PI_CON. Si los parámetros PI_CON.GAIN y PI_CON.TI tienen ambos el valor 0.0, se les asignarán los valores predeterminados PI_CON.GAIN = 1.5 y PI_CON.TI = 3600.0 s.
- PROCESS.GAIN = 999.0
- STATUS_H = 0; STATUS_D = 0

Nota

A menos que desee hacerlo expresamente, en caso de rearranque completo (en caliente) de la CPU **no** llame TUN_ES como es habitual con otras instrucciones, pues de lo contrario se perderán todos los parámetros de proceso.

Si se efectúa una llamada de la instrucción TUN_ES como DB multiinstancia, no se crea ningún objeto tecnológico. No existe interfaz de puesta en servicio. TUN_EC debe ponerse en servicio mediante una tabla de observación.

Tiempo de muestreo CYCLE

El tiempo de muestreo se indica en el parámetro CYCLE. El tiempo de muestreo no debe ser superior al 10% del tiempo de integración calculado del regulador. Además, para obtener una

resolución de la realimentación de posición de al menos 1%, el tiempo de muestreo no debe ser superior al 1% del tiempo de posicionamiento del motor. El tiempo de muestreo puede ajustarse en el parámetro CYCLE de la instrucción TUN_ES y del regulador. Debe coincidir con la diferencia de tiempo entre dos llamadas (tiempo de ciclo del OB de alarma cíclica considerando los factores de ciclo).

Si se usan los módulos de regulación FM355/455, el tiempo de muestreo de TUN_ES debe ser aproximadamente igual al tiempo de ciclo resultante de los reguladores en el FM (Ajuste de parámetros > Parámetros del módulo).

Comportamiento en caso de error

El bloque no evalúa la palabra de notificación de error RET_VAL

Consulte también

Funcionamiento TUN_ES (Página 7152)

Parámetros de entrada de TUN_ES (Página 7157)

Parámetros de salida de TUN_ES (Página 7159)

Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetro TUN_ES.PHASE (Página 7162)

Parámetro TUN_ES.STATUS_H (Página 7165)

Parámetro TUN_ES.STATUS_D (Página 7166)

Campo de aplicación de TUN_ES

El Self-Tuner resulta especialmente conveniente para

- Regulaciones de temperatura, pero también para
- Regulaciones de nivel de llenado y
- Regulaciones de caudal

Dentro de las regulaciones de caudal se distingue si debe regularse la servoválvula o si tras ésta se inicia un proceso sujeto a retardo. Si se trata de una regulación puramente de la servoválvula, es Self-Tuner no puede utilizarse.

Requisitos del proceso

El proceso debe cumplir las siguientes condiciones:

- **Comportamiento transitorio asintótico, estable y sujeto a retardo**
  Después de un escalón de la magnitud manipulada, el valor real debe pasar a un estado estacionario. Con ello se excluyen procesos que presentan un comportamiento oscilante incluso sin regulación, así como procesos regulados sin compensación (integrador en el proceso regulado).
- **Tiempos de retardo no demasiado amplios**
  El campo de utilización puede especificarse a partir de la relación entre el tiempo de retardo t_u y el tiempo de compensación t_a. El tiempo de retardo puede incluir además un tiempo muerto.
  El primer ajuste o adaptación está orientado preferentemente para el rango $t_u < 1/10\ t_a$, que abarca la mayoría de lazos de regulación de temperatura.
  Para el rango de tiempos de retardo especialmente elevados $1/10\ t_a < t_u < 1/3\ t_a$ suele ser posible diseñar un regulador PI adecuado.
- **Comportamiento suficientemente lineal durante un rango de trabajo suficientemente grande**
  Esto significa que, tanto para la identificación como para el modo de regulación normal, los efectos no lineales dentro del rango de trabajo especificado pueden ignorarse. Sin embargo, es posible volver a identificar el proceso cambiando el punto de operación si el proceso de adaptación vuelve a ejecutarse en un entorno reducido del nuevo punto de operación, y si durante la adaptación no se pasa por la no linealidad.
  Si se conocen determinadas no linealidades estáticas (p. ej. características de válvulas), es aconsejable compensarlas previamente con una línea poligonal para linealizar el comportamiento del proceso.
- **Perturbaciones reducidas en procesos de temperatura**
  Las perturbaciones como la transmisión térmica a las zonas adyacentes, o el aumento o descenso térmico por el cambio de estado de grupos constructivos no pueden afectar en exceso a todo el lazo de regulación de temperatura. Puede ser necesaria una adaptación en el punto de operación. Por ejemplo, en la optimización de zonas de un extrusor puede ser necesario calentar simultáneamente todas las zonas.
- **Calidad de las señales de medición**
  Las señales de medición deben tener una calidad suficiente, es decir, que la relación entre la señal parásita y la señal útil sea lo suficientemente reducida.
- **Ganancias del proceso no demasiado grandes**
  Las ganancias del proceso no pueden ser demasiado elevadas. No es necesaria una normalización de los valores reales. Por lo tanto, la ganancia del proceso K se ve afectada por unidades físicas, p. ej.:
  $K = \Delta PV / \Delta LMN$, [K]= °C/ %
  El diseño definitivo del regulador se basa en un cálculo de la ganancia del proceso K, y en principio puede compensar cualquier valor de K. Pero en la fase de aprendizaje K todavía no se conoce, y en combinaciones extremas de ganancia e intensidades de escalón no es posible evitar sobreoscilaciones transitorias. Reduciendo el parámetro LHLM_TUN en el proceso de calentamiento o aumentando LLLM_TUN en el de refrigeración también se reduce el rebase transitorio.

Requisitos de procesos con servoválvulas de acción integral

En procesos con servoválvulas de acción integral existen otros requisitos adicionales a los anteriores:

El tiempo de funcionamiento del motor de la servoválvula debe ser menor al necesario para encontrar un punto de inversión tras un escalón de la variable manipulada (véase la figura Comportamiento del proceso). Cuando no es así, normalmente se trata de una regulación de caudal en la que sólo actúa la servoválvula como acción dominante del proceso. En ese caso no se recomienda el uso del Self-Tuner. El regulador PI paso a paso se puede regular manualmente siguiendo la siguiente regla general:

GAIN = 1, TI = tiempo de posicionamiento del motor

Consulte también

Funcionamiento TUN_ES (Página 7152)

Parámetros de entrada de TUN_ES (Página 7157)

Parámetros de salida de TUN_ES (Página 7159)

Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetro TUN_ES.PHASE (Página 7162)

Parámetro TUN_ES.STATUS_H (Página 7165)

Parámetro TUN_ES.STATUS_D (Página 7166)

Funcionamiento TUN_ES

Para controlar u optimizar el regulador, TUN_ES dispone de las siguientes funciones:

- Modo manual
  El Self-Tuner fuerza el regulador al modo manual y le transfiere sus parámetros en manual. En los reguladores paso a paso con realimentación de posición se puede predeterminar el valor manual en formato de coma flotante, así como señales de valor en manual como abrir/cerrar. En los reguladores paso a paso sin realimentación de posición sólo se pueden predeterminar las señales de valor en manual.
- Optimización del regulador
  Para optimizar los parámetros del regulador, primero es necesario medir el tiempo de actuación del actuador, y a continuación identificar el proceso. Partiendo de las características del proceso se diseña un regulador, y a partir de dicho diseño se calculan los nuevos parámetros de regulador, que se activan entonces para el regulador:
  - Primer ajuste
    El primer ajuste sólo puede producirse con un salto de consigna positivo. Generalmente se aplica en la primera puesta en marcha o en caso de un cambio severo del comportamiento del proceso.
  - Optimización fina
    Si cambia el punto de operación o el comportamiento del proceso, el regulador puede reoptimizarse online durante un salto positivo de la consigna. La optimización sólo es posible en reguladores paso a paso con realimentación de posición.

- **Optimización del comportamiento de regulación**
  El regulador está diseñado básicamente para una respuesta óptima ante perturbaciones. Los "rigurosos" parámetros resultantes provocarían rebases transitorios de entre el 10 y el 20 % en la intensidad de los saltos de la consigna.
  La primera solución se basa en la **descomposición estructural**. Las acciones P o D se incluyen en el lazo de realimentación. De ese modo, el cambio de la consigna sólo afecta a la acción I del regulador. La descomposición estructural sólo puede usarse en reguladores paso a paso con realimentación de posición.
  Si no fuese posible una descomposición estructural para el regulador, para optimizar el comportamiento de regulador piloto puede activarse la siguiente función:
- **Conmutación entre estructuras**
  Conmutación cercana a modo de regulación P(D) o especificación predictiva de la variable manipulada con grandes saltos positivos de la consigna. En reguladores paso a paso sin realimentación de posición, sólo es posible la conmutación cercana al modo de regulación P(D).
- **Guardar los parámetros de regulador**
  Si los parámetros actuales del regulador se consideran válidos, pueden guardarse antes de modificarse manualmente en parámetros propios previstos para tal fin en el DB de instancia del Self-Tuner. En una optimización del regulador, los parámetros guardados se sobrescriben con los valores válidos antes de la optimización.
- **Cargar de nuevo parámetros de regulador guardados**
  Esta función permite volver a activar los últimos parámetros guardados del regulador.
- **Modificar los parámetros del regulador**
  Para ajustar manualmente parámetros del regulador, se pueden especificar en el DB de instancia del Self-Tuner a través de parámetros especiales, y transferirlos al regulador con esta función.

Con TUN_ES no se puede realizar una optimización de refrigeración ni un funcionamiento con zona de regulación.

Activación de las funciones

Función	Bit de inicio	Bit de habilitación	Condición de inicio: PHASE	Condición de inicio: Salto de consigna en SP
Modo manual	—	MAN_ON LMNS_ON	0 a 7	—
Primer ajuste con medición del tiempo de posicionamiento del motor	ADAPT1ST	—	0, 4	pos.: > MIN_STEP
Primer ajuste sin medición del tiempo de actuación del actuador u optimización fina (sólo es posible en un regulador paso a paso sin realimentación de posición)	ADAPT_ON	—	4	pos.: > MIN_STEP

Función	Bit de inicio	Bit de habilitación	Condición de inicio: PHASE	Condición de inicio: Salto de consigna en SP
Conmutación entre estructuras	—	STRUC_ON	4	pos.: > MIN_STEP neg.: > MIN_STEP
Guardar parámetros del regulador	SAVE_PAR	—	2 a 7	—
Cargar de nuevo parámetros del regulador	UNDO_PAR	—	4	—
Cambiar parámetros de regulador	LOAD_PAR	—	4	—

Nota

Los bits de inicio se desactivan una vez que la función se ha iniciado correctamente y los de habilitación permanecen activados.

Consulte también

Descripción de TUN_ES (Página 7149)

Parámetros de entrada de TUN_ES (Página 7157)

Parámetros de salida de TUN_ES (Página 7159)

Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetro TUN_ES.PHASE (Página 7162)

Parámetro TUN_ES.STATUS_H (Página 7165)

Parámetro TUN_ES.STATUS_D (Página 7166)

Conmutación entre estructuras

Descomposición estructural

La descomposición estructural es una función del regulador. En el modo de regulación con descomposición estructural, las acciones P y D se ubican en el lazo de realimentación, es decir, cualquier modificación de la consigna afectará únicamente a la acción I del regulador. En caso de escalones de la consigna, la variable manipulada no registra un escalón, sino que discurre en forma de rampa.

Esto no altera la respuesta a perturbaciones.

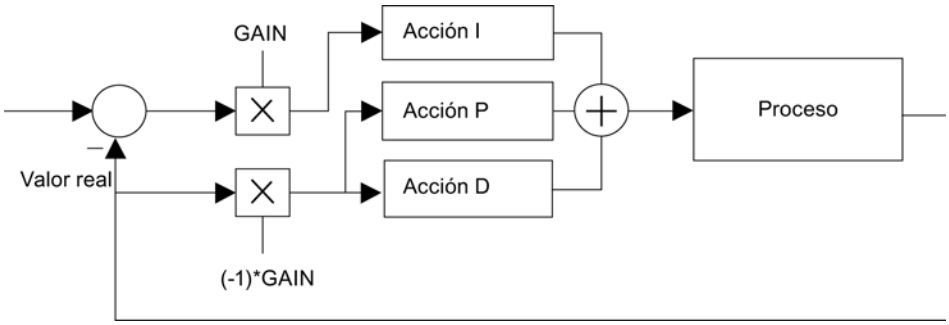


Figura 9-8 Modo de regulación con descomposición estructural

En el modo de regulación sin descomposición estructural, las acciones P y D se ubican, al igual que la acción I, en el error de regulación. En este modo de operación puede seleccionarse una conmutación entre estructuras (STRUC_ON = TRUE).

El modo de regulación con descomposición estructural es posible en los siguientes paquetes de regulación:

- FM355/455
- Standard PID Control
- Modular PID Control

Nota

- Si no se dispone de realimentación de posición de la servoválvula, no debe aplicarse la descomposición estructural. Por lo tanto, deben dejarse la acción P y la acción D en el error de regulación.
- En la regulación con descomposición estructural, la zona muerta no tiene apenas efecto en el error de regulación.
- Con CONT_S y TCONT_S solo es posible la regulación sin descomposición estructural.

Conmutación entre estructuras con acciones P y D en el error de regulación

En la configuración predeterminada, la conmutación entre estructuras está desactivada. De hecho, solo debe activarse si la descomposición estructural no está disponible. Para activar la conmutación entre estructuras, ajuste STRUC_ON = TRUE. En función de la ganancia total

del regulador (producto de la ganancia del proceso por la ganancia del regulador), el bloque elige automáticamente entre dos acciones de control:

- Aumentar la acción I temporalmente, regulación cercana a P(D):
  Con un salto de consigna positivo ≥ MIN_STEP la acción I del regulador se aumenta significativamente de forma temporal y la ganancia se reduce ligeramente, de modo que se utiliza un regulador muy próximo a P(D). Cerca de la consigna se cambia de nuevo a los valores originales de la acción I y de la ganancia (PHASE = 5).
- Especificación predictiva temporal de la variable manipulada:
  Los procesos con tiempos de retardo elevados no son adecuados para una regulación P(D). Por ello, tras un escalón positivo de consigna ≥ MIN_STEP se emite de forma constante la variable manipulada que se requiere estacionaria para la nueva consigna. Cerca de la consigna se vuelve a cambiar sin discontinuidad al modo de regulador PI o PID. Con ello se obtiene un comportamiento lento pero sin sobreoscilaciones transitorias (PHASE = 6).

Nota

En la regulación sin realimentación de posición solo es posible la primera acción de control (PHASE = 5).

Si, con escalones de consigna positivos (p. ej. en fases de calentamiento), la conmutación entre estructuras no da buenos resultados (comportamiento transitorio muy lenta), puede desactivarla con STRUC_ON = FALSE, teniendo en cuenta que pueden producirse pequeños rebases transitorios.

Modos de operación de la estructura del regulador	PFDB_SEL, DFDB_SEL [1]	STRUC_ON
Activar la descomposición estructural	TRUE	FALSE
Activar la conmutación entre estructuras	FALSE	TRUE

[1] Parámetros de regulador PFDB_SEL, DFDB_SEL: Con CONT_S, estos parámetros no están disponibles. Esto corresponde al ajuste PFDB_SEL = DFDB_SEL = FALSE (sin descomposición estructural).

Nota

- En caso de escalones de consigna, la descomposición estructural impide las sobreoscilaciones transitorias de la variable regulada. Si es posible, active la descomposición estructural.
- Si el regulador no permite la descomposición estructural, active la conmutación entre estructuras. La conmutación entre estructuras no debe estar activada al mismo tiempo que la descomposición estructural. Si desea corregir más rápidamente los escalones de consigna positivos y no tiene inconveniente en que se produzcan pequeñas sobreoscilaciones transitorias, puede desactivar la conmutación entre estructuras.

Parámetros de entrada de TUN_ES

Tabla 9-201

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP	0.0	REAL	0.0	En la entrada SP se introduce la consigna. Ésta se interconecta con el regulador a través de SP_OUT. Recuerde que la consigna debe estar comprendida dentro de los límites de consigna del regulador.
PV	4.0	REAL	0.0	PV se interconecta con el valor real del regulador.
MAN	8.0	REAL	0.0	En la entrada MAN se introduce el valor manual. Si MAN_ON = TRUE, el valor manual MAN se transmite a la variable manipulada del regulador. Con MAN_ON = TRUE y MAN = 0.0, la calefacción está desconectada. Si MAN_ON = FALSE el regulador especifica la variable manipulada. Se admiten valores entre -100 y 100 %.
LMNR	12.0	REAL	0.0	LMNR se conecta con la realimentación de posición del regulador (si existe). Se admiten valores entre 0.0 y 100 %.
MIN_STEP	16.0	REAL	10.0	En la entrada MIN_STEP se puede definir la altura mínima del escalón para la consigna a partir de la cual se puede realizar una primera optimización o adaptación. Con la conmutación entre estructuras activada (STRUC_ON=TRUE), para que se active es necesario que el salto de la consigna sea mayor que MIN_STEP. Recomendación: MIN_STEP ≥ 10 % del rango de consigna/valor real requerido
LHLM_TUN	20.0	REAL	80.0	En la entrada LHLM_TUN se introduce el salto de la variable manipulada para el primer ajuste. En posteriores optimizaciones finas se calcula automáticamente el escalón de la variable manipulada. Se admiten valores entre 0.0 y 100 %.
NORM_FAC	24.0	REAL	1.0	En reguladores con ganancia normalizada (Standard PID Control) se aplica un factor de normalización. Éste se calcula a partir de los límites de rango del valor real del regulador y se aplica en la entrada NORM_FAC.
PULSE_TM	28.0	TIME	0 ms	PULSE_TM se conecta con la duración de impulso mínima del regulador. PULSE_TM ≥ 1 ms
C_LMNUP	32.0	BOOL	FALSE	C_LMNUP se conecta con la señal de la variable manipulada QLMNUP del regulador.
C_LMNDN	32.1	BOOL	FALSE	C_LMNDN se conecta con la señal de variable manipulada QLMNDN del regulador.
MAN_ON	32.2	BOOL	FALSE	Si MAN_ON = TRUE, la variable manipulada del regulador se especifica a través de la entrada MAN. Si MAN_ON = FALSE el regulador especifica la variable manipulada.
LMNR_HS	32.3	BOOL	FALSE	LMNR_HS se conecta con la señal de tope LMNR_HR del regulador.
LMNR_ON	32.4	BOOL	FALSE	Si LMNR_ON = TRUE, se puede conmutar un regulador paso a paso con realimentación de posición con LMNS_ON = TRUE o con MAN_ON = TRUE al modo manual. Si LMNR_ON = FALSE, se puede conmutar un regulador paso a paso sin realimentación de posición con LMNS_ON = TRUE al modo manual.
LMNS_ON	32.5	BOOL	TRUE	Si LMNS_ON está activado, se pueden variar las señales de salida en los parámetros LMNUP y LMNDN.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMNUP	32.6	BOOL	FALSE	Con LMNUP se puede variar la señal de salida ABRIR del regulador.
LMNDN	32.7	BOOL	FALSE	Con LMNDN se puede variar la señal de salida CERRAR del regulador.
PID_ON	33.0	BOOL	FALSE	En la entrada PID_ON se puede determinar si el regulador optimizado debe trabajar como regulador PI o PID: Regulador PID: PID_ON = TRUE Regulador PI: PID_ON = FALSE Puede ocurrir que en algunos tipos de proceso se diseñe un regulador PI a pesar de PID_ON = TRUE.
STRUC_ON	33.1	BOOL	FALSE	Si STRUC_ON = TRUE, en caso de escalones de la consigna, una conmutación de la estructura (de regulación PI(D) a P(D)) garantiza que se eviten ampliamente sobreoscilaciones transitorias. La entrada STRUC_ON no puede ponerse = TRUE si en el regulador las acciones P y D están incluidas en el lazo de realimentación.
WRITE_DIS	33.2	BOOL	FALSE	En combinación con el FM Regulación, el Self-Tuner debe saber si el bloque driver está dispuesto a enviar datos al FM. Ése es el caso si LOAD_OP = LOAD_PAR = FALSE. El resultado de la operación lógica O entre LOAD_OP y LOAD_PAR se interconecta con WRITE_DIS.
CYCLE	34.0	TIME	100 ms	En la entrada CYCLE hay que parametrizar el tiempo de muestreo. CYCLE debe coincidir con el tiempo de muestreo del regulador ajustado y con la diferencia de tiempo entre dos llamadas (tiempo de ciclo del OB de alarma cíclica considerando los factores de ciclo). ≥ 1 ms

Consulte también

Descripción de TUN_ES (Página 7149)

Funcionamiento TUN_ES (Página 7152)

Parámetros de salida de TUN_ES (Página 7159)

Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetro TUN_ES.PHASE (Página 7162)

Parámetro TUN_ES.STATUS_H (Página 7165)

Parámetro TUN_ES.STATUS_D (Página 7166)

Parámetros de salida de TUN_ES

Tabla 9-202

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MAN_OUT	38.0	REAL	0.0	MAN_OUT se interconecta con el valor de entrada manual del regulador. Si MAN_ON = TRUE, el Self-Tuner emite en la salida MAN_OUT el valor predeterminado en la entrada MAN.
SP_OUT	42.0	REAL	0.0	SP_OUT se conecta con la entrada de consigna del regulador. El Self-Tuner escribe la consigna especificada en la entrada SP en la salida SP_OUT. En modo manual MAN_ON = TRUE la consigna se corrige según el valor real: SP_OUT = PV.
GAIN	46.0	REAL	1.5	GAIN se conecta con la ganancia proporcional del regulador (ganancia del regulador).
TI	50.0	TIME	1 h	TI se conecta con el tiempo integral (tiempo de acción integral) del regulador.
TD	54.0	TIME	0 ms	TD se conecta con el tiempo de acción derivada (tiempo de anticipación) del regulador.
TM_LAG	58.0	TIME	1 s	TM_LAG se conecta con el tiempo de retardo del diferenciador del regulador.
MTR_TM	62.0	TIME	30 s	MTR_TM se conecta con el tiempo de posicionamiento del motor del regulador.
DEADB_W	66.0	REAL	0.0	DEADB_W se conecta con la amplitud de zona muerta del regulador.
PHASE	70.0	INT	0	Parámetro TUN_ES.PHASE (Página 7162) muestra el progreso actual de la optimización del regulador. En la optimización de Standard PID Control, la salida PHASE se conecta con la entrada PHASE de la instrucción PID_ES.
STATUS_H	72.0	INT	0	Parámetro TUN_ES.STATUS_H (Página 7165) muestra resultados de la optimización a través de la búsqueda del punto de inversión en el proceso de calentamiento.
STATUS_D	74.0	INT	0	Parámetro TUN_ES.STATUS_D (Página 7166) muestra resultados de la optimización a través del diseño del regulador en el proceso de calentamiento.
QMAN_ON	76.0	BOOL	FALSE	QMAN_ON se conecta con la conmutación manual/automático del regulador.
QLMNS_ON	76.1	BOOL		QLMNS_ON se conecta con el parámetro "Activar señales en manual" del regulador.
QLMNUP	76.2	BOOL		QLMNUP se conecta con el parámetro "Señal manipulada en manual abrir" del regulador.
QLMNDN	76.3	BOOL		QLMNDN se conecta con el parámetro "Señal manipulada en manual cerrar" del regulador.
QI_SEL	76.4	BOOL	TRUE	QI_SEL se conecta con el activador/desactivador de la acción I del regulador I_SEL. En el FM355/455 es necesario conmutar TI a cero en función de QI_SEL: Si QI_SEL = FALSE, entonces TI = 0.
QD_SEL	76.5	BOOL	FALSE	QD_SEL se conecta con el activador/desactivador de la acción D del regulador D_SEL. En el FM355/455 es necesario conmutar TD a cero en función de QD_SEL: Si QD_SEL = FALSE, entonces TD = 0.
QWRITE	76.6	BOOL	FALSE	QWRITE se conecta con las entradas LOAD_PAR y LOAD_OP de la instrucción PID_FM del FM355/455. Con ello se cargan los registros del canal regulador en el FM.
PROCESS	78.0	STRUCT		La estructura PROCESS contiene los parámetros del proceso.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
GAIN	+0.0	REAL	999.0	En la salida PROCESS.GAIN se muestra la ganancia del proceso.
TM_LAG1	+4.0	REAL	0.0	En la salida PROCESS.TM_LAG1 se muestra la constante de tiempo del proceso lenta.
TM_LAG2	+8.0	REAL	0.0	En la salida PROCESS.TM_LAG2 se muestra la constante de tiempo del proceso rápida.
PV00	+12.0	REAL	0.0	En la salida PROCESS. PV00 se muestra el valor de equilibrio del proceso. El valor de equilibrio es el valor real estacionario con la variable manipulada = cero. Equivale a la temperatura ambiente con la instalación fría.
KIG	+16.0	REAL	0.0	En la salida PROCESS.KIG se muestra la relación de rampa máxima del valor real en el calentamiento.
MTR_TM	+20.0	REAL	0.0	En la salida MTR_TM se muestra el tiempo de posicionamiento del motor.
PI_CON	102.0	STRUCT		La estructura PI_CON contiene los parámetros del regulador PI.
GAIN	+0.0	REAL	1.5	PI_CON.GAIN muestra la ganancia proporcional del regulador PI. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
TI	+4.0	REAL	3600.0	PI_CON.TI muestra el tiempo integral del regulador PI. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
PID_CON	110.0	STRUCT		La estructura PID_CON contiene los parámetros del regulador PID.
GAIN	+0.0	REAL	0.0	PID_CON.GAIN muestra la ganancia proporcional del regulador PID. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
TI	+4.0	REAL	0.0	PID_CON.TI muestra el tiempo integral del regulador PID. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
TD	+8.0	REAL	0.0	PID_CON.TD muestra el tiempo de acción derivada del regulador PID. El valor puede optimizarse manualmente o hacerse efectivo activando LOAD_PAR.
PI_CON_OLD	122.0	STRUCT		La estructura PI_CON_OLD contiene los parámetros antiguos del regulador PI.
GAIN	+0.0	REAL	1.5	En la salida PI_CON_OLD.GAIN se muestra la ganancia proporcional antigua del regulador PI.
TI	+4.0	REAL	3600.0	En la salida PI_CON_OLD.TI se muestra el tiempo integral antiguo del regulador PI.
PID_CON_OLD	130.0	STRUCT		La estructura PID_CON_OLD contiene los parámetros antiguos del regulador PID.
GAIN	+0.0	REAL	1.5	En la salida PID_CON_OLD.GAIN se muestra la ganancia proporcional antigua del regulador PID.
TI	+4.0	REAL	3600.0	En la salida PID_CON_OLD.TI se muestra el tiempo integral antiguo del regulador PID.
TD	+8.0	REAL	0.0	En la salida PID_CON_OLD.TD se muestra el tiempo de acción derivada antiguo del regulador PID.
TU	142.0	REAL	0.0	En la salida TU se muestra el tiempo de retardo determinado.
TA	146.0	REAL	0.0	En la salida TA se muestra el tiempo de compensación determinado.

Consulte también

Descripción de TUN_ES (Página 7149)
Funcionamiento TUN_ES (Página 7152)
Parámetros de entrada de TUN_ES (Página 7157)
Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetros de entrada/salida de TUN_ES

Tabla 9-203

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
ADAPT1ST	150.0	BOOL	FALSE	En la optimización inicial del proceso o si no se desea utilizar las conclusiones de las optimizaciones anteriores, como la intensidad del salto la variable manipulada o las constantes del tiempo del filtro, hay que realizar un primer ajuste. Para ello hay que activar ADAPT1ST = TRUE; entonces el Self-Tuner realiza una marcha de inicialización y ajusta los parámetros predeterminados descritos en la marcha de inicialización. El Self-Tuner vuelve a desactivar ADAPT1ST a FALSE. No es conveniente poner ADAPT1ST en TRUE en un rearranque completo (en caliente) de la CPU en el OB 100.
ADAPT_ON	150.1	BOOL	FALSE	Para lanzar una reoptimización, active ADAPT_ON = TRUE. Con un salto de consigna posterior superior a MIN_STEP, se pone en marcha la identificación (PHASE = 2). El Self-Tuner vuelve a poner ADAPT_ON en FALSE.
STEADY	150.2	BOOL	FALSE	Una vez que el valor real ha alcanzado un estado estacionario, la optimización se puede finalizar anticipadamente con STEADY, aunque el Self-Tuner todavía se encuentre en PHASE 3. También es posible volver a realizar el diseño del regulador en estado estacionario con STEADY, una vez que el Self-Tuner haya pasado a PHASE 4.
UNDO_PAR	150.3	BOOL	FALSE	Los parámetros de regulador antiguos se transfieren sobre los actuales y se activan de inmediato en función de PID_ON. PI_CON = PI_CON_OLD PID_CON = PID_CON_OLD La función para deshacer cambios sólo es posible en el modo automático (PHASE 4).
SAVE_PAR	150.4	BOOL	FALSE	Los parámetros de regulador actuales se transfieren sobre los antiguos. PI_CON_OLD = PI_CON PID_CON_OLD = PID_CON
LOAD_PAR	150.5	BOOL	FALSE	Los parámetros modificados PI_CON y PID_CON se transmiten al regulador y se activan.

Consulte también

Descripción de TUN_ES (Página 7149)
Funcionamiento TUN_ES (Página 7152)
Parámetros de entrada de TUN_ES (Página 7157)

Parámetros de salida de TUN_ES (Página 7159)
Parámetro TUN_ES.PHASE (Página 7162)
Parámetro TUN_ES.STATUS_H (Página 7165)
Parámetro TUN_ES.STATUS_D (Página 7166)

Parámetro TUN_ES.PHASE

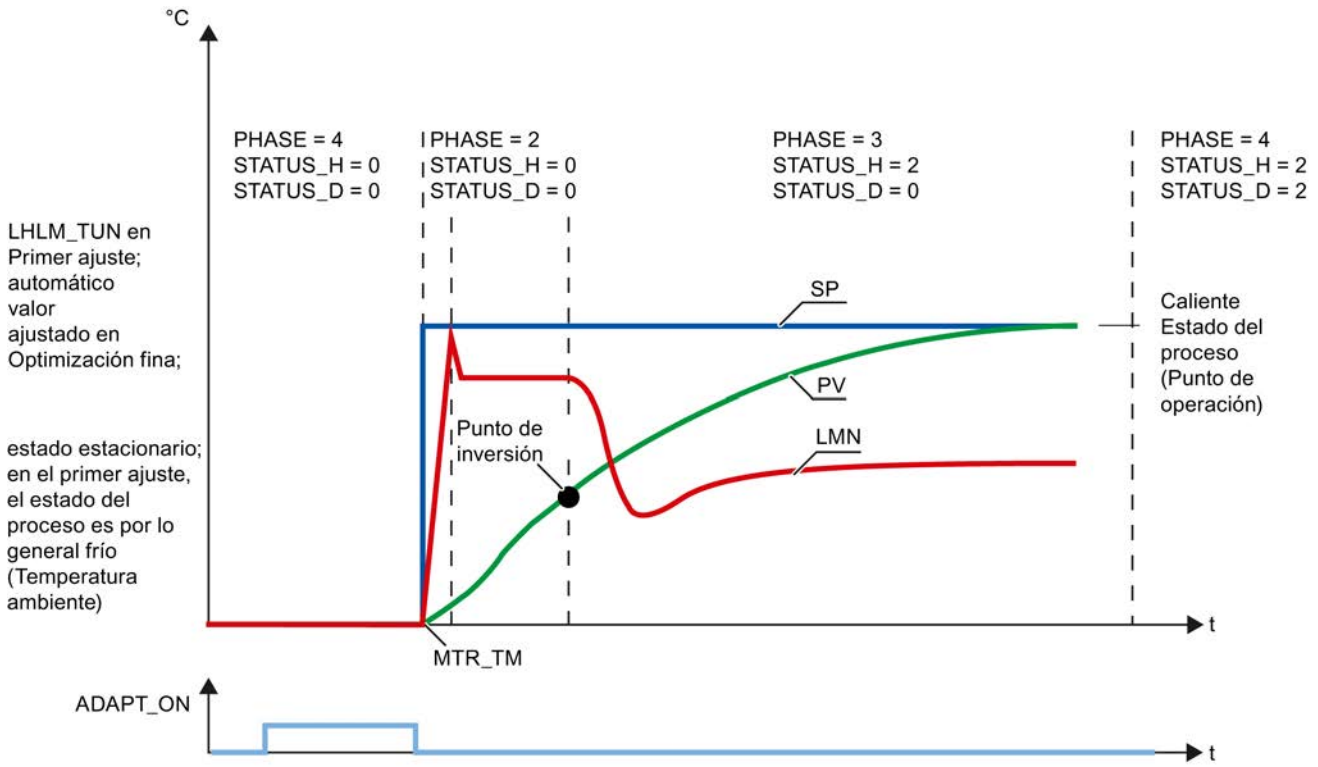
PHASE	Modo de operación	Descripción
0	Modo automático: Modo de regulación con parámetros predeterminados	Al crear un DB de instancia para el Self-Tuner, el parámetro PHASE tiene preasignado el valor cero. Dado que LMNS_ON tiene preasignado el estado TRUE, el Self-Tuner cambia a PHASE 7 (modo manual) tras la primera ejecución. Desde el modo manual se puede iniciar directamente el primer ajuste (ADAPT_ON = TRUE) o cambiar al modo automático. En el primer cambio al modo automático MAN_ON = FALSE, se adoptan los parámetros de PI_CON. Los valores predeterminados para GAIN y TI son: GAIN = 1.5, TI = 1h. Salir de la fase con MAN_ON, LMNS_ON o ADAPT1ST
1	Optimización del regulador: Desconectar calefacción y esperar a un salto de consigna	Una vez que la optimización ha comenzado con ADAPT1ST = TRUE, el Self-Tuner cierra la servoválvula (variable manipulada = 0 o LMNS_ON = LMNDN = TRUE) y espera a un salto positivo de la consigna ≥ MIN_STEP.
2	Optimización del regulador: Medición del tiempo de posicionamiento del motor, avance hasta una posición constante de la servoválvula y búsqueda de punto de inversión	En cuanto se predetermina un salto de consigna ≥ MIN_STEP en sentido positivo hacia el punto de operación del estado del proceso caliente, en el primer ajuste a MAN_OUT se le preasigna LHLM_TUN y en la reoptimización a MAN_OUT se le preasigna el valor calculado por el Self-Tuner. MIN_STEP debería ser mayor que el 10 % del rango de trabajo entre consigna y valor real. En la regulación paso a paso con realimentación de posición (LMNR_ON = TRUE) QMAN_ON obtiene el valor TRUE. El regulador regula la válvula al valor de MAN_OUT y TUN_ES calcula el tiempo de posicionamiento del motor MTR_TM. En la regulación paso a paso sin realimentación de posición, QLMNS_ON y QLMNUP se ajustan a TRUE y la válvula se mueve hasta el tope superior. Al alcanzar el tope superior (LMNR_HS =TRUE), TUN_ES calcula el tiempo de posicionamiento del motor y lo transmite al regulador. A continuación la válvula se cierra hasta el valor de MAN_OUT (QLMNDN = TRUE).

PHASE	Modo de operación	Descripción
3	Optimización del regulador: Regulación con un regulador PI aproximativo	Si se detecta el punto de inversión de la respuesta al salto (STATUS_H = 2) o si el valor real ha alcanzado el 70 % de la intensidad del salto de la consigna (STATUS_H = 3), se diseña un regulador PI de ajuste cuidado. La ganancia del regulador GAIN se limita al rango entre 0.2 y 20.0. El regulador trabaja a partir de ese momento como regulador PI e intenta regular el proceso a un estado estacionario. El Self-Tuner guarda los nuevos parámetros de regulador PI en GAIN, TI y PI_CON. Los valores anteriores del regulador PI se guardan en PI_CON_OLD. Si el tiempo necesario para alcanzar el estado estacionario es muy prolongado (respuesta transitoria muy lenta en lazos de regulación de temperatura), ya puede iniciarse el diseño del regulador con los datos actuales al aproximarse al estado estacionario activando STEADY = TRUE. También es posible volver a iniciar el diseño del regulador con los datos actuales posteriormente en la fase 3 ó 4 activando STEADY = TRUE. Con frecuencia, así se mejora un poco más el diseño del regulador. El hecho de que se produzca un rebase transitorio o de que no se encuentre un punto de inversión, en un primer ajuste puede deberse a que se ha seleccionado una intensidad de salto de la variable manipulada excesiva LHLM_TUN, y no tiene que derivar necesariamente en un mal ajuste del regulador. En el siguiente primer ajuste LHLM_TUN debe seleccionarse aprox. un 20 % más pequeño. Si el bloque ha detectado el estado estacionario o si el tiempo 30 × TI (TI: tiempo de acción integral del regulador PI ajustado en PHASE = 3) ha transcurrido desde el salto de la variable manipulada, se realiza un diseño mejorado del regulador, y se cambia a PHASE = 4. El Self-Tuner guarda los nuevos parámetros de regulador PI/PID determinados en GAIN, TI, TD y PI_CON o PID_CON. Los valores anteriores del regulador PID se guardan en PID_CON_OLD. A ser posible se diseñan un regulador PID y un regulador PI. Si PID_ON = TRUE, se activa el regulador PID, de lo contrario el regulador PI. En PHASE 4 se puede cambiar en todo momento entre los dos reguladores sin discontinuidad. En procesos complejos, la instrucción diseña siempre un regulador PI independientemente de PID_ON. La ganancia del regulador GAIN se limita de tal manera que la ganancia del lazo abierto (producto resultante de la ganancia del regulador por la del proceso) sea inferior a 40.0.
4	Modo automático: Funcionamiento normal del regulador con/sin zona de regulación	En esta fase el regulador regula con sus parámetros optimizados. Desde este estado se puede iniciar una reoptimización o un primer ajuste.
5	Modo automático, conmutación entre estructuras con salto de consigna: Acción I desactivada, regulación P(D)	Con un salto de consigna positivo ≥ MIN_STEP la acción I del regulador se aumenta significativamente de forma temporal y la ganancia se reduce ligeramente, de modo que se utiliza un regulador muy próximo a P(D). Cerca de la consigna se cambia de nuevo a los valores originales de la acción I y de la ganancia (PHASE = 5 en el proceso del Self-Tuner).

PHASE	Modo de operación	Descripción
6	Modo automático, conmutación entre estructuras con salto de consigna: Especificación predictiva de la variable manipulada	Sólo en un regulador paso a paso con realimentación de posición Los procesos con tiempos de retardo elevados no son adecuados para una regulación P(D). Por ello, tras un salto positivo de consigna ≥ MIN_STEP, la magnitud manipulada necesaria estacionaria para la nueva consigna se emite de forma constante. Cerca de la consigna se vuelve a cambiar sin discontinuidad al modo de regulador PI o PID. Con ello se obtiene un comportamiento lento pero sin rebases transitorios.
7	Modo manual	El modo manual tiene prioridad frente a todos los modos restantes. Cualquier primer ajuste o reoptimización se interrumpe. Al desactivar el modo manual (MAN_ON = FALSE y LMNS_ON = FALSE), el regulador pasa al modo automático (PHASE = 4 en el proceso del Self-Tuner) y continúa regulando con los parámetros disponibles. Si se utiliza un regulador paso a paso con realimentación de posición (LMNR_ON = TRUE), se puede cambiar al modo manual con LMNS_ON = TRUE o con MAN_ON = TRUE. Con un regulador paso a paso sin realimentación de posición (LMNR_ON = FALSE) sólo se puede cambiar al modo manual con LMNS_ON = TRUE. Si la entrada MAN_ON se pone a TRUE, la salida QMAN_ON se pone a TRUE y la salida MAN_OUT a MAN. Si la entrada LMNS_ON se pone a TRUE, la salida QLMNUP se pone a LMNUP y la salida QLMNDN a LMNDN.

Optimización de calefacción con un regulador paso a paso sin realimentación de posición

La figura muestra las fases del primer ajuste con medición del tiempo de posicionamiento del motor.



Consulte también

Descripción de TUN_ES (Página 7149)
Funcionamiento TUN_ES (Página 7152)
Parámetros de entrada de TUN_ES (Página 7157)
Parámetros de entrada/salida de TUN_ES (Página 7161)
Parámetro TUN_ES.STATUS_H (Página 7165)
Parámetro TUN_ES.STATUS_D (Página 7166)

Parámetro TUN_ES.STATUS_H

STATUS_H	Descripción	Solución
0	Inicialización	
1	Punto de inversión demasiado bajo En la mayoría de los casos, la optimización suele conducir a parámetros buenos de regulador.	Para acercarse más aún al caso ideal, aumente LHLM_TUN o reduzca la intensidad del salto de la consigna.
2	Caso ideal: Punto de inversión encontrado	
3	Punto de inversión demasiado alto Se incluye un factor de seguridad en el cálculo; la optimización suele arrojar en la mayoría de los casos parámetros válidos, pero cautos.	Para acercarse más aún al caso ideal, reduzca LHLM_TUN o aumente la intensidad del salto de la consigna.
11, 12	El tiempo de retardo no se ha podido registrar correctamente. Se sigue trabajando con un valor estimado, que puede provocar parámetros no óptimos del regulador. Si el proceso presenta un comportamiento VZ1 puro (sin tiempo de retardo), con frecuencia el Self-Tuner consigue buenos parámetros de regulador. Dado que para el cálculo posterior se estima un tiempo de retardo generoso, es posible que STATUS_D no lleve a 1 (regulador para comportamiento cercano a VZ1) y que la zona de regulación se desactive, a pesar de que una zona de regulación puede ser conveniente en procesos VZ1.	Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real. Si no está satisfecho con los parámetros del regulador, como mínimo puede utilizar los valores de PROCESS.GAIN y PROCESS.TM_LAG1 para un modelo VZ1 y un diseño de regulador propio según el libro guía.
13	El tiempo de retardo no se ha podido registrar correctamente. Se sigue trabajando con un valor estimado, que puede provocar parámetros no óptimos del regulador.	Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

Consulte también

Descripción de TUN_ES (Página 7149)
Funcionamiento TUN_ES (Página 7152)
Parámetros de entrada de TUN_ES (Página 7157)
Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetro TUN_ES.PHASE (Página 7162)

Parámetro TUN_ES.STATUS_D (Página 7166)

Parámetro TUN_ES.STATUS_D

STATUS_D	Descripción
0	Inicialización
1	Sólo diseño de regulador PI, ya que el proceso tiene un comportamiento cercano a PT1
2	Caso ideal: Reguladores PI y PID posibles, comportamiento VZ2
3	Sólo diseño de regulador PI, ya que el tiempo de retardo, el tiempo muerto o la orden de proceso son elevados
4	Mala estimación Sin diseño de regulador en reoptimización En el primer ajuste se trabaja con el regulador PI aproximado de PHASE = 3. Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

Consulte también

Descripción de TUN_ES (Página 7149)

Funcionamiento TUN_ES (Página 7152)

Parámetros de entrada de TUN_ES (Página 7157)

Parámetros de entrada/salida de TUN_ES (Página 7161)

Parámetro TUN_ES.PHASE (Página 7162)

Parámetro TUN_ES.STATUS_H (Página 7165)

Standard PID Control

Datos técnicos de las instrucciones

Carga de la CPU

El rendimiento de regulación y, por tanto, la rapidez del procesamiento de lazos de regulación concretos dependen exclusivamente del rendimiento de la CPU utilizada. Para poder estimar la carga de una CPU concreta debida a la instalación de Standard PID Control, existen los puntos de referencia siguientes:

- Para un número cualquiera de reguladores, la instrucción debe existir una sola vez en la memoria de usuario de la CPU.
- Por cada regulador se requiere un DB con unos 0,5 kB
- No se requiere memoria para una pila L.

- Las alarmas no son retardadas por PID_CP o PID_ES.
- Datos de referencia para tiempos de ejecución típicos (tiempos de procesamiento) de los bloques en caso de parametrización conforme al ajuste predeterminado en el modo de regulación:

Instrucción	Condiciones	Tiempo de procesamiento (en ms) CPU 315–2AG10	Tiempo de procesamiento (en ms) CPU 416–2XK02
PID_CP	Condiciones típicas	1,3	0,14
PID_ES	sin realimentación de posición, condiciones típicas	1,5	0,16

Ocupación de la memoria de trabajo

El tamaño del área necesaria en la memoria de usuario y, por tanto, el número de lazos de regulación que pueden instalarse en base a la capacidad de memoria disponible pueden consultarse en las tablas siguientes:

Instrucción	Memoria de carga requerida	Memoria de trabajo requerida	Datos locales
PID_CP	8956 bytes	7796 bytes	122 bytes
PID_ES	9104 bytes	7982 bytes	152 bytes
LP_SCHED	1064 bytes	976 bytes	20 bytes

DB de instancia o DB global	Memoria de carga requerida	Memoria de trabajo requerida
DB de PID_CP	1168 bytes	510 bytes
DB de PID_ES	1124 bytes	484 bytes
DB_LOOP (con 5 lazos de regulación)	184 bytes	100 bytes
DB_RMPSK (con punto inicial y 4 nodos)	142 bytes	78 bytes

Tiempo de muestreo

El tiempo de muestreo mínimo que puede ajustarse depende de la clase de rendimiento de la CPU utilizada.

Nota

La precisión de cálculo limitada restringe el tiempo de muestreo realizable hacia abajo. Si el tiempo de muestreo se va reduciendo, las constantes de los algoritmos adoptan cada vez valores numéricos más pequeños. Esto puede provocar un cálculo erróneo de la magnitud manipulada.

Recomendación:

- S7-300: tiempo de muestreo ≥ 20 ms
- S7-400: tiempo de muestreo ≥ 5 ms

PID_CP

Descripción de PID_CP

Además de las funciones de las ramas de valor real y de consigna, la instrucción realiza un regulador PID terminado con salida continua de magnitud manipulada y posibilidad de controlar manualmente la variable manipulada. Las funciones parciales pueden activarse o desactivarse.

La instrucción permite regular instalaciones y procesos técnicos con magnitudes continuas de entrada y salida en los sistemas de automatización SIMATIC S7. Se pueden regular tanto sistemas lentos (temperaturas, niveles, etc.) como sistemas muy rápidos (caudales, revoluciones, etc.).

El regulador se puede utilizar como regulador de valor fijo de manera individual o también en regulaciones de varios lazos como regulador en cascada, de mezcla o de relación.

Arranque

PID_CP dispone de una rutina de inicialización que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado.

En el arranque, la acción I se pone al valor de inicialización I_ITLVAL y se emite en la salida LMN_I. Si se llama en un nivel de alarma cíclica, continúa trabajando a partir de este valor.

Si el generador ramp soak está activado, los intervalos DB_NBR PI[0 ... NBR_PTS].TMV entre los nodos se suman y se indican en la salida del tiempo total T_TM y del tiempo residual total RT_TM. En caso de modificaciones online de PI[n].TMV o en caso de especificación de TM_CONT y TM_SNBR, cambian el tiempo total y el tiempo residual total de la curva de evolución. Ya que el cálculo de T_TM y RS_TM prolonga mucho el tiempo de procesamiento de la función RMP_SOAK si hay muchos nodos, dicho cálculo solo se realiza después del rearranque o si TUPDT_ON = TRUE.

Todas las demás salidas se ponen a sus valores predeterminados.

Rearranque

En caso de rearranque se parte del estado operativo que estaba activo en el momento de la interrupción. La regulación sigue trabajando con los valores que se calculaban en el instante de la interrupción.

Llamada

La instrucción PID_CP debe llamarse de forma equidistante. Utilice para ello un nivel de alarma cíclica (p. ej. OB35).

Si se efectúa una llamada de la instrucción PID_CP como DB multiinstancia, no se crea ningún objeto tecnológico. No se dispone de interfaz de parametrización y puesta en servicio. Los PID_CP se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Cálculo del error de regulación PID_CP

Las funciones siguientes son siempre efectivas y no es posible desactivarlas.

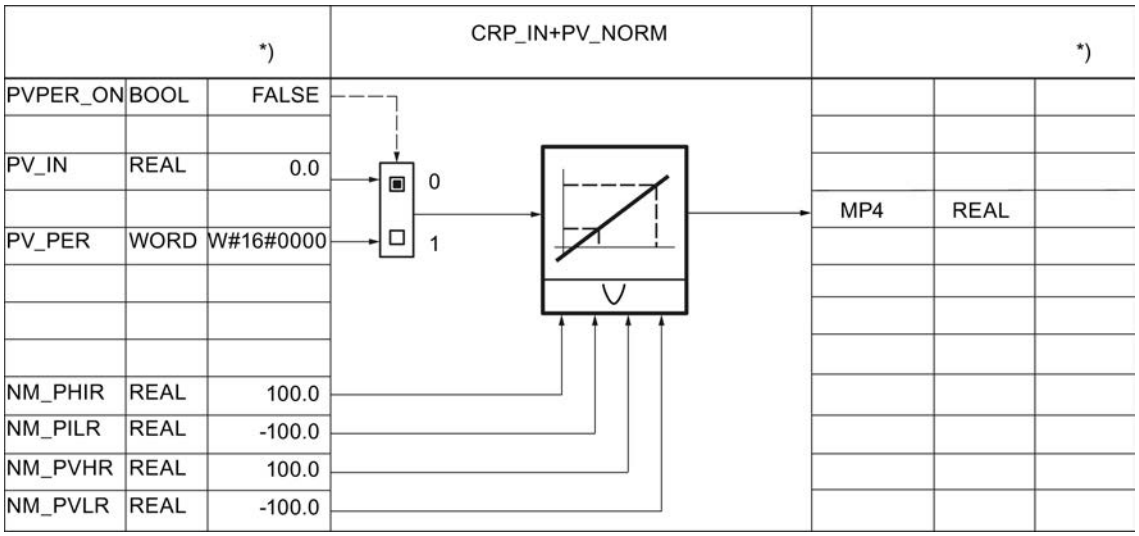
- Limitación de consigna SP_LIMIT,
- Vigilancia de valores absolutos del valor real PV_ALARM,
- Vigilancia de la magnitud de la velocidad de cambio del valor real ROCALARM,
- Vigilancia de valores absolutos del error de regulación ER_ALARM,
- Limitación de la señal de salida LMNLIMIT.

Los valores predeterminados se han seleccionado de modo que el funcionamiento sea correcto sin adaptación individual. Para conseguir buenos resultados de regulación es necesario adaptar siempre los parámetros de estas funciones al proceso.

Acondicionamiento del valor real

El valor real se normaliza a la magnitud física.

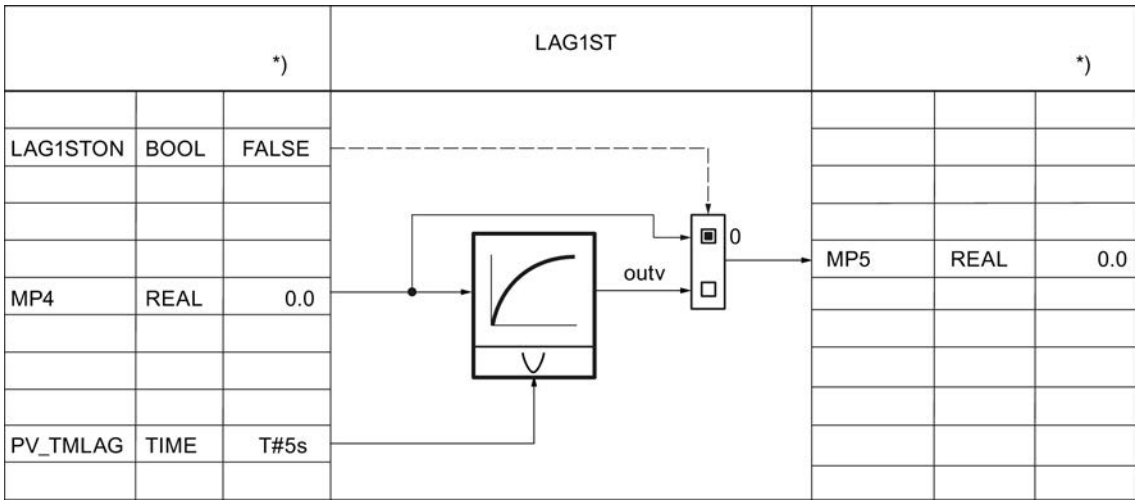
Normalización (PV_NORM) (Página 10066)



*) Ajuste predeterminado al crear el objeto tecnológico.

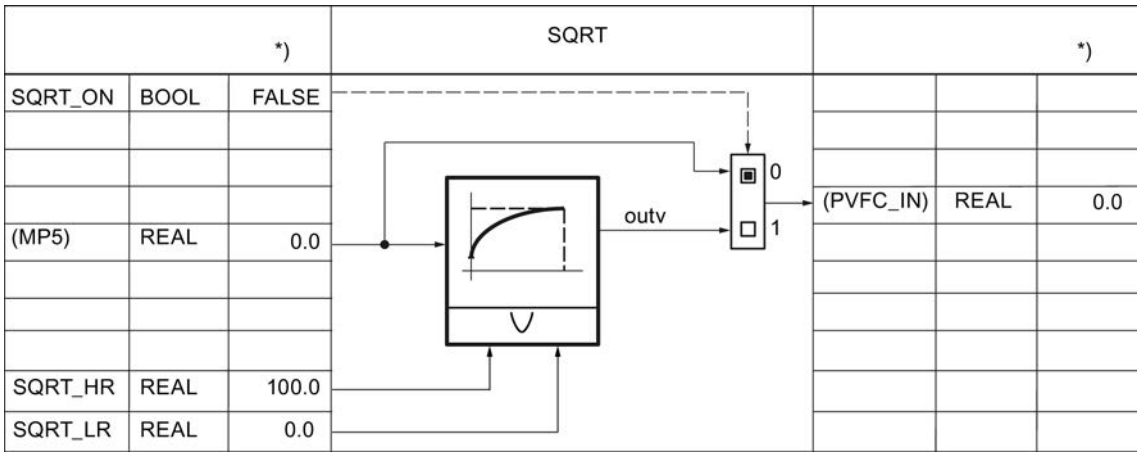
El valor real puede filtrarse con un elemento de retardo de primer orden.

Filtrar magnitud regulada (LAG1ST) (Página 10067)



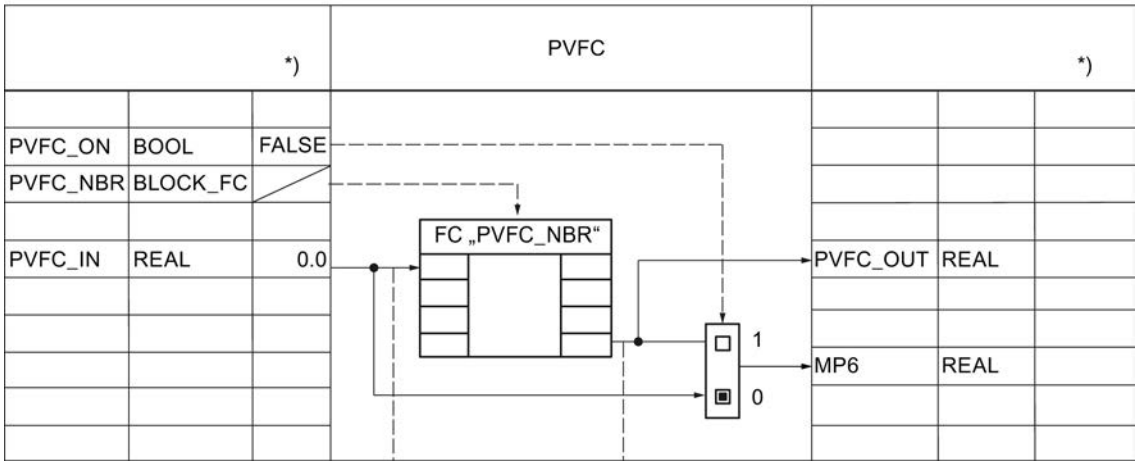
El valor real puede linealizarse con la raíz cuadrada. Para ello se requieren los parámetros siguientes:

Calcular raíz cuadrada (SQRT) (Página 10068)



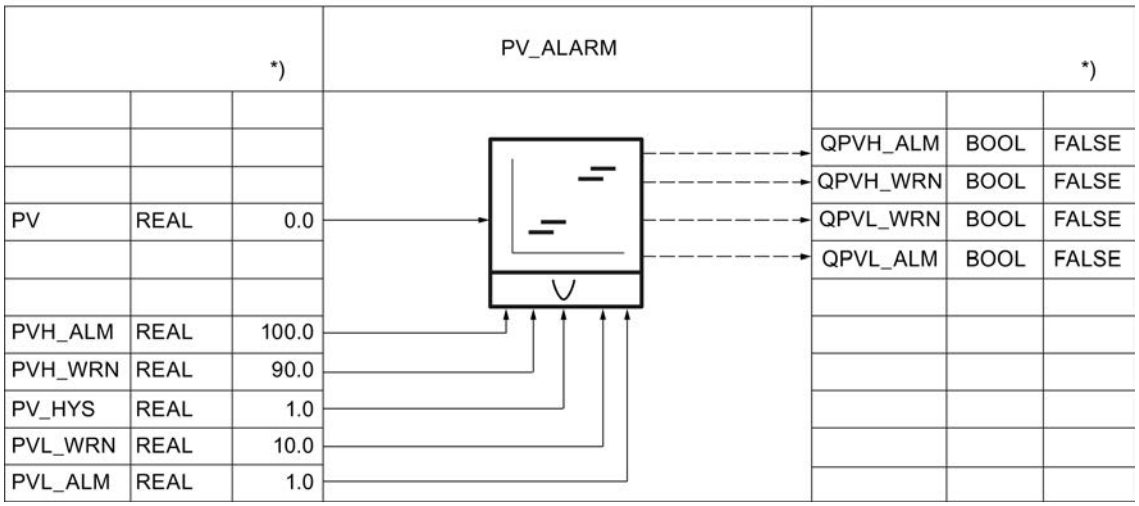
El valor real puede acondicionarse con una función específica del usuario. Para ello se requieren los parámetros siguientes. La interconexión debe programarse en la FC de usuario.

Llamada de FC en la rama de valor real (PVFC) (Página 10070)



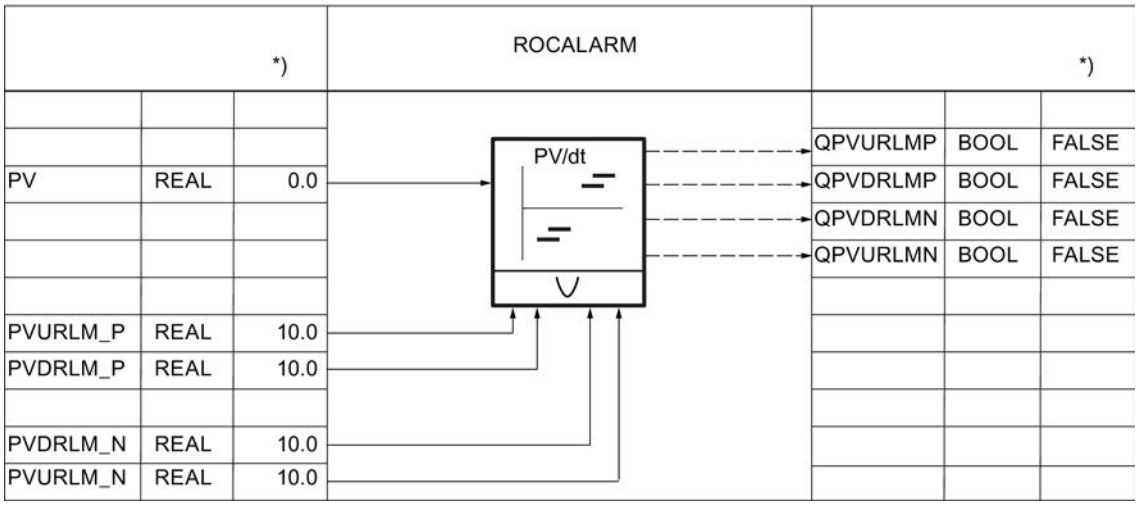
Los límites de advertencia y alarma del valor real se vigilan siempre con los parámetros siguientes.

Vigilar los límites del valor real (PV_ALARM) (Página 10070)



La velocidad de cambio del valor real se limita siempre con los parámetros siguientes.

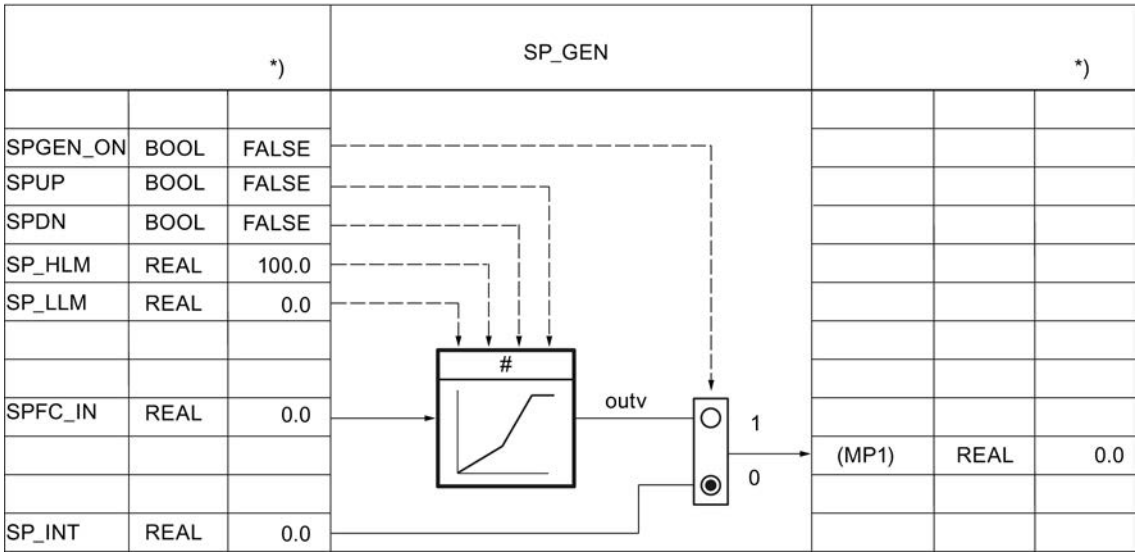
Vigilar la velocidad de cambio del valor real (ROCALARM) (Página 10072)



Acondicionamiento de la consigna

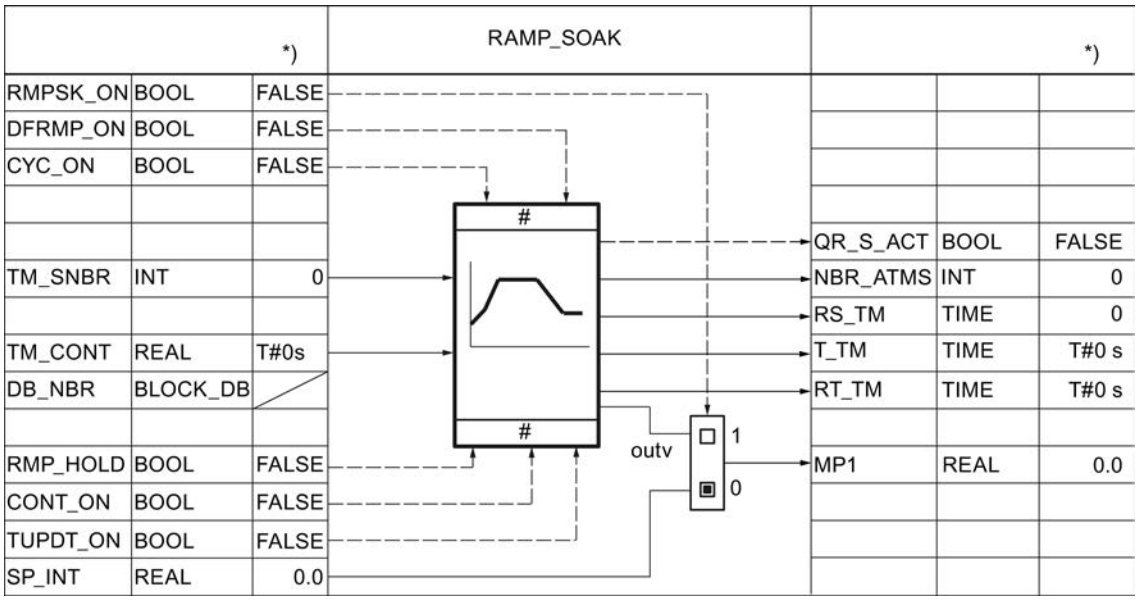
La consigna interna SP_IN se aumenta o reduce con los parámetros siguientes.

Generador de consigna (SP_GEN) (Página 10074)



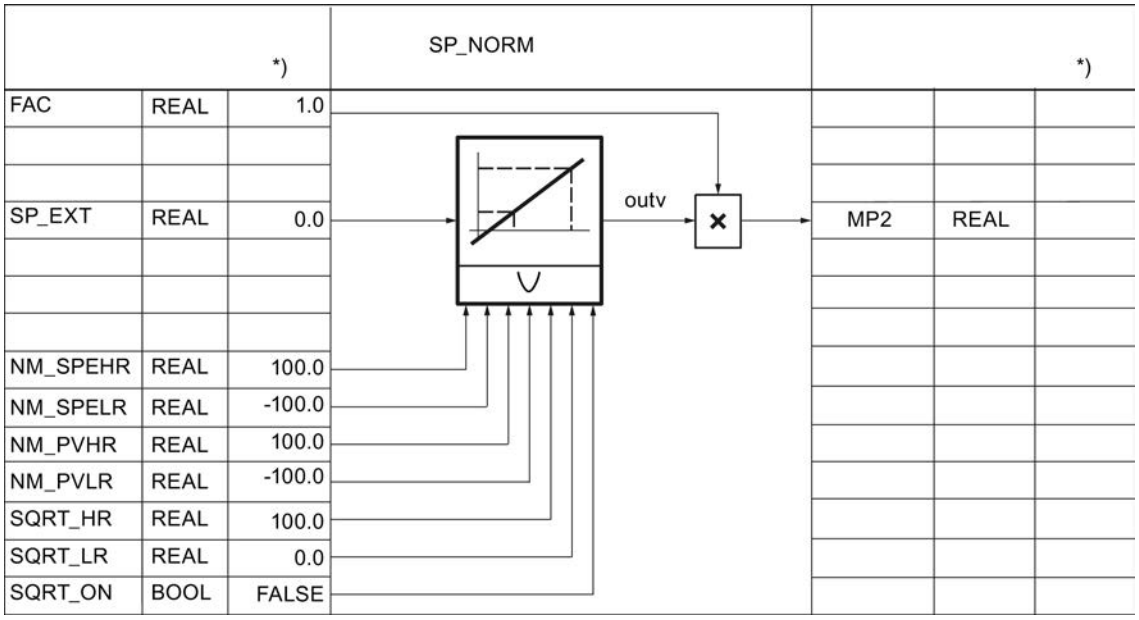
La consigna se especifica mediante un generador ramp soak con los parámetros siguientes.

Generador ramp soak (RMP_SOAK) (Página 10075)



Para reguladores de mezcla y relación, la consigna externa SP_EXT puede normalizarse con los parámetros siguientes.

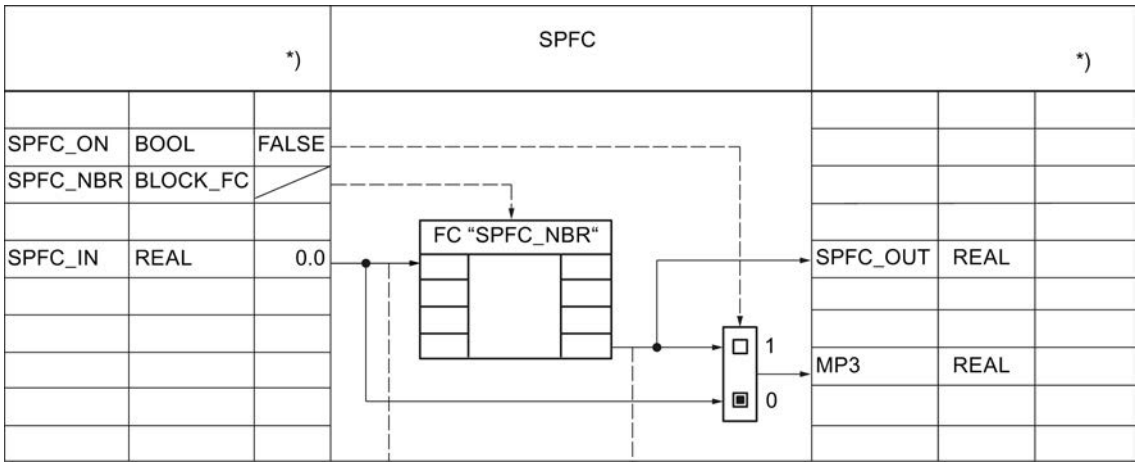
Normalización (Página 10081)



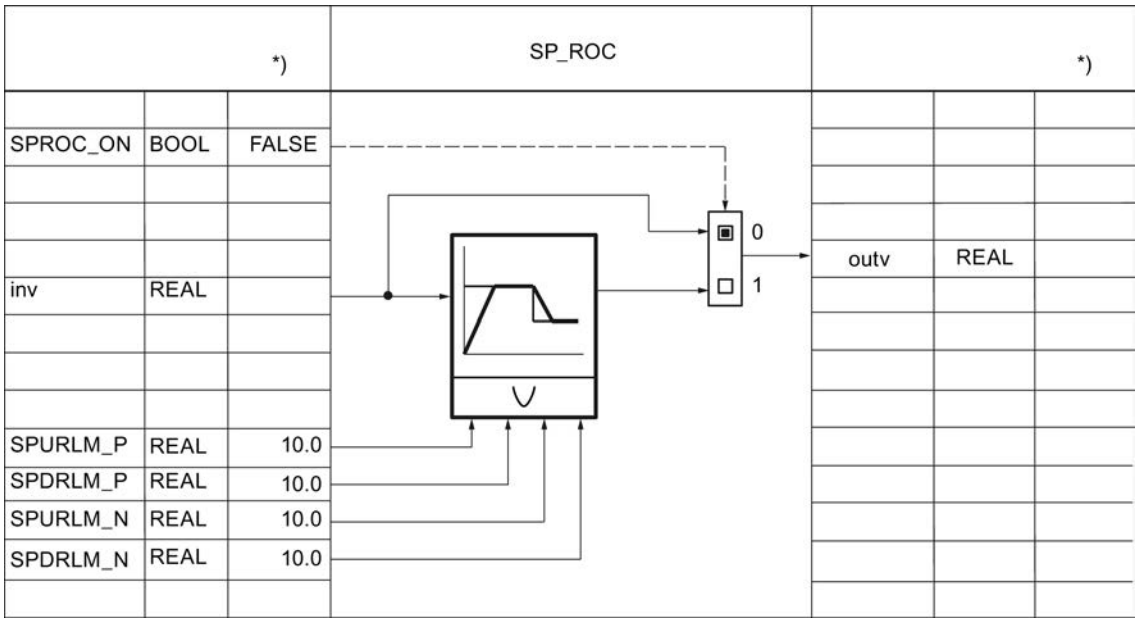
La consigna puede acondicionarse con una función específica del usuario. Para ello se requieren los parámetros siguientes. La interconexión debe programarse en la FC de usuario.

El valor de entrada SPFC_IN es un parámetro implícito. Se observa en la herramienta de configuración a través del punto de medición MP1 (SPEXT_ON = FALSE) o MP2 (SPEXT_ON = TRUE). El valor de salida es accesible en el punto de medición MP3. La entrada SPFC_IN está aplicada a la rama de consigna cuando SPFC_ON =FALSE.

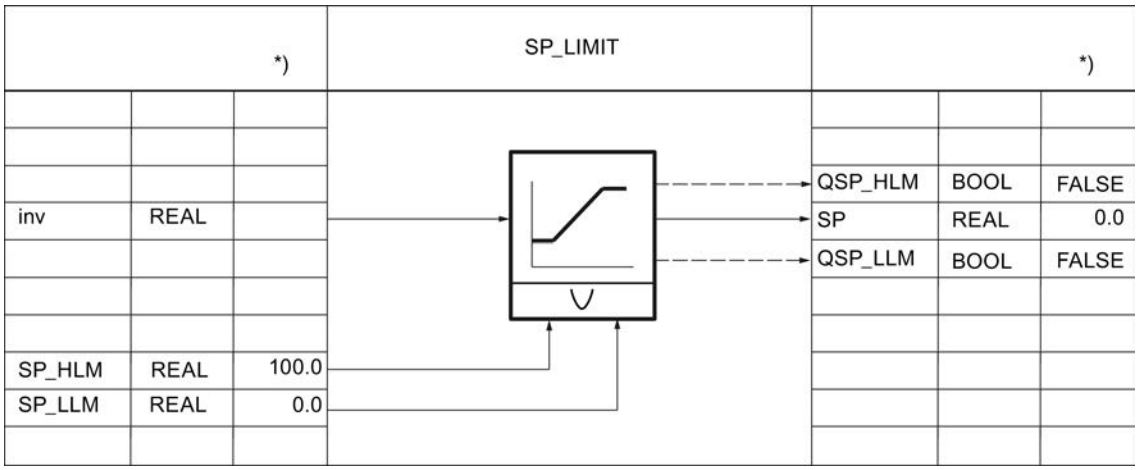
Límites (Página 10083)



La velocidad de cambio de la consigna se limita con los parámetros siguientes.



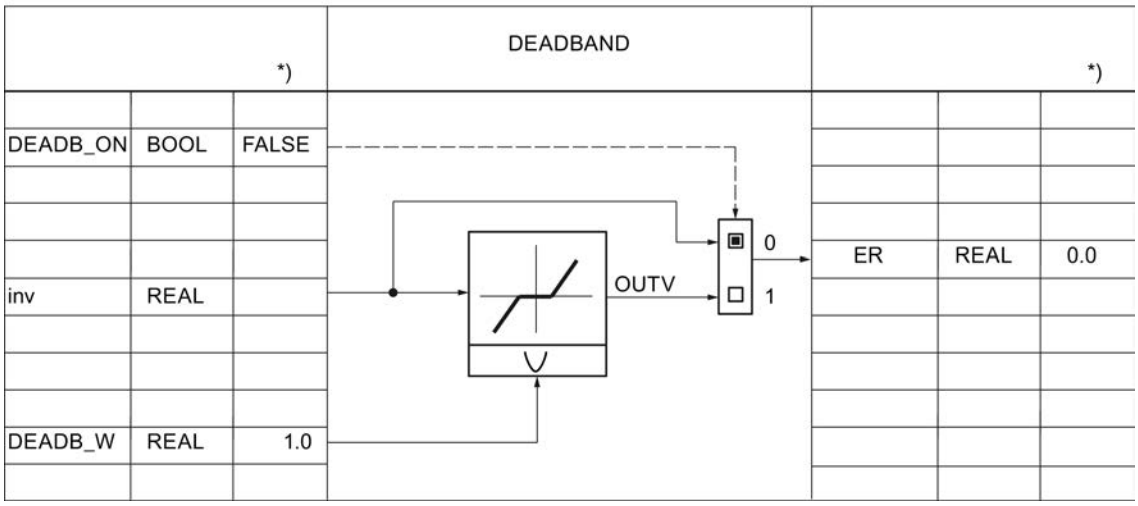
Con los parámetros siguientes se vigila siempre un límite superior e inferior absoluto de la consigna.



Calcular el error de regulación

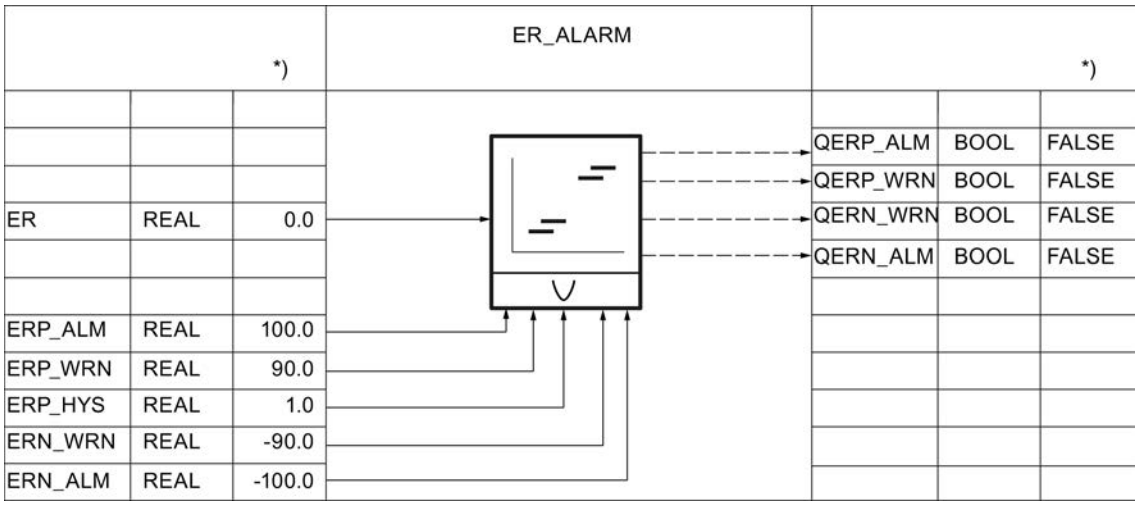
El error de regulación se conduce a través de una zona muerta con los parámetros siguientes.

Zona muerta (Página 10086)



Los límites de advertencia y alarma del error de regulación se vigilan siempre con los parámetros siguientes.

Alarma (Página 10087)



Algoritmo de regulación PID_CP

Algoritmo de regulación

Algoritmo de regulación (Página 10088)

Hay un total de cinco interruptores disponibles para definir una estructura de regulador efectiva.

Estructuras de regulador (Página 10090)

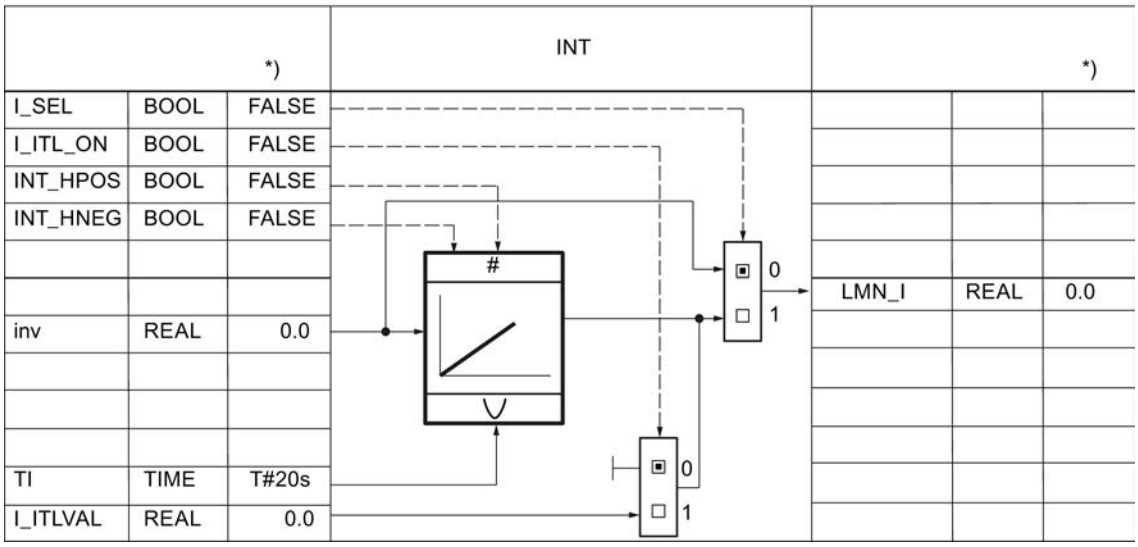
Estructura de regulador	Interruptor				
	P_SEL	I_SEL	D_SEL	PFDB_SEL	DFDB_SEL
Regulación P	TRUE	FALSE	FALSE	FALSE	FALSE
Regulación P (P en la rama de realimentación)	TRUE	FALSE	FALSE	TRUE	FALSE
Regulación PI	TRUE	TRUE	FALSE	FALSE	FALSE
Regulación PI (P en la rama de realimentación)	TRUE	TRUE	FALSE	TRUE	FALSE
Regulación PD	TRUE	FALSE	TRUE	FALSE	FALSE
Regulación PD (P en la rama de realimentación)	TRUE	FALSE	TRUE	FALSE	TRUE
Regulación PID	TRUE	TRUE	TRUE	FALSE	FALSE
Regulación PID (P/D en la rama de realimentación)	TRUE	TRUE	TRUE	FALSE	TRUE

La acción P queda determinada por la ganancia proporcional GAIN y puede observarse en LMN_P.

Acción P (Página 10093)

La acción I se calcula con los parámetros siguientes.

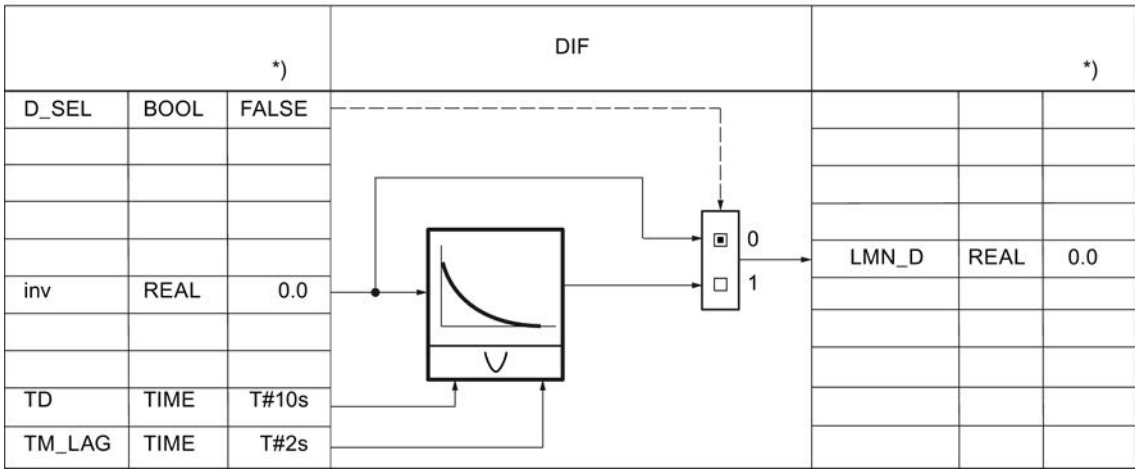
Acción I (Página 10094)



*) Ajuste predeterminado al crear el objeto tecnológico

La acción D se calcula con los parámetros siguientes.

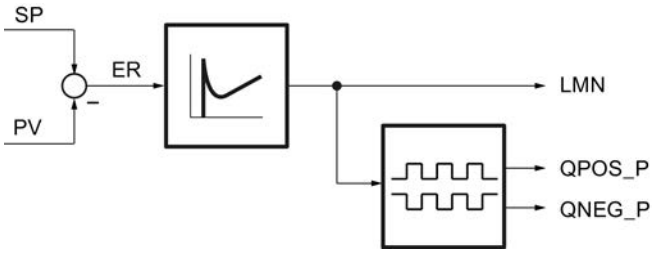
Acción D (Página 10096)



Variable manipulada PID_CP

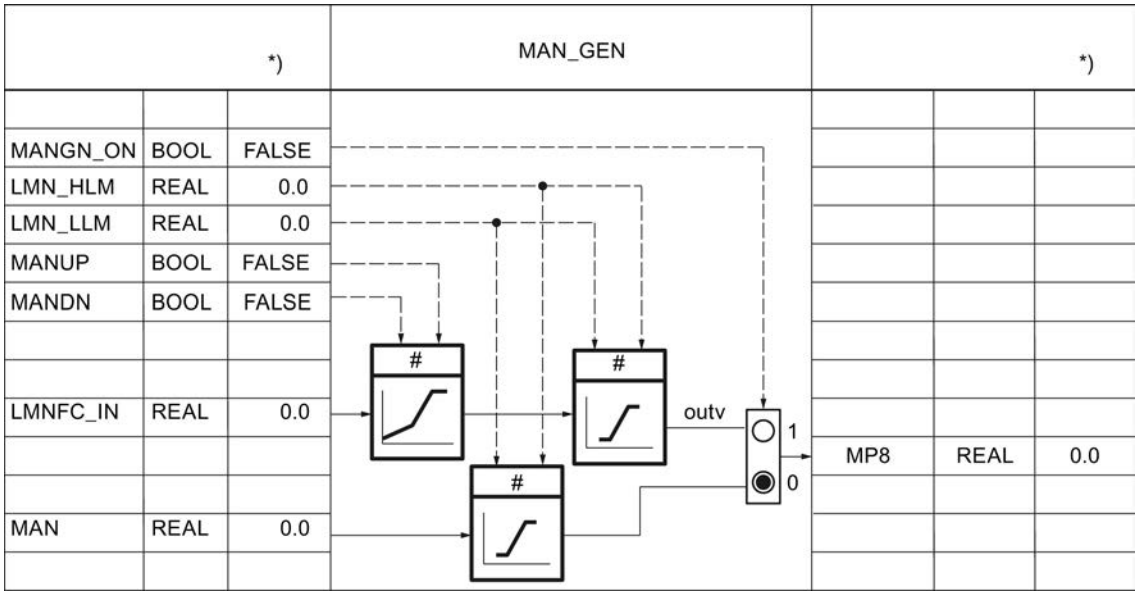
Variable manipulada continua

El algoritmo de regulación PID genera una variable manipulada analógica que, en su caso, complementa un generador de impulsos para generar señales de salida con modulación de ancho de impulsos para la regulación de dos o tres puntos con actuadores proporcionales.



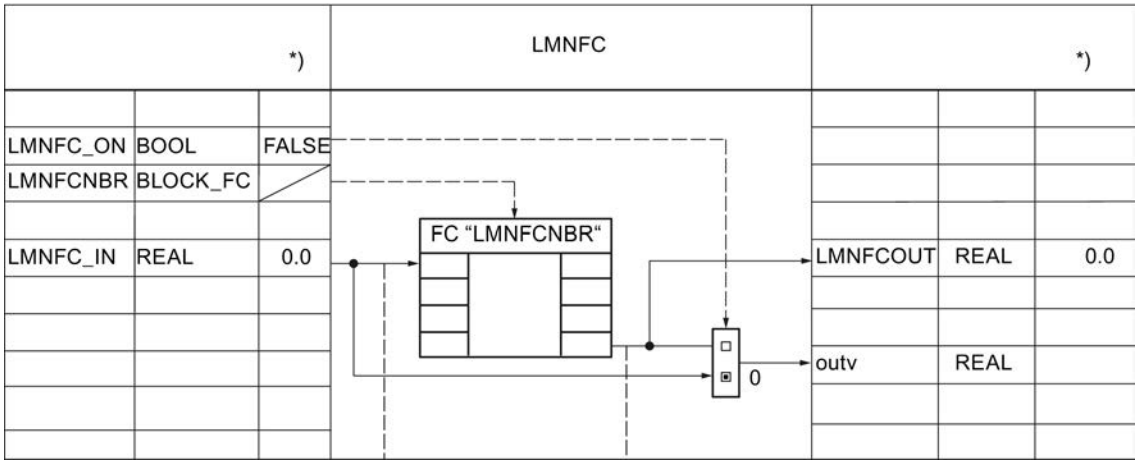
Sobre la variable manipulada puede influirse manualmente con los parámetros siguientes.

Modo manual (Página 10098)

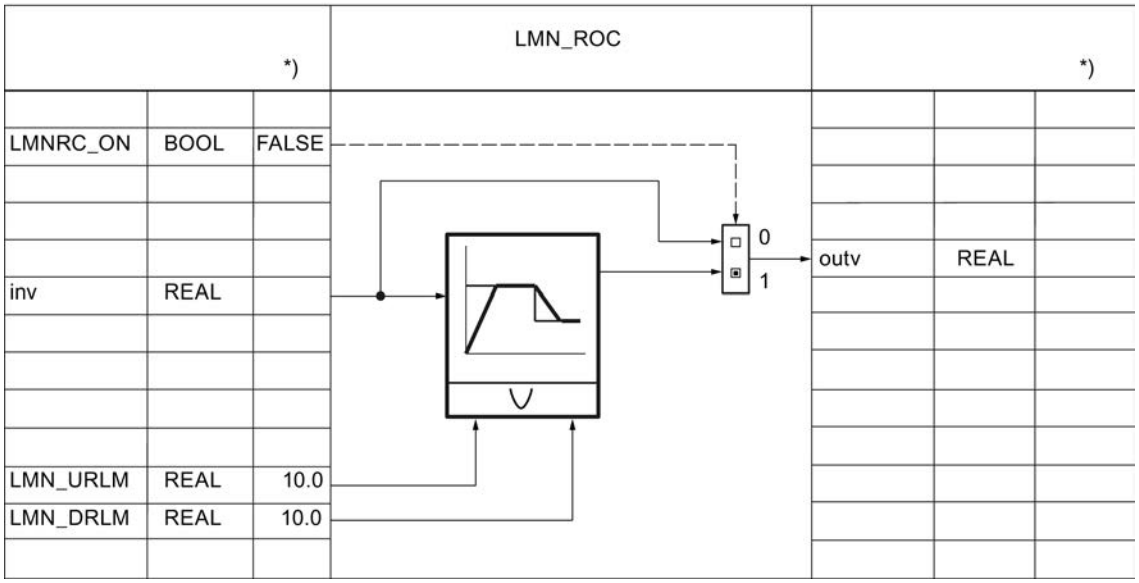


La variable manipulada puede acondicionarse con una función específica del usuario. Para ello se requieren los parámetros siguientes. La interconexión debe programarse en la FC de usuario.

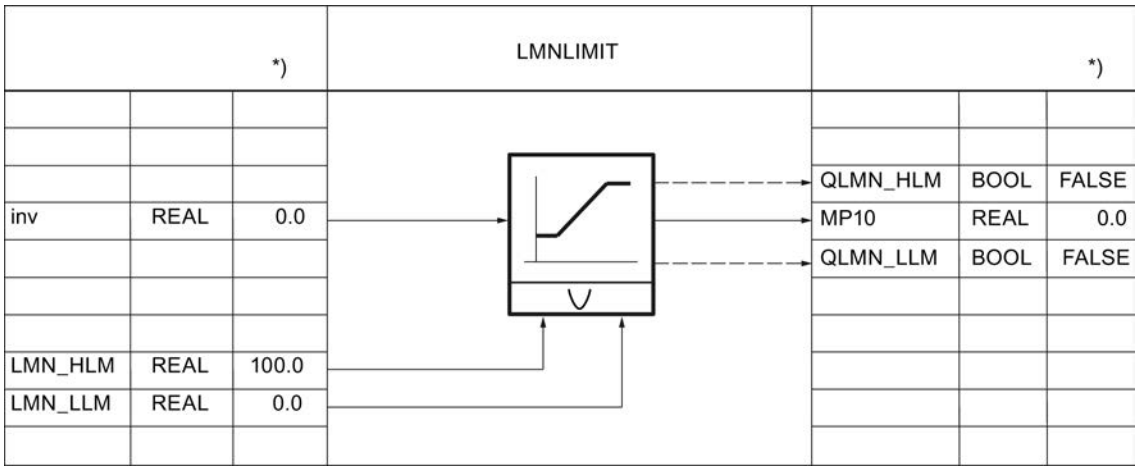
Límites (Página 10101)



La velocidad de cambio de la variable manipulada se limita con los parámetros siguientes.

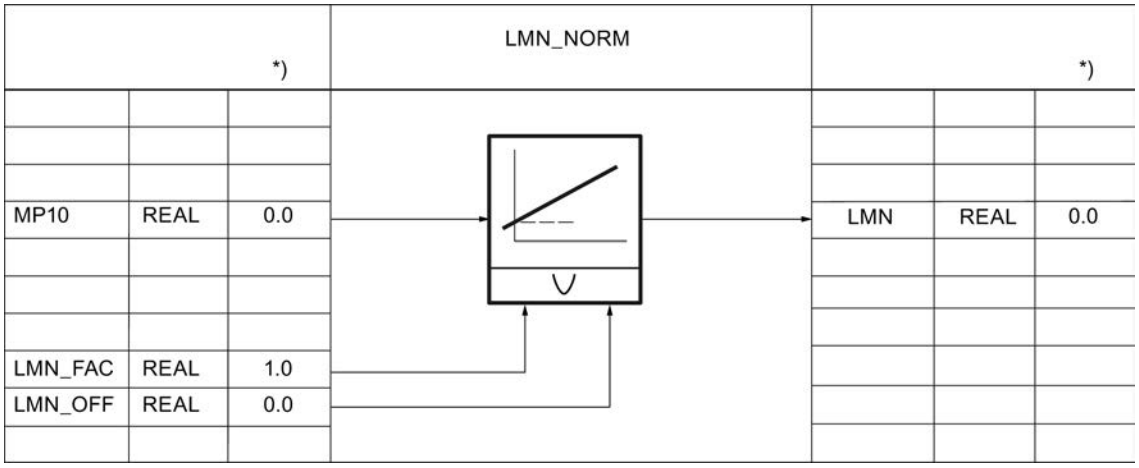


Con los parámetros siguientes se vigila siempre un límite superior e inferior absoluto de la variable manipulada.



La variable manipulada se normaliza a una magnitud física con los parámetros siguientes.

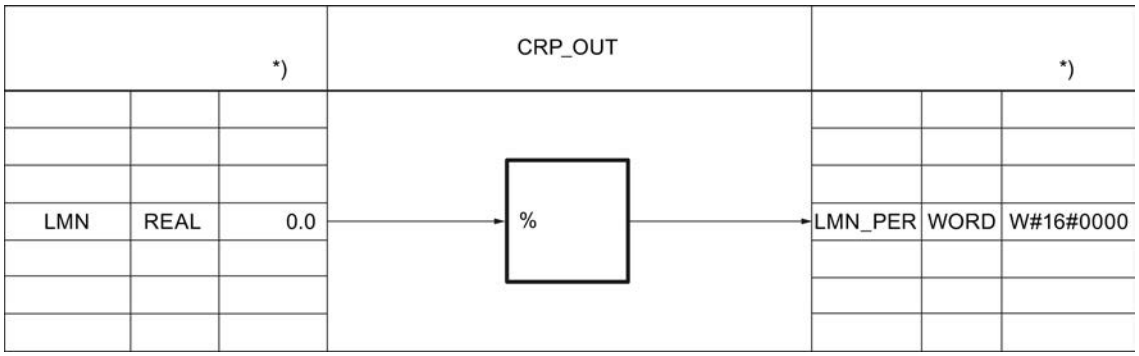
Normalización (Página 10103)



La variable manipulada en formato de coma flotante se convierte al formato de periferia. No se comprueba si se produce un desbordamiento positivo o negativo ni si se alcanza el rango de tolerancia por encima o debajo. Los tipos de módulos no se tienen en cuenta.

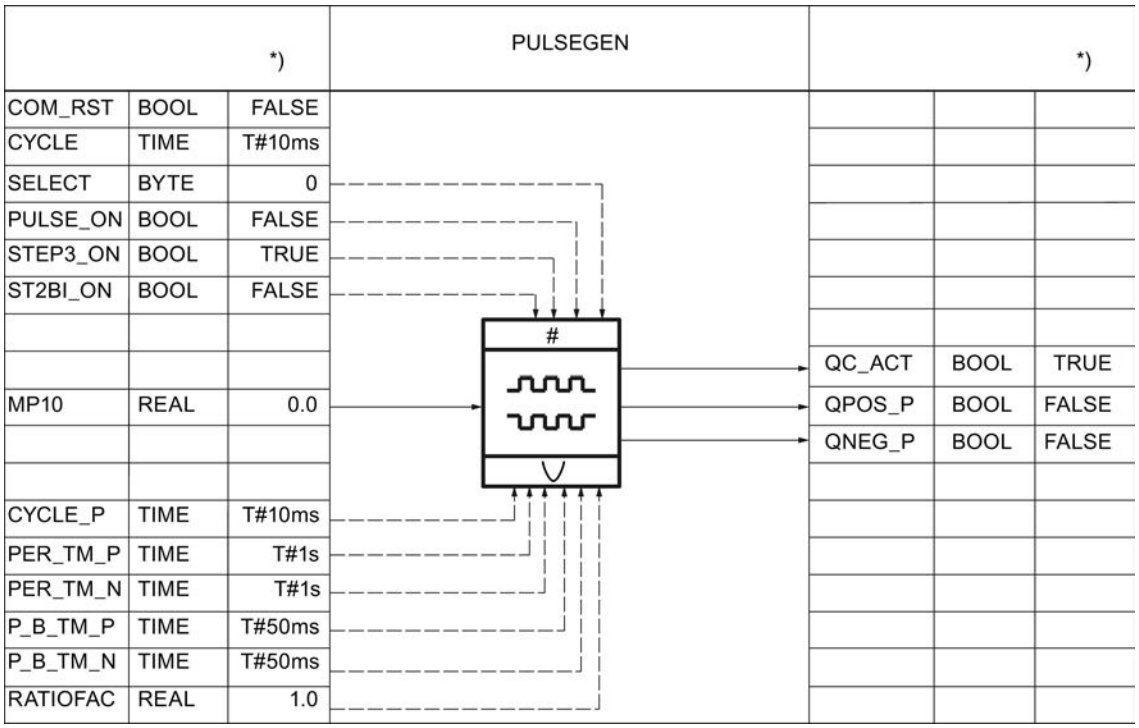
La tabla siguiente ofrece una visión de conjunto de los rangos y valores numéricos antes y después del procesamiento.

Variable manipulada LMN en %	Valor de periferia LMN_PER
118.515	32767
100.000	27648
0.003617	1
0.000	0
-0.003617	-1
-100.000	-27648
-118.519	-32768



La variable manipulada continua puede emitirse como tren de impulsos mediante una modulación de ancho de impulsos. Para ello se requieren los parámetros siguientes:

Generador de impulsos (Página 10104)



PID_CP en circuitos en cascada

Separación de la cascada del regulador

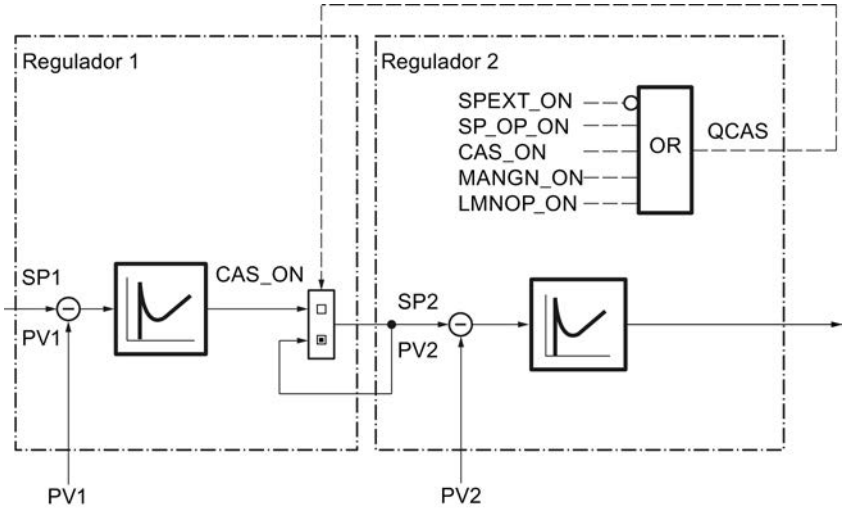
En una cascada hay varios reguladores directamente relacionados entre sí. Por ello, hay que tomar medidas para que, después de separar la estructura de la cascada, el funcionamiento en cascada pueda retomarse sin problemas en un punto cualquiera.

Por este motivo, en las regulaciones en cascada se forma una señal QCAS en los reguladores secundarios por medio de una lógica O a partir de las señales de estado de los interruptores en la rama de consigna y variable manipulada. Dicha señal acciona en los reguladores

primarios un conmutador que pasa los reguladores al modo de seguimiento. La magnitud de seguimiento es siempre la consigna del lazo secundario.

La conmutación del modo de seguimiento al modo automático se realiza con suavidad, igual que la conmutación de manual a automático.

PID_CP puede emplearse como regulador piloto en los lazos de regulación de nivel superior y como regulador secundario en los lazos de regulación secundarios.

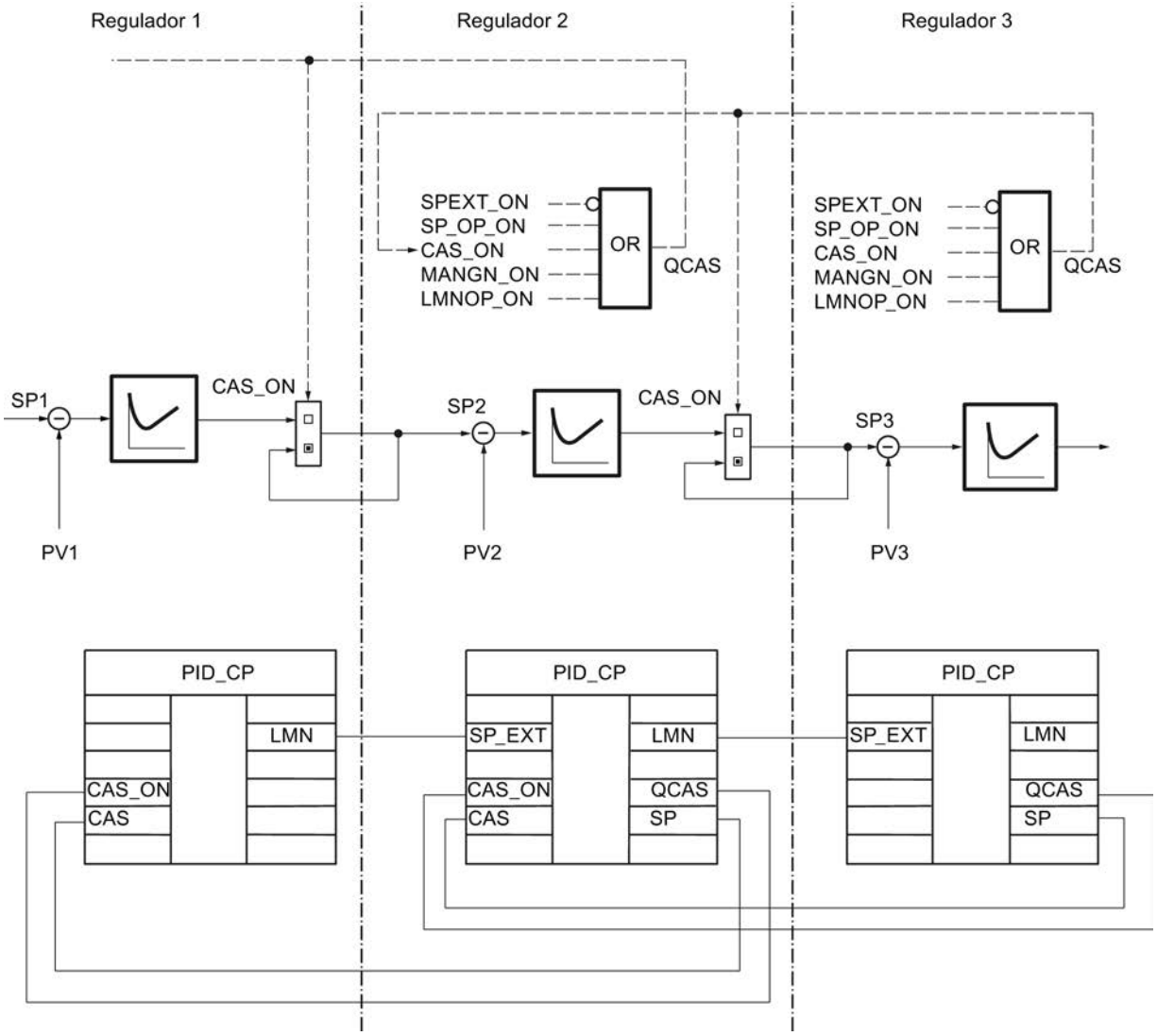


Nota

La interconexión de la variable manipulada del regulador piloto LMN debe seguir siempre la consigna externa SP_EXT del regulador secundario.

Interconexión de bloques

La figura siguiente muestra el principio de interconexión de reguladores en cascadas multilazo.



Parámetros de entrada de PID_CP

Tabla 9-204

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
COM_RST	0.0	BOOL		FALSE	Rearranque (Se procesa la rutina de inicialización de la instrucción)
I_SEL	0.1	BOOL		TRUE	Activar la acción I
D_SEL	0.2	BOOL		FALSE	Activar la acción D

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
MAN_ON	0.3	BOOL		TRUE	Activar el modo manual (Lazo de regulación deshecho, LMN ajustado manualmente)
CAS_ON	0.4	BOOL		FALSE	Activar modo de cascada (interconectado con QCAS del regulador subordinado)
SELECT	1.0	BYTE	0, 1, 2, 3	0	Si PULS_ON = TRUE: 0: PID y generador de impulsos 1: PID (llamada del bloque en OB 1) 2: Generador de impulsos (llamada del bloque en un OB de alarma cíclica) 3: PID (llamada de bloque en un OB de alarma cíclica)
CYCLE	2.0	TIME	> 20 ms (S7–300)	T#1s	Tiempo de muestreo (tiempo entre dos llamadas del bloque = constante) Es esencial asegurarse de que el parámetro se parametriza con el tiempo de alarma cíclica del OB en el que está activa la instrucción PID_CP. En caso contrario las funciones dependientes del tiempo no funcionan correctamente. (Excepción: se utiliza un reductor de tiempo, por ejemplo a través del distribuidor de llamadas del regulador.)
CYCLE_P	6.0	TIME		T#10 ms	Tiempo de muestreo del generador de impulsos Es esencial asegurarse de que el parámetro se parametriza con el tiempo de alarma cíclica del OB en el que está activa la instrucción PID_CP. En caso contrario las funciones dependientes del tiempo no funcionan correctamente. (Excepción: se utiliza un reductor de tiempo, por ejemplo a través del distribuidor de llamadas del regulador.)
SP_INT	10.0	REAL	Rango técnico (magnitud física)	0.0	Consigna interna SP_INT puede modificarse con SPUP y SPDN mediante un HMI.
SP_EXT	14.0	REAL	Rango técnico (magnitud física)	0.0	consigna externa SP_EXT se utiliza para reguladores de mezcla, relación y cascada y puede normalizarse.
PV_IN	18.0	REAL	Rango técnico (magnitud física)	0.0	Entrada de valor real (PV en formato de coma flotante)
PV_PER	22.0	INT		W#16#0000	Valor real de periferia (PV en formato de periferia)
GAIN	24.0	REAL	Rango completo (sin dimensión)	2.0	Ganancia proporcional (= ganancia del regulador)
TI	28.0	TIME	TI ≥ CYCLE	T#20s	Tiempo integrativo
TD	32.0	TIME	TD ≥ CYCLE	T#10s	Tiempo derivativo

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
TM_LAG	36.0	TIME	TM_LAG ≥ CYCLE/2	T#2s	Tiempo de retardo de la acción D
DISV	40.0	REAL		0.0	Perturbación Rango admisible: -100.0 ... 100.0
CAS	44.0	REAL	Rango técnico (magnitud física)	0.0	Entrada para el modo de cascada (interconexión con PV del regulador subordinado)
SP_HLM	48.0	REAL	Rango técnico (magnitud física)	100.0	Consigna del límite superior
SP_LLM	52.0	REAL	Rango técnico (magnitud física)	0.0	Consigna del límite inferior
LMN_HLM	56.0	REAL	LMN_LLM ... 100.0	100.0	Variable manipulada: límite superior
LMN_LLM	60.0	REAL	-100.0 ... LMN_HLM	0.0	Variable manipulada: límite inferior
DB_NBR	64.0	BLOCK_DB		DB 1	Número del bloque de datos (DB con nodos de la curva)
SPFC_NBR	66.0	BLOCK_FC		FC 0	Consigna número de FC (FC autodefinida en la rama de consigna)
PVFC_NBR	68.0	BLOCK_FC		FC 0	Valor real número de FC (FC autodefinida en la rama del valor real)
LMNFCNBR	70.0	BLOCK_FC		FC 0	Variable manipulada número de FC (FC definido por el usuario en la rama de la variable manipulada)

Parámetros de salida de PID_CP

Tabla 9-205

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
LMN	72.0	REAL	0.0	Variable manipulada (variable manipulada en formato de coma flotante)
LMN_PER	76.0	INT	W#16#0000	Variable manipulada para periferia (LMN en el formato de la periferia)
SP	78.0	REAL	0.0	Consigna (consigna con efectos reales)
PV	82.0	REAL	0.0	Valor real (emisión del valor real efectivo en el modo de cascada)

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QCAS	86.0	BOOL	FALSE	Señal para modo de cascada (Se interconecta con CAS_ON del regulador de nivel superior)
QC_ACT	86.1	BOOL	TRUE	Indicación de si se procesa la parte de regulación en la siguiente llamada del bloque (sólo relevante si SELECT tiene el valor 0 ó 1)
QPOS_P	86.2	BOOL	FALSE	Generador de impulsos: activar impulso positivo
QNEG_P	86.3	BOOL	FALSE	Generador de impulsos: activar impulso negativo

Parámetros de entrada/salida de PID_CP

Tabla 9-206

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
MAN	88.0	REAL	0.0	variable manipulada manual (para especificar la variable manipulada mediante funciones de manejo y visualización)

Variables estáticas de PID_CP

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
PVH_ALM	92.0	REAL	PVH_WRN...100.0	100.0	Valor real: límite superior de 'Alarma'
PVH_WRN	96.0	REAL	PVL_WRN... PVH_ALM	90.0	Valor real: límite superior de 'Advertencia'
PVL_WRN	100.0	REAL	PVL_ALM... PVH_WRN	-90.0	Valor real: límite inferior de 'Advertencia'
PVL_ALM	104.0	REAL	-100.0...PVL_WRN	-100.0	Valor real: límite inferior de 'Alarma'
SPGEN_ON	108.0	BOOL		FALSE	Activar control manual de la consigna (Ajustar la consigna a través de interruptores Arriba/Abajo)
SPUP	108.1	BOOL		FALSE	Consigna alta
SPDN	108.2	BOOL		FALSE	Consigna baja
RMPSK_ON	108.3	BOOL		FALSE	Activar el generador ramp soak (La consigna se predetermina como curva)
SPEXT_ON	108.4	BOOL		FALSE	Activar consigna externa (Para la interconexión con otros bloques de regulador)
MANGN_ON	108.5	BOOL		FALSE	Activar el modo manual ((LMN ajustado con interruptor)
MANUP	108.6	BOOL		FALSE	Variable manipulada manual alta
MANDN	108.7	BOOL		FALSE	Variable manipulada manual baja

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
DFRMP_ON	109.0	BOOL		FALSE	Preasignar la salida del generador ramp soak (A la salida se le preasigna SP_INT)
CYC_ON	109.1	BOOL		FALSE	Activar repetición cíclica (La curva se repite automáticamente)
RMP_HOLD	109.2	BOOL		FALSE	Detener generador ramp soak (Se congela la salida del generador ramp soak)
CONT_ON	109.3	BOOL		FALSE	(Generador ramp soak) continuar (Se continúa la curva en el siguiente nodo)
TUPDT_ON	109.4	BOOL		FALSE	Recalcular tiempo total (Se recalcula el tiempo total de la curva)
SPFC_ON	109.5	BOOL		FALSE	Llamar FC de consigna
SPROC_ON	109.6	BOOL		FALSE	Activar función de rampa (Limitación de la velocidad de cambio de SP)
PVPER_ON	109.7	BOOL		FALSE	Activar el valor real de periferia (Interconexión con los módulos de periferia)
LAG1STON	110.0	BOOL		FALSE	Activar elemento de retardo de primer orden
SQRT_ON	110.1	BOOL		FALSE	Activar función de raíz cuadrada
PVFC_ON	110.2	BOOL		FALSE	Llamar FC de valor real
DEADB_ON	110.3	BOOL		FALSE	Activar zona muerta (Se filtran pequeñas perturbaciones y ruidos)
P_SEL	110.4	BOOL		TRUE	Activar la acción P
PFDB_SEL	110.5	BOOL		FALSE	Conectar la acción P en la rama de realimentación
INT_HPOS	110.6	BOOL		FALSE	Congelar la acción I en dirección positiva
INT_HNEG	110.7	BOOL		FALSE	Congelar la acción I en dirección negativa
I_ITL_ON	111.0	BOOL		FALSE	Preasignar la acción I
DFDB_SEL	111.1	BOOL		FALSE	Aplicar la acción D en la rama de realimentación
DISV_SEL	111.2	BOOL		FALSE	Control anticipativo
LMNFC_ON	111.3	BOOL		FALSE	Llamar FC de variable manipulada
LMNRC_ON	111.4	BOOL		FALSE	Activar la función de rampa de la variable manipulada (Limitación de la velocidad de cambio de LMN)
SMOO_CHG	111.5	BOOL		TRUE	Conmutación suave de manual a automático
PULSE_ON	111.6	BOOL		FALSE	Activar el generador de impulsos
STEP3_ON	111.7	BOOL		TRUE	Generador de impulsos: activar la regulación de tres puntos
ST2BI_ON	112.0	BOOL		FALSE	Generador de impulsos: Activar la regulación de dos puntos para el margen de corrección bipolar (Para el rango unipolar es necesario que STEP3_ON = FALSE)
TM_SNBR	114.0	INT	≥ 0 (sin dimensión)	0	Número del próximo nodo para continuar

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
TM_CONT	116.0	TIME	Rango completo (sin dimensión)	T#0s	Momento de continuación (Tiempo después del nodo TM_SNBR en el que el generador ramp soak sigue procesando la curva)
FAC	120.0	REAL	Rango completo (sin dimensión)	1.0	Factor (factor de relación o mezcla)
NM_SPEHR	124.0	REAL		100.0	Normalización de la consigna: entrada del rango de trabajo arriba
NM_SPELR	128.0	REAL		-100.0	Normalización de la consigna: entrada del rango de trabajo abajo
SPFC_OUT	132.0	REAL	-100.0 ... 100.0	0.0	Consigna de la salida de FC (Se interconecta con la salida de la FC en la rama de consigna)
SPURLM_P	136.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de rampa de subida de la consigna en el rango positivo
SPDRLM_P	140.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de rampa de bajada de la consigna en el rango positivo
SPURLM_N	144.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de subida de la consigna en el rango negativo
SPDRLM_N	148.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de bajada de la consigna en el rango negativo
NM_PIHR	152.0	REAL		100.0	Normalización del valor real: entrada del rango de medición arriba
NM_PILR	156.0	REAL		-100.0	Normalización del valor real: entrada del rango de medición abajo
NM_PVHR	160.0	REAL		100.0	Normalización del valor real: salida del rango de medición arriba
NM_PVLR	164.0	REAL		-100.0	Normalización del valor real: salida del rango de medición abajo
PV_TMLAG	168.0	TIME	Rango completo	T#5s	Tiempo de retardo del valor real (Retardo del elemento PT1 en la rama PV)
SQRT_HR	172.0	REAL		100.0	Raíz cuadrada: salida del rango de trabajo arriba
SQRT_LR	176.0	REAL		0.0	Raíz cuadrada: salida del rango de trabajo abajo
PVFC_OUT	180.0	REAL	-100.0 ... 100.0	0.0	Valor real de la salida de FC (Se interconecta con la salida de la FC en la rama del valor real)
PVURLM_P	184.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de subida del valor real en el rango positivo
PVDRLM_P	188.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de bajada del valor real en el rango positivo
PVURLM_N	192.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de subida del valor real en el rango negativo
PVDRLM_N	196.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de bajada del valor real en el rango negativo

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
PV_HYS	200.0	REAL	≥ 0	1.0	Histéresis del valor real (Evita el "centelleo" de la vista de monitorización)
DEADB_W	204.0	REAL	0.0 ... 100.0	1.0	Ancho de zona muerta (= rango cero hasta el límite superior de la zona muerta) (determina el rango de la zona muerta)
ERP_ALM	208.0	REAL	0 ... 200.0	100.0	Error de regulación: límite positivo de 'Alarma'
ERP_WRN	212.0	REAL	0 ... 200.0	90.0	Error de regulación: límite positivo de 'Advertencia'
ERN_WRN	216.0	REAL	-200.0 ... 0	-90.0	Error de regulación: límite negativo de 'Advertencia'
ERN_ALM	220.0	REAL	-200.0 ... 0	-100.0	Error de regulación: límite negativo de 'Alarma'
ER_HYS	224.0	REAL	≥ 0 [%]	1.0	Error de regulación de histéresis (Evita el "centelleo" de la vista de monitorización)
I_ITLVAL	228.0	REAL	-100.0 ... 100.0 [%]	0.0	Valor de inicialización para la acción I
LMNFCOUT	232.0	REAL	-100.0 ... 100.0 [%]	0.0	Variable manipulada salida del FC (se conecta con la salida del FC en la rama de la variable manipulada)
LMN_URLM	236.0	REAL	≥ 0 [%/s]	10.0	Límite de subida de la variable manipulada
LMN_DRLM	240.0	REAL	≥ 0 [%/s]	10.0	Límite de bajada de la variable manipulada
LMN_FAC	244.0	REAL	Rango completo (sin dimensión)	1.0	Factor de la variable manipulada (Factor para la adaptación del rango de la variable manipulada)
LMN_OFF	248.0	REAL	Rango completo (sin dimensión)	0.0	Offset de la variable manipulada (Punto cero de la normalización de la variable manipulada)
PER_TM_P	252.0	TIME		T#1s	Generador de impulsos: duración de período del impulso positivo
PER_TM_N	256.0	TIME		T#1s	Generador de impulsos: duración de período del impulso negativo
P_B_TM_P	260.0	TIME		T#50ms	Generador de impulsos: duración mínima del impulso o duración mínima de pausa del impulso positivo
P_B_TM_N	264.0	TIME		T#50ms	Generador de impulsos: duración mínima del impulso o duración mínima de pausa del impulso negativo
RATIOFAC	268.0	REAL	0.1 ... 10.0 (sin dimensión)	1.0	Generador de impulsos: factor de proporción (proporción de la duración del impulso positivo y la duración del impulso negativo)
PHASE	272.0	INT		0	Fase del PID Self-Tuner
QPVH_ALM	274.0	BOOL		FALSE	Valor real: el límite superior de 'Alarma' ha reaccionado
QPVH_WRN	274.1	BOOL		FALSE	Valor real: el límite superior de 'Advertencia' ha reaccionado

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
QPVL_WRN	274.2	BOOL		FALSE	Valor real: el límite inferior de 'Advertencia' ha reaccionado
QPVL_ALM	274.3	BOOL		FALSE	Valor real: el límite inferior de 'Alarma' ha reaccionado
QR_S_ACT	274.4	BOOL		FALSE	Procesando tabla de tiempo para curva
QSP_HLM	274.5	BOOL		FALSE	Consigna: el límite superior ha reaccionado
QSP_LLM	274.6	BOOL		FALSE	Consigna: el límite inferior ha reaccionado
QPVURLMP	274.7	BOOL		FALSE	Valor real: el límite de la rampa de subida en el rango positivo ha reaccionado
QPVDRLMP	275.0	BOOL		FALSE	Valor real: el límite de la rampa de bajada en el rango positivo ha reaccionado
QPVURLMN	275.1	BOOL		FALSE	Valor real: el límite de la rampa de subida en el rango negativo ha reaccionado
QPVDRLMN	275.2	BOOL		FALSE	Valor real: el límite de la rampa de bajada en el rango negativo ha reaccionado
QERP_ALM	275.3	BOOL		FALSE	Error de regulación: el límite positivo de 'Alarma' ha reaccionado
QERP_WRN	275.4	BOOL		FALSE	Error de regulación: el límite positivo de 'Advertencia' ha reaccionado
QERN_WRN	275.5	BOOL		FALSE	Error de regulación; el límite negativo de 'Advertencia' ha reaccionado
QERN_ALM	275.6	BOOL		FALSE	Error de regulación: el límite negativo de 'Alarma' ha reaccionado
QLMN_HLM	275.7	BOOL		FALSE	Variable manipulada: el límite superior ha reaccionado
QLMN_LLM	276.0	BOOL		FALSE	Variable manipulada: el límite inferior ha reaccionado
NBR_ATMS	278.0	INT		0	Número del nodo actual del generador ramp soak
RS_TM	280.0	TIME		T#0s	"Tiempo residual actual" de la curva hasta el siguiente nodo
T_TM	284.0	TIME		T#0s	"Tiempo total" de la curva
RT_TM	288.0	TIME		T#0s	"Tiempo residual total" = tiempo hasta el final de la curva
ER	292.0	REAL		0.0	Error de regulación
LMN_P	296.0	REAL		0.0	Acción P
LMN_I	300.0	REAL		0.0	Acción I
LMN_D	304.0	REAL		0.0	Acción D
SPFC_IN	308.0	REAL		0.0	Consigna entrada de FC (Se interconecta con la entrada de la FC autodefinida)
PVFC_IN	312.0	REAL		0.0	Valor real entrada de FC (Se interconecta con la entrada de la FC autodefinida)

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
LMNFC_IN	316.0	REAL		0.0	Variable manipulada de la entrada de FC (se interconecta con la entrada del FC autodefinido)
SP_OP_ON	320.0	BOOL		FALSE	Activar control de la consigna (El valor de SP_OP se aplica como consigna)
PV_OP_ON	320.1	BOOL		FALSE	Activar control manual del valor real (El valor de PV_OP se aplica como consigna)
LMNOP_ON	320.2	BOOL		FALSE	Activar control manual de la variable manipulada (El valor de LMN_OP se aplica como consigna)
SP_OP	322.0	REAL		0.0	Control de consigna en la puesta en marcha
PV_OP	326.0	REAL		0.0	Control manual del valor real en la puesta en marcha
LMN_OP	330.0	REAL		0.0	Control de variable manipulada en la puesta en marcha
MP1	334.0	REAL		0.0	Punto de medición 1: consigna interna
MP2	338.0	REAL		0.0	Punto de medición 2: consigna externa
MP3	342.0	REAL		0.0	Punto de medición 3: consigna ilimitada
MP4	346.0	REAL		0.0	Punto de medición 4: valor real del módulo de periferia
MP5	350.0	REAL		0.0	Punto de medición 5: valor real después del elemento de retardo de primer orden
MP6	354.0	REAL		0.0	Punto de medición 6: valor real efectivo (PV)
MP7	358.0	REAL		0.0	Punto de medición 7: variable manipulada del algoritmo PID
MP8	362.0	REAL		0.0	Punto de medición 8: variable manipulada manual
MP9	366.0	REAL		0.0	Punto de medición 9: variable manipulada ilimitada
MP10	370.0	REAL		0.0	Punto de medición 10: variable manipulada limitado

Diagramas de bloques de PID_CP

Diagramas del flujo de señales

De forma análoga a la representación de los interruptores en la herramienta de configuración, el punto negro en los símbolos de los interruptores significa que la señal de conmutación tiene el valor booleano adyacente (0 = FALSE o 1 = TRUE) y que el recorrido de la señal está aplicado a este punto en cada caso. Las señales de conmutación (señales binarias) están marcadas con líneas discontinuas.

Esto significa que la consigna se especifica de modo absoluto mediante SP_INT; lo mismo es válido para la entrada del valor real mediante PV_IN. Como función de regulación se ha ajustado un regulador PI normal con función P en la rama de avance. El lazo de regulación es abierto y se influye sobre la magnitud manipulada mediante la entrada MAN en el área

porcentual. Todas las demás funciones son pasivas o, si no pueden desactivarse, no son efectivas debido al ajuste predeterminado de parámetros "marginales" en la evolución de las señales dentro del rango de medida o del área de trabajo.

Símbolos e identificaciones en los diagramas de flujo de señales

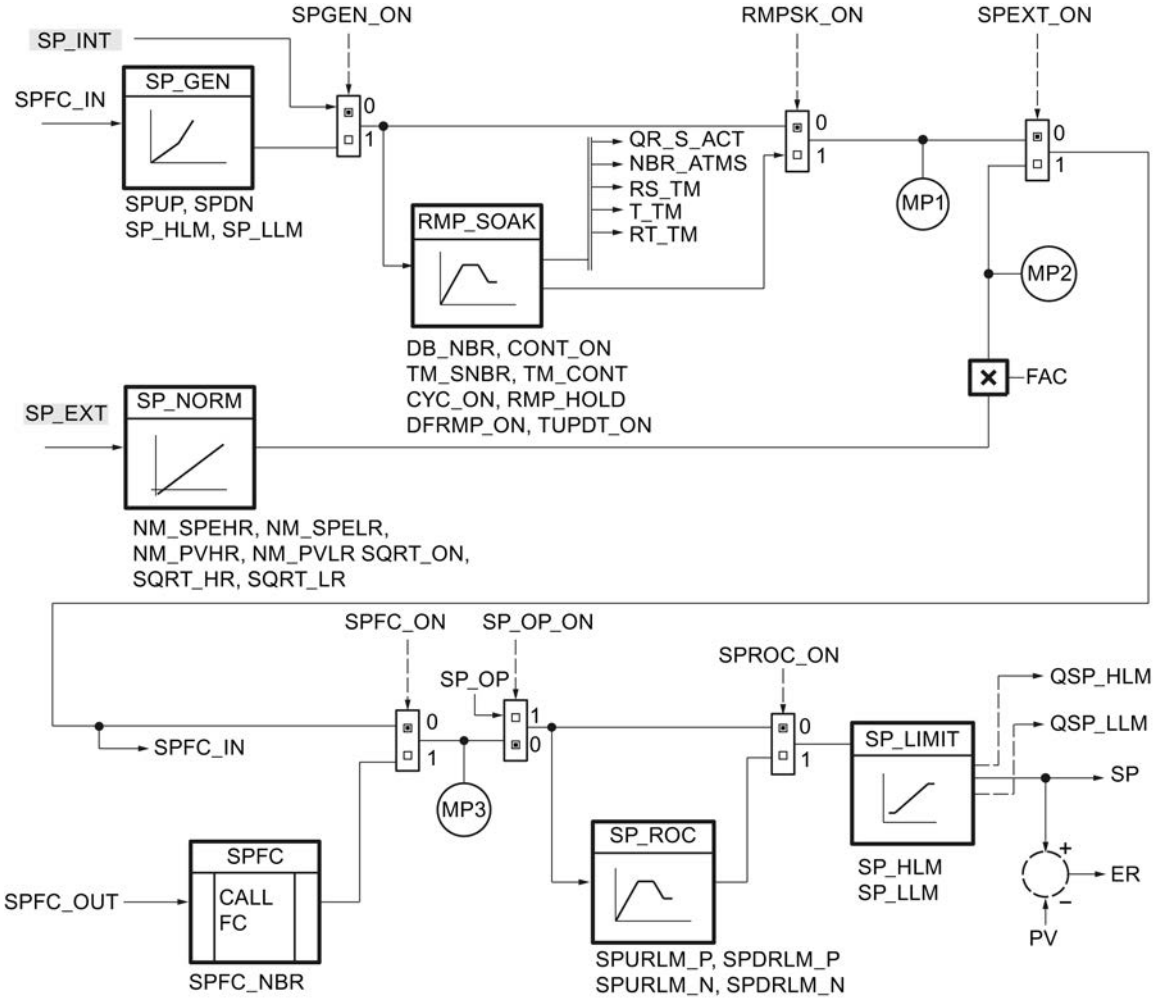
Las denominaciones de las magnitudes del proceso que pueden conectarse se representan con el fondo marcado. Gracias a ellas se reconoce dónde puede conectarse la estructura del regulador con la periferia S7 o directamente con órganos de medición y los actuadores del proceso.

Las denominaciones de parámetros con la combinación de letras "OP" (p. ej. SP_OP/ SP_OP_ON) hacen referencia a la posible intervención en este punto mediante la herramienta de configuración de Standard PID Control. La herramienta de configuración tiene una interfaz propia con el FB de regulación.

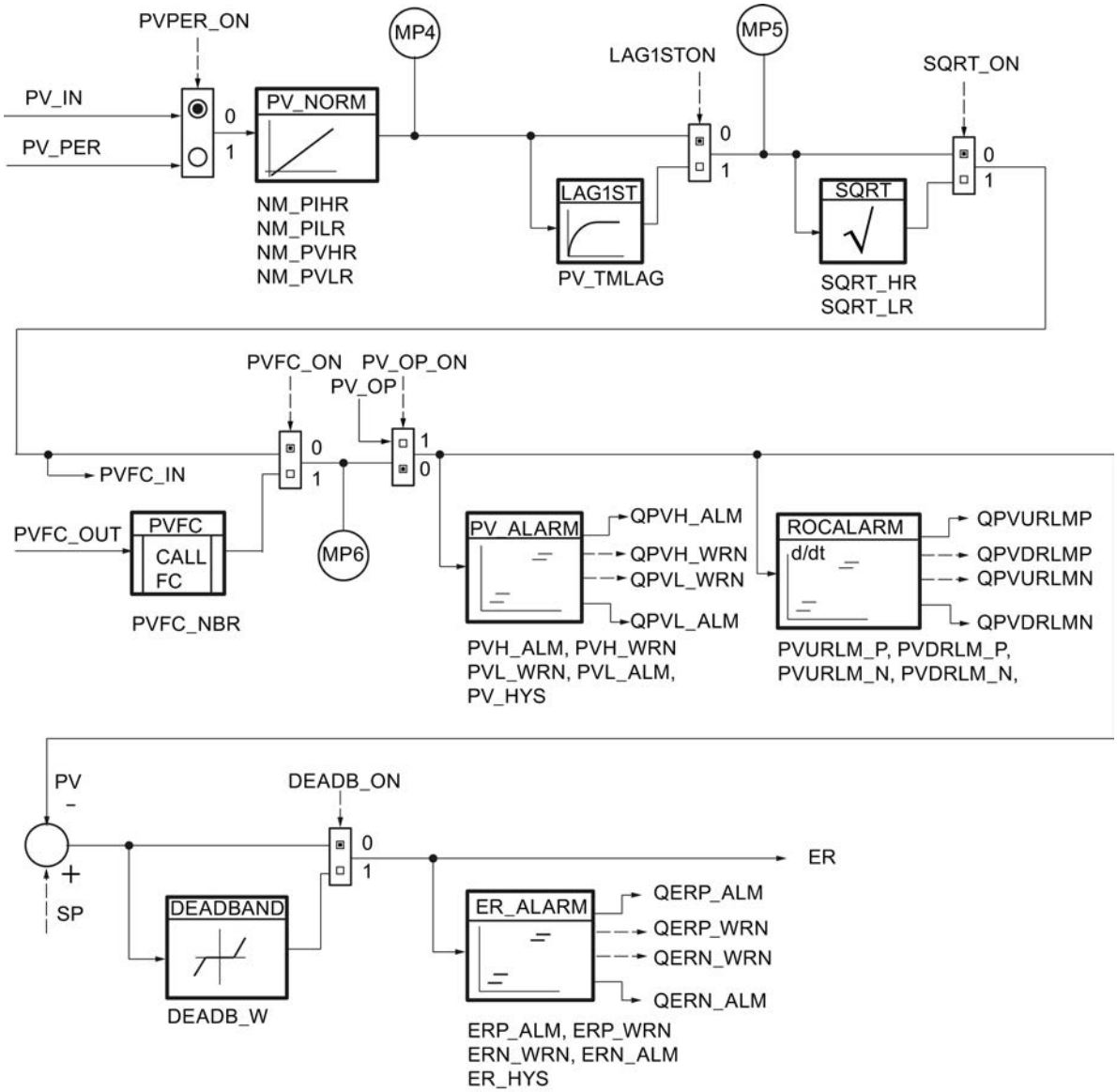
Las magnitudes intermedias en la evolución de las señales pueden observarse mediante los puntos de medición MP1 ... MP12 marcados con círculos. Dichas magnitudes intermedias son necesarias para sincronizar valores antes de lanzar operaciones de conmutación "suaves" o para inspeccionar los estados actuales de la regulación en cuestión. Las magnitudes del punto de medición en el registrador de curvas de la herramienta de configuración pueden representarse tanto estática como dinámicamente.

Por motivos de claridad, los parámetros para ajustar y dimensionar las operaciones de procesamiento (algoritmos) están indicados en los distintos bloques de función. Remitimos al respecto a las descripciones de la parte de referencia y a la representación de las diferentes funciones parciales en los apartados siguientes.

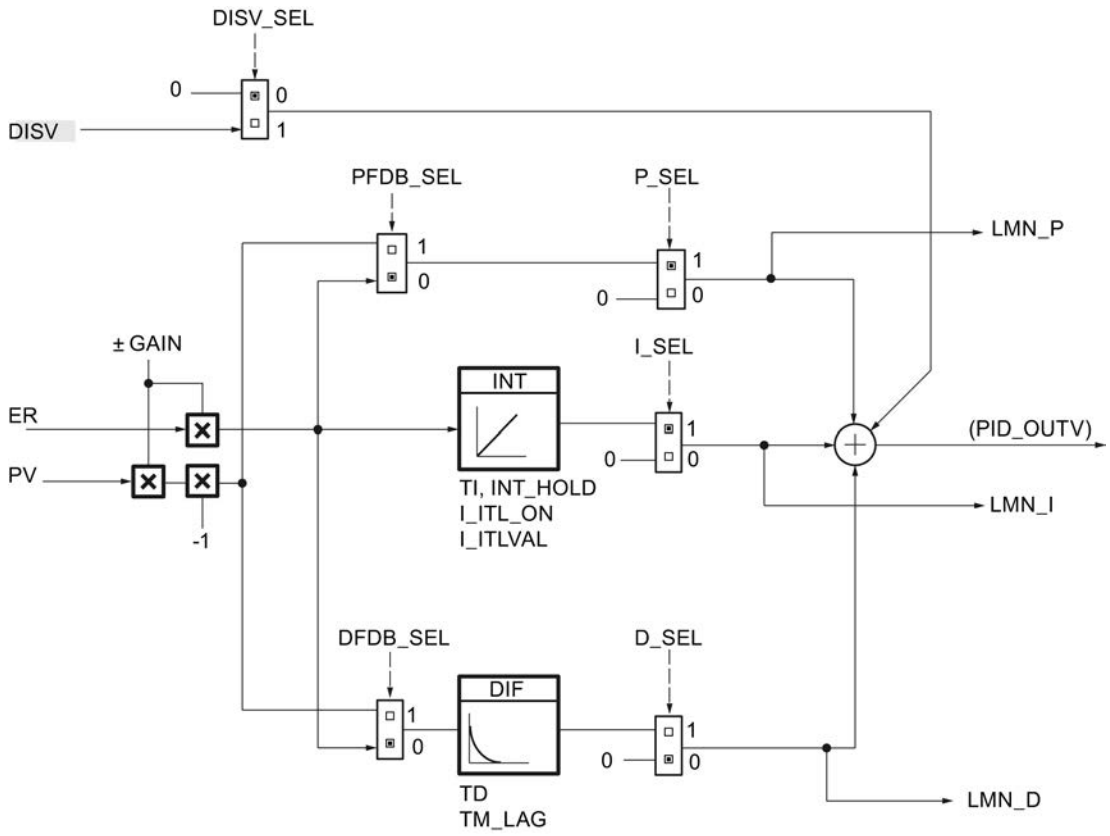
Acondicionamiento de la consigna



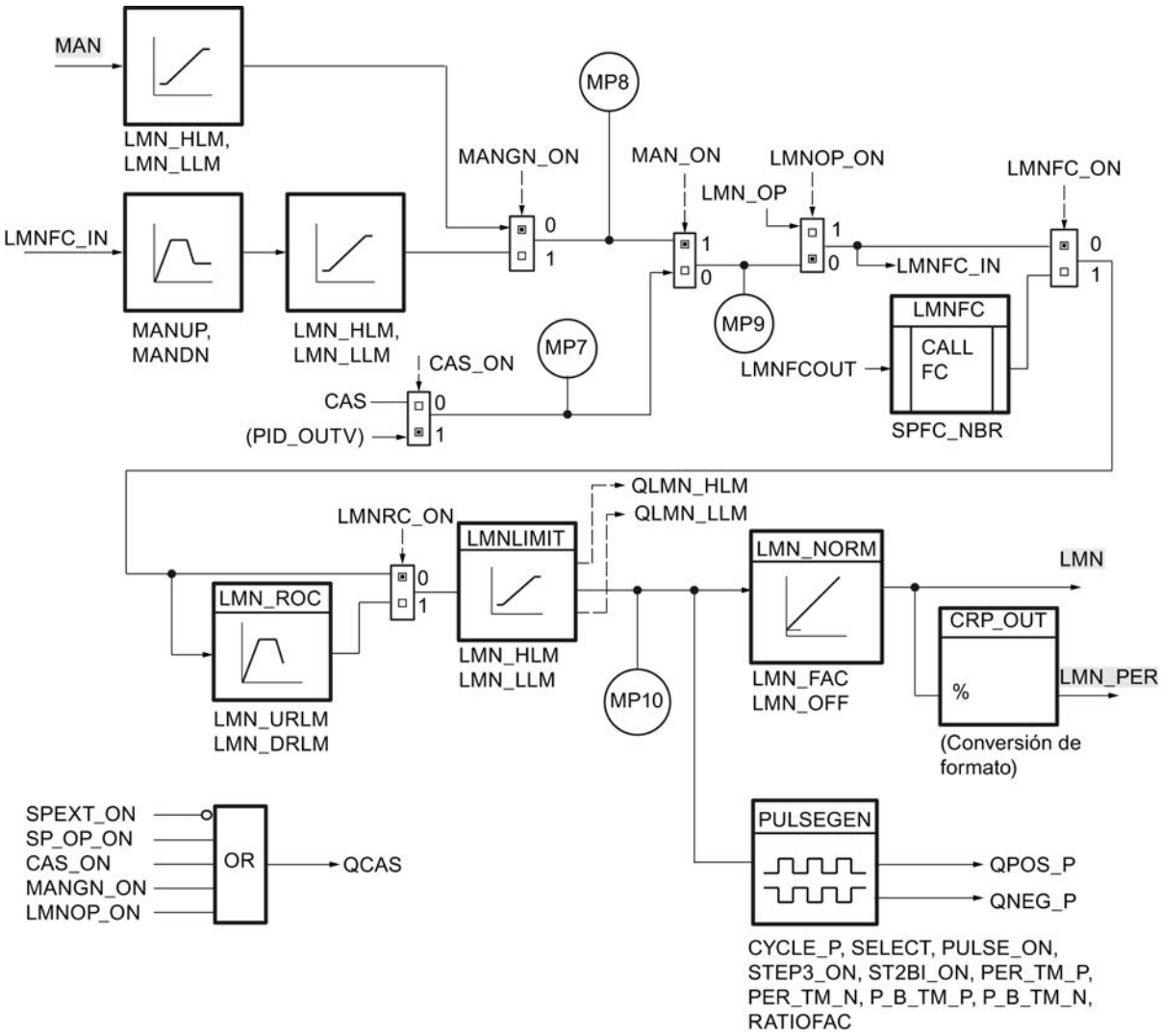
Acondicionamiento del valor real



Algoritmo de regulación



Procesamiento de la variable manipulada analógica



La puerta OR genera una señal de habilitación para el acoplamiento en cascada según sea la combinación de los estados de conmutación.

Consulte también

Generador ramp soak (RMP_SOAK) (Página 10075)

Bloque de datos RMP_SOAK

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
NBR_PTS	INT	4	Número de nodos -1 1 - 255
PI	Array [0...NBR_PTS] of Struct		Estructura de los datos de los nodos. La matriz debe ser declarada por 0...NBR_PTS.
PI[0].OUTV	REAL	0.0	Valor de salida [0] punto inicial Rango técnico
PI[0].TMV	TIME	T#1s	Valor temporal [0] punto inicial
PI[1].OUTV	REAL	0.0	Valor de salida [1] nodo 1 Rango técnico
PI[1].TMV	TIME	T#1s	Valor temporal [1] nodo 1
PI[2].OUTV	REAL	0.0	Valor de salida [2] nodo 2 Rango técnico
PI[2].TMV	TIME	T#1s	Valor temporal [2] nodo 2
PI[3].OUTV	REAL	0.0	Valor de salida [3] nodo 3 Rango técnico
PI[3].TMV	TIME	T#1s	Valor temporal [3] nodo 3
PI[4].OUTV	REAL	0.0	Valor de salida [4] nodo 4 Rango técnico
PI[4].TMV	TIME	T#0s	Valor temporal [4] nodo 4 0 ms

PID_ES

Descripción de PID_ES

Además de las funciones de las ramas de valor real y de consigna, la instrucción implementa un regulador PID terminado con salida binaria de variable manipulada. La variable manipulada puede modificarse manualmente. Las funciones parciales pueden activarse o desactivarse.

La instrucción permite regular instalaciones y procesos técnicos con actuadores de acción integrativa en los sistemas de automatización SIMATIC S7. Se pueden regular tanto sistemas lentos (temperaturas, niveles, etc.) como sistemas muy rápidos (caudales, revoluciones, etc.).

El regulador puede utilizarse como regulador de valor fijo, ya sea individualmente o en lazos de regulación secundarios en regulaciones en cascada, de mezcla o de relación, pero no como regulador piloto.

El procesamiento de señales en la rama de consigna o valor real, así como el tratamiento y vigilancia del error de regulación son idénticos a los del regulador continuo.

Arranque

PID_ES dispone de una rutina de inicialización que se ejecuta cuando el parámetro de entrada COM_RST = TRUE está activado.

En el arranque, la acción I se pone al valor de inicialización I_ITLVAL y se emite en la salida LMN_I. Si se llama en un nivel de alarma cíclica, continúa trabajando a partir de este valor.

Si el generador ramp soak está activado, los intervalos DB_NBR PI[0 ... NBR_PTS].TMV entre los nodos se suman y se indican en la salida del tiempo total T_TM y del tiempo residual total RT_TM. En caso de modificaciones online de PI[n].TMV o en caso de especificación de TM_CONT y TM_SNBR, cambian el tiempo total y el tiempo residual total de la curva de evolución. Ya que el cálculo de T_TM y RS_TM prolonga mucho el tiempo de procesamiento de la función RMP_SOAK si hay muchos nodos, dicho cálculo solo se realiza después del rearranque o si TUPDT_ON = TRUE.

Todas las demás salidas se ponen a sus valores predeterminados.

Rearranque

En caso de rearranque se parte del estado operativo que estaba activo en el momento de la interrupción. La regulación sigue trabajando con los valores que se calculaban en el instante de la interrupción.

Llamada

La instrucción PID_ES debe llamarse de forma equidistante. Utilice para ello un nivel de alarma cíclica (p. ej. OB35).

Si se efectúa una llamada de la instrucción PID_ES como DB multiinstancia, no se crea ningún objeto tecnológico. No se dispone de interfaz de parametrización y puesta en servicio. Los PID_ES se deben parametrizar directamente en el DB multiinstancia y ponerse en servicio a través de una tabla de observación.

Cálculo del error de regulación PID_ES

Las funciones siguientes son siempre efectivas y no es posible desactivarlas.

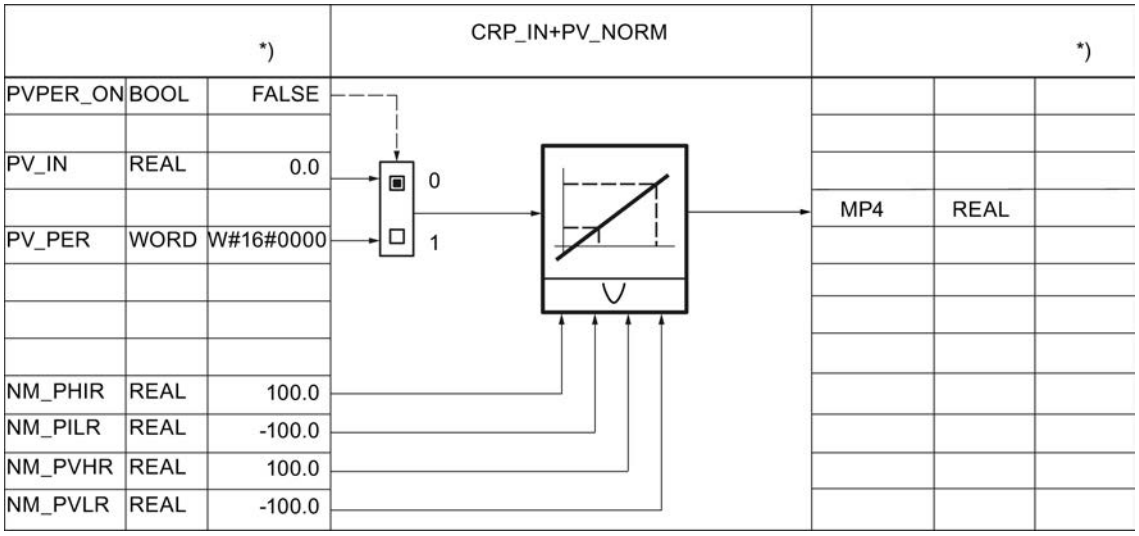
- Limitación de consigna SP_LIMIT,
- Vigilancia de valores absolutos del valor real PV_ALARM,
- Vigilancia de la magnitud de la velocidad de cambio del valor real ROCALARM,
- Vigilancia de valores absolutos del error de regulación ER_ALARM,
- Limitación de la señal de salida LMNLIMIT.

Los valores predeterminados se han seleccionado de modo que el funcionamiento sea correcto sin adaptación individual. Para conseguir buenos resultados de regulación es necesario adaptar siempre los parámetros de estas funciones al proceso.

Acondicionamiento del valor real

El valor real se normaliza a la magnitud física.

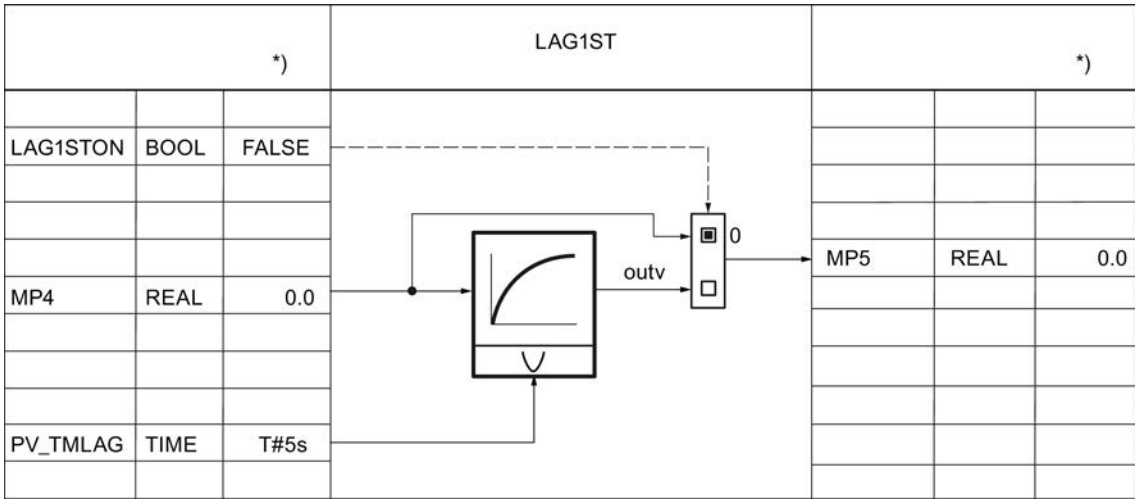
Normalización (PV_NORM) (Página 10117)



*) Ajuste predeterminado al crear el objeto tecnológico.

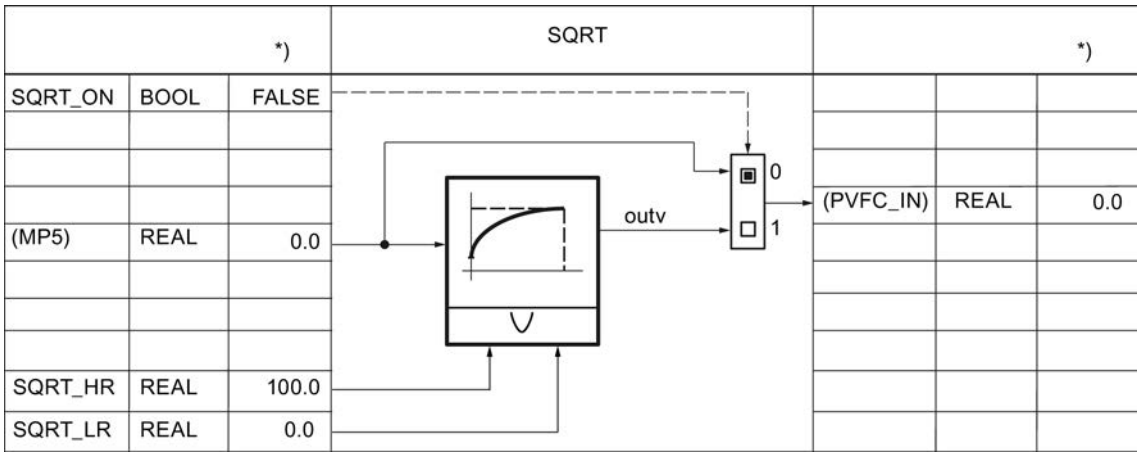
El valor real puede filtrarse con un elemento de retardo de primer orden.

Filtrar magnitud regulada (LAG1ST) (Página 10118)



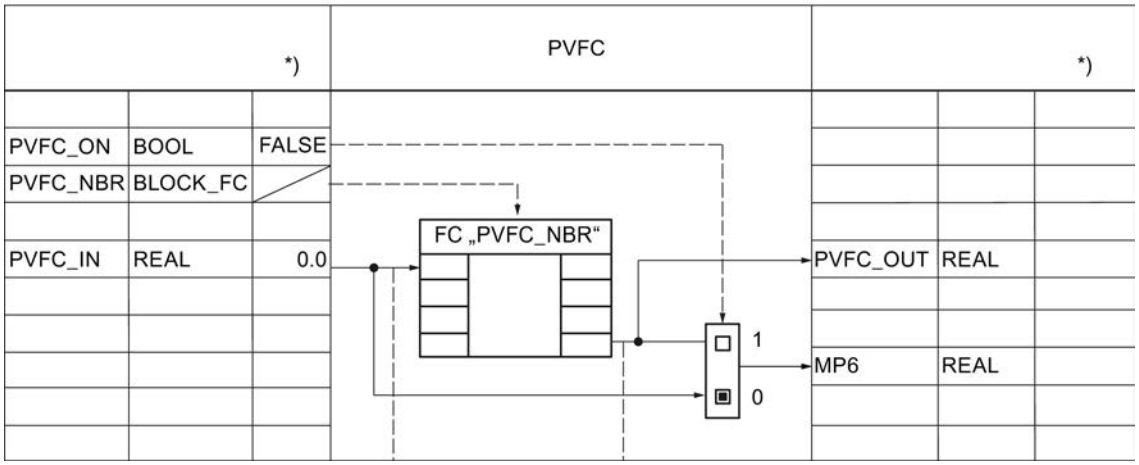
El valor real puede linealizarse con la raíz cuadrada. Para ello se requieren los parámetros siguientes:

Calcular raíz cuadrada (SQRT) (Página 10119)



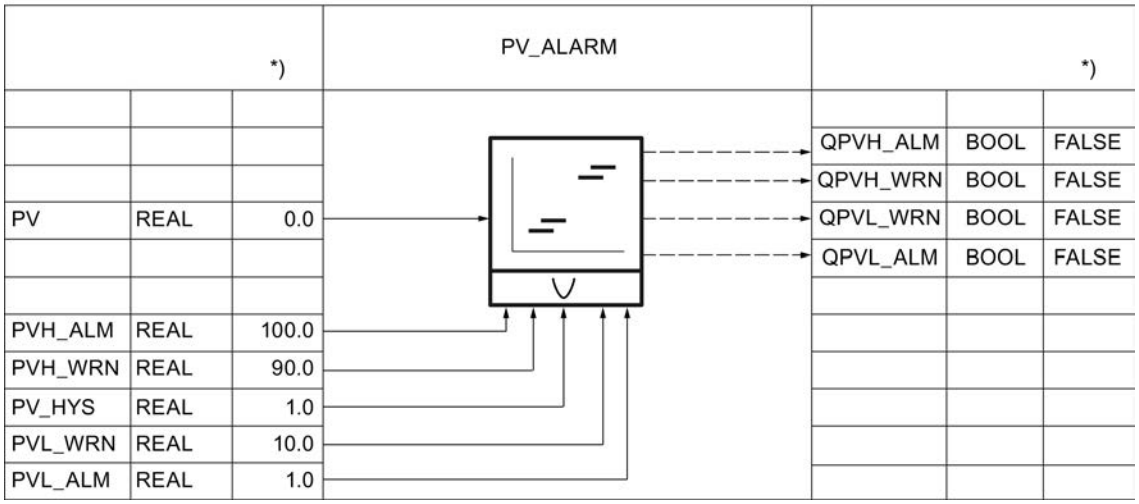
El valor real puede acondicionarse con una función específica del usuario. Para ello se requieren los parámetros siguientes. La interconexión debe programarse en la FC de usuario.

Llamada de FC en la rama de valor real (PVFC) (Página 10121)



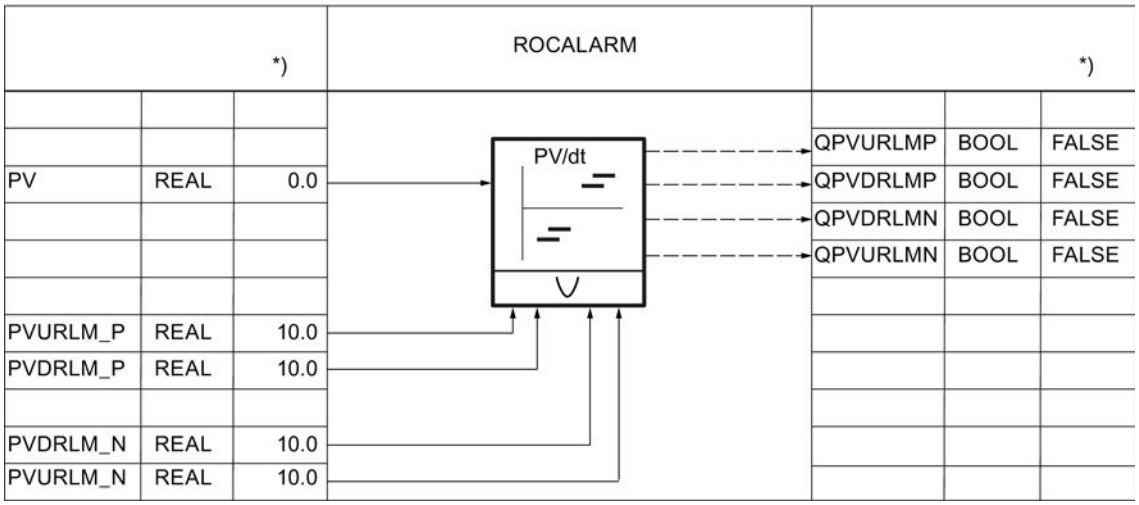
Los límites de advertencia y alarma del valor real se vigilan siempre con los parámetros siguientes.

Vigilar los límites del valor real (PV_ALARM) (Página 10121)



La velocidad de cambio del valor real se limita siempre con los parámetros siguientes.

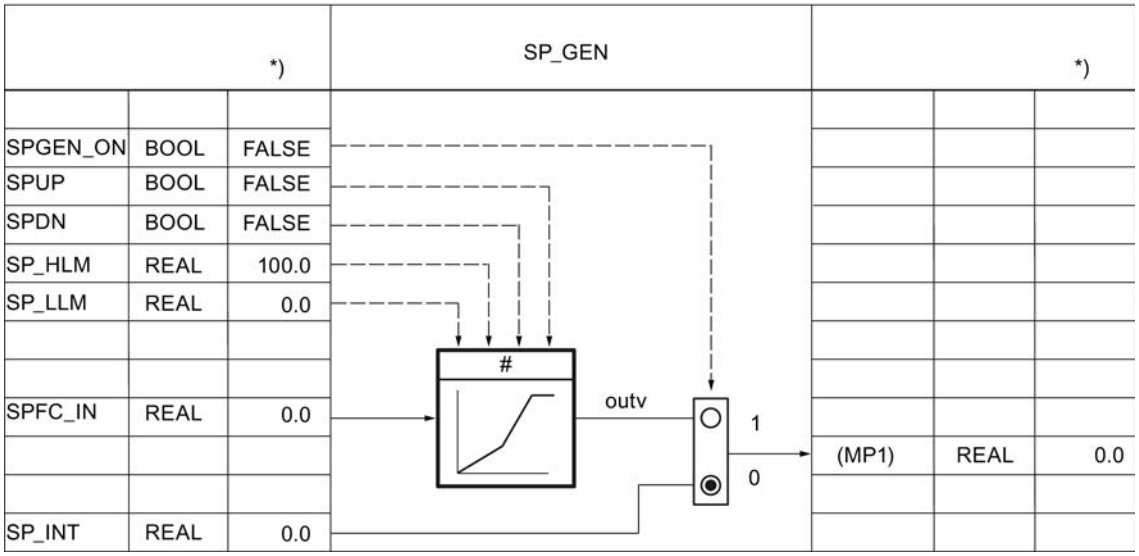
Vigilar la velocidad de cambio del valor real (ROCALARM) (Página 10123)



Acondicionamiento de la consigna

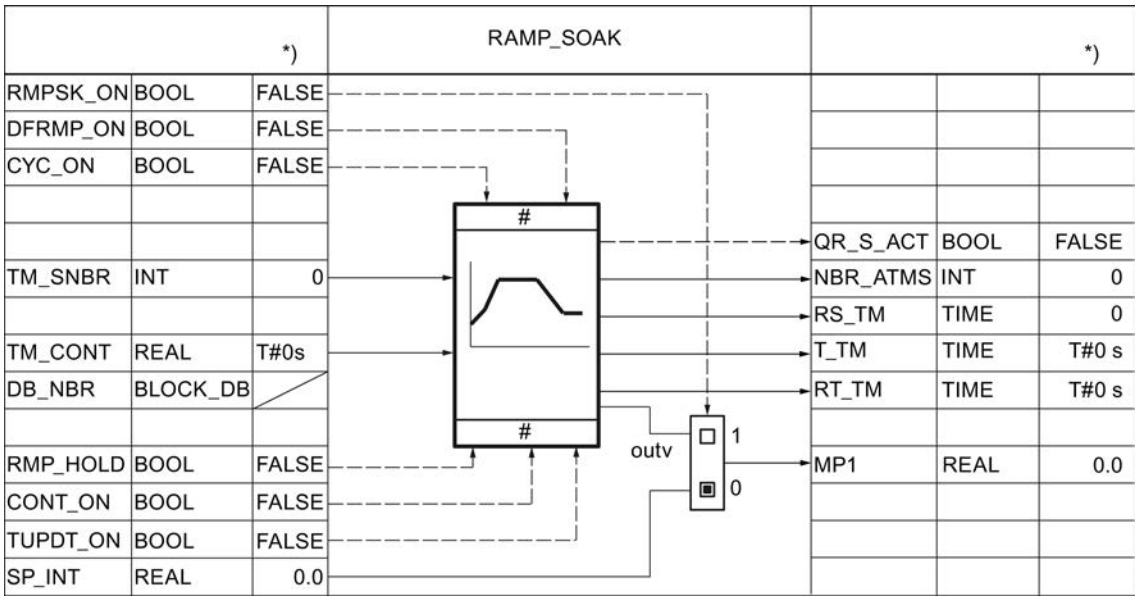
La consigna interna SP_IN se aumenta o reduce con los parámetros siguientes.

Generador de consigna (SP_GEN) (Página 10125)



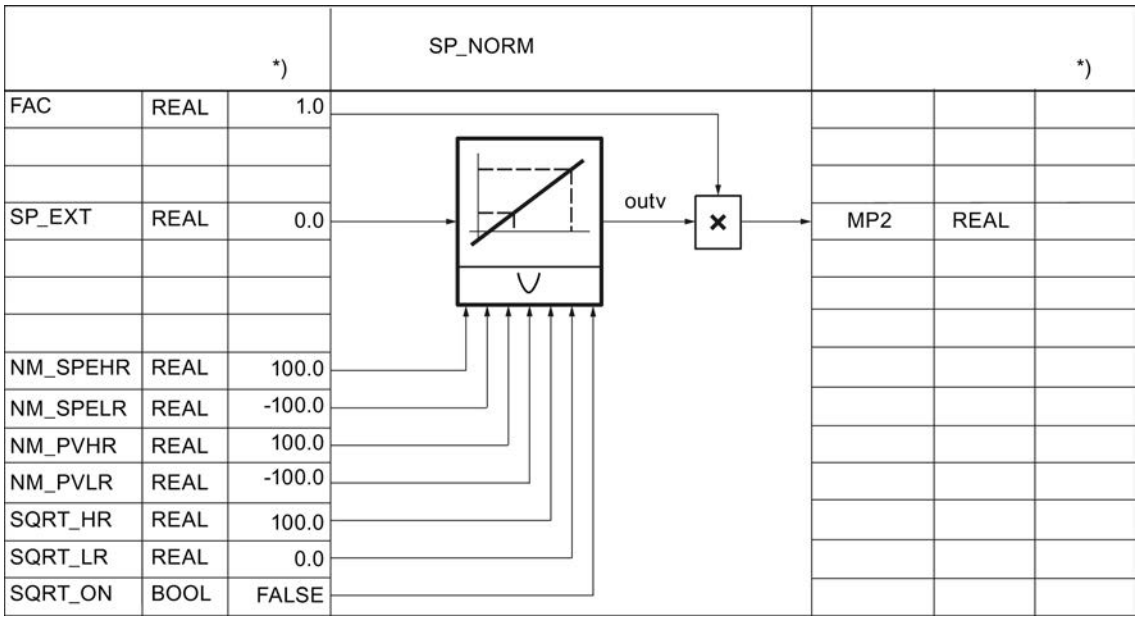
La consigna se especifica mediante un generador ramp soak con los parámetros siguientes.

Generador ramp soak (RMP_SOAK) (Página 10127)



Para reguladores de mezcla y relación, la consigna externa SP_EXT puede normalizarse con los parámetros siguientes.

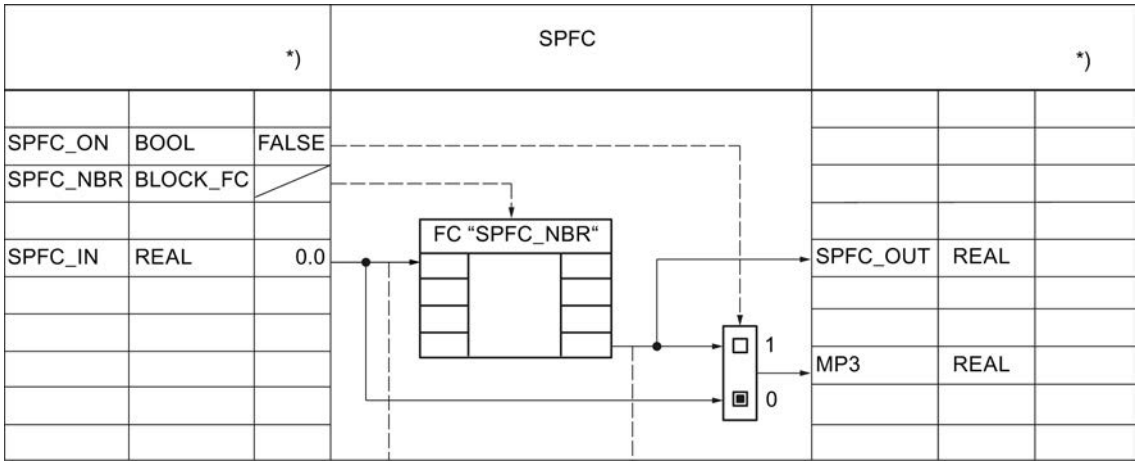
Normalización (Página 10133)



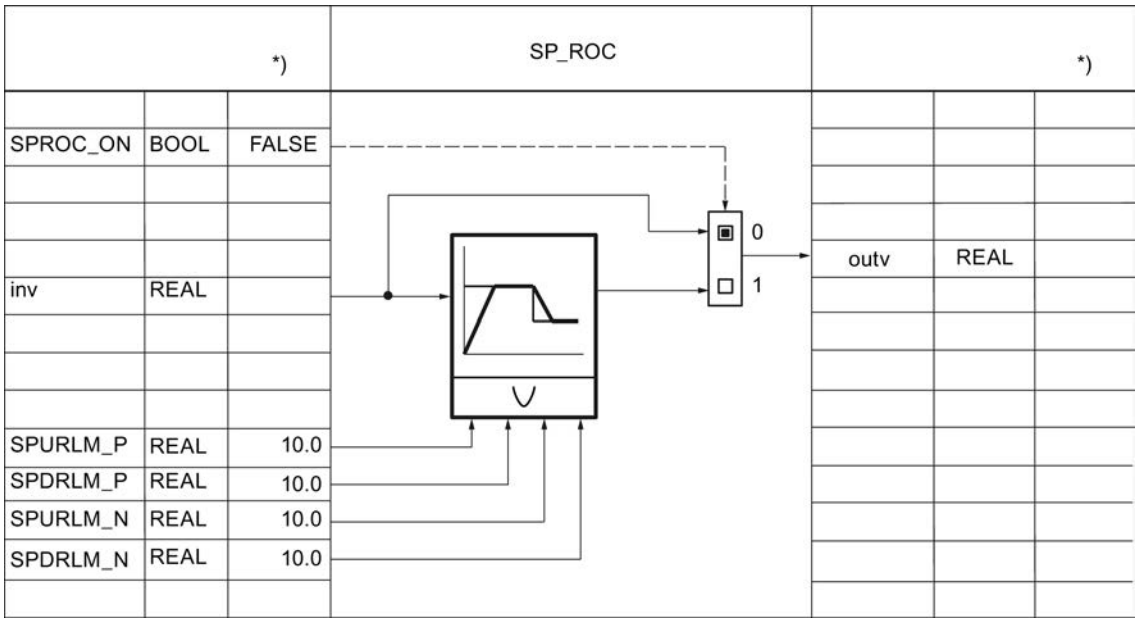
La consigna puede acondicionarse con una función específica del usuario. Para ello se requieren los parámetros siguientes. La interconexión debe programarse en la FC de usuario.

El valor de entrada SPFC_IN es un parámetro implícito. Se observa en la herramienta de configuración a través del punto de medición MP1 (SPEXT_ON = FALSE) o MP2 (SPEXT_ON = TRUE). El valor de salida es accesible en el punto de medición MP3. La entrada SPFC_IN está aplicada a la rama de consigna cuando SPFC_ON =FALSE.

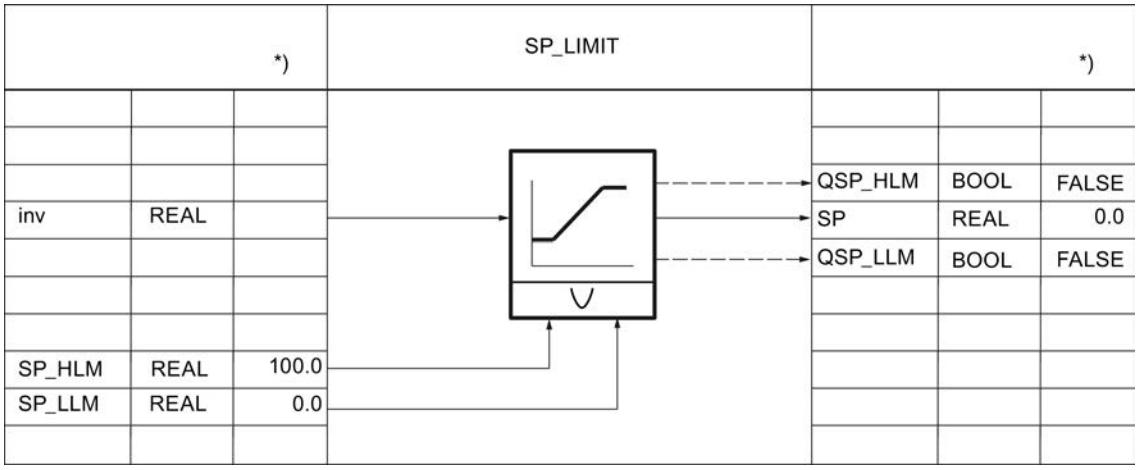
Límites (Página 10134)



La velocidad de cambio de la consigna se limita con los parámetros siguientes.



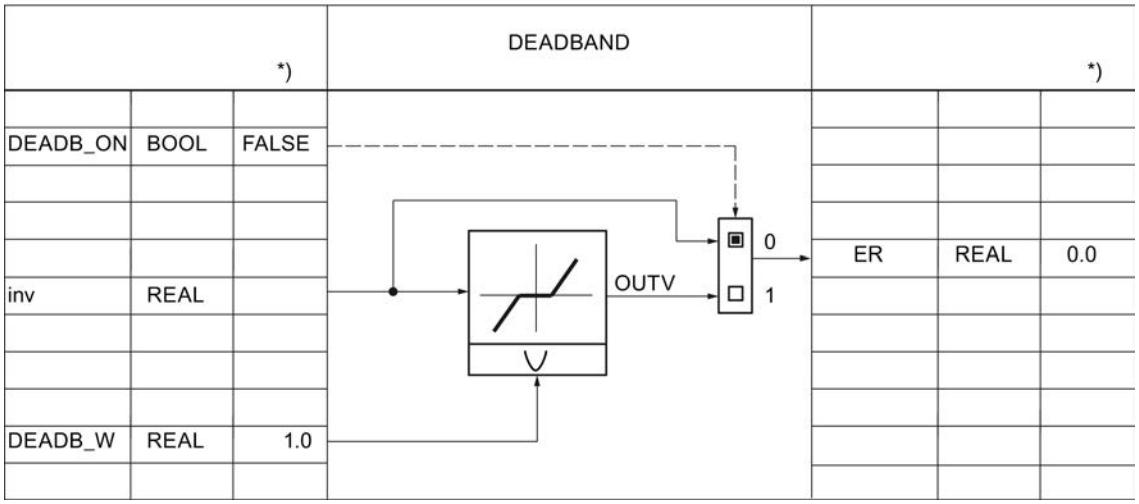
Con los parámetros siguientes se vigila siempre un límite superior e inferior absoluto de la consigna.



Calcular el error de regulación

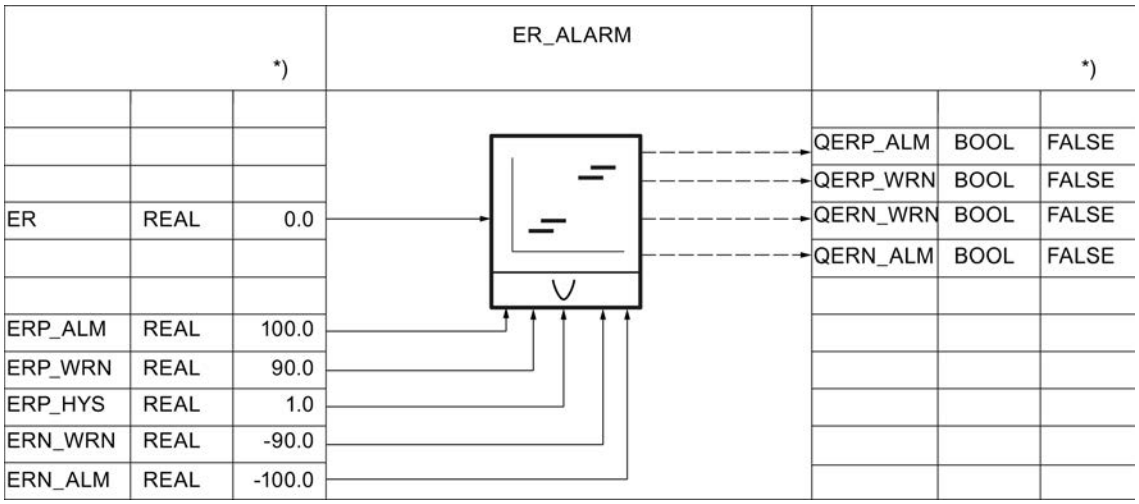
El error de regulación se conduce a través de una zona muerta con los parámetros siguientes.

Zona muerta (Página 10137)



Los límites de advertencia y alarma del error de regulación se vigilan siempre con los parámetros siguientes.

Alarma (Página 10138)



Algoritmo de regulación PID_ES

Algoritmo de regulación

Algoritmo de regulación (Página 10140)

Hay un total de cinco interruptores disponibles para definir una estructura de regulador efectiva.

Estructuras de regulador (Página 10142)

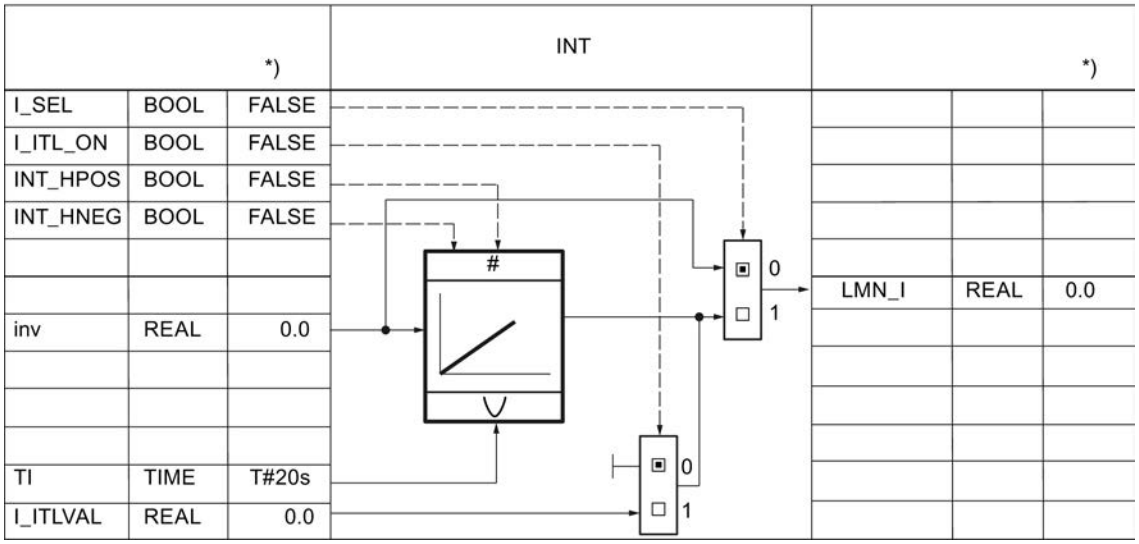
Estructura de regulador	Interruptor				
	P_SEL	I_SEL	D_SEL	PFDB_SEL	DFDB_SEL
Regulación P	TRUE	FALSE	FALSE	FALSE	FALSE
Regulación P (P en la rama de realimentación)	TRUE	FALSE	FALSE	TRUE	FALSE
Regulación PI	TRUE	TRUE	FALSE	FALSE	FALSE
Regulación PI (P en la rama de realimentación)	TRUE	TRUE	FALSE	TRUE	FALSE
Regulación PD	TRUE	FALSE	TRUE	FALSE	FALSE
Regulación PD (P en la rama de realimentación)	TRUE	FALSE	TRUE	FALSE	TRUE
Regulación PID	TRUE	TRUE	TRUE	FALSE	FALSE
Regulación PID (P/D en la rama de realimentación)	TRUE	TRUE	TRUE	FALSE	TRUE

La acción P queda determinada por la ganancia proporcional GAIN y puede observarse en LMN_P.

Acción P (Página 10145)

La acción I se calcula con los parámetros siguientes.

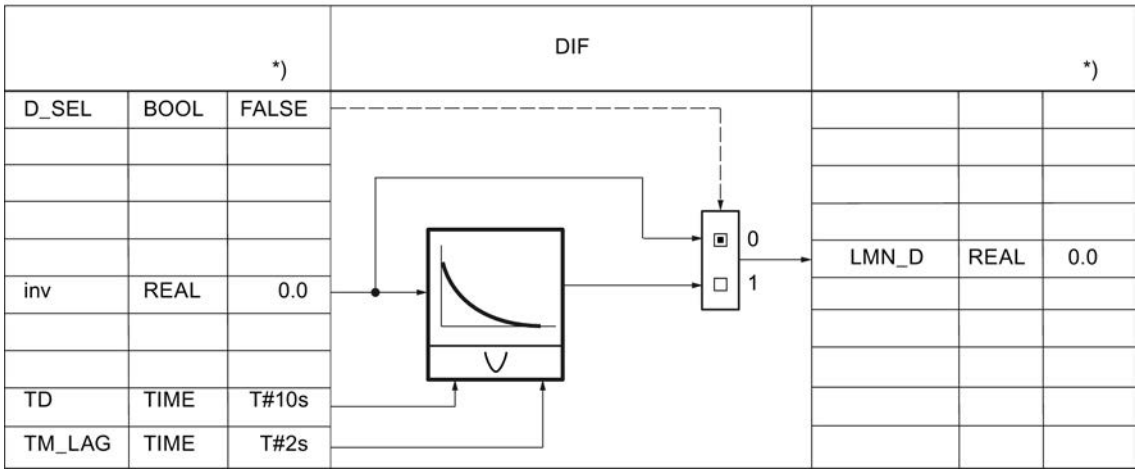
Acción I (Página 10146)



*) Ajuste predeterminado al crear el objeto tecnológico

La acción D se calcula con los parámetros siguientes.

Acción D (Página 10149)



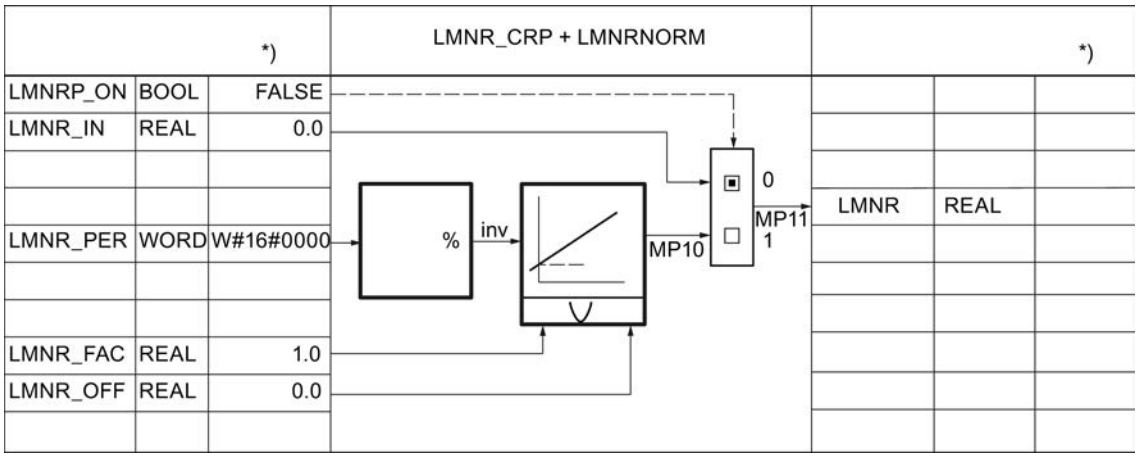
Variable manipulada PID_ES

Normalización de la realimentación de posición

La realimentación de posición en formato de periferia se convierte al formato de coma flotante. La realimentación de posición en formato de periferia puede observarse en MP10.

La tabla siguiente ofrece una visión de conjunto de los rangos y valores numéricos antes y después del procesamiento.

Valor de periferia LMNR_PER	LMNR en %
32767	118.515
27648	100.000
1	0.003617
0	0.000
-1	-0.003617
-27648	-100.000
-32768	-118.519

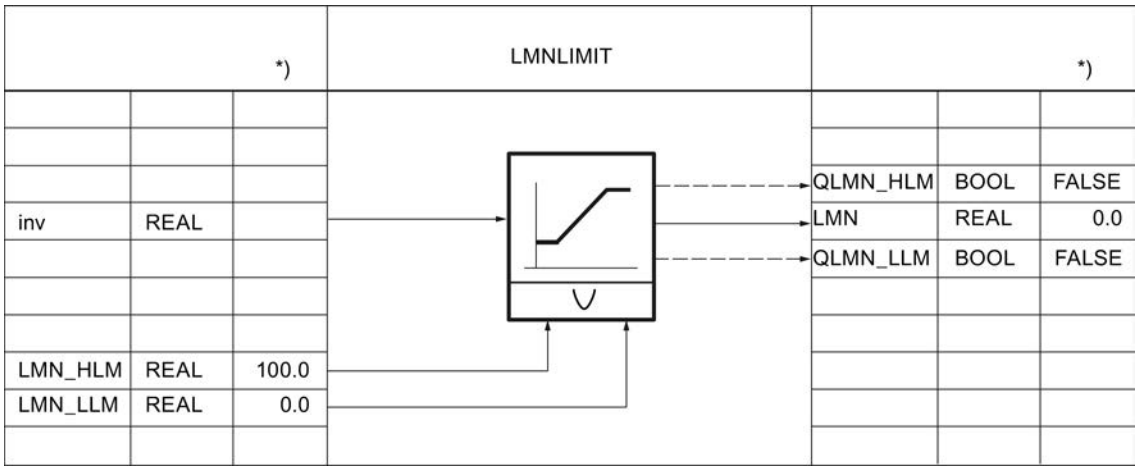


*) Ajuste predeterminado al crear el objeto tecnológico.

Limitación de la variable manipulada

Con los parámetros siguientes se vigila siempre un límite superior e inferior absoluto de la variable manipulada. El parámetro implícito puede observarse en MP9.

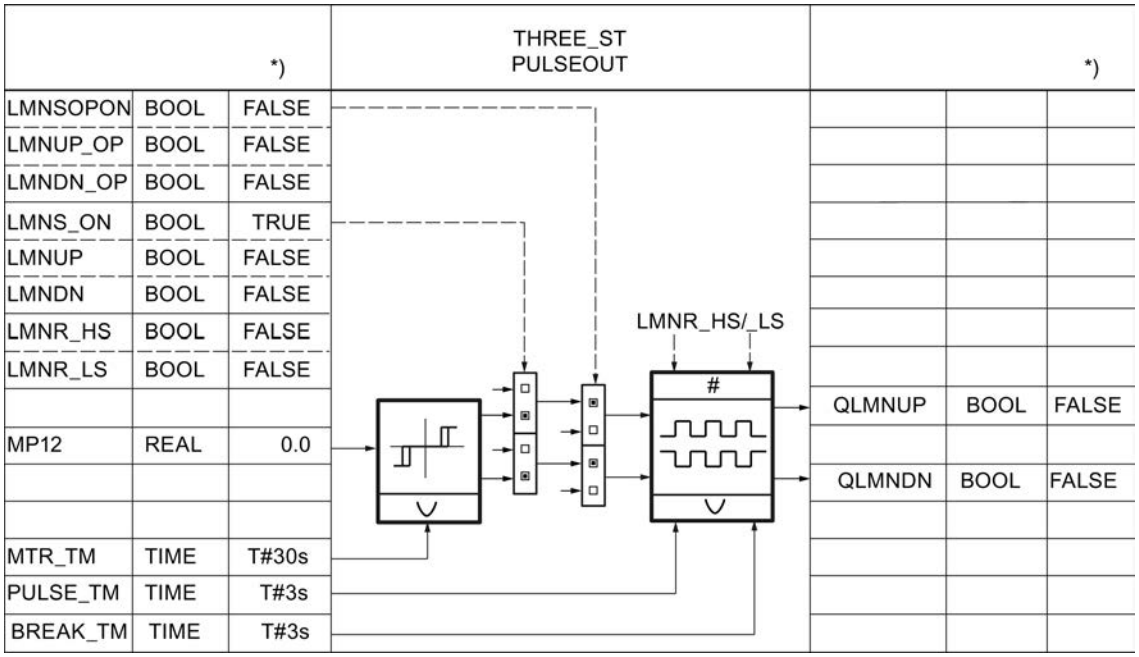
Límites (Página 10158)



Regulador paso a paso con realimentación de posición

Las señales manipuladas se forman con los parámetros siguientes en el regulador paso a paso con realimentación de posición.

Regulador paso a paso con realimentación de posición (Página 10150)

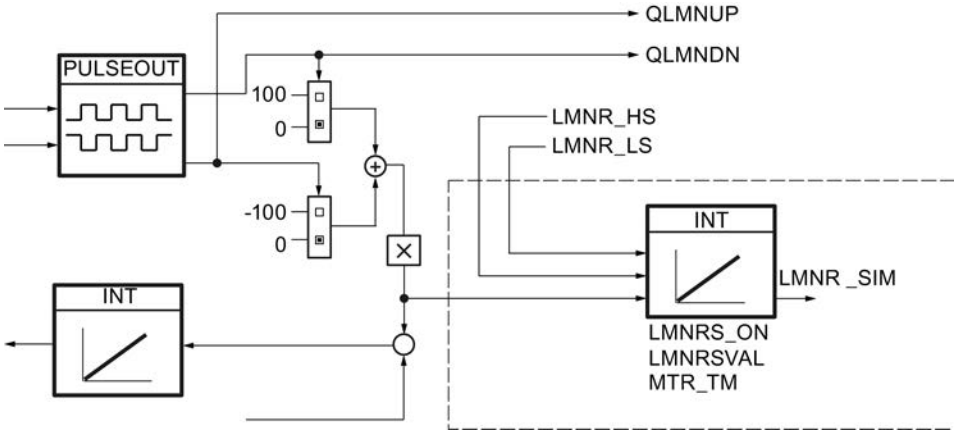


Regulador paso a paso sin realimentación de posición

La realimentación de posición puede simularse. La simulación de la realimentación de posición se realiza mediante un integrador usando el tiempo de posicionamiento del motor MTR_TM como constante de tiempo integrativo. En el estado LMNRS_ON = FALSE se emite el valor inicial del parámetro LMNRSVAL en la salida del integrador LMNR_ SIM. Después de conmutar a LMNRS_ON = TRUE, la simulación arranca con este valor inicial.

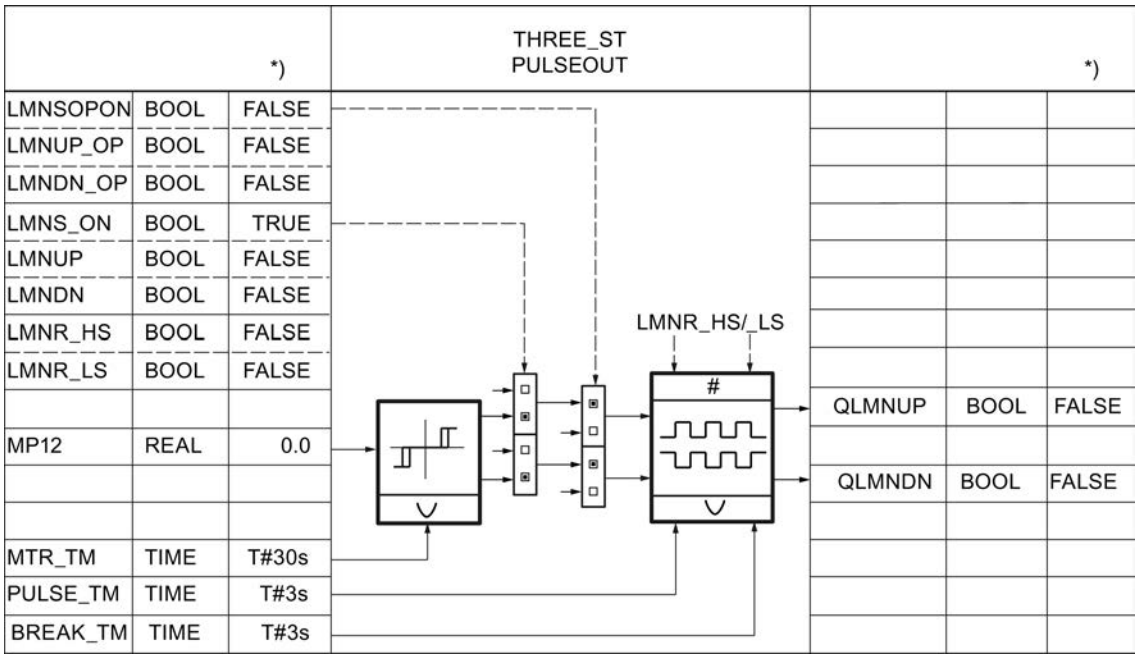
Si LMNR_HS = TRUE, la integración se detiene hacia arriba, si LMNR_LS = TRUE, se detiene hacia abajo. No se sincroniza la realimentación de posición simulada en las posiciones de tope.

Regulador paso a paso sin realimentación de posición (Página 10153)



Nota

La realimentación de posición solamente se simula. No tiene que coincidir con la posición real del actuador. Si existe una realimentación de posición real, se debería usar sin falta.



Consulte también

Modo manual (Página 10155)

Generador de impulsos (Página 10159)

PID_ES en circuitos en cascada

Separación de la cascada del regulador

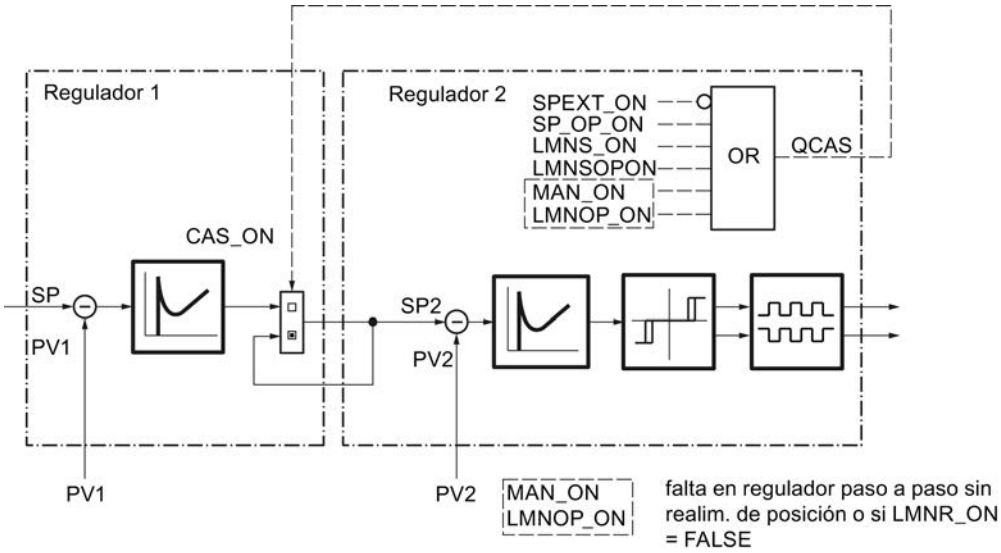
En una cascada hay varios reguladores directamente relacionados entre sí. Por ello, hay que tomar medidas para que, después de separar la estructura de la cascada, el funcionamiento en cascada pueda retomarse sin problemas en un punto cualquiera.

Por este motivo, en las regulaciones en cascada se forma una señal QCAS en los reguladores secundarios por medio de una lógica O a partir de las señales de estado de los interruptores en la rama de consigna y variable manipulada. Dicha señal acciona en los reguladores primarios un conmutador que pasa los reguladores al modo de seguimiento. La magnitud de seguimiento es siempre la consigna SP del lazo secundario.

La conmutación del modo de seguimiento al modo automático se realiza con suavidad, igual que la conmutación de manual a automático.

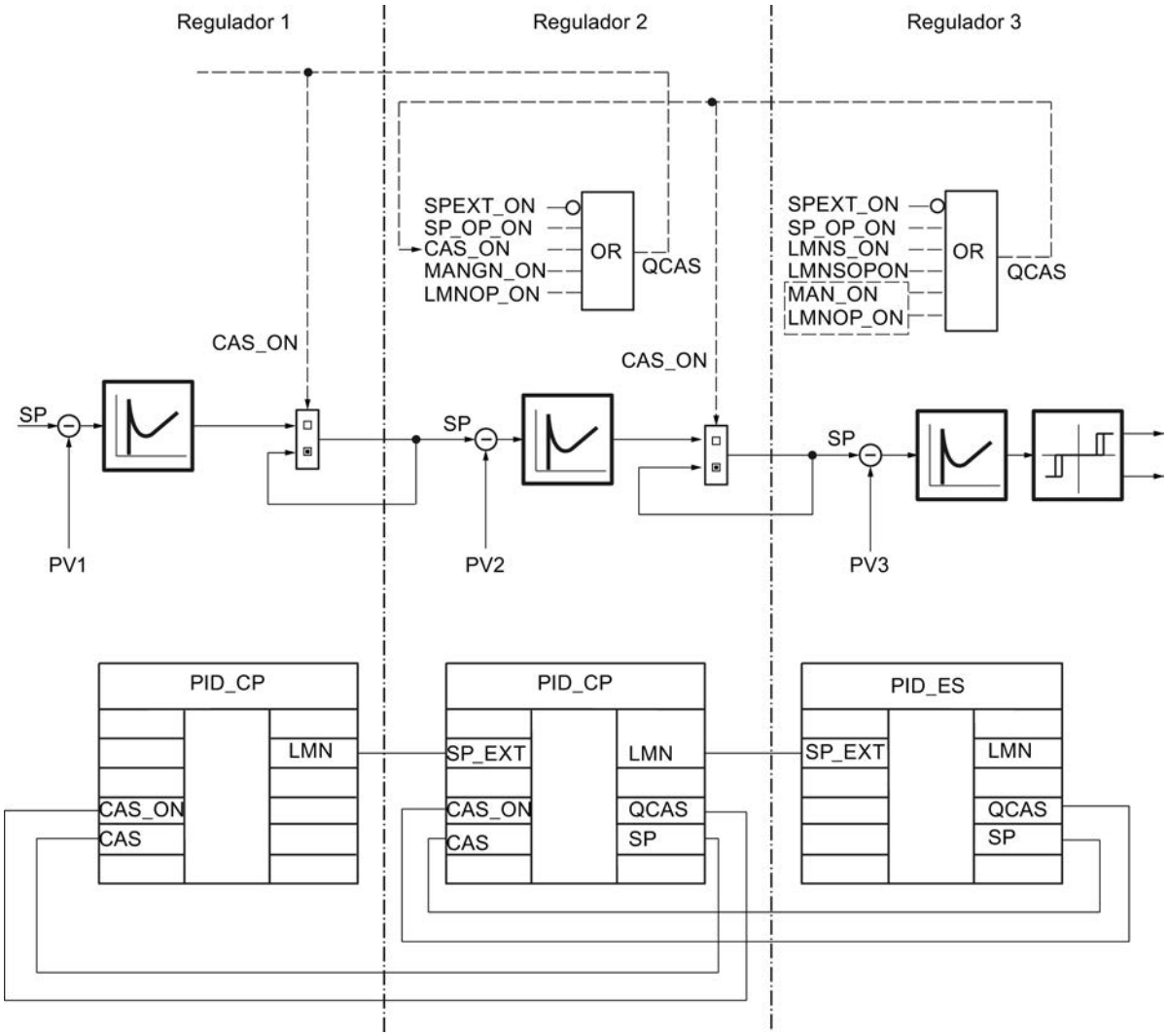
Nota

PID_ES puede utilizarse en conmutaciones en cascada únicamente como regulador secundario en lazos de regulación subordinados.


Regulador 1
Regulador 2
SPEXT_ON
SP_OP_ON
LMNS_ON
LMNSOPON
MAN_ON
LMNOP_ON
OR
QCAS
CAS_ON
SP
PV1
SP2
PV2
PV1
PV2
MAN_ON
LMNOP_ON
falta en regulador paso a paso sin realim. de posición o si LMNR_ON = FALSE

Interconexión de bloques

La figura siguiente muestra el principio de la interconexión de reguladores en cascadas multilazo.



Parámetros de entrada de PID_ES

Tabla 9-207

Parámetro	Offset	Tipo de datos	Rango	Ajuste predeterminado	Descripción
COM_RST	0.0	BOOL		FALSE	Rearranque (Se procesa la rutina de inicialización de la instrucción)
I_SEL	0.1	BOOL		TRUE	Activar la acción I
D_SEL	0.2	BOOL		FALSE	Activar la acción D

Parámetro	Offset	Tipo de datos	Rango	Ajuste predeterminado	Descripción
MAN_ON	0.3	BOOL		TRUE	Activar el modo manual (Lazo de regulación deshecho, LMN ajustado manualmente)
LMNR_HS	0.4	BOOL		FALSE	Señal de tope superior de la realimentación de posición
LMNR_LS	0.5	BOOL		FALSE	Señal de tope inferior de la realimentación de posición
CYCLE	2.0	TIME	> 20 ms (S7–300)	T#1s	Tiempo de muestreo (Tiempo entre llamadas del bloque = constante) Es esencial asegurarse de que el parámetro se parametriza con el tiempo de alarma cíclica del OB en el que está activa la instrucción PID_CP. En caso contrario las funciones dependientes del tiempo no funcionan correctamente. (Excepción: se utiliza un reductor de tiempo, por ejemplo a través del distribuidor de llamadas del regulador.)
SP_INT	6.0	REAL	Rango técnico (magnitud física)	0.0	Consigna interna SP_INT puede modificarse con SPUP y SPDN mediante un HMI.
SP_EXT	10.0	REAL	Rango técnico (magnitud física)	0.0	Consigna externa SP_EXT se utiliza para reguladores de mezcla, relación y cascada y puede normalizarse.
PV_IN	14.0	REAL	Rango técnico (magnitud física)	0.0	Entrada de valor real (PV en formato de coma flotante)
PV_PER	18.0	INT		W#16#0000	Valor real de periferia
GAIN	20.0	REAL	Rango completo (sin dimensión)	2.0	Ganancia proporcional (= ganancia del regulador)
TI	24.0	TIME	TI ≥ CYCLE	T#20s	Tiempo integrativo
TD	28.0	TIME	TD ≥ CYCLE	T#10s	Tiempo derivativo
TM_LAG	32.0	TIME	TM_LAG ≥ CYCLE/ 2	T#2s	Tiempo de retardo de la acción D
DISV	36.0	REAL	-100.0 ... 100.0 [%]	0.0	Magnitud perturbadora
LMNR_IN	40.0	REAL	0.0 ... 100.0 [%]	0.0	Realimentación de posición (LMNR en formato de coma flotante)
LMNR_PER	44.0	INT		W#16#0000	Realimentación de posición de periferia (LMNRen el formato de la periferia)
SP_HLM	46.0	REAL	Rango técnico (magnitud física)	100.0	Consigna del límite superior
SP_LLM	50.0	REAL	Rango técnico (magnitud física)	0.0	Consigna del límite inferior
LMN_HLM	54.0	REAL	LMN_LLM ... 100.0 [%]	100.0	Variable manipulada: límite superior

Parámetro	Offset	Tipo de datos	Rango	Ajuste predeterminado	Descripción
LMN_LLM	58.0	REAL	0.0 ... LMN_HLM [%]	0.0	Variable manipulada: límite inferior
DB_NBR	62.0	BLOCK_DB		DB 1	Número del bloque de datos (DB con nodos de la curva)
SPFC_NBR	64.0	BLOCK_FC		FC 0	Consigna número de FC (FC autodefinida en la rama de consigna)
PVFC_NBR	66.0	BLOCK_FC		FC 0	Valor real número de FC (FC autodefinida en la rama del valor real)

Parámetros de salida de PID_ES

Tabla 9-208

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QLMNUP	68.0	BOOL	FALSE	Señal alta de la variable manipulada
QLMNDN	68.1	BOOL	FALSE	Señal baja de la variable manipulada
QCAS	68.2	BOOL	FALSE	Señal para modo de cascada (Se interconecta con CAS_ON del regulador de nivel superior)
LMN	70.0	REAL	0.0	Señal de la variable manipulada (según algoritmo de regulación)
SP	74.0	REAL	0.0	Consigna (Consigna efectiva)
PV	78.0	REAL	0.0	Valor real (Emisión del valor real efectivo en el modo de cascada)

Parámetros de entrada/salida de PID_ES

Tabla 9-209

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
MAN	82.0	REAL	0.0	Variable manipulada manual (Para la asignación predeterminada de la variable manipulada mediante funciones de manejo y visualización)

Variables estáticas de PID_ES

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
PVH_ALM	86.0	REAL	PVH_WRN...100.0	100.0	Valor real: límite superior de 'Alarma' Rango admisible:
PVH_WRN	90.0	REAL	PVL_WRN... PVH_ALM	90.0	Valor real: límite superior de 'Advertencia' Rango admisible:
PVL_WRN	94.0	REAL	PVL_ALM... PVH_WRN	-90.0	Valor real: límite inferior de 'Advertencia' Rango admisible:
PVL_ALM	98.0	REAL	-100.0...PVL_WRN	-100.0	Valor real: límite inferior de 'Alarma' Rango admisible:
SPGEN_ON	102.0	BOOL		FALSE	Activar control de la consigna (Ajustar la consigna a través de interruptores Arriba/Abajo)
SPUP	102.1	BOOL		FALSE	Consigna alta
SPDN	102.2	BOOL		FALSE	Consigna baja
RMPSK_ON	102.3	BOOL		FALSE	Activar el generador ramp soak (La consigna se predetermina como curva)
SPEXT_ON	102.4	BOOL		FALSE	Activar consigna externa (Para la interconexión con otros bloques de regulador)
MANGN_ON	102.5	BOOL		FALSE	Activar el modo manual ((LMN ajustado con interruptor)
MANUP	102.6	BOOL		FALSE	Variable manipulada manual alta
MANDN	102.7	BOOL		FALSE	Variable manipulada manual baja
LMNS_ON	103.0	BOOL		FALSE	Activar el modo manual de las señales manipuladas
LMNUP	103.1	BOOL		FALSE	Señal alta de la variable manipulada (Se controla la señal de salida QLMNUP)
LMNDN	103.2	BOOL		FALSE	Señal baja de la variable manipulada (Se controla la señal de salida QLMNDN)
DFRMP_ON	103.3	BOOL		FALSE	Preasignar la salida del generador ramp soak (A la salida se le preasigna SP_INT)
CYC_ON	103.4	BOOL		FALSE	Activar repetición cíclica (La curva se repite automáticamente)
RMP_HOLD	103.5	BOOL		FALSE	Detener generador ramp soak (Se congela la salida del generador ramp soak)
CONT_ON	103.6	BOOL		FALSE	(Generador ramp soak) continuar (Se continúa la curva en el siguiente nodo)
TUPDT_ON	103.7	BOOL		FALSE	Recalcular tiempo total (Se recalcula el tiempo total de la curva)
SPFC_ON	104.0	BOOL		FALSE	Llamar FC de consigna
SPROC_ON	104.1	BOOL		FALSE	Activar función de rampa (Limitación de la velocidad de cambio de SP)

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
PVPER_ON	104.2	BOOL		FALSE	Activar el valor real de periferia (Interconexión con los módulos de periferia)
LAG1STON	104.3	BOOL		FALSE	Activar elemento de retardo de primer orden
SQRT_ON	104.4	BOOL		FALSE	Activar función de raíz cuadrada
PVFC_ON	104.5	BOOL		FALSE	Llamar FC de valor real
DEADB_ON	104.6	BOOL		FALSE	Activar zona muerta (Se filtran pequeñas perturbaciones y ruidos)
P_SEL	104.7	BOOL		TRUE	Activar la acción P
PFDB_SEL	105.0	BOOL		FALSE	Conectar la acción P en la rama de realimentación
INT_HPOS	105.1	BOOL		FALSE	Congelar la acción I en dirección positiva
INT_HNEG	105.2	BOOL		FALSE	Congelar la acción I en dirección negativa
I_ITL_ON	105.3	BOOL		FALSE	Preasignar la acción I
DFDB_SEL	105.4	BOOL		FALSE	Aplicar la acción D en la rama de realimentación
DISV_SEL	105.5	BOOL		FALSE	Control anticipativo
LMNR_ON	105.6	BOOL		FALSE	Activar la realimentación de posición (Modos de operación: Regulador paso a paso con/sin realimentación de posición) No conmutar en el lazo cerrado.
LMNRP_ON	105.7	BOOL		FALSE	Activar la realimentación de posición de periferia
TM_SNBR	106.0	INT	≥ 0 (sin dimensión)	0	Número del siguiente nodo para continuar la curva
TM_CONT	108.0	TIME	Rango completo (sin dimensión)	T#0s	Período hasta la continuación de la curva (Período que transcurre antes de que el generador ramp soak siga procesando la curva en el nodo TM_SNBR tras una interrupción)
FAC	112.0	REAL	Rango completo (sin dimensión)	1.0	Factor (Factor de relación o mezcla)
NM_SPEHR	116.0	REAL		100.0	Normalización de la consigna: entrada arriba
NM_SPELR	120.0	REAL		-100.0	Normalización de la consigna: entrada abajo
SPFC_OUT	124.0	REAL	-100.0 ... 100.0	0.0	Consigna de la salida de FC (Se interconecta con la salida de la FC en la rama de consigna)
SPURLM_P	128.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de rampa de subida de la consigna en el rango positivo
SPDRLM_P	132.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de rampa de bajada de la consigna en el rango positivo
SPURLM_N	136.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de subida de la consigna en el rango negativo

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
SPDRLM_N	140.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de bajada de la consigna en el rango negativo
NM_PIHR	144.0	REAL		100.0	Normalización del valor real: entrada arriba
NM_PILR	148.0	REAL		-100.0	Normalización del valor real: entrada abajo
NM_PVHR	152.0	REAL		100.0	Normalización del valor real: salida arriba
NM_PVLR	156.0	REAL		-100.0	Normalización del valor real: salida abajo
PV_TMLAG	160.0	TIME	Rango completo	T#5s	Tiempo de retardo del valor real (Retardo del elemento PT1 en la rama PV)
SQRT_HR	164.0	REAL		100.0	Raíz cuadrada: salida del rango de medición arriba
SQRT_LR	168.0	REAL		0.0	Raíz cuadrada: salida del rango de medición abajo
PVFC_OUT	172.0	REAL	-100.0 ... 100.0	0.0	Valor real de la salida de FC (Se interconecta con la salida de la FC en la rama del valor real)
PVURLM_P	176.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de subida del valor real en el rango positivo
PVDRLM_P	180.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de bajada del valor real en el rango positivo
PVURLM_N	184.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de subida del valor real en el rango negativo
PVDRLM_N	188.0	REAL	≥ 0 [magnitud física/s]	10.0	Límite de la rampa de bajada del valor real en el rango negativo
PV_HYS	192.0	REAL	≥ 0	1.0	Histéresis del valor real (Evita el "centelleo" de la vista de monitorización)
DEADB_W	196.0	REAL	0.0 ... 100.0	1.0	Ancho de zona muerta (Determina el tamaño del rango de insensibilidad)
ERP_ALM	200.0	REAL	0 ... 200.0	100.0	Error de regulación: límite positivo de 'Alarma'
ERP_WRN	204.0	REAL	0 ... 200.0	90.0	Error de regulación: límite positivo de 'Advertencia'
ERN_WRN	208.0	REAL	-200.0 ... 0	-90.0	Error de regulación: límite negativo de 'Advertencia'
ERN_ALM	212.0	REAL	-200.0 ... 0	-100.0	Error de regulación: límite negativo de 'Alarma'
ER_HYS	216.0	REAL	≥0	1.0	Error de regulación de histéresis (Evita el "centelleo" de la vista de monitorización)
I_ITLVAL	220.0	REAL	-100.0 ... 100.0 [%]	0.0	Valor de inicialización para la acción I
LMNR_FAC	224.0	REAL	Rango completo (sin dimensión)	1.0	Factor de realimentación de posición (Factor para la adaptación del rango de realimentación)

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
LMNR_OFF	228.0	REAL	-100.0 ... 100.0 [%]	0.0	Offset de realimentación de posición (Punto cero de la normalización de realimentación)
PULSE_TM	232.0	TIME	= n * CYCLE / n=0,1,2...	T#3s	Duración mínima de impulso
BREAK_TM	236.0	TIME	= n * CYCLE / n=0,1,2...	T#3s	Duración mínima de pausa
MTR_TM	240.0	TIME	≥ CYCLE	T#30s	Tiempo de posicionamiento del motor
PHASE	244.0	INT		0	Fase del PID Self-Tuner
QPVH_ALM	246.0	BOOL		FALSE	Valor real: el límite superior de 'Alarma' ha reaccionado
QPVH_WRN	246.1	BOOL		FALSE	Valor real: el límite superior de 'Advertencia' ha reaccionado
QPVL_WRN	246.2	BOOL		FALSE	Valor real: el límite inferior de 'Advertencia' ha reaccionado
QPVL_ALM	246.3	BOOL		FALSE	Valor real: el límite inferior de 'Alarma' ha reaccionado
QR_S_ACT	246.4	BOOL		FALSE	Procesando tabla de tiempo para curva
QSP_HLM	246.5	BOOL		FALSE	Consigna: el límite superior ha reaccionado
QSP_LLM	246.6	BOOL		FALSE	Consigna: el límite inferior ha reaccionado
QPVURLMP	246.7	BOOL		FALSE	Valor real: el límite de la rampa de subida en el rango positivo ha reaccionado
QPVDRLMP	247.0	BOOL		FALSE	Valor real: el límite de la rampa de bajada en el rango positivo ha reaccionado
QPVURLMN	247.1	BOOL		FALSE	Valor real: el límite de la rampa de subida en el rango negativo ha reaccionado
QPVDRLMN	247.2	BOOL		FALSE	Valor real: el límite de la rampa de bajada en el rango negativo ha reaccionado
QERP_ALM	247.3	BOOL		FALSE	Error de regulación: el límite positivo de 'Alarma' ha reaccionado
QERP_WRN	247.4	BOOL		FALSE	Error de regulación: el límite positivo de 'Advertencia' ha reaccionado
QERN_WRN	247.5	BOOL		FALSE	Error de regulación; el límite negativo de 'Advertencia' ha reaccionado
QERN_ALM	247.6	BOOL		FALSE	Error de regulación: el límite negativo de 'Alarma' ha reaccionado
QLMN_HLM	247.7	BOOL		FALSE	Variable manipulada: el límite superior ha reaccionado
QLMN_LLM	248.0	BOOL		FALSE	Variable manipulada: el límite inferior ha reaccionado
NBR_ATMS	250.0	INT		0	Número del nodo actual del generador ramp soak
RS_TM	252.0	TIME		T#0s	"Tiempo residual actual" de la curva hasta el siguiente nodo
T_TM	256.0	TIME		T#0s	"Tiempo total" = tiempo pasado de la curva

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
RT_TM	260.0	TIME		T#0s	"Tiempo residual total" = tiempo hasta el final de la curva
ER	264.0	REAL		0.0	Error de regulación
LMN_P	268.0	REAL		0.0	Acción P
LMN_I	272.0	REAL		0.0	Acción I
LMN_D	276.0	REAL		0.0	Acción D
SPFC_IN	280.0	REAL		0.0	Consigna entrada de FC (Se interconecta con la entrada de la FC autodefinida)
PVFC_IN	284.0	REAL		0.0	Valor real entrada de FC (Se interconecta con la entrada de la FC autodefinida)
SP_OP_ON	288.0	BOOL		FALSE	Activar control de la consigna (El valor de SP_OP se aplica como consigna)
PV_OP_ON	288.1	BOOL		FALSE	Activar control manual del valor real (El valor de PV_OP se aplica como consigna)
LMNOP_ON	288.2	BOOL		FALSE	Activar control manual de la variable manipulada (El valor de LMN_OP se aplica como consigna)
LMNSOPON	288.3	BOOL		FALSE	Activar control manual de la señal de la variable manipulada (LMNUP_OP y LMNDN_OP se aplican como señales manipuladas)
LMNUP_OP	288.4	BOOL		FALSE	Señal alta de la variable manipulada
LMNDN_OP	288.5	BOOL		FALSE	Señal baja de la variable manipulada
LMNRS_ON	288.6	BOOL		FALSE	Activar simulación de la realimentación de posición
SP_OP	290.0	REAL		0.0	Control manual de la consigna en la herramienta de configuración
PV_OP	294.0	REAL		0.0	Control manual del valor real en la herramienta de configuración
LMN_OP	298.0	REAL		0.0	Control manual de la variable manipulada en la herramienta de configuración
LMNRSVAL	302.0	REAL		0.0	Valor inicial de la realimentación de posición simulada
LMNR_SIM	306.0	REAL		0.0	Valor actual de la realimentación de posición simulada
MP1	310.0	REAL		0.0	Punto de medición 1: consigna interna
MP2	314.0	REAL		0.0	Punto de medición 2: consigna externa
MP3	318.0	REAL		0.0	Punto de medición 3: consigna ilimitada
MP4	322.0	REAL		0.0	Punto de medición 4: valor real del módulo de periferia
MP5	326.0	REAL		0.0	Punto de medición 5: valor real después del elemento de retardo de primer orden

Parámetro	Offset	Tipo de datos	Rango	Asignación previa	Descripción
MP6	330.0	REAL		0.0	Punto de medición 6: valor real efectivo (PV)
MP7	334.0	REAL		0.0	Punto de medición 7: variable manipulada del algoritmo PID
MP8	338.0	REAL		0.0	Punto de medición 8: variable manipulada manual
MP9	342.0	REAL		0.0	Punto de medición 9: variable manipulada ilimitada
MP10	346.0	REAL		0.0	Punto de medición 10: realimentación de posición periferia
MP11	350.0	REAL		0.0	Punto de medición 11: valor de realimentación (LMNR_ON = FALSE) Realimentación de posición (LMNR_ON = TRUE)
MP12	354.0	REAL		0.0	Punto de medición 12: entrada de elemento de tres puntos

Diagramas de bloques de PID_ES

Diagramas del flujo de señales

De forma análoga a la representación de los interruptores en la herramienta de configuración, el punto negro en los símbolos de los interruptores significa que la señal de conmutación tiene el valor booleano adyacente (0 = FALSE o 1 = TRUE) y que el recorrido de la señal está aplicado a este punto en cada caso. Las señales de conmutación (señales binarias) están marcadas con líneas discontinuas.

Esto significa que la consigna se especifica de modo absoluto mediante SP_INT; lo mismo es válido para la entrada del valor real mediante PV_IN. Como función de regulación se ha ajustado un regulador PI normal con función P en la rama de avance. El lazo de regulación es abierto y se influye sobre la magnitud manipulada mediante la entrada MAN en el área porcentual. Todas las demás funciones son pasivas o, si no pueden desactivarse, no son efectivas debido al ajuste predeterminado de parámetros "marginales" en la evolución de las señales dentro del rango de medida o del área de trabajo.

Símbolos e identificaciones en los diagramas de flujo de señales

Las denominaciones de las magnitudes del proceso que pueden conectarse se representan con el fondo marcado. Gracias a ellas se reconoce dónde puede conectarse la estructura del regulador con la periferia S7 o directamente con órganos de medición y los actuadores del proceso.

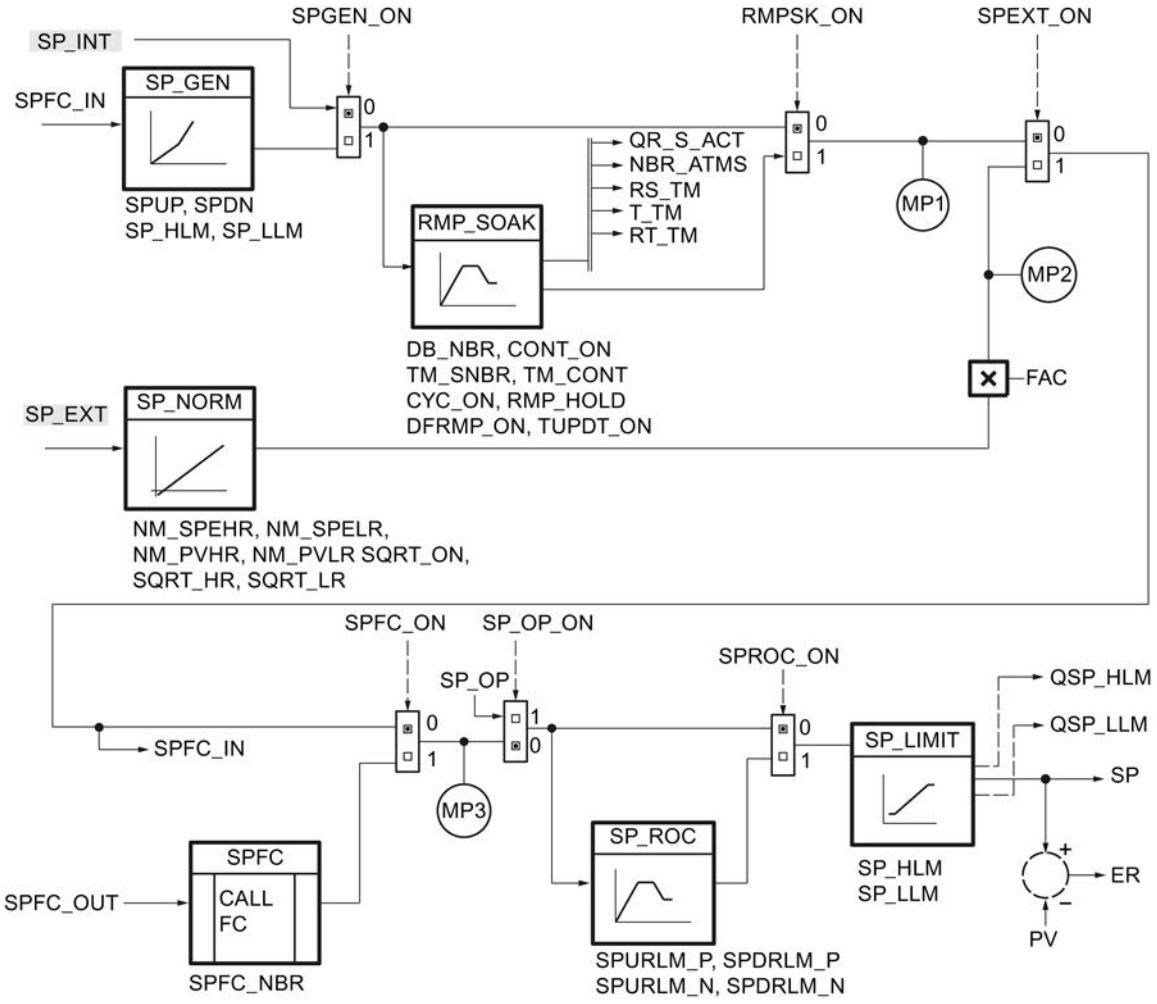
Las denominaciones de parámetros con la combinación de letras "OP" (p. ej. SP_OP/ SP_OP_ON) hacen referencia a la posible intervención en este punto mediante la herramienta de configuración de Standard PID Control. La herramienta de configuración tiene una interfaz propia con el FB de regulación.

Las magnitudes intermedias en la evolución de las señales pueden observarse mediante los puntos de medición MP1 ... MP12 marcados con círculos. Dichas magnitudes intermedias son necesarias para sincronizar valores antes de lanzar operaciones de conmutación "suaves" o

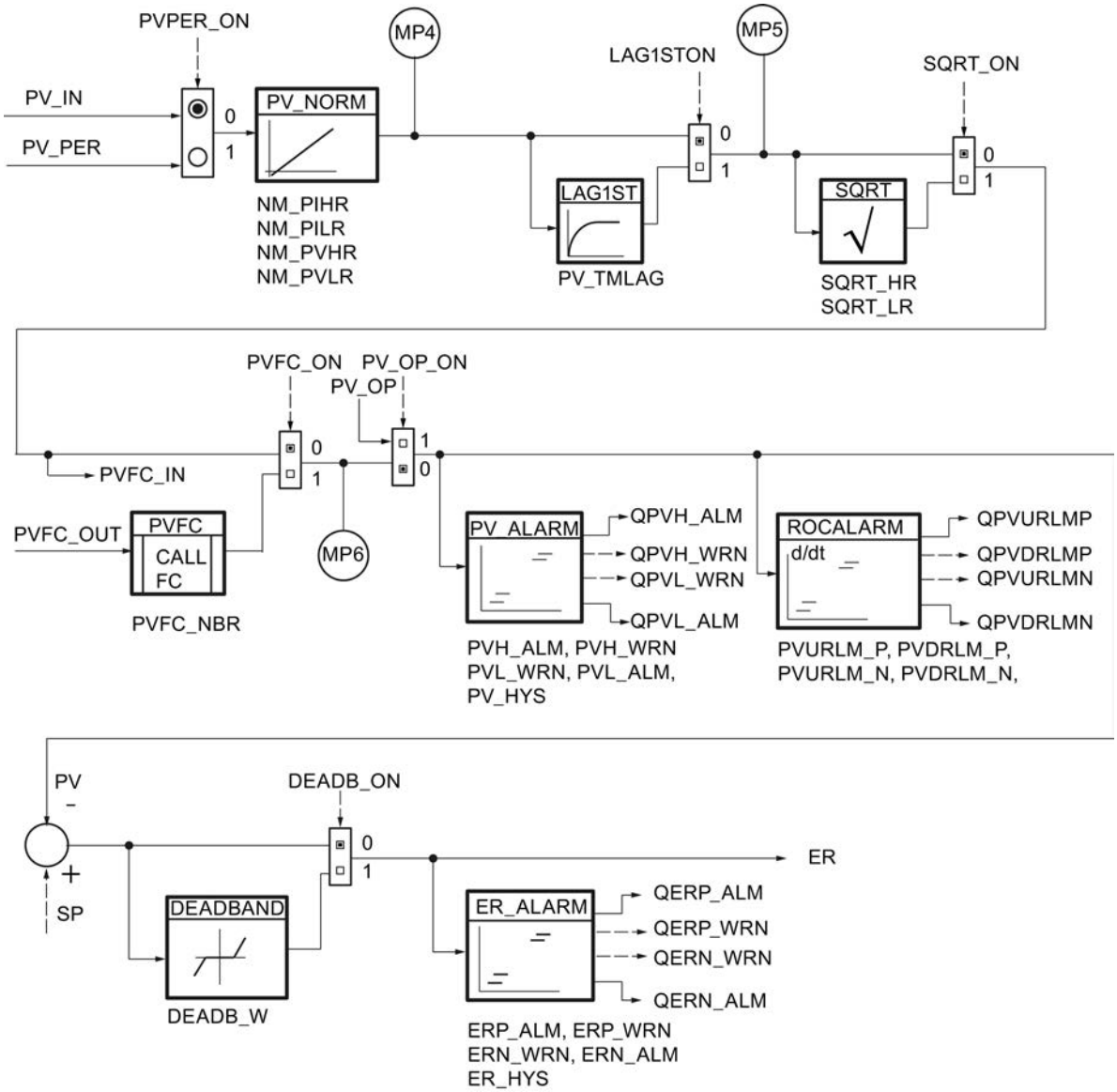
para inspeccionar los estados actuales de la regulación en cuestión. Las magnitudes del punto de medición en el registrador de curvas de la herramienta de configuración pueden representarse tanto estática como dinámicamente.

Por motivos de claridad, los parámetros para ajustar y dimensionar las operaciones de procesamiento (algoritmos) están indicados en los distintos bloques de función. Remitimos al respecto a las descripciones de la parte de referencia y a la representación de las diferentes funciones parciales en los apartados siguientes.

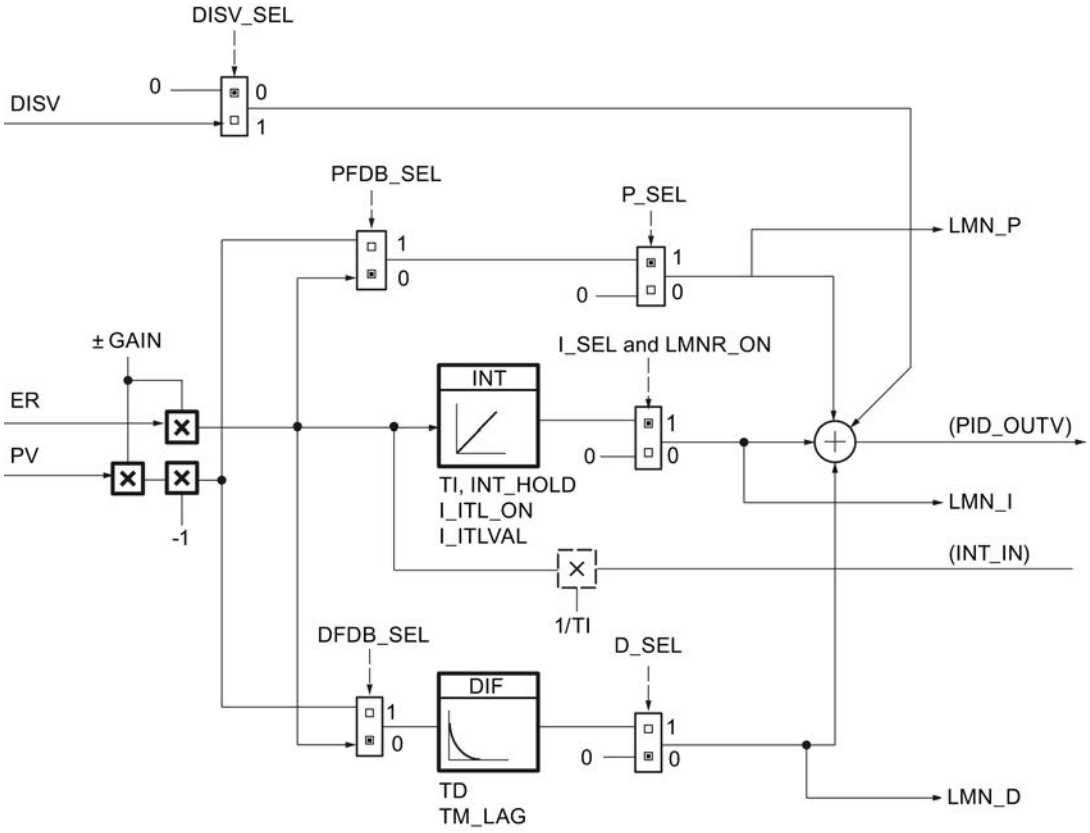
Acondicionamiento de la consigna



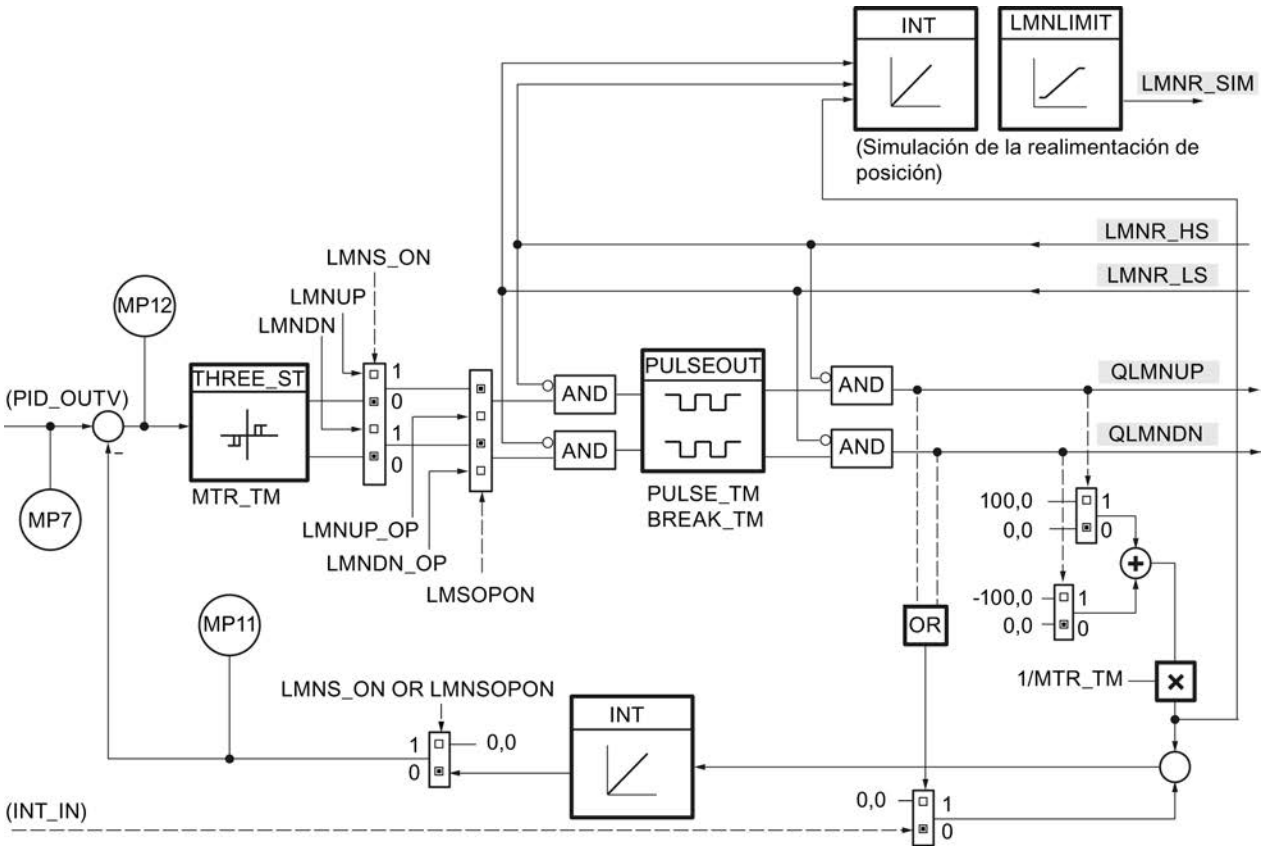
Acondicionamiento del valor real



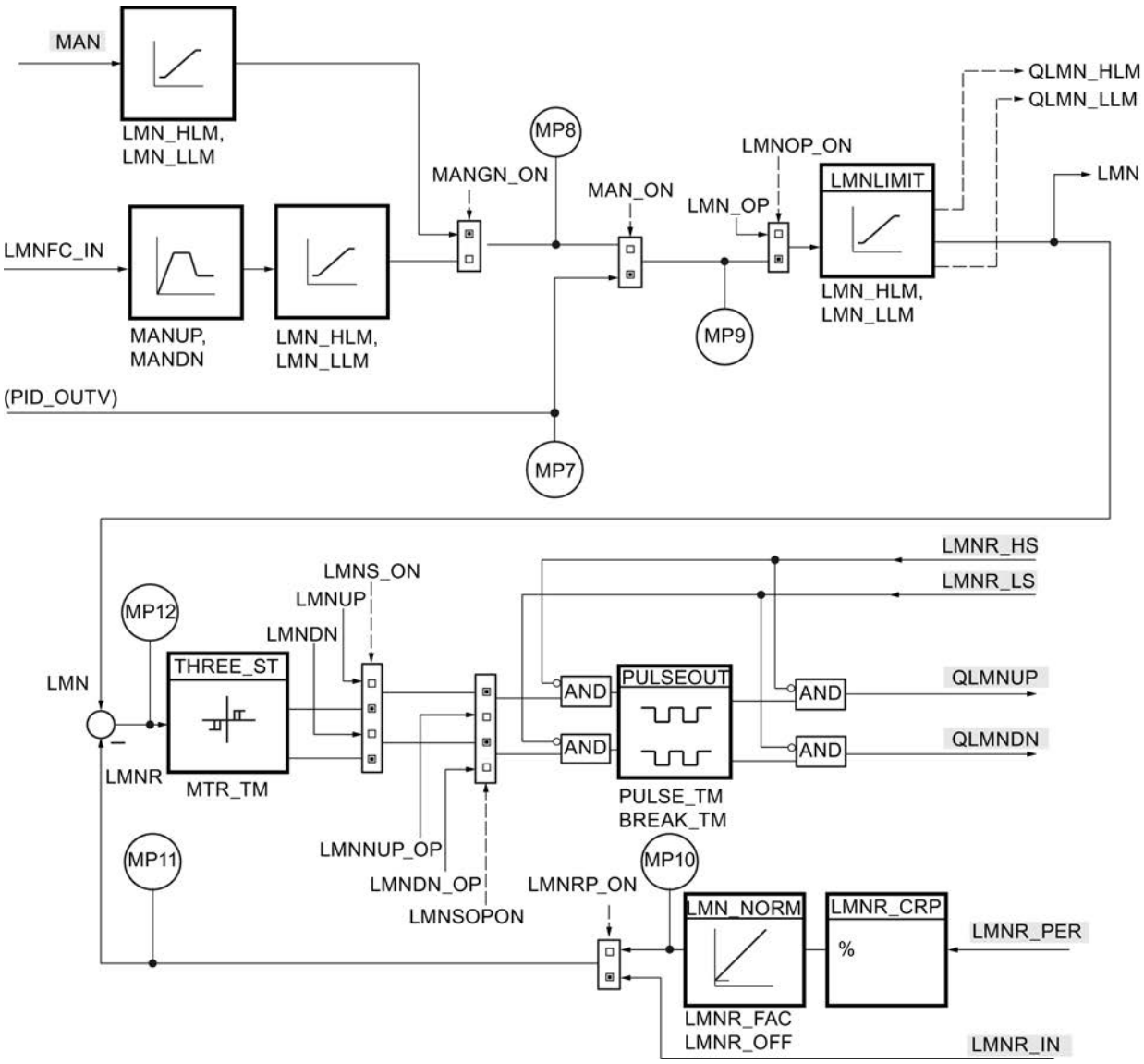
Algoritmo de regulación



Regulador paso a paso sin realimentación de posición (LMNR_ON = FALSE)



Regulador paso a paso con realimentación de posición (LMNR_ON = TRUE)



Bloque de datos RMP_SOAK

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
NBR_PTS	INT	4	Número de nodos -1 1 - 255
PI	Array [0...NBR_PTS] of Struct		Estructura de los datos de los nodos. La matriz debe ser declarada por 0...NBR_PTS.
PI[0].OUTV	REAL	0.0	Valor de salida [0] punto inicial Rango técnico

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
PI[0].TMV	TIME	T#1s	Valor temporal [0] punto inicial
PI[1].OUTV	REAL	0.0	Valor de salida [1] nodo 1 Rango técnico
PI[1].TMV	TIME	T#1s	Valor temporal [1] nodo 1
PI[2].OUTV	REAL	0.0	Valor de salida [2] nodo 2 Rango técnico
PI[2].TMV	TIME	T#1s	Valor temporal [2] nodo 2
PI[3].OUTV	REAL	0.0	Valor de salida [3] nodo 3 Rango técnico
PI[3].TMV	TIME	T#1s	Valor temporal [3] nodo 3
PI[4].OUTV	REAL	0.0	Valor de salida [4] nodo 4 Rango técnico
PI[4].TMV	TIME	T#0s	Valor temporal [4] nodo 4 0 ms

Consulte también

Generador ramp soak (RMP_SOAK) (Página 10127)

LP_SCHED

Descripción de LP_SCHED

LP_SCHED se utiliza cuando el número de alarmas cíclicas de una CPU no es suficiente para conseguir los (distintos) tiempos de muestreo deseados. Permite llamar un máximo de 256 lazos de regulación con tiempos de muestreo que suponen un múltiplo entero del ciclo de la alarma cíclica.

A partir del bloque de datos global DB_LOOP, LP_SCHED lee los parámetros especificados, calcula a partir de ellos las variables necesarias para la distribución de llamadas y las guarda en el bloque de datos DB_LOOP.

Nota

El bloque ni comprueba si un bloque de datos global con el número DB_NBR realmente existe, ni si el parámetro GLP_NBR (número más alto de lazos de regulación) coincide con la longitud del bloque de datos. Al no parametrizar correctamente, la CPU cambia al estado STOP y muestra "Error interno del sistema".

Llamada

LP_SCHED debe llamarse en un OB de alarma cíclica antes de todos los reguladores.

Al asignar valores a los parámetros de entrada, tenga en cuenta lo siguiente:

- TM_BASE: en esta entrada se indica el ciclo del OB de alarma cíclica en el que se llama la instrucción LP_SCHED.
- COM_RST: en un arranque de la CPU debe llamarse LP_SCHED una vez con COM_RST = TRUE. En el funcionamiento cíclico (alarma cíclica) hay que llamar LP_SCHED con COM_RST = FALSE.
- DB_NBR: en esta entrada se indica el número del bloque de datos "DB_LOOP" al que debe acceder LP_SCHED.

Tras llamar la instrucción LP_SCHED deben llamarse todos los respectivos reguladores de forma condicionada. La llamada de los reguladores PID_CP o PID_ES no puede realizarse mediante LP_SCHED, puesto que al llamar las instrucciones tienen que asignarse sus parámetros de entrada y salida.

A continuación se muestra un ejemplo de la llamada de la instrucción LP_SCHED y de la llamada condicionada de un regulador.

AWL

```
    CALL "LP_SCHED"
        TM_BASE:                                    Aquí se parametriza el ciclo de la alarma
                                                    cíclica. Ej.: T#100ms o #CYCLE con CYCLE =
                                                    parámetro de entrada del bloque en el que se
                                                    llama la instrucción LP_SCHED.
        COM_RST:                                    Aquí se notifica a la instrucción LP_SCHED si
                                                    debe llevarse a cabo una inicialización de
                                                    los lazos de regulación llamados. Ej.: FALSE
                                                    o #COM_RST con COM_RST = parámetro de entrada
                                                    del bloque en el que se llama la instrucción
                                                    LP_SCHED.
        DB_NBR                                      Aquí se parametriza el número del DB
                                                    "DB_LOOP" que debe procesar la instrucción
                                                    LP_SCHED. Ej.: "DB_LOOP" con DB_LOOP = nombre
                                                    del DB asignado en la tabla de símbolos.
    A "DB_LOOP".LOOP_DAT[1].ENABLE
    SPBN M002                                       Llamada del lazo de regulación si ENABLE =
                                                    TRUE
    CALL FBx,DBy
      COM_RST:= "DB_LOOP".LOOP_DAT[1].COM_RST
      :                                             Lista de operandos formales
      :                                             Lista de operandos formales
      CYCLE:= "DB_LOOP".LOOP_DAT[1].CYCLE
      :                                             Lista de operandos formales
      :                                             Lista de operandos formales
    CLR
    = "DB_LOOP".LOOP_DAT[1].ENABLE                  Desactivar bit ENABLE
M002:                                               continuar con el programa, p. ej. llamada
                                                    condicionada del siguiente FB del lazo de
                                                    regulación
    :
```

Arranque

Si COM_RST = TRUE, se realizan los siguientes ajustes predeterminados:

Número de lazo de regulación actual: ALP_NBR = 0

Los datos de llamada de todos los lazos de regulación hasta **GLB_NBR** se preasignan del siguiente modo:

Habilitación:	ENABLE =	NOT MAN_DIS
Tiempo de muestreo:	CYCLE =	GV(MAN_CYC)
Rearranque:	COM_RST =	TRUE
Número de llamada local:	ILP_COU =	0
GV(MAN_CYC) = MAN_CYC se redondea a un múltiplo entero de TM_BASE*GLP_NBR		

En un arranque de la CPU hay que llamar LP_SCHED desde el OB de arranque correspondiente y asignar el valor TRUE a la entrada COM_RST. Seguidamente, llame PID_CP o PID_ES de forma condicionada para que se inicialicen los reguladores.

En el OB de alarma cíclica hay que volver a asignar el valor FALSE a COM_RST.

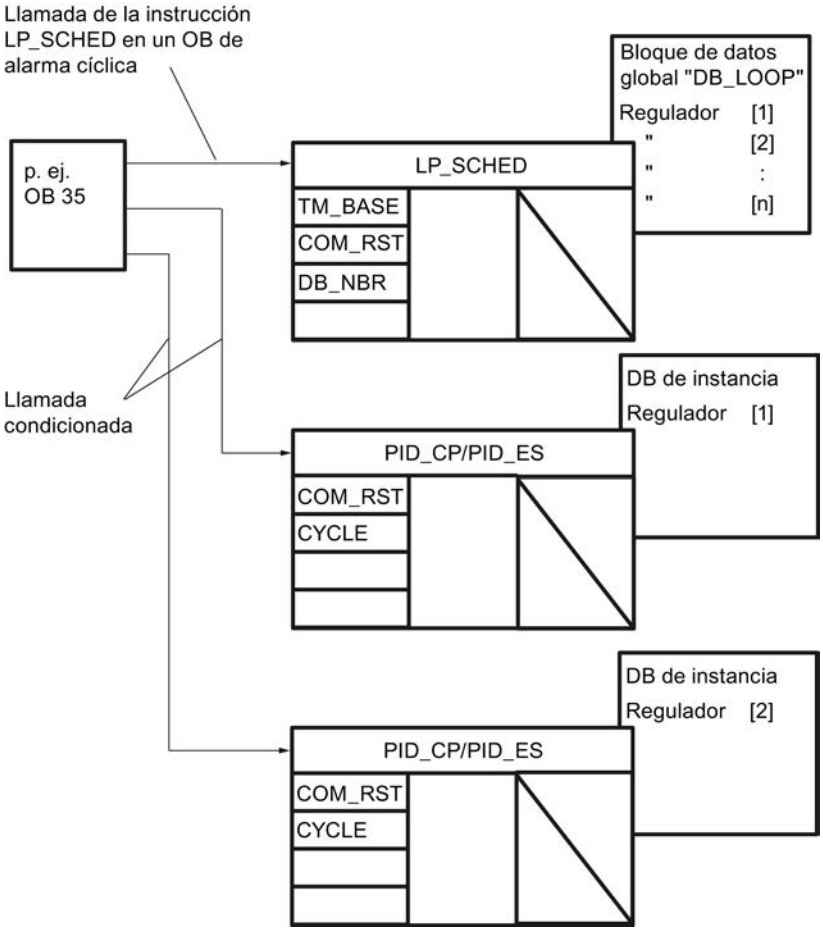
Funcionamiento de LP_SCHED

El procesamiento de la llamada

Un regulador debe procesarse cuando el bit ENABLE correspondiente en el bloque de datos DB_LOOP tenga el valor TRUE. Dicho bit ha sido escrito previamente por LP_SCHED. Si ya se ha procesado el regulador, hay que asignar el valor FALSE al bit ENABLE tras el procesamiento.

Al llamar el regulador deben interconectarse sus parámetros de entrada COM_RST y CYCLE con las variables COM_RST[x] y CYCLE[x] del bloque de datos DB_LOOP. CYCLE[x] contiene el tiempo de muestreo real del regulador x y es escrito por LP_SCHED en cada ciclo.

Durante el funcionamiento existe la posibilidad de bloquear manualmente la llamada de diferentes reguladores y también resetear los reguladores que se desee.



Parametrización del bloque de datos DB_LOOP

La parametrización del distribuidor de llamadas de regulador debe realizarse en el editor de DB.

Hay que parametrizar las variables siguientes en el bloque de datos DB_LOOP:

- GLP_NBR: número de los reguladores cuya llamada es gestionada por LP_SCHED (máx. 256)
- MAN_CYC[x], x = 1 ... GLP_NBR: el tiempo de muestreo deseado para los distintos reguladores. Tenga en cuenta para cada regulador la condición indicada más abajo para MAN_CYC[x]. En caso contrario no se garantiza el tiempo de muestreo parametrizado.

Las variables siguientes pueden modificarse durante el funcionamiento:

- MAN_DIS[x] sirve para bloquear la llamada del lazo de regulación x durante el funcionamiento.
- MAN_CRST[x] sirve para lanzar un proceso de inicialización para el lazo de regulación x durante el funcionamiento.
  La modificación se hace efectiva la próxima vez que se llame la instrucción LP_SCHED.

Las variables siguientes son escritas por LP_SCHED:

- LP_SCHED introduce la condición de llamada para el regulador x en la variable ENABLE[x].
- Las variables ALP_NBR y ILP_COU[x] son variables internas de la instrucción LP_SCHED. Pueden ser útiles para la observación.

Las variables COM_RST[x] y CYCLE[x] se interconectan con los parámetros de entrada COM_RST y CYCLE de las instrucciones PID_CP y PID_ES.

Tiempo de muestreo MAN_CYC[x]

LP_SCHED puede procesar un regulador como máximo por llamada. Por consiguiente, el tiempo

TM_BASE * GLP_NBR transcurre hasta que se han procesado todos los reguladores una vez. En la parametrización de los tiempos de muestreo deseados MAN_CYC[x] hay que observar la condición siguiente para cada regulador:

El tiempo de muestreo del regulador x debe ser un múltiplo entero del producto de la base de tiempo y el número de los reguladores que deben procesarse:

$$\text{MAN_CYC[x]} \overset{!}{=} \text{GV (TM_BASE * GLP_NBR), x = 1, ... GLP_NBR}$$

El tiempo de muestreo real CYCLE[x] del lazo de regulación x está determinado por LP_SCHED en cada ciclo a partir de MAN_CYC[x], del siguiente modo:

- Si se ha observado la condición anterior, el tiempo de muestreo real CYCLE[x] es idéntico al tiempo de muestreo especificado MAN_CYC[x].
- Si no se ha observado la condición anterior, en CYCLE[x] aparece el valor que resulta de redondear MAN_CYC[x] al próximo múltiplo entero de TM_BASE * GLP_NBR.

Agregar más lazos de regulación

Si desea insertar uno o varios lazos de regulación en el bloque de datos DB_LOOP, aumente el valor inicial de GLP_NBR y amplíe la matriz LOOP_DAT[1...GLP_NBR].

Compruebe el tiempo de muestreo deseado MAN_CYC[x] para cada regulador. Tenga en cuenta la condición para MAN_CYC.

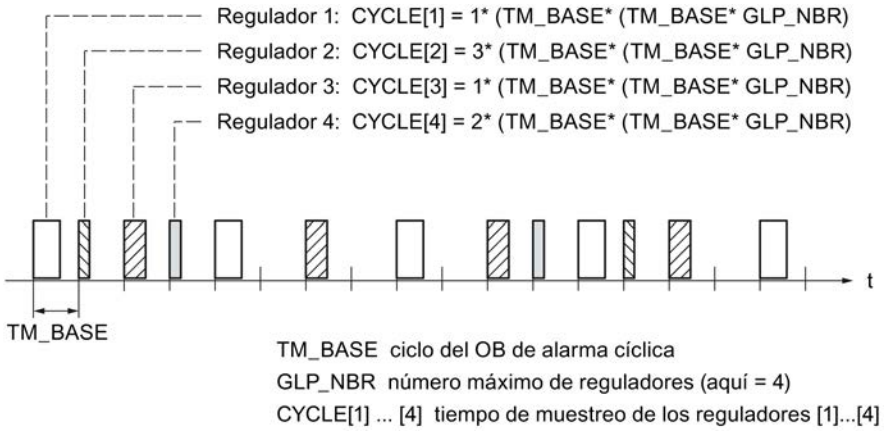
Generador de impulsos en combinación con LP_SCHED

Si en el regulador continuo PID_CP se ha activado el generador de impulsos, en lugar del parámetro CYCLE debe interconectarse el ancho de impulso CYCLE_P con el parámetro LOOP_DAT[x].CYCLE.

Ejemplo de una distribución de llamadas

El siguiente ejemplo muestra el orden de las llamadas de cuatro reguladores en un OB de alarma cíclica. Se procesa como máximo un regulador por unidad de la base de tiempo

TM_BASE. El orden de las llamadas resulta del orden de los reguladores en el bloque de datos DB_LOOP.



Llamada de varios reguladores por ciclo

Si en un ciclo deben procesarse varios reguladores, es posible llamar LP_SCHED varias veces. Todas las llamadas de la instrucción LP_SCHED deben realizarse antes de llamar los reguladores. En el parámetro de entrada TM_BASE hay que introducir el ciclo del OB de alarma cíclica dividido entre el número de llamadas de LP_SCHED.

Ejemplo: LP_SCHED se llama dos veces en el OB 35 y el OB 35 se procesa cada 100 ms. En este caso, el parámetro de entrada TM_BASE debe parametrizarse con 50 ms.

Tiempo de ejecución

La suma de los tiempos de ejecución de la instrucción LP_SCHED y de los reguladores que se procesan en un ciclo no puede ser superior al ciclo del OB de alarma cíclica.

Operaciones durante el funcionamiento

Durante el funcionamiento están permitidas las modificaciones siguientes en el bloque de datos "DB_LOOP", siempre y cuando se modifique solo el parámetro correspondiente y no se cargue todo el bloque de datos en la CPU:

- Bloquear reguladores individuales
  Al escribir la variable MAN_DIS[x] con el valor TRUE se bloquea el procesamiento del regulador x durante el funcionamiento. LP_SCHED no vuelve a poner a TRUE el bit ENABLE de este regulador hasta que no se vuelve a escribir MAN_DIS[x] con el valor FALSE.
- Inicializar un lazo de regulación
  Al escribir la variable MAN_CRST[x] con el valor TRUE es posible reiniciar diferentes reguladores: En este caso, LP_SCHED escribe la variable COM_RST[x] con TRUE en el próximo procesamiento del regulador x. En el segundo procesamiento de este lazo de regulación, LP_SCHED escribe las variables MAN_CRST[x] y COM_RST[x] con FALSE.
- Modificación del tiempo de muestreo de un lazo de regulación
  El parámetro MAN_CYC[x] puede modificarse durante el funcionamiento.

Nota

Si se carga de nuevo todo el bloque de datos DB_LOOP en la CPU sin que la CPU deba realizar un arranque, hay que inicializar con cero los contadores internos de los lazos de regulación ILP_COU[x], x = 1, ... GLP_NBR y el número del lazo de regulación actual ALP_NBR.

Observación de la instrucción LP_SCHED

LP_SCHED introduce el número del siguiente regulador por procesar en la variable ALP_NBR del bloque de datos DB_LOOP. El número del regulador correspondiente resulta del posicionamiento de sus datos de llamada en el orden de las entradas en el bloque de datos.

La variable ILP_COU[x] es el contador interno. Contiene la duración hasta la próxima llamada del regulador en cuestión. La unidad de tiempo de ILP_COU es el producto de la base de tiempo TM_BASE por el número de reguladores GLP_NBR. Si ILP_COU = 0, LP_SCHED activa el bit ENABLE del regulador correspondiente.

Cuando no se llaman lazos de regulación en contra de lo previsto

Cuando se llama LP_SCHED pero no se procesan distintos lazos de regulación, esto puede deberse a las causas siguientes:

- Al escribir la variable MAN_DIS[x] con el valor TRUE se ha bloqueado el procesamiento del lazo de regulación durante el funcionamiento.
- En el parámetro GLP_NBR se ha indicado un número insuficiente de reguladores que deben ser procesados por LP_SCHED.
- Los tiempos de muestreo especificados MAN_CYC[x] para los diferentes lazos de regulación no pueden ser menores que el producto de la base de tiempo TM_BASE por el número de reguladores GLP_NBR. Los reguladores que no cumplen esta condición no se procesan.

Parámetros de entrada de LP_SCHED

Tabla 9-210

Parámetro	Tipo de datos	Rango	Ajuste predeterminado	Explicación
TM_BASE	TIME	≥ 20 ms (S7-300) ≥ 5 ms (S7-400)	100 ms	Base de tiempo (Ciclo del nivel de alarma cíclica en el que se llama LP_SCHED) Son válidos valores
COM_RST	BOOL		FALSE	rearranque (Se procesa la rutina de rearranque del LP_SCHED)
DB_NBR	BLOCK_DB		DB 1	Número del bloque de datos (DB con los datos de llamada de los lazos de regulación)

Bloque de datos global DB_LOOP

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
GLP_NBR	INT	2	Número máximo de reguladores 1 - 256
ALP_NBR	INT	0	Número del regulador actual 0 - 256
LOOP_DAT	Array [1...GLP_NBR] of Struct		Estructura para los datos de los distintos reguladores. La matriz debe ser declarada por 1...GLP_NBR.
LOOP_DAT[1].MAN_CYC	TIME	T#1s	Datos del regulador 1: tiempo de muestreo manual >= 20 ms (S7-300) >= 5 ms (S7-400)
LOOP_DAT[1].MAN_DIS	BOOL	FALSE	Datos del regulador 1: bloquear manualmente la llamada del regulador
LOOP_DAT[1].MAN_CRST	BOOL	FALSE	Datos del regulador 1: iniciar manualmente el rearranque
LOOP_DAT[1].ENABLE	BOOL	FALSE	Datos del regulador 1: habilitación del regulador
LOOP_DAT[1].COM_RST	BOOL	FALSE	Datos del regulador 1: rearranque
LOOP_DAT[1].ILP_COU	INT	0	Datos del regulador 1: contador interno
LOOP_DAT[1].CYCLE	TIME	T#1s	Datos del regulador 1: tiempo de muestreo >= 20 ms (S7-300) >= 5 ms (S7-400)
LOOP_DAT[2].MAN_CYC			Datos del regulador 2: tiempo de muestreo manual
...			

Modular PID Control

General

Convenciones para los nombres de parámetros y bloques en el diagrama de bloques

Para las denominaciones de los parámetros no se han seleccionado más de 8 caracteres.

Se aplican las siguientes convenciones para la denominación de los parámetros:

Primera letra:	
Q	Salida general del tipo BOOL (magnitud booleana)
SP	Referencia
PV	Valor medido o magnitud regulada
LMN	Magnitud manipulada o señal analógica de salida que debe emitirse
DISV	Magnitud perturbadora
Letras sucesivas:	
MAN	Valor manual
INT	Interno
EXT	Externo
_ON	Magnitud BOOL para activar una función

Datos de llamada

La mayoría de las instrucciones de Modular PID Control requiere datos de llamada específicos del lazo de regulación como bit de rearranque y tiempo de muestreo. Estos valores se transmiten a las entradas COM_RST y CYCLE.

Observaciones sobre las barras de parámetros de bloques (parámetros de entrada, salida y entrada/salida)

- **Asignación previa:** aquí se encuentran los valores de ajuste predeterminados al crear una nueva instancia.
- **Rango de valores**: los valores del parámetro de entrada no deben superar el rango admisible. **No** se realiza ninguna comprobación del rango dentro del procesamiento de bloques. El rango técnico se refiere a una magnitud física cuyo valor debería estar entre $\pm 10^{6}$.

A_DEAD_B

Descripción de A_DEAD_B

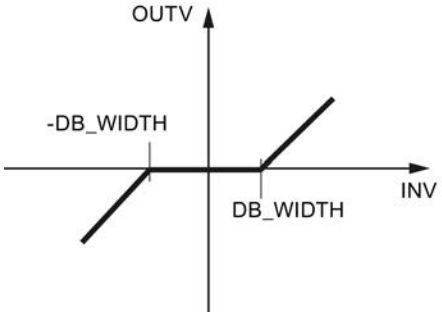
Si el valor real es ruidoso y el regulador está ajustado de forma óptima, el nivel de ruido también afectará la salida del regulador. Esto causa un deterioro más rápido del actuador debido a la frecuencia de conmutación elevada (regulador paso a paso). Al suprimir el nivel de ruido se evita la oscilación de la salida del regulador.

Modo de funcionamiento

La instrucción filtra las señales perturbadoras de alta frecuencia del error de regulación y establece un rango insensible alrededor del punto cero. Cuando la magnitud de entrada está dentro de este rango, en la salida se emite cero. El ancho del rango se adapta automáticamente a la amplitud de la señal de perturbación.

La instrucción funciona según la siguiente función:

OUTV = INV + DB_WIDTH	para	INV < -DB_WIDTH
OUTV = 0.0	para	-DB_WIDTH ≤ INV ≤ +DB_WIDTH
OUTV = INV - DB_WIDTH	para	INV > +DB_WIDTH



Adaptación de la zona muerta

La adaptación de la zona muerta puede activarse y desactivarse.

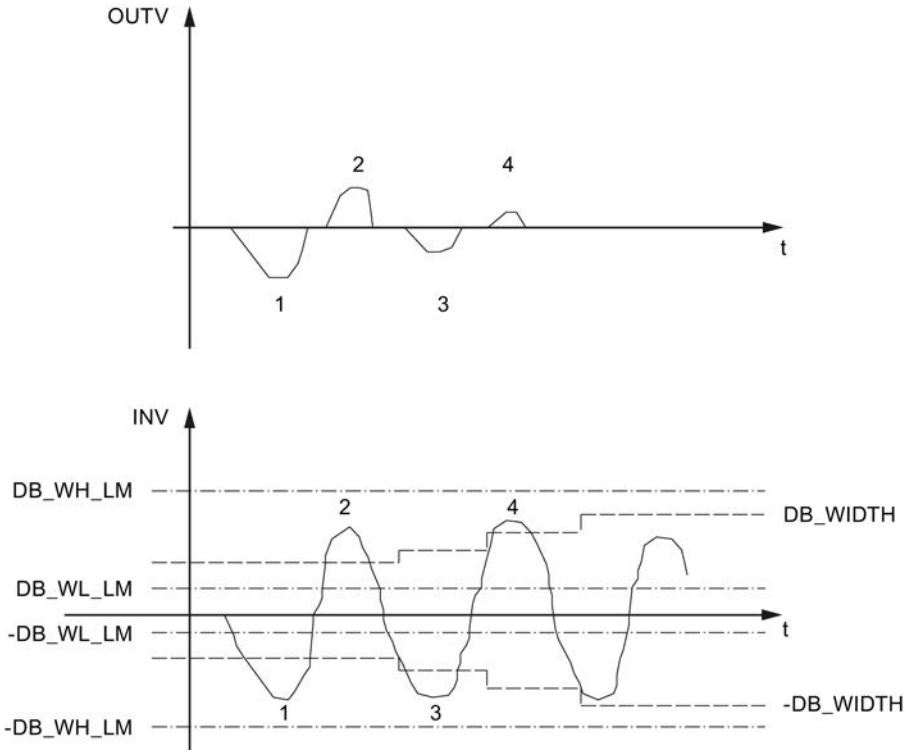
- Adaptación desactivada
  Si la adaptación está desactivada (ADAPT_ON = FALSE), como ancho de zona muerta efectivo DB_WIDTH se congela el último valor DB_WIDTH.
- Adaptación activada
  A través de ADAPT_ON = TRUE se puede aplicar un algoritmo de adaptación, que calcula el ancho efectivo de la zona muerta. Así se adapta el ancho de la zona muerta a la amplitud de la señal de ruido que está sobrepuesta en la magnitud de entrada y, por tanto, se suprime el nivel de ruido también en caso de una amplitud de ruido inconstante.
  Para llamadas del bloque no cíclicas, es necesario desactivar la adaptación (ADAP_ON = FALSE).

Si durante la puesta en marcha se activa la adaptación y si después de cierto tiempo se establece un ancho de la zona muerta estable, la adaptación se puede desactivar. El ancho de la zona muerta ajustada mediante la adaptación se mantiene hasta el próximo rearranque.

Adaptación activada

Por razones de estabilidad se limita el ancho efectivo de la zona muerta DB_WIDTH hacia abajo con el valor DB_WL_LM predeterminable en la barra de entrada. Cuando la señal de entrada INV que está superpuesta por el ruido medido supera el ancho de la zona muerta actualmente ajustada en dirección negativa (1), en dirección positiva (2) y luego otra vez en

dirección negativa (3) dentro de la duración de período 1/CRIT_FRQ, el ancho efectivo de la zona muerta aumenta en 0.1.



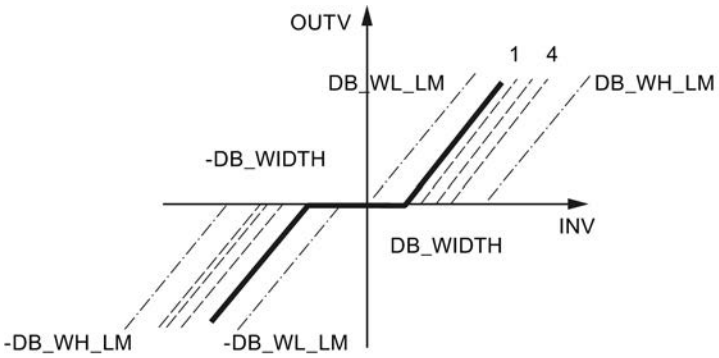
El proceso se inicia en caso de una superación positiva o negativa. Cada vez que se supera (3 -> 4) el ancho opuesto de la zona muerta dentro de la mitad de la duración de período, el ancho de la zona muerta vuelve a aumentar en 0,1. Este proceso se repite tantas veces hasta que el ancho de la zona muerta coincida con la amplitud del ruido de medición. Para no suprimir cualquier señal de entrada alta, se limita por arriba el ancho efectivo de la zona muerta mediante la entrada DB_WH_LM. Si, por el contrario, dentro del tiempo RET_FAC * 1/ CRIT_FRQ no se supera el ancho de la zona muerta, éste se reduce en 0.1.

En ese caso CRIT_FRQ indica la frecuencia límite a partir de la que se interpreta un componente de la señal como perturbación. Se limita por arriba y abajo del siguiente modo:

0.01 ≤ CRIT_FRQ ≤ 1/(3*CYCLE) con CYCLE: Tiempo de muestreo en segundos

El parámetro RET_FAC determina el múltiplo de 1/CRIT_FRQ después de que el ancho de la zona muerta se haya reducido.

La lógica de adaptación solo funciona cuando la magnitud de entrada sin componente de perturbación es similar a cero.



Rearranque

En el rearranque se activan OUTV=0.0 y el ancho efectivo de la zona muerta DB_WIDTH = DB_WL_LM.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Determinar y acondicionar el error de regulación (Página 10173)

Parámetros de entrada de A_DEAD_B

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
INV	REAL	0.0	Magnitud de entrada Rango técnico
DB_WH_LM	REAL	5.0	Límite superior del ancho de la zona muerta Rango técnico DB_WH_LM > DB_WL_LM
DB_WL_LM	REAL	1.0	Límite inferior del ancho de la zona muerta Rango técnico DB_WL_LM < DB_WH_LM
CRIT_FRQ	REAL	0.1	Frecuencia límite CRIT_FRQ ≥ 0.01 y CRIT_FRQ ≤ 1/(3 * CYCLE)
RET_FAC	INT	1	Factor de vuelta RET_FAC ≥ 1
ADAPT_ON	BOOL	FALSE	Activar el algoritmo de adaptación

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	BOOL	FALSE	Rearranque
CYCLE	TIME	T#1s	Tiempo de muestreo CYCLE ≥ 1ms

Parámetros de salida de A_DEAD_B

Parámetro	Tipo de datos	Asignación previa	Descripción
OUTV	REAL	0.0	Magnitud de salida
DB_WIDTH	REAL	0.0	Ancho efectivo de la zona muerta

CRP_IN

Descripción de CRP_IN

La instrucción adapta el rango de la periferia analógica al formato interno de la regulación modular. La instrucción puede llamarse, por ejemplo, en la rama del valor real.

Modo de funcionamiento

CRP_IN convierte un valor de entrada del formato de periferia en un valor en coma flotante normalizado de la regulación modular.

Valor de periferia	Valor de salida en %
32767	118,515
27648	100,000
1	0,003617
0	0,000
-1	-0,003617
-27648	-100,000
-32768	-118,519

Se puede adaptar el valor en coma flotante mediante un factor de escala y un offset. La salida se calcula del siguiente modo:

OUTV = INV_PER * 100/27648 * FACTOR + OFFSET

Nota

No se realiza ninguna comprobación de rebase positivo/negativo.

Utilizar el valor de puesta en marcha

Para la puesta en marcha, con fines de comprobación o en caso de perturbaciones de la periferia es posible cambiar a un valor de puesta en marcha. Si START_ON = TRUE, en la salida OUTV de deposita el valor en STARTVAL.

Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionamiento del valor real (Página 10171)

Control anticipativo (Página 10175)

Acondicionamiento de la realimentación de posición (Página 10184)

Parámetros de entrada CRP_IN

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV_PER	0.0	WORD	0	Magnitud de entrada para periferia Rango técnico
FACTOR	2.0	REAL	1.0	Factor de escala
OFFSET	6.0	REAL	0.0	Offset Rango técnico
START_ON	10.0	BOOL	TRUE	Activar el valor de puesta en marcha
STARTVAL	12.0	REAL	0.0	Valor de puesta en marcha Rango técnico

Parámetros de salida de CRP_IN

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	16.0	REAL	0.0	Magnitud de salida

CRP_OUT

Descripción de CRP_OUT

La instrucción adapta un valor en coma flotante de la regulación modular al formato de periferia.

Modo de funcionamiento

CRP_OUT convierte un valor de entrada (valor en coma flotante normalizado de la regulación modular) en el formato de la periferia analógica.

Valor de entrada en %	Valor de periferia
118,515	32767
100,000	27648
0,003617	1
0,000	0
-0,003617	-1
-100,000	-27648
-118,519	-32768

Se puede adaptar el valor en coma flotante mediante un factor de escala y un offset. La salida se calcula del siguiente modo:

OUTV_PER = (INV * FACTOR + OFFSET) * 27648/100

Nota

No se realiza ninguna comprobación de rebase positivo/negativo.

Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Crear un regulador continuo (Página 10178)

Parámetros de entrada de CRP_OUT

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
FACTOR	4.0	REAL	1.0	Factor de escala
OFFSET	8.0	REAL	0.0	Offset Rango técnico

Parámetros de salida CRP_OUT

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV_PER	12.0	WORD	0	Magnitud de salida para periferia

DEAD_T

Descripción de DEAD_T

La instrucción se utiliza para las regulaciones de relación en las que se unen los componentes individuales en vías de transporte de diferentes longitudes.

Modo de funcionamiento

La instrucción retarda la emisión de un valor de entrada en un tiempo ajustable (tiempo muerto). Cuando se modifica el tiempo muerto online pueden producirse escalones en el valor de salida.

Los valores de entrada se guardan temporalmente en un bloque de datos global.

El bloque de datos global debe contener un Array [0..n_{max}] of real.

n_{max} resulta del tiempo muerto DEAD_TM y el tiempo de muestreo CYCLE de la instrucción DEAD_T. Debe darse la siguiente condición:

- DEAD_TM debe ser un múltiplo entero de CYCLE.
- n_{max}=DEAD_TM/CYCLE

En el ciclo n se lee el valor de entrada guardado INV[n] del bloque de datos global y se deposita en OUTV. INV[n] se sobrescribe con el valor de entrada actual. El contador n aumenta en 1. Con n = n_{max} el contador se vuelve a poner a 0. El número del bloque global se introduce en el parámetro de entrada DB_NBR.

Ejemplo

Con un tiempo de muestreo de CYCLE=1s y un tiempo muerto de DEAD_TM=5s los cinco valores de entrada deben almacenarse. El rango de datos debe ser de 20 bytes.

Los datos contenidos en el bloque de datos global se procesan como en un búfer circular.

N.º de ciclo	Contador n	INV en la instrucción	Variable en el bloque de datos global	Valor de INVI[n] en el inicio del ciclo	OUTV en la instrucción	Valor de INVI [n] en el final del ciclo
0	0	80.0	INVI[0]	0.0	0.0	80.0
1	1	100.0	INVI[1]	0.0	0.0	100.0
2	2	100.0	INVI[2]	0.0	0.0	100.0
3	3	80.0	INVI[3]	0.0	0.0	80.0
4	4	0.0	INVI[4]	0.0	0.0	0.0
5	0	0.0	INVI[0]	80.0	80.0	0.0
6	1	0.0	INVI[1]	100.0	100.0	0.0
7	2	0.0	INVI[2]	100.0	100.0	0.0
8	3	10.0	INVI[3]	80.0	80.0	10.0
9	4	100.0	INVI[4]	0.0	0.0	100.0

El valor de INV en el ciclo 0 no se emite hasta el ciclo 5 en OUTV. Con un tiempo de muestreo de 1 s, es decir, retardado el tiempo correspondiente al tiempo muerto de 5 s.

Corrección

Con TRACK=TRUE el valor de entrada se emite sin retardo en OUTV. El almacenamiento temporal de los valores de entrada no se interrumpe con el fin de que después de desactivar la corrección se puedan emitir los valores de entrada después del tiempo muerto ajustado. Con un flanco negativo en TRACK, OUTV salta a INV[DEAD_TM].

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Nota

La instrucción no comprueba si existe realmente un DB global con el número DB_NBR ni si los parámetros DEAD_TM (tiempo muerto) y CYCLE (tiempo de muestreo) coinciden con la longitud del bloque de datos. Si no se ha realizado la parametrización correctamente, la CPU cambia al estado de funcionamiento STOP mostrando el mensaje "Error interno del sistema".

Rearranque

Durante el rearranque se borran todos los valores de entrada guardados y se emite OUTV=0.0.

Parámetros de entrada de DEAD_T

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
DB_NBR	4.0	BLOCK_DB	DB 1	Número del bloque de datos
DEAD_TM	6.0	TIME	10s	Tiempo muerto DEAD_TM ≥ CYCLE
TRACK	10.0	BOOL	FALSE	Corrección OUTV = INV
COM_RST	10.1	BOOL	FALSE	Rearranque
CYCLE	12.0	TIME	1s	Tiempo de muestreo CYCLE ≥ 1ms

Parámetros de salida de DEAD_T

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	16.0	REAL	0.0	Magnitud de salida

Bloque de datos global DEAD_T

Parámetro	Tipo de datos	Asignación previa	Descripción
INVI	Array[0..n_{max}] of real	0.0	Valor de entrada [n] Rango técnico

DEADBAND

Descripción de DEADBAND

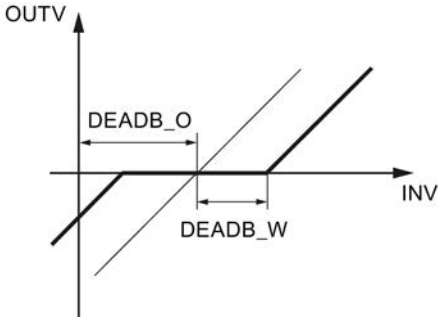
Si el valor real es ruidoso y el regulador está ajustado de forma óptima, el nivel de ruido también afectará la salida del regulador. Esto causa un deterioro más rápido del actuador debido a la frecuencia de conmutación elevada (regulador paso a paso). Al suprimir el nivel de ruido se evita la oscilación de la salida del regulador.

Modo de funcionamiento

La instrucción DEADBAND (zona muerta) suprime pequeñas oscilaciones alrededor de un punto cero definido en la magnitud de entrada INV; fuera de dichas oscilaciones, la magnitud de salida OUTV aumenta proporcionalmente a la magnitud de entrada. Funciona según la siguiente función:

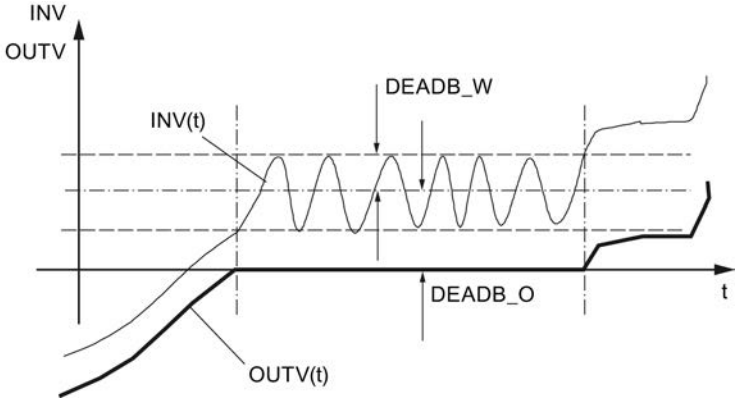
OUTV = INV + DEADB_W - DEADB_O	para	INV < DEADB_O - DEADB_W
OUTV = 0.0	para	DEADB_O - DEADB_W ≤ INV
	y	INV ≤ DEADB_O + DEADB_W
OUTV = INV - DEADB_W - DEADB_O	para	INV > DEADB_O + DEADB_W

La falsificación de señal se corresponde con el valor DEADB_W. El punto central de la zona muerta se define con DEADB_O.



El ancho de la zona muerta DEADB_W y el desplazamiento de la zona muerta DEADB_O son parametrizables. En caso de utilización para la formación del error de regulación debe ajustarse el offset DEADB_O=0.0.

La siguiente figura muestra la supresión de un ruido con offset.



Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros. El ancho de zona muerta sólo debe tener valores positivos.

Consulte también

Determinar y acondicionar el error de regulación (Página 10173)

Parámetros de entrada de DEADBAND

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
DEADB_W	4.0	REAL	1.0	Ancho de zona muerta Rango técnico y ≥ 0.0
DEADB_O	8.0	REAL	0.0	Desplazamiento de la zona muerta Rango técnico

Parámetros de salida de DEADBAND

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	12.0	REAL	0.0	Magnitud de salida

DIF

Descripción de DIF

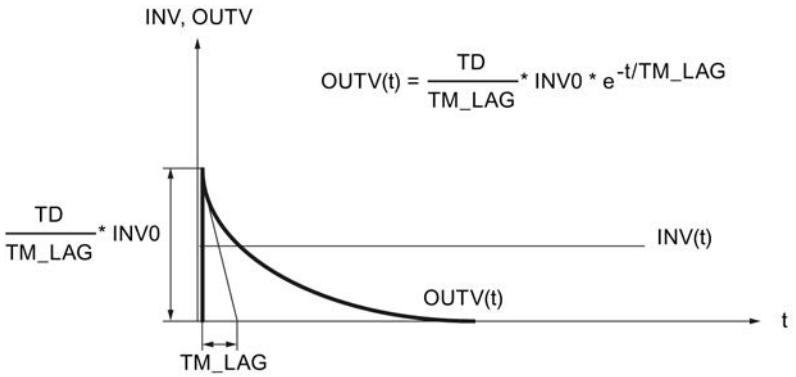
Las magnitudes de proceso se diferencian dinámicamente. De este modo se puede por ejemplo calcular la velocidad a partir del trayecto recorrido. El diferenciador se puede utilizar para el control anticipativo, la regulación previa y el diseño de un regulador.

Modo de funcionamiento

La instrucción distingue la magnitud de entrada por el tiempo, filtra la señal con un elemento de retardo de primer orden y trabaja en el rango de Laplace según la siguiente función de transmisión.

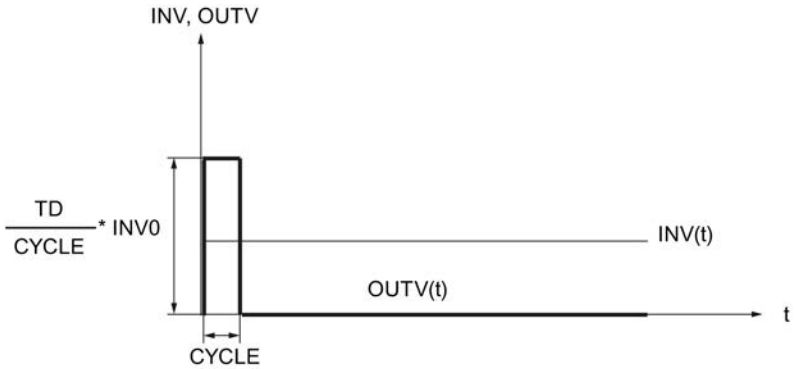
- OUTV(s) / INV(s) = TD / (1+TM_LAG*s)

El comportamiento de tiempo se determina a través del tiempo de acción derivativa TD y el tiempo de retardo TM_LAG. Las siguientes figuras muestran la respuesta indicial de DIF.



TD	Tiempo de acción derivativa
TM_LAG	Tiempo de retardo
INV0	Amplitud del escalón de entrada
INV	Magnitud de entrada
OUTV	Magnitud de salida

Si está parametrizado TM_LAG ≤CYCLE/2, DIF trabaja sin retardo. En la salida se aplica un escalón de entrada con el factor TD/CYCLE. Después de un ciclo la salida vuelve a 0.0.



Límites internos del bloque

El tiempo de acción derivativa se limita al tiempo de muestreo por abajo. El tiempo de retardo se limita a la mitad del tiempo de muestreo por abajo.

TD_{intern} = CYCLE	para	TD < CYCLE
TM_LAG_{intern} = CYCLE/2	para	TM_LAG < CYCLE/2

Los valores restantes de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Rearranque

Durante el rearranque todas las salidas de señal se ponen a cero. Al diferenciador se le asigna de forma predeterminada e interna el valor de entrada actual de INV. De este modo, se pasa de forma suave al modo normal con una magnitud de entrada constante.

Consulte también

Control anticipativo (Página 10175)

Parámetros de entrada de DIF

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
TD	4.0	TIME	T#25s	Tiempo de acción derivativa ≥ CYCLE
TM_LAG	8.0	TIME	T#5s	Tiempo de retardo
COM_RST	12.0	BOOL	FALSE	Rearranque
CYCLE	14.0	TIME	T#1s	Tiempo de muestreo ≥ 1ms

Parámetros de salida de DIF

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	18.0	REAL	0.0	Magnitud de salida

ERR_MON

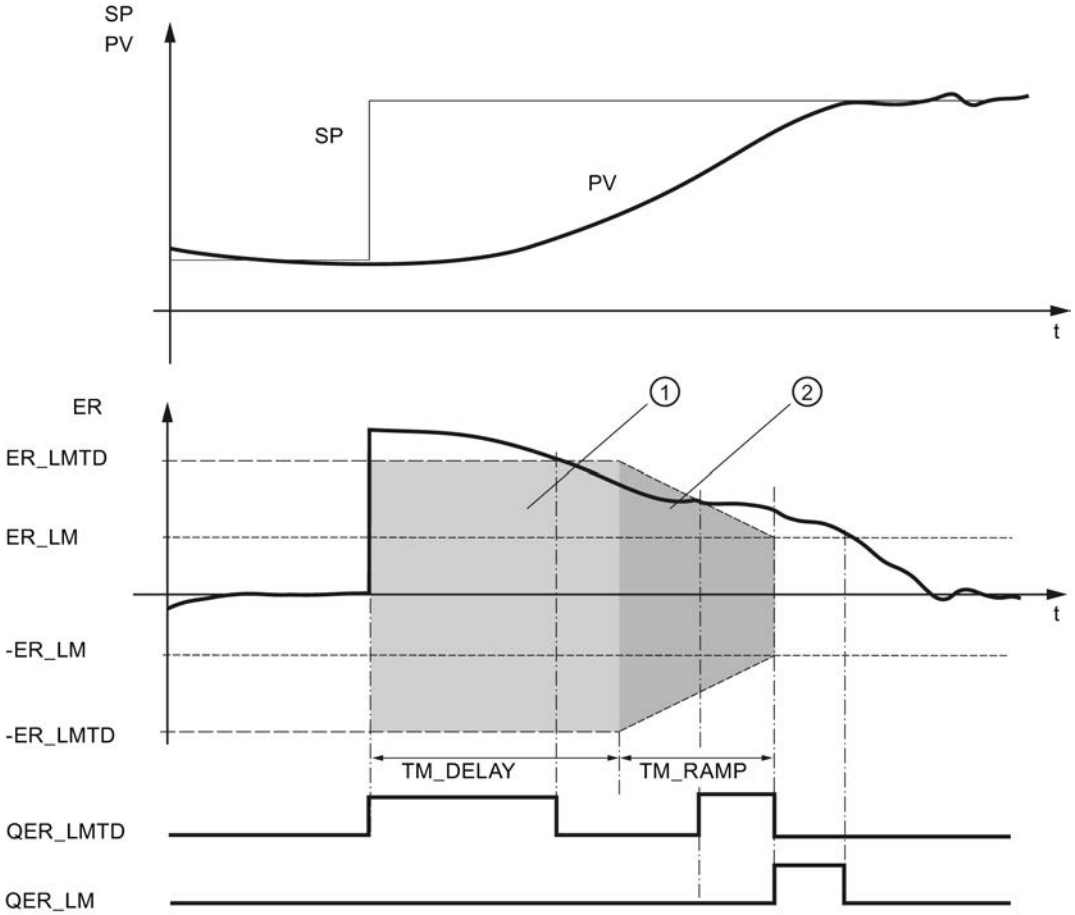
Descripción de ERR_MON

La instrucción se utiliza para crear y vigilar el error de regulación.

Modo de funcionamiento

La instrucción calcula el error de regulación ER = SP - PV y vigila si éste alcanza los límites parametrizables. Con un cambio de consigna |ΔSP| > SP_DIFF se suprime la activación de la señal de valor límite ER_LM por un tiempo de retardo ajustable (TM_DELAY+TM_RAMP). Durante ese tiempo, ER se vigila con el valor de límite superior ER_LMTD. Si se rebasa ER_LMTD, se emite QER_LMTD=TRUE. Una vez transcurrido el tiempo de retardo,

ER_LMTD se convierte en ER_LM con una rampa. La pendiente de la rampa se puede ajustar mediante el parámetro TM_RAMP.



PV:	Valor real	TM_DELAY:	Retardo de conexión de la señal de monitorización
SP:	Consigna		
ER:	Error de regulación	TM_RAMP:	Constante de tiempo de rampa
ER_LM:	Valor límite del error de regulación	QER_LM:	El valor límite del error de regulación ha reaccionado
ER_LMTD:	Valor límite del error de regulación durante el retardo de conexión	QER_LMTD:	El valor límite del error de regulación ha reaccionado durante el retardo de conexión y rampa
①	Supresión de la señal de valor límite	②	Función de rampa

Rearranque

Durante el rearranque se desactivan las señales de aviso QER_LM y QER_LMTD y la salida de error de regulación ER.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Determinar y acondicionar el error de regulación (Página 10173)

Parámetros de entrada de ERR_MON

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
ER_LM	0.0	REAL	10.0	Valor límite del error de regulación Rango técnico y > 0.0 y < ER_LMTD
ER_LMTD	4.0	REAL	100.0	Valor límite del error de regulación durante el retardo de conexión Rango técnico y > ER_LM
SP	8.0	REAL	0.0	Consigna Rango técnico
PV	12.0	REAL	0.0	Magnitud de regulación Rango técnico
SP_DIFF	16.0	REAL	10.0	Cambio de consigna Rango técnico y > 0.0
TM_DELAY	20.0	TIME	T#60s	Retardo de conexión de la señal de monitorización
TM_RAMP	24.0	TIME	T#60s	Constante de tiempo de rampa
COM_RST	28.0	BOOL	FALSE	Rearranque
CYCLE	30.0	TIME	T#1s	Tiempo de muestreo ≥ 1ms

Parámetros de salida de ERR_MON

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QER_LM	34.0	BOOL	FALSE	El valor límite del error de regulación ha reaccionado
QER_LMTD	34.1	BOOL	FALSE	El valor límite del error de regulación ha reaccionado durante el retardo de conexión y rampa
ER	36.0	REAL	0.0	Error de regulación

INTEG

Descripción de INTEG

Las magnitudes de proceso se integran dinámicamente. Así se calcula por ejemplo el trayecto recorrido a partir de la velocidad. La instrucción se puede utilizar para diseñar un regulador con acción I.

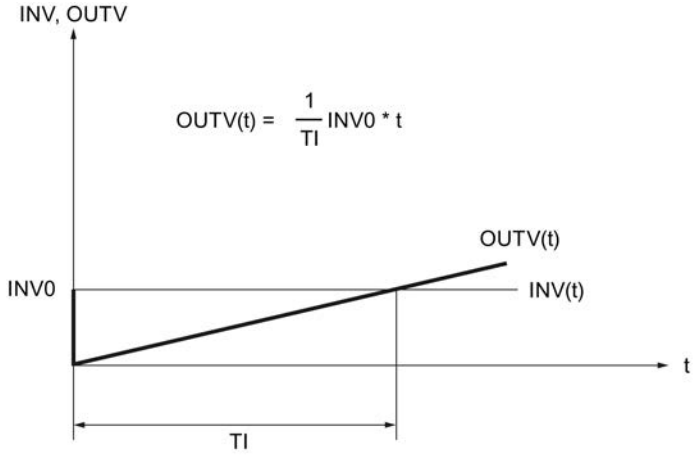
Modo de funcionamiento

La instrucción integra la magnitud de entrada por el tiempo y limita la integral a un valor superior y otro inferior predeterminables. El límite de la magnitud de salida se indica a través de bits de aviso.

La instrucción contiene las siguientes funciones:

Funciones	DFOUT_ON	HOLD
Integrar	FALSE	FALSE
Congelar la acción I	FALSE	TRUE
Preasignar salida	TRUE	cualquiera

- Integrar
  Al integrar, la instrucción funciona con la siguiente función de transmisión:
  En el rango de Laplace:
  OUTV(s) / INV(s) = 1 / (TI*s)
  El comportamiento temporal de la acción I se determina mediante el tiempo de integración TI. La respuesta indicial correspondiente se muestra en la siguiente figura.

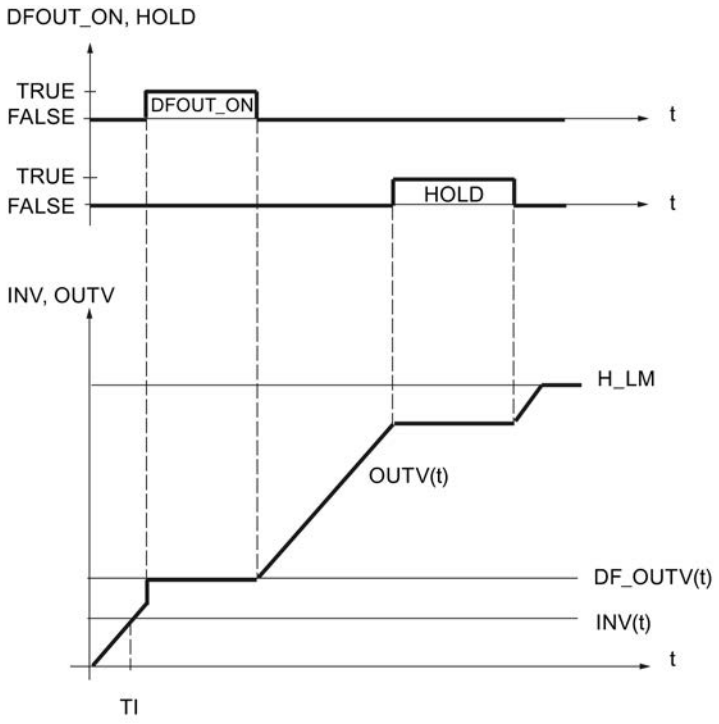


T1:	Tiempo de integración
INV0:	Amplitud del escalón de entrada
t:	Tiempo
INV:	Magnitud de entrada
OUTV:	Magnitud de salida

La salida y la memoria de la acción I se limitan en los parámetros H_LM y L_LM. Si la salida se encuentra fuera del rango L_LM...H_LM , los bits de aviso QH_LM o QL_LM pasan a TRUE.

- Congelar el integrador
  Con HOLD = TRUE la acción I permanece en su valor de salida actual OUTV. Al desactivar HOLD = FALSE se sigue integrando a partir del valor de salida actual OUTV.
- Preasignar salida
  Si está activado DFOUT_ON = TRUE, en la salida se deposita DF_OUTV. El límite es efectivo. Si DF_OUTV_ON = FALSE se desactiva, INTEG comienza a integrar a partir del valor DF_OUTV.

La siguiente figura muestra un ejemplo con DFOUT_ON, HOLD y limitación



Rearranque

En el rearranque la salida OUTV se pone a 0.0. Si está activado DFOUT_ON = TRUE, se emite DF_OUTV. La limitación de la salida y su indicación también están activados en el rearranque. Al cambiar al modo normal, la instrucción continúa integrando desde OUTV.

Si se desea iniciar la acción I desde un punto de trabajo determinado en el rearranque, es necesario introducir el punto de trabajo DF_OUTV en la entada. Al llamar la instrucción en la rutina de rearranque es necesario activar DFOUT_ON= TRUE y volver a poner DFOUT_ON= FALSE en el nivel de alarma cíclica.

Límites internos del bloque

El tiempo de integración se limita al tiempo de muestreo por abajo:

TI_{intern} = CYCLE para TI < CYCLE

Los valores restantes de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada de INTEG

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
H_LM	4.0	REAL	100.0	Límite superior Rango técnico H_LM > L_LM
L_LM	8.0	REAL	0.0	Límite inferior Rango técnico L_HM < H_LM
TI	12.0	TIME	T#25s	Tiempo de integración TI ≥ CYCLE
DF_OUTV	16.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
HOLD	20.0	BOOL	FALSE	Congelar la acción I
DFOUT_ON	20.1	BOOL	FALSE	Activar la preasignación de la magnitud de salida
COM_RST	20.2	BOOL	FALSE	Rearranque
CYCLE	22.0	TIME	T#1s	Tiempo de muestreo CYCLE ≥ 1ms

Parámetros de salida de INTEG

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	26.0	REAL	0.0	Magnitud de salida
QH_LM	30.0	BOOL	FALSE	El límite superior ha reaccionado
QL_LM	30.1	BOOL	FALSE	El límite inferior ha reaccionado

LAG1ST

Descripción de LAG1ST

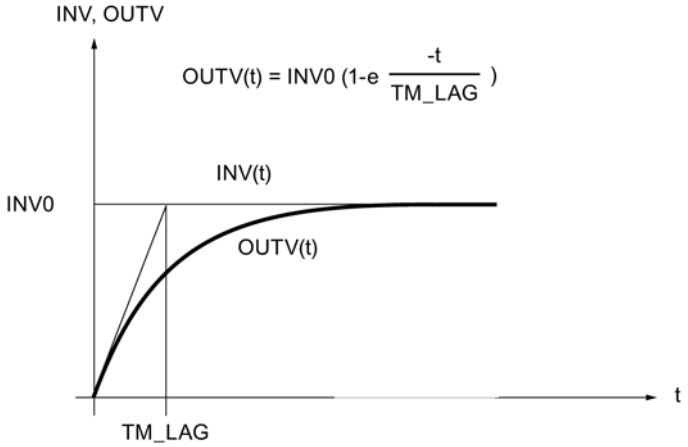
La instrucción se utiliza como elemento de retardo y filtrado de 1er orden. El tiempo de retardo se puede parametrizar.

Modo de funcionamiento

La instrucción filtra la magnitud de entrada después del retardo de primer orden y contiene las siguientes funciones:

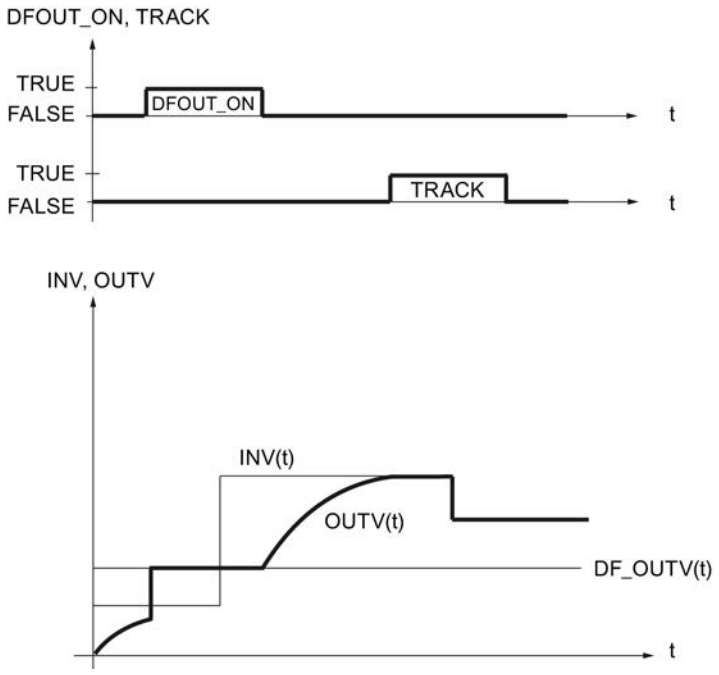
Funciones	DFOUT_ON	TRACK
Filtrar	FALSE	FALSE
Corrección	FALSE	TRUE
Preasignar salida	TRUE	Cualquiera

- **Filtrar**
  Al filtrar, la instrucción funciona en el rango de Laplace según la siguiente función de transmisión:
  OUTV(s) / INV(s) = 1 / (1+TM_LAG*s)
  El comportamiento de tiempo del elemento de retardo se determina a través del tiempo de retardo TM_LAG. La respuesta indicial correspondiente se muestra en la siguiente figura.



INV	Magnitud de entrada
OUTV	Magnitud de salida
INV0	Amplitud del escalón de entrada
TM_LAG	Tiempo de retardo

- **Corrección**
  Con TRACK = TRUE el valor de entrada INV se aplica en la salida OUTV.
- **Preasignar salida**
  Si está activado DFOUT_ON = TRUE, en la salida se deposita DF_OUTV. Si se desactiva DF_OUTV_ON = FALSE, el elemento de retardo filtra a partir del valor DF_OUTV.
  La siguiente figura muestra el efecto de DFOUT_ON y TRACK:



Rearranque

En el rearranque la salida OUTV se pone a 0.0. Si está activado DFOUT_ON = TRUE, se emite DF_OUTV. Al pasar a filtrado, la instrucción continúa a partir de OUTV.

Límites internos del bloque

El tiempo de retardo se limita en la mitad del tiempo de muestreo por abajo.

TM_LAG_{intern} = CYCLE/**2** para TM_LAG < CYCLE/**2**

Los valores restantes de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)

Acondicionamiento del valor real (Página 10171)

Control anticipativo (Página 10175)

Parámetros de entrada de LAG1ST

Parámetro	Offset	Tipo de datos	Ajuste predeterminado	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
TM_LAG	4.0	TIME	T#25s	Tiempo de retardo
DF_OUTV	8.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
TRACK	12.0	BOOL	FALSE	Corrección OUTV=INV
DFOUT_ON	12.1	BOOL	FALSE	Activar la preasignación de la magnitud de salida
COM_RST	12.2	BOOL	FALSE	Rearranque
CYCLE	14.0	TIME	T#1s	Tiempo de muestreo ≥1ms

Parámetros de salida de LAG1ST

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	18.0	REAL	0.0	Magnitud de salida

LAG2ND

Descripción de LAG2ND

La instrucción sirve para simular partes del sistema para la regulación previa y la regulación de dos lazos.

Modo de funcionamiento

La instrucción implementa un elemento de retardo de 2º orden con capacidad de oscilación y contiene las siguientes funciones:

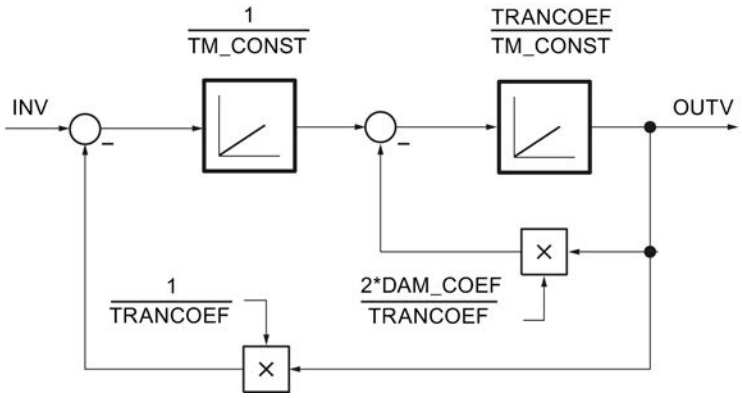
Funciones	DFOUT_ON	TRACK
Transferir	FALSE	FALSE
Corrección	FALSE	TRUE
Preasignar salida	TRUE	cualquiera

- **Transferir**
  La función de transmisión en el rango de Laplace es la siguiente:
  OUTV(s) / INV(s) = TRANCOEF / (1 + 2*DAM_COEF*TM_CONST*s + TM_CONST^2*s2)
  Si DAM_COEF >= 1 (caso aperiódico), se puede representar el elemento de transmisión en una conexión en serie de dos elementos PT1.
  OUTV(s) / INV(s) = TRANCOEF / (1 + T1*s) * 1 / (1 + T2 * s)
  Las constantes temporales se convierte del siguiente modo:

  $$T1 = TM_CONST\ (DAM_COEF + \sqrt{DAM_COEF^2 - 1}\ )$$

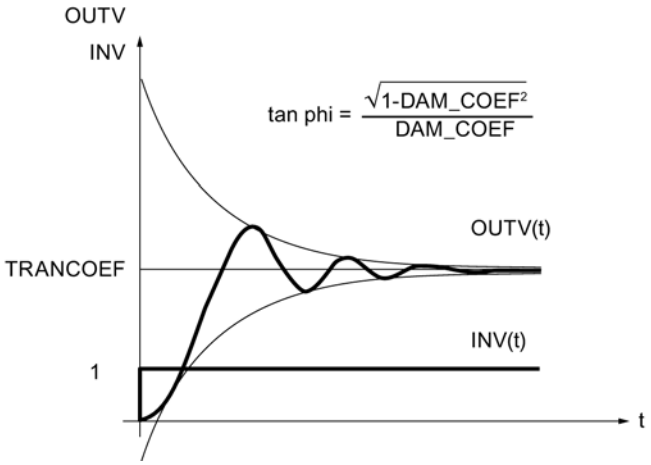
  $$T2 = TM_CONST\ (DAM_COEF - \sqrt{DAM_COEF^2 - 1}\ )$$

  La siguiente figura muestra el diagrama de bloques.



  La figura siguiente muestra la respuesta indicial del elemento LAG2ND con capacidad de oscilación:

$$OUTV(t) = TRANCOEF + \frac{TRANCOEF}{\sqrt{1-DAM_COEF^2}} \cdot e^{-\frac{DAM_COEF}{TM_CONST} t} \sin\left(\frac{\sqrt{1-DAM_COEF^2}}{TM_CONST} t - phi\right)$$



INV	Magnitud de entrada
OUTV	Magnitud de salida
TRANCOEF	Constante de transmisión
DAM_COEF	Constante de atenuación
phi	Ángulo

- **Corrección**
  Si TRACK =TRUE, entonces OUTV = INV; los valores internos pasados se ajustan a INV.
- **Preasignar salida**
  Si DFOUT_ON =TRUE, la salida es DF_OUTV, los valores pasados internos se ajustan a DF_OUTV. DFOUT_ON tiene mayor prioridad que TRACK.

Rearranque

En el rearranque la salida OUTV se pone a 0.0. Si DFOUT_ON = TRUE, la salida es DF_OUTV.

Límites internos del bloque

La constante de tiempo TM_CONST se limita por abajo a la mitad del tiempo de muestreo.

$TM_CONST_{intern} = CYCLE/2$ para $TM_CONST < CYCLE/2$

Los valores restantes de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada LAG2ND

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
TM_CONST	4.0	TIME	T#10s	Constante temporal
DAM_COEF	8.0	REAL	1.0	Constante de atenuación
TRANCOEF	12.0	REAL	1.0	Constante de transmisión
DF_OUTV	16.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
DFOUT_ON	20.0	BOOL	FALSE	Activar la preasignación de la magnitud de salida
TRACK	20.1	BOOL	FALSE	Corrección OUTV=INV
COM_RST	20.2	BOOL	FALSE	Rearranque
CYCLE	22.0	TIME	T#1s	Tiempo de muestreo ≥ 1ms

Parámetros de salida de LAG2ND

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	26.0	REAL	0.0	Magnitud de salida

LIMALARM

Descripción de LIMALARM

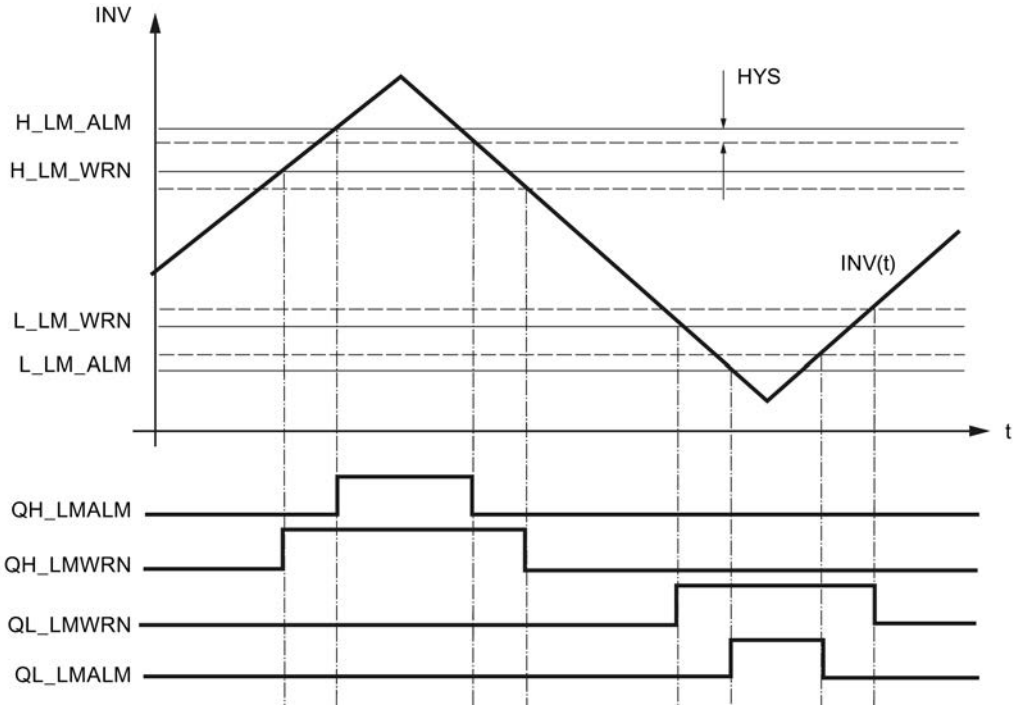
Durante la regulación pueden producirse estados de funcionamiento no permitidos o peligrosos si se sobrepasan o no se alcanzan las magnitudes de proceso (p. ej., revoluciones, presión, temperatura, ...) En caso de sobrepasar o no alcanzar los límites admisibles, se debe recopilar lo mismo y comunicarlo para que pueda suceder una reacción eficiente.

La instrucción comprueba la magnitud de entrada INV en 4 límites parametrizables. Al alcanzar y sobrepasar estos límites se emite una señal de límite correspondiente. El umbral de desconexión de las señales de límite se puede ajustar a traves de una histéresis.

Modo de funcionamiento

Los límites se ajustan en las entradas H_LM_ALM, H_LM_WRN, L_LM_WRN y L_LM_ALM. Si la magnitud de entrada INV rebasa los límites, se activan los bits de aviso QH_LMALM, QH_LMWRN, QL_LMWRN y QL_LMALM. Para evitar una activación y desactivación rápida de los bits de aviso, el valor de entrada tiene que superar además una histéresis HYS antes de que se desactiven las salidas.

La figura siguiente muestra el funcionamiento de la instrucción LIMALARM:



QH_LMALM = TRUE	si	INV aumenta	y	INV >= H_LM_ALM
	o	INV disminuye	y	INV >= H_LM_ALM - HYS
QH_LMWRN = TRUE	si	INV aumenta	y	INV >= H_LM_WRN
	o bien	INV disminuye	y	INV >= H_LM_WRN - HYS
QL_LMWRN = TRUE	si	INV disminuye	y	INV <= L_LM_WRN
	o bien	INV aumenta	y	INV <= L_LM_WRN + HYS
QL_LMALM = TRUE	si	INV disminuye	y	INV <= L_LM_ALM
	o bien	INV aumenta	y	INV <= L_LM_ALM + HYS

Para que la instrucción funcione correctamente debe cumplirse lo siguiente:

L_LM_ALM < L_LM_WRN < H_LM_WRN < H_LM_ALM

Rearranque

Durante el rearranque todas las salidas de señal se ponen en FALSE.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionamiento del valor real (Página 10171)

Parámetros de entrada de LIMALARM

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
H_LM_ALM	0.0	REAL	100.0	Límite superior de Alarma Rango técnico > H_LM_WRN
H_LM_WRN	4.0	REAL	90.0	Límite superior de Advertencia Rango técnico > L_LM_WRN
L_LM_WRN	8.0	REAL	10.0	Límite inferior de Advertencia Rango técnico > L_LM_ALM
L_LM_ALM	12.0	REAL	0.0	Límite inferior de Alarma Rango técnico < L_LM_WRN
INV	16.0	REAL	0.0	Magnitud de entrada Rango técnico
HYS	20.0	REAL	1.0	Histéresis Rango técnico
COM_RST	24.0	BOOL	FALSE	Rearranque

Parámetros de salida de LIMALARM

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QH_LMALM	26.0	BOOL	FALSE	Límite superior de Alarma ha reaccionado
QH_LMWRN	26.1	BOOL	FALSE	Límite superior de Advertencia ha reaccionado
QL_LMWRN	26.2	BOOL	FALSE	Límite inferior de Advertencia ha reaccionado
QL_LMALM	26.3	BOOL	FALSE	Límite inferior de Alarma ha reaccionado

LIMITER

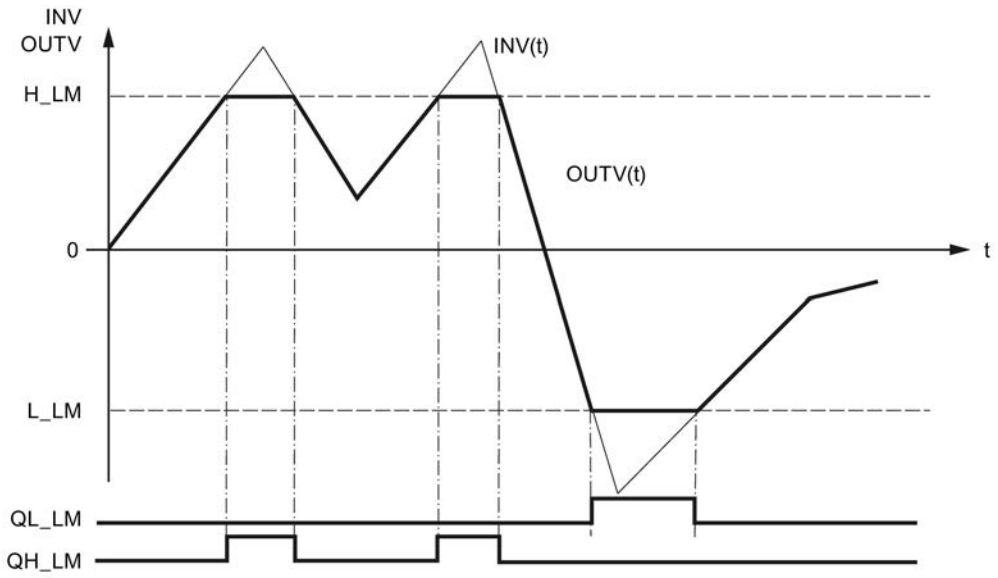
Descripción de LIMITER

En caso de una predeterminación dinámica de los parámetros (p. ej., el cálculo de consignas a partir de magnitudes de proceso), éstos pueden tener valores no admisibles para el proceso. Con la instrucción LIMITER se pueden mantener en el rango admisible.

Modo de funcionamiento

La instrucción limita la magnitud de salida OUTV a un límite superior e inferior configurables H_LM y L_LM, si la magnitud de entrada INV está fuera de dichos límites. La limitación de OUTV se notifica a través de las salidas QH_LM y QL_LM.

La figura siguiente muestra el funcionamiento de la instrucción LIMITER:



INV	OUTV	QH_LM	QL_LM
≥ H_LM	H_LM,	TRUE	FALSE
≤ L_LM	L_LM,	FALSE	TRUE
≤ L_LM y ≤H_LM	INV	FALSE	FALSE

Para que la instrucción funcione correctamente debe cumplirse lo siguiente:

L_LM < H_LM

Rearranque

Durante el rearranque todas las salidas de señal se ponen en FALSE; en la salida OUTV se ajusta 0.0.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)

Acondicionamiento del valor real (Página 10171)

Parámetros de entada LIMITER

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
H_LM	4.0	REAL	100.0	límite superior Rango técnico > L_LM
L_LM	8.0	REAL	0.0	límite inferior Rango técnico < H_LM
COM_RST	12.0	BOOL	FALSE	Rearranque

Parámetros de salida de LIMITER

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	14.0	REAL	0.0	Magnitud de salida
QH_LM	18.0	BOOL	FALSE	límite superior ha reaccionado
QL_LM	18.1	BOOL	FALSE	límite inferior ha reaccionado

LMNGEN_C

Descripción de LMNGEN_C

La instrucción permite configurar un regulador PID continuo y determina la variable manipulada del regulador. La variable manipulada puede determinarse en el modo automático o preasignarse en el modo manual. El valor manual se puede predeterminar de forma absoluta, o puede ampliarse y reducirse a través del modo de conmutador. La variable manipulada y el aumento de la variable manipulada se pueden limitar a valores especificables.

Rearranque

Durante el rearranque el valor del ajuste predeterminado DF_OUTV se conecta directamente con la salida LMN, independientemente del bit del ajuste predeterminado DFOUT_ON. El límite de la salida y su indicación también están activados tras un rearranque. Al cambiar al modo normal, la instrucción sigue funcionado desde DF_OUTV.

Los parámetros de salida QLMN_HLM y QLMN_LLM se ponen a FALSE.

Llamada

Mientras la instrucción PID se llama en un nivel de alarma cíclica, cuyo tiempo de ciclo está adaptado a la constante de tiempo dominante en el proceso, la instrucción LMNGEN_C puede llamarse para intervenciones manuales en un nivel de alarma cíclica más rápido. La interconexión se realiza con los parámetros de entrada y salida estructurados PID_LMNG y LMNG_PID.

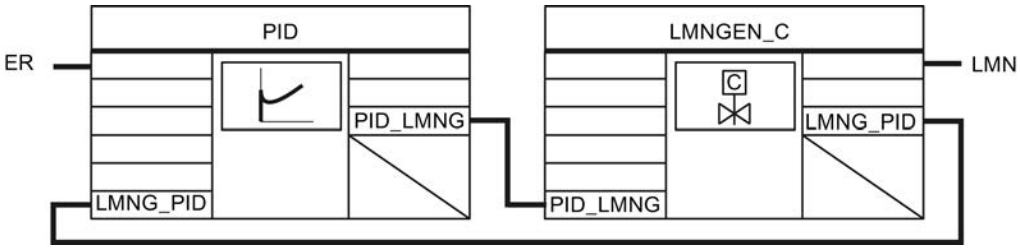
Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Modo de funcionamiento

La instrucción LMNGEN_C siempre se utiliza en combinación con la instrucción PID.

La siguiente figura muestra la interconexión del regulador PID continuo.



Consulte también

Crear un regulador continuo (Página 10178)

Crear un regulador de impulsos (Página 10179)

Funcionamiento de LMNGEN_C

Limitar variable manipulada

La variable manipulada se limita en todos los modos de operación. Los límites superiores de la variable manipulada se parametrizan en la entrada LMN_HLM, y el límite inferior en la entrada LMN_LLM. Si la variable manipulada calculada rebasa esos valores por exceso o defecto, en LMN se deposita el límite superior o inferior. La limitación de la variable manipulada

al límite superior se indica en la salida QLMN_HLM, mientras que la limitación al límite inferior se indica en la salida QLMN_LLM.

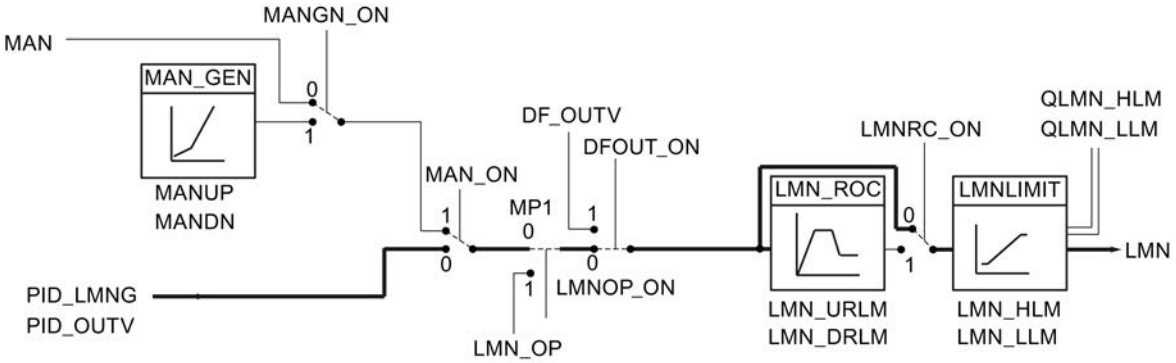
Rampa de la variable manipulada

Con la rampa de la variable manipulada se limita la velocidad de cambio de la variable manipulada. La rampa de la variable manipulada se activa con LMNRC_ON=TRUE. En el parámetro de entrada LMN_URLM se introduce la velocidad de cambio máxima para la rampa de subida de la variable manipulada y en el parámetro de entrada LMN_DRLM la velocidad de cambio máxima para la rampa de bajada de la variable manipulada.

Modos de operación

Los modos de operación se activan y desactivan con los parámetros DFOUT_ON, MAN_ON y MANGN_ON.

Modos de operación	DFOUT_ON	MAN_ON	MANGN_ON
Modo automático	FALSE	FALSE	FALSE
Modo manual	FALSE	TRUE	FALSE
Generador de valor manual	FALSE	TRUE	TRUE
Especificar variable manipulada	TRUE	no aplicable	no aplicable

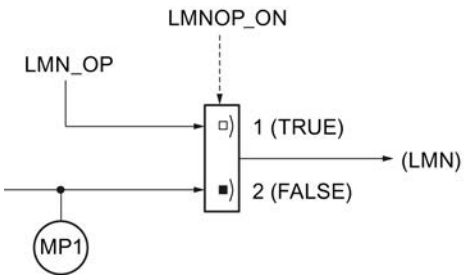


- Modo automático
  El valor determinado con el algoritmo PID se utiliza para determinar la variable manipulada LMN. La conmutación al modo automático es suave si la rampa de la variable manipulada (LMNRC_ON = TRUE) está activada.
- Modo manual
  Con MAN_ON = TRUE se pasa al modo manual, interrumpiendo así el lazo de regulación. El valor manual MAN se emite directamente como variable manipulada LMN. MAN debe estar dentro de los límites LMN_HLM y LMN_LLM.

- **Generador de valor manual**
  Si además MANGN_ON = TRUE, la variable manipulada puede aumentarse o reducirse mediante los conmutadores MANUP y MANDN. La velocidad de cambio de la variable manipulada depende de los límites:
  - Durante los primeros 3 segundos después de activar MANUP o MANDN
    dLMN/dt = (LMN_HLM - LMN_LLM) / 100 s
  - a continuación
    dLMN/dt = (LMN_HLM - LMN_LLM) / 10 s
- **Especificar variable manipulada**
  Con DFOUT_ON = TRUE el valor preasignado DF_OUTV se aplica en la salida LMN. La conmutación de o a "Especificar variable manipulada" sólo es suave si está activada la rampa de la variable manipulada. (LMNRC_ON= TRUE).

Control a través de HMI

Con el uso de variables estáticas se puede influir en una variable manipulada a través de un HMI (véase la figura siguiente). Las variables estáticas LMN_OP, LMNOP_ON y MP1 pueden, por ejemplo, manejarse y observarse a través de HMI.



Si LMNOP_ON se pone a TRUE, LMN_OP se pasa a la variable manipulada LMN. MP1 contiene la variable manipulada que se procesa en la instrucción LMNGEN_C.

Si la rampa de la variable manipulada está activada, se puede conmutar suavemente entre los valores MP1 y LMN_OP. LMN se devuelve, en función del ajuste de la rampa, al valor MP1.

Las variables estáticas modificadas no tienen efecto en el proceso hasta que han sido transferidas a la CPU.

Parámetros de entrada LMNGEN_C

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
LMN_HLM	0.0	REAL	100.0	Límite superior de la variable manipulada Rango técnico > LMN_LLM
LMN_LLM	4.0	REAL	0.0	Límite inferior de la variable manipulada Rango técnico < LMN_HLM
LMN_URLM	8.0	REAL	10.0	Límite de subida de la variable manipulada [1/s] > 0.0
LMN_DRLM	12.0	REAL	10.0	Límite de bajada de la variable manipulada [1/s] > 0.0
DF_OUTV	16.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
MAN_ON	20.0	BOOL	TRUE	Activar el modo manual
MANGN_ON	20.1	BOOL	FALSE	Activar modo de valor manual
MANUP	20.2	BOOL	FALSE	Valor manual alto
MANDN	20.3	BOOL	FALSE	Valor manual bajo
LMNRC_ON	20.4	BOOL	FALSE	Activar rampa de la variable manipulada
DFOUT_ON	20.5	BOOL	FALSE	Activar la preasignación de la magnitud de salida
COM_RST	20.6	BOOL	FALSE	Rearranque
CYCLE	22.0	TIME	T#1s	Tiempo de muestreo ≥ 1ms
PID_LMNG	26.0	STRUC		PID-LMNGEN interfaz

Parámetros de salida de LMNGEN_C

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
LMN	44.0	REAL	0.0	Variable manipulada
QLMN_HLM	48.0	BOOL	FALSE	Límite superior de la variable manipulada ha reaccionado
QLMN_LLM	48.1	BOOL	FALSE	Límite inferior de la variable manipulada ha reaccionado
LMNG_PID	50.0	STRUC		PID-LMNGEN interfaz

Parámetros de entrada/salida de LMNGEN_C

Tabla 9-211

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
MAN	68.0	REAL	0.0	Valor manual Rango técnico

Variables estáticas LMNGEN_C

Parámetro	Offset	Tipo de datos	Ajuste predetermina do	Descripción
MP1	72.0	REAL	0.0	Variable manipulada procesada por LMNGEN_C.
LMM_OP	76.0	REAL	0.0	Control a través de HMI Valor manual
LMNOP_ON	80.0	BOOL	FALSE	Control a través de HMI Activar el modo manual

LMNGEN_S

Descripción de LMNGEN_S

La instrucción permite configurar un regulador paso a paso PID para actuadores con comportamiento integral (p. ej., válvulas accionadas por motor) y determinar la variable manipulada del regulador. La variable manipulada puede determinarse en el modo automático o preasignarse en el modo manual. El valor manual se puede predeterminar de forma absoluta, o puede ampliarse y reducirse a través del modo de conmutador. La variable manipulada también puede limitarse a valores parametrizables. El regulador paso a paso puede funcionar con o sin realimentación de posición.

Rearranque

Durante el rearranque todas las salidas de señal se ponen en cero.

Llamada

Mientras la instrucción PID se llama en un nivel de alarma cíclica, cuyo tiempo de ciclo está adaptado a la constante de tiempo dominante en el proceso, la instrucción LMNGEN_S puede llamarse para intervenciones manuales en un nivel de alarma cíclica más rápido. La interconexión se realiza con los parámetros de entrada y salida estructurados PID_LMNG y LMNG_PID.

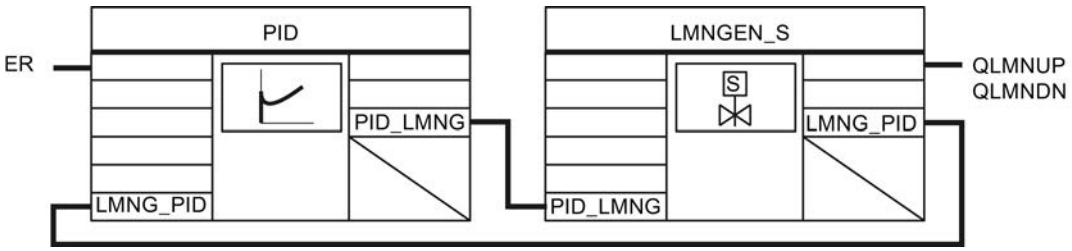
Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Modo de funcionamiento

La instrucción LMNGEN_S siempre se utiliza en combinación con la instrucción PID.

La siguiente figura muestra la interconexión de un regulador paso a paso.



Consulte también

Crear un regulador paso a paso (Página 10183)

Funcionamiento de LMNGEN_S

Limitar variable manipulada

La variable manipulada se limita en todos los modos de operación. Los límites superiores de la variable manipulada se parametrizan en la entrada LMN_HLM, y el límite inferior en la entrada LMN_LLM. Si la variable manipulada calculada rebasa esos valores por exceso o defecto, en LMN se deposita el límite superior o inferior. La limitación de la variable manipulada al límite superior se indica en la salida QLMN_HLM, mientras que la limitación al límite inferior se indica en la salida QLMN_LLM.

Señales de tope

En los parámetros de entrada LMNR_HS y LMNR_LS se aplican las señales para el tope superior e inferior del actuador. Si LMNR_HS= TRUE, el actuador no continúa abriéndose. Si LMNR_LS= TRUE, el actuador no continúa cerrándose. Las señales de salida QLMNUP y QLMNDN están bloqueadas. Los topes finales se tienen en cuenta en todos los modos de operación.

⚠ ADVERTENCIA

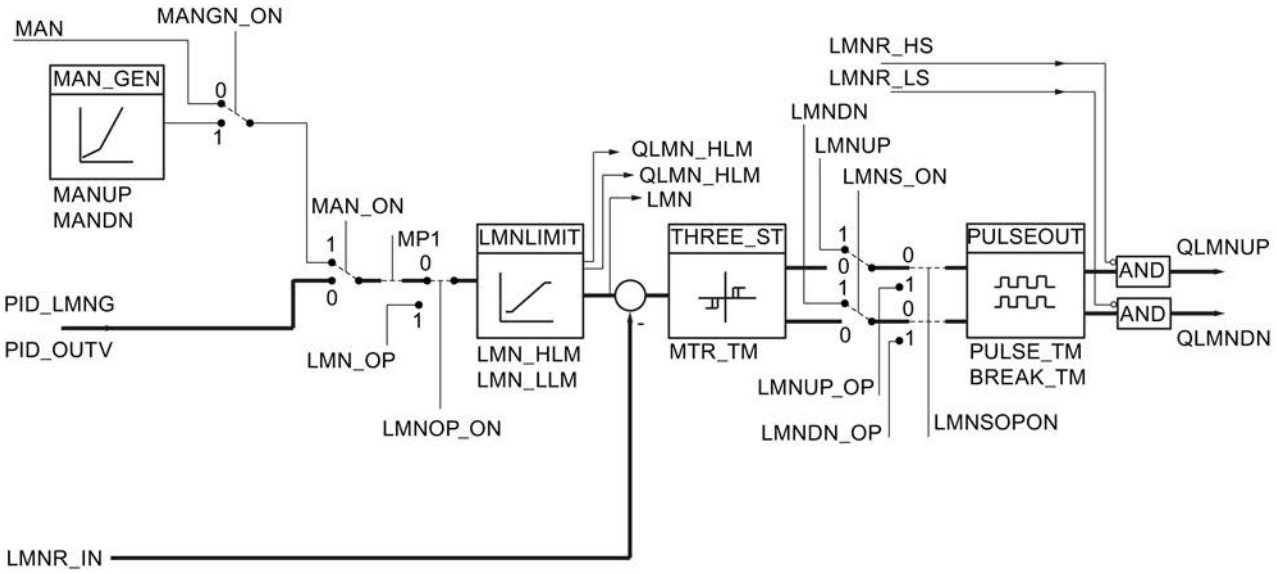
Si no hay señales de tope, el regulador no puede reconocer un tope de la válvula y es posible que, por ejemplo, el regulador emita señales para abrir la válvula, aunque la válvula haya llegado hasta el tope superior. Si no hay señales de tope, las entradas LMNR_HS y LMNR_LS deben ser FALSE.

Regulador paso a paso con realimentación de posición

Si hay disponible una realimentación de posición, la instrucción forma los impulsos para accionar la válvula de control a partir de la diferencia entre la variable manipulada y la realimentación de posición mediante un elemento de tres puntos y un generador de impulsos. Mediante una adaptación del umbral de respuesta del elemento de tres niveles se reduce la frecuencia de conmutación del regulador.

La realimentación de posición se activa con LMNR_ON = TRUE.

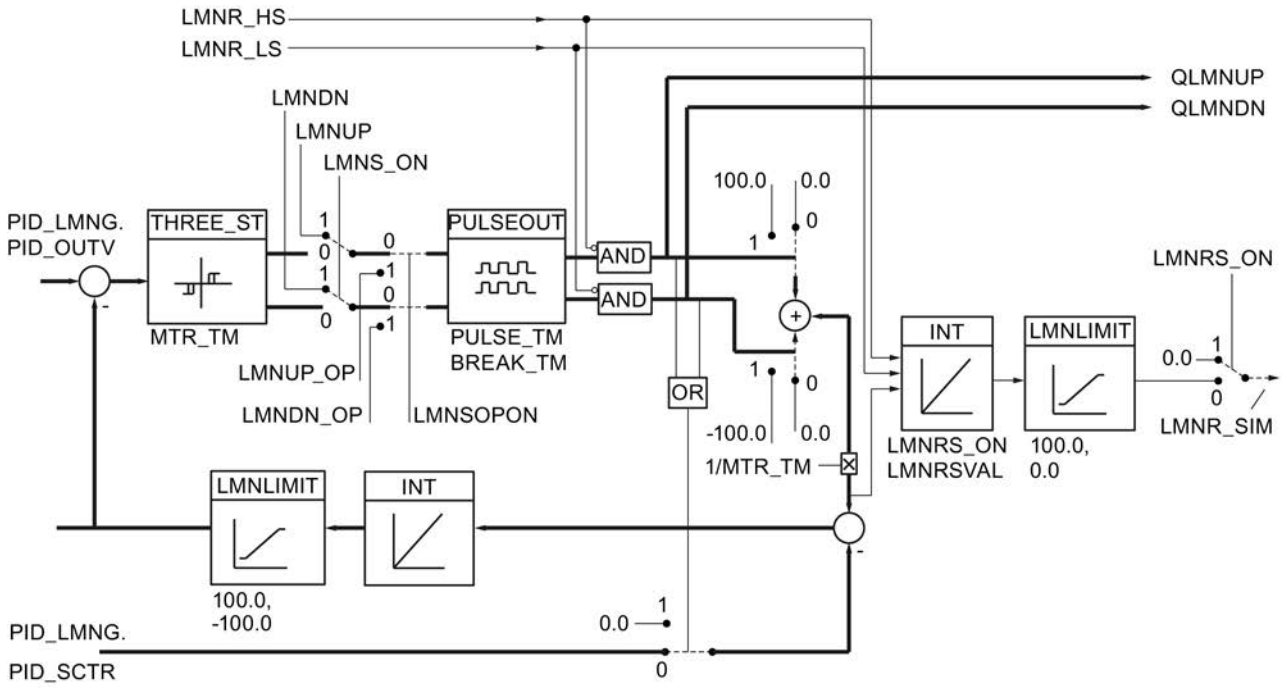
La



Regulador paso a paso sin realimentación de posición

Si no hay ninguna realimentación de posición disponible, la instrucción suma la diferencia entre la acción I del algoritmo PID y una realimentación de posición simulada en un integrador y la compara con la acción PD residual como valor de realimentación. La diferencia pasa al elemento de tres puntos y al generador de impulsos, que genera los impulsos para la válvula de control. Mediante una adaptación del umbral de respuesta del elemento de tres puntos se reduce la frecuencia de conmutación del regulador.

La realimentación de posición se desactiva con LMNR_ON = FALSE.



Modos de operación

Los modos de operación se activan y desactivan con los parámetros LMN_ON, MAN_ON y MANGN_ON.

Modos de operación	LMN_ON	MAN_ON	MANGN_ON
Modo automático	FALSE	FALSE	FALSE
Preasignar valor manual	FALSE	TRUE	FALSE
Generador de valor manual	FALSE	TRUE	TRUE
Modo manual de las señales de salida de la variable manipulada	TRUE	no aplicable	no aplicable

Los modos de operación Preasignar valor manual y Generador de valor manual solo son posibles en reguladores paso a paso con realimentación de posición. Dado que en un regulador paso a paso sin realimentación de posición no hay información sobre la posición de la válvula, el actuador no puede llevarse a un valor manual fijo.

- Preasignar valor manual
  Con MAN_ON = TRUE se pasa al modo manual, interrumpiendo así el lazo de regulación. El valor manual MAN se transmite directamente como variable manipulada. MAN debe estar dentro de los límites LMN_HLM y LMN_LLM.
- Generador de valor manual
  Si además MANGN_ON = TRUE, la variable manipulada puede aumentarse o reducirse mediante los conmutadores MANUP y MANDN. La velocidad de cambio de la variable manipulada depende de los límites:
  - Durante los primeros 3 segundos después de activar MANUP o MANDN
    dLMN/dt = (LMN_HLM - LMN_LLM) / 100 s
  - a continuación
    dLMN/dt = (LMN_HLM - LMN_LLM) / 10 s
- Modo manual de las señales de salida de la variable manipulada
  Si LMNS_ON = TRUE, se puede influir directamente en las señales binarias de salida. Con los conmutadores LMNUP y LMNDN se activan las salidas de señal manipulada QLMNUP y QLMNDN. El tiempo mínimo de impulso PULSE_TM y la duración mínima de pausa BREAK_TM también se tienen en cuenta.
- Modo automático con realimentación de posición
  El valor enviado por el algoritmo PID PID_LMNG.PID_OUTV se limita a valores parametrizables (LMN_HLM y LMN_LLM). La diferencia entre la variable manipulada y la realimentación de posición se conecta a un elemento de tres puntos con histéresis. El umbral de desconexión se calcula a partir del tiempo de posicionamiento del motor. Para reducir la frecuencia de conmutación se adapta el umbral de respuesta. El generador de impulsos PULSEOUT garantiza que se cumplan la duración mínima de impulso y la duración mínima de pausa.
- Modo automático sin realimentación de posición
  La diferencia entre la acción I del algoritmo PID y una realimentación de posición simulada se suma al integrador INT. La entrada del integrador es la diferencia de 100.0/0.0/ -100.0 (dependiendo del estado de las señales de salida QLMNUP y QLMNDN), referido al tiempo de actuación del motor MTR_TM y al error de regulación multiplicado por la ganancia del regulador, referido al tiempo de integración TI del algoritmo PID (PID_LMNG.PID_SCTR). La salida del integrador crea el valor de realimentación, que se compara con la acción PD residual del algoritmo PID (PID_LMNG.PID_OUTV). La diferencia se conecta a un elemento de tres puntos con histéresis. El umbral de desconexión se calcula a partir del tiempo de posicionamiento del motor.
  Para reducir la frecuencia de conmutación se adapta el umbral de respuesta. El generador de impulsos PULSEOUT garantiza que se cumpla la duración mínima de impulso y de pausa.

Simulación de la realimentación de posición

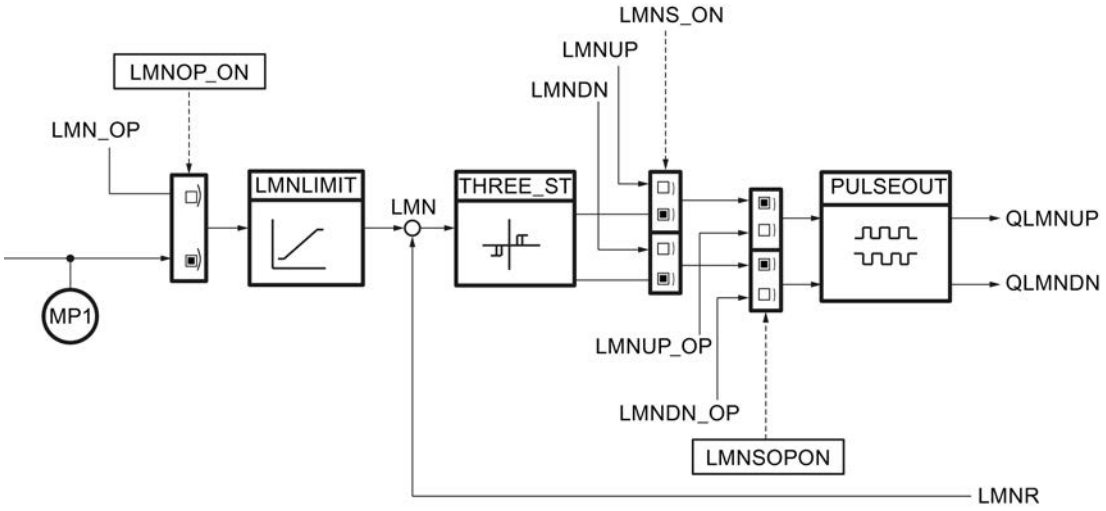
⚠ ADVERTENCIA
La respuesta de posicionamiento solamente se simula. No tiene que coincidir con la realimentación de posición real de la válvula de control. Si existe una realimentación de posición real, se debería usar.

La simulación de la realimentación de posición se realiza a través de un integrador usando el tiempo de actuación del motor MTR_TM como constante de tiempo de integración. Con un flanco positivo en LMNRS_ON se inicia la simulación con el valor inicial LMNRSVAL. Si LMNRS_ON es FALSE, en el parámetro LMNR_SIM se deposita el valor inicial LMNRSVAL.

Las señales de tope LMNR_HS = TRUE y LMNR_LS = TRUE bloquean la integración hacia arriba y abajo.

Control a través de HMI

Con el uso de variables estáticas se puede interrumpir la rama de la variable manipulada y preasignar variables manipuladas propias a través de un HMI (véase la figura siguiente).



Si LMNOP_ON se pone a TRUE, LMN_OP se pasa a la variable manipulada LMN. MP1 contiene la variable manipulada que se procesa en la instrucción LMNGEN_S.

Si la rampa de la variable manipulada está activada, se puede conmutar suavemente entre los valores MP1 y LMN_OP. LMN se devuelve, en función del ajuste de la rampa, al valor MP1.

Si LMNSOPON es TRUE, la variable manipulada se puede aumentar con LMNUP_OP o reducir con LMNDN_OP. Esto es aplicable para reguladores paso a paso con y sin realimentación de posición.

Las variables estáticas LMNOP_ON, LMN_OP, MP1, LMNSOPON, LMNUP_OP y LMNDN_OP no pueden interconectarse y solo pueden manejarse y visualizarse con HMI.

Las variables estáticas modificadas no tienen efecto en el proceso hasta que han sido transferidas a la CPU.

Parámetros de entrada de LMNGEN_S

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
LMN_HLM	0.0	REAL	100.0	Límite superior de la variable manipulada Rango técnico > LMN_LLM
LMN_LLM	4.0	REAL	0.0	Límite inferior de la variable manipulada Rango técnico < LMN_HLM
LMNR_IN	8.0	REAL	0.0	Realimentación de posición Rango técnico
MTR_TM	12.0	TIME	T#30s	Tiempo de posicionamiento del motor ≥ CYCLE
PULSE_TM	16.0	TIME	T#3s	Duración mínima de impulso ≥ CYCLE
BREAK_TM	20.0	TIME	T#3s	Duración mínima de pausa ≥ CYCLE
MAN_ON	24.0	BOOL	TRUE	Activar el modo manual
MANGN_ON	24.1	BOOL	FALSE	Activar modo de valor manual
MANUP	24.2	BOOL	FALSE	Valor manual alto
MANDN	24.3	BOOL	FALSE	Valor manual bajo
LMNR_ON	24.4	BOOL	FALSE	Activar la realimentación de posición
LMNR_HS	24.5	BOOL	FALSE	Señal de tope superior de la realimentación de posición
LMNR_LS	24.6	BOOL	FALSE	Señal de tope inferior de la realimentación de posición
LMNS_ON	24.7	BOOL	FALSE	Activar el modo manual de las señales de la variable manipulada
LMNUP	25.0	BOOL	FALSE	Señal alta de la variable manipulada
LMNDN	25.1	BOOL	FALSE	Señal baja de la variable manipulada
COM_RST	25.2	BOOL	FALSE	Rearranque
CYCLE	26.0	TIME	T#1s	Tiempo de muestreo ≥ 1ms
PID_LMNG	30.0	STRUC		PID-LMNGEN interfaz

Parámetros de salida de LMNGEN_S

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
LMN	48.0	REAL	0.0	Variable manipulada
QLMNUP	52.0	BOOL	FALSE	Señal alta de la variable manipulada
QLMNDN	52.1	BOOL	FALSE	Señal baja de la variable manipulada

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QLMN_HLM	52.2	BOOL	FALSE	El límite superior de la variable manipulada ha reaccionado
QLMN_LLM	52.3	BOOL	FALSE	El límite inferior de la variable manipulada ha reaccionado
LMNG_PID	54.0	STRUC		PID-LMNGEN interfaz

Parámetros de entrada/salida de LMNGEN_S

Tabla 9-212

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
MAN	72.0	REAL	0.0	Valor manual Rango técnico

Variables estáticas LMNGEN_S

No se deben modificar las variables que no aparecen listadas. Estas solo se utilizan internamente.

Parámetro	Offset	Tipo de datos	Ajuste predeterminado	Descripción
LMNSOPON	76.0	BOOL	FALSE	Control a través de HMI Activar el modo manual de las señales de salida de la variable manipulada
LMNUP_OP	76.1	BOOL	FALSE	Control a través de HMI Variable manipulada mayor
LMNDN_OP	76.2	BOOL	FALSE	Control a través de HMI Variable manipulada menor
LMNOP_ON	76.3	BOOL	FALSE	Control a través de HMI Activar el modo manual
LMNRS_ON	76.4	BOOL	FALSE	Activar la realimentación de posición simulada
MP1	78.0	REAL	0.0	Control a través de HMI Variable manipulada procesada por LMNGEN_C.
LMM_OP	82.0	REAL	0.0	Control a través de HMI Valor manual
LMNRSVAL	86.0	REAL	0.0	Valor inicial de la realimentación de posición simulada
LMNR_SIM	9.0	REAL	0.0	Realimentación de posición simulada

NONLIN

Descripción de NONLIN

La instrucción NONLIN permite adaptar un valor de entrada, p. ej. el valor medido de un termopar, mediante una curva característica (polilínea) predeterminable.

Modo de funcionamiento

NONLIN interpola a la magnitud de entrada actual INV la magnitud de salida correspondiente OUTV a partir de nodos en una polilínea. Los nodos de la polilínea se guardan en un bloque de datos global.

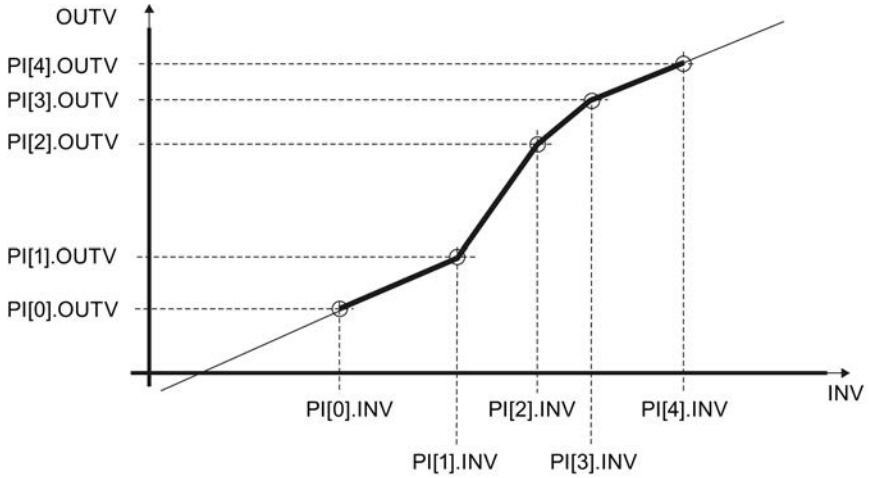
Nota

La instrucción no comprueba si existe realmente un DB global con el número DB_NBR ni si el parámetro DB_NBR. NBR_PTS (número de nodos) concuerda con la longitud del bloque de datos. Si no se ha realizado la parametrización correctamente, la CPU cambia al estado de funcionamiento STOP mostrando el mensaje "Error interno del sistema".

NONLIN dispone de las siguientes funciones:

Funciones	DFOUT_ON	TRACK	OUTV
Interpolación	FALSE	FALSE	OUTV = f(INV)
Preasignar salida	TRUE	cualquiera	DF_OUTV
Corrección	FALSE	TRUE	OUTV = INV

- Interpolación
  La siguiente figura muestra cómo se determina la magnitud de salida OUTV a partir de la magnitud de entrada INV.
  OUTV = f(INV)



  Si el valor de entrada está por debajo del nodo 0, se extrapola con el aumento del par de nodos 0/1. Si el valor está por encima del último nodo, se extrapola con el aumento del último par de nodos. Para que la instrucción dé un resultado eficiente, los valores de los nodos deben ser estrictamente crecientes en orden ascendente.

- Preasignar salida
  Si está ajustado DFOUT_ON = TRUE, en la salida se deposita DF_OUTV, el cambio de OUTV es brusco. Al conmutar a DFOUT_ON = FALSE, el cambio de OUTV también es brusco.

- Corrección
  Con TRACK=TRUE el valor de entrada se emite directamente (OUTV=INV). Igual que con DFOUT_ON, el cambio es brusco. TRACK tiene menor prioridad que DFOUT_ON.

Rearranque

En caso de rearranque, la salida es OUTV = 0.0. Si DFOUT_ON=TRUE, la salida es DF_OUTV.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)

Acondicionamiento del valor real (Página 10171)

Control anticipativo (Página 10175)

Acondicionamiento de la realimentación de posición (Página 10184)

Parámetros de entrada de NONLIN

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
DF_OUTV	4.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
DB_NBR	8.0	BLOCK_DB	DB 1	Número del bloque de datos
DFOUT_ON	10.0	BOOL	FALSE	Activar la preasignación de la magnitud de salida
TRACK	10.1	BOOL	FALSE	Corrección OUTV=INV
COM_RST	10.2	BOOL	FALSE	Rearranque

Parámetros de salida de NONLIN

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	12.0	REAL	0.0	Magnitud de salida

Bloque de datos global DB_NBR

Parámetro	Tipo de datos	Asignación previa	Descripción
DB_NBR.NBR_PTS	INT	4	Número de nodo más alto Rango 1 - 255
DB_NBR.PI[0].OUTV	REAL	0.0	Valor de salida [0] punto inicial 0 Rango técnico
DB_NBR.PI[0].INV	REAL	0.0	Valor de entrada [0] punto inicial 0 Rango técnico
DB_NBR.PI[1].OUTV	REAL	0.0	Valor de salida [1] nodo 1 Rango técnico
DB_NBR.PI[1].INV	REAL	0.0	Valor de entrada [1] nodo 1 Rango técnico
DB_NBR.PI[2].OUTV	REAL	0.0	Valor de salida [2] nodo 2 Rango técnico
DB_NBR.PI[2].INV	REAL	0.0	Valor de entrada [2] nodo 2 Rango técnico
DB_NBR.PI[3].OUTV	REAL	0.0	Valor de salida [3] nodo 3 Rango técnico
DB_NBR.PI[3].INV	REAL	0.0	Valor de entrada [3] nodo 3 Rango técnico

Parámetro	Tipo de datos	Asignación previa	Descripción
DB_NBR.PI[4].OUTV	REAL	0.0	Valor de salida [4] nodo 4 Rango técnico
DB_NBR.PI[4].INV	REAL	0.0	Valor de entrada [4] nodo 4 Rango técnico

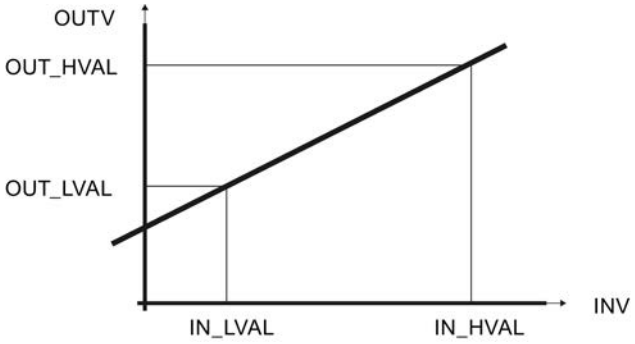
NORM

Descripción NORM

En los valores reales el valor del generador muchas veces está en un rango que no resulta adecuado para el usuario (p. ej., 0 a 10 V corresponden a 0 a 1.200 ºC o 0 a 10 V corresponden a 0 a 3.000 U/min). Debido a la normalización de la consigna o del valor real, ambas magnitudes de proceso están en el mismo rango.

Modo de funcionamiento

La instrucción normaliza la magnitud de entrada en una magnitud de salida con otro rango. La magnitud de entrada INV se transfiere a la magnitud de salida OUTV con la recta de normalización. La recta de normalización se define mediante los dos pares de valores IN_LVAL, OUT_LVAL y IN_HVAL, OUT_HVAL.



Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)
Acondicionamiento del valor real (Página 10171)
Control anticipativo (Página 10175)

Parámetros de entrada NORM

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
IN_HVAL	4.0	REAL	100.0	Valor físico de entrada bajo Rango técnico > IN_LVAL
OUT_HVAL	8.0	REAL	100.0	Valor físico de salida bajo Rango técnico > OUT_LVAL
IN_LVAL	12.0	REAL	0.0	Valor físico de entrada alto Rango técnico < IN_HVAL
OUT_LVAL	16.0	REAL	0.0	Valor físico de salida alto Rango técnico < OUT_HVAL

Parámetros de salida de NORM

Parámetro	Offset	Tipo de datos	Descripción
OUTV	20.0	REAL	Magnitud de salida

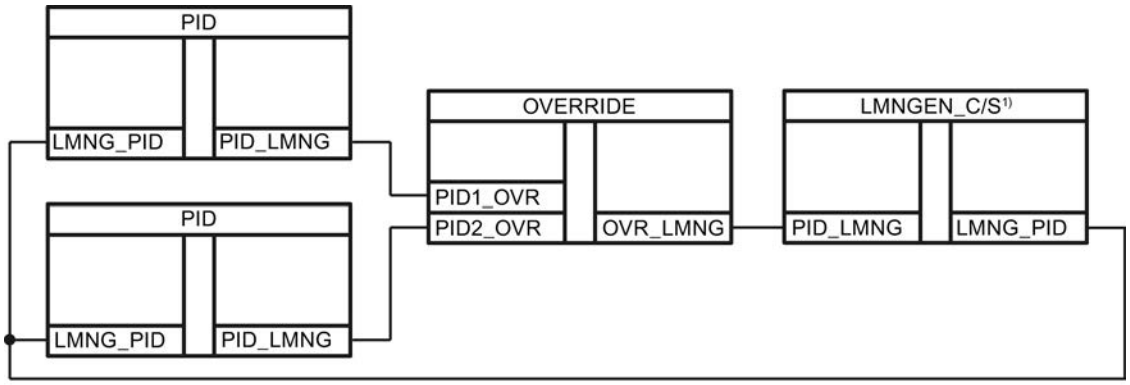
OVERRIDE

Descripción de OVERRIDE

La instrucción se utiliza para implementar una regulación de invalidación (regulación override).

Modo de funcionamiento

Dos reguladores PID se conectan en un actuador. Para ello se interconectan 2 instrucciones PID a través de la instrucción OVERRIDE con la instrucción LMNGEN_C o LMNGEN_S. Si utiliza LMNGEN_S, la realimentación de posición debe estar activada (LMNR_ON=TRUE).



La instrucción compara los valores PID1_OVR y PID2_OVR y aplica el mayor o el menor.

La instrucción transmite el valor de PID1_OVR o PID2_OVR a LMN_GEN_C o LMN_GEN_S.

Si los conmutadores PID1_ON, PID2_ON no son iguales, siempre se transmite un valor. Si los conmutadores PID1_ON, PID2_ON son iguales, se transmite el valor mayor (OVR_MODE=FALSE) o menor (OVR_MODE=TRUE) de ambos.

PID1_ON	PID2_ON	OVR_MOD	Función
TRUE	FALSE	no aplicable	PID1_OVR se transmite.
FALSE	TRUE		PID2_OVR se transmite.
FALSE	FALSE	FALSE	El valor mayor de PID1_OVR y PID2_OVR se transmite.
TRUE	TRUE	FALSE	
FALSE	FALSE	TRUE	El valor menor de PID1_OVR y PID2_OVR se transmite.
TRUE	TRUE	TRUE	

Si se transmite PID1_OVR, entonces QPID1=TRUE, si se transmite PID2_OVR, entonces QPID2=TRUE.

Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada de OVERRIDE

Parámetro	Offset	Tipo de datos	Ajuste predetermin ado	Descripción
PID1_ON	0.0	BOOL	FALSE	Activar regulador 1 PID
PID2_ON	0.1	BOOL	FALSE	Activar regulador 2 PID
OVR_MODE	0.2	BOOL	FALSE	Modo de invalidación FALSE=máximo, TRUE=mínimo
PID1_OVR	2.0	STRUC		PID-LMNGEN interfaz
PID2_OVR	20.0	STRUC		PID-LMNGEN interfaz

Parámetros de salida de OVERRIDE

Parámetro	Offset	Tipo de datos	Descripción
QPID1	38.0	BOOL	Regulador 1 PID activado
QPID2	38.1	BOOL	Regulador 2 PID activado
OVR_LMNG	40.0	STRUC	PID-LMNGEN interfaz

PARA_CTL

Descripción de PARA_CTL

La instrucción se utiliza en estructuras de regulador con conmutaciones de parámetros cuando se requieren parámetros de regulación óptimos para diferentes rangos de trabajo.

Modo de funcionamiento

Se pueden transmitir varios juegos de parámetros de regulador (GAIN, TI, TD y TM_LAG) a un regulador PID.

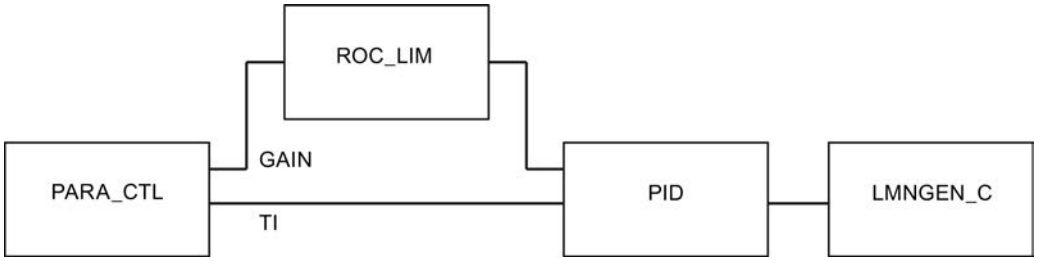
A través de los conmutadores PSET1_ON...PSET4_ON se puede transferir uno de los 4 juegos de parámetros a las salidas GAIN, TI, TD y TM_LAG.

Si	Se aplica
PSET1_ON = TRUE	GAIN = PARASET1.GAIN
	TI = PARASET1.TR
	TD = PARASET1.TD
	TM_LAG = PARASET1.TM_LAG
PSET2_ON = TRUE	GAIN = PARASET2.GAIN
	TI = PARASET2.TR
	TD = PARASET2.TD
	TM_LAG = PARASET2.TM_LAG

PSET3_ON = TRUE	GAIN = PARASET3.GAIN
	TI = PARASET3.TR
	TD = PARASET3.TD
	TM_LAG = PARASET3.TM_LAG
PSET4_ON = TRUE	GAIN = PARASET4.GAIN
	TI = PARASET4.TR
	TD = PARASET4.TD
	TM_LAG = PARASET4.TM_LAG

Si existen dos o más conmutadores en TRUE, el conmutador con el número más bajo siempre tiene la prioridad más alta. Si ningún conmutador es TRUE, se transmite el primer juego de parámetros.

Para conseguir una conmutación suave entre los juegos de parámetros, en el parámetro GAIN hay que aplicar un generador de rampa (ROC_LIM) entre las instrucciones PID y PARA_CTL.



Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada de PARA_CTL

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
PSET1_ON	0.0	BOOL	FALSE	Transferir número 1 del registro de parámetros
PSET2_ON	0.1	BOOL	FALSE	Transferir número 2 del registro de parámetros
PSET3_ON	0.2	BOOL	FALSE	Transferir número 3 del registro de parámetros
PSET4_ON	0.3	BOOL	FALSE	Transferir número 4 del registro de parámetros
PARASET1.GAIN	2.0	REAL		Ganancia proporcional 1
PARASET1.TI	6.0	TIME	5 s	Tiempo de integración 1
PARASET1.TD	10.0	TIME	10 s	Tiempo de acción derivativa 1
PARASET1.TM_LAG	14.0	TIME	2 s	Tiempo de retardo 1

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
PARASET2.GAIN	18.0	REAL		Ganancia proporcional 2
PARASET2.TI	22.0	TIME	5 s	Tiempo de integración 2
PARASET2.TD	26.0	TIME	10 s	Tiempo de acción derivativa 2
PARASET2.TM_LAG	30.0	TIME	2 s	Tiempo de retardo 2
PARASET3.GAIN	34.0	REAL		Ganancia proporcional 3
PARASET3.TI	38.0	TIME	5 s	Tiempo de integración 3
PARASET3.TD	42.0	TIME	10 s	Tiempo de acción derivativa 3
PARASET3.TM_LAG	46.0	TIME	2 s	Tiempo de retardo 3
PARASET4.GAIN	500	REAL		Ganancia proporcional 4
PARASET4.TI	54.0	TIME	5 s	Tiempo de integración 4
PARASET4.TD	58.0	TIME	10 s	Tiempo de acción derivativa 4
PARASET4.TM_LAG	62.0	TIME	2 s	Tiempo de retardo 4

Parámetros de salida de PARA_CTL

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
GAIN	66.0	REAL	1.0	Ganancia proporcional
TI	70.0	TIME	10s	Tiempo de integración
TD	74.0	TIME	5s	Tiempo de acción derivativa
TM_LAG	78.0	TIME	2s	Tiempo de retardo

PID

Descripción de PID

La instrucción contiene el algoritmo PID para configurar los siguientes tipos de regulador:

Reguladores PID continuos:	PID + LMNGEN_C
Regulador de impulsos PID para elementos finales de control proporcionales:	PID + LMNGEN_C + PULSEGEN
Regulador paso a paso PID para elementos finales de control integrados:	PID + LMNGEN_S

La instrucción implementa el algoritmo PID. Tiene una estructura puramente paralela y actúa exclusivamente como algoritmo de posición. Las acciones proporcional, integral y diferencial se pueden conectar o desconectar individualmente. De este modo se pueden parametrizar reguladores P, PI, PD y PID.

Los cálculos de las acciones P y D se pueden poner en la rama de realimentación. Mediante el desplazamiento de la acción P y D a la rama de realimentación, el comportamiento de

control es "suave" y al mismo tiempo tiene la misma velocidad de la regulación de las perturbaciones. En la mayoría de los casos no es necesaria la utilización de un integrador de consigna para evitar saltos de consigna.

Mientras la instrucción PID se sitúa en un nivel de alarma cíclica, cuyo tiempo de ciclo está adaptado a la constante de tiempo dominante en el proceso, las instrucciones que procesan el elemento final de control (LMNGEN_C o LMNGEN_S) pueden situarse en un nivel de alarma cíclica más rápido.

Rearranque

Durante el rearranque todos los parámetros de salida y de paso se preasignan con cero.

Límites internos del bloque

El tiempo de integración se limita en la mitad del tiempo de muestreo hacia abajo.

El tiempo de acción derivativa se limita en el tiempo de muestreo por abajo.

El tiempo de retardo se limita en la mitad del tiempo de muestreo por abajo.

TI_{intern} = CYCLE/2	para	TI < CYCLE/2
TD_{intern} = CYCLE	para	TD < CYCLE
TM_LAG_{intern} = CYCLE/2	para	TM_LAG < CYCLE/2

Los valores restantes de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Crear un regulador continuo (Página 10178)

Crear un regulador de impulsos (Página 10179)

Crear un regulador paso a paso (Página 10183)

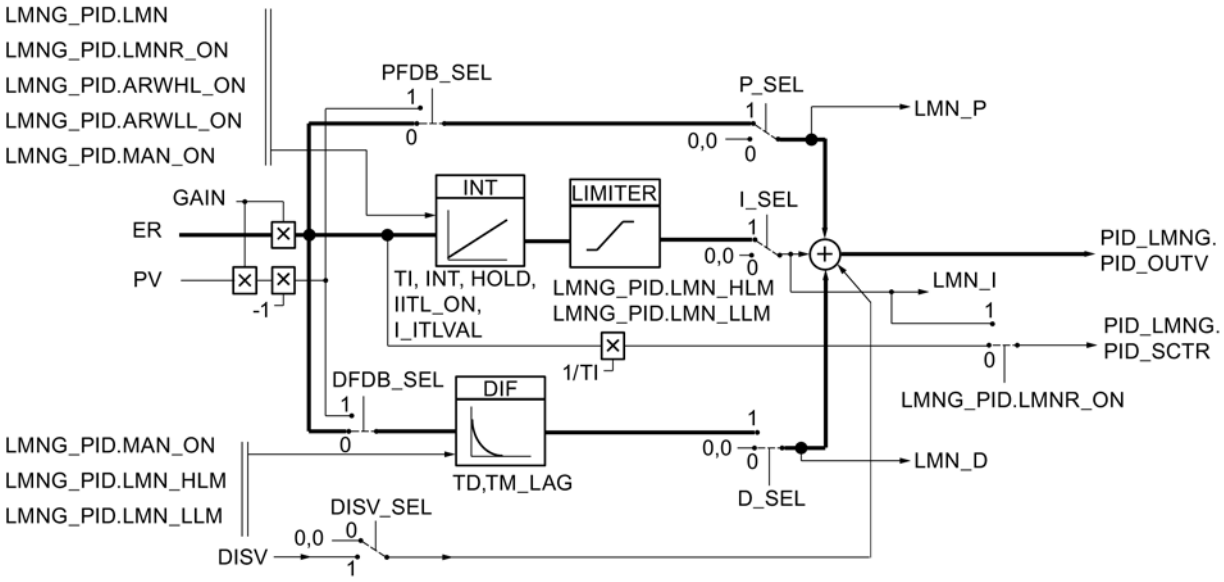
Funcionamiento de PID

Estructura de regulador

Los conmutadores P_SEL, I_SEL y D_SEL hacen posibles las siguientes estructuras de regulador:

Estructura de regulador	P_SEL	I_SEL	D_SEL
Regulación P	TRUE	TRUE bzw. FALSE	FALSE
Regulación PI	TRUE	TRUE	FALSE
Regulación PD	TRUE	FALSE	TRUE
Regulación PID	TRUE	TRUE	TRUE

Diagrama de bloques del algoritmo PID



Acción P

La acción P se puede activar o desactivar mediante el conmutador P_SEL. La acción P representa el producto del error de regulación ER y la ganancia proporcional GAIN. El conmutador PFDB_SEL permite ponerla en la rama de realimentación. Si la acción P está puesta en la rama de realimentación, es el producto entre el valor real PV y la ganancia proporcional.

Acción I

La acción I se puede activar o desactivar mediante el conmutador I_SEL. La respuesta temporal se determina mediante el tiempo integrativo TI. En estado desactivado la acción I y la memoria interna de la acción I se ponen en cero. Con INT_HOLD se puede congelar la acción I. El conmutador I_ITL_ON permite asignar el valor de la entrada I_ITLVAL a la acción I. En el caso de límites de la variable manipulada, la acción I se mantiene en el valor antiguo (anti reset wind-up).

En el regulador paso a paso sin realimentación de posición rige lo siguiente: LMN_I = 0.0

Acción D

La acción D se puede activar o desactivar mediante el conmutador D_SEL. El conmutador DFDB_SEL permite ponerla en la rama de realimentación. La magnitud de entrada de la acción D es ahora la magnitud de regulación PV. La respuesta temporal se determina mediante el tiempo derivativo TD.

Especialmente en los procesos rápidos y con la acción D activada puede ocurrir una oscilación en la señal de la variable manipulada. En este caso se puede mejorar el comportamiento de regulación con el retardo de primer orden integrado en la acción D. Normalmente suele bastar un tiempo de retardo pequeño TM_LAG para obtener el resultado deseado.

Control anticipativo

Se puede aplicar una magnitud perturbadora DISV además de la variable manipulada. Con el conmutador DISV_SEL se puede activar o desactivar esta magnitud.

Parámetros de entrada de PID

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
ER	0.0	REAL	0.0	Error de regulación Rango técnico
PV	4.0	REAL	0.0	Magnitud de regulación (acción P o D en la rama de realimentación) Rango técnico
GAIN	8.0	REAL	1.0	Ganancia proporcional
TI	12.0	TIME	T#20s	Tiempo de integración
I_ITLVAL	16.0	REAL	0.0	Valor de inicialización para la acción I Rango técnico
TD	20.0	TIME	T#10s	Tiempo de acción derivativa
TM_LAG	24.0	TIME	T#2s	Tiempo de retardo de la acción D
DISV	28.0	REAL	0.0	Perturbación Rango técnico
P_SEL	32.0	BOOL	TRUE	Activar la acción P
PFDB_SEL	32.1	BOOL	FALSE	Conectar la acción P en rama de realimentación
DFDB_SEL	32.2	BOOL	FALSE	Conectar la acción D en rama de realimentación
I_SEL	32.3	BOOL	TRUE	Activar la acción I
INT_HPOS	32.4	BOOL	FALSE	Congelar la acción I en dirección positiva
INT_HNEG	32.5	BOOL	FALSE	Congelar la acción I en dirección negativa
I_ITL_ON	32.6	BOOL	FALSE	Poner la acción I
D_SEL	32.7	BOOL	FALSE	Activar la acción D
DISV_SEL	33.0	BOOL	TRUE	Control anticipativo
SMOO_CHG	33.1	BOOL	TRUE	Pasar de forma suave del modo manual al modo automático
COM_RST	33.2	BOOL	FALSE	Rearranque
CYCLE	34.0	TIME	T#1s	Tiempo de muestreo ≥ 1ms
LMNG_PID	38.0	STRUC		Interfaz PID-LMNGEN

Parámetros de salida PID

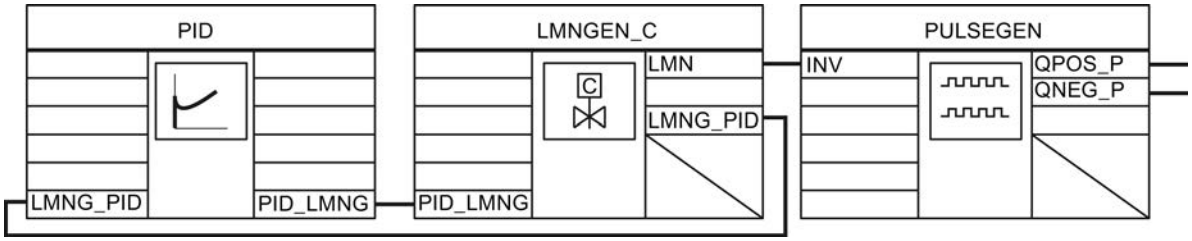
Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
LMN_P	56.0	REAL	0.0	Acción P
LMN_I	60.0	REAL	0.0	Acción I
LMN_D	64.0	REAL	0.0	Acción D
PID_LMNG	68.0	STRUC		Interfaz PID-LMNGEN

PULSEGEN_M

Descripción de PULSEGEN_M

La instrucción PULSEGEN_M se utiliza para crear un regulador PID con salida de impulsos para actuadores de acción proporcional. La instrucción PULSEGEN_M permite generar reguladores PID de dos o tres puntos con modulación de ancho de impulsos. La instrucción se aplica casi siempre en combinación con el regulador continuo PID + LMNGEN_C.

La siguiente figura muestra la interconexión de las instrucciones.



Rearranque

Durante el rearranque todas las salidas de señal se ponen en cero.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

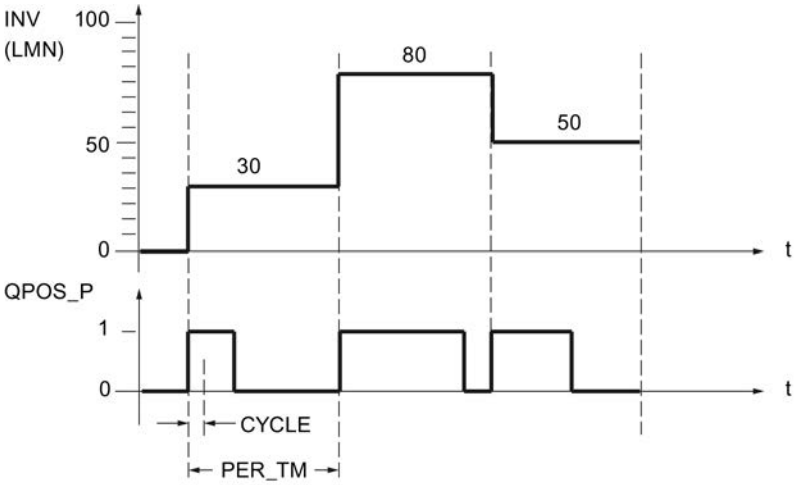
Consulte también

Crear un regulador de impulsos (Página 10179)

Funcionamiento de PULSEGEN_M

Modulación de ancho de impulsos

PULSEGEN_M transforma la magnitud de entrada INV (= LMN del regulador PID) mediante modulación del ancho de impulsos en un tren de impulsos con una duración de período constante PER_TM. La duración del período equivale al tiempo de ciclo con el que se actualiza la magnitud de entrada. La duración de un impulso por duración de período es proporcional a la magnitud de entrada. El ciclo parametrizado con PER_TM no es idéntico al ciclo de ejecución de la instrucción PULSEGEN_M. Es más, un ciclo PER_TM está compuesto por varios ciclos de ejecución de la instrucción PULSEGEN_M, y la cantidad de llamadas de PULSEGEN_M por ciclo PER_TM representa una medida para la precisión del ancho de impulso.



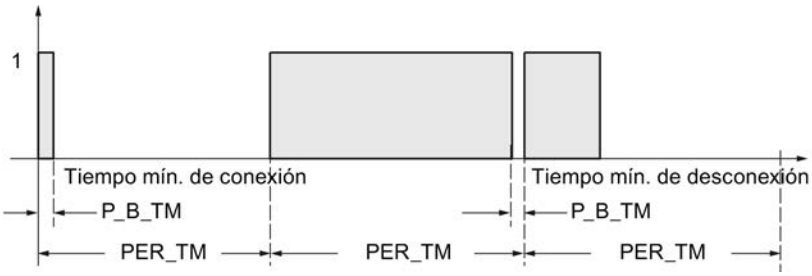
Así, una magnitud de entrada del 30 % y 10 llamadas de PULSEGEN por PER_TM significan:

- "uno" en la salida QPOS_P para las tres primeras llamadas de PULSEGEN_M (30% de 10 llamadas)
- "cero" en la salida QPOS_P para otras siete llamadas de PULSEGEN_M (70% de 10 llamadas)

Duración mínima de impulso o de pausa

La correcta parametrización de la duración mínima de impulso o pausa P_B_TM puede impedir tiempos de conexión o desconexión cortos, que merman la vida útil de los actuadores. Se suprimen valores absolutos pequeños de la magnitud de entrada LMN que generarían una duración de impulso inferior a P_B_TM. Los valores de entrada grandes que generarían una duración de impulso superior a PER_TM - P_B_TM se ajustan a 100 % o a -100 %.

Deben regir la fórmula siguiente: P_B_TM ≤ 0,1 × PER_TM



La duración de los impulsos positivos o negativos se calcula multiplicando la magnitud de entrada (en %) por la duración del período:

Duración de impulso = INV / 100 * PER_TM

Precisión de la modulación de ancho de impulsos

Mientras que el regulador continuo PID+LMGEN_C se posiciona en un nivel de alarma cíclica lento cuyo tiempo de ciclo se adapta a la constante dominante de tiempo del proceso, PULSEGEN_M debe posicionarse en un nivel de alarma cíclica más rápido.

Con una "relación de muestreo" de 1:10 (llamadas del regulador respecto a llamadas de PULSEGEN_M) la precisión de la variable manipulada se limita al 10%, es decir, los valores de entrada especificados INV solo se pueden mapear con una precisión del 10% en una duración de impulso en la salida QPOS_P.

La precisión aumenta con el número de llamadas de PULSEGEN_M por cada llamada del regulador.

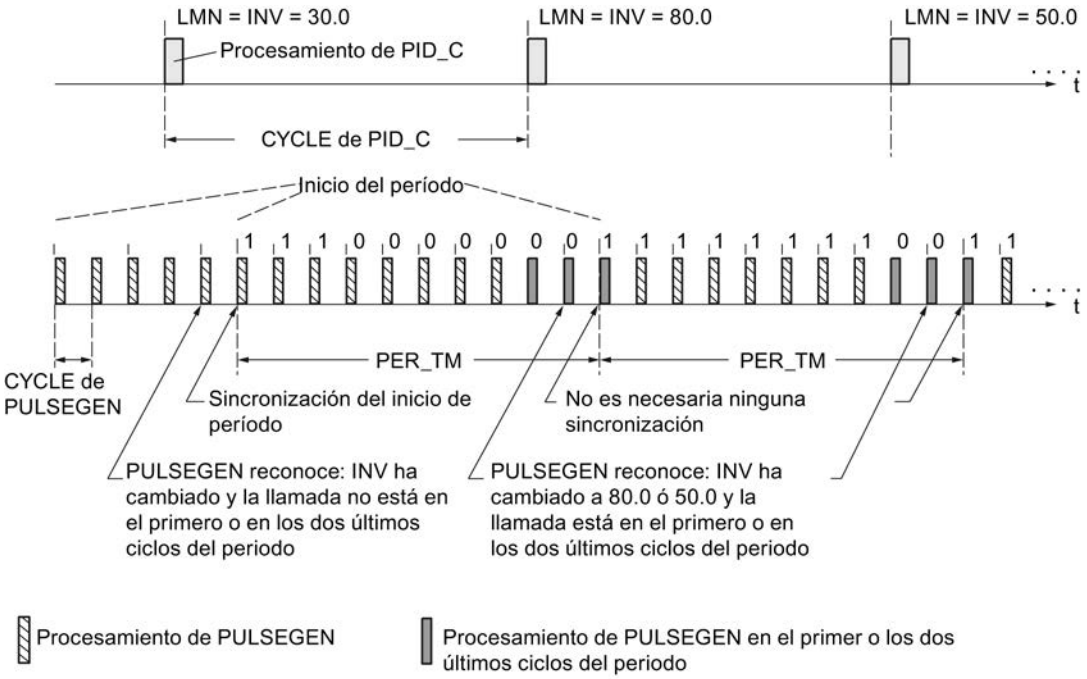
Cuanto más rápido es el nivel de alarma cíclica, tanto mayor puede ser la precisión de la variable manipulada emitida. En caso de un nivel de alarma cíclica 100 veces más rápido, se obtendrá una resolución de 1 % del rango de la variable manipulada.

Sincronización automática

La salida de impulsos se puede sincronizar automáticamente con la instrucción del regulador. Así se garantiza que un valor modificado de la magnitud manipulada LMN(t) se emita lo más rápido posible como duración de impulso modificada proporcionalmente de la señal binaria.

El formador de impulsos evalúa la magnitud de entrada INV en intervalos de la duración de período PER_TM. Pero como INV se calcula en un nivel de alarma cíclica más lento, el formador de impulsos debería iniciar la transformación del valor discreto en una señal de impulso con la mayor rapidez posible tras la actualización de INV. Para ello, la instrucción puede sincronizar por sí misma el inicio del período según el siguiente procedimiento. Si INV ha cambiado y la llamada de bloque no se encuentra en el primer ciclo de llamada o en los dos últimos ciclos de un período, se realiza una sincronización. La duración de impulso se calcula de nuevo, y en el siguiente ciclo se inicia la emisión con un nuevo período.

La duración de período PER_TM debe corresponder al tiempo de muestreo CYCLE del regulador.



La sincronización automática se desactiva con SYN_ON = FALSE.

Nota

Con el inicio del periodo nuevo, el valor antiguo de INV (es decir, de LMN) se representa en la señal de impulso de manera más o menos precisa una vez concluida la sincronización.

Modos de operación PULSEGEN_M

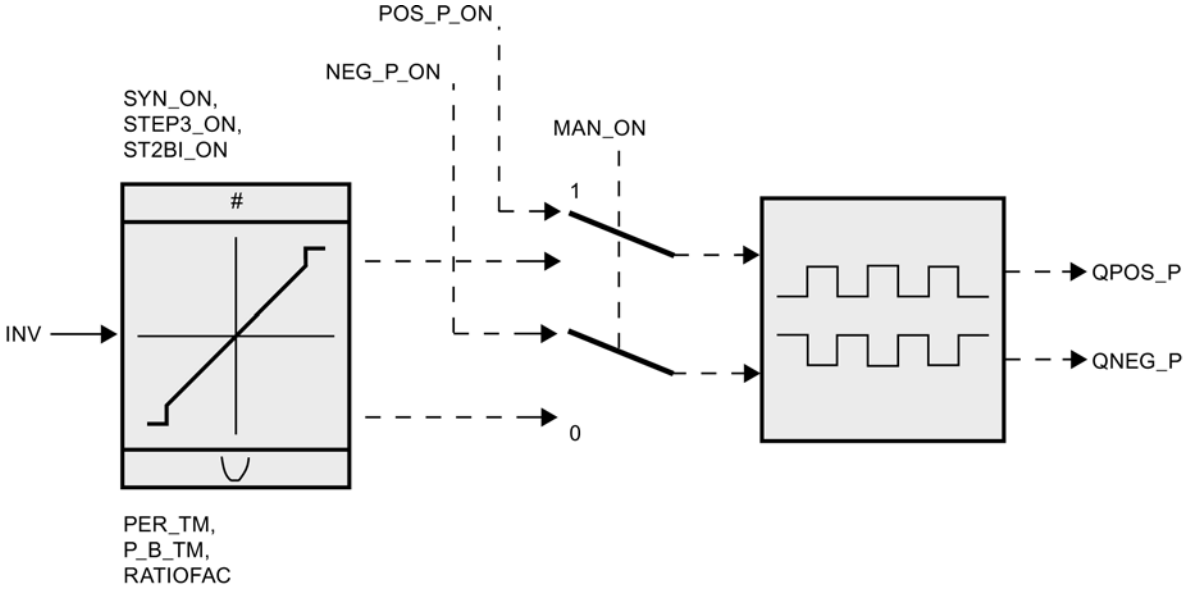
Modos de operación

Dependiendo de la parametrización del formador de impulsos, los reguladores PID se pueden configurar con comportamiento de tres puntos o con salida de dos puntos bipolar o unipolar. La siguiente tabla contiene los ajustes de las combinaciones de conmutadores para los modos de operación posibles.

Modo de operación	MAN_ON	STEP3_ON	ST2BI_ON
Regulación de tres puntos	FALSE	TRUE	cualquiera
Regulación de dos puntos con rango de valores Rango de ajuste (-100 % ... 100 %)	FALSE	FALSE	TRUE
Regulación de dos puntos con rango de valores Rango de ajuste (0 % ... 100 %)	FALSE	FALSE	FALSE
Modo manual	TRUE	cualquiera	cualquiera

Modo manual en la regulación de dos o de tres puntos

En el modo manual (MAN_ON = TRUE) las salidas binarias del regulador de tres puntos o del regulador de dos puntos pueden activarse independientemente de INV a través de las señales POS_P_ONy NEG_P_ON.



Regulación	POS_P_ON	NEG_P_ON	QPOS_P	QNEG_P
Regulación de tres puntos	FALSE	FALSE	FALSE	FALSE
	TRUE	FALSE	TRUE	FALSE
	FALSE	TRUE	FALSE	TRUE
	TRUE	TRUE	FALSE	FALSE
Regulación de dos puntos	FALSE	cualquiera	FALSE	TRUE
	TRUE	cualquiera	TRUE	FALSE

Regulación de tres puntos PULSEGEN_M

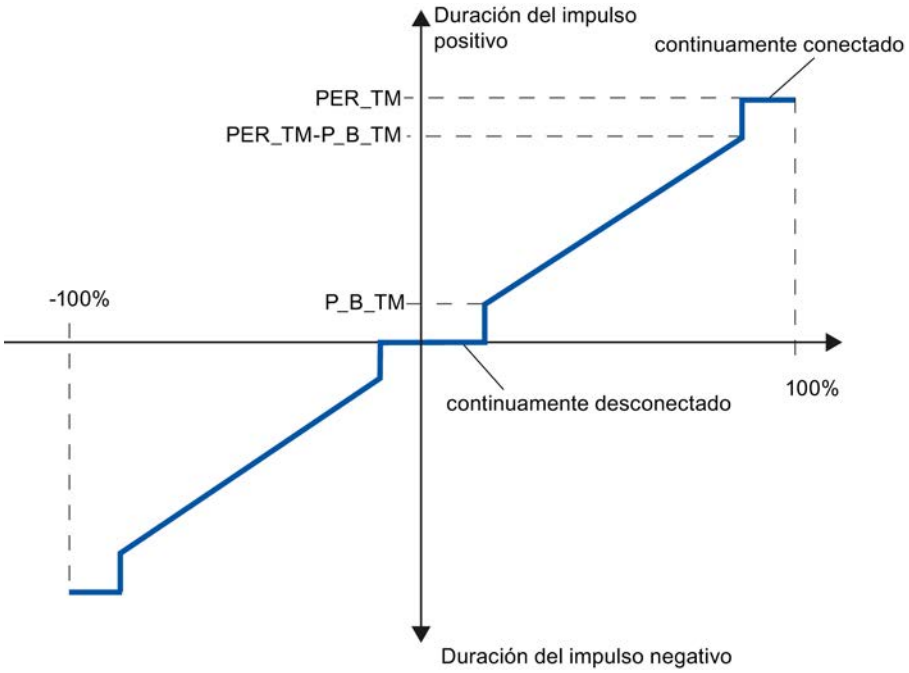
Regulación de tres puntos

En el modo de operación "Regulación de tres puntos" es posible generar tres estados de la señal manipulada, p. ej. más - apagado - menos, adelante - parado - atrás, calentar - apagado - enfriar. Para ello se asignan los valores de estado de las señales binarias de salida QPOS_P y QNEG_P a los correspondientes estados operativos del actuador. La tabla muestra el ejemplo de una regulación de temperatura:

Señales de salida	Calentar	OFF	Enfriar
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

A partir de la magnitud de entrada se calcula la duración del impulso mediante una característica. La forma de esta característica está definida por la duración mínima de impulso o de pausa y por el factor de relación. El valor normal del factor de relación es 1. Los puntos escalonados de las características son originados por la duración mínima de impulso o de pausa P_B_TM.

La siguiente figura muestra una característica simétrica del regulador de tres puntos (factor de relación = 1)



Regulación de tres puntos asimétrica

A través del factor de relación RATIOFAC puede modificarse la relación de la duración de impulsos positivos respecto a los negativos. Durante un lazo de regulación de temperatura, ello permite tener en cuenta p. ej. constantes temporales de proceso distintas para calentar y enfriar.

Factor de relación < 1

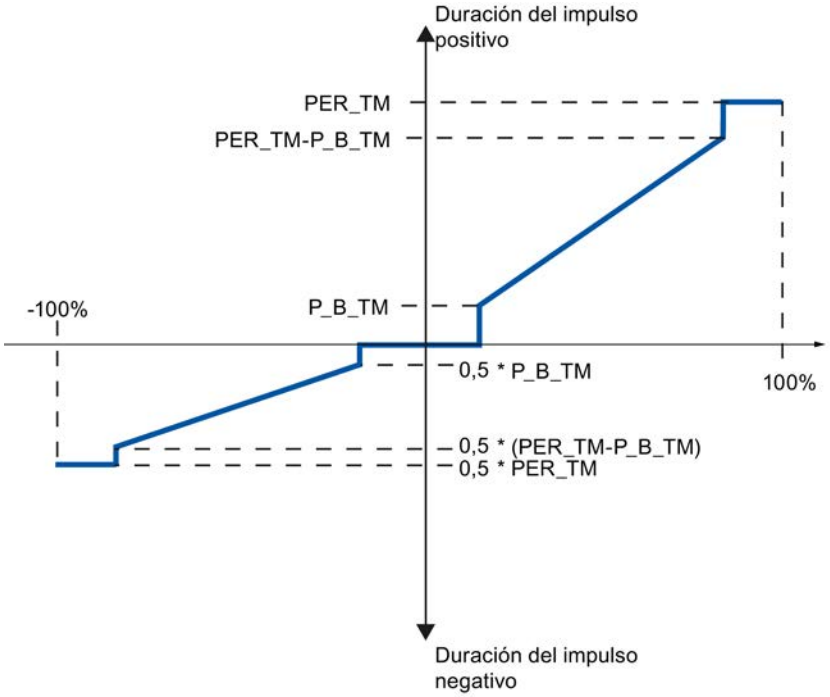
A la duración de impulso obtenida de la multiplicación de la magnitud de entrada por la duración del período en la salida de impulsos negativa se le resta el factor de relación.

Duración de impulso positiva = INV /100 * PER_TM

Duración de impulso negativa = INV / 100 * PER_TM * RATIOFAC

La duración de impulso negativa se multiplica por el factor de relación.

La siguiente figura muestra una característica asimétrica del regulador de tres puntos (factor de relación = 0.5)



El factor de relación RATIOFAC influye sobre la duración mínima de impulso o de pausa. El valor de reacción para impulsos negativos se multiplica por el factor de relación.

Factor de relación > 1

A la duración de impulso obtenida de la multiplicación de la magnitud de entrada por la duración del período en la salida de impulsos positiva se le resta el factor de relación.

Duración de impulso positiva = INV / 100 * PER_TM / RATIOFAC

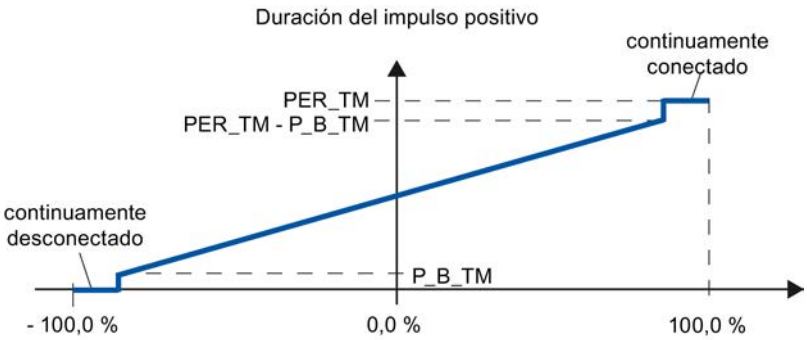
Duración de impulso negativa = INV / 100 * PER_TM

El factor de relación RATIOFAC influye sobre la duración mínima de impulso o de pausa. El valor de reacción para impulsos positivos se divide por el factor de relación.

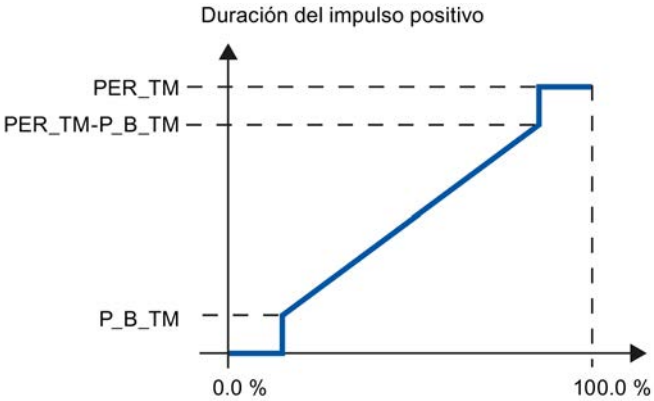
Regulación de dos puntos PULSEGEN_M

En la regulación de dos puntos solo se conecta la salida de impulsos positiva QPOS_P de PULSEGEN_M al actuador de conexión/desconexión en cuestión. Según cuál sea el rango utilizado de la variable manipulada, el regulador de dos puntos tiene un rango bipolar o unipolar.

Regulación de dos puntos con rango de variable manipulada bipolar (-100% ... 100%)



Regulación de dos puntos con rango de variable manipulada unipolar (0%...100%)



En QNEG_P está disponible la señal de salida negada, si la interconexión del regulador de dos puntos en el lazo de regulación exige una señal binaria invertida lógicamente para los impulsos de la señal manipulada.

Impulso	Actuador ON	Actuador OFF
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

Parámetros de entrada de PULSEGEN_M

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Variable de entrada -100.0..100.0 (%)
PER_TM	4.0	TIME	T#1s	Duración de período PER_TM >= 20*CYCLE
P_B_TM	8.0	TIME	T#50ms	Duración mínima del impulso o duración mínima de pausa P_B_TM >= CYCLE

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
RATIOFAC	12.0	REAL	1.0	Factor de relación 0.1..10.0 (sin dimensión)
STEP3_ON	16.0	BOOL	TRUE	Activar la regulación de tres puntos
ST2BI_ON	16.1	BOOL	FALSE	Activar la regulación de dos puntos para el rango de la variable manipulada bipolar
MAN_ON	16.2	BOOL	FALSE	Activar el modo manual
POS_P_ON	16.3	BOOL	FALSE	Activar impulso positivo
NEG_P_ON	16.4	BOOL	FALSE	Activar impulso negativo
SYN_ON	16.5	BOOL	TRUE	Activar sincronización
COM_RST	16.6	BOOL	FALSE	Rearranque
CYCLE	18.0	TIME	T#10ms	Tiempo de muestreo CYCLE >= 1ms

Parámetros de salida de PULSEGEN_M

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QPOS_P	22.0	BOOL	FALSE	Señal de salida del impulso positivo
QNEG_P	22.1	BOOL	FALSE	Señal de salida del impulso negativo

RMP_SOAK

Descripción de RMP_SOAK

Con la instrucción RMP_SOAK se preasignan diferentes consignas fundamentalmente en función del desarrollo temporal del proceso.

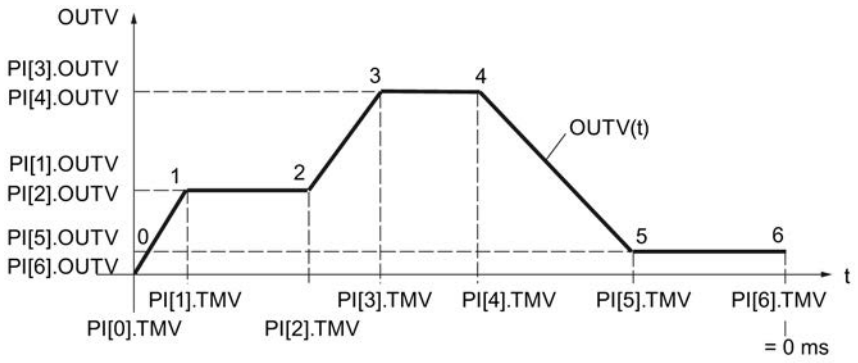
Modo de funcionamiento

En un bloque de datos global se depositan los nodos de la curva de desplazamiento. En cada llamada de la instrucción se emite la consigna válida en ese momento.

El número de nodos se introduce en el bloque de datos global, en la variable NBR_PTS. El número del bloque global se introduce en el parámetro de entrada DB_NBR.

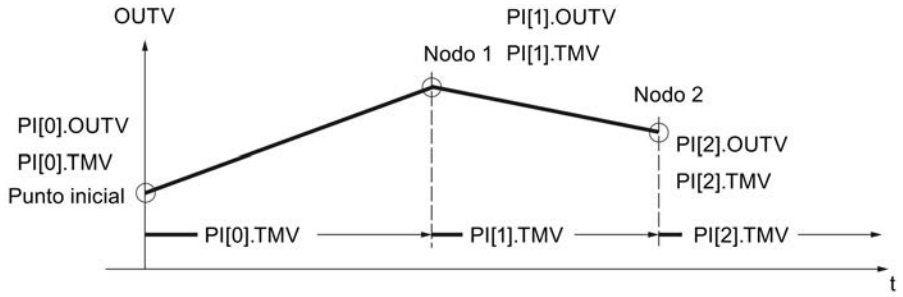
Para cada nodo se especifica la consigna en PI[i].OUTV y el tiempo en PI[i].TMV.

Ejemplo de curva de desplazamiento con punto de inicio y nodos.



Con n nodos el valor de tiempo para el nodo es n=0 ms.

El modo de contar los valores y tiempos de los nodos se muestra en la siguiente imagen:



Entre los nodos el generador ramp soak interpola con 0 ≤ n < (NBR_PTS - 1) según la siguiente función:

$$OUTV(t) = PI[n + 1].OUTV - \frac{RS_TM}{PI[n].TMV} (PI[n + 1].OUTV - PI[n].OUTV)$$

Nota

La instrucción no comprueba si existe realmente un bloque de datos global con el número DB_NBR, ni si el parámetro DB_NBR.NBR_PTS (número de nodos) concuerda con la longitud del bloque de datos. Si no se ha realizado la parametrización correctamente, la CPU cambia al estado de funcionamiento STOP mostrando el mensaje "Error interno del sistema".

Funciones del generador ramp soak

La instrucción dispone de las siguientes funciones:

- Activar generador ramp soak para ejecución única
- Preasignar salida del generador ramp soak con valor fijo
- Activar modo cíclico del generador ramp soak
- Detener procesamiento del generador ramp soak

- Predeterminar paso y tiempo de procesamiento (el tiempo sobrante RS_TM y el número de nodo TM_SNBR se determinan de nuevo)
- Actualizar tiempo total de procesamiento y tiempo sobrante total

Para ajustar el modo de operación deseado se aplica la valencia de las entradas de control de la siguiente tabla:

Modo de operación	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Señal de salida OUTV
Activar generador ramp soak	TRUE	FALSE	FALSE	**)	FALSE	**)	OUTV(t) El valor final se detiene tras el final de procesamiento.
Preasignar salida	TRUE	TRUE	**)	**)	**)	**)	DF_OUTV
Activar modo cíclico	TRUE	FALSE	FALSE	**)	TRUE	**)	OUTV(t) Tras final: arranque automático
Detener generador ramp soak	TRUE	FALSE	TRUE	FALSE	**)	**)	se mantiene el valor actual de OUTV(t) *)
Predeterminar el paso y tiempo de procesamiento	TRUE	FALSE	TRUE	TRUE	**)	**)	OUTV (antiguo) *)
			FALSE				Se sigue con los nuevos valores predeterminados.
Actualizar tiempo total	**)	**)	**)	**)	**)	FALSE	No influye en OUTV
						TRUE	No influye en OUTV
*) Hasta el próximo nodo la curva no tiene la forma parametrizada por el usuario. **) El modo de operación ajustado se ejecuta independientemente de la valencia de las señales de control en los campos marcados con **)							

Rearranque

En el rearranque la salida OUTV se pone a 0.0. Con DFOUT_ON=TRUE, la salida es DF_OUTV. Los intervalos (0 bis NBRPTS-1) entre los nodos 0 hasta NBRPTS se suman y se ponen disponibles en T_TM. La salida QR_S_ACT se resetea; las salidas NBR_ATMS y RS_TM se preasignan con 0.

Activar el generador ramp soak

Con flanco positivo en RMPSK_ON, se activa la función de generador ramp soak. Después de haber llegado al último nodo se termina la curva. Para volver a iniciar el generador ramp soak, primero ajuste RMPSK_ON = FALSE y a continuación RMPSK_ON = TRUE.

Preasignar salida, iniciar curva

Para que la curva de desplazamiento comience con un valor de salida determinado, es necesario que DFOUT_ON = TRUE. En ese caso en la salida del generador ramp soak se deposita el valor de señal DF_OUTV.

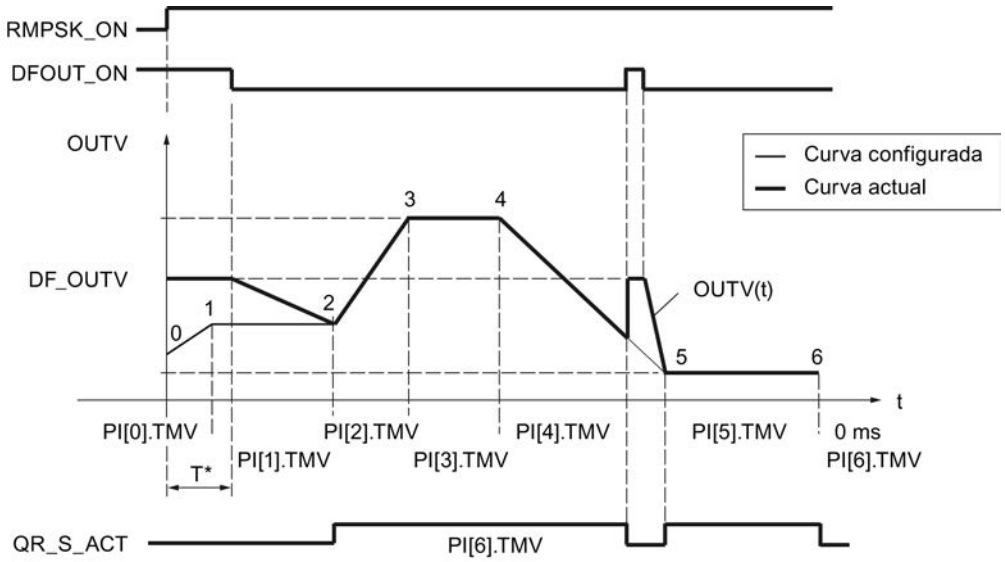
Nota

La señal para la emisión de la consigna constante DFOUT_ON tiene mayor prioridad que la señal de inicio para el generador ramp soak RMPSK_ON.

Tras la conmutación DFOUT_ON = FALSE, OUTV se lleva linealmente de la consigna ajustada (DF_OUTV) al valor de salida del número de nodo actual PI[NBR_ATMS].OUTV.

El procesamiento interno de tiempo también continúa con una consigna fija aplicada directamente (RMPSK_ON = TRUE y DFOUT_ON = TRUE).

Durante el arranque de la curva RMPSK_ON = TRUE se emite la consigna fija DF_OUTV hasta que DFOUT_ON cambia de TRUE a FALSE después de la duración T*.



En este momento se ha pasado el tiempo PI[0].TMV y parte del tiempo PI[1].TMV. OUTV se lleva de DF_OUTV a PI[2].OUTV, es decir, a la consigna del nodo 2.

A partir del nodo 2 se emite la curva configurada. La coincidencia entre la curva configurada y la actual se indica con QR_S_ACT = TRUE. Con un flanco positivo en DFOUT_ON durante el procesamiento de la curva, el valor de la salida OUTV salta sin retardo a DF_OUTV.

Modo cíclico activado

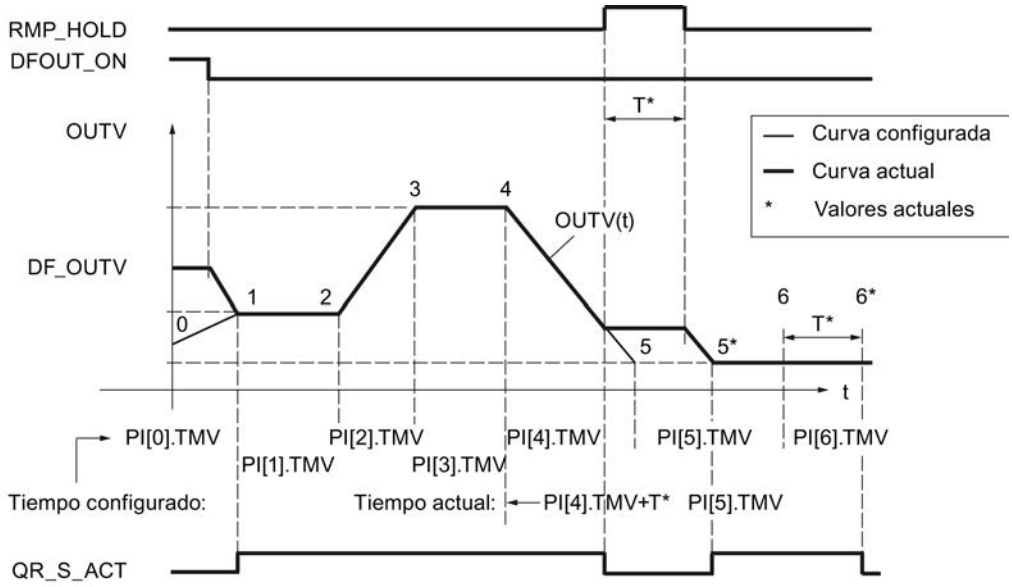
Cuando el modo de operación 'Repetición cíclica' (CYC_ON = TRUE) está activado, el generador ramp soak vuelve automáticamente al punto inicial tras emitir el último valor de nodo y se ejecuta de nuevo.

Entre el último nodo y el punto inicial no se interpola. Para una transición suave se debe aplicar lo siguiente: PI[NBR_PTS].OUTV = PI[0].OUTV.

Detener generador ramp soak

Con RMP_HOLD = TRUE se congela el valor de la magnitud de salida (incluido el procesamiento de tiempo). Cuando RMP_HOLD = FALSE se desactiva, PI[x].TMV continúa en el punto de interrupción.

El tiempo de procesamiento de la curva se prolonga por el tiempo T*. La curva presenta el desarrollo configurado desde el punto de inicio al flanco positivo de RMP_HOLD y desde el nodo 5* hasta el nodo 6*, de modo que la señal de salida QR_S_ACT tiene el valor TRUE.



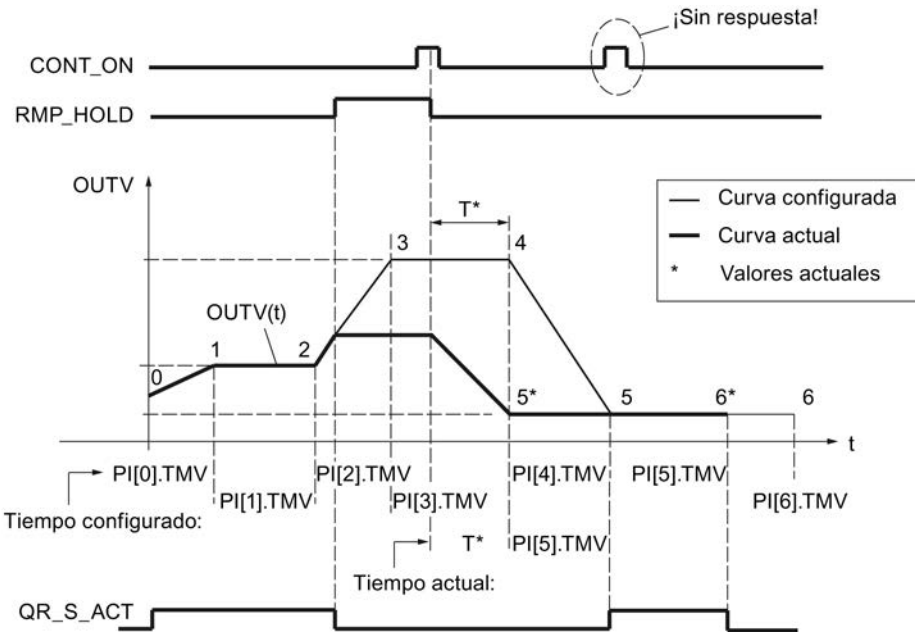
Predeterminar el paso y tiempo de procesamiento

Con CONT_ON se vuelve a registrar una curva detenida en un punto determinado. Si el generador ramp soak no está detenido, CONT_ON no tiene efecto.

Si la entrada de control para continuar CONT_ON = TRUE está activada, el generador ramp soak detenido continúa en el nodo TM_SNBR. TM_CONT determina el tiempo residual que requiere el generador hasta que se alcanza el nodo TM_SNBR.

La siguiente figura muestra la influencia de RMP_HOLD y CONT_ON en:

TM_SNBR = 5 y TM_CONT = T*



En este ciclo del generador ramp soak se prescinde de los nodos 3 y 4.

Después del cambio de señal RMP_HOLD de TRUE a FALSE, el desarrollo de la curva configurado sólo se vuelve a alcanzar a partir del nodo 5.

Actualizar el tiempo total y tiempo sobrante total

En cada ciclo se actualizan los números actuales del nodo NBR_ATMS, el tiempo sobrante actual hasta alcanzar el nodo RS_TM, el tiempo total T_TM y el tiempo sobrante total hasta alcanzar el final de la curva RT_TM.

En caso de modificaciones online de PI[n].TMV, cambian el tiempo total y el tiempo sobrante total de la curva. Ya que el cálculo de T_TM y RT_TM prolonga mucho el tiempo de procesamiento del bloque de función en muchos nodos, sólo se realiza después del rearranque o si TUPDT_ON = TRUE. Los intervalos PI[0 ... NBR_PTS].TMV entre los diferentes nodos se suman y se muestran en la salida del tiempo total T_TM y del tiempo restante total RT_TM.

Tenga en cuenta que la determinación de los tiempos totales requiere mucho tiempo de ejecución de la CPU.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)

Parámetros de entrada RMP_SOAK

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
DF_OUTV	0.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
DB_NBR	4.0	BLOCK_DB	DB 1	Número de bloque de datos, global (valores de nodos) Según la CPU
TM_SNBR	6.0	INT	0	Número del próximo nodo para continuar 0 - 255
TM_CONT	8.0	TIME	T#0s	Tiempo sobrante para continuar hasta el nodo TM_SNBR Rango técnico
DFOUT_ON	12.0	BOOL	FALSE	Activar la preasignación de la magnitud de salida
RMPSK_ON	12.1	BOOL	FALSE	Activar el generador ramp soak
HOLD	12.2	BOOL	FALSE	Congelar la magnitud de salida
CONT_ON	12.3	BOOL	FALSE	Continuar
CYC_ON	12.4	BOOL	FALSE	Activar repetición cíclica
TUPDT_ON	12.5	BOOL	FALSE	Calcular de nuevo el tiempo total tras modificar los valores temporales (tiempo de funcionamiento del bloque prolongado).
COM_RST	12.6	BOOL	FALSE	Rearranque
CYCLE	14.0	TIME	T#1s	Tiempo de muestreo Rango técnico ≥ 1ms

Parámetros de salida RMP_SOAK

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	18.0	REAL	0.0	Magnitud de salida
QR_S_ACT	22.0	BOOL	FALSE	Procesando tabla de tiempo
NBR_ATMS	24.0	INT	0	Número del nodo actual (nodo que se arranca)
RS_TM	26.0	TIME	T#0s	Tiempo sobrante actual
T_TM	30.0	TIME	T#0s	Datos globales: tiempo total ΣTMVI
RT_TM	34.0	TIME	T#0s	Datos globales: tiempo sobrante total ΣTMVI

Los nodos y el número de los nodos NBRPTS se guardan en un bloque de datos global DB_NBR. La emisión empieza con el nodo 0 y termina con el nodo NBR_PTS.

Bloque de datos global RMP_SOAK

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
NBR_PTS	INT	4	Número de nodos -1 1 - 255
PI	Array [0...NBR_PTS] of Struct		Estructura de los datos de los nodos. La matriz debe ser declarada por 0...NBR_PTS.
PI[0].OUTV	REAL	0.0	Valor de salida [0] punto inicial Rango técnico
PI[0].TMV	TIME	T#1s	Valor temporal [0] punto inicial
PI[1].OUTV	REAL	0.0	Valor de salida [1] nodo 1 Rango técnico
PI[1].TMV	TIME	T#1s	Valor temporal [1] nodo 1
PI[2].OUTV	REAL	0.0	Valor de salida [2] nodo 2 Rango técnico
PI[2].TMV	TIME	T#1s	Valor temporal [2] nodo 2
PI[3].OUTV	REAL	0.0	Valor de salida [3] nodo 3 Rango técnico
PI[3].TMV	TIME	T#1s	Valor temporal [3] nodo 3
PI[4].OUTV	REAL	0.0	Valor de salida [4] nodo 4 Rango técnico
PI[4].TMV	TIME	T#0s	Valor temporal [4] nodo 4 0 ms

ROC_LIM

Descripción de ROC_LIM

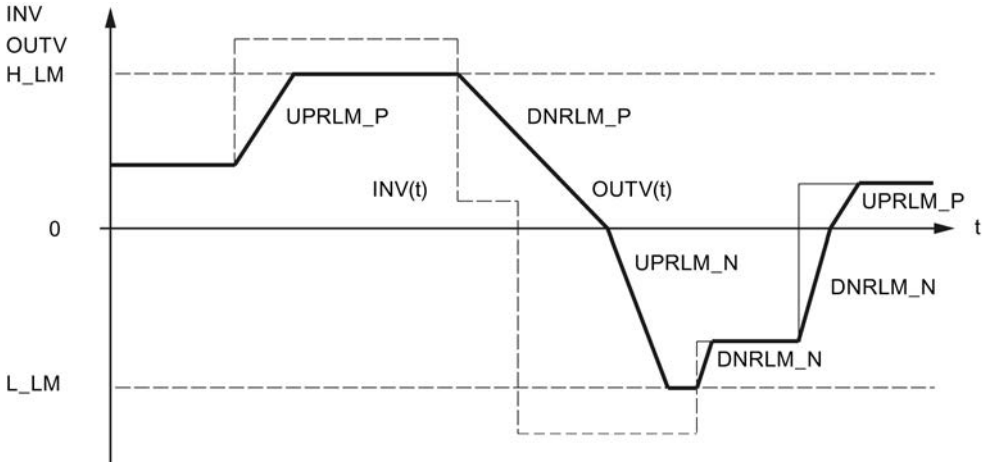
Las funciones de rampa se utilizan cuando el proceso no tolera un escalón de entrada. Esto caso se aplica por ejemplo cuando entre el motor y la carga que se desea mover se interconecta una transmisión y un aumento demasiado rápido de revoluciones causaría una sobrecarga de la transmisión. La instrucción limita la velocidad de cambio de un valor de salida. Las funciones de escalón se convierten en funciones de rampa.

Modo de funcionamiento

Para el rango de valores positivo y negativo de la magnitud de entrada INV pueden parametrizarse hasta cuatro rampas que se utilizan para determinar la magnitud de salida:

- UPRLM_P: valores ascendentes en el rango positivo
  OUTV > 0 y |OUTV| ascendentes
- DNRLM_P: valores descendentes en el rango positivo
  OUTV > 0 y |OUTV| descendentes

- UPRLM_N: valores ascendentes en el rango negativo
  OUTV < 0 y |OUTV| ascendentes
- DNRLM_N: valores descendentes en el rango negativo
  OUTV < 0 y |OUTV| descendentes



Los valores de las rampas parametrizados se refieren a una rampa de subida/bajada de la magnitud de salida por segundo.

Si UPRLM_P = 10.0 e |INV| es creciente, OUTV puede aumentar como máximo 10.0 en 1 s.

Si la magnitud de entrada es superior, OUTV aumenta del siguiente modo en cada llamada en función del tiempo de muestreo CYCLE:

CYCLE [s]	Aumento de OUTV
1	10.0
0.1	1.0
0.01	0.1

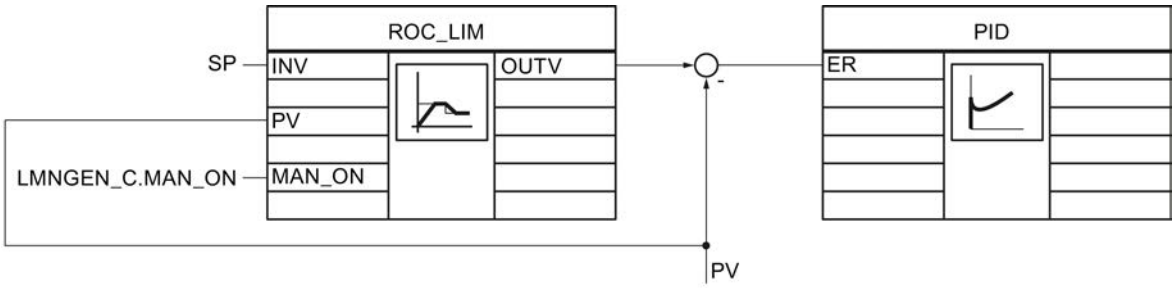
La limitación de la velocidad de cambio se muestra en QUPRLM_P, QDNRLM_P, QUPRLM_N y QDNRLM_N.

La magnitud de salida puede limitarse con H_LM y L_LM. Eso se indica en QH_LM y QL_LM.

Funciones de la instrucción

A través de los parámetros DFOUT_ON, TRACK y MAN_ON se puede influir en el funcionamiento.

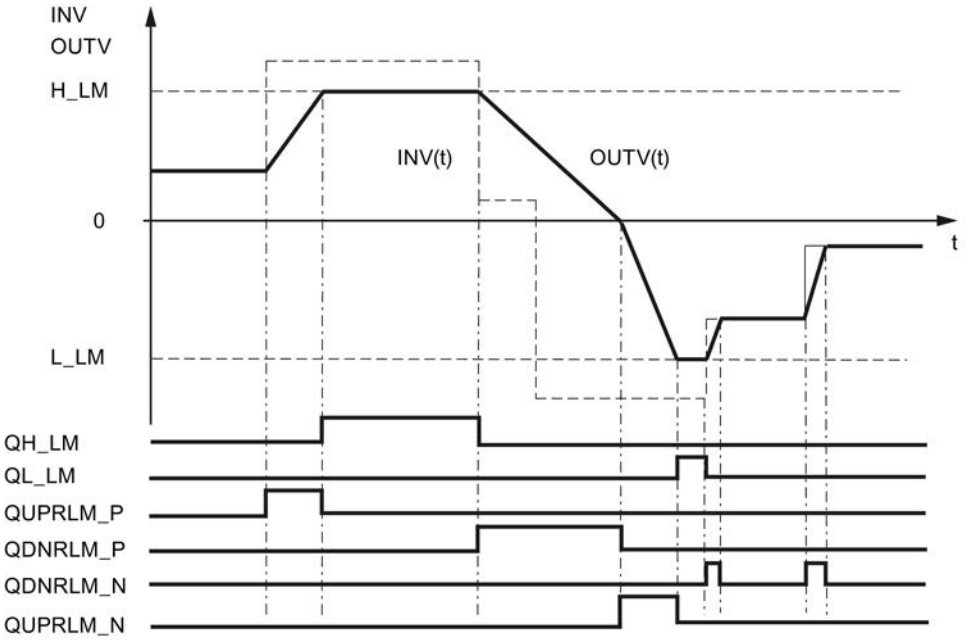
- DFOUT_ON = TRUE
  Si DFOUT_ON = TRUE está activado, DF_OUTV se deposita en la salida OUTV; con flanco negativo OUTV se lleva de DF_OUTV a INV y con flanco positivo OUTV se lleva de INV a DF_OUTV.
- TRACK = TRUE
  Para la corrección de OUTV = INV se activa el bit TRACK = TRUE; ya que la magnitud de entrada se conecta directamente con la magnitud de salida, también se emiten los escalones de la magnitud de entrada.
- MAN_ON = TRUE
  MAN_ON = TRUE posibilita la conmutación suave del modo manual al automático. Para ello la instrucción ROC_LIM debe integrarse en la rama de la consigna directamente antes del error de regulación. En la entrada PV se interconecta el valor real y en la entrada MAN_ON el bit manual-automático, es decir LMNGEN_C.MAN_ON.



Al cambiar al modo de operación manual MAN_ON = TRUE, el valor presente en la entrada PV se traslada directamente a la salida OUTV. Dado que la consigna y el valor real son iguales, el error de regulación se pone en cero y el regulador está en un estado de reposo estacionario. Al regresar al modo automático MAN_ON = FALSE, la salida OUTV va desde el valor presente en ese momento PV en rampa hasta el valor de entrada INV. Eso provoca un cambio suave de la estructura del regulador del modo manual al automático.
En caso de MAN_ON = TRUE las limitaciones H_LM y L_LM no son efectivas, DFOUT_ON no se tiene en cuenta.

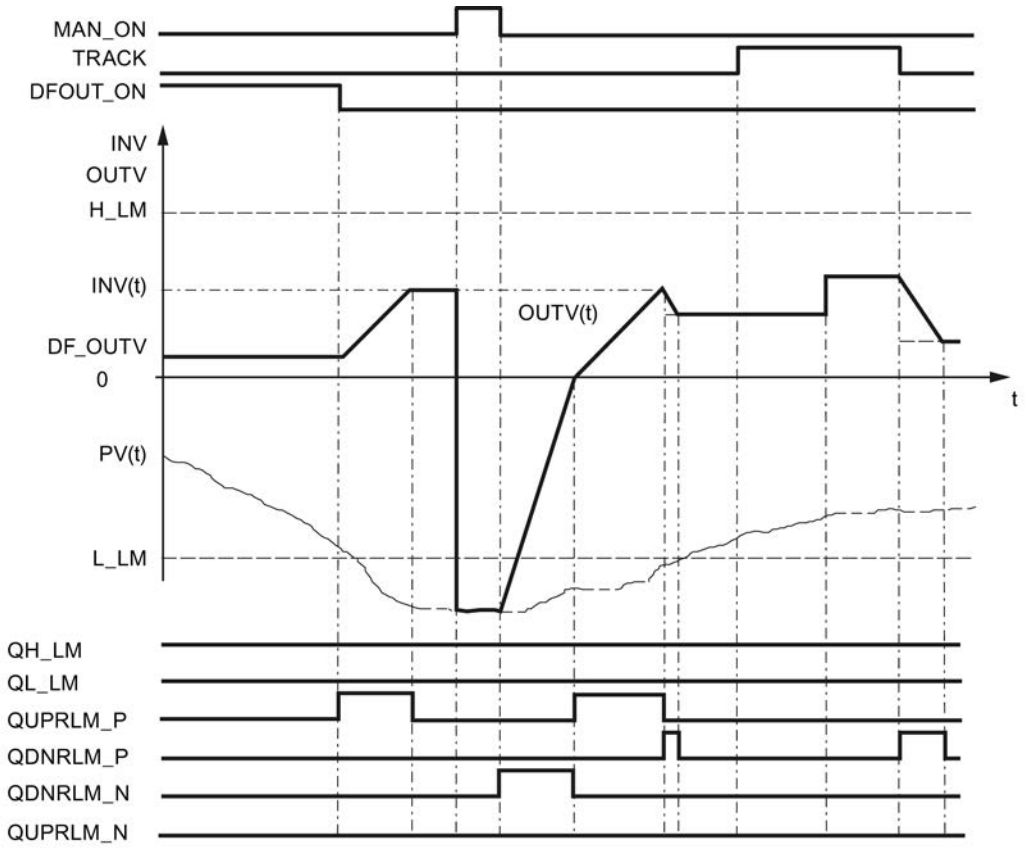
Ejemplo 1

Con MAN_ON, TRACK, DFOUT_ON igual a FALSE , resultan los siguientes desarrollos de las salidas de señal:



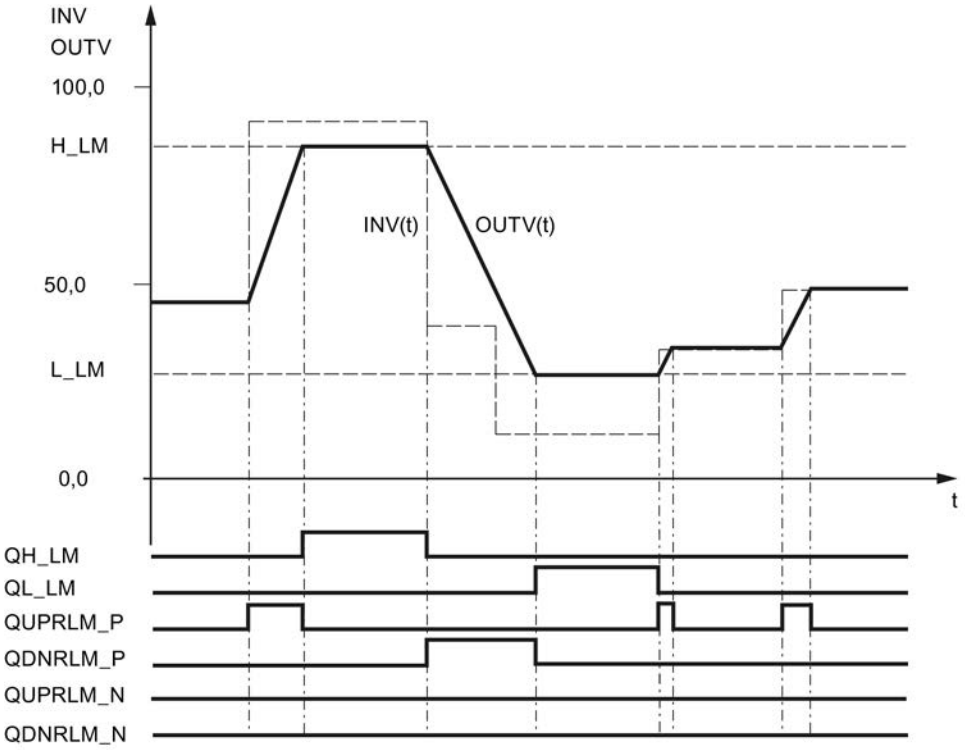
Ejemplo 2

Conmutación suave de manual a automático



Ejemplo 3

Rampa que sólo se utiliza en el rango positivo.



Rearranque

En el rearranque la salida OUTV se pone a 0.0. Si está activado DFOUT_ON = TRUE, se emite DF_OUTV. Todas las salidas de señal se ponen a FALSE.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)

Parámetros de entrada de ROC_LIM

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
UPRLM_P	4.0	REAL	10.0	Límite de subida en el rango positivo Incremento por segundo [1/s] > 0.0
DNRLM_P	8.0	REAL	10.0	Límite de bajada en el rango positivo Incremento por segundo [1/s] > 0.0
UPRLM_N	12.0	REAL	10.0	Límite de subida en el rango negativo Incremento por segundo [1/s] > 0.0
DNRLM_N	16.0	REAL	10.0	Límite de bajada en el rango negativo Incremento por segundo [1/s] > 0.0
H_LM	20.0	REAL	100.0	Límite superior Rango técnico > L_LM
L_LM	24.0	REAL	0.0	Límite inferior Rango técnico < H_LM
PV	28.0	REAL	0.0	Magnitud de regulación Rango técnico
DF_OUTV	32.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
DFOUT_ON	36.0	BOOL	FALSE	Activar la preasignación de la magnitud de salida
TRACK	36.1	BOOL	FALSE	Corrección OUTV=INV
MAN_ON	36.2	BOOL	FALSE	Activar el modo manual
COM_RST	36.3	BOOL	FALSE	Rearranque
CYCLE	38.0	TIME	T#1s	Tiempo de muestreo

Parámetros de salida ROC_LIM

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	42.0	REAL	0.0	Magnitud de salida
QUPRLM_P	46.0	BOOL	FALSE	Límite de la rampa de subida en el rango positivo ha reaccionado

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
QDNRLM_P	46.1	BOOL	FALSE	Límite de la rampa de bajada en el rango positivo ha reaccionado
QUPRLM_N	46.2	BOOL	FALSE	Límite de subida en el rango negativo ha reaccionado
QDNRLM_N	46.3	BOOL	FALSE	Límite de bajada en el rango negativo ha reaccionado
QH_LM	46.4	BOOL	FALSE	Límite superior ha reaccionado
QL_LM	46.5	BOOL	FALSE	Límite inferior ha reaccionado

SCALE_M

Descripción de SCALE_M

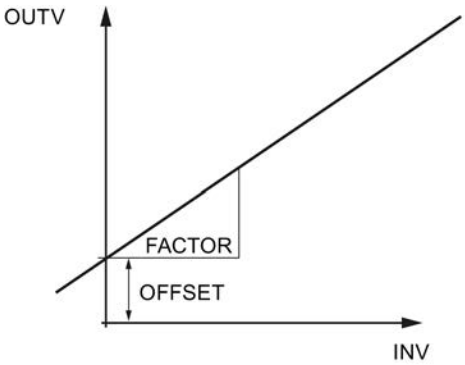
En los valores reales el valor del generador muchas veces está en un rango que no resulta adecuado para el usuario, por ejemplo 0 a 10 V corresponden a 0 a 1200 ºC ó 0 a 10 V corresponden a 0 a 3000 rpm. Mediante la adaptación de la consigna o del valor real, las dos magnitudes de proceso están en el mismo rango.

Modo de funcionamiento

La instrucción normaliza una magnitud de entrada analógica. La magnitud de entrada analógica INV se transfiere a la magnitud de salida OUTV con la recta de normalización. La recta de normalización se define mediante la subida (FACTOR) y la distancia (OFFSET) entre OUTV con INV=0 y el eje de coordenadas OUTV=0.

Fórmula:

OUTV = INV * FACTOR + OFFSET



Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)
Acondicionamiento del valor real (Página 10171)
Control anticipativo (Página 10175)

Parámetros de entrada de SCALE_M

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Magnitud de entrada Rango técnico
FACTOR	4.0	REAL	1.0	Factor de escala
OFFSET	8.0	REAL	0.0	Offset Rango técnico

Parámetros de salida de SCALE_M

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV	12.0	REAL	0.0	Magnitud de salida

SP_GEN

Descripción de SP_GEN

Para la consigna predeterminada manualmente es posible cambiar un valor de salida con la instrucción SP_GEN mediante dos entradas. Para poder realizar un cambio graduado finamente la instrucción debe tener un tiempo de muestreo ≤100 ms.

Modo de funcionamiento

DFOUT_ON, OUTVUP y OUTVDN tienen la siguiente influencia en OUTV:

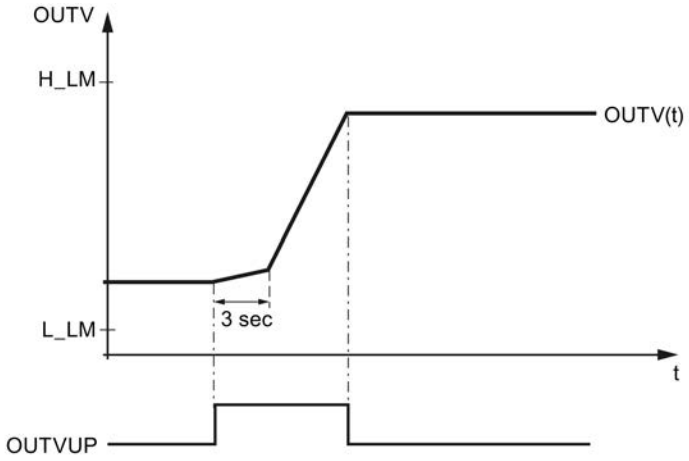
DFOUT_ON	OUTVDN	OUTVUP	OUTV
TRUE	Cualquiera	cualquiera	DF_OUTV
FALSE	TRUE	TRUE	OUTV sin modificar
	FALSE		OUTV creciente
	TRUE	FALSE	OUTV decreciente
	FALSE		OUTV sin modificar

Si está activado DFOUT_ON = TRUE, en la salida se deposita DF_OUTV. El cambio de OUTV es escalonado. La conmutación a DFOUT_ON = FALSE es suave.

A través de la entrada OUTVUP, la magnitud de salida OUTV se puede aumentar de forma continua dentro de las limitaciones H_LM y L_LM y reducirse mediante OUTVDN. La velocidad de cambio de OUTV depende de los límites.

- Durante los primeros 3 segundos después de activar OUTVUP o OUTVDN rige:
  dLMN/dt = (LMN_HLM - LMN_LLM) / 100 s
- a continuación
  dLMN/dt = (LMN_HLM - LMN_LLM) / 10 s

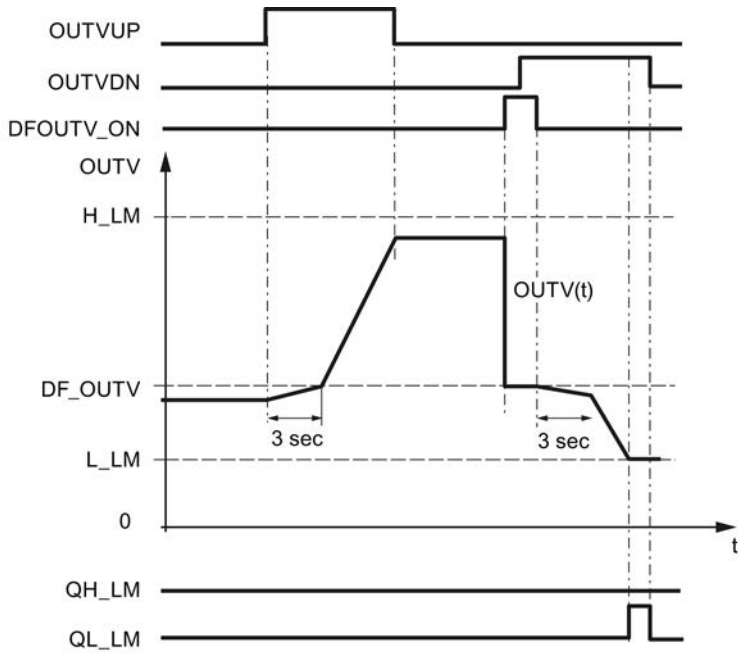
La figura muestra el cambio del valor de salida activando OUTVUP.



OUTVUP y OUTVDN solo se ejecutan si DFOUT_ON = FALSE.

La magnitud de salida OUTV está limitada por H_LM y L_LM. Eso se indica en QH_LM y QL_LM.

La figura muestra la influencia de OUTV mediante OUTVUP, OUTVDN y DFOUTV_ON.



Rearranque

En el rearranque la salida OUTV se pone a 0.0. Si está activado DFOUT_ON = TRUE la salida es DF_OUTV. Los límites también están activados durante el rearranque.

Límites internos del bloque

No se limitan los valores dentro de los bloques; no se comprueban los parámetros.

Consulte también

Acondicionar consigna (Página 10169)

Parámetros de entrada de SP_GEN

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
DF_OUTV	0.0	REAL	0.0	Preasignación de la magnitud de salida Rango técnico
H_LM	4.0	REAL	100.0	Límite superior Rango técnico > L_LM
L_LM	8.0	REAL	0.0	Límite inferior Rango técnico < H_LM

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTVUP	12.0	BOOL	FALSE	OUTV Alta
OUTVDN	12.1	BOOL	FALSE	OUTV Baja
DFOUT_ON	12.2	BOOL	FALSE	Activar la preasignación de la magnitud de salida
COM_RST	12.3	BOOL	FALSE	Rearranque
CYCLE	14.0	TIME	T#100ms	Tiempo de muestreo

Parámetros de salida de SP_GEN

Parámetro	Offset	Tipo de datos	Asignación previa (tras rearranque)	Descripción
OUTV	18.0	REAL	0.0	Magnitud de salida
QH_LM	22.0	BOOL	FALSE	El límite superior ha reaccionado
QL_LM	22.1	BOOL	FALSE	El límite inferior ha reaccionado

SPLT_RAN

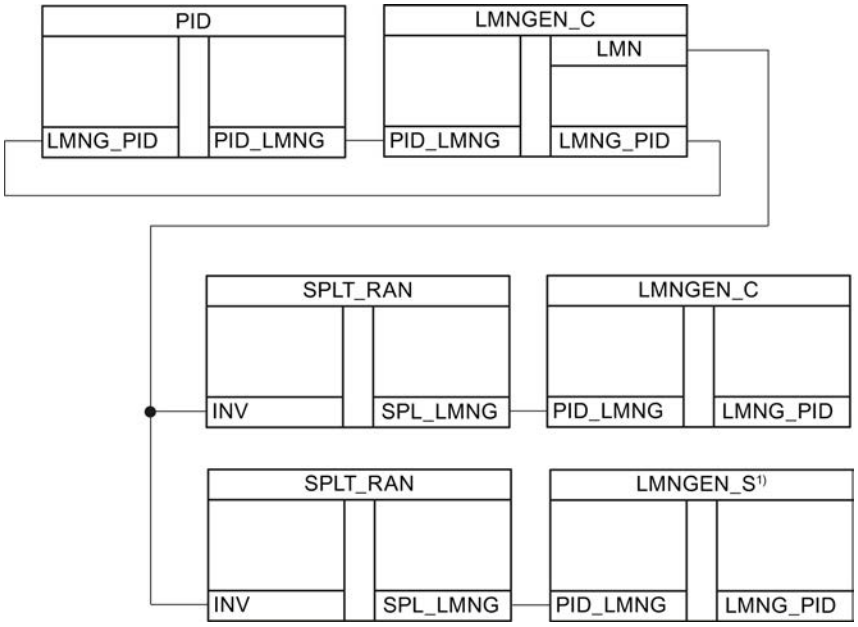
Descripción de SPLT_RAN

La instrucción es necesaria p. ej. para dividir una regulación de temperatura en un actuador para refrigeración y otro para calefacción.

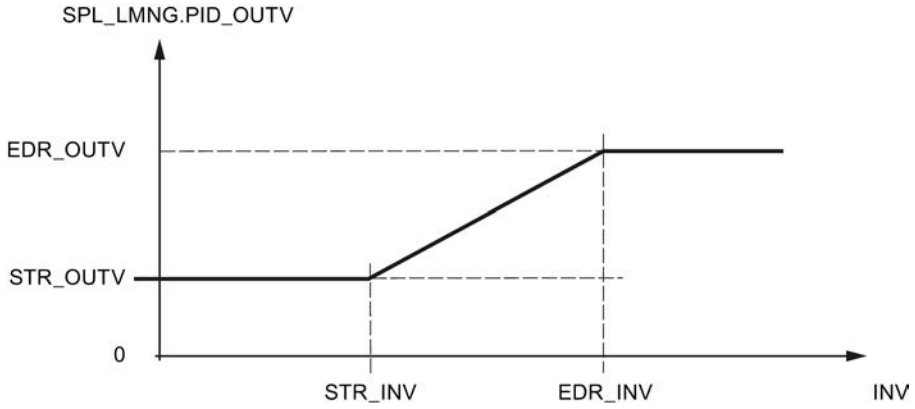
Modo de funcionamiento

La variable manipulada de un regulador PID se divide en varios rangos. Por cada rango la instrucción debe llamarse una vez e interconectarse con uno de los bloques de procesamiento

de variable manipulada LMNGEN_C o LMNGEN_S. Si interconecta LMNGEN_S, la realimentación de posición debe estar activada (LMNR_ON = TRUE).



Si el valor de entrada INV está dentro del rango de STR_INV y EDR_INV, el valor de salida SPL_LMNG.PID_OUTV se emite en el rango STR_OUTV y EDR_OUTV. Si INV < STR_INV,, la salida es STR_OUTV. Si INV > STR_INV,, la salida es EDR_OUTV.



Rearranque

La instrucción no dispone de rutina de rearranque.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada de SPLT_RAN

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV	0.0	REAL	0.0	Variable de entrada Rango técnico
STR_INV	4.0	REAL	0.0	Inicio de rango INV Rango técnico
EDR_INV	8.0	REAL	50.0	Fin de rango INV Rango técnico
STR_OUTV	12.0	REAL	0.0	Inicio de rango OUTV Rango técnico
EDR_OUTV	16.0	REAL	100.0	Fin de rango OUTV Rango técnico

Parámetros de salida de SPLT_RAN

Parámetro	Offset	Tipo de datos	Descripción
SPL_LMNG	20.0	STRUC	Interfaz PID-LMNGEN

SWITCH

Descripción de SWITCH

La instrucción se utiliza como multiplexor de entrada y/o salida de 2 magnitudes de entrada/salida.

Modo de funcionamiento

La instrucción aplica uno de los 2 valores de entrada analógicos en uno de 2 valores de salida. Los conmutadores INV1_ON y OUTV1_ON determinan el modo de funcionamiento de la instrucción.

INV1_ON	OUTV1_ON	OUTV1	OUTV2
FALSE	FALSE	Sin cambios	INV2
TRUE	FALSE	Sin cambios	INV1
FALSE	TRUE	INV2	Sin cambios
TRUE	TRUE	INV1	Sin cambios

Sin cambios significa que en el parámetro de salida se conserva el valor del último ciclo.

Rearranque

En el rearranque se activan OUTV1 = 0.0 y OUTV2 = 0.0.

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada de SWITCH

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
INV1	0.0	REAL	0.0	Magnitud de entrada 1 Rango técnico
INV2	4.0	REAL	0.0	Magnitud de entrada 2 Rango técnico
INV1_ON	8.0	BOOL	FALSE	La magnitud de entrada INV1 se conecta directamente
OUTV1_ON	8.1	BOOL	FALSE	La magnitud de salida OUTV1 se conecta directamente
COM_RST	8.2	BOOL	FALSE	Rearranque

Parámetros de salida de SWITCH

Parámetro	Offset	Tipo de datos	Asignación previa	Descripción
OUTV1	10.0	REAL	0.0	Magnitud de salida 1
OUTV2	14.0	REAL	0.0	Magnitud de salida 2

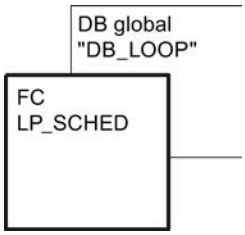
LP_SCHED_M

Descripción de LP_SCHED_M

Cuando se tienen que llamar muchos reguladores con diferentes tiempos de muestreo, especialmente regulaciones lentas con grandes tiempos de muestreo, el modelo de clases de prioridad no es suficiente respecto a los niveles de alarma cíclica útiles. El distribuidor de llamadas del regulador (LP_SCHED) permite el posicionamiento de varios reguladores con diferentes tiempos de muestreo en un nivel de alarma cíclica. Los reguladores individuales se llaman cíclicamente durante sus tiempos de muestreo.

La utilización del distribuidor de llamadas no es obligatoria. Las instrucciones de regulador también se pueden llamar directamente desde el OB sin la función de distribución.

La distribución de llamadas de varios reguladores en un nivel de alarma cíclica se realiza con la instrucción LP_SCHED. La instrucción se debe llamar **antes** de todos los lazos de regulación. Los datos para las llamadas del regulador se guardan en un bloque de datos global (DB_LOOP).



El distribuidor de llamadas procesa el bloque de datos global y pone los bits ENABLE, dependiendo del orden y de los tiempos de muestreo parametrizados de los reguladores. Por este motivo la señal de temporización del nivel de alarma cíclica se reduce. Los lazos de regulación individuales de este nivel de alarma cíclica se llaman y procesan en función del tiempo de muestreo ajustado. Tras la llamada del bloque el bit ENABLE se debe resetear. Las llamadas del bloque y el reseteado de los bits ENABLE se deben programar.

La llamada de algunos lazos de regulación se puede bloquear manualmente. También es posible resetear sólo lazos de regulación individuales (rearranque).

Nota

La instrucción no comprueba si existe realmente un bloque de datos global con el número DB_NBR ni si el parámetro GLP_NBR (número más alto de lazos de regulación) concuerda con la longitud del bloque de datos. Al no parametrizar correctamente, la CPU cambia al estado STOP y muestra "Error interno del sistema".

Consulte también

Crear un regulador de impulsos (Página 10179)

Funcionamiento de LP_SCHED_M

Rearranque

Con COM_RST = TRUE se ajustan las siguientes asignaciones previas:

Número de lazo de regulación actual: ALP_NBR = 0

Los datos de llamada de todos los lazos de regulación hasta **GLB_NBR** se preasignan del siguiente modo:

Liberación:	ENABLE =	NOT MAN_DIS
Tiempo de muestreo:	CYCLE =	GV(MAN_CYC)

Rearranque:	COM_RST =	TRUE
Número de llamada local:	ILP_COU =	0
GV(MAN_CYC) = MAN_CYC se redondea a un múltiple entero de TM_BASE*GLP_NBR		

La llamada del lazo de regulación mediante LP_SCHED_M

La instrucción LP_SCHED se integra en el esquema de llamada de la CPU mediante tres parámetros de entrada. En la entrada TM_BASE se indica la señal de temporización del nivel de alarma cíclica. La llamada del procesamiento de los lazos de regulación asignados se realiza a través de una llamada del bloque condicionada desde un nivel de alarma cíclica (p. ej., B. OB 35). Para ello se sondean los bits ENABLE en el bloque de datos global.

Si la llamada se realiza desde el nivel de rearranque, se activa la entrada COM_RST = TRUE. En el nivel de alarma cíclica esta llamada debe resetearse en FALSE. El bloque de datos global (véase la tabla anterior) con los datos relevantes de tiempo de los lazos de regulación en el nivel de alarma cíclica correspondiente se asigna mediante el parámetro de entrada DB_NBR.

Parametrización de la llamada del lazo de regulación (DB global)

La parametrización del distribuidor de llamadas del regulador debe realizarse **sin** ayuda del software de configuración.

El bloque de datos (DB_LOOP) contiene bien un parámetro que determina el número de los lazos de regulación que hay que procesar en el nivel de alarma cíclica (256 como máximo) o bien un parámetro que muestra los lazos de regulación que se procesan actualmente:

GLP_NBR	Número de lazo de regulación más alto
ALP_NBR	Número del lazo de regulación procesado en el ciclo

El número del lazo de regulación correspondiente resulta del posicionamiento de sus datos de llamada en el orden de las entradas en el DB.

Los datos de llamada de los lazos de regulación individuales están situados de forma estructurada en el campo LOOP_DAT. Si desea añadir lazos de regulación, es necesario adaptar la longitud del campo de LOOP_DAT. Para ello hay que adaptar el tipo de datos ARRAY en la vista de declaración. Por ejemplo para 10 lazos de regulación hay que introducir ARRAY[1..10]. Además hay que adaptar el parámetro GLP_NBR en la vista de datos. Nunca debe superar la longitud del campo.

Los parámetros COM_RST y CYCLE en los datos de llamada se deben interconectar con los parámetros correspondientes en el FB del lazo de regulación llamado. Esta interconexión la debe programar el usuario. Si el parámetro ENABLE está puesto, se debe llamar el lazo de regulación correspondiente. Tras la llamada del regulador el bit ENABLE se debe resetear. La llamada del regulador condicionada y el reseteado del bit ENABLE se deben programar por el usuario.

Con ayuda del parámetro manualmente ajustable MAN_CYC/MAN_DIS/MAN_CRST se puede controlar si un lazo de regulación se puede llamar o no. Estos datos de llamada se pueden cambiar online, es decir, durante el funcionamiento, si sólo se sobrescriben los parámetros y no se genera de nuevo todo el DB. Esto significa:

MAN_CYC	Tiempo de muestreo del regulador correspondiente (se redondea en CYCLE a un múltiplo entero de TM_BASE * GLP_NBR)
MAN_DIS	Bloqueo de la llamada del regulador
MAN_CRST	Rearranque para este regulador

El procesamiento de la llamada

El lazo de regulación correspondiente se procesa según la parametrización anterior del DB dependiendo del valor de la señal ENABLE de los datos de la llamada del regulador.

El bloque de datos se procesa desde arriba hacia abajo. El distribuidor de llamadas sigue en cada ciclo un número de lazos de regulación (ALP_NBR) en el orden del DB. El contador interno ILP_COU disminuye en 1. Si ILP_COU = 0, el distribuidor de llamadas activa el bit ENABLE del lazo de regulación correspondiente. El reseteado del bit ENABLE debe programarlo el usuario después de la llamada del regulador.

El parámetro MAN_CYC se transfiere a CYCLE durante el procesamiento:

CYCLE = GV (MAN_CYC), GV = múltiplo entero

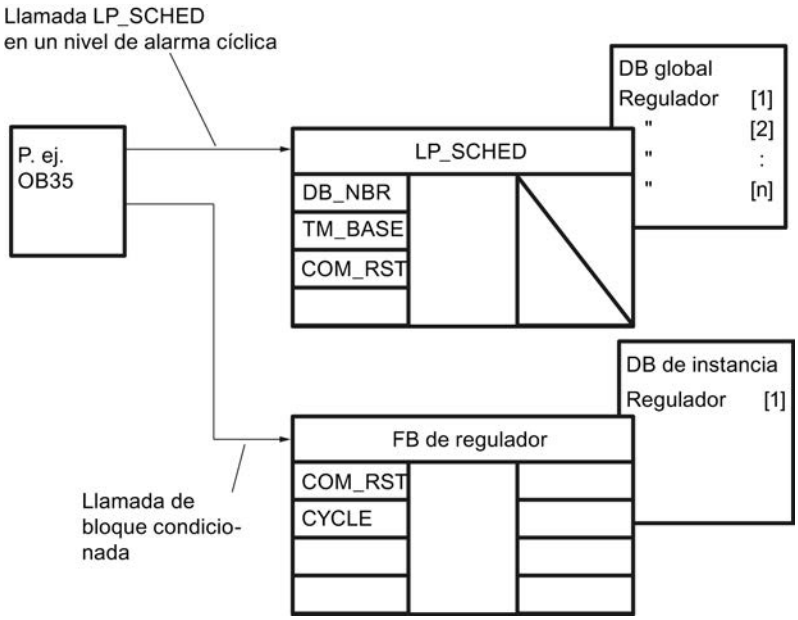


Figura 9-9 Principio de la llamada del regulador a través del distribuidor de llamadas LP_SCHED

- Bloquear lazos de regulación individuales:
  Si se pone el bit "MAN_DIS" mediante una parametrización en el DB, el bit ENABLE se resetea en FALSE y se elimina el lazo de regulación correspondiente del procesamiento en el distribuidor de llamadas.
- Resetear lazos de regulación individuales (rearranque):
  Cuando se activa el bit "MAN_CRST" mediante la parametrización en el DB, entonces COM_RST = TRUE y a continuación se desactiva MAN_CRST. El lazo de regulación correspondiente se procesa con su rutina de rearranque. En el siguiente ciclo de llamada también se desactiva automáticamente COM_RST = FALSE.

Nota

Cuando se añade o borra un lazo de regulación, es decir, cuando se regenera el DB completo, pero no se desea que el distribuidor de llamadas se procese en el rearranque, entonces se debe preasignar con cero los contadores internos de los lazos de regulación (ILP_COU[n]) y el parámetro para el número actual de lazos de regulación ALP_NBR.

Condiciones para la llamada del lazo de regulación mediante LP_SCHED

Para garantizar que los periodos entre las llamadas de cierto regulador se mantengan constantes y para utilizar la CPU de forma constante, sólo se debe procesar un lazo de regulación por señal de temporización del nivel de alarma cíclica. Por ello, durante la parametrización de los tiempos de muestreo MAN_CYC se deben cumplir las siguientes condiciones en relación con la señal de temporización (TM_BASE):

- Los tiempos de procesamiento de los lazos de regulación individuales deben ser menores que la señal de temporización (TM_BASE) del nivel de alarma cíclica.
- El tiempo de muestreo de un lazo de regulación (MAN_CYC) debe ser un múltiplo entero (GV) del producto de la base de tiempo y el número de los reguladores por procesar (GLP_NBR):
  MAN_CYC = GV (TM_BASE * GLP_NBR)

Ejemplo para la distribución de llamadas

El siguiente ejemplo muestra el orden de las llamadas de cuatro lazos de regulación en un nivel de alarma cíclica (véase la siguiente figura). Sólo se procesa un lazo de regulación por

unidad de la base de tiempo. El orden de las llamadas y por ello los tiempos de desplazamiento (VZ1 ... VZ5) resultan del orden de los datos de llamada dentro del DB global.

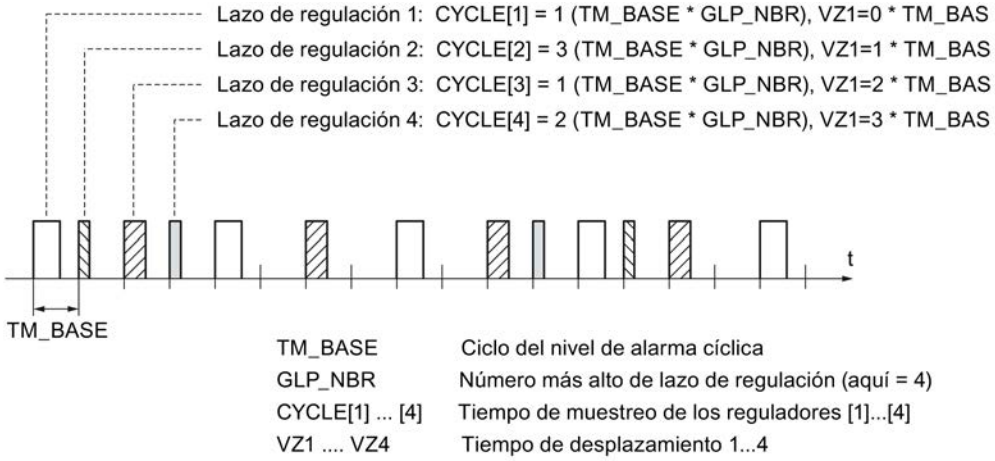


Figura 9-10 Orden de llamada de cuatro lazos de regulación procesados con diferentes frecuencias

Límites internos del bloque

Los valores de los parámetros de entrada no se limitan en la instrucción; no se comprueban los parámetros.

Parámetros de entrada de LP_SCHED_M

Parámetro	Tipo de datos	Descripción
TM_BASE	TIME	Base de tiempo >= 1ms
COM_RST	BOOL	Rearranque
DB_NBR	BLOCK_DB	Número del bloque de datos

Bloque de datos global DB_LOOP_M

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
GLP_NBR	INT	2	Número máximo de reguladores 1 - 256
ALP_NBR	INT	0	Número del regulador actual 0 - 256

Parámetro	Tipo de datos	Ajuste predetermina do	Descripción
LOOP_DAT	Array [1...GLP_N BR] of Struct		Estructura de los datos de los diferentes reguladores. La matriz debe ser declarada por 1...GLP_NBR.
LOOP_DAT[1].MAN_CYC	TIME	T#1s	Datos del regulador 1: tiempo de muestreo manual >= 1ms
LOOP_DAT[1].MAN_DIS	BOOL	FALSE	Datos del regulador 1: bloquear manualmente la llamada del regulador
LOOP_DAT[1].MAN_CRST	BOOL	FALSE	Datos del regulador 1: iniciar manualmente el rearranque
LOOP_DAT[1].ENABLE	BOOL	FALSE	Datos del regulador 1: liberación del regulador
LOOP_DAT[1].COM_RST	BOOL	FALSE	Datos del regulador 1: rearranque
LOOP_DAT[1].ILP_COU	INT	0	Datos del regulador 1: contador interno
LOOP_DAT[1].CYCLE	TIME	T#1s	Datos del regulador 1: tiempo de muestreo >= 1ms
LOOP_DAT[2].MAN_CYC			Datos del regulador 2: tiempo de muestreo manual
...			

Motion Control

Easy Motion Control

MC_Init

Descripción

La instrucción MC_Init prepara la inicialización de todos los bloques de un eje, activando todos los bits del campo de bits "Init.Ix" del DB de eje (véase "Inicialización y cambios de parámetros (Página 11257)").

Funcionamiento

La instrucción MC_Init **impide un arranque inintencionado del eje** activando en el DB de eje las señalizaciones de error "Error" = TRUE y "Err.StoppedMotion" = TRUE. Si desea desplazar el eje tras la llamada a la instrucción MC_Init, debe confirmar primero las señalizaciones de error (véase "Señalizaciones de error y acuse de errores en el DB de eje (Página 11273)").

La instrucción MC_Init comprueba los parámetros del DB de eje. Si se encuentra un error de parametrización, en el DB de eje para la señalización (agrupada) de errores se activan los bits "Error" y "Err.ConfigErr", así como uno de los bits de la estructura "ConfigErr" para una señalización más precisa del error.

Los errores de parametrización **no se pueden acusar**. Es preciso volver a llamar a la instrucción MC_Init tras corregir el error.

Para más información sobre el concepto de error, véase "Señalizaciones de error y acuse de errores en el DB de eje (Página 11273)" y "Errores de parametrización (Página 11280)".

Llamada

Nota

No se permite la llamada de la instrucción MC_Init cuando el eje se encuentra parado.

Efectúe una llamada condicionada de la instrucción MC_Init en el mismo bloque de programa en el que también se llama el resto de instrucciones de este eje. En los siguientes casos debe llamarse la instrucción MC_Init:

- Tras cada arranque de la CPU (OB 100 y OB 101)
- Si se modifica un parámetro sin el software de configuración señalado con el siguiente símbolo en "Configuración del objeto tecnológico Eje (Página 11228)":

Nota

Resetee la condición utilizada para la llamada de la instrucción MC_Init una vez que se haya ejecutado la llamada. De lo contrario, los bloques Easy Motion Control repetirán siempre únicamente su inicialización, pues los bits de inicialización del DB de eje se vuelven a setear.

MC_MoveAbsolute

Descripción

La instrucción MC_MoveAbsolute permite mover un eje sincronizado al destino absoluto especificado en "Position".

Funcionamiento

Al detectarse un flanco ascendente en la entrada "Execute", la instrucción MC_MoveAbsolute determina el sentido de desplazamiento y acelera el eje en este sentido. Al alcanzarse el punto de frenado, el eje se frena. La velocidad especificada no tiene que alcanzarse durante el desplazamiento.

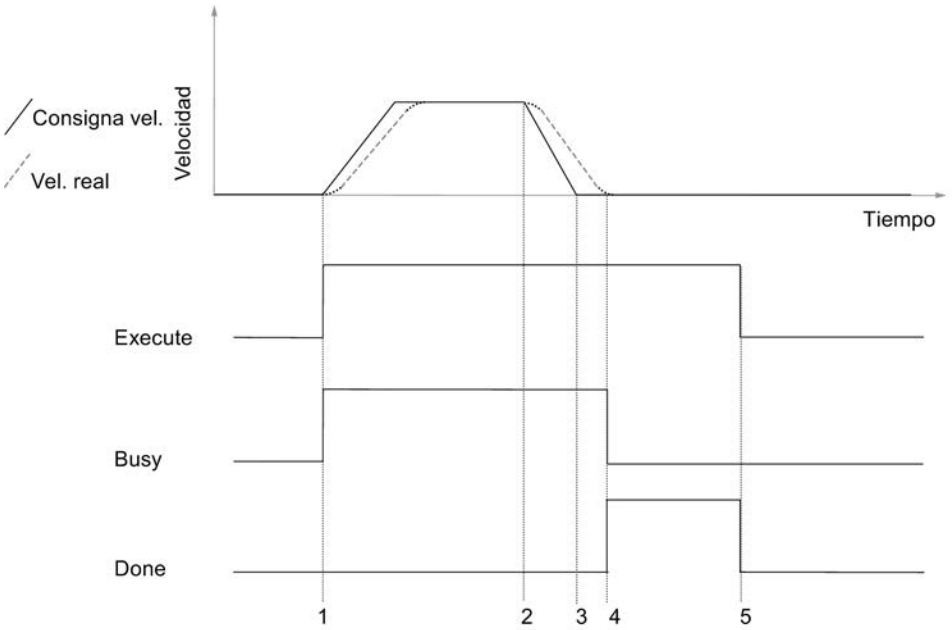
Particularidad en un eje lineal:

- El sentido del desplazamiento viene determinado por la posición consignada durante el arranque y el destino.

Particularidades en un eje rotativo:

- La instrucción MC_MoveAbsolute sólo permite efectuar un desplazamiento a partir de la posición de velocidad cero del eje que sea inferior a una vuelta del eje rotativo. Para posicionamientos de varias vueltas del eje rotativo, utilice la instrucción MC_MoveRelative (Página 7328).
- Selección de sentido para arranque desde la posición de velocidad cero del eje:
  - Sentido positivo: el desplazamiento al destino se efectúa en sentido positivo,
  - Sentido negativo: el desplazamiento al destino se efectúa en sentido negativo,
  - Recorrido más corto: el desplazamiento al destino se efectúa en el sentido para el que el recorrido entre el punto inicial y el destino sea ≤ ½ vuelta del eje rotativo.
- Selección de sentido al retomar un desplazamiento:
  - Sentido positivo: el desplazamiento al destino se efectúa en sentido positivo. Si hasta ese momento el desplazamiento era en sentido negativo, el eje se frena y se efectúa el desplazamiento al destino en sentido positivo. Si el destino se alcanza exactamente al frenar con sentido de desplazamiento negativo, el eje se desplazará a continuación una vuelta en sentido positivo.
  - Sentido negativo: el desplazamiento al destino se efectúa en sentido negativo. Si hasta ese momento el desplazamiento era en sentido positivo, el eje se frena y se efectúa el desplazamiento al destino en sentido negativo. Si el destino se alcanza exactamente al frenar con sentido de desplazamiento positivo, el eje se desplaza a continuación una vuelta en sentido negativo.
  - Recorrido más corto: el desplazamiento al destino se efectúa en el sentido para el que el recorrido entre el punto inicial y el destino sea ≤ ½ vuelta del eje rotativo. Si el recorrido de frenado es superior, el eje se desplaza hasta el destino en el sentido original.
  - Si ha seleccionado los parámetros de desplazamiento de manera que el recorrido de frenado del desplazamiento anterior era mayor que una vuelta del eje rotativo, se efectuará el desplazamiento directamente al destino tras la inversión de sentido.
  - El recorrido total entre el punto inicial y el destino puede ser mayor que una vuelta del eje rotativo.

Diagrama de evolución de la señal de MC_MoveAbsolute



1 Inicio de la instrucción de desplazamiento

2 Punto de frenado

3 La consigna ha alcanzado la posición de destino

4 El eje ha alcanzado la zona de destino

5 Al resetear "Execute", se resetea "Done". El proceso ha finalizado. Cuando se resetea "Execute" antes de alcanzarse la posición de destino, se activa "Done" durante un ciclo al alcanzarse la posición de destino, y después se vuelve a resetear.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
Execute	INPUT	BOOL	Inicio del posicionamiento con un flanco ascendente.
Position	INPUT	REAL	Posición de destino absoluta en [unidad de longitud] Con eje rotativo: Posición del eje rotativo ≤ Posición < Fin del eje rotativo
Velocity	INPUT	REAL	Velocidad del eje en [unidad de longitud/s]
Acceleration	INPUT	REAL	Aceleración del eje en [unidad de longitud/s^2]
Deceleration	INPUT	REAL	Deceleración del eje en [unidad de longitud/s^2]

Parámetro	Declaración	Tipo de datos	Descripción
Direction	INPUT	INT	Selección de sentido para desplazamiento con eje rotativo. -1 = sentido negativo 0 = recorrido más corto 1 = sentido positivo
Busy	OUTPUT	BOOL	1 = petición en curso.
Done	OUTPUT	BOOL	1 = petición finalizada sin errores. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
CommandAborted	OUTPUT	BOOL	1 = la petición ha sido cancelada por otra petición. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
Error	OUTPUT	BOOL	1 = la petición ha sido cancelada por haberse detectado un error de eje. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.

UDT2

En las propiedades del bloque de MC_MoveAbsolute se encuentra consignado y reservado un tipo de datos UDT2. Por tanto, no debe utilizarse en el mismo proyecto ningún tipo de datos UDT2 propio. De lo contrario pueden producirse mensajes de error e incoherencias dentro del proyecto. Si es necesario utilizar un tipo de datos UDT2, este UDT deberá renombrarse. Por ejemplo, como UDT3.

MC_MoveRelative

Descripción

La instrucción MC_MoveRelative permite mover un eje un recorrido especificado en "Distance" relativo a la consigna al inicio del desplazamiento. El signo de "Distance" determina el sentido.

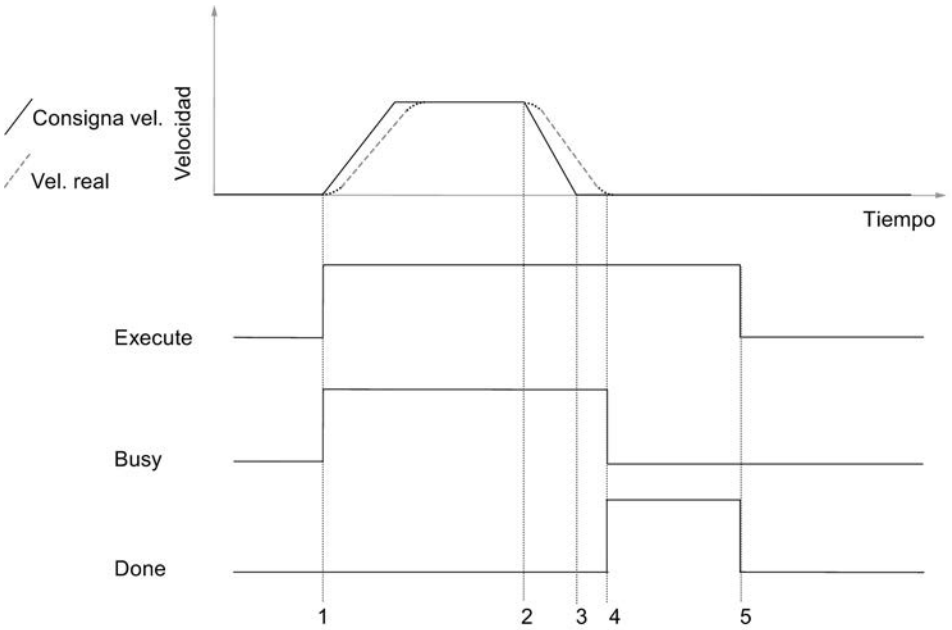
Funcionamiento

Al detectarse un flanco ascendente en la entrada "Execute", la instrucción MC_MoveRelative acelera el eje en el sentido indicado en "Distance". Al alcanzarse el punto de frenado, el eje se frena. La velocidad especificada no tiene que alcanzarse durante el desplazamiento. El destino viene determinado por la posición consignada al inicio del desplazamiento y del recorrido que se debe realizar.

Particularidad en ejes rotativos:

- Al especificarse valores superiores a una vuelta del eje rotativo en "Distance", se pueden efectuar desplazamientos de varias vueltas del eje rotativo.

Diagrama de evolución de la señal de MC_MoveRelative



1 Inicio de la instrucción de desplazamiento

2 Punto de frenado

3 La consigna ha alcanzado la posición de destino

4 El eje ha alcanzado la zona de destino

5 Al resetear "Execute", se resetea "Done". El proceso ha finalizado. Cuando se resetea "Execute" antes de alcanzarse la posición de destino, se activa "Done" durante un ciclo al alcanzarse la posición de destino, y después se vuelve a resetear.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
Execute	INPUT	BOOL	Inicio del posicionamiento con un flanco ascendente.
Distance	INPUT	REAL	Recorrido que debe efectuarse en [unidad de longitud], referido a la última consigna de posición. El signo define el sentido de desplazamiento.
Velocity	INPUT	REAL	Velocidad del eje en [unidad de longitud/s]
Acceleration	INPUT	REAL	Aceleración del eje en [unidad de longitud/s^2]
Deceleration	INPUT	REAL	Deceleración del eje en [unidad de longitud/s^2]
Busy	OUTPUT	BOOL	1 = petición en curso.

Parámetro	Declaración	Tipo de datos	Descripción
Done	OUTPUT	BOOL	1 = petición finalizada sin errores. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
CommandAborted	OUTPUT	BOOL	1 = la petición ha sido cancelada por otra petición. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
Error	OUTPUT	BOOL	1 = la petición ha sido cancelada por haberse detectado un error de eje. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.

UDT2

En las propiedades del bloque de MC_MoveRelative se encuentra consignado y reservado un tipo de datos UDT2. Por tanto, no debe utilizarse en el mismo proyecto ningún tipo de datos UDT2 propio. De lo contrario pueden producirse mensajes de error e incoherencias dentro del proyecto. Si es necesario utilizar un tipo de datos UDT2, este UDT deberá renombrarse. Por ejemplo, como UDT3.

MC_MoveJog

Descripción

La instrucción MC_MoveJog permite desplazar un eje mediante dos entradas de sentido controladas por nivel "JogPos" y "JogNeg".

Funcionamiento

Al activar una entrada de sentido, MC_MoveJog acelera el eje primeramente hasta la velocidad indicada. Mientras esté activada la entrada de sentido, el eje se desplazará en este sentido. Si se vuelve a resetear la entrada de sentido, el eje se frena.

Las dos entradas de sentido se evalúan con las siguientes dependencias:

- Mientras haya un sentido seleccionado, ajustar el sentido opuesto carece de efecto. No se genera ningún error.
- Para cambiar el sentido durante el desplazamiento, es necesario ajustar la otra entrada de sentido a TRUE y al mismo tiempo la entrada seleccionada originalmente a FALSE.
- Si al inicio de un desplazamiento se han activado las dos entradas, se generará un error de inicio ("Err.StartErr", véase Error con parada suave (Página 11275)).

Particularidades en un eje lineal:

- Si se rebasa el máximo recorrido de desplazamiento con un eje lineal, la instrucción MC_MoveJog frena el eje hasta parada con el mensaje de error acusable "Err.DistanceErr" (véase Desarrollo de los movimientos de desplazamiento (Página 11258)). Tras el acuse del error, el eje puede volver a desplazarse.
- Un eje lineal **no sincronizado** no se vigila y por tanto puede desplazarse hasta su fin físico sin frenar. Por tanto, es preciso terminar el desplazamiento a tiempo.

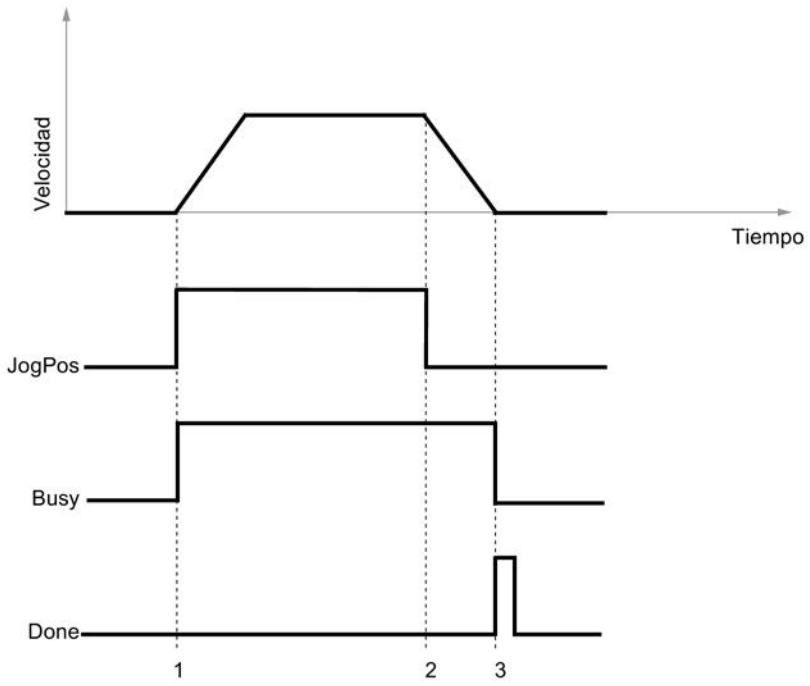
> ⚠ ADVERTENCIA
>
> **Eje lineal no sincronizado**
>
> Se pueden producir daños personales y materiales.
>
> La instrucción MC_MoveJog permite el desplazamiento hasta el fin físico del eje con un eje lineal no sincronizado.
>
> Para evitar daños personales y materiales es necesario tomar las medidas siguientes:
>
> - Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Sólo así puede garantizarse que en caso de un fallo del software o del equipo sea posible desconectar la instalación de modo seguro.
> - Instale finales de carrera de seguridad que actúen directamente en las etapas de potencia de todos los accionamientos.
> - Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.

- Un eje lineal **sincronizado** se frena al final de la zona de trabajo y se detiene en el final de carrera por software. Se emite el mensaje de error acusable "Err.DistanceErr", así como el mensaje de error "Err.SWLimitMinExceeded" o "Err.SWLimitMaxExceeded". Tras el acuse del error sólo se permite un desplazamiento en sentido contrario.

Particularidad en un eje rotativo:

- El recorrido de desplazamiento de la instrucción MC_MoveJog no está limitado para un eje rotativo (giro continuo).

Diagrama de evolución de la señal de MC_MoveJog



1 El movimiento se inicia mediante activación de "JogPos".
2 El movimiento se detiene mediante reset de "JogPos".
3 Con la parada del eje se resetea "Busy" y se activa "Done" durante un ciclo.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
JogPos	INPUT	BOOL	1 = movimiento en sentido positivo, mientras esté activado "JogPos".
JogNeg	INPUT	BOOL	1 = movimiento en sentido negativo, mientras esté activado "JogNeg".
Velocity	INPUT	REAL	Velocidad del eje en [unidad de longitud/s]
Acceleration	INPUT	REAL	Aceleración del eje en [unidad de longitud/s^2]
Deceleration	INPUT	REAL	Deceleración del eje en [unidad de longitud/s^2]
Busy	OUTPUT	BOOL	1 = petición en curso.
Done	OUTPUT	BOOL	1 = petición finalizada sin errores. El bit se activa tras alcanzarse la parada del eje durante exactamente una llamada de la instrucción.

Parámetro	Declaración	Tipo de datos	Descripción
CommandAborted	OUTPUT	BOOL	1 = la petición ha sido cancelada por otra petición. El bit permanece activado hasta que se resetea "JogPos" o "JogNeg".
Error	OUTPUT	BOOL	1 = la petición ha sido cancelada por haberse detectado un error de eje. El bit permanece activado hasta que se resetea "JogPos" o "JogNeg".

UDT2

En las propiedades del bloque de MC_MoveJog se encuentra consignado y reservado un tipo de datos UDT2. Por tanto, no debe utilizarse en el mismo proyecto ningún tipo de datos UDT2 propio. De lo contrario pueden producirse mensajes de error e incoherencias dentro del proyecto. Si es necesario utilizar un tipo de datos UDT2, este UDT deberá renombrarse. Por ejemplo, como UDT3.

MC_Home

Descripción

La instrucción MC_Home se utiliza:

- para la búsqueda del punto de referencia en encóders incrementales ("Velocity" > 0)
- para posicionar el punto de referencia en encóders incrementales y absolutos ("Velocity" = 0)
  Para un encóder absoluto, el posicionamiento del punto de referencia también se denomina reajuste del encóder absoluto.

Funcionamiento

Búsqueda del punto de referencia ("Velocity" > 0)

Tras activar "Execute", el eje se desplaza con la velocidad y sentido elegidos hasta que el módulo de lectura de recorrido y el driver de entrada detecten que se ha alcanzado el punto de referecia. Al alcanzarse el punto de referencia, el valor real de posición se ajusta al valor de "Position" y el eje se frena. El eje está ahora sincronizado ("Sync" = TRUE), pero no está en el punto de referencia.

Si se rebasa el máximo recorrido (Página 11258) durante la búsqueda del punto de referencia, la instrucción MC_Home frena el eje hasta pararse (véase Desarrollo de movimientos de desplazamiento (Página 11258)) y se activan los mensajes de error "Error" y "Err.DistanceErr" (véase Error con parada suave (Página 11275)).

Si la búsqueda del punto de referencia se detiene antes de alcanzarse éste, el eje no está sincronizado ("Sync" = FALSE).

La instrucción MC_Home no se detiene en el fin del eje y tampoco ejecuta un cambio de sentido automático.

⚠ PELIGRO
Si no se encuentra ningún punto de referencia, el cliente deberá tomar las medidas necesarias para que el eje se detenga antes del fin físico (p. ej., interruptor de parada de emergencia, final de carrera de seguridad).

Posicionamiento del punto de referencia ("Velocity" = 0)

Ajuste la velocidad "Velocity" a 0. Tras ajustar "Execute", el valor real de posición se ajusta al valor de "Position", y al mismo tiempo se activa en el DB de eje el bit de estado "Sync" = TRUE.

Tras sustituirse el encóder absoluto o si se realizan cambios mecánicos, es preciso volver a realizar el posicionamiento del punto de referencia, incluso aunque el eje todavía esté sincronizado.

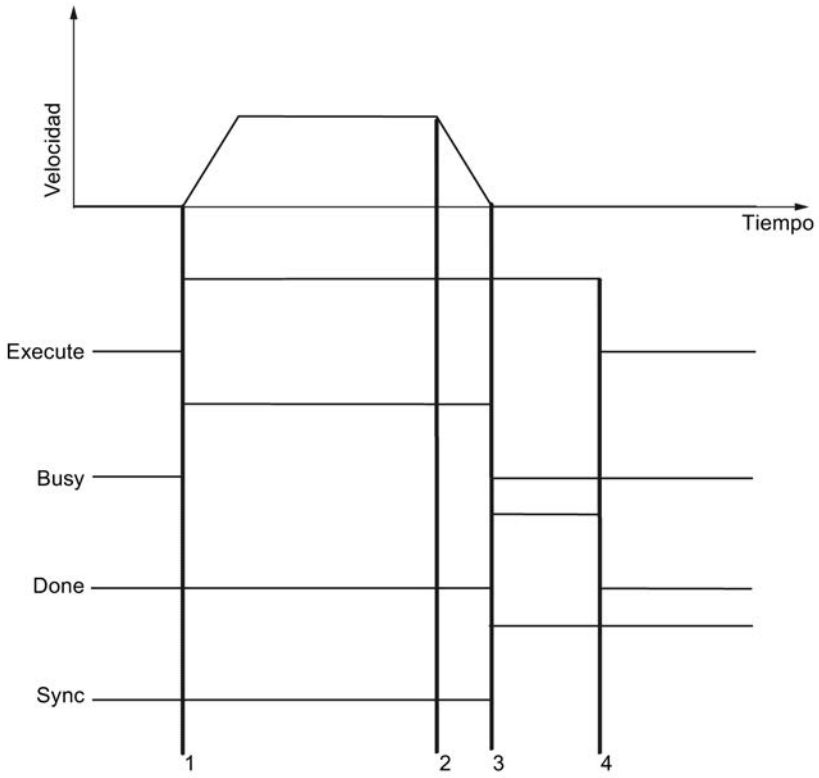
Nota

Para un encóder absoluto rige lo siguiente:

El valor del encóder en el punto de referencia se guarda en el DB de eje. Por tanto, tras posicionarse el punto de referencia debe incluirse el DB de eje online en la gestión de datos offline. De lo contrario, tras cargarse el DB en la CPU, el eje dejará de estar sincronizado.

Tenga en cuenta las indicaciones sobre la Carga del DB de eje (Página 11262).

Diagrama de evolución de la señal para la búsqueda del punto de referencia

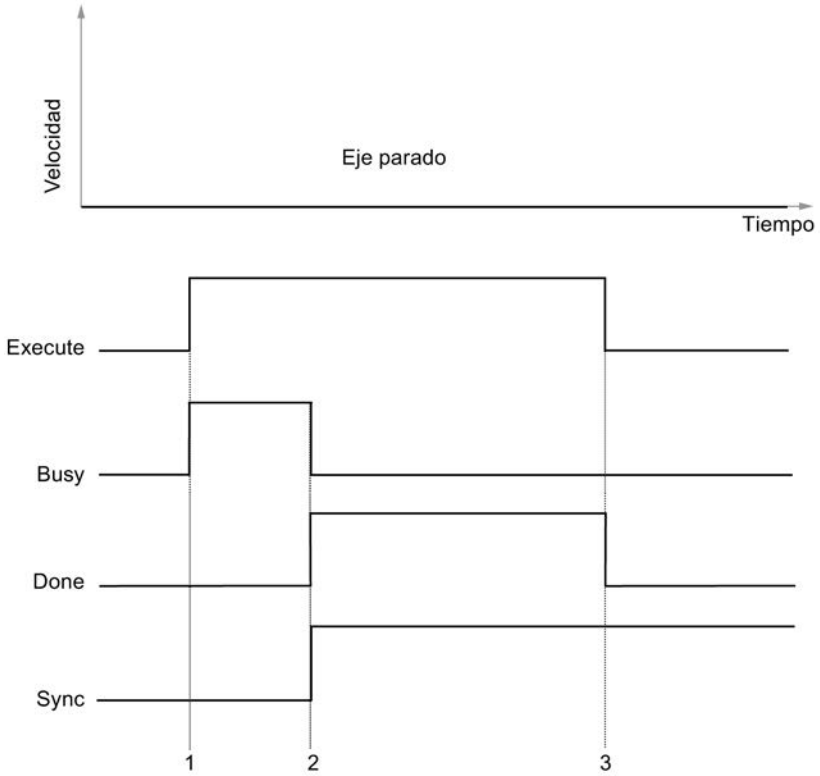


1 Inicio de la búsqueda del punto de referencia con "Execute".
2 Punto de referencia encontrado, el eje frena.
3 El eje se ha detenido, se elimina "Busy" y se activa "Done".

El eje está sincronizado.

4 Al resetear "Execute", se resetea "Done".

Diagrama de evolución de la señal para el posicionamiento del punto de referencia



1 Inicio del posicionamiento del punto de referencia o reajuste del encóder absoluto con "Execute".
2 El eje notifica sincronizado.
3 Al resetear "Execute", se resetea "Done". El proceso finaliza.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

El inicio de la búsqueda del punto de referencia o del posicionamiento del punto de referencia sólo se permite con el eje parado.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
Execute	INPUT	BOOL	Inicio de la búsqueda del punto de referencia o del posicionamiento del punto de referencia con un flanco ascendente.
Position	INPUT	REAL	**Encóder incremental:** Coordenada del punto de referencia o del punto cero de la máquina. **Encóder absoluto:** Coordenada del punto cero de la máquina. El valor debe estar dentro de la zona de trabajo: • **Eje lineal:** Dentro del final de carrera por software • **Eje rotativo:** Inicio del eje ≤ Posición < Fin del eje
Velocity	INPUT	REAL	Velocidad del eje en [unidad de longitud/s] Velocity > 0: búsqueda del punto de referencia Velocity= 0: posicionamiento del punto de referencia
Acceleration	INPUT	REAL	Aceleración del eje en [unidad de longitud/s^2]
Deceleration	INPUT	REAL	Deceleración del eje en [unidad de longitud/s^2]
Direction	INPUT	INT	Sentido inicial para la búsqueda del punto de referencia. 1 = búsqueda del punto de referencia en sentido positivo -1 = búsqueda del punto de referencia en sentido negativo
Busy	OUTPUT	BOOL	1 = petición en curso.
Done	OUTPUT	BOOL	1 = petición finalizada sin errores. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
CommandAborted	OUTPUT	BOOL	1 = la petición ha sido cancelada por otra petición. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
Error	OUTPUT	BOOL	1 = la petición ha sido cancelada por haberse detectado un error de eje. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.

MC_StopMotion

Descripción

La instrucción MC_StopMotion permite cancelar cualquier instrucción de desplazamiento de forma que el eje se detenga con la deceleración ajustada.

Caso de aplicación típico: cancelación de un posicionamiento en un momento cualquiera debido a un evento externo (p. ej., apertura de una puerta de protección).

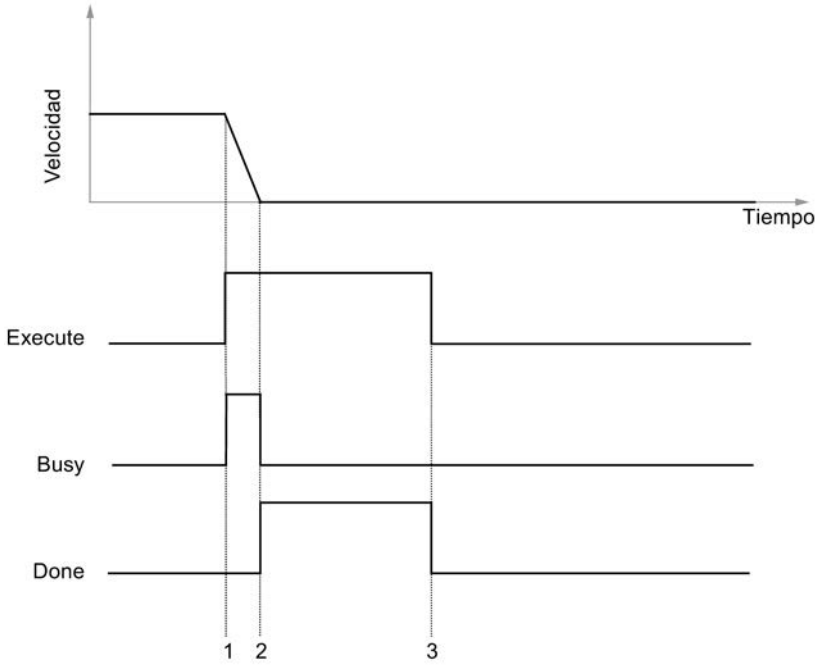
Funcionamiento

Con la detección del flanco ascendente en la entrada "Execute", la instrucción MC_StopMotion asume el control del eje y lo frena hasta detenerse. La instrucción MC_StopMotion sólo puede cancelarse debido a errores graves (Página 11278), pero no debido a una instrucción de desplazamiento. Tras alcanzarse la velocidad cero se puede volver a iniciar un nuevo desplazamiento.

La deceleración en la instrucción MC_StopMotion se puede seleccionar a discreción. Si es inferior a la de la instrucción cancelada, el eje se frena con la deceleración de la instrucción cancelada. De esta forma se garantiza que el eje no se mueva más allá del destino de la instrucción cancelada.

Si se cancela un desplazamiento con destino (MC_MoveAbsolute, MC_MoveRelative) mediante la instrucción MC_StopMotion, éste sigue calculando la distancia residual. Tras finalizar la instrucción MC_StopMotion, la distancia residual es la que existe entre la consigna de posición actual y el destino de la instrucción cancelada. Con este valor se puede dar una instrucción MC_MoveRelative para mover el eje al destino original después de la cancelación.

Diagrama de evolución de la señal de MC_StopMotion



1 Durante la ejecución de un movimiento se inicia la instrucción de parada.

2 Al alcanzarse la velocidad, MC_StopMotion activa la salida "Done".

3 Al resetearse "Execute" finaliza la instrucción de parada.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
Execute	INPUT	BOOL	Inicio del proceso de frenado para un flanco ascendente.
Deceleration	INPUT	REAL	Deceleración del eje para la rampa de frenado en [unidad de longitud/s^2] Si se introducen valores ≤ 0.0, se utilizará el valor de la deceleración máxima del DB de eje.
Busy	OUTPUT	BOOL	1 = petición en curso.
Done	OUTPUT	BOOL	1 = petición finalizada sin errores. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.
CommandAborted	OUTPUT	BOOL	1 = petición cancelada por modo manual en el regulador.
Error	OUTPUT	BOOL	1 = la petición ha sido cancelada por haberse detectado un error de eje. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.

MC_Control

Descripción

La instrucción MC_Control permite realizar la regulación de posición con los drivers de entrada y salida.

La instrucción MC_Control puede adoptar los siguientes estados del regulador:

- Modo de regulación
- Modo de seguimiento
- Modo manual

Modo de regulación

La instrucción MC_Control

- calcula la consigna de velocidad para el driver de salida a partir de la consigna de posición de una instrucción de desplazamiento y el valor real de posición del driver de entrada.
- vigila
  - parada,
  - aproximación a destino,
  - error de seguimiento.
- procesa el acuse de errores (véase Señalizaciones de error y acuse de errores (Página 11273)).

Modo de seguimiento

En este estado se corrige la consigna de posición conforme al valor real de posición. El modo de seguimiento se activa cuando la entrada "EnableDrive" = FALSE o tras un error de eje con parada brusca (véase Señalizaciones de error y confirmación de errores (Página 11273)). En el modo de seguimiento, la salida "DriveEnabled" = FALSE.

Modo manual

En este estado se emite la consigna manual "ManVelocity (Página 11239)" como consigna de velocidad. Ésta se limita a "MaxVelocity (Página 11243)". La consigna de posición se corrige conforme al valor de posición real.

Este estado se activa o desactiva mediante "ManEnable (Página 11239)".

⚠ ADVERTENCIA
El modo manual es posible en todo momento, incluso aunque el eje se encuentre en estado de error.

Funcionamiento

El algoritmo de regulación de la instrucción MC_Control es un regulador P que se parametriza mediante el parámetro "Ganancia del regulador (Página 11238)" ("FactorP").

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Llame la instrucción MC_Control una vez por eje sin condición en el OB seleccionado.

Parámetro

Parámetro	Declaración	Tipo de datos	Significado
EnableDrive	INPUT	BOOL	0 = bloquear accionamiento, el modo de seguimiento está activo. 1 = habilitar accionamiento, la regulación de posición está activa. Si no hay errores pendientes, se ajusta en consecuencia "DriveEnabled".
DriveEnabled	OUTPUT	BOOL	Habilitación de accionamiento, la regulación de posición está activa. Si se ha activado "EnableDrive" y no hay errores pendientes, se activa "DriveEnabled". Si no se ha activado "DriveEnabled", el eje se encuentra en el modo de seguimiento y no se vigila la parada del eje. Esta señal debe utilizarse para controlar la entrada de habilitación en la etapa de potencia.

MC_Simulation

Descripción

La instrucción MC_Simulation permite probar el programa de desplazamiento, incluidos los bloques driver. No se necesita periferia y el eje no se desplaza.

Funcionamiento

La instrucción MC_Simulation reacopla el valor de salida del regulador al driver de entrada, simulando un valor de encóder conforme al tipo de encóder seleccionado en el DB de eje.

El modo de simulación se activa activando en el DB de eje el bit "Sim" = TRUE.

Con la simulación activada no es posible la búsqueda del punto de referencia. Parametrice durante la simulación la entrada "Velocity" de la instrucción MC_Home con 0.0. De esta forma se ejecuta un posicionamiento del punto de referencia en lugar de una búsqueda del punto de referencia.

Si durante el modo de simulación se ejecuta un posicionamiento del punto de referencia, también se modificará el punto de referencia del eje real. Por tanto, tras desactivar el modo de simulación es preciso volver a sincronizar el eje.

Al activar y desactivar la simulación, el eje se lleva al estado de parada de acuse obligatorio (véase Error con parada brusca (Página 11278)) y además se activan todos los bits del campo de bits "Init.Ix" en el DB de eje.

Si el modo de simulación está activado, ocurre lo siguiente:

- El driver de salida no emite ningún valor a la periferia.
- El driver de entrada no lee ningún valor de encóder de la periferia.

⚠ PELIGRO
Al activar el modo de simulación, el eje debe encontrarse parado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

MC_GearIn

Descripción

Con la instrucción MC_GearIn se puede acoplar la consigna de velocidad de un eje esclavo a la consigna de velocidad de un eje maestro (acoplamiento de consigna) mediante un factor de reducción.

Funcionamiento

Al detectarse un flanco ascendente en la entrada "Execute", la instrucción MC_GearIn acopla un eje esclavo (parámetro "Slave") a la consigna de velocidad y posición de un eje maestro (parámetro "Master") mediante el factor de reducción ajustado. Por tanto, se trata de una instrucción de desplazamiento del eje esclavo.

Durante un desplazamiento acoplado en cada ciclo:

- el trayecto recorrido por el eje esclavo desde la posición de acoplamiento = el trayecto recorrido por el eje maestro desde la posición de acoplamiento × factor de reducción
- la velocidad del eje esclavo = velocidad de salida del reductor = velocidad del eje maestro × factor de reducción
- la aceleración del eje esclavo = la aceleración del eje maestro × factor de reducción
- la deceleración del eje esclavo = la deceleración del eje maestro × factor de reducción

Para que el eje esclavo pueda seguir al eje maestro sin limitaciones, ajuste los siguientes valores con un factor de reserva de 1,1 para el eje esclavo. Estos valores deben estar permitidos y ser factibles:

- Velocity(eje esclavo) = Velocity (eje esclavo) × ABS (factor de reducción) × 1,1
- Acceleration(eje esclavo) = Acceleration (eje esclavo) × ABS (factor de reducción) × 1,1
- Deceleration(eje esclavo) = Deceleration (eje esclavo) × ABS (factor de reducción) × 1,1

Para realizar el desacoplamiento, active la instrucción de parada o bien cualquier instrucción de desplazamiento para el eje esclavo.

Al igual que las demás instrucciones de desplazamiento, la instrucción MC_GearIn puede cancelar un desplazamiento y puede cancelarse automáticamente.

El eje esclavo puede acoplarse a un eje maestro que ya se esté desplazando, y el eje esclavo acelera o frena automáticamente con las rampas parametrizadas en la instrucción de reducción hasta que se haya alcanzado la velocidad de salida del reductor.

Cuando se alcanza por primera vez la velocidad de salida del reductor tras el inicio, se activa la salida "InGear". Si en este momento no está activado "Execute", "InGear" estará activado durante una llamada a la instrucción. "InGear" se resetea cuando Execute se desactiva, en caso de error y cuando se cancela el desplazamiento.

Mientras la velocidad de salida del reductor sea alcanzable, el eje esclavo se desplaza tanto en sincronismo de velocidad como de posición. La salida "Coupled" de la instrucción está activada.

Si el eje esclavo no puede alcanzar la velocidad de salida del reductor, se desplazará con los parámetros de desplazamiento indicados en la instrucción de reducción. En tal caso, la salida "Coupled" de la instrucción estará reseteada. En cuanto la velocidad de salida del reductor vuelve a ser alcanzable, el eje esclavo se desplazará a dicha velocidad. La salida "Coupled" de la instrucción se activa.

Mientras esté activa la instrucción MC_GearIn para el eje esclavo, es posible activar cualesquiera instrucciones de desplazamiento para el eje maestro, incluso con cancelación de un desplazamiento.
Excepción: Al referenciar el eje maestro se produce un desacoplamiento con un error ("Err.MasterErr").

El parámetro de eje "Override" no tiene sentido para el eje esclavo y por tanto no se evalúa.

Particularidades en un eje esclavo como eje lineal:

- Si se rebasa el máximo recorrido durante un desplazamiento de reducción, la instrucción MC_GearIn frena al eje esclavo con el mensaje de error "Err.DistanceErr" hasta parada (véase Desarrollo de movimientos de desplazamiento (Página 11258)). El acoplamiento se deshace.
- Un eje lineal **no sincronizado** no se vigila y por tanto puede desplazarse hasta su fin físico sin frenar. Por tanto, es preciso terminar el desplazamiento a tiempo.
- Un eje lineal **sincronizado** se frena al final de la zona de trabajo y se detiene en el final de carrera por software. Se emite el mensaje de error "DistanceErr", así como "Err.SWLimitMinExceeded" o "Err.SWLimitMaxExceeded". El acoplamiento se deshace.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

La instrucción MC_GearIn permite el desplazamiento hasta el fin físico del eje con un eje lineal no sincronizado.

Para evitar daños personales y materiales es necesario tomar las medidas siguientes:

- Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Sólo así puede garantizarse que en caso de un fallo del software o del equipo sea posible desconectar la instalación de modo seguro.
- Instale finales de carrera de seguridad que actúen directamente en las etapas de potencia de todos los accionamientos.
- Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.

Particularidades en un eje esclavo como eje rotativo:

- El recorrido de desplazamiento de la instrucción MC_GearIn no está limitado (giro continuo).

Tratamiento de errores

Error en el eje maestro

- Si el eje maestro detecta errores que dan lugar a una parada suave, se frena. El eje esclavo realiza este movimiento acoplado, y no se desacopla. Si se confirma el error del eje maestro y éste sigue moviéndose, el eje esclavo también vuelve a arrancar.
- Si el eje maestro detecta errores que dan lugar a una parada brusca, o si el eje maestro se encuentra en el modo manual, la instrucción de reducción frena al eje esclavo y lo desacopla. Emite el mensaje de error "Err.MasterErr".

Error en el eje esclavo

- Si el eje esclavo detecta errores que dan lugar a una parada suave, la instrucción de reducción frena el eje esclavo y lo desacopla.
- Si el eje esclavo detecta errores que dan lugar a una parada brusca, el eje esclavo se detiene y se desacopla.
- El recorrido máximo de un desplazamiento de acoplamiento en un eje lineal se limita a 2^{24} incrementos o a 2^{24} [unidad de longitud]. Si se rebasa este límite, la instrucción MC_GearIn frena al eje esclavo, emite el mensaje de error acusable "DistanceErr" y lo desacopla.

Error en el acoplamiento

- Si el eje esclavo no puede alcanzar la velocidad de salida del reductor en el ciclo actual, se desplazará con los parámetros de desplazamiento indicados en la instrucción. Esto es válido en particular para el acoplamiento a un eje maestro en movimiento, así como al frenar en caso de error.
  El parámetro "Coupled" se resetea, pero el acoplamiento no se deshace. En cuanto vuelven a ser alcanzables los valores solicitados, el eje esclavo se desplaza según dichos valores, y se activa "Coupled".

Interacción de las instrucciones en el reductor

Al utilizar la instrucción de reducción también debe tenerse en cuenta lo siguiente (véase también "Estructura de un programa de usuario (Página 11252)"):

- El eje maestro y el eje esclavo deben llamarse en **el mismo** nivel de ejecución.
- El eje esclavo vuelve a orientarse en cada ciclo de mecanizado a las especificaciones del eje maestro. Por tanto, las instrucciones del eje esclavo deben llamarse según las instrucciones del eje maestro. En particular, la instrucción de reducción del eje esclavo debe llamarse **después** del regulador del eje maestro.
  Otras secuencias de llamada, en particular la llamada anidada de las instrucciones del eje maestro y el eje esclavo, pueden generar grandes errores de seguimiento entre los ejes.
- En una aplicación isócrona, los datos de periferia del eje maestro **y** del eje esclavo deben encontrarse en la misma memoria imagen parcial de proceso.

Dependiendo de la estructura seleccionada resulta por tanto una de las siguientes imágenes:

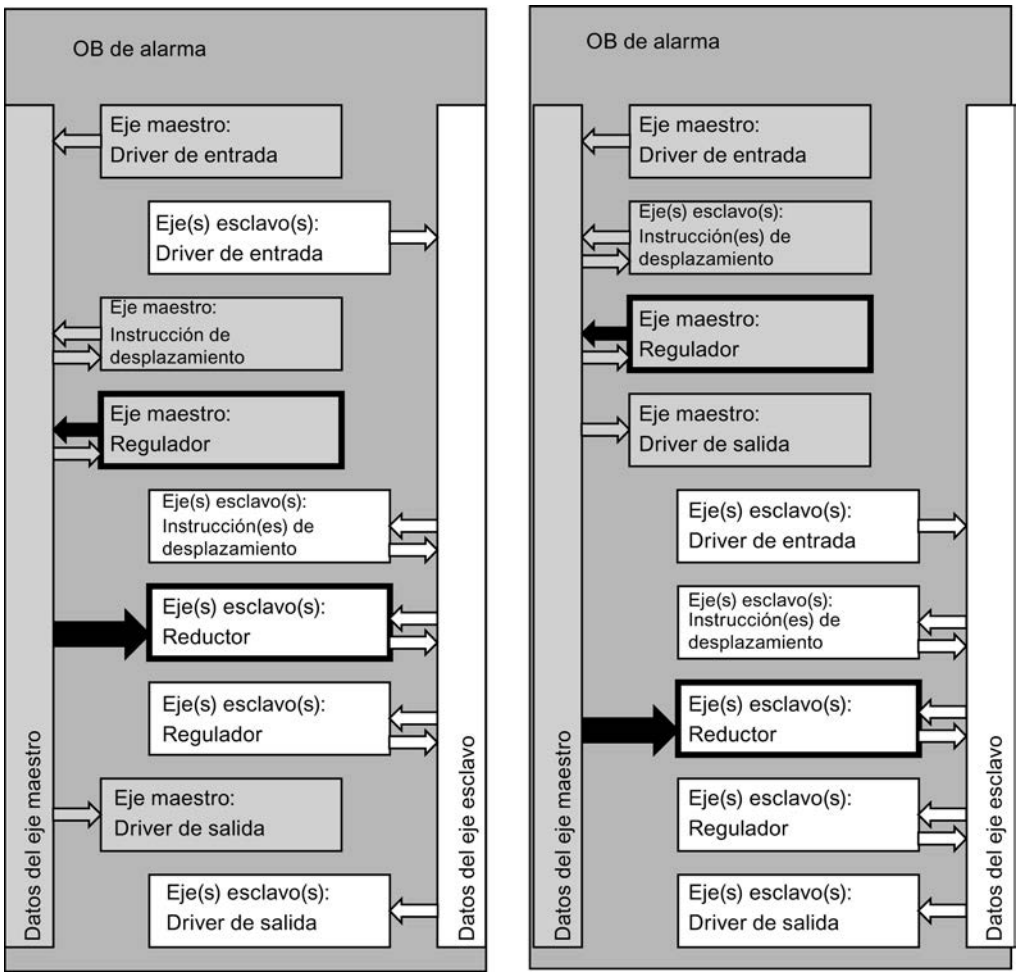
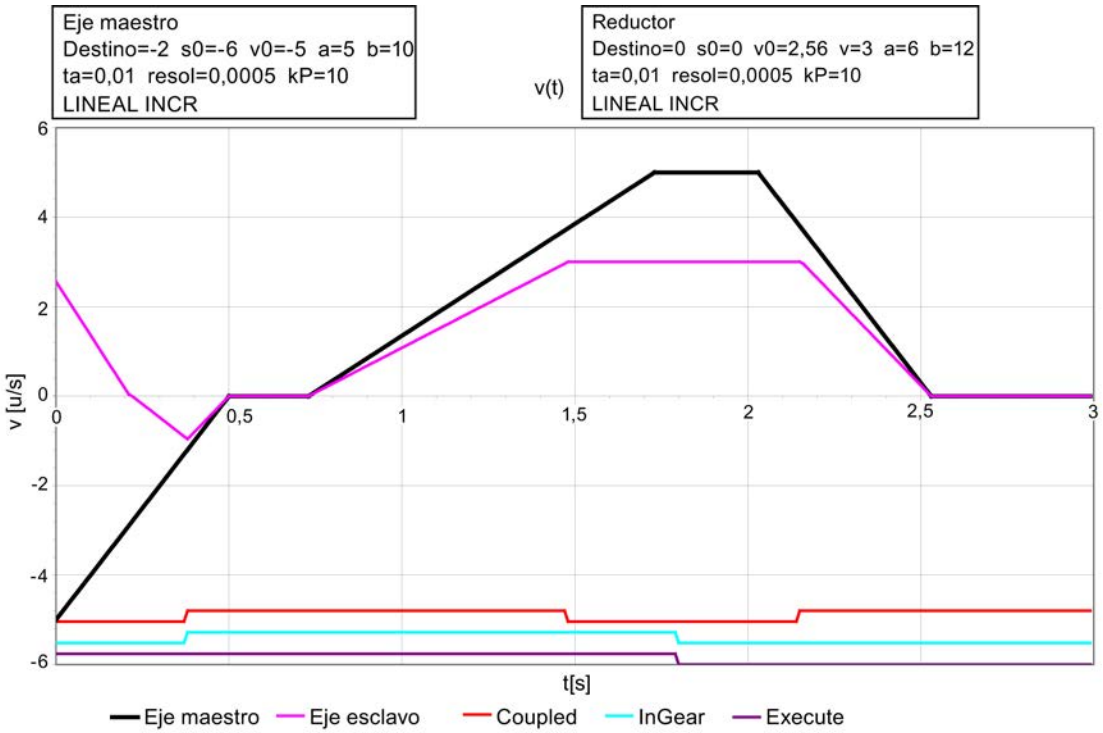


Diagrama de evolución de la señal

La siguiente imagen muestra cómo la instrucción de reducción asume el desplazamiento de un eje, y cómo se acopla a un eje maestro en movimiento. En este ejemplo, el factor de reducción es 0,8. El eje esclavo no puede alcanzar la velocidad necesaria para mantener este factor.

El eje maestro se encuentra parado entre t = 0,5 y t = 0,72. El acoplamiento se mantiene.



Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
Execute	INPUT	BOOL	Inicio del desplazamiento de reducción con un flanco positivo.
Rationumerator	INPUT	REAL	Numerador del factor de reducción Factor de reducción = "RatioNumerator"/"RatioDenominator" Si "RatioNumerator" < 0, el sentido de movimiento del eje maestro y del eje esclavo son opuestos. "RatioNumerator" = 0 se rechaza con "DataErr" = 1. Es posible modificar el factor de reducción efectivo para un eje esclavo cancelando la instrucción de reducción con otra instancia que tenga factor de reducción modificado.
RatioDenominator	INPUT	INT	Denominador del factor de reducción; debe ser un entero ≥ 1 Es posible modificar el factor de reducción efectivo para un eje esclavo cancelando la instrucción de reducción con otra instancia que tenga factor de reducción modificado.

Parámetro	Declaración	Tipo de datos	Descripción
Velocity	INPUT	REAL	Velocidad del eje en [unidad de longitud/s] La velocidad que puede alcanzar el eje esclavo en un desplazamiento de reducción queda limitada a este valor.
Acceleration	INPUT	REAL	Aceleración del eje en [unidad de longitud/s^2] La aceleración que puede alcanzar el eje esclavo en un desplazamiento de reducción queda limitada a este valor. Este parámetro se utiliza cuando • el eje esclavo se acopla a un eje maestro que se mueve más rápido, • el eje maestro se acelera más de lo que permite el eje esclavo.
Deceleration	INPUT	REAL	Deceleración del eje en [unidad de longitud/s^2] La deceleración que puede alcanzar el eje esclavo en un desplazamiento de reducción queda limitada a este valor. Este parámetro se utiliza cuando • un eje esclavo que ya está en movimiento se acopla a un eje maestro más lento, • el eje esclavo frena debido a un error, • el eje maestro se frena más de lo que permite el eje esclavo.
Master	IN-OUT	AXIS_REF	Remisión a los datos del eje maestro
Slave	IN-OUT	AXIS_REF	Remisión a los datos del eje esclavo
Init	IN-OUT	BOOL	La instrucción ejecuta su inicialización con Init = TRUE y a continuación resetea el bit.
Busy	OUTPUT	BOOL	1 = petición en curso.
InGear	OUTPUT	BOOL	1 = velocidad de salida del reductor alcanzada Se activa cuando la velocidad de salida del reductor se alcanza por primera vez. Si en este momento ya se ha reseteado Execute, InGear estará pendiente de todas formas durante una llamada de la instrucción. Se resetea cuando Execute se desactiva, en caso de error y al cancelarse el desplazamiento.
Coupled	OUTPUT	BOOL	1 = el eje está acoplado al eje maestro en este momento Indica en cada ciclo de mecanizado si se ha alcanzado la velocidad de salida del reductor.
CommandAborted	OUTPUT	BOOL	1 = la petición ha sido cancelada por otra petición.
Error	OUTPUT	BOOL	1 = la petición ha sido cancelada por haberse detectado un error de eje. El bit permanece activado hasta que se resetea "Execute", pero como mínimo durante una llamada de la instrucción.

Driver de entrada

Función de los drivers de entrada

Tarea

El driver de entrada sirve para detectar las posiciones reales de un eje. Lee la posición no normalizada del eje y la convierte a la unidad de longitud elegida.

Para una serie de módulos de lectura de recorrido se dispone de drivers de entrada adecuados, los cuales se describen a continuación. Mediante la instrucción EncoderUniversal (Página 7371) es posible la adaptación a configuraciones no soportadas directamente.

Inicialización

- En el caso de un **encóder incremental** se resetea la sincronización (Página 11224) con el eje (parámetro "Sync" = FALSE en los datos de eje). A continuación es necesario volver a buscar el punto de referencia o a posicionar el punto cero de la máquina.
- En el caso de un **encóder absoluto** se mantiene la sincronización con el eje.

Llamada

El driver de entrada de un eje debe llamarse sin condición.

Modo isócrono

Los drivers de entrada soportan el modo isócrono de los módulos de periferia adecuados. El driver detecta el modo isócrono automáticamente porque es llamado en un OB isócrono.

Tratamiento de errores

Existen los siguientes mecanismos de detección de errores:

- Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) puede notificarse al driver de entrada en la entrada "EncErr". Mientras la entrada "EncErr" sea igual a TRUE, el driver de entrada no accederá al módulo de lectura de recorrido.
- Algunos módulos de lectura de recorrido proporcionan información de errores que el driver de entrada puede evaluar por sí mismo.
- Otros módulos de lectura de recorrido proporcionan información de errores solamente mediante alarma de diagnóstico y registro 1 (DS1). Por tanto, es necesario:
  - habilitar la alarma de diagnóstico para el módulo en la configuración hardware, y
  - comparar en el OB 82 la dirección proporcionada en el dato local "OB82_MDL_ADDR" con la dirección del módulo. Si coinciden, ajuste el parámetro "ReadDiagErr" del driver de entrada a TRUE (tanto al aparecer como al desaparecer la alarma).
    Mediante "ReadDiagErr" = TRUE se permite que el driver de entrada lea el registro 1 del módulo con la instrucción RD_REC y lo guarde en el parámetro estático "DiagStatus".
    A continuación, ajusta "ReadDiagErr" = FALSE. El valor de retorno de la instrucción RD_REC se guarda en el parámetro estático "JobErr"

Reacción a errores

- En caso de error se activan las indicaciones "Error" y "Err.EncoderErr" en el DB de eje.
- En un **encóder incremental**, siempre se resetea el bit de estado "Sync" en el DB de eje.
- En un **encóder absoluto**, nunca se resetea el bit de estado "Sync" en el DB de eje.

Preste atención también al tratamiento específico de errores en los siguientes capítulos sobre los diferentes drivers.

Encontrará información detallada sobre la reacción a errores, la señalización de errores y su acuse en "Señalizaciones de error y acuse de errores en el DB de eje (Página 11273)".

Modo de simulación

Mientras esté activado el bit Sim (Página 11230) en el DB de eje, el driver de entrada no accederá al módulo de lectura de recorrido. Los valores del encóder se simulan mediante la instrucción MC_Simulation (Página 7341).

EncoderIM178

Descripción

La instrucción EncoderIM178 está contenida en STEP7 V5.x por razones de compatibilidad. El módulo IM 178-4 ya no se comercializa activamente. Como alternativa, se puede utilizar el IM 174 .

El IM 178-4 debe utilizarse con el mismo número de canal y el mismo eje en el driver de entrada **y** en el driver de salida, pues hay determinados ajustes de direcciones que se refieren a ambos drivers. No es posible mezclar con otros drivers.

Parametrización del IM 178-4 (configuración hardware)

Seleccione el IM 178-4 en el catálogo de hardware, en "Periferia descentralizada". Configure el slot 4 del IM 178-4 como ""4Word AA/12Word AE/Cons. 1Word".

Parámetro	Ajuste
Sincronización	Software
Evaluación de flancos	Ascendente, descendente o ambos flancos.

Los demás ajustes se parametrizan según el encóder conectado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
RefMode	INPUT	BOOL	Tipo de referenciación: 0 = con entrada digital I0 1 = con entrada digital I1 y marca cero
DI_0	OUTPUT	BOOL	Entrada digital 0 del correspondiente canal del IM 178-4
DI_1	OUTPUT	BOOL	Entrada digital 1 del correspondiente canal del IM 178-4
DI_2	OUTPUT	BOOL	Entrada digital 2 del correspondiente canal del IM 178-4
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
Status.PARA	STAT	BOOL	PARA = TRUE: IM 178-4 se ha parametrizado correctamente en la configuración hardware. PARA = FALSE: véase Manual SIMATIC IM 178-4 Drive Interface (http://support.automation.siemens.com/WW/view/en/1117389)
Status.EXTF0	STAT	BOOL	1 = fallo del cable de señal del encóder
Status.EXTF1	STAT	BOOL	1 = error de encóder
Status.EXTF2	STAT	BOOL	1 = error de marca cero

Sincronización

La sincronización se realiza alternativamente:

- Mediante la búsqueda del punto de referencia (sólo con encóder incremental)
- Mediante el posicionamiento del punto de referencia (con encóder incremental y absoluto)

El sensor del punto de referencia se conecta a la entrada digital I0 o I1 dependiendo del ajuste.

Para una búsqueda del punto de referencia, IM 178-4 puede detectar el punto de referencia a elección en los siguientes casos:

- Con flanco descendente o ascendente o con ambos flancos en la entrada digital I0 (RefMode = 0).
- Con el flanco ascendente de la primera marca cero tras detectarse el flanco descendente en la entrada digital I1 (RefMode = 1).

Si el eje ya se encuentra en el sensor del punto de referencia al inicio de la búsqueda del punto de referencia, se efectuará una referenciación en la posición actual, al igual que en el posicionamiento del punto de referencia.

Entradas digitales

Las entradas digitales I0 a I2 del canal seleccionado del IM 178-4 se indican en los parámetros de salida "DI_0" a "DI_2" del driver de entrada.

Tratamiento de errores

Como reacción a un error, el driver de entrada ajusta al detectar el error los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje. El driver de entrada desactiva el bit de estado "Sync" en el DB de eje, si es que se utiliza un encóder incremental. Para más información, consulte el Manual SIMATIC IM 178-4 Drive Interface (http://support.automation.siemens.com/WW/view/en/1117389).

Encontrará información detallada sobre la reacción a errores, la señalización de errores y su acuse en "Señalizaciones de error y acuse de errores en el DB de eje (Página 11273)".

Acuse de errores

Si activa el acuse de errores en el DB de eje de Easy Motion Control, se acusarán también los errores pendientes en el módulo.

EncoderFM450

Descripción

La instrucción EncoderFM450 sirve para utilizar un canal del módulo FM 450-1 como módulo de lectura de recorrido para encóders incrementales. El número de canal se parametriza mediante el DB de eje.

Parametrización del módulo (configuración hardware)

El módulo FM 450-1 debe parametrizarse con los ajustes de acuerdo a la siguiente tabla.

Diálogo	Ajuste
Selección de alarmas	Habilitar alarma de diagnóstico
Modos de operación	• Rango de contaje: -31 a +31 bits • Modo de operación: contaje sin fin • Control de puerta: ninguno
Entradas	Si desea realizar una búsqueda del punto de referencia, debe elegir el ajuste siguiente: • Sincronización: una vez • El ajuste "Evaluar marca cero para el posicionamiento" se puede activar o desactivar como criterio para el punto de referencia.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
ReadDiagErr	IN-OUT	BOOL	Active este bit en el OB 82 cuando el módulo haya disparado una alarma de diagnóstico (véase más abajo, en Tratamiento de errores).
DI_0	OUTPUT	BOOL	Entrada digital 0 del canal seleccionado
DI_1	OUTPUT	BOOL	Entrada digital 1 del canal seleccionado
DI_2	OUTPUT	BOOL	Entrada digital 2 del canal seleccionado
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
JobErr	STAT	INT	Valor de retorno de la instrucción RD_REC si se ha activado "ReadDiagErr" = TRUE.
DiagStatus.EXT_VOLTAGE	STAT	STRUCT	Fallo en la tensión auxiliar externa
DiagStatus.CONFIG_ERR	STAT	STRUCT	Error de parametrización
DiagStatus.WTCH_DOG_FLT	STAT	STRUCT	La vigilancia de tiempo (watchdog) ha saltado
DiagStatus.RAM_FLT	STAT	STRUCT	Defecto en RAM
DiagStatus.CH1_SIGA	STAT	STRUCT	Canal 1 Señal A errónea
DiagStatus.CH1_SIGB	STAT	STRUCT	Canal 1 Señal B errónea
DiagStatus.CH1_SIGZ	STAT	STRUCT	Canal 1 Señal N errónea
DiagStatus.CH1_BETW	STAT	STRUCT	Canal 1 Error entre canales
DiagStatus.CH1_5V2	STAT	STRUCT	Canal 1 Alimentación del encóder 5,2 V errónea
DiagStatus.CH2_SIGA	STAT	STRUCT	Canal 2 Señal A errónea
DiagStatus.CH2_SIGB	STAT	STRUCT	Canal 2 Señal B errónea
DiagStatus.CH2_SIGZ	STAT	STRUCT	Canal 2 Señal N errónea
DiagStatus.CH2_BETW	STAT	STRUCT	Canal 2 Error entre canales
DiagStatus.CH2_5V2	STAT	STRUCT	Canal 2 Alimentación del encóder 5,2 V errónea

Sincronización

La sincronización se puede realizar

- mediante búsqueda del punto de referencia
- mediante posicionamiento del punto cero de la máquina.

Conecte el sensor del punto de referencia a la entrada digital I2.

En una búsqueda del punto de referencia, el módulo puede detectar el punto de referencia a elección en los siguientes casos:

- con un flanco ascendente en la entrada digital I2, o bien
- con el flanco ascendente de la marca cero, si la entrada digital I2 está activada.

Si el eje ya se encuentra en el sensor del punto de referencia al inicio de la búsqueda del punto de referencia, se efectuará una referenciación en la posición actual, al igual que en el posicionamiento del punto de referencia.

Entradas digitales

Las entradas digitales I0 a I2 del canal seleccionado se indican en los parámetros de salida "DI_0" a "DI_2" del driver de entrada.

Tratamiento de errores

El módulo proporciona información de errores a través del registro 1 (DS1). En los siguientes casos, el driver de entrada lee el registro 1 del módulo mediante la instrucción RD_REC y lo almacena en el parámetro estático "DiagStatus":

- El driver de entrada se inicializa.
- El módulo notifica que ha detectado un error.
- El parámetro de entrada "ReadDiagErr" del driver es TRUE.

A continuación, ajusta "ReadDiagErr" = FALSE. El valor de retorno de la instrucción RD_REC se guarda en el parámetro estático "JobErr" (véase Avisos JOB_Err (Página 11281)).

Si se ha registrado una señalización de errores en el registro 1, el driver de entrada ajusta en el DB de eje los identificadores de error "Error" y "Err.EncoderErr" a TRUE. El bit de estado "Sync" se ajusta a FALSE.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

Nota

Si utiliza un FM 450-1 con un solo canal, debe parametrizar el segundo canal con el tipo de encóder "24 V pista A+B con desplazamiento de fase", pues de lo contrario éste enviará cíclicamente alarmas de diagnóstico (p. ej., debido a una conexión errónea).

EncoderET200S1SSI

Descripción

La instrucción EncoderET200S1SSI sirve para utilizar el módulo ET 200S 1SSI Fast Mode como módulo de lectura de recorrido para encóders absolutos.

El driver también se puede utilizar para la variante isócrona de este módulo.

Parametrización del módulo ET 200S 1SSI Fast Mode (configuración hardware)

El driver de entrada solamente es adecuado para el módulo 1SSI con el ajuste de parámetro "**Fast Mode**" como modo de funcionamiento. En el TIA Portal, el "**Fast Mode**" ya no se selecciona a través del módulo, sino que se ajusta como modo de funcionamiento.

Seleccione el módulo en el catálogo hardware bajo "Periferia descentralizada > ET 200S > FM > Lectura de recorrido".

Parámetro	Ajuste
Diagnóstico colectivo	Se puede activar si desea realizar una evaluación en su programa.
Detección	Libre
Inversión del sentido de giro	Desactivada El sentido de giro puede invertirse mediante las instrucciones de Easy Motion Control (parámetro "Corrección de sentido del encóder (Página 11238)").
Modo de funcionamiento	Fast Mode

Los demás ajustes se parametrizan según el encóder conectado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
Status.STS_UP	STAT	BOOL	Sentido hacia delante: los valores del encóder aumentan
Status.STS_DN	STAT	BOOL	Sentido hacia atrás: los valores del encóder disminuyen
Status.STS_DI	STAT	BOOL	Estado de la entrada digital
Status.EXTF	STAT	BOOL	Error agrupado o cortocircuito de la alimentación del encóder
Status.ERR_PARA	STAT	BOOL	Error de parametrización
Status.RDY	STAT	BOOL	El módulo está listo para el funcionamiento

Sincronización

El módulo ET 200S 1SSI Fast Mode permite la sincronización solamente mediante posicionamiento del punto cero de la máquina.

Tratamiento de errores

Si el driver de entrada detecta una señalización de error del módulo, activa los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

EncoderAbsSensorDP

Descripción

La instrucción EncoderAbsSensorDP sirve para utilizar el encóder absoluto SIMODRIVE sensor con PROFIBUS DP para la lectura de posición.

Parametrización del encóder absoluto SIMODRIVE sensor (configuración hardware)

Seleccione en el catálogo de hardware el encóder "PROFIBUS DP > ENCÓDER > SIMODRIVE sensor > Versión 2.2 Singleturn o Versión 2.2 Multiturn". Si el SIMODRIVE sensor aún no está disponible, intégrelo sirviéndose del archivo GSD correspondiente. A continuación encontrará el encóder en la carpeta "Otros dispositivos de campo > PROFIBUS DP > Encóder > Siemens AG > SIMODRIVE sensor".

Parámetro	Ajuste	
Sentido de contaje	Ascendente en sentido horario El sentido de giro puede invertirse mediante las instrucciones de Easy Motion Control (parámetro "Corrección de sentido del encóder (Página 11238)").	
	Multiturn	Singleturn
Función de escalado	activada	activada
Incrementos de medida deseados (high)	0	-
Incrementos de medida deseados (low)	4096	4096
Incrementos de medida físicos (high)	0	-
Incrementos de medida físicos (low)	4096	4096
Resolución deseada por	Vuelta	Vuelta
Resolución total (high)	256	-
Resolución total (low)	0	4096
Modo de puesta en marcha	Desactivado	Desactivado
Diagnóstico más corto	No	No
Final de carrera inferior	Desactivado	Desactivado
Final de carrera inferior (high)	0	-
Final de carrera inferior (low)	0	0
Final de carrera superior	Desactivado	Desactivado
Final de carrera superior (high)	0	-
Final de carrera superior (low)	32767	4096
Salida de velocidad	Incrementos/1000 ms	Incrementos/1000 ms

La adaptación a la unidad de longitud elegida se realiza con las instrucciones de Easy Motion Control. Por tanto, el encóder se puede parametrizar de forma fija (como se indica más arriba). Un "Teach-In" del encóder no es necesario (véase Instrucciones de uso del encóder absoluto SIMODRIVE sensor con PROFIBUS DP (http://support.automation.siemens.com/WW/view/en/28813901)).

En el parámetro "Incrementos por vuelta del encóder" de Easy Motion Control, introduzca 4096 ("StepsPerRev (Página 11235)).

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
Status.Ready	STAT	BOOL	1 = el encóder está listo para el funcionamiento
Status.Mode	STAT	BOOL	Modo de operación: 0 = modo de puesta en marcha, 1 = modo normal
Status.SWLimitExceeded	STAT	BOOL	1 = final de carrera por software MIN o MAX ha reaccionado
Status.Dir	STAT	BOOL	Sentido de giro: 0 = sentido horario, 1 = sentido antihorario

Sincronización

Con el encóder absoluto SIMODRIVE sensor es posible una sincronización solamente mediante posicionamiento del punto de referencia.

Tratamiento de errores

Si el driver de entrada detecta que hay pendiente un mensaje de error del encóder, activa los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje. Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

EncoderSM338

Descripción

La instrucción EncoderSM338 sirve para utilizar el SM 338 POS_INPUT como módulo de lectura de recorrido para encóders absolutos.

El driver también se puede utilizar para la variante isócrona de este módulo.

Parametrización del SM 338 (configuración hardware)

La función FREEZE debe estar desactivada.

Si el driver de entrada detecta el estado FREEZE de la entrada del encóder, activa los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje.

Los demás ajustes se parametrizan según el encóder conectado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
ReadDiagErr	IN-OUT	BOOL	Active este bit en el OB 82 cuando el módulo haya disparado una alarma de diagnóstico (véase Tratamiento de errores).
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
Freeze	STAT	BOOL	Estado FREEZE de la entrada de encóder
JobErr	STAT	INT	Valor de retorno de la instrucción RD_REC si se ha activado "ReadDiagErr" = TRUE.
DiagStatus.EXT_VOLTAGE	STAT	STRUCT	Fallo en la tensión auxiliar externa
DiagStatus.CONFIG_ERR	STAT	STRUCT	Error de parametrización
DiagStatus.WTCH_DOG_FLT	STAT	STRUCT	La vigilancia de tiempo (watchdog) ha saltado
DiagStatus.RAM_FLT	STAT	STRUCT	Defecto en RAM
DiagStatus.CH0_INTF	STAT	STRUCT	Error de configuración o de parametrización (error de canal interno)
DiagStatus.CH0_EXTF	STAT	STRUCT	Error de encóder (error de canal externo)

Sincronización

Con el SM 338 la sincronización solamente es posible mediante posicionamiento del punto de referencia.

Tratamiento de errores

El módulo proporciona información de errores solamente mediante la alarma de diagnóstico y el registro 1 (DS1). Por tanto, es necesario:

- habilitar la alarma de diagnóstico para el módulo en la configuración hardware, y
- comparar en el OB 82 la dirección proporcionada en el dato local "OB82_MDL_ADDR" con la dirección del módulo. Si coinciden, ajuste el parámetro ReadDiagErr del driver de entrada a TRUE (tanto al aparecer como al desaparecer la alarma).
  Con "ReadDiagErr" = TRUE el driver de entrada lee el registro 1 del módulo mediante la instrucción RD_REC y lo guarda en el parámetro estático "DiagStatus".
  A continuación, ajusta "ReadDiagErr" = FALSE. El valor de retorno de la instrucción RD_REC se guarda en el parámetro estático "JobErr" (véase Avisos JOB_Err (Página 11281)).

Si se ha registrado una señalización de errores en el registro 1, el driver de entrada ajusta en el DB de eje los identificadores de error "Error" y "Err.EncoderErr" a TRUE.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

EncoderET200S1Count

Descripción

La instrucción EncoderET200S1Count sirve para utilizar el módulo ET 200S 1Count5V o 1Count24V como módulo de lectura de recorrido para encóders incrementales.

El driver también se puede utilizar para la variante isócrona de este módulo.

Parametrización del módulo ET 200S 1Count (configuración hardware)

El driver de entrada solamente es adecuado para los módulos "1 Count 5V" o "1 Count 24V" del catálogo de hardware. El modo de contaje ya no se selecciona mediante el módulo, sino como modo de funcionamiento.

Seleccione el módulo desde el catálogo de hardware bajo "Periferia descentralizada > ET 200S > FM > Contaje".

Parámetro	Ajuste
Diagnóstico colectivo	Se puede activar si desea realizar una evaluación en su programa.
Modo de contaje	Contaje sin fin

Parámetro	Ajuste
Función DI, Sincronización	• Si desea realizar una búsqueda del punto de referencia, seleccione – Función DI = sincronización con flanco ascendente y – Sincronización = una vez. • De lo contrario, seleccione: – Función DI = entrada
Función de puerta	Cualquiera
Modo de funcionamiento	Contaje

Los demás ajustes se parametrizan según el encóder conectado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
DOut_1	INPUT	BOOL	Salida digital 1
DOut_2	INPUT	BOOL	Salida digital 2 (sólo existe en 1Count5V)
DIn	OUTPUT	BOOL	Entrada digital del módulo
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
Status.ERR_ENCODER	STAT	BOOL	Cortocircuito/rotura de hilo señal de encóder (sólo en 1Count5V)
Status.ERR_DO2		BOOL	Cortocircuito/rotura de hilo/sobretemperatura DO2 (sólo en 1Count5V)
Status.ERR_PARA		BOOL	Error de parametrización
Status.ERR_DO1		BOOL	Cortocircuito/rotura de hilo/sobretemperatura DO1
Status.ERR_24V		BOOL	Cortocircuito en la alimentación del encóder

Sincronización

La sincronización se realiza alternativamente:

- Mediante búsqueda del punto de referencia
- mediante posicionamiento del punto de referencia

Conecte el sensor del punto de referencia a la entrada digital DI.

En una búsqueda del punto de referencia, el módulo ET 200S 1Count puede detectar el punto de referencia con un flanco ascendente en la entrada digital DI.

Si el eje ya se encuentra en el sensor del punto de referencia al inicio de la búsqueda del punto de referencia, se efectuará una referenciación en la posición actual, al igual que en el posicionamiento del punto de referencia.

Entrada digital

La entrada digital del módulo se indica en el parámetro de salida "DIn" del driver de entrada.

Salidas digitales

Las salidas digitales se pueden controlar mediante los parámetros de entrada "DOut_1" y "DOut_2" del driver de entrada.

Tratamiento de errores

Si el driver de entrada detecta una señalización de error del módulo, activa los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje y desactiva el bit de estado "Sync" en el DB de eje.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

EncoderFM350

Descripción

La instrucción EncoderFM350 sirve para utilizar el módulo FM 350-1 como módulo de lectura de recorrido para encóders incrementales.

El driver también se puede utilizar para la variante isócrona de este módulo.

Parametrización del módulo (configuración hardware)

Diálogo	Ajuste
Selección de alarmas	Habilitar alarmas de diagnóstico
Modos de operación	• Rango de contaje: -31 a +31 bits • Modo de operación: contaje sin fin • Control de puerta: ninguno
Entradas	Si desea realizar una búsqueda del punto de referencia, debe elegir el ajuste siguiente: • Activar contador (DI-Set): una vez • El ajuste "Evaluar marca cero para el posicionamiento" se puede activar o desactivar como criterio para el punto de referencia.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se comunica al driver de entrada un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación).
ReadDiagErr	IN-OUT	BOOL	Active este bit en el OB 82 cuando el módulo haya disparado una alarma de diagnóstico (véase Tratamiento de errores).
DI_0	OUTPUT	BOOL	Entrada digital 0
DI_1	OUTPUT	BOOL	Entrada digital 1
DI_2	OUTPUT	BOOL	Entrada digital 2
Los siguientes parámetros estáticos de la instrucción sirven solamente para fines de diagnóstico en caso de fallo.			
JobErr	STAT	INT	Valor de retorno de la instrucción RD_REC si se ha activado "ReadDiagErr" = TRUE.
DiagStatus.EXT_VOLTAGE	STAT	STRUCT	Fallo en la tensión auxiliar externa
DiagStatus.NO_CONFIG	STAT	STRUCT	Falta parametrización
DiagStatus.CONFIG_ERR	STAT	STRUCT	Error de parametrización
DiagStatus.WTCH_DOG_FLT	STAT	STRUCT	La vigilancia de tiempo (watchdog) ha saltado
DiagStatus.RAM_FLT	STAT	STRUCT	Defecto en RAM
DiagStatus.CH1_SIGA	STAT	STRUCT	Señal A errónea
DiagStatus.CH1_SIGB	STAT	STRUCT	Señal B errónea
DiagStatus.CH1_SIGZ	STAT	STRUCT	Señal N errónea
DiagStatus.CH1_BETW	STAT	STRUCT	Error entre canales
DiagStatus.CH1_5V2	STAT	STRUCT	Fallo en alimentación de 5,2 V del encóder

Sincronización

La sincronización se puede realizar

- mediante búsqueda del punto de referencia
- mediante posicionamiento del punto cero de la máquina.

Conecte el sensor del punto de referencia a la entrada digital I2.

En una búsqueda del punto de referencia, el módulo puede detectar el punto de referencia a elección en los siguientes casos:

- Flanco ascendente en la entrada digital I2
- Flanco ascendente de la marca cero, si la entrada digital I2 está activada.

Si el eje ya se encuentra en el sensor del punto de referencia al inicio de la búsqueda del punto de referencia, se efectuará una referenciación en la posición actual, al igual que en el posicionamiento del punto de referencia.

Entradas digitales

Las entradas digitales I0 a I2 se indican en los parámetros de salida "DI_0" a "DI_2" del driver de entrada.

Tratamiento de errores

El módulo proporciona información de errores a través del registro 1 (DS1). En los siguientes casos, el driver de entrada lee el registro 1 del módulo mediante la instrucción RD_REC y lo almacena en el parámetro estático "DiagStatus":

- El driver de entrada se inicializa.
- El módulo notifica que ha detectado un error.
- El parámetro de entrada "ReadDiagErr" del driver es TRUE.

A continuación, ajusta "ReadDiagErr" = FALSE. El valor de retorno de la instrucción RD_REC se guarda en el parámetro estático "JobErr" (véase Avisos JOB_Err (Página 11281)).

Si se ha registrado una señalización de errores en el registro 1, el driver de entrada ajusta en el DB de eje los identificadores de error "Error" y "Err.EncoderErr" a TRUE. El bit de estado "Sync" se ajusta a FALSE.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

EncoderCPU314C

Descripción

La instrucción EncoderCPU314C sirve para conectar un canal de contaje de la CPU 314C para la lectura de recorrido para encóders incrementales de 24 V.

Pueden utilizarse hasta 4 canales de contaje.

Parametrización de la función de contaje de la CPU 314C (configuración hardware)

El driver de entrada solamente es apto para una CPU 314C a partir de la versión de firmware V2.0.

Ajuste para cada canal de contaje que desee utilizar los siguientes parámetros en el diálogo de propiedades del módulo en el slot 2.4 (contaje):

Parámetro	Ajuste
Modo de operación	Contaje sin fin
Puerta HW	Sin puerta HW (casilla de verificación desactivada)

Los demás ajustes se parametrizan en función del encóder conectado y de los requisitos de su aplicación.

En los datos de eje del parámetro "Dirección inicial de las entradas del módulo de lectura de recorrido" ("InputModuleInAddress") introduzca la dirección que ha asignado en la configuración hardware para las entradas en el slot 2.4 (contaje). El parámetro "Dirección inicial de las salidas del módulo de lectura de recorrido" ("InputModuleOutAddress") se ajusta con la dirección de las salidas de contaje.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se notifica al driver de entrada un error detectado por el programa de usuario.

Sincronización

La sincronización se puede realizar

- mediante búsqueda del punto de referencia
- mediante posicionamiento del punto cero de la máquina.

El sensor del punto de referencia se conecta en la entrada digital Latch del canal correspondiente (DI1.4 para canal 0 hasta DI1.7 para canal 3).

En una búsqueda del punto de referencia, la CPU 314C puede detectar el punto de referencia con un flanco ascendente en la entrada digital DI.

Si el eje ya se encuentra en el sensor del punto de referencia al inicio de la búsqueda del punto de referencia, se efectuará una referenciación en la posición actual, al igual que en el posicionamiento del punto de referencia.

En una búsqueda del punto de referencia, el driver lee cíclicamente, entre otros, el estado del sensor del punto de referencia. Los accesos necesarios para ello requieren un tiempo de ejecución claramente superior de la instrucción.

Máxima frecuencia de contaje

Nota

Si se rebasa la máxima frecuencia de contaje del contador CPU 314C, no se emiten valores de contaje (véase Instrucciones de servicio SIMATIC S7-300 CPU 31xC Funciones tecnológicas (http://support.automation.siemens.com/WW/view/es/12429336)). El eje se detiene con el error "Error de seguimiento rebasado (Página 11278)".

Por tanto, defina la velocidad máxima ("MaxVelocity") en los datos de eje teniendo en cuenta la frecuencia límite del contador.

Tratamiento de errores

La CPU 314C no proporciona información de errores a través del encóder.

EncoderIM174

Descripción

La instrucción EncoderIM174 sirve para utilizar el IM 174 como módulo de lectura de recorrido para encóders incrementales y absolutos.

El IM 174 ofrece hasta 4 ejes. Para cada eje se necesita una instrucción de driver EncoderIM174.

Parametrización del IM 174 (configuración hardware)

Añada un IM 174 a un PROFIBUS DP isócrono del proyecto. Tenga en cuenta que las direcciones de entrada y salida se encuentran en la memoria imagen de proceso de la CPU. En caso necesario, corrija las direcciones adaptándolas manualmente. Asigne las direcciones de entrada/salida a una memoria imagen parcial de proceso. En la configuración de la CPU S7 debe asignarse un OB isócrono a la memoria imagen parcial de proceso.

Parámetro	Ajuste
Dirección de E/S	Para el eje que se va a utilizar en Easy Motion Control, debe indicarse la primera dirección de entrada en el DB de eje. (p. ej., slot 4: dirección I 256 a 273, entonces debe indicarse la dirección de entrada 256)
Selección de telegramas	En la configuración hardware se puede seleccionar para cada eje el telegrama 3 o el telegrama 81 de PROFIBUS. Esta información debe indicarse también en la entrada TelNumber de la instrucción EncoderIM174.
Modo conforme 611U	Para utilizar una marca de referencia externa es preciso activar el modo conforme 611U.

Nota

Si se utiliza el telegrama 81, solamente se transmitirá la información del encóder desde el IM 174 a la CPU. El telegrama 81 se utiliza si sólo se desea leer un encóder con el IM 174. En este caso tampoco se necesita driver de salida para el IM 174 (OutputIM174).

Nota

De forma estándar, la configuración hardware integra el módulo IM 174 como módulo isócrono. Debido al procesamiento isócrono, las direcciones de entrada y salida del IM 174 se encuentran en la memoria imagen de proceso de la CPU.

Para información más detallada sobre el modo isócrono en SIMATIC S7, consulte el Manual de funciones Modo isócrono (http://support.automation.siemens.com/WW/view/en/15218045).

La instrucción de driver EncoderIM174 presupone el procesamiento isócrono. Si no se tiene en cuenta esta condición, la instrucción no puede acceder a las direcciones de hardware.

Parametrización del IM 174 en el DB de eje

La dirección de entrada ajustada para este eje en la configuración hardware debe indicarse en el campo "Dirección del módulo Entradas".

El número de eje para el IM 174 es calculado por el driver de entrada EncoderIM174 durante el funcionamiento. Se puede leer en los datos locales del correspondiente DB.

El driver de entrada EncoderIM174 calcula durante el funcionamiento el tiempo de ciclo ajustado en PROFIBUS DP isócrono y lo introduce en el DB de eje. Por tanto, se recomienda encarecidamente volver a leer el DB de eje modificado del controlador, a fin de adoptar los posibles cambios.

Llamada

La instrucción se llama en un OB de alarma isócrona (p. ej. OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
RefMode	INPUT	INT	Tipo de referenciación: • 0 = marca cero del encóder • 1 = flanco ascendente en marca cero externa (presupone modo conforme 611U). • 4 = palpador 1 flanco ascendente • 5 = palpador 1 flanco descendente • 6 = palpador 2 flanco ascendente • 7 = palpador 2 flanco descendente
TelNumber	INPUT	INT	Número del telegrama PROFIBUS utilizado: • 3 = telegrama PROFIBUS 3 • 81 = telegrama PROFIBUS 81
Err_ID	OUTPUT	WORD	Código de error de la instrucción: • W#16#0001 = OB erróneo, la instrucción sólo puede instalarse en el OB 6x • W#16#0002 = número de telegrama erróneo, el IM 174 sólo soporta los telegramas 3 y 81 de PROFIBUS • W#16#0004 = número de eje erróneo, el IM 174 soporta un máximo de 4 ejes. La dirección de hardware que ha indicado en el DB de eje no ha podido asignarse a ningún eje de un IM 174. • W#16#0008 = se ha indicado un modo de referenciación erróneo. Compruebe el parámetro en la entrada RefMode.

Sincronización

La sincronización se realiza alternativamente:

- Mediante la búsqueda del punto de referencia (sólo con encóder incremental)
- Mediante el posicionamiento del punto de referencia (con encóder incremental y absoluto)

Para la búsqueda del punto de referencia pueden utilizarse dos tipos de sensores:

- Si utiliza un encóder con marca cero y desea referenciar a éste, parametrice la entrada "RefMode" = 0. El modo conforme 611U no debe estar seleccionado.
- Si desea referenciar en una marca cero externa (BERO), parametrice la entrada "RefMode" = 1. El sensor del punto de referencia externo se conecta a la entrada digital B1 a B4, dependiendo del eje utilizado en la aplicación. Es imprescindible que en este caso el modo conforme 611U se seleccione en la configuración hardware.

Como entradas para los dos palpadores ajustables deben utilizarse las entradas digitales M1 o M2.

Para más información, consulte el Manual de producto Periferia descentralizada módulo PROFIBUS IM 174 (http://support.automation.siemens.com/WW/view/en/35014863).

Si el eje ya se encuentra en el sensor del punto de referencia al inicio de la búsqueda del punto de referencia, se efectuará una referenciación en la posición actual, al igual que en el posicionamiento del punto de referencia.

Tratamiento de errores

Como reacción a un error, el driver de entrada ajusta al detectar el error los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje. El driver de entrada desactiva el bit de estado "Sync" en el DB de eje, si es que se utiliza un encóder incremental.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

Acuse de errores

Si activa el acuse de errores en el DB de eje de Easy Motion Control, se acusarán también los errores pendientes en el módulo.

EncoderSINAMICS

Descripción

La instrucción EncoderSINAMICS sirve para integrar un accionamiento SINAMICS en Easy Motion Control. La integración puede realizarse mediante PROFIBUS DP o PROFINET IO. Para cada eje SINAMICS se necesita una instrucción de driver EncoderSINAMICS.

Herramientas necesarias

Para utilizar un SINAMICS en Easy Motion Control y para parametrizar un SINAMICS se necesitan las siguientes herramientas:

- La herramienta de parametrización de accionamientos STARTER o S7-Technology, incluido el actual SINAMICS Support Package (SSP).

Parametrización del SINAMICS (configuración hardware)

Añada un accionamiento SINAMICS a una línea PROFIBUS DP o PROFINET IO de su proyecto. La instrucción de driver EncoderSINAMICS soporta para PROFIBUS DP los telegramas estándar 3, 5, 102 y 105, así como los telegramas estándar 3 y 102 para PROFINET IO.

Si desea utilizar para su aplicación el método de regulación Dynamic Servo Control (DSC), necesitará un telegrama adecuado (5, 105) y una conexión isócrona. Con PROFINET IO actualmente no es posible esta funcionalidad DSC.

Para el procesamiento isócrono, las direcciones de entrada y salida deben encontrarse en la memoria imagen de proceso de la CPU. En caso necesario, corrija las direcciones adaptándolas manualmente. Asigne las direcciones de entrada/salida a una memoria imagen parcial de proceso. En la configuración de la CPU S7 debe asignarse un OB isócrono a la memoria imagen parcial de proceso. Para información más detallada sobre el modo isócrono en SIMATIC S7, consulte el Manual de funciones Modo isócrono (http://support.automation.siemens.com/WW/view/en/15218045).

Parámetro	Ajuste
Dirección de E/S	Para el eje que se va a utilizar en Easy Motion Control, debe indicarse la primera dirección de entrada en el DB de eje. (p. ej., slot 4: dirección I 256 a 273, entonces debe indicarse la dirección de entrada 256)
Selección de telegramas	Para la comunicación se selecciona en el asistente de configuración de SINAMICS un número de telegrama y se transfiere a la configuración hardware. Esta información debe indicarse también en la entrada TelNumber de la instrucción EncoderSINAMICS.

Nota

La instalación estándar del TIA Portal no soporta la conexión de SINAMICS a una CPU SIMATIC S7. Compruebe la disponibilidad. En caso necesario, instale el archivo GSD correspondiente para SINAMICS. Los datos necesarios se integran solamente una vez que se haya realizado la instalación adicional.

Parametrización del SINAMICS en el DB de eje

La dirección de entrada ajustada para este eje en la configuración hardware debe indicarse en el campo "Dirección del módulo Entradas".

El driver de entrada EncoderSINAMICS calcula durante el funcionamiento el tiempo de ciclo ajustado cuando se encuentra un modo equidistante o isócrono, y lo introduce en el DB de eje. Por tanto, se recomienda encarecidamente volver a leer el DB de eje modificado del controlador, a fin de adoptar los posibles cambios.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción	
RefMode	INPUT	INT	Tipo de referenciación:	
			0 = función 1 marca de referencia 1	Referenciación en marca cero del encóder o marca cero sustitutiva (p495[0] en SINAMICS)
			1 = función 2 marca de referencia 2	Segunda marca cero
			2 = función 3 marca de referencia 3	Tercera marca cero
			3 = función 4 marca de referencia 4	Cuarta marca cero
TelNumber	INPUT	INT	Número de telegrama:	

Parámetro	Declaración	Tipo de datos	Descripción	
			Se soportan los siguientes telegramas estándar: 3, 5, 102 y 105	Para PROFINET IO solamente se han habilitado los telegramas 3 y 102.
EncoderValid	OUTPUT	BOOL	1 = valores reales de posición válidos	SINAMICS notifica que el encóder de posición del eje en cuestión proporciona valores válidos. Esta entrada se puede utilizar como condición de conexión en MC_Control.
Err_ID	OUTPUT	WORD	Código de error de la instrucción:	
			W#16#0001 = OB erróneo	La instrucción sólo puede llamarse en el OB 6x, OB 3x o bien en el OB 1.
			W#16#0002 = número de telegrama erróneo	El número indicado no es soportado por el driver.
			W#16#0004 = se ha indicado un modo de referenciación erróneo	Compruebe el parámetro en la entrada RefMode. Los valores válidos son 0, 1, 2 o 3.

Nota

El número de telegrama en el driver de entrada debe ser idéntico al número de telegrama configurado para SINAMICS. De lo contrario no se garantiza la perfecta funcionalidad. En caso de introducción errónea, no se emite código de error.

Sincronización

La sincronización se realiza alternativamente:

- Mediante la búsqueda del punto de referencia (sólo con encóder incremental)
- Mediante el posicionamiento del punto de referencia (para encóder incremental y absoluto)

El modo de referenciación puede determinarse mediante el parámetro de entrada "RefMode" en el driver de entrada. La siguiente tabla muestra las posibilidades existentes para Easy Motion Control.

RefMode	"Velocity" en MC_Home	Señal de marca cero del encóder	Marca cero sustitutiva p495[0]	Descripción
0	> 0	primera marca cero	0	Búsqueda del punto de referencia en primera marca cero
1	> 0	Segunda marca cero	0	Búsqueda del punto de referencia en segunda marca cero
2	> 0	Tercera marca cero	0	Búsqueda del punto de referencia en tercera marca cero
3	> 0	Cuarta marca cero	0	Búsqueda del punto de referencia en cuarta marca cero

Mediante la introducción en RefMode se controla la correspondiente palabra de control del encóder para el accionamiento, de forma que se realiza la funcionalidad deseada en SINAMICS. El driver de entrada EncoderSINAMICS permite solamente la búsqueda del punto de referencia en una o varias marcas cero del encóder, así como en marcas cero externas (marca cero sustitutiva). No es posible referenciar al vuelo. Para referenciar en la marca cero sustitutiva, es preciso ajustar en SINAMICS el valor del parámetro p495[0] del eje en cuestión a un valor entre 1 y 6. Para activar una búsqueda del punto de referencia, es preciso indicar una velocidad de desplazamiento mayor que cero en la instrucción MC_HOME.

Si se introduce aquí una velocidad de desplazamiento cero, no se tienen en cuenta los contenidos de la entrada "RefMode" y del parámetro p495[0].

Tratamiento de errores

Como reacción a un error, el driver de entrada ajusta al detectar el error los identificadores de error "Error" y "Err.EncoderErr" en el DB de eje. El driver de entrada desactiva el bit de estado "Sync" en el DB de eje, si es que se utiliza un encóder incremental.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

Acuse de errores

Si activa el acuse de errores en el DB de eje de Easy Motion Control, se acusarán también los errores pendientes en el módulo.

EncoderUniversal

Descripción

La instrucción EncoderUniversal sirve para integrar los módulos de lectura de recorrido para los que Easy Motion Control no ofrece un driver de entrada específico.

Funcionamiento

La instrucción EncoderUniversal espera en la entrada "EncoderValue" un valor que corresponde a la posición real del eje. No accede por sí misma a la periferia. Es posible interconectar el valor proporcionado por el módulo de lectura de recorrido directamente con el encóder o bien, si es necesario, calcularlo antes.

Si el módulo de lectura de recorrido ofrece funciones especiales que desee utilizar, deberá integrarlas en el programa de usuario.

Parametrización del módulo (configuración hardware)

Seleccione los parámetros en consonancia con el hardware y con el encóder utilizado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EncErr	INPUT	BOOL	1 = error de encóder Aquí se notifica al driver de entrada un error detectado por el módulo de lectura de recorrido o por el programa de usuario (p. ej., módulo retirado o fallo de estación).
EncoderValue	INPUT	DINT	Valor del encóder

Sincronización

Con la instrucción EncoderUniversal la sincronización solamente es posible mediante posicionamiento del punto cero de la máquina.

Tratamiento de errores

Los errores detectados por el módulo de lectura de recorrido deben registrarse en el programa de usuario y seguir procesándose.

Si un error de este tipo afecta a las funciones que se realizan con instrucciones de Easy Motion Control, es preciso integrar este error a través de la entrada "EncErr" en el tratamiento de errores de Easy Motion Control.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

Driver de salida

Función de los drivers de salida

Tarea

El driver de salida sirve para emitir la consigna de velocidad calculada por el regulador de un eje al módulo de salida conectado. La instrucción convierte la velocidad de [unidad de longitud/s] al rango de valores adecuado del módulo de salida y emite este valor.

Si la entrada "EnableDrive" = FALSE, el driver de salida da el valor cero al módulo de salida.

Inicialización

El valor de salida se ajusta a cero.

Llamada

El driver de salida de un eje debe llamarse sin condición.

Modo isócrono

Los drivers de salida soportan el modo isócrono de los módulos de periferia adecuados. El driver detecta el modo isócrono automáticamente porque es llamado en un OB isócrono.

Tratamiento de errores

Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar al driver de salida en la entrada "OutErr". Mientras la entrada "OutErr" sea igual a TRUE, el driver de salida no accederá al módulo de salida.

Si la entrada "OutErr" está ajustada a TRUE, el driver de salida activa los identificadores de error "Error" y "Err.OutputErr" en el DB de eje.

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

Modo de simulación

Mientras el bit "Sim" esté activado en el DB de eje, el driver de salida no accederá al módulo de salida (véase la instrucción MC_Simulation (Página 7341)).

OutputIM178

Descripción

La instrucción OutputIM178 sirve para utilizar un canal del módulo IM 178-4 como módulo de salida. La instrucción OutputIM178 está contenida en STEP7 V5.x por razones de compatibilidad. El módulo IM 178-4 ya no se comercializa activamente. Como alternativa, se puede utilizar el IM 174.

El IM 178-4 debe utilizarse con el mismo número de canal y el mismo eje en el driver de entrada **y** en el driver de salida, pues hay determinados ajustes de direcciones que se refieren a ambos drivers. No es posible mezclar con otros drivers.

Llamada

La instrucción se llama en un OB de alarma isócrona (p. ej. OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica Si "EnableDrive" = FALSE, se transferirá el valor cero al módulo de salida. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
Q0	INPUT	BOOL	Salida digital 0 del canal seleccionado
Q1	INPUT	BOOL	Salida digital 1 del canal seleccionado
Q2	INPUT	BOOL	Salida digital 2 del canal seleccionado
OutErr	INPUT	BOOL	1 = error en el módulo de salida Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar aquí al driver de salida.

Salidas digitales

Las salidas digitales "Q0" a "Q2" del canal seleccionado del IM 178-4 pueden controlarse mediante los parámetros de entrada "Q0" a "Q2" del driver de salida.

OutputSM432

Descripción

La instrucción OutputSM432 sirve para utilizar un canal del SM 432 como módulo de salidas.

Parametrización del SM 432 (configuración hardware)

Parametrice el canal correspondiente del módulo SM 432 con una tensión de salida de ±10 V.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica Si "EnableDrive" = FALSE, se transferirá el valor cero al módulo de salida. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
OutErr	INPUT	BOOL	1 = error en el módulo de salida Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar aquí al driver de salida.

OutputET200S2AO

Descripción

La instrucción OutputET200S2AO sirve para utilizar un canal del módulo de salidas analógicas ET 200S 2AO U como módulo de salidas.

El driver también se puede utilizar para la variante isócrona de este módulo.

Parametrización del módulo de salidas analógicas ET 200S 2AO U (configuración hardware)

Parametrice el canal correspondiente del módulo con una tensión de salida de ±10 V.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica Si "EnableDrive" = FALSE, se transferirá el valor cero al módulo de salida. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
OutErr	INPUT	BOOL	1 = error en el módulo de salida Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar aquí al driver de salida.

OutputCPU314C

Descripción

La instrucción OutputCPU314C sirve para utilizar un canal de salida analógica de CPU 314C para una tensión de salida de ±10 V.

Parametrización de las salidas analógicas de la CPU 314C (configuración hardware)

Parametrice el canal correspondiente (salida analógica) de la CPU 314C con una tensión de salida de ±10 V.

En los datos de eje del parámetro "Dirección inicial de las salidas del módulo de salidas" ("OutputModuleOutAddress"), introduzca la dirección que ha asignado en la configuración hardware para las salidas en el slot 2.3 (AI5/AO2). El parámetro "Dirección inicial de las entradas del módulo de salidas" ("OutputModuleInAddress") no necesita ser introducido.

Hay un máximo de 2 canales de salida analógica disponibles, de manera que pueden utilizarse 2 ejes completamente a través de CPU 314C.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica Si "EnableDrive" = FALSE, se transferirá el valor cero a la salida analógica. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
OutErr	INPUT	BOOL	1 = error en la salida analógica Aquí se puede notificar al driver de salida un error detectado por el programa de usuario.

OutputSM332

Descripción

La instrucción OutputSM332 sirve para utilizar un canal del SM 332 como módulo de salidas.

El driver también se puede utilizar para la variante isócrona de este módulo.

Parametrización del SM 332 (configuración hardware)

Parametrice el canal correspondiente del módulo SM 332 con una tensión de salida de ±10 V.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica Si "EnableDrive" = FALSE, se transferirá el valor cero al módulo de salida. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
OutErr	INPUT	BOOL	1 = error en el módulo de salida Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar aquí al driver de salida.

OutputUniversal

Descripción

Con la instrucción OutputUniversal puede utilizarse un módulo de salidas analógicas cualquiera junto con instrucciones de Easy Motion Control.

Funcionamiento

La instrucción OutputUniversal, calcula una consigna de velocidad a partir de la velocidad emitida por el regulador y con ayuda del factor de proporcionalidad "IntAtMaxVelocity" y la pone a disposición en sus salidas.

El driver no accede por sí mismo a la periferia.

"IntAtMaxVelocity" es el valor con el que debe controlarse el módulo de salidas analógicas para que emita una tensión que desplace el eje a la "MaxVelocity" (el valor de esta tensión ya se ha definido en el parámetro "DriveInputAtMaxVel").

La curva característica del driver pasa siempre por el cero y es lineal.

La consigna de velocidad se emite como valor entero con signo y también como valor absoluto con signo aparte.

Si estos valores de salida ya son adecuados para la aplicación, puede emitirlos directamente a la periferia mediante el programa de usuario. Si todavía no son adecuados, es posible realizar una adaptación adicional mediante el programa de usuario y luego emitir el valor así generado.

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica En cuanto "EnableDrive" = FALSE, se emite el valor cero a las salidas. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
IntAtMaxVelocity	INPUT	INT	Valor entero para la conversión de la consigna de velocidad (ver arriba).
OutErr	INPUT	BOOL	1 = error en el módulo de salidas Aquí se puede notificar al driver de salida un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) en el módulo de salidas.
OutputValue	OUTPUT	INT	Consigna de velocidad calculada
OutValAbs	OUTPUT	INT	Valor absoluto de la consigna de velocidad
OutValSign	OUTPUT	BOOL	Signo de la consigna de velocidad (TRUE = valor negativo)

OutputMM4_DP

Descripción

La instrucción OutputMM4_DP sirve para utilizar un MICROMASTER 4 con módulo opcional DP enchufado.

Parametrización del MM4 (configuración hardware)

El MICROMASTER 4 se encuentra en el catálogo de hardware en:

"Otros dispositivos de campo > PROFIBUS DP > Accionamientos > Siemens AG > SIMOVERT > MICROMASTER 4".

Puede que primero sea necesario instalar el archivo GSD correspondiente.

Coloque el MICROMASTER en el sistema maestro DP y asígnele la dirección DP que desee. En los slots 0 y 1, inserte un módulo "0 PKW, 2 PZD (PPO3)". Acepte las direcciones de periferia propuestas o seleccione otros valores.

Las direcciones aquí introducidas se aplican a la "Dirección inicial de las salidas del módulo de salidas" o a la "Dirección inicial de las entradas del módulo de salidas" en el DB del eje.

Parametrización del MM4

Parametrice el MICROMASTER según sus necesidades (datos del motor, dirección DP, funcionamiento como esclavo DP...) sirviéndose de la documentación MICROMASTER.

La "frecuencia de referencia" introducida en el parámetro P2000 debe aplicarse al DB del eje como "Valor de referencia para una velocidad del 100%" ("DriveInputAt100").

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica En cuanto "EnableDrive" = FALSE, se emite el valor de velocidad cero al MICROMASTER. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
OutErr	INPUT	BOOL	1 = error en la salida Aquí se puede notificar al driver de salida un error detectado por el programa de usuario (p. ej. fallo de estación).
ON	INPUT	BOOL	1 = accionamiento conectado "ON" se introduce en la palabra de mando, en el bit 0 .
DriveReady	OUTPUT	BOOL	1 = accionamiento listo para el funcionamiento Si "DriveReady" = FALSE, el inicio de una marcha generará un "DriveError".

OutputIM174

Descripción

La instrucción OutputIM174 sirve para utilizar un canal del módulo IM 174 como módulo de salida. El IM 174 ofrece hasta 4 ejes. Para cada eje con telegrama 3 de PROFIBUS ajustado se necesita una instrucción de driver OutputIM174.

Parametrización del IM 174 (configuración hardware)

El driver de salida OutputIM174 solo es necesario cuando se parametriza un eje con telegrama 3 de PROFIBUS. Para un eje con telegrama 81 de PROFIBUS no se necesita driver de salida.

Añada un IM 174 a un PROFIBUS DP isócrono del proyecto. Tenga en cuenta que las direcciones de entrada y salida se encuentran en la memoria imagen de proceso de la CPU. En caso necesario, corrija las direcciones adaptándolas manualmente. Asigne las direcciones

de entrada/salida a una memoria imagen parcial de proceso. En la configuración de la CPU S7 debe asignarse un OB isócrono a la memoria imagen parcial de proceso.

Parámetro	Ajuste
Dirección de E/S	Para el eje que se va a utilizar en Easy Motion Control, debe indicarse la primera dirección de salida en el DB de eje. (p. ej., slot 5: dirección Q 256 a 265, entonces debe indicarse la dirección de salida 256)

Nota

De forma estándar, la configuración hardware integra el módulo IM 174 como módulo isócrono. Debido al procesamiento isócrono, las direcciones de entrada y salida del IM 174 se encuentran en la memoria imagen de proceso de la CPU.

Para información más detallada sobre el modo isócrono en SIMATIC S7, consulte el Manual de funciones Modo isócrono (http://support.automation.siemens.com/WW/view/en/15218045).

La instrucción de driver OutputIM174 presupone el procesamiento isócrono. Si no se tiene en cuenta esta condición, la instrucción no puede acceder a las direcciones de hardware.

Parametrización del IM 174 en el DB de eje

La dirección de salida ajustada para este eje en la configuración hardware debe indicarse en el campo "Dirección del módulo Salidas".

Llamada

La instrucción se llama en un OB de alarma isócrona (p. ej. OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
EnableDrive	INPUT	BOOL	1 = habilitación de la salida analógica En cuanto "EnableDrive" = FALSE, se emite el valor de velocidad cero al accionamiento. La habilitación del accionamiento se bloquea. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
OutErr	INPUT	BOOL	1 = error en el módulo de salida Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar aquí al driver de salida.
Err_ID	OUTPUT	WORD	Código de error de la instrucción W#16#0001 = OB erróneo, la instrucción sólo puede instalarse en el OB 6x

OutputSINAMICS

Descripción

La instrucción OutputSINAMICS sirve para integrar un accionamiento SINAMICS en Easy Motion Control. La integración puede realizarse mediante PROFIBUS DP o PROFINET IO. Para cada eje SINAMICS se necesita una instrucción de driver OutputSINAMICS.

Parametrización del SINAMICS (configuración hardware)

Añada un accionamiento SINAMICS a una línea PROFIBUS DP o PROFINET IO de su proyecto. La instrucción de driver OutputSINAMICS soporta para PROFIBUS DP los telegramas estándar 3, 5, 102 y 105, así como los telegramas estándar 3 y 102 para PROFINET IO.

Si desea utilizar para su aplicación el método de regulación Dynamic Servo Control (DSC), necesitará un telegrama adecuado (5, 105) y una conexión isócrona. Con PROFINET IO actualmente no es posible esta funcionalidad DSC.

Para el procesamiento isócrono, las direcciones de entrada y salida deben encontrarse en la memoria imagen de proceso de la CPU. En caso necesario, corrija las direcciones adaptándolas manualmente. Asigne las direcciones de entrada/salida a una memoria imagen parcial de proceso. En la configuración de la CPU S7 debe asignarse un OB isócrono a la memoria imagen parcial de proceso. Para información más detallada sobre el modo isócrono en SIMATIC S7, consulte el Manual de funciones Modo isócrono (http://support.automation.siemens.com/WW/view/en/15218045).

Parámetro	Ajuste
Dirección de E/S	Para el eje que se va a utilizar en Easy Motion Control, debe indicarse la primera dirección de salida en el DB de eje. (p. ej., slot 5: dirección Q 256 a 265, entonces debe indicarse la dirección de salida 256)

Nota

La instalación estándar del TIA Portal no soporta la conexión de SINAMICS a una CPU SIMATIC S7. Compruebe la disponibilidad. En caso necesario, instale el archivo GSD correspondiente para SINAMICS. Los datos necesarios se integran solamente una vez que se haya realizado la instalación adicional.

Parametrización del SINAMICS en el DB de eje

La dirección de salida ajustada para este eje en la configuración hardware debe indicarse en el campo "Dirección del módulo Salidas". El número de telegrama debe indicarse de nuevo porque el driver de salida también podría funcionar solo. Se comprueba si el número de telegrama coincide con el correspondiente driver de entrada EncoderSINAMICS (en su caso).

Conexión del eje

Cada módulo de motor SINAMICS tiene asignada obligatoriamente una alimentación central de energía para el circuito intermedio (un Line Module). Antes de conectar el eje en la entrada "EnableDrive" de la instrucción MC_Control es preciso asegurarse de que la alimentación central de energía ha sido conectada. Para ello, el Line Module envía las respuestas correspondientes a través de su estado operativo actual. Evalúe las respuestas necesarias del Line Module y vincúlelas como condición de conexión.

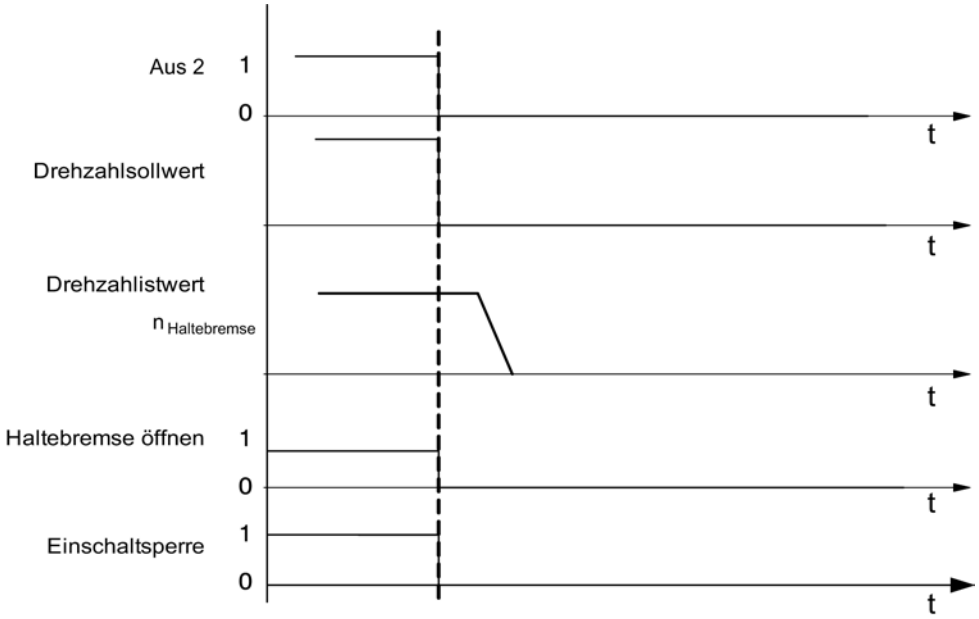
Para más información, consulte la documentación sobre accionamientos SINAMICS (http://www.automation.siemens.com/doconweb).

Para conectar un Active Line Module, encontrará un ejemplo de aplicación en Internet (http://support.automation.siemens.com/WW/view/en/21971603).

Desconexión del eje: parametrización del StopMode

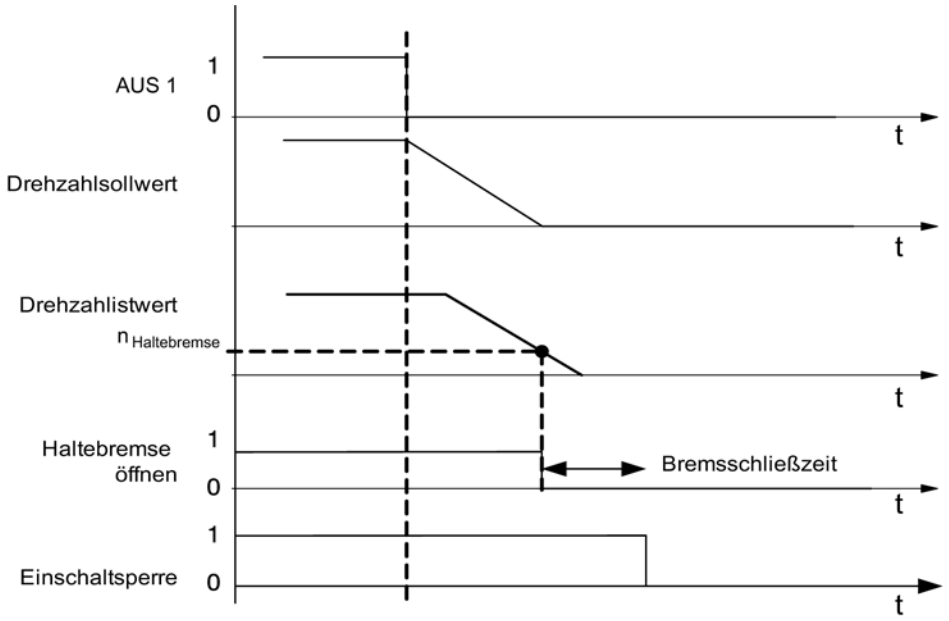
Parada natural (Coast Stop) (AUS2) – StopMode = 3:

Aquí se anula de inmediato la habilitación de potencia y el accionamiento gira por inercia hasta detenerse. Paralelamente se activa de inmediato el control del freno mecánico. El freno actúa y detiene el accionamiento. Según la carga aplicada, el freno se ve sometido a un gran esfuerzo mecánico.



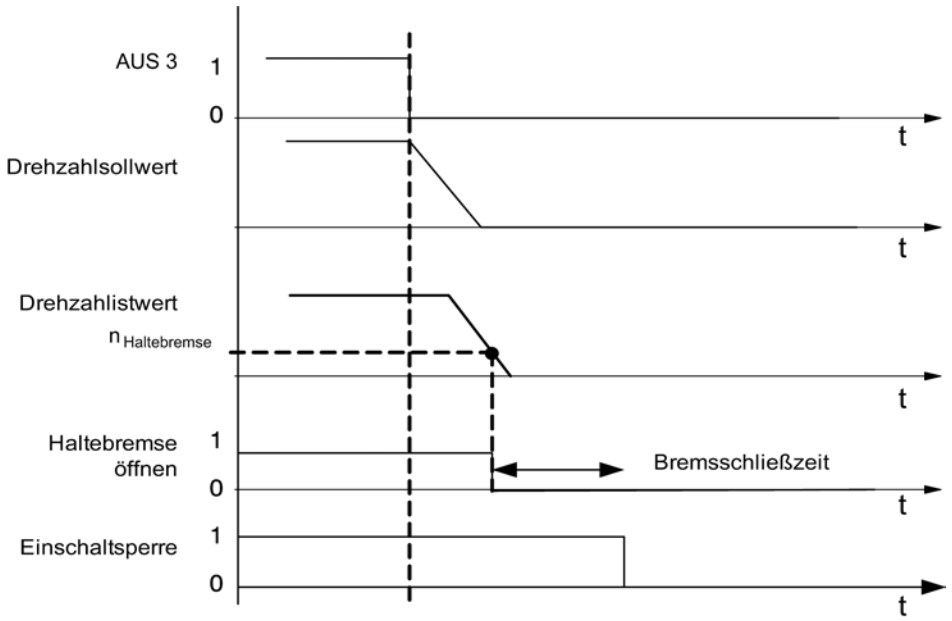
Parada en rampa (AUS1) – StopMode = 5:

Aquí se frena en la rampa del generador de rampas del accionamiento. Tras alcanzarse el umbral de velocidad el freno actúa, y transcurrido el tiempo de cierre del freno se desactiva la habilitación de impulsos.



Parada rápida (AUS3) – StopMode = 6:

Aquí se frena en la rampa de parada rápida (parada de emergencia) del accionamiento. Tras alcanzarse el umbral de velocidad el freno actúa, y transcurrido el tiempo de cierre del freno se desactiva la habilitación de impulsos.



Para más información, consulte la Información de producto SINAMICS S120 (http://support.automation.siemens.com/WW/view/en/30119684).

Llamada

La instrucción se llama en un OB de alarma cíclica (p. ej. OB 35) o en un OB de alarma de sincronismo (p. ej., OB 61). Todas las instrucciones EMC de un eje deben llamarse en un mismo bloque de programa.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción	
TelNumber	INPUT	INT	Número de telegrama:	
			Se soportan los siguientes telegramas estándar: 3, 5, 102,105	Para PROFINET IO solamente se han habilitado los telegramas 3 y 102. Si como driver de entrada se utiliza el EncoderSINAMICS, es imprescindible que el número del telegrama empleado sea el mismo.
EnableDrive	INPUT	BOOL	1 = habilitación del accionamiento	En cuanto "EnableDrive" = FALSE, se emite el valor de velocidad cero al accionamiento. La habilitación del accionamiento se bloquea. Esta entrada puede interconectarse con la salida "DriveEnabled" de la instrucción MC_Control.
StopMode	INPUT	INT	Tipo de desconexión:	
			3 = parada natural (AUS2)	El accionamiento gira por inercia hasta detenerse
			5 = parada en rampa (AUS1)	Parada del eje a través del generador de rampas
			6 = parada rápida (AUS3)	Parada del eje a través de la rampa de parada rápida
OutErr	INPUT	BOOL	1 = error en el módulo de salida	Un error detectado por el programa de usuario (p. ej., módulo retirado o fallo de estación) se puede notificar aquí al driver de salida.
Operation_Enabled	OUTPUT	BOOL	1 = SINAMICS notifica "Servicio habilitado"	El accionamiento cuenta con todas las habilitaciones para el eje y el eje señaliza la disponibilidad para desplazarse. El eje está magnetizado.
Err_ID	OUTPUT	WORD	Código de error de la instrucción:	
			W#16#0001 = OB erróneo	La instrucción solo puede instalarse en el OB 6x, OB 3x o bien en el OB 1
			W#16#0002 = número de telegrama erróneo	El número indicado no es soportado por el driver o bien difiere del número de telegrama configurado en SINAMICS.

OutErr: con un flanco ascendente de estas variables el accionamiento (SINAMICS) se desconecta. Para volver a conectar el eje es necesario que OutErr = FALSE y se haya acusado el mensaje de error.

Operation_Enabled: esta variable notifica el estado actual del eje en SINAMICS. Mientras Operation_Enabled = FALSE no se debe activar ninguna petición de desplazamiento.

Encontrará más información sobre la conexión del SINAMICS en Internet (http://support.automation.siemens.com/WW/view/en/30119684).

Otras referencias para Easy Motion Control

Bloque de datos de eje (DB de eje)

Tarea

El DB de eje es el elemento central de la gestión de datos de Easy Motion Control. Contiene:

- Parámetros que describen el eje sustancialmente (p. ej. direcciones de periferia)
- Parámetros de la regulación de posición que se calculan a partir de estos parámetros básicos al inicializarse las instrucciones EMC
- Valores actuales de la regulación de posición (p. ej. posición, estados de fallo...)
- Bits de inicialización para coordinar el arranque de las instrucciones

Índice

La siguiente tabla muestra el contenido del DB de eje.

Nota

Los datos que no se indican en esta tabla no se pueden modificar.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
0.0	Sample_T	REAL	0.01	Tiempo de muestreo de las instrucciones de eje [s]
4.0	AxisType	BOOL	FALSE	Tipo de eje • FALSE = eje lineal • TRUE = eje rotativo
4.1	EncoderType	BOOL	FALSE	Tipo de encóder • FALSE = encóder incremental • TRUE = encóder absoluto
4.2	Sim	BOOL	FALSE	TRUE = el módulo de simulación está activo
6.0	AxisLimitMax	REAL	$1.0e^{+6}$	Fin del final de carrera por software [unidad de longitud]
10.0	AxisLimitMin	REAL	$-1.0e^{+6}$	Inicio del final de carrera por software [unidad de longitud]
14.0	MaxVelocity	REAL	0.1	Velocidad máxima del eje [unidad de longitud/s]
18.0	MaxAcceleration	REAL	0.1	Aceleración máxima del eje [unidad de longitud/s^2]
22.0	MaxDeceleration	REAL	0.1	Deceleración máxima del eje [unidad de longitud/s^2]
26.0	MonTimeTargetAppr	REAL	1.0	Tiempo de vigilancia para la aproximación a destino [s]
30.0	TargetRange	REAL	1.0	Zona de destino para el posicionamiento [unidad de longitud]

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
34.0	StandstillRange	REAL	1.5	Zona de parada sin petición de desplazamiento [unidad de longitud]
38.0	MaxFollowingDist	REAL	5.0	Error de seguimiento máximo permitido [unidad de longitud]
42.0	MonitorTargetAppr	BOOL	TRUE	TRUE = activar vigilancia de la zona de destino
42.1	SWLimitEnable	BOOL	TRUE	TRUE = activar vigilancia del final de carrera
44.0	FactorP	REAL	1.0	Factor de proporcionalidad [1/s]
48.0	ManVelocity	REAL	0.0	Velocidad de consigna en el modo manual [unidad de longitud/s]
52.0	ManEnable	BOOL	FALSE	TRUE = activar el modo manual
54.0	EmergencyDec	REAL	1000.0	Deceleración del eje para parada de emergencia [unidad de longitud/s^2]
58.0	InputModuleInAddr	INT	0	Dirección inicial de las entradas del módulo de lectura de recorrido
60.0	InputModuleOutAddr	INT	0	Dirección inicial de las salidas del módulo de lectura de recorrido
62.0	InputChannelNo	INT	0	Número de canal del módulo de lectura de recorrido
64.0	StepsPerRev	DINT	L#1	Incrementos por vuelta del encóder [incrementos/ vuelta del encóder]
68.0	DisplacementPerRev	REAL	1.0	Recorrido por vuelta del encóder [unidad de longitud/ vuelta del encóder]
72.0	NumberRevs	INT	1	Número de vueltas de un encóder absoluto
74.0	PolarityEncoder	INT	1	Corrección de sentido del encóder
80.0	OutputModuleOutAddr	INT	0	Dirección inicial de las salidas del módulo de salidas
82.0	OutputModuleInAddr	INT	0	Dirección inicial de las entradas del módulo de salidas
84.0	OutputChannelNo	INT	0	Número de canal del módulo de salidas
86.0	PolarityDrive	INT	1	Corrección de sentido del accionamiento
88.0	DriveInputAtMaxVel	REAL	9.0	Valor de referencia para velocidad máxima del eje [V]
92.0	OffsetCompensation	REAL	0.0	Compensación del punto cero [V]
96.0	DriveInputAt100	REAL	10.0	Valor de referencia para una velocidad del 100% [V]
100.0	Override	REAL	100.0	Corrección de velocidad [%]
104.0	ActPosition	REAL	0.0	Valor real de posición actual [unidad de longitud]
108.0	FollowingDistance	REAL	0.0	Error de seguimiento actual [unidad de longitud]
112.0	RemainingDistance	REAL	0.0	Distancia residual actual hasta el punto de destino [unidad de longitud]
116.0	NomVelocity	REAL	0.0	Velocidad de consigna del eje [unidad de longitud/s]
120.0	ActVelocity	REAL	0.0	Velocidad real del eje [unidad de longitud/s]
124.0	Sync	BOOL	FALSE	• FALSE = eje no sincronizado • TRUE = eje sincronizado
124.1	Error	BOOL	TRUE	• FALSE = no hay error • TRUE = error agrupado
124.2	ErrorAck	BOOL	FALSE	• FALSE = no hay acuse • TRUE = acusar errores

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
128.0	Err.SWLimitMinExceeded	BOOL	FALSE	TRUE = rebase del inicio del final de carrera por software
128.1	Err.SWLimitMaxExceeded	BOOL	FALSE	TRUE = rebase del fin del final de carrera por software
128.2	Err.TargetErr	BOOL	FALSE	TRUE = destino fuera de la zona de desplazamiento
128.3	Err.NoSync	BOOL	FALSE	TRUE = eje no sincronizado
128.4	Err.DirectionErr	BOOL	FALSE	TRUE = desplazamiento no permitido en el sentido especificado
128.5	Err.DataErr	BOOL	FALSE	TRUE = parámetro inadmisible de una instrucción de desplazamiento
128.6	Err.StartErr	BOOL	FALSE	TRUE = imposible iniciar en el estado actual del eje
128.7	Err.DistanceErr	BOOL	FALSE	TRUE = recorrido demasiado largo
129.0	Err.MasterErr	BOOL	FALSE	TRUE = eje maestro en estado inadmisible
130.0	Err.StoppedMotion	BOOL	TRUE	TRUE = eje en estado de parada de acuse obligatorio
130.1	Err.EnableDriveErr	BOOL	FALSE	TRUE = falta habilitación del accionamiento
130.2	Err.FollowingDistErr	BOOL	FALSE	TRUE = error de seguimiento máximo rebasado
130.3	Err.StandstillErr	BOOL	FALSE	TRUE = zona de parada abandonada
130.4	Err.TargetApproachErr	BOOL	FALSE	TRUE = error en la aproximación al destino
130.5	Err.EncoderErr	BOOL	FALSE	TRUE = error de encóder
130.6	Err.OutputErr	BOOL	FALSE	TRUE = error en driver de salida
130.7	Err.ConfigErr	BOOL	FALSE	TRUE = datos del eje mal parametrizados
131.0	Err.DriveErr	BOOL	FALSE	TRUE = error en accionamiento
132.0	Config.Err_AxisLimit	BOOL	FALSE	TRUE = límites de eje erróneos
132.1	Config.Err_MaxVelocity	BOOL	FALSE	TRUE = velocidad máxima errónea
132.2	Config.Err_MaxAcceleration	BOOL	FALSE	TRUE = aceleración máxima errónea
132.3	Config.Err_MaxDeceleration	BOOL	FALSE	TRUE = deceleración máxima errónea
132.4	Config.Err_MonTimeTargetApr	BOOL	FALSE	TRUE = error en tiempo de vigilancia para la aproximación a destino
132.5	Config.Err_TargetRange	BOOL	FALSE	TRUE = error en zona de destino
132.6	Config.Err_StandstillRange	BOOL	FALSE	TRUE = error en zona de parada
132.7	Config.Err_MaxFollowingDist	BOOL	FALSE	TRUE = error de seguimiento máximo incorrecto
133.0	Config.Err_EmergencyDec	BOOL	FALSE	TRUE = error en deceleración para parada brusca
133.1	Config.Err_StepsPerRev	BOOL	FALSE	TRUE = error en incrementos por vuelta del encóder
133.2	Config.Err_DisplacementPerRev	BOOL	FALSE	TRUE = error en recorrido del eje por vuelta del encóder
133.3	Config.Err_NumberRevs	BOOL	FALSE	TRUE = error en número de vueltas del encóder
133.4	Config.Err_PolarityEncoder	BOOL	FALSE	TRUE = error en corrección de sentido del encóder
133.5	Config.Err_PolarityDrive	BOOL	FALSE	TRUE = error en corrección de sentido del accionamiento
133.6	Config.Err_DriveInputAtMaxVel	BOOL	FALSE	TRUE = error en valor de referencia para velocidad máxima
133.7	Config.Err_AxisLength	BOOL	FALSE	TRUE = longitud del eje errónea
134.0	Config.Err_EncoderRange	BOOL	FALSE	TRUE = el rango del encóder no concuerda con la longitud del eje

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
134.1	Config.Err_MaxVelRotaryAxis	BOOL	FALSE	TRUE = la velocidad máxima y el tiempo de muestreo no concuerdan con la longitud del eje rotativo
134.2	Config.Err_DriveInputAt100	BOOL	FALSE	TRUE = valor de referencia para una velocidad del 100% ≤ 0
136.0	EncoderValue	DINT	L#0	Valor actual del encóder
278.0	Init	STRUCT	TRUE	Campo de bits para inicialización de 32 instrucciones

Módulos de función

Contador FM x50-1

CNT_CTRL

Descripción

La instrucción CNT_CTRL permite controlar los modos de contaje y medición.

Para aprovechar la posibilidad de modificar parámetros durante el funcionamiento o activar y desactivar las salidas del módulo de contaje, hay que emplear la instrucción CNT_CTL1 (Página 7390).

Funcionamiento

La instrucción CNT_CTRL sólo funciona en un OB no isócrono.
En un OB isócrono se deberá emplear la instrucción CNT_CTL1 (Página 7390) o CNT_CTL2 (Página 7393).

Los datos necesarios para la instrucción CNT_CTRL se guardan en el DB de contaje (Página 7393) en la CPU. La instrucción CNT_CTRL transfiere cíclicamente datos de este DB al módulo de contaje y recoge datos del módulo de contaje.

Llamada

La instrucción CNT_CTRL puede llamarse una vez por contador en el ciclo o alternativamente en un programa controlado por tiempo. No se permite realizar la llamada en un programa de interrupción controlado por eventos.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
DB_NO	INPUT	INT	Número del bloque de datos de contaje	registra	consulta
SW_GATE	INPUT	BOOL	Bit de control del contador Puerta SW (iniciar/parar)	activa y desactiva	consulta

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
GATE_STP	INPUT	BOOL	Bit de control del contador Parar puerta	activa y desactiva	consulta
OT_ERR_A	INPUT	BOOL	Acusar error de operador	activa y desactiva	consulta
OT_ERR	OUTPUT	BOOL	Hay un error de operador	consulta	activa y desactiva
L_DIRECT	IN-OUT	BOOL	Bit de inicio para carga directa de un contador	activa	consulta y desactiva
L_PREPAR	IN-OUT	BOOL	Bit de inicio para carga preparatoria de un contador	activa	consulta y desactiva
T_CMP_V1	IN-OUT	BOOL	Bit de inicio para transferir el valor de comparación 1	activa	consulta y desactiva
T_CMP_V2	IN-OUT	BOOL	Bit de inicio para transferir el valor de comparación 2	activa	consulta y desactiva
RES_SYNC	IN-OUT	BOOL	Borrar el bit de estado de sincronización	activa	consulta y desactiva
RES_ZERO	IN-OUT	BOOL	Borrar el bit de estado de paso por cero	activa	consulta y desactiva

Procesar peticiones

Las peticiones para el módulo de contaje se lanzan a través de los parámetros de instrucción L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, RES_SYNC, RES_ZERO, OT_ERR_A y GATE_STP.

Dependiendo de la petición, hay que registrar el valor de carga o un valor de comparación en el DB de contaje antes de llamar la instrucción.

Un parámetro de entrada/salida asignado (L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, RES_SYNC y RES_ZERO) es borrado por la instrucción CNT_CTRL tras finalizar la petición.

Comportamiento en arranque

En cuanto la instrucción CNT_CTRL detecta un arranque (de la CPU o del FM), se posterga cualquier petición pendiente y se acusa primero el arranque. Si ya se ha iniciado una petición, ésta se ejecutará cuando finalice el arranque, pero no se perderá.

Comportamiento en caso de error

Si se produce un error de operador al llamar la instrucción, éste se notifica en el parámetro OT_ERR. La información del error se puede leer en el DB de contaje (variable OT_ERR_B). A continuación se puede acusar el error de operador mediante el parámetro OT_ERR_A. No se notifica ningún otro error de operador hasta que no se haya acusado el error anterior.

DIAG_INF

Descripción

La instrucción DIAG_INF lee el registro DS 1 del módulo de contaje si el parámetro de inicio está activado (IN_DIAG = TRUE) y lo pone a disposición del usuario en el DB de contaje (Página 7393) (a partir de DW 54).

Funcionamiento

El registro se transfiere con la instrucción RDSYSST. El código de respuesta de dicha instrucción (RET_VAL) se copia en el parámetro RET_VAL de la instrucción DIAG_INF. En cuanto se ha ejecutado la instrucción DIAG_INF, se desactiva el parámetro de disparo IN_DIAG y se informa de que ha concluido la transferencia.

Llamada

Es posible llamar la instrucción DIAG_INF en el ciclo y en el programa de interrupción. De todas formas, no es aconsejable llamarla en el programa controlado por tiempo.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
DB_NO	INPUT	INT	Número del bloque de datos de contaje	registra	consulta
RET_VAL	OUTPUT	INT	Código de respuesta de la instrucción RDSYSST	consulta	registra
IN_DIAG	IN-OUT	BOOL	Bit de inicio para leer el registro de diagnóstico DS 1	activa y consulta	desactiva

CNT_CTL1

Descripción

La instrucción CNT_CTL1 permite controlar los modos de contaje y medición.

Además de las funciones de la instrucción CNT_CTRL, la instrucción CNT_CTL1 incluye también la posibilidad de modificar los parámetros durante el funcionamiento, además de activar y desactivar las salidas del módulo de contaje.

Funcionamiento

Los datos necesarios para la instrucción CNT_CTL1 se guardan en el DB de contaje (Página 7393) en la CPU. La instrucción CNT_CTL1 transfiere cíclicamente datos de este DB al módulo de contaje y recoge datos del módulo de contaje.

Llamada

La llamada de la instrucción CNT_CTL1 se puede efectuar cíclicamente o bien desde un OB de alarma controlado por tiempo o isócrono. No se permite realizar la llamada en un programa de interrupción controlado por eventos.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
DB_NO	INPUT	INT	Número del bloque de datos de contaje	registra	consulta
SW_GATE	INPUT	BOOL	Bit de control del contador Puerta SW (iniciar/parar)	activa y desactiva	consulta
GATE_STP	INPUT	BOOL	Bit de control del contador Parar puerta	activa y desactiva	consulta
OT_ERR_A	INPUT	BOOL	Acusar error de operador	activa y desactiva	consulta
SET_DO0	INPUT	BOOL	Activar/desactivar la salida DO0	activa y desactiva	consulta
SET_DO1	INPUT	BOOL	Activar/desactivar la salida DO1	activa y desactiva	consulta
OT_ERR	OUTPUT	BOOL	Hay un error de operador	consulta	activa y desactiva
L_DIRECT 2) 3)	IN-OUT	BOOL	**Contaje:** bit de inicio para carga directa y preparatoria de un contador	activa	consulta y desactiva
			Medición: no está permitido activarlo	-	consulta y desactiva
L_PREPAR 2) 3)	IN-OUT	BOOL	**Contaje:** bit de inicio para carga preparatoria de un contador	activa	consulta y desactiva
			Medición: transferir el límite inferior	activa	consulta y desactiva
T_CMP_V1 2)	IN-OUT	BOOL	**Contaje:** bit de inicio para transferir el valor de comparación 1	activa	consulta y desactiva
			Medición: transferir el límite superior	activa	consulta y desactiva
T_CMP_V2 2)	IN-OUT	BOOL	**Contaje:** bit de inicio para transferir el valor de comparación 2	activa	consulta y desactiva
			Medición: tiempo de actualización	activa	consulta y desactiva
C_DOPARA 1)	IN-OUT	BOOL	Bit de inicio para modificar parámetros	activa	consulta y desactiva

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
RES_SYNC	IN-OUT	BOOL	Borrar el bit de estado de sincronización	activa	consulta y desactiva
RES_ZERO	IN-OUT	BOOL	Desactivar bits de estado de paso por cero, rebase por exceso, rebase por defecto, comparador o fin de la medición	activa	consulta y desactiva

1) Este parámetro no se puede activar simultáneamente con uno de los parámetros L_DIRECT, L_PREPAR, T_CMP_V1 o T_CMP_V2.

2) Este parámetro no se puede activar simultáneamente con el parámetro C_DOPARA.

3) Estos parámetros no se pueden activar simultáneamente.

Procesar peticiones

Las peticiones para el FM x50-1 se lanzan mediante los parámetros de instrucción L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA, RES_SYNC, RES_ZERO y OT_ERR_A.

Dependiendo de la petición, los valores correspondientes (valor de carga, valores de comparación, límite inferior, límite superior, tiempo de actualización) se deben introducir en el DB de contaje antes de llamar la instrucción.

Consulte también:

- Transferencia de valores con CNT_CTL1 (Página 7400)
- Parámetros del DB para transferir valores (modos de contaje) (Página 7401)
- Parámetros del DB para transferir valores (modos de medición) (Página 7403)

Un parámetro de entrada/salida activado (L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA, RES_SYNC y RES_ZERO) vuelve a borrarse empleando la instrucción CNT_CTL1 tras completarse la petición. Ello permite reconocer que el FM x50-1 ha terminado de procesar la petición. En caso necesario, dicha información se puede integrar en el programa de usuario.

Comportamiento en arranque

En cuanto la instrucción CNT_CTL1 detecta un arranque (de la CPU o del FM), se posterga cualquier petición pendiente y se acusa primero el arranque. Si ya se ha iniciado una petición, ésta se ejecutará cuando finalice el arranque, pero no se perderá.

Comportamiento en caso de error

Si se produce un error de operador al llamar la instrucción, éste se notifica en el parámetro OT_ERR. La información del error se puede leer en el DB de contaje (variable OT_ERR_B). A continuación se puede acusar el error de operador mediante el parámetro OT_ERR_A. No se notifica ningún otro error de operador hasta que no se haya acusado el error anterior.

CNT_CTL2

Descripción

La instrucción CNT_CTL2 tiene básicamente la misma funcionalidad que la instrucción CNT_CTL1 (Página 7390).

A continuación se explican las diferencias respecto de la instrucción CNT_CTL1.

Funcionamiento

La instrucción CNT_CTL2 es especialmente apropiada para aplicaciones en las que se desea lanzar rápidamente la misma petición (p. ej. cargar valor de comparación) al FM x50-1 una y otra vez. Mientras que con la instrucción CNT_CTL1, en el mejor de los casos, puede lanzar una nueva petición cada cinco ciclos PROFIBUS DP, con la instrucción CNT_CTL2 puede hacerse cada dos ciclos PROFIBUS DP.

La instrucción está lista para una petición cuando el bit de inicio correspondiente está a 0. La finalización de una petición no se muestra por separado.

Si se produce algún problema de comunicación o bien algún error de datos o de operador, no es posible atribuirlo a una petición concreta. Por esta razón, en este caso, la instrucción detiene el procesamiento de la petición y genera un error de operador acusable con la **identificación 90**. Después de acusar el error activando el parámetro OT_ERR_A, se procesan las peticiones que hayan podido quedar pendientes.

El acuse de un error de operador se acepta correctamente cuando se desactiva el parámetro OT_ERR. Para un acuse seguro, deje activado el parámetro OT_ERR_A mientras dure la operación. Sólo deberían lanzarse más peticiones cuando el acuse sea correcto.

Nota

No está permitido iniciar simultáneamente varias transferencias de valores con la instrucción CNT_CTL2.

Llamada

La instrucción CNT_CTL2 sólo funciona en un OB isócrono.

Si se llama la instrucción CNT_CTL2 en un OB no isócrono, se produce un error de operador con la **identificación 91**. No se produce un intercambio de datos con el FM x50-1.

Otras referencias para el FM x50-1 Counter V3

Bloque de datos de contaje

Tarea

El DB de contaje es la interfaz de datos entre el programa de usuario y el FM x50-1. Contiene y aplica todos los datos necesarios para el control y el funcionamiento del módulo.

Antes de parametrizar el módulo deben asignarse los siguientes datos válidos al DB de contaje:

- Dirección del módulo (dirección 6.0)
- Dirección inicial del canal (dirección 8.0)
- Longitud de la interfaz de datos de usuario (dirección 12.0)

Configuración

El DB de contaje se divide en diferentes áreas:

DB de canal
Parámetros de instrucción, direcciones
Área de transferencia para los valores que deben escribirse
Interfaz de control
Área de transferencia para valores que deben leerse
Números de error
Interfaz de realimentación
Parámetros del FM 450-1
Datos de diagnóstico

Contenido

La tabla siguiente muestra el contenido del DB de contaje.

Nota

Los datos que no se indican en esta tabla no se pueden modificar.

Dirección	Variable	Tipo de datos	Valor inicial	Comentario	
				Contaje FM 350-1 / FM 450-1	Medición FM 350-1
Parámetros de instrucción, direcciones					
0.0	AR1_BUFFER	DWORD	DW#16#0	Búfer AR1	Búfer AR1
4.0	FP	BYTE	B#16#0	Flag Byte	Flag Byte
5.0	RESERVED	BYTE	B#16#0	Reservado	Reservado
6.0	MOD_ADR	WORD	W#16#0	Dirección del módulo	Dirección del módulo
8.0	CH_ADR	DWORD	DW#16#0	Dirección del canal	Dirección del canal
12.0	U_D_LGTH	BYTE	B#16#0	Long. datos de usuario	Long. datos de usuario
13.0	A_BYTE_0	BYTE	B#16#0	Reservado	Reservado
Área de transferencia para los valores que deben escribirse					
14.0	LOAD_VAL [1)]	DINT	L#0	Nuevo valor de carga (write user)	Límite inferior (write user)

Dirección	Variable	Tipo de datos	Valor inicial	Comentario	
				Contaje FM 350-1 / FM 450-1	Medición FM 350-1
18.0	CMP_V1 [1)]	DINT	L#0	Nuevo valor de comparación 1 (write user)	Límite superior (write user)
22.0	CMP_V2 [1)]	DINT	L#0	Nuevo valor de comparación 2 (write user)	Tiempo de actualización (write user)
Interfaz de control					
26.0	A_BIT0_0	BOOL	FALSE	Reservado	Reservado
26.1	TFB	BOOL	FALSE	Reservado / Test libre	Reservado
26.2	A_BIT0_2	BOOL	FALSE	Reservado	Reservado
26.3	A_BIT0_3	BOOL	FALSE	Reservado	Reservado
26.4	A_BIT0_4	BOOL	FALSE	Reservado	Reservado
26.5	A_BIT0_5	BOOL	FALSE	Reservado	Reservado
26.6	A_BIT0_6	BOOL	FALSE	Reservado	Reservado
26.7	A_BIT0_7	BOOL	FALSE	Reservado	Reservado
27.0	ENSET_UP [1)]	BOOL	FALSE	Habilitación para inicializar en sentido ascendente (write user)	-
27.1	ENSET_DN [1)]	BOOL	FALSE	Habilitación para inicializar en sentido descendente (write user)	-
27.2	A_BIT1_2	BOOL	FALSE	Reservado	Reservado
27.3	A_BIT1_3	BOOL	FALSE	Reservado	Reservado
27.4	A_BIT1_4	BOOL	FALSE	Reservado	Reservado
27.5	A_BIT1_5	BOOL	FALSE	Reservado	Reservado
27.6	A_BIT1_6	BOOL	FALSE	Reservado	Reservado
27.7	A_BIT1_7	BOOL	FALSE	Reservado	Reservado
28.0	CTRL_DO0 [1)]	BOOL	FALSE	Habilitación salida digital DO0 (write user)	Habilitación salida digital DO0 (write user)
28.1	CTRL_DO1 [1)]	BOOL	FALSE	Habilitación salida digital DO1 (write user)	Habilitación salida digital DO1 (write user)
28.2	A_BIT2_2	BOOL	FALSE	Reservado	Reservado
28.3	A_BIT2_3	BOOL	FALSE	Reservado	Reservado
28.4	A_BIT2_4	BOOL	FALSE	Reservado	Reservado
28.5	A_BIT2_5	BOOL	FALSE	Reservado	Reservado
28.6	A_BIT2_6	BOOL	FALSE	Reservado	Reservado
28.7	A_BIT2_7	BOOL	FALSE	Reservado	Reservado
29.0	A_BIT3_0	BOOL	FALSE	Reservado	Reservado
29.1	A_BIT3_1	BOOL	FALSE	Reservado	Reservado
29.2	A_BIT3_2	BOOL	FALSE	Reservado	Reservado
29.3	A_BIT3_3	BOOL	FALSE	Reservado	Reservado

Dirección	Variable	Tipo de datos	Valor inicial	Comentario	
				Contaje FM 350-1 / FM 450-1	Medición FM 350-1
29.4	A_BIT3_4	BOOL	FALSE	Reservado	Reservado
29.5	A_BIT3_5	BOOL	FALSE	Reservado	Reservado
29.6	A_BIT3_6	BOOL	FALSE	Reservado	Reservado
29.7	A_BIT3_7	BOOL	FALSE	Reservado	Reservado
Área de transferencia para valores que deben leerse					
30.0	LATCH_LOAD [1)]	DINT	L#0	Valor de carga o congelación actual (read user)	Valor medido actual (read user)
34.0	ACT_CNTV [1)]	DINT	L#0	Valor de contaje actual (read user)	Valor de contaje actual (read user)
Números de error					
38.0	DA_ERR_W [1)]	WORD	W#16#0	Palabra de error de datos (read user)	Palabra de error de datos (read user)
40.0	OT_ERR_B [1)]	BYTE	B#16#0	Byte de error de operador (read user)	Byte de error de operador (read user)
Interfaz de realimentación					
41.0	E_BIT0_0	BOOL	FALSE	Reservado	Reservado
41.1	STS_TFB	BOOL	FALSE	Reservado / Estado test libre	Reservado
41.2	E_BIT0_2	BOOL	FALSE	FM 350-1: Evento de diagnóstico (read user) FM 450-1: Reservado	Evento de diagnóstico (read user)
41.3	E_BIT0_3	BOOL	FALSE	Reservado	Reservado
41.4	DATA_ERR [1)]	BOOL	FALSE	Bit de error de datos (read user)	Bit de error de datos (read user)
41.5	E_BIT0_5	BOOL	FALSE	Reservado	Reservado
41.6	E_BIT0_6	BOOL	FALSE	Reservado	Reservado
41.7	PARA [1)]	BOOL	FALSE	Módulo parametrizado (read user)	Módulo parametrizado (read user)
42.0	E_BYTE_0	BYTE	B#16#0	Reservado	Reservado
43.0	STS_RUN [1)]	BOOL	FALSE	Estado contador en funcionamiento (read user) Este bit se corresponde con el bit 2^0 del contador. 0 = el LED CR está apagado 1 = el LED CR se enciende	Estado contador en funcionamiento (read user) Este bit se corresponde con el bit 2^0 del contador. 0 = el LED CR está apagado 1 = el LED CR se enciende
43.1	STS_DIR [1)]	BOOL	FALSE	Estado sentido de contaje (read user)	Estado sentido de contaje (read user)
43.2	STS_ZERO [1)]	BOOL	FALSE	Estado paso por cero (read user)	Fin de la medición (read user)

Dirección	Variable	Tipo de datos	Valor inicial	Comentario	
				Contaje FM 350-1 / FM 450-1	Medición FM 350-1
43.3	STS_OFLW [1)]	BOOL	FALSE	Estado rebase por exceso (read user)	Estado rebase por exceso (read user)
43.4	STS_UFLW [1)]	BOOL	FALSE	Estado rebase por defecto (read user)	Estado rebase por defecto (read user)
43.5	STS_SYNC [1)]	BOOL	FALSE	Estado contador sincronizado (read user)	-
43.6	STS_GATE [1)]	BOOL	FALSE	Estado puerta interna (read user)	Estado puerta interna (read user)
43.7	STS_SW_G [1)]	BOOL	FALSE	Estado puerta SW (read user)	Estado puerta SW (read user)
44.0	STS_SET [1)]	BOOL	FALSE	Estado entrada digital SET (read user)	Estado entrada digital DI-Set (read user)
44.1	STS_LATCH [1)]	BOOL	FALSE	Nuevo valor de congelación (sólo en modo isócrono) / Reservado	-
44.2	STS_STA [1)]	BOOL	FALSE	Estado entrada digital START (read user)	Estado entrada digital DI-Start (read user)
44.3	STS_STP [1)]	BOOL	FALSE	Estado entrada digital STOP (read user)	Estado entrada digital DI-Stop (read user)
44.4	STS_CMP1 [1)]	BOOL	FALSE	Estado salida valor de comparación 1 (read user)	Estado salida valor de comparación 1 (read user)
44.5	STS_CMP2 [1)]	BOOL	FALSE	Estado salida valor de comparación 2 (read user)	Estado salida valor de comparación 2 (read user)
44.6	STS_COMP1 [1)]	BOOL	FALSE	Estado almacenado del comparador 1 / Reservado	-
44.7	STS_COMP2 [1)]	BOOL	FALSE	Estado almacenado del comparador 2 / Reservado	-
45.0	E_BIT3_0	BOOL	FALSE	Reservado	Reservado
45.1	E_BIT3_1	BOOL	FALSE	Reservado	Reservado
45.2	E_BIT3_2	BOOL	FALSE	Reservado	Reservado
45.3	E_BIT3_3	BOOL	FALSE	Reservado	Reservado
45.4	E_BIT3_4	BOOL	FALSE	Reservado	Reservado
45.5	E_BIT3_5	BOOL	FALSE	Reservado	Reservado
45.6	E_BIT3_6	BOOL	FALSE	Reservado	Reservado
45.7	E_BIT3_7	BOOL	FALSE	Reservado	Reservado
Parámetros para FM 450-1					
46.0	ACT_CMP1 [1)]	DINT	L#0	Reservado / Valor de comparación actual 1 (read user)	Reservado

Dirección	Variable	Tipo de datos	Valor inicial	Comentario	
				Contaje FM 350-1 / FM 450-1	Medición FM 350-1
50.0	ACT_CMP2 [1)]	DINT	L#0	Reservado / Valor de comparación actual 2 (read user)	Reservado
Los siguientes datos de diagnóstico son introducidos por la instrucción DIAG_INF					
54.0	MDL_DEFECT	BOOL	FALSE	Fallo de módulo	Fallo de módulo
54.1	INT_FAULT	BOOL	FALSE	Error interno	Error interno
54.2	EXT_FAULT	BOOL	FALSE	Error externo	Error externo
54.3	PNT_INFO	BOOL	FALSE	Error de canal (desglose a partir de DW 58)	Error de canal (desglose a partir de DW 58)
54.4	EXT_VOLTAGE	BOOL	FALSE	Fallo tensión auxiliar	Fallo tensión auxiliar
54.5	FLD_CNNCTR	BOOL	FALSE	Conector frontal	Conector frontal
54.6	NO_CONFIG	BOOL	FALSE	Falta parametrización	Falta parametrización
54.7	CONFIG_ERR	BOOL	FALSE	Error de parametrización	Error de parametrización
55.0	MDL_TYPE	BYTE	B#16#0	Tipo de módulo	Tipo de módulo
56.0	SUB_MDL_ERR	BOOL	FALSE	Módulo de interfaz erróneo/ faltante	Módulo de interfaz erróneo/faltante
56.1	COMM_FAULT	BOOL	FALSE	Error de comunicación	Error de comunicación
56.2	MDL_STOP	BOOL	FALSE	Indicador de estado operativo RUN/STOP	Indicador de estado operativo RUN/STOP
56.3	WTCH_DOG_FAULT	BOOL	FALSE	Watchdog (FM)	Watchdog (FM)
56.4	INT_PS_FLT	BOOL	FALSE	Fallo alimentación interna	Fallo alimentación interna
56.5	PRIM_BATT_FLT	BOOL	FALSE	Vigilancia de la batería	Vigilancia de la batería
56.6	BCKUP_BATT_FLT	BOOL	FALSE	Respaldo por batería erróneo	Respaldo por batería erróneo
56.7	RESERVED_2	BOOL	FALSE	Reservado	Reservado
57.0	RACK_FLT	BOOL	FALSE	Fallo rack	Fallo rack
57.1	PROC_FLT	BOOL	FALSE	Fallo de CPU	Fallo de CPU
57.2	EPROM_FLT	BOOL	FALSE	Fallo de EPROM	Fallo de EPROM
57.3	RAM_FLT	BOOL	FALSE	Fallo de RAM	Fallo de RAM
57.4	ADU_FLT	BOOL	FALSE	Error de CAD	Error de CAD
57.5	FUSE_FLT	BOOL	FALSE	Fusible	Fusible
57.6	HW_INTR_FLT	BOOL	FALSE	Alarma de proceso perdida	Alarma de proceso perdida
57.7	RESERVED_3	BOOL	FALSE	Reservado	Reservado
58.0	CH_TYPE	BYTE	B#16#0	Tipo de canal	Tipo de canal
59.0	LGTH_DIA	BYTE	B#16#0	Longitud de datos de diagnóstico por canal	Longitud de datos de diagnóstico por canal
60.0	CH_NO	BYTE	B#16#0	Número de canal	Número de canal
61.0	GRP_ERR1	BOOL	FALSE	Error agrupado canal 1	Error agrupado canal 1
61.1	GRP_ERR2	BOOL	FALSE	Reservado /Error agrupado canal 2	No asignado en el FM 350

Dirección	Variable	Tipo de datos	Valor inicial	Comentario	
				Contaje FM 350-1 / FM 450-1	Medición FM 350-1
61.2	D_BIT7_2	BOOL	FALSE	DS1 byte 7 bit 2	DS1 byte 7 bit 2
61.3	D_BIT7_3	BOOL	FALSE	DS1 byte 7 bit 3	DS1 byte 7 bit 3
61.4	D_BIT7_4	BOOL	FALSE	DS1 byte 7 bit 4	DS1 byte 7 bit 4
61.5	D_BIT7_5	BOOL	FALSE	DS1 byte 7 bit 5	DS1 byte 7 bit 5
61.6	D_BIT7_6	BOOL	FALSE	DS1 byte 7 bit 6	DS1 byte 7 bit 6
61.7	D_BIT7_7	BOOL	FALSE	DS1 byte 7 bit 7	DS1 byte 7 bit 7
62.0	CH1_SIGA	BOOL	FALSE	Canal 1, error señal A	Canal 1, error señal A
62.1	CH1_SIGB	BOOL	FALSE	Canal 1, error señal B	Canal 1, error señal B
62.2	CH1_SIGZ	BOOL	FALSE	Canal 1, error señal cero	Canal 1, error señal cero
62.3	CH1_BETW	BOOL	FALSE	Canal 1, error entre canales	Canal 1, error entre canales
62.4	CH1_5V2	BOOL	FALSE	Canal 1, error alimentación de encóder 5,2 V	Canal 1, error alimentación de encóder 5,2 V
62.5	D_BIT8_5	BOOL	FALSE	DS1 byte 8 bit 5	DS1 byte 8 bit 5
62.6	D_BIT8_6	BOOL	FALSE	DS1 byte 8 bit 6	DS1 byte 8 bit 6
62.7	D_BIT8_7	BOOL	FALSE	DS1 byte 8 bit 7	DS1 byte 8 bit 7
63.0	D_BYTE9	BYTE	B#16#0	DS1 byte 9	DS1 byte 9
64.0	CH2_SIGA	BOOL	FALSE	Reservado / Canal 2, error señal A	Reservado
64.1	CH2_SIGB	BOOL	FALSE	Reservado / Canal 2, error señal B	Reservado
64.2	CH2_SIGZ	BOOL	FALSE	Reservado / Canal 2, error señal cero	Reservado
64.3	CH2_BETW	BOOL	FALSE	Reservado / Canal 2, error entre canales	Reservado
64.4	CH2_5V2	BOOL	FALSE	Reservado / Canal 2, error alimentación de encóder 5,2 V	Reservado
64.5	D_BIT10_5	BOOL	FALSE	DS1 byte 10 bit 5	DS1 byte 10 bit 5
64.6	D_BIT10_6	BOOL	FALSE	DS1 byte 10 bit 6	DS1 byte 10 bit 6
64.7	D_BIT10_7	BOOL	FALSE	DS1 byte 10 bit 7	DS1 byte 10 bit 7
65.0	D_BYTE11	BYTE	B#16#0	DS1 byte 11	DS1 byte 11
66.0	D_BYTE12	BYTE	B#16#0	DS1 byte 12	DS1 byte 12
67.0	D_BYTE13	BYTE	B#16#0	DS1 byte 13	DS1 byte 13
68.0	D_BYTE14	BYTE	B#16#0	DS1 byte 14	DS1 byte 14
69.0	D_BYTE15	BYTE	B#16#0	DS1 byte 15	DS1 byte 15
[1] Variables en el DB que se pueden/tienen que introducir o leer al trabajar con el FM x50-1					

Transferencia de valores con CNT_CTL1

Transferencia de valores

Dependiendo del modo de operación, los valores se transfieren activando estos parámetros de instrucción:

Modo de operación	Parámetro de instrucción
Contaje	L_DIRECT, L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA
Medición	L_PREPAR, T_CMP_V1, T_CMP_V2, C_DOPARA

Antes de transferirse, los valores deben guardarse en el DB.

Consulte también:

- Parámetros del DB para transferir valores (modos de contaje) (Página 7401)
- Parámetros del DB para transferir valores (modos de medición) (Página 7403)

Es posible transferir simultáneamente varios valores:

En el modo de operación ...	... pueden transferirse simultáneamente	
Contaje	Valor de carga	(Parámetro del DB LOAD_VAL)
	Valor de comparación 1	(Parámetro del DB CMP_V1)
	Valor de comparación 2	(Parámetro del DB CMP_V2)
Medición	Límite inferior	(Parámetro del DB LOAD_VAL)
	Límite superior	(Parámetro del DB CMP_V1)
	Tiempo de actualización	(Parámetro del DB CMP_V2)

Si un valor es erróneo, hay que acusar primero este error de operador con OT_ERR_A para que el FM 350-1 pueda aplicar los demás valores. Corrija luego el valor rechazado y transfiéralo de nuevo.

Nota

Si con los parámetros de instrucción L_DIRECT, L_PREPAR, T_CMP_V1 o T_CMP_V2 se cargan los valores LOAD_VAL, CMP_V1 o CMP_V2, no está permitido modificar simultáneamente la parametrización con el parámetro de instrucción C_DOPARA.

Ello ocasionaría un error de operador OT_ERR que se debe acusar con OT_ERR_A.

Tiempo necesario para transferir valores

Consulte el tiempo necesario para transferir valores en la tabla siguiente:

Aplicación del FM 350-1		Tiempo necesario
Centralizada		Como mínimo 4 ciclos del OB 1
Descentralizada (modo no isócrono)		Como mínimo 5 ciclos PROFIBUS DP
Descentralizada (modo isócrono)		
• Al transferir sólo un valor		• 5 ciclos PROFIBUS DP
• Al lanzar simultáneamente la transferencia de varios valores		
	• Para el 1er. valor: • Para el 2º valor: • Para el 3er. valor:	• 5 ciclos PROFIBUS DP después del disparo • 6 ciclos PROFIBUS DP después del disparo • 7 ciclos PROFIBUS DP después del disparo

Parámetros del DB para transferir valores (modos de contaje)

Preparación

Antes de llamar las instrucciones CNT_CTRL (Página 7388), CNT_CTL1 (Página 7390) o CNT_CTL2 (Página 7393) hay que asignar los siguientes datos válidos al bloque de datos:

- Dirección del módulo (dirección DB 6.0)
- Dirección inicial del canal (dirección DB 8.0)
- Longitud de la interfaz de datos de usuario (dirección DB 12.0)

Estos datos deberían registrarse en el DB con cada arranque del OB 100.

Parámetros del DB para transferir valores (modos de contaje)

La siguiente tabla muestra la sección del DB a la que se transfieren los parámetros LOAD_VAL, CMP_V1 y CMP_V2.

El parámetro LOAD_VAL (bytes de 14 a 17) tiene dos significados:

- Si se activa el parámetro de instrucción L_DIRECT o L_PREPAR, LOAD_VAL se interpreta como valor de carga.
- Si activa el parámetro de instrucción C_DOPARA, es posible determinar en el byte 14 el comportamiento de las salidas DO0 y DO1. Los bytes 15 y 16 se interpretan como histéresis y duración de impulso.

Sólo es posible activar uno de los parámetros de instrucción L_DIRECT, L_PREPAR y C_DOPARA a la vez.

Dirección del DB	Parámetro	Tipo de datos	Significado				
14.0	LOAD_VAL	DINT	Valor de carga; carga directa y preparatoria con el parámetro de instrucción L_DIRECT Valor de carga; carga preparatoria con el parámetro de instrucción L_PREPAR				
			Comportamiento de las salidas DO0 y DO1, de la histéresis y de la duración de impulso; definir con el parámetro de instrucción C_DOPARA:				
			Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Comportamiento de la salida DO0**
			x	0	0	0	Inactiva
			x	0	0	1	Activo desde el valor de comparación hasta el límite superior
			x	0	1	0	Activo desde valor de comparación hasta el límite inferior
			x	0	1	1	Activa al alcanzarse el valor de comparación por la duración del impulso hacia delante/atrás
			x	1	0	0	Activa al alcanzarse el valor de comparación por la duración del impulso hacia delante
			x	1	0	1	Activa al alcanzarse el valor de comparación por la duración del impulso hacia atrás
			Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Comportamiento de la salida DO1**
			x	0	0	0	Inactivo
			x	0	0	1	Activo desde el valor de comparación hasta el límite superior
			x	0	1	0	Activo desde valor de comparación hasta el límite inferior
			x	0	1	1	Activo al alcanzar el valor de comparación durante la duración del impulso hacia delante/atrás
			x	1	0	0	Activo al alcanzar el valor de comparación en sentido ascendente durante la duración del impulso
			x	1	0	1	Activo al alcanzarse el valor de comparación en sentido descendente durante la duración del impulso
			x	1	1	0	Conmutación en los valores de comparación
15.0			Histéresis (rango de valores 0...255)				
16.0			Duración del impulso (rango de valores 0...250)				
17.0			Reservado = 0				
18.0	CMP_V1	DINT	Valor de comparación 1; cargar con el parámetro de instrucción T_CMP_V1				
22.0	CMP_V2	DINT	Valor de comparación 2; cargar con el parámetro de instrucción T_CMP_V2				
x = irrelevante							

Parámetros del DB para transferir valores (modos de medición)

Preparación

Antes de llamar las instrucciones CNT_CTRL (Página 7388), CNT_CTL1 (Página 7390) o CNT_CTL2 (Página 7393) hay que asignar los siguientes datos válidos al bloque de datos:

- Dirección del módulo (dirección DB 6.0)
- Dirección inicial del canal (dirección DB 8.0)
- Longitud de la interfaz de datos de usuario (dirección DB 12.0)

Estos datos deberían registrarse en el DB con cada arranque del OB 100.

Parámetros del DB para transferir valores (modos de medición)

La siguiente tabla muestra la sección del DB a la que se transfieren los parámetros LOAD_VAL, CMP_V1 y CMP_V2.

El parámetro LOAD_VAL (bytes de 14 a 17) tiene dos significados:

- Si se activa el parámetro de instrucción L_PREPAR, LOAD_VAL se interpreta como valor límite inferior.
- Si activa el parámetro de la instrucción C_DOPARA, es posible determinar en el byte 14 el comportamiento de la salida DO0.

Sólo es posible activar uno de los parámetros de instrucción L_PREPAR y C_DOPARA a la vez.

Dirección del DB	Parámetros	Tipo de datos	Significado			
14.0	LOAD_VAL	DINT	Valor límite inferior; cargar con el parámetro de instrucción L_PREPAR			
			Comportamiento de DO0; determinar con el parámetro de instrucción C_DOPARA			
			Bits 2 a 7	**Bit 1**	**Bit 0**	**Comportamiento de la salida DO0**
			Irrelevante	0	0	Sin comparación
			irrelevante	0	1	Activo fuera de los valores límite
			irrelevante	1	0	Activo cuando se rebasa por defecto el valor límite inferior
			irrelevante	1	1	Activo cuando se rebasa el valor límite superior
15.0			reserva = 0			
16.0			reserva = 0			
17.0			reserva = 0			
18.0	CMP_V1	DINT	Valor límite superior; cargar con el parámetro de instrucción T_CMP_V1			
22.0	CMP_V2	DINT	Tiempo de actualización; cargar con el parámetro de instrucción T_CMP_V2			

Contador FM 350-2

CNT2WRPN

Descripción

La instrucción CNT2WRPN carga los contadores y comparadores del módulo de contaje mediante peticiones de escritura. Para ello, en caso necesario debe llamarse la instrucción CNT2WRPN individualmente para cada módulo.

La instrucción CNT2WRPN debe integrarse en el programa de usuario únicamente si hay que volver a cargar los contadores y comparadores del módulo de contaje durante el funcionamiento.

Funcionamiento

La instrucción CNT2WRPN ejecuta la petición de escritura (Página 7415) (JOB_WR) desde el DB de contaje (Página 7407), transmite los datos correspondientes desde el DB de contaje e indica el estado de la petición de escritura (Página 7417). Para la petición de escritura se emplea de manera interna la instrucción WRREC y se transmite el código de respuesta de la instrucción WRREC.

Llamada

La instrucción CNT2WRPN puede llamarse de forma cíclica o bien dentro de un programa controlado por tiempo. No se permite realizar la llamada en un programa de interrupción controlado por eventos.

Antes de procesar peticiones de escritura debe proporcionar los valores necesarios al área de datos correspondiente a la petición en cuestión. La última petición de escritura tiene que haber finalizado, es decir, el JOB_WR.NO (byte de datos DBB0) del DB de contaje (Página 7407) está borrado.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo ...
DB_NO	INPUT	WORD	Número del bloque de datos de contaje	registra	consulta
RET_VAL	OUTPUT	INT	Código de respuesta de la instrucción WRREC	consulta	registra

CNT2RDPN

Descripción

La instrucción CNT2RDPN lee los valores de contaje y valores medidos del módulo de contaje con peticiones de lectura. Para ello se debe llamar la instrucción CNT2RDPN de forma cíclica una vez por cada módulo.

La instrucción CNT2RDPN no debe incorporarse al programa de usuario cuando no se procesan peticiones de lectura.

Funcionamiento

La instrucción ejecuta la petición de lectura (Página 7418) (JOB_RD) desde el DB de contaje (Página 7407), transmite los datos correspondientes al DB de contaje e indica el estado de la petición de lectura (Página 7418). Para la petición de lectura se emplea de manera interna la instrucción RDREC y se transmite el código de respuesta de la instrucción RDREC.

Llamada

La instrucción CNT2RDPN puede llamarse de forma cíclica o bien dentro de un programa controlado por tiempo. No se permite realizar la llamada en un programa de interrupción controlado por eventos.

La última petición de lectura tiene que haber finalizado, es decir, el JOB_RD.NO (byte de datos DBB2) del DB de contaje (Página 7407) está borrado.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo ...
DB_NO	INPUT	WORD	Número del bloque de datos de contaje	registra	consulta
RET_VAL	OUTPUT	INT	Código de respuesta de la instrucción RDREC	consulta	registra

CNT2_CTR

Descripción

La instrucción CNT2_CTR controla las salidas digitales (habilitar y bloquear) y las puertas software del módulo de contaje. Además, recibe las respuestas del módulo.

Funcionamiento

La instrucción CNT2_CTR

- inicializa el DB de contaje (Página 7407).
- transmite las señales de control del DB de contaje (estructura CONTROL_SIGNALS (Página 7414)) al módulo.
- lee las señales de respuesta. La instrucción guarda los valores leídos en la estructura CHECKBACK_SIGNALS (Página 7414) del DB de contaje.

Llamada

Debe llamar la instrucción CNT2_CTR cíclicamente (en el OB 1 ó en las alarmas cíclicas) para cada módulo. No se permite realizar la llamada en un programa de interrupción.

Antes de la llamada de la instrucción CNT2_CTR, introduzca las señales de control actuales en la estructura CONTROL_SIGNALS en el DB de contaje (Página 7407). Después de la llamada de la instrucción CNT2_CTR, las señales de respuesta de la estructura CHECKBACK_SIGNALS están actualizadas en el DB de contaje, desde donde puede procesarlas.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
DB_NO	INPUT	WORD	Número del bloque de datos de contaje	registra	consulta

DIAG_RD

Descripción

La instrucción DIAG_RD carga los datos de la alarma de diagnóstico en el DB de contaje cuando aparece una alarma de diagnóstico.

Funcionamiento

La instrucción DIAG_RD lee 16 bytes de datos de diagnóstico del módulo de contaje y los transfiere al DB de contaje (Página 7407) al área de datos DIAGNOSTIC_IN_INFO. Estos datos contienen el estado de diagnóstico de todo el módulo (para todos los contadores). Los datos de diagnóstico se leen internamente con la instrucción RDSYSST. El código de respuesta de esta instrucción se transmite como RET_VAL.

Llamada

La instrucción DIAG_RD sólo se puede llamar en el programa de interrupción.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción	El usuario lo ...	La instrucción lo …
DB_NO	INPUT	WORD	Número del bloque de datos de contaje	registra	consulta
RET_VAL	OUTPUT	INT	Código de respuesta de la instrucción RDSYSST	consulta	registra

Otras referencias para el FM 350-2 Counter V2

Bloque de datos de contaje

Tarea

El DB de contaje es la interfaz de datos entre el programa de usuario y el FM 350-2. Contiene y aplica todos los datos necesarios para el control y el funcionamiento del módulo.

Antes de parametrizar el módulo deben asignarse los siguientes datos válidos al DB de contaje:

- Dirección del módulo (dirección 12.0)
- Dirección inicial del canal (dirección 14.0)
- DS-OFFSET (dirección 18.0), está fijada en 0

Contenido

La tabla siguiente muestra el contenido del DB de contaje.

Nota

Los datos que no se indican en esta tabla no se pueden modificar.

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
0.0	NO	BYTE	B#16#0	Número
1.0	BUSY	BOOL	FALSE	TRUE: petición de escritura en curso FALSE: petición de escritura no en curso
1.1	DONE	BOOL	FALSE	TRUE: petición de escritura finalizada FALSE: petición de escritura no finalizada
1.2	IMPOSS	BOOL	FALSE	TRUE: petición de escritura no posible FALSE: petición de escritura posible
1.3	UNKNOWN	BOOL	FALSE	TRUE: petición de escritura desconocida FALSE: petición de escritura conocida
2.0	NO	BYTE	B#16#0	Número

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
3.0	BUSY	BOOL	FALSE	TRUE: petición de lectura en curso FALSE: petición de lectura no en curso
3.1	DONE	BOOL	FALSE	TRUE: petición de lectura finalizada FALSE: petición de lectura no finalizada
3.2	IMPOSS	BOOL	FALSE	TRUE: petición de escritura no posible FALSE: petición de lectura posible
3.3	UNKNOWN	BOOL	FALSE	TRUE: petición de lectura desconocida FALSE: petición de lectura conocida
4.0	RESERV_0	ARRAY [1..3] OF WORD	W#16#0	Reservado
10.0	RESERV_1	WORD	W#16#0	Reservado
12.0	MOD_ADR	WORD	W#16#0	Dirección del módulo
14.0	CH_ADR	DWORD	DW#16#0	Dirección del canal
18.0	DS_OFFS	BYTE	B#16#0	Offset bloque de datos
19.0	RESERV_2	BYTE	B#16#0	Reservado
20.0	BIT0_0	BOOL	FALSE	Reservado
20.1	BIT0_1	BOOL	FALSE	Reservado
20.2	BIT0_2	BOOL	FALSE	Reservado
20.3	BIT0_3	BOOL	FALSE	Reservado
20.4	BIT0_4	BOOL	FALSE	Reservado
20.5	BIT0_5	BOOL	FALSE	Reservado
20.6	BIT0_6	BOOL	FALSE	Reservado
20.7	BIT0_7	BOOL	FALSE	Reservado
21.0	CTRL_DQ0	BOOL	FALSE	TRUE: Salida 0 habilitada FALSE: Salida 0 no habilitada
21.1	CTRL_DQ1	BOOL	FALSE	TRUE: Salida 1 habilitada FALSE: Salida 1 no habilitada
21.2	CTRL_DQ2	BOOL	FALSE	TRUE: Salida 2 habilitada FALSE: Salida 2 no habilitada
21.3	CTRL_DQ3	BOOL	FALSE	TRUE: Salida 3 habilitada FALSE: Salida 3 no habilitada
21.4	CTRL_DQ4	BOOL	FALSE	TRUE: Salida 4 habilitada FALSE: Salida 4 no habilitada
21.5	CTRL_DQ5	BOOL	FALSE	TRUE: Salida 5 habilitada FALSE: Salida 5 no habilitada
21.6	CTRL_DQ6	BOOL	FALSE	TRUE: Salida 6 habilitada FALSE: Salida 6 no habilitada
21.7	CTRL_DQ7	BOOL	FALSE	TRUE: Salida 7 habilitada FALSE: Salida 7 no habilitada
22.0	SET_DQ0	BOOL	FALSE	TRUE: Salida 0 activada FALSE: Salida 0 no activada

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
22.1	SET_DQ1	BOOL	FALSE	TRUE: Salida 1 activada FALSE: Salida 1 no activada
22.2	SET_DQ2	BOOL	FALSE	TRUE: Salida 2 activada FALSE: Salida 2 no activada
22.3	SET_DQ3	BOOL	FALSE	TRUE: Salida 3 activada FALSE: Salida 3 no activada
22.4	SET_DQ4	BOOL	FALSE	TRUE: Salida 4 activada FALSE: Salida 4 no activada
22.5	SET_DQ5	BOOL	FALSE	TRUE: Salida 5 activada FALSE: Salida 5 no activada
22.6	SET_DQ6	BOOL	FALSE	TRUE: Salida 6 activada FALSE: Salida 6 no activada
22.7	SET_DQ7	BOOL	FALSE	TRUE: Salida 7 activada FALSE: Salida 7 no activada
23.0	SW_GATE0	BOOL	FALSE	TRUE: Puerta SW contador 0 abierta FALSE: Puerta SW contador 0 cerrada
23.1	SW_GATE1	BOOL	FALSE	TRUE: Puerta SW contador 1 abierta FALSE: Puerta SW contador 1 cerrada
23.2	SW_GATE2	BOOL	FALSE	TRUE: Puerta SW contador 2 abierta FALSE: Puerta SW contador 2 cerrada
23.3	SW_GATE3	BOOL	FALSE	TRUE: Puerta SW contador 3 abierta FALSE: Puerta SW contador 3 cerrada
23.4	SW_GATE4	BOOL	FALSE	TRUE: Puerta SW contador 4 abierta FALSE: Puerta SW contador 4 cerrada
23.5	SW_GATE5	BOOL	FALSE	TRUE: Puerta SW contador 5 abierta FALSE: Puerta SW contador 5 cerrada
23.6	SW_GATE6	BOOL	FALSE	TRUE: Puerta SW contador 6 abierta FALSE: Puerta SW contador 6 cerrada
23.7	SW_GATE7	BOOL	FALSE	TRUE: Puerta SW contador 7 abierta FALSE: Puerta SW contador 7 cerrado
24.0	CTRL_DWORD1	DWORD	DW#16#0	Reservado
28.0	CTRL_DWORD2	DWORD	DW#16#0	Reservado
32.0	CTRL_DWORD3	DWORD	DW#16#0	Reservado
36.0	BIT0_0	BOOL	FALSE	Reservado
36.1	STS_TFB	BOOL	FALSE	TRUE: Modo PG activo FALSE: Ningún modo PG
36.2	BIT0_2	BOOL	FALSE	Reservado
36.3	BIT0_3	BOOL	FALSE	Reservado
36.4	DATA_ERR	BOOL	FALSE	Error de datos
36.5	BIT0_5	BOOL	FALSE	Reservado
36.6	BIT0_6	BOOL	FALSE	Reservado

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
36.7	PARA	BOOL	FALSE	TRUE: Módulo parametrizado FALSE: Módulo no parametrizado
37.0	STS_CMP0	BOOL	FALSE	TRUE: Comparador 0 activado FALSE: Comparador 0 no activado
37.1	STS_CMP1	BOOL	FALSE	TRUE: Comparador 1 activado FALSE: Comparador 1 no activado
37.2	STS_CMP2	BOOL	FALSE	TRUE: Comparador 2 activado FALSE: Comparador 2 no activado
37.3	STS_CMP3	BOOL	FALSE	TRUE: Comparador 3 activado FALSE: Comparador 3 no activado
37.4	STS_CMP4	BOOL	FALSE	TRUE: Comparador 4 activado FALSE: Comparador 4 no activado
37.5	STS_CMP5	BOOL	FALSE	TRUE: Comparador 5 activado FALSE: Comparador 5 no activado
37.6	STS_CMP6	BOOL	FALSE	TRUE: Comparador 6 activado FALSE: Comparador 6 no activado
37.7	STS_CMP7	BOOL	FALSE	TRUE: Comparador 7 activado FALSE: Comparador 7 no activado
38.0	STS_UFLW0	BOOL	FALSE	TRUE: Rebase por defecto contador 0 FALSE: No hay rebase por defecto contador 0
38.1	STS_UFLW1	BOOL	FALSE	TRUE: Rebase por defecto contador 1 FALSE: No hay rebase por defecto contador 1
38.2	STS_UFLW2	BOOL	FALSE	TRUE: Rebase por defecto contador 2 FALSE: No hay rebase por defecto contador 2
38.3	STS_UFLW3	BOOL	FALSE	TRUE: Rebase por defecto contador 3 FALSE: No hay rebase por defecto contador 3
38.4	STS_UFLW4	BOOL	FALSE	TRUE: Rebase por defecto contador 4 FALSE: No hay rebase por defecto contador 4
38.5	STS_UFLW5	BOOL	FALSE	TRUE: Rebase por defecto contador 5 FALSE: No hay rebase por defecto contador 5
38.6	STS_UFLW6	BOOL	FALSE	TRUE: Rebase por defecto contador 6 FALSE: No hay rebase por defecto contador 6
38.7	STS_UFLW7	BOOL	FALSE	TRUE: Rebase por defecto contador 7 FALSE: No hay rebase por defecto contador 7
39.0	STS_OFLW0	BOOL	FALSE	TRUE: Rebase por exceso contador 0 FALSE: No hay rebase por exceso contador 0
39.1	STS_OFLW1	BOOL	FALSE	TRUE: Rebase por exceso contador 1 FALSE: No hay rebase por exceso contador 1
39.2	STS_OFLW2	BOOL	FALSE	TRUE: Rebase por exceso contador 2 FALSE: No hay rebase por exceso contador 2
39.3	STS_OFLW3	BOOL	FALSE	TRUE: Rebase por exceso contador 3 FALSE: No hay rebase por exceso contador 3

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
39.4	STS_OFLW4	BOOL	FALSE	TRUE: Rebase por exceso contador 4 FALSE: No hay rebase por exceso contador 4
39.5	STS_OFLW5	BOOL	FALSE	TRUE: Rebase por exceso contador 5 FALSE: No hay rebase por exceso contador 5
39.6	STS_OFLW6	BOOL	FALSE	TRUE: Rebase por exceso contador 6 FALSE: No hay rebase por exceso contador 6
39.7	STS_OFLW7	BOOL	FALSE	TRUE: Rebase por exceso contador 7 FALSE: No hay rebase por exceso contador 7
40.0	STS_DIR0	BOOL	FALSE	TRUE: Sentido de contaje contador 0 atrás FALSE: Sentido de contaje contador 0 adelante
40.1	STS_DIR1	BOOL	FALSE	TRUE: Sentido de contaje contador 1 atrás FALSE: Sentido de contaje contador 1 adelante
40.2	STS_DIR2	BOOL	FALSE	TRUE: Sentido de contaje contador 2 atrás FALSE: Sentido de contaje contador 2 adelante
40.3	STS_DIR3	BOOL	FALSE	TRUE: Sentido de contaje contador 3 atrás FALSE: Sentido de contaje contador 3 adelante
40.4	STS_DIR4	BOOL	FALSE	TRUE: Sentido de contaje contador 4 atrás FALSE: Sentido de contaje contador 4 adelante
40.5	STS_DIR5	BOOL	FALSE	TRUE: Sentido de contaje contador 5 atrás FALSE: Sentido de contaje contador 5 adelante
40.6	STS_DIR6	BOOL	FALSE	TRUE: Sentido de contaje contador 6 atrás FALSE: Sentido de contaje contador 6 adelante
40.7	STS_DIR7	BOOL	FALSE	TRUE: Sentido de contaje contador 7 atrás FALSE: Sentido de contaje contador 7 adelante
41.0	STS_DI0	BOOL	FALSE	TRUE: Entrada digital 0 activada FALSE: Entrada digital 0 no activada
41.1	STS_DI1	BOOL	FALSE	TRUE: Entrada digital 1 activada FALSE: Entrada digital 1 no activada
41.2	STS_DI2	BOOL	FALSE	TRUE: Entrada digital 2 activada FALSE: Entrada digital 2 no activada
41.3	STS_DI3	BOOL	FALSE	TRUE: Entrada digital 3 activada FALSE: Entrada digital 3 no activada
41.4	STS_DI4	BOOL	FALSE	TRUE: Entrada digital 4 activada FALSE: Entrada digital 4 no activada
41.5	STS_DI5	BOOL	FALSE	TRUE: Entrada digital 5 activada FALSE: Entrada digital 5 no activada
41.6	STS_DI6	BOOL	FALSE	TRUE: Entrada digital 6 activada FALSE: Entrada digital 6 no activada
41.7	STS_DI7	BOOL	FALSE	TRUE: Entrada digital 7 activada FALSE: Entrada digital 7 no activada
42.0	STS_DQ0	BOOL	FALSE	TRUE: Salida digital 0 activada FALSE: Salida digital 0 no activada

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
42.1	STS_DQ1	BOOL	FALSE	TRUE: Salida digital 1 activada FALSE: Salida digital 1 no activada
42.2	STS_DQ2	BOOL	FALSE	TRUE: Salida digital 2 activada FALSE: Salida digital 2 no activada
42.3	STS_DQ3	BOOL	FALSE	TRUE: Salida digital 3 activada FALSE: Salida digital 3 no activada
42.4	STS_DQ4	BOOL	FALSE	TRUE: Salida digital 4 activada FALSE: Salida digital 4 no activada
42.5	STS_DQ5	BOOL	FALSE	TRUE: Salida digital 5 activada FALSE: Salida digital 5 no activada
42.6	STS_DQ6	BOOL	FALSE	TRUE: Salida digital 6 activada FALSE: Salida digital 6 no activada
42.7	STS_DQ7	BOOL	FALSE	TRUE: Salida digital 7 activada FALSE: Salida digital 7 no activada
43.0	STS_GATE0	BOOL	FALSE	TRUE: Puerta interna contador 0 abierta FALSE: Puerta interna contador 0 cerrada
43.1	STS_GATE1	BOOL	FALSE	TRUE: Puerta interna contador 1 abierta FALSE: Puerta interna contador 1 cerrada
43.2	STS_GATE2	BOOL	FALSE	TRUE: Puerta interna contador 2 abierta FALSE: Puerta interna contador 2 cerrada
43.3	STS_GATE3	BOOL	FALSE	TRUE: Puerta interna contador 3 abierta FALSE: Puerta interna contador 3 cerrada
43.4	STS_GATE4	BOOL	FALSE	TRUE: Puerta interna contador 4 abierta FALSE: Puerta interna contador 4 cerrada
43.5	STS_GATE5	BOOL	FALSE	TRUE: Puerta interna contador 5 abierta FALSE: Puerta interna contador 5 cerrada
43.6	STS_GATE6	BOOL	FALSE	TRUE: Puerta interna contador 6 abierta FALSE: Puerta interna contador 6 cerrada
43.7	STS_GATE7	BOOL	FALSE	TRUE: Puerta interna contador 7 abierta FALSE: Puerta interna contador 7 cerrada
44	USER_STAT_WORD0	WORD	W#16#0	Según parametrización valores de contaje / medición
46	USER_STAT_WORD1	WORD	W#16#0	Según parametrización valores de contaje / medición
48	USER_STAT_WORD2	WORD	W#16#0	Según parametrización valores de contaje / medición
50	USER_STAT_WORD3	WORD	W#16#0	Según parametrización valores de contaje / medición
52	LOAD-VAL0	DINT	L#0	Cargar directamente contador 0
56	LOAD-VAL1	DINT	L#0	Cargar directamente contador 1
60	LOAD-VAL2	DINT	L#0	Cargar directamente contador 2
64	LOAD-VAL3	DINT	L#0	Cargar directamente contador 3
68	LOAD-VAL4	DINT	L#0	Cargar directamente contador 4

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
72	LOAD-VAL5	DINT	L#0	Cargar directamente contador 5
76	LOAD-VAL6	DINT	L#0	Cargar directamente contador 6
80	LOAD-VAL7	DINT	L#0	Cargar directamente contador 7
84	LOAD-PREPARE-VAL0	DINT	L#0	Carga preliminar del contador 0
88	LOAD-PREPARE-VAL1	DINT	L#0	Carga preliminar del contador 1
92	LOAD-PREPARE-VAL2	DINT	L#0	Carga preliminar del contador 2
96	LOAD-PREPARE-VAL3	DINT	L#0	Carga preliminar del contador 3
100	LOAD-PREPARE-VAL4	DINT	L#0	Carga preliminar del contador 4
104	LOAD-PREPARE-VAL5	DINT	L#0	Carga preliminar del contador 5
108	LOAD-PREPARE-VAL6	DINT	L#0	Carga preliminar del contador 6
112	LOAD-PREPARE-VAL7	DINT	L#0	Carga preliminar del contador 7
116	CMP-VAL0	DINT	L#0	Cargar comparador 0
120	CMP-VAL1	DINT	L#0	Cargar comparador 1
124	CMP-VAL2	DINT	L#0	Cargar comparador 2
128	CMP-VAL3	DINT	L#0	Cargar comparador 3
132	CMP-VAL4	DINT	L#0	Cargar comparador 4
136	CMP-VAL5	DINT	L#0	Cargar comparador 5
140	CMP-VAL6	DINT	L#0	Cargar comparador 6
144	CMP-VAL7	DINT	L#0	Cargar comparador 7
148	ACT_CNTV0	DINT	L#0	Estado actual contador 0
152	ACT_MSRV0	DINT	L#0	Resultado de medición 0
156	ACT_CNTV1	DINT	L#0	Estado actual contador 1
160	ACT_MSRV1	DINT	L#0	Resultado de medición 1
164	ACT_CNTV2	DINT	L#0	Estado actual contador 2
168	ACT_MSRV2	DINT	L#0	Resultado de medición 2
172	ACT_CNTV3	DINT	L#0	Estado actual contador 3
176	ACT_MSRV3	DINT	L#0	Resultado de medición 3
180	ACT_CNTV4	DINT	L#0	Estado actual contador 4
184	ACT_MSRV4	DINT	L#0	Resultado de medición 4
188	ACT_CNTV5	DINT	L#0	Estado actual contador 5
192	ACT_MSRV5	DINT	L#0	Resultado de medición 5
196	ACT_CNTV6	DINT	L#0	Estado actual contador 6
200	ACT_MSRV6	DINT	L#0	Resultado de medición 6
204	ACT_CNTV7	DINT	L#0	Estado actual contador 7
208	ACT_MSRV7	DINT	L#0	Resultado de medición 7
212.0	BYTE0	BYTE	B#16#0	Reservado
213.0	BYTE1	BYTE	B#16#0	Reservado
214.0	BYTE2	BYTE	B#16#0	Reservado
215.0	BYTE3	BYTE	B#16#0	Reservado
216.0	BYTE4	BYTE	B#16#0	Tipo de canal
217.0	BYTE5	BYTE	B#16#0	Longitud información de canal
218.0	BYTE6	BYTE	B#16#0	Número de canales

Dirección	Variable	Tipo de datos	Valor inicial	Comentario
219.0	BYTE7	BYTE	B#16#0	Vector de error de canal
220.0	BYTE8	BYTE	B#16#0	Error contador 0
221.0	BYTE9	BYTE	B#16#0	Error contador 1
222.0	BYTE10	BYTE	B#16#0	Error contador 2
223.0	BYTE11	BYTE	B#16#0	Error contador 3
224.0	BYTE12	BYTE	B#16#0	Error contador 4
225.0	BYTE13	BYTE	B#16#0	Error contador 5
226.0	BYTE14	BYTE	B#16#0	Error contador 6
227.0	BYTE15	BYTE	B#16#0	Error contador 7

CONTROL_SIGNALS

CONTROL_SIGNALS

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
21.0 … 7	CTRL_DQ0 … 7	BOOL	FALSE	Habilitar las salidas digitales 0 … 7
22.0 … 7	SET_DQ0 … 7	BOOL	FALSE	Controlar las salidas digitales 0 … 7
23.0 … 7	SW_GATE0 … 7	BOOL	FALSE	Puerta software del contador 0 … 7

CHECKBACK_SIGNALS

CHECKBACK_SIGNALS

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
36.1	STS_TFB	BOOL	FALSE	TRUE: El modo de programadora está activo, imposible controlar desde la CPU FALSE: Modo de programadora deseleccionado
36.4	DATA_ERR	BOOL	FALSE	TRUE: Error de datos FALSE: Sin errores de datos
36.7	PARA	BOOL	FALSE	TRUE: El módulo de contaje está parametrizado, todas las CHECKBACK_SIGNALS restantes son válidas FALSE: Módulo de contaje no parametrizado
37.0 … 7	STS_CMP0 … 7	BOOL	FALSE	Estado del comparador 0 … 7 Debe evaluar el estado después de la llamada de la instrucción CNT2_CTR, ya que después se desactiva
38.0 … 7	STS_UFLW0 … 7	BOOL	FALSE	Estado de rebase por defecto del contador 0 … 7 Debe evaluar el estado después de la llamada de la instrucción CNT2_CTR, ya que después se desactiva

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
39.0 … 7	STS_OFLW0 … 7	BOOL	FALSE	Estado de rebase por exceso del contador 0 … 7 Debe evaluar el estado después de la llamada de la instrucción CNT2_CTR, ya que después se desactiva
40.0 … 7	STS_DIR0 … 7	BOOL	FALSE	Estado del sentido de contaje del contador 0 … 7 Debe evaluar el estado después de la llamada de la instrucción CNT2_CTR, ya que después se desactiva
41.0 … 7	STS_DI0 … 7	BOOL	FALSE	Estado de la entrada digital 0 … 7
42.0 … 7	STS_DQ0 … 7	BOOL	FALSE	Estado de la salida digital 0 … 7
43.0 … 7	STS_GATE0 … 7	BOOL	FALSE	Estado de la puerta interna del contador 0 … 7
44	USER_STAT_DWORD0	WORD	W#16#0	Según parametrización del valor de contaje / valor medido
46	USER_STAT_DWORD1	WORD	W#16#0	Según parametrización del valor de contaje / valor medido
48	USER_STAT_DWORD2	WORD	W#16#0	Según parametrización del valor de contaje / valor medido
50	USER_STAT_DWORD3	WORD	W#16#0	Según parametrización del valor de contaje / valor medido

Petición de escritura

Petición de escritura

Para lanzar una nueva petición de escritura se registra una petición de escritura en JOB_WR.NO. Se admiten las siguientes peticiones de escritura:

JOB_WR.NO (DBB0)	Entrada en CNT2_CHANTYPE	Dirección en el DB de contaje	Significado del modo de contaje	Significado del modo de medición
0	Ninguna	Ninguna	Ninguna petición de escritura / última petición de escritura finalizada	
10	LOAD_VAL0	52	Cargar directamente contador 0	Cargar límite inferior 0
11	LOAD_VAL1	56	Cargar directamente contador 1	Cargar límite inferior 1
12	LOAD_VAL2	60	Cargar directamente contador 2	Cargar límite inferior 2
13	LOAD_VAL3	64	Cargar directamente contador 3	Cargar límite inferior 3
14	LOAD_VAL4	68	Cargar directamente contador 4	Cargar límite inferior 4
15	LOAD_VAL5	72	Cargar directamente contador 5	Cargar límite inferior 5
16	LOAD_VAL6	76	Cargar directamente contador 6	Cargar límite inferior 6
17	LOAD_VAL7	80	Cargar directamente contador 7	Cargar límite inferior 7
20	LOAD_PREPARE_VAL0	84	Carga preliminar del contador 0	Cargar límite superior 0
21	LOAD_PREPARE_VAL1	88	Carga preliminar del contador 1	Cargar límite superior 1
22	LOAD_PREPARE_VAL2	92	Carga preliminar del contador 2	Cargar límite superior 2

JOB_WR.NO (DBB0)	Entrada en CNT2_CHANTYPE	Dirección en el DB de contaje	Significado del modo de contaje	Significado del modo de medición
23	LOAD_PREPARE_VAL3	96	Carga preliminar del contador 3	Cargar límite superior 3
24	LOAD_PREPARE_VAL4	100	Carga preliminar del contador 4	Cargar límite superior 4
25	LOAD_PREPARE_VAL5	104	Carga preliminar del contador 5	Cargar límite superior 5
26	LOAD_PREPARE_VAL6	108	Carga preliminar del contador 6	Cargar límite superior 6
27	LOAD_PREPARE_VAL7	112	Carga preliminar del contador 7	Cargar límite superior 7
30	CMP_VAL0	116	Cargar comparador 0	
31	CMP_VAL1	120	Cargar comparador 1	
32	CMP_VAL2	124	Cargar comparador 2	
33	CMP_VAL3	128	Cargar comparador 3	
34	CMP_VAL4	132	Cargar comparador 4	
35	CMP_VAL5	136	Cargar comparador 5	
36	CMP_VAL6	140	Cargar comparador 6	
37	CMP_VAL7	144	Cargar comparador 7	
40	LOAD_VAL0 a LOAD_VAL3	52 a 67	Cargar directamente contadores 0 a 3	Cargar límite inferior 0 a 3
41	LOAD_VAL4 a LOAD_VAL7	68 a 83	Cargar directamente contadores 4 a 7	Cargar límite inferior 4 a 7
42	LOAD_VAL0 a LOAD_VAL7	52 a 83	Cargar directamente contadores 0 a 7	Cargar límite inferior 0 a 7
50	LOAD_PREPARE_VAL0 a LOAD_PREPARE_VAL3	84 a 99	Carga preliminar de contadores 0 a 3	Cargar límite superior 0 a 3
51	LOAD_PREPARE_VAL4 a LOAD_PREPARE_VAL7	100 a 111	Carga preliminar de contadores 4 a 7	Cargar límite superior 4 a 7
52	LOAD_PREPARE_VAL0 a LOAD_PREPARE_VAL7	84 a 111	Carga preliminar de contadores 0 a 7	Cargar límite superior 0 a 7
60	CMP_VAL0 a CMP_VAL3	116 a 131	Cargar comparadores 0 a 3	
61	CMP_VAL4 a CMP_VAL7	132 a 147	Cargar comparadores 4 a 7	
62	CMP_VAL0 a CMP_VAL7	116 a 147	Cargar comparadores 0 a 7	

Estado de la petición de escritura

Estado de una petición de escritura

El estado de una petición de escritura se indica en el DB de contaje (byte de datos DBB1).

Bit en JOB_WR (DBX1.)	Significado
.BUSY, 0	= 1: Petición de escritura en curso. La instrucción CNT2WRPN activa este bit tan pronto como procesa una petición de escritura (JOB_WR.NO > 0 y JOB_WR.IMPOSS = 0). La instrucción CNT2WRPN borra el bit en cuanto finaliza la petición de escritura (JOB_WR.NO = 0).
.DONE, 1	= 1: Petición de escritura finalizada. La instrucción CNT2WRPN activa este bit en cuanto finaliza una petición de escritura (también errónea). La instrucción CNT2WRPN borra el bit en cuanto comienza una nueva petición de escritura. También se puede borrar este bit desde el programa de usuario.
.IMPOSS, 2	= 1: La petición de escritura no puede procesarse (el módulo de contaje no está parametrizado, el arranque o el modo de programadora están activos). La petición de escritura (JOB_WR) puede dejarse pendiente o borrarse. La instrucción CNT2WRPN borra el bit cuando se cumplen las condiciones anteriores.
.UNKOWN, 3	= 1: Petición de escritura desconocida. La petición de escritura (JOB_WR) indicada no se encuentra dentro del rango permitido (véase mensaje de error). La instrucción CNT2WRPN borra este bit en cuanto se introduce un número válido en JOB_WR. El número desconocido se mantiene mientras tanto.

Los errores ocurridos se indican en el resultado binario (BIE = 0). Los errores posibles son:

- Petición de escritura desconocida (véase JOB_WR.UNKOWN).
- Error al transferir los datos con la instrucción WRREC. El error queda disponible en el parámetro de salida RET_VAL.
- El módulo comprueba e interpreta los datos transferidos en busca de errores de datos. Si se produce un error de datos, en el DB de contaje se activa el bit CHECKBACK_SIGNALS.DATA_ERR = 1.

Petición de lectura

Petición de lectura

Para lanzar una nueva petición de lectura se introduce una petición de lectura en JOB_RD.NO. Se admiten las siguientes peticiones de lectura:

JOB_RD.NO (DBB2)	Entrada en CNT2_CHANTYPE	Dirección en el DB de contaje	Significado
0	Ninguna	Ninguna	Ninguna petición de lectura / última petición de lectura finalizada
100	ACT_CNTV0 ACT_MSRV0 ACT_CNTV1 ACT_MSRV1 ACT_CNTV2 ACT_MSRV2 ACT_CNTV3 ACT_MSRV3	148	Estado actual del contador 0 a 3 y resultado de la medición 0 a 3
101	ACT_CNTV4 ACT_MSRV4 ACT_CNTV5 ACT_MSRV5 ACT_CNTV6 ACT_MSRV6 ACT_CNTV7 ACT_MSRV7	180	Estado actual del contador 4 a 7 y resultado de la medición 4 a 7

Estado de la petición de lectura

Estado de una petición de lectura

El estado de una petición de lectura se indica en el DB de contaje (byte de datos DBB3).

Bit en JOB_RD (DBX3.)	Descripción
.BUSY, 0	= 1: Petición de lectura en curso. La instrucción CNT2RDPN activa este bit en cuanto se procesa una petición de lectura (JOB_RD.NO > 0 y JOB_RD.IMPOSS = 0). La instrucción CNT2RDPN desactiva el bit en cuanto finaliza la petición de lectura (JOB_RD.NO = 0).
.DONE, 1	= 1: Petición de lectura finalizada. La instrucción CNT2RDPN activa este bit en cuanto finaliza una petición de lectura (también errónea). La instrucción CNT2RDPN borra el bit en cuanto comienza una nueva petición de lectura. También es posible borrar este bit desde el programa de usuario.

Bit en JOB_RD (DBX3.)	Descripción
.IMPOSS, 2	= 1: La petición de lectura no puede procesarse (el módulo de contaje no está parametrizado, el arranque o el modo de programadora están activos). La petición de lectura (JOB_RD.NO) puede dejarse pendiente o borrarse. La instrucción CNT2RDPN borra el bit cuando se cumplen las condiciones anteriores.
.UNKOWN, 3	= 1: Petición de lectura desconocida. La petición de lectura (JOB_RD) indicada no se encuentra dentro del rango permitido (véase evaluación de errores). La instrucción CNT2RDPN borra este bit en cuanto se introduce un número válido en JOB_RD.NO. El número desconocido se mantiene hasta que esto ocurre.

Los errores que se producen se muestran en el resultado binario (RB = 0). Los posibles errores son:

- Petición de lectura desconocida (véase JOB_RD.UNKOWN).
- Error al transferir los datos con la instrucción RDREC. El error queda disponible en el parámetro de salida RET_VAL.

Posicionamiento FM x51

ABS_INIT

Descripción

La instrucción ABS_INIT inicializa los datos siguientes en el DB de canal:

- las señales de control
- las señales de respuesta
- los bits de disparo, bits de finalización y bits de error de las peticiones
- los selectores de función y sus bits de finalización y error
- la gestión de peticiones para la instrucción ABS_CTRL (Página 7420) o bien ABS_CTRL_451 (Página 7424)

Llamada

La instrucción tiene que ejecutarse tras un arranque, es decir, después de conectar la fuente de alimentación del módulo o de la CPU para cada canal. Por ello hay que llamarla, por ej. en el OB de rearranque OB 100 y en el OB de presencia de módulo OB 83 o en la fase de inicialización del programa de usuario. De este modo queda asegurado que tras un rearranque completo de la CPU o un arranque del módulo su programa de usuario no acceda a datos antiguos.

Bloque de datos utilizado

DB de canal (Página 7428):

En el DB de canal es preciso que esté registrada la dirección del módulo. La entrada debe realizarse en el OB de arranque (OB 100).

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de canal

Valores de retorno

La instrucción no devuelve ningún valor de retorno.

ABS_CTRL

Descripción

La instrucción ABS_CTRL permite leer los datos operativos de cada canal del módulo, parametrizar los canales y controlar su funcionamiento. Para ello se utilizan señales de control, señales de respuesta, selectores de función y peticiones de escritura y lectura.

Funcionamiento

Con cada llamada la instrucción ejecuta las actividades siguientes:

- Leer señales de respuesta:
  La instrucción ABS_CTRL lee todas las señales de respuesta de un canal y las transfiere al DB de canal. Como las señales de control y las peticiones solo se procesan al final, las señales de respuesta reflejan el estado del canal antes de llamar la instrucción.
- Gestión de peticiones (Página 7438):
  La instrucción ABS_CTRL procesa las peticiones de escritura y lectura y transfiere los datos entre el DB de canal, DB de parámetros y el módulo.
- Escribir señales de control:
  Las señales de control que se encuentran en el DB de canal se transfieren al módulo.

Llamada

La instrucción ABS_CTRL tiene que llamarse cíclicamente para cada canal, por ej. en el OB 1.

Antes de llamar la instrucción ABS_CTRL, registre en el DB de canal todos los datos necesarios para la ejecución de las funciones deseadas.

Bloques de datos empleados

- DB de canal (Página 7428):
  El DB de canal es el DB instancia de la instrucción ABS_CTRL. En el DB de canal es preciso que estén registradas la dirección del módulo y el número de canal. Si se introducen datos erróneos, es posible que se produzcan errores de acceso a periferia o que se acceda a otro módulo, lo que provocaría falsificaciones de datos.
- DB de parámetros (Página 7434):
  Para leer o escribir datos máquina mediante peticiones, se requiere un DB de parámetros cuyo número tiene que estar registrado en el DB de canal.

Valores de retorno

La instrucción devuelve los siguientes valores de retorno:

RETVAL	RB	Descripción
1	1	Por lo menos 1 petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Errores: se ha producido un error de datos (DATA_ERR) o un error de comunicación (JOB_ERR)

Señales de control

Si hay una señal de PARADA o un error de operador o falta la habilitación de accionamiento, la instrucción desactiva las señales de control START, DIR_M y DIR_P.

Existe la posibilidad de volver a iniciar un desplazamiento después de acusar el error de operador con OT_ERR_A=1. Con este acuse no es posible iniciar más peticiones ni señales de control.

Cuando hay pendiente un error de operador, la instrucción pone a 0 el acuse del error de operador OT_ERR_A.

Cuando el canal notifica el inicio del desplazamiento, la instrucción desactiva las señales de inicio START, DIR_P y DIR_M, excepto con el modo de operación "Jog".

Si no se ha parametrizado el eje, la instrucción retiene todas las señales de control a excepción del acuse de error de operador OT_ERR_A.

Peticiones y señales de control

Es posible iniciar varias peticiones de forma simultánea, incluso con las señales de control necesarias para el posicionamiento. En el caso de que se hayan iniciado simultáneamente por lo menos una petición de escritura con las señales de control START, DIR_M o DIR_P, la instrucción retendrá dichas señales de control hasta que se hayan procesado las peticiones de escritura.

Arranque

Llame la instrucción ABS_INIT (Página 7419) al arrancar el módulo o la CPU. Para ello se resetean, entre otros, los selectores de función. La instrucción ABS_CTRL acusa el arranque del módulo. Durante este período, RETVAL y JOBBUSY = 1.

Comportamiento en caso de error

Si se han escrito datos erróneos en una petición de escritura, el canal envía la respuesta DATA_ERR = 1 en el DB de canal. Cuando se produce un error en la comunicación con el módulo con la petición de escritura o lectura, se guarda la causa del error en el parámetro JOB_ERR del DB de canal.

- Error en una petición de escritura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D. En todas las peticiones de escritura pendientes, el bit de inicio también vuelve a su posición anterior pero solo se activa el bit de error _ERR. Se desactivan las peticiones de escritura pendientes porque aquí es posible apilar las peticiones.
  Las peticiones de lectura pendientes se siguen procesando. Para ello se vuelve a activar JOB_ERR para cada petición.
- Error en una petición de lectura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D.
  Las peticiones de lectura todavía pendientes se siguen procesando. Para ello se vuelve a activar JOB_ERR para cada petición.

ABS_DIAG

Descripción

La instrucción ABS_DIAG lee el búfer de diagnóstico del módulo y lo pone a disposición de un indicador en el sistema de manejo y visualización (M+V) o de una evaluación programada.

Funcionamiento

La instrucción lee el búfer de diagnóstico cuando se indica una nueva entrada mediante la señal de realimentación DIAG = 1 en el DB de canal. Tras leer el búfer de diagnóstico, el módulo pone a 0 el bit DIAG en el DB de canal.

Llamada

La instrucción tiene que llamarse cíclicamente, por ej. en el OB 1. No se permite realizar una llamada adicional en un OB de alarma. Para una ejecución completa de la instrucción hacen falta por lo menos 2 llamadas (ciclos).

Bloque de datos empleado

DB de diagnóstico (Página 7436):

El DB de diagnóstico es el DB instancia de la instrucción ABS_DIAG. Es necesario que en el DB de diagnóstico esté registrada la dirección del módulo. La entrada más reciente del búfer de diagnóstico se registra en la estructura DIAG[1] y la más antigua en la estructura DIAG[9].

Valores de retorno

La instrucción proporciona los siguientes valores de retorno en el parámetro RETVAL del DB de diagnóstico:

RETVAL	RB	Descripción
1	1	Petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Error

Peticiones

El búfer de diagnóstico se puede leer aunque no haya ninguna entrada nueva activando el bit de inicio DIAGRD_EN en el DB de diagnóstico. Tras leer el búfer de diagnóstico el bit de inicio se pone a 0.

Ejecute esta petición después de realizar un arranque de la CPU y del módulo. Con ello se asegura que el contenido del DB de diagnóstico coincide con el contenido del búfer de diagnóstico del módulo, incluso cuando el módulo no haya realizado ninguna entrada nueva en el búfer de diagnóstico.

Arranque

La instrucción no ejecuta ningún procesamiento del arranque.

Comportamiento en caso de error

Si la ejecución fuera errónea, encontrará la causa del error en el DB de diagnóstico, en el parámetro JOB_ERR.

Datos de diagnóstico

Error	Evaluar mediante OB 82
Fallo del módulo	OB82_MDL_DEFECT
Error interno	OB82_INT_FAULT
Fallo externo	OB82_EXT_FAULT
Hay un error de canal	OB82_PNT_INFO
Falta tensión auxiliar externa	OB82_EXT_VOLTAGE
Activación de la vigilancia de tiempo	OB82_WTCH_DOG_FLT
Fallo de la fuente de alimentación interna del módulo	OB82_INT_PS_FLT
Se ha perdido una alarma de proceso	OB82_HW_INTR_FLT

Clases de error

Los errores que se producen están agrupados en las siguientes clases de error:

- Clase de error 1: fallo operativo (Página 7444)
- Clase de error 2: error de operador (Página 7445)
- Clase de error 4: error de datos (Página 7446)
- Clase de error 5: error de datos máquina (Página 7447)
- Clase de error 6: error de la tabla de incrementos (Página 7450)
- Clase de error 15: avisos (Página 7450)
- Clase de error 128: error de diagnóstico (Página 7451)

ABS_CTRL_451

Descripción

La instrucción ABS_CTRL_451 permite leer los datos operativos de cada canal del módulo, parametrizar los canales y controlar su funcionamiento. Para ello se utilizan señales de control, señales de respuesta, selectores de función y peticiones de escritura y lectura.

Funcionamiento

Con cada llamada la instrucción ejecuta las actividades siguientes:

- Leer señales de respuesta:
  La instrucción ABS_CTRL_451 lee todas las señales de respuesta de un canal y las transfiere al DB de canal. Como las señales de control y las peticiones sólo se procesan al final, las señales de respuesta reflejan el estado del canal antes de llamar la instrucción.
- Gestión de peticiones (Página 7441):
  La instrucción ABS_CTRL_451 procesa las peticiones de escritura y lectura y transfiere los datos entre el DB de canal, DB de parámetros y el módulo.
- Escribir señales de control:
  Las señales de control que se encuentran en el DB de canal se transfieren al módulo.

Llamada

La instrucción ABS_CTRL_451 tiene que llamarse cíclicamente para cada canal, por ej. en el OB 1.

Antes de llamar la instrucción ABS_CTRL_451, registre en el DB de canal todos los datos necesarios para la ejecución de las funciones deseadas.

Bloques de datos empleados

- DB de canal (Página 7428):
  En el DB de canal es preciso que estén registradas la dirección del módulo y el número de canal. Si se introducen datos erróneos, es posible que se produzcan errores de acceso a periferia o que se acceda a otro módulo, lo que provocaría falsificaciones de datos.
- DB de parámetros (Página 7434):
  Para leer o escribir datos máquina mediante peticiones, se requiere un DB de parámetros cuyo número tiene que estar registrado en el DB de canal.

Parámetros

Parámetros	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de canal
RET_VAL	OUTPUT	INT	Valor de retorno

Valores de retorno

La instrucción devuelve los siguientes valores de retorno:

RET_VAL	RB	Descripción
1	1	Por lo menos 1 petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Errores: se ha producido un error de datos (DATA_ERR) o un error de comunicación (JOB_ERR)

Señales de control

Si hay una señal de PARADA o un error de operador o falta la habilitación de accionamiento, la instrucción desactiva las señales de control START, DIR_M y DIR_P.

Existe la posibilidad de volver a iniciar un desplazamiento después de acusar el error de operador con OT_ERR_A=1. Con este acuse no es posible iniciar más peticiones ni señales de control.

Cuando hay pendiente un error de operador, la instrucción pone a 0 el acuse del error de operador OT_ERR_A.

Cuando el canal notifica el inicio del desplazamiento, la instrucción desactiva las señales de inicio START, DIR_P y DIR_M, excepto con el modo de operación "Jog".

Si no se ha parametrizado el eje, la instrucción retiene todas las señales de control a excepción del acuse de error de operador OT_ERR_A.

Peticiones y señales de control

Es posible iniciar varias peticiones de forma simultánea, incluso con las señales de control necesarias para el posicionamiento. En el caso de que se haya emitido al menos una petición

de escritura con las señales de control START, DIR_M o DIR_P, la instrucción retiene estas señales de control hasta que las peticiones de escritura se hayan procesado.

Arranque

Llame la instrucción ABS_INIT (Página 7419) al arrancar el módulo o la CPU. Para ello se resetan, entre otros, los selectores de función. La instrucción ABS_CTRL_451 acusa el arranque del módulo. Durante este período, RETVAL y JOBBUSY = 1.

Comportamiento en caso de error

Si se han escrito datos erróneos en una petición de escritura, el canal envía la respuesta DATA_ERR = 1 en el DB de canal. Cuando se produce un error en la comunicación con el módulo con la petición de escritura o lectura, se guarda la causa del error en el parámetro JOB_ERR del DB de canal.

- Error en una petición de escritura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D. En todas las peticiones de escritura pendientes, el bit de inicio también vuelve a su posición anterior pero sólo se activa el bit de error _ERR. Se desactivan las peticiones de escritura pendientes porque aquí es posible apilar las peticiones.
  Las peticiones de lectura en cola continúan siendo procesadas. Para ello se vuelve a activar JOB_ERR para cada petición.
- Error en una petición de lectura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D.
  Las peticiones de lectura todavía pendientes se siguen procesando. Para ello se vuelve a activar JOB_ERR para cada petición.

ABS_DIAG_451

Descripción

La instrucción ABS_DIAG_451 lee el búfer de diagnóstico del módulo y lo pone a disposición de un indicador en el sistema de manejo y visualización (M+V) o de una evaluación programada.

Funcionamiento

La instrucción lee el búfer de diagnóstico cuando en él se muestra una nueva entrada por medio de la señal de respuesta DIAG = 1 del DB de canal. Tras leer el búfer de diagnóstico, el módulo pone a 0 el bit DIAG en el DB de canal.

Llamada

La instrucción tiene que llamarse cíclicamente, por ej. en el OB 1. No se permite realizar una llamada adicional en un OB de alarma. Para una ejecución completa de la instrucción hacen falta por lo menos 2 llamadas (ciclos).

Bloque de datos utilizado

DB de diagnóstico (Página 7436):

Es necesario que en el DB de diagnóstico esté registrada la dirección del módulo. La entrada más reciente del búfer de diagnóstico se registra en la estructura DIAG[1] y la más antigua en la estructura DIAG[9].

Parámetros

Parámetros	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de diagnóstico
RET_VAL	OUTPUT	INT	Valor de retorno

Valores de retorno

La instrucción devuelve los siguientes valores de retorno en el parámetro RET_VAL del DB de diagnóstico:

RET_VAL	RB	Descripción
1	1	Petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Error

Peticiones

El búfer de diagnóstico se puede leer aunque no haya ninguna nueva entrada mediante activación del bit de inicio DIAGRD_EN en el DB de diagnóstico. Tras leer el búfer de diagnóstico el bit de inicio se pone a 0.

Ejecute esta petición después de realizar un arranque de la CPU y del módulo. Con ello se asegura que el contenido del DB de diagnóstico coincide con el contenido del búfer de diagnóstico del módulo, incluso cuando el módulo no haya realizado ninguna entrada nueva en el búfer de diagnóstico.

Arranque

La instrucción no ejecuta ningún procesamiento del arranque.

Comportamiento en caso de error

Si la ejecución fuera errónea, encontrará la causa del error en el DB de diagnóstico, en el parámetro JOB_ERR.

Datos de diagnóstico

Error	Evaluar mediante OB 82
Fallo del módulo	OB82_MDL_DEFECT
Error interno	OB82_INT_FAULT
Fallo externo	OB82_EXT_FAULT
Hay un error de canal	OB82_PNT_INFO
Falta tensión auxiliar externa	OB82_EXT_VOLTAGE
Falta conector frontal	OB82_FLD_CONNCTR
Activación de la vigilancia de tiempo	OB82_WTCH_DOG_FLT
Fallo de la fuente de alimentación interna del módulo	OB82_INT_PS_FLT
Se ha perdido una alarma de proceso	OB82_HW_INTR_FLT

Clases de error

Los errores que se producen están agrupados en las siguientes clases de error:

- Clase de error 1: error de operador (Página 7444)
- Clase de error 2: error de operador (Página 7445)
- Clase de error 4: error de datos (Página 7446)
- Clase de error 5: error de datos máquina (Página 7447)
- Clase de error 6: error de la tabla de incrementos (Página 7450)
- Clase de error 15: avisos (Página 7450)
- Clase de error 128: error de diagnóstico (Página 7451)

Más referencias para FM x51 ABS V1

Bloque de datos de canal

Tarea

El DB de canal es la interfaz de datos entre el programa de usuario y el FM x51. Contiene y aplica todos los datos necesarios para el control y el funcionamiento de un canal.

Estructura

El DB de canal se divide en diferentes áreas:

DB de canal
Direcciones *)
Señales de control
Señales de respuesta

DB de canal
Selectores de función
Bits de disparo de peticiones de escritura
Bits de disparo de peticiones de lectura
Bits de finalización
Bits de error
Gestión de peticiones para instrucciones
Datos de las peticiones
*) La dirección del módulo se puede registrar en la interfaz de parametrización.

Contenido

La siguiente tabla muestra el contenido del DB de canal.

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
Direcciones				
0.0	MOD_ADDR	INT	0	Dirección del módulo
2.0	CH_NO	INT	1	Número de canal
10.0	PARADBNO	INT	-1	Número del DB de parámetros
Señales de control				
14.3	OT_ERR_A	BOOL	FALSE	1 = Acusar error de operador
15.0	START	BOOL	FALSE	1 = Iniciar posicionamiento
15.1	STOP	BOOL	FALSE	1 = Detener desplazamiento en marcha
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.6	SPEED252	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: velocidad inicial de modo incremental con número de incremento 252: – 0 = Marcha lenta – 1 = Marcha rápida
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	Modo de operación solicitado • 0 = Ningún modo de operación • 1 = Jog • 3 = Búsqueda del punto de referencia • 4 = Modo incremental relativo • 5 = Modo incremental absoluto

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
17.0	MODE_TYPE	BYTE	B#16#0	• Velocidad inicial para el modo Jog – 0 = Marcha lenta – 1 = Marcha rápida • Número de incremento para el modo incremental
Señales de realimentación				
22.2	DIAG	BOOL	FALSE	1 = Búfer de diagnóstico modificado
22.3	OT_ERR	BOOL	FALSE	1 = Hay un error de operador
22.4	DATA_ERR	BOOL	FALSE	1 = Error de datos
22.7	PARA	BOOL	FALSE	1 = El eje está parametrizado
23.0	ST_ENBLD	BOOL	FALSE	1 = Inicio habilitado
23.1	WORKING	BOOL	FALSE	1 = Posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera habilitación externa
23.4	SPEED_OUT	BOOL	FALSE	• 0 = Marcha lenta • 1 = Marcha rápida
23.5	ZSPEED	BOOL	FALSE	1 = El eje está en la zona de parada
23.6	CUTOFF	BOOL	FALSE	1 = El eje está en la zona de desconexión
23.7	CHGOVER	BOOL	FALSE	1 = El eje está en la zona de cambio de velocidad
24.0	MODE_OUT	BYTE	B#16#0	Modo de operación activo
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado
25.1	MSR_DONE	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Medición de longitud / captura de bordes terminada
25.2	GO_M	BOOL	FALSE	1 = El eje se desplaza en sentido negativo
25.3	GO_P	BOOL	FALSE	1 = El eje se desplaza en sentido positivo
25.5	FVAL_DONE	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Ha finalizado el ajuste del valor real al vuelo
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada
26.0	ACT_POS	DINT	L#0	Valor real actual (posición actual del eje)
Selectores de función				
34.0	PLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido positivo
34.1	MLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido negativo
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación
34.3	EDGE_ON	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Captura de bordes activada
34.4	MSR_ON	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Medición de longitud activada
Bits de disparo de peticiones de escritura				
35.0	MDWR_EN	BOOL	FALSE	1 = Escribir datos máquina
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
35.2	DELDIST_EN	BOOL	FALSE	1 = Borrar distancia residual
35.3	AVALREM_EN	BOOL	FALSE	1 = Cancelar preselección del valor real
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (número de incremento 1 … 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (número de incremento 51 … 100)
35.6	REFPT_EN	BOOL	FALSE	1 = Definir punto de referencia
35.7	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.0	FVAL_EN	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Ajustar valor real al vuelo
36.1	ZOFF_EN	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Activar decalaje de señal cero
36.2	TRG252_254_EN	BOOL	FALSE	1 = Escribir incremento del número de incremento 252 ó 254
36.3	TRG255_EN	BOOL	FALSE	1 = Escribir incremento del número de incremento 255
36.4	DELDIAG_EN	BOOL	FALSE	1 = Borrar búfer de diagnóstico
Bits de disparo de peticiones de lectura				
36.5	MDRD_EN	BOOL	FALSE	1 = Leer datos máquina
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 1 (número de incremento 1 … 50)
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 2 (número de incremento 51 … 100)
37.0	MSRRD_EN	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Leer valores medidos
37.1	ACTSPD_EN	BOOL	FALSE	1 = Leer velocidad real, distancia residual e incremento actual
37.2	ENCVAL_EN	BOOL	FALSE	1 = Leer valores del encóder
Bits de finalización de selectores de función				
38.0	PLOOP_D	BOOL	FALSE	1 = Petición finalizada de "Aproximación en bucle en sentido positivo"
38.1	MLOOP_D	BOOL	FALSE	1 = Petición finalizada de "Aproximación en bucle en sentido negativo"
38.2	EI_D	BOOL	FALSE	1 = Petición finalizada de "No evaluar entrada de habilitación"
38.3	EDGE_D	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Petición finalizada de "Captura de bordes activada"
38.4	MSR_D	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Petición finalizada de "Medición de longitud activada"
Bits de finalización de peticiones de escritura				

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
39.0	MDWR_D	BOOL	FALSE	1 = Petición finalizada de "Escribir datos máquina"
39.1	MD_D	BOOL	FALSE	1 = Petición finalizada de "Activar datos máquina"
39.2	DELDIST_D	BOOL	FALSE	1 = Petición finalizada de "Borrar distancia residual"
39.3	AVALREM_D	BOOL	FALSE	1 = Petición finalizada de "Cancelar preselección del valor real"
39.4	TRGL1WR_D	BOOL	FALSE	1 = Petición finalizada de "Escribir tabla de incrementos 1"
39.5	TRGL2WR_D	BOOL	FALSE	1 = Petición finalizada de "Escribir tabla de incrementos 2"
39.6	REFPT_D	BOOL	FALSE	1 = Petición finalizada de "Definir punto de referencia"
39.7	AVAL_D	BOOL	FALSE	1 = Petición finalizada de "Ajustar valor real"
40.0	FVAL_D	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Petición finalizada de "Ajustar valor real al vuelo"
40.1	ZOFF_D	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Petición finalizada de "Activar decalaje de señal cero"
40.2	TRG252_254_D	BOOL	FALSE	1 = Petición finalizada de "Escribir incremento del número de incremento 252 ó 254"
40.3	TRG255_D	BOOL	FALSE	1 = Petición finalizada de "Escribir incremento del número de incremento 255"
40.4	DELDIAG_D	BOOL	FALSE	1 = Petición finalizada de "Borrar búfer de diagnóstico"
Bits de finalización de peticiones de lectura				
40.5	MDRD_D	BOOL	FALSE	1 = Petición finalizada de "Leer datos máquina"
40.6	TRGL1RD_D	BOOL	FALSE	1 = Petición finalizada de "Leer tabla de incrementos 1"
40.7	TRGL2RD_D	BOOL	FALSE	1 = Petición finalizada de "Leer tabla incrementos 2"
41.0	MSRRD_D	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Petición finalizada de "Leer valores medidos"
41.1	ACTSPD_D	BOOL	FALSE	1 = Petición finalizada de "Leer velocidad real, trayecto residual e incremento actual"
41.2	ENCVAL_D	BOOL	FALSE	1 = Petición finalizada de "Leer valores del encóder"
Bits de error de selectores de función				
42.0	PLOOP_ERR	BOOL	FALSE	1 = Error en la petición "Aproximación en bucle en sentido positivo"
42.1	MLOOP_ERR	BOOL	FALSE	1 = Error en la petición "Aproximación en bucle en sentido negativo"
42.2	EI_ERR	BOOL	FALSE	1 = Error en la petición "No evaluar entrada de habilitación"
42.3	EDGE_ERR	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Error en la petición "Captura de bordes activada"

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
42.4	MSR_ERR	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Error en la petición "Medición de longitud activada"
Bits de error de peticiones de escritura				
43.0	MDWR_ERR	BOOL	FALSE	1 = Error en la petición "Escribir datos máquina"
43.1	MD_ERR	BOOL	FALSE	1 = Error en la petición "Activar datos máquina"
43.2	DELDIST_ERR	BOOL	FALSE	1 = Error en la petición de "Borrar distancia residual"
43.3	AVALREM_ERR	BOOL	FALSE	1 = Error en la petición "Deshacer ajuste de valor real"
43.4	TRGL1WR_ERR	BOOL	FALSE	1 = Error en la petición "Escribir tabla de incrementos 1"
43.5	TRGL2WR_ERR	BOOL	FALSE	1 = Error en la petición "Escribir tabla de incrementos 2"
43.6	REFPT_ERR	BOOL	FALSE	1 = Error en la petición "Definir punto de referencia"
43.7	AVAL_ERR	BOOL	FALSE	1 = Error en la petición "Ajustar valor real"
44.0	FVAL_ERR	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Error en la petición "Ajustar valor real al vuelo"
44.1	ZOFF_ERR	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Error en la petición "Activar decalaje de señal cero"
44.2	TRG252_254_ERR	BOOL	FALSE	1 = Error en la petición "Escribir incremento del número de incremento 252 ó 254"
44.3	TRG255_ERR	BOOL	FALSE	1 = Error en la petición "Escribir incremento del número de incremento 255"
44.4	DELDIAG_ERR	BOOL	FALSE	1 = Error en la petición "Borrar búfer de diagnóstico"
Bits de error de peticiones de lectura				
44.5	MDRD_ERR	BOOL	FALSE	1 = Error en la petición "Leer datos máquina"
44.6	TRGL1RD_ERR	BOOL	FALSE	1 = Error en la petición "Leer tabla de incrementos 1"
44.7	TRGL2RD_ERR	BOOL	FALSE	1 = Error en la petición "Leer tabla de incrementos 2"
45.0	MSRRD_ERR	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: 1 = Error en la petición "Leer valores medidos"
45.1	ACTSPD_ERR	BOOL	FALSE	1 = Error en la petición de "Leer velocidad real, distancia residual e incremento actual"
45.2	ENCVAL_ERR	BOOL	FALSE	1 = Error en la petición "Leer valores actuales del encóder"
Gestión de peticiones para la instrucción ABS_CTRL o bien ABS_CTRL_451				
48.0	JOB_ERR	INT	0	Número del error de comunicación
50.0	JOBBUSY	BOOL	FALSE	1 = Por lo menos hay una petición en curso
50.1	JOBRESET	BOOL	FALSE	1 = Resetear todos los bits de error y finalización
Datos de la petición "Decalaje de señal cero" (FM 451)				
80.0	ZOFF	DINT	L#0	• FM 351: 0 (no utilizado) • FM 451: Decalaje de señal cero

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
Datos de la petición "Ajustar valor real"				
84.0	AVAL	DINT	L#0	Coordenada de "Ajustar valor real"
Datos de la petición "Ajustar valor real al vuelo" (FM 451)				
88.0	FVAL	DINT	L#0	• FM 351: 0 (no utilizado) • FM 451: Coordenada de "Ajustar valor real al vuelo"
Datos de la petición "Definir punto de referencia"				
92.0	REFPT	DINT	L#0	Coordenada de "Definir punto de referencia"
Datos de la petición "Escribir incremento del número de incremento 252 ó 254"				
96.0	TRG252_254	DINT	L#0	Incremento del número de incremento 252 ó 254
Datos de la petición "Escribir incremento del número de incremento 255"				
100.0	TRG255	DINT	L#0	Incremento del número de incremento 255
104.0	CHGDIF255	DINT	L#0	Distancia de cambio de velocidad del número de incremento 255
108.0	CUTDIF255	DINT	L#0	Distancia de desconexión del número de incremento 255
Datos de la petición "Leer datos de posición"				
112.0	ACTSPD	DINT	L#0	Velocidad actual
116.0	DIST_TO_GO	DINT	L#0	Distancia residual
120.0	ACT_TRG	DINT	L#0	Incremento actual
Datos de la petición "Leer datos de encóder"				
124.0	ENCVAL	DINT	L#0	Valor real del encóder (representación interna)
128.0	ZEROVAL	DINT	L#0	Último valor de señal cero (representación interna)
132.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto
Datos de la petición "Medición de longitud/Captura de bordes" (FM 451)				
136.0	BEG_VAL	DINT	L#0	• FM 351: 0 (no utilizado) • FM 451: Valor inicial de medición de longitud / captura de bordes
140.0	END_VAL	DINT	L#0	• FM 351: 0 (no utilizado) • FM 451: Valor final de medición de longitud / captura de bordes
144.0	LEN_VAL	DINT	L#0	• FM 351: 0 (no utilizado) • FM 451: Longitud

Bloque de datos de parámetros

Tarea

Si se desean modificar los datos máquina y las tablas de incrementos durante el funcionamiento, se necesita el DB de parámetros en el que están guardados dichos datos. Los parámetros pueden ser modificados por el programa de usuario o por un sistema de manejo y visualización.

Los datos que se muestran en la interfaz de parametrización se pueden exportar a un DB de parámetros. También es posible importar un DB de parámetros en la interfaz de parametrización y visualizarlo.

En todos los canales del módulo puede haber varios juegos de datos de parametrización, p. ej. para distintas recetas, a los que se puede conmutar por medio del programa.

Estructura

DB de parámetros
Datos máquina
Tablas de incrementos

Contenido

La siguiente tabla muestra el contenido del DB de parámetros.

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
Datos máquina				
4.0	EDGEDIST	DINT	L#0	• FM 351: 0 (no utilizado) • FM 451: Distancia mínima entre bordes
8.0	UNITS	DINT	L#1	Sistema de medición
12.0	AXIS_TYPE	DINT	L#0	0 = Eje lineal, 1 = Eje rotativo
16.0	ENDROTAX	DINT	L#100 000	Final del eje rotativo
20.0	ENC_TYPE	DINT	L#1	Tipo de encóder, longitud de telegrama
24.0	DISP_REV	DINT	L#80 000	Recorrido por vuelta
32.0	INC_REV	DINT	L#500	Incrementos por vuelta
36.0	NO_REV	DINT	L#1	Número de vueltas
40.0	BAUDRATE	DINT	L#0	Velocidad de transferencia
44.0	REFPT	DINT	L#0	Coordenada del punto de referencia
48.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto
52.0	REFPT_TYPE	DINT	L#0	Tipo de búsqueda del punto de referencia
59.0	CNT_DIR	BOOL	FALSE	Sentido de contaje: • 0 = normal • 1 = invertido
63.0	MON_WIRE	BOOL	TRUE	1 = Vigilancia de rotura de hilo
63.1	MON_FRAME	BOOL	TRUE	1 = Vigilancia de error de telegrama
63.2	MON_PULSE	BOOL	TRUE	1 = Vigilancia de error en señal
64.0	SSW_STRT	DINT	L#-100 000 000	Final de carrera de software Inicio

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
68.0	SSW_END	DINT	L#100 000 000	Final de carrera de software Fin
76.0	TRG_RANGE	DINT	L#1000	Zona de destino
80.0	MON_TIME	DINT	L#2000	Tiempo de vigilancia [ms]
84.0	ZSPEED_R	DINT	L#1000	Zona de parada
88.0	ZSPEED_L	DINT	L#30 000	Límite superior de velocidad de parada
92.0	CTRL_TYPE	DINT	L#1	Modo de control (1 - 4)
99.0	REFPT_SPD	BOOL	TRUE	Velocidad inicial de búsqueda del punto de referencia: • 0 = Marcha rápida • 1 = Marcha lenta
99.1	EI_TYPE	BOOL	FALSE	• FM 351: 0 (no utilizado) • FM 451: Entrada de habilitación: – 0 = Controlada por nivel – 1 = Controlada por flanco
100.0	CHGDIF_P	DINT	L#5000	Distancia de cambio de velocidad +
104.0	CHGDIF_M	DINT	L#5000	Distancia de cambio de velocidad -
108.0	CUTDIF_P	DINT	L#2000	Distancia de desconexión +
112.0	CUTDIF_M	DINT	L#2000	Distancia de desconexión -
Tabla de incrementos 1				
120.0	TRGL1.TRG[1]	DINT	L#0	Número de incremento 1
. . .	. . .	. . .	. . .	. . .
316.0	TRGL1.TRG[50]	DINT	L#0	Número de incremento 50
Tabla de incrementos 2				
320.0	TRGL2.TRG[51]	DINT	L#0	Número de incremento 51
. . .	. . .	. . .	. . .	. . .
516.0	TRGL2.TRG[100]	DINT	L#0	Número de incremento 100

Bloque de datos de diagnóstico

Tarea

El DB de diagnóstico es el archivo de datos para la instrucción ABS_DIAG (Página 7422) o ABS_DIAG_451 (Página 7426). Contiene el búfer de diagnóstico del módulo acondicionado por esta instrucción.

Estructura

DB de diagnóstico
Dirección del módulo
Datos internos
Estado de la petición
Bit de inicio
Búfer de diagnóstico acondicionado

Contenido

La siguiente tabla muestra el contenido del DB de diagnóstico.

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
0.0	MOD_ADDR	INT	0	Dirección del módulo
256.0	JOB_ERR	INT	0	Número del error de comunicación
258.0	JOBBUSY	BOOL	FALSE	1 = Petición activa
258.1	DIAGRD_EN	BOOL	FALSE	1 = Leer el búfer de diagnóstico necesariamente
260.0	DIAG_CNT	INT	0	Número de entradas válidas de la lista
262.0	DIAG[1]	STRUCT		Entrada más reciente de datos de diagnóstico
272.0	DIAG[2]	STRUCT		Segunda entrada de datos de diagnóstico
282.0	DIAG[3]	STRUCT		Tercera entrada de datos de diagnóstico
292.0	DIAG[4]	STRUCT		Cuarta entrada de datos de diagnóstico
302.0	DIAG[5]	STRUCT		Quinta entrada de datos de diagnóstico
312.0	DIAG[6]	STRUCT		Sexta entrada de datos de diagnóstico
322.0	DIAG[7]	STRUCT		Séptima entrada de datos de diagnóstico
332.0	DIAG[8]	STRUCT		Octava entrada de datos de diagnóstico
342.0	DIAG[9]	STRUCT		Novena entrada de datos de diagnóstico

Estructura de la entrada de diagnóstico

La siguiente tabla muestra la estructura de una entrada de diagnóstico DIAG[n].

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
+0.0	STATE	BOOL	FALSE	0 = Evento saliente 1 = Evento entrante
+0.1	INTF	BOOL	FALSE	1 = Error interno
+0.2	EXTF	BOOL	FALSE	1 = Error externo

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
+2.0	FCL	INT	0	Clase de error: 1: fallo operativo 2: error de operador 4: error de datos 5: error de datos máquina 6: error de la tabla de incrementos 15: avisos 128: error de diagnóstico
+4.0	FNO	INT	0	Número de error
+6.0	CH_NO	INT	0	Número de canal
+8.0	TRG_NO	INT	0	Número de incremento

Gestión de peticiones (ABS_CTRL)

Peticiones

El intercambio de datos con el módulo a partir de las señales de control y de respuesta se realiza mediante peticiones.

Para iniciar una petición active el bit de inicio correspondiente del DB de canal y, en el caso de las peticiones de escritura, proporcione además los datos correspondientes. Llame entonces la instrucción ABS_CTRL (Página 7420) para ejecutar la petición.

Si se utiliza el FM 351 de forma centralizada, una petición de lectura precisa exactamente un ciclo. Si se utiliza el FM 351 de forma descentralizada, una petición de lectura puede precisar varios ciclos.

A causa de los acuses necesarios del módulo, una petición de escritura precisa por lo menos 3 llamadas o ciclos OB.

En cuanto se ha procesado por completo una petición, se desactiva el bit de inicio. Con la siguiente llamada de la instrucción se determina y ejecuta la petición siguiente.

Además del bit de inicio con la terminación _EN de "enable", para cada petición hay un bit de finalización y otro de error. Éstos tienen en el nombre la terminación _D de "done" o _ERR de "error". La instrucción ABS_CTRL actualiza los bits de finalización y error al terminar de procesar una petición. Dichos bits deben ponerse a 0 tras la evaluación o antes del inicio de una petición.

Cuando se activa el bit JOBRESET antes del procesamiento de las peticiones pendientes, se desactivan todos los bits de finalización y error. Seguidamente, el bit JOBRESET se vuelve a poner a 0.

Selectores de función

Los selectores de función activan y desactivan estados del canal. Una petición de escritura de los selectores de función sólo se ejecuta si cambia una posición del selector. La posición del selector de función se mantiene tras ejecutar la petición.

Los selectores de función y las peticiones se pueden utilizar simultáneamente en una llamada de la instrucción ABS_CTRL.

Al igual que ocurre con las tareas, para los selectores de función existen bits de inicio con la terminación _ON/_OFF, bits de finalización con la terminación _D y bits de error con la terminación _ERR.

Para poder evaluar los bits de finalización y error de los selectores de función, dichos bits deben ponerse a 0 antes de iniciar una petición para modificar un selector de función.

Orden de procesamiento de peticiones

Es posible lanzar simultáneamente varias peticiones. Si no hay peticiones activas, la gestión de peticiones de la instrucción ABS_CTRL busca a partir de la petición MDWR_EN para comprobar si se han activado los bits de disparo o si se han realizado modificaciones en los selectores de función. Si se encuentra una petición, se procesa. Si la petición está concluida, la gestión de peticiones busca la siguiente petición para ser procesada. Cuando se llega a la última petición ENCVAL_EN, se vuelve a revisar en la petición MDWR_EN. Esta búsqueda se repite hasta que se procesen todas las peticiones.

Las peticiones se procesan siguiendo el siguiente orden lógico según consideraciones tecnológicas:

Orden	Dirección en el DB de canal	Parámetro	Descripción	Desactivado por
Peticiones de escritura				
1	35.0	MDWR_EN	Escribir datos máquina	Instrucción
2	35.1 35.2 35.3 36.4	MD_EN DELDIST_EN AVALREM_EN DELDIAG_EN	Activar datos máquina Borrar distancia residual Cancelar preselección del valor real Borrar búfer de diagnóstico	Instrucción
3	35.4	TRGL1WR_EN	Escribir tabla de incrementos 1	Instrucción
4	35.5	TRGL2WR_EN	Escribir tabla de incrementos 2	Instrucción
5	35.6	REFPT_EN	Definir punto de referencia	Instrucción
6	 34.0 34.1 34.2	Selectores de función: PLOOP_ON MLOOP_ON EI_OFF	 Aproximación en bucle en sentido positivo Aproximación en bucle en sentido negativo No evaluar entrada de habilitación	Programa de usuario
7	35.7	AVAL_EN	Ajustar valor real	Instrucción
10	36.2	TRG252_254_EN	Escribir incremento del número de incremento 254	Instrucción
11	36.3	TRG255_EN	Escribir incremento del número de incremento 255	Instrucción
Peticiones de lectura				
12	36.5	MDRD_EN	Leer datos máquina	Instrucción
13	36.6	TRGL1RD_EN	Leer tabla de incrementos 1	Instrucción
14	36.7	TRGL2RD_EN	Leer tabla de incrementos 2	Instrucción

Orden	Dirección en el DB de canal	Parámetro	Descripción	Desactivado por
16	37.1	ACTSPD_EN	Leer velocidad real, distancia residual e incremento actual	Instrucción
17	37.2	ENCVAL_EN	Leer datos de encóder	Instrucción

Esta secuencia permite lanzar por completo un posicionamiento con un juego de peticiones y señales de control. Las peticiones se extienden desde la escritura y activación de los datos máquina hasta la escritura de incrementos para los modos incrementales, pasando por el ajuste de la entrada externa de habilitación.

Peticiones durante un posicionamiento en curso

Las peticiones de escritura que aparecen en la tabla siguiente se retienen hasta finalizar el posicionamiento cuando se inician durante un posicionamiento y no se ejecutan hasta la siguiente llamada de la instrucción.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
34.0	PLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido positivo
34.1	MLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido negativo
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina
35.2	DELDIST_EN	BOOL	FALSE	1 = Borrar distancia residual
35.3	AVALREM_EN	BOOL	FALSE	1 = Cancelar preselección del valor real
35.6	REFPT_EN	BOOL	FALSE	1 = Definir coordenada del punto de referencia
35.7	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.4	DELDIAG_EN	BOOL	FALSE	1 = Borrar búfer de diagnóstico

Estado de la petición

El estado de procesamiento de la petición se lee en el valor de retorno RETVAL y en el bit de actividad JOBBUSY del DB de canal. El estado de una petición concreta se puede evaluar por medio del bit de inicio, finalización y error de dicha petición.

	RETVAL	JOBBUSY	Bit de inicio _EN	Bit de finalización _D	Bit de error _ERR
Petición activa	1	1	1	0	0
La petición ha finalizado sin errores	0	0	0	1	0

	RETVAL	JOBBUSY	Bit de inicio _EN	Bit de finalización _D	Bit de error _ERR
La petición ha finalizado con errores	-1	0	0	1	1
Petición de escritura cancelada	-1	0	0	0	1

Gestión de peticiones (ABS_CTRL_451)

Peticiones

El intercambio de datos con el módulo a partir de las señales de control y de respuesta se realiza mediante peticiones.

Para iniciar una petición active el bit de inicio correspondiente del DB de canal y, en el caso de las peticiones de escritura, proporcione además los datos correspondientes. Llame entonces la instrucción ABS_CTRL_451 (Página 7424) para ejecutar la petición.

A causa de los acuses necesarios del módulo, una petición de escritura precisa por lo menos 3 llamadas o ciclos OB. Una petición de lectura se ejecuta de inmediato.

En cuanto se ha procesado por completo una petición, se desactiva el bit de inicio. Con la siguiente llamada de la instrucción se determina y ejecuta la petición siguiente.

Además del bit de inicio con la terminación _EN de "enable", para cada petición hay un bit de finalización y otro de error. Éstos tienen en el nombre la terminación _D de "done" o _ERR de "error". La instrucción ABS_CTRL_451 actualiza los bits de finalización y error al terminar de procesar una petición. Dichos bits deben ponerse a 0 tras la evaluación o antes del inicio de una petición.

Cuando se activa el bit JOBRESET antes del procesamiento de las peticiones pendientes, se desactivan todos los bits de finalización y error. Seguidamente, el bit JOBRESET se vuelve a poner a 0.

Selectores de función

Los selectores de función activan y desactivan estados del canal. Una petición de escritura de los selectores de función sólo se ejecuta si hay un cambio en la posición del selector. La posición del selector de función se mantiene tras ejecutar la petición.

Los selectores de función y las peticiones se pueden emplear de forma simultánea en una llamada de la instrucción ABS_CTRL_451.

Al igual que ocurre con las tareas, para los selectores de función hay bits de inicio con la terminación _ON/_OFF, bits de finalización con la terminación _D y bits de error con la terminación _ERR.

Para poder evaluar los bits de finalización y error de los selectores de función, dichos bits deben ponerse a 0 antes de iniciar una petición para modificar un selector de función.

Orden de procesamiento de peticiones

Es posible iniciar simultáneamente varias peticiones. Cuando no hay peticiones activas, la gestión de peticiones de la instrucción ABS_CTRL_451 busca a partir de la petición MDWR_EN para comprobar si se han activado los bits de inicio o si se han realizado modificaciones en los selectores de función. Si se encuentra una petición, se procesa. Si la petición está concluida, la gestión de peticiones busca la siguiente petición para ser procesada. Cuando se llega a la última petición ENCVAL_EN, se vuelve a revisar en la petición MDWR_EN. Esta búsqueda se repite hasta que se procesen todas las peticiones.

Las peticiones se procesan siguiendo el siguiente orden lógico según consideraciones tecnológicas:

Orden	Dirección en el DB de canal	Parámetro	Descripción	Desactivado por
Peticiones de escritura				
1	35.0	MDWR_EN	Escribir datos máquina	Instrucción
2	35.1 35.2 35.3 36.4	MD_EN DELDIST_EN AVALREM_EN DELDIAG_EN	Activar datos máquina Borrar distancia residual Cancelar preselección de valor real Borrar búfer de diagnóstico	Instrucción
3	35.4	TRGL1WR_EN	Escribir tabla de incrementos 1	Instrucción
4	35.5	TRGL2WR_EN	Escribir tabla de incrementos 2	Instrucción
5	35.6	REFPT_EN	Definir punto de referencia	Instrucción
6	 34.0 34.1 34.2 34.3 34.4	Selectores de función: PLOOP_ON MLOOP_ON EI_OFF EDGE_ON MSR_ON	 Aproximación en bucle en sentido positivo Aproximación en bucle en sentido negativo No evaluar entrada de habilitación Captura de bordes activada Medición de longitud activada	Programa de usuario
7	35.7	AVAL_EN	Ajustar valor real	Instrucción
8	36.0	FVAL_EN	Ajustar valor real al vuelo	Instrucción
9	36.1	ZOFF_EN	Definir decalaje de señal cero	Instrucción
10	36.2	TRG252_254_EN	Escribir incremento del número de incremento 252/254	Instrucción
11	36.3	TRG255_EN	Escribir incremento del número de incremento 255	Instrucción
Peticiones de lectura				
12	36.5	MDRD_EN	Leer datos máquina	Instrucción
13	36.6	TRGL1RD_EN	Leer tabla de incrementos 1	Instrucción
14	36.7	TRGL2RD_EN	Leer tabla de incrementos 2	Instrucción
15	37.0	MSRRD_EN	Leer valores medidos	Instrucción
16	37.1	ACTSPD_EN	Leer velocidad real, distancia residual e incremento actual	Instrucción
17	37.2	ENCVAL_EN	Leer datos de encóder	Instrucción

Esta secuencia permite lanzar por completo un posicionamiento con un juego de peticiones y señales de control. Las peticiones se extienden desde la escritura y activación de los datos

máquina hasta la escritura de incrementos para los modos incrementales, pasando por el ajuste de la entrada externa de habilitación.

Peticiones durante un posicionamiento en curso

Las peticiones de escritura que aparecen en la tabla siguiente se retienen hasta finalizar el posicionamiento cuando se inician durante un posicionamiento y no se ejecutan hasta la siguiente llamada de la instrucción.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
34.0	PLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido positivo
34.1	MLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido negativo
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación
34.3	EDGE_ON	BOOL	FALSE	1 = Captura de bordes activada
34.4	MSR_ON	BOOL	FALSE	1 = Medición de longitud activada
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina
35.2	DELDIST_EN	BOOL	FALSE	1 = Borrar distancia residual
35.3	AVALREM_EN	BOOL	FALSE	1 = Cancelar preselección de valor real
35.6	REFPT_EN	BOOL	FALSE	1 = Definir coordenada del punto de referencia
35.7	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.1	ZOFF_EN	BOOL	FALSE	1 = Activar decalaje de señal cero
36.4	DELDIAG_EN	BOOL	FALSE	1 = Borrar búfer de diagnóstico

Estado de la petición

El estado de procesamiento de la petición se lee en el valor de retorno RET_VAL y en el bit de actividad JOBBUSY del DB de canal. El estado de una petición concreta se puede evaluar por medio del bit de inicio, finalización y error de dicha petición.

	RET_VAL	JOBBUSY	Bit de inicio _EN	Bit de finalización _D	Bit de error _ERR
Petición activa	1	1	1	0	0
La petición ha finalizado sin errores	0	0	0	1	0
La petición ha finalizado con errores	-1	0	0	1	1
Petición de escritura cancelada	-1	0	0	0	1

Clases de error

Clase de error 1: fallo operativo

Descripción

Los fallos operativos se detectan de forma asíncrona respecto a un control. Los fallos operativos hacen que se cancele el posicionamiento a excepción del número de error 9. El número de error 9 hace que se desactive el posicionamiento.

N.º	Significado		Alarma de diagnóstico
1	**Rebase del final de carrera de software Inicio**		sí
	Causa	El valor real se encuentra fuera de la zona de trabajo.	
2	**Rebase del final de carrera de software Fin**		sí
	Causa	El valor real se encuentra fuera de la zona de trabajo.	
3	**Rebase del inicio de la zona de desplazamiento**		sí
	Causa	Se ha rebasado el límite de la zona de desplazamiento (las coordenadas de los límites de la zona de desplazamiento son parte de la zona de desplazamiento).	
4	**Rebase del fin de la zona de desplazamiento**		sí
	Causa	Se ha rebasado el límite de la zona de desplazamiento (las coordenadas de los límites de la zona de desplazamiento son parte de la zona de desplazamiento).	
5	**Error en la aproximación a destino**		sí
	Causa	No se ha alcanzado el rango de destino dentro del tiempo de vigilancia.	
6	**Salir de la zona de parada**		sí
	Causa	El valor real se encuentra fuera de la zona de parada.	
7	**Regeneración**		sí
	Causa	Cambio del valor real > ½ zona de parada en sentido equivocado.	
8	**Cambio de valor inexistente o insuficiente**		sí
	Causa	No hay ninguna modificación del valor real o hay una modificación en sentido contrario al teórico dentro del tiempo de vigilancia.	
9	**Rebase del destino (FM 451)**		sí
	Causa	Se ha rebasado el destino al "Ajustar valor real al vuelo".	
10	**Rebase de la zona de destino**		sí
	Causa	Se ha rebasado la zona de destino después de la aproximación a destino.	
11	**Se ha conmutado erróneamente el punto de cambio de velocidad**		sí
	Causa	El eje oscila en el punto de cambio de velocidad.	
12	**Se ha conmutado erróneamente el punto de desconexión**		sí
	Causa	El eje oscila en el punto de desconexión/punto de inversión.	
13	**Se ha conmutado erróneamente el inicio de la zona de destino**		sí
	Causa	El eje oscila en la zona de destino.	
14	**Modificación superior a la mitad del rango del eje rotativo**		sí

N.º	Significado		Alarma de diagnóstico
	Causa	Excesiva velocidad/frecuencia o hay saltos erróneos del valor real.	
15	Modificación superior al rango del eje rotativo		sí
	Causa	Excesiva velocidad/frecuencia o hay saltos erróneos del valor real.	
16	No se ha transmitido el incremento del número de incremento 252 (FM 451)		sí
	Causa	No se ha transmitido el incremento.	
17	No se puede llegar al incremento del número de incremento 252 (FM 451)		sí
	Causa	La distancia entre la posición real actual y el incremento especificado es inferior a la distancia de cambio de velocidad o de desconexión.	
18	Incremento erróneo del número de incremento 252 (FM 451)		sí
	Causa	El incremento se encuentra fuera de la zona de trabajo.	

Clase de error 2: error de operador

Descripción

Los errores de operador se detectan al modificar las señales de control en el área de datos de usuario. Los errores de operador provocan la desactivación del posicionamiento.

N.º	Significado		Alarma de diagnóstico
1	Modo de operación no admisible		no
	Causa	El modo de operación seleccionado no es admisible.	
3	Petición de interfaz no admisible		no
	Causa	La señal seleccionada no es admisible con este modo de operación.	
4	Parámetro erróneo de modo de operación		no
	Causa	• Con el modo de operación "Jog", la especificación de velocidad difiere de la velocidad rápida o velocidad lenta. • Con el modo de operación "Modo incremental", el incremento difiere de 1 a 100 o difiere de 252, 254 y 255.	
5	No hay habilitación de inicio		no
	Causa	No hay habilitación al iniciar.	
7	El destino / la zona de destino está fuera de la zona de trabajo		no
	Causa	El destino especificado o calculado está fuera de los finales de carrera de software.	
8	Eje no parametrizado		no
	Causa	Se han parametrizado datos máquina erróneos o no se han parametrizado datos para el eje.	
9	El eje no está sincronizado.		no
	Causa	El modo de operación "Modo incremental" sólo es posible con un eje ya sincronizado.	
10	No es posible posicionar el destino / recorrido		no
	Causa	La distancia entre la posición real actual y el destino especificado es inferior a la distancia de desconexión.	

N.º	Significado		Alarma de diagnóstico
17	**No es posible buscar el punto de referencia**		no
	Causa	Se ha conectado un encóder SSI.	
18	**No es posible el modo incremental relativo o absoluto**		no
	Causa	El incremento no es válido.	
19	**La distancia de desconexión no debe ser superior a la ½ de la zona de destino con el número de incremento 255**		no
	Causa	La distancia de desconexión del incremento 255 es menor que la mitad de la zona de destino.	
20	**El desplazamiento en el sentido especificado no es admisible**		no
	Causa	La distancia del final de carrera por software es insuficiente.	

Clase de error 4: error de datos

Descripción

Los errores de datos se detectan de forma síncrona respecto a un control. Los errores de datos no producen ninguna reacción de error.

N.º	Significado		Alarma de diagnóstico
6	**Incremento especificado excesivo**		no
	Causa	El valor está fuera de ±100 m ó ±1000 m. El recorrido / destino no debe ser superior a la zona de desplazamiento. Con ejes rotativos, la coordenada debe ser ≥ 0 y menor que el fin del eje rotativo.	
10	**Decalaje de señal cero erróneo**		no
	Causa	El decalaje de señal cero es mayor que ± 100 m o ± 1.000 m. Los finales de carrera de software, tras ajustar el decalaje de señal cero, se encuentran fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: el valor del decalaje de señal cero es mayor que el final del eje rotativo.	
11	**Preajuste erróneo del valor real**		no
	Causa	Eje lineal: la coordenada se encuentra fuera del final de carrera de software actual (que puede estar desplazado). Eje rotativo: la coordenada es < 0 ó mayor que el final del eje rotativo.	
12	**Punto de referencia erróneo**		no
	Causa	Eje lineal: la coordenada se encuentra fuera del final de carrera de software actual (que puede estar desplazado). Eje rotativo: la coordenada es < 0 ó mayor que el final del eje rotativo.	
20	**Activar datos máquina no admisible**		no
	Causa	En el módulo no hay datos máquina nuevos (sin errores).	
27	**Ajuste codificado en bits no permitido**		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
29	**Codificación en bits no válida**		no

N.º	Significado		Alarma de diagnóstico
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
34	**No es posible deshacer la activación del valor real**		no
	Causa	Con un encóder SSI y un eje lineal, tras la ejecución del ajuste el valor real de posición se encontraría fuera de la zona de trabajo.	
36	**Distancia errónea de cambio de velocidad en el número de incremento 255**		no
	Causa	El valor se encuentra fuera del rango numérico admisible de ± 100 m ó ± 1.000 m. Con ejes rotativos, la coordenada debe ser ≥ 0 y menor que el fin del eje rotativo.	
37	**Distancia errónea de desconexión en el número de incremento 255**		no
	Causa	El valor se encuentra fuera del rango numérico admisible de ± 100 m ó ± 1.000 m. La distancia de desconexión debe ser menor que la distancia de cambio de velocidad.	
107	**Eje no parametrizado**		no
	Causa	No hay datos máquina en el eje o no se han activado.	
108	**El eje no está sincronizado.**		no
	Causa	Se ha lanzado una de las peticiones "Ajustar valor real" o "Ajustar valor real al vuelo" a pesar de que el eje no está sincronizado.	

Clase de error 5: error de datos máquina

Descripción

La alarma de diagnóstico sólo se dispara con un bloque de datos de sistema (SDB) con errores. Los errores de datos máquina no producen ninguna reacción de error.

N.º	Significado		Alarma de diagnóstico
5	**Error en el ajuste de la alarma de proceso**		sí
	Causa	Ha intentado seleccionar una alarma de proceso que el módulo no soporta.	
6	**Distancia mínima entre bordes errónea (FM 451)**		sí
	Causa	Ha introducido como distancia mínima entre bordes un valor < 0 ó > 10^9 µm.	
7	**Sistema de medida erróneo**		sí
	Causa	El valor del sistema de medición se encuentra fuera del rango admisible de 1 a 4 y 6.	
8	**Tipo de eje erróneo**		sí
	Causa	No se ha indicado ni 0 ni 1 como tipo de eje.	
9	**Final erróneo del eje rotativo**		sí
	Causa	El valor del final del eje rotativo se encuentra fuera del rango admisible de 1 a 10^9 µm o de 1 a 10^8 µm (según sea la resolución).	
10	**Tipo de encóder erróneo**		sí
	Causa	El valor del tipo de encóder se encuentra fuera del rango admisible de 1 a 4.	
11	**Recorrido por vuelta erróneo**		sí

N.º	Significado		Alarma de diagnóstico
	Causa	El valor del recorrido/vuelta se encuentra fuera del rango admisible de 1 a 10^9 µm (independientemente de la resolución).	
13	Incrementos por vuelta erróneos		sí
14	Número de vueltas erróneo		sí
15	Velocidad de transferencia errónea		sí
	Causa	Para la velocidad de transferencia se ha indicado un valor fuera del rango admisible de 0 a 3.	
16	Coordenada del punto de referencia errónea		sí
	Causa	La coordenada se encuentra fuera del rango de -100 m a +100 m o de -1.000 m a +1.000 m, respectivamente (según la resolución). Eje lineal: la coordenada se encuentra fuera de la zona de trabajo. Eje rotativo: la coordenada es mayor que el final del eje rotativo ó < 0.	
17	Reajuste erróneo del encóder absoluto		sí
	Causa	Encóder SSI: el valor de reajuste del encóder absoluto no se encuentra dentro del rango del encóder (incrementos por vuelta x número de vueltas - 1).	
18	Modo erróneo de búsqueda del punto de referencia		sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0, 1, 2 y 3.	
19	Sentido de contaje erróneo		sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0 y 1.	
20	No es posible vigilar el hardware		sí
	Causa	Ha puesto en "FALSE" la monitorización de errores de telegrama en el DB de parámetros. No es posible vigilar los impulsos erróneos con el encóder empleado. Desactive el parámetro MON_PULSE.	
21	Final erróneo de carrera de software Inicio		sí
	Causa	Eje lineal: el final de carrera de software Inicio se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución). Eje lineal: El inicio del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es menor que -100 m ó -1000 m (según la resolución).	
22	Final erróneo de carrera de software Fin		sí
	Causa	Eje lineal: el final de carrera de software Fin se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución) o es menor que el final de carrera de software Inicio. Eje lineal: el final de carrera de software Fin (dado el caso, incluido un decalaje de señal cero presente) es mayor que +100 m ó +1000 m (según la resolución).	
23	Velocidad máxima errónea		sí
	Causa	Los datos que no aparecen en la lista en el DB de parámetros tienen que ser 0.	
24	Zona de destino errónea		sí
	Causa	Eje lineal: rango de 0 a 100 m ó 1000 m, según la resolución. Eje rotativo: rango superior al final del eje rotativo.	
25	Tiempo de vigilancia erróneo		sí

N.º	Significado		Alarma de diagnóstico
	Causa	El valor del tiempo de vigilancia se encuentra fuera del rango admisible de 0 a 100 000 ms.	
26	Zona de parada errónea		sí
	Causa	Eje lineal: rango de 0 a 100 m ó 1000 m, según la resolución. Eje rotativo: rango superior al final del eje rotativo.	
127	Velocidad de parada errónea		sí
	Causa	El valor de la velocidad de parada se encuentra fuera del rango admisible de 0 a 100 000 µm/min.	
128	Modo de control erróneo		sí
	Causa	Para el modo de control se ha indicado un valor fuera del rango admisible de 1 a 4.	
129	Velocidad inicial errónea para la búsqueda del punto de referencia		sí
	Causa	No se ha indicado no 0 ni 1 como velocidad inicial.	
130	Distancia errónea de cambio de velocidad en sentido +		sí
	Causa	Eje lineal: rango de 0 a 100 m ó 1000 m (según la resolución). Eje rotativo: rango superior al final del eje rotativo y menor que ½ zona de destino.	
131	Distancia errónea de cambio de velocidad en sentido –		sí
	Causa	Eje lineal: rango de 0 a 100 m ó 1000 m (según la resolución). Eje rotativo: rango superior al final del eje rotativo y menor que ½ zona de destino.	
132	Distancia errónea de desconexión en sentido +		sí
	Causa	La distancia de desconexión es superior a la distancia de cambio de velocidad +, inferior a la mitad de la zona de destino o está fuera del rango permitido de 0 a 100 m ó 1.000 m (según la resolución).	
133	Distancia errónea de desconexión en sentido –		sí
	Causa	La distancia de desconexión es superior a la distancia de cambio de velocidad –, inferior a la mitad de la zona de destino o está fuera del rango permitido de 0 a 100 m ó 1.000 m (según la resolución).	
200	Resolución errónea		sí
	Causa	Ha indicado una resolución < 0,1 µm/impulso o >1000 µm/impulso. Ha indicado un recorrido/vuelta y un número de impulsos/vuelta que dan una resolución < 0,1 o > 1000.	
201	El encóder no se adapta a la zona de trabajo / rango del eje rotativo		sí
	Causa	Encóder SSI y eje rotativo: el encóder no cubre exactamente el rango del eje rotativo. Eje lineal: el encóder no cubre por lo menos la zona de trabajo (finales de carrera de software incluidos).	

Clase de error 6: error de la tabla de incrementos

Descripción

Los errores de la tabla de incrementos no producen ninguna reacción de error.

N.º	Significado		Alarma de diagnóstico
6	**Excesivo incremento especificado en la tabla de incrementos**		no
	Causa	El valor se encuentra fuera de ± 100 m ó ± 1.000 m. El recorrido / destino no debe ser superior a la zona de desplazamiento. Con ejes rotativos, la coordenada debe ser ≥ 0 y menor que el fin del eje rotativo.	

Clase de error 15: avisos

Descripción

Los avisos no producen ninguna reacción de error.

N.º	Significado		Alarma de diagnóstico
1	**Inicio de parametrización**		no
	Causa	El módulo ha detectado una parametrización por medio de un bloque de datos de sistema.	
2	**Final de la parametrización**		no
	Causa	El módulo ha procesado sin problemas la parametrización mediante un bloque de datos de sistema.	
11	**Distancia insuficiente del punto de cambio de velocidad**		no
	Causa	No se pueden mantener los tiempos de reacción del hardware porque la distancia entre los puntos de conmutación es insuficiente.	
12	**Distancia del punto de inversión insuficiente**		no
	Causa	No se pueden mantener los tiempos de reacción del hardware porque la distancia entre los puntos de conmutación es insuficiente.	
14	**Distancia insuficiente del punto de desconexión**		no
	Causa	No se pueden mantener los tiempos de reacción del hardware porque la distancia entre los puntos de conmutación es insuficiente.	
15	**Distancia insuficiente del inicio de la zona de destino**		no
	Causa	No se pueden mantener los tiempos de reacción del hardware porque la distancia entre los puntos de conmutación es insuficiente.	

Clase de error 128: error de diagnóstico

N.º	Significado		Alarma de diagnóstico
4	**Falta tensión auxiliar externa**		sí
	Causa	• La tensión auxiliar externa de 24 V no se ha conectado o ha fallado • El fusible del módulo está defectuoso • Subtensión • Rotura de hilo en masa • Cortocircuito (p. ej. en el encóder conectado)	
	Efectos	• Se cancela el posicionamiento en todos los canales • Desactivación de las salidas • Borrado de la sincronización en los encóders incrementales cuando falle la tensión auxiliar de alimentación del encóder • Los parámetros no han sido asignados al módulo. • Borrado de la habilitación de inicio	
	Solución	Comprobar que la conexión de 24 V sea correcta (si la conexión de 24 V es correcta, significa que el módulo está averiado)	
5	**Falta conector frontal (FM 451)**		sí
	Causa	No se ha insertado el conector frontal del módulo de posicionamiento.	
	Efectos	• Falta tensión auxiliar externa de 24 V • El módulo no está operativo	
	Solución	Enchufar el conector frontal en el módulo de posicionamiento	
51	**Ha saltado la vigilancia de tiempo (watchdog)**		sí
	Causa	• Interferencias fuertes en el módulo • Error en el módulo	
	Efectos	• El módulo se resetea • Desconexión de todas las salidas • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, éste vuelve a estar operativo • El módulo notifica que ha transcurrido la vigilancia del tiempo con "entrante" y "saliente"	
	Solución	• Eliminar las interferencias • Consulte al departamento comercial correspondiente cuando las circunstancias exactas que han dado lugar al error sean de mayor entidad. • Sustituir el módulo	
144	**Rotura de hilo en el encóder**		sí
	Causa	• El cable del encóder no está enchufado o está cortado • Encóder sin señales de cuadratura • Asignación de pines incorrecta • Longitud de cable excesiva • Cortocircuito de las señales del encóder • Error de flanco de las señales del encóder • Se ha sobrepasado la frecuencia máxima de la entrada del encóder • Corte de la alimentación del encóder	

N.º	Significado		Alarma de diagnóstico
	Efectos	• Se interrumpe el posicionamiento • Desconexión de todas las salidas • Borrado de la sincronización, con encóders incrementales • Borrado de la habilitación de inicio	
	Solución	• Comprobar el cable del encóder • Guardar la especificación del encóder • La vigilancia puede desactivarse de forma provisional con la interfaz de parametrización bajo la responsabilidad del operador. • Guardar los datos técnicos del módulo	
145	**Error de telegrama del encóder absoluto**		sí
	Causa	El tráfico de telegramas entre el módulo y el encóder absoluto (SSI) tiene errores o está interrumpido: • El cable del encóder no está enchufado o está cortado • Tipo de encóder erróneo • Encóder mal configurado (encóders programables) • Longitud de telegrama mal indicada • El encóder proporciona valores erróneos (encóder defectuoso) • Interferencia en el cable del sistema de medición • Velocidad de transferencia escogida demasiado alta • Tiempo monoestable del encóder superior a 64 µs	
	Efectos	• Se interrumpe el posicionamiento • Desactivación de las salidas • Borrado de la habilitación de inicio	
	Solución	• Comprobar el cable del encóder • Comprobar el encóder • Comprobar el tráfico de telegramas entre el encóder y el módulo	
146	**Impulsos erróneos del encóder incremental**		sí
	Causa	• La vigilancia del encóder ha detectado impulsos erróneos • El número de incrementos por vuelta está mal introducido • Encóder averiado: no proporciona el número de impulsos indicado • señal cero errónea o inexistente • Interferencias en el cable del encóder	
	Efectos	• Se interrumpe el posicionamiento • Desactivación de las salidas • Borrado de la habilitación de inicio	
	Solución	• Introducir correctamente el número de incrementos/vuelta (cuadro de diálogo de parametrización) • Comprobar el encóder y su cable • Cumplir las normas de apantallamiento y de conexión a tierra • La vigilancia puede desactivarse de forma provisional con la interfaz de parametrización bajo la responsabilidad del operador.	

Control por levas FM x52

CAM_INIT

Descripción

La instrucción CAM_INIT inicializa los datos siguientes en el DB de canal:

- Las señales de control
- Las señales de respuesta
- Los bits de inicio, los bits de finalización y los bits de error de las tareas
- Los selectores de función y sus bits de finalización y de error
- La gestión de peticiones y el búfer interno de la instrucción CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457)
- La gestión de peticiones y el búfer interno de la instrucción CAM_MSRM_452 (Página 7460) (sólo FM 452)

Llamada

La instrucción tiene que ejecutarse tras un arranque, es decir, después de conectar la fuente de alimentación del módulo o de la CPU. Por ello insértela, por ejemplo, en el OB de rearranque OB 100 y en el OB de extracción/inserción OB 83, o bien llámela en la fase de inicialización del programa de usuario. Antes es preciso escribir en el OB 100 la dirección del módulo en el DB. De este modo queda asegurado que tras un rearranque completo de la CPU o un arranque del módulo su programa de usuario no accederá a datos antiguos.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de canal

Valores de retorno

La instrucción no proporciona ningún valor de retorno.

CAM_CTRL

Descripción

La instrucción CAM_CTRL permite leer datos operativos procedentes del módulo, inicializar el módulo y controlar su funcionamiento. Para ello se utilizan señales de control, señales de respuesta, selectores de función y peticiones de escritura y lectura.

Funcionamiento

Con cada llamada la instrucción ejecuta las actividades siguientes:

- Leer señales de respuesta:
  La instrucción CAM_CTRL lee todas las señales de respuesta del módulo y las transfiere al DB de canal. Como las señales de control y las peticiones solo se procesan al final, las señales de respuesta reflejan el estado del módulo antes de llamar la instrucción.
- Escribir señales de control:
  Las señales de control que se encuentran en el DB de canal se transfieren al módulo. La habilitación del procesamiento de levas, sin embargo, se retiene hasta que se ha activado el inicio de una petición "Definir punto de referencia" o "Escribir datos de leva". La (re-)activación del procesamiento de levas se mantiene retenida durante este intervalo de tiempo.
- Gestión de peticiones (Página 7473):
  La petición siguiente en cada caso se ejecuta a partir de los bits de disparo de petición registrados en el DB de canal.

Llamada

La instrucción tiene que llamarse cíclicamente.

Antes de llamar la instrucción, registre en el DB de canal todos los datos necesarios para la ejecución de las funciones deseadas.

Bloques de datos empleados

- DB de canal (Página 7461):
  El DB de canal es el DB instancia de la instrucción CAM_CTRL. En el DB de canal es preciso que esté registrada la dirección del módulo. .
- DB de parámetros (Página 7469):
  Para leer o escribir datos máquina o datos de leva mediante peticiones se requiere un DB de parámetros cuyo número tiene que estar registrado en el DB de canal. El tamaño del DB de parámetros tiene que ser suficiente para el número de levas existente.

Valores de retorno

La instrucción proporciona los siguientes valores de retorno en el parámetro RETVAL del DB de canal:

RETVAL	RB	Descripción
1	1	Por lo menos 1 petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Errores: Error de datos (DAT_ERR) o Error de comunicación (JOB_ERR)

Arranque

Llame la instrucción CAM_INIT (Página 7453) al arrancar el módulo o la CPU. Para ello se resetean, entre otros, los selectores de función.

La instrucción CAM_CTRL acusa el arranque del módulo. Durante este período, RETVAL y JOBBUSY = 1.

Comportamiento en caso de error

Si en una petición de escritura se han escrito datos erróneos, el módulo devuelve la respuesta DATA_ERR = 1. Cuando en una petición de escritura o de lectura se produce un error en la comunicación con el módulo, la causa del error se deposita en el parámetro JOB_ERR del DB de canal.

- Error en una petición de escritura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D. En todas las peticiones de escritura en cola, el bit de inicio vuelve también a su posición anterior y se activa el bit de error _ERR.
  Las peticiones de lectura pendientes se siguen procesando. Para ello se vuelve a activar JOB_ERR para cada petición.
- Error en una petición de lectura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D.
  Las peticiones de lectura todavía pendientes se siguen procesando. Para ello se vuelve a activar JOB_ERR para cada petición.

CAM_DIAG

Descripción

La instrucción CAM_DIAG lee el búfer de diagnóstico del módulo y lo pone a disposición de un indicador en el sistema de manejo y visualización (M+V) o de una evaluación programada.

Funcionamiento

La instrucción lee el búfer de diagnóstico cuando en él se muestra una nueva entrada por medio de la señal de respuesta DIAG = 1. Tras leer el búfer de diagnóstico el módulo pone DIAG a 0.

Llamada

La instrucción tiene que llamarse cíclicamente. No se permite realizar una llamada adicional en un programa de interrupción controlado por eventos. Para una ejecución completa de la instrucción hacen falta por lo menos 2 llamadas (ciclos).

Bloque de datos empleado

DB de diagnóstico (Página 7471):

El DB de diagnóstico es el DB instancia de la instrucción CAM_DIAG. Es necesario que en el DB de diagnóstico esté registrada la dirección del módulo. Debería efectuarse la entrada en el OB de arranque (OB 100) o bien antes de emplearse por primera vez. La entrada más reciente del búfer de diagnóstico se registra en la estructura DIAG[1] y la más antigua en la estructura DIAG[4].

Valores de retorno

La instrucción proporciona los siguientes valores de retorno en el parámetro RETVAL del DB de diagnóstico:

RETVAL	RB	Descripción
1	1	Petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Error

Peticiones

El búfer de diagnóstico se puede leer aunque no haya ninguna entrada nueva activando el bit de inicio DIAGRD_EN. Tras leer el búfer de diagnóstico el bit de inicio se pone a 0.

Arranque

La instrucción no ejecuta ningún procesamiento del arranque.

Comportamiento en caso de error

En caso de una petición errónea, encontrará la causa del error en el DB de diagnóstico, en el parámetro JOB_ERR.

Clases de error

Los errores que se producen están agrupados en las siguientes clases de error:

- Clase de error 1: fallo operativo (Página 7476)
- Clase de error 4: error de datos (Página 7477)
- Clase de error 5: error de datos máquina (Página 7478)
- Clase de error 7: error de datos de leva (Página 7481)
- Clase de error 15: avisos (Página 7482)
- Clase de error 128: error de diagnóstico (Página 7482)

CAM_CTRL_452

Descripción

La instrucción CAM_CTRL_452 permite leer datos operativos procedentes del módulo, inicializar el módulo y controlar su funcionamiento. Para ello se utilizan señales de control, señales de respuesta, selectores de función y peticiones de escritura y lectura.

Funcionamiento

Con cada llamada la instrucción ejecuta las actividades siguientes:

- Leer señales de respuesta:
  La instrucción CAM_CTRL_452 lee todas las señales de respuesta del módulo y las transfiere al DB de canal. Como las señales de control y las peticiones sólo se procesan al final, las señales de respuesta reflejan el estado del módulo antes de llamar la instrucción.
- Escribir señales de control:
  Las señales de control que se encuentran en el DB de canal se transfieren al módulo. La habilitación del procesamiento de levas, sin embargo, se retiene hasta que se ha activado el inicio de una petición "Definir punto de referencia" o "Escribir datos de leva". La (re-)activación del procesamiento de levas se mantiene retenida durante este intervalo de tiempo.
- Gestión de peticiones (Página 7474):
  La petición siguiente en cada caso se ejecuta a partir de los bits de disparo de petición registrados en el DB de canal.

Llamada

La instrucción tiene que llamarse cíclicamente.

Antes de llamar la instrucción, registre en el DB de canal todos los datos necesarios para la ejecución de las funciones deseadas.

Bloques de datos empleados

- DB de canal (Página 7461):
  En el DB de canal es preciso que esté registrada la dirección del módulo.
- DB de parámetros (Página 7469):
  Para leer o escribir datos máquina o datos de leva mediante peticiones se requiere un DB de parámetros cuyo número tiene que estar registrado en el DB de canal. El tamaño del DB de parámetros tiene que ser suficiente para el número de levas existente.

Parámetros

Parámetros	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de canal
RET_VAL	OUTPUT	INT	Valor de retorno

Valores de retorno

La instrucción proporciona los siguientes valores de retorno en el parámetro RET_VAL del DB de canal:

RET_VAL	RB	Descripción
1	1	Por lo menos 1 petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Errores: Error de datos (DAT_ERR) o Error de comunicación (JOB_ERR)

Arranque

Llame la instrucción CAM_INIT (Página 7453) al arrancar el módulo o la CPU. Para ello se inicializan, entre otros, los selectores de función.

La instrucción CAM_CTRL_452 acusa el arranque del módulo. Durante este período, RET_VAL y JOBBUSY = 1.

Comportamiento en caso de error

Si en una petición de escritura se han escrito datos erróneos, el módulo devuelve la respuesta DATA_ERR = 1. Cuando en una petición de escritura o de lectura se produce un error en la comunicación con el módulo, la causa del error se deposita en el parámetro JOB_ERR del DB de canal.

- Error en una petición de escritura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D. En todas las peticiones de escritura en cola, el bit de inicio vuelve también a su posición anterior y se activa el bit de error _ERR.
  Las peticiones de lectura en cola continúan siendo procesadas. Para ello se vuelve a activar JOB_ERR para cada petición.
- Error en una petición de lectura:
  En el caso de una petición con errores, se desactiva el bit de inicio y se activan el bit de error _ERR y el de finalización _D.
  Las peticiones de lectura todavía pendientes se siguen procesando. Para ello se vuelve a activar JOB_ERR para cada petición.

CAM_DIAG_452

Descripción

La instrucción CAM_DIAG_452 lee el búfer de diagnóstico del módulo y lo pone a disposición de un indicador en el sistema de manejo y visualización (M+V) o de una evaluación programada.

Funcionamiento

La instrucción lee el búfer de diagnóstico cuando en él se muestra una nueva entrada por medio de la señal de respuesta DIAG = 1. Tras leer el búfer de diagnóstico el módulo pone DIAG a 0.

Llamada

La instrucción tiene que llamarse cíclicamente. No se admite una llamada adicional en un OB de alarma. Para una ejecución completa de la instrucción hacen falta por lo menos 2 llamadas (ciclos).

Bloque de datos empleado

DB de diagnóstico (Página 7471):

Es necesario que en el DB de diagnóstico esté registrada la dirección del módulo. La entrada más reciente del búfer de diagnóstico se registra en la estructura DIAG[1] y la más antigua en la estructura DIAG[4].

Parámetros

Parámetros	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de diagnóstico
RET_VAL	OUTPUT	INT	Valor de retorno

Valores de retorno

La instrucción proporciona los siguientes valores de retorno en el parámetro RET_VAL del DB de diagnóstico:

RET_VAL	RB	Descripción
1	1	Petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Error

Peticiones

El búfer de diagnóstico se puede leer aunque no haya ninguna entrada nueva activando el bit de inicio DIAGRD_EN. Tras leer el búfer de diagnóstico el bit de inicio se pone a 0.

Arranque

La instrucción no ejecuta ningún procesamiento del arranque.

Comportamiento en caso de error

En caso de una petición errónea, encontrará la causa del error en el DB de diagnóstico, en el parámetro JOB_ERR.

Clases de error

Los errores que se producen están agrupados en las siguientes clases de error:

- Clase de error 1: error de operador (Página 7476)
- Clase de error 4: error de datos (Página 7477)
- Clase de error 5: error de datos máquina (Página 7478)
- Clase de error 7: error de datos de leva (Página 7481)
- Clase de error 15: avisos (Página 7482)
- Clase de error 128: error de diagnóstico (Página 7482)

CAM_MSRM_452

Descripción

La instrucción CAM_MSRM_452 se utiliza para evaluar inmediatamente datos de una medición de longitud o de una captura de bordes de un FM 452 en el OB de alarma de proceso.

Llamada

La instrucción se llama en un OB de alarma de proceso (p. ej. OB 40).

Bloque de datos empleado

DB de canal (Página 7461):

En el DB de canal es preciso que esté registrada la dirección del módulo. La entrada debe realizarse en el OB de arranque (OB 100).

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del DB de canal
RET_VAL	OUTPUT	INT	Valor de retorno

Valores de retorno

La instrucción proporciona los siguientes valores de retorno en el parámetro RET_VAL del DB de canal:

RET_VAL	RB	Descripción
1	1	Petición activa
0	1	Ninguna petición activa, ningún error
-1	0	Error

Resultados de medición e información de estado

Los resultados de medición y la información de estado se encuentran en el DB de canal:

Dirección	Nombre	Tipo	Valor inicial	Comentario
112.0	BEG_VAL	DINT	L#0	Valor inicial
116.0	END_VAL	DINT	L#0	Valor final
120.0	LEN_VAL	DINT	L#0	Longitud
56.0	JOB_ERR_M	INT	0	Error de comunicación
58.0	JOBBUSY_M	BOOL	FALSE	Petición activa

Arranque

La instrucción no ejecuta ningún procesamiento del arranque.

Comportamiento en caso de error

En caso de una petición errónea, encontrará la causa del error en el DB de canal, en el parámetro JOB_ERR_M.

Más referencias para FM x52 CAM V2

Bloque de datos de canal

Tarea

El DB de canal es la interfaz de datos entre el programa de usuario y el secuenciador electrónico de levas FM x52. Contiene y aplica todos los datos necesarios para el control y el funcionamiento del módulo.

Estructura

El DB de canal se divide en diferentes áreas:

DB de canal
Direcciones* /Conmutador de versión
Señales de control
Señales de respuesta
Selectores de función
Bits de inicio para peticiones de escritura
Bits de inicio para peticiones de lectura
Bits de finalización
Bits de error
Gestión de peticiones para instrucciones
Datos de las peticiones
* La dirección del módulo se puede registrar en la interfaz de parametrización.

Contenido

La siguiente tabla muestra el contenido del DB de canal.

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
Direcciones / Conmutador de versión				
0.0	MOD_ADDR (¡Registrar!)	INT	0	Dirección del módulo
2.0	CH_NO	INT	1	Número de canal (siempre 1)
10.0	PARADBNO	INT	-1	Número del DB de parámetros -1 = DB no existente
12.0	FM_TYPE	BOOL	FALSE	0 = FM 352 hasta V4.0 1 = FM 452 o FM 352 a partir de V5.0
Señales de control				
15.2	DIR_M	BOOL	FALSE	1 = Simulación en el sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Simulación en el sentido positivo
15.4	CAM_EN	BOOL	FALSE	1 = Habilitar el procesamiento de levas
15.5	CNTC0_EN	BOOL	FALSE	1 = Habilitar la función de contaje de la pista de leva de contaje 0
15.6	CNTC1_EN	BOOL	FLASE	1 = Habilitar la función de contaje de la pista de leva de contaje 1
16.0	TRACK_EN	WORD	W#16#0	Habilitar pistas de leva de 0 a 15 Bit 0 = Pista 0

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
Señales de respuesta				
22.2	DIAG	BOOL	FALSE	1 = Búfer de diagnóstico modificado
22.4	DATA_ERR	BOOL	FALSE	1 = Error de datos
22.7	PARA	BOOL	FALSE	1 = Módulo parametrizado
23.4	CAM_ACT	BOOL	FALSE	1 = Procesamiento de levas activo
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado
25.1	MSR_DONE	BOOL	FALSE	1 = Medición de longitud o captura de bordes finalizada
25.2	GO_M	BOOL	FALSE	1 = Eje en sentido negativo
25.3	GO_P	BOOL	FALSE	1 = Eje en sentido positivo
25.4	HYS	BOOL	FALSE	1 = El eje se encuentra dentro del rango de histéresis
25.5	FVAL_DONE	BOOL	FALSE	1 = Ejecutado Ajustar valor real al vuelo
26.0	ACT_POS	DINT	L#0	Posición actual del eje
30.0	TRACK_OUT	DWORD	DW#16#0	Señales de pista actuales de las pistas 0 a 31 Bit 0 = Pista 0
Selectores de función				
34.0	EDGE_ON	BOOL	FALSE	1 = Captura de bordes activada
34.1	SIM_ON	BOOL	FALSE	1 = Simulación activa
34.2	MSR_ON	BOOL	FALSE	1 = Medición de longitud activada
34.3	REFTR_ON	BOOL	FALSE	1 = Redisparar el punto de referencia
34.4	SSW_OFF	BOOL	FALSE	1 = Final de carrera de software desconectado
Bits de disparo de peticiones de escritura				
35.0	MDWR_EN	BOOL	FALSE	1 = Escribir datos máquina
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina
35.2	AVALREM_EN	BOOL	FALSE	1 = Ajustar valor real, Ajustar valor real al vuelo, Cancelar preselección del valor real
35.3	CAM1WR_EN	BOOL	FALSE	1 = Escribir datos de leva 1 (levas de 0 a 15)
35.4	CAM2WR_EN	BOOL	FALSE	1 = Escribir datos de leva 2 (levas de 16 a 31)
35.5	CAM3WR_EN	BOOL	FALSE	1 = Escribir datos de leva 3 (levas de 32 a 47)
35.6	CAM4WR_EN	BOOL	FALSE	1 = Escribir datos de leva 4 (levas de 48 a 63)
35.7	CAM5WR_EN	BOOL	FALSE	1 = Escribir datos de leva 5 (levas de 64 a 79)
36.0	CAM6WR_EN	BOOL	FALSE	1 = Escribir datos de leva 6 (levas de 80 a 95)
36.1	CAM7WR_EN	BOOL	FALSE	1 = Escribir datos de leva 7 (levas de 96 a 111)
36.2	CAM8WR_EN	BOOL	FALSE	1 = Escribir datos de leva 8 (levas de 112 a 127)

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
36.3	REFPT_EN	BOOL	FALSE	1 = Definir coordenadas del punto de referencia
36.4	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.5	FVAL_EN	BOOL	FALSE	1 = Ajustar valor real al vuelo
36.6	ZOFF_EN	BOOL	FALSE	1 = Activar decalaje de señal cero
36.7	CH01CAM_EN	BOOL	FALSE	1 = Escribir ajuste para flanco de leva (1 leva)
37.0	CH16CAM_EN	BOOL	FALSE	1 = Escribir ajuste para un cambio rápido de levas (16 levas)
Bits de disparo de peticiones de lectura				
37.1	MDRD_EN	BOOL	FALSE	1 = Leer datos máquina
37.2	CAM1RD_EN	BOOL	FALSE	1 = Leer datos de leva 1 (Levas de 0 a 15)
37.3	CAM2RD_EN	BOOL	FALSE	1 = Leer datos de leva 2 (Levas de 16 a 31)
37.4	CAM3RD_EN	BOOL	FALSE	1 = Leer datos de leva 3 (Levas de 32 a 47)
37.5	CAM4RD_EN	BOOL	FALSE	1 = Leer datos de leva 4 (Levas de 48 a 63)
37.6	CAM5RD_EN	BOOL	FALSE	1 = Leer datos de leva 5 (Levas de 64 a 79)
37.7	CAM6RD_EN	BOOL	FALSE	1 = Leer datos de leva 6 (Levas de 80 a 95)
38.0	CAM7RD_EN	BOOL	FALSE	1 = Leer datos de leva 7 (Levas de 96 a 111)
38.1	CAM8RD_EN	BOOL	FALSE	1 = Leer datos de leva 8 (Levas de 112 a 127)
38.2	MSRRD_EN	BOOL	FALSE	1 = Leer valores medidos
38.3	CNTTRC_EN	BOOL	FALSE	1 = Leer valores de contaje de la pista de leva de contaje
38.4	ACTPOS_EN	BOOL	FALSE	1 = Leer Datos de posición y pista
38.5	ENCVAL_EN	BOOL	FALSE	1 = Leer valores del encóder
38.6	CAMOUT_EN	BOOL	FALSE	1 = Leer datos de leva y pista
Bits de finalización de selectores de función				
40.0	EDGE_D	BOOL	FALSE	1 = "Activar captura de bordes" o "Desactivar captura de bordes" finalizado
40.1	SIM_D	BOOL	FALSE	1 = "Activar simulación" o "Desactivar simulación" finalizado
40.2	MSR_D	BOOL	FALSE	1 = "Activar medición de longitud" o "Desactivar medición de longitud" finalizado
40.3	REFTR_D	BOOL	FALSE	1 = "Activar redisparar punto de referencia" o "Desactivar redisparar punto de referencia" finalizado
40.4	SSW_D	BOOL	FALSE	1 = "Activar final de carrera por software" o "Desactivar final de carrera por software" finalizado

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
Bits de finalización de peticiones de escritura				
41.0	MDWR_D	BOOL	FALSE	1 = Petición "Escribir datos máquina" finalizada
41.1	MD_D	BOOL	FALSE	1 = Petición "Habilitar datos máquina" finalizada
41.2	AVALREM_D	BOOL	FALSE	1 = "Deshacer Ajustar valor real" o "Deshacer Ajustar valor real al vuelo" cancelado
41.3	CAM1WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 1" finalizada
41.4	CAM2WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 2" finalizada
41.5	CAM3WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 3" finalizada
41.6	CAM4WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 4" finalizada
41.7	CAM5WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 5" finalizada
42.0	CAM6WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 6" finalizada
42.1	CAM7WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 7" finalizada
42.2	CAM8WR_D	BOOL	FALSE	1 = Petición "Escribir datos de leva 8" finalizada
42.3	REFPT_D	BOOL	FALSE	1 = Petición "Definir punto de referencia" finalizada
42.4	AVAL_D	BOOL	FALSE	1 = Petición "Ajustar valor real" finalizada
42.5	FVAL_D	BOOL	FALSE	1 = Petición "Ajustar valor real al vuelo" terminada
42.6	ZOFF_D	BOOL	FALSE	1 = Petición "Activar decalaje de señal cero" finalizada
42.7	CH01CAM_D	BOOL	FALSE	1 = Petición "Modificar leva 1" finalizada
43.0	CH16CAM_D	BOOL	FALSE	1 = Petición "Modificar 16 levas" finalizada (cambio rápido de levas)
Bits de finalización de peticiones de lectura				
43.1	MDRD_D	BOOL	FALSE	1 = Petición "Leer datos máquina" finalizada
43.2	CAM1RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 1" finalizada
43.3	CAM2RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 2" finalizada
43.4	CAM3RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 3" finalizada
43.5	CAM4RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 4" finalizada
43.6	CAM5RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 5" finalizada
43.7	CAM6RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 6" finalizada
44.0	CAM7RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 7" finalizada
44.1	CAM8RD_D	BOOL	FALSE	1 = Petición "Leer datos de leva 8" finalizada
44.2	MSRRD_D	BOOL	FALSE	1 = Petición "Leer valores medidos" terminada
44.3	CNTTRC_D	BOOL	FALSE	1 = Petición "Leer valores de contaje de la pista de leva de contaje" finalizada
44.4	ACTPOS_D	BOOL	FALSE	1 = Petición "Leer datos de posición y pista" finalizada
44.5	ENCVAL_D	BOOL	FALSE	1 = Petición "Leer valor actual del encóder" finalizada
44.6	CAMOUT_D	BOOL	FALSE	1 = Petición "Leer datos de leva y pista" finalizada
Bits de error de selectores de función				

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
46.0	EDGE_ERR	BOOL	FALSE	1 = Error en "Activar captura de bordes" o "Desactivar captura de bordes"
46.1	SIM_ERR	BOOL	FALSE	1 = Error al "Activar simulación" o "Desactivar simulación"
46.2	MSR_ERR	BOOL	FALSE	1 = Error al "Activar medición de longitud" o "Desactivar medición de longitud"
46.3	REFTR_ERR	BOOL	FALSE	1 = Error al "Activar redisparar punto de referencia" o "Desactivar redisparar punto de referencia"
46.4	SSW_ERR	BOOL	FALSE	1 = Error al "Activar final de carrera por software" o "Desactivar final de carrera por software"
Bits de error de peticiones de escritura				
47.0	MDWR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos máquina"
47.1	MD_ERR	BOOL	FALSE	1 = Error en la tarea "Activar datos máquina"
47.2	AVALREM_ERR	BOOL	FALSE	1 = Error en la tarea "Deshacer Ajustar valor real" o "Deshacer Ajustar valor real al vuelo"
47.3	CAM1WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 1"
47.4	CAM2WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 2"
47.5	CAM3WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 3"
47.6	CAM4WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 4"
47.7	CAM5WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 5"
48.0	CAM6WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 6"
48.1	CAM7WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 7"
48.2	CAM8WR_ERR	BOOL	FALSE	1 = Error en la tarea "Escribir datos de leva 8"
48.3	REFPT_ERR	BOOL	FALSE	1 = Error en la tarea "Definir punto de referencia"
48.4	AVAL_ERR	BOOL	FALSE	1 = Error en la tarea "Ajustar valor real"
48.5	FVAL_ERR	BOOL	FALSE	1 = Error en la tarea "Ajustar valor real al vuelo"
48.6	ZOFF_ERR	BOOL	FALSE	1 = Error en la tarea "Activar decalaje de señal cero"
48.7	CH01CAM_ERR	BOOL	FALSE	1 = Error en la tarea "Cambiar leva 1"
49.0	CH16CAM_ERR	BOOL	FALSE	1 = Error en la tarea "Cambiar 16 levas" (cambio rápido de levas)
Bits de error de las peticiones de lectura				
49.1	MDRD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos máquina"
49.2	CAM1RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 1"
49.3	CAM2RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 2"
49.4	CAM3RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 3"
49.5	CAM4RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 4"
49.6	CAM5RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 5"
49.7	CAM6RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 6"
50.0	CAM7RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 7"
50.1	CAM8RD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de leva 8"

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
50.2	MSRRD_ERR	BOOL	FALSE	1 = Error en la tarea "Leer valores medidos"
50.3	CNTTRC_ERR	BOOL	FALSE	1 = Error en la tarea "Leer valores de contaje de la pista de leva de contaje"
50.4	ACTPOS_ERR	BOOL	FALSE	1 = Error en la tarea "Leer datos de posición y pista"
50.5	ENCVAL_ERR	BOOL	FALSE	1 = Error en la tarea "Leer valor actual del encóder"
50.6	CAMOUT_ERR	BOOL	FALSE	1 = Error en la petición "Leer datos de leva y pista"
Gestión de peticiones para la instrucción CAM_CTRL o bien CAM_CTRL_452				
52.0	JOB_ERR	INT	0	Error de comunicación
54.0	JOBBUSY	BOOL	FALSE	1 = Por lo menos hay una tarea en curso
54.1	JOBRESET	BOOL	FALSE	1 = Resetear todos los bits de error y finalización
Gestión de peticiones para la instrucción CAM_MSRM_452				
56.0	JOB_ERR_M	INT	0	Error de comunicación
58.0	JOBBUSY_M	BOOL	FALSE	1 = Petición en curso
Datos de la petición "Decalaje de señal cero"				
86.0	ZOFF	DINT	L#0	Decalaje de señal cero
Datos de la petición "Ajustar valor real"				
90.0	AVAL	DINT	L#0	Coordenada de "Ajustar valor real"
Datos de la petición "Ajustar valor real al vuelo"				
94.0	FVAL	DINT	L#0	Coordenada de "Ajustar valor real al vuelo"
Datos de la petición "Definir punto de referencia"				
98.0	REFPT	DINT	L#0	Coordenada de "Definir punto de referencia"
Datos de la tarea "Modificar flancos de leva"				
102.0	CAM_NO	INT	0	Número de leva
104.0	CAM_START	DINT	L#0	Inicio de leva
108.0	CAM_END	DINT	L#0	Fin de leva
Datos de la petición "Medición de longitud/Captura de bordes"				
112.0	BEG_VAL	DINT	L#0	Valor inicial
116.0	END_VAL	DINT	L#0	Valor final
120.0	LEN_VAL	DINT	L#0	Longitud
Datos de la tarea "Leer valores de contaje"				
124.0	CNT_TRC0	INT	0	Valor de contaje actual para la pista de leva de contaje 0
126.0	CNT_TRC1	INT	0	Valor de contaje actual para la pista de leva de contaje 1
Datos de la tarea "Leer datos de posición y pista"				
128.0	ACTPOS	DINT	L#0	Posición actual
132.0	ACTSPD	DINT	L#0	Velocidad actual
136.0	TRACK_ID	DWORD	DW#16#0	Bits identificadores de pista de las pistas 0 a 31
Datos de la tarea "Leer datos de encóder"				

Dirección	Parámetros	Tipo de datos	Valor inicial	Descripción
140.0	ENCVAL	DINT	L#0	Valor del encóder
144.0	ZEROVAL	DINT	L#0	Lectura del contador con la última marca cero
148.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto
Datos de la tarea "Leer datos de leva y pista"				
152.0	CAM_00_31	DWORD	DW#16#0	Bits identificadores de leva para las levas de 0 a 31
156.0	CAM_32_63	DWORD	DW#16#0	Bits identificadores de leva para las levas de 32 a 63
160.0	CAM_64_95	DWORD	DW#16#0	Bits identificadores de leva para las levas de 64 a 95
164.0	CAM_96_127	DWORD	DW#16#0	Bits identificadores de leva para las levas de 96 a 127
168.0	TRACK_ID1	DWORD	DW#16#0	Bits identificadores de pista para las pistas de 0 a 31
172.0	ACTPOS1	DINT	L#0	Posición actual
Datos de la tarea "Cambio rápido de levas"				
176.0	C_QTY	BYTE	B#16#0	Número de levas a modificar
177.0	DIS_CHECK	BOOL	FALSE	1 = Desactivar la comprobación de datos
180.0	CAM	ARRAY [0...15] STRUCT		Nota: La estructura que sigue a continuación tiene que rellenarse en todas las levas que vayan a modificarse.
Dirección relativa				
+0.0	CAM_NO	BYTE	B#16#0	Número de la leva a modificar
+1.0	C_EFFDIR	BOOL	FALSE	1 = Cambio del sentido de actuación
+1.1	C_CBEGIN	BOOL	FALSE	1 = Cambio del inicio de la leva al valor CBEGIN (nuevo inicio de leva)
+1.2	C_CEND	BOOL	FALSE	1 = Cambio del fin de leva / del tiempo de activación al valor CEND (nuevo fin de leva)
+1.3	C_LTIME	BOOL	FALSE	1 = Cambio del tiempo de anticipación al valor LTIME (nuevo tiempo de anticipación)
+1.4	CAM_OFF	BOOL	FALSE	1 = Desactivación de la leva durante el cambio de datos de leva
+1.5	EFFDIR_P	BOOL	FALSE	1 = Nuevo sentido de actuación positivo (+)
+1.6	EFFDIR_M	BOOL	FALSE	1 = Nuevo sentido de actuación negativo (-)
+2.0	CBEGIN	DINT	L#0	Nuevo inicio de leva
+6.0	CEND	DINT	L#0	Nuevo fin de leva / Nuevo tiempo de activación
+10.0	LTIME	INT	0	Nuevo tiempo de anticipación

Bloque de datos de parámetros

Tarea

Los datos máquina y los datos de leva se encuentran depositados en el DB de parámetros. Los parámetros pueden ser modificados por el programa de usuario o por un sistema de manejo y visualización. Existe la posibilidad de importar los datos modificados a la interfaz de parametrización y visualizarlos desde ahí. Los datos que se muestran en la interfaz de parametrización se pueden exportar a un DB de parámetros.

En un módulo puede haber varios juegos de datos de parametrización (p. ej. para distintas recetas) a los que se puede conmutar por medio del programa.

Estructura

DB de parámetros
CAM_P016TYPE (UDT3) Datos máquina Datos de las levas de 0 a 15
CAM_P032TYPE (UDT4) Datos máquina Datos de las levas de 0 a 31
CAM_P064TYPE (UDT5) Datos máquina Datos de las levas de 0 a 63
CAM_P0128TYPE (UDT6) Datos máquina Datos de las levas de 0 a 127

Contenido

La siguiente tabla muestra el contenido del DB de parámetros.

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
Datos máquina				
3.1	PI_MEND	BOOL	FALSE	• FM 352: 0 • FM 452: 1: Habilitar la alarma de proceso: Fin de la medición
3.2	PI_CAM	BOOL	FALSE	1: Habilitar alarma de proceso: Leva activa / inactiva

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
3.5	PI_MSTRT	BOOL	FALSE	• FM 352: 0 • FM 452: 1: Habilitar la alarma de proceso: Comienzo de la medición
4.0	EDGEDIST	DINT	L#0	Distancia mínima entre bordes en la captura de bordes
8.0	UNITS	DINT	L#1	Sistema de medida
12.0	AXIS_TYPE	DINT	L#0	0: Eje lineal, 1: Eje rotativo
16.0	ENDROTAX	DINT	L#100 000	Final del eje rotativo
20.0	ENC_TYPE	DINT	L#1	Tipo de encóder, longitud de telegrama
24.0	DISP_REV	DINT	L#80 000	Recorrido por vuelta
32.0	INC_REV	DINT	L#500	Incrementos por vuelta
36.0	NO_REV	DINT	L#1024	Número de revoluciones
40.0	BAUDRATE	DINT	L#0	Velocidad de transferencia
44.0	REFPT	DINT	L#0	Coordenada del punto de referencia
48.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto
52.0	RETR_TYPE	DINT	L#0	Tipo de "Redisparar el punto de referencia"
56.0	CNT_DIR	DINT	L#0	Sentido de contaje: 0: normal, 1: Invertida
63.0	MON_WIRE	BOOL	TRUE	1: Supervisión rotura de hilo
63.1	MON_FRAME	BOOL	TRUE	1: Supervisión error de telegrama
63.2	MON_PULSE	BOOL	TRUE	1: Supervisión pulsos de error
64.0	SSW_STRT	DINT	L#-100 000 000	Final de carrera de software Inicio
68.0	SSW_END	DINT	L#100 000 000	Final de carrera de software Fin
76.0	C_QTY	DINT	L#0	Alcance: 0, 1, 2, 3 = máx. 16, 32, 64, 128 levas
80.0	HYS	DINT	L#0	Histéresis
84.0	SIM_SPD	DINT	L#0	Velocidad de simulación
90.0	TRACK_OUT	WORD	W#16#0	Control de las salidas de pista: 0 = Leva electrónica, 1 = CPU; Número de bit = Número de pista
95.0	EN_IN_I3	BOOL	FALSE	Habilitación Entrada I3
95.1	EN_IN_I4	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I4
95.2	EN_IN_I5	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I5
95.3	EN_IN_I6	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I6
95.4	EN_IN_I7	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I7

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
95.5	EN_IN_I8	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I8
95.6	EN_IN_I9	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I9
95.7	EN_IN_I10	BOOL	FALSE	• FM 352: 0 • FM 452: Habilitación entrada I10
99.0	SPEC_TRC0	BOOL	FALSE	1 = Pista 0 es pista de leva de contaje
99.1	SPEC_TRC1	BOOL	FALSE	1 = Pista 1 es pista de leva de contaje
99.2	SPEC_TRC2	BOOL	FALSE	1 = Pista 2 es pista de leva de freno
100.0	CNT_LIM0	DINT	L#2	Valor de contaje superior para la pista de leva de contaje 0
104.0	CNT_LIM1	DINT	L#2	Valor de contaje superior para la pista de leva de contaje 1
Datos de leva respectivamente para levas de 0 a 15 / 0 a 31 / 0 a 63 / 0 a 127				
108.0		STRUCT		(cada elemento 12 bytes de longitud)
Dirección relativa				
+0.0	CAMVALID	BOOL	FALSE	1: Leva válida
+0.1	EFFDIR_P	BOOL	TRUE	1: Sentido de actuación positivo (+)
+0.2	EFFDIR_M	BOOL	TRUE	1: Sentido de actuación negativo (-)
+0.3	CAM_TYPE	BOOL	FALSE	0: Leva de recorrido, 1: Leva de tiempo
+0.4	PI_SW_ON	BOOL	FALSE	1: Alarma de proceso al activar
+0.5	PI_SW_OFF	BOOL	FALSE	1: Alarma de proceso al desactivar
+1.0	TRACK_NO	BYTE	B#16#0	Número de pista
+2.0	CBEGIN	DINT	L#-100 000 000	Inicio de leva
+6.0	CEND	DINT	L#100 000 000	Final de leva/Tiempo de activación
+10.0	LTIME	INT	0	Tiempo de anticipación

Bloque de datos de diagnóstico

Tarea

El DB de diagnóstico es el archivo de datos para la instrucción CAM_DIAG (Página 7455) o CAM_DIAG_452 (Página 7458). Contiene el búfer de diagnóstico del módulo acondicionado por esta instrucción.

Estructura

DB de diagnóstico
Dirección del módulo
Datos internos
Estado de la tarea
Bit de inicio
Búfer de diagnóstico acondicionado

Contenido

La siguiente tabla muestra el contenido del DB de diagnóstico.

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
0.0	MOD_ADDR (¡Registrar!)	INT	0	Dirección del módulo
256.0	JOB_ERR	INT	0	Error de comunicación
258.0	JOBBUSY	BOOL	FALSE	1 = Petición activa
258.1	DIAGRD_EN	BOOL	FALSE	1 = Leer el búfer de diagnóstico necesariamente
260.0	DIAG_CNT	INT	0	Número de entradas válidas de la lista
262.0	DIAG[1]	STRUCT		Entrada más reciente de datos de diagnóstico
272.0	DIAG[2]	STRUCT		Segunda entrada de datos de diagnóstico
282.0	DIAG[3]	STRUCT		Tercera entrada de datos de diagnóstico
292.0	DIAG[4]	STRUCT		Última entrada de datos de diagnóstico

Estructura de la entrada de diagnóstico

La siguiente tabla muestra la estructura de una entrada de diagnóstico DIAG[n].

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
+0.0	STATE	BOOL	FALSE	0 = Evento saliente 1 = Evento entrante
+0.1	INTF	BOOL	FALSE	1 = Error interno
+0.2	EXTF	BOOL	FALSE	1 = Error externo

Dirección	Parámetro	Tipo de datos	Valor inicial	Descripción
+2.0	FCL	INT	0	Clase de error 1: Fallo operativo 4: Fallo de datos 5: Fallo de datos máquina 7: Fallo de datos de leva 15: Avisos 128: Error de diagnóstico
+4.0	FNO	INT	0	Número de error 0 … 255
+6.0	CH_NO	INT	0	Número de canal (siempre 1)
+8.0	CAMNO	INT	0	Número de leva de 0 a 127 con clase de error = error de datos de leva

Gestión de peticiones (CAM_CTRL)

Tareas

El intercambio de datos a partir de las señales de control y de respuesta con el módulo se realiza a través de las peticiones.

Para lanzar una petición active el bit de inicio correspondiente del bloque de datos del canal y, en el caso de las peticiones de escritura, proporcione además los datos correspondientes. Llame la instrucción CAM_CTRL (Página 7453) para ejecutar la petición.

Si se utiliza el módulo de forma centralizada, se ejecutará de inmediato una petición de lectura. Si se utiliza el FM 352 de forma descentralizada, una petición de lectura puede precisar varios ciclos.

A causa de los acuses necesarios del módulo una petición de escritura precisa por lo menos 3 llamadas (o ciclos OB). Si se utiliza el FM 352 de forma descentralizada, una petición de escritura puede requerir más de 3 llamadas.

Es posible activar varias peticiones de forma simultánea, incluso con señales de control. Excepto la petición de escribir el selector de función, las peticiones se procesan en la secuencia de los bits de inicio que está definida en el DB de canal. En cuanto se ha procesado por completo una petición, se desactiva el bit de inicio. Con la siguiente llamada de la instrucción se determina y ejecuta la petición siguiente.

Además del bit de inicio, para cada petición hay un bit de finalización y otro de error. Se diferencian en la terminación del nombre:

Bit	Terminación
Bit de inicio	_EN ("enable")
Bit de finalización	_D ("done")
Bit de error	_ERR ("error")

Los bits de finalización y de error de la petición se tienen que poner a 0 tras la evaluación o antes del inicio de esta petición.

Cuando se activa el bit JOBRESET antes del procesamiento de las peticiones pendientes, se desactivan todos los bits de finalización y error. Seguidamente, el bit JOBRESET se vuelve a poner a 0.

Selectores de función

Los selectores de función activan y desactivan estados del módulo. Una petición de escritura del selector de función sólo se ejecuta si cambia una posición del selector. La posición del selector de función se mantiene tras ejecutarse la petición.

La medición de la longitud y la captura de bordes no deben estar activados de forma simultánea. Por ello la instrucción CAM_CTRL se encarga de que cuando se active un selector de función se desactive el otro. Si a pesar de ello activa los dos selectores de función de forma simultánea (0 → 1), se activará la medición de longitud.

Los selectores de función y las peticiones se pueden emplear de forma simultánea en una llamada de la instrucción CAM_CTRL.

Al igual que ocurre con las peticiones, para los selectores de función hay bits de finalización con la terminación _D y bits de error con la terminación _ERR.

Para poder evaluar los bits de finalización y de error del selector de función se deben poner a 0 en caso de modificar un selector de función.

Estado de la petición

El estado de procesamiento de la petición se lee en el valor de retorno RETVAL y en el bit de actividad JOBBUSY del DB de canal. El estado de una petición concreta se puede evaluar por medio del bit de inicio, el bit de finalización y el bit de error de esa petición.

	RETVAL	JOBBUSY	Bit de inicio _EN	Bit de finalización _D	Bit de error _ERR
Petición activa	1	1	1	0	0
La petición ha finalizado sin errores	0	0	0	1	0
La petición ha finalizado con errores	-1	0	0	1	1
Petición de escritura finalizada	-1	0	0	0	1

Gestión de peticiones (CAM_CTRL_452))

Peticiones

El intercambio de datos con el módulo a partir de las señales de control y respuesta se realiza mediante peticiones.

Para iniciar una petición active el bit de inicio correspondiente del DB de canal y, en el caso de las peticiones de escritura, proporcione además los datos correspondientes. Llame la instrucción CAM_CTRL_452 (Página 7457) para ejecutar la petición.

Una petición de lectura se ejecuta de inmediato. A causa de los acuses necesarios del módulo una petición de escritura precisa por lo menos 3 llamadas (o ciclos OB).

Es posible activar varias peticiones de forma simultánea, incluso con señales de control. Excepto la petición de escribir los selectores de función, las peticiones se procesan en el orden de los bits de disparo que está definido en el DB de canal. En cuanto se ha procesado por completo una petición, se desactiva el bit de inicio. Con la siguiente llamada de la instrucción se determina y ejecuta la petición siguiente.

Además del bit de inicio, para cada petición hay un bit de finalización y otro de error. Se diferencian en la terminación del nombre:

Bit	Terminación
Bit de inicio	_EN ("enable")
Bit de finalización	_D ("done")
Bit de error	_ERR ("error")

Los bits de finalización y error de la petición deben ponerse a 0 tras la evaluación o antes del inicio de dicha petición.

Cuando se activa el bit JOBRESET antes del procesamiento de las peticiones pendientes, se desactivan todos los bits de finalización y error. Seguidamente, el bit JOBRESET se vuelve a poner a 0.

Selectores de función

Los selectores de función activan y desactivan estados del módulo. Una petición de escritura de los selectores de función sólo se ejecuta si cambia una posición del selector. La posición del selector de función se mantiene tras ejecutar la petición.

La medición de la longitud y la captura de bordes no deben estar activados de forma simultánea. Por ello la instrucción CAM_CTRL_452 se encarga de que cuando se active un selector de función se desactive el otro. Si se activan los dos selector de función a la vez (0 → 1), se activa la medición de longitud.

Los selectores de función y las peticiones se pueden utilizar simultáneamente en una llamada de la instrucción CAM_CTRL_452.

Al igual que ocurre con las peticiones, para los selectores de función hay bits de finalización con la terminación _D y bits de error con la terminación _ERR.

Para poder evaluar los bits de finalización y error de los selectores de función se deben poner a 0 en caso de modificar un selector de función.

Estado de la petición

El estado de procesamiento de la petición se lee en el valor de retorno RET_VAL y en el bit de actividad JOBBUSY del DB de canal. El estado de una petición concreta se puede evaluar por medio del bit de inicio, el bit de finalización y el bit de error de dicha petición.

	RET_VAL	JOBBUSY	Bit de inicio _EN	Bit de finalización _D	Bit de error _ERR
Petición activa	1	1	1	0	0
La petición ha finalizado sin errores	0	0	0	1	0
La petición ha finalizado con errores	-1	0	0	1	1
Petición de escritura cancelada	-1	0	0	0	1

Clases de error

Clase de error 1: Fallo operativo

Descripción

Los fallos operativos se detectan de forma asíncrona respecto a un control.

Nº	Significado		Alarma de diagnóstico
1	Rebase del final de carrera por software Inicio		Sí
2	Rebase del final de carrera por software Fin		sí
3	Rebase del inicio de la zona de desplazamiento		sí
4	Rebase del fin de la zona de desplazamiento		sí
13	Ajustar valor real al vuelo no ejecutable		sí
	Causa	Los finales de carrera de software, tras ajustar valor real al vuelo, se encuentran fuera del rango de desplazamiento (-100 m...+100 m ó -1.000 m...+1.000 m). El decalaje resultante de Ajustar valor real / Ajustar valor real al vuelo es mayor que ±100 m o ±1.000 m.	
	Efecto	Eje no sincronizado.	

Clase de error 4: Error de datos

Descripción

Los errores de datos se detectan de forma síncrona respecto a un control/mando.

N.º	Significado		Alarma de diagnóstico
10	Decalaje de señal cero erróneo		No
	Causa	El decalaje de señal cero es mayor que ± 100 m o ± 1.000 m. Los finales de carrera de software, tras ajustar el decalaje de señal cero, se encuentran fuera del rango de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: el valor del decalaje de señal cero es mayor que el final del eje rotativo.	
11	Indicación del valor real errónea		no
	Causa	Eje lineal: la coordenada se encuentra fuera del final de carrera de software actual (que puede estar desplazado). Eje rotativo: la coordenada es < 0 ó mayor que el final del eje rotativo.	
12	Punto de referencia erróneo		no
	Causa	Eje lineal: la coordenada se encuentra fuera del final de carrera de software actual (que puede estar desplazado). Eje rotativo: la coordenada es < 0 ó mayor que el final del eje rotativo.	
20	Activar datos máquina no admisible		no
	Causa	En el módulo no hay datos máquina nuevos (sin errores)	
21	Ajustar valor real al vuelo no admisible		no
	Causa	Se ha intentado ejecutar "Ajustar valor real al vuelo" teniendo "Redisparar punto de referencia" activado.	
27	Ajuste codificado en bits no permitido		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0. Se ha intentado seleccionar simultáneamente" Medición de longitud" y "Captura de bordes".	
28	Redisparar el punto de referencia no admisible		no
	Causa	Se ha intentado ejecutar "Redisparar el punto de referencia" estando "Ajustar valor real al vuelo" pendiente. Se ha intentado ejecutar "Redisparar el punto de referenciao" con un encóder SSI.	
29	Comando codificado en bits no permitido		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
30	Tiempo de anticipación erróneo		no
31	Número de leva erróneo		no
	Causa	La leva no es válida El número de leva no se encuentra en el rango 0...127.	
32	Inicio de leva erróneo		no

N.º	Significado		Alarma de diagnóstico
	Causa	El inicio de leva se encuentra fuera del rango de desplazamiento (-100 m...+100 m ó -1.000 m...+1.000 m). Eje rotativo: el inicio de leva es < 0 o mayor que el final del eje rotativo.	
33	Fin de leva erróneo / Tiempo de activación erróneo		no
	Causa	El fin de leva se encuentra fuera del rango de desplazamiento (-100 m...+100 m ó -1.000 m...+1.000 m). Eje rotativo: el final de leva es < 0 o mayor que el final del eje rotativo. La leva no está activada por lo menos en la duración de un pulso. En una leva invertida no hay una distancia mínima de 4 pulsos entre inicio de leva y fin de leva.	
34	No es posible cancelar la activación del valor real		no
	Causa	Con un encóder SSI y un eje líneal, tras la ejecución del ajuste el valor real de posición se encontraría fuera de la zona de trabajo.	
35	Indicación del valor real errónea al Ajustar valor real / Ajustar valor real al vuelo		no
	Causa	El preajuste del valor real se encuentra fuera del rango numérico admisible de ± 100 m ó ± 1.000 m. Los finales de carrera de software, tras ejecutar el ajuste, se encontrarían fuera del rango de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). El decalaje resultante de Ajustar valor real / Ajustar valor al vuelo sería mayor que ±100 m o ±1.000 m.	
107	Eje no parametrizado		no
	Causa	En el eje no hay ningún dato máquina disponible. En el eje no hay ningún dato máquina activado.	
108	Eje no sincronizado		no
	Causa	Uno de los ajustes "Ajustar valor real" o "Ajustar valor real al vuelo" se ha activado a pesar de que el eje no está sincronizado.	
109	Procesamiento de levas en curso.		no
110	Número erróneo de levas a modificar		no

Clase de error 5: Error de datos máquina

Descripción

La alarma de diagnóstico sólo se disparará con un módulo de datos del sistema (SDB) con errores.

N.º	Significado		Alarma de diagnóstico
5	Error en el ajuste de la alarma de proceso		sí
	Causa	Ha intentado seleccionar una alarma de proceso que el módulo no admite.	
6	Distancia mínima entre bordes errónea		sí
	Causa	Ha indicado como distancia mínima entre bordes un valor < 0 ó > 10^9 µm	
8	Tipo de eje erróneo		sí

N.º	Significado		Alarma de diagnóstico
	Causa	Como tipo de eje no ha indicado ni 0 ni 1	
9	**Final erróneo del eje rotativo**		sí
	Causa	El valor del final del eje rotativo se encuentra fuera del rango admisible de 1 a 10^9 µm o de 1 a 10^8 µm (según sea la resolución).	
10	**Tipo de encóder erróneo**		sí
	Causa	El valor del tipo de encóder se encuentra fuera del margen admisible de 1 a 10	
11	**Recorrido/Vuelta erróneos**		sí
	Causa	El valor del recorrido/vuelta se encuentra fuera del margen admisible de 1 a 10^9 µm (independientemente de la resolución).	
13	**Número de incrementos/vuelta erróneo**		sí
14	**Número de vueltas erróneo**		sí
15	**Velocidad de transferencia errónea**		sí
	Causa	Para la velocidad de transferencia ha indicado un valor fuera del rango admisible de 0 a 3.	
16	**Coordinada del punto de referencia errónea**		sí
	Causa	La coordenada se encuentra fuera del margen de -100 m a +100 m o, respectivamente, de -1000 m a +1000 m, según la resolución. Eje lineal: La coordenada se encuentra fuera de la zona de trabajo. Eje rotativo: la coordenada es mayor que el final del eje rotativo ó < 0.	
17	**Reajuste erróneo del encóder absoluto**		sí
	Causa	Encóder SSI: el valor de reajuste del encóder absoluto no se encuentra dentro del rango del encóder (incrementos por vuelta x número de vueltas - 1).	
18	**Tipo erróneo de "Redisparar el punto de referencia"**		sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0, 1, 6 y 7.	
19	**Ajuste erróneo del sentido**		sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0 y 1.	
20	**Supervisión de hardware no posible**		sí
	Causa	Ha puesto en "FALSE" la vigilancia de errores de telegrama en el DB de parámetros. El encóder no permite que se vigilen los impulsos erróneos (desactivar MON_PULSE)	
21	**Final erróneo de carrera de software Inicio**		sí
	Causa	Eje lineal: el final de carrera de software Inicio se encuentra fuera del rango de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución). Eje lineal: el final de carrera de software Inicio (dado el caso, incluido un decalaje de señal cero presente) es menor que -100 m ó -1000 m (según la resolución).	
22	**Final erróneo de carrera de software Fin**		sí

N.º	Significado		Alarma de diagnóstico
	Causa	Eje lineal: el final de carrera de software Fin se encuentra fuera del rango de desplazamiento (-100 m...+100 m ó 1000 m...+1000 m, dependiendo de la resolución) o es menor que el final de carrera de software Inicio. Eje lineal: el final de carrera de software Fin (dado el caso, incluido un decalaje de señal cero presente) es mayor que +100 m ó +1000 m (según la resolución).	
144	Alcance erróneo		sí
	Causa	Para el alcance ha indicado un valor distinto de 0 a 3.	
145	Histéresis errónea		sí
	Causa	La histéresis se encuentra fuera del rango 0...65535 × resolución. La histéresis es superior a ¼ × área de trabajo ó ¼ × rango de eje rotativo.	
146	Velocidad de simulación errónea		sí
	Causa	La velocidad de simulación se encuentra fuera del rango 1000 × RES hasta 3×10^7 × RES o es mayor que 5×10^8 µm/min. La velocidad de simulación no puede ajustarse internamente.	
147	Pista errónea		sí
	Causa	Se ha seleccionado el control de una pista fuera de 0...15 (bit 0...15).	
148	Selección errónea de entradas de habilitación		sí
	Causa	Ha querido habilitar una pista fuera de 3...10 (bit 0...7) por medio de una señal externa.	
149	Selección errónea de una pista especial		sí
	Causa	Ha querido definir una pista fuera de 0, 1 y 2 (bits 0, 1 y 2) a modo de pista especial	
150	Valor de contaje superior erróneo pista 0		sí
	Causa	Ha indicado un valor de contaje < 2 ó > 65535 como valor de contaje superior.	
151	Valor de contaje superior erróneo pista 1		sí
	Causa	Ha indicado un valor de contaje < 2 ó > 65535 como valor de contaje superior.	
200	Resolución errónea		sí
	Causa	Ha indicado una resolución < 0,1 µm/impulso o >1000 µm/impulso. Ha indicado un recorrido/vuelta y un número de impulsos/vuelta que dan una resolución < 0,1 o > 1000.	
201	El encóder no se adapta al área de trabajo / rango del eje rotativo		sí
	Causa	Encóder SSI y eje rotativo: el encóder no cubre exactamente el rango del eje rotativo. Eje lineal: el encóder no cubre por lo menos el área de trabajo (finales de carrera de software incluidos).	

Clase de error 7: Error de datos de leva

Descripción

La alarma de diagnóstico sólo se disparará con un módulo de datos del sistema (SDB) con errores.

Nº	Significado		Alarma de diagnóstico
1	Alarma de proceso no admisible		Sí
	Causa	Ha querido indicar una alarma de proceso con una leva con un número > 7.	
2	Número de pista erróneo		sí
	Causa	El número de pista se encuentra fuera del margen de 0 a 31	
3	Inicio de leva erróneo		sí
	Causa	El inicio de leva se encuentra fuera del rango de desplazamiento (-100 m...+100 m ó -1.000 m...+1.000 m). Eje rotativo: el inicio de leva es < 0 y/o mayor que el final del eje rotativo.	
4	Fin de leva erróneo		sí
	Causa	El fin de leva se encuentra fuera del rango de desplazamiento (-100 m...+100 m ó -1.000 m...+1.000 m). La leva no tiene una longitud de por lo menos 1 pulso. Eje rotativo: el final de leva es < 0 y/o mayor que el final del eje rotativo. En una leva invertida no hay una distancia mínima de 4 pulsos entre inicio de leva y fin de leva.	
5	Tiempo de activación erróneo		sí
	Causa	El tiempo de activación es < 0 µs. El valor máximo depende del alcance.	
6	Tiempo de anticipación erróneo[1)]		sí
	Causa	El tiempo de anticipación es < 0 µs. El valor máximo depende del alcance.	
50	Demasiados juegos de levas		sí
	Causa	Ha querido indicar más juegos de levas de los posibles con este alcance.	
51	Eje en funcionamiento		sí
	Causa	Ha querido indicar juegos de levas con la leva activada.	
52	Eje no parametrizado		sí
	Causa	Ha querido indicar datos de leva a pesar de que todavía no hay datos máquina activos.	

[1)] El mensaje de error también puede aparecer si ha parametrizado el sentido de contaje "invertido" en combinación con un encóder absoluto (SSI).

Clase de error 15: Avisos

N.º	Significado		Alarma de diagnóstico
1	**Inicio de parametrización**		No
	Causa	El módulo ha detectado una parametrización por medio de un bloque de datos de sistema.	
2	**Final de la parametrización**		no
	Causa	El módulo ha procesado sin problemas la parametrización por medio de un bloque de datos del sistema.	

Clase de error 128: Error de diagnóstico

N.º	Significado		Alarma de diagnóstico
4	**Falta tensión auxiliar externa**		sí
	Causa	• La tensión auxiliar externa de 24 V no se ha conectado o ha fallado • Falta conector frontal • Cortacircuito, p. ej. en el encóder conectado	
	Efecto	• El procesamiento de levas se desactivará • Desactivación de las salidas de pista • Borrado de la sincronización, con encóders incrementales • Los parámetros no han sido asignados al módulo (señal de respuesta PARA = 0).	
	Solución	Comprobar que la conexión de 24 V sea correcta (si la conexión de 24 V es correcta, significa que el módulo está averiado)	
5	**Falta conector frontal (FM 452)**		sí
	Causa	El conector frontal no está insertado	
	Efectos	• Falta tensión auxiliar externa de 24 V • El módulo no está operativo	
	Solución	Enchufar el conector frontal	
51	**Ha saltado la vigilancia de tiempo (watchdog)**		sí
	Causa	• Interferencias fuertes en el módulo • Error en el módulo	
	Efectos	• El módulo se restaura • Siempre que tras el reset del módulo no se detecte ningún defecto en el módulo, éste vuelve a estar listo para funcionar • El módulo notifica que ha transcurrido la vigilancia del tiempo con "entrante" y "saliente"	
	Solución	• Eliminación de interferencias • Consulte al departamento comercial correspondiente cuando las circunstancias exactas que han dado lugar al error son de mayor entidad. • Sustituir el módulo	
52	**Caída de la alimentación interna del módulo**		sí

N.º	Significado		Alarma de diagnóstico
	Causa	Error en el módulo	
	Efectos	• El módulo se resetea • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, éste vuelve a estar operativo.	
	Solución	Sustituir el módulo	
70	Alarma de proceso perdida		sí
	Causa	El módulo ha detectado un evento de alarma de proceso que no puede ser notificado porque el mismo evento todavía no ha sido procesado por el programa de usuario o la CPU.	
	Efectos	• El procesamiento de levas se desactiva • Desactivación de las salidas de pista • Borrado de la sincronización, con encóders incrementales	
	Solución	• Incorporar un OB de alarma de proceso al programa de usuario • Comprobar la conexión de bus del módulo • Desactivar la alarma de proceso • Adapte el hardware y software a sus necesidades de proceso (p. ej. CPU más rápida, optimizar el programa de usuario)	
144	Rotura de hilo Encóder		sí
	Causa	• El cable del encóder no está enchufado o está cortado • Encóder sin señales de cuadratura • Asignación de pines incorrecta • Longitud de cable excesiva • Cortocircuito de las señales del sensor	
	Efectos	• El procesamiento de levas se desactiva • Desactivación de las salidas de pista • Borrado de la sincronización, con encóders incrementales	
	Solución	• Comprobar el cable del encóder • Guardar los datos técnicos del encóder • La vigilancia puede desactivarse de forma provisional con la interfaz de parametrización bajo la responsabilidad del operador. • Guardar los datos técnicos del módulo	
145	Error de telegrama del encóder absoluto		sí
	Causa	El tráfico de telegramas entre el módulo y el encóder absoluto (SSI) tiene errores o está interrumpido: • El cable del encóder no está enchufado o está cortado • Tipo de encóder erróneo • Encóder mal configurado (encóders programables) • Longitud de telegrama mal indicada • El encóder proporciona valores erróneos (encóder defectuoso) • Interferencia en el cable del sistema de medición • Velocidad de transferencia escogida demasiado alta	

N.º	Significado		Alarma de diagnóstico
	Efectos	• El procesamiento de levas se desactiva • Desactivación de las salidas de pista • el último valor real correcto se mantiene sin cambios hasta el final de la siguiente transferencia SSI correcta	
	Solución	• Comprobar el cable del encóder • Comprobar el encóder • Comprobar el tráfico de telegramas entre el encóder y el módulo	
146	**Impulsos erróneos del encóder incremental**		sí
	Causa	• Vigilancia del encóder ha detectado pulsos erróneos • La cantidad de incrementos por vuelta está mal introducida • Encóder averiado: no proporciona la cantidad de pulsos indicada • Marca cero errónea o no presente • Interferencias en el cable del encóder	
	Efectos	• El procesamiento de levas se desactiva • Desactivación de las salidas de pista • Borrado de la sincronización	
	Solución	• Indicar correctamente el número de incrementos/vuelta. • Comprobar el encóder y su cable • Cumplir las normas de apantallamiento y de conexión a tierra • La vigilancia puede desactivarse de forma provisional con la interfaz de parametrización bajo la responsabilidad del operador.	

FM 355 PID Control

PID_FM

Descripción de PID_FM

La instrucción PID_FM permite realizar las siguientes acciones durante el funcionamiento:

- Transmitir parámetros de operación al FM 355
- Transmitir parámetros de regulación al FM 355
- Leer valores del proceso del FM 355

Los parámetros de operación son todos los parámetros de entrada/salida que se encuentran en el DB de instancia de la instrucción entre los parámetros op_par y cont_par.

Los parámetros de regulación son todos los parámetros de entrada/salida que se encuentran en el DB de instancia de la instrucción después del parámetro cont_par.

Los valores del proceso son todos los parámetros de salida de la instrucción situados después del parámetro out_par.

Los datos necesarios se depositan en un DB de instancia en la CPU. Para cada canal de regulación utilizado se requiere un DB de instancia. Tras generar un DB de instancia, todos los parámetros de entrada/salida tienen asignado el valor FALSE.

Llamada

PID_FM se llama generalmente en el OB 35 de alarma cíclica y requiere una marcha de inicialización, que se inicia activando el parámetro COM_RST = TRUE en el arranque de la CPU. Es posible realizar una llamada de la instrucción en el OB de arranque, pero no es necesario. Tras la marcha de inicialización, PID_FM pone el parámetro COM_RST en FALSE. Si se utiliza el FM 355 en periferia descentralizada, esto puede requerir varios ciclos de llamada.

PID_FM debe llamarse en el mismo OB así como todas las instrucciones restantes que acceden al mismo FM 355.

Si se efectúa una llamada a la instrucción PID_FM como DB multiinstancia, no se dispone de interfaz de puesta en servicio. En este caso, los FM 355 se deben poner en servicio a través de una tabla de observación.

Arranque

En el arranque y en la transición de STOP a RUN de la CPU, los parámetros del FM 355 se sobrescriben con los parámetros del SDB de la CPU.

Respuesta en caso de error

El parámetro de salida RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si los parámetros READ_VAR y LOAD_PAR no se desactivan.

En la llamada de la instrucción PID_FM puede producirse un error de acceso a la periferia, si el FM 355 no está insertado o si no recibe suministro de tensión. En ese caso, la CPU pasa a STOP si no hay ningún OB 122 cargado en la CPU.

Consulte también

Funcionamiento de PID_FM (Página 7485)

Modificar los parámetros a través de la HMI (Página 10279)

Crear y configurar DBs de instancia (Página 10271)

Funcionamiento de PID_FM

Influencia de los parámetros LOAD_OP, READ_VAR y LOAD_PAR

En la marcha de inicialización, PID_FM lee los parámetros de los datos del sistema y los deposita en el DB de instancia. El DB de instancia y los datos del sistema están sincronizados.

La tabla siguiente muestra cómo afectan los parámetros LOAD_OP, READ_VAR y LOAD_PAR al funcionamiento de la instrucción.

Parámetro	TRUE	FALSE
LOAD_OP	Transferir los parámetros de operación con las instrucciones WRREC y RDREC.	Transferir los parámetros de operación con accesos a la periferia directos en el proceso múltiplex
	READ_VAR adopta el estado TRUE	READ_VAR no cambia
	Tiempo de operación largo	Tiempo de operación reducido
	Los valores son efectivos tras un ciclo de la CPU o el FM. Rige el ciclo más largo.	Como máximo se requieren 3 ciclos de la CPU o del FM para que los valores sean efectivos. Rige el ciclo más largo
READ_VAR	Transferir valores del proceso con las instrucciones WRREC y RDREC	Los valores del proceso se transmiten con accesos a la periferia directos en el proceso múltiplex
	Tiempo de ejecución prolongado	Tiempo de ejecución reducido
	Todos los parámetros se leen del FM 355	Los siguientes parámetros **no** se leen del FM 355 • SP (consigna del FM) • ER (error de regulación) • DISV (magnitud perturbadora) • LMN_A • LMN_B
	Los siguientes parámetros se actualizan en **cada** llamada de la instrucción. • PV (valor real) • LMN (variable manipulada) • Las indicaciones binarias	Los siguientes parámetros se actualizan cada **cuatro** llamadas de la instrucción. • PV (valor real) • LMN (variable manipulada) • Las indicaciones binarias
	En el arranque de la CPU PID_FM adopta los siguientes valores del FM 355: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON Si estos valores se han cambiado en la CPU, en el primer arranque de la CPU se sobrescriben con los valores del FM 355	En el arranque de la CPU, los siguientes valores se leen de la instrucción y no del FM: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON
LOAD_PAR	Todos los parámetros de regulación se cargan de la CPU al FM 355. Esta acción es conveniente si los parámetros de regulación deben cambiarse con dependencia de los estados del proceso.	Sin transmisión de los parámetros del regulador

Una vez completada de transmisión de datos, los parámetros LOAD_OP, READ_VAR y LOAD_PAR se vuelven a poner a FALSE.

Uso en periferia descentralizada

Si se utiliza el FM 355 en la periferia descentralizada, la transmisión de datos y la desactivación de los parámetros puede requerir varios ciclos de llamada.

Particularidades de la transmisión en el proceso múltiplex

La multiplexación de los datos que se van a transmitir en el acceso al FM 355 con accesos directos a la periferia se controla con PID_FM. El control de multiplexación con PID_FM no

funciona si dos instancias de la instrucción PID_FM acceden al mismo número de canal de un módulo. El resultado son parámetros incorrectos en el FM 355 (p. ej. consigna y variable manipulada en manual) e indicaciones erróneas en los parámetros de salida de la instrucción PID_FM.

Almacenamiento de los parámetros en la EEPROM

En la reparametrización controlada por programa (LOAD_PAR, LOAD_OP) del módulo regulador con la instrucción PID_FM, aumenta su tiempo de operación. Los nuevos parámetros siempre entran en vigor inmediatamente. Los parámetros del regulador y el parámetro de operación SP_ER se guardan además en una memoria no volátil (EEPROM). Tras cada almacenamiento de los parámetros en la EEPROM, el siguiente se retrasa 30 minutos, ya que la vida útil de la EEPROM está limitada por el número de procesos de escritura. Tras una recuperación de la tensión de alimentación se pueden guardar nuevos parámetros de inmediato en la EEPROM. La selección de los parámetros determina si la reparametrización del módulo regulador con la instrucción PID_FM será con o sin discontinuidad.

Tiempos de ejecución de PID_FM

Condiciones marco			Tiempo de ejecución en	
READ_VAR	LOAD_OP	LOAD_PAR	CPU 314	CPU 414-2DP
FALSE	FALSE	FALSE	0,65 ms	0,077 ms
TRUE	FALSE	FALSE	2,85 ms	2,36 ms
TRUE	TRUE	FALSE	4,56 ms	4,48 ms
FALSE	FALSE	TRUE	3,75 ms	2,59 ms
TRUE	FALSE	TRUE	5,95 ms	5,15 ms
TRUE	TRUE	TRUE	7,66 ms	7,1 ms

Consulte también

Descripción de PID_FM (Página 7484)

Modificar los parámetros a través de la HMI (Página 10279)

Parámetros de entrada de PID_FM

Tabla 9-213

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	1	En la entrada "Número de canal" se parametriza el número del canal de regulador al que se refiere el DB de instancia. Se admiten los valores de 1 a 4.
PHASE	4.0	INT	0	El parámetro PHASE puede interconectarse con el parámetro de salida PHASE de las instrucciones TUN_EC o TUN_ES (PID Self-Tuner). En el cuadro de diálogo "Puesta en servicio" se puede visualizar el estado de fase del PID Self-Tuner en texto compresible. Este parámetro es irrelevante para el OP.

Consulte también

Descripción de PID_FM (Página 7484)

Parámetros de salida de PID_FM

Tabla 9-214

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	6.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
out_par	8.0	WORD	W#16#3130	El parámetro out_par no puede ser modificado por el usuario. Identifica el inicio de los parámetros de salida que lee el módulo cuando se activa READ_VAR = TRUE.
SP	10.0	REAL	0.0	En la salida "Consigna" se indica la consigna que actúa efectivamente.
PV	14.0	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	18.0	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.
DISV	22.0	REAL	0.0	En la salida "Variable perturbadora" se indica la magnitud perturbadora que actúa efectivamente. Se admiten valores entre -100 y 100 %.
LMN	26.0	REAL	0.0	En la salida "Variable manipulada" se indica la variable manipulada que actúa efectivamente. En el regulador paso a paso sin realimentación de posición analógica, en el parámetro LMN se indica la acción P + D ilimitada. Se admiten valores entre -100 y 100 %.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMN_A	30.0	REAL	0.0	En la salida "Variable manipulada A de la función de rango partido / realimentación de posición" se indica la variable manipulada A de la función de rango partido en el regulador continuo, y la realimentación de posición en el regulador paso a paso con realimentación de posición analógica. La salida LMN_A sólo puede utilizarse para la indicación aproximada de una magnitud manipulada simulada correspondiente. La variable manipulada LMNRSVAL de la realimentación de posición simulada debe parametrizarse de la forma correspondiente, y es efectiva al activar LMNRS_ON. Se admiten valores entre -100 y 100 %.
LMN_B	34.0	REAL	0.0	Para el regulador continuo, en la salida "Variable manipulada B de la función de rango partido" se indica la variable manipulada B de la función de rango partido. Se admiten valores entre -100 y 100 %.
QH_ALM	38.0	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite superior disparada" se notifica el rebase del límite H_ALM.
QH_WRN	38.1	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite superior disparada" se notifica el rebase del límite H_WRN.
QL_WRN	38.2	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite inferior disparada" se notifica el rebase por defecto del límite L_WRN.
QL_ALM	38.3	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite inferior disparada" se notifica el rebase por defecto del límite L_ALM.
QLMN_HLM	38.4	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de variable manipulada alcanzado" se notifica el rebase del límite superior. (No en el regulador paso a paso sin realimentación de posición analógica)
QLMN_LLM	38.5	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la variable manipulada alcanzado" indica el rebase por defecto del límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica)
QPARA_F	38.6	BOOL	FALSE	El módulo comprueba la validez de los parámetros. Un error de parametrización se indica en la salida "Error de parametrización". Este error de parametrización también se puede ver en el cuadro de diálogo "Online y diagnóstico" del módulo.
QCH_F	38.7	BOOL	FALSE	La salida "Error de canal" se activa cuando el canal de regulación no puede arrojar resultados válidos. El error de canal (p. ej. rotura de hilo) también se activa con QPARA_F = 1 o QMOD_F = 1. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.
QUPRLM	39.0	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación positiva de rampa de subida de la consigna" está activada, se limita la rampa de subida de la consigna.
QDNRLM	39.1	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación negativa de rampa de subida de la consigna" está activada, se limita la rampa de bajada de la consigna.
QSP_HLM	39.2	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de consigna alcanzado" se notifica el rebase del límite superior.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QSP_LLM	39.3	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la consigna alcanzado" indica el rebase del límite inferior.
QLMNUP	39.4	BOOL	FALSE	Se trata de la salida "Señal alta de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QLMNDN	39.5	BOOL	FALSE	Se trata de la salida "Señal baja de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QID	39.6	BOOL	FALSE	QID = TRUE indica que hay una identificación en curso (no que esté activada). Una vez completada la identificación, el resultado puede verse a través del parámetro IDSTATUS de la instrucción CH_DIAG.
QSPOPON	40.0	BOOL	FALSE	En la salida "Control manual de consigna activado" se indica si la consigna se controla con las instrucciones TUN_EC o TUN_ES. Si el bit está activado, el valor SP_OP se aplica como consigna.
QLMNSAFE	40.1	BOOL	FALSE	Si la salida "Modo de seguridad" está activada, se emite la variable manipulada de seguridad como variable manipulada.
QLMNOPON	40.2	BOOL	FALSE	En la salida "Control manual de variable manipulada activado" se indica si la variable manipulada se varía con las instrucciones TUN_EC o TUN_ES. Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
QLMNTRK	40.3	BOOL	FALSE	La salida "Modo de seguimiento" indica si la variable manipulada se corrige mediante una entrada analógica.
QLMN_RE	40.4	BOOL	FALSE	La salida "Manual = 1; Automático = 0" indica si la variable manipulada adopta la variable manipulada externa LMN_RE (Manual = 1) o no.
QLMNR_HS	40.5	BOOL	FALSE	La salida "Señal de tope superior de la realimentación de posición" indica si la servoválvula se encuentra en el tope superior. QLMNR_HS = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
QLMNR_LS	40.6	BOOL	FALSE	La salida "Señal de tope inferior de la realimentación de posición" indica si la servoválvula se encuentra en el tope inferior. QLMNR_LS = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
QLMNR_ON	40.7	BOOL	FALSE	La salida "Realimentación de posición activada" indica que está activado el modo de operación "Regulador paso a paso con realimentación de posición" o "Regulador paso a paso sin realimentación de posición".
QFUZZY	41.0	BOOL	FALSE	Si el parámetro de salida QFUZZY =1, el regulador funciona con el algoritmo difuso.
QSPLEPV	41.1	BOOL	FALSE	La salida "Indicación del regulador difuso: consigna < valor real" se activa con el regulador difuso si la consigna es menor que el valor real efectivo.
QSPR	41.2	BOOL	FALSE	Si la salida "Modo de rango partido" está activada, el regulador continuo trabaja en modo de rango partido.
QMAN_FC	41.4	BOOL	FALSE	La salida "QMAN_FC" se activa en los siguientes casos: El regulador secundario está en modo manual y el regulador piloto se rige por el valor real del secundario. La acción I del regulador piloto se detiene porque la consigna o la magnitud manipulada del regulador secundario están en el límite o porque el regulador secundario se encuentra en modo manual.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QPARABUB	41.7	BOOL	FALSE	El FM activa este parámetro cuando se han cambiado parámetros de operación desde un OP. Si READ_VAR = TRUE y el FM activa esta indicación, la instrucción PID_FM lee los parámetros SP_OP_ON, LMNOP_ON, SP_OP y LMN_OP del FM y los deposita en el DB de instancia. La instrucción adopta así el estado de operación del FM. Tras el proceso de lectura, el parámetro adopta el estado FALSE.
QMOD_F	42.0	BOOL	FALSE	La instrucción comprueba si los procesos de escritura y lectura de un registro han finalizado correctamente. Si se detectan errores, se activa la salida "Error de módulo". La causa puede ser: Una dirección de módulo incorrecta en el parámetro MOD_ADDR, un número de canal incorrecto en el parámetro CHANNEL o un módulo defectuoso.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetros de entrada/salida de PID_FM

Tabla 9-215

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
[1] Valores predeterminados del módulo tras la primera llamada del PID_FM con COM_RST = TRUE				
COM_RST	44.0	BOOL	FALSE	Si el parámetro es COM_RST = TRUE, la instrucción PID_FM realiza una marcha de inicialización. En ella se leen del FM los parámetros del regulador (que son todos los parámetros posteriores a cont_par) y se depositan en el DB de instancia. Además se comprueba la validez de los parámetros MOD_ADDR y CHANNEL. Tras la marcha de inicialización, el parámetro adopta el estado FALSE.
LOAD_OP	44.1	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros operativos en el FM 355" está activado, los parámetros operativos se cargan en el módulo y el parámetro de entrada/salida se desactiva.
READ_VAR	44.2	BOOL	FALSE	Si el parámetro de entrada/salida "Leer variables del FM 355" está activado, los parámetros de salida se leen del módulo y el parámetro de entrada/salida se desactiva.
LOAD_PAR	44.3	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros de regulación en el FM 355" está activado, los parámetros de regulación se cargan en el módulo y el parámetro de entrada/salida se desactiva.
op_par	46.0	WORD	W#16#3130[1]	El parámetro op_par no puede ser modificado por el usuario. Identifica el inicio de los parámetros de operación, que se transfieren al módulo si se activa LOAD_OP = TRUE. El final de los parámetros de operación se identifica con cont_par.
SP_RE	48.0	REAL	0.0	En la entrada "Consigna externa" se conecta una consigna externa con el regulador.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMN_RE	52.0	REAL	0.0	En la entrada "Variable manipulada externa" se conecta una variable manipulada externa con el regulador. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
SP_OP_ON	56.0	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Activar control manual de consigna". Si el bit está activado, el valor SP_OP se aplica como consigna.
SAFE_ON	56.1	BOOL	FALSE	Si la entrada "Adoptar posición de seguridad" está activada, se aplica un valor de seguridad como variable manipulada. Nota: El procesamiento de la señal de salida con LMNDN_OP, LMNUP_OP y LMNSOPON en el regulador paso a paso tiene mayor prioridad que la variable manipulada de seguridad.
LMNOP_ON	56.2	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Activar control manual de variable manipulada". Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
LMNTRKON	56.3	BOOL	FALSE	Si la entrada "Corregir (LMN a través de EA)" está activada, la variable manipulada se rige por una entrada analógica (EA). (No en el regulador paso a paso sin realimentación de posición analógica)
LMN_REON	56.4	BOOL	FALSE	Si la entrada "Aplicar variable manipulada externa" está activada, se aplica la variable manipulada externa LMN_RE como variable manipulada.
LMNRHSRE	56.5	BOOL	FALSE	La señal "Servoválvula en el tope superior" se interconecta a la entrada "Señal de tope superior de la realimentación de posición". LMNRHSRE = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
LMNRLSRE	56.6	BOOL	FALSE	La señal "Servoválvula en el tope inferior" se aplica en la entrada "Señal de tope inferior de la realimentación de posición". LMNRLSRE = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
LMNSOPON	56.7	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Activar control manual de señal de variable manipulada". Si el bit está activado, las señales LMNUP_OP y LMNDN_OP se aplican como señales de variable manipulada. (Sólo en regulador paso a paso)
LMNUP_OP	57.0	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Subir señal de variable manipulada". Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Subir señal de variable manipulada". (Sólo en regulador paso a paso)
LMNDN_OP	57.1	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Bajar señal de variable manipulada". Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Bajar señal de variable manipulada". (Sólo en regulador paso a paso)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMNRS_ON	57.3	BOOL	FALSE	Si no se dispone de una realimentación de posición, puede simularse. La función "Activar simulación de la realimentación de posición" se activa en la entrada. También las instrucciones del PID Self-Tuner tienen acceso a este parámetro, ya que para la optimización se requiere como mínimo una variable manipulada simulada si se ha parametrizado un regulador paso a paso sin realimentación de posición. El valor simulado se muestra en el parámetro LMN_A. Al activar la simulación, el valor del parámetro LMNRSVAL se activa como valor inicial. ATENCIÓN: Con el paso del tiempo, la simulación difiere cada vez más de la realimentación de posición real. (Sólo en el regulador paso a paso sin realimentación de posición analógica)
FUZID_ON	57.4	BOOL	FALSE	En la entrada "Activar identificación de difuso" se activa la identificación del algoritmo difuso.
SP_OP	58.0	REAL	0.0	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Control manual de consigna". Si el bit SP_OP_ON está activado, se aplica el valor "Control manual de consigna" como consigna.
LMN_OP	62.0	REAL	0.0	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Control manual de variable manipulada". Si el bit LMNOP_ON está activado, el valor "Control manual de variable manipulada" se aplica como variable manipulada. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMNRSVAL	66.0	REAL	0.0	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Valor inicial de la realimentación de posición simulada". En ese parámetro se indica el valor inicial de la simulación. (Sólo en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores comprendidos entre -100.0 y 100.0 (%).
cont_par	70.0	WORD	W#16#3130 [1]	El parámetro cont_par no puede ser modificado por el usuario. Identifica el inicio de los parámetros de regulador, que se leen del FM y se depositan en el DB de instancia si COM_RST = TRUE, y que se transmiten al FM si LOAD_PAR = TRUE se activa. El final de los parámetros de regulador es el final del DB de instancia.
P_SEL	72.0	BOOL	TRUE [1]	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción proporcional está conectada si la entrada "Activar acción P" está activada.
PFDB_SEL	72.1	BOOL	FALSE [1]	En el algoritmo PID las acciones P y D se pueden incluir en el lazo de realimentación. La acción proporcional se encuentra en el lazo de realimentación si la entrada "Incluir acción P en lazo de realimentación" está activada.
MONERSEL	72.2	BOOL	FALSE [1]	El regulador dispone de un avisador de valor límite, que puede utilizarse para el valor real o para el error de regulación. Si la entrada "Vigilancia: Valor real = 0, Error de regulación = 1" está activada, se vigila el error de regulación.
D_EL_SEL	74.0	INT	0 [1]	El componente D del algoritmo PID puede depositarse en una entrada separada. Ésta se selecciona en la entrada "Entrada para componente D". 0: Error de regulación 1 a 4: Entrada analógica 1 a 4 17: Valor real negativo, acción D en el lazo de realimentación
SP_HLM	76.0	REAL	100.0 [1]	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior consigna" indica el límite superior. SP_HLM > SP_LLM

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_LLM	80.0	REAL	0.0 1)	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior consigna" indica el límite inferior. SP_LLM < SP_HLM
H_ALM	84.0	REAL	100.0 1)	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más alto. H_ALM > H_WRN
H_WRN	88.0	REAL	90.0 1)	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite superior" indica el segundo límite más alto. Se admiten valores entre H_ALM y L_WRN
L_WRN	92.0	REAL	10.0 1)	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite inferior" indica el segundo límite más bajo. Se admiten valores entre H_WRN y L_ALM
L_ALM	96.0	REAL	0.0 1)	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más bajo. L_ALM < L_WRN
HYS	100.0	REAL	1.0 1)	Para evitar oscilaciones de las indicaciones de vigilancia, puede parametrizarse una histéresis en la entrada "Histéresis". HYS ≥ 0.0
DEADB_W	104.0	REAL	0.0 1)	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. DEADB_W ≥ 0.0
GAIN	108.0	REAL	1.0 1)	La entrada "Ganancia proporcional" indica la ganancia del regulador.
TI	112.0	REAL	3000.0 1)	La entrada "Tiempo de integración" determina el comportamiento temporal del integrador. Con TI = 0 el integrador está desactivado TI = 0.0 o bien TI ≥ 0.5
TD	116.0	REAL	0.0 1)	La entrada "Tiempo de acción derivada" determina el comportamiento temporal del diferenciador. Con TD = 0 el diferenciador está desactivado TD = 0.0 o bien TD ≥ 1.0
TM_LAG	120.0	REAL	5.0 1)	El algoritmo de la acción D contiene un retardo que se puede parametrizar en la entrada "Tiempo de retardo de la acción D". TM_LAG ≥ 0.5
LMN_SAFE	124.0	REAL	0.0 1)	Para la variable manipulada puede parametrizarse un valor de seguridad en la entrada "Variable manipulada de seguridad". Se admiten valores comprendidos entre -100.0 y 100.0 (%)
LMN_HLM	128.0	REAL	100.0 1)	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior de la variable manipulada" indica el límite superior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores comprendidos entre LMN_LLM y 100.0 (%)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMN_LLM	132.0	REAL	0.0 1)	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior de la variable manipulada" indica el límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre -100.0 y LMN_HLM (%)
MTR_TM	136.0	REAL	60.0 1)	En el parámetro "Tiempo de posicionamiento del motor" se introduce el tiempo de operación de la servoválvula de un tope al otro. (Sólo en regulador paso a paso) Se admiten valores de MTR_TM ≥ 0.001
PULSE_TM	140.0	REAL	0.2 1)	En el parámetro "Duración mínima de impulso" se puede parametrizar una duración de impulso mínima. (Sólo en regulador paso a paso o de impulsos) Se admiten valores PULSE_TM ≥ 0.0
BREAK_TM	144.0	REAL	0.2 1)	En el parámetro "Duración mínima de pausa" se puede parametrizar una duración de pausa mínima. (Sólo en regulador paso a paso o de impulsos) Se admiten valores BREAK_TM ≥ 0.0

Nota

Con LOAD_PAR = TRUE todos los parámetros del regulador se cargan de forma permanente en la EEPROM del FM 355.

Con LOAD_OP = TRUE, de los parámetros operativos sólo la consigna SP_RE se carga permanentemente en la EEPROM del FM 355. A todos los parámetros operativos restantes se les asignan los valores 0 o FALSE en el arranque del FM 355.

La EEPROM del módulo podría quedar destruida por un número excesivo de procesos de escritura. Para evitarlo, tras la escritura de la EEPROM el módulo retrasa una nueva escritura 30 minutos.

Consulte también

Descripción de PID_FM (Página 7484)

Diagramas de bloques de PID_FM

Sinopsis

En las siguientes figuras se representan los puntos en los que son efectivos los parámetros de la instrucción PID_FM.

Los parámetros son efectivos en los mismos puntos en los reguladores de tres componentes, de relación, mixtos, de valor fijo o cascada. Esto también rige para los parámetros que están presentes por igual en el regulador continuo, en el regulador con salida de impulsos y en el

regulador paso a paso. Por ello no se representan todos los diagramas de bloques, y no todos los parámetros están incluidos en todos los diagramas.

Sin embargo, todos los parámetros de la instrucción PID_FM están contenidos en los diagramas, excepto los parámetros MOD_ADDR, CHANNEL, QMOD_F, QPARA_F, QCH_F, QLMNR_ON, RET_VALU, COM_RST, LOAD_PAR, READ_VAR, LOAD_OP.

Efecto de los parámetros de entrada

Cálculo del error de regulación en el regulador de valor fijo o cascada

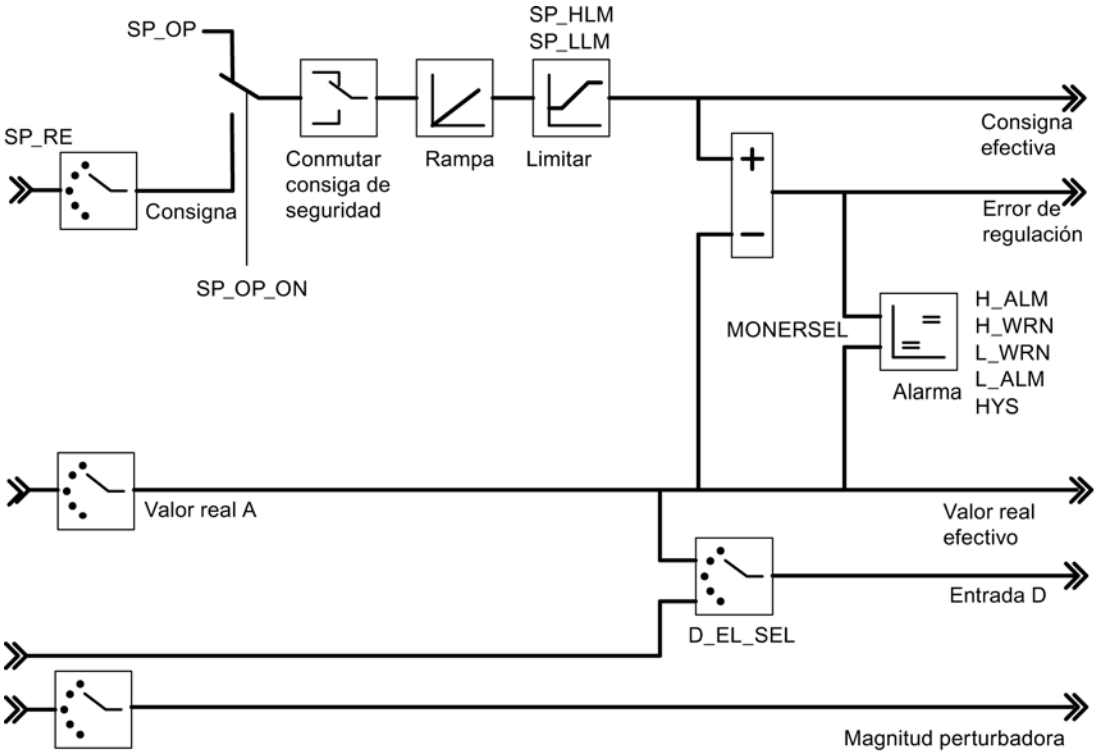
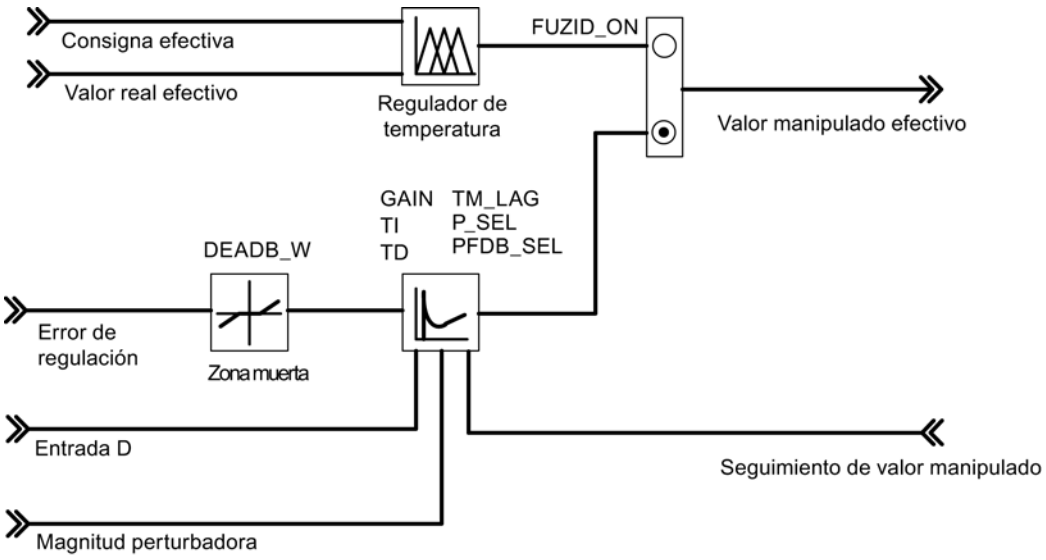
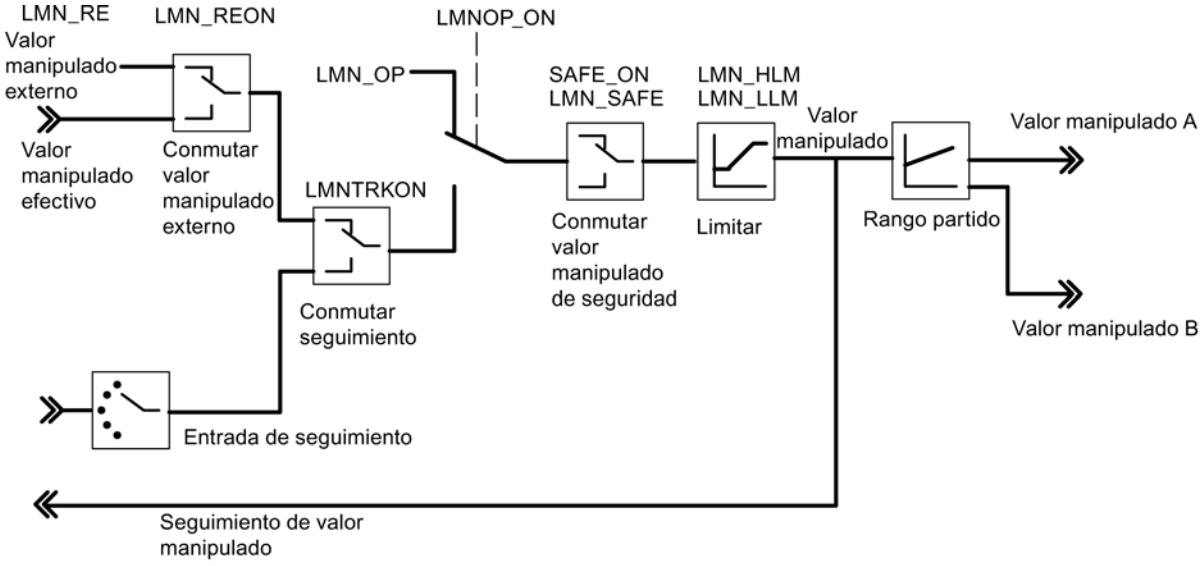


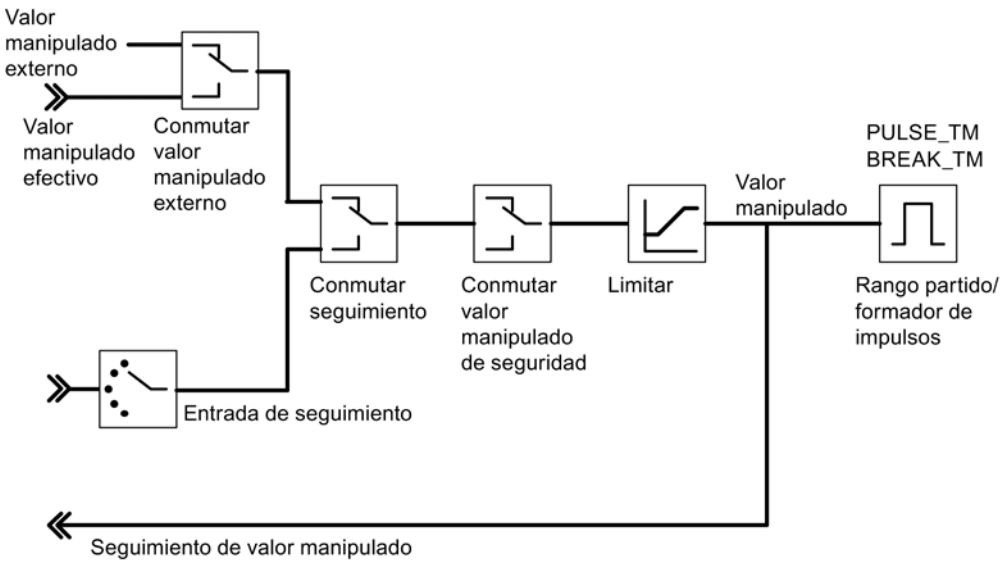
Diagrama de bloques del algoritmo de regulación



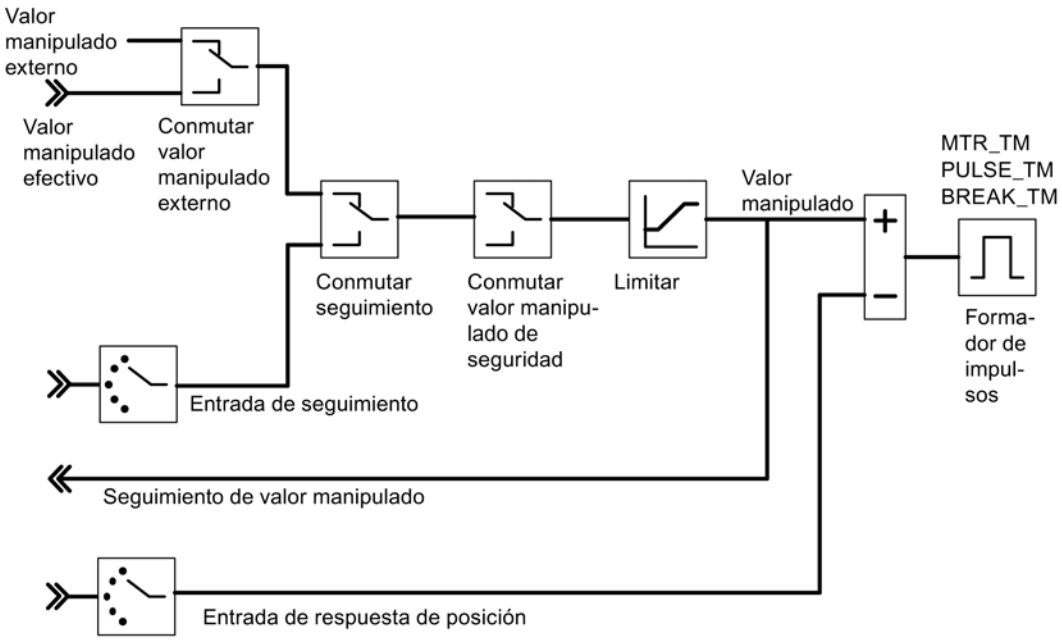
Salida de regulador del regulador K



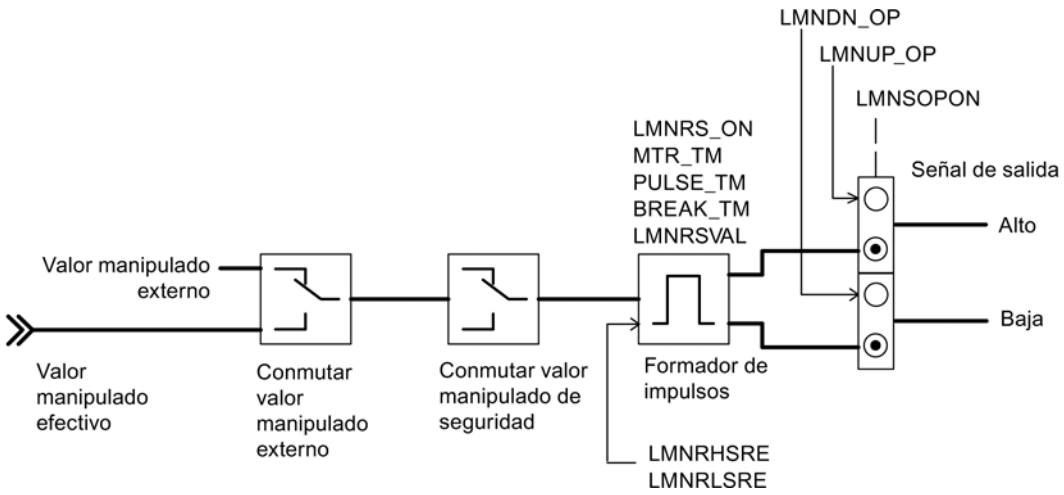
Salida de regulador del regulador S (modo de operación Regulador de impulsos)



Salida de regulador del regulador S (modo de operación Regulador paso a paso con realimentación de posición)



Salida de regulador del regulador S (modo de operación Regulador paso a paso sin realimentación de posición)



Generación de los parámetros de salida

En las siguientes figuras se representan los puntos del módulo en los que se generan los parámetros de salida de la instrucción PID_FM.

Cálculo del error de regulación en el regulador de valor fijo o cascada

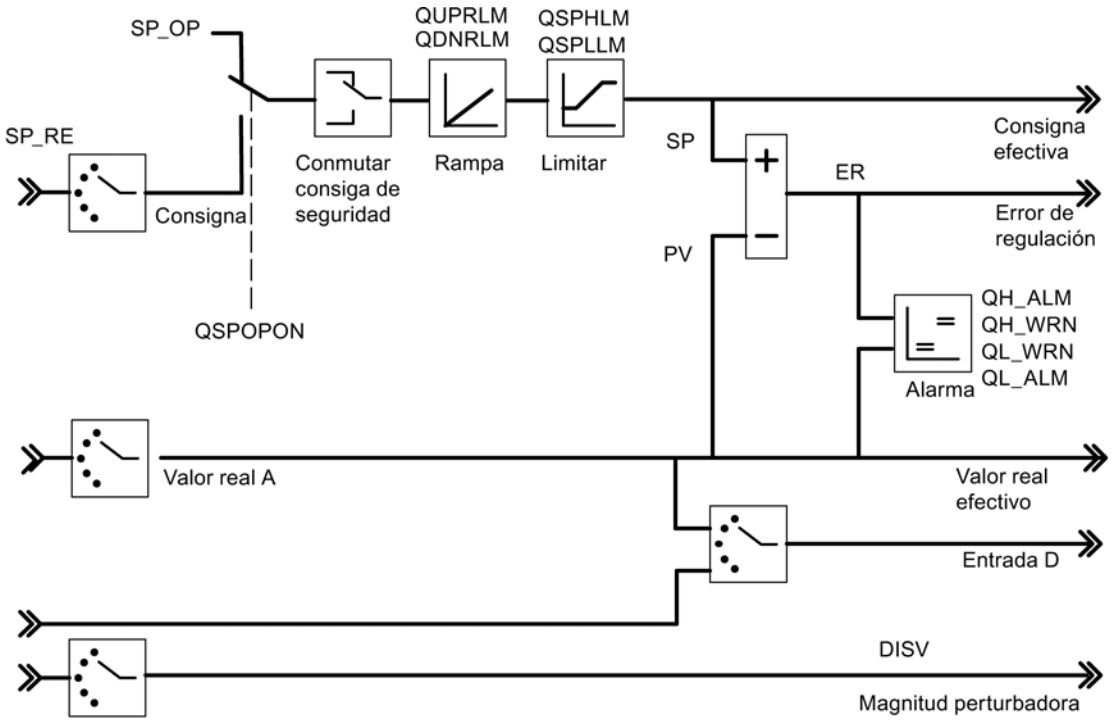
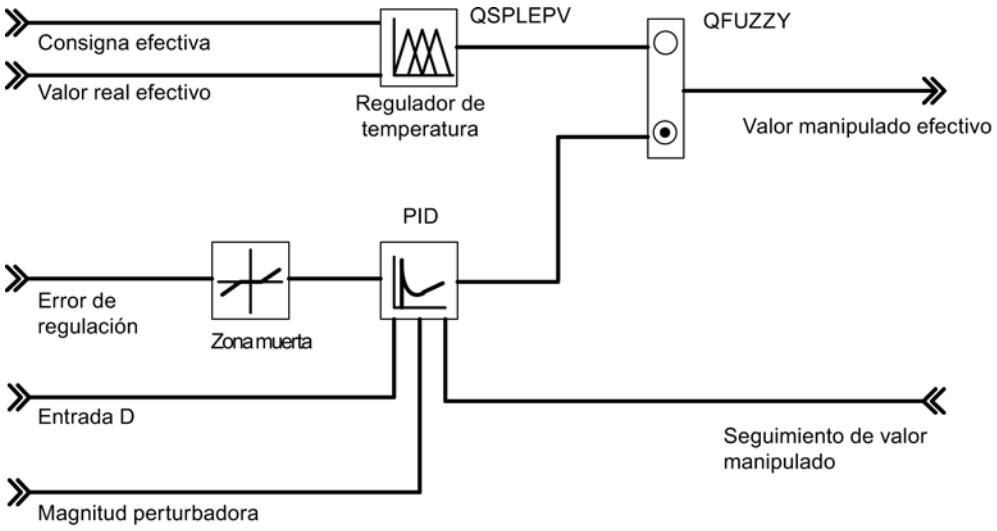
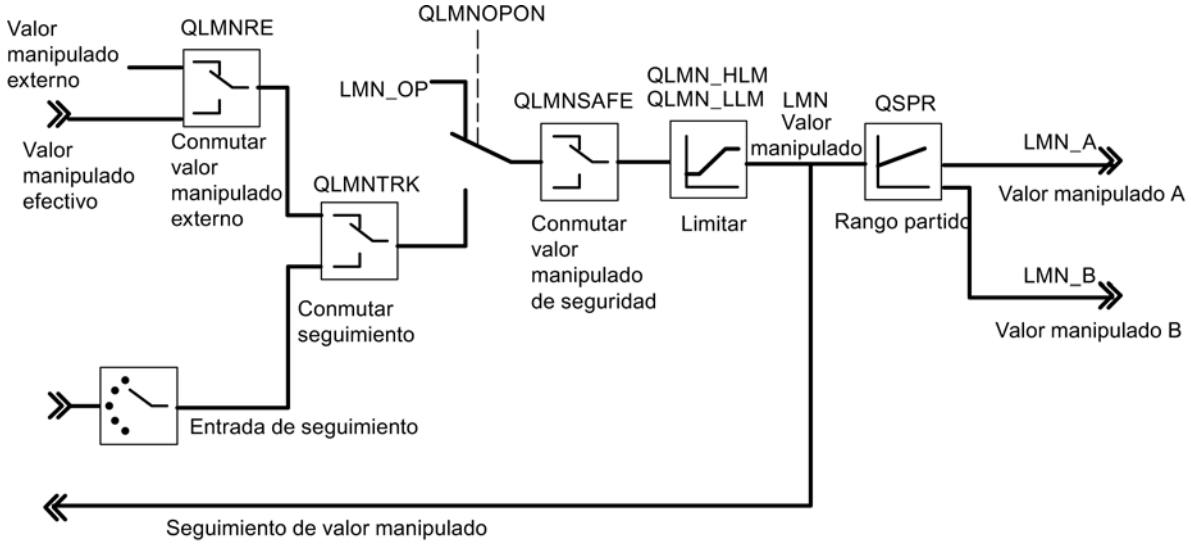


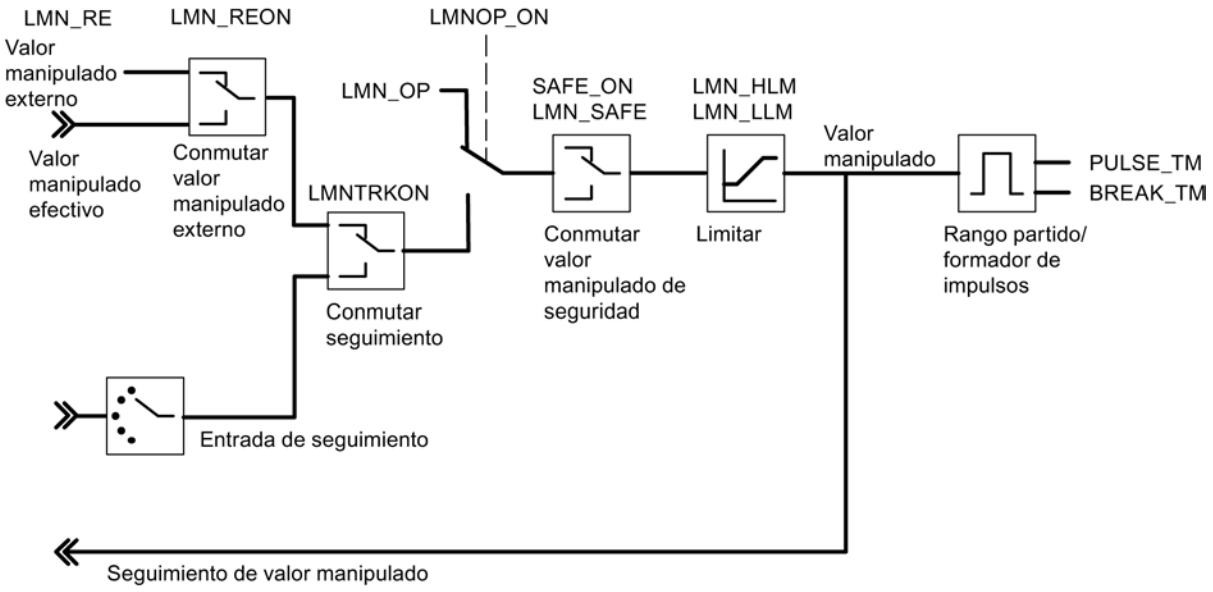
Diagrama de bloques del algoritmo de regulación



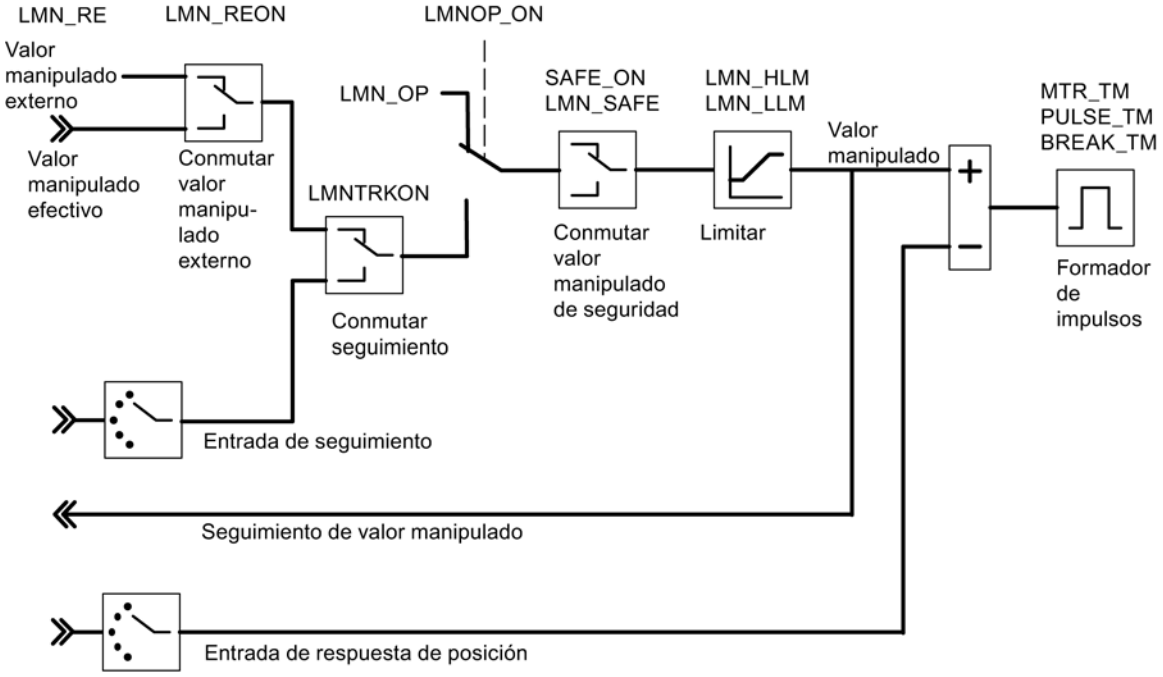
Salida de regulador del regulador K



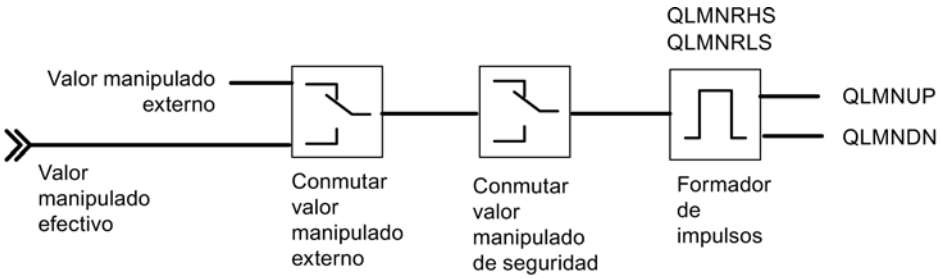
Salida de regulador del regulador S (modo de operación Regulador de impulsos)



Salida de regulador del regulador S (modo de operación Regulador paso a paso con realimentación de posición)



Salida de regulador del regulador S (modo de operación Regulador paso a paso sin realimentación de posición)



Consulte también

Descripción de PID_FM (Página 7484)

FUZ_355

Descripción de FUZ_355

La instrucción FUZ_355 permite leer y escribir parámetros de regulador del regulador de temperatura difuso del FM 355.

Esta función es idónea para las siguientes aplicaciones:

- Transmisión al FM 355 de los parámetros de regulador determinados por identificación tras un cambio de módulo
- Adaptación del FM 355 a diferentes procesos de regulación

Nota

Los parámetros determinados por el FM 355 mediante una identificación no se pueden cambiar, ya que están optimizados para el proceso de regulación.

Funcionamiento

Con los parámetros READ_PAR y LOAD_PAR se controla el funcionamiento de la instrucción FUZ_355.

Si READ_PAR = TRUE, FUZ_355 lee los parámetros de todos los reguladores de temperatura del FM 355 y los deposita en el DB de instancia. Una vez que la lectura de los parámetros del regulador de temperatura se ha completado correctamente, FUZ_355 pone el parámetro READ_PAR en FALSE.

Si LOAD_PAR = TRUE, FUZ_355 escribe los parámetros de todos los reguladores de temperatura del FM 355 del DB de instancia en el FM 355. Una vez que la transferencia de los parámetros se ha completado correctamente, FUZ_355 pone el parámetro LOAD_PAR en FALSE.

Llamada

FUZ_355 no requiere una marcha de inicialización.

Si se utiliza el FM 355 en la periferia descentralizada, pueden ser necesarios varios ciclos de llamada hasta que los parámetros READ_PAR y LOAD_PAR vuelvan a adoptar el estado FALSE. Por ello, llame la instrucción FUZ_355 de forma limitada mientras READ_PAR = TRUE o LOAD_PAR = TRUE.

FUZ_355 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355.

Arranque

En el arranque de la CPU se debe activar el parámetro LOAD_PAR de la instrucción FUZ_355, y a continuación llamar el bloque en el programa cíclico mientras LOAD_PAR = TRUE.

Parámetros de entrada de FUZ_355

Tabla 9-216

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de FUZ_355

Tabla 9-217

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	2.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
PARAFFUZ	4.0	WORD	0	En el parámetro PARAFFUZ, un error de parametrización generado por la instrucción FUZ_355 se indica de la siguiente manera: High Byte de PARAFFUZ = 01: Se ha producido un error de parametrización. High Byte de PARAFFUZ = 00: No se ha producido ningún error de parametrización. En el Low Byte se indica el offset del parámetro que ha causado el error de parametrización, contado a partir de la variable estática FUZ_PAR[1]. PARAFFUZ = W#16#0104 significa p. ej. que el segundo parámetro es erróneo

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetros de entrada/salida de FUZ_355

Tabla 9-218

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
READ_PAR	6.0	BOOL	FALSE	Si el parámetro READ_PAR se activa, se leen los parámetros difusos del módulo y se depositan en variables estáticas del DB de instancia.
LOAD_PAR	6.1	BOOL	FALSE	Si el parámetro LOAD_PAR se activa, se leen los parámetros difusos de las variables estáticas del DB de instancia y se transfieren al módulo.

FORCE355

Descripción de FORCE355

La instrucción FORCE355 permite simular (forzar) los valores de entrada analógicos y digitales para facilitar la puesta en marcha.

Simulación de los valores digitales

La simulación de los valores para las entradas digitales uno a ocho se activa a través de los conmutadores S_DION[i], donde 1 ≤ i ≤ 8. Los valores de simulación se predeterminan a través de los parámetros DI_SIM[i]. Si S_DION[i] = TRUE, en lugar del valor de la entrada digital i del módulo se utiliza el valor DI_SIM[i]. Los LEDs I1 a I8 muestran siempre el estado de la entrada digital correspondiente, incluso durante la simulación.

Simulación de los valores analógicos

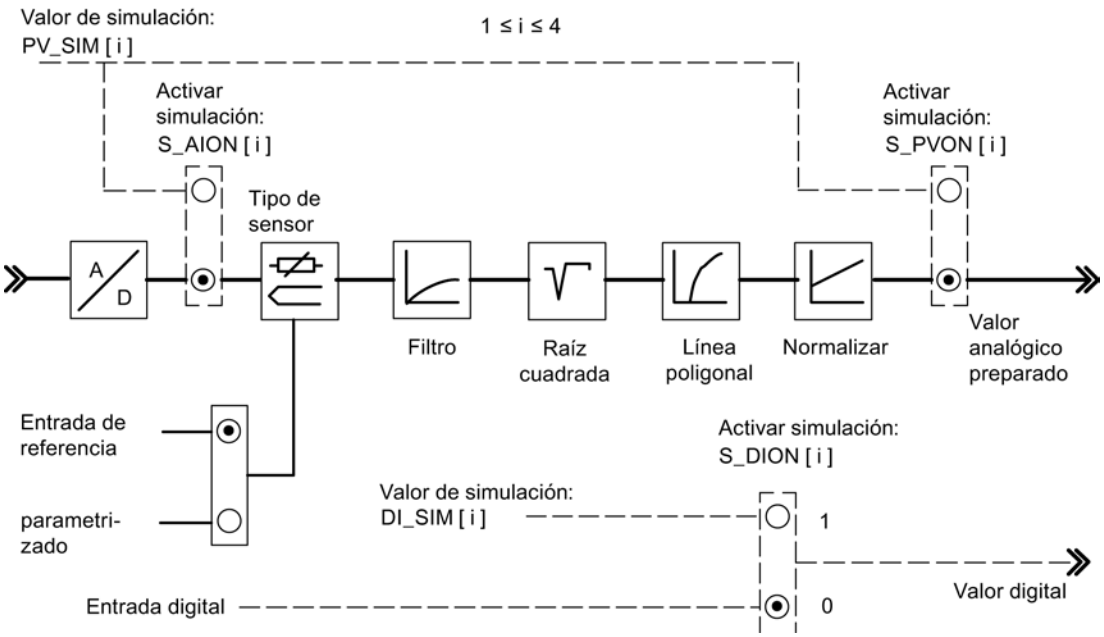
La simulación de los valores analógicos para los canales uno a cuatro se activa a través de los conmutadores S_AION[i] o S_PVON[i], donde 1 ≤ i ≤ 4.

Los valores de simulación para los canales uno a cuatro se predeterminan a través de los parámetros PV_SIM[i].

Los valores de simulación pueden hacerse efectivos en dos puntos:

- S_AION[i] = TRUE (1 ≤ i ≤ 4)
  En lugar del valor de la entrada analógica i del módulo se utiliza el valor PV_SIM[i].
- S_PVON[i] = TRUE (1 ≤ i ≤ 4)
  En lugar del valor acondicionado de la entrada analógica i del módulo se utiliza el valor PV_SIM[i].

La siguiente figura muestra en qué punto es efectivo el valor analógico simulado.



La conexión y la preasignación de los valores de simulación (forzado) no se realizan a través de la interfaz de parametrización. Por ello, los conmutadores y las líneas de conexión correspondientes se representan rayados.

Llamada

FORCE355 no requiere una marcha de inicialización y normalmente se llama cíclicamente.

FORCE355 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355.

Comportamiento en arranque

En caso de rearranque del FM 355 tras Power OFF, los conmutadores de simulación del FM 355 vuelven a estar en FALSE.

Parámetros de entrada del FORCE355

Tabla 9-219

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
S_AION	0.0	ARRAY [1..4] of BOOL	FALSE	Si p. ej. el conmutador S_AION[1] adopta el estado TRUE, en lugar del valor de entrada analógica 1 del módulo se utiliza el valor PV_SIM[1].
S_PVON	2.0	ARRAY [1..4] of BOOL	FALSE	Si p. ej. el selector S_PVON[1] adopta el estado TRUE, en lugar del valor acondicionado de entrada analógica 1 del módulo se utiliza el valor PV_SIM[1].

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PV_SIM	4.0	ARRAY [1..4] of REAL	0.0	En la entrada PV_SIM[1] se predetermina p. ej., el valor de simulación para la entrada analógica 1. Si S_PVON = TRUE, entonces se especifica aquí el valor acondicionado de entrada analógica. Si S_PVON = FALSE y S_AION = TRUE, entonces se especifica aquí el valor de entrada analógica en mA o mV, que se convierte en un valor acondicionado mediante funciones de acondicionamiento. Se admiten valores entre 0.0 y 20.0 [mA], -1500 a +10000 [mV] o un rango de valores técnico
S_DION	20.0	ARRAY [1..8] of BOOL	FALSE	Si p. ej. S_DION[1] se pone en TRUE, en lugar de la entrada digital 1 del módulo se utiliza el valor DI_SIM[1] como valor digital.
DI_SIM	22.0	ARRAY [1..8] of BOOL	FALSE	Valor de simulación para entrada digital
MOD_ADDR	24.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de FORCE355

Tabla 9-220

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	26.0	WORD		RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

Consulte también

RDREC: Leer registro (Página 6735)

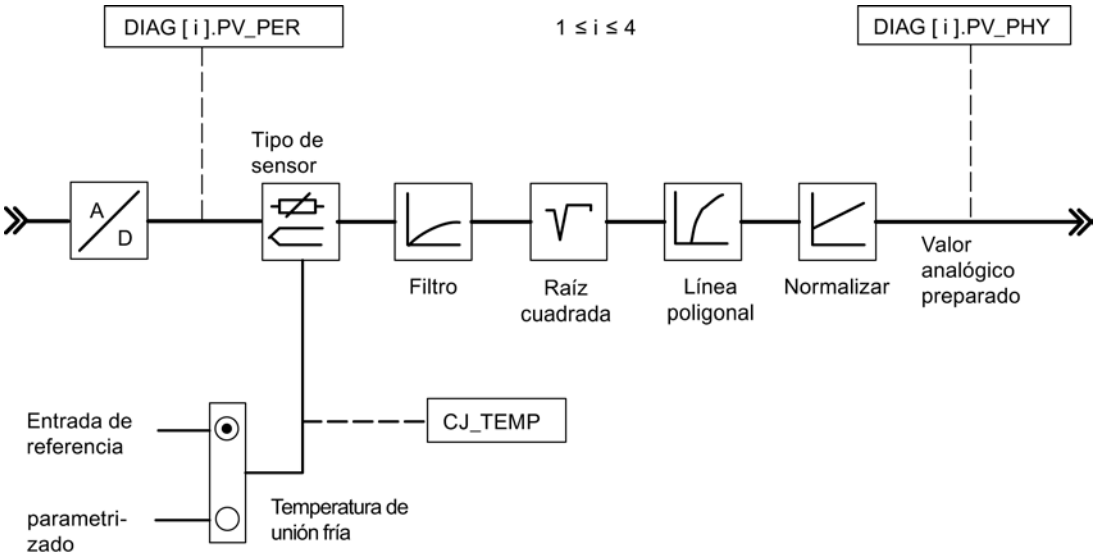
WRREC: Escribir registro (Página 6738)

READ_355

Descripción de READ_355

La instrucción READ_355 permite leer los valores de entrada digital y analógicos para ayudar en la puesta en marcha.

Se visualizan los valores siguientes.



Llamada

READ_355 no requiere una marcha de inicialización y normalmente se llama cíclicamente.

READ_355 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355.

Parámetros de entrada de READ_355

Tabla 9-221

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de READ_355

Tabla 9-222

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
CJ_TEMP	2.0	REAL	0.0	En el parámetro CJ_TEMP está depositada la temperatura medida en la unión fría en grados °C o °F (dependiendo de la unidad de temperatura parametrizada). Si no se ha parametrizado ningún tipo de sensor "Termopar" o si en todas las entradas analógicas se ha seleccionado la temperatura parametrizada para la unión fría, en el parámetro CJ_TEMP se muestra 0.0.
STAT_DI	6.0	ARRAY [1..8] of BOOL	FALSE	En los parámetros STAT_DI[1] a STAT_DI[8] se muestran los estados reales de las entradas digitales 1 a 8, aunque sean simuladas.
DIAG[x].PV_PER	(Número de canal) x 8	ARRAY [1..4] of STRUCT	0.0	En los parámetros DIAG[1].PV_PER a DIAG[4].PV_PER se muestra el valor de las entradas analógicas uno a cuatro en la unidad mA o mV. Si se ha activado la simulación del valor de entrada analógico a través de la instrucción FORCE355, se muestra el valor simulado.
DIAG[x].PV_PHY	(Número de canal) x 8 + 4	ARRAY [1..4] of STRUCT	0.0	En los parámetros DIAG[1].PV_PHY a DIAG[4].PV_PHY se muestra el valor acondicionado de entrada analógica uno a cuatro en unidad física. Si se ha activado la simulación del valor analógico físico acondicionado mediante la instrucción FORCE355, se muestra el valor simulado.
RET_VALU	40.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

CH_DIAG

Descripción de CH_DIAG

La instrucción CH_DIAG permite leer del módulo variables de diagnóstico adicionales específicas de canal.

Llamada

CH_DIAG no requiere una marcha de inicialización y normalmente se llama cíclicamente.

CH_DIAG debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355.

Parámetros de entrada de CH_DIAG

Tabla 9-223

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	1	En la entrada "Número de canal" se parametriza el número del canal de regulador al que se refiere el DB de instancia. Se admiten los valores 1 a 4.

Consulte también

Descripción de CH_DIAG (Página 7508)

Parámetros de salida de CH_DIAG (Página 7509)

Parámetro IDSTATUS (Página 7510)

Diagrama de bloques de CH_DIAG (Página 7511)

Parámetros de salida de CH_DIAG

Tabla 9-224

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_R	4.0	REAL	0.0	Con el regulador de relación ajustado, el parámetro tiene asignado el valor de entrada de la consigna.
PV_R	8.0	REAL	0.0	El parámetro sólo tiene asignado el siguiente valor con el regulador de relación ajustado: (Valor real A - Consigna Offset) / Valor real D
DIF_I	12.0	REAL	0.0	En el parámetro DIF_I se muestra la magnitud de entrada de la acción D. Esto resulta conveniente p. ej. si como magnitud de entrada de la acción D se ha parametrizado una entrada analógica.
TRACKPER	16.0	REAL	0.0	En el parámetro TRACKPER se muestra la magnitud de entrada con la que se corrige la magnitud manipulada si el regulador está ajustado para seguimiento de magnitud manipulada.
IDSTATUS	20.0	WORD	0.0	En Parámetro IDSTATUS se muestra el estado de la identificación.
LMN_P	22.0	REAL	0.0	En el parámetro LMN_P se muestra la acción P de la magnitud manipulada.
LMN_I	26.0	REAL	0.0	En el parámetro LMN_I se muestra la acción I de la magnitud manipulada.
LMN_D	30.0	REAL	0.0	En el parámetro LMN_D se muestra la acción D de la magnitud manipulada.
RET_VALU	34.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

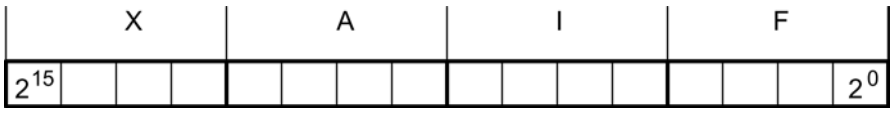
Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetro IDSTATUS

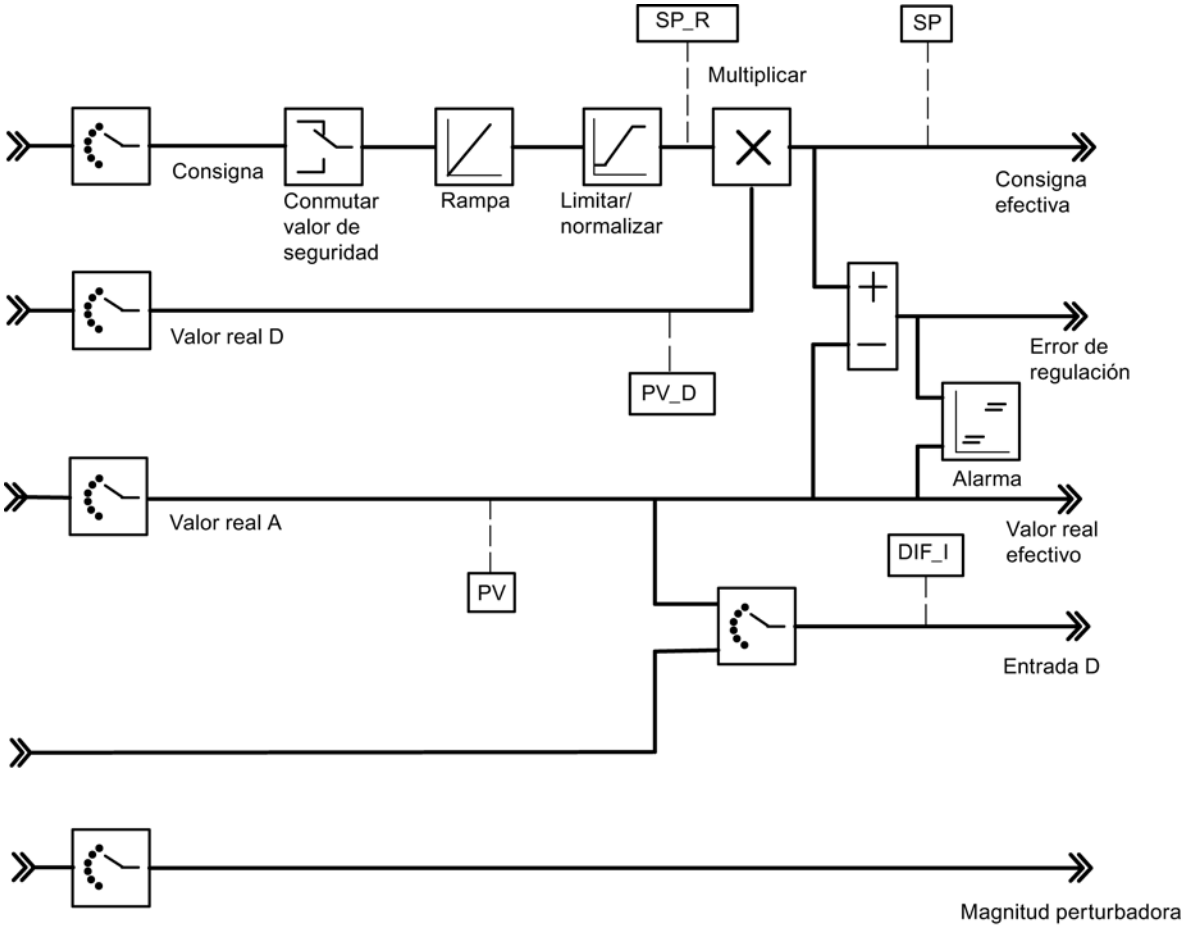
El parámetro IDSTATUS proporciona información sobre el estado de la identificación y contiene los cuatro valores hexadecimales X, A, I y F.



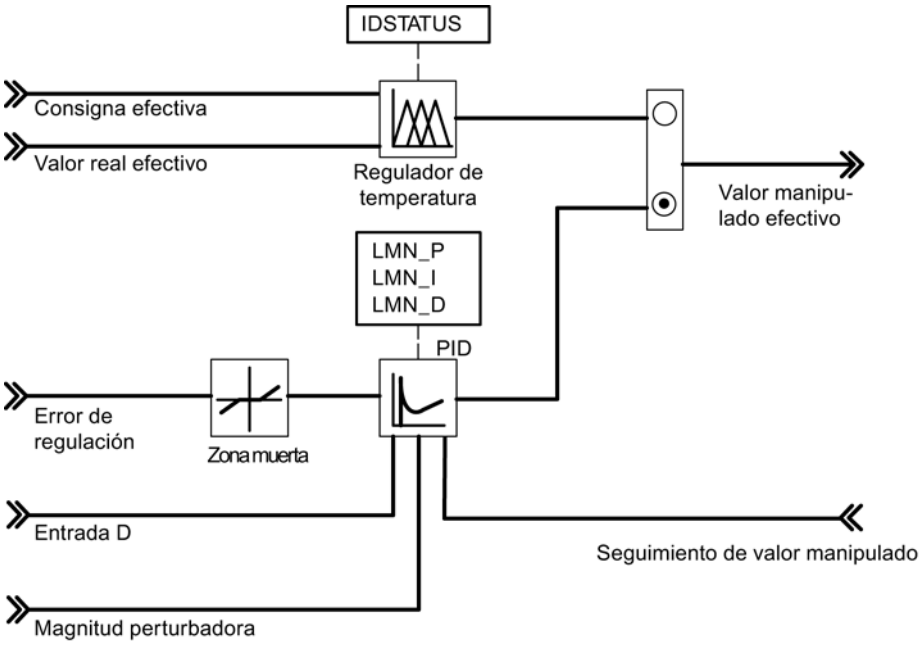
Valor	Descripción
X	Irrelevante (siempre 0)
A	Número de acción • 0 = modo manual (o bien sin modo de regulación); • 2 = regulación; • 4 = optimización activada (FUZID_ON = true); • 6 = estado de transición de modo manual a 2 ó 4;
I	Indicación "Identificación en curso" y "Los parámetros se han determinado pero aún no se han guardado en la EEPROM" • 0 = la identificación no se ejecuta, no se han determinado nuevos parámetros • 1 = la identificación está en curso, no se han determinado nuevos parámetros • 2 = la identificación no se ejecuta, se han determinado nuevos parámetros, pero aún no se han guardado en la EEPROM • 3 = la identificación está en marcha, se han determinado nuevos parámetros, pero aún no se han guardado en la EEPROM
F	Número de error. • 0 = no hay error • 4 = salto excesivo del valor real durante la identificación • 5 = relación entre tiempo de retardo y constante de tiempo de proceso excesiva o comportamiento del proceso regulado muy distante del lineal. • 6 = caída o subida de temperatura excesiva en el inicio de la identificación. El proceso no tiene la calma necesaria

Diagrama de bloques de CH_DIAG

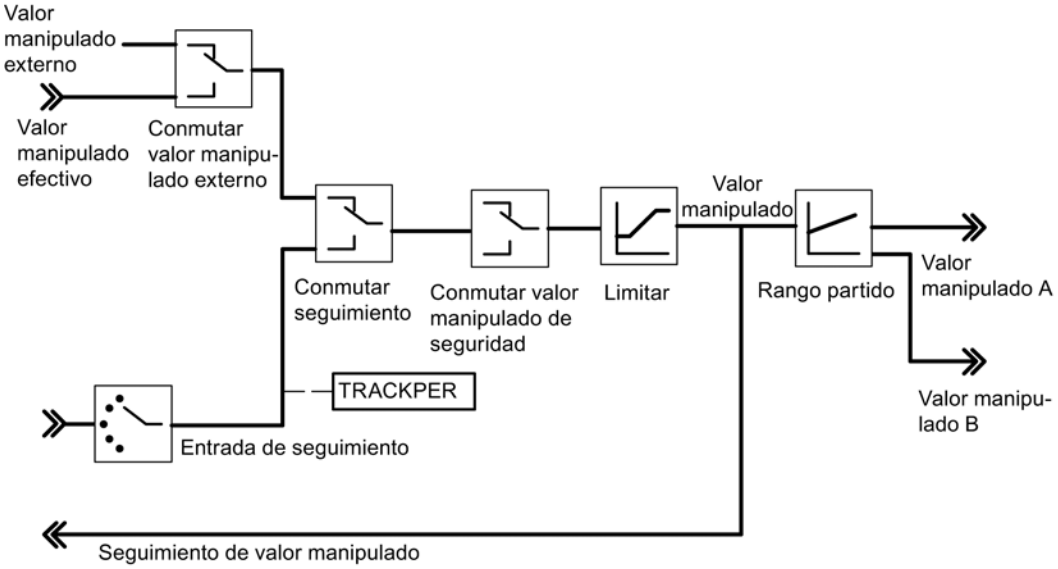
Se muestran los siguientes valores de diagnóstico del error de regulación.



Se muestran los siguientes valores del algoritmo de regulación.



Se muestran los siguientes valores del regulador K o S.



PID_PAR

Descripción de PID_PAR

La instrucción PID_PAR permite modificar parámetros online que no pueden predeterminarse con la instrucción PID_FM. Los parámetros modificables pueden verse en la tabla Parámetros modificables (Página 7515).

Llamada

PID_PAR requiere una marcha de inicialización. Se inicia automáticamente si la instrucción PID_PAR aún no ha leído ninguna vez los datos del sistema (datos preasignados del SDB del FM355). También es posible iniciar la inicialización manualmente con COM_RST=TRUE, cosa que se suele hacer en el OB 100, ya que después de STOP-RUN de la CPU los datos de sistema se transmiten al FM355. El proceso de inicialización dura varios ciclos. Durante la inicialización no se transmiten datos a través de WRREC al FM 355. Tras la inicialización, el bloque desactiva automáticamente el parámetro COM_RST.

Para reducir el tiempo de funcionamiento, PID_PAR no debería llamarse cíclicamente, sino sólo cuando fuera necesario cambiar parámetros. En ese caso, COM_RST debe ser FALSE.

PID_PAR debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355.

Si se utiliza el FM 355 en periferia descentralizada, pueden ser necesarios varios ciclos hasta que los parámetros se hayan transmitido al FM 355. Hasta que la transmisión no haya finalizado, el parámetro BUSY tiene el valor TRUE. PID_PAR debería llamarse de forma repetida hasta que BUSY = FALSE y RET_VALU = 0.

Funcionamiento

En el curso de la inicialización, PID_PAR lee los parámetros de los datos de sistema y los deposita en variables estáticas. PID_PAR cambia en cada llamada una de estas variables REAL y una de las variables INT. En el parámetro de entrada INDEX_R o INDEX_I indique el número de índice del parámetro que desea modificar. Indique los nuevos valores en el parámetro de entrada VALUE_R o VALUE_I. PID_PAR transfiere el registro completo con las variables modificadas al FM 355.

Para cambiar otros parámetros, llame **el mismo** DB de instancia varias veces seguidas con COM_RST = FALSE y con diferentes números de índice.

El parámetro COM_RST es un parámetro de entrada que no se desactiva con PID_PAR.

Nota

Recuerde que los parámetros que ha cambiado con PID_PAR se sobrescriben en el arranque con los parámetros de los datos del sistema.

Arranque

En el arranque de la CPU y en la transición de STOP a RUN de la CPU, los parámetros del FM 355 se sobrescriben con los parámetros del SDB de la CPU.

Ejemplo

Desea cambiar el tiempo de subida de la rampa para la magnitud de referencia durante el funcionamiento, y además utilizar diferentes valores de entrada analógica como valor real independientemente del estado del proceso.

- En el arranque de la CPU llame PID_PAR con COM_RST = TRUE.
- Para parametrizar el tiempo de subida de la rampa para la referencia a 10.0 llame PID_PAR durante el funcionamiento con INDEX_R = 30, VALUE_R = 10.0 y INDEX_I = 0.
- Para parametrizar el valor de entrada analógica 4 del módulo como valor real, llame PID_PAR durante el funcionamiento con INDEX_R = 0, INDEX_I = 50 y VALUE_I = 4.

Consulte también

Parámetros modificables (Página 7515)

Parámetros de entrada de PID_PAR (Página 7514)

Parámetros de salida de PID_PAR (Página 7515)

Parámetros de entrada de PID_PAR

Tabla 9-225

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	0.0	BOOL	TRUE	Si el parámetro es COM_RST = TRUE, la instrucción PID_PAR realiza una marcha de inicialización. En ella se leen los parámetros de los datos del sistema de la CPU y se depositan en el DB de instancia.
MOD_ADDR	2.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	4.0	INT	1	En la entrada "Número de canal" se parametriza el número del canal de regulador al que se refiere el DB de instancia. Se admiten valores comprendidos entre 1 y 4.
INDEX_R	6.0	INT	0.0	• 0: No se modifica ningún parámetro REAL • 1 a 48: Cada valor corresponde a un parámetro REAL modificable.
VALUE_R	8.0	REAL	0.0	En función del parámetro en cuestión
INDEX_I	12.0	INT	0.0	• 0: No se modifica ningún parámetro INT • 49 a 61: Cada valor corresponde a un parámetro INT modificable.
VALUE_I	14.0	INT	0.0	En función del parámetro en cuestión

Consulte también

Descripción de PID_PAR (Página 7512)

Parámetros de salida de PID_PAR (Página 7515)

Parámetros de salida de PID_PAR

Tabla 9-226

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	16.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
BUSY	18.0	BOOL	FALSE	Si BUSY = TRUE, el módulo aún no ha adoptado los parámetros (con periferia descentralizada). En ese caso, la llamada de la instrucción PID_PAR debería repetirse en el siguiente ciclo.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetros modificables

Tabla 9-227 Lista de los parámetros REAL e INT que se pueden modificar con PID_PAR en un FM 355

Tipo de datos	Descripción	Número de índice
–	Ningún parámetro seleccionado	0
REAL	Constante de tiempo de filtro para entrada analógica	1
REAL	Fin de medición (100%)	2
REAL	Inicio de medición (0%)	3
REAL	Línea poligonal, valor de interpolación 1 lado de entrada	4
REAL	Línea poligonal, valor de interpolación 2 lado de entrada	5
REAL	Línea poligonal, valor de interpolación 3 lado de entrada	6
REAL	Línea poligonal, valor de interpolación 4 lado de entrada	7
REAL	Línea poligonal, valor de interpolación 5 lado de entrada	8
REAL	Línea poligonal, valor de interpolación 6 lado de entrada	9
REAL	Línea poligonal, valor de interpolación 7 lado de entrada	10
REAL	Línea poligonal, valor de interpolación 8 lado de entrada	11
REAL	Línea poligonal, valor de interpolación 9 lado de entrada	12
REAL	Línea poligonal, valor de interpolación 10 lado de entrada	13
REAL	Línea poligonal, valor de interpolación 11 lado de entrada	14
REAL	Línea poligonal, valor de interpolación 12 lado de entrada	15
REAL	Línea poligonal, valor de interpolación 13 lado de entrada	16
REAL	Línea poligonal, valor de interpolación 1 lado de salida	17
REAL	Línea poligonal, valor de interpolación 2 lado de salida	18
REAL	Línea poligonal, valor de interpolación 3 lado de salida	19

Tipo de datos	Descripción	Número de índice
REAL	Línea poligonal, valor de interpolación 4 lado de salida	20
REAL	Línea poligonal, valor de interpolación 5 lado de salida	21
REAL	Línea poligonal, valor de interpolación 6 lado de salida	22
REAL	Línea poligonal, valor de interpolación 7 lado de salida	23
REAL	Línea poligonal, valor de interpolación 8 lado de salida	24
REAL	Línea poligonal, valor de interpolación 9 lado de salida	25
REAL	Línea poligonal, valor de interpolación 10 lado de salida	26
REAL	Línea poligonal, valor de interpolación 11 lado de salida	27
REAL	Línea poligonal, valor de interpolación 12 lado de salida	28
REAL	Línea poligonal, valor de interpolación 13 lado de salida	29
REAL	Tiempo de subida de la rampa para referencia	30
REAL	Consigana de seguridad o consigna de seguridad.	31
REAL	Offset de combinación de consigna (regulador de relación/mixto)	32
REAL	Factor para valor real B (regulador de tres componentes)	33
REAL	Factor para valor real C (regulador de tres componentes)	34
REAL	Offset para combinación de valor real (regulador de tres componentes)	35
REAL	Factor para combinación de referencia	36
REAL	Punto de operación	37
REAL	Agresividad en el regulador difuso	38
REAL	Vértices de la función de rango partido: Inicio de rango señal de entrada A	39
REAL	Vértices de la función de rango partido: Fin de rango señal de entrada A	40
REAL	Vértices de la función de rango partido: Inicio de rango señal de salida A	41
REAL	Vértices de la función de rango partido: Fin de rango señal de salida A	42
REAL	Vértices de la función de rango partido: Inicio de rango señal de entrada B	43
REAL	Vértices de la función de rango partido: Fin de rango señal de entrada B	44
REAL	Vértices de la función de rango partido: Inicio de rango señal de salida B	45
REAL	Vértices de la función de rango partido: Fin de rango señal de salida B	46
REAL	Duración mínima del impulso	47
REAL	Duración mínima de pausa	48
INT	Selección de la referencia SP o SP_RE para el regulador 0: Consigna SP_RE de la instrucción 1 a 4: Valor de entrada analógica 1 a 4 17 a 20: Magnitud manipulada (LMN) de regulador 1 a 4	49
INT	Selección de la magnitud reguladora principal valor real A para el regulador 0: Valor real A = 0.0 1 a 4: Valor de entrada analógica 1 a 4	50
INT	Selección de la magnitud reguladora auxiliar valor real B para el regulador 0: Valor real B = 0.0 1 a 4: Valor de entrada analógica 1 a 4	51
INT	Selección de la magnitud reguladora auxiliar valor real C para el regulador 0: Valor real C = 0.0 1 a 4: Valor de entrada analógica 1 a 4	52

Tipo de datos	Descripción	Número de índice
INT	Selección de la magnitud reguladora auxiliar valor real D para el regulador 0: Valor real D = 0.0 1 a 4: Valor de entrada analógica 1 a 4 17 a 20: Magnitud manipulada (LMN) de regulador 1 a 4	53
INT	Selección de la magnitud perturbadora DISV para el regulador 0: Magnitud perturbadora = 0.0 1 a 4: Valor de entrada analógica 1 a 4	54
INT	Selección del seguimiento de posición TRACK_PER para el regulador 0: Seguimiento de posición = 0.0 1 a 4: Valor de entrada analógica 1 a 4	55
INT	Selección del seguimiento de posición LMNR_PER para el regulador 0: Seguimiento de posición = 0.0 1 a 4: Valor de entrada analógica 1 a 4	56
INT	Selección de la señal para la conmutación al valor de seguridad para la variable manipulada del regulador 0: Sólo asignación predeterminada a través del parámetro SAFE_ON de la instrucción PID_FM 1 a 8: Asignación predeterminada con el parámetro SAFE_ON de la instrucción PID_FM predeterminado por la entrada digital 1 a 8	57
INT	Selección de la señal para la conmutación a la función de seguimiento de la variable manipulada del regulador 0: Sólo asignación predeterminada a través del parámetro LMNTRKON de la instrucción PID_FM 1 a 8: Asignación predeterminada con el parámetro LMNTRKON de la instrucción PID_FM predeterminado por la entrada digital 1 a 8	58
INT	Selección de la señal para la conmutación de la variable manipulada del regulador a LMN_RE 0: Sólo asignación predeterminada a través del parámetro LMN_REON de la instrucción PID_FM 1 a 8: Asignación predeterminada con el parámetro LMN_REON de la instrucción PID_FM predeterminado por la entrada digital 1 a 8	59
INT	Selección de la señal de tope superior de la realimentación de posición 0: Sólo asignación predeterminada a través del parámetro LMNRHSRE de la instrucción PID_FM 1 a 8: Asignación predeterminada con el parámetro LMNRHSRE de la instrucción PID_FM predeterminado por la entrada digital 1 a 8	60
INT	Selección de la señal de tope inferior de la realimentación de posición 0: Sólo asignación predeterminada a través del parámetro LMNRLSRE de la instrucción PID_FM 1 a 8: Asignación predeterminada con el parámetro LMNRLSRE de la instrucción PID_FM predeterminado por la entrada digital 1 a 8	61

Consulte también

Parámetros de entrada de PID_PAR (Página 7514)

Parámetros de salida de PID_PAR (Página 7515)

CJ_T_PAR

Descripción de CJ_T_PAR

La instrucción CJ_T_PAR permite cambiar online la temperatura de unión fría parametrizada. Esto es necesario si una regulación de temperatura debe funcionar con varios FM 355 con entradas de termopar sin necesidad de conectar un Pt 100 en cada FM 355.

Llamada

CJ_T_PAR requiere una marcha de inicialización. Para ello debe llamarse una sola vez en el arranque de la CPU con el parámetro COM_RST = TRUE.

Normalmente, CJ_T_PAR se llama cíclicamente. Por motivos relacionados con el tiempo de operación, COM_RST debería tener el valor FALSE. El parámetro COM_RST es un parámetro de entrada que no se desactiva.

CJ_T_PAR debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355.

Si se utiliza el FM 355 en periferia descentralizada, pueden ser necesarios varios ciclos hasta que el parámetro se haya transmitido al FM 355. Hasta que la transmisión no haya finalizado, el parámetro BUSY tiene el valor TRUE. Por ello, al modificar parámetros debe llamarse la instrucción CJ_T_PAR de forma repetida hasta que BUSY = FALSE.

Ejemplo

Si en un regulador de extrusionador con más de cuatro zonas de calentamiento la temperatura de las uniones frías se mide con un FM 355, puede leerse con READ_355 en el parámetro CJ_TEMP y parametrizarse en los FM 355 restantes con CJ_T_PAR.

Consulte también

Parámetros de entrada de CJ_T_PAR (Página 7519)

Parámetros de salida de CJ_T_PAR (Página 7519)

Parámetros de entrada de CJ_T_PAR

Tabla 9-228

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	0.0	BOOL	FALSE	Si el parámetro es COM_RST = TRUE, la instrucción CJ_T_PAR realiza una marcha de inicialización. En ella se leen los parámetros de los datos del sistema de la CPU y se depositan en el DB de instancia.
MOD_ADDR	2.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CJ_T	4.0	REAL	0.0	En el parámetro CJ_T se puede predeterminar la temperatura de las uniones frías. El rango de valores permitido varía en función del tipo de sensor.

Consulte también

Descripción de CJ_T_PAR (Página 7518)

Parámetros de salida de CJ_T_PAR (Página 7519)

Parámetros de salida de CJ_T_PAR

Tabla 9-229

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	8.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
BUSY	10.0	BOOL	FALSE	Si BUSY = TRUE, el módulo aún no ha adoptado los parámetros (con periferia descentralizada). En ese caso, la llamada de la instrucción PID_PAR debería repetirse en el siguiente ciclo.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

DB para HMI

Descripción de DB 101 a 104

Para poder manejar el FM 355 con el OP en caso de fallo de la CPU se requieren los bloques de datos 101 a 104. Con MPI se pueden establecer hasta tres conexiones desde el FM 355 a los OPs.

El manejo del FM 355 con el OP sólo es posible en el estado STOP de la CPU en caso de fallo de la CPU.

El FM 355 siempre se puede observar con el OP.

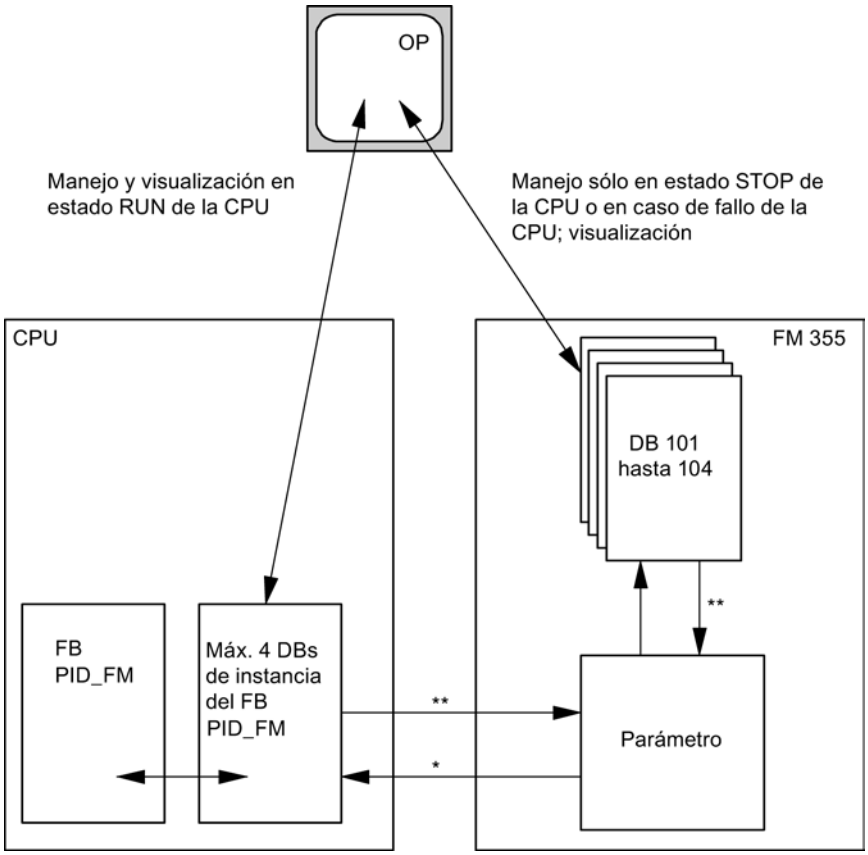
La interfaz de variables del FM 355 contiene cuatro bloques de datos con los números de bloque 101 a 104 para los canales de regulador 1 a 4 (véase la siguiente figura).

Nota

El contenido de los bloques de datos 101 a 104 no refleja automáticamente el valor de parámetro efectivo en el FM 355. Los parámetros cambiados con ayuda del OP sólo se aplican en el FM 355 tras haberse activado los bits de operación LOAD_PAR o LOAD_OP.

Si se cambia un parámetro manejando el OP sin activar el bit de operación correspondiente, el valor de parámetro modificado está en el bloque de datos, pero el FM 355 trabaja internamente con el valor anterior no modificado del parámetro.

Tras activar los bits de operación y aplicar los parámetros en el FM 355, los bits de operación LOAD_PAR o LOAD_OP son desactivados por el FM 355.



* Controlado por el parámetro READ_VAR del DB de instancia

** Controlado por los parámetros LOAD_OP y LOAD_PAR

Consulte también

Descripción de PID_FM (Página 7484)

Parámetros de entrada de los DB 101 a 104

Tabla 9-230

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_HLM	0.0	REAL	100.0	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior consigna" indica el límite superior. SP_HLM > SP_LLM
SP_LLM	4.0	REAL	0.0	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior consigna" indica el límite inferior. SP_LLM < SP_HLM

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
H_ALM	8.0	REAL	100.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más alto. H_ALM > H_WRN
H_WRN	12.0	REAL	90.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite superior" indica el segundo límite más alto. Se admiten valores entre H_ALM y L_WRN
L_WRN	16.0	REAL	10.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite inferior" indica el segundo límite más bajo. Se admiten valores entre H_WRN y L_ALM
L_ALM	20.0	REAL	0.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más bajo. L_ALM < L_WRN
HYS	24.0	REAL	1.0	Para evitar oscilaciones de las indicaciones de vigilancia, puede parametrizarse una histéresis en la entrada "Histéresis". HYS ≥ 0.0
DEADB_W	28.0	REAL	0.0	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. DEADB_W ≥ 0.0
GAIN	32.0	REAL	1.0	La entrada "Ganancia proporcional" indica la ganancia del regulador.
TI	36.0	REAL	3000.0	La entrada "Tiempo de integración" determina el comportamiento temporal del integrador. Con TI = 0 el integrador está desactivado. TI = 0.0 o bien TI ≥ 0.5
TD	40.0	REAL	0.0	La entrada "Tiempo de acción derivada" determina el comportamiento temporal del diferenciador. Con TD = 0 el diferenciador está desactivado. TD = 0.0 o bien TD ≥ 1.0
TM_LAG	44.0	REAL	5.0	El algoritmo de la acción D contiene un retardo que se puede parametrizar en la entrada "Tiempo de retardo de la acción D". TM_LAG ≥ 0.5
LMN_SAFE	48.0	REAL	0.0	Para la variable manipulada puede parametrizarse un valor de seguridad en la entrada "Variable manipulada de seguridad". Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_HLM	52.0	REAL	100.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior de la variable manipulada" indica el límite superior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre LMN_LLM y 100.0 (%).
LMN_LLM	56.0	REAL	0.0	La variable manipulada está limitado siempre por un límite superior y uno inferior. La entrada "Límite inferior de la variable manipulada" indica el límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre -100.0 y LMN_HLM (%).

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MTR_TM	60.0	REAL	60.0	En el parámetro "Tiempo de posicionamiento del motor" se introduce el tiempo de operación de la servoválvula de un tope al otro. (Sólo en regulador paso a paso) MTR_TM ≥ 0.001
PULSE_TM	64.0	REAL	0.2	En el parámetro "Duración mínima de impulso" se puede parametrizar una duración de impulso mínima. (Sólo en regulador paso a paso o de impulsos) PULSE_TM ≥ 0.0
BREAK_TM	68.0	REAL	0.2	En el parámetro "Duración mínima de pausa" se puede parametrizar una duración de pausa mínima. (Sólo en regulador paso a paso o de impulsos) BREAK_TM ≥ 0.0
SP_RE	72.0	REAL	0.0	En la entrada "Consigna externa" se conecta una consigna externa con el regulador.
LMN_RE	76.0	REAL	0.0	En la entrada "Variable manipulada externo" se conecta una variable manipulada externa con el regulador.
LMNRSVAL	80.0	REAL	0.0	Desde el cuadro de diálogo Puesta en marcha se puede acceder a la entrada "Valor inicial de la realimentación de posición simulada". En ese parámetro se indica el valor inicial de la simulación. (Sólo en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores comprendidos entre -100.0 y 100.0 (%).
SAFE_ON	84.0	BOOL	FALSE	Si la entrada "Adoptar posición de seguridad" está activada, se aplica un valor de seguridad como variable manipulada. Nota: El procesamiento de la señal de salida con LMNDN_OP, LMNUP_OP y LMNSOPON en el regulador paso a paso tiene mayor prioridad que la variable manipulada de seguridad.
LMNTRKON	84.1	BOOL	FALSE	Si la entrada "Corregir (LMN a través de EA)" está activada, la variable manipulada se rige por una entrada analógica (EA). (No en el regulador paso a paso sin realimentación de posición analógica)
LMN_REON	84.2	BOOL	FALSE	Si la entrada "Aplicar variable manipulada externa" está activada, se aplica la variable manipulada externo LMN_RE como variable manipulada.
LMNRHSRE	84.3	BOOL	FALSE	La señal "Servoválvula en el tope superior" se interconecta a la entrada "Señal de tope superior de la realimentación de posición". LMNRHSRE = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
LMNRLSRE	84.4	BOOL	FALSE	La señal "Servoválvula en el tope inferior" se interconecta a la entrada "Señal de tope inferior de la realimentación de posición". LMNRLSRE = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
LMNSOPON	84.5	BOOL	FALSE	Si el bit de la entrada "Activar control manual de señal de valor manipulada" está activado, las señales LMNUP_OP y LMNDN_OP se aplican como señales de variable manipulada. (Sólo en regulador paso a paso)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMNUP_OP	84.6	BOOL	FALSE	Si LMNSOPON está activado, el valor se adopta como señal de la variable manipulada en la entrada "Control manual señal de variable manipulada alta". (Sólo en el regulador paso a paso)
LMNDN_OP	84.7	BOOL	FALSE	Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Bajar señal de variable manipulada". (Sólo en regulador paso a paso)
MONERSEL	85.0	BOOL	FALSE	El regulador dispone de un avisador de valor límite, que puede utilizarse para el valor real o para el error de regulación. Si la entrada "Vigilancia: Valor real = 0, Error de regulación = 1" está activada, se vigila el error de regulación.
LMNRS_ON	85.1	BOOL	FALSE	Si no se dispone de una realimentación de posición, puede simularse. La función se activa en la entrada "Activar simulación de la realimentación de posición". ATENCIÓN: Con el paso del tiempo, la simulación difiere cada vez más de la realimentación de posición real. (Sólo en el regulador paso a paso sin realimentación de posición analógica)
FUZID_ON	85.2	BOOL	FALSE	En la entrada "Activar identificación de difuso" se activa la identificación del algoritmo difuso.
SPINT_EN	85.3	BOOL	FALSE	La entrada "Entrada de control manual: externo = 0, interno = 1" determina la entrada que se transfiere como consigna al módulo. SPINT_EN = TRUE: se transfiere SP_INT. SPINT_EN = FALSE: se transfiere SP_RE.
P_SEL	85.4	BOOL	TRUE	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción proporcional está conectada si la entrada "Activar acción P" está activada.
PFDB_SEL	85.5	BOOL	FALSE	En el algoritmo PID las acciones P y D se pueden incluir en el lazo de realimentación. La acción proporcional se encuentra en el lazo de realimentación si la entrada "Incluir acción P en lazo de realimentación" está activada.
D_EL_SEL	86.0	INT	0	El componente D del algoritmo PID puede aplicarse en una entrada separada. Ésta se selecciona en la entrada "Entrada para componente D". 0: Error de regulación 1 a 4: Entrada analógica 1 a 4 17: Valor real negativo

Consulte también

Descripción de PID_FM (Página 7484)

Parámetros de salida de los DB 101 a 104

Tabla 9-231

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP	94.0	REAL	0.0	En la salida "Consigna" se indica la consigna que actúa efectivamente.
PV	98.0	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	102.0	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.
DISV	106.0	REAL	0.0	En la salida "Variable perturbadora" se indica la magnitud perturbadora que actúa efectivamente. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN	110.0	REAL	0.0	En la salida "Variable manipulada" se indica la variable manipulada que actúa efectivamente. En el regulador paso a paso sin realimentación de posición analógica, en el parámetro LMN se indica la acción P + D ilimitada. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_A	114.0	REAL	0.0	En la salida "Variable manipulada A de la función de rango partido / realimentación de posición" se indica la variable manipulada A de la función de rango partido en el regulador continuo, y la realimentación de posición en el regulador paso a paso con realimentación de posición analógica. En el regulador paso a paso sin realimentación de posición analógica se muestra la realimentación de posición simulada. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_B	118.0	REAL	0.0	Para el regulador continuo, en la salida "Variable manipulada B de la función de rango partido" se indica la variable manipulada B de la función de rango partido. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
QH_ALM	122.0	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite superior disparada" se notifica el rebase del límite H_ALM.
QH_WRN	122.1	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite superior disparada" se notifica el rebase del límite H_WRN.
QL_WRN	122.2	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite inferior disparada" se notifica el rebase del límite L_WRN.
QL_ALM	122.3	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite inferior disparada" se notifica el rebase del límite L_ALM.
QLMN_HLM	122.4	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de variable manipulada alcanzado" se notifica el rebase del límite superior. (No en el regulador paso a paso sin realimentación de posición analógica)
QLMN_LLM	122.5	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la variable manipulada alcanzado" indica el rebase por defecto del límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QSPINTON	122.6	BOOL	FALSE	La salida "Consigna interna activada" indica que SP_INT se ha transferido al módulo.
QPARA_F	123.0	BOOL	FALSE	El módulo comprueba la validez de los parámetros. Un error de parametrización se indica en la salida "Error de parametrización". Este error de parametrización también se puede ver en el cuadro de diálogo "Online y diagnóstico" del módulo.
QCH_F	123.1	BOOL	FALSE	La salida "Error de canal" se activa cuando el canal de regulación no puede arrojar resultados válidos. El error de canal (p. ej. rotura de hilo) también se activa con QPARA_F = 1 o QMOD_F = 1. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.
QUPRLM	123.2	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación positiva de rampa de subida de la consigna" está activada, se limita la rampa de subida de la consigna.
QDNRLM	123.3	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación negativa de rampa de subida de la consigna" está activada, se limita la rampa de bajada de la consigna.
QSP_HLM	123.4	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de consigna alcanzado" se notifica el rebase del límite superior.
QSP_LLM	123.5	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la consigna alcanzado" indica el rebase del límite inferior.
QSPOPON	123.6	BOOL	FALSE	En la salida "Control manual de consigna activado" se indica si la consigna se varía a través del cuadro de diálogo "Puesta en marcha". Si el bit está activado, el valor SP_OP se aplica como consigna.
QLMNSAFE	123.7	BOOL	FALSE	Si la salida "Modo de seguridad" está activada, como variable manipulada se emite la variable manipulada de seguridad.
QLMNOPON	124.0	BOOL	FALSE	En la salida "Control manual de variable manipulada activado" se indica si la variable manipulada se controla desde el cuadro de diálogo "Puesta en servicio". Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
QLMNTRK	124.1	BOOL	FALSE	La salida "Modo de seguimiento" indica si la variable manipulada se corrige con una entrada analógica.
QLMN_RE	124.2	BOOL	FALSE	La salida "Manual = 1; Automático = 0" indica si la variable manipulada adopta la variable manipulada externo LMN_RE (Manual = 1) o no.
QLMNR_HS	124.3	BOOL	FALSE	La salida "Señal de tope superior de la realimentación de posición" indica si la servoválvula se encuentra en el tope superior. QLMNR_HS = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
QLMNR_LS	124.4	BOOL	FALSE	La salida "Señal de tope inferior de la realimentación de posición" indica si la servoválvula se encuentra en el tope inferior. QLMNR_LS = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
QLMNR_ON	124.5	BOOL	FALSE	La salida "Realimentación de posición activada" indica que está activado el modo de operación "Regulador paso a paso con realimentación de posición" o "Regulador paso a paso sin realimentación de posición".
QFUZZY	124.6	BOOL	FALSE	Si la salida "Algoritmo PID = 0; Difuso = 1" está activada, el regulador trabaja con el algoritmo difuso.
QSPLEPV	124.7	BOOL	FALSE	La salida "Indicación del regulador difuso: consigna < valor real" se activa con el regulador difuso si la consigna es menor que el valor real efectivo.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QSPR	125.0	BOOL	FALSE	Si la salida "Modo de rango partido" está activada, el regulador continuo trabaja en modo de rango partido.
QLMNUP	125.1	BOOL	FALSE	Se trata de la salida "Señal alta de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QLMNDN	125.2	BOOL	FALSE	Se trata de la salida "Señal baja de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QBACKUP	125.4	BOOL	FALSE	0= no hay estado de copia de seguridad (CPU en RUN) 1= estado de copia de seguridad (CPU e STOP o falla)
QID	125.5	BOOL	FALSE	QID = TRUE indica que hay una identificación en curso (no que esté activada). Una vez completada la identificación, el resultado puede verse a través del parámetro IDSTATUS de la instrucción CH_DIAG.
QMAN_FC	125.6	BOOL	FALSE	El regulador es un regulador piloto que, mediante la operación manual de un regulador secundario, se corrige según el valor real de este último, o cuya acción I se detiene porque la consigna o la magnitud manipulada del regulador secundario están en el límite.
RET_VALU	126.0	WORD	0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Descripción de PID_FM (Página 7484)

Parámetros de entrada/salida de DB 101 a 104

Tabla 9-232

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_INT	128.0	REAL	0.0	El parámetro de entrada/salida "Consigna interna" sirve para preseleccionar una consigna mediante funciones de manejo y visualización.
SP_OP	132.0	REAL	0.0	El cuadro de diálogo "Puesta en marcha" tiene acceso al parámetro de entrada/salida "Control manual de consigna" Si el bit SP_OP_ON está activado, se aplica el valor "Control manual de consigna" como consigna.
LMN_OP	136.0	REAL	0.0	El cuadro de diálogo "Puesta en servicio" tiene acceso al parámetro de entrada/salida "Control manual de variable manipulada" Si el bit LMNOP_ON está activado, el valor "Control manual de variable manipulada" se aplica como variable manipulada. Se admiten valores comprendidos entre -100.0 y 100.0 (%).

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_OP_ON	140.0	BOOL	FALSE	El cuadro de diálogo "Puesta en marcha" tiene acceso al parámetro de entrada/salida "Activar control manual de consigna". Si el bit está activado, el valor SP_OP se aplica como consigna.
LMNOP_ON	140.1	BOOL	FALSE	El cuadro de diálogo "Puesta en marcha" tiene acceso al parámetro de entrada/salida "Activar control manual de variable manipulada" Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
LOAD_PAR	140.2	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros de regulación en el FM 355" está activado, los parámetros de regulación se cargan en el módulo y el parámetro de entrada/salida se desactiva.
LOAD_OP	140.3	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros operativos en el FM 355" está activado, los parámetros operativos se cargan en el módulo y el parámetro de entrada/salida se desactiva.

Consulte también

Descripción de DB 101 a 104 (Página 7520)

Descripción de PID_FM (Página 7484)

FM 355-2 Temperature Control

FMT_PID

Descripción de FMT_PID

La instrucción FMT_PID permite realizar las siguientes acciones durante el funcionamiento:

- Transmitir parámetros de operación al FM 355-2
- Transmitir parámetros de regulación al FM 355-2
- Leer valores del proceso del FM 355-2

Los parámetros de operación son todas las variables estáticas del DB de instancia en la estructura OP.

Los parámetros del regulador son todas las variables estáticas del DB de instancia en la estructura PAR.

Los parámetros del proceso son todas las variables estáticas del DB de instancia en la estructura OUT.

Los datos necesarios para el FMT_PID se depositan en un DB de instancia en la CPU. El FMT_PID lee datos del FM 355-2 controlado por el programa y los escribe, controlado por el programa, en el FM 355-2.

Llamada

La instrucción FMT_PID debe llamarse en el mismo OB así como todas las instrucciones restantes que acceden al mismo FM 355-2.

FMT_PID se llama en el OB de alarma cíclica. Requiere una marcha de inicialización, que se inicia activando el parámetro COM_RST = TRUE en el arranque de la CPU. Tras la marcha de inicialización, FMT_PID pone el parámetro COM_RST en FALSE.

Si se efectúa una llamada a la instrucción FMT_PID como DB multiinstancia, no se dispone de interfaz de puesta en servicio. En este caso, los FM 355-2 se deben poner en servicio a través de una tabla de observación, o bien con la instrucción FMT_TUN.

Llamada en periferia descentralizada

En una configuración descentralizada y en caso de llamada simultánea de FMT_PID y FMT_TUN, deben considerarse los siguientes aspectos:

- LOAD_OP no puede activarse simultáneamente en FMT_PID y FMT_TUN.
- READ_OUT no puede activarse simultáneamente en FMT_PID y FMT_TUN.

Motivo: Ambos FBs acceden al FM 355-2 a través de los mismos registros.

Por ello hay que garantizar que en el mismo momento sólo uno de los dos FBs lea o escriba un registro.

Arranque

En el arranque y en la transición de STOP a RUN de la CPU, los parámetros del FM 355-2 se sobrescriben con los parámetros del SDB de la CPU.

Respuesta en caso de error

El parámetro de salida RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC. RET_VALU puede evaluarse si los parámetros READ_PAR y LOAD_PAR no se desactivan.

En la llamada de la instrucción FMT_PID puede producirse un error de acceso a la periferia si el FM 355-2 no está insertado o si no recibe suministro de tensión. En ese caso, la CPU pasa a STOP si no hay ningún OB 122 cargado en la CPU.

Si se produce un error en la lectura/escritura del registro, QMOD_F y QCH_F se activan. En caso de error de parametrización, se activan QPAR_F y QCH_F.

Consulte también

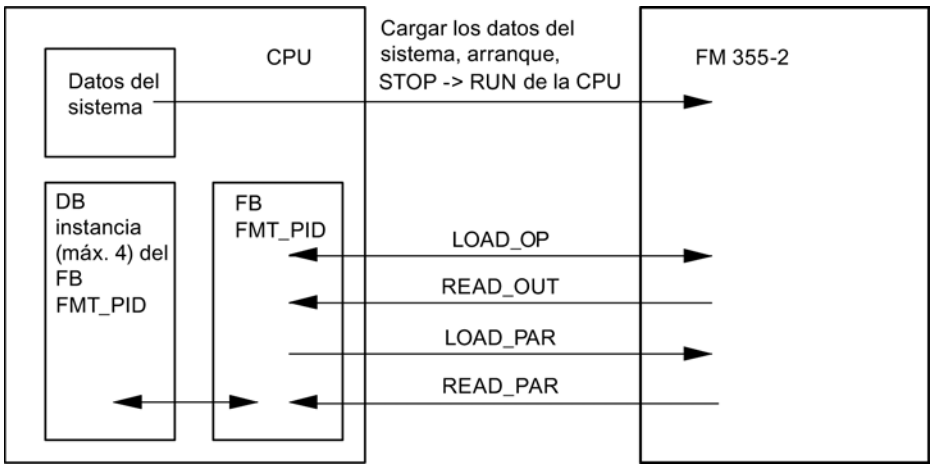
Diagramas de bloques (Página 7540)

Funcionamiento de FMT_PID

Influencia de los parámetros LOAD_OP, READ_OUT, LOAD_PAR y READ_PAR

En la marcha de inicialización, FMT_PID lee los parámetros de los datos del sistema y los deposita en el DB de instancia. El DB de instancia y los datos del sistema están sincronizados.

La figura siguiente muestra cómo afectan los parámetros LOAD_OP, READ_OUT, LOAD_PAR y READ_PAR al funcionamiento de la instrucción.



La tabla siguiente contiene la información detallada en relación con los valores de los parámetros.

Parámetro	TRUE	FALSE
LOAD_OP	Las variables estáticas de la estructura OP se transmiten cíclicamente con las instrucciones WRREC y RDREC al FM 355-2. Los parámetros de salida se leen del FM 355-2.	Las variables estáticas de la estructura OP se transmiten a través de áreas de entradas y salidas del módulo. No es posible si dos instancias de la instrucción FMT_PID acceden al mismo número de canal de un módulo.
	READ_VAR adopta el estado TRUE	READ_VAR no cambia
	En las siguientes situaciones, LOAD_OP adopta automáticamente el estado TRUE: • Si ha activado uno de los siguientes parámetros: TUN_ST, TUN_CST, SAVE_PAR,UNDO_PAR y LOAD_PID • Si modifica TUN_ON • Si modifica la consigna SP_RE.	
	Tiempo de operación largo	Tiempo de operación reducido
	Los valores son efectivos tras un ciclo de la CPU o el FM. Rige el ciclo más largo.	En la configuración centralizada ya se requieren de 3 a 4 ciclos de la CPU o del FM para que los valores sean efectivos. Rige el ciclo más largo

Parámetro	TRUE	FALSE
	Se transmiten todos los parámetros de operación	Sólo se transmiten los parámetros de operación hasta LMN_DN incluido. En modo automático sólo se transmite la consigna SP_RE. En modo manual sólo se transmite la variable manipulada en manual LMN_RE.
READ_OUT	Las variables estáticas de la estructura OUT se transmiten cíclicamente con las instrucciones WRREC y RDREC al FM 355-2.	Las variables estáticas de la estructura OUT se transmiten a través de áreas de entradas y salidas del módulo y en la estructura OUT.
	En las siguientes situaciones, READ_OUT adopta automáticamente el estado TRUE: • Durante una optimización QTUN_ON = TRUE • Si modifica la consigna SP_RE	No es posible si dos instancias de la instrucción FMT_PID acceden al mismo número de canal de un módulo.
	Tiempo de ejecución prolongado	Tiempo de ejecución reducido
	Todos los parámetros se leen del FM 355	Los siguientes parámetros **no** se leen del FM 355 • SP (consigna del FM) • ER (error de regulación) • DISV (magnitud perturbadora) • LMN_A • LMN_B • PHASE • STATUS_H • STATUS_C • STATUS_D • ZONE_TUN
READ_PAR	Todas las variables estáticas de la estructura PAR son leídas por el FM 355-2 y se guardan en el DB de instancia en la CPU.	Sin transmisión de los parámetros del regulador a la CPU En el programa de usuario se pueden modificar parámetros individuales en la CPU.
LOAD_PAR	Todas las variables estáticas de la estructura PAR se cargan desde la CPU al FM 355-2. Esto es necesario si se han cambiado los parámetros de regulador en la CPU en función de los estados del proceso.	Sin transmisión de los parámetros del regulador al FM 355-2

Una vez completada la transmisión de datos, los parámetros LOAD_OP, READ_OUT, LOAD_PAR y READ_PAR se vuelven a poner a FALSE.

Uso en periferia descentralizada

Si se utiliza el FM 355-2 en la periferia descentralizada, la transmisión de datos y la desactivación de los parámetros puede requerir varios ciclos de llamada.

Tiempos de ejecución de FMT_PID

En la reparametrización controlada por programa (LOAD_PAR, LOAD_OP) del FM 355-2 con FMT_PID aumenta el tiempo de ejecución. Los nuevos parámetros siempre entran en vigor inmediatamente.

Condiciones marco			Tiempo de ejecución en	
READ_OUT	LOAD_OP	LOAD_PAR READ_PAR	CPU 315-2-DP	CPU 416-2 DP
FALSE	FALSE	FALSE	0,65 ms	0.04 ms
TRUE	FALSE	FALSE	2.85 ms	0.30 ms
TRUE	TRUE	FALSE	4.56 ms	0.55 ms
FALSE	FALSE	TRUE	3.58 ms*)	0.30 ms
TRUE	FALSE	TRUE	5.8 ms*)	0.56 ms
TRUE	TRUE	TRUE	7.41 ms*)	0.82 ms
*) Si se activa READ_PAR en lugar de LOAD_PAR, aumenta el tiempo de ejecución en la CPU 315-2 DP en 0.35 ms.				

Consulte también

Diagramas de bloques (Página 7540)

Parámetros de entrada de FMT_PID

Tabla 9-233

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	0	Número de canal de regulador al que se refiere el DB de instancia. Son válidos los valores entre 0 y 3

Consulte también

Diagramas de bloques (Página 7540)

Parámetros de salida de FMT_PID

Tabla 9-234

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QMOD_F	4.0	BOOL	FALSE	La instrucción comprueba si los procesos de escritura y lectura de un registro han finalizado correctamente. Si se detectan errores, se activa la salida QMOD_F. La causa puede ser: Una dirección de módulo incorrecta en el parámetro MOD_ADDR, un número de canal incorrecto en el parámetro CHANNEL o un módulo defectuoso.
RET_VALU	6.0	WORD	W#16#0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetros de entrada/salida de FMT_PID

Tabla 9-235

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	8.0	BOOL	FALSE	Con COM_RST = TRUE la instrucción FMT_PID ejecuta una marcha de inicialización y desactiva COM_RST. Esta marcha de inicialización es imprescindible en cada arranque de la CPU.
LOAD_OP	8.1	BOOL	FALSE	Si el parámetro de entrada/salida LOAD_OP está activado, los parámetros de operación se cargan en el módulo, los parámetros de salida se leen y el parámetro de entrada/salida se desactiva.
READ_OUT	8.2	BOOL	FALSE	Si el parámetro de entrada/salida READ_OUT está activado, se leen los parámetros de salida del módulo, se depositan en la estructura OUT del DB de instancia y el parámetro de entrada/salida se desactiva.
LOAD_PAR	8.3	BOOL	FALSE	Si el parámetro de entrada/salida LOAD_PAR está activado, los parámetros del regulador se cargan en el módulo y el parámetro de entrada/salida se desactiva.
READ_PAR	8.4	BOOL	FALSE	Si el parámetro es READ_PAR = TRUE, los parámetros del regulador se leen del FM 355-2 y se depositan en la estructura PAR del DB de instancia. A continuación se desactiva el parámetro de entrada/salida.

Consulte también

Diagramas de bloques (Página 7540)

Variables estáticas de FMT_PID.OP

Tabla 9-236

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
vers_nr	10.0	WORD	W#16#3230	El parámetro vers_nr no puede ser modificado por el usuario. Identifica el inicio de los parámetros de operación, que se transfieren al módulo si se activa LOAD_OP = TRUE.
SP_RE	12.0	REAL	0.0	En la entrada SP_RE se conecta una consigna externa con el regulador.
LMN_RE	16.0	REAL	0.0	En la entrada LMN_RE se conecta una variable manipulada externa con el regulador. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
SAFE_ON	20.0	BOOL	FALSE	Si SAFE_ON está activada, se aplica un valor de seguridad como variable manipulada.
LMNTRKON	20.1	BOOL	FALSE	Si LMNTRKON está activada, la variable manipulada se corrige mediante una entrada analógica (EA). (No en el regulador paso a paso sin realimentación de posición analógica)
LMN_REON	20.2	BOOL	FALSE	Si LMN_REON está activada, se aplica una variable manipulada externa LMN_RE como variable manipulada (modo manual).
LMNRHSRE	20.3	BOOL	FALSE	La señal "Servoválvula en tope superior" puede aplicarse en una entrada digital de FM 355-2 o en la entrada LMNRHSRE. LMNRHSRE = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en el regulador paso a paso).
LMNRLSRE	20.4	BOOL	FALSE	La señal "Servoválvula en tope inferior" puede aplicarse en una entrada digital del FM 355-2 o en la entrada LMNRLSRE. LMNRLSRE = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en el regulador paso a paso).
LMNS_ON	20.5	BOOL	FALSE	Activar control manual de variable manipulada (Sólo en el regulador paso a paso; modo manual).
LMN_UP	20.6	BOOL	FALSE	Subir señal de variable manipulada (Sólo en el regulador paso a paso).
LMN_DN	20.7	BOOL	FALSE	Bajar señal de variable manipulada (Sólo en el regulador paso a paso).
SAVE_PAR	22.0	BOOL	FALSE	Guarda los parámetros de regulador actuales en los parámetros de salida de la instrucción FMT_TUN. • FMT_TUN.SAV_PFAC=PFAC_SP • FMT_TUN.SAV_GAIN =GAIN • FMT_TUN.SAV_TI =TI • FMT_TUN.SAV_TD = TD • FMT_TUN.SAV_D_F =D_F • FMT_TUN.SAV_CONZ = CON_ZONE • FMT_TUN.SAV_RATI =RATIOFAC • FMT_TUN.SAV_CZON=CONZ_ON • FMT_TUN.SAV_PSEL=P_SEL

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
UNDO_PAR	22.1	BOOL	FALSE	Los parámetros de regulador guardados se recuperan, también en el modo automático. • PFAC_SP= FMT_TUN.SAV_PFAC • GAIN = FMT_TUN.SAV_GAIN • TI = FMT_TUN.SAV_TI • TD = FMT_TUN.SAV_TD • D_F = FMT_TUN.SAV_D_F • CON_ZONE = FMT_TUN.SAV_CONZ • RATIOFAC = FMT_TUN.SAV_RATI • CONZ_ON= FMT_TUN.SAV_CZON • P_SEL= FMT_TUN.SAV_PSEL
LOAD_PID	22.2	BOOL	FALSE	Carga con PID_ON=TRUE los parámetros PID (PID_GAIN, PID_TI, PID_TD). Con PID_ON=FALSE se cargan los parámetros PI (PI_GAIN, PI_TI). (También en modo automático)
TUN_ON	22.3	BOOL	FALSE	TUN_ON=TRUE cambia el FM 355-2 a la disponibilidad para optimización (PHASE=1): Se calcula el valor medio de la variable manipulada hasta que se aplica la excitación de la variable manipulada.
TUN_ST	22.4	BOOL	FALSE	Si la consigna debe permanecer constante durante la optimización del regulador en el punto de operación, se aplica un escalón de variable manipulada de TUN_DLMN con TUN_ST=TRUE (PHASE = 1 o PHASE = 2).
TUN_CST	22.5	BOOL	FALSE	Inicia la optimización de refrigeración aplicando un salto de la variable manipulada de TUN_CLMN (PHASE = 1 o PHASE = 2)

Consulte también

Descripción de FMT_PID (Página 7528)

Diagramas de bloques (Página 7540)

Variables estáticas FMT_PID.PAR

Tabla 9-237

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
vers_nr	24.0	WORD	W#16#3230	El parámetro vers_nr no puede ser modificado por el usuario. Identifica el inicio de los parámetros de regulador que se leen del FM con READ_PAR = TRUE y que se depositan en el DB de instancia o bien se transmiten al FM con LOAD_PAR = TRUE.
P_SEL	26.0	BOOL	TRUE	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción proporcional está conectada si la entrada P_SEL está activada.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MONERSEL	26.1	BOOL	FALSE	El regulador dispone de un avisador de valor límite, que puede utilizarse para el valor real o para el error de regulación. Si MONERSEL está activado, se vigila el error de regulación.
PID_ON	26.2	BOOL	TRUE	Tras la optimización o con LOAD_PID =TRUE se activan los parámetros PID. Con PID_ON =FALSE se cargan los parámetros PI. Puede ocurrir que en algunos tipos de proceso sólo se diseñe un regulador PI a pesar de PID_ON = TRUE.
CONZ_ON	26.3	BOOL	FALSE	TRUE = zona de regulación activada FALSE = zona de regulación desactivada
D_EL_SEL	28.0	INT	16	El componente D del algoritmo PID puede depositarse en una entrada separada. Se selecciona en el parámetro D_EL_SEL. • D_EL_SEL = 0 hasta D_EL_SEL = 3: Entrada analógica 0 a 3 • D_EL_SEL = 16: error de regulación • D_EL_SEL = 17: valor real negativo
SP_HLM	30.0	REAL	100.0	La consigna está limitada siempre por un límite superior y uno inferior. El parámetro SP_HLM indica el límite superior. SP_HLM > SP_LLM
SP_LLM	34.0	REAL	0.0	La consigna está limitada siempre por un límite superior y uno inferior. El parámetro SP_LLM indica el límite inferior. SP_LLM < SP_HLM
H_ALM	38.0	REAL	100.0	Límite más alto para la vigilancia del valor real o del error de regulación. H_ALM > H_WRN
H_WRN	42.0	REAL	90.0	Segundo límite más alto para la vigilancia del valor real o del error de regulación. H_WRN > H_ALM
L_WRN	46.0	REAL	10.0	Segundo límite más bajo para la vigilancia del valor real o del error de regulación. Se admiten valores entre H_WRN y L_ALM
L_ALM	50.0	REAL	0.0	Límite más bajo para la vigilancia del valor real o del error de regulación. Se admiten valores entre L_ALM y L_WRN.
HYS	54.0	REAL	1.0	Para evitar oscilaciones de los indicadores de vigilancia, puede parametrizarse una histéresis. HYS >= 0.0
DEADB_W	58.0	REAL	0.0	El error de regulación se conduce por una zona muerta. El parámetro DEADB_W determina el tamaño de la zona muerta. DEADB_W >= 0.0
PFAC_SP	62.0	REAL	1.0	PFAC_SP indica la acción P efectiva en caso de modificación de la consigna. Se admiten valores comprendidos entre 0.0 y 1.0.
GAIN	66.0	REAL	1.0	El parámetro GAIN indica la ganancia del regulador. El sentido de actuación del regulador se invierte mediante el signo negativo de GAIN.
TI	70.0	REAL	3000.0	El parámetro TI (tiempo de acción integral) determina el comportamiento temporal del integrador. Con TI = 0.0 el integrador está desactivado. TI = 0.0 o bien TI >= 0.5

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
TD	74.0	REAL	0.0	El parámetro TD (tiempo de anticipación) determina el comportamiento temporal del diferenciador (componente D). Con TD = 0.0 el diferenciador está desactivado. TD = 0.0 o bien TD >= 1.0
D_F	78.0	REAL	5.0	La constante de tiempo TD/D_F efectiva para el componente D se limita internamente a ≥ tiempo de muestreo/2. Se admiten valores comprendidos entre 5.0 y 10.0.
LMN_SAFE	82.0	REAL	0.0	Para la variable manipulada puede parametrizarse un valor de seguridad. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_HLM	86.0	REAL	100.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. El parámetro LMN_HLM indica el límite superior. (No en el regulador paso a paso sin realimentación de posición analógica). Se admiten valores entre LMN_LLM y 100.0 (%).
LMN_LLM	90.0	REAL	0.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. El parámetro LMN_LLM indica el límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica). Se admiten valores entre -100.0 y LMN_HLM (%).
MTR_TM	94.0	REAL	60.0	Tiempo de operación de la servoválvula de tope a tope (Sólo en el regulador paso a paso). MTR_TM >= 0.001
PULSE_TM	98.0	REAL	0.0	Duración de impulso mínima (Sólo en el regulador paso a paso o de impulsos)
BREAK_TM	102.0	REAL	0.0	Duración mínima de pausa de impulso (Sólo en el regulador paso a paso o de impulsos).
RATIOFAC	106.0	REAL	0.0	Relación entre ganancia de calentamiento/ganancia de refrigeración del proceso. RATIOFAC <> 0.0 provoca lo siguiente: • LMN = LMN * RATIOFAC para LMN<0.0; • CON_ZONE 50% mayor Se admiten los valores de 0.01 a 100.0 ó 0.0.
CON_ZONE	110.0	REAL	100.0	Ancho de zona de regulación Si ER >= CON_ZONE, entonces LMN = LMN_HLM. Si ER <= -CON_ZONE, entonces LMN = LMN_LLM. Si RATIOFAC <> 0.0 y ER <= -CON_ZONE/RATIOFAC, entonces LMN = LMN_LLM. CON_ZONE > 0.0
TUN_DLMN	114.0	REAL	20.0	La excitación del proceso para la optimización del regulador se realiza mediante un salto de la variable manipulada de TUN_DLMN. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
TUN_CLMN	118.0	REAL	-20.0	Salto de la variable manipulada tras el inicio de la optimización de la refrigeración con TUN_CST. Se admiten valores comprendidos entre -100.0 y 100.0 (%).

Consulte también

Diagramas de bloques (Página 7540)

Variables estáticas de FMT_PID.OUT

Tabla 9-238

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
vers_nr	122.0	WORD	W#16#3230	El parámetro vers_nr no puede ser modificado por el usuario. Identifica el inicio de los parámetros de salida que lee el módulo cuando se activa READ_OUT = TRUE.
PV	124.0	REAL	0.0	Valor real efectivo.
LMN	128.0	REAL	0.0	En la salida LMN se indica la variable manipulada que actúa efectivamente. En el regulador paso a paso sin realimentación de posición analógica, se indica la acción P + D ilimitada en la salida LMN. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
QLMNSAFE	132.1	BOOL	FALSE	Si la salida QLMNSAFE está activada, como variable manipulada se indica la variable manipulada de seguridad.
QLMNTRK	132.3	BOOL	FALSE	La salida QLMNTRK indica si la variable manipulada se corrige mediante una entrada analógica.
QLMN_RE	132.4	BOOL	FALSE	La salida QLMN_RE indica si LMN_REON está activada y si la variable manipulada externa LMN_RE se aplica como variable manipulada.
QLMNR_HS	132.5	BOOL	FALSE	QLMNR_HS = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
QLMNR_LS	132.6	BOOL	FALSE	QLMNR_LS = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en el regulador paso a paso).
QLMNR_ON	132.7	BOOL	FALSE	QLMNR_ON = TRUE significa: Regulador paso a paso con realimentación de posición.
QSTEPCON	133.0	BOOL	FALSE	Regulador paso a paso/regulador de impulsos • QSTEPCON = 0: regulador de impulsos o regulador continuo • QSTEPCON = 1: regulador paso a paso
QSPR	133.1	BOOL	FALSE	Si la salida QSPR está activada, el regulador continuo trabaja en modo de rango partido.
QMAN_FC	133.3	BOOL	FALSE	El regulador es un regulador piloto que, mediante la operación manual de un regulador secundario, se corrige según el valor real de este último, o cuya acción I se detiene porque la consigna o la magnitud manipulada del regulador secundario están en el límite.
QH_ALM	134.0	BOOL	FALSE	Límite más alto H_ALM rebasado por valor real o error de regulación.
QH_WRN	134.1	BOOL	FALSE	Segundo límite más alto H_WRN rebasado por valor real o error de regulación.
QL_WRN	134.2	BOOL	FALSE	Segundo límite más bajo L_WRN rebasado por valor real o error de regulación.
QL_ALM	134.3	BOOL	FALSE	Límite más bajo L_ALM rebasado por valor real o error de regulación.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QLMN_HLM	134.4	BOOL	FALSE	La salida QLMN_HLM notifica que se ha alcanzado el límite superior de la variable manipulada LMN_HLM. (No en el regulador paso a paso sin realimentación de posición analógica).
QLMN_LLM	134.5	BOOL	FALSE	La salida QLMN_LLM notifica que se ha alcanzado el límite inferior de la variable manipulada LMN_LLM. (No en el regulador paso a paso sin realimentación de posición analógica)
QPAR_F	134.6	BOOL	FALSE	El módulo comprueba la validez de los parámetros. Un error de parametrización se indica en la salida QPAR_F.
QCH_F	134.7	BOOL	FALSE	La salida QCH_F se activa cuando el canal de regulación no puede arrojar resultados válidos. El error de canal (p. ej. rotura de hilo) también se activa con QPAR_F = TRUE o QMOD_F = TRUE. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.
QUPRLM	135.0	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida QUPRLM está activada, se limita la rampa de subida de la consigna.
QDNRLM	135.1	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida QDNRLM está activada, se limita la rampa de bajada de la consigna.
QSP_HLM	135.2	BOOL	FALSE	La salida QSP_HLM notifica que se ha alcanzado el límite superior de la consigna SP_HLM.
QSP_LLM	135.3	BOOL	FALSE	La salida QSP_LLM notifica que se ha alcanzado el límite inferior de la consigna SP_LLM.
QLMNUP	135.4	BOOL	FALSE	Salida "Señal alta de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QLMNDN	135.5	BOOL	FALSE	Salida "Señal baja de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QTUN_ON	135.6	BOOL	FALSE	QTUN_ON = TRUE indica que se está produciendo una optimización.
PAR_ACT	135.7	BOOL	FALSE	Actualizar parámetros de regulador
SP	136.0	REAL	0.0	Consigna efectiva.
ER	140.0	REAL	0.0	Error de regulación efectivo (SP- PV)
DISV	144.0	REAL	0.0	Magnitud perturbadora efectiva. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_A	148.0	REAL	0.0	Con el regulador continuo o de impulsos, en la salida LMN_A se muestra la variable manipulada A de la función de rango partido, y con el regulador paso a paso con realimentación de posición analógica se muestra la realimentación de posición. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_B	152.0	REAL	0.0	Con el regulador continuo o de impulsos, en la salida LMN_B se muestra la variable manipulada B de la función de rango partido. Se admiten valores comprendidos entre -100.0 y 100.0 (%).

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PHASE	156.0	INT	0	Muestra el progreso de la optimización del regulador. • PHASE = -1: interrupción de la optimización del regulador p. ej. con optimización paralela de varios canales. El valor está presente durante unos pocos ciclos y ejerce de aviso de interrupción entre la instrucción FMT_PID y el módulo. • PHASE = 0: no hay modo de optimización; modo automático o manual • PHASE = 1: disponibilidad para la optimización; comprobar parámetros, esperar excitación, medir los tiempos de muestreo • PHASE = 2: optimización propiamente dicha: Búsqueda del punto de inversión con variable manipulada constante. • PHASE = 3: cálculo de los parámetros del proceso. Guardar los parámetros del regulador válidos antes de la optimización. • PHASE = 4: diseño del regulador • PHASE = 5: corrección del regulador a la nueva variable manipulada • PHASE = 6: corrección del regulador a la nueva variable manipulada • PHASE = 7: comprobación del tipo de sistema si se ha determinado el tipo II o III (con con optimización de calentamiento).
STATUS_H	158.0	INT	0	Parámetro STATUS_H indica un valor de diagnóstico a través de la búsqueda del punto de inversión en el proceso de calentamiento.
STATUS_C	160.0	INT	0	Parámetro STATUS_C indica un valor de diagnóstico a través de la búsqueda del punto de inversión en el proceso de refrigeración.
STATUS_D	162.0	INT	0	Parámetro STATUS_D indica un valor de diagnóstico a través del diseño del regulador en el proceso de calentamiento.
ZONE_TUN	164.0	WORD	W#16#0	En la representación HEX cada una de las cuatro posiciones corresponde a un canal, en el siguiente orden de izquierda a derecha: Canal 0, 1, 2, 3. ZONE_TUN = 0 significa que el canal utilizado no puede optimizarse conjuntamente con el resto de canales. Un valor <> 0000 muestra - con un 1 en cada caso - los canales que se optimizan conjuntamente.

Consulte también

Diagramas de bloques (Página 7540)

Diagramas de bloques

Sinopsis

En las siguientes figuras se representan los puntos en los que son efectivos los parámetros de la instrucción FMT_PID.

En reguladores de tres componentes, de relación y mixtos, los parámetros actúan en los mismos puntos que en los reguladores de valor fijo o en cascada. Esto también rige para los parámetros que están presentes por igual en reguladores continuos, con salida de impulsos y paso a paso. Como norma general puede decirse que los mismos botones contienen los

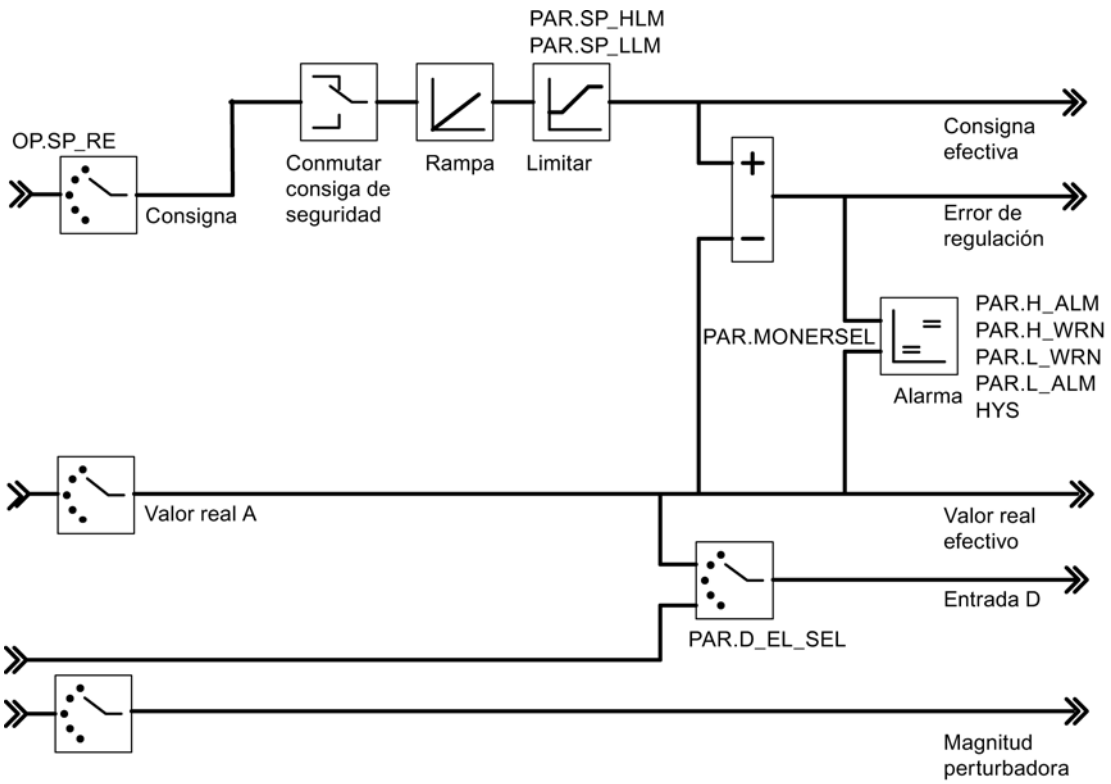
mismos parámetros. Por ello, para una mejor comprensión no se representan todos los diagramas de estructuras, y no todos los parámetros están incluidos en todos los diagramas.

Sin embargo, todos los parámetros de la instrucción FMT_PID están contenidos en los diagramas, excepto MOD_ADDR, CHANNEL, QMOD_F, QPAR_F, QCH_F, QLMNR_ON, RET_VALU, COM_RST, LOAD_PAR, READ_PAR, READ_OUT y LOAD_OP.

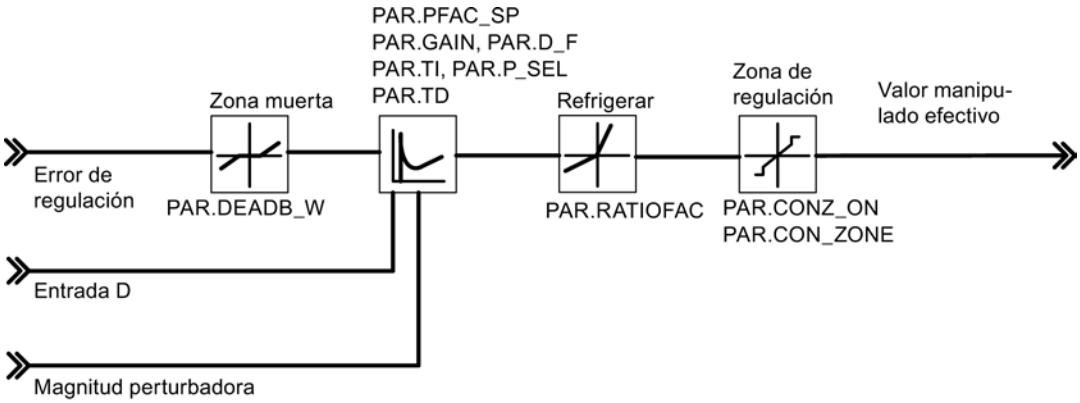
En los siguientes diagramas se representan los puntos en los que son efectivos los parámetros del FMT_PID.

Efecto de los parámetros de entrada

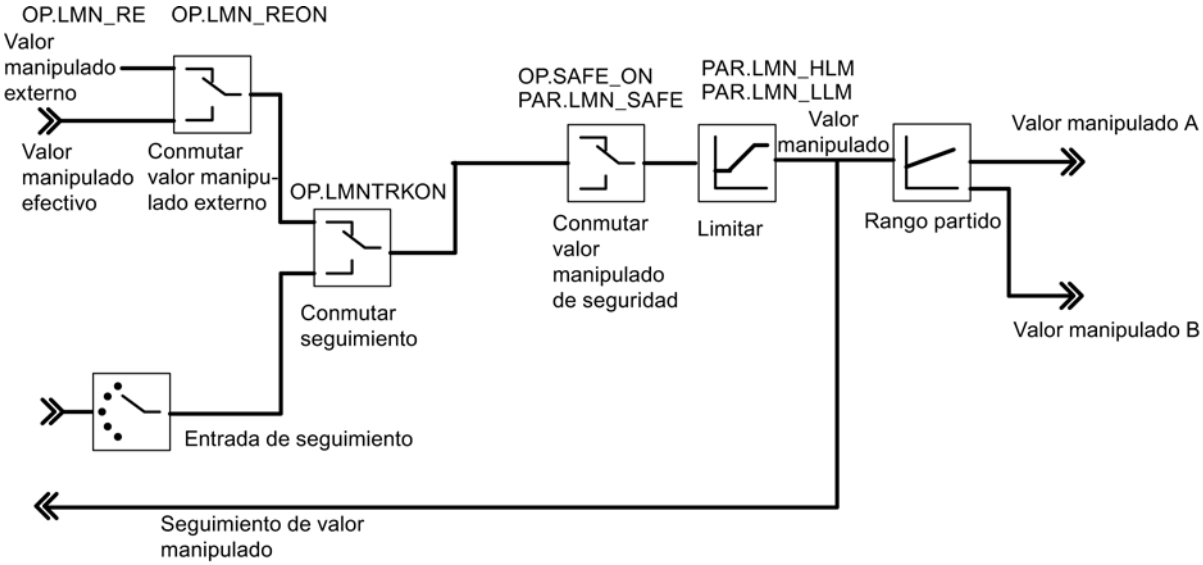
La figura siguiente muestra el cálculo del error de regulación en el regulador de valor fijo o en cascada



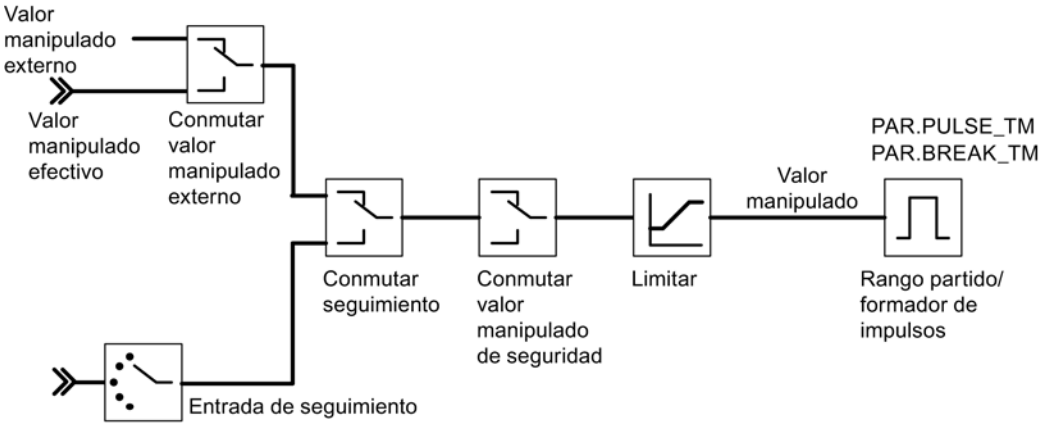
La figura siguiente muestra el diagrama de bloques del algoritmo de regulación



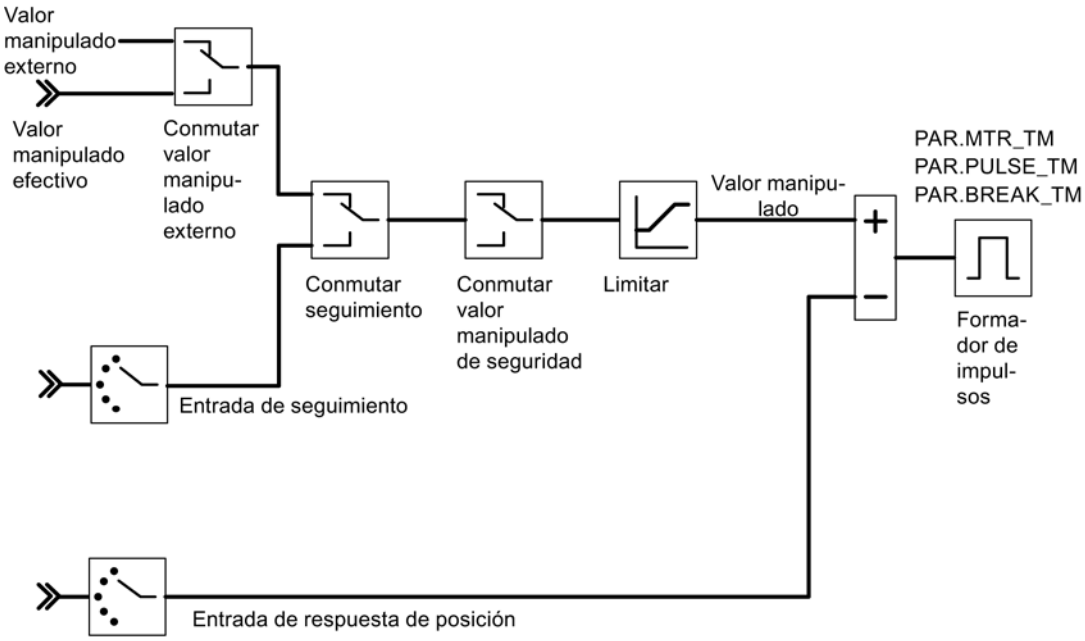
La figura siguiente muestra la salida del regulador continuo (FM 355-2 C)



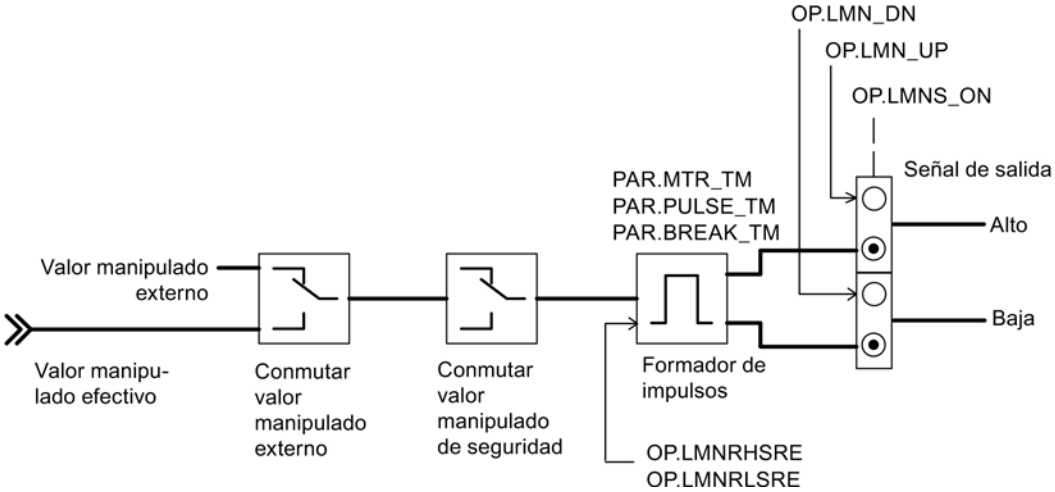
La figura siguiente muestra la salida del regulador de impulsos (FM 355-2 S)



La figura siguiente muestra la salida del regulador paso a paso con realimentación de posición (FM 355-2 S)



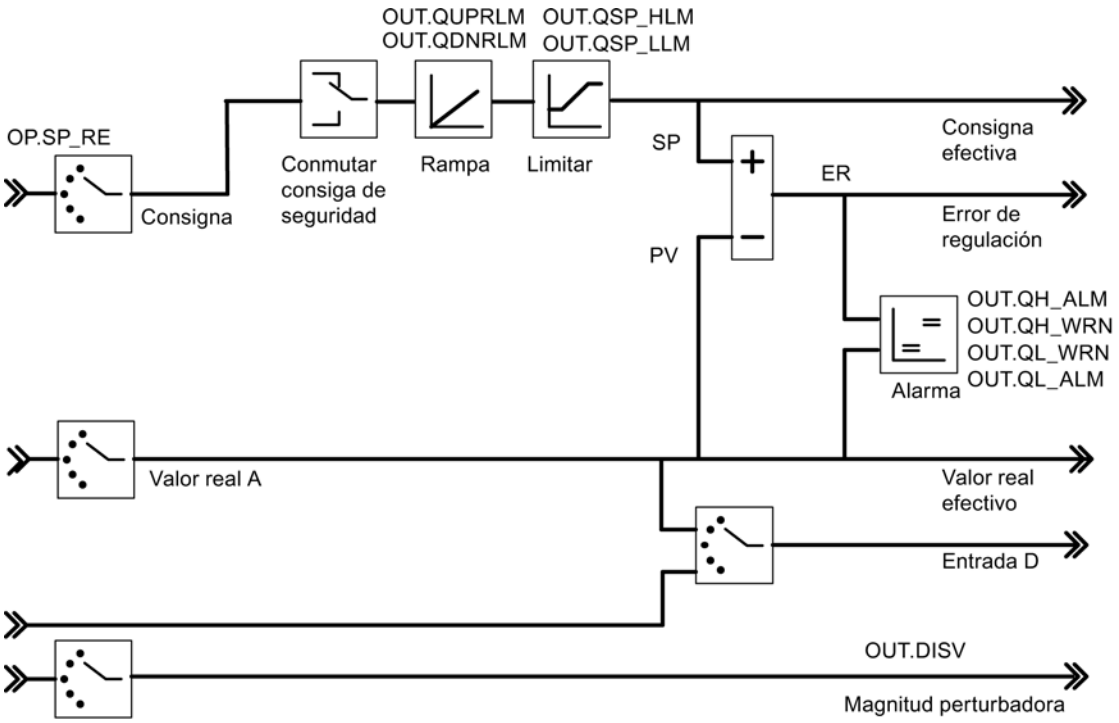
La figura siguiente muestra la salida del regulador paso a paso sin realimentación de posición (FM 355-2 S)



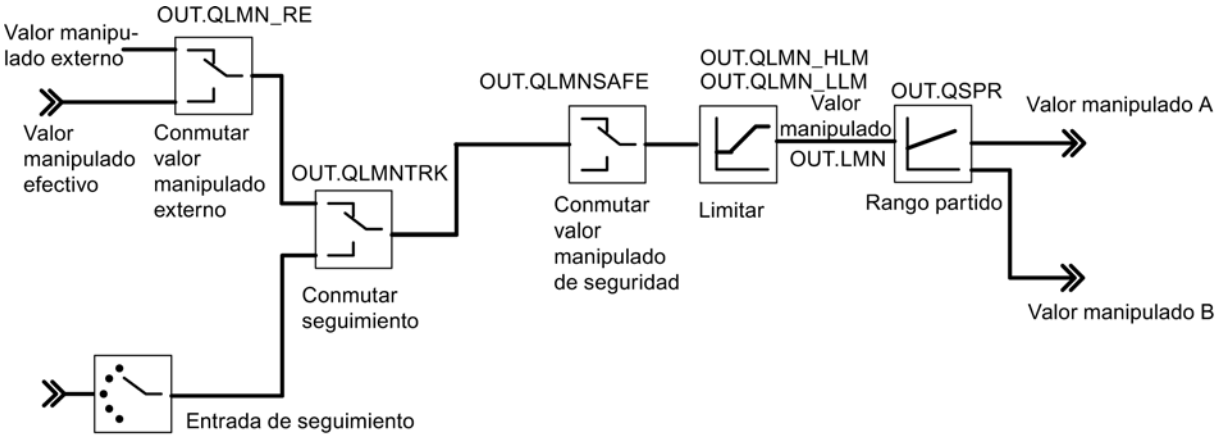
Generación de los parámetros de salida

En las siguientes figuras se representan los puntos del módulo en los que se generan los parámetros de salida de la instrucción FMT_PID.

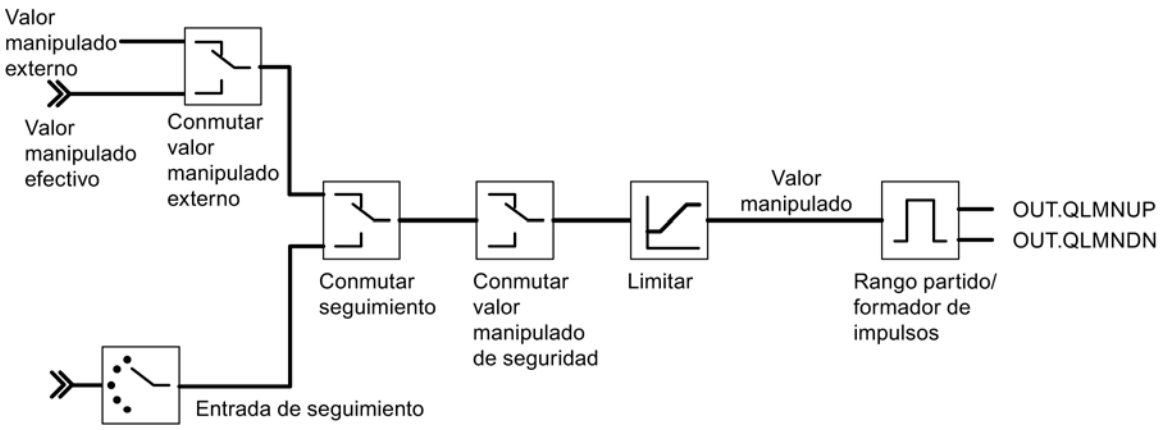
La figura siguiente muestra el cálculo del error de regulación en el regulador de valor fijo o en cascada



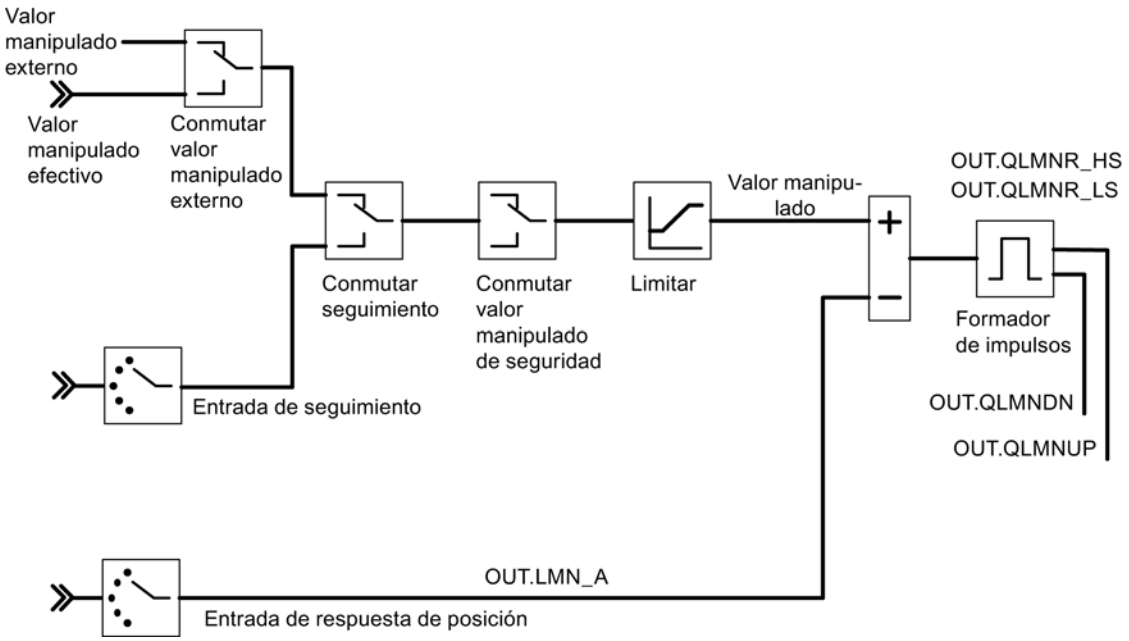
La figura siguiente muestra la salida del regulador continuo (FM 355-2 C)



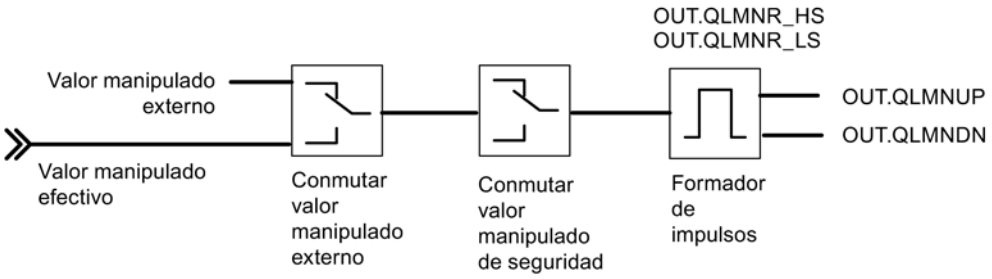
La figura siguiente muestra la salida del regulador de impulsos (FM 355-2 S)



La figura siguiente muestra la salida del regulador paso a paso con realimentación de posición (FM 355-2 S)



La figura siguiente muestra la salida del regulador paso a paso sin realimentación de posición (FM 355-2 S)



Consulte también

Descripción de FMT_PID (Página 7528)

Parámetro STATUS_H

STATUS_H indica un valor de diagnóstico a través de la búsqueda del punto de inversión en el proceso de calentamiento.

Si se muestra un valor de diagnóstico 2xxxx, la optimización finaliza, pero los parámetros del regulador son inseguros.

Si se muestra un valor de diagnóstico 3xxxx, la optimización se interrumpe (PHASE = 0 y TUN_ON = FALSE) y el regulador cambia al modo de operación anterior.

STATUS_ H	Descripción	Solución
0	Parámetros del regulador predeterminados, ninguno o ninguno nuevo	
10000	Optimización finalizada + parámetros adecuados encontrados para el regulador	
2xxxx	Optimización finalizada + parámetros del regulador inseguros	
2xxx1	Tiempo de posicionamiento del motor efectivo >= 65% del tiempo del punto de inversión T_P_INF	La compensación de TU y T_P_INF ya no se realiza. De ese modo se diseña un regulador suave.
2xx2x	Punto de inversión no alcanzado (sólo en caso de excitación con salto de consigna)	Si el regulador oscila, atenuar los parámetros del regulador o repetir el ensayo con TUN_DLMN menor.
2x1xx	Error de estimación TU <= 3*Tiempo de muestreo	Repetir el ensayo.
2x2xx	Error de estimación grave (TU < tiempo de muestreo)	Repetir el ensayo. Caso especial, proceso puramente PT1: No repetir el ensayo, atenuar los parámetros del regulador si procede.
2x3xx	Error de estimación TU demasiado grande	Repetir el ensayo con mejores condiciones.
21xxx	Error de estimación N_PTN < 1	Repetir el ensayo con mejores condiciones.
22xxx	Error de estimación N_PTN > 10	Repetir el ensayo con mejores condiciones.
3xxxx	Optimización cancelada en fase 1 debido a parametrización errónea	
30001	Activación simultánea de TUN_ON y TUN_ST	Volver a iniciar la optimización.

STATUS_ H	Descripción	Solución
30002	|TUN_DLMN| < 5 % o cifra de la modificación efectiva de LMN < 5 % • Transición a fase 99 (fase 0 en optimización paralela) • Salida LMN = LMN0	Puede que haya parametrizado un valor demasiado bajo para TUN_DLMN o que la magnitud manipulada estuviera próxima a un límite antes del inicio de la optimización. En ese caso hay que impedir que el regulador se ajuste a la nueva consigna y abandone innecesariamente el punto de operación estacionario (no es posible en la optimización paralela). Proceda del siguiente modo: • Dado el caso, retire la consigna y ponga TUN_ON en FALSE • Corrija TUN_DLMN • Compruebe los límites de la variable manipulada si TUN_DLMN >= 5%. • Volver a iniciar la optimización.
30004	|TUN_DLMN| < 5 % o modificación efectiva de LMN < 5 % Diferencia efectiva de variable manipulada limitada por los límites de rango partido y no por los límites de la variable manipulada. • Transición a fase 99 (fase 0 en optimización paralela) • Salida LMN = LMN0	Véase STATUS_H=30002: Recuerde además que p. ej. con LMN_A < 5 % no es posible una optimización de calentamiento con TUN_DLMN negativo (razón: La potencia de refrigeración no puede modificarse).
30009	Modo de seguridad	Finalice el modo de seguridad y vuelva a iniciar la optimización.

Nota

Al final de la fase 1 STATUS_H se pone a 0.

Si se cancela la optimización en la fase 2, STATUS_H = 0. Sin embargo, STATUS_D sigue mostrando el estado del último cálculo del regulador.

Consulte también

Diagramas de bloques (Página 7540)

Parámetro STATUS_C

STATUS_C indica un valor de diagnóstico a través de la búsqueda del punto de inversión en la refrigeración.

Si se muestra un valor de diagnóstico 2xxxx, la optimización finaliza, pero los parámetros del regulador son inseguros.

Si se muestra un valor de diagnóstico 3xxxx, la optimización se interrumpe (PHASE = 0 y TUN_ON = FALSE) y el regulador cambia al modo de operación anterior.

STATUS_ C	Descripción	Solución
0	Parámetros del regulador predeterminados, ninguno o ninguno nuevo	
10000	Optimización finalizada + parámetros adecuados encontrados para el regulador	
2xxxx	Optimización finalizada + parámetros del regulador inseguros	
2x3xx	Error de estimación TU demasiado grande	Repetir el ensayo con mejores condiciones.
21xxx	Error de estimación N_PTN < 1	Repetir el ensayo con mejores condiciones.
22xxx	Error de estimación N_PTN > 10	Repetir el ensayo con mejores condiciones.
3xxxx	Optimización cancelada en fase 1 debido a parametrización errónea	
30001	Activación simultánea de TUN_ON y TUN_CST	Volver a iniciar la optimización.
30002	Sólo en el regulador paso a paso sin realimentación de posición analógica: |TUN_CLMN| < 5 % o cifra de la modificación efectiva de LMN < 5 % • Transición a la fase 0 • TUN_ON = FALSE	Puede que haya parametrizado un valor demasiado bajo para TUN_CLMN o que la magnitud manipulada estuviera próxima a un límite antes del inicio de la optimización. En ese caso hay que impedir que el regulador se ajuste a la nueva consigna y abandone innecesariamente el punto de operación estacionario (no es posible en la optimización paralela). Proceda del siguiente modo: • Dado el caso, retire la consigna y ponga TUN_ON en FALSE. • Corrija TUN_CLMN • Compruebe los límites de la variable manipulada si |TUN_CLMN| >= 5% y el signo era correcto. • Volver a iniciar la optimización.
30003	TUN_CLMN <= -5%, pero LMN_LLM > -5%. • Transición a la fase 0 • TUN_ON = FALSE	El límite inferior LMN_LLM es demasiado alto (p. ej. 0%) e impide por ello la indicación de una potencia de refrigeración. • Corrija el límite inferior LMN_LLM. • Reiniciar la optimización.
30004	TUN_CLMN > -5 % o modificación efectiva de LMN > -5 % Diferencia efectiva de la variable manipulada limitada por los límites de rango partido y no por los límites de la variable manipulada. • Transición a la fase 0 • TUN_ON = FALSE	Véase STATUS_C=30002: Recuerde además que p. ej. con LMN_A < 5 % no es posible una optimización de calentamiento con TUN_DLMN negativo (razón: La potencia de refrigeración no puede modificarse).
30008	TUN_CST pero ninguna optimización de calentamiento anterior • Transición a la fase 0 • TUN_ON = FALSE	Realizar primero una optimización de calentamiento
30009	Modo de seguridad	Finalice el modo de seguridad y vuelva a iniciar la optimización.

Consulte también

Diagramas de bloques (Página 7540)

Parámetro STATUS_D

STATUS_D	Descripción
0	No se han calculado parámetros del regulador.
110	N_PTN <= 1.5 Tipo de sistema I rápido
121	N_PTN > 1.5 Tipo de sistema I
122	N_PTN = 1.9 Tipo de sistema I tras fase 7 (antes N_PTN > 1.9)
200	N_PTN > 1.9 Tipo de sistema II (rango de transición)
310	N_PTN >= 2.1 Tipo de sistema III rápido
320	N_PTN > 2.6 Tipo de sistema III

Nota

Si la optimización se cancela en la fase 2, entonces STATUS_H = 0. Sin embargo, STATUS_D sigue mostrando el estado del último cálculo del regulador.

Cuanto mayor sea el valor de STATUS_D, mayor será el orden del proceso regulado, mayor la relación TU/TA y más suaves los parámetros del regulador.

Consulte también

Diagramas de bloques (Página 7540)

FMT_PAR

Descripción de FMT_PAR

La instrucción FMT_PAR permite modificar online otros parámetros, que no pueden predeterminarse con la instrucción FMT_PID.

Llamada

FMT_PAR debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355-2.

Para ahorrar tiempo de ejecución, FMT_PAR sólo debe llamarse con LOAD=TRUE cuando deben modificarse parámetros.

Funcionamiento

Con FMT_PAR se pueden modificar un parámetro REAL y un parámetro INT en cada llamada.

En Parámetro INDEX indique el número de índice del parámetro que desea modificar. En el parámetro de entrada VALUE_R o VALUE_I indique los nuevos valores.

Para modificar varios parámetros, llame el mismo DB de instancia varias veces consecutivas con LOAD_PAR = TRUE y con números de índice diferentes.

Si se utiliza el FM 355-2 en periferia descentralizada, pueden ser necesarios varios ciclos hasta que los parámetros se hayan transmitido al FM 355-2. Hasta que la transmisión no haya finalizado, el parámetro LOAD_PAR mantiene el valor TRUE. Por ello, para modificar parámetros llame la instrucción FMT_PAR de forma repetida hasta que LOAD_PAR = FALSE.

Nota

Recuerde que los parámetros que ha cambiado con FMT_PAR se sobrescriben en el arranque de la CPU con los parámetros de los datos del sistema.

Ejemplo

Desea cambiar el tiempo de subida de la rampa para la magnitud de referencia durante el funcionamiento, y además utilizar diferentes valores de entrada analógica como valor real independientemente del estado del proceso.

- Para parametrizar el tiempo de subida de la rampa para la referencia a 10.0 llame FMT_PAR con INDEX = 30 y VALUE_R = 10.0 durante el funcionamiento.
- Para parametrizar el valor de entrada analógica 4 como valor real del módulo, llame FMT_PAR durante el funcionamiento con INDEX = 50 y VALUE_I = 4.

Parámetros de entrada de FMT_PAR

Tabla 9-239

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	0	Número de canal de regulador al que se refiere el DB de instancia. Son válidos los valores entre 0 y 3.
INDEX	4.0	INT	0	0 a 100
VALUE_R	6.0	REAL	0.0	En función del parámetro en cuestión
VALUE_I	10.0	INT	0	En función del parámetro en cuestión

Parámetros de salida de FMT_PAR

Tabla 9-240

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	12.0	WORD	W#16#0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de la instrucción WRREC.

Consulte también

WRREC: Escribir registro (Página 6738)

Parámetros de entrada/salida de FMT_PAR

Tabla 9-241

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LOAD_PAR	+14.0	BOOL	FALSE	Si el parámetro de entrada/salida LOAD_PAR está activado, VALUE_R o VALUE_I se escriben en el módulo. INDEX indica qué parámetro se sobrescribe. A continuación LOAD_PAR se desactiva.

Parámetro INDEX

INDEX	Descripción	Tipo de datos
0	Ningún parámetro seleccionado	-
1	Constante de tiempo de filtro para entrada analógica	REAL
2	Normalización entrada analógica fin de rango de medición equivale al 100%	REAL
3	Normalización entrada analógica inicio de rango de medición equivale al 0%	REAL
4	Línea poligonal, valor de interpolación 1 lado de entrada	REAL
5	Línea poligonal, valor de interpolación 2 lado de entrada	REAL
6	Línea poligonal, valor de interpolación 3 lado de entrada	REAL
7	Línea poligonal, valor de interpolación 4 lado de entrada	REAL
8	Línea poligonal, valor de interpolación 5 lado de entrada	REAL
9	Línea poligonal, valor de interpolación 6 lado de entrada	REAL
10	Línea poligonal, valor de interpolación 7 lado de entrada	REAL
11	Línea poligonal, valor de interpolación 8 lado de entrada	REAL
12	Línea poligonal, valor de interpolación 9 lado de entrada	REAL
13	Línea poligonal, valor de interpolación 10 lado de entrada	REAL

INDEX	Descripción	Tipo de datos
14	Línea poligonal, valor de interpolación 11 lado de entrada	REAL
15	Línea poligonal, valor de interpolación 12 lado de entrada	REAL
16	Línea poligonal, valor de interpolación 13 lado de entrada	REAL
17	Línea poligonal, valor de interpolación 1 lado de salida	REAL
18	Línea poligonal, valor de interpolación 2 lado de salida	REAL
19	Línea poligonal, valor de interpolación 3 lado de salida	REAL
20	Línea poligonal, valor de interpolación 4 lado de salida	REAL
21	Línea poligonal, valor de interpolación 5 lado de salida	REAL
22	Línea poligonal, valor de interpolación 6 lado de salida	REAL
23	Línea poligonal, valor de interpolación 7 lado de salida	REAL
24	Línea poligonal, valor de interpolación 8 lado de salida	REAL
25	Línea poligonal, valor de interpolación 9 lado de salida	REAL
26	Línea poligonal, valor de interpolación 10 lado de salida	REAL
27	Línea poligonal, valor de interpolación 11 lado de salida	REAL
28	Línea poligonal, valor de interpolación 12 lado de salida	REAL
29	Línea poligonal, valor de interpolación 13 lado de salida	REAL
30	Tiempo de subida de la rampa para referencia	REAL
31	Consigna de seguridad	REAL
32	Offset de combinación de consigna (regulador de relación/mixto)	REAL
33	Factor para valor real B (regulador de tres componentes)	REAL
34	Factor para valor real C (regulador de tres componentes)	REAL
35	Offset para combinación de valor real (regulador de tres componentes)	REAL
36	Factor para combinación de referencia	REAL
37	Punto de operación	REAL
38	No se utiliza	REAL
39	Vértices de la función de rango partido: Inicio de rango señal de entrada A	REAL
40	Vértices de la función de rango partido: Fin de rango señal de entrada A	REAL
41	Vértices de la función de rango partido: Inicio de rango señal de salida A	REAL
42	Vértices de la función de rango partido: Fin de rango señal de salida A	REAL
43	Vértices de la función de rango partido: Inicio de rango señal de entrada B	REAL
44	Vértices de la función de rango partido: Fin de rango señal de entrada B	REAL
45	Vértices de la función de rango partido: Inicio de rango señal de salida B	REAL
46	Vértices de la función de rango partido: Fin de rango señal de salida B	REAL
47	Duración mínima del impulso	REAL
48	Duración mínima de pausa	REAL

INDEX	Descripción	Tipo de datos
49	Selección de la referencia SP o SP_RE para el regulador • -1: Consigna SP_RE de la instrucción • 0 a 3: Valor de entrada analógica 0 a 3 • 16 a 19: Magnitud manipulada (LMN) de regulador 0 a 3 • 32 a 35: Magnitud manipulada A de regulador 0 a 3 • 48 a 51: Magnitud manipulada B de regulador 0 a 3	INT
50	Selección de la magnitud reguladora principal valor real A para el regulador • -1: Valor real A = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3	INT
51	Selección de la magnitud reguladora auxiliar valor real B para el regulador • -1: Valor real B = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3	INT
52	Selección de la magnitud reguladora auxiliar valor real C para el regulador • -1: Valor real C = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3	INT
53	Selección de la magnitud reguladora auxiliar valor real D para el regulador • -1: Valor real D = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3 • 16 a 19: Magnitud manipulada (LMN) de regulador 0 a 3	INT
54	Selección de la magnitud perturbadora DISV para el regulador • -1: Magnitud perturbadora = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3	INT
55	Selección en el conmutador de entrada de seguimiento. En el punto de medida MP TRACKPER se puede leer el valor con tooltip. • -1: Seguimiento de posición = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3	INT
56	Selección en el conmutador de entrada de realimentación de posición en regulador paso a paso con realimentación de posición. • -1: Seguimiento de posición = 0.0 • 0 a 3: Valor de entrada analógica 0 a 3	INT
57	Selección de la señal para la conmutación al valor de seguridad para la variable manipulada del regulador • -1: Sólo asignación predeterminada a través del parámetro SAFE_ON de la instrucción FMT_PID • 0 a 7: Asignación predeterminada con el parámetro SAFE_ON de la instrucción FMT_PID predeterminado por la entrada digital 0 a 7	INT
58	Selección de la señal para la conmutación a la función de seguimiento de la variable manipulada del regulador • -1: Sólo asignación predeterminada a través del parámetro LMNTRKON de la instrucción FMT_PID • 0 a 7: Asignación predeterminada con el parámetro LMNTRKON de la instrucción FMT_PID predeterminado por la entrada digital 0 a 7	INT

INDEX	Descripción	Tipo de datos
59	Selección de la señal para la conmutación de la variable manipulada del regulador a LMN_RE • -1: Sólo asignación predeterminada a través del parámetro LMN_REON de la instrucción FMT_PID • 0 a 7: Asignación predeterminada con el parámetro LMN_REON de la instrucción FMT_PID predeterminado por la entrada digital 0 a 7	INT
60	Selección de la señal de tope superior de la realimentación de posición • -1: Sólo asignación predeterminada a través del parámetro LMNRHSRE de la instrucción FMT_PID • 0 a 7: Asignación predeterminada con el parámetro LMNRHSRE de la instrucción FMT_PID predeterminado por la entrada digital 0 a 7	INT
61	Selección de la señal de tope inferior de la realimentación de posición • -1: Sólo asignación predeterminada a través del parámetro LMNRLSRE de la instrucción FMT_PID • 0 a 7: Asignación predeterminada con el parámetro LMNRLSRE de la instrucción FMT_PID predeterminado por la entrada digital 0 a 7	INT

FMT_DS1

Descripción de FMT_DS1

Para leer el registro de diagnóstico DS1 se dispone de la instrucción FMT_DS1.

Funcionamiento

Si el parámetro es READ_DS1 = TRUE, se lee el registro de diagnóstico DS1 del FM 355-2 y se deposita en la estructura DS1 del DB de instancia. A continuación se desactiva el parámetro de entrada/salida.

Esa lectura sólo tiene sentido si en el DS0 se notifica un error en un canal.

Llamada

FMT_DS1 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355-2.

FMT_DS1 no requiere una marcha de inicialización.

Para obtener siempre los valores de diagnóstico actuales, ponga el parámetro READ_DS1 cíclicamente en TRUE en el OB 82.

Una vez que la lectura de los valores de diagnóstico se ha completado correctamente, FMT_DS1 desactiva el parámetro READ_DS1.

Parámetros de entrada de FMT_DS1

Tabla 9-242

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de FMT_DS1

Tabla 9-243

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
DS1	2.0	STRUCT		El registro de diagnóstico DS1 se compone de los bytes 0 a 12. Los primeros 4 bytes son idénticos al registro de diagnóstico DS0. Auto-Hotspot
RET_VALU	16.0	WORD	W#16#0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de la instrucción RDREC.

Consulte también

RDREC: Leer registro (Página 6735)

Parámetros de entrada/salida de FMT_DS1

Tabla 9-244

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
READ_DS1	18.0	BOOL	FALSE	Si el parámetro es READ_DS1 = TRUE, se lee el registro de diagnóstico DS1 del FM 355-2 y se deposita en la estructura DS1 del DB de instancia. A continuación se desactiva el parámetro de entrada/salida.

Ocupación de DS1

La siguiente tabla muestra la ocupación del registro de diagnóstico DS1. Los 4 primeros bytes son idénticos a los del registro de diagnóstico DS0. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Byte	Bit	Significado	Observación	Nº de evento
0	0	Avería en módulo	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Error interno	Se activa para todos los errores internos: • Vigilancia de tiempo activada • Falla la EPROM • Error ADC/DAC • Entrada analógica error de hardware	8:x:01
	2	Error externo	Se activa para todos los errores externos: • Falta tensión auxiliar externa • Error de parametrización • Entrada analógica rotura de hilo (sólo en el rango de 4 a 20 mA) • Entrada analógica Rango de medida rebasado por defecto • Entrada analógica Rango de medida rebasado por exceso • Salida analógica Rotura de carga aparente • Salida analógica Cortocircuito	8:x:02
	3	Error en un canal	Consultar a partir del byte 7 para más detalles sobre la descodificación	8:x:03
	4	Falta tensión auxiliar externa	Fallo de la alimentación de 24 V del FM 355-2	8:x:04
	6	No utilizado	-	-
	7	Error de parametrización	El módulo no puede evaluar un parámetro. Causa: Parámetros desconocidos o combinación de parámetros no permitida.	8:x:07
1	0 ... 3	Clase de módulo	Siempre tiene asignado el valor 8.	-
	4	Diagnóstico específico de canal	Se activa cuando el módulo puede proporcionar información de canal adicional y hay un error de canal (ver bytes 7 a 12)	-
2	3	Vigilancia de tiempo activada (watch dog)	Error de hardware	8:x:33
3	2	Falla la EPROM	Módulo defectuoso	8:x:42
	4	Error ADC/DAC	Módulo defectuoso	8:x:44
4	0 ... 7	Tipo de canal	Siempre tiene asignado el valor 75H.	-
5	0 ... 7	Longitud de la información de diagnóstico	Siempre tiene asignado el valor 8.	-
6	0 ... 7	Número de canales	Siempre tiene asignado el valor 5 (4 reguladores + 1 canal de referencia)	-
7	0 ... 4	Vector de error de canal	Cada canal tiene asignado un bit (0...3; 4 para canal de referencia).	-

Byte	Bit	Significado	Observación		Nº de evento
8	0	Entrada analógica error de hardware	Diagnóstico de canal, canal 0		8:x:B0
	1	No utilizado			8:x:B1
	2	Entrada analógica rotura de hilo (sólo en el rango de 4 a 20 mA)			8:x:B2
	3	No utilizado			8:x:B3
	4	Entrada analógica Rango de medida rebasado por defecto			8:x:B4
	5	Entrada analógica Rango de medida rebasado por exceso			8:x:B5
	6	Salida analógica Rotura de hilo		Sólo con salida de corriente del regulador K	8:x:B6
	7	Salida analógica Cortocircuito		Sólo con salida de tensión del regulador K	8:x:B7
9	0 ... 7	Véase el byte 8	Diagnóstico de canal, canal 1		Véase arriba
10	0 ... 7	Véase el byte 8	Diagnóstico de canal, canal 2		Véase arriba
11	0 ... 7	Véase el byte 8	Diagnóstico de canal, canal 3		Véase arriba
12	0 ... 5	Véase el byte 8	Diagnóstico para canal de referencia		Véase arriba

FMT_TUN

Descripción de FMT_TUN

Con la instrucción FMT_TUN se obtiene información detallada adicional (p. ej. parámetros de regulador guardados) durante la optimización del regulador.

Nota

No obstante, para realizar una optimización de regulador es suficiente la instrucción FMT_PID. Dicha instrucción permite iniciar la optimización y observar la información de estado.

Llamada

FMT_TUN debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355-2.

FMT_TUN no requiere una marcha de inicialización.

Ponga el parámetro READ_OUT cíclicamente en TRUE para ver los valores actuales en cada momento.

Una vez que la lectura de los parámetros se ha completado correctamente, FMT_TUN desactiva el parámetro READ_OUT.

Funcionamiento

Tras cargar los parámetros del regulador con LOAD_PAR = TRUE o los parámetros de operación con LOAD_OP = TRUE, los parámetros no son efectivos de forma inmediata. Hasta que no ha transcurrido el tiempo de ciclo del FM355-2 (hasta 500 ms dependiendo del número de canales), los parámetros del regulador no podrán leerse con READ_PAR ni los parámetros de salida con READ_OUT. Lo mismo rige tras la carga de los datos del sistema (transición de la CPU de STOP a RUN) o al leer los parámetros de salida con FMT_TUN.

Llamada en una configuración descentralizada

En una configuración descentralizada y en caso de llamada simultánea de FMT_PID y FMT_TUN, deben considerarse los siguientes aspectos:

- LOAD_OP no puede activarse simultáneamente en FMT_PID y FMT_TUN.
- READ_OUT no puede activarse simultáneamente en FMT_PID y FMT_TUN.

Motivo: Ambas instrucciones acceden al FM355-2 a través de los mismos registros. Por ello hay que garantizar que en el mismo momento sólo uno de los dos FBs lea o escriba un registro.

Parámetros de entrada de FMT_TUN

Tabla 9-245

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	0	Número de canal de regulador al que se refiere el DB de instancia. Son válidos los valores entre 0 y 3.

Parámetros de salida de FMT_TUN

Tabla 9-246

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PI_GAIN	4.0	REAL	0.0	Parámetros de regulador PI
PI_TI	8.0	REAL	0.0	Parámetros de regulador PI

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PID_GAIN	12.0	REAL	0.0	Parámetros de regulador PID
PID_TI	16.0	REAL	0.0	Parámetros de regulador PID
PID_TD	20.0	REAL	0.0	Parámetros de regulador PID
SAV_PFAC	24.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.PFAC_SP
SAV_GAIN	28.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.GAIN
SAV_TI	32.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.TI
SAV_TD	36.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.TD
SAV_D_F	40.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.D_T
SAV_RATI	44.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.RATIOFAC
SAV_CONZ	48.0	REAL	0.0	Parámetros de regulador guardados FMT_PID.PAR.CON_ZONE
SAV_PSEL	52.0	BOOL	FALSE	Parámetros de regulador guardados FMT_PID.PAR.P_SEL
SAV_CZON	52.1	BOOL	FALSE	Parámetros de regulador guardados FMT_PID.PAR.CONZ_ON
RET_VALU	54.0	WORD	W#16#0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de la instrucción RDREC.

Consulte también

RDREC: Leer registro (Página 6735)

Parámetros de entrada/salida de FMT_TUN

Tabla 9-247

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
READ_OUT	56.0	BOOL	FALSE	Si el parámetro de entrada/salida READ_OUT está activado, el FM 355-2 lee los parámetros de salida.

Variables estáticas FMT_TUN.OUT

Tabla 9-248

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
vers_nr	58.0	WORD	W#16#3230	El parámetro vers_nr no puede ser modificado por el usuario. Identifica el inicio de los parámetros de salida que lee el módulo cuando se activa READ_OUT = TRUE.
PV	60.0	REAL	0.0	Valor real efectivo.
LMN	64.0	REAL	0.0	En la salida LMN se indica la variable manipulada que actúa efectivamente. En el regulador paso a paso sin realimentación de posición analógica, se indica la acción P + D ilimitada en la salida LMN. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
QLMNSAFE	68.1	BOOL	FALSE	Si la salida QLMNSAFE está activada, como variable manipulada se indica la variable manipulada de seguridad.
QLMNTRK	68.3	BOOL	FALSE	La salida QLMNTRK indica si la variable manipulada se corrige mediante una entrada analógica.
QLMN_RE	68.4	BOOL	FALSE	La salida QLMN_RE indica si LMN_REON está activada y si la variable manipulada externo LMN_RE se aplica como variable manipulada.
QLMNR_HS	68.5	BOOL	FALSE	QLMNR_HS = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
QLMNR_LS	68.6	BOOL	FALSE	QLMNR_LS = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en el regulador paso a paso).
QLMNR_ON	68.7	BOOL	FALSE	QLMNR_ON = TRUE significa: Regulador paso a paso con realimentación de posición.
QSTEPCON	69.0	BOOL	FALSE	Regulador paso a paso/regulador de impulsos 0 = regulador de impulsos o continuo 1 = regulador paso a paso
QSPR	69.1	BOOL	FALSE	Si la salida QSPR está activada, el regulador continuo trabaja en modo de rango partido.
QMAN_FC	69.3	BOOL	FALSE	El regulador es un regulador piloto que, mediante la operación manual de un regulador secundario, se corrige según el valor real de este último, o cuya acción I se detiene porque la consigna o la magnitud manipulada del regulador secundario están en el límite.
QH_ALM	70.0	BOOL	FALSE	Límite más alto H_ALM rebasado por valor real o error de regulación.
QH_WRN	70.1	BOOL	FALSE	Segundo límite más alto H_WRN rebasado por valor real o error de regulación.
QL_WRN	70.2	BOOL	FALSE	Segundo límite más bajo L_WRN rebasado por valor real o error de regulación.
QL_ALM	70.3	BOOL	FALSE	Límite más bajo L_ALM rebasado por valor real o error de regulación.
QLMN_HLM	70.4	BOOL	FALSE	La salida QLMN_HLM notifica que se ha alcanzado el límite superior de la variable manipulada LMN_HLM. (No en el regulador paso a paso sin realimentación de posición analógica).

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QLMN_LLM	70.5	BOOL	FALSE	La salida QLMN_LLM notifica que se ha alcanzado el límite inferior de la variable manipulada LMN_LLM. (No en el regulador paso a paso sin realimentación de posición analógica)
QPAR_F	70.6	BOOL	FALSE	El módulo comprueba la validez de los parámetros. Un error de parametrización se indica en la salida QPAR_F.
QCH_F	70.7	BOOL	FALSE	La salida QCH_F se activa cuando el canal de regulación no puede arrojar resultados válidos. El error de canal (p. ej. rotura de hilo) también se activa con QPAR_F = 1 o QMOD_F = 1. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.
QUPRLM	71.0	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida QUPRLM está activada, se limita la rampa de subida de la consigna.
QDNRLM	71.1	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida QDNRLM está activada, se limita la rampa de bajada de la consigna.
QSP_HLM	71.2	BOOL	FALSE	La salida QSP_HLM notifica que se ha alcanzado el límite superior de la consigna SP_HLM.
QSP_LLM	71.3	BOOL	FALSE	La salida QSP_LLM notifica que se ha alcanzado el límite inferior de la consigna SP_LLM.
QLMNUP	71.4	BOOL	FALSE	Salida "Señal alta de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QLMNDN	71.5	BOOL	FALSE	Salida "Señal baja de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QTUN_ON	71.6	BOOL	FALSE	QTUN_ON = TRUE indica que se está produciendo una optimización.
PAR_ACT	71.7	BOOL	FALSE	Actualizar parámetros de regulador
SP	72.0	REAL	0.0	Consigna efectiva.
ER	76.0	REAL	0.0	Error de regulación
DISV	80.0	REAL	0.0	Magnitud perturbadora efectiva. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_A	84.0	REAL	0.0	Con el regulador continuo o de impulsos, en la salida LMN_A se muestra la variable manipulada A de la función de rango partido, y con el regulador paso a paso con realimentación de posición analógica se muestra la realimentación de posición. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
LMN_B	88.0	REAL	0.0	Con el regulador continuo o de impulsos, en la salida LMN_B se indica la variable manipulada B de la función de rango partido. Se admiten valores comprendidos entre -100.0 y 100.0 (%).
PHASE	92.0	INT	0	Muestra el progreso de la optimización del regulador. *) 0..7
STATUS_H	94.0	INT	0	Valor de diagnóstico para búsqueda de punto de inversión en optimización de calentamiento. Véase Parámetro FMT_PID.STATUS_H (Página 7546)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
STATUS_C	96.0	INT	0	Valor de diagnóstico para búsqueda de punto de inversión en optimización de refrigeración. Véase Parámetro FMT_PID.STATUS_C (Página 7547)
STATUS_D	98.0	INT	0	Valor de diagnóstico para diseño de regulador en optimización de calentamiento. Véase Parámetro FMT_PID.STATUS_D (Página 7549)
ZONE_TUN	100.0	WORD	W#16#0	En la representación HEX cada una de las cuatro posiciones corresponde a un canal, en el siguiente orden de izquierda a derecha: Canal 0, 1, 2, 3. ZONE_TUN = 0 significa que el canal utilizado no puede optimizarse conjuntamente con el resto de canales. Un valor <> 0000 muestra - con un 1 en cada caso - los canales que se optimizan conjuntamente.
GAIN_P	152.0	REAL	0.0	Ganancia del proceso identificada (magnitud física) /%. En el tipo de sistema I, se tiende a estimar un valor demasiado bajo para GAIN_P.
TU	156.0	REAL	0.0	Tiempo de retardo identificado del proceso. TU≥ 3*Tiempo de muestreo
TA	160.0	REAL	0.0	Tiempo de compensación identificado del proceso. En el tipo de sistema I, se tiende a estimar un valor demasiado bajo para TA.
KIG	164.0	REAL	0.0	Rampa de subida máxima posible del valor real. (magnitud física) /s GAIN_P = 0.01 * KIG * TA
N_PTN	168.0	REAL	0.0	Orden del proceso. También se pueden utilizar "valores no enteros". Se admiten valores comprendidos entre 1.01 y 10.0.
TM_LAG_P	172.0	REAL	0.0	Constante temporal de un modelo PTN. (Valores razonables sólo para N_PTN>=2).
T_P_INF	176.0	REAL	0.0	Tiempo desde la excitación del proceso hasta el punto de inversión.
P_INF	180.0	REAL	0.0	Modificación del valor real desde la excitación del proceso hasta el punto de inversión.
LMN0	184.0	REAL	0.0	Variable manipulada al principio de la optimización Se calcula en la fase 1 (valor medio) Se admiten valores comprendidos entre -100.0 y 100.0 (%).
PV0	188.0	REAL	0.0	Valor real al principio de la optimización Se calcula en la fase 1 (valor medio).
PVDT0	192.0	REAL	0.0	Subida del valor real al iniciar la optimización Con adaptación de signo. (magnitud física) /s
PVDT	196.0	REAL	0.0	Subida actual de valor real Con adaptación de signo. (magnitud física) /s
PVDT_MAX	200.0	REAL	0.0	Cambio máximo de consigna (magnitud física) /s La derivación del valor real alcanza un máximo (signo adaptado, siempre > 0), se utiliza para calcular TU y KIG.
NOI_PVDT	204.0	REAL	0.0	Nivel de ruido en PVDT_MAX en % Cuanto mayor sea el nivel de ruido, menos precisos (de efecto más suave) serán los parámetros del regulador.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
NOISE_PV	208.0	REAL	0.0	Ruido absoluto en valor real (magnitud física) /s Diferencia entre el valor real máximo y mínimo en la fase 1.
FIL_CYC	212.0	INT	0	Número de ciclos del filtro de valores medios Se calcula un valor medio del valor real a partir de FIL_CYC ciclos. En caso necesario, FIL_CYC se aumenta automáticamente entre 1 y máx. 1024.
POI_CMAX	214.0	INT	0	Número máximo de ciclos tras el punto de inversión Este tiempo se utiliza para encontrar otro punto de inversión (es decir, uno mejor) en caso de ruidos de medición. Sólo entonces concluye la optimización.
POI_CYCL	216.0	INT	0	Número de ciclos tras el punto de inversión

Consulte también

Parámetro STATUS_H (Página 7546)

Parámetro STATUS_C (Página 7547)

Parámetro STATUS_D (Página 7549)

FMT_PV

Descripción de FMT_PV

La instrucción FMT_PV permite leer o escribir valores del proceso (valores de entrada analógicos y digitales) para facilitar la puesta en marcha.

Llamada

FMT_PV debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355-2.

FMT_PV no requiere una marcha de inicialización.

Ponga el parámetro LOAD_PV cíclicamente en TRUE para escribir cíclicamente valores de proceso simulados en FM 355-2.

Ponga el parámetro READ_PV cíclicamente en TRUE para ver los valores del proceso actuales en cada momento.

Una vez que la lectura o escritura de los valores del proceso se ha completado correctamente, FMT_PV desactiva el parámetro LOAD_PV o READ_PV.

Simulación de los valores analógicos (LOAD_PV = TRUE)

La simulación de los valores analógicos para los canales 0 a 3 se activa a través de los conmutadores S_AION[i] o S_PVON[i], donde 0 ≤ i ≤ 3. En la figura de abajo se representa en qué punto es efectivo el valor analógico simulado.

Los valores de simulación para los canales 0 a 3 se predeterminan a través de los parámetros PV_SIM[i].

Los valores de simulación pueden hacerse efectivos en dos puntos:

- S_AION[i] = TRUE (0 ≤ i ≤ 3)
  En lugar del valor de la entrada analógica i del módulo se utiliza el valor PV_SIM[i].
- S_PVON[i] = TRUE (0 ≤ i ≤ 3)
  En lugar del valor acondicionado de la entrada analógica i del módulo se utiliza el valor PV_SIM[i].

Simulación de los valores digitales (LOAD_PV = TRUE)

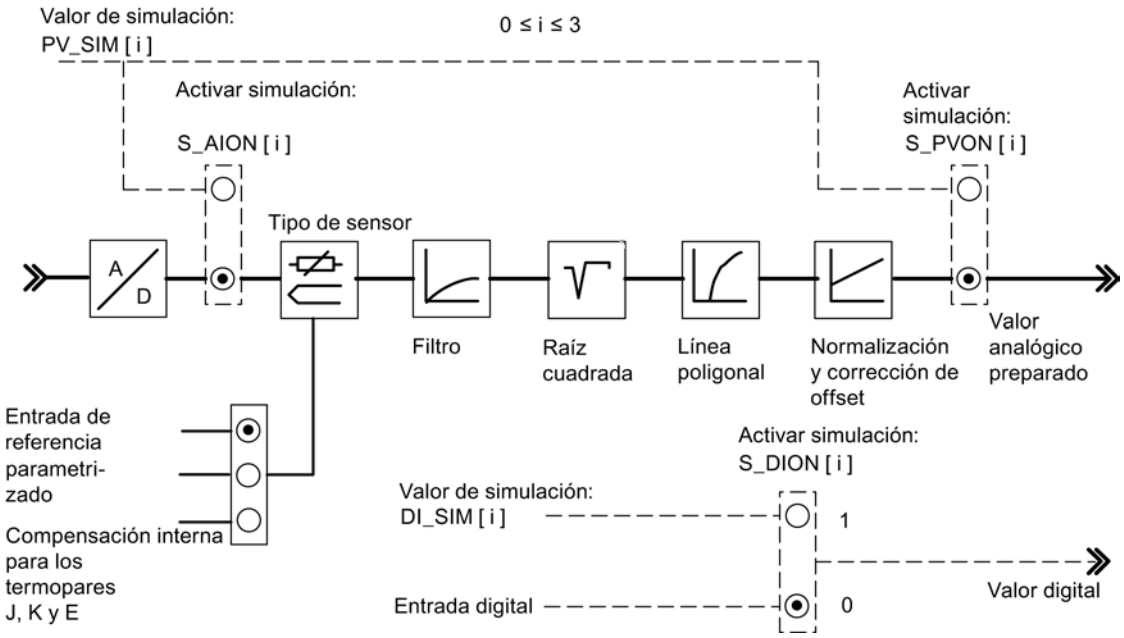
La simulación de los valores para las entradas digitales 0 a 7 se activa a través de los conmutadores S_DION[i], donde 0 ≤ i ≤ 7.

Los valores de simulación se predeterminan a través de los parámetros DI_SIM[i].

- S_DION[i] = TRUE (0 ≤ i ≤ 7)
  En lugar del valor de la entrada digital i del módulo se utiliza el valor DI_SIM[i].

Nota

Los LEDs I0 a I7 muestran siempre el estado de la entrada digital correspondiente, incluso durante la simulación.



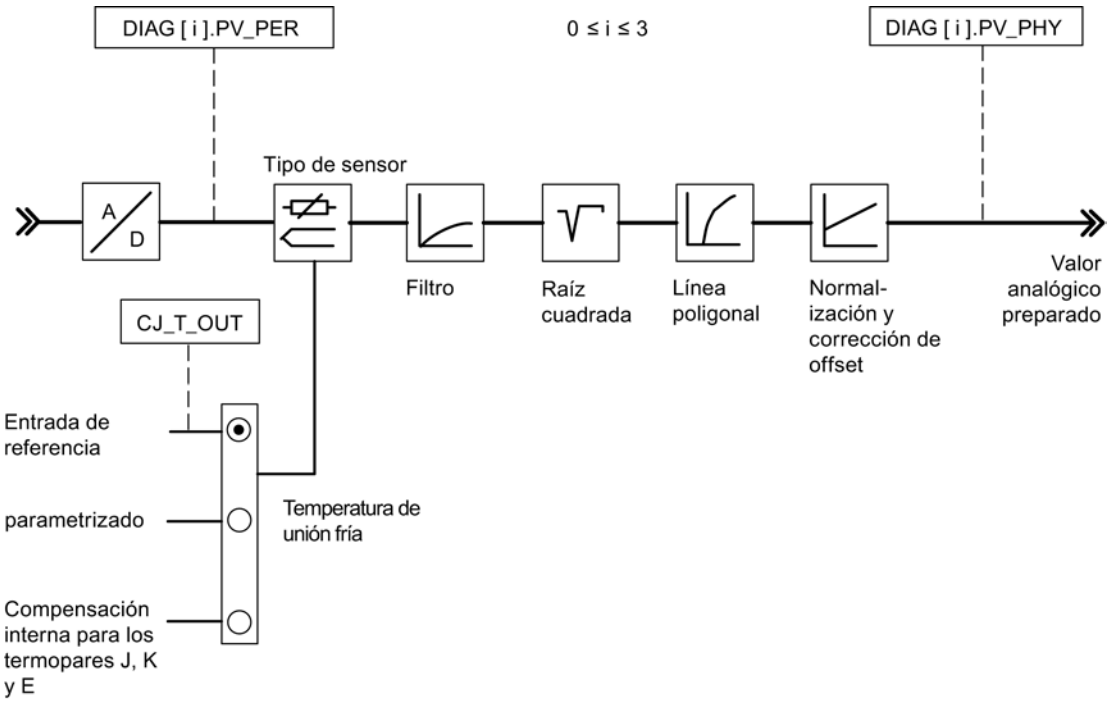
Nota

La conexión y la preasignación de los valores de simulación (forzado) no se realizan a través del software de configuración. Por ello, los conmutadores y las líneas de conexión correspondientes se representan rayados.

Indicación de los valores del proceso (READ_PV = TRUE)

Se visualizan los valores siguientes:

- En los parámetros STAT_DI[0] a STAT_DI[7] se muestra el estado real de las entradas digitales 0 a 7, aunque sean simuladas.
- En los parámetros DIAG[0].PV_PER a DIAG[3].PV_PER se muestra el valor de las entradas analógicas 0 a 3 en la unidad mA o mV. Si se ha activado la simulación del valor de entrada analógico, se muestra el valor simulado.
- En los parámetros DIAG[0].PV_PHY a DIAG[3].PV_PHY se muestra el valor acondicinoado de entrada analógica 0 a 3 en unidad física. Si se ha activado la simulación del valor analógico físico acondicionado, se muestra el valor simulado.



Nota

La actualización de los valores del proceso puede retrasarse con respecto a la instrucción FMT_PID si el FM 355-2 está muy cargado.

Arranque

En caso de rearranque del FM 355-2 tras Power OFF, los conmutadores de simulación del FM 355-2 vuelven a estar en FALSE.

Parámetros de entrada de FMT_PV

Tabla 9-249

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
S_AION	0.0	ARRAY [0..3] of BOOL	FALSE	Si p. ej. el conmutador S_AION[1] adopta el estado TRUE, en lugar del valor de entrada analógica 1 del módulo se utiliza el valor PV_SIM[1].
S_PVON	2.0	ARRAY [0..3] of BOOL	FALSE	Si p. ej. el selector S_PVON[1] adopta el estado TRUE, en lugar del valor acondicionado de entrada analógica 1 del módulo se utiliza el valor PV_SIM[1].
PV_SIM	4.0	ARRAY [0..3] of REAL	0.0	En la entrada PV_SIM[1] se predetermina p. ej., el valor de simulación para la entrada analógica 1. Si S_PVON = TRUE, entonces se especifica aquí el valor de entrada analógica acondicionado. Si S_PVON = FALSE y S_AION = TRUE, entonces se especifica aquí el valor de entrada analógica en mA o mV, que se convierte en un valor acondicionado mediante funciones de acondicionamiento. 0.0 a 20.0 [mA] o -1500 hasta +10000 [mV] o rango de valores técnico
S_DION	20.0	ARRAY [0..7] of BOOL	FALSE	Si p. ej. S_DION[1] se pone en TRUE, en lugar de la entrada digital 1 del módulo se utiliza el valor DI_SIM[1] como valor digital.
DI_SIM	22.0	ARRAY [0..7] of BOOL	FALSE	
MOD_ADDR	24.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Consulte también

Parámetros de salida de FMT_PV (Página 7568)

Parámetros de entrada/salida de FMT_PV (Página 7568)

Parámetros de salida de FMT_PV

Tabla 9-250

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
STAT_DI	26.0	ARRAY [0..7] of BOOL	FALSE	En los parámetros STAT_DI se muestran los estados de las entradas digitales 0 a 7.
DIAG[x].PV_PER	28+ (Número de canal x8)	ARRAY [0..3] of STRUCT	0.0	En el parámetro DIAG[x].PV_PER se muestra p. ej. el valor de entrada analógica del módulo en la unidad mA o mV.
DIAG[x].PV_PHY	32+ (Número de canal x8)	ARRAY [0..3] of STRUCT	0.0	En el parámetro DIAG[x].PV_PHY se muestra p. ej. el valor acondicionado de entrada analógica del módulo en unidad física.
RET_VALU	60.0	WORD	W#16#0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetros de entrada/salida de FMT_PV

Tabla 9-251

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LOAD_PV	62.0	BOOL	FALSE	Si el parámetro de entrada/salida LOAD_PV está activado, los valores de simulación de PV_SIM y DI_SIM se escriben en el FM 355-2 conforme a las posiciones de los conmutadores y el parámetro de entrada/salida se desactiva.
READ_PV	62.1	BOOL	FALSE	Si el parámetro READ_PV = TRUE, la temperatura de unión fría se lee del módulo. A continuación se desactiva el parámetro de entrada/salida.

Consulte también

Parámetros de entrada de FMT_PV (Página 7567)

Parámetros de salida de FMT_PV (Página 7568)

FMT_CJ_T

Descripción de FMT_CJ_T

La instrucción FMT_CJ_T permite leer la temperatura de unión fría y cambiar online la temperatura de unión fría parametrizada. Esto es necesario si una regulación de temperatura debe funcionar con varios FM 355-2 con entradas de termopar sin necesidad de conectar un Pt 100 en cada FM 355-2.

Llamada

FMT_PAR debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 355-2.

Si se utiliza el FM 355-2 en periferia descentralizada, pueden ser necesarios varios ciclos hasta que el parámetro se haya transmitido al FM 355-2. Hasta que la transmisión no haya finalizado, el parámetro LOAD_CJ mantiene el valor TRUE. Por ello, para modificar parámetros llame FMT_CJ_T de forma repetida hasta que el bloque active LOAD_CJ = FALSE.

Funcionamiento

Si p. ej. en una regulación de extrusionador con más de cuatro zonas calientes la temperatura de las uniones frías se mide con un FM 355-2, puede leerse con READ_CJ = TRUE en el parámetro CJ_T_OUT y parametrizarse en el otro FM 355-2 a través de CJ_TEMP y LOAD_CJ.

En el parámetro CJ_T_OUT está depositada la temperatura medida en la unión fría en grados °C o °F (depende de la unidad de temperatura parametrizada).

En el parámetro CJ_T_OUT se indica 0.0 si:

- No se ha parametrizado un tipo de sensor "Termopar"
- En todas las entradas analógicas se ha seleccionado la temperatura de unión fría parametrizada
- En todas las entradas analógicas se ha seleccionado la compensación interna

Parámetros de entrada de FMT_CJ_T

Tabla 9-252

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CJ_TEMP	2.0	REAL	0.0	En el parámetro CJ_TEMP se puede predeterminar la temperatura de las uniones frías.

Parámetros de salida de FMT_CJ_T

Tabla 9-253

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
CJ_T_OUT	6.0	REAL	0.0	En la salida CJ_T_OUT se muestra la temperatura de unión fría medida por el módulo si se ha parametrizado una entrada de termopar y si la temperatura de unión fría no se predetermina de forma parametrizada.
RET_VALU	10.0	WORD	W#16#0	RET_VALU contiene los bytes 2º y 3º del parámetro STATUS de las instrucciones RDREC y WRREC.

Consulte también

RDREC: Leer registro (Página 6735)

WRREC: Escribir registro (Página 6738)

Parámetros de entrada/salida FMT_CJ_T

Tabla 9-254

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LOAD_CJ	12.0	BOOL	FALSE	Si el parámetro de entrada/salida LOAD_CJ está activado, la temperatura de unión fría parametrizada en el módulo se sobrescribe con el valor CJ_TEMP y el parámetro de entrada/salida se desactiva.
READ_CJ	12.1	BOOL	FALSE	Si el parámetro READ_CJ = TRUE, la temperatura de unión fría se lee del módulo. A continuación se desactiva el parámetro de entrada/salida.

FM 455 PID Control

PID_FM_455

Descripción de PID_FM_455

La instrucción PID_FM_455 permite realizar las siguientes acciones durante el funcionamiento:

- Transferir parámetros de operación al FM 455
- Transferir parámetros de regulador al FM 455
- Leer valores del proceso del FM 455

Los parámetros de operación son todos los parámetros de entrada/salida que se encuentran en el DB de instancia de la instrucción entre los parámetros op_par y cont_par.

Los parámetros de regulación son todos los parámetros de entrada/salida que se encuentran en el DB de instancia de la instrucción después del parámetro cont_par.

Los valores del proceso son todos los parámetros de salida de la instrucción situados después del parámetro out_par.

Los datos necesarios se depositan en un DB de instancia en la CPU. Para cada canal de regulación utilizado se requiere un DB de instancia. Tras generar un DB de instancia, todos los parámetros de entrada/salida tienen asignado el valor FALSE.

Llamada

PID_FM_455 se llama generalmente en el OB 35 de alarma cíclica y requiere una marcha de inicialización, que se inicia activando el parámetro COM_RST = TRUE en el arranque de la CPU. Es posible realizar una llamada de la instrucción en el OB de arranque, pero no es necesario. Tras la marcha de inicialización, PID_FM_455 pone el parámetro COM_RST en FALSE.

PID_FM_455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Si se efectúa una llamada a la instrucción PID_FM_455 como DB multiinstancia, no se dispone de interfaz de puesta en servicio. En este caso, los FM 455 se deben poner en servicio a través de una tabla de observación.

Arranque

En el arranque y en la transición de STOP a RUN de la CPU, los parámetros del FM 455 se sobrescriben con los parámetros del SDB de la CPU.

Respuesta en caso de error

El parámetro de salida RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones RD_REC y WR_REC. RET_VALU puede evaluarse si los parámetros READ_VAR y LOAD_PAR no se desactivan.

En la llamada de la instrucción PID_FM_455 puede producirse un error de acceso a la periferia, si el FM 455 no está insertado o si no recibe suministro de tensión. En ese caso, la CPU pasa a STOP si no hay ningún OB 122 cargado en la CPU.

Funcionamiento de PID_FM_455

Influencia de los parámetros LOAD_OP, READ_VAR y LOAD_PAR

En la marcha de inicialización, PID_FM_455 lee los parámetros de los datos del sistema y los deposita en el DB de instancia. El DB de instancia y los datos del sistema están sincronizados.

La tabla siguiente muestra cómo afectan los parámetros LOAD_OP, READ_VAR y LOAD_PAR al funcionamiento de la instrucción.

Parámetro	TRUE	FALSE
LOAD_OP	Transferencia de los parámetros de operación de la CPU al FM 455 con la instrucción WR_REC	Transferencia de los parámetros de operación de la CPU al FM 455 con accesos de periferia directos en el proceso multiplex
	Tiempo de operación largo	Tiempo de operación reducido
	Los valores son efectivos tras un ciclo de la CPU o el FM. Rige el ciclo más largo.	Como máximo se requieren 3 ciclos de la CPU o del FM para que los valores sean efectivos. Rige el ciclo más largo
READ_VAR	Lectura de los valores del proceso del FM 455 con la instrucción RD_REC	Lectura de los parámetros del proceso del FM 455 con accesos directos a la periferia en el proceso multiplex
	Tiempo de ejecución prolongado	Tiempo de ejecución reducido
	Todos los parámetros se leen del FM 455	Los siguientes parámetros **no** se leen del FM 455 • SP (consigna del FM) • ER (error de regulación) • DISV (magnitud perturbadora) • LMN_A • LMN_B
	Los siguientes parámetros se actualizan en **cada** llamada de la instrucción. • PV (valor real) • LMN (variable manipulada) • Las indicaciones binarias	Los siguientes parámetros se actualizan cada **cuatro** llamadas de la instrucción. • PV (valor real) • LMN (variable manipulada) • Las indicaciones binarias
	En el arranque de la CPU PID_FM_455 adopta los siguientes valores del FM 455: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON Si estos valores se han cambiado en la CPU, en el primer arranque de la CPU se sobrescriben con los valores del FM 455	En el arranque de la CPU, los siguientes valores se leen de la instrucción y no del FM: • SP_OP • LMN_OP • SP_OP_ON • LMNOP_ON
LOAD_PAR	Todos los parámetros de regulación de la CPU se cargan en el FM 455. Esta acción es conveniente si los parámetros de regulación deben cambiarse con dependencia de los estados del proceso.	Sin transmisión de los parámetros del regulador

Una vez completada de transmisión de datos, los parámetros LOAD_OP, READ_VAR y LOAD_PAR se vuelven a poner a FALSE.

Particularidades de la transmisión en el proceso multiplex

La multiplexación de los datos que se van a transmitir en el acceso al FM 455 con accesos directos a la periferia se controla con PID_FM_455. El control de multiplexación con PID_FM_455 no funciona si dos instancias de la instrucción PID_FM_455 acceden al mismo número de canal de un módulo. El resultado son parámetros incorrectos en el FM 455 (p. ej. consigna y variable manipulada en manual) e indicaciones erróneas en los parámetros de salida de la instrucción PID_FM_455.

Almacenamiento de los parámetros en la EEPROM

En la reparametrización controlada por programa (LOAD_PAR, LOAD_OP) del módulo regulador con la instrucción PID_FM_455, aumenta su tiempo de operación. Los parámetros nuevos siempre tienen vigencia inmediata y además se depositan en una memoria no volátil (EEPROM). Tras cada almacenamiento de los parámetros en la EEPROM, el siguiente se retrasa 30 minutos, ya que la vida útil de la EEPROM está limitada por el número de procesos de escritura. Tras una recuperación de la tensión de alimentación se pueden guardar nuevos parámetros de inmediato en la EEPROM. La selección de los parámetros determina si la reparametrización del módulo regulador con la instrucción PID_FM_455 será con o sin discontinuidad.

Tiempos de ejecución de PID_FM_455

Condiciones marco			Tiempo de ejecución en CPU 414-2DP
READ_VAR	LOAD_OP	LOAD_PAR	
FALSE	FALSE	FALSE	0,077 ms
TRUE	FALSE	FALSE	2,36 ms
TRUE*)	TRUE	FALSE	4,48 ms
FALSE	FALSE	TRUE	2,59 ms
TRUE	FALSE	TRUE	5,15 ms
TRUE*)	TRUE	TRUE	7,1 ms
*) Si LOAD_OP = TRUE, PID_FM_455 también pone READ_VAR en TRUE.			

Parámetros de entrada de PID_FM_455

Tabla 9-255

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	1	En la entrada "Número de canal" se parametriza el número del canal de regulador al que se refiere el DB de instancia. Se admiten los valores 1 a 16.
PHASE	4.0	INT	0	El parámetro PHASE puede interconectarse con el parámetro de salida PHASE de las instrucciones TUN_EC o TUN_ES (PID Self-Tuner). En el cuadro de diálogo "Puesta en servicio" se puede visualizar el estado de fase del PID Self-Tuner en texto compresible. Este parámetro es irrelevante para el OP.

Parámetros de salida de PID_FM_455

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	6.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
out_par	8.0	WORD	W#16#3130	El parámetro out_par no puede ser modificado por el usuario. Identifica el inicio de los parámetros de salida que lee el módulo cuando se activa READ_VAR = TRUE.
SP	10.0	REAL	0.0	En la salida "Consigna" se indica la consigna que actúa efectivamente.
PV	14.0	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	18.0	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.
DISV	22.0	REAL	0.0	En la salida "Variable perturbadora" se indica la magnitud perturbadora que actúa efectivamente. Se admiten valores entre -100 y 100 %.
LMN	26.0	REAL	0.0	En la salida "Variable manipulada" se indica la variable manipulada que actúa efectivamente. En el regulador paso a paso sin realimentación de posición analógica, en el parámetro LMN se indica la acción P + D ilimitada. Se admiten valores entre -100 y 100 %.
LMN_A	30.0	REAL	0.0	En la salida "Variable manipulada A de la función de rango partido / realimentación de posición" se indica la variable manipulada A de la función de rango partido en el regulador continuo, y la realimentación de posición en el regulador paso a paso con realimentación de posición analógica. La salida LMN_A sólo puede utilizarse para la indicación aproximada de una magnitud manipulada simulada correspondiente. La variable manipulada LMNRSVAL de la realimentación de posición simulada debe parametrizarse de la forma correspondiente, y es efectivo al activar LMNRS_ON. Se admiten valores entre -100 y 100 %.
LMN_B	34.0	REAL	0.0	Para el regulador continuo, en la salida "Variable manipulada B de la función de rango partido" se indica la variable manipulada B de la función de rango partido. Se admiten valores entre -100 y 100 %.
QH_ALM	38.0	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite superior disparada" se notifica el rebase del límite H_ALM.
QH_WRN	38.1	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite superior disparada" se notifica el rebase del límite H_WRN.
QL_WRN	38.2	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite inferior disparada" se notifica el rebase por defecto del límite L_WRN.
QL_ALM	38.3	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite inferior disparada" se notifica el rebase por defecto del límite L_ALM.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QLMN_HLM	38.4	BOOL	FALSE	La variable manipulada está limitado siempre por un límite superior y uno inferior. En la salida "Límite superior de la variable manipulado alcanzado" se notifica el rebase del límite superior. (No en el regulador paso a paso sin realimentación de posición analógica)
QLMN_LLM	38.5	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la variable manipulada alcanzado" indica el rebase por defecto del límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica)
QPARA_F	38.6	BOOL	FALSE	El módulo comprueba la validez de los parámetros. Un error de parametrización se indica en la salida "Error de parametrización". Este error de parametrización también se puede ver en el cuadro de diálogo "Online y diagnóstico" del módulo.
QCH_F	38.7	BOOL	FALSE	La salida "Error de canal" se activa cuando el canal de regulación no puede arrojar resultados válidos. El error de canal (p. ej. rotura de hilo) también se activa con QPARA_F = 1 o QMOD_F = 1. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.
QUPRLM	39.0	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación positiva de rampa de subida de la consigna" está activada, se limita la rampa de subida de la consigna.
QDNRLM	39.1	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación negativa de rampa de subida de la consigna" está activada, se limita la rampa de bajada de la consigna.
QSP_HLM	39.2	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de consigna alcanzado" se notifica el rebase del límite superior.
QSP_LLM	39.3	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la consigna alcanzado" indica el rebase del límite inferior.
QLMNUP	39.4	BOOL	FALSE	Se trata de la salida "Señal alta de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QLMNDN	39.5	BOOL	FALSE	Se trata de la salida "Señal baja de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QID	39.6	BOOL	FALSE	QID = TRUE indica que hay una identificación en curso (no que esté activada). Una vez completada la identificación, el resultado puede verse a través del parámetro IDSTATUS de la instrucción CH_DIAG_455.
QSPOPON	40.0	BOOL	FALSE	En la salida "Control manual de consigna activado" se indica si la consigna se controla mediante la herramienta de configuración. Si el bit está activado, el valor SP_OP se aplica como consigna.
QLMNSAFE	40.1	BOOL	FALSE	Si la salida "Modo de seguridad" está activada, como variable manipulada se emite la variable manipulada de seguridad.
QLMNOPON	40.2	BOOL	FALSE	En la salida "Control manual de variable manipulada activado" se indica si la variable manipulada se controla desde la herramienta de configuración. Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
QLMNTRK	40.3	BOOL	FALSE	La salida "Modo de seguimiento" indica si la variable manipulada se corrige mediante una entrada analógica.
QLMN_RE	40.4	BOOL	FALSE	La salida "Manual = 1; Automático = 0" indica si la variable manipulada adopta la variable manipulada externa LMN_RE (Manual = 1) o no.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QLMNR_HS	40.5	BOOL	FALSE	La salida "Señal de tope superior de la realimentación de posición" indica si la servoválvula se encuentra en el tope superior. QLMNR_HS = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
QLMNR_LS	40.6	BOOL	FALSE	La salida "Señal de tope inferior de la realimentación de posición" indica si la servoválvula se encuentra en el tope inferior. QLMNR_LS = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
QLMNR_ON	40.7	BOOL	FALSE	La salida "Realimentación de posición activada" indica que está activado el modo de operación "Regulador paso a paso con realimentación de posición" o "Regulador paso a paso sin realimentación de posición".
QFUZZY	41.0	BOOL	FALSE	Si el parámetro de salida QFUZZY = 1, el regulador funciona con el algoritmo difuso.
QSPLEPV	41.1	BOOL	FALSE	La salida "Indicación del regulador difuso: consigna < valor real" se activa con el regulador difuso si la consigna es menor que el valor real efectivo.
QSPR	41.2	BOOL	FALSE	Si la salida "Modo de rango partido" está activada, el regulador continuo trabaja en modo de rango partido.
QMAN_FC	41.4	BOOL	FALSE	La salida "QMAN_FC" se activa en los siguientes casos: • El regulador secundario está en modo manual y el regulador piloto se rige por el valor real del secundario. • La acción I del regulador piloto se detiene porque la consigna o la magnitud manipulada del regulador secundario están en el límite o porque el regulador secundario se encuentra en modo manual.
QPARABUB	41.7	BOOL	FALSE	El FM activa este parámetro cuando se han cambiado parámetros de operación desde un OP. Si READ_VAR = TRUE y el FM activa esta indicación, la instrucción PID_FM_455 lee los parámetros SP_OP_ON, LMNOP_ON, SP_OP y LMN_OP del FM y los deposita en el DB de instancia. La instrucción adopta así el estado de operación del FM. Tras el proceso de lectura, el parámetro adopta el estado FALSE.
QMOD_F	42.0	BOOL	FALSE	La instrucción comprueba si los procesos de escritura y lectura de un registro han finalizado correctamente. Si se detectan errores, se activa la salida "Error de módulo". La causa puede ser: Una dirección de módulo incorrecta en el parámetro MOD_ADDR, un número de canal incorrecto en el parámetro CHANNEL o un módulo defectuoso.

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

Parámetros de entrada/salida de PID_FM_455

Tabla 9-256

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
[1] Valores predeterminados del módulo tras la primera llamada del PID_FM_455 con COM_RST = TRUE				
COM_RST	44.0	BOOL	FALSE	Si el parámetro es COM_RST = TRUE, la instrucción PID_FM_455 realiza una marcha de inicialización. En ella se leen del FM los parámetros del regulador (que son todos los parámetros posteriores a cont_par) y se depositan en el DB de instancia. Además se comprueba la validez de los parámetros MOD_ADDR y CHANNEL. Tras la marcha de inicialización, el parámetro adopta el estado FALSE.
LOAD_OP	44.1	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros operativos en el FM 455" está activado, los parámetros operativos se cargan en el módulo y el parámetro de entrada/salida se desactiva.
READ_VAR	44.2	BOOL	FALSE	Si el parámetro de entrada/salida "Leer variables del FM 455" está activado, los parámetros de salida se leen del módulo y el parámetro de entrada/salida se desactiva.
LOAD_PAR	44.3	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros de regulación en el FM 455" está activado, los parámetros de regulación se cargan en el módulo y el parámetro de entrada/salida se desactiva.
op_par	46.0	WORD	W#16#3130 [1]	El parámetro op_par no puede ser modificado por el usuario. Identifica el inicio de los parámetros de operación, que se transfieren al módulo si se activa LOAD_OP = TRUE. El final de los parámetros de operación se identifica con cont_par.
SP_RE	48.0	REAL	0.0	En la entrada "Consigna externa" se conecta una consigna externa con el regulador.
LMN_RE	52.0	REAL	0.0	En la entrada "Variable manipulada externa" se conecta una variable manipulada externa con el regulador. Se admiten valores entre -100 y 100 %.
SP_OP_ON	56.0	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Activar control manual de consigna". Si el bit está activado, el valor SP_OP se aplica como consigna.
SAFE_ON	56.1	BOOL	FALSE	Si la entrada "Adoptar posición de seguridad" está activada, se aplica un valor de seguridad como variable manipulada. Nota: El procesamiento de la señal de salida con LMNDN_OP, LMNUP_OP y LMNSOPON en el regulador paso a paso tiene mayor prioridad que la variable manipulada de seguridad.
LMNOP_ON	56.2	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Activar control manual de variable manipulada". Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
LMNTRKON	56.3	BOOL	FALSE	Si la entrada "Corregir (LMN a través de EA)" está activada, la variable manipulada se rige por una entrada analógica (EA). (No en el regulador paso a paso sin realimentación de posición analógica)
LMN_REON	56.4	BOOL	FALSE	Si la entrada "Aplicar variable manipulada externa" está activada, se aplica la variable manipulada externa LMN_RE como variable manipulada.
LMNRHSRE	56.5	BOOL	FALSE	La señal "Servoválvula en el tope superior" se interconecta a la entrada "Señal de tope superior de la realimentación de posición". LMNRHSRE = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMNRLSRE	56.6	BOOL	FALSE	La señal "Servoválvula en el tope inferior" se aplica en la entrada "Señal de tope inferior de la realimentación de posición". LMNRLSRE = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
LMNSOPON	56.7	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Activar control manual de señal de variable manipulada". Si LMNSOPON está activado, las señales LMNUP_OP y LMNDN_OP se aplican como señales de variable manipulada. (Sólo en regulador paso a paso)
LMNUP_OP	57.0	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Subir señal de variable manipulada". Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Subir señal de variable manipulada". (Sólo en regulador paso a paso)
LMNDN_OP	57.1	BOOL	FALSE	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Bajar señal de variable manipulada". Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Bajar señal de variable manipulada". (Sólo en regulador paso a paso)
LMNRS_ON	57.3	BOOL	FALSE	Si no se dispone de una realimentación de posición, puede simularse. La función "Activar simulación de la realimentación de posición" se activa en la entrada. También las instrucciones del PID Self-Tuner tienen acceso a este parámetro, ya que para la optimización se requiere como mínimo una variable manipulada simulada si se ha parametrizado un regulador paso a paso sin realimentación de posición. El valor simulado se muestra en el parámetro LMN_A. Al activar la simulación, el valor del parámetro LMNRSVAL se activa como valor inicial. ATENCIÓN: Con el paso del tiempo, la simulación difiere cada vez más de la realimentación de posición real. (Sólo en el regulador paso a paso sin realimentación de posición analógica)
FUZID_ON	57.4	BOOL	FALSE	En la entrada "Activar identificación de difuso" se activa la identificación del algoritmo difuso.
SP_OP	58.0	REAL	0.0	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Control manual de consigna". Si el bit SP_OP_ON está activado, se aplica el valor "Control manual de consigna" como consigna.
LMN_OP[1]	62.0	REAL	0.0	Las instrucciones del PID Self-Tuner tienen acceso al parámetro de entrada/salida "Control manual de variable manipulada". Si el bit LMNOP_ON está activado, el valor "Control manual de variable manipulada" se aplica como variable manipulada. Se admiten valores entre -100 y 100 %.
LMNRSVAL	66.0	REAL	0.0	Las instrucciones del PID Self-Tuner tienen acceso a la entrada "Valor inicial de la realimentación de posición simulada". En ese parámetro se indica el valor inicial de la simulación. (Sólo en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre -100 y 100 %.
cont_par	70.0	WORD	W#16#3130 [1]	El parámetro cont_par no puede ser modificado por el usuario. Identifica el inicio de los parámetros de regulador, que se leen del FM y se depositan en el DB de instancia si COM_RST = TRUE, y que se transmiten al FM si LOAD_PAR = TRUE se activa. El final de los parámetros de regulador es el final del DB de instancia.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
P_SEL	72.0	BOOL	TRUE [1)]	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción proporcional está conectada si la entrada "Activar acción P" está activada.
PFDB_SEL	72.1	BOOL	FALSE [1)]	En el algoritmo PID las acciones P y D se pueden incluir en el lazo de realimentación. La acción proporcional se encuentra en el lazo de realimentación si la entrada "Incluir acción P en lazo de realimentación" está activada.
MONERSEL	72.2	BOOL	FALSE [1)]	El regulador dispone de un avisador de valor límite, que puede utilizarse para el valor real o para el error de regulación. Si la entrada "Vigilancia: Valor real = 0, Error de regulación = 1" está activada, se vigila el error de regulación.
SDB_SEL	72.4	BOOL	FALSE	SDB_SEL=TRUE: los parámetros de regulador de la instrucción PID_FM_455 no se sobrescriben en el FMx55 con los parámetros de SDB tras una transición de STOP a RUN de la CPU.
D_EL_SEL	74.0	INT	0 [1)]	El componente D del algoritmo PID puede depositarse en una entrada separada. Ésta se selecciona en la entrada "Entrada para componente D". • 0: Error de regulación • 1 a 16: Entrada analógica 1 a 16 • 17: Valor real negativo, acción D en el lazo de realimentación
SP_HLM	76.0	REAL	100.0 [1)]	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior consigna" indica el límite superior. SP_HLM > SP_LLM
SP_LLM	80.0	REAL	0.0 [1)]	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior consigna" indica el límite inferior. SP_LLM < SP_HLM
H_ALM	84.0	REAL	100.0 [1)]	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más alto. H_ALM > H_WRN
H_WRN	88.0	REAL	90.0 [1)]	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite superior" indica el segundo límite más alto. Se admiten valores entre H_ALM y L_WRN
L_WRN	92.0	REAL	10.0 [1)]	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite inferior" indica el segundo límite más bajo. Se admiten valores entre H_WRN y L_ALM
L_ALM	96.0	REAL	0.0 [1)]	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más bajo. L_ALM < L_WRN
HYS	100.0	REAL	1.0 [1)]	Para evitar oscilaciones de las indicaciones de vigilancia, puede parametrizarse una histéresis en la entrada "Histéresis". HYS ≥ 0.0
DEADB_W	104.0	REAL	0.0 [1)]	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. DEADB_W ≥ 0.0
GAIN	108.0	REAL	1.0 [1)]	La entrada "Ganancia proporcional" indica la ganancia del regulador.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
TI	112.0	REAL	3000.0[1)]	La entrada "Tiempo de integración" determina el comportamiento temporal del integrador. Con TI = 0 el integrador está desactivado. TI = 0.0 o bien TI ≥ 0.5
TD	116.0	REAL	0.0[1)]	La entrada "Tiempo de acción derivada" determina el comportamiento temporal del diferenciador. Con TD = 0 el diferenciador está desactivado. TD = 0.0 o bien TD ≥ 1.0
TM_LAG	120.0	REAL	5.0[1)]	El algoritmo de la acción D contiene un retardo que se puede parametrizar en la entrada "Tiempo de retardo de la acción D". TM_LAG ≥ 0.5
LMN_SAFE	124.0	REAL	0.0[1)]	Para la variable manipulada puede parametrizarse un valor de seguridad en la entrada "Variable manipulada de seguridad". Se admiten valores entre -100 y 100 %.
LMN_HLM	128.0	REAL	100.0[1)]	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior de la variable manipulada" indica el límite superior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre LMN_LLM y 100 %.
LMN_LLM	132.0	REAL	0.0[1)]	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior de la variable manipulada" indica el límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre -100 y LMN_HLM.
MTR_TM	136.0	REAL	60.0[1)]	En el parámetro "Tiempo de posicionamiento del motor" se introduce el tiempo de operación de la servoválvula de un tope al otro. (Sólo en regulador paso a paso)
PULSE_TM	140.0	REAL	0.2[1)]	En el parámetro "Duración mínima de impulso" se puede parametrizar una duración de impulso mínima. (Sólo en regulador paso a paso o de impulsos)
BREAK_TM	144.0	REAL	0.2[1)]	En el parámetro "Duración mínima de pausa" se puede parametrizar una duración de pausa mínima. (Sólo en regulador paso a paso o de impulsos)

Nota

Con LOAD_PAR = TRUE todos los parámetros de regulación se cargan de forma permanente en la EEPROM del FM 455.

Con LOAD_OP = TRUE, sólo la consigna SP_RE se carga permanentemente por medio de los parámetros operativos en la EEPROM del FM 455. A todos los parámetros operativos restantes se les asignan los valores 0 ó FALSE en el arranque del FM 455.

La EEPROM del módulo podría quedar destruida por un número excesivo de procesos de escritura. Para evitarlo, tras la escritura de la EEPROM el módulo retrasa una nueva escritura 30 minutos.

Diagramas de bloques de PID_FM_455

Sinopsis

En las siguientes figuras se representan los puntos en los que son efectivos los parámetros de la instrucción PID_FM_455.

Los parámetros son efectivos en los mismos puntos en los reguladores de tres componentes, de relación, mixtos, de valor fijo o cascada. Esto también rige para los parámetros que están presentes por igual en el regulador continuo, en el regulador con salida de impulsos y en el regulador paso a paso. Por ello no se representan todos los diagramas de bloques, y no todos los parámetros están incluidos en todos los diagramas.

Sin embargo, todos los parámetros de la instrucción PID_FM_455 están contenidos en los diagramas, excepto los parámetros MOD_ADDR, CHANNEL, QMOD_F, QPARA_F, QCH_F, QLMNR_ON, RET_VALU, COM_RST, LOAD_PAR, READ_VAR, LOAD_OP.

Efecto de los parámetros de entrada

Cálculo del error de regulación en el regulador de valor fijo o cascada

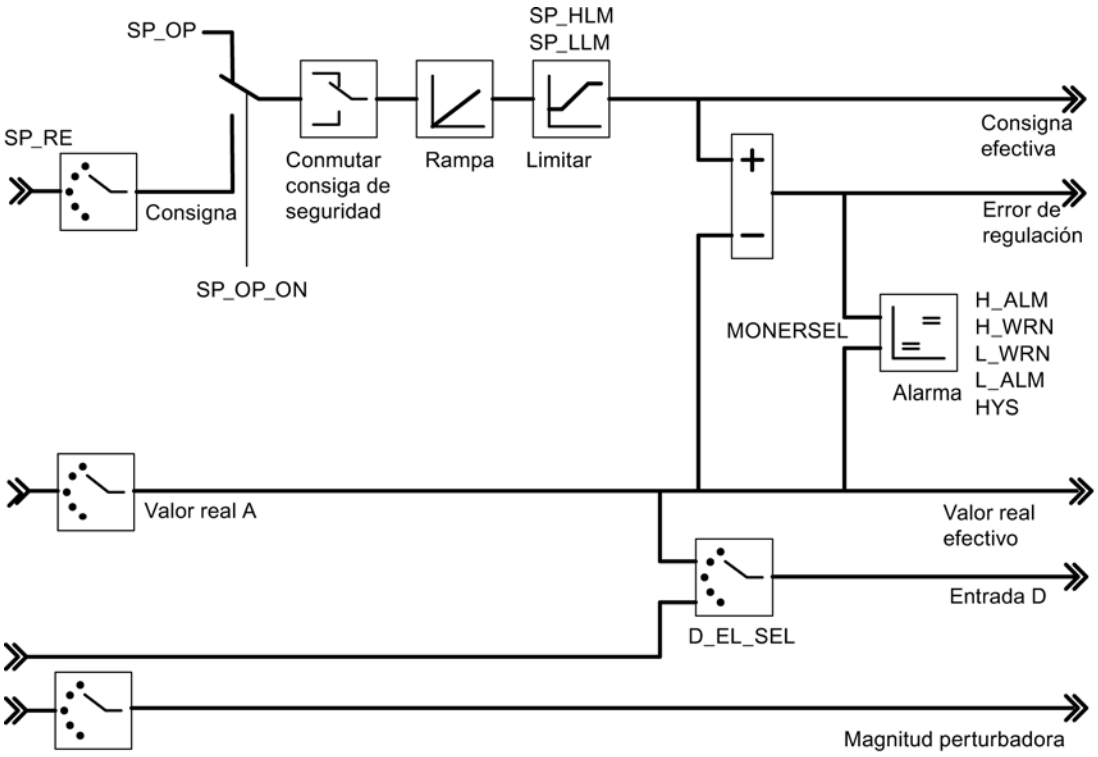
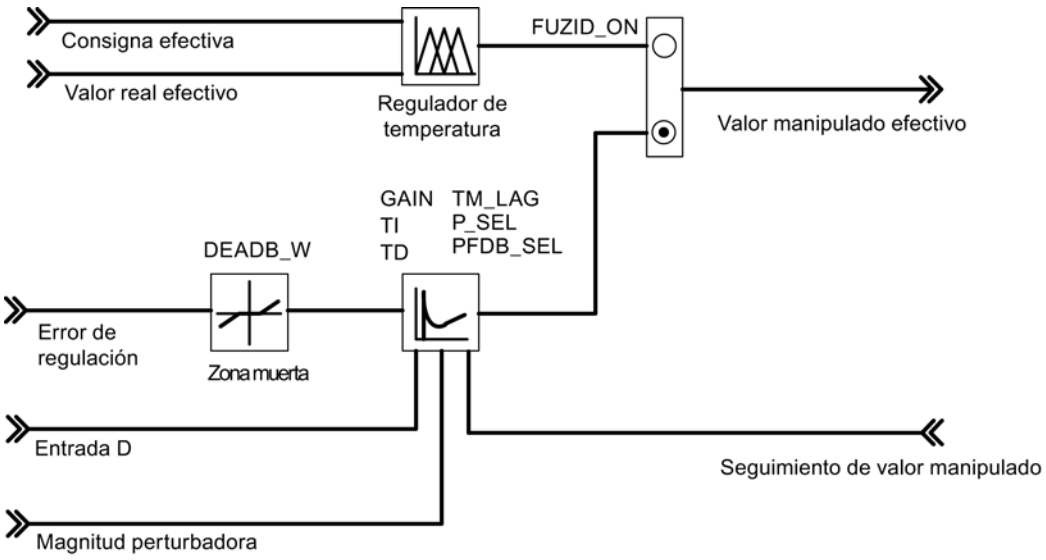
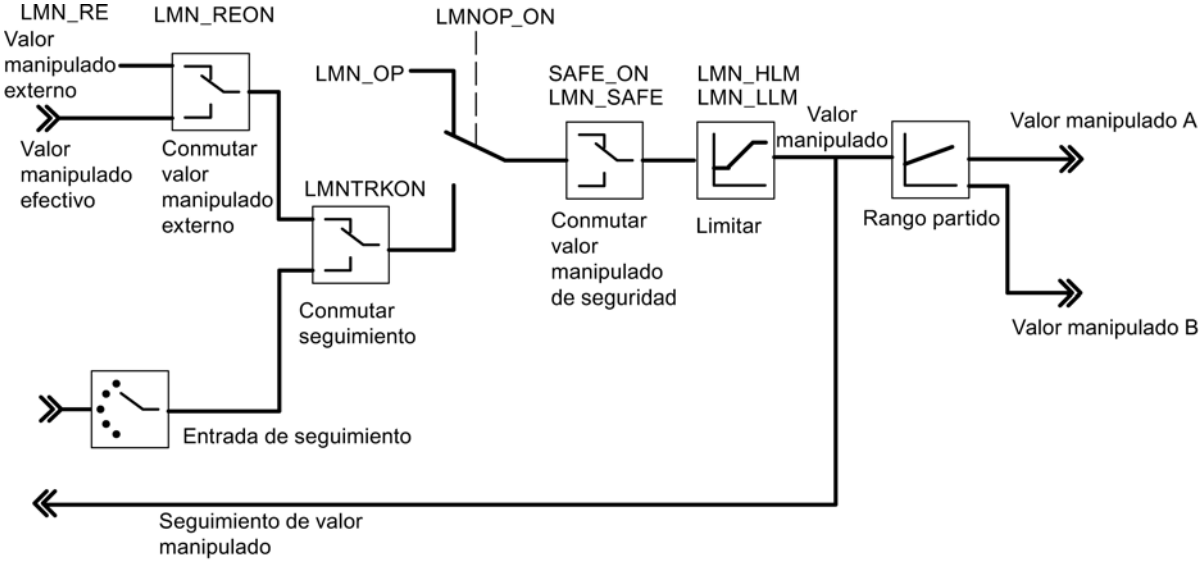


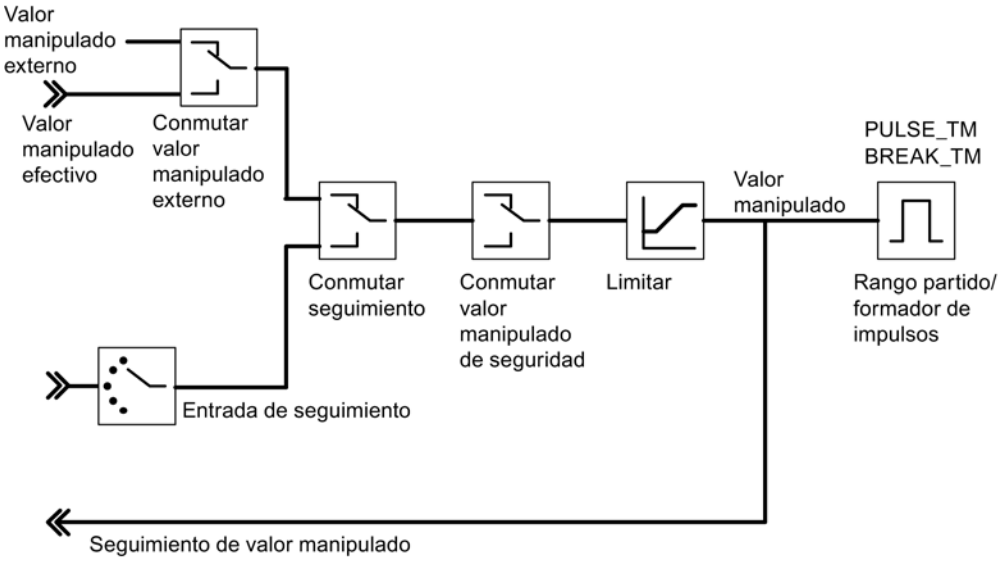
Diagrama de bloques del algoritmo de regulación



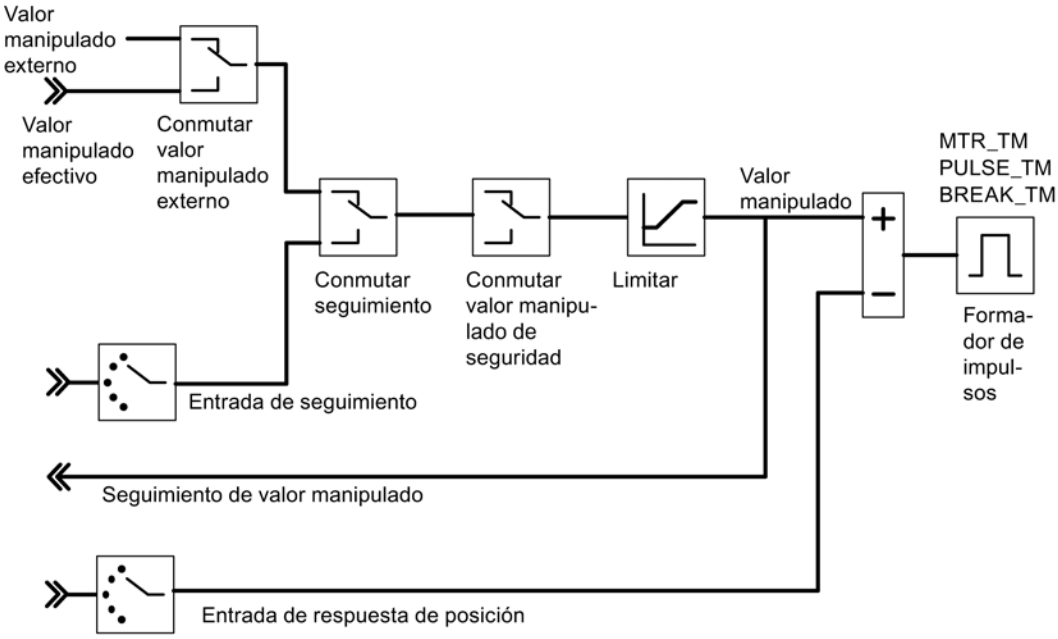
salida del regulador K



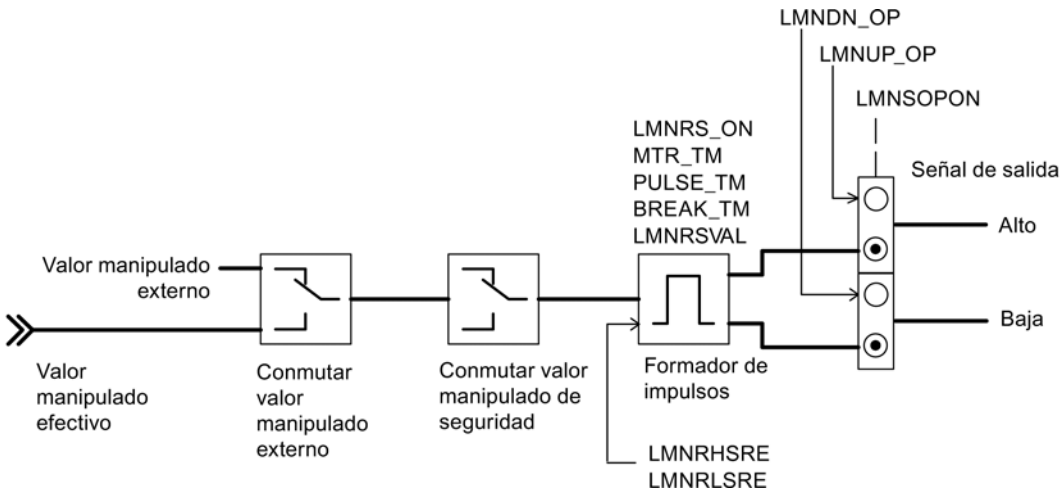
salida del regulador S (modo de operación Regulador de impulsos)



salida del regulador S (modo de operación Regulador paso a paso con realimentación de posición)



salida del regulador S (modo de operación Regulador paso a paso sin realimentación de posición)



Generación de los parámetros de salida

En las siguientes figuras se representan los puntos del módulo en los que se generan los parámetros de salida de la instrucción PID_FM_455.

Cálculo del error de regulación en el regulador de valor fijo o cascada

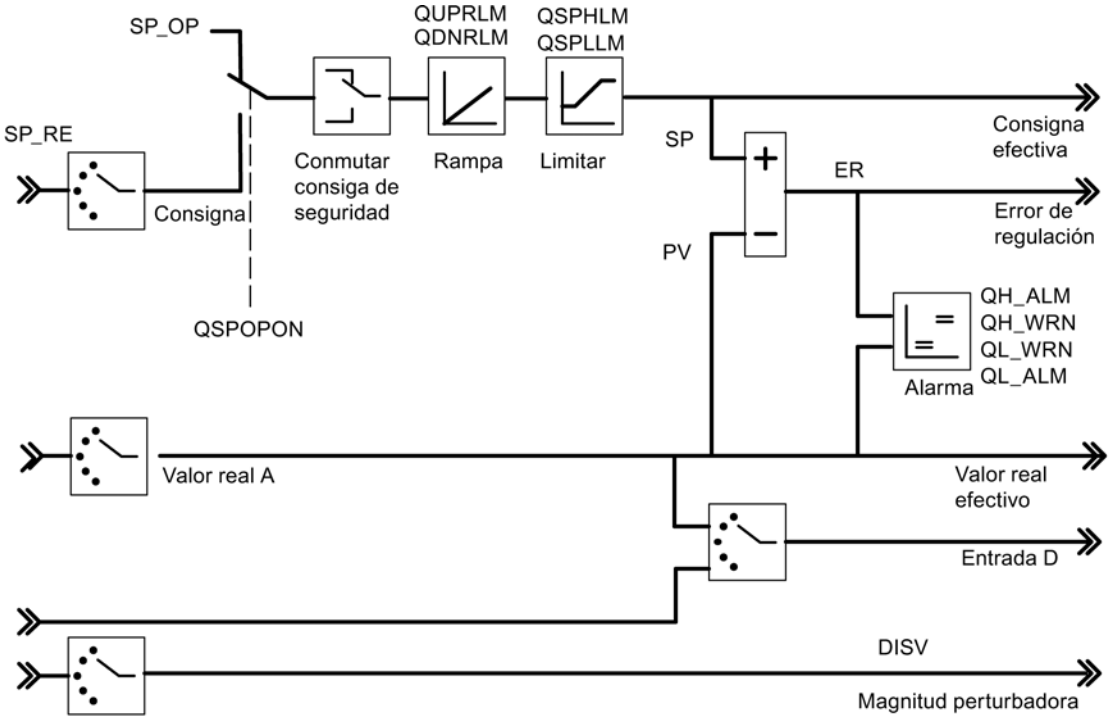
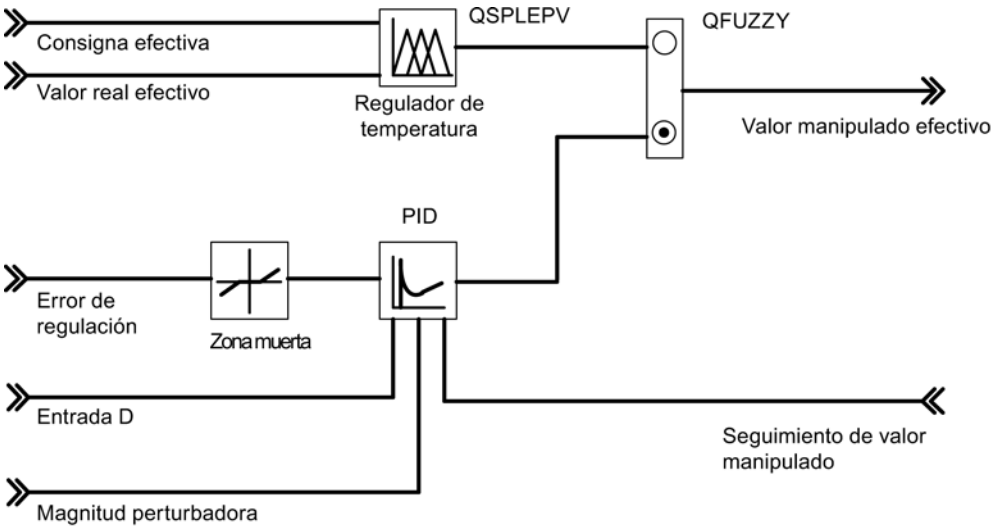
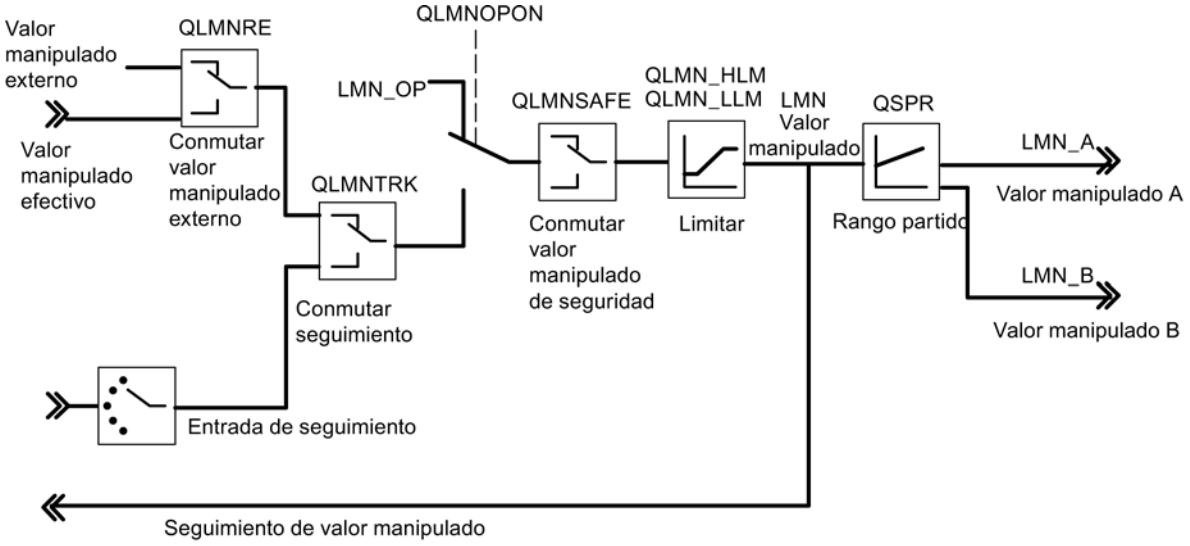


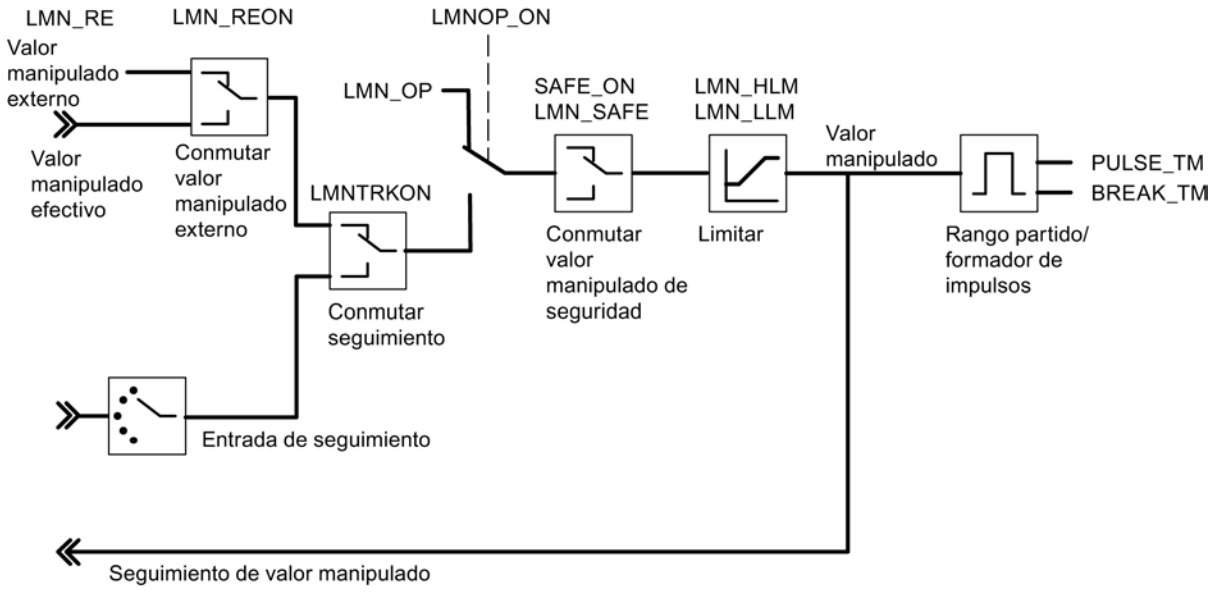
Diagrama de bloques del algoritmo de regulación



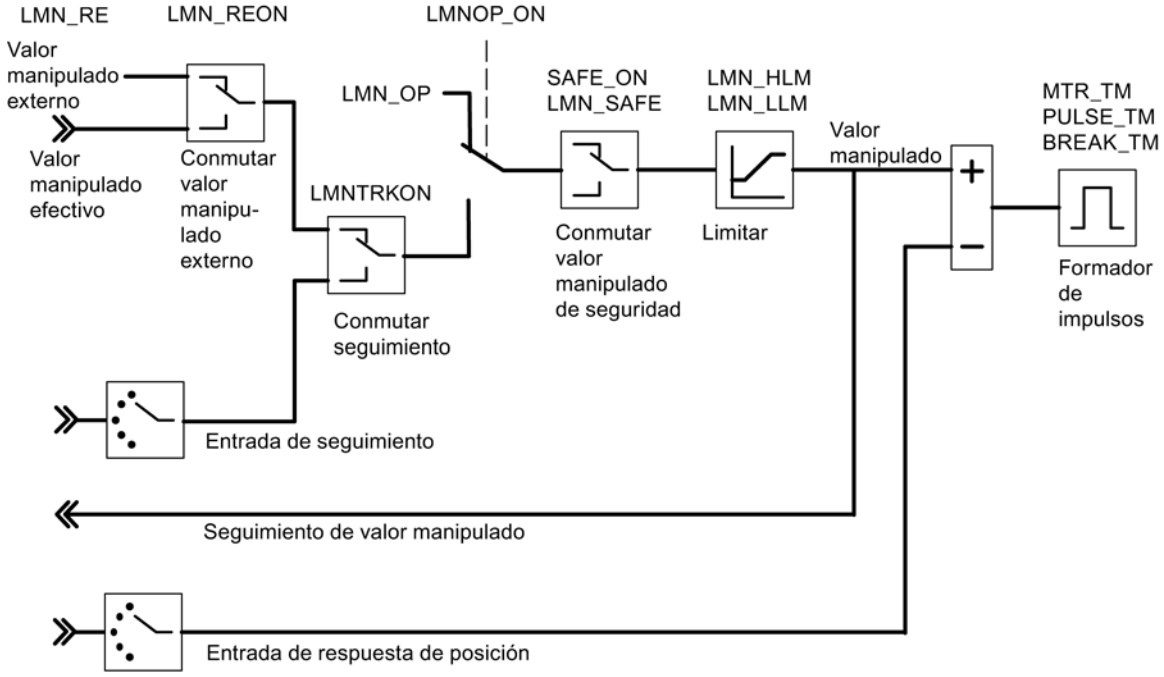
salida del regulador K



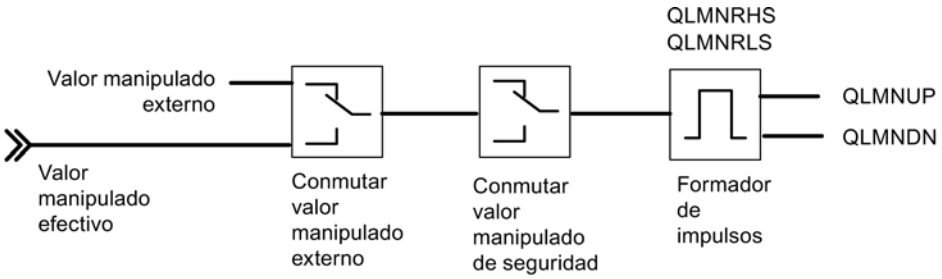
salida del regulador S (modo de operación Regulador de impulsos)



salida del regulador S (modo de operación Regulador paso a paso con realimentación de posición)



salida del regulador S (modo de operación Regulador paso a paso sin realimentación de posición)



FUZ_455

Descripción de FUZ_455

La instrucción FUZ_455 permite leer y escribir parámetros de regulador del regulador de temperatura difuso del FM 455.

Esta función es idónea para las siguientes aplicaciones:

- Transmisión al FM 455 de los parámetros de regulador determinados por identificación tras un cambio de módulo
- Adaptación del FM 455 a diferentes procesos de regulación

Nota

Los parámetros determinados por el FM 455 mediante una identificación no se pueden cambiar, ya que están optimizados para el sistema regulado.

Funcionamiento

Con los parámetros READ_PAR y LOAD_PAR se controla el funcionamiento de la instrucción FUZ_455.

Si READ_PAR = TRUE, FUZ_455 lee los parámetros de todos los reguladores de temperatura del FM 455 y los deposita en el DB de instancia. Una vez que la lectura de los parámetros del regulador de temperatura se ha completado correctamente, FUZ_455 pone el parámetro READ_PAR en FALSE.

Si LOAD_PAR = TRUE, FUZ_455 escribe los parámetros de todos los reguladores de temperatura del FM 455 del DB de instancia en el FM 455. Una vez que la transferencia de los parámetros se ha completado correctamente, FUZ_455 pone el parámetro LOAD_PAR en FALSE.

Llamada

FUZ_455 no requiere una marcha de inicialización.

FUZ_455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Arranque

En el arranque de la CPU se debe activar el parámetro LOAD_PAR de la instrucción FUZ_455, y a continuación llamar el bloque en el programa cíclico mientras LOAD_PAR = TRUE.

Parámetros de entrada de FUZ_455

Tabla 9-257

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	512	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de FUZ_455

Tabla 9-258

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	2.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
PARAFFUZ	4.0	WORD	0	En el parámetro PARAFFUZ, un error de parametrización generado por la instrucción FUZ_455 se indica de la siguiente manera: Byte High de PARAFFUZ = 01: Se ha producido un error de parametrización. Byte High de PARAFFUZ = 00: No se ha producido ningún error de parametrización. En el byte Low se indica el offset del parámetro que ha causado el error de parametrización, contado a partir de la variable estática FUZ_PAR[1]. PARAFFUZ = W#16#0104 significa p. ej. que el segundo parámetro es erróneo

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

Parámetros de entrada/salida de FUZ_455

Tabla 9-259

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
READ_PAR	6.0	BOOL	FALSE	Si el parámetro READ_PAR se activa, se leen los parámetros difusos del módulo y se depositan en variables estáticas del DB de instancia.
LOAD_PAR	6.1	BOOL	FALSE	Si el parámetro LOAD_PAR se activa, se leen los parámetros difusos de las variables estáticas del DB de instancia y se transfieren al módulo.

FORCE455

Descripción de FORCE455

La instrucción FORCE455 permite simular (forzar) los valores de entrada analógicos y digitales para facilitar la puesta en marcha.

Simulación de los valores digitales

La simulación de los valores para las entradas digitales 1 a 16 se activa a través de los conmutadores S_DION[i], donde 1 ≤ i ≤ 16. Los valores de simulación se predeterminan a través de los parámetros DI_SIM[i]. Si S_DION[i] = TRUE, en lugar del valor de la entrada digital i del módulo se utiliza el valor DI_SIM[i]. Los LEDs I1 a I16 muestran siempre el estado de la entrada digital correspondiente, incluso durante la simulación.

Simulación de los valores analógicos

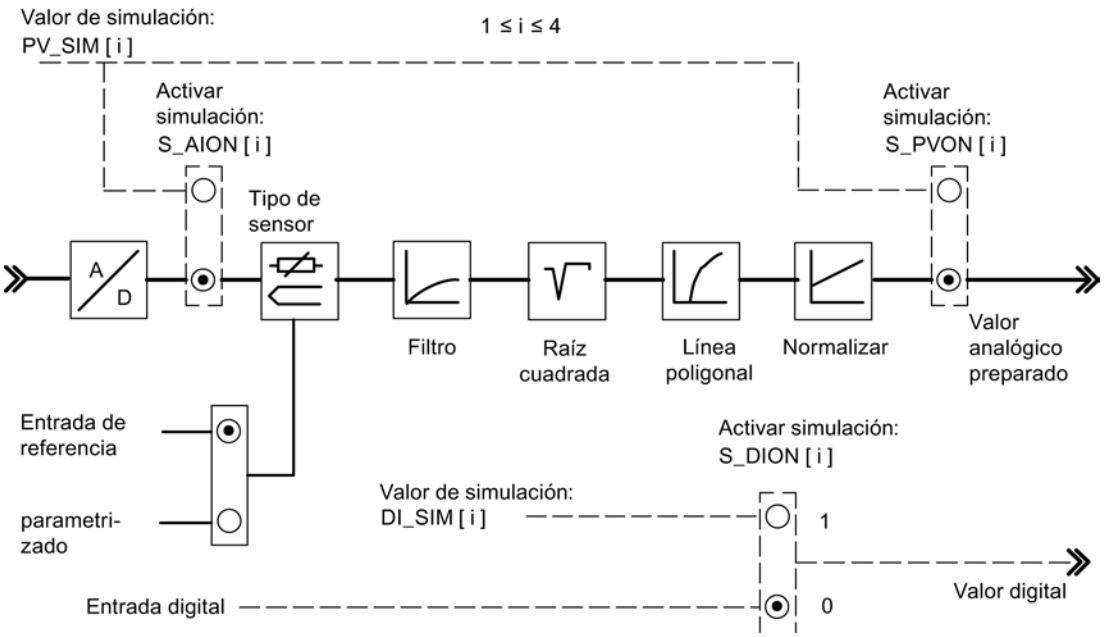
La simulación de los valores analógicos para los canales 1 a 16 se activa a través de los conmutadores S_AION[i] o S_PVON[i], donde 1 ≤ i ≤ 16.

Los valores de simulación para los canales 1 a 16 se predeterminan a través de los parámetros PV_SIM[i].

Los valores de simulación pueden hacerse efectivos en dos puntos:

- S_AION[i] = TRUE (1 ≤ i ≤ 16)
  En lugar del valor de la entrada analógica i del módulo se utiliza el valor PV_SIM[i].
- S_PVON[i] = TRUE (1 ≤ i ≤ 16)
  En lugar del valor acondicionado de la entrada analógica i del módulo se utiliza el valor PV_SIM[i].

En la siguiente figura se representa en qué punto es efectivo el valor analógico simulado.



La conexión y la preasignación de los valores de simulación (forzado) no se realizan a través de la interfaz de parametrización. Por ello, los conmutadores y las líneas de conexión correspondientes se representan rayados.

Llamada

FORCE455 no requiere ninguna fase de inicialización y normalmente se llama cíclicamente.

FORCE455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Comportamiento en arranque

En caso de rearranque del FM 455 tras Power OFF, los conmutadores de simulación del FM 455 vuelven a estar en FALSE.

Parámetros de entrada de FORCE455

Tabla 9-260

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
S_AION	0.0	ARRAY [1..16] of BOOL	FALSE	Si p. ej. el conmutador S_AION[1] adopta el estado TRUE, en lugar del valor de entrada analógica 1 del módulo se utiliza el valor PV_SIM[1].
S_PVON	2.0	ARRAY [1..16] of BOOL	FALSE	Si p. ej. el selector S_PVON[1] adopta el estado TRUE, en lugar del valor acondicionado de entrada analógica 1 del módulo se utiliza el valor PV_SIM[1].
PV_SIM	4.0	ARRAY [1..16] of REAL	0.0	En la entrada PV_SIM[1] se predetermina p. ej., el valor de simulación para la entrada analógica 1. Si S_PVON = TRUE, entonces se especifica aquí el valor de entrada analógica acondicionado. Si S_PVON = FALSE y S_AION = TRUE, entonces se especifica aquí el valor de entrada analógica en mA o mV, que se convierte en un valor acondicionado mediante funciones de acondicionamiento. Se admiten valores entre 0.0 y 20.0 [mA], -1500 a +10000 [mV] o un rango de valores técnico.
S_DION	68.0	ARRAY [1..16] of BOOL	FALSE	Si p. ej. S_DION[1] se pone en TRUE, en lugar de la entrada digital 1 del módulo se utiliza el valor DI_SIM[1] como valor digital.
DI_SIM	70.0	ARRAY [1..16] of BOOL	FALSE	Valor de simulación para entrada digital
MOD_ADDR	72.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de FORCE455

Tabla 9-261

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	74.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

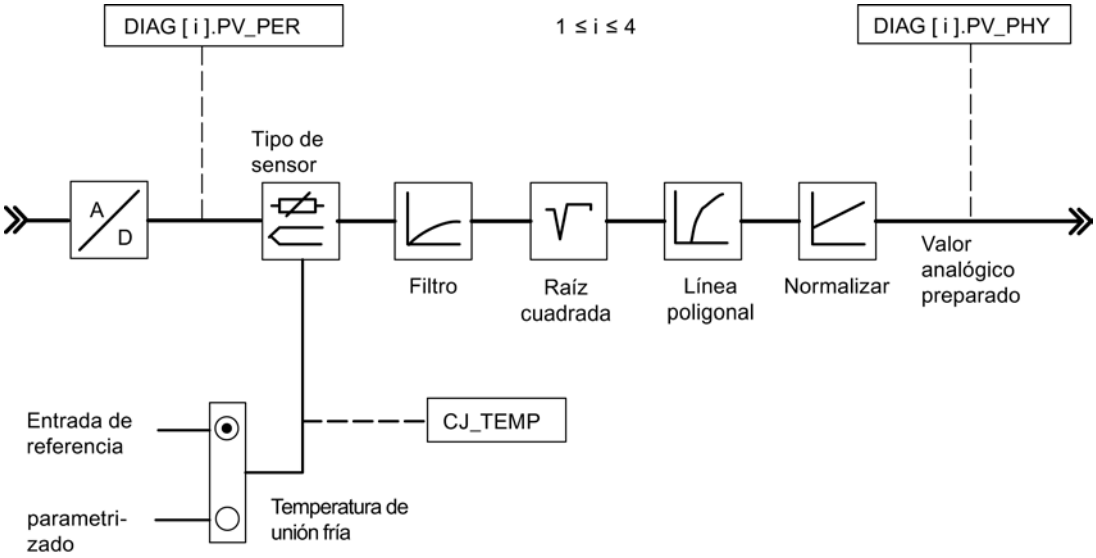
WR_REC: Escribir registro en periferia (Página 6774)

READ_455

Descripción de READ_455

La instrucción READ_455 permite leer los valores de entrada digital y analógicos para ayudar en la puesta en marcha.

Se visualizan los valores siguientes.



Llamada

READ_455 no requiere una marcha de inicialización y normalmente se llama cíclicamente.

READ_455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Parámetros de entrada de READ_455

Tabla 9-262

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	512	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.

Parámetros de salida de READ_455

Tabla 9-263

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
CJ_TEMP	2.0	REAL	0.0	En la salida CJ_TEMP se muestra la temperatura de unión fría medida por el módulo si se ha parametrizado una entrada de termopar y si la temperatura de unión fría no se predetermina de forma parametrizada.
STAT_DI	6.0	ARRAY [1..16] of BOOL	FALSE	En los parámetros STAT_DI se muestran los estados de las entradas digitales 1 a 16.
DIAG[x].PV_PER	(Número de canal) x 8	ARRAY [1..16] of STRUCT	0.0	En el parámetro DIAG[1].PV_PER se muestra p. ej. el valor de entrada analógica del módulo en la unidad mA o mV.
DIAG[x].PV_PHY	(Número de canal) x 8 + 4	ARRAY [1..16] of STRUCT	0.0	En el parámetro DIAG[1].PV_PHY se muestra p. ej. el valor acondicionado de entrada analógica del módulo en unidad física.
RET_VALU	136.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

CH_DIAG_455

Descripción de CH_DIAG_455

La instrucción CH_DIAG_455 permite leer del módulo variables de diagnóstico adicionales específicas de canal.

Llamada

CH_DIAG_455 no requiere una marcha de inicialización y normalmente se llama cíclicamente.

CH_DIAG_455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Parámetros de entrada de CH_DIAG_455

Tabla 9-264

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	2.0	INT	1	En la entrada "Número de canal" se parametriza el número del canal de regulador al que se refiere el DB de instancia. Se admiten los valores 1 a 16.

Parámetros de salida de CH_DIAG_455

Tabla 9-265

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_R	4.0	REAL	0.0	Con el regulador de relación ajustado, el parámetro tiene asignado el valor de entrada de la consigna.
PV_R	8.0	REAL	0.0	El parámetro sólo tiene asignado el siguiente valor con el regulador de relación ajustado: (Valor real A - Consigna Offset) / Valor real D
DIF_I	12.0	REAL	0.0	En el parámetro DIF_I se muestra la magnitud de entrada de la acción D. Esto resulta conveniente p. ej. si como magnitud de entrada de la acción D se ha parametrizado una entrada analógica.
TRACKPER	16.0	REAL	0.0	En el parámetro TRACKPER se muestra la magnitud de entrada con la que se corrige la magnitud manipulada si el regulador está ajustado para seguimiento de magnitud manipulada.
IDSTATUS	20.0	WORD	0.0	En Parámetro IDSTATUS se muestra el estado de la identificación
LMN_P	22.0	REAL	0.0	En el parámetro LMN_P se muestra la acción P de la magnitud manipulada.
LMN_I	26.0	REAL	0.0	En el parámetro LMN_I se muestra la acción I de la magnitud manipulada.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMN_D	30.0	REAL	0.0	En el parámetro LMN_D se muestra la acción D de la magnitud manipulada.
RET_VALU	34.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

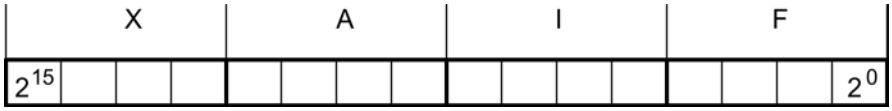
Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

Parámetro IDSTATUS

El parámetro IDSTATUS proporciona información sobre el estado de la identificación y contiene los cuatro valores hexadecimales X, A, I y F.

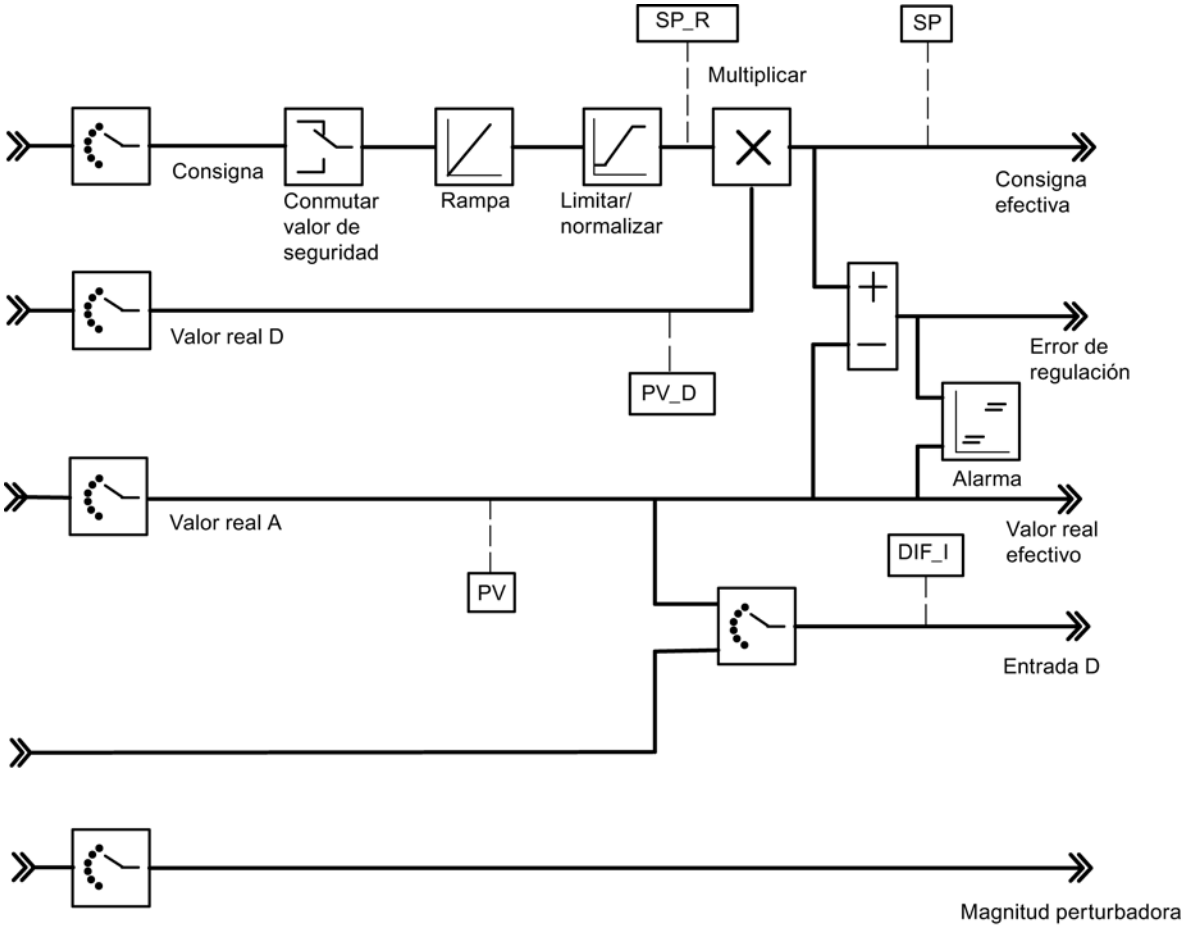


Valor	Descripción
X	Irrelevante (siempre 0)
A	Número de acción • 0 = modo manual (o bien sin modo de regulación); • 2 = regulación; • 4 = optimización activada (FUZID_ON = true); • 6 = estado de transición de modo manual a 2 ó 4;
I	Indicación "Identificación en curso" y "Los parámetros se han determinado pero aún no se han guardado en la EEPROM" • 0 = la identificación no se ejecuta, no se han determinado nuevos parámetros • 1 = la identificación está en curso, no se han determinado nuevos parámetros • 2 = la identificación no se ejecuta, se han determinado nuevos parámetros, pero aún no se han guardado en la EEPROM • 3 = la identificación está en marcha, se han determinado nuevos parámetros, pero aún no se han guardado en la EEPROM
F	Número de error. • 0 = no hay error • 4 = salto excesivo del valor real durante la identificación • 5 = relación entre tiempo de retardo y constante de tiempo de proceso excesiva o comportamiento del proceso regulado muy distante del lineal. • 6 = caída o subida de temperatura excesiva en el inicio de la identificación. El proceso no tiene la calma necesaria

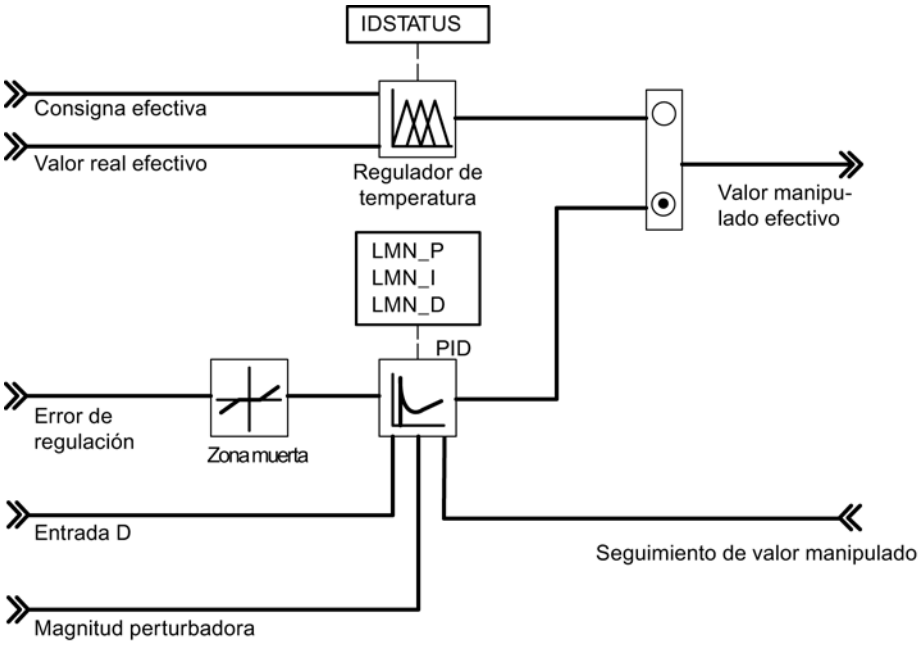
Diagramas de bloques de CH_DIAG_455

Diagramas de bloques de CH_DIAG_455

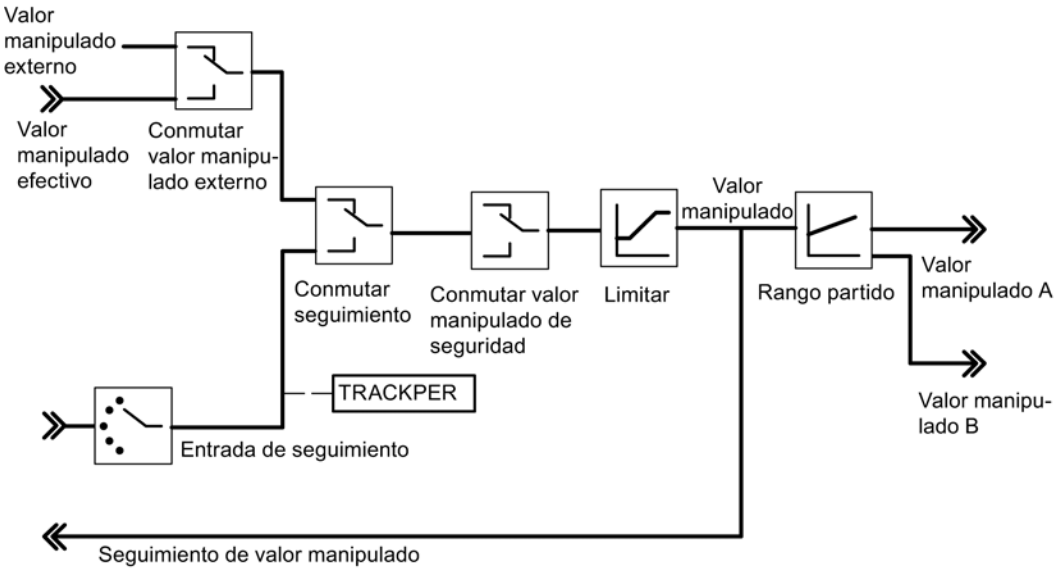
Se muestran los siguientes valores de diagnóstico del error de regulación.



Se muestran los siguientes valores del algoritmo de regulación.



Se muestran los siguientes valores del regulador K o S.



PID_PAR_455

Descripción de PID_PAR_455

La instrucción PID_PAR_455 permite modificar parámetros online que no pueden predeterminarse con la instrucción PID_FM_455. Los parámetros modificables pueden verse en la tabla Parámetros modificables.

Llamada

PID_PAR_455 requiere una marcha de inicialización, que se inicia activando el parámetro COM_RST = TRUE. Para reducir el tiempo de ejecución, PID_PAR_455 no debería llamarse cíclicamente, sino sólo cuando fuera necesario cambiar parámetros. En ese caso, COM_RST debe ser FALSE.

PID_PAR_455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Funcionamiento

En el curso de la inicialización, PID_PAR_455 lee los parámetros de los datos del sistema y los deposita en variables estáticas. PID_PAR_455 cambia en cada llamada una de estas variables REAL y una de las variables INT. En el parámetro de entrada INDEX_R o INDEX_I indique el número de índice del parámetro que desea modificar. Indique los nuevos valores en el parámetro de entrada VALUE_R o VALUE_I. PID_PAR_455 transfiere el registro completo con las variables modificadas al FM 455.

Para cambiar otros parámetros, llame **el mismo** DB de instancia varias veces seguidas con COM_RST = FALSE y con diferentes números de índice.

El parámetro COM_RST es un parámetro de entrada que no se desactiva con PID_PAR_455.

Nota

Recuerde que los parámetros que ha cambiado con PID_PAR_455 se sobrescriben en el arranque con los parámetros de los datos del sistema.

Arranque

En el arranque de la CPU y en la transición de STOP a RUN de la CPU, los parámetros del FM 455 se sobrescriben con los parámetros del SDB de la CPU.

Ejemplo

Desea cambiar el tiempo de subida de la rampa para la magnitud de referencia durante el funcionamiento, y además utilizar diferentes valores de entrada analógica como valor real independientemente del estado del proceso.

- En el arranque de la CPU llame PID_PAR_455 con COM_RST = TRUE.
- Para parametrizar el tiempo de subida de la rampa para la referencia a 10.0 llame PID_PAR_455 durante el funcionamiento con INDEX_R = 30, VALUE_R = 10.0 y INDEX_I = 0.
- Para parametrizar el valor de entrada analógica 4 del módulo como valor real, llame PID_PAR_455 durante el funcionamiento con INDEX_R = 0, INDEX_I = 50 y VALUE_I = 4.

Parámetros de entrada de PID_PAR_455

Tabla 9-266

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	0.0	BOOL	TRUE	Si el parámetro es COM_RST = TRUE, la instrucción PID_PAR_455 realiza una marcha de inicialización. En ella se leen los parámetros de los datos del sistema de la CPU y se depositan en el DB de instancia.
MOD_ADDR	2.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CHANNEL	4.0	INT	1	En la entrada "Número de canal" se parametriza el número del canal de regulador al que se refiere el DB de instancia. Se admiten los valores 1 a 16.
INDEX_R	6.0	INT	0.0	• 0: No se modifica ningún parámetro REAL • 1 a 48: Cada valor corresponde a un parámetro REAL modificable.
VALUE_R	8.0	REAL	0.0	En función del parámetro en cuestión
INDEX_I	12.0	INT	0.0	• 0: No se modifica ningún parámetro INT • 49 a 61: Cada valor corresponde a un parámetro INT modificable.
VALUE_I	14.0	INT	0.0	En función del parámetro en cuestión

Parámetros de salida de PID_PAR_455

Tabla 9-267

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	16.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
BUSY	18.0	BOOL	FALSE	Si BUSY = TRUE, el módulo aún no ha adoptado los parámetros del módulo. En ese caso, la llamada de la instrucción PID_PAR_455 debería repetirse en el siguiente ciclo.

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

Parámetros modificables

Tabla 9-268 Lista de los parámetros REAL e INT que se pueden modificar con PID_PAR_455 en un FM 455

Tipo de datos	Descripción	Número de índice
-	Ningún parámetro seleccionado	0
REAL	Constante de tiempo de filtro para entrada analógica	1
REAL	Fin de medición (100%)	2
REAL	Inicio de medición (0%)	3
REAL	Línea poligonal, valor de interpolación 1 lado de entrada	4
REAL	Línea poligonal, valor de interpolación 2 lado de entrada	5
REAL	Línea poligonal, valor de interpolación 3 lado de entrada	6
REAL	Línea poligonal, valor de interpolación 4 lado de entrada	7
REAL	Línea poligonal, valor de interpolación 5 lado de entrada	8
REAL	Línea poligonal, valor de interpolación 6 lado de entrada	9
REAL	Línea poligonal, valor de interpolación 7 lado de entrada	10
REAL	Línea poligonal, valor de interpolación 8 lado de entrada	11
REAL	Línea poligonal, valor de interpolación 9 lado de entrada	12
REAL	Línea poligonal, valor de interpolación 10 lado de entrada	13
REAL	Línea poligonal, valor de interpolación 11 lado de entrada	14
REAL	Línea poligonal, valor de interpolación 12 lado de entrada	15
REAL	Línea poligonal, valor de interpolación 13 lado de entrada	16
REAL	Línea poligonal, valor de interpolación 1 lado de salida	17
REAL	Línea poligonal, valor de interpolación 2 lado de salida	18
REAL	Línea poligonal, valor de interpolación 3 lado de salida	19
REAL	Línea poligonal, valor de interpolación 4 lado de salida	20
REAL	Línea poligonal, valor de interpolación 5 lado de salida	21
REAL	Línea poligonal, valor de interpolación 6 lado de salida	22
REAL	Línea poligonal, valor de interpolación 7 lado de salida	23
REAL	Línea poligonal, valor de interpolación 8 lado de salida	24
REAL	Línea poligonal, valor de interpolación 9 lado de salida	25
REAL	Línea poligonal, valor de interpolación 10 lado de salida	26
REAL	Línea poligonal, valor de interpolación 11 lado de salida	27
REAL	Línea poligonal, valor de interpolación 12 lado de salida	28
REAL	Línea poligonal, valor de interpolación 13 lado de salida	29
REAL	Tiempo de subida de la rampa para referencia	30
REAL	Referencia de seguridad o relación de referencia de seguridad	31
REAL	Offset de combinación de consigna (regulador de relación/mixto)	32
REAL	Factor para valor real B (regulador de tres componentes)	33
REAL	Factor para valor real C (regulador de tres componentes)	34
REAL	Offset para combinación de valor real (regulador de tres componentes)	35
REAL	Factor para combinación de magnitud perturbadora	36

Tipo de datos	Descripción		Número de índice
REAL	Punto de operación		37
REAL	Agresividad en el regulador difuso		38
REAL	Vértices de la función de rango partido: Inicio de rango señal de entrada A		39
REAL	Vértices de la función de rango partido: Fin de rango señal de entrada A		40
REAL	Vértices de la función de rango partido: Inicio de rango señal de salida A		41
REAL	Vértices de la función de rango partido: Fin de rango señal de salida A		42
REAL	Vértices de la función de rango partido: Inicio de rango señal de entrada B		43
REAL	Vértices de la función de rango partido: Fin de rango señal de entrada B		44
REAL	Vértices de la función de rango partido: Inicio de rango señal de salida B		45
REAL	Vértices de la función de rango partido: Fin de rango señal de salida B		46
REAL	Duración mínima del impulso		47
REAL	Duración mínima de pausa		48
INT	Selección de la referencia SP o SP_RE para el regulador		49
	0	Consigna SP_RE de la instrucción	
	1 a 16	Valor de entrada analógica 1 a 16	
	17 a 32	Magnitud manipulada (LMN) de regulador 1 a 16	
INT	Selección de la magnitud reguladora principal valor real A para el regulador		50
	0:	Valor real A = 0.0	
	1 a 16:	Valor de entrada analógica 1 a 16	
INT	Selección de la magnitud reguladora auxiliar valor real B para el regulador		51
	0	Valor real B = 0.0	
	1 a 16	Valor de entrada analógica 1 a 16	
INT	Selección de la magnitud reguladora auxiliar valor real C para el regulador		52
	0	Valor real C = 0.0	
	1 a 16	Valor de entrada analógica 1 a 16	
INT	Selección de la magnitud reguladora auxiliar valor real D para el regulador		53
	0	Valor real D = 0.0	
	1 a 16	Valor de entrada analógica 1 a 16	
	17 a 32	Magnitud manipulada (LMN) de regulador 1 a 16	
INT	Selección de la magnitud perturbadora DISV para el regulador		54
	0	Magnitud perturbadora = 0.0	
	1 a 16	Valor de entrada analógica 1 a 16	
INT	Selección del seguimiento de posición TRACK_PER para el regulador		55
	0	Seguimiento de posición = 0.0	
	1 a 16	Valor de entrada analógica 1 a 16	
INT	Selección del seguimiento de posición LMNR_PER para el regulador		56
	0	Seguimiento de posición = 0.0	
	1 a 16	Valor de entrada analógica 1 a 16	
INT	Selección de la señal para la conmutación al valor de seguridad para la variable manipulada del regulador		57
	0	Sólo asignación predeterminada a través del parámetro SAFE_ON de la instrucción PID_FM_455	

Tipo de datos	Descripción		Número de índice
	1 a 16	Asignación predeterminada con el parámetro SAFE_ON de la instrucción PID_FM_455 predeterminado por la entrada digital 1 a 16	
INT	Selección de la señal para la conmutación a la función de seguimiento de la variable manipulada del regulador		58
	0	Sólo asignación predeterminada a través del parámetro LMNTRKON de la instrucción PID_FM_455	
	1 a 16	Asignación predeterminada con el parámetro LMNTRKON de la instrucción PID_FM_455 predeterminado por la entrada digital 1 a 16	
INT	Selección de la señal para la conmutación de la variable manipulada del regulador a LMN_RE		59
	0	Sólo asignación predeterminada a través del parámetro LMN_REON de la instrucción PID_FM_455	
	1 a 16	Asignación predeterminada con el parámetro LMN_REON de la instrucción PID_FM_455 predeterminado por la entrada digital 1 a 16	
INT	Selección de la señal de tope superior de la realimentación de posición		60
	0	Sólo asignación predeterminada a través del parámetro LMNRHSRE de la instrucción PID_FM_455	
	1 a 16	Asignación predeterminada con el parámetro LMNRHSRE de la instrucción PID_FM_455 predeterminado por la entrada digital 1 a 16	
INT	Selección de la señal de tope inferior de la realimentación de posición		61
	0	Sólo asignación predeterminada a través del parámetro LMNRLSRE de la instrucción PID_FM_455	
	1 a 16	Asignación predeterminada con el parámetro LMNRLSRE de la instrucción PID_FM_455 predeterminado por la entrada digital 1 a 16	

CJ_T_PAR_455

Descripción de CJ_T_PAR_455

La instrucción CJ_T_PAR_455 permite cambiar online la temperatura de unión fría parametrizada. Esto es necesario si una regulación de temperatura debe funcionar con varios FM 455 con entradas de termopar sin necesidad de conectar un Pt 100 en cada FM 455.

Llamada

CJ_T_PAR_455 requiere una marcha de inicialización. Para ello debe llamarse una sola vez en el arranque de la CPU con el parámetro COM_RST = TRUE.

Normalmente, CJ_T_PAR_455 se llama cíclicamente. Por motivos relacionados con el tiempo de operación, COM_RST debería tener el valor FALSE. El parámetro COM_RST es un parámetro de entrada que no se desactiva.

CJ_T_PAR_455 debe llamarse en el mismo OB que todas las instrucciones restantes que acceden al mismo FM 455.

Ejemplo

Si en una regulación de extrusionador con más de 16 zonas de calentamiento la temperatura de las uniones frías se mide con un FM 455, puede leerse con READ_455 en el parámetro CJ_TEMP y parametrizarse en los FM 455 restantes con CJ_T_PAR_455.

Parámetros de entrada de CJ_T_PAR_455

Tabla 9-269

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	0.0	BOOL	–	Si el parámetro es COM_RST = TRUE, la instrucción CJ_T_PAR_455 realiza una marcha de inicialización. En ella se leen los parámetros de los datos del sistema de la CPU y se depositan en el DB de instancia.
MOD_ADDR	2.0	INT	256	En esta entrada se encuentra la dirección del módulo resultante de la configuración con STEP 7.
CJ_T	4.0	REAL	0.0	En el parámetro CJ_T se puede predeterminar la temperatura de las uniones frías. El rango de valores permitido varía en función del tipo de sensor.

Parámetros de salida de CJ_T_PAR_455

Tabla 9-270

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
RET_VALU	8.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones WR_REC y RD_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.
BUSY	10.0	BOOL	FALSE	Si BUSY = TRUE, el módulo aún no ha adoptado los parámetros del módulo. En ese caso, la llamada de la instrucción CJ_T_PAR_455 debería repetirse en el siguiente ciclo.

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

LP_ZONE

Descripción de LP_ZONE

Para un grado de utilización uniforme de la CPU y una ejecución equidistante de la instrucción ZONE, en una optimización de regulador es necesario utilizar el distribuidor de llamadas de regulador LP_ZONE.

LP_ZONE garantiza que siempre se ejecute sólo una zona por llamada del OB de alarma cíclica.

La instrucción LP_ZONE calcula los tiempos de muestreo de las zonas y los transmite a las instancias de la instrucción ZONE.

Llamada

LP_ZONE se llama en un OB de arranque o de alarma cíclica. Las instrucciones LP_ZONE, ADM_ZONE y ZONE deben llamarse todas en el mismo nivel de alarma cíclica, preferentemente de forma inmediata en el OB de alarma cíclica. No pueden programarse factores de ciclo de tiempo propios para las instrucciones.

Tras insertar o eliminar una zona o si se cambia el orden de llamada, LP_ZONE requiere una marcha de inicialización. Para ello hay que llamarlo con el parámetro COM_RST = TRUE. El bit COM_RST se transmite a través de la interfaz LP_SCH a las diferentes zonas y al bloque de gestión correspondiente.

Los números de zona NBR_CALL de las instancias de las instrucciones ZONE y ADM_ZONE y el número máximo de zona MNR_CALL de la instrucción LP_ZONE se activan automáticamente según el siguiente mecanismo:

Primera ejecución del OB de alarma cíclica tras la activación de COM_RST de la instrucción LP_ZONE:

- LP_ZONE pone el contador de llamadas LP_SCH. CALL_CNT a 1.
- ZONE y ADM_ZONE leen el contador de llamadas y escriben el valor en su parámetro de entrada/salida NBR_CALL. ZONE aumenta el contador de llamadas en 1. ADM_ZONE no lo modifica.

Segunda ejecución del OB de alarma cíclica tras la activación de COM_RST de la instrucción LP_ZONE:

- LP_ZONE lee el contador de llamadas (número de todas las zonas) y escribe el valor en MNR_CALL.

Funcionamiento

Con cada llamada de la instrucción LP_ZONE el parámetro LP_SCH.CALL_CNT aumenta en 1. Cuando el contador alcanza el número máximo de zonas de MNR_CALL se vuelve a poner a 1. En la llamada de la instrucción ZONE, ésta primero comprueba el contador LP_SCH.CALL_CNT. Si coincide con su propio número de zona NBR_CALL, ejecuta su programa. El bloque vuelve a activar los parámetros MNR_CALL de la instrucción LP_ZONE

y NBR_CALL de la instrucción ADM_ZONE y ZONE automáticamente tras un rearranque completo.

Nota

Interconecte el parámetro de entrada/salida LP_SCH de las instrucciones ZONE y ADM_ZONE en la estructura interna LP_SCH del DB de instancia de la instrucción LP_ZONE.

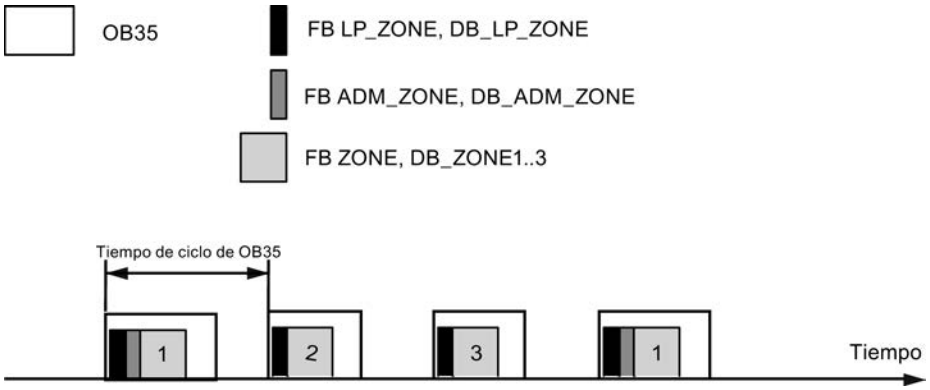
Interconecte el parámetro de entrada OB_EXC_FREQ de la instrucción LP_ZONE con el parámetro temporal OBxx_EXC_FREQ del OB de alarma cíclica.

Ejemplo

El orden de llamada en el OB 33

```
OB33
 ...
 call LP_ZONE, DB_LP_ZONE
 call ADM_ZONE, DB_ADM_ZONE
 call ZONE, DB_ZONE1
 call ZONE, DB_ZONE2
 call ZONE, DB_ZONE3
 …
 BE
```

se traduce en el siguiente desarrollo en el procesamiento de zonas:



Parámetros de entrada de LP_ZONE

Tabla 9-271

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
OB_EXC_FREQ	0.0	INT	0	Tiempo de muestreo [msec]; interconecte el parámetro con las correspondientes variables locales del OB de alarma cíclica.

Parámetro de salida LP_ZONE

Tabla 9-272

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MNR_CALL	2.0	INT	0	Número de zonas. El formato se determina automáticamente tras un rearranque completo.

Parámetros de entrada/salida de LP_ZONE

Tabla 9-273

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
COM_RST	4.0	BOOL	TRUE	Rearranque completo; asegúrese de que este bit está activado en el OB 100.

Variables estáticas de LP_ZONE

Tabla 9-274

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LP_SCH	6.0	STRUCT		Interfaz entre las instrucciones LP_ZONE y ADM_ZONE o bien ZONE

A continuación siguen parámetros internos que no pueden ser utilizados por el usuario.

ADM_ZONE

Descripción de ADM_ZONE

Con frecuencia, en la práctica las zonas de calentamiento y refrigeración están construidas de tal forma que resulta fácil que el calor se transmita de una zona a la otra. En esos casos, estas zonas con acoplamiento térmico deben unificarse en un grupo. La instrucción ADM_ZONE gestiona un grupo de zonas de calentamiento/refrigeración y se encarga de las siguientes funciones:

- Conexión y desconexión de la calefacción de todas las zonas
- Conexión y desconexión de la optimización del regulador
- Indicación de grupo de un error de canal
- Indicación de grupo de un error de optimización de regulador
- Indicación de grupo de una advertencia de optimización de regulador
- Control del salto común de la variable manipulada en optimización del regulador

Llamada

Llame las instrucciones LP_ZONE, ADM_ZONE y ZONE todas en el mismo nivel de alarma cíclica, preferentemente de forma inmediata en el OB de alarma cíclica. No pueden programarse factores de ciclo de tiempo propios para las instrucciones. Llame ADM_ZONE después de LP_ZONE y antes de ZONE.

Arranque

Active el bit de rearranque completo de ADM_ZONE p. ej. en el OB 100 o en el primer ciclo del OB de alarma cíclica. Tras un rearranque completo se desconectan las calefacciones de las diferentes zonas y se interrumpe cualquier optimización de regulador en curso.

Funcionamiento

El programa en sí sólo se ejecuta si el contador de llamadas de la instrucción LP_ZONE tiene el mismo valor que el número de llamadas propio: LP_SCH.CALL_CNT = NBR_CALL.

Conexión y desconexión de la calefacción de todas las zonas

Una modificación del selector HEAT_OFF se transmite como control manual a la interfaz ADMIN, y con ello consecutivamente a las zonas que deben gestionarse. Si no hay control manual, el resultado se transmite de las zonas ADMIN.QHEAT_OFF al parámetro de entrada/salida HEAT_OFF. Sólo si la calefacción está desactivada en todas las zonas, el conmutador de grupo HEAT_OFF adopta el estado TRUE.

Un control manual del selector HEAT_OFF por parte del usuario se indica en el parámetro Done QHO_DONE. El estado predeterminado del parámetro es TRUE. En caso de control manual, adopta el estado FALSE. Si la calefacción está conectada o desconectada en todas las zonas correspondientes, QHO_DONE vuelve a adaptar el valor TRUE.

La calefacción de una zona está desactivada si rige: cont_FM455.QLMN_RE = TRUE y cont_FM455.LMN = 0.0

Conexión y desconexión de la optimización de regulador de todas las zonas

Una modificación del conmutador TUN_ON se transmite como control manual a la interfaz ADMIN, y con ello sucesivamente a las zonas que deben gestionarse. Si no hay control manual, el resultado se transmite de las zonas ADMIN.QTUN_ON al parámetro de entrada/salida TUN_ON. Sólo si la optimización de regulador está desactivada en todas las zonas, el conmutador de grupo TUN_ON vuelve a adoptar el valor FALSE.

Indicación de grupo de un error de canal

El bit QCH_F indica si hay un error de canal del FM 455 presente como mínimo en una zona. Compruebe en qué zonas hay un error de canal presente. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.

Indicación de grupo de un error de optimización de regulador

El bit QTUN_ERR indica si como mínimo en un DB de instancia de la instrucción ZONESTATUS_H = 3xxxx o STATUS_C = 3xxxx. Compruebe las zonas correspondientes y subsane el error de la forma descrita en Parámetros STATUS_H y STATUS_C.

Indicación de grupo de una advertencia de optimización de regulador

El bit QTUN_ERR indica si como mínimo en un DB de instancia de la instrucción ZONESTATUS_H = 2xxxx o STATUS_C = 2xxxx. Compruebe las zonas correspondientes y subsane el error de la forma descrita en Parámetros STATUS_H y STATUS_C.

Control de proceso en una optimización de regulador

El bloque de administración garantiza que en la optimización de regulador el salto de la variable manipulada sea simultáneo en todas las zonas del grupo. Una vez que las diferentes zonas han encontrado el punto de inversión, pasan a la fase 7.

Nota

Interconecte el parámetro de entrada/salida ADMIN de la instrucción ZONE en la estructura interna ADMIN del DB de instancia de ADM_ZONE.

ADM_ZONE siempre se utiliza en combinación con LP_ZONE.

Parámetros de salida de ADM_ZONE

Tabla 9-275

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QCH_F	0.0	BOOL	FALSE	Aviso de grupo: Error de canal
QTUN_ERR	0.1	BOOL	FALSE	Aviso de grupo: Error en la optimización del regulador
QTUN_WRN	0.2	BOOL	FALSE	Aviso de grupo: Advertencia en la optimización del regulador
QHO_DONE	0.3	BOOL	TRUE	Respuesta de un contro manual HEAT_OFF • 0: Se ha iniciado un control manual HEAT_OFF • 1: Control manual HEAT_OFF efectivo en todas las zonas
NBR_CALL	2.0	INT	0	Número de llamada; la instrucción lo registra automáticamente tras un rearranque completo.

Parámetros de entrada/salida de ADM_ZONE

Tabla 9-276

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
HEAT_OFF	4.0	BOOL	TRUE	Calefacción ON/OFF; 1=OFF
TUN_ON	4.1	BOOL	FALSE	Optimización de regulador ON/OFF; 1=ON
COM_RST	4.2	BOOL	TRUE	Rearranque completo; asegúrese de que este bit está activado en el OB 100.
LP_SCH	6.0	STRUCT	P#P 0.0	Interfaz entre las instrucciones LP_ZONE y ADM_ZONE o bien ZONE

Variables estáticas de ADM_ZONE

Tabla 9-277

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
ADMIN	12.0	STRUCT		Interfaz entre las instrucciones ADM_ZONE y ZONE

A continuación siguen parámetros internos que no pueden ser utilizados por el usuario.

ZONE

Descripción de ZONE

El DB de instancia de la instrucción ZONE contiene los datos de regulador de una sola zona de calentamiento o calentamiento/refrigeración. La instrucción ZONE lee y escribe datos en un canal del FM 455.

Las funciones de regulador del FM 455 se complementan con las siguientes funciones:

- Modo con zona de regulación
- Factor de proporción para zonas de calentamiento/refrigeración
- Optimización del regulador

ZONE es por ello especialmente adecuada para regular zonas de calentamiento o calentamiento/refrigeración de un extrusionador.

Requisitos

El canal del FM 455 está parametrizado como regulador de impulsos o continuo.

Las instrucciones PID_FM_455 y PID_PAR_455 están copiadas en el programa de usuario.

Llamada

Llame las instrucciones LP_ZONE, ADM_ZONE y ZONE todas en el mismo nivel de alarma cíclica, preferentemente de forma inmediata en el OB de alarma cíclica. No pueden programarse factores de ciclo de tiempo propios para las instrucciones. Es importante que las zonas asociadas siempre se encuentren en un bloque tras la llamada de la instrucción ADM_ZONE.

Arranque

Active el bit de rearranque completo de ZONE en el OB 100. Si se utiliza LP_ZONE, la activación es automática.

Las siguientes acciones se ejecutan cuando se lanza la rutina de inicialización COM_RST=TRUE

:

- cont_FM455.SDB_SEL = TRUE: los parámetros de regulador de SDB cont_FM455.P_SEL...cont_FM455.BREAK_TM no se aplican en el FM 455. Los parámetros efectivos en el FM continúan siendo vigentes.
- cont_FM455.SP_OP_ON = FALSE: así es efectiva la consigna SP_RE.
- cont_FM455.LOAD_PAR = TRUE: los parámetros de regulador se transmiten del DB de instancia al FM 455.

Funcionamiento

ZONE contiene las instrucciones PID_FM_455 y PID_PAR_455 como instancias locales.

ZONE activa en cada llamada los parámetros cont_FM455.LOAD_OP = TRUE y cont_FM455.READ_VAR = TRUE. Con ello se actualizan todos los parámetros de operación (cont_FM455.SP_RE...cont_FM455.LMNRSVAL) y todos los valores del proceso (cont_FM455.SP...cont_FM455.QMOD_F).

Es posible modificar parámetros de regulador (cont_FM455.P_SEL...cont_FM455.BREAK_TM) cambiando directamente el valor del DB de instancia en la CPU y activando el bit cont_FM455.LOAD_PAR durante un ciclo. No se requiere un DB auxiliar. Una vez que la transmisión ha concluido con éxito, el bit se desactiva. Tras el rearranque completo de la CPU los parámetros del regulador no se sobrescriben con los parámetros de regulador del SDB.

Si trabaja sin LP_ZONE, active el parámetro NBR_CALL=0.

Parámetros de la instrucción PID_FM_455 modificables con ZONE

Tabla 9-278 La siguiente lista contiene los parámetros de la instrucción PID_FM_455 que se ven afectados por ZONE:

Parámetro	Comentario
cont_FM455.READ_VAR	Se pone cíclicamente en TRUE
cont_FM455.LOAD_OP	Se pone cíclicamente en TRUE
cont_FM455.LOAD_PAR	Adopta el valor TRUE en una reparametrización y en caso de modificación de los parámetros CONZONE y RATIOFAC.
cont_FM455.MOD_ADDR	Se aplica cíclicamente a la entrada MOD_ADDR.
cont_FM455.CHANNEL	Se aplica cíclicamente a la entrada CHANNEL.
cont_FM455.SDB_SEL	Adopta el valor TRUE en caso de rearranque completo En una transición de RUN a STOP de la CPU el FM 455 no aplica los parámetros de regulador (P_SEL...BREAK_TM) del SDB.
cont_FM455.SP_OP_ON	Adopta el valor FALSE en caso de rearranque completo, de modo que SP_RE es efectivo.
cont_FM455.SP_RE	Se aplica cíclicamente en SP_IN
cont_FM455.GAIN cont_FM455.TI cont_FM455.TD cont_FM455.TM_LAG cont_FM455.D_EL_SEL cont_FM455.PFDB_SEL cont_FM455.LOAD_PAR	Se activan en caso de optimización de regulador, de nueva carga de los parámetros de regulador guardados y de carga de los parámetros de regulador PI/PID.
cont_FM455.LMN_LLM cont_FM455.LOAD_PAR	Se activan en caso de modificación de RATIOFAC y COOLZONE
cont_FM455.LMN_REON cont_FM455.LMN_RE cont_FM455.LOAD_OP	Se activan cíclicamente

Parámetros de la instrucción PID_PAR_455 modificables con ZONE

Tabla 9-279 La siguiente lista contiene los parámetros de la instrucción PID_PAR_455, que se ven afectados por ZONE:

Parámetro	Comentario
pidpar_FM455.COM_RST	Adopta el valor TRUE en caso de rearranque completo
pidpar_FM455.MOD_ADDR	Se aplica cíclicamente a la entrada MOD_ADDR.
pidpar_FM455.CHANNEL	Se aplica cíclicamente a la entrada CHANNEL.
pidpar_FM455. INDEX_R	Adoptan el valor 43 en caso de modificación de RATIOFAC y COOLZONE
pidpar_FM455. VALUE _R	Se activa en caso de modificación de RATIOFAC y COOLZONE

Parámetros de entrada ZONE

Tabla 9-280

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
MOD_ADDR	0.0	INT	512	Dirección del módulo
CHANNEL	2.0	INT	1	Número de canal
LPSCH_EN	4.0	BOOL	TRUE	Si LPSCH_EN = TRUE, ZONE se controla con la instrucción LP_ZONE. El parámetro de entrada/salida LP_SCH debe estar interconectado con la variable estática LP_SCH de la instrucción LP_ZONE. De lo contrario la CPU pasa a STOP con "Error de área al escribir". Si LPSCH_EN = FALSE, ZONE se ejecuta en cada ciclo de alarma cíclica. No es necesario asignar un valor al parámetro de entrada/salida LP_SCH. El tiempo de muestreo debe estar registrado en la variable estática CYCLE.
ADMIN_EN	4.1	BOOL	TRUE	Si ADMIN_EN = TRUE, la instrucción ADM_ZONE gestiona varias zonas de calentamiento/refrigeración. El parámetro de entrada/salida ADMIN debe estar interconectado con la variable estática ADMIN de la instrucción ADM_ZONE. De lo contrario la CPU pasa a STOP con "Error de área al escribir". Si ADMIN_EN = FALSE, ZONE trabaja sin la instrucción ADM_ZONE. No es necesario asignar un valor al parámetro de entrada/salida ADMIN.

Parámetros de entrada/salida de ZONE

Tabla 9-281

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LP_SCH	6.0	STRUCT	P#P 0.0	Interfaz entre las instrucciones LP_ZONE y ADM_ZONE o bien ZONE
ADMIN	12.0	STRUCT	P#P 0.0	Interfaz entre las instrucciones ADM_ZONE y ZONE

Variables estáticas de ZONE

Tabla 9-282

Parameter	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
cont_FM455	18.0	FB PID_FM_455		Instancia local de la instrucción PID_FM_455
pidpar_FM455	172.0	FB PID_PAR_455		Instancia local de la instrucción PID_PAR_455
SP_IN	426.0	REAL	0.0	La entrada SP_IN permite predeterminar una consigna.
CYCLE	430.0	REAL	0.1	Valor predeterminado del tiempo de muestreo para el algoritmo PID. Si se utiliza la instrucción LP_ZONE, LP_ZONE calcula automáticamente el tiempo de muestreo.
H_OFF_EN	434.0	BOOL	TRUE	TRUE= conmutador HEAT_OFF es efectivo e influye en los parámetros MAN_ON y MAN.
TUN_EN	434.1	BOOL	TRUE	Si en una optimización de regulador TUN_EN=0, no se calculan parámetros de regulador nuevos. Atención: el salto de la variable manipulada se aplica de todas formas.
COOLZONE	434.2	BOOL	FALSE	COOLZONE = FALSE: zona sólo de calentamiento COOLZONE = TRUE: zona de calentamiento/refrigeración
PID_ON	434.3	BOOL	TRUE	En la entrada PID_ON se puede determinar si el regulador optimizado debe trabajar como regulador PI o PID. Regulador PID: PID_ON = TRUE Regulador PI: PID_ON = FALSE Puede ocurrir que en algunos tipos de proceso sólo se diseñe un regulador PI a pesar de PID_ON = TRUE.
QCH_F	434.4	BOOL	FALSE	QCH_F = TRUE: en una de las llamadas de la instancia local de la instrucción PID_FM_455 (parámetro cont_FM455.QCH_F) se ha producido un error de canal del FM 455.
QTUN_ERR	434.5	BOOL	FALSE	QTUN_ERR = TRUE: se ha producido un error de operación en la optimización de regulador. STATUS_H o bien STATUS_C = 3xxxx
QTUN_WRN	434.6	BOOL	FALSE	QTUN_WRN = TRUE: se ha producido una advertencia en la optimización de regulador. STATUS_H o bien STATUS_C = 2xxxx
QTUNSTCP	434.7	BOOL	FALSE	QTUNSTCP = TRUE: la optimización se ha iniciado desde el estado frío del proceso y tras la optimización debe regularse al punto de operación. MAN_ON = FALSE; MAN = 0 (calefacción desconectada) SP_IN > PV0 +10 (Punto de operación mayor que el valor real en 10 en el inicio de la optimización)
QCONTINU	435.0	BOOL	TRUE	Indica si la zona está preparada para la siguiente fase de la optimización de regulador.

Parameter	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
PHASE	436.0	INT	0	En el parámetro PHASE se muestra la fase actual de la optimización de regulador (0..7, 11..13).
STATUS_H	438.0	INT	0	El parámetro STATUS_H indica un valor de diagnóstico a través de la búsqueda del punto de inversión en el proceso de calentamiento.
STATUS_C	440.0	INT	0	El parámetro STATUS_C indica un valor de diagnóstico a través de la búsqueda del punto de inversión en el proceso de refrigeración.
STATUS_D	442.0	INT	0	El parámetro STATUS_D indica un valor de diagnóstico a través del diseño del regulador en el proceso de calentamiento.
PI_CON	444.0	STRUCT		Parámetros PI del regulador
GAIN	+0.0	REAL	0.0	Ganancia PI del regulador
TI	+4.0	REAL	0.0	Tiempo de integración PI [s]
	=8.0	END_STRUCT		
PID_CON	452.0	STRUCT		Parámetros PID del regulador
GAIN		REAL	0.0	Ganancia PID del regulador
TI		REAL	0.0	Tiempo de integración PID [s]
TD		REAL	0.0	Tiempo de acción derivada PID [s]
	=12.0	END_STRUCT		
PAR_SAVE	464.0	STRUCT		En esta estructura se guardan los parámetros PID.
GAIN	+0.0	REAL	0.0	Ganancia del regulador
TI	+4.0	REAL	0.0	Tiempo de integración [s]
TD	+8.0	REAL	0.0	Tiempo de acción derivada [s]
TM_LAG	+12.0	REAL	0.0	Retardo de la acción D [s]
D_EL_SEL	+16.0	INT	0	Entrada para el componente D
PFDB_SEL	+18.0	BOOL	FALSE	Acción P en lazo de realimentación
CONZ_ON	+18.1	BOOL	FALSE	Activar zona de regulación
CON_ZONE	+20.0	REAL	100.0	Ancho de zona de regulación
RATIOFAC	+24.0	REAL	1.0	Relación entre ganancia de calentamiento/refrigeración del proceso.
	=28.0	END_STRUCT		
HEAT_OFF	492.0	BOOL	FALSE	TRUE= desconectar la calefacción Hay que activar H_OFF_EN = TRUE para que HEAT_OFF sea efectivo. • Al activar HEAT_OFF, MAN_ON=TRUE y MAN=0.0. • Al desactivar HEAT_OFF, MAN_ON=FALSE.

Parameter	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
TUN_ON	492.1	BOOL	FALSE	Activar optimización del regulador Active TUN=TRUE sólo por un ciclo. TUN_ON se desactiva automáticamente tras la optimización.
CTUN_ON	492.2	BOOL	FALSE	Activar sólo la optimización de refrigeración Active CTUN =TRUE sólo por un ciclo. CTUN _ON se desactiva automáticamente tras la optimización.
COM_RST	492.3	BOOL	FALSE	Se inicia la rutina de inicialización
TUN_LMN	494.0	REAL	80.0	La excitación del proceso para la optimización del regulador desde el estado frío del proceso se realiza mediante un salto de la variable manipulada de TUN_LMN.
TUN_DLMN	498.0	REAL	20.0	La excitación del proceso para la optimización del regulador en el punto de operación se realiza mediante un salto de la variable manipulada de TUN_DLMN.
TUN_CLMN	502.0	REAL	-10.0	Variable manipulada tras iniciar la optimización de la refrigeración.
STAT_TM	506.0	REAL	30.0	Tiempo para la condición de estacionariedad para la fase 1 u 11 de la optimización del regulador El proceso se considera estacionario si dentro del tiempo STAT_TM el valor real o manipulado no cambia más de 2*STAT_D_PV o 2*STATDLMN.
STAT_DPV	510.0	REAL	1.0	Delta de valor real para la condición de estacionariedad para la fase 1 u 11 de la optimización de regulador Véase STAT_TM
STATDLMN	514.0	REAL	4.0	Delta de variable manipulada para el control manual de estacionariedad para la fase 1 u 11 de la optimización del regulador Véase STAT_TM
RATIOFAC	518.0	REAL	1.0	Relación entre ganancia de calentamiento/refrigeración del proceso. El límite inferior de la variable manipulada e "Inicio de rango de señal de entrada de variable manipulada B de la función de rango partido" se adaptan automáticamente.
CON_ZONE	522.0	REAL	100.0	Zona de regulación Si ER >= CON_ZONE, entonces LMN = LMN_HLM. Si ER <= -CON_ZONE (o bien -CON_ZONE /RATIOFAC), entonces LMN = LMN_LLM.
MAN	526.0	REAL	0.0	Valor manual Si H_OFF_EN=TRUE, el parámetro se ve afectado por HEAT_OFF.
MAN_ON	530.0	BOOL	TRUE	Conmutación manual/automática Si H_OFF_EN=TRUE, el parámetro se ve afectado por HEAT_OFF.
CONZ_ON	530.1	BOOL	FALSE	TRUE = zona de regulación activada

Parameter	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
UNDO_PAR	530.2	BOOL	FALSE	Copia los parámetros de regulador de la estructura PAR_SAVE en los parámetros: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD • cont_FM455.TM_LAG • cont_FM455.D_EL_SEL • cont_FM455.PFDB_SEL • CONZ_ON • CON_ZONE • RATIOFAC A continuación los parámetros se cargan en el FM 455. La función no se ejecuta en el modo automático.
SAVE_PAR	530.3	BOOL	FALSE	Guarda los siguientes parámetros de regulador en la estructura PAR_SAVE: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD • cont_FM455.TM_LAG • cont_FM455.D_EL_SEL • cont_FM455.PFDB_SEL • CONZ_ON • CON_ZONE • RATIOFAC
LOAD_PID	530.4	BOOL	FALSE	Cargar parámetros PI/PID: Con PID_ON=TRUE copia los parámetros PID de la estructura PID_CON en los parámetros: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD • cont_FM455.TM_LAG Con PID_ON=FALSE copia los parámetros PI de la estructura PI_CON en los parámetros: • cont_FM455.GAIN • cont_FM455.TI • cont_FM455.TD = 0.0 • cont_FM455.TM_LAG = 0.5 A continuación los parámetros se cargan en el FM 455. La función no se ejecuta en el modo automático.
BTUN_DPV	532	REAL	0.0	Diferencia de valor real para la cancelación automática de la optimización.
BTUN_TM	536	REAL	300.0	Tiempo tras el salto de la variable manipulada para la cancelación automática de la optimización.

Variables para la optimización de regulador ZONE

Tabla 9-283

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
GAIN_P	540.0	REAL	0.0	Ganancia del proceso identificada.
TU	544.0	REAL	0.0	Tiempo de retardo identificado del proceso.
TA	548.0	REAL	0.0	Tiempo de compensación identificado del proceso.
KIG	552.0	REAL	0.0	Rampa máxima de subida del valor real en el calentamiento.
KIG_C	556.0	REAL	0.0	Rampa máxima de bajada del valor real en la refrigeración.
N_PTN	560.0	REAL	0.0	Orden del proceso. También se pueden utilizar "valores no enteros".
TM_LAG_P	564.0	REAL	0.0	Constante temporal de un modelo PTN. (valores razonables sólo para N_PTN>=2).
T_P_INF	568.0	REAL	0.0	Tiempo desde la excitación del proceso hasta el punto de inversión.
P_INF	572.0	REAL	0.0	Modificación del valor real desde la excitación del proceso hasta el punto de inversión.
LMN0	576.0	REAL	0.0	Variable manipulada al iniciar la optimización. Se calcula en la fase 1 (valor medio).
PV0	580.0	REAL	0.0	Valor real al inicio de la optimización. Se calcula en la fase 1 (valor medio).
PVDT0	584.0	REAL	0.0	Rampa de subida de valor real al inicio de la optimización.
PVDT	588.0	REAL	0.0	Rampa de subida actual de valor real
PVDT_MAX	592.0	REAL	0.0	Derivación máxima del valor real en el punto de inversión (con adaptación de signo, siempre > 0); se utiliza para calcular TU y KIG.
NOI_PVDT	596.0	REAL	0.0	Nivel de ruido en PVDT_MAX en %. Cuanto mayor sea el nivel de ruido, menos precisos (de efecto más suave) serán los parámetros del regulador.
NOISE_PV	600.0	REAL	0.0	Ruido absoluto en el valor real; Diferencia entre el valor real máximo y mínimo en la fase 1.
FIL_CYC	604.0	INT	1	Número de ciclos del filtro de valores medios 1 ... 1024; Se calcula un valor medio del valor real a partir de FIL_CYC ciclos. En caso necesario, FIL_CYC se aumenta automáticamente entre 1 y máx. 1024.
POI_CMAX	606.0	INT	0	Número máximo de ciclos tras el punto de inversión. Este tiempo se utiliza para encontrar otro punto de inversión (es decir, uno mejor) en caso de ruidos de medición. Sólo entonces concluye la optimización.
POI_CYCL	608.0	INT	0	Número de ciclos tras el punto de inversión.
NBR_CALL	610.0	INT	-1	Número de llamada; se registra automáticamente tras un rearranque completo.

A continuación siguen parámetros internos que no pueden ser utilizados por el usuario.

Parámetros STATUS_H y STATUS_C

STATUS_H indica un valor de diagnóstico a través de la búsqueda del punto de inversión, STATUS_C en la refrigeración.

Si se indica un valor de diagnóstico 2xxxx, QTUN_WRN = TRUE.

Si se indica un valor de diagnóstico 3xxxx, QTUN_ERR = TRUE.

STATUS_ H STATUS_C	Descripción	Solución
0	Parámetros del regulador predeterminados, ninguno o ninguno nuevo	
10000	Optimización finalizada + parámetros adecuados encontrados para el regulador	
2xxxx	Optimización finalizada + parámetros del regulador inseguros	
2xx2x	Punto de inversión no alcanzado (sólo en caso de optimización desde el estado frío del proceso al punto de operación QTUNSTCP =TRUE)	Si el regulador oscila, atenuar los parámetros del regulador o repetir el ensayo con una diferencia menor de la variable manipulada TUN_LMN.
2x1xx	Error de estimación TU demasiado pequeño TU < 3*CYCLE	Reducir CYCLE y repetir el ensayo. Caso especial, proceso puramente PT1: No repetir el ensayo, atenuar los parámetros del regulador si procede.
2x2xx	Error de estimación T_P_INF demasiado pequeño T_P_INF < 72.8*CYCLE	Véase 2x1xx
2x3xx	Error de estimación TU demasiado pequeño y error de estimación T_P_INF demasiado pequeño	Véase 2x1xx
3xxxx	La optimización se ha cancelado. No se ha determinado ningún parámetro de regulador.	
30002	En la optimización de calentamiento: Diferencia efectiva de la variable manipulada < 5 % En la optimización de la refrigeración: TUN_CLMN > -5 %	Corregir la diferencia de variable manipulada TUN_LMN, TUN_DLMN o bien TUN_CLMN.
30003	En la optimización de refrigeración el límite inferior de la variable manipulada no es correcto: cont_FM455.LMN_LLM > -5.0	Ajuste la zona como zona de calentamiento/ refrigeración COOLZONE =TRUE y compruebe cont_FM455.LMN_LLM.
30004	Diferencia efectiva de la variable manipulada limitada por los límites de rango partido y no por los límites de la variable manipulada.	Compruebe los límites de la variable manipulada y de rango partido. Corrija la diferencia de la variable manipulada TUN_LMN, TUN_DLMN o bien TUN_CLMN.
30005	El tiempo de muestreo CYCLE difiere en más del 5% del valor medido.	Compare CYCLE con el tiempo de ciclo del nivel de alarma cíclica y, dado el caso, observe los posibles distribuidores de llamada. Compruebe el grado de utilización de la CPU. Una CPU con un grado de utilización excesivo conlleva tiempos de muestreo excesivos, que no concuerdan con CYCLE.
30008	Se intenta realizar una optimización de refrigeración, pero antes no se ha llevado a cabo una optimización de calentamiento (KIG=0.0).	Realizar primero una optimización de calentamiento.
30009	Modo de seguridad: cont_FM455.QLMNSAFE = TRUE	Finalice el modo de seguridad y vuelva a iniciar la optimización.
30010	Una modificación de la potencia de calefacción/ refrigeración no provoca un cambio suficiente en la temperatura.	Compruebe la evolución de la señal de la variable manipulada y del valor de tiempo. Compruebe si los parámetros BTUN_TM y BTUN_DPV son plausibles.

Consulte también

Parámetros de salida de ADM_ZONE (Página 7609)

Parámetro STATUS_D

STATUS_D	Descripción
0	No se han calculado parámetros del regulador
110	N_PTN <= 1.5 tipo de sistema I rápido
121	N_PTN > 1.5 tipo de sistema I
200	N_PTN > 1.9 tipo de sistema II (rango de transición)
310	N_PTN >= 2.1 tipo de sistema III rápido
320	N_PTN > 2.6 tipo de sistema III
111, 122, 201, 311, 321	Los parámetros se han corregido en la fase 7.

Nota

Al final de la fase 1 STATUS_H = 0 se desactiva. Si la optimización se cancela en la fase 2, entonces STATUS_H = 0. Sin embargo, STATUS_D sigue mostrando el estado del último cálculo del regulador.

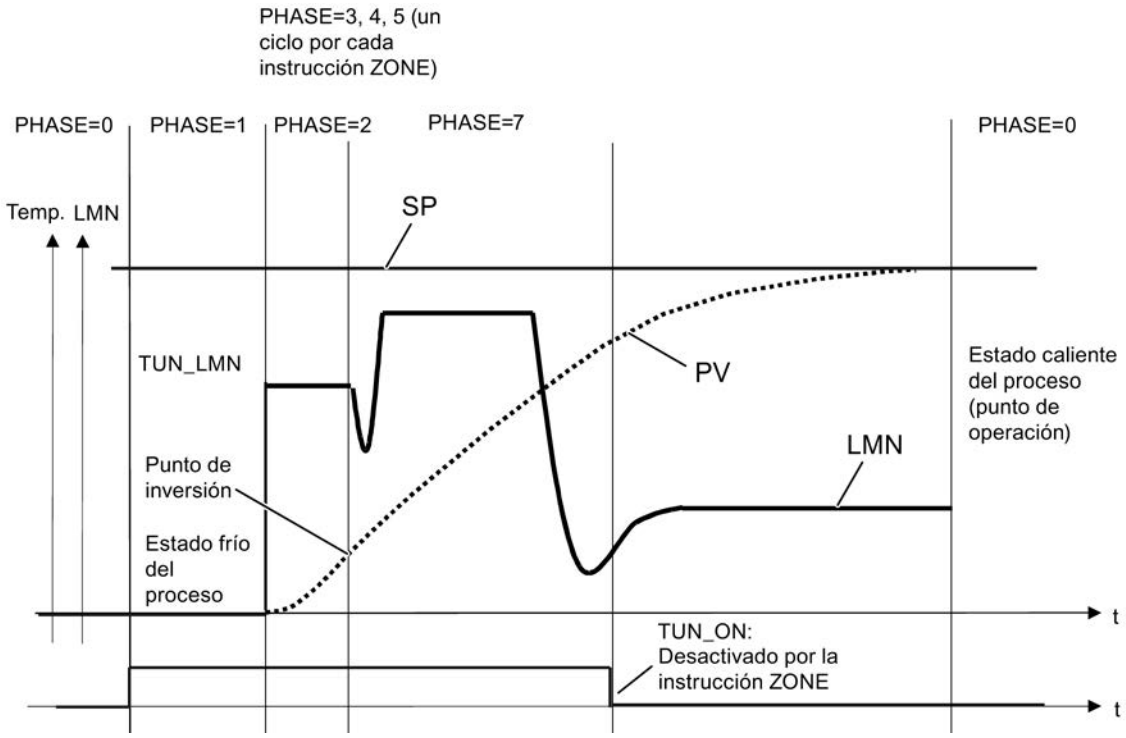
Cuanto mayor sea el valor de STATUS_D, mayor será el orden del proceso regulado, mayor la relación TU/TA y más suaves los parámetros del regulador.

Parámetro PHASE

PHASE	Descripción
0	No hay modo de optimización; modo automático o manual
1	Prueba de estacionariedad; comprobar parámetros; aplicar una variable manipulada de calentamiento constante
2	Búsqueda del punto de inversión con variable manipulada constante
3 (1 ciclo por instrucción ZONE)	Calcular los parámetros del proceso. Guardar los parámetros de regulador válidos antes de la optimización
4 (1 ciclo por instrucción ZONE)	Diseño del regulador
5 (1 ciclo por instrucción ZONE)	Corregir el regulador a la nueva magnitud manipulada
7	Verificación del tipo de sistema (sólo con optimización de calefacción)
10	Optimización de refrigeración: Esperar hasta que todas las zonas estén en modo automático (fase > 2).
11	Optimización de refrigeración: Prueba de estacionariedad; comprobar parámetros; aplicar una variable manipulada de refrigeración constante
12	Optimización de refrigeración: Búsqueda de punto de inversión con variable manipulada constante
13	Optimización de refrigeración: Cálculo de la relación calentamiento/refrigeración

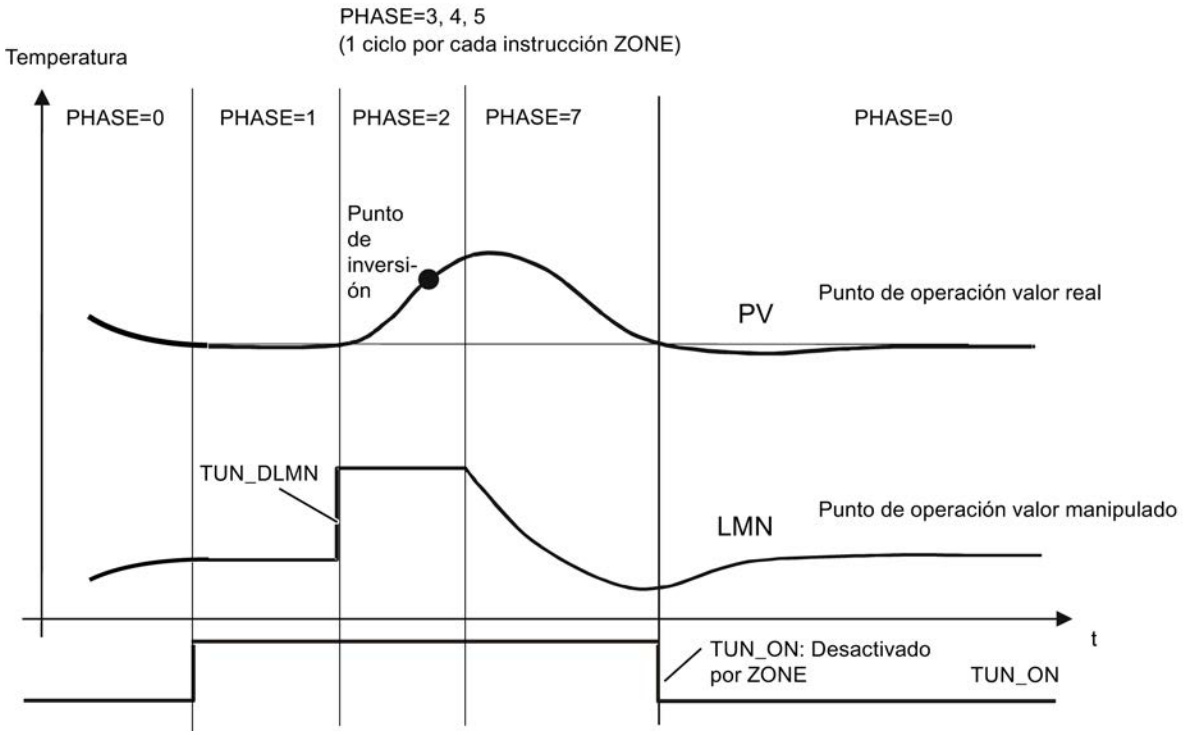
Fases de la optimización de calefacción en el primer arranque

La figura siguiente muestra las fases de una optimización sólo de calentamiento partiendo de la temperatura ambiente hasta el punto de operación:



Fases de la optimización de calentamiento en el punto de operación

La figura siguiente muestra las fases de una optimización sólo de calentamiento en el punto de operación:



Al final de la optimización, si la instrucción vuelve a cambiar a la fase 0 y TUN_ON=FALSE, en los parámetros STATUS_H, STATUS_C o QTUN_WRN, QTUN_ERR se puede saber si la optimización se ha desarrollado sin errores.

DB para HMI

Descripción de DB 101 a 116

Para poder manejar el FM 455 con el OP en caso de fallo de la CPU se requieren los bloques de datos 101 a 104. Con MPI se pueden establecer hasta tres conexiones desde el FM 455 a los OPs.

El manejo del FM 455 con el OP sólo es posible en el estado STOP de la CPU en caso de fallo de la CPU.

El FM 455 siempre se puede observar con el OP.

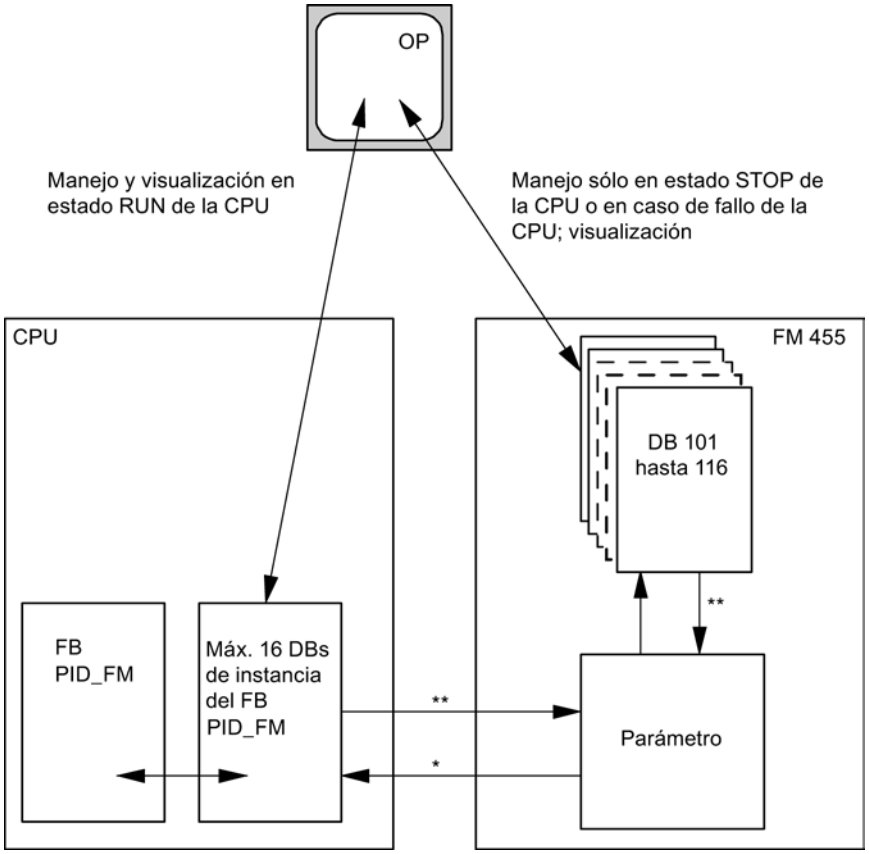
La interfaz de variables del FM 455 contiene 16 bloques de datos con los números de bloque 101 a 116 para los canales de regulador 1 a 16 (vea la siguiente figura).

Nota

El contenido de los bloques de datos 101 a 116 no refleja automáticamente el valor de parámetro efectivo en el FM 455. Los parámetros cambiados con ayuda del OP sólo se aplican en el FM 455 tras haberse activado los bits de operación LOAD_PARo LOAD_OP.

Si se cambia un parámetro manejando el OP sin activar el bit de operación correspondiente, el valor de parámetro modificado está en el bloque de datos, pero el FM 455 trabaja internamente con el valor anterior no modificado del parámetro.

Tras activar los bits de operación y aplicar los parámetros en el FM 455, los bits de operación LOAD_PAR o LOAD_OP son desactivados por el FM 455.



* Controlado por el parámetro READ_VAR del DB de instancia

** Controlado por los parámetros LOAD_OP y LOAD_PAR

Parámetros de entrada de los DB 101 a 116

Nota

La EPROM del módulo podría quedar destruida por un número excesivo de procesos de escritura. Para evitarlo, tras la escritura de la EEPROM el módulo retrasa una nueva escritura 30 minutos.

Tabla 9-284

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
[1] parámetros de regulador; [2] parámetros operativos				
SP_HLM[1]	0.0	REAL	100.0	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite superior consigna" indica el límite superior. SP_HLM > SP_LLM
SP_LLM[1]	4.0	REAL	0.0	La consigna está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior consigna" indica el límite inferior. SP_LLM < SP_HLM
H_ALM[1]	8.0	REAL	100.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más alto. H_ALM > H_WRN
H_WRN[1]	12.0	REAL	90.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite superior" indica el segundo límite más alto. Se admiten valores entre H_ALM y L_WRN
L_WRN[1]	16.0	REAL	10.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Advertencia de límite inferior" indica el segundo límite más bajo. Se admiten valores entre H_WRN y L_ALM
L_ALM[1]	20.0	REAL	0.0	Para la vigilancia del valor real o del error de regulación se pueden parametrizar cuatro límites. La entrada "Alarma de límite superior" indica el límite más bajo. L_ALM < L_WRN
HYS[1]	24.0	REAL	1.0	Para evitar oscilaciones de las indicaciones de vigilancia, puede parametrizarse una histéresis en la entrada "Histéresis". HYS ≥ 0.0
DEADB_W[1]	28.0	REAL	0.0	El error de regulación se conduce por una zona muerta. La entrada "Ancho de zona muerta" determina el tamaño de la zona muerta. DEADB_W ≥ 0.0
GAIN[1]	32.0	REAL	1.0	La entrada "Ganancia proporcional" indica la ganancia del regulador.
TI[1]	36.0	REAL	3000.0	La entrada "Tiempo de integración" determina el comportamiento temporal del integrador. Con TI = 0 el integrador está desactivado. TI = 0.0 o bien TI ≥ 0.5
TD[1]	40.0	REAL	0.0	La entrada "Tiempo de acción derivada" determina el comportamiento temporal del diferenciador. Con TD = 0 el diferenciador está desactivado. TD = 0.0 o bien TD ≥ 1.0

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
TM_LAG[1)]	44.0	REAL	5.0	El algoritmo de la acción D contiene un retardo que se puede parametrizar en la entrada "Tiempo de retardo de la acción D". TM_LAG ≥ 0.5
LMN_SAFE[1)]	48.0	REAL	0.0	Para la variable manipulada puede parametrizarse un valor de seguridad en la entrada "Variable manipulada de seguridad". Se admiten valores entre -100 y 100 %.
LMN_HLM[1)]	52.0	REAL	100.0	La variable manipulada está limitado siempre por un límite superior y uno inferior. La entrada "Límite superior de la variable manipulada" indica el límite superior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre LMN_LLM y 100 %.
LMN_LLM[1)]	56.0	REAL	0.0	La variable manipulada está limitada siempre por un límite superior y uno inferior. La entrada "Límite inferior de la variable manipulada" indica el límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre -100 y LMN_HLM.
MTR_TM[1)]	60.0	REAL	60.0	En el parámetro "Tiempo de posicionamiento del motor" se introduce el tiempo de operación de la servoválvula de un tope al otro. (Sólo en regulador paso a paso) MTR_TM ≥ 0.001
PULSE_TM[1)]	64.0	REAL	0.2	En el parámetro "Duración mínima de impulso" se puede parametrizar una duración de impulso mínima. (Sólo en regulador paso a paso o de impulsos)
BREAK_TM[1)]	68.0	REAL	0.2	En el parámetro "Duración mínima de pausa" se puede parametrizar una duración de pausa mínima. (Sólo en regulador paso a paso o de impulsos)
SP_RE[2)]	72.0	REAL	0.0	En la entrada "Consigna externa" se conecta una consigna externa con el regulador. Rango de valores técnico (magnitud física)
LMN_RE[2)]	76.0	REAL	0.0	En la entrada "Variable manipulada externa" se conecta una variable manipulada externo con el regulador. Se admiten valores entre -100 y 100 %.
LMNRSVAL[2)]	80.0	REAL	0.0	Desde el cuadro de diálogo Puesta en marcha se puede acceder a la entrada "Valor inicial de la realimentación de posición simulada". En ese parámetro se indica el valor inicial de la simulación. (Sólo en el regulador paso a paso sin realimentación de posición analógica) Se admiten valores entre -100 y 100 %.
SAFE_ON[2)]	84.0	BOOL	FALSE	Si la entrada "Adoptar posición de seguridad" está activada, se aplica un valor de seguridad como variable manipulada. Nota: El procesamiento de la señal de salida con LMNDN_OP, LMNUP_OP y LMNSOPON en el regulador paso a paso tiene mayor prioridad que la variable manipulada de seguridad.
LMNTRKON[2)]	84.1	BOOL	FALSE	Si la entrada "Corregir (LMN a través de EA)" está activada, la variable manipulada se rige por una entrada analógica (EA). (No en el regulador paso a paso sin realimentación de posición analógica)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
LMN_REON 2)	84.2	BOOL	FALSE	Si la entrada "Aplicar variable manipulada externa" está activada, se aplica la variable manipulada externa LMN_RE como variable manipulada.
LMNRHSRE2)	84.3	BOOL	FALSE	La señal "Servoválvula en el tope superior" se interconecta a la entrada "Señal de tope superior de la realimentación de posición". LMNRHSRE = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
LMNRLSRE 2)	84.4	BOOL	FALSE	La señal "Servoválvula en el tope inferior" se interconecta a la entrada "Señal de tope inferior de la realimentación de posición". LMNRLSRE = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
LMNSOPON2)	84.5	BOOL	FALSE	Si el bit de la entrada "Activar control manual de la señal de variable manipulada" está activada, las señales LMNUP_OP y LMNDN_OP se aplican como señales de variable manipulada. (Sólo en regulador paso a paso)
LMNUP_OP 2)	84.6	BOOL	FALSE	Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Subir señal de variable manipulada". (Sólo en regulador paso a paso)
LMNDN_OP 2)	84.7	BOOL	FALSE	Si LMNSOPON está activado, el valor se adopta como señal de variable manipulada en la entrada "Bajar señal de variable manipulada". (Sólo en regulador paso a paso)
MONERSEL1)	85.0	BOOL	FALSE	El regulador dispone de un avisador de valor límite, que puede utilizarse para el valor real o para el error de regulación. Si la entrada "Vigilancia: Valor real = 0, Error de regulación = 1" está activada, se vigila el error de regulación.
LMNRS_ON 2)	85.1	BOOL	FALSE	Si no se dispone de una realimentación de posición, puede simularse. La función se activa en la entrada "Activar simulación de la realimentación de posición". ATENCIÓN: Con el paso del tiempo, la simulación difiere cada vez más de la realimentación de posición real. (Sólo en el regulador paso a paso sin realimentación de posición analógica)
FUZID_ON2)	85.2	BOOL	FALSE	En la entrada "Activar identificación de difuso" se activa la identificación del algoritmo difuso.
SPINT_EN2)	85.3	BOOL	FALSE	La entrada "Entrada de control manual: externo = 0, interno = 1" determina la entrada que se transmite al módulo como consigna. SPINT_EN = TRUE: se transfiere SP_INT. SPINT_EN = FALSE: se transfiere SP_RE.
P_SEL1)	85.4	BOOL	TRUE	En el algoritmo PID las acciones PID se pueden conectar y desconectar individualmente. La acción proporcional está conectada si la entrada "Activar acción P" está activada.
PFDB_SEL1)	85.5	BOOL	FALSE	En el algoritmo PID las acciones P y D se pueden incluir en el lazo de realimentación. La acción proporcional se encuentra en el lazo de realimentación si la entrada "Incluir acción P en lazo de realimentación" está activada.
D_EL_SEL1)	86.0	INT	0	El componente D del algoritmo PID puede depositarse en una entrada separada. Ésta se selecciona en la entrada "Entrada para componente D". • 0: Error de regulación • 1 a 16: Entrada analógica 1 a 16 • 17: Valor real negativo

Parámetros de salida de los DB 101 a 116

Tabla 9-285

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP	94.0	REAL	0.0	En la salida "Consigna" se indica la consigna que actúa efectivamente.
PV	98.0	REAL	0.0	En la salida "Valor real" se indica el valor real que actúa efectivamente.
ER	102.0	REAL	0.0	En la salida "Error de regulación" se indica el error de regulación que actúa realmente.
DISV	106.0	REAL	0.0	En la salida "Variable perturbadora" se indica la magnitud perturbadora que actúa efectivamente. Se admiten valores entre -100 y 100 %.
LMN	110.0	REAL	0.0	En la salida "Variable manipulada" se indica la variable manipulada que actúa efectivamente. En el regulador paso a paso sin realimentación de posición analógica, en el parámetro LMN se indica la acción P + D ilimitada. Se admiten valores entre -100 y 100 %.
LMN_A	114.0	REAL	0.0	En la salida "Variable manipulada A de la función de rango partido / realimentación de posición" se indica la variable manipulada A de la función de rango partido en el regulador continuo, y la realimentación de posición en el regulador paso a paso con realimentación de posición analógica. En el regulador paso a paso sin realimentación de posición analógica se muestra la realimentación de posición simulada. Se admiten valores entre -100 y 100 %.
LMN_B	118.0	REAL	0.0	Para el regulador continuo, en la salida "Variable manipulada B de la función de rango partido" se indica la variable manipulada B de la función de rango partido. Se admiten valores entre -100 y 100 %.
QH_ALM	122.0	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite superior disparada" se notifica el rebase del límite H_ALM.
QH_WRN	122.1	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite superior disparada" se notifica el rebase del límite H_WRN.
QL_WRN	122.2	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Advertencia de límite inferior disparada" se notifica el rebase por defecto del límite L_WRN.
QL_ALM	122.3	BOOL	FALSE	El valor real o la magnitud regulada se somete a un control de cuatro valores límite. En la salida "Alarma de límite inferior disparada" se notifica el rebase por defecto del límite L_ALM.
QLMN_HLM	122.4	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de variable manipulada alcanzado" se notifica el rebase del límite superior. (No en el regulador paso a paso sin realimentación de posición analógica)
QLMN_LLM	122.5	BOOL	FALSE	La variable manipulada está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la variable manipulada alcanzado" indica el rebase por defecto del límite inferior. (No en el regulador paso a paso sin realimentación de posición analógica)

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QSPINTON	122.6	BOOL	FALSE	La salida "Consigna interna activada" indica que SP_INT se ha transferido al módulo.
QPARA_F	123.0	BOOL	FALSE	El módulo comprueba la validez de los parámetros. Un error de parametrización se indica en la salida "Error de parametrización". Este error de parametrización también se puede ver en el cuadro de diálogo "Online y diagnóstico" del módulo.
QCH_F	123.1	BOOL	FALSE	La salida "Error de canal" se activa cuando el canal de regulación no puede arrojar resultados válidos. El error de canal (p. ej. rotura de hilo) también se activa con QPARA_F = 1 o QMOD_F = 1. Si QCH_F = TRUE, la información precisa del error puede verse en el registro de diagnóstico DS1 del módulo.
QUPRLM	123.2	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación positiva de rampa de subida de la consigna" está activada, se limita la rampa de subida de la consigna.
QDNRLM	123.3	BOOL	FALSE	La consigna se limita en las rampas de subida y bajada. Si la salida "Limitación negativa de rampa de subida de la consigna" está activada, se limita la rampa de bajada de la consigna.
QSP_HLM	123.4	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. En la salida "Límite superior de consigna alcanzado" se notifica el rebase del límite superior.
QSP_LLM	123.5	BOOL	FALSE	La consigna está limitada siempre por un límite superior y uno inferior. La salida "Límite inferior de la consigna alcanzado" indica el rebase del límite inferior.
QSPOPON	123.6	BOOL	FALSE	En la salida "Control manual de consigna activado" se indica si la consigna se controla a través del cuadro de diálogo "Puesta en servicio". Si el bit está activado, el valor SP_OP se aplica como consigna.
QLMNSAFE	123.7	BOOL	FALSE	Si la salida "Modo de seguridad" está activada, como variable manipulada se emite la variable manipulada de seguridad.
QLMNOPON	124.0	BOOL	FALSE	En la salida "Control manual de variable manipulada activado" se indica si la variable manipulada se controla desde el cuadro de diálogo "Puesta en servicio". Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
QLMNTRK	124.1	BOOL	FALSE	La salida "Modo de seguimiento" indica si la variable manipulada se corrige mediante una entrada analógica.
QLMN_RE	124.2	BOOL	FALSE	La salida "Manual = 1; Automático = 0" indica si la variable manipulada adopta la variable manipulada externa LMN_RE (Manual = 1) o no.
QLMNR_HS	124.3	BOOL	FALSE	La salida "Señal de tope superior de la realimentación de posición" indica si la servoválvula se encuentra en el tope superior. QLMNR_HS = TRUE significa: La servoválvula se encuentra en el tope superior. (Sólo en regulador paso a paso)
QLMNR_LS	124.4	BOOL	FALSE	La salida "Señal de tope inferior de la realimentación de posición" indica si la servoválvula se encuentra en el tope inferior. QLMNR_LS = TRUE significa: La servoválvula se encuentra en el tope inferior. (Sólo en regulador paso a paso)
QLMNR_ON	124.5	BOOL	FALSE	La salida "Realimentación de posición activada" indica que está activado el modo de operación "Regulador paso a paso con realimentación de posición" o "Regulador paso a paso sin realimentación de posición".
QFUZZY	124.6	BOOL	FALSE	Si la salida "Algoritmo PID = 0; Difuso = 1" está activada, el regulador trabaja con el algoritmo difuso.
QSPLEPV	124.7	BOOL	FALSE	La salida "Indicación del regulador difuso: consigna < valor real" se activa con el regulador difuso si la consigna es menor que el valor real efectivo.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
QSPR	125.0	BOOL	FALSE	Si la salida "Modo de rango partido" está activada, el regulador continuo trabaja en modo de rango partido.
QLMNUP	125.1	BOOL	FALSE	Se trata de la salida "Señal alta de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QLMNDN	125.2	BOOL	FALSE	Se trata de la salida "Señal baja de variable manipulada". (Sólo en regulador paso a paso o de impulsos)
QBACKUP	125.4	BOOL	FALSE	• 0= no hay estado de copia de seguridad (CPU en RUN) • 1= estado de copia de seguridad (CPU e STOP o falla)
QID	125.5	BOOL	FALSE	QID = TRUE indica que hay una identificación en curso (no que esté activada). Una vez completada la identificación, el resultado puede verse a través del parámetro IDSTATUS de la instrucción CH_DIAG_455.
QMAN_FC	125.6	BOOL	FALSE	El regulador es un regulador piloto que, mediante la operación manual de un regulador secundario, se corrige según el valor real de este último, o cuya acción I se detiene porque la consigna o la magnitud manipulada del regulador secundario están en el límite.
RET_VALU	126.0	WORD	0	RET_VALU contiene el valor de respuesta RET_VAL de las instrucciones RD_REC y WR_REC. RET_VALU puede evaluarse si se notifica un error a través de QMOD_F.

Consulte también

RD_REC: Leer registro de periferia (Página 6769)

WR_REC: Escribir registro en periferia (Página 6774)

Parámetros de entrada/salida de DB 101 a 116

Tabla 9-286

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
1) Parámetros operativos				
SP_INT	128.0	REAL	0.0	El parámetro de entrada/salida "Consigna interna" sirve para preseleccionar una consigna mediante funciones de manejo y visualización.
SP_OP[1)]	132.0	REAL	0.0	El cuadro de diálogo "Puesta en servicio" tiene acceso al parámetro de entrada/salida "Control manual de consigna" Si el bit SP_OP_ON está activado, se aplica el valor "Control manual de consigna" como consigna.
LMN_OP[1)]	136.0	REAL	0.0	El cuadro de diálogo "Puesta en servicio" tiene acceso al parámetro de entrada/salida "Control manual de variable manipulada" Si el bit LMNOP_ON está activado, el valor "Control manual de variable manipulada" se aplica como variable manipulada. Se admiten valores entre -100 y 100 %.

Parámetro	Dirección	Tipo de datos	Ajuste predeterminado	Descripción
SP_OP_ON [1)]	140.0	BOOL	FALSE	El cuadro de diálogo "Puesta en servicio" tiene acceso al parámetro de entrada/salida "Activar control manual de consigna" Si el bit está activado, el valor SP_OP se aplica como consigna.
LMNOP_ON [1)]	140.1	BOOL	FALSE	El cuadro de diálogo "Puesta en servicio" tiene acceso al parámetro de entrada/salida "Activar control manual de la variable manipulada" Si el bit está activado, el valor LMN_OP se aplica como variable manipulada.
LOAD_PAR	140.2	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros de regulación en el FM 455" está activado, los parámetros de regulación se cargan en el módulo y el parámetro de entrada/salida se desactiva.
LOAD_OP	140.3	BOOL	FALSE	Si el parámetro de entrada/salida "Cargar parámetros operativos en el FM 455" está activado, los parámetros operativos se cargan en el módulo y el parámetro de entrada/salida se desactiva.

Nota

La EEPROM del módulo podría quedar destruida por un número excesivo de procesos de escritura. Para evitarlo, tras la escritura de la EEPROM el módulo retrasa una nueva escritura 30 minutos.

Funciones 300C

ANALOG: Posicionar con salida analógica

Descripción

Utilice la instrucción "ANALOG" para controlar las funciones de posicionamiento desde el programa de usuario.

Una salida analógica con asignación fija controla la etapa de potencia con una tensión (señal de tensión) entre ±10 V o una intensidad (señal de intensidad) de ±20 mA.

- Una vez concluida la fase de aceleración (RAM_UP) se efectúa primero una aproximación al destino a la velocidad (V_{nom}).
- En el punto de deceleración calculado por la CPU se inicia la deceleración (RAMP_DN) hasta el punto de cambio de velocidad.
- Al alcanzarse el punto de cambio de velocidad, se sigue avanzando a velocidad lenta (V_{lenta}).
- En el punto de desconexión se desconecta el accionamiento.

- El punto de cambio de velocidad y el punto de desconexión se establecen para cada destino que debe alcanzarse mediante los valores especificados en los parámetros (distancia de cambio de velocidad y distancia de desconexión). Es posible definir valores distintos para la distancia de cambio de velocidad y la distancia de desconexión según si el sentido de desplazamiento es hacia delante (positivo) o hacia atrás (negativo).
- El desplazamiento finaliza (WORKING = FALSE) cuando se alcanza el punto de desconexión. A partir de este momento puede iniciarse otro desplazamiento.
- El destino especificado se alcanza (POS_RCD = TRUE) cuando el valor real de posición alcanza la zona de destino. Si el valor real de posición abandona la zona de destino sin que se inicie otro desplazamiento, la señal "Posición alcanzada" no se desactiva.

Si la distancia de cambio de velocidad es menor que la distancia de desconexión, a partir del punto de deceleración se desacelera linealmente hasta la consigna de velocidad 0.

Parámetros básicos

Aquí se describen los parámetros de la instrucción que son idénticos para todos los modos de operación. Los parámetros específicos de cada modo de operación se describen en los distintos modos de operación.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "ANALOG":

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
LADDR	Input	WORD	0	según CPU	W#16#0310	Dirección E/S del submódulo que se ha definido en la configuración hardware. Si las direcciones E y S son diferentes, debe indicarse la menor de las dos direcciones.
CHANNEL	Input	INT	2	0	0	Número de canal
STOP	Input	BOOL	4.4	TRUE/FALSE	FALSE	Detener desplazamiento Si STOP = TRUE es posible finalizar/interrumpir el desplazamiento anticipadamente.
ERR_A	Input	BOOL	4.5	TRUE/FALSE	FALSE	Acuse de grupo error externo Con ERR_A = TRUE se acusan errores externos.

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
SPEED	Input	DINT	12	Velocidad lenta hasta 1 000 000 impulsos/s Como máximo hasta la velocidad máxima parametrizada	1000	El eje se acelera hasta la velocidad "V_{nom}". No es posible modificar la velocidad durante el desplazamiento.
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Desplazamiento en curso
ACT_POS	Output	DINT	18	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado
ERR	Output	WORD	24	Cada bit 0 ó 1	0	Error externo: Bit 2: vigilancia de marca cero Bit 11: vigilancia de la zona de desplazamiento (siempre 1) Bit 12: vigilancia de la zona de trabajo Bit 13: vigilancia del valor real Bit 14: vigilancia de aproximación a destino Bit 15: vigilancia de la zona de destino Bits restantes reservados
ST_ENBLD	Output	BOOL	26.0	TRUE/FALSE	TRUE	La CPU activa la habilitación de inicio cuando se cumplen todas las condiciones siguientes: • STOP no está presente (STOP = FALSE) • No hay ningún error externo (ERR = 0) • La habilitación del accionamiento está activada (DRV_EN = TRUE) • No hay ningún posicionamiento activo (WORKING = FALSE)
ERROR	Output	BOOL	26.1	TRUE/FALSE	FALSE	Error al iniciar/continuar un desplazamiento
STATUS	Output	WORD	28.0	W#16#0000 a W#16#FFFF	W#16#0000	Número de error

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
ACCEL	STATIC	DINT	30	de 1 a 100 000 impulsos/s^2	100	Aceleración No se puede modificar durante el desplazamiento.
DECEL	STATIC	DINT	34	de 1 a 100 000 impulsos/s^2	100	Deceleración No se puede modificar durante el desplazamiento.
CHGDIFF_P	STATIC	DINT	38	de 0 a $+10^8$ impulsos	1000	Distancia de cambio de velocidad positiva: La "distancia de cambio de velocidad positiva" define el punto de cambio de velocidad a partir del cual el accionamiento avanza a marcha lenta.
CUTOFF-DIFF_P	STATIC	DINT	42	de 0 a $+10^8$ impulsos	100	Distancia de desconexión positiva: La "distancia de desconexión positiva" define el punto de desconexión a partir del cual el accionamiento desconecta la marcha lenta al avanzar.
CHGDIFF_M	STATIC	DINT	46	de 0 a $+10^8$ impulsos	1000	Distancia de cambio de velocidad negativa: La "distancia de conmutación negativa" define el punto de cambio de velocidad a partir del cual el accionamiento retrocede a marcha lenta.
CUTOFF-DIFF_M	STATIC	DINT	50	de 0 a $+10^8$ impulsos	100	Distancia de desconexión negativa: La "distancia de desconexión negativa" define el punto a partir del cual el accionamiento desconecta la marcha lenta al retroceder.
PARA	STATIC	BOOL	54.0	TRUE/FALSE	FALSE	Eje parametrizado
DIR	STATIC	BOOL	54.1	TRUE/FALSE	FALSE	Sentido actual/último del movimiento FALSE = adelante (sentido positivo) TRUE = atrás (sentido negativo)
CUTOFF	STATIC	BOOL	54.2	TRUE/FALSE	FALSE	Accionamiento en la zona de desconexión (desde el punto de desconexión hasta iniciar el siguiente desplazamiento)

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
CHGOVER	STATIC	BOOL	54.3	TRUE/FALSE	FALSE	Accionamiento en la zona de cambio de velocidad (desde que se alcanza la velocidad lenta hasta que inicia el siguiente desplazamiento)
RAMP_DN	STATIC	BOOL	54.4	TRUE/FALSE	FALSE	El accionamiento se desacelera (desde el punto de deceleración hasta el punto de cambio de velocidad)
RAMP_UP	STATIC	BOOL	54.5	TRUE/FALSE	FALSE	El accionamiento se acelera (desde el inicio hasta que se alcanza la velocidad SPEED (V_{nom}))
DIST_TO_GO	STATIC	DINT	56	de $-5x10^8$ a $+5x10^8$ impulsos	0	Trayecto residual actual
LAST_TRG	STATIC	DINT	60	de $-5x10^8$ a $+5x10^8$ impulsos	0	Destino último/actual • Modo incremental absoluto: al iniciar el desplazamiento, LST_TRG = destino absoluto actual (TARGET). • Modo incremental relativo: al iniciar el desplazamiento, LST_TRG = LAST_TRG del desplazamiento anterior +/- el recorrido indicado (TARGET).

Parámetros del modo de operación "Jog"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Jog en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Jog en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 1 = Jog
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Desplazamiento en curso
ACT_POS	Output	DINT	18	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Modo de operación activo/ajustado

Parámetros del modo de operación "Búsqueda del punto de referencia"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Búsqueda del punto de referencia en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Búsqueda del punto de referencia en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 3 = "Búsqueda del punto de referencia"
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Desplazamiento en curso
SYNC	Output	BOOL	16.3	TRUE/FALSE	FALSE	SYNC = TRUE: el eje está sincronizado
ACT_POS	Output	DINT	18	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros del modo de operación "Modo incremental relativo"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Desplazamiento en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Desplazamiento en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 4 = Modo incremental relativo
TARGET	Input	DINT	8	de 0 a 10^9 impulsos	1000	Recorrido en impulsos (sólo se admiten valores positivos)
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Desplazamiento en curso
POS_RCD	Output	BOOL	16.1	TRUE/FALSE	FALSE	Posición alcanzada
ACT_POS	Output	DINT	18	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros del modo de operación "Modo incremental absoluto"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predetermina do	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
START	Input	BOOL	4.1	TRUE/FALSE	FALSE	Iniciar desplazamiento (flanco ascendente)
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Desplazamiento en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Desplazamiento en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 5 = Modo incremental absoluto
TARGET	Input	DINT	8	Eje lineal: de $-5x10^8$ a $+5x10^8$ Eje rotativo: de 0 a fin de eje rotativo -1	1000	Destino en impulsos
WORKING	Output	BOOL	16.0	TRUE/FALSE	FALSE	Desplazamiento en curso
POS_RCD	Output	BOOL	16.1	TRUE/FALSE	FALSE	Posición alcanzada
ACT_POS	Output	DINT	18	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	22	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros de la petición "Posicionar el punto de referencia"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predetermina do	Descripción
SYNC	Output	BOOL	16.3	TRUE/FALSE	FALSE	El eje está sincronizado

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predetermina do	Descripción
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE	FALSE	Petición errónea
JOB_ID	STATIC	INT	78	1, 2	0	Petición, 1 = "Posicionar el punto de referencia"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
JOB_STAT	STATIC	WORD	80	W#16#0000 a W#16#FFFF	W#16#0000	Número de error de petición
JOB_VAL	STATIC	DINT	82	de $5x10^8$ a $+5x10^8$ impulsos	0	Parámetros de la petición Coordenada del punto de referencia

Parámetros de la petición "Borrar trayecto residual"

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
JOB_REQ	STATIC	BOOL	76.0	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_DONE	STATIC	BOOL	76.1	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	STATIC	BOOL	76.2	TRUE/FALSE	FALSE	Petición errónea
JOB_ID	STATIC	INT	78	1, 2	0	Petición, 2 = "Borrar trayecto residual"
JOB_STAT	STATIC	WORD	80	W#16#0000 a W#16#FFFF	W#16#0000	Número de error de petición
JOB_VAL	STATIC	DINT	82	-	0	Ajuste cualquiera

Parámetros de la función "Medición de longitud"

La función se inicia mediante un flanco en la entrada digital. No hay parámetros específicos de entrada.

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
MSR_DONE	OUT	BOOL	16.2	TRUE/FALSE	FALSE	Medición de longitud finalizada

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
BEG_VAL	STATIC	DINT	64	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición para inicio de medición de longitud
END_VAL	STATIC	DINT	68	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición para fin de medición de longitud
LEN_VAL	STATIC	DINT	72	de 0 a 10^9 impulsos	0	Longitud medida

Parámetros ERROR y STATUS

Error del modo de operación (ERROR = TRUE)

Si se detecta un error, el parámetro de salida ERROR se pone a TRUE. En el parámetro STATUS se indica la causa del error.

Clase de evento código de error	Explicación
W#16#2002	Instrucción errónea, utilizar la instrucción "ANALOG"
W#16#2004	Número de canal erróneo (CHANNEL). Ajuste el número de canal "0".
W#16#3001	Petición de desplazamiento no aceptada por tarea errónea en la misma llamada. Corrija los parámetros del JOB correspondiente.
W#16#3002	No está permitido modificar Output _IN mientras el accionamiento está en marcha. Espere hasta que el posicionamiento haya terminado.
W#16#3003	Modo de operación desconocido (MODE_IN). Se admiten 1(Jog), 3 (Búsqueda del punto de referencia), 4 (Modo incremental relativo) y 5 (Modo incremental absoluto).
W#16#3004	No debe haber más de una solicitud de inicio activada al mismo tiempo. Las solicitudes de inicio permitidas son DIR_P o DIR_M o START.
W#16#3005	START sólo está permitido en el modo de operación "Modo incremental absoluto". Inicie el desplazamiento con DIR_P o DIR_M.
W#16#3006	DIR_P o DIR_M no está permitido en un eje lineal con el modo de operación "Modo incremental absoluto". Inicie el desplazamiento con START.
W#16#3007	El eje no está sincronizado. "Modo incremental absoluto" sólo es posible con el eje sincronizado.
W#16#3008	Abandonar la zona de trabajo. El desplazamiento retrocediendo a la zona de trabajo sólo está permitido en modo "Jog".
W#16#3101	No hay habilitación de inicio, puesto que el eje no está parametrizado. Parametrice el submódulo "Posicionar" en HW Config.
W#16#3102	No hay habilitación de inicio, puesto que la habilitación del accionamiento no está activada. Active la "habilitación del accionamiento" en la instrucción (DRV_EN=TRUE).
W#16#3103	No hay habilitación de inicio, puesto que STOP está activado. Borre STOP en la instrucción (STOP=FALSE)
W#16#3104	No hay habilitación de inicio, puesto que el eje se está posicionando (WORKING=TRUE). Espere hasta que el posicionamiento haya terminado.

Clase de evento código de error	Explicación
W#16#3105	No hay habilitación de inicio, puesto que todavía hay por lo menos un error externo no acusado. Elimine y acuse primero todos los errores externos y reinicie después el desplazamiento.
W#16#3202	Ajuste predeterminado de velocidad SPEED erróneo. El ajuste predeterminado de velocidad está fuera del rango admisible de velocidad lenta de hasta 1000000 impulsos/s. Pero como máximo hasta la velocidad máxima parametrizada.
W#16#3203	El ajuste predeterminado de aceleración ACCEL está fuera del rango admisible de 1 a 100000 impulsos/s^2.
W#16#3204	El ajuste predeterminado de deceleración DECEL está fuera del rango admisible de 1 a 100000 impulsos/s^2.
W#16#3206	El ajuste predeterminado de velocidad SPEED debe ser mayor o igual a la frecuencia parametrizada de búsqueda de referencia.
W#16#3301	Distancia de cambio de velocidad/desconexión demasiado grande. Especificar una distancia de cambio de velocidad/desconexión de 10^8 como máximo.
W#16#3304	Distancia de desconexión insuficiente. La distancia de desconexión debe ser como mínimo igual a la mitad de la zona de destino.
W#16#3305	Distancia de cambio de velocidad insuficiente. La distancia de cambio de velocidad debe ser como mínimo igual a la mitad de la zona de destino.
W#16#3401	Destino especificado fuera de la zona de trabajo. En un eje lineal con modo incremental absoluto, el destino especificado debe encontrarse entre los finales de carrera por software (éstos inclusive).
W#16#3402	Destino especificado erróneo. En un eje rotativo, el destino especificado debe ser mayor que 0 y menor que el fin del eje rotativo.
W#16#3403	Indicación de recorrido errónea. El trayecto que debe recorrerse en modo incremental relativo debe ser positivo.
W#16#3404	Indicación de recorrido errónea. La coordenada de destino absoluta resultante debe ser mayor que $-5x10^8$.
W#16#3405	Indicación de recorrido errónea. La coordenada de destino absoluta resultante debe ser menor que $5x10^8$.
W#16#3406	Indicación de recorrido errónea. La coordenada de destino absoluta resultante debe encontrarse dentro de la zona de trabajo (+/- la mitad de la zona de destino).
W#16#3501	Zona de desplazamiento demasiado grande. La coordenada de destino + el trayecto residual actual debe ser mayor/igual a $-5x10^8$.
W#16#3502	Zona de desplazamiento demasiado grande. La coordenada de destino + el trayecto residual actual debe ser menor/igual a $5x10^8$.
W#16#3503	Recorrido de desplazamiento insuficiente. El recorrido de desplazamiento en sentido positivo debe ser mayor que la distancia de desconexión indicada para el sentido positivo.
W#16#3504	Recorrido de desplazamiento insuficiente. El recorrido de desplazamiento en sentido negativo debe ser mayor que la distancia de desconexión indicada para el sentido negativo.
W#16#3505	Recorrido de desplazamiento insuficiente o final de carrera sobrepasado en sentido positivo. El último destino alcanzable en sentido positivo (límite de la zona de trabajo o de la zona de desplazamiento) está demasiado cerca de la posición actual.
W#16#3506	Recorrido de desplazamiento insuficiente o final de carrera sobrepasado en sentido negativo. El último destino alcanzable en sentido negativo (límite de la zona de trabajo o de la zona de desplazamiento) está demasiado cerca de la posición actual.

Error de petición (JOB_ERR = TRUE)

Si se detecta un error, el parámetro de salida JOB_ERR se pone a TRUE. En el parámetro JOB_STAT se indica la causa del error.

Clase de evento código de error	Explicación
W#16#4001	El eje no está parametrizado. Parametrice el submódulo "Posicionar" en HW Config.
W#16#4002	La petición no es posible, pues hay un posicionamiento activo. Espere hasta que WORKING = FALSE y vuelva a ejecutar la petición.
W#16#4004	Petición desconocida. Compruebe el número de petición y ejecute nuevamente la petición.
W#16#4101	En un eje lineal, la coordenada del punto de referencia no puede estar fuera de los límites de la zona de trabajo.
W#16#4102	En un eje lineal, la coordenada indicada para el punto de referencia + el trayecto residual actual todavía debe ser mayor/igual a $-5x10^8$.
W#16#4103	En un eje lineal, la coordenada indicada para el punto de referencia + el trayecto residual actual todavía debe ser menor/igual a $5x10^8$.
W#16#4104	En un eje lineal, la coordenada indicada para el punto de referencia + la distancia actual del punto inicial del desplazamiento todavía debe ser mayor/igual a $-5x10^8$.
W#16#4105	En un eje lineal, la coordenada indicada para el punto de referencia + la distancia actual del punto inicial del desplazamiento todavía debe ser menor/igual a $5x10^8$.
W#16#4106	En un eje rotativo, la coordenada del punto de referencia no puede ser menor que 0 ni mayor/igual al fin del eje rotativo.

Error externo (ERR)

La tecnología vigila el desplazamiento, la zona de desplazamiento y la periferia conectada. Para ello, es imprescindible haber conectado previamente las respectivas vigilancias en las máscaras de parametrización "Accionamiento", "Eje" y "Encoder".

Cuando hay una respuesta de las vigilancias se notifica un error externo. Los errores externos pueden producirse en cualquier momento, independientemente de las funciones iniciadas. Dichos errores deben acusarse siempre con ERR_A = TRUE.

Los errores externos se indican en el parámetro ERR (WORD) mediante la activación de un bit.

Vigilancia	Código de error	Bit en ERR-WORD
Error en señal (señal cero)	W#16#0004	2
Zona de desplazamiento	W#16#0800	11
Zona de trabajo	W#16#1000	12
Valor real	W#16#2000	13
Aproximación a destino	W#16#4000	14
Zona de destino	W#16#8000	15

Errores de sistema

Un error de sistema se señaliza con BR = FALSE. Es originado por un error al escribir/leer el DB de instancia o bien por una llamada múltiple de la instrucción.

DIGITAL: Posicionar con salida digital

Descripción

Utilice la instrucción "DIGITAL" para controlar las funciones de posicionamiento desde el programa de usuario.

Cuatro salidas digitales de 24 V fijamente asignadas controlan la etapa de potencia (para más información, consulte la documentación del hardware utilizado). Según el tipo de control parametrizado, la salidas digitales controlan el sentido y las marchas de velocidad (marcha rápida/lenta).

La lectura de recorrido se realiza mediante un encoder incremental asimétrico de 24 V con dos señales desfasadas en 90 grados.

- Primero se efectúa una aproximación al destino a la velocidad ($V_{rápida}$).
- En el punto de cambio de velocidad se cambia a la velocidad lenta (V_{lenta}).
- En el punto de desconexión se desconecta el accionamiento.
- El punto de cambio de velocidad y el punto de desconexión se establecen con los valores de los parámetros CHGDIFF_P/M y CUTOFFDIFF_P/M. Es posible definir valores distintos para la distancia de cambio de velocidad y la distancia de desconexión según si el sentido de desplazamiento es hacia delante (positivo) o hacia atrás (negativo).
- El desplazamiento finaliza (WORKING = FALSE) cuando se alcanza el punto de desconexión. A partir de este momento puede iniciarse otro desplazamiento.
- El destino especificado se alcanza (POS_RCD = TRUE) cuando el valor real de posición alcanza la zona de destino. Si el valor real de posición abandona la zona de destino sin que se inicie otro desplazamiento, la señal "Posición alcanzada" no se desactiva.

Parámetros básicos:

Aquí se describen los parámetros de la instrucción que son idénticos para todos los modos de operación. Los parámetros específicos de cada modo de operación se describen en los distintos modos de operación.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
LADDR	Input	WORD	0	según CPU	W#16#0310	Dirección E/S del submódulo que se ha definido en la configuración hardware. Si las direcciones de E y S son diferentes, debe indicarse la menor de las dos direcciones.
CHANNEL	Input	INT	2	0	0	Número de canal
STOP	Input	BOOL	4.4	TRUE/FALSE	FALSE	Detener desplazamiento Si STOP = TRUE es posible finalizar/interrumpir el desplazamiento anticipadamente.
ERR_A	Input	BOOL	4.5	TRUE/FALSE	FALSE	Acuse de grupo error externo Con ERR_A = TRUE se acusan errores externos.
SPEED	Input	BOOL	12.0	TRUE/FALSE	FALSE	Dos marchas de velocidad para marcha rápida/lenta TRUE=marcha rápida FALSE=marcha lenta
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Desplazamiento en curso
ACT_POS	Output	DINT	16	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Modo de operación activo/ajustado
ERR	Output	WORD	22	Cada bit 0 ó 1	0	Error externo • Bit2: vigilancia de marca cero • Bit11: vigilancia de la zona de desplazamiento (siempre 1) • Bit12: vigilancia de la zona de trabajo • Bit13: vigilancia del valor real • Bit14: vigilancia de aproximación a destino • Bit15: vigilancia de la zona de destino Bits restantes reservados

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predetermina do	Descripción
ST_ENBLD	Output	BOOL	24.0	TRUE/FALSE	TRUE	La CPU activa la habilitación de inicio cuando se cumplen todas las condiciones siguientes: • STOP no está presente (STOP = FALSE) • No hay ningún error externo (ERR = 0) • La habilitación del accionamiento está activada (DRV_EN = TRUE) • No hay ningún posicionamiento activo (WORKING = FALSE)
ERROR	Output	BOOL	24.1	TRUE/FALSE	FALSE	Error al iniciar/continuar un desplazamiento
STATUS	Output	WORD	26.0	W#16#0000 a W#16#FFFF	W#16#0000	Número de error

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
CHGDIFF_P	STATIC	DINT	28	de 0 a $+10^8$ impulsos	1000	Distancia de cambio de velocidad positiva: La "distancia de cambio de velocidad positiva" define el punto en el que el accionamiento cambia de marcha rápida a marcha lenta avanzando.
CUTOFF-DIFF_P	STATIC	DINT	32	de 0 a $+10^8$ impulsos	100	Distancia de desconexión positiva: La "distancia de desconexión positiva" define el punto a partir del cual el accionamiento desconecta la marcha lenta al avanzar.
CHGDIFF_M	STATIC	DINT	36	de 0 a $+10^8$ impulsos	1000	Distancia de cambio de velocidad negativa: La "distancia de cambio de velocidad negativa" define el punto en el que el accionamiento cambia de marcha rápida a marcha lenta retrocediendo.

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
CUTOFF-DIFF_M	STATIC	DINT	40	de 0 a $+10^8$ impulsos	100	Distancia de desconexión negativa: La "distancia de desconexión negativa" define el punto de desconexión a partir del cual el accionamiento desconecta la marcha lenta al retroceder.
PARA	STATIC	BOOL	44.0	TRUE/FALSE	FALSE	Eje parametrizado
DIR	STATIC	BOOL	44.1	TRUE/FALSE	FALSE	Sentido actual/último del movimiento FALSE = adelante (sentido positivo) TRUE = atrás (sentido negativo)
CUTOFF	STATIC	BOOL	44.2	TRUE/FALSE	FALSE	Accionamiento en la zona de desconexión (desde el punto de desconexión hasta iniciar el siguiente desplazamiento)
CHGOVER	STATIC	BOOL	44.3	TRUE/FALSE	FALSE	Accionamiento en la zona de cambio de velocidad (desde que se alcanza la velocidad lenta hasta que se inicia el siguiente desplazamiento)
DIST_TO_GO	STATIC	DINT	46	de $-5x10^8$ a $+5x10^8$ impulsos	0	Trayecto residual actual
LAST_TRG	STATIC	DINT	50	de $-5x10^8$ a $+5x10^8$ impulsos	0	Destino último/actual • Modo incremental absoluto: al iniciar el desplazamiento, LST_TRG = destino absoluto actual (TARGET). • Modo incremental relativo: al iniciar el desplazamiento, LST_TRG = LAST_TRG del desplazamiento anterior +/- el recorrido indicado (TARGET).

Parámetros del modo de operación "Jog"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Jog en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Jog en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 1 = Jog
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Desplazamiento en curso
ACT_POS	Output	DINT	16	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros del modo de operación "Búsqueda del punto de referencia"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Búsqueda del punto de referencia en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Búsqueda del punto de referencia en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 3 = "Búsqueda del punto de referencia"
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Desplazamiento en curso
SYNC	Output	BOOL	14.3	TRUE/FALSE	FALSE	SYNC = TRUE: el eje está sincronizado
ACT_POS	Output	DINT	16	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros del modo de operación "Modo incremental relativo"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Desplazamiento en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Desplazamiento en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 4 = Modo incremental relativo
TARGET	Input	DINT	8	de 0 a 10^9 impulsos	1000	Recorrido en impulsos (sólo se admiten valores positivos)
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Desplazamiento en curso
POS_RCD	Output	BOOL	14.1	TRUE/FALSE	FALSE	Posición alcanzada
ACT_POS	Output	DINT	16	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros del modo de operación "Modo incremental absoluto"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
DRV_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Habilitación del accionamiento
START	Input	BOOL	4.1	TRUE/FALSE	FALSE	Iniciar desplazamiento (flanco ascendente)
DIR_P	Input	BOOL	4.2	TRUE/FALSE	FALSE	Desplazamiento en sentido positivo (flanco ascendente)
DIR_M	Input	BOOL	4.3	TRUE/FALSE	FALSE	Desplazamiento en sentido negativo (flanco ascendente)
MODE_IN	Input	INT	6	0, 1, 3, 4, 5	1	Modo de operación, 5 = Modo incremental absoluto
TARGET	Input	DINT	8	Eje lineal: de $-5x10^8$ a $+5x10^8$ Eje rotativo: de 0 a fin de eje rotativo -1	1000	Destino en impulsos
WORKING	Output	BOOL	14.0	TRUE/FALSE	FALSE	Desplazamiento en curso
POS_RCD	Output	BOOL	14.1	TRUE/FALSE	FALSE	Posición alcanzada

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
ACT_POS	Output	DINT	16	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición actual
MODE_OUT	Output	INT	20	0, 1, 3, 4, 5	0	Modo de operación activo/ ajustado

Parámetros de la petición "Posicionar el punto de referencia"

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
SYNC	Output	BOOL	14.3	TRUE/FALSE	FALSE	El eje está sincronizado

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE	FALSE	Petición errónea
JOB_ID	STATIC	INT	68	1, 2	0	Petición, 1 = "Posicionar el punto de referencia"
JOB_STAT	STATIC	WORD	70	W#16#0000 a W#16#FFFF	W#16#0000	Número de error de petición
JOB_VAL	STATIC	DINT	72	de $-5x10^8$ a $+5x10^8$ impulsos	0	Parámetros de la petición Coordenada del punto de referencia

Parámetros de la petición "Borrar trayecto residual"

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
JOB_REQ	STATIC	BOOL	66.0	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_DONE	STATIC	BOOL	66.1	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	STATIC	BOOL	66.2	TRUE/FALSE	FALSE	Petición errónea

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
JOB_ID	STATIC	INT	68	1, 2	0	Petición, 2 = "Borrar trayecto residual"
JOB_STAT	STATIC	WORD	70	de 0 a FFFF hex	0	Número de error de petición
JOB_VAL	STATIC	DINT	72	-	0	Ninguna

Parámetros de la función "Medición de longitud"

La función se inicia mediante un flanco en la entrada digital. No hay parámetros específicos de entrada.

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
MSR_DONE	Output	BOOL	14.2	TRUE/FALSE	FALSE	Medición de longitud finalizada

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
BEG_VAL	STATIC	DINT	54	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición para inicio de medición de longitud
END_VAL	STATIC	DINT	58	de $-5x10^8$ a $+5x10^8$ impulsos	0	Valor real de posición para fin de medición de longitud
LEN_VAL	STATIC	DINT	62	de 0 a 10^9 impulsos	0	Longitud medida

Parámetros ERROR y STATUS

Si se detecta un error, el parámetro de salida ERROR se pone a TRUE. En el parámetro STATUS se indica la causa del error.

Código de error STATUS	Explicación
W#16#2001	Instrucción errónea, utilizar la instrucción "DIGITAL".
W#16#2004	Número de canal erróneo (CHANNEL). Ajuste el número de canal "0".
W#16#3001	Petición de desplazamiento no aceptada por tarea errónea en la misma llamada. Corrija los parámetros del JOB correspondiente.
W#16#3002	No está permitido modificar MODE_IN mientras el accionamiento está activo. Espere hasta que el posicionamiento haya terminado.
W#16#3003	Modo de operación desconocido (MODE_IN). Se admiten 1(Jog), 3 (Búsqueda del punto de referencia), 4 (Modo incremental relativo) y 5 (Modo incremental absoluto).
W#16#3004	No debe haber más de una solicitud de inicio activada al mismo tiempo. Las solicitudes de inicio permitidas son DIR_P o DIR_M o START.

Código de error STATUS	Explicación
W#16#3005	START sólo está permitido en el modo de operación "Modo incremental absoluto". Inicie el desplazamiento con DIR_P o DIR_M.
W#16#3006	DIR_P o DIR_M no está permitido en un eje lineal con el modo de operación "Modo incremental absoluto". Inicie el desplazamiento con START.
W#16#3007	El eje no está sincronizado. "Modo incremental absoluto" sólo es posible con el eje sincronizado.
W#16#3008	Abandonar la zona de trabajo. El desplazamiento retrocediendo a la zona de trabajo sólo está permitido en modo "Jog".
W#16#3101	No hay habilitación de inicio, puesto que el eje no está parametrizado. Parametrice el submódulo "Posicionar" en HW Config.
W#16#3102	No hay habilitación de inicio, puesto que la habilitación del accionamiento no está activada. Active la "habilitación del accionamiento" en la instrucción (DRV_EN=TRUE).
W#16#3103	No hay habilitación de inicio, puesto que STOP está activado. Borre STOP de la instrucción (STOP=FALSE).
W#16#3104	No hay habilitación de inicio, puesto que el eje se está posicionando (WORKING=TRUE). Espere hasta que el posicionamiento haya terminado.
W#16#3105	No hay habilitación de inicio, puesto que todavía hay por lo menos un error externo no acusado. Elimine y acuse primero todos los errores externos y reinicie después el desplazamiento.
W#16#3201	Ajuste predeterminado de velocidad SPEED erróneo. En el posicionamiento con salidas digitales, sólo se admite "marcha lenta" (0) y "marcha rápida" (1).
W#16#3301	Distancia de cambio de velocidad/desconexión demasiado grande. Determinar una distancia de cambio de velocidad/desconexión igual a 10^8 como máximo.
W#16#3303	Distancia de cambio de velocidad insuficiente. La distancia de cambio de velocidad debe ser mayor o igual a la distancia de desconexión.
W#16#3304	Distancia de desconexión insuficiente. La distancia de desconexión debe ser como mínimo igual a la mitad de la zona de destino.
W#16#3401	Destino especificado fuera de la zona de trabajo. En un eje lineal con modo incremental absoluto, el destino especificado debe encontrarse entre los finales de carrera por software (éstos inclusive).
W#16#3402	Destino especificado erróneo. En un eje rotativo, el destino especificado debe ser mayor que "0" y menor que el fin del eje rotativo.
W#16#3403	Indicación de recorrido errónea. El trayecto que debe recorrerse en modo incremental relativo debe ser positivo.
W#16#3404	Indicación de recorrido errónea. La coordenada de destino absoluta resultante debe ser mayor que $-5x10^8$.
W#16#3405	Indicación de recorrido errónea. La coordenada de destino absoluta resultante debe ser menor que $5x10^8$.
W#16#3406	Indicación de recorrido errónea. La coordenada de destino absoluta resultante debe encontrarse dentro de la zona de trabajo (+/- la mitad de la zona de destino).
W#16#3501	Recorrido de desplazamiento demasiado grande. La coordenada de destino + el trayecto residual actual debe ser mayor/igual a $-5x10^8$.
W#16#3502	Recorrido de desplazamiento demasiado grande. La coordenada de destino + el trayecto residual actual debe ser menor/igual a $5x10^8$.
W#16#3503	Recorrido de desplazamiento insuficiente. El recorrido de desplazamiento en sentido positivo debe ser mayor que la distancia de desconexión indicada para el sentido positivo.

Código de error STATUS	Explicación
W#16#3504	Recorrido de desplazamiento insuficiente. El recorrido de desplazamiento en sentido negativo debe ser mayor que la distancia de desconexión indicada para el sentido negativo.
W#16#3505	Recorrido de desplazamiento insuficiente o final de carrera sobrepasado en sentido positivo. El último destino alcanzable en sentido positivo (límite de la zona de trabajo o de la zona de desplazamiento) está demasiado cerca de la posición actual.
W#16#3506	Recorrido de desplazamiento insuficiente o final de carrera sobrepasado en sentido negativo. El último destino alcanzable en sentido negativo (límite de la zona de trabajo o de la zona de desplazamiento) está demasiado cerca de la posición actual.

Error de petición (JOB_ERR = TRUE)

Si se detecta un error, el parámetro JOB_ERR se pone a TRUE. En el parámetro JOB_STAT se indica la causa del error.

Clase de evento código de error	Explicación
W#16#4001	El eje no está parametrizado. Parametrice el submódulo "Posicionar" en HW Config.
W#16#4002	La petición no es posible, pues hay un posicionamiento activo. Las peticiones sólo pueden ejecutarse si no hay ningún posicionamiento activo. Espere hasta que WORKING = FALSE y vuelva a ejecutar la petición.
W#16#4004	Petición desconocida. Compruebe el número de petición y ejecute nuevamente la petición.
W#16#4101	En un eje lineal, la coordenada del punto de referencia no puede estar fuera de los límites de la zona de trabajo.
W#16#4102	En un eje lineal, la coordenada indicada para el punto de referencia + el trayecto residual actual todavía debe ser mayor/igual a $-5x10^8$.
W#16#4103	En un eje lineal, la coordenada indicada para el punto de referencia + el trayecto residual actual todavía debe ser menor/igual a $5x10^8$.
W#16#4104	En un eje lineal, la coordenada indicada para el punto de referencia + la distancia actual del punto inicial del desplazamiento todavía debe ser mayor/igual a $-5x10^8$.
W#16#4105	En un eje lineal, la coordenada indicada para el punto de referencia + la distancia actual del punto inicial del desplazamiento todavía debe ser menor/igual a $5x10^8$.
W#16#4106	En un eje rotativo, la coordenada del punto de referencia no puede ser menor que 0 ni mayor/igual al fin del eje rotativo.

Error externo (ERR)

La tecnología vigila el desplazamiento, la zona de desplazamiento y la periferia conectada. Para ello, es imprescindible haber conectado previamente las respectivas vigilancias en las máscaras de parametrización "Accionamiento", "Eje" y "Encoder".

Cuando hay una respuesta de las vigilancias se notifica un error externo. Los errores externos pueden producirse en cualquier momento, independientemente de las funciones iniciadas. Dichos errores deben acusarse siempre con ERR_A = TRUE.

Los errores externos se indican en el parámetro ERR (WORD) mediante la activación de un bit.

Vigilancia	Código de error	Bit en ERR-WORD
Error en señal (señal cero)	W#16#0004	2
Zona de desplazamiento	W#16#0800	11
Zona de trabajo	W#16#1000	12
Valor real	W#16#2000	13
Aproximación a destino	W#16#4000	14
Zona de destino	W#16#8000	15

Errores de sistema

Un error de sistema se señaliza con BR = FALSE. Es originado por un error al escribir/leer el DB de instancia o por una llamada múltiple de la instrucción.

COUNT: Controlar contador

Descripción

Utilice la instrucción "COUNT" para controlar el contador desde el programa de usuario.

Se dispone de la funcionalidad siguiente:

- Iniciar/parar el contador con la puerta de software SW_GATE
- Habilitar/controlar la salida DO
- Leer los bits de estado STS_CMP, STS_OFLW, STS_UFLW y STS_ZP
- Leer el estado actual del contador COUNTVAL
- Peticiones para leer y escribir los registros de contaje internos
- Leer el período actual TIMEVAL

Parámetros

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predetermina do	Descripción
LADDR	Input	WORD	0	según CPU	W#16#0300	Dirección E/S del submódulo que se ha definido en la configuración hardware. Si las direcciones de entrada y salida son diferentes, debe indicarse la menor de las dos direcciones.
CHANNEL	Input	INT	2	CPU 312C: de 0 a 1 CPU 313C: de 0 a 2 CPU 314C: de 0 a 3	0	Número de canal

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
SW_GATE	Input	BOOL	4.0	TRUE/FALSE	FALSE	Puerta de software para iniciar/parar el contador
CTRL_DO	Input	BOOL	4.1	TRUE/FALSE	FALSE	Habilitar salida
SET_DO	Input	BOOL	4.2	TRUE/FALSE	FALSE	Controlar salida
JOB_REQ	Input	BOOL	4.3	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_ID	Input	WORD	6	W#16#0000 Petición sin función W#16#0001 Escribir valor de contaje W#16#0002 Escribir valor de carga W#16#0004 Escribir valor de comparación W#16#0008 Escribir histéresis W#16#0010 Escribir duración de impulso W#16#0082 Leer valor de carga W#16#0084 Leer valor de comparación W#16#0088 Leer histéresis W#16#0090 Leer duración de impulso	W#16#0000	Número de petición
JOB_VAL	Input	DINT	8	-2^{31} a $+2^{31}$ -1	0	Valor para peticiones de escritura.
STS_GATE	Output	BOOL	12.0	TRUE/FALSE	FALSE	Estado de puerta interna
STS_STRT	Output	BOOL	12.1	TRUE/FALSE	FALSE	Estado de puerta hardware (entrada de inicio)
STS_LTCH	Output	BOOL	12.2	TRUE/FALSE	FALSE	Estado de entrada de congelación
STS_DO	Output	BOOL	12.3	TRUE/FALSE	FALSE	Estado de salida
STS_C_DN	Output	BOOL	12.4	TRUE/FALSE	FALSE	Estado de sentido hacia atrás. Se indica siempre el último sentido de contaje. Después de la primera llamada de la instrucción, STS_C_DN tiene el valor FALSE.

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predetermina do	Descripción
STS_C_UP	Output	BOOL	12.5	TRUE/FALSE	FALSE	Estado de sentido hacia delante Se indica siempre el último sentido de contaje. Después de la primera llamada de la instrucción, STS_C_UP tiene el valor TRUE
COUNTVAL	Output	DINT	14	-2^{31} a $+2^{31}$ -1	0	Valor actual de contaje
LATCHVAL	Output	DINT	18	-2^{31} a $+2^{31}$ -1	0	Valor actual de congelación
JOB_DONE	Output	BOOL	22.0	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	Output	BOOL	22.1	TRUE/FALSE	FALSE	Petición errónea
JOB_STAT	Output	WORD	24	de 0 a W#16#FFFF	0	Número de error de petición

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se ha ajustado el parámetro "Comportamiento de la salida" (que se ajusta mediante la interfaz de parametrización) a "Sin comparación", rige lo siguiente:

- la salida se conecta como una salida normal.
- los parámetros de entrada CTRL_DO y SET_DO no son efectivos.
- Los bits de estado STS_DO y STS_CMP (estado del comparador en el IDB) permanecen desactivados

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
STS_CMP	STATIC	BOOL	26.3	TRUE/FALSE	FALSE	Estado del comparador. Se desactiva con RES_STS. El bit de estado STS_CMP indica que la condición de comparación del comparador se cumple o se cumplía. Con STS_CMP también se indica que la salida estaba activada (STS_DO = TRUE)
STS_OFLW	STATIC	BOOL	26.5	TRUE/FALSE	FALSE	Estado de rebase por exceso Se desactiva con RES_STS.
STS_UFLW	STATIC	BOOL	26.6	TRUE/FALSE	FALSE	Estado de rebase por defecto Se desactiva con RES_STS.
STS_ZP	STATIC	BOOL	26.7	TRUE/FALSE	FALSE	Estado de paso por cero Se desactiva con RES_STS. Sólo se activa en contajes sin sentido normal de contaje. Indica el paso por cero. También se activa si el contador se pone a 0 o si el contador cuenta a partir del valor de carga=0.
JOB_OVAL	STATIC	DINT	28	-2^{31} a $+2^{31}-1$	0	Valor de salida para peticiones de lectura.
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE	FALSE	Desactivar bits de estado. Desactiva los bits de estado STS_CMP, STS_OFLW, STS_UFLW y STS_ZP. Para desactivar los bits de estado se requieren dos llamadas de la instrucción.

Parámetros JOB_ERR y JOB_STAT

Si se ha producido un error de petición, se pone JOB_ERR = TRUE. La causa exacta del error se indica en JOB_STAT.

Clase de evento código de error	Explicación
W#16#0121	Valor de comparación demasiado bajo.
W#16#0122	Valor de comparación demasiado alto.
W#16#0131	Histéresis insuficiente.
W#16#0132	Histéresis excesiva.
W#16#0141	Duración de impulso insuficiente.
W#16#0142	Duración de impulso excesiva.
W#16#0151	Valor de carga demasiado bajo.

Clase de evento código de error	Explicación
W#16#0152	Valor de carga demasiado alto.
W#16#0161	Estado de contaje demasiado bajo.
W#16#0162	Estado de contaje demasiado alto.
W#16#01FF	Número de petición no válido.

Errores de sistema

Si se ha producido un error de sistema, BR se pone = FALSE.

Clase de evento código de error	Explicación
W#16#8001	Modo de operación erróneo o error de parametrización. Ajuste el modo de operación correcto en la configuración hardware o utilice la instrucción adecuada al modo de operación ajustado.
W#16#8009	Número de canal no válido. Ajuste un número de canal ≤ 3 (valor específico de la CPU).

FREQUENC: Controlar frecuencímetro

Descripción

Utilice la instrucción "FREQUENC" para manejar el frecuencímetro desde el programa de usuario.

Se dispone de la funcionalidad siguiente:

- Iniciar/parar con la puerta de software SW_GATE
- Habilitar/controlar la salida DO
- Leer los bits de estado STS_CMP, STS_OFLW y STS_UFLW
- Leer el valor de frecuencia actual MEAS_VAL
- Peticiones para leer y escribir los registros internos de medida de frecuencia

Parámetros

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
LADDR	Input	WORD	0	según CPU	W#16#0300	Dirección E/S del submódulo que se ha definido en la configuración hardware. Si las direcciones E y S son diferentes, debe indicarse la menor de las dos direcciones.
CHANNEL	Input	INT	2	CPU 312C: de 0 a 1 CPU 313C: de 0 a 2 CPU 314C: de 0 a 3	0	Número de canal
SW_GATE	Input	BOOL	4.0	TRUE/FALSE	FALSE	Puerta de software para iniciar/parar la medida de frecuencia
MAN_DO	Input	BOOL	4.1	TRUE/FALSE	FALSE	Habilitar el control manual de la salida
SET_DO	Input	BOOL	4.2	TRUE/FALSE	FALSE	Controlar salida
JOB_REQ	Input	BOOL	4.3	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_ID	Input	WORD	6	W#16#0000 = petición sin función W#16#0001 = escribir límite inferior W#16#0002 = escribir límite superior W#16#0004 = escribir tiempo de integración W#16#0081 = leer límite inferior W#16#0082 = leer límite superior W#16#0084 = leer tiempo de integración	0	Número de petición
JOB_VAL	Input	DINT	8	-2^{31} a $+2^{31}-1$	0	Valor para peticiones de escritura
STS_GATE	Output	BOOL	12.0	TRUE/FALSE	FALSE	Estado de puerta interna
STS_STRT	Output	BOOL	12.1	TRUE/FALSE	FALSE	Estado de puerta hardware (entrada de inicio)

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
STS_DO	Output	BOOL	12.2	TRUE/FALSE	FALSE	Estado de salida
STS_C_DN	Output	BOOL	12.3	TRUE/FALSE	FALSE	Estado de sentido hacia atrás Se indica siempre el último sentido de contaje. Después de la primera llamada de la instrucción, STS_C_DN tiene el valor FALSE.
STS_C_UP	Output	BOOL	12.4	TRUE/FALSE	FALSE	Estado de sentido hacia delante Se indica siempre el último sentido de contaje. Después de la primera llamada de la instrucción, STS_C_UP tiene el valor TRUE.
MEAS_VAL	Output	DINT	14	de 0 a $+2^{31}$ -1	0	Valor de frecuencia actual
COUNTVAL	Output	DINT	18	-2^{31} a $+2^{31}$ -1	0	Valor de contaje actual (empieza por 0 cada vez que se abre la puerta)
JOB_DONE	Output	BOOL	22.0	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	Output	BOOL	22.1	TRUE/FALSE	FALSE	Petición errónea
JOB_STAT	Output	WORD	24	W#16#0000 a W#16#FFFF	W#16#0000	Número de error de petición

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Si se ha ajustado el parámetro "Comportamiento de la salida" (que se ajusta mediante la interfaz de parametrización) a "Sin comparación", rige lo siguiente:

- la salida se conecta como una salida normal.
- los parámetros de entrada MAN_DO y SET_DO no son efectivos.
- el bit de estado STS_DO permanece desactivado.

Parámetros no interconectados en el bloque (datos locales estáticos sin los parámetros reservados RESxx):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
STS_CMP	STATIC	BOOL	26.3	TRUE/FALSE	FALSE	Estado de fin de medición Se desactiva con RES_STS . El valor medido se actualiza cada vez que finaliza un intervalo. El fin de una medición se notifica con el bit de estado STS_CMP
STS_OFLW	STATIC	BOOL	26.5	TRUE/FALSE	FALSE	Estado de rebase por exceso Se desactiva con RES_STS.
STS_UFLW	STATIC	BOOL	26.6	TRUE/FALSE	FALSE	Estado de rebase por defecto Se desactiva con RES_STS.
JOB_OVAL	STATIC	DINT	28	-2^{31} a 2^{31} -1	0	Valor de salida para peticiones de lectura
RES_STS	STATIC	BOOL	32.2	TRUE/FALSE	FALSE	Desactivar bits de estado. Desactiva los bits de estado STS_CMP, STS_OFLW y STS_UFLW. Para desactivar los bits de estado se requieren dos llamadas de la instrucción.

Parámetros JOB_ERR y JOB_STAT

Si se ha producido un error de petición, se pone JOB_ERR = TRUE. La causa exacta del error se indica en JOB_STAT.

Clase de evento código de error	Explicación
W#16#0221	Tiempo de integración demasiado corto.
W#16#0222	Tiempo de integración demasiado largo.
W#16#0231	Límite inferior de frecuencia demasiado bajo.
W#16#0232	Límite inferior de frecuencia demasiado alto.
W#16#0241	Límite superior de frecuencia demasiado bajo.
W#16#0242	Límite superior de frecuencia demasiado alto.
W#16#02FF	Número de petición no válido.

Errores de sistema

Si se ha producido un error de sistema, BR se pone = FALSE.

Clase de evento código de error	Explicación
W#16#8001	Modo de operación erróneo o error de parametrización. Ajuste el modo de operación correcto en la configuración hardware o utilice la instrucción adecuada al modo de operación ajustado
W#16#8009	Número de canal no válido. Ajuste un número de canal ≤ 3 (valor específico de la CPU).

PULSE: Controlar modulación por ancho de impulso

Descripción

Utilice la instrucción "PULSE" para controlar la modulación por ancho de impulso desde el programa de usuario.

Se dispone de la funcionalidad siguiente:

- Iniciar/parar con la puerta de software SW_EN
- Habilitar/controlar la salida DO
- Leer los bits de estado STS_EN, STS_STRT y STS_DO
- Introducir el valor de salida
- Peticiones para leer y escribir los registros

Parámetros

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Significado
LADDR	Input	WORD	0	según CPU	W#16#0300	Dirección E/S del submódulo que se ha definido en la configuración hardware. Si las direcciones E y S son diferentes, debe indicarse la menor de las dos direcciones.
CHANNEL	Input	INT	2	CPU 312C: de 0 a 1 CPU 313C: de 0 a 2 CPU 314C: de 0 a 3	0	Número de canal
SW_EN	Input	BOOL	4.0	TRUE/FALSE	FALSE	Puerta de software para iniciar/parar la salida
MAN_DO	Input	BOOL	4.1	TRUE/FALSE	FALSE	Habilitar el control manual de la salida

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Significado
SET_DO	Input	BOOL	4.2	TRUE/FALSE	FALSE	Controlar salida
OUTP_VAL	Input	INT	6.0	en tanto por mil: de 0 a 1000 como valor analógico S7: de 0 a 27648	0	Especificación del valor de salida Si se determina un valor de salida > 1 000 ó 27648, la CPU lo limita a 1 000 ó 27648, respectivamente
JOB_REQ	Input	BOOL	8.0	TRUE/FALSE	FALSE	Lanzamiento de petición (flanco ascendente)
JOB_ID	Input	WORD	10	W#16#0000 = petición sin función W#16#0001 = escribir período W#16#0002 = escribir retardo a la conexión W#16#0004 = escribir duración mínima de impulso W#16#0081 = leer período W#16#0082 = leer retardo a la conexión W#16#0084 = leer duración mínima de impulso	W#16#0000	Número de petición
JOB_VAL	Input	DINT	12	-2^{31} a $+2^{31}$ -1	0	Valor para peticiones de escritura.
STS_EN	Output	BOOL	16.0	TRUE/FALSE	FALSE	Estado de la habilitación
STS_STRT	Output	BOOL	16.1	TRUE/FALSE	FALSE	Estado de puerta hardware (entrada de inicio)
STS_DO	Output	BOOL	16.2	TRUE/FALSE	FALSE	Estado de salida
JOB_DONE	Output	BOOL	16.3	TRUE/FALSE	TRUE	La nueva petición puede iniciarse
JOB_ERR	Output	BOOL	16.4	TRUE/FALSE	FALSE	Petición errónea
JOB_STAT	Output	WORD	18	W#16#0000 a W#16#FFFF	W#16#0000	Número de error de petición

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros no interconectados en el bloque (datos locales estáticos):

Parámetro	Declaración	Tipo de datos	Dirección (DB de instancia)	Rango de valores	Ajuste predeterminado	Descripción
JOB_OVAL	Output	DINT	20	-2^{31} a 2^{31} -1	0	Valor de salida para peticiones de lectura

Parámetros JOB_ERR y JOB_STAT

Si se ha producido un error de petición, se pone JOB_ERR = TRUE. La causa exacta del error se indica en JOB_STAT.

Clase de evento código de error	Explicación
W#16#0411	Período demasiado corto.
W#16#0412	Período demasiado largo.
W#16#0421	Retardo a la conexión demasiado corto.
W#16#0422	Retardo a la conexión demasiado largo.
W#16#0431	Duración mínima de impulso demasiado corta.
W#16#0432	Duración mínima de impulso demasiado larga.
W#16#04FF	Número de petición no válido.

Errores de sistema

Si se ha producido un error de sistema, BR se pone = FALSE.

Clase de evento código de error	Explicación
W#16#8001	Modo de operación erróneo o error de parametrización. Ajuste el modo de operación correcto en la configuración hardware o utilice la instrucción adecuada al modo de operación ajustado.
W#16#8009	Número de canal no válido. Ajuste un número de canal ≤ 3 (valor específico de la CPU).

9.8.2.5 Comunicación

Información común de parámetros y errores

Parámetros comunes a las instrucciones para la comunicación básica S7

Parámetro REQ

El parámetro de entrada REQ (request to activate) es un parámetro de control disparado por nivel. Permite lanzar la petición (la transmisión de datos o la interrupción de la conexión):

- Si se llama la instrucción para una petición que no está activada en ese momento, la petición se lanza con REQ=1. Si en el momento de la primera llamada de una instrucción todavía no existe una conexión con el interlocutor, ésta se establece antes de iniciar la transmisión.
- Si ha lanzado una petición y ésta todavía no ha finalizado y vuelve a llamar la instrucción para la misma petición, la instrucción no evalúa REQ.

Parámetro REQ_ID (sólo "X_SEND" y "X_RCV")

El parámetro de entrada REQ_ID permite identificar los datos de transmisión. Se transmite desde el sistema operativo de la CPU emisora a la instrucción "X_RCV (Página 8270)" de la CPU del interlocutor.

En el lado receptor es necesario el parámetro REQ_ID en los siguientes casos:

- Para llamar varias instrucciones "X_SEND (Página 8269)" con diferentes parámetros REQ_ID en una CPU emisora y transmitir los datos a un interlocutor.
- Para transmitir datos de varias CPUs emisoras a un interlocutor con la instrucción "X_SEND (Página 8269)".

Evaluando REQ_ID es posible depositar los datos de recepción en diferentes áreas de memoria.

Parámetros RET_VAL y BUSY

Las instrucciones para la comunicación básica S7 son asíncronas, de modo que la ejecución de una petición se prolonga a lo largo de varias llamadas. El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Véase también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Parámetro CONT

El parámetro de entrada CONT (continue) es un parámetro de control. Con él se especifica si la conexión con el interlocutor debe mantenerse una vez finalizada la petición.

- Si en la primera llamada selecciona CONT=0, la conexión se deshace tras finalizar la transmisión. De ese modo estará disponible para el intercambio de datos con otro interlocutor.
  Con este procedimiento se garantiza que sólo estén ocupados los recursos de conexión necesarios en cada momento.
- Si en la primera llamada selecciona CONT=1, la conexión se mantiene tras finalizar la transmisión.
  Este procedimiento resulta apropiado, por ejemplo, para intercambiar datos cíclicamente entre dos estaciones.

Nota

Una conexión establecida con CONT=1 también puede interrumpirse explícitamente con "X_ABORT (Página 8278)" o "I_ABORT (Página 8262)".

Información de error de las instrucciones de comunicación para conexiones S7 no configuradas

Mostrar

Las informaciones de error "reales" indicadas, entre otras, en la siguiente tabla para las instrucciones "X_SEND (Página 8269)", "X_RCV (Página 8270)", "X_GET (Página 8274)", "X_PUT (Página 8276)", "X_ABORT (Página 8278)", "I_PUT (Página 8260)", "I_GET (Página 8258)" y "I_ABORT (Página 8262)" pueden clasificarse de la siguiente manera:

Código de error (W#16# ...)	Explicación
809x	Error en la CPU en la que se ejecuta la instrucción
80Ax	Error de comunicación permanente
80Bx	Error en el interlocutor
80Cx	Error temporal

La tabla siguiente contiene información de errores específicos de las instrucciones.

Código de error (W#16# ...)	Explicación (general)	Explicación (específica de instrucción)
0000	La ejecución ha finalizado sin errores.	"X_ABORT (Página 8269)" y "I_ABORT (Página 8262)": REQ=1, y la conexión indicada no está establecida.
		"X_RCV (Página 8270)": EN_DT=1 y RD=NIL

Código de error (W#16# ...)	Explicación (general)	Explicación (específica de instrucción)
00xy	-	"X_RCV (Página 8270)" con NDA=1 y RD<>NIL: RET_VAL contiene la longitud del bloque de datos recibido (con EN_DT=0) o copiado en RD (con EN_DT=1).
		"X_GET (Página 8274)": RET_VAL contiene la longitud del bloque de datos recibido.
		"I_GET (Página 8258)": RET_VAL contiene la longitud del bloque de datos recibido.
7000	-	"X_SEND (Página 8269)", "X_GET (Página 8274)", "X_PUT (Página 8276)", "X_ABORT (Página 8278)", "I_GET (Página 8258)", "I_PUT (Página 8260)" y "I_ABORT (Página 8262)": Llamada con REQ=0 (llamada sin procesamiento), BUSY tiene el valor "0", no hay ninguna transmisión de datos activa.
		"X_RCV (Página 8270)": EN_DT=0/1 y NDA=0
7001	Primera llamada con REQ=1: La transmisión de datos se ha iniciado; BUSY tiene el valor "1".	-
7002	Llamada intermedia (REQ irrelevante): La transmisión de datos ya está activa; BUSY tiene el valor "1".	"X_ABORT (Página 8278)" y "I_ABORT (Página 8262)": Llamada intermedia con REQ=1
8090	La dirección de destino del interlocutor indicada no es válida, p. ej. • IOID incorrecta • Dirección básica incorrecta • Dirección MPI incorrecta (> 126)	-
8092	Error en SD o RD, p. ej.: El direccionamiento del área de datos local no está permitido.	"X_SEND (Página 8269)", p. ej.. • Longitud no permitida de SD • SD=NIL no está permitido
		"X_RCV (Página 8270)", p. ej. • Se han recibido más datos de los que caben en el área especificada por RD. • RD es del tipo de datos BOOL, pero la longitud de los datos recibidos es mayor que un byte.
		"X_GET (Página 8274)" e "I_GET (Página 8258)", p. ej. • Longitud no permitida en RD • La longitud o el tipo de datos de RD no coincide con los datos recibidos. • RD=NIL no está permitido.

Código de error (W#16# ...)	Explicación (general)	Explicación (específica de instrucción)
		"X_PUT (Página 8276)" e "I_PUT (Página 8260)", p. ej. • Longitud no permitida en SD • SD=NIL no está permitido
8095	El bloque ya se está ejecutando en una clase de prioridad inferior.	
80A0	Error en el acuse recibido	"X_PUT (Página 8276)" e "I_PUT (Página 8260)": El interlocutor no soporta el tipo de datos indicado en SD de la CPU emisora.
80A1	Problemas de comunicación: Llamada de la instrucción tras la interrupción de una conexión existente	"AS_DIAL (Página 8284)": • El TS Adapter local ha rechazado la ejecución de una instrucción de comunicación porque no hay ninguna conexión remota configurada con "AS_DIAL". • Durante la ejecución de la instrucción de comunicación se ha interrumpido la conexión remota con el TS Adapter remoto.
80B0	El objeto no está accesible, p. ej. DB no cargado	Posible con "X_GET (Página 8274)" y "X_PUT (Página 8276)", y con "I_GET (Página 8258)" e "I_PUT (Página 8260)"
80B1	Error en el puntero ANY. La longitud del área de datos que debe transmitirse es incorrecta.	-
80B2	Error de HW: El módulo no existe. • El slot configurado no está ocupado. • El tipo de módulo real no coincide con el tipo de módulo configurado • La periferia descentralizada no está disponible. • En el SDB correspondiente no hay ninguna entrada para el módulo.	Posible con "X_GET (Página 8274)" y "X_PUT (Página 8276)", y con "I_GET (Página 8258)" e "I_PUT (Página 8260)"
80B3	Los datos sólo pueden leerse o sólo pueden escribirse, p. ej. DB protegido contra escritura	Posible con "X_GET (Página 8274)" y "X_PUT (Página 8276)", y con "I_GET (Página 8258)" e "I_PUT (Página 8260)"
80B4	Error de tipo de datos en el puntero ANY, o ARRAY del tipo de datos indicado no permitido.	"X_GET (Página 8274)" y "X_PUT (Página 8276)", así como "I_GET (Página 8258)" e "I_PUT (Página 8260)": El interlocutor no soporta el tipo de datos indicado en VAR_ADDR.

Código de error (W#16# ...)	Explicación (general)	Explicación (específica de instrucción)
80B5	Ejecución rechazada debido a un estado operativo no permitido	Posible con "X_SEND (Página 8269)". Con "AS_DIAL (Página 8284)": • El TS Adapter local ha rechazado la ejecución de una instrucción de comunicación porque la función DIAL de "AS_DIAL" todavía no ha finalizado. • La función DIAL del "AS_DIAL" se ha solicitado a pesar de que ya hay una conexión remota configurada para el acoplamiento remoto inter-AS.
80B6	En el acuse recibido hay un código de error desconocido.	-
80B7	Tipo de datos y/o longitud de los datos transmitidos no son compatibles con el área de la CPU interlocutora en la que deben escribirse.	Posible con "X_PUT (Página 8276)" e "I_PUT (Página 8260)"
80B8	-	"X_SEND (Página 8269)": "X_RCV (Página 8270)" del interlocutor ha rechazado los datos (RD=NIL).
80B9	-	"X_SEND (Página 8269)": El interlocutor ha identificado el bloque de datos (llamada de "X_RCV (Página 8270)" con EN_DT=0), pero debido al estado operativo STOP todavía no se ha aplicado en el programa de usuario.
80BA	La respuesta del interlocutor no cabe en el telegrama de comunicación.	-
80C0	La conexión indicada ya está ocupada por otra petición.	-
80C1	Escasez de recursos en la CPU en la que se ejecuta la instrucción. p. ej.: • Ya se está procesando el número máximo de peticiones de transmisión diferentes del módulo. • El recurso de conexión está ocupado p. ej. por la recepción.	-

Código de error (W#16# ...)	Explicación (general)	Explicación (específica de instrucción)
80C2	Falta temporal de recursos del interlocutor, p. ej.: • El interlocutor está procesando el número máximo de peticiones. • Los recursos necesarios (memoria, etc.) están ocupados. • La memoria de trabajo disponible no es suficiente. (Lance un proceso de compresión.)	Con "AS_DIAL (Página 8284)": • Escasez de recursos temporal en la CPU remota. • La CPU remota con la dirección MPI no está disponible o no existe.
80C3	Error al establecer la conexión, p. ej.: • La estación S7 propia no se encuentra en la subred MPI. • Ha direccionado la estación propia en la subred MPI. • El interlocutor ya no está accesible. • Falta de recursos temporal del interlocutor	-

Comunicación S7

Sinopsis de las instrucciones para la comunicación S7

Clasificación

Para la comunicación S7 es necesario configurar las conexiones. La llamada de las funciones de comunicación integradas se efectúa en el programa de usuario mediante instrucciones.

Estas instrucciones pueden clasificarse en las siguientes categorías:

- Instrucciones para el intercambio de datos
- Instrucciones para cambiar de estado operativo
- Instrucciones para la consulta del estado operativo
- Una instrucción para cada consulta de conexión

Nota

Si la CPU S7-300 utilizada no dispone de interfaz Ethernet, para ejecutar las instrucciones S7-300 se requiere un CP SIMATIC NET (Página 8047) de la serie S7-300. Encontrará más información al respecto en la documentación correspondiente.

Instrucciones para el intercambio de datos

Las instrucciones de comunicación para el intercambio de datos permiten el intercambio de datos entre dos interlocutores. Dependiendo de si hay una instrucción de comunicación sólo

en el módulo local o tanto en éste como en el remoto, se habla de un intercambio de datos unilateral o bilateral.

Instrucción	Descripción
"USEND (Página 7683)" / "URCV (Página 7685)"	Transmisión rápida y no acusada de datos independientemente de la ejecución temporal de la función de comunicación ("URCV") en el interlocutor (p. ej. avisos de operación y de mantenimiento). Ello significa que en el interlocutor los datos pueden ser sobrescritos por datos más actuales.
"BSEND (Página 7688)" / "BRCV (Página 7691)"	Transmisión segura de un bloque de datos al interlocutor. En este caso, la transmisión no finaliza hasta que la función de recepción ("BRCV") en el interlocutor no haya recibido los datos.
"GET (Página 7677)"	Lectura controlada por programa de variables sin función de comunicación adicional en el programa de usuario del interlocutor.
"PUT (Página 7680)"	Escritura controlada por programa de variables sin función de comunicación adicional en el programa de usuario del interlocutor.
"PRINT (Página 7708)"	Enviar datos a una impresora (sólo S7-400: CP441).

Instrucciones para cambiar de estado operativo

Con instrucciones para cambiar de estado operativo se controla el estado operativo de un dispositivo remoto.

En las instrucciones para el cambio de estado operativo, el intercambio de datos es unilateral.

Instrucción	Descripción
"START (Página 7714)"	Lanzar el REARRANQUE completo de una CPU S7-300/400 cuando ésta se encuentra en el estado operativo STOP.
"STOP (Página 7716)"	Lanzar el STOP de una CPU S7-300/400 cuando ésta se encuentra en el estado operativo RUN, PARADA o en arranque.
"RESUME (Página 7718)"	Lanzar el rearranque de una CPU S7-400 cuando ésta se encuentra en el estado operativo STOP.

Instrucciones para la consulta del estado operativo

Las instrucciones de consulta del estado operativo permiten recibir información sobre el estado operativo de un dispositivo remoto.

Con la instrucción "STATUS", el intercambio de datos es unilateral, mientras que con la instrucción "USTATUS (Página 7722)" es bilateral.

Instrucción	Descripción
"STATUS (Página 7720)"	Devuelve el estado operativo de un interlocutor (CPU S7-400) cuando el usuario lo consulta.
"USTATUS (Página 7722)"	Recibe el estado operativo de una CPU S7-400 cuando cambia su estado operativo, siempre que el atributo de conexión correspondiente (Enviar avisos sobre el estado operativo) esté activado.

Instrucción para la consulta de una conexión

Instrucción S7-400	Instrucción S7-300	Descripción
"CONTROL (Página 7705)"	-	Consulta del estado de una conexión perteneciente a una instancia de una instrucción.
-	"C_CNTRL (Página 7694)"	Consulta del estado de una conexión a través de la ID de conexión.

Nota

El estado de la conexión actual también puede diagnosticarse con la instrucción "C_DIAG (Página 7027)" (sólo con S7-400).

Coherencia de datos

Definición

El tamaño del área de datos que no puede ser modificado simultáneamente por procesos concurrentes se denomina área de datos coherente. Así, un área de datos que supera el tamaño del área de datos coherente puede falsearse en su totalidad.

Esto significa que un área de datos conexa cuyo volumen supere el del área de datos coherente puede estar compuesta en un momento dado en parte por bloques de datos nuevos y en parte por bloques coherentes antiguos.

Ejemplo

Las incoherencias pueden surgir en el momento en que un bloque de comunicación es interrumpido p. ej. por un OB de alarma de proceso de mayor prioridad. Si el programa de usuario modifica entonces en este OB los datos, que en parte ya han sido procesados por el bloque de comunicación, los datos transmitidos procederán:

- en parte, de antes de la ejecución de la alarma de proceso
- y, en parte, de después de ejecutar la alarma de proceso.

Esto significa que los datos son incoherentes (inconexos).

Efecto

Cuando deben transmitirse grandes cantidades de datos con coherencia, la transmisión no puede ser interrumpida. Con ello puede, por ejemplo, prolongarse el tiempo de reacción a alarmas de la CPU.

Coherencia de datos en SIMATIC

- Si en el programa de usuario existe una función de comunicación como p. ej. "BSEND (Página 7688)" / "BRCV (Página 7691)", que accede a datos conjuntos, el acceso a esa área de datos puede coordinarse p. ej. con el parámetro DONE. La coherencia de datos de las áreas de comunicación que se transmite localmente con un bloque de comunicación puede garantizarse por tanto en el programa de usuario.
- En las funciones de comunicación S7 como "PUT (Página 7680)" / "GET (Página 7677)" o escribir/leer vía comunicación OP, el tamaño de las áreas de datos coherentes ya debe tenerse en cuenta en la programación o configuración, puesto que en el programa de usuario del dispositivo de destino (servidor) no hay un bloque de comunicación que pueda sincronizar los datos de comunicación al entrar en el programa de usuario.
- En el S7-300 los datos de comunicación se copian de forma coherente en la memoria de usuario, en forma de bloques de 32 bytes y en el punto de control del ciclo del sistema operativo. Para todas las áreas de datos de mayor tamaño no se garantiza la coherencia de datos. Si se requiere una coherencia de datos definida, los datos de comunicación en el programa de usuario no pueden superar los 32 bytes (según la versión, máximo 8 bytes).
- Por el contrario, en la CPU S7-400 los datos de comunicación no se procesan en el punto de control del ciclo, sino en intervalos fijos del ciclo del programa. El propio sistema garantiza la coherencia de una variable.
- Así, un OP o una OS, por ejemplo, puede acceder con coherencia a estas áreas de comunicación con las instrucciones "PUT (Página 7680)" / "GET (Página 7677)" o bien leer/escribir variables.

Nota

Encontrará más indicaciones sobre la coherencia de datos en la descripción de las distintas instrucciones.

Parámetros comunes de las instrucciones para la comunicación S7

Clasificación

Los parámetros de las instrucciones para la comunicación S7 pueden clasificarse en las siguientes cinco categorías funcionales:

1. Los parámetros de control permiten activar una instrucción.
2. Los parámetros de direccionamiento sirven para direccionar el interlocutor remoto.
3. Los parámetros de transmisión indican las diferentes áreas de datos que deben transmitirse al interlocutor remoto.

4. Los parámetros de recepción indican las áreas de datos en las que se registran los datos recibidos del interlocutor remoto.
5. Los parámetros de estado permiten controlar si la instrucción ha terminado su cometido correctamente, o bien analizar los errores ocurridos.

Parámetros de control

El intercambio de datos no se activa hasta que en la llamada de la instrucción los parámetros de control correspondientes no tengan un valor definido (p. ej. que estén activados) o hasta que el valor haya cambiado en un modo determinado respecto a la última llamada (p. ej. flanco ascendente).

Nota

Primera llamada en el S7-300

En la primera llamada, al parámetro REQ debe asignársele FALSE .

Parámetros de direccionamiento

Parámetro	Descripción
ID	Referencia a la descripción de la conexión local (predeterminada por la configuración de conexiones). Nota: En las instrucciones de la comunicación S7 no se permite la ID W#16#EEEE.
R_ID	Con el parámetro R_ID se define la correspondencia entre una instrucción de transmisión y una de recepción: El parámetro R_ID debe coincidir en las instrucciones del lado de transmisión y del lado de recepción. Ello permite la comunicación de varios pares de instrucciones a través de la misma conexión lógica. • R_ID debe indicarse en la forma DW#16#wxyzWXYZ. • Los pares de instrucciones de una conexión lógica definidos con R_ID deben ser unívocos para dicha conexión.

El parámetro PI_NAME sólo se describe en las instrucciones relevantes (sólo S7-400).

Nota

Parámetros de direccionamiento ID y R_ID

S7-300: Los parámetros de direccionamiento ID y R_ID pueden reparametrizarse en runtime. Los nuevos parámetros se hacen efectivos con cada nueva petición al finalizar la petición anterior.

Las siguientes posibilidades permiten reducir el número de DBs de instancia y, con ello, ahorrar memoria de trabajo:

1. Con IDs variables pueden utilizarse varias conexiones a través de un bloque de datos de instancia.
2. Con R_IDs variables es posible definir con una instancia varias correspondencias entre pares de instrucciones de transmisión y recepción para una petición.
3. Los casos 1 y 2 pueden combinarse entre sí.

Recuerde que los nuevos parámetros no se hacen efectivos hasta no haberse completado la petición anterior. Al activar el proceso de transmisión, el parámetro R_ID debe coincidir en la instrucción del lado de transmisión y en la del lado de recepción.

S7-400: Los parámetros de direccionamiento ID y R_ID sólo se evalúan en la primera llamada de la instrucción o cuando se carga la instancia (los parámetros actuales o los valores predefinidos de la instancia). Así, con la primera llamada se fija la relación de comunicación (conexión) con el interlocutor remoto hasta el siguiente rearranque completo (en caliente) o el siguiente arranque en frío.

Parámetros de estado

Con los parámetros de control se controla si la instrucción ha terminado su cometido correctamente o si todavía está activa. Además, indican los errores ocurridos.

Nota

Los parámetros de estado sólo son válidos durante un ciclo, es decir, desde el primer comando que sigue a la llamada, hasta la siguiente llamada. Por ello es necesario evaluar estos parámetros cada vez que se ejecuta la instrucción.

Parámetros de transmisión y recepción

Si en una instrucción no se utilizan todos los parámetros de transmisión o recepción, cada primer parámetro no utilizado debe ser un puntero NIL, y los parámetros utilizados deben ir seguidos y sin espacios.

Nota

Primera llamada en el S7-400

En la primera llamada los punteros ANY fijan la cantidad de datos de usuario máxima transmisible de la petición; en la memoria de trabajo de la CPU se crea un búfer de datos de comunicación para asegurar la coherencia de los datos. Este búfer ocupa hasta 480 bytes de la memoria de trabajo. Se recomienda realizar la primera llamada en el OB de rearranque completo (en caliente) o en el OB de arranque en frío si el bloque que contiene la llamada de la instrucción no se carga a posteriori en el estado operativo RUN de la CPU.

En llamadas consecutivas se puede transmitir/recibir un número indefinido de datos, pero como máximo tantos como en la primera llamada.

La excepción a esta regla son las instrucciones "BSEND (Página 7688)" y "BRCV (Página 7691)". Éstas permiten transmitir hasta 64 Kbytes por petición.

En las instrucciones para la comunicación bilateral

- el número de parámetros SD_i y RD_i utilizados en los lados de transmisión y recepción debe coincidir
- en los parámetros SD_i y RD_i conexos deben coincidir los tipos de datos en los lados de transmisión y recepción
- el número de datos que deben transferirse a través del parámetro SD_i no puede ser superior al área abarcada por el parámetro correspondiente RD_i (no es válido para "BSEND (Página 7688)" / "BRCV (Página 7691)").

La infracción de estas reglas se indica con ERROR = 1 y STATUS = 4.

Volumen de datos de usuario

En las instrucciones "USEND (Página 7683)", "URCV (Página 7685)", "GET (Página 7677)" y "PUT (Página 7680)", el número de datos que deben transferirse no puede superar una longitud de datos de usuario determinada. El volumen máximo de datos de usuario depende:

- de la instrucción utilizada y
- del interlocutor

El volumen mínimo garantizado de datos de usuario para una instrucción con 1-4 variables se puede consultar en la siguiente tabla:

Instrucción	Interlocutor: S7-300	Interlocutor: S7-400
PUT / GET	160 bytes	400 bytes
USEND / URCV	160 bytes	440 bytes
BSEND / BRCV	32768/65534 bytes	65534 bytes

Observe las limitaciones relativas al volumen de datos de usuario en los datos técnicos de la respectiva CPU.

Volumen exacto de datos de usuario

Si el volumen de datos de usuario indicado anteriormente no es suficiente, es posible determinar la longitud de datos de usuario máxima exacta en bytes del siguiente modo:

- En primer lugar consulte en la tabla siguiente el tamaño válido para bloques de datos en la comunicación:

CPU propia	CPU remota	Tamaño del bloque de datos en bytes
S7-300	cualquiera	240 (S7-300)
S7-400	S7-300	240 (S7-400)
S7-400	S7-400	480

- Partiendo de este valor, consulte en la tabla siguiente la longitud de datos de usuario máxima posible en bytes, como suma de los parámetros de usuario. Es aplicable a longitudes pares de las áreas SD_i, RD_i, ADDR_i. Para cada área de longitud impar la longitud de datos de usuario máxima posible se reduce en un byte.

		Número de parámetros utilizados SD_i, RD_i, ADDR_i			
Tamaño del bloque de datos	Instrucción	1	2	3	4
240 (S7-300)	PUT/GET/ USEND	160	-	-	-
240 (S7-300 vía interfaz integrada)	PUT	212	-	-	-
	GET	222	-	-	-
	USEND	212	-	-	-
240 (S7-400)	PUT	212	196	180	164
	GET	222	218	214	210
	USEND	212	-	-	-
480	PUT	452	436	420	404
	GET	462	458	454	450
	USEND	452	448	444	440

Comportamiento de arranque de las instrucciones para la comunicación S7

Requisitos

En adelante, para la S7-400 se parte de los siguientes supuestos:

- Las descripciones de las conexiones (SDBs) están disponibles en los módulos.
- Las conexiones configuradas están establecidas.
- En cada instrucción el parámetro actual de ID coincide con la ID de conexión configurada.

Rearranque completo (en caliente) y arranque en frío

En el rearranque completo (en caliente) y en el arranque en frío todas las instrucciones se ponen en estado NO_INIT. Los parámetros actuales guardados en los DBs de instancia permanecen invariables.

Rearranque completo (en caliente) y arranque en frío con instrucciones para la comunicación bilateral

En general, en las instrucciones de comunicación bilateral, los dos módulos no ejecutan un rearranque completo (en caliente) o un arranque en frío simultáneamente. La adaptación es realizada implícitamente por la instrucción siguiendo las reglas siguientes.

- Los bloques de recepción ("URCV (Página 7685)", "BRCV (Página 7691)") se comportan del siguiente modo:
  - Si la instrucción ha recibido una petición, pero en el momento del rearranque completo (en caliente) o del arranque en frío aún no la ha acusado, genera un telegrama de interrupción de secuencia ("BRCV (Página 7691)") y a continuación pasa inmediatamente al estado NO_INIT.
  - Con "BRCV (Página 7691)" puede ocurrir que entre un segmento de datos adicional a pesar de haberse enviado la interrupción de la secuencia. En ese caso, el segmento se rechaza localmente.
  - En el caso de "URCV (Página 7685)" se produce una transición inmediata al estado NO_INIT.
- Los bloques de transmisión ("USEND (Página 7683)", "BSEND (Página 7688)") se comportan del siguiente modo:
  - Si la instrucción "BSEND (Página 7688)" ha iniciado una secuencia de la petición que todavía no ha finalizado, transmite una interrupción de secuencia en el rearranque completo (en caliente) o en el arranque en frío. Inmediatamente después deriva al estado NO_INIT. Cualquier acuse que entre con posterioridad se rechaza localmente.
  - Si la instrucción "BSEND (Página 7688)" ya ha enviado o recibido una interrupción de secuencia al solicitarse el rearranque completo (en caliente) o arranque en frío, cambia inmediatamente al estado NO_INIT.
  - En cualquier otro caso, y cuando la instrucción sólo envía mensajes (p. ej. "USEND (Página 7683)"), la ejecución local se interrumpe y la instrucción pasa inmediatamente al estado NO_INIT.

Rearranque completo (en caliente) y arranque en frío con instrucciones para la comunicación unilateral

Se parte del supuesto de que una vez establecidas las conexiones, el servidor del interlocutor está operativo, es decir, que puede procesar peticiones o enviar mensajes en cualquier momento.

Las instrucciones que transmiten peticiones y esperan la llegada de acuses tienen la siguiente respuesta:

- La ejecución en curso se interrumpe y a continuación se pasa inmediatamente al estado NO_INIT. Cualquier acuse que llegue después para la petición transmitida antes del rearranque completo (en caliente) o arranque en frío se rechaza localmente.
- Es posible lanzar una nueva petición antes de que llegue el acuse pendiente.

Las instrucciones que envían o reciben mensajes se comportan de la siguiente manera:

- La ejecución en curso se interrumpe y se deriva inmediatamente al estado NO_INIT.
- Con "USTATUS (Página 7722)", los mensajes que llegan en los estados NO_INIT y DISABLED se rechazan localmente.

Comportamiento en rearranque

Las instrucciones para la comunicación S7 sólo pasan al estado NO_INIT en el rearranque completo (en caliente) o en el arranque en frío. De ahí resulta que en el rearranque se comporten como bloques de función de usuario continuables.

Comportamiento tras un borrado total

El borrado total siempre causa la interrupción de todas las conexiones. Dado que tras el borrado total el único modo de arranque posible para el programa de usuario es el rearranque completo (en caliente) o el arranque el frío, todas las instrucciones de la comunicación S7 (si todavía las hay) se ponen en estado NO_INIT y se inicializan. Como consecuencia de interrumpirse la conexión, los bloques interlocutores de un módulo que no se ha sometido al borrado total pasan a los estados IDLE, ENABLED o DISABLED.

Comportamiento en caso de fallo de las instrucciones para la comunicación S7

Introducción

A continuación se describe el comportamienot de las instrucciones para la comunicación S7 en caso de fallo para la S7-400.

Interrupción de la conexión

Las conexiones asignadas a las instancias de la instrucción se someten a un control de interrupción.

En caso de interrupción, la reacción de la instrucción dependerá de su estado interno:

- Si se detecta la interrupción de la conexión en los estados IDLE o ENABLED, la instrucción reacciona del siguiente modo:
  - Cambia al estado ERROR y devuelve la ID de error "Problemas de comunicación" a través de los parámetros de salida ERROR y STATUS.
  - En la siguiente llamada recobra su estado original y vuelve a comprobar la conexión.
- Una instrucción que no se encuentra en los estados IDLE o DISABLED reacciona del siguiente modo:
  - Cancela su ejecución, cambia inmediatamente o en la siguiente llamada al estado ERROR y devuelve la ID de error "Problemas de comunicación" a través de los parámetros de salida ERROR y STATUS.
  - En la siguiente llamada, la instrucción adopta el estado IDLE, DISABLED o ENABLED. En los estados IDLE y ENABLED se vuelve a comprobar la conexión.

Este procedimiento también se lleva a cabo aunque entretanto se haya restablecido la conexión.

Fallo de red

Un fallo de red respaldado por pila con rearranque conlleva la interrupción de todas las conexiones establecidas. Por ello, para todas las instrucciones afectadas rige lo arriba descrito.

En el caso de un fallo de red respaldado por pila con rearranque completo (en caliente) o arranque en frío automáticos son aplicables las reacciones descritas tanto para el caso de interrupción de la conexión, como para el rearranque completo (en caliente) o el arranque en frío.

En el caso especial del rearranque completo (en caliente) o arranque en frío automático no respaldado por pila, en el que con la recuperación de la red se produce un borrado total automático, las instrucciones para la comunicación S7 se comportan de la forma descrita en el capítulo "Comportamiento de arranque de las instrucciones para la comunicación S7 (Página 7673)".

Respuesta a las transiciones de estado operativo

En una transición entre los estados STOP, ARRANQUE, RUN Y PARADA la instrucción permanece en su estado actual (excepción: en un rearranque completo (en caliente) o arranque en frío se adopta el estado NO_INIT). Esto es aplicable tanto en instrucciones para la comunicación unilateral como bilateral.

Interfaz de error con el programa de usuario

Si se produce un error durante la ejecución de una instrucción, siempre se pasa al estado ERROR; al mismo tiempo el parámetro de salida ERROR se pone a "1" y en el parámetro de salida STATUS se registra la ID de error correspondiente. Estas indicaciones pueden evaluarse en el programa de usuario.

Ejemplos de posibles errores:

- Error al recoger los datos de transmisión
- Error al copiar los datos de recepción en las áreas de recepción (p. ej. acceso a un DB no existente)
- La longitud del área de datos transferida no coincide con la longitud del área de recepción registrada en la instrucción del interlocutor.

GET: Leer datos de una CPU remota

Descripción

La instrucción permite leer datos de una CPU remota si no se realiza la conexión a través de un CP.

- **S7-300:** El proceso de lectura se inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros ID, ADDR_1 y RD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros ID, ADDR_1 y RD_1.
- **S7-400:** Con un flanco ascendente en la entrada de control REQ se inicia la instrucción. Los punteros hacia las áreas que deben leerse (ADDR_i) se transfieren a la CPU interlocutora.

El interlocutor remoto devuelve los contenidos de los datos. Los datos recibidos se copian en la siguiente llamada en las áreas de recepción configuradas (RD_i). Hay que asegurarse de que las áreas definidas por los parámetros ADDR_i y RD_i se correspondan entre sí en cuanto a la cantidad, la longitud y el tipo de datos.

El término de este proceso se indica con "1" en el parámetro de estado NDR. No es posible activar otro proceso de lectura hasta que no haya finalizado el último.

La CPU remota puede encontrarse en el estado operativo RUN o STOP. En caso de problemas de acceso al leer los datos, o si la comprobación del tipo de datos devolvió un error, los errores y advertencias correspondientes se indican a través de ERROR y STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR / STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
S7-300: ADDR_1 **S7-400:** ADDR_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero hacia las áreas de la CPU interlocutora que deben leerse. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10). En la transferencia de estructuras de datos (p. ej. Struct, Array) en el parámetro ADDR debe utilizarse el tipo de datos CHAR, BYTE, WORD o DWORD.
S7-300: RD_1 **S7-400:** RD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero hacia las áreas de la CPU propia en las que se depositan los datos leídos. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "GET" que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.

ERROR	STATUS (decimal)	Explicación
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Adicionalmente para el S7-300: – Se ha superado el número máximo de peticiones/instancias paralelas.
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	4	Error en los punteros de áreas de recepción RD_i en cuanto a la longitud o el tipo de datos.
1	8	Error de acceso en la CPU interlocutora.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "GET" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	20	• S7-400 La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria. • S7-300 – Se ha superado el número máximo de peticiones/instancias paralelas – Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) – Es posible en la primera llamada
1	27	Solo con S7-300: Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

Los datos se reciben con coherencia si se observa lo siguiente:

Evalúe la parte actualmente utilizada del área de recepción RD_i por completo antes de activar una nueva petición.

PUT: Escribir datos en una CPU remota

Descripción

La instrucción permite escribir datos en una CPU remota si no se realiza la conexión a través de un CP.

- **S7-300:** El proceso de transmisión se inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros ID, ADDR_1 y SD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros ID, ADDR_1 y SD_1.
- **S7-400:** Con un flanco ascendente en la entrada de control REQ se inicia la instrucción. Los punteros hacia las áreas que deben escribirse (ADDR_i) y los datos (SD_i) se transfieren a la CPU interlocutora.

El interlocutor remoto deposita los datos transmitidos en las direcciones indicadas y devuelve un acuse de ejecución. Hay que asegurarse de que las áreas definidas por los parámetros ADDR_i y SD_i se correspondan entre sí en cuanto a la cantidad, la longitud y el tipo de datos.

Si no se ha producido ningún error, esto se indica en la siguiente llamada con "1" en el parámetro de estado DONE . No es posible activar otro proceso de escritura hasta que no haya finalizado el último.

La CPU remota puede encontrarse en el estado operativo RUN o STOP. En caso de problemas de acceso al escribir los datos, o si la comprobación del acuse de ejecución devolvió un error, los errores y advertencias correspondientes se indican a través de ERROR y STATUS .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PUT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
S7-300: ADDR_1 **S7-400:** ADDR_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero hacia las áreas de la CPU interlocutora en las que debe escribirse. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10). En la transferencia de estructuras de datos (p. ej. Struct, Array) en el parámetro ADDR debe utilizarse el tipo de datos CHAR, BYTE, WORD o DWORD.
S7-300: SD_1 **S7-400:** SD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero hacia las áreas de la CPU propia que contienen los datos que deben transmitirse. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de errores específicos de PUT, que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.

ERROR	STATUS (decimal)	Explicación
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Adicionalmente para el S7-300: – Se ha superado el número máximo de peticiones/instancias paralelas.
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	4	Error en los punteros de área de transmisión SD_i en cuanto a la longitud o el tipo de datos.
1	8	Error de acceso en la CPU interlocutora
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "PUT" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente)
1	20	• S7-400 La memoria de trabajo disponible no es suficiente. Solución: vuelva a cargar el DB de instancia correspondiente. • S7-300 – Se ha superado el número máximo de peticiones/instancias paralelas – Las instancias se han sobrescrito en el estado RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) – Es posible en la primera llamada
1	27	Solo con S7-300: Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos en el S7-300:

Para garantizar la coherencia de datos, no puede volver a escribirse en el área de transmisión SD_1 hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

Coherencia de datos en el S7-400 y S7-300 a través de una interfaz integrada:

Con la activación de un proceso de transmisión (flanco ascendente en REQ) los datos que deben transmitirse de las áreas de transmisión SD_i se copian del programa de usuario. Se puede volver a escribir en estas áreas tras la llamada de la instrucción, sin que se falsifiquen los datos de transmisión actuales.

Nota

El proceso de transmisión completo finaliza sólo cuando el parámetro de estado DONE adopta el valor "1".

USEND: Envío de datos sin coordinación

Descripción

La instrucción "USEND" envía datos a una instrucción remota de interlocutor del tipo "URCV (Página 7685)". El proceso de transmisión se desarrolla sin coordinación con la instrucción del interlocutor, es decir, que la transmisión de datos se produce sin acuse por parte de la instrucción del interlocutor.

- **S7-300:** El proceso de transmisión se inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros R_ID, ID y SD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros R_ID, ID y SD_1.
- **S7-400:** El proceso de transmisión se inicia tras un flanco ascendente en la entrada de control REQ. Los datos que deben transmitirse se referencian con los parámetros SD_1, ... SD_4, aunque no es necesario que los cuatro parámetros estén ocupados.

No obstante, hay que asegurarse de que las áreas definidas por los parámetros SD_1 a SD_4 / SD_1 y RD_1 a RD_4 / RD_1 (en la respectiva instrucción de interlocutor "URCV (Página 7685)") coincidan en:

- Número
- Longitud
- Tipo de datos

El parámetro R_ID debe ser idéntico en ambas instrucciones.

La finalización correcta del proceso de transmisión se indica con un "1" lógico en el parámetro de estado DONE .

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "USEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
R_ID	Input	DWORD	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error:

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	I, Q, M, D, L	• ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
S7-300: SD_1 **S7-400:** SD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero al área de transmisión n.º i. Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Adicionalmente para el S7-300: – Se ha superado el número máximo de peticiones/instancias paralelas
1	4	Error en los punteros de área de transmisión SD_i en cuanto a la longitud o el tipo de datos.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "USEND" • No se ha indicado ningún DB de instancia, sino un DB global • Se ha encontrado un DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	18	• R_ID ya existe en la conexión ID. • Adicionalmente para el S7-300: Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP).

ERROR	STATUS (decimal)	Explicación
1	20	• **S7-400** La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria. • **S7-300** – Se ha superado el número máximo de peticiones/instancias paralelas – Las instancias se han sobrescrito en el estado RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) – Es posible en la primera llamada
1	27	**Solo con S7-300:** Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

S7-300: Para garantizar la coherencia de datos, no puede volver a escribirse en el área de transmisión SD_1 hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

S7-400 y S7-300 a través de una interfaz integrada: Con la activación de un proceso de transmisión (flanco ascendente en REQ) los datos que deben transmitirse de las áreas de transmisión SD_i se copian del programa de usuario. Se puede volver a escribir en estas áreas tras la llamada de la instrucción, sin que se falsifiquen los datos de transmisión actuales.

Nota

El proceso de transmisión completo finaliza sólo cuando el parámetro de estado DONE adopta el valor "1".

URCV: Recibir datos sin coordinación

Descripción

La instrucción "URCV" recibe datos de forma asíncrona de una instrucción remota del interlocutor del tipo "USEND (Página 7683)" y los copia en las áreas de recepción configuradas.

La instrucción está lista para recibir cuando en la entrada EN_R hay un "1" lógico. Con EN_R=0 no es posible interrumpir una petición en curso.

- **S7-300:** Con cada flanco ascendente en EN_R se aplican los parámetros R_ID, ID y RD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros R_ID, ID y RD_1.
- **S7-400:** Las áreas de datos de recepción se referencian con los parámetros RD_1, ... RD_4.

Hay que asegurarse de que las áreas definidas por los parámetros RD_i / RD_1 y SD_i / SD_1 (en la respectiva instrucción del interlocutor correspondiente "USEND (Página 7683)") coincidan en:

- Número
- Longitud
- Tipo de datos

La finalización correcta del proceso de copia se indica con un "1" lógico en el parámetro de estado NDR . El parámetro R_ID debe ser idéntico en ambas instrucciones.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "URCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	El parámetro de control enabled to receive señaliza disponibilidad de recepción cuando la entrada está activada.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
R_ID	Input	DWORD	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	
S7-300: RD_1 **S7-400:** RD_i (1≤ i ≤4)	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero al área de recepción n.º i. Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (decimal)	Explicación
0	9	Advertencia Overrun: Los datos de recepción antiguos han sido sobrescritos por datos de recepción más actuales.
0	11	Advertencia: Los datos de recepción ya se están procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Adicionalmente para el S7-300: Se ha superado el número máximo de peticiones/instancias paralelas
1	4	Error en los punteros de áreas de recepción RD_i en cuanto a la longitud o el tipo de datos.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "URCV" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	18	• R_ID ya existe en la ID de conexión • Adicionalmente para el S7-300: Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP.)
1	19	La instrucción correspondiente "USEND (Página 7683)" envía datos más rápido de lo que "URCV" puede copiarlos en las áreas de recepción.
1	20	• **S7-400** La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria. • **S7-300** – Se ha superado el número máximo de peticiones/instancias paralelas – Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP.) – Es posible en la primera llamada
1	27	**Solo con S7-300:** Para este bloque no existe ningún código de función en la CPU.

Coherencia de datos

Los datos se reciben con coherencia si se observa lo siguiente:

- **S7-300:** Una vez que el parámetro de estado NDR ha adoptado el valor "1", hay que volver a llamar inmediatamente "URCV" con el valor "0" en EN_R. Con ello queda garantizado que el área de recepción no se sobrescriba antes de haberse evaluado. Evalúe el área de recepción RD_1 por completo antes de volver a llamar el bloque con el valor "1" en la entrada de control EN_R.
- **S7-400:** Una vez que el parámetro de estado NDR ha adoptado el valor "1", hay nuevos datos de recepción en las áreas de recepción (RD_i). Una nueva llamada del bloque puede sobrescribir estos datos con nuevos datos de recepción. Para impedirlo, llame "URCV" con el valor "0" en EN_R (p. ej. en caso de ejecución cíclica del bloque) hasta haber finalizado el procesamiento de los datos de recepción.

BSEND: Transmitir datos en segmentos

Descripción

La instrucción "BSEND" envía datos a una instrucción remota del interlocutor del tipo "BRCV (Página 7691)". En este tipo de transferencia de datos es posible transportar una cantidad de datos mayor entre los interlocutores que con las demás instrucciones de comunicación para conexiones S7 configuradas. Pueden transmitirse las siguientes cantidades de datos:

- 32768 bytes con el S7-300 a través de CPs SIMATIC-Net
- 65534 bytes con el S7-400 y S7-300 a través de una interfaz integrada

Funcionamiento

El área de datos que se va a transmitir se segmenta. Cada segmento se transmite individualmente al interlocutor. El interlocutor acusa el último segmento ya en el momento de su llegada, independientemente de la respectiva llamada de la instrucción "BRCV (Página 7691)".

- **S7-300:** El proceso de transmisión se inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros R_ID, ID, SD_1 y LEN. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros R_ID, ID, SD_1 y LEN. Para la transmisión de datos segmentados es necesario llamar la instrucción cíclicamente en el programa de usuario.
  La dirección de inicio y la longitud máxima de los datos que van a transmitirse vienen determinadas por SD_1. La longitud del bloque de datos se fija individualmente para cada petición con LEN.
- **S7-400 y S7-300 a través de una interfaz integrada:** La activación del proceso de transmisión se produce tras llamar la instrucción y un flanco ascendente en la entrada de control REQ. La transmisión de los datos desde la memoria de usuario es asíncrona respecto a la ejecución del programa de usuario.
  La dirección inicial de los datos que deben transmitirse viene determinada por SD_1. La longitud de los datos de transmisión se define individualmente para cada petición con LEN. LEN sustituye así la parte de SD_1 correspondiente a la longitud.

El parámetro R_ID debe ser idéntico en las instrucciones conexas. Con un flanco ascendente en la entrada de control R se cancela el proceso de transmisión en curso. La finalización correcta del proceso de transmisión se indica con "1" en el parámetro de estado DONE .

Una vez finalizado el proceso de transmisión, no se puede procesar otra petición de transmisión hasta que los parámetros de estado DONE o ERROR no hayan adoptado el valor "1".

Debido a la transmisión de datos asíncrona, no es posible iniciar una nueva transmisión de datos mientras los datos anteriores no hayan sido recogidos por la llamada de la instrucción del interlocutor. Hasta que los datos no hayan sido recogidos, con la llamada de "BSEND" se devuelve el valor de estado "7" (ver a continuación).

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "BSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
R	Input	BOOL	I, Q, M, D, L	El parámetro de control reset activa la cancelación de un intercambio de datos todavía en curso con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
R_ID	Input	DWORD	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669) En caso de acoplamiento vía CP441 a dispositivos S5 o de otros fabricantes, R_ID contiene la información de direccionamiento del dispositivo remoto. Encontrará más información en la descripción del CP441.
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error
STATUS	Output	WORD	I, Q, M, D, L	

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
SD_1	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero al área de memoria. Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse. (p. ej.: P#DB10.DBX5.0 byte 10)
LEN	InOut	WORD	I, Q, M, D, L	Longitud en bytes del bloque de datos que debe transferirse

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de "BSEND", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	• Problemas de comunicación, p. ej. descripción de conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Adicionalmente para el S7-300: Se ha superado el número máximo de peticiones/instancias paralelas.
1	2	Acuse negativo de la instrucción del interlocutor. La función no puede ejecutarse.
1	3	R_ID no es conocido en la conexión de comunicación especificada por ID, o el bloque de recepción no se ha llamado nunca.
1	4	Error en el puntero de área de transmisión SD_1 referente a la longitud o al tipo de datos, o en el parámetro LEN se ha transmitido el valor "0".
1	5	Se ha ejecutado la solicitud de reset.
1	6	La instrucción del interlocutor se encuentra en estado DISABLED (EN_R tiene el valor "0"). Compruebe también si los parámetros de entrada de BRCV son coherentes con BSEND.
1	7	La instrucción del interlocutor está en estado incorrecto Las instrucciones de recepción no se han vuelto a llamar desde la última transmisión de datos.
1	8	Se ha rechazado el acceso al objeto remoto en la memoria de usuario: El área de destino en el "BRCV" correspondiente es demasiado pequeña. En el BRCV correspondiente se notifica ERROR = 1, STATUS = 4 ó ERROR = 1, STATUS = 10.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).

ERROR	STATUS (decimal)	Explicación
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "BSEND" . • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	18	• R_ID ya existe en la conexión. • Adicionalmente para el S7-300: Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP.)
1	20	• S7-400 La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria. • S7-300 – Se ha superado el número máximo de peticiones/instancias paralelas – Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP.) – Es posible en la primera llamada – Escasez de memoria del CP
1	27	Solo con S7-300: Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

Para garantizar la coherencia de datos, no puede volver a escribirse en la parte del área de transmisión SD_1 utilizada actualmente hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

BRCV: Recibir datos en segmentos

Descripción

La instrucción "BRCV" recibe datos de una instrucción remota del interlocutor del tipo "BSEND (Página 7688)". Después de cada segmento de datos recibido se envía un acuse a la instrucción del interlocutor, y el parámetro LEN se actualiza.

La instrucción está lista para recibir tras la llamada con el valor "1" en la entrada de control EN_R. Con EN_R=0 no es posible interrumpir una petición en curso.

La dirección inicial y la longitud máxima del área de recepción vienen dadas por RD_1. La longitud del bloque de datos recibido se indica en LEN .

- S7-300: Con cada flanco ascendente en EN_R se aplican los parámetros R_ID, ID y RD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros R_ID, ID y RD_1. Para transmitir datos segmentados es necesario llamar la instrucción cíclicamente en el programa de usuario.
- S7-400 y S7-300 a través de una interfaz integrada: La recepción de los datos de la memoria de usuario es asíncrona al procesamiento del programa de usuario.

El parámetro R_ID debe ser idéntico en las instrucciones conexas.

La recepción sin errores de todos los segmentos de datos se indica en el parámetro de estado NDR con el valor "1". Los datos recibidos permanecen invariables hasta la siguiente llamada con EN_R=1.

Si se vuelve a llamar la instrucción durante una recepción asíncrona, se devuelve una advertencia en el parámetro de estado STATUS; si la llamada se produce con EN_R=0, el proceso de recepción se interrumpe y la instrucción pasa a su estado inicial.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "BRCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	El parámetro de control enabled to receive señaliza disponibilidad de recepción cuando la entrada está activada.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
R_ID	Input	DWORD	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669) En caso de acoplamiento vía CP441 a dispositivos S5 o de otros fabricantes, R_ID contiene la información de direccionamiento del dispositivo remoto. Encontrará más información en la descripción del CP441.
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
STATUS	Output	WORD	I, Q, M, D, L	

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RD_1	InOut	ANY	S7-300: M, D S7-400: I, Q, M, D, T, C	Puntero al área de recepción. La indicación de longitud indica la longitud máxima del bloque que se va a recibir. Solo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse. (p. ej.: P# DB10.DBX5.0 byte 10).
LEN	InOut	WORD	I, Q, M, D, L	Longitud de los datos recibidos hasta el momento en bytes.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de "BRCV", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: Los datos de recepción ya se están procesando en una clase de menor prioridad.
0	17	Advertencia: La instrucción recibe datos de forma asíncrona. El parámetro LEN indica el número de datos recibidos hasta ahora en bytes.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Adicionalmente para el S7-300: Se ha superado el número máximo de peticiones/instancias paralelas.
1	2	Función no ejecutable (error de protocolo)
1	4	Error en el puntero del área de recepción RD_1 en cuanto a la longitud o el tipo de datos. El bloque de datos enviado supera la longitud del área de recepción.
1	5	Petición de reset recibida, transmisión incompleta.
1	8	Error de acceso en el respectivo "BSEND (Página 7688)": Tras enviar el último segmento de datos válido, se indica ERROR = 1 y STATUS = 4 o ERROR = 1 y STATUS = 10.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "BRCV" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).

ERROR	STATUS (decimal)	Explicación
1	18	• R_ID ya existe en la conexión. • Adicionalmente para el S7-300: Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP.)
1	20	• **S7-400** La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria. • **S7-300** – Se ha superado el número máximo de peticiones/instancias paralelas – Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP.) – Es posible en la primera llamada – Escasez de memoria del CP
1	27	**Solo con S7-300:** Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

Los datos se reciben con coherencia si se observa lo siguiente: Evalúe por completo la parte del área de recepción RD_1 utilizada en ese momento antes de volver a llamar el bloque con el valor "1" en la entrada de control EN_R.

Caso especial de una recepción de datos (sólo S7-400)

Si la CPU receptora pasa a STOP con una instrucción "BRCV" lista para recibir (es decir, ya ha habido una llamada con EN_R =1) antes de que la instrucción de transmisión correspondiente haya enviado el primer segmento de datos de una petición, ocurre lo siguiente:

- Los datos de la primera petición tras la transición de la CPU receptora a STOP se registran por completo en el área de recepción.
- La instrucción del interlocutor "BSEND (Página 7688)" recibe un acuse positivo al respecto.
- La CPU receptora ya no puede aceptar más peticiones "BSEND (Página 7688)" en estado STOP.
- Mientras se encuentre en estado STOP, NDR y LEN tienen el valor "0".

Para no perder la información sobre los datos recibidos, será necesario ejecutar un rearranque de la CPU receptora y llamar "BRCV" con EN_R=1.

C_CNTRL: Consultar el estado de la conexión

Descripción

Con esta instrucción se determina el estado de una conexión **en el S7-300**. Tras la llamada de la instrucción con el valor "1" en la entrada de control EN_R se determina el estado actual de la conexión direccionada con ID.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "C_CNTRL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	El parámetro de control señaliza disponibilidad de recepción cuando la entrada está activada.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
RET_VAL	Output	INT	I, Q, M, D, L	Información de error
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
C_CONN	Output	BOOL	I, Q, M, D, L	Estado de la respectiva conexión. Valores posibles: • 0: Conexión interrumpida o no establecida. • 1: Conexión disponible.
C_STATUS	Output	WORD	I, Q, M, D, L	Estado de la conexión: • W#16#0000: la conexión no está establecida • W#16#0001: la conexión se está estableciendo • W#16#0002: la conexión está establecida • W#16#000F: no hay datos disponibles para el estado de la conexión (p. ej. en el arranque del CP) • W#16#00FF: la conexión no está configurada

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros RET_VAL, ERROR y STATUS

El parámetro de salida RET_VAL puede adoptar los dos siguientes valores con "C_CNTRL":

- 0000H: no se ha producido ningún error durante le ejecución.
- 8000H: se ha producido un error durante le ejecución.

ERROR	STATUS (decimal)	Explicación
1	10	Error durante el acceso al CP, porque hay otra petición en curso. Repetición de la petición con posterioridad.
1	27	Para esta instrucción no existe ningún código de función en la CPU.

Nota

Incluso si en el parámetro de salida RET_VAL se indica el valor 0000H , deben evaluarse los parámetros de salida ERROR y STATUS .

Otras

GET_S: Leer datos de una CPU remota

Descripción

En el S7-300 la instrucción permite leer datos de una CPU remota. La conexión puede realizarse a través de la interfaz integrada o a través de un CP.

El proceso de lectura inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros ID, ADDR_1 y RD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros ID, ADDR_1 y RD_1.

El interlocutor remoto devuelve los contenidos de los datos. Los datos recibidos se copian en la siguiente llamada en las áreas de recepción configuradas (RD_1). Hay que asegurarse de que las áreas definidas por los parámetros ADDR_1 y RD_1 se correspondan entre sí en cuanto a la cantidad, la longitud y el tipo de datos.

El término de este proceso se indica con "1" en el parámetro de estado NDR. No es posible activar otro proceso de lectura hasta que no haya finalizado el último.

La CPU remota puede encontrarse en el estado operativo RUN o STOP. En caso de problemas de acceso al leer los datos, o si la comprobación del tipo de datos devolvió un error, los errores y advertencias correspondientes se indican a través de ERROR y STATUS.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "GET_S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR / STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
ADDR_1	InOut	ANY	M, D	Puntero hacia las áreas de la CPU interlocutora que deben leerse.
RD_1	InOut	ANY	M, D	Puntero hacia las áreas de la CPU propia en las que se depositan los datos leídos. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de la instrucción "GET_S" que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Se ha superado el número máximo de peticiones/instancias paralelas.
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	4	Error en los punteros de áreas de recepción RD_1 en cuanto a la longitud o el tipo de datos.

ERROR	STATUS (decimal)	Explicación
1	8	Error de acceso en la CPU interlocutora.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "GET_S" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	20	• Se ha superado el número máximo de peticiones/instancias paralelas • Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) • Es posible en la primera llamada
1	27	Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

Los datos se reciben con coherencia si se observa lo siguiente:

Evalúe la parte actualmente utilizada del área de recepción RD_1 por completo antes de activar una nueva petición.

PUT_S: Escribir datos en una CPU remota

Descripción

En el S7-300 la instrucción permite escribir datos en una CPU remota. La conexión puede realizarse a través de la interfaz integrada o a través de un CP.

El proceso de transmisión inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros ID, ADDR_1 y SD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros ID, ADDR_1 y SD_1.

El interlocutor remoto deposita los datos transmitidos en las direcciones indicadas y devuelve un acuse de ejecución. Hay que asegurarse de que las áreas definidas por los parámetros ADDR_1 y SD_1 se correspondan entre sí en cuanto a la cantidad, la longitud y el tipo de datos.

Si no se ha producido ningún error, esto se indica en la siguiente llamada con "1" en el parámetro de estado DONE . No es posible activar otro proceso de escritura hasta que no haya finalizado el último.

La CPU remota puede encontrarse en el estado operativo RUN o STOP. En caso de problemas de acceso al escribir los datos, o si la comprobación del acuse de ejecución devolvió un error, los errores y advertencias correspondientes se indican a través de ERROR y STATUS .

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "PUT_S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
ADDR_1	InOut	ANY	M, D	Puntero hacia las áreas de la CPU interlocutora en las que debe escribirse.
SD_1	InOut	ANY	M, D	Puntero hacia las áreas de la CPU propia que contienen los datos que deben transmitirse. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de "PUT_S", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Se ha superado el número máximo de peticiones/instancias paralelas.
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	4	Error en los punteros de área de transmisión SD_1 en cuanto a la longitud o el tipo de datos.
1	8	Error de acceso en la CPU interlocutora
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado)
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "PUT_S" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente)
1	20	• Se ha superado el número máximo de peticiones/instancias paralelas • Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) • Es posible en la primera llamada
1	27	Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

Para garantizar la coherencia de datos, no puede volver a escribirse en el área de transmisión SD_1 hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

USEND_S: Enviar datos de forma no coordinada

Descripción

La instrucción "USEND_S" envía **en el S7-300** datos a una instrucción remota del interlocutor del tipo "URCV_S (Página 7703)". El proceso de transmisión se desarrolla sin coordinación con la instrucción del interlocutor, es decir, que la transmisión de datos se produce sin acuse por parte de la instrucción del interlocutor.

El proceso de transmisión inicia tras un flanco ascendente en REQ. Con cada flanco ascendente en REQ se aplican los parámetros R_ID, ID y SD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros R_ID, ID y SD_1.

Hay que asegurase de que las áreas definidas por los parámetros SD_1 y RD_1 (en la respectiva instrucción del interlocutor "URCV_S (Página 7703)") coincidan en:

- Número
- Longitud
- Tipo de datos

El parámetro R_ID debe ser idéntico en ambas instrucciones.

La finalización correcta del proceso de transmisión se indica con un "1" lógico en el parámetro de estado DONE .

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "USEND_S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
R_ID	Input	DWORD	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
SD_1	InOut	ANY	M, D	Puntero hacia área de transmisión. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Se ha superado el número máximo de peticiones/instancias paralelas
1	4	Error en los punteros de área de transmisión SD_1 en cuanto a la longitud o el tipo de datos.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "USEND_S" • No se ha indicado ningún DB de instancia, sino un DB global • Se ha encontrado un DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	18	• R_ID ya existe en la conexión ID. • Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP).

ERROR	STATUS (decimal)	Explicación
1	20	• Se ha superado el número máximo de peticiones/instancias paralelas • Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) • Es posible en la primera llamada
1	27	Para esta instrucción no existe ningún código de función en la CPU.

Coherencia de datos

Para garantizar la coherencia de datos, no puede volver a escribirse en el área de transmisión SD_1 hasta que no haya finalizado el proceso de transmisión actual. Esto ocurre cuando el parámetro de estado DONE adopta el valor "1".

URCV_S: Recibir datos de forma no coordinada

Descripción

La instrucción "URCV_S" recibe **en el S7-300** datos de forma asíncrona de una instrucción remota del interlocutor del tipo "USEND_S (Página 7700)" y los copia en las áreas de recepción configuradas.

La instrucción está lista para recibir cuando en la entrada EN_R hay un "1" lógico. Con EN_R=0 no es posible interrumpir una petición en curso.

Con cada flanco ascendente en EN_R se aplican los parámetros R_ID, ID y RD_1. Una vez finalizada una petición se pueden asignar nuevos valores a los parámetros R_ID, ID y RD_1.

Hay que asegurarse de que las áreas definidas por los parámetros RD_1 y SD_1 (en la respectiva instrucción del interlocutor "USEND_S (Página 7700)") coincidan en:

- Número
- Longitud
- Tipo de datos

La finalización correcta del proceso de copia se indica con un "1" lógico en el parámetro de estado NDR . El parámetro R_ID debe ser idéntico en ambas instrucciones.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "URCV_S":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	El parámetro de control enabled to receive señaliza disponibilidad de recepción cuando la entrada está activada.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
R_ID	Input	DWORD	I, Q, M, D, L o constante	Parámetro de direccionamiento R_ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
RD_1	InOut	ANY	M, D	Puntero hacia el área de recepción n.º i. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, COUNTER, TIMER. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P#DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (decimal)	Explicación
0	9	Advertencia Overrun: Los datos de recepción antiguos han sido sobrescritos por datos de recepción más actuales.
0	11	Advertencia: Los datos de recepción ya se están procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.

ERROR	STATUS (decimal)	Explicación
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto) • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP) • Conexión con el interlocutor todavía no establecida • Se ha superado el número máximo de peticiones/instancias paralelas
1	4	Error en los punteros de área de recepción RD_1 en cuanto a la longitud o el tipo de datos.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "URCV_S". • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	18	• R_ID ya existe en la ID de conexión • Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP)
1	19	La instrucción correspondiente "USEND_S (Página 7700)" envía datos más rápido de lo que "URCV_S" los puede copiar en las áreas de recepción.
1	20	• Se ha superado el número máximo de peticiones/instancias paralelas • Las instancias se han sobrescrito con RUN de la CPU (se requiere una transición STOP-RUN de la CPU o del CP) • Es posible en la primera llamada
1	27	Para este bloque no existe ningún código de función en la CPU.

Coherencia de datos

Los datos se reciben con coherencia si se observa lo siguiente:

Una vez que el parámetro de estado NDR ha adoptado el valor "1", hay que volver a llamar inmediatamente "URCV_S" con el valor "0" en EN_R. Con ello queda garantizado que el área de recepción no se sobrescriba antes de haberse evaluado. Evalúe el área de recepción RD_1 por completo antes de volver a llamar el bloque con el valor "1" en la entrada de control EN_R.

CONTROL: Consultar estado de la conexión con la instancia de una instrucción

Descripción

Con la instrucción "CONTROL" se determina **en el S7-400** el estado de la conexión que pertenece a una instancia local de una instrucción de comunicación.

Tras la llamada de la instrucción con el valor "1" en la entrada de control EN_R se determina el estado actual de la conexión que pertenece a la instancia de la instrucción de comunicación seleccionada con I_DB.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "CONTROL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	El parámetro de control señaliza disponibilidad de recepción cuando la entrada está activada.
I_DB	Input	WORD	I, Q, M, D, L o constante	Número del DB de instancia
OFFSET	Input	WORD	I, Q, M, D, L o constante	Offset del registro en bytes en el DB multiinstancia (si no hay ningún DB multiinstancia, aquí debe indicarse "0").
RET_VAL	Return	INT	I, Q, M, D, L	Información de error
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
I_TYP	Output	BYTE	I, Q, M, D, L	Identificación del tipo de bloque que pertenece a la instancia seleccionada.
I_STATE	Output	BYTE	I, Q, M, D, L	• = 0: La instancia correspondiente de la instrucción no se ha llamado desde el último arranque en frío, rearranque completo (en caliente) o carga. • <> 0: La instancia correspondiente de la instrucción se ha llamado como mínimo una vez desde el último arranque en frío, rearranque completo (en caliente) o carga.
I_CONN	Output	BOOL	I, Q, M, D, L	Estado de la respectiva conexión. Valores posibles: • 0: Conexión interrumpida o no establecida. • 1: Conexión disponible.
I_STATUS	Output	WORD	I, Q, M, D, L	Parámetro de estado STATUS de la instancia de la instrucción de comunicación consultada.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro de salida I_TYP

La siguiente tabla muestra qué identificación pertenece a qué tipo de instrucción.

Tipo de instrucción	Identificación (W#16#...)
USEND (Página 7683)	00
URCV (Página 7685)	01

Tipo de instrucción	Identificación (W#16#...)
BSEND (Página 7688)	04
BRCV (Página 7691)	05
GET (Página 7677)	06
PUT (Página 7680)	07
PRINT (Página 7708)	08
START (Página 7714)	0B
STOP (Página 7716)	0C
RESUME (Página 7718)	0D
STATUS (Página 7720)	0E
USTATUS (Página 7722)	0F
ALARM (Página 6930)	15
ALARM_8 (Página 6923)	16
ALARM_8P (Página 6925)	17
NOTIFY (Página 6928)	18
AR_SEND (Página 6950)	19
NOTIFY_8P (Página 6920)	1A
(ninguna instrucción disponible; I_DB o OFFSET incorrectos)	FF

Parámetros RET_VAL, ERROR y STATUS

El parámetro de salida RET_VAL puede adoptar los dos siguientes valores con la instrucción "CONTROL":

- 0000H: no se ha producido ningún error durante le ejecución.
- 8000H: se ha producido un error durante le ejecución.

ERROR	STATUS (decimal)	Explicación
1	10	El acceso a la memoria de usuario local no es posible (p. ej.: Como parámetro actual para I_TYP se ha indicado un byte de marcas que no existe en la CPU utilizada.)
1	12	Al llamar la instrucción • en el parámetro I_DB se ha indicado un DB de instancia que no pertenece a la instrucción "CONTROL". • no se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).

Nota

Incluso si en el parámetro de salida RET_VAL se indica el valor 0000H , deben evaluarse los parámetros de salida ERROR y STATUS .

PRINT: Enviar datos a una impresora

Descripción de PRINT

Descripción

La instrucción "PRINT" permite enviar **en el S7-400** datos junto con una instrucción de formato a una impresora remota, p. ej. con el CP 441.

Con un flanco ascendente en la entrada de control REQ se envían la descripción de formato (parámetro FORMAT) y los datos (SD_i) a la impresora seleccionada con ID y PRN_NR .

Si no se utilizan las cuatro áreas de transmisión, asegúrese de escribir la primera área con el parámetro SD_1, la segunda (si procede) con SD_2, la tercera (si procede) con SD_3.

La finalización correcta de la petición se indica en el parámetro de estado DONE con "1", y los errores ocurridos se indican a través de los parámetros de estado ERROR y STATUS.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PRINT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa el intercambio de datos con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error
PRN_NR	InOut	BYTE	I, Q, M, D, L	Número de impresora

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
FORMAT (Página 7710)	InOut	STRING	I, Q, M, D, L	Descripción de formato
SD_i (1≤ i ≤4)	InOut	ANY	M, D, T, C	Puntero al área de transmisión n.º i. Sólo se permiten los tipos de datos BOOL (no permitido: mapa de bits), BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME. **Nota:** Si el puntero ANY accede a un DB, éste siempre debe especificarse (p. ej.: P# DB10.DBX5.0 byte 10).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de los errores específicos de "PRINT", que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP).
1	2	Acuse negativo de la impresora. La función no puede ejecutarse.
1	3	PRN_NR no es conocido en la conexión de comunicación especificada por ID.
1	4	Error en el parámetro de entrada/salida FORMAT o en los punteros de área de transmisión SD_i en cuanto a la longitud o el tipo de datos.
1	6	La impresora remota está en estado OFFLINE.
1	7	La impresora remota se encuentra en un estado erróneo (p. ej. paper out).
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	Al llamar la instrucción • se ha indicado un DB de instancia que no pertenece a "PRINT". • no se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	13	Error en el parámetro de entrada/salida FORMAT (Página 7710).
1	20	**S7-400** La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria.

Número de datos transmisibles

El número de datos transmisibles a una impresora remota no puede superar una longitud máxima. La longitud de datos máxima en bytes se calcula de la siguiente manera:
`maxleng = 420 - format`

`format` es la longitud actual del parámetro FORMAT en bytes. Es irrelevante si los datos que se van a imprimir se reparten entre una o varias áreas de transmisión.

Parámetro FORMAT

Descripción

La cadena de caracteres FORMAT contiene los caracteres y elementos de formato que deben imprimirse. Su estructura es la siguiente:

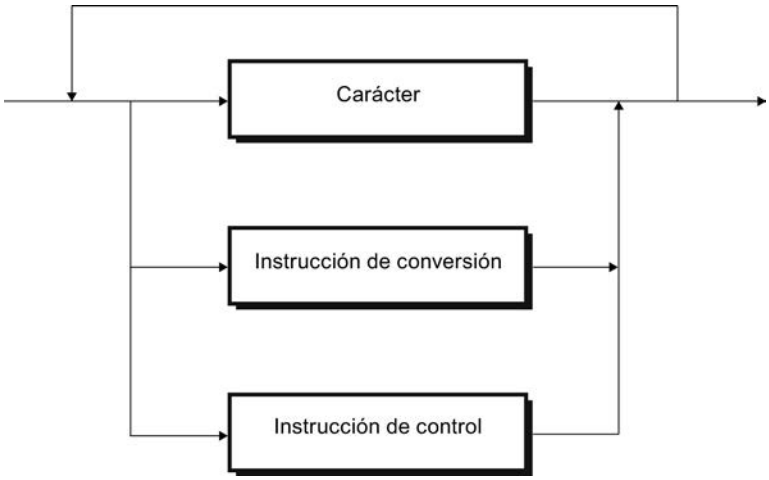


Figura 9-11 Estructura de caracteres y elementos de formato

Para cada área de transmisión que debe imprimirse SD_1 a SD_4 debe haber exactamente una instrucción de conversión en FORMAT. Las instrucciones de conversión se aplican a las áreas de transmisión SD_i siguiendo su orden. Además, los caracteres y las instrucciones pueden figurar en cualquier orden.

Carácter

Son admisibles:

- Todos los caracteres imprimibles
- $$ (símbolo del dólar), $' (signo de exclamación sencillo), $L y $l (avance de línea), $P y $p (página), $R y $r (retorno de carro), $T y $t (tabulador)

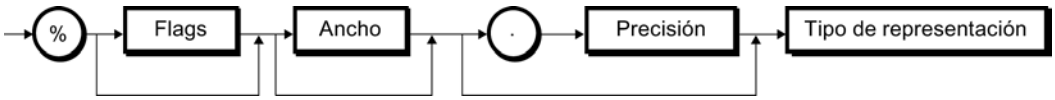


Figura 9-12 Diagrama sintáctico de una instrucción de conversión

Elemento de una instrucción de conversión	Significado	
Flags	• Sin: • -:	Impresión con sangría a derecha Impresión con sangría a izquierda
Ancho	• Sin: • n:	La impresión se realiza en la representación estándar Se imprimen exactamente n caracteres. En una impresión con sangría a la derecha, en caso necesario se anteponen espacios, mientras que con sangría a la izquierda se colocan al final.
Precisión	La precisión sólo es relevante para los tipos de representación A, D, F y R (ver la tabla siguiente).	
	• Sin: • 0: • n:	La impresión se realiza en representación estándar Sin impresión del punto decimal ni de decimales en los tipos de representación F y R • Con F y R: Impresión del punto decimal y de n decimales. • Con A y D (fecha): Número de posiciones en el número de año. Valores posibles: 2 y 4.
Tipo de representación	La tabla siguiente contiene • Los tipos de representación posibles • Los tipos de datos posibles para cada tipo de representación • Para cada tipo de representación, la representación estándar (la impresión se realiza en representación estándar si en el parámetro FORMAT no se indica el ancho ni la precisión) y la longitud máxima.	

Instrucción de conversión

La tabla siguiente muestra los tipos de representación posibles en la instrucción de conversión del parámetro FORMAT.

Tipo de representación	Tipos de datos posibles	Ejemplo	Longitud	Observaciones
A, a	DATE	25.07.1996	10	-
	DWORD			
C, c	CHAR	K	1	-
	BYTE	M	1	
	WORD	KL	2	
	DWORD	KLMN	4	
	ARRAY of CHAR	KLMNOP	Número de caracteres	
	ARRAY of BYTE			
D, d	DATE	1996-07-25	10	-
	DWORD			
F, f	REAL	0.345678	8	-
	DWORD			

Tipo de representación	Tipos de datos posibles	Ejemplo	Longitud	Observaciones
H, h	Todos los tipos de datos incluido ARRAY of BYTE	según el tipo de datos	según el tipo de datos	Representación hexadecimal
I, i	INT	- 32 768	Máx. 6	-
	WORD	- 2 147 483 648	Máx. 11	
N, n	WORD	Impresión de texto	-	El área de transmisión SD_i correspondiente contiene una referencia (número) a un texto que debe imprimirse. El texto se encuentra en el módulo (p. ej. CP 441) que genera una cadena de caracteres imprimible. Se imprime **** si no se encuentra ningún texto con el número indicado.
R, r	REAL	0.12E-04	8	-
	DWORD			
S, s	STRING	Impresión de texto		-
T, t	TIME	2d_3h_10m_5s_250ms	Máx. 21	Se imprime **** en caso de error.
	DWORD			
U, u	BYTE	255	Máx. 3	-
	WORD	65 535	Máx. 5	
	DWORD	4 294 967 295	Máx. 10	
X, x	BOOL	1	1	-
	BYTE	101 ..	8	
	WORD	101 ..	16	
	DWORD	101 ..	32	
Z, z	TIME_OF_DAY (TOD)	15:38:59.874	12	-

En los puntos de esta tabla en los que se indica una longitud máxima para la representación estándar, también es posible que la impresión sea más corta.

Nota

Con los tipos de datos C y S depende de la impresora utilizada

- qué caracteres podrán imprimirse
- qué imprimirá la impresora para caracteres no imprimibles, a no ser que el driver de la impresora contenga una tabla de conversión para esos caracteres.

Instrucción de control

Con ayuda de las instrucciones de control es posible:

- Imprimir los caracteres % y \
- Modificar ajustes de impresora

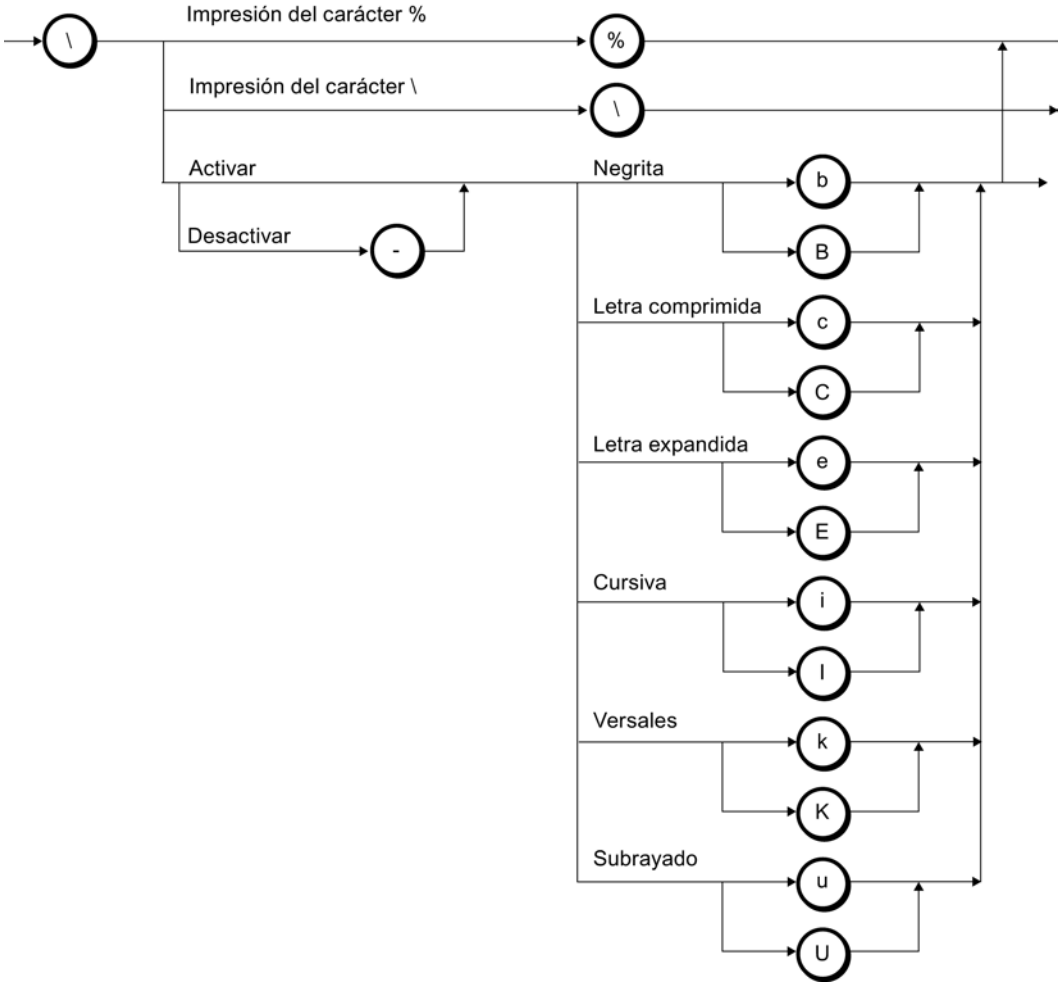


Figura 9-13 Diagrama sintáctico de la instrucción de control

Si se solicita por ejemplo la desactivación de una fuente, a pesar de que ésta no está activada, o si se pretende ejecutar una función desconocida por la impresora, la instrucción de control se ignora. La tabla siguiente contiene los errores en el parámetro de entrada/salida FORMAT.

Error	Impresión
Instrucción de conversión no ejecutable	Conforme a la longitud (máx.) de la representación estándar o del ancho indicado se imprimen caracteres *.
Ancho indicado insuficiente	En los tipos de representación A, C, D, N, S, T, Z se imprimen tantos caracteres como determine el ancho indicado. Para el resto de tipos de representación se imprimen caracteres * conforme al ancho indicado.
Demasiadas instrucciones de conversión	Las instrucciones de conversión a las que no pertenece ningún puntero de área de transmisión SD_i se ignoran.

Error	Impresión
Instrucciones de conversión insuficientes	Las áreas de transmisión para las que no existe una instrucción de conversión no se imprimen.
Instrucciones de conversión no definidas o no soportadas	Se imprime ******.
Instrucción de conversión incompleta	Se imprime ******.
Instrucciones de control no definidas o no soportadas	Las instrucciones de control que no se correspondan con la sintaxis de la imagen anterior se ignoran.

START: Ejecutar un rearranque completo o arranque en frío en un dispositivo remoto

Descripción

Con un flanco ascendente en la entrada de control REQ, la instrucción "START" activa **en el S7-400** un rearranque completo (en caliente) o un arranque en frío en el dispositivo remoto direccionado por ID.

Para la ejecución en el dispositivo remoto (si se trata de una CPU) deben cumplirse las siguientes condiciones:

- La CPU está en estado operativo STOP.
- El selector de modo de la CPU debe estar en "RUN".

Una vez que éste ha realizado el rearranque completo (en caliente) o un arranque en frío, pasa al estado operativo RUN y envía un acuse de ejecución positivo. Con la evaluación del acuse positivo, el parámetro de estado DONE se pone a "1". Los errores ocurridos se indican a través de los parámetros de estado ERROR y STATUS.

Una nueva activación de un rearranque completo (en caliente) o un arranque en frío en el mismo dispositivo remoto no será posible hasta que no haya finalizado la última activación.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "START":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa la instrucción con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
PI_NAME	InOut	ANY	I, Q, M, D, T, C	Puntero hacia el área de memoria en la que se encuentra el nombre del programa que debe iniciarse (código ASCII). El nombre no puede constar de más de 32 caracteres. Con un sistema estándar de la familia S7 debe ser P_PROGRAM.
ARG	InOut	ANY	I, Q, M, D, T, C	Argumento de ejecución. • Si no se asigna ningún valor a ARG en el dispositivo remoto se ejecutará un rearranque completo (en caliente). • Si se asigna el valor "C", en el dispositivo remoto se ejecutará un arranque en frío (si el dispositivo remoto conoce este tipo de arranque).
IO_STATE	InOut	BYTE	I, Q, M, D, L	Actualmente irrelevante. A este parámetro es posible no asignarle ningún parámetro siempre que el interlocutor sea un sistema de automatización de la familia S7.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de los errores específicos de START , que puede indicarse a través de los parámetros ERROR y STATUS .

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP).
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.

ERROR	STATUS (decimal)	Explicación
1	3	El nombre de programa guardado en PI_NAME es desconocido.
1	4	Error en los punteros PI_NAME o ARG en cuanto a la longitud o el tipo de datos.
1	7	En el dispositivo interlocutor no puede ejecutarse un rearranque completo.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "START" • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	20	S7-400: La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria.

STOP: Poner el dispositivo remoto en el estado operativo STOP

Descripción

Con un flanco ascendente en la entrada de control REQ , la instrucción "STOP" activa **en el S7-400** el paso del dispositivo remoto direccionado por ID al estado operativo STOP; el cambio de estado operativo es posible si el dispositivo se encuentra en RUN, PARADA o arranque.

La finalización correcta de la petición se indica en el parámetro de estado DONE con "1", y los errores ocurridos se indican a través de los parámetros de estado ERROR y STATUS.

Una nueva ejecución de la transición de estado operativo descrita en el mismo dispositivo remoto sólo será posible cuando haya finalizado el lanzamiento anterior de la instrucción "STOP".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "STOP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa la instrucción con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
PI_NAME	InOut	ANY	I, Q, M, D	Puntero hacia el área de memoria en la que se encuentra el nombre del programa que debe pararse (código ASCII). El nombre no puede constar de más de 32 caracteres. Con un sistema estándar de la familia S7 debe ser P_PROGRAM.
IO_STATE	InOut	BYTE	I, Q, M, D, L	Actualmente irrelevante. A este parámetro es posible no asignarle ningún parámetro siempre que el interlocutor sea un sistema de automatización de la familia S7.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información sobre los errores específicos de la instrucción "STOP" y que puede indicarse a través de los parámetros ERRORy STATUS.

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP).
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	3	El nombre de programa guardado en PI_NAME es desconocido.
1	4	Error en el puntero PI_NAME en cuanto a la longitud o el tipo de datos.
1	7	El dispositivo interlocutor ya se encuentra en estado STOP.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).

ERROR	STATUS (decimal)	Explicación
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "". • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	20	S7-400: La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria.

RESUME: Ejecutar un rearranque en un dispositivo remoto

Descripción

Con un flanco ascendente en la entrada de control REQ, la instrucción "RESUME" activa **en el S7-400** un rearranque en el dispositivo remoto direccionado por ID. Si el dispositivo remoto es una CPU, deben cumplirse las siguientes condiciones:

- La CPU está en estado operativo STOP.
- El selector de modo de la CPU debe estar en RUN.
- Es necesario haber permitido el rearranque manual en la configuración.
- No puede haber ningún impedimento para el rearranque.

Una vez que se ha realizado el rearranque, pasa al estado operativo RUN y transmite un acuse de ejecución positivo. Con la evaluación del acuse positivo, el parámetro de estado DONE se pone a "1". Los errores ocurridos se indican a través de los parámetros de estado ERROR y STATUS.

Una nueva activación de un rearranque en el mismo dispositivo remoto no será posible hasta que no haya finalizado la última activación.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "RESUME":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa la instrucción con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: Advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
PI_NAME	InOut	ANY	I, Q, M, D	Puntero hacia el área de memoria en la que se encuentra el nombre del programa que debe iniciarse (código ASCII). El nombre no puede constar de más de 32 caracteres. En el S7 debe ser P_PROGRAM.
ARG	InOut	ANY	I, Q, M, D, T, C	Argumento de ejecución. Actualmente irrelevante. A este parámetro es posible no asignarle ningún parámetro siempre que el interlocutor sea un sistema de automatización de la familia S7.
IO_STATE	InOut	BYTE	I, Q, M, D, L	Actualmente irrelevante. A este parámetro es posible no asignarle ningún parámetro siempre que el interlocutor sea un sistema de automatización de la familia S7.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de los errores específicos de "RESUME", que puede indicarse a través de los parámetros ERROR y STATUS.

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	• Problemas de comunicación, p. ej. descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP).
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	3	El nombre de programa guardado en PI_NAME es desconocido.
1	4	Error en los punteros PI_NAME o ARG en cuanto a la longitud o el tipo de datos.
1	7	Rearranque no ejecutable.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).

ERROR	STATUS (decimal)	Explicación
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a RESUME. • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	20	S7-400: La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria.

STATUS: Consultar el estado de dispositivo de un interlocutor remoto

Descripción

La instrucción "STATUS" permite consultar **en el S7-400** el estado del dispositivo de un interlocutor remoto.

Con un flanco ascendente en la entrada de control REQ se envía una petición al interlocutor remoto. La respuesta se evalúa para comprobar si ha habido algún problema. Si no ha ocurrido ningún error, el estado recibido se copia en las variables de los parámetros PHYS, LOG y LOCAL en la siguiente llamada. El término de este proceso se indica con "1" en el parámetro de estado NDR.

No es posible volver a realizar una consulta en el mismo interlocutor hasta que no haya finalizado la última.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "STATUS":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	El parámetro de control request activa la instrucción con un flanco ascendente.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Véase también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error <> 0000H: advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
PHYS	InOut	ANY	I, Q, M, D	Estado físico (longitud mínima: un byte) Valores posibles: • 10H funciona correctamente. • 13H requiere trabajos de servicio.
LOG	InOut	ANY	I, Q, M, D	Estado lógico (longitud mínima: un byte) Valor posible: • 00H cambio de estado permitido.
LOCAL	InOut	ANY	I, Q, M, D	Estado operativo en caso de que el dispositivo interlocutor sea una CPU S7 (longitud mínima: dos bytes).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro de entrada/salida LOCAL

Si el interlocutor es una CPU S7, el parámetro de entrada/salida LOCAL contiene su estado operativo actual: El primer byte está reservado, el segundo byte contiene una identificación del estado operativo.

Estado operativo	Identificación correspondiente
STOP	00H
ARRANQUE (rearranque completo)	01H
RUN	02H
ARRANQUE (rearranque)	03H
PARADA	04H
ARRANQUE (arranque en frío)	06H
RUN	09H
ACOPLAR	0BH
SINCRONIZAR	0CH

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de los errores específicos de STATUS, que puede indicarse a través de los parámetros ERROR y STATUS.

ERROR	STATUS (decimal)	Explicación
0	11	Advertencia: • La nueva petición no tiene efecto, ya que la anterior todavía no ha finalizado. • La petición ya se está procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP).
1	2	Acuse negativo del interlocutor. La función no puede ejecutarse.
1	4	Error en PHYS, LOG, o LOCAL en cuanto a la longitud o el tipo de datos.
1	8	Se ha rechazado el acceso a un objeto remoto.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a STATUS. • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente).
1	20	S7-400: La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria.

USTATUS: Recepción no coordinada de un estado de dispositivo remoto

Descripción

La instrucción "USTATUS" recibe **en el S7-400** el cambio de estado del dispositivo de un interlocutor remoto; si se ha configurado así, el interlocutor envía su estado en caso de cambio sin que se le solicite.

Si en la llamada hay EN_R 1 en la entrada de control y hay un telegrama del interlocutor, en la siguiente llamada se deposita la información de estado en las variables de los parámetros PHYS, LOG y LOCAL. El término de este proceso se indica con "1" en el parámetro de estado NDR.

En la conexión utilizada por la instrucción "USTATUS" debe estar habilitada la transmisión de avisos sobre el estado operativo.

Nota

Por cada conexión sólo se puede posicionar una instancia de la instrucción "USTATUS".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "USTATUS":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L	El parámetro de control (enabled to receive) señaliza la disponibilidad de recepción cuando la entrada está activada.
ID	Input	WORD	M, D o constante	Parámetro de direccionamiento ID Consulte también: Parámetros comunes de las instrucciones para la comunicación S7 (Página 7669)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • 0: La petición aún no se ha iniciado o todavía está en curso. • 1: La petición ha finalizado correctamente.
ERROR STATUS	Output Output	BOOL WORD	I, Q, M, D, L I, Q, M, D, L	Parámetros de estado ERROR y STATUS, indicación de error: • ERROR=0 STATUS tiene el valor: 0000H: ni advertencia ni error 0000H: advertencia, STATUS devuelve información detallada. • ERROR=1 Hay un error. STATUS devuelve información detallada sobre el tipo de error.
PHYS	InOut	ANY	I, Q, M, D	Estado físico (longitud mínima: un byte) Valores posibles: • 10H funciona correctamente. • 13H requiere trabajos de servicio.
LOG	InOut	ANY	I, Q, M, D	Estado lógico (longitud mínima: un byte) Valor posible: • 00H cambio de estado permitido.
LOCAL	InOut	ANY	I, Q, M, D	Estado operativo en caso de que el dispositivo interlocutor sea una CPU S7 (longitud mínima: dos bytes).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro de entrada/salida LOCAL

Si el interlocutor es una CPU S7, el parámetro de entrada/salida LOCAL contiene su estado operativo actual: El primer byte está reservado, el segundo byte contiene una identificación del estado operativo.

Estado operativo	Identificación correspondiente
STOP	00H
ARRANQUE (rearranque completo)	01H

Estado operativo	Identificación correspondiente
RUN	02H
ARRANQUE (rearranque)	03H
PARADA	04H
ARRANQUE (arranque en frío)	06H
RUN	09H
ACOPLAR	0BH
SINCRONIZAR	0CH

Parámetros ERROR y STATUS

La siguiente tabla contiene toda la información de error específica de "USTATUS", que puede indicarse a través de los parámetros ERROR y STATUS.

ERROR	STATUS (decimal)	Explicación
0	9	Advertencia Overrun: Un estado de dispositivo anterior ha sido sobrescrito con un estado más actual.
0	11	Los datos de recepción ya se están procesando en una clase de menor prioridad.
0	25	La comunicación se ha iniciado. La petición está en proceso.
1	1	Problemas de comunicación, p. ej. • Descripción de la conexión no cargada (local o remoto). • Conexión interrumpida (p. ej. cable, CPU desconectada, CP en STOP).
1	4	Error en PHYS, LOG, o LOCAL en cuanto a la longitud o el tipo de datos.
1	10	El acceso a la memoria de usuario local no es posible (p. ej. acceso a un DB borrado).
1	12	En la llamada de la instrucción ha ocurrido lo siguiente: • Se ha indicado un DB de instancia que no pertenece a "USTATUS". • No se ha indicado ningún DB de instancia, sino un DB global • No se ha encontrado ningún DB de instancia (solución: vuelva a cargar el DB de instancia correspondiente)
1	18	Para la conexión especificada por ID ya existe una instancia de "USTATUS".
1	19	La CPU remota transmite datos más rápido de lo que la instrucción puede aplicarlos en el programa de usuario.
1	20	S7-400: La memoria de trabajo disponible no es suficiente. Solución: Reduzca el código del programa que se encuentra en la memoria.

Open User Communication

Vista general

Open User Communication vía Industrial Ethernet

Para poder intercambiar datos con otros interlocutores vía Ethernet (protocolo TCP, ISO-on-TCP y UDP) a través del programa de usuario, se ponen a disposición las siguientes instrucciones y UDTs:

- Protocolos orientados a la conexión: TCP según RFC 793, ISO on TCP según RFC 1006:
  - UDT 65 "TCON_PAR (Página 7735)" con la estructura de datos para parametrizar la conexión
  - UDT 651 hasta UDT 656 con la asignación de valores predeterminados específica del protocolo
  - "TCON (Página 7748)" para establecer la conexión
  - "TDISCON (Página 7751)" para deshacer la conexión
  - "TSEND (Página 7753)" para enviar datos
  - "TRCV (Página 7756)" para recibir datos
- Protocolo orientado a la no-conexión: UDP según RFC 768
  - UDT 65 "TCON_PAR (Página 7735)" con la estructura de datos para parametrizar el punto de acceso local de comunicación
  - UDT 657 con la asignación de valores predeterminados específica del protocolo
  - UDT 66 "TCON_ADR (Página 7737)" con la estructura de datos de los parámetros de direccionamiento del interlocutor remoto
  - UDT 661 con la asignación de valores predeterminados específica del protocolo
  - "TCON (Página 7748)" para crear el punto de acceso local de comunicación
  - "TDISCON (Página 7751)" para deshacer el punto de acceso local de comunicación
  - "TUSEND (Página 7760)" para enviar datos
  - "TURCV (Página 7763)" para recibir datos

Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet

Protocolos orientados a la conexión y a la no-conexión

En la comunicación de datos se distingue entre las siguientes variantes de protocolo:

- Variantes de protocolo orientadas a la conexión:
  establecen una conexión (lógica) con el interlocutor antes de la transferencia de datos y, dado el caso, la deshacen una vez que ésta ha concluido. Los protocolos orientados a la conexión se utilizan especialmente cuando lo importante en la transferencia de datos es la seguridad. A través de una línea física es posible establecer, por lo general, varias conexiones lógicas.
  En las instrucciones para Open User Communication vía Industrial Ethernet se soportan los siguientes protocolos orientados a la conexión:
  - TCP según RFC 793
  - ISO on TCP según RFC 1006
- Variantes de protocolo orientadas a la no-conexión:
  trabajan sin conexión. Así, no se establece ni deshace la conexión con el interlocutor remoto. Los protocolos orientados a la no-conexión transmiten datos al interlocutor remoto sin acuse y, por lo tanto, de forma insegura; esto significa que se pueden extraviar datos sin que su pérdida se indique en el bloque.
  En las instrucciones para Open User Communication vía Industrial Ethernet se soportan los siguientes protocolos orientados a la no-conexión:
  - UDP según RFC 768 (broadcast o singlecast; no se soporta multicast)

El funcionamiento de las instrucciones depende de la variante de protocolo utilizada. Este tema se describe más adelante.

Área de recepción

Este término se utiliza repetidamente en los siguientes apartados. Se entiende como tal el área en la que la instrucción registra los datos recibidos.

El área de recepción está definida por los dos valores siguientes:

- Puntero hacia el inicio del área
- Longitud del área

En función de la variante de protocolo utilizada, la longitud del área se fija con el parámetro LEN (si LEN <> 0) o por la indicación de longitud del parámetro DATA (si LEN = 0).

En el puntero ANY se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME.

TCP

Al transferir datos no se transfiere ni información relacionada con la longitud de un mensaje ni con su principio o fin. Esto no supone un problema a la hora de enviar, ya que el emisor sabe cuántos bytes de datos quiere enviar. Sin embargo, el receptor no tiene ninguna posibilidad de detectar dónde termina un mensaje dentro del flujo de datos y dónde empieza el siguiente. Por ello se recomienda elegir el área de recepción de la instrucción "TRCV

(Página 7756)" de un tamaño equivalente al valor del parámetro LEN de la instrucción "TRCV (Página 7753)" en el interlocutor (número de bytes, que se deben enviar).

- Recepción de datos en modo Ad hoc:
  El área de recepción es idéntica al área especificada por el parámetro DATA de la instrucción "TRCV (Página 7756)".
  - Inmediatamente después de la recepción de un bloque de datos, la instrucción "TRCV (Página 7756)" lo transmite al área de recepción y pone NDR a "1". La longitud máxima es de 8192 bytes.
  - Si ha elegido una longitud del área de recepción más grande que la longitud de los datos enviados, la instrucción "TRCV (Página 7756)" copia por completo los datos recibidos en el área de recepción. A continuación, pone NDR a TRUE y escribe en RCVD_LEN la longitud de los datos recibidos.
  - Si ha elegido una longitud del área de recepción más pequeña que la longitud de los datos enviados, "TRCV (Página 7756)" copiará tantos bytes en el área de recepción como quepan hasta llenarla. A continuación, pone NDR a TRUE y escribe en RCVD_LEN la longitud del área de recepción. Con cada llamada posterior se obtiene un bloque adicional de los datos enviados.
- Recepción de datos con longitud especificada:
  - El área de recepción está formada por los parámetros DATA (dirección inicial del área de recepción) y LEN (longitud del área de recepción) de "TRCV (Página 7756)".
  - Si se reciben datos que no llenan completamente el área de recepción, en un principio éstos no pueden ponerse a disposición. No se pondrán a disposición hasta que otros datos hayan llenado completamente el área de recepción. Tenga en cuenta que, en este caso, hay datos de dos peticiones de envío distintas en la misma área de recepción. Si no puede definir el final del primer mensaje o el inicio del segundo mensaje, no tiene posibilidad alguna de reconocer el primer o el segundo mensaje.
  - Si ha elegido una longitud del área de recepción más pequeña que la longitud de los datos enviados, "TRCV (Página 7756)" copiará tantos bytes en el área de recepción como quepan hasta llenarla.
  - Si "TRCV (Página 7756)" ha llenado completamente el área de recepción, pone NDR a TRUE y escribe en RCVD_LEN el valor de LEN. Con cada llamada posterior se obtiene así un bloque adicional de los datos enviados.

ISO on TCP

Al transferir datos se transfieren informaciones relacionadas con la longitud y el fin de un mensaje.

Si ha elegido un área recepción más grande que la longitud de los datos enviados, la instrucción "TRCV (Página 7756)" copia completamente los datos enviados en el área de recepción. A continuación, pone NDR a TRUE y escribe en RCVD_LEN la longitud de los datos enviados.

Si ha elegido un área de recepción más pequeña que la longitud de los datos enviados, "TRCV (Página 7756)" no copia los datos en el área de recepción, sino que devuelve la siguiente información de error: ERROR=1, STATUS=W#16#8088.

UDP

No se establece una conexión como con los protocolos TCP y ISO on TCP. Por esa razón, al llamar el bloque de envío "TUSEND (Página 7760)" se tiene que indicar una referencia a los parámetros de dirección del receptor (dirección IP y n° de puerto). Paralelamente, se recibe una referencia a los parámetros de dirección del emisor (dirección IP y n° de puerto) tras concluir el bloque de recepción "TURCV (Página 7763)".

- Para poder utilizar las instrucciones "TUSEND (Página 7760)" y "TURCV (Página 7763)" se tiene que llamar con anterioridad la instrucción "TCON (Página 7748)" tanto en el lado del emisor como en el del receptor para configurar el punto de acceso local de comunicación.
- En cada llamada de "TUSEND (Página 7760)" se puede referenciar de nuevo al interlocutor remoto indicando su dirección IP y su n° de puerto.
- Al transferir datos se transfieren informaciones relacionadas con la longitud y el fin de un mensaje.

Si ha elegido un área recepción más grande que la longitud de los datos enviados, "TURCV (Página 7763)" copia completamente los datos enviados en el área de recepción. A continuación, pone NDR a TRUE y escribe en RCVD_LEN la longitud de los datos enviados.

Si ha elegido un área recepción más pequeña que la longitud de los datos enviados, "TURCV" no copia los datos en el área de recepción, sino que devuelve la siguiente información de error: ERROR = 1, STATUS = W#16#8088.

Números de puerto en TCP y UDP

El direccionamiento de una conexión se realiza así:

- Con TCP:
  - Conexión activa: A través de la dirección IP remota, el puerto remoto y el puerto local (el número de puerto del puerto local se debe especificar explícitamente o será proporcionado por el sistema operativo).
  - Conexión pasiva: A través del puerto local. Si desea utilizar más el puerto local de una vez, tiene que especificar también o la dirección IP remota o la dirección IP remota y el puerto remoto.
- Con UDP: A través de la dirección IP remota, el puerto remoto y el puerto local (el número de puerto del puerto local se debe especificar explícitamente).

Para el puerto local valen las áreas mencionadas a continuación. Para el puerto remoto, esas áreas sólo son válidas si el interlocutor remoto es una CPU S7. De otro modo no hay restricciones para el puerto remoto.

- Para CPU 31x-2 PN/DP hasta la versión de firmware V2.6 incluida y CPU 41x3 PN/DP hasta la versión de firmware V5.1 incluida:
  - Números de puerto permitidos en la parametrización (UDT 65): de 2000 hasta 5000
  - En parametrizaciones sin un número de puerto predeterminado, el sistema operativo asigna un puerto del rango numérico dinámico comprendido entre 49152 y 65534.
- Para CPU 31x-2 PN/DP a partir de la versión de firmware V2.7 y CPU 41x-3 PN/DP a partir de la versión de firmware V5.2:
  - Números de puerto permitidos en la parametrización (UDT 65): 1 a 49151
    Se recomienda limitarse a los números de puerto 2000 a 5000, ya que Siemens se reserva utilizar en el sistema puertos del rango numérico de 1 hasta 1999 y 5001 hasta 49151.
  - En parametrizaciones sin un número de puerto predeterminado, el sistema operativo asigna un puerto del rango numérico dinámico comprendido entre 49152 y 65534.
- Para CPU 31x-2 PN/DP a partir de la versión de firmware V3.2 y CPU 41x-3 PN/DP a partir de la versión de firmware V6.0 y CPU 41x-5 PN/DP a partir de la versión de firmware V6.0:
  - Es posible utilizar un puerto más de una vez.

Los números de puerto reservados para el sistema se indican en la siguiente lista:

Protocolo	Número de puerto	Servicio
-	0	(Este puerto tiene una asignación fija. No puede utilizarse.)
TCP	20 y 21	FTP
TCP	25	SMTP
TCP	80	HTTP
TCP	102	RFC 1006
UDP	135	RPC-DCOM
UDP	161	SNMP REQUEST
TCP, UDP	34962	PNIO
TCP, UDP	34963	PNIO
TCP, UDP	34964	PNIO
UDP	65532	NTP
UDP	65533	NTP
UDP	65534	NTP
UDP	65535	NTP

Para más detalles, consulte también http://www.iana.org/assignments/port-numbers.

Nota

No utilice números de puerto reservados para el sistema.

Parametrización de las conexiones de comunicación en TCP e ISO on TCP

Bloque de datos para la parametrización de conexiones

Para parametrizar las conexiones en TCP e ISO on TCP se debe crear un DB que contenga la estructura de datos de UDT 65 "TCON_PAR". Esta estructura de datos contiene los parámetros necesarios que se requieren para establecer la conexión. Para cada una de las conexiones se necesita una estructura de datos que se pueda agrupar en un DB global.

El parámetro de conexión CONNECT de la instrucción "TCON (Página 7748)" contiene una referencia a la dirección de la correspondiente descripción de conexión (p. ej. P#DB100.DBX0.0 byte 64).

Estructura de la descripción de la conexión (UDT 65)

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
de 0 a 1	block_length	WORD	W#16#40	Longitud del UDT 65: 64 bytes (fija)
de 2 a 3	id	WORD	W#16#0001	Referencia a esta conexión (rango de valores: W#16#0001 a W#16#0FFF) El valor de este parámetro se tiene que indicar en el bloque respectivo en ID.
4	connection_type	BYTE	B#16#11	Variante de protocolo: • B#16#11: TCP • B#16#12: ISO on TCP • B#16#01: TCP (modo de compatibilidad)
5	active_est	BOOL	FALSE	Identificador del tipo de establecimiento de conexión: • FALSE: establecimiento pasivo de la conexión • TRUE: establecimiento activo de la conexión

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
6	local_device_id	BYTE	B#16#02	• B#16#00: comunicación a través del CP 443-1EX (solo en S7-400 y connection_type = B#16#12) CP admisibles: CP443-1EX4x, CP443-1EX20, CP443-1GX20, CP443-1EX30, CP443-1GX30 • B#16#01: comunicación a través de la interfaz IE en slot para interfaz 1 (IF1) con WinAC RTX (sólo TCP) • B#16#02: comunicación a través de la interfaz IE integrada en las CPUs 315-2 PN/DP y 317-2 PN/DP • B#16#03: comunicación a través de la interfaz IE integrada en la CPU 319-3 PN/DP • B#16#05: comunicación a través de la interfaz IE integrada en las CPU 414-3 PN/DP, 416-3 PN/DP, 416-3F PN/DP y 41x-5H PN/DP (rack 0) • B#16#06: comunicación a través de la interfaz IE en slot para interfaz 2 (IF2) con WinAC RTX (sólo TCP) • B#16#0B: comunicación a través de la interfaz IE en slot para interfaz 3 (IF3) con WinAC RTX (sólo TCP) • B#16#0F: comunicación a través de la interfaz IE en slot para interfaz 4 (IF4) con WinAC RTX (sólo TCP) • B#16#15: comunicación a través de la interfaz IE integrada en las CPU 41x-5H PN/DP (rack 1)
7	local_tsap_id_len	BYTE	B#16#02	Longitud utilizada del parámetro local_tsap_id; valores posibles: • 0 ó 2, si connection_type = B#16#01 (lado activo: 0, lado pasivo: 2) • 0 ó 2, si connection_type = B#16#11 (lado activo: 0 ó 2, lado pasivo: 2) • de 2 a 16, si connection_type = B#16#12
8	rem_subnet_id_len	BYTE	B#16#00	Este parámetro no se utiliza actualmente. Se le debe asignar el valor B#16#00.
9	rem_staddr_len	BYTE	B#16#00	Longitud de la dirección del punto final de conexión remoto: • 0: Sin especificar, es decir, el parámetro rem_staddr es irrelevante. • 4: Dirección IP válida en el parámetro rem_staddr
10	rem_tsap_id_len	BYTE	B#16#00	Longitud utilizada del parámetro rem_tsap_id; valores posibles: • 0 ó 2, si connection_type = B#16#01 Para el lado pasivo sólo se admite el valor B#16#00. • 0 ó 2, si connection_type = B#16#11 • 0 ó 2 a 16, si connection_type = B#16#12 (lado activo: 2 a 16; lado pasivo: 0 ó 2 a 16, 0 significa no especificado)
11	next_staddr_len	BYTE	B#16#00	longitud utilizada del parámetro next_staddr (relevante solo con ISO on TCP)

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
de 12 a 27	local_tsap_id	ARRAY [1..16] of BYTE	B#16#07 B#16#D0 B#16#00 ...	con connection_type = • B#16#11: n.º de puerto local (valores posibles véase: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)), local_tsap_id[1] = byte alto del n.º de puerto en representación hexadecimal, local_tsap_id[2] = byte bajo del n.º de puerto en representación hexadecimal, local_tsap_id[3-16] = B#16#00. • B#16#12: ID TSAP local: Véase abajo. • B#16#01: n.º de puerto local (valores posibles véase: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)), local_tsap_id[1] = byte bajo del n.º de puerto en representación hexadecimal, local_tsap_id[2] = byte alto del n.º de puerto en representación hexadecimal, local_tsap_id[3-16] = B#16#00 Nota: Si existen varias conexiones con el mismo interlocutor, hay que asegurarse de que cada uno de los valores utilizados de local_tsap_id sea unívoco en la CPU.
de 28 a 33	rem_subnet_id	ARRAY [1..6] of BYTE	B#16#00 ...	Este parámetro no se utiliza actualmente. Debe asignársele el valor 0.
de 34 a 39	rem_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Dirección IP del punto final remoto de la conexión, p. ej. 192.168.0.1: con connection_type = • B#16#1x: rem_staddr[1] = B#16#C0 (192), rem_staddr[2] = B#16#A8 (168), rem_staddr[3] = B#16#00 (0), rem_staddr[4] = B#16#01 (1), rem_staddr[5-6]= B#16#00 (reservado) • B#16#01: rem_staddr[1] = B#16#01 (1), rem_staddr[2] = B#16#00 (0), rem_staddr[3] = B#16#A8 (168), rem_staddr[4] = B#16#C0 (192), rem_staddr[5-6]= B#16#00 (reservado)

Byte	Parámetro	Tipo de datos	Valor de arranque	Descripción
de 40 a 55	rem_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	con connection_type = • B#16#11: n.º de puerto remoto (valores posibles véase: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)), rem_tsap_id[1] = byte alto del n.º de puerto en representación hexadecimal, rem_tsap_id[2] = byte bajo del n.º de puerto en representación hexadecimal, rem_tsap_id[3-16] = B#16#00 • B#16#12: ID TSAP remota: Véase abajo. • B#16#01: n.º de puerto remoto (valores posibles véase: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)), local_tsap_id[1] = byte bajo del n.º de puerto en representación hexadecimal, local_tsap_id[2] = byte alto del n.º de puerto en representación hexadecimal, local_tsap_id[3-16] = B#16#00
de 56 a 61	next_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	con local_device_id = • B#16#00: next_staddr[1]: rack y slot del correspondiente CP (local) (bits 0 a 4: slot, bits 5 a 7: número de rack) next_staddr[2-6]: B#16#00 • B#16#02, B#16#03, B#16#05: next_staddr[1-6]: B#16#00 • B#16#01, B#16#06, B#16#0B, B#16#0F: next_staddr[1-6]: B#16#00 (en este caso, next_staddr_len=B#16#00)
de 62 a 63	spare	WORD	W#16#0000	Reserva. Asigne a este parámetro el valor "0".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Longitud y estructura del ID TSAP local y remoto en ISO on TCP

- Establecimiento activo de la conexión:
  - ID TSAP remota:
    todas las secuencias de caracteres con una longitud entre 1 y 16 bytes están permitidas. Cada carácter puede tener un valor entre B#16#00 y B#16#FF.
  - ID TSAP local:
    en una interfaz IE integrada están permitidas todas las secuencias de caracteres con una longitud entre 1 y 16 bytes. Cada carácter puede tener un valor entre B#16#00 y B#16#FF.
- Establecimiento pasivo de la conexión:
  - ID TSAP remota:
    todas las secuencias de caracteres con una longitud entre 0 y 16 bytes están permitidas. Cada carácter puede tener un valor entre B#16#00 y B#16#FF.
  - ID TSAP local:

loc_tsap_id_len	local_tsap_id[1]	local_tsap_id[2]	local_tsap_id[3 bis 16]
2	B#16#E0 (tipo de conexión "conexión T")	0 (sólo en interfaz IE integrada) o rack y slot de la propia CPU (slot bits 0 a 4, número de rack bits 5 a 7)	No existe
> 2	B#16#E0 (tipo de conexión "conexión T")	0 (sólo en interfaz IE integrada) o rack y slot de la propia CPU (slot bits 0 a 4, número de rack bits 5 a 7)	Extensión TSAP
	Sólo en interfaz IE integrada: Un carácter ASCII (B#16#20 a B#16#7E)	Irrelevante	Extensión TSAP

Dependencia de la CPU de las variantes de protocolo TCP e ISO on TCP

En el párrafo siguiente se indica cuál de las dos variantes de protocolo TCP e ISP on TOP se puede utilizar en qué CPU:

Relación entre la CPU utilizada, la variante de protocolo (connection_type) y la longitud de datos transferible (Página 7738).

El número de conexiones posibles se indica en los datos técnicos de la CPU.

Establecimiento de la conexión

Un interlocutor A debe iniciar el establecimiento activo de la conexión. Un interlocutor B debe iniciar el establecimiento pasivo de la conexión. Cuando ambos interlocutores han lanzado su establecimiento de conexión, el sistema operativo puede establecer por completo la conexión.

En la parametrización se determina el interlocutor que activa el establecimiento de la conexión y el que responde a la petición del interlocutor estableciendo una conexión pasiva.

En UDP, ambos interlocutores tienen que iniciar el establecimiento pasivo.

Parametrización del punto de acceso local de comunicación en UDP

Estructura de datos para la parametrización del punto de acceso local de comunicación

Para parametrizar el punto de acceso local de comunicación se debe crear un DB que contenga la estructura de datos de UDT 65 "TCON_PAR". Esta estructura de datos contiene los parámetros necesarios para establecer la conexión entre el programa de usuario y el nivel de comunicación del sistema operativo.

El parámetro CONNECT de la instrucción "TCON (Página 7748)" contiene una referencia a la dirección de la correspondiente descripción de la conexión (p. ej. P#DB100.DBX0.0 byte 64).

Estructura de la descripción de la conexión en UDP (UDT 65)

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
de 0 a 1	block_length	WORD	W#16#40	Longitud del UDT 65: 64 bytes (fija)
de 2 a 3	id	WORD	W#16#0001	Referencia a esta conexión entre el programa de usuario y el nivel de comunicación del sistema operativo (rango de valores: W#16#0001 a W#16#0FFF) El valor de este parámetro se tiene que indicar en el bloque respectivo en ID.
4	connection_type	BYTE	B#16#13	Variante de protocolo: • B#16#13: UDP
5	active_est	BOOL	FALSE	Identificador del tipo de establecimiento de conexión: A este parámetro debe asignársele el valor FALSE.
6	local_device_id	BYTE	B#16#02	• B#16#01: comunicación a través de la interfaz IE en slot para interfaz 1 (IF1) con WinAC RTX • B#16#02: comunicación a través de la interfaz IE integrada en la CPU 317-2 PN/DP • B#16#03: comunicación a través de la interfaz IE integrada en la CPU 319-3 PN/DP • B#16#05: comunicación a través de la interfaz IE integrada en las CPU 414-3 PN/DP, 416-3 PN/DP, 416-3F PN/DP y 41x-5H PN/DP (rack 0) • B#16#06: comunicación a través de la interfaz IE en slot para interfaz 2 (IF2) con WinAC RTX • B#16#0B: comunicación a través de la interfaz IE en slot para interfaz 3 (IF3) con WinAC RTX • B#16#0F: comunicación a través de la interfaz IE en slot para interfaz 4 (IF4) con WinAC RTX • B#16#15: comunicación a través de la interfaz IE integrada en las CPU 41x-5H PN/DP (rack 1)
7	local_tsap_id_len	BYTE	B#16#02	Longitud utilizada del parámetro local local_tsap_id (puerto local); 2 bytes

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
8	rem_subnet_id_len	BYTE	B#16#00	Este parámetro no se utiliza. Se le debe asignar el valor B#16#00.
9	rem_staddr_len	BYTE	B#16#00	Este parámetro no se utiliza. Se le debe asignar el valor B#16#00.
10	rem_tsap_id_len	BYTE	B#16#00	Este parámetro no se utiliza. Se le debe asignar el valor B#16#00.
11	next_staddr_len	BYTE	B#16#00	Este parámetro no se utiliza. Se le debe asignar el valor B#16#00.
de 12 a 27	local_tsap_id	ARRAY [1..16] of BYTE	B#16#07 B#16#D0 B#16#00 ...	N.º de puerto local (valores posibles véase: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)), local_tsap_id[1] = byte alto del n.º de puerto en representación hexadecimal, local_tsap_id[2] = byte bajo del n.º de puerto en representación hexadecimal, local_tsap_id[3-16] = B#16#00 (reservado) Nota: Asegúrese de que todos los valores de local_tsap_id que utiliza en la CPU sean unívocos.
de 28 a 33	rem_subnet_id	ARRAY [1..6] of BYTE	B#16#00 ...	Este parámetro no se utiliza. Debe asignársele el valor "0".
de 34 a 39	rem_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Este parámetro no se utiliza. Debe asignársele el valor "0".
de 40 a 55	rem_tsap_id	ARRAY [1..16] of BYTE	B#16#00 ...	Este parámetro no se utiliza. Debe asignársele el valor "0".
de 56 a 61	next_staddr	ARRAY [1..6] of BYTE	B#16#00 ...	Este parámetro no se utiliza. Debe asignársele el valor "0".
de 62 a 63	spare	WORD	W#16#0000	Reserva. Asigne a este parámetro el valor "0".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Dependencia de la CPU de la variante de protocolo UDP

En el párrafo siguiente se indica en qué CPU se puede utilizar la variante de protocolo UDP:

Relación entre la CPU utilizada, la variante de protocolo (connection_type) y la longitud de datos transferible (Página 7738)

El número de conexiones posibles entre el programa de usuario y el nivel de comunicación del sistema operativo se indica en los datos técnicos de la CPU.

Configuración del punto de acceso local de comunicación

Todos los interlocutores tienen que configurar su punto de acceso local de comunicación independientemente del otro, es decir, establecer la conexión entre el programa de usuario y el nivel de comunicación del sistema operativo.

En UDP, ambos interlocutores tienen que iniciar el establecimiento pasivo.

Estructura de la información de dirección del interlocutor remoto en UDP

Vista general

- En "TUSEND (Página 7760)", en el parámetro ADDR se transfiere la información de dirección del receptor. La información de dirección tiene que tener la estructura indicada a continuación.
- En "TURCV (Página 7756)" se recibe la dirección del remitente de los datos recibidos en el parámetro ADDR. La información de dirección tiene que tener la estructura indicada a continuación.

Bloque de datos para la información de dirección del interlocutor remoto

Se tiene que crear un DB que contenga una o más estructuras de datos según UDT 66 "TADDR_PAR".

En el parámetro ADDR de "TUSEND (Página 7760)" se transfiere un puntero hacia la dirección del interlocutor remoto correspondiente (p. ej.P#DB100.DBX0.0 byte 8) y en el parámetro ADDR de TURCV (Página 7756) se recibe dicho puntero.

Estructura de la información de dirección del interlocutor remoto (UDT 66)

Byte	Parámetro	Tipo de datos	Valor inicial	Descripción
de 0 a 3	rem_ip_addr	ARRAY [1..4] of BYTE	B#16#00 ...	Dirección IP del interlocutor remoto, p. ej. 192.168.2.3: • rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#02 (2) • rem_ip_addr[4] = B#16#03 (3)
4 a 5	rem_port_nr	ARRAY [1..2] of BYTE	B#16#00 ...	N.º de puerto remoto (valores posibles véase: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)) • rem_port_nr[1] = byte alto del n.º de puerto en representación hexadecimal • rem_port_nr[2] = byte bajo del n.º de puerto en representación hexadecimal
6 a 7	spare	ARRAY [1..2] of BYTE	B#16#00 ...	Reserva. Asigne a este parámetro el valor "0".

Relación entre la CPU utilizada, la variante de protocolo (connection_type) y la longitud de datos transferible

Relación entre la CPU utilizada, la variante de protocolo (connection_type) y la longitud de datos transferible

En la siguiente tabla se describe, qué variante se utiliza en qué CPU y cuáles son las longitudes de datos que se pueden transferir

Variante de protocolo	Parámetro "connection _type" en UDT 65	CPU	Longitud de datos en bytes en la comunicación vía interfaz integrada IE	Longitud de datos en bytes en la comunicación vía CP
TCP	B#16#11	CPUs 31x-2 PN/DP a partir de la versión de firmware V2.4	de 1 a 8192	-
		CPUs 31x-2 PN/DP a partir de la versión de firmware V3.1	de 1 a 32767	
		CPUs 31x-3 PN/DP a partir de la versión de firmware V3.2	de 1 a 32767	
		CPU 41x-3 PN/DP	de 1 a 32767	-
		CPU 41x-5H PN/DP	de 1 a 32767	-
	B#16#01	CPUs 31x-2 PN/DP	de 1 a 1472	-
ISO on TCP	B#16#12	CPUs 31x-2 PN/DP a partir de la versión de firmware V2.4	de 1 a 8192	-
		CPUs 31x-2 PN/DP a partir de la versión de firmware V3.1	de 1 a 32767	
		CPUs 31x-3 PN/DP a partir de la versión de firmware V3.2	de 1 a 32767	
		CPUs 41x a partir de la versión de firmware V4.1 (sin CPUs 41x-4H)	-	de 1 a 1452
		CPU 41x-3 PN/DP	de 1 a 32767	de 1 a 1452
		CPU 41x-5H PN/DP	de 1 a 32767	de 1 a 1452
UDP	B#16#13	CPU 31x-2 PN/DP a partir de la versión de firmware V2.4, CPU 41x-3 PN/DP y 41x-5H PN/DP	de 1 a 1472	-

Ejemplos de parametrización de las conexiones de comunicación

Ejemplo 1: Comunicación vía ISO on TCP y CP 443-1 entre dos CPUs S7-400

Los dos interlocutores son dos CPUs 414-2. La comunicación se lleva a cabo a través de dos CPs 443-1.

La siguiente tabla muestra los datos más importantes de los dos interlocutores:

Propiedad	Interlocutor A: CPU 414-2 con CP 443-1	Interlocutor B: CPU 414-2 con CP 443-1
Establecimiento de la conexión	Activo	Pasivo
Dirección IP	192.168.4.14	192.168.4.16
Dirección física de la CPU	Rack 0, slot 3	Rack 0, slot 4
Dirección física del CP correspondiente	Rack 0, slot 6	Rack 1, slot 8
ID TSAP local (nota: Para diferenciar la conexión, la codificación del TSAP real se lleva a cabo a partir del tercer byte).	0xE0 03 54 43 50 2D 31	0xE0 04 54 43 50 2D 31

La siguiente tabla muestra las entradas de parámetros en el DB relevantes para que el interlocutor A establezca la conexión de forma activa:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
id	WORD	W#16#0414	Referencia a esta conexión
connection_type	BYTE	B#16#12	Variante de protocolo: ISO on TCP
active_est	BOOL	TRUE	Establecimiento activo de la conexión
local_device_id	BYTE	B#16#00	Comunicación interna del AS vía CP
local_tsap_id_len	BYTE	B#16#07	Longitud utilizada del parámetro local_tsap_id
rem_staddr_len	BYTE	B#16#04	Longitud de la dirección del punto final de conexión remoto: • 4: dirección IP válida en el parámetro rem_staddr
rem_tsap_id_len	BYTE	B#16#07	Longitud utilizada del parámetro rem_tsap_id
next_staddr_len	BYTE	B#16#01	Longitud utilizada del parámetro next_staddr
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#54 (equivalente ASCII de "T") • local_tsap_id[4] = B#16#43 (equivalente ASCII de "C") • local_tsap_id[5] = B#16#50(equivalente ASCII de "P") • local_tsap_id[6] = B#16#2D(equivalente ASCII de "-") • local_tsap_id[7] = B#16#31(equivalente ASCII de "1") • local_tsap_id[8-16] = B#16#00	ID TSAP local: 0xE0035443502D31

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
rem_staddr	ARRAY [1..6] of BYTE	"192.168.4.16" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#04 (4) • rem_staddr[4] = B#16#10 (16) • rem_staddr[5-6] = B#16#00	Dirección IP del punto final de conexión remoto
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#04 • rem_tsap_id[3] = B#16#54 (equivalente ASCII de "T") • rem_tsap_id[4] = B#16#43 (equivalente ASCII de "C") • rem_tsap_id[5] = B#16#50 (equivalente ASCII de "P") • rem_tsap_id[6] = B#16#2D (equivalente ASCII de "-") • rem_tsap_id[7] = B#16#31 (equivalente ASCII de "1") • rem_tsap_id[8-16] = B#16#00	ID TSAP remota: 0xE0045443502D31
next_staddr	ARRAY [1..6] of BYTE	• next_staddr[1] = B#16#06 • next_staddr[2-6] = B#16#00	Rack = 0, slot = 6 (bits 7 a 5: N.° de rack, bits de 4 a 0: N.º de slot)

La siguiente tabla muestra las entradas de parámetros en el DB relevantes para que el interlocutor B establezca la conexión de forma pasiva:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
id	WORD	W#16#0416	Referencia a esta conexión
connection_type	BYTE	B#16#12	Variante de protocolo: ISO on TCP
active_est	BOOL	FALSE	Establecimiento pasivo de la conexión
local_device_id	BYTE	B#16#00	Comunicación interna del AS vía CP
local_tsap_id_len	BYTE	B#16#07	Longitud utilizada del parámetro local_tsap_id
rem_staddr_len	BYTE	B#16#04	Longitud de la dirección del punto final de conexión remoto: • 4: dirección IP válida en el parámetro rem_staddr
rem_tsap_id_len	BYTE	B#16#07	Longitud utilizada del parámetro rem_tsap_id
next_staddr_len	BYTE	B#16#01	Longitud utilizada del parámetro next_staddr

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#04 • local_tsap_id[3] = B#16#54 (equivalente ASCII de "T") • local_tsap_id[4] = B#16#43 (equivalente ASCII de "C") • local_tsap_id[5] = B#16#50 (equivalente ASCII de "P") • local_tsap_id[6] = B#16#2D(equivalente ASCII de "-") • local_tsap_id[7] = B#16#31 (equivalente ASCII de "1") • local_tsap_id[8-16] = B#16#00	ID TSAP local: 0xE0045443502D31
rem_staddr	ARRAY [1..6] of BYTE	"192.168.4.14" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#04 (4) • rem_staddr[4] = B#16#0E (14) • rem_staddr[5-6] = B#16#00	Dirección IP del punto final de conexión remoto
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#54 (equivalente ASCII de "T") • rem_tsap_id[4] = B#16#43 (equivalente ASCII de "C") • rem_tsap_id[5] = B#16#50 (equivalente ASCII de "P") • rem_tsap_id[6] = B#16#2D (equivalente ASCII de "-") • rem_tsap_id[7] = B#16#31 (equivalente ASCII de "1") • rem_tsap_id[8-16] = B#16#00	ID TSAP remota: 0xE0035443502D31
next_staddr	ARRAY [1..6] of BYTE	• next_staddr[1] = B#16#28 • next_staddr[2-6] = B#16#00	Rack = 1, slot = 8 (bits 7 a 5: N.° de rack, bits de 4 a 0: N.º de slot)

Ejemplo 2: Comunicación vía TCP e interfaces PROFINET integradas entre una CPU S7-400 y una CPU S7-300

Un interlocutor es una CPU 416-3 PN/DP y el otro es una CPU 319-3 PN/DP. La siguiente tabla muestra los datos más importantes de los dos interlocutores:

Propiedad	Interlocutor A: CPU 416-3 PN/DP	Interlocutor B: CPU 319-3 PN/DP
Establecimiento de la conexión	Activo	Pasivo
Dirección IP	192.168.3.142	192.168.3.125
N.º de puerto local	Irrelevante	2005

La siguiente tabla muestra las entradas de parámetros en el DB relevantes para que el interlocutor A establezca la conexión de forma activa:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
id	WORD	W#16#0014	Referencia a esta conexión
connection_type	BYTE	B#16#11	Variante de protocolo: TCP
active_est	BOOL	TRUE	Establecimiento activo de la conexión
local_device_id	BYTE	B#16#05	Comunicación vía interfaz Ethernet integrada en CPUs S7-400
local_tsap_id_len	BYTE	B#16#00 (sólo es posible este valor)	El parámetro local_tsap_id no se utiliza
rem_staddr_len	BYTE	B#16#04	Longitud de la dirección del punto final de conexión remoto: • 4: Dirección IP válida en el parámetro rem_staddr
rem_tsap_id_len	BYTE	B#16#02 (sólo es posible este valor)	Longitud utilizada del parámetro rem_tsap_id
rem_staddr	ARRAY [1..6] of BYTE	"192.168.3.125" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#03 (3) • rem_staddr[4] = B#16#7D (125) • rem_staddr[5-6] = B#16#00	Dirección IP del punto final de conexión remoto
rem_tsap_id	ARRAY [1..16] of BYTE	"2005" • rem_tsap_id[1] = B#16#07 • rem_tsap_id[2] = B#16#D5 • rem_tsap_id[3-16] = B#16#00	N.º de puerto remoto: 2005 = W#16#07D5

La siguiente tabla muestra las entradas de parámetros en el DB relevantes para que el interlocutor B establezca la conexión de forma pasiva:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
id	WORD	W#16#000F	Referencia a esta conexión
connection_type	BYTE	B#16#11	Variante de protocolo: TCP
active_est	BOOL	FALSE	Establecimiento pasivo de la conexión
local_device_id	BYTE	B#16#03	Comunicación a través de la interfaz Ethernet integrada en la CPU 319-3 PN/DP
local_tsap_id_len	BYTE	B#16#02 (sólo es posible este valor)	Longitud utilizada del parámetro local_tsap_id
rem_staddr_len	BYTE	B#16#04	Longitud de la dirección del punto final de conexión remoto: • 4: dirección IP válida en el parámetro rem_staddr
rem_tsap_id_len	BYTE	B#16#00 (sólo es posible este valor)	Longitud utilizada del parámetro rem_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	"2005" • local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D5 • local_tsap_id[3-16] = B#16#00	N.º de puerto local: 2005 = W#16#07D5
rem_staddr	ARRAY [1..6] of BYTE	"192.168.3.142" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#03 (3) • rem_staddr[4] = B#16#8E (142) • rem_staddr[5-6] = B#16#00	Dirección IP del punto final de conexión remoto

Ejemplo 3: Comunicación vía UDP e interfaces PROFINET integradas entre una CPU S7-300 y una CPU S7-400

Un interlocutor es una CPU 319-3 PN/DP y el otro es una CPU 414-3 PN/DP. La siguiente tabla muestra los datos más importantes de los dos interlocutores:

Propiedad	Interlocutor A: CPU 319-3 PN/DP	Interlocutor B: CPU 414-3 PN/DP
Emisor/receptor	Emisor	Receptor
Dirección IP	192.168.3.142	192.168.3.125
N.º de puerto local	2004	2005

La siguiente tabla muestra las entrada en el DB relevantes para el emisor (interlocutor A) a la hora de parametrizar el punto de acceso de comunicación local:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
id	WORD	W#16#0014	Referencia a esta conexión entre el programa de usuario y el nivel de comunicación del sistema operativo
connection_type	BYTE	B#16#13	Variante de protocolo: UDP
active_est	BOOL	FALSE	Sólo este valor es posible en la variante de protocolo UDP.
local_device_id	BYTE	B#16#03	Comunicación a través de la interfaz Ethernet integrada en la CPU 319-3 PN/DP
local_tsap_id_len	BYTE	B#16#02	Longitud utilizada del parámetro local_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D4 • local_tsap_id[3-16] = B#16#00	N.º de puerto local: 2004 = W#16#07D4

La siguiente tabla muestra las entrada en el DB relevantes para el receptor (interlocutor B) a la hora de parametrizar el punto de acceso de comunicación local:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
id	WORD	W#16#000F	Referencia a esta conexión entre el programa de usuario y el nivel de comunicación del sistema operativo
connection_type	BYTE	B#16#13	Variante de protocolo: UDP
active_est	BOOL	FALSE	Sólo este valor es posible en la variante de protocolo UDP.
local_device_id	BYTE	B#16#05	Comunicación vía interfaz Ethernet integrada en CPUs S7-400
local_tsap_id_len	BYTE	B#16#02	Longitud utilizada del parámetro local_tsap_id
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#07 • local_tsap_id[2] = B#16#D5 • local_tsap_id[3-16] = B#16#00	N.º de puerto local: 2005 = W#16#07D5

Al llamar "TUSEND (Página 7760)" en el emisor, se transfieren en un DB los siguientes parámetros de dirección del receptor:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
rem_ip_addr	ARRAY [1..4] of BYTE	• rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#3 (3) • rem_ip_addr[4] = B#16#7D (125)	Dirección IP del receptor: 192.168.3.125
rem_port_nr	ARRAY [1..2] of BYTE	• rem_port_nr[1] = B#16#07 • rem_port_nr[2] = B#16#D5	N.° de puerto del receptor: 2005 = W#16#07D5

Al llamar "TURCV (Página 7763)" en el receptor, se reciben en un DB los siguientes parámetros de dirección del emisor:

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
rem_ip_addr	ARRAY [1..4] of BYTE	• rem_ip_addr[1] = B#16#C0 (192) • rem_ip_addr[2] = B#16#A8 (168) • rem_ip_addr[3] = B#16#3 (3) • rem_ip_addr[4] = B#16#8E (142)	Dirección IP del emisor: 192.168.3.142
rem_port_nr	ARRAY [1..2] of BYTE	• rem_port_nr[1] = B#16#07 • rem_port_nr[2] = B#16#D4	N.° de puerto del emisor: 2004 = W#16#07D4

Ejemplo 4: Comunicación vía ISO on TCP e interfaces PROFINET integradas entre dos CPUs S7-400

Un interlocutor es una CPU 414-3 PN/DP a partir de la versión de firmware V5.0 y el otro es una CPU 416-3 PN/DP con versión de firmware V5.0. La comunicación se lleva a cabo a través de las interfaces PROFINET integradas.

La siguiente tabla muestra los datos más importantes de los dos interlocutores.

Propiedad	Interlocutor A: CPU 414-3 PN/DP	Interlocutor B: CPU 416-3 PN/DP
Emisor/receptor	Emisor	Receptor
Establecimiento de la conexión	Pasivo	Activo
Dirección IP	192.168.0.14	192.168.0.16

La siguiente tabla muestra todas las entradas de parámetros del interlocutor A en el DB correspondiente.

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
block_length	WORD	W#16#40	Longitud del UDT 65: 64 bytes (fija)
id	WORD	W#16#0001	Referencia a esta conexión

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
connection_type	BYTE	B#16#12	Variante de protocolo: • B#16#12: ISO on TCP
active_est	BOOL	FALSE	Establecimiento pasivo de la conexión
local_device_id	BYTE	B#16#05	Comunicación a través de la interfaz IE integrada en las CPUs 41x PN/DP
local_tsap_id_len	BYTE	B#16#03	Longitud utilizada del parámetro local_tsap_id
rem_subnet_id_len	BYTE	B#16#00	Este parámetro no se utiliza. Se le debe asignar el valor B#16#00.
rem_staddr_len	BYTE	B#16#04	Longitud de la dirección del punto final de conexión remoto: • 4: dirección IP válida en el parámetro rem_staddr
rem_tsap_id_len	BYTE	B#16#03	Longitud utilizada del parámetro rem_tsap_id
next_staddr_len	BYTE	B#16#00	Este parámetro no se utiliza (debido a la comunicación vía interfaz integrada). Se le debe asignar el valor B#16#00 .
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#01 • local_tsap_id[4-16] = B#16#00	ID TSAP local: 0xE00301
rem_subnet_id	ARRAY [1..6] of BYTE	rem_subnet_id[1-6] = B#16#00	Este parámetro no se utiliza. Debe asignársele el valor "0".
rem_staddr	ARRAY [1..6] of BYTE	"192.168.0.16" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#00 (0) • rem_staddr[4] = B#16#10 (16) • rem_staddr[5-6] = B#16#00	Dirección IP del punto final de conexión remoto
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#01 • rem_tsap_id[4-16] = B#16#00	ID TSAP remota: 0xE00301
next_staddr	ARRAY [1..6] of BYTE	next_staddr[1-6] = B#16#00	Este parámetro no se utiliza (debido a la comunicación vía interfaz integrada).
spare	WORD	W#16#0000	Este parámetro no se utiliza. Debe asignársele el valor "0".

La siguiente tabla muestra todas las entradas de parámetros del interlocutor B en el DB correspondiente.

Parámetro	Tipo de datos	Valor en el ejemplo	Descripción
block_length	WORD	W#16#40	Longitud del UDT 65: 64 bytes (fija)
id	WORD	W#16#0001	Referencia a esta conexión
connection_type	BYTE	B#16#12	Variante de protocolo: • B#16#12: ISO on TCP
active_est	BOOL	TRUE	Establecimiento activo de la conexión
local_device_id	BYTE	B#16#05	Comunicación a través de la interfaz IE integrada en las CPUs 41x PN/DP
local_tsap_id_len	BYTE	B#16#03	Longitud utilizada del parámetro local_tsap_id
rem_subnet_id_len	BYTE	B#16#00	Este parámetro no se utiliza. Se le debe asignar el valor B#16#00.
rem_staddr_len	BYTE	B#16#04	Longitud de la dirección del punto final de conexión remoto: • 4: dirección IP válida en el parámetro rem_staddr
rem_tsap_id_len	BYTE	B#16#03	Longitud utilizada del parámetro rem_tsap_id
next_staddr_len	BYTE	B#16#00	Este parámetro no se utiliza (debido a la comunicación vía interfaz integrada). Se le debe asignar el valor B#16#00.
local_tsap_id	ARRAY [1..16] of BYTE	• local_tsap_id[1] = B#16#E0 • local_tsap_id[2] = B#16#03 • local_tsap_id[3] = B#16#01 • local_tsap_id[4-16] = B#16#00	ID TSAP local: 0xE00301
rem_subnet_id	ARRAY [1..6] of BYTE	rem_subnet_id[1-6] = B#16#00	Este parámetro no se utiliza. Debe asignársele el valor "0".
rem_staddr	ARRAY [1..6] of BYTE	"192.168.0.14" • rem_staddr[1] = B#16#C0 (192) • rem_staddr[2] = B#16#A8 (168) • rem_staddr[3] = B#16#00 (0) • rem_staddr[4] = B#16#0E (14) • rem_staddr[5-6] = B#16#00	Dirección IP del punto final de conexión remoto
rem_tsap_id	ARRAY [1..16] of BYTE	• rem_tsap_id[1] = B#16#E0 • rem_tsap_id[2] = B#16#03 • rem_tsap_id[3] = B#16#01 • rem_tsap_id[4-16] = B#16#00	ID TSAP remota: 0xE00301
next_staddr	ARRAY [1..6] of BYTE	next_staddr[1-6] = B#16#00	Este parámetro no se utiliza (debido a la comunicación vía interfaz integrada).
spare	WORD	W#16#0000	Este parámetro no se utiliza. Debe asignársele el valor "0".

Tipos de datos

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

TCON: Establecer una conexión

Número de conexiones posibles

El número de conexiones TCP o UDP posibles se indica en los datos técnicos de la CPU.

Conexión vía TCP e ISO on TCP

Ambos interlocutores llaman la instrucción "TCON" para crear y establecer la conexión. En la parametrización se especifica cuál es el punto final de la comunicación activo y cuál el pasivo.

Una vez creada y establecida la conexión, la CPU la mantiene y la vigila automáticamente.

Si se interrumpe la conexión p. ej. debido a una interrupción de la línea o por el interlocutor remoto, el interlocutor activo intenta restablecerla. No es preciso volver a llamar "TCON".

Con la llamada de la instrucción "TDISCON" o en el estado operativo STOP de la CPU se interrumpe una conexión existente y se elimina la conexión creada. Para volver a crear y establecer la conexión es preciso ejecutar "TCON" de nuevo.

Uso con UDP

Ambos interlocutores llaman la instrucción "TCON" para establecer su punto de acceso local (puerto local). Se configura una conexión entre el programa de usuario y la capa de comunicación del sistema operativo. No se establece comunicación con el interlocutor remoto.

El punto de acceso local se utiliza para transmitir y recibir telegramas UDP.

En la parametrización del punto de acceso local (UDT65) o del interlocutor remoto (UDT 66) deben coincidir los siguientes parámetros.

- Con UDP: Referencia a la conexión y n.º de puerto local.

Funcionamiento

La instrucción "TCON" trabaja de forma asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. Para iniciar la petición para la creación y el establecimiento de la conexión, llame "TCON" con REQ = 1.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Le ayudará a determinar en qué estado se encuentra actualmente "TCON" o cuándo se ha creado o establecido la conexión (con TCP e ISO on TCP).

BUSY	DONE	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna petición (nueva) a la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TCON":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C	Parámetro de control REQUEST, inicia la petición para el establecimiento de la conexión indicada en ID. La petición se inicia con un flanco ascendente.
ID	Input	WORD	M, D o constante	Referencia a la conexión que debe establecerse con el interlocutor remoto o bien entre el programa de usuario y la capa de comunicación del sistema operativo. ID debe ser idéntica al respectivo parámetro ID en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. • BUSY = 0: la petición ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: se ha producido un error durante la ejecución de la petición. STATUS devuelve información detallada sobre el tipo de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	M, D	Parámetro de estado STATUS: Información de error
CONNECT	InOut	ANY	D	Puntero hacia la descripción de conexión correspondiente (UDT 65) Consulte también: • Parametrización de las conexiones de comunicación en TCP e ISO on TCP (Página 7730) • Parametrización del punto de acceso local de comunicación en UDP (Página 7735) Nota: El parámetro CONNECT se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: CONN_DB.Conn1

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Explicación
0	0000	Se ha podido establecer la conexión
0	7000	No se está procesando ninguna petición
0	7001	Inicio de la ejecución de la petición, estableciendo conexión
0	7002	Llamada intermedia (REQ irrelevante), estableciendo conexión
1	8086	El parámetro ID se encuentra fuera del rango admisible.
1	8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones.
1	8089	El parámetro CONNECT no apunta a un bloque de datos.
1	809A	El parámetro CONNECT apunta a un campo que no tiene la longitud de la descripción de la conexión (UDT65).
1	809B	El local_device_id indicado en la descripción de la conexión no concuerda con la CPU.
1	80A0	Error de grupo para los códigos de error W#16#80A1 y W#16#80A2
1	80A1	La conexión o el puerto ya están ocupados por el usuario
1	80A2	El puerto local o remoto está ocupado por el sistema
1	80A3	Se está intentando volver a establecer una conexión existente.
1	80A4	La dirección IP del punto final de la conexión no es válida o coincide con la dirección IP propia
1	80A7	Error de comunicación: Tras un "TCON" no finalizado se ha llamado un "TDISCON (Página 7751)". El establecimiento de la conexión ha sido interrumpido anticipadamente por la llamada de un "TDISCON (Página 7751)".
1	80B2	El parámetro CONNECT apunta a un bloque de datos generado con la palabra clave UNLINKED.
1	80B3	Parametrización incoherente: Error de grupo para los códigos de error W#16#80A0 a W#16#80A2, W#16#80A4, W#16#80B4 a W#16#80B9

ERROR	STATUS (W#16#...)	Explicación
1	80B4	Con la variante de protocolo ISO on TCP (connection_type = B#16#12), en el establecimiento pasivo de la conexión (active_est = FALSE) ha vulnerado una o varias de las siguientes condiciones: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0 • Con local_tsap_id_len >= B#16#03, local_tsap_id[1] es un carácter ASCII • local_tsap_id[1] es un carácter ASCII y local_tsap_id_len >= B#16#03
1	80B5	Error en el parámetro active_est (UDT 65) en la variante de protocolo UDP
1	80B6	Error de parametrización respecto al parámetro connection_type (UDT 65)
1	80B7	Error en uno de los parámetros siguientes de UDT 65: block_length, local_tsap_id_len, rem_subnet_id_len, rem_staddr_len, rem_tsap_id_len, next_staddr_len
1	80B8	Los parámetros de la descripción de la conexión local (UDT 65) y el parámetro ID son diferentes
1	80C3	Todos los recursos de conexión están ocupados.
1	80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz se está reparametrizando • La conexión creada está siendo eliminada por un "TDISCON (Página 7751)". • El sistema H se encuentra en Acoplar y Sincronizar.
1	8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Consulte también

Parametrización de las conexiones de comunicación en TCP e ISO on TCP (Página 7730)

Parametrización del punto de acceso local de comunicación en UDP (Página 7735)

TDISCON: Desconectar comunicación

Conexión vía TCP e ISO on TCP

La instrucción "TDISCON" deshace la conexión entre la CPU y un interlocutor.

Uso con UDP

La instrucción "TDISCON" deshace el punto de acceso local, es decir, que se deshace la conexión entre el programa de usuario y la capa de comunicación del sistema operativo.

Funcionamiento

La instrucción "TDISCON" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. Para iniciar la petición para el establecimiento de la conexión, llame la instrucción "TDISCON" con REQ = 1.

Tras la ejecución correcta de la instrucción "TDISCON", el ID indicado en la instrucción "TCON (Página 7748)" ya no es válido, por lo que no puede utilizarse para transmitir ni para recibir.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas (véase también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)).

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "TDISCON":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C	Inicia la petición de deshacer la conexión especificada en ID cuando se detecta un flanco ascendente.
ID	Input	WORD	M, D o constante	Referencia a la conexión establecida con "TCON (Página 7748)". Rango de valores: W#16#0001 a W#16#0FFF
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o en ejecución • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	Parámetro de estado con los valores siguientes: • 0: Petición no iniciada aún o ya finalizada • 1: Petición no finalizada aún. No es posible iniciar una nueva petición.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • 0: Ningún error • 1: Con errores
STATUS	Output	WORD	M, D	Estado de la instrucción

Parámetros BUSY, DONE y ERROR

El estado de ejecución se controla a través de los parámetros BUSY, DONE, ERROR y STATUS. El parámetro BUSY indica el estado de ejecución. El parámetro DONE permite comprobar si una petición se ha ejecutado correctamente. El parámetro ERRORse activa cuando ocurren errores durante la ejecución de TDISCON. La información de error se indica en el parámetro STATUS.

La tabla siguiente muestra la relación entre los parámetros BUSY, DONE y ERROR:

BUSY	DONE	ERROR	Descripción
1	-	-	La petición se está procesando.
0	1	0	La petición se ha ejecutado correctamente.

BUSY	DONE	ERROR	Descripción
0	0	1	La petición se ha finalizado con un error. La causa del error se indica en el parámetro STATUS.
0	0	0	No se ha asignado ninguna petición nueva.

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Explicación
0	0000	La conexión se ha deshecho correctamente
0	7000	No se está procesando ninguna petición
0	7001	Inicio del procesamiento de la petición, deshaciendo la conexión.
0	7002	Deshaciendo la conexión (REQ es irrelevante).
1	8086	El parámetro ID se encuentra fuera del rango de direcciones admisible.
1	80A3	Se está intentando finalizar una conexión no existente.
1	80C4	Error de comunicación temporal: La interfaz está recibiendo nuevos parámetros o bien la conexión se está estableciendo o el sistema H se encuentra en Acoplar y Sincronizar.

Consulte también

Vista general de los tipos de datos válidos (Página 2321)

TSEND: Enviar datos a través de la conexión

Descripción

La instrucción "TSEND" envía datos a través de una conexión existente.

Funcionamiento

"TSEND" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición de transmisión, llame la instrucción "TSEND" con REQ = 1.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

El parámetro de salida DONE se activa si los datos se han transferido completamente a la interfaz local para su envío.

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Con su ayuda se puede determinar en qué estado se encuentra actualmente "TSEND".

BUSY	DONE	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Nota

Debido al funcionamiento asíncrono de "TSEND", los datos del área de transmisión deben mantenerse coherentes hasta que el parámetro DONE o el parámetro ERROR adopten el valor TRUE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C	El parámetro de control REQUEST activa la petición de transmisión con un flanco ascendente. Los datos se transmiten desde el área indicada con DATA y LEN.
ID	Input	WORD	M, D o constante	Referencia a la conexión correspondiente. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	INT	I, Q, M, D, L	Número máximo de bytes que deben enviarse con la petición Véase también: Relación entre la CPU utilizada, la variante de protocolo (connection_type) y la longitud de datos transferible (Página 7738)
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Los datos se han transferido completamente a la interfaz local para su envío.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. No es posible iniciar otra petición. • BUSY = 0: la petición ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	M, D	Parámetro de estado STATUS: Información de error
DATA	InOut	ANY	I, Q, M, D	Área de transmisión, contiene la dirección y la longitud La dirección remite a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos Nota: No utilice ARRAY of BOOL como área de transmisión. Nota: El parámetro DATA se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Explicación
0	0000	La petición de transmisión ha finalizado sin errores
0	7000	No se está procesando ninguna petición
0	7001	Inicio del procesamiento de la petición, enviando los datos. Nota: En esta fase de la ejecución, el sistema operativo accede a los datos del área de transmisión DATA.
0	7002	Llamada intermedia (REQ irrelevante), la petición se está ejecutando Nota: En esta fase de la ejecución, el sistema operativo accede a los datos del área de transmisión DATA.
1	8085	El parámetro LEN tiene el valor "0" o excede el valor máximo admisible
1	8086	El parámetro ID se encuentra en un rango de valores no válido
1	8088	El parámetro LEN es mayor que el área de memoria indicada en DATA
1	80A1	Error de comunicación: • La conexión indicada no se ha establecido todavía. • La conexión indicada se está estableciendo en este momento. No es posible transmitir a través de esta conexión. • La interfaz se está reinicializando.
1	80B3	El tipo de conexión parametrizado (parámetro connection_type en la descripción de la conexión) es UDP. Utilice "TUSEND (Página 7760)".
1	80C3	• Ya se está ejecutando un bloque con esta ID en otra clase de prioridad. • Falta de recursos internos

ERROR	STATUS (W#16#...)	Explicación
1	80C4	Error de comunicación temporal: • La conexión con el interlocutor no se puede establecer en estos momentos. • La interfaz se está reparametrizando o estableciendo.
1	8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

TRCV: Recibir datos a través de la conexión

Descripción

La instrucción "TRCV" recibe datos a través de una conexión existente.

Área de recepción

Se entiende como tal el área en la que el bloque de función registra los datos recibidos.

El área de recepción está definida por los dos valores siguientes:

- Puntero hacia el inicio del área
- Longitud del área

En función de la variante de protocolo utilizada, la longitud del área se fija con el parámetro LEN (si LEN <> 0) o por la indicación de longitud del parámetro DATA (si LEN = 0).

Modos de recepción de TRCV

En la tabla siguiente se describe cómo "TRCV" registra los datos recibidos en el área de recepción.

Variante de protocolo	Introducir los datos en el área de recepción	Parámetro connection_type	Valor del parámetro LEN
TCP	Modo Ad hoc	B#16#01, B#16#11	0
TCP	Recepción de datos con la longitud especificada	B#16#01, B#16#11	<> 0
ISO on TCP	Controlado por protocolo	B#16#12	0 (recomendado) o <> 0

TCP/modo Ad hoc

El modo Ad hoc sólo está disponible con el protocolo TCP. El modo Ad hoc se establece asignando el valor 0 al parámetro LEN.

El área de recepción es idéntica al área especificada en el parámetro DATA. Como máximo se reciben 8192 bytes. Inmediatamente después de la recepción de un bloque de datos, la instrucción "TRCV" los transmite al área de recepción y pone NDR a "1".

TCP / Recepción de datos con la longitud especificada

Asignando al parámetro LEN un valor distinto a cero se establece la recepción de datos con la longitud dada.

El área de recepción está formada por los parámetros LEN y DATA. En cuanto se han recibido LEN bytes, TRCV los transmite al área de recepción y pone NDR a "1".

ISO on TCP / transferencia de datos controlada por protocolo

En el caso del protocolo ISO on TCP, los datos se transfieren de forma controlada por el protocolo.

El área de recepción está formada por los parámetros LEN y DATA. En cuanto se han recibido todos los datos de una petición, "TRCV" los transmite al área de recepción y pone NDR a "1".

Intercambio de datos de tipo STRING

Para intercambiar datos de tipo STRING con las instrucciones "TSEND" y "TRCV", se recomienda utilizar únicamente el modo Ad hoc con LEN=0. Además, en la instrucción "TRCV" hay que seleccionar una longitud del string al menos igual de larga que en la instrucción "TSEND". Si no se respeta esta regla, pueden producirse errores durante la posterior ejecución del string.

Cuando se utiliza ARRAY of BYTE como área de recepción para una cadena de caracteres enviada, el primer byte contiene la longitud máxima definida por el emisor y el segundo byte, la longitud actual de la cadena.

Funcionamiento

"TRCV" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar la petición de recepción, llame la instrucción TRCV con EN_R = 1.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

En la tabla siguiente se muestra la correspondencia entre BUSY, NDR y ERROR. Permite determinar en qué estado se encuentra "TRCV" actualmente o cuándo finalizó la recepción.

BUSY	NDR	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Nota

Debido al funcionamiento asíncrono de "TRCV", los datos no son coherentes en el área de recepción hasta que el parámetro NDR no adopta el valor TRUE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TRCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L, T, C	Parámetro de control enabled to receive: Con EN_R = 1 "TRCV" pasa a disponibilidad de recepción. La petición de recepción se está ejecutando.
ID	Input	WORD	M, D o constante	Referencia a la conexión correspondiente. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	INT	I, Q, M, D, L	Longitud del área de recepción en bytes Para conocer el significado de LEN = 0 ó LEN <> 0, ver arriba (modos de recepción de "TRCV"). Respecto al rango de valores, véase también: Relación entre la CPU utilizada, la variante de protocolo (connection_type) y la longitud de datos transferible (Página 7738).
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • NDR = 0: la petición todavía no se ha iniciado o está en curso • NDR = 1: la petición ha finalizado correctamente
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. No es posible iniciar otra petición. • BUSY = 0: la petición ha finalizado.
STATUS	Output	WORD	M, D	Parámetro de estado STATUS: Información de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RCVD_LEN	Output	INT	I, Q, M, D, L	Número de datos recibidos realmente en bytes.
DATA	InOut	ANY	I, Q, M, D	Área de recepción (para definición, véase arriba), contiene la dirección y la longitud La dirección remite a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos Nota: No utilice ARRAY of BOOL como área de recepción. Nota: El parámetro DATA se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Explicación
0	0000	Se han aplicado nuevos datos. La longitud actual de los datos recibidos se indica en RCVD_LEN.
0	7000	El bloque no está preparado para la recepción
0	7001	El bloque está preparado para la recepción, la petición de recepción se ha activado
0	7002	Llamada intermedia, la petición de recepción está en proceso Nota: En esta fase de ejecución, la instrucción escribe datos en el área de recepción. Por ello, en caso de error es posible que haya datos incoherentes en el área de recepción.
1	8085	El parámetro LEN es mayor que el valor máximo permitido, o ha modificado uno de los parámetros LEN o DATA respecto a la primera llamada
1	8086	El parámetro ID se encuentra en un rango de valores no válido
1	8088	• El área de recepción es demasiado pequeña • El valor de LEN es mayor que el área de recepción especificada por DATA
1	80A1	Error de comunicación: • La conexión indicada no se ha establecido todavía. • La conexión indicada se está estableciendo en este momento. No es posible ejecutar una petición de recepción a través de esta conexión. • La interfaz se está reparametrizando.
1	80B3	La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) es UDP. Utilice "TURCV (Página 7763)".
1	80C3	• Ya se está ejecutando un bloque con esta ID en otra clase de prioridad. • Falta de recursos internos

ERROR	STATUS (W#16#...)	Explicación
1	80C4	Error de comunicación temporal: • La conexión con el interlocutor no se puede establecer en estos momentos. • La interfaz se está reparametrizando o estableciendo.
1	8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

TUSEND: Enviar datos vía Ethernet (UDP)

Descripción

La instrucción "TUSEND" envía datos a través de UDP al interlocutor remoto direccionado con el parámetro ADDR.

ADVERTENCIA

Transmisión de datos a través de UDP

En la transmisión de datos a través de UDP según RFC 768, los datos se transmiten al interlocutor remoto sin acuse y, por lo tanto, sin seguridad. Ello significa que es posible que se pierdan datos, sin que ello se indique en el bloque.

Nota

En procesos de transmisión consecutivos a diferentes interlocutores, en las llamadas de "TUSEND" sólo es necesario adaptar el parámetro ADDR. Sin embargo, se prescinde de la nueva llamada de las instrucciones "TCON (Página 7748)" y "TDISCON (Página 7751)".

Funcionamiento

"TUSEND" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. Para iniciar la petición de transmisión, llame la instrucción "TUSEND" con REQ = 1.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra "TUSEND" actualmente o cuándo finalizó la transmisión.

BUSY	DONE	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.

BUSY	DONE	ERROR	Descripción
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Nota

Debido al funcionamiento asíncrono de "TUSEND", los datos del área de transmisión deben mantenerse coherentes hasta que el parámetro DONE o el parámetro ERROR adopten el valor TRUE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TUSEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C	El parámetro de control REQUEST activa la petición de transmisión con un flanco ascendente. Los datos se transmiten desde el área indicada con DATA y LEN.
ID	Input	WORD	M, D o constante	Referencia a la conexión correspondiente entre el programa de usuario y la capa de comunicación del sistema operativo. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	INT	I, Q, M, D, L	Número de bytes que deben enviarse con la petición Rango de valores: de 1 a 1472
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. No es posible iniciar otra petición. • BUSY = 0: la petición ha finalizado.
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error
STATUS	Output	WORD	M, D	Parámetro de estado STATUS: Información de error

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DATA	InOut	ANY	I, Q, M, D	Área de transmisión, contiene la dirección y la longitud La dirección remite a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos Nota: El parámetro DATA se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable
ADDR	InOut	ANY	D	Puntero hacia la dirección del receptor (p. ej. P#DB100.DBX0.0 byte 8) Consulte también: Estructura de la información de dirección del interlocutor remoto en UDP (Página 7737) Nota: El parámetro ADDR se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Explicación
0	0000	La petición de transmisión ha finalizado sin errores
0	7000	No se está procesando ninguna petición
0	7001	Inicio del procesamiento de la petición, enviando los datos. Nota: En esta fase de la ejecución, el sistema operativo accede a los datos del área de transmisión DATA.
0	7002	Llamada intermedia (REQ irrelevante), la petición se está ejecutando Nota: En esta fase de la ejecución, el sistema operativo accede a los datos del área de transmisión DATA.
1	8085	El parámetro LEN tiene el valor "0" o excede el valor máximo admisible
1	8086	El parámetro ID se encuentra en un rango de valores no válido
0	8088	El parámetro LEN es mayor que el área de memoria indicada en DATA
1	8089	El parámetro ADDR no apunta a un bloque de datos

ERROR	STATUS (W#16#...)	Explicación
1	80A1	Error de comunicación: • Todavía no se ha establecido la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. • Se está estableciendo la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. No es posible transmitir a través de esta conexión. • La interfaz se está reinicializando.
1	80A4	La dirección IP del punto final de la conexión no es válida o coincide con la dirección IP propia.
1	80B3	• La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) no es UDP. Utilice "TSEND (Página 7753)". • Parámetro ADDR: Indicación no válida para n.º de puerto.
1	80C3	• Ya se está ejecutando un bloque con esta ID en otra clase de prioridad. • Falta de recursos internos
1	80C4	Error de comunicación temporal: • En este momento no es posible establecer la conexión entre el programa de usuario y la capa de comunicación del sistema operativo. • La interfaz se está reparametrizando.
1	8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

TURCV: Recibir datos vía Ethernet (UDP)

Descripción

La instrucción "TURCV" recibe datos a través de UDP. Tras la finalización correcta de "TURCV" en el parámetro ADDR se pone a disposición la dirección del interlocutor remoto, es decir, del transmisor.

⚠ ADVERTENCIA

Transmisión de datos a través de UDP según RFC 768

En la transmisión de datos a través de UDP según RFC 768, los datos se transmiten al interlocutor remoto sin acuse y, por lo tanto, sin seguridad. Ello significa que es posible que se pierdan datos, sin que ello se indique en el bloque.

Funcionamiento

"TURCV" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas. Para iniciar la petición de recepción, llame la instrucción "TURCV" con EN_R = 1.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

En la tabla siguiente se muestra la correspondencia entre BUSY, NDR y ERROR. Permite determinar en qué estado se encuentra TURCV actualmente o cuándo finalizó la recepción.

BUSY	NDR	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Nota

Debido al funcionamiento asíncrono de "TURCV", los datos no son coherentes en el área de recepción hasta que el parámetro NDR no adopta el valor TRUE.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "TURCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_R	Input	BOOL	I, Q, M, D, L, T, C	Parámetro de control enabled to receive: Con EN_R = 1 "TURCV" pasa a disponibilidad de recepción. La petición de recepción se está ejecutando.
ID	Input	WORD	M, D o constante	Referencia a la conexión correspondiente entre el programa de usuario y la capa de comunicación del sistema operativo. ID debe coincidir con el parámetro ID correspondiente en la descripción local de la conexión. Rango de valores: W#16#0001 a W#16#0FFF
LEN	Input	INT	I, Q, M, D, L	Longitud del área de recepción en bytes: 0 (recomendado) o bien 1 a 1472 Consulte también: Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro de estado NDR: • NDR = 0: la petición todavía no se ha iniciado o está en curso • NDR = 1: la petición ha finalizado correctamente
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: • ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la petición todavía no ha finalizado. No es posible iniciar otra petición. • BUSY = 0: la petición ha finalizado.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
STATUS	Output	WORD	M, D	Parámetro de estado STATUS: Información de error
RCVD_LEN	Output	INT	I, Q, M, D, L	Número de datos recibidos realmente en bytes.
DATA	InOut	ANY	I, Q, M, D	Área de recepción (véase la definición en Funcionamiento de las instrucciones para Open User Communication vía Industrial Ethernet (Página 7726)) La dirección remite a: • La memoria imagen de proceso de las entradas • La memoria imagen de proceso de las salidas • Una marca • Un bloque de datos Nota: El parámetro DATA se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable
ADDR	InOut	ANY	D	Puntero hacia la dirección del emisor (p. ej. P#DB100.DBX0.0 byte 8), véase también: Estructura de la información de dirección del interlocutor remoto en UDP (Página 7737) Nota: El parámetro ADDR se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

ERROR	STATUS (W#16#...)	Explicación
0	0000	Se han aplicado nuevos datos. La longitud actual de los datos recibidos se indica en RCVD_LEN.
0	7000	El bloque no está preparado para la recepción
0	7001	El bloque está preparado para la recepción, la petición de recepción se ha activado
0	7002	Llamada intermedia, la petición de recepción está en proceso Nota: En esta fase de ejecución, "TURCV" escribe datos en el área de recepción. Por ello, en caso de error es posible que haya datos incoherentes en el área de recepción.
1	8085	El parámetro LEN es mayor que el valor máximo permitido, o ha modificado uno de los parámetros LEN o DATA respecto a la primera llamada
1	8086	El parámetro ID se encuentra en un rango de valores no válido
1	8088	• El área de recepción es demasiado pequeña • El valor de LEN es mayor que el área de recepción especificada por DATA

ERROR	STATUS (W#16#...)	Explicación
1	8089	El parámetro ADDR no apunta a un bloque de datos
1	80A1	Error de comunicación: • Todavía no se ha establecido la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. • Se está estableciendo la conexión indicada entre el programa de usuario y la capa de comunicación del sistema operativo. No es posible ejecutar una petición de recepción a través de esta conexión. • La interfaz se está reparametrizando.
1	80B3	La variante de protocolo parametrizada (parámetro connection_type en la descripción de la conexión) no es UDP. Utilice "TRCV (Página 7756)".
1	80C3	• Ya se está ejecutando un bloque con esta ID en otra clase de prioridad. • Falta de recursos internos
1	80C4	Error de comunicación temporal: La interfaz se está reparametrizando.
1	8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

IP_CONF: Modificar parámetros de configuración IP

Descripción de IP_CONF

Descripción

La instrucción "IP_CONF" sirve para realizar la configuración controlada por el programa de la interfaz PROFINET integrada en la CPU. Los datos de configuración hasta ahora válidos se sobrescriben.

Se pueden realizar los siguientes ajustes de configuración de la interfaz:

- Parámetro IP: dirección IP, máscara de subred, dirección de router
- Nombre de dispositivo PROFINET IO (en caso de que la CPU se utilice como dispositivo PROFINET IO)

Los datos de configuración deben almacenarse en un DB de configuración.

Los ajustes de la configuración IP controlados por el programa con la instrucción "IP_CONF" también se pueden configurar alternativamente con STEP 7. No obstante, sólo son efectivos si se preselecciona explícitamente en la configuración de hardware que la asignación de parámetros IP se lleva a cabo a través del programa de usuario.

Nota

Como alternativa a la configuración, los datos de configuración de un procesador de comunicaciones también pueden establecerse con la instrucción "IP_CONFIG" (véase SIMATIC NET CP).

Funcionamiento

La instrucción "IP_CONF" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. Para iniciar el proceso de transferencia, llame la instrucción "IP_CONF" con REQ = 1.

Sólo puede haber una petición activa cada vez.

El estado de la petición se indica a través de los parámetros de salida BUSY y STATUS. STATUS corresponde al parámetro de salida RET_VAL de las instrucciones asíncronas (véase también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595)).

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra la instrucción actualmente o cuándo se completó la transferencia de los datos de configuración.

BUSY	DONE	ERROR	Descripción
TRUE	Irrelevante	Irrelevante	La petición está en proceso.
FALSE	TRUE	FALSE	La petición ha finalizado correctamente.
FALSE	FALSE	TRUE	La petición se ha finalizado con un error. La causa del error se puede consultar en el parámetro STATUS.
FALSE	FALSE	FALSE	No se ha asignado ninguna (nueva) petición a la instrucción.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "IP_CONF":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L, T, C	El parámetro de control REQUEST. activa la instrucción con un flanco ascendente.
LADDR	Input	WORD	M, D o constante	Dirección de diagnóstico de la interfaz PROFINET
CONF_DB (Página 7769)	Input	ANY	D	Puntero hacia los datos de configuración (tipos de datos admisibles: BYTE, WORD, BLOCK_DB) Nota: El parámetro CONF_DB se puede indicar de las dos formas siguientes: • absoluta. Ej.: P#DB13.DBX0.0 byte 64 • simbólica. Ej.: DB_name.variable
DONE	Output	BOOL	I, Q, M, D, L	Parámetro de estado DONE: • 0: La petición aún no se ha iniciado o todavía se está ejecutando. • 1: Petición finalizada sin errores
BUSY	Output	BOOL	I, Q, M, D, L	• 0: La petición ha finalizado. • 1: La petición todavía no ha finalizado. No es posible iniciar otra petición.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	Parámetro de estado ERROR: ERROR=1: se ha producido un error durante la ejecución. STATUS devuelve información detallada sobre el tipo de error
STATUS	Output	DWORD	I, Q, M, D, L	Información de error
ERR_LOC	Output	DWORD	I, Q, M, D, L	Origen del error (field_id y subfield_type_id del bloque de parámetros que ha causado el error)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros ERROR y STATUS

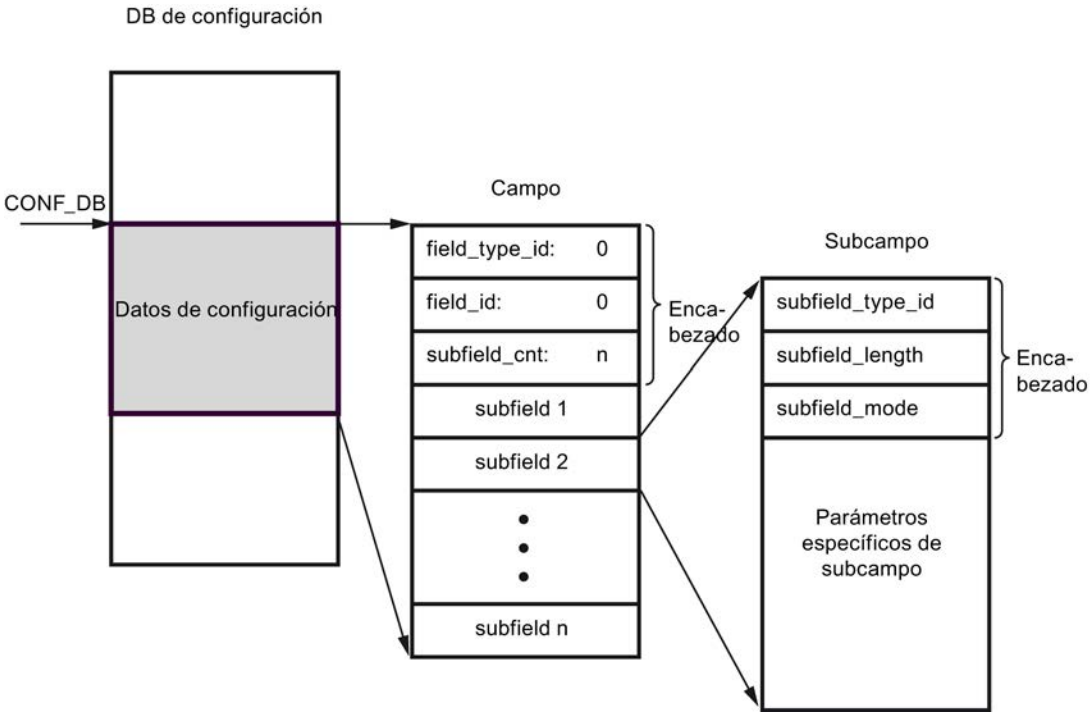
ERROR	STATUS (DW#16#..)	ERR_LOC*	Explicación
0	00000000	0	Procesamiento de la petición finalizado sin errores
0	00700000	0	No se está procesando ninguna petición
0	00700100	0	Inicio del procesamiento de la petición
0	00700200	0	Llamada intermedia (REQ es irrelevante)
1	C08xyy00	0	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)
1	C0808000	0	LADDR no es válido
1	C0808100	0	LADDR no está asignado a la interfaz PROFINET soportada
1	C0808200	0	Error en el parámetro CONF_DB: No se soporta el tipo de datos.
1	C0808300	0	Error en el parámetro CONF_DB: El puntero señala a un área no soportada.
1	C0808400	0	Error en el parámetro CONF_DB: Longitud incorrecta del puntero ANY
1	C0808700	0	Error en el parámetro CONF_DB: Incoherencia de la indicación de longitud en los datos de configuración
1	C0808800	s, 0	field_type_id tiene un valor no admisible
1	C0808900	s, 0	field_id tiene un valor no admisible
1	C0808A00	s, 0	Número incorrecto en subfield_cnt
1	C0808B00	s, t	subfield_id tiene un valor no admisible
1	C0808C00	s, t	Subcampo utilizado más de una vez
1	C0808D00	s, t	subfield_len tiene un valor incorrecto o no admisible
1	C0808E00	s, t	subfield_mode tiene un valor no admisible
1	C0808F00	s, t	En el subcampo hay un conflicto con un subcampo anterior.
1	C0809000	s, t	Los parámetros del subcampo están protegidos contra escritura. P. ej. se realizó el preajuste de parámetros por configuración o el modo PNIO está activo.
1	C0809400	s, t	El valor del parámetro en el subcampo no está definido o no es admisible
1	C080C200	0	No se puede realizar la transferencia (p. ej., porque la interfaz no es accesible).
1	C080C300	0	Recursos insuficientes (p. ej., llamada múltiple de "IP_CONF" con parámetros diferentes)

ERROR	STATUS (DW#16#..)	ERR_LOC*	Explicación
1	C080C400	0	Error de comunicación temporal
* En la tabla anterior f es la field_id y s la subfield_type_id del bloque de parámetros que ha causado el error.			

Bloque de datos CONF_DB

Descripción

En la siguiente imagen se muestra cómo se almacenan en el DB de configuración los datos de configuración que deben transferirse.



Los datos de configuración se componen exactamente de un campo y varios subcampos:

- El campo se compone de un encabezado y de los subcampos. A su vez, el encabezado se compone de los elementos siguientes:
  - field_type_id (tipo de datos INT): Cero
  - field_id (tipo de datos INT): Cero
  - subfield_cnt (tipo de datos INT): Número de subcampos
- Cada subcampo se compone de un encabezado (subfield_type_id, subfield_length, subfield_mode) y de los parámetros específicos del subcampo. Cada subcampo debe componerse de un número entero de bytes.
  subfield_mode admite los siguientes valores:
  - 1: validez permanente de los datos de configuración
  - 2: validez temporal de los datos de configuración, incluido el borrado de los datos de configuración permanentes existentes

Nota

Actualmente se permite un campo. Sus parámetros field_type_id y field_id deben tener el valor cero. Los campos con otros valores para field_type_id y field_id quedan reservados para futuras ampliaciones.

Subcampos admisibles

subfield_type_id	Nombre del subcampo	Explicación
30	SUB_IP_SUITE_IPV4	Parámetro IP: dirección IP, máscara de subred, dirección de router
40	SUB_NOS	Nombre de dispositivo PROFINET IO (Name of Station)

Definiciones de tipo para el subcampo "SUB_IP_SUITE_IPV4"

Nombre		Tipo de datos	Valor de arranque	Descripción
SUB_IP_SUITE_V4		Struct		
	Id	INT	30	subfield_type_id
	len	INT	18	subfield_length
	mode	INT	1	subfield_mode (1: permanente, 2: temporal)
	ipaddr_3	BYTE	b#16#C8	Byte High de la dirección IP: 200
	ipaddr_2	BYTE	b#16#0C	Byte High de la dirección IP: 12
	ipaddr_1	BYTE	b#16#01	Byte Low de la dirección IP: 1
	ipaddr_0	BYTE	b#16#90	Byte Low de la dirección IP: 144
	snmask_3	BYTE	b#16#FF	Byte High de la máscara de subred: 255
	snmask_2	BYTE	b#16#FF	Byte High de la máscara de subred: 255
	snmask_1	BYTE	b#16#FF	Byte Low de la máscara de subred: 255
	snmask_0	BYTE	b#16#00	Byte Low de la máscara de subred: 0
	router_3	BYTE	b#16#C8	Byte High del router: 200

Nombre		Tipo de datos	Valor de arranque	Descripción
	router_2	BYTE	b#16#0C	Byte High del router: 12
	router_1	BYTE	b#16#01	Byte Low del router: 1
	router_0	BYTE	b#16#01	Byte Low del router: 1

Definiciones de tipo para el subcampo "SUB_NOS"

Nombre		Tipo de datos	Valor de arranque	Descripción
SUB_NOS		Struct		
	id	INT	40	subfield_type_id
	len	INT	246	subfield_length
	mode	INT	1	subfield_mode (1: permanente, 2: temporal)
	nos	ARRAY[1..240] of BYTE	0	Nombre de dispositivo: ARRAY debe ocuparse desde el primer byte. Si ARRAY es más largo que el nombre de dispositivo que se va a asignar, se debe introducir un byte cero después del nombre de dispositivo (de acuerdo con IEC 61158-6-10). De lo contrario, nos se rechaza y la instrucción "IP_CONF (Página 7766)" registra el código de error DW#16#C0809400 en STATUS. Si se ocupa el primer byte con un cero, se borra el nombre de dispositivo.

Los nombres de dispositivo están sujetos a las siguientes restricciones:

- Limitación a 240 caracteres en total (letras minúsculas, cifras, guión o punto)
- Los distintos componentes del nombre de dispositivo, es decir, las cadenas de caracteres entre dos puntos, no deben superar los 63 caracteres de longitud, respectivamente. Los componentes del nombre no pueden empezar ni terminar con el carácter "-".
- No se admiten caracteres especiales como vocales acentuadas o con diéresis, los paréntesis, caracteres de subrayado, barras, espacios en blanco, etc. El único carácter especial permitido es el guión.
- El nombre de dispositivo no puede empezar ni acabar por los caracteres "-" o ".".
- El nombre de dispositivo no puede empezar con una cifra.
- El nombre de dispositivo no puede tener el formato n.n.n.n (n = 0, ... 999).

- El nombre de dispositivo no puede empezar con la cadena de caracteres "port-xyz" o "port-xyz-abcde" (a, b, c, d, e, x, y, z = 0... 9).
- No se admite la cadena de caracteres ".." dentro del nombre de dispositivo.

Nota

El ARRAY "nos" también se puede crear con menos de 240 bytes, siendo 2 bytes la longitud mínima. En este caso se debe adaptar la variable "len" (longitud del subcampo) en correspondencia.

Intercambio de datos a través de FETCH y WRITE

FW_TCP: Intercambiar datos mediante FETCH y WRITE vía TCP

Descripción

Con la instrucción "FW_TCP" se ofrecen los servicios FETCH/WRITE a través de la conexión TCP.

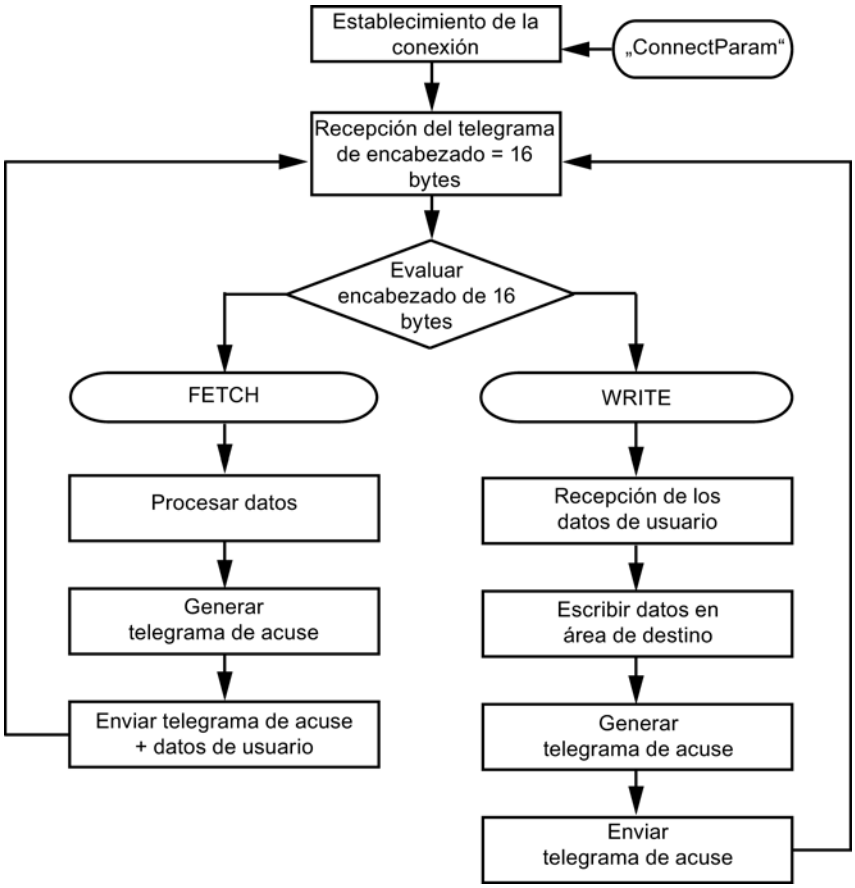
Llamada

Dado que para los servicios FETCH/WRITE se requiere una conexión TCP separada, respectivamente, la instrucción se llama dos veces en el OB1:

- La primera llamada con el bloque de datos de instancia FETCH_TCP_DB ofrece el servicio FETCH a través de las conexiones TCP.
- La segunda llamada con el bloque de datos de instancia WRITE_TCP_DB ofrece la función WRITE a través de la conexión TCP.

La conexión TCP se establece automáticamente tras un arranque en frío, un arranque en caliente y un rearranque de la CPU, así como después de modificar los parámetros de conexión. Los parámetros para el establecimiento de la conexión se encuentran en el DB "ConnectParam" (descripción: véase más abajo). A través del parámetro de entrada ENABLE existe la posibilidad de controlar manualmente la conexión y desconexión de la comunicación. Si se modifican los parámetros de la conexión, ésta se desconecta y restablece automáticamente.

En la siguiente figura se representa esquemáticamente el desarrollo de la instrucción "FW_TCP", que se llama en el OB1 y que ejecuta los servicios FETCH y WRITE a través de conexiones TCP.



Parámetros de conexión del bloque de datos de instancia "ConnectParam"

Los siguientes parámetros de conexión se comprueban antes de establecer la conexión:

- connection_type
- id
- local_tsap_id

Si uno de ellos no está definido correctamente, la conexión no se establece. No se transmiten ni reciben datos.

- Una vez que la conexión se ha establecido, se recibe un telegrama de encabezado de 16 bytes de los servicios FETCH/WRITE. El telegrama de encabezado contiene la información que indica si se trata de una petición FETCH o una petición WRITE.
- Si se trata de una petición FETCH, los datos solicitados por el interlocutor se procesan en el bloque de datos de instancia DB210 de la instrucción y se genera el telegrama de acuse. El telegrama de acuse se envía junto con los datos procesados.
- Si se trata de una petición WRITE, tras la evaluación del telegrama de encabezado de 16 bytes se reciben los datos de usuario, que se escriben en el área de destino. A continuación se genera y transmite el telegrama de acuse en el bloque de datos de instancia DB310.

La tabla siguiente muestra la estructura de los parámetros de una conexión TCP. Ésta tiene una longitud de 64 bytes.

Parámetro	Tipo de datos	Valor de arranque	Descripción
id	WORD	W#16#1	ID de la conexión: El valor del parámetro debe encontrarse dentro del siguiente rango de valores: id = W#16#0001 a W#16#0FFF (1 a 4095)
connection_type	BYTE	B#16#11	Tipo de conexión. Para la conexión TCP rige lo siguiente: • connection_type = B#16#11 (TCP native) o • connection_type = B#16#1 (modo de compatibilidad TCP native)
active_est	BOOL	FALSE	Establecimiento pasivo de la conexión.
local_device_id	BYTE	B#16#3	Identificación de la interfaz local PN/IE (aquí: CPU 319-3PN/DP).
local_tsap_id_len	BYTE	B#16#2	Longitud utilizada del parámetro local_tsap_id en bytes: • 0 ó 2, con el tipo de conexión = 17 (TCP) • Para el lado activo sólo se admite el valor "0".
rem_subnet_id_len	BYTE	B#16#0	Este parámetro no se utiliza.
rem_staddr_len	BYTE	B#16#0	Longitud de la dirección del punto final del interlocutor (parámetro rem_staddr) en bytes. Dado que se trata de una conexión no especificada, el parámetro es irrelevante.
rem_tsap_id_len	BYTE	B#16#0	Longitud utilizada del parámetro rem_tsap_id en bytes: • 0 ó 2, con el tipo de conexión = 17 (TCP). • Para el lado pasivo sólo se admite el valor 0.
next_staddr_len	BYTE	B#16#0	Este parámetro no se utiliza.
local_tsap_id	ARRAY [1..16] of BYTE	B#16#7 B#16#D0 B#16#0 : : B#16#0	Puerto local para la conexión TCP. Valores posibles: 1...49151 (valores recomendados: 2000...5000). • local_tsap_id[1] = byte alto del n.º de puerto en representación hexadecimal; • local_tsap_id[2] = byte bajo del n.º de puerto en representación hexadecimal; • local_tsap_id[3-16] = irrelevante Nota: Asegúrese de que todos los valores de local_tsap_id sean unívocos dentro de la CPU.
rem_subnet_id	ARRAY [1..6] of BYTE	B#16#0 …	Este parámetro no se utiliza.
rem_staddr	ARRAY [1..6] of BYTE	B#16#0	Dirección IP del punto final del interlocutor, p. ej. para 192.168.2.3: • rem_staddr[1] = 192 • rem_staddr[2] = 168 • rem_staddr[3] = 2 • rem_staddr[4] = 3 • rem_staddr[5-6] = irrelevante Irrelevante para conexiones no especificadas con establecimiento pasivo de la conexión.

Parámetro	Tipo de datos	Valor de arranque	Descripción
rem_tsap_id	ARRAY [1..16] of BYTE	B#16#0	Puerto del interlocutor. Valores posibles con 17 (TCP): 1...49151 (valores recomendados: 2000...5000). • rem_tsap_id[1] = byte alto del n.º de puerto en representación hexadecimal; • rem_tsap_id[2] = byte bajo del n.º de puerto en representación hexadecimal; • rem_tsap_id[3-16] = irrelevante Irrelevante para conexiones no especificadas con establecimiento pasivo de la conexión.
next_staddr	ARRAY [1..6] of BYTE	B#16#0	Este parámetro no se utiliza.
spare	WORD	W#16#0	Este parámetro no se utiliza.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "FW_TCP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ENABLE	Input	BOOL	I, Q, M, D, L, T, C	Con este parámetro se inicia el establecimiento o la interrupción de la conexión: • 0 = la conexión se deshace • 1 = la conexión se establece
CONNECT	Input	ANY	D	Puntero ANY hacia el área de datos del DB211 que contiene los parámetros para el establecimiento de la conexión TCP. Esta área de datos debe tener un tamaño mínimo de 64 bytes.
ADDRMODE	Input	INT	I, Q, M, D, L o constante	Con este parámetro se especifica el modo en que se produce el direccionamiento de los datos de la petición FETCH o WRITE (en el modo de direccionamiento S7 o S5). • 0 = modo de direccionamiento S7 (La dirección inicial de los datos se interpreta como dirección de byte) • 1 = modo de direccionamiento S5 (La dirección inicial de los datos se interpreta como dirección de palabra)
NDR	Output	BOOL	I, Q, M, D, L	Parámetro new data record: Con este parámetro se indica que los datos de la petición WRITE se han aplicado correctamente y que el telegrama de acuse se ha generado y enviado.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ERROR	Output	BOOL	I, Q, M, D, L	Este parámetro se activa en los siguientes casos: • Si se ha producido un error en el establecimiento o la interrupción de la conexión • Si se ha producido un error al enviar o recibir los datos. • Si se ha recibido un telegrama de encabezado no válido de FETCH/WRITE.
MODE	Output	BYTE	I, Q, M, D, L	Con este parámetro se indica si se está ejecutando una petición FETCH o WRITE. • 0 = ninguna petición activa • 1 = Petición WRITE • 2 = Petición FETCH
STATUS (Página 7780)	Output	WORD	M, D	En el parámetro STATUS se indican bits de comunicación internos.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

FW_IOT: Intercambiar datos mediante FETCH y WRITE vía ISO-on-TCP

Descripción

Con la instrucción "FW_IOT" se ofrecen los servicios FETCH/WRITE a través de la conexión ISO on TCP.

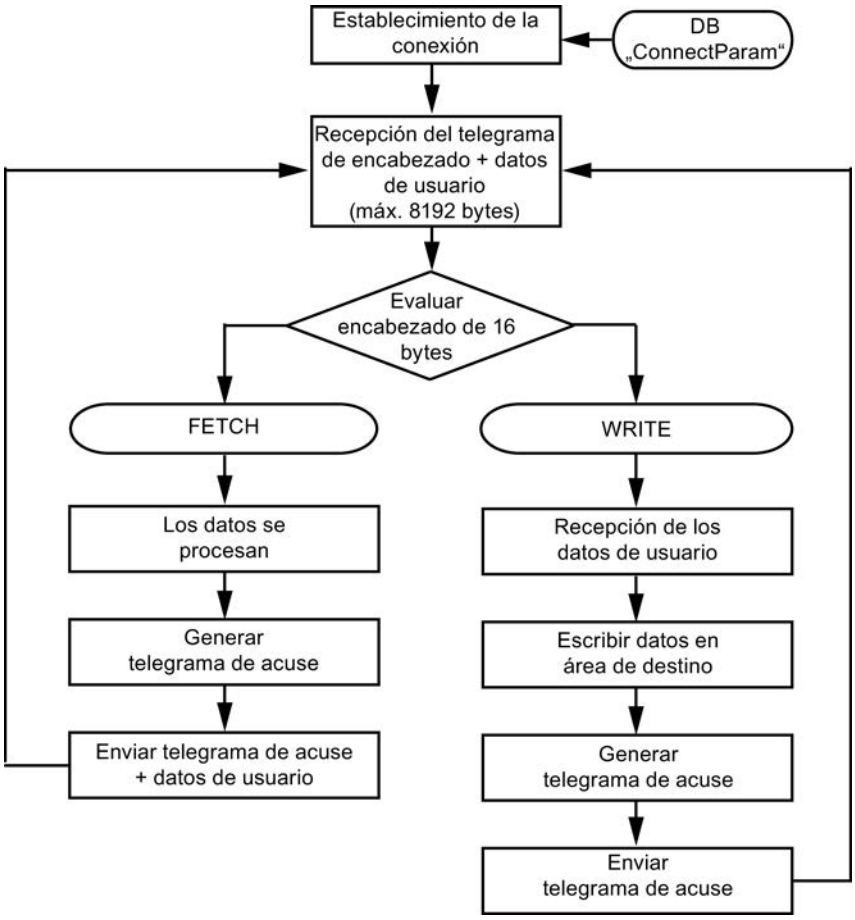
Llamada

Dado que para los servicios FETCH/WRITE se requiere una conexión ISO on TCP separada, respectivamente, la instrucción "FW_IOT" se llama dos veces en el OB1.

- La primera llamada con el bloque de datos de instancia FETCH_IOT_DB ofrece el servicio FETCH a través de las conexiones ISO on TCP.
- La segunda llamada con el bloque de datos de instancia WRITE_IOT_DB ofrece el servicio WRITE a través de la conexión ISO on TCP.

La conexión ISO on TCP se establece automáticamente tras un arranque en frío, un arranque en caliente y un rearranque de la CPU, así como después de modificar los parámetros de conexión. Los parámetros para el establecimiento de la conexión se encuentran en el DB "ConnectParam". El parámetro de entrada ENABLE permite al usuario controlar manualmente el establecimiento y la interrupción de la conexión. Si se modifican los parámetros de la conexión, ésta se desconecta y restablece automáticamente. La tabla siguiente muestra la estructura de datos de los parámetros de una conexión ISO on TCP. Ésta tiene una longitud de 64 bytes.

En la siguiente figura se representa esquemáticamente el desarrollo de la instrucción "FW_IOT", que se llama en el OB1 y que ejecuta los servicios FETCH y WRITE a través de conexiones ISO on TCP.



Parámetros de conexión del bloque de datos de instancia "ConnectParam"

Los siguientes parámetros de conexión se comprueban antes de establecer la conexión:

- connection_type
- id
- local_tsap_id
- local_tsap_id _len

Si uno de ellos no está definido correctamente, la conexión no se establece. No se transmiten ni reciben datos.

- Una vez que la conexión se ha establecido, se recibe un telegrama de encabezado de 16 bytes de los servicios FETCH/WRITE. El telegrama de encabezado contiene la información que indica si se trata de una petición FETCH o una petición WRITE.
- Si se trata de una petición FETCH, los datos solicitados por el interlocutor se procesan en el bloque de datos de instancia FETCH_IOT_DB de la instrucción y se genera el telegrama de acuse. El telegrama de acuse se envía junto con los datos procesados.
- Si se trata de una petición WRITE, tras la evaluación del telegrama de encabezado de 16 bytes se reciben los datos de usuario, que se escriben en el área de destino. A continuación se genera y transmite el telegrama de acuse en el bloque de datos de instancia WRITE_IOT_DB.

La tabla siguiente muestra la estructura de los parámetros de una conexión TCP. Ésta tiene una longitud de 64 bytes.

Parámetro	Tipo de datos	Valor de arranque	Descripción
id	WORD	W#16#1	ID de la conexión: El valor del parámetro debe encontrarse dentro del siguiente rango de valores: id = W#16#0001 a W#16#0FFF (1 a 4095)
connection_type	BYTE	B#16#12	Tipo de conexión. Para la conexión ISO on TCP rige lo siguiente: connection_type = B#16#12
active_est	BOOL	FALSE	Establecimiento pasivo de la conexión.
local_device_id	BYTE	B#16#3	Identificación de la interfaz local PN/IE (aquí: CPU 319-3PN/DP).
local_tsap_id_len	BYTE	B#16#2	Longitud utilizada del parámetro local_tsap_id en bytes: • De 2 a 16, si el tipo de conexión = 18 (ISO on TCP)
rem_subnet_id_len	BYTE	B#16#0	Este parámetro no se utiliza.
rem_staddr_len	BYTE	B#16#0	Longitud de la dirección del punto final del interlocutor (parámetro rem_staddr) en bytes. Dado que se trata de una conexión no especificada, el parámetro es irrelevante.
rem_tsap_id_len	BYTE	B#16#0	Longitud utilizada del parámetro rem_tsap_id en bytes: • De 2 a 16, si el tipo de conexión = 18 (ISO on TCP)
next_staddr_len	BYTE	B#16#0	Este parámetro no se utiliza.
local_tsap_id	ARRAY [1..16] of BYTE	B#16#7 B#16#D0 B#16#0 : : B#16#0	ID TSAP local de la conexión ISO on TCP. • local_tsap_id[1] = B#16#E0; • local_tsap_id[2] = rack y slot del punto final local (bits 0 a 4: número de slot, bits de 5 a 7: número de rack); • local_tsap_id[3-16] = extensión TSAP, opcional Nota: Asegúrese de que todos los valores de local_tsap_id sean unívocos dentro de la CPU.
rem_subnet_id	ARRAY [1..6] of BYTE	B#16#0 …	Este parámetro no se utiliza.

Parámetro	Tipo de datos	Valor de arranque	Descripción
rem_staddr	ARRAY [1..6] of BYTE	B#16#0	Dirección IP del punto final del interlocutor, p. ej. para 192.168.2.3: • rem_staddr[1] = 192 • rem_staddr[2] = 168 • rem_staddr[3] = 2 • rem_staddr[4] = 3 • rem_staddr[5-6] = irrelevante Irrelevante para conexiones no especificadas con establecimiento pasivo de la conexión.
rem_tsap_id	ARRAY [1..16] of BYTE	B#16#0	Puerto del interlocutor. Para ID TSAP del interlocutor con ISO on TCP: • rem_tsap_id[1] = B#16#E0; • rem_tsap_id[2] = rack y slot del punto final del interlocutor (bits 0 a 4: número de slot, bits de 5 a 7: número de rack); • rem_tsap_id[3-16] = extensión TSAP, opcional Irrelevante para conexiones no especificadas con establecimiento pasivo de la conexión.
next_staddr	ARRAY [1..6] of BYTE	B#16#0	Este parámetro no se utiliza.
spare	WORD	W#16#0	Este parámetro no se utiliza.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "FW_IOT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ENABLE	Input	BOOL	I, Q, M, D, L, T, C	Con este parámetro se inicia el establecimiento o la interrupción de la conexión: • 0 = la conexión se deshace • 1 = la conexión se establece
CONNECT	Input	ANY	D	Puntero ANY hacia el área de datos del DB que contiene los parámetros para el establecimiento de la conexión TCP. Esta área de datos debe tener un tamaño mínimo de 64 bytes.
ADDRMODE	Input	INT	I, Q, M, D, L o constante	Con este parámetro se especifica el modo en que se produce el direccionamiento de los datos de la petición FETCH o WRITE (en el modo de direccionamiento S7 o S5). • 0 = modo de direccionamiento S7 (La dirección inicial de los datos se interpreta como dirección de byte) • 1 = modo de direccionamiento S5 (La dirección inicial de los datos se interpreta como dirección de palabra)

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NDR	Output	BOOL	I, Q, M, D, L	Parámetro new data record: Con este parámetro se indica que los datos de la petición WRITE se han aplicado correctamente y que el telegrama de acuse se ha generado y enviado.
ERROR	Output	BOOL	I, Q, M, D, L	Este parámetro se activa en los siguientes casos: • Si se ha producido un error en el establecimiento o la interrupción de la conexión • Si se ha producido un error al enviar o recibir los datos. • Si se ha recibido un telegrama de encabezado no válido de FETCH/WRITE.
MODE	Output	BYTE	I, Q, M, D, L	Con este parámetro se indica si se está ejecutando una petición FETCH o WRITE. • 0 = ninguna petición activa • 1 = Petición WRITE • 2 = Petición FETCH
STATUS (Página 7780)	Output	WORD	M, D	En el parámetro STATUS se indican bits de comunicación internos.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro STATUS

Estructura del parámetro STATUS

El parámetro de salida STATUS de las instrucciones "FW_TCP" y "FW_IOT" devuelve información sobre el estado de la comunicación entre la estación S7 y la estación PC.

Esta palabra de estado presenta la siguiente estructura:

Bit	DB de instancia.Estado FB	Descripción
High Byte		
0	NotConnected	Ninguna conexión establecida o interrumpida • El cliente FETCH/WRITE todavía no ha iniciado el establecimiento de la conexión • Error al establecer la conexión
1	ErrorConnect	Error al establecer la conexión (p. ej.: Error en la estructura de los parámetros de conexión)
2	ErrorDisConnect	Error al deshacer la conexión
3	ErrorConType	Tipo de conexión errónea en la estructura de los parámetros de conexión
4	No asignado	
5	BusyConnect	Establecimiento de conexión en curso
6	BusyDisConnect	Desconexión en curso
7	No asignado	

Bit	DB de instancia.Estado FB	Descripción
Low Byte		
8	ErrorRec1	Conexión TCP: • Error en la petición de recepción • FETCH/WRITE No se ha podido recibir el telegrama de encabezado Conexión ISO on TCP • Error en la petición de recepción • FETCH/WRITE No se ha podido recibir el telegrama de encabezado junto con los datos de usuario
9	ErrorRec2	Conexión TCP • Error en la petición de recepción • No se han podido recibir los datos de usuario
10	ErrorSend	Conexión TCP e ISO on TCP • Error en la petición de transmisión • FETCH/WRITE No se ha podido transmitir el telegrama de acuse junto con los datos de usuario
11	No asignado	
12	BusyRec1	Petición de recepción en curso
13	BusyRec2	Petición de recepción en curso
14	BusySend	Petición de transmisión en curso
15	ErrorHeader	Se ha recibido un telegrama de encabezado FETCH/WRITE no definido

Direcciones y variables del bloque de datos de instancia

En el bloque de datos de instancia de la instrucción "FW_TCP" y "FW_IOT" se guardan el estado de error del establecimiento y la interrupción de la conexión, así como el estado de error de la petición de transmisión y recepción.

- El bloque de datos de instancia de la instrucción "FW_TCP" abarca 8348 bytes.
- El bloque de datos de instancia de la instrucción "FW_IOT" abarca 16544 bytes.

La tabla siguiente ofrece una vista general de las direcciones y variables del bloque de datos de instancia en donde se guardan los estados.

Variable	Tipo de datos	Dirección	Descripción
DoneError.STATUS_Connect	WORD	22.0	Estado de establecimiento de la conexión
DoneError.STATUS_REC_1	WORD	24.0	Estado de la primera petición de recepción
DoneError.STATUS_REC_2	WORD	26.0	Estado de la segunda petición de recepción (sólo con TCP)
DoneError.STATUS_SEND	WORD	28.0	Estado de la petición de transmisión
DoneError.STATUS_DisConnect	WORD	30.0	Estado de interrupción de la conexión

Valores del parámetro STATUS

La tabla siguiente ofrece una vista general de los valores de estado del establecimiento de la conexión.

STATUS (W#16#...)	Descripción
0000	Se ha podido establecer la conexión.
8086	El parámetro ID se encuentra fuera del rango admisible.
8087	Se ha alcanzado el número máximo de conexiones, no es posible establecer más conexiones.
8089	El parámetro CONNECT no apunta a un DB.
809A	El parámetro CONNECT apunta a un campo que no tiene la longitud de la descripción de la conexión.
809B	El local_device_id indicado en la descripción de la conexión no concuerda con la CPU.
80A1	La conexión o el puerto ya están ocupados.
80A2	El puerto local o remoto está ocupado por el sistema.
80A3	Se está intentando deshacer una conexión que no existe.
80A4	La dirección IP del punto final de conexión remoto no es válida o coincide con la dirección IP propia
80B2	El parámetro CONNECT apunta a un DB generado con la palabra clave UNLINKED.
80B3	Parametrización incoherente
80B4	Con la variante de protocolo ISO on TCP (connection_type = B#16#12), en el establecimiento pasivo de la conexión (active_est = FALSE) ha vulnerado una o varias de las siguientes condiciones: • local_tsap_id_len >= B#16#02 • local_tsap_id[1] = B#16#E0
80B6	Error de parametrización referente al parámetro connection_type.
80B7	Error en la descripción de la conexión
80C3	Todos los recursos de conexión están ocupados: • Ya se está procesando otro bloque con esta ID en una categoría de prioridad diferente. • Falta de recursos internos
80C4	Error de comunicación temporal: • La conexión no se puede establecer en estos momentos. • La interfaz se está reparametrizando. • La conexión creada está siendo eliminada por la instrucción "TDISCON".
82xx	Información de error general

Servidor web

WWW: Sincronizar las páginas web definidas por el usuario

Descripción

La instrucción WWW inicializa el servidor web de la CPU, o bien sincroniza las páginas web definidas por el usuario con el programa de usuario en la CPU.

Las páginas web definidas por el usuario, junto con el servidor web de la CPU, ofrecen la posibilidad de acceder con un navegador web a páginas web de libre configuración de la CPU.

Mediante instrucciones de scripts (por ej., Javascript) y de código HTML en páginas web definidas por el usuario, es posible transmitir datos a la CPU desde un navegador para su posterior procesamiento, así como visualizar en el navegador datos del área de operandos. Para la sincronización del programa de usuario con el servidor web, y también para la inicialización, es necesario llamar la instrucción WWW en el programa de usuario.

Inicialización

Las páginas web definidas por el usuario se "comprimen" en bloques de datos para que la CPU pueda procesarlas. Durante la configuración deberá generar los correspondientes bloques de datos a partir de los datos fuente (archivos HTML, imágenes, archivos Javascript...). El DB Web Control (predeterminado: DB 333), que contiene información de estado y de control, así como referencias a otros bloques de datos con páginas web encriptadas, desempeña una función particular. Los bloques de datos con páginas web codificadas se denominan DBs de fragmento.

Si los bloques de datos se han cargado en la CPU, esta no "sabe" que las páginas web definidas por el usuario que contiene están encriptadas. Por medio de la instrucción "WWW", por ejemplo, en el OB de arranque, se notifica a la CPU qué DB es el DB de Web Control. Tras esta inicialización, es posible acceder a las páginas web definidas por el usuario desde un navegador web.

Sincronización

Cuando el programa de usuario debe interactuar con las páginas web definidas por el usuario, se debe utilizar la instrucción WWW en la parte cíclica del programa.

Ejemplos de interacción entre programa de usuario y página web:

- Comprobar los datos recibidos
- Recopilar y reenviar los datos para el navegador web solicitante

En este caso, la información de estado ha de poder evaluarse en el acto, debiendo notificarse al servidor la información de control, por ejemplo, para habilitar una página web solicitada por el navegador web.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "WWW":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
CTRL_DB	Input	BLOCK_DB	D	Bloque de datos que describe las páginas web definidas por el usuario (DB Web Control)
RET_VAL	Output	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error. No hay pendiente ninguna solicitud de página web que deba ser habilitada por el programa de usuario.
00xy	x: indica si ha ocurrido un error al inicializar el DB Web Control (CTRL_DB): x=0: no ha ocurrido ningún error. x=1: se han producido errores. El error está codificado en el byte "CTRL_DB.commandstate.last_error" del DB Web Control, consulte la descripción del DB Web Control. y: número de peticiones pendientes. Son posibles varias peticiones (p. ej., petición "0" y "1" pendientes: y="3". y="1": petición "0" y="2": petición "1" y="4": petición "2" y="8": petición "3"
803A	El DB Web Control indicado no se encuentra en la CPU.
8081	La versión o el formato del DB Web Control son incorrectos.
80C1	No se dispone de recursos para inicializar la aplicación web, p. ej., porque solamente pueden ejecutarse dos o cuatro aplicaciones web.

Procesador de comunicaciones

Punto a punto

Sinopsis de la comunicación Freeport

STEP 7 ofrece instrucciones avanzadas que permiten realizar la comunicación Freeport utilizando un protocolo especificado en el programa de usuario. Dichas instrucciones se dividen en dos categorías:

- Instrucciones de configuración
- Instrucciones de comunicación

Nota

Capacidad de la CPU

Las instrucciones Freeport se comunican con los módulos de comunicaciones mediante lectura o escritura de registros.
Por ello, al utilizar las instrucciones debe tenerse en cuenta la capacidad de lectura y escritura de registros de la CPU correspondiente.
Si varias instrucciones de una CPU deben leer o escribir registros de forma simultánea, es posible que el programa de usuario tenga que llamar las instrucciones con un desfase temporal.

Instrucciones de configuración

Antes de que el programa de usuario pueda iniciar la comunicación Freeport, es necesario configurar la interfaz de comunicación y los parámetros para transmitir y recibir datos.

La configuración de interfaces y datos se realiza para cada CM en la configuración de dispositivos o con las siguientes instrucciones del programa de usuario:

- Port_Config (Página 7789)
- Send_Config (Página 7792)
- Receive_Config (Página 7793)
- P3964_Config (Página 7798)

ATENCIÓN

Configuración de dispositivos <-> Instrucciones de configuración

Los parámetros de la configuración de dispositivos se transfieren al CM en cada Power On de la CPU (retorno de la tensión).

Los parámetros de las instrucciones de configuración se transfieren al CM de acuerdo con su programa de usuario.

No existe ninguna sincronización entre los parámetros de la configuración de dispositivos y los parámetros de las instrucciones de configuración, es decir, los parámetros de las instrucciones de configuración no se aplican a la configuración de dispositivos de la CPU. El programa de usuario permite controlar qué parámetros se aplican en el CM en cada momento.

Instrucciones de comunicación

Las instrucciones de la comunicación Freeport permiten al programa de usuario enviar datos a las interfaces de comunicación y recibirlos de estas. Los CM envían datos a los dispositivos de comunicación y los reciben de ellos.

- Send_P2P (Página 7800)
- Receive_P2P (Página 7803)

Nota

Coherencia de datos

- En el caso de que los datos que vayan a enviarse deban transferirse de forma coherente, no deben modificarse desde el ascenso del flanco del parámetro REQ hasta establecer DONE con la instrucción Send_P2P .
- Si los datos recibidos deben leerse de forma coherente, solo deben evaluarse en el ciclo en el que NDR = TRUE.

Con instrucciones adicionales es posible inicializar el búfer de recepción, así como consultar y activar señales RS232 especiales.

- Receive_Reset (Página 7805)
- Signal_Get (Página 7805)
- Signal_Set (Página 7806)

Las siguientes instrucciones permiten leer y escribir funciones avanzadas, siempre que el módulo las soporte.

- Get_Features (Página 7808)
- Set_Features (Página 7808)

Todas las instrucciones Freeport funcionan de forma asíncrona. Por ello, las instrucciones deben llamarse repetidas veces hasta que el parámetro de salida DONE indique que ha finalizado la ejecución.

Utilizando una arquitectura de consultas, el programa de usuario puede determinar el estado de transmisión y recepción. Send_P2P y Receive_P2P pueden ejecutarse a la vez. Los módulos de comunicaciones respaldan los datos enviados y recibidos hasta un tamaño máximo del búfer específico del módulo.

Nota

Resolución de tiempos de bit

Para diversos parámetros se indica el número de tiempos de bits en la velocidad de transferencia configurada. Debido a la indicación del parámetro en tiempos de bit, dicho parámetro es independiente de la velocidad de transferencia. Todos los parámetros con la unidad de tiempos de bit pueden indicarse con un número máximo de 65535.

Aplicación de las instrucciones

Las instrucciones Freeport deben llamarse cíclicamente para consultar los datos recibidos o el fin de una transmisión.

En función del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).
Si la petición se ha completado con DONE = TRUE o NDR = TRUE, se ha ejecutado sin errores.

Nota

Guardar STATUS

Los parámetros DONE, NDR, ERROR y STATUS solo permanecen durante una ejecución del bloque. Por ello, para visualizar STATUS debe copiarlo en un área de datos libre.

maestro

Secuencia típica de un maestro:

1. La instrucción Send_P2P lanza una transferencia al CM.
   Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos.
2. La instrucción Send_P2P se ejecuta en ciclos consecutivos para consultar el estado del proceso de transferencia.
3. Si la instrucción Send_P2P en la salida DONE notifica que la transferencia ha finalizado, el código de usuario puede preparar la recepción de la respuesta.
4. La instrucción Receive_P2P se ejecuta repetidamente para consultar una respuesta. Una vez que el CM ha leído los datos de respuesta, la instrucción Receive_P2P copia la respuesta en la CPU y notifica en la salida NDR que se han recibido datos nuevos.
5. El programa de usuario puede procesar la respuesta.
6. Regreso al paso 1 y repetición de la secuencia.

esclavo

Secuencia típica de un esclavo:

1. El programa de usuario ejecuta la instrucción Receive_P2P en cada ciclo.
2. Una vez que el CM ha recibido una solicitud, la instrucción Receive_P2P notifica en la salida NDR que hay datos nuevos preparados y la solicitud se copia en la CPU.
3. El programa de usuario procesa la solicitud y genera una respuesta.
4. Con la instrucción Send_P2P , la respuesta se devuelve al maestro.
5. La instrucción Send_P2P debe ejecutarse repetidamente para asegurarse de que el proceso de transmisión tiene lugar realmente.
6. Regreso al paso 1 y repetición de la secuencia.

El esclavo debe procurar que Receive_P2P se llame tantas veces como sea necesario para que el maestro pueda recibir una transferencia antes de que cancele el proceso por rebasar el tiempo mientras espera una respuesta. Para cumplir esta tarea, el programa de usuario puede llamar a Receive_P2P desde un OB de ciclo cuyo tiempo de ciclo sea lo suficientemente corto como para recibir una transferencia del maestro antes de que transcurra el timeout ajustado. Si el tiempo de ciclo para el OB se ajusta de modo que se lleven a cabo dos

ejecuciones dentro del timeout ajustado, el programa de usuario puede recibir todas las transferencias sin pérdidas.

Parámetros comunes para las operaciones Freeport

Tabla 9-287 Parámetros de entrada comunes de las instrucciones Freeport

Parámetro	Descripción
REQ	Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. La creación de un flanco adicional en REQ solamente puede tener lugar tras finalizar la petición (DONE o bien ERROR). Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa). Al insertar una instrucción Freeport en el programa, STEP 7 le pedirá que indique el DB de instancia (o que cree mediante STEP 7 un DB de instancia correspondiente). Utilice un DB unívoco para cada llamada de instrucción PtP.
PORT	Al configurar el módulo de comunicaciones se asigna una dirección de puerto. Con el parámetro PORT se comunica a la instrucción la asignación a un determinado módulo de comunicaciones. Tras la configuración se puede seleccionar un nombre simbólico para el puerto estándar. El valor de puerto asignado del CM es la propiedad "identificación de hardware" de la configuración de dispositivos en el S7-1500 o la "dirección de entrada" en el S7-300/400. El nombre simbólico del puerto se asigna en la tabla de símbolos.

Los parámetros de salida DONE, NDR, ERROR y STATUS de las instrucciones Freeport indican el estado de ejecución de las funciones Freeport.

Tabla 9-288 Parámetros de salida DONE, NDR, ERROR y STATUS

Parámetro	Tipo de datos	Estándar	Descripción
DONE	Bool	FALSE	Durante un ciclo se pone a TRUE para indicar que la última solicitud ha concluido correctamente; en caso contrario, FALSE.
NDR	Bool	FALSE	Durante un ciclo se pone a TRUE para indicar que se han recibido nuevos datos; en caso contrario, FALSE.
ERROR	Bool	FALSE	Durante un ciclo se pone a TRUE para indicar que la última solicitud ha concluido con error, el código de error correspondiente se encuentra en STATUS; en caso contrario, FALSE.
STATUS	Word	16#0000 o 16#7000	Estado del evento: • Si está activado el bit DONE o NDR, STATUS se ajusta a 0/16#7000 o a un código de error específico. • Si está activado el bit ERROR, se muestra en STATUS un código de error. • Si no se activa ninguno de los bits mencionados, la instrucción puede devolver resultados de estado que describen el estado actual de la función. El valor en STATUS es válido hasta la siguiente llamada de esta instrucción (con la misma dirección única de puerto).

Tabla 9-289 Parámetros de entrada/salida COM_RST

Parámetro	Tipo de datos	Estándar	Descripción
COM_RST	Bool	FALSE	Inicialización de la instrucción Con TRUE se inicializa la instrucción. Seguidamente, COM_RST se ajusta de nuevo a FALSE. Nota: En el arranque, COM_RST debe ajustarse a TRUE y, después, no deberían modificarse más los parámetros, p. ej., no asignar ningún valor al llamar la instrucción. Tras inicializar el DB de instancia, la instrucción resetea COM_RST.

Nota

Tenga en cuenta que los parámetros DONE, NDR, ERROR y STATUS solo están activados durante un ciclo.

Tabla 9-290 Códigos de error comunes

Código de error	Descripción
16#0000	No hay error
16#7000	Función inactiva
16#7001	Primera llamada tras iniciar una solicitud.
16#7002	Llamada posterior tras iniciar una solicitud.
16#8x3A	Puntero no admisible en parámetro x

Tabla 9-291 Clases de errores comunes del parámetro STATUS

Descripción de la clase	Clases de error	Descripción
Configuración del puerto	16#81Ax	Para la descripción de errores frecuentes en la configuración de interfaces
Configuración de transmisión	16#81Bx	Para la descripción de errores en la configuración de transmisión
Configuración de recepción	16#81Cx	Para la descripción de errores en la configuración de recepción
Transferir	16#81Dx	Para la descripción de errores del tiempo de ejecución al transmitir
Recibir	16#81Ex	Para la descripción de errores del tiempo de ejecución al recibir
Señales cualificadoras RS232	16#81Fx	Para la descripción de errores en relación con el procesamiento de señales

Port_Config: configurar el puerto de comunicación PtP

Descripción

La instrucción Port_Config (configuración del puerto) permite modificar parámetros como la velocidad de transferencia desde el programa durante el funcionamiento. Los datos pendientes del CM se borran al ejecutar Port_Config.

Los cambios de configuración de Port_Config no se guardan en la CPU sino en el CM. En caso de retorno de la tensión, el CM se parametriza con los datos guardados en la configuración de dispositivos.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
PROTOCOL	IN	UInt	Word	0	Protocolo • 0 = protocolo Freeport • 1 = protocolo 3964(R)
BAUD	IN	UInt	Word	6	Velocidad de transferencia del puerto: • 1 = 300 Bit/s • 2 = 600 Bit/s • 3 = 1200 Bit/s • 4 = 2400 Bit/s • 5 = 4800 Bit/s • 6 = 9600 Bit/s • 7 = 19200 Bit/s • 8 = 38400 Bit/s • 9 = 57600 Bit/s • 10 = 76800 Bit/s • 11 = 115200 Bit/s
PARITY	IN	UInt	Word	1	Paridad del puerto: • 1 = ninguna • 2 = paridad par • 3 = paridad impar • 4 = paridad Mark • 5 = paridad Space • 6 = cualquiera
DATABITS	IN	UInt	Word	1	Bits por carácter: • 1 = 8 bits de datos • 2 = 7 bits de datos

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
STOPBITS	IN	UInt	Word	1	Bits de parada: • 1 = 1 bit de parada • 2 = 2 bits de parada
FLOWCTRL	IN	UInt	Word	1	Control del flujo: • 1 = no hay control del flujo • 2 = XON/XOFF • 3 = RTS de HW siempre ON • 4 = RTS de HW con protocolo de intercambio • 5 = RTS de HW siempre ON, ignorar DTR/DSR
XONCHAR	IN	Char		16#0011	Indica el carácter que sirve de carácter XON. Suele ser un carácter DC1 (11H). Este parámetro solo se evalúa cuando está activado el control del flujo de software.
XOFFCHAR	IN	Char		16#0013	Indica el carácter que sirve de carácter XOFF. Suele ser un carácter DC3 (13H). Este parámetro solo se evalúa cuando está activado el control del flujo de software.
WAITIME	IN	UInt	Word	2000	Indica el tiempo durante el que se espera un carácter XON tras recibir un carácter XOFF o bien el tiempo durante el que se espera la señal CTS = ON tras CTS = OFF (de 0 a 65535 ms). Este parámetro solo se evalúa cuando está activado el control del flujo.
MODE	IN	USInt	Byte	0	Modo de operación Los modos de operación admitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) modo a cuatro hilos (punto a punto) • 2 = dúplex (RS422) modo a cuatro hilos (maestro multipunto, CM PtP (ET 200SP)) • 3 = dúplex (RS422) modo a cuatro hilos (esclavo multipunto, CM PtP (ET 200SP)) • 4 = semidúplex (RS485) modo a dos hilos
LINE_PRE	IN	USInt	Byte	0	Estado inicial de la línea de recepción Los estados iniciales admitidos son: • 0 = "sin" estado inicial • 1 = señal R(A) = 5 V, señal R(B) = 0 V (detección Break): en este estado inicial es posible una detección de rotura (Break). Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • 2 = señal R(A) = 0 V, señal R(B) = 5 V: este estado inicial corresponde al estado de reposo (ninguna transmisión activa). Con este estado inicial no es posible la detección de rotura (Break).

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
CABLE_BRK	IN	USInt	Byte	0	Detección Break Se admiten: • 0 = detección Break desactivada • 1 = detección Break activada
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Port_Config Con 1 se inicializa la instrucción. Seguidamente, la instrucción pone COM_RST de nuevo a 0.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Send_Config: configurar el emisor PtP

Descripción

Con la instrucción Send_Config (configuración de transmisión) puede cambiar los parámetros de transmisión durante el funcionamiento desde el programa (condiciones que marcan el principio y fin de los datos que deben enviarse). Los datos pendientes en un CM se borran cuando se ejecuta Send_Config.

Los cambios de configuración de Send_Config no se guardan en la CPU sino en el CM. Los parámetros guardados en la configuración de dispositivos se restablecen al retornar la tensión de la CPU o del módulo de comunicaciones.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Activa el cambio de configuración con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC debe asignarse al puerto CM la dirección de entrada indicada en HWCN.

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
RTSONDLY	IN	UInt	Word	0	Número de milisegundos que debe esperarse tras la activación de RTS antes de iniciar una transferencia de datos de envío. Este parámetro solo es válido cuando está activado el control del flujo de hardware. El rango válido es de 0 a 65535 ms. El valor 0 desactiva la función.
RTSOFFDLY	IN	UInt	Word	0	Número de milisegundos que debe esperarse tras la transferencia de datos de envío antes de que se desactive RTS: este parámetro solo es válido cuando está activado el control del flujo de hardware. El rango válido es de 0 a 65535 ms. El valor 0 desactiva la función.
BREAK	IN	UInt	Word	0	Este parámetro indica que se envía un BREAK antes de iniciar cada trama durante el número indicado de tiempos de bit. El número máximo de tiempos de bit es 65535. El valor 0 desactiva la función.
IDLELINE	IN	UInt	Word	0	Este parámetro indica que la línea permanece vacía antes de iniciar cada trama durante el número indicado de tiempos de bit. El número máximo de tiempos de bit es 65535. El valor 0 desactiva la función.
USR_END	IN	STRING[2]		0	Entrada de los caracteres de fin. Se pueden configurar como máximo 2 caracteres de fin. Se envían el o los caracteres de fin, independientemente de la longitud de trama parametrizada.
APP_END	IN	STRING[5]		0	Entrada de los caracteres que deben añadirse. Se pueden añadir como máximo 5 caracteres.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Send_Config Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Receive_Config: configurar el receptor PtP

Descripción

Con la instrucción Receive_Config (configuración de recepción) puede cambiar los parámetros de recepción durante el funcionamiento desde el programa. Esta instrucción configura las condiciones que marcan el principio y fin de los datos recibidos. Los datos pendientes en un CM se borran cuando se ejecuta Receive_Config.

Los cambios de configuración de Port_Config no se guardan en la CPU sino en el CM. Los parámetros guardados en la configuración de dispositivos se restablecen al retornar la tensión de la CPU o del módulo de comunicaciones.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Activa el cambio de configuración con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
Receive_Conditions	IN	Variant	Any	-	La estructura de datos de Receive_Conditions indica las condiciones de inicio y fin en virtud de las cuales se identifica el principio y fin de una trama.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Receive_Config Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

.

Condiciones de inicio de la instrucción Receive_P2P

La instrucción Receive_P2P utiliza la configuración de dispositivos o bien la indicada por la instrucción Receive_Config para determinar el inicio y el fin de tramas de la comunicación Freeport. El inicio de la trama queda definido por las condiciones de inicio. El inicio de la trama puede especificarse mediante una o varias condiciones de inicio.
Si están activados tanto Break como Idle Line, debe cumplirse Break y, además, Idle Line. A continuación, basta una de las condiciones restantes (carácter de inicio o secuencia de inicio) para iniciar la transmisión de datos.
La condición de inicio "cualquier carácter" no puede combinarse con otras condiciones de inicio.

Estructura del tipo de datos del parámetro Receive_Conditions, parte 1 (condiciones de inicio)

Tabla 9-292 Estructura de Receive_Conditions para condiciones de inicio

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
START.START COND	IN	Word	16#0002	Indicación de la condición de inicio • 01H - detección del carácter de inicio • 02H - cualquier carácter • 04H - detección de Line Break • 08H - detección de Idle Line • 10H - detección de la secuencia de inicio 1 • 20H - detección de la secuencia de inicio 2 • 40H - detección de la secuencia de inicio 3 • 80H - detección de la secuencia de inicio 4 Las condiciones de inicio pueden combinarse sumando los valores.
START.IDLETI ME	IN	Word	16#0028	Número de tiempos de bit necesarios en vacío para que se detecte un nuevo inicio de la trama (valor predeterminado: W#16#28). Solo en combinación con la condición "detección de Idle Line". 0 a FFFF
START.START CHAR	IN	Byte	16#0002	Carácter de inicio para la condición "carácter de inicio". (valor predeterminado: B#16#2)
START.SEQ[1]. CTL	IN	Byte	0	Secuencia de inicio 1, desactivar/activar comparación para cada carácter: (valor predeterminado: B#16#0) Estos son los bits de activación para cada carácter de la secuencia de inicio. • 01H - carácter 1 • 02H - carácter 2 • 04H - carácter 3 • 08H - carácter 4 • 10H - carácter 5 Si se desactiva un bit para un carácter determinado, significa que en esta posición de la secuencia cada carácter representa una secuencia de inicio válida (p. ej. 1FH = se interpretan los 5 caracteres).
START.SEQ[1]. STR[1] .. START.SEQ[1]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 1, carácter de inicio (5 caracteres).
START.SEQ[2]. CTL	IN	Byte	0	Secuencia de inicio 2, desactivar/activar comparación para cada carácter. Valor predeterminado: B#16#0)
START.SEQ[2]. STR[1] .. START.SEQ[2]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 2, carácter de inicio (5 caracteres).

Parámetro	Declaración	Tipo de datos	Ajuste predetermina do	Descripción
START.SEQ[3]. CTL	IN	Byte	0	Secuencia de inicio 3, desactivar/activar comparación para cada carácter. Valor predeterminado: B#16#0
START.SEQ[3]. STR[1] .. START.SEQ[3]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 3, carácter de inicio (5 caracteres).
START.SEQ[4]. CTL	IN	Byte	0	Secuencia de inicio 4, desactivar/activar comparación para cada carácter. Valor predeterminado: B#16#0
START.SEQ[4]. STR[1] .. START.SEQ[4]. STR.[5]	IN	Char[5]	0	Secuencia de inicio 4, carácter de inicio (5 caracteres).

Ejemplo

Observe los siguientes datos recibidos en código hexadecimal: "68 10 aa 68 bb 10 aa 16". Encontrará las secuencias de caracteres de inicio configuradas en la tabla siguiente. Las secuencias de caracteres de inicio se evalúan una vez se ha recibido correctamente el primer carácter 68H. Tras la recepción correcta del cuarto carácter (el segundo 68H) se ha cumplido la condición de inicio 1. Si se cumplen las condiciones de inicio, empieza la evaluación de las condiciones de fin.

El procesamiento de la secuencia de caracteres inicial puede finalizarse debido a diferentes errores de paridad, framing o intervalos entre los caracteres. Dichos errores hacen que los datos no se reciban porque no se cumple la condición de inicio (se genera un mensaje de error).

Tabla 9-293 Condiciones de inicio

Condición de inicio	Primer carácter	Primer carácter +1	Primer carácter +2	Primer carácter +3	Primer carácter +4
1	68H	xx	xx	68H	xx
2	10H	aaH	xx	xx	xx
3	dcH	aaH	xx	xx	xx
4	e5H	xx	xx	xx	xx

Condiciones de fin de la instrucción Receive_P2P

El final de una trama se define por la primera aparición de una o varias condiciones de fin configuradas.

Las condiciones de fin se configuran en las propiedades de la interfaz de comunicación de la configuración de dispositivos o bien mediante la instrucción Receive_Config. Tras retornar la tensión de la CPU o del módulo de comunicaciones, los parámetros de recepción (condiciones de inicio y fin) se ponen de nuevo a los ajustes de la configuración de dispositivos. Si el programa de usuario de STEP 7 ejecuta Receive_Config, los ajustes cambian a los parámetros de Receive_Config .

Estructura del tipo de datos del parámetro Receive_Conditions, parte 2 (condiciones de fin)

Tabla 9-294 Estructura de Receive_Conditions para condiciones de fin

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
END.ENDCOND	IN	Word	0	Este parámetro indica la condición para el fin de trama: • 01H - tiempo de respuesta excedido • 02H - tiempo de espera del mensaje excedido • 04H - tiempo de retardo de caracteres • 08H - longitud máxima de trama • 10H - leer longitud del mensaje en el mensaje (N+LEN+M) • 20H - secuencia de fin • 40H - longitud fija de trama
END.FIXLEN	IN	Word	1	Longitud fija de trama: solo se utiliza si se selecciona la condición de fin "longitud fija". De 1 a 4000 bytes (en función del módulo hasta 4 kbytes)
END.MAXLEN	IN	Word	1	Longitud máxima de trama: solo se utiliza si se selecciona la condición de fin "longitud máxima". De 1 a 4000 bytes (en función del módulo hasta 4 kbytes)
END.N	IN	Word	0	Posición de byte del campo de longitud en la trama. Solo se utiliza con la condición de fin N+LEN+M. De 1 a 4000 bytes (en función del módulo hasta 4 kbytes)
END.LENGTHSIZE	IN	Word	0	Tamaño del campo de longitud (1, 2 o 4 bytes). Solo se utiliza con la condición de fin N+LEN+M.
END.LENGTHM	IN	Word	0	Número de caracteres posteriores al campo de longitud que no están incluidos en el valor del mismo. Este dato solo se utiliza con la condición de fin N+LEN+M. 0 a 255 bytes
END.RCVTIME	IN	Word	200	Indique el tiempo durante el que debe esperarse el primer carácter recibido tras enviar una trama. La instrucción de recepción finalizará con un mensaje de error si dentro del tiempo indicado no se recibe ningún carácter correctamente. Este dato solo se utiliza con la condición "tiempo de respuesta excedido". (0 a 65535 ms). Nota: Este parámetro no puede utilizarse como criterio único de fin, sino solamente en combinación con al menos otra condición de fin.
END.MSGTIME	IN	Word	200	Indique el tiempo que debe esperarse a la recepción completa de la trama tras haber recibido el primer carácter. Este parámetro solo se utiliza cuando está seleccionada la condición "tiempo de espera del mensaje excedido". (0 a 65535 ms)
END.CHARGAP	IN	Word	12	Indique el número máximo de tiempos de bit entre los caracteres. Si el número de tiempos de bit entre los caracteres rebasa por exceso el valor indicado, se cumple la condición de fin. Este dato solo se utiliza con la condición "tiempo de retardo de caracteres". (0 a 65535 tiempos de bit)

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
END.SEQ.CTL	IN	Byte	0	Secuencia de caracteres 1, desactivar/activar comparación para cada carácter: Estos son los bits de activación para cada carácter de la secuencia de fin. El carácter 1 es el bit 0, el carácter 2 es el bit 1, ..., el carácter 5 es el bit 4. Si se desactiva un bit para un carácter concreto, significa que en esta posición de la secuencia cada carácter representa una coincidencia.
END.SEQ.STR[1] .. END.SEQ.STR[5]	IN	Char[5]	0	Secuencia de caracteres 1, carácter de inicio (5 caracteres)

Tabla 9-295 Parámetros generales de la instrucción Receive_P2P

Parámetro	Declaración	Tipo de datos	Ajuste predeterminado	Descripción
GENERAL.MBUF_SIZE	IN	Byte	255	Entrada del número de tramas que deben guardarse en el búfer de recepción del CM. Si no hay otras condiciones activas que influyan en el comportamiento del búfer de recepción (impedir sobrescritura, control del flujo de datos), tras alcanzar el límite se rechazan las nuevas tramas (1 a 255 tramas).
GENERAL.OW_PROT	IN	Byte	0	Activa la función para no sobrescribir la trama en búfer, en caso de que el CM reciba una nueva trama sin que se haya efectuado la lectura del búfer de recepción del CM. Con esto se evita una eventual pérdida de las tramas de recepción que ya están en el búfer. • 0 - no activado • 1 - activado
GENERAL.CLR_MBUF	IN	Byte	0	Activa el borrado del búfer de recepción en el arranque de la CPU. Si este parámetro está activado, el búfer de recepción se borra automáticamente en la transición de la CPU de STOP a RUN. En el búfer de recepción solo hay tramas que fueron recibidas después del arranque de la CPU. • 0 - no activado • 1 - activado

P3964_Config: Configurar el protocolo 3964(R)

Descripción

La instrucción P3964_Config (configuración del protocolo) permite modificar desde el programa y durante el funcionamiento los parámetros del protocolo 3964(R) como tiempo de retardo de caracteres, prioridad y control de bloque.

Los cambios de configuración de P3964_Config no se guardan en la CPU sino en el CM. Los parámetros guardados en la configuración de dispositivos se restablecen al retornar la tensión de la CPU o del módulo de comunicaciones.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BCC	IN	SInt	Byte	1	Activa/desactiva el uso de la comprobación de bloque • 0 = sin control de bloque • 1 = con control de bloque
Priority	IN	SInt	Byte	1	Ajuste de la prioridad • 0 = prioridad baja • 1 = prioridad alta
CharacterDelayTime	IN	UInt	Word	16#00DC	Ajuste del tiempo de retardo de caracteres (en función de la velocidad de transferencia de datos ajustada) (valor predeterminado: 220 ms) De 1 a 65535 ms
AcknDelayTime	IN	UInt	Word	16#07D0	Ajuste del tiempo de retardo de acuse (en función de la velocidad de transferencia de datos ajustada) (valor predeterminado: 2000 ms) De 1 a 65535 ms
BuildupAttempts	IN	SInt	Byte	16#0006	Ajuste del número de intentos de establecer conexión (valor predeterminado: 6 intentos de establecer conexión) 1 a 255
RepetitionAttempts	IN	SInt	Byte	16#0006	Ajuste del número de intentos de transmisión (valor predeterminado: 6 intentos de establecer conexión) 1 a 255
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción P3964_Config Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Send_P2P: transferir datos

Descripción

La instrucción Send_P2P (enviar datos punto a punto) inicia la transferencia de los datos y transfiere el contenido del búfer asignado al módulo de comunicaciones. El programa de la CPU se sigue ejecutando mientras el CM envía los datos a la velocidad de transferencia. No puede haber más de una instrucción de transmisión a la vez para cada módulo de comunicaciones. El CM emite un error si se ejecuta una segunda instrucción Send_P2P mientras el CM está enviando una trama.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.

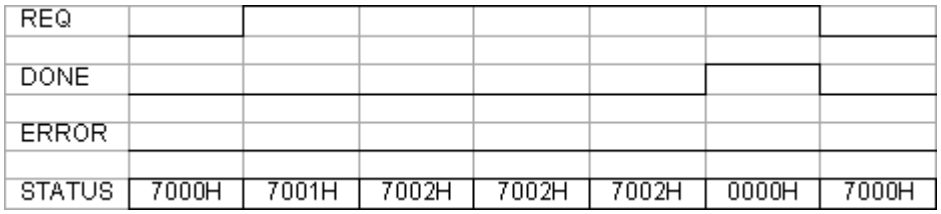
Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
BUFFER	IN	Variant	Any	0	Este parámetro señala el área de memoria del búfer de transmisión. **Notas:** • No se soportan datos booleanos ni campos booleanos. • Si el búfer de transmisión está en el área de memoria optimizada, la máxima longitud admisible de datos enviados es de 1024 bytes. Excepción: los arrays de Byte, Word o DWord se soportan hasta una longitud de 4096 bytes. • Si el búfer de transmisión es de tipo String o WString, el contenido del string se transmite sin la longitud actual y máxima.
LENGTH	IN	UInt	Word	0	Longitud de los datos que deben transmitirse en bytes. Con LENGTH = 0 se transfiere completamente el área de memoria direccionada en el parámetro BUFFER.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Send_P2P Con 1 se inicializa la instrucción. Seguidamente, la instrucción pone COM_RST de nuevo a 0.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Parámetros

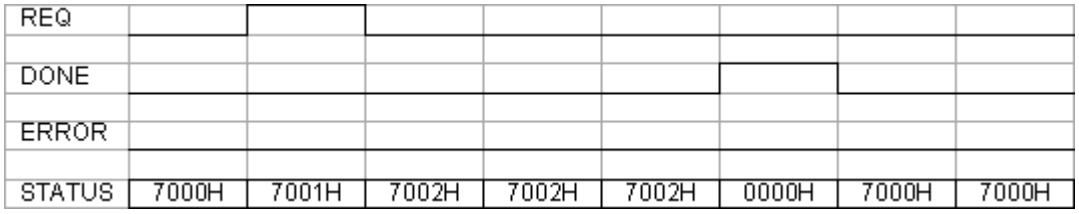
Cuando una instrucción de transmisión se está procesando, las salidas DONE y ERROR están en estado FALSE. Una vez finalizada una instrucción de transmisión, una de las salidas DONE o ERROR se pone a TRUE durante un ciclo para notificar el estado de la instrucción de transmisión. Si ERROR está en estado TRUE, puede evaluarse el código de error en la salida STATUS.

La instrucción emite el estado 16#7001 si la interfaz de comunicación acepta los datos transmitidos. Las ejecuciones posteriores de Send_P2P emiten el valor 16#7002 mientras el CM sigue enviando. Tras finalizar la instrucción de transmisión, el CM emite el estado 16#0000 para la instrucción de transmisión (siempre que no se produzca ningún error). Las ejecuciones posteriores de Send_P2P con REQ = 0 dan el estado 16#7000 (libre).

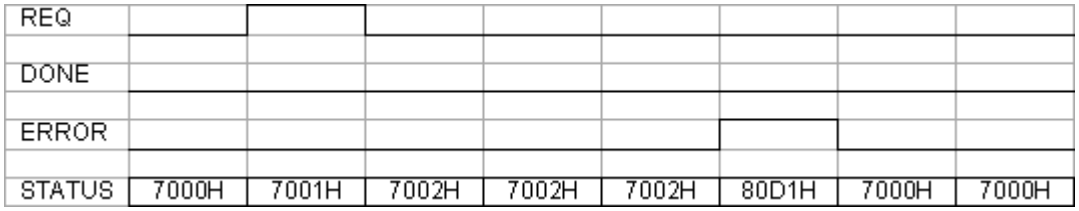
El diagrama siguiente muestra la relación entre los valores de salida y REQ. Se presupone que la instrucción se llama cíclicamente para comprobar el estado del proceso de transmisión (se representa mediante los valores de STATUS).



La figura siguiente muestra cómo los parámetros DONE y STATUS solo son válidos durante un ciclo si en la línea REQ (durante un ciclo) hay un impulso para lanzar la instrucción de transmisión.



La figura siguiente muestra la relación de los parámetros DONE, ERROR y STATUS en caso de error.



Los valores de DONE, ERROR y STATUS son válidos únicamente hasta que Send_P2P se ejecuta con el mismo DB de instancia.

Uso de los parámetros LENGTH y BUFFER en operaciones de comunicación

Interacción de los parámetros LENGTH y BUFFER para Send_P2P

El tamaño mínimo de los datos que puede enviar la instrucción Send_P2P es de un byte. El parámetro BUFFER define el tamaño de los datos que deben enviarse si durante una llamada se transfiere un "0" en el parámetro LENGTH. Para el parámetro BUFFER no es posible utilizar el tipo de datos Bool ni matrices del tipo Bool.

Tabla 9-296 Parámetro LENGTH

LENGTH	Descripción
> 0	Se transfiere el número de bytes parametrizado.
= 0	Se transfiere todo el contenido del búfer de transmisión direccionado mediante BUFFER. Si BUFFER señala a un String, se transfiere todo el contenido del String sin los bytes con la longitud máxima y real.

Tabla 9-297 Parámetro BUFFER

BUFFER	Descripción
Tipo de datos simple	En el envío, se aplica lo siguiente: El valor LENGTH debe contener el número de bytes de este tipo de datos. Ejemplo: con un valor Word, LENGTH debe ser dos. Con un valor DWord o Real, LENGTH debe ser cuatro.
Estructura	En relación con la memoria optimizada, se aplica lo siguiente: la longitud máxima permitida de BUFFER es 1024 Byte; de lo contrario, se admiten hasta 4 kbytes en función del módulo. En la transmisión se aplica lo siguiente: el valor de LENGTH puede contener un número de bytes inferior a la longitud completa de bytes de la estructura; en este caso, solo se envían los primeros bytes LENGTH de la estructura de BUFFER.
Array	En relación con la memoria optimizada, se aplica lo siguiente: si el tipo de datos de la matriz es diferente de Byte, Word o DWord, la longitud máxima permitida del búfer es 1024 Byte. Si la memoria no está optimizada, pueden transferirse hasta 4 kbytes en función del módulo, independientemente de la estructura de datos. En la transmisión se aplica lo siguiente: el valor de LENGTH puede contener un número de bytes inferior a la longitud completa de bytes de la matriz, siendo siempre un múltiplo del número de bytes del elemento de datos. Ejemplo: el parámetro LENGTH de una matriz del tipo Word debe ser un múltiplo de dos y, con una matriz del tipo Real, un múltiplo de cuatro. Si se indica LENGTH, se transfiere el número de elementos de matriz. Si, por ejemplo, BUFFER contiene una matriz con 15 elementos DWord (60 Bytes en total) y se indica LENGTH = 20, se transfieren los cinco primeros elementos DWord de la matriz.
String	El parámetro LENGTH contiene el número de caracteres que deben transmitirse. Solo se transmiten los caracteres de String. Los bytes con la longitud máxima y real de String no se transmiten.

Receive_P2P: recibir datos

Descripción

La instrucción Receive_P2P (datos recibidos mediante una comunicación punto a punto) comprueba las tramas que se han recibido en el CM. Cuando hay una trama disponible, el CM la transfiere a la CPU. Un error de recepción se indica en el parámetro STATUS.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7 300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BUFFER	IN	Variant	Any	0	Este parámetro señala la dirección de inicio del búfer de recepción. Este búfer debe ser lo suficientemente grande como para recibir la longitud máxima de la trama. Nota: • No se soportan datos booleanos o campos booleanos. • Si el búfer de recepción está en el área de memoria optimizada, la máxima longitud admisible de datos recibidos es de 1024 bytes. Excepción: los arrays de Byte, Word o DWord se soportan hasta una longitud de 4096 bytes. • Si el búfer de recepción es de tipo String o WString, los datos recibidos se escriben en el contenido del string, y la longitud actual del string se ajusta de forma correspondiente.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Receive_P2P Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
LENGTH	OUT	UInt	Word	0	Longitud de la trama recibida en bytes
NDR	OUT	Bool		FALSE	TRUE durante un ciclo cuando hay datos nuevos preparados y la instrucción ha finalizado correctamente.
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Si ERROR está en estado TRUE, puede evaluarse el código de error en la salida STATUS. El valor de STATUS suministra el motivo para finalizar la operación de recepción en el CM. Este suele ser un valor positivo que indica que la operación de recepción ha sido correcta y qué criterio de fin de trama se ha detectado.
Si el valor de STATUS es negativo (se ha activado el bit más significativo del valor hexadecimal), la operación de recepción ha finalizado debido a una condición de error, como error de paridad, framing o desbordamiento.

Cada módulo de comunicaciones puede respaldar un número de tramas específico del módulo. Si hay varias tramas disponibles en el CM, la instrucción Receive_P2P emite la más antigua (FIFO).

Receive_Reset: borrar el búfer de recepción

Descripción

La instrucción Receive_Reset (resetear receptor) borra el búfer de recepción del CM.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Receive_Reset Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo significa que la última solicitud se ha ejecutado correctamente.
ERROR	OUT	Bool		FALSE	TRUE significa que la última solicitud se ha ejecutado con errores. Si esta salida es TRUE, la salida STATUS contiene los códigos de error correspondientes.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Signal_Get: leer estado

Descripción

La instrucción Signal_Get (consultar señales RS232) lee los estados actuales de las señales cualificadoras RS232 y las muestra en las salidas correspondientes de la instrucción.

Nota
Limitación

Esta función solo puede aplicarse en los CM RS232 BA y RS232 HF.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la transferencia de datos al CM con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Signal_Get Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
NDR	OUT	Bool		FALSE	TRUE durante un ciclo cuando las señales cualificadoras RS232 se han leído y la instrucción ha finalizado correctamente.
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))
DTR	OUT	Bool		FALSE	Terminal de datos preparado, módulo disponible (salida)
DSR	OUT	Bool		FALSE	Terminal de datos preparado, dispositivo de comunicación preparado (entrada)
RTS	OUT	Bool		FALSE	Solicitud de transmisión, módulo preparado para enviar (salida)
CTS	OUT	Bool		FALSE	Listo para transmitir, el dispositivo de comunicación puede recibir datos (entrada)
DCD	OUT	Bool		FALSE	Señal del soporte de datos detectada, nivel de señal recibido
RING	OUT	Bool		FALSE	Indicador de llamada, aviso de una llamada entrante

Signal_Set: activar señales cualificadoras

Descripción

La instrucción Signal_Set (establecer señales RS232) permite establecer las señales de comunicación RS232.

Esta función solo es válida para el CM RS232.

Nota
Limitaciones

Esta función solo puede aplicarse en los CM RS232 BA y RS232 HF.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
SIGNAL	IN	Byte		0	Selección de la señal que debe establecerse (se admiten varias): • 01H = RTS • 02H = DTR • 04H = DSR (solo válido para el tipo de interfaz DCE)
RTS	IN	Bool		FALSE	Solicitud de transmisión, módulo preparado para enviar Definir este valor en la salida (TRUE o FALSE), valor predeterminado: FALSE
DTR	IN	Bool		FALSE	Terminal de datos preparado, módulo preparado Definir este valor en la salida (TRUE o FALSE), valor predeterminado: FALSE
DSR	IN	Bool		FALSE	Terminal de datos preparado (solo válido para el tipo de interfaz DCE), no utilizado.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Signal_Set Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la última solicitud se haya ejecutado con errores
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

.

Get_Features: obtener funciones avanzadas

Descripción

La instrucción Get_Features (obtener funciones avanzadas) permite consultar información relacionada con la capacidad del módulo para soportar CRC y generar avisos de diagnóstico, siempre que el módulo la soporte.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	PORT	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Get_Features Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
NDR	OUT	Bool		FALSE	TRUE durante un ciclo cuando hay datos nuevos preparados y la instrucción ha finalizado correctamente
MODBUS_CRC	OUT	Bool		FALSE	Soporte de Modbus CRC
DIAG_ALARM	OUT	Bool		FALSE	Generación de avisos de diagnóstico
SUPPLY_VOLT	OUT	Bool		FALSE	Diagnóstico disponible para Falta tensión de alimentación L+
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Set_Features: activar funciones avanzadas

Descripción

La instrucción Set_Features (seleccionar funciones avanzadas) permite activar la compatibilidad con CRC y la generación de avisos de diagnóstico, siempre que el módulo la soporte.

Parámetros

Parámetro	Declaración	Tipo de datos		Ajuste predeterminado	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	La instrucción para establecer funciones avanzadas se inicia con un flanco ascendente en esta entrada.
PORT	IN	PORT (UInt)	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
EN_MODBUS_CRC	IN	Bool		FALSE	Activar soporte de Modbus CRC
EN_DIAG_ALARM	IN	Bool		FALSE	Activar generación de avisos de diagnóstico
EN_SUPPLY_VOLT	IN	Bool		FALSE	Activar diagnóstico para Falta tensión de alimentación L +
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Set_Features Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	TRUE durante una ejecución después de que la última solicitud se haya ejecutado correctamente
ERROR	OUT	Bool		FALSE	TRUE durante un ciclo después de que la instrucción haya finalizado con errores.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7809))

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-

Código de error	Descripción	Solución
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		

Código de error	Descripción	Solución
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)

Código de error	Descripción	Solución
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Recibir datos con Freeport (Página 1028)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)

Código de error	Descripción	Solución
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.

Código de error	Descripción	Solución
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.

Código de error	Descripción	Solución
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		

Código de error	Descripción	Solución
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 [1)]	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 [1)]	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 [1)]	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)

Código de error	Descripción	Solución
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.

Código de error	Descripción	Solución
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.

Código de error	Descripción	Solución
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		

Código de error	Descripción	Solución
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.

Código de error	Descripción	Solución
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.

Código de error	Descripción	Solución
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble

Código de error	Descripción	Solución
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

1) Solo en instrucciones para CPU S7-300/400

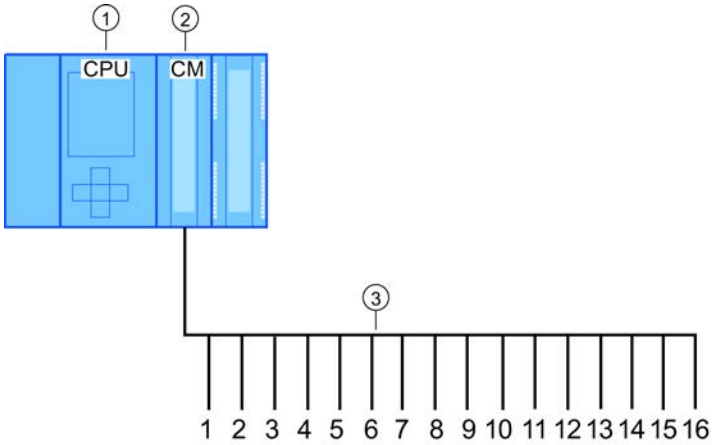
USS

Resumen de la comunicación USS

Comunicación USS

Las instrucciones USS controlan el funcionamiento de accionamientos que soportan el protocolo de la interfaz serie universal (USS). Los módulos de comunicación PtP hacen posible la comunicación con varios accionamientos a través de conexiones RS485 e instrucciones USS. Cada puerto RS485 puede operar un máximo de 16 accionamientos.

El protocolo USS utiliza una red de maestros y esclavos para la comunicación a través de un bus serie. El maestro utiliza un parámetro de dirección para enviar datos a un esclavo concreto. Un esclavo no puede enviar nunca por sí mismo sin haber recibido previamente una solicitud. La comunicación entre los diferentes esclavos no es posible. La comunicación USS funciona en modo semidúplex. La figura siguiente muestra un diagrama de red para un ejemplo de aplicación con 16 accionamientos.



① CPU
② CM
③ Accionamientos USS en una red USS

Figura 9-14 Ejemplo de interconexión con un módulo de comunicación S7-1500

Nota

Comunicación con un accionamiento mediante RS232

En principio, también es posible utilizar CM PtP RS232 BA y CM PtP RS232 HF para la comunicación con un accionamiento. Sin embargo, solo es posible conectar **un** accionamiento a un puerto RS232.

Comunicación con un accionamiento mediante RS422

En principio, también es posible utilizar la interfaz RS422 del CM PtP RS422/485 BA y CM PtP RS422/485 HF para la comunicación con un accionamiento. Sin embargo, solo es posible conectar **un** accionamiento a un puerto RS422.

Instrucciones USS en el programa

- USS_Port_Scan: la instrucción USS_Port_Scan permite la comunicación a través de un módulo de comunicaciones con hasta 16 accionamientos, mediante una red USS (debe ser llamada de forma cíclica).
  En el programa solo hay una instrucción USS_Port_Scan por puerto de comunicación PtP que controla la transferencia a todos los accionamientos.
- USS_Drive_Control: la instrucción USS_Drive_Control permite preparar los datos de USS_Port_Scan para un accionamiento y mostrar los datos recibidos.
  USS_Drive_Control configura los datos que deben enviarse y evalúa los datos recibidos por USS_Port_Scan en una solicitud anterior.
- USS_Read_Param: la instrucción USS_Read_Param permite leer parámetros de un accionamiento.
- USS_Write_Param: la instrucción USS_Write_Param permite modificar los parámetros de un accionamiento.

Requisitos para utilizar el protocolo USS

Las cuatro instrucciones USS utilizan 2 FB y 2 FC para soportar el protocolo USS. Para cada red USS se emplea un bloque de datos de instancia para USS_Port_Scan y un bloque de datos de instancia común para todas las llamadas de USS_Drive_Conrol.

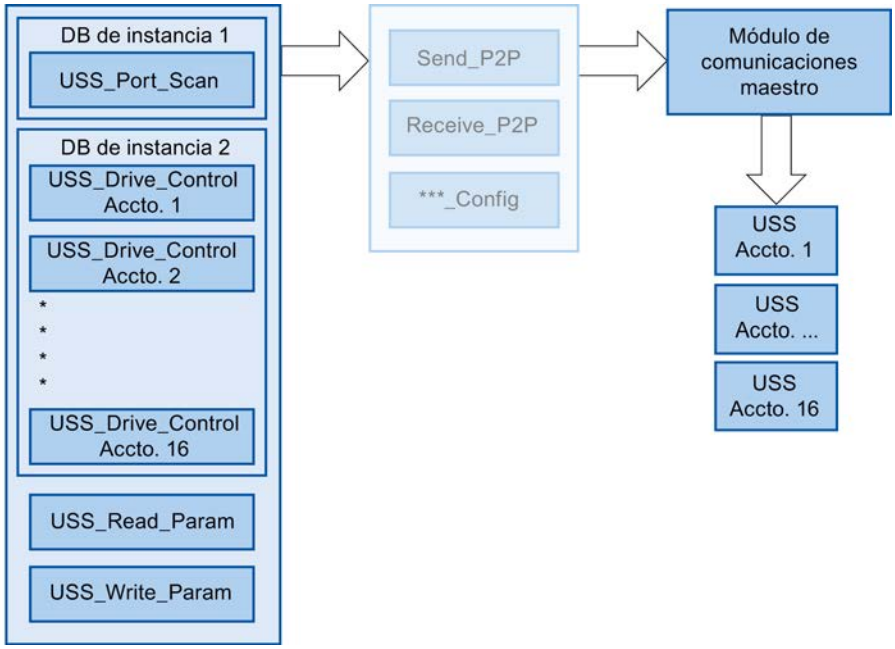


Figura 9-15 Ejecución del programa - USS

Todos los accionamientos (máx. 16) que están conectados a un puerto RS485 forman parte de la misma red USS. Todos los accionamientos que están conectados a otro puerto RS485 forman parte de otra red USS. Cada red USS se gestiona por medio de un bloque de datos de instancia único para todas las instrucciones USS_Drive_Control y otro bloque de datos de instancia para la instrucción USS_Port_Scan. Todas las instrucciones que forman parte de una red USS deben utilizar conjuntamente el bloque de datos de instancia para

USS_Drive_Conrol. Para ello, las instrucciones USS_Port_Scan, USS_Read_Param y USS_Write_Param cuentan con el parámetro USS_DB, que debe interconectarse con el parámetro (estático) USS_DB del DB de instancia de la instrucción USS_Drive_Control.

- Las instrucciones USS_Drive_Control y USS_Port_Scan son bloques de función (FB). Si se inserta la instrucción USS_Drive_Control o USS_Port_Scan en el editor de programación, deberá asignarse un DB a este FB en el cuadro de diálogo "Opciones de llamada". Si se trata de la primera instrucción USS_Drive_Control del programa para esta red USS, existe la posibilidad de aplicar la asignación estándar de DB (o cambiar el nombre en caso necesario) y el nuevo DB se creará automáticamente. Sin embargo, si no se trata de la primera instrucción USS_Drive_Control para este accionamiento, es necesario seleccionar en la lista desplegable del cuadro de diálogo "Opciones de llamada" el DB que se ha asignado previamente a esta red USS.
- Las instrucciones USS_Port_Scan y USS_Read_Param son funciones (FC). Si se insertan estas FC en el editor, no se asignará ningún DB. Si se insertan estas FC o la instrucción USS_Port_Scan en el editor, deberá asignarse a la entrada USS_DB de estas instrucciones el parámetro USS_DB del DB de instancia correspondiente de USS_Drive_Control. Haga doble clic en el campo de parámetros y pulse seguidamente el icono para visualizar los DB disponibles. Introduzca un punto "." y seleccione el parámetro USS_DB en la lista desplegable.
- La función USS_Port_Scan controla la comunicación entre la CPU y los accionamientos a través del puerto de comunicación punto a punto RS485. Cada vez que se llama esta función se procesa una comunicación con un accionamiento. El programa debe llamar a esta función con la suficiente rapidez como para que los accionamientos no notifiquen un tiempo excedido. Para garantizar un comportamiento temporal constante de la transmisión de tramas, esta instrucción debe llamarse con un OB de alarma cíclica.
- La instrucción USS_Drive_Control proporciona al programa acceso a un accionamiento concreto de la red USS. Sus entradas y salidas corresponden a los estados y las funciones de control del accionamiento. Si hay 16 accionamientos en la red, USS_Drive_Control debe llamarse 16 veces en el programa, es decir, una vez por accionamiento.
  La instrucción USS_Drive_Control solo debería llamarse desde un OB cíclico.
- Con las funciones USS_Read_Param y USS_Write_Param se leen y escriben los parámetros operativos del accionamiento. Dichos parámetros controlan el funcionamiento interno del accionamiento. Encontrará una definición de estos parámetros en el manual del accionamiento. El programa puede contener un número cualquiera de estas funciones, aunque únicamente puede estar activa una solicitud de lectura o escritura para un accionamiento. Las funciones USS_Read_Param y USS_Write_Param solo deben llamarse desde el OB de ciclo de un programa principal.

ATENCIÓN
Llamadas de las instrucciones USS Llame USS_Drive_Control, USS_Read_Param y USS_Write_Param únicamente desde un OB de ciclo del programa principal. La instrucción USS_Port_Scan puede llamarse desde cualquier OB, aunque normalmente se llama desde un OB de alarma cíclica. No utilice las instrucciones USS_Drive_Control, USS_Read_Param y USS_Write_Param en un OB con una prioridad más alta que la instrucción USS_Port_Scan correspondiente. No inserte USS_Port_Scan en el programa principal ni USS_Read_Param en un OB de alarma cíclica, por ejemplo. Si la ejecución de USS_Port_Scan es interrumpida por otra instrucción, pueden producirse errores imprevistos.

Calcular el tiempo para la comunicación con el accionamiento

La comunicación con el accionamiento se lleva a cabo de forma asíncrona al ciclo de la CPU. La CPU suele ejecutar varios ciclos antes de que finalice la comunicación con un accionamiento.

Para que la vigilancia de tiempo parametrizada durante el accionamiento no responda, las tramas de transmisión deben enviarse al accionamiento dentro de esta vigilancia de tiempo. También debe incluirse el número de intentos si se requieren varios para completar la transacción debido a errores de comunicación. De forma predeterminada, para el protocolo USS se realizan hasta 2 intentos en cada transacción.
El intervalo entre dos tramas de transmisión se calcula del siguiente modo:

(N * 5 * tiempo de ciclo + 2 * tiempo de ejecución de tramas) * número de intentos + (timeout de la trama de recepción) * (número de intentos - 1)

N	Número de accionamientos en esta red
Factor 5	Para la transmisión y recepción de tramas se requieren 5 ciclos.
Tiempo de ciclo	Tiempo de ciclo máx. de los OB de alarma cíclica en los que se llama la instrucción USS_Port_Scan.
2 * tiempo de ejecución de tramas	Una vez para cada transmisión y una para cada recepción
Tiempo de ejecución de tramas	Tiempo de ejecución de tramas = (número de caracteres por trama) * (11 Bit por carácter) / (velocidad de transferencia en Bit/s)
Número de intentos	3

Para el "Timeout de la trama de recepción" se aplican los siguientes tiempos (ms)

Bits/s	115200	57600	38400	19200	9600	4800	2400	1200
Receive_Conditions.END.RCVTIME	25	29	33	56	72	100	100	100
Receive_Conditions.END.MSGTIME	25	29	33	56	72	124	240	460

Ejemplo:
5 accionamientos

Velocidad de transferencia = 9600 bits/s
28 caracteres por trama
Tiempo de ciclo = 0,020 s
Intervalo = (5 * 5 * 0,02 s + 2 * 28 * 11 / 1200 s) * 3 = 1,69 s
En este caso, la vigilancia de tiempo del accionamiento debe ajustarse aprox. a 2 segundos.

Consulte también

USS_Port_Scan: comunicación vía red USS (Página 7828)

USS_Port_Scan: comunicación vía red USS

Descripción

La instrucción USS_Port_Scan procesa la comunicación a través de una red USS. STEP 7 crea automáticamente un DB de instancia cuando se inserta la instrucción.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
PORT	IN	Port	Word	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/ WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BAUD	IN	DInt		9600	Velocidad de transferencia de datos para la comunicación USS Se admiten: • 1200 Bit/s • 2400 Bit/s • 4800 Bit/s • 9600 Bit/s • 19200 Bit/s • 38400 Bit/s • 57600 Bit/s • 115200 Bit/s
USS_DB	INOUT	USS_BASE		–	El parámetro USS_DB debe interconectarse con el parámetro (estático) USS_DB del DB de instancia, que se crea e inicializa si se inserta una instrucción USS_Drive_Control en el programa.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
COM_RST	INOUT	---	Bool	FALSE	Inicialización de la instrucción USS_Port_Scan Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
ERROR	OUT	Bool		FALSE	Si es TRUE, la salida indica que se ha producido un error y la salida STATUS es válida.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 7840)).

Solo hay una instrucción USS_Port_Scan por puerto de comunicación PtP en el programa y cada llamada de dicha instrucción controla una transferencia hacia o desde todos los accionamientos de esta red. Todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo DB de instancia.

El programa debe ejecutar la instrucción USS_Port_Scan tantas veces como sea necesario para que no se produzca ningún timeout en el accionamiento (consulte Auto-Hotspot la sección "Calcular el tiempo para la comunicación con el accionamiento").

La instrucción USS_Port_Scan se llama normalmente desde un OB de alarma cíclica con el fin de impedir timeouts del accionamiento y tener disponibles las últimas actualizaciones de datos USS para llamadas de USS_Drive_Control .

Variables de bloques de datos de USS_Port_Scan

La tabla siguiente muestra las variables estáticas públicas del DB de instancia de USS_Port_Scan que pueden utilizarse en el programa.

Tabla 9-298 Variables estáticas en el DB de instancia

Variable	Tipo de datos	Estándar	Descripción
MODE	USInt	4	Modo de operación Los modos de operación admitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) modo a cuatro hilos (punto a punto) • 2 = dúplex (RS422) modo a cuatro hilos (maestro multipunto, CM PtP (ET 200SP)) • 3 = dúplex (RS422) modo a cuatro hilos (esclavo multipunto, CM PtP (ET 200SP)) • 4 = semidúplex (RS485) modo a dos hilos
LINE_PRE	USInt	2	Estado inicial de la línea de recepción Los estados iniciales admitidos son: • 0 = "sin" estado inicial • 1 = señal R(A) = 5 V, señal R(B) = 0 V (detección Break): en este estado inicial es posible una detección de rotura (Break). Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • 2 = señal R(A) = 0 V, señal R(B) = 5 V: este estado inicial corresponde al estado de reposo (ninguna transmisión activa). Con este estado inicial no es posible la detección de rotura (Break).
BRK_DET	USInt	0	Activar aviso de diagnóstico: • 0 - no activado • 1 - activado
RETRIES_MAX	SInt/Byte	2	Número de intentos cuando se producen errores de comunicación. Este parámetro permite establecer el número de intentos de envío de una trama de solicitud en el caso de que la trama de respuesta no se reciba dentro del tiempo ajustado.
EN_DIAG_ALARM	Bool	0	Activar aviso de diagnóstico: • 0 - no activado • 1 - activado
EN_SUPPLY_VOLT	Bool	0	Activar diagnóstico para Falta tensión de alimentación L+ • 0 - no activado • 1 - activado

USS_Drive_Control: Preparar y mostrar datos para el accionamiento

Descripción

La instrucción USS_Drive_Control prepara datos para enviarlos al accionamiento y evalúa los datos de respuesta de este. Para cada accionamiento debe utilizarse una instancia propia de

la instrucción, y todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el mismo bloque de datos de instancia. Debe introducirse el nombre del DB al insertar la primera instrucción USS_Drive_Control. Seguidamente, hay que remitir al DB que se ha creado al insertar la primera instrucción.

STEP 7 crea automáticamente el DB cuando se inserta la instrucción.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
RUN	IN	Bool		FALSE	Bit de arranque del accionamiento: si este parámetro es TRUE, la entrada hace posible el funcionamiento del accionamiento con el número de revoluciones predeterminado. Si RUN pasa a FALSE mientras el accionamiento está en marcha, el motor se desacelera hasta pararse. Este comportamiento difiere de la desconexión de la alimentación (OFF2) y del frenado del motor (OFF3).
OFF2	IN	Bool		FALSE	Bit "Decelerar hasta parar": si este parámetro es FALSE, este bit provoca la parada natural del accionamiento sin frenar.
OFF3	IN	Bool		FALSE	Bit de parada rápida: si este parámetro es FALSE, este bit provoca una parada rápida frenando el accionamiento.
F_ACK	IN	Bool		FALSE	Bit de acuse de error: con este bit se desactiva el bit de error de un accionamiento. El bit se activa tras borrar el error, con lo que el accionamiento detecta que el error anterior no debe seguir notificándose.
DIR	IN	Bool		FALSE	Control de sentido del accionamiento: este bit se activa cuando el accionamiento debe funcionar hacia delante (cuando SPEED_SP es positivo, ver tabla "Interacción de los parámetros SPEED_SP y DIR").
DRIVE	IN	USInt	Byte	1	Dirección del accionamiento: esta entrada es la dirección del accionamiento USS. El rango válido está entre accionamiento 1 y accionamiento 16.
PZD_LEN	IN	USInt	Byte	2	Longitud de palabra: es el número de palabras de datos PZD. Los valores válidos son 2, 4, 6 u 8 palabras.
SPEED_SP	IN	Real		0.0	Consigna de velocidad: es el número de revoluciones del accionamiento proporcionalmente a la frecuencia configurada. Un valor positivo significa que el accionamiento funciona hacia delante (si DIR es true). El rango válido es de 200,00 a -200,00.
CTRL3	IN	Word		0	Palabra de control 3: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL4	IN	Word		0	Palabra de control 4: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
CTRL5	IN	Word		0	Palabra de control 5: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL6	IN	Word		0	Palabra de control 6: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL7	IN	Word		0	Palabra de control 7: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
CTRL8	IN	Word		0	Palabra de control 8: valor que se escribe en un parámetro personalizable del accionamiento. Debe configurarse en el accionamiento (parámetro opcional).
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción USS_Drive_Control Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
NDR	OUT	Bool		FALSE	Datos nuevos preparados: si este parámetro es TRUE, el bit notifica que en la salida hay preparados datos de una nueva solicitud de comunicación.
ERROR	OUT	Bool		FALSE	Se ha producido un error: Si es TRUE, indica que se ha producido un error y la salida STATUS es válida. Todas las salidas restantes se ponen a cero en caso de error. Los errores de comunicación solo se notifican en las salidas ERROR y STATUS de la instrucción USS_Port_Scan.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 7840)).
RUN_EN	OUT	Bool		FALSE	Servicio habilitado: este bit notifica si el accionamiento está en marcha.
D_DIR	OUT	Bool		FALSE	Sentido del accionamiento: este bit notifica si el accionamiento funciona hacia delante. • FALSE – adelante • TRUE – atrás
INHIBIT	OUT	Bool		FALSE	Accionamiento bloqueado: este bit notifica el estado del bit de bloqueo para el accionamiento. • FALSE – no bloqueado • TRUE – bloqueado
FAULT	OUT	Bool		FALSE	Error en accionamiento: este bit notifica que en el accionamiento se ha producido un error. Hay que eliminar el fallo y activar el bit F_ACK para borrar este bit.
SPEED	OUT	Real		0.0	Valor real de velocidad del accionamiento (valor escalado de STATUS 2 del accionamiento): es la velocidad del accionamiento proporcionalmente a la velocidad configurada.
STATUS1	OUT	Word		0	STATUS 1 del accionamiento Este valor contiene bits de estado fijos de un accionamiento.
STATUS3	OUT	Word		0	STATUS 3 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
STATUS4	OUT	Word		0	STATUS 4 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS5	OUT	Word		0	STATUS 5 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS6	OUT	Word		0	STATUS 6 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS7	OUT	Word		0	STATUS 7 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.
STATUS8	OUT	Word		0	STATUS 8 del accionamiento Este valor contiene una palabra de estado personalizable del accionamiento.

Cuando tiene lugar la primera ejecución de USS_Drive_Control, se inicializa en el DB de instancia el accionamiento indicado en la dirección USS (parámetro DRIVE). Tras la inicialización, las instrucciones USS_Port_Scan posteriores pueden iniciar la comunicación con el accionamiento en este número.

Si se cambia el número de accionamiento, la CPU debe ponerse primero a STOP y después a RUN de nuevo para que se inicialice el DB de instancia. Los parámetros de entrada se configuran en el búfer de transmisión USS y las salidas, si existen, se leen de un búfer de respuesta "anterior" válido. USS_Drive_Control configura solo los datos que deben enviarse y evalúa los datos recibidos en una solicitud anterior.

El sentido de giro del accionamiento se controla desde la entrada D_IR (Bool) o mediante el signo (positivo o negativo) de la entrada SPEED_SP (Real). La tabla siguiente explica cómo funcionan estas entradas conjuntamente para especificar el sentido de giro del accionamiento, suponiendo que el motor gira hacia delante.

Tabla 9-299 Interacción de los parámetros SPEED_SP y DIR

SPEED_SP	DIR	Sentido de giro del accionamiento
Valor > 0	0	Atrás
Valor > 0	1	Adelante
Valor < 0	0	Adelante
Valor < 0	1	Atrás

USS_Read_Param: leer datos del accionamiento

Descripción

La instrucción USS_Read_Param lee un parámetro de un accionamiento. Todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar

el bloque de datos de instancia de la instrucción USS_Drive_Control . USS_Read_Param debe llamarse desde un OB de ciclo del programa principal.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
REQ	IN	Bool		–	Con un flanco ascendente en REQ se genera una nueva solicitud de lectura.
DRIVE	IN	USInt	Byte	–	Dirección del accionamiento: DRIVE es la dirección del accionamiento USS. El rango válido está entre accionamiento 1 y accionamiento 16.
PARAM	IN	UInt		–	Número de parámetro: PARAM indica el parámetro de accionamiento que se escribe. El rango de este parámetro está entre 0 y 2047. En algunos accionamientos, se puede acceder a valores de parámetro mayores que 2047 con el byte más significativo del parámetro INDEX. Encontrará más información sobre el acceso a un rango ampliado en el manual del accionamiento.
INDEX	IN	UInt		–	Índice de parámetro: INDEX indica el índice de parámetro del accionamiento en el que debe escribirse. Es un valor de 16 bits en el que el byte menos significativo es el valor real del índice, con un rango de (0 a 255). El byte más significativo también puede ser utilizado por el accionamiento y es específico del mismo. Encontrará más información en el manual del accionamiento.
USS_DB	INOUT	USS_BASE		–	El parámetro USS_DB debe interconectarse con el parámetro (estático) USS_DB del DB de instancia, que se crea e inicializa si se inserta una instrucción USS_Drive_Control en el programa.
DONE[1]	OUT	Bool		FALSE	Si este parámetro es TRUE, en la salida VALUE está pendiente el valor previamente solicitado para el parámetro de lectura. Este bit se activa cuando la instrucción USS_Drive_Control detecta la respuesta de lectura del accionamiento. Este bit se inicializará con la siguiente llamada de USS_Read_Param .
ERROR	OUT	Bool		FALSE	ERROR = TRUE: se ha producido un error y la salida STATUS es válida. Todas las salidas restantes se ponen a cero en caso de error. Los errores de comunicación solo se notifican en las salidas ERROR y STATUS de la instrucción USS_Port_Scan.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 7840)).
VALUE	OUT	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Es el valor del parámetro que se ha leído y solo es válido si el bit DONE es true.

[1] El bit DONE indica que se han leído datos válidos del accionamiento de motor referenciado y se han suministrado a la CPU. No hace referencia al hecho de que la instrucción está preparada para leer inmediatamente otro parámetro. Una solicitud de lectura vacía debe enviarse al accionamiento del motor y también debe ser acusada por la instrucción antes de que se habilite el canal de parámetros para que sea utilizado por el accionamiento correspondiente. La llamada inmediata de USS_Read_Param o USS_Write_Param para el accionamiento de motor específico provoca el error 0x818A.

USS_Write_Param: modificar datos del accionamiento

Nota

Para instrucciones de escritura EEPROM (EEPROM en un accionamiento USS) se aplica lo siguiente:

Mantenga en un número reducido las operaciones de escritura EEPROM para garantizar una larga vida útil del EEPROM.

Descripción

La instrucción USS_Write_Param modifica un parámetro del accionamiento. Todas las funciones USS que están asignadas a una red USS y a un puerto de comunicación PtP deben utilizar el bloque de datos de instancia del USS_Drive_Control.

USS_Write_Param debe llamarse desde el OB de ciclo de un programa principal.

Parámetros

Tabla 9-300 Tipos de datos de los parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
REQ	IN	Bool		–	Con un flanco ascendente en REQ se genera una nueva solicitud de escritura.
DRIVE	IN	USInt	Byte	–	Dirección del accionamiento: DRIVE es la dirección del accionamiento USS. El rango válido está entre accionamiento 1 y accionamiento 16.
PARAM	IN	UInt		–	Número de parámetro: PARAM indica el parámetro de accionamiento que se escribe. El rango de este parámetro está entre 0 y 2047. En algunos accionamientos, se puede acceder a valores de parámetro mayores que 2047 con el byte más significativo del parámetro INDEX. Encontrará más información sobre el acceso a un rango ampliado en el manual del accionamiento.
INDEX	IN	UInt		–	Índice de parámetro: INDEX indica el índice de parámetro del accionamiento en el que debe escribirse. Es un valor de 16 bits en el que el byte menos significativo es el valor real del índice, con un rango de (0 a 255). El byte más significativo también puede ser utilizado por el accionamiento y es específico del mismo. Encontrará más información en el manual del accionamiento.
EEPROM	IN	Bool		–	Guardar en el EEPROM del accionamiento: si es TRUE, la transacción de un parámetro para escribir en el accionamiento se guarda en el EEPROM del accionamiento. Si es FALSE, el valor escrito solo se guarda temporalmente y se perderá la próxima vez que se conecte el accionamiento.
VALUE	IN	Variant (Word, Int, UInt, DWord, DInt, UDInt, Real)	Any (Word, Int, UInt, DWord, DInt, UDInt, Real)	–	Valor del parámetro en el que debe escribirse. Debe ser válido con un flanco ascendente de REQ.
USS_DB	INOUT	USS_BASE		–	El parámetro USS_DB debe interconectarse con el parámetro (estático) USS_DB del DB de instancia, que se crea e inicializa si se inserta una instrucción USS_Drive_Control en el programa.
DONE[1]	OUT	Bool		FALSE	Si es TRUE, la entrada VALUE se ha escrito en el accionamiento. Este bit se activa cuando la instrucción USS_Drive_Control detecta la respuesta de escritura del accionamiento. Este bit se inicializará con la siguiente llamada de USS_Write_Param.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
ERROR	OUT	Bool		FALSE	Si es TRUE, se ha producido un error y la salida STATUS es válida. Todas las salidas restantes se ponen a cero en caso de error. Los errores de comunicación solo se notifican en las salidas ERROR y STATUS de la instrucción USS_Port_Scan .
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 7840)).

[1] El bit DONE indica que se han leído datos válidos del accionamiento de motor referenciado y se han suministrado a la CPU. No hace referencia al hecho de que la librería USS está preparada para leer inmediatamente otro parámetro. Una solicitud de escritura vacía debe enviarse al accionamiento del motor y también debe ser acusada por la instrucción antes de que se habilite el canal de parámetros para que sea utilizado por el accionamiento correspondiente. La llamada inmediata de USS_Read_Param o USS_Write_Param FC para el accionamiento de motor específico provoca el error 0x818A.

Información general sobre el dispositivo de accionamiento

Requisitos para la configuración del accionamiento

- Para los accionamientos debe configurar el uso de 4 palabras PKW (**P**arameter**k**ennungs**w**ert o valor de identificación de parámetro).
- Configure los accionamientos para 2, 4, 6 u 8 palabras PZD (**P**ro**z**ess**d**atenbereich o rango de datos de proceso).
- Procure que el número de palabras PZD en el accionamiento se corresponda con la entrada PZD_LEN de la instrucción USS_Drive_Control del accionamiento.
- Procure que la velocidad de transferencia de todos los accionamientos se corresponda con la entrada BAUD de la instrucción USS_Port_Scan.
- Procure que el accionamiento esté configurado para la comunicación USS.
- Asegúrese de que se haya determinado en el accionamiento que la consigna de frecuencia se proporcione a través de la interfaz USS.
- Asegúrese de que la dirección del accionamiento esté definida (rangos: 1-16).
  Esta dirección debe corresponderse con la entrada DRIVE en el bloque USS_Drive_Control del accionamiento.
- Asegúrese de que la red RS485 esté correctamente cerrada.

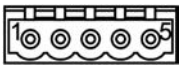
Conexión y configuración del accionamiento SINAMICS V20

Un ejemplo de aplicación para el funcionamiento de un SINAMICS V20 en una S7-1200 se puede consultar en Internet. (http://support.automation.siemens.com/WW/view/es/63696870)

Conexión del accionamiento SINAMICS V20

Ejemplo de conexión de un accionamiento SIEMENS G120(C) a una red USS. Encontrará ejemplos de conexión para otros accionamientos en el manual del correspondiente accionamiento.

La conexión de un accionamiento SINAMICS G120(C) a la red USS se realiza a través de un conector. La conexión es resistente a cortocircuitos y está aislada galvánicamente.



1 0 V potencial de referencia
2 RS485N, recepción y transmisión (-)
3 RS485N, recepción y transmisión (+)
4 Pantalla del cable
5 No utilizado

Figura 9-16 Conexión USS

ATENCIÓN
Diferentes tensiones de referencia Si se conectan entre sí dispositivos que no tienen la misma tensión de referencia, pueden producirse intensidades no deseadas en el cable de conexión. Estas intensidades no deseadas pueden provocar errores de comunicación o daños materiales en los dispositivos. Asegúrese de que todos los dispositivos que estén conectados entre sí a través de un cable de comunicación tengan el mismo conductor de referencia en el circuito o bien estén separados eléctricamente para que no se produzcan intensidades no deseadas. Asegúrese de que la pantalla esté conectada a masa o al pin 1 del conector de bus del accionamiento. Asegúrese de que el borne de cableado 2 (GND) del G120(C) esté conectado a masa.

En caso de que el maestro RS485 (p. ej., una CPU S7-1200 con módulo de comunicaciones CM1241) esté conectado a través de un conector PROFIBUS, los cables de bus deben tenderse como sigue:

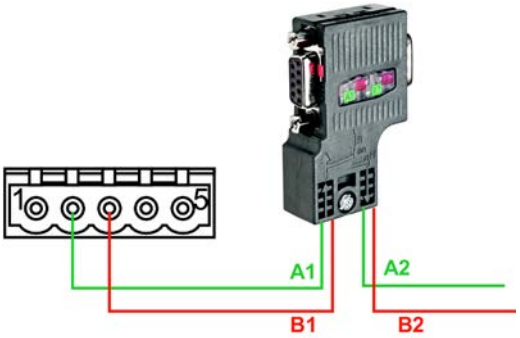


Figura 9-17 Conexión del módulo de comunicaciones

Si el maestro RS485 es un dispositivo terminador de red o si se tiene una conexión punto a punto, deben utilizarse los bornes A1 y B1 (no A2 y B2) del conector PROFIBUS, porque dichos bornes permiten los ajustes de terminación (p. ej. en el conector DP 6ES7972-0BB52-0XA0).

En caso de que el G120(C) esté configurado como dispositivo terminador de red, debe ajustar el interruptor para la resistencia terminadora de bus en "ON".

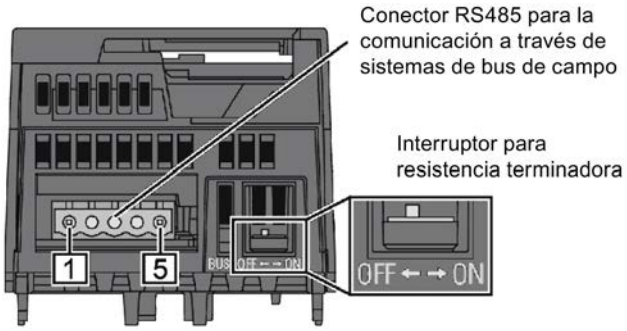


Figura 9-18 Conexión del dispositivo terminador

Configuración de un accionamiento G120(C)

Antes de conectar un accionamiento al S7-1500 o al ET 200SP, hay que asegurarse de que se dispone de los siguientes parámetros de sistema.

Paso	Instrucción	Instrucciones de servicio	
		G120 [1]	G120C [2]
1	Efectúe una puesta en marcha básica del accionamiento con el panel de operador BOP-2. El convertidor ofrece diferentes asignaciones predefinidas (macros) para sus entradas y salidas y la interfaz del bus de campo. Elija en el 9.º paso de la puesta en marcha básica (MAC PAR p15) la macro 21 para la comunicación USS. Con ello estarán también predefinidos los siguientes parámetros: • Velocidad de transferencia (p2020): 38400 bits/s • Número de PZD (p2022): 2 • Número de PKW (p2023): variable Nota: La puesta en marcha básica también se puede ejecutar con el software de puesta en marcha STARTER o con SINAMICS Startdrive.	Cap. 4.4.3	Cap. 6.4.1
2	Defina la dirección de USS del convertidor a través del interruptor de dirección de la Control Unit del G120 o del G120(C). • Área de direcciones válida: 1 … 30 Nota: La dirección USS también se puede ajustar a través del parámetro p2021 o a través de STARTER o de SINAMICS Startdrive.	Cap. 6.2.2.1	Cap. 8.4.2.1
	En los pasos siguientes, con el BOP-2 se accede directamente a parámetros seleccionando sus números y modificando sus valores.	Cap. 4.4.2	Cap. 6.4.2

Paso	Instrucción	Instrucciones de servicio	
		G120 [1]	G120C [2]
3	Adapte los siguientes parámetros de convertidor de su aplicación relativos a la comunicación: • Velocidad de transferencia (p2020), siempre que sea ≠ 38400 bits/s (Preste atención a la concordancia con el parámetro BAUD de la instrucción de comunicación USS_Port_Scan.) • Número de PZD (p2022), siempre que sea ≠ 2 (Preste atención a la concordancia con el parámetro PZD_LEN de la instrucción de comunicación USS_Drive_Control.) • Número de PKW (p2023) = 4 (Modifique a "4" el valor ajustado por defecto a través de la macro 21 a "variable" (127) (las instrucciones USS_Read_Param y USS_Write_Param lo requieren.) • Bus de campo SS Tiempo de vigilancia [ms] (p2040)	Cap. 6.2.2.2	Cap. 8.4.2.1
4	Defina el origen para la consigna de velocidad. • Selec n_cons (p1000[0]) = 6 (La consigna de velocidad se proporciona a través del bus USS.)		
5	Ajuste la magnitud de referencia para la velocidad y la frecuencia. • n_ref f_ref (p2000) = (6.00 min^{-1} hasta 210000.00 min^{-1}) (Todas las velocidades o frecuencias indicadas como relativas se refieren a esta magnitud de referencia. La magnitud de referencia corresponde al 100% o a 4000_{hex} (Palabra) o 4000 0000_{hex} (Palabra doble). En ese caso: $f_{magnitud\ ref}$ (en Hz) = $n_{magnitud\ ref}$ (en ((min^{-1}) / 60) x número de pares de polos))		
6	Transfiera los parámetros a la memoria no volátil. • Guardar par (p0971) = 1		

1) G120 (http://support.automation.siemens.com/CN/view/es/62089662)

2) G120(C) (http://support.automation.siemens.com/WW/view/es/61462568)

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-

Código de error	Descripción	Solución
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).

Código de error	Descripción	Solución
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).

Código de error	Descripción	Solución
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Recibir datos con Freeport (Página 1028)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)

Código de error	Descripción	Solución
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.

Código de error	Descripción	Solución
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		

Código de error	Descripción	Solución
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING

Código de error	Descripción	Solución
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 1)	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 1)	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 1)	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255

Código de error	Descripción	Solución
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.

Código de error	Descripción	Solución
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información

Código de error	Descripción	Solución
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.

Código de error	Descripción	Solución
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1)]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1)]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1)]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1)]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1)]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1)]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG

Código de error	Descripción	Solución
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.

Código de error	Descripción	Solución
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

[1] Solo en instrucciones para CPU S7-300/400

MODBUS (RTU)

Resumen de la comunicación Modbus RTU

Comunicación Modbus RTU

Modbus RTU (Remote Terminal Unit) es un protocolo estándar para la comunicación en la red y utiliza la conexión RS232 o RS422/485 para la transferencia de datos serie entre dispositivos Modbus de la red.

Modbus RTU utiliza una red de maestro/esclavos en la que toda la comunicación sale de un solo dispositivo maestro, mientras que los esclavos simplemente reaccionan a la solicitud del maestro. El maestro envía una solicitud a una dirección de esclavo y solo responde al comando el esclavo con dicha dirección de esclavo.

Caso especial: la dirección de esclavo Modbus 0 envía una trama Broadcast (difusión) a todos los esclavos (sin respuesta del esclavo).

Códigos de función Modbus

- Una CPU que se utiliza como maestro Modbus RTU puede leer y escribir datos y estados E/S en un esclavo Modbus RTU conectado por medio de una conexión de comunicación.
- Una CPU que se utiliza como esclavo Modbus RTU permite a un maestro Modbus RTU conectado por medio de una conexión de comunicación leer y escribir datos y estados E/S en la propia CPU.

Tabla 9-301 Funciones para leer datos: leer datos de programa y E/S descentralizadas

Código de función Modbus	Funciones para lectura de datos del esclavo (servidor) - direccionamiento estándar
01	Leer bits de salida: 1 hasta 2000/1992[1)] bits por solicitud
02	Leer bits de entrada: 1 hasta 2000/1992[1)] bits por solicitud
03	Leer registro de paradas: 1 hasta 125/124[1)] palabras por solicitud
04	Leer palabras de entrada: 1 hasta 125/124[1)] palabras por solicitud

1) Con direccionamiento avanzado

Tabla 9-302 Funciones para escribir datos: escribir E/S descentralizadas y modificar datos de programa

Código de función Modbus	Funciones para escritura de datos en el esclavo (servidor) - direccionamiento estándar
05	Escribir un bit de salida: 1 bit por solicitud
06	Escribir un registro de paradas: 1 palabra por solicitud
15	Escribir uno o varios bits de salida: 1 a 1960 bits por solicitud
16	Escribir uno o varios registros de paradas: 1 a 122 palabras por solicitud

- Los códigos de función Modbus 08 y 11 ofrecen posibilidades de diagnóstico para la comunicación con el dispositivo esclavo.
- La dirección de esclavo Modbus 0 envía una trama Broadcast (de difusión) a todos los esclavos (sin respuesta del esclavo; para códigos de función 5, 6, 15, 16).

Tabla 9-303 Direcciones de estación en la red Modbus

Estación		Dirección
Equipo RTU	Dirección de estación estándar	1 a 247 , así como 0 para Broadcast
	Dirección de estación avanzada	1 a 65535, así como 0 para Broadcast

Direcciones de memoria Modbus

El número de direcciones de memoria Modbus (direcciones de entrada/salida) realmente disponibles se rige por la variante de CPU y la memoria de trabajo disponible.

Instrucciones Modbus RTU en el programa

- Modbus_Comm_Load: es necesario ejecutar Modbus_Comm_Load para establecer parámetros PtP como velocidad de transferencia de datos, paridad y control del flujo de datos. Una vez se ha configurado el módulo de comunicación para el protocolo Modbus RTU, solo pueden utilizarlo la instrucción Modbus_Master o la instrucción Modbus_Slave.
- Modbus_Master: con la instrucción de maestro Modbus es posible utilizar la CPU como dispositivo maestro Modbus RTU para la comunicación con uno o varios dispositivos esclavos Modbus.
- Modbus_Slave: con la instrucción de esclavo Modbus es posible utilizar la CPU como dispositivo esclavo Modbus RTU para la comunicación con un dispositivo maestro Modbus.

Modbus_Comm_Load: configurar un módulo de comunicaciones para Modbus

Descripción

La instrucción Modbus_Comm_Load configura un módulo de comunicaciones para la comunicación a través del protocolo Modbus RTU. Si se inserta la instrucción Modbus_Comm_Load en el programa, se asigna automáticamente un bloque de datos de instancia.

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
REQ	IN	Bool		FALSE	Inicia la instrucción con un flanco ascendente en esta entrada.
PORT	IN	Port	Laddr	0	Una vez que se ha configurado el CM, debe buscarse el valor de puerto CM en la propiedad "identificación de hardware" de la configuración de dispositivos (S7-1500) (dirección de entrada; S7-300/400). El nombre simbólico del puerto se asigna en la ficha "Constantes de sistema" de la tabla de variables PLC. En los sistemas S7-300/400/WinAC, debe asignarse al puerto CM la dirección de entrada indicada en HWCN.
BAUD	IN	UDInt	DWord	9600	Selección de la velocidad de transferencia de datos Los valores admitidos son: 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 76800, 115200 bit/s.
PARITY	IN	UInt	Word	1	Selección de la paridad: • 0 – ninguna • 1 – impar • 2 – par
FLOW_CTRL	IN	UInt	Word	0	Selección del control del flujo: • 0 – (estándar) no hay control del flujo • 1 – control del flujo de hardware con RTS siempre ON (no es válido para CM RS422/485) • 2 – control del flujo de hardware con RTS con protocolo de intercambio (no es válido para CM RS422/485)
RTS_ON_DLY	IN	UInt	Word	0	Selección del retardo a la conexión RTS: • 0 – no hay retardo de "RTS activo" hasta que se envía el primer carácter de la trama. • 1 a 65535 – retardo en milisegundos de "RTS activo" hasta que se envía el primer carácter de la trama (no es válido para CM RS422/485). Los retardos RTS deben aplicarse independientemente de la selección FLOW_CTRL.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/WinAC		
RTS_OFF_DLY	IN	UInt	Word	0	Selección del retardo a la desconexión RTS: • 0 – no hay retardo tras la transferencia del último carácter hasta "RTS inactivo" • 1 a 65535 – retardo en milisegundos desde la transferencia del último carácter hasta "RTS inactivo" (no es válido para puertos RS422/485). Los retardos RTS deben aplicarse independientemente de la selección FLOW_CTRL.
RESP_TO	IN	UInt	Word	1000	Timeout de respuesta: 5 ms a 65535 ms - tiempo en milisegundos que Modbus_Master espera una respuesta del esclavo. Si el esclavo no responde durante este período, Modbus_Master repite la solicitud o la finaliza con un error una vez que se ha enviado el número indicado de repeticiones (ver abajo, parámetro RETRIES).
MB_DB	IN/OUT	MB_BASE		-	Una referencia al bloque de datos de instancia de las instrucciones Modbus_Master o Modbus_Slave. El parámetro MB_DB debe interconectarse con el parámetro MB_DB (estático y, por tanto, no visible en la instrucción) de la instrucción Modbus_Master o Modbus_Slave .
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Modbus_Comm_Load Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DONE	OUT	Bool		FALSE	El bit DONE es TRUE durante un ciclo después de que se haya ejecutado correctamente la última solicitud.
ERROR	OUT	Bool		FALSE	El bit ERROR es TRUE durante un ciclo después de que la última solicitud haya finalizado con errores. El código de error del parámetro STATUS solo es válido en el ciclo en que ERROR = TRUE.
STATUS	OUT	Word		16#7000	Código de error (consulte Mensajes de error (Página 7876))

Modbus_Comm_Load se ejecuta para configurar un puerto para el protocolo Modbus RTU. Una vez se ha configurado un puerto para el protocolo Modbus RTU, solo pueden utilizarlo la instrucción Modbus_Master o Modbus_Slave.

Para la configuración de cada puerto de comunicación utilizado para la comunicación Modbus debe utilizarse una ejecución de Modbus_Comm_Load. Hay que asignar un DB de instancia Modbus_Comm_Load unívoco a cada puerto utilizado. Posteriormente, solo debe volver a ejecutarse Modbus_Comm_Load cuando deban modificarse parámetros de comunicación como velocidad de transferencia de datos o paridad, así como tras restablecerse la red, en caso necesario.

Si se inserta, p. ej., Modbus_Master o Modbus_Slave en el programa, se asigna automáticamente un bloque de datos de instancia a la instrucción. Debe interconectar el parámetro MB_DB de la instrucción Modbus_Comm_Load con el parámetro MB_DB de la instrucción Modbus_Master o Modbus_Slave.

Variables de bloques de datos de Modbus_Comm_Load

La tabla siguiente muestra las variables estáticas públicas del DB de instancia de Modbus_Comm_Load que pueden utilizarse en el programa.

Tabla 9-304 Variables estáticas en el DB de instancia

Variable	Tipo de datos		Estándar	Descripción
	S7-1500	S7-300/400/WinAC		
ICHAR_GAP	Word		0	Tiempo de retardo de caracteres máximo entre caracteres. Este parámetro se indica en milisegundos y sirve para aumentar el período previsto entre los caracteres recibidos. El número correspondiente de tiempos de bit para este parámetro se suma al valor estándar Modbus de 35 tiempos de bit (3,5 tiempos de carácter).
RETRIES	Word		2	Número de intentos repetidos que lleva a cabo el maestro antes de devolver el código de error 0x80C8 para "sin respuesta".
EN_SUPPLY_VOLT	Bool		0	Activar diagnóstico para Falta tensión de alimentación L+
MODE	USInt	Byte	0	Modo de operación Los modos de operación admitidos son: • 0 = dúplex (RS232) • 1 = dúplex (RS422) modo a cuatro hilos (punto a punto) • 2 = dúplex (RS422) modo a cuatro hilos (maestro multipunto, CM PtP (ET 200SP)) • 3 = dúplex (RS422) modo a cuatro hilos (esclavo multipunto, CM PtP (ET 200SP)) • 4 = semidúplex (RS485) modo a dos hilos
LINE_PRE	USInt	Byte	0	Estado inicial de la línea de recepción Los estados iniciales admitidos son: • 0 = "sin" estado inicial • 1 = señal R(A) = 5 V, señal R(B) = 0 V (detección Break): en este estado inicial es posible una detección de rotura (Break). Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • 2 = señal R(A) = 0 V, señal R(B) = 5 V: este estado inicial corresponde al estado de reposo (ninguna transmisión activa). Con este estado inicial no es posible la detección de rotura (Break).
CABLE_BRK	USInt	Byte	0	Detección Break Se admiten: • 0 = detección Break desactivada • 1 = detección Break activada
EN_DIAG_ALARM	Bool		0	Activar aviso de diagnóstico: • 0 - no activado • 1 - activado

Modbus_Master: comunicarse como maestro Modbus

Descripción

La instrucción Modbus_Master se comunica en calidad de maestro Modbus a través de un puerto configurado con la instrucción Modbus_Comm_Load. Si se inserta la instrucción Modbus_Master en el programa, se asigna automáticamente un bloque de datos instancia. El parámetro MB_DB de la instrucción Modbus_Comm_Load debe interconectarse con el parámetro MB_DB (estático) de la instrucción Modbus_Master .

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
REQ	IN	Bool		FALSE	FALSE = no hay solicitud TRUE = solicitud de enviar datos al esclavo Modbus
MB_ADDR	IN	UInt	Word	-	Dirección de estación Modbus RTU: Área de direccionamiento estándar (1 a 247 y 0 para Broadcast) Área de direccionamiento ampliada (1 a 65535 y 0 para Broadcast) El valor 0 está reservado para el Broadcast (difusión) de una trama a todos los esclavos Modbus. Para el Broadcast (difusión) solo se soportan los códigos de función Modbus 05, 06, 15 y 16.
MODE	IN	USInt	Byte	0	Selección de modo: indica el tipo de solicitud (lectura, escritura o diagnóstico). Encontrará más información en la tabla de las funciones Modbus más adelante.
DATA_ADDR	IN	UDInt	DWord	0	Dirección inicial en el esclavo: indica la dirección inicial de los datos a los que debe accederse en el esclavo Modbus. Encontrará las direcciones válidas en la tabla de las funciones Modbus más adelante.
DATA_LEN	IN	UInt	Word	0	Longitud de datos: indica el número de bits o palabras a los que debe acceder esta solicitud. Encontrará las longitudes válidas en la tabla de las funciones Modbus más adelante.
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Modbus_Master Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
DATA_PTR	IN/OUT	Variant	Any	-	Puntero de datos: apunta a la dirección de marca o DB de los datos que deben escribirse o leerse.
DONE	OUT	Bool		FALSE	El bit DONE es TRUE durante un ciclo después de que se haya ejecutado correctamente la última solicitud.
BUSY	OUT	Bool		-	• FALSE – no hay peticiones activas para Modbus_Master • TRUE – se está procesando una petición para Modbus_Master

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
ERROR	OUT	Bool		FALSE	El bit ERROR es TRUE durante un ciclo después de que la última solicitud haya finalizado con errores. El código de error del parámetro STATUS solo es válido en el ciclo en que ERROR = TRUE.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 7876))

Variables en el bloque de datos del maestro Modbus

La tabla siguiente muestra las variables estáticas públicas del DB de instancia de Modbus_Master que pueden utilizarse en el programa.

Tabla 9-305 Variables estáticas en el DB de instancia

Variable	Tipo de datos	Estándar	Descripción
Blocked_Proc_Timeout	Real	0.5	Tiempo (en segundos) que debe esperarse una instancia bloqueada del maestro Modbus antes de que esta se elimine como ACTIVA. Esto puede suceder, por ejemplo, cuando se ha emitido una solicitud de maestro y, seguidamente, el programa deja de llamar la función del maestro antes de que la solicitud haya finalizado por completo. El valor de tiempo debe ser mayor que 0 y menor que 55 segundos, de lo contrario se produce un error. Ver también los apartados "Reglas para la comunicación del Modbus-Master" y "Llamada de la instrucción Modbus_Master con diferentes ajustes de parámetros".
Extended_Addressing	Bool	FALSE	Configura la dirección de estación del esclavo como byte único o doble. • FALSE = dirección de un byte; 0 a 247 • TRUE = dirección de dos bytes (corresponde al direccionamiento avanzado); 0 a 65535

Variable	Tipo de datos	Estándar	Descripción
Compatibility_Mode [1]	Bool	FALSE	Modo de compatibilidad con CP 341 y CP 441-2 con driver para Modbus RTU y con ET 200SP 1SI para Modbus. El valor predeterminado es 0. • FALSE = según especificación Modbus, no compatible • TRUE = compatible – Con FC1 y FC2 se aplica lo siguiente: Los datos leídos en la trama de recepción se escriben palabra por palabra en la memoria direccionada de la CPU y se sustituyen byte por byte. Si el número de bits que deben transmitirse no es múltiplo de 16, los bits no relevantes de la última palabra se ponen a cero. – Con FC15 se aplica lo siguiente: Las palabras que deben transmitirse se leen una a una en la memoria direccionada y se escriben byte por byte en la trama de transmisión. Si el número de bits que deben transmitirse no es múltiplo de 8, los bits no relevantes del último byte se leen sin cambios en la memoria direccionada y se insertan en la trama de transmisión.
MB_DB	MB_BASE	-	El parámetro MB_DB de la instrucción Modbus_Comm_Load debe conectarse con el parámetro MB_DB de la instrucción Modbus_Master.

[1] Los módulos de comunicaciones PtP se comportan según la especificación Modbus. Para que Modbus se comporte como en CP 341, CP 441-2 y ET 200SP 1SI, utilice el parámetro "Compatibility_Mode".

El programa puede escribir valores en las variables Blocked_Proc_Timeout y Extended_Addressing para controlar las operaciones del maestro Modbus.

Reglas para la comunicación del Modbus-Master

- Modbus_Comm_Load debe ejecutarse para configurar un puerto con el fin de que la instrucción Modbus_Master pueda comunicarse con dicho puerto.
- Si debe utilizarse un puerto como Modbus Master, Modbus _Slave no debe utilizar este puerto. Con este puerto pueden utilizarse una o varias instancias de Modbus_Master [1]. Sin embargo, todas las ejecuciones de Modbus_Master deben utilizar el mismo DB de instancia para el puerto.
- Las instrucciones Modbus no utilizan eventos de alarma de la comunicación para controlar el proceso de comunicación. El programa debe consultar la instrucción Modbus_Master si hay peticiones terminadas (DONE, ERROR).
- Se recomienda llamar todas las ejecuciones de Modbus_Master para un puerto concreto desde un OB de ciclo de programa. Las instrucciones de maestro Modbus solo pueden ejecutarse en cada caso en un ciclo de programa o bien en un nivel de procesamiento cíclico/controlado por tiempo. No deben procesarse en distintos niveles de procesamiento. La interrupción de preferencia de una instrucción de maestro Modbus por otra instrucción de maestro Modbus en un nivel de procesamiento con mayor prioridad provoca un funcionamiento incorrecto. Las instrucciones de maestro Modbus no deben procesarse en niveles de arranque, diagnóstico o errores de tiempo.

[1)] El concepto "instancia de maestro Modbus" significa en este caso una llamada de la instrucción Modbus_Master con la misma interconexión que una instrucción Modbus_Comm_Load y el mismo ajuste para los parámetros MB_ADDR, MODE, DATA_ADDR y DATA_LEN.

Ejemplo.

Modbus_Master se llama con MODE=0 y DATA_ADDR=10

Esta petición está activa hasta que finaliza con DONE=1 o ERROR=1 o hasta que ha transcurrido la vigilancia de tiempo parametrizada en el parámetro Blocked_Proc_Timeout . Si una vez finalizada la vigilancia de tiempo se inicia una nueva petición antes de que la petición anterior haya finalizado, la petición anterior se cancelará sin mensaje de error.

En caso de que, mientras se ejecuta esta petición, se produzca una segunda llamada de la instrucción con los mismos datos de instancia pero con ajustes distintos en los parámetros MODE y DATA_ADDR, esta segunda llamada finalizará con ERROR=1 y STATUS=8200.

Llamada de la instrucción Modbus_Master con diferentes ajustes de parámetros

Si el programa debe emplazar varias llamadas de la instrucción Modbus_Master con distintos ajustes para MB_ADDR, MODE, DATA_ADDR o DATA_LEN, debe asegurarse que solo haya una llamada activa a la vez, ya que, de lo contrario, se emitirá el mensaje de error 0x8200 (la interfaz está ocupada por una petición en curso).

Si el procesamiento de la llamada no puede completarse, se activará la vigilancia de tiempo a través del parámetro Blocked_Proc_Timeout y se cancelará la petición en curso.

Parámetro REQ

FALSE = no hay solicitud; TRUE = solicitud de enviar datos al esclavo Modbus

Habilitación de la transferencia solicitada. Con ello, el contenido del búfer se transfiere a la interfaz de comunicación punto a punto.

Por medio de los parámetros DATA_ADDR y MODE se selecciona el código de función Modbus.

DATA_ADDR (dirección inicial Modbus en el esclavo): indica la dirección inicial de los datos a los que debe accederse en el esclavo Modbus.

La instrucción Modbus_Master utiliza una entrada MODE en lugar de una entrada de código de función. La combinación entre MODE y DATA_ADDR especifica el código de función que se utiliza en la trama Modbus propiamente dicha. La tabla siguiente muestra la asignación entre el parámetro MODE, el código de función Modbus y el área de direcciones Modbus en DATA_ADDR.

Tabla 9-306 Funciones Modbus

MODE	DATA_ADDR (dirección Modbus)			DATA_LEN (longitud de los datos)			Código de función Modbus	Operación y datos		
0				Bits por solicitud			01	Leer bits de salida:		
	1	a	9999	1	a	2000/1992 [1]		0	a	9998
0				Bits por solicitud			02	Leer bits de entrada:		

MODE	DATA_ADDR (dirección Modbus)			DATA_LEN (longitud de los datos)			Código de función Modbus	Operación y datos		
	10001	a	19999	1	a	2000/1992 [1]		0	a	9998
0				Palabras por solicitud			03	Leer registro de paradas:		
	40001	a	49999	1	a	125/124 [1]		0	a	9998
	400001	a	465535	1	a	125/124 [1]		0	a	65534
0				Palabras por solicitud			04	Leer palabras de entrada:		
	30001	a	39999	1	a	125/124 [1]		0	a	9998
1				Bit por solicitud			05	Escribir un bit de salida:		
	1	a	9999	1				0	a	9998
1				1 palabra por solicitud			06	Escribir un registro de paradas:		
	40001	a	49999	1				0	a	9998
	400001	a	465535	1				0	a	65524
1				Bits por solicitud			15	Escribir varios bits de salida:		
	1	a	9999	2	a	1968/1960 [1]		0	a	9998
1				Palabras por solicitud			16	Escribir varios registros de parada:		
	40001	a	49999	2	a	123/122		0	a	9998
	400001	a	465534	2	a	123/122 [1]		0	a	65534
2 [2]				Bits por solicitud			15	Escribir uno o varios bits de salida:		
	1	a	9999	2	a	1968/1960 [1]		0	a	9998
2 [2]				Palabras por solicitud			16	Escribir uno o varios registros de paradas:		
	40001	a	49999	1	a	123		0	a	9998
	400001	a	465535	1	a	122 [1]		0	a	65534
11	Los dos operandos DATA_ADDR y DATA_LEN de Modbus_Master se ignoran en esta función.						11	Leer palabra de estado y contador de eventos de la comunicación del esclavo. La palabra de estado indica la ocupación (0 - no ocupado, 0xFFFF - ocupado). El contador de eventos se incrementa cada vez que se procesa correctamente una trama.		
80				1 palabra por solicitud			08	Comprobar el estado del esclavo mediante el código de diagnóstico de datos 0x0000 (test de bucle de control - el esclavo devuelve un eco de la solicitud)		
	-			1				-		
81				1 palabra por solicitud			08	Inicializar el contador de eventos del esclavo mediante el código de diagnóstico de datos 0x000A		

MODE	DATA_ADDR (dirección Modbus)			DATA_LEN (longitud de los datos)			Código de función Modbus	Operación y datos		
	-			1				-		
3 a 10, 12 a 79, 82 a 255	-			-				Reservado		

[1] En el direccionamiento avanzado (ver parámetro Extended_Adressing), la longitud de datos máxima es 1 byte o 1 palabra menos, según sea el tipo de datos de la función.

[2] MODE 2 permite escribir uno o varios bits de salida y uno o varios registros Holding utilizando las funciones Modbus 15 y 16. MODE 1 utiliza las funciones Modbus 5 y 6 para escribir 1 bit de salida y 1 registro Holding y las funciones Modbus 15 y 16 para escribir varios bits de salida y varios registros Holding.

Parámetro DATA_PTR

El parámetro DATA_PTR indica la dirección DB o M en la que se escribe o de la que se lee. Si se utiliza un bloque de datos, hay que crear un bloque de datos global que ponga a disposición la memoria de datos para procesos de lectura y escritura en esclavos Modbus.

Nota

S7-1500 - El bloque de datos direccionado mediante DATA_PTR debe permitir el direccionamiento directo

El bloque de datos debe permitir el direccionamiento directo (absoluto) y simbólico.

Estructuras de bloques de datos para el parámetro DATA_PTR

- Estos tipos de datos son válidos para **leer palabras** del área de direcciones Modbus (DATA_PTR) 30001 a 39999, 40001 a 49999 y 400001 a 465535, así como para **escribir palabras** del área de direcciones Modbus (parámetro DATA_PTR) 40001 a 49999 y 400001 a 465535.
  - Matriz estándar de los tipos de datos WORD, UINT o INT
  - Estructura denominada del tipo WORD, UINT o INT en la que cada elemento tiene un nombre unívoco y un tipo de datos de 16 bits.
  - Estructura compleja denominada en la que cada elemento tiene un nombre unívoco y un tipo de datos de 16 o 32 bits.
- Para **leer** y escribir bits para el área de direcciones Modbus (parámetro DATA_PTR) 00001 a 09999 y para leer bits de 10001 a 19999.
  - Campo estándar de tipos de datos booleanos.
  - Estructura booleana denominada de variables booleanas con una denominación unívoca.

- Si bien no es necesario que cada instrucción Modbus_Master tenga un área de memoria propia y separada, es recomendable. Esto se debe a que la posibilidad de que se destruyan datos es mayor cuando varias instrucciones Modbus_Master leen y escriben en la misma área de memoria.
- No es necesario que las áreas de memoria de DATA_PTR estén en el mismo bloque de datos global. Es posible crear un bloque de datos con varias áreas para procesos de lectura Modbus, un bloque de datos para procesos de escritura Modbus o un bloque de datos para cada estación esclava.

Consulte también

Modbus_Slave: comunicarse como esclavo Modbus (Página 7865)

Modbus_Slave: comunicarse como esclavo Modbus

Descripción

La instrucción Modbus_Slave permite al programa comunicarse como esclavo Modbus a través de un CM (RS422/485 o RS232). STEP 7 crea automáticamente un DB de instancia cuando se inserta la instrucción. El parámetro MB_DB de la instrucción Modbus_Comm_Load debe interconectarse con el parámetro MB_DB (estático) de la instrucción Modbus_Slave .

Parámetros

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
MB_ADDR	IN	UInt	Word	-	Dirección de estación del esclavo Modbus: área de direccionamiento estándar (1 a 247) área de direccionamiento ampliada (0 a 65535) Nota: 0 es la dirección Broadcast
COM_RST	IN/OUT	---	Bool	FALSE	Inicialización de la instrucción Modbus_Slave Con TRUE se inicializa la instrucción. Seguidamente, la instrucción ajusta COM_RST de nuevo a FALSE.
MB_HOLD_REG	IN/OUT	Variant	Any	-	Puntero hacia el DB de registro de paradas Modbus: el registro de paradas Modbus puede ser el área de memoria de las marcas o bien un bloque de datos.
NDR	OUT	Bool		FALSE	Datos nuevos preparados: • FALSE – No hay datos nuevos • TRUE – Indica que se han escrito datos nuevos del maestro Modbus El bit NDR es TRUE durante un ciclo después de que la última solicitud haya finalizado sin errores.

Parámetro	Declaración	Tipo de datos		Estándar	Descripción
		S7-1500	S7-300/400/ WinAC		
DR	OUT	Bool		FALSE	Leer datos: • FALSE – No se han leído datos • TRUE – Indica que la instrucción del maestro Modbus ha guardado datos recibidos en la zona de destino. El bit DR es TRUE durante un ciclo después de que la última solicitud haya finalizado sin errores.
ERROR	OUT	Bool		FALSE	El bit ERROR es TRUE durante un ciclo después de que la última solicitud haya finalizado con errores. Si la ejecución finaliza con un error, el código de error del parámetro STATUS solo es válido en el ciclo en que ERROR = TRUE.
STATUS	OUT	Word		0	Código de error (consulte Mensajes de error (Página 7876))

Los códigos de función de la comunicación Modbus (1, 2, 4, 5 y 15) pueden leer y escribir bits y palabras directamente en la memoria imagen de proceso de las entradas y en la memoria imagen de proceso de las salidas de la CPU. En estos códigos de función debe definirse el parámetro MB_HOLD_REG como un tipo de datos mayor que un byte. La tabla siguiente muestra la asignación de ejemplo de las direcciones Modbus a la memoria imagen de proceso de la CPU.

Tabla 9-307 Asignación de las direcciones Modbus a la memoria imagen de proceso

Funciones Modbus						S7-1500/S7-300/S7-400			
Código de función	Función	Área de datos	Área de direcciones			Área de datos	Dirección de la CPU		
01	Leer bits	Salida	0	a	9998	Memoria imagen de proceso de las salidas	Q0.0	a	Q1248.6
02	Leer bits	Entrada	0	a	9998	Memoria imagen de proceso de las entradas	I0.0	a	I1248.6
04	Leer palabras	Entrada	0	a	9998	Memoria imagen de proceso de las entradas	IW0	a	IW19996
05	Escribir bit	Salida	0	a	9998	Memoria imagen de proceso de las salidas	Q0.0	a	Q1248.6
15	Escribir bits	Salida	0	a	9998	Memoria imagen de proceso de las salidas	Q0.0	a	Q1248.6

Nota

El área de direcciones disponible puede ser también más pequeña, en función de la configuración de memoria de la CPU.

Los códigos de función de la comunicación Modbus (3, 6, 16) utilizan un registro de paradas Modbus que puede ser un área de direcciones del área de memoria de las marcas o bien un

bloque de datos. El tipo del registro de paradas se indica en el parámetro MB_HOLD_REG de la instrucción Modbus_Slave.

Nota

S7-1500 - Tipo del bloque de datos MB_HOLD_REG

Un bloque de datos con registro de paradas Modbus debe permitir el direccionamiento directo (absoluto) y simbólico.

Variables del esclavo Modbus

Esta tabla muestra las variables estáticas públicas del bloque de datos de instancia de Modbus_Slave que pueden utilizarse en el programa.

Tabla 9-308 Variables del esclavo Modbus

Variable	Tipo de datos	Estándar	Descripción
HR_Start_Offset	Word	0	Indica la dirección inicial del registro de paradas Modbus (estándar = 0)
Extended_Addressing	Bool	FALSE	Direccionamiento avanzado, configura el direccionamiento del esclavo como byte individual o doble (FALSE = dirección de byte único, TRUE = dirección de byte doble)
Request_Count	Word	0	Número de todas las solicitudes recibidas por este esclavo
Slave_Message_Count	Word	0	Número de las solicitudes recibidas para este esclavo específico
Bad_CRC_Count	Word	0	Número de solicitudes recibidas que presentan un error CRC
Broadcast_Count	Word	0	Número de solicitudes Broadcast recibidas
Exception_Count	Word	0	Errores específicos de Modbus, que se acusan en el maestro con una excepción
Success_Count	Word	0	Número de las solicitudes recibidas para este esclavo específico sin errores de protocolo
MB_DB	MB_BASE	-	El parámetro MB_DB de la instrucción Modbus_Comm_Load debe conectarse con el parámetro MB_DB de la instrucción Modbus_Master.

El programa puede escribir valores en las variables HR_Start_Offset y Extended_Addressing y controlar las operaciones del esclavo Modbus. Las demás variables pueden leerse para vigilar el estado de Modbus.

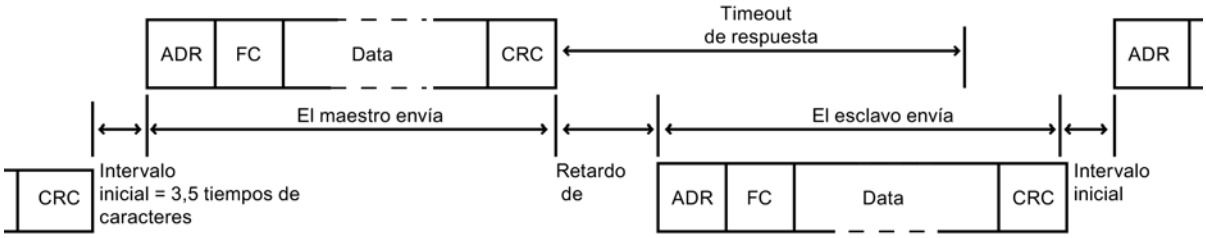
Reglas para la comunicación del esclavo Modbus

- Modbus_Comm_Load debe ejecutarse para configurar un puerto con el fin de que la instrucción Modbus_Slave pueda comunicarse a través de dicho puerto.
- Si un puerto debe reaccionar como esclavo a un maestro Modbus, dicho puerto no debe programarse con la instrucción Modbus_Master .
- Solo es posible utilizar una instancia de Modbus_Slave con un puerto concreto, de lo contrario pueden producirse comportamientos imprevistos.

- Las instrucciones Modbus no utilizan eventos de alarma de la comunicación para controlar el proceso de comunicación. El programa debe controlar el proceso de comunicación consultando la instrucción Modbus_Slave respecto de procesos de transmisión y recepción terminados.
- La instrucción Modbus_Slave debe ejecutarse regularmente y con una frecuencia que permita una respuesta temprana a solicitudes entrantes de un maestro Modbus. Se recomienda ejecutar Modbus_Slave en cada ciclo desde un OB de ciclo de programa. Modbus_Slave puede ejecutarse desde un OB de alarma cíclica, aunque no es recomendable, porque unos retardos de tiempo excesivos en el programa de interrupción pueden bloquear temporalmente la ejecución de otros programas de interrupción.

Control de tiempo de la señal Modbus

Modbus_Slave debe ejecutarse regularmente para recibir todas las solicitudes del maestro Modbus y responder de un modo adecuado. La frecuencia de ejecución de Modbus_Slave se rige por el valor de timeout para la respuesta especificado por el maestro Modbus. Esto se representa en la figura siguiente.



El período de timeout de la respuesta (RESP_TO) es el tiempo que espera un maestro Modbus al principio de una respuesta de un esclavo Modbus. Dicho período no se define en el protocolo Modbus, sino en un parámetro de la instrucción Modbus_Comm_Load. Puesto que tanto la recepción como la transmisión de una trama requieren varias llamadas (tres como mínimo) de la instrucción Modbus_Slave, debería ejecutarse Modbus_Slave un mínimo de doce veces durante el período de timeout para la respuesta del maestro Modbus para que la recepción y transmisión de datos por parte del esclavo Modbus se ejecute el doble de veces de las especificadas por el período de timeout.

HR_Start_Offset

Las direcciones del registro de paradas Modbus empiezan en 40001 o 400001. Estas direcciones equivalen a la dirección inicial del registro de paradas en la memoria del sistema de destino. De todas formas, es posible configurar la variable HR_Start_Offset para configurar una dirección inicial distinta de 40001 o 400001 para el registro de paradas Modbus.

La dirección 0 en el telegrama de recepción corresponde a la dirección inicial del registro de paradas en la memoria del sistema de destino. A través de la variable HR_Start_Offset se puede configurar otra dirección inicial como 0 para el registro de paradas de Modbus.

Así, por ejemplo, existe la posibilidad de configurar un registro de paradas que empiece en MW100 y una longitud de 100 palabras. Con HR_Start_Offset = 20, la dirección 20 del telegrama de recepción corresponde a la dirección inicial del registro de paradas en la memoria de destino (MW100). Toda dirección del telegrama de recepción por debajo de 20 y por encima de 119 provoca un error de direccionamiento.

Tabla 9-309 Ejemplo de direccionamiento del registro de paradas de Modbus si DATA_PTR señala a MW100 con una longitud de 100 palabras

HR_Start_Offset	Dirección	Mínimo	Máximo
0	Dirección Modbus (palabra)	0	99
	Dirección S7-1500	MW100	MW298
20	Dirección Modbus (palabra)	20	119
	Dirección S7-1500	MW100	MW298

HR_Start_Offset es un valor de palabra que indica la dirección inicial del registro de paradas Modbus y está guardado en el bloque de datos de instancia Modbus_Slave . Esta variable estática pública se selecciona mediante la lista desplegable de parámetros después de haber insertado Modbus_Slave en el programa.

Si, por ejemplo, se ha insertado Modbus_Slave en un segmento KOP, es posible pasar a un segmento anterior y asignar el valor HR_Start_Offset con el comando Move. El valor debe asignarse antes de ejecutar Modbus_Slave.

Introducir una variable de esclavo Modbus mediante el nombre de DB estándar:

1. Coloque el cursor en el campo de parámetros OUT1 e introduzca el carácter m.
2. Seleccione en la lista desplegable el DB de instancia que desee para la instrucción Modbus_Slave.
3. Coloque el cursor a la derecha del nombre de DB (después de las comillas) e introduzca un punto.
4. Seleccione en la lista desplegable "Modbus_Slave_DB.HR_Start_Offset".

Estructura de la trama

Extended_Addressing

A la variable Extended_Addressing se accede de un modo similar que a la referencia HR_Start_Offset, excepto por el hecho de que la variable Extended_Addressing es un valor booleano.

Para el direccionamiento de esclavo Modbus puede configurarse un byte individual (el estándar de Modbus) con Extended_Adressing = FALSE, o bien dos bytes (Extended_Adressing = TRUE). El direccionamiento avanzado sirve para direccionar más de 247 dispositivos en una sola red. Con Extended_Adressing = TRUE es posible activar un máximo de 65535 direcciones. A continuación se muestra una trama Modbus a modo de ejemplo.

Tabla 9-310 Dirección del esclavo con un byte (byte 0)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	
Solicitud	Dirección del esclavo	Código de función	Dirección inicial		Datos		
Respuesta válida	Dirección del esclavo	Código de función	Longitud	Datos...			
Mensaje de error	Dirección del esclavo	0xxx	Código de excepción				

Tabla 9-311 Dirección del esclavo con dos bytes (byte 0 y byte 1)

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Solicitud	Dirección del esclavo		Código de función	Dirección inicial		Datos	
Respuesta válida	Dirección del esclavo		Código de función	Longitud	Datos...		
Mensaje de error	Dirección del esclavo		0xxx	Código de excepción			

Descripción de las tramas

El intercambio de datos entre maestro y esclavo comienza con la dirección del esclavo, seguida del código de función. A continuación se transfieren los datos. La estructura del campo de datos depende del código de función utilizado. Al final de la trama se transmitirá la suma de comprobación (CRC).

Código de función 1 - Esta función permite leer bits de salida individuales

Tabla 9-312 FC 1 - Leer bits de salida

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 1	Dirección inicial		Número de salidas	
Respuesta válida	Dirección del esclavo	Código de función 1	Longitud [1]	Datos de salida [3)]		
Mensaje de error	Dirección del esclavo	0x81	Código de excepción [2]	---		

[1] Longitud: si de la división de las salidas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

[3] Los datos de salida pueden abarcar varios bytes

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 1	Dirección inicial		Número de salidas	
Respuesta válida	Dirección del esclavo		Código de función 1	Longitud [1]	Datos de salida		
Mensaje de error	Dirección del esclavo		0x81	Código de excepción [2]	---		

[1] Longitud: si de la división de las salidas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

[3] Los datos de salida pueden abarcar varios bytes

Código de función 2 - Esta función permite leer bits de entrada individuales

Tabla 9-313 FC 2 - Leer bits de entrada

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 2	Dirección inicial		Número de entradas	
Respuesta válida	Dirección del esclavo	Código de función 2	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo	0x82	Código de excepción [2]	---		

[1] Longitud: si de la división de las entradas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 2	Dirección inicial		Número de entradas	
Respuesta válida	Dirección del esclavo		Código de función 2	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo		0x82	Código de excepción [2]	---		

[1] Longitud: si de la división de las entradas entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

Código de función 3 - Esta función permite leer registros individuales

Tabla 9-314 FC 3 - Leer registro de paradas

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 3	Dirección inicial		Número de registros	
Respuesta válida	Dirección del esclavo	Código de función 3	Longitud [1]	Datos de registro		
Mensaje de error	Dirección del esclavo	0x83	Código de excepción [2]	---		

[1] Longitud: Número de bytes

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 3	Dirección inicial		Número de registros	
Respuesta válida	Dirección del esclavo		Código de función 3	Longitud [1]	Datos de registro		
Mensaje de error	Dirección del esclavo		0x83	Código de excepción [2]	---		

[1] Longitud: Número de bytes

[2] Código E: 01, 02, 03 o 04

Código de función 4 - Esta función permite leer registros individuales

Tabla 9-315 FC 4 - Leer palabras de entrada

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 4	Dirección inicial		Número de palabras de entrada	
Respuesta válida	Dirección del esclavo	Código de función 4	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo	0x84	Código de excepción [2]	---		

[1] Longitud: 2 * número de palabras de entrada

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 4	Dirección inicial		Número de palabras de entrada	
Respuesta válida	Dirección del esclavo		Código de función 4	Longitud [1]	Datos de entrada		
Mensaje de error	Dirección del esclavo		0x84	Código de excepción [2]	---		

[1] Longitud: 2 * número de palabras de entrada

[2] Código E: 01, 02, 03 o 04

Código de función 5- Con esta función puede activarse o borrarse un bit individual

Tabla 9-316 FC 5 - Escribir un bit de salida

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 5	Dirección inicial		Valor	
Respuesta válida	Dirección del esclavo	Código de función 5	Longitud	Valor		
Mensaje de error	Dirección del esclavo	0x85	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 5	Dirección inicial		Valor	
Respuesta válida	Dirección del esclavo		Código de función 5	Longitud	Valor		
Mensaje de error	Dirección del esclavo		0x85	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

Código de función 6 - Esta función permite escribir registros individuales

Tabla 9-317 FC 6 - Escribir registro de paradas

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 6	Dirección		Registros	
Respuesta válida	Dirección del esclavo	Código de función 6	Dirección		Registros	
Mensaje de error	Dirección del esclavo	0x86	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 6	Dirección		Registros	
Respuesta válida	Dirección del esclavo		Código de función 6	Dirección		Registros	
Mensaje de error	Dirección del esclavo		0x86	Código de excepción [1]	---		

[1] Código E: 01, 02, 03 o 04

Código de función 8 - Esta función permite comprobar la conexión de comunicación

Tabla 9-318 FC 8 - Estado del esclavo

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 8	Código de diagnóstico		Valor de prueba	
Respuesta válida	Dirección del esclavo	Código de función 8	Código de diagnóstico		Valor de prueba	
Mensaje de error	Dirección del esclavo	0x88	Código de excepción [1]	---		

[1] Código E: 01, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 8	Código de diagnóstico		Valor de prueba	
Respuesta válida	Dirección del esclavo		Código de función 8	Código de diagnóstico		Valor de prueba	
Mensaje de error	Dirección del esclavo		0x88	Código de excepción [1]	---		

[1] Código E: 01, 03 o 04

Código de función 11 - Esta función permite leer una "Status-Word" de 2 bytes y un "Event-Counter" de 2 bytes

Tabla 9-319 FC 11 - Contador de eventos de la comunicación del esclavo

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Petición	Dirección del esclavo	Código de función 11	---			
Respuesta válida	Dirección del esclavo	Código de función 11	Status		Contador de eventos	
Mensaje de error	Dirección del esclavo	0x8B	Código de excepción [1]	---		

[1] Código E: 01 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6
Petición	Dirección del esclavo		Código de función 11	---			
Respuesta válida	Dirección del esclavo		Código de función 11	Status		Contador de eventos	
Mensaje de error	Dirección del esclavo		0x8B	Código de excepción [1]	---		

[1] Código E: 01 o 04

Código de función 15 - Esta función permite escribir varios bits

Tabla 9-320 FC 15 - Escribir uno o varios bits de salida

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Petición	Dirección del esclavo	Código de función 15	Dirección inicial		Número de palabras de salida		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo	Código de función 15	Dirección inicial		Número de palabras de salida		---		
Mensaje de error	Dirección del esclavo	0x8F	Código de excepción [2]	---					

[1] Contador de bytes: si de la división del número de bytes entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Petición	Dirección del esclavo		Código de función 15	Dirección inicial		Número de palabras de salida		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo		Código de función 15	Dirección inicial		Número de palabras de salida		---		
Mensaje de error	Dirección del esclavo		0x8F	Código de excepción [2]		---				

[1] Contador de bytes: si de la división del número de bytes entre 8 se obtiene un residuo, el número de bytes debe incrementarse en 1.

[2] Código E: 01, 02, 03 o 04

Código de función 16 - Esta función permite escribir uno o varios registros

Tabla 9-321 FC 16 - Escribir uno o varios registros de paradas

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte n
Petición	Dirección del esclavo	Código de función 16	Dirección inicial		Número de registros		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo	Código de función 16	Dirección inicial		Número de registros		---		
Mensaje de error	Dirección del esclavo	0x90	Código de excepción [2]	---					

[1] Contador de bytes: Número de registros * 2

[2] Código E: 01, 02, 03 o 04

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte n
Petición	Dirección del esclavo		Código de función 16	Dirección inicial		Número de registros		Contador de bytes [1]	Valor	
Respuesta válida	Dirección del esclavo		Código de función 16	Dirección inicial		Número de registros		---		
Mensaje de error	Dirección del esclavo		0x90	Código de excepción [2]		---				

[1] Contador de bytes: Número de registros * 2

[2] Código E: 01, 02, 03 o 04

Mensajes de error

Sinopsis de los mensajes de error - PtP

Los mensajes de error se ponen a disposición en la salida STATUS de una instrucción y allí pueden ser evaluados o procesados en el programa de usuario.

Código de error	Descripción	Solución
16#0000	No hay error	-
Estado de RECEIVE y códigos de error		
16#0094	Fin de trama detectado por "recepción de una longitud fija/máxima de trama"	-
16#0095	Fin de trama detectado por "tiempo de mensaje excedido"	-
16#0096	Fin de trama detectado por haber transcurrido el "tiempo de retardo de caracteres"	-
16#0097	La trama se ha cancelado porque se ha alcanzado el tiempo de respuesta máximo.	-
16#0098	Fin de trama detectado por cumplirse las condiciones de "leer longitud del mensaje en el mensaje"	-
16#0099	Fin de trama detectado por recepción de la "secuencia de fin"	-
Estado de SEND y códigos de error		
16#7000	Bloque en vacío	-
16#7001	Primera llamada para una trama nueva: transferencia de datos lanzada	-
16#7002	Llamada intermedia: transferencia de datos en curso	-
16#8085	Indicación de longitud no válida	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)

Código de error	Descripción	Solución
16#8088	Indicación de longitud mayor que el área ajustada en el búfer de recepción. Nota: En el caso de que en el parámetro BUFFER se haya especificado el tipo de datos STRING, también aparece este código de error si la longitud de String actual es inferior al valor indicado en el parámetro LENGTH.	Modifique el área en el búfer de recepción o seleccione una longitud de trama adecuada al área ajustada en el búfer. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Estado de RECEIVE y códigos de error		
16#8088	El número de caracteres recibidos es superior al especificado en el parámetro BUFFER.	Seleccione una longitud de trama adecuada. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte)
16#8090	Error de configuración: número impar de bytes como WString	Elija un número par de bytes.
Códigos de mensajes de error de las funciones especiales		
16#818F	Ajuste erróneo del número de parámetro (solo en USS)	Seleccione un número de parámetro adecuado (PARAM). Se admiten: 0-2047
16#8190	Ajuste erróneo del cálculo CRC	Seleccione un valor adecuado para el cálculo CRC. Se admiten: desactivado o activado. Compruebe si el módulo activado soporta el cálculo CRC.
16#8191	Ajuste erróneo de la alarma de diagnóstico	Seleccione un valor adecuado para "Alarma de diagnóstico". Se admiten: alarma de diagnóstico desactivada o alarma de diagnóstico activada. Compruebe si el módulo direccionado soporta la generación de alarmas de diagnóstico.
Códigos de mensajes de error de la "Configuración del puerto"		
16#81A0	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo (PROTOCOL).
16#81A1	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo (BAUD).
16#81A2	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" (PARITY). Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)

Código de error	Descripción	Solución
16#81A3	El módulo no soporta este número de bits de datos.	Seleccione un valor adecuado para "Número de bits de datos" (DATABITS). Se admiten: • 7 (2) • 8 (1)
16#81A4	El módulo no soporta este número de bits de parada.	Seleccione un valor adecuado para "Número de bits de parada" (STOPBITS). Se admiten: • 1 (1) • 2 (2)
16#81A5	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo (FLOWCTRL).
16#81A7	Valor no admisible para XON o XOFF	Seleccione valores adecuados para XON (XONCHAR) y XOFF (XOFFCHAR). Rango admisible: 0...255
16#81AA	Modo de operación no admisible	Los modos de operación admitidos son: • Dúplex (RS232) (0) • Dúplex (RS422) modo a cuatro hilos (punto a punto) (1)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (maestro multipunto) (2)/(CM PtP (ET 200SP)) • Dúplex (RS422) modo a cuatro hilos (esclavo multipunto) (3) • Semidúplex (RS485) modo a dos hilos (4)
16#81AB	Estado inicial no admisible de la línea de recepción	Los estados iniciales admitidos son: • "Sin" estado inicial (0) • Señal R(A) = 5 V, señal R(B) = 0 V (detección Break) (1): Solo disponible con: "Dúplex (RS422) modo a cuatro hilos (acoplamiento punto a punto)" y "Dúplex (RS422) modo a cuatro hilos (esclavo multipunto)". • Señal R(A) = 0 V, señal R(B) = 5 V (2): este estado inicial corresponde al estado de reposo (ninguna transmisión activa).
16#81AC	Valor no admisible para "Detección Break"	Seleccione un valor adecuado para "Detección Break". Se admiten: • Detección Break desactivada (0) • Detección Break activada (1)
16#81AF	El módulo no soporta este protocolo.	Seleccione un protocolo compatible con el módulo.
Códigos de error de la "Configuración de transmisión"		

Código de error	Descripción	Solución
16#81B5	Más de 2 caracteres de fin -o bien- secuencia de fin > 5 caracteres	Seleccione valores adecuados para "Carácter de fin" y "Secuencia de fin". Se admiten: • desactivado (0) • 1 (1) o 2 (2) caracteres de fin o bien: • desactivado (0) • de 1 (1) a 5 (5) caracteres para la secuencia de fin
16#81B6	Send Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de transmisión cuando esté ajustado el protocolo 3964(R).
Códigos de error de la "Configuración de recepción"		
16#81C0	Condición de inicio no admisible	Seleccione una condición de inicio adecuada. Se admiten: • Enviar Break antes de iniciar la trama • Enviar Idle Line.
16#81C1	Condición de fin no admisible o no se ha seleccionado ninguna condición de fin	Seleccione una condición de fin adecuada (consulte Recibir datos con Freeport (Página 1028)).
16#81C3	Valor no admisible para "Longitud máxima del mensaje"	Seleccione un valor adecuado para "Longitud máxima del mensaje" (MAXLEN). Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C4	Valor no admisible para "Offset del campo de longitud en el mensaje"	Seleccione un valor adecuado para "Offset del campo de longitud en el mensaje". Rango admisible (en función del módulo): 1-1024/2048/4096 (Byte)
16#81C5	Valor no admisible para "Tamaño del campo de longitud"	Seleccione un valor adecuado para "Tamaño del campo de longitud" (LENGTHSIZE). Rango admisible en bytes: • 1 (1) • 2 (2) • 4 (4)
16#81C6	Valor no admisible para "Número de caracteres que se excluyen de la longitud"	Seleccione un valor adecuado para "Número de caracteres que se excluyen de la longitud" (LENGTHM). Rango admisible: de 0 a 255 (bytes)
16#81C7	La suma de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres que se excluyen" es mayor que la longitud máxima de la trama	Seleccione valores adecuados para "Offset en el mensaje", "Tamaño del campo de longitud" y "Número de caracteres excluidos". Rango admisible: • Offset en el mensaje (en función del módulo): 0-1024/2048/4096 (Byte) • Tamaño del campo de longitud: 1, 2 o 4 (bytes) • Número de caracteres excluidos: 0-255 (bytes)

Código de error	Descripción	Solución
16#81C8	Valor no admisible para "Tiempo de respuesta excedido"	Seleccione un valor adecuado para "Tiempo de respuesta excedido". Rango admisible: 1-65535(ms)
16#81C9	Valor no admisible para "Tiempo de retardo de caracteres"	Seleccione un valor adecuado para "Tiempo de retardo de caracteres". Rango admisible: 1-65535 (tiempos de bit)
16#81CB	Secuencia de fin de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CC	Secuencia de principio de trama activada, pero ninguno de los caracteres está activado para la comprobación	Active uno o varios caracteres para la comprobación.
16#81CD	Valor no admisible para "Impedir sobrescritura"	Seleccione un valor adecuado para "Impedir sobrescritura". Se admiten: • Impedir sobrescritura desactivado (0) • Impedir sobrescritura activado (1)
16#81CE	Valor no admisible para "Borrar búfer de recepción al arrancar"	Seleccione un valor adecuado para "Borrar búfer de recepción al arrancar". Se admiten: • Borrar búfer de recepción en el arranque desactivado (0) • Borrar búfer de recepción en el arranque activado (1)
Estado de SEND y códigos de error		
16#81D0	Recibir solicitud de transmisión durante el tiempo de ejecución de una petición de transmisión	Asegúrese de que no se solicitan más transmisiones durante el tiempo de ejecución de una petición de transmisión.
16#81D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#81D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#81D3	Desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	Seleccione una longitud de trama menor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
16#81D5	Transferencia cancelada por cambio en la parametrización, detección de rotura de hilo o CPU en STOP	Compruebe la parametrización, rotura de hilo o el estado de la CPU.
16#81D6	Transferencia cancelada porque no se ha recibido el delimitador de fin	Compruebe la parametrización de los caracteres de fin y la trama del interlocutor.
16#81D7	Fallo de comunicación entre programa de usuario y módulo	Compruebe la comunicación (p. ej. coincidencia del número de secuencia).
16#81D8	Intento de transferencia rechazado porque el módulo no está parametrizado	Parametrice el módulo.
Códigos de error de la configuración de recepción		

Código de error	Descripción	Solución
16#81E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario, llame la función para la recepción con mayor frecuencia o parametrice una comunicación con control del flujo de datos.
16#81E1	Trama cancelada: Error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#81E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#81E3	Trama cancelada: error de desbordamiento de caracteres	Error de firmware: póngase en contacto con el Customer Support.
16#81E4	Trama cancelada: la longitud de "Offset en el mensaje + Tamaño del campo de longitud + Número de caracteres excluidos" es mayor que el búfer de recepción	Compruebe los ajustes de Offset en el mensaje, Tamaño del campo de longitud y Número de caracteres excluidos.
16#81E5	Trama cancelada: Break	Se ha interrumpido la línea de recepción del interlocutor. Restablezca la conexión o conecte el interlocutor.
16#81E6	Número máximo de "Tramas de recepción en búfer" excedido	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos, o bien aumente el número de tramas en búfer.
16#81E7	Error de sincronización entre módulo y Receive_P2P	Asegúrese de que no haya diferentes instancias de Receive_P2P accediendo al mismo módulo.
16#81E8	Trama cancelada: tiempo de retardo de caracteres transcurrido antes de detectar el criterio de fin del mensaje	El interlocutor es demasiado lento o está fallando. Si fuese necesario, utilice un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#81E9	Error de Modbus CRC (solo módulos de comunicaciones compatibles con Modbus)	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#81EA	Trama Modbus demasiado corta (solo módulos de comunicaciones compatibles con Modbus)	Rebase por defecto de la longitud mínima de la trama Modbus. Compruebe el interlocutor.
16#81EB	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (Byte) Compruebe los parámetros de detección de fin de trama.
Códigos de error de las señales cualificadoras V24		
16#81F0	El módulo no soporta señales cualificadoras V24	Ha intentado ajustar señales cualificadoras para un módulo que no soporta señales cualificadoras V24. Asegúrese de estar utilizando un módulo RS232 y de que esté ajustado el modo RS232 (ET 200SP).
16#81F1	No es posible manejar las señales cualificadoras V24	Si el control del flujo de datos de hardware está activo no es posible controlar manualmente las señales cualificadoras V24.
Códigos de error de la configuración de recepción		

Código de error	Descripción	Solución
16#8201 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Introduzca un puntero hacia uno de los siguientes tipos de datos: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8225	Receive_Conditions señala al área de memoria optimizada mayor que 1 kbyte o bien Receive_Conditions señala al área de memoria optimizada y la longitud de recepción es mayor que la del área direccionada por Receive_Conditions.	Introduzca un puntero hacia un área con la longitud máxima siguiente: • Área de memoria optimizada: 1 kByte • Área de memoria no optimizada: 4 kByte Nota: Si el puntero señala a un área de memoria optimizada, no envíe más de 1 kByte.
16#8229 [1)]	Receive_Conditions señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.
Códigos de error generales		
16#8280	Acuse negativo al leer el módulo	Encontrará información más detallada sobre la causa del error en los parámetros estáticos RDREC.STATUS, así como en la descripción del SFB RDREC. • Compruebe la entrada en el parámetro PORT • Ajuste el parámetro COM_RST antes de la 1.ª llamada.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT Encontrará información más detallada sobre la causa del error en los parámetros estáticos WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Códigos de error de la configuración de recepción		
16#82C1	Valor no admisible para "Tramas de recepción en búfer".	Seleccione un valor adecuado para "Tramas de recepción en búfer". Rango admisible: 1-255
16#82C2	Receive Configuration rechazado porque está seleccionado el protocolo 3964(R)	Asegúrese de que no se envía ninguna configuración de recepción cuando está ajustado el protocolo 3964(R).
16#8301 [1)]	Receive_Conditions señala a un tipo de datos no permitido	Seleccione un tipo de datos compatible. Se admiten: DB, BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TIME_OF_DAY, TIME, S5TIME, DATE_AND_TIME, STRING
16#8322	Error de longitud de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8324	Error de área al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
16#8328	Error de ajuste al leer un parámetro	Compruebe la entrada en el parámetro Receive_Conditions
Estado de SEND y códigos de error		
16#8328 [1)]	BUFFER señala a BOOL con un número de bits diferente de n * 8	Si utiliza un puntero hacia BOOL, el número de bits debe ser un múltiplo de 8.

Código de error	Descripción	Solución
Códigos de error de la configuración de recepción		
16#8332	Bloque de datos no admisible en el parámetro Receive_Conditions	Compruebe la entrada en el parámetro Receive_Conditions
16#833A	El nombre del bloque de datos del parámetro Receive_Conditions señala un bloque de datos no cargado.	Compruebe la entrada en el parámetro Receive_Conditions
16#8351	Tipo de datos no admisible	Compruebe la entrada en el parámetro Receive_Conditions
16#8352 [1)]	Receive_Conditions no señala un bloque de datos	Compruebe el puntero hacia Receive_Conditions
16#8353 [1)]	Receive_Conditions no señala a una estructura del tipo Receive_Conditions	Compruebe el puntero hacia Receive_Conditions
Códigos de error del protocolo 3964(R)		
16#8380	Error de parametrización: valor no admisible para "Tiempo de retardo de caracteres".	Seleccione un valor adecuado para "Tiempo de retardo de caracteres" (CharacterDelayTime). Rango admisible: 1-65535 (ms)
16#8381	Error de parametrización: valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" (AcknDelayTime). Rango admisible: 1-65535 (ms)
16#8382	Error de parametrización: valor no admisible para "Prioridad".	Seleccione un valor adecuado para "Prioridad" (Priority). Se admiten: • Alta (1) • Baja (0)
16#8383	Error de parametrización: valor no admisible para "Comprobación de bloque"	Seleccione un valor adecuado para "Comprobación de bloque" (BCC). Se admiten: • con comprobación de bloque (1) • sin comprobación de bloque (0)
16#8384	Error de parametrización: valor no admisible para "Intentos de establecer conexión".	Seleccione un valor adecuado para "Intentos de establecer conexión" (BuildupAttempts). Rango admisible: 1-255
16#8385	Error de parametrización: valor no admisible para "Intentos de transmisión".	Seleccione un valor adecuado para "Intentos de transmisión" (RepetitionAttempts). Rango admisible: 1-255
16#8386	Error en tiempo de ejecución: se ha excedido el número de intentos de establecer conexión	Compruebe el cable de la interfaz y los parámetros de transferencia. Compruebe también que la función de recepción esté correctamente parametrizada en el interlocutor.
16#8387	Error en tiempo de ejecución: se ha excedido el número de intentos de transmisión	Compruebe el cable de la interfaz, los parámetros de transferencia y la parametrización del interlocutor.
16#8388	Error en tiempo de ejecución: error en el "Carácter de control de bloque" El valor del carácter de control de bloque determinado internamente no concuerda con el recibido por el interlocutor al final de la conexión.	Compruebe que la conexión no se encuentre seriamente dañada. De ser así, observará de vez en cuando códigos de error. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.

Código de error	Descripción	Solución
16#8389	Error en tiempo de ejecución: se ha recibido un carácter no admisible mientras se esperaba un búfer de recepción libre	Cuando el búfer de recepción está vacío, se responde a la petición de transmisión del interlocutor (STX, 02H) con DLE. Antes no se debe recibir ningún otro carácter (excepto STX de nuevo). Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838A	Error en tiempo de ejecución: error lógico durante la recepción. Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado de la trama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Si fuese necesario, compruebe el comportamiento correcto del dispositivo interlocutor con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838B	Error en tiempo de ejecución: tiempo de retardo de caracteres excedido	El dispositivo interlocutor es demasiado lento o es defectuoso. Si fuese necesario, compruébelo con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838C	Error en tiempo de ejecución: se ha iniciado el tiempo de espera a un búfer de recepción libre	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#838D	Error en tiempo de ejecución: después de NAK, la repetición de la trama no se inicia en un intervalo de 4 s	Compruebe el interlocutor. Si una trama se recibe con errores, el interlocutor debe repetirlo en un intervalo de 4 segundos.
16#838E	Error en tiempo de ejecución: se han recibido uno o más caracteres durante el estado de reposo (excepto NAK y STX).	Compruebe el comportamiento correcto del dispositivo interlocutor si fuese necesario con un dispositivo de comprobación de interfaces que se conecte a la línea de transferencia.
16#838F	Error en tiempo de ejecución: conflicto de inicialización; ambos interlocutores tienen ajustada una prioridad alta	Ajuste una prioridad "baja" en uno de los interlocutores
16#8391	Error de parametrización: se han rechazado los datos de parametrización 3964 porque está ajustado Freeport	Asegúrese de que no se envía ningún dato de parametrización 3964 cuando está ajustado el protocolo Freeport.

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - Modbus

Código de error	Descripción	Solución
16#0000	No hay error	-
Error de configuración de la interfaz - Modbus_Comm_Load		
16#8181	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo en el parámetro BAUD.

Código de error	Descripción	Solución
16#8182	El módulo no soporta este ajuste de paridad.	Seleccione un valor adecuado para "Paridad" en el parámetro PARITY. Se admiten: • Sin (1) • Par (2) • Impar (3) • Mark (4) • Space (5) • Cualquiera (6)
16#8183	El módulo no soporta este tipo de control del flujo de datos.	Seleccione un control del flujo de datos compatible con el módulo en el parámetro FLOW_CTRL.
16#8184	Valor no admisible para "Tiempo de respuesta excedido".	Seleccione un valor adecuado para "Tiempo de respuesta excedido" en el parámetro RESP_TO. Rango admisible: 1-65535 (ms)
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.RDREC.STATUS, Receive_Config.RDREC.STATUS o RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_Config.WRREC.STATUS, Receive_Config.WRREC.STATUS o WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8282	Módulo no disponible	Compruebe la entrada en el parámetro PORT y asegúrese de que el módulo está accesible.
Error de configuración - Modbus_Slave		
16#8186	Dirección del esclavo no admisible	Seleccione una dirección del esclavo adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8187	Valor no admisible en el parámetro MB_HOLD_REG	Seleccione un valor adecuado para el Holding Register en el parámetro MB_HOLD_REG.
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#818C	El puntero hacia un área MB_HOLD_REG debe ser un bloque de datos o un área de marcas.	Seleccione un valor adecuado para el puntero hacia el área MB_HOLD_REG.

Código de error	Descripción	Solución
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS o Receive_P2P.WRREC.STATUS, así como en la descripción del SFB WRREC.
16#8452 [1)]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1)]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1)]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1)]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1)]	Error al ejecutar la instrucción. La causa del error figura en el parámetro estático STATUS.	Determine el valor del parámetro SFCSTATUS. Lea su significado en la descripción del SFC51, parámetro STATUS.
Error de configuración - Modbus_Master		
16#8180	Valor no admisible para el parámetro MB_DB	El valor parametrizado para MB_DB (DB de datos de instancia) en la instrucción Modbus_Comm_Load no es admisible. Compruebe la interconexión de la instrucción Modbus_Comm_Load y sus mensajes de error.
16#8186	Dirección de estación no admisible	Seleccione una dirección de estación adecuada en el parámetro MB_ADDR. Se admiten: 1-247 con área de direcciones estándar; 1-65535 con área de direcciones ampliada (0 está reservado para Broadcast)
16#8188	Modo de operación no admisible o Broadcast (MB_ADDR = 0) y parámetro MODE ≠ 1	En el modo de operación Broadcast, seleccione el valor 1 para MODE o elija un modo de operación distinto.
16#8189	Dirección de datos no admisible	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818A	Indicación de longitud no válida	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información

Código de error	Descripción	Solución
16#818B	Valor no admisible para DATA_PTR	Seleccione un valor adecuado para el puntero de datos (dirección M o DB) en el parámetro DATA_PTR. Consulte la descripción Modbus_Master (Página 7859) en el sistema de información
16#818C	Error de interconexión del parámetro DATA_PTR	Compruebe la interconexión de la instrucción.
16#818D	La longitud del área direccionada con DATA_PTR supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero DATA_PTR
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Send_P2P.WRREC.STATUS, Receive_P2P.WRREC.STATUS o Receive_Reset, así como en la descripción del SFB WRREC.
Error de comunicación - Modbus_Master y Modbus_Slave		
16#80D1	El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor falla, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
16#80D2	"Hardware RTS siempre ON": petición de transmisión cancelada por cambio de DSR = ON a OFF	Compruebe el interlocutor. Asegúrese de que DSR es ON durante toda la transferencia.
16#80E0	Trama cancelada: desbordamiento del búfer de transmisión / trama de transmisión demasiado grande	En el programa de usuario llame la instrucción con más frecuencia o parametrice una comunicación con control del flujo de datos.
16#80E1	Trama cancelada: error de paridad	Compruebe la línea de conexión de los interlocutores y si la velocidad de transmisión, la paridad y el número de bits de parada tienen la misma configuración en ambos dispositivos.
16#80E2	Trama cancelada: error de trama	Compruebe los ajustes de bit de inicio, bits de datos, bit de paridad, velocidad de transferencia de datos y bit(s) de parada.
16#80E3	Trama cancelada: error de desbordamiento de caracteres	Compruebe el número de datos en la trama del interlocutor.
16#80E4	Trama cancelada: longitud máxima de trama alcanzada	Seleccione una longitud de trama menor en el interlocutor. Se admiten (en función del módulo): 1-1024/2048/4096 (bytes)
Error de comunicación - Modbus_Master		
16#80C8	El esclavo no responde dentro del tiempo ajustado	Compruebe la velocidad de transferencia de datos, la paridad y el cableado del esclavo.

Código de error	Descripción	Solución
16#8200	La interfaz está ocupada por una petición en curso.	Repita la petición más tarde. Asegúrese de que no se esté ejecutando ninguna petición antes de iniciar una nueva.
Error de protocolo - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	El código de función no se soporta o no se soporta para Broadcast.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Indicación de longitud no válida en la trama de solicitud	Seleccione una longitud de datos adecuada en el parámetro DATA_LEN.
16#8383	Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Error de valor de datos no válido en la trama de solicitud	Compruebe el valor de datos en la trama de solicitud del maestro Modbus
16#8385	El esclavo Modbus no soporta el valor de diagnóstico (código de función 08)	El esclavo Modbus solo soporta los valores de diagnóstico 16#0000 y 16#000A.
Error de protocolo - Modbus_Master (solo módulos de comunicaciones compatibles con Modbus)		
16#8380	Error CRC	Error de suma de verificación de la trama Modbus. Compruebe el interlocutor.
16#8381	Trama de respuesta de Modbus Slave con mensaje de error: El código de función no se soporta.	Compruebe el interlocutor y asegúrese de que se envía un código de función válido.
16#8382	Trama de respuesta de Modbus Slave con mensaje de error: Indicación de longitud no válida	Seleccione una longitud de datos adecuada.
16#8383	Trama de respuesta de Modbus Slave con mensaje de error: Dirección de datos no válida en la trama de solicitud	Seleccione un valor adecuado para la dirección de datos en el parámetro DATA_ADDR.
16#8384	Trama de respuesta de Modbus Slave con mensaje de error: Error de valor de datos	Compruebe la trama de solicitud en el esclavo Modbus.
16#8385	Trama de respuesta de Modbus Slave con mensaje de error: El esclavo Modbus no soporta el valor de diagnóstico	Los esclavos Modbus solo soportan los valores de diagnóstico 16#0000 y 16#000A.
16#8386	El código de función devuelto no concuerda con el código de función solicitado.	Compruebe la trama de respuesta y el direccionamiento del esclavo.
16#8387	Un esclavo no consultado responde	Compruebe la trama de respuesta del esclavo. Compruebe los ajustes de direcciones de los esclavos.
16#8388	Error en la respuesta del esclavo a una petición de escritura.	Compruebe la trama de respuesta del esclavo.
16#8828 [1)]	DATA_PTR señala a una dirección de bit diferente de n * 8	Compruebe el puntero DATA_PTR
16#8852 [1)]	DATA_PTR no señala a ningún DB ni área de marcas	Compruebe el puntero DATA_PTR
16#8853 [1)]	DATA_PTR no es un puntero del tipo BOOL o WORD	Compruebe el puntero DATA_PTR
16#8855 [1)]	DATA_PTR señala a un DB protegido contra escritura	Compruebe el puntero DATA_PTR
16#8856 [1)]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Master
Error - Modbus_Slave (solo módulos de comunicaciones compatibles con Modbus)		
16#8428 [1)]	MB_HOLD_REG señala a una dirección de bit diferente de n * 8	Compruebe el puntero MB_HOLD_REG

Código de error	Descripción	Solución
16#8452 [1]	MB_HOLD_REG no señala a ningún DB ni área de marcas	Compruebe el puntero MB_HOLD_REG
16#8453 [1]	MB_HOLD_REG no es un puntero del tipo BOOL o WORD	Compruebe el puntero MB_HOLD_REG
16#8454 [1]	La longitud del área direccionada con MB_HOLD_REG supera la longitud del DB o el área direccionada es demasiado pequeña para el número de datos que deben leerse o escribirse.	Compruebe el puntero MB_HOLD_REG
16#8455 [1]	MB_HOLD_REG señala a un DB protegido contra escritura	Compruebe el puntero MB_HOLD_REG
16#8456 [1]	Error al llamar el SFC51	Repita la llamada de la instrucción Modbus_Slave

[1] Solo en instrucciones para CPU S7-300/400

Sinopsis de los mensajes de error - USS

Código de error	Descripción	Solución
16#0000	No hay error	-
16#8180	Error de longitud en la respuesta del accionamiento	Compruebe la trama de respuesta del accionamiento.
16#8181	Error del tipo de datos	Seleccione un tipo de datos adecuado. Se admiten: • Real • Palabra • Palabra doble
16#8182	Error del tipo de datos: no se permite devolver "palabra doble" o "real" a la solicitud de "palabra".	Compruebe la trama de respuesta del accionamiento.
16#8183	Error del tipo de datos: no se permite devolver "palabra" a la solicitud de "palabra doble" o "real".	Compruebe la trama de respuesta del accionamiento.
16#8184	Error de suma de verificación en la respuesta del accionamiento	Compruebe el accionamiento y la conexión de comunicación.
16#8185	Error de direccionamiento	Rango de direcciones admisible del accionamiento: de 1 a 16
16#8186	Error de consigna	Rango de consigna admisible: -200 % a +200 %
16#8187	Devolución de número de accionamiento erróneo	Compruebe la trama de respuesta del accionamiento.
16#8188	Longitud de PZD no admisible	Longitudes de PZD admisibles: 2, 4, 6 u 8 palabras
16#8189	El módulo no soporta esta velocidad de transferencia de datos.	Seleccione una velocidad de transferencia de datos compatible con el módulo.
16#818A	Para este accionamiento está activa otra petición.	Repita la petición de lectura o de escritura de parámetros más tarde.
16#818B	El accionamiento no responde.	Compruebe el accionamiento.
16#818C	El accionamiento responde a una consulta de parámetros con un mensaje de error.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.
16#818D	El accionamiento responde a una consulta de parámetros con un error de acceso.	Compruebe la trama de respuesta del accionamiento. Compruebe la consulta de parámetros.

Código de error	Descripción	Solución
16#818E	El accionamiento no se ha inicializado.	Compruebe el programa de usuario y asegúrese de que se llama la instrucción USS_Drive_Control para este accionamiento.
16#8280	Acuse negativo al leer el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.RDREC.STATUS, Send_Config.RDREC.STATUS , Receive_Config.RDREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB RDREC.
16#8281	Acuse negativo al escribir en el módulo	Compruebe la entrada en el parámetro PORT. Encontrará información más detallada sobre la causa del error en los parámetros estáticos Port_Config.WRREC.STATUS, Send_Config.WRREC.STATUS , Receive_Config.WRREC.STATUS , Send_P2P.RDREC.STATUS o Receive_P2P.RDREC.STATUS, así como en la descripción del SFB WRREC.

[1] Solo en instrucciones para CPU S7-300/400

Acoplamiento PtP CP 340

P_SEND: Enviar datos

Descripción

La instrucción P_SEND transmite un conjunto de datos de un bloque de datos especificado por los parámetros DB_NO, DBB_NO y LEN, al CP 340. La instrucción P_SEND se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción P_SEND puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al CP 340 y la instrucción P_SEND pasa al estado inicial. Los datos que ya ha recibido el CP 340 se envían al interlocutor. Si la entrada R tiene el estado lógico estático "1", la transmisión está desactivada.

Con el parámetro LADDR se indica la dirección del CP 340 que debe llamarse.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en

Parámetro STATUS (Página 7904). Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_SEND. Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del CP 340 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º de DB emisor: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8190 datos de transmisión a partir del byte de datos
LEN	INPUT	INT	Longitud de datos 1 ≤ LEN ≤ 1024, indicados en número de bytes
DONE[1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores Parámetro STATUS (Página 7904) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el Parámetro STATUS (Página 7904) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

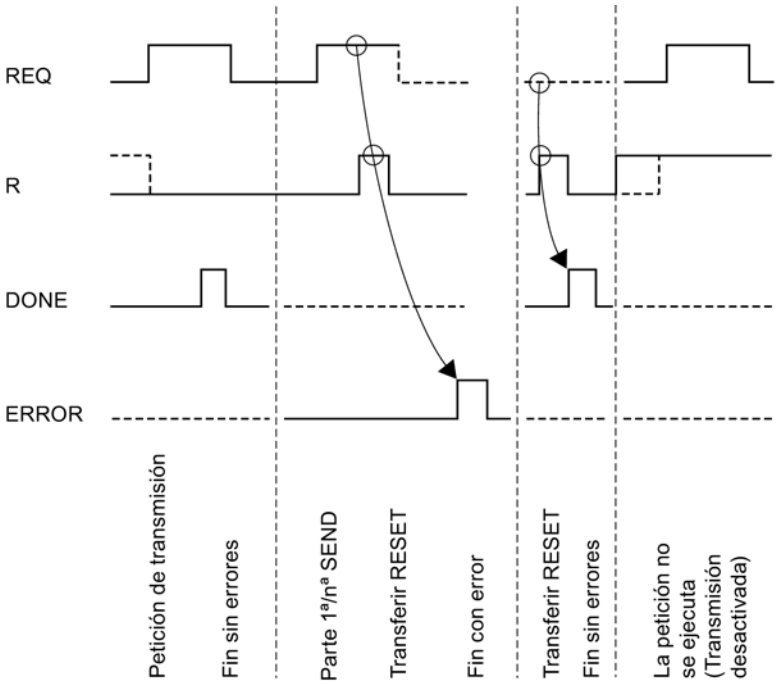
La instrucción P_SEND opera conjuntamente con un DB instancia I_SEND. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0F, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ . El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción P_SEND no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP .

Nota

Antes de que el CP 340 pueda procesar una petición iniciada después de una transición de estado de la CPU de STOP a RUN , debe haber concluido el mecanismo de arranque CP–CPU de la instrucción P_SEND. Una petición iniciada entretanto no se pierde. Se transfiere al CP 340 una vez concluida la coordinación de arranque.

P_RCV: Recibir datos

Descripción

La instrucción P_RCV transmite datos del CP 340 a un área de datos S7, especificada por los parámetros DB_NO, DBB_NO y LEN. La instrucción P_RCV se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del CP 340. Es posible cancelar una transmisión en curso con el estado lógico "0" en el parámetro EN_R. La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desactivada mientras exista el estado lógico "0" en el parámetro EN_R. Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción P_RCV pasa al estado inicial. La recepción está desactivada mientras exista el estado lógico "1" en el parámetro R. Cuando el estado de señal vuelve a ser "0", el telegrama interrumpido se vuelve a recibir desde el principio.

Con el parámetro LADDR se selecciona el CP 340 que se llamará.

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS (Página 7904). Si no se produce ningún error, STATUS asume el valor 0. NDR y ERROR/STATUS también se visualizan en caso de RESET del P_RCV (parámetro LEN == 16#00). Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitar lectura de datos
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Bloqueo de recepción.
LADDR	INPUT	INT	Dirección inicial del CP 340 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º del DB de recepción: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8190 datos de recepción a partir del byte de datos

Parámetro	Declaración	Tipo de datos	Descripción
NDR[1]	OUTPUT	BOOL	Petición finalizada sin errores, datos aceptados Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7904) contiene la información de error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido 1 ≤ LEN ≤ 1024, indicados en número de bytes
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7904) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

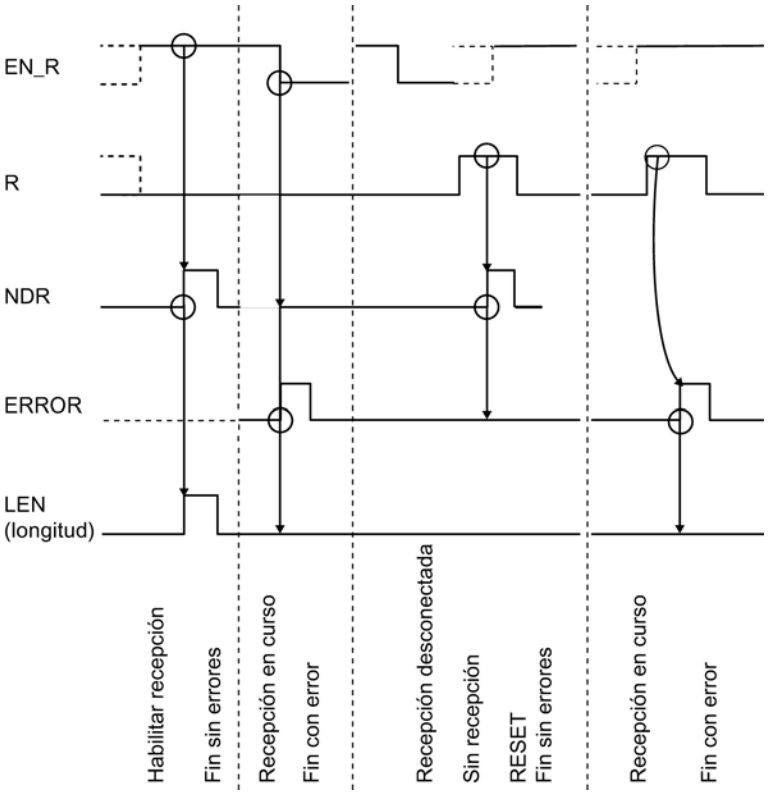
La instrucción P_RCV opera conjuntamente con un DB instancia I_RCV. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0E, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros NDR, LEN y ERROR en función de la conexión de las entrada EN_R y R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Regulación

Nota

La instrucción P_RCV no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el CP 340 pueda recibir una petición después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CP–CPU de la instrucción P_RCV .

P_PRINT: Imprimir un texto de aviso con hasta cuatro variables

Descripción

Para imprimir textos de aviso se dispone de la instrucción P_PRINT . La instrucción P_PRINT transmite, por ejemplo, un aviso de proceso al CP 340. El CP 340 registra el aviso de proceso en la impresora conectada.

La instrucción P_PRINT transmite un texto de aviso con hasta cuatro variables al CP 340. La instrucción P_PRINT se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

A través de los parámetros DB_NO y DBB_NO se accede a los punteros (puntero hacia bloques de datos) para la cadena de formato y las cuatro variables. Los punteros deben estar depositados sin huecos y en un orden determinado en el bloque de datos parametrizado (DB de punteros) (véase la figura "DB de punteros").

La transmisión del texto de aviso se dispara con un flanco ascendente en la entrada REQ. Primero se transfiere la cadena de formato del texto de aviso. Después se transfieren las variables 1 a 4.

Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción P_PRINT puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al CP 340y la instrucción P_PRINT pasa al estado inicial. Los datos que ya ha recibido el CP 340 se envían al interlocutor. Si en la entrada R está activado el estado lógico estático "1", significa que la transmisión de trabajos de impresión está desactivada.

En el parámetro LADDR se indica la dirección del CP 340 que se va a llamar.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS (##enlace al tema código de error). Si no se producen errores, STATUSasume el valor 0. DONEy ERROR/STATUStambién se visualizan en caso de RESETde la instrucción P_PRINT . Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Bloqueada salida por impresora.

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del CP 340 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos Puntero del DB de punteros: específico de la CPU. No se admite el valor 0. (Los punteros a variables y cadena de formato están registrados en el DB de punteros en un orden fijo. (Véase la imagen siguiente)).
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8162 puntero a partir de byte de datos
DONE [1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7904) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7904) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos, DB de instancia

La instrucción P_PRINT opera conjuntamente con un DB instancia I_PRINT. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

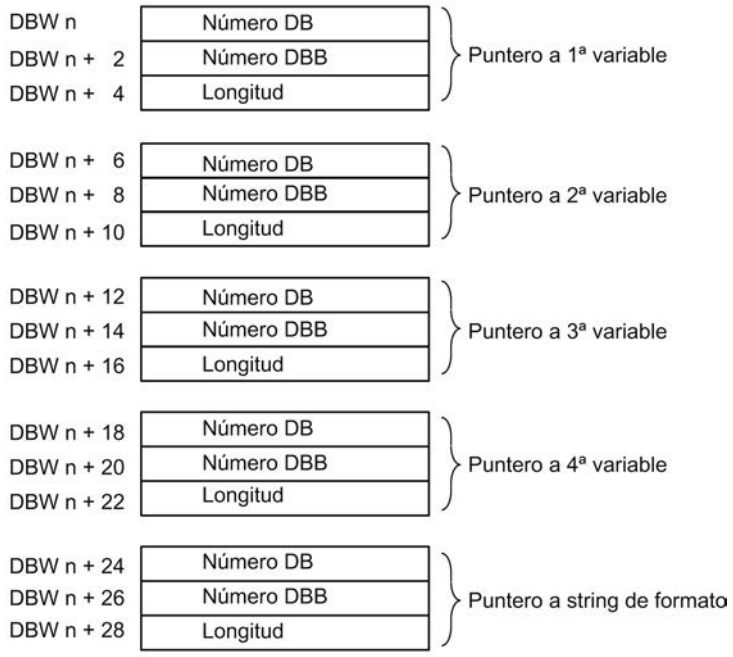
Nota

Excepción: en caso de error, STATUS == W#16#1E0F, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS.

Asignación en el área de datos, DB de punteros

La instrucción P_PRINT accede, a través de los parámetros DB_NO y DBB_NO, a un DB de punteros en el que están depositados en un orden fijo los punteros hacia los bloques de datos con los textos de aviso y las variables. El DB de punteros debe crearlo usted.

La figura muestra la estructura del DB de punteros, al que se accede con los parámetros DB_NO y DBB_NO de la instrucción P_PRINT .



Números de DB permitidos

Los números de DB permitidos son específicos de la CPU. Si con "Puntero en variable" se indica como número de DB el valor 16#00, dicha variable se interpreta como inexistente, y el puntero se coloca en la siguiente variable o en la cadena de formato.

Si con "Puntero hacia cadena de formato" el número de DB es igual al valor 16#00, se interrumpe el trabajo de impresión y se indica el número de evento 16#1E43 en la salida del parámetro STATUS de la instrucción P_PRINT .

Números de DBB permitidos

A partir del número de DBB parametrizado figura la variable o la cadena de formato. La máxima longitud permitida de las variables es 32 bytes, y de la cadena de formato 150 bytes.

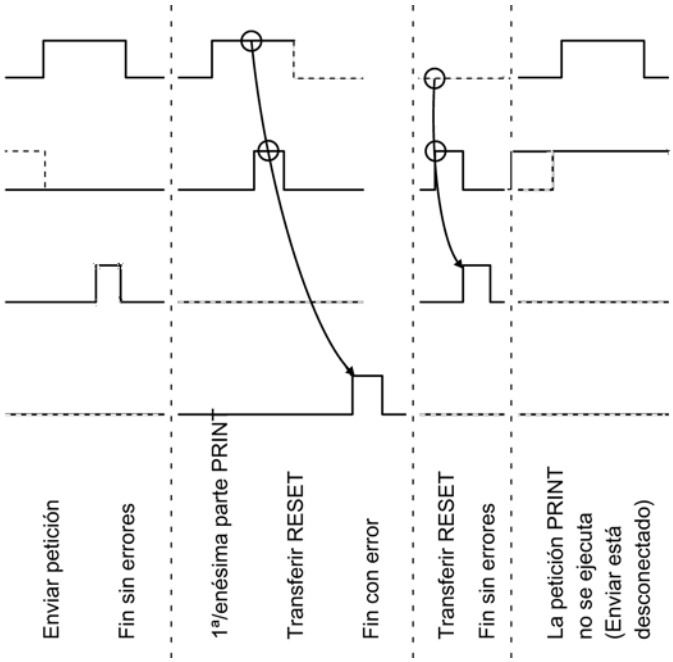
Si se sobrepasa la longitud máxima, se interrumpe el trabajo de impresión y se indica el número de evento 16#1E41 en la salida del parámetro STATUS de la instrucción P_PRINT.

Longitud permitida

La longitud indicada en el DB de punteros debe adaptarse para cada modo de representación (tipos de datos) y en función de la precisión utilizada.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONEy ERRORen función de la conexión de las entradas REQy R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. No tiene que estar durante toda la transmisión en el estado lógico "1".

Regulación

Nota

La instrucción P_PRINT no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el CP 341 pueda procesar una petición iniciada después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CP–CPU de la instrucción P_PRINT . Una petición iniciada entretanto no se pierde. Se transfiere al CP 340una vez concluida la coordinación de arranque.

Consulte también

Transmisión de datos con el driver de impresora (Página 1114)

P_RESET: Borrar el búfer de recepción

Descripción

La instrucción P_RESET borra completamente el búfer de recepción del CP 340. Todos los telegramas guardados se rechazan. Un telegrama entrante en el momento de llamar a la instrucción P_RESET se guarda.

Con un flanco ascendente en la entrada REQ se activa la instrucción. La petición puede extenderse varias llamadas (ciclos de programa).

Con el parámetro LADDR se indica la dirección del CP 340 que debe llamarse.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en Parámetro STATUS (Página 7904). Si no ha aparecido ningún error, STATUS tiene el valor 0. Si ha aparecido un error, se borra el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
LADDR	INPUT	INT	Dirección inicial del CP 340 La dirección inicial se toma de STEP 7.
DONE [1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00;
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores Parámetro STATUS (Página 7904) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el Parámetro STATUS (Página 7904) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

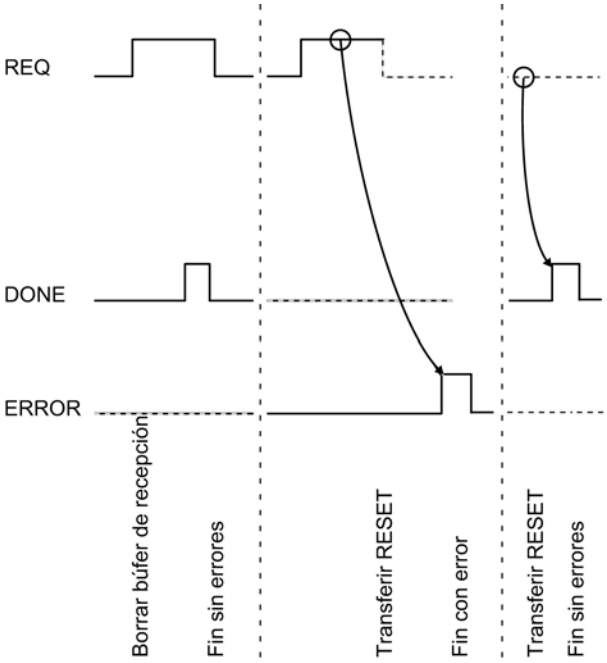
La instrucción P_RESET opera conjuntamente con un DB instancia I_P_RESET. El número del DB se indica con la llamada. No se puede acceder a los datos del DB de instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0F, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de la entrada REQ.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ . El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción P_RESET no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP .

V24_STAT_340 / V24_STAT: Leer señales cualificadoras en la interfaz RS232C

Descripción

La instrucción V24_STAT_340 (CP 340) / V24_STAT (CP 341) lee del CP 34x las señales cualificadoras RS 232C y las pone a disposición del usuario en los parámetros del bloque. La instrucción V24_STAT_340 / V24_STAT se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Las señales cualificadoras RS 232C se actualizan cada vez que se llama la instrucción (sondeo cíclico). El CP 34x actualiza el estado de las entradas/salidas en un tiempo de 20 ms. Con independencia de ello, las entradas/salidas se actualizan continuamente.

El resultado binario RB permanece igual. La instrucción no emite mensajes de error.

Con el parámetro LADDR se selecciona el CP 34x que se llamará.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del CP 34x La dirección inicial se toma de STEP 7.
DTR_OUT	OUTPUT	BOOL	Data terminal ready, CP 34x operativo (salida del CP 34x)
DSR_IN	OUTPUT	BOOL	Data set ready, interlocutor operativo (entrada del CP 34x)
RTS_OUT	OUTPUT	BOOL	Request to send, CP 34x listo para transmitir (salida del CP 34x)
CTS_IN	OUTPUT	BOOL	Clear to send, el interlocutor puede recibir datos del CP 34x (respuesta a RTS = ON del CP 34x) (entrada del CP 34x)

Parámetro	Declaración	Tipo de datos	Descripción
DCD_IN	OUTPUT	BOOL	**D**ata **C**arrier **d**etect, nivel de señal de recepción (entrada del CP 34x)
RI_IN	OUTPUT	BOOL	**R**ing **I**ndicator, indicador de llamada (entrada del CP 34x)

Asignación en el área de datos

La instrucción V24_STAT_340 / V24_STAT no ocupa áreas de datos.

Regulación

Nota

Para detectar un cambio de señal se requiere una duración de impulso mínima. Las magnitudes decisivas son el tiempo de ciclo de la CPU, el tiempo de actualización en la CP 34x y el tiempo de reacción del interlocutor.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

V24_SET_340 / V24_SET: Escribir señales cualificadoras en la interfaz RS232C

Descripción

El usuario puede activar o desactivar las correspondientes salidas de la interfaz a través de las entradas de parámetros de la instrucción V24_SET_340 (CP 340) / V24_SET (CP 341). La instrucción V24_SET_340 / V24_SET se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

El resultado binario RB permanece igual. La instrucción no emite mensajes de error.

Con el parámetro LADDR se selecciona el CP 34x que se llamará.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del CP 34x La dirección inicial se toma de STEP 7.
RTS	INPUT	BOOL	Request to send, CP 34x listo para transmitir (controlar la salida del CP 34x)
DTR	INPUT	BOOL	Data terminal ready, CP 34x operativo (controlar la salida del CP 34x)

Asignación en el área de datos

La instrucción V24_SET_340 / V24_SETno ocupa áreas de datos.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

Parámetro STATUS

Clases de evento

Clase de evento	Significado
5	Error al procesar una petición de la CPU
7	Error de transmisión
8	Error de recepción
30 ($1E_H$)	Error en la comunicación entre el CP y la CPU

Parámetro STATUS

Código de error (W#16#...)	Descripción	Solución
0502	La petición no es válida en este estado operativo del CP (p. ej. interfaz de dispositivo no parametrizada).	Evalúe la alarma de diagnóstico y corrija el error correspondiente.
0505	**Sólo con driver de impresora:** Bloque de datos del sistema con textos de aviso no disponible en CP	Configure los textos de aviso con el software de parametrización y reinicie.
0506	**Sólo con driver de impresora:** Texto de aviso no disponible	Configure los textos de aviso con el software de parametrización y reinicie.

Código de error (W#16#...)	Descripción	Solución
0507	**Sólo con driver de impresora:** Texto de aviso demasiado largo	Modifique el texto de aviso hasta un máximo de 150 caracteres (a un máximo de 250 caracteres en caso de variables)
0508	**Sólo con driver de impresora:** Demasiadas instrucciones de conversión	Ha configurado más instrucciones de conversión que variables. Las instrucciones de conversión sin la variable correspondiente se ignoran.
0509	**Sólo con driver de impresora:** Demasiadas variables	Ha configurado más variables que instrucciones de conversión. Las variables sin instrucción de conversión no se imprimen.
050A	**Sólo con driver de impresora:** Instrucción de conversión desconocida	Revise la instrucción de conversión. Las instrucciones de conversión no definidas o no compatibles se sustituyen por ****** en la impresión.
050B	**Sólo con driver de impresora:** Instrucción de control desconocida	Revise la instrucción de control. Las instrucciones de control indefinidas o incompatibles se ignoran. La instrucción de control tampoco se imprime en forma de texto.
050C	**Sólo con driver de impresora:** Instrucción de conversión no ejecutable	Revise la instrucción de conversión. Las instrucciones de conversión no ejecutables se imprimen con caracteres * de acuerdo con el ancho definido y la parte válida restante de la instrucción, o bien de la representación estándar.
050D	**Sólo con driver de impresora:** Ancho insuficiente o excesivo en la instrucción de conversión	Corrija el ancho especificado para la variable de la instrucción de conversión a partir del número máximo de caracteres de la variable en los modos de representación basados en texto (A, C, D, S, T, Y, Z). En la impresión sólo se imprime el número de caracteres correspondiente al ancho especificado, el resto no se imprime. En el resto de casos se imprimen caracteres * en la cantidad equivalente al ancho.
050E	**Sólo para driver 3964(R) y ASCII:** Longitud de telegrama no válida	La longitud de telegrama es > 1024 bytes. El CP 340 continúa recibiendo el resto del telegrama (> 1024 bytes) y, por tanto, rechaza la primera parte del telegrama. Seleccione una longitud de telegrama menor.
051B	**Sólo con driver de impresora:** Precisión no válida	Corrija la precisión indicada en la instrucción de conversión. La precisión siempre está precedida por un punto que identifica y limita el ancho (p. ej.: ".2" para la impresión de punto flotante y 2 posiciones tras la coma). La precisión sólo es relevante en los modos de representación F, R, A y D. En el resto de casos se ignora.

Código de error (W#16#...)	Descripción	Solución
051C	**Sólo con driver de impresora:** Variable no válida (Longitud de variable incorrecta/tipo incorrecto)	Corrija la variable indicada. Consulte los tipos de datos válidos para el modo de representación en cuestión en la tabla respectiva.
051E	**Sólo con driver de impresora:** Las "secuencias de fin de línea" enviadas con esta petición (p. ej.: $R / $L / $N) (ya) no encajan en la página (empezada)	Aumente la longitud de la página, reduzca la cantidad de líneas (o los saltos de línea) o reparta la impresión en varias páginas
0701	**Sólo con 3964(R):** Envío de la primera repetición: • Se ha detectado un error durante el envío del telegrama o bien • El interlocutor ha solicitado una repetición con un carácter de confirmación negativo (NAK).	Un reintento no constituye un error, pero puede ser el indicio de un fallo en la línea de transferencia o de un comportamiento erróneo del equipo interlocutor. Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se notifica un número de error que describe el error que se produjo en primer lugar.
0702	**Sólo con 3964(R):** Error de establecimiento de conexión: Después del envío del carácter STX, se ha recibido el carácter NAK o cualquier otro carácter (excepto DLE o STX).	Compruebe por qué falla el dispositivo del interlocutor y, si es necesario, con un dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0703	**Sólo con 3964(R):** Se ha excedido el tiempo de retardo de acuse (TRA): Después del envío de STX no se ha recibido ninguna respuesta del interlocutor en el tiempo de retardo de acuse.	El dispositivo interlocutor es demasiado lento o no está preparado para recibir o bien hay p. ej. una rotura de la línea de envío. Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0704	**Sólo con 3964(R):** Interrupción por parte del interlocutor: Durante el proceso de envío actual el interlocutor ha recibido uno o más caracteres.	Compruebe si el interlocutor también ha detectado errores. Puede ser que no se hayan recibido todos los datos enviados (p. ej. debido a una rotura en la línea de envío) o que haya fallos graves, o bien un comportamiento erróneo del interlocutor. Compruebe si es así con el dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0705	**Sólo con 3964(R):** Acuse negativo durante el envío.	Compruebe si el interlocutor también ha detectado errores. Puede ser que no se hayan recibido todos los datos enviados (p. ej. debido a una rotura en la línea de envío) o que haya fallos graves, o bien un comportamiento erróneo del interlocutor. Compruebe si es así con el dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0706	**Sólo con 3964(R):** Error en el extremo del enlace: • El telegrama ha sido rechazado por el interlocutor con NAK u otro carácter al final (con la excepción de DLE) o bien • El carácter de confirmación (DLE) se ha recibido demasiado pronto.	Compruebe si el interlocutor también ha detectado errores. Puede ser que no se hayan recibido todos los datos enviados (p. ej. debido a una rotura en la línea de envío) o que haya fallos graves, o bien un comportamiento erróneo del interlocutor. Compruebe si es así con el dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0707	**Sólo con 3964(R):** Se ha excedido el tiempo de retardo de acuse al final del enlace/tiempo de vigilancia de respuesta tras el telegrama: No se ha recibido ninguna respuesta del interlocutor en el TRA después de establecerse la conexión con DLE ETX.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe si es así con el dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0708	**Sólo para driver ASCII y driver de impresora:** El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor está defectuoso, es demasiado lento o se encuentra offline. Compruebe el interlocutor o modifique si es necesario la parametrización.
070B	**Sólo con 3964(R):** No se puede resolver el conflicto de inicialización porque ambos interlocutores tienen prioridad alta.	Modifique la parametrización.
070C	**Sólo con 3964(R):** No se puede resolver el conflicto de inicialización porque ambos interlocutores tienen prioridad baja.	Modifique la parametrización.
0801	**Sólo con 3964(R):** Espera del primer reintento: Al recibir un telegrama se ha detectado un error y el CP exige mediante un acuse negativo (NAK) un reintento del interlocutor.	Un reintento no constituye un error, pero puede ser el indicio de un fallo en la línea de transferencia o de un comportamiento erróneo del equipo interlocutor. Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se notifica un número de error que describe el error que se produjo en primer lugar.

Código de error (W#16#...)	Descripción	Solución
0802	**Sólo con 3964(R):** Error de establecimiento de conexión: • En posición de reposo se han recibido uno o más caracteres (cualquiera con la excepción de NAK o STX) o bien • Tras la recepción de un STX el interlocutor ha enviado otros caracteres sin esperar la recepción de la respuesta DLE. Tras RED-ON del interlocutor: • Durante el tiempo de conexión del interlocutor, el CP recibe un carácter indefinido.	Compruebe el comportamiento erróneo del interlocutor, en caso necesario con un dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0805	**Sólo con 3964(R):** Error lógico durante la recepción: Después de la recepción del DLE se ha recibido otro carácter (diferente a DLE O ETX).	Compruebe que el interlocutor no haya incluido el DLE por duplicado en la cabecera del telegrama o en la cadena de datos, y que el enlace se establezca con DLE ETX. Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0806	**Se ha excedido el tiempo de retardo de caracteres (TRC):** • No se han recibido dos caracteres consecutivos dentro del TRC o bien **Sólo con 3964(R):** • 1. El 1er carácter tras el envío de DLE en el establecimiento de la conexión no se ha recibido dentro del TRC.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruébelo con el dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
0807	**Longitud de telegrama no válida:** Se ha recibido un telegrama de longitud 0.	La recepción de un telegrama de longitud 0 no constituye un error. Compruebe el motivo por el que el interlocutor envía telegramas sin datos útiles.
0808	**Sólo con 3964(R):** Error en el carácter de control de bloques BCC: El valor interno del BCC no concuerda con el BCC recibido por el interlocutor en el extremo del enlace.	Compruebe que el enlace no se encuentre seriamente dañado. De ser así, observará de vez en cuando códigos de error. Compruebe el comportamiento erróneo del interlocutor, en caso necesario con un dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0809	**Sólo con 3964(R):** El número de repeticiones debe ajustarse igual.	Parametrice en el interlocutor el mismo tiempo de espera de bloques que en el CP 340. Compruebe el comportamiento erróneo del interlocutor, en caso necesario con el dispositivo de comprobación de interfaz (FOXPG) conectado a la línea de transmisión.
080A	No se encuentra ningún búfer de recepción disponible: Durante la recepción no había ningún búfer de recepción vacío.	La instrucción P_RCV se debe llamar con más frecuencia.
080C	Error de transferencia: • Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de desbordamiento). **Sólo con 3964(R):** • Si esto ocurre durante el modo de transferencias o recepción, se inician repeticiones. • Si durante la posición de reposo se recibe un carácter defectuoso, el error se indica inmediatamente para facilitar la detección prematura de interferencias en la línea de transmisión. • Si están encendidos los LEDs SF (rojo) y RxD (verde), existe una interrupción (rotura de hilo) en el cable que conecta ambos interlocutores.	Las perturbaciones de la línea de transferencia ocasionen repeticiones de los telegramas y disminuyen con ello la media de datos útiles. Todo ello aumenta los riesgos de que existan errores no localizados. Modifique la estructura de su sistema o el tendido de cables. Compruebe la línea de enlace de los interlocutores y asegúrese de que en ambos interlocutores esté ajustada la misma velocidad de transferencia, la misma paridad y el mismo número de bits de parada.
080D	BREAK: Se ha interrumpido la línea de recepción del interlocutor.	Restablezca el enlace o conecte el interlocutor.
0810	**Sólo con drivers ASCII:** Error de paridad: Si están encendidos el LED SF (rojo) y el LED RxD (verde), existe una interrupción del cable de unión (interrupción de cable) entre ambos interlocutores.	Compruebe la línea de enlace de los interlocutores y asegúrese de que en ambos interlocutores esté ajustada la misma velocidad de transferencia, la misma paridad y el mismo número de bits de parada. Modifique la estructura de su sistema o el tendido de cables.
0811	**Sólo con drivers ASCII:** Error de trama de caracteres: Si están encendidos el LED SF (rojo) y el LED RxD (verde), existe una interrupción del cable de unión (interrupción de cable) entre ambos interlocutores.	Compruebe la línea de enlace de los interlocutores y asegúrese de que en ambos interlocutores esté ajustada la misma velocidad de transferencia, la misma paridad y el mismo número de bits de parada. Modifique la estructura de su sistema o el tendido de cables.

Código de error (W#16#...)	Descripción	Solución
0812	**Sólo con drivers ASCII:** Después de que el CP enviara XOFF o pusiera CTS a OFF, se recibieron más caracteres.	Vuelva a parametrizar el interlocutor o procure una lectura más rápida del CP.
0818	**Sólo con drivers ASCII:** DSR = OFF o CTS = OFF	Antes o durante un proceso de envío el interlocutor ha puesto todas las señales DSR o CTS a "OFF". Revise el control de las señales cualificadoras RS 232C en el interlocutor.
1E0D	Interrupción del trabajo por rearranque completo, rearranque o reset	
1E0E	Error estático al acceder a la SFC RD_REC o al SFB RDREC. Puede disponer del valor de retorno RET_VAL de la SFC / el SFB en la variable SFCERR o SFCSTATUS del DB de instancia, para su evaluación.	Cargue la variable SFCERR o SFCSTATUS del DB de instancia.
1E0F	Error estático al acceder a la SFC WR_REC o al SFB RDREC. Puede disponer del valor de retorno RET_VAL de la SFC / el SFB en la variable SFCERR o SFCSTATUS del DB de instancia, para su evaluación.	Cargue la variable SFCERR o SFCSTATUS del DB de instancia.
1E41	No está permitido el número de bytes indicado en el parámetro LEN de los FB	Respete el intervalo de 1 a 1024 bytes.
1E41	Instrucción P_PRINT: El número de bytes indicado para la variable o la cadena de formato dentro de la longitud del DB de punteros no es válido.	Observe las longitudes permitidas: 32 bytes para variables, 150 bytes para cadena de formato
1E43	Instrucción P_PRINT: No existe puntero para la cadena de formato.	Indique en el DB de punteros el nº de bloque de datos y el nº de palabra de datos para la cadena de formato.

Nota

Sólo se envía un mensaje de error si simultáneamente está activado el bit ERROR (petición interrumpida con errores). En otro caso, la palabra de STATUS será cero.

Llamar la variable SFCERR o SFCSTATUS

Con la variable SFCERR o SFCSTATUS puede obtener más información sobre el error producido, 14 (1E0EH) y 15 (1E0FH), de la categoría de eventos 30.

Puede cargar la variable SFCERR o SFCSTATUS sólo desde un acceso simbólico al DB instancia de la instrucción correspondiente.

Los mensajes de error que se registran en la variable SFCERR se encuentran en las funciones de sistema SFC 58 "WR_REC: Escribir registro en periferia (Página 6774)" y SFC 59 "RD_REC: Leer registro de periferia (Página 6769)".

Los mensajes de error que se registran en la variable SFCSTATUS se encuentran en las funciones de sistema SFB 52 "RDREC: Leer registro (Página 6735)" y SFB 53 "WRREC: Escribir registro (Página 6738)".

Acoplamiento PtP CP 341

P_SND_RK: Enviar o recuperar

P_SND_RK

La instrucción P_SND_RK distingue los siguientes casos de aplicación:

- P_SND_RK: Enviar datos con 3964(R) o driver ASCII (Página 7911)
- P_SND_RK: Enviar datos con RK 512 (Página 7914)
- P_SND_RK: Recuperar datos con RK 512 (Página 7918)

Consulte también

Transferencia de datos de la CPU al módulo de comunicación con P_SND_RK (CP 341) (Página 1134)

P_SND_RK: Enviar datos con 3964(R) o driver ASCII

Descripción

La instrucción P_SND_RK transmite un conjunto de datos de un bloque de datos especificado por los parámetros DB_NO, DBB_NO y LEN, al CP 341. La instrucción P_SND_RK se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción P_SND_RK puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al CP 341 y la instrucción P_SND_RK pasa al estado inicial. Los datos que ya ha recibido el CP 341 se envían al interlocutor. Si en la entrada R hay un estado lógico estático "1", la transmisión está desactivada.

Con el parámetro LADDR se indica la dirección del CP 341 que debe llamarse.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_SND_RK . Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del CP 341 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º de DB emisor: específico de CPU, no está permitido el cero
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8190 datos de transmisión a partir del byte de datos
LEN	INPUT	INT	Longitud de datos 1 ≤ LEN ≤ 4096, indicados en número de bytes
DONE[1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Nota

Los parámetros R_CPU_NO, R_TYP, R_NO, R_OFFSET, R_CF_BYT y R_CF_BIT no son importantes para el procedimiento 3964(R) y el driver ASCII y no tienen que recibir valores. El parámetro SF tampoco tiene que recibir valores, porque "S" está predeterminado para enviar.

Asignación en el área de datos

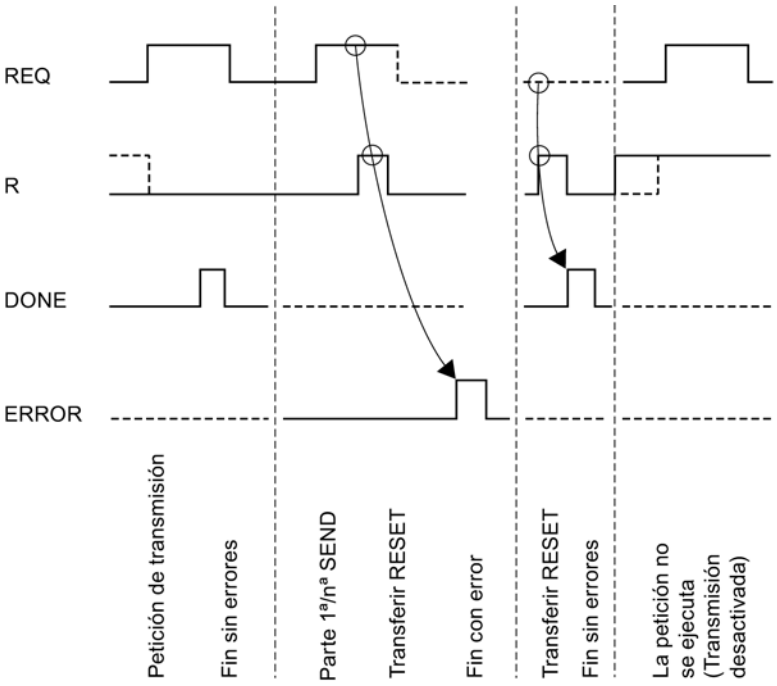
La instrucción P_SND_RKopera conjuntamente con un DB instancia I_SND_RK. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0F, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS. Esta variable de error sólo puede cargarse en el DB instancia mediante un acceso simbólico.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción P_SND_RK no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el CP 341 pueda procesar una petición iniciada después de una transición de estado de la CPU de STOP a RUN , debe haber concluido el mecanismo de arranque CP–CPU de la instrucción P_SND_RK . Una petición iniciada entretanto no se pierde. Se transfiere al CP 341 una vez concluida la coordinación de arranque.

P_SND_RK: Enviar datos con RK 512

Descripción

La instrucción P_SND_RK puede transmitir datos desde un área de datos S7 a un CP 341 configurando el parámetro SF = 'S'.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo del volumen de datos (LEN), la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Con el parámetro LADDR se indica la dirección del CP 341 que debe llamarse.

La única fuente permitida para los datos a enviar es el área de los bloques de datos. La fuente está completamente especificada si se indica el número de bloque de datos (DB_NO) y el offset (DBB_NO) del primer byte de datos que debe enviarse de dicho bloque de datos.

Los tipos de datos (R_TYP) permitidos como áreas de destino son los bloques de datos (DB) y los bloques de datos ampliados (DX). El destino está completamente especificado mediante el número de CPU (R_CPU_NO sólo relevante en la comunicación multiprocesador), el tipo de datos (R_TYP: DB o DX), el número de bloque de datos (R_NO) y el offset (R_OFFSET), en el que debe escribirse el primer byte.

Con R_CF_BYT y R_CF_BIT se definen en la CPU del interlocutor el byte y el bit de marca de acoplamiento.

La instrucción P_SND_RK puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al CP 341 y la instrucción P_SND_RK pasa al estado inicial. Los datos que ya ha recibido el CP 341 se envían al interlocutor. Si la entrada R tiene el estado lógico estático "1", la transmisión está desactivada.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_SND_RK . Si se ha producido

un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
SF	INPUT	CHAR	Selección para enviar o recuperar datos SF = 'S' (enviar) Valor predeterminado: 'S'
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se cancela la petición en curso. Transmisión bloqueada. Valor predeterminado: 0
LADDR	INPUT	INT	Dirección inicial del CP 341 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos del origen N.º de DB emisor: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos del origen 0 ≤ DBB_NO ≤ 8190 datos de transmisión a partir del byte de datos
LEN	INPUT	INT	Longitud de datos del telegrama que debe enviarse 1 ≤ LEN ≤ 4096, indicación en número de bytes, sólo se recomiendan valores pares
R_CPU_NO	INPUT	INT	N.º de la CPU del interlocutor 0 ≤ R_CPU_NO ≤ 4, sólo en modo multiprocesador, valor predeterminado: 1
R_TYP	INPUT	CHAR	Tipo de dirección en la CPU del interlocutor 'D': Bloque de datos 'X': bloque de datos ampliado
R_NO	INPUT	INT	Número de bloque de datos en la CPU del interlocutor 0 ≤ R_NO ≤ 255
R_OFFSET	INPUT	INT	Número de byte de datos en la CPU del interlocutor 0 ≤ R_OFFSET ≤ 510,sólo valores pares
R_CF_BYT	INPUT	INT	Byte de marca de acoplamiento en la CPU del interlocutor 0 ≤ R_CF_BYTE ≤ 255 Valor predeterminado: 255 (significa: sin marca de acoplamiento)
R_CF_BIT	INPUT	INT	Bit de marca de acoplamiento en la CPU del interlocutor 0 ≤ R_CF_BIT ≤ 7
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores Parámetro STATUS == 16#00;

Parámetro	Declaración	Tipo de datos	Descripción
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Información del encabezado del telegrama

En la siguiente tabla encontrará las indicaciones que figuran en el encabezado del telegrama RK 512.

Tabla 9-322 Información del encabezado del telegrama RK 512 para la petición "Enviar datos"

Origen en el sistema de automatización S7 (CPU local)	Para el destino, CPU del interlocutor	Encabezado del telegrama, bytes		
		3/4 tipo de comando	5/6 DBNR D/Offset D	7/8 Número en
Bloque de datos	Bloque de datos	AD	DB/DW	Palabras
Bloque de datos	Bloque de datos ampliado	AD	DB/DW	Palabras
Z-DBNR: Número del bloque de datos de destino Z-Offset: Dirección inicial del destino DW: Offset en palabras				

Asignación en el área de datos

La instrucción P_SND_RK opera conjuntamente con un DB instancia I_SND_RK. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

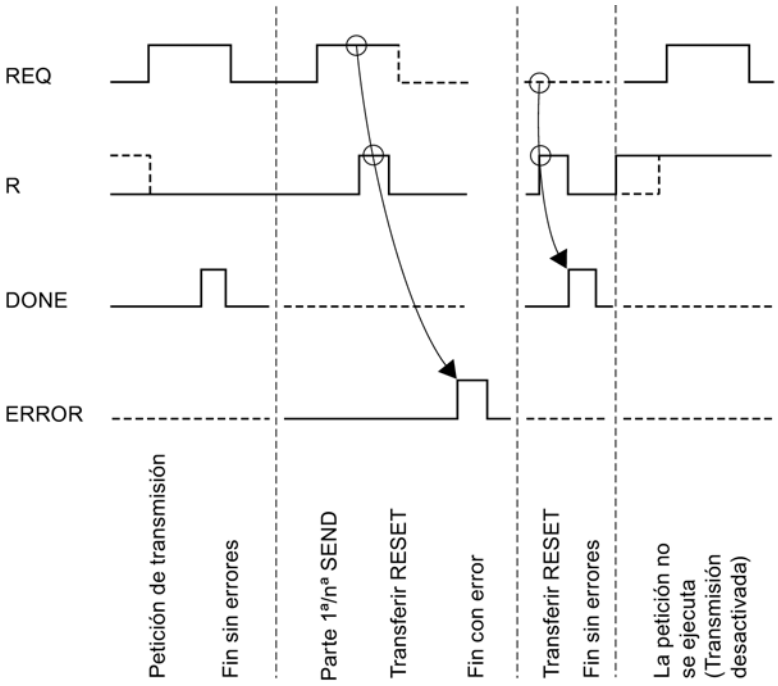
Particularidades al enviar datos

Observe las siguientes particularidades al "Enviar datos":

- Con el protocolo RK 512 sólo es posible enviar un número par de datos. Por esta razón, si especifica una longitud (LEN) impar de datos, al final de los datos se transmite un byte de relleno adicional con el valor "0".
- Con el protocolo RK 512 sólo puede indicarse un offset par. Por esta razón, si se especifica un offset impar, se depositan los datos a partir del siguiente Offset par inmediatamente inferior del interlocutor.
  Ejemplo: offset es 7; se depositan los datos a partir del byte 6.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción P_SND_RK no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

P_SND_RK: Recuperar datos con RK 512

Descripción

La instrucción P_SND_RK recupera datos de un interlocutor remoto configurando el parámetro SF = F y los deposita en un área de datos S7 del sistema de automatización.

Nota

Si se recuperan los datos de un CP 341, hay que programar siempre en el CP 341 una instrucción P_RCV_RK.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo del volumen de datos (LEN), la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Con el parámetro LADDR se indica la dirección del CP 341 que debe llamarse.

Indicando el número de CPU (R_CPU_NO, sólo relevante en la comunicación multiprocesador) se especifica el interlocutor del que se recuperan los datos. Las fuentes permitidas para recuperar los datos son los siguientes tipos de datos (R_TYP): bloques de datos, bloques de datos ampliados, marcas, entradas, salidas, contadores y temporizadores. La fuente está completamente especificada indicando el tipo de datos (R_TYP), el número de bloque de datos (R_NO, sólo relevante en bloques de datos y bloques de datos ampliados) y el offset (R_OFFSET) del primer byte de datos que debe enviarse de dicha área.

Con R_CF_BYT y R_CF_BIT se definen en la CPU del interlocutor el byte y el bit de marca de acoplamiento.

Las únicas áreas de destino permitidas son los bloques de datos (DB). La fuente está completamente especificada si se indica el número de bloque de datos (DB_NO) y el offset (DBB_NO) en el que debe escribirse el primer byte de datos.

La instrucción P_SND_RK puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia del CP 341 y la instrucción P_SND_RK pasa al estado inicial. Si la entrada R tiene el estado lógico estático "1", la recuperación está desactivada.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_SND_RK . Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
SF	INPUT	CHAR	Selección para enviar o recuperar datos SF = 'F' (Fetch = recoger), valor por defecto: 'S' (enviar)
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se cancela la petición en curso. Bloqueo de recogida. Valor por defecto: 0
LADDR	INPUT	INT	Dirección inicial del CP 341 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos del destino N.º de DB emisor: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos del destino 0 ≤ DBB_NO ≤ 8190 datos de transmisión a partir del byte de datos
LEN	INPUT	INT	Longitud de datos del telegrama que debe recogerse 1 ≤ LEN ≤ 4096, indicación en número de bytes[1]
R_CPU_NO	INPUT	INT	N.º de la CPU del interlocutor 0 ≤ R_CPU_NO ≤ 4, sólo en modo multiprocesador, valor predeterminado: 1
R_TYP	INPUT	CHAR	Tipo de dirección en la CPU del interlocutor 'D': bloque de datos 'X': bloque de datos ampliado 'M': marcas 'I': entradas 'Q': salidas 'C': contadores 'T': temporizadores
R_NO	INPUT	INT	Número de bloque de datos en la CPU del interlocutor 0 ≤ R_NO ≤ 255
R_OFFSET	INPUT	INT	Número de byte de datos en la CPU del interlocutor
R_CF_BYT	INPUT	INT	Byte de marca de acoplamiento en la CPU del interlocutor 0 ≤ CF_BYTE ≤ 255 valor por defecto: 255 (significa: sin marca de acoplamiento)
R_CF_BIT	INPUT	INT	Bit de marca de acoplamiento en la CPU del interlocutor 0 ≤ CF_BIT ≤ 7
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.

Parámetro	Declaración	Tipo de datos	Descripción
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Parámetros R_TYP, R_NO y R_OFFSET

Origen en la CPU del interlocutor	R_TYP	R_NO	R_OFFSET (en bytes) (Este valor está predeterminado por la CPU del interlocutor.)
Bloque de datos	'D'	0 - 255	0 - 510 (sólo se recomiendan valores pares)
Bloque de datos ampliado	'X'	0 - 255	0 - 510 (sólo se recomiendan valores pares)
Marcas	'M'	Irrelevante	0 - 255
Entradas	'I'	Irrelevante	0 - 255
Salidas	'Q'	Irrelevante	0 - 255
Contadores	'C'	Irrelevante	0 - 255
Temporizadores	'T'	Irrelevante	0 - 255

Información del encabezado del telegrama

En la siguiente tabla encontrará las indicaciones que figuran en el encabezado del telegrama RK 512.

Origen en la CPU del interlocutor	Para el destino de su sistema de automatización S7 (CPU local)	Encabezado del telegrama, bytes		
		3/4 tipo de comando	5/6 DBNR F/Offset F	7/8 Número en
Bloque de datos	Bloque de datos	ED	DB/DW	Palabras
Bloque de datos ampliado	Bloque de datos	EX	DB/DW	Palabras
Marcas	Bloque de datos	EM	Dirección de byte	Bytes
Entradas	Bloque de datos	EI	Dirección de byte	Bytes
Salidas	Bloque de datos	EQ	Dirección de byte	Bytes
Contadores	Bloque de datos	EC	Número del contador	Palabras
Temporizadores	Bloque de datos	ET	Número del temporizador	Palabras
Q-DBNR: Número de bloque de datos fuente Q-Offset: Dirección inicial de la fuente				

Asignación en el área de datos

La instrucción P_SND_RK opera conjuntamente con un DB instancia I_SND_RK. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Particularidades de los bloques de datos (ampliados)

Al "Recoger datos" de bloques de datos y de bloques de datos ampliados, observe las siguientes particularidades:

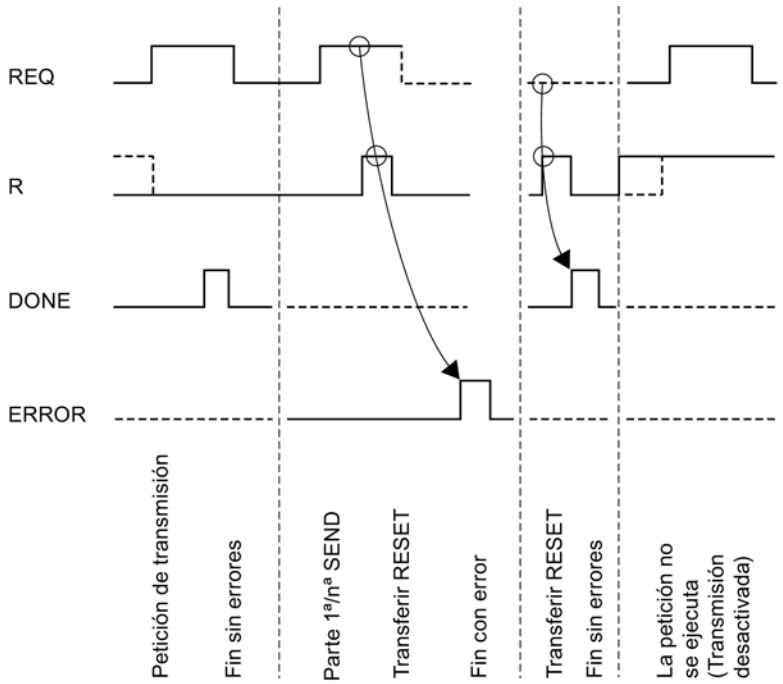
- Con el protocolo RK 512 sólo es posible recoger un número par de datos. Si se ha especificado como longitud (LEN) un número impar, se transmite siempre un byte más. Sin embargo, en el DB de destino se registra el número correcto de datos.
- Con RK 512 sólo es posible indicar un offset par. Por esta razón, si se especifica un offset impar, se recogen los datos del siguiente offset par inmediatamente inferior del interlocutor. Ejemplo: Offset es 7; se recogen los datos a partir del byte 6.

Particularidades de temporizadores y contadores

Si recoge temporizadores o contadores del interlocutor, debe tener en cuenta que para cada temporizador o contador se recogen 2 bytes. Si, p. ej., desea recoger 10 contadores, debe indicar la longitud 20.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción P_SND_RK no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

P_RCV_RK: Recibir datos o preparar datos

P_RCV_RK

La instrucción P_RCV_RK distingue los siguientes casos de aplicación:

- P_RCV_RK: Recibir datos con 3964(R) o driver ASCII (Página 7922)
- P_RCV_RK: Recibir datos con RK 512 (Página 7925)
- P_RCV_RK: Preparar datos con RK 512 (Página 7929)

Consulte también

Transferencia de datos del módulo de comunicación a la CPU con P_RCV_RK (CP 341) (Página 1136)

P_RCV_RK: Recibir datos con 3964(R) o driver ASCII

Descripción

La instrucción P_RCV_RK transmite datos del CP 341 a un área de datos S7, especificada por los parámetros DB_NO, DBB_NO y LEN. La instrucción P_RCV_RK se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del CP 341 . Es posible cancelar una transmisión en curso con el estado

lógico "0" en el parámetro EN_R . La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desactivada mientras exista el estado lógico "0" en el parámetro EN_R . Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción P_RCV_RK pasa al estado inicial. La recepción está desactivada mientras el parámetro R tenga el estado lógico "1".

Con el parámetro LADDR se selecciona el CP 341 que se llamará.

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS . Si no se producen errores, STATUS asume el valor 0. NDR y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_RCV_RK (parámetro LEN == 16#00). Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitar lectura de datos
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Bloqueo de recepción.
LADDR	INPUT	INT	Dirección inicial del CP 341 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º del DB de recepción: específico de la CPU, no se admite el valor cero
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8190 datos de recepción a partir del byte de datos
NDR[1]	OUTPUT	BOOL	Petición finalizada sin errores, datos aceptados Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido 1 ≤ LEN ≤ 4096, indicados en número de bytes
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Nota

Los parámetros L_TYP, L_NO, L_OFFSET, L_CF_BYT y L_CF_BIT no son importantes para el procedimiento 3964(R) y el driver ASCII y no tienen que recibir valores.

Asignación en el área de datos

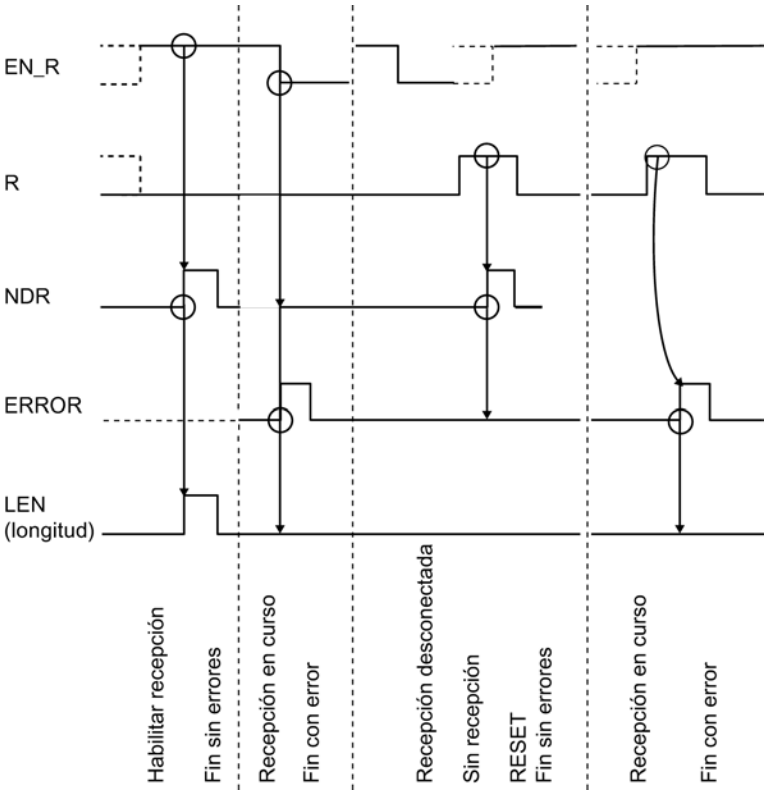
La instrucción P_RCV_RK opera conjuntamente con un DB instancia I_RCV_RK. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0E, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS. Esta variable de error sólo puede cargarse en el DB instancia mediante un acceso simbólico.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros NDR, LEN y ERROR en función de la conexión de las entrada EN_R y R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Regulación

Nota

La instrucción P_RCV_RK no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP .

Nota

Antes de que el CP 341 pueda recibir una petición después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CP–CPU de la instrucción P_RCV_RK .

P_RCV_RK: Recibir datos con RK 512

Descripción

La instrucción P_RCV_RK transmite datos del CP 341 a un área de datos S7. La instrucción P_RCV_RK se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del CP 341 . Es posible cancelar una transmisión en curso con el estado lógico "0" en el parámetro EN_R . La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desactivada mientras exista el estado lógico "0" en el parámetro EN_R . Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Con el parámetro LADDR se indica la dirección del CP 341 que debe llamarse.

Si el interlocutor indica como destino de datos "DB", los datos se depositan en el área de datos especificada en el encabezado del telegrama RK 512. Con ayuda de los parámetros (L_...) se indica al usuario el tipo del área de destino (L_TYP), el número de bloque de datos de destino (L_NO, sólo relevante si L_TYP = DB), el offset del área de destino (L_OFFSET) y la longitud (LEN) de los datos transmitidos. Si el interlocutor indica el destino de datos "DX", los datos se depositan en el bloque de datos (DB) especificado por los parámetros DB_NO y DBB_NO .

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción P_RCV_RK pasa al estado inicial. La recepción estará desactivada mientras el parámetro R tenga el estado lógico "1".

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). A continuación, en los parámetros L_TYP, L_NO y L_OFFSET se indica dónde se van a depositar los datos de un mismo ciclo. Además, para un mismo ciclo se indican los parámetros L_CF_BYT y L_CF_BIT y la longitud LEN de la petición correspondiente.

Utilización de las marcas de acoplamiento

Antes de recibir los datos se comprueba la marca de acoplamiento indicada en el encabezado del telegrama RK 512. Los datos sólo se envían si la marca de acoplamiento tiene el valor "0". Una vez concluido el proceso de transmisión, la instrucción pone la marca de acoplamiento al valor "1", y en la instrucción se visualiza la marca de acoplamiento para un ciclo (NDR).

Evaluando la marca de acoplamiento en el programa de usuario puede detectarse que los datos transmitidos pueden procesarse. En cuanto han sido procesados los datos, el usuario debe poner nuevamente a "0" la marca de acoplamiento. Es posible una nueva petición SEND del interlocutor.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitación para recibir datos
R	INPUT	BOOL	Interrupción de la petición Se cancela la petición en curso. Recepción bloqueada. Valor predeterminado: 0
LADDR	INPUT	INT	Dirección base del CP 341 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos de recepción (destino) N.º DB de recepción: específico de la CPU. No se permite el valor 0. (sólo relevante si el destino de datos es DX)
DBB_NO	INPUT	INT	Número de byte de los datos de recepción (destino) 0 ≤ DBB_NO ≤ 8190 datos de recepción a partir del byte de datos (sólo relevante si el destino de datos es DX)
L_TYP[1]	OUTPUT	CHAR	Tipo de área en la CPU local (destino) 'D': Bloque de datos
L_NO [1]	OUTPUT	INT	Número de bloque de datos en la CPU local (destino) 0 ≤ L_NO ≤ 255

Parámetro	Declaración	Tipo de datos	Descripción
L_OFFSET[1]	OUTPUT	INT	Número de byte de datos en la CPU local (destino) 0 ≤ L_OFFSET ≤ 510
L_CF_BYT[1]	OUTPUT	INT	Byte de marcas de acoplamiento en la CPU local 0 ≤ L_CF_BYTE ≤ 255255 significa: sin marca de acoplamiento
L_CF_BIT [1]	OUTPUT	INT	Bit de marca de acoplamiento en la CPU local 0 ≤ L_CF_BIT ≤ 7
NDR[1]	OUTPUT	BOOL	La petición ha finalizado sin errores, datos transferidos Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido 0 ≤ LEN ≤ 4096, indicación en número de bytes
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

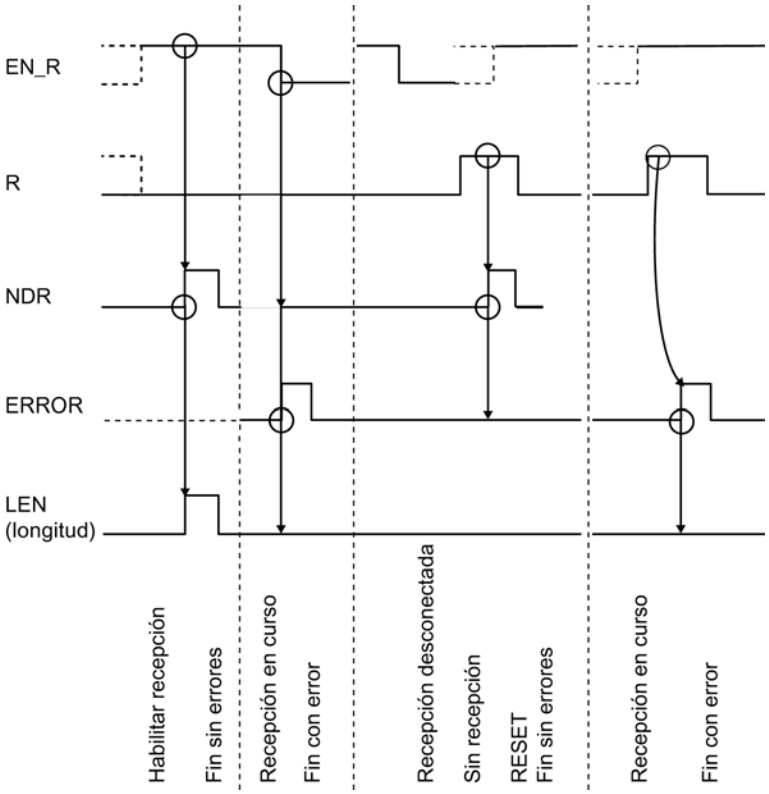
La instrucción P_RCV_RK opera conjuntamente con un DB instancia I_RCV_RK. El número del DB se indica con la llamada. No está permitido acceder a los datos en el DB de instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0E, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS. Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Cronograma de ejecución

La siguiente figura muestra el comportamiento de los parámetros NDR, LEN y ERROR en función de la conexión de las entradas EN_R y R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Comportamiento en caso de error

ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS . Si no se producen errores, STATUS asume el valor 0. NDR y ERROR/STATUS también se indican en caso de RESET de la instrucción P_RCV_RK (parámetro LEN == 16#00). Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Regulación

Nota

La instrucción P_RCV_RK no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP .

P_RCV_RK: Preparar datos con RK 512

Descripción

Es necesario llamar la instrucción P_RCV_RK cuando el interlocutor ejecuta una petición "Recuperar datos" (petición FETCH).

La instrucción P_RCV_RK prepara datos procedentes de un área de datos S7 para el CP 341 . La instrucción P_RCV_RK se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden prepararse datos en el CP 341 . Es posible cancelar una transmisión en curso con el estado lógico "0" en el parámetro EN_R . La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La petición está desactivada mientras exista el estado lógico "0" en el parámetro EN_R . Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

El tipo del área de origen (L_TYP), el número de bloque de datos de origen (L_NO, sólo relevante si L_TYP = DB), el offset en el área de origen (L_OFFSET) así como la longitud (LEN) de los datos que deben prepararse se determinan a partir del primer telegrama RK 512. La instrucción evalúa la información de ese telegrama y transmite los datos solicitados al CP 341. Los parámetros DB_NO y DBB_NO no son importantes para la instrucción P_RCV_RK .

Con el parámetro LADDR se indica la dirección del CP 341 que debe llamarse.

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción P_RCV_RK pasa al estado inicial. La petición está desactivada mientras exista el estado de señal "1" en el parámetro R.

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). A continuación, en los parámetros L_TYP, L_NO y L_OFFSET se indica desde dónde se han recuperado los datos dentro de un mismo ciclo (posibles tipos de datos: bloques de datos, bytes de entrada, bytes de salida, temporizadores y contadores). Además, para un mismo

ciclo se indican los parámetros L_CF_BYT y L_CF_BIT y la longitud LEN de la petición correspondiente.

Nota

Si el interlocutor recupera temporizadores o contadores del CP 341 , la longitud máxima está limitada a 32 bytes (16 temporizadores o contadores, cada uno de 2 bytes).

Utilización de las marcas de acoplamiento

Una vez recibido el telegrama se comprueban las marcas de acoplamiento indicadas en el encabezado del telegrama RK 512. Los datos sólo se proporcionan si la marca de acoplamiento tiene el valor "0". Una vez concluido el proceso de transmisión, la instrucción pone la marca de acoplamiento al valor "1", y en la instrucción se visualiza la marca de acoplamiento para un ciclo (NDR).

Evaluando la marca de acoplamiento en el programa de usuario puede detectarse que puede accederse nuevamente a los datos proporcionados. En cuanto han sido procesados los datos, el usuario debe poner nuevamente a "0" la marca de acoplamiento. Es posible una nueva petición FETCH del interlocutor.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitación para preparar datos
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Preparación bloqueada. Valor por defecto: 0
LADDR	INPUT	INT	Dirección inicial del CP 341 La dirección base se toma de STEP 7.
DB_NO	INPUT	INT	Irrelevante
DBB_NO	INPUT	INT	irrelevante
L_TYP[1]	OUTPUT	CHAR	Tipo de área en la CPU local (origen) 'D': Bloque de datos 'M': marcas 'I': entradas 'Q': salidas 'C': contadores 'T': Temporizadores
L_NO[1]	OUTPUT	INT	Número del bloque de datos en la CPU local (origen) 0 ≤ L_NO ≤ 255 (sólo relevante si L_TYP = D)
L_OFFSET[1]	OUTPUT	INT	Número del byte de datos en la CPU local (origen) 0 ≤ L_OFFSET ≤ 510 (en función del tipo de área)
L_CF_BYT[1]	OUTPUT	INT	Byte de marca de acoplamiento en la CPU local 0 ≤ CF_BYTE ≤ 255 255 significa: sin marca de acoplamiento

Parámetro	Declaración	Tipo de datos	Descripción
L_CF_BIT[1]	OUTPUT	INT	Bit de marca de acoplamiento en la CPU local 0 ≤ CF_BIT ≤ 7
NDR[1]	OUTPUT	BOOL	La petición ha finalizado sin errores, datos transferidos Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido 0 ≤ LEN ≤ 4096, indicados en número de bytes
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

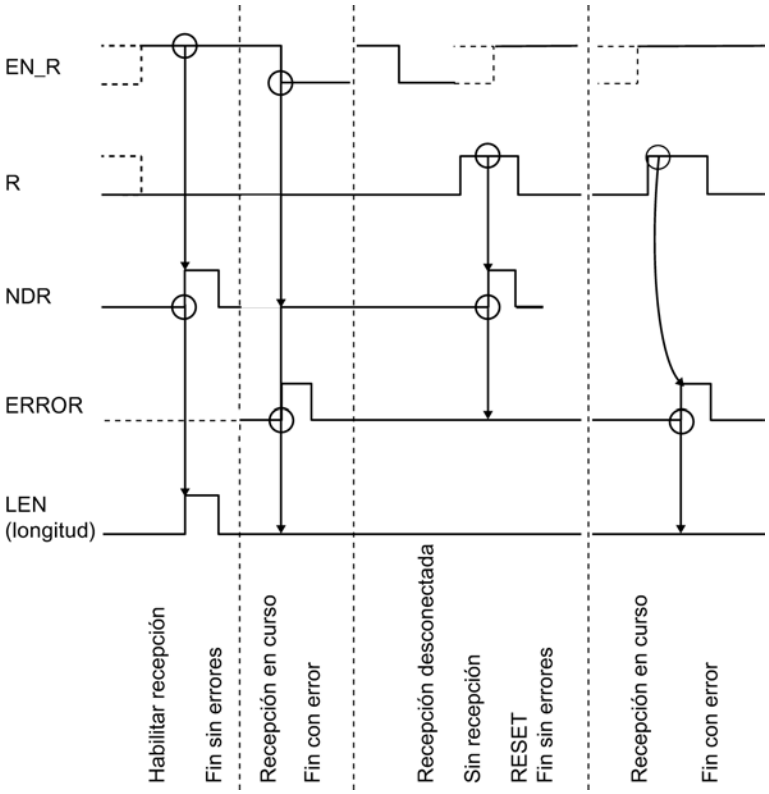
La instrucción P_RCV_RK opera conjuntamente con un DB instancia I_RCV_RK. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, STATUS == W#16#1E0E, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros NDR, LEN y ERROR en función de la conexión de las entradas EN_Ry R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Comportamiento en caso de error

ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. NDR y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_RCV_RK (parámetro LEN == 16#00). Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Regulación

Nota

La instrucción P_RCV_RK no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

P_PRT341: Imprimir un texto de aviso con hasta cuatro variables

Descripción

Para imprimir textos de aviso se dispone de la instrucción P_PRT341 . La instrucción P_PRT341 transmite, por ejemplo, un aviso de proceso al CP 341. El CP 341 registra el aviso de proceso en la impresora conectada.

Funcionamiento

La instrucción P_PRT341 transmite al CP 341 un texto de aviso que contiene hasta cuatro variables. La instrucción P_PRT341 se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

A través de los parámetros DB_NO y DBB_NO se accede a los punteros (puntero hacia bloques de datos) para la cadena de formato y las cuatro variables. Los punteros deben estar depositados sin huecos y en un orden determinado en el bloque de datos parametrizado (DB de punteros) (véase la figura "DB de punteros").

La transmisión del texto de aviso se dispara con un flanco ascendente en la entrada REQ. Primero se transfiere la cadena de formato del texto de aviso. Después se transfieren las variables 1 a 4.

Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción P_PRT341 puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al CP 341 y la instrucción P_PRT341 pasa al estado inicial. Los datos que ya ha recibido el CP 341 se envían al interlocutor. Si en la entrada R está activado el estado lógico estático "1", significa que la transmisión de trabajos de impresión está desactivada.

En el parámetro LADDR se indica la dirección del CP 341 que se va a llamar.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción P_PRT341 . Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición en flanco positivo
R	INPUT	BOOL	Interrupción de la petición Se cancela la petición en curso. Impresión bloqueada.
LADDR	INPUT	INT	Dirección base del CP 341 La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos Puntero hacia DB de punteros: específico de la CPU. No se permite el valor 0 (Los punteros en variable y cadena de formato están guardados en el DB de punteros en un orden fijo).
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8162, puntero a partir del byte de datos
DONE[1]	OUTPUT	BOOL	Petición terminada sin errores Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7939) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7939) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Ocupación en el área de datos, DB de instancia

La instrucción P_PRT341 opera conjuntamente con un DB instancia I_PRINT. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

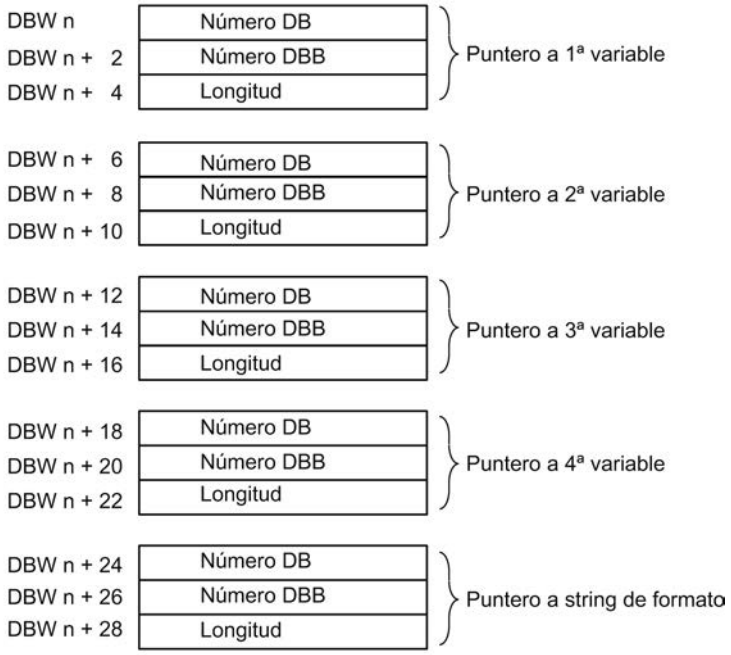
Nota

Excepción: en caso de error, STATUS == W#16#1E0F, encontrará una indicación más precisa sobre el error en la variable SFCERR o SFCSTATUS.

Ocupación en el área de datos - DB de punteros

La instrucción P_PRT341 accede, a través de los parámetros DB_NO y DBB_NO, a un DB de punteros en el que están depositados en un orden fijo los punteros hacia los bloques de datos con los textos de aviso y las variables. El DB de punteros debe crearlo el usuario.

La figura muestra la estructura del DB de punteros, al que se accede con los parámetros DB_NO y DBB_NO de la instrucción P_PRT341 .



Número de DB permitido

Los números de DB permitidos son específicos de cada CPU. Si con "Puntero en variable" se indica como número de DB el valor 16#00, dicha variable se interpreta como inexistente, y el puntero se coloca en la siguiente variable o en la cadena de formato.

Si con "Puntero hacia cadena de formato" el número de DB es igual al valor 16#00, se interrumpe el trabajo de impresión y se indica el número de evento 16#1E43 en la salida del parámetro STATUS de la instrucción P_PRT341 .

Número de DBB permitido

A partir del número de DBB parametrizado figura la variable o la cadena de formato. La máxima longitud permitida de las variables es 32 bytes, y de la cadena de formato 150 bytes.

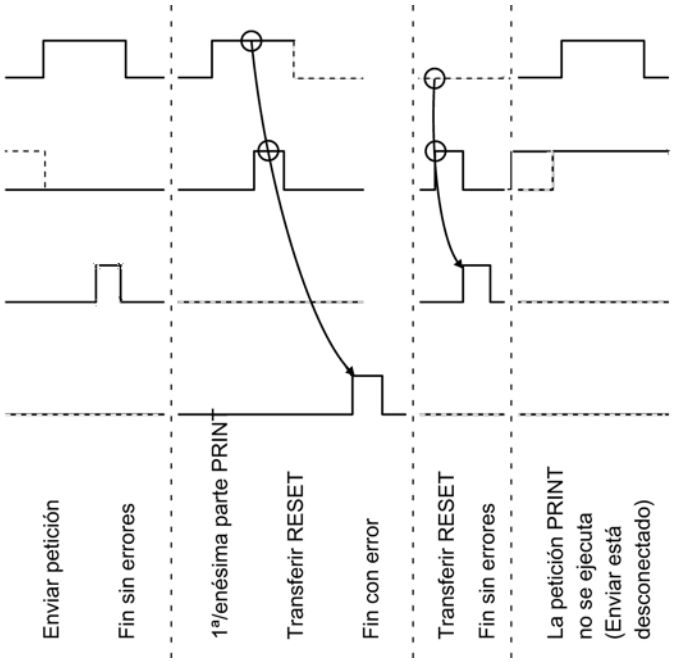
Si se sobrepasa la longitud máxima, se interrumpe el trabajo de impresión y se indica el número de evento 16#1E41 en la salida del parámetro STATUS de la instrucción P_PRT341 .

Longitud permitida

La información sobre la longitud en el DB de punteros debe adaptarse para cada modo de representación (tipos de datos) y en función de la precisión utilizada.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. No tiene que estar durante toda la transmisión en el estado lógico "1".

Regulación

Nota

La instrucción P_PRT341 no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el CP 341 pueda procesar una petición iniciada después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CP–CPU de la instrucción P_PRT341 . Una petición iniciada entretanto no se pierde. Se transfiere al CP 341 una vez concluida la coordinación de arranque.

Consulte también

Transmisión de datos con el driver de impresora (Página 1114)

V24_STAT_340 / V24_STAT: Leer señales cualificadoras en la interfaz RS232C

Descripción

La instrucción V24_STAT_340 (CP 340) / V24_STAT (CP 341) lee del CP 34x las señales cualificadoras RS 232C y las pone a disposición del usuario en los parámetros del bloque. La instrucción V24_STAT_340 / V24_STAT se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Las señales cualificadoras RS 232C se actualizan cada vez que se llama la instrucción (sondeo cíclico). El CP 34x actualiza el estado de las entradas/salidas en un tiempo de 20 ms. Con independencia de ello, las entradas/salidas se actualizan continuamente.

El resultado binario RB permanece igual. La instrucción no emite mensajes de error.

Con el parámetro LADDR se selecciona el CP 34x que se llamará.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del CP 34x La dirección inicial se toma de STEP 7.
DTR_OUT	OUTPUT	BOOL	**D**ata **t**erminal **r**eady, CP 34x operativo (salida del CP 34x)
DSR_IN	OUTPUT	BOOL	**D**ata **s**et **r**eady, interlocutor operativo (entrada del CP 34x)
RTS_OUT	OUTPUT	BOOL	**R**equest **t**o **s**end, CP 34x listo para transmitir (salida del CP 34x)
CTS_IN	OUTPUT	BOOL	**C**lear **t**o **s**end, el interlocutor puede recibir datos del CP 34x (respuesta a RTS = ON del CP 34x) (entrada del CP 34x)
DCD_IN	OUTPUT	BOOL	**D**ata **C**arrier **d**etect, nivel de señal de recepción (entrada del CP 34x)
RI_IN	OUTPUT	BOOL	**R**ing **I**ndicator, indicador de llamada (entrada del CP 34x)

Asignación en el área de datos

La instrucción V24_STAT_340 / V24_STAT no ocupa áreas de datos.

Regulación

Nota

Para detectar un cambio de señal se requiere una duración de impulso mínima. Las magnitudes decisivas son el tiempo de ciclo de la CPU, el tiempo de actualización en la CP 34x y el tiempo de reacción del interlocutor.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

V24_SET_340 / V24_SET: Escribir señales cualificadoras en la interfaz RS232C

Descripción

El usuario puede activar o desactivar las correspondientes salidas de la interfaz a través de las entradas de parámetros de la instrucción V24_SET_340 (CP 340) / V24_SET (CP 341). La instrucción V24_SET_340 / V24_SET se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

El resultado binario RB permanece igual. La instrucción no emite mensajes de error.

Con el parámetro LADDR se selecciona el CP 34x que se llamará.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del CP 34x La dirección inicial se toma de STEP 7.
RTS	INPUT	BOOL	Request to send, CP 34x listo para transmitir (controlar la salida del CP 34x)
DTR	INPUT	BOOL	Data terminal ready, CP 34x operativo (controlar la salida del CP 34x)

Asignación en el área de datos

La instrucción V24_SET_340 / V24_SETno ocupa áreas de datos.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

Parámetro STATUS

Clases de eventos

Clase de evento	Significado
0	Arranque del CP
1	Error de hardware en el CP
2	Error al inicializar
3	Error al parametrizar las instrucciones
4	Error detectado por el CP en la comunicación CP - CPU
5	Error al procesar una petición de la CPU
6	Error al procesar una petición del interlocutor (sólo con RK 512)
7	Error de transmisión
8	Error de recepción
9	Se recibió telegrama de respuesta erróneo o telegrama de error del interlocutor acoplado
10 ($0A_H$)	Errores del telegrama de respuesta del interlocutor que fueron detectados por el CP
30 ($1E_H$)	Error en la comunicación entre el CP y la CPU

Parámetro STATUS

Código de error (W#16#...)	Descripción	Solución
0003	Parámetros PtP aceptados	-
0004	Parámetros ya en el CP (sellos de tiempo idénticos)	-
0007	Cambio de estado de la CPU a STOP	-
0008	Cambio de estado de la CPU a RUN/ ARRANQUE	-
0101	Error al comprobar el EPROM del sistema operativo del CP	CP defectuoso, debe cambiarse el CP.
0102	Error al comporbar la RAM del CP	
0103	Interfaz de petición del CP defectuosa	
0110	Error en el firmware del CP	Desconecte el módulo y vuelva a conectarlo. Si es necesario, cambie el módulo.
020F	Al iniciarse la comunicación parametrizada se detecta una parametrización no válida. No se ha podido parametrizar la interfaz.	Corrija la parametrización no válida y reinicie.

Código de error (W#16#...)	Descripción	Solución
0301	Tipo de datos fuente/destino no válido o no encontrado Área (dirección inicial, longitud) no válida DB no encontrado o no válido (por ejemplo DB 0) u otro tipo de datos de fuente o destino no válido o no disponible N.º del byte de marcas de acoplamiento no válido o N.º del bit de marca de acoplamiento no válido o no está seleccionado ni 'S' ni 'F' (en la instrucción P_SND_RK)	Revise la parametrización de CPU y CP y corríjala en caso necesario. **Sólo en RK 512:** El interlocutor transmite parámetros no válidos en el encabezado del telegrama. Revise la parametrización de CPU y CP y configure un bloque en caso necesario. Consulte en las tablas de peticiones los tipos de datos permitidos. **Sólo en RK 512:** El interlocutor transmite parámetros incorrectos en el encabezado del telegrama.
0403	Tipo de datos erróneo, desconocido o no autorizado	Revise el programa, p. ej. error en la parametrización de la instrucción.
0407	Error en la transmisión de datos entre la CPU y el CP	Si el error se notifica de forma permanente, compruebe si las instrucciones a las que accede en el programa de usuario están parametrizadas correctamente. Si el error se notifica inmediatamente tras conectar la alimentación, en ese momento todavía no hay ninguna conexión establecida con la CPU. Con el procedimiento 3964(R) y con el driver ASCII, el CP 341 que recibe repite la transmisión de datos hasta que éstos hayan sido transferidos a la CPU. En caso de RK 512, la petición se confirma negativamente y debe repetirse en el programa de usuario. Si el mensaje aparece esporádicamente durante el desarrollo de una transmisión de datos, la CPU no acepta los datos temporalmente. Con el procedimiento 3964(R) y con el driver ASCII, el CP 341 que recibe repite la transmisión de datos hasta que éstos hayan sido transferidos a la CPU. En caso de RK 512, la petición se confirma negativamente y debe repetirse en el programa de usuario. Para solucionar el problema, en su programa de usuario debe acceder con más frecuencia la instrucción P_RCV_RK.
0408	Error en la transmisión de datos entre la CPU y el CP (recepción)	
	• La CPU está temporalmente sobrecargada, la petición se repite	• Reduzca el número de llamadas de comunicación
	• No es posible acceder temporalmente al área de datos de la CPU, p. ej., porque no se accede con suficiente frecuencia al bloque de recepción.	• Llame el bloque de recepción con más frecuencia

Código de error (W#16#...)	Descripción	Solución
	• No es posible acceder temporalmente al área de datos de la CPU, p. ej., porque el bloque de recepción está temporalmente bloqueado (EN=false).	• Compruebe si el bloque de recepción está bloqueado durante demasiado tiempo
0409	No es posible recibir datos. Error en la transmisión de datos entre la CPU y el CP (recepción). No es posible recibir datos. Tras varios intentos se ha interrumpido la petición al cabo de 10 s, porque	
	• no se accede al bloque de recepción.	• Compruebe en su programa de usuario si se accede al bloque de recepción.
	• el bloque de recepción está bloqueado.	• Compruebe si el bloque de recepción está bloqueado.
	• no puede accederse al área de datos de la CPU.	• Compruebe si existe el área de datos a la que van a transmitirse los datos.
	• el área de datos de la CPU es demasiado corta.	• Compruebe la longitud del área de datos.
040A	Error en la transmisión de datos entre la CPU y el CP. La transmisión de datos se ha interrumpido con RESET porque: • no existe DB de destino • el DB de destino es demasiado corto • en la instrucción está activado el bit RESET	Cree el DB de destino en el programa de usuario o prolongue el DB de destino ya existente.
0501	La petición en marcha ha sido interrumpida por un rearranque del CP.	En caso de conexión de la alimentación no hay ninguna solución. Al reparametrizar el CP desde la unidad de programación, antes de describir una interfaz debe asegurarse de que no se ejecuten más peticiones desde la CPU.
0502	La petición no es válida en este estado operativo del CP (p. ej. interfaz de dispositivo no parametrizada).	Parametrice la interfaz de dispositivo.
0505	**Sólo con driver de impresora:** Bloque de datos del sistema con textos de aviso no disponible en CP	Configure los textos de aviso con el software de parametrización y reinicie.
0506	**Sólo con driver de impresora:** Texto de aviso no disponible	Configure los textos de aviso con el software de parametrización y reinicie.
0507	**Sólo con driver de impresora:** Texto de aviso demasiado largo	Modifique el texto de aviso hasta un máximo de 150 caracteres (a un máximo de 250 caracteres en caso de variables)
0508	**Sólo con driver de impresora:** Demasiadas instrucciones de conversión	Ha configurado más instrucciones de conversión que variables. Las instrucciones de conversión sin la variable correspondiente se ignoran.

Código de error (W#16#...)	Descripción	Solución
0509	**Sólo con driver de impresora:** Demasiadas variables	Ha configurado más variables que instrucciones de conversión. Las variables sin instrucción de conversión no se imprimen.
050A	**Sólo con driver de impresora:** Instrucción de conversión desconocida	Revise la instrucción de conversión. Las instrucciones de conversión no definidas o no compatibles se sustituyen por ****** en la impresión.
050B	**Sólo con driver de impresora:** Instrucción de control desconocida	Revise la instrucción de control. Las instrucciones de control indefinidas o incompatibles se ignoran. La instrucción de control tampoco se imprime en forma de texto.
050C	**Sólo con driver de impresora:** Instrucción de conversión no ejecutable	Revise la instrucción de conversión. Las instrucciones de conversión no ejecutables se imprimen con asteriscos * de acuerdo con el ancho definido y el resto válido de la instrucción, o bien de la representación estándar.
050D	**Sólo con driver de impresora:** Ancho insuficiente o excesivo en la instrucción de conversión	Corrija el ancho especificado para la variable de la instrucción de conversión a partir del número máximo de caracteres de la variable en los modos de representación basados en texto (A, C, D, S, T, Y, Z). En la impresión sólo se imprime el número de caracteres correspondiente al ancho especificado, el resto no se imprime. En el resto de casos se imprimen asteriscos * en la cantidad equivalente al ancho.
050E	**Sólo con drivers ASCII:** Error al enviar. Los caracteres de marca de fin parametrizados no han aparecido dentro de la longitud máxima admitida o bien se ha excedido la longitud de envío máxima al generarse anexos automáticamente.	Complemente los caracteres de marca de fin en el lugar deseado del búfer de envío o seleccione una longitud de telegrama menor en la generación automática de anexos.
0514	Direcciones de inicio demasiado elevadas para el tipo de datos deseado o dirección de inicio o n.º DB/DX demasiado bajo.	Consulte en la tabla de peticiones las direcciones de inicio y los números DB/DX que pueden especificarse en el programa.
0515	**Sólo con RK 512:** Número de bit erróneo en la marca de acoplamiento.	N.º de bit permitido: de 0 a 7
0516	**Sólo con RK 512:** Número de CPU demasiado elevado.	N.º permitido de CPU: ninguno, 0, 1, 2, 3 ó 4
0517	Longitud de transmisión > 1 kByte es demasiado elevado para CP o longitud insuficiente para parámetros de interfaz.	Divida la petición en varias peticiones de una longitud menor.

Código de error (W#16#...)	Descripción	Solución
051A	**Sólo con RK 512:** Error al enviar un telegrama de orden Se ha registrado un número de error de procedimiento inmediatamente antes en STATUS.	Vea las medidas a tomar en el número de evento anterior.
051B	**Sólo con driver de impresora:** Precisión no válida	Corrija la precisión indicada en la instrucción de conversión. La precisión siempre va precedida de un punto que identifica y limita el ancho (p. ej.: ".2" para la impresión del punto decimal y 2 decimales). La precisión sólo es relevante en los modos de representación F, R, A y D. En el resto de casos se ignora.
051C	**Sólo con driver de impresora:** Variable no válida (Longitud de variable incorrecta/tipo incorrecto)	Corrija la variable indicada. Consulte los tipos de datos válidos para el modo de representación en cuestión en la tabla respectiva.
051E	**Sólo con driver de impresora:** Las "secuencias de fin de línea" enviadas con esta petición (p. ej.: $R / $L / $N) (ya) no encajan en la página (empezada)	Aumente la longitud de la página, reduzca la cantidad de líneas (o los saltos de línea) o reparta la impresión en varias páginas
0601	Error en el 1er byte de orden (excepto 00 ó FFH)	Error de estructura del encabezado en el interlocutor. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0602	Error en el 3er byte de orden (excepto A, 0 ó E)	Error de estructura del encabezado en el interlocutor. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0603	Error en el 3er byte de orden en telegramas de continuación (orden diferente a la del 1er telegrama).	Error de estructura del encabezado en el interlocutor. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0604	Error en el 4º byte de orden (letra de orden incorrecta)	Error de estructura del encabezado en el interlocutor o se ha solicitado una combinación de órdenes no autorizada en el CP. Compruebe las órdenes permitidas. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0605	Error en el 4º byte de orden en el telegrama de continuación (la orden no es como en el 1er telegrama)	Error de estructura del encabezado en el interlocutor. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0606	Error en el 5º byte de orden (número de DB no permitido)	Consulte los números de DB, las direcciones de inicio y las longitudes válidas en las tablas de peticiones.
0607	Error en el 5º ó 6º byte de orden (dirección de inicio demasiado elevada)	Consulte los números de DB, las direcciones de inicio y las longitudes válidas en las tablas de peticiones.
0608	Error en el 7º u 8º byte de orden (longitud no válida)	Consulte los números de DB/DX, las direcciones de inicio y las longitudes válidas en las tablas de peticiones.
0609	Error en el 9º y 10º byte de orden (marca de coordinación no válida en este tipo de datos o número de bit demasiado elevado)	Error de estructura del encabezado en el interlocutor. Consulte en las tablas de peticiones los casos en los que se permiten utilizar las marcas de coordinación.
060A	Error en el 10º byte de orden (número de CPU no válido)	Error de estructura de encabezado en el interlocutor
060B	Telegrama SEND de mayor/menor longitud que la esperada (se han recibido más/menos datos que los anunciados en el encabezado del telegrama).	Es necesaria una corrección en el interlocutor
060C	Recibido telegrama de orden FETCH con datos útiles.	Es necesaria una corrección en el interlocutor
060D	EL CP ha recibido el telegrama en un modo de operación no válido:	
	• El enlace de recepción entre la CPU y el CP no está estructurado correctamente	• Compruebe si la conexión que se intenta establecer está (bien) parametrizada
	• El CP todavía no ha arrancado por completo	• Este aviso de error sólo puede aparecer durante el arranque del CP. Repita la petición.
	• La CPU receptora se encuentra en estado operativo STOP	• Ponga de nuevo la CPU en estado operativo RUN y repita la petición.
	• La interfaz solicitada se está reparametrizando es este momento	• Se trata de un error temporal. Repita la petición.

Código de error (W#16#...)	Descripción	Solución
060E	Error de sincronización del interlocutor • Se ha recibido un nuevo telegrama de orden (de continuación) antes de haber enviado un telegrama de reacción. • 1. Se esperaba 1er telegrama de orden y se ha recibido un telegrama de continuación. • Se esperaba telegrama de orden de continuación y se ha recibido el 1er telegrama.	Este error puede indicarse tras un reinicio del autómata propio en caso de telegramas de gran longitud o de reinicio del interlocutor. En estos casos se trata de un comportamiento de arranque normal del sistema. Durante el funcionamiento normal, el error también puede aparecer como consecuencia de estados de error que sólo ha detectadoel interlocutor. En cualquier otro caso debe partir de un comportamiento erróneo del dispositivo interlocutor. Es posible que el error no se produzca en peticiones < 128 bytes.
060F	DB bloqueado por función de coordinación	En el propio programa: Desactive la marca de acoplamiento después de procesar los últimos datos de transmisión. En el programa del interlocutor: Repetición de la petición
0610	Se ha recibido un telegrama de longitud insuficiente (longitud < 4 bytes para telegramas de continuación o reacción o longitud < 10 bytes para telegramas de orden)	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0611	La longitud del telegrama y la longitud especificada en el encabezado de telegrama no coinciden.	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0612	Error al enviar el telegrama de reacción (de continuación), se ha registrado un número de error de procedimiento inmediatamente antes en STATUS.	Vea las medidas de solución correspondientes al número de error registrado inmediatamente antes en STATUS.
0701	Envío del primer reintento: • Se ha detectado un error durante el envío del telegrama o bien • El interlocutor ha solicitado una repetición con un carácter de confirmación negativo (NAK).	Un reintento no es un error, pero puede ser un indicio de que hay interferencias en la línea de transmisión o de que el interlocutor no funciona correctamente. Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se registra un n.º de error que describe el error que apareció en primer lugar.
0702	Sólo con 3964(R): Error al establecer la conexión: Tras el envío de STX se ha recibido NAK o un carácter cualquiera (excepto DLE o STX).	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0703	**Sólo con 3964(R):** Tiempo de retardo de acuse (TRA) transcurrido: Tras en envío de STX no ha llegado ninguna respuesta del interlocutor dentro del tiempo de retardo de acuse.	El dispositivo interlocutor es demasiado lento o no está preparado para recibir o bien hay p. ej. una rotura de la línea de envío. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0704	**Sólo con 3964(R):** Cancelaciónpor parte del interlocutor: Durante el proceso de envío en curso el interlocutor ha recibido uno o varios caracteres.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0706	**Sólo con 3964(R):** Error al terminar la conexión: • El telegrama ha sido rechazado por el interlocutor con NAK u otro carácter al final (con la excepción de DLE) o bien • El carácter de confirmación (DLE) se ha recibido demasiado pronto.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0707	**Sólo con 3964(R):** Tiempo de retardo de acuse al final de la conexión/tiempo de vigilancia de la respuesta excedido tras el telegrama de transmisión: Tras deshacer la conexión con DLE ETX no ha llegado ninguna respuesta del interlocutor dentro del TRA.	El dispositivo interlocutor es demasiado lento o está defectuoso. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0708	**Sólo para driver ASCII y driver de impresora:** El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor está averiado, es demasiado lento o está offline. Compruebe el interlocutor o modifique la parametrización en caso necesario.
0709	No es posible establecer la conexión, se ha sobrepasado el número de intentos de establecimiento permitido.	Revise el cable de la interfaz o los parámetros de la transmisión. Compruebe también en el interlocutor si la función de recepción entre la CPU y el CP está correctamente parametrizada.
070A	Los datos no han podido ser transmitidos, se ha sobrepasado el número de intentos de transmisión permitido.	Revise el cable de la interfaz o los parámetros de la transmisión.

Código de error (W#16#...)	Descripción	Solución
0801	Esperando al primer reintento: Al recibir un telegrama se ha detectado un error y el CP exige mediante un acuse negativo (NAK) un reintento del interlocutor.	Un reintento no es un error, pero puede ser un indicio de que hay interferencias en la línea de transmisión o de que el interlocutor no funciona correctamente. Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se registra un n.º de error que describe el error que apareció en primer lugar.
0802	**Sólo con 3964(R):** Error al establecer la conexión: • En posición de reposo se han recibido uno o más caracteres (cualquiera con la excepción de NAK o STX) o bien • Tras la recepción de un STX el interlocutor ha enviado otros caracteres sin esperar la recepción de la respuesta DLE. Tras la conexión de la alimentación del interlocutor: • Durante el tiempo de conexión del interlocutor, el CP recibe un carácter indefinido.	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0805	**Sólo con 3964(R):** Error lógico durante la recepción: Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado del telegrama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Compruebe el comportamiento erróneo del dispositivo interlocutor con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0806	Transcurrido el tiempo de retardo caracteres (TRC): • No se han recibido dos caracteres consecutivos dentro del TRC o bien Sólo con 3964(R): • 1. El 1er carácter tras el envío de DLE en el establecimiento de la conexión no se ha recibido dentro del TRC.	El dispositivo interlocutor es demasiado lento o está defectuoso. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0808	**Sólo con 3964(R):** Error en el carácter de comprobación de bloque BCC: El valor del BCC determinado internamente no concuerda con el BCC recibido por el interlocutor al final de la conexión.	Compruebe si la conexión tiene interferencias importantes; en este caso, ocasionalmente también pueden aparecer códigos de error. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
080A	No hay un búfer de recepción libre disponible: Durante la recepción no se disponía de ningún búfer de recepción vacío.	La instrucción P_RCV_RK se debe llamar con mayor frecuencia.
080C	Error de transmisión: • Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de desbordamiento). **Sólo con 3964(R):** • Si durante la posición de reposo se recibe un carácter defectuoso, el error se indica inmediatamente para facilitar la detección prematura de interferencias en la línea de transmisión. **Sólo con RK 512 y 3964(R):** • Si esto ocurre durante el modo de transferencias o recepción, se inician repeticiones.	Las interferencias en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El riesgo de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe la línea de conexión del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos equipos.
080D	BREAK: La línea de recepción que llega al interlocutor está interrumpida.	Restablezca la conexión o conecte el interlocutor.
0815	La parametrización del número de intentos de transmisión del CP y del interlocutor no concuerda.	Parametrice en el interlocutor el mismo número de intentos de transmisión que en el CP. Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0816	La longitud de un telegrama recibido era superior a la longitud máxima parametrizada.	Es necesaria una corrección en el interlocutor.
0818	**Sólo con driver ASCII:** DSR = OFF o CTS = OFF	El interlocutor ha puesto todas las señales DSR o CTS a "OFF" antes o durante una transmisión. Compruebe el control de las señales cualificadoras RS 232C en el interlocutor.
0902	**Sólo con RK 512:** Error de acceso a la memoria en el interlocutor (memoria no disponible) **Con SIMATIC S5 como interlocutor:** • Área incorrecta en la palabra indicadora o • área de datos no disponible (excepto DB/DX) o • área de datos demasiado corta (excepto DB/DX)	Compruebe si el área de datos en cuestión existe en el interlocutor y si es suficientemente grande o revise los parámetros del bloque de función de sistema llamado. Compruebe también la longitud indicada en el bloque de función del sistema.

Código de error (W#16#...)	Descripción	Solución
0903	**Sólo con RK 512:** Error de acceso DB/DX en el interlocutor (DB/DX no disponible o demasiado corto) **Con SIMATIC S5 como interlocutor:** • DB/DX no disponible o • DB/DX demasiado corto o • N.º DB/DX no válido Se ha sobrepasado el área de origen permitida en la petición FETCH	Compruebe si el área de datos en cuestión existe en el interlocutor y si es suficientemente grande o revise los parámetros del bloque de función de sistema llamado. Compruebe también la longitud indicada en el bloque de función del sistema.
0904	**Sólo con RK 512:** El interlocutor notifica: "Tipo de petición no permitido".	Comportamiento erróneo del interlocutor, ya que el CP nunca envía una orden del sistema.
0905	**Sólo con RK 512:** Error en el interlocutor o con SIMATIC S5 como interlocutor: • Tipo de origen/destino no válido o bien • Error de memoria en el autómata del interlocutor o bien • Error de entendimiento CP/CPU en el interlocutor o bien • el autómata del interlocutor se encuentra en estado STOP	Compruebe si el interlocutor puede transmitir el tipo de datos en cuestión. Compruebe la instalación del hardware del interlocutor. Conmute el selector del autómata interlocutor a la posición RUN.
0908	**Sólo con RK 512:** El interlocutor ha detectado un error de sincronización: El orden de los telegramas es incorrecto.	Este error se produce durante el rearranque completo del propio autómata o del del interlocutor. Se trata de un comportamiento normal de arranque de la instalación. No es necesario tomar ninguna medida. Durante el funcionamiento, este error también puede producirse como consecuencia de errores anteriores. De lo contrario, puede partir de la base de un comportamiento erróneo del interlocutor.
0909	**Sólo con RK 512:** DB/DX se encuentra bloqueado en el interlocutor por marcas de coordinación	En el programa del interlocutor: Desactive la marca de coordinación después de procesar los últimos datos de transmisión. En el programa: Repita la petición.
090A	**Sólo con RK 512:** El interlocutor ha detectado errores en el encabezado del telegrama: el 3er byte de orden en el encabezado es erróneo	Compruebe si el error se debe a interferencias o a un comportamiento erróneo del interlocutor. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
090B	**Sólo con RK 512:** Error en el encabezado del telegrama: 1. ó 4º byte de orden en el encabezado es erróneo	Compruebe si el error se debe a interferencias o a un comportamiento erróneo del interlocutor. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
090C	**Sólo con RK 512:** El interlocutor detecta una longitud de telegrama errónea (longitud total).	Compruebe si el error se debe a interferencias o a un comportamiento erróneo del interlocutor. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
090D	**Sólo con RK 512:** Hasta el momento no se ha llevado a cabo el rearranque completo del interlocutor.	Lleve a cabo el rearranque completo del autómata del interlocutor o ponga el selector de modo del CP en posición RUN.
090E	**Sólo con RK 512:** Se ha recibido un número de error desconocido en el telegrama de reacción.	Compruebe si el error se debe a interferencias o a un comportamiento erróneo del interlocutor. Utilice para ello un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0A01	**Sólo con RK 512:** Error de sincronización del interlocutor porque • telegrama de reacción sin petición • telegrama de reacción recibido antes de que se enviara el telegrama de continuación • tras enviar un primer telegrama se ha recibido un telegrama de reacción de continuación • tras enviar un telegrama de continuación se ha recibido el primer telegrama de reacción	Este error aparece tras un rearranque completo del propio autómata en telegramas de gran longitud o en caso de un rearranque completo del interlocutor. Se trata de un comportamiento normal de la instalación, no debe tomar ninguna medida. Durante el funcionamiento normal, el error también puede aparecer como consecuencia de estados de error que sólo ha detectado el interlocutor. En cualquier otro caso puede partir de un comportamiento erróneo del dispositivo interlocutor. Es posible que el error no se produzca en peticiones < 128 bytes.
0A02	Sólo con RK 512: Se han localizado errores en la estructura del telegrama de reacción recibido (1er byte diferente de 00 o FF)	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0A03	Sólo con RK 512: El telegrama de reacción recibido contiene demasiados o bien muy pocos datos.	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.
0A04	Sólo con RK 512: El telegrama de reacción a una petición SEND ha llegado con datos.	Si fuese necesario, compruebe el comportamiento erróneo del equipo interlocutor con un dispositivo de comprobación de interfaces que se conecta a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0A05	Sólo con RK 512: No se ha recibido ningún telegrama de reacción del interlocutor dentro del tiempo de vigilancia.	¿Es el interlocutor un dispositivo muy lento? Este error se presenta a menudo como consecuencia de un error que se ha producido con anterioridad. Por ejemplo, pueden visualizarse errores de recepción de procedimiento (clase de evento 8) después de enviar un telegrama FETCH. Motivo: no se ha podido recibir el telegrama de reacción porque se han detectado interferencias. El tiempo de vigilancia ha expirado. Este error también puede producirse cuando se rearranca el interlocutor, antes de que éste responda al último telegrama FETCH recibido.
1E0D	Cancelación de la petición por rearranque completo, rearranque o reset	
1E0E	Error estático al llamar la SFC RD_REC. El valor de retorno RET_VAL de la SFC queda disponible en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E0F	Error estático al llamar la SFC WR_REC. El valor de retorno RET_VAL de la SFC queda disponible en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E41	El número de bytes indicado en el parámetro LEN de las instrucciones no está permitido	Respete el rango de 1 a 4096 bytes.
1E42	Instrucción P_PRINT_RK: El número de bytes indicado para la variable o la cadena de formato dentro de la longitud del DB de punteros no es válido.	Observe las longitudes permitidas: 32 bytes para variables, 150 bytes para cadena de formato
1E43	Instrucción P_PRINT_RK: No existe ningún puntero para la cadena de formato.	Indique en el DB de punteros el n.º del bloque de datos y el nº de la palabra de datos para la cadena de formato.

Nota

Sólo se envía un mensaje de error si simultáneamente está activado el bit ERROR (petición interrumpida con errores). En cualquier otro caso la palabra de STATUS es cero.

Llamar la variable SFCERR o SFCSTATUS

Obtendrá más información sobre el error producido, 14 (1E0EH) y 15 (1E0FH), de la clase de evento 30 mediante la variable SFCERR o SFCSTATUS.

Puede cargar la variable SFCERR o SFCSTATUS sólo desde un acceso simbólico al DB instancia de la instrucción correspondiente.

Los mensajes de error que se registran en la variable SFCERR se encuentran en las funciones de sistema SFC 58 "WR_REC: Escribir registro en periferia (Página 6774)" y SFC 59 "RD_REC: Leer registro de periferia (Página 6769)".

Los mensajes de error que se registran en la variable SFCSTATUS se encuentran en las funciones de sistema SFB 52 "RDREC: Leer registro (Página 6735)" y SFB 53 "WRREC: Escribir registro (Página 6738)".

Acoplamiento PtP CP 440

SEND_440: Enviar datos

Descripción

La instrucción SEND_440 transmite un conjunto de datos de un bloque de datos especificado por los parámetros DB_NO, DBB_NO y LEN, al CP 440. La instrucción SEND_440 se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción SEND_440 puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al CP 440 y la instrucción SEND_440 pasa al estado inicial. Los datos que ya ha recibido el CP 440 se envían al interlocutor. Si la entrada R tiene el estado lógico estático "1", la transmisión está desactivada.

Con el parámetro LADDR se indica la dirección del CP 440 que debe llamarse.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción SEND_440 . Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección base del CP 440 La dirección inicial se toma de STEP 7.

Parámetro	Declaración	Tipo de datos	Descripción
DB_NO	INPUT	INT	Número del bloque de datos Nº de DB de envío: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos 0 ≤ DBB_NO ≤ 8190 datos de transmisión a partir del byte de datos; Offset es específico de la CPU
LEN	INPUT	INT	Longitud de datos 1 ≤ LEN ≤ 400, indicados en número de bytes
DONE[1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7960) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7960) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

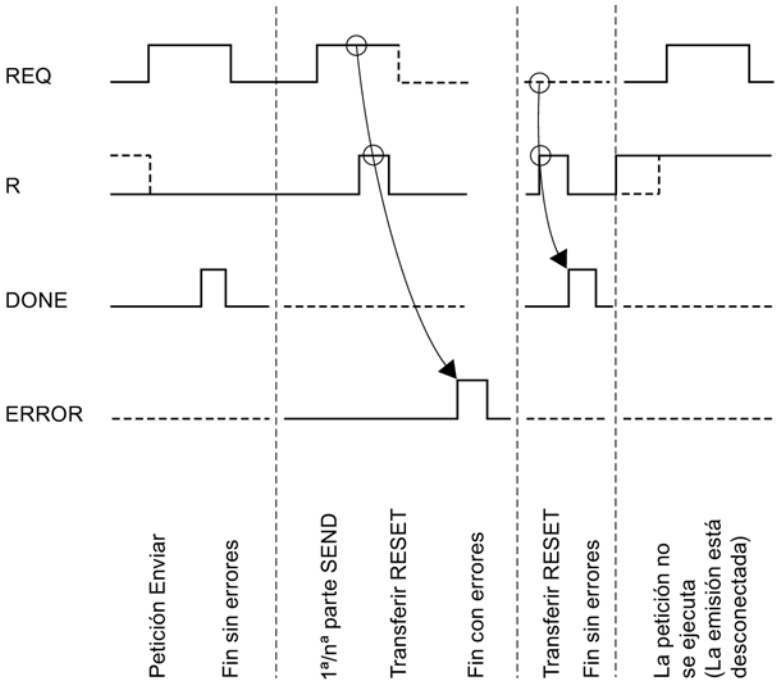
La instrucción SEND_440 opera conjuntamente con un DB instancia I_SEND_440. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, (STATUS == W#16#1E0F), encontrará una indicación más precisa sobre el error en la variable SFCERR.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción SEND_440 no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

RECV_440: Recibir datos

Descripción

La instrucción RECV_440 transmite datos del CP 440 a un área de datos S7, especificada por los parámetros DB_NO, DBB_NO y LEN. La instrucción RECV_440 se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo .

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del CP 440. Es posible cancelar una transmisión en curso con el estado lógico "0" en el parámetro EN_R. La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desactivada mientras exista el estado lógico "0" en el parámetro EN_R. Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción RECV_440 pasa al estado inicial. La recepción está desactivada mientras exista el estado lógico "1" en el parámetro R.

Con el parámetro LADDR se selecciona el CP 440 que se llamará.

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. NDR y ERROR/STATUS también se indican en caso de RESET de la instrucción RECV_440 (parámetro LEN == 16#00). Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitar lectura de datos
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Bloqueo de recepción.
LADDR	INPUT	INT	Dirección inicial del CP 440 La dirección base se toma de la tabla de configuración de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º del DB de recepción: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número del byte de datos El offset es específico de la CPU
NDR[1]	OUTPUT	BOOL	Petición finalizada sin errores, datos aceptados Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7960) contiene la información de error.
LEN[1]	OUTPUT	INT	Longitud del telegrama recibido 1 ≤ LEN ≤ 400, indicados en número de bytes

Parámetro	Declaración	Tipo de datos	Descripción
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7960) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

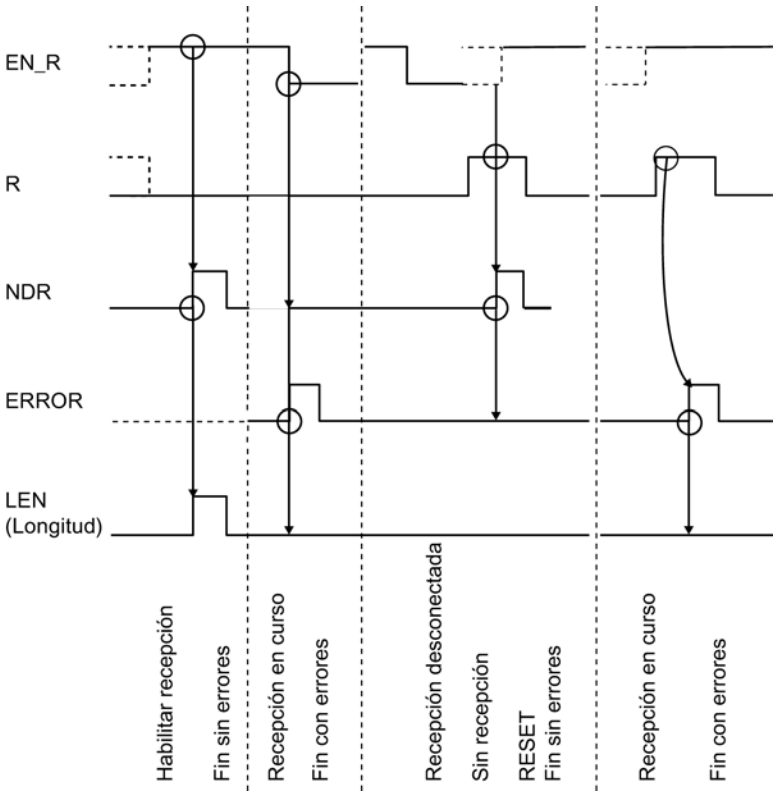
La instrucción RECV_440 opera conjuntamente con un DB instancia I_RECV_440. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: En caso de error, (STATUS == W#16#1E0E), encontrará una indicación más precisa sobre el error en la variable SFCERR.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros NDR, LEN y ERROR en función de la conexión de las entrada EN_R y R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Regulación

Nota

La instrucción RECV_440 no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

RES_RECV: Borrado del búfer de recepción

Descripción

La instrucción RES_RECV borra completamente el búfer de recepción del CP 440. Todos los telegramas guardados se rechazan. Un telegrama entrante en el momento de llamar a la instrucción RES_RECV se guarda.

Funcionamiento

Con un flanco ascendente en la entrada REQ se activa la instrucción. La petición puede recorrer varias llamadas (ciclos de programa).

La instrucción RES_RECV puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De esta forma se interrumpe el proceso de borrado y la instrucción RES_RECV pasa al estado inicial.

Con el parámetro LADDR se indica la dirección del CP 440 que debe llamarse.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se visualizan en caso de RESET de la instrucción RES_RECV . Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del CP 440 La dirección inicial se toma de STEP 7.
DONE[1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00;
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 7960) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 7960) figura la información de error.
[1] El parámetro está disponible hasta la siguiente llamada de la instrucción.			

Asignación en el área de datos

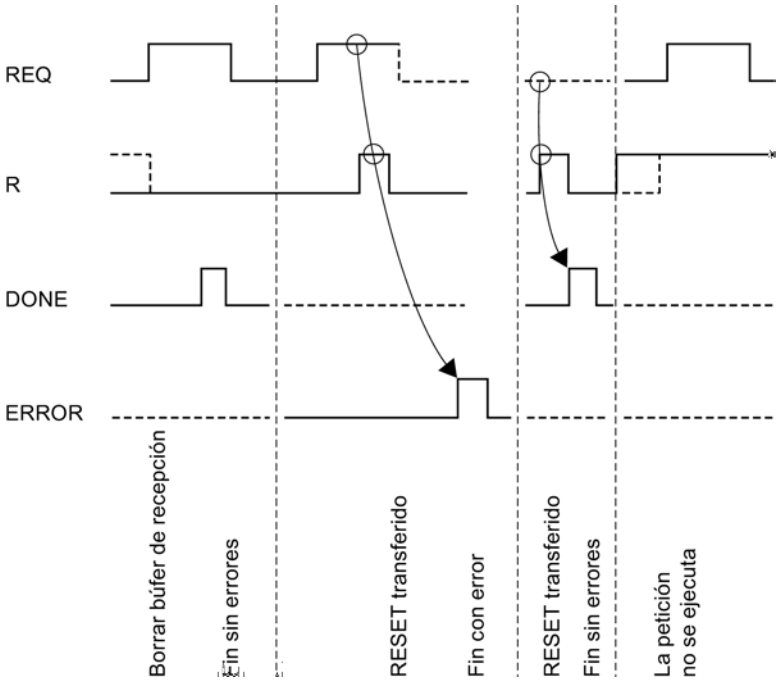
La instrucción RES_RECV opera conjuntamente con un DB instancia I_RES_RECV. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, (STATUS == W#16#1E0F), encontrará una indicación más precisa sobre el error en la variable SFCERR.

Cronograma

La siguiente figura muestra el comportamiento de los parámetros DONE y ERROR en función de la conexión de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción RES_RECV no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Parámetro STATUS

Clases de evento

Clase de evento	Significado
0 (0_H)	Arranque del CP
2 (02_H)	Error al inicializar
5 (05_H)	Error al procesar una petición de la CPU
7 (07_H)	Error de transmisión
8 (08_H)	Error de recepción
11 ($0B_H$)	Advertencias
30 ($1E_H$)	Error en la comunicación entre el CP y la CPU
128 (80_H)	Fallo de módulo

Parámetro STATUS

Código de error (W#16#...)	Descripción	Solución
0001	Init del CP concluida	-
0003	Parámetros PtP aceptados	-
0004	Parámetros ya en el CP (estados de tiempo idénticos)	-
0007	Transición de estado de la CPU a STOP	-
0008	Transición de estado de la CPU a RUN/ ARRANQUE	-
0201	No hay parámetros disponibles Memoria de parámetros vacía o contenido desconocido	Cargue los parámetros de la interfaz.
0204	Trama de caracteres no permitida	Corrija la parametrización no permitida, cargue los parámetros en el módulo y realice un rearranque completo.
0205	Velocidad de transmisión no permitida	Corrija la parametrización no permitida, cargue los parámetros en el módulo y realice un rearranque completo.
020F	Al iniciarse la comunicación parametrizada se detecta una parametrización no válida. No se ha podido parametrizar la interfaz.	Corrija la parametrización no válida y realice un rearranque completo.
0212	No permitido • Tiempo de vigilancia • Tiempo de retardo de caracteres Sólo con 3964(R): • Tiempo de retardo de confirmación	Corrija la parametrización no permitida, cargue los parámetros en el módulo y realice un rearranque completo.

Código de error (W#16#...)	Descripción	Solución
0501	La petición en curso ha sido interrumpida por un rearranque del CP.	En caso de conexión (POWER ON) no hay solución. Al reparametrizar el CP desde la unidad de programación, debe asegurarse, antes de escribir una interfaz, de que no se ejecuten más peticiones desde la CPU.
0502	La petición no está permitida en este estado operativo del CP (p. ej. interfaz del dispositivo no parametrizada).	Parametrice la interfaz del dispositivo.
050E	• Longitud del telegrama no válida o	• La longitud del telegrama es > 400 bytes. Seleccione una longitud menor o
	• Los caracteres de marca de fin parametrizados no se encuentran dentro de la máxima longitud permitida.	• Complete los caracteres de marca de fin en el búfer de emisión en el lugar deseado.
0517	La longitud de transmisión > 400 bytes es demasiado grande para el CP	Divida la petición en varias peticiones de una longitud menor.
051D	Petición de envío/recepción interrumpida por • reset del bloque de comunicación • reparametrización	Repita la llamada al bloque del comunicación.
0522	Se ha iniciado una nueva petición SEND aunque la petición antigua aún no ha finalizado.	No inicie la nueva petición SEND hasta que la petición antigua haya finalizado con DONE o ERROR.
0701	Sólo con 3964(R): Envío del primer reintento: • Se ha detectado un error durante el envío del telegrama o bien • El interlocutor ha solicitado una repetición con un carácter de confirmación negativo (NAK).	Un reintento no es un error, pero puede ser un indicio de que hay interferencias en la línea de transmisión o de que el interlocutor no funciona correctamente. Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se registra un número de error que describe el error que apareció en primer lugar.
0702	Sólo con 3964(R): Error al establecer la conexión: Tras el envío de STX se ha recibido NAK o un carácter cualquiera (excepto DLE o STX).	Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0703	Sólo con 3964(R): Tiempo de retardo de acuse (TRA) transcurrido: Tras en envío de STX no ha llegado ninguna respuesta del interlocutor dentro del tiempo de retardo de acuse.	El dispositivo interlocutor es demasiado lento o no está preparado para recibir o bien hay p. ej. una rotura de la línea de envío. Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0704	Sólo con 3964(R): Interrupción por parte del interlocutor: Durante el proceso de envío en curso el interlocutor ha recibido uno o varios caracteres.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias graves o el interlocutor no funciona correctamente. Dado el caso, verifíquelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0705	Sólo con 3964(R): Acuse negativo durante el envío	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias graves o el interlocutor no funciona correctamente. Dado el caso, verifíquelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0706	Sólo con 3964(R): Error en el fin del enlace: • El telegrama ha sido rechazado por el interlocutor con NAK u otro carácter al final (con la excepción de DLE) o bien • El carácter de confirmación (DLE) se ha recibido demasiado pronto.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias graves o el interlocutor no funciona correctamente. Dado el caso, verifíquelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0707	Sólo con 3964(R): Tiempo de retardo de acuse al final del enlace/tiempo de vigilancia de respuesta tras el telegrama de envío agotados: Tras la interrupción de la conexión con DLE ETX no ha llegado ninguna respuesta del interlocutor dentro del TRA.	El interlocutor es demasiado lento o es defectuoso. Dado el caso, verifíquelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0708	Sólo con driver ASCII: El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor tiene un defecto, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
0709	Sólo con 3964(R): No es posible establecer la conexión, se ha sobrepasado el número de intentos de establecimiento permitido.	Revise el cable de la interfaz o los parámetros de la transmisión. Compruebe también en el interlocutor si la función de recepción entre la CPU y el CP está correctamente parametrizada.
070A	Sólo con 3964(R): Los datos no han podido ser transmitidos, se ha sobrepasado el número de intentos de transmisión permitido.	Revise el cable de la interfaz o los parámetros de la transmisión.

Código de error (W#16#...)	Descripción	Solución
070B	Sólo con 3964(R): El conflicto de inicialización no puede resolverse, puesto que ambos interlocutores tienen predeterminada prioridad alta.	Cambie la parametrización.
070C	Sólo con 3964(R): El conflicto de inicialización no puede resolverse, puesto que ambos interlocutores tienen predeterminada prioridad baja.	Cambie la parametrización.
0801	Sólo con 3964(R): Esperando al primer reintento: Al recibir un telegrama se ha detectado un error y el CP exige mediante un acuse negativo (NAK) un reintento del interlocutor.	Un reintento no es un error, pero puede ser un indicio de que hay interferencias en la línea de transmisión o de que el interlocutor no funciona correctamente. Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se notifica un número de error que describe el error que se produjo en primer lugar.
0802	Sólo con 3964(R): Error al establecer la conexión: • En posición de reposo se han recibido uno o más caracteres (cualquiera con la excepción de NAK o STX) o bien • Tras la recepción de un STX el interlocutor ha enviado otros caracteres sin esperar la recepción de la respuesta DLE. Tras la conexión de la alimentación del interlocutor: • Durante el tiempo de conexión del interlocutor, el CP recibe un carácter indefinido.	Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0805	Sólo con 3964(R): Error lógico durante la recepción: Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado del telegrama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0806	Transcurrido el tiempo de retardo caracteres (TRC): • No se han recibido dos caracteres consecutivos dentro del TRC o bien Sólo con 3964(R): • 1. El 1er carácter tras el envío de DLE en el establecimiento de la conexión no se ha recibido dentro del TRC.	El interlocutor es demasiado lento o es defectuoso. Dado el caso, verifíquelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0807	Longitud de telegrama no permitida: Se ha recibido un telegrama de longitud 0.	Laz recepción de un telegrama de longitud 0 no es un error. Compruebe por qué el interlocutor envía telegramas sin datos útiles.
0808	Sólo con 3964(R): Error en el carácter de comprobación de bloque BCC: El valor del BCC determinado internamente no concuerda con el BCC recibido por el interlocutor al final de la conexión.	Compruebe si el enlace tiene grandes interferencias; en este caso, ocasionalmente también pueden aparecer códigos de error. Compruebe el comportamiento erróneo del dispositivo interlocutor, en caso necesario con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0809	Sólo con 3964(R): Agotado tiempo de espera a repetición de bloque	Parametrice en el interlocutor el mismo tiempo de espera de bloques que en el CP 440. Compruebe el comportamiento erróneo del interlocutor, en caso necesario con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
080A	No hay ningún búfer de recepción libre: Durante la recepción no se disponía de ningún búfer de recepción vacío.	La instrucción RECV_440 se debe llamar con más frecuencia.
080C	Error de transmisión: • Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de rebosamiento). Sólo con 3964(R): • Si durante la posición de reposo se recibe un carácter defectuoso, el error se indica inmediatamente para facilitar la detección prematura de fallos en la línea de transmisión. Sólo con 3964(R): • Si esto ocurre durante el modo de envío o recepción, se inician repeticiones.	Los fallos en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El aumento de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos.
080D	BREAK: La línea de recepción hacia el interlocutor está interrumpida.	Restablezca la conexión o conecte al interlocutor.

Código de error (W#16#...)	Descripción	Solución
080E	Desbordamiento de búfer de recepción con control de flujo no habilitado.	En el programa de usuario debe llamarse con más frecuencia la instrucción para la recepción, o bien debe parametrizarse una comunicación con control de flujo.
0810	Error de paridad	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos.
0811	Error de trama de caracteres	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
0812	Sólo con driver ASCII: Después de que el CP enviara XOFF o pusiera CTS a OFF, se recibieron más caracteres.	Vuelva a parametrizar el interlocutor o procure una lectura más rápida de los datos en el CP.
0814	Sólo con driver ASCII: Se han perdido uno o varios telegramas, puesto que se ha trabajado sin control de flujo.	Siempre que sea posible, trabaje con control de flujo. Utilice el búfer de recepción completo. En los parámetros predeterminados, configure el parámetro "Reacción a CPU-STOP" predeterminando "Continuar funcionamiento".
0816	La longitud de un telegrama recibido era superior a la longitud máxima parametrizada	Es necesaria una corrección en el interlocutor
0820	Error estático al llamar la SFC RD_REC. Puede disponer del valor de retorno RET_VAL de la SFC en la variable SFCERR del DB de instancia para su evaluación.	Analice la variable SFCERR del DB de instancia.
0B01	El búfer de recepción está lleno en más de 2/3 partes	Llame con más frecuencia al bloque de recepción, para evitar el desbordamiento del búfer de recepción.
8000	No existe firmware del módulo	Realice una actualización de firmware.
1E0D	• Interrupción de petición por rearranque completo, rearranque, reset, o	• Repita la petición.
	• porque no existe el módulo.	• Compruebe si está enchufado el módulo.
1E0E	Error estático al llamar la SFC RD_REC. El valor de retorno RET_VAL de la SFC queda a disposición en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.

Código de error (W#16#...)	Descripción	Solución
1E0F	Error estático al llamar la SFC WR_REC. El valor de retorno RET_VAL de la SFC queda disponible en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E41	El número de bytes indicado en el parámetro LEN de las instrucciones no está permitido.	Respete el rango de valores de 1 a 400 bytes.

Nota

Sólo se envía un mensaje de error si simultáneamente está activado el bit ERROR (petición interrumpida con errores). En cualquier otro caso la palabra de STATUS es cero.

Llamada a variable SFCERR

Puede obtener más información sobre el error producido, 14 (1E0EH) y 15 (1E0FH) de la clase de evento 30, a través de la variable SFCERR.

La variable SFCERR se puede cargar desde el DB instancia de la instrucción correspondiente.

Los mensajes de error que se registran en la variable SFCERR se encuentran en las funciones de sistema SFC 58 "WR_REC: Escribir registro en periferia (Página 6774)" y SFC 59 "RD_REC: Leer registro de periferia (Página 6769)".

Acoplamiento PtP CP 441

Instrucciones de comunicación

Introducción

El CP 441 no sólo se comunica a través de bloques propios sino también a través de instrucciones del sistema de automatización S7-400.

El sistema de automatización S7-400 pone a disposición una serie de instrucciones que inician y controlan la comunicación entre la CPU y el procesador de comunicaciones CP 441 en el programa de usuario. Las instrucciones están almacenadas de forma fija en la memoria de la CPU.

Instrucciones del sistema de automatización S7-400

La tabla siguiente muestra las instrucciones del sistema de automatización S7-400 que están disponibles para la comunicación entre la CPU y el CP 441.

Tabla 9-323 Instrucciones del sistema de automatización S7-400

SFB	Significado
BSEND	La instrucción BSEND permite enviar datos desde un área de datos S7 a un interlocutor con una indicación de destino fija.
BRCV	La instrucción BRCV permite recibir datos de un interlocutor y depositarlos en área de datos S7.
FETCH	Sólo con RK 512: La instrucción FETCH permite recuperar datos de un interlocutor.
PUT	Sólo con RK 512: la instrucción PUT permite enviar datos a un interlocutor con una indicación de destino modificable dinámicamente.
PRINT	La instrucción PRINT permite imprimir un texto de aviso con hasta cuatro variables.
STATUS	La instrucción STATUS permite consultar el estado de una conexión.

Información adicional

Para obtener un descripción detallada sobre las instrucciones del sistema de automatización S7-400 y sus parámetros, consulte: Sinopsis de las instrucciones para la comunicación S7 (Página 7666)

Consulte también

Transferencia de datos de la CPU al módulo de comunicación con BSEND (CP 441-2) (Página 1137)

Transferencia de datos del módulo de comunicación a la CPU con BRCV (CP 441-2) (Página 1139)

Recoger datos con RK512 (Página 1109)

V24_STAT_441: Leer señales cualificadoras en la interfaz RS232C

Descripción

La instrucción V24_STAT_441 lee de una interfaz del CP 441 las señales cualificadoras RS 232C y las pone a disposición del usuario en los parámetros del bloque. La instrucción V24_STAT_441 se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Las señales cualificadoras RS 232C se actualizan cada vez que se llama la instrucción (sondeo cíclico). La interfaz se selecciona indicando en la instrucción V24_STAT_441 como ID la "ID local" de una de las conexiones que pasan por esta interfaz.

El resultado binario RB permanece igual.

Nota

Una tensión de entrada positiva en las señales de entrada DSR, CTS, DCD y RI del RS232 C se reproduce en el estado lógico "1" de las señales de entrada DSR_IN, CTS_IN, DCD_IN y RI_IN de la instrucción V24_STAT_441 , respectivamente.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	VAR_INPUT	BOOL	Activa una transmisión con flanco positivo
ID	VAR_INPUT	WORD	Enlace de comunicación unívoco con un interlocutor
NDR	VAR_OUTPUT	BOOL	Señalizado flanco positivo, el programa de usuario dispone de nuevos datos recibidos
ERROR	VAR_OUTPUT	BOOL	Flanco positivo indica error
STATUS	VAR_OUTPUT	WORD	Contiene información de error detallada o advertencia
DTR_OUT	OUTPUT	BOOL	**D**ata **t**erminal **r**eady, CP 441 listo para entrar en funcionamiento (salida del CP 441)
DSR_IN	OUTPUT	BOOL	**D**ata **s**et **r**eady, interlocutor operativo (entrada del CP 441)
RTS_OUT	OUTPUT	BOOL	**R**equest **t**o **s**end, CP 441 listo para enviar[1] (salida del CP 441)
CTS_IN	OUTPUT	BOOL	**C**lear **t**o **s**end, el interlocutor puede recibir datos del CP 441 (respuesta a RTS = ON del CP 441) (entrada del CP 441)
DCD_IN	OUTPUT	BOOL	**D**ata **C**arrier **d**etect, nivel de señal de recepción (entrada del CP 441)
RI_IN	OUTPUT	BOOL	**R**ing **I**ndicator, indicador de llamada (entrada del CP 441)

Ejemplo

Tabla 9-324 Ejemplo para acceder a la instrucción V24_STAT_441

```
AWL
        CALL FB 5, DB55
                 REQ                 := DB30.DBX10.0
                 ID                  :=W#16#1000
                 NDR                 := DB30.DBX10.1
                 ERROR               := DB30.DBX10.2
                 STATUS              := DB30.DBW20
                 DTR_OUT             := DB30.DBX30.0
                 DSR_IN              := DB30.DBX30.1
                 RTS_OUT             := DB30.DBX30.2
                 CTS_IN              := DB30.DBX30.3
                 DCD_IN              := DB30.DBX30.4
                 RI_IN               := DB30.DBX30.5
```

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

V24_SET_441: Escribir señales cualificadoras en la interfaz RS232C

Descripción

El usuario puede activar o desactivar las correspondientes salidas de la interfaz a través de las entradas de parámetros de la instrucción V24_SET_441 desde una interfaz del CP 441 . La instrucción V24_SET_441 se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

La interfaz se selecciona indicando en la instrucción V24_SET_441 como ID la "ID local" de una de las conexiones que pasan por esta interfaz.

El resultado binario RB permanece igual.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	VAR_INPUT	BOOL	Activa una transmisión con flanco ascendente
ID	VAR_INPUT	WORD	Conexión unívoca con un interlocutor
DONE	VAR_OUTPUT	BOOL	Indica con flanco ascendente que la petición se ha completado sin errores

Parámetro	Declaración	Tipo de datos	Descripción
ERROR	VAR_OUTPUT	BOOL	Flanco ascendente indica error
STATUS	VAR_OUTPUT	WORD	Contiene información de error detallada o advertencia
RTS	INPUT	BOOL	Request to send, CP 441 listo para enviar (Forzar salida del CP 441)
DTR	INPUT	BOOL	Data terminal ready, CP 441 listo para entrar en funcionamiento (Forzar salida del CP 441)

Ejemplo

Tabla 9-325 Ejemplo para acceder a la instrucción V24_SET_441

AWL

```
        CALL FB 6, DB56
                  REQ         := DB40.DBX10.0
                  ID          :=W#16#1000
                  DONE        := DB40.DBX10.1
                  ERROR       := DB40.DBX10.2
                  STATUS      := DB40.DBW20
                  RTS         := DB40.DBX30.2
                  DTR         := DB40.DBX30.0
```

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

Área de registro de errores SYSTAT

Área de errores del SYSTAT

El área de registro de errores SYSTAT es un área de errores existente en todas las interfaces. En el SYSTAT se registran todos los errores/eventos que se han producido durante la transmisión de datos a una interfaz.

Nota

Dado que la petición STATUS se produce de forma asincrónica con respecto al resto de peticiones que se desarrollan en una interfaz, no es posible asignar a los mensajes de error un SFB con R_ID determinado. Es decir, puede indicarse qué errores se han producido en una interfaz pero no qué acceso (número R_ID) ha provocado el error.

Errores/eventos

Los mensajes SYSTAT se registran en los accesos del SFB STATUS en los bytes 2 a 15 del parámetro LOCAL. Además del byte de error (byte 2) se indican los primeros seis errores/eventos. El evento 1 es el más antiguo.

Si se producen otros eventos de error, éstos no pueden indicarse si no se borran antes las entradas más antiguas. Por ello es importante borrar a tiempo el área de registro de errores. Para ello hay que acceder al SFB STATUS.

Los errores/eventos se registran del siguiente modo:

- Byte 0 — Estado operativo del CP (02H para RUN, 05H para defectuoso)
- Byte 1 — Reservado
- Byte 2 — Bit 0 -F Error registrado en el SYSTAT
  Bit 1 -U Desbordamiento de errores
  Bit 2 -B Break
- Byte 3 — Reservado
- Byte 4/5 — Evento 1:
- Byte 6/7 — Evento 2:
- Byte 8/9 — Evento 3:
- Byte 10/11 — Evento 4:
- Byte 12/13 — Evento 5:
- Byte 14/15 — Evento 6:

Categorías de evento

Clase de evento	Significado
1	Error de hardware en el CP
2	Error al inicializar
3	Error al parametrizar las instrucciones
4	Error detectado por el CP en la comunicación CP - CPU
5	Error al procesar una petición de la CPU
6	Error al procesar una petición del interlocutor (sólo con RK 512)
7	Error de transmisión
8	Error de recepción
9	Se recibió telegrama de respuesta erróneo o telegrama de error del interlocutor acoplado
10 ($0A_H$)	Errores del telegrama de respuesta del interlocutor que fueron detectados por el CP

Números de evento

Código de error (W#16#...)	Descripción	Remedio
0101	Error en la comprobación del EPROM del sistema operativo del CP	CP defectuoso, debe cambiarse el CP.
0102	Error en comprobación de RAM del CP	
0103	Interfaz solicitante del CP defectuosa	
0104	No hay submódulo interfaz insertado	Inserte un submódulo interfaz apropiado para el CP.
0105	• Memoria de parámetros defectuosa • Submódulo interfaz extraído tras parametrización	Cambie el CP o bien inserte un submódulo interfaz apropiado para el CP.
0110	Error en el firmware del CP	Desconecte el módulo y vuelva a conectarlo. Si es necesario, cambie el módulo.
0201	No hay parámetros disponibles Memoria de parámetros vacía o contenido desconocido	Cargue los parámetros de la interfaz.
0208	La parametrización no concuerda con el submódulo interfaz.	Comprobar la configuración de parámetros del submódulo interfaz con la parametrización.
020F	Al iniciarse la comunicación parametrizada se detecta una parametrización no válida. No se ha podido parametrizar la interfaz.	Corrija la parametrización no válida y reinicie.
0210	Velocidad total de transferencia sobrepasada	Reduzca la velocidad de transferencia de ambas interfaces de modo que no se sobrepase la velocidad total de transferencia.
0211	Velocidad total de transferencia de nuevo dentro del margen permitido	Se ha sobrepasado la velocidad total de transferencia con anterioridad. Ahora vuelve a estar dentro del margen permitido
0301	Tipo de datos de fuente o destino no válido o no disponible Área (dirección de inicio, longitud) no válida DB no disponible o no válido (p. ej. DB 0) o bien otro tipo de datos de fuente o destino no válido o no disponible	Revise la parametrización de CPU y CP y corríjala en caso necesario. Solo en RK 512: El interlocutor transmite parámetros no válidos en el encabezado del telegrama. Revise la parametrización de CPU y CP y configure un bloque en caso necesario. Consulte los tipos de datos válidos en las tablas de peticiones. Sólo con RK 512: El interlocutor transmite parámetros incorrectos en el encabezado del telegrama.

Código de error (W#16#...)	Descripción	Remedio
0302	Área demasiado corta	Revise la parametrización de CPU y CP y compruebe el bloque/área en caso necesario. **Sólo en RK 512:**El interlocutor transmite parámetros incorrectos en el encabezado del telegrama.
0303	No es posible acceder al área	Revise la parametrización de CPU y CP. Consulte las direcciones de inicio y las longitudes válidas en las tablas de peticiones. **Sólo con RK 512:** El interlocutor transmite parámetros incorrectos en el encabezado del telegrama.
0401	CP no puede aceptar ninguna petición (sobrecarga).	Reduzca el número de peticiones simultáneas al CP en el programa de usuario.
0402	El CP no puede procesar este tipo de petición.	Compruebe si los bloques de función de sistema a que ha accedido desde el programa de usuario están autorizados para el CP.
0403	Tipo de datos erróneo, desconocido o no autorizado	Revise el programa, p. ej. error en la parametrización del SFB.
0407	Error en la transferencia de datos entre la CPU y el CP. No es posible recibir datos porque no puede accederse al área de datos de destino de la CPU o bien el área de destino de la CPU no existe o es demasiado corta. No es posible escribir datos en el área de destino ni leer datos del área de datos fuente de la CPU en estado STOP de la CPU a causa de la parametrización.	Compruebe el área de datos de destino en la CPU. Compruebe la parametrización "Respuesta a una parada de la CPU".
0408	• **Sólo para 3964R y driver ASCII** • Error temporal en la transferencia de datos entre la CPU y el CP (recepción). La petición se repetirá porque la CPU está temporalmente sobrecargada, el bloque de recepción (BRCV) se llama demasiado poco o el bloque de recepción está bloqueado temporalmente.	• Reducir la cantidad de llamadas de comunicación. Llamar el bloque de recepción con más frecuencia. Comprobar si el bloque de recepción está bloqueado durante demasiado tiempo.
0409	• **Sólo para 3964R y driver ASCII** • Error en la transferencia de datos entre la CPU y el CP. No es posible recibir datos. Tras varios intentos (véase (04)07H), la petición se ha cancelado al cabo de 10s porque el bloque de recepción (BRCV) no se llama o está bloqueado.	• Comprobar si el bloque de recepción se llama o está bloqueado.

Código de error (W#16#...)	Descripción	Remedio
040B	Error en la transmisión de datos entre la CPU y el CP porque • no se ha configurado ningún enlace • no es posible la recepción a través del enlace configurado	• Configure la conexión en "NetPro" • En "NetPro" indique en "Propiedades del objeto del enlace" para sentido de la comunicación: – 2: Interlocutor → Local o – 3: Local ↔ Interlocutor
0501	La petición en marcha ha sido interrumpida por un rearranque del CP.	En caso de conexión de la alimentación no hay ninguna solución. Al reparametrizar el CP desde la unidad de programación, antes de describir una interfaz debe asegurarse de que no se ejecuten más peticiones desde la CPU.
0502	La petición no es válida en este estado operativo del CP (p. ej. interfaz de dispositivo no parametrizada).	Parametrice la interfaz de dispositivo.
0503	Hora errónea, formato incorrecto	Compruebe los parámetros de la hora.
0505	**Sólo con driver de impresora:** Bloque de datos del sistema con textos de aviso no disponible en CP	Configure el texto de aviso con el software de parametrización y reinicie.
0506	**Sólo con driver de impresora:** Texto de aviso no disponible	Configure los textos de aviso con el software de parametrización y reinicie.
0507	**Sólo con driver de impresora:** Texto de aviso demasiado largo	Modifique el texto de aviso hasta un máximo de 150 caracteres (a un máximo de 250 caracteres en caso de variables)
0508	**Sólo con driver de impresora:** Demasiadas instrucciones de conversión	Ha configurado más instrucciones de conversión que variables. Las instrucciones de conversión sin la variable correspondiente se ignoran.
0509	**Sólo con driver de impresora:** Demasiadas variables	Ha configurado más variables que instrucciones de conversión. Las variables sin instrucción de conversión no se imprimen.
050A	**Sólo con driver de impresora:** Instrucción de conversión desconocida	Revise la instrucción de conversión. Las instrucciones de conversión no definidas o no compatibles se sustituyen por ****** en la impresión.
050B	**Sólo con driver de impresora:** Instrucción de control desconocida	Revise la instrucción de control. Las instrucciones de control indefinidas o incompatibles se ignoran. La instrucción de control tampoco se imprime en forma de texto.
050C	**Sólo con driver de impresora:** Instrucción de conversión no ejecutable	Revise la instrucción de conversión. Las instrucciones de conversión no ejecutables se imprimen con caracteres * de acuerdo con el ancho definido y la parte válida restante de la instrucción o bien de la representación estándar.

Código de error (W#16#...)	Descripción	Remedio
050D	**Sólo con driver de impresora:** Ancho insuficiente o excesivo en la instrucción de conversión	Corrija el ancho especificado para la variable de la instrucción de conversión a partir del número máximo de caracteres de la variable en los modos de representación basados en texto (A, C, D, S, T, Y, Z). En la impresión sólo se imprime el número de caracteres correspondiente al ancho especificado, el resto no se imprime. En el resto de casos se imprimen caracteres * en la cantidad equivalente al ancho.
050E	**Sólo con driver ASCII:** Error al enviar. Los caracteres de marca de fin parametrizados no han aparecido dentro de la longitud máxima admitida o bien se ha excedido la longitud de envío máxima al generarse anexos automáticamente.	Complemente los caracteres de marca de fin en el lugar deseado del búfer de envío o seleccione una longitud de telegrama menor en la generación automática de anexos.
050F	Número de peticiones simultáneas demasiado elevado	Modifique su programa STEP 7 de modo que se ejecuten menos peticiones simultáneamente.
0510	Área ocupada (recurso).	Repita la petición.
0511	Longitud no válida para este tipo de petición.	Divida los datos a transmitir en varias peticiones.
0512	**Sólo en RK 512:** Los parámetros de fuente y destino especificados en el SFB no se corresponden.	Consulte los valores válidos en las tablas de peticiones.
0513	Error en el tipo de datos (DB ...): Tipo de datos desconocido o tipo de datos no autorizado (p. ej. DE) **Sólo en RK 512:** Los tipos de datos de fuente y destino especificados en el SFB no se corresponden.	Consulte los tipos de datos autorizados y las correspondientes combinaciones en las tablas de peticiones.
0514	Direcciones de inicio demasiado elevadas para el tipo de datos deseado o dirección de inicio o nº DB/DX demasiado bajo.	Consulte en la tabla de peticiones las direcciones de inicio y los números DB/DX que pueden definirse en el programa.
0515	**Sólo en RK 512:** Número de bit erróneo en la marca de coordinación.	Nº de bit válido: 0 a 7
0516	**Sólo en RK 512:** Número de CPU demasiado elevado.	Nº de CPU válido: ninguno, 1, 2, 3 ó 4
0517	Error al recibir. El telegrama de recepción es mayor que 4 kbytes o mayor que la "longitud de recepción fija" parametrizada o bien no se ajusta al área de datos de destino.	Reduzca la longitud de telegrama de su interlocutor o prolongue el DB de recepción.
0518	Longitud de transmisión demasiado elevada en el envío (>4 kbytes).	Sólo en RK 512: Consulte las longitudes válidas en las tablas de peticiones. Divida la petición en varias peticiones de una longitud menor.

Código de error (W#16#...)	Descripción	Remedio
0519	CP en modo de operación incorrecto para petición de autómata	Compruebe si la interfaz solicitada está parametrizada.
051A	**Sólo en RK 512:** Error al enviar un telegrama de orden Se ha registrado un número de error de procedimiento inmediatamente antes en STATUS.	Vea las medidas de solución en el número de evento anterior.
051B	**Sólo con driver de impresora:** Precisión no válida	Corrija la precisión indicada en la instrucción de conversión. La precisión siempre está precedida por un punto que identifica y limita el ancho (p. ej.: ".2" para la impresión de punto flotante y 2 posiciones tras la coma). La precisión sólo es relevante en los modos de representación F, R, A y D. En el resto de casos se ignora.
051C	**Sólo con driver de impresora:** Variable no válida (Longitud de variable incorrecta/tipo incorrecto)	Corrija la variable de envío indicada.
0601	Error en el 1er byte de orden (no es 00 ó FFH)	En principio, error de estructura de encabezado en el interlocutor. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0602	Error en el 3er byte de orden (no es A, 0 ó E)	En principio, error de estructura de encabezado en el interlocutor. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0603	Error en el 3er byte de orden en telegramas de continuación (la orden no es como en el 1er telegrama)	En principio, error de estructura de encabezado en el interlocutor. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0604	Error en el 4º byte de orden (letra de orden incorrecta)	En principio, error de estructura de encabezado en el interlocutor o se ha solicitado una combinación de órdenes no autorizada en el CP. Compruebe que las órdenes son válidas. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Remedio
0605	Error en el 4º byte de orden en el telegrama de continuación (la orden no es como en el 1er telegrama)	En principio, error de estructura de encabezado en el interlocutor. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0606	Error en el 5º byte de orden (Nº de DB no válido)	Consulte los números de DB, las direcciones de inicio y las longitudes válidas en las tablas de peticiones.
0607	Error en el 5º ó 6º byte de orden (dirección de inicio demasiado elevada)	Consulte los números de DB, las direcciones de inicio y las longitudes válidas en las tablas de peticiones.
0608	Error en el 7º u 8º byte de orden (longitud no válida)	Consulte los números de DB/DX, las direcciones de inicio y las longitudes válidas en las tablas de peticiones.
0609	Error en el 9º y 10º byte de orden (marca de coordinación no válida en este tipo de datos o número de bit demasiado elevado)	En principio, error de estructura de encabezado en el interlocutor. Consulte en la tabla de peticiones cuándo está permitida una marca de coordinación.
060A	Error en el 10º byte de error (Nº de CPU no válido)	En principio, error de estructura de encabezado en el interlocutor
060B	Telegrama SEND de mayor/menor longitud que la esperada (se han recibido más/menos datos de los anunciados en el encabezado de telegrama).	Es necesaria una corrección en el interlocutor
060C	Recibido telegrama de orden GET con datos útiles.	Es necesaria una corrección en el interlocutor
060D	EL CP ha recibido el telegrama en un modo de operación no válido. • El enlace de recepción entre la CPU y el CP no está estructurado correctamente • El CP todavía no ha arrancado por completo • La interfaz solicitada se está reparametrizando es este momento	• Compruebe si la conexión que se intenta establecer está (bien) parametrizada. • Este aviso de error sólo puede aparecer durante el arranque del CP. Repita la petición • Se trata de un error temporal. Repita la petición.

Código de error (W#16#...)	Descripción	Remedio
060E	Error de sincronización del interlocutor • Se ha recibido un nuevo telegrama de orden (de continuación) antes de haber enviado un telegrama de respuesta. • 1. Se esperaba 1er telegrama de orden y se ha recibido un telegrama de continuación. • Se esperaba telegrama de orden de continuación y se ha recibido el 1er telegrama.	Este error puede indicarse tras un reinicio del autómata propio en caso de telegramas de gran longitud o de reinicio del interlocutor. En estos casos se trata de un comportamiento de arranque normal del sistema. Durante el funcionamiento normal, el error también puede aparecer como consecuencia de estados de error que sólo ha reconocido el interlocutor. En cualquier otro caso debe partir de un comportamiento erróneo del dispositivo interlocutor. El error puede no producirse en peticiones <128 bytes.
060F	DB bloqueado por función de coordinación	En el propio programa: Tras el procesamiento de los últimos datos transmitidos, desbloquear el bloque de recepción con "EN". En el programa del interlocutor: Repetición de la petición
0610	Se ha recibido un telegrama de longitud demasiado reducida (longitud <4 bytes para telegramas de continuación o reacción o longitud <10 bytes para telegramas de orden)	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0611	La longitud del telegrama y la longitud especificada en el encabezado de telegrama no coinciden.	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0612	Error al enviar el telegrama de respuesta (de continuación), se ha registrado un número de error de procedimiento inmediatamente antes en STATUS.	Vea las medidas de solución correspondientes al número de error registrado inmediatamente antes en STATUS.
0701	Envío del primer reintento: • Se ha detectado un error durante el envío del telegrama o bien • El interlocutor ha solicitado una repetición con un carácter de confirmación negativo (NAK).	Un reintento no es un error, si bien puede ser un indicio de que se ha producido un fallo en la línea de transmisión o que el interlocutor no funciona correctamente (comportamiento erróneo del interlocutor). Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se registra un Nº de error que describe el error que apareció en primer lugar.

Código de error (W#16#...)	Descripción	Remedio
0702	Error al establecer la conexión: • Tras el envío de STX se ha recibido un carácter cualquiera (menos DLE o STX) o bien • La respuesta se produjo demasiado pronto o bien • Se ha producido un conflicto de inicialización	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0703	Tiempo de retardo de acuse (TRA) transcurrido: Tras en envío de STX no ha llegado ninguna respuesta del interlocutor dentro del tiempo de retardo de acuse.	El dispositivo interlocutor es demasiado lento o no está preparado para recibir o bien hay p. ej. una rotura de la línea de envío. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0704	Interrupción por parte del interlocutor: Durante el proceso de envío actual el interlocutor ha recibido uno o más caracteres	Compruebe si también se indica un error en el interlocutor, ya que puede que no todos los datos de envío hayan llegado (p. ej. rotura en la línea de envío) o que haya fallos graves o un comportamiento defectuoso del dispositivo interlocutor. Compruebe si es así con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0705	Acuse negativo durante el envío	Compruebe si también se indica un error en el interlocutor, ya que puede que no todos los datos de envío hayan llegado (p. ej. rotura en la línea de envío) o que haya fallos graves o un comportamiento defectuoso del dispositivo interlocutor. Compruebe si es así con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0706	Error en el fin del enlace: • El telegrama ha sido rechazado por el interlocutor con NAK u otro carácter al final (con la excepción de DLE) o bien • El carácter de confirmación (DLE) se ha recibido demasiado pronto.	Compruebe si también se indica un error en el interlocutor, ya que puede que no todos los datos de envío hayan llegado (p. ej. rotura en la línea de envío) o que haya fallos graves o un comportamiento defectuoso del dispositivo interlocutor. Compruebe si es así con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0707	Tiempo de retardo de acuse al final del enlace/tiempo de vigilancia de respuesta tras el telegrama de envío agotados: Tras la interrupción de la conexión con DLE ETX no ha llegado ninguna respuesta del interlocutor dentro del TRA.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe si es así con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Remedio
0708	Sólo con driver ASCII y driver de impresora: El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor tiene un defecto, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
0709	No es posible establecer la conexión, se ha sobrepasado el número de intentos de establecimiento permitido.	Revise el cable de la interfaz o los parámetros de la transmisión.
070A	Los datos no han podido ser transmitidos, se ha sobrepasado el número de intentos de transmisión permitido.	Revise el cable de la interfaz o los parámetros de la transmisión.
0801	Esperando al primer reintento: Al recibir un telegrama se ha detectado un error y el CP exige mediante un acuse negativo (NAK) un reintento del interlocutor.	Un reintento no es un error, si bien puede ser un indicio de que se ha producido un fallo en la línea de transmisión o que el interlocutor no funciona correctamente (comportamiento erróneo del interlocutor). Si una vez alcanzado el número máximo de repeticiones el telegrama no ha podido ser enviado, se registra un Nº de error que describe el error que apareció en primer lugar.
0802	**Error al establecer la conexión:** • En posición de reposo se han recibido uno o más caracteres (cualquiera con la excepción de NAK o STX) o bien • Tras la recepción de un STX el interlocutor ha enviado otros caracteres sin esperar la recepción de la respuesta DLE. **Tras la conexión de la alimentación del interlocutor:** • Durante el tiempo de conexión del interlocutor, el CP recibe un carácter indefinido.	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0805	Error lógico durante la recepción: Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado del telegrama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0806	Tiempo de retardo de caracteres (TRC) transcurrido: • No se han recibido dos caracteres consecutivos dentro del TRC o bien • 1. El 1er carácter tras el envío de DLE en el establecimiento de la conexión no se ha recibido dentro del TRC.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruébelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Remedio
0808	Error en el carácter de comprobación de bloques BCC (sólo con RK 512 con procedimiento 3964R y procedimiento 3964R) El valor del BCC determinado internamente no concuerda con el BCC recibido por el interlocutor al final de la conexión.	Compruebe si el enlace está muy deteriorado; en ese caso también pueden aparecer códigos de error de la categoría de evento 8/ número de error 12. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
080A	No hay un búfer de recepción libre disponible: Tras la recepción de STX el procedimiento no ha podido disponer de un búfer de recepción libre durante el establecimiento de la conexión y un tiempo de espera posterior.	En el programa de usuario debe llamarse con más frecuencia la instrucción para la recepción.
080C	Error de transmisión: Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de rebosamiento). Si durante la posición de reposo se recibe un carácter defectuoso, el error se indica inmediatamente para facilitar la detección prematura de fallos en la línea de transmisión. Sólo con RK 512 y 3964(R): Si esto ocurre durante el modo de envío o recepción, se inician repeticiones.	Los fallos en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El aumento de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe si la velocidad de transmisión, la paridad y el número de bit de parada coinciden en ambos equipos.
080D	BREAK La línea de enlace (línea de recepción) que llega al equipo interlocutor está interrumpida.	Establezca la conexión entre los equipos o conecte el equipo interlocutor. En el modo TTY, compruebe si hay corriente en la línea en estado de reposo.
0812	Sólo con driver ASCII: Después de que el CP enviara XOFF o pusiera CTS a OFF, se recibieron más caracteres.	Vuelva a parametrizar el interlocutor o procure una lectura más rápida del CP.
0815	La parametrización del número de intentos de transmisión del CP y del interlocutor no concuerda	Parametrice en el interlocutor el mismo número de intentos de envío que en el CP. Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0816	• La longitud de un telegrama recibido era superior a la longitud máxima parametrizada o bien • La longitud del buzón configurado (sólo en CP 441) es demasiado reducida.	• Es necesaria una corrección en el interlocutor o bien • Aumente la longitud del buzón (sólo en CP 441).

Código de error (W#16#...)	Descripción	Remedio
0818	**Sólo con driver ASCII:** DSR = OFF o CTS = OFF	Antes o durante un proceso de envío el interlocutor ha puesto todas las señales DSR o CTS a "OFF". Revise el control de las señales cualificadoras RS 232C en el interlocutor.
0902	**Sólo en RK 512:** Error de acceso a memoria en el interlocutor (no hay memoria) Con SIMATIC S5 como interlocutor: • Área errónea en palabra indicadora o bien • Área de datos no existente (con la excepción de DB/DX) o bien • Área de datos demasiado reducida (con la excepción de DB/DX)	Compruebe si el área de datos en cuestión existe en el interlocutor y si es suficientemente grande o revise los parámetros del bloque de función de sistema llamado. Compruebe la longitud indicada en el bloque de función de sistema.
0903	**Sólo en RK 512:** Error de acceso DB/DX en el interlocutor (DB/DX no existente o demasiado reducido) Con SIMATIC S5 como interlocutor: • DB/DX no existente o bien • DB/DX demasiado reducido o bien • Nº DB/DX no válido Sobrepasada área de fuente permitida en petición GET	Compruebe si el área de datos en cuestión existe en el interlocutor y si es suficientemente grande o revise los parámetros del bloque de función de sistema llamado. Compruebe la longitud indicada en el bloque de función de sistema.
0904	**Sólo en RK 512:** El interlocutor indica "Tipo de petición no permitido".	Comportamiento erróneo del interlocutor, ya que el CP 441 nunca envía una orden de sistema.
0905	**Sólo en RK 512:** Error en el interlocutor o con SIMATIC S5 como interlocutor: • Tipo de fuente/destino no válido o bien • Error de memoria en el autómata del interlocutor o bien • Error de entendimiento CP/CPU en el interlocutor o bien • Autómata del interlocutor en modo STOP	Compruebe si el interlocutor puede transmitir el tipo de datos en cuestión. Compruebe la configuración del hardware del interlocutor. Ponga el selector del autómata del interlocutor en posición RUN.
0908	**Sólo en RK 512:** El interlocutor detecta error de sincronización: La secuencia del telegrama es defectuosa.	Este error se produce al reiniciar el autómata propio o el del interlocutor. Se trata de un comportamiento de arranque normal del sistema. No tiene que tomar ninguna medida. Durante el funcionamiento normal el error también puede deberse a errores que se han producido anteriormente. En cualquier otro caso debe partir de un comportamiento erróneo del interlocutor.

Código de error (W#16#...)	Descripción	Remedio
0909	**Sólo en RK 512:** DB/DX bloqueado en el interlocutor por marca de coordinación	En el programa del interlocutor: Una vez procesados los últimos datos transmitidos, vuelva a anular la marca de coordinación En el programa: Repita la petición.
090A	**Sólo en RK 512:** Error en el encabezado del telegrama que reconoce el interlocutor: 3er byte de orden en el encabezado erróneo	Compruebe si el error se debe a alguna avería o a un comportamiento erróneo del interlocutor. Compruébelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
090B	**Sólo en RK 512:** Error en el encabezado de telegrama: 1. ó 4º byte de orden en el encabezado erróneo	Compruebe si el error se debe a alguna avería o a un comportamiento erróneo del interlocutor. Compruébelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
090C	**Sólo en RK 512:** El interlocutor reconoce una longitud de telegrama errónea (longitud total).	Compruebe si el error se debe a alguna avería o a un comportamiento erróneo del interlocutor. Compruébelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
090D	**Sólo en RK 512:** Hasta ahora no se ha producido ningún reinicio en el interlocutor.	Reinicie el autómata del interlocutor o bien ponga el selector de modo de operación del CP o de la CPU en la posición RUN.
090E	**Sólo en RK 512:** Se ha recibido un número de error desconocido en el telegrama de respuesta.	Compruebe si el error se debe a alguna avería o a un comportamiento erróneo del interlocutor. Compruébelo con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0A01	**Sólo en RK 512:** Error de sincronización del interlocutor porque • telegrama de respuesta sin petición • telegrama de respuesta recibido antes de que se enviara el telegrama de continuación • tras el envío de un primer telegrama se ha recibido un telegrama de respuesta de continuación • tras el envío de un telegrama de continuación se ha recibido el primer telegrama de respuesta	Este error aparece tras un reinicio del autómata propio en caso de telegramas de gran longitud o en caso de reinicio del interlocutor. Se trata de un comportamiento normal de la instalación, no debe tomar ninguna medida. Durante el funcionamiento normal, el error también puede aparecer como consecuencia de estados de error que sólo ha reconocido el interlocutor. En cualquier otro caso debe partir de un comportamiento erróneo del equipo interlocutor. El error puede no producirse en peticiones <128 bytes.
0A02	**Sólo en RK 512:** Error de parametrización del telegrama de respuesta recibido (1er byte no es 00 o FF)	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0A03	**Sólo en RK 512:** El telegrama de respuesta recibido tiene exceso o escasez de datos.	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Código de error (W#16#...)	Descripción	Remedio
0A04	**Sólo en RK 512:** El telegrama de respuesta a una petición SEND ha llegado con datos.	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0A05	**Sólo en RK 512:** Dentro del tiempo de vigilancia no se ha recibido ningún telegrama de respuesta del interlocutor.	¿Es el interlocutor un equipo demasiado lento? Este error se presenta a menudo como consecuencia de un error que se ha producido con anterioridad. Pueden indicarse por ejemplo errores de recepción de procedimiento (categoría de eventos 8) después de que se enviara un telegrama GET. Causa: No ha sido posible recibir el telegrama, el tiempo de vigilancia se ha agotado. Este error también puede producirse si se ha reiniciado el interlocutor antes de que éste hubiera contestado al último telegrama GET recibido.
0A06	**Sólo en RK 512:** El telegrama de respuesta recibido tras la petición GET tiene poco datos	Compruebe el comportamiento erróneo del dispositivo interlocutor si corresponde con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.

Esclavo Modbus (RTU)

Instrucciones Modbus

Introducción

Para el maestro Modbus no son necesarias instrucciones separadas Modbus para la comunicación.

Instrucciones del CP 341

La transferencia de datos entre el CP 341 y la CPU se realiza con las instrucciones P_SND_RK y P_RCV_RK.

Tabla 9-326 Instrucciones del CP 341

Instrucción	Significado
P_RCV_RK	La instrucción P_RCV_RK permite recibir datos de un interlocutor y depositarlos en un bloque de datos o poner los datos a disposición del interlocutor.
P_SND_RK	La instrucción P_SND_RK permite enviar a un interlocutor todo el área o un área parcial de un bloque de datos o recuperar datos del interlocutor.

Instrucciones del sistema de automatización S7-400

La transferencia de datos entre el CP 441 y la CPU se realiza con las instrucciones BSEND y BRCV.

Tabla 9-327 Instrucciones del sistema de automatización S7-400

SFB	Significado
BSEND	La instrucción BSEND permite enviar datos desde un área de datos S7 a un interlocutor con una indicación de destino fija.
BRCV	La instrucción BRCV permite recibir datos de un interlocutor y depositarlos en área de datos S7.

MODB_341: Instrucción de esclavo Modbus para CP 341

Descripción

La instrucción MODB_341 y el respectivo driver permiten una conexión entre el sistema de control del maestro Modbus y el módulo de comunicación CP 341 como un sistema de esclavo "apto para Modbus".

Funcionamiento

En el programa de usuario, la instrucción MODB_341 debe llamarse en el programa cíclico. La instrucción utiliza el DB instancia respectivo IDB_MODB_341 como área de trabajo.

La instrucción MODB_341 inicializa el CP y ejecuta las funciones Modbus que el driver no puede realizar por sí solo.

Después de cada rearranque completo o rearranque de la CPU se tiene que realizar una inicialización de la instrucción MODB_341. La inicialización se activa con un flanco ascendente en la entrada CP_START. La instrucción borra primero el DB instancia, lee las áreas de operandos I, Q, M, T y C con la instrucción de sistema SZL_LESEN de la CPU y las deposita en el DB instancia. De ese modo, es posible comprobar si los requisitos de escritura del sistema maestro Modbus exceden el rango.

Con la petición SEND se notifica al CP el número de los DBs instancia y el transcurso correcto de la inicialización hasta el momento.

Tan pronto finalice la petición SEND sin errores, la salida CP_START_OK se activa y la inicialización de la instrucción concluye.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	Int	Dirección inicial del CP Aplicar asignación de HW Config
START_TIMER	INPUT	Timer	Temporizador para iniciar el tiempo de vigilancia
START_TIME	INPUT	S5Time	Valor de tiempo para iniciar el tiempo de vigilancia

Parámetro	Declaración	Tipo de datos	Descripción
OB_MASK	INPUT	BOOL	Enmascarar errores de acceso a periferia, retardar alarmas FALSE: Los errores de acceso a la periferia no se enmascaran TRUE: Los errores de acceso a la periferia se enmascaran y las alarmas se retardan
CP_START	INPUT	BOOL	Instrucción Iniciar inicialización
CP_START_FM	INPUT	BOOL	La inicialización se activa con el flanco ascendente de CP_START.
CP_START_NDR	OUTPUT	BOOL	Información: Requisitos de escritura del CP
CP_START_OK	OUTPUT	BOOL	El arranque ha finalizado sin errores TRUE: La petición de inicialización se ha podido ejecutar sin errores antes de transcurrir el tiempo de vigilancia.
CP_START_ERROR	OUTPUT	BOOL	El arranque se ha cancelado con errores TRUE: La petición de inicialización no se ha podido ejecutar sin errores ni siquiera después de transcurrir el tiempo de vigilancia.
ERROR_NR	OUTPUT	Word	Número de error Asignación véase Parámetro ERROR_NR, ERROR_INFO (Página 7989)
ERROR_INFO	OUTPUT	Word	Información adicional de error Asignación véase Parámetro ERROR_NR, ERROR_INFO (Página 7989)

Comportamiento en caso de error

Si la petición SEND concluye con errores, se desactivará CP_START y se activará CP_START_ERROR.

Si la inicialización ha finalizado con errores no será posible una comunicación Modbus. Todos los requisitos del sistema maestro Modbus se responden con un telegrama de código de excepción.

Tras una conexión (POWER ON) el CP necesita varios segundos para el test de hardware y memoria hasta que esté operativo. Los intentos de inicialización realizados por la instrucción MODB_341 durante este tiempo finalizan con errores. Por eso, la instrucción repite varias veces su petición de inicialización durante el tiempo de vigilancia.

Si la inicialización finaliza sin errores dentro del START-TIME parametrizado del START-TIMERS, se activará CP_START_OK. Si la petición de inicialización no se ha podido ejecutar sin errores incluso después de transcurrir el tiempo de vigilancia, se activará CP_START_ERROR.

MODB_441: Instrucción de esclavo Modbus para CP 441

Descripción

La instrucción MODB_441 y el respectivo driver permiten una conexión entre el sistema de control del maestro Modbus y el módulo de comunicación CP 441-2 como un sistema de esclavo "apto para Modbus".

Funcionamiento

En el programa de usuario, la instrucción MODB_441 debe llamarse en el programa cíclico. La instrucción utiliza el DB instancia respectivo IDB_MODB_441 como área de trabajo.

Los códigos de función Modbus 01, 02, 03, 04, 06 y 16 se procesan directamente con el CP. En los códigos de función 05 y 15, la instrucción aplica la entrada de los datos de bit en bit en el área de memoria SIMATIC.

La instrucción MODB_441 inicializa el CP y ejecuta las funciones Modbus que el driver no puede realizar por sí solo.

Después de cada rearranque completo o rearranque de la CPU se tiene que realizar una inicialización de la instrucción MODB_441 . La inicialización se activa con un flanco ascendente en la entrada CP_START. La instrucción borra primero el DB instancia, lee las áreas de operandos I, Q, M, T y C con la instrucción de sistema SZL_LESEN de la CPU y las deposita en el DB instancia. De ese modo, es posible comprobar si los requisitos de escritura del sistema maestro Modbus exceden el rango.

Con la petición SEND se notifica al CP el número de los DBs instancia y el transcurso correcto de la inicialización hasta el momento.

Tan pronto finalice la petición SEND sin errores, la salida CP_START_OK se activa y la inicialización de la instrucción concluye.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
ID	INPUT	Word	ID de conexión ID de conexión de la configuración
START_TIMER	INPUT	Timer	Temporizador para iniciar el tiempo de vigilancia
START_TIME	INPUT	S5Time	Valor de tiempo para iniciar el tiempo de vigilancia
STATUS_TIMER	INPUT	Timer	Temporizador para leer SYSTAT
STATUS_TIME	INPUT	S5Time	Intervalo de tiempo para leer SYSTAT

Parámetro	Declaración	Tipo de datos	Descripción
OB_MASK	INPUT	BOOL	Enmascarar errores de acceso a periferia, retardar alarmas FALSE: Los errores de acceso a la periferia no se enmascaran. TRUE: Los errores de acceso a la periferia se enmascaran y las alarmas se retardan.
CP_START	INPUT	BOOL	Instrucción Iniciar inicialización
CP_START_FM	INPUT	BOOL	La inicialización se activa con el flanco ascendente de CP_START
CP_START_NDR	OUTPUT	BOOL	Información: Requisitos de escritura del CP
CP_START_OK	OUTPUT	BOOL	El arranque ha finalizado sin errores TRUE: La petición de inicialización se ha podido ejecutar sin errores antes de transcurrir el tiempo de vigilancia.
CP_START_ERROR	OUTPUT	BOOL	El arranque se ha cancelado con errores TRUE: La petición de inicialización no se ha podido ejecutar sin errores ni siquiera después de transcurrir el tiempo de vigilancia.
ERROR_NR	OUTPUT	Word	Número de error Asignación véase Parámetro ERROR_NR, ERROR_INFO (Página 7989)
ERROR_INFO	OUTPUT	Word	Información adicional de error Asignación véase Parámetro ERROR_NR, ERROR_INFO (Página 7989)

Comportamiento en caso de error

Si la petición SEND concluye con errores, se desactivará CP_START y se activará CP_START_ERROR.

Si la inicialización ha finalizado con errores no será posible una comunicación Modbus. Todos los requisitos del sistema maestro Modbus se responden con un telegrama de código de excepción.

Tras una conexión (POWER ON) el CP necesita varios segundos para el test de hardware y memoria hasta que esté operativo. Los intentos de inicialización realizados por la instrucción MODB_441 durante este tiempo finalizan con errores. Por eso, la instrucción repite varias veces su petición de inicialización durante el tiempo de vigilancia.

Si la inicialización finaliza sin errores dentro del START-TIME parametrizado del START-TIMERS, se activará CP_START_OK. Si la petición de inicialización no se ha podido ejecutar sin errores incluso después de transcurrir el tiempo de vigilancia, se activará CP_START_ERROR.

Parámetro ERROR_NR, ERROR_INFO

Clases de error

Números de error de ... a	Descripción	Significado
1 ... 9	Error en la inicialización, instrucción y CP	La inicialización ha finalizado con errores en los números de error 1...9. El parámetro CP_START_ERROR es 1. No es posible la comunicación MODBUS con el sistema maestro.
10 ... 19	Error al procesar un código de función	Ha ocurrido un error al procesar un código de función en los números de error 10...19. El CP ha enviado una petición de procesamiento no permitida a la instrucción de comunicación. El error también será notificado al driver. Las siguientes peticiones se seguirán procesando.
90 ... 99	Otros errores	Se ha producido un error de procesamiento. El error no será notificado al driver. Las siguientes peticiones se seguirán procesando.

Parámetro ERROR_NR, ERROR_INFO

ERROR_NR (decimal)	ERROR_INFO	Texto del error	Solución
0	0	No hay errores	
1	SFC 51->RET_VAL	Error al leer la lista de estado del sistema con la SFC 51.	Analizar RET_VAL en ERROR_INFO, eliminar la causa.
2		Timeout en la inicialización del CP o error en la inicialización del CP	Compruebe si el "esclavo MODBUS" ha sido parametrizado como protocolo para la interfaz CP. Analice ERROR_INFO.
	P_SND_RK -> STATUS	Error en la petición P_SND_RK	Compruebe la dirección del módulo LADDR en la instrucción de comunicación.
	SFB 12->STATUS	Error en la petición BSEND	Compruebe la ID de conexión en la instrucción de comunicación.
10	Código de procesamiento	Función de procesamiento no válida transferida por el driver a la instrucción de comunicación.	Rearranque del CP (POWER ON)
11	Dirección inicial	Dirección inicial no válida transferida por el driver a la instrucción de comunicación.	Comprobar la dirección MODBUS del sistema maestro MODBUS.

ERROR_NR (decimal)	ERROR_INFO	Texto del error	Solución
12	Número de registros	Número de registros no válido transferido por el driver a la instrucción de comunicación: Número de registros = 0.	Comprobar número de registro del sistema maestro MODBUS, dado el caso, rearranque del CP (POWER ON)
13	Número de registros	Número de registros no válido transferido por el driver a la instrucción de comunicación: Número de registros > 128.	Comprobar número de registro del sistema maestro MODBUS, dado el caso, rearranque del CP (POWER ON)
14	Marca M - Dirección final	Intento de acceso a la "marca" del área de memoria SIMATIC más allá del final del área. Atención: La longitud del área en la CPU SIMATIC depende del tipo de CPU.	Reducir la dirección inicial de MODBUS o la longitud de acceso en el sistema maestro MODBUS.
15	Salidas Q - Dirección final	Intento de acceso a las "salidas" del área de memoria SIMATIC más allá del final del área. Atención: La longitud del área en la CPU SIMATIC depende del tipo de CPU.	Reducir la dirección inicial de MODBUS o la longitud de acceso en el sistema maestro MODBUS.
16	Temporizadores T - Dirección final	Intento de acceso a los "temporizadores (timer)" del área de memoria SIMATIC más allá del final del área. Atención: La longitud del área en la CPU SIMATIC depende del tipo de CPU.	Reducir la dirección inicial de MODBUS o la longitud de acceso en el sistema maestro MODBUS.
17	Contadores C - Dirección final	Intento de acceso a los "contadores" del área de memoria SIMATIC más allá del final del área. Atención: La longitud del área en la CPU SIMATIC depende del tipo de CPU.	Reducir la dirección inicial de MODBUS o la longitud de acceso en el sistema maestro MODBUS.
18	0	Área de memoria SIMATIC no válida transferida por el driver a la instrucción de comunicación.	Dado el caso, rearranque del CP (POWER ON)
19		Error al acceder a la periferia SIMATIC.	Compruebe si la periferia necesaria existe y no tiene errores.
90	P_SND_RK -> STATUS	Error al enviar al driver un aviso de acuse con la instrucción P_SND_RK	Analizar información de STATUS
	SFB 12->STATUS	Error al enviar al driver un aviso de acuse con el SFB 12 BSEND	Analizar información de STATUS
91	SFB 22->STATUS	Error al leer SYSTAT con el SFB 22 (STATUS)	Analizar información de STATUS
92	P_RCV_RK -> STATUS	Error al ejecutar una llamada RECEIVE/ FETCH con la instrucción P_RCV_RK	Analizar P_RCV_RK-STATUS

Área de registro de errores SYSTAT

Clases de evento

Clase de evento	Significado
5	Error al procesar una petición de la CPU
8	Error de recepción
14 ($0E_H$)	Error general de procesamiento

Parámetro STATUS

Código de error (W#16#...)	Descripción	Solución
0518	La longitud de transmisión al enviar es demasiado grande (> 4 KB) o la longitud de transmisión SEND es demasiado pequeña.	Comprobar la instrucción de comunicación, dado el caso volver a cargar.
0806	Tiempo de retardo de caracteres TRC excedido	Eliminar error en el dispositivo interlocutor o fallos en la línea de transmisión.
080C	En un carácter se ha detectado un error de transmisión (error de paridad, error de desbordamiento, error de bit de parada (trama)).	Compruebe si los fallos afectan a la línea de transmisión. Modifique la estructura del sistema o el cableado si fuera necesario. Compruebe si los parámetros de protocolo de velocidad de transmisión, número de bits de datos, paridad, número de bits de parada están ajustados en el CP y el interlocutor acoplado.
080D	BREAK La línea de recepción que llega al dispositivo interlocutor está interrumpida.	Establezca la conexión entre los dispositivos o conecte el dispositivo interlocutor. Compruebe en el modo TTY si la corriente de línea fluye en el estado de reposo. Compruebe y modifique, dado el caso, la asignación previa de la línea de recepción R(A) y R(B) en una conexión RS422/485(X27).
0830	Broadcast no está permitido con este código de función.	El sistema maestro MODBUS sólo puede emplear broadcast en los códigos de función habilitados para ello.
0831	El código de función recibido no está permitido.	Este código de función no puede emplearse para este driver.
0832	Se ha sobrepasado el número máximo de bits o registros o el número de bits no es divisible entre 16 al acceder al área de memoria SIMATIC de temporizadores o contadores.	Limitar el número máximo de bits a 2040 y el número máximo de registros a 127. Acceso a los temporizadores/contadores SIMATIC únicamente en tramas de 16 bits.
0833	El número de bits o registros en el código de función FC 15/16 y el elemento de telegrama byte_count no concuerdan.	Corregir número de bits, número de registros o byte_count.
0834	Se ha detectado una codificación de bits no permitida para "activar bit"/"desactivar bit".	En FC 05 se deben utilizar únicamente las codificaciones 0000Hex o FF00Hex.

Código de error (W#16#...)	Descripción	Solución
0835	Se ha detectado un subcódigo de diagnóstico no permitido (≠ 0000Hex) en el código de función FC 08 "Loop Back Test".	En FC 08 se debe utilizar únicamente el subcódigo 0000Hex.
0836	El valor creado internamente para la suma de verificación CRC 16 no coincide con la suma de verificación CRC recibida.	Comprobar la formación de la suma de verificación CRC en el sistema maestro MODBUS.
0837	Error de ejecución de telegrama: el sistema maestro MODBUS ha enviado un nuevo telegrama de solicitud antes de que el driver haya transferido el último telegrama de respuesta.	Aumente el timeout del telegrama de respuesta del esclavo en el sistema maestro MODBUS.
0E01	Error al inicializar el proceso SCC específico del driver.	Parametrizar y cargar el driver de nuevo.
0E02	Error de arranque del driver: Proceso SCC activo erróneo (driver SCC). El driver no funciona con este driver SCC.	Parametrizar y cargar el driver de nuevo.
0E03	Error de arranque del driver: Proceso de transmisión de datos activo erróneo (interfaz hacia los SFBs). El driver no funciona con este proceso de transmisión de datos.	Parametrizar y cargar el driver de nuevo.
0E04	Error de arranque del driver: Módulo de interfaz inadmisible. El driver no puede utilizarse con el módulo de interfaz parametrizado.	Comprobar la parametrización y corregirla.
0E05	Error de dongle del driver: No hay ningún dongle insertado o el dongle insertado es defectuoso. El driver no está operativo.	Compruebe si el dongle del driver está insertado en el CP. Si hay un dongle insertado, sustituya el dongle defectuoso por uno correcto.
0E06	Error de dongle del driver: El dongle no tiene un contenido válido. El driver no está operativo.	Consiga un dongle correcto en la sucursal Siemens en la que adquirió el driver.
0E10	Error de procedimiento interno: Rama predeterminada en el autómata emisor.	Rearranque del CP (POWER ON)
0E11	Error de procedimiento interno: Rama predeterminada en el autómata receptor.	Rearranque del CP (POWER ON)
0E12	Error interno de autómata activo: Rama predeterminada	Rearranque del CP (POWER ON)
0E13	Error interno de autómata pasivo: Rama predeterminada	Rearranque del CP (POWER ON)
0E20	El número de bits de datos de este acoplamiento debe ser 8. El driver no está operativo.	Corrija la parametrización del driver.
0E21	El factor de multiplicación parametrizado para el tiempo de retardo de caracteres no está entre 1 y 10. El driver trabaja con la configuración estándar 1.	Corrija la parametrización del driver.
0E22	El modo de operación parametrizado del driver no es válido. Debe indicarse "Modo normal" o "Modo con supresión de frecuencias perturbadoras". El driver no está operativo.	Corrija la parametrización del driver.

Código de error (W#16#...)	Descripción	Solución
0E23	Se ha parametrizado un valor no válido como dirección de esclavo. La dirección de esclavo 0 no es válida. El driver no está operativo.	Corrija la parametrización del driver.
0E24	Se han parametrizado límites no válidos para accesos de escritura. El driver no está operativo.	Corrija la parametrización del driver.
0E25	Se ha parametrizado una combinación "de/a" no válida en la introducción de las áreas "cambio del direccionamiento Modbus en la FC 01, 05, 15". (Área de marcas, salidas, temporizadores, contadores). El driver no está operativo.	Corrija la parametrización del driver.
0E26	Se ha parametrizado una combinación "de/a" no válida en la introducción de las áreas "cambio del direccionamiento Modbus en la FC 02". (Área de marcas, entradas). El driver no está operativo.	Corrija la parametrización del driver.
0E27	Se ha parametrizado un solapamiento de combinaciones "de/a" en la introducción de las áreas "cambio del direccionamiento Modbus en la FC 01, 05, 15". (Área de marcas, salidas, temporizadores, contadores). El driver no está operativo.	Corrija la parametrización del driver.
0E28	Se ha parametrizado un solapamiento de combinaciones "de/a" en la introducción de las áreas "cambio del direccionamiento Modbus en la FC 02". (Área de marcas, entradas). El driver no está operativo.	Corrija la parametrización del driver.
0E2E	Se ha producido un error al leer el archivo de parámetros de la interfaz. El driver no está operativo.	Rearranque del CP (POWER ON).
0E30	Error interno de entrega a la CPU: Acuse inesperado pasivo.	Se puede ignorar si ocurre esporádicamente.
0E31	Timeout en la entrega de datos a la CPU.	Comprobar la interfaz CP-CPU.
0E32	Error en la entrega de datos con RCV a la CPU: Analizar la causa - El error detallado se encuentra antes de esta entrada en SYSTAT.	Comprobar la interfaz CP-CPU.
0E33	Error interno en la entrega de datos CP-CPU: Estado del autómata no permitido	Comprobar la interfaz CP-CPU.
:		
0E38	Se ha producido un error al acceder a una de las áreas SIMATIC "marca, salida, temporizadores, contadores, entrada" con el código de función FC 01 o FC 02: P. ej. la entrada no existe o el intento de lectura va más allá del final del área.	Compruebe si existe el área SIMATIC que se ha llamado y si se ha intentado acceder más allá del final del área.
0E39	Se ha producido un error al acceder al área SIMATIC "bloque de datos" con el código de función FC 03, 04, 06, 16: El bloque de datos no existe o es demasiado corto.	Compruebe si existe el bloque de datos que se ha llamado y si es suficientemente largo.

Código de error (W#16#...)	Descripción	Solución
0E3A	Se ha producido un error al ejecutar una petición de escritura con el código de función FC 05, 15: No existe el bloque de datos instancia de la instrucción MODBUS o es demasiado corto.	Compruebe si existe el DB instancia parametrizado en la instrucción de comunicación MODBUS y si es suficientemente largo.
0E3B	Timeout (esperando acuse) al ejecutar una petición de escritura por medio de la instrucción de comunicación Modbus.	Comprobar configuración de conexión e interfaz CP-CPU (SFB BSEND), dado el caso, volver a cargar la instrucción de comunicación MODBUS.
0E3C	Petición no permitida con este driver.	Únicamente están permitidos SFB SEND, STATUS (sólo con el CP 441-2)
0E50	En las áreas SIMATIC de temporizadores/contadores, que están orientadas a palabras, el número de bits restante que resulta de la dirección Modbus es diferente a 0	Utilizar solamente direcciones MODBUS con las que resulten números de bits válidos.
0E51	La dirección Modbus recibida se encuentra fuera de las áreas parametrizadas "de/a".	Utilizar únicamente direcciones como indicación de dirección en el telegrama de solicitud. Éstas tienen que haber sido definidas antes en la parametrización.
0E52	• Rebase por exceso de un límite de área SIMATIC en el intento de acceso por medio del sistema maestro Modbus: N° de DB resultante < 1 • Acceso de escritura a un área no habilitada (parametrización) • Acceso de escritura a un DB instancia de la instrucción de comunicación	Limitar el área de acceso a las áreas de memoria SIMATIC válidas.
0E53	Rebase por exceso de un límite de área SIMATIC en el intento de acceso por medio del sistema maestro Modbus: P. ej. rebase por exceso al calcular el n.° de DB resultante (> 65535)	Limitar el área de acceso a las áreas de memoria SIMATIC válidas.
0E54	El acceso va más allá del final de área parametrizado o del final de área SIMATIC.	Limitar el área de acceso a las áreas de memoria SIMATIC válidas.
0E55	No está permitido un acceso de escritura a esta área de memoria SIMATIC.	Realizar accesos de escritura únicamente a las áreas de datos SIMATIC de marcas y salidas.
0E56	No es posible un funcionamiento del acoplamiento porque la instrucción de comunicación no se está ejecutando.	Llame cíclicamente la instrucción de comunicación MODBUS en el programa de usuario STEP 7. Dado el caso, ejecute de nuevo una inicialización de la instrucción de comunicación.
0E57	Se ha producido un error en la instrucción de comunicación al procesar el código de función Modbus.	Analice la causa exacta como se indica en el capítulo "Auto-Hotspot".

Interfaz serie ET 200S

Notas sobre el uso en una S7-1500

S_SEND / S_RCV / S_USSI / S_MODB a S7-1500

Tenga en cuenta lo siguiente si utiliza las instrucciones S_SEND, S_RCV, S_USSI o S_MODB de la periferia descentralizada ET 200S 1SI en una S7-1500:

- S_SEND
  Tipo de datos del parámetro DB_NO: UInt.
  Solamente pueden enviarse datos de áreas de memoria no optimizadas. Tenga esto en cuenta durante la parametrización de los parámetros DB_NO, DBB_NO y LEN.
- S_RCV
  Tipo de datos del parámetro DB_NO: UInt.
  Solamente pueden recibirse datos de áreas de memoria no optimizadas. Tenga esto en cuenta durante la parametrización de los parámetros DB_NO, DBB_NO y LEN.
- S_USSI
  - Tipo de datos de los parámetros DBND, DBPA y DBCP: UInt
    Números de DB admisibles para los parámetros DBND, DBPA y DBCP: 60000 a 60999
  - Comportamiento de la CPU en caso de error
    En caso de error, la CPU no pasa a STOP.
    Si se desea que la CPU pase a STOP en caso de error, debe evaluarse el parámetro ANZ en el programa de usuario
    (si ANZ es distinto de cero, debe llamarse a la instrucción STP):
- S_MODB
  Tipo de datos del parámetro DB_NO: DB_ANY.
  Solamente pueden intercambiarse datos de áreas de memoria no optimizadas. Tenga esto en cuenta durante la parametrización del parámetro DB_NO y del bloque de datos con la tabla de direcciones de datos Modbus.

Instrucciones de la librería "Procesador de comunicaciones > Interfaz serie ET200S"

Tenga en cuenta lo siguiente si utiliza las instrucciones de la periferia descentralizada ET 200S 1SI en una S7-1500 (librería: "Procesador de comunicaciones > Interfaz serie ET200S"):

- En el parámetro LADDR debe parametrizarse el identificador de hardware del módulo de comunicaciones.
- Tanto el búfer de transmisión como el de recepción deben estar en el área de memoria no optimizada.

ET 200S 1SI para 3964(R) y ASCII

S_RCV: Recibir datos

Descripción

La instrucción S_RCV transmite datos del módulo ET 200S 1SI a un área de datos S7, especificada por los parámetros DB_NO y DBB_NO. La instrucción S_RCV se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del módulo ET 200S 1SI. Una transferencia en curso se puede cancelar con el estado de señal "0" en el parámetro EN_R y, adicionalmente, con el estado de señal "1" en el parámetro R. (Puesto que puede suceder esporádicamente que tras ajustar el parámetro EN_R a "1" la transferencia ya no se inicie, para cancelar una transferencia en curso, debe ajustar el parámetro R a "1" en paralelo al ajuste del parámetro EN_R a "0".) La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desactivada mientras exista el estado lógico "0" en el parámetro EN_R. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción S_RCV pasa al estado inicial. La recepción está desactivada mientras exista el estado lógico "1" en el parámetro R. Cuando el estado de señal vuelve a ser "0", el telegrama interrumpido se vuelve a recibir desde el principio.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI que se va a llamar.

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. En caso de producirse un error, el número correspondiente se indica en STATUS. Si el búfer de recepción está lleno en más de 2/3 partes, STATUS contiene una advertencia después de cada llamada de S_RCV. Si no hay errores ni advertencias, STATUS tiene el valor "0".

NDR y ERROR/STATUS también se visualizan cuando se desactiva la instrucción S_RCV (parámetro LEN == 16#00) (véase el cronograma). Si se ha producido un error, se desactiva el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitar lectura de datos
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Bloqueo de recepción.

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º DB de recepción: Específico de la CPU. No se permite el valor 0.
DBB_NO	INPUT	INT	Número de bytes de datos (datos recibidos) **S7-1500**: 0 ≤ DBB_NO y DBB_NO + LEN ≤ tamaño del DB **S7-300/400**: 0 ≤ DBB_NO y DBB_NO + LEN < 8190
NDR [1]	OUTPUT	BOOL	La petición ha finalizado sin errores, datos transferidos Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
LEN [1]	OUTPUT	INT	Longitud del telegrama recibido 1 ≤ LEN ≤ 224 indicación en número de bytes
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición de recepción correcta, los parámetros NDR, ERROR, LEN y STATUS están disponibles durante **un** ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_RCV opera conjuntamente con un DB instancia I_RCV. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: en caso de error, (STATUS == W#16#1Exx), encontrará una indicación más precisa sobre el error en la variable SFCERR. Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Arranque

El parámetro COM_RST de la instrucción S_RCV permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

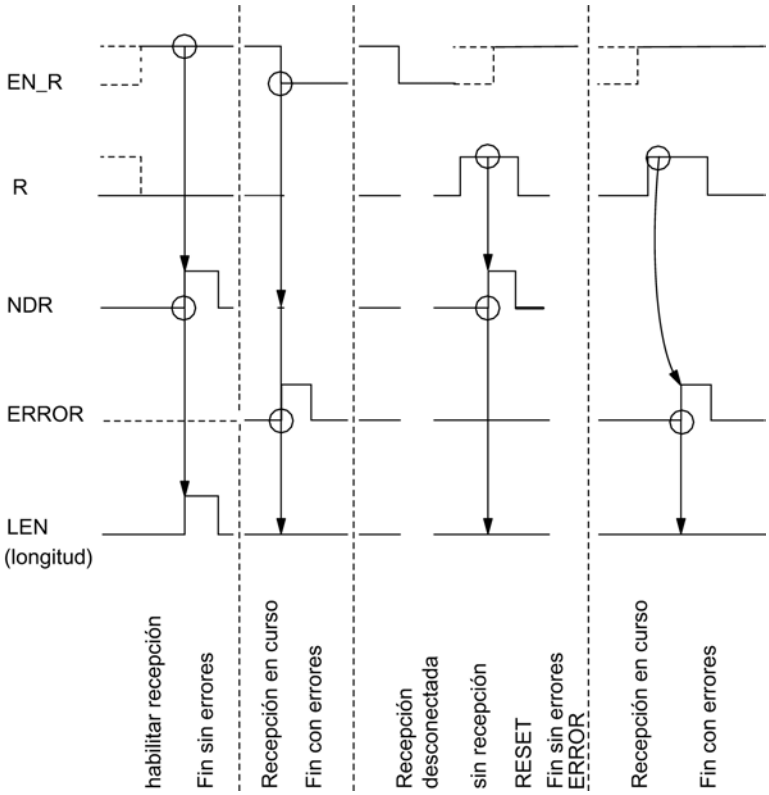
Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Cronograma

La figura siguiente muestra el comportamiento de los parámetros NDR, LEN y ERROR según sea el cableado de las entradas EN_R y R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Regulación

Nota

La instrucción S_RCV no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el ET 200S 1SI pueda recibir una petición iniciada después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CPU del módulo ET 200S para la instrucción S_RCV .

S_SEND: Enviar datos

Descripción

La instrucción S_SEND transmite un conjunto de datos de un bloque de datos especificado por los parámetros DB_NO, DBB_NO y LEN, al módulo ET 200S 1SI. La instrucción S_SEND se llama para transferir datos en el ciclo estáticamente (sin condiciones) o **alternativamente** en un programa controlado por tiempo.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción S_SEND puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al módulo ET 200S 1SI y la instrucción S_SEND pasa al estado inicial. Los datos que ya ha recibido el módulo ET 200S 1SI se envían al interlocutor. Si la entrada R tiene el estado lógico estático "1", la transmisión está desactivada.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI que se va a llamar.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se indican en caso de RESET de la instrucción S_SEND (véase el cronograma). Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco positivo
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos Nº de DB de envío: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número de bytes de datos (datos de transmisión) **S7-1500**: 0 ≤ DBB_NO y DBB_NO + LEN ≤ tamaño del DB **S7-300/400**: 0 ≤ DBB_NO y DBB_NO + LEN < 8190
LEN	INPUT	INT	Longitud de datos 1 ≤ LEN ≤ 224 indicación en número de bytes
DONE [1]	OUTPUT	BOOL	La petición ha finalizado sin errores Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición de transmisión correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_SEND opera conjuntamente con un DB instancia I_SEND. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, (STATUS == W#16#1Exx), encontrará una indicación más precisa sobre el error en la variable SFCERR. Esta variable de error sólo puede cargarse en el DB instancia mediante un acceso simbólico.

Arranque

El parámetro COM_RST de la instrucción S_SEND permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

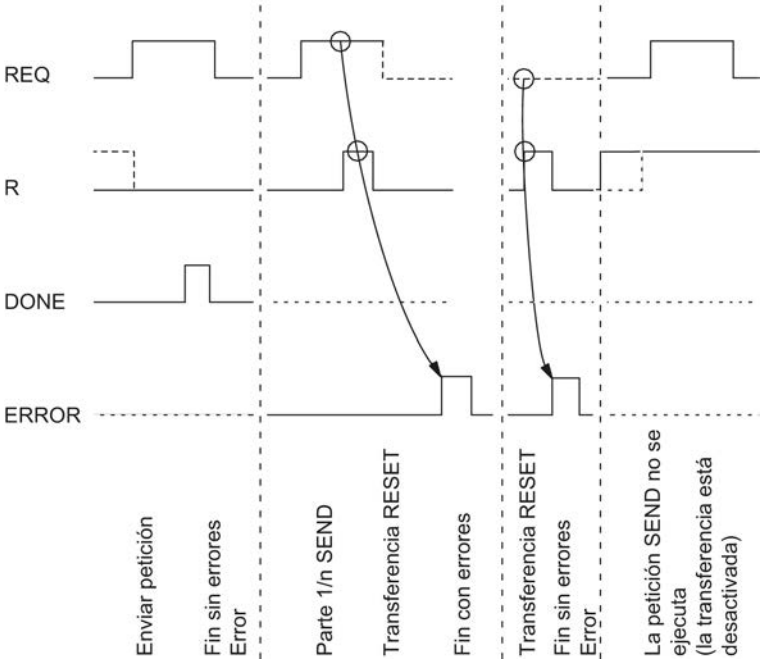
Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Cronograma

La figura siguiente muestra el comportamiento de los parámetros DONE y ERROR según sea el cableado de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción S_SEND no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el ET 200S 1SI pueda procesar una petición iniciada después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CPU del módulo ET 200S para la instrucción S_SEND . Una petición iniciada entre tanto no se pierde. Se transfiere al módulo ET 200S 1SI una vez concluida la coordinación de arranque.

S_VSTAT: Leer señales cualificadoras en la interfaz RS 232C

Descripción

La instrucción S_VSTAT lee las señales cualificadoras RS 232C del módulo ET 200S 1SI y las pone a disposición del usuario en los parámetros de bloque. La instrucción S_VSTAT se llama de forma estática (sin condiciones) para transferir datos en el ciclo o, alternativamente, en un programa controlado por tiempo.

Funcionamiento

Las señales cualificadoras RS 232C se actualizan cada vez que se llama la instrucción (sondeo cíclico).

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI con el que debe realizarse la conexión.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
DONE [1]	OUTPUT	BOOL	Indica que la instrucción ha finalizado (Salida ET 200S 1SI)

Parámetro	Declaración	Tipo de datos	Descripción
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.
DTR_OUT [1]	OUTPUT	BOOL	Data terminal ready, ET 200S 1SI listo para funcionar. (salida ET 200S 1SI)
DSR_IN [1]	OUTPUT	BOOL	Data set ready, interlocutor listo para funcionar. (Entrada ET 200S 1SI)
RTS_OUT [1]	OUTPUT	BOOL	Request to send, ET 200S 1SI listo para enviar. (salida ET 200S 1SI)
CTS_IN [1]	OUTPUT	BOOL	Clear to send, el interlocutor puede recibir datos del módulo ET 200S 1SI (respuesta a RTS = ON del ET 200S 1SI) (entrada ET 200S 1SI)
DCD_IN [1]	OUTPUT	BOOL	Data carrier detect, nivel de señal de recepción (entrada ET 200S 1SI)
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición correcta, estos parámetros están disponibles durante un ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_VSTAT opera conjuntamente con un DB instancia I_STAT. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Para detectar un cambio de señal es necesario una duración de impulso mínima. Las magnitudes decisivas son el tiempo de ciclo de la CPU, el tiempo de actualización en el módulo ET 200S 1SI y el tiempo de respuesta del interlocutor.

Arranque

El parámetro COM_RST de la instrucción S_VSTAT permite notificar un arranque a la instrucción.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

S_VSET: Escribir señales cualificadoras en la interfaz RS 232C

Descripción

Las salidas de las interfaces se activan y desactivan mediante las correspondientes entradas de parámetros de la instrucción S_VSET . La instrucción S_VSET se llama cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI con el que debe realizarse la conexión.

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
RTS	INPUT	BOOL	Request to send, ET 200S 1SI listo para enviar. (forzar salida ET 200S 1SI)
DTR	INPUT	BOOL	Data terminal ready, ET 200S 1SI operativo. (forzar salida ET 200S 1SI)
DONE [1]	OUTPUT	BOOL	(salida ET 200S 1SI)
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.

Parámetro	Declaración	Tipo de datos	Descripción
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición correcta, estos parámetros están disponibles durante un ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_VSET opera conjuntamente con un DB instancia I_SET. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Arranque

El parámetro COM_RST de la instrucción S_VSET permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

S_XON: Parametrizar el control de flujo de datos con XON/XOFF

Descripción

La instrucción S_XON permite ajustar otros parámetros adicionales si el módulo ha sido parametrizado para el control de flujo de datos XON/XOFF.

Nota

La instrucción S_XON debe llamarse antes de las instrucciones S_SEND, S_RCV, S_VSET y S_VSTAT y haber finalizado su petición (DONE = TRUE).

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
XON	INPUT	BYTE	Carácter XON 0 a $7F_H$ (7 bits de datos) 0 a FF_H (8 bits de datos) Configuración predeterminada: 11 (DC1)
XOFF	INPUT	BYTE	Carácter XOFF 0 a $7F_H$ (7 bits de datos) 0 a FF_H (8 bits de datos) Configuración predeterminada: 13 (DC3)
WAIT_FOR_XON	INPUT	TIME	Tiempo de espera para XON tras XOFF 20 ms a 10 min 55 s 350ms Configuración predeterminada: 2 s
DONE[1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00
ERROR[1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
STATUS[1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_XON opera conjuntamente con un DB instancia I_XON. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: En caso de error (STATUS == W#16#1Exx) se pueden consultar más detalles sobre el error en la variable SFCERR .

Arranque

El parámetro COM_RST de la instrucción S_XON permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Consulte también

Control del flujo de datos y señales cualificadoras (Página 1119)

S_RTS: Parametrizar el control de flujo de datos con RTS/CTS

Descripción

La instrucción S_RTS permite ajustar otros parámetros adicionales si el módulo ha sido parametrizado para el control de flujo de datos RTS/CTS.

Nota

La instrucción S_RTS debe llamarse antes de las instrucciones S_SEND, S_RCV, S_VSET y S_VSTAT y haber finalizado su petición (DONE = TRUE).

Parámetro

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco ascendente
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Transmisión bloqueada.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S SI La dirección inicial se toma de STEP 7.
WAIT_FOR_CTS	INPUT	TIME	Tiempo de espera para CTS = ON 20 ms a 10 min 55 s 350 ms Configuración predeterminada: 2 s
DONE[1]	OUTPUT	BOOL	Petición finalizada sin errores Parámetro STATUS == 16#00

Parámetro	Declaración	Tipo de datos	Descripción
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_RTS opera conjuntamente con un DB instancia I_RTS. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error (STATUS == W#16#1Exx) encontrará una indicación más precisa sobre el error en la variable SFCERR.

Arranque

El parámetro COM_RST de la instrucción S_RTS permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Consulte también

Control del flujo de datos y señales cualificadoras (Página 1119)

S_V24: Parametrizar el control de flujo de datos para el manejo automático de las señales cualificadoras RS 232C

Descripción

La instrucción S_V24 permite configurar parámetros adicionales, si el módulo está parametrizado para el manejo automático de las señales cualificadoras RS 232C.

Nota

La instrucción S_V24 debe llamarse antes de las instrucciones S_SEND, S_RCV, S_VSET y S_VSTAT y haber finalizado su petición (DONE = TRUE).

Parámetro

Parámetros	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco positivo
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
TIME_RTS_OFF	INPUT	TIME	Tiempo que debe transcurrir después de la transmisión antes de que se desconecte RTS. 0 ms a 10 min 55 s 350ms Configuración predeterminada: 10 ms
DATA_WAIT_TIME	INPUT	TIME	Tiempo que se espera para que el interlocutor active CTS = ON después de activar RTS. 0 ms a 10 min 55 s 350 ms Configuración predeterminada: 10 ms
DONE[1]	OUTPUT	BOOL	La petición ha finalizado sin errores Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8010) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8010) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición correcta, los parámetros DONE, ERROR y STATUS están disponibles durante un ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_V24 opera conjuntamente con un DB instancia I_V24. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: En caso de error (STATUS == W#16#1Exx) se pueden consultar más detalles sobre el error en la variable SFCERR .

Arranque

El parámetro COM_RST de la instrucción S_V24 permite notificar un arranque a la instrucción.

Asigne el estado 1 al parámetro COM_RST del OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Consulte también

Modo RS232 (CP 34x, 44x, CPU 31xC-2 PtP + ET 200S 1SI) (Página 1090)

Control del flujo de datos y señales cualificadoras (Página 1119)

Parámetro STATUS

Clases de evento

Clase de evento	Significado
2	Error al inicializar
5	Error al procesar una petición de la CPU
7	Error de transmisión
8	Error de recepción
11 ($0B_H$)	Advertencia
30 ($1E_H$)	Error en la comunicación entre el módulo y la CPU

Parámetro STATUS

Código de error (W#16#...)	Descripción	Solución
0201	Sin parametrización (válida).	Asigne parámetros correctos al módulo. Dado el caso, compruebe que el equipo se haya instalado de forma adecuada.
0502	La petición no está permitida en este estado operativo del módulo ET 200S 1SI (p. ej. interfaz del equipo no parametrizada).	El telegrama de envío tiene más de 224 bytes. La petición de envío ha sido cancelada por el módulo ET 200S 1SI. Seleccione una longitud de telegrama menor.
050E	Longitud de telegrama no válida	El telegrama de envío tiene más de 224 bytes. La petición de envío ha sido cancelada por el módulo ET 200S 1SI. Seleccione una longitud de telegrama menor.
0550	Petición de actualización de parámetros no válida para el control del flujo de datos actual del módulo ET 200S 1SI.	Modifique los parámetros de las instrucciones (S_XON, S_RTS, S_V24) en el programa del AS o modifique el control del flujo de datos del módulo ET 200S 1SI en la configuración de hardware, de modo que ambos coincidan.
0551	Error de secuencia de trama en la comunicación entre el módulo ET 200S 1SI y el sistema de automatización. El error ha surgido al transmitir un telegrama recibido del módulo ET 200S 1SI en el sistema de automatización.	El módulo y el sistema de automatización han cancelado la transferencia. Repita la petición de recepción; el módulo ET 200S 1SI vuelve a enviar el mensaje recibido.
0702	Sólo con 3964(R): Error al establecer la conexión: Tras el envío de STX se ha recibido NAK o un carácter cualquiera (excepto DLE o STX).	Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0703	Sólo con 3964(R): Tiempo de retardo de acuse (TRA) transcurrido: Tras en envío de STX no ha llegado ninguna respuesta del interlocutor dentro del tiempo de retardo de acuse.	El dispositivo interlocutor es demasiado lento o no está preparado para recibir o bien hay p. ej. una rotura de la línea de envío. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0704	Sólo con 3964(R): Interrupción por parte del interlocutor: Durante el proceso de envío en curso el interlocutor ha recibido uno o varios caracteres.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0705	Sólo con 3964(R): Acuse negativo al enviar	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0706	Sólo con 3964(R): Error al terminar la conexión: • El telegrama ha sido rechazado por el interlocutor con NAK u otro carácter al final (con la excepción de DLE) o bien • El carácter de confirmación (DLE) se ha recibido demasiado pronto.	Compruebe si el interlocutor también indica error porque quizá no han llegado todos los datos de transmisión (p. ej. rotura de la línea de transmisión), existen interferencias importantes o el interlocutor no funciona correctamente. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0707	Sólo con 3964(R): Tiempo de retardo de acuse al terminar la conexión/tiempo de supervisión de respuesta tras el telegrama de envío agotados: Tras la interrupción de la conexión con DLE ETX no ha llegado ninguna respuesta del interlocutor dentro del TRA.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe si es así con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
0708	Sólo con driver ASCII: El tiempo de espera a XON o CTS = ON ha transcurrido.	El interlocutor tiene un defecto, es demasiado lento o está offline. Revise el interlocutor o modifique la parametrización en caso necesario.
070B	Sólo con 3964(R): El conflicto de inicialización no puede resolverse, puesto que ambos interlocutores tienen predeterminada prioridad alta.	Cambie la parametrización.
070C	Sólo con 3964(R): El conflicto de inicialización no puede resolverse, puesto que ambos interlocutores tienen ajustada prioridad baja.	Cambie la parametrización.
0802	Sólo con 3964(R): Error al establecer la conexión: • En posición de reposo se han recibido uno o más caracteres (cualquiera con la excepción de NAK o STX) o bien • Tras la recepción de un STX el interlocutor ha enviado otros caracteres sin esperar la recepción de la respuesta DLE. Tras Power ON del interlocutor: • Mientras el interlocutor se conecta, el módulo recibe un carácter indefinido.	Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0805	Sólo con 3964(R): Error lógico durante la recepción: Tras la recepción de DLE se ha recibido otro carácter cualquiera (con la excepción de DLE, ETX).	Compruebe si el interlocutor siempre duplica DLE en el encabezado del telegrama y en la cadena de datos o si la interrupción de la conexión se realiza con DLE ETX. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.

Código de error (W#16#...)	Descripción	Solución
0806	Tiempo de retardo entre caracteres (TRC) transcurrido: • No se han recibido dos caracteres consecutivos dentro del TRC o bien Sólo con 3964(R): • 1. El 1er carácter tras el envío de DLE en el establecimiento de la conexión no se ha recibido dentro del TRC.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0807	Sólo con 3964(R): Longitud de telegrama no permitida: Se ha recibido un telegrama de longitud 0.	La recepción de un telegrama de longitud 0 no es un error. Compruebe el motivo por el que el interlocutor envía telegramas sin datos útiles.
0808	Sólo con 3964(R): Error en el carácter de comprobación de bloque BCC: El valor del BCC determinado internamente no concuerda con el BCC recibido por el interlocutor al final de la conexión.	Compruebe si el enlace tiene grandes interferencias; en ese caso, ocasionalmente también pueden aparecer códigos de error. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
0809	Sólo con 3964(R): El número de repeticiones debe ajustarse igual.	Parametrice en el interlocutor el mismo tiempo de espera de bloque que en el módulo. En caso necesario, compruebe el comportamiento erróneo del interlocutor con un dispositivo de comprobación de interfaz conectado a la línea de transmisión.
080A	No hay un búfer de recepción libre disponible: Durante la recepción no se disponía de ningún búfer de recepción vacío.	La instrucción S_RCV se debe llamar con más frecuencia.
080C	Error de transmisión: • Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de rebosamiento). Sólo con 3964(R): • Si esto ocurre durante el modo de envío o recepción, se inician repeticiones. • Si durante la posición de reposo se recibe un carácter defectuoso, el error se indica inmediatamente para facilitar la detección prematura de fallos en la línea de transmisión. • Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión de ambos interlocutores.	Los fallos en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El riesgo de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos.
080D	BREAK: La línea de recepción que llega al interlocutor está interrumpida.	Restablezca la conexión o conecte el interlocutor.
0810	Sólo con driver ASCII: Error de paridad: • Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.

Código de error (W#16#...)	Descripción	Solución
0811	Sólo con driver ASCII: Error de trama de caracteres: • Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
0812	Sólo con driver ASCII: Después de que el módulo enviara XOFF o pusiera CTS a OFF, se recibieron más caracteres.	Vuelva a parametrizar el interlocutor o lea los datos del módulo con más rapidez.
0818	Sólo con driver ASCII: DSR = OFF o CTS = OFF	Antes o durante un proceso de envío el interlocutor ha puesto todas las señales DSR o CTS a "OFF". Revise el control de las señales cualificadoras RS 232C en el interlocutor.
0850	La longitud del telegrama de recepción es superior a 224 bytes o mayor que la longitud de telegrama parametrizada	Adaptar la longitud de telegrama del interlocutor
0B01	El búfer de recepción está lleno en más de 2/3 partes	
1E0D	"Interrupción del trabajo por rearranque completo, rearranque o reset"	
1E0E	Error estático al llamar la SFC DPRD_DAT. Se puede disponer del valor de retorno RET_VAL de la SFC en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E0F	Error estático al llamar la SFC DPWR_DAT. Se puede disponer del valor de retorno RET_VAL de la SFC en la variable SFCERR del DB de instancia para su evaluación.	Cargue la variable SFCERR del DB de instancia.
1E10	Error estático al llamar la SFC RD_LGADR. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
1E11	Error estático al llamar la SFC RDSYSST. El valor de retorno RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB instancia.	Cargue la variable SFCERR del DB instancia.
1E20	Parámetro fuera del rango.	Cambie la entrada de la instrucción para que esté en el área válida.
1E41	El número de bytes indicado en el parámetro LEN de las instrucciones no está permitido	Respete el rango de 1 a 224 bytes.

Evaluación de la variable SFCERR

Obtendrá más información sobre los errores ocurridos (1E) $0E_H$, (1E) $0F_H$, (1E) 10_H y (1E) 11_H de la clase de evento 30 a través de la variable SFCERR.

La variable SFCERR se puede cargar desde el DB instancia de la instrucción correspondiente.

Los mensajes de error registrados en la variable SFCERR se encuentran en las funciones de sistema "DPRD_DAT" y SFC15 "DPWR_DAT" en el manual de referencia *Software de sistema para S7-300 y S7-400. Funciones estándar y funciones de sistema*.

ET 200S 1SI de Modbus

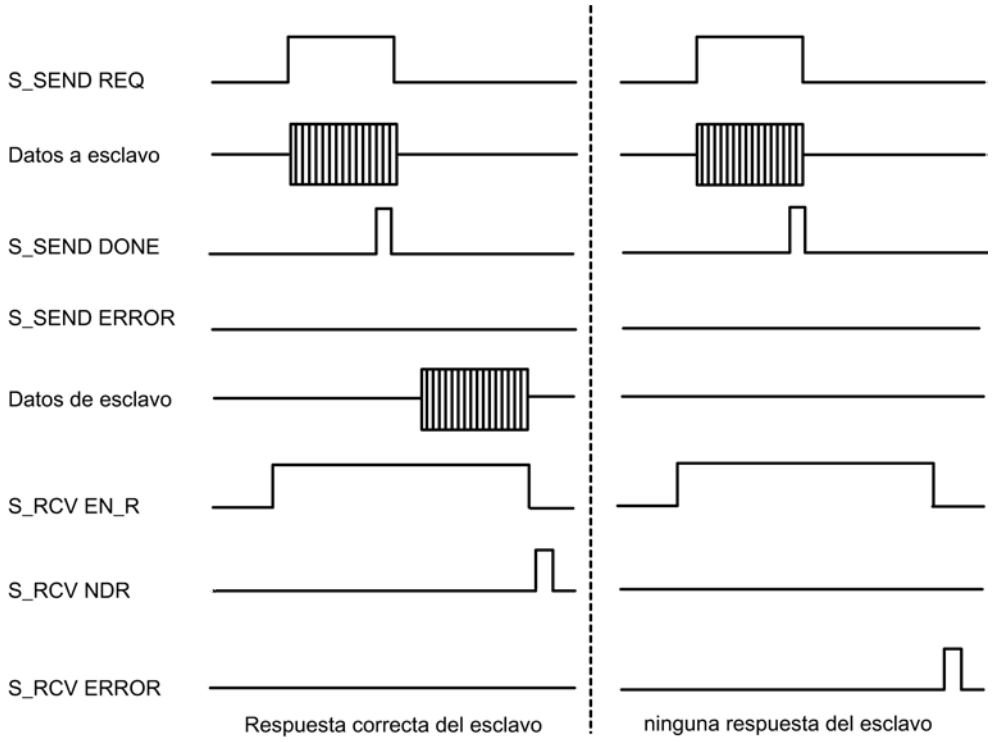
Enviar y recibir como maestro Modbus

Descripción

La transmisión de datos entre el módulo y la CPU se realiza mediante las instrucciones S_SEND y S_RCV. La instrucción S_SEND se activa con un flanco en la entrada REQ cuando deben emitirse datos. La instrucción S_RCV se prepara para la recepción con EN_R=1. En todos los códigos de función de lectura es necesario un S_RCV.

Enviar y recibir con el protocolo del maestro Modbus

Para ejecutar una petición del maestro Modbus tienen que activarse las instrucciones S_SEND y S_RCV. La instrucción S_SEND se activa con un flanco en la entrada REQ cuando deben emitirse datos al módulo. La instrucción S_RCV se prepara con EN_R=1 para recibir datos del módulo. En todos los códigos de función de lectura es necesario un S_RCV. La figura siguiente muestra el comportamiento global de los parámetros de S_SEND y S_RCV cuando se ejecuta una petición Modbus.



Nota

Secuencia de enviar y recibir

Dado que la interfaz entre el programa de usuario y el módulo interfaz funciona en modo semidúplex, hay que tener en cuenta lo siguiente:

Después de una petición de lectura del maestro Modbus con acuse positivo, hay que recuperar los datos recibidos del módulo de interfaz con la instrucción S_RCV antes de iniciar una nueva petición de envío del maestro Modbus.

S_RCV: Recibir datos

Descripción

La instrucción S_RCV transmite datos del módulo ET 200S 1SI a un área de datos S7, especificada por los parámetros DB_NO y DBB_NO. La instrucción S_RCV se llama para transferir datos cíclica o estáticamente (sin condiciones) en un programa controlado por tiempo.

Funcionamiento

Con el estado lógico (estático) "1" en el parámetro EN_R se habilita la comprobación de si pueden leerse datos del módulo ET 200S 1SI. Una transferencia en curso se puede cancelar con el estado de señal "0" en el parámetro EN_R y, adicionalmente, con el estado de señal "1" en el parámetro R. (Puesto que puede suceder esporádicamente que tras ajustar el parámetro EN_R a "1" la transferencia ya no se inicie, para cancelar una transferencia en curso, debe ajustar el parámetro R a "1" en paralelo al ajuste del parámetro EN_R a "0".) La petición de recepción cancelada finaliza con un mensaje de error (salida STATUS). La recepción está desactivada mientras exista el estado lógico "0" en el parámetro EN_R. Dependiendo de la cantidad de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

Si la instrucción detecta el estado lógico "1" en el parámetro R, se cancela la petición actual de transmisión y la instrucción S_RCV pasa al estado inicial. La recepción está desactivada mientras exista el estado lógico "1" en el parámetro R. Cuando el estado de señal vuelve a ser "0", el telegrama interrumpido se vuelve a recibir desde el principio.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI que se va a llamar.

La salida NDR indica "Petición finalizada sin errores/Datos aceptados" (leídos todos los datos). ERROR indica que se ha producido un error. En caso de producirse un error, el número correspondiente se indica en STATUS. Si el búfer de recepción está lleno en más de 2/3 partes, STATUS contiene una advertencia después de cada llamada de S_RCV. Si no hay errores ni advertencias, STATUS tiene el valor "0".

NDR y ERROR/STATUS también se visualizan cuando se desactiva la instrucción S_RCV (parámetro LEN == 16#00) (véase el cronograma). Si se ha producido un error, se desactiva

el resultado binario RB. Si se termina el bloque sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
EN_R	INPUT	BOOL	Habilitar lectura de datos
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Bloqueo de recepción.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos N.º DB de recepción: Específico de la CPU. No se permite el valor 0.
DBB_NO	INPUT	INT	Número de bytes de datos (datos recibidos) **S7-1500**: 0 ≤ DBB_NO y DBB_NO + LEN ≤ tamaño del DB **S7-300/400**: 0 ≤ DBB_NO y DBB_NO + LEN < 8190
NDR [1]	OUTPUT	BOOL	La petición ha finalizado sin errores, datos transferidos Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8024) contiene la información de error.
LEN [1]	OUTPUT	INT	Longitud del telegrama recibido 1 ≤ LEN ≤ 224 indicación en número de bytes
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8024) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición de recepción correcta, los parámetros NDR, ERROR, LEN y STATUS están disponibles durante **un** ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_RCV opera conjuntamente con un DB instancia I_RCV. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB de instancia.

Nota

Excepción: en caso de error, (STATUS == W#16#1Exx), encontrará una indicación más precisa sobre el error en la variable SFCERR. Esta variable de error sólo puede cargarse en el DB de instancia mediante un acceso simbólico.

Arranque

El parámetro COM_RST de la instrucción S_RCV permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

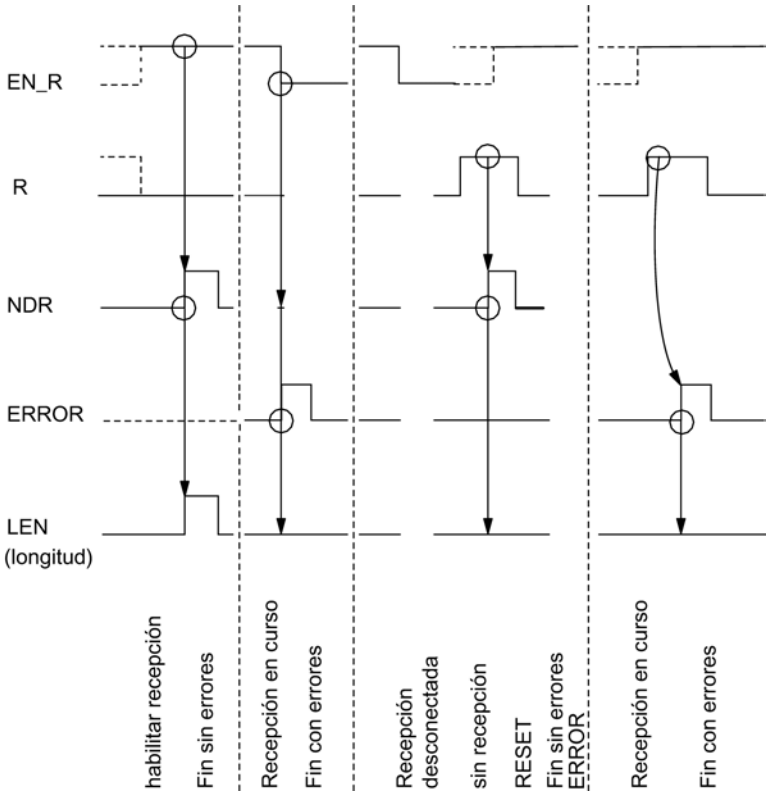
Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Cronograma

La figura siguiente muestra el comportamiento de los parámetros NDR, LEN y ERROR según sea el cableado de las entradas EN_R y R.



Nota

La entrada EN_R debe ponerse estáticamente a "1". Mientras dure toda la petición de recepción, el parámetro EN_R debe recibir el RLO "1" (resultado lógico).

Regulación

Nota

La instrucción S_RCV no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el ET 200S 1SI pueda recibir una petición iniciada después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CPU del módulo ET 200S para la instrucción S_RCV .

Consulte también

Transferencia de datos con el maestro Modbus ET 200S 1SI (Página 1140)

S_SEND: Enviar datos

Descripción

La instrucción S_SEND transmite un conjunto de datos de un bloque de datos especificado por los parámetros DB_NO, DBB_NO y LEN, al módulo ET 200S 1SI. La instrucción S_SEND se llama para transferir datos en el ciclo estáticamente (sin condiciones) o **alternativamente** en un programa controlado por tiempo.

Funcionamiento

Con un flanco ascendente en la entrada REQ se inicia la transmisión de los datos. Dependiendo del volumen de datos, la transmisión puede desarrollarse en varias llamadas (ciclos del programa).

La instrucción S_SEND puede llamarse en el ciclo con el estado lógico "1" en la entrada de parámetros R. De este modo, se cancela la transferencia al módulo ET 200S 1SI y la instrucción S_SEND pasa al estado inicial. Los datos que ya ha recibido el módulo ET 200S 1SI se envían al interlocutor. Si la entrada R tiene el estado lógico estático "1", la transmisión está desactivada.

En el parámetro LADDR se indica la dirección del módulo ET 200S 1SI que se va a llamar.

La salida DONE indica "Fin de petición sin errores". ERROR indica que se ha producido un error. En caso de producirse un error, el número de evento correspondiente se indica en STATUS. Si no se producen errores, STATUS asume el valor 0. DONE y ERROR/STATUS también se indican en caso de RESET de la instrucción S_SEND (véase el cronograma). Si se ha producido un error, se desactiva el resultado binario RB. Si el bloque finaliza sin errores, el estado del resultado binario es "1".

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
REQ	INPUT	BOOL	Inicio de petición con flanco positivo
R	INPUT	BOOL	Interrupción de la petición Se interrumpe la petición en curso. Envío bloqueado.
LADDR	INPUT	INT	Dirección inicial del módulo ET 200S 1SI La dirección inicial se toma de STEP 7.
DB_NO	INPUT	INT	Número del bloque de datos Nº de DB de envío: específico de la CPU. No se admite el valor 0.
DBB_NO	INPUT	INT	Número de bytes de datos (datos de transmisión) **S7-1500**: 0 ≤ DBB_NO y DBB_NO + LEN ≤ tamaño del DB **S7-300/400**: 0 ≤ DBB_NO y DBB_NO + LEN < 8190
LEN	INPUT	INT	Longitud de datos 1 ≤ LEN ≤ 224 indicación en número de bytes
DONE [1]	OUTPUT	BOOL	La petición ha finalizado sin errores Parámetro STATUS == 16#00
ERROR [1]	OUTPUT	BOOL	Petición interrumpida con errores El parámetro STATUS (Página 8024) contiene la información de error.
STATUS [1]	OUTPUT	WORD	Especificación del error Si ERROR == 1, en el parámetro STATUS (Página 8024) figura la información de error.
COM_RST	IN_OUT	BOOL	Reinicio de la instrucción
[1] Tras una petición de transmisión correcta, los parámetros DONE, ERROR y STATUS están disponibles durante **un** ciclo de la CPU.			

Asignación en el área de datos

La instrucción S_SEND opera conjuntamente con un DB instancia I_SEND. El número del DB se indica con la llamada. No está permitido acceder a los datos del DB instancia.

Nota

Excepción: en caso de error, (STATUS == W#16#1Exx), encontrará una indicación más precisa sobre el error en la variable SFCERR. Esta variable de error sólo puede cargarse en el DB instancia mediante un acceso simbólico.

Arranque

El parámetro COM_RST de la instrucción S_SEND permite notificar un arranque a la instrucción.

Ponga el parámetro COM_RST a 1 en el OB de arranque.

Llame la instrucción en modo cíclico sin activar o desactivar el parámetro COM_RST.

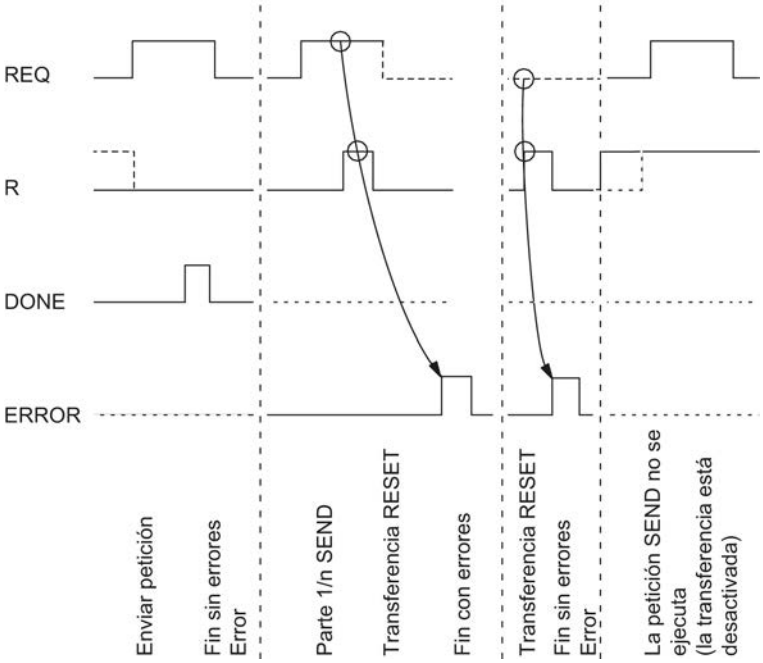
Si el parámetro COM_RST está activado,

- la instrucción proporciona información sobre el módulo ET 200S 1SI (número de bytes en el área de periferia, en periferia descentralizada o no).
- la instrucción se desactiva y finaliza cualquier petición que pudiera haberse iniciado antes (antes de la última transición a Stop de la CPU).

Una vez que la instrucción ha proporcionado información sobre el módulo ET 200S 1SI, desactiva el parámetro COM_RST automáticamente.

Cronograma

La figura siguiente muestra el comportamiento de los parámetros DONE y ERROR según sea el cableado de las entradas REQ y R.



Nota

La entrada REQ está activada por flanco. Es suficiente un flanco ascendente en la entrada REQ. El RLO (resultado lógico) no tiene que estar a "1" durante toda la transmisión.

Regulación

Nota

La instrucción S_SEND no dispone de comprobación de parámetros. Si la parametrización es incorrecta, la CPU puede cambiar al estado STOP.

Nota

Antes de que el ET 200S 1SI pueda procesar una petición iniciada después de una transición de estado de la CPU de STOP a RUN, debe haber concluido el mecanismo de arranque CPU del módulo ET 200S para la instrucción S_SEND . Una petición iniciada entre tanto no se pierde. Se transfiere al módulo ET 200S 1SI una vez concluida la coordinación de arranque.

Consulte también

Transferencia de datos con el maestro Modbus ET 200S 1SI (Página 1140)

S_MODB: Instrucción de esclavo Modbus para ET 200S 1SI

Descripción

Para ejecutar una petición de esclavo Modbus debe activarse cíclicamente la instrucción S_MODB en el programa de usuario. La instrucción S_MODB recibe la petición del módulo de interfaz serie ET 200S Modbus/USS, la ejecuta y devuelve la respuesta al módulo. La comunicación entre la CPU y el módulo se realiza mediante las instrucciones S_SEND y S_RCV , que se llaman con S_MODB .

Funcionamiento

Tras cada rearranque de la CPU, el programa de usuario tiene que inicializar la instrucción de comunicación Modbus. La inicialización se activa con un flanco ascendente en la entrada CP_START. La instrucción registra los tamaños de las áreas de operandos I, Q, M, T y C de la CPU en el bloque de datos instancia de la instrucción. Una vez finalizada correctamente la inicialización, la instrucción activa la salida CP_START_OK.

La salida CP_START_ERROR indica un error de inicialización. En este caso, no es posible una comunicación Modbus y todas las peticiones del maestro Modbus se responden con un mensaje de código de excepción.

S_MODB utiliza una tabla de conversión de datos Modbus, que se encuentra en el bloque de datos, para reproducir las direcciones Modbus en las áreas de memoria de SIMATIC S7.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
LADDR	INPUT	INT	Dirección inicial del módulo de interfaz serie ET 200S 1SI La dirección inicial se toma de STEP 7.
START_TIMER	INPUT	TIMER	Temporizador del timeout de inicialización: Temporizador de la CPU que utiliza la instrucción para impedir que la inicialización se cuelgue.
START_TIME	INPUT	S5TIME	Valor de tiempo del timeout de inicialización: Tiempo máximo que puede durar la inicialización de la instrucción
DB_No	INPUT	BLOCK_DB	Número del bloque de datos que contiene la tabla de conversión Modbus: Véase áreas de datos en la SIMATICCPU
OB_MASK	INPUT	BOOL	Enmascara errores de acceso E/S y retarda las alarmas: TRUE permite que se notifiquen errores de acceso a E/S no existentes sólo al esclavo Modbus
CP_START	INPUT	BOOL	Empezar la inicialización de la instrucción
CP_START_FM	INPUT	BOOL	La inicialización se activa con un flanco ascendente en la entrada START: Elemento de memoria en el que la instrucción guarda el valor anterior de CP_START
CP_NDR	OUTPUT	BOOL	Nueva función de escritura Modbus finalizada
CP_START_OK	OUTPUT	BOOL	Inicialización finalizada sin errores: La inicialización ha finalizado sin errores antes de que el tiempo de vigilancia transcurriera.
CP_START_ERROR	OUTPUT	BOOL	Inicialización finalizada con errores: La inicialización no se ha podido finalizar sin errores, tampoco tras haber transcurrido el tiempo de vigilancia.
ERROR_NR	OUTPUT	WORD	Número de error Véase el diagnóstico
ERROR_INFO	OUTPUT	WORD	Información adicional de error Véase el diagnóstico

Asignación en el área de datos

La instrucción S_MODB opera conjuntamente con un DB instancia IDB_MODB cuyo número se indica en la llamada.

No es posible acceder a los datos del DB instancia.

Comportamiento en caso de error

El parámetro de entrada OB_MASK permite indicar a la instrucción Modbus que enmascare los errores de acceso E/S. Si se produce un acceso de escritura a periferia no existente, la CPU no pasa al estado operativo STOP y no llama ningún OB de error. La instrucción detecta el error de acceso y se finaliza con una respuesta de error en el maestro Modbus.

Parámetro STATUS (Modbus / USS)

Clases de evento

Clase de evento	Significado
2	Error al inicializar
5	Error al procesar una petición de la CPU
8	Error de recepción
14 ($0E_H$)	Errores generales de procesamiento: parametrización, procesamiento de una petición S_SEND. Evaluación de recepción, recepción del aviso del código de excepción
30 ($1E_H$)	Error en la comunicación entre el módulo y la CPU

Parámetro STATUS

Código de error (W#16#...)	Descripción	Solución
0201	Sin parametrización (válida).	Asigne parámetros correctos al módulo. Dado el caso, compruebe que el equipo se haya instalado de forma adecuada.
0502	Petición no permitida en este estado operativo del módulo interfaz serie ET 200S Modbus/USS (ejemplo: la interfaz del equipo no está parametrizada).	Evalúe la alarma de diagnóstico y corrija el error correspondiente.
050E	Longitud de telegrama no válida	El telegrama de envío tiene más de 224 bytes. La petición de envío ha sido cancelada por el módulo ET 200S Modbus/USS. Seleccione una longitud de telegrama menor.
0530	Petición de envío del maestro Modbus rechazada, ya que todavía no se ha consultado la respuesta del interlocutor a la petición de envío de escritura anterior del maestro Modbus.	Después de una petición de envío de lectura del maestro Modbus completada con éxito, hay que recoger del módulo la respuesta del interlocutor antes de iniciar una nueva petición de envío del maestro Modbus.
0551	Error de proceso de trama en la comunicación entre el módulo interfaz serie ET 200S Modbus/USS y la CPU. El error ha surgido al transferir un telegrama recibido del módulo interfaz serie ET 200S 1SI en la CPU.	El módulo y la CPU han cancelado la transferencia. Repita la petición de recepción. El módulo interfaz serie ET 200S Modbus/USS vuelve a enviar el mensaje recibido.

Código de error (W#16#...)	Descripción	Solución
0806	Tiempo de retardo entre caracteres excedido. Dos caracteres consecutivos no se han recibido dentro del tiempo de retardo entre caracteres.	El dispositivo interlocutor es demasiado lento o es defectuoso. Compruebe por qué falla el dispositivo interlocutor conectando, si procede, un dispositivo de comprobación de interfaz (FOXPG) a la línea de transmisión.
080A	Desbordamiento por exceso del búfer de recepción en el maestro durante la recepción del telegrama de respuesta.	Compruebe las configuraciones de protocolo del esclavo.
080C	Se ha detectado un error de transmisión (error de paridad, error de bit de parada, error de rebosamiento).	Los fallos en la línea de transmisión provocan repeticiones del telegrama reduciendo el paso de datos útiles. El riesgo de que se produzca un error no detectado aumenta. Modifique la configuración de su sistema o la disposición de la línea. Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos.
080D	BREAK: La línea de recepción hacia el interlocutor está interrumpida.	Restablezca la conexión o conecte al interlocutor.
0810	Error de paridad: Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión (rotura de línea) de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
0811	Error de trama de caracteres: Si el LED SF (rojo) está encendido, se ha interrumpido la línea de conexión (rotura de línea) de ambos interlocutores.	Compruebe la línea de enlace del interlocutor, o compruebe si la velocidad de transmisión, la paridad y el número de bits de parada coinciden en ambos dispositivos. Modifique la configuración de su sistema o la disposición de la línea.
0812	Se han recibido más caracteres después de que la interfaz serie haya puesto CTS en OFF.	Parametrice de nuevo el interlocutor o descargue la interfaz de serie más rápidamente.

Código de error (W#16#...)	Descripción	Solución
0830	Maestro: Se ha enviado un telegrama de solicitud y ya se ha agotado el tiempo de supervisión de respuesta sin que se haya reconocido el principio de un telegrama de respuesta. Esclavo: Broadcast no está permitido con este código de función.	Compruebe si la línea de transferencia está interrumpida (puede ser necesario un análisis de la interfaz). Compruebe que el módulo y el interlocutor tengan la misma configuración para los parámetros de protocolo velocidad de transferencia, número de bits de datos, paridad y número de bits de parada. Compruebe si el valor del tiempo de supervisión de respuesta en PtP_PARAM es suficientemente elevado. Compruebe si la dirección de esclavo indicada existe. El sistema maestro Modbus sólo puede emplear broadcast en los códigos de función habilitados para ello.
0831	Maestro: El primer carácter del telegrama de respuesta del esclavo es diferente de la dirección de esclavo enviada en el telegrama de solicitud (para el modo normal). Esclavo: El código de función recibido no está permitido.	Ha respondido el esclavo erróneo. Compruebe si la línea de transferencia está interrumpida (puede ser necesario un análisis de la interfaz). Este código de función no puede emplearse para este driver.
0832	Se ha superado el número máximo de bits o de registros o el número de bits no es divisible por 16 cuando se accede a las áreas de memoria SIMATIC de temporizadores o contadores.	Limite la cantidad máxima de bits a 2040 y la cantidad máxima de registros a 127. Acceso a temporizadores/contadores SIMATIC sólo en intervalos de 16 bits.
0833	El número de bits o de registros en el código de función FC 15/16 y el elemento de telegrama byte_count no concuerdan.	Corrija el número de bits/registros o byte_count.
0834	Se ha detectado una codificación de bits no permitida para "activar bit"/"desactivar bit".	Utilice para FC05 sólo las codificaciones 0000Hex o FF00Hex.
0835	Se ha detectado un subcódigo de diagnóstico no permitido (diferente a 0000Hex) en el código de función FC 08 "Loop Back Test".	Utilice para FC08 sólo el subcódigo 0000Hex.
0836	El valor creado internamente para la suma de verificación CRC16 no coincide con la suma de verificación CRC recibida.	Compruebe la formación de la suma de verificación CRC en el sistema maestro Modbus.
0837	Error en proceso de telegrama: el sistema maestro Modbus ha enviado un nuevo telegrama de solicitud antes de que el driver haya transferido el último telegrama de respuesta.	Aumente el timeout del telegrama de respuesta del esclavo en el sistema maestro Modbus.

Código de error (W#16#...)	Descripción	Solución
0850	La longitud del telegrama de recepción es superior a 224 bytes o mayor que la longitud de telegrama parametrizada.	Adaptar la longitud de telegrama del interlocutor.
0E20	El número de bits de datos de este acoplamiento debe ser 8. El driver no está listo.	Corrija la parametrización del driver.
0E21	El factor de multiplicación parametrizado para el tiempo de retardo entre caracteres no está entre 1 y 10. El driver trabaja con la configuración estándar 1.	Corrija la parametrización del driver.
0E22	El modo de operación parametrizado del driver no es válido. Debe indicarse "Modo normal" o "Modo con supresión de frecuencias perturbadoras". El driver no está listo.	Corrija la parametrización del driver.
0E23	Maestro: Se ha ajustado un valor no válido para el tiempo de supervisión de respuesta: Los valores válidos están entre 50 y 655000 ms. El driver no está listo. Esclavo: Se ha ajustado un valor no válido para la dirección de esclavo: La dirección de esclavo 0 no es válida. El driver no está listo.	Corrija la parametrización del driver. Corrija la parametrización del driver.
0E2E	Se ha producido un error al leer el archivo de parámetros de la interfaz. El driver no está listo.	Reinicie el maestro (Mains_ON).
0E40	El valor indicado para LEN en S_SEND es demasiado pequeño.	La longitud mínima es de 2 bytes.
0E41	El valor indicado para LEN en S_SEND es demasiado pequeño. Es necesaria una longitud mayor para el código de función transferido.	La longitud mínima para este código de función es de 6 bytes.
0E42	El código de función transferido no es válido.	Utilice sólo códigos de función permitidos.
0E43	Dirección del esclavo 0 (= broadcast) no permitida en este código de función.	Utilice la dirección de esclavo 0 sólo en los códigos de función adecuados.
0E44	El valor del "número de bits" transferido no está entre 1 y 2040.	El "número de bits" debe estar entre 1 y 2040.
0E45	El valor del "número de registros" transferido no está entre 1 y 127.	El "número de registros" debe estar entre 1 y 127.
0E46	Código de función 15 ó 16: Los valores del "número de bits" o "número de registros" transferidos no están entre 1 y 2040 o entre 1 y 127, respectivamente.	El "número de bits" o "número de registros" debe estar entre 1 y 2040 o entre 1 y 127, respectivamente.

Código de error (W#16#...)	Descripción	Solución
0E47	Código de función 15 ó 16: LEN para S_SEND no coincide con el "número de bits" o "número de registros" transmitido. LEN es demasiado pequeño.	Aumente LEN para SEND hasta que se transmita una cantidad suficiente de datos de usuario al módulo. Debe transmitirse una cantidad mayor de datos de usuario al módulo debido al "número de bits" o "número de registros".
0E48	Código de función 5: El código indicado en el DB de origen SEND para "activar bit" (FF00H) o para "desactivar bit" (0000H) es erróneo.	Los únicos códigos válidos son "activar bit" (FF00H), "desactivar bit" o 0000H.
0E49	Código de función 8: El código indicado en el DB de origen SEND para "código de diagnóstico" es erróneo.	El único código válido es "código de diagnóstico" 0000H.
0E4A	La longitud de este código de función es superior a la longitud máxima.	En el manual encontrará los datos de longitud máxima para cada código de función.
0E50	El maestro ha recibido una respuesta sin enviar.	Hay un esclavo u otro maestro en la red. Compruebe si la línea de transferencia está interrumpida (puede ser necesario un análisis de la interfaz).
0E51	Código de función erróneo: El código de función recibido en el telegrama de respuesta es distinto del código de función enviado.	Compruebe el equipo esclavo.
0E52	Rebase por defecto de bytes: El número de caracteres recibidos es inferior al que resultaría del contador de bytes del telegrama de respuesta o bien es inferior al esperado para este código de función.	Compruebe el equipo esclavo.
0E53	Rebase por exceso de bytes: El número de caracteres recibidos es superior al que resultaría del contador de bytes del telegrama de respuesta o bien es superior al esperado para este código de función.	Compruebe el equipo esclavo.
0E54	Error de contador de bytes: El contador de bytes recibido en el telegrama de respuesta es demasiado pequeño.	Compruebe el equipo esclavo.
0E55	Error de contador de bytes: El contador de bytes recibido en el telegrama de respuesta es incorrecto.	Compruebe el equipo esclavo.
0E56	Error de eco: Los datos del telegrama de respuesta (número de bits, ...) repetidos por el esclavo son distintos de los datos enviados en el telegrama de solicitud.	Compruebe el equipo esclavo.

Código de error (W#16#...)	Descripción	Solución
0E57	Error en la comprobación CRC: Al comprobar la suma de verificación CRC16 del telegrama de respuesta del esclavo se ha producido un error.	Compruebe el equipo esclavo.
0E61	Telegrama de respuesta con código de excepción 01: función no válida	Véase el manual del equipo esclavo.
0E62	Telegrama de respuesta con código de excepción 02: dirección de datos no válida	Véase el manual del equipo esclavo.
0E63	Telegrama de respuesta con código de excepción 03: valor de datos no válido	Véase el manual del equipo esclavo.
0E64	Telegrama de respuesta con código de excepción 04: fallo en el equipo correspondiente	Véase el manual del equipo esclavo.
0E65	Telegrama de respuesta con código de excepción 05: acuse	Véase el manual del equipo esclavo.
0E66	Telegrama de respuesta con código de excepción 06: ocupado, telegrama rechazado	Véase el manual del equipo esclavo.
0E67	Telegrama de respuesta con código de excepción 07: acuse negativo	Véase el manual del equipo esclavo.
1E0D	"Interrupción de petición por rearranque completo, rearranque o reset"	
1E0E	Error estático al llamar la SFC DP_RDDAT. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
1E0F	Error estático al llamar la SFC DP_WRDAT. El valor de respuesta RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB de instancia.	Cargue la variable SFCERR del DB de instancia.
1E10	Error estático al llamar la SFC RD_LGADR. El valor de retorno RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB instancia.	Cargue la variable SFCERR del DB instancia.
1E11	Error estático al llamar la SFC RDSYSST. El valor de retorno RET_VAL de la SFC está disponible para la evaluación en las variables SFCERR del DB instancia.	Cargue la variable SFCERR del DB instancia.

Código de error (W#16#...)	Descripción	Solución
1E20	Parámetro fuera del rango.	Introduzca un parámetro dentro del rango válido para la instrucción.
1E41	El número de bytes indicado en el parámetro LEN de las instrucciones no está permitido	Respete el rango de 1 a 256 bytes.

Evaluación de la variable SFCERR

Obtendrá más información sobre los errores ocurridos (1E) $0E_H$, (1E) $0F_H$, (1E) 10_H y (1E) 11_H de la clase de evento 30 a través de la variable SFCERR.

La variable SFCERR se puede cargar desde el DB instancia de la instrucción correspondiente.

Los mensajes de error, que se registran en la variable SFCERR, se encuentran en las funciones de sistema "DPRD_DAT: Leer datos coherentes de un esclavo DP normalizado (Página 6776)" y SFC 15 "DPWR_DAT: Escribir datos coherentes de un esclavo DP normalizado (Página 6778)" y RD_LGADR: Determinar todas las direcciones lógicas de un módulo (Página 7072).

Parámetro ERROR_NR, ERROR_INFO

Clase de errores del diagnóstico

Números de error de ... a	Descripción	Significado
1 ... 9	Error en la inicialización, instrucción y CP	La inicialización ha finalizado con errores en los números de error 1...9. El parámetro CP_START_ERROR es 1. No es posible la comunicación MODBUS con el sistema maestro.
10 ... 19	Error al procesar un código de función	Ha ocurrido un error al procesar un código de función en los números de error 10...19. El CP ha enviado una petición de procesamiento no permitida a la instrucción de comunicación. El error también será notificado al driver. Las siguientes peticiones se seguirán procesando.
90 ... 99	Otros errores	Se ha producido un error de procesamiento. El error no será notificado al driver. Las siguientes peticiones se seguirán procesando.

Números de error

ERROR_NR (decimal)	ERROR_INFO	Texto del error	Solución
0	0	Ningún error.	
1	SFC51->RET_VAL	Error al leer la lista de estado del sistema con la SFC51.	Analizar RET_VAL en ERROR_INFO, eliminar la causa.
2	S_SEND->STATUS, S_RCV->STATUS	Timeout en la inicialización del módulo o error al inicializar el módulo (error en la petición S_SEND).	Compruebe si para esta interfaz se ha configurado "MODBUS Slave" como protocolo. Compruebe si la "ID" indicada en la instrucción de comunicación es correcta. Analice ERROR_INFO.
11	Dirección inicial	Dirección inicial no válida transferida por el driver a la instrucción de comunicación.	Compruebe la dirección Modbus del sistema maestro Modbus.
12	Número de registros	Número de registros no válido transferido por el driver a la instrucción de comunicación. Número de registros = 0.	Compruebe el número de registros del sistema maestro Modbus, en caso necesario, reinicie el módulo (Mains_ON).
13	Número de registros	Número de registros no válido transferido por el driver a la instrucción de comunicación: Número de registros > 128.	Compruebe el número de registros del sistema maestro Modbus, en caso necesario, reinicie el módulo (Mains_ON).
14	Marcas M – Dirección final	Intento de acceso al área de memoria SIMATIC de las marcas más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.
15	Salidas Q – Dirección final Entradas I – Dirección final	Intento de acceso al área de memoria SIMATIC de las salidas más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.
16	Temporizadores T – Dirección final	Intento de acceso al área de memoria SIMATIC de los temporizadores más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.

ERROR_NR (decimal)	ERROR_INFO	Texto del error	Solución
17	Contadores C – Dirección final	Intento de acceso al área de memoria SIMATIC de los contadores más allá del límite final del área. Atención: La longitud de área en la CPU SIMATIC depende del tipo de CPU.	Reduzca la dirección inicial Modbus o la longitud de acceso en el sistema maestro Modbus.
18	0	Área de memoria SIMATIC no válida transferida por el driver a la instrucción de comunicación.	En caso necesario, reinicie el módulo (Mains_ON).
19		Error al acceder a la periferia SIMATIC.	Compruebe si la periferia necesaria existe y no tiene errores.
20	DB#	El DB no existe.	Incluya el DB en su proyecto.
21	DB#	Longitud de DB no válida	Aumente la longitud de DB.
22	DB#	DB# por debajo del valor mínimo de DB.	Cambie el valor mínimo de DB.
23	DB#	DB# por encima del valor máximo de DB.	Cambie el valor máximo de DB.
24	Dirección de marca	Marca por debajo del límite mínimo.	Cambie los límites mínimos de las marcas en el DB de conversión.
25	Dirección de marca	Marca por encima del límite máximo.	Cambie los límites máximos de las marcas en el DB de conversión.
26	Dirección de salida	Salida por debajo del límite mínimo.	Cambie los límites mínimos de las salidas en el DB de conversión.
27	Dirección de salida	Salida por encima del límite máximo.	Cambie los límites máximos de las salidas en el DB de conversión.
90	S_SEND-> STATUS	Error al transferir un telegrama de acuse al driver con S_SEND.	Analice la información de STATUS.
94	S_RCV->STATUS	Error al leer de SYSTAT con S_RCV (STATUS).	Analice la información de STATUS.

ET 200S 1SI de USS

S_USST: Enviar datos a un esclavo USS

Descripción

La instrucción S_USST ejecuta la transferencia de los datos de red (PZD y, si procede, datos PKW) a los esclavos según la estructura de datos de red utilizados.

Funcionamiento

La instrucción toma la parametrización del esclavo actual de la lista de sondeo (DB de parametrización) y envía los datos desde el DB de datos de red. Evalúa la palabra de control de comunicación del esclavo actual (lanzamiento de una petición de PKW o acuse de un

informe de modificación de parámetros), completa los datos de envío de USS y los transfiere al búfer de envío del DB de procesador de comunicaciones. Finalmente, lanza la transferencia de los datos de red al esclavo mediante la instrucción S_SEND .

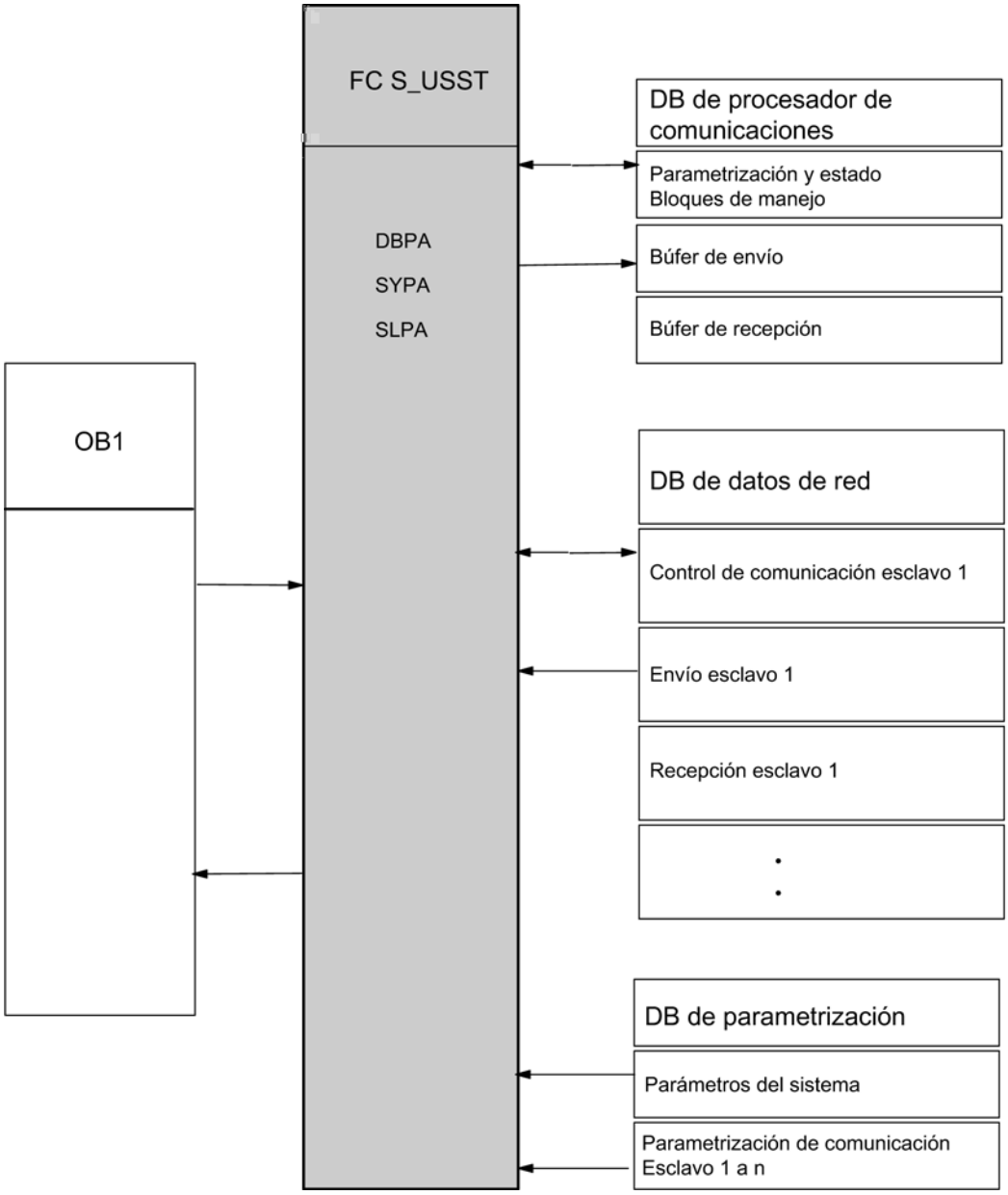
Si la instrucción detecta un error de parametrización en el DB de parametrización, se guarda una señal de error en el byte PAFE 2 (byte de error de parametrización) del DB de datos de red.

La instrucción se llama una vez en cada ciclo del sistema de automatización.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
DBPA	INPUT	INT	Número de bloque del DB de parametrización Específico de la CPU (no se admite el valor cero)
SYPA	INPUT	INT	Dirección inicial de los parámetros de sistema en el DB de parametrización 0 <= SYPA <= 8174
SLPA	INPUT	INT	Dirección inicial de los parámetros del esclavo en el DB de parametrización 0 <= SLPA <= 8184

Estructura del programa de S_USST



Consulte también

Resumen de la comunicación USS (Página 1192)

S_USSR: Recibir datos de un esclavo USS

Descripción

La instrucción S_USSR ejecuta la recepción de datos de red (datos de PZD y, dado el caso, también de PKW) de los esclavos según la estructura de datos de red utilizada.

Funcionamiento

La instrucción toma la parametrización del esclavo actual de la lista de sondeo (DB de parametrización) y evalúa la palabra de estado del bloque TRANSMIT.

Si la petición actual ha finalizado sin errores (bit 9 = 0 en la palabra de estado de comunicación del DB de datos de red), los datos entrantes se transfieren del búfer de recepción del DB de procesador de comunicaciones al DB de datos de red y se evalúan. A continuación, se actualiza la palabra de estado de comunicación en el DB de datos de red.

Si la petición actual no ha finalizado sin errores (bit 9 = 1 en la palabra de estado de comunicación del DB de datos de red), el búfer de recepción del DB de procesador de comunicaciones no acepta los datos del esclavo actual. La instrucción indica esta situación en la palabra de estado de comunicación del DB de datos de red y registra la causa del error en la palabra de errores de comunicación.

Si el bloque detecta un error de parametrización en el DB de parametrización, se guarda una señal de error en el byte PAFE 1 del DB de datos de red.

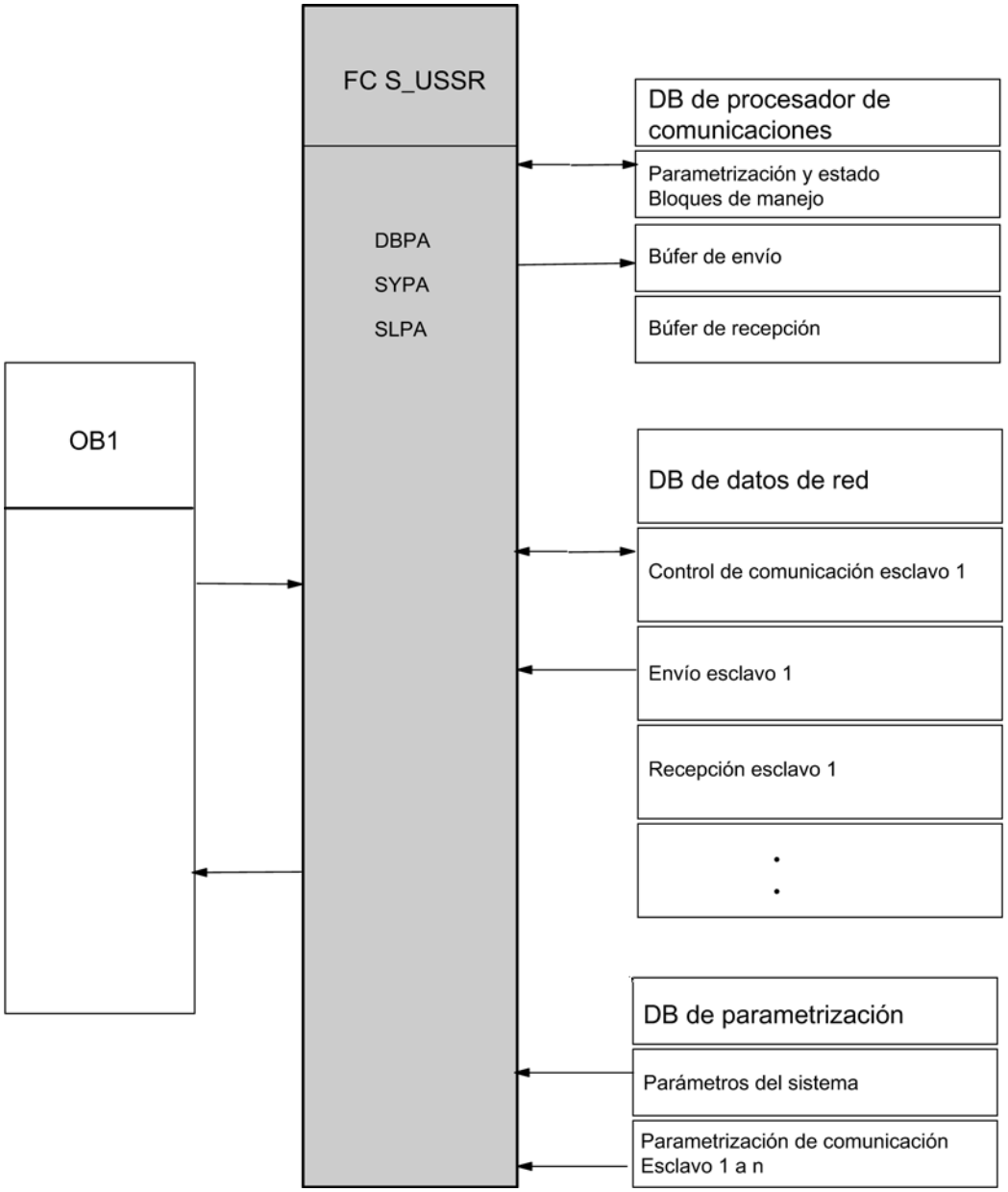
La instrucción se llama una vez en cada ciclo del sistema de automatización.

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
DBPA	INPUT	INT	Número de bloque del DB de parametrización Específico de la CPU (no se admite el valor cero)
SYPA	INPUT	INT	Dirección inicial de los parámetros de sistema en el DB de parametrización 0 <= SYPA <= 8174
SLPA	INPUT	INT	Dirección inicial de los parámetros del esclavo en el DB de parametrización 0 <= SLPA <= 8184

Estructura del programa de S_USSR

La figura siguiente muestra la estructura del programa de S_USSR.



Regulación

Nota

Los parámetros de la instrucción U_USST corresponden a los parámetros de la instrucción S_USSR. Ambas instrucciones acceden a la misma parametrización (parámetros del sistema y del esclavo) en el DB de parametrización y, por ello, deben parametrizarse de forma idéntica.

Consulte también

Resumen de la comunicación USS (Página 1192)

S_USSI: Inicializar USS

Descripción

La instrucción S_USSI es una función opcional.

Si se llama esta instrucción al arrancar el sistema S7, se generan los bloques de datos de procesador de comunicaciones, datos de red y parametrización necesarios para la comunicación. Asimismo, se asignan valores predeterminados a DBPA. La instrucción S_USSI sirve para generar las áreas de datos mencionadas y asignarles valores predeterminados, sólo cuando la estructura de los datos de red es igual para todos los esclavos.

Funcionamiento

Cuando se llama, la instrucción comprueba primero la plausibilidad de su parametrización en relación con el número de esclavos, la estructura de los datos de red, el número de nodo de inicio y las repeticiones de PKW. Si el bloque detecta un error, no se ejecuta ni la generación ni la asignación de valores predeterminados de los bloques de datos. La CPU pasa al estado operativo STOP y el usuario obtiene un mensaje de error a través del byte de error de la instrucción S_USSI. Una vez solucionado el error de parametrización, antes de reiniciar deben borrarse todos los bloques de datos generados.

Después de verificar la plausibilidad, el bloque comprueba si los bloques de datos que deben generarse ya existen:

- Si los bloques de datos que deben generarse aún no existen, se crean y se asignan valores predeterminados a DBPA.
- Si ya existen, se comprueba la longitud de cada bloque. Si el DB es lo suficientemente largo, se vuelve a crear la asignación de valores predeterminados del DB de parametrización y se borra el contenido del DB de datos de red y del DB de procesador de comunicaciones. Si un DB es demasiado corto, la CPU pasa al estado operativo STOP. El usuario reconoce el DB defectuoso en el byte de visualización de la instrucción S_USSI. Para solucionar el error, deben borrarse por completo los tres bloques de datos. Cuando se realiza un rearranque completo, se crean de nuevo y se asignan valores predeterminados al DB de parametrización.

La instrucción S_USSI tiene que llamarse una vez durante el arranque del sistema (OB100).

Parámetros

Parámetro	Declaración	Tipo de datos	Descripción
SANZ	INPUT	INT	Número de esclavos con la misma estructura de datos de red (parámetros del sistema en DBPA) 1 <= SANZ <= 31
TNU1	INPUT	INT	Número de estación de inicio (número de equipo) 0 <= TNU1 <= 31
PKW	INPUT	INT	PKW, cantidad Número de palabras de la interfaz de PKW 0, 3 ó 4
PZD	INPUT	INT	PZD, cantidad Número de palabras de la interfaz PZD 0 <= PZD <= 16
DBND	INPUT	INT	Número de DB de datos de red Específico de la CPU (no se admite el valor cero)
DBPA	INPUT	INT	Número del DB de parametrización Específico de la CPU (no se admite el valor cero)
DBCP	INPUT	INT	Número del DB de procesador de comunicaciones Específico de la CPU (no se admite el valor cero)
WDH	INPUT	INT	Número de repeticiones permitidas de una petición de PKW 0 <= WDH <= 32767
ANZ	OUTPUT	BYTE	Byte de error 0: ningún error 1: número de esclavos excesivo 2: datos no permitidos para estructura de datos de red 3: DB de parametrización: número no admisible o DB demasiado corto 4: DB de datos de red: número no admisible o DB demasiado corto 5: error en número de estación 6: DB del procesador de comunicaciones: número no admisible o DB demasiado corto 7: libre 8: Contador de repeticiones: valor incorrecto

Consulte también

Resumen de la comunicación USS (Página 1192)

DB de datos de red

Descripción

El DB de datos de red constituye la interfaz entre el programa de comunicación y el programa de control. El usuario debe preparar este bloque "vacío" con la longitud suficiente.

Funcionamiento

Estos bloques de datos pueden crearse con la instrucción S_USSI al arrancar la CPU; también se les pueden asignar valores predeterminados (sólo DBPA) o introducirse manualmente.

En el búfer de envío del DB de datos de red, asignado al esclavo por el programa de control, sólo se registran los datos de envío para un esclavo. Los datos de respuesta del esclavo se toman desde el búfer de recepción correspondiente (tras la evaluación del bit 9 en la palabra de control de comunicación). Las palabras de estado permiten controlar la comunicación y la palabra de mando, el lanzamiento selectivo de una petición de parametrización.

La interfaz de comunicación contiene los datos siguientes una vez para cada esclavo:

- Datos de comunicación relacionados con el esclavo (control de comunicación, seguimiento, 6 palabras de datos)
- Búfer para la petición de PKW en curso (sólo si existe un área PKW)
- Búfer de envío para datos de red (máximo 20 palabras de datos)
- Búfer de recepción para datos de red (máximo 20 palabras de datos)

La longitud del búfer de envío y de recepción depende de la estructura de datos de red seleccionada. Si falta la interfaz de PKW, se suprime el búfer para la petición de PKW actual.

La longitud total del DB de datos de red necesario depende de la cantidad de esclavos y de la estructura de datos de red empleada.

Número de palabras de datos por esclavo = 2 x (PKW + PZD) + PKW + 6

Con PKW = 0, 3 ó 4 y 0 <= PZD <= 16

Ejemplo: Un accionamiento con un área PKW de 3 palabras y un área PZD de 2 palabras ocupa 19 palabras de datos en el DB de datos de red.

El DB de datos de red tiene 1550 palabras de datos con 31 esclavos y la longitud máxima de datos de red. DBW0 está reservado.

Asignación de datos del esclavo en el DB de datos de red con 4 palabras en el área PKW y de 0 a 16 palabras en el área PZD

DBWn	Palabra de control de comunicación (KSTW)	Control de la comunicación
DBWn+2	Interna	Seguimiento de la comunicación
DBWn+4	Palabra de estado de comunicación	Estado de error
DBWn+6	Palabra de error de comunicación	Contador de intentos de PKW
DBW n+8	Interna	Error de parámetro
DBW n+10	Byte PAFE 1, byte PAFE 2	

DBW n+12	ID de parámetro	PKE	Búfer para petición de PKW actual	
DBW n+14	Índice	IND		
DBW n+16	Valor de parámetro 1	PWE1		
DBW n+18	Valor de parámetro 2	PWE2		
DBW n+20	ID de parámetro	PKE	Área PKW	Búfer de envío
DBW n+22	Índice	IND		
DBW n+24	Valor de parámetro 1	PWE1		
DBW n+26	Valor de parámetro 2	PWE2		
DBW n+28	Palabra de mando (STW)	PZD1	Área PZD (máx. 16 palabras PZD)	
DBW n+30	Valor de consigna principal (HSW)	PZD2		
DBW n+32	Valor consigna/palabra de mando adicional	PZD3		
DBW n+34	Valor consigna/palabra de mando adicional	PZD4		
...	...			
DBW n+58	Valor consigna/palabra de mando adicional	PZD16	Área PKW	Búfer de recepción
DBW n+60	ID de parámetro	PKE		
DBW n+62	Índice	IND		
DBW n+64	Valor de parámetro 1	PWE1		
DBW n+66	Valor de parámetro 2	PWE2		
DBW n+68	Palabra de estado (ZSW)	PZD1	Área PZD (máx. 16 palabras PZD)	
DBW n+70	Valor real principal (HIW)	PZD2		
DBW n+72	Valor real/palabra de estado adicional	PZD3		
DBW n+74	Valor real/palabra de estado adicional	PZD4		
...	...			
DBW n+98	Valor real/palabra de estado adicional	PZD16		
	•			
(n = 2,4,6...)	•			

Nota

Si hay un área PKW, el búfer para las peticiones de PKW actuales y el área PKW del búfer de envío no existen.

Palabra de control de comunicación KSTW (DBW n)

Los bits de la palabra de control de comunicación coordinan el programa de usuario y la instrucción S_USST.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Inicio de petición de PKW
  El usuario activa el bit 0 cuando hay una nueva petición de PKW nueva en el búfer de envío y debe procesarse. La instrucción desactiva el bit cuando se ha aceptado la petición de PKW.
- Bit 1: Aceptación del informe de modificación de parámetros
  El usuario activa el bit 1 cuando se ha aceptado el informe de modificación de parámetros. La instrucción desactiva el bit para acusar la admisión. Tras este acuse, el esclavo prosigue con el procesamiento interrumpido de la petición en curso o transfiere el siguiente informe de modificación de parámetros.

Palabra de estado de comunicación (DBW n+4)

Los bits de la palabra de estado de comunicación son activados por las instrucciones S_USST y S_USSR.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Petición de PKW en proceso
  La instrucción S_USST activa el bit 0 cuando se ha aceptado la petición de PKW y la ID de parámetro (PKE) contiene una ID de petición válida. La instrucción S_USSR desactiva el bit cuando se ha ejecutado la petición de PKW (con o sin errores) o cuando la interfaz de PKW está defectuosa.
- Bit 1: La petición ha finalizado sin errores
  La instrucción S_USSR activa el bit 1 cuando se ha ejecutado una petición de PKW sin errores. La respuesta debe tomarse del búfer de recepción. La instrucción S_USST desactiva el bit cuando se lanza una nueva petición de PKW.

Nota

Las peticiones de PKW para el esclavo se procesan en el orden de la lista de sondeo (DBPA). Cada esclavo sólo tiene una petición activa. Si hay más de un esclavo registrado en la lista de sondeo, los datos de respuesta para una nueva petición de PKW sólo están disponibles en un flanco positivo del bit 1 (o del bit 2).

- Bit 2: La petición de PKW ha finalizado con errores
  La instrucción S_USSR activa el bit 2 en caso de ID de respuesta en PKE. El número de error se encuentra en PWE de la respuesta del esclavo. La instrucción S_USST desactiva el bit cuando se lanza una nueva petición de PKW.

  Nota

  La última petición de PKW que ha transferido el usuario se guarda en la interfaz de envío tras su procesamiento. La transferencia al esclavo se repite hasta que se introduce una nueva petición. De este modo, es posible que se requieran respuestas adicionales en el programa de usuario si la petición de PKW de estado finaliza con errores (bit 2) y con un fallo de interfaz de PKW (bit 4).

- Bit 3: ID de petición de PKW no válido.
  La instrucción S_USST activa el bit 3 cuando se determina la ID de petición 15 en PKE o cuando se registra el índice 255 en la ID de petición 4. La instrucción S_USST desactiva el bit cuando se lanza la siguiente petición de PKW con una ID de petición válida en PKE.
- Bit 4: Interfaz de PKW con errores (desbordamiento del contador).
  La instrucción S_USSR activa el bit 4 cuando el esclavo no responde a una petición de PKW tras repetirla un número de veces parametrizable (parámetro WDH en el DB de parametrización) o cuando la ID de respuesta en PKE es 8. La instrucción S_USSR desactiva el bit cuando se lanza y ejecuta correctamente una nueva petición de PKW.
- Bit 5: Los datos de respuesta contienen un informe de modificación de parámetros.
  La instrucción S_USSR activa el bit 5 cuando existe un informe de modificación de parámetros del esclavo (ID de respuesta 9 - 12 y bit de conmutación 11 invertido). La instrucción S_USST desactiva el bit cuando el usuario acusa el informe de modificación de parámetros (palabra de mando de comunicación, bit 1).
- Bit 6: Fallo de funcionamiento en el esclavo.
  La instrucción S_USSR activa y desactiva el bit 6. La instrucción evalúa la palabra de estado (bit 3) del esclavo.
- Bit 7: Hay una advertencia del esclavo
  La instrucción S_USSR activa y desactiva el bit 7. La instrucción evalúa la palabra de estado (bit 7) del esclavo.
- Bit 8: Se requiere control desde el sistema de automatización.
  La instrucción S_USSR activa y desactiva el bit 8. La instrucción evalúa la palabra de estado (bit 9) y la palabra de control (bit 10).
- Bit 9: Error colectivo de comunicación.
  La instrucción S_USSR activa y desactiva el bit 9. La instrucción evalúa las respuestas de los bloques estándar S_SEND y S_RCV y comprueba el telegrama recibido en relación con ADR, STX, BCC y LGE. La instrucción comunica además que se ha excedido el tiempo de vigilancia de telegrama.

  Nota

  Los datos de recepción del DB de datos de red sólo son válidos si el bit 9 = 0.

Estructura de la palabra de error de comunicación (DBW n+6)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

- Bit 0: Error de direccionamiento (ADR)
- Bit 3: Inicio de telegrama no detectado (el primer carácter no es STX)
- Bit 4: Error de carácter de comprobación de bloque (BCC)
- Bit 6: Error de longitud de telegrama (LGE)
  La instrucción S_USSR activa los bits 0, 3, 4 y 6 cuando se detecta un error al verificar el telegrama recibido (ADR, STX, BCC, LGE).
- Bit 7: Tiempo de supervisión de telegrama transcurrido
  La instrucción S_USSR activa el bit 7 cuando el tiempo entre la transmisión del telegrama por parte del maestro al esclavo y la llegada de la respuesta del esclavo supera el tiempo permitido calculado por el programa (tiempo de vigilancia de telegrama).

Los demás bits no se utilizan.

Byte PAFE 1

Mensaje de error de la instrucción S_USSR, error de parametrización en el DB de parametrización

- Valor 0: No hay fallo
- Valor 1: Datos erróneos para PKW / PZD

Byte PAFE 2

Mensaje de error de la instrucción S_USST, error de parametrización en el DB de parametrización

- Valor 0: No hay fallo
- Valor 1: Datos erróneos para PKW / PZD

ID de parámetro PKE en el búfer de envío

El usuario debe introducir el número de parámetros (bits 0 a 10) y el ID de petición (bits 12 a 15). El bit de conmutación para el informe de modificación de parámetros (bit 11) es enmascarado por las instrucciones S_USSR y S_USST.

DB de parametrización

Descripción

El DB de parametrización contiene los parámetros de programa necesarios para controlar la comunicación. El usuario debe crear este bloque y asignar valores predeterminados a la

configuración del sistema de comunicación de forma pertinente (S_USSI o manualmente). Los esclavos del bus se procesan en el orden de la entrada en DBPA (lista de sondeo).

Un esclavo también puede registrarse varias veces en el DB de parametrización, con lo que su prioridad aumenta de forma efectiva.

Funcionamiento

La longitud del DB de parametrización depende del número n de los esclavos a los que debe accederse en un ciclo de bus.
Número de palabras de datos del DB de parametrización = (n x 4) + 5.

Para cada comunicación con un esclavo se requieren 4 palabras de datos y para los parámetros de sistema se ocupan 4 palabras de datos una sola vez. DBW0 está reservado.

Asignación del bloque de datos

DBW 0	Libre	Parámetros del sistema
DBW 2	DBCP	
DBW 4	SANZ	
DBW 6	SLAV	
DBW 8	WDH	
DBW 10	Número de PKW, número de PZD	Comunicación Registro de parámetros esclavo 1
DBW 12	TUN	
DBW 14	DBND	
DBW 16	KSTW	
DBW 18	Número de PKW, número de PZD	Comunicación Registro de parámetros esclavo 2
DBW 20	TUN	
DBW 22	DBND	
DBW 24	KSTW	
	Número de PKW, número de PZD	Comunicación Registro de parámetros esclavo n
	TUN	
	DBND	
DBW (n x 8 + 8)	KSTW	

Parámetros del sistema

DBCP	Número de bloque del DB de procesador de comunicaciones
SANZ	Número total de registros de parámetros de esclavo en el DB de parametrización Si hay que acceder a determinados esclavos con mayor frecuencia que a otros dentro de un ciclo de bus, registre los parámetros de esclavo varias veces en el DB de parametrización. El parámetro de sistema SANZ debe adaptarse de la forma correspondiente.

SLAV	Número correlativo del esclavo actual. Es requerido por la FC S_USST y la FC S_USSR para determinar el registro de parámetros actual. A esta palabra de datos debe asignársele el valor predeterminado 1 De ello se encarga la FC S_USSI, siempre que se utilice.
WDH	Número de repeticiones permitidas de una petición de PKW (rango de valores: 0 a 32767). Si la petición actual de PKW no finaliza dentro del número de registro, se notifica que la interfaz PKW está defectuosa.

Parametrización de comunicación de esclavo:

Número de PKW, número de PZD	Definición de la estructura de datos de red Byte izquierdo: Número de palabras para el área PKW (0, 3, 4) Byte derecho: Número de palabras para el área PZD (0 a 16) Los datos divergentes se reconocen como errores de parametrización (por las FCs S_USST y S_USSR) y se registran en los bytes PAFE 1 y 2 del DB de datos de red.
TUN	Número de estación, que equivale a la dirección de bus ajustada en el accionamiento (0 a 31).
DBND	Número de bloque del DB de datos de red.
KSTW	Dirección de la palabra de mando de comunicación KSTW para el esclavo en el DB de datos de red.

DB del procesador de comunicaciones

Descripción

A través de este bloque de datos tiene lugar el intercambio de datos entre la CPU y el módulo de interfaz serie ET 200S Modbus/USS. El usuario debe poner este bloque a disposición con la longitud suficiente. El DB del procesador de comunicaciones debe tener una longitud mínima de 50 palabras (DBW 0 a 98)

Estructura del DB del procesador de comunicaciones

DBW 0	Estado de la comunicación		ENVIAR y RECIBIR
DBW 2	Número máximo de ciclos al esperar a recibir	Contador de ciclos para formar timeout al esperar a recibir	FC17
DBW 4	Pausa de inicio medida		FC17
DBW 6	Duración dle último ciclo (OB1_MIN_CYCLE)		FC17, OB1
DBW 8	Longitud del telegrama de envío (LEN)		ENVIAR
DBB10	Libre		

DBB 11 : : DBB 54	Búfer de transmisión	Enviar telegrama a módulo (La longitud se rige por la estructura de los datos útiles del esclavo actual)
DBB 55 : : DBB 98	Búfer de recepción	Telegrama recibido por el módulo (La longitud se rige por la estructura de los datos útiles del esclavo actual)

Estado de comunicación DBW0

Bit n.º		
0	Entrada REQ para S_SEND	Este bit se pone a 0 cuando se activa el bit 8.
1	Entrada R para S_SEND	S_USST pone cíclicamente este bit a cero.
2	Salida DONE de S_SEND	
3	Salida ERROR de S_SEND	
4	Entrada EN_R para S_RCV	S_USSR pone cíclicamente este bit a 1.
5	Entrada R para S_RCV	S_USSR pone cíclicamente este bit a 0.
6	Salida NDR de S_RCV	
7	Salida ERROR de S_RCV	
8	Petición en curso (bit DONE de S_SEND guardado)	S_USST pone cíclicamente este bit a 0.

Duración dle último ciclo DBW6

S_USST mide con este parámetro el tiempo de respuesta de un esclavo. El programa de usuario debe copiar en este parámetro el tiempo de ciclo del sistema de automatización (OB1_MIN_CYCLE) antes de cada llamada del S_USST .

CPs SIMATIC NET

Industrial Ethernet

Introducción de Ethernet

Indicaciones generales

Instrucciones disponibles

La lista siguiente muestra los números de los bloques de programa (instrucciones) tal y como se suministran. Los números pueden ser modificados por el usuario.

Tenga en cuenta que se tienen que utilizar instrucciones diferentes para S7-300 y S7-400 (bibliotecas separadas).

Servicio de comunicación / área de funciones	Tipo de instrucciones	Biblioteca	
		CP 300	CP 400
SEND / RECEIVE (Servicios de comunicación abiertos)	AG_SEND	x	x
	AG_RECV	x	x
	AG_LSEND	x [2]	x
	AG_LRECV	x [2]	x
	AG_LOCK	x	x
	AG_UNLOCK	x	x
	AG_CNTEX	x	x [3]
	AG_CNTRL	x	x [3]
Conexiones de comunicación programadas	IP_CONFIG	x	x
Comunicación S7	BSEND	x	
	BRCV	x	
	PUT	x	
	GET	x	
	USEND	x	
	URCV	x	
	C_CNTRL	x	
FTP	FTP_CMD	x	x
PROFINET IO	PNIO_SEND	x	
	PNIO_RECV	x	
	PNIO_RW_REC	x	
	PNIO_ALARM	x	

1) El PN_InOut / PN_InOut_Fast se entrega conjuntamente con la herramienta de ingeniería SIMATIC iMap.

2) No debe utilizarse para CPs actuales y ya no forma parte de la entrega de la biblioteca actual SIMATIC_NET_CP

3) Depende del tipo de CP

¿Qué versión de instrucciones se debe utilizar?

Las descripciones siguientes contienen también referencias a comportamientos discrepantes en caso de versiones de instrucciones diferentes. Preste también atención a la identificación de versión de las instrucciones utilizadas por usted.

Encontrará una edición actual del historial de la versión para las instrucciones SIMATIC NET bajo la siguiente ID de artículo:

9836605 (http://support.automation.siemens.com/WW/view/de/9836605)

Las bibliotecas globales instaladas con el Automation Workbench contienen las versiones de instrucciones actuales del momento de la habilitación del Workbench.

Nota

Se recomienda utilizar siempre las versiones de instrucciones actuales para todos los tipos de módulos.

Encontrará informaciones sobre las versiones de instrucciones actuales así como las instrucciones actuales en nuestro Customer Support en Internet, desde donde las podrá descargar:

http://support.automation.siemens.com/WW/view/de/8797900

En el caso de tipos de módulos menos recientes, esta recomendación presupone que usted utilice la versión de firmware actual para este tipo de módulo.

Instrucciones en caso de repuesto

Se entiende por caso de sustitución en este contexto la sustitución de un módulo por otro módulo de versión eventualmente más moderna.

Nota

Tenga en cuenta que, en el caso de utilizar recambios, en el programa de usuario se deben utilizar solo las instrucciones autorizadas para el tipo de CP configurado.

Se recomienda utilizar siempre las versiones de instrucciones actuales para todos los tipos de módulos. En el caso de tipos de módulos menos recientes, esta recomendación presupone que usted utilice la versión de firmware actual para este tipo de módulo.

Más información acerca de casos de sustitución, encontrará en nuestro Customer Support en Internet.

Los manuales de producto específicos de los CPs S7 informan sobre la compatibilidad de los CPs S7 y las instrucciones correspondientes.

Consulte también

Manual de programación "Funciones (FC) y bloques de función (FB) para CPs SIMATIC NET S7" (http://support.automation.siemens.com/ww/view/de/30564821)

Parametrización de la llamada de las instrucciones

Antes de pasar a describir en detalle las instrucciones, se ofrecen aquí algunas informaciones de índole general sobre la llamada y la parametrización de las instrucciones.

Aquí se pueden dar informaciones generales sobre los siguientes grupos de parámetros, existentes en todas las instrucciones:

- Parámetros para coordinación de CP y enlaces (parámetros de entrada)
- Parámetros para definir una determinada área de datos de la CPU (parámetros de entrada)
- Informaciones sobre estado/status (parámetros de salida)

Llamar instrucciones de comunicación para S7-300

ATENCIÓN
No se permite llamar las instrucciones de comunicación para S7-300 en varios niveles de proceso. Si, por ejemplo, llama una instrucción de comunicación en OB1 y en OB35, el procesamiento de la instrucción podría ser interrumpido por el OB de prioridad respectivamente superior. Si llama instrucciones en varios OBs, tiene que procurar, a través de la programación, que una instrucción de comunicación en procesamiento no sea interrumpida por otra instrucción de comunicación (por ejemplo, por medio de las instrucciones del sistema Bloquear/Liberar alarmas).

Parámetros para definir una determinada área de datos de la CPU (parámetros de entrada)

Indicar el área de datos en la CPU

Al llamar una instrucción transmite usted la dirección y la longitud del área de datos de la CPU en la que se deben disponer o almacenar datos útiles o bien que puede contener otras informaciones relativas a la parametrización.

Para el direccionamiento de esta área se utiliza el tipo de datos del puntero ANY.

Informaciones sobre estado/status (parámetros de salida)

Evaluar indicaciones de estados

Para la evaluación del estado se tienen que evaluar estos parámetros en el programa de usuario:

- DONE o NDR
  Estos parámetros (DONE en el caso de peticiones de envío o NDR para peticiones de recepción) notifica la conclusión (con resultado positivo) de una ejecución de la petición.
- ERROR
  Notifica que la petición no se ha podido ejecutar sin errores.
- STATUS
  El parámetro proporciona información detallada sobre la ejecución de la petición. Se pueden proporcionar indicaciones de estado ya durante la ejecución de la petición (DONE=0 y ERROR=0).

Nota

Tenga en cuenta que las indicaciones de estado DONE, NDR, ERROR y STATUS se actualizan con cada llamada de una instrucción.

Indicaciones de estado al arrancar el CP

Al rearrancar el CP Ethernet, los parámetros de salida de la instrucción se resetean como sigue:

- DONE = 0
- NDR = 0
- ERROR = 0
- STATUS =
  - 8180_H en el caso de AG_RECV / AG_LRECV
  - 8181_H en el caso de AG_SEND / AG_LSEND / AG_SSEND

Instrucciones para los Servicios de comunicación abiertos (interfaz SEND/RECEIVE)

Instrucciones para la interfaz SEND/RECEIVE

Panorámica

En la interfaz SEND/RECEIVE están disponibles las siguientes instrucciones para la transmisión de datos:

Instrucción	utilizable para [1]		Significado
S7-300	S7-400		
AG_SEND	x	x	para enviar datos
AG_RECV	x	x	para recibir datos
AG_LSEND		x	para enviar datos
AG_LRECV		x	para recibir datos
AG_SSEND		x	para enviar datos
AG_SRECV		x	para recibir datos

1) Observaciones sobre las instrucciones para S7-300 y S7-400

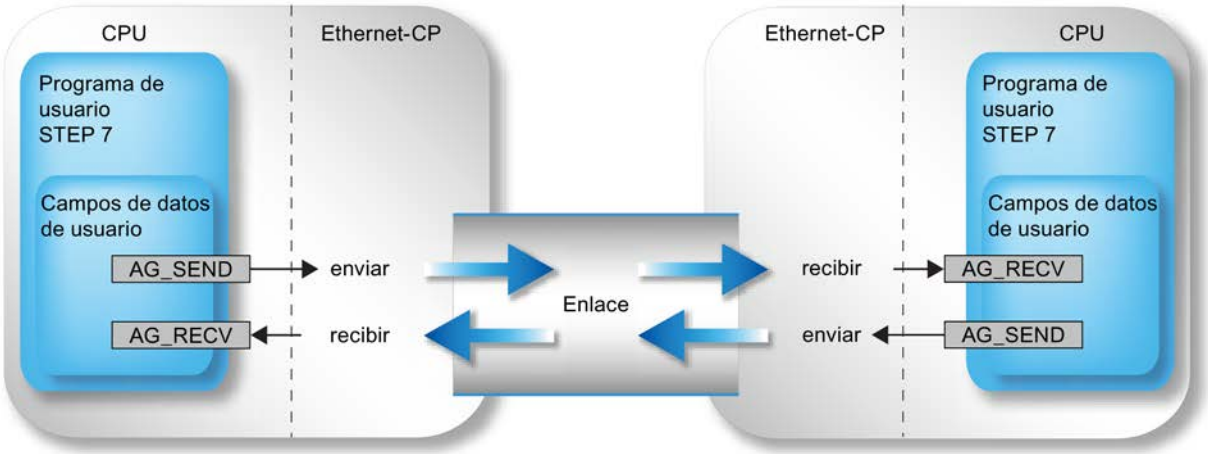
- para S7-300 rige:
  - En las versiones actuales de los CPs Ethernet se utilizan exclusivamente AG_SEND y AG_RECV; la longitud de datos puede ser aquí de hasta 8192 bytes.
  - En CPs S7–300 (hasta 6GK7 343–1EX10–0XE0 con versión de firmware V2.2), se deberá utilizar AG_LRECV para conexiones TCP en lugar de AG_RECV.
- para S7-400 rige:
  - En el caso de AG_SEND / AG_RECV, la longitud de datos por petición está limitada a <=240 bytes.
    Los registros más largos (hasta 8192 bytes) se pueden transmitir con la ayuda de las instrucciones AG_LSEND o AG_LRECV.
  - AG_SSEND y AG_SRECV sirven para la transmisión acelerada de datos, utilizando una comunicación optimizada entre la CPU y el CP en la estación S7. La comunicación rápida no tiene repercusiones en la comunicación LAN. Las dos instrucciones son soportadas a partir de STEP 7 V5.4 SP3.
  - En S7–400 no se puede utilizar AG_RECV para conexiones TCP, sino solo las instrucciones AG_LRECV o AG_SRECV.

Informaciones adicionales

Infórmese sobre al área de datos a la que se da soporte para el CP S7 utilizado por usted en el manual específico. Encontrará una tabla general de las versiones de instrucciones en el historial de bloques SIMATIC NET.

Uso

El esquema siguiente explica la aplicación de las instrucciones aquí descritos para la transferencia de datos bidireccional a través de una conexión configurada.



Indicar el área de datos en la CPU

Al llamar una instrucción se transfiere la dirección y la longitud del área de datos a la CPU. Tenga en cuenta que la longitud máxima del área de datos depende del tipo y de la versión de instrucción utilizados.

- AG_SEND y AG_RECV
  Con estas instrucciones se pueden enviar o recibir, hasta la versión V3.0, 240 bytes como máximo. Las versiones actuales permiten para S7-300 un área de datos de hasta 8192 bytes. En el caso de S7-400 se tienen que seguir utilizando AG_LSEND / AG_LRECV para la transmisión de áreas de datos mayores.
- AG_LSEND / AG_LRECV
  En el caso de los CPs de S7-400, así como en el de versiones anteriores de S7-300 solo es posible transmitir áreas de datos mayores por medio de AG_LSEND o AG_LRECV. Consulte en la información de producto correspondiente al CP el área de datos a que se da soporte.
- AG_SSEND / AG_SRECV
  Con CPs de S7−400, que soportan la comunicación PROFINET junto con CPUs a partir de la versión 5.1, los datos se pueden transmitir por medio de AG_SSEND o AG_SRECV con una velocidad de transmisión superior (esto no es válido para el CP 443−1 Advanced 6GK7 443–1EX41–0XE0).
  Los tipos de CPs compatibles con CPUs a partir de la versión 5.1 se pueden consultar en el manual de producto del CP (capítulo "Condiciones para el uso").

La tabla siguiente indica los límites válidos para los diferentes tipos de conexión.

INSTRUCCIÓN	ISO-Transport	ISO-on-TCP	TCP	UDP
AG_LSEND (S7-400) AG_SEND (S7-300)	8192 bytes	8192 bytes	8192 bytes	2048 bytes
AG_SEND (S7-400)	240 bytes	240 bytes	240 bytes	240 bytes
AG_LRECV (S7-400) AG_RECV (S7-300)	8192 bytes	8192 bytes	8192 bytes	2048 bytes
AG_RECV (S7-400)	240 bytes	240 bytes	240 bytes	240 bytes
AG_SSEND (S7-400) AG_SRECV (S7-400)	1452 bytes	1452 bytes	1452 bytes	1452 bytes

Nota

En cuanto a la longitud del área de datos transmitible en el caso de versiones menos recientes de los CPs Ethernet, tenga en cuenta lo dicho en la información de producto / el manual de producto del CP Ethernet utilizado.

Aplicación sin encabezado de petición

En el caso de una conexión especificada, los parámetros de dirección y petición son definidos por la configuración de la conexión. Por lo tanto, el programa de usuario solo proporciona los datos de usuario en el área de datos UDP al enviar con AG_SEND / AG_LSEND / AG_SSEND, o los recibe con AG_RECV / AG_LRECV / AG_SRECV.

Aplicación con encabezado de petición

Las conexiones UDP libres exigen un encabezado de petición en el área de datos de usuario.

Vea en la ilustración siguiente la estructura del búfer de peticiones y el significado y el emplazamiento (high byte / low byte) de los parámetros en el encabezado de la petición.

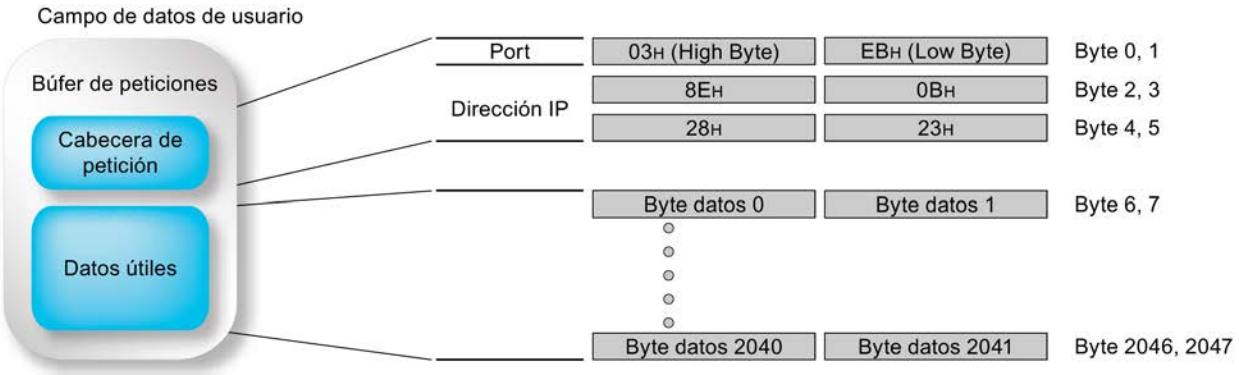


Figura 9-19 Enviar y recibir a través de una conexión UDP libre por programa

- En la figura (entradas hexadecimal) se parte, a modo de ejemplo, de la siguiente dirección IP: 142.11.40.35;
- Para la dirección de puerto 1003 se tiene que introducir, por ejemplo: para byte High: 03_H; para byte Low: EB_H.
- El área de datos de usuario puede tener hasta 2048 bytes. Pueden transmitirse hasta 2042 bytes de datos de usuario. 6 bytes están reservados para el encabezado de la petición. Tenga en cuenta que la longitud de datos indicada al llamar la instrucción (parámetro LEN) ha de contener el encabezado y los datos de usuario.

No modificar los parámetros de llamada hasta después de la confirmación de la petición

Nota

Los parámetros de llamada en la interfaz de llamada de las instrucciones AG_SEND o AG_RECV no se pueden modificar, tras la activación de la llamada, hasta después de que la instrucción haya confirmado la ejecución de la petición con DONE=1 o con ERROR=1.

Si no se tiene en cuenta esto, puede suceder que se cancele con error la ejecución de la petición.

Indicador de estado en la interfaz de llamada de la instrucción; particularidades de AG_SEND y AG_RECV (solo para S7-300) *)

En el caso de AG_SEND y AG_RECV obtendrá las indicaciones citadas a continuación en los siguientes casos operativos:

- el CP se encuentra en STOP;
- la conexión no está configurada;
- conexión no establecida;
- conexión cancelada;

Indicaciones:

- AG_SEND:
  DONE=0; ERROR=1; Status=8183_H
- AG_RECV:
  DONE=0; ERROR=0; Status=8180_H
  o
  DONE=0; ERROR=1; Status=8183_H

*) válido para instrucciones a partir de la versión 4.0

Consulte también

Indicaciones generales (Página 8046)

AG_SEND / AG_LSEND / AG_SSEND

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet)

Descripción

AG_SEND / AG_LSEND / AG_SSEND transfieren datos al CP Ethernet para su transmisión a través de una conexión configurada.

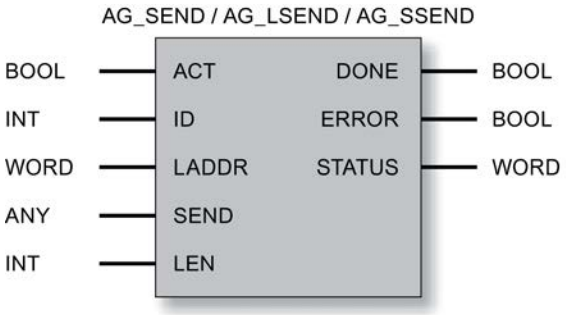
El área de datos indicada puede ser un área de marcas o un área de bloques de datos.

Si ha sido posible enviar a través de Ethernet todo el área de datos de usuario, esto se señaliza para notificar la ejecución correcta.

Observación:
Todos los datos siguientes son igualmente válidos para AG_SEND, AG_LSEND y AG_SSEND, a no ser que se indique lo contrario.

Llamada

Interfaz de llamada en representación FUP



Nota

Si desea utilizar la instrucción AG_SSEND, al configurar las conexiones tendrá que seleccionar en las propiedades de la conexión el modo "SPEED SEND/RECV".

Consulte también

Parámetros en el caso de AG_SEND / AG_LSEND / AG_SSEND (IE) (Página 8057)

Parámetros DONE, ERROR, STATUS (Página 8058)

Manual de programación "Funciones (FC) y bloques de función (FB) para CPs SIMATIC NET S7" (http://support.automation.siemens.com/ww/view/de/30564821)

Parámetros en el caso de AG_SEND / AG_LSEND / AG_SSEND (IE)

Explicación de los parámetros formales

La tabla representada a continuación explica los parámetros formales para las instrucciones AG_SEND / AG_LSEND / AG_SSEND:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ACT	INPUT	BOOL	0,1	En el caso de una llamada de la instrucción con ACT = 1 se envían LEN Bytes del área de datos indicada con el parámetro SEND. En el caso de una llamada de la instrucción con ACT = 0 se actualizan las indicaciones de estado DONE, ERROR y STATUS.
ID	INPUT	INT	1,2...64 (S7-400) 1,2...16 (S7-300)	En el parámetro ID se indica el número de la conexión.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo en la tabla de configuración. Indique aquí esta dirección.
SEND	INPUT	ANY		Indicar la dirección y la longitud La dirección del área de datos remite como alternativa a: • Área de marcas • Área de bloques de datos
LEN	INPUT	INT	en caso de ISO-Transport e ISO-on-TCP / TCP: 1,2,...8192 (o hasta "dato de longitud en parámetro SEND") en UDP: 1,2,...2048 (o hasta "dato de longitud en parámetro SEND")	Cantidad de bytes que se deben enviar junto con la petición desde el área de datos. Este número puede estar comprendido entre 1 y "dato de longitud en parámetro SEND". • Considerar el tipo de instrucciones: – para S7-300 Con las versiones actuales de AG_SEND se pueden transmitir hasta 8192 bytes (2048 bytes con UDP). – para S7-400 Con AG_SEND, el área de datos generalmente está limitada a 240 bytes, como máximo. En S7-400 considerar: • Rendimiento incrementado en caso de registros cortos: la transmisión de registros de hasta 240 bytes tiene lugar con un rendimiento incrementado. Esto es válido con independencia del tipo utilizado (AG_SEND / AG_LSEND). • Con AG_SSEND, el área de datos está limitado a como máximo 1452 bytes.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
DONE	OUTPUT	BOOL	0: Petición en curso 1: Petición ejecutada	El parámetro de estado indica si la petición se ha ejecutado sin errores. Mientras DONE = 0, no se puede impulsar ninguna otra petición. En caso de aceptarse la petición, el CP pone DONE a 0.
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Parámetros DONE, ERROR, STATUS (Página 8058)
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Parámetros DONE, ERROR, STATUS (Página 8058)

Consulte también

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet) (Página 8055)

Parámetros DONE, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-328 Códigos de condición AG_SEND / AG_LSEND / AG_SSEND

DONE	ERROR	STATUS	Significado
1	0	0000_H	Petición terminada sin errores.
0	0	0000_H	Ninguna petición en proceso.
0	0	8181_H	Petición en curso.
0	1	7000_H	Este código es solo posible en S7-400: La instrucción se ha llamado con ACT=0; pero la petición no se procesa.
0	1	8183_H	Falta la configuración o el servicio ISO/TCP en el CP Ethernet no ha sido aún arrancado.
0	1	8184_H	• Se ha indicado un tipo de datos no permitido para el parámetro SEND. • Error del sistema (el área de datos fuente tiene errores).
0	1	8185_H	Parámetro LEN mayor que el área fuente SEND.

DONE	ERROR	STATUS	Significado
0	1	8186_H	Parámetro ID no válido. • ID != 1,2....16 (S7-300) • ID != 1,2....64.(S7-400)
0	1	8302_H	No hay recursos de recepción en la estación de destino. La estación receptora no puede procesar con suficiente rapidez los datos recibidos o no ofrece recursos de recepción.
0	1	8304_H	La conexión no está establecida. La petición de emisión solo se debería emitir de nuevo tras un tiempo de espera > 100 ms.
0	1	8311_H	Estación de destino no accesible bajo la dirección Ethernet indicada.
0	1	8312_H	Error Ethernet en el CP.
0	1	$8F22_H$	Área de origen no valida, p. ej.: Área no existente en DB Parámetro LEN < 0
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Número de DB erróneo
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F44_H$	Está bloqueado el acceso a un parámetro a leer en el procesamiento de la instrucción.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible p. ej., parámetro LEN = 0 .
0	1	8090_H	• No existe módulo con esta dirección inicial de módulo; • La instrucción utilizada no es acorde con la familia de sistemas utilizada (se tienen que utilizar instrucciones diferentes para S7-300 y S7-400).
0	1	8091_H	La dirección base lógica no está en formato de palabra doble.
0	1	8092_H	En la referencia ANY se ha indicado un tipo distinto de BYTE. (solo para S7-400)
0	1	$80A4_H$	La conexión de bus K entre CPU y CP no está establecida. (en CPUs con versiones más actuales)
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	Longitud (en parámetro LEN) errónea.
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Recursos ocupados (memoria) temporalmente en la CPU.
0	1	$80C4_H$	Error de comunicación se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario
0	1	$80D2_H$	Dirección inicial del módulo errónea.

Consulte también

AG_SEND / AG_LSEND / AG_SSEND (Ind.Ethernet) (Página 8055)

AG_RECV / AG_LRECV / AG_SRECV

AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet)

Descripción

La instrucción AG_RECV / AG_LRECV / AG_SRECV importa del CP Ethernet los datos transmitidos a través de una conexión configurada.

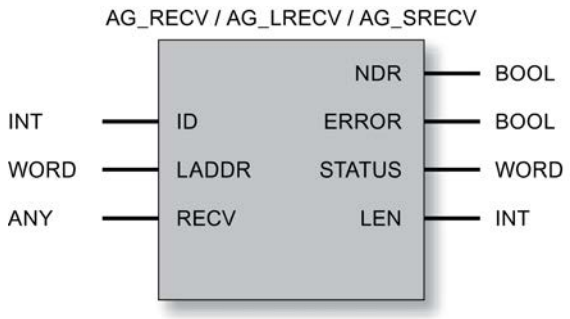
El área de datos indicada para la toma de datos puede ser un área de marcas o un área de bloques de datos.

Se señala que la función ha sido ejecutada sin errores cuando se hayan podido recibir los datos del CP Ethernet.

Observación:
Todos los datos siguientes son igualmente válidos para AG_RECV y AG_LRECV / AG_SRECV, a no ser que se indique lo contrario.

Llamada

Interfaz de llamada en representación FUP



Nota

Si desea utilizar la instrucción AG_SRECV, al configurar las conexiones tendrá que seleccionar en las propiedades de la conexión el modo "SPEED SEND/RECV".

Consulte también

Parámetros en el caso de AG_RECV / AG_LRECV / AG_SRECV (IE) (Página 8061)

Parámetros DONE, ERROR, STATUS (IE) (Página 8062)

Parámetros en el caso de AG_RECV / AG_LRECV / AG_SRECV (IE)

Explicación de los parámetros formales

La tabla representada a continuación explica los parámetros formales para las instrucciones AG_RECV / AG_LRECV / AG_SRECV:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ID	INPUT	INT	1,2...64 (S7-400) 1,2...16 (S7-300)	En el parámetro ID se indica el número de la conexión ISO-Transport.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo en la tabla de configuración. Indique aquí esta dirección.
RECV	INPUT	ANY		Indicar la dirección y la longitud La dirección del área de datos remite como alternativa a: • Área de marcas • Área de bloques de datos Para la longitud, considerar lo siguiente: La transmisión de registros de hasta 212 bytes se efectúa con mayor rendimiento si en el parámetro RECV se limita también a 212 bytes la longitud. Observe para AG_SRECV: En el caso de AG_SRECV, tiene que ajustar RECV siempre a la longitud máxima del búfer de recepción de 1452 bytes. En otro caso se pueden producir, en ciertas situaciones, los siguientes errores: NDR=0; ERROR=1; STATUS=8185_H
NDR	OUTPUT	BOOL	0: - 1: nuevos datos	El parámetro indica si se han adoptado nuevos datos. Consulte el significado en relación con los parámetros ERROR y STATUS en AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Página 8059)
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros NDR y STATUS en AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Página 8059)

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros NDR y ERROR en AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Página 8059)
LEN	OUTPUT	INT	**en ISO-Transport e ISO-on-TCP:** 1,2,...8192 **en UDP:** 1,2,...2048	Indica el número de bytes que se han tomado del CP Ethernet en el área de datos. Considerar el tipo de instrucciones: • para S7-300 Con las versiones actuales de AG_RECV se pueden transmitir hasta 8192 bytes (2048 bytes con UDP). • para S7-400 Con el AG_RECV, el área de datos generalmente está limitada a 240 bytes, como máximo. Con el AG_SRECV, el área de datos está limitada a 1452 bytes, como máximo.

Parámetros DONE, ERROR, STATUS (IE)

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por los parámetros NDR, ERROR y STATUS, que tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-329 Códigos de condición - AG_RECV / AG_LRECV / AG_SRECV

NDR	ERROR	STATUS	Significado
1	0	0000_H	Nuevos datos aceptados.
0	0	8180_H	Todavía no hay datos (no para AG_SRECV).
0	0	8181_H	Petición en curso.
0	1	8183_H	• Falta la configuración; • El servicio ISO-Transport no se ha iniciado aún en el CP Ethernet; • La conexión no está establecida.
0	1	8184_H	• Se ha indicado un tipo de datos no permitido para el parámetro RECV. • Error del sistema.
0	1	8185_H	Búfer de destino (RECV) demasiado pequeño.

NDR	ERROR	STATUS	Significado
0	1	8186_H	Parámetro ID no válido. ID != 1,2....16 (S7-300). ID != 1,2....64.(S7-400)
0	1	8304_H	La conexión no está establecida. La petición de recepción solo se debería emitir de nuevo tras un tiempo de espera >100 ms.
0	1	$8F23_H$	Área fuente no valida, p. ej.: Área no presente en DB.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos actual protegido de escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos actual protegido de escritura.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Error de número de DB
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F45_H$	La dirección del parámetro que debe escribirse está bloqueada en la pista de acceso.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	1	8090_H	• No existe ningún módulo con esta dirección inicial del módulo o CPU en STOP • La instrucción utilizada no es acorde con la familia de sistemas utilizada (se tienen que utilizar instrucciones diferentes para S7-300 y S7-400).
0	1	8091_H	La dirección base lógica no está en formato de palabra doble.
0	1	8092_H	En la referencia ANY se ha indicado un tipo distinto de BYTE. (solo para S7-400)
0	1	$80A0_H$	Acuse negativo al leer del módulo.
0	1	$80A4_H$	La conexión de bus K entre CPU y CP no está establecida.
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	Posibles causas: • El área de destino no es válida. • El área de destino es demasiado pequeña. • El área de destino para los datos de recepción no se ha calculado bien. Solución: realice otra llamada de recepción con el tamaño máximo del búfer de recepción. Esto sirve independientemente del tipo de conexión (Unicast/Multicast/ Broadcast) y de la familia de dispositivos (S7-300 / S7-400).
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Recursos ocupados (memoria) temporalmente en la CPU.
0	1	$80C4_H$	Error de comunicación se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario
0	1	$80D2_H$	Dirección inicial del módulo errónea.

Consulte también

AG_RECV / AG_LRECV / AG_SRECV (Ind.Ethernet) (Página 8059)

Instrucciones para coordinación de accesos en caso de FETCH/WRITE

Instrucciones para coordinación de accesos en caso de FETCH/WRITE

Panorámica

Para la función FETCH/WRITE están disponibles las siguientes instrucciones para la coordinación de accesos:

Instrucción	utilizable para		Significado
S7-300	S7-400		
AG_LOCK	x	x	Bloqueo del acceso externo a datos mediante FETCH/WRITE.
AG_UNLOCK	x	x	Liberación del acceso externo a datos mediante FETCH/WRITE.

Tener en cuenta la configuración

Si utiliza AG_LOCK y AG_UNLOCK, para CPs en estaciones S7-400 debe introducir las siguientes indicaciones en la configuración:

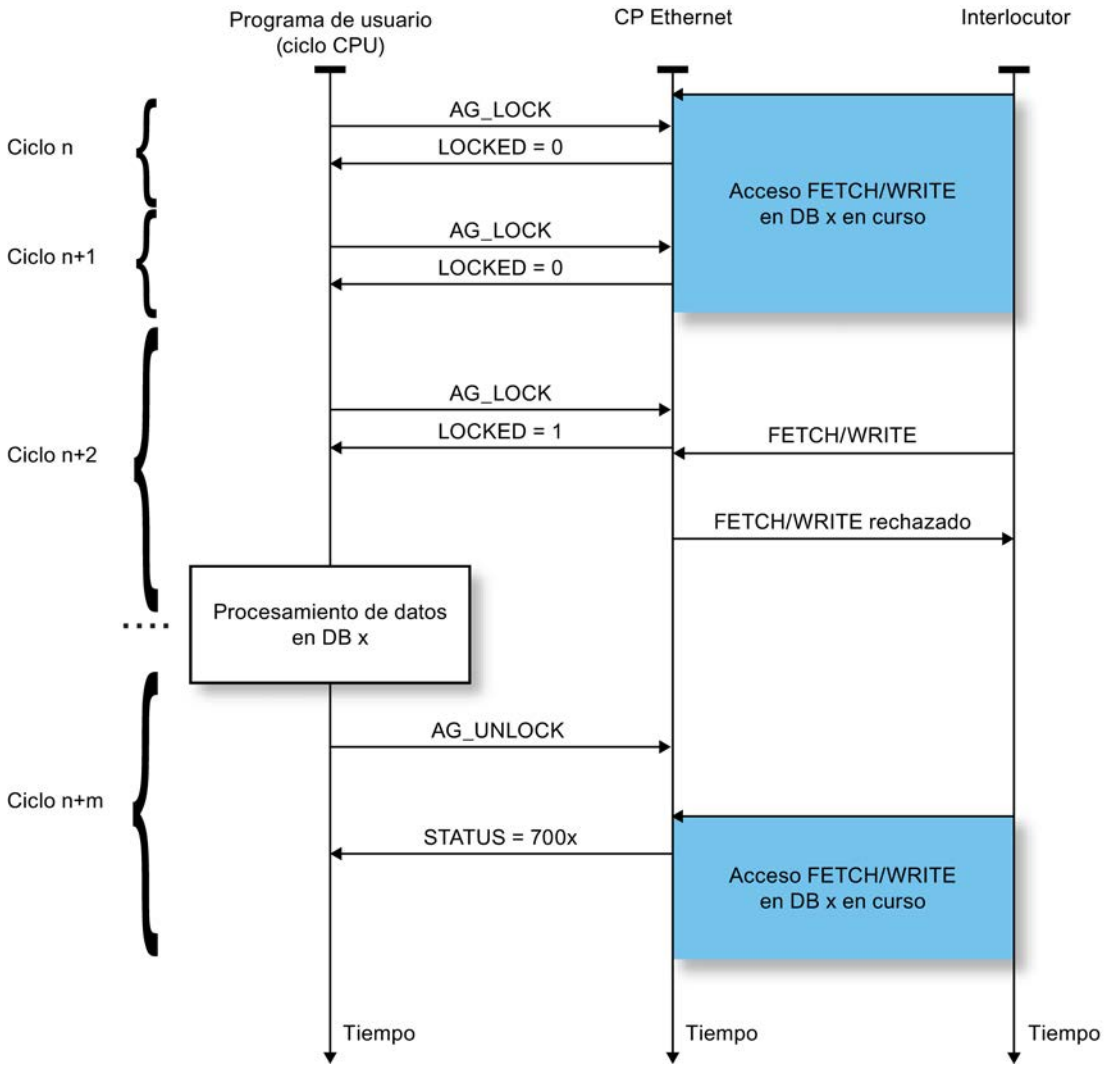
- en "Propiedades > Direcciones"
  Se tiene que seleccionar la opción "Ajuste de direcciones para LOCK/UNLOCK", si existe tal posibilidad de selección.

Forma de trabajar

Con estas instrucciones tiene la posibilidad de coordinar el acceso a áreas de memoria del sistema de manera que no se generen y transmitan datos incoherentes. El control tiene lugar en este caso desde el programa de usuario instalado en la CPU S7, que puede bloquear, si procede, por medio de la llamada de AG_LOCK un acceso externo de FETCH/WRITE. Después de un cierto tiempo o una vez concluido el acceso de escritura/lectura propio, se puede liberar de nuevo el acceso externo por medio de una petición de AG_UNLOCK.

Además se puede aprovechar el hecho de que el bloqueo de acceso sólo es válido en el enlace FETCH/WRITE indicado en la llamada. Si se configuran varios enlaces FETCH/WRITE, éstos se pueden utilizar, por ejemplo, concretamente para determinadas áreas de memoria del sistema y se puede realizar así una coordinación de accesos correspondientemente selectiva.

La siguiente representación muestra el desarrollo cronológico usual de una coordinación de accesos a la memoria en el programa de usuario, controlada por AG_LOCK y AG_UNLOCK.



La petición de bloqueo tiene que ser supervisada primero en el programa de usuario a través de lo indicado en el parámetro de retorno LOCKED. Mientras se indique LOCKED=0, se tiene que partir de que todavía está en marcha un acceso externo FETCH/WRITE.

Con LOCKED=1 se indica que el bloqueo está activado; ahora se pueden modificar datos desde el programa de usuario.

La indicación de estado se actualiza con cada llamada de la instrucción.

AG_LOCK

AG_LOCK

Descripción

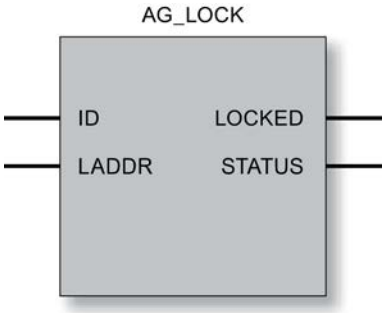
Con ayuda de la instrucción AG_LOCK se bloquea el intercambio de datos mediante FETCH o WRITE a través de la conexión seleccionada por el parámetro ID. La salida LOCKED indica

si el bloqueo ha dado resultado o no. Si el bloqueo ha sido infructuoso, la petición se tiene que impulsar de nuevo en un siguiente ciclo de CPU.

La salida STATUS indica el estado del CP para esta conexión.

Llamada

Interfaz de llamada en representación FUP



Consulte también

Parámetros en el caso de AG_LOCK (Página 8066)

Parámetro STATUS (Página 8067)

Parámetros en el caso de AG_LOCK

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para AG_LOCK:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ID	INPUT	INT	1,2...16 para S7-300 1,2...64 para S7-400	En el parámetro ID se indica el número de la conexión.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo en la tabla de configuración. Indique aquí esta dirección.
LOCKED	OUTPUT	BOOL	0: (aún) no bloqueado 1: bloqueado	Indicación del estado del bloqueo de acceso solicitado en la conexión FETCH/WRITE indicada.
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en Parámetro STATUS (Página 8067)

Consulte también

AG_LOCK (Página 8064)

Parámetro STATUS

Códigos de condición

La tabla siguiente muestra el significado de las informaciones que tienen que ser evaluadas por el programa de usuario.

Tabla 9-330 Códigos de condición de AG_LOCK

STATUS	Significado
7000_H	El CP no procesa ninguna petición
7001_H	FETCH en curso
7002_H	WRITE en curso
8183_H	FETCH/WRITE no configurado para esta conexión (solo para S7-400)
8186_H	El número de ID no está en el rango permitido (p. ej. 1...64 para CPs S7-400 Industrial Ethernet)
$80A4_H$	La conexión de bus K entre CPU y CP no está establecida (en caso de versiones de CPU más recientes).
$80B0_H$	El módulo no conoce el registro.
$80B1_H$	Longitud (en parámetro LEN) errónea.
$80B2_H$	La conexión de bus K entre CPU y CP no está establecida.
$80C0_H$	No se puede leer el registro.
$80C1_H$	El registro indicado está siendo procesado.
$80C2_H$	Hay demasiadas peticiones pendientes.
$80C3_H$	Recursos ocupados (memoria) temporalmente en la CPU.
$80C4_H$	Error de comunicación se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.
$80D2_H$	Dirección inicial del módulo errónea.

Consulte también

AG_LOCK (Página 8064)

AG_UNLOCK

AG_UNLOCK

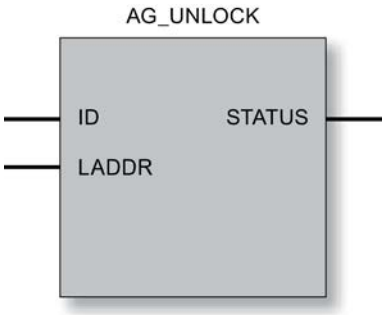
Descripción

La instrucción AG_UNLOCK permite habilitar el acceso externo a las áreas de memoria de usuario de la CPU S7. A continuación es posible acceder con FETCH o WRITE a través de la conexión seleccionada por el parámetro ID.

Esto ha estado precedido de un bloqueo de acceso por AG_LOCK.

Llamada

Interfaz de llamada en representación FUP



Forma de trabajar

Para volver a habilitar la conexión, la instrucción debe volver a desactivar el bit de petición LOCK. La instrucción indica además el estado actual con mensajes de error.

Consulte también

Parámetro STATUS (Página 8069)

Parámetros en el caso de AG_UNLOCK (Página 8068)

Parámetros en el caso de AG_UNLOCK

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para AG_UNLOCK:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ID	INPUT	INT	1,2...16 para S7-300 1,2...64 para S7-400	En el parámetro ID se indica el número de la conexión. (v. Configuración)
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo. Indique aquí esta dirección.
STATUS	OUTPUT	WORD		Indicador de estado: Consulte el significado en Parámetro STATUS (Página 8069)

Consulte también

AG_UNLOCK (Página 8066)

Parámetro STATUS

Códigos de condición

La tabla siguiente informa sobre la indicación de STATUS (código de condición) que tiene que ser evaluada por el programa de usuario.

Tabla 9-331 Códigos de condición de AG_UNLOCK

STATUS	Significado
7000_H	El CP no procesa ninguna petición
7001_H	FETCH en curso
7002_H	WRITE en curso
8183_H	FETCH/WRITE no configurado para esta conexión (solo para S7-400)
8186_H	El número de ID no está en el rango permitido (p. ej. 1...64 para CPs S7-400 Industrial Ethernet)
$80A4_H$	La conexión de bus K entre CPU y CP no está establecida (en caso de versiones de CPU más recientes).
$80B0_H$	El módulo no conoce el registro.
$80B1_H$	Longitud (en parámetro LEN) errónea.
$80B2_H$	La conexión de bus K entre CPU y CP no está establecida.
$80C0_H$	No se puede leer el registro.
$80C1_H$	El registro indicado está siendo procesado.
$80C2_H$	Hay demasiadas peticiones pendientes.
$80C3_H$	Recursos ocupados (memoria) temporalmente en la CPU.
$80C4_H$	Error de comunicación se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.
$80D2_H$	Dirección inicial del módulo errónea.

Consulte también

AG_UNLOCK (Página 8066)

Instrucciones para el diagnóstico del enlace

AG_CNTEX

AG_CNTEX

Significado y funcionamiento

Con la instrucción AG_CNTEX existe la posibilidad de diagnosticar conexiones y activar dispositivos en la red mediante el comando ping. Si es necesario se puede inicializar un nuevo establecimiento de conexión a través de AG_CNTEX.

Nota

AG_CNTRL y AG_CNTEX

El bloque de programa AG_CNTEX ofrece ampliaciones con respecto al bloque de programa AG_CNTRL.

Algunas funciones del AG_CNTRL se incluyen en el AG_CNTEX y pueden utilizarse de la misma forma en la interfaz del programa de usuario.

Las acciones siguientes se pueden ejecutar a través de comandos parametrizables:

- Lectura de informaciones sobre la conexión
  Sobre la base de informaciones de estado para todas las conexiones del CP o solo para algunas de ellas puede decidir si es conveniente resetear las conexiones.
- Reset de conexiones configuradas
  Es posible resetear conexiones concretas o todas las conexiones de un CP.
- Cancelar la conexión activa y establecerla de nuevo
- Leer tipos de conexión configurados en el CP (ampliación respecto a AG_CNTRL)
- Enviar comando PING (ampliación respecto a AG_CNTRL)
  Existe la posibilidad de comprobar si un dispositivo concreto está accesible en la red.

Los comandos del bloque de programa AG_CNTEX solo se permiten para conexiones SEND/RECV, que se basan en los protocolos ISO / RFC / TCP / UDP.

Nota

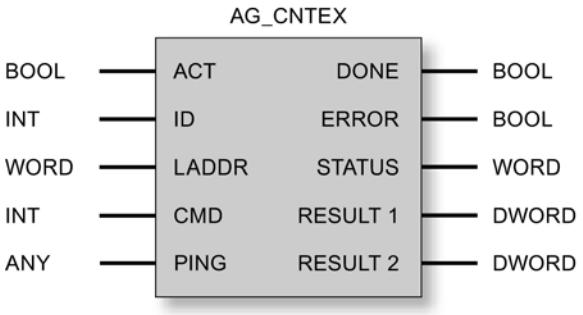
Disponibilidad en la librería del bloque

Si el bloque de programa AG_CNTEX aún no está disponible en la librería de bloques SIMATIC_NET_CP, instale la librería de bloques SIMATIC NET actual, que se encuentra en Internet bajo la siguiente ID de artículo:

8797900 (http://support.automation.siemens.com/WW/view/de/8797900)

Interfaz de llamada

Interfaz de llamada en representación FUP



Consulte también

Funcionamiento de la función PING (Página 8071)

Parámetros con AG_CNTEX (Página 8072)

Parámetros DONE, ERROR, STATUS (Página 8074)

Comandos y resultados de las peticiones - AG_CNTEX (Página 8075)

Funcionamiento de la función PING

Funcionamiento / secuencia de llamada

Con el comando PING CMD=8 se consigue que el CP deposite 4 peticiones PING consecutivas de la red en la dirección IP indicada en la petición. El eco PING es esperado por el CP dentro del intervalo especificado en el bloque de peticiones PING.

El CP registra los tiempos de respuesta y los deposita en los parámetros RESULT 1/2.

Los dos parámetros RESULT 1/2 se consultan mediante el comando PING CMD=9. En cuanto se han respondido las 4 peticiones PING o bien se ha excedido su tiempo de vigilancia especificado, la ejecución se confirma en el parámetro DONE=1. El resultado PING puede consultarse en un tiempo máximo de 30 segundos; después, las entradas de RESULT ya no son válidas.

Nota

PING requiere una conexión configurada

El comando PING requiere que como mínimo esté configurada **una** conexión para la interfaz SEND/RECEIVE (TCP, ISO-on-TCP, ISO-Transport, UDP)

Varias peticiones PING al mismo tiempo

Es posible depositar un máximo de 4 peticiones PING simultáneas en diferentes direcciones IP. Para ello se debe utilizar el mismo DB instancia en cada caso para las peticiones PING.

No es posible depositar más peticiones PING hasta que se haya procesado al menos una de las que están en curso.

Si se depositan demasiadas peticiones PING al mismo tiempo, se muestra un mensaje de error (parámetro STATUS = $828A_H$).

¿Cuándo se han procesado las peticiones PING?

Las peticiones PING se consideran procesadas cuando se cumple una de las condiciones siguientes:

- El resultado PING se ha leído;
- El resultado PING no se ha leído pero han transcurrido 30 segundos desde que el resultado PING está disponible.

Consulte también

AG_CNTEX (Página 8069)

Comandos y resultados de las peticiones - AG_CNTEX (Página 8075)

Parámetros con AG_CNTEX

Explicación de los parámetros formales

La tabla siguiente explica todos los parámetros formales de la función AG_CNTEX:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ACT	INPUT	BOOL	0, 1	El FB se tiene que llamar con ACT=1. Al llamar con ACT=0 no se produce llamada de la función y el bloque se abandona de inmediato.
ID	INPUT	INT	• 1, 2, .., n, o bien • 0	En el parámetro ID se indica el número de la conexión. El número de conexión se tiene que tomar de la configuración. n es el número máximo de conexiones y depende del producto (S7-300 o S7-400). En una llamada que redirecciona todas las conexiones se debe indicar la ID 0. Esto se refiere a lo siguiente: • las funciones CN_STATUS_ALL (CMD 3) y CN_RESET_ALL (CMD 4) • comando PING con CMD 8 o CMD 9
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo. Indique aquí esta dirección.
CMD	INPUT	INT		Comando a FB AG_CNTEX

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
PING	INPUT	ANY		Hace referencia al bloque de datos (p. ej. DB) que contiene la estructura de datos para el comando PING. El bloque de datos contiene la dirección IP así como, opcionalmente, los datos sobre la vigilancia de tiempo y sobre el número de bytes que deben transferirse en la petición PING. Para la estructura de datos véase más abajo
DONE	OUTPUT	BOOL	0: la petición está en procesamiento o aún no se ha lanzado 1: petición ejecutada	El parámetro de estado indica si la petición se ha desarrollado correctamente. Consulte el significado en relación con los parámetros ERROR y STATUS en Parámetros DONE, ERROR, STATUS (Página 8074) Nota: con DONE=1 se puede evaluar RESULT
ERROR	OUTPUT	BOOL	0: ningún error 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Parámetros DONE, ERROR, STATUS (Página 8074)
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Parámetros DONE, ERROR, STATUS (Página 8074)
RESULT1	OUTPUT	DWORD		Respuesta a AG_CNTEX según comando.
RESULT2	OUTPUT	DWORD		Respuesta de la parte 2 a AG_CNTEX según comando.

Bloque de datos PING

El parámetro PING hace referencia a un bloque de datos con la siguiente estructura:

Parámetro	Tipo de datos	Rango de valores	Nota
Dirección IP	ARRAY [1..4] of Byte		
TIMEOUT	INT	1 - 60000 ms	Indicación opcional; valor predeterminado = 1000 ms
Size	INT	1 - 1000 bytes	Indicación opcional; valor predeterminado = 32 bytes

Consulte también

AG_CNTEX (Página 8069)

Parámetros DONE, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Adicionalmente se tienen que evaluar los resultados de comandos en los parámetros RESULT1/2 según Comandos y resultados de las peticiones - AG_CNTEX (Página 8075).

Tabla 9-332 Códigos de condición AG_CNTEX

DONE	ERROR	STATUS	Significado
1	0	0000_H	Se ha transmitido con éxito una petición (CMD) al CP (p. ej. RESET) o se ha leído con éxito un estado del CP. Se han podido evaluar los parámetros RESULT1/2.
0	0	0000_H	No se ha producido aún ninguna llamada de bloque o el bloque de programa se llama con ACT=0.
0	0	8181_H	Petición en curso La llamada del bloque se tiene que repetir con los mismos parámetros, hasta que se notifique DONE o ERROR.
0	1	8183_H	Falta la configuración o no se ha iniciado aún el servicio en el CP Ethernet.
0	1	8184_H	Error de sistema o tipo de parámetro erróneo. La causa puede ser: • El tipo de datos del puntero ANY para el parámetro PING no es correcto. • El puntero ANY hace referencia a una dirección de bit impar.
0	1	8186_H	El parámetro ID no es válido. La ID admisible depende del comando seleccionado.
0	1	8187_H	El parámetro CMD no es válido.
0	1	8090_H	Significados posibles: • No existe un módulo con esta dirección inicial de módulo; • El bloque de programa utilizado no es acorde con la familia de sistemas utilizada (se tienen que utilizar bloques de programa diferentes para S7-300 y S7-400); • Este módulo no da soporte a esta función.
0	1	8091_H	La dirección inicial del módulo no está en la trama de doble palabra.
0	1	8092_H	La dirección inicial del módulo es incorrecta.
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida. En el sistema H, la CPU correspondiente se encuentra en el estado operativo STOP.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Recursos ocupados (memoria) temporalmente en la CPU.
0	1	$80C4_H$	Error de comunicación El error se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.
0	1	8286_H	El valor de la indicación "Timeout" en el bloque de datos PING está fuera del rango de valores válido.
0	1	8287_H	La dirección IP indicada en el DB PING está reservada y, por tanto, no es admisible.

DONE	ERROR	STATUS	Significado
0	1	8288_H	La indicación solo aparece con el comando de petición de resultado PING. Significados posibles: • La dirección IP no ha sido alcanzada por el comando PING (por ejemplo porque el CP se ha puesto en modo STOP después de que se haya depositado el comando de petición PING); • El resultado PING ya se ha leído; • El resultado PING no se ha leído dentro del intervalo máximo de 30 segundos.
0	1	8289_H	El volumen de datos de la petición PING ha excedido el rango admisible (máximo 1000 bytes; consulte la estructura de datos del comando PING)
0	1	$828A_H$	Ya se están procesando 4 peticiones PING. No se admiten peticiones nuevas hasta que se hayan procesado las existentes.
0	1	$828B_H$	Ya se está procesando una petición PING para la dirección IP indicada. Utilice el comando de petición de resultado PING para concluir el procesamiento en curso.

Consulte también

AG_CNTEX (Página 8069)

Comandos y resultados de las peticiones - AG_CNTEX

Comandos y evaluación de los resultados de la petición

Vea en las tablas siguientes los comandos posibles y los resultados evaluables en los parámetros RESULT1/2.

Nota

Evaluación de comandos para tipos de CP o versiones de firmware menos recientes

Los comandos descritos a continuación son compatibles con tipos de CP o versiones de firmware actuales. Observe las indicaciones adicionales que aparecen bajo la siguiente referencia en Internet:

http://support.automation.siemens.com/WW/view/de/33414377 (http://support.automation.siemens.com/ww/view/es/33414377)

Tabla 9-333 Comandos a FC AG_CNTRL.

CMD	Significado		
0	NOP – no operation Se pasa por el bloque sin petición al CP.		
	RESULT (para CMD = 0)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	Secuencia sin error
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado			
1	CN_STATUS – connection status Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	
	RESULT1	0000 000*$_H$	Bits 0-3: códigos de condición para el sentido de emisión (valores excluidos: 0x2)	
			Bit 0	Tipo de conexión • 0: no hay conexión de transmisión ni recepción • 1: la conexión está reservada para peticiones de transmisión y recepción
			Bit 1	Estado de la petición actual • 0: Ninguna petición de emisión en proceso • 1: Petición de emisión en proceso
			Bits 2+3	Petición precedente: • 00: Ninguna información disponible sobre la petición de emisión precedente • 01: Petición de emisión precedente concluida positivamente • 10: Petición de emisión precedente concluida negativamente

CMD	Significado			
1	CN_STATUS – connection status **(continuación para CMD=1)** Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	
	RESULT1	0000 00*0$_H$	Bits 4−7: códigos de condición para el sentido de recepción (valores excluidos: 0x2)	
			Bit 4	Tipo de conexión • 0: no hay conexión de transmisión ni recepción • 1: la conexión está reservada para peticiones de transmisión y recepción

CMD	Significado			
			Bit 5	Estado de la petición actual • 0: Ninguna petición de recepción en proceso • 1: Petición de recepción en proceso
			Bits 6+7	Petición precedente: • 00: Ninguna información disponible sobre la petición de recepción precedente • 01: Petición de recepción precedente concluida positivamente • 10: Petición de recepción precedente concluida negativamente

CMD	Significado			
1	CN_STATUS – connection status **(continuación para CMD=1)** Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	
	RESULT1	0000 0*00_H	Bits 8-11: Códigos de condición para FETCH/WRITE (valores excluidos:0x3,0x7,0x8,0xB,0xF)	
			Bit 8	Tipo de conexión: • 0: ninguna conexión FETCH • 1: conexión reservada para peticiones FETCH
			Bit 9	Tipo de conexión: • 0: ninguna conexión WRITE • 1: conexión reservada para peticiones WRITE
			Bit 10	Estado de la petición (FETCH/WRITE): • 0: Estado de la petición OK • 1: Estado de la petición NOT OK este identificador se pone en los siguientes casos: – La petición ha sido acusada negativamente por la CPU – La petición no se ha podido transmitir a la CPU por estar la conexión en el estado "LOCKED". – La petición ha sido rechazada porque el encabezado FETCH/WRITE no tenía la estructura correcta.
			Bit 11	Estado de petición FETCH/WRITE • 0: ninguna petición en curso • 1: en curso una petición de LAN

CMD	Significado			
1	CN_STATUS – connection status **(continuación para CMD=1)** Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	
	RESULT1	0000 *000$_H$	Bits 12-15: Informaciones generales del CP (valores excluidos: 0x3,0xB)	
			Bit 12 + 13	Información sobre el estado de la conexión: (disponible solo para conexiones SEND/RECV que utilicen los protocolos ISO/RFC/TCP; en caso de UDP se emiten las informaciones internas correspondientes) • 00: conexión deshecha • 01: estableciéndose la conexión • 10: deshaciéndose la conexión • 11: conexión establecida
			Bit 14	Información del CP: • 0: CP en STOP • 1: CP en RUN
			Bit 15	Identificador de reset • 0: no se ha ejecutado aún ningún reset de la conexión a través de AG_CNTEX o se ha retirado la identificación de reset. • 1: se ha realizado un reset de la conexión a través de AG_CNTEX.
	RESULT1	**** 0000$_H$		Bits 16-31: Reservado 0 – reservado para futuras ampliaciones
	RESULT2	0000 0000$_H$		– reservado para futuras extensiones

CMD	Significado		
2	CN_RESET – connection reset Este comando resetea la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. El reset de la conexión provoca una cancelación de la misma y un nuevo establecimiento (activo o pasivo, según la configuración). Se borran los datos recibidos en el momento de la cancelación de la conexión, pero todavía no adoptados en el programa de usuario. Se genera adicionalmente una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 2)		**Significado**
	Parámetro	**Valor Hex/rango**	
	RESULT1	0000 0001$_H$	La petición de Reset se ha transmitido con éxito al CP. Se han impulsado la cancelación de la conexión y su subsiguiente establecimiento.

CMD	Significado		
		0000 0002_H	La petición de Reset no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	0000 0000_H	Valor predeterminado

CMD	Significado		
3	CN_STATUS_ALL – all connections status Este comando proporciona en los parámetros RESULT1/2 (en total 8 bytes de información colectiva) el estado de conexión de todas las conexiones (establecidas/deshechas). El parámetro ID tiene que estar puesto a "0" (se verifica en cuanto a 0). El CP se ha seleccionado a través del parámetro LADDR. Si se necesitan, se pueden obtener informaciones detalladas sobre una conexión deshecha o no configurada a través de una nueva llamada de estado, dirigida a la conexión, con CMD=1.		
	RESULT (para CMD = 3)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	**** $****_H$ 32 bits con la siguiente validez: • Para S7-400: Bits 0-31 para conexiones 1 - 32 • Para S7-300: Bits 0-15 para conexiones 1 - 16	Para cada conexión: • 0 – conexión deshecha / no configurada • 1 – conexión establecida
	RESULT2	**** $****_H$ 32 bits con la siguiente validez: • Para S7-400: Bits 0-31 para conexiones 33 - 64	Para cada conexión: • 0 – conexión deshecha / no configurada • 1 – conexión establecida

CMD	Significado		
4	CN_RESET_ALL – all connections reset Este comando resetea todas las conexiones. El parámetro ID tiene que estar puesto a "0" (se verifica en cuanto a 0). El CP se ha seleccionado a través del parámetro LADDR. El reset de las conexiones provoca una cancelación de las mismas y un nuevo establecimiento (activo o pasivo, según la configuración). Se borran los datos recibidos en el momento de la cancelación de la conexión, pero todavía no adoptados en el programa de usuario. Se genera adicionalmente una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 4)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	0000 0001_H	La petición de Reset se ha transmitido con éxito al CP. Se han impulsado la cancelación y el subsiguiente establecimiento de todas las conexiones.
	RESULT1	0000 0002_H	La petición de Reset no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	0000 0000_H	Valor predeterminado

CMD	Significado		
5	CN_CLEAR_RESET – inicialización de la identificación de Reset Este comando resetea el identificador de reset (bit 15 en RESULT1) para la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Esta petición se ejecuta también automáticamente al leer el estado de la conexión (CMD=1); la petición enviable por separado aquí descrita solo se necesita por lo tanto en casos especiales.		
	RESULT (para CMD = 5)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición de Clear se ha transmitido con éxito al CP.
	RESULT1	$0000\ 0002_H$	La petición de Clear no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado		
6	CN_DISCON − connection disconnect Este comando resetea la conexión que se había seleccionado con ID y LADDR. El reset de la conexión se realiza con una cancelación de la misma. Los datos guardados eventualmente en la pila se pierden sin advertencia. Después de esto no se produce ningún establecimiento automático de la conexión. La conexión se puede restablecer con la petición de Control CN_STARTCON. Se crea una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 6)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición se ha transmitido con éxito al CP. Se ha iniciado la cancelación de la conexión.
	RESULT1	$0000\ 0002_H$	La petición no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado		
7	CN_STARTCON − start connection Este comando establece una conexión seleccionada con ID y LADDR y que se canceló antes con la petición de Control CN_DISCON. Se crea una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 7)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición de establecimiento de la conexión se ha transmitido con éxito al CP. Se ha iniciado el establecimiento de la conexión.
	RESULT1	$0000\ 0002_H$	La petición de establecimiento de la conexión no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado		
8	PING_REQUEST - Enviar petición PING Este comando envía una petición PING al CP. Seguidamente, el CP ordena 4 peticiones eco PING para la dirección IP indicada.		
	RESULT (para CMD = 8)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición PING se ha enviado correctamente al CP.
	RESULT2	$0000\ 0002_H$	No ha sido posible enviar correctamente la petición PING al CP porque el servicio correspondiente no estaba accesible en el CP. La causa posible es, por ejemplo: CP en modo STOP.

CMD	Significado		
9	PING_RESULT - Consultar resultado PING Este comando envía una consulta de resultado PING al CP. El CP transfiere los resultados de las 4 peticiones eco PING realizadas en el parámetro RESULT. La llamada no se considera correcta hasta que el CP ha concluido las 4 peticiones eco PING.		
	RESULT (para CMD = 9)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$****\ ****_H$	1. palabra: tiempo de respuesta en ms a la 1.ª petición eco PING. 2. palabra: tiempo de respuesta en ms a la 2.ª petición eco PING. Ejemplo: $0005\ FFFF_H$ eco 1 -> recibida tras 5 ms eco 2 -> sin eco en el tiempo de vigilancia indicado,
	RESULT2	$****\ ****_H$	1. palabra: tiempo de respuesta en ms a la 3.ª petición eco PING. 2. palabra: tiempo de respuesta en ms a la 4.ª petición eco PING. Ejemplo: $0002\ 3456_H$ eco 3 -> recibida tras 2 ms eco 4 -> recibida tras 13398 ms
	Rango de valores para palabras de datos en RESULT1 / RESULT 2:		
		0000_H	no utilizado
		0001_H ... $EA60_H$	Tiempo de respuesta en ms 0001_H = 1 ms $EA60_H$ = 60000 ms
		$EA61_H$ ... $FFFE_H$	no utilizado
		$FFFF_H$	Timeout: sin eco dentro del tiempo de vigilancia especificado.

CMD	Significado		
10	CONN_TYPE - Tipo de conexión Este comando induce al CP a indicar el tipo de conexión actual para la ID de conexión indicada.		
	RESULT (para CMD = 10)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	0000 000*$_H$	La petición proporciona los valores siguientes para los posibles tipos de conexión: 0: no se ha creado ninguna conexión 1: conexión UDP 2: conexión SMTP 3: conexión TCP 4: conexión UDP libre 5: conexión FTP 6: conexión ISO-Transport 7: conexión ISO-on-TCP
	RESULT2	0000 0000$_H$	Valor predeterminado

Consulte también

Observe las indicaciones adicionales que aparecen bajo la siguiente ID de artículo en las preguntas frecuentes (FAQ): http://support.automation.siemens.com/WW/view/de/33414377 (http://support.automation.siemens.com/ww/view/es/33414377)

Consulte también

AG_CNTEX (Página 8069)

Funcionamiento de la función PING (Página 8070)

AG_CNTRL para el diagnóstico del enlace

AG_CNTRL

Descripción

Con la instrucción AG_CNTRL se tiene la posibilidad de diagnosticar conexiones. Si fuera necesario se puede inicializar un nuevo establecimiento de la conexión a través de la instrucción.

Nota

AG_CNTRL y AG_CNTEX

La instrucción AG_CNTEX ofrece ampliaciones con respecto a la instrucción AG_CNTRL.

Todas las funciones del AG_CNTRL se incluyen en el AG_CNTEX y pueden utilizarse de la misma forma en la interfaz del programa de usuario.

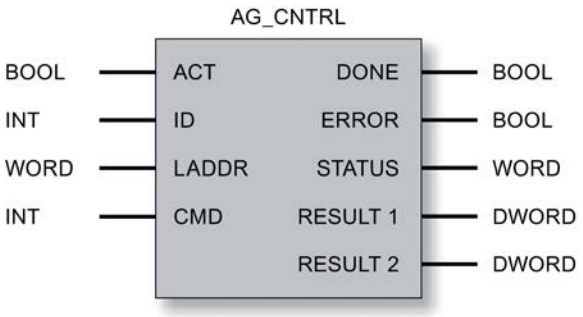
Las acciones siguientes se pueden ejecutar a través de comandos parametrizables:

- Lectura de informaciones sobre la conexión
  Sobre la base de informaciones de estado para todas las conexiones del CP o solo para algunas de ellas puede decidir si es conveniente resetear las conexiones.
- Reset de conexiones configuradas
  Es posible resetear conexiones concretas o todas las conexiones de un CP.
- Cancelar la conexión activa y establecerla de nuevo

Los comandos de la instrucción AG_CNTRL solo se permiten para conexiones SEND/RECV, que se basan en los protocolos ISO / RFC / TCP / UDP.

Interfaz de llamada

Interfaz de llamada en representación FUP



Informaciones adicionales

FAQ con ID de artículo 33414377 (http://support.automation.siemens.com/WW/view/de/33414377)

Consulte también

Parámetros en el caso de AG_CNTRL (Página 8084)

Parámetros DONE, ERROR, STATUS (Página 8085)

Comandos y resultados de las peticiones AG_CNTRL (Página 8086)

Parámetros en el caso de AG_CNTRL

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para AG_CNTRL:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ACT	INPUT	BOOL	0, 1	AG_UNLOCK se tiene que llamar con ACT=1. Al llamar con ACT=0 no se produce la llamada de la función y la instrucción se abandona de inmediato.
ID	INPUT	INT	• 1, 2, .., n, o bien • 0	En el parámetro ID se indica el número de la conexión. El número de conexión se tiene que tomar de la configuración. n es el número máximo de conexiones y depende del producto (S7-300 o S7-400). En caso de una llamada dirigida a todas las conexiones, (función _ALL con CMD 3 o 4) se tiene que indicar ID 0.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo en la tabla de configuración. Indique aquí esta dirección.
CMD	INPUT	INT		Comando a AG_CNTRL.
DONE	OUTPUT	BOOL	0: la petición está en procesamiento o aún no se ha lanzado 1: petición ejecutada	El parámetro de estado indica si la petición se ha desarrollado correctamente. Consulte el significado en relación con los parámetros ERROR y STATUS en Parámetros DONE, ERROR, STATUS (Página 8085) Nota: con DONE=1 se puede evaluar RESULT
ERROR	OUTPUT	BOOL	0: ningún error 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Parámetros DONE, ERROR, STATUS (Página 8085)
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Parámetros DONE, ERROR, STATUS (Página 8085)
RESULT1	OUTPUT	DWORD		Respuesta según comando a AG_CNTRL.
RESULT2	OUTPUT	DWORD		evaluar solo para S7-400: Respuesta de la parte 2 según comando a AG_CNTRL.

Consulte también

AG_CNTRL (Página 8081)

Parámetros DONE, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Adicionalmente se tienen que evaluar los resultados de comandos en los parámetros RESULT1/2 según "Comandos y resultados de las peticiones AG_CNTRL (Página 8086)".

Tabla 9-334 Códigos de condición de AG_CNTRL

DONE	ERROR	STATUS	Significado
1	0	0000_H	Se ha transmitido con éxito una petición (CMD) al CP (p. ej. RESET) o se ha leído con éxito un estado del CP. Se han podido evaluar los parámetros RESULT1/2.
0	0	0000_H	No se ha producido aún ninguna llamada de la instrucción o esta se ha llamado con ACT=0.
0	0	8181_H	Petición en curso La llamada de la instrucción se tiene que repetir con los mismos parámetros, hasta que se notifique DONE o ERROR.
0	1	8183_H	Falta la configuración o no se ha iniciado aún el servicio en el CP Ethernet.
0	1	8186_H	El parámetro ID no es válido. La ID admisible depende del comando seleccionado; véase Parámetro CMD en Comandos y resultados de las peticiones AG_CNTRL (Página 8086).
0	1	8187_H	El parámetro CMD no es válido.
0	1	8188_H	Error de secuencia en el control de ACT (observación: esta indicación no se presenta en la versión de producto del CP / del firmware).
0	1	8189_H	La versión de CP / firmware utilizada no da soporte a AG_CNTRL. El código de condición se pone en caso de llamada en un CP3431-EX20 con firmware a partir de V1.3.9; en otros tipos de CP, en lugar de esto se envía el código de condición 80B0H. Nota: el AG_CNTRL en la versión V1.0 es soportado por CPs a partir de CP 343-1 EX21/ GX21; en estos no aparece este código de condición.
0	1	8090_H	• No existe un módulo con esta dirección inicial de módulo. o bien • La instrucción utilizada no es acorde con la familia de sistemas utilizada (se tienen que utilizar instrucciones diferentes para S7-300 y S7-400). o bien • Este módulo no da soporte a esta función.
0	1	8091_H	La dirección inicial del módulo no está en la trama de doble palabra.
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida. En el sistema H, la CPU correspondiente se encuentra en el estado operativo STOP.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Recursos ocupados (memoria) temporalmente en la CPU.

DONE	ERROR	STATUS	Significado
0	1	$80C4_H$	Error de comunicación El error se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.
0	1	$80D2_H$	La dirección inicial del módulo es incorrecta.

Consulte también

AG_CNTRL (Página 8081)

Comandos y resultados de las peticiones AG_CNTRL

Comandos y evaluación de los resultados de la petición

Vea en las tablas siguientes los comandos posibles y los resultados evaluables en los parámetros RESULT1/2.

Tabla 9-335 Comandos a AG_CNTRL

CMD	Significado		
0	NOP – no operation Se pasa por AG_CNTRL sin petición al CP.		
	RESULT (para CMD = 0)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	Secuencia sin error
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado			
1	CN_STATUS – connection status Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			Valor / Significado
	Parámetro	Valor Hex/rango	Bit	
	RESULT1	$0000\ 000*_H$	Bits 0-3: códigos de condición para el sentido de emisión (valores excluidos: 0x2)	
			Bit 0	Tipo de conexión • 0: no hay conexión de transmisión ni recepción • 1: la conexión está reservada para peticiones de transmisión y recepción

CMD	Significado			
			Bit 1	Estado de la petición actual • 0: Ninguna petición de emisión en proceso • 1: Petición de emisión en proceso
			Bits 2+3	Petición precedente: • 00: Ninguna información disponible sobre la petición de emisión precedente • 01: Petición de emisión precedente concluida positivamente • 10: Petición de emisión precedente concluida negativamente

CMD	Significado			
1	CN_STATUS – connection status **(continuación para CMD=1)** Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	
	RESULT1	0000 00*0_H	Bits 4−7: códigos de condición para el sentido de recepción (valores excluidos: 0x2)	
			Bit 4	Tipo de conexión • 0: no hay conexión de transmisión ni recepción • 1: la conexión está reservada para peticiones de transmisión y recepción
			Bit 5	Estado de la petición actual • 0: Ninguna petición de recepción en proceso • 1: Petición de recepción en proceso
			Bits 6+7	Petición precedente: • 00: Ninguna información disponible sobre la petición de recepción precedente • 01: Petición de recepción precedente concluida positivamente • 10: Petición de recepción precedente concluida negativamente

CMD	Significado			
1	CN_STATUS – connection status **(continuación para CMD=1)** Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	

CMD	Significado			
	RESULT1	0000 0*00$_H$	Bits 8-11: Códigos de condición para FETCH/WRITE (valores excluidos:0x3,0x7,0x8,0xB,0xF)	
			Bit 8	Tipo de conexión: • 0: ninguna conexión FETCH • 1: conexión reservada para peticiones FETCH
			Bit 9	Tipo de conexión: • 0: ninguna conexión WRITE • 1: conexión reservada para peticiones WRITE
			Bit 10	Estado de la petición (FETCH/WRITE): • 0: Estado de la petición OK • 1: Estado de la petición NOT OK este identificador se pone en los siguientes casos: – La petición ha sido acusada negativamente por la CPU – La petición no se ha podido transmitir a la CPU por estar la conexión en el estado "LOCKED". – La petición ha sido rechazada porque el encabezado FETCH/WRITE no tenía la estructura correcta.
			Bit 11	Estado de petición FETCH/WRITE • 0: ninguna petición en curso • 1: en curso una petición de LAN

CMD	Significado			
1	CN_STATUS – connection status **(continuación para CMD=1)** Este comando proporciona el estado de la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Si estuviera activado el bit 15 (identificador de reset), se desactiva automáticamente (este comportamiento concuerda con la petición CN_CLEAR_RESET - véase CMD = 5).			
	RESULT (para CMD = 1)			**Valor / Significado**
	Parámetro	**Valor Hex/rango**	**Bit**	
	RESULT1	0000 *000$_H$	Bits 12-15: Informaciones generales del CP (valores excluidos: 0x3,0xB)	
			Bit 12 + 13	Información sobre el estado de la conexión: (disponible solo para conexiones SEND/RECV que utilicen los protocolos ISO/RFC/TCP; en caso de UDP se emiten las informaciones internas correspondientes) • 00: conexión deshecha • 01: estableciéndose la conexión • 10: deshaciéndose la conexión • 11: conexión establecida
			Bit 14	Información del CP: • 0: CP en STOP • 1: CP en RUN

CMD	Significado			
			Bit 15	Identificador de reset • 0: No se ha ejecutado aún ningún reset de la conexión a través de AG_CNTRL o se ha retirado la identificación de reset. • 1: Se ha realizado un reset de la conexión mediante la instrucción de control.
	RESULT1	**** 0000_H		Bits 16-31: Reservado 0 – reservado para futuras ampliaciones
	RESULT2	0000 0000_H		– reservado para futuras extensiones

CMD	Significado		
2	CN_RESET – connection reset Este comando resetea la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. El reset de la conexión provoca una cancelación de la misma y un nuevo establecimiento (activo o pasivo, según la configuración). Se borran los datos recibidos en el momento de la cancelación de la conexión, pero todavía no adoptados en el programa de usuario. Se genera adicionalmente una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 2)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	0000 0001_H	La petición de Reset se ha transmitido con éxito al CP. Se han impulsado la cancelación de la conexión y su subsiguiente establecimiento.
		0000 0002_H	La petición de Reset no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	0000 0000_H	Valor predeterminado

CMD	Significado		
3	CN_STATUS_ALL – all connections status Este comando proporciona en los parámetros RESULT1/2 (en total 8 bytes de información colectiva) el estado de conexión de todas las conexiones (establecidas/deshechas). El parámetro ID tiene que estar puesto a "0" (se verifica en cuanto a 0). El CP se ha seleccionado a través del parámetro LADDR. Si se necesitan, se pueden obtener informaciones detalladas sobre una conexión deshecha o no configurada a través de una nueva llamada de estado, dirigida a la conexión, con CMD=1.		
	RESULT (para CMD = 3)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	**** $****_H$	32 Bit: conexión 1 - 32 • 0 – conexión deshecha / no configurada • 1 – conexión establecida
	RESULT2	**** $****_H$	32 Bit: conexión 33 - 64 • 0 – conexión deshecha / no configurada • 1 – conexión establecida

CMD	Significado		
4	CN_RESET_ALL – all connections reset: Este comando resetea todas las conexiones. El parámetro ID tiene que estar puesto a "0" (se verifica en cuanto a 0). El CP se ha seleccionado a través del parámetro LADDR. El reset de las conexiones provoca una cancelación de las mismas y un nuevo establecimiento (activo o pasivo, según la configuración). Se borran los datos recibidos en el momento de la cancelación de la conexión, pero todavía no adoptados en el programa de usuario. Se genera adicionalmente una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 4)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición de Reset se ha transmitido con éxito al CP. Se han impulsado la cancelación y el subsiguiente establecimiento de todas las conexiones.
	RESULT1	$0000\ 0002_H$	La petición de Reset no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado		
5	CN_CLEAR_RESET – inicialización de la identificación de Reset Este comando resetea el identificador de reset (bit 15 en RESULT1) para la conexión seleccionada con la ID. El CP se ha seleccionado a través del parámetro LADDR. Esta petición se ejecuta también automáticamente al leer el estado de la conexión (CMD=1); la petición enviable por separado aquí descrita solo se necesita por lo tanto en casos especiales.		
	RESULT (para CMD = 5)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición de Clear se ha transmitido con éxito al CP.
	RESULT1	$0000\ 0002_H$	La petición de Clear no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado		
6	CN_DISCON − connection disconnect Este comando resetea la conexión que se había seleccionado con ID y LADDR. El reset de la conexión se realiza con una cancelación de la misma. Los datos guardados eventualmente en la pila se pierden sin advertencia. Después de esto no se produce ningún establecimiento automático de la conexión. La conexión se puede restablecer con la petición de control CN_STARTCON. Se crea una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 6)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición se ha transmitido con éxito al CP. Se ha iniciado la cancelación de la conexión.
	RESULT1	$0000\ 0002_H$	La petición no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

CMD	Significado		
7	CN_STARTCON − start connection Este comando establece una conexión seleccionada con ID y LADDR y que se canceló antes con la petición de control CN_DISCON. Se crea una entrada en el búfer de diagnóstico, en la que se puede ver el resultado de la petición.		
	RESULT (para CMD = 6)		Significado
	Parámetro	Valor Hex/rango	
	RESULT1	$0000\ 0001_H$	La petición de establecimiento de la conexión se ha transmitido con éxito al CP. Se ha iniciado el establecimiento de la conexión.
	RESULT1	$0000\ 0002_H$	La petición de establecimiento de la conexión no se ha podido transmitir al CP por no estar iniciado el servicio en el CP (p. ej. CP en STOP).
	RESULT2	$0000\ 0000_H$	Valor predeterminado

Informaciones adicionales

Observe las indicaciones adicionales que aparecen bajo la siguiente ID de artículo en las preguntas frecuentes (FAQ): 33414377 (http://support.automation.siemens.com/WW/view/es/33414377)

Consulte también

AG_CNTRL (Página 8081)

Instrucciones para PROFINET IO

Instrucciones para PROFINET IO

Panorámica

Para la transmisión cíclica de datos en la interfaz PROFINET IO están disponibles las instrucciones citadas a continuación. El significado de las instrucciones difiere dependiendo

de que el CP se utilice como PROFINET IO-Controller o como PROFINET IO-Device en un equipo S7.

Instrucción	utilizable para		Significado
S7-300	S7-400		
PNIO_SEND	x	-	Dependiendo del modo de funcionamiento del CP: • En el PROFINET IO-Controller Recibir datos de salida de procesos de los PROFINET IO-Devices. • En el PROFINET IO-Device Transmitir datos de entrada de procesos al PROFINET IO-Controller.
PNIO_RECV	x	-	Dependiendo del modo de funcionamiento del CP: • En el PROFINET IO-Controller Recibir datos de entrada de procesos de los PROFINET IO-Devices. • En el PROFINET IO-Device Recibir datos de salida de procesos del PROFINET IO-Controller.

Para CPs en funcionamiento paralelo de PROFINET IO−Controller e IO-Device están disponibles las instrucciones a partir de la versión 2.0.

Para la transmisión de datos acíclica (registros de datos, informaciones de alarma) en la interfaz PROFINET IO están disponibles las instrucciones citadas a continuación. Ambas instrucciones se pueden utilizar sólo en el modo de PROFINET IO-Controller.

Instrucción	utilizable para		Significado
S7-300	S7-400		
PNIO_RW_REC	x	-	• Leer registro de datos (de un PROFINET IO-Device) • Escribir registro de datos (en un PROFINET IO-Device)
PNIO_ALARM	x	-	Recibir informaciones de alarma de los PROFINET IO-Devices

Instrucciones para la transmisión cíclica de datos útiles

PNIO_SEND

PNIO_SEND

Descripción

La instrucción PNIO_SEND se utiliza para la transferencia de datos en los modos de funcionamiento del CP como controlador PROFINET IO o dispositivo PROFINET IO.

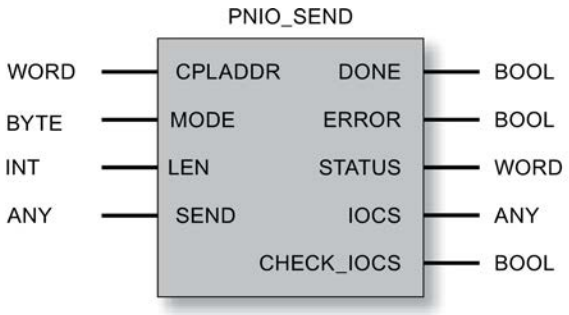
- Funcionamiento como controlador PROFINET IO
  La instrucción transfiere los datos de proceso (salidas) de un área de salida indicada al CP, para su transmisión a los dispositivos PROFINET IO y proporciona como indicador de estado el IO Consumer Status (IOCS) de las salidas de los dispositivos PROFINET IO.
- Funcionamiento como dispositivo PROFINET IO
  La instrucción lee las entradas de procesos pretratadas de la CPU en el dispositivo PROFINET IO y las transfiere al controlador PROFINET IO (direcciones de entrada configuradas). Adicionalmente, la instrucción proporciona como indicador de estado el IO Consumer Status (IOCS) del controlador PROFINET IO.

Los datos de proceso pretratados se ponen a disposición en un DB o en el área de marcas.

Para CPs en funcionamiento paralelo de controlador y dispositivo PROFINET IO está disponible la instrucción a partir de la versión 2.0. Con el parámetro adicional MODE se ajusta el modo de operación para el que se debe llamar la instrucción.

Interfaz de llamada (versión 2.0)

Interfaz de llamada en representación FUP



Consulte también

Comportamiento general de las instrucciones para PROFINET IO (Página 8106)

Coherencia de los datos (Página 8107)

Parámetros en el caso de PNIO_SEND (Página 8094)

Parámetros DONE, ERROR, STATUS (IE) (Página 8098)

Parámetros en el caso de PNIO_SEND

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción PNIO_SEND:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CPLADDR	INPUT	WORD	-	Dirección inicial del módulo
MODE (parámetros a partir de la versión 2.0)	INPUT	BYTE	Valores para X: • $0Y_H$: los bits de estado se transfieren a IOCS. • $8Y_H$: limitación a aviso de grupo en CHECK_IOCS; ningún bit de estado en IOCS. Valores para Y: • $X0_H$: – Modo de controlador IO – Modo de dispositivo IO (cuando no hay funcionamiento paralelo) Hay compatibilidad con la instrucción en la versión 1.0 • $X1_H$: modo de dispositivo IO (con funcionamiento paralelo)	Indicación en forma XY (hexadecimal): • X ajusta la transferencia de información de estado. • Y indica el modo de operación del CP como controlador IO o dispositivo IO. Observaciones sobre la compatibilidad: • La instrucción de la versión 1.0 se puede seguir utilizando si el CP no opera en paralelo como controlador IO y como dispositivo IO. • La instrucción de la versión a partir de 2.0 se comporta con MODE=0 como la instrucción de la versión 1.0.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
SEND	IN_OUT	ANY (como VARTYPE solo se permite BYTE)	La dirección del área de datos remite como alternativa a: • Área de marcas • Área de bloques de datos	Indicar la dirección y la longitud **Modo de controlador IO:** La longitud debería ser acorde con la longitud total configurada de la periferia descentralizada; los huecos en las direcciones se transmiten conjuntamente. La longitud puede ser también menor que la longitud total de la periferia descentralizada, por ejemplo si la instrucción se llama varas veces en 1 OB. Sin embargo debe tener la longitud total en al menos una llamada. **Modo de dispositivo IO:** La estructura de datos resulta del orden de los slots de los módulos de entrada configurados en la línea del controlador PROFINET IO para este dispositivo PROFINET IO y de su longitud sin huecos en direcciones. (Observe al respecto las explicaciones complementarias o los ejemplos relativos a su CP en la Parte B de este manual, específica del equipo) Notas: • La instrucción comienza con la transmisión de datos con la dirección 0, independientemente de cómo haya configurado las direcciones (con independencia de la mínima dirección configurada). • No se permite indicar un área de periferia, ya que primero tiene que comprobar el IOCS en cuanto a GOOD, antes de que se puedan adoptar los datos en la periferia.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
LEN	INPUT	INT	Valor > 0 Consulte la máxima longitud total de las áreas de datos a transmitir en la Parte B de este manual, específica del equipo, en el capitulo “Datos de rendimiento“. Puede ser diferente para el modo de controlador y el modo de dispositivo.	Longitud del sector de datos a transmitir en bytes. La transmisión de los datos comienza forzosamente con la dirección 0, con independencia de la configuración. Observe que se tiene en cuenta la dirección IO "0" con la longitud=1. **Modo de controlador IO:** • Aquí se tiene que indicar la dirección configurada de valor más alto de los dispositivos. Las distintas áreas no se agrupan. En el caso de una llamada repetida de la instrucción, LEN puede ser también menor que la mayor dirección. Al menos en una llamada se debería indicar la mayor dirección (véase el parámetro "SEND"). • Los datos se transmiten en el orden de las direcciones lógicas (como en PROFIBUS DP). **Modo de dispositivo IO:** • Los datos se transmiten en el orden de los slots, tal como se han configurado los módulos de entrada en la línea del controlador PROFINET IO para este dispositivo PROFINET IO. Nota: Tiene que asegurarse de que haya coherencia entre la longitud aquí programada y la configuración del controlador PROFINET IO. En el caso del dispositivo se transmite la longitud total del área de datos, incluidas eventuales lagunas.
DONE	OUTPUT	BOOL	0: - 1: nuevos datos aceptados	El parámetro de estado indica si la petición se ha desarrollado correctamente.
ERROR	OUTPUT	BOOL	0: -1: Error	Indicador de error
STATUS	OUTPUT	WORD	-	Indicador de estado

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CHECK_IOCS	OUTPUT	BOOL	0: todos los IOCS en GOOD 1: al menos un IOCS en BAD	Bit auxiliar que señaliza si es necesario evaluar el área de estado IOCS
IOCS	OUTPUT	ANY (como VARTYPE solo se permite BYTE)	La dirección del área de datos remite como alternativa a: • Área de marcas • Área de bloques de datos Longitud: Vea el valor máximo en la Parte B de este manual, específica del equipo, capítulo “Datos de rendimiento“. Puede ser diferente para el modo de controlador y el modo de dispositivo.	Por cada byte de datos útiles se transmite un bit de estado. El dato de longitud depende de la longitud en el parámetro LEN (un bit por cada byte) = (longitud LEN + 7/ 8) Modo de controlador: conforme al parámetro SEND se transmiten también huecos en direcciones. Huecos en direcciones se transmiten con el estado GOOD. Modo de dispositivo: los huecos en direcciones no se transmiten. La instrucción comienza con la transmisión del estado para la dirección 0. Nota: la longitud mínima del puntero ANY es (longitud LEN + 7/ 8)

Nota

Tenga en cuenta que todos los parámetros de salida solo se deben evaluar cuando la instrucción señalice DONE = 1 o ERROR = 1.

Nota

Tiene que partir de que el estado de IOCS suministrado no llega sincronizado en cuanto al tiempo con los datos (parámetro SEND), sino con una demora equivalente al ciclo del programa de usuario. Esto significa: Los datos de usuario y IOCS no son coherentes.

Consulte también

PNIO_SEND (Página 8092)

Parámetros DONE, ERROR, STATUS (IE)

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga también en cuenta lo indicado en el manual de referencia STEP 7 Standard y funciones del sistema. Allí encontrará informaciones en el capítulo "Evaluación de fallos con el parámetro de salida RET_VAL".

A través del diálogo de propiedades de las instrucciones aquí descritas se puede visualizar en la ficha “Llamadas" qué instrucciones del sistema que son relevantes para la evaluación de errores se utilizan.

Tabla 9-336 Códigos de condición PNIO_SEND

DONE	ERROR	STATUS	Significado
0	0	8180_H	• Transfiriendo datos; o bien • El CP está en el estado operativo STOP.
0	0	8181_H	El módulo no es compatible con la versión 2.0 de la instrucción. Solución: utilizar la versión 1.0.
1	0	0000_H	Nuevos datos transferidos sin error.
0	1	8183_H	• Falta configuración PROFINET IO; o bien • CPLADDR incorrecta; o bien • El CP está en el estado operativo STOP. o bien • El conexionado de MODE no armoniza con la configuración del módulo o se tiene un conexionado incorrecto con MODE > 1 Para el modo Device, adicionalmente: • La conexión entre PROFINET IO-Controller y PROFINET IO-Device está interrumpida, o bien • PROFINET IO-Controller no accesible o bien • Longitudes totales (configuración y parámetro LEN) no coherentes.
0	1	8184_H	Error del sistema o tipo de parámetro no permitido.
0	1	8185_H	El parámetro LEN es mayor que el área de origen SEND o el búfer de destino (IOCS) no es suficientemente grande.
0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).

DONE	ERROR	STATUS	Significado
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro.
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos act. protegido de escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos act. protegido de escritura.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F44_H$	Está bloqueado el acceso a un parámetro a leer para el procesamiento de la instrucción.
0	1	$8F45_H$	Está bloqueado el acceso a un parámetro a escribir para el procesamiento de la instrucción.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	1	8090_H	No existe ningún módulo con esta dirección.
0	1	$80A0_H$	Acuse negativo al leer del módulo.
0	1	$80A1_H$	Acuse negativo al escribir en el módulo.
0	1	$80B0_H$	El módulo desconoce el registro.
0	1	$80B1_H$	• La longitud de registro indicada es incorrecta. o bien • El CP pasa al estado STOP.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Recursos (memoria) ocupados.
0	1	$80C4_H$	Error de comunicación (aparece temporalmente; por ello es conveniente una repetición en el programa de usuario.)

Consulte también

PNIO_SEND (Página 8092)

PNIO_RECV

PNIO_RECV

Descripción

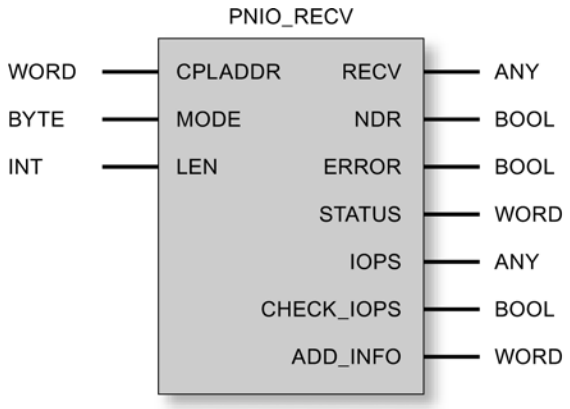
La instrucción PNIO_RECV se utiliza para la transmisión de datos en los modos de funcionamiento del CP como controlador PROFINET IO o dispositivo PROFINET IO.

- Funcionamiento como controlador PROFINET IO
  La instrucción transfiere los datos de proceso de los dispositivos PROFINET IO (entradas del controlador), así como el IO Provider Status (IOPS) de los dispositivos PROFINET IO a las áreas de entrada indicadas.
- Funcionamiento como dispositivo PROFINET IO
  La instrucción transfiere los datos de proceso transmitidos por el controlador PROFINET IO (direcciones de salida configuradas), así como el IO Provider Status (IOPS) del controlador PROFINET IO y los escribe en las áreas de datos reservadas para las salidas de procesos en la CPU del dispositivo PROFINET IO.

Para CPs en funcionamiento paralelo de controlador y dispositivo PROFINET IO está disponible la instrucción a partir de la versión 2.0. Con el parámetro adicional MODE se ajusta el modo de operación para el que se debe llamar la instrucción.

Interfaz de llamada (versión 2.0)

Interfaz de llamada en representación FUP



Consulte también

Comportamiento general de las instrucciones para PROFINET IO (Página 8106)

Coherencia de los datos (Página 8107)

Parámetros en el caso de PNIO_RECV (Página 8101)

Parámetros NDR, ERROR, STATUS (IE) (Página 8105)

Parámetros en el caso de PNIO_RECV

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción PNIO_RECV:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CPLADDR	INPUT	WORD	-	Dirección inicial del módulo
MODE (parámetros a partir de la versión 2.0)	INPUT	BYTE	Valores para X: • $0Y_H$: los bits de estado se transfieren a IOCS. • $8Y_H$: limitación a aviso de grupo en CHECK_IOCS; ningún bit de estado en IOCS. Valores para Y: • $X0_{H:}$ – Modo de controlador IO – Modo de dispositivo IO (cuando no hay funcionamiento paralelo) Hay compatibilidad con la instrucción en la versión 1.0 • $X1_H$: modo de dispositivo IO (con funcionamiento paralelo)	Indicación en forma XY (hexadecimal): • X ajusta la transferencia de información de estado. • Y indica el modo de operación del CP como controlador IO o dispositivo IO. Observaciones sobre la compatibilidad: • La instrucción de la versión 1.0 se puede seguir utilizando si el CP no opera en paralelo como controlador IO y como dispositivo IO. • La instrucción de la versión a partir de 2.0 se comporta con MODE=0 como la instrucción de la versión 1.0.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
RECV	IN_OUT	ANY (como VARTYPE solo se permite BYTE)	La dirección del área de datos remite como alternativa a: • Área de marcas • Área de bloques de datos	Indicar la dirección y la longitud Modo de controlador IO: La longitud debería ser acorde con la longitud total configurada de la periferia descentralizada; los huecos en las direcciones se transmiten conjuntamente. La longitud puede ser también menor que la longitud total de la periferia descentralizada, por ejemplo si la instrucción se llama varas veces en 1 OB. Sin embargo debe tener la longitud total en al menos una llamada. Modo de dispositivo IO: La estructura de datos resulta del orden de las ranuras (slots) de los módulos de salida configurados en el ramal del PROFINET IO-Controller para este PROFINET IO-Device y de su longitud sin huecos en direcciones. Notas: • La instrucción comienza con la transmisión de datos con la dirección 0, independientemente de cómo haya configurado las direcciones (con independencia de la mínima dirección configurada). • No se permite indicar un área de periferia, ya que primero tiene que comprobar el IOPS en cuanto a GOOD, antes de que se puedan adoptar los datos en la periferia.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
LEN	INPUT	INT	Valor > 0 Consulte la máxima longitud total de los datos a transmitir en la Parte B de este manual en el capitulo “Datos de rendimiento“. Puede ser diferente para el modo de controlador y el modo de dispositivo.	Longitud del sector de datos a transmitir en bytes. La transmisión de los datos comienza forzosamente con la dirección 0, con independencia de la configuración. Observe que se tiene en cuenta la dirección IO "0" con la longitud=1. Modo de controlador IO: • Aquí se tiene que indicar la dirección configurada de valor más alto de los dispositivos. Las distintas áreas no se agrupan. En el caso de una llamada repetida de la instrucción, LEN puede ser también menor que la mayor dirección. Al menos en una llamada se debería indicar la mayor dirección (véase el parámetro "RECV"). • Los datos se transmiten en el orden de las direcciones lógicas (como en PROFIBUS DP). Modo de dispositivo IO: • Los datos se transmiten en el orden de los slots, tal como se han configurado los módulos de entrada en la línea del controlador PROFINET IO para este dispositivo PROFINET IO. • Nota: Tiene que asegurarse de que haya coherencia entre la longitud aquí programada y la configuración del controlador PROFINET IO. En el caso del dispositivo se transmite la longitud total del área de datos, incluidas eventuales lagunas.
NDR	OUTPUT	BOOL	0: - 1: datos aceptados	El parámetro de estado indica si la petición se ha desarrollado correctamente.
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error
STATUS	OUTPUT	WORD	-	Indicador de estado
CHECK_ IOPS	OUTPUT	BOOL	0: todos los IOPS en GOOD 1: al menos un IOPS en BAD	Bit auxiliar que señaliza si es necesario evaluar el área de estado IOPS

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
IOPS	OUTPUT	ANY (como VARTYPE solo se permite BYTE)	La dirección del área de datos remite como alternativa a: • Área de marcas • Área de bloques de datos Longitud: Consulte el valor máximo en la Parte B de este manual en el capítulo “Datos de rendimiento“. Puede ser diferente para el modo de controlador y el modo de dispositivo.	Por cada byte de datos útiles se transmite un bit de estado. El dato de longitud depende de la longitud en el parámetro RECV (un bit por cada byte) = (longitud LEN + 7/ 8) Modo Controller: Conforme al parámetro RECV se transmiten también huecos en direcciones. Huecos en direcciones se transmiten con el estado GOOD. Modo Device: Huecos en direcciones no se transmiten. La instrucción comienza con la transmisión del estado para la dirección 0. Nota: • La longitud mínima del puntero ANY es (longitud LEN + 7/ 8)
ADD_INFO	OUTPUT	WORD	Información de diagnóstico adicional En el modo Controller: • 0: ninguna alarma • >0: cantidad de alarmas existentes En el modo Device el parámetro es siempre = 0.	Extensión de parámetro Nota: el parámetro ADD_INFO se actualiza también aunque en el controlador PROFINET IO no esté configurada ninguna dirección INPUT. En este caso se llama la instrucción PNIO_RECV con una longitud LEN > 0 (p. ej. LEN = 1 byte). Transmite entonces una laguna de direcciones de 1 Byte. La extensión de parámetros se puede utilizar para CPs a partir de la siguiente versión de firmware (FW): • CP 343−1 (EX30) a partir de FW V2.0 • CP 343−1 Lean (CX10) a partir de FW V2.0 • CP 343−1 Advanced (GX30) a partir de FW V1.0 En versiones de firmware más antiguas, el parámetro está reservado.

Nota

Tenga en cuenta que todos los parámetros de salida solo se deben evaluar cuando la instrucción señalice NDR = 1 o ERROR = 1.

Consulte también

PNIO_RECV (Página 8099)

Parámetros NDR, ERROR, STATUS (IE)

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por los parámetros NDR, ERROR y STATUS, que tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga también en cuenta lo indicado en el manual de referencia STEP 7 Standard y funciones del sistema. Allí encontrará informaciones en el capítulo "Evaluación de fallos con el parámetro de salida RET_VAL".

A través del diálogo de propiedades de las instrucciones aquí descritas se puede visualizar en la ficha “Llamadas" qué instrucciones del sistema que son relevantes para la evaluación de errores se utilizan.

Tabla 9-337 Códigos de condición de PNIO_RECV

NDR	ERROR	STATUS	Significado
0	0	8180_H	• Adoptando datos; o bien • El CP está en el estado operativo STOP.
0	0	8181_H	El módulo no es compatible con la versión 2.0 de la instrucción. Solución: utilizar la versión 1.0.
1	0	0000_H	Nuevos datos aceptados sin error.
0	1	8183_H	• Falta configuración PROFINET IO; o bien • CPLADDR incorrecta; o bien • El CP está en el estado operativo STOP. o bien • El conexionado de MODE no armoniza con la configuración del módulo o se tiene un conexionado incorrecto con MODE > 1. Para el modo Device, adicionalmente: • La conexión entre PROFINET IO-Controller y PROFINET IO-Device está interrumpida o bien • PROFINET IO-Controller no accesible o bien • Longitudes totales (configuración y parámetro LEN) no coherentes.
0	1	8184_H	Error del sistema o tipo de parámetro no permitido.
0	1	8185_H	Búfer de destino (RECV o IOCS) demasiado pequeño.
0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).

NDR	ERROR	STATUS	Significado
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro.
0	1	$8F30_H$	El parámetro se encuentra en el 1er. bloque de datos act. protegido contra escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos act. protegido contra escritura.
0	1	$8F32_H$	El parámetro contiene un número de DB demasiado alto.
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F44_H$	Está bloqueado el acceso a un parámetro a leer para el procesamiento de la instrucción.
0	1	$8F45_H$	Está bloqueado el acceso a un parámetro a escribir para el procesamiento de la instrucción.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	1	8090_H	No existe ningún módulo con esta dirección.
0	1	$80A0_H$	Acuse negativo al leer del módulo.
0	1	$80A1_H$	Acuse negativo al escribir en el módulo.
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	• La longitud de registro indicada es incorrecta. o bien • El CP pasa al estado operativo STOP.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Medio de servicio (memoria) ocupado.
0	1	$80C4_H$	Error de comunicación (se produce temporalmente; por ello es conveniente una repetición en el programa de usuario.)

Consulte también

PNIO_RECV (Página 8099)

Comportamiento general de las instrucciones para PROFINET IO

IO Consumer Status (IOCS) y IO Provider Status (IOPS)

En los dos interlocutores de comunicación, CPU/CP por un lado e IO-Device por otro, hay disponible respectivamente una información de estado GOOD o BAD para los datos. Esta información de estado se transmite paralela a los datos. El estado del interlocutor que envía los datos se denomina IOPS (IO Provider Status) y el estado del interlocutor que los recibe IOCS (IO Consumer Status).

El estado IOPS y el estado IOCS no son forzosamente idénticos. Puede ocurrir, por ejemplo, que la CPU S7-300 se encuentre en el estado Stop (Output Disable o no se ejecutan

instrucciones PROFINET IO). En tal caso, el CP transmite, como PROFINET IO-Controller, el estado BAD a los IO-Devices.

Relación entre llamada de la instrucción y datos IO

- Funcionamiento como PROFINET IO-Controller
  Como PROFINET IO-Controller, el CP no supervisa la llamada cíclica de las instrucciones PNIO_SEND/RECV. Si no se llaman las instrucciones, se considerarán válidos los últimos datos IO y IOCS/IOPS transmitidos.
- Funcionamiento como PROFINET IO-Device
  PNIO_SEND y PNIO_RECV tienen respectivamente un Watchdog propio. Dependiendo del tiempo de ciclo de la CPU se deshace la conexión con el PROFINET IO-Controller si tras la fase de inicialización no se llama ya más una de las dos instrucciones.

Optimización de la transmisión de datos (sólo en el modo de PROFINET IO-Controller)

Es posible llamar las instrucciones con una longitud (parámetro LEN) menor que la longitud total configurada de los datos de E/S en el ramal PNIO.

Esto se puede aprovechar en el sentido de que datos críticos en cuanto al tiempo se transmitan en cada ciclo de CPU, no transmitiéndose por el contrario en cada ciclo datos no críticos.

Ejemplo:
Transmita, por ejemplo, en cada ciclo sólo el primer sector de datos (datos críticos en cuanto al tiempo) y en cada segundo ciclo la longitud total de los datos de E/S configurados. Para ello, al configurar tiene que poner los datos críticos en cuanto al tiempo en el sector inferior (a partir de la dirección de E/S 0).

Coherencia de los datos

Se transmite siempre todo el área de datos de entradas y salidas del controlador PROFINET IO de forma completa y, por lo tanto, coherente.

- Funcionamiento como PROFINET IO-Controller
  Independientemente de ello, indicando la longitud en la llamada de la instrucción tiene la posibilidad de leer o emitir de forma coherente un área de datos de entrada o salida menor que la configurada.

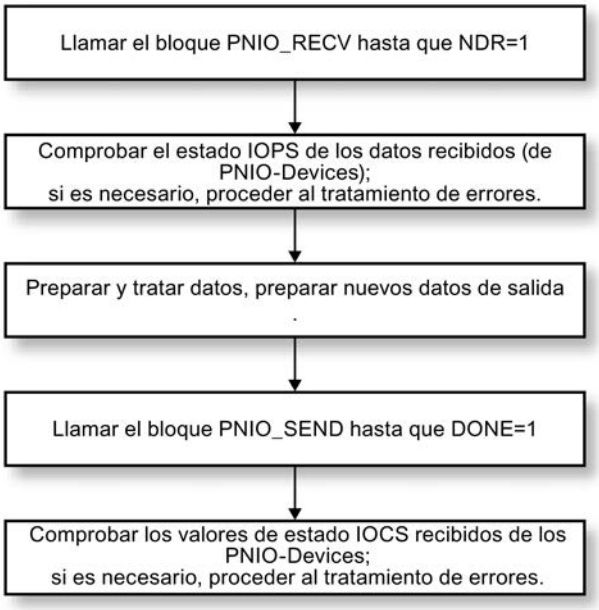
Observación: Tenga en cuenta, sin embargo, que en lo que se refiere a los "datos útiles IO", dentro de un sistema PROFINET IO sólo se puede garantizar la coherencia de los datos dentro de los distintos slots IO. Esto es independiente de que para las instrucciones aquí descritas se garantice una transferencia de datos coherente entre la CPU y el IO-Controller.

Llamada de la instrucción

Sin embargo, para asegurar la coherencia de los datos sólo se debe acceder a los datos IO si la instrucción se ha finalizado sin errores (parámetro Output NDR = TRUE). Además se tiene que comprobar si el estado IOCS o IOPS para los datos es = GOOD.

Ejemplo

Normalmente (dependiendo de la longitud total de los datos IO), la instrucción se ejecuta a lo largo de varios ciclos del programa de usuario, hasta que se notifica el indicador DONE/NDR = 1.



Observación: El ciclo del programa de usuario y el ciclo del intercambio de datos IO entre el controlador PROFINET IO y los dispositivos PROFINET IO son independientes el uno del otro.

Valores de sustitución

Casos de operación

La conexión de valores de sustitución se soporta para los dos casos de operación siguientes:

- Valores de sustitución en el arranque (cambio de estado operativo de la CPU de STOP a RUN)
- Valores sustitutivos en caso de anomalías (desenchufado/enchufado o fallo/restauración de la estación)

Valores de sustitución en el arranque

Las salidas se pueden inicializar con valores de sustitución poniendo en el OB de arranque un marcador ("Arranque"). En régimen cíclico (OB1), evalúe entonces este marcador de "arranque" para llamar, si procede, la instrucción PNIO_SEND con los valores de inicialización.

Valores sustitutivos en caso de anomalías (sólo en el modo de PROFINET IO-Controller)

En caso de avería (ha fallado el device/módulo) puede determinar qué módulos han fallado consultando las informaciones de estado IOCS / IOPS. Entonces tiene la posibilidad de aplicar valores sustitutivos.

PNIO_RW_REC

PNIO_RW_REC

Descripción

RW_REC sirve, en el modo de controlador PROFINET IO, tanto para la función "Leer registro" como para la función "Escribir registro". RW_REC solo puede ejecutar una de las dos funciones al mismo tiempo. La función “Leer registro" o “Escribir registro" se controla a través del parámetro WRITE_REC.

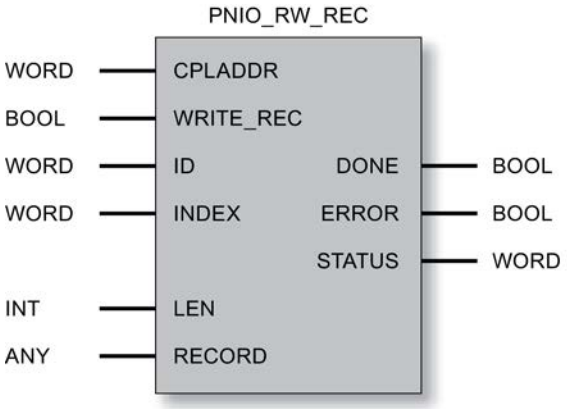
Ejemplo: El indicativo del sistema y el indicativo de localización se le pueden comunicar al CP a través de la función "Escribir registro" (en tanto estos parámetros no se hayan ajustado ya en STEP 7 en el diálogo de propiedades del CP). Para ello se usa el registro de mantenimiento “IM1" con el índice AFF1H.

Si desea detalles sobre los registros a los que se da soporte así como sobre su estructura, puede consultarlos a través de la siguiente dirección de Internet:

`http://support.automation.siemens.com/WW/view/de/19289930`

Interfaz de llamada

Interfaz de llamada en representación FUP:



Consulte también

Parámetros DONE, ERROR, STATUS (IE) (Página 8111)

Parámetros en el caso de PNIO_RW_REC (Página 8110)

Parámetros en el caso de PNIO_RW_REC

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción PNIO_RW_REC:

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CPLADDR	INPUT	WORD	-	Dirección inicial del módulo
WRITE_REC	INPUT	BOOL	0: Leer registro de datos 1: Escribir registro	Tipo de petición; El parámetro no se debe modificar durante el tiempo de ejecución de la instrucción.
ID	INPUT	WORD		Dirección lógica del componente PROFINET IO (grupo o módulo). En caso de un módulo de salida se tiene que poner Bit 15 (ejemplo para la dirección de salida 5: ID:=DW#16#8005). En caso de un módulo mixto se tiene que indicar la menor de ambas direcciones.
INDEX	INPUT	WORD	Véase en la información del fabricante qué números de registro de datos son soportados por el grupo o el módulo.	Números de los registros de datos que el usuario desea leer o escribir.
DONE	OUTPUT	BOOL	0: - 1: Registro de datos transmitido correctamente	El parámetro de estado indica si se han aceptado nuevos datos.
ERROR	OUTPUT	BOOL	0: - 1: error	Indicación de fallo
STATUS	OUTPUT	WORD	0: ningún error otro valor: error (ver "Auto-Hotspot")	Indicación de estado

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
LEN	IN_OUT	INT	La longitud máxima es 480 Byte.	• Leer registro de datos: Parámetro OUTPUT puro; tras una lectura correcta se indica aquí la longitud del registro de datos leído; en otro caso 0. • Escribir registro de datos: Parámetro INPUT puro; el usuario tiene que indicar aquí la longitud del registro de datos a escribir. La longitud tiene que ser conforme a la definición del registro de datos.
RECORD	IN_OUT	ANY (como VARTYPE se permiten BYTE, WORD y DWORD)	La dirección del sector de datos remite como alternativa a: • Área de marcas • Área de bloques de datos La longitud del puntero Any tiene que ser mayor o igual a la definición del registro de datos.	• Leer registro de datos: Parámetro OUTPUT puro; tras una lectura correcta se almacenan aquí los datos del registro de datos. Si la longitud del puntero ANY es insuficiente, se transmiten tantos datos como sea posible. • Escribir registro de datos: Parámetro INPUT puro; el usuario almacena aquí los datos a escribir del registro de datos. La longitud del puntero ANY tiene que ser al menos tal como predetermina el parámetro LEN.

Consulte también

PNIO_RW_REC (Página 8108)

Parámetros DONE, ERROR, STATUS (IE)

Indicadores

La tabla siguiente informa sobre la indicación a evaluar por el programa de usuario, formada por DONE, ERROR y STATUS.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, observe también lo indicado en el manual de referencia "STEP 7 - Funciones del sistema y estándar para S7-300 y S7-400". Allí encontrará informaciones en el capítulo "Evaluación de fallos con el parámetro de salida RET_VAL".

Tabla 9-338 Indicadores PNIO_RW_REC

DONE	ERROR	STATUS	Significado
0	0	8180_H	Transfiriendo datos
1	0	0000_H	Registro de datos transmitido correctamente

DONE	ERROR	STATUS	Significado
0	1	8183_H	• Falta la configuración del controlador PROFINET IO, • CPLADDR incorrecta o bien • CP en estado operativo STOP
0	1	8184_H	Error del sistema o tipo de parámetro no permitido
0	1	8185_H	Búfer de destino (RECORD) demasiado pequeño.
0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	Error de campo al leer un parámetro
0	1	$8F25_H$	Error de campo al escribir un parámetro
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos activo protegido de escritura
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos activo protegido de escritura
0	1	$8F32_H$	El parámetro contiene número de DB demasiado grande
0	1	$8F3A_H$	Área de destino no cargada (DB)
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia
0	1	$8F44_H$	Está bloqueado el acceso a un parámetro a leer para el procesamiento de la instrucción
0	1	$8F45_H$	Está bloqueado el acceso a un parámetro a escribir para el procesamiento de la instrucción
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	1	8090_H	No existe módulo con esta dirección
0	1	$80A0_H$	Acuse negativo al leer del módulo
0	1	$80A1_H$	Acuse negativo al escribir en el módulo
0	1	$80A3_H$	Error general del PROFINET IO-Context-Management
0	1	$80A9_H$	PROFINET IO-Device o módulo señaliza un tipo no permitido
0	1	$80B0_H$	El módulo no conoce el registro
0	1	$80B1_H$	• La longitud de registro de datos indicada es incorrecta o bien • El CP pasa al estado STOP
0	1	$80B2_H$	La dirección lógica o la ranura configurada no está ocupada
0	1	$80B4_H$	PROFINET IO-Device o módulo señaliza un acceso a un sector no permitido
0	1	$80B6_H$	PROFINET IO-Device o módulo deniega el acceso
0	1	$80B8_H$	El módulo señaliza un parámetro no permitido
0	1	$80C0_H$	No se puede leer el registro de datos
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Medios de operación (memoria) ocupados
0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.)

Consulte también

PNIO_RW_REC (Página 8108)

PNIO_Alarm

PNIO_ALARM

Descripción

La instrucción PNIO_ALARM sirve para la evaluación de alarmas por un CP 343-1 empleado como controlador PROFINET IO y se debería llamar en el programa de usuario del mismo si en PNIO_RECV el parámetro ADD_INFO es distinto de 0. Tras una transmisión completa y sin errores de todos los parámetros OUTPUT del PNIO_ALARM se confirman automáticamente las alarmas recibidas.

Las alarmas se transmiten en el orden cronológico de su señalización al programa de usuario. Alarmas más antiguas, aún no señalizadas al programa de usuario, que se tornen obsoletas debido a otras alarmas más recientes, no son borradas por nuevas alarmas.

Nota

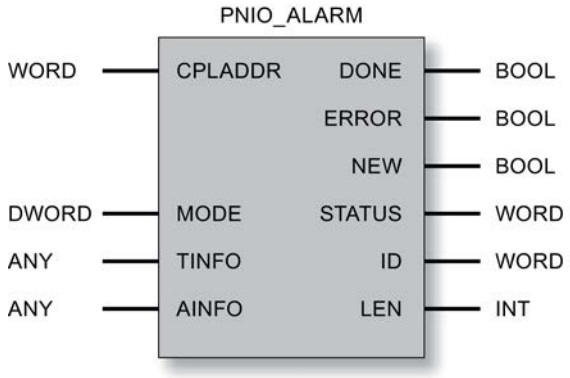
Mientras no se haya llamado aún PNIO_ALARM, las alarmas se acusan automáticamente a nivel interno del CP.

Si PNIO_ALARM se ha llamado (al menos) una vez en el programa de usuario, también se tiene que seguir llamando para acusar recibo de alarmas pendientes. Tal es el caso si la instrucción PNIO_RECV presenta en el parámetro ADD_INFO un valor distinto de "0".

Si PNIO_ALARM no se vuelve a llamar después de haberlo llamado una o más veces en el programa de usuario, no se acusa recibo de las alarmas y no está garantizado que la representación de IO se actualice correctamente. Tal puede ser, por ejemplo, el caso tras una alarma de retorno de estación. La necesidad de la llamada de la instrucción PNIO_ALARM solo se puede anular con un rearranque del CP (desconexión y reconexión de la tensión).

Interfaz de llamada

Interfaz de llamada en representación FUP



Consulte también

Parámetros DONE, NEW, ERROR, STATUS (IE) (Página 8116)

Parámetros en el caso de PNIO_ALARM (Página 8114)

Parámetros en el caso de PNIO_ALARM

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción PNIO_ALARM:

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CPLADDR	INPUT	WORD	-	Dirección inicial del módulo causante de error
DONE	OUTPUT	BOOL	0: - 1: información de alarma transmitida correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente. Con DONE = 1 se tiene que comprobar adicionalmente el parámetro NEW.
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error
NEW	OUTPUT	BOOL	0: transmitiendo datos o no hay ninguna nueva alarma 1: nueva alarma recibida y acusada	Con DONE = 1 y NEW = 1 se señaliza aquí una nueva alarma recibida.
STATUS	OUTPUT	WORD	0: ningún error otro valor: error (ver tabla 1–14)	Indicador de estado

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ID	OUTPUT	WORD		Dirección inicial lógica del componente PNIO causante de la alarma (grupo o módulo). En caso de un módulo de salida se pone Bit 15 (ejemplo para la dirección de salida 5: ID:=DW#16#8005). En caso de un módulo mixto se indica la menor de ambas direcciones.
LEN	OUTPUT	INT		Longitud de la información de alarma recibida (AINFO)
MODE	IN_OUT	DWORD	0	Reservado
TINFO	IN_OUT	ANY (como VARTYPE se permiten BYTE, WORD y DWORD)	La dirección del sector de datos remite como alternativa a: • Área de marcas • Área de bloques de datos La longitud del puntero Any tiene que ser >= 32 bytes.	(task information) Área de destino para información de administración de alarmas. La información de inicio de OB de error (OB-Header = Byte 0...19 de TINFO) es reproducida por el firmware del CP, en la medida de lo posible. Véase también 1)
AINFO	IN_OUT	ANY (como VARTYPE se permiten BYTE, WORD y DWORD)	La dirección del sector de datos remite como alternativa a: • Área de marcas • Área de bloques de datos La longitud del puntero ANY tiene que ser mayor o igual que la máxima información adicional de alarma que se pueda recibir, pero como máximo 1432 bytes (ver el parámetro LEN)	(alarm information) Área de destino para información de cabecera e información adicional de alarma. Si ANY-Pointer AINFO es demasiado pequeño, se recorta la información. Véase también 1)

1) Manual de referencia "STEP 7 - Funciones del sistema y estándar para S7-300 y S7-400", Recibir alarma con la instrucción "RALRM"

Consulte también

PNIO_ALARM (Página 8112)

Parámetros DONE, NEW, ERROR, STATUS (IE)

Indicadores

La tabla siguiente informa sobre la indicación a evaluar por el programa de usuario, formada por DONE, NEW, ERROR y STATUS.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, observe también lo indicado en el manual de referencia "STEP 7 - Funciones del sistema y estándar para S7-300 y S7-400". Allí encontrará informaciones en el capítulo "Evaluación de fallos con el parámetro de salida RET_VAL".

Tabla 9-339 Indicadores PNIO_ALARM

DONE	NEW	ERROR	STATUS	Significado
0	0	0	8180_H	Transfiriendo datos
1	1	0	0000_H	Datos de alarma transmitidos correctamente y alarma acusada
1	0	0	0000_H	No hay datos de alarma
0	0	1	8183_H	• Falta la configuración del controlador PROFINET IO, • CPLADDR incorrecta o bien • CP en estado operativo STOP
0	0	1	8184_H	Error del sistema o tipo de parámetro no permitido
0	0	1	8185_H	Búfer de destino (TINFO o AINFO) demasiado pequeño
0	0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	0	1	$8F24_H$	Error de campo al leer un parámetro
0	0	1	$8F25_H$	Error de campo al escribir un parámetro
0	0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	0	1	$8F29_H$	Error de alineación al escribir un parámetro
0	0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos activo protegido de escritura
0	0	1	$8F31_H$	El parámetro está en el segundo bloque de datos activo protegido de escritura
0	0	1	$8F32_H$	El parámetro contiene número de DB demasiado grande
0	0	1	$8F3A_H$	Área de destino no cargada (DB)
0	0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia
0	0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia
0	0	1	$8F44_H$	Está bloqueado el acceso a un parámetro a leer para el procesamiento de la instrucción.
0	0	1	$8F45_H$	Está bloqueado el acceso a un parámetro a escribir para el procesamiento de la instrucción.
0	0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	0	1	8090_H	No existe módulo con esta dirección
0	0	1	$80A0_H$	Acuse negativo al leer del módulo
0	0	1	$80A1_H$	Acuse negativo al escribir en el módulo

DONE	NEW	ERROR	STATUS	Significado
0	0	1	$80B0_H$	El módulo no conoce el registro
0	0	1	$80B1_H$	• La longitud de registro de datos indicada es incorrecta o bien • El CP pasa al estado STOP
0	0	1	$80C0_H$	No se puede leer el registro de datos
0	0	1	$80C1_H$	El registro indicado está siendo procesado.
0	0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	0	1	$80C3_H$	Medios de operación (memoria) ocupados
0	0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.)

Consulte también

PNIO_ALARM (Página 8112)

PROFIenergy

PROFIenergy

Las funciones de PROFIenergy en PROFINET sirven para gestionar la energía de las instalaciones. Aquí se incluye la desconexión planificada o espontánea de diferentes aparatos de campo, grupos o unidades de proceso con el fin de ahorrar energía. La desconexión se lleva a cabo en tiempos no productivos o en pausas de producción. Los datos energéticos y de diagnóstico se pueden leer en aparatos integrados en el concepto de energía que, además, soporten estas funciones.

Controlador PROFIenergy

Los comandos de desconexión los da el controlador de nivel superior; en PROFINET IO, el controlador IO.

En SIMATIC S7-300, el controlador PROFIenergy puede ser el CP o una CPU S7-300 con función PROFIenergy.

Dispositivos PROFIenergy

Los comandos del controlador PROFIenergy son procesados por el dispositivo IO con función PROFIenergy, para desconectar aparatos conectados en el campo.

En el contexto de los bloques de programa PROFIenergy, se designa un dispositivo IO con función PROFIenergy como dispositivo PROFIenergy. En SIMATIC S7-300, el dispositivo PROFIenergy puede ser el CP o una CPU S7-300 con función PROFIenergy.

Dispositivos I

En SIMATIC S7, un dispositivo inteligente (dispositivo I) puede tener dispositivos PROFIenergy subordinados. En este caso, el dispositivo I también puede funcionar como un controlador PROFIenergy.

Modos de ahorro de energía y PE_MODE_ID

Muchos aparatos soportan sólo los estados operativos "operativo" (tensión conectada) y "pausa" (tensión desconectada). En el controlador del dispositivo PROFIenergy es posible configurar estados de ahorro de energía escalonados con diferentes modos de ahorro para aquellos aparatos que lo soporten o para grupos. Con PROFIenergy, a los aparatos de campo se les pueden asignar diferentes estados de consumo de energía.

Los distintos estados de consumo de energía se denominan "modos de ahorro de energía". Para cada modo de ahorro de energía se especifica una "PE_MODE_ID" concreta.

Programación de las propiedades de los modos de ahorro de energía

Los detalles acerca de los modos de ahorro de energía (aparato de campo activado, duración de la pausa, etc.) se programan en el programa de usuario de la CPU del dispositivo PROFIenergy.

Bloques de programa PROFIenergy para el CP 300

Realización de las funciones de PROFIenergy en un S7-300

En el SIMATIC S7-300 se preparan las funciones PROFIenergy para el controlador IO y el dispositivo IO mediante bloques de programa.

Tenga en cuenta que la CPU S7-300 y el CP 300 utilizan diferentes bloques de programa PROFIenergy.

Especificación PROFIenergy

Las funciones de los bloques de programa PROFIenergy para el CP 300 se basan en la siguiente especificación de la Organización de Usuarios PROFIBUS e.V. (PNO):

Common Application Profile PROFIenergy, Technical Specification for PROFINET, Version 1.0, January 2010, Order No. 3.802

Bloques de programa PROFIenergy para el CP 300

Los bloques de programa PROFIenergy se llaman desde el programa de usuario de la CPU. Para las funciones PROFIenergy del CP 300 están disponibles los siguientes bloques de programa PROFIenergy:

- CP 300 como controlador IO:
  - PE_START_END_CP
    Bloque de programa para iniciar y finalizar pausas en la alimentación de energía así como para predefinir determinados modos de ahorro de energía para el dispositivo PROFIenergy.
  - PE_CMD_CP
    Bloque de programa para iniciar y finalizar pausas en la alimentación de energía, predefinir determinados modos de ahorro de energía y consultar valores energéticos medidos del dispositivo PROFIenergy.

  Los dos bloques de programa pueden utilizarse alternativamente. En comparación con PE_START_END_CP, PE_CMD_CP tiene una funcionalidad avanzada para integrar valores energéticos medidos.
  El bloque de programa debe llamarse por separado para cada dispositivo PROFIenergy.
  - PE_DS3_WRITE_ET200_CP
    No forma parte de los bloques de función PROFIenergy, pero complementa las funciones de PROFIenergy de un ET 200S.
    Con PE_DS3_WRITE_ET200_CP se definen los ajustes para el comportamiento de conmutación de 8 slots como máximo (en este caso: módulos de potencia) del ET 200S.
- CP 300 como dispositivo IO:
  - PE_I_DEV_CP
    Recibe todos los comandos PROFIenergy y permite al programa de usuario ejecutar las funciones de PROFIenergy.
    Prepara los telegramas de respuesta del dispositivo IO para el controlador IO.
    PE_I_DEV_CP se llama cíclicamente desde el programa de usuario del dispositivo IO.
  - Bloques de programa complementarios (FC 0 - FC 8) para PE_I_DEV_CP:
    Estos FCs preparan los datos de respuesta para PE_I_DEV_CP. Los FCs deben llamarse en el programa de usuario y vincularse con PE_I_DEV_CP.

  Si el dispositivo PROFIenergy es un I-Device que tiene dispositivos PROFIenergy subordinados, en la CPU del I-Device se llama PE_START_END_CP o PE_CMD_CP para los dispositivos PROFIenergy subordinados.

Bloques de sistema y programa para transferir registros

Los comandos PROFIenergy y la información de estado se intercambian entre el controlador IO y el dispositivo IO mediante la lectura y escritura de registros. Esto se consigue por medio de los bloques de programa RDREC y RWREC.

Los registros PROFIenergy se describen a continuación en los datos de respuesta de los diferentes bloques de programa.

Nota

Llamadas de bloques

PE_START_END_CP, PE_CMD_CP, PE_I_DEV_CP y PE_DS3_WRITE_ET200_CP no deben llamarse al mismo tiempo. No se puede llamar el siguiente bloque de programa hasta que un bloque haya notificado "ningún error" (VALID = 1) o "error" (ERROR = 1).

El bloque de programa PNIO_RW_REC tampoco debe llamarse al mismo tiempo que PE_START_END_CP, PE_CMD_CP, PE_I_DEV_CP o PE_DS3_WRITE_ET200_CP.

PE_START_END_CP

Significado y llamada - PE_START_END_CP

Significado y funcionamiento

PE_START_END_CP puede utilizarse como alternativa a PE_CMD_CP.

PE_START_END_CP se utiliza en el controlador IO. Provoca una pausa de ahorro de energía o la finaliza en el dispositivo PROFIenergy asignado.

Este bloque de programa se utiliza preferentemente en controladores IO cuyos dispositivos IO asignados tengan conectados únicamente aparatos de campo en los que no pueden o deben leerse datos energéticos.

Los modos de ahorro de energía se configuran en el programa de usuario de la CPU del dispositivo IO. El modo de ahorro de energía que se adopta realmente es notificado por el dispositivo IO después de ejecutar PE_START_END_CP y devuelto en el parámetro PE_MODE_ID.

El parámetro Pause_Time especifica la duración de la pasa de ahorro de energía para el dispositivo IO. Mediante el bloque de programa PE_I_DEV_CP, en el dispositivo IO se comprueba si la duración predefinida para la pausa es suficiente y si puede aplicarse.

Secuencia

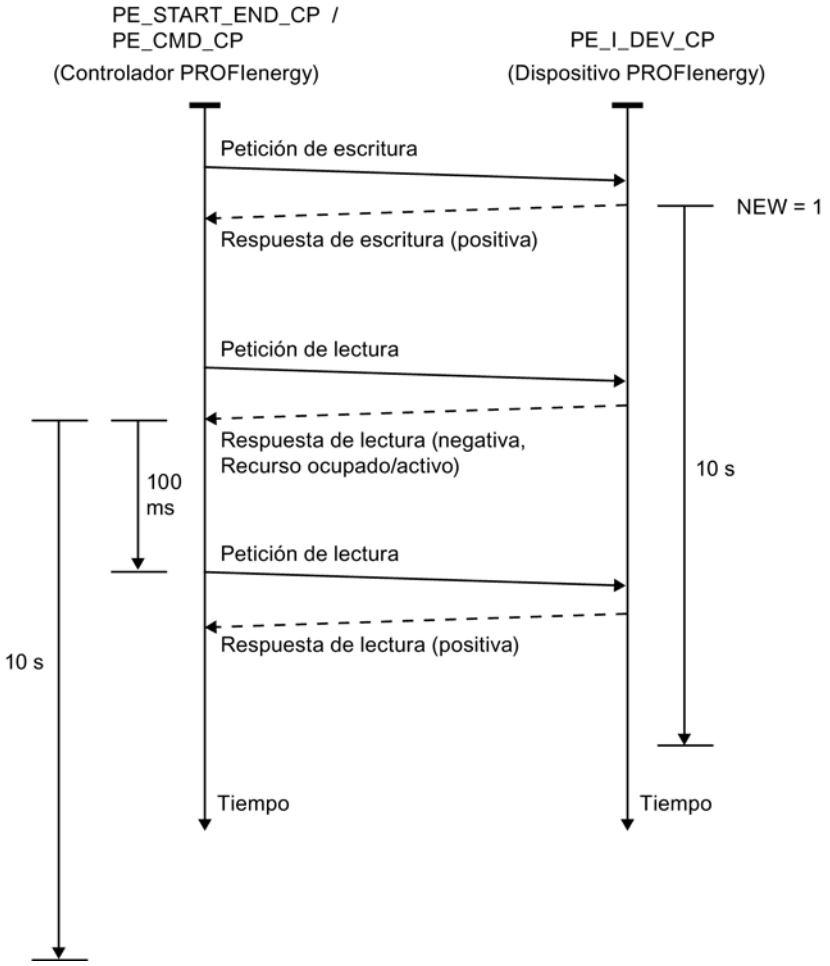


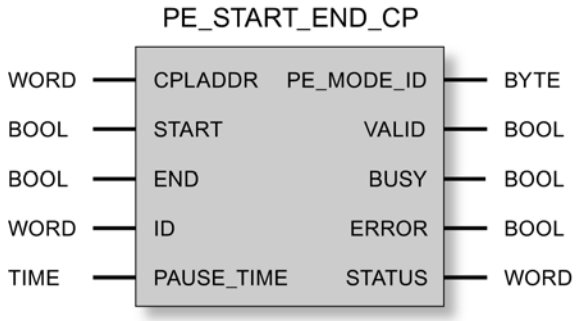
Figura 9-20 Diagrama secuencial de las peticiones de escritura/lectura de PE_START_END_CP y PE_CMD_CP

PE_START_END_CP utiliza WRREC para enviar un comando PROFIenergy al dispositivo IO como petición de escritura. Seguidamente, PE_START_END_CP espera el acuse del dispositivo IO. Para ello, cada 100 milisegundos se lee el registro de acuse con el bloque de programa RDREC.

Mientras no llega el acuse del dispositivo IO, la petición de lectura se repite durante 10 segundos en intervalos de 100 milisegundos.

Los datos de respuesta del dispositivo IO se leen con el RDREC.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales de PE_START_END_CP

Explicación de los parámetros formales de PE_START_END_CP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CPLADDR	INPUT	WORD	I, Q, M, D, L, const.	Dirección inicial del módulo del CP
START	INPUT	BOOL	• 1 = comando activo • 0 = comando no activo	Un flanco ascendente activa el comando "Start_Pause" (iniciar pausa de ahorro de energía)
END	INPUT	BOOL	• 1 = comando activo • 0 = comando no activo	Un flanco ascendente activa el comando "End_Pause" (finalizar pausa de ahorro de energía)
ID	INPUT	WORD		Dirección lógica del dispositivo PROFIenergy de destino
PAUSE_ TIME	INPUT	TIME	T#-24D_20H_31M_23S_648MS a T#24D_20H_31M_23S_647MS	Temporizador IEC en incrementos de 1 ms, entero con signo
PE_MODE_ ID	OUTPUT	BYTE	• 00_h: tensión OFF (pausa) • 01_h...FE_h: configurable • FF_h: operativo	ID del modo de ahorro de energía adoptado por el dispositivo IO tras la ejecución del comando.
VALID	OUTPUT	BOOL	0: - 1: procesamiento finalizado correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente.
BUSY	OUTPUT	BOOL	0: procesamiento finalizado, cancelado o todavía no iniciado 1: procesamiento en curso	Indicador del estado de procesamiento del bloque de programa

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Ver el significado en relación con el parámetro STATUS en Indicadores de PE_START_END_CP (Página 8123).
STATUS	OUTPUT	WORD		Indicador de estado Ver el significado en relación con el parámetro ERROR en Indicadores de PE_START_END_CP (Página 8123).

Consulte también

Datos de respuesta (Página 8129)

Indicadores de PE_START_END_CP

Indicadores de PE_START_END_CP

PE_START_END_CP se basa en el bloque de programa PNIO_RW_REC y proporciona todos los indicadores de PNIO_RW_REC, consulte los indicadores del bloque PNIO_RW_REC.

Además, se muestran los siguientes indicadores específicos de PROFIenergy. Los indicadores de error de STATUS solo son válidos en relación con ERROR = 1.

Tabla 9-340 Indicadores específicos de PE_START_END_CP

STATUS	Significado
Errores específicos del bloque	
8080_h	Flanco ascendente simultáneamente a START y END
8081_h	Conflicto de longitud con CMD_PARAM y CMD_PARAM_LEN
Errores específicos de PROFIenergy	
$FE01_h$	Service_Request_ID no válida
$FE02_h$	Request_Reference no válida
$FE03_h$	CMD_MODIFIER no válido
$FE04_h$	Indicación no válida sobre la estructura de datos de un comando (Data_Structure_Identifier_RQ) en el telegrama para el registro PROFIenergy de escritura
$FE05_h$	Indicación no válida sobre la estructura de datos de un comando (Data_Structure_Identifier_RS) en el telegrama para el registro PROFIenergy de lectura
$FE06_h$	Modo de ahorro de energía (PE_Mode_ID) no soportado
$FE07_h$	La respuesta es más larga que la longitud de transferencia máx.
$FE08_h$	Número de comandos no válido
$FE09_h$	Block Type no válido (consulte el encabezado del telegrama)
$FE0A_h$	Block Length no válida (consulte el encabezado del telegrama)
$FE0B_h$	Block Version no válida (consulte el encabezado del telegrama)
$FE50_h$	No hay ningún modo de ahorro de energía adecuado (PE_Mode_ID)

STATUS	Significado
$FE51_h$	No se soporta el valor de PAUSE_TIME
$FE52_h$	No se soporta PE_Mode_ID

Consulte los detalles relacionados con los parámetros de los errores específicos de PROFIenergy en el capítulo Datos de respuesta.

PE_CMD_CP

Significado y llamada - PE_CMD_CP

Significado y funcionamiento

PE_CMD_CP puede utilizarse como alternativa a PE_START_END_CP.

PE_CMD_CP se utiliza en el controlador IO y lanza una pausa de ahorro de energía o la finaliza en el dispositivo PROFIenergy asignado. Adicionalmente, PE_CMD_CP puede leer otras informaciones y valores energéticos medidos de un dispositivo IO.

Este bloque de programa se utiliza preferentemente en controladores IO cuyos dispositivos IO asignados tengan conectados aparatos de campo de los que deben leerse datos energéticos medidos.

Encontrará un diagrama de secuencia de las peticiones de escritura/lectura de PE_CMD_CP en el capítulo Significado y llamada - PE_START_END_CP.

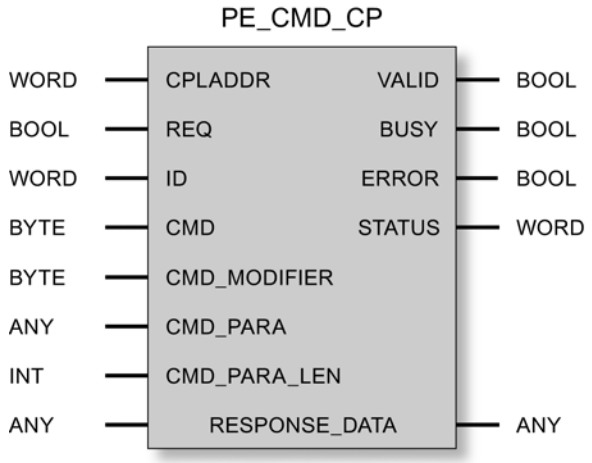
Los diferentes comandos que se transfieren al dispositivo IO con el bloque de programa tienen asignadas "Service_Request_IDs" concretas. Las Service_Request_IDs 01...05 y 16 se asignan en el parámetro CMD.

Con el parámetro CMD_MODIFIER se especifican con más detalle los comandos 04 (Query_Modes) y 16 (Query_Measurement).

El parámetro CMD_PARA asigna los valores de determinados parámetros a algunos comandos mediante un puntero Any. El parámetro CMD_PARA_LEN define la longitud de dichos parámetros.

El parámetro RESPONSE_DATA apunta al área de datos de respuesta del dispositivo IO.

Interfaz de llamada en representación FUP



Consulte también

Explicación de los parámetros formales de PE_CMD_CP (Página 8125)

Explicación de los parámetros formales de PE_CMD_CP

Explicación de los parámetros formales de PE_CMD_CP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CPLADDR	INPUT	WORD	I, Q, M, D, L, const.	Dirección inicial del módulo del CP
REQ	INPUT	BOOL		Con un flanco ascendente comienza la transferencia de los comandos PROFIenergy.
ID	INPUT	WORD		Dirección lógica del dispositivo PROFIenergy de destino
CMD	INPUT	BYTE	• 01: Start_Pause • 02: End_Pause • 03: Query_Modes • 04: PEM_Status • 05: PE_Identity • 16: Query_Measurement	Service_Request_ID del comando PROFIenergy. Debajo de la tabla encontrará el significado de los comandos.

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CMD_ MODIFIER	INPUT	BYTE	Para "Start_Pause": 00 Para "End_Pause": 00 Para "Query_Modes": • 01: List_Energy_Saving _Modes • 02: Get_Mode Para "PEM_Status": 00 Para "PE_Identity": 00 Para "Query_Measurement": • 01: Get_Measurement_List • 02: Get_Measurement_Values	Modificador del comando PROFIenergy, significado: • Comando "Query_Modes" – Modificador 01: lee todos los modos de ahorro de energía soportados (PE_Mode_ID). – Modificador 02: lee los parámetros de la PE_Mode_ID seleccionada. • Comando "Query_Measurement" – Modificador 01: lee las Measurement_IDs configuradas. – Modificador 02: lee los valores medidos de las Measurement_IDs seleccionadas. Encontrará información sobre los parámetros en el apartado del comando correspondiente del capítulo Datos de respuesta. El modificador 00 significa "sin opciones".
CMD_PARA	INPUT	ANY		Puntero Any hacia parámetros para comandos • Para comando 01 Start_Pause: "Pause_Time" • Para comando 02 End_Pause: irrelevante • Para comando 03 Query_Modes: – Para modificador 01: irrelevante – Para modificador 02 Get_Mode: "PE_Mode_ID" • Para comando 04 PEM_Status: irrelevante • Para comando 05 PE_Identity: irrelevante • Para comando 16 Query_Measurement: – Para modificador 01: irrelevante – Para modificador 02 Get_Measurement_Values: Puntero ANY hacia la estructura de datos con parámetros "Count" y "Measurement_IDs" Encontrará información sobre los parámetros en el apartado del comando correspondiente del capítulo Datos de respuesta. Se registra el área de datos completa del registro que debe escribirse (Service_Data_Request). Longitud máx.: 234 bytes
CMD_PARA _LEN	INPUT	INT		Longitud real de los parámetros en CMD_PARA. Longitud máx.: 234 bytes

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
RESPONSE _DATA	INOUT	ANY		Puntero hacia la dirección de los datos de respuesta del dispositivo IO (telegrama completo incl. Block Header) Nota: Si se ha seleccionado un rango demasiado pequeño solo se guardará el número de bytes configurado.
VALID	OUTPUT	BOOL	0: - 1: procesamiento finalizado correctamente	El parámetro de estado del bloque de programa indica si la petición se ha desarrollado correctamente.
BUSY	OUTPUT	BOOL	0: procesamiento todavía no iniciado, finalizado o cancelado 1: procesamiento en curso	Indicador del estado de procesamiento del bloque de programa
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Ver el significado en relación con el parámetro STATUS en Indicadores de PE_CMD_CP.
STATUS	OUTPUT	WORD		Indicador de estado Ver el significado en relación con el parámetro ERROR en Indicadores de PE_CMD_CP.

Service_Request_IDs y significado de los comandos PROFIenergy

Los comandos PROFIenergy con Service_Request_ID 01...05 y 16 tienen el significado siguiente:

- 01 = Start_Pause
  Comando para iniciar una pausa de ahorro de energía.
  El dispositivo IO selecciona el modo de ahorro de energía configurado. El modo de ahorro de energía se notifica al controlador en los datos de respuesta.
- 02 = End_Pause
  Comando para finalizar una pausa de ahorro de energía
- 03 = Query_Modes
  Consulta los modos de ahorro de energía configurados en el dispositivo IO con toda la información correspondiente de tiempo y energía.
  Mediante el parámetro CMD_MODIFIER se detalla la información consultada:
  - List_Energy_Saving_Modes
    Lee todos los modos PROFIenergy soportados por el dispositivo IO.
  - Get_Mode
    Lee los datos del modo PROFIenergy seleccionado.
- 04 = PEM_Status
  Consulta del modo de ahorro de energía adoptado realmente por el aparato de campo o por el grupo.

- **05 = PE_Identity**
  Consulta los servicios PROFIenergy soportados por el dispositivo IO.
- **16 = Query_Measurement**
  Consulta datos energéticos del dispositivo IO.
  Mediante el parámetro CMD_MODIFIER se detalla la información consultada:
  - **Get_Measurement_List**
    Lee todas las Measurement_IDs configuradas en el dispositivo IO.
  - **Get_Measurement_Values**
    Lee los valores energéticos medidos de las Measurement_IDs seleccionadas.

Comandos para distintas clases de aparatos

Los aparatos que pueden integrarse en conceptos PROFIenergy pueden dividirse en tres clases, que son activadas por el controlador IO utilizando distintos comandos:

- Módulos E/S, actuadores, arrancadores de motor
  Comandos soportados:
  - Start_Pause, End_Pause
  - Query_Modes, PEM_Status, PE_Identify
- Instrumentos de medida para magnitudes eléctricas
  Comandos soportados:
  - Query_Measurement
- Convertidores de frecuencia
  Comandos soportados:
  - Start_Pause, End_Pause
  - Query_Modes, PEM_Status, PE_Identify
  - Query_Measurement
    También es posible consultar datos de magnitudes eléctricas que registran los convertidores de frecuencia.

Indicadores de PE_CMD_CP

Indicadores de PE_CMD_CP

PE_CMD_CP se basa en el bloque de programa PNIO_RW_REC y proporciona todos los indicadores de PNIO_RW_REC; consulte los indicadores del bloque PNIO_RW_REC.

Además, se muestran los siguientes indicadores específicos de PROFIenergy. Los indicadores de error de STATUS solo son válidos en relación con ERROR = 1.

Tabla 9-341 Indicadores específicos de PE_CMD_CP

STATUS	Significado
Errores específicos del bloque	
8081_h	Conflicto de longitud con CMD_PARAM y CMD_PARAM_LEN

STATUS	Significado
Errores específicos de PROFIenergy	
$FE01_h$	Service_Request_ID no válida
$FE02_h$	Request_Reference no válida
$FE03_h$	CMD_MODIFIER no válido
$FE04_h$	Indicación no válida sobre la estructura de datos de un comando (Data_Structure_Identifier_RQ) en el telegrama para el registro PROFIenergy de escritura
$FE05_h$	Indicación no válida sobre la estructura de datos de un comando (Data_Structure_Identifier_RS) en el telegrama para el registro PROFIenergy de lectura
$FE06_h$	No se soporta el modo de ahorro de energía (PE_Mode_ID)
$FE07_h$	La respuesta es más larga que la longitud de transferencia máx.
$FE08_h$	Número de comandos no válido
$FE09_h$	Block Type no válido (consulte el encabezado del telegrama)
$FE0A_h$	Block Length no válida (consulte el encabezado del telegrama)
$FE0B_h$	Block Version no válida (consulte el encabezado del telegrama)
$FE50_h$	No hay ningún modo de ahorro de energía adecuado (PE_Mode_ID)
$FE51_h$	No se soporta el valor de PAUSE_TIME
$FE52_h$	No se soporta PE_Mode_ID

Consulte los detalles relacionados con los parámetros de los errores específicos de PROFIenergy en el capítulo Datos de respuesta.

Datos de respuesta

Estructura de los datos de respuesta

Las tablas siguientes muestran la estructura del registro ($80A0_h$) de los datos de respuesta de PE_START_END_CP y PE_CMD_CP.

La tabla siguiente muestra una sinopsis de la estructura del registro de los datos de respuesta devueltos de acuerdo con la especificación PROFIenergy. La composición del área "Service Data Response" se describe a continuación para los diferentes comandos PROFIenergy.

Tabla 9-342 Estructura de los datos de respuesta

Definiciones de bloque	Atributos	Valor	Tipo de datos	Descripción
Block Header	BlockType	0801_h	Unsigned16	
	BlockLength		Unsigned16	Longitud de telegrama (sin los campos "BlockType" y "BlockLength")
	BlockVersionHigh	01_h	Unsigned8	
	BlockVersionLow	00_h	Unsigned8	

Definiciones de bloque	Atributos	Valor	Tipo de datos	Descripción
Response Header	Service_Request_ID	01_h...FF_h	Unsigned8	01_h: Start_Pause 02_h: End_Pause 03_h: Query_Modes 04_h: PEM_Status 05_h: PE_Identify 06_h...09_h: - reservado - 10_h: Query_Measurement 11_h...CF: - reservado - $D0_h$...FF_h: específico del fabricante
	Request_Reference	01_h...FF_h	Unsigned8	Número de identificación de la consulta (reflejado en la respuesta del dispositivo IO)
Service Header Response	Estado	01_h...FF_h	Unsigned8	00_h: - reservado - 01_h: listo 02_h: listo con error 03_h: datos incompletos 04_h...CF_h: - reservado - $D0_h$...FF_h: en función de la Service_Request_ID
	Data_Structure_Identifier_RS	01_h...FF_h	Unsigned8	00_h: - reservado - 01_h...FF_h: estructura de datos en función de la Service_Request_ID FF_h: error
Service Data Response				Datos de respuesta del dispositivo IO En función del comando PROFIenergy correspondiente (descrito a continuación)

Significado de "Service Data Request" y "Service Data Response"

Los apartados siguientes explican los valores de parámetros para los diferentes comandos PROFIenergy en las consultas del controlador IO al dispositivo IO (Service Data Request) y la estructura de los datos de respuesta del dispositivo IO (Service Data Response).

- Service Data Request
  Valores de parámetros en la consulta del controlador IO
- Service Data Response
  Estructura de los datos de respuesta del dispositivo IO

Comando PROFIenergy "Start_Pause"

- Service Data Request
  - CMD = 01
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 04
  - CMD_PARA = puntero Any hacia el valor de "Pause_Time" (tipo de datos "TIME") Temporizador IEC en incrementos de 1 ms, entero con signo
    Valor: T#-24D_20H_31M_23S_648MS a T#24D_20H_31M_23S_647MS
- Service Data Response

Parámetro	Valor	Tipo de datos
PE_Mode_ID *	01_h...FF_h	Unsigned8
- reservado -	00_h	Unsigned8

* Número de identificación del modo de ahorro de energía

Comando PROFIenergy "End_Pause"

- Service Data Request
  - CMD = 02
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevante
- Service Data Response

Parámetro	Valor	Tipo de datos
Time_to_operate *		Unsigned32

* Tiempo previsto para conmutar el dispositivo PROFIenergy a "operativo" (ready_to_operate)

Comando PROFIenergy "Query_Modes" – List_Energy_Saving_Modes

- Service Data Request
  - CMD = 03
  - CMD_MODIFIER = 01
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevante
- Service Data Response

Parámetro	Valor	Tipo de datos
Number_of_PE_Mode_IDs *	01_h	Unsigned8
PE_Mode_IDs		Unsigned8-Array of Number_of_PE_Mode_IDs (Unique ID for mode)

* Número de modos de ahorro de energía

Comando PROFIenergy "Query_Modes" – Get_Mode

- Service Data Request
  - CMD = 03
  - CMD_MODIFIER = 02
  - CMD_PARA_LEN = 01
  - CMD_PARA = puntero Any hacia el valor de PE_MODE_ID (unsigned8)
- Service Data Response

Parámetro	Valor	Tipo de datos
PE_Mode_ID	01_h...FF_h	Unsigned8
PE_Mode_Attributes *	00_h...01_h	Unsigned8
Time_min_Pause		Unsigned32
Time_to_Pause		Unsigned32
Time_to_operate		Unsigned32
Time_min_length_of_stay		Unsigned32
Time_max_length_of_stay		Unsigned32
Mode_Power_Consumption		Float32
Energy_Consumption_to_pause		Float32
Energy_Consumption_to_operate		Float32

* Codificación del bit 0:
0 = solo están disponibles valores medidos de tiempo y energía estáticos.
1 = están disponibles valores medidos de tiempo y energía dinámicos.
Bits 1...7: reservados

Comando PROFIenergy "PEM_Status"

- Service Data Request
  - CMD = 04
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevante
- Service Data Response

Parámetro	Valor	Tipo de datos
PE_Mode_ID_Source *	00_h 01_h...FE_h FF_h	Unsigned8
PE_Mode_ID_Destination *	00_h 01_h...FE_h FF_h	Unsigned8
Time_to_operate		Unsigned32
Remaining_time_to_destination		Unsigned32
Mode_Power_Consumption		Float32
Energy_Consumption_to_Destination		Float32
Energy_Consumption_to_operate		Float32

* Valores posibles de "PE_Mode_ID_Source" y "PE_Mode_ID_Destination":
00_h: PE_Power_off (fuente de alimentación desconectada)
01_h...FE_h: libremente configurable
FF_h: PE_Ready_to_operate (operativo)

Comando PROFIenergy "PE_Identify"

- Service Data Request
  - CMD = 05
  - CMD_MODIFIER = 00
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevante
- Service Data Response

Parámetro	Valor	Tipo de datos
Count *	6	Unsigned8
Start_Pause **	01_h	Unsigned8
End_Pause	02_h	Unsigned8
Query_Modes	03_h	Unsigned8
PEM_Status	04_h	Unsigned8
PE_Identify	05_h	Unsigned8
Query_Measurement ***	10_h	Unsigned8

* Número de comandos PROFIenergy soportados

** Service_Request_ID del primer comando PROFIenergy soportado

*** Service_Request_ID del último comando PROFIenergy soportado

Comando PROFIenergy "Query_Measurement" – Get_Measurement_List

- Service Data Request
  - CMD = 16
  - CMD_MODIFIER = 01
  - CMD_PARA_LEN = 00
  - CMD_PARA = irrelevante
- Service Data Response

Parámetro	Valor	Tipo de datos
Count *		Unsigned8
- reservado -		Unsigned8
Measurement_ID **		Unsigned16
Accuracy_Domain [1]		Unsigned8
Accuracy_Class [2]		Unsigned8
Range [3]		Float32
…		
Measurement_ID ***		Unsigned16
Accuracy_Domain [1]		Unsigned8
Accuracy_Class [2]		Unsigned8
Range [3]		Float32

* Número de Measurement_IDs

** Primera Measurement_ID soportada

*** Última Measurement_ID soportada

[1] Dominio de precisión (rango 1...4):
0 = reservado
1 = porcentaje del rango de medición
2 = porcentaje del valor medido actual
3 = precisión conforme a IEC 61557-12
4 = precisión conforme a EN 50470-3 capítulo 8

[2] Clase de precisión (rango 1...15):
0 = reservado
1 (0,01%) ... 15 (>20%)

[3] Rango de medición si Accuracy_Domain = 1; de lo contrario, indefinido

Comando PROFIenergy "Query_Measurement" – Get_Measurement_Values

- Service Data Request
  - CMD = 16
  - CMD_MODIFIER = 02
  - CMD_PARA_LEN = longitud de la estructura de datos en bytes
  - CMD_PARA = puntero Any hacia una estructura de datos con la forma siguiente:

Parámetro	Valor	Tipo de datos
Count *		Unsigned8
- reservado -		Unsigned8
Measurement_ID **		Unsigned16
…		
Measurement_ID ***		Unsigned16

* Número de Measurement_IDs
** Primer valor medido consultado
*** Último valor medido consultado

- Service Data Response

Parámetro	Valor	Tipo de datos
Count *		Unsigned8
- reservado -		Unsigned8
Length_of_Structure	0002_h...$FFFF_h$	Unsigned16
Measurement_Data_Structure_ID	1 = valor simple	Unsigned8
Measurement_ID **	00_h...FF_h	Unsigned16
Status_of_Measurement_Value	1 = válido 2 = no disponible 3 = no disponible temporalmente	Unsigned8
Transmission_Data_Type		Float32
End_of_demand		Unsigned32 o Unsigned16
Length_of_Structure		Unsigned16
Measurement_Data_Structure_ID		Unsigned8
Measurement_ID ***		Unsigned16
Status_of_Measurement_Value		Unsigned8
Transmission_Data_Type		Float32
End_of_demand		Unsigned32 o Unsigned16

* Número de Measurement_IDs
** Primer valor medido consultado
*** Último valor medido consultado

Consulte también

Explicación de los parámetros formales de PE_START_END_CP (Página 8121)

PE_I_DEV_CP

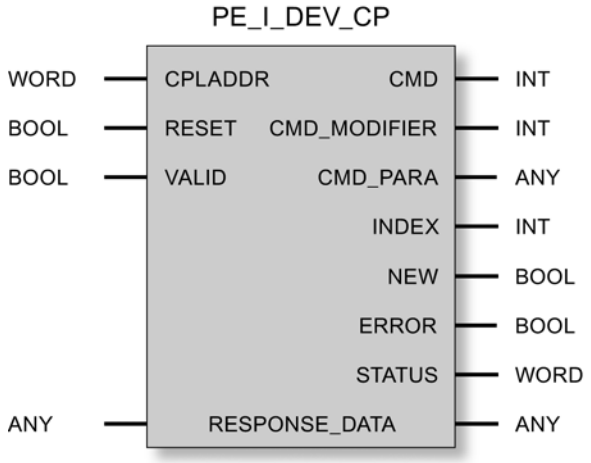
Significado y llamada - PE_I_DEV_CP

Significado y funcionamiento

El bloque de programa PE_I_DEV_CP se utiliza en el dispositivo PROFIenergy, donde ejecuta los comandos PROFIenergy del controlador IO. Los registros enviados por el controlador IO ($80A0_h$) son transmitidos a PE_I_DEV_CP por el firmware del CP. Los datos PROFIenergy del dispositivo IO, a su vez, son puestos a disposición del controlador IO en forma de respuesta por parte de PE_I_DEV_CP mediante el registro PROFIenergy ($80A0_h$).

Los datos de respuesta de PE_I_DEV_CP se generan mediante las funciones complementarias FC 0 a FC 8, consulte el capítulo Bloques de programa complementarios para PE_I_DEV_CP.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales de PE_I_DEV_CP

Explicación de los parámetros formales de PE_I_DEV_CP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CPLADDR	INPUT	WORD	I, Q, M, D, L, const.	Dirección inicial del módulo del CP
RESET	INPUT	BOOL		Resetea el procesamiento del bloque de programa. NEW se pone a 0.
VALID	INPUT	BOOL		Cuando se han escrito los datos de respuesta en el área de memoria correspondiente del dispositivo IO, el programa de usuario debe poner VALID = 1. Seguidamente, el bloque de programa pone los datos a disposición del controlador IO. NEW se pone a 0.
CMD	OUTPUT	INT	• 01: Start_Pause • 02: End_Pause • 03: Query_Modes • 04: PEM_Status • 05: PE_Identity • 16: Query_Measurement	ID de servicio del comando PROFIenergy
CMD_ MODIFIER	OUTPUT	INT	• Modificador para Start_Pause: 00 • Modificador para End_Pause: 00 • Query_Modes, modificador: – 01 (List_Energy_ Saving_Modes) – 02 (Get_Mode) • Modificador para PEM_Status: 00 • Modificador para PE_Identity: 00 • Query_Measurement, modificador: – 01 (Get_Measurement _List) – 02 (Get_Measurement _Values)	Modificador de los comandos PROFIenergy Significado de los modificadores para comandos: • Comando "Query_Modes", modificador: – 01 (List_Energy_Saving_Modes): lee todos los modos PROFIenergy soportados – 02 (Get_Mode): lee los datos del modo PROFIenergy seleccionado • Comando "Query_Measurement", modificador: – 01 (Get_Measurement_List): lee todas las Measurement_IDs configuradas. – 02 (Get_Measurement_Values): lee los valores medidos de las Measurement_IDs seleccionadas.

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CMD_PARA	OUTPUT	ANY		Puntero Any hacia los parámetros para los siguientes modificadores de comandos (consulte el parámetro CMD_MODIFIER): • Para "Get_Mode": PE_Mode_ID (ID del modo de ahorro de energía)longitud = 1 • Para "Get_Measurement_Values":valores medidos de las Measurement_IDs longitud = máx. 236 bytes (telegrama completo del comando del controlador sin encabezado)
INDEX	OUTPUT	INT		Número del registro PROFIenergy ($80A0_h$)
NEW	OUTPUT	BOOL	0: procesamiento todavía no iniciado, finalizado o cancelado 1: procesamiento en curso	Indicadores del estado de procesamiento del bloque de programa
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error Ver el significado en relación con el parámetro STATUS en Indicadores de PE_I_DEV_CP (Página 8138).
STATUS	OUTPUT	WORD		Indicador de estado Ver el significado en relación con el parámetro ERROR en Indicadores de PE_I_DEV_CP (Página 8138).
RESPONSE _DATA	INOUT	ANY	Consulte los "datos de respuesta" del bloque de programa	Puntero hacia el área de datos de respuesta del dispositivo IO (telegrama de respuesta completo incluido encabezado). El área de datos debe coincidir con la de los bloques de programa complementarios FC 0 - FC 8 (parámetro"DATA_ERRORRSP"). Tamaño recomendado: mín. 244 bytes. Si el área de datos es demasiado pequeña solo se transferirán los datos de los bytes configurados.

Indicadores de PE_I_DEV_CP

Indicadores de PE_I_DEV_CP

PE_I_DEV_CP se basa en el bloque de programa PNIO_RW_REC y proporciona todos los indicadores de PNIO_RW_REC; consulte los indicadores del bloque PNIO_RW_REC.

Bloques de programa complementarios para PE_I_DEV_CP

Vista general de los FCs

Función

Los bloques de programa complementarios de FC 0 a FC 8 soportan la preparación de datos de respuesta que PE_I_DEV_CP pone a disposición del controlador:

- Hay una función separada para los datos de respuesta de cada bloque de programa PROFIenergy (FC 1 - FC 8).
- FC 0 genera una respuesta negativa común para todos los comandos PROFIenergy.

Los FCs se llaman en el programa de usuario. En STEP 7 V5.5 se proporcionarán en la librería estándar, en la carpeta "PROFIenergy".

Los FCs tienen algunos parámetros comunes y otros individuales para cada una. Algunos de los parámetros comunes de los FCs se interconectan con parámetros de PE_I_DEV_CP. En algunos de los parámetros de entrada individuales de los FCs, el usuario introduce los datos de respuesta en forma de texto claro o los almacena en el área de memoria del dispositivo IO.

Vista general de los FCs

Se proporcionarán los siguientes bloques de programa complementarios:

Tabla 9-343 Vista general de los FCs complementarios

Número	Nombre
FC 0	PE_ERROR_RSP
FC 1	PE_START_RSP
FC 2	PE_END_RSP
FC 3	PE_LIST_MODES_RSP
FC 4	PE_GET_MODE_RSP
FC 5	PE_PEM_STATUS_RSP
FC 6	PE_IDENTIFY_RSP
FC 7	PE_MEASUREMENT_LIST_RSP
FC 8	PE_MEASUREMENT_VALUE_RSP

Interconexión de los FCs con PE_I_DEV_CP

Interconexión de los FCs con el bloque de programa PE_I_DEV_CP

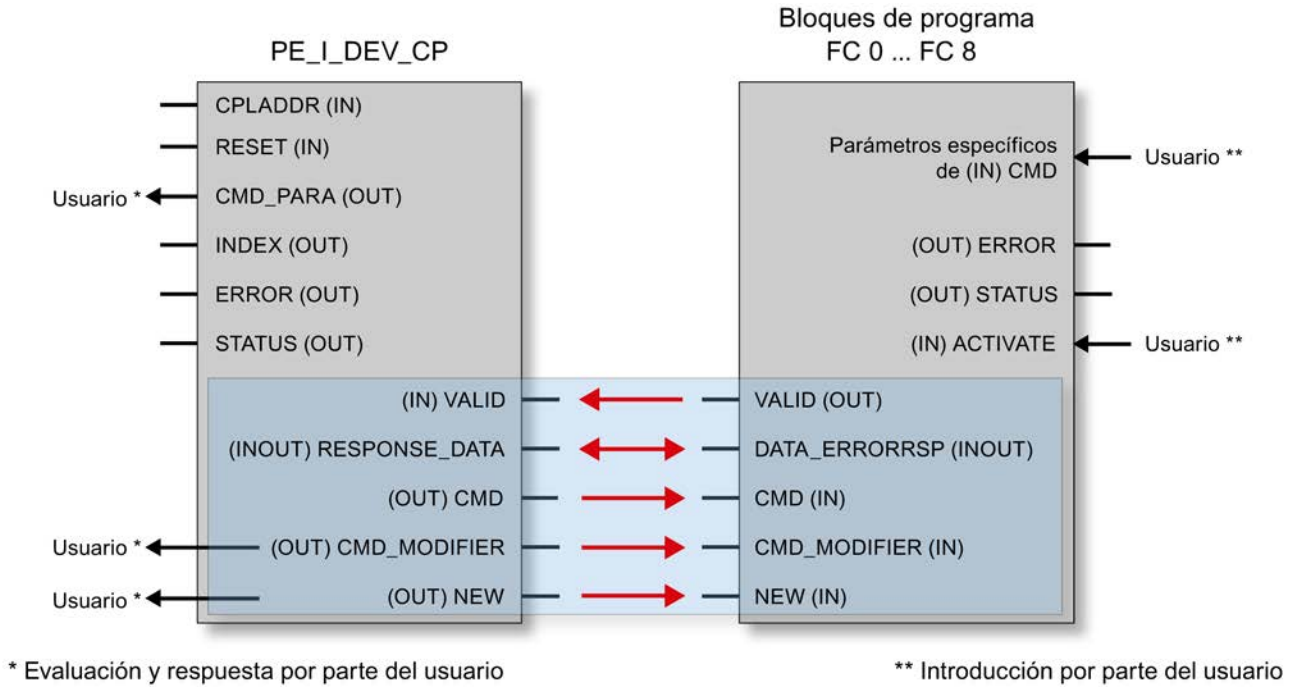


Figura 9-21 Interconexión de los FCs con PE_I_DEV_CP

Nota

La interconexión de los bloques de programa es obligatoria

PE_I_DEV_CP debe interconectarse con FC 0...FC 8 en los parámetros con fondo azul claro que están asignados a los parámetros correspondientes de los FCs mediante flechas rojas.

Parámetros comunes de los FCs

Parámetros comunes de los bloques de programa complementarios FC 0 - FC 8

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Parámetros de entrada				
ACTIVATE	INPUT	BOOL		Hace que el bloque copie los parámetros de entrada en el área de datos "DATA_ERRORRSP" cuando hay un flanco ascendente. Seguidamente, el bloque lo resetea. El usuario debe activarlo en un plazo de 10 segundos después de haberse detectado un flanco ascendente en PE_I_DEV_NEW.
PE_I_DEV_NEW	INPUT	BOOL		Debe interconectarse con el parámetro de salida NEW de PE_I_DEV_CP. El bloque solo se procesa cuando hay un 1.
CMD	INPUT	INT		Debe interconectarse con el parámetro de salida CMD de PE_I_DEV_CP.
CMD_MODIFIER	INPUT	INT		Debe interconectarse con el parámetro de salida CMD_MODIFIER de PE_I_DEV_CP.
Parámetros de salida				
DATA_ERRORRSP	OUTPUT	ANY		Puntero hacia el área de datos en la que se almacenan los datos de respuesta (telegrama de respuesta completo incluido encabezado). Debe interconectarse con el parámetro de salida RESPONSE_DATA de PE_I_DEV_CP. Tamaño recomendado: mín. 244 bytes.
VALID	OUTPUT	BOOL	0: - 1: ningún error	Es activado por el bloque. Debe interconectarse con el parámetro de entrada VALID de PE_I_DEV_CP.
ERROR	OUTPUT	BOOL	0: ningún error 1: error	Indicador de error
STATUS	OUTPUT	WORD	0: ningún error	Indicador de estado $80B1_h$: error en la indicación de ANY (p. ej. área errónea)

Parámetros individuales de los FCs

Parámetros individuales de FC 0 a FC 8

A continuación se describen los parámetros individuales de los FCs.

PE_ERROR_RSP

Genera una respuesta negativa si el comando PROFIenergy solicitante no se soporta en general o temporalmente. La respuesta negativa es independiente del comando solicitante.

Tabla 9-344 Parámetros individuales de FC 0 PE_ERROR_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ERROR_CODE	INPUT	BYTE		Número de error

PE_START_RSP

Inicia una pausa de ahorro de energía. Genera la respuesta al comando "Start_Pause". Devuelve el modo de ahorro de energía que adopta el dispositivo.

Tabla 9-345 Parámetros individuales de FC 1 PE_START_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
PE_Mode_ID	INPUT	BYTE		ID del modo de ahorro de energía que adopta el dispositivo o grupo.

Notificación de la PE_Mode_ID del modo de ahorro de energía que ha adoptado el aparato de campo o el grupo.

PE_END_RSP

Genera la respuesta al comando "End_Pause".

Tabla 9-346 Parámetros individuales de FC 2 PE_END_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Time_to_Operate	INPUT	DWORD		Tiempo necesario para cambiar del modo de ahorro de energía actual a "operativo" (Ready_to_Operate).

PE_LIST_MODES_RSP

Genera la respuesta al comando Query_Modes" > modificador "List_Modes" (lista de los modos de ahorro de energía soportados).

Las IDs de los modos de ahorro de energía deben especificarse en el programa de usuario.

Tabla 9-347 Parámetros individuales de FC 3 PE_LIST_MODES_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Number_of_PE_Mode_IDs	INPUT	BYTE		Número de modos de ahorro de energía soportados
PE_Mode_ID	INPUT	ANY	• 00_h • 01_h...FE_h • FF_h	Puntero hacia el área en el que están almacenados los modos de ahorro de energía. El usuario debe almacenar aquí las IDs de los modos de ahorro de energía. La ID de un modo de ahorro de energía se configura en formato Unsigned8. Rango admisible: de 1 a 254 bytes.

Si los aparatos de un grupo reaccionan de forma distinta en pausas de diferente longitud, es posible establecer diferentes modos de ahorro de energía (PE_Mode). Los diferentes modos de ahorro de energía reciben una PE_Mode_ID distinta.

Valore posibles para "PE_Mode_ID":

- 00_h: PE_Power_off (fuente de alimentación desconectada)
- 01_h...FE_h: libremente configurable o específico del fabricante
- FF_h: PE_Ready_to_operate (operativo)

PE_GET_MODE_RSP

Genera la respuesta al comando "Query_Modes" > Modifier "Get_Mode".

Tabla 9-348 Parámetros individuales de FC 4 PE_GET_MODE_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
PE_Mode_ID	INPUT	BYTE		ID de modo de ahorro de energía que se utiliza actualmente
Time_Min_Pause *	INPUT	Unsigned32		Duración mín. de pausa para este modo de ahorro de energía PE. Es la suma de los tres parámetros: • Time_to_Pause • Time_to_operate • Time_min_length_of_stay
Time_to_Pause *	INPUT	Unsigned32		Duración desde el flanco START hasta alcanzar el modo de ahorro de energía solicitado
Time_to_operate *	INPUT	Unsigned32		Tiempo de conexión máx. hasta PE_ready_to_operate Time_to_operate puede utilizarse directamente para los cálculos correspondientes. El valor puede ser bien un valor MAX estático bien ser calculado por el dispositivo PE.
Time_min_length_of_stay *	INPUT	Unsigned32		Duración mínima de permanencia del dispositivo PE en este PE_Mode.

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Time_max_length_of_stay *	INPUT	Unsigned32		Duración máxima de permanencia del dispositivo PE en este PE_Mode.
Mode_Power _Consumption **	INPUT	Float32		Consumo de energía en el PE_Mode actual [kW]
Energy_Consumption _to_pause **	INPUT	Float32		Consumo de energía de PE_ready_to_operate hasta el PE_Mode actual [kWh]
Energy_Consumption _to_operate **	INPUT	Float32		Consumo de energía del PE_Mode actual hasta PE_ready_to_operate [kWh]

* El perfil PROFIenergy no especifica ningún formato de tiempo no válido.
Si la duración es ilimitada es posible indicar el valor máx. $FFFFFFFF_h$.
Si la duración es cero es posible utilizar 00_h.

** Si no se ha definido ningún valor de consumo de energía es posible indicar 0,0 (Float32).

PE_PEM_STATUS_RSP

Genera la respuesta al comando "PEM_STATUS".

Tabla 9-349 Parámetros individuales de FC 5 PE_PEM_STATUS_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
PE_Mode_ID_Source	INPUT	BYTE	• 00_h • 01_h...FE_h • FF_h	ID del modo de ahorro de energía que se ha adoptado realmente
PE_Mode_ID_Destination	INPUT	BYTE	• 00_h • 01_h...FE_h • FF_h	ID del modo de ahorro de energía especificado por el controlador
Time_to_operate *	INPUT	Unsigned32		Tiempo de conexión máx. hasta PE_ready_to_operate Time_to_operate puede utilizarse directamente para los cálculos correspondientes. El valor puede ser bien un valor MAX estático bien ser calculado por el dispositivo PE.
Remaining_time_to _destination *	INPUT	Unsigned32		Opcional: tiempo restante hasta el PE_Mode solicitado. Valor dinámico o valor MAX estático
Mode_Power _Consumption **	INPUT	Float32		Consumo de energía en el PE_Mode actual [kW]
Energy_Consumption _to_Destination **	INPUT	Float32		Consumo de energía hasta el PE_Mode solicitado [kWh]
Energy_Consumption _to_operate **	INPUT	Float32		Consumo de energía del PE_Mode actual hasta PE_ready_to_operate [kWh]

* El perfil PROFIenergy no especifica ningún formato de tiempo no válido.
Si la duración es ilimitada es posible indicar el valor máx. $FFFFFFFF_h$.
Si la duración es cero es posible utilizar 00_h.

** Si no se ha definido ningún valor de consumo de energía es posible indicar 0,0 (Float32).

Valore posibles para "PE_Mode_ID_Source" y "PE_Mode_ID_Destination":

- 00_h: PE_Power_off (fuente de alimentación desconectada)
- 01_h...FE_h: libremente configurable o específico del fabricante
- FF_h: PE_Ready_to_operate (operativo)

PE_IDENTIFY_RSP

Genera la respuesta al comando "PE_Identify".

El usuario debe indicar los comandos PROFIenergy que se soportan.

Tabla 9-350 Parámetros individuales de FC 6 PE_IDENTIFY_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Count	INPUT	BYTE	0...6	Número de comandos PROFIenergy soportados
Start_Pause	INPUT	BOOL	0...1	• 1: el comando se soporta • 0: el comando no se soporta
End_Pause	INPUT	BOOL	0...1	• 1: el comando se soporta • 0: el comando no se soporta
Query_Modes	INPUT	BOOL	0...1	• 1: el comando se soporta • 0: el comando no se soporta
PEM_Status	INPUT	BOOL	0...1	• 1: el comando se soporta • 0: el comando no se soporta
PEM_Identify	INPUT	BOOL	0...1	• 1: el comando se soporta • 0: el comando no se soporta
Query_Measurement	INPUT	BOOL	0...1	• 1: el comando se soporta • 0: el comando no se soporta

PE_MEASUREMENT_LIST_RSP

Genera la respuesta al comando "Query_Measurement" > Modifier "Get_Measurement_List".

Tabla 9-351 Parámetros individuales de FC 7 PE_MEASUREMENT_LIST_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Count	INPUT	BYTE		Número de IDs de valor medido soportadas (Measurement_ID)
Measurement_List	INPUT	ANY		Puntero hacia el área de datos con las IDs de valor medido soportadas. El usuario debe almacenar las IDs de valor medido en esta área de datos. Es posible transferir un máximo de 29 IDs de valor medido por telegrama. Consulte la estructura de la matriz en el capítulo Datos de respuesta > "Query_Measurement" – Get_Measurement_List.

PE_MEASUREMENT_VALUE_RSP

Genera la respuesta al comando "Query_Measurement" > Modifier "Get_Measurement_Values".

Tabla 9-352 Parámetros individuales de FC 8 PE_MEASUREMENT_VALUE_RSP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
Count	INPUT	BYTE		Número de Measurement_Values soportados
Measurement_Values	INPUT	ANY		Puntero hacia el área de datos de los valores medidos (Measurement_Values). El usuario debe almacenar los valores medidos en esta área de datos. Es posible transferir un máximo de 116 valores medidos por telegrama. Consulte la estructura de la matriz en el capítulo Datos de respuesta > "Query_Measurement" – Get_Measurement_List.

PE_DS3_WRITE_ET200S_CP

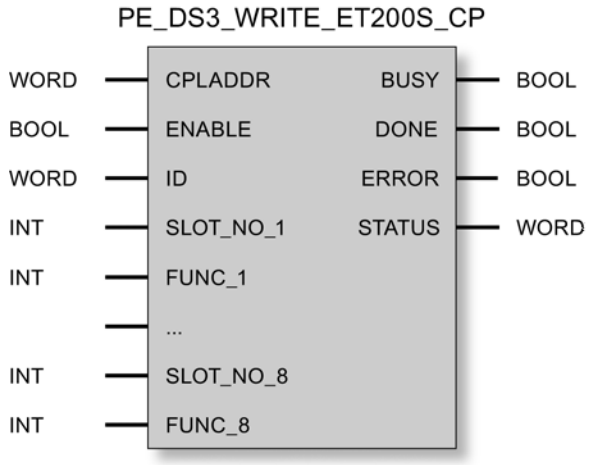
Significado y llamada - PE_DS3_WRITE_ET200_CP

Significado y funcionamiento

En la CPU del CP 300 funcionando como controlador PROFIenergy se utiliza PE_DS3_WRITE_ET200_CP para transferir ajustes del comportamiento de conmutación de los módulos de potencia a un ET 200S. Es posible transferir el comportamiento de conmutación para un máximo de 8 slots (en este caso: módulos de potencia).

PE_DS3_WRITE_ET200_CP no es un bloque de programa PROFIenergy.

Interfaz de llamada en representación FUP



Explicación de los parámetros formales de PE_DS3_WRITE_ET200S_CP

Explicación de los parámetros formales de PE_DS3_WRITE_ET200_CP

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
CPLADDR	INPUT	WORD	I, Q, M, D, L, const.	Dirección inicial del módulo del CP
ENABLE	INPUT	BOOL		Con flanco ascendente comienza el procesamiento del bloque de programa.
ID	INPUT	WORD		Dirección lógica del slot de encabezado del dispositivo IO
SLOT_NO_1	INPUT	INT		Número de slot del primer módulo de potencia
FUNC_1	INPUT	INT		Definición del comportamiento de conmutación para el módulo de potencia en relación al inicio y fin de las pausas de ahorro de energía • 0 (FALSE) – PAUSE_START: no influye (el módulo de potencia permanece conectado) – PAUSE_STOP: vuelve a conectar el módulo de potencia. • 1 (TRUE) – PAUSE_START: desconecta el módulo de potencia. – PAUSE_STOP: vuelve a conectar el módulo de potencia.

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
...	INPUT	INT		
...	INPUT	INT		
SLOT_NO_8	INPUT	INT		Número de slot del octavo módulo de potencia
FUNC_8	INPUT	INT		Consulte "FUNC_1"
BUSY	OUTPUT	BOOL	0: procesamiento todavía no iniciado, finalizado o cancelado 1: procesamiento en curso	Indicador del estado de procesamiento del bloque de programa
DONE	OUTPUT	BOOL	0: - 1: registro transferido correctamente	El parámetro de estado indica si la petición se ha desarrollado correctamente.
ERROR	OUTPUT	BOOL	0: - 1: error	Indicador de error
STATUS	OUTPUT	WORD		Indicador de estado

Encontrará el significado de DONE, ERROR y STATUS en Indicadores de PE_DS3_WRITE_ET200S_CP (Página 8148).

Indicadores de PE_DS3_WRITE_ET200S_CP

Indicadores de PE_DS3_WRITE_ET200_CP

PE_DS3_WRITE_ET200_CP se basa en el bloque de programa PNIO_RW_REC y proporciona todos los indicadores de PNIO_RW_REC, consulte los indicadores del bloque PNIO_RW_REC.

Instrucciones para enlaces programados

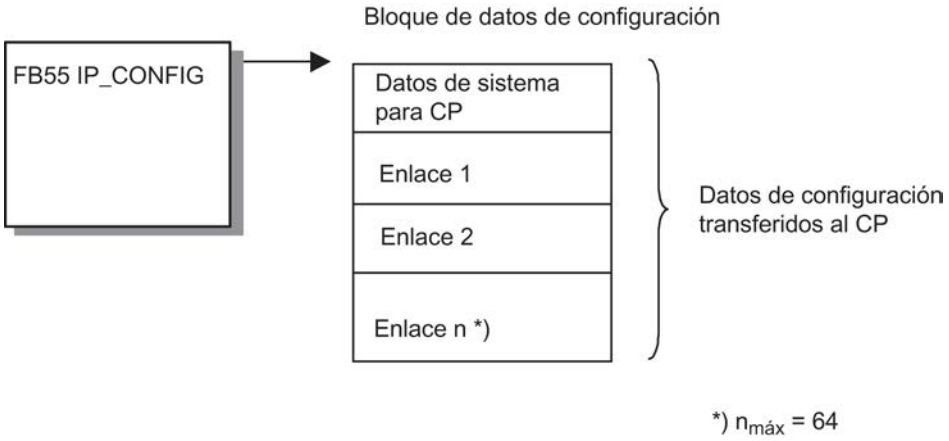
Sinopsis

Relación entre programación y configuración

Los enlaces en la interfaz SEND/RECEIVE, así como la configuración IP de un CP se configuran con STEP 7 o bien mediante el programa de usuario durante el funcionamiento del equipo S7. ¡En un CP no puede utilizarse una combinación de ambas variantes!

Principio de la configuración programada

Con una instrucción llamada en el programa de usuario se pueden transferir al CP datos de configuración para enlaces de comunicación y la configuración IP (datos de sistema para CP).



El DB de configuración puede cargarse en el CP en cualquier momento. Se sobrescriben los enlaces y los datos de configuración (dirección IP, máscara de subred, router predeterminado, servidor de reloj NTP y otros parámetros) que antes eran actuales.

A partir de los datos de configuración, el CP Ethernet reconoce que los enlaces de comunicación deben configurarse a través del programa de usuario.

Nota

Las funciones sólo pueden ejecutarse si el nivel de protección de los módulos ha sido configurado como "No bloqueado": Véase el cuadro de diálogo de propiedades del CP, ficha "Opciones" (no con todos los CPs)

Además debe estar activada la opción "Crear dirección IP en el programa de usuario" (véase el cuadro de propiedades del CP o de la interfaz Ethernet del CP, ficha "Configuración IP")

A partir de los datos de configuración, el CP Ethernet reconoce que los enlaces de comunicación deben configurarse a través del programa de usuario.

Nota

En cuanto el programa de usuario transfiere los datos de enlace a través de la instrucción IP_CONFIG, la CPU pone el CP brevemente en modo STOP. El CP adopta los datos de sistema (incluida la dirección IP) y los nuevos datos de enlace y los procesa en el arranque (RUN).

Datos cuantitativos

En la instrucción IP_CONFIG pueden indicarse como máximo 64 enlaces. Sin embargo, el factor determinante es el número máximo de enlaces soportados por el tipo de CP utilizado.

Particularidades/limitaciones

- Comprobación de coherencia solo en la configuración
  La configuración de enlaces está vinculada a comprobaciones de coherencia, que para la configuración programada no son posibles, o lo son de forma limitada.
- Configuración de enlace necesaria en el interlocutor
  En la configuración de enlaces especificados se crea implícitamente el enlace para el interlocutor, pero en la configuración programada esto no es posible. En este caso deben configurarse los enlaces correspondientes para los interlocutores.
- Configuración del de protección IP
  A través del nivel de protección IP existe la posibilidad de limitar la comunicación a través del CP del equipo S7 local a interlocutores con direcciones IP específicas. Esta parametrización también es aplicable para enlaces de comunicación programadas. Debe desactivar por completo (ajuste predeterminado) la protección de acceso IP en la configuración o autorizar los interlocutores.
- DHCP/DNS soportado
  El direccionamiento IP es posible en la configuración programada también a través de DHCP (y DNS para el servicio de correo electrónico)
  En este caso, el uso de un servidor DHCP se define en la instrucción IP_CONFIG (no en la configuración).
- Ninguna información de enlace durante la carga
  Al cargar los datos de equipo S7 en la configuración no están contenidos los datos de la configuración programada.
- Configuración de enlaces en CPs con varias interfaces
  En CPs con varias interfaces (p. ej. con interfaz Gigabit), asegúrese, consultando el manual de producto del CP, de que éste soporta la configuración de enlaces para ambas interfaces.
- PROFINET IO no es posible simultáneamente
  Para dispositivos que deban funcionar como controlador PROFINET IO o dispositivo IO no es posible la configuración de enlaces a través del FB55 aquí descrito.
- No utilizar la instrucción IP_CONFIG para uso del CP con enlaces S7 altamente disponibles
  Si configura enlaces S7 altamente disponibles a través del CP, no debe utilizar la instrucción IP_CONFIG para la configuración IP del CP.

Consulte también

Bloque de datos de configuración (Página 8156)

IP_CONFIG para enlaces de comunicación programados (Página 8151)

IP_CONFIG

IP_CONFIG para enlaces de comunicación programados

Descripción

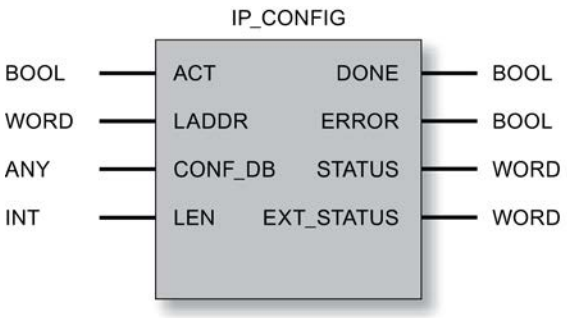
La instrucción IP_CONFIG transmite al CP los datos de conexión especificados en un bloque de datos (DB de configuración). El DB de configuración contiene todos los datos para configurar las conexiones para la interfaz SEND/RECEIVE de un CP Ethernet.

Esta variante de las conexiones de comunicación programadas se puede utilizar como alternativa a la configuración de conexiones con STEP 7.

Dependiendo del tamaño del DB de configuración, la transmisión al CP puede tener lugar en varios segmentos. Por esta razón se tiene que llamar de nuevo una y otra vez la instrucción hasta que esta señalice que se ha completado la transmisión con el bit DONE=1.

Llamada

Interfaz de llamada en representación FUP



Nota

No utilizar la instrucción IP_CONFIG en caso de emplear un CP con conexiones S7 de alta disponibilidad

Si configura conexiones S7 de alta disponibilidad a través del CP, no debe utilizar la instrucción IP_CONFIG para la configuración IP del CP.

Nota

Atender a un posible direccionamiento doble

Si desea utilizar la instrucción IP_CONFIG, deberá prestar una especial atención a la asignación unívoca de las direcciones IP. Si se detecta una dirección duplicada, puede suceder que el CP no se active en la red.

Consulte también

Sinopsis (Página 8147)

Bloque de datos de configuración (Página 8156)

Parámetros en el caso de IP_CONFIG (Página 8152)

Números de puerto reservados - IP_CONFIG (Página 8153)

Parámetros DONE, ERROR, STATUS (Página 8154)

Parámetros en el caso de IP_CONFIG

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la interfaz de llamada de la instrucción "IP_CONFIG":

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
ACT	INPUT	BOOL, TIMER, COUNTER	0,1	En el caso de una llamada de la instrucción con ACT = 1 se envía el DBxx al CP. En el caso de una llamada de la instrucción con ACT = 0 solo se actualizan las indicaciones de estado DONE, ERROR y STATUS.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7 aparece la dirección inicial del módulo en la tabla de configuración. Indique aquí esta dirección.
CONF_DB	INPUT	ANY		El parámetro hace referencia a la dirección inicial del área de datos de configuración en un bloque de datos (tipo de datos: Byte).
LEN	INPUT	INT		Indicación de longitud en byte para el área de datos de configuración.
DONE	OUTPUT	BOOL	0: - 1: petición concluida con transmisión de datos.	El parámetro indica si el área de datos de configuración se ha transmitido completa. Tenga en cuenta que, dependiendo del tamaño del área de datos de configuración, la instrucción se tiene que impulsar varias veces (en varios ciclos), hasta que la indicación DONE=1 señaliza la finalización. La tabla siguiente informa sobre el significado en relación con los parámetros ERROR y STATUS.
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error La tabla siguiente informa sobre el significado en relación con los parámetros DONE y STATUS.

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
STATUS	OUTPUT	WORD	Véase la tabla siguiente	Indicador de estado La tabla siguiente informa sobre el significado en relación con los parámetros DONE y ERROR.
EXT_Status	OUTPUT	WORD		En caso de una ejecución incorrecta de la petición, el parámetro indica cual de los parámetros del DB de configuración se ha reconocido como causante del fallo. Byte High: Índice del bloque de parámetros Byte Low: Índice del subbloque dentro del bloque de parámetros

Consulte también

IP_CONFIG para enlaces de comunicación programados (Página 8150)

Números de puerto reservados - IP_CONFIG

Números de puerto reservados

Los siguientes números de puertos locales están reservados; no los utilice para la configuración de enlaces.

Tabla 9-353 Números de puerto reservados

Protocolo	Número de puerto	Servicio
TCP	20, 21	FTP
TCP	25	SMTP
TCP	80	HTTP
TCP	102	RFC1006
TCP	135	RPC-DCOM
TCP	502	ASA Application Protocol
UDP	161	SNMP_REQUEST
UDP	34964	PN IO
UDP	65532	NTP
UDP	65533	NTP
UDP	65534	NTP
UDP	65535	NTP

Consulte también

IP_CONFIG para enlaces de comunicación programados (Página 8150)

Parámetros DONE, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Tabla 9-354 Códigos de condición IP_CONFIG

DONE	ERROR	STATUS	Significado
Indicaciones generales sobre la ejecución de la petición			
1	0	0000_H	Petición terminada sin errores.
0	0	8181_H	Petición en curso
Errores que se han detectado en la interfaz entre la CPU y el CP.			
0	1	$80A4_H$	• Error de comunicación en el bus de comunicaciones o bien • Error de datos: No esta ajustado que la configuración se efectúe a través del programa de usuario.
0	1	$80B1_H$	La cantidad de datos a enviar supera el límite superior admisible para este servicio. (Limite superior = 16 KB)
0	1	$80C4_H$	Error de comunicación El error se puede presentar temporalmente; por ello, es conveniente una repetición en el programa de usuario.
0	1	$80D2_H$	Error de configuración El módulo utilizado no da soporte a este servicio.
Errores que se han detectado durante la evaluación de IP_CONFIG en la CPU o en la interfaz entre la CPU y el CP.			
0	1	8183_H	El CP no admite el número de registro de datos solicitado.
0	1	8184_H	Error del sistema o tipo de parámetro no permitido. (tipo de datos del puntero ANY CONF_DB no es correcto) (actualmente sólo se acepta el tipo de datos bytes)
0	1	8185_H	El valor del parámetro LEN es mayor que el CONF_DB menos la cabecera reservada (4 bytes) o la longitud indicada no es correcta.
0	1	8186_H	Se ha detectado parámetro no permitido El puntero ANY CONF_DB no hace referencia a ningún bloque de datos.
0	1	8187_H	Estado no válido de IP_CONFIG Es posible que se hayan sobrescrito datos en la cabecera del CONF_DB.
Errores adicionales que se han detectado en la interfaz entre la CPU y el CP.			
0	1	$8A01_H$	La indicación de estado en el registro de datos leído no es válida (valor >= 3).
0	1	$8A02_H$	No existe ninguna petición en curso en el CP. Sin embargo, el IP_CONFIG estaba esperando a un acuse para la petición en curso.
0	1	$8A03_H$	No existe ninguna petición en curso en el CP y el CP no está listo. El IP_CONFIG ha iniciado una primera petición para leer el registro de datos.
0	1	$8A04_H$	No existe ninguna petición en curso en el CP y el CP no está listo. Sin embargo, el IP_CONFIG estaba esperando a un acuse para la petición en curso.
0	1	$8A05_H$	Existe una petición en curso, sin embargo, no se ha producido el acuse. Sin embargo, el IP_CONFIG ha iniciado una primera petición para leer el registro de datos.
0	1	$8A06_H$	Una petición ha terminado. Sin embargo, el IP_CONFIG ha iniciado una primera petición para leer el registro de datos.

DONE	ERROR	STATUS	Significado
Errores que se han detectado durante las evaluaciones del IP_CONFIG en el CP.			
0	1	$8B01_H$	Error de comunicación No se ha podido transmitir el DB.
0	1	$8B02_H$	Error de parámetro Bloque de parámetros doble
0	1	$8B03_H$	Error de parámetro El subbloque en el bloque de parámetros no está permitido.
0	1	$8B04_H$	Error de parámetro La longitud indicada en la instrucción no coincide con la longitud de los bloques de parámetros/subbloques.
0	1	$8B05_H$	Error de parámetro La longitud del bloque de parámetros no es válida.
0	1	$8B06_H$	Error de parámetro La longitud del subbloque no es válida.
0	1	$8B07_H$	Error de parámetro La ID del bloque de parámetros no es válida.
0	1	$8B08_H$	Error de parámetro La ID del subbloque no es válida.
0	1	$8B09_H$	Error del sistema El enlace no existe
0	1	$8B0A_H$	Error de datos El contenido del subbloque no es correcto.
0	1	$8B0B_H$	Error de estructura Un subbloque aparece por duplicado.
0	1	$8B0C_H$	Error de datos El bloque de parámetros no contiene todos los parámetros necesarios.
0	1	$8B0D_H$	Error de datos El CONF_DB contiene ningún bloque de parámetros para los datos de sistema.
0	1	$8B0E_H$	Error de datos / error de estructura El tipo de CONF_DB no es válido.
0	1	$8B0F_H$	Error del sistema El CP no dispone de suficientes recursos para poder procesar el CONF_DB completamente.
0	1	$8B10_H$	Error de datos No esta ajustado que la configuración se efectúe a través del programa de usuario.
0	1	$8B11_H$	Error de datos El tipo indicado del bloque de parámetros no es válido.
0	1	$8B12_H$	Error de datos Se ha indicado una cantidad excesiva de enlaces (o bien en total o demasiados de un tipo determinado; por ejemplo, sólo se permite una conexión de correo electrónico).
0	1	$8B13_H$	Error interno del CP
0	1	$8B14_H$	El nivel de protección activo no permite la realización del cambio.
Errores adicionales que se han detectado en las interfaces del programa a nivel interno de la CPU (errores de instrucción del sistema).			

DONE	ERROR	STATUS	Significado
0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro.
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos actual protegido de escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos actual protegido de escritura.
0	1	$8F32_H$	El parámetro contiene un número de DB demasiado alto.
0	1	$8F33_H$	Error de número de DB
0	1	$8F3A_H$	El área de destino no se ha cargado (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F44_H$	Está bloqueado el acceso a un parámetro a leer para el procesamiento de la instrucción.
0	1	$8F45_H$	Está bloqueado el acceso a un parámetro a escribir para el procesamiento de la instrucción.
0	1	$8F7F_H$	Error interno Se ha detectado p. ej. referencia ANY no admisible.

Consulte también

IP_CONFIG para enlaces de comunicación programados (Página 8150)

Bloque de datos de configuración

Bloque de datos de configuración (CONF_DB)

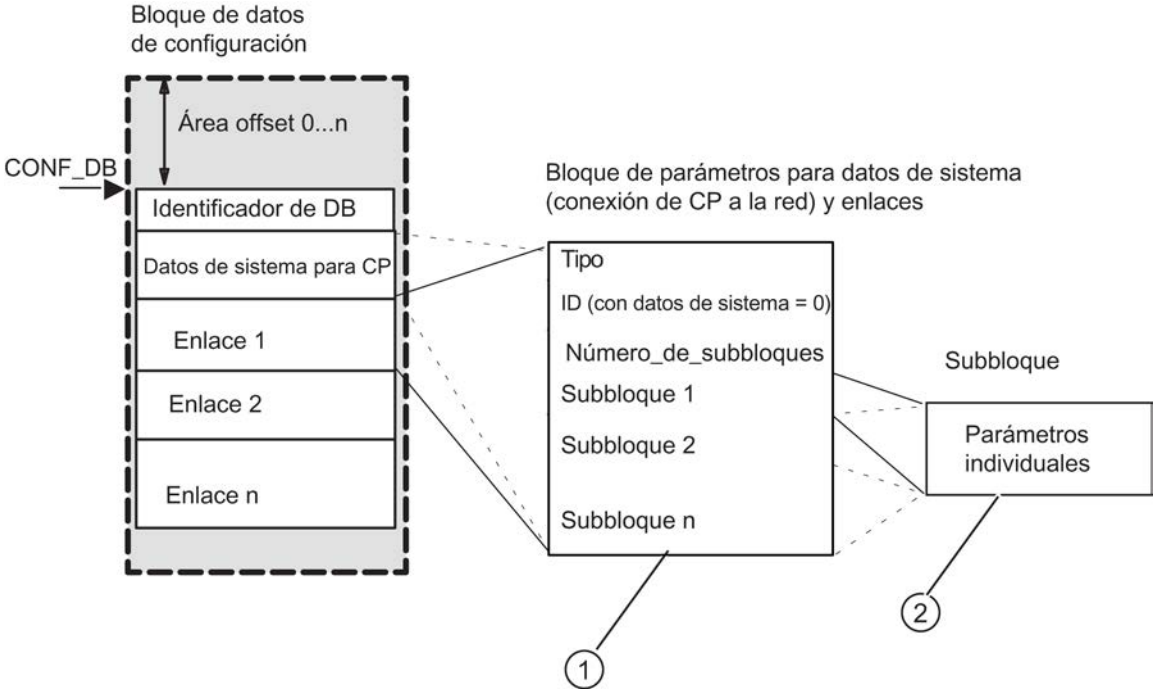
Significado

El bloque de datos de configuración (CONF_DB) contiene todos los datos de enlace y configuración (dirección IP, máscara de subred, router predeterminado, servidor de reloj NTP y otros parámetros) para un CP Ethernet. El bloque de datos de configuración se transfiere al CP con la instrucción IP_CONFIG.

Composición/estructura de datos

En la siguiente representación puede observar:

- La estructuración en bloques de parámetros y subbloques
  - Cada enlace y, en especial, los datos de sistema se describen mediante un bloque de parámetros de idéntica estructura.
  - Los distintos parámetros se tipifican mediante subbloques.
- Área offset
  El CONF_DB puede comenzar en cualquier punto dentro de un bloque de datos a través de un área offset. La única condición es que la dirección (o bien el offset) sea par.



① Los bloques de parámetros se describen a continuación en Bloque de parámetros para datos de sistema (configuración IP) (Página 8157)

② Los tipos de subbloque se describen a continuación en Bloques de parámetros para tipos de enlaces (Página 8161)

Bloque de parámetros para datos de sistema (configuración IP)

Significado

A continuación encontrará el bloque de parámetros relevante para la configuración IP del CP para datos de sistema y los subbloques que deben indicarse en él.

Dependiendo de cada caso, no son necesarios todos los tipos de subbloque (véase la tabla para más información).

Estructura

Para los CPs con varias interfaces, la estructura de bloque de parámetros descrita a continuación sólo es aplicable para la interfaz PROFINET.

• **Tipo = 0**
• **ID = 0**
• Número_subbloques = n
• Subbloque 1
• Subbloque 2
• Subbloque n ...

Subbloques utilizables

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
1	SUB_IP_V4	Dirección IP local	++
2	SUB_NETMASK	-	++
8	SUB_DEF_ROUTER	-	+
4	SUB_DNS_SERV_ADDR *)	Este subbloque puede aparece hasta 0-4 veces. La primera entrada es el Primary DNS Server	+
14	SUB_DHCP_ENABLE	0: sin DHCP 1: DHCP	+
15	SUB_CLIENT_ID	- Observación: Sólo útil si SUB_DHCP_ENABLE = 1	+
30**)	SUB_DEVICE_NAME	Nombre de dispositivo según la convención PROFINET IO Indique un nombre de dispositivo para permitir la identificación unívoca del dispositivo para fines de análisis y diagnóstico en la red.	+
*) El tipo de subbloque sólo se utiliza para enlaces de correo electrónico. **) El ID sólo es soportado por determinados tipos de CP. ***) ++ = obligatorio; + = opcional			

Consulte también

Tipos de subbloque (Página 8158)

Tipos de subbloque

Dependiendo del bloque de parámetros, se requieren diferentes parámetros. Cada parámetro es descrito por un subbloque. Los subbloques necesarios pueden consultarse en las descripciones de los datos de sistema y de los tipos de enlace de los capítulos anteriores.

Cada subbloque se compone de la sección de parámetro específica y de un encabezado (4 bytes).

Ejemplo

El siguiente extracto de un CONF_DB muestra la estructura de un subbloque a partir del ejemplo del tipo de subbloque SUB_NETMASK.

Dirección	Nombre	Tipo	Valor inicial	Comentario
+14.0	Sub_bloque_2	STRUCT		// Subbloque 2 tipo SUB_NETMASK
+0.0	ID_de_sub_bloque	INT	2	// ID de subbloque
+2.0	Sub_bloque_Len	INT	8	// Longitud total del subbloque en bytes
+4.0	Parámetro	STRUCT		Área de parámetros SUB_NETMASK
+0.0	Valor_1	BYTE	B#16#FF	
+1.0	Valor_2	BYTE	B#16#FF	
+2.0	Valor_3	BYTE	B#16#FF	
+3.0	Valor_4	BYTE	B#16#0	
=4.0	END_	STRUCT		
=8.0	END_	STRUCT		

En total están disponibles los siguientes tipos de subbloque:

ID de subbloque [1)]	Tipo de subbloque	Longitud de subbloque (en bytes)	Significado del parámetro
1	SUB_IP_V4	4 + 4	Dirección IP según IPv4
2	SUB_NETMASK	4 + 4	Máscara de subred
3	SUB_DNS_NAME	Longitud nombre DNS + 4	Nombre DNS
4	SUB_DNS_SERV_ADDR	4 + 4	Dirección servidor DNS
8	SUB_DEF_ROUTER	4 + 4	Dirección IP del router predeterminado
9	SUB_LOC_PORT	2 + 4	Puerto local
10	SUB_REM_PORT	2 + 4	Puerto remoto, también para enlaces de correo electrónico
11	SUB_LOC_TSAP	Longitud TSAP + 4	TSAP local
12	SUB_REM_TSAP	Longitud TSAP + 4	TSAP remoto
13	SUB_EMAIL_SENDER	Longitud de la dirección del remitente del mensaje + 4	Dirección de correo electrónico del remitente
14	SUB_DHCP_ENABLE	2 + 4	Obtener la dirección IP de un servidor DHCP. • Rango de valores: 0 = sin DHCP 1 = DHCP (opcional)
15	SUB_CLIENT_ID	Longitud del ID de cliente + 4	(opcional)

ID de subbloque [1)]	Tipo de subbloque	Longitud de subbloque (en bytes)	Significado del parámetro
18	SUB_CONNECT_NAME	Longitud del nombre + 4	Nombre del enlace. Los caracteres posibles son: a...z, A...Z, 0...9, -, _
19	SUB_LOC_MODE	1 + 4	Modo de operación local del enlace • Rango de valores: 0x00 = SEND/RECV 0x01 = protocolo FTP (sólo enlace TCP) 0x10 = modo de direccionamiento S5 con FETCH/WRITE *) 0x20 = SPEED SEND/RECV (sólo permitido con CP 443-1 Advanced) 0x80 = FETCH *) 0x40 = WRITE *) El ajuste predeterminado renunciando al parámetro es SEND/RECV. Observación: FETCH / WRITE requieren el ajuste de una estructura de enlace pasiva (véase SUB_CON_ESTABL).
20	SUB_REM_MODE	1 + 4	Ajuste del modo de operación en el interlocutor. (actualmente no soportado)
21	SUB_KBUS_ADR	5	Dirección KBUS de la CPU (sólo relevante para S7–400)
22	SUB_CON_ESTABL	1 + 4	Tipo de establecimiento del enlace. Con esta opción se determina si el enlace debe ser establecido por este equipo S7 de forma activa o pasiva. • Rango de valores: 0 = pasivo 1 = activo
23	SUB_ADDR_IN_DATA-BLOCK	1 + 4	Seleccionar un enlace UDP libre. Con la llamada de AG_SEND, el programa de usuario registra la estación remota en el encabezado de petición del búfer de peticiones. De ese modo es posible acceder a cualquier estación de Ethernet/LAN/WAN. • Rango de valores: 1 = enlace UDP libre. 0 = otros El parámetro sólo es útil para el enlace UDP.
24	SUB_NTP_SERVER	4 + 4	El subbloque define un servidor NTP del cual el CP puede obtener la hora a través del protocolo NTP. Si se definen uno o más servidores NTP, pueden definirse hasta 4 subbloques del ID 24. Los subbloques del ID 24 sólo pueden integrarse en el bloque de parámetros de sistema del tipo 0 / ID 0.

ID de subbloque [1]	Tipo de subbloque	Longitud de subbloque (en bytes)	Significado del parámetro
30	SUB_DEVICE_NAME	Longitud del nombre + 4	Nombre de dispositivo según la convención PROFINET IO El nombre del dispositivo debe cumplir las convenciones DNS, es decir, • estar limitado a 127 caracteres en total (letras, cifras, guiones o puntos) • Cada componente del nombre del dispositivo, es decir, una cadena de caracteres entre dos puntos, no puede tener una longitud superior a 63 caracteres. • No se admiten caracteres especiales, como diéresis, paréntesis, guiones bajos, barras oblicuas, espacios en blanco, etc. El guión es el único carácter especial permitido. • El nombre del dispositivo no puede comenzar ni terminar con los caracteres "-" o ".". • El nombre del dispositivo no puede comenzar con una cifra. • El nombre de dispositivo no puede tener el formato n.n.n.n (n = 0...999). • El nombre de dispositivo no puede comenzar con la secuencia de caracteres "port-xyz-" (x,y,z = 0...9).
1) Nota: Actualmente no se utilizan números de ID no indicados.			

Bloques de parámetros para tipos de enlaces

Información general

A continuación se indica qué valores deben introducirse en los bloques de parámetros y qué subbloques deben utilizarse para los correspondientes tipos de enlace.

Dependiendo de cada caso, no son necesarios todos los tipos de subbloque (véase la tabla para más información).

ID de enlace

El parámetro ID que precede a cada bloque de parámetros de enlace junto al identificador de tipo es de especial relevancia.

En los enlaces programados, este ID puede asignarse libremente dentro del rango de valores admisible. A continuación se debe utilizar ese ID en la interfaz de llamada de las FCs para la interfaz SEND/RECV para identificar el enlace.

Rangos de valores para el ID de enlace:

- S7-400: 1,2...64
- S7-300: 1,2...16

Consulte también

Tipos de subbloque (Página 8157)

Bloque de parámetros para enlaces TCP

Estructura

Guarde los parámetros en el bloque de parámetros para enlaces TCP del siguiente modo:

• **Tipo = 1 -> ①**
• **ID = ID de enlace -> ②**
• Número_subbloques = n
• Subbloque 1
• Subbloque 2
• Subbloque n ...

Leyenda:
① Identificador del tipo de enlace
② Referencia de enlace de libre elección; debe indicarse en AG_SEND / AG_RECV.
Rango de valores para el ID de enlace:
para S7-400: 1, 2...64
para S7-300: 1,2...16

Subbloques utilizables

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
1	SUB_IP_V4	Dirección IP del interlocutor	++ *)
9	SUB_LOC_PORT	-	++
10	SUB_REM_PORT	-	++ **)
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	-	+
21	SUB_KBUS_ADR	En CPs para S7-300, este valor tiene asignado el valor predeterminado fijo 2, por lo que no es necesario indicarlo.	++ (para S7-400)
22	SUB_CON_ESTABL	-	++
*) Opcional para enlace pasivo. ***) ++ = obligatorio; + = opcional			

Consulte también

Tipos de subbloque (Página 8157)

Bloque de parámetros para enlaces UDP

Estructura

Guarde los parámetros en el bloque de parámetros para enlaces UDP del siguiente modo:

• **Tipo = 2 -> ①**
• **ID = ID de enlace -> ②**
• Número_subbloques = n
• Subbloque 1
• Subbloque 2
• Subbloque n ...

Leyenda:

① Identificador del tipo de enlace

② Referencia de enlace de libre elección; debe indicarse en AG_SEND / AG_RECV.
Rango de valores para el ID de enlace:
para S7-400: 1, 2...64
para S7-300: 1,2...16

Subbloques utilizables

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
1	SUB_IP_V4	Dirección IP del interlocutor	++
9	SUB_LOC_PORT	-	++
10	SUB_REM_PORT	-	++
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	-	+
21	SUB_KBUS_ADR	En CPs para S7-300, este valor tiene asignado el valor predeterminado fijo 2, por lo que no es necesario indicarlo.	++ (para S7-400)
23	SUB_ADDR_IN_DATABLOCK	Si se selecciona la opción "Enlace UDP libre" a través de este parámetro, se prescinde de los parámetros SUB_IP_V4 y SUB_REM_PORT.	+
***) ++ = obligatorio; + = opcional			

Consulte también

Tipos de subbloque (Página 8157)

Bloque de parámetros para enlaces ISO-on-TCP

Estructura

Guarde los parámetros en el bloque de parámetros para enlaces ISO-on-TCP del siguiente modo:

• **Tipo = 3 -> ①**
• **ID = ID de enlace -> ②**
• Número_subbloques = n
• Subbloque 1
• Subbloque 2
• Subbloque n ...

Leyenda:
① Identificador del tipo de enlace
② Referencia de enlace de libre elección; debe indicarse en AG_SEND / AG_RECV.
Rango de valores para el ID de enlace:
para S7-400: 1, 2...64
para S7-300: 1,2...16

Subbloques utilizables

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
1	SUB_IP_V4	Dirección IP del interlocutor	++ *)
11	SUB_LOC_TSAP	-	++
12	SUB_REM_TSAP	-	++ *)
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	-	+
21	SUB_KBUS_ADR	En CPs para S7-300, este valor tiene asignado el valor predeterminado fijo 2, por lo que no es necesario indicarlo.	++ (para S7-400)
22	SUB_CON_ESTABL	-	++
*) Opcional para enlace pasivo. ***) ++ = obligatorio; + = opcional			

Consulte también

Tipos de subbloque (Página 8157)

Bloque de parámetros para enlaces de correo electrónico

Significado

Para enviar mensajes de correo electrónico debe configurarse básicamente un enlace de correo electrónico por cada CP Advanced. Con el enlace de correo electrónico se define el servidor de correo a través del cual se pondrán a disposición todos los mensajes de correo electrónico enviados por el CP Advanced.

Estructura

Guarde los parámetros en el bloque de parámetros para enlaces de correo electrónico del siguiente modo:

• **Tipo = 4 -> ①**
• **ID = ID de enlace -> ②**
• Número_subbloques = n
• Subbloque 1
• Subbloque 2
• Subbloque n ...

Leyenda:

① Identificador del tipo de enlace

② Referencia de enlace de libre elección; debe indicarse en AG_SEND / AG_RECV.
Rango de valores para el ID de enlace:
para S7-400: 1, 2...64
para S7-300: 1,2...16

Subbloques utilizables

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
1	SUB_IP_V4	Dirección IP del servidor de correo electrónico a través del cual se envían los mensajes. La dirección IP puede indicarse en formato absoluto o simbólico. El formato simbólico presupone que el CP Advanced conoce la dirección del Domain Name Server (DNS). La entrada correspondiente debe realizarse en la configuración del CP Advanced en HW Config (allí encontrará información adicional en la Ayuda en pantalla).	++ / + *)
3	SUB_DNS_NAME	Nombre DNS del servidor de correo electrónico	++ / + *)

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
13	SUB_EMAIL_SENDER	Dirección remitente del mensaje de correo electrónico	++
18	SUB_CONNECT_NAME	-	+
21	SUB_KBUS_ADR	En CPs para S7-300, este valor tiene asignado el valor predeterminado fijo 0, por lo que no es necesario indicarlo.	++ (para S7-400)
22	SUB_CON_ESTABL	-	++
*) En este caso, los parámetros SUB_IP_V4 y SUB_DNS_NAME se excluyen mutuamente; sólo puede indicarse uno de los parámetros. ***) ++ = obligatorio; + = opcional			

Nota

Los Mail-Server-Ports son "well known ports" y no es necesario indicarlos.

Consulte también

Tipos de subbloque (Página 8157)

Bloque de parámetros para enlaces FTP

Significado

Para el desarrollo de una secuencia de petición FTP entre el equipo S7 como cliente FTP y un servidor FTP es necesario que el CP Advanced establezca un enlace con la CPU S7. Aquí, dicho enlace se denomina "enlace FTP".

Los enlaces FTP son enlaces TCP que se ajustan al modo "FTP" con el parámetro SUB_LOC_MODE.

Estructura

Guarde los parámetros en el bloque de parámetros para enlaces FTP del siguiente modo:

• **Tipo = 1 -> ①**
• **ID = ID de enlace -> ②**
• Número_subbloques = n
• Subbloque 1

• Subbloque 2
• Subbloque n ...

Leyenda:
① Identificador del tipo de enlace
② Referencia de enlace de libre elección; debe indicarse en AG_SEND / AG_RECV.
Rango de valores para el ID de enlace:
para S7-400: 1, 2...64
para S7-300: 1,2...16

Subbloques utilizables

Subbloque		Parámetro	
ID	Tipo	Particularidades/indicaciones	Aplicación ***)
18	SUB_CONNECT_NAME	-	+
19	SUB_LOC_MODE	Aquí: 0x01 = protocolo FTP	++
21	SUB_KBUS_ADR	En CPs para S7-300, este valor tiene asignado el valor predeterminado fijo 0, por lo que no es necesario indicarlo.	++ (para S7-400)
***) ++ = obligatorio; + = opcional			

Consulte también

Tipos de subbloque (Página 8157)

Instrucciones para servicios FTP

FTP_CMD para servicios FTP

Migración de FC 40-44 a FB 40

Comparación del bloque de programa FTP_CMD con las anteriores funciones FC40...44

Todos los CPs con funcionalidad FTP son compatibles con las funciones FC40...44. Por lo tanto, los programas de usuario disponibles se pueden seguir utilizando sin modificación.

Si desea cambiar de las funciones FTP FC40...44 a FTP_CMD, tendrá que modificar su programa de usuario.

La tabla siguiente muestra con qué comandos del FTP_CMD puede implementar las funciones FC40...44.

- Las coincidencias están marcadas con "X".
- La falta de correspondencias está marcada con "-".

	Comandos del parámetro "CMD" del FTP_CMD					
Antiguas funciones FTP FC40...44	CMD = 1	CMD = 2	CMD = 3	CMD = 4	CMD = 5	CMD = 6 CMD = 7 CMD = 17
FC40	X [1]					
FC41		X [2]				
FC42			X [3]			
FC43				X [4]		
FC44					X	
						-
[1]...[4] Los parámetros de FC 40...43 y CMD 1...4 (FTP_CMD) no son idénticos. (véase la tabla siguiente)						

Los parámetros correspondientes que especifican la respectiva función en las funciones FC40...FC43 o en los comandos del FTP_CMD se listan en la tabla siguiente.

Parámetros de FC			Parámetros del FTP_CMD (con CMD 1...4)	
FC40:	LOGIN	→	CMD = 1:	NAME_STR
FC41:	FILE_NAME	→	CMD = 2:	NAME_STR
FC42:	FILE_NAME	→	CMD = 3:	NAME_STR
FC43:	FILE_NAME	→	CMD = 4:	NAME_STR
FC40...43:	BUFFER_DB_NR	→	suprimido (sustituido por DB de instancia)	

Consulte también

Panorámica de FTP_CMD (Página 8168)

Panorámica de FTP_CMD

Significado

Con la instrucción FTP_CMD se pueden establecer conexiones FTP y transmitir archivos desde y a un servidor FTP.

La instrucción FTP_CMD aporta las siguientes ventajas:

- Simplificación en el programa de usuario por desarrollo a través de variables de comandos en lugar de diferentes llamadas de funciones.
- Función adicional "APPEND"
  "APPEND" permite añadir datos a un archivo ya existente.
- Función adicional "RETR_PART"
  "RETR_PART" permite leer selectivamente áreas de datos de un archivo.
- Función adicional "CONNECT_TLS_PRIVATE"
  "CONNECT_TLS_PRIVATE" permite establecer conexiones SSL seguras
- La función AG_SEND (FC5) no se necesita para la ejecución.

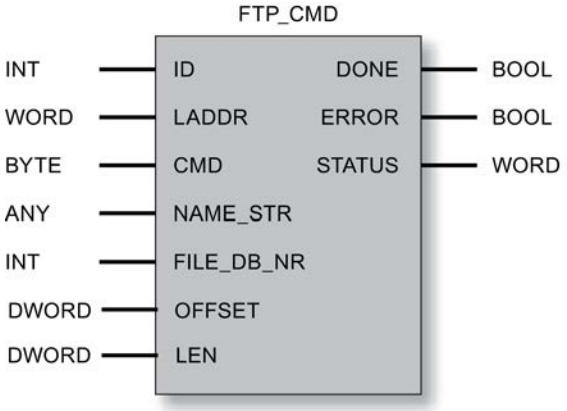
Validez

El FB40 se puede utilizar a partir de los siguientes tipos de módulos:

- A partir de CP 343-1 Advanced (GX30* / GX31)
- A partir de CP 443-1 Advanced (GX20* / GX30)
  *) Función "CONNECT-TLS-PRIVATE" no utilizable.

Interfaz de llamada

Interfaz de llamada en representación FUP



Funciones del sistema llamadas

El bloque de programa FTP_CMD llama las siguientes funciones de sistema:

SFC 1, SFC 20, SFC 24, SFC 58, SFC 59

Nota

Tenga en cuenta que los servicios de cliente FTP de CPUs SIMATIC S7-300 menos recientes, por ejemplo CPU 312 o CPU 315-1AF01, no se pueden ejecutar, ya que no soportan la función especial SFC 24.

Consulte también

Estructura y aplicación del DB File (Página 8178)

Parámetros de salida e informaciones de estado FTP_CMD (Página 8174)

Parámetros de entrada FTP_CMD (Página 8170)

Migración de FC 40-44 a FB 40 (Página 8166)

Parámetros de entrada FTP_CMD

Explicación de los parámetros de entrada

Cada llamada de bloque FTP se tiene que dotar de los siguientes parámetros de entrada:

Tabla 9-355 Parámetros formales del FB40 (FTP_CMD) - parámetros de entrada

Parámetro	Declaración	Tipo	Rango de valores	Significado / Observación
ID	INPUT	INT	Para S7-300: 1, 2...16 Para S7-400: 1, 2...64	Las peticiones de FTP se desarrollan a través de conexiones FTP. El parámetro identifica la conexión utilizada.
LADDR	INPUT	WORD		Dirección inicial del módulo Al llamar un bloque FC transfiere en el parámetro LADDR la dirección del inicio de módulo del ADVANCED CP. La dirección inicial de módulo del ADVANCED-CP se puede consultar en la configuración del ADVANCED-CP bajo "Propiedades>Direcciones>Entradas".
CMD	INPUT	BYTE	Consulte la tabla inferior Comandos FTP en el parámetro "CMD"	Comandos FTP que se ejecutan con la llamada del FTP_CMD. Encontrará más información a continuación de la tabla. Si un comando no es compatible con el firmware del CP, se emite un mensaje de error con STATUS = $8F6B_H$. Ejemplos de comandos FTP: • RETRIEVE: B#16#3 • CONNECT_TLS_PRIVATE: B#16#11

Parámetro	Declaración	Tipo	Rango de valores	Significado / Observación
NAME_STR	INPUT	ANY	como VARTYPE solo se admite "BYTE".	La dirección indicada hace referencia a un área del bloque de datos. Aquí se tiene que indicar la dirección y la longitud del área de datos en la que están registrados las informaciones de destino. • Con CMD = 1, 17: En este comando, el parámetro "NAME_STR" especifica el servidor FTP al que se debe acceder con la conexión FTP con los siguientes atributos: - dirección IP del servidor FTP - nombre del usuario - contraseña para el inicio de sesión Estos valores se tienen que indicar como tres strings consecutivos en el área de destino del puntero ANY. • Con CMD = 2, 3, 4, 6, 7: En este comando, el parámetro "NAME_STR" especifica el nombre del archivo en el servidor FTP, es decir, la fuente de los datos o el destino de los datos. El nombre del archivo se tiene que indicar como string en el área de destino del puntero ANY. • Con CMD = 5: parámetro no relevante Encontrará ejemplos de contenidos más adelante.
FILE_DB_NR	INPUT	INT		El bloque de datos aquí indicado contiene el File-DB que se debe leer / escribir. El parámetro solo es relevante con CMD = 2, 3, 6 y 7.

Parámetro	Declaración	Tipo	Rango de valores	Significado / Observación
OFFSET	INPUT	DWORD		Solo con CMD = 7: Offset en bytes, a partir del cual se debe leer el archivo.
LEN	INPUT	DWORD		Solo con CMD = 7: Longitud parcial en bytes que se debe leer a partir del valor indicado en "OFFSET". Peculiaridades: • Si se indica "DW#16#FFFFFFFF" se lee el resto disponible del archivo. Resultado OK (DONE = 1, STATUS = 0), si no se presenta ningún otro error. • Si OFFSET > longitud del archivo original: En este caso, la longitud del archivo de destino se muestra del siguiente modo en el parámetro ACT_LENGTH del File-DB: 0 bytes en la CPU. Resultado OK (DONE = 1, STATUS = 0), si no se presenta ningún otro error. • Si OFFSET + LEN > longitud del archivo original (y LEN ≠ 0xFFFFFFFF): En este caso, la longitud del archivo de destino se muestra del siguiente modo en el parámetro ACT_LENGTH del File-DB: Bytes disponibles a partir de "OFFSET". Resultado OK (DONE = 1, STATUS = 0), si no se presenta ningún otro error.

Comandos FTP en el parámetro "CMD"

Vea en la tabla siguiente qué significado tienen los comandos del parámetro "CMD" y qué parámetros de entrada (Input) se tienen que abastecer en cada caso. Los parámetros ID y LADDR se tienen que activar siempre para identificar la conexión.

CMD	Parámetro de entrada relevante (además de ID y LADDR)	Significado / Manejo
0 (NOOP)	-	La instrucción llamada no ejecuta ninguna acción. Las indicaciones de estado se ponen como sigue para este abastecimiento de parámetros: • DONE=1; ERROR=0; STATUS=0
1 (CONNECT)	NAME_STR	El cliente FTP establece con este comando una conexión FTP con un servidor FTP (puerto 21). La conexión está disponible, con la ID de conexión aquí asignada, para todos los demás comandos FTP. Los datos se intercambian entonces con el servidor FTP indicado para este usuario.
2 (STORE)	NAME_STR FILE_DB_NR	Con esta llamada de función se transmite un bloque de datos (File-DB) del cliente FTP (CPU S7) al servidor FTP. Atención: Si el archivo (File-DB) ya existe en el servidor FTP, se sobrescribe.

CMD	Parámetro de entrada relevante (además de ID y LADDR)	Significado / Manejo
3 (RETRIEVE)	NAME_STR FILE_DB_NR	Con esta llamada de función se transmite un archivo del servidor FTP al cliente FTP (CPU S7). Atención: Si el bloque de datos (File-DB) del cliente FTP ya contiene un archivo, este se sobrescribe.
4 (DELETE)	NAME_STR	Con esta llamada de función se borra un archivo en el servidor FTP.
5 (QUIT)	Ninguno más	Con esta llamada de función se deshace la conexión FTP identificada con la ID.
6 (APPEND)	NAME_STR FILE_DB_NR	De forma similar a "STORE", el comando "APPEND" (añadir) guarda un archivo en el servidor FTP. Pero con "APPEND" no se sobrescribe el archivo en el servidor FTP, sino que el nuevo contenido a guardar se añade al archivo. Si el archivo no existe en el servidor FTP, se crea.
7 (RETR_PART)	NAME_STR FILE_DB_NR OFFSET LEN	Con el comando "RETR_PART" (leer longitud parcial) se puede pedir del servidor FTP una parte de un archivo. En caso de archivos muy grandes se puede limitar así la lectura a la parte necesaria. Para esto se tiene que conocer la estructura del archivo. Indique la parte deseada del archivo con ayuda de los dos parámetros "OFFSET" y "LEN" en el FB 40.
17 (CONNECT_TLS_PRIVATE)	NAME_STR	Con el comando "CONNECT_TLS_PRIVATE", el cliente FTP establece una conexión FTP segura vía SSL (FTPS) con el servidor FTP (puerto 21). Los datos de la conexión de control y los de la conexión de datos son seguros. La conexión está disponible, con la ID de conexión aquí asignada, para todos los demás comandos FTP. Los datos se intercambian entonces con el servidor FTP indicado para este usuario. Requisitos: • La opción "Seguridad" debe estar activada en el CP. • Para la conexión FTP asegurada con SSL se deben suministrar los certificados en la configuración del CP.

Ejemplos de contenidos del parámetro "NAME_STR"

El registro de parámetros tiene los siguientes contenidos:

Tabla 9-356 Contenido del registro de parámetros para CMD = 1, 17

Dirección relativa 2)	Nombre	Tipo 1)	Ejemplo	Significado
0.0	ip_address	STRING[100]	’142.11.25.135’	Dirección IP del servidor FTP
102.0	username	STRING[32]	’usuario’	Nombre del usuario para iniciar sesión en el servidor FTP
136.0	password	STRING[32]	’contraseña’	Contraseña para iniciar sesión en el servidor FTP
1) Se indica la longitud máxima posible del string en cada caso 2) Los valores indicados se refieren a las longitudes de string indicadas en "Tipo".				

Tabla 9-357 Contenido del registro de parámetros para CMD = 2, 3, 4, 6, 7

Dirección relativa 2)	Nombre	Tipo 1)	Ejemplo	Significado
170.0	filename	STRING[220] o STRING[212]	'instalación1/caldera2/ presión.dat'	Nombre del fichero de destino o fuente Nota: con CMD=7 (RETR_PART) la longitud máxima del nombre del archivo está limitada a 212 caracteres.
1) Se indica la longitud máxima posible del string en cada caso 2) Los valores indicados se refieren a las longitudes de string indicadas en "Tipo".				

Consulte también

Panorámica de FTP_CMD (Página 8167)

Parámetros de salida e informaciones de estado FTP_CMD

Introducción

Para la evaluación del estado se tienen que evaluar estos parámetros en el programa de usuario:

Tabla 9-358 Parámetros formales del FB40 (FTP_CMD) - parámetros de salida

Parámetro	Declaración	Tipo	Rango de valores	Significado / Observación
DONE	OUTPUT	BOOL	0: - 1: petición ejecutada	El parámetro de estado indica si la petición se ha desarrollado correctamente.
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error El parámetro notifica que la petición no se ha podido ejecutar sin errores.
STATUS	OUTPUT	WORD	Véase la tabla siguiente	Indicador de estado El parámetro proporciona informaciones detalladas sobre la ejecución de la petición.

Los parámetros DONE, ERROR y STATUS se actualizan a cada llamada del bloque.

Ejemplo

Durante una ejecución de petición, el FTP_CMD proporciona los códigos de condición:

- DONE=0
- ERROR=0
- STATUS=8181_H

Significado: La petición aún está en curso.

Evaluar indicaciones de estados

Nota

Para las entradas con la codificación 8FxxH en STATUS, tenga en cuenta también lo dicho en el manual de referencia STEP 7 Standard y funciones del sistema. Allí encontrará informaciones en el capítulo "Evaluación de fallos con el parámetro de salida RET_VAL".

Tabla 9-359 FB 40: Significado del parámetro STATUS en relación con DONE y ERROR

DONE	ERROR	STATUS	Significado
0	0	0000_H	Ninguna petición en proceso.
1	0	0000_H	Petición terminada sin errores.
0	0	8181_H	Petición en curso. En caso de indicación continua de 8181_H: el CP no está habilitado para el FTP_CMD (se ha llamado un comando CMD 6, CMD 7 o CMD 17 no permitido para la versión de firmware.)
0	1	8090_H	• No existe un módulo con esta dirección inicial de módulo. • El bloque utilizado no es acorde con la familia de sistemas utilizada (se tienen que utilizar FCs diferentes para S7-300 y S7-400).
0	1	8091_H	La dirección inicial del módulo no está en formato de palabra doble.
0	1	8092_H	El tipo indicado en el puntero ANY no es Byte
0	1	$80A4_H$	La conexión de bus K entre CPU y CP no está establecida (en caso de versiones de CPU más recientes). Esto se puede deber, por ejemplo, a lo siguiente: • falta configuración de conexiones • se supera la cantidad máxima de CPs que pueden trabajar en paralelo
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	Área de destino no válida; por ejemplo, área de destino > 240 bytes.
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida (en caso de versiones de CPU menos recientes). (en caso de versiones de CPU más recientes, véase $80A4_H$)
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Medio de servicio (memoria) ocupado.
0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.)
0	1	$80D2_H$	Dirección inicial del módulo errónea.
0	1	8183_H	La configuración no es acorde con los parámetros de la petición.
0	1	8184_H	Se ha indicado un tipo de datos no permitido para el parámetro NAME_STR.
0	1	8186_H	Parámetro ID no válido. ID = 1, 2....64
0	1	8302_H	

DONE	ERROR	STATUS	Significado
0	1	$8F22_H$	Área fuente no valida, por ejemplo: Área no existente en DB
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	Error de área al leer un parámetro
0	1	$8F28_H$	Error de alineación al leer un parámetro
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Error de número de DB
0	1	$8F3A_H$	Área de destino no cargada (DB)
0	1	$8F50_H$	File-DB DB 0 o DB no existe
0	1	$8F51_H$	El área de datos de File-DB se ha indicado mayor que la existente
0	1	$8F52_H$	File-DB en memoria protegida de grabación
0	1	$8F53_H$	File-DB con longitud máx. < longitud actual
0	1	$8F54_H$	File-DB no contiene datos válidos.
0	1	$8F55_H$	Bit de estado del encabezado: Locked
0	1	$8F56_H$	El bit NEW en el encabezado del File-DB no se ha desactivado
0	1	$8F57_H$	El cliente FTP no tiene derecho de escritura en el File-DB, sino el servidor FTP (bit de estado del encabezado: WriteAccess).
0	1	$8F60_H$	Datos de usuario no válidos, por ejemplo, dirección IP del servidor FTP no válida
0	1	$8F61_H$	Servidor FTP no accesible
0	1	$8F62_H$	Significados posibles: • El servidor FTP no da soporte a la petición o la rechaza • El servidor FTP no soporta conexiones seguras vía SSL.
0	1	$8F63_H$	Transferencia de datos cancelada por el servidor FTP
0	1	$8F64_H$	Error en la conexión de control FTP; no se han podido transmitir o recibir datos; la conexión de control FTP se tiene que establecer de nuevo después de tal error.
0	1	$8F65_H$	Error en la conexión de datos FTP; no se han podido transmitir o recibir datos. La petición se tiene que llamar de nuevo. El error se puede deber, por ejemplo en la función RETRIEVE (CMD=3), a que el archivo a que se quiere acceder en el servidor FTP ya está abierto.
0	1	$8F66_H$	Error al leer/escribir datos de/en la CPU (por ejemplo, DB inexistente o demasiado pequeño)
0	1	$8F67_H$	Error en el cliente FTP del CP; por ejemplo, al intentar abrir más conexiones FTP del número máximo posible.
0	1	$8F68_H$	La petición ha sido rechazada por el cliente FTP. El error se puede deber, por ejemplo en la función RETRIEVE (CMD=3), a que el valor del parámetro MAX_LENGTH se ha elegido demasiado pequeño en el encabezado del File-DB.

DONE	ERROR	STATUS	Significado
0	1	$8F69_H$	La conexión FTP se encuentra en un estado incorrecto, p. ej.: • Se produce una llamada de la conexión sin deshacerla previamente (con igual ID de conexión); • Se deshace una conexión ya deshecha; • Se ha enviado un comando STORE para una conexión no establecida.
0	1	$8F6A_H$	La conexión no se ha podido establecer debido a un déficit transitorio de recursos. Solución: repetir la llamada del bloque.
0	1	$8F6B_H$	Posibles causas: • Valor erróneo para el parámetro CMD • No se soporta un comando del FTP_CMD. Posible causa: Firmware incorrecto del CP Solución: actualizar el firmware
0	1	$8F6C_H$	En el parámetro OFFSET se ha indicado un valor > 7FFF FFF_H.
0	1	$8F6D_H$	El cliente FTP no soporta conexiones seguras vía SSL.
0	1	$8F6F_H$	Posibles causas: • La indicación de tiempo "notBefore" del certificado contiene un valor no válido. • El certificado no es válido: la entrada "notBefore" contiene una indicación de tiempo posterior al tiempo actual.
0	1	$8F70_H$	Posibles causas: • La indicación de tiempo "notAfter" del certificado contiene un valor no válido. • El certificado ha caducado: la entrada "notAfter" contiene una indicación de tiempo anterior al tiempo actual.
0	1	$8F71_H$	No se ha encontrado el certificado del emisor de un certificado no fiable.
0	1	$8F72_H$	El certificado CA original no es válido. Es posible que no se trate de un certificado CA o que sus ampliaciones no sean coherentes con la finalidad prevista.
0	1	$8F73_H$	El certificado CA original está marcado como no fiable para la finalidad indicada.
0	1	$8F74_H$	Se han producido otros errores al verificar un certificado.
0	1	$8F7F_H$	Error interno p. ej., referencia ANY no permitida.

Consulte también

Panorámica de FTP_CMD (Página 8167)

Estructura y aplicación del DB File

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Server

Funcionamiento

Para la transmisión de datos mediante FTP tiene que crear bloques (File-DBs) en la CPU de su estación S7. Estos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como datos transmisibles. Constan de los siguientes sectores

- Sector 1: File-DB Header (tiene una longitud fija de 20 byte y una estructura fija)
- Sector 2: datos útiles (longitud y estructura variables)

File-DB Header para modo FTP-Server

Observación: El File-DB Header aquí descrito es prácticamente idéntico al File-DB Header descrito para el modo Client. Las diferencias residen en los parámetros

- WRITE_ACCESS
- FTP_REPLY_CODE

Parámetro	Tipo	Valor / Significado	Alimentación
EXIST	BOOL	El bit EXIST indica si hay datos válidos en el área de datos útiles. El comando FTP retrieve sólo procesa la petición si EXIST=1. • 0: El File-DB no contiene datos útiles válidos ("Archivo no existe"). • 1: El File-DB contiene datos útiles válidos ("Archivo existe").	El comando FTP dele pone EXIST=0; El comando FTP store pone EXIST=1;
LOCKED	BOOL	El bit LOCKED sirve para la protección de acceso al File-DB. • 0: Se puede acceder al File-DB. • 1: El File-DB está bloqueado.	Los comandos FTP store y retr ponen LOCKED=1 durante la edición. Para una operación de escritura desde el programa de usuario rige: El programa de usuario almacenado en la CPU S7 puede poner o reponer LOCKED para asegurar la coherencia durante un acceso de escritura. Procedimientom recomendado para el programa de usuario: 1. Comprobar el bit LOCKED; si =0 2. poner WRITEACCESS-Bit=0 3. Comprobar el bit LOCKED; si =0 4. poner LOCKED-Bit=1 5. Escribir datos 6. poner LOCKED-Bit=0
NEW	BOOL	El bit NEW-Bit informa sobre si los datos se han modificado desde la última operación de lectura. • 0: El contenido del File-DB no ha cambiado desde la última operación de escritura. El programa de usuario de la CPU S7 ha registrado el último cambio. • 1: El programa de usuario de la CPU S7 no ha registrado la última operación de escritura.	El comando FTP store pone NEW=1 tras la edición El programa de usuario de la CPU S7 tiene que poner NEW=0 tras la lectura de los datos, para hacer posible un nuevo store o para poder borrar el archivo con el comando FTP dele.

Parámetro	Tipo	Valor / Significado	Alimentación
WRITE_ACCESS	BOOL	0: El FTP-Client del PG/PC no tiene derecho de escritura para los File-DBs de la CPU S7. 1: El FTP-Client del PG/PC tiene derecho de escritura para los File-DBs de la CPU S7.	El bit se pone a un valor de inicialización al configurar el DB. Recomendación: A ser posible, el bit debería dejarse inalterado. En caso especiales es posible una adaptación durante el funcionamiento.
ACT_LENGTH	DINT	Longitud actual del área de datos útiles. El contenido de este campo sólo es válido si EXIST = 1.	La longitud actual se actualiza después de una operación de escritura.
MAX_LENGTH	DINT	Longitud máxima del área de datos útiles (longitud total del DB menos 20 bytes del Header).	La longitud máxima se debería definir al configurar el DB. El valor puede ser modificado por el programa de usuario también durante el funcionamiento.
FTP_REPLY_CODE	INT	Este parámetro no tiene relevancia en el modo FTP-Server.	El FTP-Server lo pone a "0".
DATE_TIME	DATE_AND_TIME	Fecha y hora de la última modificación del archivo. El contenido de este campo sólo es válido si EXIST = 1.	La fecha actual se actualiza después de una operación de escritura. Si se utiliza la función "Transmisión de la hora", la entrada se corresponde con la hora transmitida. Si no se utiliza la función "Transmisión de la hora", se introduce una hora relativa. La referencia es el momento de arranque del CP IT (el valor de inicialización es 1.1.1994 0.0 horas).

Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Client

Funcionamiento

Para la transmisión de datos mediante FTP tiene que crear bloques (File-DBs) en la CPU de su estación S7. Estos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como datos transmisibles. Constan de los siguientes sectores:

- Sector 1: File-DB Header (tiene una estructura fija con una longitud de 20 byte)
- Sector 2: datos útiles (longitud y estructura variables)

File-DB Header para modo FTP-Client

Observación: El File-DB Header aquí descrito es prácticamente idéntico al File-DB Header descrito para el modo Server. Las diferencias residen en los parámetros:

- WRITE_ACCESS
- FTP_REPLY_CODE

Parámetro	Tipo	Valor / Significado	Alimentación
EXIST	BOOL	El bit EXIST indica si hay datos válidos en el área de datos útiles. El comando FTP retrieve sólo procesa la petición si EXIST=1. • 0: El File-DB no contiene datos útiles válidos ("Archivo no existe"). • 1: El File-DB contiene datos útiles válidos ("Archivo existe").	El comando FTP dele pone EXIST=0; El comando FTP store pone EXIST=1;
LOCKED	BOOL	El bit LOCKED sirve para la protección de acceso al File-DB. • 0: Se puede acceder al File-DB. • 1: El File-DB está bloqueado.	Los comandos FTP store y retr ponen LOCKED=1 durante la edición. Para una operación de escritura desde el programa de usuario rige: El programa de usuario almacenado en la CPU S7 puede poner o reponer LOCKED para asegurar la coherencia durante un acceso de escritura. Procedimientom recomendado para el programa de usuario: 1. Comprobar el bit LOCKED; si =0 2. poner WRITEACCESS-Bit=0 3. Comprobar el bit LOCKED; si =0 4. poner LOCKED-Bit=1 5. Escribir datos 6. poner LOCKED-Bit=0

Parámetro	Tipo	Valor / Significado	Alimentación
NEW	BOOL	El bit NEW-Bit informa sobre si los datos se han modificado desde la última operación de lectura. • 0: El contenido del File-DB no ha cambiado desde la última operación de escritura. El programa de usuario de la CPU S7 ha registrado el último cambio. • 1: El programa de usuario de la CPU S7 no ha registrado la última operación de escritura.	El comando FTP store pone NEW=1 tras la edición El programa de usuario de la CPU S7 debe activar NEW=0 tras la lectura de los datos para permitir un nuevo comando retr.
WRITE_ACCESS	BOOL	0: El programa de usuario (bloques FTP-Client) tiene derecho de escritura para los File-DBs de la CPU S7. 1: El programa de usuario (bloques FTP-Client) no tiene derecho de escritura para los File-DBs de la CPU S7.	El bit se pone a un valor de inicialización al configurar el DB. Recomendación: A ser posible, el bit debería dejarse inalterado. En caso especiales es posible una adaptación durante el funcionamiento.
ACT_LENGTH	DINT	Longitud actual del área de datos útiles. El contenido de este campo sólo es válido si EXIST = 1.	La longitud actual se actualiza después de una operación de escritura.
MAX_LENGTH	DINT	Longitud máxima del área de datos útiles (longitud total del DB menos 20 bytes del Header).	La longitud máxima se debería definir al configurar el DB. El valor puede ser modificado por el programa de usuario también durante el funcionamiento.
FTP_REPLY_CODE	INT	Cifra sin signo (16 bit) que contiene el último código Reply de FTP como valor binario. El contenido de este campo sólo es válido si EXIST = 1.	Es actualizado por el FTP-Client al editar comandos FTP.
DATE_TIME	DATE_AND_TIME	Fecha y hora de la última modificación del archivo. El contenido de este campo sólo es válido si EXIST = 1.	La fecha actual se actualiza después de una operación de escritura. Si se utiliza la función "Transmisión de la hora", la entrada se corresponde con la hora transmitida. Si no se utiliza la función "Transmisión de la hora", se introduce una hora relativa. La referencia es el momento de arranque del CP IT (el valor de inicialización es 1.1.1994 0.0 horas).

Bloque de datos FILE_DB_HEADER como plantilla

Significado

Para la creación de un encabezado del File-DB está predeterminado el tipo de datos FILE_DB_HEADER.

Funcionamiento

Para la transmisión de datos mediante FTP hay que crear bloques de datos (File-DBs) en la CPU de la estación S7. Estos bloques de datos han de tener una determinada estructura para que los servicios FTP puedan tratarlos como datos transmisibles. Constan de los siguientes sectores:

- Sector 1: encabezado del File-DB (tiene una estructura fija con una longitud de 20 byte)
- Sector 2: datos de usuario (longitud y estructura variables)

Proceda del siguiente modo:

1. Cree un bloque de datos del tipo "DB global" en la CPU de PLC en la que se crea el programa de usuario con las instrucciones FTP.
2. Seleccione la fila que va a utilizar como línea de inicio para el File-DB.
3. En la lista desplegable de la columna "Tipo de datos", seleccione un elemento de estructura del tipo FILE_DB_HEADER.

Resultado: Se crea una estructura de datos con la estructura de encabezado necesaria para el File-DB.

Nota

Función "Agregar nuevo bloque" - Selección de tipo

Al crear nuevos bloques de datos, en la entada "Tipo" de la lista desplegable también se ofrece el tipo de bloque "FILE_DB_HEADER". ¡No utilice esta opción! El bloque de datos creado de esa manera solo obtiene la estructura de encabezado y no puede ampliarse con el área necesaria para depositar datos de usuario.

Bloque de datos FILE_DB_HEADER - Ejemplo y plantilla para el encabezado del File-DB

En la vista de declaración se reconoce la siguiente estructura:

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	bit08	BOOL	FALSE	reservado
+0.1	bit09	BOOL	FALSE	reservado
+0.2	bit10	BOOL	FALSE	reservado
+0.3	bit11	BOOL	FALSE	reservado
+0.4	bit12	BOOL	FALSE	reservado

Dirección	Nombre	Tipo	Valor inicial	Comentario
+0.5	bit13	BOOL	FALSE	reservado
+0.6	bit14	BOOL	FALSE	reservado
+0.7	bit15	BOOL	FALSE	reservado
+1.0	EXIST	BOOL	FALSE	if TRUE: FileDB content is valid data
+1.1	LOCKED	BOOL	FALSE	it TRUE: FileDB is locked caused by changes of the content
+1.2	NEW	BOOL	FALSE	if TRUE: FileDB content is new and may not be overwritten
+1.3	WRITE_ACCESS	BOOL	FALSE	if TRUE: Ftp-Server of the IT-CP has write access, else Ftp-Server
+1.4	bit04	BOOL	FALSE	reservado
+1.5	bit05	BOOL	FALSE	reservado
+1.6	bit06	BOOL	FALSE	reservado
+1.7	bit07	BOOL	FALSE	reservado
+2.0	ACT_LENGTH	DINT	L#0	actual size of the content in bytes (not including the header of 20 bytes)
+6.0	MAX_LENGTH	DINT	L#0	max. size of the content in bytes (not including the header of 20 bytes)
+10.0	FTP_REPLY_CODE	INT	0	last reply code from the remote FTP-Server
+12.0	DATE_TIME	DATE_AND_TIME	DT#00-1-1-0:0:0.000	date and time of last change of the content of the FileDB
=20.0		END_STRUCT		

Diferencias en los modos de funcionamiento

Encabezado del File-DB para modo cliente FTP

El encabezado del File-DB aquí descrito es prácticamente idéntico para los modos de cliente y servidor FTP. Las diferencias residen en los parámetros

- WRITE_ACCESS
- FTP_REPLY_CODE

Observe al respecto las diferentes descripciones de los parámetros:

- Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Server (Página 8177)
- Estructura de los bloques de datos (File-DB) para servicios FTP - modo FTP-Client (Página 8179)

Instrucciones para CP ERPC

LOGICAL_TRIGGER para el trigger lógico

Significado de la instrucción

Para el uso de un trigger lógico para la comunicación ERPC se dispone de la instrucción LOGICAL_TRIGGER .

Para iniciar un trigger lógico, llame la instrucción LOGICAL_TRIGGER en el OB1 del programa de usuario de la CPU.

Para llamar la instrucción LOGICAL_TRIGGER se requieren otros bloques adicionales:

- Un DB de instancia generado automáticamente
- Un bloque de datos "CONF_DB"
  Este DB de configuración contiene los datos de configuración del trigger lógico. El DB de configuración debe crearse y configurarse en el proyecto STEP 7.
  Para llamar varios triggers lógicos es necesario disponer varios DBs de configuración.

Los números de la instrucción LOGICAL_TRIGGER y del DB de instancia se pueden modificar.

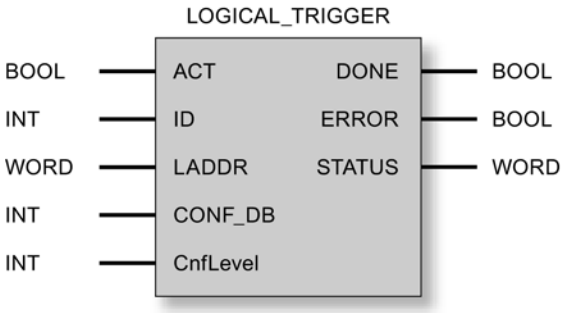
Validez

La instrucción LOGICAL_TRIGGER se puede utilizar en combinación con los siguientes tipos de módulos:

- CP 343-1 ERPC

Llamada

Interfaz de llamada en representación FUP



Consulte también

Bloque de datos de configuración (Página 8189)

Forma de trabajar LOGICAL_TRIGGER (Página 8186)

Explicación de los parámetros formales - LOGICAL_TRIGGER (Página 8186)

Indicadores de la instrucción LOGICAL_TRIGGER (Página 8188)

Forma de trabajar LOGICAL_TRIGGER

Forma de trabajar

La siguiente tabla muestra los pasos que se realizan en una llamada de trigger por parte del programa de usuario de la CPU.

Paso	Significado
1	La instrucción LOGICAL_TRIGGER se llama en el punto previsto del programa de usuario de la CPU con el correspondiente DB de instancia y el bloque de datos de configuración CONF_DB configurado. • Si la instrucción LOGICAL_TRIGGER se llama con ACT = 1, se leen los datos de trigger actuales y se envían al firmware del CP. • Si la instrucción LOGICAL_TRIGGER se llama con ACT = 0, se actualizan sólo los indicadores de estado DONE, ERROR y STATUS.
2	La instrucción LOGICAL_TRIGGER lee los datos de trigger actuales.
3	La instrucción LOGICAL_TRIGGER genera la PDU, que se envía al firmware del CP con los datos de trigger actuales.
4	El firmware del CP genera el telegrama de datos y lo transfiere a la aplicación ERPC.
5	La aplicación ERPC envía el telegrama de datos a la estación ERP (sistema ERP o MES).

Consulte también

LOGICAL_TRIGGER para el trigger lógico (Página 8184)

Explicación de los parámetros formales - LOGICAL_TRIGGER

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la interfaz de llamada de la instrucción LOGICAL_TRIGGER:

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ACT	INPUT	BOOL	0	En el caso de una llamada de la instrucción LOGICAL_TRIGGER con ACT = 0 se actualizan los indicadores de estado DONE, ERROR y STATUS.
			1	Si se llama la instrucción LOGICAL_TRIGGER con ACT = 1, los datos de trigger actuales se leen y se envían al CP.
ID	INPUT	INT		ID del trigger Este valor identifica el trigger lógico configurado en el Workbench ILS.

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP con STEP 7, aparece la dirección inicial de módulo en la tabla de configuración de HW Config. Indique aquí esta dirección.
CONF_DB	INPUT	INT		En este bloque de datos se encuentran los datos de configuración de los trigger lógicos configurados.
CnfLevel	INPUT	INT	0: Acuse de transporte 1: Acuse fin a fin	Modo de acuse El acuse correspondiente se indica en el valor STATUS de los indicadores de la instrucción LOGICAL_TRIGGER. • 0 = acuse de transporte (STATUS = 0000_H) Se notifica el éxito de la ejecución de la petición en cuanto los datos se han transferido a la aplicación ERPC. Sin embargo, ello no significa necesariamente que el telegrama de datos se haya enviado a la estación ERP (sistema ERP o MES) o que la aplicación ERPC no pueda determinar posteriormente un error. • 1 = acuse fin a fin (STATUS = 0001_H) La petición no se acusa hasta que la aplicación ERPC no ha comprobado los datos. Con la variable "TriggerResponse" del DB de configuración (DB_CONF) se notifica si la estación ERP estaba accesible o si la aplicación ERPC se encuentra en modo Store-and-Forward. El acuse fin a fin supone un tiempo mayor de ejecución de la petición que el acuse de transporte.
DONE	OUTPUT	BOOL	0: petición en curso 1: petición concluida	El parámetro indica si la petición para la transmisión del área de datos de configuración se ha desarrollado sin errores. En caso de aceptarse la petición, el CP pone DONE a 0. Mientras sea DONE = 0, no se puede activar ninguna otra petición. La tabla "Indicadores de la instrucción LOGICAL_TRIGGER" informa sobre el significado en relación con los parámetros ERROR y STATUS.
ERROR	OUTPUT	BOOL	0: - 1: Caso de error	Indicador de error La tabla "Indicadores de la instrucción LOGICAL_TRIGGER" informa sobre el significado en relación con los parámetros DONE y STATUS.
STATUS	OUTPUT	WORD	Consulte la tabla "Indicadores de la instrucción LOGICAL_TRIGGER"	Indicador de estado La tabla "Indicadores de la instrucción LOGICAL_TRIGGER" informa sobre el significado en relación con los parámetros DONE y ERROR.

Consulte también

LOGICAL_TRIGGER para el trigger lógico (Página 8184)

Indicadores de la instrucción LOGICAL_TRIGGER (Página 8188)

Indicadores de la instrucción LOGICAL_TRIGGER

Indicadores

La tabla siguiente informa sobre la indicación a evaluar por el programa de usuario, formada por DONE, ERROR y STATUS.

Tabla 9-360 Indicadores de la instrucción LOGICAL_TRIGGER

DONE	ERROR	STATUS	Significado
Indicadores sobre la ejecución de la petición			
1	0	0000_H	Petición terminada sin errores. El trigger lógico se ha concluido con éxito.
1	0	0001_H	Petición terminada sin errores. La base de datos no está disponible (modo Store-and-forward).
0	0	8181_H	Petición en curso.
0	1	7000_H	La instrucción LOGICAL_TRIGGER se ha llamado con ACT = 0. Sin embargo, la petición no se procesa. Llame la instrucción LOGICAL_TRIGGER como mínimo una vez con ACT = 1.
Indicadores sobre la configuración y el desarrollo del trigger lógico			
0	1	$80D2_H$	El CP utilizado no soporta la comunicación ERPC (tipo de CP erróneo).
0	1	8183_H	El CP utilizado no soporta la comunicación ERPC (tipo de CP erróneo).
0	1	8187_H	Estado no válido de la instrucción LOGICAL_TRIGGER (LOGICAL_TRIGGER_STATE desconocido). Vuelva a llamar la instrucción LOGICAL_TRIGGER .
0	1	$8A01_H$	La cantidad de trigger lógicos configurados es igual a 0.
0	1	$8A02_H$	Para este trigger lógico no hay ninguna configuración en el bloque de datos de configuración. Compruebe la configuración del Workbench ILS.
0	1	$8A03_H$	La estructura del bloque de datos de configuración no es correcta. El "header identifier" no tiene el valor correcto. Corrija el valor de la variable "ident" (véase el manual de producto ERPC-CP) en el DB de configuración.
0	1	$8A04_H$	La estructura del bloque de datos de configuración no es correcta. Vuelva a cargar la configuración del Workbench ILS de nuevo en el CP, vuelva a crear el o los DB de configuración y configúrelo(s) (véase el manual de producto ERPC-CP).
0	1	$8A05_H$	El bloque de datos de configuración configurado no existe en la CPU.
0	1	$8A06_H$	En una llamada sucesiva se ha llamado un trigger aún en ejecución con otro ID. Compruebe la "ID" en la instrucción LOGICAL_TRIGGER llamada.
0	1	$8A08_H$	Los datos de configuración del bloque de datos de configuración no existen, o no están completos. Si el error sólo se produce al arrancar la estación S7, la causa puede ser que los datos de configuración del trigger lógico no se han transferido todavía por completo al bloque de datos de configuración. Si el error persiste, compruebe la configuración de los símbolos ERPC.
0	1	$8A09_H$	En el bloque de datos de configuración se ha notificado un error desconocido.
0	1	$8A0A_H$	El trigger lógico no se puede iniciar porque se está cargando una nueva configuración de trigger.

DONE	ERROR	STATUS	Significado
0	1	$8A0B_H$	Error al determinar el sello de tiempo del registro de datos actual (datos de la CPU)
0	1	$8A0C_H$	El DB de configuración se ha generado con la propiedad "Unlinked". Corrija las propiedades de objeto de la instrucción LOGICAL_TRIGGER.
0	1	$8A0D_H$	Error en el parámetro de entrada CONF_DB de la instrucción LOGICAL_TRIGGER. El parámetro tiene el valor "0" o es mayor que el número de DB máximo permitido para la CPU.
0	1	$8A0E_H$	El ID de trigger transferido no está dentro del rango admisible de 1...16. Corrija el valor en la llamada de la instrucción LOGICAL_TRIGGER en el programa de usuario.
0	1	$8A0F_H$	El modo de acuse predeterminado (CnfLevel) no es válido. Corrija el valor en la llamada de la instrucción LOGICAL_TRIGGER en el programa de usuario.
0	1	$8Bxx_H$	Error al copiar valores de variables actuales en la PDU del trigger lógico. Los últimos dos dígitos (xx) proporcionan el número de la variable. Compruebe la configuración del símbolo afectado en la tabla de símbolos de la CPU y en la lista de símbolos ERPC en el diálogo de propiedades del CP.
0	1	$8C01_H$	El indicador de estado interno de la instrucción LOGICAL_TRIGGER no es válido. Vuelva a cargar la configuración del Workbench ILS de nuevo en el CP, vuelva a crear el o los DB de configuración y configúrelo(s) (véase el manual de producto ERPC-CP).
0	1	$8C02_H$	El valor de respuesta del acuse fin a fin no es válido. Vuelva a cargar la configuración del Workbench ILS de nuevo en el CP, vuelva a crear el o los DB de configuración y configúrelo(s) (véase el manual de producto ERPC-CP).
0	1	$8C03_H$	El trigger lógico contiene más de 255 variables.
0	1	$8C06_H$	Leer error en registro.
0	1	$8D03_H$	En una acción de base de datos, el firmware notifica un timeout.
0	1	$8D04_H$	La aplicación de base de datos notifica un error general en el acuse de la acción actual.
0	1	$8E01_H$	El bloque de datos de configuración configurado en la CPU no tiene el tamaño suficiente. Modifique el tamaño del bloque de datos de configuración.
0	1	$8EXX_H$	Este código de estado con valores para XX_H en el rango entre 02_H y FF_H corresponde a representaciones de una respuesta interna de trigger. Si aparecen este tipo de valores, son relevantes para el servicio técnico.

Consulte también

LOGICAL_TRIGGER para el trigger lógico (Página 8184)

Explicación de los parámetros formales - LOGICAL_TRIGGER (Página 8185)

Bloque de datos de configuración

Puesta a disposición del bloque de datos de configuración "CONF_DB"

Si se utiliza la función ERPC "Trigger lógico", es necesario crear en STEP 7 un bloque de datos (DB) para los datos de configuración del trigger lógico e indicarlo en los parámetros de llamada de la instrucción LOGICAL_TRIGGER . La instrucción LOGICAL_TRIGGER accede al CONF_DB. Para el programa de usuario, el CONF_DB no tiene mayor relevancia.

Programación del bloque de datos de configuración

Para identificar el nuevo DB creado es necesario abrir el DB y definir el "header identifier" y el tamaño de DB en las dos primeras líneas libres.

Abra el DB en STEP 7 y configure las dos primeras líneas libres con las variables "ident" y "data" del siguiente modo:

Dirección	Nombre	Tipo	Valor inicial	Comentario (*opcional*)
*)		STRUCT *)		
*)	ident	DWORD	DW#16#45525043	*header identifier*
*)	data	array[1..2048]		*Tamaño de DB (véanse las advertencias indicadas a continuación)*
*)		Byte		
*)		END_STRUCT *)		
*) Los valores son definidos por el programa				

Nota

Tamaño de DB

Para el tamaño de DB se recomienda el valor 2 048 bytes. Si durante la puesta en marcha constata que el valor no es suficiente, auméntelo. La instrucción LOGICAL_TRIGGER notifica un valor demasiado bajo con un error y con STATUS "$8A05_H$".

Consulte también

LOGICAL_TRIGGER para el trigger lógico (Página 8184)

PROFIBUS

Introducción de PROFIBUS

Indicaciones generales sobre las instrucciones para CPs PROFIBUS

Instrucciones disponibles

La siguiente lista muestra los nombres simbólicos de los bloques de programa (instrucciones) que se utilizan en el estado de suministro. Los nombres simbólicos pueden ser modificados por el usuario.

Servicio de comunicación / área de funciones	Instrucción (nombre simbólico) 1)	Biblioteca	
		CP 300	CP 400
PROFIBUS DP	DP_SEND	x	x
	DP_RECV	x	x
	DP_DIAG	x	x
	DP_CTRL	x	x
SEND / RECEIVE (Servicios de comunicación abiertos)	AG_SEND		x
	AG_RECV		x
	AG_LSEND		x
	AG_LRECV		x
Comunicación S7	BSEND		x
	BRCV		x
	PUT		x
	GET		x
	USEND		x
	URCV		x
	C_CNTRL		x

1) Observación:
Las descripciones siguientes contienen también referencias a comportamientos discrepantes en caso de versiones de instrucciones diferentes. Preste también atención a la identificación de versión de las instrucciones utilizadas por usted. Las bibliotecas globales del SIMATIC Manager instaladas con el Automation Workbench contienen las versiones de instrucciones actuales del momento de la habilitación de STEP 7.

¿Qué versión de bloque se debe utilizar?

Las descripciones siguientes contienen también referencias a comportamientos discrepantes en caso de versiones de instrucciones diferentes. Preste también atención a la identificación de versión de las instrucciones utilizadas por usted.

Nota

Se recomienda utilizar siempre las versiones de instrucciones actuales para todos los tipos de módulos.

Encontrará informaciones sobre las versiones de instrucciones actuales, así como las instrucciones actuales en nuestro Customer Support en Internet, desde donde las podrá descargar:

http://support.automation.siemens.com/WW/view/de/8797900

En el caso de tipos de módulos menos recientes, esta recomendación presupone que usted utilice la versión de firmware actual para este tipo de módulo.

Instrucciones en caso de repuesto

Se entiende por caso de sustitución en este contexto la sustitución de un módulo por otro módulo de versión eventualmente más moderna.

Nota

Tenga en cuenta que, en el caso de utilizar recambios, en el programa de usuario se deben utilizar sólo las instrucciones autorizadas para el tipo de CP configurado.

Se recomienda utilizar siempre las versiones de instrucciones actuales para todos los tipos de módulos. En el caso de tipos de módulos menos recientes, esta recomendación presupone que usted utilice la versión de firmware actual para este tipo de módulo.

Más información acerca de casos de sustitución, encontrará en nuestro Customer Support en Internet.

http://support.automation.siemens.com/WW/view/de/7806643

Los manuales del equipo informan sobre la compatibilidad de los CPs S7 y las instrucciones correspondientes.

Parametrización de llamadas de las instrucciones

Antes de pasar a describir en detalle las instrucciones, se ofrecen aquí algunas informaciones de índole general sobre la llamada y la parametrización de las instrucciones.

Aquí se pueden dar informaciones generales sobre los siguientes grupos de parámetros, existentes en todas las instrucciones:

- Parámetros para coordinación de CP y enlaces (parámetros de entrada)
- Parámetros para definir una determinada área de datos de la CPU (parámetros de entrada)
- Informaciones sobre estado/status (parámetros de salida)

Llamar instrucciones de comunicación para S7-300

ATENCIÓN
No se permite llamar las instrucciones de comunicación para S7-300 en varios niveles de proceso. Si, por ejemplo, llama una instrucción de comunicación en OB1 y en OB35, el procesamiento de la instrucción podría ser interrumpido por el OB de prioridad respectivamente superior. Si llama instrucciones en varios OBs, tiene que procurar, a través de la programación, que una instrucción de comunicación en procesamiento no sea interrumpida por otras instrucciones de comunicación (por ejemplo, por medio de las instrucciones del sistema Bloquear/Liberar alarmas).

Parámetros para definir una determinada área de datos de la CPU (parámetros de entrada)

Indicar el área de datos en la CPU

Al llamar una instrucción transmite usted la dirección y la longitud del área de datos de la CPU en la que se deben disponer o almacenar datos útiles o bien que puede contener otras informaciones relativas a la parametrización.

Para el direccionamiento de esta área se utiliza el tipo de datos del puntero ANY.

Informaciones sobre estado/status (parámetros de salida)

Evaluar indicaciones de estados

Para la evaluación del estado se tienen que evaluar estos parámetros en el programa de usuario:

- DONE o NDR
  Estos parámetros (DONE en el caso de peticiones de envío o NDR para peticiones de recepción) notifica la conclusión (con resultado positivo) de una ejecución de la petición.
- ERROR
  Notifica que la petición no se ha podido ejecutar sin errores.
- STATUS
  El parámetro proporciona información detallada sobre la ejecución de la petición. Se pueden proporcionar indicaciones de estado ya durante la ejecución de la petición (DONE=0 y ERROR=0).

Nota

Tenga en cuenta que las indicaciones de estado DONE, NDR, ERROR y STATUS se actualizan con cada llamada de la instrucción.

Indicaciones de estado al arrancar el CP

Al rearrancar el CP Ethernet, los parámetros de salida de la instrucción se resetean como sigue:

- DONE = 0
- NDR = 0
- ERROR = 0
- STATUS = 8180H ó 8181H

Instrucciones para los Servicios de comunicación abiertos (interfaz SEND/RECEIVE)

Instrucciones para enlaces FDL (interfaz SEND/RECEIVE)

Panorámica

En la interfaz SEND/RECEIVE están disponibles las siguientes instrucciones para la transmisión de datos a través de enlaces FDL configurados:

Instrucción	utilizable para [1]		Significado
S7-300	S7-400		
AG_SEND	x	x	para enviar datos
AG_RECV	x	x	para recibir datos
AG_LSEND		x	para enviar datos
AG_LRECV		x	para recibir datos

1) Observaciones sobre las instrucciones para S7-300 y S7-400

Para garantizar la compatibilidad de PROFIBUS y Industrial Ethernet en la interfaz del programa de usuario se pueden utilizar las instrucciones AG_LSEND y AG_LRECV para PROFIBUS como alternativa a AG_SEND o AG_RECV. No hay diferencias en la interfaz ni en el comportamiento. Pero en PROFIBUS, con estas instrucciones, destinadas para la transmisión de registros de datos largos en Industrial Ethernet, sólo se pueden transmitir cantidades de datos de 240 bytes, como máximo.

Condición para ello es que el tipo y la versión de las instrucciones estén autorizados para el tipo de CP utilizado.

En el caso de los CPs S7 para S7-300 se utilizan exclusivamente las instrucciones AG_SEND y AG_RECV; en Industrial Ethernet, también para la transmisión de registros de datos largos.

Los manuales del equipo informan sobre la compatibilidad de los CPs S7 y las instrucciones correspondientes. Encontrará una tabla general de las versiones de instrucciones en el historial de la documentación y los bloques.

Uso

El esquema siguiente explica la aplicación de las instrucciones AG_SEND / AG_LSEND y AG_RECV / AG_LRECV para la transferencia de datos bidireccional a través de un enlace

FDL configurado.
Para determinados tipos de enlaces se tiene que prever una cabecera (header) de petición en el área de datos del usuario.

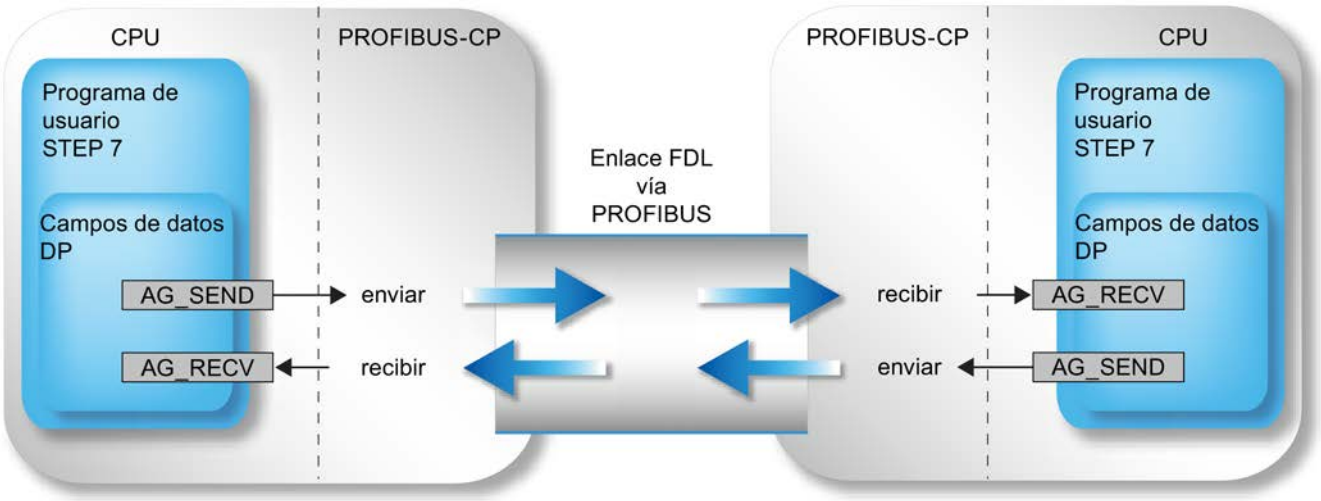


Figura 9-22 Utilizar AG_SEND y AG_RECV en ambos interlocutores de comunicación

Aplicación sin cabecera de petición

En el caso de un enlace FDL especificado, los parámetros de dirección y petición son definidos por la configuración del enlace. Por lo tanto, el programa de usuario sólo proporciona los datos útiles en el campo de datos FDL al enviar con AG_SEND / AG_LSEND, o los recibe con AG_RECV / AG_LRECV.

Pueden transmitirse hasta 240 Byte de datos útiles. Esto es válido en PROFIBUS para AG_SEND y AG_LSEND.

Aplicación con cabecera de petición

Los siguientes tipos de enlaces exigen una cabecera de petición en el área de datos (de usuario) FDL:

- Enlace FDL no especificado con acceso libre a Layer 2
- Enlace FDL con Broadcast
- Enlace FDL con Multicast

Vea en la ilustración siguiente la estructura del búfer de peticiones y el significado y el emplazamiento de los parámetros en la cabecera de la petición.

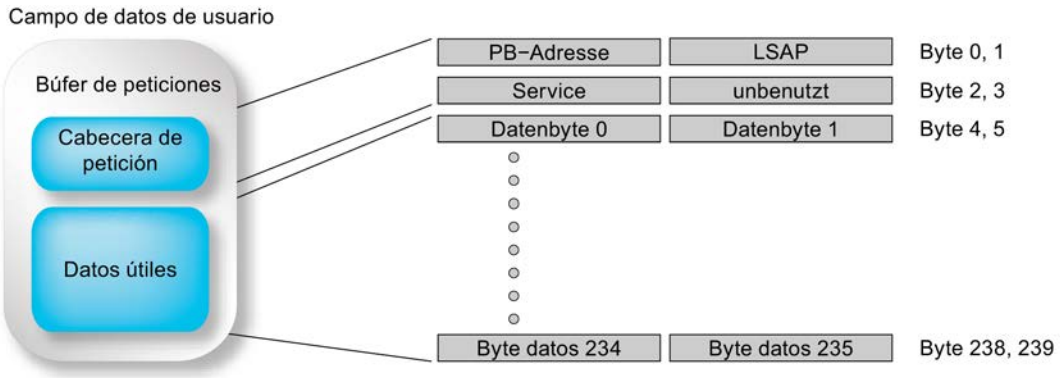


Figura 9-23 Enviar y recibir a través de un enlace FDL con direccionamiento Broadcast por programa

El campo de datos de usuario puede tener hasta 240 Byte. Pueden transmitirse hasta 236 Byte de datos útiles. 4 Byte están reservados para la cabecera de la petición.

Tenga en cuenta que la longitud de datos indicada al llamar la instrucción (parámetro LEN) ha de contener la cabecera (Header) y los datos útiles.

AG_SEND / AG_LSEND

AG_SEND / AG_LSEND (PROFIBUS)

Descripción

La instrucción AG_SEND / AG_LSEND entrega datos al CP PROFIBUS para su transmisión a través de una conexión FDL configurada.

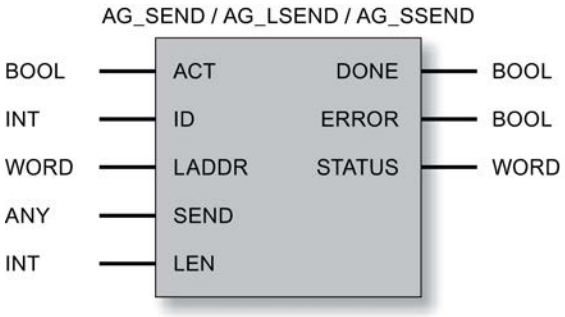
El área de datos indicada puede ser un área PA, un área de marcas o un área de bloques de datos.

Si ha sido posible enviar a través de PROFIBUS toda el área de datos de usuario, esto se señaliza para notificar la ejecución correcta.

Observación:
Todos los datos siguientes son igualmente válidos para AG_SEND y AG_LSEND, a no ser que se indique lo contrario.

Llamada

Interfaz de llamada en representación FUP



Llamadas con encabezado de petición

Vea en la tabla siguiente para qué tipos de conexiones y qué tipos de peticiones se tienen que abastecer parámetros en el encabezado de la petición.

El encabezado de petición está en el área de datos (de usuario) FDL. Allí ocupa los primeros 4 bytes y se tiene que sumar al indicar la longitud en el parámetro LEN. La longitud máxima de datos de usuario se reduce así a 236 bytes en el caso de peticiones con encabezado.

Tabla 9-361 Abastecimiento del encabezado de petición en el área de datos de usuario

Parámetro	Tipo de conexión FDL		
no especificado: Layer 2 libre [2]	Broadcast	Multicast	
Dirección PB	Dirección de la estación de destino Rango de valores: 0..126 según dispositivo / 127 para Broadcast/Multicast	para AG_SEND sin relevancia; pero el área se tiene que reservar.	para AG_SEND sin relevancia; pero el área se tiene que reservar.
LSAP	LSAP de la estación de destino Rango de valores: 0..62 según dispositivos / 63 para Broadcast	sin relevancia; pero el área se tiene que reservar.	sin relevancia; pero el área se tiene que reservar.
Service [1]	SDA (Send Data with Acknowledge): valor: 00_H SDN (Send Data with No Acknowledge): valor: 01_H	sin relevancia; pero el área se tiene que reservar.	sin relevancia; pero el área se tiene que reservar.

[1] para Broadcast y Multicast solo es posible Service SDN.

[2] Los datos relativos a Broadcast y Multicast de esta columna solo son relevantes en el caso de que se utilice una conexión FDL no especificada para Broadcast o Multicast. En el caso de una conexión FDL configurada (aplicación recomendada) con Broadcast o Multicast como interlocutor de conexión, los parámetros de dirección se asignan automáticamente de acuerdo con la configuración.

Consulte también

Parámetros en el caso de AG_SEND / AG_LSEND (PB) (Página 8198)
Parámetros DONE, ERROR, STATUS (PB) (Página 8199)
Ejemplo para AG_SEND (PB) (Página 8201)

Parámetros en el caso de AG_SEND / AG_LSEND (PB)

Explicación de los parámetros formales

La tabla representada a continuación explica los parámetros formales de la instrucción AG_SEND / AG_LSEND:

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ACT	INPUT	BOOL	0,1	En el caso de una llamada de la instrucción con ACT = 1 se envían LEN Bytes del área de datos indicada con el parámetro SEND. En el caso de una llamada de la instrucción con ACT = 0 se actualizan las indicaciones de estado DONE, ERROR y STATUS.
ID	INPUT	INT	1,2...64 (S7-400) 1,2...16 (S7-300)	En el parámetro ID se indica el número del enlace FDL.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
SEND	INPUT	ANY (como VARTYPE sólo se admite: BYTE, WORD y DWORD		Indicar la dirección y la longitud La dirección del área de datos remite como alternativa a: • Área PA • Área de marcas • Área de bloques de datos En caso de llamada con cabecera de petición, el área de datos FDL contiene la cabecera de petición y los datos útiles.
LEN	INPUT	INT	1,2,...240 (o hasta "dato de longitud en parámetro SEND")	Cantidad de bytes que se deben enviar junto con la petición desde el área de datos FDL. Este número puede estar comprendido entre 1 y "dato de longitud en parámetro SEND". En caso de llamada con cabecera de petición, la longitud de compone de la cabecera de petición (4 bytes) + datos útiles (1..236 bytes). Por lo tanto es LEN >= 4.
DONE	OUTPUT	BOOL	0: - 1: nuevos datos	El parámetro de estado indica si la petición se ha ejecutado sin errores. La tabla siguiente informa sobre el significado en relación con los parámetros ERROR y STATUS

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicación de fallo La tabla siguiente informa sobre el significado en relación con los parámetros DONE y STATUS.
STATUS	OUTPUT	WORD	Véase la tabla siguiente	Indicación de estado La tabla siguiente informa sobre el significado en relación con los parámetros DONE y ERROR.

Consulte también

AG_SEND / AG_LSEND (PROFIBUS) (Página 8195)

Parámetros DONE, ERROR, STATUS (PB)

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación 8FxxH en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-362 Códigos de condición de AG_SEND

DONE	ERROR	STATUS	Significado
1	0	0000_H	Petición terminada sin errores.
0	0	0000_H	Ninguna petición en proceso.
0	0	8181_H	Petición en curso.
0	1	7000_H	Este código es solo posible en S7-400: La instrucción se ha llamado con ACT=0; pero la petición no se procesa.
0	1	8183_H	Falta la configuración o no se ha iniciado aún el servicio FDL en el CP PROFIBUS.
0	1	8184_H	Posibles causas: • Se ha indicado un tipo de datos no permitido para el parámetro SEND. • Conexión FDL sin búfer de peticiones Error del sistema. • Conexión FDL con búfer de peticiones Parámetro LEN<4 o parámetro no permitido en el encabezado de petición (con acceso libre a Layer 2).
0	1	8185_H	Parámetro LEN mayor que el área fuente SEND.
0	1	8186_H	Parámetro ID no válido. ID != 1,2....15,16.
0	1	8301_H	SAP no activado en la estación de destino.

DONE	ERROR	STATUS	Significado
0	1	8302_H	No hay recursos de recepción en la estación de destino, la estación emisora no puede procesar con suficiente rapidez los datos recibidos o no ofrece recursos de recepción suficientes.
0	1	8303_H	El servicio PROFIBUS (SDA-Send Data with Acknowledge) no es soportado en este SAP por la estación de destino. El código de condición puede aparecer también temporalmente al cargar conexiones o transiciones de red "en RUN".
0	1	8304_H	La conexión FDL no está establecida.
0	1	8311_H	No se puede acceder a la estación de destino con la dirección PROFIBUS indicada o bien el servicio utilizado no es posible para la dirección PROFIBUS indicada.
0	1	8312_H	Error PROFIBUS en el CP: p. ej. cortocircuito, la estación propia no está en el anillo.
0	1	8315_H	Posibles causas: • Error de parámetro interno en una conexión FDL con encabezado de petición: Parámetro LEN<4 o parámetro no permitido en el encabezado de petición (con acceso libre a Layer 2). • Fallo del bus Posible significado adicional: • El indicador de error se puede presentar también en caso de perturbaciones en el bus (por ejemplo, si se producen fallos físicos debido a terminaciones de línea defectuosas o a diferencias en los ajustes de la velocidad de transmisión en los dispositivos).
0	1	$8F22_H$	Área fuente no valida, p. ej.: Área no existente en DB Parámetro LEN < 0
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Error del número DB.
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F44_H$	La dirección del parámetro a leer está inhibida en la pista de acceso.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible. p. ej., parámetro LEN = 0 .
0	1	8090_H	• No existe un módulo con esta dirección inicial de módulo. • La instrucción utilizada no es acorde con la familia de sistemas utilizada (se tienen que utilizar instrucciones diferentes para S7-300 y S7-400).
0	1	8091_H	La dirección base lógica no está en formato de palabra doble.
0	1	8092_H	En la referencia ANY se ha indicado un tipo distinto de BYTE. (solo para S7-400)
0	1	$80A4_H$	La conexión de bus K entre CPU y CP no está establecida. (en CPUs con versiones más actuales). Esto se puede deber, por ejemplo, a lo siguiente: • falta configuración de conexiones; • se supera la cantidad máxima de CPs que pueden trabajar en paralelo (ver información al respecto en el manual del CP).
0	1	$80B0_H$	El módulo no conoce el registro.

DONE	ERROR	STATUS	Significado
0	1	$80B1_H$	Área de destino no válida. El número de datos que deben enviarse sobrepasa el límite superior admisible para este servicio (p. ej. área de destino > 240 bytes).
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida (en caso de versiones de CPU menos recientes; en otro caso $80A4_H$; consulte allí más información)
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Medio de servicio (memoria) ocupado.
0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.
0	1	$80D2_H$	Dirección inicial del módulo errónea.

Consulte también

AG_SEND / AG_LSEND (PROFIBUS) (Página 8195)

Ejemplo para AG_SEND (PB)

Ejemplo de llamada y análisis de los parámetros

A continuación encontrará un ejemplo ejecutable para la llamada y el análisis de los parámetros de una instrucción AG_SEND.

La instrucción "AE_460_1" aquí elegida, en la que tiene lugar la llamada de emisión, incluye el OB100 mostrado más abajo; el OB100 ajusta correctamente el bit ACT al arrancar la CPU.

Para lograr un funcionamiento correcto ha de estar cargado un DB100 con un tamaño mínimo de 240 bytes.

El programa exige un CP en la dirección 256 así como la existencia de un enlace configurado del tipo ISO-Transport / ISO-on-TCP / TCP- o FDL con el ID=1 (hacer las adaptaciones necesarias).

```
//-----------------------------------------------------------------
---
FUNCTION FC 100: VOID
TITLE = SENDE_DEMO
AUTHOR : Tester
FAMILY : S7300
NAME : FC5_Demo
VERSION : 1.0
//-----------------------------------------------------------------
---
BEGIN
    CALL FC 5 (
    ACT        := M100.0,
    ID            := 1,
    LADDR        := W#16#100,
    SEND        := P#DB100.dbx0.0 BYTE 240,
    LEN        := 240,
    DONE        := M100.1,
    ERROR    := M100.2,
    STATUS    := MW102 );
//-----------------------------------------------------------------
---
    R M100.0;          //   Resetear parámetro ACT para todas las
    SET;               //   futuras llamadas de AG_SEND
    U M100.1;          //
    SPB done;          //   Comprobar si DONE = TRUE
    SET;               //
    U M100.2;          //
    SPB err;           //   Comprobar si ERROR = TRUE
//-----------------------------------------------------------------
---
    BEA;               //   No está puesto DONE ni ERROR; la petición
                       //   aún está en ejecución.
//-----------------------------------------------------------------
---
done:    S M100.0;     //   Petición terminada sin errores. Poner
    BEA;               //   ACT = TRUE para que la llamada siguiente
                           pueda impulsar la nueva petición.
//-----------------------------------------------------------------
---
err:    NOP 1;         //   Se ha presentado un error Aquí se puede
    NOP 1;             //   analizar la palabra de estado. Poner en
    S M100.0;          //   cualquier caso ACT en TRUE, para que se
    BEA;               //   pueda impulsar una nueva petición de
                           emisión si desaparece el error.
```

```
//-----------------------------------------------------------------
---
END_FUNCTION
ORGANIZATION_BLOCK OB100
TITLE         = Init_for_FC100
FAMILY:    S7300
NAME:        SENDE_DEMO_INIT
VERSION:    1.0
VAR_TEMP
OB1_System: array [1..20] of byte;
END_VAR
//-----------------------------------------------------------------
---
BEGIN                  //
    SET                //
    S M100.0           //   Inicialización del parámetro ACT
END_ORGANIZATION_BLOCK
```

AG_RECV / AG_LRECV

AG_RECV / AG_LRECV (PROFIBUS)

Descripción

La instrucción AG_RECV / AG_LRECV toma del CP PROFIBUS los datos transmitidos a través de una conexión FDL configurada.

El área de datos indicada para la toma de datos puede ser un área PA, un área de marcas o un área de bloques de datos.

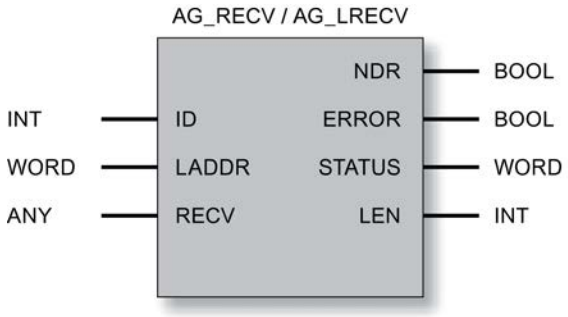
Se señala que la función ha sido ejecutada sin errores cuando se hayan podido recibir los datos del CP PROFIBUS.

Observación:

Todos los datos siguientes son igualmente válidos para AG_RECV y AG_LRECV, a no ser que se indique lo contrario.

Interfaz de llamada

Interfaz de llamada en representación FUP



Llamadas con encabezado de petición

Tabla 9-363 Parámetros de retorno en el encabezado de petición en el área de datos (de usuario) FDL.

Parámetro	Tipo de conexión FDL		
no especificado: Layer libre 2)	Broadcast	Multicast	
Dirección PB	Dirección del emisor Rango de valores: 0..126 según dispositivo		
LSAP	LSAP del emisor Rango de valores: 0..63 según dispositivo		
Service	Indicador SDN (Send Data with No Acknowledge - Indication): valor: 01_H o bien Indicador SDA (Send Data with Acknowledge - Indication): valor: 00_H	Indicador SDN (Send Data with No Acknowledge - Indication): valor: $7F_H$	Indicador SDN (Send Data with No Acknowledge - Indication): valor: $7F_H$

Consulte también

Parámetros en el caso de AG_RECV / AG_LRECV (PB) (Página 8205)

Parámetros DONE, ERROR, STATUS (PB) (Página 8206)

Parámetros en el caso de AG_RECV / AG_LRECV (PB)

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales de la instrucción AG_RECV / AG_LRECV:

Parámetro	Declaración	Tipo de datos	Valores posibles	Descripción
ID	INPUT	INT	1,2...16 (S7-300) 1,2...32 (S7-400)	En el parámetro ID se indica el número del enlace FDL.
LADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
RECV	INPUT	ANY (como VARTYPE sólo se admite: BYTE, WORD y DWORD		Indicar la dirección y la longitud La dirección del área de datos FDL remite como alternativa a: • Área PA • Área de marcas • Área de bloques de datos En caso de llamada con cabecera de petición, el área de datos FDL contiene la cabecera de petición y los datos útiles.
LEN	OUTPUT	INT	1,2,...240	Indica el número de bytes que se han tomado del CP PROFIBUS en el área de datos. En caso de llamada con cabecera de petición, la longitud de compone de la cabecera de petición (4 bytes) + datos útiles (1..236 bytes). Por lo tanto es LEN >= 4.
NDR	OUTPUT	BOOL	0: - 1: nuevos datos	Este parámetro señaliza si se han aceptado nuevos datos. La tabla siguiente informa sobre el significado en relación con los parámetros ERROR y STATUS.
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicación de fallo Para saber su significado en relación con los parámetros NDR y STATUS, véase la tabla siguiente.
STATUS	OUTPUT	WORD	Véase la tabla siguiente	Indicación de estado Para saber su significado en relación con los parámetros NDR y ERROR, véase la tabla siguiente.

Consulte también

AG_RECV / AG_LRECV (PROFIBUS) (Página 8202)

Parámetros DONE, ERROR, STATUS (PB)

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por los parámetros NDR, ERROR y STATUS, que tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-364 Códigos de condición de AG_RECV / AG_LRECV

NDR	ERROR	STATUS	Significado
1	0	0000_H	Nuevos datos aceptados.
0	0	8180_H	• Aún no hay datos. Falta la configuración o aún no se ha iniciado el servicio FDL en el CP PROFIBUS (aparece en lugar del código de condición 0,1,8183_H).
0	0	8181_H	Petición en curso.
0	1	8183_H	Falta la configuración o no se ha iniciado aún el servicio FDL en el CP PROFIBUS.
0	1	8184_H	• Se ha indicado un tipo de datos no permitido para el parámetro RECV. • Error del sistema.
0	1	8185_H	Búfer de destino (RECV) demasiado pequeño.
0	1	8186_H	Parámetro ID no válido. ID != 1,2....15,16.
0	1	8303_H	El servicio PROFIBUS (SDA-SendDatawithAcknowledge) no se soporta en este SAP. El código de condición puede aparecer también temporalmente al cargar conexiones o transiciones de red "en RUN".
0	1	8304_H	La conexión FDL no está establecida.
0	1	$8F23_H$	Área fuente no valida, p. ej.: Área no presente en DB.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos act. protegido de escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos act. protegido de escritura.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Error del número DB.
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F45_H$	La dirección del parámetro que debe escribirse está bloqueada en la pista de acceso.

NDR	ERROR	STATUS	Significado
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	1	8090_H	• No existe un módulo con esta dirección inicial de módulo. • La instrucción utilizada no es acorde con la familia de sistemas utilizada (se tienen que utilizar instrucciones diferentes para S7-300 y S7-400).
0	1	8091_H	La dirección base lógica no está en formato de palabra doble.
0	1	8092_H	En la referencia ANY se ha indicado un tipo distinto de BYTE. (solo para S7-400)
0	1	$80A0_H$	Acuse negativo al leer del módulo.
0	1	$80A4_H$	La conexión de bus K entre CPU y CP no está establecida. (en CPUs con versiones más actuales). Esto se puede deber, por ejemplo, a lo siguiente: • falta configuración de conexiones; • se supera la cantidad máxima de CPs que pueden trabajar en paralelo (ver información al respecto en el manual del CP).
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	Posibles causas: • El área de destino no es válida. • El área de destino es demasiado pequeña. • El área de destino para los datos de recepción no se ha calculado bien. Solución: realice otra llamada de recepción con el tamaño máximo del búfer de recepción. Esto sirve independientemente del tipo de conexión (Unicast/Multicast/ Broadcast) y de la familia de dispositivos (S7-300 / S7-400).
0	1	$80B2_H$	La conexión de bus K entre CPU y CP no está establecida.
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	80C3H	Medio de servicio (memoria) ocupado.
0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario).
0	1	$80D2_H$	Dirección inicial del módulo errónea.

Consulte también

AG_RECV / AG_LRECV (PROFIBUS) (Página 8202)

Instrucciones para la DP (periferia descentralizada) para S7-300

Instrucciones para el modo DP en S7-300

Panorámica

Para los modos maestro DP y esclavo DP se dispone de las siguientes instrucciones para S7-300:

Instrucción	utilizable para		Significado
	Maestro DP	Esclavo DP	
DP_SEND	X	X	para enviar datos
DP_RECV	X	X	para recibir datos
DP_DIAG	X	-	para funciones de diagnóstico desde el maestro DP
DP_CTRL	X	-	para funciones de control

Uso

La representación siguiente aclara la aplicación de las instrucciones DP_SEND y DP_RECV en el maestro DP y el esclavo DP.

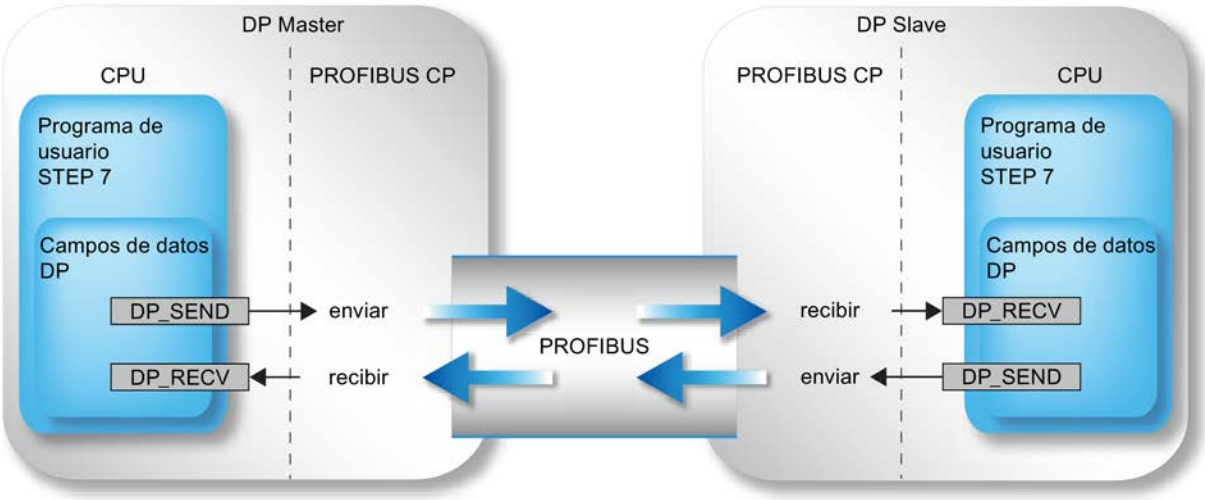


Figura 9-24 Uso de las instrucciones DP_SEND y DP_RECV en el maestro DP y el esclavo DP

DP_SEND

DP_SEND

Descripción

La instrucción DP_SEND transmite datos al CP PROFIBUS. Dependiendo del modo operatiivo del CP PROFIBUS, DP_SEND tiene el siguiente significado:

- Para uso en el maestro DP

La instrucción transfiere los datos de un área de salida DP indicada al CP PROFIBUS, para su emisión a la periferia descentralizada.

- Para uso en el esclavo DP

La instrucción transfiere los datos de entrada del esclavo DP al CP PROFIBUS, para su transmisión al maestro DP

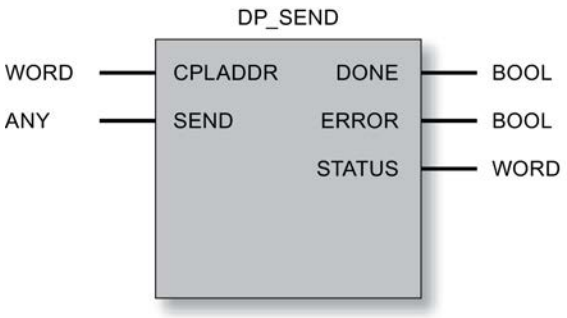
El área de datos indicada puede ser un área PA, un área de marcas o un área de bloques de datos.

Se señala que la función ha sido ejecutada sin errores cuando se haya podido recibir toda el área de datos DP del CP PROFIBUS.

Para iniciar el maestro DP hay que anteponer una llamada de DP_SEND o DP_RECV en la secuencia de llamada. Para esta primera llamada rige lo siguiente:

Si se utiliza DP_SEND para la inicialización, el área de datos transferida no se aplicará y se enviará "0" a los esclavos. Los datos de usuario transferidos no se enviarán hasta la segunda llamada del bloque.

Interfaz de llamada



Consulte también

Parámetros en el caso de DP_SEND (Página 8210)

Parámetros DONE, ERROR, STATUS (Página 8211)

Parámetros en el caso de DP_SEND

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales de la instrucción DP_SEND:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CPLADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
SEND	INPUT	ANY (como VARTYPE solo se admite: para DP_SEND a partir de V3: BYTE para DP_SEND hasta V2.x: BYTE, WORD y DWORD)		Indicar la dirección y la longitud La dirección del área de datos DP remite como alternativa a: • Área PA • Área de marcas • Área de bloques de datos Se tiene que ajustar la longitud para • Maestro DP: 1...21600 • Esclavo DP: 1...240
DONE	OUTPUT	BOOL	0: - 1: nuevos datos	El parámetro de estado indica si la petición se ha ejecutado sin errores. Consulte el significado en relación con los parámetros ERROR y STATUS en Parámetros DONE, ERROR, STATUS (Página 8211).
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Parámetros DONE, ERROR, STATUS (Página 8211)
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Parámetros DONE, ERROR, STATUS (Página 8211)

Consulte también

DP_SEND (Página 8208)

Parámetros DONE, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-365 Indicadores de DP_SEND

DONE	ERROR	STATUS	Significado
0	0	8180_H	• Arranque: El servicio DP se ha iniciado, pero todavía no es posible aplicar los datos. • Funcionamiento normal Se transfieren datos. • El DP no se ha iniciado debido a la situación siguiente: – CP STOP o – "no hay parametrización" (aparece aquí en lugar del indicador 0,1,8183_H).
1	0	0000_H	Nuevos datos transferidos sin error.
0	1	8183_H	Falta la configuración o no se ha iniciado aún el servicio DP en el CP PROFIBUS.
0	1	8184_H	Error del sistema o tipo de parámetro no permitido.
0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro.
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos act. protegido de escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos act. protegido de escritura.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Error de número de DB.
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F44_H$	La dirección del parámetro a leer está inhibida en la pista de acceso.
0	1	$8F45_H$	La dirección del parámetro que debe escribirse está bloqueada en la pista de acceso.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.

DONE	ERROR	STATUS	Significado
0	1	8090_H	No existe módulo con esta dirección.
0	1	8091_H	La dirección base lógica no está en formato de palabra doble.
0	1	$80A1_H$	Acuse negativo al escribir en el módulo.
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	La cantidad de datos a enviar supera el límite superior admisible para este servicio (válido para los modos maestro DP y esclavo DP).
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Medio de servicio (memoria) ocupado.
0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario.)
0	1	$80D2_H$	Dirección base lógica incorrecta.

Consulte también

DP_SEND (Página 8208)

DP_RECV

DP_RECV

Descripción

La instrucción DP_RECV recibe datos a través de PROFIBUS. Dependiendo del modo operatiivo del CP PROFIBUS, DP_RECV tiene el siguiente significado:

- Para uso en el maestro DP
  DP_RECV acepta los datos de proceso de la periferia descentralizada así como una información de estado en un área de entrada DP indicada.
- Para uso en esclavo DP
  DP_RECV acepta los datos de salida transmitidos por el maestro DP en el área de datos DP indicada en la instrucción.

El área de datos indicada para la toma de datos puede ser un área PA, un área de marcas o un área de bloques de datos.

Se señala que la función ha sido ejecutada sin errores cuando se haya podido transferir toda el área de entrada de datos DP del CP PROFIBUS.

Tenga en cuenta que la instrucción DP_RECV para el esclavo DP se tiene que llamar al menos una vez correctamente en el programa de usuario si se han configurado datos de salida para este esclavo DP. Observe lo dicho en el manual de producto.

Para iniciar el maestro DP hay que anteponer una llamada de DP_SEND o DP_RECV en la secuencia de llamada. Para esta primera llamada rige lo siguiente:

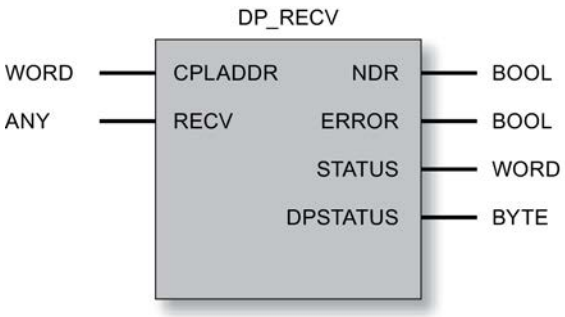
- Si se utiliza DP_RECV para la inicialización, los datos recibidos no se aplicarán. Los datos de usuario recibidos no se obtienen hasta la segunda llamada del bloque.

Tarea adicional: Entrar byte de estado

La instrucción DP_RECV tiene la siguiente tarea adicional:

- Actualizar el byte de estado DPSTATUS. DP_RECV asume con esto tareas para el diagnóstico DP.
  Si no se han configurado datos de recepción, para la actualización del byte de estado DPSTATUS se tiene que llamar DP_RECV con la longitud 1 (esto es válido solo para el maestro DP).
  Observe también lo dicho en el manual de producto.
- Habilitación de la lista de estaciones (véase DP_DIAG (Página 8218)).

Interfaz de llamada



Consulte también

Parámetros en el caso de DP_RECV (Página 8214)

Parámetros NDR, ERROR, STATUS (Página 8215)

Parámetro DPSTATUS - DP_RECV (Página 8216)

Parámetros en el caso de DP_RECV

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción DP_RECV:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CPLADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
RECV	INPUT	ANY (como VARTYPE solo se admite: para DP_RECV a partir de V3: BYTE para DP_RECV hasta V2.x: BYTE, WORD y DWORD)		Indicar la dirección y la longitud La dirección del área de datos DP remite como alternativa a: • Área PA • Área de marcas • Área de bloques de datos Se tiene que ajustar la longitud para: • Maestro DP: 1...2160 • Esclavo DP: 1...240 • Maestro DP; solo leer byte de estado: 1 (véase también el manual de producto del CP)
NDR	OUTPUT	BOOL	0: - 1: nuevos datos aceptados	El parámetro de estado indica si se han adoptado nuevos datos. Consulte el significado en relación con los parámetros ERROR y STATUS en Parámetros NDR, ERROR, STATUS (Página 8215).
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros NDR y STATUS en Parámetros NDR, ERROR, STATUS (Página 8215)
STATUS	OUTPUT	WORD		Indicador de estado Consulte el significado en relación con los parámetros NDR y ERROR en Parámetros NDR, ERROR, STATUS (Página 8215)
DPSTATUS	OUTPUT	Byte	Veáse la codificación a continuación, en DPSTATUS	Indicador de estado DP

Consulte también

DP_RECV (Página 8211)

Parámetros NDR, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por los parámetros NDR, ERROR y STATUS, que tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

NDR	ERROR	STATUS	Significado
0	0	8180_H	• Arranque: El servicio DP se ha iniciado, pero todavía no es posible aplicar los datos. • Funcionamiento normal Se transfieren datos. • El DP no se ha iniciado debido a la situación siguiente: – CP STOP o – "no hay parametrización" (aparece aquí en lugar del indicador $0,1,8183_H$).
1	0	0000_H	Nuevos datos aceptados sin error.
0	1	8183_H	Falta la configuración o no se ha iniciado aún el servicio DP en el CP PROFIBUS.
0	1	8184_H	Error del sistema o tipo de parámetro no permitido.
0	1	$8F22_H$	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	Error de área al leer un parámetro.
0	1	$8F25_H$	Error de área al escribir un parámetro.
0	1	$8F28_H$	Error de alineación al leer un parámetro.
0	1	$8F29_H$	Error de alineación al escribir un parámetro.
0	1	$8F30_H$	El parámetro está en el 1er. bloque de datos act. protegido de escritura.
0	1	$8F31_H$	El parámetro está en el segundo bloque de datos act. protegido de escritura.
0	1	$8F32_H$	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	Error de número de DB.
0	1	$8F3A_H$	Área de destino no cargada (DB).
0	1	$8F42_H$	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F43_H$	Retardo en acuse al escribir un parámetro en el área de periferia.
0	1	$8F44_H$	La dirección del parámetro a leer está inhibida en la pista de acceso.
0	1	$8F45_H$	La dirección del parámetro que debe escribirse está bloqueada en la pista de acceso.
0	1	$8F7F_H$	Error interno, p. ej. referencia ANY no admisible.
0	1	8090_H	No existe módulo con esta dirección.

NDR	ERROR	STATUS	Significado
0	1	8091_H	La dirección base lógica no está en formato de palabra doble.
0	1	$80A0_H$	Acuse negativo al leer del módulo.
0	1	$80B0_H$	El módulo no conoce el registro.
0	1	$80B1_H$	La cantidad de datos a enviar supera el límite superior admisible para este servicio (válido para los modos maestro DP y esclavo DP).
0	1	$80C0_H$	No se puede leer el registro.
0	1	$80C1_H$	El registro indicado está siendo procesado.
0	1	$80C2_H$	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	Medio de servicio (memoria) ocupado.
0	1	$80C4_H$	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario).
0	1	$80D2_H$	Dirección base lógica incorrecta.

Consulte también

DP_RECV (Página 8211)

Parámetro DPSTATUS - DP_RECV

DPSTATUS

La codificación del parámetro de salida DP-STATUS es diferente para los modos maestro DP y esclavo DP.

Modo maestro DP

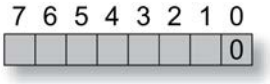


Tabla 9-366 Significado de los bits en el DPSTATUS en el modo de maestro DP

Bit	Significado
7	libre
6	Este bit no se pone. Observe también lo dicho en el manual del equipo.
5,4	Valores para DPSTATUS del maestro DP: 00 RUN 01 CLEAR 10 STOP (se representa en el estado operativo OFFLINE) 11 OFFLINE Observe también lo dicho en el manual del equipo.
3	Valor 1: la sincronización horaria está conectada
2	Valor 0: no hay nuevos datos de diagnóstico Valor 1: es conveniente evaluar la lista de diagnóstico; al menos una estación tiene datos de diagnóstico nuevos

Bit	Significado
1	Valor 0: todos los esclavos DP están en la fase de transferencia de datos Valor 1: es conveniente evaluar la lista de estaciones
0	Modo DP Valor 0: Modo de maestro DP Los demás bits sólo son válidos con el significado indicado si no está puesto este bit.

Modo esclavo DP

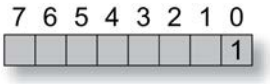


Tabla 9-367 Significado de los bits en el DPSTATUS en el modo de esclavo DP

Bit	Significado
7-5	libre
4	Este bit no se pone. Observe también lo dicho en el manual del equipo.
3	Este bit no se pone. Observe también lo dicho en el manual del equipo.
2	Valor 1: El maestro DP 1 se encuentra en el estado CLEAR. El esclavo DP recibe en los datos DP destinados a las salidas para todos los datos el valor 0. No hay ninguna influencia sobre los datos de emisión.
1	Valor 1: la configuración / parametrización aún no ha terminado con éxito.
0	Valor 1: modo de esclavo DP. Los demás bits sólo son válidos con el significado indicado si está puesto este bit.

Nota

Tenga en cuenta que sólo se debe evaluar DPSTATUS una vez esté puesto el parámetro de retorno NDR=1.

Consulte también

DP_RECV (Página 8211)

DP_DIAG

DP_DIAG

Descripción

La instrucción DP_DIAG se utiliza para solicitar informaciones de diagnóstico. Se distinguen los siguientes tipos de peticiones:

- Solicitar lista de estaciones DP;
- Solicitar lista de diagnóstico DP;
- Solicitar estado individual DP;
- Leer acíclicamente datos de entrada / salida de un esclavo DP;
- Leer diagnóstico individual DP menos reciente;
- Leer el estado operativo de DP.
- Leer el estado operativo de DP para AG-/CP-STOP;
- Leer el estado actual del esclavo DP.

Se pueden solicitar datos de diagnóstico indicando una dirección de estación específica del esclavo.

Para la transferencia de los datos de diagnóstico a la CPU se tiene que prever en la CPU un área de memoria que se indicará al hacer la llamada. Esta área de memoria puede ser un área de bloques de datos o un área de marcas. En la petición se tiene que indicar la longitud máxima del área de memoria disponible.

Nota

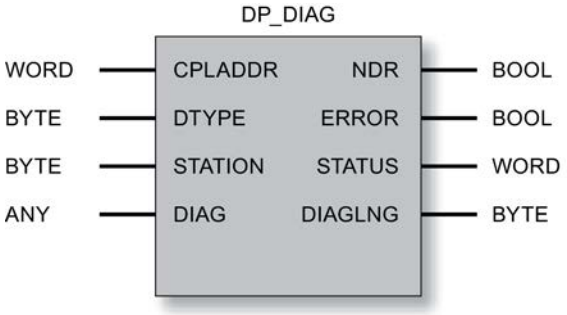
La instrucción DP_DIAG solo es útil para el modo operativo de DP con maestro DP.

Exclusión

Mientras esté en marcha la instrucción, no se debe abastecer la misma con nuevos datos de petición.

Excepción: solicitar lista de estaciones DP o lista de diagnóstico DP.

Interfaz de llamada



Consulte también

Parámetros en el caso de DP_DIAG (Página 8220)
Tipos de petición de DP_DIAG (Página 8221)
Búfer de anillo para datos de diagnóstico - DP_DIAG (Página 8223)
Parámetros NDR, ERROR, STATUS (Página 8224)

Parámetros en el caso de DP_DIAG

Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción DP_DIAG:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CPLADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
DTYPE	INPUT	BYTE	0: Lista de estaciones 1: Lista de diagnóstico 2: Diagnóstico actual 3: Diagnóstico antiguo 4: Leer estado operativo 5: Leer estado operativo para CPU-STOP 6: Leer estado operativo para CP-STOP 7: Leer datos de entrada (acíclicamente) 8: Leer datos de salida (acíclicamente) 10: Leer el estado actual del esclavo DP	Tipo de diagnóstico
STATION	INPUT	BYTE		Dirección de estación del esclavo DP
DIAG	INPUT	ANY (como VARTYPE solo se admite: BYTE, WORD y DWORD)	Se tiene que ajustar la longitud 1...240	Indicar la dirección y la longitud Dirección del área de datos. Remite como alternativa a: • Área PA • Área de marcas • Área de bloques de datos
NDR	OUTPUT	BOOL	0: - 1: nuevos datos	Este parámetro de estado señaliza si se han aceptado nuevos datos. Consulte el significado en relación con los parámetros ERROR y STATUS en Parámetros NDR, ERROR, STATUS (Página 8224)
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros NDR y STATUS en Parámetros NDR, ERROR, STATUS (Página 8224)

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
STATUS	OUTPUT	WORD	véase la lista	Indicador de estado Consulte el significado en relación con los parámetros NDR y ERROR en Parámetros NDR, ERROR, STATUS (Página 8224)
DIAGLNG	OUTPUT	BYTE	véase la lista	Contiene la longitud real (en byte) de los datos proporcionados por el CP PROFIBUS, con independencia del tamaño de búfer indicado en el parámetro DIAG. Para los tipos de petición con DTYPE 4, 5 y 6 rige: A DIAGLNG aquí siempre se le asigna el valor "1". En estos casos el valor depositado en el parámetro DIAG no es relevante para la valoración. El valor relevante está contenido en el parámetro STATUS.

Consulte también

DP_DIAG (Página 8217)

Tipos de petición de DP_DIAG

Tipos de petición

Las formas de petición admisibles y convenientes resultan, según la relación siguiente, de lo indicado para DTYPE, STATION y DIAGLNG.

Tabla 9-368 Tipos de petición para DP_DIAG

DTYPE	Equivale a Petición	Parámetro STATION	DIAGLNG	Código de confirmación (contenido en el parámetro STATUS; indicado en la tabla "Códigos de condición DP_DIAG")
0	Leer lista de estaciones DP.	---	- no se tiene en cuenta -	Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
1	Leer lista de diagnóstico DP	---	- no se tiene en cuenta -	Con la lista de diagnóstico DP se obtiene en el programa de la CPU la información sobre en qué esclavos DP hay nuevos datos de diagnóstico.
2	Leer diagnóstico individual DP actual	1...126	>=6	Con el diagnóstico individual DP actual se obtienen en el programa de la CPU los datos de diagnóstico actuales de un esclavo DP.

DTYPE	Equivale a Petición	Parámetro STATION	DIAGLNG	Código de confirmación (contenido en el parámetro STATUS; indicado en la tabla "Códigos de condición DP_DIAG")
3	Leer diagnóstico individual DP menos reciente	1...126	>=6	Con el diagnóstico individual DP menos reciente se obtienen en el programa de la CPU los datos de diagnóstico menos recientes de un esclavo DP. Estos datos están almacenados en el CP PROFIBUS y se leen en el búfer de anillo según el principio "Last in first out". Véase más abajo la explicación relativa al búfer de anillo. En caso de cambios rápidos en los datos de diagnóstico de esclavos DP se pueden registrar así estos esclavos DP para ser evaluados en el programa de la CPU del maestro DP.
4	Leer el estado operativo solicitado con la petición DP-CTRL (CTYPE=4)		=1	Con esta petición se puede leer el estado operativo del DP, ajustado previamente con la petición DP-CTRL (CTYPE=4). Observación: El estado operativo leído no tiene que coincidir necesariamente con el estado operativo actual del CP. Son posibles los siguientes estados operativos: : • RUN • CLEAR • STOP (se representa en el estado operativo OFFLINE) *) • OFFLINE
5	Leer el estado operativo de DP para CPU-STOP		=1	Con esta petición se determina a qué estado operativo DP pasa el CP PROFIBUS en caso de CPU-STOP: • RUN • CLEAR • STOP (se representa en el estado operativo OFFLINE) *) • OFFLINE Como valor predeterminado (DEFAULT) con CPU-STOP, el CP PROFIBUS pasa al estado operativo de DP CLEAR.
6	Leer el estado operativo de DP para CP-STOP		=1	Con esta petición se determina a qué estado operativo DP pasa el CP PROFIBUS en caso de CP-STOP: • STOP (se representa en el estado operativo OFFLINE) *) • OFFLINE Como valor predeterminado (DEFAULT) con CP-STOP, el CP PROFIBUS pasa al estado operativo de DP OFFLINE.
7	Leer datos de entrada	1...126	>=1	Con esta petición se leen, como maestro DP (clase 2), los datos de entrada de un esclavo DP. Este proceso recibe también el nombre de "Shared Input".

DTYPE	Equivale a Petición	Parámetro STATION	DIAGLNG	Código de confirmación (contenido en el parámetro STATUS; indicado en la tabla "Códigos de condición DP_DIAG")
8	Leer datos de salida	1...126	>=1	Con esta petición se leen, como maestro DP (clase 2), los datos de salida de un esclavo DP. Este proceso recibe también el nombre de "Shared Output".
10	Leer el estado actual del esclavo DP	1...126	>=0	Con esta petición se lee el estado operativo actual del esclavo DP. Son posibles los siguientes estados: • El maestro DP intercambia cíclicamente datos con el esclavo DP. • El maestro DP lee cíclicamente los datos de entrada del esclavo DP. • El maestro DP lee cíclicamente los datos de salida del esclavo DP. • El maestro DP no procesa cíclicamente este esclavo DP de momento.

*) El estado operativo STOP ya no es soportado en los módulos actuales (a partir del tipo de módulo DA02).

Consulte también

DP_DIAG (Página 8217)

Búfer de anillo para datos de diagnóstico - DP_DIAG

Búfer de anillo para datos de diagnóstico

La representación siguiente muestra el principio de lectura para el tipo de petición "Leer diagnóstico individual DP menos reciente". Con el primer acceso de lectura se lee el **diagnóstico antiguo más reciente**.

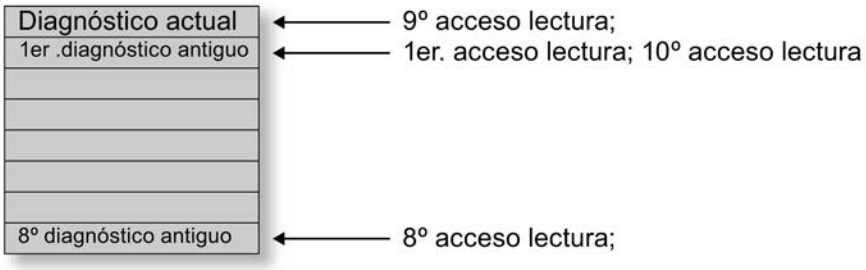


Figura 9-25 Búfer de anillo para datos de diagnóstico

Al leer un diagnóstico actual, el puntero de lectura se repone al primer diagnóstico antiguo.

Consulte también

DP_DIAG (Página 8217)

Parámetros NDR, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por los parámetros NDR, ERROR y STATUS, que tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-369 Códigos de condición DP_DIAG

NDR	ERROR	STATUS	Posible para DTYPE	Significado
0	0	8181_H	2-10	Petición en curso. El maestro DP no se ha iniciado por: • CP STOP o • no hay parametrización (aparece aquí en lugar del código de condición 0,1,8183_H).
0	0	8182_H	0	Activación no conveniente. El maestro DP no se ha iniciado por: • CP STOP o • no hay parametrización (aparece aquí en lugar del código de condición 0,1,8183_H).
0	0	8182_H	1	No hay ningún diagnóstico nuevo. El maestro DP no se ha iniciado por: • CP STOP o • no hay parametrización (aparece aquí en lugar del código de condición 0,1,8183_H).
1	0	0000_H	0, 1 y 4-9	Petición concluida sin errores. Nota: En el caso de DTYPE 2,3 y 10, la ejecución sin errores se señaliza con un código de estado distinto de "0". En correspondencia, a continuación encontrará códigos de estado detallados (para una ejecución sin errores) para el área: 82XXH En caso de ejecución con errores, obtendrá códigos de estado en las siguientes áreas: 80XXH, 83XXH, 8FXXH

NDR	ERROR	STATUS	Posible para DTYPE	Significado
1	0	8222_H	7,8	Petición terminada sin errores. La longitud de datos de esclavo DP leídos es distinta de la longitud de datos que el maestro DP espera sobre la base de la lista de módulos del esclavo DP en la base de datos del CP.
1	0	8227_H	7,8	Petición terminada sin errores. Advertencia: no hay datos.
1	0	8231_H	4,5,6	Petición terminada sin errores. Advertencia: el estado operativo de DP ya es "RUN"
1	0	8232_H	4,5,6	Petición terminada sin errores. Advertencia: el estado operativo de DP ya es "CLEAR"
1	0	8233_H	4,5,6	Petición terminada sin errores. Advertencia: el estado operativo de DP ya es "STOP" El estado STOP se representa en el estado OFFLINE (aquí, código de condición 8234_H). Observe también lo dicho en el manual de producto.
1	0	8234_H	4,5,6	Petición terminada sin errores. Advertencia: el estado operativo de DP ya es "OFFLINE"
1	0	$823A_H$	2,3,7,8	Petición terminada sin errores. Advertencia: se han leído 241 o 242 bytes de datos. Se proporcionan 240 bytes.
1	0	8241_H	2,3,10	Petición terminada sin errores. Advertencia: el esclavo DP indicado no se ha configurado.
1	0	8243_H	2,3,10	Petición terminada sin errores. Advertencia: la lista de módulos del esclavo CP en la base de datos del CP solo contiene módulos vacíos.
1	0	8245_H	2,3,10	Petición terminada sin errores. Advertencia: el esclavo DP se encuentra en el estado "Leer cíclicamente datos de entrada".
1	0	8246_H	2,3,10	Petición terminada sin errores. Advertencia: el esclavo DP se encuentra en el estado "Leer cíclicamente datos de salida".
1	0	8248_H	2,3,10	Petición concluida sin errores. Observación: esta es la indicación predeterminada para los tipos de diagnóstico mencionados, si no existe ninguna peculiaridad a señalizar.
1	0	8249_H	2,3,10	Petición terminada sin errores. Advertencia: el esclavo DP está desactivado debido a un cambio del estado operativo de DP (p. ej. interruptor del CP en STOP).
1	0	$824A_H$	2,3,10	Petición terminada sin errores. Advertencia: el esclavo DP está desactivado debido a una petición de DP_CTRL en el programa de la CPU.
0	1	8090_H	0-10	La dirección base lógica del módulo no es válida
0	1	$80B0_H$	0-10	El módulo no conoce el registro o se encuentra en la transición RUN --> STOP.
0	1	$80B1_H$	0-10	La longitud de registro indicada es incorrecta
0	1	$80C0_H$	0-10	No se puede leer el registro
0	1	$80C1_H$	0-10	El registro indicado está siendo procesado.
0	1	$80C2_H$	0-10	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$	0-8	Medios de operación (memoria) ocupados

NDR	ERROR	STATUS	Posible para DTYPE	Significado
0	1	$80C4_H$	0-10	Error de comunicación
0	1	$80D2_H$	0-10	Dirección base lógica incorrecta
0	1	8183_H	0-10	Maestro DP no configurado.
0	1	8184_H	0-8	Error del sistema o tipo de parámetro no permitido.
0	1	8311_H	>=2	Parámetro DTYPE fuera del rango de valores.
0	1	8313_H	2,3,7,8,10	Parámetro STATION fuera del rango de valores.
0	1	8321_H	>=2	El esclavo DP no proporciona datos válidos.
0	1	8326_H	7,8	El esclavo DP proporciona más de 242 bytes de datos. El CP PROFIBUS solo da soporte a 242 bytes como máximo.
0	1	8335_H	7,8	El CP PROFIBUS se encuentra en el estado PROFIBUS: "Estación no en el anillo".
0	1	8341_H	2,3,7,8,10	El esclavo indicado no se ha configurado.
0	1	8342_H	7,8	No se puede acceder al esclavo DP con la dirección PROFIBUS indicada en el parámetro STATION.
0	1	8349_H	7,8	El maestro DP se encuentra en el estado OFFLINE.
0	1	$8F22_H$	0-10	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	0-10	Error de longitud de área al escribir un parámetro (p. ej. DB demasiado corto).
0	1	$8F24_H$	0-10	Error de área al leer un parámetro
0	1	$8F25_H$	0-10	Error de área al escribir un parámetro
0	1	$8F28_H$	0-10	Error de alineación al leer un parámetro
0	1	$8F29_H$	0-10	Error de alineación al escribir un parámetro
0	1	$8F30_H$	0-10	El parámetro está en el 1er. bloque de datos act. protegido de escritura.
0	1	$8F31_H$	0-10	El parámetro está en el segundo bloque de datos act. protegido de escritura.
0	1	$8F32_H$	0-10	El parámetro contiene número de DB demasiado grande
0	1	$8F33_H$	0-10	Error de número de DB
0	1	$8F3A_H$	0-10	Área de destino no cargada (DB)
0	1	$8F42_H$	0-10	Retardo de acuse al leer un parámetro del área de periferia
0	1	$8F43_H$	0-10	Retardo de acuse al escribir un parámetro en el área de periferia
0	1	$8F44_H$	0-10	La dirección del parámetro a leer está inhibida en la pista de acceso.
0	1	$8F45_H$	0-10	La dirección del parámetro a escribir está inhibida en la pista de acceso.
0	1	$8F7F_H$	0-10	Error interno, p. ej. referencia ANY no admisible.

Consulte también

DP_DIAG (Página 8217)

DP_CTRL

DP_CTRL

Descripción

La instrucción DP_CTRL transfiere peticiones de control al CP PROFIBUS. Con la indicación de un bloque de petición (parámetro CONTROL) se especifica con mayor detalle la petición de control.

Se distinguen los siguientes tipos de peticiones:

- Global Control acíclico / cíclico;
- Borrar diagnósticos antiguos;
- Poner el estado operativo actual de DP;
- Poner el estado operativo de DP para AG-/CP-STOP;
- Leer cíclicamente datos de entrada/salida;
- Poner el modo de procesamiento del esclavo DP

Existen restricciones en cuanto a los tipos de peticiones aquí citados (tenga en cuenta al respecto lo dicho en el manual de producto).

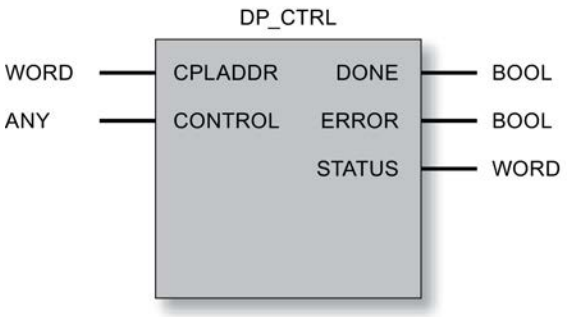
Nota

La instrucción DP_CTRL solo es útil para el modo operativo de DP con maestro DP.

Conexión

Mientras esté en marcha la instrucción, no se debe abastecer la misma con nuevos datos de petición.

Interfaz de llamada



Consulte también

Parámetros en el caso de DP_CTRL (Página 8228)
Tipos de petición de DP_CTRL (Página 8230)
Command Mode y Group Select - DP_CTRL (Página 8233)
Parámetros DONE, ERROR, STATUS (Página 8234)

Parámetros en el caso de DP_CTRL

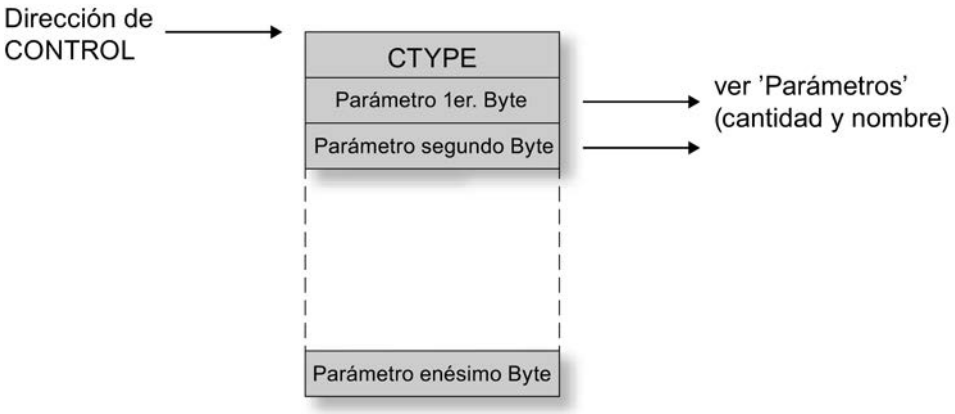
Explicación de los parámetros formales

La tabla representada a continuación explica todos los parámetros formales para la instrucción DP_CTRL:

Parámetro	Declaración	Tipo de datos	Rango de valores	Descripción
CPLADDR	INPUT	WORD		Dirección inicial del módulo Al configurar el CP aparece la dirección inicial de módulo en la tabla de configuración. Indique aquí esta dirección.
CONTROL	INPUT	ANY (como VARTYPE solo se admite: BYTE, WORD y DWORD)	Se tiene que ajustar la longitud 1...240	Indicar la dirección y la longitud del bloque de petición CONTROL Dirección del área de datos. Remite como alternativa a: • Área PA • Área de marcas • Área de bloques de datos La longitud tiene que ser al menos tan grande como la cantidad de parámetros.
DONE	OUTPUT	BOOL	0: - 1: Petición ejecutada sin error	Indica si la petición se ha enviado y se ha concluido sin errores. Consulte el significado en relación con los parámetros ERROR y STATUS en Parámetros DONE, ERROR, STATUS (Página 8234)
ERROR	OUTPUT	BOOL	0: - 1: caso de error	Indicador de error Consulte el significado en relación con los parámetros DONE y STATUS en Parámetros DONE, ERROR, STATUS (Página 8234)
STATUS	OUTPUT	WORD	Véase la tabla siguiente 'Indicadores'	Indicador de estado Consulte el significado en relación con los parámetros DONE y ERROR en Parámetros DONE, ERROR, STATUS (Página 8234)

Estructura del bloque de petición CONTROL

La petición de control tiene la siguiente estructura:



Ejemplo del bloque de petición

Con un bloque de petición conforme al modelo abajo mostrado se emite una petición cíclica de Global Control SYNC y UNFREEZE para los grupos 4 y 5 sin la opción Autoclear.

DB 14

Byte 0	01H	CTYPE
Byte 1	24H	Command Mode
Byte 2	18H	Group Select
Byte 3	00H	Autoclear

La longitud indicada en el puntero ANY ha de ser al menos 4 (para el ejemplo de petición se han seleccionado 30).

Consulte también

DP_CTRL (Página 8226)

Tipos de petición de DP_CTRL

Tipos de petición

Las formas de petición admisibles y convenientes resultan, según la relación siguiente, de lo indicado para CTYPE así como de lo indicado en el bloque de petición.

CTYPE	Equivale a Petición	Parámetros en el bloque de petición		Descripción
Nombre	Número			
0	Impulsar Global Control	1. byte: Command Mode 2º byte: Group Select (ver a continuación de esta tabla)	2	Se envía una única petición de Global Control a los esclavos DP seleccionados con Group Select. En el parámetro Command Mode se determinan las siguientes peticiones de Global Control: • SYNC • UNSYNC • FREEZE • UNFREEZE • CLEAR no es compatible (observe también lo dicho en el manual de producto). Es posible indicar más de una petición en el parámetro Command Mode.

CTYPE	Equivale a Petición	Parámetros en el bloque de petición		Descripción
1*)	Impulsar cíclicamente Global Control	1. byte: Command Mode 2º byte: Group Select 3er. byte: Autoclear (ver a continuación de esta tabla)	3	El impulso se da al CP PROFIBUS que envía estas peticiones cíclicas de Global Control a los esclavos DP seleccionados con Group Select. El parámetro Autoclear solo se evalúa en el caso de la petición de Global Control SYNC. Si al menos un esclavo DP del grupo seleccionado no está en la fase de transferencia de datos, en caso de Autoclear = 1 se conecta adicionalmente el modo CLEAR, lo que significa que los datos de salida de los esclavos DP se ponen a "0". En el parámetro Command Mode se pueden conectar las siguientes peticiones de Global Control: • SYNC • FREEZE • CLEAR (CLEAR-Bit = 1) no es compatible (observe también lo dicho en el manual de producto). o se puede desconectar: • UNSYNC • UNFREEZE • UNCLEAR (CLEAR-Bit = 0) Es posible indicar más de una petición en el parámetro Command Mode. Para terminar una petición cíclica en curso de Global Control, se tiene que ejecutar una nueva petición de Global Control (cíclica o acíclica). Para terminar la petición ajustada en el Command Mode, se tiene que desconectar la petición correspondiente. Por ejemplo, la petición SYNC se desconecta con una petición UNSYNC.
3	Borrar diagnóstico individual DP menos reciente	1. byte: Dirección escl. 1..126 127 = todos los esclavos	1	Se borran para uno o todos los esclavos los diagnósticos individuales DP más antiguos almacenados en el CP PROFIBUS.

CTYPE	Equivale a Petición	Parámetros en el bloque de petición		Descripción
4	Poner el estado operativo actual de DP	1. byte: RUN = 00_H CLEAR = 01_H OFFLINE = 03_H RUN con AUTOCLEAR = 04_H RUN sin AUTOCLEAR = 04_H	1	Con esta petición se puede poner el estado operativo de DP del siguiente modo: • RUN • CLEAR • OFFLINE El parámetro AUTOCLEAR significa que el maestro DP clase 1 pasa automáticamente al estado operativo CLEAR si se cumple la siguiente condición: al menos uno de los esclavos DP con los que intercambia datos el maestro DP clase 1 no está en el estado de transferencia de datos. Con el parámetro RUN sin AUTOCLEAR se suprime AUTOCLEAR. Notas: El estado operativo STOP = 02_H ya no es soportado en los módulos actuales (a partir del tipo de módulo DA02). Una indicación STOP = 02_H se refleja en el estado operativo OFFLINE.
5	Ajustar el estado operativo DP para STOP de la CPU	1. byte: RUN = 00_H CLEAR = 01_H OFFLINE = 03_H	1	Con esta petición se establece a qué estado operativo DP pasa el CP PROFIBUS en caso de CPU-STOP: • RUN • CLEAR • OFFLINE Como valor predeterminado (DEFAULT) con CPU-STOP, el CP PROFIBUS pasa al estado operativo de DP CLEAR. El estado operativo puesto se conserva en caso de un cambio de estado del CP de RUN --> STOP --> RUN. Notas: El estado operativo STOP = 02_H ya no es soportado en los módulos actuales (a partir del tipo de módulo DA02). Una indicación STOP = 02_H se refleja en el estado operativo OFFLINE.
6	Ajustar el estado operativo DP para STOP del CP	1. byte: OFFLINE=03_H	1	Con esta petición se establece a qué estado operativo DP pasa el CP PROFIBUS en caso de CP-STOP: • OFFLINE Como valor predeterminado (DEFAULT) con CP-STOP, el CP PROFIBUS pasa al estado operativo de DP OFFLINE. El estado operativo puesto se conserva en caso de un cambio de estado del CP de RUN --> STOP --> RUN. Notas: El estado operativo STOP = 02_H ya no es soportado en los módulos actuales (a partir del tipo de módulo DA02). Una indicación STOP = 02_H se refleja en el estado operativo OFFLINE.

CTYPE	Equivale a Petición	Parámetros en el bloque de petición		Descripción
7 *)	Lectura cíclica de los datos de entrada (maestro DP clase 2)	1. byte: dirección de esclavo 1..125	1	Esta petición no es compatible. Observe también las indicaciones del manual de producto.
8 *)	Lectura cíclica de los datos de salida (maestro DP clase 2)	1. byte: dirección de esclavo 1..125	1	Esta petición no es compatible. Observe también las indicaciones del manual de producto.
9	Terminar el procesamiento cíclico del esclavo DP por el maestro DP (clase 1 /clase 2)	1. byte: dirección de esclavo 1..125	1	Con esta petición se termina la lectura cíclica de los datos de entrada o los datos de salida del esclavo DP direccionado o la transferencia de datos (maestro DP clase 1). Después de esto, el esclavo DP ya no es procesado por el CP PROFIBUS como maestro DP (clase 2). **Esta operación desactiva el esclavo DP.**
10	Iniciar el procesamiento cíclico como maestro DP (clase 1)	1. byte: dirección de esclavo 1..125	1	El CP PROFIBUS que actúa como maestro DP (clase 1) parametriza el esclavo DP seleccionado e inicia la transferencia de datos cíclica (escribir salidas / leer entradas). **Esta operación activa el esclavo DP.**

*) Este CTYPE ya no es soportado en los módulos actuales (a partir del tipo de módulo DA02).

Consulte también

DP_CTRL (Página 8226)

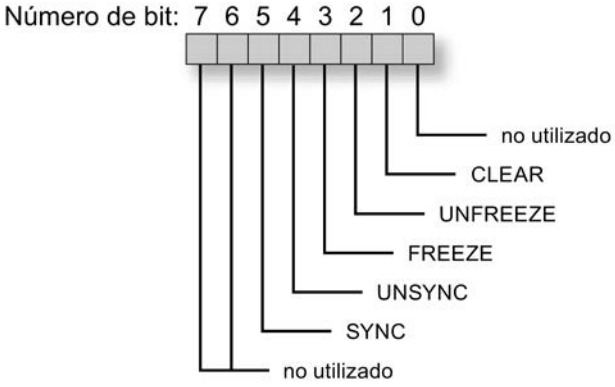
Command Mode y Group Select - DP_CTRL

Estructura de Command Mode

Para las peticiones de Global Control cíclicas y acíclicas, indique en el parámetro Command Mode los modos de operación para los datos de entrada y salida.

Al respecto significa:

1 = activado
0 = no activado



Estructura de Group Select

Con el parámetro Group Select se define en qué grupo se debe aplicar la petición de control indicada en Command Mode. Group Select ocupa el segundo byte de la petición de control. Cada bit define un grupo de esclavos DP posible.

Al respecto significa:

1 = asignado
0 = no asignado

Número de bit:	7	6	5	4	3	2	1	0
Grupo:	8	7	6	5	4	3	2	1

Consulte también

DP_CTRL (Página 8226)

Parámetros DONE, ERROR, STATUS

Códigos de condición

La tabla siguiente muestra el significado de la información suministrada por los parámetros DONE, ERROR y STATUS, la cual tiene que ser evaluada por el programa de usuario.

Nota

Para las entradas con la codificación $8Fxx_H$ en STATUS, tenga en cuenta también lo dicho sobre el parámetro de salida RET_VAL en las descripciones de los bloques de programa del sistema referenciados.

Puede consultar mediante STEP 7 los bloques de programa del sistema que se utilizan y los que son relevantes para la evaluación de errores.

Tabla 9-370 Indicadores DP_CTRL

DONE	ERROR	STATUS	Posible para CTYPE	Significado
0	0	8181_H	0..10	Petición en curso. El maestro DP no se ha iniciado por: • CP STOP o • "no hay parametrización" Observación: La indicación descrita aparece en lugar de una de las siguientes indicaciones descritas a continuación: 0,1,8183_H 0,1, 8333_H 0,1, 8334_H
1	0	0000_H	0..10	**Petición terminada sin errores.**
1	0	8214_H	0,1	**Petición terminada sin errores.** Advertencia: la petición cíclica de Global Control se envía como petición acíclica de Global Control
1	0	8215_H	0,1	**Petición terminada sin errores.** Todos los esclavos DP aludidos en el grupo seleccionado están desactivados.
1	0	8219_H	0,1	**Petición terminada sin errores.** Se ha intentado enviar de nuevo una petición cíclica de Global Control ya en curso. Global Control sigue en marcha sin cambios.
1	0	8228_H	0,1	**Petición terminada sin errores.** Advertencia: los esclavos DP aludidos en los grupos seleccionados no poseen módulos de entrada.
1	0	8229_H	0,1	**Petición terminada sin errores.** Advertencia: los esclavos DP aludidos en los grupos seleccionados no poseen módulos de salida.
1	0	8231_H	4,5,6	**Petición terminada sin errores.** Advertencia: el estado operativo de DP ya es "RUN"
1	0	8232_H	4,5,6	**Petición terminada sin errores.** Advertencia: el estado operativo de DP ya es "CLEAR"
1	0	8233_H	4,5,6	**Petición terminada sin errores.** Advertencia: el estado operativo de DP ya es "STOP"
1	0	8234_H	4,5,6	**Petición terminada sin errores.** Advertencia: el estado operativo de DP ya es "OFFLINE"
1	0	8235_H	4	**Petición terminada sin errores.** Advertencia: el estado operativo de DP ya es "RUN" con AUTOCLEAR conectado
1	0	8236_H	4	**Petición terminada sin errores.** Advertencia: el estado operativo de DP ya es "RUN" con AUTOCLEAR desconectado
1	0	8241_H	7-10	**Petición terminada sin errores.** Advertencia: el esclavo DP indicado no se ha configurado.
1	0	8243_H	7-10	**Petición terminada sin errores.** Advertencia: el esclavo DP ya está desactivado, porque la lista de módulos del esclavo DP en la base de datos del CP solo contiene módulos vacíos.

DONE	ERROR	STATUS	Posible para CTYPE	Significado
1	0	8245_H	7-10	**Petición terminada sin errores.** Advertencia: el esclavo DP se encuentra ya en el estado "Leer cíclicamente datos de entrada".
1	0	8246_H	7-10	**Petición terminada sin errores.** Advertencia: el esclavo DP se encuentra ya en el estado "Leer cíclicamente datos de salida".
1	0	8248_H	7-10	**Petición terminada sin errores.** Advertencia: la lista de módulos del esclavo CP en la base de datos del CP contiene módulos de entrada, de salida o de entrada/salida.
1	0	8249_H	7-10	**Petición terminada sin errores.** Advertencia: el esclavo está desactivado debido a un cambio del estado operativo de DP.
1	0	$824A_H$	7-10	**Petición terminada sin errores.** Advertencia: el esclavo DP está ya desactivado debido a una petición de DP_CTRL en el programa de la CPU.
0	1	8090_H	0..10	No existe módulo con esta dirección.
0	1	8091_H	0..10	La dirección lógica no está en formato de palabra doble.
0	1	$80B0_H$	0..10	El módulo no conoce el registro.
0	1	$80B1_H$	0..10	La longitud de registro indicada es incorrecta.
0	1	$80C0_H$	0..10	No se puede leer el registro.
0	1	$80C1_H$	0..10	El registro indicado está siendo procesado.
0	1	$80C2_H$	0..10	Hay demasiadas peticiones pendientes.
0	1	$80C3_H$		Medio de servicio (memoria) ocupado.
0	1	8183_H	0..10	Maestro DP no configurado... Observación: Si el maestro DP se encuentra en el estado operativo "STOP", se puede emitir también el estado 8181_H.
0	1	8184_H		Error del sistema o tipo de parámetro no permitido...
0	1	8311_H	0..10	Parámetro CTYPE fuera del rango de valores.
0	1	8312_H	0..10	La longitud del área en el parámetro CONTROL es demasiado pequeña.
0	1	8313_H	3,7,8,9, 10	Parámetro Dirección de esclavo fuera del rango de valores.
0	1	8315_H	0,1	Todos los esclavos DP del grupo indicado en Global Control están desactivados (se presenta siempre en caso de grupos vacíos).
0	1	8317_H	8	La longitud de los datos de salda configurados es mayor que el área de recepción configurada del esclavo DP. No es posible una activación del modo de esclavo "Leer datos de salida".
0	1	8318_H	0,1,4,5,6	El parámetro 1er. byte del bloque de datos de petición está fuera del rango de valores. Para GLOBAL CONTROL se ha utilizado CLEAR con SYNC o se ha enviado al grupo 0 un GLOBAL CONTROL con CLEAR puesto.
0	1	$831A_H$	0,1	Al menos un esclavo DP no domina FREEZE.
0	1	$831B_H$	0,1	Al menos un esclavo DP no domina SYNC.
0	1	8333_H	0,1	La petición no está permitida en el estado operativo de DP "STOP". Observación: Si no hay configurado ningún maestro DP, se puede emitir también el estado 8181_H.

DONE	ERROR	STATUS	Posible para CTYPE	Significado
0	1	8334_H	0, 1	La petición no está permitida en el estado operativo de DP "OFFLINE". Observación: Si no hay configurado ningún maestro DP, se puede emitir también el estado 8181_H.
0	1	8335_H	0, 1	El CP PROFIBUS se encuentra en el estado PROFIBUS: "Estación no en el anillo".
0	1	8339_H	0, 1	Al menos un esclavo DP del grupo seleccionado no se encuentra en la fase de transferencia de datos.
0	1	$833C_H$	1	El Global Control cíclico no se debe utilizar en el modo "Plc <-> CP libre". Este error no se presenta en el CP 3425, ya que dicho modo no es posible en el mismo (la transmisión de datos tiene lugar siempre a través de registros PBUS).
0	1	8341_H	7-10	El esclavo DP indicado no se ha configurado.
0	1	8183_H	0..10	Maestro DP no configurado.
0	1	8184_H	-	Error del sistema o tipo de parámetro no permitido.
0	1	$8F22_H$	0..10	Error de longitud de área al leer un parámetro (p. ej. DB demasiado corto).
0	1	$8F23_H$	0..10	Error de longitud de área al escribir un parámetro.
0	1	$8F24_H$	0..10	Error de área al leer un parámetro.
0	1	$8F25_H$	0..10	Error de área al escribir un parámetro.
0	1	$8F28_H$	0..10	Error de alineación al leer un parámetro.
0	1	$8F29_H$	0..10	Error de alineación al escribir un parámetro.
0	1	$8F30_H$	0..10	El parámetro está en el 1er. bloque de datos actual protegido de escritura.
0	1	$8F31_H$	0..10	El parámetro está en el segundo bloque de datos actual protegido de escritura.
0	1	$8F32_H$	0..10	El parámetro contiene número de DB demasiado alto.
0	1	$8F33_H$	0..10	Error de número de DB.
0	1	$8F3A_H$	0..10	Área de destino no cargada (DB).
0	1	$8F42_H$	0..10	Retardo en acuse al leer un parámetro del área de periferia.
0	1	$8F43_H$	0..10	Retardo en acuse al escribir el parámetro en el área de periferia.
0	1	$8F44_H$	0..10	Está bloqueado el acceso a un parámetro a leer para el procesamiento de la instrucción.
0	1	$8F45_H$	0..10	Está bloqueado el acceso a un parámetro a escribir para el procesamiento de la instrucción.
0	1	$8F7F_H$	0..10	Error interno, p. ej. referencia ANY no admisible.
0	1	$80C4_H$	0..10	Error de comunicación (se presenta temporalmente; por ello es conveniente una repetición en el programa de usuario).
0	1	$80D2_H$	0..10	Dirección base lógica incorrecta.

Consulte también

DP_CTRL (Página 8226)

Funciones 300C

ASCII, 3964(R)

SEND_PTP: Enviar datos (ASCII, 3964(R))

Descripción

La instrucción "SEND_PTP" permite enviar un conjunto de datos de un bloque de datos.

- El proceso de transmisión se activa después de la llamada del bloque y de un flanco ascendente en la entrada de control REQ.
- El área de los datos que deben enviarse se indica con SD_1 (número de DB y dirección inicial), la longitud del conjunto de datos con LEN.
- Para que la instrucción pueda procesar la petición, hay que llamarla con R(Reset)=FALSE. En caso de un flanco ascendente en la entrada de control R se cancela el proceso de transmisión en curso y se pone la instrucción al estado inicial. Una petición cancelada finaliza con un mensaje de error (salida STATUS).
- Con LADDR se indica la dirección E/S del submódulo que se ha definido en la configuración hardware.
- Si la petición ha finalizado sin errores, DONE se pone a TRUE, y si ha finalizado con errores, ERROR se pone a TRUE.

Si la petición se ha ejecutado con DONE=TRUE, significa lo siguiente:

- En caso de utilizar el driver ASCII: los datos se han enviado al interlocutor de la comunicación. Lo que no es seguro es si el interlocutor ha recibido los datos.
- En caso de utilizar el procedimiento 3964(R): los datos se han enviado al interlocutor de la comunicación y éste los ha acusado positivamente. Lo que no es seguro es si los datos también se han transmitido a la CPU interlocutora.

En STATUS, la CPU indica en caso de error o advertencia el número de evento correspondiente. DONE y ERROR/STATUS también se devuelven en caso de RESET de la instrucción (R=TRUE).

En caso de producirse un error, el resultado binario BR se borra. Si la instrucción finaliza sin errores, el resultado binario tiene el estado TRUE.

Nota

La instrucción no comprueba los parámetros; en caso de parametrización errónea la CPU puede derivar al estado STOP.

DB de instancia

La instrucción "SEND_PTP" trabaja con un DB de instancia. El número de DB se indica en la llamada. No está permitido acceder a los datos del DB de instancia.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SEND_PTP":

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
REQ	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Request: activa el intercambio de datos en caso de un flanco ascendente.
R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Reset. La petición se cancela. Transmisión bloqueada.
LADDR	Input	WORD	según CPU	W#16#03FF	Dirección E/S del submódulo que se ha definido en la configuración hardware.
DONE	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): • FALSE: la petición no se ha iniciado aún o se está ejecutando. • TRUE: la petición se ha ejecutado sin errores.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): petición terminada con error
STATUS	Output	WORD	W#16#0000 a W#16#FFFF	W#16#0000	Parámetro de estado (el parámetro sólo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre) STATUS tiene el significado siguiente en función del bit ERROR: • ERROR=FALSE: – STATUS tiene el valor W#16#0000: ni advertencias ni errores – STATUS tiene el valor<> W#16#0000: advertencia, STATUS da información detallada. • ERROR=TRUE: hay un error, STATUS da información detallada sobre el tipo de error.

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
SD_1	InOut	ANY	según CPU	0	Parámetro de transmisión: Aquí se indican los valores siguientes: • número del DB desde el que deben enviarse los datos. • número del byte de datos a partir del que deben enviarse los datos. Valores posibles: de 0 a 8190 P. ej.: DB 10 a partir del byte 2 -> DB10.DBB2 Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro SD_1 siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
LEN	InOut	INT	de 1 a 1024	1	Aquí se indica en bytes la longitud del conjunto de datos que debe enviarse (la longitud se indica indirectamente).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Coherencia de datos

La coherencia de datos está limitada a 206 bytes. Para la transferencia coherente de más de 206 bytes hay que tener en cuenta lo siguiente:

No vuelva a escribir en la parte utilizada actualmente del área de transmisión SD_1 hasta que no haya finalizado la transmisión. Esto sucede cuando el parámetro de estado DONE adopta el valor TRUE.

RCV_PTP: Recibir datos (ASCII, 3964(R))

Descripción

La instrucción "RCV_PTP" permite recibir datos y guardarlos en un bloque de datos.

- Después de la llamada, la instrucción está lista para recibir con el valor TRUE en la entrada de control EN_R. Es posible cancelar una transmisión en curso con el estado lógico FALSE del parámetro EN_R. Una petición cancelada finaliza con un mensaje de error (salida STATUS). La recepción estará desactivada mientras haya un estado lógico de FALSE en el parámetro EN_R.
- El área de recepción se indica con RD_1 (número de DB y dirección inicial), y la longitud del bloque de datos con LEN.

- Para que la instrucción pueda procesar la petición, hay que llamarla con R(Reset)=FALSE. En caso de un flanco ascendente en la entrada de control R se cancela la transmisión en curso y se pone la instrucción al estado inicial. Una petición de recepción cancelada finaliza con un mensaje de error (salida STATUS).
- Con LADDR se indica la dirección E/S del submódulo que se ha definido en la configuración hardware.
- Si la petición ha finalizado sin errores, NDR se pone a TRUE, y si ha finalizado con errores, ERROR se pone a TRUE.
- En STATUS , la CPU indica el número de evento correspondiente en caso de error o advertencia.
- NDR y ERROR/STATUS también se devuelven en caso de RESET de la instrucción (R=TRUE) (parámetro LEN == 16#00).

En caso de producirse un error, el resultado binario BR se borra. Si la instrucción finaliza sin errores, el resultado binario tiene el estado TRUE.

Nota

La instrucción no comprueba los parámetros; en caso de parametrización errónea la CPU puede derivar al estado STOP.

DB de instancia

La instrucción "RCV_PTP" trabaja con un DB de instancia. El número de DB se indica en la llamada. No está permitido acceder a los datos del DB de instancia.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RCV_PTP":

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
EN_R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Enable to receive: Habilitación de recepción
R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Reset: La petición se cancela.
LADDR	Input	WORD	según CPU	W#16#03FF	Dirección E/S del submódulo que se ha definido en la configuración hardware.
NDR	Output	BOOL	TRUE/FALSE	FALSE	Petición finalizada sin errores, datos aplicados • FALSE: la petición no se ha iniciado aún o se está ejecutando. • TRUE: la petición se ha concluido correctamente.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): petición terminada con error

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
STATUS	Output	WORD	W#16#0000 a W#16#FFFF	W#16#0000	Parámetro de estado (el parámetro sólo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre) STATUS tiene el significado siguiente en función del bit ERROR: • ERROR=FALSE: – STATUS tiene el valor W#16#0000: ni advertencias ni errores – STATUS tiene el valor<> W#16#0000: advertencia, STATUS da información detallada. • ERROR=TRUE: hay un error, STATUS da información detallada sobre el tipo de error.
RD_1	InOut	ANY	según CPU	0	Parámetro de recepción: Aquí se indica lo siguiente: • número del DB en el que se guardan los datos recibidos. • número del byte de datos a partir del que deben guardarse los datos recibidos. Valores posibles: de 0 a 8190 P. ej.: DB 20 a partir del byte 5 -> DB20.DBB5 Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RD_1 siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
LEN	InOut	INT	de 0 a 1024	0	Salida de la longitud de datos (número de bytes)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Coherencia de datos

La coherencia de datos está limitada a 206 bytes. Para la transferencia coherente de más de 206 bytes hay que tener en cuenta lo siguiente:

No vuelva a acceder al DB de recepción hasta que no se hayan recibido los datos por completo (NDR = TRUE). Posteriormente, bloquee el DB de recepción (EN_R = FALSE) hasta que se hayan procesado los datos.

RES_RCVB: Borrar búfer de recepción (ASCII, 3964(R))

Descripción

Con la instrucción "RES_RECV" se borra todo el búfer de recepción del módulo. Todos los telegramas guardados se rechazan. Si en el momento de llamar la instrucción "RES_RCVB" está entrando un telegrama, se guardará.

- La petición se activa después de la llamada de la instrucción y de un flanco ascendente en la entrada de control REQ. La petición puede durar varias llamadas (ciclos de programa).
- Para que la instrucción pueda procesar la petición, hay que llamarla con R(Reset)=FALSE. En caso de un flanco ascendente en la entrada de control R se cancela el proceso de borrado y se pone la instrucción al estado inicial. Una petición cancelada finaliza con un mensaje de error (salida STATUS).
- Con LADDR se indica la dirección E/S del submódulo que se ha definido en la configuración hardware.
- Si la petición ha finalizado sin errores, DONE se pone a TRUE, y si ha finalizado con errores, ERROR se pone a TRUE.
- En STATUS, la CPU indica el número de evento correspondiente en caso de error o advertencia.
- DONE y ERROR/STATUS también se devuelven en caso de RESET de la instrucción (R=TRUE).

En caso de producirse un error, el resultado binario BR se borra. Si el bloque finaliza sin errores, el resultado binario tiene el estado TRUE.

Nota

La instrucción no comprueba los parámetros; en caso de parametrización errónea la CPU puede derivar al estado STOP.

DB de instancia

La instrucción "RES_RCVB" trabaja con un DB de instancia. El número de DB se indica en la llamada. No está permitido acceder a los datos del DB de instancia.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "RES_RCVB":

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
REQ	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Request: activa la petición en caso de un flanco ascendente
R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Reset: la petición se cancela.
LADDR	Input	WORD	según CPU	W#16#03FF	Dirección E/S del submódulo que se ha definido en la configuración hardware.
DONE	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): • FALSE: la petición no se ha iniciado aún o se está ejecutando. • TRUE: la petición se ha ejecutado sin errores.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): petición terminada con error
STATUS	Output	WORD	W#16#0000 a W#16#FFFF	W#16#0000	Parámetro de estado (el parámetro sólo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre) STATUS tiene el significado siguiente en función del bit ERROR: • ERROR=FALSE: – STATUS tiene el valor W#16#0000: ni advertencias ni errores – STATUS tiene el valor <> W#16#0000: advertencia, STATUS da información detallada. • ERROR=TRUE: hay un error, STATUS da información detallada sobre el tipo de error.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

RK 512

SEND_RK: Enviar datos (RK 512)

Descripción

La instrucción SEND_RK permite enviar un conjunto de datos de un bloque de datos.

- El proceso de transmisión se activa después de la llamada de la instrucción y del flanco ascendente en la entrada de control REQ.
- El área de los datos que deben enviarse se indica con SD_1 (número de DB y dirección inicial), la longitud del conjunto de datos con LEN.
- En la instrucción se indica también el área de recepción en el interlocutor. La CPU registra esta información en el encabezado del telegrama y la transmite al interlocutor.
- El destino se indica con el número de CPU R_CPU (sólo relevante en caso de comunicación multiprocesador), el tipo de datos R_TYPE (bloques de datos (DB) y bloques de datos ampliados (DX)), el número de bloque de datos R_DBNO y el offset R_OFFSET en el que debe escribirse el primer byte.
- Con R_CF_BYT y R_CF_BIT se definen el byte y bit de marcas de acoplamiento en la CPU interlocutora.
- Con el parámetro SYNC_DB se establece el DB en el que se guardarán los datos comunes a todas las instrucciones utilizadas para la inicialización en el arranque y la sincronización. El número de DB debe ser idéntico para todas las instrucciones utilizadas en el programa de usuario.
- Para que la instrucción pueda procesar la petición, hay que llamarla con R(Reset)=FALSE. En caso de un flanco ascendente en la entrada de control R se cancela el proceso de transmisión en curso y se pone la instrucción al estado inicial. Una petición cancelada finaliza con un mensaje de error (salida STATUS).
- Con LADDR se indica la dirección E/S del submódulo que se ha definido en la configuración hardware.
- Si la petición ha finalizado sin errores, DONE se pone a TRUE, y si ha finalizado con errores, ERROR se pone a TRUE.
- Si la petición se ha ejecutado con DONE = TRUE, los datos se han enviado al interlocutor de la comunicación, que los ha acusado positivamente, y los datos se han transferido a la CPU interlocutora.
- En STATUS, la CPU indica el número de evento correspondiente en caso de error o advertencia.
- DONE y ERROR/STATUS también se devuelven en caso de RESET de la instrucción (R=TRUE).

En caso de producirse un error, el resultado binario BR se borra. Si el bloque finaliza sin errores, el resultado binario tiene el estado TRUE.

Nota

La instrucción no comprueba los parámetros; en caso de parametrización errónea la CPU puede derivar al estado STOP.

DB de instancia

La instrucción "SEND_RK" trabaja con un DB de instancia. El número de DB se indica en la llamada. No está permitido acceder a los datos del DB de instancia.

Particularidades de la transmisión de datos

Tenga en cuenta las particularidades siguientes al "enviar datos":

- Con RK 512 sólo es posible enviar un número par de datos. Si se indica un número impar de datos como longitud (LEN), se transferirá un byte de relleno adicional con el valor "0" al final de los datos.
- Con RK 512 sólo es posible indicar un offset par. Si se indica un offset impar, los datos se guardan en el interlocutor a partir del siguiente offset par más pequeño.
  Ejemplo: el offset es "7", los datos se guardan a partir del byte "6".

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SEND_RK":

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
SYNC_DB	Input	INT	según CPU	0	Número del DB en el que se guardan los datos comunes para la sincronización de las instrucciones RK (longitud mínima 240 bytes).
REQ	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Request: activa la petición en caso de un flanco ascendente
R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Reset: la petición se cancela.
LADDR	Input	WORD	según CPU	W#16#03FF	Dirección E/S del submódulo que se ha definido en la configuración hardware.
R_CPU	Input	INT	de 0 a 4	1	Número de la CPU interlocutora (sólo en modo multiprocesador)
R_TYPE	Input	CHAR	'D', 'X'	'D'	Tipo de dirección en la CPU interlocutora (sólo se admiten mayúsculas) 'D': bloque de datos 'X': bloque de datos avanzado

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
R_DBNO	Input	INT	de 0 a 255	0	Número del bloque de datos en la CPU interlocutora
R_OFFSET	Input	INT	de 0 a 510 (sólo valores pares)	0	Número del byte de datos en la CPU interlocutora
R_CF_BYT	Input	INT	de 0 a 255	255	Byte de marcas de acoplamiento en la CPU interlocutora (255: significa: sin marcas de acoplamiento)
R_CF_BIT	Input	INT	de 0 a 7	0	Bit de marcas de acoplamiento en la CPU interlocutora
DONE	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): • FALSE: la petición no se ha iniciado aún o se está ejecutando. • TRUE: la petición se ha ejecutado sin errores.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): petición terminada con error
STATUS	Output	WORD	W#16#0000 a W#16#FFFF	W#16#0000	Parámetro de estado (el parámetro sólo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre) STATUS tiene el significado siguiente en función del bit ERROR: • ERROR=FALSE: – STATUS tiene el valor W#16#0000: ni advertencias ni errores – STATUS tiene el valor<> W#16#0000: advertencia, STATUS da información detallada. • ERROR=TRUE: hay un error, STATUS da información detallada sobre el tipo de error.

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
SD_1	Input	ANY	según CPU	0	Parámetro de transmisión: Aquí se indica lo siguiente: • número del DB desde el que deben enviarse los datos. • número del byte de datos a partir del que deben enviarse los datos. Valores posibles: de 0 a 8190 P. ej.: DB 10 a partir del byte 2 -> DB10.DBB2 Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro SD_1 siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
LEN	Input	INT	de 1 a 1024	1	Aquí se indica en bytes la longitud del conjunto de datos que debe enviarse (la longitud se indica indirectamente).

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Datos del encabezado del telegrama

La tabla siguiente muestra los datos del encabezado del telegrama RK 512.

Origen en el propio sistema de automatización S7 (CPU local)	Al destino, CPU interlocutora	Encabezado del telegrama, bytes		
		3/4 tipo de comando	5/6 Z-DBNR[1] / Z-Offset[2]	7/8 número en
Bloque de datos	Bloque de datos	AD	DB/DW[3]	palabras
Bloque de datos	Bloque de datos avanzado	AD	DB/DW[3]	palabras

[1] Z-DBNR: número del bloque de datos de destino
[2] Z-Offset: dirección inicial de destino
[3] DW: offset en palabras

Coherencia de datos

La coherencia de datos está limitada a 128 bytes. Para la transferencia de datos coherente de más de 128 bytes hay que tener en cuenta lo siguiente:

No vuelva a escribir en la parte utilizada actualmente del área de transmisión SD_1 hasta que no haya finalizado la transmisión. Esto sucede cuando el parámetro de estado DONE adopta el valor TRUE.

FETCH_RK: Recoger datos (RK 512)

Descripción

La instrucción "FETCH_RK" permite recoger un conjunto de datos del interlocutor y guardar los datos en un bloque de datos.

- El proceso de transmisión se activa después de la llamada del bloque y de un flanco ascendente en la entrada de control REQ.
- El área en el que se guardan los datos recogidos se indica con RD_1 (número de DB y dirección inicial), la longitud del conjunto de datos con LEN.
- En la instrucción también se indica el área a partir de la cual deben recogerse los datos del interlocutor. La CPU registra esta información en el encabezado del telegrama RK512 y la transmite al interlocutor.
- El área del interlocutor se indica con el número de CPU R_CPU (sólo relevante en caso de comunicación multiprocesador), el tipo de datos R_TYPE (bloques de datos y bloques de datos ampliados, marcas, entradas, salidas, contadores y temporizadores), el número del bloque de datos R_DBNO (sólo relevante en bloques de datos y bloques de datos ampliados) y el offset R_OFFSET del que debe recogerse el primer byte.
- Con R_CF_BYT y R_CF_BIT se definen el byte y bit de marcas de acoplamiento en la CPU interlocutora.
- Con el parámetro SYNC_DB se establece el DB en el que se guardarán los datos comunes a todas las instrucciones utilizadas para la inicialización en el arranque y la sincronización. El número de DB debe ser idéntico para todas las instrucciones utilizadas en el programa de usuario.
- Para que la instrucción pueda procesar la petición, hay que llamarla con R(Reset)=FALSE. En caso de un flanco ascendente en la entrada de control R se cancela la transmisión en curso y se pone la instrucción al estado inicial. Una petición cancelada finaliza con un mensaje de error (salida STATUS).
- Con LADDR se indica la dirección E/S del submódulo que se ha definido en la configuración hardware.
- Si la petición ha finalizado sin errores, DONE se pone a TRUE, y si ha finalizado con errores, ERROR se pone a TRUE.
- En STATUS, la CPU indica el número de evento correspondiente en caso de error o advertencia.
- DONE y ERROR/STATUS también se devuelven en caso de RESET de la instrucción (R=TRUE).

En caso de producirse un error, el resultado binario BR se borra. Si la instrucción finaliza sin errores, el resultado binario tiene el estado TRUE.

Nota

La instrucción no comprueba los parámetros; en caso de parametrización errónea la CPU puede derivar al estado STOP. Si se recogen datos de la propia CPU, debe programarse una instrucción "SERVE_RK (Página 8254)" en la CPU.

DB de instancia

La instrucción "FETCH_RK" trabaja con un DB de instancia. El número de DB se indica en la llamada. No está permitido acceder a los datos del DB de instancia.

Particularidades de los bloques de datos (avanzados)

Tenga en cuenta las particularidades siguientes al "recoger datos" de bloques de datos y bloques de datos ampliados:

- Con RK 512 sólo es posible recoger un número par de datos. Si se indica un número impar como longitud (LEN), siempre se transmite un byte adicional. Sin embargo, en el DB de destino se registra la cantidad correcta de datos.
- Con RK 512 sólo es posible indicar un offset par. Si se indica un offset impar, los datos se recogen del interlocutor desde el siguiente offset par más pequeño.
  Ejemplo: el offset es 7, los datos se recogen a partir del byte 6.

Particularidades en temporizadores y contadores

Si se recogen temporizadores o contadores del interlocutor, hay que tener en cuenta que para cada temporizador y cada contador se recogen 2 bytes. Así, p. ej., si desea recoger 10 contadores, deberá indicar una longitud de 20.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "FETCH_RK".

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
SYNC_DB	Input	INT	según CPU	0	Número del DB en el que se guardan los datos comunes para la sincronización de las instrucciones RK (longitud mínima 240 bytes).
REQ	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Request: activa la petición en caso de un flanco ascendente
R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Reset: la petición se cancela.

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
LADDR	Input	WORD	según CPU	W#16#03FF	Dirección E/S del submódulo que se ha definido en la configuración hardware.
R_CPU	Input	INT	de 0 a 4	1	N.º de la CPU interlocutora (sólo en modo multiprocesador)
R_TYPE	Input	CHAR	'D', 'X', 'M', 'E', 'A', 'Z', 'T'	'D'	Tipo de dirección en la CPU interlocutora • 'D': bloque de datos • 'X': bloque de datos avanzado • 'M': marcas • 'E': entradas • 'A': salidas • 'Z': contadores • 'T': temporizadores
R_DBNO	Input	INT	de 0 a 255	0	Número del bloque de datos en la CPU interlocutora
R_OFFSET	Input	INT	Véase la tabla: "Parámetros del origen de los datos (CPU interlocutora)"	0	Número del byte de datos en la CPU interlocutora
R_CF_BYT	Input	INT	de 0 a 255	255	Byte de marcas de acoplamiento en la CPU interlocutora (255: significa: sin marcas de acoplamiento)
R_CF_BIT	Input	INT	de 0 a 7	0	Bit de marcas de acoplamiento en la CPU interlocutora
DONE	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): • FALSE: la petición no se ha iniciado aún o se está ejecutando. • TRUE: la petición se ha ejecutado sin errores.
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): petición finalizada con errores

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
STATUS	Output	WORD	W#16#0000 a W#16#FFFF	W#16#0000	Parámetro de estado (el parámetro sólo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre) STATUS tiene el significado siguiente en función del bit ERROR: • ERROR=FALSE: – STATUS tiene el valor W#16#0000: ni advertencias ni errores – STATUS tiene el valor<> W#16#0000: advertencia, STATUS da información detallada. • ERROR=TRUE: hay un error, STATUS da información detallada sobre el tipo de error.
RD_1	Input	ANY	según CPU	0	Parámetro de recepción: Aquí se indica lo siguiente: • número del DB en el que se guardan los datos recogidos. • número del byte de datos a partir del que se guardan los datos recogidos. Valores posibles: de 0 a 8190 P. ej.: DB 10 a partir del byte 2 -> DB10.DBB2 Nota: Tenga en cuenta que en las CPUs S7-300, el parámetro RD_1 siempre requiere que se especifiquen por completo los parámetros de DB (ej.: P#DB13.DBX0.0 byte 100). En las CPUs S7-300 no está permitido omitir un número de DB explícito; de lo contrario se genera un aviso de error en el programa de usuario.
LEN	Input	INT	de 1 a 1024	1	Aquí se indica la longitud del conjunto de datos que debe recogerse en bytes (la longitud se indica indirectamente). Para cada temporizador y contador debe indicarse una longitud de dos bytes.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros del origen de los datos (CPU interlocutora)

La tabla siguiente muestra los tipos de datos que pueden transferirse.
El valor "R_OFFSET" está determinado por la CPU interlocutora.

Origen en la CPU interlocutora	R_TYPE	R_NO	R_OFFSET (en bytes)
Bloque de datos	'D'	0 - 255	0 - 510, sólo tienen sentido valores pares
Bloque de datos avanzado	'X'	0 - 255	0 - 510, sólo tienen sentido valores pares
Marcas	'M'	irrelevante	0 - 255
Entradas	'E'	irrelevante	0 - 255
Salidas	'A'	irrelevante	0 - 255
Contadores	'Z'	irrelevante	0 - 255
Temporizadores	'T'	irrelevante	0 - 255

Datos del encabezado del telegrama

La tabla siguiente muestra los datos del encabezado del telegrama RK 512.

Origen en la CPU interlocutora	Al destino, su sistema de automatización S7 (CPU local)	Encabezado del telegrama, bytes		
		3/4 tipo de comando	5/6 Q-DBNR[1] / Q-Offset[2]	7/8 número en
Bloque de datos	Bloque de datos	ED	DB/DW	Palabras
Bloque de datos avanzado	Bloque de datos	EX	DB/DW	Palabras
Marcas	Bloque de datos	EM	Dirección de byte	Bytes
Entradas	Bloque de datos	EE	Dirección de byte	Bytes
Salidas	Bloque de datos	EA	Dirección de byte	Bytes
Contadores	Bloque de datos	EZ	Número de contador	Palabras
Temporizadores	Bloque de datos	ET	Número de temporizador	Palabras

[1] Q-DBNR: número del bloque de datos de origen
[2] Q-Offset: dirección inicial de origen

Coherencia de datos

La coherencia de datos está limitada a 128 bytes. Para la transferencia de datos coherente de más de 128 bytes hay que tener en cuenta lo siguiente:

No vuelva a escribir en la parte utilizada actualmente del área de recepción RD_1 hasta que no haya finalizado la transmisión. Esto sucede cuando el parámetro de estado DONE adopta el valor TRUE.

SERVE_RK: Recibir y preparar datos (RK 512)

Descripción

La instrucción "SERVE_RK" se utiliza para recibir datos y ponerlos a disposición.

- Recibir datos: los datos se guardan en el área de datos especificada por el interlocutor en el encabezado del telegrama RK 512. La llamada de la instrucción es necesaria cuando el interlocutor de la comunicación ejecuta una petición "Enviar datos" (petición SEND).
- Poner datos a disposición: los datos se recogen del área de datos especificada por el interlocutor en el encabezado del telegrama RK 512. La llamada de la instrucción es necesaria cuando el interlocutor de la comunicación ejecuta una petición "Recoger datos" (petición FETCH).

Después de la llamada, la instrucción está disponible con el valor TRUE en la entrada de control EN_R. Es posible cancelar una transmisión en curso con el estado lógico FALSE del parámetro EN_R. Una petición cancelada finaliza con un mensaje de error (salida STATUS). La recepción estará desactivada mientras haya un estado lógico de FALSE en el parámetro EN_R.

Con el parámetro "SYNC_DB" se establece el DB en el que se guardarán los datos comunes a todas las instrucciones utilizadas para la inicialización en el arranque y la sincronización. El número de DB debe ser idéntico para todas las instrucciones utilizadas en el programa de usuario.

Para que la instrucción pueda procesar la petición, hay que llamarla con R(Reset)=FALSE. En caso de un flanco ascendente en la entrada de control R se cancela la transmisión en curso y se pone la instrucción al estado inicial. Una petición cancelada finaliza con un mensaje de error (salida STATUS).

Con LADDR se indica la dirección E/S del submódulo que se ha definido en HW Config.

Si la petición ha finalizado sin errores, NDR se pone a TRUE, y si ha finalizado con errores, ERROR se pone a TRUE.

Con NDR=TRUE para una llamada de la instrucción en los parámetros L_TYPE, L_DBNO y L_OFFSET, la CPU indica dónde se guardan los datos y de dónde se recogen. Además, para una llamada se muestran los parámetros L_CF_BYT y L_CF_BIT y la longitud LEN de la petición en cuestión.

En STATUS, la CPU indica el número de evento correspondiente en caso de error o advertencia.

NDR y ERROR/STATUS también se devuelven en caso de RESET de la instrucción (R=TRUE) (parámetro LEN == 16#00).

En caso de producirse un error, el resultado binario BR se borra. Si el bloque finaliza sin errores, el resultado binario tiene el estado TRUE.

Nota

La instrucción no comprueba los parámetros; en caso de parametrización errónea la CPU puede derivar al estado STOP.

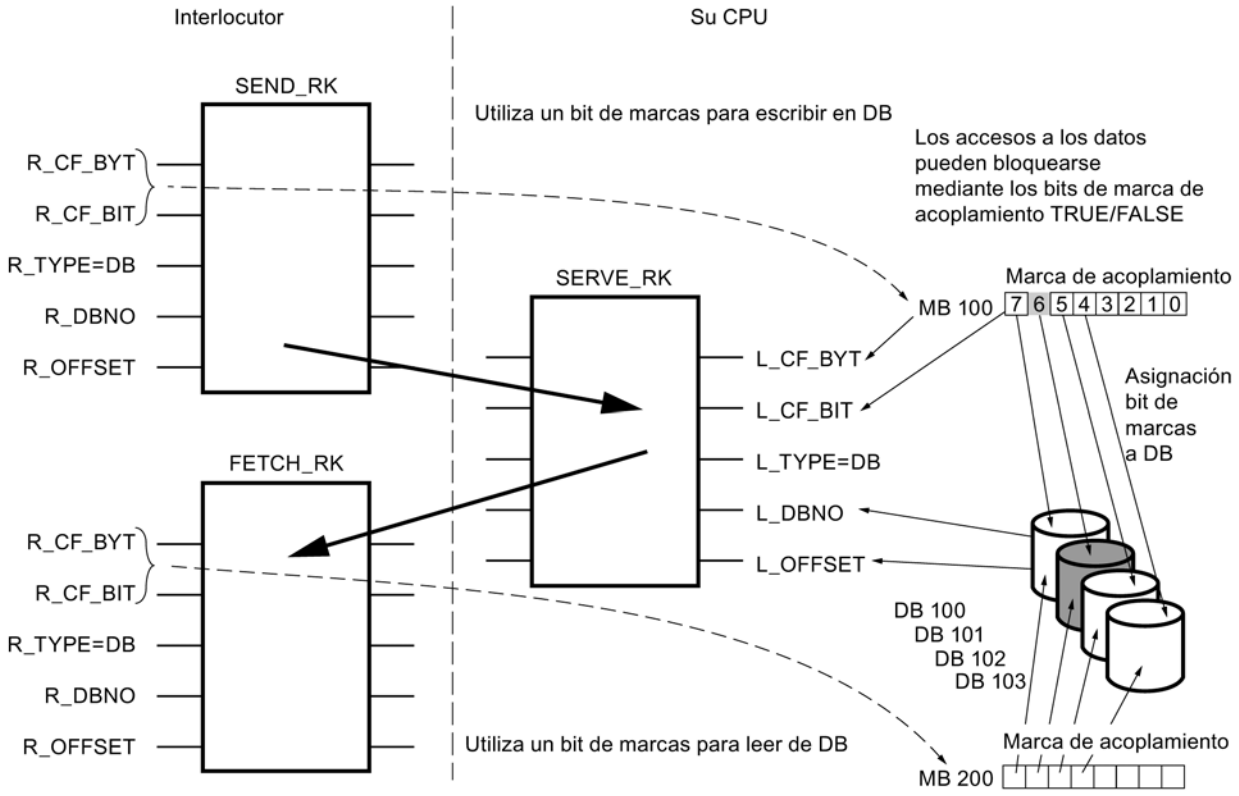
DB de instancia

La instrucción "SERVE_RK" trabaja con un DB de instancia. El número de DB se indica en la llamada. No está permitido acceder a los datos del DB de instancia.

Aplicación de las marcas de acoplamiento

Mediante una marca de acoplamiento es posible bloquear y habilitar las peticiones SEND y FETCH del interlocutor de la comunicación. Existe la posibilidad de impedir que los datos que todavía no se han procesado, se sobrescriban o lean.

Puede definirse una marca de acoplamiento para cada petición.



Ejemplo: SEND_RK con marca de acoplamiento:

En este ejemplo, el interlocutor de la comunicación envía datos al DB 101 en su CPU

1. Ponga la marca de acoplamiento 100.6 de la propia CPU a FALSE.
2. En la petición SEND del interlocutor, indique la marca de acoplamiento 100.6 (parámetros R_CF_BYT, R_CF_BIT).
   La marca de acoplamiento se transfiere a la propia CPU en el encabezado del telegrama RK 512.
   Antes de procesar la petición, la CPU comprueba la marca de acoplamiento indicada en el encabezado del telegrama RK 512. La petición sólo se procesa si la marca de acoplamiento de la propia CPU tiene el valor FALSE. Si la marca de acoplamiento tiene el valor TRUE, en el telegrama de reacción se enviará el mensaje de error "32 hex" al interlocutor de la comunicación.
   Una vez transferidos los datos al DB101, la marca de acoplamiento 100.6 de la propia CPU pasará de SERVE al valor TRUE y el byte y bit de marcas de acoplamiento se devolverán en SERVE durante una llamada (si NDR =TRUE).
3. Evaluando la marca de acoplamiento en el programa de usuario (marca de acoplamiento 100.6 =TRUE) es posible saber si la petición ha finalizado y si los datos transferidos pueden procesarse.
4. Una vez procesados los datos en el programa de usuario, se debe volver a poner la marca de acoplamiento 100.6 a FALSE. Sólo entonces, el interlocutor acoplado podrá volver a ejecutar la petición sin errores.

Parámetros

La tabla siguiente muestra los parámetros de la instrucción "SERVE_RK".

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
SYNC_DB	Input	INT	según CPU	0	Número del DB en el que se guardan los datos comunes para la sincronización de las instrucciones RK (longitud mínima 240 bytes).
EN_R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Enable to receive: petición habilitada
R	Input	BOOL	TRUE/FALSE	FALSE	Parámetro de control Reset: la petición se cancela.
LADDR	Input	WORD	según CPU	W#16#03FF	Dirección E/S del submódulo que se ha definido en la configuración hardware.
NDR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado New Data Ready (el parámetro sólo está activado durante una llamada): • FALSE: la petición no se ha iniciado aún o se está ejecutando. • TRUE: la petición se ha ejecutado correctamente.

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
ERROR	Output	BOOL	TRUE/FALSE	FALSE	Parámetro de estado (el parámetro sólo está activado durante una llamada): petición finalizada con errores
STATUS	Output	WORD	W#16#0000 a W#16#FFFF	W#16#0000	Parámetro de estado (el parámetro sólo está activado durante una llamada. Por ello, para visualizar el estado debe copiarse el STATUS en un área de datos libre) STATUS tiene el significado siguiente en función del bit ERROR: • ERROR=FALSE: – STATUS tiene el valor W#16#0000: ni advertencias ni errores – STATUS tiene el valor <> W#16#0000: advertencia, STATUS da información detallada. • ERROR=TRUE: hay un error, STATUS da información detallada sobre el tipo de error.
L_TYPE	Output	CHAR	'D' 'D', 'M', 'I', 'O', 'C', 'T',	' '	**Recibir datos:** Tipo de zona de destino en la CPU local (sólo se admiten mayúsculas): • 'D': bloque de datos **Poner datos a disposición:** Tipo de zona de origen en la CPU local (sólo se admiten mayúsculas): • 'D': bloque de datos • 'M': marcas • 'I': entradas • 'O': salidas • 'C': contadores • 'T': temporizadores El parámetro sólo está activado durante una llamada.
L_DBNO	Output	INT	según CPU	0	Número del bloque de datos en la CPU local. El parámetro sólo está activado durante una llamada.
L_OFFSET	Output	INT	0 - 510	0	Número del byte de datos en la CPU local. El parámetro sólo está activado durante una llamada.
L_CF_BYT	Output	INT	de 0 a 255	0	Byte de marcas de acoplamiento en la CPU local. El parámetro sólo está activado durante una llamada. (255: significa: sin marcas de acoplamiento)

Parámetro	Declaración	Tipo de datos	Rango de valores	Ajuste predeterminado	Descripción
L_CF_BIT	Output	INT	de 0 a 7	0	Bit de marcas de acoplamiento en la CPU local. El parámetro sólo está activado durante una llamada.
LEN	InOut	INT	de 0 a 1024	0	Longitud del telegrama, número en bytes (el parámetro sólo está activado durante una llamada)

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Coherencia

La coherencia está limitada a 128 bytes. Para la transferencia de datos coherente de más de 128 bytes hay que tener en cuenta lo siguiente:

Utilice la función de marca de acoplamiento. No vuelva a acceder a los datos hasta que éstos no se hayan transferido por completo (evaluación de la marca de acoplamiento definida para esta petición; la marca de acoplamiento está pendiente para una llamada en la instrucción si NDR = TRUE). No vuelva a poner la marca de acoplamiento a FALSE hasta que no se hayan procesado los datos.

Comunicación con iSlave/iDevice

I_GET: Leer datos de un interlocutor en la propia estación S7

Descripción

La instrucción "I_GET" permite leer datos de un interlocutor que se encuentra dentro de la estación S7 propia. El interlocutor puede estar en el aparato central, en un aparato de ampliación o bien descentralizado. Recuerde que tiene que haber asignado los interlocutores descentralizados a la CPU propia. En el interlocutor no existe una instrucción correspondiente.

La activación del proceso de recepción se produce tras la llamada de la instrucción con REQ=1. A continuación se llama la instrucción hasta que la recepción de datos se indique con BUSY=0. RET_VAL contiene entonces la longitud del bloque de datos recibido en bytes.

Asegúrese de que el área de recepción definida por el parámetro RD (en la CPU recibida) tenga como mínimo la misma longitud que el área de lectura definida por el parámetro VAR_ADDR (en el interlocutor). Además deben coincidir los tipos de datos en RD y VAR_ADDR.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "I_GET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
CONT	Input	BOOL	I, Q, M, D, L	Parámetro de control "continue" Consulte también: Auto-Hotspot
IOID	Input	BYTE	I, Q, M, D, L, P o constante	Identificación del área de direccionamiento del módulo interlocutor: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si las direcciones son idénticas, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L, P o constante	Dirección lógica del módulo interlocutor Con un módulo mixto debe indicarse la dirección más baja de las dos.
VAR_ADDR	Input	ANY	I, Q, M, D	Referencia al área de la CPU interlocutora que debe leerse. Seleccione un tipo de datos soportado por el interlocutor.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente. Si no se produce ningún error, RET_VAL contiene la longitud en bytes del bloque de datos copiado en el área de recepción RD en forma de número positivo.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: recepción todavía no finalizada. • BUSY=0: la recepción ha finalizado, o bien no hay ningún proceso de recepción activo.
RD	Output	ANY	I, Q, M, D	Referencia al área de recepción (receive data area). Se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME así como matrices de los tipos de datos indicados, excepto BOOL. El área de recepción RD debe ser como mínimo tan larga como el área de lectura VAR_ADDR en el interlocutor. Además es necesario que los tipos de datos coincidan en RD y VAR_ADDR. El tamaño máximo del área de recepción es de 76 bytes.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Transición de estado operativo tras STOP

Si la CPU cambia al estado operativo STOP, la conexión establecida por "I_GET" se deshace. El tipo de la llamada realizada a continuación determina si los datos ya recibidos, que se encuentran en una memoria intermedia del sistema operativo, se pierden o no:

- Con un rearranque (no con S7-300 ni S7-400H), estos datos se copian en el área definida por RD.
- En un rearranque completo (en caliente) o arranque en frío, los datos se rechazan.

Transición de estado operativo del interlocutor a STOP

Un cambio de la CPU del interlocutor al estado operativo STOP no tiene efecto alguno sobre la transmisión con "I_GET": Los datos también se leen en el estado operativo STOP.

Coherencia de datos

Los datos se reciben con coherencia.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

I_PUT: Escribir datos de un interlocutor en la propia estación S7

Descripción

La instrucción "I_PUT" permite escribir datos en un interlocutor que se encuentra dentro de la estación S7 propia. El interlocutor puede estar en el aparato central, en un aparato de ampliación o bien descentralizado. Recuerde que tiene que haber asignado los interlocutores descentralizados a la CPU propia. En el interlocutor no existe una instrucción correspondiente.

El proceso de transmisión se produce tras la llamada de la instrucción con nivel 1 en la entrada de control REQ.

Asegúrese de que el área de transmisión definida por el parámetro SD (en la CPU emisora) tenga la misma longitud que el área de recepción definida por el parámetro VAR_ADDR (en el interlocutor). Además deben coincidir los tipos de datos en SD y VAR_ADDR.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "I_PUT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
CONT	Input	BOOL	I, Q, M, D, L	Parámetro de control "continue" Consulte también: Auto-Hotspot
IOID	Input	BYTE	I, Q, M, D, L, P o constante	Identificación del área de direccionamiento del módulo interlocutor: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si las direcciones son idénticas, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L, P o constante	Dirección lógica del módulo interlocutor. Con un módulo mixto debe indicarse la dirección más baja de las dos.
VAR_ADDR	Input	ANY	I, Q, M, D, L	Referencia al área de la CPU interlocutora en la que debe escribirse. Seleccione un tipo de datos soportado por el interlocutor.
SD	Input	ANY	I, Q, M, D	Referencia al área de la CPU propia que contiene los datos que deben enviarse. Se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME así como matrices de los tipos de datos indicados, excepto BOOL. SD debe tener la misma longitud que el parámetro VAR_ADDR del interlocutor. Además es necesario que los tipos de datos coincidan en SD y VAR_ADDR. El tamaño máximo del área de transmisión es de 76 bytes.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: la transmisión todavía no ha finalizado. • BUSY=0: la transmisión ha finalizado, o bien no hay ningún proceso de transmisión activo.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Transición de estado operativo tras STOP

Si la CPU cambia al estado operativo STOP, la conexión establecida por "I_PUT" se deshace. Ya no es posible enviar los datos. Si los datos enviados ya estaban copiados en el búfer interno en el momento de la transición de estado operativo, el contenido del búfer se rechaza.

Transición de estado operativo del interlocutor a STOP

Un cambio de la CPU del interlocutor al estado operativo STOP no tiene efecto alguno sobre la transmisión de datos con "I_PUT". Los datos enviados se escriben de todos modos.

Coherencia de datos

Los datos se envían con coherencia.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

I_ABORT: Deshacer conexión con interlocutor en la propia estación S7

Descripción

La instrucción "I_ABORT" deshace una conexión con un interlocutor dentro de la estación S7 propia que se había establecido con "I_GET (Página 8257)" o "I_PUT (Página 8259)". Si la petición correspondiente a "I_GET (Página 8257)" o "I_PUT (Página 8259)" ha finalizado (BUSY = 0), tras la llamada de "I_ABORT" los recursos de conexión ocupados para dicha petición se liberan de nuevo en ambos lados.

Si la petición correspondiente a "I_GET (Página 8257)" o "I_PUT (Página 8259)" no ha finalizado (BUSY = 1), después de interrumpirse la conexión el propio usuario debe volver a llamar la instrucción correspondiente con REQ = 0 y CONT = 0 y esperar a BUSY = 0. Sólo así quedarán liberados todos los recursos de conexión ocupados.

"I_ABORT" sólo puede llamarse en el lado en el que se ejecuta "I_PUT (Página 8259)" o "I_GET (Página 8257)" (es decir, en el lado del cliente).

La interrupción de la conexión se activa tras la llamada de la instrucción con REQ=1.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "I_ABORT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
IOID	Input	BYTE	I, Q, M, D, L, P o constante	Identificación del área de direccionamiento del módulo interlocutor: • B#16#54 = entrada de periferia (PI) • B#16#55 = salida de periferia (PQ) Si se trata de un módulo mixto, debe indicarse la identificación de área de la dirección más baja. Si las direcciones son idénticas, debe indicarse B#16#54.
LADDR	Input	WORD	I, Q, M, D, L, P o constante	Dirección lógica del módulo interlocutor. Con un módulo mixto debe indicarse la dirección más baja de las dos.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: la interrupción de la conexión todavía no se ha finalizado. • BUSY=0: la interrupción de la conexión ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Transición de estado operativo tras STOP

Si la CPU cambia al estado operativo STOP, cualquier interrupción iniciada con "I_ABORT" finaliza.

Transición de estado operativo del interlocutor a STOP

Un cambio de la CPU del interlocutor al estado operativo STOP no tiene efecto alguno sobre la interrupción de la conexión con "I_ABORT": La conexión se deshace.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

PROFINET CBA

Información sobre PN_IN, PN_OUT y PN_DP

Nota

El sistema operativo actualiza de forma estándar tanto el componente PROFINET CBA como las interconexiones DP en el punto de control del ciclo. No obstante, en caso de haber desconectado estas actualizaciones automáticas en la configuración (p. ej. para poder influir mejor en el comportamiento en el tiempo de la CPU), tendrá que realizar las actualizaciones personalmente. Para ello, llame las instrucciones "PN_IN (Página 8266)", "PN_OUT (Página 8267)" y "PN_DP (Página 8268)" en momentos adecuados. En la configuración, las actualizaciones indicadas sólo pueden desactivarse conjuntamente.

Memoria sombra

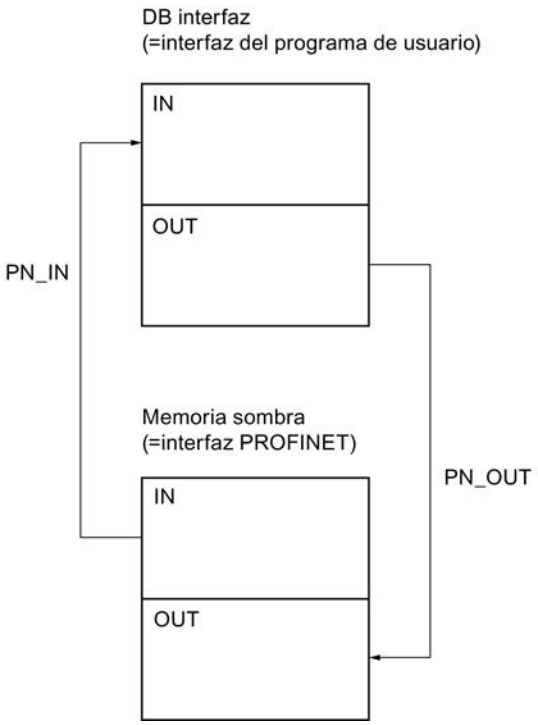
El DB interfaz es la interfaz con el programa de usuario del componente PROFINET CBA. Para garantizar que sus entradas y salidas sean coherentes durante la ejecución de las partes correspondientes del programa de usuario, para cada DB interfaz existe un área de memoria de idéntica estructura gestionada por el sistema operativo, que se conoce como "memoria sombra". En el programa de usuario sólo es posible acceder al DB interfaz, mientras que otros componentes PROFINET CBA (externos) sólo pueden acceder a la memoria sombra. De este modo se excluye la posibilidad de un conflicto de acceso a las entradas y salidas del DB interfaz.

Actualización del componente PROFINET CBA

De la realización arriba descrita de la coherencia de datos mediante una memoria sombra se deriva que la actualización de los componentes PROFINET CBA se compone de las siguientes acciones:

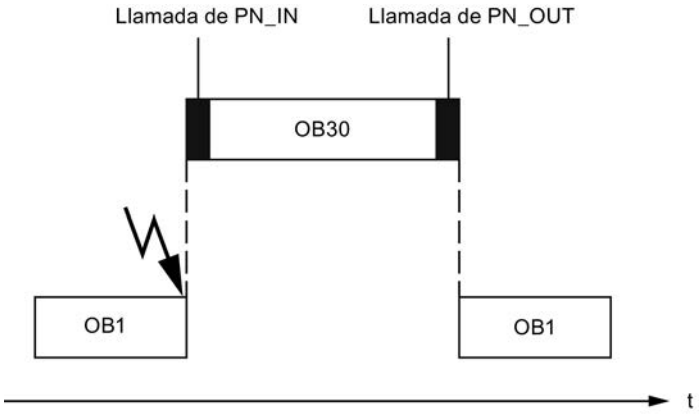
- Copia de las entradas de la memoria sombra en las entradas del DB interfaz antes de la ejecución del programa correspondiente al componente PROFINET CBA
- Copia de las salidas del DB interfaz en las salidas de la memoria sombra tras la ejecución del programa correspondiente al componente PROFINET CBA

La actualización del componente PROFINET CBA se ejecuta en CPUs con interfaz PROFINET integrada, ya sea por orden del sistema operativo o de las instrucciones "PN_IN (Página 8266)" y "PN_OUT (Página 8267)" (según se haya configurado al crear el componente PROFINET CBA). En la siguiente figura se representa esquemáticamente la actualización con las instrucciones "PN_IN (Página 8266)" y "PN_OUT (Página 8267)".



La actualización por parte del sistema siempre tiene lugar en el punto de control del ciclo.

Si por el contrario, las actualizaciones se llevan a cabo por orden de las instrucciones "PN_IN (Página 8266)" y "PN_OUT (Página 8267)", llame "PN_IN (Página 8266)" al principio del OB en el que se encuentra el programa correspondiente al componente PROFINET CBA, y "PN_OUT (Página 8267)" al final de este OB. Este comportamiento se representa en la siguiente imagen a partir del ejemplo del OB 30.



Nota

(varios DB interfaz PROFINET en un dispositivo):

Con componentes multifunción existe la posibilidad de actualizar todos los DBs interfaz con una petición (DBNO=0) (un proceso similar a la actualización en el punto de control del ciclo, disparado por instrucción). Si se produce un error durante la actualización de un DB interfaz, la actualización de los DBs restantes continúa. Un RET_VAL negativo se refiere a uno de los DBs interfaz actualizados.

Actualización de las interconexiones DP

Las interconexiones DP son actualizadas por el sistema operativo o por la instrucción "PN_DP (Página 8268)", según se haya configurado al crear el componente PROFINET CBA.

PN_IN: Actualizar entradas de la interfaz del programa de usuario

Descripción

La instrucción "PN_IN" copia los datos de entrada recibidos en PROFINET CBA desde la memoria sombra del componente PROFINET CBA al DB interfaz correspondiente. Una vez finalizada la instrucción, los datos de entrada actuales quedan a disposición de la aplicación.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PN_IN":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DBNO	Input	WORD	I, Q, M, D, L o constante	N.º del DB interfaz (DBNO=0: Actualización de todos los DBs interfaz PROFINET CBA)
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8001	La configuración de PROFINET CBA no existe o no es válida.
8002	El n.º de DB no concuerda con la configuración del componente.
8004	El n.º de DB concuerda con la configuración del componente, pero el DB no está cargado.

Código de error (W#16#...)	Explicación
8005	El DB interfaz se ha compilado con la palabra clave UNLINKED, de modo que sólo se encuentra en la memoria de carga, pero no en la de trabajo.
8006	El DB interfaz está protegido contra escritura en la CPU.
80B1	Error de longitud al leer o escribir. La configuración del componente no concuerda con el DB cargado.
8xyy	Información de error general Véase también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

PN_OUT: Actualizar salidas de la interfaz del programa de usuario

Descripción

La instrucción "PN_OUT" copia los datos de salida generados en la aplicación desde el DB interfaz del componente PROFINET CBA a la memoria sombra correspondiente. Una vez finalizada la instrucción, los datos de salida actuales quedan disponibles para otros componentes PROFINET CBA.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PN_OUT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
DBNO	Input	WORD	I, Q, M, D, L o constante	N.º del DB interfaz (DBNO=0: Actualización de todos los DBs interfaz PROFINET)
RET_VAL	Return	INT	I, Q, M, D, L	Información de error

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	No ha ocurrido ningún error.
8001	La configuración de PROFINET CBA no existe o no es válida.
8002	El n.º de DB no concuerda con la configuración del componente.
8004	El n.º de DB concuerda con la configuración del componente, pero el DB no está cargado.
8005	El DB interfaz se ha compilado con la palabra clave UNLINKED, de modo que sólo se encuentra en la memoria de carga, pero no en la de trabajo.
8006	El DB interfaz está protegido contra escritura en la CPU.

Código de error (W#16#...)	Explicación
80B1	Error de longitud al leer o escribir. La configuración del componente no concuerda con el DB cargado.
8xyy	Información de error general Véase también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

PN_DP: Actualizar interconexiones DP

Descripción

Con la instrucción "PN_DP" se actualizan:

- las interconexiones entre componentes PROFINET CBA en el PROFIBUS local y
- las interconexiones con transmisión cíclica entre componentes PROFINET CBA en el PROFIBUS local y componentes PROFINET CBA externos. Estas interconexiones abarcan varias redes (Industrial Ethernet y PROFIBUS DP).

Funcionamiento

"PN_DP" es una instrucción asíncrona, de modo que la ejecución puede prolongarse a lo largo de varias llamadas. Para iniciar la actualización de las interconexiones DP, llame la instrucción "PN_DP" con REQ= 1.

El estado de la petición se indica a través de los parámetros de salida RET_VAL y BUSY.

Consulte también: Significado de REQ, RET_VAL y BUSY en las instrucciones que funcionan asíncronamente (Página 5595).

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PN_DP":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	REQ=1: actualización de las interconexiones DP lanzada
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá un código de error.
BUSY	Output	BOOL	I, Q, M, D, L	BUSY=1: La actualización de las interconexiones DP aún no ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetro RET_VAL

Código de error (W#16#...)	Explicación
0000	La petición se ha ejecutado sin errores.
7000	Primera llamada con REQ=0. No se lanza la actualización de las interconexiones DP. BUSY tiene el valor "0".
7001	(sólo relevante para S7-400:) Primera llamada con REQ=1. BUSY tiene el valor "1".
7002	(sólo relevante para S7-400:) Llamada intermedia (REQ irrelevante). La actualización de las interconexiones DP aún no ha finalizado. BUSY tiene el valor "1".
8001	La configuración de PROFINET CBA no existe o no es válida.
8095	Ha lanzado otra actualización de las interconexiones DP en una clase de prioridad superior. Sin embargo, todavía se está procesando la actualización en la clase de prioridad menor (por el sistema operativo o un procesamiento "PN_DP").
8xyy	Información de error general Consulte también: Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

Comunicación MPI

X_SEND: Enviar datos a un interlocutor fuera de la propia estación S7

Descripción

La instrucción "X_SEND" permite enviar datos a un interlocutor fuera de la estación S7 propia. La recepción de datos en el interlocutor se efectúa a través de la instrucción "X_RCV (Página 8270)". La transmisión inicia tras la llamada con REQ=1.

Asegúrese de que el área de transmisión definida por el parámetro SD (en la CPU emisora) sea menor o igual que el área de recepción definida por el parámetro RD (en el interlocutor). Si SD es del tipo de datos BOOL, RD también tiene que ser del tipo de datos BOOL.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "X_SEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
CONT	Input	BOOL	I, Q, M, D, L	Parámetro de control "continue" Consulte también: Auto-Hotspot
DEST_ID	Input	WORD	I, Q, M, D, L o constante	Parámetro de direccionamiento "destination ID". Contiene la dirección MPI configurada del interlocutor.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ_ID	Input	DWORD	I, Q, M, D, L o constante	Identificación de la petición para identificar los datos en el interlocutor.
SD	Input	ANY	I, Q, M, D	Referencia al área de transmisión. Se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5TIME, DATE_AND_TIME, así como matrices de los tipos de datos indicados, excepto BOOL. El tamaño máximo del área de transmisión es de 76 bytes.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: la transmisión todavía no ha finalizado. • BUSY=0: la transmisión ha finalizado, o bien no hay ningún proceso de transmisión activo.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Coherencia de datos

Los datos se envían con coherencia.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

X_RCV: Recibir datos de un interlocutor fuera de la propia estación S7

Descripción

Con la instrucción "X_RCV" se reciben datos enviados por uno o más interlocutores fuera de la estación S7 propia con la instrucción "X_SEND (Página 8268)".

Con la instrucción "X_RCV":

- se puede determinar si existen datos enviados en ese momento. Si es así, el sistema operativo los habrá dispuesto en una cola de espera interna.
- se puede copiar el bloque de datos más antiguo de la cola de espera en un área de recepción definida por el usuario.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "X_RCV":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
EN_DT	Input	BOOL	I, Q, M, D, L	Parámetro de control "enable data transfer". Con el valor 0 se comprueba si existe como mínimo un bloque de datos. Con el valor "1" el bloque de datos más antiguo de la cola de espera se copia en el área de la memoria de trabajo especificada por el usuario en RD.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente. Si no se produce ningún error, RET_VAL contiene • con EN_DT=0/1 y NDA=0:W#16#7000. En este caso no hay ningún bloque de datos en la cola de espera. • con EN_DT=0 y NDA=1 la longitud del bloque de datos más antiguo registrado en la cola de espera en bytes en forma de número positivo. • con EN_DT=1 y NDA=1 la longitud del bloque de datos copiado en el área de recepción RD en bytes en forma de número positivo.
REQ_ID	Output	DWORD	I, Q, M, D, L	Identificación de petición de la instrucción "X_SEND (Página 8268)" cuyos datos enviados se encuentran en primer lugar en la cola de espera, es decir, los datos más antiguos de la cola. Si no hay ningún bloque de datos en la cola de espera, REQ_ID tiene el valor "0".

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
NDA	Output	BOOL	I, Q, M, D, L	Parámetro de estado "new data arrived". NDA=0: • En la cola de espera no hay ningún bloque de datos. NDA=1: • En la cola de espera hay como mínimo un bloque de datos. (Llamada de "X_RCV" con EN_DT=0). • El bloque de datos más antiguo de la cola de espera se ha copiado en el programa de usuario (llamada de "X_RCV" con EN_DT=1).
RD	Output	ANY	I, Q, M, D	Referencia al área de recepción (receive data area). Se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME así como matrices de los tipos de datos indicados, excepto BOOL. Para rechazar el bloque de datos más antiguo de la cola de espera, asigne a RD el valor NIL. El tamaño máximo del área de recepción es de 76 bytes.

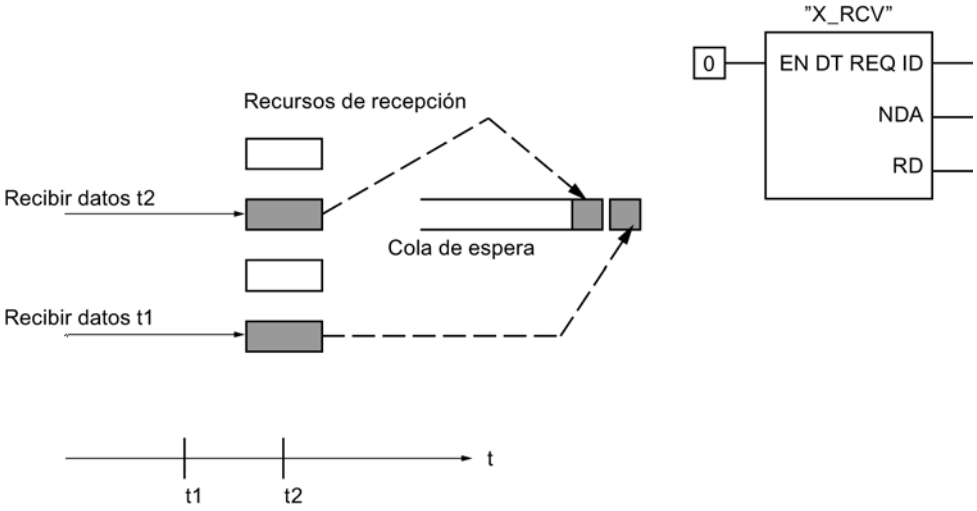
Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Indicar la recepción de datos con EN_DT=0

En cuanto se han recibido los datos de un interlocutor, el sistema operativo los dispone en la lista de espera por orden de recepción.

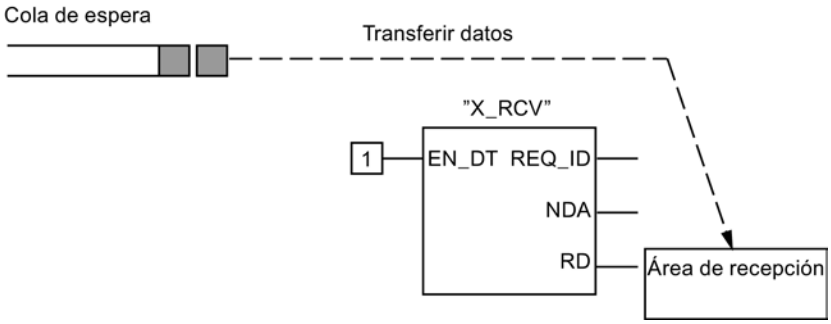
Para comprobar si existe como mínimo un bloque de datos, llame "X_RCV" con EN_DT=0 y evalúe el parámetro de salida NDA:

- NDA=0 significa que no hay ningún bloque de datos registrado en la cola de espera. REQ_ID es irrelevante, RET_VAL contiene W#16#7000.
- NDA=1 significa que en la cola de espera hay como mínimo un bloque de datos listo para ser recogido.
  En este caso, evalúe también los parámetros de salida RET_VAL y, si procede, REQ_ID. RET_VAL contiene la longitud del bloque de datos en bytes, REQ_ID la identificación de petición del bloque de transmisión. Si en la cola de espera hay varios bloques de datos, REQ_ID y RET_VAL corresponden al último bloque de la cola.



Aplicar datos en el área de recepción con EN_DT=1

Si se llama "X_RCV" con EN_DT=1, el bloque de datos más antiguo de la cola de espera se copia en el área de la memoria de trabajo especificada por RD. RD debe ser mayor o igual que el área de transmisión definida por el parámetro SD de la instrucción correspondiente "X_SEND (Página 8268)". Si el parámetro de entrada correspondiente SD es del tipo de datos BOOL, RD también tiene que ser del tipo de datos BOOL. Para depositar los datos recibidos en diferentes áreas, determine REQ_ID (llamada de instrucción con EN_DT=0) y en la llamada siguiente (con EN_DT=1) seleccione RD convenientemente. Si se produjo un error durante el proceso de copia, en RET_VAL se indica la longitud del bloque de datos copiado en bytes, y se devuelve un acuse positivo al emisor.



Rechazar datos

Si no desea aplicar los datos, asigne a RD el valor NIL. En este caso, el emisor recibe un acuse negativo (RET_VAL de la instrucción correspondiente "X_SEND (Página 8268)" tiene el valor W#16#80B8), y en RET_VAL de la instrucción "X_RCV" se registra "0".

Coherencia de datos

Tras la llamada con EN_DT=1 y RETVAL=W#16#00xy hay nuevos datos en el área de recepción RD. Otra llamada del bloque puede sobrescribir estos datos. Para impedirlo, no vuelva a llamar "X_RCV" con la misma área de recepción RD antes de haber evaluado los datos recibidos.

Transición de estado operativo tras STOP

En caso de transición al estado operativo STOP

- todas las nuevas peticiones entrantes se acusan negativamente.
- rige lo siguiente para las nuevas peticiones: todas las peticiones registradas en la cola de espera de recepción se acusan negativamente.
  - Si a continuación se produce un rearranque completo (en caliente) o arranque en frío se rechazan todos los bloques de datos.
  - Si a continuación se produce un rearranque (no con S7-300 ni S7-400H), en el programa de usuario se aplica el bloque de datos correspondiente a la petición más antigua, siempre que se haya consultado antes de la transición al estado operativo STOP (llamada de "X_RCV" con EN_DT=0). De lo contrario se rechaza.
    Todos los bloques de datos restantes se rechazan.

Interrupción de la conexión

Si se cancela una conexión y en la cola de espera de recepción ya hay registrada una petición que pertenece a dicha conexión, la petición se rechaza.

Excepción: Si dicha petición es la más antigua de la cola de espera y ya la ha reconocido con la llamada de "X_RCV" con EN_DT=0, se puede aplicar en el área de recepción con EN_DT=1.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

X_GET: Leer datos de un interlocutor fuera de la propia estación S7

Descripción

La instrucción "X_GET" permite leer datos de un interlocutor que se encuentra fuera de la estación S7 propia. En el interlocutor no existe una instrucción correspondiente.

La activación del proceso de lectura se produce tras la llamada de la instrucción con REQ=1. A continuación se llama "X_GET" hasta que la recepción de datos se indique con BUSY=0. RET_VAL contiene entonces la longitud del bloque de datos recibido en bytes.

Asegúrese de que el área de recepción definida por el parámetro RD (en la CPU recibida) tenga como mínimo la misma longitud que el área de lectura definida por el parámetro VAR_ADDR (en el interlocutor). Además deben coincidir los tipos de datos en RD y VAR_ADDR.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "X_GET":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
CONT	Input	BOOL	I, Q, M, D, L	Parámetro de control "continue" Consulte también: Auto-Hotspot
DEST_ID	Input	WORD	I, Q, M, D, L o constante	Parámetro de direccionamiento "destination ID". Contiene la dirección MPI del interlocutor. Dicha dirección ha sido configurada previamente por el usuario.
VAR_ADDR	Input	ANY	I, Q, M, D	Referencia al área de la CPU interlocutora que debe leerse. Seleccione un tipo de datos soportado por el interlocutor.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente. Si no se produce ningún error, RET_VAL contiene la longitud en bytes del bloque de datos copiado en el área de recepción RD en forma de número positivo.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: recepción todavía no finalizada. • BUSY=0: la recepción ha finalizado, o bien no hay ningún proceso de recepción activo.
RD	Output	ANY	I, Q, M, D	Referencia al área de recepción (receive data area). Se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME, S5_TIME, DATE_AND_TIME así como matrices de los tipos de datos indicados, excepto BOOL. El área de recepción RD debe ser como mínimo tan larga como el área de lectura VAR_ADDR en el interlocutor. Además es necesario que los tipos de datos coincidan en RD y VAR_ADDR. El tamaño máximo del área de recepción es de 76 bytes.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Transición de estado operativo tras STOP

Si la CPU cambia al estado operativo STOP, la conexión establecida por "X_GET" se deshace. El tipo de la llamada realizada a continuación determina si los datos ya recibidos, que se encuentran en una memoria intermedia del sistema operativo, se pierden o no:

- Con un rearranque (no con S7-300 ni S7-400H), estos datos se copian en el área definida por RD.
- En un rearranque completo (en caliente) o arranque en frío, los datos se rechazan.

Transición de estado operativo del interlocutor a STOP

Un cambio de la CPU del interlocutor al estado operativo STOP no tiene efecto alguno sobre la transmisión con "X_GET": Los datos también se leen en el estado operativo STOP.

Coherencia de datos

Los datos se reciben con coherencia.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

X_PUT: Escribir datos en un interlocutor fuera de la propia estación S7

Descripción

La instrucción "X_PUT" permite escribir datos en un interlocutor que se encuentra fuera de la estación S7 propia. En el interlocutor no existe una instrucción correspondiente.

La activación del proceso de escritura se produce tras la llamada de la instrucción con REQ=1. A continuación se llama "X_PUT" hasta que la recepción de datos se indique con BUSY=0.

Asegúrese de que el área de transmisión definida por el parámetro SD (en la CPU emisora) tenga la misma longitud que el área de recepción definida por el parámetro VAR_ADDR (en el interlocutor). Además deben coincidir los tipos de datos en SD y VAR_ADDR.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "X_PUT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
CONT	Input	BOOL	I, Q, M, D, L	Parámetro de control "continue" Consulte también: Auto-Hotspot.
DEST_ID	Input	WORD	I, Q, M, D, L o constante	Parámetro de direccionamiento "destination ID". Contiene la dirección MPI del interlocutor. Dicha dirección ha sido configurada previamente por el usuario.
VAR_ADDR	Input	ANY	I, Q, M, D	Referencia al área de la CPU interlocutora en la que debe escribirse. Seleccione un tipo de datos soportado por el interlocutor.
SD	Input	ANY	I, Q, M, D	Referencia al área de la CPU propia que contiene los datos que deben enviarse. Se permiten los siguientes tipos de datos: BOOL, BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, DATE, TOD, TIME,S5_TIME, DATE_AND_TIME así como matrices de los tipos de datos indicados, excepto BOOL. SD debe tener la misma longitud que el parámetro VAR_ADDR del interlocutor. Además es necesario que los tipos de datos coincidan en SD y VAR_ADDR. El tamaño máximo del área de transmisión es de 76 bytes.
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: la transmisión todavía no ha finalizado. • BUSY=0: la transmisión ha finalizado, o bien no hay ningún proceso de transmisión activo.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Transición de estado operativo tras STOP

Si la CPU cambia al estado operativo STOP, la conexión establecida por "X_PUT" se deshace. Ya no es posible enviar los datos. Si los datos enviados ya estaban copiados en el búfer interno en el momento de la transición de estado operativo, el contenido del búfer se rechaza.

Transición de estado operativo del interlocutor a STOP

Un cambio de la CPU del interlocutor al estado operativo STOP no tiene efecto alguno sobre la transmisión con "X_PUT": Los datos enviados se escriben de todos modos.

Coherencia de datos

Los datos se envían con coherencia.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

X_ABORT: Deshacer conexión con un interlocutor fuera de la propia estación S7

Descripción

La instrucción "X_ABORT" deshace la conexión con una estación S7 fuera de la estación S7 propia, que se había establecido con las instrucciones "X_SEND (Página 8268)", "X_GET (Página 8273)" o "X_PUT (Página 8275)".

- Si la petición correspondiente a "X_SEND (Página 8268)", "X_GET (Página 8273)" o "X_PUT (Página 8275)" ha finalizado (BUSY=0), tras la llamada de la instrucción "X_ABORT" los recursos ocupados para ella se liberan de nuevo en ambos lados.
- Si la petición correspondiente a "X_SEND (Página 8268)", "X_GET (Página 8273)" o "X_PUT (Página 8275)" no ha finalizado (BUSY=1), una vez interrumpida la conexión el propio usuario debe volver a llamar la instrucción correspondiente con REQ=0 y CONT=0 y esperar a BUSY=0. Sólo así quedarán liberados todos los recursos de conexión ocupados.

"X_ABORT" sólo puede llamarse en el lado en el que se está ejecutando "X_SEND (Página 8268)", "X_PUT (Página 8275)" o "X_GET (Página 8273)". La interrupción de la conexión se activa tras la llamada de la instrucción con REQ=1.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "X_ABORT":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control "request to activate" Consulte también: Auto-Hotspot
DEST_ID	Input	WORD	I, Q, M, D, L o constante	Parámetro de direccionamiento "destination ID". Contiene la dirección MPI del interlocutor. Dicha dirección ha sido configurada previamente por el usuario.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
RET_VAL	Return	INT	I, Q, M, D, L	Si se produce un error durante la ejecución de la instrucción, el valor de respuesta contendrá el código de error correspondiente.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY=1: la interrupción de la conexión todavía no se ha finalizado. • BUSY=0: la interrupción de la conexión ha finalizado.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Transición de estado operativo tras STOP

Si la CPU cambia al estado operativo STOP, cualquier interrupción iniciada con "X_ABORT" finaliza.

Transición de estado operativo del interlocutor a STOP

Un cambio de la CPU del interlocutor al estado operativo STOP no tiene efecto alguno sobre la interrupción de la conexión con "X_ABORT": La conexión se deshace.

Parámetro RET_VAL

Consulte también:

- Auto-Hotspot
- Evaluación de errores con el parámetro de salida RET_VAL (Página 5598)

TeleService

PG_DIAL: Establecer una conexión remota con una PG/un PC

Descripción de PG_DIAL

Descripción

La instrucción "PG_DIAL" transmite un número de teléfono y una ID de evento a un TS Adapter. El TS Adapter establece una conexión remota con una programadora o un PC con el número de teléfono indicado. La ID de evento se transmite a la programadora o el PC y se transfiere a una aplicación en espera.

Si la ID de evento (parámetro EVENT_ID (Página 8282)) de la aplicación se transfiere correctamente, el TS Adapter recibe un acuse, que se transmite a la instrucción "PG_DIAL". A continuación finaliza la ejecución de "PG_DIAL" y el estado se transmite a la parte que llama

"PG_DIAL". La aplicación de la programadora o el PC es responsable de deshacer la conexión remota.

Si se produce un error durante la ejecución, que provoca su interrupción, el código de error se transmite a la parte que llama la instrucción "PG_DIAL". El TS Adapter deshace cualquier conexión remota que pueda haber establecida.

Llamada de "PG_DIAL"

La instrucción "PG_DIAL" puede llamarse cíclicamente o en un programa controlado por tiempo. En la llamada de "PG_DIAL" es imprescindible indicar un DB de instancia.

Para procesar una función de bloque son necesarias varias llamadas de "PG_DIAL". Por ello, la llamada de "PG_DIAL" en un "bucle de espera" no es conveniente. El final de la función de bloque se indica con BUSY = 0.

Interrupción de la conexión

Si la CPU pasa al estado operativo STOP mientras "PG_DIAL" está activa, la conexión con el TS Adapter se deshace.

La conexión con el TS Adapter también se pierde cuando en el bus MPI se producen problemas de comunicación graves o, por ejemplo, si la CPU se desconecta. En estos casos el TS Adapter no rechaza el número de teléfono y la ID de evento que ya le habían sido transmitidos. El TS Adapter establece la conexión remota con la programadora o el PC y transmite la ID de evento. Sin embargo, el TS Adapter rechaza el acuse transmitido por la programadora o el PC.

Si un programa de usuario intenta establecer una conexión remota mientras el TS Adapter se encuentra en el estado arriba descrito, "PG_DIAL" finaliza con el valor de respuesta W#16#B10A. El programa de usuario puede repetir la llamada de una conexión remota más adelante. El valor de retorno W#16#B10A se obtiene también cuando varias CPUs intentan establecer una conexión remota simultáneamente a través del mismo TS Adapter.

ATENCIÓN
Cambiar programas de usuario Las partes del programa de usuario con efecto directo sobre llamadas de "PG_DIAL" sólo pueden modificarse con la CPU en el estado operativo STOP. Esto se refiere en especial al borrado e intercambio de bloques de programa que contienen llamadas de "PG_DIAL". En caso de incumplimiento, es posible que los recursos de conexión permanezcan ocupados. El sistema de automatización puede pasar a un estado no definido con las instrucciones de comunicación para conexiones S7 no configuradas. Una vez transmitidos los cambios es necesario ejecutar un rearranque completo (en caliente) o un arranque en frío de la CPU.

Coherencia de datos

Los parámetros de entrada del bloque de función se copian en un búfer interno en la primera llamada de "PG_DIAL". No modifique estos datos antes de que la primera llamada haya finalizado (valor de respuesta W#16#7001) ya que de lo contrario pueden transmitirse datos incoherentes.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "PG_DIAL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MPI_TS_ADAPTER	Input	INT	I, Q, M, D, L o constante	Dirección MPI del TS Adapter
PHONE_NO (Página 8281)	Input	ANY	D	Referencia a un string de datos con una longitud máxima de 31 caracteres
EVENT_ID (Página 8282)	Input	ANY	I, Q, M, D	Referencia a una matriz de bytes con una longitud máxima de 16 caracteres
BUSY (Página 8282)	Output	BOOL	I, Q, M, D, L	• BUSY=1: el establecimiento de la conexión remota todavía no ha finalizado. • BUSY=0: la ejecución de "PG_DIAL" ha finalizado.
STATUS (Página 8282)	Output	INT	I, Q, M, D, L	Valor de retorno de "PG_DIAL"

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros de bloque virtuales EN y ENO y el bit BR

Los parámetros de bloque virtuales EN y ENO sólo aparecen en caso de integración de "PG_DIAL" en la representación KOP o FUP del editor STEP 7. Están estrechamente vinculados con el bit BR (resultado binario) de la palabra de estado.

- Parámetros de entrada EN
  El estado del resultado lógico (RLO) se almacena durante la llamada del bloque en el bit BR.
- Parámetros de salida ENO
  Si "PG_DIAL" se ejecuta sin errores, el estado del bit BR disponible en el momento de la llamada del bloque se restablece al final de la misma.
  Cuando se indica un mensaje de error a través del parámetro de salida STATUS, el bit BR se pone a "0" inmediatamente antes de abandonar "PG_DIAL".

Consulte también

Establecer conexiones desde y hacia instalaciones remotas (acoplamiento remoto PG-AS) (Página 11419)

Parámetro PHONE_NO

Descripción

El parámetro de entrada PHONE_NO indica el número de teléfono con el que debe establecerse una conexión remota. Indique el número de teléfono completo, incluidos los

prefijos y el número de conexión. El string se transmite al módem sin modificaciones. Si utiliza caracteres no numéricos, asegúrese de que el módem utilizado los soporta.

Para establecer una conexión remota, el TS Adapter utiliza los siguientes valores parametrizados:

- Ubicación: Sistema de marcación, prefijo de línea externa
- Ajustes de llamada: "Espere señal antes de marcar", "Número de intentos de marcado" y "Tiempo de espera tras intento de marcado sin éxito"

Parámetro EVENT_ID

Descripción

El parámetro de entrada EVENT_ID indica el ID de evento. Éste se transmite con transparencia desde el programa de usuario del sistema de automatización al programa de usuario de la PG/el PC a través de "PG_DIAL", el TS Adapter y la aplicación TeleService de la PG/el PC. La ID de evento puede tener una estructura cualquiera y transmitir cualquier información del sistema de automatización a la PG/el PC.

Si en la llamada de "PG_DIAL" se transmite una longitud inferior a 16, los bytes restantes de la matriz se rellenan con B#16#00.

Parámetro BUSY

Descripción

La instrucción "PG_DIAL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas.

- Si el parámetro de salida BUSY = 1, en el parámetro de salida STATUS (Página 8282) se indica el estado interno de "PG_DIAL".
- Cuando la ejecución ha finalizado, se indica con el parámetro de salida BUSY = 0. El parámetro de salida STATUS (Página 8282) indica si la petición ha finalizado sin errores (STATUS = W#16#0000) o con errores.

Parámetro STATUS

Descripción

Los valores de retorno de "PG_DIAL" pueden clasificarse del siguiente modo:

- W#16#0000: "PG_DIAL" ha finalizado correctamente
- W#16#7xxx: estado de "PG_DIAL"
- W#16#8xxx: en la llamada interna de una instrucción de comunicación o de la instrucción "BLKMOV" se ha notificado un error

- W#16#9xxx: error de parámetro en la llamada de "PG_DIAL"
- W#16#Bxxx: el TS Adapter ha notificado un error

La tabla siguiente muestra los valores de retorno de "PG_DIAL" a excepción del código de error de las instrucciones de comunicación utilizadas.

Valor de retorno (W#16#...)	Explicación	Indicaciones
0000	La ejecución de "PG_DIAL" ha finalizado sin errores.	
7000	"PG_DIAL" se ha borrado (entretanto se ha deshecho la comunicación con el TS Adapter).	Vuelva a llamar "PG_DIAL".
7001	"PG_DIAL" está activa (primera llamada, BUSY = 1). La función se acaba de iniciar.	
7002	"PG_DIAL" está activa (llamada siguiente, BUSY = 1). La ejecución de la función todavía no ha finalizado.	
8xxx o bien 8zxx	La ejecución de "PG_DIAL" ha finalizado con un código de error de las instrucciones de comunicación llamadas internamente. Si el mensaje de error procede de la instrucción "BLKMOV" significa • z = 2: error al copiar el parámetro PHONE_NO en el búfer interno • z = 3: error al copiar el parámetro EVENT_ID en el búfer interno	Encontrará más información en la información de error de las instrucciones de comunicación: Auto-Hotspot
9001	Longitud de PHONE_NO = 0 ó > 31	El número de teléfono debe contener como mínimo 1 carácter y como máximo 31.
9002	Longitud de EVENT_ID = 0 ó > 16	La ID de evento debe contener como mínimo 1 carácter y como máximo 16 caracteres.
B000	Se ha podido establecer la conexión remota con la programadora o el PC. Sin embargo, en TeleService no se ha registrado ningún programa para la ID de evento transmitido.	
B10A	En el TS Adapter ya hay una petición guardada, que aún está siendo ejecutada por dicho adaptador.	Vuelva a llamar "PG_DIAL".
B10B	Tras el establecimiento de la conexión remota, ésta ha sido desconectada por el interlocutor antes de que se pudiera transmitir la ID de evento.	Vuelva a llamar "PG_DIAL".
B206	No se ha podido establecer la conexión remota con la programadora o el PC.	Compruebe la parametrización del módem en el TS Adapter (local y remoto).
B302	El TS Adapter está conectado directamente con la programadora o el PC (conexión directa).	Finalice la comunicación vía conexión directa.
B305	Todavía hay una conexión remota establecida en la programadora o el PC.	Vuelva a llamar "PG_DIAL" más tarde.

AS_DIAL: Establecer una conexión remota con el AS

Descripción de AS_DIAL

Definición

Sistema de automatización local y remoto

- Local: El sistema de automatización S7 del que parte la iniciativa de establecer la conexión remota se denomina local.
- Remoto: El sistema de automatización S7 con el que debe establecerse la conexión remota se denomina remoto.

Descripción

La instrucción "AS_DIAL" permite establecer una conexión remota entre un sistema de automatización S7 local y uno remoto y a continuación intercambiar datos de proceso. Los datos de proceso se pueden intercambiar con las instrucciones "X_GET (Página 8273)", "X_PUT (Página 8275)" y "X_SEND (Página 8268)".

La instrucción "AS_DIAL" cuenta con dos funciones:

- **Función DIAL:** Establecimiento de una conexión remota con un TS Adapter remoto. Esta función se solicita con el parámetro de entrada REQ_DIAL (Página 8287).
- **Función HANGUP:** Finalización de una conexión remota existente con un TS Adapter remoto. Esta función se solicita con el parámetro de entrada REQ_HANGUP (Página 8287).

Sólo puede haber una función activa. Solicitando la función HANGUP se cancela la función DIAL.

Si se produce un error durante la ejecución, que provoca su interrupción, el código de error se transmite a la parte que llama la instrucción "AS_DIAL". El TS Adapter deshace cualquier conexión remota que pueda haber establecida.

Llamada de "AS_DIAL"

"AS_DIAL" puede llamarse cíclicamente o en un programa controlado por tiempo. Si ejecuta "AS_DIAL" en diferentes niveles de ejecución, tiene que garantizar un bloqueo de llamada entre niveles.

- En la llamada de "AS_DIAL" es imprescindible indicar un DB de instancia. Para una conexión remota siempre debe utilizarse la misma instancia.
  Excepción: Si la CPU local se comunica a través de varios TS Adapter locales, hay que asignar una instancia propia a cada TS Adapter.
- Para procesar una función de bloque son necesarias varias llamadas de "AS_DIAL". Por ello, la llamada de "AS_DIAL" en un "bucle de espera" no es conveniente. El final de la función de bloque se indica con BUSY = 0.
- Para la comunicación con el TS Adapter, "AS_DIAL" también utiliza instrucciones de comunicación para conexiones S7 no configuradas.
  Por ello, en el programa de usuario local hay que garantizar que al establecer y deshacer la conexión remota con "AS_DIAL" no haya ninguna instrucción de comunicación para conexiones S7 no configuradas ("X_PUT (Página 8275)", "X_GET (Página 8273)", "X_SEND (Página 8268)", "X_RCV (Página 8269)", "X_ABORT (Página 8277)") que haya sido iniciada con DEST_ID = "dirección MPI del TS Adapter local".

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "AS_DIAL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
ADDR_TS_ ADAPTER	Input	INT	I, Q, M, D, L o constante	Parámetro de entrada de la función DIAL y HANGUP: Dirección MPI del TS Adapter en la red MPI local
PHONE_NO (Página 8286)	Input	ANY	D	Parámetro de entrada de la función DIAL: Número de teléfono del módem del sistema de automatización remoto (máx. 31 caracteres)
LOGIN	Input	ANY	D	Parámetro de entrada de la función DIAL: Nombre de usuario para la legitimación en el TS Adapter remoto (máx. 8 caracteres)
PASSWORD	Input	ANY	D	Parámetro de entrada de la función DIAL: Contraseña para la legitimación en el TS Adapter remoto (máx. 8 caracteres)
ADDR_CPU	Input	INT	I, Q, M, D, L o constante	Parámetro de entrada de la función DIAL: Dirección MPI de la CPU en la red MPI remota con la que debe establecerse la conexión
REQ_DIAL (Página 8287)	Input	BOOL	I, Q, M, D	Parámetro de entrada para la solicitud de la función DIAL: Establecimiento de una conexión remota con un TS Adapter remoto

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
REQ_HANGUP (Página 8287)	Input	BOOL	I, Q, M, D	Parámetro de entrada para la solicitud de la función HANGUP: Deshacer una conexión remota existente con un TS Adapter remoto
STATUS (Página 8288)	Output	INT	I, Q, M, D, L	Valor de retorno de "AS_DIAL"
BUSY (Página 8288)	Output	BOOL	I, Q, M, D, L	• BUSY = 1: la función DIAL o HANGUP todavía está activa, de modo que el establecimiento/la finalización de la conexión remota todavía no ha finalizado. • BUSY = 0: la ejecución de la instrucción "AS_DIAL" ha finalizado. Ya no hay ninguna función activa.

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros de bloque virtuales EN y ENO y el bit BR

Los parámetros de bloque virtuales EN y ENO sólo aparecen en caso de integración de "AS_DIAL" en la representación KOP o FUP.

- Parámetro de entrada EN: El estado del resultado lógico (RLO) se almacena durante la llamada del bloque en el bit BR.
- Parámetro de salida ENO:
  Si "AS_DIAL" se ejecuta sin errores, el estado del bit BR disponible en el momento de la llamada del bloque se restablece al final de la misma.
  Cuando se indica un mensaje de error a través del parámetro de salida STATUS (Página 8288), el bit BR se pone a "0" inmediatamente antes de abandonar "AS_DIAL".

Consulte también

Intercambio de datos entre instalaciones remotas (acoplamiento remoto AS-AS) (Página 11421)

Parámetro PHONE_NO

Descripción

El parámetro de entrada PHONE_NO indica el número de teléfono con el que debe establecerse una conexión remota. Indique el número de teléfono completo, incluidos los prefijos de país y población y el número de abonado. El string se transmite al módem sin modificaciones. Si utiliza caracteres no numéricos, asegúrese de que el módem utilizado los soporta.

Para establecer una conexión remota, el TS Adapter utiliza los siguientes valores parametrizados:

- Ubicación: sistema de marcación, prefijo de línea externa
- Ajustes de llamada: "Espere señal antes de marcar", "Número de intentos de marcado" y "Tiempo de espera tras intento de marcado sin éxito"

Parámetro REQ_DIAL

Descripción

La función DIAL sólo puede solicitarse si en la última llamada de "AS_DIAL" se notificó BUSY (Página 8288) = 0. Con el parámetro de entrada REQ_DIAL se inicia la función DIAL. El parámetro de entrada REQ_DIAL se dispara por nivel.

REQ_DIAL lanza la función DIAL de la instrucción "AS_DIAL". La instrucción "AS_DIAL" comprueba la plausibilidad de los parámetros y los transmite al TS Adapter local.

Si se detecta un error durante la ejecución de la función DIAL, la ejecución de la función DIAL se cancela. "AS_DIAL" ejecuta automáticamente la función HANGUP (Página 8287) y a continuación notifica el mensaje de error en el valor de retornoSTATUS (Página 8288).

Resultado: Con BUSY (Página 8288) = 0, el resultado del establecimiento de conexión se indica en el valor de retorno STATUS (Página 8288).

Parámetro REQ_HANGUP

Descripción

Con el parámetro de entrada REQ_HANGUP se inicia la función HANGUP. La función HANGUP termina una conexión remota con el TS Adapter remoto que se había establecido anteriormente con la función DIAL (Página 8286).

El parámetro de entrada REQ_HANGUP se dispara por nivel. La función HANGUP también puede iniciarse aunque ya se haya iniciado la función DIAL (Página 8286).

La finalización de la función HANGUP se notifica a través del parámetro de salida BUSY (Página 8288) = 0. El parámetro de salida STATUS (Página 8288) indica entonces el resultado de la ejecución de la función.

Valor	Significado
W#16#0000	La conexión remota ha finalizado de forma regular con la función HANGUP.
W#16#3007	El TS Adapter local ha finalizado la conexión remota porque se ha producido un error.
W#16#3008	La conexión remota se ha interrumpido de forma espontánea o el TS Adapter remoto ha colgado.

Parámetro BUSY

Descripción

La instrucción "AS_DIAL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas.

- Si BUSY = 1, en el parámetro de salida STATUS (Página 8288) se indica el STATUS (Página 8288) interno de "AS_DIAL". Se requiere como mínimo una nueva llamada de "AS_DIAL".
- Si BUSY = 0, la ejecución ha finalizado. El parámetro de salida STATUS (Página 8288) indica si la petición ha finalizado sin o con errores.

Parámetro STATUS

Descripción

Los valores de retorno de "AS_DIAL" pueden clasificarse del siguiente modo:

- W#16#0000: "AS_DIAL" ha finalizado sin errores
- W#16#3xxx: último estado de la conexión remota en el TS Adapter
- W#16#7xxx: estado de "AS_DIAL"
- W#16#8xxx: en la llamada interna de una instrucción de comunicación ("X_ABORT (Página 8277)", "X_GET (Página 8273)", "X_PUT (Página 8275)", "X_RCV (Página 8269)" y "X_SEND (Página 8268)") se ha notificado un error.
- W#16#9xxx: error de parámetro en la llamada de "AS_DIAL"
- W#16#Bxxx: Error al establecer la conexión.

La tabla siguiente muestra los valores de retorno de "AS_DIAL" a excepción del código de error de las instrucciones de comunicación utilizadas:

Valor de retorno (W#16#...)	Explicación	Indicaciones
0000	No hay ninguna función activa. La ejecución de "AS_DIAL" ha finalizado sin errores.	Este mensaje sólo se recibe una vez al finalizar correctamente la función correspondiente.
3007	El TS Adapter local ha colgado.	Sólo lo indica la función HANGUP.
3008	La conexión remota se ha interrumpido de forma espontánea o el TS Adapter remoto ha colgado.	Sólo lo indica la función HANGUP.
7000	No hay ninguna función activa.	Ni se ha ejecutado una función ni se ha solicitado a través de los parámetros de entrada.
7001	La función DIAL está activa (primera llamada, BUSY =1). La función se acaba de iniciar.	Hasta el final de la función (BUSY = 0) no se evalúa el parámetro de entrada REQ_DIAL.
7002	La función DIAL está activa (siguiente llamada, BUSY =1). La ejecución de la función todavía no ha finalizado.	Hasta el final de la función (BUSY = 0) no se evalúa el parámetro de entrada REQ_DIAL.

Valor de retorno (W#16#...)	Explicación	Indicaciones
7701	La función HANGUP está activa (primera llamada). La función se acaba de iniciar.	Hasta el final de la función (BUSY = 0) no se evalúan los parámetros de entrada REQ_DIAL ni REQ_HANGUP.
7702	La función HANGUP está activa (llamada siguiente). La ejecución de la función todavía no ha finalizado.	Hasta el final de la función (BUSY = 0) no se evalúan los parámetros de entrada REQ_DIAL ni REQ_HANGUP.
8xxx	Mensaje de error: La ejecución de la función de bloque activa se ha cancelado debido a un mensaje de error (RET_VAL < 0) de una instrucción llamada internamente en un bloque.	Encontrará más información en la información de error de las instrucciones de comunicación: Auto-Hotspot.
9001	Longitud de PHONE_NO = 0 ó > 31	El número de teléfono debe contener como mínimo 1 carácter y como máximo 31.
9003	Longitud de LOGIN > 8	El nombre de usuario para la legitimación en el TS Adapter remoto no puede tener más de 8 caracteres.
9004	Longitud de PASSWORD > 8	La contraseña para la legitimación en el TS Adapter remoto no puede tener más de 8 caracteres.
B100	El TS Adapter no puede establecer ninguna conexión remota en este momento.	El TS Adapter no puede establecer ninguna conexión con el módem en este momento. Vuelva a llamar "AS_DIAL".
B10A	En el TS Adapter ya hay guardada una ID de evento de una función "PG_DIAL" o hay una función "AS_DIAL" activa que aún debe transmitirse.	Vuelva a llamar "AS_DIAL".
B10B	Tras el correcto establecimiento de la conexión remota, ésta ha vuelto a desconectarse.	Vuelva a llamar "AS_DIAL".
B206	No se ha podido establecer la conexión remota con el TS Adapter remoto.	Compruebe la parametrización del módem en el TS Adapter (local y remoto).
B20A	El tipo de TS Adapter remoto no soporta un acoplamiento remoto inter-AS.	Utilice un tipo de TS Adapter remoto que soporte el acoplamiento remoto inter-AS.
B20B	La instrucción "AS_DIAL" y la versión del TS Adapter son incompatibles.	Utilice un tipo de TS Adapter remoto que soporte el acoplamiento remoto inter-AS.
B20C	El TS Adapter remoto no se integra en la red MPI. Causa: Error de parámetros de red	Compruebe la parametrización de los parámetros de red en el TS Adapter remoto.
B20D	La versión del TS Adapter remoto no soporta un acoplamiento remoto inter-AS.	Utilice un tipo de TS Adapter remoto que soporte el acoplamiento remoto inter-AS.
B20E	Se ha interrumpido la transmisión de datos con el TS Adapter remoto. Causa: Error de transmisión	Vuelva a llamar "AS_DIAL".
B20F	El TS Adapter remoto no ha devuelto la llamada.	Compruebe la parametrización de la protección de acceso (número de llamada) y del módem en el TS Adapter remoto.
B210	Fallo en la transmisión al TS Adapter remoto. Causa: Timeout	Vuelva a llamar "AS_DIAL".
B252	El TS Adapter remoto rechaza la ejecución de la petición, ya que falta la legitimación.	No ha indicado ningún nombre de usuario ni contraseña en la llamada de "AS_DIAL".

Valor de retorno (W#16#...)	Explicación	Indicaciones
B253	El TS Adapter remoto rechaza la legitimación. Causa: Nombre de usuario desconocido	Compruebe los parámetros de llamada de "AS_DIAL" en su programa de usuario.
B254	El TS Adapter remoto rechaza la legitimación. Causa: Contraseña incorrecta	Compruebe los parámetros de llamada de "AS_DIAL" en su programa de usuario.
B302	El TS Adapter está conectado directamente con la programadora o el PC (conexión directa).	Finalice la comunicación vía conexión directa.
B305	Todavía hay una conexión remota con una PG/un PC establecida por el TS Adapter local.	Vuelva a llamar "AS_DIAL" más tarde.

SMS_SEND: Enviar un mensaje SMS

Descripción de SMS_SEND

Descripción

La instrucción "SMS_SEND" transmite un número de teléfono, un número de servicio y un mensaje SMS a un TS Adapter. El TS Adapter transmite esos datos mediante comandos GSM a un módem de radiofrecuencia.

Tras el envío del mensaje SMS, el TS Adapter recibe un acuse, que se transmite a la instrucción "SMS_SEND". A continuación se termina de ejecutar "SMS_SEND" y a la parte que llama se le transmite el estado. Este estado no es más que un acuse del envío del SMS, pero no es una confirmación de recepción.

Si se produce un error durante la ejecución, que provoca su interrupción, el código de error se transmite a la parte que llama la instrucción "SMS_SEND".

Llamada de "SMS_SEND"

La instrucción "SMS_SEND" puede llamarse cíclicamente o en un programa controlado por tiempo.

En la llamada de "SMS_SEND" es imprescindible indicar un DB de instancia.

Para procesar una función de bloque son necesarias varias llamadas de "SMS_SEND". Por ello, la llamada de "SMS_SEND" en un "bucle de espera" no es conveniente. El final de la función de bloque se indica con BUSY = 0.

Interrupción de la conexión

Si la CPU pasa al estado operativo STOP mientras "SMS_SEND" está activa, la conexión con el TS Adapter se deshace. La conexión con el TS Adapter también se pierde cuando en el bus MPI se producen problemas de comunicación graves o, por ejemplo, si la CPU se desconecta.

En esos casos el TS Adapter no rechaza el número de teléfono, el número del centro de servicios ni el mensaje SMS que ya le habían sido transmitidos. El TS Adapter transmite los

datos al módem de radiofrecuencia. Sin embargo, el TS Adapter rechaza el acuse que le envía el módem de radiofrecuencia.

Si un programa de usuario instalado en la CPU intenta enviar un SMS mientras el TS Adapter se encuentra en el estado arriba descrito, "SMS_SEND" finaliza con el valor de retornoW#16#B10A. El programa de usuario puede repetir el envío de un SMS más adelante. El valor de retorno W#16#B10A se obtiene también cuando varias CPUs intentan enviar un SMS simultáneamente a través del mismo TS Adapter.

ATENCIÓN
Modificación del programa de usuario STEP 7 Las partes del programa de usuario STEP 7 con efecto directo sobre llamadas de "SMS_SEND" sólo pueden modificarse con la CPU en el estado operativo STOP. Esto se refiere en especial al borrado e intercambio de bloques de programa que contienen llamadas de "SMS_SEND". En caso de incumplimiento, es posible que los recursos de conexión permanezcan ocupados. El sistema de automatización puede entrar en un sistema no definido debido a las instrucciones de comunicación para conexiones S7 no configuradas ("X_PUT (Página 8275)", "X_GET (Página 8273)", "X_SEND (Página 8268)", "X_RCV (Página 8269)", "X_ABORT (Página 8277)"). Una vez transmitidos los cambios es necesario ejecutar un rearranque completo (en caliente) o un arranque en frío de la CPU.

Coherencia de datos

Los parámetros de entrada del bloque de función se copian en un búfer interno en la primera llamada de "SMS_SEND". No modifique estos datos antes de que la primera llamada haya finalizado (valor de respuesta W#16#7001) ya que de lo contrario pueden transmitirse datos incoherentes.

Transmisión del PIN al módem de radiofrecuencia

Si utiliza una tarjeta SIM con comprobación de PIN activada, el PIN debe transmitirse al módem de radiofrecuencia en la fase de inicialización. Esto es posible con el string Init del TS Adapter, que puede parametrizarse con TeleService. Parametrice el string Init de la siguiente manera (ejemplo para PIN = 4711):

- AT+CPIN="4711";AT&F.....

Tras recuperarse la tensión, el TS Adapter envía una vez el string "AT+CPIN="4711" con el PIN de la tarjeta SIM integrada en el módem de radiofrecuencia al componente conectado. A continuación, el módem se inicializa con "AT&F...".

Atención: Si el PIN es erróneo, no se notifica en la inicialización, sino al enviar un SMS, a través de la respuesta de "SMS_SEND".

Enviar un fax

Anteponiendo un número especial del proveedor de la red (p. ej. "99") al número en sí, el proveedor convierte el SMS en un fax y lo envía a un aparato de fax. Se trata de un servicio del proveedor de la red, y no de una función del TS Adapter.

El número de fax se indica con el prefijo del proveedor de la red en el parámetro PHONE_NO.

Ejemplo de una red con el número de fax 07214711:

- PHONE_NO = '9907214711'

Envío de un correo electrónico

Marcando un número especial del proveedor de la red (p. ej. "8000") y anteponiendo una dirección de correo electrónico al mensaje SMS es posible enviar el SMS a una dirección de correo electrónico. Se trata de un servicio del proveedor de la red, y no de una función del TS Adapter.

El número correspondiente se transmite en el parámetro PHONE_NO (Página 8293).

La dirección de correo electrónico se antepone al texto en sí del mensaje SMS, y se separa de él con un carácter de separación. La longitud máxima del texto útil del SMS se reduce en la dirección de correo electrónico + el carácter de separación. Los separadores son, p. ej. un espacio (" ") o dos puntos (":").

En lugar del carácter "@", algunos proveedores de red requieren un "*".

Ejemplo:

- Se pretende enviar el texto 'CPU in STOP' a la dirección de correo electrónico "Surname*provider.com".
- PHONE_NO = 8000
- MESSAGE = Surname*provider.comCPU in STOP

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "SMS_SEND":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
MPI_TS_ ADAPTER	Input	INT	I, Q, M, D, L o constante	Dirección MPI del TS Adapter
PHONE_NO (Página 8293)	Input	ANY	D	Referencia a un string de datos con una longitud máxima de 31 caracteres
SCENTER_NO (Página 8293)	Input	ANY	D	Referencia a un string de datos con una longitud máxima de 20 caracteres
MESSAGE (Página 8293)	Input	ANY	D	Referencia a un string de datos con una longitud máxima de 160 caracteres
BUSY (Página 8294)	Output	BOOL	I, Q, M, D, L	• BUSY=1: el envío del SMS todavía no ha finalizado. • BUSY=0: la ejecución de "SMS_SEND" ha finalizado
STATUS (Página 8294)	Output	INT	I, Q, M, D, L	Valor de retorno de "SMS_SEND"

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Parámetros virtuales EN y ENO y el bit BR

Los parámetros de bloque virtuales EN y ENO sólo aparecen en caso de integración de "SMS_SEND" en la representación KOP o FUP.

- Parámetro de entrada EN:
  El estado del resultado lógico (RLO) se almacena durante la llamada del bloque en el bit BR.
- Parámetro de salida ENO:
  Si no ha habido errores en la ejecución de "SMS_SEND", el estado del bit BR que había en el momento de la llamada del bloque se restablece al final de dicha llamada.
  Si se emite un mensaje de error a través del parámetro de salida STATUS, el bit BR se pone a "0" inmediatamente antes abandonar "SMS_SEND".

Consulte también

Enviar un SMS desde una instalación (Página 11423)

Parámetro PHONE_NO

Descripción

El parámetro de entrada PHONE_NO indica el número de teléfono al que debe enviarse un SMS. Indique el número de teléfono completo, incluidos los prefijos de país y población y el número de abonado. El string se transmite al módem de radiofrecuencia sin modificaciones. Si utiliza caracteres no numéricos, asegúrese de que el módem de radiofrecuencia utilizado los soporta.

Parámetro SCENTER_NO

Descripción

El parámetro de entrada SCENTER_NO indica el número del service center a través del cual debe enviarse el SMS. El string se transmite al módem de radiofrecuencia sin modificaciones.

Parámetro MESSAGE

Descripción

El parámetro de entrada MESSAGE indica el mensaje SMS. Éste se transmite a través de la instrucción "SMS_SEND", el TS Adapter y el módem de radiofrecuencia GSM al proveedor de red.

Parámetro BUSY

Descripción

La instrucción "SMS_SEND" es una instrucción asíncrona, de modo que la ejecución de la petición se prolonga a lo largo de varias llamadas.

- Si el parámetro de salida BUSY = 1, en el parámetro de salida STATUS (Página 8294) se indica el estado interno de "SMS_SEND".
- Cuando la ejecución ha finalizado, se indica con el parámetro de salida BUSY = 0. El parámetro de salida STATUS (Página 8294) indica si la petición ha finalizado sin errores (STATUS = W#16#0000) o con errores.

Parámetro STATUS

Descripción

Los valores de retorno de "SMS_SEND" pueden clasificarse del siguiente modo:

- W#16#0000: "SMS_SEND" ha finalizado correctamente
- W#16#7xxx: estado de "SMS_SEND"
- W#16#8xxx: en la llamada interna de una instrucción de comunicación o de la instrucción "BLKMOV" se ha notificado un error
- W#16#9xxx: error de parámetro en la llamada de "SMS_SEND"
- W#16#Bxxx: el TS Adapter ha notificado un error
- W#16#Cxxx: el módem de radiofrecuencia ha notificado un error

La tabla siguiente muestra los valores de retorno de "SMS_SEND" con la excepción del código de error de las instrucciones de comunicación utilizadas, así como el del módem de radiofrecuencia.

Valor de retorno (W#16#...)	Explicación	Indicaciones
0000	La ejecución de "SMS_SEND" ha finalizado sin errores.	
7000	"SMS_SEND" se ha borrado (entretanto se ha deshecho la comunicación con el TS Adapter).	Vuelva a llamar "SMS_SEND".
7001	"SMS_SEND" está activa (primera llamada, BUSY = 1). La función se acaba de iniciar.	
7002	"SMS_SEND" está activa (llamada siguiente, BUSY = 1). La ejecución de la función todavía no ha finalizado.	

Valor de retorno (W#16#...)	Explicación	Indicaciones
8xxx o bien 8zxx	La ejecución de "SMS_SEND" ha finalizado con un código de error de las instrucciones de comunicación llamadas internamente o de la instrucción "BLKMOV". Si el mensaje de error procede de "BLKMOV" significa • z = 2 : error al copiar el parámetro PHONE_NO en el búfer interno • z = 3 : error al copiar el parámetro SCENTER_NO en el búfer interno • z = 4 : error al copiar el parámetro MESSAGE en el búfer interno	Encontrará información detallada en la descripción de la instrucción "BLKMOV", así como en la información de errores de las instrucciones de comunicación: Auto-Hotspot
9001	Longitud de PHONE_NO = 0 ó > 31	El número de teléfono debe contener como mínimo 1 carácter y como máximo 31.
9002	Longitud de SCENTER_NO = 0 ó > 20	El número del service center debe contener como mínimo 1 carácter y como máximo 20.
9003	Longitud de MESSAGE = 0 ó > 160	El mensaje SMS debe contener como mínimo 1 carácter y como máximo 160 caracteres.
B10A	En el TS Adapter ya hay una petición guardada, que aún está siendo ejecutada por dicho adaptador.	Vuelva a llamar "SMS_SEND".
B301	Todavía hay una conexión remota establecida en la programadora o el PC.	Vuelva a llamar "SMS_SEND" más tarde.
B302	El TS Adapter está conectado directamente con una PG/un PC (conexión directa)	Conecte el TS Adapter con un módem de radiofrecuencia GSM
B303	El TS Adapter no está conectado ni con un módem de radiofrecuencia ni con una PG/un PG.	Conecte el TS Adapter con un módem de radiofrecuencia GSM
B304	La interfaz para el módem de radiofrecuencia no está lista en este momento para enviar un SMS.	Vuelva a llamar "SMS_SEND" más tarde.
B614	El módem conectado no ha respondido al comando específico de SMS dentro del tiempo de vigilancia.	Compruebe si hay un módem de radiofrecuencia GSM conectado. Vuelva a llamar "SMS_SEND" más tarde.
B615	El TS Adapter ha recibido un mensaje de error no específico del módem conectado.	Compruebe si hay un módem de radiofrecuencia GSM conectado.
Cxxx	La ejecución de "SMS_SEND" se ha interrumpido con un código de error del módem GSM. xxx: Número de error	Encontrará una lista de los números de error en el manual de su módem de radiofrecuencia o en los estándares GSM GSM 04.11, GSM 03.40 y GSM 07.05. Ejemplos: • C136 = falta la tarjeta SIM del módem GSM • C137 = el PIN del módem GSM no es correcto (véase Transferencia del PIN al módem de radiofrecuencia) • C001 = número de service center erróneo

AS_MAIL: Transmitir un mensaje de correo electrónico

Descripción de AS_MAIL

Instrucciones necesarias

La instrucción "AS_MAIL" transmite un mensaje de correo electrónico desde una CPU a un servidor de correo a través del Simple Mail Transfer Protocol (SMTP). "AS_MAIL" requiere las siguientes instrucciones para su ejecución:

- "S_COMP: Comparar cadenas de caracteres (Página 6715)"
- "FIND: Buscar caracteres en una cadena (Página 6728)"
- "INSERT: Insertar caracteres en una cadena (Página 6727)"
- "LEFT: Leer los caracteres izquierdos de una cadena (Página 6724)"
- "LEN: Determinar la longitud de una cadena de caracteres (Página 6722)"
- "RIGHT: Leer los caracteres derechos de una cadena (Página 6724)"

Descripción

La instrucción "AS_MAIL" es una instrucción asíncrona, de modo que la ejecución se prolonga a lo largo de varias llamadas. En la llamada de la instrucción "AS_MAIL" es imprescindible indicar un DB de instancia.

El bloque se inicializa con una llamada con COM_RST = 1.

El envío de un mensaje de correo electrónico se inicia con un cambio de flanco de "0" a "1" en el parámetro REQ.

A través de los parámetros de salida "BUSY", "DONE", "ERROR", así como "STATUS" y "SFC_STATUS" se indica el estado de la petición. "SFC_STATUS" corresponde al parámetro de salida "STATUS" de los bloques de comunicación llamados, o bien a la instrucción "BLKMOV".

Los parámetros de salida DONE, ERROR, STATUS y SFC_STATUS sólo se indican durante un ciclo si el estado del parámetro de salida BUSY cambia de "1" a "0".

En la tabla siguiente se muestra la correspondencia entre BUSY, DONE y ERROR. Permite determinar en qué estado se encuentra la instrucción "AS_MAIL" actualmente o cuándo se completó el envío del mensaje de correo electrónico.

BUSY	DONE	ERROR	Descripción
1	Irrelevante	Irrelevante	La petición está en proceso.
0	1	0	La petición ha finalizado correctamente.
0	0	1	La petición se ha finalizado con un error. La causa del error se puede consultar en los parámetros STATUS y SFC_STATUS.
0	0	0	No se ha asignado ninguna (nueva) petición a la instrucción "AS_MAIL".

Si la CPU pasa al estado operativo STOP mientras "AS_MAIL" está activa, la conexión con el servidor de correo se deshace. La conexión con el servidor de correo también se pierde si aparecen problemas de comunicación en el bus Industrial Ethernet. En esos casos, el envío del mensaje de correo electrónico se cancela y éste no llega al receptor.

ATENCIÓN
Cambiar programas de usuario Las partes del programa de usuario con efecto directo sobre llamadas de "AS_MAIL" sólo pueden modificarse en los siguientes casos: • Si la CPU está en estado operativo "STOP" • Si no se envía ningún mensaje (REQ = 0 y BUSY = 0). Esto se refiere en especial al borrado e intercambio de bloques de programa que contienen llamadas de "AS_MAIL" o llamadas del DB de instancia de "AS_MAIL" En caso de incumplimiento, es posible que los recursos de conexión permanezcan ocupados. El sistema de automatización puede pasar a un estado no definido con las funciones de comunicación TCP/IP a través de Industrial Ethernet. Una vez transmitidas las modificaciones es necesario ejecutar un rearranque completo (en caliente) o un arranque en frío de la CPU o activar el parámetro COM_RST de la instrucción "AS_MAIL".

Coherencia de datos

El parámetro de entrada ADDR_MAIL_SERVER de la instrucción se aplica de nuevo cada vez que se lanza el envío de un mensaje de correo electrónico de la instrucción "AS_MAIL". Si se produce un cambio durante el funcionamiento, el valor "nuevo" no tendrá efecto hasta que no se vuelva a lanzar un mensaje de correo electrónico.

Por el contrario, los parámetros WATCH_DOG_TIME, TO_S, CC, FROM, SUB, TEXT, ATTACHMENT, y, dado el caso, USERNAME y PASSWORD se aplican durante el funcionamiento de la instrucción "AS_MAIL", por lo que no pueden modificarse hasta que la petición no haya finalizado (BUSY = 0)

Parametrización del TS Adapter IE

En el TS Adapter IE hay que introducir los parámetros de las llamadas salientes de forma que el TS Adapter IE establezca una conexión con el servidor de marcado del proveedor de servicios de Internet.

Si para establecer la conexión se ajusta "Bajo demanda", la conexión se establecerá cuando deba enviarse un mensaje de correo electrónico.

Con una conexión analógica vía módem, el establecimiento de la llamada puede requerir cierto tiempo (aprox. un minuto). El tiempo necesario para el establecimiento de la conexión debe tenerse en cuenta al especificar el parámetro WATCH_DOG_TIME.

Parámetro

La tabla siguiente muestra los parámetros de la instrucción "AS_MAIL":

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
COM_RST	Input / Output	BOOL	I, Q, M, D, L	Parámetro de entrada/salida "COMPLETE" RESTART: el bloque tiene una rutina de inicialización, que se ejecuta cuando la entrada COM_RST está activada. El parámetro COM_RST de la instrucción "AS_MAIL" debe ponerse a "1": • Una vez antes de la primera llamada • Cada vez que se cargue el DB de instancia correspondiente Tras la inicialización, la instrucción "AS_MAIL" desactiva el parámetro COM_RST.
REQ	Input	BOOL	I, Q, M, D, L	Parámetro de control REQUEST: Activa el envío de un mensaje de correo electrónico con un flanco ascendente.
ADDR_MAIL_ SERVER	Input	DWORD	I, Q, M, D, L	Parámetro de entrada Dirección IP del servidor de correo: Debe indicarse en forma de palabra de datos en formato HEX, p. ej.: Dirección IP = 192.168.0.200. ADDR_MAIL_SERVER = DW#16#C0A800C8, teniendo en cuenta lo siguiente: • 192 = 16#C0, • 168 =16#A8 • 0 = 16#00 y • 200 = 16#C8.
WATCH_DOG_ TIME	Input	TIME	I, Q, M, D, L	Parámetro de entrada Intervalo máx.: En el tiempo predeterminado (WATCH_DOG_TIME), la instrucción "AS_MAIL" debería establecer una conexión. Si se supera este tiempo, el bloque se finaliza con un error. El tiempo hasta que el bloque finaliza y el error se notifica puede superar el WATCH_DOG_TIME, ya que el establecimiento de la conexión también requiere tiempo. Para comenzar debe ajustarse un tiempo de 2 minutos. En la conexión telefónica vía ISDN, puede seleccionarse un tiempo notablemente inferior.
USERNAME	Input	ANY	D	Parámetro de entrada Nombre de usuario: Referencia a una variable del tipo STRING con una longitud máxima de 180 caracteres. Para los procesos de autenticación es imprescindible un nombre de usuario.

Parámetro	Declaración	Tipo de datos	Área de memoria	Descripción
PASSWORD	Input	ANY	D	Parámetro de entrada Contraseña: Referencia a una variable del tipo STRING con una longitud máxima de 180 caracteres. Para los procesos de autenticación es imprescindible una contraseña.
TO_S	Input	ANY	D	Parámetro de entrada Direcciones de destinatario: Referencia a una variable del tipo STRING con una longitud máxima de 240 caracteres (véase el ejemplo de llamada).
CC	Input	ANY	D	Parámetro de entrada Direcciones de destinatario CC: Referencia a una variable del tipo STRING con una longitud máxima de 240 caracteres (véase el ejemplo de llamada).
FROM	Input	ANY	D	Parámetro de entrada Dirección de remitente: Referencia a una variable del tipo STRING con una longitud máxima de 240 caracteres (véase el ejemplo de llamada).
SUB	Input	ANY	D	Parámetro de entrada Asunto del mensaje: Referencia a una variable del tipo STRING con una longitud máxima de 240 caracteres.
TEXT	Input	ANY	D	Parámetro de entrada Texto del mensaje: Referencia a un string de datos con una longitud máxima de 240 caracteres.
ATTACHMENT	Input	ANY	I, Q, M, D, L	Parámetro de entrada Anexo del mensaje: Referencia a un campo de byte/palabra/palabra doble con una longitud máxima de 65534 bytes.
BUSY	Output	BOOL	I, Q, M, D, L	• BUSY = 1: el envío del mensaje de correo electrónico todavía no ha finalizado. • BUSY=0: la ejecución de "AS_MAIL" ha finalizado.
DONE	Output	BOOL	I, Q, M, D, L	• DONE = 0: la petición aún no se ha iniciado o todavía se está ejecutando. • DONE = 1: Petición finalizada sin errores.
ERROR	Output	BOOL	I, Q, M, D, L	ERROR = 1: durante la ejecución se ha producido un error. STATUS y SFC_STATUS devuelve información detallada sobre el tipo de error.
STATUS (Página 8301)	Output	INT	I, Q, M, D, L	Parámetro de salida/estado STATUS: Valor de retorno o información de error de la instrucción "AS_MAIL".
SFC_STATUS (Página 8301)	Output	INT	I, Q, M, D, L	Parámetro de salida/estado "SFC_STATUS": Información de error de los bloques de comunicación llamados o de la instrucción "BLKMOV".

Encontrará más información sobre los tipos de datos válidos en "Vista general de los tipos de datos válidos (Página 2321)".

Nota

Indicaciones sobre los parámetros de la instrucción

- Si en el bloque en el que se llama "AS_MAIL" se elimina, se inserta de nuevo o se modifica la asignación de los parámetros USERNAME y PASSWORD, el cambio no tendrá efecto hasta que el DB de instancia correspondiente no se vuelva a cargar y se active el parámetro COM_RST. En cualquier otro caso el puntero hacia el parámetro correspondiente se mantiene en el DB de instancia.
- Por motivos relacionados con el tiempo de ejecución y el espacio en memoria, la instrucción "AS_MAIL" no realiza una prueba de sintaxis de los parámetros TO_S, CC y FROM. Observe la sintaxis correcta de los parámetros.
- Los parámetros opcionales CC, TEXT y ATTACHMENT sólo se envían con el mensaje de correo electrónico si se asignan dentro del bloque en el que se llama la instrucción "AS_MAIL". Si una asignación de este tipo se elimina de la interfaz de llamada de la instrucción "AS_MAIL", el cambio no tendrá efecto hasta que el DB de instancia correspondiente no se vuelva a cargar y el parámetro COM_RST se active. En cualquier otro caso, el puntero hacia el parámetro correspondiente se mantiene en el DB de instancia.
  Recuerde que en una asignación de parámetros no están permitidos los strings vacíos.
- Si el parámetro ATTACHMENT apunta a una matriz de un bloque de datos y se cambia el tamaño de dicha matriz y se guarda, la interfaz de llamada de la instrucción "AS_MAIL" de los punteros ANY del parámetro ATTACHMENT debe volver a registrarse.

Autenticación SMTP

Bajo el término "autenticación" se entiende un proceso con el que se garantiza una identidad, como puede ser una consulta de contraseña.

La instrucción "AS_MAIL" soporta el método de autenticación SMTP AUTH-LOGIN, requerido por la mayoría de servidores de correo. Para más información sobre el método de autenticación de su servidor de correo, consulte el manual del servidor o la página web del proveedor de servicios de Internet.

Para utilizar el método de autenticación AUTH-LOGIN, la instrucción "AS_MAIL" requiere el nombre de usuario con el que poder iniciar sesión en el servidor de correo. Este nombre de usuario equivale al nombre de usuario con el que se ha configurado la cuenta de correo en el servidor de correo. Se da a conocer a través del parámetro USERNAME de la instrucción "AS_MAIL".

Asimismo, para iniciar sesión la instrucción "AS_MAIL" requiere la contraseña correspondiente. Esta contraseña equivale a la contraseña indicada al configurar la cuenta de correo. Se da a conocer a través del parámetro PASSWORD de la instrucción "AS_MAIL".

El nombre de usuario y la contraseña se transfieren sin cifrar (codificación BASE64) al servidor de correo.

Si en el DB no se indica ningún nombre de usuario, no se utiliza el método de autenticación AUTH-LOGIN. El mensaje de correo electrónico se envía entonces sin autenticación.

Consulte también

Enviar un correo electrónico desde una instalación (Página 11424)

Parámetros STATUS y SFC_STATUS

Descripción

Los valores de retorno de la instrucción "AS_MAIL" pueden clasificarse del siguiente modo:

- W#16#0000: "AS_MAIL" ha finalizado correctamente
- W#16#7xxx: estado de "AS_MAIL"
- W#16#8xxx: se ha notificado un error en la llamada interna de un bloque de comunicación, en la llamada de la instrucción "BLKMOV" o por parte del servidor de correo.

La tabla siguiente muestra los valores de retorno de "AS_MAIL" con la excepción del código de error de los bloques de comunicación llamados internamente, así como de la instrucción "BLKMOV".

Valor de retorno STATUS (W#16#...):	Valor de retorno SFC_STATUS (W#16#...):	Explicación	Indicaciones
0000	-	La ejecución de "AS_MAIL" ha finalizado sin errores.	La finalización sin errores de "AS_MAIL" no significa que el mensaje de correo electrónico enviado sea recibido (véase más abajo - Punto 1 de la nota)
7001		"AS_MAIL" está activo (BUSY = 1).	
7002	7002	"AS_MAIL" está activo (BUSY = 1).	
8xxx	xxxx	La ejecución de "AS_MAIL" ha finalizado con un código de error de las instrucciones de comunicación llamadas internamente o de la instrucción "BLKMOV".	Encontrará información detallada sobre la evaluación del parámetro SFC_STATUS en las descripciones del parámetro STATUS de las instrucciones "BLKMOV".

Valor de retorno STATUS (W#16#...):	Valor de retorno SFC_STATUS (W#16#...):	Explicación	Indicaciones
800z	xxxx	El mensaje de error procede de la instrucción "BLKMOV" y significa: • z = 1: error al copiar el parámetro TO_S en el búfer interno • z = 2: error al copiar el parámetro CC en el búfer interno • z = 3: error al copiar el parámetro FROM en el búfer interno • z = 4: error al copiar el parámetro SUB en el búfer interno • z = 5: error al copiar el parámetro TEXT en el búfer interno • z = 6: error al copiar el parámetro ATTACHMENT en el búfer interno • z = 7: error al copiar el parámetro USERNAME en el búfer interno • z = 8: error al copiar el parámetro PASSWORD en el búfer interno	
8010	xxxx	Error al establecer la conexión.	Posiblemente COM_RST no se activó después de cargar el DB de instancia.
8011	xxxx	Error al enviar los datos.	Encontrará más información sobre la evaluación de SFC_STATUS en las descripciones del parámetro STATUS de la instrucción "TSEND (Página 7753)".
8012	xxxx	Error al recibir los datos.	Encontrará más información sobre la evaluación de SFC_STATUS en las descripciones del parámetro STATUS de la instrucción "TRCV (Página 7756)".
8013	xxxx	Error al establecer la conexión.	Encontrará más información sobre la evaluación de SFC_STATUS en las descripciones del parámetro STATUS de la instrucción "TCON (Página 7748)" y "TDISCON (Página 7751)".
8014	-	No es posible establecer una conexión.	Posiblemente ha introducido una dirección IP del servidor de correo incorrecta (ADDR_MAIL_SERVER) o un intervalo demasiado reducido (WATCH_DOG_TIME) para establecer la conexión. También existe la posibilidad de que la CPU no tenga conexión con la red o que la configuración de la CPU sea incorrecta.

Valor de retorno STATUS (W#16#...):	Valor de retorno SFC_STATUS (W#16#...):	Explicación	Indicaciones
82xx, 84xx, o bien 85xx	-	El mensaje de error procede del servidor de correo y, exceptuando el "8", corresponde al número de error del protocolo SMTP. En las siguientes columnas se indican algunos códigos de error posibles:	Véase el punto 2 de la nota.
8450	-	Acción no ejecutada: Buzón de correo no disponible/no accesible.	Vuelva a intentarlo más tarde.
8451	-	Acción cancelada: Error local en el procesamiento	Vuelva a intentarlo más tarde.
8500	-	Error de sintaxis: Error no detectado. Ello incluye también el error debido a una cadena de comandos demasiado larga. Otra posible causa es que el servidor de correo electrónico no soporte el método de autenticación LOGIN.	Compruebe los parámetros de "AS_MAIL". Intente enviar un mensaje de correo electrónico sin autenticación. Sustituya para ello el parámetro USERNAME por un string vacío.
8501	-	Error de sintaxis: Parámetro o argumento incorrecto	Posiblemente ha introducido una dirección incorrecta en TO_S o CC.
8502	-	Comando desconocido o no implementado.	Compruebe las entradas realizadas, especialmente el parámetro FROM. Es posible que esté incompleto y que haya olvidado "@" o ".".
8535	-	Autenticación SMTP incompleta.	Puede que haya introducido un nombre de usuario o una contraseña incorrectas.
8550	-	No es posible acceder al servidor de correo, no tiene derechos de acceso.	Posiblemente ha introducido un nombre de usuario o una contraseña incorrectos o el servidor de correo no soporta el LOGIN. Otra posible causa del error puede ser la entrada incorrecta del nombre del dominio después de "@" en TO_S o CC.
8552	-	Acción cancelada: Rebase del espacio de memoria asignado	Vuelva a intentarlo más tarde.
8554	-	Transmisión fallida.	Vuelva a intentarlo más tarde.

Nota

Error de estado

1. La entrada incorrecta de las direcciones de los destinatarios no genera un error de estado de la instrucción "AS_MAIL". En tal caso no es posible garantizar el envío del mensaje de correo electrónico a otros destinatarios, aunque se hayan introducido correctamente.
2. Encontrará más información sobre el código de error SMTP y otros códigos de error en el protocolo SMTP en Internet, o bien en la documentación sobre errores del servidor de correo. También es posible ver el último mensaje de error enviado por el servidor de correo en forma de texto en el DB de instancia en el parámetro BUFFER1. En él encontrará en "Datos" los últimos datos enviados por la instrucción "AS_MAIL".

Ejemplo de llamada de "AS_MAIL"

Descripción

En un bloque de datos se guardan los datos que se dan a conocer a través de punteros ANY de la instrucción "AS_MAIL". En este ejemplo se utiliza para ello el DB 2.

Por motivos relacionados con el tiempo de ejecución y el espacio en memoria, la instrucción "AS_MAIL" no realiza una prueba de sintaxis de los parámetros TO_S, CC y FROM.

Parámetros TO_S, CCy FROM

Los parámetros TO_S, CC y FROM son punteros ANY hacia strings con, por ejemplo, el siguiente contenido:

- TO: <wenna@mydomain.com>, <ruby@mydomain.com>,
- CC: <admin@mydomain.com>, <judy@mydomain.com>,
- FROM: <admin@mydomain.com>

Las siguientes reglas deben observarse a la hora de introducir los parámetros:

- Deben introducirse los caracteres "TO:", "CC:" y "FROM:".
- Antes de cada dirección debe introducirse un espacio y un corchete en "<".
- Después de cada dirección debe introducirse un corchete en ">".
- Después de cada dirección en TO_S y CC debe introducirse una coma.
- En FROM solo puede introducirse una dirección de correo electrónico que además no debe tener una coma al final.

Inicialización / arranque en el OB 100

El parámetro COM_RST de la instrucción "AS_MAIL" debe activarse antes de la primera llamada de la instrucción
, o bien cada vez que se cargue el DB de instancia correspondiente.

En el siguiente ejemplo, DB 1 es el DB de instancia de la instrucción "AS_MAIL".

SET		//Activar RLO
R	DB1.REQ	//REQ se desactiva, de modo que la instrucción "AS_MAIL" no comienza a enviar un mensaje de correo electrónico inmediatamente después de la activación de COM_RST.
S	DB1.COM_RST	//COM_RST se activa; esto tiene que ocurrir una vez antes de la primera llamada de la instrucción "AS_MAIL" para la inicialización, o bien después de cada carga del DB de instancia correspondiente.

Llamada cíclica

Este ejemplo de llamada de la instrucción "AS_MAIL" debe llamarse cíclicamente (p. ej. en el OB1).
El parámetro REQ debe ponerse a "1" en el programa de usuario a través de "Forzar variable".

Esta línea puede copiarse en una fuente AWL y compilarse. Para ello es necesario que los bloques DB 1 y DB 2 hayan sido registrados en la tabla de variables del proyecto.

Ejemplo de llamada de "AS_MAIL"	Explicación
DATA_BLOCK "DB2"	//Bloque de datos global con los datos para la instrucción "AS_MAIL"
TITLE =	
VERSION : 1.0	
STRUCT	
TO_S : STRING [240] := 'TO: <wenna@mydomain.com>, <ruby@mydomain.com>,';	//Dirección del destinatario
CC : STRING [240] := 'CC: <admin@mydomain.com>, <judy@mydomain.com>,';	//Dirección del destinatario CC
FROM : STRING [60] := 'FROM: <admin@mydomain.com>';	//Dirección del remitente
SUB : STRING [60] := 'Status plant 7';	//Asunto
TEXT : STRING [240] := 'Fault in plant 7 sector 2';	//Texto
USERNAME : STRING [60] := 'admin';	//Nombre de usuario para autentificación LOGIN
PASSWORD : STRING [60] := 'test';	//Contraseña para autenticación LOGIN
ATTACHMENT : ARRAY [1 .. 178] OF BYTE;	//Archivos adjuntos
END_STRUCT ;	
BEGIN	
END_DATA_BLOCK	
DATA_BLOCK "DB1"	//DB de instancia de la instrucción "AS_MAIL"
TITLE =	
AUTHOR : TELESERV	
FAMILY : TELESERV	
VERSION : 1.0	
"AS_MAIL"	
BEGIN	
END_DATA_BLOCK	
FUNCTION_BLOCK FB 1	
TITLE =Example AS_MAIL	//El parámetro COM_RST de la instrucción "AS_MAIL" debe activarse antes de la primera llamada de "AS_MAIL", o bien cada vez que se cargue el DB de instancia correspondiente. La

Ejemplo de llamada de "AS_MAIL"	Explicación
	condición de llamada se activa durante la ejecución del programa de usuario.
VERSION : 1.0	
VAR	
CallCondition: BOOL ;	
END_VAR	
BEGIN	
NETWORK	
CALL AS_MAIL, DB1 (	//Llamada de AS_MAIL
REQ := # CallCondition,	//Inicio de la petición con flanco ascendente
ADR_MAIL_SERVER := DW#16#C0A800C8,	//Equivale a la dirección IP 192.168.0.200 en formato HEX
WATCH_DOG_TIME := T#2M,	//Intervalo de 2 min. dentro del cual debería concluir el envío
USERNAME := DB2.USERNAME,	//Puntero al nombre de usuario del remitente. Si DB2.USERNAME se omite, es decir, si no se le asigna ningún parámetro, no se utiliza el proceso de autenticación LOGIN.
PASSWORD := DB2.PASSWORD,	//Puntero hacia contraseña
TO_S := DB2.TO_S,	//Puntero hacia dirección de destinatario
CC := DB2.CC,	//Puntero a dirección de destinatario CC (opcional). Si DB2.CC se omite, es decir, si no se asigna, el mensaje de correo electrónico no se envía a los destinatarios CC.
FROM := DB2.FROM,	//Puntero hacia dirección de remitente
SUB := DB2.SUB,	//Puntero hacia el asunto
TEXT := DB2.TEXT,	//Puntero hacia el texto del mensaje (opcional). Si DB2.TEXT se omite, es decir, si no se asigna, el mensaje de correo electrónico se envía sin texto.
ATTACHMENT := DB2.ATTACHMENT,	//Puntero hacia los archivos adjuntos (opcional). Ejemplo: 101 bytes adjuntos al mensaje - ATTACHMENT := P#DB2.DBX974.0 BYTE 101 (el ejemplo se refiere al DB anterior). //Si DB2.ATTACHMENT se omite, es decir, si no se asigna, el mensaje de correo electrónico se envía sin archivos adjuntos.
BUSY := M46.0,	//Parámetro de salida/estado BUSY
DONE := M46.1,	//Parámetro de salida/estado DONE
ERROR := M46.2,	//Parámetro de salida/estado ERROR
STATUS := MW48,	//Parámetro de salida/estado STATUS
SFC_STATUS := MW50	//Parámetro de salida/estado SFC_STATUS
);	
	//Comprobación de los resultados de la llamada al final de la ejecución
U M 46.0;	//¿es BUSY ==0?
SPB End;	//No, BUSY ==1. "AS_MAIL" todavía está ejecutando la petición

Ejemplo de llamada de "AS_MAIL"	Explicación
`U M 46.1;`	`//¿La ejecución de "AS_MAIL" ha finalizado sin errores?`
`SPB End;`	
`NOP 0;`	`//No, se ha producido un error en la ejecución, BUSY ==0, BR ==0`
	`//Completar aquí valoración del error si procede, STATUS, SFC_STATUS`
`End:BE;`	
`END_FUNCTION_BLOCK`	

Nota

Parámetros opcionales

Para los parámetros opcionales USERNAME, PASSWORD, CC, TEXT y ATTACHMENT rige:

Si se elimina una asignación o se vuelve a insertar, el cambio no tiene efecto hasta que no se vuelve a cargar el DB de instancia correspondiente (en el ejemplo DB 1), y se activa el parámetro "COM_RST".

Visualización de procesos 10

10.1 Crear imágenes

10.1.1 Principios básicos

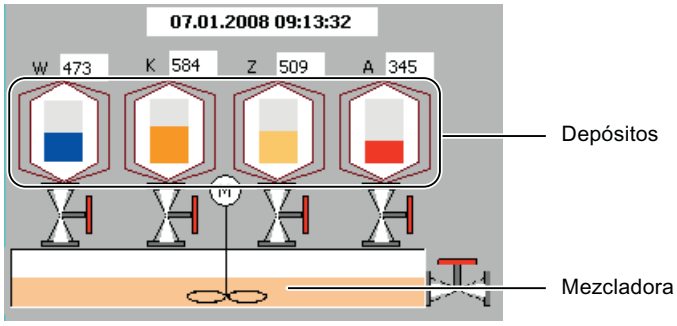
10.1.1.1 Principios básicos de las imágenes

Introducción

WinCC permite crear imágenes para controlar y supervisar máquinas e instalaciones. Para crear imágenes se dispone de objetos predefinidos para reproducir la instalación, visualizar los distintos procesos y preseleccionar valores de proceso.

Ejemplo de aplicación

Esta figura muestra una imagen creada con WinCC. Los operadores de la instalación manejan y visualizan mediante esta imagen una mezcladora para elaborar distintos zumos de fruta. Desde distintos depósitos se llenan y mezclan los distintos componentes del zumo de frutas en una mezcladora. Se indica el nivel de llenado de los depósitos.



Composición de una imagen

Inserte en la imagen los objetos que necesita para representar el proceso. Configure los objetos de acuerdo con las necesidades del proceso.

La imagen se puede componer de elementos estáticos y dinámicos.

- Los elementos estáticos, tales como los textos y los gráficos, no cambian en runtime. En el ejemplo de la mezcladora, son estáticos p. ej. los rótulos de los depósitos (W, K, Z, A).
- Los elementos dinámicos van cambiando en función del proceso. Los valores de proceso actuales se visualizan como se indica a continuación:
  - Desde la memoria del controlador
  - Desde la memoria del panel de operador mediante indicadores alfanuméricos, curvas y barras.

  A los objetos dinámicos pertenecen también los campos de entrada del panel de operador. En el ejemplo de la mezcladora, los niveles de llenado de los depósitos son objetos dinámicos.

El intercambio de valores de proceso y entradas entre el controlador y el panel de operador se realiza mediante variables.

Características de las imágenes

La representación de la imagen depende del panel de operador para el que se configure. La representación equivale al aspecto de la interfaz de usuario del panel de operador. Las propiedades de la imagen, como la resolución, las fuentes y los colores disponibles, dependen también del panel de operador configurado. Si el panel de operador configurado dispone de teclas de función, éstas se visualizan en la imagen.

Una tecla de función en una tecla del panel de operador a la que se pueden asignar una o varias funciones en WinCC. Las funciones se lanzan en cuanto el operador pulsa la tecla en el panel de operador.

Una tecla de función se asigna de forma global o local:

- Las teclas de función con asignación global disparan siempre la misma acción, independientemente de la imagen visualizada en ese momento.
- Las teclas de función con asignación local disparan acciones distintas en función de la imagen visualizada en el panel de operador. La asignación vale sólo para la imagen en la que se ha definido la tecla de función.

Llamada de las imágenes

Todas las imágenes configuradas se deben integrar en la secuencia de mando para que el operador pueda acceder en runtime a una imagen en el panel de operador. Para ello se dispone de varias posibilidades:

- Utilizar el editor "Imágenes" para configurar en las imágenes los botones y las teclas de función para acceder a otras imágenes.
- Utilizar el editor "Imagen general" para configurar teclas de función con asignación global.

10.1.1.2 Volumen de funciones de las imágenes en función del dispositivo

Introducción

Las funciones de un panel de operador determinan la representación del dispositivo en WinCC y la funcionalidad de los editores.

Las siguientes propiedades de imágenes dependen del panel de operador configurado:

- Formato
- Resolución
- Cantidad de colores
- Fuentes
- Objetos utilizables

Formato

El formato de una imagen lo determina el panel de operador para el que se realiza la configuración. Si el panel de operador tiene p. ej. teclas de función, éstas aparecen en el diseño de la imagen.

Resolución

Puesto que los paneles de operador tienen tamaños de pantalla distintos, la resolución de la imagen depende del panel de operador configurado. Sólo se puede cambiar la resolución cuando configure el panel de operador "WinCC Runtime Advanced" o "WinCC Runtime Professional".

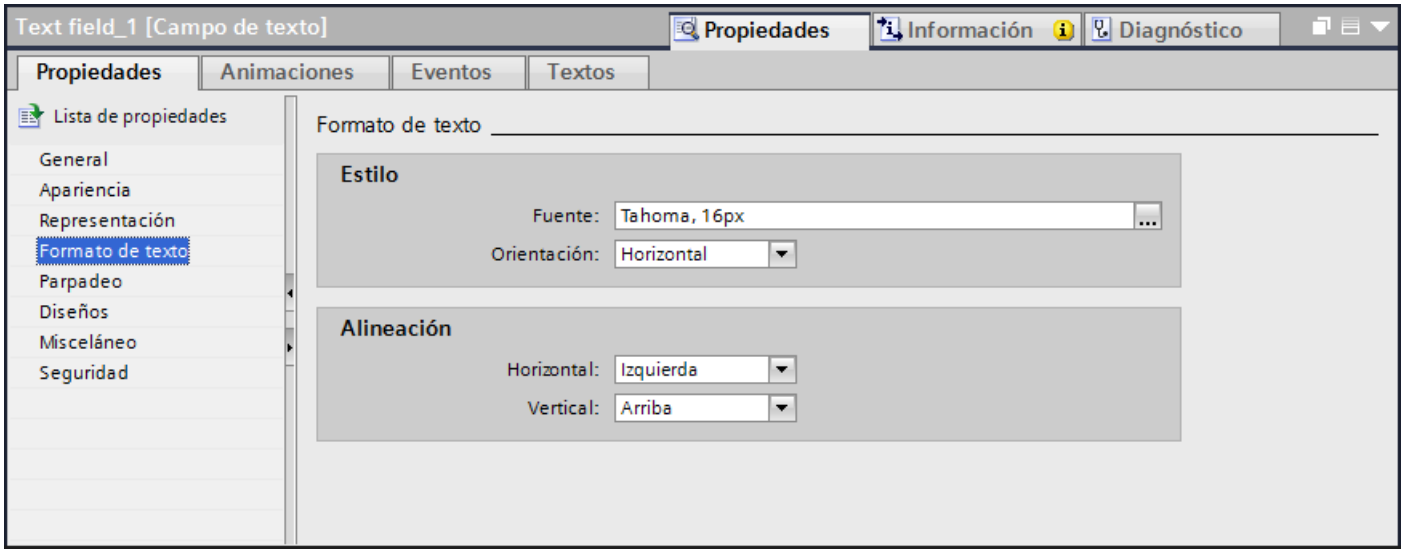
Cantidad de colores

Es posible asignar colores a los objetos de una imagen. El número de colores posibles depende de la cantidad y el tipo de colores que admita el panel de operador configurado.

Fuentes

En todos los objetos de imagen que contengan texto estático o dinámico puede adaptarse la apariencia de los textos. Así se resaltan textos individuales en una imagen. Puede seleccionar

p. ej. el tipo, estilo y tamaño de la fuente, así como configurar efectos adicionales, p. ej. el subrayado.



Los ajustes para marcar textos, como el estilo y los efectos, afecta siempre a todo el texto de un objeto de imagen. Así, por ejemplo, es posible visualizar un título completo en negrita, pero no letras o palabras individuales de dicho título.

Objetos utilizables

Algunos objetos gráficos no pueden configurarse para todos los paneles de operador. Dichos objetos no se muestran en la Task Card "Herramientas". Así, p. ej., no es posible configurar un deslizador para el panel de operador KTP1000.

10.1.1.3 Principios básicos

Task Cards

Introducción

El editor "Imágenes" ofrece las siguientes Task Cards:

- Herramientas: objetos de manejo y visualización
- Animaciones: plantillas para la configuración dinámica
- Diseño: recursos para adaptar la representación
- Librerías: administración de la librería del proyecto y las librerías globales

Nota
WinCC Basic

La Task Card "Animaciones" no está disponible en WinCC Basic.

Herramientas

La Task Card "Herramientas" contiene objetos de diferentes paletas:

- Objetos básicos
- Elementos
- Controles
- Controles propios (opcional)
- Gráficos

Los objetos se añaden desde las paletas mediante Drag&Drop o haciendo doble clic en sus imágenes. La selección de objetos depende del panel de operador que se esté configurando. Los símbolos siguientes permiten cambiar el modo de visualización:

Símbolo	Significado
	Representa los objetos en forma de lista.
	Representa los objetos en forma de gráfico.

Animaciones

La Task Card "Animaciones" contiene las posibles dinamizaciones de un objeto de imagen en las paletas. Desde las paletas "Movimientos", "Visualización" y "Enlace de variables", mediante Drag&Drop o haciendo doble clic se añade la animación a un objeto de imagen.

Diseño

La Task Card "Diseño" contiene las paletas siguientes para visualizar objetos y elementos:

- Niveles: sirve para administrar los niveles de los objetos de imagen. Los niveles se muestran en una vista de árbol e incluyen información sobre el nivel activo y la visibilidad de todos los niveles.
- Retícula: indica si quiere alinear los objetos junto a una cuadrícula o junto a otros objetos y ajusta el tamaño de cuadrícula para una cuadrícula.
- Objetos fuera del rango: se muestran los objetos con nombre, posición y tipo que están fuera del área visible

Librerías

La Task Card "Librerías" muestra las librerías siguientes en paletas separadas:

- Librería del proyecto: la librería del proyecto se guarda junto con el proyecto.
- Librería global: la librería global se guarda en un archivo aparte de la ruta indicada en el equipo de configuración.

Desplazar la vista

Introducción

Para representar en el área de trabajo únicamente una sección de la imagen total, utilice el símbolo del editor "Imágenes".

Requisitos

- Hay una imagen abierta.
- En el área de trabajo se visualiza sólo una sección de la imagen.

Procedimiento

Para desplazar una vista:

1. Haga clic en el símbolo , situado en la esquina inferior derecha del área de trabajo, y pulse el botón izquierdo del ratón.
   Aparecerá una vista en miniatura de toda la imagen. Un marco de color naranja señala la sección actual.
2. Mantenga pulsado el botón del ratón y desplace el marco a la sección deseada.

Nota

Al desplazar un objeto de imagen desde la sección visible hasta una sección oculta en la vista actual, se desplazará también la imagen.

Ampliar la vista

Introducción

Para ver una sección de una imagen de forma más detallada, es posible ampliar la vista de la imagen en el área de trabajo. El zoom máximo es del 800 %.

Puede ampliar el zoom mediante la barra de herramientas del área de trabajo o mediante la Task Card "Diseño > Zoom".

Hay diferentes métodos para ampliar la imagen, p. ej., mediante el factor de ampliación o adaptando el área de trabajo a la altura de la imagen.

Requisitos

La imagen está abierta.

Procedimiento

Para ampliar una vista con el marco de selección, haga lo siguiente:

1. Haga clic en el botón de la barra de herramientas.
2. Utilizando el ratón, trace un marco de selección en la imagen.

Tras soltar el botón del ratón, la sección se ampliará hasta ocupar todo el área de trabajo que se encuentra dentro del marco de selección.

Alternativamente, utilice el deslizador de la esquina inferior derecha de la imagen.

Resultado

La vista seleccionada de la imagen se mostrará ampliada.

10.1.1.4 Trabajar con imágenes

Pasos a seguir

Pasos a seguir

Para crear imágenes es necesario seguir los pasos siguientes:

- Planifique la estructura de la representación del proceso: cantidad de imágenes y su orden. Ejemplo: Los procesos parciales se representan en imágenes individuales y se agrupan en una imagen principal.
- Planifique el tipo de navegación dentro de una imagen y entre las distintas imágenes.
- Adapte las plantillas y la imagen general.
  Se definen objetos de forma centralizada y asignan p. ej., teclas de función.
- Cree las imágenes. Para trabajar de forma más eficiente, aproveche las posibilidades que se indican a continuación:
  - Trabajar con librerías
  - Trabajar con niveles
  - Trabajar con faceplates

Crear una imagen nueva

Introducción

Para representar procesos en su instalación, puede crear imágenes.

Requisitos

- Se ha creado un proyecto.
- La ventana de inspección está abierta.

Procedimiento

1. Haga doble clic en "Imágenes > Agregar imagen" en el árbol del proyecto.
   La imagen se crea en el proyecto y se muestra en el área de trabajo. Las propiedades de la imagen se visualizarán en la ventana de inspección.
2. Introduzca un nombre que describa la imagen.
3. Configure las propiedades de la imagen en la ventana de inspección:
   - Determine si la imagen debe basarse en la plantilla y en cuál.
   - Defina el "Color de fondo" y el "Número de imagen".
   - Introduzca un texto explicativo en "Tooltip".
   - En "Niveles", defina qué niveles deben mostrarse en el sistema de ingeniería.
   - Dinamice la visualización de la imagen en "Animaciones".
   - Defina bajo "Eventos" qué funciones deben ejecutarse al abrir y cerrar una imagen o con otros eventos en runtime.

   Nota

   No todos los paneles de operador soportan la animación "Visibilidad".

Resultado

La imagen ha sido creada en el proyecto. En pasos de trabajo posteriores puede insertar objetos y elementos de manejo de la Task Card "Herramientas" y asignar teclas de función.

Administrar imágenes

Introducción

En el árbol del proyecto se pueden desplazar imágenes a otros grupos del proyecto o bien copiarlas, cambiar su nombre o borrarlas.

Desplazar una imagen a un grupo

1. Seleccione la carpeta "Imágenes" en el árbol del proyecto.
2. En el menú contextual, elija el comando "Agregar grupo".
   Se insertará una carpeta con el nombre "Grupo_x".
3. Seleccione la imagen en el árbol del proyecto.
4. Arrastre la imagen con la función de arrastrar y soltar al grupo deseado.
   La imagen se desplaza a dicho grupo.

Copiar una imagen

1. Seleccione la imagen en el árbol del proyecto.
2. Elija el comando "Copiar" del menú contextual para copiar la imagen en el portapapeles.

3. Marque en el árbol del proyecto la posición en la que desea insertar la imagen.
4. En el menú contextual seleccione el comando "Pegar" para insertar la imagen.
   Se insertará una copia de la imagen. El nombre del original se complementará en la copia con un número correlativo.

También puede mantener pulsada la tecla <Ctrl> y arrastrar la imagen hasta la posición deseada.

Nota

Si se copia en todo el panel de operador o en todo el proyecto una imagen con una plantilla enlazada, también se copiará la plantilla. Si ya existe una plantilla adecuada, no se utilizará. Esto sucede especialmente cuando se copian imágenes por Drag&Drop.

Cambiar el nombre de la imagen

1. Seleccione la imagen en el árbol del proyecto.
2. En el menú contextual, seleccione el comando "Cambiar nombre".
3. Introduzca un nuevo nombre.
4. Pulse la tecla <Intro>.

También puede utilizar la tecla <F2> para cambiar el nombre de la imagen.

Borrar la imagen

1. Seleccione la imagen en el árbol del proyecto.
2. En el menú contextual, elija el comando "Borrar".
   Se borra la imagen del proyecto con todos los objetos que contiene.

Definir la imagen inicial del proyecto

Introducción

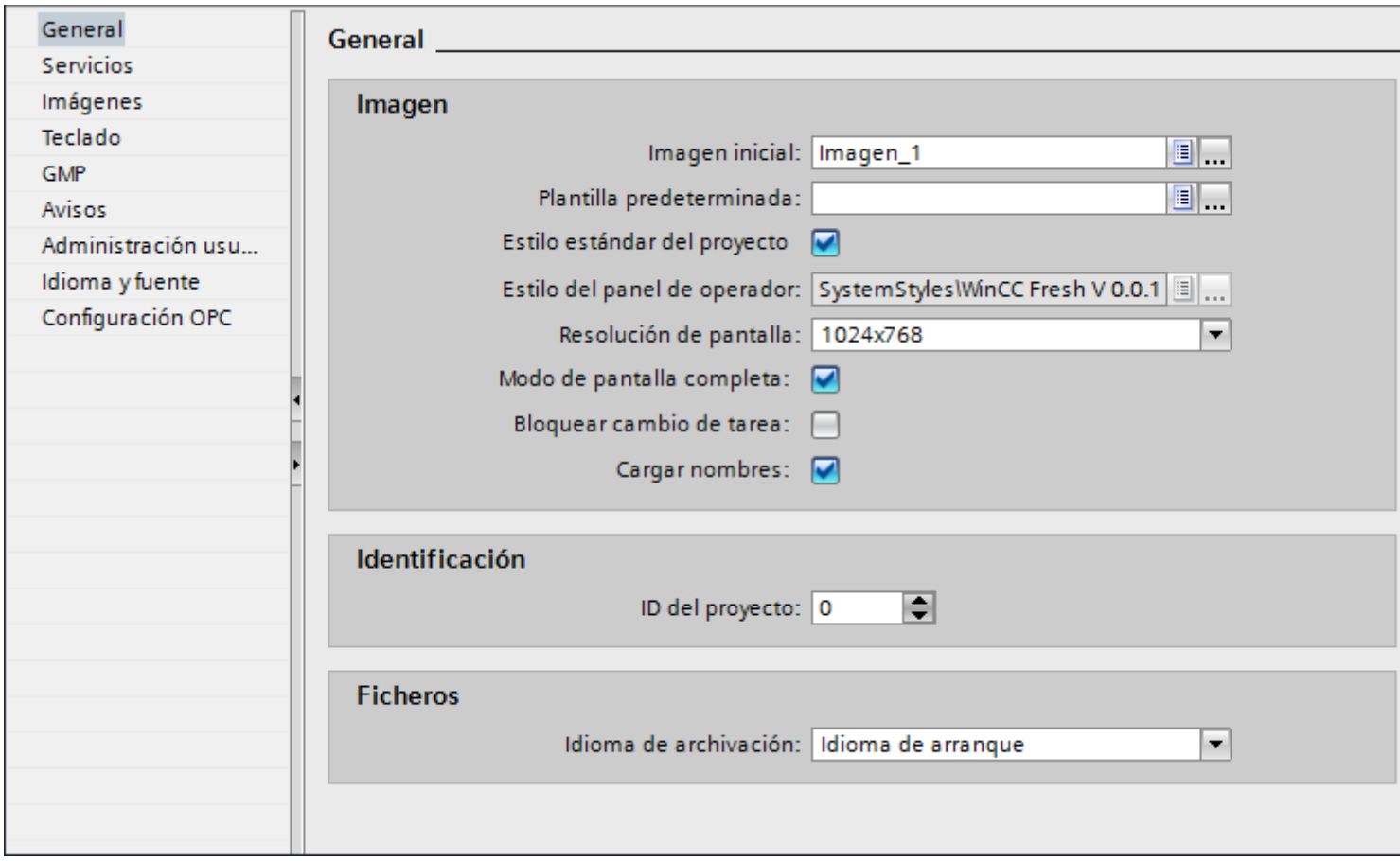
La imagen inicial es la primera imagen que aparece al iniciar el proyecto en runtime. Para cada panel de operador se puede definir una imagen inicial propia. El operador accede a las restantes imágenes desde la imagen inicial.

Requisitos

La imagen a definir como imagen inicial está creada en el proyecto.

Procedimiento

1. Haga doble clic en "Configuración de runtime > General" en el árbol del proyecto.



2. Elija la "Imagen inicial" deseada.

   Otra posibilidad es seleccionar una imagen en el árbol del proyecto y seleccionar en el menú contextual "Utilizar como imagen inicial".

Resultado

Cuando se inicie runtime, la imagen inicial se visualizará en el panel de operador.

10.1.1.5 Trabajar con plantillas

Principios básicos para trabajar con plantillas

Introducción

En una plantilla se configuran objetos que se representan en todas las imágenes que se basan en dicha plantilla.

Rigen las reglas siguientes:

- una imagen no debe basarse en una plantilla.
- una imagen no consiste en una plantilla.
- para un dispositivo se pueden crear varias plantillas.
- una plantilla no se puede basar en otra.

Objetos para una plantilla

En la plantilla se definen las funciones y objetos que se aplican a todas las imágenes que se basan en dicha plantilla:

- Asignación de teclas de función: En los paneles de operador con teclas de función, éstas se asignan también en la plantilla. Esta asignación sobrescribe una posible asignación global.
- Ventana permanente: algunos dispositivos muestran en el área superior de la imagen una ventana permanente para todas las imágenes. A diferencia de la plantilla, la ventana permanente ocupa un área de la imagen sólo para ella.
- Objetos de manejo: se pueden insertar todos los objetos de imagen en una plantilla que también utilice para una imagen.

Ejemplos de aplicación

- En la plantilla se puede asociar p. ej. una tecla de función a la función "ActivarImagen". Cuando el operador pulse esta tecla en runtime, pasará a otra imagen. Este ajuste es aplicable a todas las imágenes basadas en esa plantilla.
- Inserte en la plantilla un gráfico con el logo de la empresa. El logo de la empresa se representa en todas las imágenes basadas en la plantilla.

Nota

Si un objeto de la plantilla se encuentra en la misma posición que un objeto de la imagen, el objeto de la plantilla se ocultará.

Consulte también

Crear una plantilla nueva (Página 8321)

Administrar plantillas (Página 8322)

Imagen general (Página 8320)

Utilizar una plantilla en la imagen (Página 8323)

Imagen general

Introducción

Para todas las imágenes de un panel de operador se definen elementos globales, independientemente de la plantilla utilizada.

Teclas de función

En el caso de paneles de operador con teclas de función, las teclas de función se asignan de manera global en el editor "Imagen general". Esta asignación global sirve para todas las imágenes del panel de operador.

Para asignar teclas de función de manera local en imágenes o plantillas, haga lo siguiente:

1. Haga clic en sus imágenes o plantillas en la tecla de función.
2. Desactive en la ventana de inspección "Propiedades > Propiedades > General > Utilizar asignación general".

Objetos de manejo y visualización para avisos

Los objetos generales disponibles "Ventana de avisos" e "Indicador de avisos" se configuran dentro del editor "Imagen general".

Los objetos "Ventana de avisos" e "Indicador de avisos" se muestran siempre en primer plano.

Para Comfort Panels también se puede configurar una "Visualización de diagnóstico del sistema" en la imagen general.

Nota

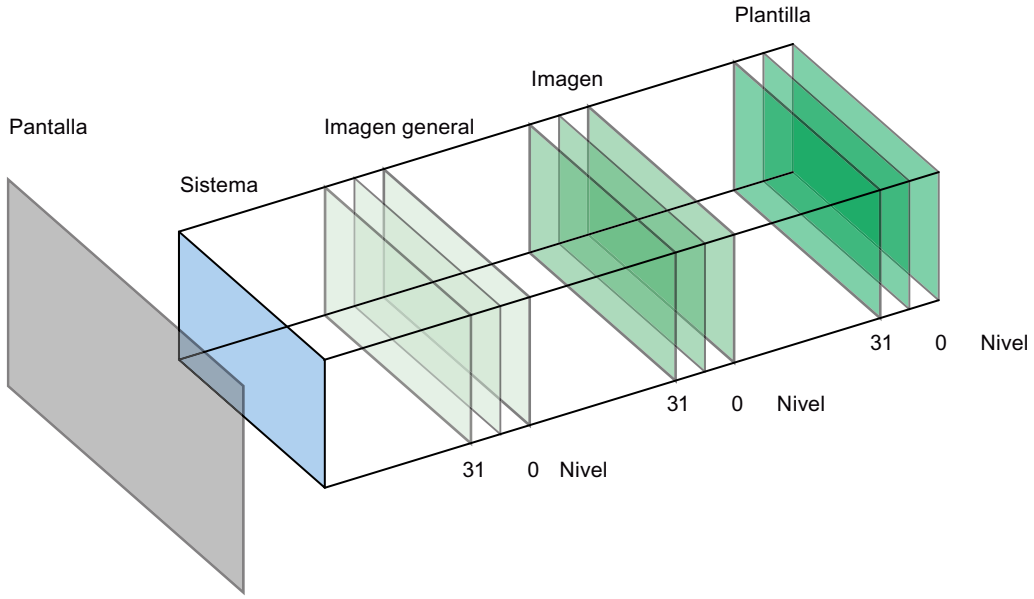
Si ha configurado una plantilla en una ventana permanente, no coloque la ventana de avisos y el indicador de avisos en el área de la ventana permanente. En caso contrario, la ventana de avisos y el indicador de avisos no se verán en runtime.

Sin embargo, la ventana permanente no se ve en el editor "Imagen general".

Orden de la configuración de imágenes

En la configuración se sigue el siguiente orden:

- la imagen global precede a imágenes y plantillas
- las imágenes preceden a las plantillas



El nivel de sistema no es configurable. En él hay

- diálogos de entrada
- avisos del sistema operativo
- en los paneles táctiles las teclas directas

Consulte también

Principios básicos para trabajar con plantillas (Página 8316)

Crear una plantilla nueva

Introducción

En una plantilla, los objetos y las teclas de función se editan de forma centralizada. En caso de modificar un objeto o la asignación de una tecla de función en la plantilla, el objeto o tecla también se modificará en todas las imágenes que se basen en la plantilla.

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

Requisitos

- Se ha creado un proyecto.
- La ventana de inspección está abierta.

Procedimiento

1. En el árbol del proyecto, haga doble clic en "Administración de imágenes > Plantillas > Agregar plantilla".
   La plantilla se crea en el proyecto y se muestra en el área de trabajo.
   Las propiedades de la plantilla se visualizan en la ventana de inspección.
2. Defina el nombre de la plantilla en la ventana de inspección "Propiedades > Propiedades > General".
3. En la ventana de inspección "Propiedades > Propiedades > Niveles", defina qué niveles se mostrarán en el sistema de ingeniería.
4. Inserte los objetos necesarios desde la Task Card "Herramientas".
5. Configure las teclas de función.

Resultado

La plantilla ha sido creada en el proyecto.

Consulte también

Principios básicos para trabajar con plantillas (Página 8316)

Administrar plantillas

Introducción

En la ventana de proyecto se desplazan, copian, cambian de nombre o borran plantillas dentro de un proyecto.

Desplazar una plantilla a un grupo

1. Seleccione "Administración de imágenes > Plantillas" en el árbol del proyecto.
2. En el menú contextual, elija "Agregar grupo".
   Se insertará una carpeta con el nombre "Grupo_x".
3. Seleccione la plantilla en el árbol del proyecto.
4. Arrastre la plantilla con la función de arrastrar y soltar al grupo deseado.
   La plantilla se desplaza a este grupo.

Copiar una plantilla

1. Seleccione la plantilla en el árbol del proyecto.
2. En el menú contextual, elija "Copiar".
3. Marque en el árbol del proyecto la posición en la que desea insertar la plantilla.
4. Para insertar la plantilla elija "Pegar" en el menú contextual.
   La copia recibirá automáticamente un nombre unívoco.

Otra posibilidad es arrastrar la plantilla a la posición deseada con la tecla <Ctrl> pulsada.

Borrar una plantilla

1. Seleccione la plantilla a borrar en el árbol del proyecto.
2. En el menú contextual, elija "Borrar".
   Se borra la plantilla del proyecto con todos los objetos que contiene.

Asignar una plantilla a una imagen

1. Marque en el árbol del proyecto la imagen a la que desea asignar la plantilla.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > General".
3. Elija la plantilla deseada en "Plantilla".
   La plantilla seleccionada se asignará a la imagen junto con todos sus objetos.

Consulte también

Principios básicos para trabajar con plantillas (Página 8316)

Utilizar una plantilla en la imagen

Introducción

Utilizar una plantilla en la imagen. Todo los objetos configurados en la plantilla también están disponibles en la imagen.

Requisitos

Se ha creado una plantilla.

Se ha creado una imagen.

Procedimiento

Para utilizar una plantilla en una imagen, haga lo siguiente:

1. Haga doble clic en el árbol del proyecto en una imagen. La imagen se abre en el área de trabajo.
2. Abra en la ventana de inspección "Propiedades > Propiedades > General".
3. Elija en "Plantilla" una plantilla que se utilice en la imagen.

Mostrar la plantilla en la imagen

Cuando edite una imagen, puede mostrar una plantilla existente en la imagen.

Para mostrar una plantilla en la imagen, haga lo siguiente:

1. Active en el menú "Herramientas > Configuración > Visualización > Mostrar plantillas en imágenes".

Resultado

La imagen se basa en la plantilla seleccionada. Todos los objetos que se han configurado en la plantilla existen en la imagen. La plantilla se muestra en la imagen.

Consulte también

Principios básicos para trabajar con plantillas (Página 8316)

10.1.1.6 Trabajar con estilos

Principios básicos para trabajar con estilos

Introducción

El editor de estilos es un editor global. En el editor de estilos puede definir una apariencia uniforme para los elementos de manejo y visualización. De esta manera se armoniza la representación de los objetos en runtime. El editor de estilos ofrece estilos predefinidos. Puede elegir un estilo predefinido o crear un estilo propio. El editor de estilos se utiliza independientemente de los dispositivos y proyectos.

Vistas en el editor de estilos

En el editor de estilos se agrupan los objetos según sus características visuales más importantes.

Existen los siguientes grupos:

- objetos basados en botones, líneas y polígonos,
- objetos basados en textos,

- objetos basados en diagramas, como p. ej. visor de curvas,
- objetos basados en gráficos o valores, como p. ej. tabla de valores.

Visualización de estilos predefinidos

Introducción

En el editor "Estilos" puede ver los estilos predefinidos y crear estilos propios para la representación de los objetos de manejo y visualización en runtime.

Apertura del editor "Estilos"

1. Haga doble clic en "Datos comunes" en el árbol del proyecto.
2. Haga clic en el editor "Estilos". Se abre el editor.

Nota

Editor de estilos en el árbol del proyecto

El editor de estilos solo está visible en el árbol del proyecto si en el proyecto se ha creado un dispositivo que admita estilos, p. ej., Comfort Panels.

3. En el área de trabajo del editor "Estilos" se muestran los estilos estándar:

- WinCC Light
- WinCC Dark
- WinCC Fresh
- WinCC Wireframe.

Resultado

En el área de trabajo se muestra la configuración para los grupos y los distintos objetos.

En los siguientes pasos podrá asignar estilos predefinidos o definir estilos propios.

Los estilos predefinidos están protegidos contra escritura y no se pueden modificar.

Administración de estilos

Borrado de un estilo

Puede borrar los estilos que haya definido.

1. Abra la carpeta "Datos comunes" en el árbol del proyecto.
2. Abra el editor "Estilo".

3. En el área de trabajo, seleccione el estilo que desea borrar.

Nota

Los estilos predefinidos de WinCC no se pueden borrar.

4. En el menú contextual, elija la opción "Borrar".

El estilo se borra.

Editar un estilo

Puede editar los estilos que haya definido usted mismo.

1. Abra la librería del proyecto.
2. Seleccione la versión habilitada del estilo que desee modificar.
3. Seleccione "Editar tipo" del menú contextual
   Se crea una versión nueva "en proceso".
4. Realice las adaptaciones necesarias en la ventana de inspección.
5. Para aplicar los cambios, elija "Habilitar versión" en el menú contextual de la versión editada.
   Se aplican los cambios efectuados.

Definición del estilo estándar para un proyecto

Para conseguir la homogeneidad visual de los objetos indicadores dentro de un proyecto, se puede asignar un estilo estándar a un proyecto.

1. Abra la carpeta "Datos comunes" en el árbol del proyecto.
2. Abra el editor "Estilo".
3. Active un estilo en el área de trabajo.

El estilo activado se utilizará como estilo estándar en el proyecto. Si agrega un dispositivo al proyecto, el dispositivo utilizará el estilo estándar.

Modificación del estilo para un dispositivo predeterminado

Dentro de un proyecto, el estilo se puede modificar para un dispositivo determinado.

1. Abra el editor "Configuración de runtime" del panel de operador.
2. Seleccione en "General" un estilo de dispositivo.

El estilo seleccionado se utilizará en el dispositivo.

Si el estilo del dispositivo no está asignado, este utilizará el estilo estándar del proyecto.

10.1.2 Trabajar con objetos

10.1.2.1 Sinopsis de los objetos

Introducción

Los objetos son elementos gráficos que permiten diseñar las imágenes del proyecto.

La Task Card "Herramientas" contiene todos los objetos disponibles para el panel de operador. La Task Card se visualiza con el comando de menú "Ver" activando la opción "Task Card".

En función del editor abierto actualmente, la ventana de herramientas contendrá diversas paletas. Si está abierto el editor "Imágenes", la ventana de herramientas contendrá las siguientes paletas:

- "Objetos básicos"
  A los objetos básicos pertenecen los objetos gráficos básicos, p. ej., "Línea", "Círculo", "Campo de texto" o "Visor de gráficos".
- "Elementos"
  A los elementos pertenecen los elementos de manejo básico, p. ej., "Campo E/S", "Botón" o "Indicador gauge".
- "Controles"
  Los controles tienen una funcionalidad ampliada. Representan también procesos, como el visor de curvas y el visor de recetas.

- "Gráficos"
  Los gráficos están divididos temáticamente en forma de árbol de directorios. Las diferentes carpetas contienen las siguientes representaciones gráficas, por ejemplo:
  - Áreas de máquinas e instalaciones
  - Instrumentos de medición
  - Elementos de manejo
  - Banderas
  - Edificios

  El usuario crea accesos directos a sus propias carpetas de gráficos. Los gráficos externos están en dichas carpetas y sus subcarpetas. Se muestran en la ventana de herramientas y se integran en el proyecto con el acceso directo.
- Task Card "Librerías"
  Además de los objetos de manejo y visualización existen también los objetos de librerías. Están dentro de las paletas de la Task Card "Librerías". Una librería contiene objetos configurados, tales como gráficos de tuberías, bombas, o bien botones preconfigurados. Los objetos de librerías se pueden integrar varias veces en el proyecto, sin tener que configurarlos de nuevo.
  WinCC se suministra con librerías, p. ej. "HMI Buttons & Switches".
  También se introducen objetos definidos por el usuario y faceplates en librerías propias. Los faceplates son objetos que se crean a partir de objetos de imagen existentes y para los que se definen propiedades configurables.

Nota

Dependencia del panel de operador

Dependiendo del panel de operador que se esté configurando, determinados objetos de la ventana de herramientas no dispondrán de ciertas funciones o éstas estarán limitadas. Las propiedades no disponibles de un objeto aparecen desactivadas y no se pueden seleccionar.

Objetos básicos

Símbolo	Objeto	Indicaciones
	"Línea"	-
	"Elipse"	-
	"Círculo"	-
	"Rectángulo"	-
A	"Campo de texto"	Muestra un texto de una línea o varias. La fuente y la representación pueden configurarse.
	"Visor de gráficos"	Muestra gráficos de programas gráficos externos e inserta objetos OLE. Pueden emplearse los formatos gráficos siguientes: "*.emf", "*.wmf", "*.dib", "*.bmp", "*.jpg", "*.jpeg", "*.gif" y "*.tif".

Elementos

Símbolo	Objeto	Indicaciones
	"Campo E/S"	Devuelve los valores de una variable y/o escribe valores en una variable. Es posible determinar límites para los valores de variables representados en el campo E/S. Para una entrada invisible del operador en runtime, configure "Entrada oculta".
	"Botón"	Ejecuta una lista de funciones o un script, en función de la configuración.
	"Campo E/S simbólico"	Devuelve los valores de una variable y/o escribe valores en una variable. En función del valor de la variable se mostrará un texto de una lista de textos.
	"Campo E/S gráfico"	Devuelve los valores de una variable y/o escribe valores en una variable. En función del valor de la variable se mostrará un gráfico de una lista de gráficos.
	"Campo de fecha y hora"	Devuelve la fecha y hora de la hora del sistema o de una variable. Permite al operador introducir valores nuevos. El formato de visualización puede configurarse.
	"Barra"	Representa un valor del controlador en forma de columna provista de una escala.
	"Interruptor"	Cambia entre dos estados definidos. Un interruptor puede rotularse con texto o con un gráfico.

Controles

Símbolo	Objeto	Descripción
	"Visor de avisos"	Muestra avisos o eventos de aviso pendientes del búfer de avisos o del fichero de avisos.
	"Visor de curvas"	Representa varias curvas con la evolución de los valores del controlador o de un fichero.
	"Visor de usuarios"	Permite a un administrador gestionar los usuarios del panel de operador. Permite a un operador sin derechos de administrador cambiar su contraseña.
	"Visor de recetas"	Muestra registros y permite editarlos.
	"Visor de diagnóstico del sistema"	Ofrece una vista general de todos los dispositivos aptos para diagnóstico. Indica los fallos de la instalación.

Consulte también

Insertar un objeto (Página 8331)

Posicionar un objeto (Página 8334)

Modificar el tamaño de un objeto (Página 8335)

Seleccionar varios objetos (Página 8337)
Alinear objetos (Página 8339)
Desplazar un objeto hacia adelante o hacia atrás (Página 8341)
Mostrar objetos fuera del área de la imagen (Página 8342)
Girar un objeto (Página 8342)
Voltear un objeto (Página 8344)
Insertar varios objetos del mismo tipo (sellar) (Página 8355)
Modificar la posición y el tamaño de varios objetos (Página 8357)
Gráficos externos (Página 8358)
Administrar gráficos externos (Página 8359)
Agregar un gráfico externo a la colección de gráficos (Página 8360)
Principios básicos sobre grupos (Página 8363)
Vista general del acceso con el teclado (Página 8370)
Ejemplo: Insertar y configurar un rectángulo (Página 8373)
Objetos para Basic Panels (Página 8425)

10.1.2.2 Posibilidades de edición de objetos

Introducción

Los objetos son elementos gráficos que permiten diseñar las imágenes del proyecto.

Existen las siguientes posibilidades para editar objetos:

- Copiar, pegar o borrar objetos con el menú contextual. Si copia un objeto en una imagen y ésta ya contiene un objeto de igual nombre, cambiará el nombre del objeto.
- Insertar objetos en el tamaño estándar, o bien determinar su tamaño al insertarlos
- Desplazar un objeto delante o detrás de otros objetos
- Girar un objeto
- Voltear objetos
- Determinar el orden de tabulación de los objetos
- Insertar consecutivamente: Insertar varios objetos del mismo tipo
- Seleccionar varios objetos simultáneamente

- Modificar la posición y el tamaño de varios objetos
- A los objetos se asignan gráficos externos, p. ej. en el visor de gráficos.
  Se visualizan únicamente los gráficos depositados previamente en el juego de gráficos del proyecto WinCC.
  Los gráficos se depositan en la colección de gráficos como se indica a continuación:
  - Utilizando el método de arrastrar y soltar desde el grupo de objetos "Gráficos" hasta el área de trabajo
  - Como archivo gráfico en uno de los formatos siguientes: *.bmp, *.dib, *.ico, *.emf, *.wmf, *.gif, *.tif, *.jpeg o *.jpg
  - Como objeto OLE
    Es posible crear un nuevo objeto OLE, o bien depositar un archivo de imagen existente como objeto OLE. Para depositar objetos OLE, en el equipo de configuración deberá estar instalado un programa gráfico apto para OLE.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.3 Insertar un objeto

Introducción

En el editor "Imágenes" o "Informes" puede agregar los objetos de la de la Task Card "Herramientas". Utilizando el ratón, arrastre los objetos hasta el área de trabajo. Los objetos se pueden insertar en su tamaño estándar, o bien determinar su tamaño al insertarlos.

Además, puede copiar o mover objetos de un editor a otro mediante el portapapeles, p. ej. para incluir un objeto de imagen en un informe. Para copiar y desplazar también puede utilizar el ratón en lugar del portapapeles:

- Copiar: <Ctrl + Drag&Drop>
- Desplazar: Drag&Drop

Nota
Basic Panels

El editor "Informes" no está disponible para Basic Panels.

Requisitos

La Task Card "Herramientas" está abierta.

Insertar un objeto de tamaño estándar

1. En la Task Card "Herramientas", seleccione el objeto en cuestión, o bien el gráfico deseado en la carpeta de gráficos de WinCC.
   Al desplazar el puntero del ratón por el área de trabajo adopta la forma de una cruz y se visualiza junto con el icono del objeto.
2. Haga clic en la posición del área de trabajo en la que desea insertar el objeto o el gráfico.
   El objeto se inserta en su tamaño estándar en la posición deseada del área de trabajo.

También puede hacer doble clic sobre un objeto en la Task Card "Herramientas".

Copiar un objeto

1. Seleccione el objeto deseado.
2. En el menú contextual, elija "Copiar".
3. Haga clic en el lugar deseado y seleccione en el menú contextual "Pegar".

WinCC pega una copia del objeto en el lugar deseado. Sólo pueden modificarse las propiedades adecuadas en el contexto correspondiente.

Ejemplo: En el editor "Imágenes" se define el modo de entrada y salida para los campos E/S. En el editor "Informes" se define el modo como "Salida".

El original y la copia no están vinculados entre sí, y se configuran independientemente uno del otro.

Insertar líneas

1. Seleccione el objeto deseado en la Task Card "Herramientas".
2. Haga clic en un lugar del área de trabajo. Se inserta una línea en tamaño estándar.

Insertar un polígono o una polilínea

1. Seleccione el objeto deseado "Polilínea" o "Polígono" en la Task Card "Herramientas".
2. Haga clic en un lugar del área de trabajo. El punto inicial del objeto queda definido.
3. Haga clic en otro lugar del área de trabajo. Queda definido un vértice.
4. Para cada uno de los demás vértices haga clic en el lugar correspondiente del área de trabajo.
5. Haga doble clic en un lugar del área de trabajo. Queda definido el último vértice.
   Ahora todos los puntos del polígono o de la polilínea quedan definidos.

Nota

Basic Panels

Los objetos "Polilínea" y "Polígono" no están disponibles para Basic Panels.

Nota

Si desea insertar uno tras otro varios objetos de un mismo tipo, utilice la función "Sello". Así evitará tener que seleccionar el objeto en la Task Card "Herramientas" cada vez que desee insertarlo. Para ello, seleccione el botón de la barra de herramientas de la Task Card "Herramientas".

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.4 Borrar un objeto

Introducción

Los objetos se borran individualmente o bien mediante selección múltiple.

Requisitos

El área de trabajo está abierta con un objeto como mínimo.

Procedimiento

1. Seleccione el objeto que desea borrar.
   Para borrar varios objetos, mantenga pulsada la tecla <Mayús> y seleccione sucesivamente los objetos que desea borrar. También puede crear con el ratón un área en torno a los objetos deseados.
2. En el menú contextual, elija el comando "Borrar".

Resultado

Los objetos seleccionados se borran.

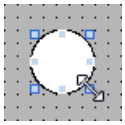
Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.5 Posicionar un objeto

Introducción

Cuando se selecciona un objeto, éste queda rodeado por un rectángulo con controladores de tamaño. Este rectángulo rodea al objeto. La posición de un objeto se define mediante las coordenadas de la esquina superior izquierda del rectángulo que rodea al objeto.



Nota

Si la posición está fuera del área de trabajo, el objeto no se representará en runtime.

Posicionar y alinear

Puede mostrar una cuadrícula en el área de trabajo. Para posicionar objetos más fácilmente, tiene a su disposición tres opciones:

- "Alinear a la cuadrícula": Cuando mueva objetos, éstos se alinearán a la cuadrícula e insertarán automáticamente. Si, al mismo tiempo, mantiene pulsada la tecla <Alt>, el objeto no se alineará a la cuadrícula.
- "Alinear a otros objetos": Cuando mueva objetos, se mostrarán líneas de ayuda. Al posicionar, puede orientarse con otros objetos.
- "Ninguno": se puede posicionar los objetos en cualquier posición.

La cuadrícula y las opciones se activan y desactivan tal y como sigue:

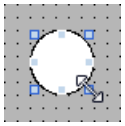
- En el menú "Herramientas > Configuración > Visualización > Imágenes"
- En la Task Card "Diseño > Cuadrícula"

Requisitos

El área de trabajo está abierta con un objeto como mínimo.

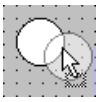
Procedimiento

1. Seleccione el objeto que desea desplazar.
   El objeto seleccionado quedará rodeado por un rectángulo con controladores de tamaño.

2. Haga clic en el objeto y mantenga presionado el botón izquierdo del ratón.

3. Desplace el puntero del ratón hasta la nueva posición.
   El contorno del objeto se desplazará con el ratón y mostrará la nueva posición del objeto.



   Inicialmente, el objeto permanecerá en su posición original.
4. Suelte el botón del ratón.
   El objeto se desplazará hasta la posición donde se había visualizado el contorno del objeto.

Procedimiento alternativo

1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".
2. En "Posición y tamaño" indique los valores X e Y de la posición.

Resultado

El objeto aparecerá en la nueva posición.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.6 Modificar el tamaño de un objeto

Introducción

Cuando se selecciona un objeto, éste queda rodeado por un rectángulo con controladores de tamaño. Existen las siguientes posibilidades para modificar el tamaño de un objeto:

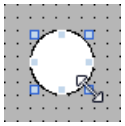
- Arrastrar los controladores de tamaño utilizando el ratón
- Modificar la propiedad "Tamaño" en la ventana de inspección

Requisitos

El área de trabajo está abierta con un objeto como mínimo.

Procedimiento

1. Seleccione el objeto cuyo tamaño desea modificar.
   Aparecerá el rectángulo que rodea al objeto. La figura siguiente muestra un objeto seleccionado.



2. Arrastre el controlador de tamaño del rectángulo hasta una nueva posición.
   Se modificará el tamaño del objeto.
   – Si se ha activado la función "Alinear a la cuadrícula", el tamaño del objeto se modificará según los puntos de la cuadrícula.
   – Si mantiene pulsada la tecla <Alt> mientras arrastra el controlador, esta función no se aplicará.
     Para escalar el objeto de forma proporcional, mantenga pulsada la tecla <Mayús> mientras modifica el tamaño con el ratón.

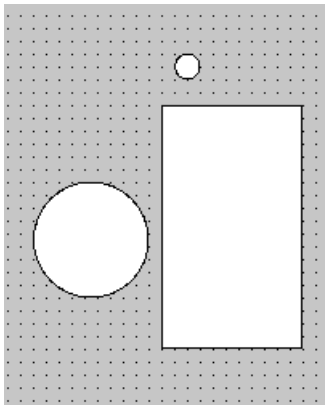
Procedimiento alternativo

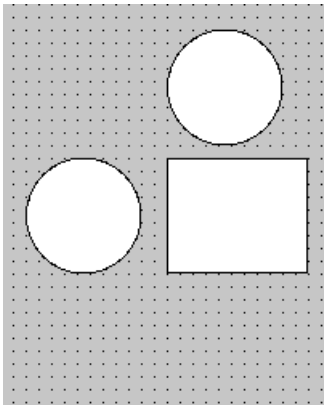
1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Indique el tamaño del objeto en "Posición y tamaño".

Unificar el tamaño de objetos

1. Seleccione los objetos.
2. Haga clic en uno de los botones siguientes: o o
   El tamaño de los objetos seleccionados se adaptará mutuamente.

La figura siguiente muestra cómo adaptar los objetos seleccionados a la altura del objeto de referencia:





Símbolo	Descripción
	Adapta los objetos seleccionados al ancho del objeto de referencia.
	Adapta los objetos seleccionados a la altura del objeto de referencia.
	Adapta los objetos seleccionados a la anchura y altura del objeto de referencia.

Resultado

El objeto se representará en su nuevo tamaño.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.7 Seleccionar varios objetos

Introducción

Para modificar las propiedades de varios objetos simultáneamente, o bien para alinear los objetos entre sí, deberá seleccionar todos los objetos en cuestión. Esta operación se denomina "selección múltiple".

En la ventana de inspección aparecen todas las propiedades de los objetos seleccionados.

Existen las siguientes posibilidades para seleccionar varios objetos:

- Trace un marco de selección alrededor de los objetos deseados.
- Haga clic en los objetos deseados manteniendo pulsada la tecla <Mayús>.

Marco para una selección múltiple

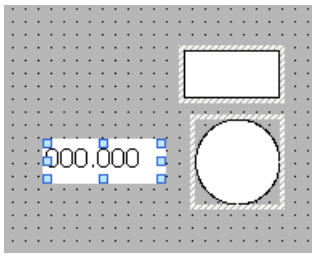
El marco de selección abarca todos los objetos de una selección múltiple. Este marco de selección es comparable al rectángulo que rodea a un objeto individual.

El marco de selección no se visualiza. Una vez concluida la selección múltiple se mostrarán los marcos siguientes:

- El objeto de referencia aparecerá con el rectángulo circundante.
- Los demás objetos seleccionados aparecerán con un marco discontinuo.

Definir un objeto de referencia

El objeto de referencia es el objeto utilizado para alinear los demás objetos. El objeto de referencia está rodeado por un rectángulo con controladores de tamaño. La figura siguiente muestra un objeto de referencia con dos objetos más seleccionados:



Para definir el objeto de referencia existen las posibilidades siguientes:

- Seleccione los objetos mediante una selección múltiple. El objeto seleccionado en primer lugar es el objeto de referencia.
- Trace un marco de selección alrededor de los objetos deseados. El objeto de referencia se determinará automáticamente. Si desea determinar como objeto de referencia otro objeto de la selección, haga clic en el objeto deseado. Esta acción no cancela la selección múltiple.

Requisitos

El área de trabajo está abierta con dos objetos como mínimo.

Seleccionar varios objetos con un marco de selección

1. Sitúe el puntero del ratón en el área de trabajo cerca de uno de los objetos que desea seleccionar.
2. Manteniendo pulsado el botón del ratón, trace un marco de selección alrededor de los objetos que desea seleccionar.

O:

1. Mantenga pulsada la tecla <Mayús>.
2. Haga clic con el ratón consecutivamente en los objetos deseados.
   Todos los objetos seleccionados se enmarcarán.
   El objeto de referencia será el primer objeto seleccionado.

Nota

Para eliminar un objeto de la selección múltiple, vuelva a hacer clic en el objeto manteniendo pulsada la tecla <Mayús>.

Resultado

Quedarán seleccionados varios objetos. Uno de ellos se identificará como objeto de referencia. A continuación, podrá realizar los pasos siguientes:

- Modificar las propiedades de todos los objetos
- El tamaño de todos los objetos con un mismo factor se modifica arrastrando el cuadro de selección con el ratón haciéndolo más grande o más pequeño.
- Desplazar todos los objetos simultáneamente.
- Alinear los objetos al objeto de referencia.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.8 Alinear objetos

Procedimiento

1. Seleccione los objetos deseados mediante una selección múltiple.
2. Defina un objeto como objeto de referencia.
3. Seleccione el comando deseado en la barra de herramientas o en el menú contextual; véase tabla más abajo.
   Los objetos seleccionados se alinearán.

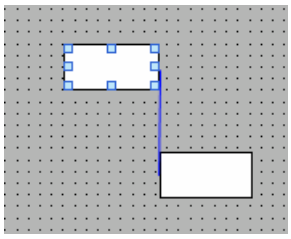
Justificar objetos

Los objetos seleccionados se alinean justificados al objeto de referencia.

Símbolo	Descripción
	Justifica los objetos seleccionados al borde izquierdo del objeto de referencia.
	Justifica los objetos seleccionados al eje central vertical del objeto de referencia.
	Justifica los objetos seleccionados al borde derecho del objeto de referencia.
	Justifica los objetos seleccionados al borde superior del objeto de referencia.
	Justifica los objetos seleccionados al eje central horizontal del objeto de referencia.
	Justifica los objetos seleccionados al borde inferior del objeto de referencia.
	Centra los objetos seleccionados alrededor del punto central del objeto de referencia.
	Centra verticalmente los objetos seleccionados en la imagen.

Alinear a un objeto

Cuando mueva objetos, aparecerán líneas auxiliares. Al posicionar, puede orientarse con otros objetos.



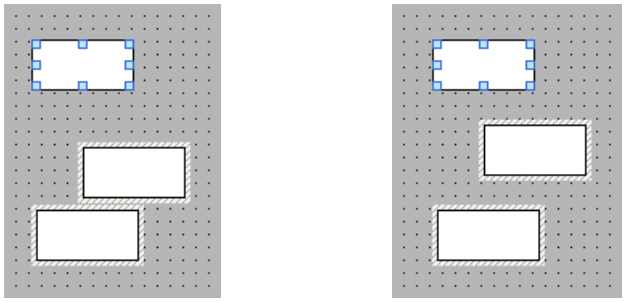
Si trabaja con el teclado, pulse la tecla Alt. Si mueve el objeto seleccionado con las teclas de flecha, se le mostrará el siguiente punto de anclaje.

Distribuir los objetos uniformemente

Es preciso haber seleccionado tres objetos como mínimo. No se necesita un objeto de referencia.

1. Seleccione los objetos.
2. Haga clic en uno de los botones "Espaciar horizontalmente por igual" o "Espaciar verticalmente por igual".
   Los objetos seleccionados se distribuyen de forma equidistante.

La figura siguiente muestra cómo ordenar los objetos de forma equidistante en sentido vertical:



Símbolo	Descripción
	Ordena los objetos de forma equidistante en sentido horizontal. La posición de los objetos extremos (derecho e izquierdo) permanece invariable. Los demás objetos se distribuyen entre ellos de forma equidistante.
	Ordena los objetos de forma equidistante en sentido vertical. La posición del objeto superior y del objeto inferior (derecho e izquierdo) no se modifica. Los demás objetos se distribuyen entre ellos de forma equidistante.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.9 Desplazar un objeto hacia adelante o hacia atrás

Introducción

Con la función "Orden" del menú contextual de un objeto seleccionado o de la barra de herramientas se desplaza el objeto delante o detrás de otros objetos dentro de un nivel.

Nota

Los controles ActiveX se encuentran siempre delante de los demás objetos de un nivel (propiedad .NET).

Requisitos

Está abierta una imagen con varios objetos en un mismo nivel.

Procedimiento

1. Seleccione el objeto que desea desplazar hacia delante o hacia atrás.
2. Elija el comando "Orden" en el menú contextual y uno de los comandos siguientes:

Símbolo	Descripción
	Desplaza el objeto seleccionado delante de todos los demás objetos de ese mismo nivel
	Desplaza el objeto seleccionado detrás de todos los demás objetos de ese mismo nivel
	Desplaza el objeto seleccionado una posición hacia delante
	Desplaza el objeto seleccionado una posición hacia atrás

Procedimiento alternativo

1. Abra la paleta "Niveles" de la Task Card "Diseño".
2. Navegue hasta el objeto deseado.
3. Arrastre el objeto por el árbol hasta la posición deseada dentro del nivel manteniendo pulsado el botón del ratón.
4. Suelte el botón del ratón.

Resultado

El objeto se desplazará hacia adelante o hacia atrás.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.10 Mostrar objetos fuera del área de la imagen

Introducción

Si asigna a objetos posiciones que están fuera del área configurable, dichos objetos quedarán ocultos. Con las funciones de la paleta "Objetos fuera del rango" de la Task Card "Diseño", estos objetos vuelven a desplazarse hasta dentro de la imagen.

Requisitos

- Está abierta una imagen con objetos fuera del área configurable.
- La Task Card "Diseño" está abierta.

Procedimiento

1. Abra la Task Card "Diseño > Objetos fuera del rango".
   Aparecerá la lista de los objetos que están fuera del área configurable.
2. Seleccione en la lista el objeto que quiera volver a arrastrar a la imagen.
3. Seleccione en el menú contextual del objeto "Arrastrar a la imagen".

Otra posibilidad es abrir la Task Card "Diseño > Nivel". Los objetos situados fuera del área se representan con el símbolo . Si hace clic en este símbolo el objeto se desplazará de nuevo hasta la imagen.

Resultado

Los objetos se arrastran al área configurable.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.11 Girar un objeto

Introducción

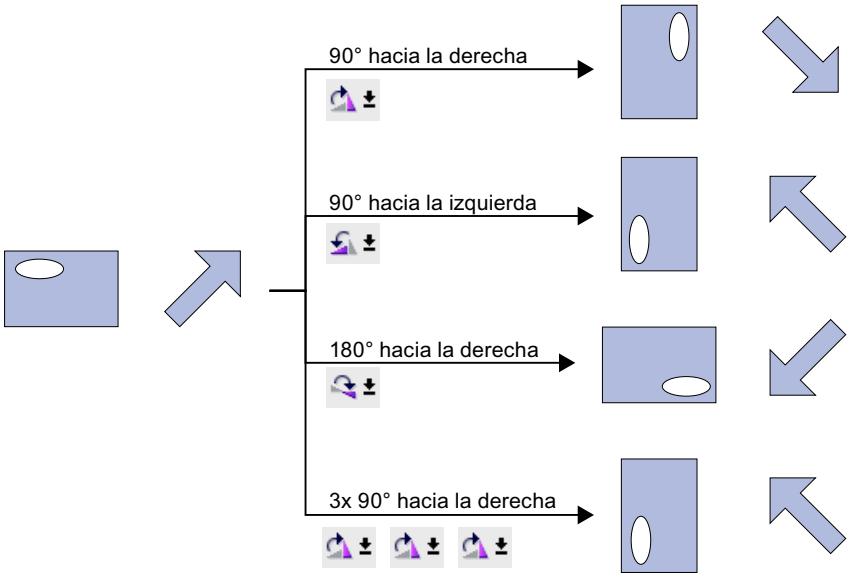
Un objeto adecuado se puede girar alrededor de sí mismo en pasos de 90° hacia la derecha o hacia la izquierda.

Nota

No todos los objetos pueden girarse. Algunos objetos que pueden girarse en las imágenes no pueden girarse en los informes.

Utilizando la selección múltiple también se pueden girar varios objetos simultáneamente. Algunos objetos de WinCC (p. ej. los botones) no se pueden girar.

Al girar un objeto se modifica la orientación de los elementos que contiene. La figura siguiente muestra cómo se comportan un rectángulo y una elipse con los diferentes comandos para girar un objeto:



Requisitos

El área de trabajo está abierta con un objeto como mínimo.

Procedimiento

1. Seleccione el objeto que desea girar.
2. Haga clic en uno de los siguientes botones de la barra de herramientas:
   para girar el objeto hacia la derecha alrededor de su punto central. El ángulo de giro es de 90°.
   para girar el objeto hacia la izquierda alrededor de su punto central. El ángulo de giro es de 90°.
   para girar el objeto hacia la derecha en 180°.

Resultado

El objeto aparecerá girado.

Consulte también

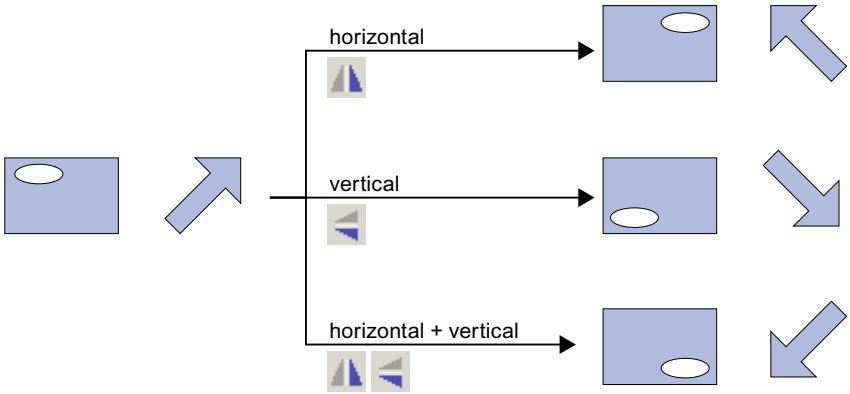
Sinopsis de los objetos (Página 8325)

Elipse (Página 8431)

10.1.2.12 Voltear un objeto

Introducción

Los objetos se pueden voltear tomando como referencia su eje horizontal o vertical. Al voltear un objeto se modifica la orientación de los elementos que contiene. La figura siguiente muestra cómo se comportan un rectángulo y una elipse con los diferentes comandos para voltear un objeto.



Requisitos

Hay como mínimo una imagen abierta con un objeto.

Procedimiento

1. Seleccione el objeto que desea voltear.
2. En el menú contextual, haga clic en el comando "Voltear" y elija una de las posibilidades que se ofrecen:
   – para voltear el objeto seleccionado tomando como referencia su eje vertical.
   – para voltear el objeto seleccionado tomando como referencia su eje horizontal.

Resultado

El objeto aparecerá volteado.

Consulte también

Sinopsis de los objetos (Página 8325)

Elipse (Página 8431)

10.1.2.13 Diseñar un objeto

Introducción

Hay que diseñar el borde y el fondo de un objeto.

Requisitos

Hay una línea creada en una imagen.

Procedimiento

1. Seleccione la línea de la imagen.
2. Haga clic en la ventana de inspección "Propiedades > Propiedades > Apariencia".
3. Seleccione el estilo "Raya".
4. Para representar la línea discontinua en dos colores, seleccione el ancho de línea "1".
5. Elija el ajuste "Flecha" en el área "Finales de línea".

Resultado

La línea se representa discontinua y en dos colores. El final de la línea es una flecha.

10.1.2.14 Diseñar patrón de relleno

Introducción

WinCC ofrece la posibilidad de diseñar el color de fondo y el patrón de relleno de un objeto. En función del objeto en el que se diseñe el patrón de relleno, cambiarán las opciones de diseño en la ventana de inspección.

Para determinados objetos no solo se puede definir un fondo de color, sino también un fondo transparente o con gradiente de color.

Nota

Dependencia de las propiedades según el dispositivo

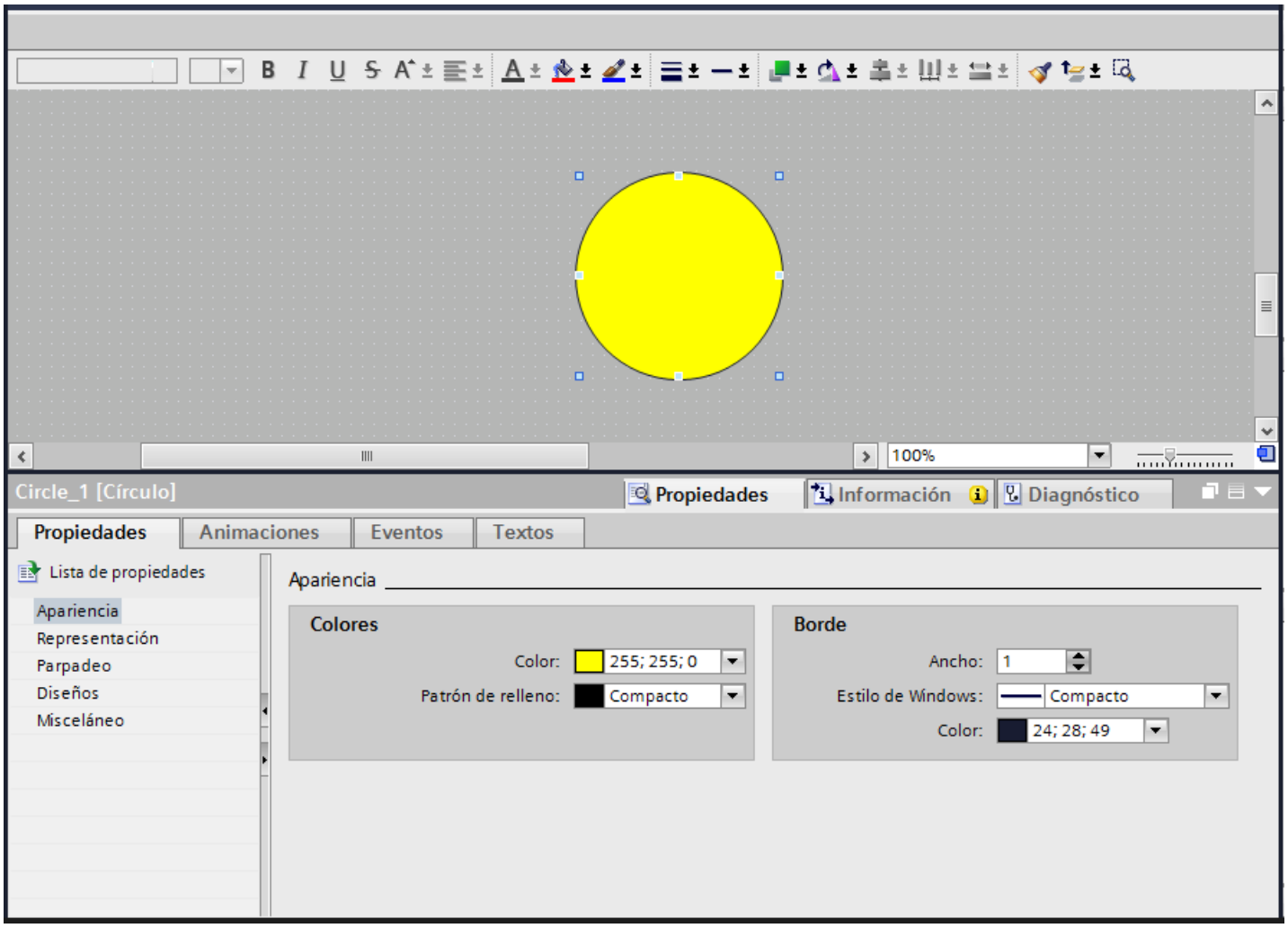
Los estilos de relleno disponibles dependen del objeto y del panel de operador utilizados.

Requisitos

Se ha creado y seleccionado el objeto.

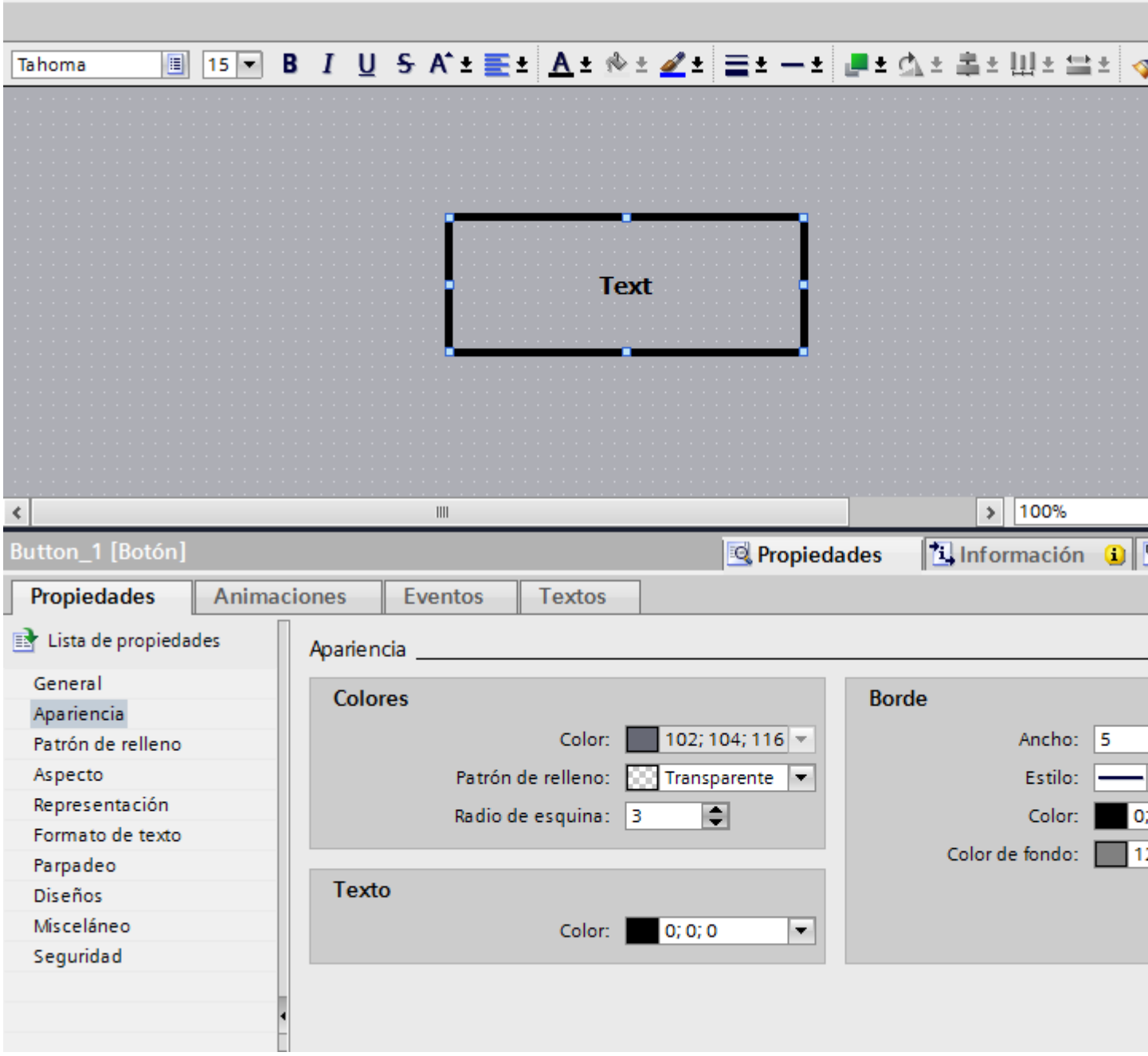
Diseñar el color de fondo de un objeto

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Apariencia".
2. Seleccione el color para el fondo del objeto, p. ej., amarillo.
   El objeto se rellena con el color seleccionado.



Diseñar el patrón de relleno de un objeto

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Apariencia".
2. Para definir un fondo transparente para el objeto, seleccione "Transparente".
   Encontrará más información sobre el diseño de un patrón de relleno con un gradiente de color en el capítulo Definición del gradiente de color (Página 8353)
   El objeto se representa transparente.



También puede diseñar el patrón de relleno en la ventana de inspección, en "Propiedades > Propiedades > Patrón de relleno".

Consulte también

Definición del gradiente de color (Página 8353)

10.1.2.15 Formatear los gráficos en un objeto

Introducción

WinCC ofrece la posibilidad de insertar gráficos en algunos objetos y formatearlos. Puede modificar el tamaño, la alineación y las distancias de un gráfico al borde del objeto. Puede modificar las propiedades de un gráfico en la ventana de inspección del objeto en cuestión.

Requisitos

El objeto seleccionado está en modo gráfico.

El objeto seleccionado contiene al menos un gráfico.

Ajustar el tamaño de un gráfico

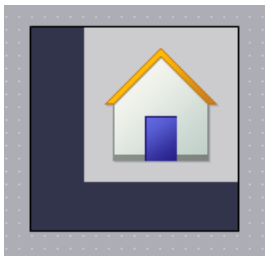
1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Active "No extender imagen" o "Extender imagen".
   El gráfico que contiene el objeto se muestra en el tamaño original o se expande para ajustarse al tamaño del objeto.

Nota

Para ajustar el tamaño del objeto al del gráfico que contiene, active "Adaptar objeto al contenido".

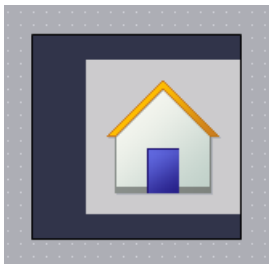
Alinear el gráfico horizontal y verticalmente

1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Para determinar la posición horizontal del gráfico, seleccione, p. ej., "derecha".
3. Para determinar la posición vertical del gráfico, seleccione, p. ej., "arriba".
   El gráfico se representa en la parte superior derecha del objeto.



Definir la distancia hasta el borde del objeto

1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Introduzca el valor para la distancia hasta el borde del objeto, p. ej., "20" para el borde superior.
   El gráfico se representa a una distancia de 20 píxeles del borde superior del objeto.



Símbolo	Descripción
	Determina la distancia hasta el borde izquierdo del objeto.
	Determina la distancia hasta el borde derecho del objeto.
	Determina la distancia hasta el borde superior del objeto.
	Determina la distancia hasta el borde inferior del objeto.

Consulte también

Botón (Página 8450)

Interruptor (Página 8449)

10.1.2.16 Formatear los textos dentro de un objeto

Introducción

Algunos objetos, p. ej., el campo E/S, soportan textos en su interior.

Existen distintas opciones para alinear el texto.

Configurar la distancia hasta el borde del objeto

1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Introduzca, p. ej., el valor "5" para el borde izquierdo.

El texto se alinea a cinco píxeles del borde izquierdo del objeto.

Alinear la posición del texto

1. En la ventana de inspección seleccione "Propiedades > Propiedades > Formato de texto".
2. Seleccione la alineación horizontal, p. ej., centrado.
3. Seleccione la alineación vertical, p. ej., arriba.
   El texto se muestra en la parte superior central del objeto.

Definir la orientación del texto

1. En la ventana de inspección seleccione "Propiedades > Propiedades > Formato de texto".
2. Seleccione la orientación del texto, p. ej., vertical, derecha.
   El texto se muestra verticalmente de abajo arriba.

10.1.2.17 Diseño de objetos basados en tablas

Introducción

WinCC ofrece distintas propiedades para diseñar los objetos de manejo y visualización.

Puede modificar las propiedades de los objetos basados en tablas en la ventana de inspección del objeto en cuestión.

Diseño de colores

En "Propiedades > Propiedades > Apariencia" puede elegir los colores para la fuente, las marcas y las celdas de una tabla.

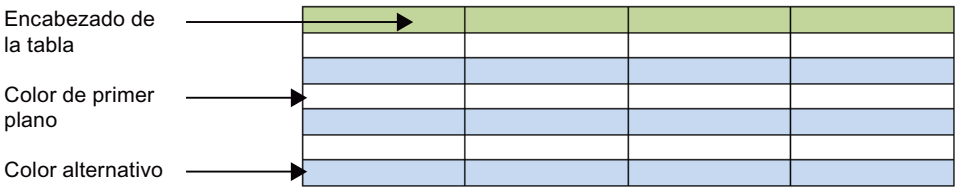


Figura 10-1 Selección de las propiedades de las tablas

Diseño del borde del encabezado de la tabla

1. Abra en la ventana de inspección "Propiedades > Propiedades > Encabezado de tabla Borde".
2. Seleccione para "Estilo", p. ej., compacto.
3. Seleccione para "Ancho", p. ej., 5.
4. Seleccione un color de primer plano, p. ej., rojo.

El borde del encabezado de la tabla se representará en rojo con un ancho de 5 píxeles. En el estilo de borde "Compacto" solamente es visible el color de primer plano.

Colores y gradiente de color en el encabezado de la tabla

1. Seleccione en la imagen el objeto basado en tabla.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > Patrón de relleno del encabezado de tabla".
3. Seleccione para el patrón de relleno, p. ej., "Gradiente horizontal".
4. Seleccione en "Gradiente" un color de fondo, p. ej., azul.
5. Active "Gradiente 1".
6. Elija un color para el "Gradiente 1", p. ej., blanco.
7. Elija un "Ancho" para el gradiente de color, p. ej., 12.
8. Active el gradiente 2.
9. Elija un color para el gradiente 2, p. ej., amarillo.
10. Elija un "Ancho" para el gradiente de color, p. ej., 10.

El encabezado de la tabla se representará con un gradiente de color.

Consulte también

Diseño de bordes (Página 8351)

10.1.2.18 Diseño de bordes

Introducción

WinCC ofrece distintas propiedades para diseñar los objetos de manejo y visualización.

Tomando como ejemplo un campo E/S puede ver la posible configuración del diseño de bordes.

Puede modificar el diseño en la ventana de inspección del objeto en cuestión.

Diseño de bordes

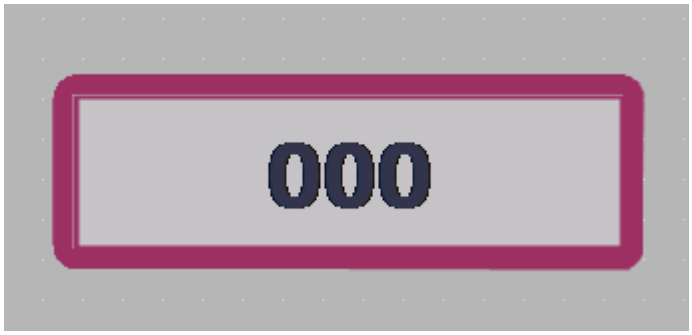
1. Abra la ventana "Propiedades > Propiedades > Apariencia".
2. Elija el ancho del borde, p. ej., "5".
3. Elija el estilo del borde, p. ej., doble línea.
4. Seleccione el color de primer plano.
5. Seleccione el color de fondo.

Figura 10-2 Estilo de borde "Doble línea "

Estilo del borde

La figura siguiente muestra un borde de "Estilo 3D".

La figura siguiente muestra un borde de estilo "Compacto".



Nota

Dependencia de las propiedades según el dispositivo

Los estilos de borde disponibles dependen del objeto y del panel de operador utilizados.

Consulte también

Diseño de objetos basados en tablas (Página 8348)

10.1.2.19 Definición del gradiente de color

Introducción

Para los objetos en WinCC se puede dotar de gradientes de color a diferentes superficies.

Dependiendo de qué superficie se rellene con un gradiente de color, cambiará su categoría en la ventana de inspección. El procedimiento sigue siendo el mismo.

A continuación se configurará el gradiente de color de botones de comando.

Gradiente de color horizontal con dos colores

1. Seleccione un objeto con botones, p. ej., un botón de comando.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > Patrón de relleno".
3. Seleccione "Patrón de relleno > Gradiente horizontal".
4. Elija un color de fondo para el gradiente de color vertical, p. ej., naranja.
5. Active "Gradiente 1".
6. Elija un "Color" para el gradiente 1, p. ej., rojo.
7. Elija un "Ancho" para el gradiente 1, p. ej., "10".

El fondo del botón se representará en naranja.

El gradiente 1 se representará horizontalmente desde el margen izquierdo. El gradiente 1 tiene 10 píxeles de ancho.

Gradiente de color vertical con tres colores

1. Seleccione un botón en la imagen.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > Patrón de relleno".
3. Seleccione el fondo "Gradiente vertical".
4. Seleccione en "Gradiente" un color de fondo, p. ej., naranja.
5. Active "Gradiente 1".
6. Elija un color para el "Gradiente 1", p. ej., rojo.
7. Elija un "Ancho" para el gradiente de color, p. ej., 8.
8. Active el gradiente 2.
9. Elija un color para el gradiente 2, p. ej., amarillo.
10. Elija un "Ancho" para el gradiente de color, p. ej., 10.

	← Color de gradiente 1
	← Color de fondo
	← Color gradiente 2

Con el gradiente de color vertical, el gradiente 1 se muestra de arriba abajo. El gradiente 1 tiene 8 píxeles de ancho.

El fondo del botón se representará en naranja.

El gradiente 2 se muestra en el borde inferior. El gradiente 2 tiene 10 píxeles de ancho.

10.1.2.20 Aplicación de estilos predefinidos

Introducción

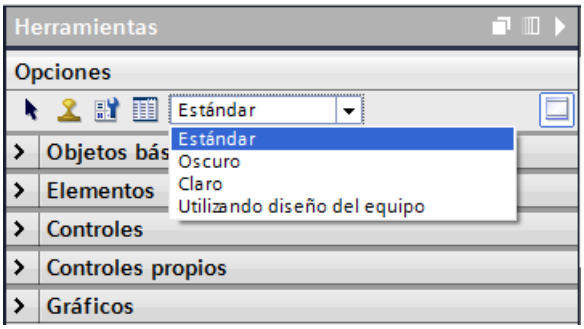
En runtime se pueden asignar estilos predefinidos a objetos y faceplates. Con ayuda de los estilos predefinidos, puede cambiar los colores de fondo de los objetos de indicación y manejo. De esta manera se armoniza la representación en runtime.

Requisitos

La Task Card "Herramientas" está abierta.

Activación de estilos predefinidos

1. Seleccione el objeto que desea insertar en la Task Card "Herramientas".
2. En la barra de herramientas de la Task Card "Herramientas", seleccione una de las siguientes opciones:
   - "Predeterminado" para utilizar el estilo predeterminado
   - "Oscuro" para utilizar el esquema cromático oscuro
   - "Claro" para utilizar el esquema cromático claro
   - "Utilizando diseño del equipo" para utilizar los ajustes del diseño actual del equipo



3. Inserte el objeto deseado en el área de trabajo.
   El objeto se representa con el estilo seleccionado.
4. Los objetos se representan con el estilo seleccionado mientras este permanezca activado en la barra de herramientas.
   Para restablecer el estilo predeterminado, seleccione la opción "Predeterminado" en la barra de herramientas.

Consulte también

Trabajar con estilos (Página 8322)

10.1.2.21 Insertar varios objetos del mismo tipo (sellar)

Introducción

WinCC ofrece la posibilidad de "insertar consecutivamente" varios objetos del mismo tipo directamente uno tras otro, es decir, insertar sin seleccionar de nuevo cada vez el objeto. Además, existe la posibilidad de reproducir las veces que se quiera un objeto ya insertado.

Requisitos

La Task Card "Herramientas" está abierta.

Insertar varios objetos del mismo tipo

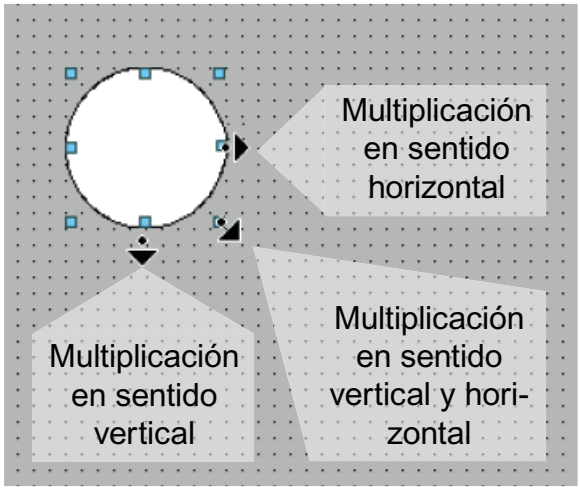
1. Seleccione el objeto que desea insertar en la Task Card "Herramientas".
2. Haga clic en el botón de la barra de herramientas de la Task Card "Herramientas".
   Se activará la función "Sello".
3. Si desea insertar un objeto en su tamaño estándar, haga clic en la posición deseada en el área de trabajo.
   Si desea insertar un objeto en otro tamaño, sitúe el puntero del ratón en la posición deseada en el área de trabajo. Manteniendo pulsado el botón izquierdo del ratón, arrastre el objeto hasta que alcance el tamaño deseado.
   En cuanto suelte el botón del ratón, el objeto quedará insertado en el área de trabajo.
4. Repita el paso 3 para insertar más objetos del mismo tipo.
5. Vuelva a hacer clic en el botón .
   Se desactivará la función "Sello".

Nota

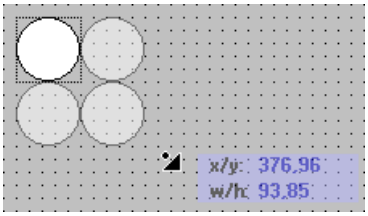
Los objetos existentes se copian utilizando el método de arrastrar y soltar, manteniendo pulsada la tecla <Ctrl>. El objeto existente no se desplazará. En cambio, se insertará una copia del mismo en la nueva posición.

Insertar y reproducir un objeto

1. Inserte el objeto deseado de la Task Card "Herramientas".
2. Pulse la tecla <Ctrl> y sitúe el puntero del ratón en uno de los controladores de tamaño que muestra la figura siguiente.



3. Arrastre el rectángulo con los controladores de tamaño hacia la derecha y/o hacia abajo manteniendo pulsado el botón izquierdo del ratón.
4. Cuanto más desplace el puntero del ratón, más aumentará el objeto hasta rellenar el espacio disponible.



Resultado

Ha insertado un objeto en una imagen y lo ha multiplicado.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.22 Modificar la posición y el tamaño de varios objetos

Cambios posibles

Después de seleccionar varios objetos, edítelos del siguiente modo:

- Desplazar con el ratón
  - Para modificar la posición absoluta de los objetos seleccionados, sitúe el puntero del ratón sobre un objeto y arrastre la selección múltiple manteniendo pulsado el botón del ratón.
  - Para modificar el tamaño de todos los objetos con el mismo factor, arrastre con el ratón el rectángulo con controladores del objeto de referencia.
- Desplazar por el área de trabajo con los botones de la barra de herramientas
  - Modificar la posición de los objetos seleccionados entre sí
  - Igualar la altura y el ancho de los objetos seleccionados
- Desplazar con los comandos del menú contextual del área de trabajo
  - Modificar la posición de los objetos seleccionados entre sí
  - Igualar la altura y el ancho de los objetos seleccionados

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.23 Gráficos externos

Introducción

En WinCC se pueden usar gráficos que han sido creados con un programa gráfico externo. Para utilizar dichos gráficos, guárdelos en el juego de gráficos del proyecto WinCC.

Los gráficos se depositan en la colección de gráficos como se indica a continuación:

- Arrastrando objetos gráficos de la paleta "Gráficos" al área de trabajo con la función de arrastrar y soltar, éstos se guardarán automáticamente en el juego de gráficos. Los nombres de los gráficos se numeran según el orden de creación, p. ej, "Gráfico_1". Para cambiar el nombre del gráfico utilice la tecla <F2>.
- Como archivo gráfico con los formatos siguientes:
  *.bmp, *.ico, *.emf, *.wmf, *.gif, *.tif, *.png, *.jpeg o *.jpg
- Como objeto OLE enlazado con un programa de gráficos externo e incrustado en WinCC. En caso de un enlace OLE, abra WinCC desde el programa de gráficos externo. Edite el objeto enlazado con el programa de gráficos. Un enlace OLE sólo funciona si el programa de gráficos externo es compatible con OLE y el programa de gráficos está instalado en el dispositivo de configuración.

Uso de gráficos del juego de gráficos

Los gráficos de la colección de gráficos se utilizan en las imágenes del modo siguiente:

- En un visor de gráficos
- En una lista de gráficos
- Como rotulación de un botón/tecla de función

Gráficos transparentes

En WinCC también se utilizan gráficos con un fondo transparente. Insertando un gráfico con fondo transparente en un objeto gráficos de WinCC, la transparencia se sustituirá por el color de fondo definido en el objeto gráfico. El color de fondo seleccionado quedará firmemente enlazado con el gráfico. Si se utiliza el gráfico en otro objeto gráfico de WinCC, éste se representará con el mismo color de fondo que el objeto gráfico configurado en primer lugar. Si desea utilizar el gráfico con colores de fondo distintos, vuelva a incorporarlo al juego de gráficos con otro nombre. El color de fondo adicional se configura en el objeto gráfico correspondiente de WinCC al utilizar el gráfico.

Administrar gráficos

Con WinCC se instala un juego completo de gráficos y símbolos, p. ej.:

En la paleta "Gráfico" de la ventana de herramientas, los objetos gráficos están divididos por temas en la "Carpeta de gráficos de WinCC". El acceso directo a la carpeta de gráficos de WinCC no puede eliminarse, editarse ni cambiarse de nombre.

Desde la paleta "Gráficos" se administran también los gráficos externos. Existen las posibilidades siguientes:

- Crear un acceso directo a carpetas de gráficos
  Los gráficos externos depositados en esta carpeta y las subcarpetas se visualizan en la ventana de herramientas, integrándose así en el proyecto.
- Editar accesos directos a carpetas
- El programa para editar el gráfico externo se abre desde WinCC.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.24 Administrar gráficos externos

Introducción

Los gráficos externos a utilizar en WinCC se gestionan en el editor "Imágenes" desde la paleta "Gráficos" de la Task Card "Herramientas".

Requisitos

- El editor "Imágenes" está abierto.
- La Task Card "Herramientas" está abierta.
- Los gráficos existen.
- Los gráficos tienen los siguientes formatos:
  *.bmp, *.ico, *.emf, *.wmf, *.gif, *.tif, *.jpeg o *.jpg

Crear un acceso directo a una carpeta

1. Haga clic en "Carpetas de gráficos propias".
2. Seleccione "Enlace…" en el menú contextual.
   Se abre el cuadro de diálogo "Crear acceso directo a carpeta". Se propondrá un nombre para el nuevo acceso directo.
3. Modifique el nombre si fuese necesario. Seleccione la ruta en la que están depositados los gráficos.
4. Confirme haciendo clic en "Aceptar".
   El nuevo acceso directo se agregará al grupo de objetos "Gráficos". Los gráficos externos contenidos en la carpeta de destino y en las subcarpetas se visualizarán en la ventana de herramientas.

Editar accesos directos a carpetas

1. Seleccione el acceso directo a la carpeta que desea editar.
2. En el menú contextual, elija el comando "Editar acceso directo...".
   Se abre el cuadro de diálogo "Editar acceso directo a carpeta".

3. En caso necesario, cambie el nombre del acceso directo y/o la ruta.
4. Confirme haciendo clic en "Aceptar".

Cambiar el nombre de accesos directos a carpetas

1. Seleccione el acceso directo cuyo nombre desea cambiar.
2. En el menú contextual, seleccione el comando "Cambiar nombre".
3. Ponga un nombre al acceso directo a una carpeta.

Borrar accesos directos a carpetas

1. Seleccione el acceso directo que desea borrar.
2. En el menú contextual, elija el comando "Eliminar".

Editar gráficos externos

1. Seleccione el gráfico que desea editar.
2. En el menú contextual, elija el comando "Editar gráfico".
   Se abrirá el programa de edición de imágenes vinculado al archivo gráfico.

Editar carpetas de gráficos desde WinCC

1. Seleccione el gráfico que desea editar.
2. En el menú contextual, elija el comando "Abrir carpeta de orden superior".
   Se abrirá el Explorador de Windows.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.25 Agregar un gráfico externo a la colección de gráficos

Introducción

Para poder visualizar los gráficos creados con un programa externo en las imágenes, depositelos previamente en el juego de gráficos del proyecto de WinCC.

Requisitos

- Se ha creado una imagen.
- En la imagen se deberá haber insertado un visor de gráficos.
- La ventana de inspección del visor de gráficos deberá estar abierta.

Para depositar un gráfico externo en el juego de gráficos:

- Deberá existir un gráfico.

Para agregar un objeto OLE a la colección:

- Deberá estar instalado un programa de gráficos apto para OLE en el equipo de configuración.

Depositar el archivo gráfico

1. Abra el Explorador de Windows.
2. Seleccione el gráfico que desee depositar.
3. Arrastre el gráfico al juego de gráficos con la función de arrastrar y soltar

Crear y agregar un gráfico nuevo como objeto OLE

1. Seleccione el visor de gráficos en la imagen.
2. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
3. Abra la lista de selección de gráficos.
4. Haga clic en el botón .
5. Se abre el cuadro de diálogo "Insertar objeto".

   Nota

   También se abre el cuadro de diálogo "Ejecutando aplicación externa...". El cuadro de diálogo no se cerrará hasta que se cierre la aplicación externa.

6. En el cuadro de diálogo "Insertar objeto", seleccione la opción "Crear nuevo" y un tipo de objeto. Los tipos de objeto mostrados dependen de los ajustes realizados en "Configuración > Configuración OLE".
7. Haga clic en "Aceptar". Se abre el programa de gráficos correspondiente.
   Cuando haya creado el gráfico, cierre el programa de gráficos con "Archivo > Salir" o "Archivo > Salir y volver a WinCC".
   El gráfico se guarda en el formato estándar del programa de gráficos y se muestra en la colección.

Insertar el gráfico creado en WinCC

Nota

Si crea un gráfico nuevo como objeto OLE, es posible que el gráfico no se muestre directamente en WinCC después de guardarlo en el programa de gráficos externo.

1. Vuelva a abrir el cuadro de diálogo para insertar un gráfico.
2. En el cuadro de diálogo "Insertar objeto", seleccione la opción "Crear de archivo".
3. Haga clic en el botón de comando "Examinar".
4. Desplácese hasta el gráfico que ha creado y selecciónelo.

Agregar un gráfico existente como objeto OLE

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. Abra la lista de selección de gráficos.
3. Haga clic en el botón .
4. Se abre el cuadro de diálogo "Insertar objeto".

   Nota

   También se abre el cuadro de diálogo "Ejecutando aplicación externa...". El cuadro de diálogo no se cerrará hasta que se cierre la aplicación externa.

5. En el cuadro de diálogo "Insertar objeto", seleccione la opción "Crear de archivo".
6. Haga clic en el botón de comando "Examinar".
7. Navegue desde este cuadro de diálogo hasta la carpeta en la que está guardado el archivo gráfico.

   Nota

   Para importar archivos gráficos tenga en cuenta las restricciones siguientes en cuanto al tamaño:

   *.bmp, *.tif, *.emf, *.wmf ≤4 MB

   *.jpg, *.jpeg, *.ico, *.gif "*≤1 MB

Resultado

El gráfico se encuentra en el juego de gráficos. Se puede visualizar en una imagen mediante un visor de gráficos, o bien incluirlo como elemento en una lista de gráficos.

Los objetos OLE depositados en la colección de gráficos se abren haciendo doble clic en ellos con el programa gráfico correspondiente para editarlos. Cuando haya editado el gráfico, cierre el programa de gráficos con "Archivo > Salir" o "Archivo > Salir y volver a WinCC". Los cambios se aplican en WinCC.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.2.26 Trabajar con grupos de objetos

Principios básicos sobre grupos

Introducción

Los grupos se componen de varios objetos que han sido agrupados mediante la función "Agrupar". Un grupo se edita de la misma manera que un objeto individual.

Resumen

WinCC ofrece los métodos siguientes para editar varios objetos conjuntamente.

- Selección múltiple
- Agrupar objetos

Modo de edición

Para editar individualmente un objeto de un grupo, éste se selecciona en la Task Card "Diseño > Niveles".

Otra posibilidad es seleccionar en el menú contextual el grupo de objetos "Agrupar > Editar grupo".

Grupos jerárquicos

Para ampliar un grupo, inserte más objetos o grupos. El grupo se amplia hacia los nuevos objetos y se estructura jerárquicamente en grupos principales y subgrupos u objetos principales y subobjetos. Un grupo jerárquico de este tipo sólo puede disolverse por niveles. El grupo se disuelve siguiendo el mismo orden en el que se agruparon los objetos o grupos. Para disolver un grupo jerárquico de este tipo se requieren los mismos pasos que para agruparlo.

Recuadro que rodea al objeto

En un grupo se mostrará además un rectángulo que rodea el objeto para el grupo completo. Por el contrario, en el caso de una selección múltiple, se mostrarán los rectángulos que rodean el objeto en todos los objetos.

Niveles

Todos los objetos de un grupo se encuentran en el mismo nivel.

Consulte también

Sinopsis de los objetos (Página 8325)

Agrupar objetos

Introducción

El comando de menú "Agrupar" permite juntar varios objetos en un grupo.

El tamaño y la posición del grupo se puede modificar. Para ello rigen las reglas siguientes:

- Si se cambia la posición de un grupo, la posición de los objetos agrupados se adaptará a las coordenadas nuevas. La posición de los objetos agrupados no cambia respecto del grupo.
- Si se cambia el tamaño de un grupo, la altura y el ancho de los objetos agrupados se escalarán en correspondencia.
- Para modificar el tamaño del grupo proporcionalmente, arrastre el recuadro circundante con el ratón hasta alcanzar el tamaño adecuado manteniendo pulsada la tecla <Mayús>.

Nota

Para crear un grupo jerárquico, agrupe los grupos individuales como si fueran objetos.

Requisitos

- Hay como mínimo una imagen abierta con dos objetos.

Agrupar objetos

1. Seleccione todos los objetos que desee agrupar.
2. En el menú contextual, elija el comando "Agrupar > Agrupar".

Los objetos del grupo se representarán con un recuadro circundante.

Agrupar objetos dentro de un grupo

1. Seleccione el grupo que desea editar.
2. En el menú contextual, elija el comando "Agrupar > Editar grupo".
   El grupo que edite se resaltará mediante un marco rojo.
3. Seleccione todos los objetos del grupo que desee agrupar en un subgrupo.
4. En el menú contextual, elija el comando "Agrupar > Agrupar".

Se creará un subgrupo con los objetos.

Incorporar objetos a un grupo existente

1. Seleccione el grupo al que desea agregar objetos.
2. Pulse la tecla <Mayús> y seleccione el objeto que desea incorporar al grupo.
3. En el menú contextual, elija el comando "Agrupar > Agregar al grupo".

El objeto se inserta en el grupo.

Procedimiento alternativo

Los grupos también se pueden editar en la Task Card "Diseño". La función Drag&Drop también permite editar fácilmente los grupos jerárquicos en la paleta "Niveles".

Resultado

Los objetos seleccionados quedan reunidos en un grupo. El recuadro circundante de la selección múltiple se convierte en el recuadro que rodea al grupo. Los recuadros con controladores de tamaño sólo se muestran para el grupo. El grupo se encuentra en el nivel activo.

Deshacer un grupo

Introducción

El comando "Desagrupar" sirve para disolver un grupo y obtener de nuevo objetos individuales.

Requisitos

- Hay una imagen abierta con un grupo.

Deshacer un grupo

1. Seleccione el grupo.
2. En el menú contextual, seleccione el comando "Agrupar > Desagrupar".

Desagrupar objetos dentro de un grupo

1. Seleccione el grupo de nivel superior.
2. En el menú contextual, elija el comando "Agrupar > Editar grupo".
   El grupo que edite se resaltará mediante un marco rojo.
3. Seleccione el grupo de nivel inferior.
4. En el menú contextual, seleccione el comando "Agrupar > Desagrupar".

Resultado

Se deshace el grupo subordinado. Los objetos se asignarán al grupo inmediatamente superior.

Procedimiento alternativo

Los grupos también se pueden editar en la Task Card "Diseño". La función Drag&Drop también permite editar fácilmente los grupos jerárquicos en la paleta "Niveles".

Añadir objetos a un grupo

Introducción

El comando "Añadir al grupo" inserta objetos a un grupo sin desagruparlo primero.

Requisitos

Una imagen con un grupo y por lo menos un objeto adicional deben estar abiertos.

Procedimiento

1. Seleccione el grupo.
2. Pulse la tecla <Mayús> y seleccione el objeto que desea incorporar al grupo.
3. En el menú contextual, elija el comando "Agrupar > Añadir al grupo".

Resultado

El grupo consta ahora de los objetos originales más los objetos añadidos. Los objetos añadidos están situados delante dentro del escalonamiento del grupo.

Procedimiento alternativo

Los grupos también se pueden editar en la Task Card "Diseño". La función Drag&Drop también permite editar fácilmente los grupos jerárquicos en la paleta "Niveles".

Eliminar objetos del grupo

Introducción

El comando "Eliminar del grupo" elimina objetos individuales de un grupo sin deshacerlo primero.

Para editar un objeto en un grupo no es necesario eliminarlo del grupo. Los objetos de un grupo pueden editarse individualmente.

Requisitos

- Hay una imagen abierta con un grupo.

Eliminar objetos de un grupo

Para eliminar un objeto de un grupo:

1. Seleccione el grupo.
2. En el menú contextual, elija el comando "Agrupar > Editar grupo".
   El grupo que se editará se resalta con un marco rojo.
3. Seleccione todos los objetos del grupo que desee eliminar del mismo.
4. En el menú contextual, elija el comando "Agrupar > Eliminar del grupo".

Los objetos se eliminan del grupo.

Nota

Si sólo quedan dos objetos en el grupo, el comando de menú "Eliminar del grupo" no estará disponible.

Borrar objetos de un grupo

Para eliminar un objeto del grupo y de la imagen:

1. Seleccione el grupo.
2. En el menú contextual, elija el comando "Agrupar > Editar grupo".
   El grupo que se editará se resalta con un marco rojo.
3. Seleccione todos los objetos del grupo que desea borrar.
4. En el menú contextual, elija el comando "Borrar".

   Nota

   Si sólo quedan dos objetos en el grupo, el comando de menú "Borrar" no estará disponible.

Procedimiento alternativo

Los grupos también se pueden editar en la Task Card "Diseño". La función Drag&Drop también permite editar fácilmente los grupos jerárquicos en la paleta "Niveles".

Editar un objeto en un grupo

Introducción

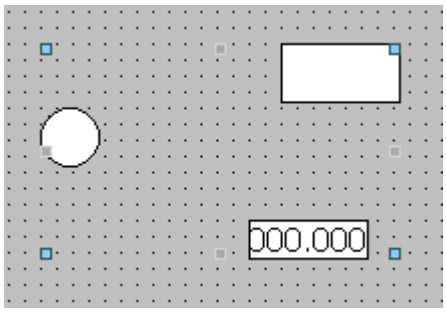
Los objetos de un grupo pueden editarse individualmente.

Requisitos

Hay una imagen abierta con un grupo.

Editar objetos agrupados

1. Seleccione el grupo.

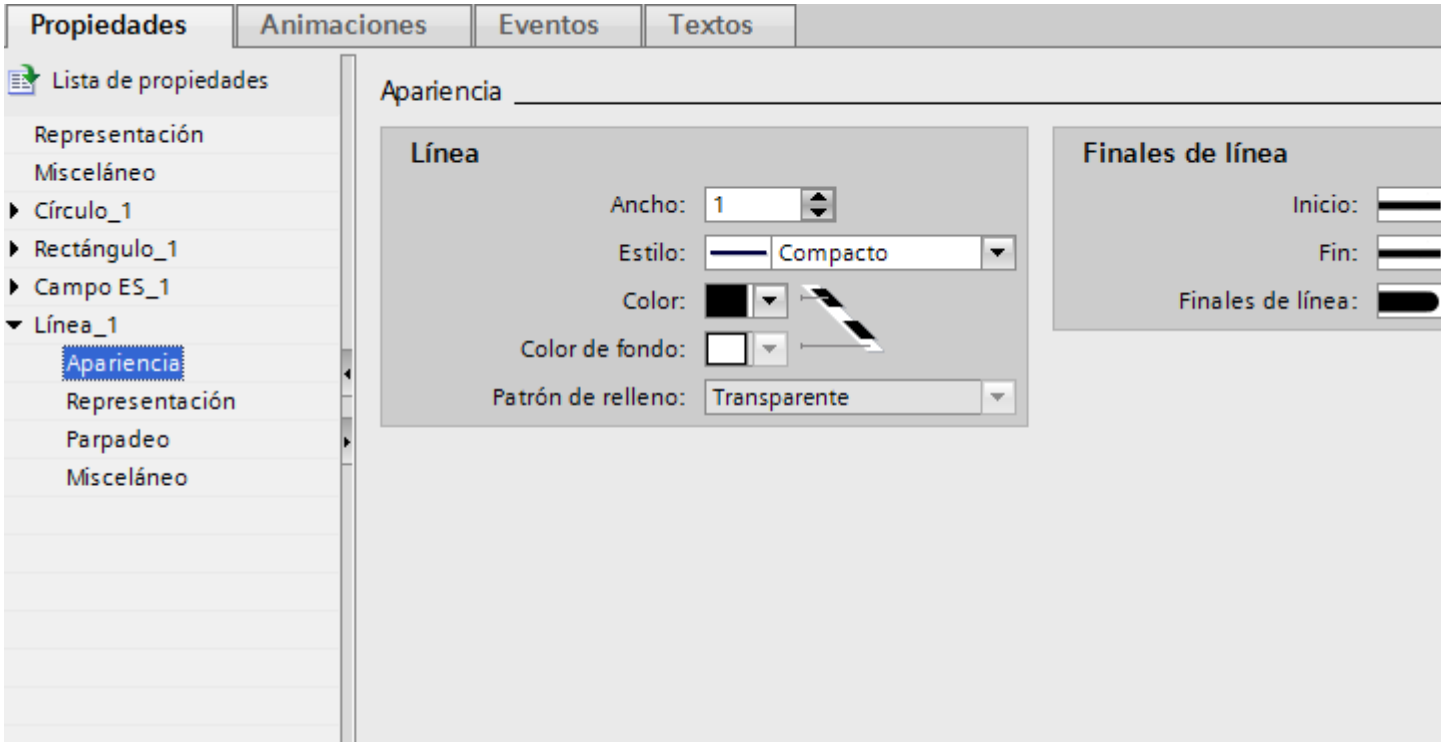


   Las propiedades del grupo se visualizarán en la ventana de inspección.

2. Cambie la posición y el tamaño de los objetos agrupados en "Propiedades > Propiedades > Representación".
3. Cambie el nombre del grupo en "Propiedades > Propiedades > Misceláneo".

Modificar las propiedades de un objeto en un grupo

1. Seleccione el grupo.
2. En la ventana de inspección, seleccione el objeto cuyas propiedades desea modificar.



Se mostrarán las propiedades del objeto.

3. Modifique las propiedades del objeto.

Nota

Dentro de un grupo no es posible dinamizar propiedades de todos los objetos del grupo que contengan la propiedad en cuestión. Solo es posible dinamizar las propiedades de los objetos de un grupo de forma individual para cada objeto.

Resultado

El objeto se modificará, pero seguirá formando parte del grupo. Los demás objetos del grupo no se modificarán.

10.1.2.27 Configuración del acceso con el teclado

Vista general del acceso con el teclado

Introducción

En paneles con teclado y sin ratón, el operador activa los objetos de manejo con la tecla <Tab>. Para que el manejo resulte lo más cómodo posible y para asegurarse de que el operador pueda introducir todos los valores necesarios, debe configurarse la entrada con el teclado. Cuando se utiliza el teclado, los objetos se activan con la tecla <Tab> en un orden determinado y después de introducen los valores necesarios.

En los paneles de operador sin teclado, la tecla <Tab> se simula configurando la función de sistema "SimularTeclaSistema" en una tecla de función.

Autorización de uso y habilitación de operación

Si se configura un objeto para que se maneje con la tecla <Tab>, dicho objeto debe tener una autorización de uso y una habilitación de operación.

Editar el orden de tabuladores

El orden de tabuladores se determina automáticamente al crear los objetos de manejo. Los números del orden de tabuladores se asignan en el mismo orden en el que se crean los objetos de imagen.

En los casos siguientes resulta conveniente modificar el orden de tabuladores:

- El operador cambia directamente a un objeto de manejo determinado
- La imagen requiere un orden concreto

Para modificar el orden de tabuladores, pase al modo de orden de tabuladores. En este modo se visualiza el número de orden de tabuladores en la parte superior izquierda, junto a los objetos de manejo. También se ven los números del orden de tabuladores de los objetos ocultos. La distribución de dichos números se lleva a cabo con el ratón.

Nota

En el modo de orden de tabuladores no hay más funciones disponibles.

Consulte también

Ejemplo: Insertar y configurar un rectángulo (Página 8373)

Sinopsis de los objetos (Página 8325)

Definir la autorización de uso y la habilitación de operación para un objeto

Introducción

Si se configura un objeto para que se maneje con la tecla <Tab>, dicho objeto debe tener una autorización de uso y una habilitación de operación.

Requisitos

Hay como mínimo una imagen abierta con un objeto.

Procedimiento

1. Seleccione el objeto.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > Seguridad".
3. Elija la autorización de uso en "Autorización".
4. Active la autorización de uso.

Resultado

En runtime, el operador puede activar el objeto con la tecla <Tab>.

Consulte también

Ejemplo: Insertar y configurar un rectángulo (Página 8373)

Definir el orden de tabulación

Introducción

En runtime se puede acceder a todos los objetos manejables con la tecla <Tab>. Con el comando "Orden de tabulación" se define el orden en el que se activarán los objetos en runtime.

Nota

En runtime no se puede acceder a los objetos que están en modo "Salida" o "Dos estados" con la tecla <Tab>.

En runtime la imagen se maneja del siguiente modo:

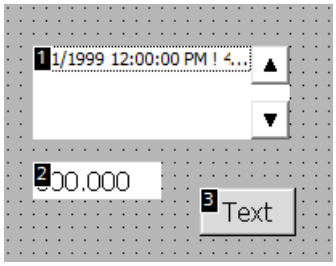
- Con la tecla <Tab>
- Con el ratón
- Con una tecla de acceso directo previamente ajustada

Requisitos

- La imagen activa contiene objetos manejables.
- No hay ningún objeto seleccionado.
- Los objetos están habilitados para poderlos manejar en runtime y tienen una autorización de uso.

Procedimiento

1. Elija el menú "Edición > Orden de tabulación".
   Se activará el modo de orden de tabulación. El número del orden de tabulación se mostrará en todos los objetos manejables. El número del orden de tabulación se mostrará también en los objetos ocultos.
2. Para editar el modo de orden de tabulación, haga clic en los objetos manejables siguiendo el mismo orden en el que deben activarse los objetos en runtime con <Tab>.
   La figura siguiente muestra el orden de tabulación en la imagen. En runtime, la tecla <Tab> activa primero el visor de avisos (número 1), después el campo E/S (número 2) y después el botón (número 3):



3. Para excluir un objeto de imagen del orden de tabulación, mantenga pulsada la combinación de teclas <Mayús+Ctrl> y haga clic en el objeto deseado.
   El número del orden de tabulación ya no se muestra en el objeto de imagen. Ahora, este objeto de imagen está excluido del orden de tabulación. Los números restantes del orden de tabulación se reducen automáticamente en 1.
4. Para volver a incluir un objeto de imagen en el orden de tabulación, repita el paso 3.
   El objeto de imagen se incorpora en el primer lugar del orden de tabulación.

Resultado

En runtime, el operador selecciona con la tecla <Tabulador> los objetos en el orden definido.

Consulte también

Ejemplo: Insertar y configurar un rectángulo (Página 8373)

10.1.2.28 Ejemplos

Ejemplo: Insertar y configurar un rectángulo

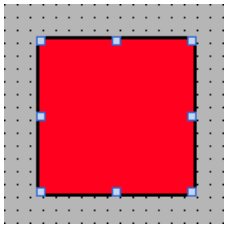
Tarea

En el presente ejemplo se inserta un rectángulo en una imagen. Se configuran las propiedades siguientes:

- Nombre = "MyRectangle"
- Posición = (20, 20)
- Tamaño = (100,100)
- Color = rojo
- Borde negro con un ancho de 2 píxeles

Principio

El rectángulo es un objeto cerrado que se puede rellenar con un color o con una trama. La altura y el ancho del rectángulo se pueden modificar a discreción, por lo que este objeto se puede orientar en sentido horizontal o vertical.



Resumen

Para crear un rectángulo son necesarios los siguientes pasos:

- Insertar un rectángulo
- Configurar un rectángulo

Consulte también

Principios básicos sobre grupos (Página 8361)

Sinopsis de los objetos (Página 8325)

Ejemplo: Insertar un rectángulo

Tarea

En el presente ejemplo se inserta un rectángulo y se cambia su nombre. No utilice los caracteres especiales ?, ", /, \, *, <, > para el nombre.

Requisitos

- Hay una imagen abierta.
- La ventana de inspección está abierta.
- La Task Card "Herramientas" está abierta.

Procedimiento

1. En la Task Card "Herramientas" haga clic en la paleta "Objetos básicos".
2. Arrastre el objeto "Rectángulo" hasta la imagen.
3. Seleccione en la ventana de inspección "Propiedades > Propiedades > Misceláneo".
4. Introduzca el nuevo nombre "MyRectangle".

Resultado

El rectángulo se inserta y su nombre es "MyRectangle". El rectángulo tendrá las propiedades estándar del objeto "Rectángulo".

Ejemplo: Configurar un rectángulo

Tarea

En el presente ejemplo, el rectángulo se configura del siguiente modo:

- Color = rojo
- Borde negro con un ancho de 2 píxeles
- Posición = (20, 20)
- Tamaño = (100,100)

Modificar el color del rectángulo

Para cambiar el color del rectángulo:

1. Seleccione el rectángulo.
2. Defina el color de fondo en la ventana de inspección "Propiedades > Propiedades > Apariencia > Fondo > Color".
3. Seleccione "Compacto" como patrón de relleno.

4. Defina el color del borde en la ventana de inspección "Propiedades > Propiedades > Apariencia > Borde > Color".
5. Introduzca el valor "2" para "Ancho".
6. Seleccione "Compacto" como "Estilo".

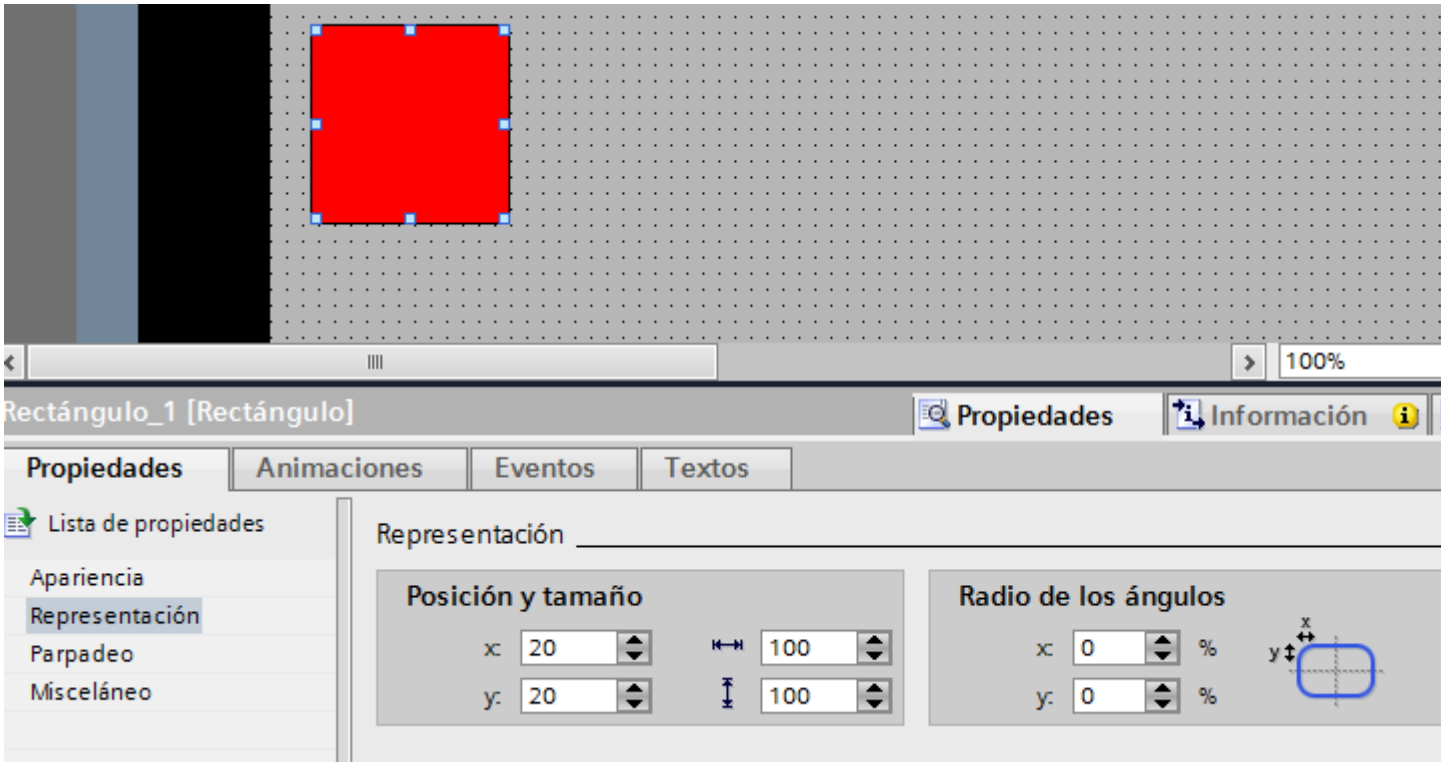
Resultado intermedio

El rectángulo será rojo y tendrá un borde negro de 2 píxeles de ancho.

Modificar la posición y el tamaño del rectángulo.

Para cambiar la posición y el tamaño del rectángulo:

1. Seleccione el rectángulo.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > Representación".



3. Introduzca el valor "20" tanto para X como para Y en "Posición y tamaño".
4. Asigne el valor "100" tanto al ancho como a la altura.

Resultado

El rectángulo se posiciona en las coordenadas (20, 20) y tiene 100 píxeles de ancho y de alto.

10.1.3 Trabajar con listas de textos y gráficos

10.1.3.1 Trabajar con listas de textos

Principios básicos de las listas de textos

Introducción

En una lista de textos los textos están asignados a los valores de una variable. Durante la configuración se asigna la lista de textos, p. ej. a un campo E/S simbólico. Con ello se transfieren al objeto los textos que se van a visualizar. En el editor "Listas de textos" se crean las listas. La conexión de la lista de textos con una variable se configura en el objeto que utiliza la lista de textos.

La selección de los objetos que pueden tener una lista de textos depende del runtime.

Utilización

La lista de textos se utiliza, p. ej. para visualizar una lista de selección en un campo E/S simbólico.

Si el campo E/S simbólico es un campo de visualización, los textos correspondientes se visualizarán en función del valor de la variable configurada. Si el campo E/S simbólico es un campo de entrada, la variable configurada adoptará el valor correspondiente en cuanto el operador selecciona el texto en cuestión en runtime.

Nota

Visualización de valores de variables sin texto

La representación de valores de variables que no tienen asignado ningún texto depende del runtime:

- El objeto de manejo y visualización queda vacío.
- Se muestran tres asteriscos ***.

Textos en varios idiomas

Los textos de una lista pueden configurarse en varios idiomas. Los textos aparecen en runtime en el idioma configurado para el mismo. Para ello, ajuste los idiomas en la ventana del proyecto, bajo "Configuración del idioma > Idiomas del proyecto".

Pasos de configuración

Para visualizar textos, p. ej., en un campo E/S simbólico, son necesarias las siguientes etapas:

1. Crear la lista de textos
2. Asignar los textos a valores o rangos de valores de una lista de textos
3. Asignar una lista de textos en el objeto de visualización, p. ej. al campo E/S simbólico.

Crear una lista de textos

Introducción

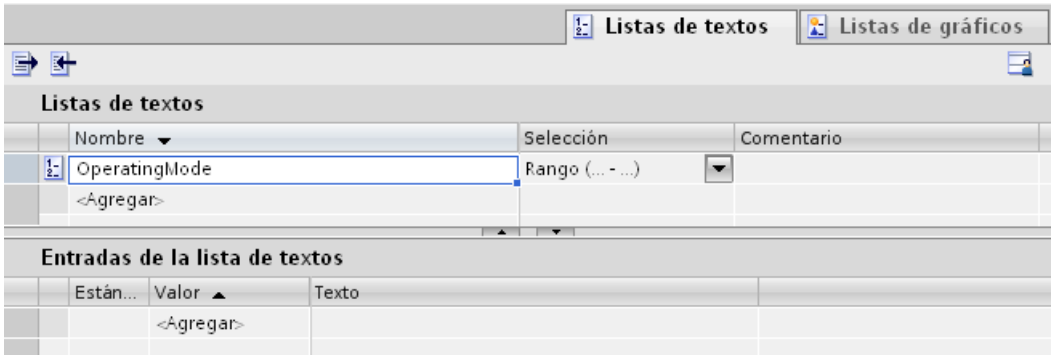
La lista de textos ofrece la posibilidad de asignar textos específicos a los valores y visualizarlos en runtime, p. ej. en un campo E/S simbólico. En ello, puede indicarse el tipo del campo E/S simbólico, p. ej. como mero campo de entrada.

Existen los tipos de listas siguientes:

- Valor/rango
- Bit
- Número de bit

Procedimiento

1. Haga doble clic en "Listas de textos y gráficos" en la ventana del proyecto.
2. Abra la ficha "Listas de textos".



3. Haga clic en "Agregar" en la tabla "Listas de textos".
   Se abre la ventana de inspección de la lista de textos.
4. Introduzca un nombre que describa la lista de textos.
5. En "Selección" elija el tipo de lista de textos:
   - Valor/rango: se mostrará texto de la lista cuando la variable adopte un valor dentro del rango indicado.
   - Bit (0,1): se mostrará un texto de la lista cuando la variable adopte el valor 0, se mostrará otro texto cuando la variable adopte el valor 1.
   - Número de bit (0-31): se mostrará un texto de la lista cuando la variable adopte el número de bit asignado.
6. Introduzca un comentario para la lista de textos.

Nota

En los textos de una lista no se puede emplear punto y coma. El punto y coma es un carácter de control y se borra automáticamente del texto.

Resultado

Se ha creado una lista de textos.

Asignar textos y valores a una lista de textos para rangos

Introducción

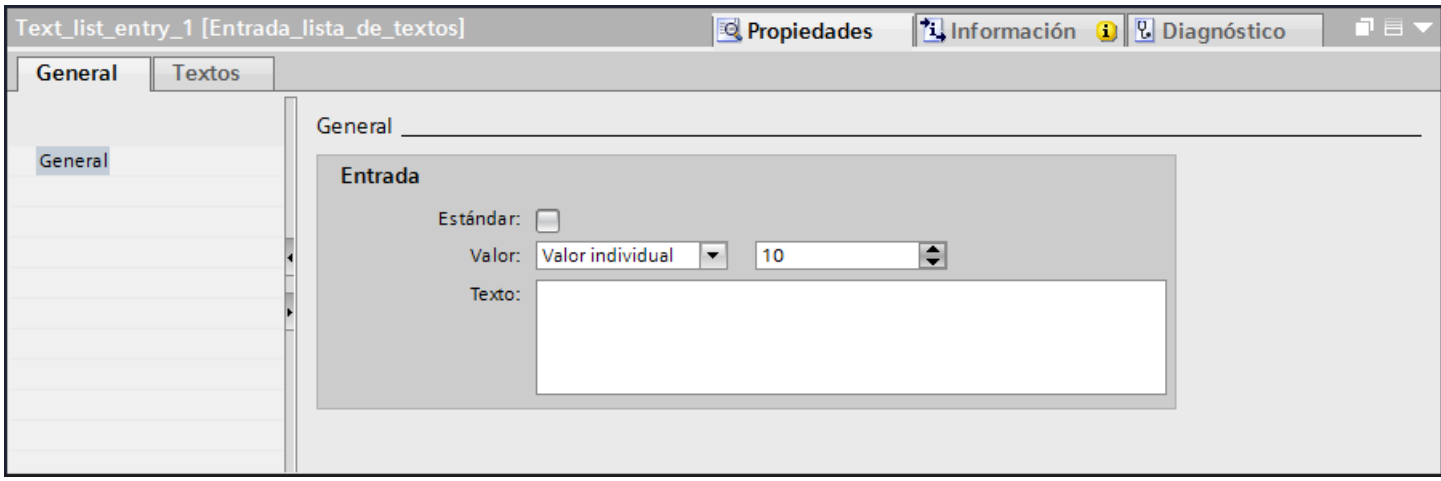
Para cada lista de textos de un rango se indica con qué rango de valores se visualizan los textos y cuáles.

Requisitos

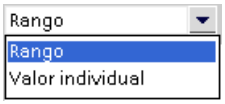
- El editor "Listas de textos y gráficos" está abierto.
- La ficha "Listas de textos" está abierta.
- Se ha creado y seleccionado una lista de textos de área.

Procedimiento

1. Haga clic en "Agregar" en la tabla "Entradas de la lista de textos".
   Se abre la ventana de inspección para esta entrada de la lista.



2. Seleccione el ajuste "Rango" en la ventana de inspección "Propiedades > Propiedades > General > Valor".



   – En "Min" introduzca, p. ej. el valor "1".
   – En "Max" introduzca, p. ej. el valor "20".
   – En "Texto" introduzca el texto que se mostrará en runtime cuando la variable esté dentro del rango de valores especificado.

   Nota

   No utilice punto y coma ni exceda un máximo de 255 caracteres para el texto.

3. Haga clic en "Agregar" en la tabla "Entradas de la lista de textos". Se crea una segunda entrada en la lista.
4. Seleccione el ajuste "Rango" en la ventana de inspección "Propiedades > Propiedades > General > Valor".
   – En "Min" introduzca, p. ej. el valor "21".
   – En "Max" introduzca, p. ej. el valor "40".
   – En "Texto" introduzca el texto que se mostrará en runtime cuando la variable esté dentro del rango de valores indicado.
5. Active "Entrada predeterminada" en caso necesario.
   El texto introducido aparecerá siempre que la variable adopte un valor no definido. Sólo se admite una entrada predeterminada por lista.

Resultado

Se ha creado una lista de textos de área. Los rangos de valores posibles tienen textos asignados.

Asignar textos y valores a una lista de textos para bits

Introducción

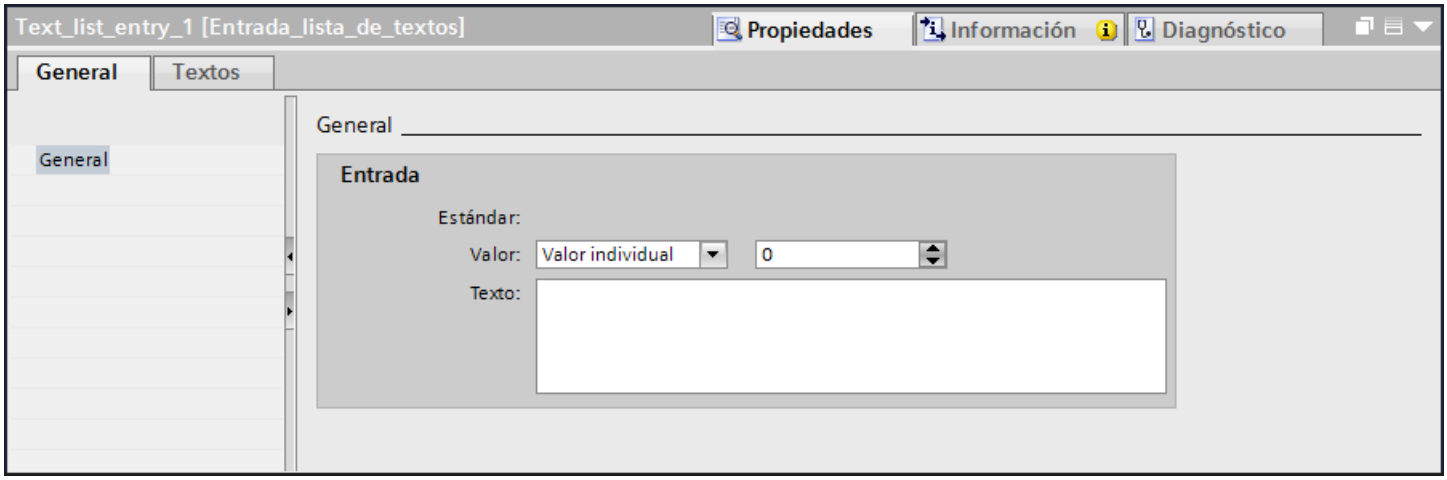
Para cada lista de textos se indica con qué valor del bit se visualiza qué texto.

Requisitos

- El editor "Listas de textos y gráficos" está abierto.
- La ficha "Listas de textos" está abierta.
- Se ha creado y seleccionado una lista de textos para bits.

Procedimiento

1. Haga clic en "Agregar" en la tabla "Entradas de la lista de textos".
   Se abre la ventana de inspección para esta entrada de la lista.



2. Seleccione el ajuste "Valor individual" en la ventana de inspección "Propiedades > Propiedades > General > Valor".
   - En "Valor" introduzca "0".
   - En "Texto" introduzca el texto que se mostrará en runtime cuando la variable de bit se ponga a "0".

   Nota

   No utilice punto y coma ni exceda un máximo de 255 caracteres para el texto.

3. Haga clic en "Agregar" en la tabla "Entradas de la lista de textos". Se crea una segunda entrada en la lista.
4. Seleccione el ajuste "Valor individual" en la ventana de inspección "Propiedades > Propiedades > General > Valor".
   – En "Valor" introduzca "1".
   – En "Texto" introduzca el texto que se mostrará en runtime cuando la variable de bit se ponga a "1".

Resultado

Se ha creado una lista de textos para bits. Los valores posibles "0" y "1" tienen textos asignados que se muestran en runtime.

Asignar textos y valores a una lista de textos para números de bit

Introducción

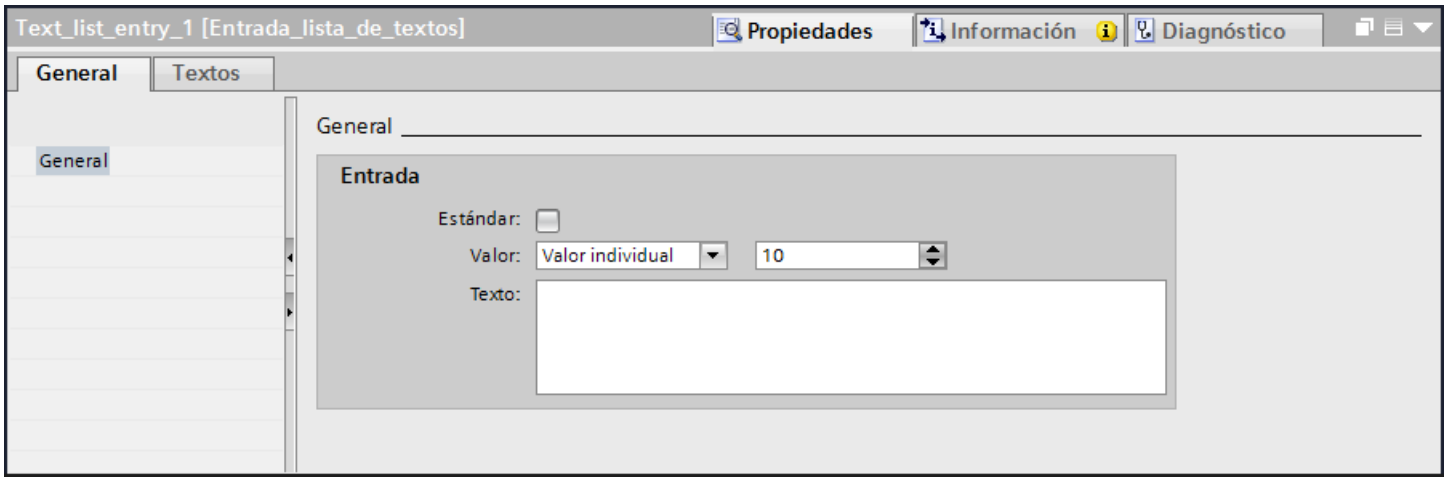
Para cada lista de textos para números de bit se indica con qué número de bit se visualizan los textos y cuáles.

Requisitos

- El editor "Listas de textos y gráficos" está abierto.
- La ficha "Listas de textos" está abierta.
- Se ha creado y seleccionado una lista de textos para números de bit.

Procedimiento

1. Haga clic en "Agregar" en la tabla "Entradas de la lista de textos".
   Se abre la ventana de inspección para esta entrada de la lista.



2. Seleccione el ajuste "Valor individual" en la ventana de inspección "Propiedades > Propiedades > General > Valor".
   - En "Valor" introduzca "10", p. ej.
   - En "Texto" introduzca el texto que se mostrará en runtime cuando la variable adopte el valor "10".

   Nota

   No utilice punto y coma ni exceda un máximo de 255 caracteres para el texto.

3. Active "Entrada predeterminada" en caso necesario.
   El texto introducido aparecerá siempre que la variable adopte un valor no definido. Sólo se admite una entrada predeterminada por lista.
4. Cree entradas adicionales en la lista para otros números de bit de la misma lista de textos.

Resultado

Se ha creado una lista de textos para números de bit. Los números de bit indicados tienen textos asignados que se visualizan en runtime.

Configurar un objeto con lista de textos

Introducción

El valor de salida y la aplicación del valor para las listas de textos se definen en el objeto de manejo y visualización que visualiza los textos de la lista en runtime. Las propiedades de estos objetos se configuran según convenga.

Requisitos

- Se ha creado una lista de textos.
- Se ha creado una variable.
- El editor "Imágenes" está abierto.
- Hay una imagen abierta con un campo E/S simbólico. El objeto está seleccionado.

Procedimiento

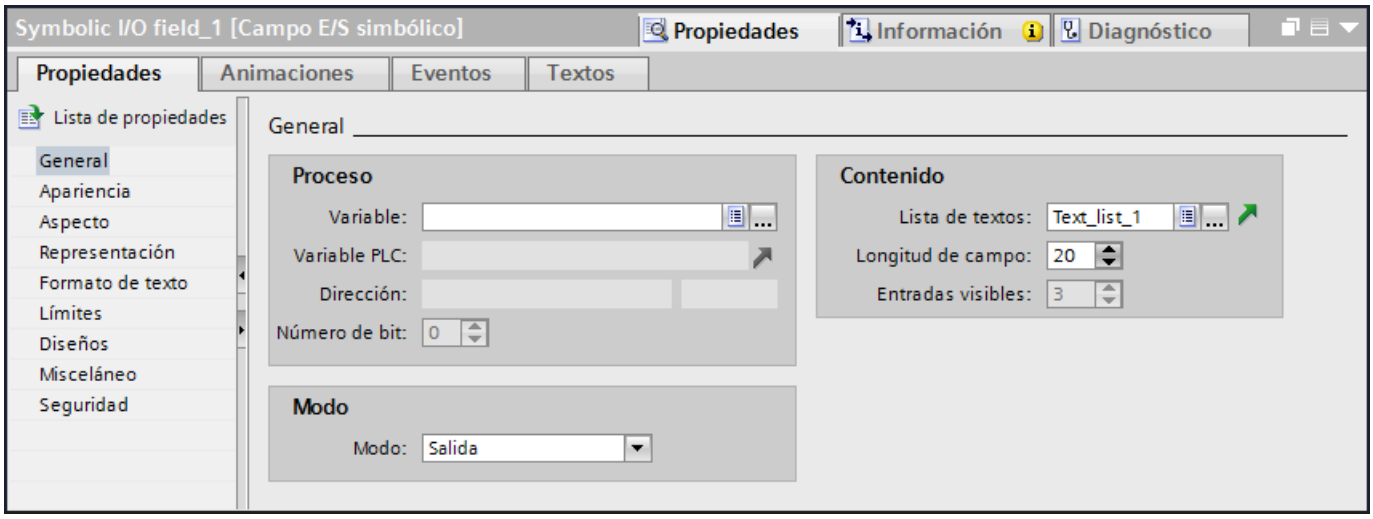
1. En la ventana de inspección "Propiedades > Propiedades > General > Lista de textos", seleccione la lista de textos que debe visualizarse en runtime.
2. En "Modo", seleccione el ajuste "Salida".

   Nota

   Dependencia de runtime

   Según sea el runtime habrá distintos tipos de campos disponibles para un campo E/S simbólico.

3. En "Variable" seleccione la variable cuyos valores determinan la visualización en el campo E/S simbólico.



Resultado

Cuando la variable adopte el valor indicado, en runtime se mostrarán los textos definidos en la lista de textos en el campo E/S simbólico.

10.1.3.2 Trabajar con listas de gráficos

Principios básicos de las listas de gráficos

Introducción

En una lista de gráficos hay determinados gráficos asignados a los valores posibles de una variable. Durante la configuración se asigna la lista de gráficos a un botón o a un campo E/S gráfico. Con ello se transfieren al objeto los gráficos que se visualizarán.

Las listas de gráficos se crean con el editor "Listas de textos y gráficos". La conexión de la lista de gráficos con una variable se configura en el objeto que utiliza la lista de gráficos. La disponibilidad de la lista de gráficos depende del panel de operador utilizado.

Utilización

La lista de gráficos se configura para los casos siguientes:

- Lista de selección con un campo E/S gráfico
- Gráfico dependiente del estado en un botón

Los gráficos de una lista de gráficos pueden configurarse en varios idiomas. Los gráficos se visualizan en runtime en el idioma configurado en cada caso.

Origen de los gráficos

Los gráficos pueden insertarse en la lista de gráficos desde los orígenes siguientes:

- Selección del juego de gráficos
- Selección de un archivo existente
  Pueden emplearse los tipos de archivos siguientes:
  *.bmp, *.ico, *.emf, *.wmf, *.gif, *.tiff, *.png, *.jpeg y *.jpg.
- Creación de un archivo nuevo

Función

Si el campo E/S gráfico es un campo de visualización, los gráficos correspondientes se mostrarán en función del valor de la variable configurada. Si el campo E/S gráfico es un campo de entrada, la variable configurada adopta el valor correspondiente en cuanto el operador seleccione un gráfico en runtime.

Pasos de configuración

Para visualizar gráficos, p. ej. en un campo E/S gráfico, se requieren los pasos siguientes:

1. Crear la lista de gráficos
2. Asignar los gráficos a valores o rangos de valores de una lista de gráficos
3. Asignar una lista de gráficos en el objeto de visualización, p. ej. el campo E/S gráfico

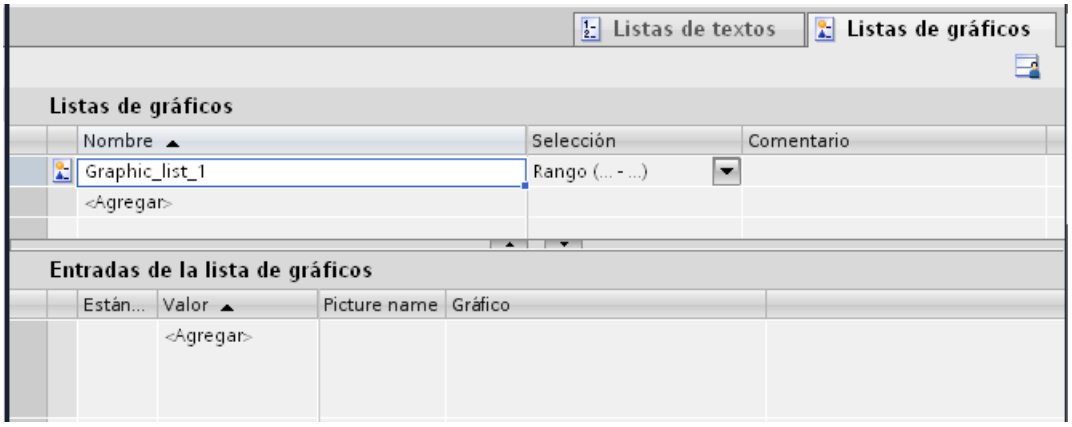
Crear una lista de gráficos

Introducción

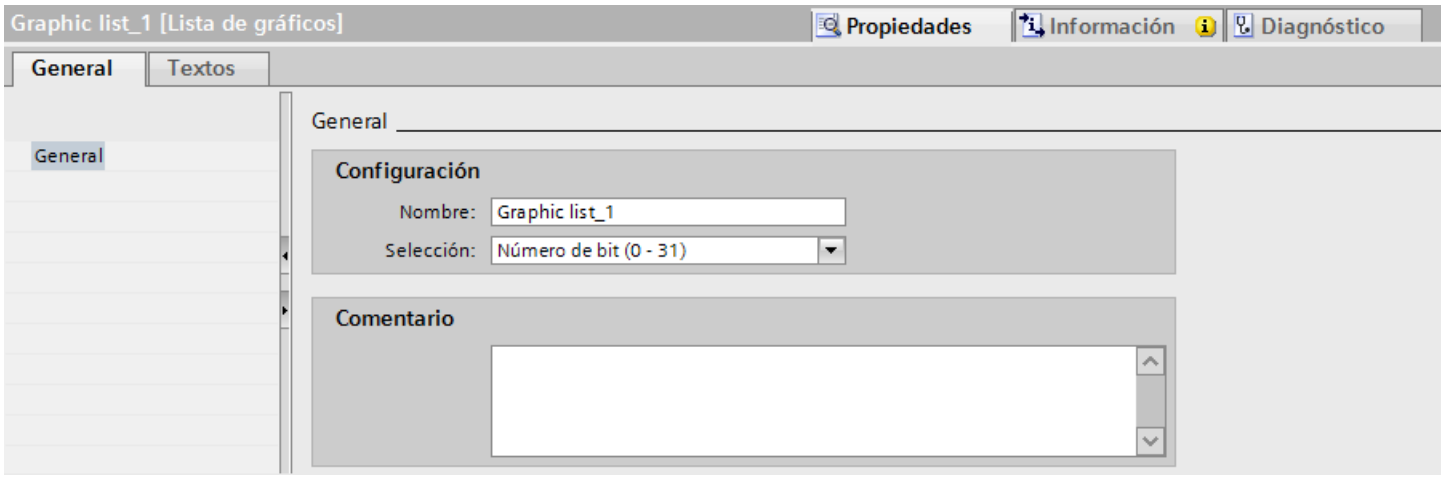
La lista de gráficos ofrece la posibilidad de asignar gráficos específicos a valores variables y visualizarlos en runtime en un campo E/S gráfico. En ello, puede indicarse el tipo del campo E/S gráfico, p. ej. como mero campo de salida.

Procedimiento

1. Haga doble clic en "Listas de textos y gráficos" en la ventana del proyecto
2. Abra la ficha "Listas de gráficos".



3. Haga clic en "Agregar" en la tabla "Listas de gráficos". Se abre la ventana de inspección de la lista de gráficos.



4. Introduzca un nombre que describa la lista de gráficos.
5. En "Selección" elija, p. ej., el tipo de lista de gráficos "Número de bit (0 - 31)"
6. Introduzca un comentario para la lista de gráficos.

Resultado

Se ha creado una lista de gráficos del tipo "Rango (0 - 31)".

Asignar gráfico y valores a una lista de gráficos para rangos

Introducción

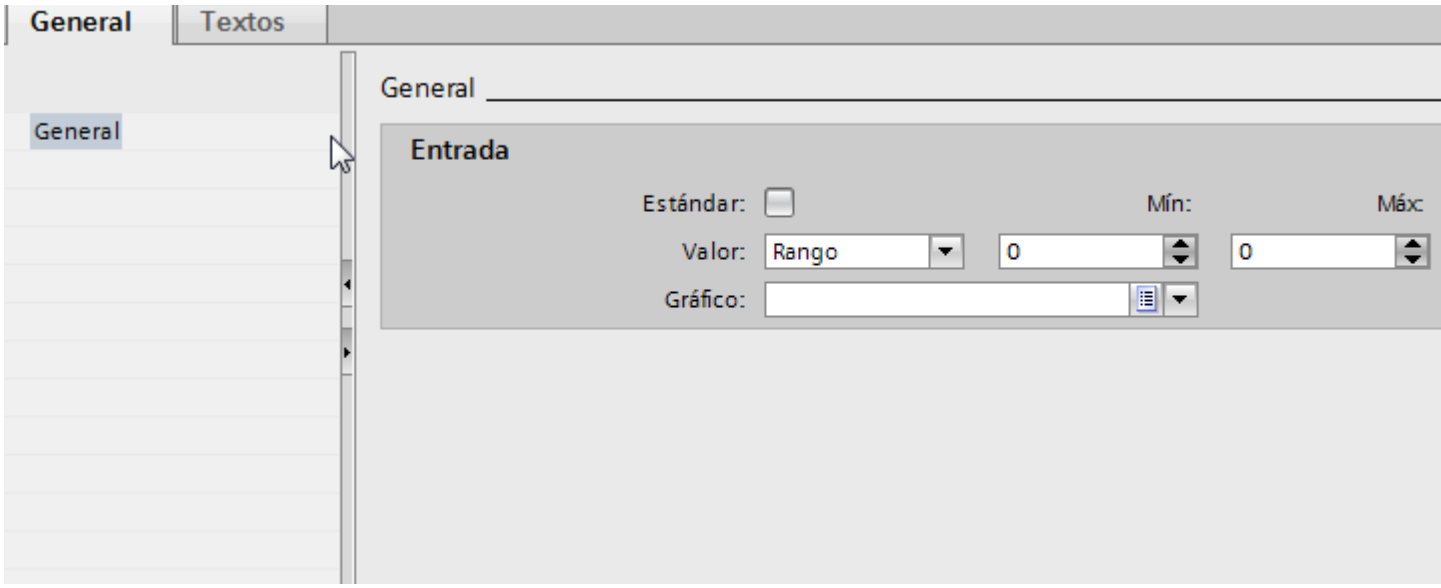
Para cada lista de gráficos de rango se indica con qué rango de valores se visualizan los gráficos y cuáles.

Requisitos

- El editor "Listas de textos y gráficos" está abierto.
- La ficha "Listas de gráficos" está abierta.
- Se ha creado y seleccionado una lista de gráficos de un rango.

Procedimiento

1. Haga clic en "Agregar" en la tabla "Entradas de la lista de gráficos".
   Se abre la ventana de inspección para esta entrada de la lista.



2. En la ventana de inspección "Propiedades > Propiedades > General > Valor" seleccione el ajuste "Rango":
   - En "Min" introduzca, p. ej. el valor "1".
   - En "Max" introduzca, p. ej. el valor "20".
   - Seleccione el gráfico que se mostrará en runtime cuando la variable esté dentro del rango de valores indicado.

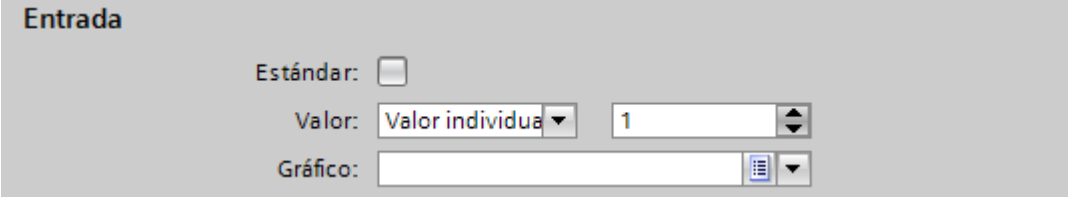
Nota

En lugar de utilizar la lista desplegable, también es posible insertar gráficos desde librerías o desde el sistema de archivos:

1. Seleccione un gráfico de la librería o del sistema de archivos.
2. Arrastre el gráfico a la tabla "Entradas de la lista de gráficos > Gráfico" con la función Drag&Drop.

3. Haga clic en "Agregar" en la tabla "Entradas de la lista de gráficos". Se crea una segunda entrada en la lista.

4. En la ventana de inspección "Propiedades > Propiedades > General > Valor" seleccione el ajuste "Valor individual":
   – Introduzca, p. ej. el valor "21".
   – Seleccione el gráfico que se mostrará en runtime cuando se establezca el bit "21" en la variable.



5. Active "Entrada predeterminada" en caso necesario.
   El gráfico aparecerá siempre que la variable adopte un valor no definido. Sólo se admite una entrada predeterminada por lista.

Resultado

Se ha creado una lista de gráficos de un rango. Los valores posibles tienen gráficos asignados que se visualizan en runtime.

Asignar gráficos y valores a una lista de gráficos para bits

Introducción

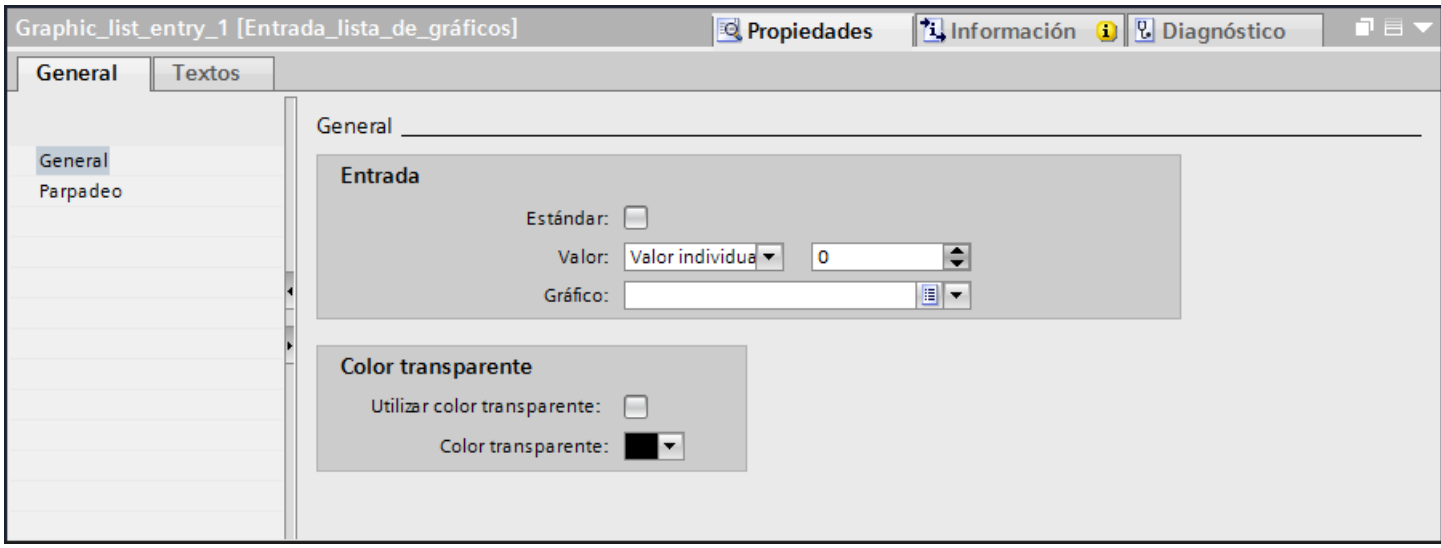
Para cada lista de gráficos se indica con qué valor del bit se visualizan los gráficos y cuáles.

Requisitos

- El editor "Listas de textos y gráficos" está abierto.
- La ficha "Lista de gráficos" está abierta.
- Se ha creado y seleccionado una lista de gráficos para bits.

Procedimiento

1. Haga clic en "Agregar" en la tabla "Entradas de la lista de gráficos".
   Se abre la ventana de inspección para esta entrada de la lista.



2. En la ventana de inspección "Propiedades > Propiedades > General > Valor" seleccione el ajuste "Valor individual":
   - Introduzca el valor "0".
   - Seleccione el gráfico que se mostrará en runtime cuando se establezca el bit "0" en la variable.

   Nota

   En lugar de utilizar la lista desplegable, también es posible insertar gráficos desde librerías o desde el sistema de archivos:
   1. Seleccione un gráfico de la librería o del sistema de archivos.
   2. Arrastre el gráfico a la tabla "Entradas de la lista de gráficos > Gráfico" con la función Drag&Drop.

3. Haga clic en "Agregar" en la tabla "Entradas de la lista de gráficos". Se crea una nueva entrada en la lista.
4. Seleccione en la ventana de inspección "Propiedades > Propiedades > General > Valor > Valor individual":
   - Introduzca el valor "1".
   - Seleccione el gráfico que se mostrará en runtime cuando se establezca el bit "1" en la variable.

Resultado

Se ha creado una lista de gráficos para bits. Los valores "0" y "1" tienen gráficos asignados que se visualizan en runtime.

Asignar gráficos y valores a una lista de gráficos para números de bit

Introducción

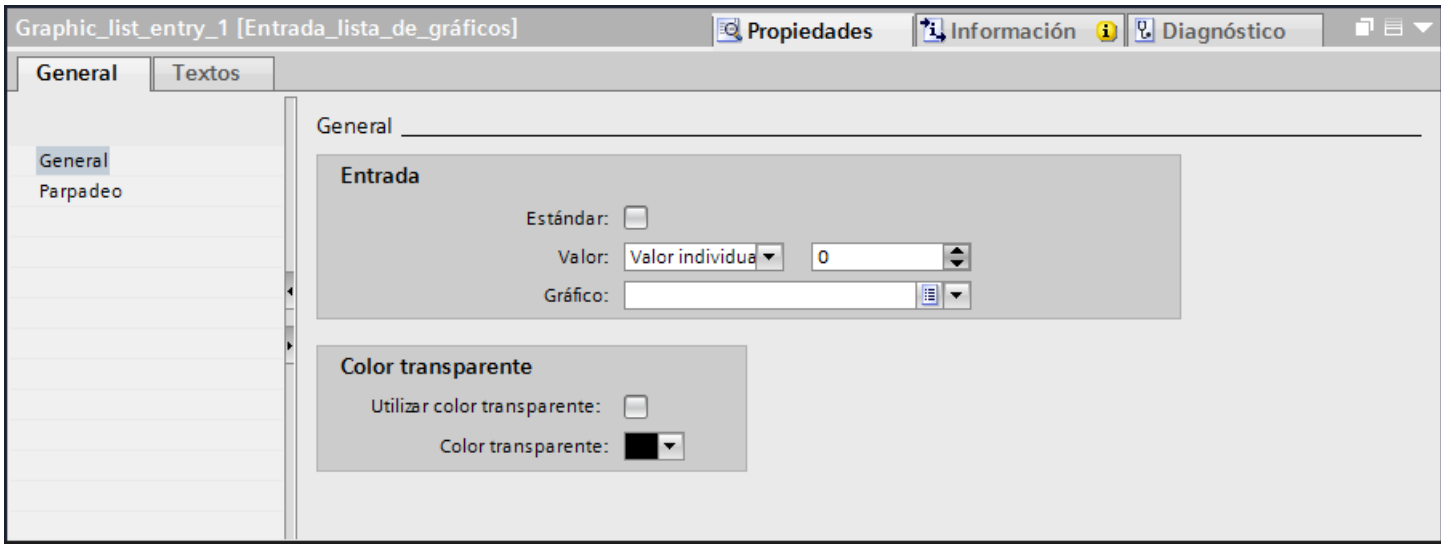
Para cada lista de gráficos para números de bit se indica con qué número de bit deben visualizarse los gráficos y cuáles.

Requisitos

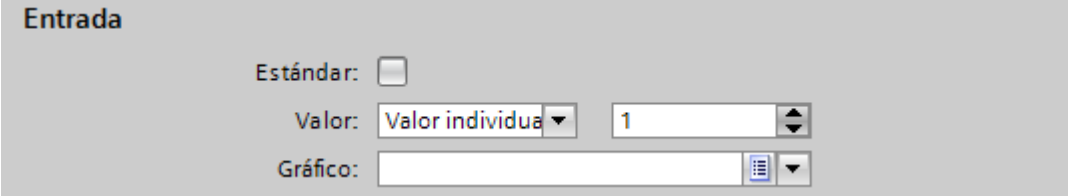
- El editor "Listas de textos y gráficos" está abierto.
- La ficha "Listas de gráficos" está abierta.
- Se ha creado y seleccionado una lista de gráficos para números de bit.

Procedimiento

1. Haga clic en "Agregar" en la tabla "Entradas de la lista de gráficos".
   Se abre la ventana de inspección para esta entrada de la lista.



2. En la ventana de inspección "Propiedades > Propiedades > General > Valor" seleccione el ajuste "Valor individual":
   - Introduzca como valor, p.°ej.,"0".
   - Seleccione el gráfico que se mostrará en runtime cuando se establezca el bit "0" en la variable.



> **Nota**
>
> En lugar de utilizar la lista desplegable, también es posible insertar gráficos desde librerías o desde el sistema de archivos:
>
> 1. Seleccione un gráfico de la librería o del sistema de archivos.
> 2. Arrastre el gráfico a la tabla "Entradas de la lista de gráficos > Gráfico" con la función Drag&Drop.

3. Active "Entrada predeterminada" en caso necesario.
   El gráfico aparecerá siempre que la variable adopte un valor no definido. Sólo se admite una entrada predeterminada por lista.
4. Cree entradas adicionales para otros números de bit de la misma lista de gráficos.

Resultado

Se ha creado una lista de gráficos para números de bit. Los números de bit indicados tienen gráficos asignados que se visualizan en runtime.

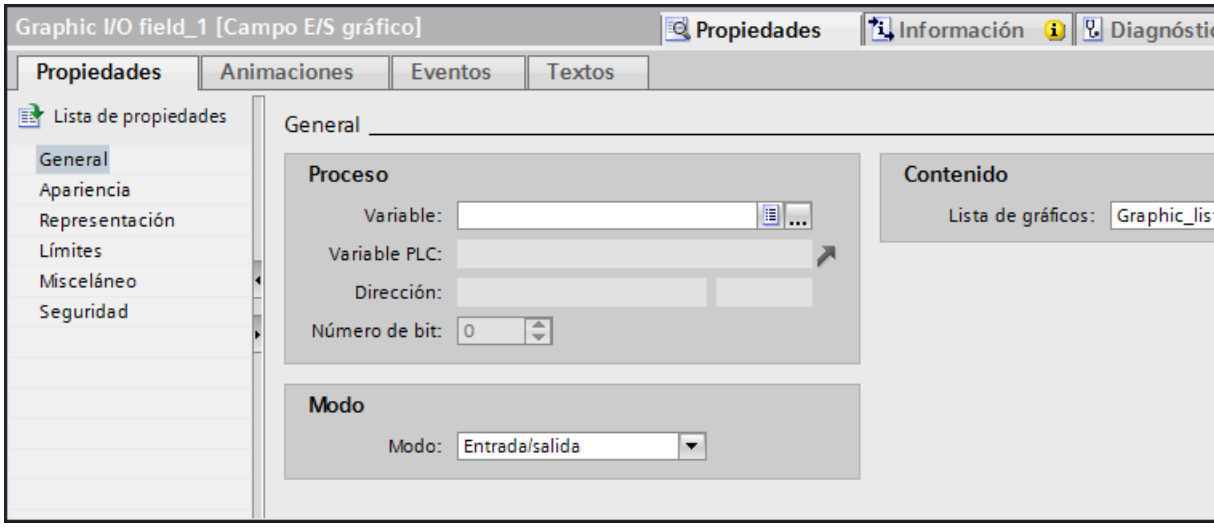
Configurar un objeto con lista de gráficos

Introducción

El valor de salida y la aplicación del valor para las listas de gráficos se definen en el objeto de manejo y visualización que visualiza los gráficos de la lista en runtime. Las propiedades de estos objetos se configuran según convenga.

Requisitos

-
-
-
-



Procedimiento

1.

2.

Nota

Dependencia de runtime

Según sea el runtime habrá distintos tipos de campos disponibles para un campo E/S gráfico.

3. En "Variable" seleccione la variable cuyos valores determinan la visualización en el campo E/S gráfico.

Resultado

En runtime se mostrarán los gráficos definidos de la lista en el campo E/S gráfico cuando la variable adopte el valor indicado.

10.1.4 Dinamización de imágenes

10.1.4.1 Principios básicos de la dinamización

Dinamización de objetos

En WinCC se dinamizan objetos para proyectar su instalación en paneles de operador y visualizar el transcurso de los procesos.

Las dinamizaciones se realizan mediante

- Animaciones
- Variables
- Funciones de sistema

Un ejemplo es la reproducción de un depósito cuyo nivel de líquido aumenta o disminuye en función de un valor de proceso.

Las posibilidades de dinamización dependen del objeto. Al copiar un objeto se copian también sus dinamizaciones.

Consulte también

Dinamización en la ventana de inspección (Página 8393)

Configurar una animación nueva (Página 8395)

Principios básicos de los eventos (Página 8404)

10.1.4.2 Dinamización en la ventana de inspección

Introducción

En principio se pueden dinamizar todos los objetos de imagen que se hayan configurado en una imagen. Las posibilidades de dinamización y los eventos dependen del panel de operador y del objeto seleccionado.

Animaciones

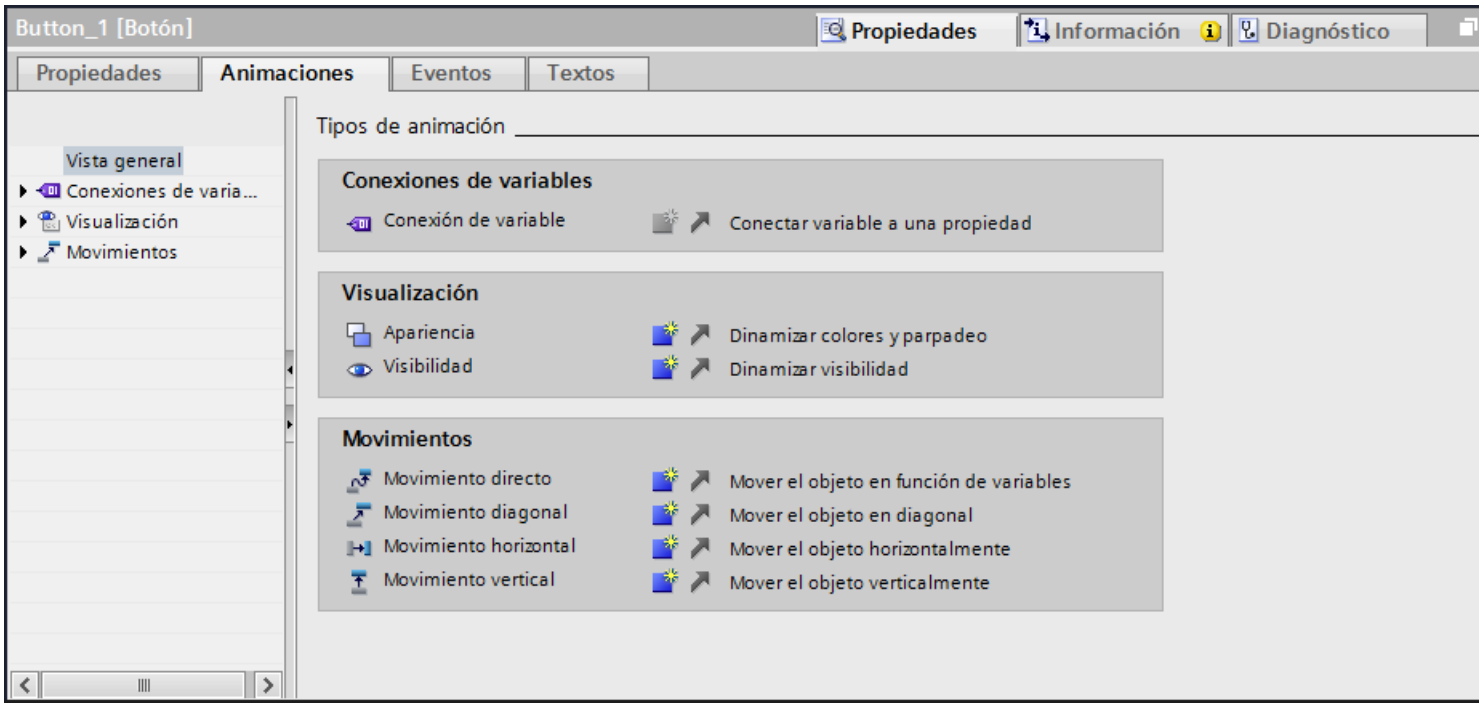
WinCC le asiste en la dinamización mediante animaciones predefinidas. Si se desea animar un objeto, en primer lugar debe configurarse la animación deseada en la ventana de inspección del objeto. A continuación, adapte la animación a las exigencias de su proyecto.

La selección de animaciones asistidas depende del panel de operador y del objeto seleccionado. Seleccione uno de los siguientes tipos de animación:

- Representación: apariencia, visibilidad
- Movimientos: movimiento directo, diagonal, horizontal y vertical
- Enlace de variables

Se puede configurar varias veces el tipo de animación "Enlace de variables" para un mismo objeto.

Las animaciones se configuran en la ventana de inspección "Propiedades > Animaciones".

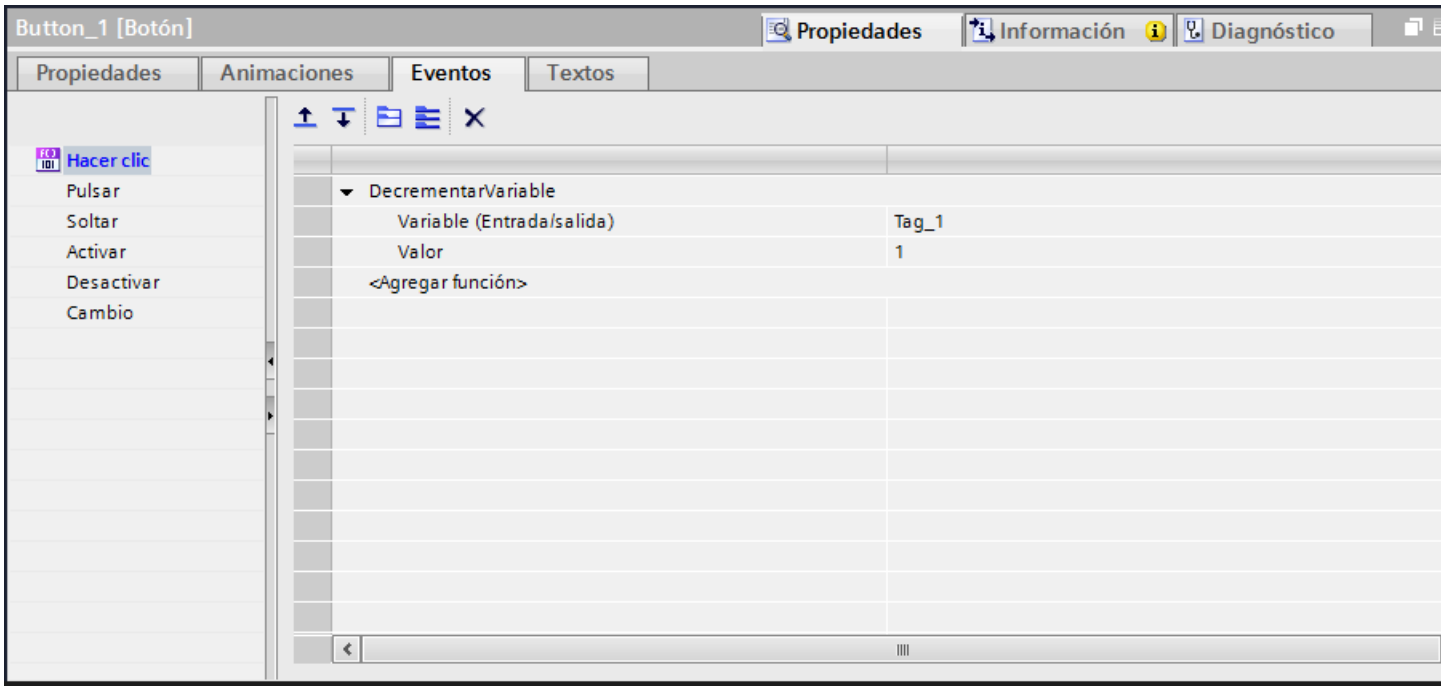


Eventos

Los objetos manejables reaccionan además a eventos, p. ej. a un clic del ratón.

Dentro de un evento, se configura una lista de funciones con funciones de sistema. Como reacción al evento desencadenado, se procesarán las funciones de sistema.

Los eventos se configuran en la ventana de inspección "Propiedades > Eventos".



Encontrará más información sobre el tema en "Trabajar con listas de funciones".

Consulte también

Principios básicos de la dinamización (Página 8391)

10.1.4.3 Dinamizar con animaciones

Configurar una animación nueva

Introducción

Para dinamizar objetos de imagen se utilizan animaciones predefinidas.

Requisitos

- Hay una imagen abierta con un objeto dinamizable, como mínimo.
- La ventana de inspección está abierta.
- Se visualiza la ventana de herramientas.

Procedimiento en la ventana de inspección

1. Haga clic en la ventana de inspección "Propiedades > Animaciones".
2. Seleccione la animación deseada.
3. Haga clic en el botón .

Procedimiento en la Task Card "Animaciones"

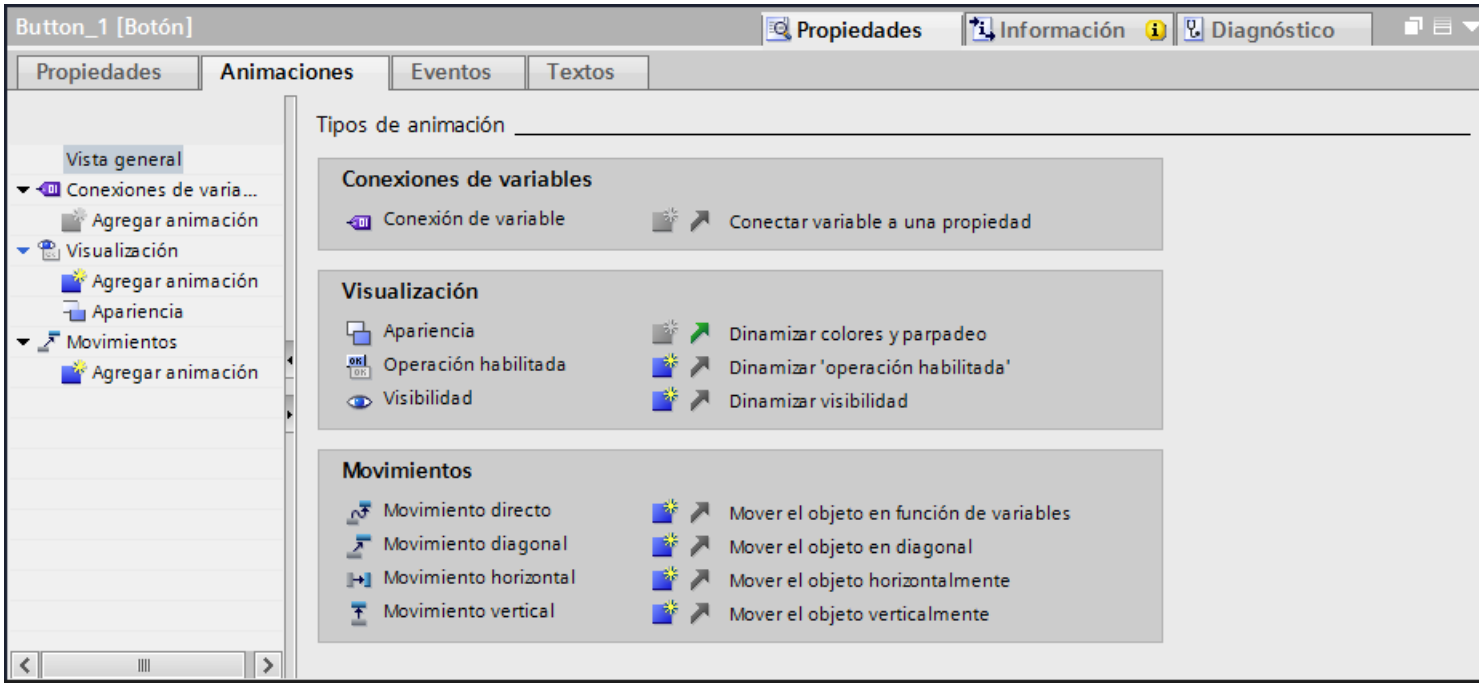
1. En la Task Card "Animaciones" abra el grupo de objetos que contenga la animación deseada.
2. Arrastre la animación deseada hasta el objeto que desee dinamizar.

Otra posibilidad es seleccionar el objeto en la imagen y hacer doble clic en la animación deseada en la Task Card "Animación".

Resultado

La animación se registrará en la ventana de inspección del objeto. En las siguientes etapas se configura la animación.

En la vista general de las animaciones se simboliza mediante una flecha verde la animación que ya está configurada. Si hace clic en una flecha verde, se abre la animación configurada en la ventana de inspección.



Consulte también

Dinamizar la visibilidad de un objeto (Página 8401)

Principios básicos de la dinamización (Página 8391)

Dinamizar la apariencia de un objeto

Introducción

La apariencia de un objeto de imagen se modifica en runtime a través del cambio de valor de una variable. Cuando la variable adopte un valor determinado, cambiará el color o el comportamiento de parpadeo del objeto de imagen según lo que se haya configurado.

Tipo

Según la selección se contemplarán rangos o valores individuales de las variables en runtime. La apariencia del objeto cambia según la configuración.

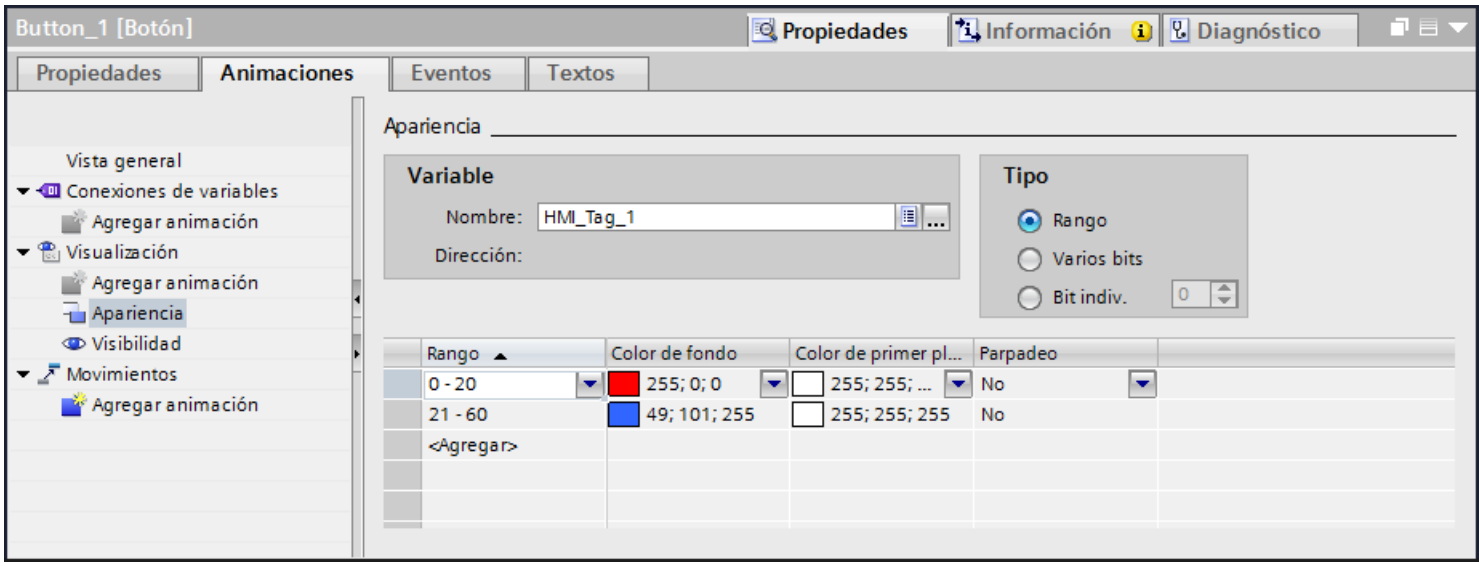
Requisitos

- Hay una imagen abierta.
- En la imagen hay un objeto dinamizable y está seleccionado.
- La ventana de inspección está abierta.
- Se visualiza la ventana de herramientas.

Procedimiento

1. Haga clic en la ventana de inspección "Propiedades > Animaciones".
   Aparecerán las animaciones disponibles para el objeto seleccionado.
2. Seleccione la animación "Apariencia" y haga clic en el botón .
   Se visualizarán los parámetros de la animación.
3. Seleccione una variable en "Variable > Nombre".
4. Seleccione, p. ej., "Tipo > Rango".
5. Haga clic en "Agregar" en la tabla.
6. Introduzca en la columna "Rango", p. ej., el intervalo de variables "0 - 20".
7. Seleccione en "Color de primer plano" y "Color de fondo" los colores que adoptará el objeto cuando la variable alcance el intervalo en runtime.

8. En la lista "Parpadeo", seleccione el comportamiento de parpadeo del objeto.
9. Para crear un intervalo de valores más amplio, p. ej., "21 - 60", repita los pasos 5 a 8.



Resultado

El comportamiento de parpadeo y el color del objeto cambiarán en runtime en función del valor de proceso de la variable.

Configurar un movimiento

Introducción

Los objetos dinámicos se pueden configurar de forma que se muevan a lo largo de una pista determinada. El movimiento se controla con variables. El objeto se mueve en cada actualización de las variables.

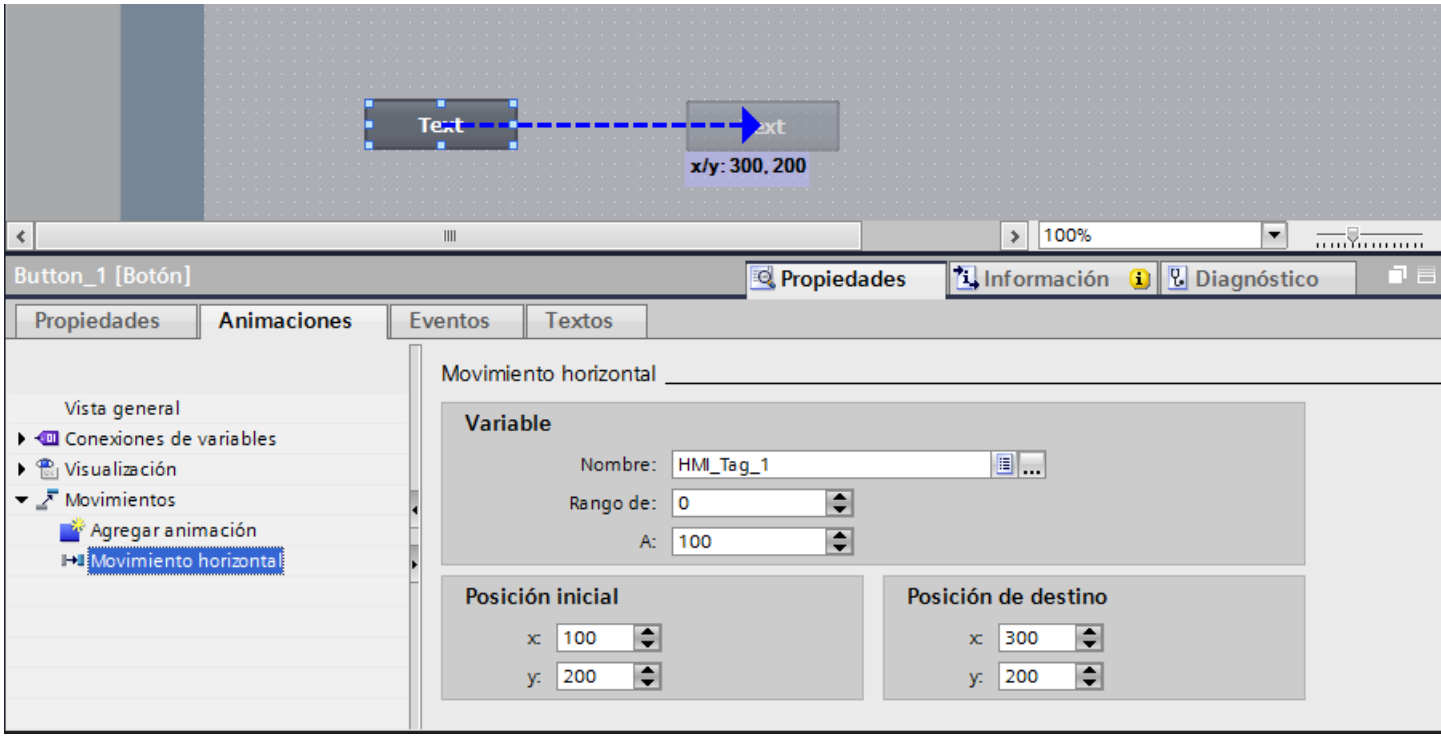
Sólo se puede configurar un tipo de movimiento por objeto.

Requisitos

- Se ha creado una variable.
- Hay una imagen abierta con un objeto dinamizable, como mínimo.
- La ventana de inspección está abierta.
- Se visualiza la ventana de herramientas.

Procedimiento

1. Seleccione en la imagen el objeto que desea dinamizar.
   Las propiedades del objeto se visualizan en la ventana de inspección.
2. Haga clic en la ventana de inspección "Propiedades > Animaciones".
   Aparecerán las animaciones disponibles para el objeto seleccionado.
3. Seleccione "Movimiento horizontal" y haga clic en el botón .
   Se visualizarán los parámetros de la animación.
   En el área de trabajo aparecerá una copia transparente del objeto unida al objeto original mediante una flecha.
4. Seleccione una variable para controlar el movimiento.
5. Desplace la copia del objeto hasta la posición de destino deseada. Los valores de píxel de la posición final se introducen automáticamente en la ventana de inspección.
6. En caso necesario, adapte el rango de valores de las variables.



Resultado

El objeto se desplazará en runtime cada vez que se modifique el valor de la variable utilizada para controlar el movimiento. El sentido del movimiento equivale al tipo de movimiento configurado "horizontal".

Nota

Los movimientos verticales y diagonales se configuran de la misma manera que el movimiento horizontal.

Configurar el movimiento directo

Introducción

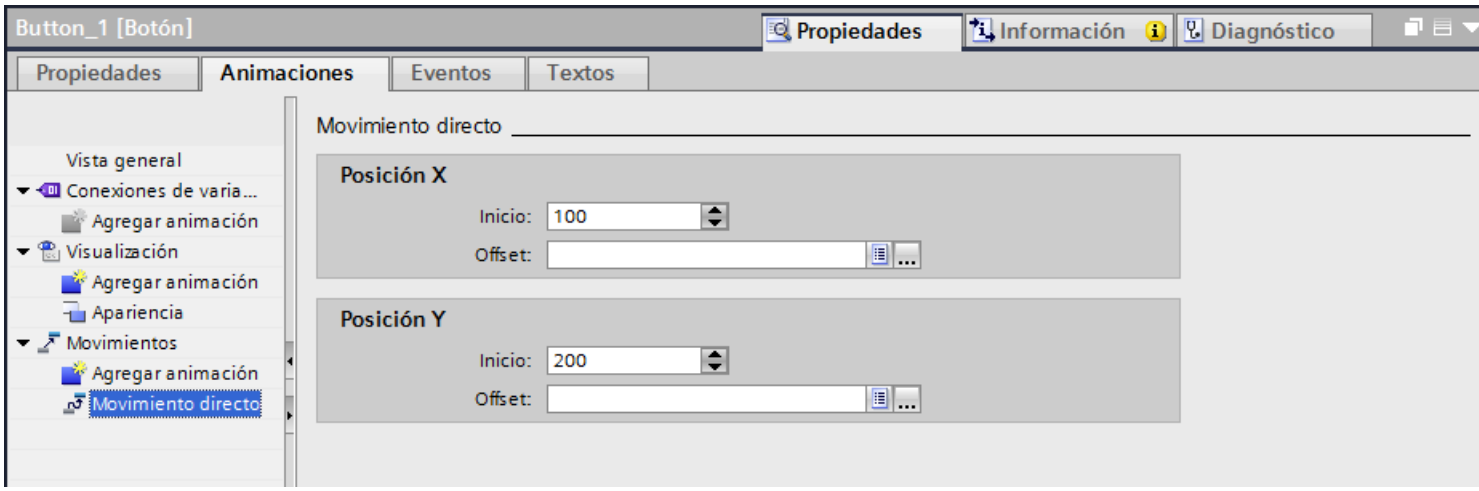
En el "Movimiento directo", el objeto se desplaza en sentido X e Y. Dos variables determinan la cantidad de píxeles por la que se mueve el objeto desde su posición inicial estática original.

Requisitos

- Se han creado dos variables.
- Hay una imagen abierta con un objeto dinamizable, como mínimo.
- La ventana de inspección está abierta.
- Se visualiza la ventana de herramientas.

Configurar el "Movimiento directo"

1. Seleccione en la imagen el objeto que desea dinamizar.
   Las propiedades del objeto se visualizan en la ventana de inspección.
2. Haga clic en la ventana de inspección "Propiedades > Animaciones".
3. Seleccione "Movimiento directo" y haga clic en el botón .
   Se visualizarán los parámetros de la animación.
4. Seleccione para la posición X una variable que controle el movimiento en sentido X.
5. Seleccione para la posición Y una variable que controle el movimiento en sentido Y.



Resultado

El objeto se desplazará en runtime cada vez que se modifique el valor de la variable utilizada para controlar el movimiento.

Dinamizar la visibilidad de un objeto

Introducción

Dinamizando la propiedad "Visibilidad" se puede p. ej. visualizar una advertencia en una imagen cuando una variable exceda un valor crítico. La advertencia se ocultará de nuevo cuando el valor de la variable deje de ser crítico.

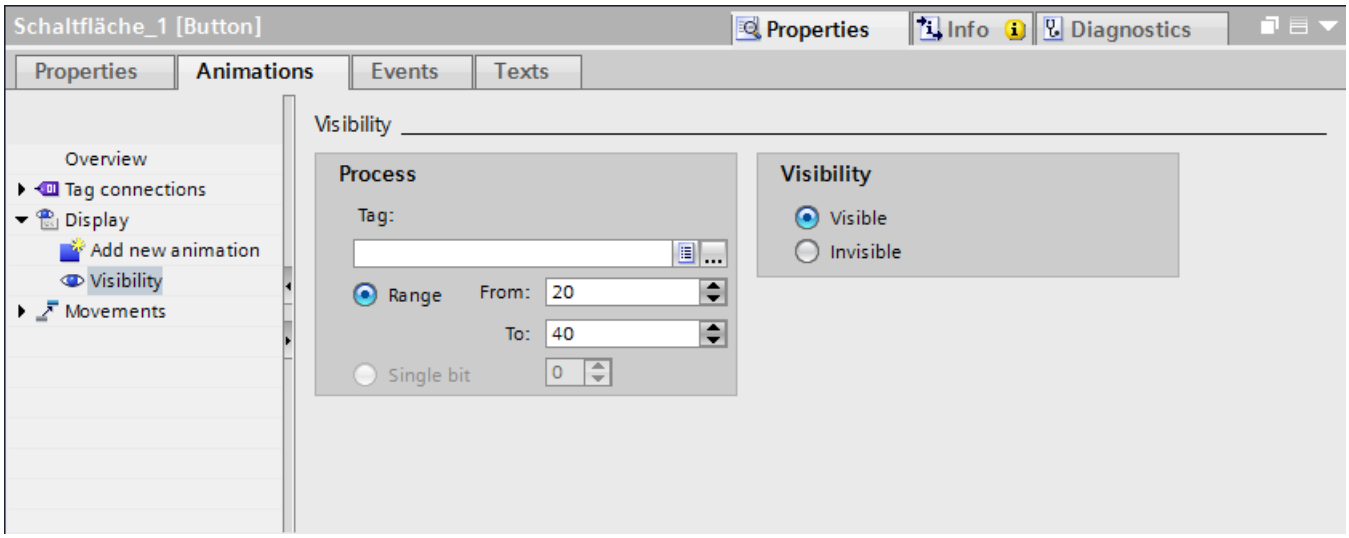
Los objetos "Visor de recetas simple" y "Visor de avisos simple" son siempre visibles.

Requisitos

- Se ha creado una variable.
- Hay una imagen abierta con un objeto que debe mostrarse u ocultarse en runtime.
- La ventana de inspección está abierta.

Procedimiento

1. Seleccione en la imagen el objeto que desea dinamizar.
   Las propiedades del objeto se visualizan en la ventana de inspección.
2. Haga clic en la ventana de inspección "Propiedades > Animaciones".
   Aparecerán las animaciones disponibles para el objeto seleccionado.
3. Seleccione "Visibilidad" y haga clic en el botón .
   Se visualizarán los parámetros de la animación.
4. Seleccione una variable.
5. Active "Rango".
6. Seleccione, p. ej., "de 20" y "a 40".
7. Active "Visible".



Resultado

El objeto de imagen se muestra u oculta en runtime en función del valor de la variable:

- Si el valor de variable coincide con el rango configurado de 20 a 40, se mostrará el objeto de imagen.
- Si el valor de la variable está fuera del rango configurado, el objeto de imagen se ocultará.

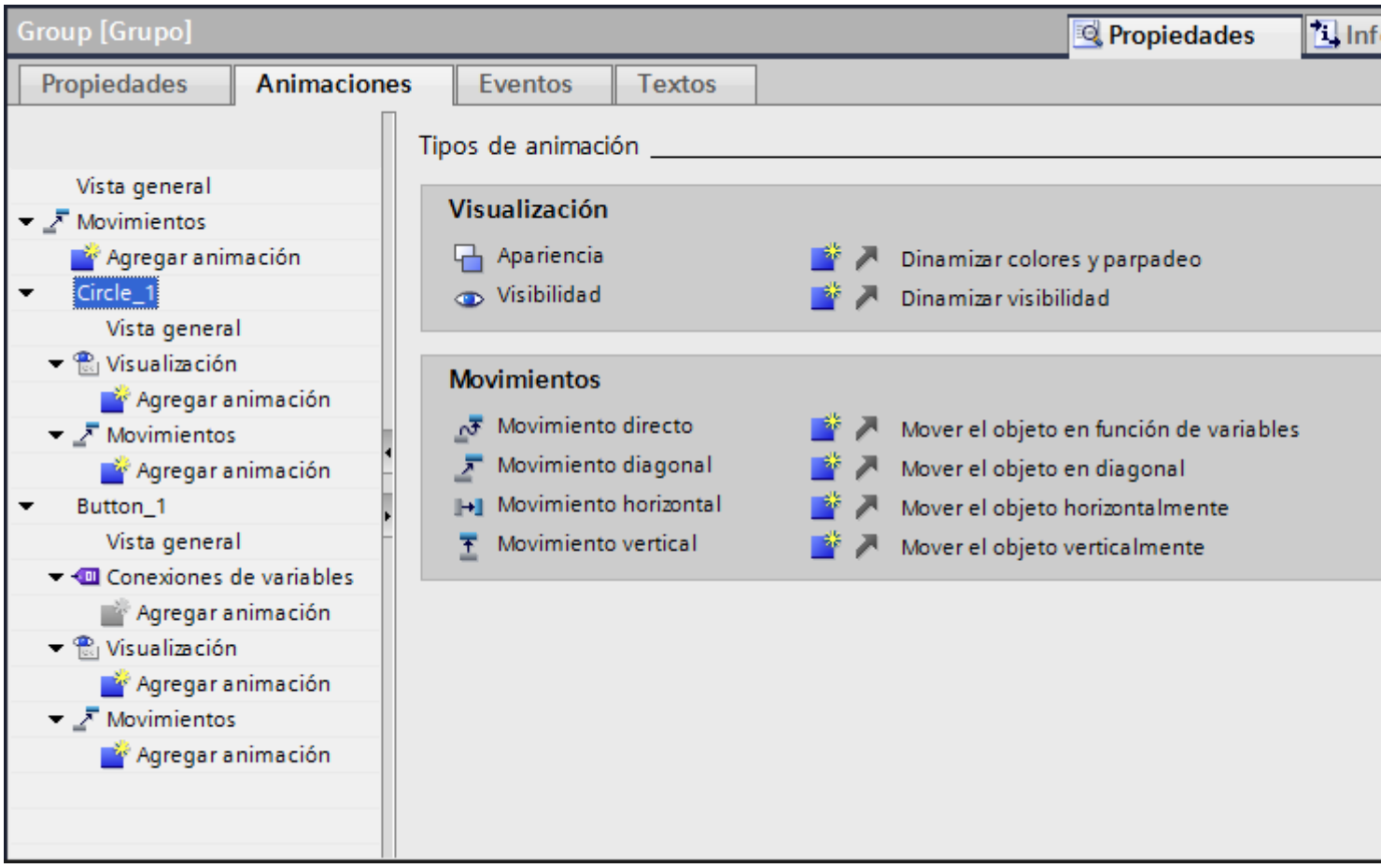
Consulte también

Configurar una animación nueva (Página 8393)

Animaciones de grupos de objetos

Aplicar animaciones a grupos de objetos

La ventana de inspección muestra todos los objetos de un grupo y sus posibles animaciones. Además, se indican tipos de animaciones por separado que son asistidas por todos los objetos del grupo.



Si se configura una animación para un grupo de objetos, esta animación será válida para todos los objetos individuales que asistan dicha animación.

Ejemplo de aplicación

Para el objeto de un grupo de objetos se configura la animación "Movimiento horizontal". Para todo el grupo de objetos se configura la animación "Movimiento directo". En runtime sólo se anima el grupo de objetos, es decir, "Movimiento directo". Ello también se aplica a grupos de objetos dentro de grupos de objetos. Sólo se anima el grupo del nivel superior.

Animaciones en grupos de objetos y en la selección múltiple

Modificar las animaciones de varios objetos

En caso de selección múltiple, en la ventana de inspección se mostrarán las animaciones configuradas para el objeto de referencia. Las animaciones se modifican del modo habitual. Los cambios afectan a todos los objetos de la selección múltiple que admitan la animación configurada. De este modo, la animación del objeto de referencia se aplica también a objetos que hasta el momento no tenían ninguna animación configurada.

Ejemplo de aplicación

Seleccione al mismo tiempo un botón y un círculo. El botón es el objeto de referencia. El botón ya tiene configurada la animación "Apariencia" y, por tanto, aparecerá en la ventana de inspección de la selección múltiple. Si en la ventana de inspección se activa "Propiedades > Animaciones > Apariencia > Parpadeo", la configuración de la animación "Apariencia" se aplica al botón y al círculo.

Configurar animaciones nuevas para varios objetos

Si se configura una animación nueva para los objetos de una selección múltiple, esta animación será válida para todos los objetos seleccionados que admitan la animación configurada. Si la animación nueva sustituyera a una animación existente, se mostraría una consulta de seguridad.

Ejemplo de aplicación

Seleccione un círculo y un rectángulo. El círculo ya tiene configurada la animación "Movimiento diagonal". Configure la animación "Movimiento horizontal" para la selección múltiple. La animación se aplica al rectángulo, ya que todavía no tenía configurada ninguna animación del tipo movimiento. Para el círculo se le preguntará si desea sustituir la animación existente "Movimiento diagonal" por la animación nueva "Movimiento horizontal".

10.1.4.4 Dinamizar con funciones de sistema

Principios básicos de los eventos

Introducción

Los objetos de imagen reaccionan a los eventos. Dentro de los eventos de un objeto se configura una lista de función con funciones de sistema.

Eventos

Los eventos y funciones de sistema disponibles dependen del objeto utilizado.

Cuando el operador, p. ej., active un objeto de imagen, la función de sistema configurada se ejecuta.

Encontrará más información al respecto en "Principios básicos de la lista de funciones (Página 8638)".

Consulte también

Principios básicos de la dinamización (Página 8391)

Ejemplo: Configurar un botón para el cambio de idioma (Página 8405)

Configurar función de sistema en el evento "Hacer clic" (Página 8404)

Configurar función de sistema en el evento "Hacer clic"

Introducción

Se configura una lista de funciones dentro de un evento de un objeto. Cuando el evento tenga lugar en runtime, se ejecuta la función de sistema enlazada.

Requisitos

Hay una imagen abierta.

Se ha creado un botón en la imagen.

La ventana de inspección está abierta.

Procedimiento

1. Seleccione el botón.
2. Haga clic en la ventana de inspección "Propiedades > Eventos".
3. Seleccione el evento "Hacer clic".

4. Haga clic en "Agregar función" en la tabla
5. Seleccione la función de sistema "MostrarVentanaDeAvisos".

Resultado

Cuando el usuario haga clic en este botón en runtime, se abrirá la ventana de avisos en la imagen.

Consulte también

Principios básicos de los eventos (Página 8402)

Ejemplo: Configurar un botón para el cambio de idioma

Introducción

En este ejemplo se configurará un botón con el que podrá cambiar entre varios idiomas en runtime.

Requisitos

- Debe haber realizado el ejemplo "Configurar la etiqueta multilingüe de un botón".
- La imagen "Screen_1" está abierta.
- El botón se selecciona en la imagen.

Procedimiento

1. En la ventana de inspección haga clic en "Propiedades > Eventos > Pulsar".
2. En la tabla haga clic en "Agregar función".
3. Seleccione la función de sistema "EstablecerIdioma" y el ajuste "Conmutar".

Resultado

Se ha asignado la función "AjustarIdioma" al botón. Si pulsa el botón en runtime, se cambia el idioma de runtime. La secuencia de los idiomas de runtime se aplica tal como se ha predeterminado en el editor "Idioma y fuente" mediante la numeración de los idiomas.

Consulte también

Principios básicos de los eventos (Página 8402)

Botón (Página 8450)

10.1.5 Trabajar con teclas de función

10.1.5.1 Trabajar con teclas de función

Introducción

Una tecla de función es una tecla del panel de operador con funcionalidad configurable. A los eventos "Tecla pulsada" y "Soltar tecla" se puede asignar una lista de funciones.

Las teclas de función se pueden asignar de forma global o local.

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

Teclas de función globales

Las teclas de función con asignación global disparan siempre la misma acción, independientemente de la imagen visualizada.

Las teclas de función globales se configuran en el editor "Imagen general". La asignación global se aplicará a todas las imágenes del panel de operador configurado.

Las teclas de función globales permiten reducir en gran medida el esfuerzo de configuración, puesto que no es necesario asignarles funciones en cada imagen.

Teclas de función locales en imágenes

Las teclas de función locales de las imágenes pueden disparar una acción diferente en cada imagen. La asignación vale sólo para la imagen en la que se ha definido la tecla de función.

Dentro de una imagen, una tecla de función solo tiene asignada una función, ya sea global o local. Es el ingeniero quien determina la prioridad de las asignaciones.

Nota

Si un visor de avisos o una ventana de avisos oculta una imagen que incorpore teclas de función locales, éstas estarán activas en runtime de todas maneras. Ello puede ocurrir sobre todo en los paneles de operador equipados con un display pequeño.

Teclas de función locales en plantillas

Las teclas de función locales que están asignadas en plantillas se aplican a todas las imágenes que se basan en dicha plantilla. Pueden disparar una acción distinta en cada imagen. Las teclas de función de las plantillas se asignan en la plantilla en el editor "Imágenes". Dentro de una plantilla, cada tecla de función solo tiene asignada una función, ya sea global o local. Es el ingeniero quien determina la prioridad de las asignaciones.

Asignación con hotkeys

Es posible asignar hotkeys a los objetos de manejo (p. ej. botones). Los hotkeys disponibles dependen del panel de operador utilizado.

Nota

La tecla de función está asignada a una acción local o global. Si, además, se asigna un hotkey a la tecla de función, en runtime se ejecutará la función del hotkey.

Gráficos

Cuando una tecla de función se encuentra directamente junto al display, se le puede asignar un gráfico que describa la función de dicha tecla.

Vista de la asignación

Tabla 10-1 La tabla siguiente muestra qué símbolos visualizan la asignación de las teclas de función:

Tecla de función	Descripción
F1	No asignada
F2	Asignada globalmente
F6	Asignada localmente en la plantilla
F3	Asignada localmente
F8	Asignada localmente (la asignación local de la plantilla sobrescribe la asignación global)
F4	Asignada localmente (la asignación local sobrescribe la asignación global)
F6	Asignada localmente (la asignación local sobrescribe la asignación local de la plantilla)
F8 F5	Asignada localmente (la asignación local sobrescribe la asignación local de la plantilla, que ya sobrescribe la asignación global)
	Navegación de imágenes asignada al botón

Nota

Basic Panels

El editor "Navegación de imágenes" no está disponible para Basic Panels.

10.1.5.2 Asignar teclas de función globalmente

Introducción

La asignación global de una tecla de función se define en el editor "Imagen general". La asignación global se aplica a todas las imágenes del panel de operador configurado.

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

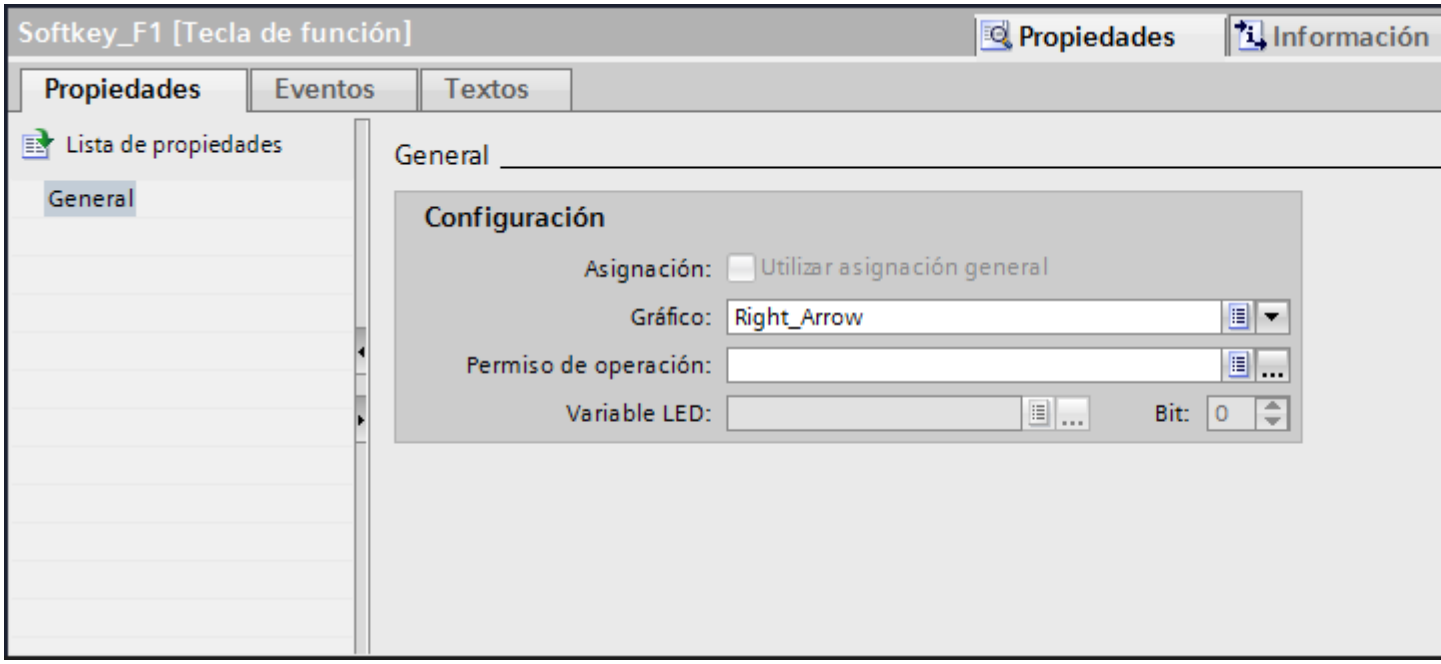
Requisitos

- El proyecto está abierto.
- La ventana de inspección está abierta.

Procedimiento

Para asignar una función independiente de la imagen a una tecla de función, proceda del siguiente modo:

1. Para abrir el editor "Imagen general", haga doble clic en "Imagen general" del grupo "Administración de imágenes de la ventana del proyecto.
2. Seleccione la tecla de función deseada.
   Las propiedades de la tecla de función se visualizarán en la ventana de inspección.



3. En "General", configure un gráfico y una autorización de uso para la tecla de función.
4. Asocie una lista de funciones al evento deseado en "Eventos".

Resultado

Si ninguna asignación local sobrescribe la asignación global, en todas las imágenes del panel de operador configurado se modificará la asignación de la tecla de función en correspondencia con la entrada realizada.

10.1.5.3 Asignar teclas de función localmente

Introducción

Las teclas de función se asignan global y localmente. Una asignación local de las teclas de función sólo se aplica a la imagen o plantilla en la que se ha definido. Existen las siguientes teclas de función locales:

- Teclas de función locales de una imagen
  Para cada imagen se asignan funciones distintas a la tecla de función. La asignación vale sólo para la imagen en la que se ha definido la tecla de función.
- Teclas de función locales de una plantilla
  Las teclas de función se asignan en una plantilla. La asignación se aplica a todas las imágenes que se basan en dicha plantilla y que no quedan sobrescritas por una asignación local en una imagen.

Una asignación local sobrescribe la asignación global de una tecla de función.

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

Uso de asignaciones existentes

La opción para utilizar asignaciones existentes se denomina del modo siguiente en la ventana de inspección:

- En una plantilla: "Utilizar asignación general"
- En una imagen:
  - Si la imagen se basa en una plantilla: "Utilizar plantilla local"
  - Si la imagen no se basa en una plantilla: "Utilizar asignación general"

La opción "Utilizar plantilla local" incluye el uso de la asignación local en la plantilla y la asignación global.

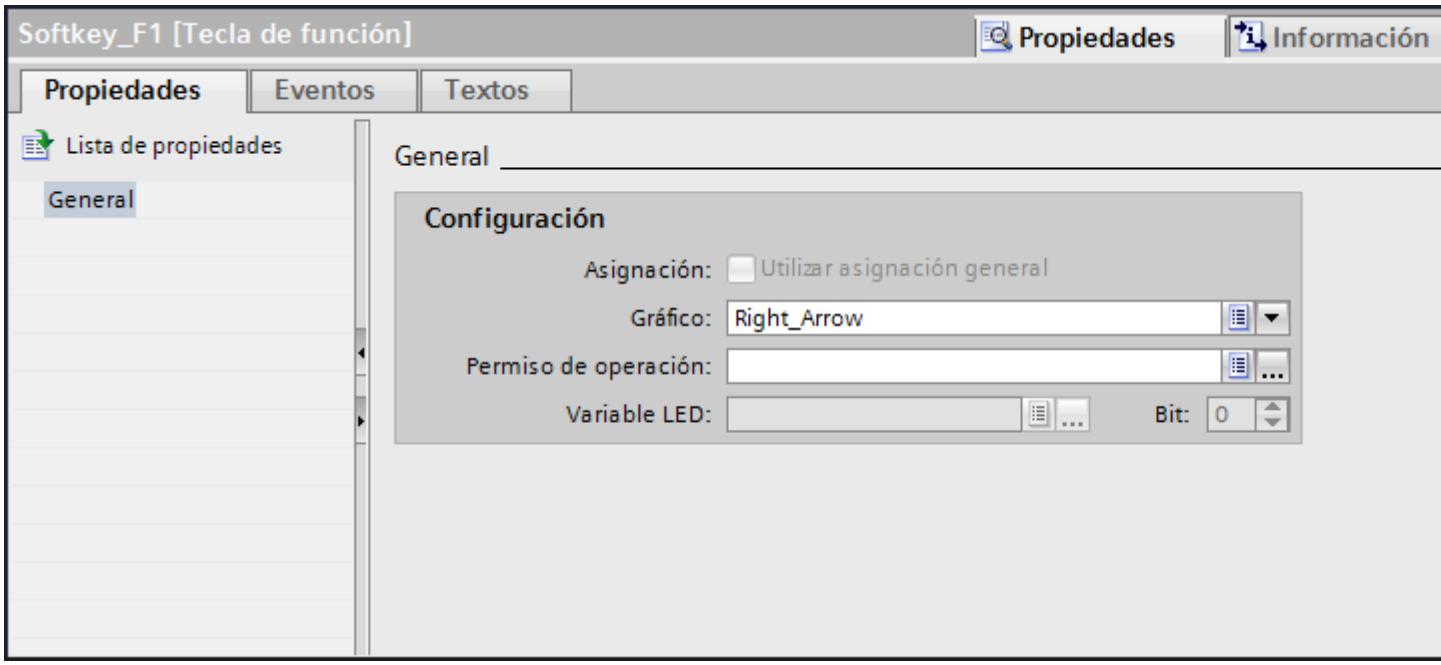
Requisitos

- Está abierta la imagen o la plantilla a la que desea asignar localmente una tecla de función.
- La ventana de inspección está abierta.

Procedimiento

Proceda del siguiente modo:

1. Seleccione la tecla de función deseada en la imagen o plantilla.
   Las propiedades de la tecla de función se visualizarán en la ventana de inspección.
2. En la ventana de inspección haga clic en "General".



3. Desactive la opción "Utilizar plantilla local" o "Utilizar asignación general".
4. En "General", configure un gráfico y una autorización de uso para la tecla de función.
5. Asocie una lista de funciones al evento deseado en "Eventos".

Resultado

La imagen o plantilla tiene asignada la tecla de función con las funciones configuradas.

10.1.5.4 Asignar una función a una tecla de función

Introducción

Una tecla de función puede tener dos estados, a saber:

- Pulsado: definido por el evento "Tecla pulsada".
- Soltado: definido por el evento "Soltar tecla".

Ambos eventos se configuran en la ventana de inspección de la tecla de función. A cada evento se puede asignar una lista de funciones que contenga las funciones de sistema o los scripts. La lista de funciones se procesará en runtime cuando se presente el evento.

Nota
Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

Nota
Basic Panels

Los scripts no están disponibles para Basic Panels.

Requisitos

Si desea asignar una tecla de función globalmente:

- El editor "Imagen general" está abierto.

Si desea asignar una tecla de función localmente:

- La imagen en la que desea asignar una tecla de función está abierta.

Si desea asignar una tecla de función localmente en una plantilla:

- La plantilla en la que desea asignar una tecla de función está abierta.
- La ventana de inspección está abierta.

Procedimiento

Proceda del siguiente modo:

1. Seleccione la tecla de función que desea definir.
   Las propiedades de la tecla de función se visualizarán en la ventana de inspección.
2. En el grupo "General" de "Propiedades" de la ventana de inspección, asigne una lista de funciones al evento deseado.

Resultado

La lista de funciones se procesará en runtime cuando el operador pulse o suelte la tecla de función.

10.1.5.5 Autorizar el uso de una tecla de función

Introducción

WinCC permite asignar una autorización de uso de una tecla de función en runtime. Gracias a ello, al crear el proyecto podrá limitar el manejo de las teclas de función a determinadas

personas o grupos de usuarios. Sólo el personal autorizado podrá modificar en runtime los parámetros y ajustes importantes.

Configurando una protección de acceso se puede impedir un manejo incorrecto y aumentar así la seguridad de la instalación o la máquina.

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

Requisitos

- Los grupos de usuario se deberán haber definido.

Si desea proteger una tecla de función global:

- El editor "Imagen general" está abierto.

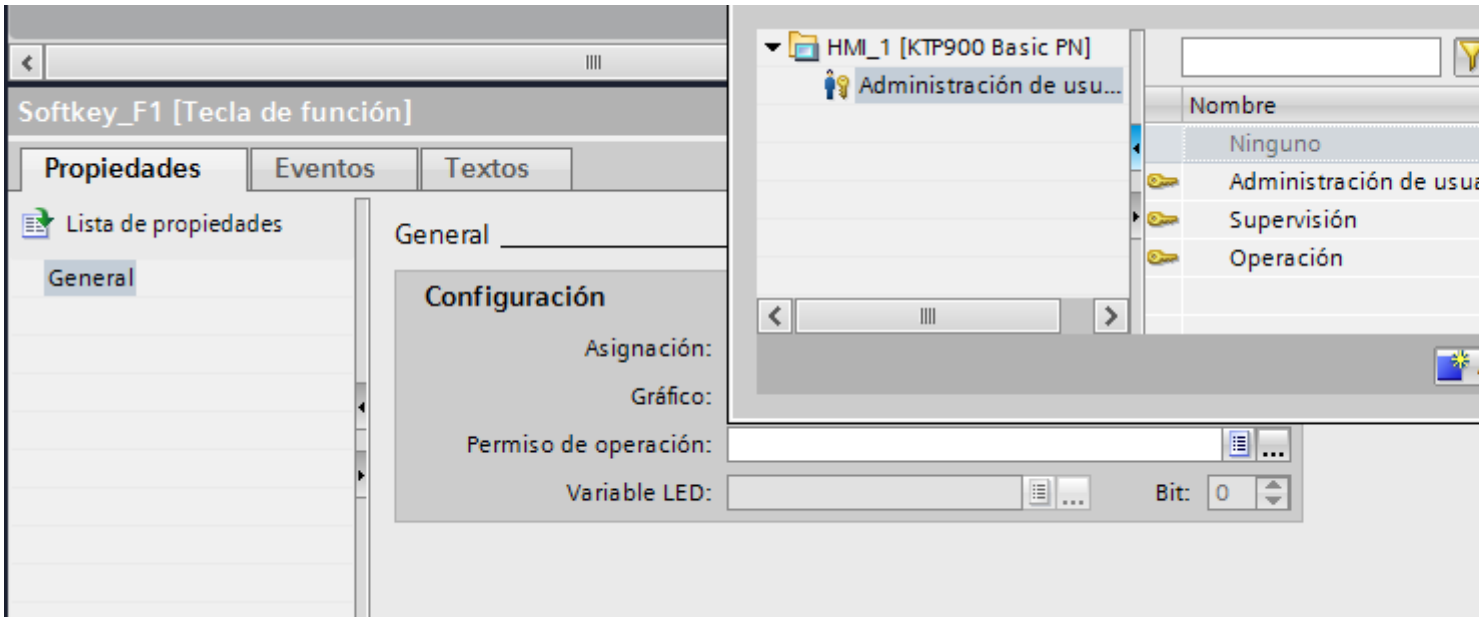
Si desea proteger una tecla de función local de una imagen o plantilla:

- La imagen o plantilla con la tecla de función está abierta.
- La ventana de inspección está abierta.

Procedimiento

Proceda del siguiente modo:

1. Seleccione la tecla de función deseada.
   Las propiedades de la tecla de función se visualizarán en la ventana de inspección.
2. En la ventana de inspección haga clic en "General".



3. En la lista "Autorización", seleccione el grupo de usuarios que debe poder usar la tecla de función en runtime.

Resultado

La autorización de manejo quedará configurada.

10.1.5.6 Asignar un gráfico a una tecla de función

Introducción

Para describir la función de una tecla se puede insertar un gráfico en una imagen junto a la tecla de función. Los gráficos se asignan sólo a las teclas de función que limiten con la pantalla del panel de operador.

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

Requisitos

Si desea asignar un gráfico a una tecla de función global:

- El editor "Imagen general" está abierto.

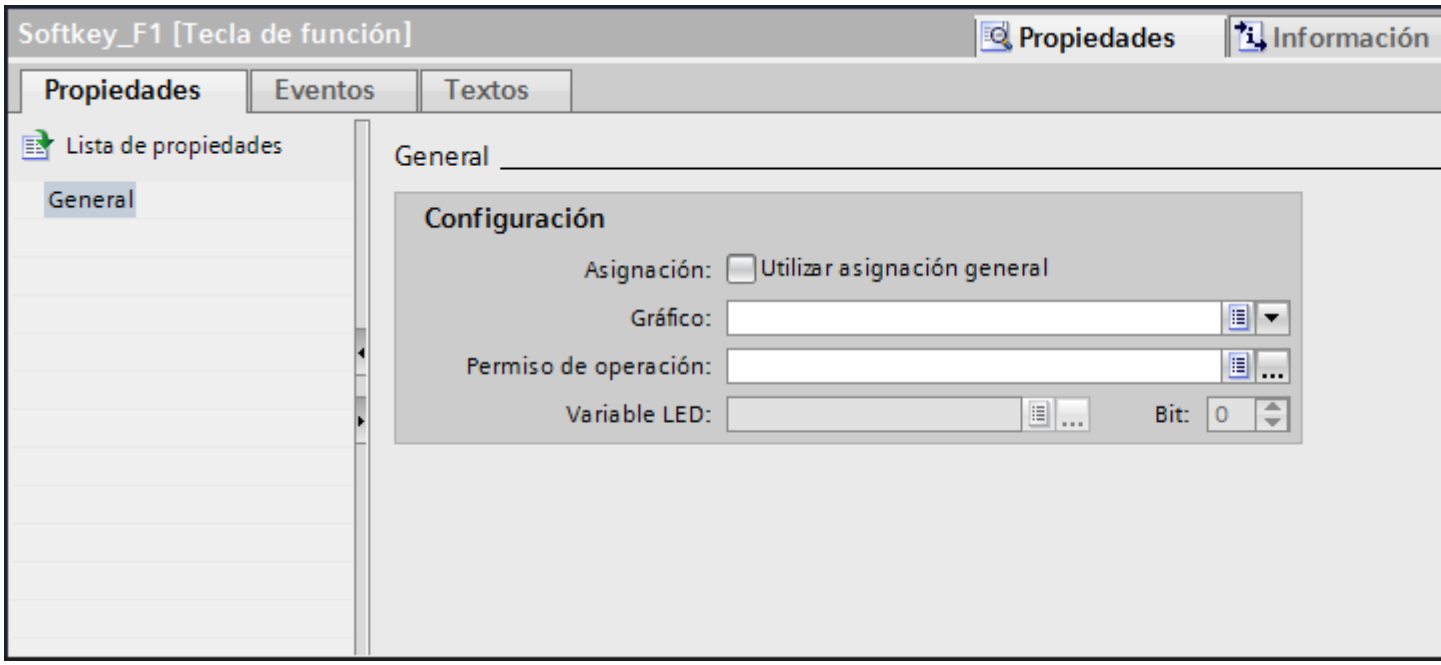
Si desea asignar un gráfico a una tecla de función local en una imagen o plantilla:

- La imagen o plantilla con la tecla de función en cuestión está abierta.
- La ventana de inspección está abierta.
- El gráfico destinado a la tecla de función se deberá haber creado.

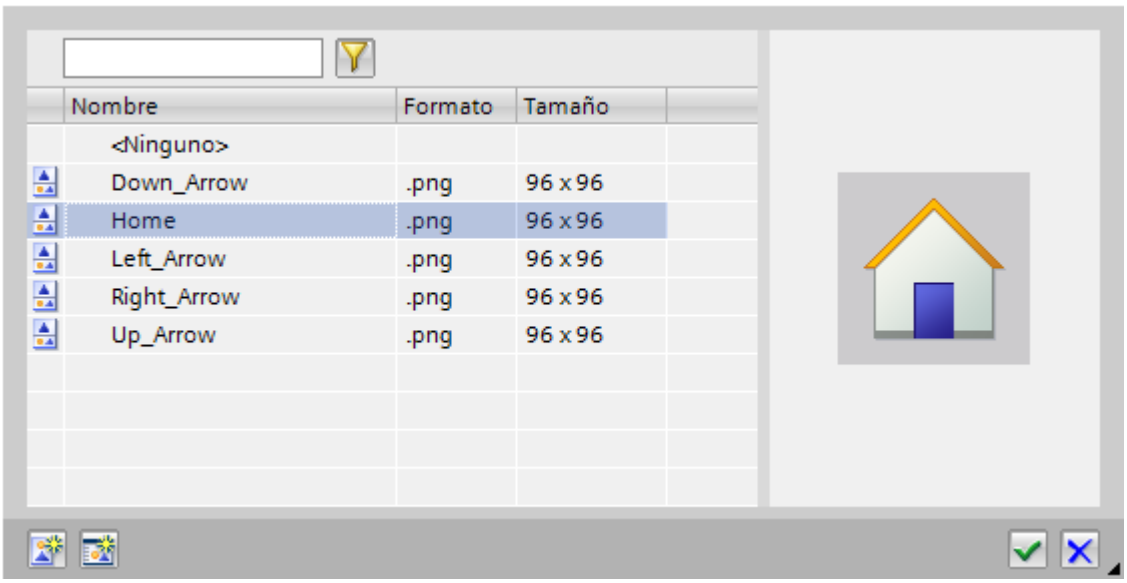
Procedimiento

Proceda del siguiente modo:

1. Seleccione la tecla de función deseada.
   Las propiedades de la tecla de función se visualizarán en la ventana de inspección.
2. En la ventana de inspección haga clic en "General".



3. Haga clic en la lista bajo "Gráfico".
   Aparecerá la colección de gráficos del proyecto de WinCC flexible. A la izquierda se visualizan los gráficos externos que ya se han agregado al juego de gráficos. En el lado derecho aparece una vista preliminar del gráfico seleccionado en la colección.



   Con los símbolos y , la colección se visualiza en forma de miniaturas o de lista. Para abrir y editar un objeto OLE en el programa gráfico correspondiente, haga doble clic en el objeto en cuestión.

4. En la colección, haga clic en el gráfico deseado, o bien deposite el gráfico en la colección. El gráfico se visualizará a la derecha en la vista preliminar.
5. Para insertar el gráfico en la imagen haga clic en "Seleccionar".
   Haciendo clic en "Deseleccionar" podrá quitar un gráfico de la imagen.

Resultado

El gráfico se muestra junto a la tecla de función.

10.1.5.7 Configurar variables LED

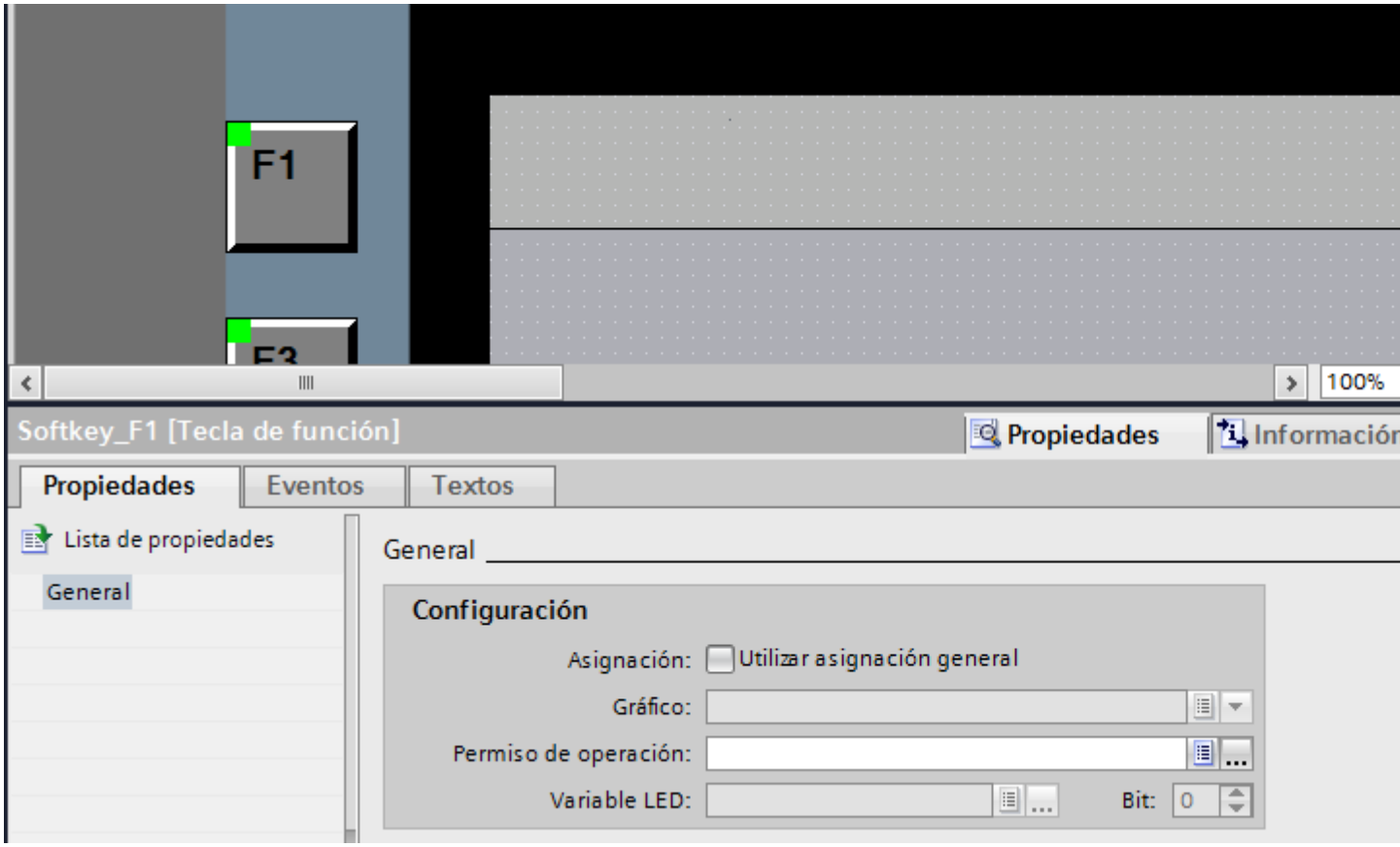
Requisitos

- Se ha creado un panel de operador con teclado.
- Se ha creado una variable LED.

Procedimiento

1. Cree una imagen nueva.
2. Haga clic en una tecla F del panel de operador.

3. Haga clic en "General" en "Propiedades > Propiedades" de la ventana de inspección.



4. En el área "General > Configuración", seleccione una variable bajo "Variable LED".
5. Introduzca el número de bit adecuado en "Bit".
   El número de bit adecuado depende del panel de operador y de la ocupación de las entradas y salidas del panel de operador.

Ocupación de las entradas y salidas

La ocupación exacta de las entradas y salidas la encontrará en:

- Teclas directas PROFINET IO: Auto-Hotspot
- Teclas directas PROFIBUS DP: Auto-Hotspot

10.1.5.8 Ejemplo: Utilizar teclas de función para navegar por imágenes

Tarea

En el presente ejemplo, una tecla de función se asigna localmente en una imagen. Cuando el operador pulse esa tecla de función, se disparará un cambio a una imagen predefinida denominada, p. ej., "Caldera2".

Nota

Dependencia del panel de operador

Las teclas de función no están disponibles en todos los paneles de operador.

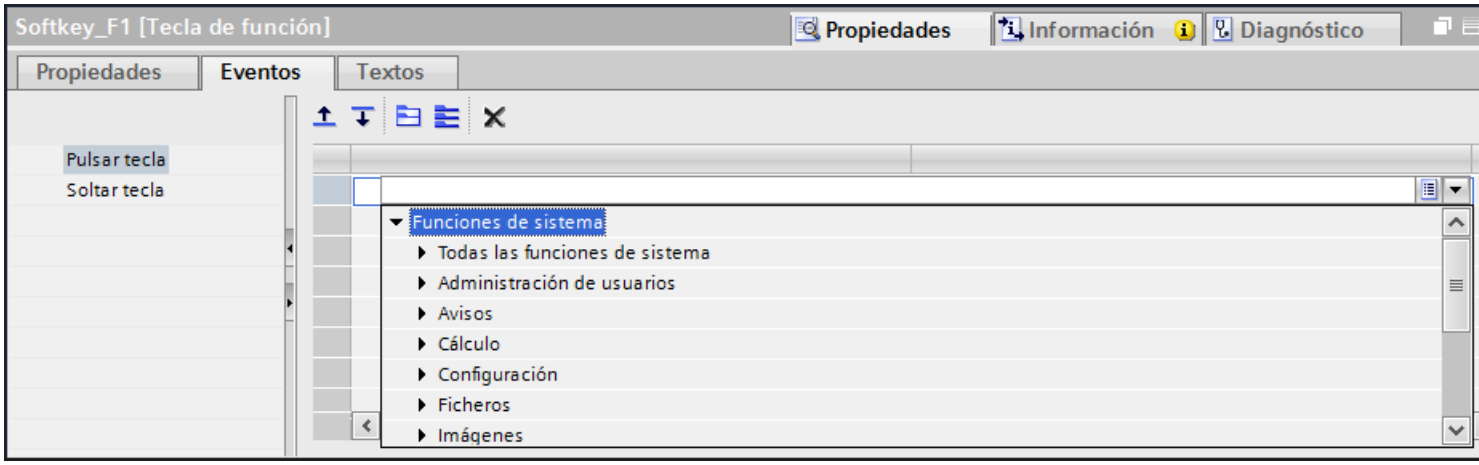
Requisitos

- La imagen en la que desea asignar la tecla de función está abierta.
- La imagen "Caldera2" se deberá haber creado.
- La ventana de inspección está abierta.

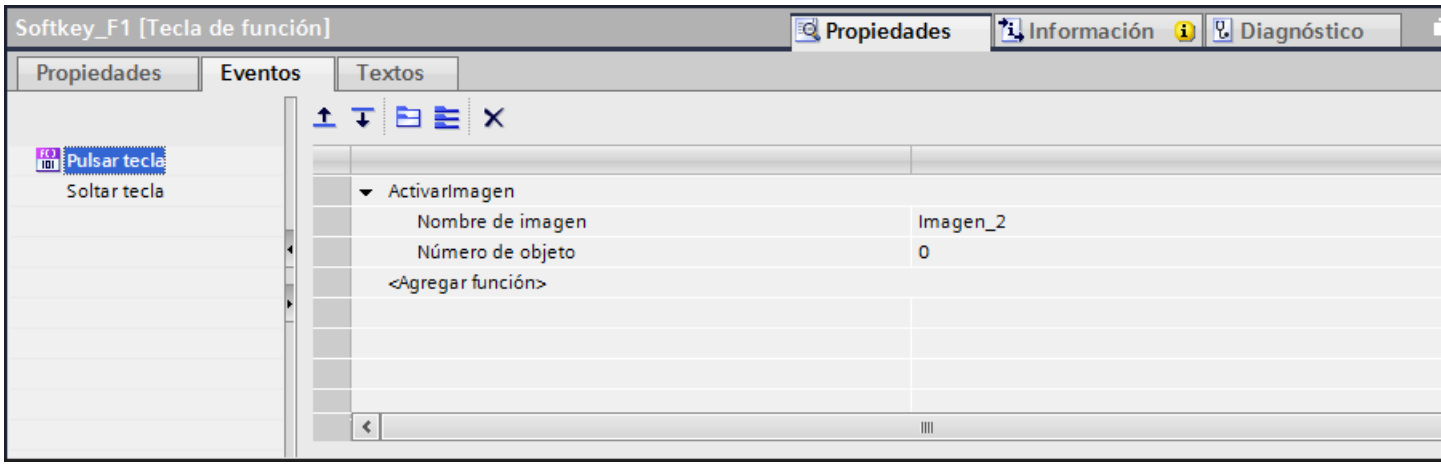
Procedimiento

Para utilizar la función "ActivarImagen", proceda del siguiente modo:

1. Seleccione la tecla de función deseada.
   Las propiedades de la tecla de función se visualizarán en la ventana de inspección.
2. Haga clic en "General".
3. Para sobrescribir una asignación global, desactive la opción "Utilizar plantilla local".
4. En "Eventos", haga clic en "Tecla pulsada".



5. Seleccione la función de sistema "ActivarImagen" en la lista.
   La función "ActivarImagen" aparecerá con los parámetros "Nombre de imagen" y "Número de objeto" en el grupo "Lista de funciones".



6. En la lista, seleccione el nombre de imagen "Caldera2".

Resultado

Cuando el usuario pulse en runtime la tecla de función seleccionada, se cambiará a la imagen "Caldera 2".

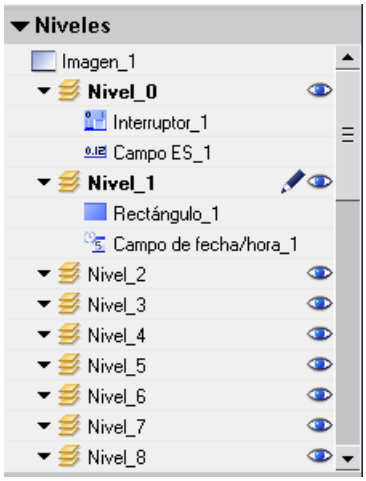
10.1.6 Trabajar con niveles

10.1.6.1 Principios básicos para trabajar con niveles

Niveles

Los niveles se utilizan para editar los objetos de una imagen de forma diferenciada. Una imagen tiene 32 niveles a los que puede poner el nombre que quiera. Asignando objetos a los niveles se determina el escalonamiento de la imagen. Los objetos del nivel 0 se encuentran

en el último plano de la imagen, en tanto que los objetos del nivel 31 se encuentran en primer plano.



Los objetos de un mismo nivel también están organizados de forma jerárquica. Cuando se crea una imagen, el primer objeto que se ha insertado se dispone atrás en el nivel. Cada objeto adicional se insertará una posición más adelante. Dentro de un nivel es posible desplazar los objetos hacia adelante y hacia atrás.

Principio de los niveles

Uno de los 32 niveles está activo siempre. Al insertar objetos en una imagen, éstos se asignarán por defecto al nivel activo. Los números de los niveles activos se muestran en la ventana de inspección de la imagen y en la Task Card "Diseño > Niveles".

Al abrir una imagen se visualizarán siempre los 32 niveles de la misma. En la ventana de inspección de la imagen y en la Task Card "Diseño > Niveles" se pueden ocultar todos los niveles excepto el nivel activo. Ello permite editar concretamente los objetos del nivel activo.

En la vista de árbol de la paleta "Niveles", en la Task Card "Diseño", los niveles y los objetos se administran con la función de arrastrar y soltar y con el menú contextual.

Ejemplos de aplicación

Los niveles se utilizan p. ej. en los casos siguientes:

- Para ocultar la rotulación de objetos durante la edición
- Para ocultar objetos, p. ej., ventanas de avisos, mientras se configuran otros objetos

10.1.6.2 Desplazar objetos a niveles

Introducción

De forma predeterminada, los objetos recién insertados están en el nivel activo. Un objeto se puede asignar a otro nivel con posterioridad.

Requisitos

- Hay una imagen abierta con un objeto.
- La ventana de inspección está abierta.

Procedimiento

1. Seleccione el objeto en la imagen.
   Las propiedades del objeto se visualizan en la ventana de inspección.
2. En la ventana de inspección "Propiedades > Propiedades > Misceláneo > Nivel", indique el nivel al que desea desplazar el objeto.

Otra posibilidad es seleccionar el objeto en la Task Card "Diseño" y arrastrarlo al nivel deseado con la función de arrastrar y soltar.

Modificar el orden de los objetos

1. Seleccione el objeto en la imagen.
   Las propiedades del objeto se visualizan en la ventana de inspección.
2. Para desplazar el objeto hacia delante o detrás, seleccione el comando "Orden > Uno atrás" o "Uno adelante" del menú contextual.
   También puede utilizar los botones o de la barra de herramientas.

Resultado

El objeto se asignará al nivel seleccionado y se colocará en primer plano.

10.1.6.3 Configurar el nivel activo

Introducción

Los objetos de una imagen están asignados siempre a uno de 32 niveles. En la imagen hay siempre un nivel activo. Al insertar objetos en una imagen, éstos se asignarán por defecto al nivel activo.

El número del nivel activo se visualiza en la barra de herramientas "Nivel". El nivel activo se marca con el símbolo en la Task Card "Diseño > Niveles".

Al comenzar la configuración, el nivel 0 es el nivel activo. Durante la configuración es posible activar un nivel diferente en caso necesario.

Requisitos

- Hay como mínimo una imagen abierta con un objeto.
- La ventana de inspección de la imagen actual está abierta.

Procedimiento

1. Haga clic en la ventana de inspección de la imagen actual "Propiedades > Propiedades > Niveles".
2. En "Configuración > Nivel activo" introduzca el número del nivel.

Procedimiento alternativo

1. Seleccione la Task Card "Diseño > Niveles".
2. En el menú contextual, elija el comando "Ajustar a activado".

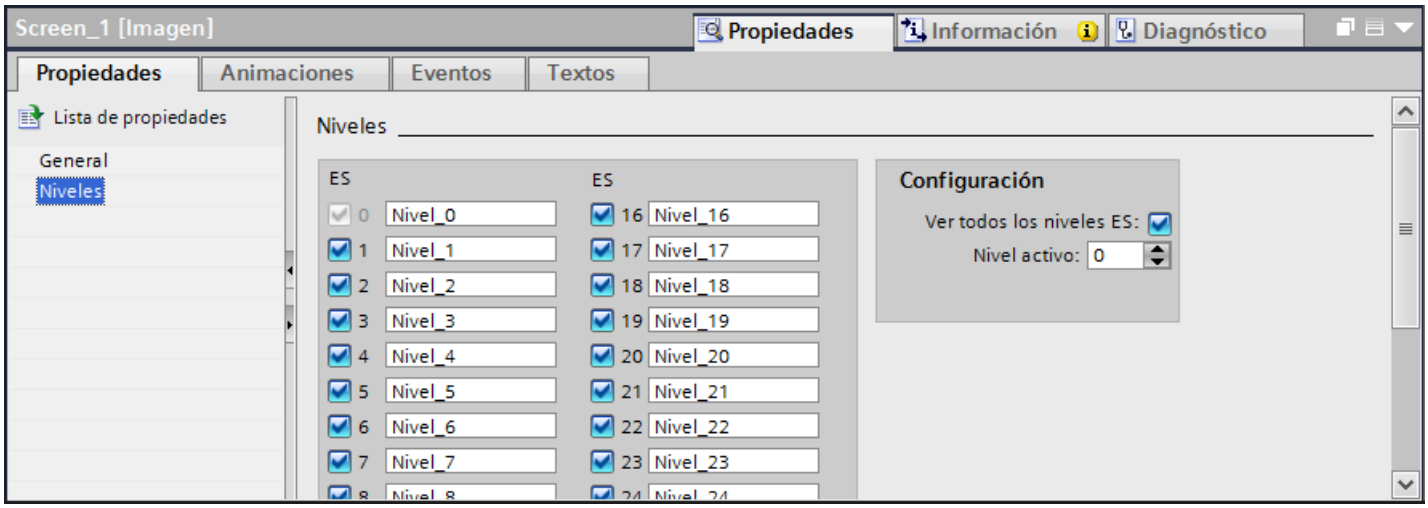
Resultado

El nivel con el número indicado se convertirá en el nivel activo.

10.1.6.4 Mostrar y ocultar niveles

Introducción

Los niveles de una imagen se muestran u ocultan según se precise. Defina qué niveles se mostrarán en el sistema de ingeniería. Al abrir una imagen se muestran siempre todos los niveles.



Requisitos

- La imagen está abierta.
- La Task Card "Diseño" está abierta.

Procedimiento

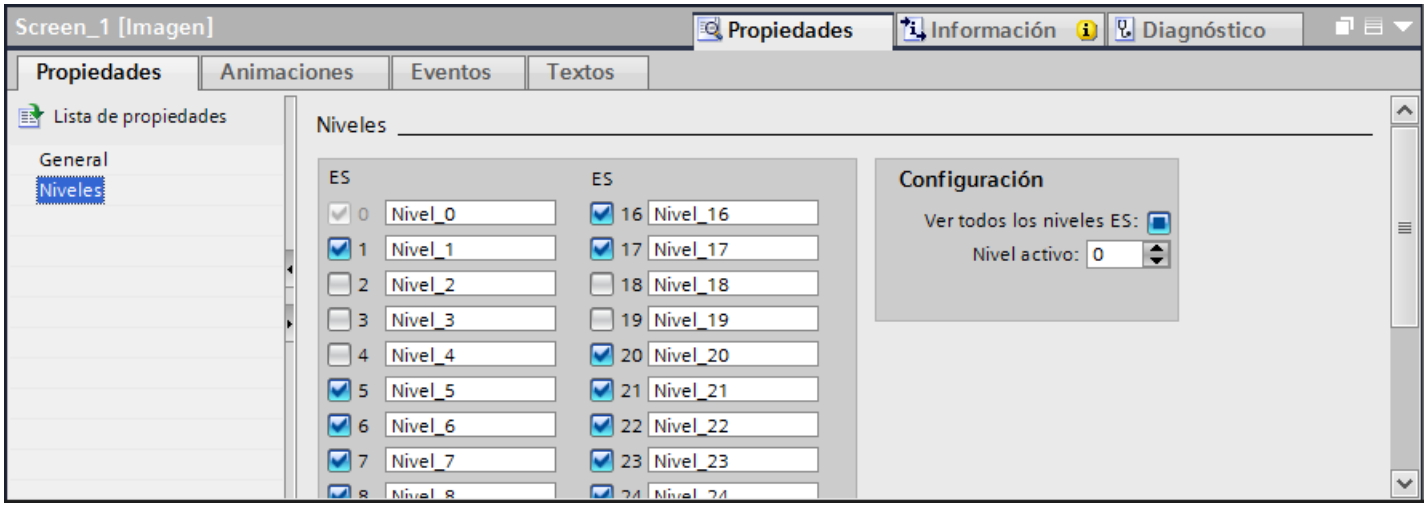
1. En la Task Card "Diseño > Niveles" seleccione el nivel que desea mostrar u ocultar.
2. Haga clic en uno de los símbolos situados junto al nivel correspondiente:
   – el nivel mostrado se ocultará
   – el nivel ocultado se mostrará

   Nota

   El nivel activo no se puede ocultar.

Procedimiento alternativo

1. En la imagen, haga clic en una posición en la que no exista ningún objeto.
   Las propiedades de la imagen se visualizarán en la ventana de inspección.
2. Haga clic en la ventana de inspección "Propiedades > Propiedades > Niveles".



3. Desactive en la lista los niveles que desea ocultar.
   Si se activa "Todos los niveles ES" para un nivel, los objetos de dicho nivel se mostrarán en el sistema de ingeniería.

Resultado

Los niveles se mostrarán o se ocultarán conforme a sus ajustes.

10.1.6.5 Cambiar el nombre de los niveles

Introducción

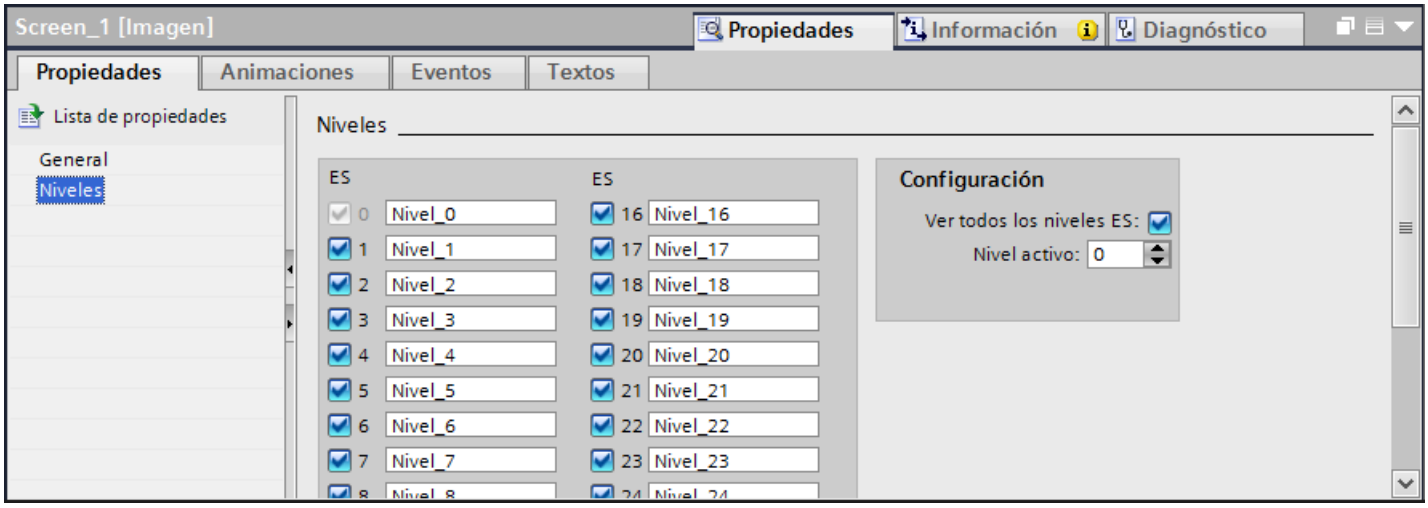
De forma estándar, al crear una imagen, los 32 niveles están numerados correlativamente. Para obtener una visión más clara, cambie el nombre de los niveles de acuerdo con sus necesidades.

Requisitos

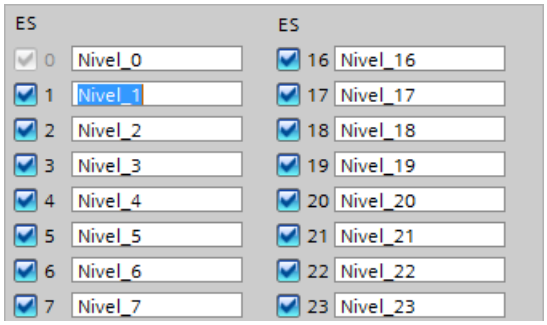
- La imagen está abierta.

Procedimiento

1. En la imagen, haga clic en una posición en la que no exista ningún objeto.
   Las propiedades de la imagen se visualizarán en la ventana de inspección.
2. En la ventana de inspección haga clic en "Propiedades > Propiedades > Niveles".



3. Introduzca el nuevo nombre del nivel.



Resultado

El nivel se mostrará con el nombre nuevo.

10.1.7 Objetos de manejo y visualización

10.1.7.1 Objetos dependientes de los paneles

Objetos para Basic Panels

Disponibilidad de los objetos de indicación y manejo para Basic Panels

En la ventana de objetos se visualizan únicamente los objetos que pueden utilizarse en el panel para el que desea configurar. La tabla siguiente muestra la disponibilidad de los objetos de indicación y manejo en los Basic Panels.

Resumen

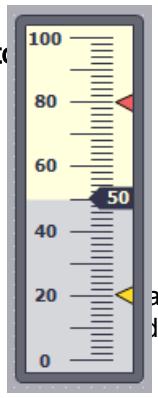
	KP300 Basic KP400 Basic	KTP400 Basic KTP600 Basic KTP1000 Basic TP1500 Basic	KTP700 Basic KTP900 Basic
Barra	sí	sí	sí
Visor de usuarios	sí	sí	sí [1)]
Campo de fecha y hora	sí	sí	sí
Campo E/S	sí	sí	sí
Elipse	sí	sí	sí
Visor de gráficos	sí	sí	sí
Campo E/S gráfico	sí	sí	sí
Indicador de ayuda	sí	no	no
Círculo	sí	sí	sí
Visor de curvas	sí	sí	sí [1)]
Línea	sí	sí	sí
Visor de avisos Ventana de avisos	sí	sí	sí [1)]
Indicador de avisos	sí	sí	sí
Rectángulo	sí	sí	sí
Visor de recetas	sí	sí	sí [1)]
Botón	sí	sí	sí
Interruptor	sí	sí	sí
Campo E/S simbólico	sí	sí	sí
Visor de diagnóstico del sistema	sí	Sí	sí [1)]
Campo de texto	sí	sí	sí

1) Los objetos se muestran en los Panels como objetos basados en tablas.

Consulte también

Sinopsis de los objetos (Página 8325)

10.1.7.2 Objet



Barra

Utilización

arra" representa variables gráficamente. El visor de barras puede rotularse con de valores.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, colores y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Cambio de color: determina la representación del cambio de color en caso de excederse los valores límite.
- Mostrar líneas y marcas de valores límite: muestra el valor límite configurado en forma de línea o marca.
- Definir segmentos de barras: define la subdivisión de la escala de barras.
- Definir la subdivisión de la escala: define las subdivisiones, marcas e intervalos de una escala de barras.

Cambio de color

Defina la representación del cambio de color en la ventana de inspección "Propiedades > Propiedades > Apariencia".

Cambio de color	Descripción
"Segmentado"	Cuando se alcanza un límite determinado, la barra se colorea por segmentos. La representación segmentada permite visualizar, p. ej., qué límites del valor mostrado se han rebasado por exceso.
"Barra completa"	Cuando se alcanza un límite determinado, se colorea toda la barra.

Mostrar líneas y marcas de valores límite

Las propiedades "Líneas" y "Marcas" permiten mostrar en runtime el valor límite configurado para la barra en forma de línea o marca:

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Apariencia".
2. Active "Líneas" y "Marcas".

Definir segmentos de barras

La propiedad "Subdivisiones" permite definir el número de segmentos en el que se subdividirá la barra con las marcas principales de la escala.

La propiedad "Intervalo" define la distancia entre las marcas principales. El valor se indica en forma de diferencia de valor entre dos marcas principales colindantes:

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Escalas".
2. Active "Mostrar escala".
3. Seleccione el valor correspondiente en "Configuración > Subdivisiones".
4. Seleccione el valor correspondiente en "Configuración > Rotulación de las marcas".
5. Seleccione el valor correspondiente en "Intervalo grande > Intervalo".

Consulte también

Objetos dependientes de los paneles (Página 8423)

Visor de usuarios

Utilización

El objeto "Visor de usuarios" permite configurar y gestionar usuarios y sus respectivos derechos.

Administrador	Grupo de administradores
PLC User	No autorizado
Usuarios_1	Usuarios
Usuarios_2	Grupo_1
<Nuevo usuario>	

Nota

No utilice el visor de usuarios en un grupo.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Número de líneas: establece la cantidad máxima de registros visibles.

Número de líneas

En la ventana de inspección se establece el número de líneas del visor de usuarios que pueden visualizarse en runtime. El número de líneas configurado sólo tendrá efecto si se ha activado la propiedad "Adaptar objeto al contenido".

1. Haga clic en el grupo "Propiedades > Propiedades > Visualización" de la ventana de inspección.
2. En el área "Número de filas", introduzca un valor numérico entero.
3. Active en la ventana de inspección "Propiedades > Propiedades > Representación".
4. Active "Adaptar objeto al contenido".

Consulte también

Objetos dependientes de los paneles (Página 8423)

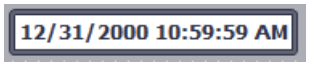
Visor de usuarios simple (Página 9316)

Configurar el visor de usuarios (Página 8619)

Campo de fecha y hora

Utilización

El objeto "Campo de fecha y hora" muestra la fecha y hora del sistema. La representación del campo de fecha y hora depende del idioma configurado en el panel de operador.



Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, estilo, colores y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Mostrar hora del sistema: establece que se muestre la hora del sistema.
- Utilizar variable: Establece que se visualice la hora de la variable conectada.
- Formato de fecha y hora largo: define el formato en el que se mostrarán la fecha y hora.

Mostrar hora del sistema

En la ventana de inspección se especifica la hora que debe visualizarse en el "Campo de fecha y hora" del panel de operador.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. Active "Formato > Hora del sistema".

Utilizar variable

En el campo de fecha y hora aparecerá la hora de la variable incluida.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. En el área "Formato" elija una variable del tipo de datos "DateTime", p. ej. una variable interna.

Formato de fecha y hora largo

La representación de la fecha y hora se especifica en "Propiedades > Propiedades > General > Formato" de la ventana de inspección.

Opción	Descripción
"Activado"	La fecha y la hora se representan en formato largo, p. ej. "Sunday, December 31, 2000 10:59:59 AM"
"Desactivado"	La fecha y la hora se representan en formato abreviado, p. ej. "12/31/2000 10:59:59 AM"

Consulte también

Objetos dependientes de los paneles (Página 8423)

Campo E/S

Utilización

El objeto "Campo E/S" permite la entrada y la visualización de valores de proceso.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, colores y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Modo: determina el comportamiento del objeto en runtime.
- Formato de visualización: especifica el formato de visualización para introducir y emitir valores en el campo E/S.
- Entrada oculta: especifica si el valor de entrada debe visualizarse en formato normal o codificado al introducirlo.

Nota

Informes

En los informes, los campos E/S muestran datos exclusivamente. El modo "Salida" está predeterminado. Las propiedades para configurar la entrada no están disponibles, p. ej. "Entrada oculta".

Modo

El comportamiento del campo E/S se determina en "Propiedades > Propiedades > General > Tipo" de la ventana de inspección.

Modo	Descripción
"Entrada"	En el campo E/S sólo es posible introducir valores en runtime.
"Entrada/salida"	En el campo E/S es posible introducir y emitir valores en runtime.
"Salida"	El campo E/S sólo se utiliza para emitir valores.

Representación

El "formato de visualización" para la entrada y salida de valores se especifica en "Propiedades > Propiedades > General > Formato" de la ventana de inspección.

Representación	
"Binario"	Entrada y salida de valores en formato binario.
"Fecha"	Entrada y salida de fechas. El formato depende del idioma configurado en el panel de operador.
"Fecha/hora"	Entrada y salida de fechas y horas. El formato depende del idioma configurado en el panel de operador.
"Decimal"	Entrada y salida de valores en formato decimal.
"Hexadecimal"	Entrada y salida de valores en formato hexadecimal.
"Hora"	Entrada y salida de tiempos. El formato depende del idioma configurado en el panel de operador.
"Cadena de caracteres"	Entrada y salida de cadenas de caracteres.

Nota

Formatos de datos

En Runtime Professional no están disponibles todos los formatos de datos.

Entrada oculta

La entrada se puede visualizar en runtime de forma normal o codificada, p. ej. para introducir una contraseña de forma oculta. En la entrada oculta se muestra un "*" por cada carácter. El formato de datos del valor de entrada no es reconocible.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Comportamiento".
2. Active "Entrada oculta".

Evitar solapamientos en los campos de salida

Si en una imagen se han configurado varios campos E/S como campos de salida con fondo transparente, es posible que dichos campos E/S se solapen. La parte transparente de un campo cubre las cifras del otro campo. Ello puede ocasionar problemas de visualización en runtime. Para evitar estos solapamientos, ponga a cero los bordes de los campos E/S en "Propiedades > Propiedades > Apariencia" en las propiedades del objeto. Active "Propiedades > Propiedades > Representación > Adaptar objeto al contenido".

Límites

En la ventana de inspección, en "Propiedades > Propiedades > Límites", defina los colores para los valores que rebasan el límite por exceso o por defecto.
Los límites se definen mediante las propiedades de una variable.
En Runtime Professional, además, puede definir un intervalo de valores límite para la entrada en el campo E/S mediante "Propiedades > Propiedades > Límites".

Consulte también

Objetos dependientes de los paneles (Página 8423)

Elipse

Utilización

El objeto "Elipse" es un objeto cerrado que se puede rellenar con un color o una trama.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo y color del objeto. En particular, se adaptan las propiedades siguientes:

- Radio horizontal: determina el radio horizontal del objeto elíptico.
- Radio vertical: determina el radio vertical del objeto elíptico.

Radio horizontal

El radio horizontal del objeto "Elipse" se determina en la ventana de inspección. El valor se indica en píxeles.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. En "Horizontal" introduzca un valor entre 0 y 2500.

Radio vertical

El radio vertical del objeto "Elipse" se determina en la ventana de inspección. El valor se indica en píxeles.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. En "Vertical" introduzca un valor entre 0 y 2500.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Girar un objeto (Página 8340)

Voltear un objeto (Página 8342)

Diseño de bordes (Página 8349)

Visor de gráficos

Utilización

El objeto "Visor de gráficos" permite visualizar gráficos.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Gráfico: determina el archivo gráfico que se visualiza en el objeto.
- Ajustar automáticamente: determina que se adapte automáticamente el tamaño de los objetos que contengan gráficos.
- Color transparente: determina si se utilizará el color transparente para el gráfico.

Insertar un gráfico

En el objeto "Visor de gráficos" se utilizan los formatos siguientes: *.bmp, *.tif, *.png, *.ico, *.emf, *.wmf, *.gif, *.jpg o *.jpeg. Los gráficos también pueden utilizarse como objetos OLE en el visor de gráficos.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. Seleccione el gráfico que desea insertar.
   El gráfico se visualizará a la derecha en la vista preliminar.
3. Para insertar el gráfico en el visor de gráficos haga clic en "Aplicar".

Adaptar gráfico

En la ventana de inspección se determina si el gráfico representado en un visor de gráficos debe adaptarse en runtime al tamaño del visor de gráficos.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Seleccione la adaptación de tamaño deseada para el gráfico.

Color transparente

Con esta propiedad se define si el color transparente se utilizará para el gráfico a visualizar.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Apariencia".
2. Active "Fondo > Transparente".
3. Elija un color transparente.

Nota

Si se utilizan bitmaps en imágenes de WinCC, el ajuste "Color transparente" exige un elevado rendimiento de caracteres en la representación de Panels. Para mejorar dicho rendimiento, desactive el ajuste "Transparente" en las propiedades del objeto de indicación en cuestión. Esta limitación es aplicable sobre todo si los bitmaps se utilizan como imágenes de fondo.

Nota

Basic Panels

La propiedad "Transparente" no está disponible para los Basic Panels.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Agregar un gráfico externo a la colección de gráficos (Página 8358)

Posibilidades de edición de objetos (Página 8328)

Objetos para Basic Panels (Página 8423)

Campo E/S gráfico

Utilización

El objeto "Campo E/S gráfico" permite configurar una lista que sirve para visualizar y seleccionar archivos gráficos.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Modo: determina el comportamiento del objeto en runtime.
- Tipo de barra de desplazamiento: determina la representación gráfica de la barra de desplazamiento.

Nota

Basic Panels

La barra de desplazamiento no está disponible para Basic Panels.

Nota

Informes

En los informes, los campos E/S gráficos muestran exclusivamente gráficos. El modo "Salida" está predeterminado. Las propiedades para configurar la selección de gráficos no están disponibles, p. ej. "Barra de desplazamiento".

Modo

El comportamiento del objeto "Campo E/S gráfico" se establece en "Propiedades > Propiedades > General > Tipo > Modo" de la ventana de inspección.

Modo	Descripción
"Entrada"	El objeto "Campo E/S gráfico" sólo se utiliza para seleccionar gráficos.
"Entrada/salida"	El objeto "Campo E/S gráfico" se utiliza para seleccionar y visualizar gráficos.
"Salida"	El objeto "Campo E/S gráfico" sólo se utiliza para visualizar gráficos.
"Dos estados"	El objeto "Campo E/S gráfico" sólo se utiliza para visualizar gráficos, pudiendo adoptar dos estados como máximo. No utiliza ninguna lista de gráficos, sino que añade un gráfico para el estado "ON" y "OFF".

Adaptar gráfico

En la ventana de inspección se determina si el gráfico representado en un campo E/S gráfico debe adaptarse en runtime al tamaño de la visualización.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Seleccione la adaptación de tamaño deseada para el gráfico.

Tipo de barra de desplazamiento

El comportamiento de la representación gráfica de la barra de desplazamiento se define en "Propiedades > Propiedades > Apariencia > Barra de desplazamiento > Tipo" de la ventana de inspección.

Tipo	Descripción
"Permanente"	La barra de desplazamiento es siempre visible.
"Sin barra de desplazamiento"	La barra de desplazamiento no es visible.
"Visible tras hacer clic"	La barra de desplazamiento se visualiza al hacer clic.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Campo E/S simbólico (Página 8452)

Indicador de ayuda

Utilización

El objeto "Indicador de ayuda" está disponible para los paneles de operador OP 73 y KP300 Basic. El indicador de ayuda aparece en runtime si existe un tooltip para el objeto

seleccionado. El indicador de ayuda estará siempre visible si se ha configurado un tooltip para la imagen abierta.

El objeto "Indicador de ayuda" sólo se puede configurar en la imagen general.

Representación

Adapte la siguiente propiedad en la ventana de inspección:

- Posición: determina la posición del objeto "Indicador de ayuda".

Posición

Esta propiedad determina la posición del objeto "Indicador de ayuda".

1. Seleccione en la plantilla el objeto "Indicador de ayuda".
2. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
3. Introduzca los valores deseados en "X" e "Y". También puede posicionar el objeto seleccionado con ayuda de las teclas de flecha.

Si se ha configurado un objeto de imagen en esa posición de la imagen, el indicador de ayuda visualizado recubrirá dicho objeto. Sólo los avisos del sistema y los cuadros de diálogo que aparezcan pueden ocultar el indicador de ayuda.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Círculo

Utilización

El objeto "Círculo" es un objeto cerrado que se puede rellenar con un color o trama.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo y color del objeto. En particular, se adaptan las propiedades siguientes:

- Radio: determina el tamaño del círculo.

Radio

El radio del objeto "Círculo" se determina en la ventana de inspección. El valor se indica en píxeles.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. En el área "Radio" introduzca un valor entre 0 y 2500.

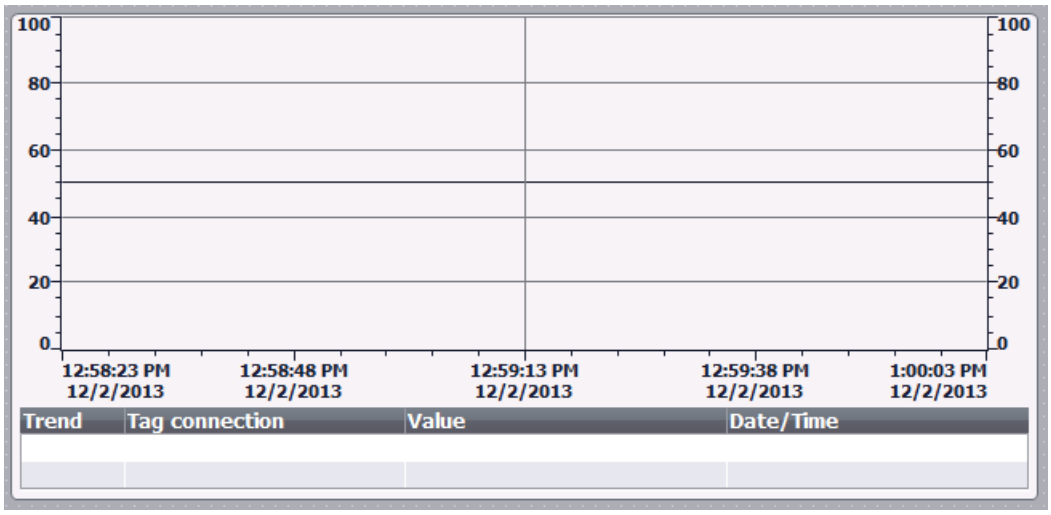
Consulte también

Objetos dependientes de los paneles (Página 8423)

Visor de curvas

Utilización

El visor de curvas permite representar gráficamente variables del proceso actual o bien del fichero en forma de curvas.



Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Mostrar tabla de valores, regla y retícula: determina si, además del sistema de coordenadas, se debe visualizar una tabla de valores, una regla o una retícula para facilitar la lectura.
- Barras de herramientas: define la visualización de los elementos de manejo.

Mostrar tabla de valores, regla y cuadrícula

Para facilitar la lectura se visualiza en runtime una tabla de valores, una regla y una retícula.

1. Active en "Propiedades > Propiedades > Apariencia > Mostrar regla".
2. Active en "Propiedades > Propiedades > Apariencia > Tabla > Mostrar tabla".
3. Active en "Propiedades > Propiedades > Apariencia > Tabla > Mostrar cuadrícula".

Barras de herramientas

La representación de elementos de manejo se especifica en la ventana de inspección "Propiedades > Propiedades > Barra de herramientas".

Nota

Basic Panels

Puesto que la archivación no es posible en los Basic Panels, los elementos de manejo no están disponibles.

Botón "Barra de herramientas"	Descripción abreviada	Descripción
	"Ir al principio"	Retrocede hasta el principio del registro de la curva. Se visualizan los valores iniciales con los que ha empezado el registro de curvas.
	"Ampliar"	Reduce el intervalo de tiempo representado
	"Reducir"	Aumenta el intervalo de tiempo representado.
	"Regla hacia atrás"	Desplaza la regla hacia atrás.
	"Regla hacia adelante"	Desplaza la regla hacia adelante.
	"Hacia atrás"	Retrocede un ancho de visualización.
	"Hacia adelante"	Avanza un ancho de visualización.
	"Regla"	Muestra u oculta la regla. La regla muestra el valor Y correspondiente a un valor X.
	"Inicio/parada"	Detiene o reinicia el registro de la curva

Comportamiento durante la configuración

Visualización de los títulos de las columnas

La representación de la tabla en el visor de curvas dependerá de cómo esté configurada la pantalla en el Panel de control. En función de los ajustes realizados aparecerán cortados los títulos de las columnas de la tabla. Esta opción se encuentra en el Panel de control bajo "Pantalla > Apariencia". Para visualizar correctamente los títulos de las columnas, elija la representación "Windows estilo clásico" en "Ventanas y botones".

Este comportamiento sólo se produce durante la configuración. En runtime se visualizan correctamente los títulos de las columnas.

Comprobación de coherencia

Si al comprobar la coherencia en el visor de curvas aparecen advertencias o errores en la ventana de resultados, no siempre se saltará al lugar exacto del error con el comando "Ir a error/variable" del menú contextual. En ocasiones se muestra únicamente el visor de curvas como causa del error.

Agregar, configurar y eliminar curvas

Las curvas del visor de curvas se administran en la ventana de inspección "Propiedades > Propiedades > Curva". Se pueden copiar curvas entre distintos visores.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Configurar el visor de curvas para los valores del controlador (Página 8501)

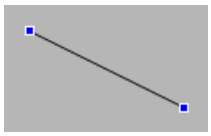
Manejo con pantalla táctil y con teclado (Página 9310)

Visión de conjunto (Página 9309)

Línea

Utilización

El objeto "Línea" es un objeto abierto. La longitud y la inclinación de la línea se determinan a partir de la altura y la anchura del rectángulo que abarca el objeto.



Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo y color del objeto. En particular, se adaptan las propiedades siguientes:

- Estilo de línea
- Principio y fin de línea

Estilo de línea

La representación de la línea se especifica en la ventana de inspección "Propiedades > Propiedades > Apariencia". Si elige "Compacta", p. ej., la línea se representa seguida.

Nota

Los estilos de línea disponibles dependen del panel de operador seleccionado.

Principio y fin de línea

El punto inicial y final de la línea se especifican en la ventana de inspección "Propiedades > Propiedades > Apariencia > Finales de línea".

Los puntos iniciales y finales pueden ser, p. ej., puntas de flecha. Los puntos iniciales y finales disponibles dependen del panel de operador.

Consulte también

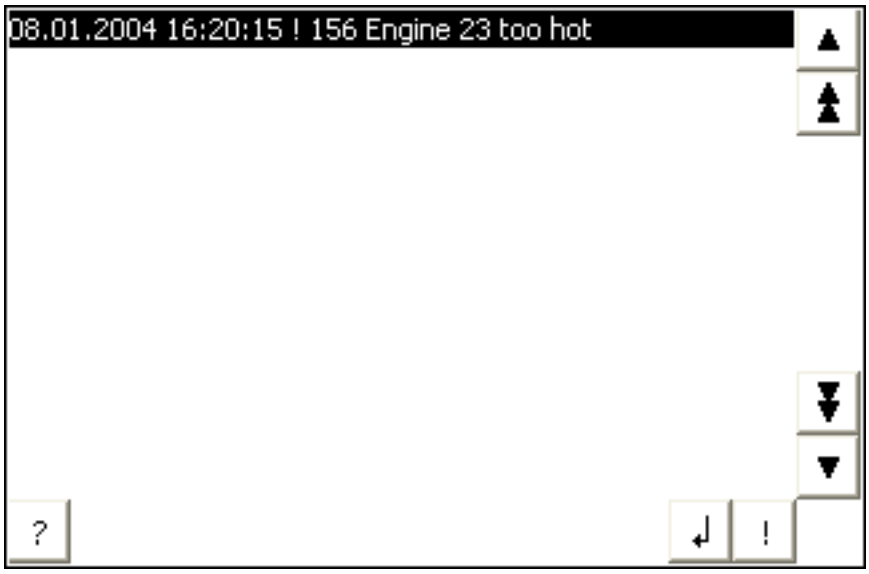
Objetos dependientes de los paneles (Página 8423)

Visor de avisos

Utilización

Los avisos se visualizan en el panel de operador en el visor de avisos, o bien en la ventana de avisos.

La figura siguiente muestra un visor de avisos simple:



Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto.

Nota

La selección de las fuentes depende de las fuentes que se hayan configurado en los ajustes de runtime, en "Idioma y fuente" y de las fuentes soportadas por el panel de operador.

En particular, se adaptan las propiedades siguientes:

- Elementos de control: determina los elementos de control del visor de avisos.
- Categorías: determina qué categorías se mostrarán en el visor de avisos.
- Columnas: determina las columnas a visualizar en runtime.

  Nota

  Si visualiza distintas categorías, en runtime se ordenará primero por categorías y luego por aparición del aviso.

Elementos de control

Los elementos de control que permiten manejar el visor de avisos en runtime se determinan en "Visualización > Configuración" de la ventana de inspección. La tabla siguiente muestra los elementos de control del visor de avisos y su función:

Botón		Función
"Texto de ayuda"	?	Muestra el texto de ayuda de un aviso.
"Acusar"	!	Acusa un aviso.
"Loop-In-Alarm"	↲	Si se ha configurado un cambio de imagen, la visualización cambia a una imagen que contiene información relacionada con el aviso.

Seleccionar categorías

1. En la ventana de inspección haga clic en "Propiedades".
2. En "Categorías" active las categorías que deben representarse en el visor de avisos en runtime.

Definir columnas

Las columnas que deben visualizarse en runtime en el visor de avisos se determinan en la ventana de inspección.

1. En la ventana de inspección haga clic en "Propiedades > Columnas".
2. En "Columnas", active las columnas que deben visualizarse en runtime.

Visualización de los títulos de las columnas

La visualización del visor de avisos depende de cómo está configurada la visualización en el Panel de control. En función de los ajustes realizados aparecerán cortados los títulos de columna. Esta opción se encuentra en el Panel de control bajo "Pantalla > Ficha Apariencia". Para visualizar correctamente los títulos de columna, elija la apariencia "Windows estilo clásico" en "Ventanas y botones".

Este comportamiento sólo se produce durante la configuración. En runtime se visualizan correctamente los títulos de columna.

Nota

En el grupo "Animaciones" de la ventana de inspección del sistema de ingeniería se puede p. ej. dinamizar la visibilidad de un objeto. El objeto "Visor de avisos simple" no soporta ninguna animación en runtime. Si ha configurado una animación y comprueba p. ej. la coherencia del proyecto, se visualizará un mensaje de error en la ventana de resultados.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Ventana de avisos (Página 8443)

Indicador de avisos (Página 8445)

Configurar un visor de avisos (Página 8528)

Ventana de avisos

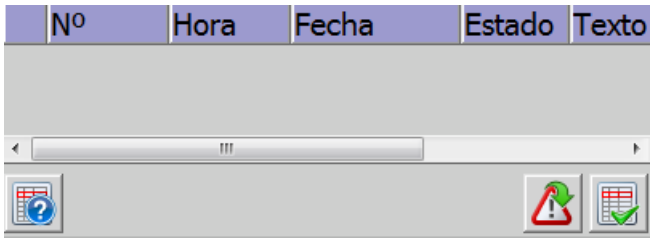
Utilización

Los avisos se visualizan en el panel de operador en el visor de avisos, o bien en la ventana de avisos. La apariencia y el manejo de la ventana de avisos son similares a los del visor de avisos. La ventana de avisos tiene las características siguientes iguales a las del visor de avisos:

- Ventana de avisos
- Línea de avisos: la línea de avisos no está disponible en los Basic Panels.

La ventana de avisos se configura en el editor "Imagen general".

La ventana de avisos no está asignada a ninguna imagen. Dependiendo de la configuración, la ventana de avisos se abre cuando hay un aviso pendiente que pertenece a una determinada categoría. Según la configuración, no volverá a cerrarse hasta que se haya acusado el aviso.



Nota

En la ventana de inspección "Propiedades > Animaciones" del sistema de ingeniería se dinamiza p. ej. la visibilidad de un objeto. El objeto "Ventana de avisos simple" no soporta ninguna animación en runtime. Si ha configurado una animación y comprueba p. ej. la coherencia del proyecto, se visualizará un mensaje de error en la ventana de resultados.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. La ventana de avisos se configura igual que el visor de avisos. La única diferencia es que se adaptan las propiedades siguientes:

- Ventanas de avisos modales: determina que la ventana de avisos se mantiene destacada al cambiar de imagen.
- Ventana: se define el manejo y el comportamiento de la ventana de avisos en runtime.

Nota

Si visualiza distintas categorías, en runtime se ordenará primero por categorías y luego por aparición del aviso.

Elementos de manejo

Los elementos que permiten manejar el visor de avisos en runtime se determinan en la ventana de inspección "Propiedades > Visualización > Configuración". La tabla siguiente muestra los elementos de manejo de la ventana de avisos y su función:

Botón		Función
"Tooltip"		Muestra el tooltip de un aviso.
"Acusar"		Acusa un aviso.
"Loop-In-Alarm"		Si se ha configurado un cambio de imagen, la visualización cambia a una imagen que contiene información relacionada con el aviso.

Protección de acceso en runtime

La protección de acceso se configura en "Propiedades > Propiedades > Seguridad" en la ventana de inspección del visor de avisos. Si un usuario conectado dispone de la autorización necesaria, podrá acusar y editar los avisos en el visor de avisos, utilizando para ello los elementos de manejo. Si el usuario conectado no tiene la autorización necesaria o si ningún usuario ha iniciado una sesión, aparecerá el cuadro de diálogo de inicio de sesión al pulsar los botones "Acusar" o "Editar", o bien al hacer doble clic en una línea de avisos.

Nota

Basic Panels

La protección de acceso no está disponible para Basic Panels.

Destacar la ventana de avisos

Para que al cambiar de imagen la ventana de avisos siga estando destacada, seleccione la opción siguiente:

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Modo".
2. Active "Rotulación".

Ventana

El comportamiento de la ventana de avisos se establece en la ventana de inspección "Propiedades > Propiedades > Modo > Ventana". La tabla siguiente muestra las posibles propiedades:

Opción	Función
Mostrar automáticamente	La ventana de avisos se muestra automáticamente, p. ej. cuando aparece un aviso de sistema.
Cerrable	La ventana se cierra una vez transcurrido un tiempo determinado. El tiempo de visualización se define en la configuración de avisos.
Modal	La ventana de avisos está vinculada a una confirmación, p. ej.: el aviso tiene que acusarse. Si está destacada la ventana de avisos modal, no será posible pulsar los botones en la imagen que se encuentre detrás de ella. Las funciones asociadas a una tecla de función sí se ejecutan.
Tamaño modificable	En runtime puede modificarse el tamaño de la ventana de avisos.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Visor de avisos (Página 8438)

Configurar la ventana de avisos (Página 8529)

Indicador de avisos

Utilización

El indicador de avisos es un símbolo gráfico que, dependiendo de la configuración, puede indicar avisos pendientes o que deban acusarse. El indicador de avisos se configura en el editor "Imagen general". La figura siguiente muestra un indicador de avisos:

Indicador de avisos del OP°73

Para el panel de operador OP 73 se dispone de un indicador de avisos "simple". La figura siguiente muestra el indicador de avisos del panel de operador OP 73:

El indicador de avisos "simple" indica los avisos que deben acusarse o los avisos que ya se han acusado pero todavía no han desaparecido. En el indicador de avisos "simple" sólo es posible especificar la posición. El indicador de avisos se muestra en la posición seleccionada en el panel. Si se ha configurado un objeto de imagen en esa posición de la imagen, el indicador de avisos visualizado ocultará dicho objeto. El indicador de avisos queda oculto por los cuadros de diálogo del sistema (p. ej. cuadro de diálogo de inicio de sesión o de ayuda), así como por las ventanas de avisos.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Categorías: determina las categorías en las que debe visualizarse el indicador de avisos.
- Operación en runtime: determina con qué acción del operador se mostrará la ventana de avisos en runtime.

Categorías

En la ventana de inspección "General > Categorías" se determina qué categorías se mostrarán con indicador de avisos. Las categorías son "Warnings" o "Errors", p. ej.

Definir la operación en runtime

1. Seleccione el indicador de avisos en la imagen.
2. Haga clic en la ventana de inspección "Eventos > Hacer clic" o "Hacer clic cuando parpadee".
3. Se abrirá la "Lista de funciones". Haga clic en la primera fila de la lista de funciones. Aparecerán la lista de las funciones del sistema y de los scripts disponibles en el proyecto.
4. Seleccione la función de sistema "MostrarVentanaDeAvisos" en el grupo "Avisos".
5. En la lista desplegable "Nombre de objeto", seleccione el nombre de la ventana de avisos. Determine en "Representación" si la ventana de avisos se debe mostrar u ocultar, o bien conmutar entre ambos estados.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Visor de avisos (Página 8438)

Configurar el indicador de avisos (Página 8530)

Rectángulo

Utilización

El objeto "Rectángulo" es un objeto cerrado que puede rellenarse con un color.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Radio de las esquinas: determina la distancia horizontal y vertical, respectivamente, entre la esquina del rectángulo que abarca el objeto y el punto inicial de una esquina redondeada.

Radio de las esquinas

Los ángulos del objeto "Rectángulo" se pueden redondear libremente. Al ajustarse las propiedades "X" e "Y" al valor 100 %, el rectángulo se representará con forma de elipse. Si una de las dos propiedades tiene el valor 0 %, se representará un rectángulo normal sin esquina redondeada.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. En el área "Radio de las esquinas", introduzca un valor en "X".
   El valor introducido es la cuota en tantos por ciento de la mitad del ancho del rectángulo.
3. En el área "Radio de las esquinas", introduzca un valor en "Y".
   El valor introducido es la cuota en tantos por ciento de la mitad de la altura del rectángulo.

Consulte también

Objetos dependientes de los paneles (Página 8423)

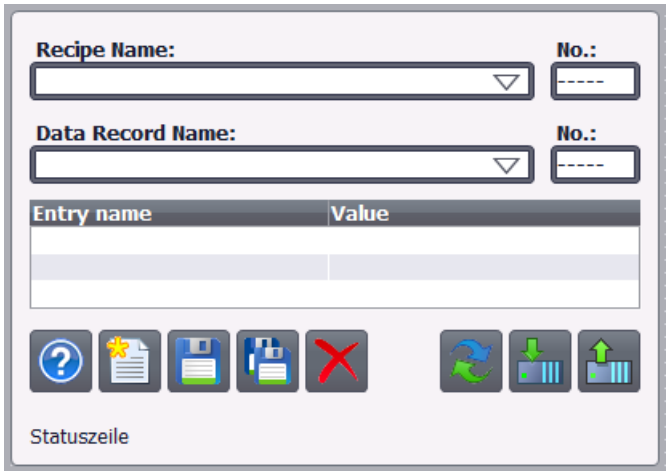
Girar un objeto (Página 8340)

Voltear un objeto (Página 8342)

Visor de recetas

Utilización

El objeto "Visor de recetas" permite visualizar y modificar recetas.



Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Barra de herramientas: determina las entradas de menú del visor de recetas.

Barra de herramientas

Las entradas de menú con las que se maneja el visor de recetas en runtime se configuran en la ventana de inspección en "Propiedades > Propiedades > Barra de herramientas".

Entrada de menú	Descripción
"Tooltip"	Llama al tooltip configurado para la receta seleccionada.
"Agregar registro"	Crea un registro en la receta.
"Borrar registro"	Borra el registro seleccionado.
"Guardar"	Guarda el registro modificado con el nombre actual.
"Guardar como"	Guarda el registro modificado con un nombre nuevo.
"Escribir en controlador"	Transfiere el valor actual al controlador.
"Leer del controlador"	Lee el valor actual del controlador.
"Cambiar nombre"	Determina que se muestre el botón "Cambiar nombre".
"Hacia adelante"	Determina que los botones del menú sean visibles.
"Atrás"	Determina que se muestre el botón "Atrás".

Consulte también

Objetos dependientes de los paneles (Página 8423)

Visor de recetas simple (Página 8579)

Visualización de las recetas (Página 8573)

Configurar el visor de recetas simple (Página 8590)

Descripción del visor de recetas simple (Página 8593)

Interruptor

Utilización

El objeto "Interruptor" permite configurar un interruptor con el que podrá conmutar en runtime entre dos estados predefinidos. El estado actual del objeto "Interruptor" se visualiza con una rotulación o un gráfico.

La figura siguiente muestra un interruptor del tipo "Interruptor".

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adapta la propiedad siguiente:

- Tipo: determina la representación gráfica del objeto.

Tipo

La representación del interruptor se determina en la ventana de inspección "Propiedades > Propiedades > General > Tipo".

Tipo	Descripción
"Interruptor"	Ambos estados del interruptor se representan en forma de un interruptor. La posición del interruptor indica el estado actual. El interruptor se conmuta desplazándolo en runtime. El sentido de desplazamiento de este tipo de interruptor se determina bajo "Dirección de conmutación".
"Interruptor con texto"	El interruptor se representa en forma de botón. El estado actual se visualiza mediante un texto. El interruptor se conmuta en runtime haciendo clic en el botón.
"Interruptor con gráfico"	El interruptor se representa en forma de botón. El estado actual se visualiza mediante un gráfico. El interruptor se conmuta en runtime haciendo clic en el botón.

Nota

Basic Panels

El tipo "Interruptor" no está disponible en los Basic Panels.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Sinopsis de los objetos (Página 8325)

Botón

Utilización

El objeto "Botón" permite configurar el objeto con el que el operador ejecuta en runtime una función configurable cualquiera.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Modo: determina la representación gráfica del objeto.
- Texto / gráfico: determina si la representación gráfica debe ser estática o dinámica.
- Determinar tecla de acceso directo: determina una tecla o una combinación de teclas con la que el operador acciona el botón.

  Nota

  La definición de un hotkey sólo es posible en paneles de operador con teclado.

Modo

La representación del botón se define en la ventana de inspección "Propiedades > Propiedades > General > Modo".

Modo	Descripción
"Invisible"	El botón no se visualiza en runtime.
"Texto"	El botón se visualiza con texto. El texto sirve para explicar la función del botón.
"Gráfico"	El botón se visualiza con un gráfico. Esta imagen sirve para representar la función del botón.

Dependiendo del dispositivo, existen otras opciones.

Texto / gráfico

Dependiendo de la propiedad "Modo", la visualización puede ser estática o dinámica. La visualización se determina en la ventana de inspección "Propiedades > Propiedades > General > Texto" o "Gráfico".

Del tipo "Gráfico" o "Texto", p. ej., puede seleccionar las siguientes opciones.

Tipo	Opción	Descripción
"Gráfico"	"Gráfico"	En "Gráfico si botón "no pulsado"" se define un gráfico que indica el estado "OFF" en el botón. Si activa la opción "Gráfico si botón "pulsado"", puede introducir un gráfico para el estado "ON".
	"Lista de gráficos"	El gráfico del botón depende del estado. En función del estado se visualiza la entrada correspondiente de la lista de gráficos.
"Texto"	"Texto"	En "Texto si botón "no pulsado"" se define un texto que indica el estado "OFF" en el botón. Si activa la opción "Texto si botón "pulsado"", puede introducir un texto para el estado "ON".
	"Lista de textos"	El texto del botón depende del estado. En función del estado se visualiza la entrada correspondiente de la lista de textos.

Definir hotkey

En la ventana de inspección se determina una tecla o una combinación de teclas que permite al operador accionar el botón en runtime.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. En el área "Tecla de acceso directo", seleccione una tecla o una combinación de teclas en la lista de selección.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Ejemplo: Configurar un botón para el cambio de idioma (Página 8403)

Ejemplo: Configurar un botón con el cuadro de diálogo de inicio de sesión (Página 8627)

Ejemplo: Configurar un botón con protección de acceso (Página 8631)

Campo E/S simbólico

Utilización

El objeto "Campo E/S simbólico" permite configurar una lista desplegable para introducir y visualizar textos en runtime.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Modo: determina el comportamiento del objeto en runtime.
- Lista de texto: determina la lista de textos vinculada al objeto.
- Botón para lista desplegable: determina que el objeto dispone de un botón para abrir la lista desplegable.

Nota

Informes

En los informes, los campos E/S muestran datos exclusivamente. El modo "Salida" está predeterminado. Las propiedades para configurar la selección de gráficos no están disponibles, p. ej. "Botón de lista de selección".

Modo

El comportamiento del campo E/S simbólico se determina en la ventana de inspección "Propiedades > Propiedades > General > Tipo".

Modo	Descripción
"Salida"	El campo E/S simbólico se utiliza para visualizar valores.
"Entrada"	El campo E/S simbólico se utiliza para introducir valores.
"Entrada/salida"	El campo E/S simbólico se utiliza para introducir y visualizar valores.
"Dos estados"	El campo E/S simbólico se utiliza para la salida de valores y tiene como máximo dos estados. El campo conmuta entre dos textos predefinidos. Así, p. ej., se visualizan los dos estados de una válvula, cerrada o abierta.

Nota

En función del runtime, el campo E/S simbólico tendrá un comportamiento u otro.

Lista de textos

En la ventana de inspección se determina qué lista de textos se vinculará al campo E/S simbólico.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. En "Contenido" abra la lista de selección para "Lista de textos".
3. Seleccione una lista de textos.

Botón para lista desplegable

La propiedad "Botón para lista desplegable" permite visualizar un botón para abrir la lista desplegable.

1. Haga clic en "Propiedades > Propiedades > Representación" de la ventana de inspección.
2. Active "Comportamiento > Botón para lista desplegable".

Nota
Basic Panels

La opción "Botón de lista de selección" no está disponible para Basic Panels.

Consulte también

Objetos dependientes de los paneles (Página 8423)

Campo E/S gráfico (Página 8432)

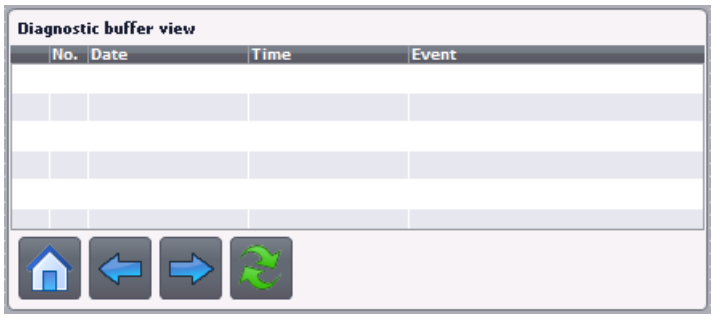
Visor de diagnóstico del sistema

Introducción

El visor de diagnóstico del sistema ofrece una vista general de todos los dispositivos disponibles en la instalación. Permite navegar directamente hasta la causa de un error y hasta el dispositivo correspondiente. Permite acceder a todos los dispositivos aptos para diagnóstico que ha configurado en el editor "Dispositivos y redes".

Utilización

Gracias al visor de diagnóstico del sistema se consigue el máximo grado de detalle posible de los datos de diagnóstico. Se obtiene un diagnóstico preciso, dado que se muestran todos los datos disponibles. A simple vista se puede ver el estado de toda la instalación.



El visor de diagnóstico del sistema ofrece tres vistas distintas.

- Vista de dispositivos
- Vista del búfer de diagnóstico
- Vista detallada

Vista de dispositivos

La vista de dispositivos del visor de diagnóstico del sistema muestra todas las conexiones disponibles en una tabla. Haciendo doble clic sobre una conexión se abre la vista detallada. La vista de dispositivos sólo se muestra, si se ha creado más de una conexión en el editor "Dispositivos y redes".

Vista del búfer de diagnóstico

En la vista del búfer de diagnóstico se muestran los datos actuales del búfer de diagnóstico.

Vista detallada

La vista detallada muestra información detallada sobre la conexión seleccionada. No es posible escoger los textos de error en la vista detallada. La vista detallada sólo está disponible si existe una conexión integrada para una S7 1200 o S7 1500.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuente del objeto. En particular, se adaptan las propiedades siguientes:

- Filas por entrada: define el número de filas que se muestran para una entrada.

Configurar el visor de diagnóstico del sistema

1. Arrastre el visor de diagnóstico del sistema fuera de la ventana de herramientas utilizando la función Drag&Drop.
2. En la ventana de inspección, haga clic en "Propiedades > Representación".
3. Seleccione para "Filas por entrada", p. ej. 5.
4. Seleccione una autorización de manejo en "Propiedades > Propiedades > Seguridad".

Consulte también

Vistas en el diagnóstico de sistema (Página 8566)

Campo de texto

Utilización

El objeto "Campo de texto" es un objeto cerrado que puede rellenarse con un color.

Representación

En la ventana de inspección se modifican los ajustes correspondientes a la posición, geometría, estilo, color y fuentes del objeto. En particular, se adaptan las propiedades siguientes:

- Texto: determina el texto para el campo de texto.
- Tamaño del campo de texto: determina si el tamaño del objeto debe adaptarse al espacio necesario para el registro más largo de la lista.

Texto

El texto para el campo de texto se determina en la ventana de inspección.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > General".
2. Introduzca un texto.
   Si el texto tiene varias líneas, el salto de línea se activa pulsando la combinación de teclas <Mayús + Intro>.

Tamaño del campo de texto

En la ventana de inspección se determina si el tamaño del objeto debe adaptarse al espacio necesario para el registro más largo de la lista.

1. Haga clic en la ventana de inspección "Propiedades > Propiedades > Representación".
2. Active "Adaptación de tamaño > Adaptar tamaño al contenido".

Consulte también

Objetos dependientes de los paneles (Página 8423)

10.1.8 Configurar la navegación de imágenes

10.1.8.1 Principios básicos de la navegación de imágenes

Tipos de navegación para el cambio de imagen

Para un proceso de fabricación compuesto de varios procesos parciales se configuran diferentes imágenes. Para que el operador pueda cambiar de una imagen a otra en runtime, existen las posibilidades siguientes:

- Asociar un cambio de imagen a botones
- Asociar un cambio de imagen a una tecla de función local

Procedimiento

Antes de crear un cambio de imagen, defina la estructura de la instalación y derive de ella los cambios de imagen que desee configurar.

La imagen inicial se define en "Configuración de runtime > General > Imagen inicial".

Consulte también

Asignar un cambio de imagen a un botón (Página 8457)

10.1.8.2 Asignar un cambio de imagen a un botón

Introducción

Para poder cambiar entre las diversas imágenes del panel de operador durante el funcionamiento, configure un botón en la imagen.

Nota

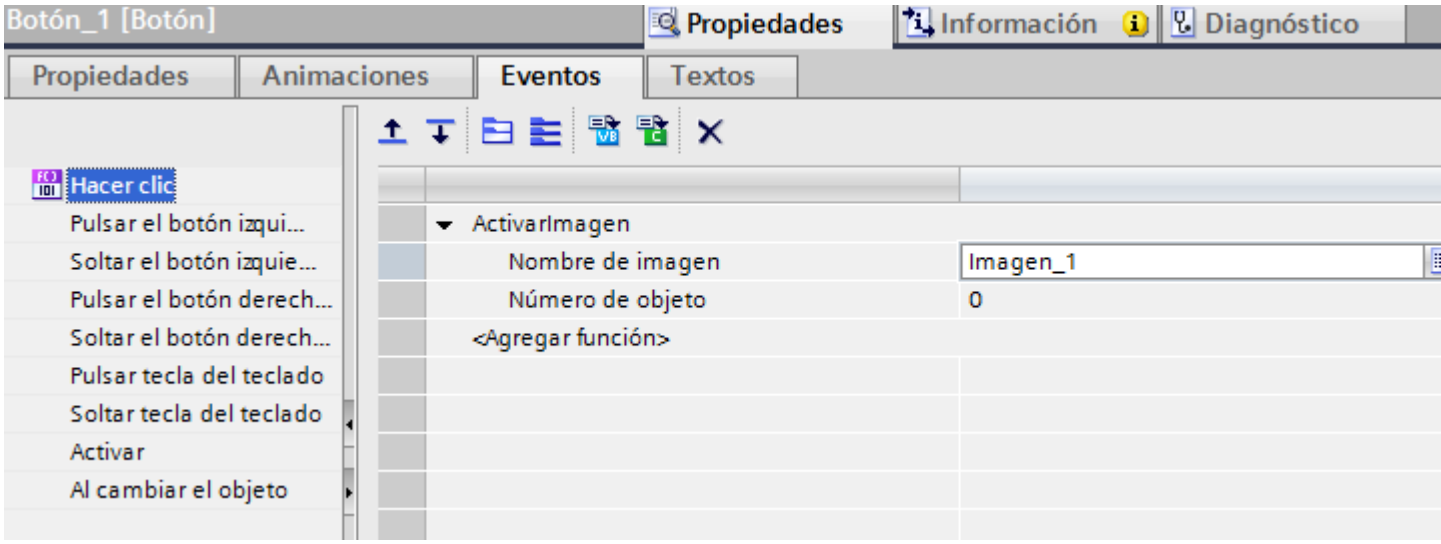
Si en la ventana de inspección de una imagen se ha configurado que la animación "Visibilidad" esté "oculta", esta imagen no se podrá abrir en runtime.

Requisitos

- Se ha creado un proyecto.
- La imagen "Screen_2" está creada.
- La imagen "Screen_1" está creada.

Procedimiento

1. Haga doble clic en "Screen_1" en el árbol del proyecto. La imagen se muestra en el área de trabajo.
2. Arrastre "Screen_2" desde el árbol del proyecto hasta la imagen abierta mediante Drag&Drop.
   Se insertará un botón con el nombre "Screen_1".
3. Haga clic en la ventana de inspección "Propiedades > Eventos > Hacer clic".
   En la "Lista de funciones" se visualizará la función de sistema "ActivarImagen".



4. En caso necesario, introduzca como "Número de objeto" el número de orden de tabuladores del objeto que debe quedar resaltado tras el cambio de imagen. También puede indicar una variable que contenga el número del objeto.

Procedimiento alternativo

1. Arrastre un botón a la imagen "Screen2" desde la Task Card "Herramientas" mediante arrastrar y soltar.
2. Haga clic en la ventana de inspección "Propiedades > Eventos > Hacer clic".
3. Elija la función de sistema "ActivarImagen".
4. Seleccione la imagen "Screen_2" para "Número de imagen".

Resultado

Cuando el operador pulse el botón en runtime, se cambiará a la imagen "Screen_1". Si ha indicado un número de objeto, el objeto correspondiente se resaltará tras el cambio de imagen.

Consulte también

Principios básicos de la navegación de imágenes (Página 8454)

10.1.8.3 Asignar un cambio de imagen a una tecla de función

Introducción

Para poder cambiar entre las diversas imágenes del panel de operador durante el funcionamiento, configure una tecla de función para el cambio de imagen.

Nota

Si en la ventana de inspección de una imagen se ha configurado que la animación "Visibilidad" esté "oculta", esta imagen no se podrá abrir en runtime.

Requisitos

- Se ha creado un proyecto.
- La imagen "Screen_2" está creada.
- La imagen "Screen_1" está creada.

Procedimiento

1. Haga doble clic en "Screen_1" en el árbol del proyecto. La imagen se muestra en el área de trabajo.
2. Arrastre "Screen_2" desde el árbol del proyecto a una tecla de función, p. ej., "F2".
   La tecla de función configurada muestra un triángulo amarillo.
3. Haga clic en la ventana de inspección "Propiedades > Eventos > Pulsar tecla".
   La función de sistema mostrará "ActivarImagen".

Resultado

Cuando el operador pulse la tecla de función "F2" en runtime, cambiará a la imagen "Screen_2".

10.2 Trabajar con variables

10.2.1 Principios básicos

10.2.1.1 Principios básicos de las variables

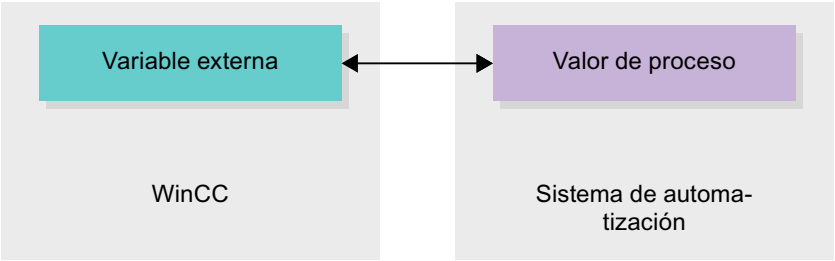
Introducción

Los valores de proceso se transfieren en runtime por medio de variables. Los valores de proceso son datos que se almacenan en la memoria de uno de los sistemas de automatización conectados. Representan el estado de una instalación en forma de temperaturas, niveles de llenado o estados de conexión. Para la edición de los valores de proceso en WinCC se definen variables externas.

WinCC trabaja con dos tipos de variables:

- Variables externas
- Variables internas

Las variables externas constituyen el vínculo entre el WinCC y los sistemas de automatización. Los valores de variables externas se corresponden con los valores de proceso de la memoria de un sistema de automatización. El valor de una variable externa se determina mediante la lectura del valor de proceso de la memoria del sistema de automatización. A la inversa, también se puede reescribir un valor de proceso en la memoria del sistema de automatización.



Las variables internas no poseen ninguna conexión a proceso y sólo transportan valores dentro de WinCC. Las variables solo están disponibles mientras se ejecute runtime.

Variables en WinCC

En el caso de las variables externas, en las propiedades de la variable correspondiente se determina a través de qué conexión se comunica WinCC con el sistema de automatización y cómo se realiza el intercambio de datos.

Las variables que no reciben datos del proceso se denominan variables internas y no están conectadas con el sistema de automatización. Esto se reconoce por la propiedad "Conexión" de la variable, en la entrada "Variable interna".

Para tener una visión más clara, cree diferentes tablas de variables. En la navegación del proyecto, acceda directamente a las distintas tablas de variables en el nodo "Variables HMI". Con ayuda de la tabla "Mostrar todas las variables" se pueden visualizar las variables de todas las tablas.

Consulte también

Resumen de las tablas de variables HMI (Página 8460)

Variables internas (Página 8466)

Variables externas (Página 8461)

Direccionamiento de variables externas (Página 8463)

Crear una variable externa (Página 8467)

Principios básicos de las matrices (Página 8496)

Principios básicos de los ciclos (Página 8500)

10.2.1.2 Resumen de las tablas de variables HMI

Introducción

Las tablas de variables HMI contienen las definiciones de las variables HMI vigentes para el dispositivo. Para cada dispositivo HMI creado en el proyecto se genera automáticamente una tabla de variables.

En el árbol del proyecto existe para cada dispositivo HMI una carpeta "Variables HMI". Ésta puede incluir las siguientes tablas:

- Tabla de variables estándar
- Tablas de variables definidas por el usuario
- Todas las variables

En una tabla de variables HMI están disponibles adicionalmente las siguientes tablas:

- Avisos de bit
- Avisos analógicos

Con ayuda de estas tablas se configuran los avisos para la variable HMI actualmente seleccionada.

En el árbol del proyecto pueden crearse otras tablas de variables en la carpeta Variables HMI para clasificar y agrupar en ellas las variables y las constantes. Las variables pueden desplazarse a otra tabla de variables con la función Drag&Drop o por medio del campo "Tabla de variables". El campo "Tabla de variables" se activa con el menú contextual de los títulos de columna.

Tabla de variables estándar

Para cada dispositivo HMI del proyecto existe una tabla de variables estándar. No es posible borrar, cambiar el nombre o desplazar dicha tabla. La tabla de variables estándar contiene variables HMI, pero también variables de sistema que dependen del dispositivo HMI. Es posible declarar todas las variables HMI en la tabla de variables estándar, o bien crear otras tablas de variables definidas por el usuario en caso necesario.

Tablas de variables definidas por el usuario

Existe la posibilidad de crear varias tablas de variables definidas por el usuario para cada dispositivo HMI, con el fin de agrupar las variables en función de sus necesidades. Es posible borrar, agrupar o cambiar el nombre de una tabla de variables definida por el usuario. Para la agrupación de tablas de variables deben crearse nuevas subcarpetas dentro de la carpeta Variables HMI.

Todas las variables

La tabla "Todas las variables" muestra un resumen de todas las variables HMI y variables de sistema del dispositivo HMI pertinente. No es posible borrar, cambiar el nombre o desplazar dicha tabla.

Tabla Avisos de bit

En la tabla "Avisos de bit" se asocian los avisos de bit a la variable HMI seleccionada en la tabla de variables HMI. Cuando se configura un aviso de bit, no es posible realizar una selección múltiple en la tabla de variables HMI. Los avisos de bit deben configurarse por separado para cada variable HMI.

Tabla Avisos analógicos

En la tabla "Avisos analógicos" se asocian los avisos analógicos a la variable HMI seleccionada en la tabla de variables HMI. Cuando se configura un aviso analógico, no es posible realizar una selección múltiple en la tabla de variables HMI. Los avisos analógicos deben configurarse por separado para cada variable HMI.

Consulte también

Principios básicos de las variables (Página 8457)

10.2.1.3 Variables externas

Introducción

Las variables externas hacen posible la comunicación, es decir, el intercambio de datos entre los componentes de un sistema de automatización, p. ej. entre el panel de operador y el controlador.

Principio

Una variable externa es la imagen de una posición de memoria definida en el autómata. Es posible acceder con derechos de lectura y escritura a esta posición de memoria tanto desde el panel de operador como desde el controlador.

Como las variables externas son la imagen de una posición de memoria del controlador, los tipos de datos que se pueden utilizar dependerán del controlador al que esté conectado el panel de operador.

Si se escribe un programa de control para el PLC en STEP 7, las variables del PLC creadas en el programa de control se registrarán en la tabla de variables del PLC. Si desea conectar una variable externa con una variable del PLC, acceda directamente a las variables del PLC desde la tabla de variables y conéctelas con la variable externa.

Tipos de datos

En una variable externa el usuario tiene disponible en WinCC todos los tipos de datos que, a su vez, estén disponibles en el controlador conectado. En la documentación de los drivers de comunicación correspondientes encontrará información sobre los tipos de datos disponibles para la conexión a otros controles.

Encontrará más información al respecto en "Principios básicos de la comunicación (Página 8736)".

Nota

Además de las variables externas, para la comunicación entre el panel de operador y el controlador hay disponibles punteros de área. Los punteros de área se pueden configurar y activar en el editor "Conexiones".

Sincronización con variables PLC

En "Configuración de variables" de los ajustes de runtime se ofrecen distintas opciones para sincronizar variables externas con las variables PLC.

Al sincronizar, se pueden aplicar automáticamente los nombres de variables del PLC a variables externas, así como reconectar las variables existentes.

El nombre de variable generado se deriva de la posición del valor de datos en la estructura jerárquica del bloque de datos.

Actualización de valores de variables

En las variables externas, los valores de variable actuales se transmiten en runtime a través del enlace de comunicación entre el WinCC y los sistemas de automatización conectados y se almacenan en la memoria de runtime. A continuación, el valor de la variable se actualiza conforme al tiempo de ciclo configurado. Para la aplicación en el proyecto runtime, WinCC accede a valores de variable de la memoria de runtime leídos del controlador en el instante

anterior del ciclo. De este modo se puede modificar ya el valor en el controlador mientras se edita el valor de la memoria de runtime.

Nota

Elementos de matrices PLC en combinación con S7 1200 o S7 1500

El índice de los elementos de matrices PLC puede empezar con cualquier número. En WinCC, la indexación siempre empieza con 0.

Una variable PLC "Array[1..3] of Int" se representa en WinCC, p. ej., con "Array[0..2] of Int".

Si accede a una matriz en un script, asegúrese de que la indexación sea correcta.

Consulte también

Direccionamiento de variables externas (Página 8463)

Principios básicos de la comunicación (Página 8736)

Principios básicos de las variables (Página 8457)

10.2.1.4 Direccionamiento de variables externas

Introducción

Las posibilidades de direccionamiento de variables externas dependen del tipo de conexión entre WinCC y el controlador correspondiente. Se deben diferenciar los siguientes tipos de conexión:

- Conexión integrada
  Las conexiones de dispositivos que se encuentran dentro de un proyecto y se han elaborado con el editor "Dispositivos y redes" se denominan conexiones integradas.
- Conexión no integrada
  Las conexiones de dispositivos elaboradas con el editor "Conexiones" se denominan conexiones no integradas. No todos los dispositivos se deben encontrar dentro de un proyecto.

El tipo de conexión se reconoce por su símbolo.

	Conexión integrada
	Conexión no integrada

Encontrará más información al respecto en el capítulo Principios básicos de la comunicación (Página 8736).

Direccionamiento en conexión integrada

Una conexión integrada ofrece la ventaja de que una variable se puede direccionar tanto de forma simbólica como absoluta.

En un direccionamiento simbólico se selecciona la variable PLC por su nombre para conectarla con la variable HMI. El sistema selecciona automáticamente el tipo de datos válido para la variable HMI. Al direccionar elementos en bloques de datos deben distinguirse los casos siguientes:

Direccionamiento simbólico de bloques de datos con acceso optimizado y acceso estándar:

Al direccionar simbólicamente un bloque de datos con acceso optimizado y con acceso estándar, se asigna dinámicamente la dirección de un elemento en el bloque de datos y, en caso de modificar la dirección, el cambio se aplica automáticamente en la variable HMI. Para ello no es necesario compilar el bloque de datos conectado o bien el proyecto de WinCC. En los bloques de datos con acceso optimizado sólo está disponible el direccionamiento simbólico.

Al direccionar simbólicamente los elementos de un bloque de datos, el proyecto de WinCC debe compilarse y cargarse nuevamente sólo si se realizan los cambios siguientes:

- Si se han cambiado el nombre o el tipo de datos del elemento de bloque de datos conectado o de la variable PLC global.
- Si se han cambiado el nombre o el tipo de datos de un nodo de estructura de orden superior contenido en el elemento de bloque de datos conectado o la variable PLC global del elemento conectado.
- Si ha cambiado el número del bloque de datos conectado.

Actualmente se dispone de direccionamiento simbólico con los siguientes controladores:

- SIMATIC S7 1200
- SIMATIC S7 1500

Además, se dispone del direccionamiento simbólico en una conexión integrada.

Nota

Si utiliza el puntero de área "Fecha/hora", no es posible el direccionamiento simbólico.

El direccionamiento absoluto también se puede utilizar en una conexión integrada. Las variables PLC de un controlador SIMATIC S7 300/400 sólo se pueden direccionar de modo absoluto. Si ha conectado una variable HMI con una variable PLC y la dirección de la variable PLC cambia, hay que volver a compilar el programa de control para que se actualice la nueva dirección en WinCC. A continuación compile de nuevo el proyecto de WinCC y cárguelo en el panel de operador.

En WinCC, el direccionamiento simbólico está predeterminado como configuración básica. Para modificar el ajuste estándar, elija el comando de menú "Herramientas > Configuración". En el cuadro de diálogo "Configuración" seleccione "Visualización > Variables". Si es necesario, desactive la opción "Acceso simbólico".

La disponibilidad de una conexión integrada depende del controlador empleado. La tabla siguiente muestra la disponibilidad:

Controlador	Conexión integrada	Observaciones
S7 300/400	Sí	El enlace de las variables no se comprueba en runtime. Si se modifica la dirección de variable en el controlador y no se compila y carga de nuevo el dispositivo HMI, no se registra el cambio en runtime.
S7 1200	Sí	En el direccionamiento simbólico se realiza una comprobación de validez de la conexión de variables en runtime. Si se produce un cambio de dirección en el controlador, dicho cambio se registra y se emite un mensaje de error. En el caso de direccionamiento absoluto se aplica el procedimiento descrito para el S7 300/400.
S7 1500	Sí	En el direccionamiento simbólico se realiza una comprobación de validez de la conexión de variables en runtime. Si se produce un cambio de dirección en el controlador, dicho cambio se registra y se emite un mensaje de error. En el caso de direccionamiento absoluto se aplica el procedimiento descrito para el S7 300/400.

Una conexión integrada se crea en el editor "Dispositivos y redes". Si el controlador está incluido en el proyecto y soporta las conexiones integradas, también se puede generar la conexión de forma automática. Para ello sólo tiene que seleccionar durante la configuración de la variable HMI una variable PLC existente con la que desee conectar la variable HMI. Entonces, el sistema genera de forma automática la conexión integrada.

Direccionamiento en conexión no integrada

En un proyecto con conexión no integrada siempre se proyecta una conexión de variables con direccionamiento absoluto. Usted mismo debe seleccionar el tipo de datos válido. Si en el transcurso de un proyecto con conexión no integrada cambia la dirección de una variable PLC, debe realizar asimismo la modificación en WinCC. No se puede comprobar la validez de la conexión de variables en runtime; no se produce ningún mensaje de error.

Una conexión no integrada está disponible para todos los controladores soportados.

El direccionamiento simbólico no está disponible en una conexión no integrada.

En una conexión no integrada, el programa de control no debe formar parte del proyecto WinCC. La configuración del controlador y del proyecto de WinCC puede llevarse a cabo de forma independiente. Para la configuración en WinCC se deben conocer simplemente las direcciones empleadas en el controlador y su función.

Consulte también

Variables externas (Página 8459)

Principios básicos de las variables (Página 8457)

Principios básicos de la comunicación (Página 8736)

10.2.1.5 Variables internas

Introducción

Las variables internas no tienen conexión con el controlador. Las variables internas transportan valores dentro del
panel de operador. Las variables solo están disponibles mientras se ejecute runtime.

Principio

Las variables internas se almacenan en la memoria del panel de operador. De esta manera, sólo se puede acceder a ellas con derechos de lectura y escritura desde este panel de operador. Las variables internas se crean, p. ej., para realizar cálculos locales.

Las variables internas disponen de los tipos de datos HMI.

Están disponibles los tipos de datos HMI siguientes:

Tipo de datos HMI	Formato de datos
Matriz	Matriz unidimensional
Bool	Variable binaria
DateTime	Formato fecha/hora
DInt	Valor de 32 bits con signo
Int	Valor de 16 bits con signo
LReal	Coma flotante de 64 bits IEEE 754
Real	Coma flotante de 32 bits IEEE 754
SInt	Valor de 8 bits con signo
UDInt	Valor de 32 bits sin signo
UInt	Valor de 16 bits sin signo
USInt	Valor de 8 bits sin signo
WString	Variable de texto de 16 bits juego de caracteres

Consulte también

Principios básicos de las variables (Página 8457)

10.2.2 Trabajar con variables

10.2.2.1 Crear variables

Crear una variable externa

Introducción

Con una variable externa se accede a una dirección del controlador a través de una variable PLC. Para el direccionamiento existen las siguientes posibilidades:

- Direccionamiento simbólico
- Direccionamiento absoluto

Encontrará más información sobre el direccionamiento simbólico en el capítulo "Direccionamiento de variables externas (Página 8461)". Utilice siempre que sea posible el direccionamiento simbólico al configurar una variable. Las variables se crean en la tabla de variables estándar o en una tabla de variables definida por el usuario.

Requisitos

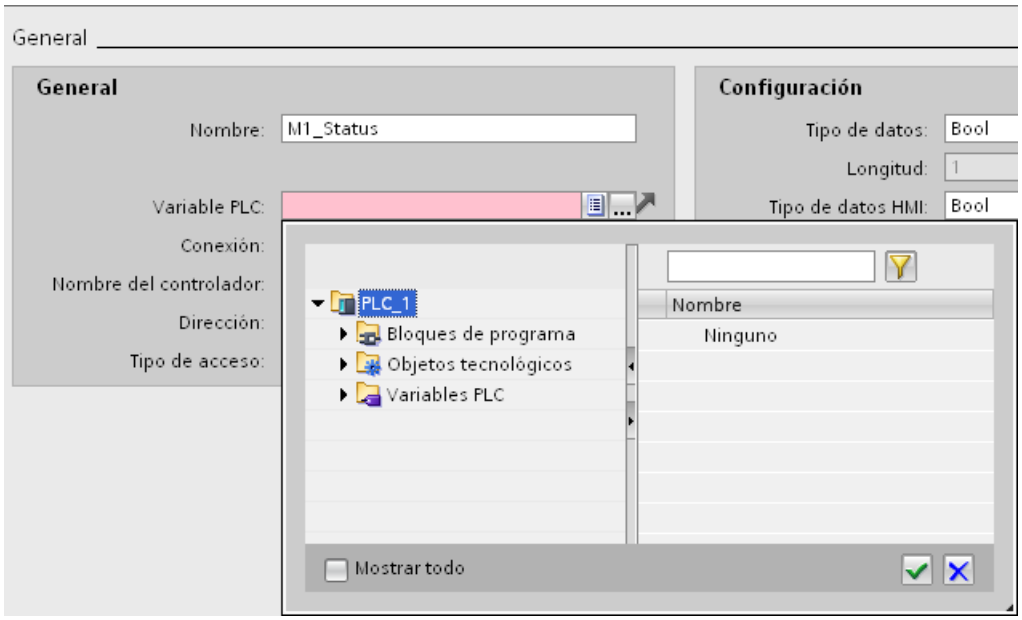
- El proyecto está abierto.
- Se ha configurado una conexión con el controlador.
- La ventana de inspección está abierta.

Procedimiento

Para crear una variable externa, proceda del siguiente modo:

1. En la navegación del proyecto, abra la carpeta "Variables HMI" y haga doble clic en la tabla de variables estándar. Se abre la tabla de variables.
   También puede crear una tabla de variables nueva y abrirla a continuación.
2. En la tabla de variables, haga doble clic en "Agregar" de la columna "Nombre". Se creará una nueva variable.
3. En la ventana de inspección seleccione la categoría "Propiedades > Propiedades > General" y, en caso necesario, introduzca un nombre de variable propio en el campo "Nombre". El nombre de la variable debe ser unívoco en todo el equipo.
4. Si es necesario, introduzca el nombre que debe aparecer en runtime en el campo "Nombre de visualización". El nombre de visualización depende del idioma y puede traducirse a los idiomas de runtime deseados. El nombre de visualización está disponible para Basic Panels, Panels y Runtime Advanced.

5. En el campo "Conexión" seleccione la conexión con el controlador deseado. Si no aparece la conexión deseada, primero deberá crear la conexión con el controlador. La conexión con un controlador SIMATIC S7 se crea en el editor "Dispositivos y redes". La conexión con controladores de otros fabricantes se crea en el editor "Conexiones".
   Si el controlador está incluido en el proyecto y soporta conexiones integradas, la conexión también puede crearse de forma automática. Para ello, al configurar la variable HMI, seleccione sencillamente una variable PLC existente con la que desee conectar la variable HMI. A continuación, el sistema creará automáticamente la conexión integrada.
6. Si trabaja con una conexión integrada, haga clic en el botón ... del campo "Variable PLC" y seleccione una variable PLC ya creada de la lista de objetos. Confirme la selección con el botón ✓.



7. Si trabaja con una conexión no integrada, introduzca la dirección del controlador en el campo "Dirección". El campo "Variable PLC" permanece vacío.
8. Configure las otras propiedades de la variable en la ventana de inspección.

También puede configurar las propiedades de la variable directamente en la tabla de variables. Las columnas ocultas se activan con el menú contextual de los títulos de columna.

Las nuevas variables también pueden crearse directamente en el punto de aplicación, p. ej., en un campo E/S. Para ello, haga clic en el botón Agregar... de la lista de objetos. A continuación, la nueva variable se configura en la ventana de inspección.

Resultado

Se ha creado una variable externa y se ha conectado con una variable PLC o una dirección en el controlador.

Procedimiento alternativo

También se pueden crear las variables externas HMI por medio de la función Drag&Drop desde elementos de bloque de datos o de variables PLC globales hacia una tabla de variables HMI.

Consulte también

Crear una variable interna (Página 8469)

Crear varias variables (Página 8470)

Editar variables (Página 8471)

Valores límite de una variable (Página 8477)

Principios básicos de las variables (Página 8457)

Direccionamiento de variables externas (Página 8461)

Crear una variable interna

Introducción

Las variables internas deben tener definidos como mínimo el nombre y el tipo de datos. En lugar de una conexión con un controlador se selecciona la entrada "Variable interna".

Para fines de documentación se recomienda introducir un comentario para cada variable.

Requisitos

El proyecto está abierto.

Procedimiento

1. En la navegación del proyecto, abra la carpeta "Variables HMI" y haga doble clic en la entrada "Tabla de variables estándar". Se abre la tabla de variables.
   También puede crear una tabla de variables nueva y abrirla a continuación.
2. En la tabla de variables, haga doble clic en "Agregar", en la columna "Nombre". Se creará una nueva variable.
3. Si la ventana de inspección no está abierta, elija la opción "Ventana de inspección" del menú "Ver".
4. En la ventana de inspección seleccione la categoría "Propiedades > Propiedades > General" y, en caso necesario, introduzca un nombre de variable propio en el campo "Nombre". El nombre de la variable debe ser unívoco en todo el proyecto.
5. Si es necesario, introduzca el nombre que debe aparecer en runtime en el campo "Nombre de visualización". El nombre de visualización depende del idioma y puede traducirse a los idiomas de runtime deseados. El nombre de visualización está disponible para Basic Panels, Panels y Runtime Advanced.
6. En el campo "Conexión" seleccione la entrada "Variable interna".

7. En el campo "Tipo de datos", seleccione el tipo de datos deseado.
8. En función del tipo de datos seleccionado, en el campo "Longitud" deberá definirse el número máximo de caracteres que pueden depositarse en la variable. En las variables numéricas, la longitud queda automáticamente definida por el tipo de datos.
9. Opcionalmente, es posible introducir un comentario sobre el uso de la variable. Para ello, haga clic en la ventana de inspección en "Propiedades > Propiedades > Comentario" e introduzca un texto.

También puede configurar las propiedades de la variable directamente en la tabla de variables. Las columnas ocultas se activan con el menú contextual de los títulos de columna.

Las nuevas variables también pueden crearse directamente en el punto de aplicación, p. ej., en un campo E/S. Para ello, haga clic en el botón Agregar... de la lista de objetos. A continuación, la nueva variable se configura en la ventana de propiedades que se visualiza.

Resultado

Se crea una variable interna que puede emplearse en el proyecto.

Luego puede configurar la variable p. ej. determinando el valor inicial y los valores límite.

Consulte también

Crear una variable externa (Página 8465)

Crear varias variables

Introducción

En una tabla de variables se crea un gran número de variables idénticas haciendo que las filas de la tabla que estén por debajo de una variable se rellenen automáticamente.

Al rellenar las filas automáticamente, los nombres de las variables se incrementan automáticamente.

Rellenando las filas automáticamente también es posible transferir propiedades concretas de una variable a las celdas inferiores de la tabla y, con ello, a las variables correspondientes.

Si aplica el llenado automático a celdas de una tabla de variables que ya estén llenas, se le consultará si desea sobrescribir las celdas o insertar nuevas variables.

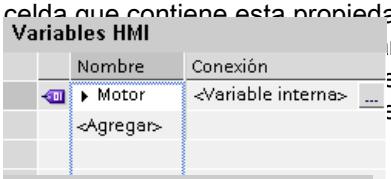
Si no desea sobrescribir variables ya configuradas, active el modo de inserción. El modo de inserción se activa manteniendo pulsada la tecla <Ctrl> durante la inserción. Con el modo de inserción activado, las entradas ya existentes de la tabla de variables se desplazan hacia abajo.

Requisitos

- El proyecto está abierto.
- Hay una tabla de variables abierta.
- La variable que debe servir de modelo para otras variables está configurada.

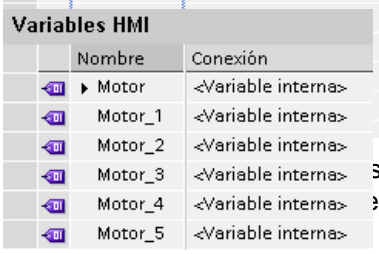
Procedimiento

1. Si desea crear variables nuevas, en la columna "Nombre" marque la variable que debe servir de modelo para las variables nuevas.
   Si desea transferir una propiedad de una variable a las variables inferiores, seleccione la celda que contiene esta propiedad.



   rá en color y en la esquina inferior derecha aparecerá un
   e el puntero del ratón sobre el cuadrado y el cursor
   egra.

2. el ratón, arrastre el cursor hacia abajo hasta las celdas
   ente.
   área.



3. s las celdas marcadas se rellenarán automáticamente.
   ea marcada se crearán variables nuevas.

Resultado

En función de la celda que estuviera seleccionada, se habrán rellenado automáticamente diferentes propiedades o se habrán creado variables nuevas.

Consulte también

Crear una variable externa (Página 8465)

10.2.2.2 Edición de variables

Editar variables

Introducción

En cualquier momento es posible copiar, borrar o cambiar el nombre de una variable.

Al cambiar el nombre de una variable, el nuevo nombre debe ser unívoco para todo el aparato.

Si se copia una variable en el portapapeles con el comando "Copiar", también se copiarán las referencias vinculadas a la variable.

Si se inserta una variable en otro dispositivo con el comando "Pegar", la variable se insertará sin las referencias vinculadas. De una referencia sólo se insertará el nombre de objeto. Si en el sistema de destino existe una referencia homónima con propiedades válidas, la referencia existente se enlazará con la variable copiada.

Si se copia una variable, también se copiarán los avisos vinculados a la variable. Si la variable copiada se inserta en otro equipo, se insertará junto con los avisos vinculados.

Requisitos

- La variable que debe copiarse, borrarse o a la que debe cambiarse el nombre está creada.
- La tabla de variables está abierta.

Cambiar el nombre a una variable

1. Seleccione el campo "Nombre" de la variable en la tabla de variables.
2. En el menú contextual, seleccione el comando "Cambiar nombre".
3. Introduzca un nuevo nombre.
   La variable aparecerá con el nuevo nombre.

Copiar variable

1. Seleccione una o varias variables en la tabla de variables o en la ventana detallada.
2. En el menú contextual, elija el comando "Copiar".
3. Haga clic en la posición en la que desea insertar la variable. Así, p. ej., haga clic en otra tabla de variables del mismo dispositivo o en la tabla de variables en un segundo dispositivo.
4. En el menú contextual seleccione el comando "Pegar" o "Inserción avanzada". La variable se insertará del modo descrito anteriormente.

Borrar variable

1. Seleccione una o varias variables en la tabla de variables.
2. En el menú "Herramientas", seleccione el comando "Referencias cruzadas". En el editor "Referencias cruzadas" compruebe dónde se utilizan las variables. De este modo se verá cómo quedará afectado el proyecto si se borran las variables.
3. Seleccione el comando "Borrar" del menú contextual de las variables.
   Se borrarán todas las variables marcadas.

Exportación e importación de variables

WinCC ofrece la posibilidad de exportar e importar variables. Con la exportación e importación se pueden exportar variables de un proyecto e importarlas a otro. Además, se pueden crear grandes cantidades de variables fuera de WinCC, editarlas y finalmente importarlas a los proyectos WinCC que se desee. Encontrará más información al respecto en Importar y exportar variables (Página 9182).

Consulte también

Modificación de la configuración de variables (Página 8473)

Configurar diferentes variables simultáneamente (Página 8473)

Utilizar varias variables simultáneamente en una imagen (Página 8474)

Sincronizar variables (Página 8476)

Crear una variable externa (Página 8465)

Importar y exportar variables (Página 9182)

Modificación de la configuración de variables

Introducción

Las variables pueden ajustarse en todo momento a diferentes requisitos del proyecto.

Modificar la configuración de las variables

Si desea modificar la configuración de una variable, abra la tabla de variables que contiene la variable. También puede abrir la tabla de variables "Mostrar todas las variables".

En las tablas de variables es posible, p. ej., comparar y sincronizar las propiedades de varias variables o clasificar las variables según sus propiedades.

Las propiedades pueden modificarse directamente en la tabla o en la ventana de inspección.

Si se modifica una propiedad de una variable y esto provoca contradicciones con otras propiedades, se avisará con un fondo resaltado en color. Esto sucede, p. ej. cuando la variable se conecta con otro controlador que no admite el tipo de datos configurado.

Consulte también

Editar variables (Página 8469)

Configurar diferentes variables simultáneamente

Introducción

En WinCC se asignan simultáneamente propiedades coincidentes a varias variables. Esto facilita una configuración eficiente.

Requisitos

- Las variables que deben configurarse están creadas.
- La tabla de variables está abierta.
- La ventana de inspección está abierta.

Procedimiento

1. Seleccione todas las variables que desee configurar conjuntamente en la tabla de variables. Si las variables coinciden en la propiedad seleccionada, la configuración de esta propiedad aparecerá en la ventana de inspección. Si las propiedades no coinciden, el campo en cuestión permanece vacío.
2. Defina la propiedad conjunta en la ventana de inspección o directamente en la tabla de variables.

Si modifica conjuntamente una propiedad en varias variables, solo se modifica esa propiedad. Las demás propiedades de las variables no sufren cambios.

Resultado

Todas las variables marcadas se configurarán de nuevo.

Si a continuación desea configurar propiedades de variables en las que las variables sean distintas, elimine la selección múltiple.

Consulte también

Editar variables (Página 8469)

Utilizar varias variables simultáneamente en una imagen

Introducción

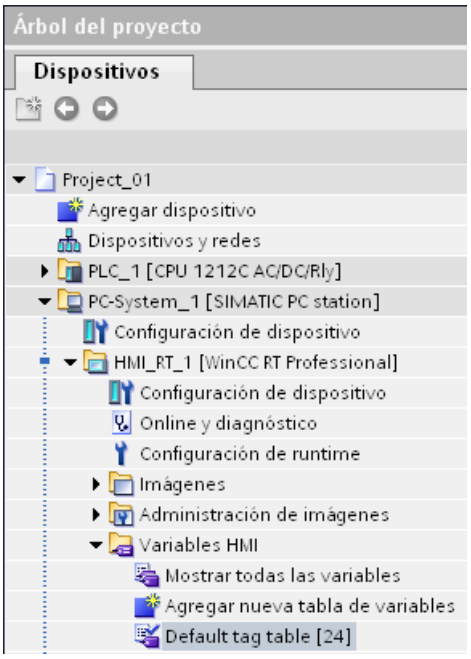
En WinCC es posible crear simultáneamente en una imagen varios campos E/S asociados a variables. Esto facilita una configuración eficiente.

Requisitos

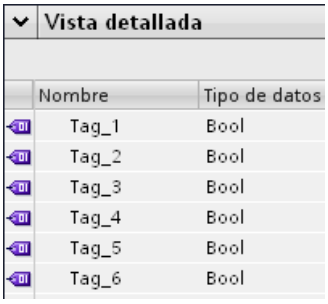
- Se han creado varias variables.
- Hay una imagen abierta.

Procedimiento

1. En la navegación del proyecto, seleccione la tabla de variables deseada en "Variables HMI".



2. Seleccione la ventana detallada en el área inferior del árbol del proyecto. En la ventana detallada se muestran las variables existentes en el grupo de variables seleccionado.



3. Seleccione las variables en la ventana detallada.
4. Utilizando el método de arrastrar y soltar, desplace las variables hasta la imagen. Para cada variable se crea un campo E/S asociado a la misma.

Nota

Si se arrastra una variable PLC desde la ventana detallada hasta el área de trabajo con la función Drag&Drop, se crearán además una red y una conexión en el editor "Dispositivos y redes".

Consulte también

Editar variables (Página 8469)

Sincronizar variables

Introducción

WinCC ofrece las siguientes opciones para sincronizar variables PLC y HMI:

- Sincronizar variables con o sin sincronización de nombre entre PLC y WinCC
  Existen distintas posibilidades.
- Conectar variables con direcciones en el controlador
  Este procedimiento resulta útil, p. ej., cuando se ha modificado la conexión entre el panel de operador y el controlador y, como consecuencia, se han perdido las conexiones de variables. La función puede utilizarse también cuando el programa de control y el proyecto HMI se han configurado por separado.

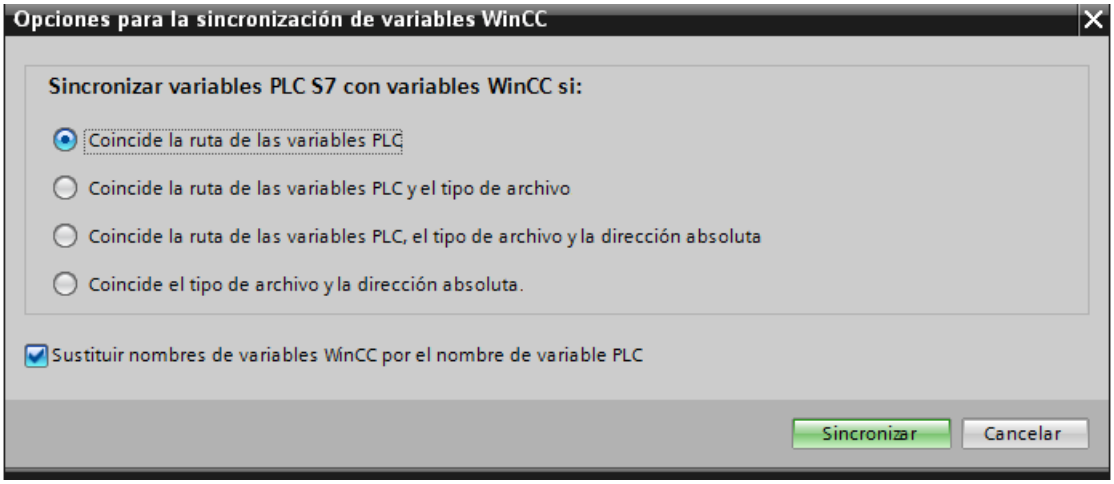
Requisitos

- Se han creado las variables HMI externas.
- Se han creado las variables PLC.
- Se ha creado una conexión HMI con un controlador del proyecto.

Procedimiento

Para sincronizar variables HMI con variables PLC, proceda del siguiente modo:

1. Seleccione en el árbol del proyecto el directorio que contiene las variables en cuestión.
2. Seleccione en el menú contextual "Sincronizar con la variable PLC".
   Se abre un cuadro de diálogo.



3. Seleccione la opción deseada.
   Si desea sincronizar las variables sin sincronización de nombre, desactive "Sustituir nombre de variable WinCC por el nombre de variable PLC".
   Si desea reconectar las variables HMI con acceso absoluto, seleccione "Coincide el tipo de archivo y la dirección absoluta".
4. Confirme haciendo clic en "Sincronizar".
   El sistema busca una variable PLC apropiada de acuerdo con la opción seleccionada.

Resultado

Las variables HMI externas están sincronizadas con las variables PLC.

Si ha seleccionado la opción "Coincide el tipo de archivo y la dirección absoluta", la conexión de variables se establece en cuanto se encuentra una variable PLC adecuada.

Si ha seleccionado otra opción, las variables WinCC se actualizan de forma correspondiente y los nombres de variables PLC se aplican en WinCC.

Consulte también

Editar variables (Página 8469)

10.2.2.3 Configurar variables

Valores límite de una variable

Introducción

Para variables numéricas, el rango de valores puede limitarse con valores límite.

Principio

Para variables numéricas se determina un rango de valores que se define por un valor límite superior y un valor límite inferior.

Si el valor de proceso rebasa por exceso o por defecto el rango de valores, active una lista de funciones. Cuando el operador introduce un valor para la variable fuera del rango de valores configurado, la entrada se rechaza. El valor no se aplica.

Nota

El texto de los avisos analógicos para violación de valores límite se introduce en el editor "Avisos analógicos".

Ejemplo de aplicación

Utilice valores límite, p. ej. para avisar al operador a tiempo de que un valor de una variable está entrando en un rango crítico.

Consulte también

Escalar variable de forma lineal (Página 8483)
Definir el valor inicial de una variable (Página 8480)
Definir el ciclo de adquisición de una variable (Página 8487)
Definir los valores límite de una variable (Página 8478)
Valor inicial de una variable (Página 8479)
Actualización del valor de variable en runtime (Página 8481)
Escala lineal de una variable (Página 8482)
Conectar una variable con otro controlador (Página 8484)
Direccionamiento indirecto de variables (Página 8484)
Direccionar variable indirectamente (Página 8485)
Utilizar una variable para lanzar funciones (Página 8486)
Multiplexado de direcciones (Página 8488)
Configuración del multiplexado de direcciones con direccionamiento absoluto (Página 8490)
Configuración del multiplexado de direcciones con direccionamiento simbólico (Página 8493)
Crear una variable externa (Página 8465)

Definir los valores límite de una variable

Introducción

Para variables numéricas se determina un rango de valores definiendo un valor límite superior y un valor límite inferior.

Además, es posible configurar la ejecución de una lista de funciones para cuando se rebase el rango de valores por exceso o por defecto.

Requisitos

- La variable para la que deben definirse los límites está creada.
- La ventana de inspección con las propiedades de esta variable está abierta.

Procedimiento

Para determinar los valores límite de una variable, proceda del siguiente modo:

1. En la ventana de inspección seleccione "Propiedades > Propiedades > Valores límite". Si desea definir un valor constante para uno de los valores límite, seleccione la entrada "Constante" con el botón . Introduzca un número en el campo pertinente.
   Si desea emplear el valor de una variable para uno de los valores límite, seleccione la entrada "Variable HMI" con el botón . Con la ayuda de la lista de objetos se define la variable para el valor límite.
2. Repita el paso 1 para definir otro valor límite para la variable.

Procedimiento alternativo

También se puede configurar el valor límite superior e inferior directamente en la tabla de variables. Las columnas ocultas se activan con el menú contextual de los títulos de columna.

Configurar la lista de funciones

En caso de rebase por exceso o por defecto del rango de valores se puede configurar una lista de funciones del modo siguiente:

1. Si desea iniciar una lista de funciones al rebasar el rango de valores por defecto, haga clic en la ventana de inspección en "Propiedades > Eventos > Rebase por defecto del mínimo". Cree una lista de funciones allí.
2. Si desea iniciar una lista de funciones al rebasar el rango de valores por exceso, haga clic en la ventana de inspección en "Propiedades > Eventos > Rebase por exceso del máximo". Cree una lista de funciones allí.

Resultado

Para la variable seleccionada se ha determinado un rango de valores que está definido por un valor límite superior y un valor límite inferior. Si se rebasa el rango de valores por exceso o por defecto se procesará una lista de funciones.

Consulte también

Valores límite de una variable (Página 8475)

Valor inicial de una variable

Valor de una variable al inicio de runtime

Puede configurarse un valor inicial para variables numéricas y variables relativas a valores de fecha y hora. Al iniciar runtime, la variable se pone a este valor. De este modo se establece un estado definido de la variable al inicio de runtime.

Con variables externas, el valor inicial se visualiza en el panel de operador hasta que quede sobrescrito por el controlador o por una entrada.

Si no se ha configurado ningún valor inicial, la variable adopta el valor "0" al iniciarse runtime.

En WinCC Runtime Professional, en una variable con el tipo de datos "String" puede indicarse un valor de variable en lugar de un valor inicial. El valor de la variable es multilingüe y se guarda en el editor "Textos del proyecto". Una vez traducido el texto, se muestra en runtime como valor inicial dependiente del idioma.

Ejemplo de aplicación

Un campo E/S puede preasignarse con un valor predeterminado. Defina el valor predeterminado como valor inicial de la variable vinculada al campo E/S.

Consulte también

Definir el valor inicial de una variable (Página 8480)

Valores límite de una variable (Página 8475)

Definir el valor inicial de una variable

Introducción

En WinCC puede configurarse un valor inicial para una variable numérica y una variable relativa a valores de fecha y hora; la variable adoptará el valor al iniciarse runtime.

Requisitos

- La variable para la que debe definirse el valor inicial está creada.
- La ventana de inspección con las propiedades de la variable está abierta.

Procedimiento

Para configurar un valor inicial, proceda del siguiente modo:

1. Seleccione en la ventana de inspección "Propiedades > Propiedades > Valores".
2. Introduzca el "Valor inicial" deseado.

Procedimiento alternativo

También puede configurar el valor inicial directamente en la tabla de variables. Las columnas ocultas se activan con el menú contextual de los títulos de columna.

Resultado

El valor inicial seleccionado para la variable se aplicará al proyecto.

Consulte también

Valores límite de una variable (Página 8475)

Valor inicial de una variable (Página 8477)

Actualización del valor de variable en runtime

Introducción

Las variables contienen valores de proceso que se modifican en el transcurso del runtime. En las variables internas y externas, las modificaciones de valores se disparan de distinta manera.

Principio

Al iniciar el runtime, el valor de una variable es igual a su valor inicial. En runtime se modifica el valor de la variable.

En runtime existen las siguientes posibilidades para modificar el valor de una variable:

- Modificando el valor de una variable externa en el control.
- Realizando una entrada, p. ej. en un campo E/S.
- Ejecutando una función de sistema para modificar valores, p. ej. la función de sistema "FijarValor".

Actualizando el valor de variables externas.

El valor de una variable externa se actualiza del modo siguiente:

- Cíclico durante el funcionamiento
  Si se selecciona el modo de adquisición "Cíclico durante el funcionamiento", la variable se actualizará en runtime mientras se visualice en una imagen. El ciclo de adquisición determina en qué intervalo de tiempo se actualiza el valor de la variable en el panel de operador. La adquisición cíclica se basa en el tiempo de ciclo seleccionado.
- Cíclico continuo
  Si se seleccione al modo de adquisición "Cíclico continuo", la variable se actualizará en runtime de forma continua, aunque no se encuentre en la imagen abierta. Esta configuración se activa p. ej. para variables que tienen una lista de funciones configurada para una modificación de su valor.
  Utilice la opción "Cíclico continuo" sólo para variables que realmente deban actualizarse de forma continua. Las lecturas frecuentes aumentan la carga de la comunicación.
- Bajo demanda
  Si se selecciona el modo de adquisición "Bajo demanda", la variable no se actualizará cíclicamente, sino sólo cuando se solicite, p. ej. con la función de sistema "ActualizarVariable".

Consulte también

Valores límite de una variable (Página 8475)

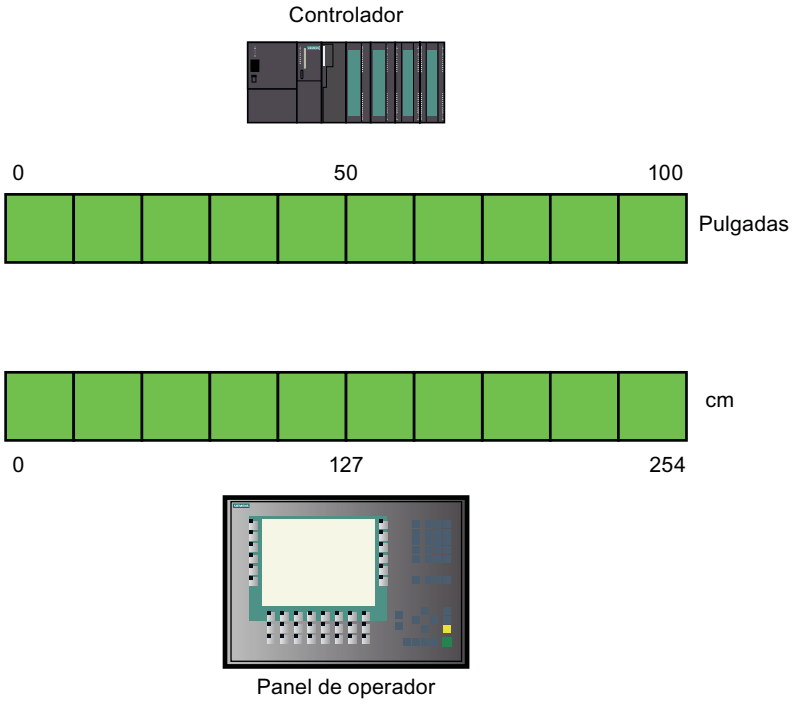
Escala lineal de una variable

Introducción

Los tipos de datos numéricos pueden escalarse de forma lineal. Los valores de proceso de una variable externa existentes en el controlador pueden reproducirse en un rango de valores determinado en el proyecto.

Principio

Para la escala lineal de una variable se indican un rango de valores en el panel de operador y un rango de valores en el controlador. Los rangos de valores se reproducen de forma lineal entre sí.



En cuanto el panel de operador escribe un valor de proceso en una variable externa, se reproduce automáticamente en el rango de valores del control. En cuanto el panel de operador lee un valor de proceso de una variable externa, se realiza una transformación inversa pertinente.

Nota

Para convertir automáticamente valores de proceso también pueden emplearse las funciones del sistema "EscalaLineal" e "InvertirEscalaLineal".

Ejemplo de aplicación

El usuario indica la longitud en cm, pero el controlador espera recibir pulgadas. Los valores introducidos se convierten automáticamente antes de ser transferidos al control. Con la escala lineal se ajusta el rango de valores [0 ... 100] del controlador al rango de valores [0 ... 254] del panel de operador.

Consulte también

Valores límite de una variable (Página 8475)

Escalar variable de forma lineal

Introducción

Para la escala lineal de una variable se indican un rango de valores en el panel de operador y un rango de valores en el controlador. Los rangos de valores se reproducen de forma lineal entre sí.

Requisitos

- La variable externa que desea escalar linealmente está creada.
- La ventana de inspección con las propiedades de esta variable está abierta.

Procedimiento

Para escalar una variable linealmente, proceda de la manera siguiente:

1. En la ventana de inspección seleccione "Propiedades > Propiedades > Escala lineal".
2. Haga clic en "Activar" para activar la escala lineal.
   Con esta opción se desactiva temporalmente la escala lineal, p. ej. para fines de test. Las configuraciones que ya se hayan realizado para la escala lineal se mantienen.
3. En el área "Controlador", introduzca el valor final y valor inicial del rango de valores que debe aplicarse a los valores de proceso del controlador.
4. En el área "Panel de operador", introduzca el valor final y valor inicial del rango de valores que debe utilizarse para los valores de proceso del panel de operador.

Resultado

En runtime, los valores de proceso se reproducirán automáticamente de un rango de valores al otro.

Nota

Para convertir valores de proceso automáticamente también pueden emplearse las funciones de sistema "EscalaLineal" e "InvertirEscalaLineal".

Consulte también

Valores límite de una variable (Página 8475)

Conectar una variable con otro controlador

Introducción

En WinCC es posible modificar en cualquier momento la conexión de una variable con un controlador. Esto es necesario, p. ej., cuando se modifica la configuración de la instalación.

En función del controlador seleccionado, seguidamente debe ajustarse la configuración de la variable. El fondo resaltado en color le indicará qué propiedades de la variable deben modificarse.

Requisitos

- La variable externa cuya conexión desea modificarse está creada.
- Las conexiones con los controladores están creadas.
- La ventana de propiedades de la variable está abierta.

Procedimiento

Para modificar la conexión de una variable con el controlador, proceda del siguiente modo:

1. En la ventana de inspección seleccione "Propiedades > Propiedades > General".
2. En el campo "Conexión" seleccione la nueva conexión.
   Las propiedades de variables que deban ajustarse se resaltarán en color en la tabla de variables y en la ventana de inspección.
3. Ajuste todas las propiedades resaltadas de la variable a los requisitos del nuevo controlador.

Resultado

La variable externa queda conectada con el nuevo controlador.

Consulte también

Valores límite de una variable (Página 8475)

Direccionamiento indirecto de variables

Principio

En el multiplexado, que es un tipo de direccionamiento indirecto, la variable utilizada se determina sólo en runtime. Para la variable multiplexada se define una lista de variables. A

partir de la lista de variables se selecciona la variable relevante en runtime. La selección de la variable depende del valor de la variable de índice.

En runtime se lee primero el valor de la variable indexada. A continuación se accede a la variable que está en la posición pertinente de la lista de variables.

Ejemplo de aplicación

Con el direccionamiento indirecto es posible configurar la siguiente situación:

El usuario selecciona una máquina entre varias de una lista de selección. En función de la selección del usuario, se visualiza un valor de proceso de la máquina en un campo de salida.

Para configurar esta situación se debe configurar la variable indexada en un campo E/S simbólico. La variable multiplexada se configura en un campo E/S. La lista de variables de la variable multiplexada se configura paralelamente al crearse la lista de selección.

Si el operador selecciona otra máquina, se modificará el valor de la variable indexada. A continuación, el campo de selección muestra el contenido de las variables que pertenecen al nuevo valor de índice en la lista de la variable multiplexada.

Consulte también

Valores límite de una variable (Página 8475)

Direccionar variable indirectamente

Introducción

En el direccionamiento indirecto, la variable utilizada se determina únicamente en el tiempo de ejecución. En lugar de una sola variable se define una lista de variables. Las entradas en la lista consisten en un valor de índice y en el nombre de la variable que debe utilizarse. Mediante una variable de índice se controla la entrada de la lista de variables a la que se accederá.

Requisitos

- La variable que debe utilizarse para el direccionamiento indirecto está creada.
- La variable de índices está creada.
- Las variables que deben incluirse en la lista de variables están creadas.
- La ventana de inspección está abierta con las propiedades de variables.

Procedimiento

Para direccionar variables indirectamente, proceda del siguiente modo:

1. En la ventana de inspección seleccione "Propiedades > Propiedades > Multiplexado".
2. Active la opción "Multiplexado" para activar el direccionamiento indirecto.
   Con esta opción se desactiva temporalmente el direccionamiento indirecto, p. ej. para fines de test. Las configuraciones que ya se hayan realizado para el direccionamiento indirecto se mantienen.
3. Seleccione la variable indexada prevista en el campo "Variable indexada" o defina una nueva variable con la ayuda de la lista de objetos.
4. En la lista de variables, haga clic en la primera entrada de la columna "Variables".
5. Seleccione una variable como entrada en la lista o cree una variable nueva con la ayuda de la lista de objetos.
   La entrada de la columna "Índice" se crea automáticamente.
6. Repita el paso 5 para todas las variables que desee incluir en la lista de variables.
7. En caso necesario, modifique el orden de las entradas en la lista con la función Drag&Drop.

Resultado

En runtime se accede dinámicamente a la variable de la lista cuyo índice sea el mismo que el valor de la variable de índice.

Consulte también

Valores límite de una variable (Página 8475)

Utilizar una variable para lanzar funciones

Introducción

Los valores de variables pueden utilizarse como eventos para lanzar una acción en runtime. Para iniciar una acción en runtime, configure una lista de funciones en una variable. Incluya en la lista de funciones una o varias funciones de sistema. La lista de funciones se ejecuta cuando se produce el evento configurado.

Los eventos siguientes están disponibles en una variable:

- Modificación del valor de variables
  La lista de funciones se ejecuta cada vez que se modifica el valor de la variable.
  Si la variable contiene matrices, la lista de funciones se ejecutará cada vez que se modifique un elemento de matriz.
- Rebase por exceso del valor límite superior de las variables
  La lista de funciones se ejecuta cuando se rebasa el valor límite superior.
- Rebase por defecto del valor límite inferior de las variables
  La lista de funciones se ejecuta cuando se rebasa el valor límite inferior.

Requisitos

- La variable cuyo valor desea utilizar como evento está creada.
- La ventana de inspección con las propiedades de esta variable está abierta.

Procedimiento

Para configurar una lista de funciones en una variable, proceda del siguiente modo:

1. En la ventana de inspección seleccione en "Propiedades > Eventos" el evento para el que desea crear una lista de funciones.
   Se visualiza la lista de funciones que pertenece al evento seleccionado.
2. Haga clic en "<Añadir función>". En la segunda columna de la tabla aparecerá un botón de selección.
3. Haga clic en el botón de selección y elija una función de sistema.
4. Defina los valores de los parámetros.

Resultado

Cuando se produzca el evento configurado en runtime se procesará la lista de funciones.

Consulte también

Valores límite de una variable (Página 8475)

Definir el ciclo de adquisición de una variable

Introducción

El valor de una variable externa puede modificarse en runtime mediante el controlador con el que está vinculada la variable. Para que el panel de operador perciba una modificación de valor por parte del controlador, el valor de la variable debe actualizarse en el panel de operador. La actualización se realiza en intervalos regulares, mientras la variable se visualice en la imagen de proceso o se archive. Un ciclo de adquisición controla la actualización regular. La actualización también puede realizarse de forma continua.

Requisitos

- La variable para la que debe definirse un ciclo de adquisición está creada.
- La ventana de inspección con las propiedades de la variable está abierta.

Procedimiento

Para configurar un ciclo de adquisición para una variable, proceda de la manera siguiente:

1. Seleccione en la ventana de inspección "Propiedades > Propiedades > General".
2. Si desea actualizar la variable regularmente, mientras aparezca en la imagen o se archive, seleccione el modo de adquisición "Cíclico durante el funcionamiento".
   O bien:
   Si desea actualizar la variable regularmente, aunque no aparezca en la imagen o no se archive, seleccione el modo de adquisición "Cíclico continuo".
   El ajuste "Cíclico continuo" se activa p. ej. para una variable que tiene configurada una lista de funciones para cuando se modifique un valor y que no es visible directamente en ninguna imagen.
3. Seleccione el tiempo de ciclo deseado en el campo "Ciclo de adquisición" o defina un nuevo ciclo de adquisición con la ayuda de la lista de objetos.

Otra alternativa consiste en configurar el ciclo de adquisición directamente en el área de trabajo de la tabla de variables. Las columnas ocultas se activan con el menú contextual de los títulos de columna.

Nota

Utilice el modo de adquisición "Cíclico continuo" sólo para variables que realmente deban actualizarse de forma continua. Los procesos de lectura continuos aumentan la carga de la comunicación.

Resultado

La variable configurada se actualiza en runtime con el ciclo de adquisición seleccionado.

Consulte también

Valores límite de una variable (Página 8475)

Multiplexado de direcciones

Introducción

En el multiplexado de direcciones se puede acceder con una única variable a un gran número de espacios de memoria en el área de direcciones del controlador. Se puede obtener acceso en lectura y escritura a las direcciones sin tener que definir una variable para cada dirección.

Multiplexado con direccionamiento absoluto

En el multiplexado con direcciones absolutas, las variables se configuran como comodín para la dirección del controlador a la que se accede.

Si p. ej. desea acceder a una dirección del formato "%DBx.DBWy", la expresión para el multiplexado sería el siguiente:

"%DB[HMITag1].DBW[HMITag2]"

A la variable "HMITag1" se le transfiere en runtime el valor deseado para el bloque de datos que se desea direccionar.

A la variable "HMITag2" se le transfiere en runtime la dirección deseada del bloque de datos.

Las variables reciben los datos con ayuda de valores del controlador o a través de un script, por ejemplo.

El multiplexado con direcciones absolutas se soporta con los siguientes controladores y drivers de comunicación.

- SIMATIC S7 300/400
- SIMATIC S7-1200
- SIMATIC S7-1500

El multiplexado con direcciones absolutas no está disponible para bloques de datos con acceso optimizado.

Multiplexado con direccionamiento simbólico

En el multiplexado con direccionamiento simbólico, se accede a un elemento de matriz de una variable de matriz de un bloque de datos del controlador conectado a través de una variable multiplexada y una variable de índice. La variable multiplexada contiene la dirección simbólica del bloque de datos al que se desea acceder. La dirección simbólica contiene por otro lado la variable de índice a través de la que se accede al índice de la variable de matriz.

Si p. ej. desea acceder a la variable de matriz "Arraytag_1" del bloque de datos "Datablock_1", la expresión para el direccionamiento simbólico sería la siguiente:

"Datablock_1.Arraytag_1["HMITag_1"]

Con la variable HMI "HMITag_1" se controla el acceso al índice de los elementos de matriz. A la variable se le transfiere en runtime el índice del elemento de matriz al que se desea acceder en cada caso.

El multiplexado con direccionamiento simbólico solo está disponible si los siguientes componentes soportan el direccionamiento simbólico:

- el panel de operador HMI;
- el controlador;
- el driver de comunicación.

Los drivers de comunicación soportan el direccionamiento simbólico:

- SIMATIC S7-1200
- SIMATIC S7-1500

Consulte también

Valores límite de una variable (Página 8475)

Configuración del multiplexado de direcciones con direccionamiento absoluto

Introducción

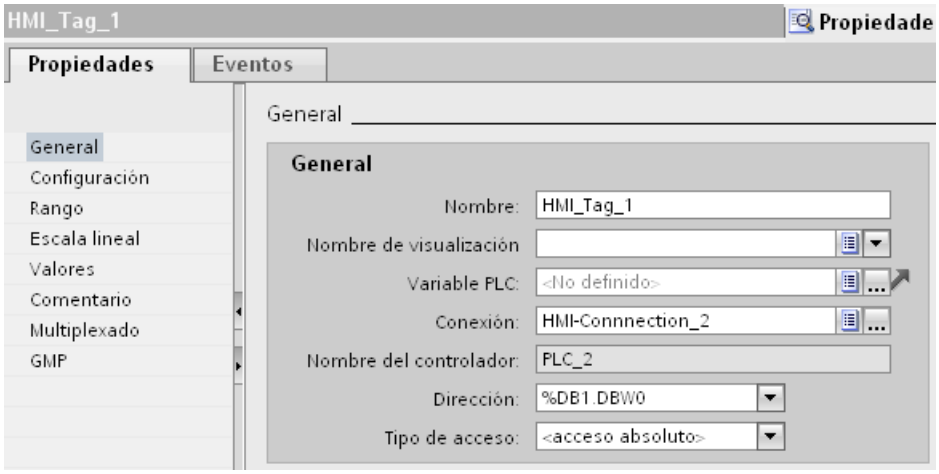
En el multiplexado de direcciones se accede eficientemente con pocas variables a diferentes direcciones del controlador. En vez de la dirección absoluta en el controlador se utilizan variables para poder cambiar la dirección en runtime.

Requisitos

- Se ha creado una variable para el multiplexado de direcciones y se ha asociado al controlador.
- La ventana de propiedades de esta variable está abierta.

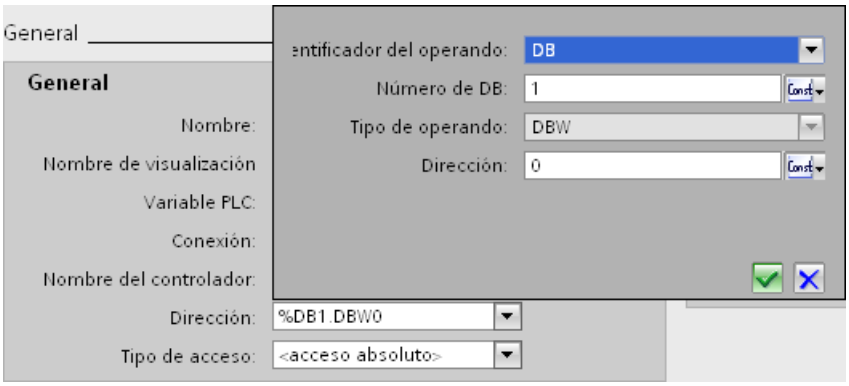
Procedimiento

1. En la tabla de variables seleccione la variable para el multiplexado de direcciones y seleccione luego en la ventana de inspección "Propiedades > Propiedades > General". Se visualizan las propiedades generales de la variable.

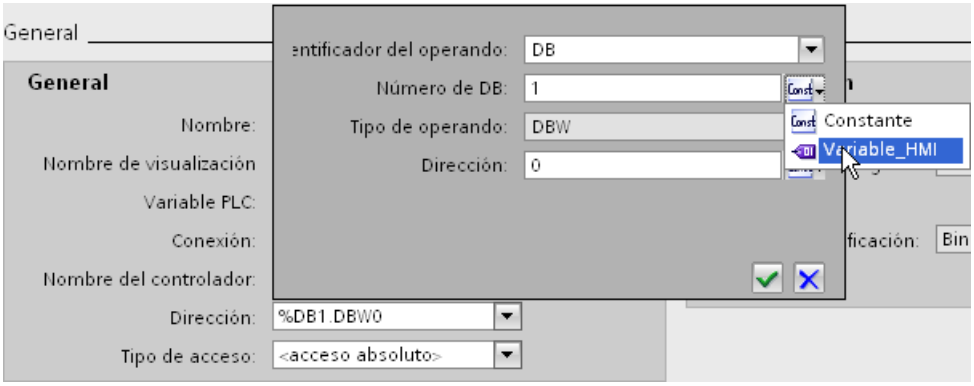


2. Seleccione para este ejemplo el tipo de datos "Int".
3. Seleccione el tipo de acceso "Direccionamiento absoluto".

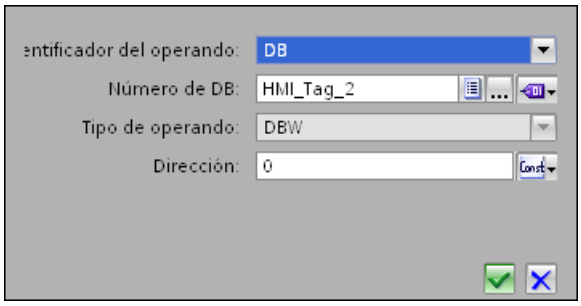
4. Haga clic en el botón de selección del campo "Dirección". Se abrirá el cuadro de diálogo de direcciones.



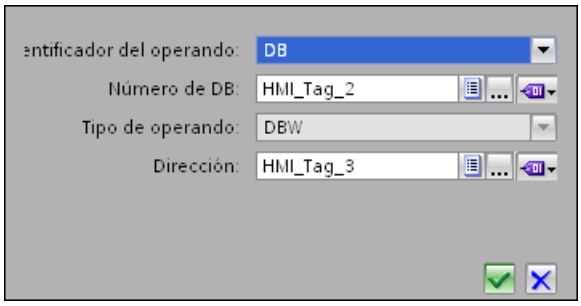
5. Haga clic en el botón de selección del campo "Número de DB" y seleccione la entrada "Variable HMI".



6. En el campo "Número de DB" haga clic en el botón ... y seleccione en la lista de objetos una variable para el número de DB. Otra posibilidad es crear una nueva variable con ayuda de la lista de objetos. Aplique la variable haciendo clic en el botón ✓.



7. Repita los pasos 3 y 4 para el campo "Dirección" y configure una variable adicional para llamar el área de direcciones en el bloque de datos.



Las posibilidades de selección en el cuadro de diálogo de direcciones dependen del tipo de datos seleccionado de la variable multiplexada. El cuadro de diálogo de direcciones ofrece solamente los ajustes de direcciones que se pueden configurar con el tipo de datos seleccionado.

Resultado

En runtime se accederá con la variable multiplex al área de memoria cuya dirección contiene la variable actualmente. Con la variable del campo Número de DB se controla el acceso al bloque de datos. Con la variable del campo "Dirección" se controla el acceso a la dirección en el bloque de datos seleccionado.

Nota

El valor del área de memoria no se leerá hasta el próximo ciclo de actualización de la variable en cuestión.

Si se utiliza p. ej. una variable multiplexada en un script, no debe accederse al contenido del área de memoria justo después de modificarla.

Consulte también

Valores límite de una variable (Página 8475)

Configuración del multiplexado de direcciones con direccionamiento simbólico

Introducción

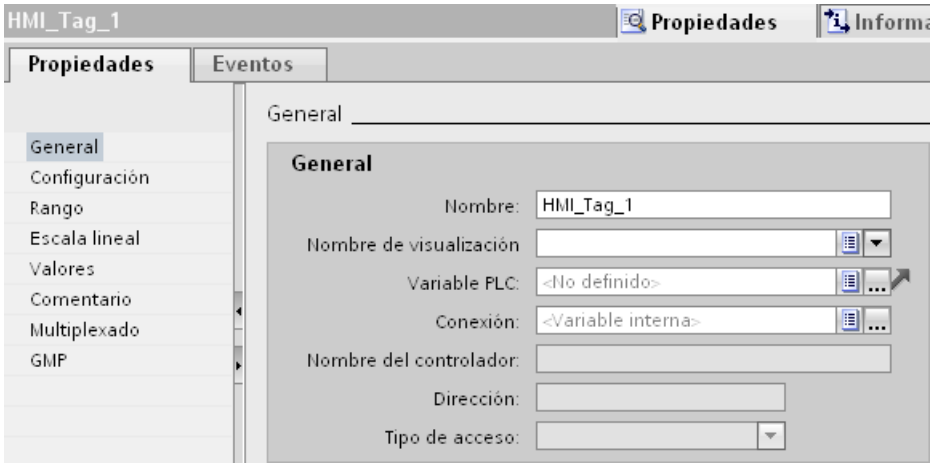
En el multiplexado de direcciones se accede eficientemente con pocas variables a diferentes direcciones del controlador. En vez de la dirección simbólica en el controlador se utilizan variables para poder cambiar la dirección en runtime.

Requisitos

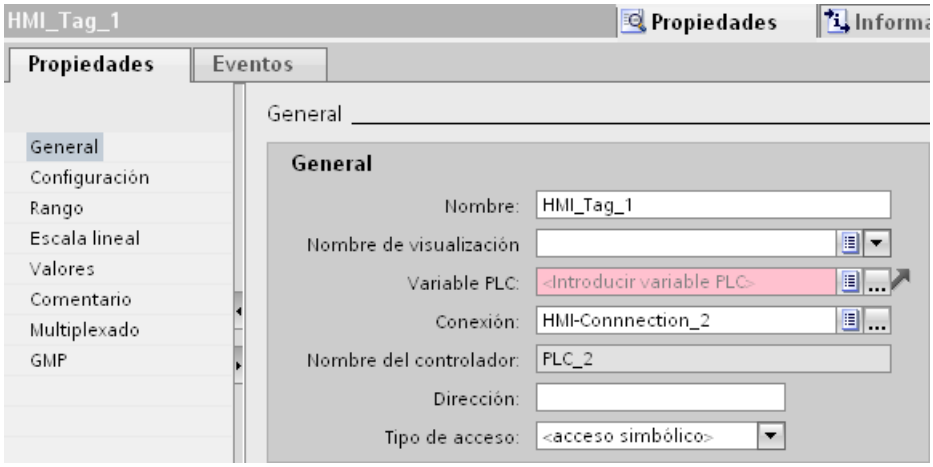
- Se ha creado una variable para el multiplexado de direcciones.
- La ventana de propiedades de esta variable está abierta.
- En el controlador asociado se ha creado un bloque de datos con una variable de matriz.
- Se ha compilado el bloque de datos.

Procedimiento

1. En la tabla de variables seleccione la variable para el multiplexado de direcciones y seleccione luego en la ventana de inspección "Propiedades > Propiedades > General". Se visualizan las propiedades generales de la variable.

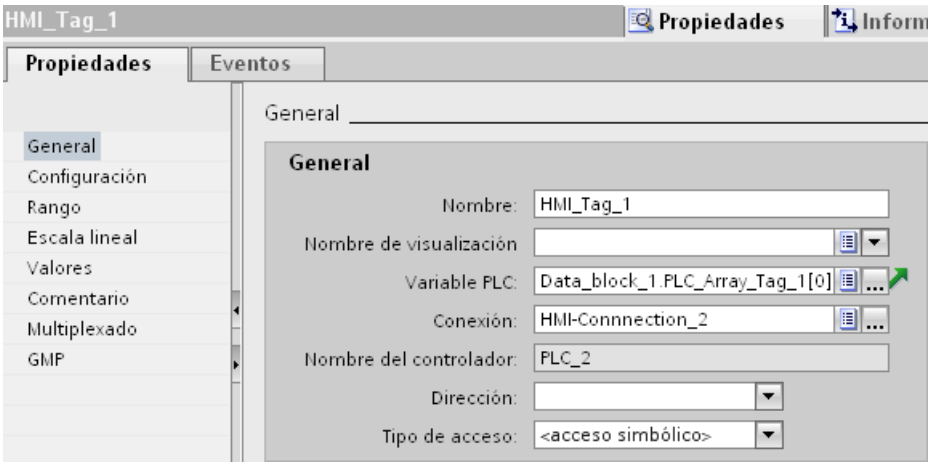


2. En el campo "Conexión" seleccione la conexión con el controlador.

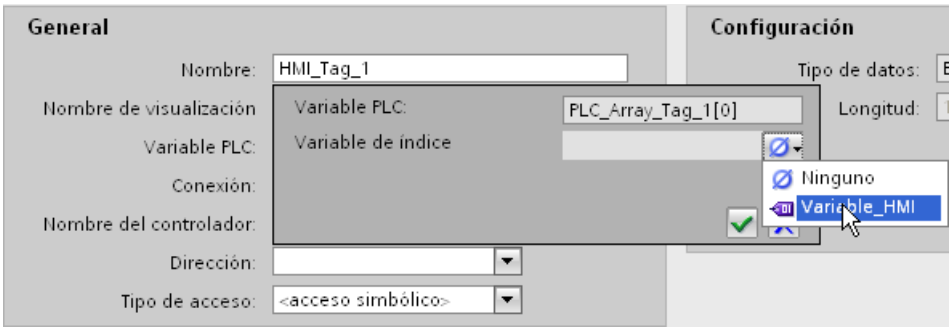


3. Seleccione el tipo de acceso "Direccionamiento simbólico".

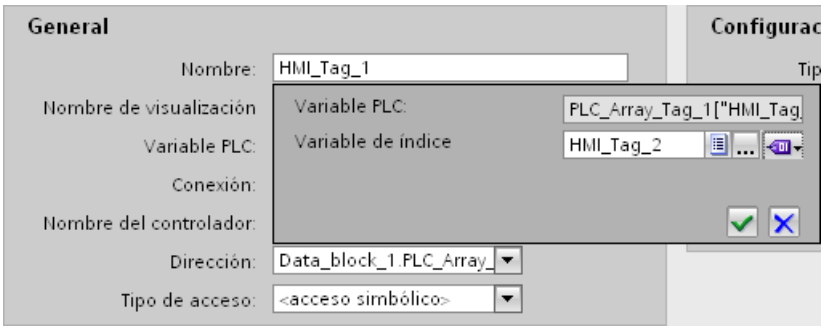
4. Navegue mediante el campo "Variable PLC" al bloque de datos del controlador y seleccione un elemento de la variable de matriz.



5. Haga clic en el botón de selección del campo "Dirección". Se abrirá el cuadro de diálogo de direcciones.
6. Haga clic en el botón de selección del campo "Variable indexada" y seleccione la entrada "Variable HMI".



7. En el campo "Variable indexada" haga clic en el botón ... y seleccione en la lista de objetos una variable para el índice de matriz. Otra posibilidad es crear una nueva variable con ayuda de la lista de objetos. Aplique la variable haciendo clic en el botón ✓.



Resultado

En runtime se accederá al elemento de matriz cuyo valor de índice se encuentra en la variable indexada.

Consulte también

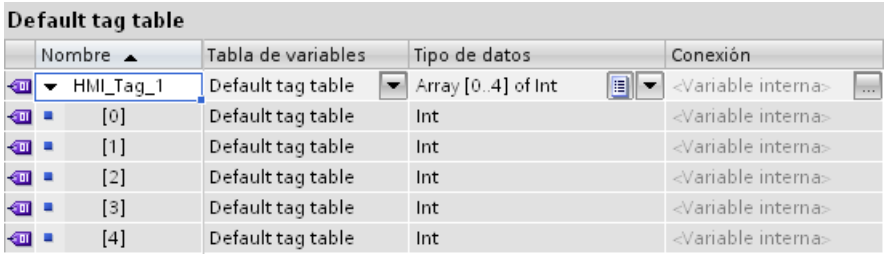
Valores límite de una variable (Página 8475)

10.2.3 Trabajar con matrices

10.2.3.1 Principios básicos de las matrices

Definición

Las matrices permiten ordenar sucesivamente datos de un mismo tipo y direccionarlos de forma consecutiva en el área de direcciones, de modo que se pueda acceder a los datos desde un índice. La matriz utiliza un índice de números enteros para direccionar sus diferentes elementos. Las propiedades de cada uno de los elementos de matriz son iguales y se configuran en la variable de matriz, en un bloque de datos del programa de control.



Default tag table

Nombre	Tabla de variables	Tipo de datos	Conexión
HMI_Tag_1	Default tag table	Array [0..4] of Int	<Variable interna>
[0]	Default tag table	Int	<Variable interna>
[1]	Default tag table	Int	<Variable interna>
[2]	Default tag table	Int	<Variable interna>
[3]	Default tag table	Int	<Variable interna>
[4]	Default tag table	Int	<Variable interna>

Ventajas

Con una sola variable de matriz es posible configurar simultáneamente muchos elementos de matriz con las mismas propiedades. Después, cada elemento de matriz se utiliza en la configuración igual que cualquier otra variable.

Limitaciones

Existen las limitaciones siguientes en el uso de matrices:

- No todos los paneles de operador soportan variables de matriz.
- Una matriz sólo puede contener una dimensión.
- El índice inferior de una matriz debe comenzar por "0".

Ejemplos de aplicación

Las variables de matriz se utilizan en los casos siguientes:

- Para agrupar valores de proceso en curvas de perfil: p. ej. represente valores de proceso en curvas que se hayan registrado en diferentes momentos.
- Para acceder selectivamente a los valores de proceso agrupados en curvas: p. ej. emita todos los valores registrados de la curva de perfil aumentando la variable de índice gradualmente.
- Para configurar avisos de bit con un número consecutivo.
- Para guardar registros de máquina como un todo en una receta.

Normas de licencia para runtime

En WinCC Runtime, una variable de matriz cuenta como 1 PowerTag, independientemente del número de elementos de la matriz.

Particularidades

⚠ ADVERTENCIA

Aumento de la carga del sistema y pérdida de rendimiento

Cuando se accede a un solo elemento de matriz tanto en modo de lectura como de escritura, se leen o escriben siempre todos los elementos de la variable de matriz. Por ello, en el caso de matrices grandes, la transferencia de datos desde y hacia el controlador suele durar más que con un tipo de datos elemental. Esto puede sobrecargar la comunicación y, por tanto, perturbarla.

Ejemplo:

- Se ha configurado una variables de matriz con 100 elementos del tipo de datos "Real".
- Si un elemento de matriz de 4 bytes cambia, se escriben 100 x 4 bytes en el controlador.

⚠ PRECAUCIÓN

Incoherencia de datos en variables de matriz

Si se debe modificar el valor de un elemento en particular en una variable de matriz, se lee toda la matriz, se modifica y se vuelve a escribir como matriz completa. Las modificaciones de otros elementos de matriz que se realicen entretanto en el controlador se sobrescriben al reescribir.

Asegúrese de que posiciones diferentes, p. ej. el panel de operador y el controlador, no puedan escribir valores en la misma variable de matriz al mismo tiempo. Para sincronizar la variable de matriz con el controlador, utilice p. ej. la transferencia síncrona de registros de receta.

Consulte también

Crear una variable de matriz (Página 8498)

Ejemplos de matrices (Página 8499)

Principios básicos de las variables (Página 8457)

10.2.3.2 Crear una variable de matriz

Introducción

Una variable de matriz se crea con objeto de configurar un gran número de variables de un mismo tipo de datos. Los distintos elementos de la matriz se guardan en un área de direccionamiento con direcciones correlativas.

Una variable de matriz puede crearse como variable interna o externa.

Si desea crear una variable de matriz como variable externa, configure primero una variable de matriz en un bloque de datos del controlador conectado. Seguidamente, la variable de matriz se asocia a una variable HMI.

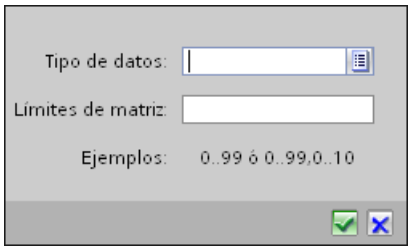
Requisitos

- La tabla de variables HMI está abierta.

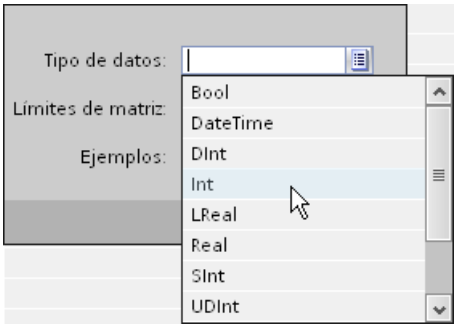
Procedimiento

Para crear una variable de matriz, proceda del siguiente modo:

1. En la tabla de variables HMI, haga doble clic en <Agregar> en la columna "Nombre". Se crea una nueva variable HMI.
2. En la columna Tipo de datos, haga clic en y seleccione el tipo de datos "Array".
3. En la columna Tipo de datos, haga clic en . Se abre el cuadro de diálogo para configurar la matriz.



4. En el campo "Tipo de datos", seleccione el tipo de datos deseado para la variable de matriz.



5. Especifique el número de elementos de matriz en el campo "Límites de matriz". El límite inferior debe comenzar por "0".
6. Haga clic en ✓. Se adoptan los ajustes de la matriz.
7. Guarde el proyecto.

Resultado

Se ha creado una variable de matriz. Las propiedades de los elementos de matriz se adoptan de las variables de matriz de orden superior.

Consulte también

Principios básicos de las matrices (Página 8494)

10.2.3.3 Ejemplos de matrices

Introducción

Las variables de matriz reúnen un gran número de variables (p. ej., 100 elementos de matriz). Las variables de matriz se utilizan como matrices completas en los siguientes puntos:

- En el editor "Avisos"
- En el editor "Recetas"
- En el multiplexado de direcciones
- En el visor de curvas

Los distintos elementos de matriz se utilizan como variables HMI en toda la configuración.

Ejemplos

Si desea utilizar numerosas variables de un mismo tipo de datos, configure sólo una variable de matriz con la respectiva cantidad de elementos.

- Acceda indirectamente a los distintos elementos de matriz, p. ej., a través de una variable de índice multiplexada.
- La variable de índice sirve para controlar y supervisar los elementos de matriz.

Consulte también

Principios básicos de las matrices (Página 8494)

10.2.4 Trabajar con ciclos

10.2.4.1 Principios básicos de los ciclos

Introducción

Los ciclos se utilizan para controlar acciones que se repiten regularmente en runtime. Las aplicaciones clásicas son el ciclo de adquisición y el ciclo de actualización.

Principio

En runtime, las acciones que se repiten periódicamente se controlan con ciclos. Aplicaciones típicas para ciclos:

- Adquisición de variables externas
  El ciclo de adquisición determina cuándo el panel de operador lee el valor de proceso de una variable externa desde el controlador. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico.
  No seleccione un ciclo de adquisición demasiado pequeño porque aumentaría considerablemente la carga de comunicación del proceso.

El valor más pequeño para un ciclo depende del panel de operador para el que se configura. Para la mayoría de paneles de operador, este valor es de 100 ms. Los valores de todos los demás ciclos son siempre un múltiplo entero del valor más pequeño.

Ejemplo de aplicación

Los ciclos se utilizan p. ej. en las tareas siguientes:

- Para actualizar regularmente una variable.
- Para avisar de los intervalos de mantenimiento.

Consulte también

Principios básicos de las variables (Página 8457)

10.2.5 Representar variables

10.2.5.1 Emitir valores de variables en imágenes

Introducción

Los valores de variables se pueden emitir en runtime en las imágenes del panel de operador en forma de curva. Una curva es la representación gráfica de valores que adopta una variable en runtime. Para la representación se utiliza el objeto gráfico "Visor de curvas". El controlador carga los valores de proceso para la representación en forma de curva desde el proceso que se está ejecutando.

Los valores a representar se determinan individualmente en un ciclo fijo ajustable. Las curvas disparadas por ciclos son apropiadas para representar procesos continuos, p. ej. la representación de la temperatura de servicio de un motor.

Valores representados

Para poder visualizar valores de variables en el panel de operador, es preciso configurar un visor de curvas en una imagen. Al configurar el visor de curvas deberá indicar qué valores de variables se deben representar.

La actualización del visor de curvas se controla con la definición del tiempo de ciclo.

10.2.5.2 Configurar el visor de curvas para los valores del controlador

Introducción

Con un visor de curvas se representan gráficamente los valores que adopta una variable en el proceso.

Requisitos

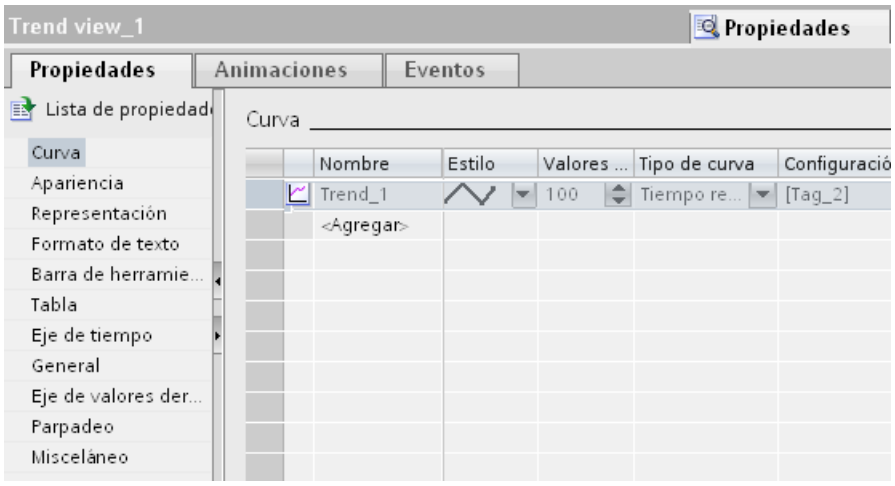
- Hay una imagen abierta.
- La ventana de inspección está abierta con las propiedades para el visor de curvas.

Procedimiento

Para configurar un visor de curvas, proceda del siguiente modo:

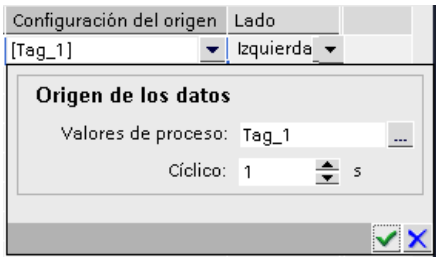
1. Inserte en la imagen el objeto "Visor de curvas" del grupo "Control" de la ventana de herramientas.

2. En el grupo "Propiedades" de la ventana de inspección seleccione la categoría "Curva" y haga doble clic en la entrada "<Agregar>" de la columna "Nombre".



3. Asigne un nombre a la curva en la columna "Nombre".
4. Con el botón de selección de la columna "Estilo" abra el cuadro de diálogo "Estilo" y seleccione la representación de la línea.
5. Elija el número de valores de la curva en la columna "Valores de la curva".

6. Con el botón de selección de la columna "Configuración" abra el cuadro de diálogo "Origen de los datos" y seleccione la variable que deba transferir valores a la curva.
   Indique un ciclo para leer las variables del controlador.



7. Efectúe los demás ajustes necesarios en los cuadros de diálogo de la ventana de inspección. Así, p. ej., active la opción "Mostrar tabla" de la categoría "Tabla" para mostrar una tabla de valores en el visor de curvas.

Nota

Si mantiene pulsada la tecla <Ctrl> y hace doble clic en el visor de curvas, éste se activará. En el modo activo puede modificar el ancho y la posición de las columnas de la tabla de valores en el encabezado de tabla. Para poder activar el visor de curvas, el factor de zoom se deberá ajustar a 100 %.

Resultado

Los valores de proceso de la variable seleccionada se visualizarán en runtime en el visor de curvas configurada.

10.3 Trabajar con avisos

10.3.1 Principios básicos

10.3.1.1 El sistema de avisos de WinCC

Introducción

El sistema de avisos permite visualizar en el panel de operador los estados operativos y las averías que están pendientes o se producen en una instalación.

Un aviso tiene el contenido siguiente, p. ej.:

No	Hora	Fecha	Texto de aviso	Estado	Categoría
5	12:50:24:590	24.02.2007	La presión de la caldera supera el límite superior.	Aparecido Desaparecido	Advertencia: Color rojo

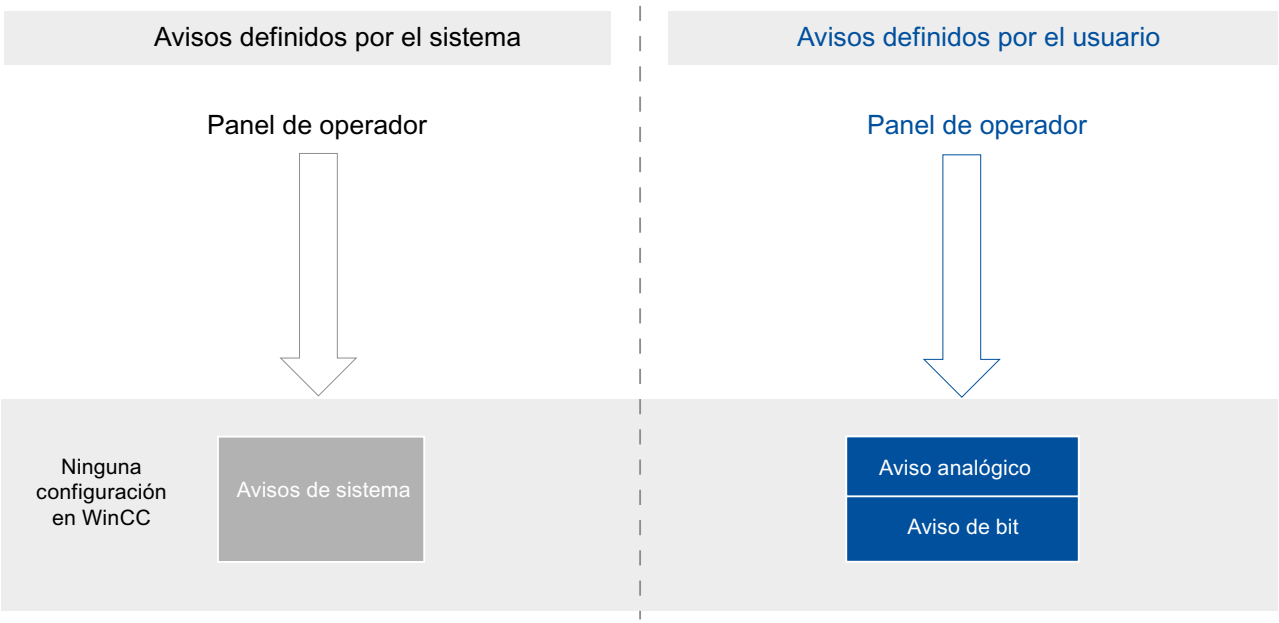
El sistema de avisos de WinCC

El sistema de avisos procesa diferentes procedimientos de aviso. Los procedimientos de aviso se dividen en avisos definidos por el sistema y avisos definidos por el usuario:

- Los avisos definidos por el usuario sirven para vigilar el proceso de la instalación.
- Los avisos definidos por el sistema sirven para supervisar el panel de operador.

Los eventos de aviso registrados se muestran en el panel de operador. Los accesos selectivos a los avisos y la información complementaria acerca de los diferentes avisos garantizan una localización y solución rápida de las averías. Los tiempos de parada se reducen o suprimen por completo.

La figura siguiente muestra la estructura del sistema de avisos:



10.3.1.2 Procedimientos de aviso

Sinopsis de los procedimientos de aviso

Introducción

Los procedimientos de aviso sirven para supervisar la instalación con fines distintos. Los avisos de los diferentes procedimientos se configuran y lanzan de modos distintos.

Los avisos se configuran en función de los diferentes procedimientos en la ficha correspondiente del editor "Avisos HMI".

Procedimientos de aviso en WinCC

WinCC asiste los siguientes procedimientos de aviso:

Avisos definidos por el usuario

- Avisos analógicos
  - Los avisos analógicos sirven para supervisar los valores límite.
- Avisos de bit
  - Los avisos de bit sirven para supervisar los estados.

Avisos definidos por el sistema

- Avisos de sistema
  - Los avisos de sistema pertenecen al panel de operador correspondiente y se importan al proyecto.
  - Los avisos de sistema sirven para supervisar el panel de operador.

Avisos definidos por el sistema

Avisos de sistema

Ejemplos de avisos

- "Se ha establecido correctamente la conexión online con el controlador."

Descripción

Un aviso de sistema muestra el estado del sistema y los errores de comunicación entre el panel de operador y el sistema.

En "Configuración de runtime > Avisos" se determina el tiempo durante el que se visualizará un aviso de sistema en el panel de operador.

Soporte

En la referencia encontrará una lista de los posibles avisos de sistema con la causa y su posible solución. Si acude al soporte online debido a un aviso de sistema del panel de operador, necesitará el número de aviso y las variables existentes del aviso de sistema.

Avisos definidos por el usuario

Avisos analógicos

Descripción

Los avisos analógicos muestran las violaciones de valores límite durante el proceso en curso.

Ejemplo

La velocidad de la mezcladora en una planta de producción de zumos no puede ser ni demasiado alta ni demasiado baja. Para vigilar la velocidad de la mezcladora se configuran los avisos analógicos correspondientes. Si se rebasa el límite de velocidad de la mezcladora por exceso o por defecto, el panel de operador mostrará un aviso con el texto siguiente, p. ej.: "Velocidad de la mezcladora demasiado baja."

Avisos de bit

Descripción

Los avisos de bit muestran un estado durante el proceso en curso.

Ejemplo

Un planta de producción de zumos tiene varios depósitos con los ingredientes. Para obtener la proporción de mezcla correcta de agua, concentrado de zumo, azúcar y aroma, las válvulas de entrada deben abrirse y cerrarse en el momento adecuado. Este proceso debe vigilarse.

Para ello, se configura un aviso de bit para cada estado de las válvulas. Cuando se abre o se cierra una válvula de uno de los cuatro depósitos se muestra un aviso, p. ej. "Válvula de agua cerrada".

De este modo el usuario vigila si la planta produce correctamente.

10.3.1.3 Estados de aviso

Introducción

Un aviso adopta diferentes estados en runtime. En función de los estados de aviso, el usuario analiza y registra el desarrollo del proceso.

Descripción

Cada aviso tiene un estado determinado. Los estados de aviso se componen de los eventos siguientes:

- Aparecido
  La condición para lanzar un aviso se cumple. El aviso se visualiza, p. ej. "Presión de caldera excesiva".
- Desaparecido
  La condición para lanzar un aviso ya no se cumple. El aviso ya no se visualiza porque la caldera se ha despresurizado.
- Acusar
  El usuario acusa el aviso.

Avisos de acuse no obligado

La tabla siguiente muestra los estados de los avisos de acuse no obligado:

Estado	Descripción
Aparecido	La condición de un aviso se cumple.
Desaparecido	La condición de un aviso ya no se cumple.

Avisos de acuse obligado

La tabla siguiente muestra los estados de los avisos de acuse obligado:

Estado	Descripción
Aparecido	La condición de un aviso se cumple.
Desaparecido, no acusado	La condición de un aviso ya no se cumple. El usuario no ha acusado el aviso.
Desaparecido, después acusado	La condición de un aviso ya no se cumple. El usuario ha acusado el aviso después de este momento.
Aparecido, acusado	La condición de un aviso se cumple. El usuario ha acusado el aviso.
Desaparecido, previamente acusado	La condición de un aviso ya no se cumple. El usuario ha acusado el aviso cuando la condición todavía se cumplía.

Cada aparición de uno de estos estados se muestra en el panel de operador.

Nota

El texto visualizado para los estados de un aviso depende del idioma y de la configuración.

10.3.1.4 Categorías

Principios básicos de las categorías

Introducción

En una instalación se producen muchos avisos de distinta importancia. Para que el usuario vea claramente qué avisos son los más importantes, clasifique los avisos del proyecto en categorías.

Descripción

La categoría define la representación de un aviso. La categoría determina si el usuario debe acusar los avisos de esta categoría y cómo.

En WinCC se crea una categoría nueva como categoría de acuse obligatorio.

Nota

La selección de los tipos de representación para las categorías depende de las posibilidades del panel de operador.

Ejemplos de aplicación de las categorías

- El aviso "Velocidad del ventilador 1 en el rango de tolerancia superior" pertenece a la categoría "Warnings". El aviso de representa con fondo blanco. El aviso no es de acuse obligado.
- El aviso "La velocidad del ventilador 2 ha rebasado por exceso el rango de advertencia superior" está asignado a la categoría "Errors". Se representa sobre fondo rojo y parpadea rápidamente en runtime. El aviso es visualizado hasta que el usuario lo acusa.

Utilización de categorías

Para definir el concepto de acuse y la representación de los avisos en el proyecto, utilice las categorías siguientes:

- Categorías predefinidas
  Las categorías predefinidas no pueden borrarse y sólo pueden editarse con limitaciones. Para cada panel de operador se han creado categorías predefinidas en "Avisos HMI > Categorías".
- Categorías personalizadas
  Las categorías se definen de nuevo en "Avisos HMI > Categorías", donde también se configura la representación deseada de avisos y un concepto de acuse para los avisos de esta categoría. El número posible de categorías personalizadas depende del runtime utilizado en el proyecto.

Consulte también

Crear categorías (Página 8514)

Categorías predefinidas

Categorías predefinidas

Las categorías siguientes ya están creadas en WinCC para cada panel de operador:

Categorías de avisos personalizados

- "Warnings"
  La categoría "Warnings" está prevista para mostrar estados regulares y desarrollos del proceso. El usuario no acusa los avisos de esta categoría.
- "Errors"
  La categoría "Errors" está prevista para mostrar estados críticos o peligrosos o bien la violación de valores límite del proceso. El usuario debe acusar los avisos de esta categoría.

Categoría de avisos definidos por el sistema

- "Sistema"
  La categoría "Sistema" contiene avisos que muestran los estados del panel de operador y de los controladores.

Consulte también

Crear categorías (Página 8514)

10.3.1.5 Acuse

Acuse de avisos

Introducción

Para asegurarse de que el usuario de una instalación ha leído un aviso, éste debe configurarse de modo que se visualice hasta que el usuario lo haya acusado. Los avisos que muestran estados críticos o peligrosos en el proceso, deben ser de acuse obligatorio.

Descripción

Con el acuse de un aviso cambia el estado del aviso de "Aparecido" a "Acusado". Con el acuse de un aviso el usuario confirma el procesamiento del estado que ha lanzado el aviso.

Lanzar el acuse de un aviso

El operador dispara el acuse de un aviso en runtime de modos diversos:

- Acuse por parte del usuario autorizado en el panel de operador
- Acuse automático por parte del sistema sin acción del operador, p. ej. con
  - Variables
  - Controlador
  - Funciones de sistema en listas de funciones

Acusar conjuntamente avisos relacionados

Para tener una visión más clara del sistema de avisos en runtime y para que sea más fácil de manejar, configure un grupo de avisos en caso necesario. El usuario acusa con una acción todos los avisos pertenecientes al grupo de avisos.

Acuse por el controlador

Dado el caso, el controlador acusa automáticamente los avisos de bit. El acuse se lanza desde un bit de la "Variable de acuse del controlador". El bit y la variable se definen durante la configuración.

Acuse de un aviso en el panel de operador

El usuario acusa un aviso en runtime en función de la configuración, del siguiente modo:

- Con la tecla de acuse <ACK> en el panel de operador
- Con el botón del visor de avisos
- Con las teclas de función configuradas o los botones de las imágenes

Nota

Tecla de acuse <ACK> en el panel de operador

Para que los avisos críticos sean procesados únicamente por usuarios autorizados, proteja la tecla "ACK" de los paneles de operador, así como los objetos de manejo e indicadores de los avisos. Para ello, utilice una autorización de uso apropiada.

Nota

Dependencia del panel de operador

La tecla de acuse <ACK> no está disponible en todos los paneles de operador.

Concepto de acuse

Resumen

El concepto de acuse se define para una categoría. Los avisos asignados a esta categoría se acusan de acuerdo con el concepto de acuse definido. En WinCC existen los conceptos de acuse siguientes:

- Aviso sin acuse
  Este aviso aparece y desaparece sin que se requiera ningún acuse. El sistema no muestra ninguna reacción.
- Aviso con acuse simple
  Este aviso requiere un acuse en cuanto se produce el evento que lanza el aviso. El aviso está pendiente hasta que se acusa.

10.3.1.6 Grupos de avisos

Introducción

En una instalación se producen muchos avisos de distintas zonas y procesos. Los grupos de avisos agrupan avisos relacionados.

Grupos de avisos

Con ayuda de los grupos de avisos se observan partes de la instalación y se acusan conjuntamente los avisos correspondientes cuando se producen.

Dado el caso, los grupos de aviso contienen avisos de distintas categorías. Asigne sólo avisos de acuse obligado a un grupo de avisos.

Utilizar grupos de avisos

La formación de grupos de avisos es aconsejable para los avisos siguientes, p. ej.:

- Avisos que se originan con el mismo fallo.
- Avisos similares
- Avisos de una unidad de máquina, p. ej. "Fallo accionamiento XY"
- Avisos de una parte relacionada del proceso, p. ej. "Fallo suministro de agua de enfriamiento"

Representación en runtime

En runtime, la columna "Grupo de avisos" muestra el número del grupo al que pertenece el aviso.

10.3.1.7 Número de aviso

Asignación de números de aviso

El sistema asigna números de aviso unívocos dentro de un procedimiento de aviso.

Nota

Cuando se adaptan números de aviso, es preciso prestar atención a la univocidad de los números en todo el procedimiento de aviso.

10.3.2 Trabajar con avisos

10.3.2.1 Componentes y propiedades de los avisos

Sinopsis

En WinCC se configuran los componentes de los avisos. La tabla siguiente muestra los componentes básicos de los avisos:

Categoría	Número de aviso	Hora	Fecha	Estado de aviso	Texto de aviso	Grupo de aviso	Tooltip	Variable de disparo	Valor límite
Warning	1	11:09:14	06.08.2007	AD	Velocidad máxima alcanzada.	2	Este aviso es...	speed_1	27
Sistema	110001	11:25:58	06.08.2007	A	Cambio al modo de operación "Online"	0	Este aviso es...	Variable PLC_1	–

Categoría

Las categorías son "Warnings" o "Errors", p. ej. La categoría define lo siguiente para un aviso:

- Concepto de acuse
- Representación en runtime (p. ej. color)

Número de aviso

Un aviso se identifica con un número unívoco. Dicho número es asignado por el sistema. El número de aviso cambia en caso necesario, p. ej. para identificar con un número correlativo avisos relacionados dentro del proyecto.

Hora y fecha

Cada aviso muestra en un sello de tiempo la hora y fecha en la que se lanzó el aviso.

Estado de aviso

Un aviso tiene los eventos "Aparecido", "Desaparecido" y "Acusar". Para cada evento se emite un aviso nuevo con el estado actual del aviso.

Texto de aviso

El texto del aviso describe la causa del mismo.

El texto del aviso puede contener campos de salida para valores actuales. En función del runtime utilizado podrá insertar unos valores u otros. El valor se registra en el instante en el que cambia el estado del aviso.

Grupo de avisos

El grupo de avisos agrupa avisos distintos.

Tooltip

En caso necesario, para cada aviso se configura un tooltip propio, que el usuario puede visualizar en runtime.

Variable de disparo

Cada aviso tiene asignado una variable como disparador. El aviso se visualiza si la variable de disparo cumple la condición establecida, p. ej. cuando modifica el estado o rebasa por exceso un valor límite.

Valor límite

Los avisos analógicos muestran violaciones de valores límite. Dependiendo de la configuración, WinCC visualiza el aviso analógico tan pronto como la variable de disparo rebasa por exceso o por defecto el valor límite configurado.

10.3.2.2 Configuración de avisos

Resumen de la configuración de avisos

Pasos necesarios para configurar los avisos

Los avisos se configuran en WinCC siguiendo los pasos siguientes:

1. Editar y crear categorías
   Con la categoría se define la representación de un aviso en runtime y su concepto de acuse.
2. Crear variables en el editor "Variables HMI"
   - Configure las variables para su proyecto.
   - Cree valores de rango para las variables.
3. Crear avisos en el editor "Avisos HMI"
   - Se crean avisos definidos por el usuario y se les asignan la variable que debe vigilarse, las categorías, los grupos y otras propiedades.
   - Dado el caso, se asignan funciones de sistema o scripts a los eventos de los avisos.
4. Configurar la visualización de avisos
   Para visualizar los avisos configurados, configure un visor de avisos o una ventana de avisos en el editor "Imágenes".

Pasos de configuración avanzados

En función de los requisitos del proyecto se requerirán más pasos para configurar los avisos:

1. Crear grupos de avisos
   En los grupos de avisos se ordenan los avisos del proyecto según su relación, p. ej. según la causa del error (p. ej. fallo de alimentación) o el origen del error (p. ej. motor 1).
2. Configurar Loop-In-Alarm
   Para poder cambiar a la imagen que contiene información sobre el error ocurrido, tras aparecer el aviso correspondiente se configura una Loop-In-Alarm.

Crear categorías

Introducción

Las categorías se crean en la ficha "Categorías" del editor "Avisos HMI". Para cada proyecto ya están creadas algunas categorías predefinidas. En caso necesario, se crean otras categorías personalizadas. En total se pueden crear 32 categorías como máximo.

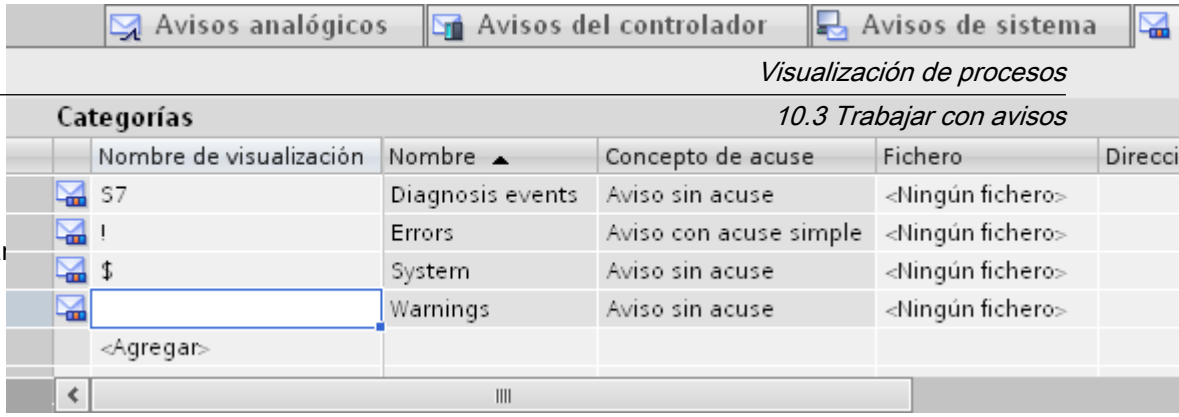
Requisitos

- El editor "Avisos HMI" está abierto.
- La ventana de inspección está abierta.

Procedimiento

Pa

1.



1. Haga doble clic en "<Agregar>" en la tabla.
   Se creará una nueva categoría. A cada categoría nueva se asigna automáticamente una ID fija.
   Las propiedades de la nueva categoría se visualizarán en la ventana de inspección.
2. En la ventana de inspección configure la categoría en "Propiedades > Propiedades > General".
   - Introduzca un "Nombre" y un "Nombre de visualización".
   - Dependiendo del panel de operador, en caso necesario también será preciso activar la archivación o la transmisión automática de correos electrónicos.
3. Defina el concepto de acuse de la categoría en "Propiedades > Propiedades > Acuse" de la ventana de inspección.
4. Modifique el texto predeterminado en "Propiedades > Propiedades > Estado" de la ventana de inspección.
   Con este texto se mostrará en runtime el estado de un aviso.
5. Modifique los colores predeterminados en "Propiedades > Propiedades > Colores" de la ventana de inspección. Dependiendo del panel de operador, también puede ser preciso modificar la intermitencia.

Estos ajustes determinan la representación de los avisos de esta categoría en runtime.

Nota

Para representar categorías en color en runtime, debe haberse activado la opción "Utilizar colores de las categorías". Para ello, active "Configuración de runtime > Avisos > General > Utilizar colores de las categorías" en el árbol del proyecto. En un proyecto nuevo de WinCC, esta opción está activada.

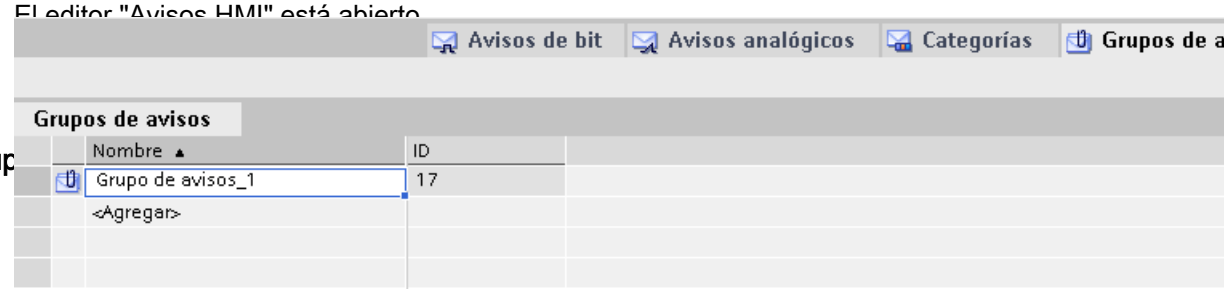
Configurar grupos de avisos

Introducción

Los grupos de avisos se crean en la ficha "Grupos de avisos" del editor "Avisos HMI". El grupo de avisos es una agrupación de diferentes avisos. Los avisos de un grupo se ordenan según su relación, p. ej. causa del error u origen del error. Si en runtime se acusa un aviso de un grupo, también se acusarán automáticamente todos los demás avisos del mismo grupo.

Requisitos

- Se ha creado un proyecto.
- El editor "Avisos HMI" está abierto.
-



Crear un nuevo grup

1.
2. Haga doble clic en "<Agregar>" en la primera fila libre de la tabla del área de trabajo.
   Se creará un nuevo grupo.
3. Sobrescriba el "Nombre" propuesto en caso necesario.

Resultado

Se ha creado un grupo de avisos. Para acusar conjuntamente avisos en runtime, asigne a un grupo los avisos de acuse obligado relacionados.

Configurar avisos de bit

Introducción

Los avisos de bit muestran cambios de estado en una instalación y son lanzados por el controlador. Indican, p. ej., si una válvula está abierta o cerrada.

A continuación se describe la configuración en el editor "Avisos HMI". También es posible configurar avisos de bit en el editor "Variables HMI".

Requisitos

- El editor "Avisos HMI" está abierto.
- La ventana de inspección está abierta.
- Las categorías y los grupos de avisos necesarios están creados.

Procedimiento

Para configurar un aviso de bit, proceda del siguiente modo:

1. Abra la ficha "Avisos de bit".
2. Para crear un aviso de bit, haga doble clic en "<Agregar>" en el área de trabajo.
   Se crea un aviso de bit.

3. En la ventana de inspección configure el aviso en "Propiedades > Propiedades > General":
   - Introduzca el texto de aviso deseado como texto de evento.
     Utilice las funciones del menú contextual para formatear el texto carácter por carácter y para insertar campos de salida para variables HMI o textos de listas.
   - Modifique el número del aviso en caso necesario.
   - Seleccione la categoría y, dado el caso, el grupo de avisos.
4. En la ventana de inspección, seleccione la variable y el bit que debe disparar el aviso en "Propiedades > Propiedades > Disparador". Tenga en cuenta lo siguiente:
   - Si elige una variable HMI, utilice los tipos de datos "Int" o "UInt".
   - Si elige una variable PLC, utilice los tipos de datos "Int" o "Word". Las áreas de entrada y de salida de una variable PLC no son adecuadas como disparador.
   - Utilice un bit de una variable de disparo sólo para un aviso.
   - Utilice una variable de disparo únicamente como variable de disparo.
   - Si el aviso debe acusarse desde el controlador, utilice también esta variable como variable de acuse del PLC.

Nota

Al indicar el bit, tenga en cuenta el modo de contaje de los bits en el controlador utilizado. Encontrará más información al respecto en el apartado "Comunicación" de la Ayuda en pantalla del controlador utilizado.

Nota

Si el objeto deseado todavía no existe en la selección, créelo directamente en la lista de objetos y modifique sus propiedades más adelante.

Textos de aviso dependientes del estado

Para mostrar otro texto en función del estado del aviso, vincule una lista de textos al texto de aviso. La lista de textos se controla a través de una variable.

Ajustes adicionales para avisos de bit

Crear un tooltip

Para dotar al aviso de un tooltip, proceda del siguiente modo:

- Introduzca el texto deseado en "Propiedades > Propiedades > Tooltip".

Configurar tareas controladas por eventos

Para configurar tareas controladas por eventos, p. ej., un "Loop-In-Alarm", proceda del siguiente modo:

1. Seleccione el aviso de bit.
2. En la ventana de inspección seleccione "Propiedades > Eventos" y configure una lista de funciones para el evento deseado.

Consulte también

Configurar una Loop-in-Alarm (Página 8522)

Configurar avisos analógicos

Introducción

Loa avisos analógicos muestran violaciones de valores límite. Si, p. ej., la velocidad de un motor baja por debajo de un valor determinado, se lanza un aviso analógico.

Requisitos

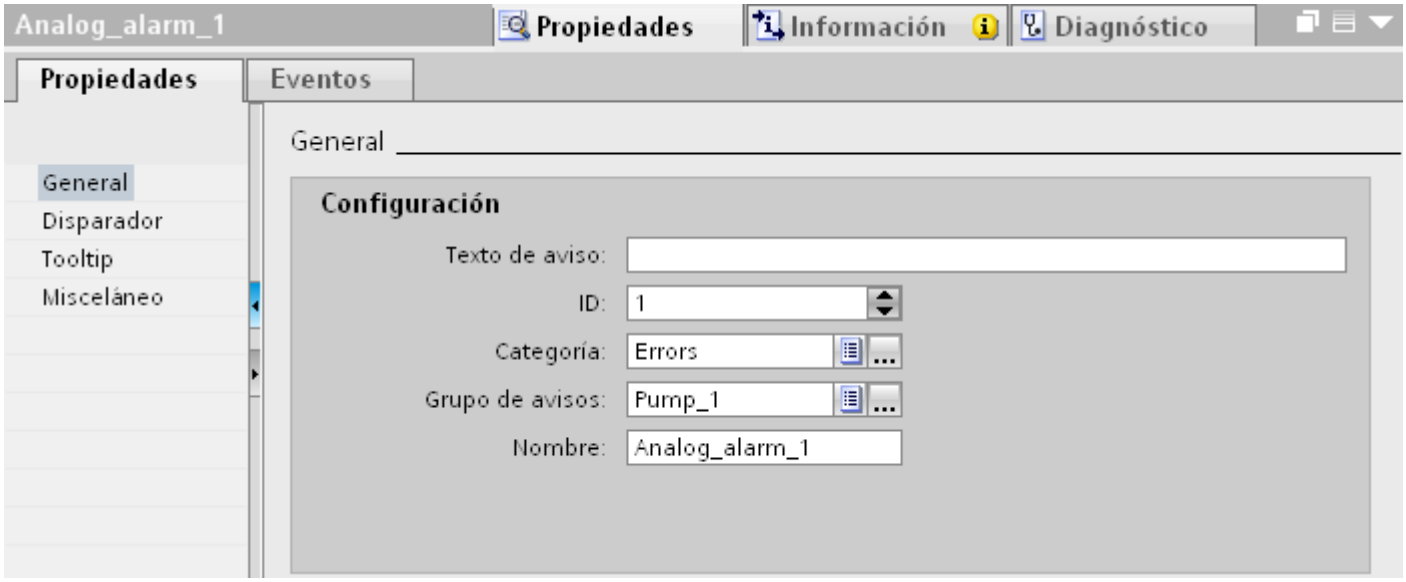
- El editor "Avisos HMI" está abierto.
- La ventana de inspección está abierta.
- Las categorías y los grupos de avisos necesarios están creados.

Procedimiento

Para configurar un aviso analógico, proceda del siguiente modo:

1. Haga clic en la ficha "Avisos analógicos".
2. Para crear un aviso analógico, haga doble clic en "<Agregar>" en la tabla.
   Se creará un nuevo aviso analógico.

3. En la ventana de inspección configure el aviso en "Propiedades > Propiedades > General":
   - Introduzca el texto de aviso deseado como texto de evento.
     En caso necesario, formatee el texto carácter por carácter desde el menú contextual.
     Si es necesario, utilice el menú contextual para insertar campos de salida para variables HMI o textos desde listas.
   - Modifique el número del aviso en caso necesario.
   - Seleccione la categoría y, dado el caso, el grupo de avisos.



4. Configure la variable que dispara el aviso en "Propiedades > Propiedades > Disparador".
   Utilice una variable de disparo únicamente como variable de disparo.

Configurar valores límite para un aviso analógico

1. Haga clic en el botón en "Propiedades > Propiedades > Disparador > Límite > Valor" de la ventana de inspección.
   - Si desea utilizar una constante como valor límite, elija "Constante".
     Introduzca el valor límite deseado.
   - Si desea utilizar una variable como valor límite, elija "Variable_HMI".
     Aparece el botón. Seleccione allí la variable deseada.

   Nota

   Si la variable deseada todavía no existe en la selección, créela en la lista de objetos y modifique sus propiedades más adelante.

2. Seleccione el modo:
   - "Rebase por exceso": el aviso se disparará cuando el valor límite se rebase por exceso.
   - "Rebase por defecto": el aviso se disparará cuando el valor límite se rebase por defecto.

Ajustes opcionales para avisos analógicos

Ajustar un tiempo de retardo

Para ajustar un tiempo de retardo, proceda del siguiente modo:

- Introduzca un intervalo en "Propiedades > Propiedades > Disparador > Configuración > Retardo" de la ventana de inspección.
  El aviso sólo se lanzará si la condición de disparo sigue cumpliéndose una vez transcurrido el tiempo de retardo.

Configurar una zona muerta

Nota

Si un valor de proceso oscila alrededor del valor límite, el aviso correspondiente a este fallo se dispara varias veces. Para evitar esta situación, configure una zona muerta o bien un tiempo de retardo.

Para ajustar la zona muerta, proceda del siguiente modo:

1. En "Propiedades > Propiedades > Disparador > Zona muerta > Modo" seleccione con qué modificación del estado de aviso se debe tener en cuenta la zona muerta.
2. Introduzca un valor constante en "Valor".
3. Active la opción "en %" para establecer que el valor de la zona muerta se indique como porcentaje del valor límite.

Crear un tooltip

Para dotar al aviso de un tooltip, proceda del siguiente modo:

- En la ventana de inspección seleccione "Propiedades > Propiedades > Tooltip" e introduzca el texto deseado.

Configurar tareas controladas por eventos

Para configurar tareas controladas por eventos, p. ej. un "Loop-In-Alarm", proceda del siguiente modo:

1. Seleccione el aviso analógico.
2. En la ventana de inspección seleccione "Propiedades > Eventos" y configure una lista de funciones para el evento deseado.

Consulte también

Configurar una Loop-in-Alarm (Página 8522)

Insertar un campo de salida en el texto del aviso

Introducción

En WinCC, en el texto de aviso se insertan campos de salida que muestran el contenido de las variables.

Requisitos

- El editor "Avisos HMI" está abierto.
- El aviso está seleccionado.

Visualizar el valor de una variable en el texto de aviso

Para insertar un campo de salida para el valor de una variable, proceda del siguiente modo:

1. Sitúe el cursor en la posición deseada del texto de evento.
2. En el menú contextual, elija "Insertar campo de salida de variables".
3. Abra la lista de objetos en "Variable" y seleccione la variable deseada.
   Dado el caso, se puede crear la variable en la lista de objetos.
4. Determine en "Formato" la longitud del campo de salida y el formato en el que se visualizará el valor de la variable en el texto de aviso.
   Configure el campo de salida con un tamaño suficiente. De lo contrario, el contenido de las variables en el aviso no se visualiza por completo.
5. Haga clic en el símbolo para guardar las entradas.

WinCC inserta un comodín en el texto de aviso para el campo de salida: "<tag: n, [nombre de variable]>" donde n = longitud del texto en caracteres.

Modificar las propiedades de un campo de salida

Para modificar las propiedades de un campo de salida, proceda del siguiente modo:

- Haga doble clic en el campo de salida del texto de aviso y modifique los ajustes como desee.

Borrar un campo de salida del texto de aviso

Para borrar un campo de salida del texto de aviso, proceda del siguiente modo:

- Seleccione el campo de salida en el texto de aviso y elija "Borrar" en el menú contextual.

Nota

El orden de los campos de salida para variables en el texto de aviso depende del idioma. Si se modifica la variable de un campo de salida en un idioma, en todos los demás idiomas el campo de salida modificado figura al final del texto de aviso.

Formatear el texto de aviso

Requisitos

- El editor "Avisos HMI" está abierto.
- Se ha creado un aviso.

Procedimiento

Para formatear un texto de aviso, proceda del siguiente modo:

1. Seleccione el aviso que desea editar.
2. Seleccione los caracteres que quiera formatear en "Propiedades > Propiedades > General > Texto de aviso" de la ventana de inspección.
3. Elija en el menú contextual el formato deseado, p. ej. "Subrayado" o "Mayúsculas".

Nota

No es posible visualizar variables y entradas de listas de textos subrayadas.

Resultado

Los caracteres seleccionados se representan en runtime en el formato elegido.

Eliminar formatos

Para eliminar todos los formatos de un texto, proceda del siguiente modo:

1. Seleccione los caracteres cuyo formato desee eliminar en el texto de aviso de la ventana de inspección.
2. En el menú contextual, elija "Borrar marcas de formato".

Resultado

Los caracteres seleccionados se representan sin formato en runtime.

Configurar una Loop-in-Alarm

Introducción

Para poder cambiar a la imagen que contiene información sobre el error ocurrido, tras aparecer el aviso correspondiente se configura una Loop-In-Alarm.

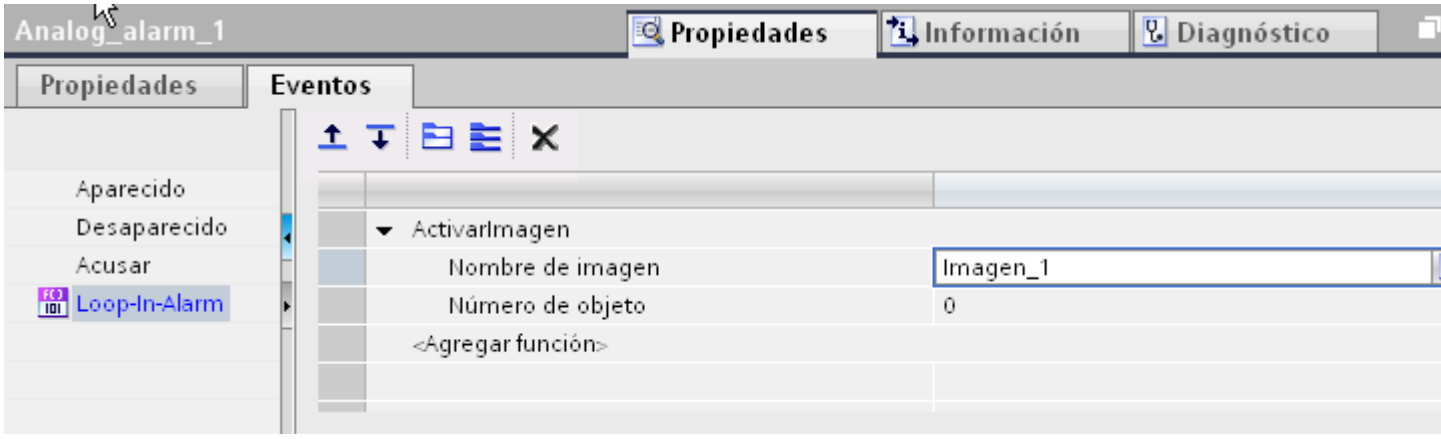
Requisitos

- La imagen que llama el Loop-In-Alarm está creada.
- El editor "Avisos HMI" está abierto.

Procedimiento

Para configurar un Loop-In-Alarm para un aviso, proceda del siguiente modo:

1. Haga clic en la ficha que contiene el aviso para el que desea configurar el Loop-In-Alarm.
2. Seleccione el aviso.
3. Elija en la ventana de inspección "Propiedades > Eventos > Loop-In-Alarm".
4. Seleccione la función de sistema "ActivarImagen".
5. Seleccione como parámetro la imagen que debe llamarse con el Loop-In-Alarm .



> **Nota**
>
> Para configurar el Loop-In-Alarm para un visor de avisos con la característica "Línea de avisos", utilice un botón con las siguientes funciones de sistema:
>
> - "EditarAviso" para paneles de operador con teclas
> - "VisorDeAvisosEditarAviso" para paneles de operador sin teclas
>
> Estas funciones de sistema lanzan el evento "Loop-In-Alarm". La línea de avisos no tiene botones.

Resultado

Si se hace clic en runtime en el botón "Loop-In-Alarm" del visor de avisos, se abre una imagen con información sobre el aviso seleccionado.

Consulte también

Configurar avisos analógicos (Página 8516)

Configurar avisos de bit (Página 8514)

Avisos en el editor "Variables HMI"

Configurar avisos de bit en el editor "Variables HMI"

Introducción

En WinCC también pueden crearse y editarse avisos de bit y analógicos junto con las variables de disparo en el editor "Variables HMI".

Nota

Si se borra, mueve o copia en el editor "Variables HMI", los cambios afectan también al editor "Avisos HMI".

Requisitos

El editor "Variables HMI" está abierto.

Procedimiento

Para configurar un aviso de bit, proceda del siguiente modo:

1. Para crear una variable, haga clic en "<Agregar>" en la tabla de la parte superior del área de trabajo.
   Se crea una nueva variable.
2. Según precise, configure una variable interna o externa.
   - Si elige una variable HMI, utilice los tipos de datos "Int" o "UInt".
   - Si elige una variable PLC, utilice los tipos de datos "Int" o "Word". Las áreas de entrada y de salida de una variable PLC no son adecuadas como disparador.
3. Seleccione la variable en la parte superior del área de trabajo.
4. En la parte inferior del área de trabajo de la ficha "Avisos de bit", haga clic en "<Agregar>" en la tabla.
   Se crea un aviso de bit para la variable. Si se ha seleccionado el tipo de datos erróneo, la variable se marca en el aviso de bit.

5. Configure el aviso de bit en la ventana de inspección:
   - Introduzca el texto de aviso deseado en "Propiedades > Propiedades > General > Texto de aviso".
     En caso necesario, inserte campos de salida en el texto de aviso.
   - Seleccione una categoría.
   - En "Propiedades > Propiedades > Disparador", seleccione el bit de disparo de la variable que dispara el aviso de bit.
6. En caso necesario, cree otros avisos de bit para vigilar las variables.

Nota

Una variable sólo se vigila con un procedimiento de aviso. Así pues, cree para una variable bien avisos analógicos **bien** avisos de bit.

Resultado

Los avisos de bit configurados están creados en el editor "Variables HMI" y se muestran en los editores "Avisos HMI" y "Variables HMI".

Configurar avisos analógicos en el editor "Variables HMI"

Introducción

En WinCC se crean avisos de bit y analógicos junto con las variables de disparo en el editor "Variables HMI". Además, los avisos se procesan como en el editor "Avisos HMI". Para una variable se crean dos valores de rango que se vigilan con avisos analógicos.

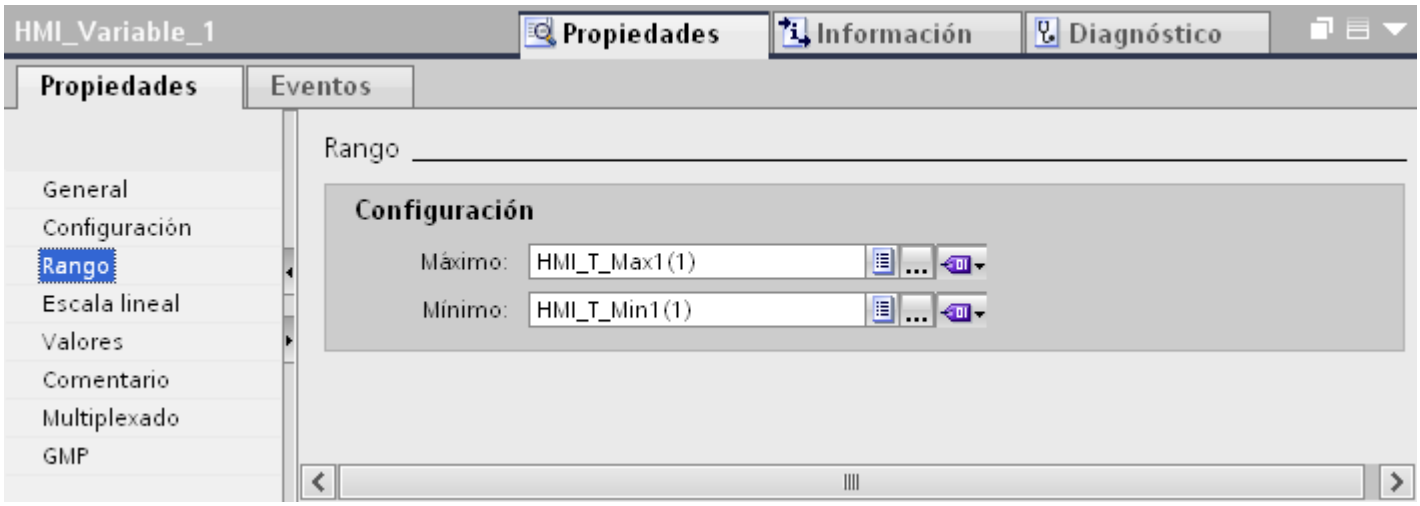
Requisitos

El editor "Variables HMI" está abierto.

Procedimiento

Para configurar un aviso analógico en el editor "Variables HMI", proceda del siguiente modo:

1. Para crear una variable, haga clic en "<Agregar>" en la tabla de la parte superior del área de trabajo.
   Se crea una nueva variable.
2. Según precise, configure una variable interna o externa.
3. Configure los valores de rango de la variable en "Propiedades > Propiedades > Rango" de la ventana de inspección:
   - Para los valores de rango deseados, elija si se va a utilizar una "Constante" o una "Variable_HMI" como valor límite. Si se ha seleccionado "Variable_HMI" se abre la lista de objetos. Seleccione la variable deseada.



1. En la parte inferior del área de trabajo haga clic en la ficha "Avisos analógicos".
   Cree un aviso analógico para ambos valores de rango.
2. Seleccione un aviso analógico y configúrelo en la ventana de inspección:
   – Introduzca el texto de aviso deseado en "Propiedades > Propiedades > General > Texto de aviso".
   – En caso necesario, inserte campos de salida en el texto de aviso.
   – Modifique la categoría predeterminada en caso necesario.
3. Continúe la configuración de los avisos analógicos como en el editor "Avisos HMI".
4. Prosiga hasta haber configurado todos los avisos analógicos.

Nota

Una variable sólo se vigila con un procedimiento de aviso. Así pues, para una variable deben crearse avisos analógicos **o** avisos de bit.

Resultado

Los avisos analógicos configurados están creados en el editor "Variables HMI" y se muestran en los editores "Avisos HMI" y "Variables HMI".

10.3.2.3 Configurar la salida de avisos

Sinopsis de cómo configurar la salida de avisos

Pasos necesarios para configurar la salida de avisos

La salida de avisos se configura en WinCC siguiendo los pasos siguientes:

1. Crear un visor de avisos
   Para visualizar avisos en runtime se utilizan los objetos de manejo y visualización del editor "Imágenes".
2. Configurar el acuse
   En el editor "Imágenes" se configura la acción que disparará el acuse.

Pasos de configuración avanzados

En función de las exigencias del proyecto se requerirán más pasos para configurar el visor de avisos:

1. Configurar autorizaciones
   Para garantizar que sólo los usuarios autorizados editan los avisos, asigne autorizaciones al visor de avisos y las teclas de función del panel de operador.
2. Configurar filtros en el visor de avisos
   En el editor "Imágenes" se configura el filtro de los avisos en runtime. En caso necesario también se configuran visores de avisos que muestran sólo los avisos seleccionados.
3. Configurar avisos de bit
   En el editor "Imágenes" configure avisos de bit en los elementos de control del equipo de manejo y visualización. Con una acción del operador se visualizará un aviso de bit preconfigurado. Una acción es p. ej. el acuse de un aviso.

Visualización de los avisos

Posibilidades para visualizar avisos en el panel de operador

WinCC ofrece las siguientes posibilidades para visualizar avisos en el panel de operador:

- Visor de avisos
  El visor de avisos se configura en una imagen. En función del tamaño configurado, muestra varios avisos a la vez. Es posible configurar varios visores de avisos de distinto contenido.
- Ventana de avisos
  La ventana de avisos se configura en el editor "Imagen general". En función del tamaño configurado, la ventana de avisos muestra varios avisos a la vez. La ventana de avisos no está asignada a ninguna imagen. Dependiendo de la configuración, la ventana de avisos se abre cuando aparece un aviso que pertenece a una categoría determinada. Según la configuración, no volverá a cerrarse hasta que se haya acusado el aviso.

Señales adicionales

- Indicador de avisos
  El indicador de avisos es un símbolo gráfico configurable. Cuando llega un aviso, se muestra el indicador de avisos en el panel de operador. El indicador de avisos se configura en el editor "Imagen general".
  El indicador de avisos tiene dos estados:
  - Intermitente: Hay como mínimo un aviso de acuse obligado.
  - Estático: Los avisos se han acusado, pero al menos uno de ellos no ha desaparecido todavía.
    En función del panel de operador, el indicador de avisos muestra también el número de avisos pendientes.
- Funciones de sistema
  En un evento de un aviso se configura una lista de funciones que se ejecutará cuando se produzca el evento en runtime.
  Para no manejar el visor de avisos o la ventana de avisos con la barra de herramientas, utilice funciones de sistema para avisos en WinCC.

Visualización de las categorías predefinidas en runtime

La tabla siguiente muestra los símbolos con los que se representan las categorías predefinidas en el visor de avisos:

Categoría	Símbolo visualizado
"Errors"	!
"Sistema"	$
"Warnings"	<ningún símbolo>

Configurar un visor de avisos

Introducción

Los avisos actuales se visualizan en runtime en un visor o en una ventana de aviso.

Requisitos

- Hay una imagen abierta en el editor "Imágenes".
- La Task Card "Herramientas" está abierta.

Configurar avisos para el visor de avisos

Para determinar qué avisos aparecerán en el visor, proceda del siguiente modo:

1. Inserte el objeto "Visor de avisos" en la imagen desde la Task Card "Herramientas".
2. Seleccione el visor de avisos.
   - Seleccione en la ventana de inspección "Propiedades > Propiedades > General > Visualización > Estados de aviso actuales".
     Dado el caso, determine si se visualizarán avisos de acuse obligado o avisos sin obligación de acuse.
   - Para visualizar todos los avisos que se encuentran en el búfer de avisos, active "Búfer de avisos".
3. Active en la tabla las categorías que se van a visualizar en el visor de avisos.

Configurar la representación del visor de avisos

Para determinar cómo se representarán los avisos en el visor, proceda del siguiente modo:

1. En "Propiedades > Propiedades > Representación > Configuración > Filas por aviso" de la ventana de inspección, determine el número de filas que se deben visualizar en cada aviso.
2. En "Propiedades > Propiedades > Visualización", seleccione los elementos de control disponibles en el panel de operador.
3. Configure las columnas en "Propiedades > Propiedades > Columnas":
   - En "Columnas visibles" seleccione las columnas que se verán en el visor de avisos.
   - En "Propiedades de columnas", determine las propiedades de las columnas.
   - En "Ordenar", elija el orden de clasificación de los avisos.

Resultado

Los avisos de distintas categorías se visualizarán en runtime en el visor de avisos.

Consulte también

Visor de avisos (Página 8438)

Configurar la ventana de avisos

Introducción

La ventana de avisos muestra los avisos actuales. La ventana de avisos se configura en el editor "Imagen general". La ventana de avisos no está asignada a ninguna imagen. Dependiendo de la configuración, la ventana de avisos se abre cuando hay un aviso pendiente que pertenece a una determinada categoría. Según la configuración, no volverá a cerrarse hasta que se haya acusado el aviso. El panel de operador se puede seguir utilizando aunque

haya avisos pendientes y visualizados. Una ventana de avisos se representa y configura como un visor de avisos.

Para ocultarla selectivamente en la configuración, créela en un nivel propio.

Requisitos

- El editor "Imagen general" está abierto.
- La Task Card "Herramientas" se visualiza.
- La ventana de inspección está abierta.

Procedimiento

Para configurar una ventana de avisos, proceda del siguiente modo:

1. Inserte el objeto "Ventana de avisos" en la imagen global desde la Task Card "Herramientas".
2. La ventana de avisos se configura igual que un visor de avisos.
3. En "Propiedades > Propiedades > Modo > Ventana" de la ventana de inspección, seleccione el comportamiento y el manejo de la ventana de avisos en runtime.
   - Si la ventana de avisos debe permanecer enfocada en runtime al cambiar de imagen, active "Modal".
     Esta opción es importante puesto que no se asiste la conmutación entre la imagen y las diversas ventanas mediante <Ctrl+Tab>.

Resultado

Los avisos de la categoría seleccionada se visualizarán en runtime en la ventana de avisos.

Consulte también

Ventana de avisos (Página 8441)

Configurar el indicador de avisos

Introducción

El indicador de avisos indica mediante un triángulo de advertencia que hay avisos pendientes o que se deben acusar. El indicador de avisos se visualizará cuando se reciba un aviso de una categoría configurada con dicho indicador.

El indicador de avisos tiene dos estados:

- Intermitente: Hay como mínimo un aviso de acuse obligado.
- Estático: Por lo menos uno de los avisos acusados no se ha desactivado todavía.

En caso necesario, durante la configuración puede determinarse que la ventana de avisos se abra en runtime cuando el usuario accione el indicador de avisos.

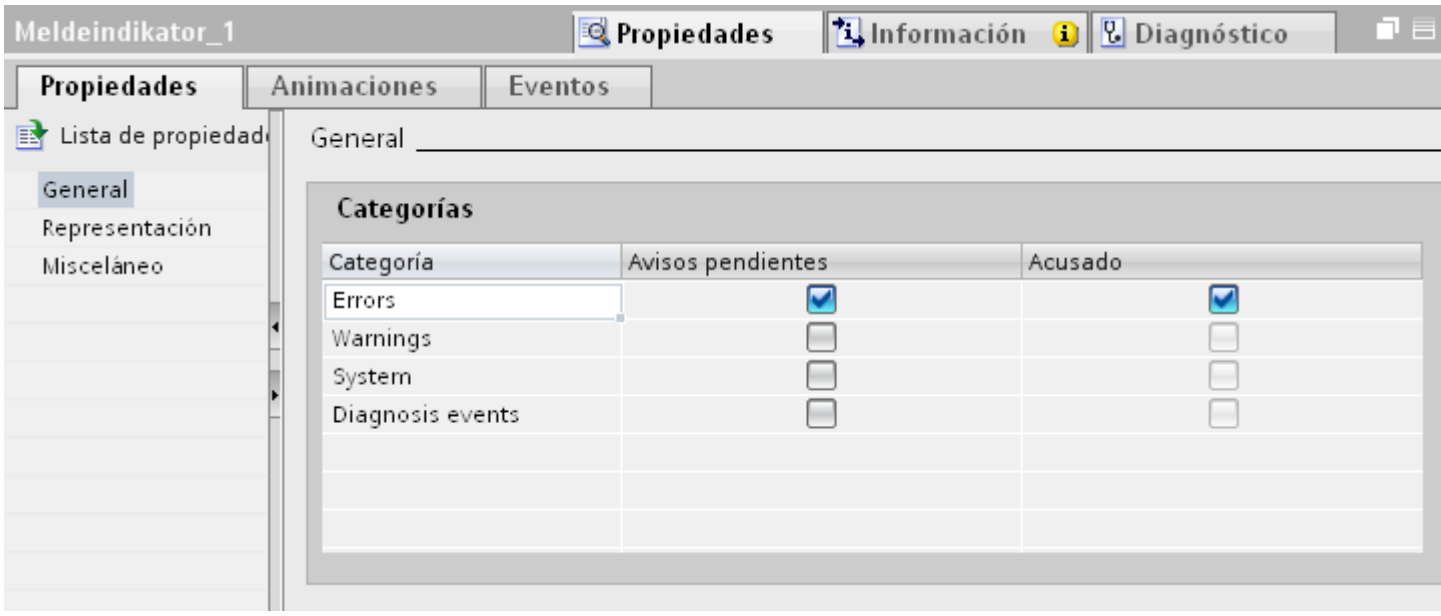
Requisitos

- El editor "Imagen general" está abierto.
- La Task Card "Herramientas" está abierta.
- La ventana de inspección está abierta.

Procedimiento

Para configurar el indicador de avisos, proceda del siguiente modo:

1. Inserte el objeto "Indicador de avisos" en el área de trabajo desde la Task Card "Herramientas".
2. Seleccione el indicador de avisos.
3. En "Propiedades > Propiedades > General" de la ventana de inspección, seleccione las categorías que deben visualizarse con el indicador de avisos.
   Indique si los avisos pendientes y/o acusados se deben visualizar junto con el indicador de avisos.



4. Configure en "Propiedades > Evento" la función de sistema "MostrarVentanaDeAvisos" asociada a un evento del indicador de avisos.

Nota

Si se ha configurado una ventana permanente en la imagen o en la plantilla, no coloque la ventana de avisos y el indicador de avisos en el área de la ventana permanente. En caso contrario, la ventana de avisos y el indicador de avisos no se verán en runtime. Sin embargo, la ventana permanente no se ve en el editor "Imagen general".

Resultado

Si en runtime hay avisos de la categoría seleccionada pendientes o que deban acusarse, se mostrará el indicador de avisos. Cuando el usuario accione el indicador de avisos se abrirá la ventana de avisos.

Consulte también

Indicador de avisos (Página 8443)

10.3.2.4 Acuse de avisos

Configurar la obligación de acuse de un aviso por categoría

Introducción

Para configurar un aviso con acuse obligado, asigne el aviso en cuestión a una categoría con el concepto de acuse "Aviso con acuse simple".

Requisitos

- El editor "Avisos HMI" está abierto.
- La categoría deseada se ha creado.
- El aviso deseado se ha creado.

Seleccionar el concepto de acuse para una categoría

El concepto de acuse de las categorías predefinidas está especificado. Sólo hay que elegir el concepto de acuse de las categorías personalizadas. Para ello proceda del siguiente modo:

1. En el editor "Avisos HMI" haga clic en la ficha "Categoría" y seleccione la categoría deseada.
2. En la ventana de inspección seleccione el concepto de acuse deseado en "Propiedades > Propiedades > Acuse".

Asignar avisos a una categoría de acuse obligatorio

Para asignar una categoría de acuse obligatorio a un aviso, proceda del siguiente modo.

1. En el editor "Avisos HMI" haga clic en la ficha del procedimiento de aviso deseado y seleccione el aviso deseado.
2. Seleccione la categoría del aviso en "Propiedades > Propiedades > General" de la ventana de inspección.

Resultado

El aviso sólo desaparece en runtime cuando el operador lo ha acusado.

Configurar un disparador para el acuse de avisos

Introducción

Básicamente, la obligación de acuse de un aviso se define por la categoría. Luego el operador acusa el aviso con la tecla de función "ACK" del panel de operador o bien con el botón "Acusar" del visor de avisos.

Además, existen las posibilidades siguientes de disparar un acuse:

- Configura un botón para acusar un aviso
- Acusar un aviso de bit con el controlador

Requisitos

- El editor "Avisos HMI" está abierto.
- La categoría deseada se ha creado.
- El aviso deseado se ha creado.
- En el editor "Imágenes" se ha creado un visor de avisos y un botón.

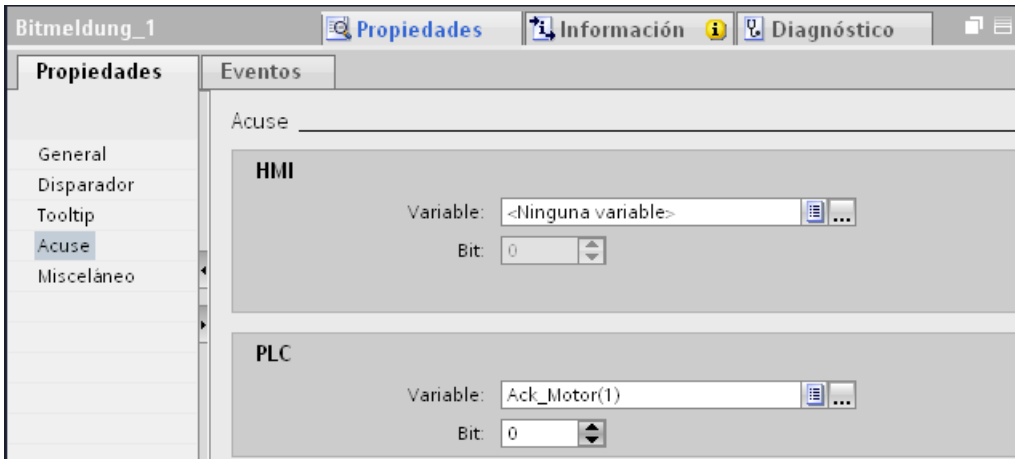
Configurar un botón para acusar un aviso

Para configurar un botón que acuse un aviso proceda del siguiente modo:

1. Seleccione el botón en el editor "Imágenes".
2. En "Propiedades > Eventos" de la ventana de inspección configure la función de sistema "VisorDeAvisosAcusarAviso" asociada al evento "Hacer clic".
3. Seleccione como parámetro el visor de avisos.

Acusar un aviso de bit con el controlador

1. En el editor "Avisos HMI" haga clic en la ficha "Aviso de bit" y seleccione el aviso de bit deseado.
2. En la ventana de inspección, seleccione la variable y el bit que debe acusar el aviso desde el controlador en "Propiedades > Propiedades > Acuse > PLC".



Acuse de avisos de un grupo

Las distintas posibilidades de acuse tienen los siguientes efectos en los grupos de avisos:

- Acuse del aviso por parte del operador
  Se acusa el aviso de un grupo de avisos, p. ej., con la tecla de acuse <ACK> o con una tecla de función. Se acusan todos los avisos de un grupo de avisos.
- Acuse del aviso por parte del controlador
  El aviso de un grupo se acusa mediante la activación de un bit de la variable.
  Solamente se acusa el aviso correspondiente.

Enviar acuse de aviso al controlador

Requisitos

- El editor "Avisos HMI" está abierto.
- El aviso deseado está creado y asignado a una categoría con acuse obligado.

Nota

El acuse de avisos analógicos no puede enviarse al controlador.

Enviar acuse de aviso al controlador

Para configurar que el acuse de un aviso sea enviado al controlador, proceda del siguiente modo:

1. En el editor "Avisos HMI" haga clic en la ficha "Aviso de bit" y seleccione el aviso de bit deseado.
2. Seleccione en la ventana de inspección "Propiedades > Propiedades > Acuse".
3. En "HMI", elija la variable y el bit que debe activarse con el acuse del aviso.

Nota

Tanto el panel de operador como el controlador tienen únicamente acceso de lectura al área de memoria de la variable de acuse.

Resultado

Cuando el operador acusa el aviso en runtime, la acción se transmite al controlador.

10.3.3 Manejar avisos en runtime

10.3.3.1 Avisos en runtime

Avisos

Los avisos indican en el panel de operador eventos y estados que se presentan en la instalación, en el proceso, o bien en el panel de operador. Al ocurrir un estado, éste se indica.

En el caso de los avisos pueden ocurrir los siguientes eventos:

- Aparecido
- Desaparecido
- Acusar
- Loop-In-Alarm

El ingeniero determina qué avisos deben ser acusados por el usuario.

Un aviso puede contener las informaciones siguientes:

- Fecha
- Hora
- Texto de aviso
- Ubicación del fallo
- Estado
- Categoría

- Número de aviso
- Grupo de avisos

Categorías

Los avisos están asignados a distintas categorías.

- "Warnings"
  Por lo general, los avisos de esta categoría indican estados en la instalación, p. ej. "Motor encendido". Los avisos de esta categoría no deben acusarse.
- "Errors"
  Los avisos de esta categoría siempre deben ser acusados. Por lo general, las alarmas indican estados críticos en la instalación, p. ej. " Temperatura del motor demasiado elevada".
- "Sistema"
  Los avisos de sistema indican estados o eventos del panel de operador.
  Los avisos del sistema informan p. ej. acerca de operaciones erróneas o de averías en la comunicación.
- Categorías personalizadas
  Las propiedades de esta categoría se definen durante la configuración.

Búfer de avisos

Los eventos de avisos se guardan en un búfer interno. El tamaño de este búfer de avisos depende del tipo de panel de operador.

Visor de avisos

En el visor de avisos se visualizan los avisos o eventos de avisos seleccionados del búfer de avisos. En la configuración se define si es necesario acusar los avisos que van apareciendo.

Ventana de avisos

La ventana de avisos no está asignada a ninguna imagen. Dependiendo de la configuración, la ventana de avisos se abre cuando hay un aviso pendiente que pertenece a una determinada categoría.

El orden de los avisos visualizados puede configurarse. Se puede elegir entre visualizar primero el aviso actual o el más antiguo. Asimismo, en la ventana de avisos puede visualizarse el lugar exacto del fallo con fecha y hora del evento de aviso. La visualización se puede configurar de manera que aparezcan sólo los avisos que contengan en el texto una cadena de caracteres determinada.

Indicador de avisos

El indicador de avisos es un símbolo gráfico que se visualiza en la pantalla cuando aparece un aviso de la categoría determinada.

El indicador de avisos puede tener dos estados:

- Intermitente: Hay como mínimo un aviso pendiente de acuse.
- Estático: Los avisos se han acusado, pero al menos uno de ellos no ha desaparecido todavía. El número que aparece indica la cantidad de avisos que todavía están pendientes.

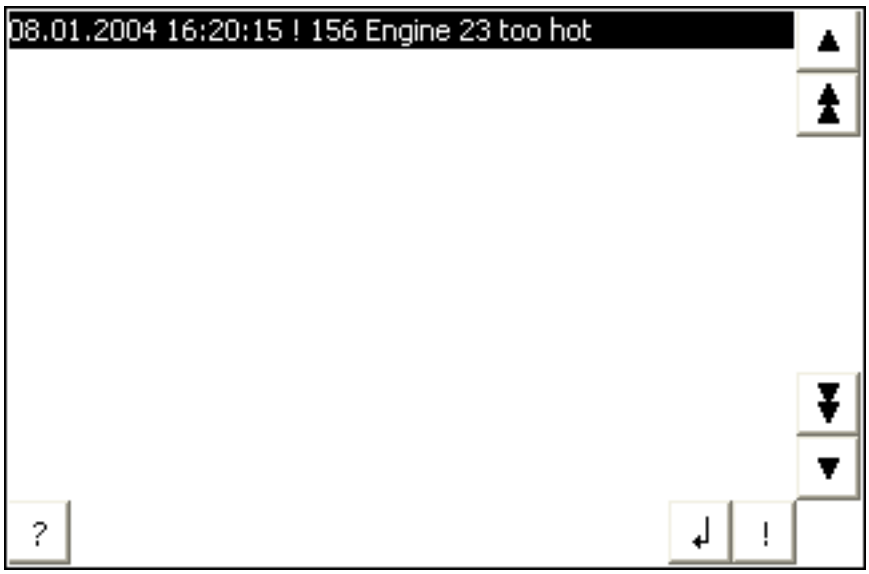
10.3.3.2 Visor de avisos simple, ventana de avisos simple en runtime

Utilización

En el visor de avisos simple se visualizan los avisos o eventos de avisos seleccionados del búfer de avisos. La apariencia y el manejo de la ventana de avisos simple son similares a los del visor de avisos simple.

Nota

En la ficha "Animaciones" de la ventana de inspección del sistema de ingeniería se dinamiza la visibilidad de un objeto, p. ej. El objeto "Visor de avisos simple" no soporta ninguna animación en runtime. Si se ha configurado una animación y se comprueba la coherencia del proyecto, p. ej., en la ventana de resultados aparece un mensaje de error.



Representación

Dependiendo de la configuración, en el visor de avisos se visualizan diversas columnas con información sobre un aviso o un evento de aviso.

Con objeto de distinguir las diferentes categorías, la primera columna del visor de avisos contiene un símbolo:

Símbolo	Categoría
!	"Errors"
vacío	"Warnings"
en función de la configuración	Categorías personalizadas
$	"Sistema"

Manejo

Según sea la configuración, el visor de avisos se maneja del modo siguiente:

- Acusar avisos
- Editar avisos

Elementos de control

Los botones tienen las funciones siguientes:

Botón	Función
!	Acusar un aviso
↲	Loop-In-Alarm Cambia a la imagen que contiene información sobre el error ocurrido.
?	Mostrar el tooltip de un aviso
▶	Muestra el texto completo del aviso seleccionado en una ventana independiente (es decir, en la ventana de textos de aviso) En la ventana de textos de aviso se visualizan los textos de aviso que exceden el espacio disponible en el visor de avisos. Cierre la ventana del texto de aviso con el botón ✕.
▲	Retrocede un aviso hacia arriba.
⏶	Retrocede una página hacia arriba en el visor de avisos.
⏷	Avanza una página hacia abajo en el visor de avisos.
▼	Avanza un aviso hacia abajo.

Representación de los elementos de control

La visualización de los botones de control en el visor de avisos simple depende del tamaño configurado. Por tanto, compruebe en el panel de operador si están disponibles todos los botones necesarios.

10.3.3.3 Indicador de avisos en runtime

Utilización

Si hay avisos de la categoría definida que estén pendientes o deban acusarse, se mostrará el indicador de avisos.

Representación

El indicador de avisos puede tener dos estados:

- Intermitente: Hay como mínimo un aviso pendiente de acuse.
- Estático: Los avisos se han acusado, pero al menos uno de ellos no ha desaparecido todavía. El número que aparece indica la cantidad de avisos que todavía están pendientes.

Manejo

Dependiendo de la configuración, al accionar el indicador de avisos se abre una ventana de avisos. El indicador de avisos sólo se puede manejar con la pantalla táctil.

10.3.3.4 Acusar avisos

Introducción

Los avisos se acusan en runtime según la configuración del proyecto. Acuse los avisos del modo siguiente:

- Con los botones de los objetos de manejo y visualización
- Con la tecla "ACK" del panel de operador
- Con teclas de función o botones configurados individualmente

Si se ha configurado una autorización de uso para los elementos de control, sólo los usuarios autorizados podrán acusar los avisos.

Para acusar los avisos en runtime automáticamente, utilice funciones de sistema y la opción "Acuse por parte del controlador".

Variantes del acuse

En runtime los avisos se acusan uno por uno o varios a la vez. Se distinguen las opciones siguientes:

- Acuse individual
  Acuse de un aviso con un botón o una tecla de función.
- Acusar grupos de avisos
  Acuse de todos los aviso de un grupo con un botón o una tecla de función.

Requisitos

- Se visualiza un aviso en el panel de operador.

Procedimiento

Para acusar un aviso, proceda del siguiente modo:

1. Seleccione el aviso.
2. Haga clic en el botón !.

Resultado

El aviso recibe el estado "acusado". Cuando la condición para que se lance un aviso ya no se cumple, el aviso recibe además el estado "desaparecido" y ya no se muestra en el panel de operador.

10.3.4 Referencia

10.3.4.1 Funciones de sistema para avisos

Funciones de sistema

Las funciones de sistema son funciones predefinidas que permiten realizar numerosas tareas en runtime incluso sin tener conocimientos de programación. Las funciones de sistema se utilizan en una lista de funciones.

La tabla muestra todas las funciones de sistema para representar y editar avisos.

Función de sistema	Efecto
EditarAviso	Lanza el evento "Loop-In-Alarm" para todos los avisos seleccionados.
BorrarBuferDeAvisos	Borra avisos del búfer de avisos del panel de operador.

Función de sistema	Efecto
BorrarAutorizacionBuferDeAvisosProtool	La misma función que "BorrarBúferDeAvisos". Esta función de sistema está incluida por motivos de compatibilidad y utiliza la numeración de ProTool.
VistaDeAvisosEditarAviso	Lanza el evento "Loop-In-Alarm" para todos los avisos seleccionados en el visor de avisos indicado.
VistaDeAvisosAcusarAviso	Acusa los avisos seleccionados en el visor de avisos indicado.
VistaDeAvisosMostrarTextoAyuda	Muestra el tooltip configurado para el aviso seleccionado en el visor de avisos indicado.
AcusarAviso	Acusa todos los avisos seleccionados.
MostrarVentanaDeAvisos	Muestra u oculta la ventana de avisos en el panel de operador.

10.3.4.2 Avisos de sistema

Principios básicos de los avisos de sistema

Avisos de sistema

En el panel de operador los avisos de sistema ofrecen información sobre los estados internos del panel de operador y del controlador.

A continuación encontrará un sinóptico en el que se indica cuándo se presenta un aviso de sistema y cómo se corrige la causa del error.

Nota

Dependencia del panel de operador

Dependiendo de la funcionalidad del panel de operador, serán aplicables sólo algunos de los avisos de sistema descritos.

Nota

Los avisos de sistema se visualizan en un visor de avisos. Los avisos de sistema se visualizan en el idioma configurado en ese momento en el panel de operador.

Parámetros de los avisos de sistema

Los avisos de sistema pueden contener parámetros encriptados. Los parámetros son relevantes para el seguimiento de un error, ya que ofrecen indicios sobre el código fuente del software runtime. Los parámetros aparecen después del texto "Código de error:"

30000 - Avisos de error al utilizar funciones de sistema

Significado de los avisos de sistema

A continuación se representan todos los avisos de sistema que se pueden visualizar. Los avisos de sistema están divididos en diversas áreas.

Tabla 10-2 30000 - Avisos de error al utilizar funciones de sistema

Número	Efecto/causas	Solución
30010	La variable no ha podido acoger el resultado de la función de sistema, p. ej. al excederse el rango de valores.	Compruebe los tipos de variable de los parámetros de la función de sistema.
30011	No se ha podido ejecutar una función de sistema, puesto que en el parámetro de la función de sistema se ha transferido un valor o tipo inadmisible.	Compruebe el valor del parámetro y el tipo de variable del parámetro inadmisible. Si se utiliza una variable como parámetro, compruebe su valor.
30012	No se ha podido ejecutar una función de sistema, puesto que en el parámetro de la función de sistema se ha transferido un valor o tipo inadmisible.	Compruebe el valor del parámetro y el tipo de variable del parámetro inadmisible. Si se utiliza una variable como parámetro, compruebe su valor.

40000 - Avisos de escala lineal

Significado de los eventos de sistema

A continuación se representan todos los eventos de sistema que se pueden visualizar. Los eventos de sistema están divididos en distintas áreas.

Tabla 10-3 40000 - Avisos de escala lineal

Número	Efecto/causas	Solución
40010	No se ha podido ejecutar la función de sistema, debido a que no es posible convertir los parámetros a un tipo de variable común.	Compruebe los tipos de parámetros en la configuración.
40011	No se ha podido ejecutar la función de sistema, debido a que no es posible convertir los parámetros a un tipo de variable común.	Compruebe los tipos de parámetros en la configuración.

50000 – Avisos del servidor de datos

Significado de los eventos de sistema

A continuación se representan todos los eventos de sistema que se pueden visualizar. Los eventos de sistema están divididos en distintas áreas.

Tabla 10-4 50000 – Avisos del servidor de datos

Número	Efecto/causas	Solución
50000	El panel de operador recibe datos más rápido de lo que los puede procesar. Por tanto, no se aceptarán nuevos datos hasta que no se hayan procesado los datos existentes. Luego se reanudará el intercambio de datos.	--
50001	El intercambio de datos ha sido reanudado.	--

70000 - Avisos de funciones de Win32

Significado de los avisos de sistema

A continuación se representan todos los avisos de sistema que se pueden visualizar.

Tabla 10-5 70000 - Avisos de funciones de Win32

Número	Efecto/causas	Solución
70010	No se ha podido iniciar el programa debido a que no se encuentra en la ruta indicada, o bien, porque no hay suficiente espacio de memoria libre.	Compruebe si el programa existe en la ruta o en la ruta de búsqueda indicada o cierre otros programas.
70011	No se ha podido cambiar la fecha y hora del sistema. El mensaje de error aparece únicamente en combinación con el puntero de área "Fecha/hora del autómata". Causas posibles: • En la tarea de control se ha transferido una fecha / hora inadmisible. • El usuario de Windows no está autorizado para cambiar la fecha y hora del sistema. Si en el aviso de sistema se indica como primer parámetro el valor 13, entonces el segundo parámetro indica el byte que tiene el valor incorrecto.	Compruebe la hora que se debe ajustar. En Windows NT/XP: El usuario que ejecuta WinCC Runtime debe obtener la autorización para cambiar la fecha y hora del sistema operativo.
70012	Ha ocurrido un error al ejecutar la función de sistema "PararRuntime" con la opción "Runtime y sistema operativo". No se cierran Windows ni WinCC Runtime. Una causa posible es que otros programas no se puedan cerrar.	Cierre todos los programas que se están ejecutando. Luego cierre Windows.
70013	No se ha podido cambiar la fecha y hora del sistema debido a que el valor introducido no es admisible. Es probable que se hayan usado separadores incorrectos.	Compruebe la fecha y hora que se deben ajustar.
70014	No se ha podido cambiar la fecha y hora del sistema. Causas posibles: • Se ha transferido una fecha u hora inadmisible. • El usuario de Windows no está autorizado para cambiar la fecha y hora del sistema. Windows rechaza el ajuste.	Compruebe la hora que se debe ajustar. En Windows NT/XP: El usuario que ejecuta WinCC Runtime debe obtener la autorización para cambiar la fecha y hora del sistema operativo.

Número	Efecto/causas	Solución
70015	No se ha podido leer la fecha y hora del sistema debido a que Windows rechaza la lectura.	--
70016	Se ha intentado seleccionar una imagen a través de una función de sistema o de una tarea. Ello no es posible, puesto que no existe el número de imagen configurado. O: Debido a la falta de memoria del sistema, no se ha podido crear una imagen. O: La imagen está bloqueada. O: La imagen no se puede abrir correctamente.	En la función de sistema o en la tarea, compare los números de imágenes con los números de imágenes configurados. En caso necesario, asigne el número a una imagen. Compruebe los datos de llamada de la imagen y si ésta ha sido bloqueada para determinados usuarios.
70017	La fecha/hora no se puede leer desde el puntero de área porque la dirección configurada en el controlador no existe o no ha sido configurada.	Modifique la dirección o configure la dirección en el controlador.
70018	Confirmación de importación correcta de la lista de contraseñas.	--
70019	Confirmación de exportación exitosa de la lista de contraseñas.	--
70020	Confirmación de activación del informe de avisos.	--
70021	Confirmación de desactivación del informe de avisos.	--
70022	Confirmación de inicio de la acción "Importar lista de contraseñas".	--
70023	Confirmación de inicio de la acción "Exportar lista de contraseñas".	--
70024	Se ha excedido el rango de valores de la variable al ejecutar la función de sistema. No se realizará el cálculo de la función de sistema.	Compruebe el cálculo deseado y corríjalo en caso necesario.
70025	Se ha excedido el rango de valores de la variable al ejecutar la función de sistema. No se realizará el cálculo de la función de sistema.	Compruebe el cálculo deseado y corríjalo en caso necesario.
70026	En la memoria de imágenes interna ya no hay más imágenes guardadas. No se puede seleccionar ninguna imagen más.	--
70027	Se ha iniciado la copia de seguridad del sistema de archivos RAM.	--
70028	Se ha creado la copia de seguridad del sistema de archivos RAM. Los archivos de la RAM se copian, a prueba de fallos, en la memoria flash. En un nuevo arranque, estos archivos con copia de seguridad se recopian otra vez en el sistema de archivos RAM.	--
70029	La copia de seguridad del sistema de archivos RAM ha fallado. No se ha hecho la copia de seguridad del sistema de archivos RAM.	Compruebe la configuración en el diálogo "Control Panel > OP" y realice la copia de seguridad del sistema de archivos RAM mediante del botón "Save Files" en la ficha "Persistent Storage".
70030	Los parámetros configurados de la función de sistema son incorrectos. No se ha establecido la conexión con el nuevo controlador.	Compare los parámetros configurados de la función de sistema con los parámetros configurados de los controladores y corríjalos en caso necesario.

Número	Efecto/causas	Solución
70031	El controlador configurado en la función de sistema no es un controlador S7. No se ha establecido la conexión con el nuevo controlador.	Compare el parámetro configurado "Nombre de autómata S7" de la función de sistema con los parámetros configurados de los controladores y corríjalo en caso necesario.
70032	En la imagen seleccionada no existe el objeto configurado con este número en el orden de tabulación. Se ejecuta el cambio de imagen. No obstante, se resalta el primer objeto.	Compruebe el número en el orden de tabulación y corríjalo en caso necesario.
70033	No se ha podido enviar un e-mail porque ya no hay conexión TCP/IP con el servidor SMTP. Este aviso de sistema se genera sólo en el primer intento fallido. Los demás intentos fallidos para enviar un e-mail ya no generan ningún otro aviso de sistema. El aviso tan sólo se genera nuevamente cuando se consigue enviar un correo electrónico. El componente central del correo electrónico en WinCC Runtime intenta establecer, en intervalos regulares (1 min.), la conexión con el servidor SMTP y enviar los correos eléctricos pendientes.	Compruebe la conexión de red con el servidor SMTP y, de ser necesario, restablezca la conexión.
70034	Después de interrumpirse la conexión se ha podido restablecer la conexión TCP/IP con el servidor SMTP. Los e-mails pendientes en la cola de espera se envían.	--
70036	No se ha configurado ningún servidor SMTP para el envío de e-mails. Por ello no se puede establecer una conexión con un servidor SMTP y no se pueden enviar correos electrónicos. El aviso de sistema es generado por WinCC Runtime cuando se intenta enviar un correo electrónico por primera vez.	Configure un servidor SMTP: En WinCC Engineering System mediante "Configuración del panel > Configuración del panel" En el sistema operativo Windows CE a través de "Control Panel > Internet Settings > Email > SMTP Server"
70037	Por razones desconocidas no se ha podido enviar un e-mail. El contenido del e-mail se rechaza.	Compruebe los parámetros del e-mail (p. ej. destinatario, etc.).
70038	El servidor SMTP ha rechazado transmitir o enviar el e-mail porque el dominio del destinatario no se conoce en el servidor, o bien porque el servidor SMTP necesita una autenticación. El contenido del e-mail se rechaza.	Compruebe el dominio de la dirección del destinatario o, de ser posible, desactive la autenticación en el servidor SMTP. Una autenticación SMTP no está siendo procesada actualmente por WinCC Runtime.
70039	La sintaxis de la dirección de e-mail es incorrecta o contiene caracteres no válidos. El contenido del e-mail se rechaza.	Comprobar la dirección de e-mail del destinatario.
70040	La sintaxis de la dirección de e-mail es incorrecta o contiene caracteres no válidos.	--
70041	La importación de la administración de usuarios se ha cancelado debido a un error. La importación no se ha realizado.	Compruebe la administración de usuarios o cárguela de nuevo en el panel de operador.
70042	El rango de valores de la variable se ha excedido al ejecutar la función de sistema. No se calcula la función de sistema.	Compruebe el cálculo deseado y corríjalo en caso necesario.

Número	Efecto/causas	Solución
70043	El rango de valores de la variable se ha excedido al ejecutar la función de sistema. No se calcula la función de sistema.	Compruebe el cálculo deseado y corríjalo en caso necesario.
70044	Ha ocurrido un error al enviar los correos electrónicos. No se han enviado los correos electrónicos.	Compruebe la configuración SMTP y tenga en cuenta el mensaje de error del aviso de sistema.
70045	No se ha podido cargar un archivo necesario para la encriptación del e-mail.	Actualice el sistema operativo y el software runtime.
70046	El servidor no soporta la encriptación.	Seleccione un servidor SMTP que soporte la encriptación.
70047	Es probable que no concuerden las versiones SSL del panel de operador y el servidor SMTP.	Contacte con el administrador de la red o el proveedor del servidor SMTP.

110000 - Avisos de funciones offline

Significado de los avisos de sistema

A continuación se representan todos los avisos de sistema que se pueden visualizar.

Tabla 10-6 110000 - Avisos de funciones offline

Número	Efecto/causas	Solución
110000	Se ha ejecutado un cambio del modo de operación. El modo de operación actual es "Offline".	--
110001	Se ha ejecutado un cambio del modo de operación. El modo de operación actual es "Online".	--
110002	No se ha cambiado el modo de operación.	Compruebe la conexión con los controladores. Compruebe si en el controlador existe el área de direcciones para el puntero de área 88 "Coordinación".
110003	El modo de operación del controlador indicado ha sido cambiado por la función de sistema "EstablecerModoDeConexión". El modo de operación actual es "Offline".	--
110004	El modo de operación del controlador indicado ha sido cambiado por la función de sistema "EstablecerModoDeConexión". El modo de operación actual es "Online".	--
110005	Se ha intentado conmutar el controlador indicado al modo de operación "Online" con la función de sistema "EstablecerModoDeConexión", a pesar de que todo el sistema está en el modo de operación "Offline". Esta conmutación es inadmisible. El modo de operación del controlador sigue siendo "Offline".	Conmute todo el sistema al modo de operación "Online" y ejecute otra vez la función de sistema.
110006	El contenido del puntero de área "Identificación del proyecto" no coincide con la identificación de proyecto configurada en WinCC. Por ello se cierra WinCC Runtime.	Compruebe: • la identificación de proyecto registrada en el controlador. • la identificación de proyecto registrada en WinCC.

120000 – Avisos de la representación de curvas

Significado de los eventos de sistema

A continuación se representan todos los eventos de sistema que se pueden visualizar.

Tabla 10-7 120000 – Avisos de la representación de curvas

Número	Efecto/causas	Solución
120000	La curva no se visualiza porque se ha configurado un eje incorrecto para la misma, o bien una curva incorrecta.	Modifique la configuración.
120001	La curva no se visualiza porque se ha configurado un eje incorrecto para la misma, o bien una curva incorrecta.	Modifique la configuración.
120002	La curva no se representa porque la variable asignada accede a una dirección no válida en el controlador.	Compruebe si en el controlador existe el área de datos para la variable, si la dirección configurada es correcta, o si concuerda el rango de valores de la variable.

140000 – Avisos de la conexión: conexión + panel de operador

Significado de los avisos de sistema

A continuación se representan todos los avisos de sistema que se pueden visualizar.

Tabla 10-8 140000 – Avisos de la conexión: conexión + panel de operador

Número	Efecto/causas	Solución
140000	Se ha establecido correctamente la conexión online con el controlador.	--
140001	Se ha deshecho la conexión online con el controlador.	--
140003	No se actualiza ni se escribe ninguna variable.	Compruebe la conexión y verifique si está conectado el controlador. En el Panel de control, compruebe con "Ajustar interface PG/PC" los parámetros configurados. Ejecute un rearranque.
140004	No se actualiza ni se escribe ninguna variable porque el punto de acceso o la parametrización del módulo es incorrecto.	Compruebe la conexión y verifique si el controlador está conectado. En el Panel de control, compruebe con "Ajustar interface PG/PC" el punto de acceso o la parametrización del módulo (MPI, PPI, PROFIBUS). Ejecute un rearranque.
140005	No se actualiza ni se escribe ninguna variable porque la dirección del panel de operador es incorrecta (probablemente es demasiado grande).	Utilice una dirección diferente para el panel de operador. Compruebe la conexión y verifique si está conectado el controlador. En el Panel de control, compruebe con "Ajustar interface PG/PC" los parámetros configurados. Ejecute un rearranque.

Número	Efecto/causas	Solución
140006	No se actualiza ni se escribe ninguna variable porque la velocidad de transferencia es incorrecta.	En WinCC, seleccione una velocidad de transferencia diferente (dependiente del módulo, perfil, interlocutor, etc.).
140007	No se actualiza ni se escribe ninguna variable porque el perfil del bus es incorrecto (v. %1). Los siguientes parámetros no pueden ser anotados en la base de datos de registro: 1: Tslot 2: Tqui 3: Tset 4: MinTsdr 5: MaxTsdr 6: Trdy 7: Tid1 8: Tid2 9: Gap Factor 10: Retry Limit	Compruebe el perfil de bus definido por el usuario. Compruebe la conexión y verifique si está conectado el controlador. En el Panel de control, compruebe con "Ajustar interface PG/PC" los parámetros configurados. Ejecute un rearranque.
140008	No se actualiza ni se escribe ninguna variable porque los datos de configuración son incorrectos: Los siguientes parámetros no pueden ser anotados en la base de datos de registro: 0: Error general 1: Versión incorrecta 2: El perfil no puede ser anotado en la base de datos de registro. 3: El tipo de subred no puede ser anotado en la base de datos de registro. 4: Target Rotation Time no puede ser anotado en la base de datos de registro. 5: La dirección más alta (HSA) es incorrecta.	Compruebe la conexión y verifique si está conectado el controlador. En el Panel de control, compruebe con "Ajustar interface PG/PC" los parámetros configurados. Ejecute un rearranque.
140009	No se actualiza ni se escribe ninguna variable porque no se ha encontrado el módulo para la comunicación S7.	Con "Ajustar interface PG/PC", vuelva a instalar el módulo en el Panel de control.
140010	No se ha encontrado ningún interlocutor S7 porque el controlador está desconectado. DP/T: En "Ajustar interface PG/PC" del Panel de control no está ajustada la opción "PG/PC es el único maestro del bus".	Conecte el controlador. DP/T: Si en la red hay solo un maestro, en "Ajustar interface PG/PC" active la opción "PG/PC es el único maestro del bus". Si hay más de un maestro en la red, conéctelos. No modifique la configuración, pues de hacerlo se presentarán fallos en el bus.
140011	No se actualiza ni se escribe ninguna variable porque se ha interrumpido la comunicación.	Compruebe la conexión y verifique si está conectado el interlocutor.
140012	Existe un problema de inicialización (p. ej. si se ha finalizado WinCC Runtime en el Administrador de tareas). O bien: Otro programa (p. ej. STEP 7) está activo con otros parámetros de bus y los drivers no pueden ser iniciados con los nuevos parámetros de bus (p. ej. velocidad de transferencia).	Reinicie el panel de operador. O bien: Inicie primero WinCC Runtime y, después, el resto de programas.
140013	El cable MPI no está enchufado y, por consiguiente, no hay suministro de corriente.	Compruebe las conexiones.

Número	Efecto/causas	Solución
140014	La dirección configurada en el bus ya está ocupada.	Durante la configuración modifique en el controlador la dirección del panel de operador.
140015	Velocidad de transferencia incorrecta O bien: Parámetros de bus incorrectos (p. ej. HSA) O bien: Dirección OP > HSA o: Vector de interrupción incorrecto (la interrupción no llega hasta el driver)	Corrija los parámetros incorrectos.
140016	La interrupción configurada no es asistida por el hardware.	Modifique el número de interrupción.
140017	La interrupción configurada está siendo usada por otro driver.	Modifique el número de interrupción.
140018	La comprobación de coherencia ha sido desactivada por SIMOTION Scout. Aparece únicamente un aviso correspondiente.	Active de nuevo la comprobación de coherencia con SIMOTION Scout y cargue otra vez el proyecto en el controlador.
140019	SIMOTION Scout carga un nuevo proyecto en el controlador. Se cancela la conexión con el controlador.	Espere a que se termine el cambio de configuración.
140020	La versión en el controlador no coincide con la versión en la configuración (archivo FWX). Se cancela la conexión con el controlador.	Existen las siguientes soluciones posibles: Con SIMOTION Scout cargue en el controlador la versión actual. Con WinCC ES vuelva a compilar el proyecto, cierre WinCC Runtime y comience con una nueva configuración.
140021	No se ha establecido la conexión con el controlador. La "Contraseña de acceso" para la conexión con el controlador se ha configurado erróneamente.	Compruebe en el editor "Conexiones", en el área "Contraseña de acceso", la contraseña introducida para la conexión con el controlador. Asigne la contraseña correcta. La "Contraseña" para la conexión con el controlador se asigna en las propiedades del controlador en "Protección".
140022	No se ha establecido la conexión con el controlador. No se ha configurado ninguna contraseña de acceso para la conexión con el controlador.	Introduzca en el editor "Conexiones", en el área "Contraseña de acceso", la contraseña para la conexión con el controlador. La "Contraseña" se asigna en las propiedades del controlador en "Protección".
140025	No se ha establecido la conexión con el controlador. Se ha bloqueado la contraseña de acceso del controlador en la pantalla del controlador.	Desbloquee la contraseña de acceso en la pantalla del controlador.

180000 - Avisos generales

Significado de los avisos de sistema

A continuación se representan todos los avisos de sistema que se pueden visualizar.

Tabla 10-9 180000 - Avisos generales

Número	Efecto/causas	Solución
180000	Un componente/OCX ha recibido datos de configuración con una identificación de versión no compatible.	Instale un nuevo componente.
180001	El sistema está sobrecargado debido a que se han activado demasiadas acciones simultáneamente. No todas las acciones pueden ser ejecutadas, algunas serán rechazadas.	Existen diversas posibilidades de ayuda: • Genere los avisos más despacio (sondeo). • Active los scripts y funciones de sistema en intervalos de tiempo más distantes. En caso de que el aviso aparezca con frecuencia: Reinicie el panel de operador.
180002	No se ha podido activar el teclado de pantalla. Causa posible: El archivo "TouchInputPC.exe" no ha sido registrado porque la instalación se ha realizado incorrectamente.	Vuelva a instalar WinCC Runtime.

190000 – Avisos de variables

Significado de los avisos de sistema

A continuación se representan todos los avisos de sistema que se pueden visualizar.

Tabla 10-10 190000 – Avisos de variables

Número	Efecto/causas	Solución
190000	Probablemente la variable no se actualiza.	--
190001	La variable se actualiza otra vez después de un estado incorrecto una vez que se elimina el último estado de error (retorno al funcionamiento normal).	--
190002	La variable no se actualiza porque se ha interrumpido la comunicación con el controlador.	Conecte la comunicación mediante la función de sistema "SetOnline".
190004	La variable no se actualiza porque no existe la dirección configurada para esta variable.	Compruebe la configuración.
190005	La variable no se actualiza porque no existe el tipo de controlador configurado para esta variable.	Compruebe la configuración.
190006	La variable no se actualiza porque no se puede reproducir el tipo de controlador en el tipo de datos de la variable.	Compruebe la configuración.
190007	El valor de la variable no se modifica porque se ha interrumpido la conexión con el controlador o la variable está offline.	Cambie al modo online o restablezca la conexión con el controlador.
190008	Los valores límite configurados de la variable han sido infringidos por uno de estos eventos: • Entrada de valores • Función de sistema • Script	Tenga en cuenta los valores límite configurados o actuales de la variable.

Número	Efecto/causas	Solución
190009	Se ha intentado asignar a la variable un valor que está fuera del rango de valores admisible para este tipo de datos. P..ej. entrada del valor 260 para una variable de byte o entrada del valor -3 para una variable de palabra sin signo.	Tenga en cuenta el rango de valores del tipo de datos de la variable.
190010	Se escriben valores en la variable con demasiada frecuencia (p. ej. en un bucle desde un script). Se pierden valores porque sólo se guardan como máximo 100 procesos en caché.	Existen las siguientes soluciones posibles: • Incremente el tiempo entre varias escrituras. • No utilice variables de matriz de una longitud superior a 6 palabras en acuses para "Variable de acuse HMI" configurados en el panel de operador.
190011	Causa posible 1: El valor introducido no se ha podido escribir en la variable de controlador configurada porque el rango de valores se ha rebasado por exceso o defecto. La entrada se ha rechazado y se ha restablecido el valor original. Causa posible 2: Se ha interrumpido la conexión con el controlador.	Tenga en cuenta que el valor introducido debe estar dentro del rango de valores de la variable de controlador. Compruebe la conexión con el controlador.
190012	No se puede convertir el valor de un formato de origen a un formato de destino, p. ej.: Para un contador se debe escribir un valor que está fuera del rango de valores válido dependiente del controlador. A una variable del tipo Integer se le debe asignar un valor del tipo String.	Compruebe el rango de valores o el tipo de datos de la variable.
190013	El usuario ha introducido un string de una longitud mayor que la variable. El string se acorta automáticamente a la longitud permitida.	Introduzca sólo strings que no excedan la longitud de la variable.

190100 - Avisos del puntero de área

190100 - Avisos del puntero de área

Número	Efecto/causas	Solución
190100	El puntero de área no se actualiza porque no existe la dirección configurada para el mismo. Tipo: 1 Advertencias 2 Errores 3 Acuse de controlador 4 Acuse de panel de operador 5 Imagen LED 6 Solicitud de curva 7 Transferencia de curva 1 8 Transferencia de curva 2 N.º: número correlativo indicado en WinCC ES.	Compruebe la configuración.
190101	El puntero de área no se actualiza porque no se puede reproducir el tipo de controlador en el tipo del puntero de área. Parámetro Tipo y N.º: véase el aviso 190100	--
190102	El puntero de área se actualiza otra vez después de un estado incorrecto porque se ha eliminado el último estado de error (retorno al funcionamiento normal).Parámetro Tipo y Nº: véase el aviso 190100	--

200000 – Avisos de coordinación con el PLC

200000 – Avisos de coordinación con el PLC

Número	Efecto/causas	Solución
200000	La coordinación no se ejecuta porque la dirección ajustada no existe/no se ha configurado en el controlador.	Modifique la dirección o configure la dirección en el controlador.
200001	La coordinación no se ejecuta porque no se puede escribir en la dirección configurada en el controlador.	Modifique la dirección o configure la dirección en el controlador en un área en la que se pueda escribir.
200002	La coordinación no se ejecuta en este momento porque el formato de dirección del puntero de área no coincide con el formato interno de almacenamiento.	Error interno
200003	La coordinación se ejecuta otra vez porque se ha corregido el último estado de error (retorno al funcionamiento normal).	--

Número	Efecto/causas	Solución
200004	Probablemente no se ejecutará la coordinación.	--
200005	Ya no se escriben ni se leen más datos. Causas posibles: • El cable está interrumpido. • El controlador no reacciona, está defectuoso, etc. • El sistema está sobrecargado.	Compruebe si el cable está enchufado o si el controlador funciona correctamente. Reinicie el sistema en caso de que el aviso de sistema persista por tiempo prolongado.

210000 – Avisos de PLC-Job

210000 - Avisos de PLC-Job

Número	Efecto/causas	Solución
210000	Las peticiones no se procesan porque la dirección ajustada no existe/no se ha configurado en el controlador.	Modifique la dirección o configure la dirección en el controlador.
210001	Las peticiones no se procesan porque la dirección configurada no se puede leer/escribir en el controlador.	Modifique la dirección o configure la dirección en el controlador en una zona en la que se pueda leer/escribir.
210002	Las peticiones no se ejecutan porque el formato de dirección del puntero de área no coincide con el formato interno de almacenamiento.	Error interno
210003	El buzón de tareas se procesa otra vez porque se ha eliminado el último estado de error (retorno al funcionamiento normal).	--
210004	Probablemente no se procesa el buzón de tareas.	--
210005	Se ha activado una petición de control con un número inadmisible.	Compruebe el programa de control.
210006	Se ha presentado un error durante la ejecución de la petición de control. Por esta razón la petición de control no se ejecuta. En caso necesario, considere también el aviso de sistema siguiente/anterior.	Compruebe los parámetros de la petición de control. Vuelva a compilar la configuración.

220000 - Avisos del driver de comunicación de WinCC

220000 - Avisos del driver de comunicación de WinCC

Número	Efecto/causas	Solución
220001	La variable no se transfiere porque el driver de comunicación subordinado/panel de operador no soporta el tipo de datos Bool/Bit al escribir.	Modifique la configuración.
220002	La variable no se transfiere porque el driver de comunicación subordinado/panel de operador no soporta el tipo de datos Byte al escribir.	Modifique la configuración.
220003	No se ha podido cargar el driver de comunicación. Probablemente no está instalado.	Instale el driver reinstalando WinCC Runtime.
220004	La comunicación se ha interrumpido; no se realiza ninguna actualización porque el cable no está enchufado o está defectuoso, etc.	Compruebe la conexión.
220005	Se está ejecutando la comunicación.	--
220006	En el puerto indicado se ha establecido la conexión con el controlador indicado.	--
220007	Se ha interrumpido la conexión con el controlador indicado en el puerto indicado.	Compruebe lo siguiente: • si el cable está enchufado • si el controlador funciona correctamente • si se utiliza el puerto correcto • si la configuración es correcta (parámetros del puerto, configuración del informe, dirección del controlador) Reinicie el sistema en caso de que el aviso de sistema permanezca invariable por tiempo prolongado.
220008	El driver de comunicación no puede acceder o abrir el puerto indicado. Posiblemente otro programa esté utilizando ya este puerto o se está utilizando un puerto no existente en el dispositivo de destino. No hay comunicación con el controlador.	Cierre todos los programas que acceden al puerto y reinicie el equipo. Utilice otro puerto disponible en el sistema.

230000 - Avisos de objetos de imagen

230000 - Avisos de objetos de imagen

Número	Efecto/causas	Solución
230000	No se ha podido aceptar el valor introducido. La entrada se rechaza y el valor anterior se restablece. Causa posible: • El rango de valores se ha rebasado por exceso • Se han introducido caracteres no válidos • Se ha excedido el número máximo admisible de usuarios.	Introduzca un valor apropiado o borre un usuario que ya no se necesite.
230002	Como el usuario que ha iniciado la sesión no tiene la autorización requerida, se rechaza la entrada y se restablece el valor anterior.	Inicie la sesión como usuario con la autorización adecuada.
230003	No se realiza el cambio a la imagen indicada porque la imagen no existe/no se ha configurado. La imagen seleccionada hasta ahora permanece activa.	Configure la imagen y compruebe la función de selección.
230005	Se ha excedido el rango de valores de la variable en el campo E/S. Se conserva el valor original de la variable.	Tenga en cuenta el rango de valores de la variable al introducirla.
230100	Después de la navegación con el navegador web se ha devuelto un aviso que podría ser de interés para el usuario. El navegador web sigue ejecutándose, pero es posible que no se visualice el nuevo sitio (de forma completa).	Navegue a un sitio diferente.
230200	La conexión del canal HTTP se ha interrumpido porque se ha presentado un error. Este error se explicará más detalladamente con otro aviso de sistema. Ya no se intercambian más datos.	Compruebe la conexión de red. Compruebe la configuración del servidor.
230201	Se ha establecido la conexión del canal HTTP. Se intercambian datos.	--

Número	Efecto/causas	Solución
230202	WININET.DLL ha detectado un error. Este error se presenta casi siempre cuando no se puede establecer una conexión con el servidor, o cuando el servidor rechaza una conexión porque el cliente no tiene la autorización correcta. En el caso de una conexión encriptada vía SSL la causa puede ser un certificado de servidor no aceptado. En el texto de error del aviso encontrará explicaciones más detalladas. Este texto aparece siempre en el idioma de la instalación de Windows, ya que es suministrado por Windows. No se intercambia ningún valor de proceso. Posiblemente no se visualiza la parte del aviso que se transfiere desde el sistema operativo Windows, p. ej. "Ha ocurrido un error. WININET.DLL proporciona estos errores: número: 12055 texto:HTTP: <no error text available>."	En función de la causa: Si la conexión no puede ser establecida o si ocurre un timeout: • Compruebe la conexión de la red y la red. • Compruebe la dirección del servidor. • Compruebe si el WebServer se ejecuta realmente en el equipo de destino. Si la autorización es incorrecta: • El nombre de usuario configurado o la contraseña no coinciden con los del servidor. Haga que los datos coincidan. Si no se acepta el certificado del servidor: El certificado ha sido firmado por un CA desconocido ( ): • Ignore este punto o instale un certificado que haya sido firmado por un certificado Root conocido por el equipo cliente. Si la fecha del certificado no es válida: • Ignore este punto o instale un certificado con fecha válida en el servidor. Si CN (Common Name o Computer Name) no es válido: • Ignore este punto o instale un certificado con un nombre correspondiente a la dirección del servidor.
230203	A pesar de que se puede establecer una conexión con el servidor, el servidor HTTP ha rechazado la conexión. Causas posibles: • WinCC Runtime no se ejecuta en el servidor • No se asiste el canal HTTP (503 Service unavailable). Sólo pueden aparecer otros errores si el Webserver no soporta el canal HTTP. El idioma del texto de error depende del Webserver. No se intercambian datos.	Si ocurre el error 503 Service unavailable: Compruebe si WinCC Runtime se está ejecutando en el servidor y si se soporta el canal HTTP.
230301	Ha ocurrido un error interno. Un texto en inglés explica más detalladamente el error del aviso. Una posible causa puede ser p. ej. que no hay suficiente memoria. El OCX no funciona.	--
230302	El nombre del servidor remoto no puede identificarse. No se puede establecer ninguna conexión.	Compruebe la dirección de servidor configurada. Compruebe si el servicio DNS de la red está activo.
230303	El servidor remoto no está activo en el equipo direccionado. La dirección del servidor es incorrecta. No se puede establecer ninguna conexión.	Compruebe la dirección de servidor configurada. Vigile si está funcionando el servidor remoto del equipo de destino.
230304	El servidor remoto del equipo direccionado no es compatible con VNCOCX. No se puede establecer ninguna conexión.	Utilice un servidor remoto compatible.

Número	Efecto/causas	Solución
230305	La autenticación ha fallado porque la contraseña es incorrecta. No se puede establecer ninguna conexión.	Configure la contraseña correcta.
230306	Falla la conexión con el servidor remoto. Esto se puede presentar cuando existen problemas con la red. No se puede establecer ninguna conexión.	Compruebe si el cable de red está enchufado o si hay problemas con la red.
230307	El servidor remoto ha terminado la conexión. Causas posibles: • Se ha cerrado el servidor remoto • El usuario ha indicado al servidor que cierre todas las conexiones. La conexión se cancela.	--
230308	Este aviso le informa sobre el establecimiento de la conexión. En estos momentos se está estableciendo una conexión.	--

260000 – Avisos de sistema de contraseñas

260000 – Avisos del sistema de contraseñas

Número	Efecto/causas	Solución
260000	En el sistema se ha introducido un usuario desconocido o una contraseña desconocida. El sistema cierra la sesión del usuario actual.	Inicie la sesión en el sistema como usuario con una contraseña válida.
260001	El usuario que ha iniciado la sesión no tiene una autorización adecuada para ejecutar la funcionalidad protegida.	Inicie la sesión en el sistema como usuario que dispone de la autorización adecuada.
260002	Este aviso se emite cuando se activa la función de sistema "IdentificarCambioDeUsuario".	--
260003	El usuario ha cerrado la sesión en el sistema.	--
260004	El nuevo nombre de usuario introducido en el visor de usuarios ya existe en la administración de usuarios.	Seleccione un nombre de usuario diferente, ya que los nombres de usuarios deben ser unívocos en la administración de usuarios.
260005	La entrada se rechaza.	Introducir un nombre de usuario más corto.
260006	La entrada se rechaza.	Introducir una contraseña más corta o una más larga.
260007	El tiempo de cierre de sesión introducido está fuera del rango válido de 0 a 60 minutos. El valor introducido será rechazado y se conservará el valor original.	Introduzca un valor entre 0 y 60 minutos para el tiempo de cierre de sesión.
260008	Se ha intentado leer en WinCC un archivo PTProRun.pwl creado con ProTool V 6.0. Se ha cancelado la lectura del archivo por incompatibilidad de formato.	--

Número	Efecto/causas	Solución
260009	Ha intentado borrar el usuario "Administrador" o "PLC User". Estos dos usuarios son componentes fijos de la administración de usuarios, por lo que no pueden borrarse.	Si fuese necesario borrar un usuario (p. ej. al alcanzarse la cantidad máxima de usuarios), borre un usuario diferente.
260012	No concuerdan la contraseña introducida en el cuadro de diálogo "Cambiar contraseña" y su confirmación. La contraseña no se cambiará. Se cierra la sesión del usuario actual.	Inicie de nuevo la sesión en el sistema. Introduzca luego dos contraseñas idénticas para cambiar la contraseña antigua.
260013	La contraseña introducida en el cuadro de diálogo "Cambiar contraseña" ya se ha utilizado, por lo que no es válida. La contraseña no se cambiará. Se cierra la sesión del usuario actual.	Inicie de nuevo la sesión en el sistema. Introduzca luego una nueva contraseña que no se haya utilizado todavía.
260014	Ha intentado iniciar sesión tres veces consecutivas con una contraseña errónea. Se bloqueará su acceso al sistema y se le asignará al grupo 0.	Puede iniciar la sesión introduciendo su contraseña correcta. No obstante, el administrador es el único que puede modificar la asignación a grupos.
260024	La contraseña introducida no cumple con las directivas de seguridad necesarias.	Introduzca una contraseña que contenga al menos un número.
260025	La contraseña introducida no cumple con las directivas de seguridad necesarias.	Introduzca una contraseña que contenga al menos tres caracteres.
260028	Se ha intentado acceder al SIMATIC Logon Server durante el arranque del sistema, al intentar iniciar la sesión, o bien al intentar cambiar la contraseña de un usuario de SIMATIC Logon. Al intentar iniciar la sesión, el nuevo usuario no se da de alta. Si un usuario diferente se había dado de alta anteriormente, se cierra la sesión de éste.	Compruebe la conexión con el SIMATIC Logon Server y la configuración de éste, por ejemplo: 1. Número de puerto 2. Dirección IP 3. Nombre de servidor 4. Cable de transferencia apto. Como alternativa, utilice un usuario local.
260030	El usuario de SIMATIC Logon no ha podido cambiar su contraseña en el SIMATIC Logon Server. Puede ser que la nueva contraseña no cumpla las convenciones de contraseñas en el servidor o que el usuario no esté autorizado a cambiar su contraseña. Se conserva la contraseña antigua y se cierra la sesión del usuario.	Vuelva a iniciar la sesión con una contraseña diferente. Compruebe las convenciones de contraseñas en el SIMATIC Logon Server.
260033	No se ha podido cambiar la contraseña o iniciar la sesión del usuario.	Compruebe la conexión con el SIMATIC Logon Server y la configuración de éste, por ejemplo: 1. Número de puerto 2. Dirección IP 3. Nombre de servidor 4. Cable de transferencia capaz de funcionar Como alternativa, utilice un usuario local.

Número	Efecto/causas	Solución
260034	El último inicio de sesión no ha concluido todavía. Por tanto, el usuario no puede realizar ninguna acción ni tampoco abrir un cuadro de diálogo de inicio de sesión. El cuadro de diálogo de inicio de sesión no se abre. La acción del usuario no se ejecuta.	Espere hasta que haya concluido el inicio de sesión.
260035	No ha concluido todavía el último intento de cambiar la contraseña. Por tanto, el usuario no puede realizar ninguna acción ni tampoco abrir un cuadro de diálogo de inicio de sesión. El cuadro de diálogo de inicio de sesión no se abre. La acción del usuario no se ejecuta.	Espere hasta que haya concluido el proceso.
260036	Faltan licencias en el SIMATIC Logon Server. La sesión se puede iniciar todavía.	Compruebe las licencias en el SIMATIC Logon Server.
260037	No hay licencias disponibles en el SIMATIC Logon Server. La sesión no se puede iniciar. La sesión no se puede iniciar a través del SIMATIC Logon Server, sino sólo como usuario local.	Compruebe las licencias en el SIMATIC Logon Server.
260040	Se ha intentado acceder al SIMATIC Logon Server al intentar iniciar la sesión o al intentar cambiar la contraseña de un usuario de SIMATIC Logon. Al intentar iniciar la sesión, el nuevo usuario no se da de alta. Si un usuario diferente se había dado de alta anteriormente, se cierra la sesión de éste.	Compruebe la conexión con el dominio y su configuración en el editor Configuración de seguridad de runtime. Como alternativa, utilice un usuario local.
260043	No se ha podido iniciar la sesión del usuario en el SIMATIC Logon Server. Puede ser que sean incorrectos el nombre de usuario o la contraseña, o bien que el usuario no esté autorizado para iniciar la sesión. No se inicia la sesión del nuevo usuario. Si un usuario diferente se había dado de alta anteriormente, se cierra la sesión de éste.	Vuelva a intentarlo. Dado el caso, compruebe los datos de usuario en el SIMATIC Logon Server.
260044	No se ha podido iniciar la sesión del usuario en el SIMATIC Logon Server porque su cuenta está bloqueada. No se inicia la sesión del nuevo usuario. Si un usuario diferente se había dado de alta anteriormente, se cierra la sesión de éste.	Compruebe los datos de usuario en el SIMATIC Logon Server.
260045	El usuario de SIMATIC Logon pertenece a varios grupos, o bien a ningún grupo. No se inicia la sesión del nuevo usuario. Si un usuario diferente se había dado de alta anteriormente, se cierra la sesión de éste.	Compruebe los datos de usuario en el SIMATIC Logon Server y la configuración en el proyecto de WinCC. Un usuario sólo puede estar asignado a un grupo.

270000 – Avisos de sistema

270000 - Avisos de sistema

Número	Efecto/causas	Solución
270000	Una variable no se visualiza en el aviso porque accede a una dirección no válida en el controlador.	Compruebe si en el controlador existe el rango de datos para la variable, si la dirección configurada es correcta o si el rango de valores de la variable concuerda.
270001	Existe un número máximo de avisos (que depende del dispositivo) que pueden estar pendientes de visualización al mismo tiempo (véanse las instrucciones de servicio). Este número se ha excedido. La visualización ya no contiene todos los avisos. Sin embargo, en el búfer de avisos se anotan todos los avisos.	--
270002	Se visualizan avisos de un fichero para los que no existen datos en el proyecto actual. Se visualizan comodines para los avisos.	En caso necesario, borre datos antiguos de ficheros.
270003	Imposible configurar el servicio porque demasiados dispositivos desean inicializar este servicio. Cuatro dispositivos como máximo pueden ejecutar esta acción.	Conecte menos paneles de operador que deban usar el servicio.
270004	Imposible acceder al búfer de avisos permanente. Los avisos no se pueden restablecer ni almacenar.	Si el problema persiste en el próximo reinicio, diríjase al Customer Support (servicio de atención al cliente) (borrar flash).
270005	El búfer de avisos permanente está dañado: no es posible restablecer los avisos.	Si el problema persiste en el próximo reinicio, diríjase al Customer Support (servicio de atención al cliente) (borrar flash).
270006	El proyecto se ha modificado: los avisos no se pueden restablecer desde el búfer de aviso permanente.	El proyecto se ha compilado y se ha cargado de nuevo en el panel de operador. El error ya no debe ocurrir en el siguiente arranque del panel de operador.
270007	Un problema de configuración impide restablecer los datos (p. ej. se ha borrado un archivo DLL, se ha cambiado el nombre de un directorio, etc.).	Actualice el sistema operativo y vuelva a cargar el proyecto en el panel de operador.

290000 – Avisos del sistema de recetas

290000 - Avisos del sistema de recetas

Número	Efecto/causas	Solución
290000	No se ha podido leer ni escribir la variable de receta. Recibirá el valor inicial. El aviso se registrará, dado el caso, para otras cuatro variables incorrectas como máximo en el búfer de avisos. Después se visualiza el aviso n.º 290003.	Compruebe en la configuración si se ha configurado la dirección en el controlador.
290001	Se ha intentado asignar a la variable de receta un valor que está fuera del rango de valores admisible para este tipo. El aviso se registra, en caso necesario, en otras cuatro variables incorrectas como máximo en el búfer de avisos. Después se visualiza el aviso n.º 290004.	Tenga en cuenta el rango de valores del tipo de variables.
290002	No se puede convertir el valor de un formato de origen a un formato de destino. El aviso se registra, en caso necesario, en otras cuatro variables de receta incorrectas como máximo en el búfer de avisos. Después se visualiza el aviso n.º 290005.	Compruebe el rango de valores o el tipo de las variables.
290003	Este aviso se visualiza cuando el aviso n.º 290000 se ha activado más de cinco veces. En este caso ya no se generan avisos individuales.	Compruebe en la configuración si se han ajustado las direcciones de las variables en el controlador.
290004	Este aviso se visualiza cuando el aviso n.º 290001 se ha activado más de cinco veces. En este caso ya no se generan avisos individuales.	Tenga en cuenta el rango de valores del tipo de variables.
290005	Este aviso se visualiza cuando el aviso n.º 290002 se ha activado más de cinco veces. En este caso ya no se generan avisos individuales.	Compruebe el rango de valores o el tipo de las variables.
290006	Los valores límite configurados de las variables se han infringido debido a la entrada de valores.	Tenga en cuenta los valores límite configurados o actuales de la variable.
290007	Existe una diferencia entre la estructura de origen y la estructura de destino en la receta que se procesa actualmente. La estructura de destino contiene una variable de receta adicional que no existe en la estructura de origen. La variable de receta indicada recibe su valor inicial.	Incluya la variable de receta indicada en la estructura de origen.

Número	Efecto/causas	Solución
290008	Existe una diferencia entre la estructura de origen y la estructura de destino en la receta que se procesa actualmente. La estructura de origen contiene una variable de receta adicional que no existe en la estructura de destino y por eso no puede asignarse. El valor se rechaza.	En la configuración elimine la variable de receta indicada de la receta indicada.
290010	La ubicación configurada para la receta no es admisible. Causas posibles: Caracteres inadmisibles, sólo lectura, soporte de datos lleno o no existente.	Compruebe la ubicación configurada.
290011	El registro con el número indicado no existe.	Compruebe el origen del número (valor constante o variable).
290012	La receta con el número indicado no existe.	Compruebe el origen del número (valor constante o variable).
290013	Se ha intentado guardar un registro con un número de registro ya existente. La operación no se ejecuta.	Existen las siguientes soluciones posibles: • Compruebe el origen del número (valor constante o variable). • Borre previamente el registro. • Modifique el parámetro de función "Sobrescribir".
290014	No se ha encontrado el archivo indicado que debe importarse.	Compruebe lo siguiente: • Compruebe el nombre del archivo. • Asegúrese de que el archivo está en el directorio indicado.
290020	Confirmación de que se ha iniciado la transferencia de registros desde el panel de operador al controlador.	--
290021	Confirmación de que la transferencia de registros desde el panel de operador al controlador ha finalizado sin errores.	--
290022	Confirmación de que la transferencia de registros desde el panel de operador al controlador se ha cancelado con errores.	En la configuración compruebe lo siguiente: • si se han configurado las direcciones de las variables en el controlador. • si existe el número de receta. • si existe el número de registro. • si se ha configurado el parámetro de función "Sobrescribir".
290023	Confirmación de que se ha iniciado la transferencia de registros desde el controlador al panel de operador.	--
290024	Confirmación de que la transferencia de registros desde el controlador al panel de operador ha finalizado sin errores.	---

Número	Efecto/causas	Solución
290025	Confirmación de que la transferencia de registros desde el controlador al panel de operador se ha cancelado con errores.	En la configuración compruebe lo siguiente: • si se han configurado las direcciones de las variables en el controlador. • si existe el número de receta. • si existe el número de registro. • si se ha configurado el parámetro de función "Sobrescribir".
290026	Se ha intentado leer/escribir un registro aunque el buzón de datos no está libre en estos momentos. Este error se puede presentar en recetas para las que se ha configurado una transferencia con sincronización.	Ponga el estado a cero en el buzón de datos.
290027	Actualmente no se puede establecer ninguna conexión con el controlador. Por esta razón no se puede leer ni escribir el registro. Causas posibles: No hay conexión física con el controlador (cable no enchufado, cable defectuoso) o el controlador está desconectado.	Compruebe la conexión con el controlador.
290030	Este aviso se visualiza tras volver a seleccionar una imagen que contiene un visor de recetas en el que ya se ha seleccionado un registro.	Cargue otra vez el registro existente en la ubicación o conserve los valores actuales.
290031	Al guardar se ha detectado que ya existe un registro con el número indicado.	Sobrescriba el registro o cancele la operación.
290032	Al exportar registros se ha detectado que ya existe un archivo con el nombre indicado.	Sobrescriba el archivo o cancele la operación.
290033	Consulta de seguridad antes de borrar registros.	--
290040	Se ha presentado un error de registro, que no tiene que especificarse más detalladamente, con el código de error %1. La acción se cancela. Probablemente en el controlador no se ha configurado correctamente el buzón de datos.	Compruebe la ubicación, el registro, el puntero de área "Registro" y, de ser necesario, la conexión con el controlador. Espere unos instantes y lance la acción de nuevo. Si persiste el error, póngase en contacto con el Customer Support (servicio de atención al cliente). Al hacerlo indique el código de error que se ha presentado.
290041	No se puede guardar un registro o archivo porque la ubicación está llena.	Borre los archivos que ya no se necesitan.
290042	Se ha tratado de ejecutar al mismo tiempo varias acciones de recetas. No se ejecuta la última acción.	Después de un breve tiempo de espera active otra vez la acción.
290043	Consulta de seguridad antes de guardar registros.	--
290044	La ubicación para la receta está destruida y será borrada.	--
290050	Confirmación de que se ha iniciado la exportación de registros.	--
290051	Confirmación de que la exportación de registros ha finalizado sin errores.	--

Número	Efecto/causas	Solución
290052	Confirmación de que la exportación de registros se ha cancelado con errores.	Asegúrese de que la estructura de los registros en la ubicación y la estructura actual de la receta en el panel de operador son idénticas.
290053	Confirmación de que se ha iniciado la importación de registros.	--
290054	Confirmación de que la importación de registros ha finalizado sin errores.	--
290055	Confirmación de que la importación de registros se ha cancelado con errores.	Asegúrese de que la estructura de los registros en la ubicación y la estructura actual de la receta en el panel de operador son idénticas.
290056	No se ha podido leer/escribir sin errores el valor indicado en la fila/columna. La acción se cancela.	Compruebe la fila/columna indicada.
290057	Las variables de la receta indicada se han conmutado del modo de operación "Offline" al modo "Online". Cualquier modificación de una variable de esta receta se transfiere inmediatamente al controlador.	--
290058	Las variables de la receta indicada se han conmutado del modo de operación "Online" al modo "Offline". Las modificaciones de variables de esta receta ya no se transfieren inmediatamente al controlador sino, dado el caso, se transfieren explícitamente al controlador mediante una transferencia de registros.	--
290059	Confirmación de que el registro indicado se ha guardado correctamente.	--
290060	Confirmación de que la memoria de registros se ha borrado correctamente.	--
290061	Confirmación de que el borrado de la memoria de registros se ha cancelado con errores.	--
290062	El número máximo de registros es de más de 65536. Este registro no puede crearse.	Seleccione un número diferente.
290063	Se presenta con la función de sistema "ExportarRegistros" con el parámetro "Sobrescribir" puesto en "No". Se ha intentado guardar una receta con un nombre de archivo que ya existe. La exportación se cancela.	Compruebe los parámetros de la función de sistema "ExportarRegistros".
290064	Confirmación de que se han comenzado a borrar los registros.	--
290065	Confirmación de que los registros se han borrado correctamente.	--
290066	Consulta de seguridad antes de borrar registros.	--
290068	Consulta de seguridad de si se deben borrar todos los registros de la receta.	--

Número	Efecto/causas	Solución
290069	Consulta de seguridad de si se deben borrar todos los registros de la receta.	--
290070	El registro especificado no existe en el archivo de importación.	Compruebe el origen del número o nombre del registro (valor constante o variable)
290071	Al editar los valores de registros se ha introducido un valor inferior al valor límite inferior de la variable de receta. La entrada se rechaza.	Introduzca un valor dentro de los valores límite de la variable de receta.
290072	Al editar los valores de registros se ha introducido un valor mayor que el valor límite superior de la variable de receta. La entrada se rechaza.	Introduzca un valor dentro de los valores límite de la variable de receta.
290073	Por razones desconocidas no se ha podido ejecutar una acción (p. ej. guardar un registro). El error corresponde al aviso de estado IDS_OUT_CMD_EXE_ERR del visor de recetas grande.	--
290074	Al guardar se ha detectado que ya existe un registro con el número indicado pero con otro nombre.	Sobrescriba el registro, modifique el número de registro o cancele la operación.
290075	Ya existe un registro con este nombre. Se cancelará el almacenamiento del registro.	Seleccione otro nombre de registro.
290110	La activación de los valores estándar se ha cancelado debido a un error.	--
290111	No es posible utilizar el subsistema de recetas. Los visores de recetas están vacíos y las funciones relativas a las recetas no se ejecutan. Causas posibles: • Ha ocurrido un error al cargar las recetas. • La estructura de las recetas se ha modificado en el ES. El proyecto se ha descargado de nuevo, pero las recetas no se han transferido junto con el proyecto. Por tanto, los nuevos datos de configuración ya no concuerdan con las recetas antiguas en el panel de operador.	Cargue de nuevo el proyecto junto con las recetas en el panel de operador (la correspondiente casilla de verificación debe estar activada en el cuadro de diálogo de carga).

10.3.5 Configurar diagnóstico de sistema

10.3.5.1 Principios básicos del diagnóstico de sistema

Introducción

Con el diagnóstico de sistema se pueden mostrar avisos del búfer de diagnóstico de todas las conexiones integradas.

Visor de diagnóstico del sistema

El visor de diagnóstico del sistema es un objeto de control y visualización que se utiliza en una imagen.

Permite navegar directamente hasta la causa de un error y la conexión correspondiente. Permite acceder a todas las conexiones integradas que ha configurado en el editor "Dispositivos y redes".

10.3.5.2 Vistas en el diagnóstico de sistema

Introducción

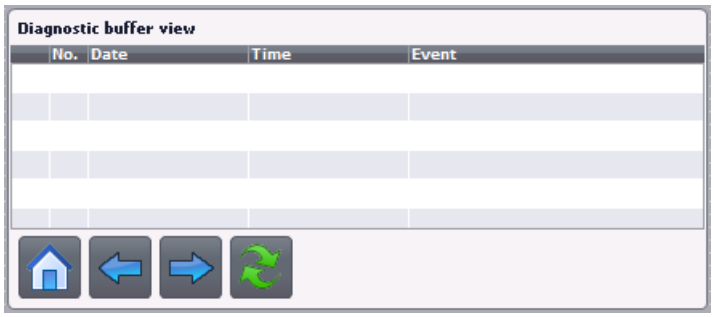
En el visor de diagnóstico del sistema simple se dispone de tres vistas diferentes.

- Vista de dispositivos
- Vista del búfer de diagnóstico
- Vista detallada

Vista de dispositivos

La vista de dispositivos sólo se muestra, si se ha configurado más de una conexión integrada.

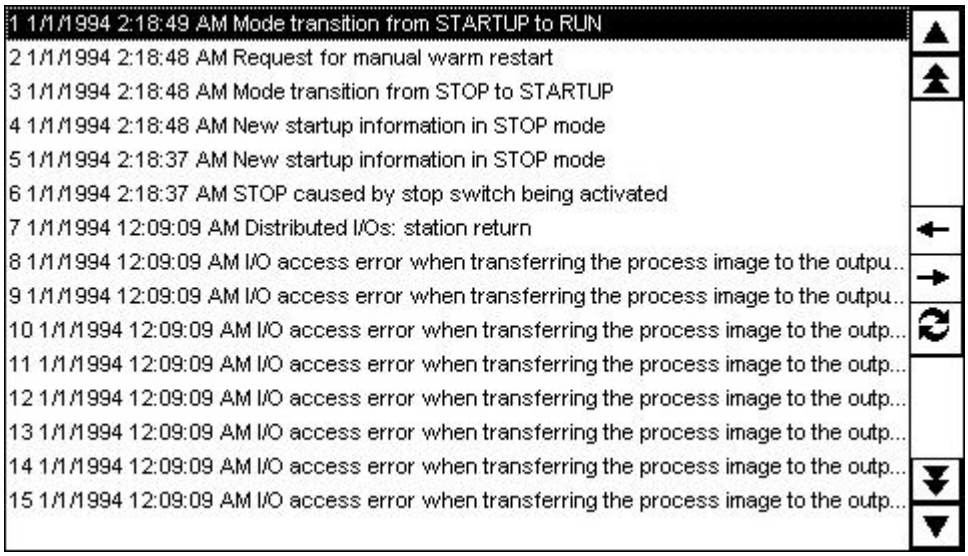
La vista de dispositivos muestra una tabla con todas las conexiones disponibles. Haciendo doble clic sobre una conexión se abre la vista del búfer de diagnóstico.



Vista del búfer de diagnóstico

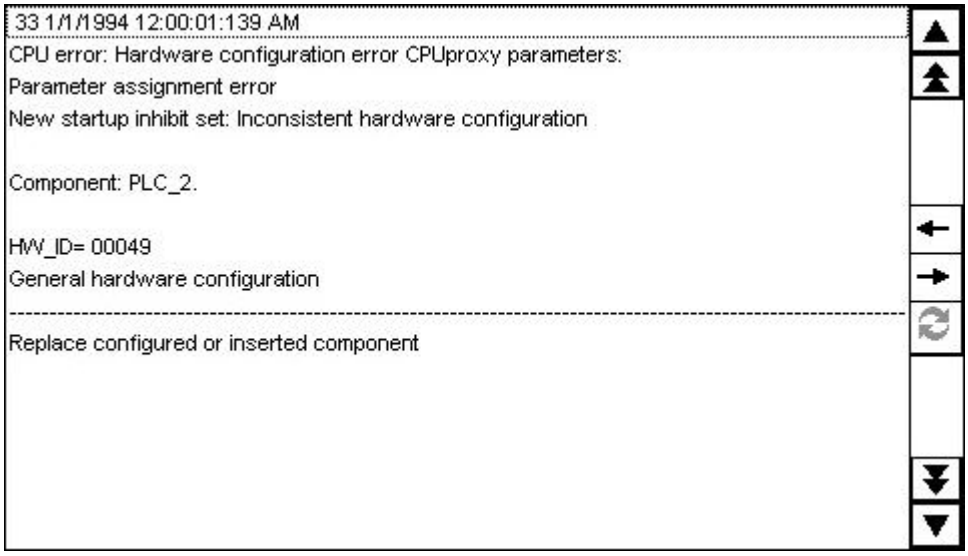
La vista del búfer de diagnóstico muestra los datos actuales del búfer de diagnóstico.

Para actualizar la vista del búfer de diagnóstico, seleccione el botón .



Vista detallada

La vista detallada muestra información concreta acerca de la conexión seleccionada y de los errores existentes. En la vista detallada se comprueba si los datos son correctos.



Nota

Contenidos de la vista detallada

Los contenidos de la vista detallada sólo están disponibles para conexiones integradas con los controladores S7 1200 y S7 1500.

Botones de navegación

Botón	Tecla	Función
	Tecla Intro	En la vista de dispositivos: Abre la vista del búfer de diagnóstico del dispositivo seleccionado. En la vista del búfer de diagnóstico: Abre la vista detallada.
	Tecla ESC	En la vista del búfer de diagnóstico: Abre la vista de dispositivos. En la vista detallada: Abre la vista del búfer de diagnóstico.
	Tecla de función configurada, p. ej. F1.	Actualiza la vista del búfer de diagnóstico.

Consulte también

Visor de diagnóstico del sistema (Página 8451)

Configurar el visor de diagnóstico del sistema (Página 8568)

10.3.5.3 Configurar el visor de diagnóstico del sistema

Introducción

Para tener un resumen de todas las conexiones integradas, inserte un visor de diagnóstico del sistema en el proyecto.

Requisitos

- Se ha creado un controlador.
- Se ha creado un Basic Panel.
- Se ha creado una conexión integrada en el editor "Dispositivos y redes".
- Se ha creado una imagen.
- La ventana de inspección está abierta.

Procedimiento

1. En la Task Card "Herramientas" haga doble clic en el objeto "Visor de diagnóstico del sistema". El objeto se inserta en la imagen.
2. En la ventana de inspección, haga clic en "Propiedades > Representación".
3. Seleccione para "Filas por entrada", p. ej. 5.

Resultado

El visor de diagnóstico del sistema está insertado en la imagen.

Para ver los avisos actuales, actualice el búfer de diagnóstico.

Consulte también

Vistas en el diagnóstico de sistema (Página 8564)

10.4 Trabajar con recetas

10.4.1 Principios básicos

10.4.1.1 Definición y campos de aplicación

Introducción

En las recetas se recogen los datos afines, p. ej., juegos de parámetros de máquinas o los datos de producción.

Ejemplos:

- Parametrizaciones de máquinas necesarias para conmutar a la fabricación de una variante de producto diferente.
- Ingredientes que se combinan en diferentes proporciones para crear distintos productos finales.

Una receta se caracteriza por una estructura de datos fija. Durante la configuración se define la estructura de una receta. Una receta contiene registros que se distinguen por sus valores, pero no por su estructura.

Las recetas se guardan en el panel de operador. Básicamente, un registro de receta se transfiere por completo en un paso de trabajo entre el panel de operador y el controlador.

Nota

Limitaciones en la importación/exportación

No es posible exportar ni importar recetas en los Basic Panels.

Los datos de receta completos pueden exportarse e importarse con ProSave en formato CSV y transferirse al panel de operador; sin embargo, no es posible hacerlo con registros de receta concretos. El runtime se detiene durante este proceso.

Utilización de recetas

Para el uso de recetas se distinguen p. ej. los siguientes casos:

- Producción manual
  El usuario selecciona los datos de receta deseados y los visualiza en el panel de operador. En caso necesario, modifica los datos de receta y los guarda en el panel de operador. Luego transfiere los datos de receta al controlador.
- Producción automática
  El programa de control inicia la transferencia de los datos de receta entre el controlador y el panel de operador. El usuario también puede iniciar la transferencia realizando una acción en el panel de operador. La producción cambia entonces de forma automática. No es imprescindible visualizar ni modificar los datos.
- Modo "Teach-In"
  El usuario optimiza manualmente los datos de producción en la instalación, p. ej. las posiciones de los ejes o los niveles de llenado. Los valores calculados se transfieren al panel de operador y se guardan en un registro de receta. Posteriormente, el usuario retransfiere al controlador los datos de receta almacenados.

Introducir y modificar los datos de receta

El usuario introduce los datos en los registros de receta y los modifica en caso necesario. Para ello dispone de varias posibilidades, a saber:

- Introducir los datos durante la configuración
  Si los datos de producción ya están definidos, introdúzcalos ya durante la configuración de la receta en el editor "Recetas".
- Introducir los datos en runtime
  Si los datos de producción se deben modificar con frecuencia, modifíquelos directamente en runtime:
  - El usuario introduce los datos directamente en el panel de operador.
  - El usuario configura los parámetros directamente en la máquina. Posteriormente, transfiere los datos del controlador al panel de operador y los guarda en la receta.

10.4.1.2 Ejemplos de utilización de recetas

Las recetas se utilizan p. ej. en la industria manufacturera y en la ingeniería mecánica. En los ejemplos siguientes se describen casos de aplicación típicos que se pueden realizar con ayuda de las funciones de recetas de WinCC:

- Parametrización de máquinas
  La parametrización de máquinas en la industria de procesos es uno de los campos de aplicación de las recetas. Una máquina corta tableros de madera a un tamaño determinado y realiza una serie de taladros. Dependiendo del tamaño del tablero, las guías y los taladros se desplazan a nuevas posiciones. Los datos de posición necesarios están guardados en forma de registros de receta. En modo "Teach-In", reparametrice la máquina p. ej. cuando se deba mecanizar un nuevo tamaño de plancha de madera. Transfiera los nuevos datos de posición directamente del controlador al panel de operador y guárdelos en un registro nuevo.
- Producción por lotes
  Un ejemplo adicional de aplicación de recetas lo constituye la producción por lotes en la industria alimentaria: En una planta embotelladora de zumos de fruta se fabrican zumos, néctares y bebidas de distintos sabores. Los ingredientes utilizados son siempre idénticos, diferenciándose sólo por la proporción de la mezcla. Cada sabor equivale a una receta. Cada proporción de mezcla equivale a un registro. Todos los datos necesarios para una proporción de mezcla se pueden transferir al control de la máquina pulsando un botón.

10.4.1.3 Estructura de las recetas

Introducción

La estructura básica de una receta se representa tomando como ejemplo una planta embotelladora de zumos de fruta.

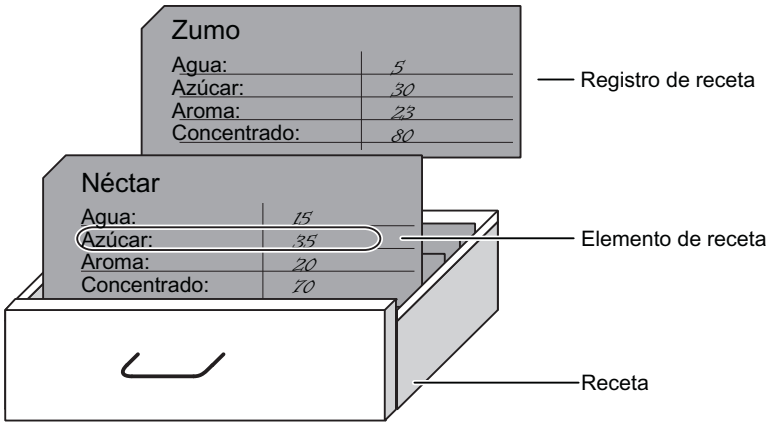
En un panel de operador pueden existir varias recetas diferentes. Una receta se puede comparar p. ej. con un fichero que contiene varias fichas. El fichero comprende diversas variantes de fabricación de una familia de productos. Los datos de una variante de fabricación se recogen íntegramente en una ficha.

Ejemplo:

En una planta embotelladora de bebidas se requiere una receta para las distintas variantes de bebidas. Las variantes de bebidas son p. ej. "Refresco", "Zumo" y "Néctar".

Receta

La receta contiene todos los registros de receta, p. ej. para las distintas variantes de refrescos:



Registros de receta

Cada ficha representa un registro de receta necesario para la fabricación de una variante del producto.

Elementos de las recetas

Todas las fichas de un cajón tienen la misma estructura. Todas las fichas contienen campos para los diversos ingredientes. Cada campo equivale a un elemento de la receta. Por tanto, todos los registros de una receta contienen elementos idénticos. No obstante, los registros se diferencian en el valor de los elementos individuales.

Ejemplo:

Todos los refrescos contienen los ingredientes siguientes:

- Agua
- Concentrado
- Azúcar
- Aroma

No obstante, los registros correspondientes a las variantes "Refresco", "Zumo" y "Néctar" difieren p. ej. en la cantidad de azúcar utilizada para su fabricación.

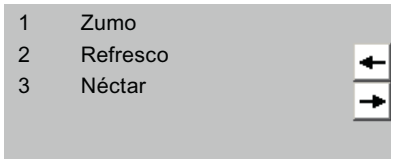
10.4.1.4 Visualización de las recetas

Introducción

Para visualizar recetas configure el visor de recetas. Los valores de una receta se modifican en el visor de recetas p. ej. para influir en el proceso de fabricación, o bien en una máquina.

Visor de recetas

El visor de recetas es un objeto de manejo y visualización de WinCC que se utiliza para gestionar los registros de recetas. El visor de recetas siempre forma parte de una imagen. El visor de recetas muestra registros de recetas en forma de tabla. La representación y las posibilidades de manejo se adaptan a las necesidades individuales.



Si las recetas del proyecto se editan en un visor de recetas, los valores se guardan en registros de receta. Los valores se transfieren del panel de operador al controlador tan solo cuando se acciona el objeto de manejo correspondiente.

10.4.1.5 Flujo de datos en las recetas

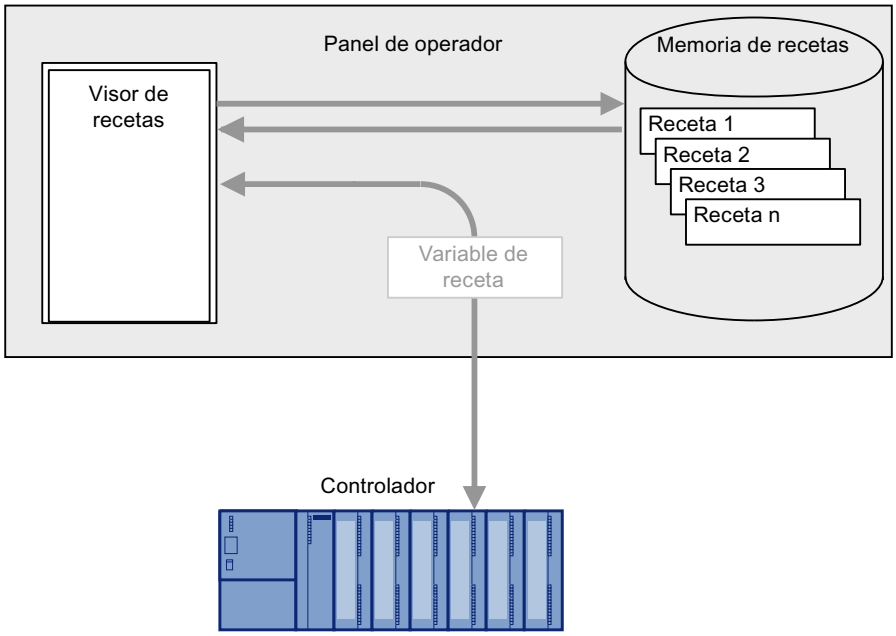
Interacción de los componentes

Los siguientes componentes interactúan en runtime:

- Visor de recetas
  En el panel de operador, las recetas de visualizan y se editan en el visor de recetas.
  Los registros de recetas se visualizan y se editan en el visor de recetas desde la memoria interna del panel de operador.
- Memoria de recetas del panel de operador
  Las recetas se almacenan en forma de registros en la memoria de recetas del panel de operador.
- Variables de receta
  Las variables de receta contienen datos de recetas.

Vista general del flujo de datos

La figura siguiente representa el flujo de datos en las recetas:



Para transferir registros de receta al controlador utilice el botón "hacia el controlador" del visor de recetas o un objeto de manejo con la función de sistema "VisorDeRecetasEscribirRegistroEnControlador".

Los datos se intercambian con el controlador a través de variables de receta. En los Basic Panels, las variables de receta no se pueden utilizar fuera de una receta, p. ej., en campos E/S.

10.4.1.6 Sincronizar registros de receta con el controlador

Resumen

Al transferir registros de receta entre el panel de operador y el controlador, ambos interlocutores acceden alternativamente a las áreas de comunicación conjuntas.

Los registros de receta se transfieren siempre directamente. Los valores de las variables se escriben o se leen directamente en/de las direcciones configuradas, sin necesidad de utilizar el portapapeles.

Tipos de transferencia

Los registros de receta se pueden transferir de dos maneras entre el panel de operador y el controlador, a saber:

- Transferencia sin coordinación
- Transferencia coordinada mediante el puntero de área "Registro".

Nota

Transferencia coordinada

En la transferencia coordinada se impide que los datos se sobrescriban unos a otros de forma incontrolada en el programa de control.

Requisitos para la transferencia coordinada

Los requisitos siguientes rigen para la transferencia coordinada:

- En el editor "Comunicación > Conexiones" debe estar configurado el puntero de área "Registro" para la conexión deseada.
- En las propiedades de la receta debe estar activada la "Transferencia coordinada de los registros".
- En las propiedades de la receta se debe haber indicado la conexión con el controlador con el que el panel de operador coordinará la transferencia.

Transferencia coordinada

Durante la transferencia coordinada, tanto el controlador como el panel de operador activan bits de estado en el buzón de datos común.

La transferencia coordinada de registros de receta resulta útil en los casos siguientes:

- El controlador es el "interlocutor activo" al transferir registros de recetas.
- El controlador evalúa las informaciones mediante el número y el nombre de la receta así como mediante el número y el nombre del registro de la receta.
- Las siguientes tareas de control inician la transferencia de registros de receta:
  - "Escribir_registro_en_controlador"
  - "Leer_registro_de_controlador"

10.4.2 Elementos y configuración básica

10.4.2.1 Editor "Recetas"

Introducción

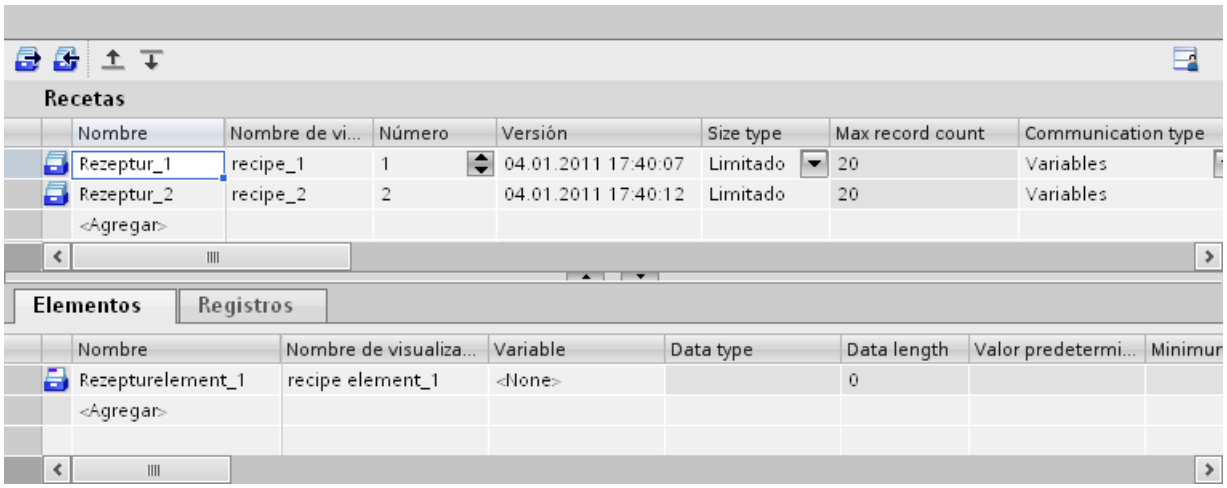
En el editor "Recetas" se crean, configuran y editan las recetas, los elementos de receta y los registros de receta. Además, se pueden introducir valores en los registros de las recetas ya existentes.

Estructura del editor "Recetas"

Las recetas se crean en la parte superior del editor de tablas y se configuran allí mismo o bien en la ventana de inspección.

La parte inferior del editor de tablas contiene las fichas siguientes:

- Elementos
  Aquí se definen los elementos de la receta seleccionada con las filas de la tabla. Dentro de la tabla los elementos de receta se desplazan con los comandos del menú contextual "Arriba" y "Abajo".
- Registros
  Aquí se definen los valores de los registros de la receta seleccionada con las filas de la tabla.



En la ventana de inspección se configura la receta seleccionada en cada caso, el elemento de receta o el registro de receta. Encontrará más información sobre la configuración de los ingredientes de una receta en "Configurar recetas".

Ajustes de recetas

Para las recetas son posibles los ajustes siguientes:

Ajuste	Descripción
Nombre de la receta	Identifica la receta unívocamente en el panel de operador.
Nombre de visualización	Aparece en runtime en el visor de recetas, p. ej. El nombre de visualización se puede configurar en varios idiomas. Ponga nombres descriptivos o designaciones que el operador pueda asignar directamente a una receta, p. ej., "Bebida de zumo de fruta".
Número de receta	Identifica la receta unívocamente en el panel de operador.
Versión	Información sobre la receta. De forma predeterminada se muestran la fecha y la hora de la última modificación de la receta.
Ruta	Define la ubicación de las recetas. Las recetas se guardan como archivo.
Tipo de tamaño [inalterable]	El número máximo disponible de registros de receta está predeterminado de manera estándar.
Número de registros [inalterable]	Número máximo de registros de una receta en runtime. Este número está limitado por la memoria de recetas del panel de operador.
Tipo de comunicación [inalterable]	Los registros de receta se escriben directamente en las direcciones de las variables de receta y se leen desde allí.
Tooltip	Tooltip relativo a la receta que se muestra al operador en runtime.

Nota

Ruta

La ubicación depende de los medios de almacenamiento con que esté equipado el panel de operador.

Basic Panels, OP77A y TP177A (retrato)

Estos paneles de operador no disponen de memoria externa. Las recetas se guardan siempre en la memoria flash interna. De tal modo, el ajuste "Ruta" no está disponible.

Ajustes de los elementos de receta

En la ficha "Elementos" se admiten los ajustes siguientes:

Ajuste	Descripción
Nombre del elemento de receta	Identifica el elemento de receta unívocamente en la receta. Ponga los nombres o las designaciones descriptivas que pueda asignar unívocamente, p. ej. denominaciones de los ejes de una máquina o ingredientes como "Aroma".
Nombre de visualización	Aparece en runtime en el visor de recetas, p. ej. El nombre de visualización se puede configurar en varios idiomas. Ponga nombres descriptivos o designaciones que el operador pueda asignar directamente, p. ej., "Aroma de zumo de fruta".
Variable de receta	Una variable asignada guarda en runtime el valor actual del elemento en el registro de receta.
Tipo de datos	Tipo de datos de la variable de receta.

Ajuste	Descripción
Longitud de los datos [inalterable]	El tipo de datos determina la longitud de los datos de una variable de receta.
Lista de textos	En una lista de textos se asigna texto a un valor o rango de valores. Dicho texto se puede visualizar en un campo de salida, p. ej. La variable de receta asignada debe ser de un tipo de datos numérico. El valor de la variable debe estar dentro del rango de valores de la lista de textos.
Valor estándar	Se utiliza como entrada estándar cuando se crea un registro de receta nuevo.
Valor mínimo [inalterable]	Valor más pequeño representable de una variable de receta basada en números, depende del tipo de datos.
Valor máximo [inalterable]	Valor más grande representable de una variable de receta basada en números, depende del tipo de datos.
Decimales	Determina con cuántos decimales se redondea exactamente un valor decimal (p. ej., 3 decimales) y, a la inversa, por qué potencia de diez se multiplica un valor entero (p. ej., 1000).
Tooltip	Tooltip del elemento de receta que se muestra al operador en runtime.

Ajustes de los registros de receta

En la ficha "Registros" se admiten los ajustes siguientes:

Ajuste	Descripción
Nombre del registro de receta	Identifica el registro de receta unívocamente dentro de una receta.
Nombre de visualización	Aparece en runtime en el visor de recetas, p. ej. El nombre de visualización se puede configurar en varios idiomas. Ponga nombres descriptivos o números de producto que el operador pueda asignar directamente a un producto, p. ej., "Zumo de fruta amarillo E231".
Número de registro de receta	Identifica el registro de receta unívocamente dentro de una receta.
Elementos de receta 1 a n	Incluso durante la configuración ya se pueden guardar diferentes valores para cada elemento de receta. Un registro de receta está formado conjuntamente por un valor y los valores de los demás elementos de receta. Es posible guardar varios registros de receta. Si ha activado la opción respectiva en los ajustes de transferencia, al cargar el proyecto en el panel de operador también se cargarán los registros de receta y en el panel de operador se sobrescribirán los registros existentes.
Comentario	Comentario sobre el registro de receta

10.4.3 Visualización y edición de recetas en runtime

10.4.3.1 Visor de recetas simple

Visor de recetas

El visor de recetas simple es un objeto de manejo y visualización preconfeccionado que se utiliza para gestionar los registros de recetas. El visor de recetas muestra registros de recetas en forma de tabla.

Los botones mostrados y las informaciones de las columnas pueden ajustarse.

Los valores visualizados o introducidos en el visor de recetas se guardan en registros de receta. Los botones permiten escribir el registro de receta visualizado en el controlador y leer valores del controlador.

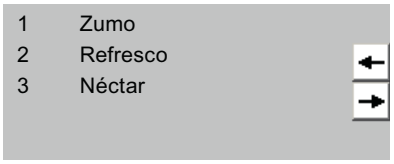
Estructura de la visualización

El visor de recetas simple comprende tres áreas de visualización, a saber:

- Lista de recetas
- Lista de registros
- Lista de elementos

El visor de recetas simple representa cada una de estas áreas por separado en el panel de operador. Dependiendo de la configuración, el visor de recetas simple se inicia con la lista de recetas o la lista de registros.

La figura siguiente muestra un ejemplo de la lista de registros.



Mostrar el valor

Nota

El registro de receta editado se modifica en segundo plano

Solo vale para los Basic Panels: si un operador ha modificado un registro de receta y una petición del controlador solicita leer o escribir un registro cualquiera de esta receta, la petición del controlador se detiene y se emite un aviso de sistema. En cambio, si la modificación de los datos de receta se debe únicamente a la petición del controlador y no a la acción de un operador, el valor modificado se mostrará de inmediato.

No vale para Basic Panels: si un operador ha modificado un registro de receta y una petición del controlador modifica valores del registro de receta en cuestión, el visor de recetas no se actualizará automáticamente. Para actualizar el visor de recetas debe volver a seleccionar el registro de recetas correspondiente.

Consulte también

Visor de recetas (Página 8446)

10.4.3.2 Comportamiento del visor de recetas en runtime

Cambio de imagen

Si cambia a una imagen diferente sin haber guardado todavía los datos modificados en el visor de recetas, se le solicitará que almacene los datos de la receta. Para poder determinar qué datos de receta no se han guardado todavía, se visualizarán el nombre de la receta y del registro de receta.

Crear, modificar, copiar o borrar registros de recetas

Si desea crear de nuevo un registro de receta y ya existe uno se visualizará un aviso de sistema en la pantalla.

Manejo del visor de recetas con teclas de función

El visor de recetas se puede manejar mediante teclas de función, p. ej. si el panel de operador no dispone de pantalla táctil. A las teclas de función del panel de operador se les pueden asociar funciones, p. ej. "GuardarRegistro".

Vista tras importar datos de receta

Nota

Disponibilidad

La importación y exportación de datos de receta no está disponible para Basic Panels, OP77A y TP177A (retrato).

Si abre el visor de recetas mientras importa datos de receta, se mostrarán sólo los datos de receta que se acaben de importar por completo. El visor de recetas no se actualiza automáticamente durante la importación de datos. Para visualizar de forma completa de todos los datos de receta, no abra el visor de recetas hasta que no aparezca el aviso de sistema indicando que se han importado correctamente los datos de receta. Como alternativa el visor de recetas se actualiza tras finalizar correctamente la importación.

Actualizar la variable para recetas y registros de recetas

Nota

Disponibilidad

Las variables para recetas y los registros de receta no están disponibles para Basic Panels, OP77A y TP177A (Portrait).

Dependiendo de la configuración, el registro de receta actual y/o su número se pueden almacenar en una variable. La variable se actualiza cuando se cumplen las siguientes condiciones:

- Ha finalizado la carga del registro de receta.
- No se ha salido de la imagen que contiene el visor de recetas durante la carga.

Este proceso puede tardar varios minutos.

10.4.4 Configurar recetas

10.4.4.1 Procedimiento general de configuración

Si desea crear una nueva receta, siga los pasos de configuración siguientes:

Paso	Descripción
1	Defina la estructura de la receta.
2	Cree las variables conforme a la estructura de la receta. Asigne nombres tecnológicos a las variables.
3	Cree la receta.

Paso	Descripción
4	Introduzca las propiedades deseadas para la receta. • Nombre de visualización de la receta dependiente del idioma • Opción "Transferencia coordinada de los registros" No válido para Basic Panels: • Ubicación de las recetas • Opción "Sincronizar el visor de recetas y las variables de receta" • Opción "Transferencia manual de valores individuales modificados (modo "Teach-In")".
5	Cree los elementos de receta e indique las propiedades deseadas: • Nombres de visualización de los elementos de receta dependientes del idioma • Enlace de variables de los elementos de receta • Valores estándar y decimales (potencia de diez) de los elementos de receta
6	Cree los registros de receta. Introduzca los nombres de visualización dependientes del idioma para los registros de receta.
7	Configure una imagen con un visor de recetas o una imagen de receta.

Nota

Basic Panels, OP77A y TP177A (retrato)

La selección de la ubicación no está disponible para estos equipos. Las recetas se guardan siempre en la memoria flash interna.

Las variables de receta no se pueden utilizar fuera de una receta, p. ej., no en campos E/S, no como variables de disparo en avisos, no como parámetros en funciones de sistema, etc.

Nota

Limitaciones en el visor de recetas y la imagen de receta

En Basic Panels, OP77A y TP177A solo está disponible el visor de recetas simple. En Basic Panels, OP73, OP77A y TP177A (retrato), las imágenes de receta no están disponibles.

10.4.4.2 Crear y editar recetas

Crear una receta nueva

Introducción

Para crear una receta completa primero debe crearse una receta nueva, asignar elementos a la receta y definir los valores correspondientes en un registro de receta.

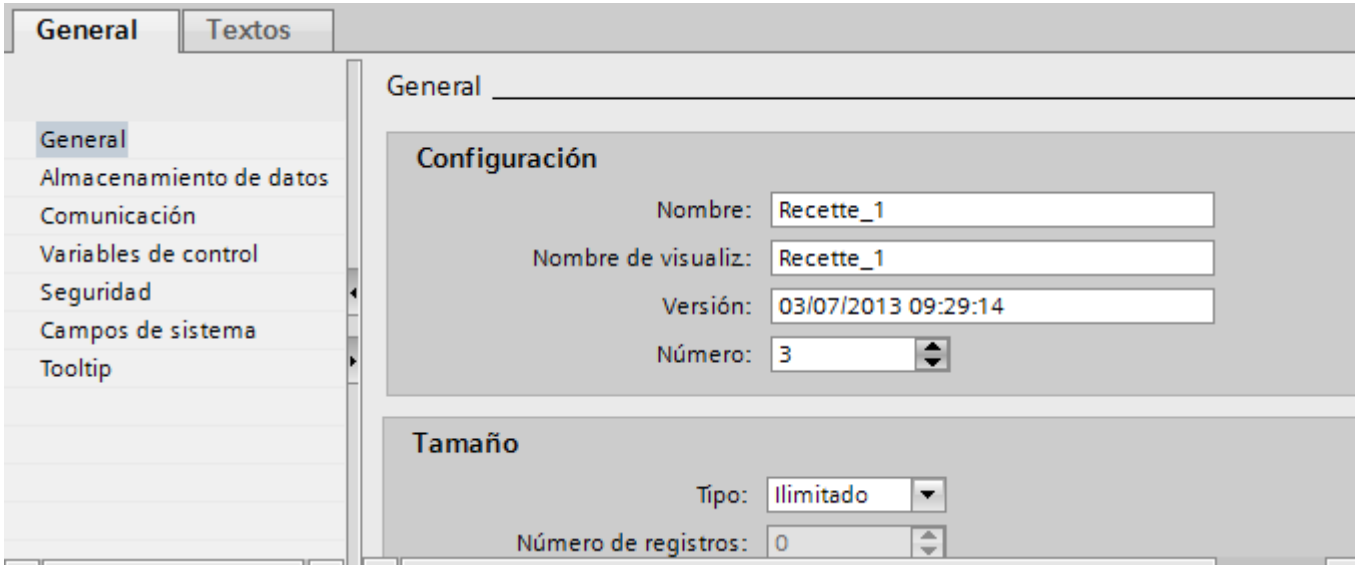
Requisitos

- Las variables de la receta deben existir.
- El editor "Recetas" está abierto.

Crear una receta

Para crear una receta, proceda del siguiente modo:

1. En el editor "Recetas", haga clic en "Añadir" en la primera fila libre de la tabla.
   La receta nueva se crea y se muestra en una fila.



2. En el área "General", introduzca un nombre que describa la receta en "Nombre".
   El nombre identifica de forma unívoca la receta dentro de un panel de operador.
3. En "Nombre de visualización" introduzca el nombre dependiente del idioma que deba aparecer en el visor de recetas.
4. Seleccione un número de receta bajo "Número".
   El número identifica de forma unívoca la receta dentro del panel de operador.
   La receta recibe automáticamente una versión que muestra la fecha y hora de la última modificación. Aquí también se puede introducir información relacionada con la receta.
5. Bajo "Archivo de datos", determine la ubicación de los registros de receta. Las posibilidades ofrecidas dependen del panel de operador utilizado.

   Nota

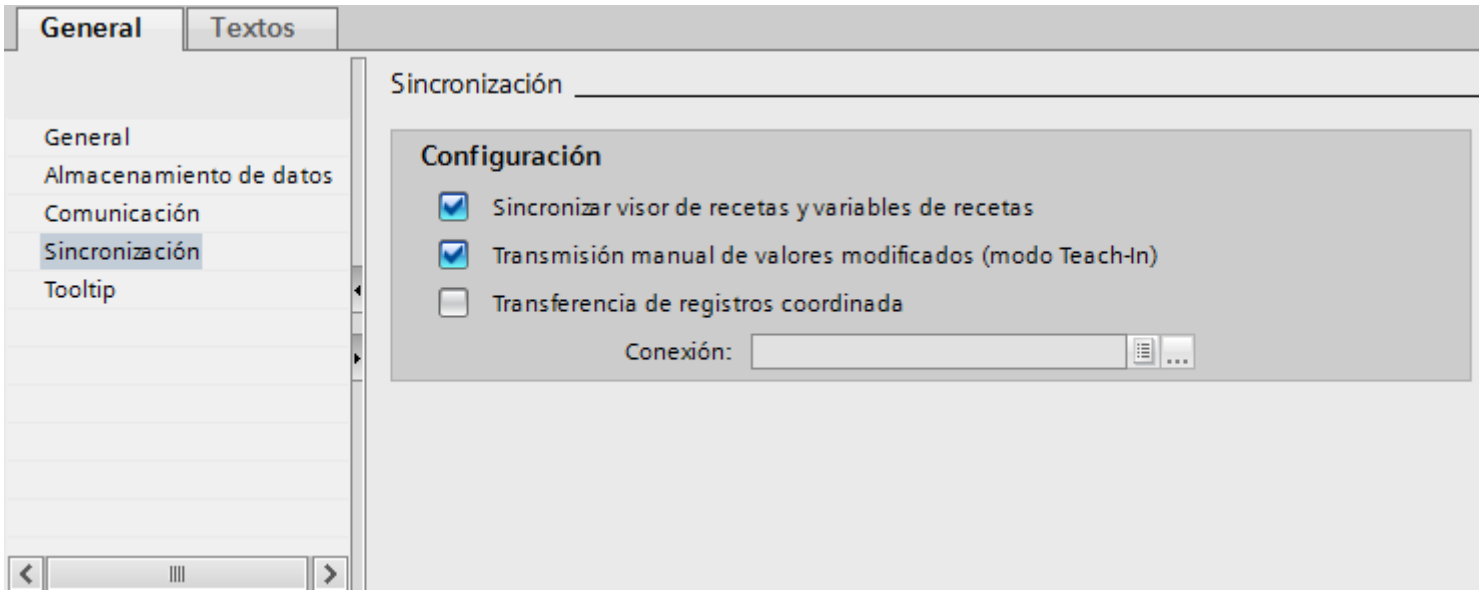
   Basic Panels, OP77A y TP177A (retrato)

   La selección de la ubicación no está disponible para estos equipos. Las recetas se guardan siempre en la memoria flash interna.

   Las variables de receta no se pueden utilizar fuera de una receta, p. ej., no en campos E/S, no como variables de disparo en avisos, no como parámetros en funciones de sistema, etc.

6. Introduzca un tooltip que se mostrará al operador en runtime.

7. Para sincronizar en runtime las variables de receta configuradas en campos E/S con el visor de recetas, active "Sincronizar el visor de recetas y las variables de receta" en "Propiedades > Sincronización" de la ventana de inspección.



Nota

Basic Panels, OP77A y TP177A (retrato)

Puesto que en los Basic Panels las variables de receta no pueden utilizarse adicionalmente en los campos E/S de imágenes, la opción "Sincronizar el visor de recetas y las variables de receta" no está disponible, y en lo sucesivo tampoco la opción "Transferencia manual de valores individuales modificados (modo "Teach-In")".

8. Para determinar que las variables de receta se transfieran automáticamente al controlador cuando se editen en los campos E/S, desactive "Transferencia manual de valores individuales modificados (modo "Teach-In")".
9. Para poder supervisar en runtime la transferencia de datos de receta mediante punteros de área, active "Transferencia coordinada de los registros".
10. Para la transferencia coordinada seleccione la conexión correspondiente con el controlador en "Sincronizar con".

Crear un elemento de receta

Para crear elementos de receta, proceda del siguiente modo:

1. Haga clic en la ficha "Elementos".
2. En la primera fila libre del editor de tablas haga clic en "Añadir".
   Se crea un elemento de receta nuevo.
3. En "Nombre" introduzca un nombre que describa el elemento.
   El nombre identifica de forma unívoca el elemento en la receta.

4. En "Nombre de visualización" introduzca un nombre dependiente del idioma para el elemento.
   El nombre de visualización aparece en runtime en el visor de recetas, p. ej.
5. En "Variable", seleccione la variable que desea asociar al elemento de receta.
   En esta variable se guarda en runtime el valor del elemento de receta que está guardado en un registro de receta.

Elementos | Registros

	Nombre	Nombre de visualización	Variable	Valor predeterminado	Decimales	Texto de ayuda
	Agua	Agua	HMI_3	0	0	
	<Agregar>					

6. Introduzca un tooltip.
   El tooltip se muestra al operador en runtime.
7. En "Valor estándar" introduzca el valor que desea utilizar como entrada estándar al crear un registro de receta nuevo.
8. Si desea asignar texto a un valor o a un rango de valores, seleccione aquí la lista de textos correspondiente. La variable de receta asignada debe ser de un tipo de datos numérico. El valor de la variable debe estar dentro del rango de valores de la lista de textos.
   El texto almacenado en la lista de textos se mostrará en runtime en un campo de salida, por ejemplo.
9. Especifique en la columna "Decimales" con cuántos decimales se redondea exactamente un valor decimal (p. ej., 3 decimales) y, a la inversa, por qué potencia de diez se multiplica un valor entero (p. ej., 1000).
   Ejemplo con 3 decimales: la entrada "5" en un elemento de receta con el tipo de datos "Integer" devuelve el valor "5000". La entrada "5,6789" en un elemento de receta con el tipo de datos "Real" devuelve el valor "5,679".
10. Cree todos los elementos que necesite para la receta. La cantidad máxima admisible de elementos de receta depende del panel de operador utilizado.

Elementos | Registros

	Nombre	Nombre de visualización	Variable	Valor predeterminado	Decimales	Texto de ayuda
	Agua	Agua	HMI_10	0	0	
	Concentrado	Concentrado	HMI_11	0	0	
	Azúcar	Azúcar	HMI_12	0	0	
	Aroma	Aroma	HMI_13	0	0	
	<Agregar>					

Crear un registro de receta con valores de receta conocidos

Para crear elementos de receta, proceda del siguiente modo:

1. Haga clic en la ficha "Registros".
2. En la primera fila libre del editor de tablas haga clic en "Añadir".
   Se crea un registro de receta nuevo. El registro de recetas contiene una columna propia para cada elemento creado en la receta.

Elementos | Registros

Nombre	Nombre de visualizac..	Número	Agua	Concentrado	Azúcar	Aroma	Comentario
Registro de receta_1	Registro de receta_1	1	0	0	0	0	
<Agregar>							

3. En "Nombre" introduzca un nombre que describa el registro de receta.
   El nombre identifica de forma unívoca el registro en la receta.
4. En "Nombre de visualización" introduzca un nombre dependiente del idioma.
   El nombre de visualización aparece en runtime en el visor de recetas, p. ej.
5. En "Número" introduzca un número para el registro de receta.
   El número del registro de receta identifica de forma unívoca el registro en la receta.
6. Si ya conoce los valores de la receta durante la configuración, introduzca el valor correspondiente a cada elemento de receta.

Elementos | Registros

Nombre	Nombre de visualizac..	Número	Agua	Concentrado	Azúcar	Aroma	Comentario
Refresco	Refresco	1	30	70	45	600	
<Agregar>							

7. Cree todos los registros que necesite para la receta.

Elementos | Registros

Nombre	Nombre de visualizac..	Número	Agua	Concentrado	Azúcar	Aroma	Comentario
Refresco	Refresco	1	30	70	45	600	
Néctar	Néctar	2	50	50	10	300	
Zumo	Zumo	3	5	95	3	100	
<Agregar>							

Entrada de valores en runtime

Existen las posibilidades siguientes para introducir valores en los registros de receta en runtime:

- Transferir datos directamente del controlador (modo "Teach-In")
- Importar valores de un archivo CSV
- Introducir valores en el panel de operador.

Nota

Basic Panels, OP77A y TP177A (retrato)

La importación de valores no está disponible para estos equipos.

Resultado

La receta completa está configurada.

Registros de receta con fecha y hora

Si utiliza indicaciones de fecha y hora, asegúrese de que la fecha y hora del sistema en el equipo de configuración coincida con la del sistema de destino. Ejemplo: a las 13:55 se carga un registro de receta en el sistema de destino en el que está guardado el tiempo de ejecución 14:00. Si en el equipo de destino ya son las 14:05, la receta ya no se procesa. Si un operador edita la receta, los datos de modificación no se reescribirán correctamente en la base de datos.

Después de cargar en el sistema de destino, es necesario comprobar las recetas con indicación de fecha y hora en el sistema de destino.

Editar recetas

Finalidad

Desea modificar o ampliar una receta, o bien borrar partes de la misma.

Requisitos

- Se ha creado una receta como mínimo.
- El editor "Recetas" está abierto.

Modificar la configuración de una receta

Para modificar la configuración de una receta, proceda del siguiente modo:

1. En el editor "Recetas" seleccione la receta que desea modificar.
   Se abre la ventana de inspección.
2. Modifique la configuración de la receta en la ventana de inspección.

Los elementos y registros de receta se modifican del mismo modo.

Modificar los valores de una receta

Para modificar los valores de una receta, proceda del siguiente modo:

1. Seleccione la receta cuyos valores desea modificar.
2. Haga clic en la ficha "Registros".
3. Introduzca los valores nuevos en las columnas correspondientes.

Añadir un elemento de receta

Para añadir elementos adicionales a una receta, proceda del siguiente modo:

1. En el editor "Recetas" seleccione la receta a la que desea añadir elementos.
2. Haga clic en la ficha "Elementos".
3. En la primera fila libre haga clic en "Añadir".
   Se crea el elemento de receta.
4. Configure el elemento de receta.

Los registros de receta se añaden del mismo modo.

Administrar recetas

Requisitos

- Se ha creado una receta con elementos y registro de receta.
- El editor "Recetas" está abierto.

Cambiar el nombre de las recetas

En las recetas, en los elementos de receta y en los registros de receta se distingue entre el nombre interno y el nombre de visualización.

Para cambiar el nombre de las recetas, proceda del siguiente modo:

1. Seleccione la receta cuyo nombre desea modificar.
   Se abre la ventana de inspección.
2. En el menú contextual, seleccione el comando "Cambiar nombre".
3. Introduzca el nuevo nombre.
   Los nombres de los elementos y registros de receta se cambian del mismo modo en la ficha correspondiente.

Nota

Los nombres de visualización que aparecen en el editor "Recetas" también se pueden cambiar en "Idiomas y recursos > Textos del proyecto". Ello resulta útil si ha realizado la configuración en varios idiomas, por ejemplo.

Copiar y pegar recetas

Para copiar y pegar recetas, proceda del siguiente modo:

1. Seleccione la receta que desee copiar.
2. En el menú contextual, elija el comando "Copiar".
3. En la primera fila libre de la tabla elija el comando "Pegar" del menú contextual.

La receta copiada se insertará en la tabla. Junto con la receta también se copian los elementos y registros de receta en la ficha correspondiente.

Los elementos y registros de receta se copian además del mismo modo en la ficha correspondiente.

Si ya existe un registro de receta con el mismo nombre, se le añadirá un número al nombre del registro de receta copiado. Así se garantiza que el nombre sea unívoco. Los registros de receta sólo se pueden copiar e insertar en una misma receta.

Borrar una receta

Para borrar una receta, proceda del siguiente modo:

1. Seleccione la receta que desea borrar.
2. En el menú contextual, elija el comando "Borrar".
   La receta se borrará.

Los elementos y registros de receta se borran del mismo modo en la ficha correspondiente.

Nota

Al borrar una receta se borran los registros contenidos en ella.

Nota

Al borrar un elemento de receta se borrarán también los valores correspondientes en los registros de la receta. Las variables asociadas se conservan.

10.4.4.3 Configurar la visualización de la receta

Configurar el visor de recetas simple

Requisitos

- La receta se ha creado.
- El editor "Imágenes" está abierto.
- La imagen está creada y abierta.

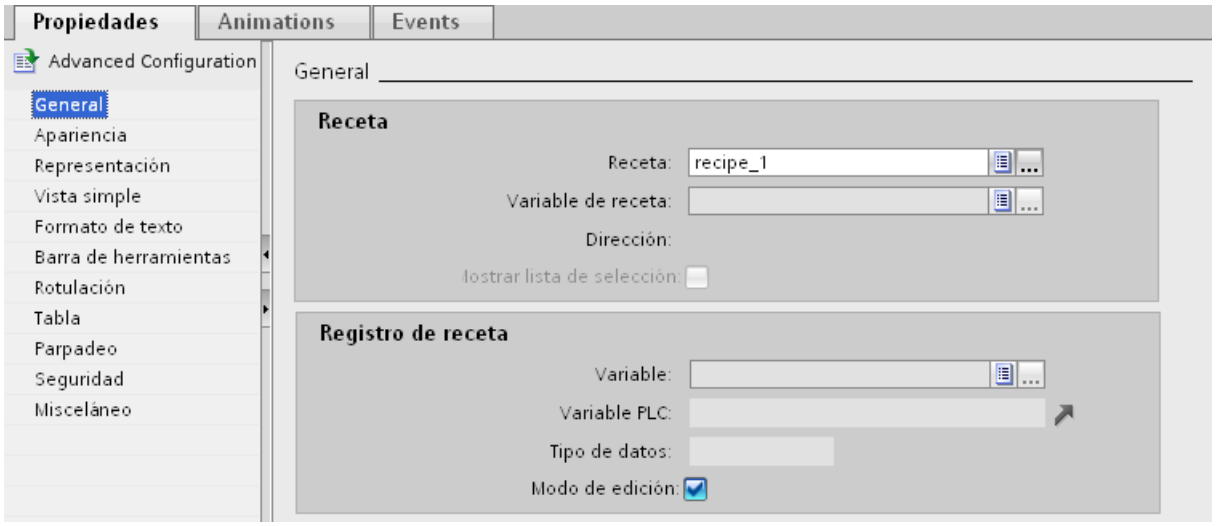
ATENCIÓN
Pérdida de datos en caso de varios visores de recetas en la imagen Sólo rige para Basic Panels, OP73, OP77A, TP177A y TP177A (Portrait): Si dos o más visores de recetas muestran la misma receta en una imagen, al acceder a los datos se crea un conflicto. Esto genera un estado impredecible de los datos de receta o bien la pérdida de los mismos. Asegúrese de que el operador no pueda seleccionar ni modificar la misma receta en diferentes visores de recetas. • Muestre en cada visor de recetas sólo una receta, respectivamente. • Muestre en cada visor de recetas una receta distinta, respectivamente.

Procedimiento

Para configurar un visor de recetas simple, proceda del siguiente modo:

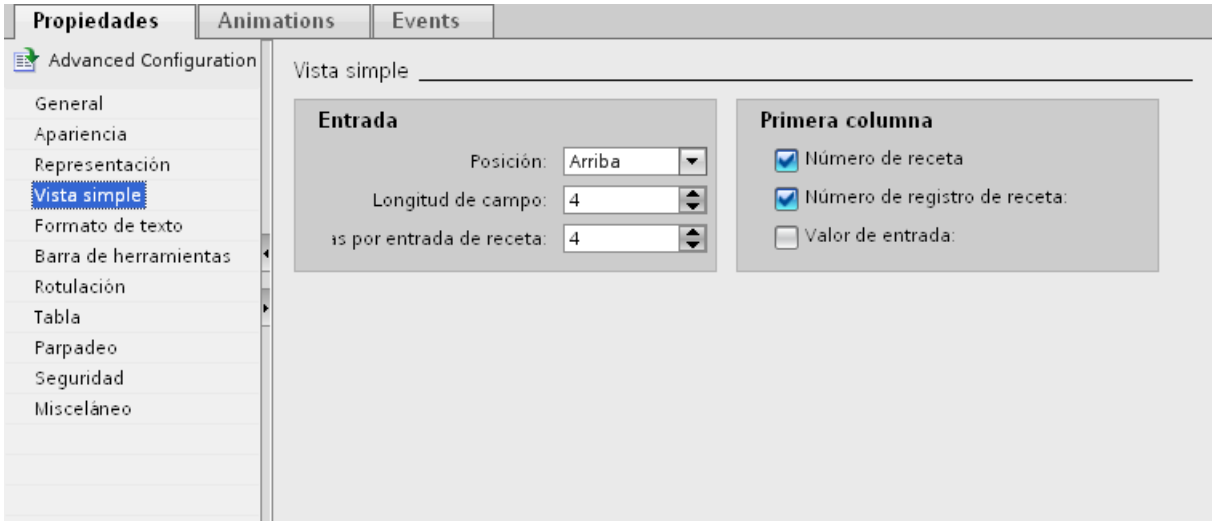
1. Inserte el visor de recetas en la imagen. El visor de recetas se encuentra en "Controles" de la Task Card "Herramientas".
2. Solo con equipos que también soporten el visor de recetas avanzado: Active "Vista simple" en "Propiedades > Representación > Modo".

3. Si desea visualizar sólo los registros de una determinada receta en el visor de recetas, seleccione la receta deseada bajo "Propiedades > General > Receta".

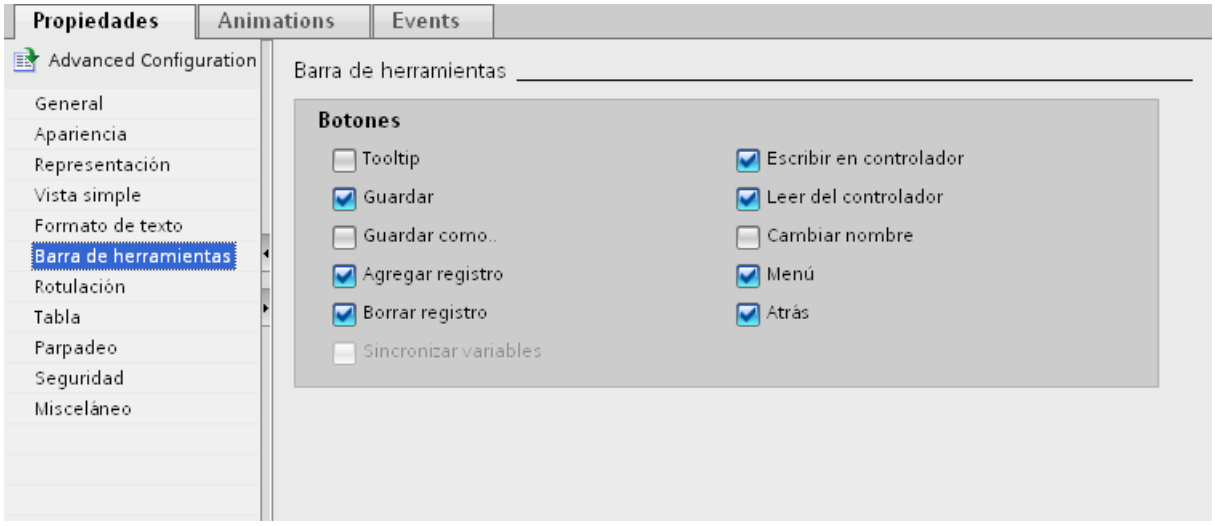


4. Si en el visor de recetas desea ver únicamente los datos de receta, desactive "Modo de edición" en el área "Registro de receta".
5. En "Propiedades > Apariencia" y "Propiedades > Representación" defina opciones de visualización adicionales del visor de recetas.

6. En "Propiedades > Vista simple" seleccione la posición, la longitud del campo y el número de filas deseado.
   En "Posición > Arriba" aparece el valor de la receta en la primera fila de la entrada.
   En "Posición > Abajo" aparece el valor de la receta en la última fila de la entrada.



7. En "Propiedades > Barra de herramientas", determine qué comandos de menú deben estar disponibles en runtime en el visor de recetas.



Resultado

El visor de recetas simple quedará configurado. En runtime podrá utilizar el visor de recetas para visualizar o editar los datos de las recetas.

Desactivar el modo de edición en "Propiedades > Propiedades > General" no tiene efecto sobre los botones de la barra de herramientas. Los botones activados en "Propiedades > Barra de herramientas" también se pueden manejar con el modo de edición desactivado.

10.4.5 Manejar recetas en runtime

10.4.5.1 Manejar el visor de recetas simple

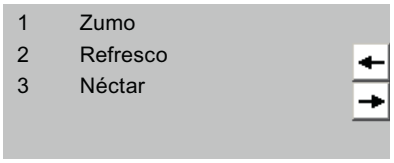
Descripción del visor de recetas simple

Representación

El visor de recetas simple comprende las áreas de representación siguientes:

- Lista de recetas
- Lista de registros
- Lista de elementos

La figura siguiente muestra una lista de registros:



El visor de recetas simple representa cada una de estas áreas por separado en el panel de operador. Todas estas áreas de visualización se manejan mediante un menú contextual.

Por defecto se visualiza primero el visor de recetas simple con la lista de recetas.

Manejo

Para manejar el visor de recetas simple existen las posibilidades siguientes, dependiendo de la configuración:

- Crear, modificar, copiar o borrar registros de recetas
- Leer un registro de recetas del controlador o transferirlo al controlador

Manejar el área de visualización y el menú contextual

Para manejar el visor de recetas simple, conmute entre las áreas de visualización y los menús contextuales.

La tabla siguiente muestra las posibilidades de manejo del área de visualización.

Botón	Tecla	Función
	<Intro>	Abre la siguiente área de visualización subordinada, es decir, la lista de registros o la lista de elementos.
←	<Esc>	Abre el área de visualización anterior.

Botón	Tecla	Función
	<INS>	Crea un nuevo registro de la receta seleccionada cuando se muestra la lista de recetas o de registros de receta. A continuación, pasa a la lista de elementos de receta. Requisitos: "Propiedades > General > Modo de edición" está activado. El botón puede simularse con la función "Tecla SimulateSystemKey" (también en equipos sin teclas).
	<Supr>	Borra el registro de receta seleccionado en la lista de registros de receta. Requisitos: "Propiedades > General > Modo de edición" está activado.
	<Arriba>/<Abajo>	Selecciona la entrada anterior/siguiente.
	<Re Pág>/<Av Pág>	Avanza una página de visualización hacia arriba o hacia abajo.
	<Inicio>/<Fin>	Selecciona la primera/última entrada. La primera/última entrada aparece destacada.

La tabla siguiente muestra las posibilidades de manejo del menú contextual.

Botón	Tecla	Función
→	<Derecha>	Abre el menú contextual del área de visualización.
←	<Esc>	Cierra el menú. Abre el área de visualización.
	Introducir el número del comando de menú	Ejecuta el comando de menú.

Menús contextuales de el visor de recetas simple

En cada área de visualización puede llamarse una selección de comandos con el botón →. Dicha selección contiene los comandos disponibles para el área de visualización en cuestión. Cada comando tiene un número asignado. El comando se ejecuta cuando se introduce el número correspondiente. También se puede seleccionar el comando y pulsar la tecla <Enter>.

Menús contextuales de la lista de recetas

Comando	Función
Nuevo	Permite crear un nuevo registro para la receta seleccionada. Si se ha configurado un valor inicial, éste se visualizará en el campo de entrada.
Mostrar tooltip	Se muestra el tooltip configurado para la receta.
Abrir	Abre la lista de registros de la receta seleccionada.

Menús contextuales de la lista de registros de receta

Comando	Función
Nuevo	Permite crear un nuevo registro de receta. Si se ha configurado un valor inicial, éste se visualizará en el campo de entrada.
Borrar	Borra el registro seleccionado.
Guardar como	Guarda el registro seleccionado con un nombre diferente. El nombre se introduce en un cuadro de diálogo.
Cambiar nombre	Permite cambiar el nombre del registro seleccionado. El nombre se introduce en un cuadro de diálogo.
Abrir	Abre la lista de elementos del registro seleccionado.
Atrás	Abre la lista de recetas.

Menús contextuales de la lista de elementos de receta

Comando	Función
Guardar	Se guarda el registro marcado con el elemento de receta.
Al controlador	Los valores visualizados del registro de seleccionado se transfieren del panel de operador al controlador.
Del controlador	Los valores de receta del controlador se visualizan en el visor de recetas del panel de operador.
Guardar como	Guarda el registro seleccionado con un nombre diferente. El nombre se introduce en un cuadro de diálogo.
Mostrar tooltip	Se muestra el tooltip configurado para el elemento de receta.
Cambiar nombre	Se cambia el nombre del elemento de receta marcado. El nombre se introduce en un cuadro de diálogo.
Atrás	Se abre la lista de registros.

Menús contextuales de la lista de registros

Nota

Dependencia del panel de operador

Tanto en Basic Panels como en OP 77A, TP 177A, TP 177A (retrato) y TB 177B están configurados adicionalmente los siguientes comandos de menú.

Comando	Función
Al controlador	Los valores visualizados del registro de seleccionado se transfieren del panel de operador al controlador.
Del controlador	Los valores de receta del controlador se visualizan en el visor de recetas del panel de operador.

Administrar registros de receta

Administrar registros de recetas

Para administrar el visor de recetas simple existen las posibilidades siguientes, dependiendo de la configuración:

- Crear nuevos registros de recetas
- Copiar registros de recetas
- Modificar registros de recetas
- Borrar registros de recetas

Crear un nuevo registro de receta

Para crear un registro de receta nuevo, proceda del siguiente modo:

1. Seleccione en el panel de operador la receta en la que desee crear un nuevo registro de receta.
2. Seleccione el comando "Nuevo" del menú contextual de la lista de recetas.
   Se creará un nuevo registro de datos con el siguiente número libre.
   Se abrirá la lista de elementos del nuevo registro de receta.
3. Introduzca los valores para los elementos del registro de receta.
   Dependiendo de la configuración, los valores del registro de la receta ya tienen asignados valores predeterminados.
4. Seleccione el comando "Guardar" del menú contextual de la lista de elementos.
   Se abrirá el cuadro de diálogo "Guardar como".
5. Introduzca el nombre y el número del registro de receta.
6. Haga clic en el botón "Aceptar".

Resultado

El nuevo registro de receta será guardado en la receta seleccionada. Si el registro de receta ya existe se emitirá en la pantalla un aviso del sistema.

Copiar un registro de receta

Para copiar un registro de receta, proceda del siguiente modo:

1. Seleccione en el panel de operador la receta en la que desea copiar un registro de receta existente.
2. En el panel de operador, seleccione el registro de receta del que desea guardar una copia.
3. Seleccione el comando "Guardar como" del menú contextual de la lista de registros.
   Se abrirá el cuadro de diálogo "Guardar como". El registro de receta obtendrá automáticamente el siguiente número libre.
4. En caso necesario, introduzca el nombre y el número del registro de receta.
5. Haga clic en el botón "Aceptar".

Resultado

El registro del a receta se creará con el nuevo nombre.

Modificar un registro de receta

Para modificar un registro de receta, proceda del siguiente modo:

1. Seleccione en el panel de operador la receta en la que desea modificar un registro de receta existente.
2. Seleccione en el panel de operador el registro de receta que desea modificar.
3. Elija el registro de receta deseado.
   Se visualizará la lista de elementos del nuevo registro de receta.
4. Sustituya los valores existentes por los nuevos valores.
5. Seleccione el comando "Guardar" del menú contextual de la lista de elementos.

Resultado

Los valores modificados se adoptan en el registro de la receta.

Borrar un registro de receta

Para borrar un registro de receta, proceda del siguiente modo:

1. Seleccione en el panel de operador la receta en la que desea borrar un registro de receta existente.
2. Seleccione en el panel de operador el registro de receta que desea borrar.
3. Seleccione el comando "Borrar" del menú contextual de la lista de registros.
4. Confirme la consulta de seguridad para borrar el registro.

Resultado

El registro de receta se ha borrado.

Leer registro de receta del controlador

Introducción

En runtime se pueden modificar directamente en la instalación los valores que también están almacenados en las recetas del panel de operador. Esto sucede, p. ej. cuando una válvula se ha abierto en la instalación más de lo que está almacenado en la receta. En este caso, es posible que los valores de los registros de receta guardados en el panel de operador ya no concuerden con los valores del controlador.

Los valores de las variables de receta se pueden leer del controlador y escribir en un registro de receta.

Los valores leídos se escriben en el registro de receta que se visualiza en ese momento en el panel de operador.

Procedimiento

Para leer un registro de receta del controlador, proceda del siguiente modo:

1. Abra la receta en el panel de operador.
   Se abre la lista de registros.
2. Abra la lista de elementos del registro de receta en el que desea aplicar los valores del controlador.
3. Seleccione el comando "desde el controlador" del menú contextual de la lista de elementos.
   Los valores se leerán del controlador y se visualizarán en el registro de receta actual.
4. Si desea guardar los valores, elija el comando "Guardar" o "Guardar como".

Resultado

Los valores se leen del controlador, se representarán en el panel de operador y se almacenan en el registro de la receta.

Transferir un registro de receta al controlador

Introducción

Para que los valores de un registro modificado en el visor de recetas sean válidos en el controlador, es preciso transferirlos a éste último.

Los valores visualizados en el visor de recetas son los que se transfieren al controlador.

Procedimiento

Para transferir un registro de receta al controlador, proceda del siguiente modo:

1. Abra la receta deseada.
   Se abre la lista de registros.
2. Abra la lista de elementos del registro de receta cuyos valores desea transferir al controlador.
3. Seleccione el comando "hacia el controlador" del menú contextual de la lista de elementos.

Resultado

Los valores del registro de receta se transfieren al controlador.

10.4.6 Ejemplo

10.4.6.1 Ejemplo de creación de una receta

Tarea

En este ejemplo se crean tres recetas para una planta de producción de zumos de frutas. En la planta de producción de zumos se fabrican los sabores "naranja", "manzana" y "tropical". Para cada uno de los sabores se crea una receta correspondiente.

Cada receta contiene un registro de receta para cada una de las siguientes proporciones de mezcla:

- Refresco
- Néctar
- Zumo

Configuración

Los ajustes se refieren a un panel de operador acoplado a un controlador SIMATIC S7-300 o SIMATIC S7-400.

Para el ejemplo necesitará las siguientes variables, recetas, elementos de receta y registros de receta:

Variables:

Nombre	Conexión al controlador	Dirección	Tipo
LitreWater	Sí	DB 120, DBW 0	Integer
LitreConcentrate	Sí	DB 120, DBW 4	Integer
KiloSugar	Sí	DB 120, DBW 8	Integer
GramAroma	Sí	DB 120, DBW 12	Integer

Recetas:

- Naranja
- Manzana
- Tropical

Elementos de receta:

Elemento de receta	Variable asociada
Litro de agua	LitreWater
Litro de concentrado	LitreConcentrate
Kilo de azúcar	KiloSugar
Gramo de aroma	GramAroma

Registros de receta "Refresco", "Néctar" y "Zumo":

Nombre de registro	Litro de agua	Litro de concentrado	Kilo de azúcar	Gramo de aroma
Refresco	30	70	45	600
Néctar	50	50	10	300
Zumo	5	95	3	100

Procedimiento

Para crear la receta, proceda de la manera siguiente:

1. Cree las variables "LitreWater", "LitreConcentrate", "KiloSugar" y "GramAroma" con los ajustes indicados más arriba.
2. Cree las recetas "Naranja", "Manzana" y "Tropical" con los ajustes indicados más arriba. Cree los elementos de receta en cada una de las recetas.

Elementos | Registros

Nombre	Nombre de visualización	Variable	Valor predeterminado	Decimales	Texto de ayuda
Agua	Agua	HMI_10	0	0	
Concentrado	Concentrado	HMI_11	0	0	
Azúcar	Azúcar	HMI_12	0	0	
Aroma	Aroma	HMI_13	0	0	
<Agregar>					

3. No en los Basic Panels: Configure cada receta de modo que los registros de receta se sincronicen entre la imagen de receta y el visor de recetas. Los valores de las variables de receta no se deben transferir automáticamente al controlador.
   Para ello es preciso realizar los siguientes ajustes en la ventana de propiedades de la receta en cuestión:
   En "Propiedades > Herramientas":
   – Active "Sincronizar el visor de recetas y las variables de receta".
   – Active "Transferencia manual de valores individuales modificados (modo "Teach-In")".
4. Cree en cada receta los registros de datos mencionados. Introduzca en cada registro los valores indicados.

Elementos | Registros

Nombre	Nombre de visualizac..	Número	Agua	Concentrado	Azúcar	Aroma	Comentario
Refresco	Refresco	1	30	70	45	600	
Néctar	Néctar	2	50	50	10	300	
Zumo	Zumo	3	5	95	3	100	
<Agregar>							

Resultado

Se crearán las tres recetas "Naranja", "Manzana" y "Tropical".

10.5 Configurar una administración de usuarios

10.5.1 Campo de aplicación de la administración de usuarios

Principio

La protección de acceso regula el acceso a los datos y las funciones en runtime. De este modo se protegen las aplicaciones de operaciones no autorizadas. Al crear el proyecto, algunas operaciones relevantes para la seguridad ya se restringen a determinados grupos de usuarios. Para ello, se crean usuarios y grupos de usuarios a los que se adjudican derechos de acceso característicos, las autorizaciones. Seguidamente, en los objetos relevantes para la seguridad se configuran las autorizaciones necesarias para el manejo. Así, p. ej., los operadores sólo tienen acceso a determinados objetos de manejo. Los operadores de puesta en marcha disponen de acceso ilimitado en runtime, p. ej.

Definición

Los usuarios, grupos de usuarios y autorizaciones se gestionan de forma centralizada en la administración de usuarios de WinCC. Los usuarios y grupos de usuarios se transfieren al panel de operador junto con el proyecto. Los usuarios y las contraseñas se gestionan en el panel de operador con el visor de usuarios.

Ejemplo de aplicación

La autorización "Servicio" se configura para que sólo los técnicos del servicio de mantenimiento y reparación puedan acceder a los parámetros de ajuste. La autorización se asigna al grupo de usuarios "Técnico de servicio". Todos los miembros de este grupo utilizan así los parámetros de ajuste protegidos.

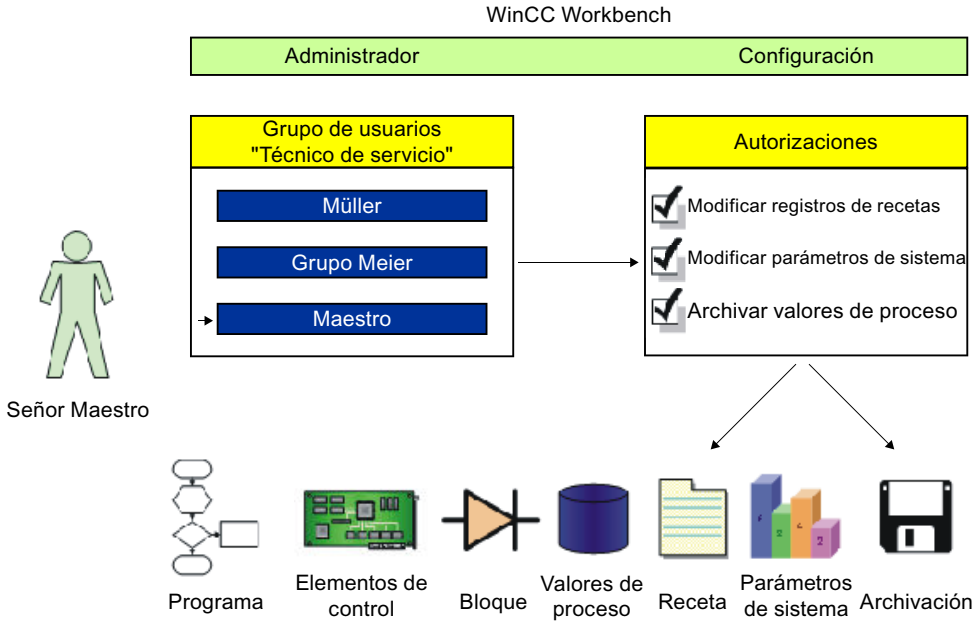
ATENCIÓN
La protección de acceso no protege de manipulaciones erróneas. Es su obligación asegurarse de que sólo el personal autorizado y con la formación adecuada se encarga de construir máquinas e instalaciones, ponerlas en marcha, manejarlas y mantenerlas. La protección de acceso no sirve para definir procesos de trabajo y supervisar su cumplimiento.

10.5.2 Estructura de la administración de usuarios

Introducción

En un proyecto de producción, el entorno del fabricante de máquinas debe ser diferente del entorno del cliente final en su función de operador de la planta.

El fabricante de máquinas autoriza al usuario, p. ej. Sr. Maestro, a acceder a la aplicación o al panel de operador de un modo determinado. Sin embargo, en el cliente final no hay ningún usuario Maestro. Los usuarios del cliente final y sus tareas todavía se desconocen durante la configuración por parte del fabricante de máquinas. Por norma general, los usuarios finales no se determinan hasta después de la puesta en marcha en las instalaciones del cliente final.



Principio

Para mantener el trabajo de administración en niveles reducidos, las autorizaciones no se asignan directamente a un solo usuario, sino que están desacopladas en grupos de usuarios.

Un grupo de usuarios agrupa autorizaciones configuradas en función de las tareas. Así, p. ej., todas las autorizaciones necesarias para una actuación del servicio técnico se agrupan en un grupo "Técnico de servicio". Si crea un usuario responsable del servicio técnico, asígnelo únicamente al grupo "Técnico de servicio".

El visor de usuarios puede utilizarse para administrar usuarios en runtime, así como para crearlos, borrarlos y asignarles una autorización.

La administración de usuarios separa la administración de los usuarios de la configuración de autorizaciones. De este modo, ofrece mayor flexibilidad en la protección de acceso.

Durante la configuración ya se puede realizar una asignación previa de la administración de usuarios en el sistema de ingeniería.

10.5.3 Principios básicos

10.5.3.1 Usuarios

Introducción

Los usuarios se crean en la ficha "Usuarios" del editor "Administración de usuarios" y se asignan a los grupos de usuarios. La ficha "Usuarios" forma parte de la administración de usuarios de WinCC.

Abrir

La ficha "Usuarios" se abre haciendo doble clic en "Administración de usuarios" de la ventana del proyecto.

Área de trabajo

En el área de trabajo se administran los usuarios:

- Se crean o borran usuarios.
- Se asignan usuarios a grupos.

Nota

Un usuario sólo puede asignarse a un grupo de usuarios.

Ventana de inspección

Si se selecciona un usuario, se puede modificar la contraseña en el grupo "General". En "Cierre de sesión automático" se determina si el usuario debe desconectarse automáticamente del panel de operador en caso de que transcurra el tiempo establecido sin que se realice una entrada.

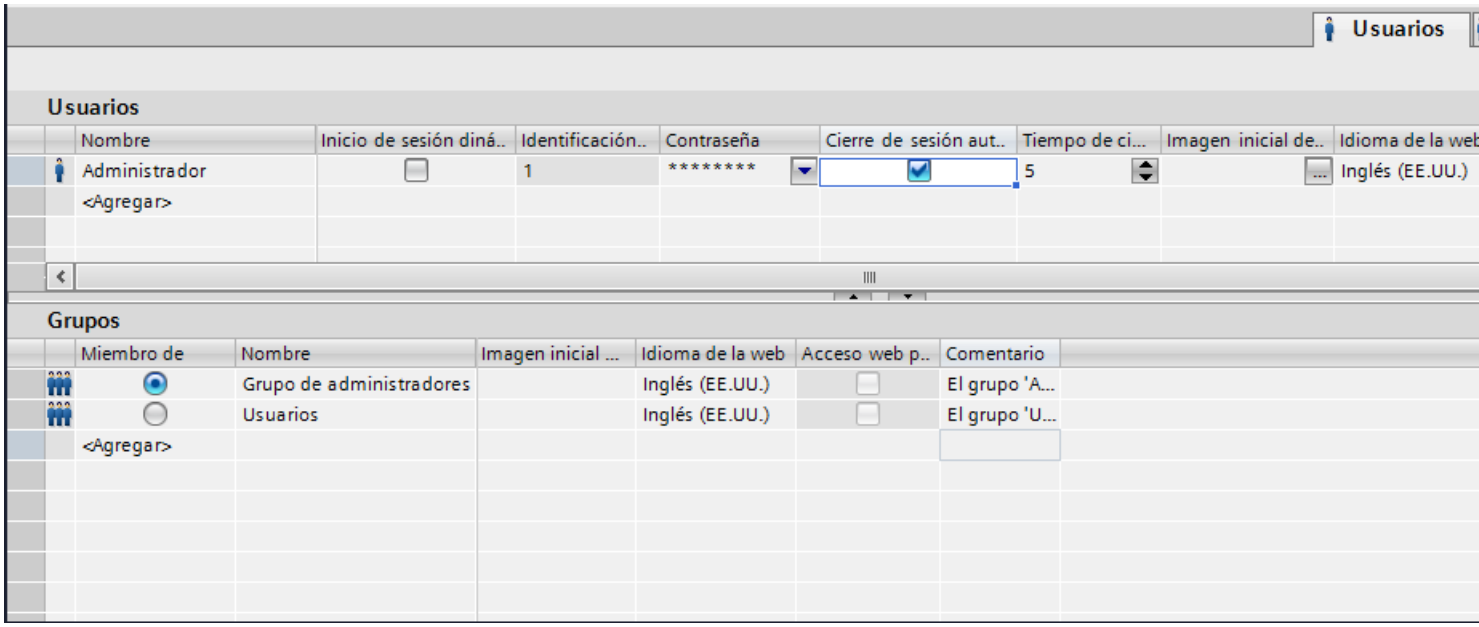
10.5.3.2 Área de trabajo de usuarios

Introducción

El área de trabajo "Usuarios" muestra los usuarios y grupos de usuarios en forma de tablas. En esta área es posible administrar los usuarios y asignarlos a un grupo de usuarios determinado.

Principio

El área de trabajo se compone de las tablas "Usuarios" y "Grupos de usuarios".



La tabla "Usuarios" muestra los usuarios creados. Si selecciona un usuario en esta tabla, en la tabla "Grupos" aparecerá el grupo de usuarios al que pertenece.

Nota

Para el usuario "Administrador" está preconfigurada la contraseña "administrator". Por motivos de seguridad es conveniente cambiar la contraseña de este usuario.

10.5.3.3 Grupos de usuarios

Introducción

Los grupos de usuarios y las autorizaciones se crean en la ficha "Grupos de usuarios" del editor "Administración de usuarios". La ficha "Grupos de usuarios" forma parte de la administración de usuarios de WinCC.

Abrir

Haga doble clic en "Administración de usuarios" de la ventana del proyecto. Seleccione la ficha "Grupos de usuarios".

Área de trabajo

En el área de trabajo se administran los grupos de usuarios y las autorizaciones:

- Se crean nuevos grupos de usuarios y autorizaciones o se borran.
- Se asignan autorizaciones a los grupos de usuarios.

Ventana de inspección

Si se selecciona un grupo de usuarios o una autorización, se puede editar el nombre en el grupo "General". Introduzca además una descripción breve en el grupo "Comentario".

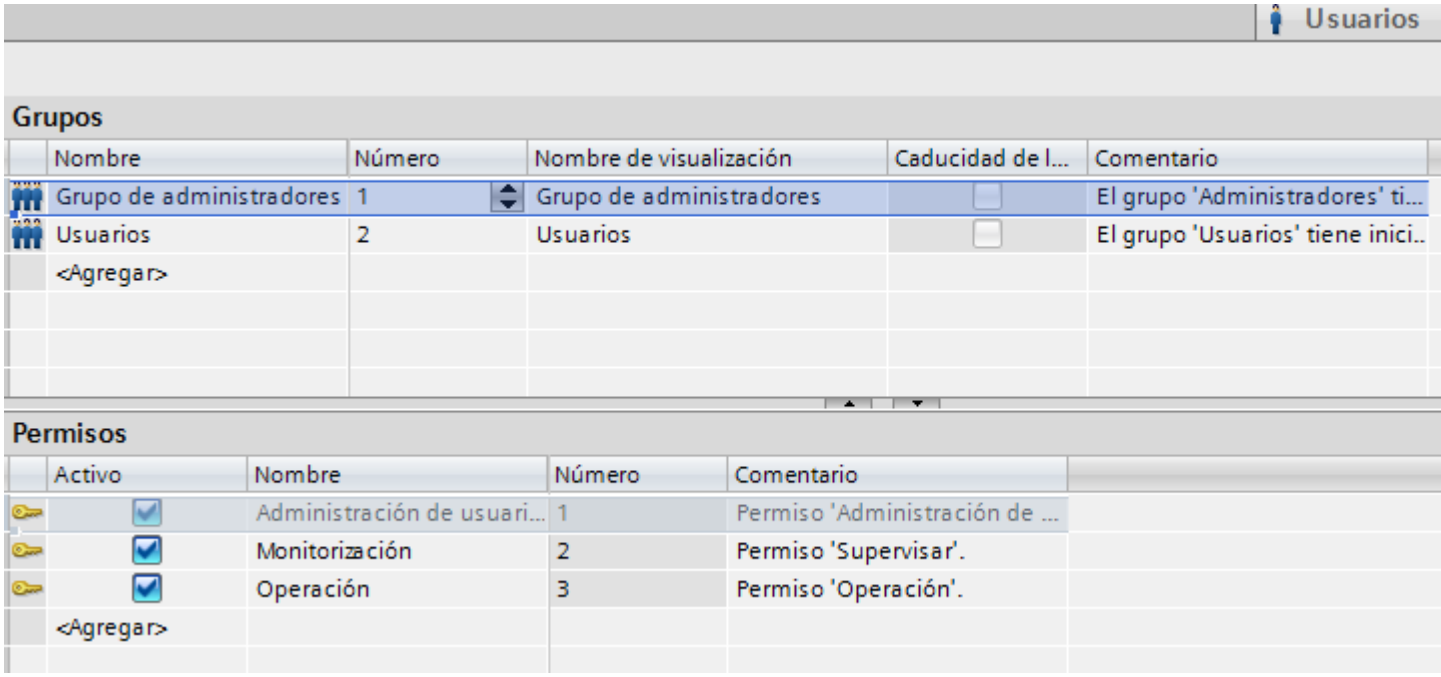
10.5.3.4 Área de trabajo de grupos de usuarios

Introducción

El área de trabajo "Grupos de usuarios" muestra los grupos y sus autorizaciones en forma de tabla. En esta tabla es posible administrar los grupos de usuarios y asignarles autorizaciones.

Principio

El área de trabajo se compone de las tablas "Grupos" y "Autorizaciones".



La tabla "Grupos" muestra los grupos de usuarios creados. Si selecciona un grupo de usuarios en esta tabla, en la columna "Activo" de la tabla "Autorizaciones" aparecerán las autorizaciones que se le hayan asignado.

La administración de usuarios asigna el número de grupo de usuarios y de autorización. El usuario asigna las denominaciones y descripciones.

Los números de las autorizaciones predefinidas están asignados de antemano. Las autorizaciones que cree el usuario pueden editarse libremente. No obstante, al asignarlas debe observarse la univocidad de los números.

10.5.3.5 Configuración de la administración de usuarios

Introducción

En el editor "Configuración de runtime > Administración de usuarios" se configuran los ajustes de seguridad para los usuarios y sus contraseñas en runtime.

Abrir

Haga doble clic en el editor "Configuración de runtime" en la ventana del proyecto. Haga clic en "Administración de usuarios".

Área de trabajo

En el área de trabajo se realizan ajustes para la validez de las contraseñas en runtime. Así, p. ej., se determina la complejidad de la contraseña.

Efectos en runtime

Dependiendo de la configuración, los ajustes de seguridad tienen los efectos siguientes en runtime.

- Grupo "General"
  - Casilla de verificación "Activar límite de intentos de inicio de sesión" activada
    El número introducido en el campo "Intentos no válidos de inicio de sesión" determina cuántos intentos de inicio de sesión puede realizar un usuario antes de ser asignado al grupo "No autorizado".
    Casilla de verificación "Activar límite de intentos de inicio de sesión" no activada
    El usuario tiene en runtime un número ilimitado de intentos de inicio de sesión.
  - Campo "Intentos no válidos de inicio de sesión"
    Si se ha introducido un "4" en el campo, p. ej., y el cuarto intento de inicio de sesión del usuario no da resultado, se asigna automáticamente al grupo "No autorizado".
    Puede introducir entre 1 y 9 intentos.
  - Casilla de verificación "Inicio de sesión sólo posible con contraseña"
    Si la casilla de verificación está activada, se realiza una autenticación de los usuarios mediante la contraseña. El nombre de usuario no es necesario.
    Por consiguiente, para que una asignación sea posible no se debe configurar dos veces la misma contraseña.
- Grupo "Nivel de jerarquía"
  - Casilla de verificación "Activar derechos específicos de grupo para la administración de usuarios"
    Si esta casilla de verificación está activada, los administradores sólo gestionan a los usuarios cuyo número de grupo es inferior o igual al suyo propio.
    Un administrador cuyo número de grupo sea 5, p. ej., sólo podrá administrar usuarios cuyo número de grupo sea menor o igual a 5. Es decir, el administrador sólo podrá asignar a grupos los usuarios que tengan un número de grupo menor o igual a 5.

- Grupo "Contraseña"
  - Casilla de verificación "Activar caducidad de la contraseña" activada
    La contraseña caduca al cabo del número de días indicado en el campo "Vigencia de la contraseña (días)".
    En el editor "Grupos de usuarios" se activará además la columna "Caducidad de la contraseña". Con ello se determina para cada grupo en particular si caducan las contraseñas y si deben guardarse generaciones de contraseñas. Si la caducidad de la contraseña no se ha activado para un grupo, las contraseñas no caducarán nunca.
  - Campo "Tiempo de advertencia (días)"
    El usuario obtiene la información de que su contraseña caducará al cabo del número de días indicado en este campo.
  - Campo "Generaciones de contraseñas"
    Cuando el usuario modifique su contraseña, ésta deberá diferenciarse del número de contraseñas anteriores indicado en este campo. El número posible de generaciones de contraseñas está comprendido entre 1 y 5.
- Grupo "Complejidad de la contraseña"
  - Casilla de verificación "Debe contener caracteres especiales" activada
    El usuario debe introducir una contraseña que contenga al menos un carácter especial, sin importar en qué posición.
  - Casilla de verificación "Debe contener cifra" activada
    El usuario debe introducir una contraseña que contenga al menos una cifra, sin importar en qué posición.
  - Campo "Longitud mínima de la contraseña"
    El usuario debe introducir una contraseña que tenga una longitud mínima, conforme al número introducido en el campo "Longitud mínima de la contraseña".
    La longitud mínima de la contraseña es de 3 caracteres.

10.5.4 Estructurar la administración de usuarios

10.5.4.1 Principios básicos de la administración de usuarios

Principio

Este capítulo está dirigido a cuatro grupos destinatarios. Los temas están ordenados conforme a ello. Los grupos destinatarios son ejemplos de diferentes grupos de personas que trabajan con la administración de usuarios.

1. Administrador OEM
2. Administrator RT
3. Ingeniero
4. Operadores

El administrador OEM crea p.ej. los grupos de usuarios, los usuarios y las autorizaciones de runtime en el sistema de ingeniería del fabricante de máquinas.

El administrador RT administra usuarios en el "Visor de usuarios" de runtime.

El ingeniero asigna las autorizaciones a los grupos de usuarios en el sistema de ingeniería. Asimismo, configura autorizaciones en los objetos.

El operador inicia la sesión en runtime. Sólo puede acceder a un objeto protegido si dispone de la autorización necesaria.

Nota

El grupo destinatario Administrador RT está creado en la administración de usuarios de runtime con el nombre "Grupo de administradores". De ahora en adelante y para una mejor comprensión no se utilizarán las autorizaciones y los grupos de usuarios predefinidos.

10.5.4.2 Administrar usuarios para runtime

Crear una autorización

Introducción

Una autorización se crea y se asigna a uno o varios grupos de usuarios.

Requisitos

El área de trabajo "Grupos de usuarios" está abierta.

Procedimiento

1. Haga doble clic en "Agregar" en la tabla "Autorizaciones".
2. Introduzca "Salir de runtime" como nombre de la autorización.
3. Introduzca una descripción breve en "Comentario".

Crear un grupo de usuarios

Introducción

Para no tener que asignar una autorización a cada usuario, cree grupos de usuarios. Cree un grupo de usuarios y asígnele autorizaciones y usuarios.

Nota

El nombre del grupo de usuarios debe ser unívoco dentro del proyecto. En caso contrario, la entrada no se aceptará.

Nota

Utilización del SIMATIC Logon

Asegúrese de que los nombres de los grupos de usuarios concuerden exactamente en Windows y WinCC.

Requisitos

El área de trabajo "Grupos de usuarios" está abierta.

Procedimiento

1. Haga doble clic en "Agregar" en la tabla "Grupos".
2. Introduzca "Operadores" como "Nombre" del grupo de usuarios.
3. En caso necesario, modifique el "Número" del grupo de usuarios.
4. Introduzca "Operadores" como "Nombre de visualización" del grupo de usuarios.
5. Introduzca una descripción breve en "Comentario".

El visor de usuarios muestra en runtime el nombre de visualización del grupo de usuarios. El nombre de visualización del grupo de usuarios depende del idioma. Configure el nombre en varios idiomas y cambie de idioma en runtime.

Consulte también

Asignar una autorización (Página 8610)

Crear un usuario (Página 8611)

Asignar un usuario a un grupo de usuarios (Página 8612)

Administrar grupos de usuarios (Página 8614)

Administrar usuarios (Página 8613)

Asignar una autorización

Introducción

Cuando se asigna una autorización a un grupo de usuarios, todos los usuarios del grupo tendrán está autorización.

Requisitos

- Se ha creado la autorización "Salir de runtime".
- Un grupo de usuarios "Operadores" está creado.
- El área de trabajo "Grupos de usuarios" está abierta.

Procedimiento

1. En la tabla "Grupos", haga clic en el grupo de usuarios "Operadores". La tabla "Autorizaciones" muestra todas las autorizaciones.
2. En la tabla "Autorizaciones", active la autorización "Salir de runtime".

Nota

La autorización "Salir de runtime" es solo una denominación y no hace referencia a la función "PararRuntime". Esta relación debe crearse a parte. Para ello, configure en un botón la función de sistema "PararRuntime" y seleccione como autorización "Salir de runtime".

Consulte también

Crear un grupo de usuarios (Página 8607)

Crear un usuario

Introducción

Un usuario se crea para que las personas puedan iniciar una sesión en runtime con su nombre de usuario tras la carga en el panel de operador.

Alternativamente, pueden crearse y modificarse usuarios en runtime mediante el "Visor de usuarios".

Para que un usuario tenga autorizaciones debe asignarse a un grupo de usuarios que, a su vez, debe tener autorizaciones asignadas.

La sesión se inicia correctamente si el nombre de usuario entrado coincide con un usuario de runtime. Asimismo, la contraseña introducida debe coincidir con la contraseña almacenada para el usuario.

Nota

Observe el uso de mayúsculas y minúsculas.

Requisitos

El área de trabajo "Usuarios" está abierta.

Procedimiento

1. Haga doble clic en "Agregar" en la tabla "Usuarios".
2. Introduzca el nombre de usuario "Maestro".

   Nota

   El nombre de usuario debe ser unívoco dentro del proyecto. En caso contrario, la entrada no se aceptará.

   Nota

   Al introducir un nombre de usuario y una contraseña, no utilice caracteres especiales como / " § $ % ? ' &.

3. En la columna "Contraseña", haga clic en el botón . Se abre un cuadro de diálogo para introducir la contraseña.
4. Introduzca la contraseña del usuario.
5. Introduzca de nuevo la contraseña en el campo de confirmación situado debajo.
6. Cierre el cuadro de diálogo con el símbolo .
7. Si la sesión del usuario debe cerrarse pasado un tiempo determinado, active "Cierre de sesión automático".
8. Haga clic en la columna "Tiempo de cierre de sesión". El "Tiempo de cierre de sesión" predeterminado es de 5 minutos.
9. Introduzca una descripción breve en "Comentario".

Consulte también

Crear un grupo de usuarios (Página 8607)

Asignar un usuario a un grupo de usuarios

Introducción

Cuando se asigna un usuario a un grupo de usuarios, tendrá las autorizaciones del grupo de usuarios.

Nota

Un usuario sólo puede asignarse a un grupo de usuarios. Al verificar la consistencia y compilar el proyecto se comprueba la asignación.

Requisitos

- El usuario "Maestro" está creado.
- Un grupo de usuarios "Operadores" está creado.
- El área de trabajo "Usuarios" está abierta.

Procedimiento

1. En la tabla "Usuarios", haga clic en el usuario "Maestro". La tabla "Grupos" muestra todos los grupos de usuarios.
2. En la tabla "Grupos", active el grupo de usuarios "Operadores".

Consulte también

Crear un grupo de usuarios (Página 8607)

Administrar usuarios

Introducción

En la vista se administran los usuarios y se asignan los grupos.

Requisitos

El área de trabajo "Usuarios" está abierta.

Cambiar nombre de usuario

1. En la columna "Nombre" de la tabla "Usuarios" haga doble clic en el campo del nombre a cambiar.
2. Cambie el nombre de usuario.
3. Confirme con la tecla <Intro>.

Opcionalmente, seleccione el usuario en el área de trabajo. Modifique el nombre de usuario en la ventana de inspección "Propiedades > Propiedades > General".

Modificar la contraseña del usuario

1. En la columna "Contraseña" de la tabla "Usuarios" haga clic en el botón . Se abrirá un cuadro de diálogo para entrar la contraseña.
2. Introduzca la nueva contraseña en el campo "Introducir contraseña".
3. Vuelva a introducir la nueva contraseña en el campo "Confirmar contraseña".
4. Confirme con la tecla <Intro>.

Opcionalmente, seleccione el usuario en el área de trabajo. Modifique la contraseña en la ventana de inspección "Propiedades > Propiedades > General".

Visualizar columnas no visibles

1. Mueva el puntero del ratón hacia el encabezado de la tabla "Usuarios".
2. Abra el menú contextual con el botón derecho del ratón y active la vista de la columna "Tiempo de cierre de sesión", p. ej.

Modificar el tiempo de cierre de sesión del usuario

1. Hacer doble clic en la columna "Tiempo de cierre de sesión" del área "Usuario" que se encuentra en el campo del nombre a cambiar.
2. Modifique el tiempo de cierre de sesión.
3. Confirme con la tecla <Intro>.

Opcionalmente, seleccione el usuario en el área de trabajo. Modifique el tiempo de cierre de sesión en la ventana de inspección "Propiedades > Propiedades > Cierre de sesión automático".

Borrar un usuario

1. Seleccione la fila del usuario que debe borrarse.
2. Abra el menú contextual con el botón derecho del ratón y seleccione el comando "Borrar".

Nota

Los usuarios predefinidos no pueden borrarse.

Consulte también

Crear un grupo de usuarios (Página 8607)

Administrar grupos de usuarios

Introducción

En el área de trabajo se administran grupos de usuarios y se asignan autorizaciones para la utilización en runtime.

Requisitos

El área de trabajo "Grupos de usuarios" está abierta.

Cambiar el nombre del grupo de usuarios

1. En la columna "Nombre" de la tabla "Grupos" haga doble clic en el campo del nombre de grupo a cambiar.
2. Cambie el nombre del grupo de usuarios.
3. Confirme con la tecla <Return>.

Opcionalmente, seleccione el grupo de usuarios en el área de trabajo. Modifique el nombre en la ventana de inspección "Propiedades > Propiedades > General".

Nota

Los grupos de usuarios predefinidos no pueden borrarse.

Visualizar columnas no visibles

1. Mueva el puntero del ratón hacia el encabezado de la tabla "Grupos".
2. Abra el menú contextual con el botón derecho del ratón y active la vista de la columna "Nombre de visualización", p. ej.

Modificar el nombre de visualización del grupo de usuarios

1. En la columna "Nombre de visualización" de la tabla "Grupos" haga doble clic en el campo del nombre de visualización a cambiar.
2. Cambie el nombre de visualización del grupo de usuarios.
3. Confirme con la tecla <Return>.

Opcionalmente, seleccione el grupo de usuarios en el área de trabajo. Modifique el nombre de visualización en la ventana de inspección "Propiedades > Propiedades > General".

Borrar un grupo de usuarios

1. Seleccione la fila del grupo de usuarios que debe borrarse.
2. Abra el menú contextual con el botón derecho del ratón y seleccione el comando "Borrar".

Nota

Los grupos de usuarios predefinidos no pueden borrarse.

Cambiar el nombre de la autorización

1. En la columna "Nombre" de la tabla "Autorizaciones" haga doble clic en el campo del nombre a cambiar.
2. Cambie el nombre de la autorización.
3. Confirme con la tecla <Return>.

Opcionalmente, seleccione la autorización en el área de trabajo. Modifique el nombre en la ventana de inspección "Propiedades > Propiedades > General".

Borrar una autorización

1. Seleccione la fila de la autorización que debe borrarse.
2. Abra el menú contextual con el botón derecho del ratón y seleccione el comando "Borrar".

Nota

Las autorizaciones predefinidas no pueden borrarse.

Consulte también

Crear un grupo de usuarios (Página 8607)

10.5.4.3 Administrar usuarios en runtime

Usuarios en runtime

Principio

Se crean usuarios y grupos de usuarios y se les asignan autorizaciones. Los objetos se configuran con autorizaciones. Después de la cargar en el panel de operador, todos los objetos de runtime que han sido configurados con una autorización están protegidos contra accesos no autorizados.

Visor de usuarios

Cuando se configura un visor de usuarios en el sistema de ingeniería, es posible administrar usuarios en el visor de usuarios tras la carga en el panel de operador.

ATENCIÓN
Las modificaciones en el visor de usuarios de runtime son efectivas inmediatamente. Las modificaciones en runtime no se actualizan en el sistema de ingeniería. Al cargar la administración de usuarios en el panel de operador, se sobrescriben todas las modificaciones del visor de usuarios previa consulta y en función de las opciones de carga seleccionadas.

Los usuarios que tienen la autorización "Administración de usuarios" pueden acceder sin limitaciones al visor de usuarios. Desde allí se administran todos los usuarios. Los demás usuarios sólo tienen un acceso restringido al visor de usuarios, donde pueden administrarse a sí mismos.

Visor de usuarios

Finalidad

Un visor de usuarios se configura en el sistema de ingeniería para administrar los usuarios también en runtime.

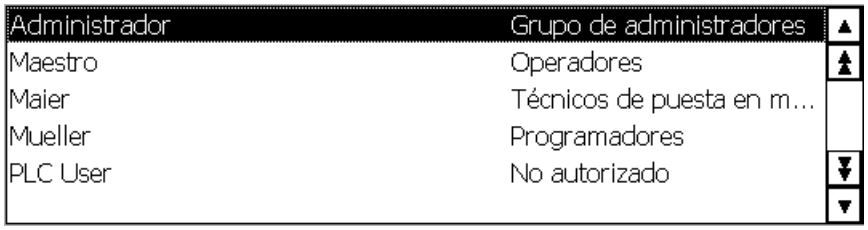
Estructura

El visor de usuarios muestra lo siguiente en cada fila:

- El usuario
- El grupo de usuarios correspondiente.

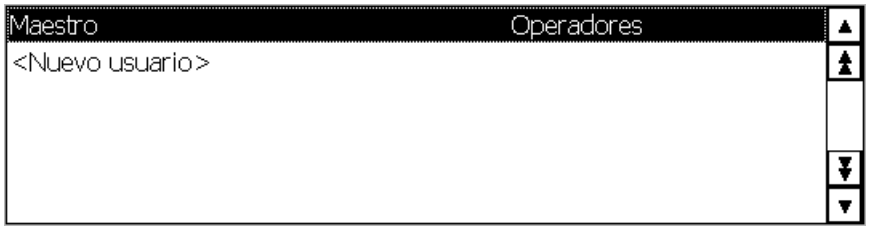
Si no hay ningún usuario conectado, el visor de usuarios está vacío. Tras iniciar la sesión se visualizarán los contenidos de sendos campos.

Visor de usuarios de un administrador



Cuando un administrador está conectado, el visor de usuarios muestra todos los usuarios. El administrador modifica el nombre del usuario y la contraseña. El administrador crea usuarios nuevos y los asigna a un grupo de usuarios existentes.

Visor de usuarios de un usuario



Si no hay ningún administrador conectado, el visor de usuarios sólo muestra el usuario conectado. El usuario puede modificar su contraseña.

Visor de usuarios

Finalidad

Un visor de usuarios se configura en el sistema de ingeniería para administrar los usuarios también en runtime.

Estructura

El visor de usuarios muestra lo siguiente en cada fila:

- El usuario
- Su contraseña codificada
- El grupo de usuarios correspondiente
- El tiempo de cierre de sesión

Si no hay ningún usuario conectado, el visor de usuarios está vacío. Tras iniciar la sesión se visualizarán los contenidos de sendos campos.

Visor de usuarios de un administrador

Cuando un administrador está conectado, el visor de usuarios muestra todos los usuarios. El administrador modifica el nombre del usuario y la contraseña. El administrador crea usuarios nuevos y los asigna a un grupo de usuarios existentes.

Visor de usuarios de un usuario

Si no hay ningún administrador conectado, el visor de usuarios sólo muestra el usuario conectado. El usuario puede modificar su contraseña y su tiempo de cierre de sesión.

Configurar el visor de usuarios

Introducción

Un visor de usuarios se configura en el sistema de ingeniería para administrar los usuarios también en runtime.

Requisitos

Se ha creado una imagen.

Procedimiento

1. En la ventana de herramientas, seleccione el objeto "Visor de usuarios" de la categoría "Controles".
2. Arrastre el objeto "Visor de usuarios" hasta la imagen con la función Drag&Drop.
3. En la ventana de inspección haga clic en "Propiedades > Propiedades".
4. Defina el aspecto del "Visor de usuarios".
5. Seleccione, por ejemplo, "Representación > Adaptación del tamaño > Adaptar objeto al contenido".

Resultado

Se ha creado un visor de usuarios en la imagen.

Crear un usuario

Introducción

Un usuario se crea para que las personas puedan iniciar una sesión en runtime con su nombre de usuario.

Alternativamente pueden crearse usuarios en el sistema de ingeniería y cargarlos en el panel de operador.

La sesión sólo se inicia correctamente si el nombre de usuario entrado coincide con un usuario de runtime. Asimismo, la contraseña entrada debe coincidir con la del usuario.

Nota

Observe el uso de mayúsculas y minúsculas.

Nota

Al introducir un nombre de usuario y una contraseña, no utilice caracteres especiales como / " § $ % ? ' &.

El usuario se asigna a un grupo de usuarios. A continuación, el usuario tendrá las autorizaciones del grupo de usuarios.

Nota

En runtime debe asignarse un usuario a un grupo de usuarios. El grupo de usuarios se crea en el sistema de ingeniería. La denominación del grupo de usuarios depende del idioma.

Requisitos

- El visor de usuarios está abierto.
- Un grupo de usuarios "Group 2" está creado.

Procedimiento

1. En el visor de usuarios, haga clic en <Nuevo usuario>. Se abre un cuadro de diálogo.
2. Introduzca el nombre de usuario "Maestro".
3. Pulse la tecla <Intro>.
4. Haga clic en "Contraseña".
5. Introduzca la contraseña del usuario.
6. Pulse la tecla <Intro>. La contraseña es invisible.
7. Haga clic en la columna "Grupo".

8. Seleccione el grupo "Group 2".

User	Password	Group	Logoff time
Adminis...	********	Grupo de adm...	5
Johnson	********	Grupo de adm...	5
Maestro	********	Group 2	5
PLC User	********	Unauthorized	5

9. Pulse la tecla <Intro>.
10. Haga clic en la columna "Tiempo de cierre de sesión".
11. Introduzca el tiempo tras el cual se cerrará automáticamente la sesión del usuario.

Administrar usuarios con el visor de usuarios

Introducción

En caso de haber configurado un visor de usuarios en el sistema de ingeniería, es posible administrar los usuarios y los grupos de usuarios en runtime.

ATENCIÓN
Las modificaciones en el visor de usuarios de runtime son efectivas inmediatamente. Las modificaciones en runtime no se actualizan en el sistema de ingeniería. Al cargar la administración de usuarios en el panel de operador, se sobrescriben todas las modificaciones del visor de usuarios previa consulta y en función de las opciones seleccionadas.

Requisitos

- Runtime activado.
- Se ha creado un visor de usuarios complejo.
- La imagen con el visor de usuarios está abierta.
- Tiene la autorización predeterminada "Administración de usuarios".

Nota

Si no tiene la autorización "Administración de usuarios", sólo podrá modificar su propia contraseña y su tiempo de cierre de sesión.

Cambiar nombre de usuario

1. Introduzca un nuevo nombre de usuario en la columna "Usuarios" del visor de usuarios.
2. Confirme con la tecla <Intro>.

Nota

El usuario no podrá volver a iniciar una sesión en runtime con su antiguo nombre. Si borra el nombre y pulsa la tecla <Intro> se borrará el usuario.

Modificar la contraseña del usuario

La disponibilidad del visor de usuarios complejo depende del dispositivo.

1. Introduzca una nueva contraseña en la columna "Contraseña" del visor de usuarios.
2. Confirme con la tecla <Intro>.

Nota

El usuario no podrá volver a iniciar una sesión en runtime con su antigua contraseña.

Si borra la contraseña en el visor de usuarios complejo y pulsa la tecla <Intro> se borrará el usuario.

Modificar el tiempo de cierre de sesión del usuario

1. Introduzca un nuevo tiempo de cierre de sesión en la columna "Tiempo de cierre de sesión" del visor de usuarios.
2. Confirme con la tecla <Intro>.

Borrar un usuario

1. Haga clic en el nombre del usuario que debe borrarse.
2. Borre el nombre.
3. Pulse la tecla <Intro>.

Nota

El usuario no podrá volver a iniciar una sesión en runtime.

Asignar un usuario a otro grupo de usuarios

1. Activar el campo Grupo de usuarios para el usuario correspondiente.
2. Seleccione un grupo de usuarios.
3. Confirme la selección con la tecla <Intro>.

Desbloquear usuarios bloqueados

Desbloquear usuarios bloqueados

En los "Ajustes de runtime > Administración de usuarios" está activada la casilla de verificación "Activar límite de intentos de inicio de sesión"
En el campo "Intentos de inicio de sesión no válidos" aparece el número 3.

Si un usuario hace tres intentos fallidos de iniciar una sesión, p. ej. por introducir una contraseña errónea, será asignado al grupo "No autorizado". En este caso, el usuario perderá todas las autorizaciones. Aunque todavía puede iniciar la sesión, ya no tendrá ninguna autorización. Sólo un usuario con derechos de administrador puede vuelve a asignar el usuario no autorizado a un grupo.

Iniciar una sesión de usuario

Introducción

Normalmente, una sesión de usuario se inicia con un botón especial. Entonces aparece el cuadro de diálogo de inicio de sesión.

El cuadro de diálogo de inicio de sesión aparece por defecto al acceder a un objeto protegido si:

- No se ha iniciado ninguna sesión de usuario en runtime
- El usuario que ha iniciado la sesión no posee la autorización necesaria.

Nota

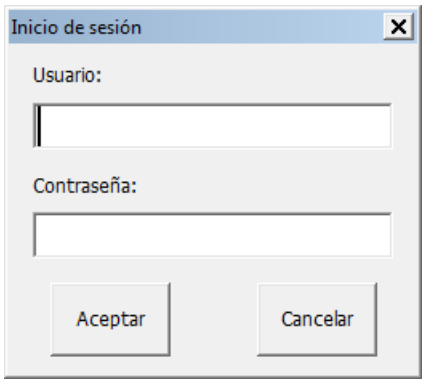
Al pulsar un botón con protección de acceso en los paneles de operador OP 73, OP 77A, TP 177A y en los paneles básicos (Basic Panels) aparece siempre el cuadro de diálogo de inicio de sesión.

Requisitos

- En "Ajustes de runtime > Administración de usuario"
  - está activada la casilla de verificación "Activar límite de intentos de inicio de sesión".
  - aparece el número 3 en el campo "Intentos de inicio de sesión no válidos".
- La función de sistema "MostrarDiálogoIniciarSesion" está configurada en un botón con el título "Inicio de sesión".

Procedimiento

1. Haga clic en el botón "Logon". Aparecerá el cuadro de diálogo de inicio de sesión.



2. Introduzca su nombre de usuario, que debe ser idéntico al asignado en la administración de usuarios, p. ej. "Maestro".
   Si antes de usted se ha conectado otro usuario, se visualizará su nombre.
3. Escriba la contraseña requerida. La entrada se mantiene oculta.
4. Cierre el cuadro de diálogo con el botón "Aceptar".

Inicio de sesión correcto

Si ha introducido el nombre de usuario "Maestro" y la contraseña entrada coincide con la almacenada, se iniciará una sesión del usuario "Maestro" en runtime. Tiene la autorización del usuario "Maestro".

El acceso a un objeto protegido, p. ej. el botón "Archivar", sólo se permitirá si el usuario "Maestro" dispone de la autorización necesaria. La acción configurada se ejecutará inmediatamente.

Si la próxima conexión es correcta pero no dispone de la autorización necesaria, aparecerá un aviso de error al respecto. De todas formas, seguirá conectado en runtime.

Inicio de sesión incorrecto

Aparecerá un aviso de error.

Por motivos de seguridad, usted o el usuario conectado previamente ya no tendrá ninguna autorización. De todas formas, podrá acceder a objetos no protegidos, aunque el visor de usuarios no mostrará ninguna entrada. El visor de usuarios y las autorizaciones cambian en el próximo inicio de sesión correcto.

Si el tercer intento de inicio de sesión no ha dado resultado, el usuario se asigna al grupo predefinido "No autorizado" Por tanto, no configure ningún grupo de usuarios con ese nombre.

El usuario será desconectado cuando se llame la función "Cerrar sesión" o cuando haya transcurrido el tiempo de cierre de sesión del usuario.

10.5.4.4 Configurar una protección de acceso

Protección de acceso

Introducción

Para proteger el acceso a un objeto, configure una autorización para ese objeto. Todos los usuarios conectados que posean esta autorización, podrán acceder al objeto. Cuando un usuario no posee la autorización de uso de un objeto, automáticamente aparece el cuadro de diálogo de inicio de sesión.

Nota

Para procesar usuarios, contraseñas y grupos de usuarios, p. ej. en el controlador, existen varias funciones de sistema en la "Administración de usuarios".

Configurar una autorización de uso (Basic, Advanced; Professional)

Introducción

La autorización "Salir de runtime" se configura en un botón. De esta manera sólo podrán tener acceso los usuarios que tengan esa autorización, p. ej. todos los usuarios del grupo "Operadores".

De este modo, el acceso al botón queda protegido. Si un usuario conectado, perteneciente al grupo "Operadores" y con la autorización necesaria, hace clic en el botón, se saldrá del runtime.

En un ejemplo se describe detalladamente cómo configurar un botón con protección de acceso.

Requisitos

- El grupo de usuarios "Operadores" está creado.
- Se ha creado la autorización "Salir de runtime".
- Se ha creado y abierto una imagen.
- La imagen contiene un botón.

Procedimiento

1. Haga clic en el botón de la imagen.
2. En la ventana de inspección haga clic en "Propiedades > Propiedades >Seguridad".
3. Seleccione como autorización "Salir de runtime".

4. En la ventana de inspección seleccione "Propiedades > Eventos > Hacer clic".
5. Seleccione una función de sistema de la lista de funciones, p. ej., "PararRuntime".

Nota

Los eventos "Activar" y "Desactivar" sirven exclusivamente para reconocer si un objeto está seleccionado o deseleccionado. No obstante, los eventos no lanzan ninguna petición de contraseña.

Por lo tanto, no utilice los eventos "Activar" o "Desactivar" si quiere configurar una protección de acceso en la llamada de la función del objeto.

10.5.5 Referencia

10.5.5.1 Objetos con protección de acceso

Introducción

Los objetos siguientes pueden configurarse con una autorización:

- Campo de fecha y hora
- Campo E/S
- Campo E/S gráfico
- Visor de recetas
- Interruptor
- Botón
- Campo E/S simbólico
- Visor de diagnóstico del sistema

10.5.5.2 Autorizaciones y grupos de usuarios predefinidos

Principio

Las autorizaciones y los grupos de usuarios predefinidos tienen los números siguientes:

Grupo de usuarios	Número
"Grupo de administradores"	1
"Usuarios"	2

Autorización	Número
"Administración de usuarios"	1
"Supervisar"	2
"Manejar"	3

10.5.6 Ejemplos

10.5.6.1 Ejemplo: Configurar un botón con el cuadro de diálogo de inicio de sesión

Tarea

En el siguiente ejemplo, configure en un botón la función "MostrarDialogoIniciarSesión". Este botón se utilizará, por ejemplo, durante el cambio de turno, cuando inicie la sesión un usuario distinto. Además, en ese momento se cerrará la sesión del usuario anterior.

Nota

De forma predeterminada, el cuadro de diálogo de inicio de sesión aparece en runtime cuando se accede a un objeto protegido. Existen dos posibilidades: O no ha iniciado la sesión ningún usuario, o el usuario no tiene la autorización necesaria.

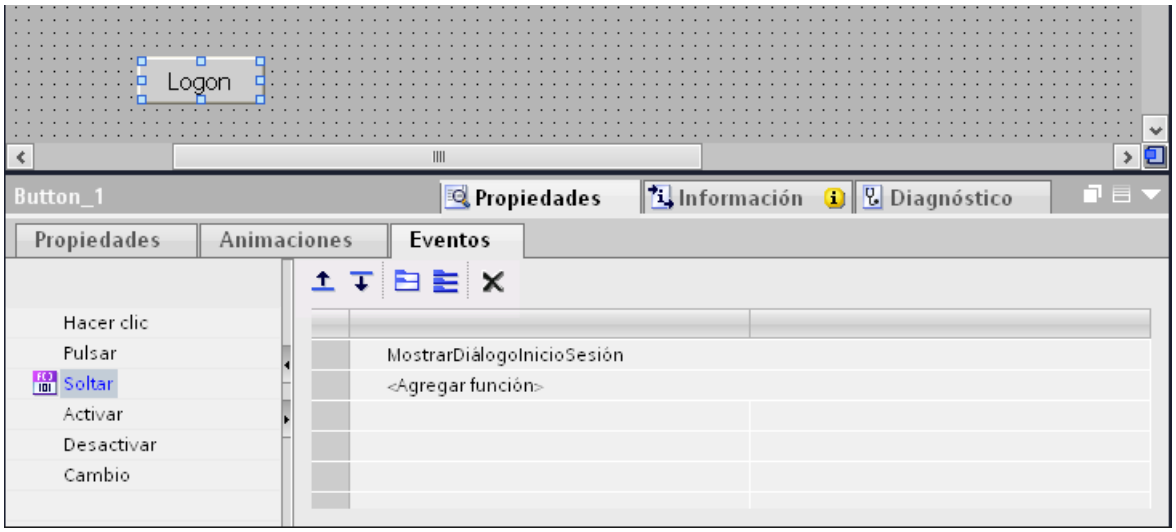
Requisitos

- Se ha creado una imagen.
- Se ha creado un botón nuevo en la imagen.

Procedimiento

1. Haga clic en el botón de la imagen.
2. En la ventana de inspección haga clic en "Propiedades > Eventos > Soltar".

3. En la tabla "Lista de funciones", haga clic en la entrada "Agregar función".
4. Seleccione la función del sistema "MostrarDialogoIniciarSesion" en el grupo "Administración de usuarios".



Resultado

Cuando el usuario haga clic en este botón en runtime, se ejecutará la función "MostrarDialogoIniciarSesion". Al ejecutarse la función "MostrarDialogoIniciarSesion" aparecerá el cuadro de diálogo de inicio de sesión. El usuario inicia la sesión con su nombre de usuario y su contraseña.

10.5.6.2 Ejemplo: Registrar el inicio y el cierre de sesión

Tarea

En el siguiente ejemplo, configure la función "IdentificarCambioDeUsuario" para que se ejecute con el evento "Cambio de usuario".

Principio

Cuando un usuario inicia o cierra su sesión, se ejecuta la función "IdentificarCambioDeUsuario". Cuando se llama una función, se emite un aviso del sistema indicando el usuario correspondiente.

Este aviso del sistema se puede archivar. Al archivar, el aviso del sistema se marcará con la fecha y hora. De este modo es posible saber qué usuario ha estado conectado al panel de operador, cuándo y por cuánto tiempo.

Requisitos

- Se ha creado un panel de operador con Runtime Advanced.
- La ventana de inspección está abierta.

Procedimiento

1. Haga doble clic en "Planificador de tareas" de la ventana del proyecto.
2. Haga doble clic en "Agregar" en la tabla del área de tareas.
3. En "Nombre", introduzca "Logon-Protocol".
4. Seleccione el disparador "Cambio de usuario".
5. En la ventana de inspección abra "Propiedades > Eventos".
6. En la tabla "Lista de funciones", haga clic en la entrada "Agregar función".
7. Seleccione la función de sistema "IdentificarCambioDeUsuario".

Resultado

Cuando un usuario inicie o cierre la sesión, se emitirá un aviso del sistema.

10.5.6.3 Ejemplo de una administración de usuarios

Ejemplo: Estructura de una administración de usuarios

Tarea

En el ejemplo siguiente se crea una administración de usuarios para diferentes usuarios y grupos de usuarios. El ejemplo se basa en un perfil de necesidades típico de la técnica de fabricación.

Principio

En una instalación y en un proyecto participan grupos de personas muy diversos. Cada grupo de personas protege sus datos y funciones de los demás. Con este fin se crean usuarios y se asignan a un grupo de usuarios.

Es posible reproducir diferentes vistas mediante grupos de usuarios.

Ejemplo:

- Vista organizativa: operadores de puesta en marcha, operadores, turno I, turno II
- Vista tecnológica: control de ejes, cambiadores de herramientas, instalación norte, instalación sur

El ejemplo siguiente se basa en la vista organizativa.

Cada grupo de usuarios requiere una protección de acceso característica: un grupo de usuarios posee autorizaciones para determinados casos de aplicación. Así, p. ej., un programador modifica registros de recetas.

En el ejemplo se crean los usuarios Müller, Grupo Meier y Maestro y se asignan a diferentes grupos de usuarios.

La Sra. Müller trabaja de programadora con el sistema de ingeniería. El Grupo Meier son los operadores de puesta en marcha. El Sr. Maestro es un operador.

Requisitos

- Debe haberse creado un proyecto nuevo.
- El editor "Administración de usuarios" está abierto.

Resumen de los pasos a seguir

El manejo de la administración de usuarios en el ejemplo funciona de la siguiente manera:

1. Crear autorizaciones: el ingeniero determina qué autorizaciones se requieren para la protección de acceso.
2. Configurar autorizaciones: el ingeniero determina qué objetos pueden manejarse y qué funciones pueden ejecutarse.
3. Crear grupos de usuarios y asignar autorizaciones: el administrador crea los grupos de usuarios junto con el ingeniero. Mediante las autorizaciones, el ingeniero determina quién puede manejar objetos y quién puede modificar parámetros.
4. Crear usuarios y asignarlos a un grupo de usuarios: el administrador administra los usuarios.

Resultado

El objetivo es obtener la estructura siguiente de la administración de usuarios, compuesta por usuarios, grupos de usuarios y autorizaciones:

Usuarios			Grupos de usuarios	Autorizaciones			
Müller	Meier	Maestro	Funciones	Modificar registros de receta	Modificar parámetros del sistema	Modificar parámetros del proceso	Administrar
			Grupo de administradores				x
X			Programadores	X			
	X		Técnicos de puesta en marcha	X	X	X	
		X	Operadores	x			

El usuario "Maestro", que pertenece al grupo "Operadores", tiene acceso al botón configurado "Hacia el visor de recetas".

Nota

Alternativamente, puede crear como operadores varios grupos de usuarios con diferentes autorizaciones de uso, p. ej. Operadores de nivel 1, Operadores de nivel 2.

Ejemplo: Crear y configurar una autorización

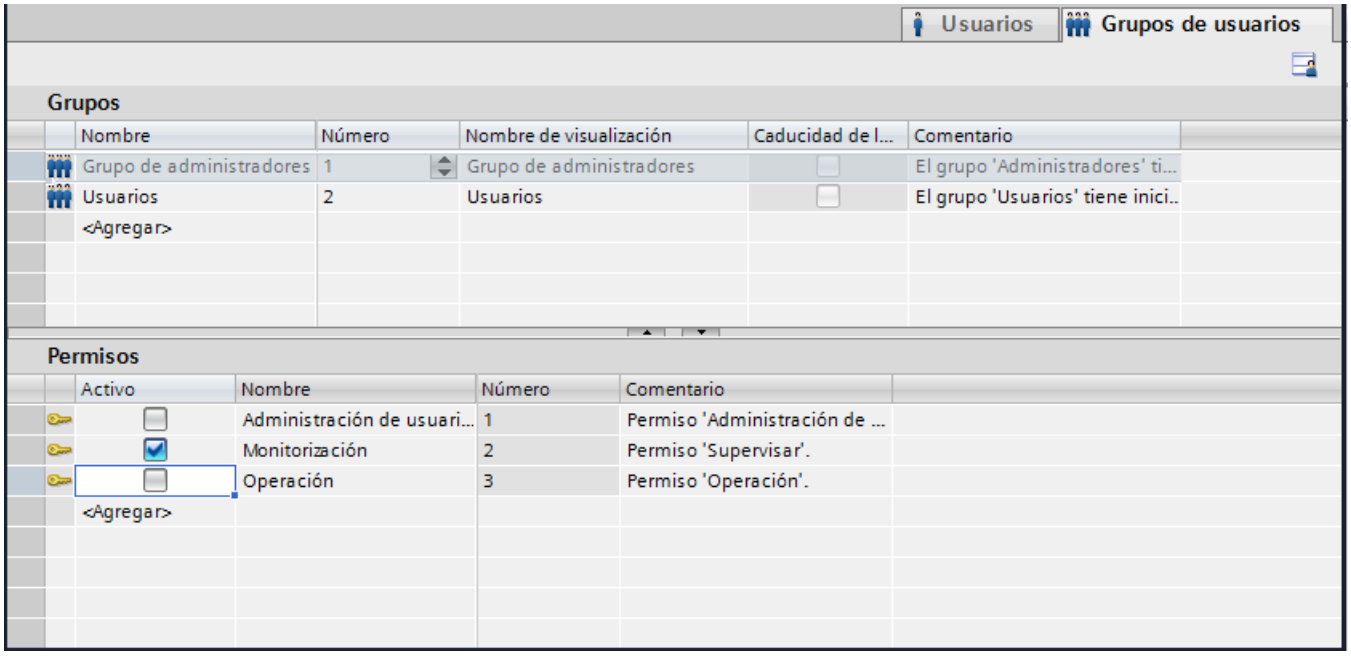
Tarea

En el ejemplo siguiente se crean las autorizaciones.

Procedimiento

1. Abra el área de trabajo "Grupos de usuarios".
2. Haga doble clic en "Agregar" en la tabla "Autorizaciones".
3. Introduzca "Modificar registros de receta" en el nombre de la autorización.
4. Repita los pasos 2 y 3 para crear más autorizaciones: "Modificar parámetros de sistema", "Modificar parámetros de proceso".

Resultado



Ejemplo: Configurar un botón con protección de acceso

Tarea

En el ejemplo siguiente se crea un botón para cambiar de imagen con la ayuda de una función de sistema. Proteja el botón "Hacia el visor de recetas" de manejos no autorizados. Para ello, configure la autorización "Modificar registros de receta" en el botón "Hacia el visor de recetas".

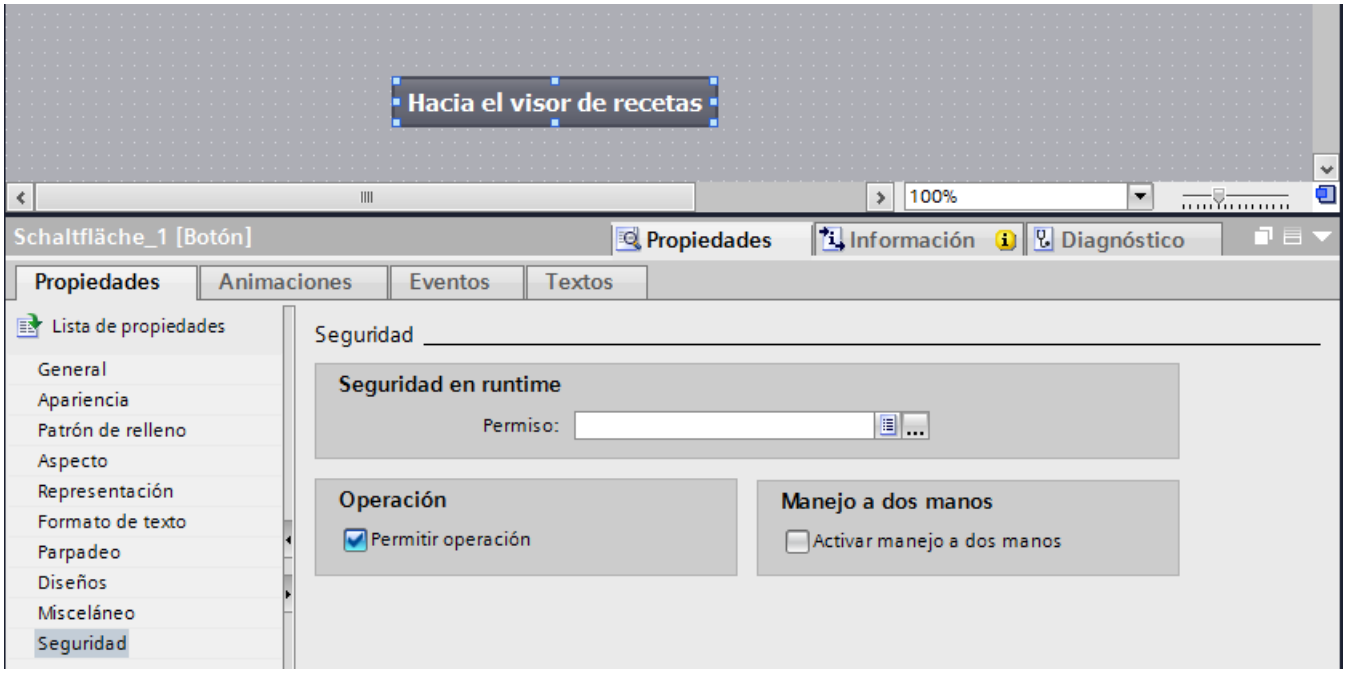
Requisitos

- Se ha creado una autorización "Modificar registros de receta".
- Se ha creado una imagen "Recetas".
- Se ha creado y abierto una imagen "Inicio".
- Se ha creado y seleccionado un botón en la imagen "Inicio".

Procedimiento

1. En la ventana de inspección haga clic en "Propiedades > Propiedades >General".
2. Introduzca el texto "Hacia el visor de recetas".
3. En la ventana de inspección haga clic en "Propiedades > Eventos > Hacer clic".
4. En la primera fila de la tabla "Lista de funciones", haga clic en la entrada "Agregar función".
5. Seleccione la función del sistema "ActivarImagen" en el grupo "Imágenes".
6. Haga clic en el botón ... del campo "Nombre de imagen". Se abre un cuadro de diálogo para seleccionar la imagen.
7. Seleccione la imagen "Recetas" y cierre el cuadro de diálogo con el botón ✓.
8. En la ventana de inspección haga clic en "Propiedades > Propiedades > Seguridad".
9. Seleccione la autorización "Modificar registros de receta".

Resultado



El acceso al botón "Hacia el visor de recetas" queda protegido. Cuando el usuario "Meier" haga clic en este botón en runtime, p. ej., se abrirá la imagen "Visor de recetas". Siempre y

cuando el usuario "Meier" haya iniciado una sesión correctamente y tenga la autorización necesaria. La imagen "Recetas" contiene un visor de recetas y otros objetos de imagen.

Si el usuario conectado no tiene la autorización necesaria o no ha iniciado la sesión ningún usuario, aparecerá el cuadro de diálogo de inicio de sesión.

Ejemplo: Crear grupos de usuarios y asignar autorizaciones

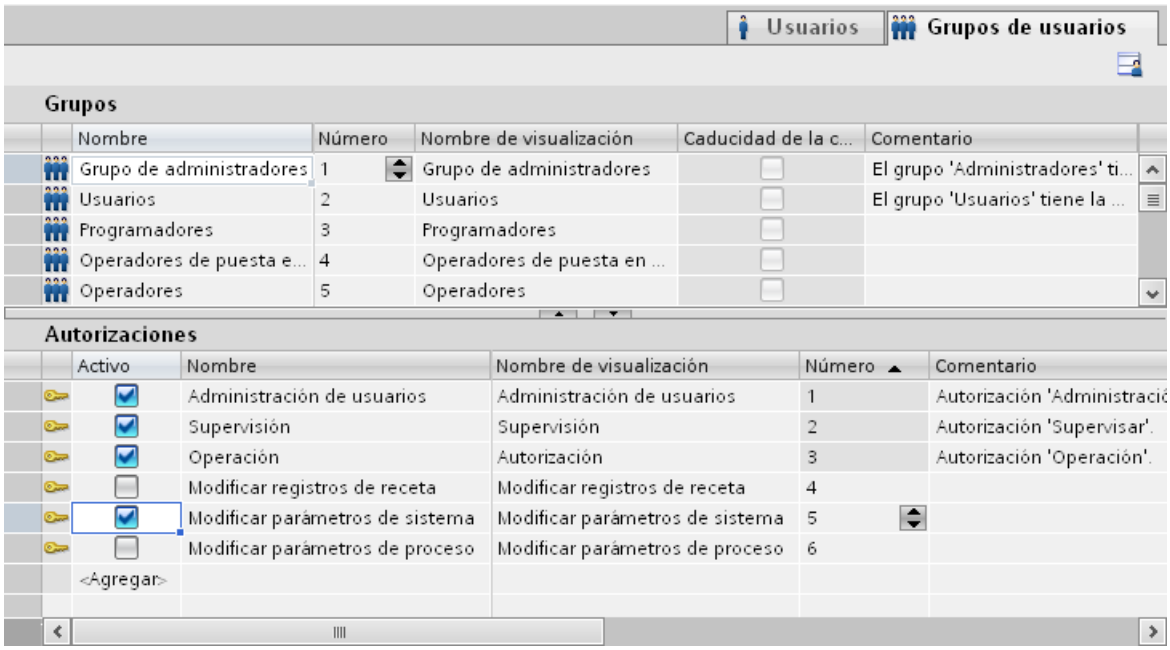
Tarea

En el ejemplo siguiente se crean los grupos de usuarios y se les asignan autorizaciones.

Procedimiento

1. Abra el área de trabajo "Grupos de usuarios".
2. Haga doble clic en "Agregar" en la tabla "Grupos".
3. En "Nombre", escriba "Programadores".
4. Para crear los grupos de usuarios "Operadores de puesta en marcha" y "Operadores" se deberán repetir los pasos 2 y 3.
5. Haga clic en "Grupo de administradores" de la tabla "Grupos".
6. En la tabla "Autorizaciones" active la autorización "Modificar parámetros de sistema".

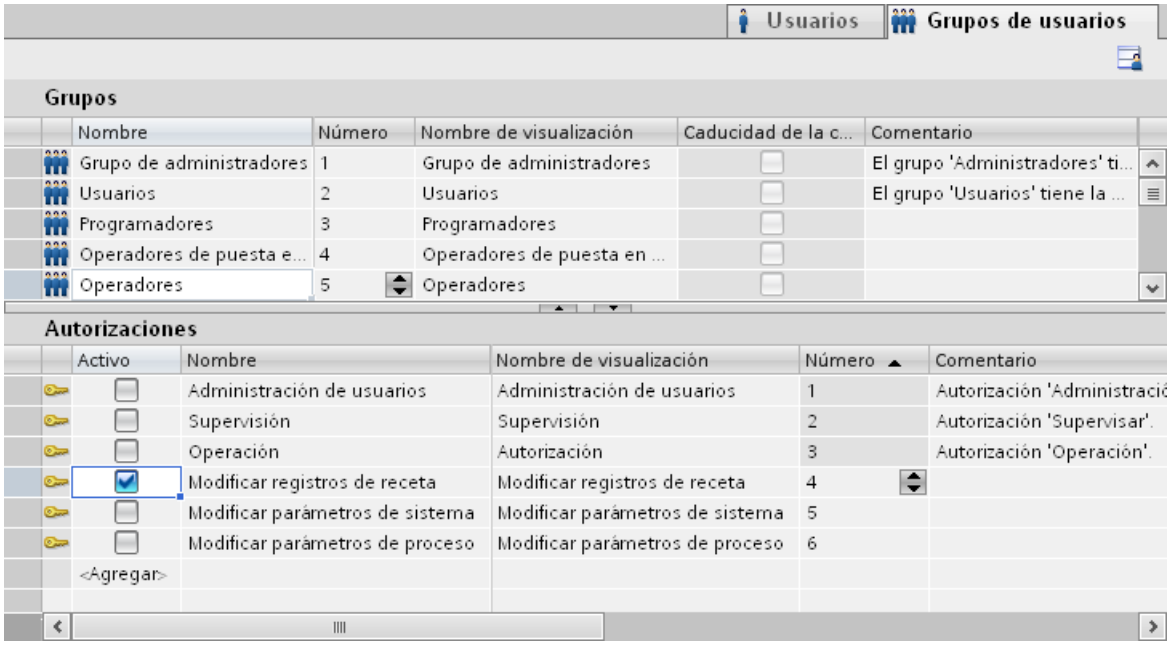
Resultado intermedio



Procedimiento

1. Haga clic en "Operadores" de la tabla "Grupos".
2. En la tabla "Autorizaciones" active la autorización "Modificar registros de receta".
3. Haga clic en "Operadores de puesta en marcha" de la tabla "Grupos".
4. En la tabla "Autorizaciones" active la autorización "Modificar registros de receta", "Modificar parámetros de sistema" y "Modificar parámetros de proceso".
5. Haga clic en "Programadores" de la tabla "Grupos".
6. En la tabla "Autorizaciones" active la autorización "Modificar registros de receta".

Resultado



Ejemplo: Crear usuarios y asignarlos a un grupo de usuarios

Tarea

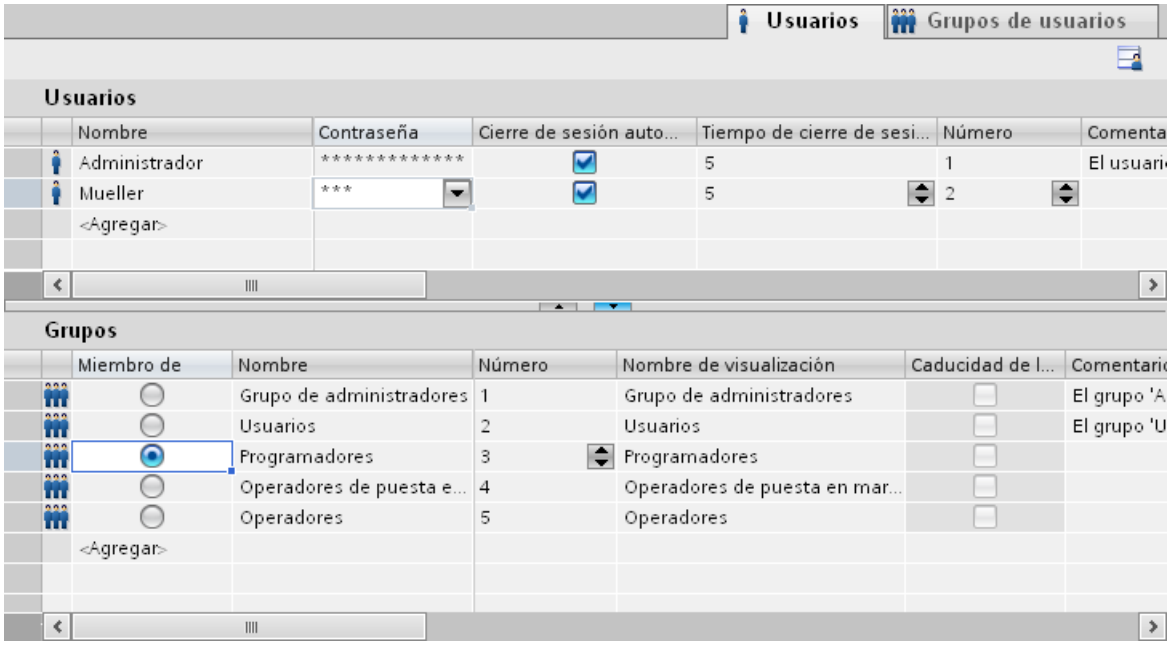
En el ejemplo siguiente se crean los usuarios y se asignan a los grupos de usuarios. El usuario se ordenará alfabéticamente justo después de entrar el nombre.

Procedimiento

1. Abra el área de trabajo "Usuarios".
2. Haga doble clic en "Agregar" en la tabla "Usuarios".
3. Introduzca el nombre de usuario "Mueller".

4. En la columna "Contraseña", haga clic en el botón . Se abre el cuadro de diálogo para introducir la contraseña.
5. Introduzca la contraseña "mueller".
6. Introduzca de nuevo la contraseña en el campo de confirmación situado debajo.
7. Cierre el cuadro de diálogo con el símbolo .
8. En la tabla "Grupos", active el grupo de usuarios "Programadores".

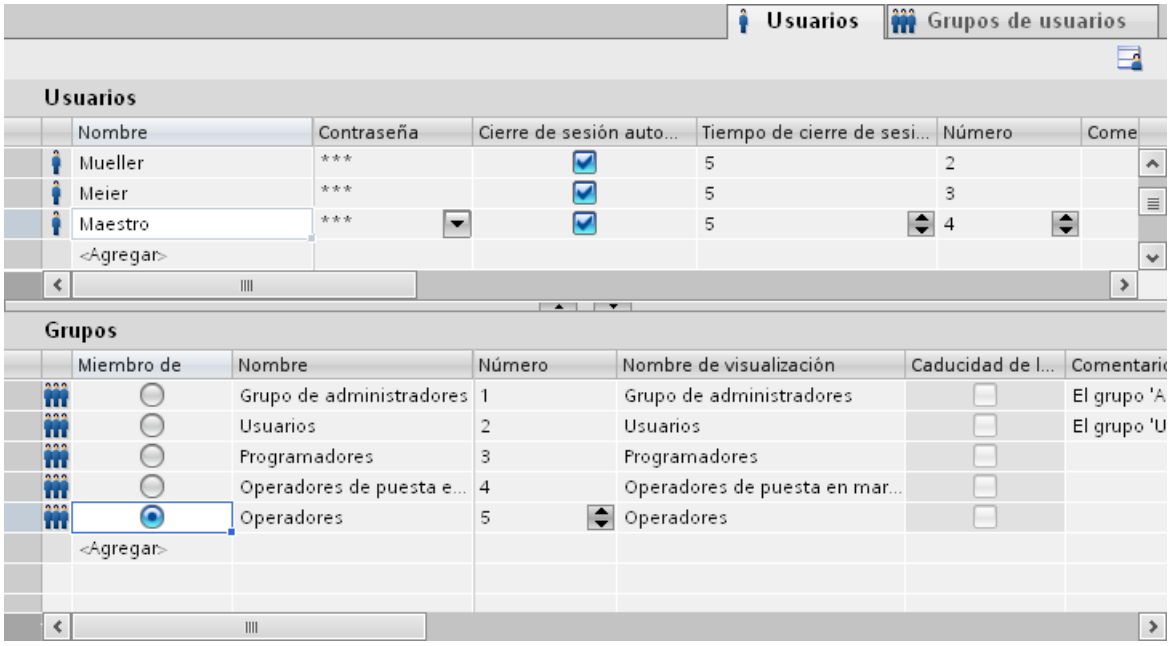
Resultado intermedio



Procedimiento

1. Haga doble clic en "Agregar" en la tabla "Usuarios".
2. Introduzca el nombre de usuario "Meier".
3. En la columna "Contraseña", haga clic en el botón . Se abre el cuadro de diálogo para introducir la contraseña.
4. Introduzca la contraseña "meier".
5. Introduzca de nuevo la contraseña en el campo de confirmación situado debajo.
6. Cierre el cuadro de diálogo con el símbolo .
7. En la tabla "Grupos", active el grupo de usuarios "Operadores de puesta en marcha".
8. Repita los pasos 2 a 6 para el usuario "Maestro".
9. En la tabla "Grupos", active el grupo de usuarios "Operadores".

Resultado



10.6 Trabajar con funciones de sistema

10.6.1 Principios básicos

10.6.1.1 Funciones de sistema

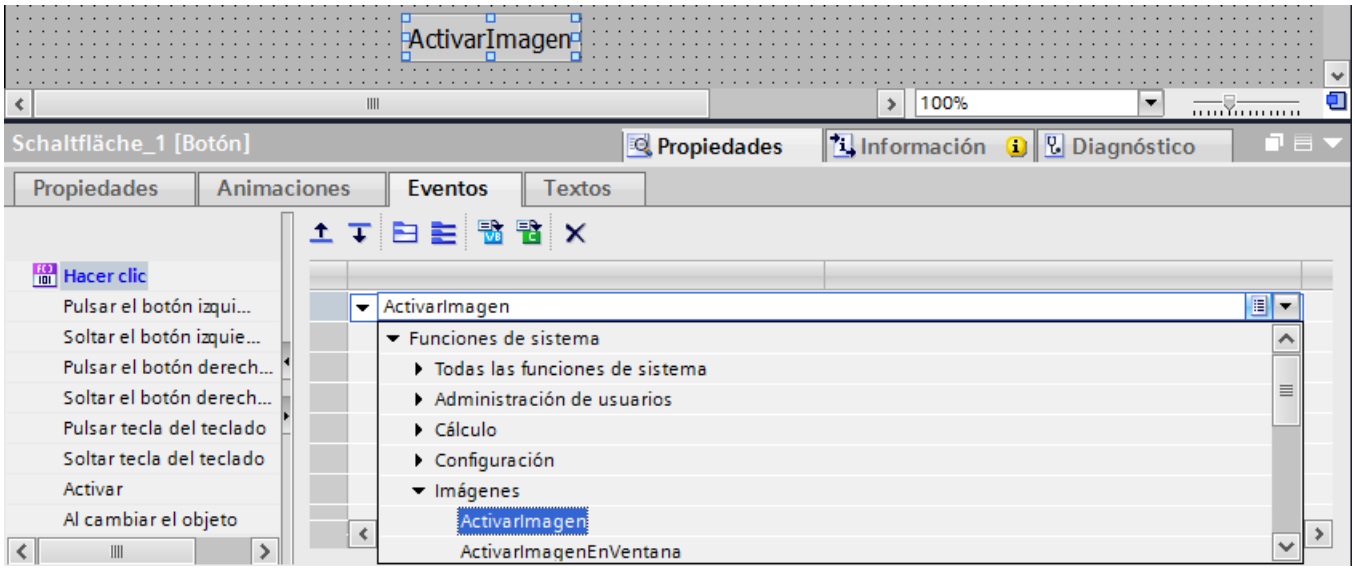
Introducción

Las funciones de sistema son funciones suministradas por WinCC. Están predefinidas y no se pueden modificar. Las funciones de sistema permiten realizar numerosas tareas en runtime, incluso sin tener conocimientos de programación, tales como:

- Cálculos (p. ej. aumentar el valor de una variable en una cantidad fija o variable)
- Funciones de archivación (p. ej. iniciar un fichero de valores de proceso)
- Ajustes (p. ej. cambiar de controlador o activar un bit en el controlador)
- Avisos (p. ej. tras cambiar de usuario)

Utilización

Las funciones de sistema se utilizan en una lista de funciones Cuando configure una lista de funciones, seleccione las funciones de sistema en una lista de selección ordenada por categorías:



En WinCC, cada función de sistema está asociada a un objeto y un evento. En cuanto aparece el evento, se lanza la función de sistema.

Dependencia del idioma

Los nombres de las funciones de sistema dependen del idioma de configuración ajustado. Gracias a ello, el ingeniero programador podrá reconocer las funciones de inmediato.

Disponibilidad

En una lista de funciones sólo podrá configurar las funciones de sistema que soporte el panel de operador en cuestión. Si desea utilizar un proyecto en varios paneles de operador, las funciones de sistema no soportadas por un panel en particular se identificarán en color.

Eventos

Los eventos que se pueden definir como disparador para la ejecución de una función de sistema dependen del objeto en cuestión y de la función seleccionada.

Así, por ejemplo, los eventos "Cambio de valor", "Rebase por defecto del límite inferior" y "Rebase por exceso del límite superior" pertenecen al objeto "Variable". Los eventos "Aparecido" y "Desaparecido" pertenecen al objeto "Imagen".

10.6.1.2 Utilizar funciones de sistema

Introducción

Cuando ocurre un evento configurado, la lista de funciones se procesa en runtime. El operador dispara un evento p. ej. pulsando una tecla de función en el panel de operador. El sistema también puede disparar eventos, p. ej. cuando un valor de proceso rebasa por defecto un valor límite.

Posibilidades de aplicación

Las funciones de sistema se pueden configurar en todos los objetos que reaccionen a eventos. Además, se pueden utilizar directamente en listas de funciones, lo que permite controlar el proceso.

- Lista de funciones
  En una lista de funciones se procesan las funciones de sistema fila por fila. Para evitar tiempos de espera, las funciones de sistema de ejecución prolongada se procesan de manera paralela en WinCC Runtime. Gracias a ello, la función de sistema siguiente se puede comenzar a ejecutar aunque no haya finalizado todavía la función anterior.

Encontrará un ejemplo de cómo configurar una lista de funciones en "Cambiar el modo de operación del panel de operador y visualizar el estado actual".

10.6.2 Trabajar con listas de funciones

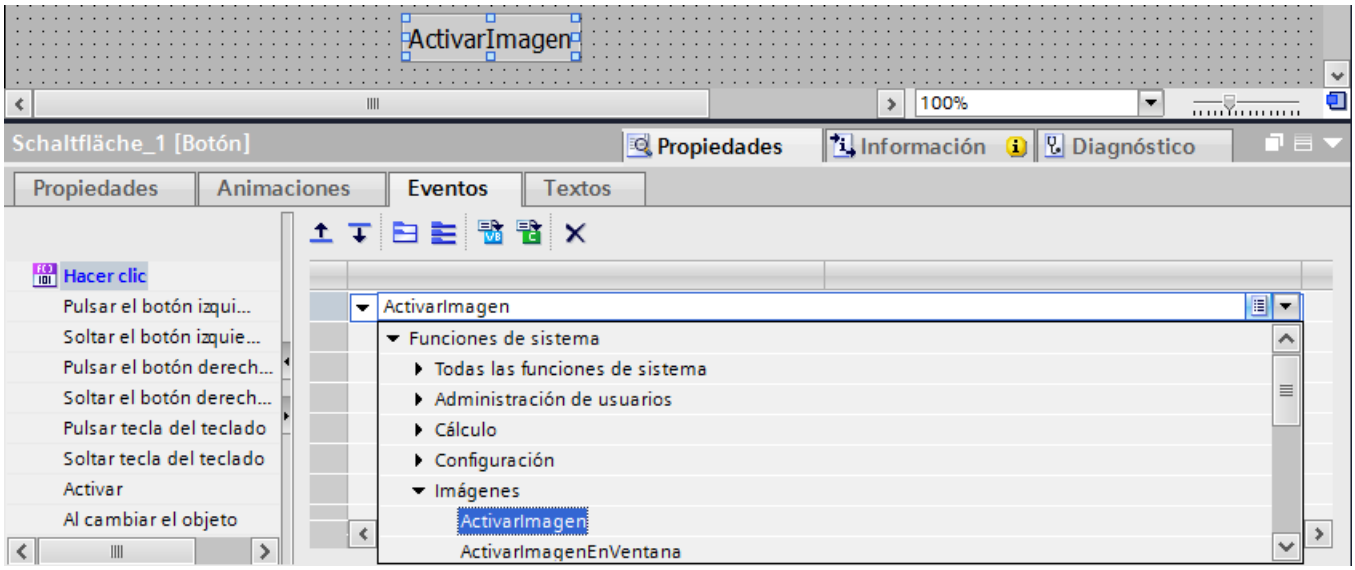
10.6.2.1 Principios básicos de la lista de funciones

Introducción

Una lista de funciones permite ejecutar varias funciones de sistema cuando aparece el evento configurado.

Principio

La lista de funciones se configura en el evento de un objeto (p. ej. un objeto de imagen o una variable). Los eventos disponibles dependen del objeto seleccionado y del panel de operador utilizado.



Los eventos sólo ocurren cuando el proyecto está en runtime. Los eventos son, por ejemplo:

- Cambiar el valor de una variable
- Pulsar un botón
- Activar runtime

En cada evento se puede configurar exactamente una lista de funciones.

Nota

Las funciones de sistema configurables que se pueden seleccionar en una lista de funciones dependen del panel de operador utilizado.

10.6.2.2 Propiedades de la lista de funciones

Información de estado

Durante la configuración, los datos del proyecto se comprueban en segundo plano.

Las siguientes causas provocan que la lista de funciones no se ejecute en runtime y se marquen en rojo las entradas erróneas:

- Hay al menos una función de sistema que no se ha parametrizado por completo.
- Hay por lo menos una función de sistema no soportada por el panel de operador seleccionado (p. ej. por un cambio del tipo de panel).

Procesamiento de funciones de sistema

Las funciones de sistema contenidas en una lista de funciones se procesan en runtime de forma secuencial, es decir, de arriba a abajo. Para evitar tiempos de espera, las funciones de sistema de ejecución prolongada (p. ej. operaciones con archivos) se procesan de manera paralela. Gracias a ello, la función de sistema siguiente se puede comenzar a ejecutar aunque no haya finalizado todavía la función anterior.

Para programar procesos no secuenciales y condicionados, utilice un script con bucles, instrucciones condicionadas y condiciones de interrupción.

Nota

Dependencia del panel

Los scripts de usuario no están disponibles en Basic Panels.

10.6.2.3 Configurar la lista de funciones

Introducción

Para configurar una lista de funciones, seleccione las funciones de sistema en una lista desplegable. Las funciones de sistema están ordenadas por categorías en la lista desplegable.

Requisitos

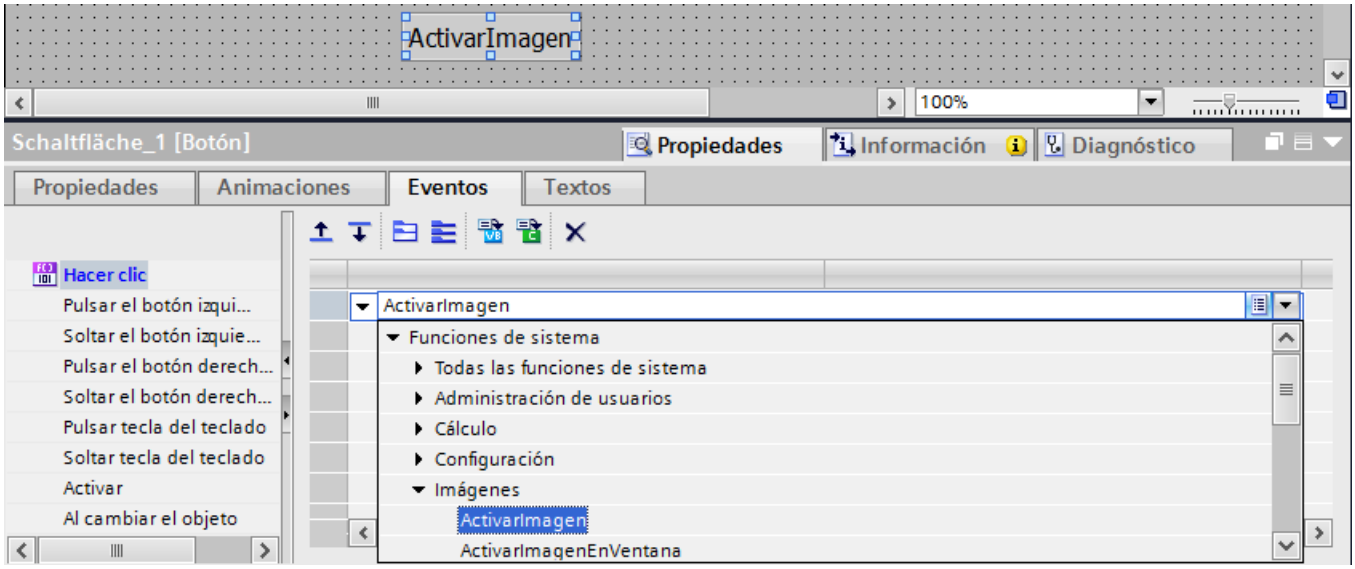
El objeto deberá tener por lo menos un evento configurable.

Procedimiento

Para configurar una lista de funciones, proceda del siguiente modo:

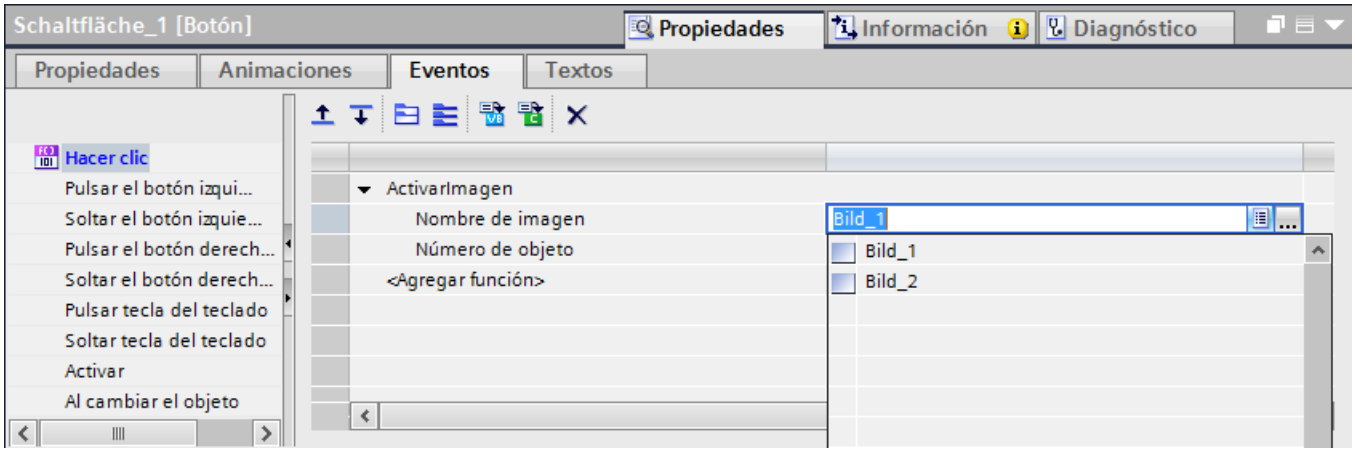
1. En WinCC, abra el editor que contiene el objeto.
2. Seleccione el objeto.
3. En la ventana de inspección haga clic en "Propiedades > Eventos". Seleccione el evento en el que desea configurar la lista de funciones.
4. En la lista desplegable de la ventana de inspección, seleccione la entrada "<Ninguna función>".
5. Seleccione la función de sistema deseada.

6. También puede introducir el nombre de la función de sistema.



La función de sistema se introducirá en la lista de funciones.

7. Si se ha parametrizado la función de sistema, seleccione los valores correspondientes para los parámetros.



8. Si desea agregar más funciones de sistema o scripts a la lista de funciones, repita los pasos 4 a 7.

Resultado

La lista de funciones deberá estar configurada. El estado de la lista de funciones se visualizará en la ventana de inspección, junto al evento configurado. Cuando el evento configurado ocurra en runtime, la lista de funciones se procesará de arriba a abajo.

10.6.2.4 Editar la lista de funciones

Introducción

La lista de funciones se puede editar como se indica a continuación:

- Modificar el orden de procesamiento de las funciones de sistema
- Eliminar una función de sistema

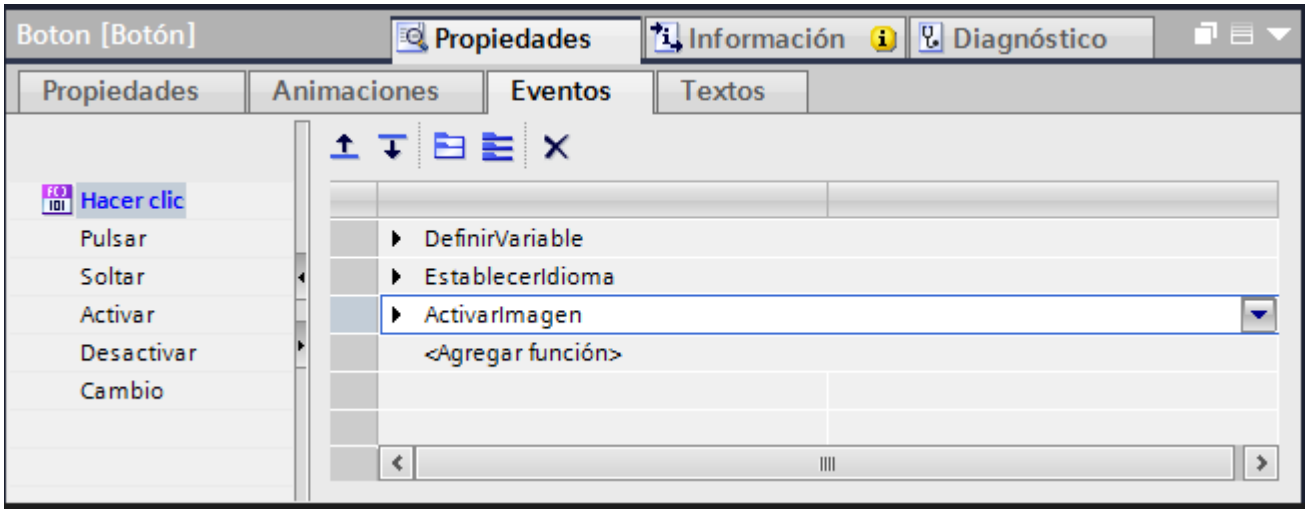
Encontrará más información en "Configurar la lista de funciones".

Requisitos

La lista de funciones deberá estar configurada.

Cambiar el orden de una función de sistema

1. Seleccione la función de sistema deseada en la lista de funciones.
2. A continuación, haga clic en la correspondiente flecha de sentido de la ventana de inspección hasta que la función de sistema o el script de usuario aparezca en la posición deseada.



Cambiar el orden de varias funciones de sistema

1. Mantenga pulsada la tecla <Mayús>.
2. Haga clic con el ratón consecutivamente en las funciones de sistema deseadas.
3. Desplace la selección a la posición deseada con Drag&Drop.

Eliminar una función de sistema

1. Seleccione la función deseada en la lista de funciones.
2. En el menú contextual, elija el comando "Borrar".

10.6.2.5 Procesar la lista de funciones en runtime

Principio

Una lista de funciones se procesa en runtime de arriba a abajo. Para evitar tiempos de espera durante el procesamiento, se distingue entre los tipos de procesamiento síncrono y asíncrono. El sistema efectúa esta distinción, evaluando para ello los diversos tiempos de ejecución de las funciones de sistema. Independientemente del tiempo de ejecución, los scripts de usuario se procesan siempre de forma síncrona. El procesamiento de la lista de funciones se interrumpirá si una función de sistema retorna un estado de error.

Procesamiento síncrono

En el procesamiento síncrono, las funciones de sistema contenidas en una lista de funciones se procesan de forma sucesiva. Para que una función de sistema se ejecute, la función anterior debe haber finalizado.

Procesamiento asíncrono

El tiempo de ejecución de las funciones de sistema que efectúan operaciones con archivos (tales como guardar o leer) es más largo que el de las funciones de sistema que p. ej. activan un valor de variable.

Por tanto, las funciones de sistema de ejecución prolongada se procesan de forma asíncrona. Por ejemplo, mientras una función de sistema escribe un registro de receta en un soporte de datos, se ejecuta ya la siguiente función. Gracias al procesamiento paralelo de las funciones de sistema se evitan tiempos de espera en el panel de operador.

10.6.3 Ejemplo

10.6.3.1 Cambiar el modo de operación del panel de operador y visualizar el estado actual

Tarea

En el presente ejemplo se utiliza la función de sistema "EstablecerModoDeOperaciónPanel" para cambiar entre los modos de operación "Online" y "Offline" en el panel de operador. El modo de operación ajustado actualmente se debe visualizar adicionalmente en el panel de operador.

Requisitos

Se ha creado una imagen de proceso.

Configuración

Para este ejemplo se necesitan una variable HMI y una lista de texto configuradas como se indica a continuación:

Variable HMI:

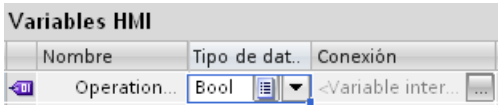
Nombre	Conexión al controlador	Tipo
OperatingMode	no	Bool

Lista de texto:

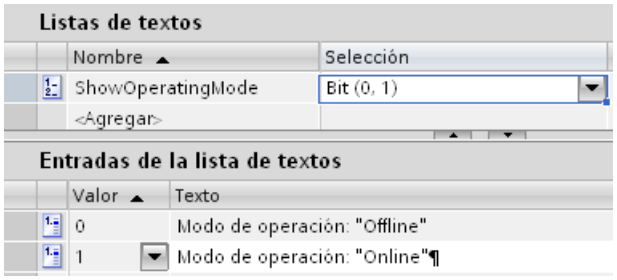
Nombre	Contiene	Valores
ShowOperatingMode	Bit (0/1)	1: Modo de operación: "Online": 0: Modo de operación: "Offline"

Procedimiento

1. Cree la variable HMI "OperatingMode" indicada arriba.

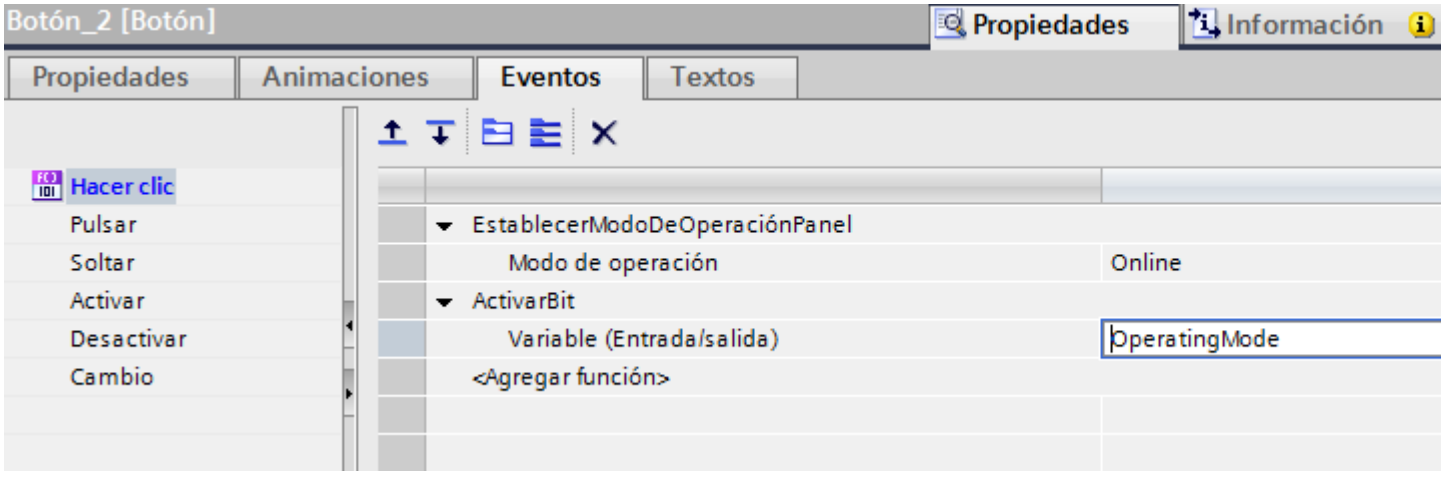


2. Cree la lista de textos "ShowOperatingMode" indicada arriba.

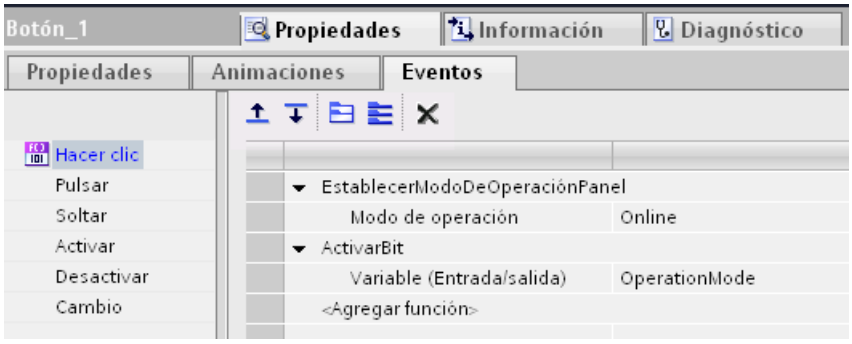


3. Abra la imagen e inserte un botón en el que se configurará el cambio de modo de operación a "Online".
4. Haga clic en la ventana de inspección "Propiedades > Eventos". Seleccione el evento "Pulsar".
5. Configure la función de sistema "EstablecerModoDeOperaciónPanel" asociada al evento "Pulsar". Esta función de sistema se encuentra en la lista de selección bajo "Configuración".
6. Seleccione la entrada "Online" para el parámetro "Modo de operación".
7. Configure la función de sistema "ActivarBit" asociada al evento "Pulsar". Esta función de sistema se encuentra en la lista de selección bajo "Procesamiento de bits".

8. En la lista, seleccione la variable HMI "OperatingMode" para el parámetro "Variable".



9. En el sinóptico del proceso, inserte un botón en el que se configurará el cambio de modo de operación a "Offline".
10. Repita los pasos 4 a 7. Seleccione la entrada "Offline" para el parámetro "Modo de operación". Configure la función de sistema "DesactivarBit" en vez de la función "ActivarBit".



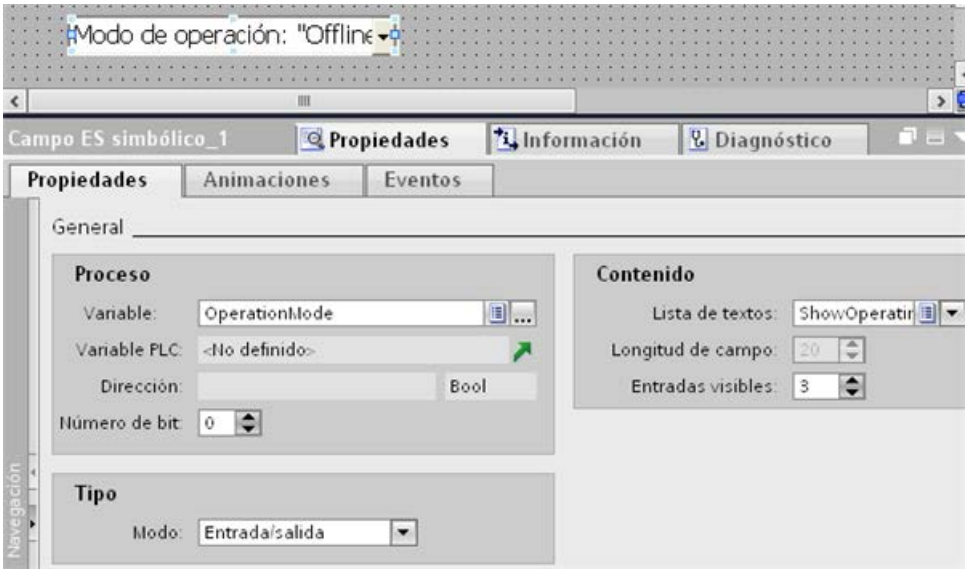
Resultado intermedio

El modo de operación del panel de operador se podrá cambiar en runtime mediante ambos botones.

El modo de operación ajustado actualmente se debe visualizar en un campo de salida en el panel de operador.

Procedimiento

1. Cree un "Campo E/S simbólico" en el sinóptico del proceso. En la ventana de inspección haga clic en "Propiedades > Propiedades".
2. Realice los ajustes siguientes en el grupo "General":
   - Bajo "Modo", seleccione "Salida".
   - Seleccione la lista de texto "ShowOperatingMode".
   - Seleccione la variable "OperatingMode".



Resultado

Si cambia el modo de operación utilizando los botones, el modo ajustado actualmente se visualizará en el panel de operador.

10.6.4 Referencia

10.6.4.1 Lista de funciones

Funciones de sistema disponibles según el equipo

Disponibilidad de las funciones de sistema

Las siguientes tablas muestran las funciones de sistema y las funciones definidas por el usuario disponibles en los paneles de operador.

Sujeto a cambios técnicos.

Resumen

	KP300 Basic PN	KTP400 Basic PN	KTP 600 Basic DP	KTP 600 Basic PN	KTP 1000 Basic DP	KTP 1000 Basic PN	TP 1500 Basic PN
Funciones definidas por el usuario	no	no	no	no	no	no	no
CerrarSesión (Página 8649)	sí	sí	sí	sí	sí	sí	sí
ModificarContraste (Página 8650)	sí	sí	no	sí [1)]	sí	sí	sí
ActivarImagen (Página 8651)	sí	sí	sí	sí	sí	sí	sí
ActivarImagenConNúmero (Página 8652)	sí	sí	sí	sí	sí	sí	sí
ActivarPantallaDeLimpieza (Página 8653)	no	sí	sí	sí	sí	sí	sí
ActivarImagenAnterior (Página 8654)	sí	sí	sí	sí	sí	sí	sí
ActualizarVariable (Página 8655)	sí	sí	sí	sí	sí	sí	sí
IniciarSesión (Página 8655)	sí	sí	sí	sí	sí	sí	sí
EditarAviso (Página 8656)	sí	sí	sí	sí	sí	sí	no
ObjetoDeImagenCursorAbajo (Página 8657)	sí	sí	sí	sí	sí	sí	sí
ObjetoDeImagenCursorArriba (Página 8657)	sí	sí	sí	sí	sí	sí	sí
ObjetoDeImagenAvanzarPágina (Página 8658)	sí	sí	sí	sí	sí	sí	sí
ObjetoDeImagenRetrocederPágina (Página 8658)	sí	sí	sí	sí	sí	sí	sí
IncrementarValorEnfocado (Página 8659)	sí	sí	sí	sí	sí	sí	no
IncrementarVariable (Página 8660)	sí	sí	sí	sí	sí	sí	sí
IrAInicio (Página 8660)	sí	sí	sí	sí	sí	sí	no
IrAFin (Página 8661)	sí	sí	sí	sí	sí	sí	no
InvertirBit (Página 8662)	sí	sí	sí	sí	sí	sí	sí
InvertirBitEnVariable (Página 8663)	sí	sí	sí	sí	sí	sí	sí
InvertirEscalaLineal (Página 8664)	sí	sí	sí	sí	sí	sí	sí
CalibrarPantallaTáctil (Página 8665)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasAvancePágina (Página 8666)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasRetrocesoPágina (Página 8667)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasAmpliar (Página 8667)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasComprimir (Página 8668)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasReglaDerecha (Página 8669)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasReglaIzquierda (Página 8668)	sí	sí	sí	sí	sí	sí	sí

	KP300 Basic PN	KTP400 Basic PN	KTP 600 Basic DP	KTP 600 Basic PN	KTP 1000 Basic DP	KTP 1000 Basic PN	TP 1500 Basic PN
VisorDeCurvasActivarModoRegla (Página 8669)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasIniciarParar (Página 8670)	sí	sí	sí	sí	sí	sí	sí
VisorDeCurvasIrAlPrincipio (Página 8670)	sí	sí	sí	sí	sí	sí	sí
LeerNombreDeUsuario (Página 8671)	sí	sí	sí	sí	sí	sí	sí
LeerNúmeroDeGrupo (Página 8672)	sí	sí	sí	sí	sí	sí	sí
LeerContraseña (Página 8672)	sí	sí	sí	sí	sí	sí	sí
EscalaLineal (Página 8674)	sí	sí	sí	sí	sí	sí	sí
BorrarBúferDeAvisos (Página 8675)	sí	sí	sí	sí	sí	sí	sí
BorrarBúferDeAvisosProtool (Página 8676)	sí	sí	sí	sí	sí	sí	sí
VisorDeAvisosEditarAviso (Página 8677)	sí	sí	sí	sí	sí	sí	sí
VisorDeAvisosAcusarAviso (Página 8678)	sí	sí	sí	sí	sí	sí	sí
VisorDeAvisosMostrarTextoAyuda (Página 8678)	sí	sí	sí	sí	sí	sí	sí
AcusarAviso (Página 8679)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasRegistroNuevo (Página 8680)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasLeerRegistroDelControlador (Página 8680)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasBorrarRegistro (Página 8680)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasMenú (Página 8681)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasAbrir (Página 8681)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasEscribirRegistroEnControlador (Página 8682)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasGuardarRegistro (Página 8683)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasGuardarRegistroComo (Página 8683)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasCambiarNombreDeRegistro (Página 8683)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasMostrarTextoDeAyuda (Página 8684)	sí	sí	sí	sí	sí	sí	sí
VisorDeRecetasAtrás (Página 8684)	sí	sí	sí	sí	sí	sí	sí
DesactivarBit (Página 8685)	sí	sí	sí	sí	sí	sí	sí
DesactivarBitEnVariable (Página 8686)	sí	sí	sí	sí	sí	sí	sí
BotónPulsar (Página 8687)	sí	sí	sí	sí	sí	sí	no

	KP300 Basic PN	KTP400 Basic PN	KTP 600 Basic DP	KTP 600 Basic PN	KTP 1000 Basic DP	KTP 1000 Basic PN	TP 1500 Basic PN
BotónSoltar (Página 8688)	sí	sí	sí	sí	sí	sí	no
DesplazarYEnmascarar (Página 8689)	sí	sí	sí	sí	sí	sí	sí
AvanzarPágina (Página 8691)	sí	sí	sí	sí	sí	sí	no
RetrocederPágina (Página 8691)	sí	sí	sí	sí	sí	sí	no
EstablecerModoDeOperaciónPanel (Página 8692)	sí	sí	sí	sí	sí	sí	sí
ActivarBit (Página 8693)	sí	sí	sí	sí	sí	sí	sí
ActivarBitDeVariable (Página 8694)	sí	sí	sí	sí	sí	sí	sí
ActivarBitMientrasTeclaPulsada (Página 8695)	sí	sí	sí	sí	sí	sí	sí
EstablecerColorRetroiluminación (Página 8696)	sí	no	no	no	no	no	no
EstablecerIdioma (Página 8698)	sí	sí	sí	sí	sí	sí	sí
DefinirVariable (Página 8699)	sí	sí	sí	sí	sí	sí	sí
EstablecerModoDeConexión (Página 8701)	sí	sí	sí	sí	sí	sí	sí
SimularTeclaSistema (Página 8702)	sí	sí	sí	sí	sí	sí	no
SimularVariable (Página 8703)	sí	sí	sí	sí	sí	sí	sí
PararRuntime (Página 8704)	sí	sí	sí	sí	sí	sí	sí
IdentificarCambioDeUsuario (Página 8705)	sí	sí	sí	sí	sí	sí	sí
ReducirValorEnfocado (Página 8706)	sí	sí	sí	sí	sí	sí	no
ReducirVariable (Página 8706)	sí	sí	sí	sí	sí	sí	sí
CambiarConexión (Página 8707)	sí	sí	sí	sí	sí	sí	sí
MostrarDiálogoInicioSesión (Página 8709)	sí	sí	sí	sí	sí	sí	sí
MostrarTextoDeAyuda (Página 8709)	sí	sí	sí	sí	sí	sí	sí
MostrarVentanaDeAvisos (Página 8710)	sí	sí	sí	sí	sí	sí	sí

[1] Sólo para KTP600 Basic mono PN

Funciones de sistema

CerrarSesión

Descripción

Cierra la sesión del usuario actual en el panel de operador.

Utilización en la lista de funciones

CerrarSesión

Utilización en funciones definidas por el usuario

Logoff

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

IniciarSesión (Página 8655)

ModificarContraste

Descripción

Cambia el contraste de la pantalla del panel de operador al siguiente nivel de contraste.

Utilización en la lista de funciones

ModificarContraste (Cambio)

Utilización en scripts de usuario

-

Parámetros

Cambio

Determina cómo se debe modificar el contraste:

0 (hmiDecrease) = Reducir: Reduce el contraste un nivel.

1 (hmiIncrease) = Aumentar: Aumenta el contraste un nivel.

Ejemplo de aplicación

Objetivo

Desea aumentar y reducir el contraste de la pantalla utilizando para un botón de comando, respectivamente.

Indicaciones relativas a la configuración

Configure dos botones y asigne en cada uno de ellos el evento "Pulsar" a la función de sistema "ModificarContraste". Asigne a dichos botones los parámetros "Aumentar" y "Reducir", respectivamente.

Proceso en el panel de operador

Si pulsa uno de estos botones en runtime, el contraste se aumentará o reducirá un nivel.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

CerrarSesión (Página 8647)

ActivarImagen

Descripción

Efectúa un cambio de imagen para visualizar la imagen indicada.

Para conmutar entre la imagen raíz y la ventana permanente, y viceversa, utilice la función de sistema "ActivarImagenConNúmero".

Utilización en la lista de funciones

ActivarImagen (Nombre de imagen, Número del objeto)

Utilización en funciones definidas por el usuario

ActivateScreen (Screen_name, Object_number)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Nombre de imagen

Nombre de la imagen a la que se cambia.

Número del objeto

Elemento de manejo que se resalta en la imagen indicada después del cambio de imagen. El número del elemento de manejo se define mediante el orden de tabulación durante la configuración.

Si indica "0":

- Si al ejecutarse la función de sistema está resaltada la ventana permanente, ésta permanecerá destacada.
- En cambio, si está resaltada la imagen raíz, se destacará el primer elemento de manejo de la imagen indicada.

Nota

Si la función de sistema "ActivarImagen" se asocia al evento "Margen alcanzado", sólo se admite el valor 0 para el parámetro "Número del objeto". El objeto activo no se establece conforme al número de objeto, sino según la coordenada X del objeto activo antes del cambio de imagen.

Ejemplo

El siguiente código de programa activa la imagen "Screen_2" con la función ActivateScreen al pulsar cualquier tecla.

```
{

// User defined code
// i.e. when pressing a button
ActivateScreen ("Screen_2", 0);
...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

ActivarImagenConNúmero (Página 8652)

ActivarImagenConNúmero

Descripción

Dependiendo del valor de una variable, cambia a una imagen determinada.

La imagen se identifica por su número de imagen.

Utilización en la lista de funciones

ActivarImagenConNúmero (Número de imagen, Número del objeto)

Utilización en funciones definidas por el usuario

ActivateScreenByNumber (Screen_number, Object_number)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Número de imagen

Variable que contiene el número de imagen a la que se cambia.

Si desea conmutar entre la imagen raíz y la ventana permanente, y viceversa, introduzca "0" o "-1", respectivamente.

0 = Cambia de la imagen raíz a la ventana permanente.

-1 = Cambia de la ventana permanente a la imagen raíz.

Número del objeto

Número del objeto de imagen que se resalta en la imagen indicada después del cambio de imagen. El número del elemento de manejo se define mediante el orden de tabulación durante la configuración.

Si indica "0":

- Si al ejecutarse la función de sistema está resaltada la ventana permanente, ésta permanecerá destacada.
- En cambio, si está resaltada la imagen raíz, se destacará el primer elemento de manejo de la imagen indicada.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

ActivarImagen (Página 8649)

ActivarPantallaDeLimpieza

Descripción

Activa la pantalla de limpieza en el panel de operador. La pantalla del panel de operador se desactiva por el periodo indicado.

Mientras está desactivada la pantalla del panel de operador, podrá limpiarla sin activar accidentalmente funciones de la pantalla táctil.

Utilización en la lista de funciones

ActivarPantallaDeLimpieza (Período)

Utilización en scripts de usuario

--

Parámetros

Período

Período durante el que está desactivada la pantalla. El tiempo restante se indica mediante una barra de progreso.

Rango de valores en segundos (de 10 a 300).

Nota

La función de sistema ActivarPantallaDeLimpieza no puede ser simulada.

Consulte también

CerrarSesión (Página 8647)

ActivarImagenAnterior

Descripción

Cambia a la imagen que estaba activada antes de cambiar a la imagen actual. El cambio de imagen no se ejecuta si antes no había ninguna imagen activa.

Se guardan las últimas 10 imágenes activadas. Si cambia a una imagen que ya no está guardada, aparecerá un aviso del sistema.

Nota

Si desea utilizar la función de sistema, la imagen a la que desea cambiar deberá estar contenida en la estructura de navegación.

Utilización en la lista de funciones

ActivarImagenAnterior

Utilización en funciones definidas por el usuario

ActivatePreviousScreen

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

ActualizarVariable

Descripción

Lee del controlador el valor actual de la variable con la ID de actualización indicada.

Utilización en la lista de funciones

ActualizarVariable (ID de actualización)

Utilización en scripts de usuario

-

Parámetros

ID de actualización

ID de actualización asignada a la variable que se desea actualizar.

Consulte también

CerrarSesión (Página 8647)

IniciarSesión

Descripción

Inicia la sesión del usuario actual en el panel de operador.

Utilización en la lista de funciones

IniciarSesión (Contraseña, Nombre de usuario)

Utilización en funciones definidas por el usuario

Logon (Password, User_name)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Contraseña

La variable de la cual se lee la contraseña del usuario que va a iniciar la sesión.

Si el usuario ha iniciado la sesión, se borra la contraseña de la variable.

Nombre de usuario

La variable de la cual se lee el Nombre de usuario que va a iniciar la sesión.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

CerrarSesión (Página 8647)

EditarAviso

Descripción

Lanza el evento "Edición" para todos los avisos seleccionados.

Si no ha acusado todavía los avisos a editar, el acuse se efectuará automáticamente al ejecutarse esta función de sistema.

Esta función de sistema sólo se puede utilizar para teclas de función.

Utilización en la lista de funciones

EditarAviso

Utilización en funciones definidas por el usuario

EditAlarm

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

ObjetoDeImagenCursorArriba

Descripción

Hace que el cursor se desplace por líneas hacia arriba en el objeto de imagen indicado.

Esta función de sistema se puede utilizar para los objetos de imagen indicados a continuación:

- Visor de usuarios
- Visor de avisos
- Visor de recetas

Utilización en la lista de funciones

ObjetoDeImagenCursorArriba (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del objeto de imagen en el que se ejecuta la función de la tecla.

ObjetoDeImagenCursorAbajo

Descripción

Hace que el cursor se desplace por líneas hacia abajo en el objeto de imagen indicado.

Esta función de sistema se puede utilizar para los objetos de imagen indicados a continuación:

- Visor de usuarios
- Visor de avisos
- Visor de recetas

Utilización en la lista de funciones

ObjetoDeImagenCursorAbajo (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del objeto de imagen en el que se ejecuta la función de la tecla.

ObjetoDeImagenRetrocederPágina

Descripción

Hace que el cursor se desplace por páginas hacia arriba en el objeto de imagen indicado.

Esta función de sistema se puede utilizar para los objetos de imagen indicados a continuación:

- Visor de usuarios
- Visor de avisos
- Visor de recetas

Utilización en la lista de funciones

ObjetoDeImagenRetrocederPágina (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del objeto de imagen en el que se ejecuta la función de la tecla.

ObjetoDeImagenAvanzarPágina

Descripción

Hace que el cursor se desplace por páginas hacia abajo en el objeto de imagen indicado.

Esta función de sistema se puede utilizar para los objetos de imagen indicados a continuación:

- Visor de usuarios
- Visor de avisos
- Visor de recetas

Utilización en la lista de funciones

ObjetoDeImagenAvanzarPágina (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del objeto de imagen en el que se ejecuta la función de la tecla.

IncrementarValorEnfocado

Descripción

Suma el valor indicado al valor de la variable asignada al campo de entrada (campo de selección simbólico, campo de selección gráfico, deslizador) resaltado actualmente.

Esta función de sistema sólo se puede utilizar en teclas de función.

Utilización en la lista de funciones

IncrementarValorEnfocado (Valor)

Utilización en scripts de usuario

-

Parámetros

Valor

Valor a sumar al valor de la variable.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

IncrementarVariable

Descripción

Suma el valor indicado al valor de la variable.

X = X + a

Nota

La función de sistema utiliza la misma variable como valor de entrada y como valor de salida. Si utiliza esta función de sistema para convertir un valor, deberá utilizar una variable auxiliar. A la variable auxiliar se le puede asignar un valor con la función de sistema "DefinirVariable".

Si asocia la función de sistema a eventos de un aviso y la variable no se utiliza en la imagen actual, no se garantiza que el valor real de la variable se utilice en el controlador. Ello se puede mejorar activando el modo de adquisición "Cíclico continuo".

Utilización en la lista de funciones

IncrementarVariable (Variable, Valor)

Utilización en funciones definidas por el usuario

IncreaseTag (Tag, Value)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable a la que se debe agregar el valor indicado.

Valor

Valor que se suma.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

DefinirVariable (Página 8699)

IrAInicio

Descripción

Ejecuta la función de tecla <Inicio> en el panel de operador.

Utilice esta función de sistema si la versión estándar del panel de operador no dispone de esta funcionalidad.

Esta función de sistema sólo se puede utilizar para teclas de función.

Utilización en la lista de funciones

IrAInicio

Utilización en funciones definidas por el usuario

GoToHome

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

IrAFin

Descripción

Ejecuta la función de tecla <Fin> en el panel de operador.

Utilice esta función de sistema si la versión estándar del panel de operador no dispone de esta funcionalidad.

Esta función de sistema sólo se puede utilizar para teclas de función.

Utilización en la lista de funciones

IrAFin

Utilización en funciones definidas por el usuario

GoToEnd

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

InvertirBit

Descripción

Invierte el valor de la variable indicada del tipo "Bool".

- Si la variable tiene el valor 1 (TRUE), se ajusta a 0 (FALSE).
- Si la variable tiene el valor 0 (FALSE), se ajusta a 1 (TRUE).

Utilización en la lista de funciones

InvertirBit (Variable)

Utilización en funciones definidas por el usuario

InvertBit (Tag)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable cuyo bit se debe ajustar.

Ejemplo

El siguiente código de programa invierte el valor de la variable booleana b_value y muestra el resultado junto con el valor original b_saved.

```
{
BOOL b_value = 0;
BOOL b_saved = b_value;

//Invert variable
invertBit(b_value);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",b_value, b_saved);
  ...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

InvertirBitEnVariable (Página 8663)

InvertirBitEnVariable

Descripción

Invierte un bit en la variable indicada:

- Si el bit de la variable tiene el valor 1 (TRUE), se ajusta a 0 (FALSE).
- Si el bit de la variable tiene el valor 0 (FALSE), se ajusta a 1 (TRUE).

Tras modificar el bit indicado, la función de sistema vuelve a transferir toda la variable al controlador. No se comprueba si entretanto se han modificado otros bits en la variable. El operador y el controlador sólo podrán acceder a la variable indicada en modo de lectura hasta que la variable vuelva a transferirse al controlador.

Nota

No utilice esta función de sistema si el controlador admite variables BOOL. En su lugar utilice la función de sistema "InvertirBit".

Utilización en la lista de funciones

InvertirBitEnVariable (Variable, Bit)

Utilización en funciones definidas por el usuario

InvertBitInTag (Tag, Bit)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Tag

Variable en la que se debe ajustar el bit indicado.

Bit

Número del bit a ajustar.

Si utiliza esta función de sistema en un script de usuario, los bits de una variable se cuentan de derecha a izquierda. El contaje comienza a partir de "0".

Ejemplo

El siguiente código de programa invierte un bit en la posición indicada bitposition de la variable bvalue y muestra el resultado junto con el valor original bsaved.

```
{
BYTE bvalue;
BYTE bsaved = bvalue;
BYTE bitposition = 2;

//Invert bit in bitposition
InvertBitInTag (bvalue, bitposition);
//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bvalue, bsaved);
  ...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

InvertirBit (Página 8660)

InvertirEscalaLineal

Descripción

Asigna a la variable X un valor que se calcula a partir del valor de la variable Y indicada mediante la función lineal X = (Y - b) / a.

Las variables X e Y no pueden ser idénticas. Esta función de sistema es la función inversa de "EscalaLineal".

Nota

Las variables X e Y no pueden ser idénticas. Si desea convertir una variable en Sí misma, debe utilizar una variable auxiliar

Con la función de sistema "DefinirVariable" se puede asignar a la variable auxiliar el valor de la variable que se va convertir.

Utilización en la lista de funciones

InvertirEscalaLineal (X, Y, b, a)

Utilización en funciones definidas por el usuario

InverseLinearScaling (X, Y, b, a)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

X

Variable a la que se asigna el valor calculado en la ecuación lineal.

Y

Variable cuyo valor se utiliza para el cálculo.

b

Valor que se va a restar.

a

Valor por el que se divide.

Ejemplo

El siguiente código de programa asigna un valor a la variable varX con la función InverseLinearScaling.

```
{
BYTE varX;
BYTE Yvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

//Inverse linear scaling
 InverseLinearScaling (varX, Yvalue, bvalue, avalue);

printf ("varX = %d\r\n, varX);
...
}
```

El valor de retorno guardado puede procesarse en el código subsiguiente .

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

EscalaLineal (Página 8674)

CalibrarPantallaTáctil

Descripción

Llama un programa para calibrar la pantalla táctil.

Durante la calibración se pedirá al operador que toque cinco posiciones en la pantalla táctil. Para confirmar el proceso de calibración, toque la pantalla táctil en un intervalo de 30 segundos. Si no toca la pantalla táctil dentro de ese intervalo, los ajustes de calibrado se rechazan. La guía del usuario está en inglés.

Utilice esta función de sistema al poner en marcha el panel de operador por primera vez.

Utilización en la lista de funciones

CalibrarPantallaTáctil

Utilización en funciones definidas por el usuario

CalibrateTouchScreen

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Nota

La función de sistema CalibrarPantallaTáctil no puede ser simulada.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

VisorDeCurvasAvancePágina

Descripción

Avanza un ancho de visualización en el visor de curvas indicado.

Utilización en la lista de funciones

VisorDeCurvasAvancePágina (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que se debe avanzar un ancho de visualización.

VisorDeCurvasRetrocesoPágina

Descripción

Retrocede un ancho de visualización en el visor de curvas indicado.

Utilización en la lista de funciones

VisorDeCurvasRetrocesoPágina (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que se debe retroceso un ancho de visualización.

VisorDeCurvasAmpliar

Descripción

Reduce el intervalo de tiempo que se visualiza en el visor de curvas indicado.

Utilización en la lista de funciones

VisorDeCurvasAmpliar (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que se debe reducir el intervalo de tiempo visualizado.

VisorDeCurvasComprimir

Descripción

Aumenta el intervalo de tiempo que se visualiza en el visor de curvas indicado.

Utilización en la lista de funciones

VisorDeCurvasComprimir (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que se debe aumentar el intervalo de tiempo visualizado.

VisorDeCurvasReglaIzquierda

Descripción

Desplaza la regla hacia atrás (a la izquierda) en el visor de curvas indicado.

Nota

Para poder desplazar la regla, ésta debe estar activada. Para activar la regla utilice la función de sistema "VisorDeCurvasActivarModoRegla".

Utilización en la lista de funciones

VisorDeCurvasReglaIzquierda (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que la regla se debe desplazar hacia atrás.

Consulte también

VisorDeCurvasActivarModoRegla (Página 8669)

VisorDeCurvasReglaDerecha

Descripción

Desplaza la regla hacia adelante (a la derecha) en el visor de curvas indicado.

Nota

Para poder desplazar la regla, ésta debe estar activada. Para activar la regla utilice la función de sistema "VisorDeCurvasActivarModoRegla".

Utilización en la lista de funciones

VisorDeCurvasReglaDerecha (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que la regla se debe desplazar hacia adelante.

Consulte también

VisorDeCurvasActivarModoRegla (Página 8669)

VisorDeCurvasActivarModoRegla

Descripción

Muestra u oculta la regla en el visor de curvas indicado. La regla muestra el valor Y correspondiente a un valor X.

Nota

Para visualizar la regla es necesario activar el ajuste "Mostrar regla" en las propiedades del visor de curvas.

Utilización en la lista de funciones

VisorDeCurvasActivarModoRegla (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que la regla se debe mostrar u ocultar.

Consulte también

VisorDeCurvasReglaIzquierda (Página 8666)
VisorDeCurvasReglaDerecha (Página 8667)

VisorDeCurvasIniciarParar

Descripción

Detiene o prosigue el registro de curvas en el visor de curvas indicado.

Utilización en la lista de funciones

VisorDeCurvasIniciarParar (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que se debe iniciar o parar el registro de curvas.

VisorDeCurvasIrAlPrincipio

Descripción

Retrocede al principio del área de representación en el visor de curvas.

Utilización en la lista de funciones

VisorDeCurvasIrAlPrincipio (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de curvas en el que se retrocede al principio del área de representación.

LeerNombreDeUsuario

Descripción

Escribe el nombre del usuario conectado actualmente al panel de operador en la variable indicada.

Si la variable indicada contiene una conexión con el controlador, el nombre del usuario también estará disponible en el controlador. Con esta función de sistema se pueden p. ej. liberar determinadas funciones para usuarios específicos.

Utilización en la lista de funciones

LeerNombreDeUsuario (Variable)

Utilización en funciones definidas por el usuario

GetUserName (Tag)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable en la que se escribe el nombre del usuario.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

LeerNúmeroDeGrupo

Descripción

Lee el número del grupo al que pertenece el usuario registrado en el panel de operador y lo escribe en la variable indicada.

Utilización en la lista de funciones

LeerNúmeroDeGrupo (Variable)

Utilización en funciones definidas por el usuario

GetGroupNumber (Tag)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable en la que se escribe el número del grupo.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

LeerContraseña

Descripción

Lee la contraseña del usuario conectado al panel de operador y la escribe en la variable indicada.

Nota

Asegúrese de que el valor de la variable indicada no aparece en ningún otro lugar del proyecto.

Nota

Las contraseñas de usuarios de SIMATIC Logon no se pueden leer.

Utilización en la lista de funciones

LeerContraseña (Variable)

Utilización en funciones definidas por el usuario

GetPassword (Tag)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable en la que se escribe la contraseña.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

LeerModoPLC

Descripción

Evalúa el estado actual del controlador conectado.

La función de sistema "LeerModoPLC" sólo se puede configurar para los siguientes controladores:

- SIMATIC S7 1200
- SIMATIC S7 1500

Utilización en la lista de funciones

LeerModoPLC (conexión, modo)

Utilización en funciones definidas por el usuario

GetPLCMode (Connection, Mode)

Parámetros

Conexión

Conexión de controlador y panel de operador.

Modo

Evalúa el estado del controlador conectado.
8: RUN = se procesa el programa de control.
4: STOP = se ha interrumpido el programa de control.
Seleccione una variable de usuario adecuada para la evaluación.

EscalaLineal

Descripción

Asigna a la variable Y un valor que se calcula a partir del valor de las variables X indicadas mediante la función lineal Y = (a * X) + b.

La función inversa es la función de sistema "InvertirEscalaLineal".

Nota

Las variables X e Y no pueden ser idénticas. Si desea convertir una variable en Sí misma, debe utilizar una variable auxiliar

Con la función de sistema "DefinirVariable" se puede asignar a la variable auxiliar el valor de la variable que se va convertir.

Utilización en la lista de funciones

EscalaLineal (Y, a, X, b)

Utilización en funciones definidas por el usuario

LinearScaling (Y, a, X, b)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Y

Variable a la que se asigna el valor calculado en la ecuación lineal.

a

Valor con el que se multiplica.

X

Variable cuyo valor se utiliza para el cálculo.

b

Valor que se suma.

Ejemplo

El siguiente código de programa asigna con la función LinearScaling un valor a la variable Yvar.

```
{
BYTE Yvar;
BYTE Xvalue = 10;
BYTE bvalue = 3;
BYTE avalue = 4;

// linear scaling
 LinearScaling ( Yvar, avalue, Xvalue, bvalue);

printf ("Yvar = %d\r\n, Yvar);
...
}
```

El valor de retorno guardado puede procesarse en el código subsiguiente.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

InvertirEscalaLineal (Página 8662)

BorrarBúferDeAvisos

Descripción

Borra avisos del búfer de avisos en el panel de operador.

Nota

Los avisos no acusados también se borran.

Utilización en la lista de funciones

BorrarBúferDeAvisos (Número de categoría)

Utilización en funciones definidas por el usuario

ClearAlarmBuffer (Alarm_class_number)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Número de categoría

Determina qué avisos se deben borrar del búfer de avisos:

0 (hmiAll) = Todos los avisos

1 (hmiAlarms) = Avisos de la categoría "Errors"

2 (hmiEvents) = Avisos de la categoría "Warnings"

3 (hmiSystem) = Avisos de la categoría "System"

4 (hmiS7Diagnosis) = Avisos de la categoría "Diagnosis Events"

Nota

Dependencia del dispositivo

Los avisos de la categoría "Diagnosis Events" no están disponibles en Basic Panels.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

BorrarBúferDeAvisosProtool

Descripción

Esta función de sistema existe por motivos de compatibilidad.

Tiene la misma funcionalidad que la función de sistema "BorrarBúferDeAvisos" pero utiliza la antigua numeración de ProTool.

Utilización en la lista de funciones

BorrarBúferDeAvisosProtool (Número de categoría)

Utilización en funciones definidas por el usuario

ClearAlarmBufferProtoolLegacy (Alarm_class_number)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Número de categoría

Número de la categoría cuyos avisos se borrarán:

-1 (hmiAllProtoolLegacy) = Todos los avisos

0 (hmiAlarmsProtoolLegacy) = Avisos de la categoría "Errors"

1 (hmiEventsProtoolLegacy) = Avisos de la categoría "Warnings"

2 (hmiSystemProtoolLegacy) = Avisos de la categoría "System"

3 (hmiS7DiagnosisProtoolLegacy) = Avisos de la categoría "Diagnosis Events"

Nota

Dependencia del dispositivo

Los avisos de la categoría "Diagnosis Events" no están disponibles en Basic Panels.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

BorrarBúferDeAvisos (Página 8673)

VisorDeAvisosEditarAviso

Descripción

Lanza el evento "Edición" para todos los avisos seleccionados en el visor de avisos indicado.

Utilice esta función de sistema si no se debe utilizar el botón integrado en el control ActiveX.

Para el evento "Edición" también se puede configurar una función de sistema. Por ejemplo, puede conmutar a la imagen del proceso en la que se ha presentado el aviso.

Nota

Si no ha acusado todavía los avisos a editar, el acuse se efectuará automáticamente al ejecutarse esta función de sistema.

Utilización en la lista de funciones

VisorDeAvisosEditarAviso (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de avisos en el que se debe lanzar el evento.

Nota

Los siguientes paneles de operador no soportan esta función de sistema para el objeto "Imagen": OP 73, OP 77A, TP 177A.

VisorDeAvisosAcusarAviso

Descripción

Acusa los avisos seleccionados en el visor de avisos indicado.

Utilice esta función de sistema si no se debe utilizar el botón integrado en el control ActiveX.

Utilización en la lista de funciones

VisorDeAvisosAcusarAviso (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de avisos en el que se debe lanzar el evento.

Nota

Los siguientes paneles de operador no soportan esta función de sistema para el objeto "Imagen": OP 73, OP 77A, TP 177A.

VisorDeAvisosMostrarTextoAyuda

Descripción

Muestra el tooltip configurado para el aviso seleccionado en el visor de avisos indicado.

Utilización en la lista de funciones

VisorDeAvisosMostrarTextoAyuda (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de avisos en el que se debe lanzar el evento.

Nota

Los siguientes paneles de operador no soportan esta función de sistema para el objeto "Imagen": OP 73, OP 77A, TP 177A.

AcusarAviso

Descripción

Acusa todos los avisos seleccionados.

Utilice esta función de sistema si el panel de operador no tiene ninguna tecla de acuse o si no debe emplearse la tecla integrada del visor de avisos.

Esta función de sistema sólo se puede utilizar para teclas de función.

Utilización en la lista de funciones

AcusarAviso

Utilización en funciones definidas por el usuario

AcknowledgeAlarm

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

--

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

VisorDeRecetasRegistroNuevo

Descripción

Crea un registro nuevo en el visor de recetas visualizado.

Utilización en la lista de funciones

VisorDeRecetasRegistroNuevo (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe crear el nuevo registro.

VisorDeRecetasLeerRegistroDelControlador

Descripción

Transfiere el registro cargado en el controlador al panel de operador y lo muestra en el visor de recetas.

Utilización en la lista de funciones

VisorDeRecetasLeerRegistroDelControlador (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se visualiza el registro transferido desde el controlador.

VisorDeRecetasBorrarRegistro

Descripción

Borra el registro que se visualiza en el visor de recetas.

Utilización en la lista de funciones

VisorDeRecetasBorrarRegistro (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe borrar el registro visualizado.

VisorDeRecetasMenú

Descripción

Abre el menú del visor de recetas simple indicado.

Utilice esta función de sistema sólo en un visor de recetas simple.

Utilización en la lista de funciones

VisorDeRecetasMenú (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe abrir el menú.

VisorDeRecetasAbrir

Descripción

Muestra los valores de registro en el visor de recetas indicado o cambia al siguiente campo de selección. Esta función de sistema no tiene ningún efecto si el campo de selección de los valores del registro de receta se muestra en el panel de operador.

Orden de manejo de los campos de selección en runtime:

- Nombre de receta
- Nombre de registro
- Valores de registros de recetas

Utilice esta función de sistema si ha configurado el visor de recetas simple. En el visor de recetas simple se muestra en cada caso un solo campo de selección en el panel de operador. Utilice la función de sistema "VisorDeRecetasAtrás" para ver el campo de selección anterior.

Utilización en la lista de funciones

VisorDeRecetasAbrir (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe ejecutar el comando.

Consulte también

VisorDeRecetasAtrás (Página 8684)

VisorDeRecetasEscribirRegistroEnControlador

Descripción

Transfiere al controlador el registro de receta visualizado actualmente en el visor de recetas.

Utilización en la lista de funciones

VisorDeRecetasEscribirRegistroEnControlador (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas desde el que se debe transferir el registro al controlador conectado.

VisorDeRecetasGuardarRegistro

Descripción

Guarda el registro de receta visualizado actualmente en el visor de recetas.

Utilización en la lista de funciones

VisorDeRecetasGuardarRegistro (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe guardar el registro.

VisorDeRecetasGuardarRegistroComo

Descripción

Guarda con otro nombre el registro que se visualiza en el visor de recetas.

Utilización en la lista de funciones

VisorDeRecetasGuardarRegistroComo (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Guarda con otro nombre o número el registro que se visualiza en el visor de recetas.

VisorDeRecetasCambiarNombreDeRegistro

Descripción

Cambia el nombre del registro seleccionado en el visor de recetas indicado.

Utilice esta función de sistema sólo en un visor de recetas simple.

Utilización en la lista de funciones

VisorDeRecetasCambiarNombreDeRegistro (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe cambiar el nombre del registro.

VisorDeRecetasMostrarTextoDeAyuda

Descripción

Muestra el tooltip configurado para el visor de recetas indicado.

Utilización en la lista de funciones

VisorDeRecetasMostrarTextoDeAyuda (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas cuyo tooltip configurado se visualiza.

VisorDeRecetasAtrás

Descripción

Vuelve al campo de selección anterior en el visor de recetas indicado.

Esta función de sistema no tiene ningún efecto si el campo de selección de la receta se muestra en el panel de operador. Orden de manejo de los campos de selección en runtime:

- Nombre de receta
- Nombre de registro
- Valores de registros de recetas

Utilice esta función de sistema si ha configurado el visor de recetas simple. En el visor de recetas simple se muestra en cada caso un solo campo de selección en el panel de operador. Utilice la función de sistema "VisorDeRecetasAbrir" para ver los valores del registro de receta o el siguiente campo de selección.

Utilización en la lista de funciones

VisorDeRecetasAtrás (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del visor de recetas en el que se debe ejecutar el comando.

DesactivarBit

Descripción

Pone a 0 (FALSE) el valor de una variable del tipo "Bool".

Utilización en la lista de funciones

DesactivarBit (Variable)

Utilización en funciones definidas por el usuario

ResetBit (Tag)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable del tipo BOOL que se ajusta a "0" (FALSE).

Ejemplo

El siguiente código de programa pone a 0 el valor de la variable booleana b_value con la función ResetBit y muestra el resultado junto con el valor original b_saved.

```
{
BOOL b_value = 1;
BOOL b_saved = b_value;

//Reset bit
  ResetBit (b_value);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",b_value, b_saved);
...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

DesactivarBitEnVariable (Página 8686)

DesactivarBitEnVariable

Descripción

Pone un bit a 0 (FALSE) en las variables indicadas.

Tras modificar el bit indicado, la función de sistema vuelve a transferir toda la variable al controlador. No se comprueba si entretanto se han modificado otros bits en la variable. El operador y el controlador sólo podrán acceder a la variable indicada en modo de lectura hasta que la variable vuelva a transferirse al controlador.

Nota

No utilice esta función de sistema si el controlador admite variables BOOL. En su lugar utilice la función de sistema "DesactivarBit".

Utilización en la lista de funciones

DesactivarBitEnVariable (Variable, Bit)

Utilización en funciones definidas por el usuario

ResetBitInTag (Tag, Bit)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable en la que un bit se ajusta a "0" (FALSE).

Bit

Número del bit que se ajusta a "0" (FALSE).

Si utiliza esta función de sistema en un script de usuario, los bits de la variable indicada se contarán de derecha a izquierda, independientemente del controlador utilizado. El contaje comienza a partir de "0".

Ejemplo

El siguiente código de programa pone a 0 un bit en la posición indicada bitposition de la variable bvalue y muestra el resultado junto con el valor original bsaved.

```
{
BYTE bvalue;
BYTE bsaved = bvalue;
BYTE bitposition = 2;

//Reset bit in bitposition
ResetBitInTag (bvalue, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bvalue, bsaved);
  ...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

DesactivarBit (Página 8683)

BotónPulsar

Descripción

La función de sistema sólo se puede asociar a las teclas de función de un panel de operador y lanza el evento "Pulsar tecla" en el objeto de imagen indicado.

Utilice esta función de sistema por ejemplo si desea activar un botón de la imagen desde una tecla de función del panel de operador.

Nota

Configure las funciones de sistema "BotónPulsar" y "BotónSoltar" siempre conjuntamente. Por tanto, si se asocia la función de sistema "BotónPulsar" al evento "Pulsar tecla" de una tecla de función, debe asociarse la función de sistema "BotónSoltar" al evento "Soltar" de la misma tecla de función.

Utilización en la lista de funciones

BotónPulsar (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del objeto de imagen en el que se lanza el evento.

BotónSoltar

Descripción

La función de sistema sólo se puede asociar a las teclas de función de un panel de operador y lanza el evento "Soltar tecla" en el objeto de imagen indicado.

Utilice esta función de sistema por ejemplo si desea activar un botón de la imagen desde una tecla de función del panel de operador.

Nota

Configure las funciones de sistema "BotónPulsar" y "BotónSoltar" siempre conjuntamente. Por tanto, si se asocia la función de sistema "BotónPulsar" al evento "Pulsar tecla" de una tecla de función, debe asociarse la función de sistema "BotónSoltar" al evento "Soltar tecla" de la misma tecla de función.

Utilización en la lista de funciones

BotónSoltar (Objeto de imagen)

Utilización en scripts de usuario

-

Parámetros

Objeto de imagen

Nombre del objeto de imagen en el que se lanza el evento.

DesplazarYEnmascarar

Descripción

La función de sistema convierte la configuración binaria de entrada de la variable de origen en una configuración binaria de salida de la variable de destino. Durante este proceso se desplazan y se enmascaran bits.

Nota

Si las variables de origen y destino tienen un número de bits diferente, la aplicación de la función de sistema en la variable de destino no puede causar un rebase del rango de valores.

Utilización en la lista de funciones

DesplazarYEnmascarar (Variable de origen, Variable de destino, Bits a desplazar, Bits a enmascarar)

Utilización en funciones definidas por el usuario

ShiftAndMask (Source_tag, Target_tag, Bits_to_shift, Bits_to_mask)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable de origen

La variable contiene la configuración binaria de entrada. Se admiten variables de números enteros, p. ej. "Byte", "Char", "Int", "UInt", "Long" y "ULong".

Ejemplo: La variable de origen del tipo "16 Bit Integer" tiene el valor 72: 0000000001001000.

Variable de destino

En la variable se guarda la configuración binaria de salida. Se admiten variables de tipo entero, p. ej. "Byte", "Char", "Int", "UInt", "Long" y "ULong".

Ejemplo: La configuración binaria de entrada desplazada se multiplica con la máscara de bits, bit por bit con la operación lógica Y: 0000000000001001. El resultado tiene el valor decimal "8" y se guarda en la variable de destino.

Tenga en cuenta lo siguiente:

- Las variables de origen y destino tienen el mismo número de bits.
- La cantidad de bits a desplazar es menor que la cantidad de bits de las variables de origen y de destino.
- "Bits a enmascarar" no contiene más bits que las variables de origen y destino.

Bits a desplazar

Cantidad de bits según la que se debe desplazar hacia la derecha la configuración binaria de entrada. Un valor negativo desplaza la configuración binaria de entrada hacia la izquierda.

Ejemplo: "Bits a desplazar" tiene el valor "+3". Al llamar la función de sistema, la configuración binaria de entrada se desplaza 3 bits hacia la derecha: 0000000000001001.
El lado izquierdo se rellena con "0". En el lado derecho se truncan 3 bits. El nuevo valor decimal es "9".

Nota

Si la variable de origen tiene un tipo de datos Integer con el signo "-", el bit izquierdo será "1". Al desplazar hacia la derecha, este bit de signo se rellena con "0". El signo se conmuta a "+".

Bits a enmascarar

Un número del tipo Integer sirve de máscara de bits Con su configuración binaria se multiplica la configuración binaria de entrada desplazada. Ejemplo: número "2478" con la configuración binaria "0000100110101110".

La máscara de bits se puede indicar de 3 maneras distintas:

- Hexadecimal: Introduzca primero "0h" o "0H" como prefijo, seguido de un espacio en blanco opcional para una mejor legibilidad. Seguidamente agrupe la configuración binaria en bloques de cuatro bits (0000)(1001)(1010)(1110) y represente cada bloque en el sistema hexadecimal. (0)(9)(A)(E). Sólo se admite la entrada de los caracteres 0-9, A-F, a-f: "0h 09AE".
- Binario: Introduzca primero "0b" o "0B" como prefijo, seguido de un espacio en blanco opcional para una mejor legibilidad. Para comprobarlo, agrupe después la configuración binaria en bloques 0000 1001 1010 1110 separados por espacios en blanco. Sólo se admite la entrada de los caracteres "0" ó "1": "0b 0000 1001 1010 1110".
- Decimal: Introduzca directamente el valor "2478" sin prefijo.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

RetrocederPágina

Descripción

Ejecuta la función de tecla <RePág> en el panel de operador.

Esta función de sistema sólo se puede utilizar para teclas de función y para tareas con disparador temporal.

Utilización en la lista de funciones

RetrocederPágina

Utilización en funciones definidas por el usuario

PageUp

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

-

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

AvanzarPágina

Descripción

Ejecuta la función de tecla <Av Pág> en el panel de operador.

Esta función de sistema sólo se puede utilizar para teclas de función.

Utilización en la lista de funciones

AvanzarPágina

Utilización en funciones definidas por el usuario

PageDown

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

-

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

EstablecerModoDeOperaciónPanel

Descripción

Cambia el modo de operación en el panel de operador. Se dispone de los modos de operación siguientes: "Online", "Offline" y "Cargar".

Utilización en la lista de funciones

EstablecerModoDeOperaciónPanel (Modo de operación)

Utilización en funciones definidas por el usuario

SetDeviceMode (Operating_mode)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Modo de operación

Determina el modo de operación del panel de operador:

0 (hmiOnline) = Online: Se establece la conexión con el controlador. Al hacerlo, siempre se ajusta el estado de conexión configurado. No se tienen en cuenta los estados que se utilizaron por última vez en runtime.

1 (hmiOffline) = Offline: Se deshace la conexión con el controlador.

2 (hmiTransfer) = Cargar: Permite transferir proyectos desde el equipo de configuración al panel de operador.

Nota

Si utiliza un PC como panel de operador, al conmutar al modo de operación "Cargar" se cierra el software runtime.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

EstablecerModoDeConexión (Página 8701)

ActivarBit

Descripción

Ajusta a "1" (TRUE) el valor de una variable del tipo "Bool".

Utilización en la lista de funciones

ActivarBit (Variable)

Utilización en funciones definidas por el usuario

SetBit (Tag)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable del tipo BOOL que se ajusta a "1" (TRUE).

Ejemplo

El siguiente código de programa pone a 1 el valor de la variable booleana b_value con la función SetBit y muestra el resultado junto con el valor original b_saved.

```
{
BOOL b_value = 0;
BOOL b_saved = b_value;

//Set bit
  SetBit (b_value);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",b_value, b_saved);
...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

ActivarBitDeVariable

Descripción

Ajusta un bit a "1" (TRUE) en la variable indicada.

Tras modificar el bit indicado, la función de sistema vuelve a transferir toda la variable al controlador. No se comprueba si entretanto se han modificado otros bits en la variable. El operador y el controlador sólo podrán acceder a la variable indicada en modo de lectura hasta que la variable vuelva a transferirse al controlador.

Nota

No utilice esta función de sistema si el controlador admite variables BOOL. En su lugar utilice la función de sistema "ActivarBit".

Utilización en la lista de funciones

ActivarBitDeVariable (Variable, Bit)

Utilización en funciones definidas por el usuario

SetBitInTag(Tag, Bit)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable en la que un bit se ajusta a "1" (TRUE).

Bit

Número del bit que se ajusta a "1" (TRUE).

Si utiliza esta función de sistema en un script de usuario, los bits de la variable indicada se contarán de derecha a izquierda, independientemente del controlador utilizado. El contaje comienza a partir de "0".

Nota

Para garantizar una funcionalidad fiable es imprescindible una actualización garantizada de las variables utilizadas con los valores de proceso actuales. Por tanto, configure la variable en un campo E/S, o bien asocie la función de sistema a un objeto de imagen (p. ej. un botón).

Si asocia la función de sistema a un evento breve (p. ej. aparición de un mensaje), los valores de proceso actuales sólo se alcanzarán si la variable puede leerse de forma permanente.

Ejemplo

El siguiente código de programa pone a 1 un bit en la posición indicada bitposition de la variable bvalue y muestra el resultado junto con el valor original bsaved.

```
{
BYTE bvalue;
BYTE bsaved = bvalue;
BYTE bitposition = 2;

//Reset bit in bitposition
SetBitInTag (bvalue, bitposition);

//print current and saved value
printf ("Current value: %d\r\n, Saved value: %d\r\n",bvalue, bsaved);
  ...
}
```

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

ActivarBit (Página 8691)

ActivarBitMientrasTeclaPulsada

Descripción

Ajusta un bit a "1" (TRUE) en la variable indicada mientras el usuario mantiene pulsada la tecla configurada.

Tras modificar el bit indicado, la función de sistema vuelve a transferir toda la variable al controlador. No se comprueba si entretanto se han modificado otros bits en la variable. El operador y el controlador sólo podrán acceder a la variable indicada en modo de lectura hasta que la variable vuelva a transferirse al controlador. Para evitar problemas debidos a accesos simultáneos a una misma variable, es recomendable que utilice esta función de sistema sólo para acceder a variables del tipo BOOL.

Nota

Con el cambio de imagen configurado para una tecla se ejecutan inmediatamente todas las funciones configuradas para el evento "Soltar", aunque la tecla todavía esté pulsada.

Si para una tecla de función está configurada la función de sistema "ActivarBitMientrasTeclaPulsada", el bit se desactivará inmediatamente tras ejecutar el cambio de imagen. Este comportamiento es necesario, puesto que la asignación de las teclas será diferente tras el cambio de imagen.

No utilice esta función de sistema si el controlador admite variables BOOL. En su lugar utilice la función de sistema "ActivarBit".

Utilización en la lista de funciones

ActivarBitMientrasTeclaPulsada (Variable, Bit)

Utilización en scripts de usuario

-

Parámetros

Variable

Variable en la que un bit se ajusta temporalmente a "1" (TRUE). Si el controlador lo permite, utilice sólo variables del tipo BOOL.

Bit

Número del bit que se ajusta temporalmente a "1" (TRUE).

Nota

Para garantizar una funcionalidad fiable es imprescindible una actualización garantizada de las variables utilizadas con los valores de proceso actuales. Por tanto, configure la variable en un campo ES, o bien asocie la función a un elemento de imagen (p. ej. un botón).

Si asocia la función a un evento breve (p. ej. aparición de un mensaje), los valores de proceso actuales sólo se alcanzarán si la variable puede leerse de forma permanente.

Consulte también

ActivarBit (Página 8691)

Funciones de sistema disponibles según el equipo (Página 8644)

EstablecerColorRetroiluminación

Descripción

Define el color de la retroiluminación.

Nota

Cuando rearranque el panel de operador se restablecerá la configuración que estaba ajustada al apagarlo.

Utilización en la lista de funciones

EstablecerColorRetroiluminación (Valor)

Utilización en scripts de usuario

-

Parámetros

Valor

Establece el color de la retroiluminación:

0 (hmiWhite) = blanco: Sin color

1 (hmiGreen) = verde: Color verde

2 (hmiYellow) = amarillo: Color amarillo

3 (hmiRed) = rojo: Color rojo

DefinirFechaHoraPLC

Descripción

Modifica la fecha y la hora del controlador conectado.

La función de sistema "AjustarFechaHoraPLC" solo se puede configurar para los siguientes controladores:

- SIMATIC S7 1200
- SIMATIC S7 1500

Utilización en la lista de funciones

AjustarFechaHoraPLC (conexión, hora)

Utilización en funciones definidas por el usuario

SetPLCDateTime (Connection, Time)

Parámetros

Conexión

Conexión de controlador y panel de operador.

Hora

Transfiere la fecha y la hora del panel de operador al controlador. El controlador aplica la fecha y la hora del panel de operador.

AjustarModoPLC

Descripción

Cambia el modo de operación del controlador a uno de los siguientes estados:

- RUN
- STOP

La función de sistema "AjustarModoPLC" sólo se puede configurar para los siguientes controladores:

- SIMATIC S7 1200
- SIMATIC S7 1500

Utilización en la lista de funciones

AjustarModoPLC (conexión, modo)

Utilización en funciones definidas por el usuario

SetPLCMode (Connection, Mode)

Parámetros

Conexión

Conexión de controlador y panel de operador.

Modo

Determina el modo de operación del panel de operador:

RUN = el controlador pasa al estado RUN. Se procesa el programa de control.

STOP = el controlador pasa al estado STOP. Se interrumpe el programa de control.

EstablecerIdioma

Descripción

Cambia el idioma del panel de operador. Todos los textos y avisos del sistema configurados aparecerán en el panel de operador en el idioma recién ajustado.

Utilización en la lista de funciones

EstablecerIdioma (Idioma)

Utilización en funciones definidas por el usuario

SetLanguage (Language)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Idioma

Determina qué idioma se debe ajustar en el panel de operador. Se pueden realizar los siguientes ajustes:

- -1 (hmiToggle) = Conmutar: Cambia al siguiente idioma. El orden se establece en el editor "Idiomas del proyecto" durante el proceso de configuración.
- Número que se ha especificado en "Idioma y fuente" en el editor "Configuración de runtime". Cambia al idioma con el número indicado.
- Idioma que se ha especificado en "Idioma y fuente" en el editor "Configuración de runtime".
- Abreviatura del idioma conforme a la referencia VBScript5: Cambia al idioma correspondiente a la abreviatura indicada, p. ej. "de-DE" para alemán (Alemania) o "en-US" para inglés (EE.UU.).
  En el tema "Cuadro de Id. de configuración regional (LCID)" que forma parte de los principios básicos de VBScript encontrará una lista de todas las abreviaturas de idioma.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

DefinirVariable

Descripción

Asigna un valor nuevo a la variable indicada.

Nota

Dependiendo del tipo de variable, con esta función de sistema puede asignar cadenas de caracteres y números.

Utilización en la lista de funciones

DefinirVariable (Variable, Valor)

Utilización en funciones definidas por el usuario

SetTag (Tag, Value)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable a la que se debe asignar el valor indicado.

Valor

Valor que se asigna a la variable indicada.

Nota

La función de sistema "DefinirVariable" se ejecuta únicamente tras establecer una conexión.

Ejemplo

El siguiente código de programa ajusta con la función SetTag el valor de las variables gs_tag_bit a TRUE y guarda el valor de retorno en la variable ok.

```
{
BOOL ok;
BOOL bvalue;

//Set the tag to true
ok = SetTag("gs_tag_bit", TRUE);
//error handling
if(ok)
{
  // succeeded
  printf ( "Function has run through.\r\n" );
bvalue = GetTagBit("gs_tag_bit");
  printf ("Value of gs_tag_bit: %d\r\n", bvalue);
}
else
{
  // failed
  printf ( "Error - function failed." );
}
...
}
```

El valor de retorno guardado puede procesarse en el código subsiguiente.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

IncrementarVariable (Página 8658)

EstablecerModoDeConexión

Descripción

Establece o desconecta la conexión indicada.

Nota

Sólo puede establecerse una conexión con el controlador si el panel de operador funciona en modo "online". Utilice para ello la función de sistema "EstablecerModoDeOperaciónPanel".

Utilización en la lista de funciones

EstablecerModoDeConexión (Modo, Conexión)

Utilización en funciones definidas por el usuario

SetConnectionMode (Mode, Connection)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Modo

Determina si la conexión con el controlador se debe establecer o desconectar:

0 (hmiOnline) = Online: Se establece la conexión.

1 (hmiOffline) = Offline: Se deshace la conexión.

Conexión

Controlador que se debe conectar con el panel de operador. El nombre del controlador se establece en el editor de conexiones.

Utilización múltiple de la función de sistema en un script de usuario

Si utiliza la función de sistema "EstablecerModoDeConexión" para diferentes conexiones, puede suceder que no todas las funciones de sistema se ejecuten correctamente. Para evitar este problema, proceda del siguiente modo:

1. Cree una variable del tipo de datos "BOOL" con el valor inicial "0".
2. Asocie la función de sistema "EstablecerModoDeConexión" al evento "Modificación de valor" de la variable HMI. Si desea deshacer p. ej. 3 conexiones, deberá configurar la función de sistema tres veces.
3. En el script de usuario, aplique a la variable HMI la función de sistema "InvertirBit".

Ejemplo de aplicación

A continuación se indican dos ejemplos de aplicación típicos de esta función de sistema:

- Test
  Mientras que no haya ningún controlador conectado al panel de operador, durante el test no se indicarán mensajes de error en el panel de operador. Si el panel de operador está conectado a un controlador, podrá establecer la conexión con éste último pulsando una tecla.
- Puesta en servicio
  Se deben configurar varios controladores para una instalación. Configure primero todos los controladores "offline", exceptuando uno de ellos. Tras poner en marcha el primer controlador, establezca la conexión con todos los demás, pulsando para ello la tecla correspondiente. Así podrá poner en marcha los demás controladores uno tras otro.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

EstablecerModoDeOperaciónPanel (Página 8690)

SimularTeclaSistema

Descripción

Simula el comportamiento de una tecla del sistema. Utilice esta función de sistema si el panel de operador no dispone de una tecla de sistema, p. ej. la tecla "ACK", la tecla ENTER o el bloque numérico.

Utilización en la lista de funciones

SimularTeclaSistema(Tecla del sistema)

Utilización en scripts de usuario

-

Parámetros

Tecla del sistema

Tecla del sistema cuyo comportamiento debe ser simulado.

Tecla del sistema "+/-"

En la función de sistema SimularTeclaSistema, la tecla de sistema "+/-" sólo es soportada por los paneles de operador siguientes:

- KP300 Basic
- KP400 Basic
- KTP400 Basic mono PN
- KTP400 Basic color PN
- KTP400 Basic color PN Portrait
- KTP600 Basic mono PN
- KTP600 Basic color PN
- KTP600 Basic color DP
- KTP1000 Basic PN
- KTP1000 Basic DP

En el resto de paneles de operador, utilice las teclas de sistema "+" y "-" por separado.

SimularVariable

Descripción

Simula el comportamiento de variables y objetos dinámicos (p.ej. listas de texto), sin que el panel de operador esté conectado al controlador. Por ejemplo, puede configurar la función de sistema y el evento "Creada" de una imagen.

Utilice esta función de sistema p. ej. para presentar la funcionalidad de un proyecto.

Para la simulación sólo se pueden utilizar variables del tipo de datos Integer. Sin embargo, en el OP 73, OP 77A y TP 177A pueden utilizarse variables de los tipos de datos Integer y Double Integer.

Nota

Si se utiliza la función de sistema "SimularVariable" en combinación con un tiempo de ciclo corto en un Basic Panel, el panel de operador se puede sobrecargar.

Utilización en la lista de funciones

SimularVariable (Variable, Ciclo, Valor máximo, Valor mínimo, Valor)

Utilización en funciones definidas por el usuario

-

Parámetros

Variable

Variable cuyo valor se debe modificar.

Ciclo

Factor con el que se multiplica el ciclo básico de 200 milisegundos. El ciclo determina cuándo debe cambiar el valor de la variable por el valor indicado. El ciclo puede estar comprendido entre 1 y 32767.

Valor máximo

Valor máximo que puede adoptar el valor de la variable durante la simulación. El valor máximo debe ser mayor que el valor mínimo, pero menor o igual a 32767.

Valor mínimo

Valor mínimo que puede adoptar el valor de la variable durante la simulación. El valor mínimo debe ser menor que el valor máximo, pero mayor o igual a -32768.

Valor

Valor por el que se cambia el valor de la variable en cada ciclo. El valor puede estar comprendido entre -32768 y 32767.

- Un valor positivo incrementa el valor de la variable. Al alcanzarse el valor máximo, el valor de la variable se ajusta al valor mínimo tras concluir el siguiente ciclo de actualización.
- Un valor negativo reduce el valor de la variable. Al alcanzarse el valor mínimo, el valor de la variable se ajusta al valor máximo tras concluir el siguiente ciclo de actualización.

PararRuntime

Descripción

Sale del software runtime y, con ello, del proyecto que se está ejecutando en el panel de operador.

Utilización en la lista de funciones

PararRuntime (Modo)

Utilización en funciones definidas por el usuario

StopRuntime (Mode)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Modo

Determina si al finalizar el software runtime también se debe cerrar el sistema operativo.

0 (hmiStopRuntime) = Runtime: No cerrar el sistema operativo

1 (hmiStopRuntimeAndOperatingSystem) = Runtime y sistema operativo: Cerrar el sistema operativo (no en Windows CE)

Ejemplo

El siguiente código de programa cierra Runtime y el sistema operativo.

```
{

//Stop runtime and shutdown
StopRuntime (hmiStopRuntimeAndOperationSystem);

}
```

El valor de retorno guardado puede procesarse en el código subsiguiente.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

IdentificarCambioDeUsuario

Descripción

Visualiza un aviso de sistema que indica qué usuarios están conectados al panel de operador.

Esta función de sistema sólo se puede utilizar en el planificador de tareas.

Utilización en la lista de funciones

IdentificarCambioDeUsuario

Utilización en scripts de usuario

-

Parámetros

--

ReducirValorEnfocado

Descripción

Resta el valor indicado del valor de la variable vinculada al objeto de imagen enfocado actualmente.

Esta función de sistema sólo se puede utilizar en teclas de función.

Utilización en la lista de funciones

ReducirValorEnfocado (Valor)

Utilización en scripts de usuario

-

Parámetros

Valor

Valor que se va a restar del valor de la variable.

ReducirVariable

Descripción

Resta el valor indicado del valor de la variable.

X = X - a

Nota

La función de sistema utiliza la misma variable como valor de entrada y como valor de salida. Si utiliza esta función de sistema para convertir un valor, deberá utilizar una variable auxiliar. A la variable auxiliar se le asigna un valor con la función de sistema "DefinirVariable".

Si asocia la función de sistema a eventos de un aviso y la variable no se utiliza en la imagen actual, no se garantiza que el valor real de la variable se utilice en el controlador. Ello se puede mejorar activando el modo de adquisición "Cíclico continuo".

Utilización en la lista de funciones

ReducirVariable (Variable, Valor)

Utilización en funciones definidas por el usuario

DecreaseTag (Tag, Value)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Variable

Variable de la que se debe restar el valor indicado.

Valor

Valor que se va a restar.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

CambiarConexión

Descripción

Deshace la conexión con el controlador que se está utilizando y establece una conexión con un controlador en una dirección distinta. El controlador recién conectado debe pertenecer a la misma clase de dispositivos (S7-1200, S7-300, ...etc.). Además, en el S7-1200 la utilización de la función sólo es admisible con direccionamiento absoluto.

Nota

Al cambiar a una dirección diferente, vigile que esta dirección no esté siendo utilizada por otro panel de operador.

Se admiten los siguientes tipos de dirección:

- Dirección IP
- Dirección MPI

Se admiten los siguientes tipos de controlador:

- SIMATIC S7 300/400
- SIMATIC S7 NC
- SIMOTION

Utilización en la lista de funciones

CambiarConexión (Conectar, Dirección, Slot, Rack)

Utilización en funciones definidas por el usuario

ChangeConnection (Connection, Address, Slot, Rack)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Conexión

Nombre de la conexión que se desconecta. El nombre se establece en el editor "Conexiones" durante el proceso de configuración.

Dirección

Dirección MPI/PROFIBUS o IP del controlador con el que se establece la conexión.

Nota

La dirección se establece mediante una variable. La lista de objetos muestra variables de todo los tipos. Seleccione exclusivamente variables de los siguientes tipos:

- Conexión Ethernet: Tipo de datos "String"
- Conexión MPI: Tipos de datos "Int"

Slot

Slot del controlador con el que se establece la conexión.

Rack

Rack del controlador con el que se establece la conexión.

Ejemplo de aplicación

Desea utilizar un panel de operador junto con varias máquinas. Configure para ello en el proyecto los autómatas necesarios a los que desea cambiar pulsando un botón. Al cambiar de controlador se deshace la conexión con el controlador utilizado actualmente. A continuación, la conexión se establece con el nuevo controlador, utilizando parámetros de direccionamiento diferentes. Para acceder a los valores del nuevo controlador, configure las mismas variables para el controlador utilizado.

Se utiliza por defecto el controlador especificado al crear el proyecto.

1. En el editor "Conexiones", indique el nombre y la dirección del controlador.
2. Configure un botón en la imagen del proceso.
3. Asocie el evento "Pulsar" a la función de sistema "CambiarConexión".
4. Introduzca como parámetros el nombre de la conexión y la dirección del controlador.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

MostrarDiálogoInicioSesión

Descripción

Abre un cuadro de diálogo en el panel de operador con el que el usuario puede iniciar una sesión en el panel de operador.

Utilización en la lista de funciones

MostrarDiálogoInicioSesión

Utilización en scripts de usuario

-

Parámetros

--

MostrarTextoDeAyuda

Utilización

Muestra el tooltip configurado para el objeto seleccionado.

Si la función de sistema se ha asociado a una tecla de función, se mostrará el tooltip correspondiente al objeto de imagen resaltado en ese momento. Si al mismo tiempo se ha configurado un tooltip para la imagen, será posible conmutar a ésta bien sea pulsando la tecla <Intro> o bien haciendo doble clic en la ventana de ayuda.

Si la función de sistema se ha asociado a un botón, se mostrará sólo el tooltip de la imagen actual. Si se ha configurado también un tooltip asociado al botón, se mostrará primero dicho texto. Pulsando la tecla <Intro> o haciendo doble clic en la ventana de ayuda se conmuta al tooltip de la imagen actual.

Nota

Mientras esté abierta la ventana de ayuda, no se podrá operar ningún otro objeto de imagen. Para operar los objetos de imagen, cierre la ventana de ayuda.

Cerrar la ventana de ayuda

La ventana de ayuda se puede cerrar de las maneras siguientes:

En paneles de operador con teclado:

- Pulsando de nuevo la tecla <HELP>
- Pulsando la tecla <HELP>

En paneles de operador con pantalla táctil:

- Tocando el botón ❌

Utilización en la lista de funciones

MostrarTextoDeAyuda (Representación)

Utilización en funciones definidas por el usuario

ShowOperatorNotes (Display_mode)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Representación

Determina si el tooltip configurado se debe mostrar u ocultar:

0 (hmiOff) = Off: ocultar el tooltip configurado

1 (hmiOn) = On: mostrar el tooltip configurado

-1 (hmiToggle) = Conmutar: Conmuta entre ambos estados.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

MostrarVentanaDeAvisos

Descripción

Muestra u oculta la ventana de avisos en el panel de operador.

Utilización en la lista de funciones

MostrarVentanaDeAvisos (Nombre del objeto, Representación)

Utilización en funciones definidas por el usuario

ShowAlarmWindow (Object_name, Display_mode)

Utilizable, siempre que el dispositivo configurado soporte funciones definidas por el usuario. Encontrará más información al respecto en "Dependencia del dispositivo".

Parámetros

Nombre del objeto

Nombre del visor de avisos que se debe mostrar u ocultar.

Representación

Determina si la ventana de avisos se debe mostrar u ocultar.

0 (hmiOff) = Off: Ocultar el visor de avisos

1 (hmiOn) = On: Mostrar el visor de avisos

-1 (hmiToggle) = Conmutar: Conmuta entre ambos estados.

Consulte también

Funciones de sistema disponibles según el equipo (Página 8644)

10.6.4.2 Eventos

Resumen

Editores

Introducción

La siguiente tabla muestra los eventos que se producen en los distintos editores.

Sujeto a cambios técnicos.

Símbolo	Editor
	Imágenes
	Avisos HMI
	Variables HMI
	Planificador de tareas

1	Borrada (Página 8717)	X	--	--	--
2	Activar (Página 8717)	--	--	--	--
3	Cambio (Página 8717)	--	--	--	--
4	Creada (Página 8718)	X	--	--	--
5	Ejecutar (Página 8718)	--	--	--	X
6	Selección modificada (Página 8718)	--	--	--	--
7	Al rebasar por exceso (Página 8719)	--	--	X	--
8	Al rebasar por defecto (Página 8719)	--	--	X	--

9	Cuando el cuadro de diálogo está abierto (Página 8719)	--	--	--	X
10	Cuando el cuadro de diálogo está cerrado (Página 8720)	--	--	--	X
11	Cambio de usuario (Página 8720)	--	--	--	X
12	Cambio de imagen (Página 8720)	--	--	--	X
13	Desactivar (Página 8720)	--	--	--	X
14	Hacer doble clic (Página 8721)	--	--	--	--
15	Pulsar (Página 8721)	--	--	--	--
16	Entrada finalizada (Página 8722)	--	--	--	--
17	Pulsar ESC dos veces (Página 8722)	--	--	--	--
18	Desaparecido (Página 8722)	--	X	--	--
19	Aparecido (Página 8723)	--	X	--	--
20	Hacer clic (Página 8723)	--	--	--	--
21	Hacer clic al parpadear (Página 8723)	--	--	--	--
22	Loop-In-Alarm (Página 8724)	--	X	--	--
23	Soltar (Página 8724)	--	--	--	--
24	Desbordamiento del búfer de avisos (Página 8724)	--	--	--	X
25	Acusar (Página 8725)	--	X	--	--
26	Margen alcanzado (Página 8725)	--	--	--	--
27	Parada de runtime (Página 8725)	--	--	--	X
28	Pulsar tecla (Página 8726)	--	--	--	--
29	Soltar tecla (Página 8726)	--	--	--	--
30	Desbordamiento (Página 8726)	--	--	--	--
31	Conmutar OFF (Página 8727)	--	--	--	--
32	Conmutar ON (Página 8727)	--	--	--	--
33	Poca memoria disponible (Página 8727)	--	--	--	--
34	Poca memoria disponible, crítico (Página 8727)	--	--	--	--
35	Cambio de valor (Página 8728)	--	--	X	--
36	Tiempo agotado (Página 8728)	--	--	--	--

Objetos básicos

Introducción

La siguiente tabla muestra los eventos que se producen en los distintos objetos.

Sujeto a cambios técnicos.

Símbolo	Objeto
	Línea
	Elipse

Símbolo	Objeto
	Círculo
	Rectángulo
A	Campo de texto
	Visor de gráficos

						A	
1	Borrada (Página 8717)	--	--	--	--	--	--
2	Activar (Página 8717)	--	--	--	--	--	--
3	Cambio (Página 8717)	--	--	--	--	--	--
4	Creada (Página 8718)	--	--	--	--	--	--
5	Ejecutar (Página 8718)	--	--	--	--	--	--
6	Selección modificada (Página 8718)	--	--	--	--	--	--
7	Al rebasar por exceso (Página 8719)	--	--	--	--	--	--
8	Al rebasar por defecto (Página 8719)	--	--	--	--	--	--
9	Cuando el cuadro de diálogo está abierto (Página 8719)	--	--	--	--	--	--
10	Cuando el cuadro de diálogo está cerrado (Página 8720)	--	--	--	--	--	--
11	Cambio de usuario (Página 8720)	--	--	--	--	--	--
12	Cambio de imagen (Página 8720)	--	--	--	--	--	--
13	Desactivar (Página 8720)	--	--	--	--	--	--
14	Hacer doble clic (Página 8721)	--	--	--	--	--	--
15	Pulsar (Página 8721)	--	--	--	--	--	--
16	Entrada finalizada (Página 8722)	--	--	--	--	--	--
17	Pulsar ESC dos veces (Página 8722)	--	--	--	--	--	--
18	Desaparecido (Página 8722)	--	--	--	--	--	--
19	Aparecido (Página 8723)	--	--	--	--	--	--
20	Hacer clic (Página 8723)	--	--	--	--	--	--
21	Hacer clic al parpadear (Página 8723)	--	--	--	--	--	--
22	Loop-In-Alarm (Página 8724)	--	--	--	--	--	--
23	Soltar (Página 8724)	--	--	--	--	--	--
24	Desbordamiento del búfer de avisos (Página 8724)	--	--	--	--	--	--
25	Acusar (Página 8725)	--	--	--	--	--	--
26	Margen alcanzado (Página 8725)	--	--	--	--	--	--
27	Parada de runtime (Página 8725)	--	--	--	--	--	--
28	Pulsar tecla (Página 8726)	--	--	--	--	--	--
29	Soltar tecla (Página 8726)	--	--	--	--	--	--
30	Desbordamiento (Página 8726)	--	--	--	--	--	--
31	Conmutar OFF (Página 8727)	--	--	--	--	--	--
32	Conmutar ON (Página 8727)	--	--	--	--	--	--

						A	
33	Poca memoria disponible (Página 8727)	--	--	--	--	--	--
34	Poca memoria disponible, crítico (Página 8727)	--	--	--	--	--	--
35	Cambio de valor (Página 8728)	--	--	--	--	--	--
36	Tiempo agotado (Página 8728)	--	--	--	--	--	--

Elementos

Introducción

La siguiente tabla muestra los eventos que se producen en los distintos objetos.

Sujeto a cambios técnicos.

Símbolo	Objeto
	Campo E/S
	Botón
	Campo E/S simbólico
	Campo E/S gráfico
	Campo de fecha y hora
	Barra
	Interruptor

1	Borrada (Página 8717)	--	--	--	--	--	--	--
2	Activar (Página 8717)	X	X	X	X	--	--	X
3	Cambio (Página 8717)	--	X	X	--	--	--	X
4	Creada (Página 8718)	--	--	--	--	--	--	--
5	Ejecutar (Página 8718)	--	--	--	--	--	--	--
6	Selección modificada (Página 8718)	--	--	--	--	--	--	--
7	Al rebasar por exceso (Página 8719)	--	--	--	--	--	--	--
8	Al rebasar por defecto (Página 8719)	--	--	--	--	--	--	--
9	Cuando el cuadro de diálogo está abierto (Página 8719)	--	--	--	--	--	--	--
10	Cuando el cuadro de diálogo está cerrado (Página 8720)	--	--	--	--	--	--	--
11	Cambio de usuario (Página 8720)	--	--	--	--	--	--	--
12	Cambio de imagen (Página 8720)	--	--	--	--	--	--	--
13	Desactivar (Página 8720)	X	X	X	X	--	--	X
14	Hacer doble clic (Página 8721)	--	--	--	--	--	--	--
15	Pulsar (Página 8721)	--	X	--	--	--	--	--
16	Entrada finalizada (Página 8722)	--	--	--	--	--	--	--
17	Pulsar ESC dos veces (Página 8722)	--	--	--	--	--	--	--
18	Desaparecido (Página 8722)	--	--	--	--	--	--	--

19	Aparecido (Página 8723)	--	--	--	--	--	--	--
20	Hacer clic (Página 8723)	--	X	--	--	--	--	--
21	Hacer clic al parpadear (Página 8723)	--	--	--	--	--	--	--
22	Loop-In-Alarm (Página 8724)	--	--	--	--	--	--	--
23	Soltar (Página 8724)	--	X	--	--	--	--	--
24	Desbordamiento del búfer de avisos (Página 8724)	--	--	--	--	--	--	--
25	Acusar (Página 8725)	--	--	--	--	--	--	--
26	Margen alcanzado (Página 8725)	--	--	--	--	--	--	--
27	Parada de runtime (Página 8725)	--	--	--	--	--	--	--
28	Pulsar tecla (Página 8726)	--	--	--	--	--	--	--
29	Soltar tecla (Página 8726)	--	--	--	--	--	--	--
30	Desbordamiento (Página 8726)	--	--	--	--	--	--	--
31	Conmutar OFF (Página 8727)	--	--	--	--	--	--	X
32	Conmutar ON (Página 8727)	--	--	--	--	--	--	X
33	Poca memoria disponible (Página 8727)	--	--	--	--	--	--	--
34	Poca memoria disponible, crítico (Página 8727)	--	--	--	--	--	--	--
35	Cambio de valor (Página 8728)	--	--	--	--	--	--	--
36	Tiempo agotado (Página 8728)	--	--	--	--	--	--	--

Controles

Introducción

La siguiente tabla muestra los eventos que se producen en los distintos objetos.

Sujeto a cambios técnicos.

Símbolo	Objeto
	Visor de avisos/ventana de avisos
	Indicador de avisos
	Visor de curvas
	Visor de usuarios
	Visor de recetas
	Indicador de ayuda

1	Borrada (Página 8717)	--	--	--	--	--	--
2	Activar (Página 8717)	X	--	X	X	--	--
3	Cambio (Página 8717)	--	--	--	--	--	--
4	Creada (Página 8718)	--	--	--	--	--	--
5	Ejecutar (Página 8718)	--	--	--	--	--	--

6	Selección modificada (Página 8718)	--	--	--	--	--	--
7	Al rebasar por exceso (Página 8719)	--	--	--	--	--	--
8	Al rebasar por defecto (Página 8719)	--	--	--	--	--	--
9	Cuando el cuadro de diálogo está abierto (Página 8719)	--	--	--	--	--	--
10	Cuando el cuadro de diálogo está cerrado (Página 8720)	--	--	--	--	--	--
11	Cambio de usuario (Página 8720)	--	--	--	--	--	--
12	Cambio de imagen (Página 8720)	--	--	--	--	--	--
13	Desactivar (Página 8720)	X	--	X	X	--	--
14	Hacer doble clic (Página 8721)	--	--	--	--	--	--
15	Pulsar (Página 8721)	--	--	--	--	--	--
16	Entrada finalizada (Página 8722)	--	--	--	--	--	--
17	Pulsar ESC dos veces (Página 8722)	--	--	--	--	--	--
18	Desaparecido (Página 8722)	--	--	--	--	--	--
19	Aparecido (Página 8723)						
20	Hacer clic (Página 8723)	--	X	--	--	--	--
21	Hacer clic al parpadear (Página 8723)	--	X	--	--	--	--
22	Loop-In-Alarm (Página 8724)	--	--	--	--	--	--
23	Soltar (Página 8724)	--	--	--	--	--	--
24	Desbordamiento del búfer de avisos (Página 8724)	--	--	--	--	--	--
25	Acusar (Página 8725)	--	--	--	--	--	--
26	Margen alcanzado (Página 8725)	--	--	--	--	--	--
27	Parada de runtime (Página 8725)	--	--	--	--	--	--
28	Pulsar tecla (Página 8726)	--	--	--	--	--	--
29	Soltar tecla (Página 8726)	--	--	--	--	--	--
30	Desbordamiento (Página 8726)	--	--	--	--	--	--
31	Conmutar OFF (Página 8727)	--	--	--	--	--	--
32	Conmutar ON (Página 8727)	--	--	--	--	--	--
33	Poca memoria disponible (Página 8727)	--	--	--	--	--	--
34	Poca memoria disponible, crítico (Página 8727)	--	--	--	--	--	--
35	Cambio de valor (Página 8728)	--	--	--	--	--	--
36	Tiempo agotado (Página 8728)	--	--	--	--	--	--

Eventos

Borrada

Descripción

Aparece cuando la imagen activa desaparece del panel de operador.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Activar

Descripción

Aparece cuando el usuario selecciona un objeto de manejo y visualización mediante el orden de tabulación configurado.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Nota

El evento "Hacer clic" se activa cuando el usuario hace clic con el ratón en un botón, p.ej. Para disparar el evento "Activar", el usuario deberá seleccionar el botón mediante el orden de tabulación.

El evento "Activar" sirve exclusivamente para reconocer si un objeto está seleccionado. Sin embargo, el evento no lanza una petición de contraseña.

Por tanto, no utilice el evento "Activar" si quiere asociar una protección de acceso a la llamada de función del objeto.

Cambio

Descripción

Aparece cuando cambia el estado de un objeto de manejo y visualización.

Cuando el usuario acciona un interruptor, p. ej., cambie el estado de un objeto.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Creada

Descripción

Aparece cuando, tras producirse un cambio de imagen, todos los objetos de manejo y visualización configurados se cargan en la imagen activa.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Nota

Para asegurarse de que la conexión con el controlador se establece una vez conectado, active un cambio de imagen.

Ejecutar

Descripción

Aparece cuando se ejecuta la tarea planificada.

Selección modificada

Descripción

Aparece cuando el usuario cambia la selección.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Al rebasar por exceso

Descripción

Aparece cuando se rebasa por exceso el valor límite superior de una variable.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Al rebasar por defecto

Descripción

Aparece cuando se rebasa por defecto el valor límite inferior de una variable.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Cuando el cuadro de diálogo está abierto

Descripción

Ocurre cuando se abre un cuadro de diálogo modal.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Cuando el cuadro de diálogo está cerrado

Descripción

Ocurre cuando se cierra un cuadro de diálogo modal.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Cambio de usuario

Descripción

Aparece cuando un usuario finaliza la sesión en el panel de operador, o bien cuando otro usuario inicia una sesión.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Cambio de imagen

Descripción

Aparece cuando, tras producirse un cambio de imagen, todos los objetos de manejo y visualización configurados se cargan en la imagen.

Utilice el evento "Creada" para ejecutar una función de sistema diferente al cambiar a una imagen determinada.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Desactivar

Descripción

Aparece cuando el usuario quita el foco de un objeto de manejo y visualización.

Un objeto de imagen se puede desactivar mediante el orden de tabulación configurado, o bien utilizando el ratón para realizar una acción diferente.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Nota

Las funciones de sistema o los scripts de usuario asociados al evento "Desactivar" de una imagen no se ejecutan al desmarcar la imagen.

El evento "Desactivar" sirve exclusivamente para reconocer si un objeto se ha desmarcado. Sin embargo, el evento no lanza una petición de contraseña.

Por tanto, no utilice el evento "Desactivar" si quiere asociar una protección de acceso a la llamada de función del objeto.

Hacer doble clic

Descripción

Aparece cuando el usuario hace doble clic en un objeto de la librería de símbolos.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Pulsar

Descripción

Aparece cuando el usuario hace clic en un botón con el botón izquierdo del ratón, pulsa la tecla <INTRO> o <ESPACIO>.

También aparece si el usuario hace clic con el botón derecho del ratón en un objeto de la librería de símbolos.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Entrada finalizada

Descripción

Aparece cuando la entrada en un campo ES se confirma con la tecla Intro, con el ratón o con la pantalla táctil.

El evento "Entrada finalizada" también se inicia cuando el valor de una variable no cambia, p. ej. al rebasar un valor, o cuando un usuario cancela el cuadro de diálogo para acusar una variable de acuse obligatorio, p. ej. en el paquete opcional Audit.

Por el contrario, el evento no se dispara al iniciar la sesión como usuario y en los campos de entrada configurados con un permiso.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Pulsar ESC dos veces

Descripción

Aparece cuando el operador pulsa dos veces la tecla <ESC> en el panel de operador.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Desaparecido

Descripción

Aparece cuando desaparece un aviso.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Aparecido

Descripción

Aparece cuando se ha activado un aviso y éste se visualiza en el visor de avisos.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Hacer clic

Descripción

Aparece cuando el usuario hace clic con el ratón en un objeto de manejo y visualización, o bien cuando lo toca con el dedo en una pantalla táctil.

Si ha hecho clic en un objeto erróneo, puede evitar que se procese la lista de funciones configurada del modo siguiente:

- Manteniendo pulsado el botón del ratón, aleje el puntero del ratón del objeto. Una vez que el puntero del ratón haya salido del objeto, suelte el botón del ratón. En este caso, la lista de funciones no se procesará.
- Si utiliza una pantalla táctil, toque la pantalla con el dedo hasta que se produzca una acción (p. ej. un cambio de imagen).

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Hacer clic al parpadear

Descripción

Se activa cuando un usuario hace clic sobre un indicador de avisos intermitente con el ratón o lo toca con el dedo.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Loop-In-Alarm

Descripción

Aparece en cuanto el usuario selecciona un aviso en el visor y hace clic en el botón "Loop-In-Alarm" o doble clic en el aviso.

Se puede asociar una función de sistema al evento "Loop-In-Alarm", p. ej. cambiar a la imagen en la que ha aparecido el aviso.

No se puede asociar ningún script local al evento "Loop-In-Alarm" en Runtime Professional.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Soltar

Descripción

Aparece cuando el usuario suelta un botón de comando.

Este evento no aparecerá mientras mantenga pulsado el botón de comando.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Desbordamiento del búfer de avisos

Descripción

Aparece cuando se ha alcanzado el tamaño configurado del búfer de avisos.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Acusar

Descripción

Aparece cuando el usuario acusa un aviso.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Margen alcanzado

Descripción

Aparece cuando el usuario alcanza el comienzo o el final de un área de desplazamiento.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Nota

Al evento "Margen alcanzado" no se le puede asociar ninguna función definida por el usuario.

Objetos configurables

Este evento sólo se pueden configurar en las teclas <Arriba> y <Abajo>, o bien en las teclas en las que haya configurado las funciones de sistema "ObjetoDeImagenCursorArriba" u "ObjetoDeImagenCursorAbajo".

Parada de runtime

Descripción

Aparece cuando el usuario finaliza el software runtime en el panel de operador.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Nota

Al evento "Parada de runtime" no se le puede asociar ninguna función definida por el usuario.

Pulsar tecla

Descripción

Aparece cuando el usuario pulsa una tecla de función.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Soltar tecla

Descripción

Aparece cuando el usuario suelta una tecla de función.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Desbordamiento

Descripción

Aparece cuando se ha alcanzado el tamaño configurado de un fichero. Se utilizar el tipo de fichero "Disparar evento".

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Conmutar OFF

Descripción

Aparece cuando el usuario mueve el objeto de manejo e indicación "Interruptor" a la posición OFF.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Conmutar ON

Descripción

Aparece cuando el usuario mueve el objeto de manejo e indicación "Interruptor" a la posición ON.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Poca memoria disponible

Descripción

Aparece cuando en el medio de almacenamiento en el que se guarda un Audit Trail hay menos espacio de memoria disponible que la cantidad mínima configurada.

Poca memoria disponible, crítico

Descripción

Aparece cuando en el medio de almacenamiento en el que se guarda un Audit Trail no hay suficiente espacio de memoria disponible (debido al hardware).

Cambio de valor

Descripción

Aparece cuando cambia el valor de una variable o el valor de un elemento de matriz.

El valor de una variable es modificado por el controlador o bien por el usuario (p. ej. si introduce un nuevo valor). Si una función del sistema cambia el valor, no se notificará ningún evento.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

Tiempo agotado

Descripción

Aparece una vez transcurrido el tiempo configurado en el planificador de tareas.

Nota

Tenga en cuenta que la disponibilidad del evento depende del panel de operador y del tipo de objeto.

10.7 Planificar tareas

10.7.1 Campo de aplicación del planificador de tareas

Definición

En el planificador de tareas se configuran las tareas que se ejecutarán en segundo plano, independientemente de la imagen. Las tareas se crean vinculando las funciones de sistema o scripts a un disparador. Cuando se produce el evento activador, se llaman las funciones vinculadas.

Ejemplo de aplicación

El planificador de tareas sirve para ejecutar tareas controladas por eventos de forma automatizada. Con una tarea es posible automatizar p. ej. que

- los datos de fichero se trasvasen en intervalos regulares
- se imprima un informe al desbordarse el búfer de avisos

- se imprima un informe al finalizar el turno
- se vigile una variable
- se vigile un cambio de usuario

Nota

La disponibilidad de los ejemplos mencionados depende del panel de operador.

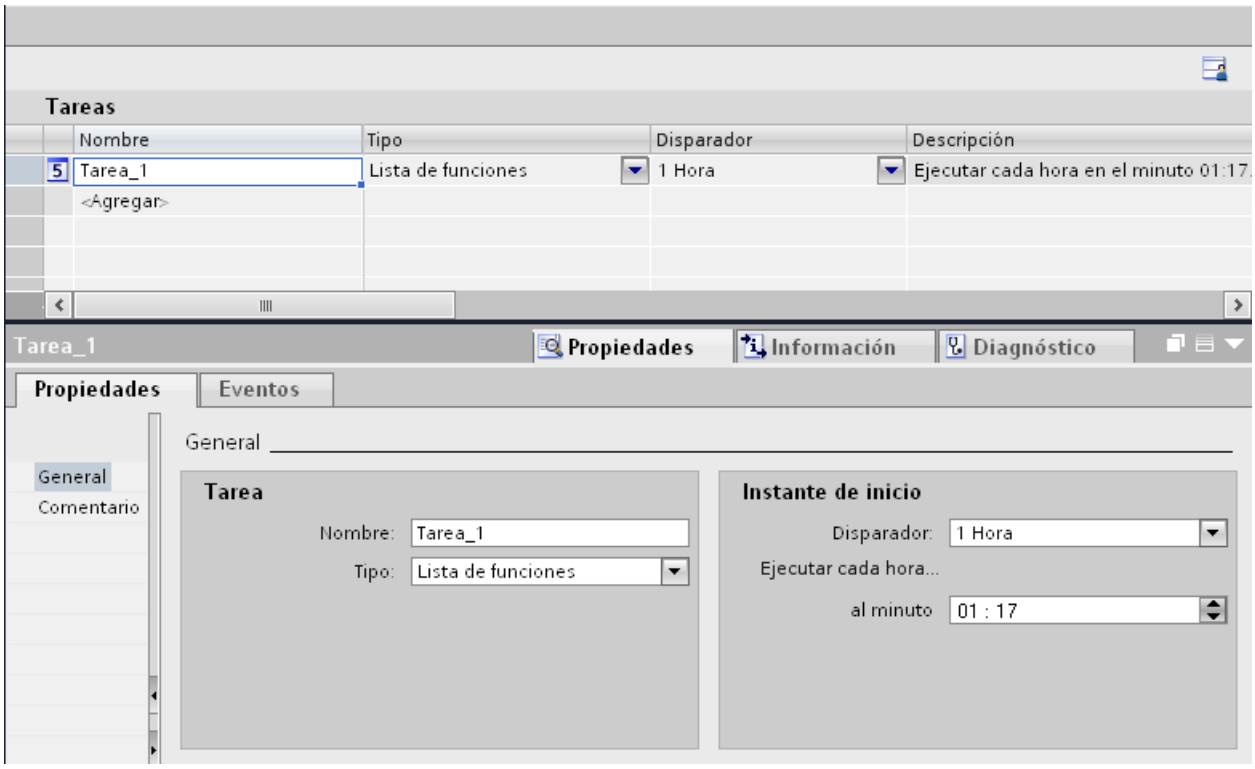
Consulte también

Trabajar con tareas y disparadores (Página 8729)
Ejemplo: Actualizar el usuario al cambiar de usuario (Página 8734)
Área de trabajo del editor Planificador de tareas (Página 8730)

10.7.2 Trabajar con tareas y disparadores

Introducción

Una tarea comprende un disparador y un tipo de tarea.



Inicio de una tarea

Controlada por un disparador, el planificador de tareas inicia la tarea vinculada al disparador.

Consulte también

Campo de aplicación del planificador de tareas (Página 8726)

10.7.3 Principios básicos

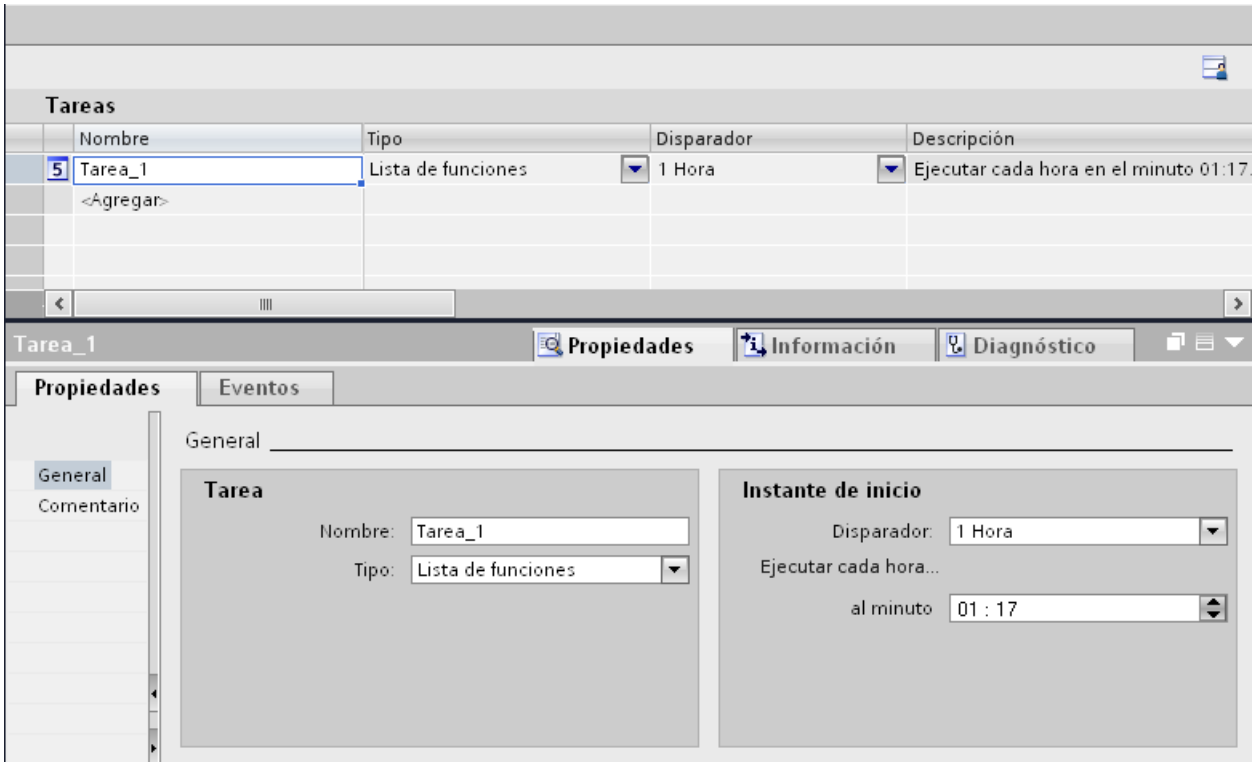
10.7.3.1 Área de trabajo del editor Planificador de tareas

Introducción

El planificador de tareas se abre en la ventana del proyecto haciendo doble clic en "Planificador de tareas". El área de trabajo muestra las tareas planificadas, que se componen del disparador que las desencadena y el tipo de tarea, p. ej. la lista de funciones.

Estructura

El área de trabajo está formada por la tabla de tareas.



La tabla de las tareas muestra las tareas definidas con sus propiedades, p. ej. el disparador que las desencadena. Hay que seleccionar un tipo de tarea y un disparador y asignar un nombre y un comentario a la tarea. La descripción resume en forma de texto la tarea con la hora planificada.

Ventana de inspección

La ficha "Propiedades" de la ventana de inspección se divide en dos áreas.

El área "Tarea" muestra también el nombre y el tipo de tarea. El área "Instante de inicio" muestra el disparador desencadenante. El área varía según sea la selección del disparador.

En la ficha "Eventos" se configura la lista con las funciones de sistema que se ejecutarán en la tarea.

Nota

Encontrará más información sobre los objetos de la interfaz en los tooltips. Desplace para ello el puntero del ratón hasta el objeto deseado, o bien pulse <F1> si el objeto está seleccionado.

Consulte también

Campo de aplicación del planificador de tareas (Página 8726)

Planificar una tarea con el disparo por eventos (Página 8732)

Disparador (Página 8732)

Lista de funciones (Página 8731)

10.7.3.2 Lista de funciones

Lista de funciones

Un disparador inicia la lista de funciones. La lista de funciones se procesa fila por fila. Cada fila contiene una función de sistema. En cada tarea se configura exactamente una lista de funciones.

Nota

Las funciones de sistema configurables que se pueden seleccionar en una lista de funciones dependen del disparador seleccionado y del panel de operador utilizado.

Consulte también

Área de trabajo del editor Planificador de tareas (Página 8728)

10.7.3.3 Disparador

Introducción

Un disparador se vincula a una tarea y, de este modo, forma el evento que desencadena la llamada de esta tarea. Cuando aparece el disparador, se ejecuta la tarea.

Disparo por eventos

Cuando una tarea está vinculada a un evento de sistema, se lanza controlada por el evento. Son eventos del sistema, por ejemplo, la parada de runtime, el cambio de imagen, el cambio de usuario, etc.

Cada evento de sistema puede configurarse una sola vez por panel de operador.

Desactivar una tarea

Si una tarea no se necesita temporalmente, es posible desactivarla en el sistema de ingeniería. El disparador "Desactivado" vuelve a poner a disposición un evento de sistema ya configurado.

Ejemplo: Ha planificado una tarea "A" con el evento de sistema "Parada de runtime". Por tanto, este evento de sistema ya no estará disponible para la tarea "B". Para volver a poner a disposición el evento de sistema "Parada de runtime", seleccione en la tarea "A" el disparador "Desactivado".

Nota

Los disparadores disponibles dependen del panel de operador.

Consulte también

Área de trabajo del editor Planificador de tareas (Página 8728)

10.7.3.4 Planificar una tarea con el disparo por eventos

Introducción

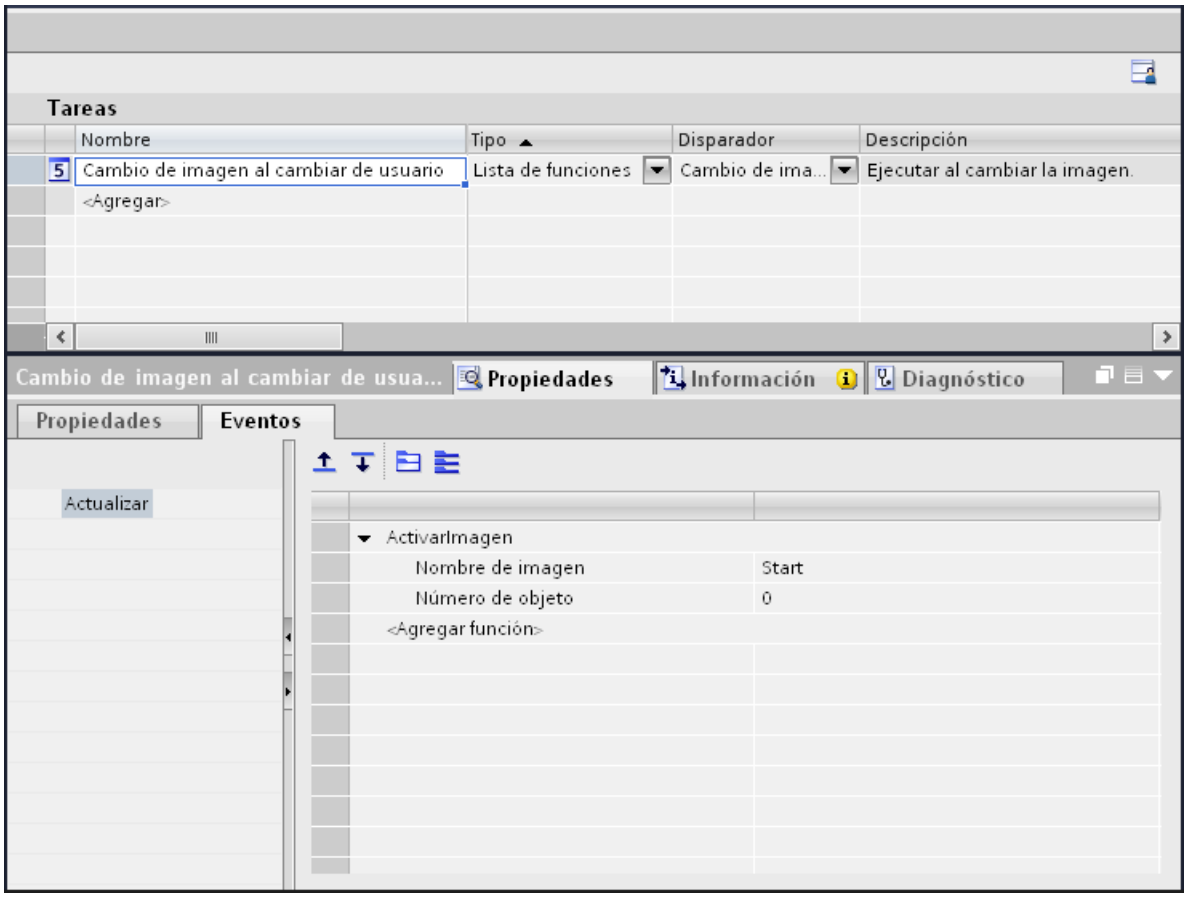
Se planifica una tarea que lance un cambio de imagen al cambiar de usuario.

Requisitos

- El área de trabajo "Planificador de tareas" está abierta.
- La imagen "Inicio" se deberá haber creado.

Procedimiento

1. Haga clic en "Agregar..." en la tabla del área de tareas.
2. En "Nombre", introduzca "Cambio de imagen al cambiar de usuario".
3. Seleccione el disparador "Cambio de usuario".
4. En la ventana de inspección haga clic en "Propiedades > Eventos".
5. Seleccione la función de sistema "Imágenes/ActivarImagen" en la lista de funciones.
6. Seleccione la imagen "Inicio" en el campo "Nombre de imagen".



Resultado

La tarea se ejecutará con el evento "Cambio de usuario". Si un usuario nuevo inicia una sesión, aparecerá la imagen "Inicio".

Consulte también

Área de trabajo del editor Planificador de tareas (Página 8728)

10.7.4 Ejemplos

10.7.4.1 Ejemplo: Actualizar el usuario al cambiar de usuario

Tarea

Desea configurar un campo E/S en el que se visualice el usuario que ha iniciado la sesión. Desea planificar una tarea que actualice el campo E/S al cambiar de usuario.

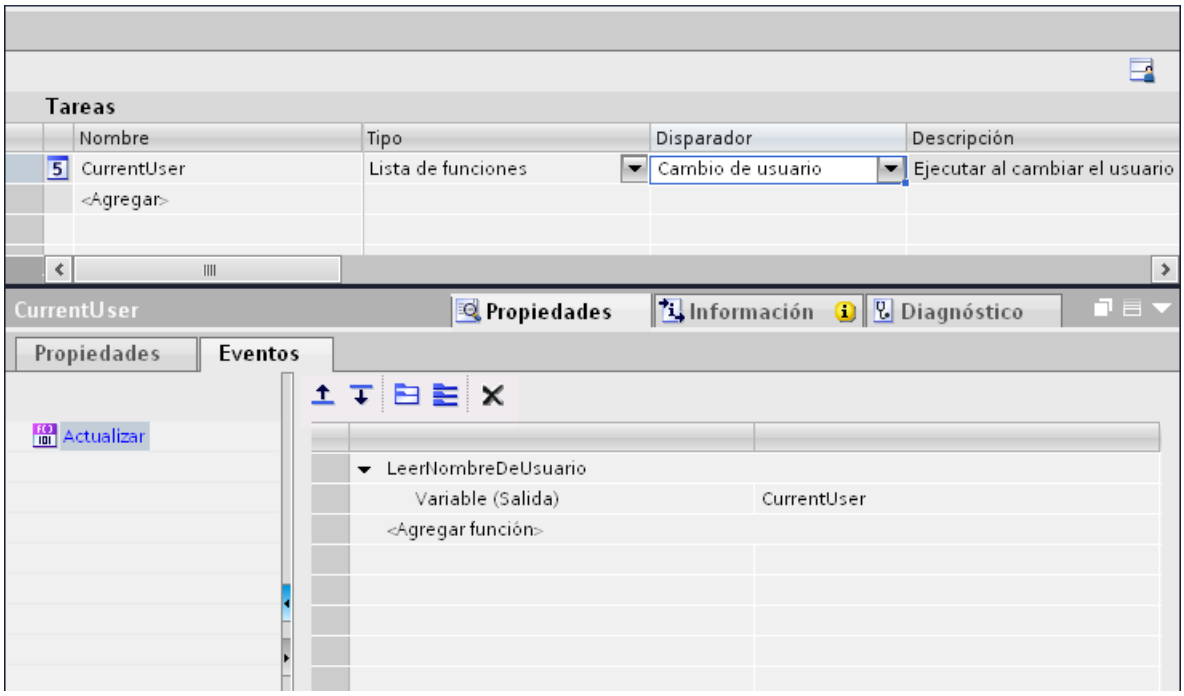
Requisitos

- Se deberá haber creado una variable "CurrentUser" del tipo "String".
- Se ha creado y abierto una imagen.
- Se ha creado un campo E/S en la imagen.

Procedimiento

1. Haga clic en el objeto "Campo E/S".
2. En la ventana de inspección haga clic en "Propiedades > Propiedades > General":
   – Seleccione el formato de visualización "Cadena de caracteres".
   – Seleccione la variable "CurrentUser".
   – Seleccione el modo "Salida".
3. Conmute al área de trabajo del planificador de tareas.
4. Haga clic en "Agregar..." en la tabla del área de tareas.
5. En "Nombre", introduzca "CurrentUser".
6. Seleccione el disparador "Cambio de usuario".
7. En la ventana de inspección haga clic en "Propiedades > Eventos".

8. Seleccione la función de sistema "LeerNombreDeUsuario" en el grupo "Administración de usuarios" de la lista de funciones.
9. Seleccione la variable "CurrentUser".



Resultado

Cuando un nuevo usuario inicie la sesión correctamente, se ejecutará la función de sistema "LeerNombreDeUsuario". La variable "CurrentUser" se actualizará y en el campo E/S aparecerá el nuevo usuario que ha iniciado la sesión.

Si un usuario no inicia la sesión correctamente, se cerrará la sesión del usuario que la ha iniciado. En el campo E/S se seguirá visualizando el usuario antiguo, hasta que un nuevo usuario inicie una sesión correctamente.

Consulte también

Campo de aplicación del planificador de tareas (Página 8726)

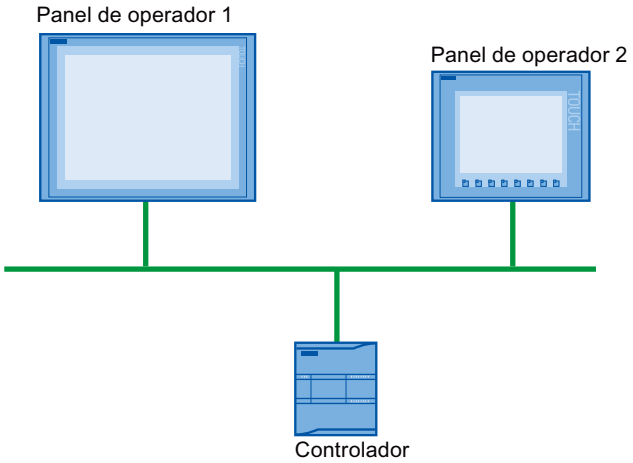
10.8 Comunicación con controladores

10.8.1 Principios básicos de la comunicación

10.8.1.1 Comunicación entre dispositivos

Comunicación

El intercambio de datos entre varios dispositivos se denomina "Comunicación". Los dispositivos pueden estar interconectados a través de una conexión directa o de una red. Los dispositivos interconectados se denominan interlocutores de una comunicación.



Los datos transferidos entre los interlocutores pueden servir para distintos fines:

- representar procesos
- manejar procesos
- visualizar avisos
- administrar parámetros de proceso y de máquina

Interlocutores

En este capítulo se explica la comunicación entre los siguientes dispositivos:

- Controlador
  El controlador controla un proceso por medio de un programa de usuario.
- Panel de operador
  El panel de operador permite manejar y observar el proceso.

Principio básico de toda comunicación

El principio básico de cualquier tipo de comunicación es una configuración de red. En una configuración de red se especifica qué conexión existe entre los dispositivos configurados.

Además, la configuración de una red crea los requisitos necesarios para la comunicación, es decir:

- todos los nodos de una red disponen de direcciones unívocas.
- los nodos llevan a cabo la comunicación con propiedades de transferencia coherentes.

Sistema de automatización

Un sistema de automatización describe las siguientes propiedades:

- el controlador y el panel de operador están interconectados
- la red entre el controlador y el panel de operador está configurada

Comunicación entre paneles de operador

Para la comunicación entre paneles de operador está disponible el protocolo HTTP.

Encontrará más información en la documentación del SIMATIC HMI HTTP Protocol.

Comunicación a través de una interfaz unitaria e independiente del fabricante

Con OPC (Openess Productivity Collaboration), WinCC dispone de una interfaz de software unitaria e independiente del fabricante. Dicha interfaz permite un intercambio de datos estandarizado entre aplicaciones de las áreas de industria, oficinas y fabricación.

Encontrará más información en la documentación de OPC.

10.8.1.2 Dispositivos y redes en el sistema de automatización

Introducción

Para configurar un sistema de automatización, primero hay que configurar, parametrizar e interconectar los diferentes dispositivos.

Tanto los controladores como los paneles de operador se insertan en el proyecto del mismo modo y también se configuran del mismo modo.

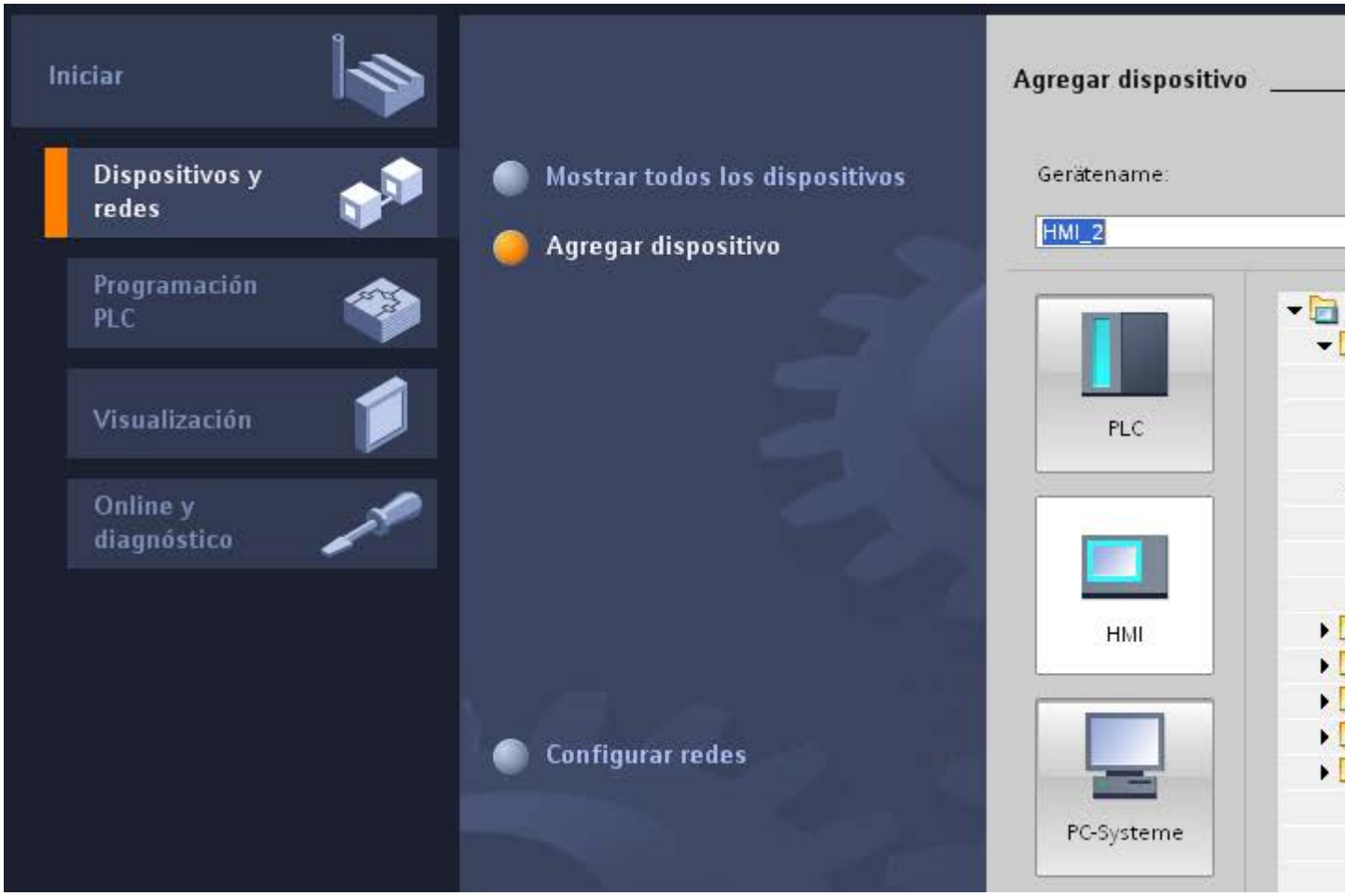
Configurar un sistema de automatización:

1. Insertar el controlador en el proyecto.
2. Insertar el panel de operador en el proyecto.
3. Conectar los dispositivos en red.
4. Interconectar los dispositivos.

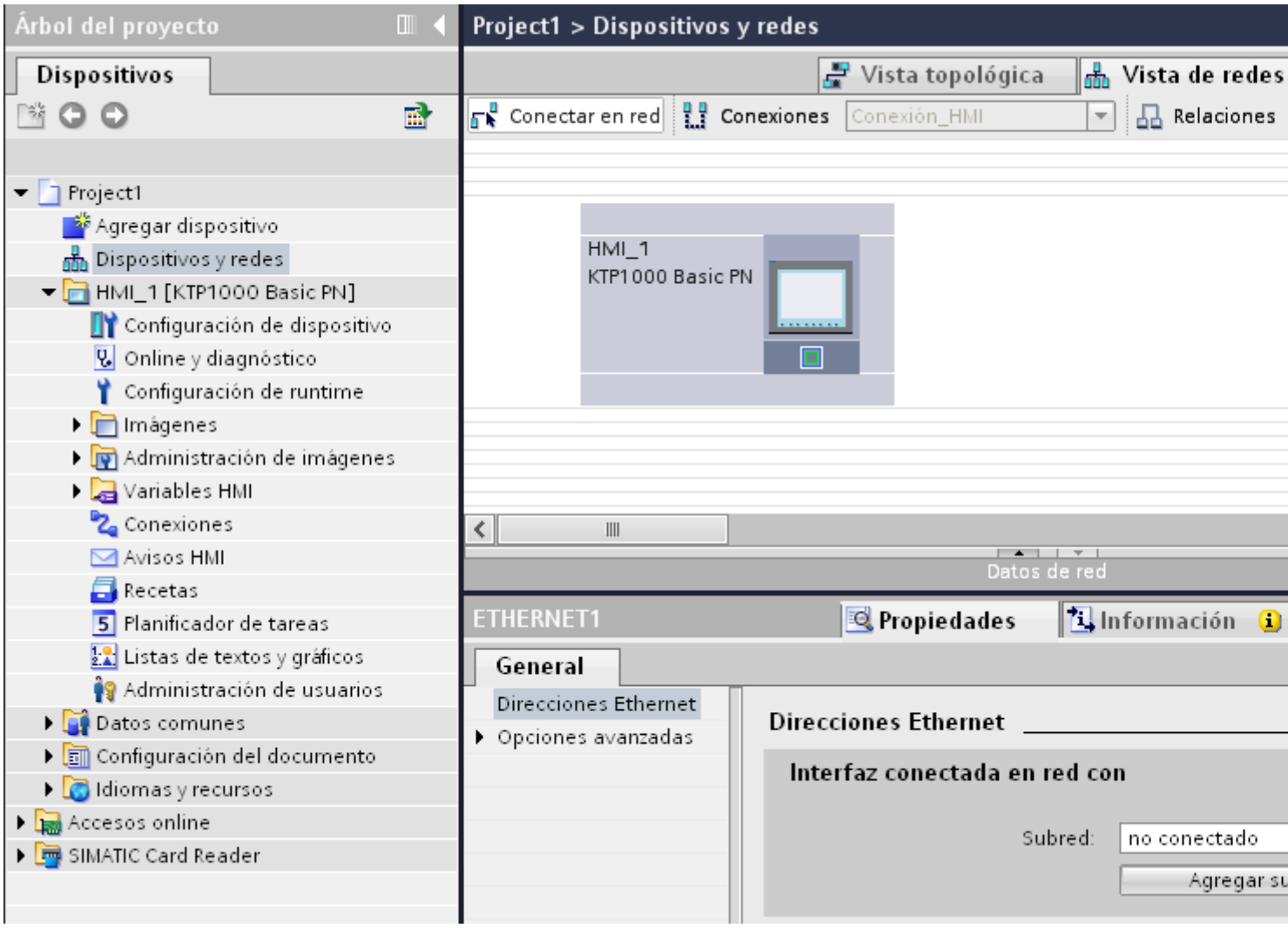
Insertar dispositivos

Una vez se ha creado un proyecto, los dispositivos nuevos pueden agregarse tanto en la vista de portal como en la vista de proyecto.

- Vista del portal

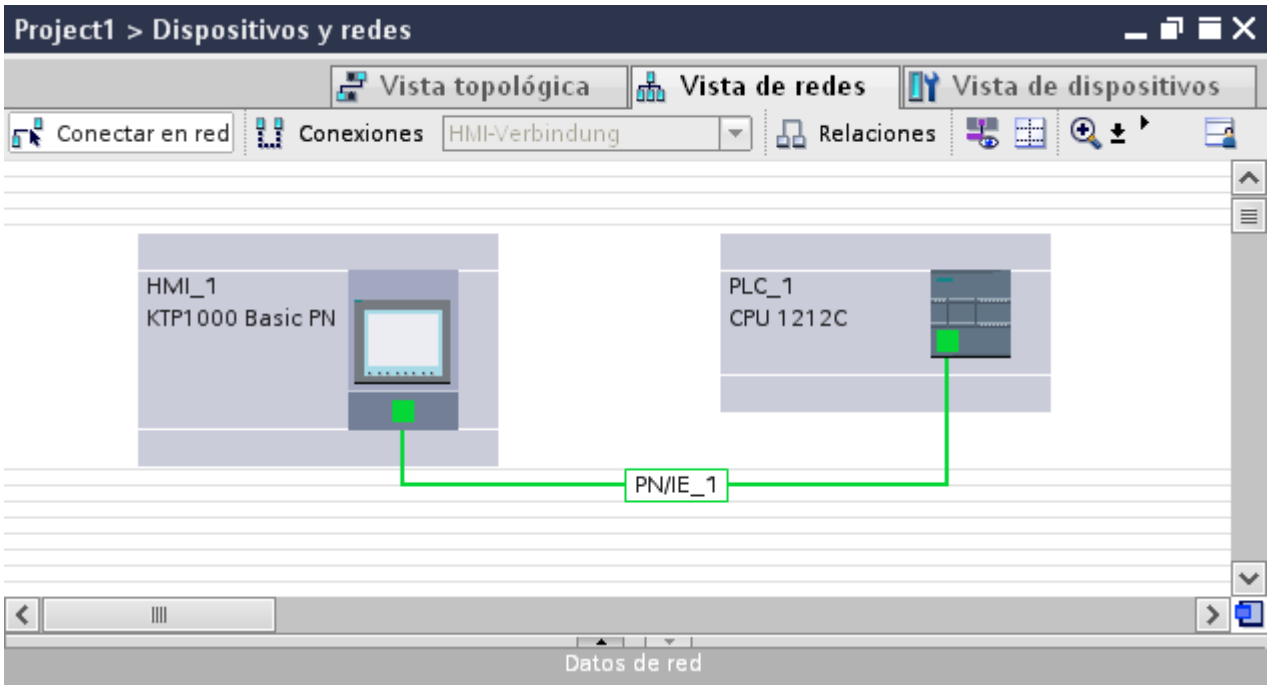


- Vista del proyecto


Árbol del proyecto
Dispositivos
Project1
Agregar dispositivo
Dispositivos y redes
HMI_1 [KTP1000 Basic PN]
Configuración de dispositivo
Online y diagnóstico
Configuración de runtime
Imágenes
Administración de imágenes
Variables HMI
Conexiones
Avisos HMI
Recetas
Planificador de tareas
Listas de textos y gráficos
Administración de usuarios
Datos comunes
Configuración del documento
Idiomas y recursos
Accesos online
SIMATIC Card Reader
Project1 > Dispositivos y redes
Vista topológica
Vista de redes
Conectar en red
Conexiones
Conexión_HMI
Relaciones
HMI_1
KTP1000 Basic PN
Datos de red
ETHERNET1
Propiedades
Información
General
Direcciones Ethernet
Opciones avanzadas
Direcciones Ethernet
Interfaz conectada en red con
Subred:
no conectado

Conectar dispositivos en red

En el editor "Dispositivos y redes" se conectan cómodamente en red las interfaces de los dispositivos aptos para la comunicación. Al realizar la conexión en red se configura la conexión física de los dispositivos.

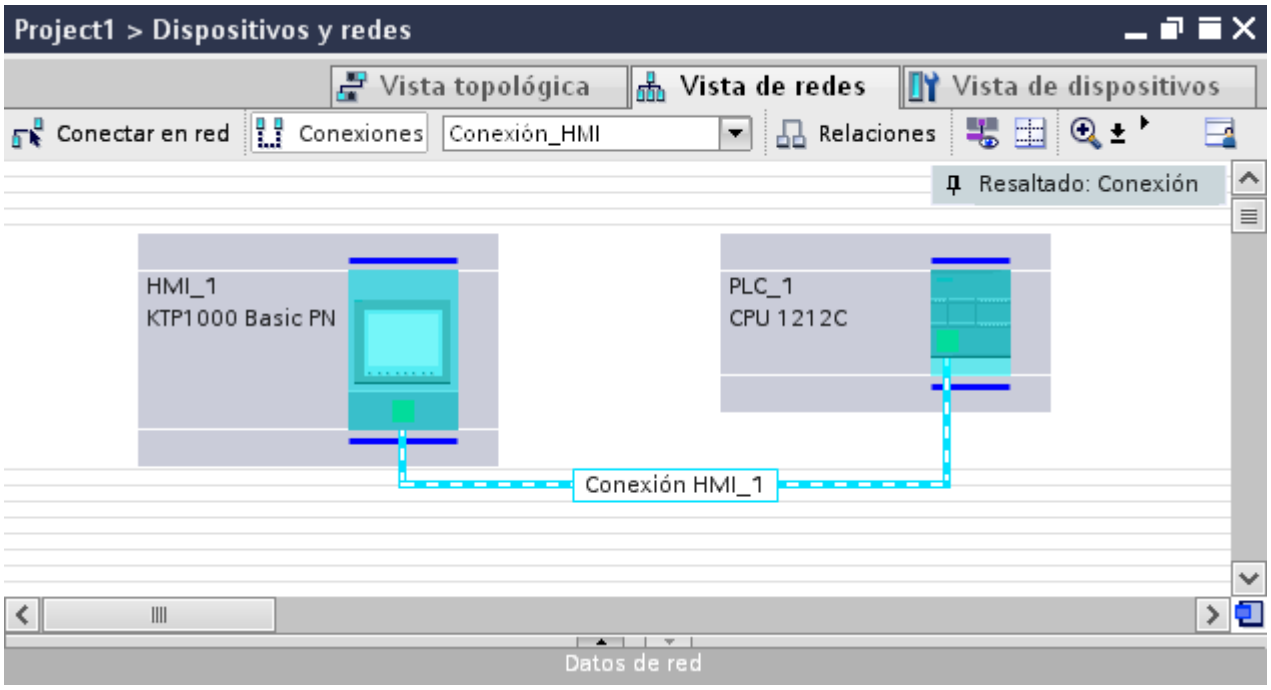


El área de tabla de la vista general de redes complementa la vista gráfica de redes con las funciones siguientes:

- Ofrece información detallada sobre la configuración y parametrización de los dispositivos.
- Desde la columna "Subred" se conectan componentes aptos para la comunicación con subredes creadas.

Conectar dispositivos

Una vez se han conectado los dispositivos en red se configura la conexión. Para una comunicación con el panel de operador se configura el tipo "Conexión HMI".



10.8.1.3 Intercambio de datos mediante variables

Comunicación a través de variables

Los valores de proceso se transfieren en runtime por medio de variables. Los valores de proceso son datos que se guardan en la memoria de un sistema de automatización conectado. Representan el estado de una instalación en forma de temperaturas, niveles de llenado o estados de conexión. Para procesar los valores de proceso en WinCC se definen variables externas.

WinCC trabaja con dos tipos de variables:

- Variables externas
- Variables internas

Trabajar con variables

Encontrará más información sobre la configuración de variables en el capítulo "Trabajar con variables (Página 8465)".

10.8.1.4 Intercambio de datos mediante punteros de área

Comunicación vía punteros de área

Los punteros de área son campos de parámetros. A partir de dichos campos de parámetros, WinCC obtiene información en runtime (tiempo de ejecución del proyecto). Esta información contiene datos sobre la posición y el tamaño de áreas de datos en el controlador.

El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación.

Evaluando los datos almacenados en los rangos de datos, el controlador y el panel de operador ejecutan acciones bien definidas.

Los punteros de área se gestionan centralmente en el editor "Conexiones". Los punteros de área sirven para intercambiar datos de determinadas áreas de datos de usuario.

WinCC utiliza los siguientes punteros de área:

- Registro
- Fecha/hora
- Coordinación
- Petición de control
- Fecha/hora PLC
- Identificación del proyecto
- Número de imagen

Los punteros de área disponibles dependen del panel de operador utilizado.

10.8.1.5 Driver de comunicación

Driver de comunicación

Un driver de comunicación es un componente de software que establece una conexión entre un controlador y un panel de operador. Gracias al driver de comunicación, las variables HMI se abastecen con valores de proceso.

Dependiendo del panel de operador utilizado y de los interlocutores conectados, es posible seleccionar la interfaz utilizada, así como el perfil y la velocidad de transferencia.

10.8.2 Editores para la comunicación

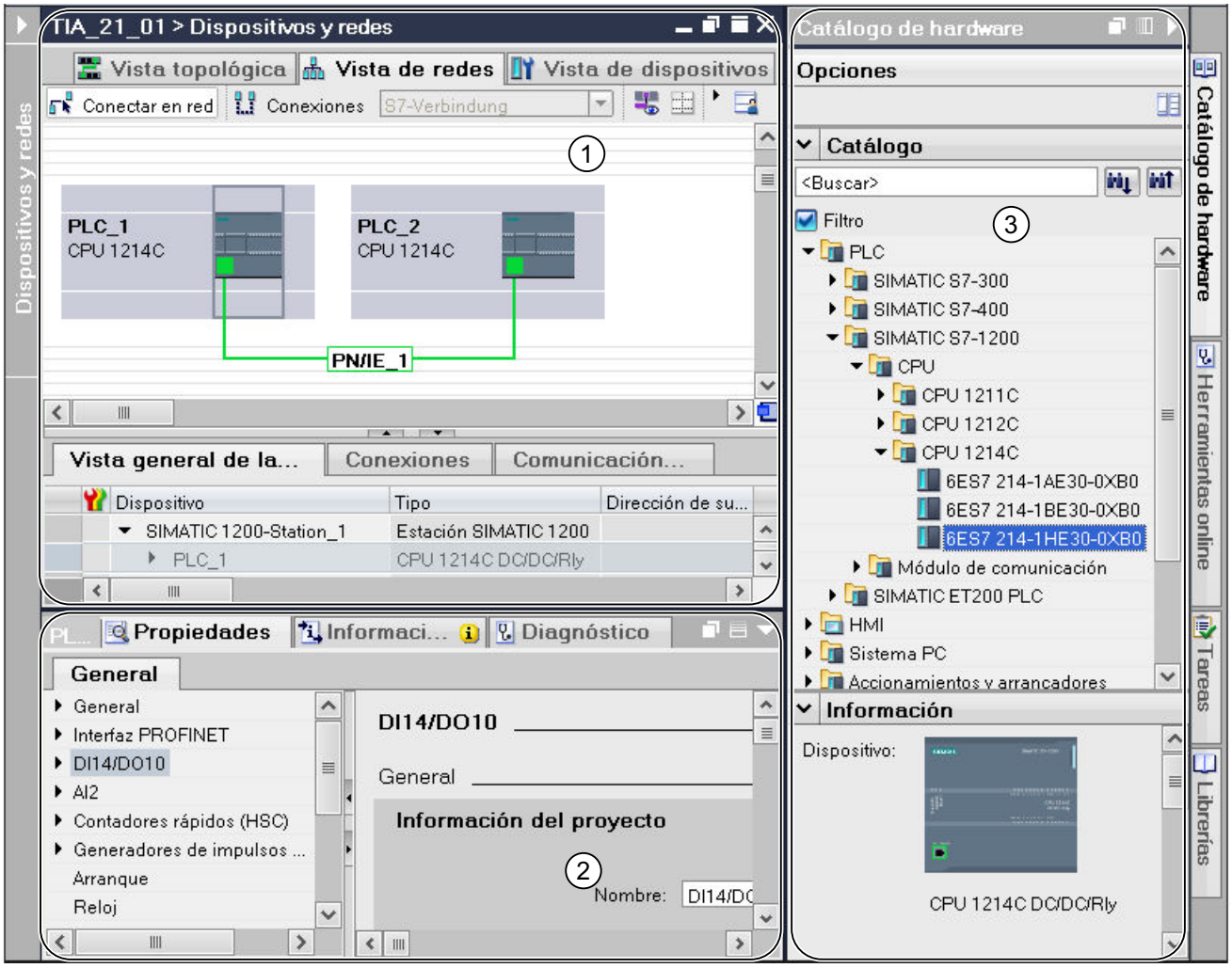
10.8.2.1 Editor "Dispositivos y redes"

Función del editor de hardware y redes

El editor "Dispositivos y redes" es el entorno de desarrollo que se utiliza para conectar en red, configurar y parametrizar dispositivos y módulos.

Configuración

El editor "Dispositivos y redes" incluye los componentes siguientes:



1 Vista de dispositivos, vista de redes y vista topológica
2 Catálogo de hardware
3 Ventana de inspección

El editor "Dispositivos y redes" ofrece tres vistas distintas del proyecto. Es posible cambiar entre estas tres vistas en cualquier momento, en función de si deben crearse y editarse dispositivos y módulos individuales, redes y configuraciones de dispositivos enteras o bien la estructura topológica del proyecto.

La ventana de inspección ofrece información sobre el objeto actualmente seleccionado. También permite modificar los ajustes del objeto seleccionado.

Desde el catálogo de hardware se arrastran los dispositivos y módulos necesarios para el sistema de automatización hasta la vista de dispositivos, redes o topología.

10.8.2.2 Vista de redes

Introducción

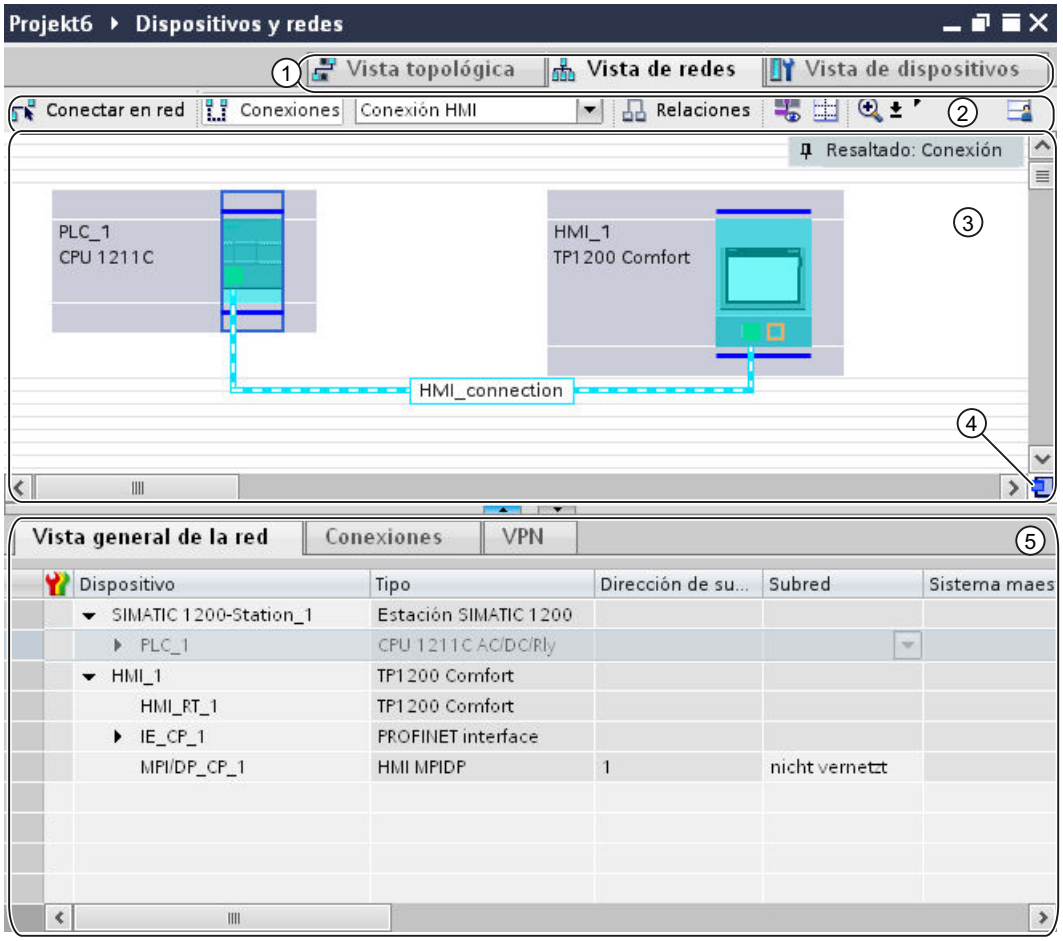
La vista de redes es un área de trabajo del editor de hardware y de redes.

En la vista de redes se realizan las tareas siguientes:

- Configurar y parametrizar dispositivos
- Interconectar dispositivos

Estructura

La figura siguiente muestra los dos componentes de la vista de redes:



1 Conmutador vista de redes/vista de dispositivos/vista topológica

2 Barra de herramientas de la vista de redes

3 Área gráfica de la vista de redes

4 Navegación general

5 Área de tabla de la vista de redes

La división entre el área gráfica y de tabla de la vista de redes se modifica con el ratón.

Para ello, haga clic con el ratón en el borde superior del área de tabla y arrástrela para hacerla más grande o más pequeña manteniendo pulsado el botón del ratón.

Las dos teclas de flecha pequeñas permiten minimizar o maximizar el área de tabla con un solo clic, así como restablecer la última división seleccionada.

Barra de herramientas

La barra de herramientas ofrece las siguientes funciones:

Símbolo	Significado
	Modo de conexión en red de los dispositivos.
	Modo para crear conexiones. El tipo de conexión se ajusta desde la lista desplegable adjunta.
	Modo para crear relaciones.
	Visualización de direcciones de interfaces.
	Ajuste del nivel de zoom. Es posible seleccionar o introducir directamente el nivel de zoom desde la lista desplegable adjunta. El símbolo de zoom permite ampliar o reducir la imagen gradualmente o bien rodear con un borde un área que deba ser ampliada.
	Mostrar saltos de página Activa la vista preliminar del salto de página. Se visualizan líneas punteadas allí donde se insertará un salto de página al imprimir.
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Área gráfica

El área gráfica de la vista de redes muestra los dispositivos relacionados con la red, redes, conexiones y relaciones. Aquí se insertan dispositivos desde el catálogo de hardware, se conectan entre sí mediante las interfaces y se editan los ajustes de comunicación.

Navegación general

Haga clic con el ratón en la navegación general para obtener una vista general de los objetos creados en el área gráfica. Manteniendo pulsado el botón del ratón en la navegación general se navega rápidamente a los objetos deseados y se visualizan en el área gráfica.

Área de tabla

El área de tabla de la vista de redes contiene diversas tablas relacionadas con los dispositivos existentes, las conexiones y los ajustes de comunicación:

- Vista general de redes
- Conexiones
- Comunicación E/S

En el menú contextual de la barra de título de la tabla se puede adaptar la representación de la tabla.

10.8.2.3 Datos de red

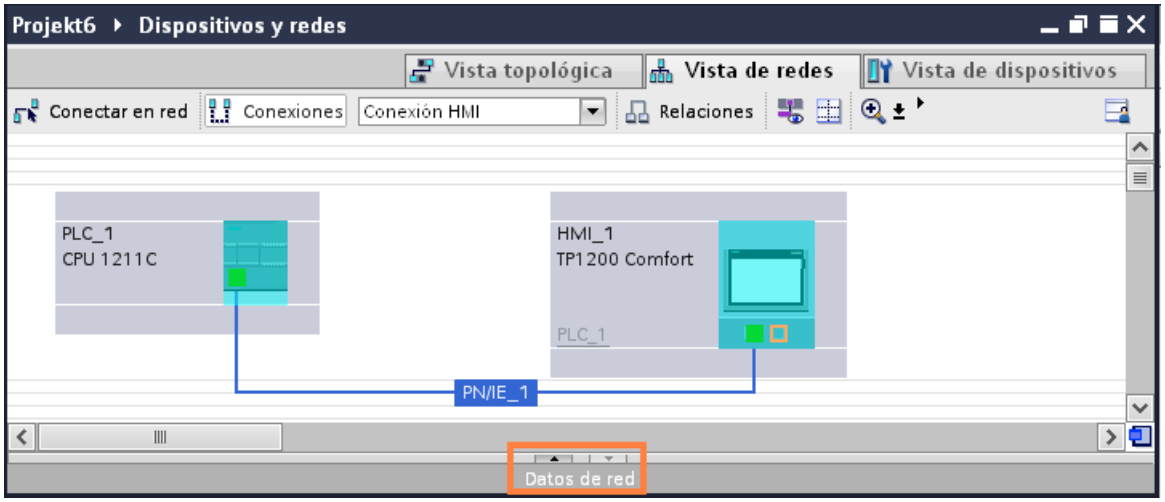
Introducción

El editor "Vista de redes" ofrece, además de la vista de redes gráfica, también una representación en forma de tabla de los "Datos de red".

En el editor "Datos de red" se dispone de las siguientes opciones:

- Vista general de redes
- Conexiones
- VPN
- Comunicación E/S

Los "Datos de red" se abren por debajo de la vista general de redes gráfica.



Funciones básicas

Los datos de red se muestran en formato de tabla y soportan las siguientes funciones básicas para editar una tabla:

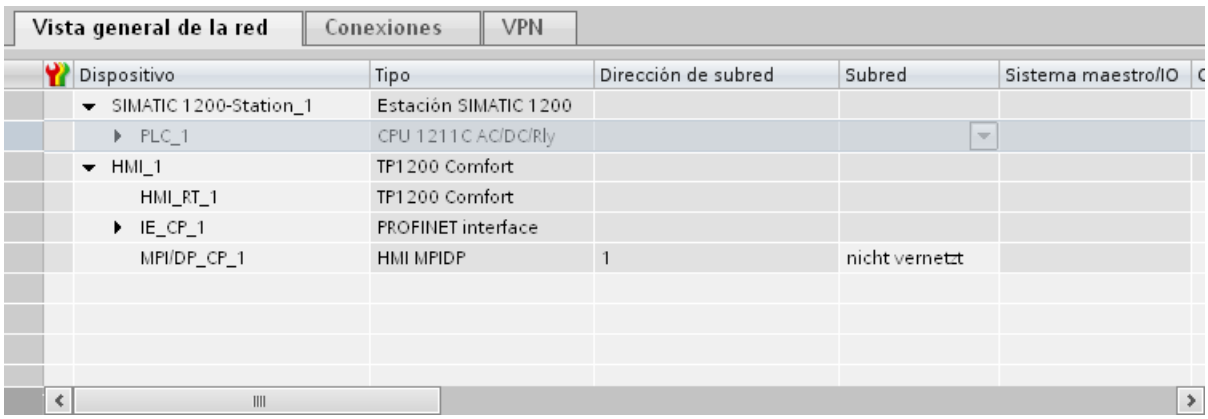
- Mostrar y ocultar las columnas de la tabla
  Nota: no es posible ocultar las columnas relevantes para la configuración.
- Optimizar el ancho de las columnas
- Ordenar la tabla
- Explicar el significado de una columna, fila o campo mediante tooltips

Vista general de redes

El área de tabla de la vista general de redes complementa la vista gráfica de redes con las funciones siguientes:

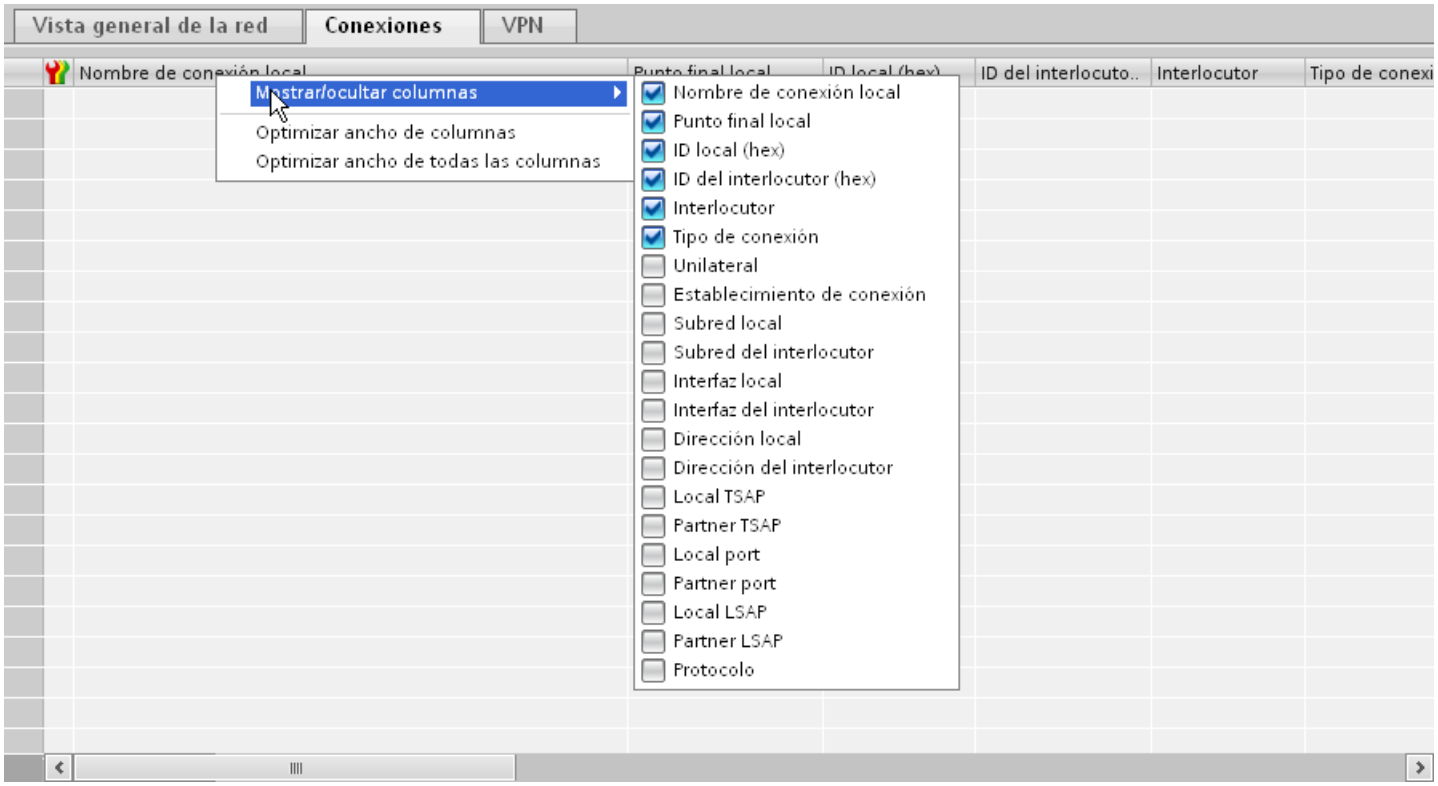
- Ofrece información detallada sobre la estructura y parametrización de los dispositivos.
- Desde la columna "Subred" se conectan componentes aptos para la comunicación con subredes creadas.

Las



Conexiones

En "Conexiones" se pueden consultar otros datos de red.



10.8.2.4 Diagnóstico de conexiones online

Diagnóstico de conexiones

En el TIA Portal puede leer los datos de diagnóstico de las conexiones existentes.

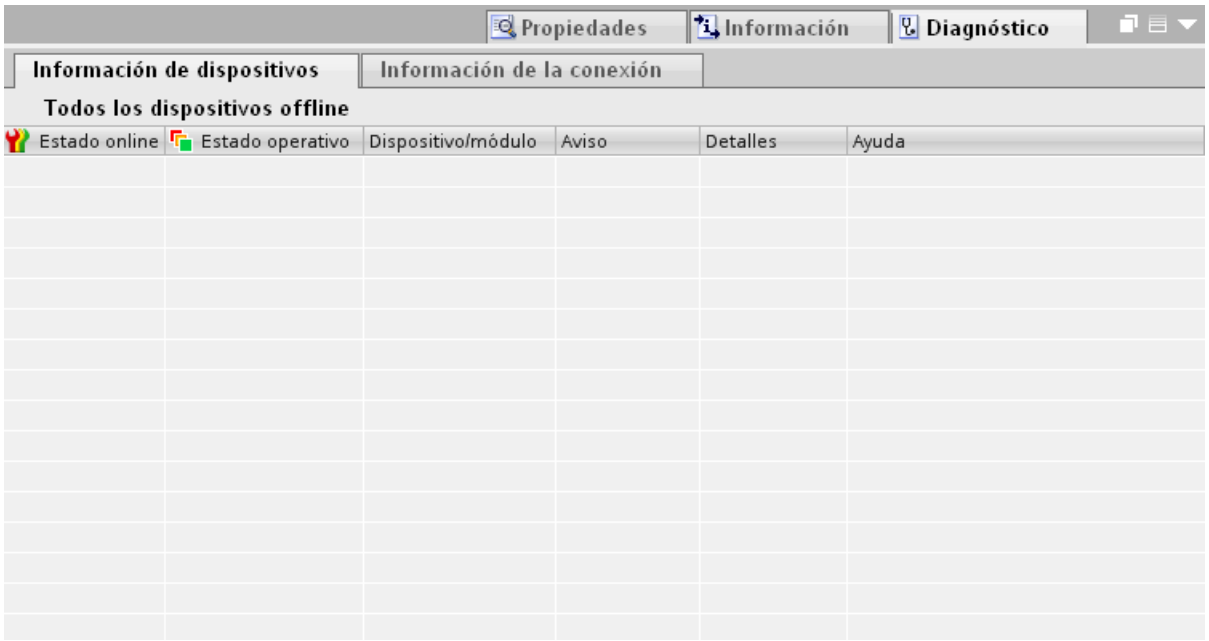
La función "Diagnóstico" de la ventana de inspección muestra los datos de conexión en formato de tabla.

Requisitos

- Los dispositivos deben encontrarse en el modo "Online"

Información del dispositivo

En la ventana de inspección "Diagnóstico > Información del dispositivo" se muestran todos los datos de diagnóstico de los dispositivos que se encuentran en el modo "Online".



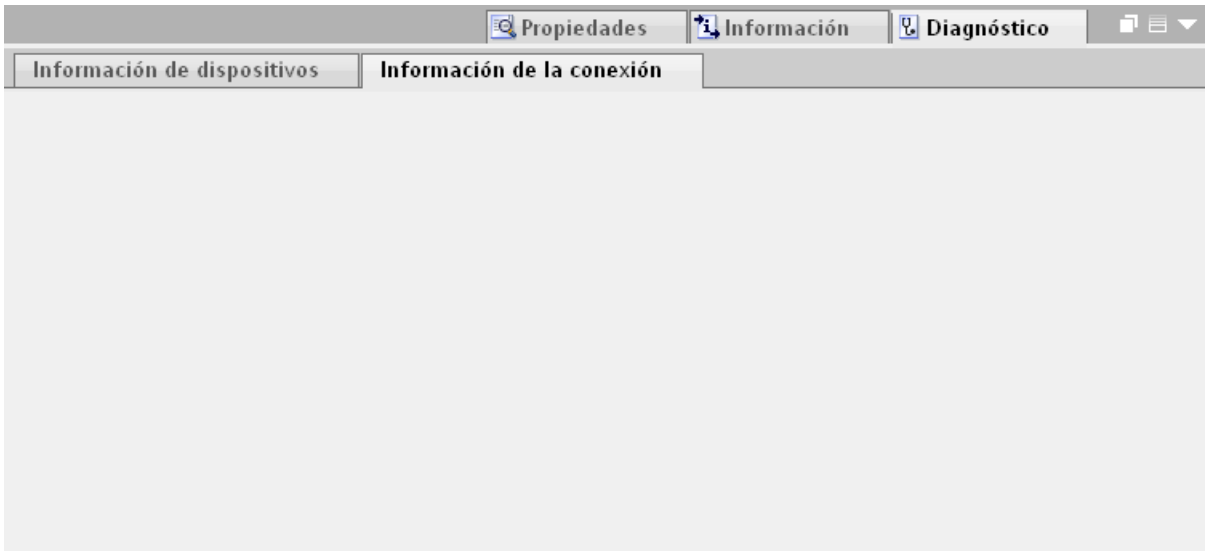
Se muestran los siguientes datos:

- Estado online
- Estado operativo
- Dispositivo/módulo
- Aviso
- Detalles
- Ayuda

Información de conexión

Con la función "Información de conexión" puede visualizar los datos de diagnóstico de la conexión marcada en el editor "Dispositivos y redes".

Un gráfico muestra los interlocutores de la conexión y a través de qué driver de comunicación están conectados entre sí.



Se muestran los siguientes datos:

- Punto final
- Interfaz
- Subred
- Dirección
- TSAP
- Número de los recursos de HMI

10.8.2.5 Vista de dispositivos

Introducción

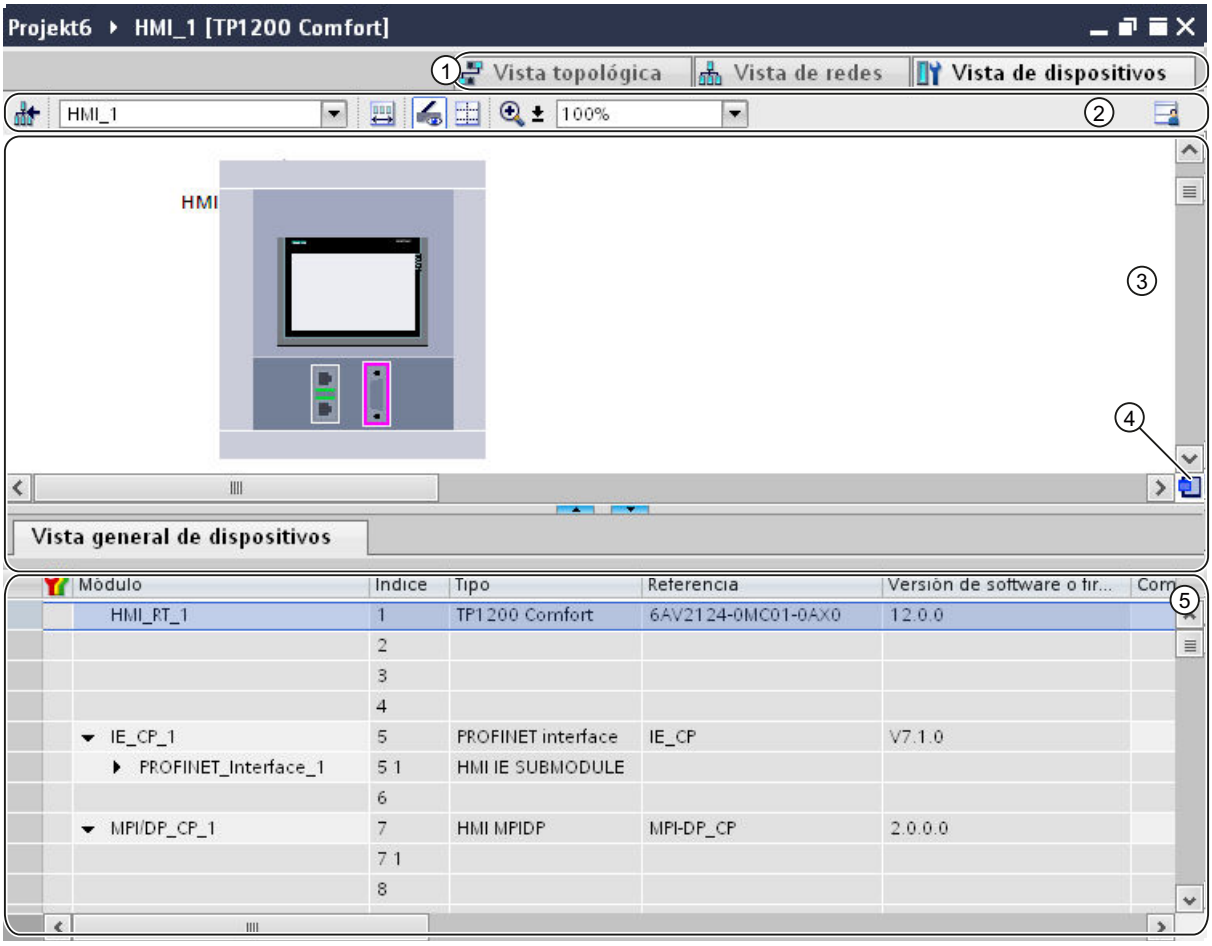
La vista de dispositivos es un área de trabajo del editor de hardware y de redes.

En la vista de dispositivos se realizan las tareas siguientes:

- Configurar y parametrizar dispositivos
- Configurar y parametrizar módulos

Estructura

La figura siguiente muestra los dos componentes de la vista de dispositivos:



1 Conmutador vista de redes/vista de dispositivos/vista topológica

2 Barra de herramientas de la vista de dispositivos

3 Área gráfica de la vista de dispositivos

4 Navegación general

5 Área de tabla de la vista de dispositivos

La división entre el área gráfica y de tabla de la vista de dispositivos se modifica con el ratón.

Para ello, haga clic con el ratón en el borde superior del área de tabla y arrástrela para hacerla más grande o más pequeña manteniendo pulsado el botón del ratón. Las dos teclas de flecha pequeñas permiten minimizar o maximizar el área de tabla con un solo clic, así como restablecer la última división seleccionada.

Barra de herramientas

La barra de herramientas ofrece las siguientes funciones:

Símbolo	Significado
	Cambia a la vista de redes. Nota: en la vista de dispositivos se puede cambiar entre los dispositivos disponibles desde la lista desplegable adjunta.
	Visualización del área de módulos no enchufados
	Visualización de la rotulación de módulos.
	Ajuste del nivel de zoom. Es posible seleccionar o introducir directamente el nivel de zoom en la lista desplegable adjunta. El icono de zoom también permite aumentar o reducir la vista gradualmente o bien trazar un marco alrededor de un área que desea ampliarse. A partir de un nivel de zoom del 200% es posible distinguir las denominaciones de direcciones de los canales E/S en los módulos de señales.
	Mostrar saltos de página Activa la vista preliminar del salto de página. Se visualizan líneas punteadas allí donde se insertará un salto de página al imprimir.
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Área gráfica

El área gráfica de la vista de dispositivos muestra los componentes de hardware y, en su caso, los módulos correspondientes que están asignados unos a otros a través de uno o varios racks. En los dispositivos con racks es posible insertar otros objetos de hardware desde el catálogo de hardware hasta los slots de los racks.

Navegación general

Haga clic con el ratón en la navegación general para obtener una vista general de los objetos creados en el área gráfica. Manteniendo pulsado el botón del ratón en la navegación general se navega rápidamente a los objetos deseados y se visualizan en el área gráfica.

Área de tabla

El área de tabla de la vista de dispositivos ofrece una vista general del hardware utilizado con los principales datos técnicos y organizativos.

En el menú contextual de la barra de título de la tabla se puede adaptar la representación de la tabla.

10.8.2.6 Vista topológica

Introducción

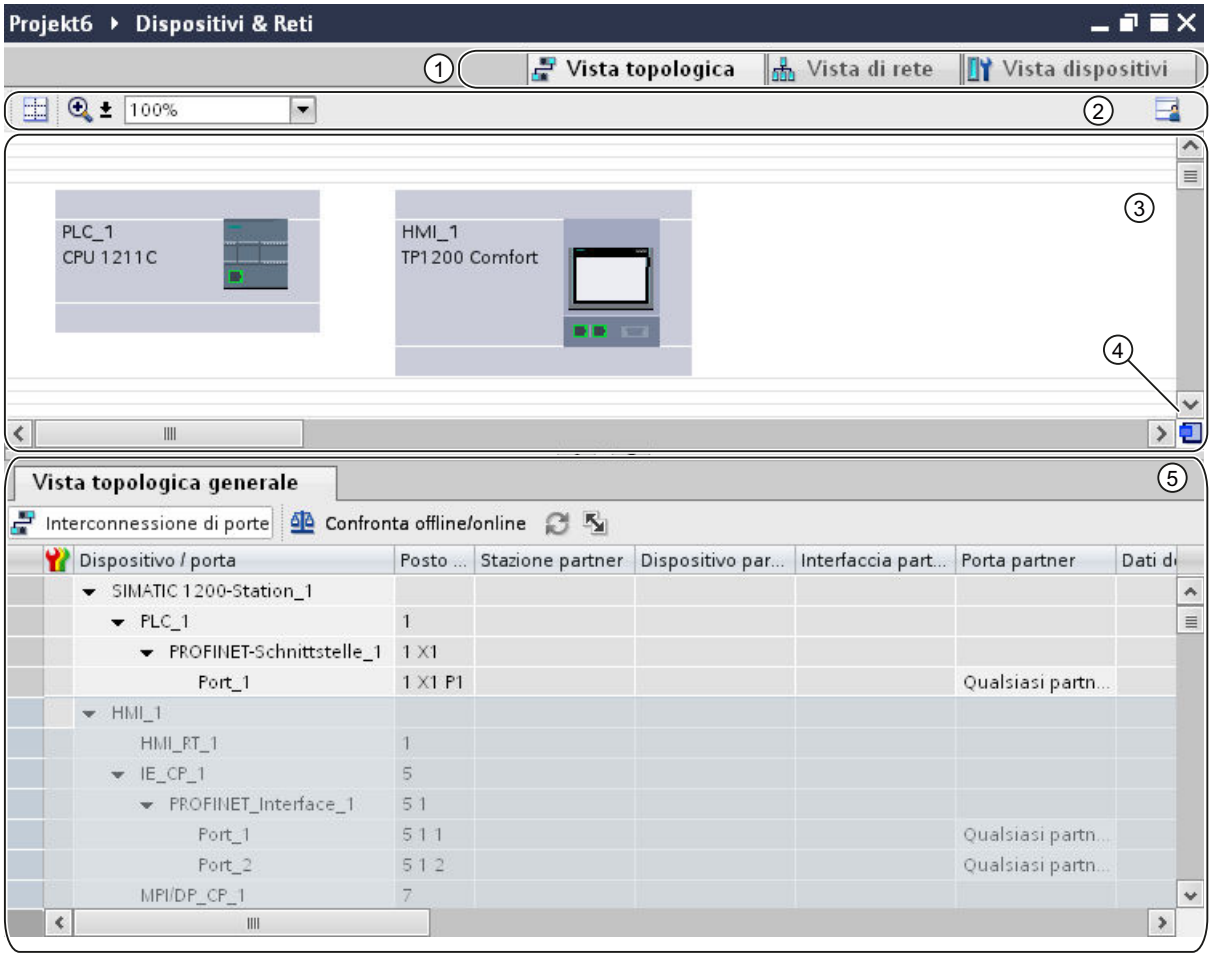
La vista topológica es un área de trabajo del editor de hardware y de redes.

En la vista topológica se realizan las tareas siguientes:

- Mostrar la topología Ethernet
- Configurar la topología Ethernet
- Determinar y minimizar diferencias entre la topología prevista y la topología real

Estructura

La siguiente figura muestra una vista general de la vista topológica.



1 Conmutador vista de dispositivos/vista de redes/vista topológica
2 Barra de herramientas de la vista topológica
3 Área gráfica de la vista topológica
4 Navegación general
5 Área de tabla de la vista topológica

La división entre el área gráfica y de tabla de la vista topológica se modifica con el ratón.

Para ello, haga clic con el ratón en el borde superior del área de tabla y arrástrela para hacerla más grande o más pequeña manteniendo pulsado el botón del ratón. Las dos teclas de flecha

pequeñas permiten minimizar o maximizar el área de tabla con un solo clic, así como restablecer la última división seleccionada.

Barra de herramientas

La barra de herramientas ofrece las siguientes funciones:

Símbolo	Significado
	Ajuste del nivel de zoom. Es posible seleccionar o introducir directamente el nivel de zoom desde la lista desplegable adjunta. El símbolo de zoom permite ampliar o reducir la imagen gradualmente o bien rodear con un borde un área que deba ser ampliada.
	Mostrar saltos de página Activa la vista preliminar del salto de página. Se visualizan líneas punteadas allí donde se insertará un salto de página al imprimir.
	Anotar disposición Guarda la representación actual de la tabla. La disposición, el ancho y la visibilidad de las columnas se almacenan en la vista de tabla.

Área gráfica

En el área gráfica de la vista topológica se visualizan los módulos Ethernet con los correspondientes puertos e interconexiones de puertos. Aquí se pueden agregar otros objetos de hardware con interfaces Ethernet.

Navegación general

Haga clic con el ratón en la navegación general para obtener una vista general de los objetos creados en el área gráfica. Manteniendo pulsado el botón del ratón en la navegación general se navega rápidamente a los objetos deseados y se visualizan en el área gráfica.

Área de tabla

Aquí se muestran los módulos Ethernet y PROFINET cuyos puertos e interconexiones de puertos tienen formato de tabla. Esta tabla corresponde a la tabla de la vista general de la red en la vista de redes.

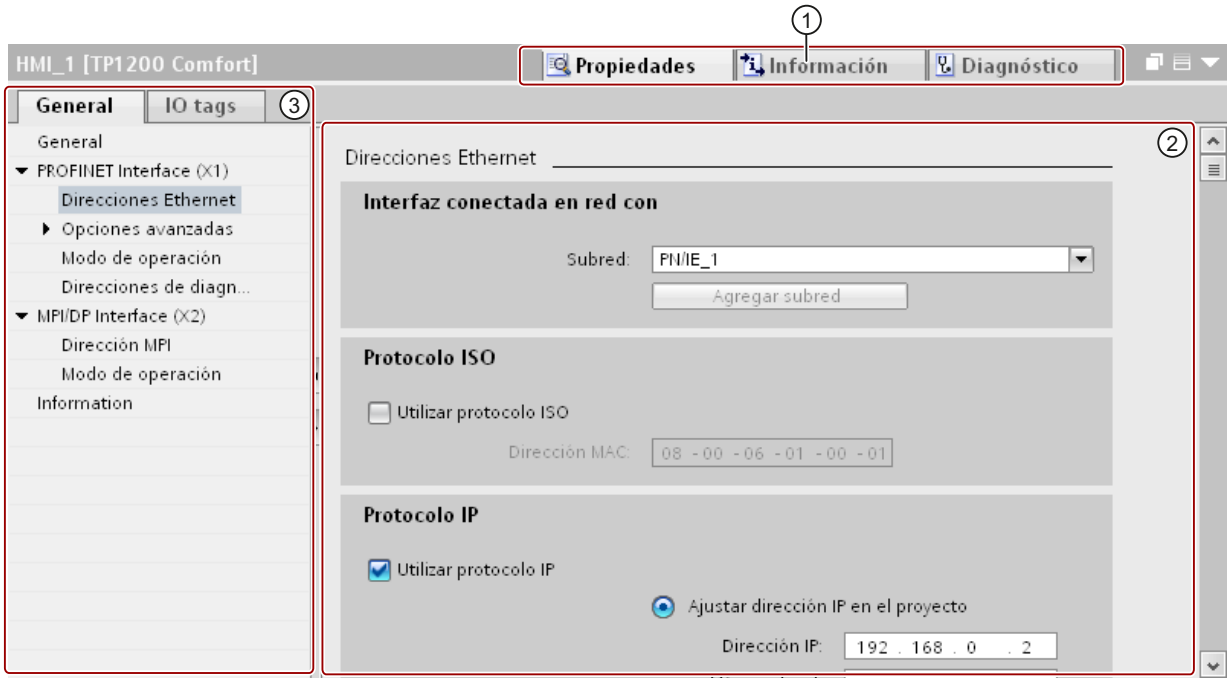
En el menú contextual de la barra de título de la tabla se puede adaptar la representación de la tabla.

10.8.2.7 Ventana de inspección

Las propiedades y los parámetros visualizados del objeto seleccionado se editan dentro de la ventana de inspección.

Estructura

La ventana de inspección incluye los componentes siguientes:



1 Conmutación entre diferentes áreas de información y trabajo

2 Navegación entre distintas informaciones y parámetros

3 Visualización de las informaciones y parámetros seleccionados

Función

La información y los parámetros de la ventana de inspección se dividen en distintos tipos de información:

- Propiedades
- Información
- Diagnóstico

Para visualizar la información y los parámetros correspondientes haga clic en el área deseada. Para configurar un sistema de automatización, el área más importante es la de "Propiedades". Dicha área es la que se muestra por defecto.

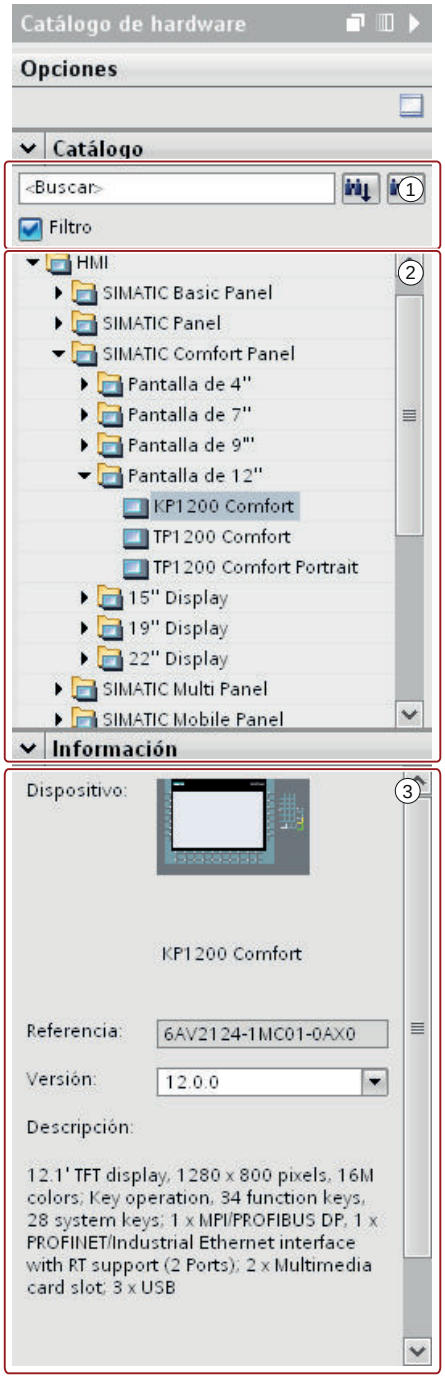
En la parte izquierda de la ventana de inspección se encuentra la navegación local, en la que la información y los parámetros aparecen clasificados en grupos. Si se hace clic en el símbolo de flecha situado a la izquierda del nombre de grupo se abrirá el grupo con los subgrupos existentes. Si se selecciona un grupo o subgrupo, la información y los parámetros correspondientes aparecerán en la parte derecha de la ventana de inspección, donde también pueden editarse.

10.8.2.8 Catálogo de hardware

La Task Card "Catálogo de hardware" permite acceder fácilmente a los componentes de hardware más diversos.

Estructura

La Task Card "Catálogo de hardware" consta de las paletas siguientes:



1 Paleta "Catálogo", función de búsqueda y filtro

2 Paleta "Catálogo", selección de componentes

3 Paleta "Información"

Función de búsqueda y filtro

Las funciones de búsqueda y filtro de la paleta "Catálogo" permiten buscar cómodamente determinados componentes de hardware. La función de filtro permite limitar la visualización de los componentes HW según determinados criterios. Por ejemplo, es posible limitar la visualización a objetos que se pueden colocar también en el contexto actual o que contienen determinadas funciones.

Los objetos utilizables en el contexto actual son, p. ej., objetos que pueden conectarse en red en la vista de redes, o bien los únicos módulos compatibles con el dispositivo en la vista de dispositivos.

Selección de componentes

La selección de componentes de la paleta "Catálogo" contiene los componentes de hardware instalados en una estructura de árbol. Desde allí se arrastran los dispositivos o módulos deseados al área de trabajo gráfica de la vista de dispositivos o redes.

Los componentes de hardware que están instalados pero que todavía no disponen de licencia se visualizan atenuados. Los componentes HW sin licencia no se pueden utilizar.

Los componentes de hardware que pertenecen a diversos grupos de componentes temáticos se presentan parcialmente como objeto enlazado. Si se hace clic en uno de estos componentes de hardware enlazados, se abre la estructura del catálogo donde se encuentran los respectivos componentes de hardware.

Información

En la paleta "Información" aparece información detallada del objeto seleccionado en el catálogo:

- Representación esquemática
- Nombre
- Referencia
- Número de versión
- Descripción

10.8.2.9 Información sobre los componentes de hardware

En el catálogo de hardware existe la posibilidad de visualizar información sobre determinados componentes de hardware en la paleta "Información". También desde el menú contextual es posible visualizar información adicional relacionada con los componentes de hardware seleccionados.

Acceso a información relacionada

Si se selecciona un objeto en el catálogo de hardware y se abre el menú contextual, además de la función "Copiar" hay otras tres opciones disponibles para acceder a información del Service & Support:

- Información de soporte para el producto
- Preguntas frecuentes
- Manuales

La información deseada aparece en el área de trabajo del editor de hardware y redes.

Nota

El acceso al Service & Support sólo se puede utilizar si existe una conexión a Internet y la función está activada. Dicha función está desactivada por defecto.

Para activarla, consulte el capítulo "Activar soporte del producto".

Información de soporte para el producto

Desde aquí se accede a información general relacionada con los componentes de hardware y software. La referencia del objeto de hardware seleccionado está predefinida en la máscara de búsqueda, aunque también es posible buscar otros componentes de hardware y software.

Preguntas frecuentes

Desde aquí se accede a las preguntas frecuentes (Frequently Asked Questions - FAQ), en las que podrá consultar diferentes artículos sobre cuestiones relacionadas con el hardware y software. Una máscara de busca detallada permite filtrar el área temática deseada.

Manuales

Desde aquí se accede a los manuales de los distintos componentes de hardware. Sirve especialmente cuando la configuración, el direccionamiento o la parametrización que debe realizarse requiere conocimientos precisos del hardware utilizado.

10.8.3 Redes y conexiones

10.8.3.1 Redes de comunicación SIMATIC

Redes de comunicación

Sinopsis

Las redes de comunicación son un componente central de las soluciones de automatización modernas. Las redes industriales deben cumplir unos requisitos especiales, p. ej.:

- Acoplamiento de sistemas de automatización así como de sensores, actuadores y ordenadores sencillos.
- La información debe transferirse de forma correcta y en el momento adecuado.
- Resistencia frente a perturbaciones electromagnéticas, cargas mecánicas y ensuciamiento.
- Adaptación flexible a las exigencias de producción.

Las redes industriales están establecidas en el ámbito de LANs (Local Area Networks) y permiten la comunicación en un espacio limitado.

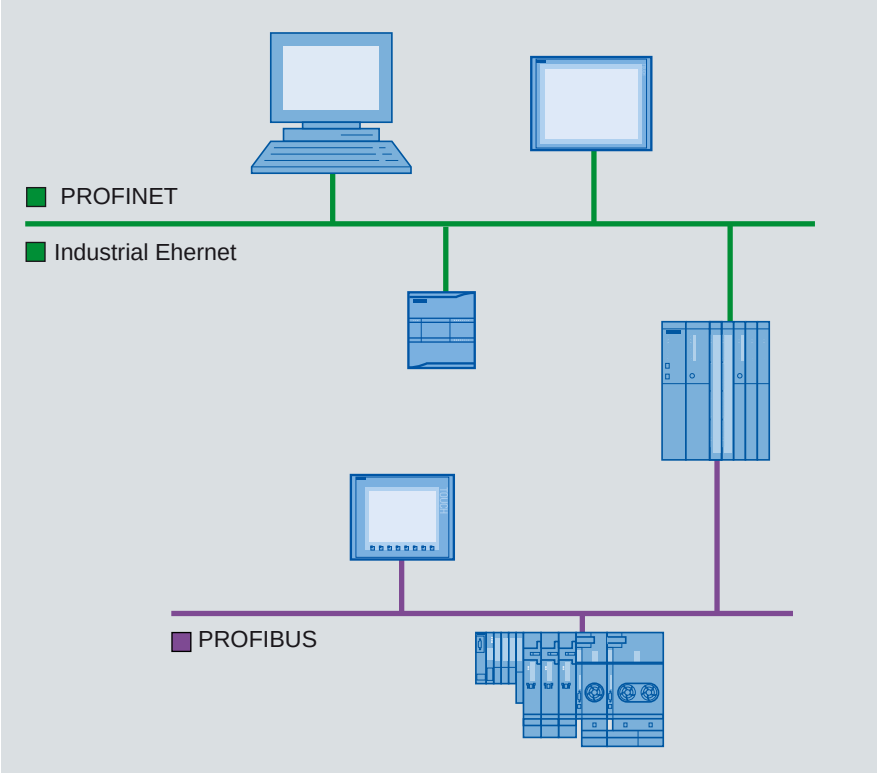
Las redes industriales realizan las siguientes funciones de comunicación:

- Comunicación de procesos y de campo en los sistemas de automatización, inclusive sensores y actuadores
- Comunicación de datos entre sistemas de automatización
- Comunicación TI para integrar la tecnología de la información moderna

Descripción general de las redes

En este capítulo se tratan las redes siguientes:

- Industrial Ethernet
  El estándar de red industrial para todos los niveles
- PROFINET
  El estándar abierto Industrial Ethernet para la automatización
- PROFIBUS
  El estándar internacional para el entorno de campo y el líder del mercado en los buses de campo
- MPI
  La interfaz integrada de los productos SIMATIC
- PPI
  La interfaz integrada especialmente para la S7-200



PROFINET y Ethernet

Industrial Ethernet

Industrial Ethernet se basa en la norma IEEE 802.3 y permite conectar el sistema de automatización con la red de oficinas. Industrial Ethernet ofrece servicios de TI con los que se accede a los datos de producción desde la oficina.

Red Ethernet

En una red Ethernet es posible interconectar todos los dispositivos que están conectados a la red mediante una interfaz Ethernet o un módulo de comunicación. Las posibilidades incluyen la conexión de varios paneles de operador a un controlador SIMATIC S7 y de varios controladores SIMATIC S7 a un panel de operador. El número máximo de interlocutores que pueden conectarse a un panel de operador depende del panel utilizado. Encontrará más información al respecto en la documentación del panel de operador correspondiente.

PROFINET

PROFINET es un estándar abierto conforme a la norma IEEE 61158 para la automatización industrial basada en Industrial Ethernet. PROFINET utiliza los estándares de TI hasta el nivel de campo y permite una ingeniería que abarca toda la instalación.

PROFINET permite llevar a cabo con un elevado rendimiento soluciones de automatización que requieren un tiempo real estricto.

PROFIBUS

PROFIBUS DP

PROFIBUS DP (periferia descentralizada) sirve para conectar los dispositivos siguientes:

- controladores, PCs, paneles de operador
- aparatos de campo descentralizados, p. ej. SIMATIC ET 200
- válvulas
- accionamientos

PROFIBUS DP, con sus rápidos tiempos de reacción, es especialmente adecuado para la industria de producción.

Entre las funcionalidades básicas se incluyen el intercambio cíclico de datos de proceso entre un maestro y los esclavos PROFIBUS DP, así como el diagnóstico.

Red PROFIBUS

Es posible conectar un panel de operador de la red PROFIBUS a módulos SIMATIC S7 que tienen una interfaz PROFIBUS o PROFIBUS DP integrada. Las posibilidades incluyen la conexión de varios paneles de operador a un controlador SIMATIC S7 y de varios controladores SIMATIC S7 a un panel de operador.

El número máximo de interlocutores que pueden conectarse a un panel de operador depende del panel utilizado. Encontrará más información al respecto en la documentación del panel de operador correspondiente.

El controlador SIMATIC S7-200 se configura como nodo pasivo de la red. El SIMATIC S7-200 se conecta mediante un conector DP o un módulo de comunicación PROFIBUS.

MPI

MPI

MPI (Multi-Point Interface, interfaz multipunto) es la interfaz integrada de los productos SIMATIC:

- controladores
- paneles de operador
- PG/PC

Con MPI se crean subredes pequeñas con las propiedades siguientes:

- poca expansión
- pocos nodos
- poco volumen de datos

Red MPI

El panel de operador se conecta a la interfaz MPI del controlador SIMATIC S7. Las posibilidades incluyen la conexión de varios paneles de operador a un controlador SIMATIC S7 y de varios controladores SIMATIC S7 a un panel de operador. El número máximo de interlocutores que pueden conectarse a un panel de operador depende del panel utilizado. Encontrará más información al respecto en la documentación del panel de operador correspondiente.

Arquitecturas de red

MPI se basa en el estándar PROFIBUS (IEC 61158 y EN 50170) y soporta las siguientes topologías de bus:

- línea
- estrella
- árbol

Una subred MPI abarca 127 nodos como máximo y consta de varios segmentos. Un segmento abarca 32 nodos como máximo y está limitado por resistencias terminadoras. Los segmentos se acoplan mediante repetidores. La longitud de línea máxima sin repetidor es de 50 m.

PPI

Introducción

PPI (Point-to-Point Interface) es una interfaz integrada que se ha desarrollado especialmente para el SIMATIC S7-200. Por norma general, una red PPI conecta controladores S7-200. Sin embargo, otros controladores SIMATIC (p. ej. S7-300 y S7-400) o bien paneles de operador también pueden comunicarse con un SIMATIC S7-200 dentro de la red PPI.

Red PPI

Un acoplamiento PPI es una conexión punto a punto. El panel de operador es el maestro. El SIMATIC S7-200 es el esclavo.

Sólo es posible conectar un SIMATIC S7-200 a un panel de operador. El panel de operador se conecta por medio del conector serie de la CPU. Es posible conectar varios paneles de operador a un SIMATIC S7-200. En este caso, desde el SIMATIC S7-200 sólo es posible una conexión a la vez.

Nota

En la red PPI, además del panel de operador puede haber cuatro maestros como máximo. Por motivos de rendimiento se recomienda no conectar más de cuatro nodos a la vez como maestros en la red PPI.

Arquitecturas de red

PPI se basa en el estándar PROFIBUS (IEC 61158 y EN 50170) y soporta las siguientes topologías de bus:

- línea
- estrella

Con PPI se crean redes multimaestro de 32 maestros como máximo:

- El número de maestros que puede comunicarse con cada esclavo no está limitado.
- Un esclavo puede asignarse a varios maestros.

El repetidor RS 485 permite expandir la red PPI. También es posible conectar módems a la red PPI.

10.8.3.2 Configurar redes y conexiones

Conectar dispositivos a una red

Introducción

Para configurar una conexión existe el editor "Dispositivos y redes". En este editor se conectan los dispositivos en una red. Asimismo, se configuran y parametrizan los dispositivos y las interfaces. Seguidamente, se configuran las conexiones necesarias entre los dispositivos conectados a la red.

En el editor "Dispositivos y redes" se configuran conexiones HMI con los siguientes controladores:

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

Las conexiones HMI con otros controladores se configuran en el editor "Conexiones" del panel de operador correspondiente.

Conectar dispositivos en red

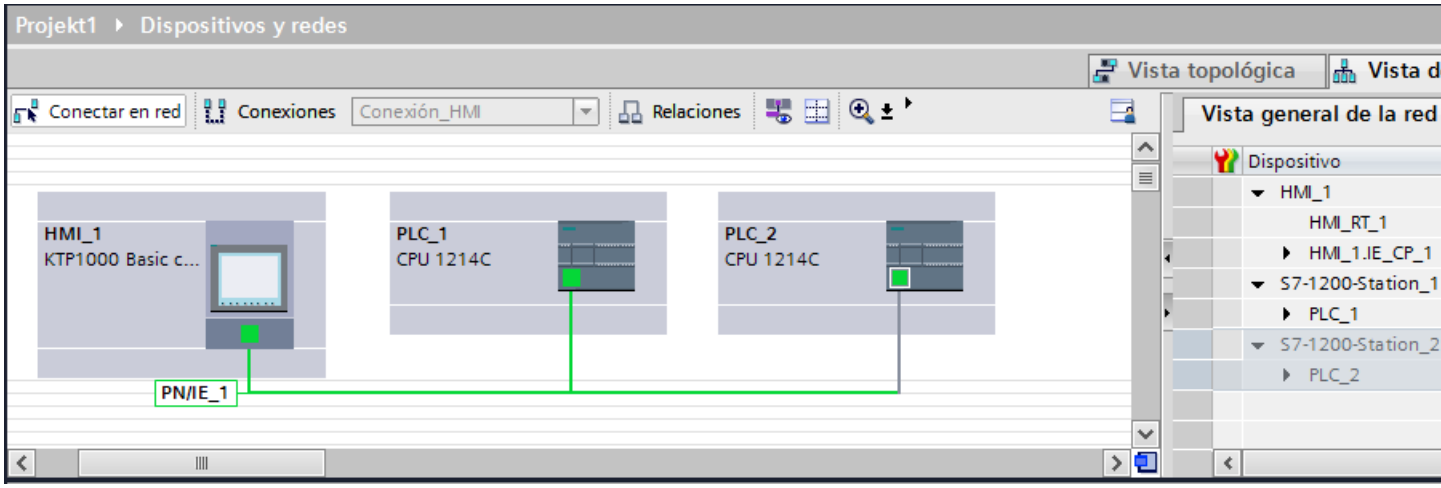
En la vista de redes del editor "Dispositivos y redes" hay disponibles un área gráfica y un área en forma de tabla para la conexión en red. En el área gráfica, los dispositivos incluidos en el proyecto se conectan con la función Drag&Drop. En el área de tabla se presenta una vista general de los dispositivos y sus componentes.

El editor "Dispositivos y redes" permite conectar a red los siguientes controladores y paneles de operador:

- SIMATIC S7 1500
- SIMATIC S7 1200

- SIMATIC S7 300
- SIMATIC S7 400

Los demás controladores están disponibles en el TIA Portal y se configuran como "No integrado". Las conexiones "no integradas" se configuran en el editor "Conexiones" del panel de operador.



Al realizar la conexión en red se configura la conexión física de los interlocutores. La conexión en red de los dispositivos se representa con líneas de colores, según sea la interfaz.

Configurar una conexión integrada en el editor "Dispositivos y redes"

Introducción

Una conexión HMI entre un panel de operador y un SIMATIC S7 se configura en el editor "Dispositivos y redes". Dicha conexión HMI es la conexión directa entre los interlocutores que se han creado en un proyecto.

Conexiones integradas

Las conexiones de dispositivos que están dentro de un proyecto se denominan conexiones integradas. Las conexiones integradas permiten configurar directamente las direcciones de variables de un controlador.

Nota

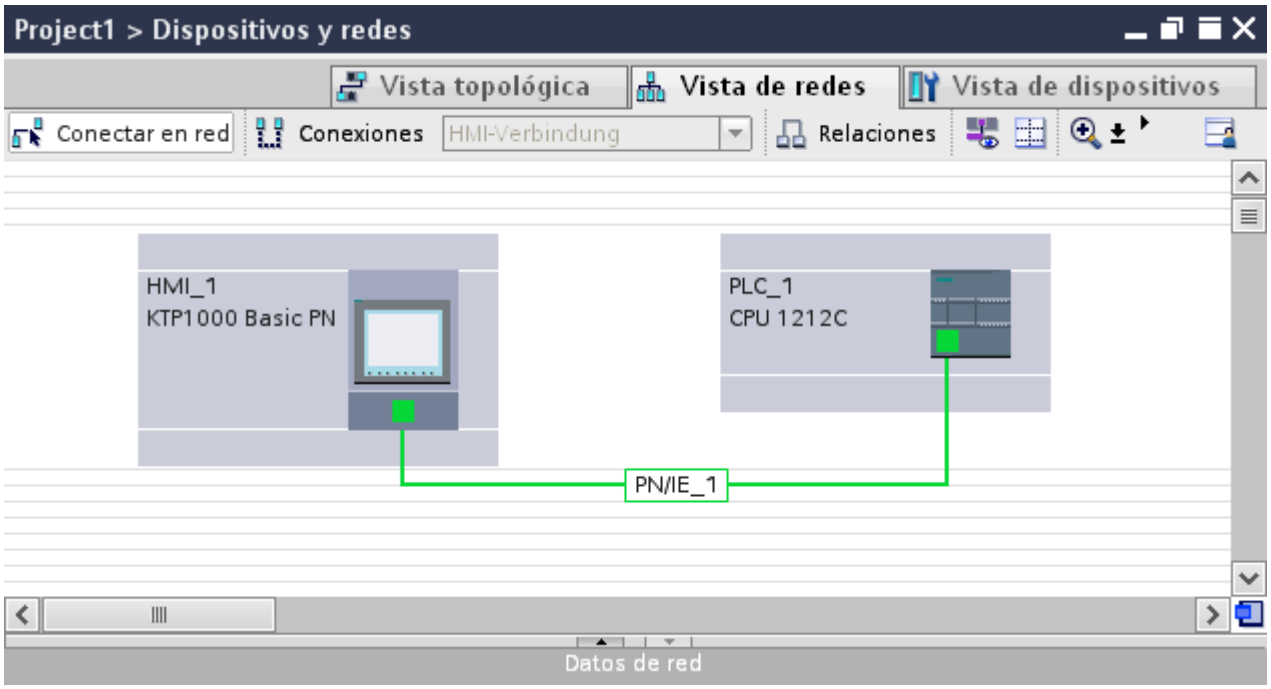
Una conexión HMI sólo se puede configurar en el editor "Dispositivos y redes" para los controladores siguientes:

- SIMATIC S7 1500
- SIMATIC S7 1200
- SIMATIC S7 300
- SIMATIC S7 400

Las conexiones HMI con todos los demás controladores se configuran en el editor "Conexiones" del panel de operador.

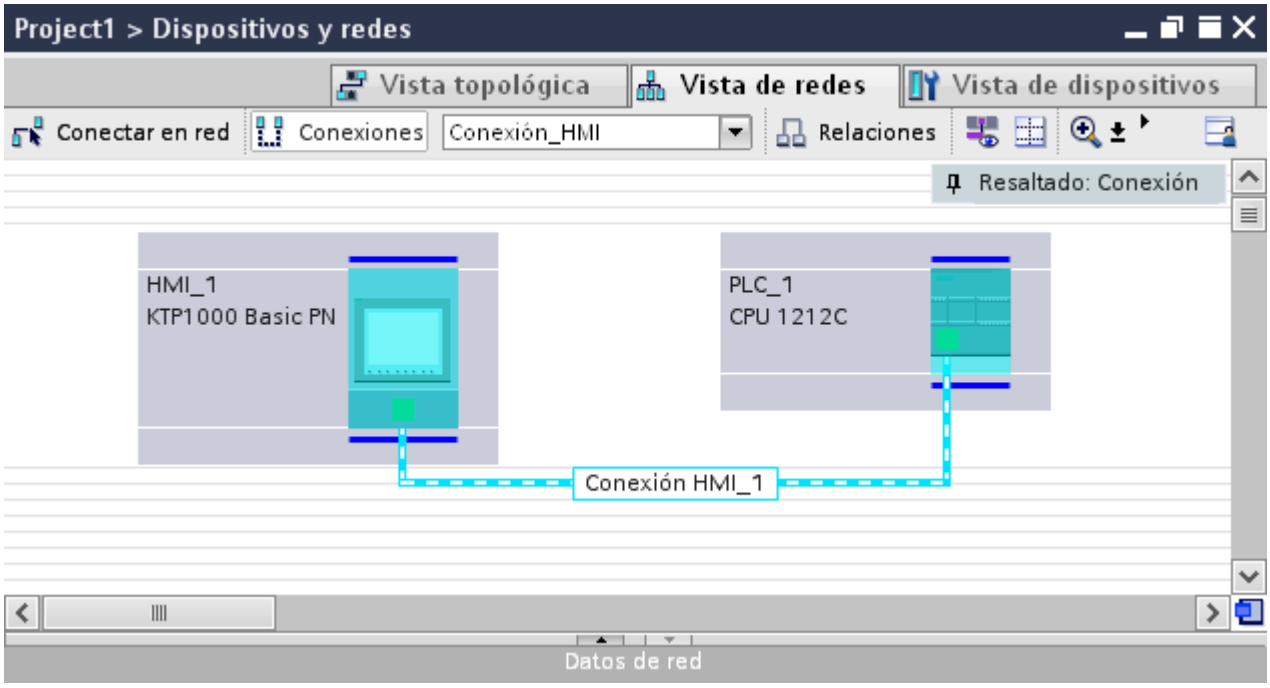
Configurar una conexión HMI en el editor "Dispositivos y redes"

1. Inserte un panel de operador y un controlador SIMATIC S7 en el proyecto.



2. Cambie al modo "Conexiones".
3. Seleccione el tipo "Conexión HMI".

4. Interconecte las dos interfaces PROFINET con la función Drag&Drop.



5. Modifique los parámetros de dirección "Dirección IP" y "Máscara de subred" de acuerdo con los requisitos del proyecto.

Particularidades del editor "Dispositivos y redes"

Introducción

Cuando configure redes o conexiones HMI o ya las haya configurado, le apoyará el editor "Dispositivos y redes" con las siguientes funciones:

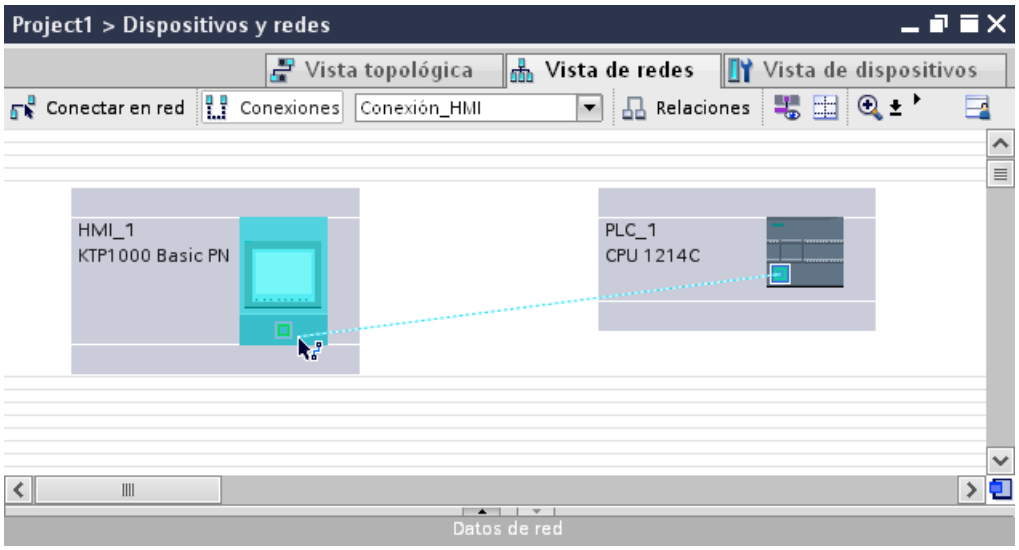
- Resaltar interlocutores
- Resaltar conexiones HMI
- Creación automática de subredes

Resaltar interlocutores

Si ha seleccionado el tipo de conexión "Conexión HMI", todos los interlocutores en los que sea posible una conexión HMI se resaltarán en colores turquesa.

A partir de la interfaz de un dispositivo cree una conexión HMI con la interfaz de otro dispositivo mediante Drag&Drop. Al utilizar la función Drag&Drop todos los interlocutores potenciales se resaltan en colores turquesa.

La conexión de interfaces mediante Drag&Drop se interrumpe con la tecla ESC.



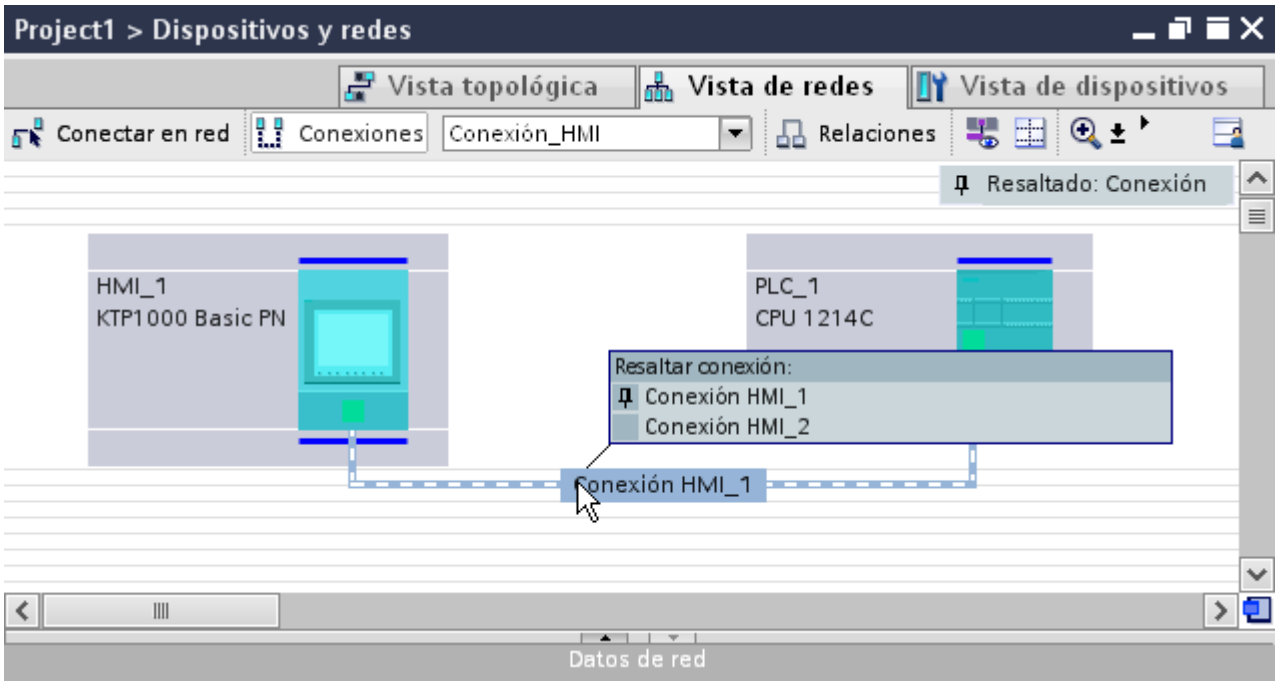
Los símbolos siguientes señalizan si una conexión es posible al pasar el puntero del ratón sobre la interfaz de un dispositivo:

La conexión es posible.

La conexión no es posible.

Resaltar conexiones HMI

Una conexión resaltada en color turquesa señala que se ha creado una conexión HMI. Si se han creado varias conexiones HMI, en un cuadro de diálogo se puede seleccionar una de las conexiones HMI ya creadas.



Después se pueden configurar en la ventana de inspección los parámetros de la conexión HMI seleccionada y del interlocutor.

Subredes

Las subredes se crean o utilizan automáticamente si se dan los siguientes requisitos:

- Si ambos interlocutores no están ya conectados entre sí en diferentes redes.
- Si ambos interlocutores disponen de una interfaz libre.
- Si ya existe una subred, se utiliza automáticamente esta subred para la conexión HMI.

Configurar una conexión no integrada en el editor "Conexiones"

Introducción

Una conexión entre el panel de operador y un controlador que no pueda crearse en el editor "Dispositivos y redes" debe configurarse en el editor "Conexiones" del panel de operador.

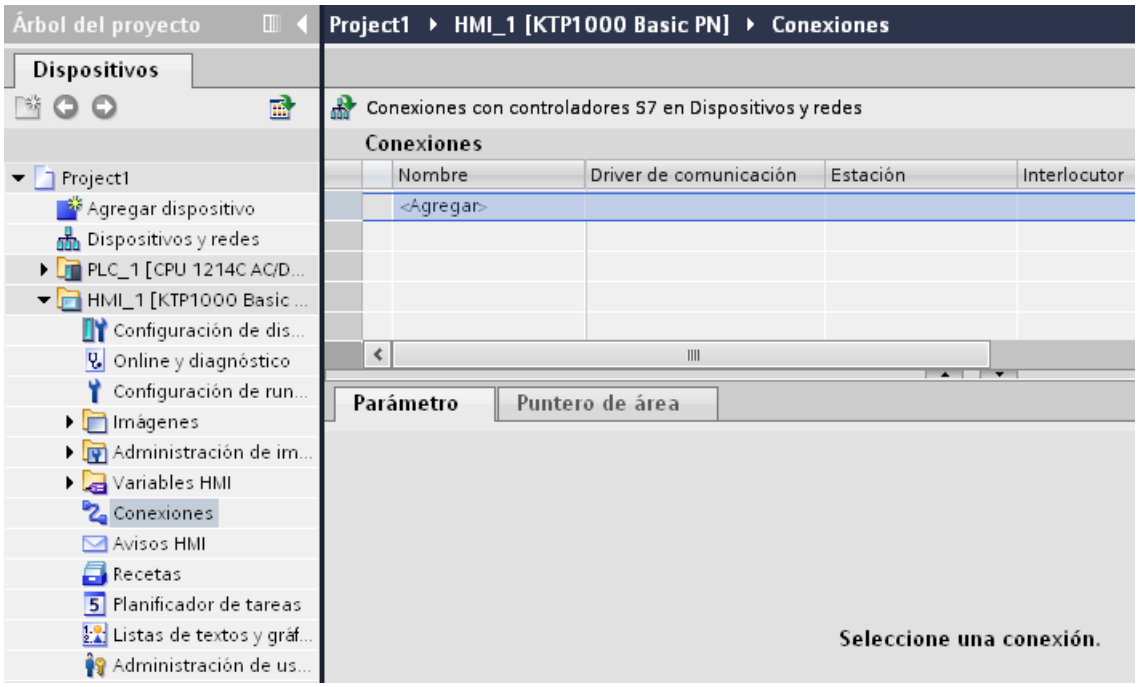
Dichas conexiones se denominan conexiones no integradas.

Requisitos

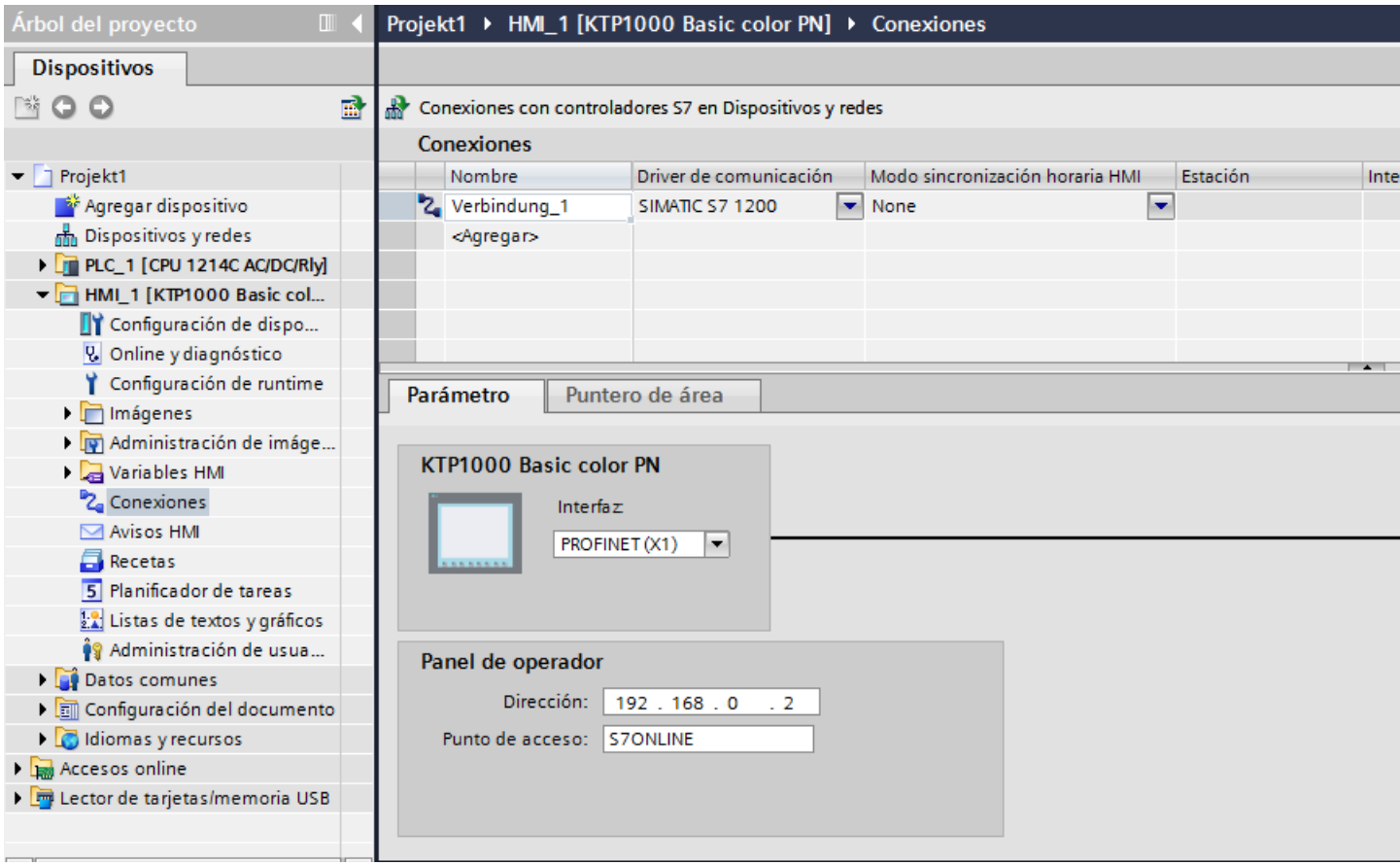
- Hay un proyecto abierto.
- Se ha creado un panel de operador.

Configurar una conexión en el editor "Conexiones"

1. Abra el editor "Conexiones" del panel de operador.
2. Cree una conexión nueva.

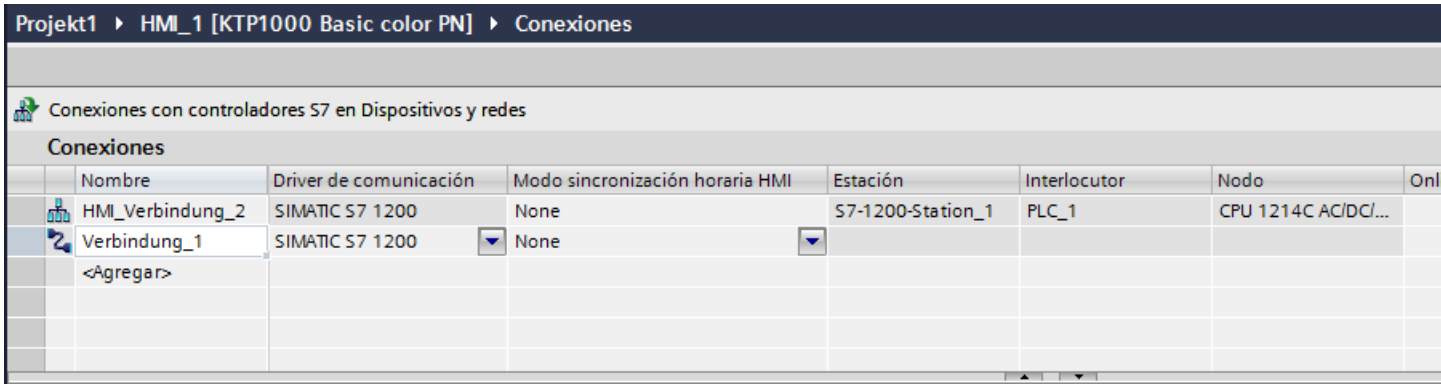


3. Seleccione el "driver de comunicación".
4. Ajuste los parámetros de la conexión.



Conexiones integradas en el editor "Conexiones"

Si ya se han configurado conexiones integradas del panel de operador en el editor "Dispositivos y redes", éstas también se mostrarán en el editor "Conexiones".



Significado de los iconos utilizados:

Conexión integrada

Conexión no integrada

10.8.4 Intercambio de datos

10.8.4.1 Intercambio de datos mediante variables

Principios básicos de las variables

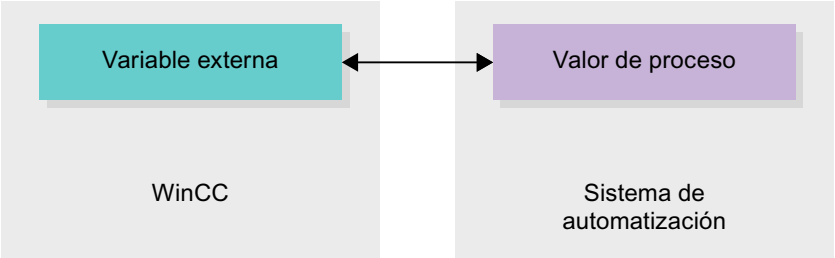
Introducción

Los valores de proceso se transfieren en runtime por medio de variables. Los valores de proceso son datos que se almacenan en la memoria de uno de los sistemas de automatización conectados. Representan el estado de una instalación en forma de temperaturas, niveles de llenado o estados de conexión. Para la edición de los valores de proceso en WinCC se definen variables externas.

WinCC trabaja con dos tipos de variables:

- Variables externas
- Variables internas

Las variables externas constituyen el vínculo entre el WinCC y los sistemas de automatización. Los valores de variables externas se corresponden con los valores de proceso de la memoria de un sistema de automatización. El valor de una variable externa se determina mediante la lectura del valor de proceso de la memoria del sistema de automatización. A la inversa, también se puede reescribir un valor de proceso en la memoria del sistema de automatización.



Las variables internas no poseen ninguna conexión a proceso y sólo transportan valores dentro de WinCC. Los valores de las variables solo están disponibles mientras se ejecuta el runtime.

Variables en WinCC

En el caso de las variables externas, en las propiedades de la variable correspondiente se determina a través de qué conexión se comunica WinCC con el sistema de automatización y cómo se realiza el intercambio de datos.

Las variables que no reciben datos del proceso se denominan variables internas y no están conectadas con el sistema de automatización. Esto se reconoce por la propiedad "Conexión" de la variable, en la entrada "Variable interna".

Para tener una visión más clara, cree diferentes tablas de variables. En la navegación del proyecto, acceda directamente a las distintas tablas de variables en el nodo "Variables HMI". Con ayuda de la tabla "Mostrar todas las variables" se pueden visualizar las variables de todas las tablas.

En una estructura es posible agrupar diversas variables, constituyendo así una unidad lógica. Las estructuras son datos pertenecientes al proyecto y se encuentran disponibles para todos los equipos HMI del mismo. Para crear y editar una estructura utilice el editor "Tipos" de la librería del proyecto.

Resumen de las tablas de variables HMI

Introducción

Las tablas de variables HMI contienen las definiciones de las variables HMI válidas para el dispositivo. Para cada dispositivo HMI que se crea en el proyecto se genera automáticamente una tabla de variables.

En el árbol del proyecto hay una carpeta "Variables HMI" para cada dispositivo HMI. En ella pueden encontrarse las siguientes tablas:

- Tabla de variables estándar
- Tablas de variables personalizadas
- Tabla de todas las variables

En el árbol del proyecto se pueden crear otras tablas variables en la carpeta Variables HMI para ordenar y agrupar las variables y constantes. Las variables se pueden mover a otras tablas de variables con la función Drag&Drop o con ayuda del campo "Tabla de variables". El campo "Tabla de variables" se activa a través del menú contextual de los títulos de columna.

Tabla de variables estándar

Hay una tabla de variables estándar para cada dispositivo HMI del proyecto. Ésta no puede borrarse ni moverse. La tabla de variantes estándar contiene variables HMI y también variables de sistema dependientes del dispositivo HMI. Todas las variables HMI pueden declararse en la tabla de variables estándar, o se pueden crear otras tablas de variables personalizadas según necesidad.

Tablas de variables personalizadas

Para cada dispositivo HMI pueden crearse varias tablas de variables personalizadas para agrupar variables según las necesidades. A las tablas de variables personalizadas se les puede cambiar el nombre, pueden reunirse en grupos o borrarse. Para la agrupación de tablas de variables, cree otras subcarpetas en la carpeta Variables HMI.

Todas las variables

La tabla "Todas las variables" muestra una vista general de todas las variables HMI y variables de sistema del dispositivo HMI correspondiente. Esta tabla no puede borrarse ni moverse, ni es posible cambiar su nombre. Además, esta tabla contiene la columna "Tabla de variables", que muestra en qué tabla de variables está contenida una variable. A través del campo "Tabla de variables" se puede modificar la asignación de una variable a una tabla de variables.

En los dispositivos para Runtime Professional la tabla "Todas las variables" contiene otra ficha "Variables de sistema". Las variables de sistema son creadas por el sistema y se utilizan para la gestión interna del proyecto. Los nombres de las variables de sistema comienzan con el carácter "@". Las variables de sistema no se pueden borrar ni cambiar de nombre. El valor de una variable de sistema se pueden evaluar, pero no modificar.

Tablas adicionales

En una tabla de variables HMI se encuentran disponibles las siguientes tablas adicionales:

- Avisos de bit
- Avisos analógicos
- Variables de fichero

Con ayuda de estas tablas se configuran avisos y variables de fichero para la variable HMI actualmente seleccionada.

Tabla avisos de bit

En la tabla "Avisos de bit" se asocian avisos de bit a la variable HMI seleccionada en la tabla de variables HMI. Si se configura un aviso de bit, no es posible realizar una selección múltiple en la tabla de variables HMI. Configure los avisos de bit de cada variable HMI por separado.

Tabla Avisos analógicos

En la tabla "Avisos analógicos" se asocian avisos analógicos a la variable HMI seleccionada en la tabla de variables HMI. Si asocia un aviso analógico no es posible una selección múltiple en la tabla de variables HMI. Proyecte los avisos analógicos de cada variable HMI por separado.

Tabla Variables de fichero

En la tabla "Variables de fichero" se asocian variables de fichero a la variable HMI seleccionada en la tabla de variables HMI. Si asocia una variable de fichero no es posible una selección múltiple en la tabla de variables HMI. Proyecte todas las variables de fichero de cada variable HMI por separado. La tabla "Variables de fichero" sólo está disponible si el dispositivo HMI empleado soporta el archivado.
Si se emplea WinCC Runtime Professional, pueden asignarse también varias variables de fichero a una variable. En los demás dispositivos HMI sólo se puede asignar una variable de fichero a una variable.

Variables externas

Introducción

Las variables externas hacen posible la comunicación, es decir, el intercambio de datos entre los componentes de un sistema de automatización, p. ej. entre el panel de operador y el controlador.

Principio

Una variable externa es la imagen de una posición de memoria definida en el autómata. Es posible acceder con derechos de lectura y escritura a esta posición de memoria tanto desde el panel de operador como desde el controlador.

Como las variables externas son la imagen de una posición de memoria del controlador, los tipos de datos que se pueden utilizar dependerán del controlador al que esté conectado el panel de operador.

Si se escribe un programa de control para el PLC en STEP 7, las variables del PLC creadas en el programa de control se registrarán en la tabla de variables del PLC. Si desea conectar una variable externa con una variable del PLC, acceda directamente a las variables del PLC desde la tabla de variables y conéctelas con la variable externa.

Tipos de datos

En una variable externa el usuario tiene disponible en WinCC todos los tipos de datos que, a su vez, estén disponibles en el controlador conectado. En la documentación de los drivers de comunicación correspondientes encontrará información sobre los tipos de datos disponibles para la conexión a otros controles.

Encontrará más información al respecto en "Auto-Hotspot".

Nota

Además de las variables externas, para la comunicación entre el panel de operador y el controlador hay disponibles punteros de área. Los punteros de área se pueden configurar y activar en el editor "Conexiones".

Sincronización con variables PLC

En "Configuración de variables" de los ajustes de runtime se ofrecen distintas opciones para sincronizar variables externas con las variables PLC.

Al sincronizar, se pueden aplicar automáticamente los nombres de variables del PLC a variables externas, así como reconectar las variables existentes.

El nombre de variable generado se deriva de la posición del valor de datos en la estructura jerárquica del bloque de datos.

Actualización de valores de variables

En las variables externas, los valores de variable actuales se transmiten en runtime a través del enlace de comunicación entre el WinCC y los sistemas de automatización conectados y se almacenan en la memoria de runtime. A continuación, el valor de la variable se actualiza conforme al tiempo de ciclo configurado. Para la aplicación en el proyecto runtime, WinCC accede a valores de variable de la memoria de runtime leídos del controlador en el instante anterior del ciclo. De este modo se puede modificar ya el valor en el controlador mientras se edita el valor de la memoria de runtime.

Nota

Elementos de matrices PLC en combinación con S7 1200 o S7 1500

El índice de los elementos de matrices PLC puede empezar con cualquier número. En WinCC, la indexación siempre empieza con 0.

Una variable PLC "Array[1..3] of Int" se representa en WinCC, p. ej., con "Array[0..2] of Int".

Si accede a una matriz en un script, asegúrese de que la indexación sea correcta.

Direccionamiento de variables externas

Introducción

Las posibilidades de direccionamiento de variables externas dependen del tipo de conexión entre WinCC y el controlador correspondiente. Se deben diferenciar los siguientes tipos de conexión:

- Conexión integrada
  Las conexiones de dispositivos que se encuentran dentro de un proyecto y se han elaborado con el editor "Dispositivos y redes" se denominan conexiones integradas.
- Conexión no integrada
  Las conexiones de dispositivos elaboradas con el editor "Conexiones" se denominan conexiones no integradas. No todos los dispositivos se deben encontrar dentro de un proyecto.

El tipo de conexión se reconoce por su símbolo.

	Conexión integrada
	Conexión no integrada

Encontrará más información al respecto en el capítulo Principios básicos de la comunicación (Página 8734).

Direccionamiento en conexión integrada

Una conexión integrada ofrece la ventaja de que una variable se puede direccionar tanto de forma simbólica como absoluta.

En un direccionamiento simbólico se selecciona la variable PLC por su nombre para conectarla con la variable HMI. El sistema selecciona automáticamente el tipo de datos válido para la

variable HMI. Al direccionar elementos en bloques de datos deben distinguirse los casos siguientes:

Direccionamiento simbólico de bloques de datos con acceso optimizado y acceso estándar:

Al direccionar simbólicamente un bloque de datos con acceso optimizado y con acceso estándar, se asigna dinámicamente la dirección de un elemento en el bloque de datos y, en caso de modificar la dirección, el cambio se aplica automáticamente en la variable HMI. Para ello no es necesario compilar el bloque de datos conectado o bien el proyecto de WinCC. En los bloques de datos con acceso optimizado sólo está disponible el direccionamiento simbólico.

Al direccionar simbólicamente los elementos de un bloque de datos, el proyecto de WinCC debe compilarse y cargarse nuevamente sólo si se realizan los cambios siguientes:

- Si se han cambiado el nombre o el tipo de datos del elemento de bloque de datos conectado o de la variable PLC global.
- Si se han cambiado el nombre o el tipo de datos de un nodo de estructura de orden superior contenido en el elemento de bloque de datos conectado o la variable PLC global del elemento conectado.
- Si ha cambiado el número del bloque de datos conectado.

Actualmente se dispone de direccionamiento simbólico con los siguientes controladores:

- SIMATIC S7 1200
- SIMATIC S7 1500

Además, se dispone del direccionamiento simbólico en una conexión integrada.

Nota

Si utiliza el puntero de área "Fecha/hora", no es posible el direccionamiento simbólico.

El direccionamiento absoluto también se puede utilizar en una conexión integrada. Las variables PLC de un controlador SIMATIC S7 300/400 sólo se pueden direccionar de modo absoluto. Si ha conectado una variable HMI con una variable PLC y la dirección de la variable PLC cambia, hay que volver a compilar el programa de control para que se actualice la nueva dirección en WinCC. A continuación compile de nuevo el proyecto de WinCC y cárguelo en el panel de operador.

En WinCC, el direccionamiento simbólico está predeterminado como configuración básica. Para modificar el ajuste estándar, elija el comando de menú "Herramientas > Configuración". En el cuadro de diálogo "Configuración" seleccione "Visualización > Variables". Si es necesario, desactive la opción "Acceso simbólico".

La disponibilidad de una conexión integrada depende del controlador empleado. La tabla siguiente muestra la disponibilidad:

Controlador	Conexión integrada	Observaciones
S7 300/400	Sí	El enlace de las variables no se comprueba en runtime. Si se modifica la dirección de variable en el controlador y no se compila y carga de nuevo el dispositivo HMI, no se registra el cambio en runtime.
S7 1200	Sí	En el direccionamiento simbólico se realiza una comprobación de validez de la conexión de variables en runtime. Si se produce un cambio de dirección en el controlador, dicho cambio se registra y se emite un mensaje de error. En el caso de direccionamiento absoluto se aplica el procedimiento descrito para el S7 300/400.
S7 1500	Sí	En el direccionamiento simbólico se realiza una comprobación de validez de la conexión de variables en runtime. Si se produce un cambio de dirección en el controlador, dicho cambio se registra y se emite un mensaje de error. En el caso de direccionamiento absoluto se aplica el procedimiento descrito para el S7 300/400.

Una conexión integrada se crea en el editor "Dispositivos y redes". Si el controlador está incluido en el proyecto y soporta las conexiones integradas, también se puede generar la conexión de forma automática. Para ello sólo tiene que seleccionar durante la configuración de la variable HMI una variable PLC existente con la que desee conectar la variable HMI. Entonces, el sistema genera de forma automática la conexión integrada.

Direccionamiento en conexión no integrada

En un proyecto con conexión no integrada siempre se proyecta una conexión de variables con direccionamiento absoluto. Usted mismo debe seleccionar el tipo de datos válido. Si en el transcurso de un proyecto con conexión no integrada cambia la dirección de una variable PLC, debe realizar asimismo la modificación en WinCC. No se puede comprobar la validez de la conexión de variables en runtime; no se produce ningún mensaje de error.

Una conexión no integrada está disponible para todos los controladores soportados.

El direccionamiento simbólico no está disponible en una conexión no integrada.

En una conexión no integrada, el programa de control no debe formar parte del proyecto WinCC. La configuración del controlador y del proyecto de WinCC puede llevarse a cabo de forma independiente. Para la configuración en WinCC se deben conocer simplemente las direcciones empleadas en el controlador y su función.

Consulte también

Principios básicos de la comunicación (Página 8734)

Las variables internas

Introducción

Las variables internas no tienen conexión con el controlador. Las variables internas transportan valores dentro del panel de operador. Los valores de las variables solo están disponibles mientras se ejecuta el runtime.

Principio

Las variables internas se almacenan en la memoria del panel de operador. De esta manera, sólo se puede acceder a ellas con derechos de lectura y escritura desde este panel de operador. Las variables internas se crean, p. ej., para realizar cálculos locales.

Las variables internas disponen de los tipos de datos HMI. La disponibilidad depende del panel de operador utilizado.

Están disponibles los tipos de datos HMI siguientes:

Tipo de datos HMI	Formato de datos
Matriz	Matriz unidimensional
Bool	Variable binaria
DateTime	Formato fecha/hora
DInt	Valor de 32 bits con signo
Int	Valor de 16 bits con signo
LReal	Coma flotante de 64 bits IEEE 754
Real	Coma flotante de 32 bits IEEE 754
SInt	Valor de 8 bits con signo
UDnt	Valor de 32 bits sin signo
UInt	Valor de 16 bits sin signo
USInt	Valor de 8 bits sin signo
WString	Variable de texto de juego de caracteres de 16 bits

10.8.4.2 Intercambio de datos mediante punteros de área

Principios básicos de los punteros de área

Introducción

A través de un puntero de área se accede a un área de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichas áreas durante la comunicación. Al evaluar los datos almacenados, el controlador y el panel de operador ejecutan acciones fijamente definidas.

Los punteros de área son necesarios con los datos siguientes, p. ej.:

- recetas
- peticiones del controlador
- vigilancia de señal de vida

Punteros de área

Se soportan los siguientes punteros de área:

Puntero de área

Los punteros de área pueden configurarse para conexiones.

- Registro
- Fecha/hora
- Coordinación
- Petición del controlador

Punteros de área globales del panel de operador

Los punteros de área globales solo pueden configurarse en una conexión para cada proyecto.

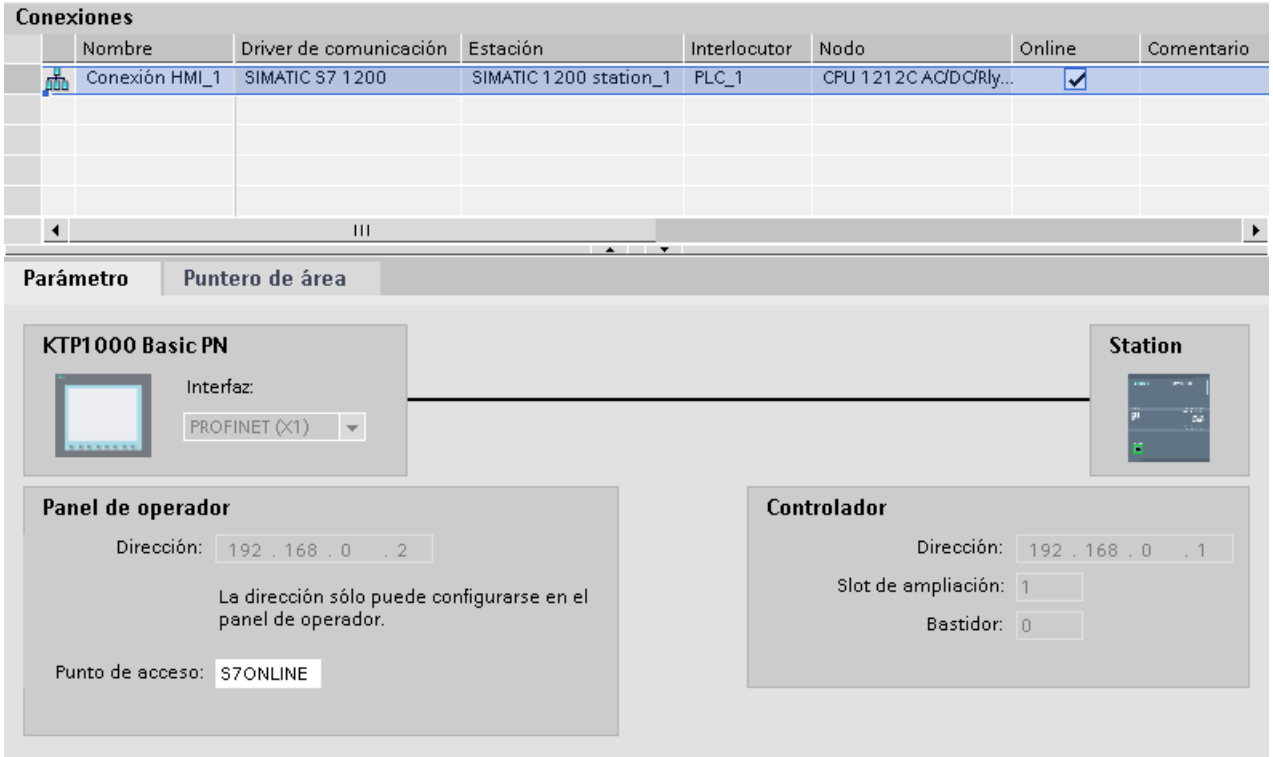
- Número de imagen
- Fecha/hora PLC
- Identificación del proyecto

Punteros de área para las conexiones

Introducción

En la ficha "Punteros de área" del editor "Conexiones" se configura la utilización de los punteros de área disponibles y sus respectivos ajustes.

Para configurar los punteros de área, abra el editor "Conexiones" y active la ficha "Punteros de área".



Estructura

La ficha "Punteros de área" comprende dos tablas con punteros de área. La tabla superior contiene los punteros de área que se crean y que se pueden activar por separado para cada conexión existente.

La tabla "Punteros de área generales del panel de operador" contiene los punteros de área que se crean sólo una vez en el proyecto y que sólo se pueden utilizar para una conexión.

Uso de punteros de área

Ficha "Punteros de área"

Antes de utilizar un puntero de área, actívelos en "Conexiones > Punteros de área". Seguidamente, parametrícelos.

En la ficha "Punteros de área" se parametriza lo siguiente:

Parámetro | Puntero de área

Activo	Nombre de visualiza...	Variable PLC	Tipo de acceso	Direcci...	Longitud	Modo de adquisi...	Ciclo de adquisici...	Comentario
☐	Coordinación	<No definido>	<acceso simbólico>		1	Cíclico continuo	<No definido>	
☐	Fecha/hora	<No definido>	<acceso simbólico>		6	Cíclico continuo	<No definido>	
☐	Petición del controla...	<No definido>	<acceso simbólico>		4	Cíclico continuo	<No definido>	
☐	Registro	<No definido>	<acceso simbólico>		5	Cíclico continuo	<No definido>	

Punteros de área generales del panel de operador

Conexión	Nombre de visualiza...	Variable PLC	Tipo de acceso	Dirección	Longitud	Modo de adquisi...	Ciclo de adquisic...	Comentario
<No definido>	Identificación del pro...	<No definido>	<acceso simbólico>		1	Cíclico continuo	<No definido>	
<No definido>	Número de imagen	<No definido>	<acceso simbólico>		5	Cíclico continuo	<No definido>	
<No definido>	Fecha/hora PLC	<No definido>	<acceso simbólico>		6	Cíclico continuo	<No definido>	

- Activo
  Activa el puntero de área.
- Nombre de visualización
  Nombre del puntero de área predeterminado por WinCC.
- Variable PLC
  Aquí se selecciona la variable PLC o la matriz de variables que se ha configurado como rango de datos para el puntero de área.
- Tipo de acceso
  Aquí se elige entre los siguientes tipos de acceso:
  - Acceso simbólico
  - Acceso absoluto
- Dirección
  Si ha elegido "Acceso simbólico", en este campo no aparecerá ninguna dirección.
  Si ha elegido "Acceso absoluto", entonces introduzca la dirección de una variable en el campo "Dirección".
- Longitud
  La longitud del puntero de área viene dada por WinCC.

- Ciclo de adquisición
  Determine aquí el ciclo de adquisición para punteros de área que serán leídos por el panel de operador. Tenga en cuenta que un tiempo de adquisición muy corto podría repercutir negativamente en el rendimiento del panel de operador.
- Comentario
  Introduzca un comentario p. ej. relativo a la utilización del puntero de área.

Acceder a los rangos de datos

Acceder a los rangos de datos

La tabla siguiente muestra cómo el panel de operador y el controlador acceden a los distintos rangos de datos, bien sea leyendo (R) o escribiendo (W) en ellos.

Rango de datos	Necesario para	Panel de operador	Controlador
Número de imagen	Evaluación desde el controlador cuya imagen está abierta en ese momento.	W	R
Registro	Transferencia de registros con sincronización	R/W	R/W
Fecha/hora	Transferencia de la fecha y hora del panel de operador al controlador	W	R
Fecha/hora PLC	Transferencia de la fecha y hora del controlador al panel de operador	R	W
Coordinación	Consultar el estado del panel de operador en el programa de control	W	R
Identificación del proyecto	Runtime comprueba si la identificación del proyecto de WinCC y el proyecto del controlador son coherentes.	R	W
Petición de control	Activación de funciones en el panel de operador a través del programa de control	R/W	R/W

Configurar un puntero de área

Configurar punteros de área

Introducción

A través de un puntero de área se accede a un área de datos del controlador. El área de datos se almacena en el controlador.

Antes de configurar punteros de área

Antes de utilizar un puntero de área es preciso activarlo y parametrizarlo en "Conexiones > Punteros de área".

Bloque de datos global

Para acceder al área de datos del controlador, hay que crear un bloque de datos global en el programa de control. El ejemplo siguiente muestra la utilización de un bloque de datos.

Longitud de los punteros de área

Para punteros de área con una longitud >= 1 se crea el área de datos en forma de matriz de variables en un bloque de datos global o un bloque de datos de instancia.

Para los punteros de área con una longitud = 1 también es posible utilizar una variable PLC.

La configuración de la variable en un bloque de datos depende de la longitud del puntero de área que se desea utilizar. La unidad de longitud de un puntero de área es una palabra de 16 bits.

Si, p. ej., se desea utilizar un puntero de área con una longitud "5", deberá crearse una matriz con 5 elementos de matriz del tipo de datos UINT en el bloque de datos.

Procedimiento alternativo

Como alternativa también se puede utilizar el modo de acceso absoluto para acceder a los punteros de área. El tipo de acceso absoluto sólo funciona en bloques de datos estándar del controlador.

Parametrizar un bloque de datos global

Introducción

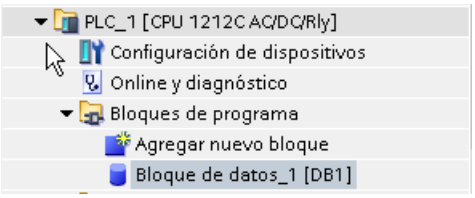
Para acceder al área de datos del controlador, hay que parametrizar un bloque de datos global para el puntero de área en el programa de control.

Requisitos

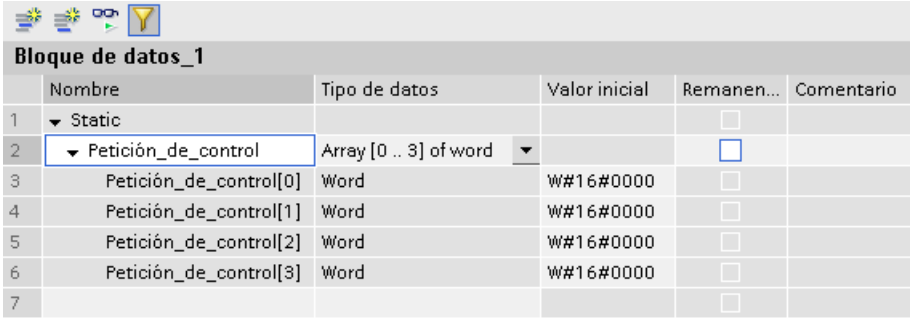
- En el proyecto se ha creado un controlador.
- Entre el controlador y el dispositivo HMI se ha configurado una conexión.
- El programa de control contiene un bloque de datos global.

Procedimiento

1. Abra "PLC > Bloques de programa" en el árbol del proyecto.
2. Haga doble clic en el bloque de datos global que ya se ha creado.
   Se abre el bloque de datos.



3. Introduzca un nombre de variable en la columna "Nombre".
4. Seleccione el tipo de datos "Array[lo .. hi] of type" en la columna "Tipo de datos".
5. Reemplace la entrada "lo" por el valor inferior de la dimensión de la matriz.
6. Reemplace la entrada "hi" por el valor superior de la dimensión de la matriz.
   Ejemplo: Si configura un puntero de área con una longitud "4", introduzca el valor "0" en "lo" y el valor "3" en "hi" entre corchetes.
7. Reemplace la palabra "type" por el tipo de datos "word".
   El tipo de datos completo para una matriz de 4 variables es el siguiente: "Array[0 .. 3] of word".
   Una vez confirmada la entrada se creará la matriz de variables.
8. Haga clic en "Compilar".
   El proyecto se compila.



Bloque de datos_1

	Nombre	Tipo de datos	Valor inicial	Remanen...	Comentario
1	Static				
2	Petición_de_control	Array [0 .. 3] of word			
3	Petición_de_control[0]	Word	W#16#0000		
4	Petición_de_control[1]	Word	W#16#0000		
5	Petición_de_control[2]	Word	W#16#0000		
6	Petición_de_control[3]	Word	W#16#0000		
7					

Configurar punteros de área para una conexión

Introducción

Una vez se ha parametrizado el bloque de datos global, hay que crear el puntero de área para la conexión.

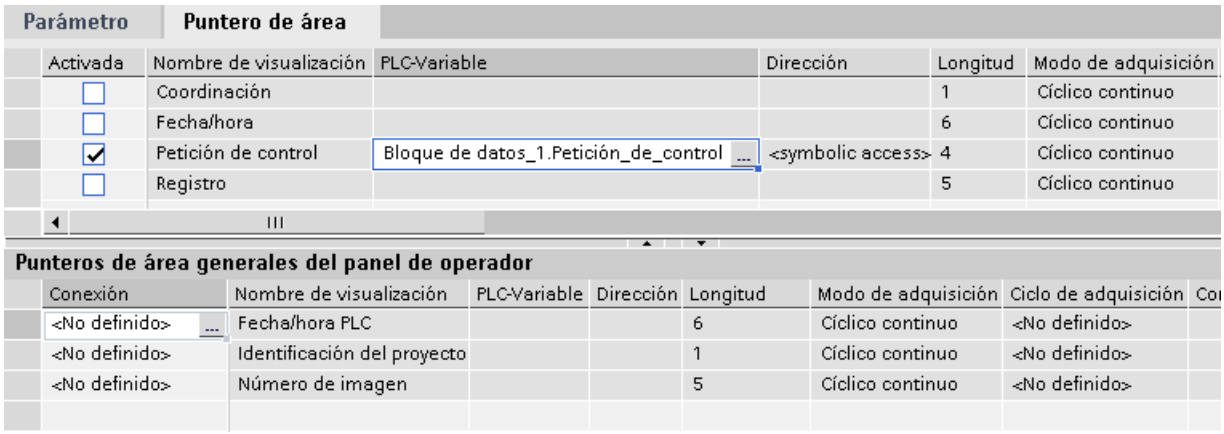
Requisitos

- El bloque de datos global se ha parametrizado en el programa de control.

Procedimiento

1. Abra "HMI > Conexiones" en el árbol del proyecto.
2. Haga clic en la ficha "Punteros de área".
3. Active el puntero de área deseado.
   Un puntero de área global se activa seleccionando la conexión en el campo "Conexión".
4. Haga clic en el botón de navegación del campo "Variable de controlador".
   Se abrirá la lista de objetos.

5. En la lista de objetos, navegue hasta el bloque de datos y seleccione la variable en la ventana derecha.
   Para configurar un puntero de área con una longitud "1" no se requieren variables de matriz.



6. Al crear la variable, seleccione el tipo de datos "Word" en el bloque de datos.

En caso necesario, durante la configuración se pueden modificar otros parámetros, p. ej. el ciclo de adquisición.

Resultado

El puntero de área está activado y conectado con la variable de controlador en el bloque de datos global.

10.8.5 Dependencia del panel

10.8.5.1 Basic Panel

Drivers de comunicación

Drivers de comunicación SIMATIC

Dependencia de los Basic Panels

Para los Basic Panels se configuran distintos drivers de comunicación en función de la comunicación.

Las siguientes tablas muestran qué drivers de comunicación están habilitados para la comunicación integrada y no integrada.

Dentro de la comunicación integrada existen diferentes versiones de paneles de operador.

Drivers de comunicación para la comunicación integrada (V11.0)

Paneles de operador	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)	SIMATIC S7-1500	SIMATIC S7-300/400
KP300 Basic	sí	sí	sí	sí	sí	no	sí
KP400 Basic	sí	sí	sí	sí	sí	no	sí
KTP400 Basic PN	sí	sí	sí	sí	sí	no	sí
KTP600 Basic DP	sí	sí	sí	sí	sí	no	sí
KTP600 Basic PN	sí	sí	sí	sí	sí	no	sí
KTP1000 Basic DP	sí	sí	sí	sí	sí	no	sí
KTP1000 Basic PN	sí	sí	sí	sí	sí	no	sí
TP1500 Basic PN	sí	sí	sí	sí	sí	no	sí

Drivers de comunicación para la comunicación integrada (V12.0)

Paneles de operador	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)	SIMATIC S7-1500	SIMATIC S7-300/400
KP300 Basic	no	sí	sí	sí	sí	sí	sí
KP400 Basic	no	sí	sí	sí	sí	sí	sí
KTP400 Basic PN	no	sí	sí	sí	sí	sí	sí
KTP600 Basic DP	no	sí	sí	sí	sí	sí	sí
KTP600 Basic PN	no	sí	sí	sí	sí	sí	sí
KTP1000 Basic DP	no	sí	sí	sí	sí	sí	sí
KTP1000 Basic PN	no	sí	sí	sí	sí	sí	sí
TP1500 Basic PN	no	sí	sí	sí	sí	sí	sí

Drivers de comunicación para la comunicación integrada (V13.0)

Paneles de operador	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)	SIMATIC S7-1500	SIMATIC S7-300/400
KTP400 Basic	no	sí	sí	sí	sí	sí	sí
KTP700 Basic	no	sí	sí	sí	sí	sí	sí
KTP900 Basic	no	sí	sí	sí	sí	sí	sí
KTP1200 Basic	no	sí	sí	sí	sí	sí	sí

Drivers de comunicación para la comunicación no integrada

Paneles de operador	SIMATIC S7-1200	SIMATIC S7-1500	SIMATIC S7-300/400	SIMATIC S7-200	SIMATIC LOGO!	SIMATIC HTTP Protocol
KP300 Basic	sí	sí	sí	sí	sí	no
KP400 Basic	sí	sí	sí	sí	sí	no
KTP400 Basic PN	sí	sí	sí	sí	sí	no
KTP600 Basic DP	sí	sí	sí	sí	sí	no
KTP600 Basic PN	sí	sí	sí	sí	sí	no
KTP700 Basic PN	sí	sí	sí	sí	sí	no
KTP900 Basic PN	sí	sí	sí	sí	sí	no
KTP1000 Basic DP	sí	sí	sí	sí	sí	no
KTP1000 Basic PN	sí	sí	sí	sí	sí	no
TP1500 Basic PN	sí	sí	sí	sí	sí	no

Otros drivers de comunicación

Dependencia de los Basic Panels

La siguiente tabla muestra qué drivers de comunicación pueden configurarse con los diferentes Basic Panels.

Drivers de comunicación

Paneles de operador	OPC	Allen-Bradley EtherNet/IP	Allen-Bradley DF1	Mitsubishi MC TCP/IP	Mitsubishi FX	Modicon Modbus TCP/IP	Modicon Modbus RTU	Omron Host Link
KP300 Basic	no	sí	no	sí	no	sí	no	no
KP400 Basic	no	sí	no	sí	no	sí	no	no
KTP400 Basic PN	no	sí	no	sí	no	sí	no	no
KTP600 Basic DP	no	no	sí [2]	no	sí	no	sí [1]	sí
KTP600 Basic PN	no	sí	no	sí	no	sí	no	no
KTP700 Basic PN	no	sí	no	sí	no	sí	no	no
KTP900 Basic PN	no	sí	no	sí	no	sí	no	no
KTP1000 Basic DP	no	no	sí [2]	no	sí	no	sí [1]	sí
KTP1000 Basic PN	no	sí	no	sí	no	sí	no	no
TP1500 Basic PN	no	sí	no	sí	no	sí	no	no

[1] Sólo con convertidor RS 422-RS232
Referencia: 6AV6 671-8XE00-0AX0

[2] Comunicación directa con PLC 5 o módulo KF2; por lo demás, sólo autorizado con convertidor RS422-RS232 (opcional).
Referencia: 6AV6 671-8XE00-0AX0

Interfaces de los Basic Panels

Dependencia de los Basic Panels

La siguiente tabla muestra qué puertos están disponibles en el panel de operador para los protocolos de los drivers de comunicación.

Tabla 10-11 Basic Panels

	KP300 Basic KP400 Basic KTP400 Basic PN KTP600 Basic PN KTP700 Basic PN KTP900 Basic PN KTP1000 Basic PN TP1500 Basic PN	KTP600 Basic DP KTP1000 Basic DP
SIMATIC S7 - PPI [1]	—	MPI/DP (X2)
SIMATIC S7 - MPI	—	MPI/DP (X2)
SIMATIC S7 - PROFIBUS	—	MPI/DP (X2)
SIMATIC S7 - PROFINET	PROFINET (X1)	—
Protocolo SIMATIC HMI HTTP	—	—
OPC	—	—
Allen-Bradley EtherNet/IP	PROFINET (X1)	—
Allen-Bradley DF1	—	MPI/DP (X2) [2]
Mitsubishi TCP/IP	PROFINET (X1)	—
Mitsubishi FX	—	MPI/DP (X2) (RS422)
Modicon Modbus TCP	PROFINET (X1)	—
Modicon Modbus RTU	—	MPI/DP (X2) [3]
Omron Host Link	—	MPI/DP (X2) (RS422)

[1] Sólo con SIMATIC S7-200

[2] Comunicación directa con PLC5 o módulo KF2; por lo demás, sólo autorizado con convertidor RS422-RS232 (opcional).
Referencia: 6AV6 671-8XE00-0AX0

[3] Sólo autorizado con convertidor RS 422-RS232
Referencia: 6AV6 671-8XE00-0AX0

Punteros de área para Basic Panels

Introducción

Los punteros de área son campos de parámetros de los que el panel de operador obtiene información sobre la posición y el tamaño de los rangos de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación. Evaluando los datos almacenados en los rangos de datos, el controlador y el panel de operador ejecutan acciones bien definidas.

WinCC utiliza los siguientes punteros de área:

- petición de control
- identificación del proyecto
- número de imagen
- registro
- fecha/hora
- fecha/hora PLC
- coordinación

Disponibilidad de los punteros de área

La tabla siguiente muestra los punteros de área disponibles en los paneles de operador. Tenga en cuenta que los punteros de área sólo se pueden utilizar si están disponibles los drivers de comunicación.

Punteros de área

	KP300 Basic	KTP400 Basic PN	KTP600 Basic PN	KTP600 Basic DP	KTP700 Basic PN	KTP900 Basic PN	KTP1000 Basic PN	KTP1000 Basic DP	TP1500 Basic PN
Número de imagen	sí	sí	sí	sí	sí	sí	sí	sí	sí
Registro	sí	sí	sí	sí	sí	sí	sí	sí	sí
Fecha/hora	sí	sí	sí	sí	sí	sí	sí	sí	sí
Fecha/hora PLC	sí	sí	sí	sí	sí	sí	sí	sí	sí
Coordinación	sí	sí	sí	sí	sí	sí	sí	sí	sí
Identificación del proyecto	sí	sí	sí	sí	sí	sí	sí	sí	sí
Petición de control	sí	sí	sí	sí	sí	sí	sí	sí	sí

10.8.6 Comunicación con SIMATIC S7 1500

10.8.6.1 Comunicación con SIMATIC S7 1500

Introducción

Este apartado describe la comunicación entre un panel de operador y el controlador SIMATIC S7 1500.

Para el controlador SIMATIC S7 1500 es posible configurar los siguientes canales de comunicación:

- PROFINET
- PROFIBUS

Conexión HMI para la comunicación

Las conexiones entre un panel de operador y un SIMATIC S7 1500 se configuran en el editor "Dispositivos y redes".

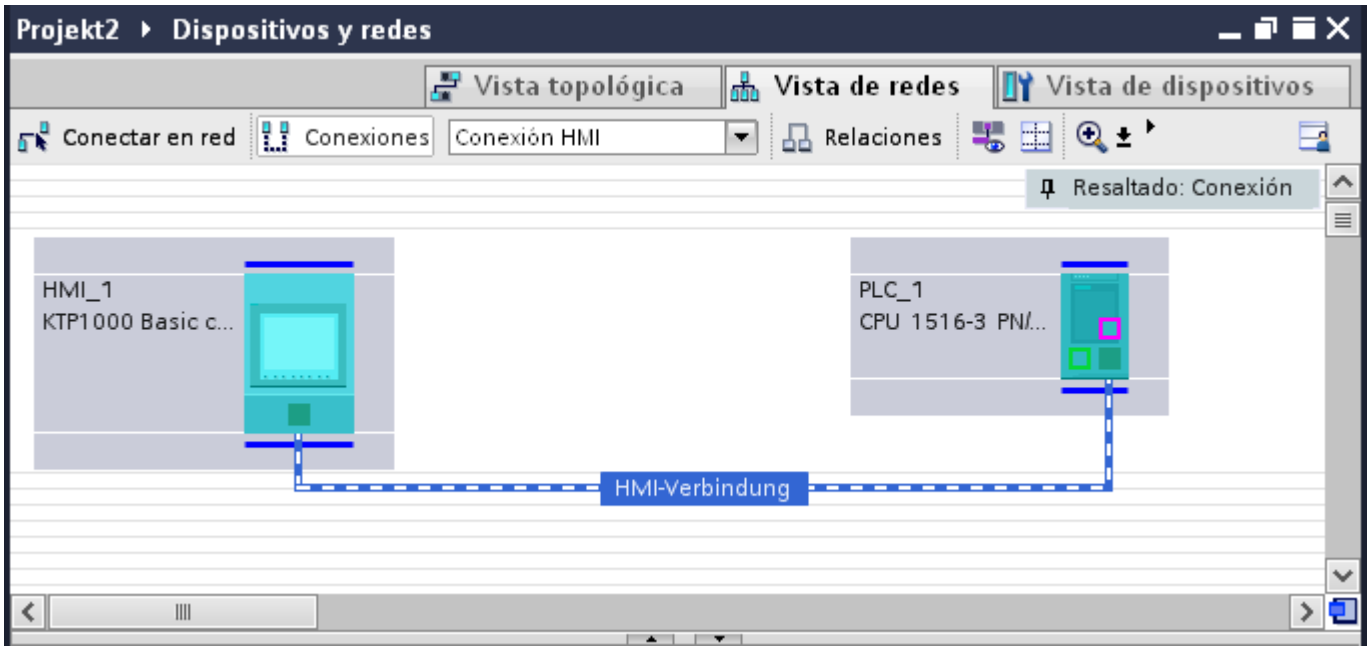
10.8.6.2 Comunicación vía PROFINET

Configurar una conexión HMI

Comunicación vía PROFINET

Conexiones HMI mediante PROFINET

Si se han insertado en el proyecto un panel de operador y un SIMATIC S7 1500, interconecte las dos interfaces PROFINET en el editor "Dispositivos y redes".



También es posible conectar varios paneles de operador a un SIMATIC S7 1500 y varios SIMATIC S7 1500 a un panel de operador.

El número máximo de interlocutores que pueden conectarse a un panel de operador depende del panel utilizado.

Encontrará más información al respecto en la documentación del panel de operador correspondiente.

Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFINET entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".

Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión vía PROFINET entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Configurar una conexión HMI mediante PROFINET

Introducción

Una conexión HMI vía PROFINET o Ethernet entre paneles de operador y un SIMATIC S7 1500 se configura en el editor "Dispositivos y redes".

⚠ PRECAUCIÓN
Comunicación vía Ethernet En una comunicación basada en Ethernet, el usuario final es responsable de la seguridad de su red de datos. Los ataques selectivos pueden provocar la sobrecarga del dispositivo y perjudicar su funcionalidad.

Requisitos

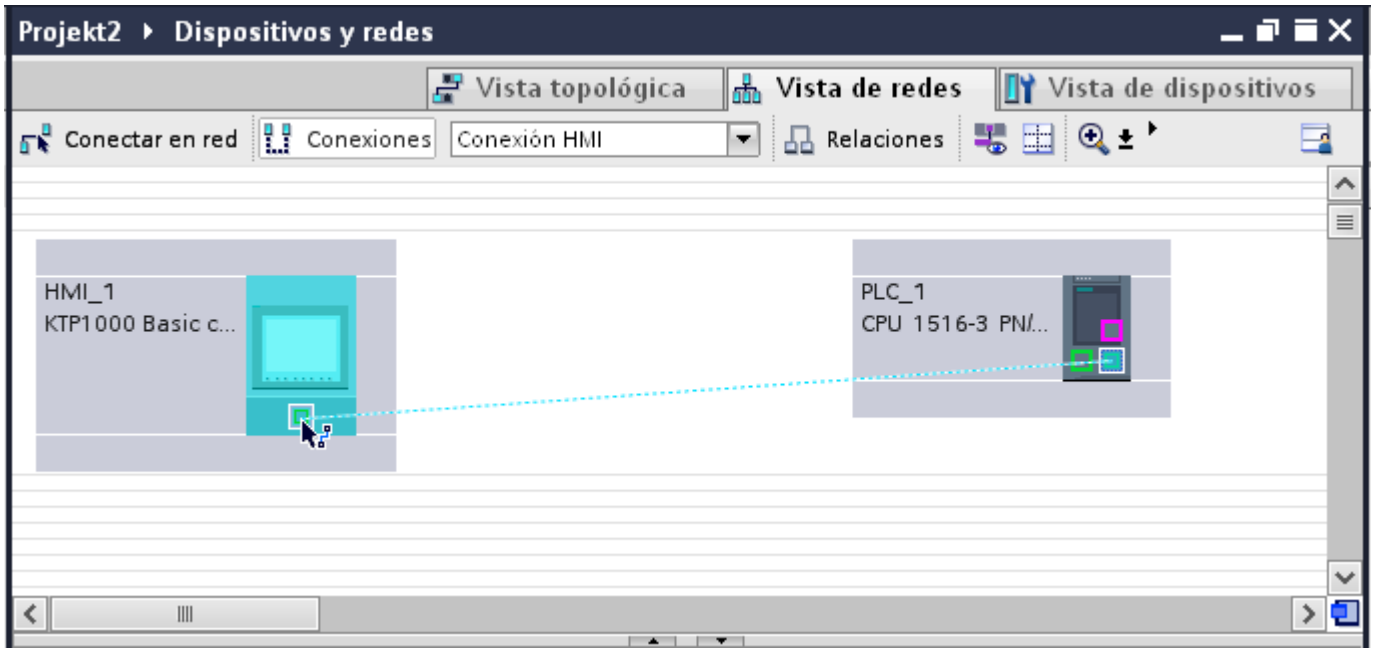
Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

- panel de operador con interfaz PROFINET o Ethernet
- SIMATIC S7 1500 con interfaz PROFINET.

Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.
2. Haga clic en el botón "Conexiones" y elija el tipo "Conexión HMI".
   Los dispositivos disponibles para conectar se representan marcados en color.

3. Haga clic en la interfaz PROFINET del controlador y arrastre una conexión a la interfaz PROFINET o Ethernet del panel de operador utilizando la función Drag&Drop.



4. Haga clic en la línea de conexión.
5. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.
   En la ventana de inspección se representa la conexión gráficamente.
6. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros PROFINET en la ventana de inspección de acuerdo con los requisitos del proyecto.
   Encontrará más información al respecto en el capítulo "Auto-Hotspot".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión.

El nombre local de la conexión sólo se puede modificar en la tabla.

Resultado

Se ha creado una conexión entre un panel de operador y un SIMATIC S7 1500. Los parámetros de conexión "Dirección IP" y "Máscara de subred" están configurados.

Parámetros PROFINET

Parámetros PROFINET de la conexión HMI

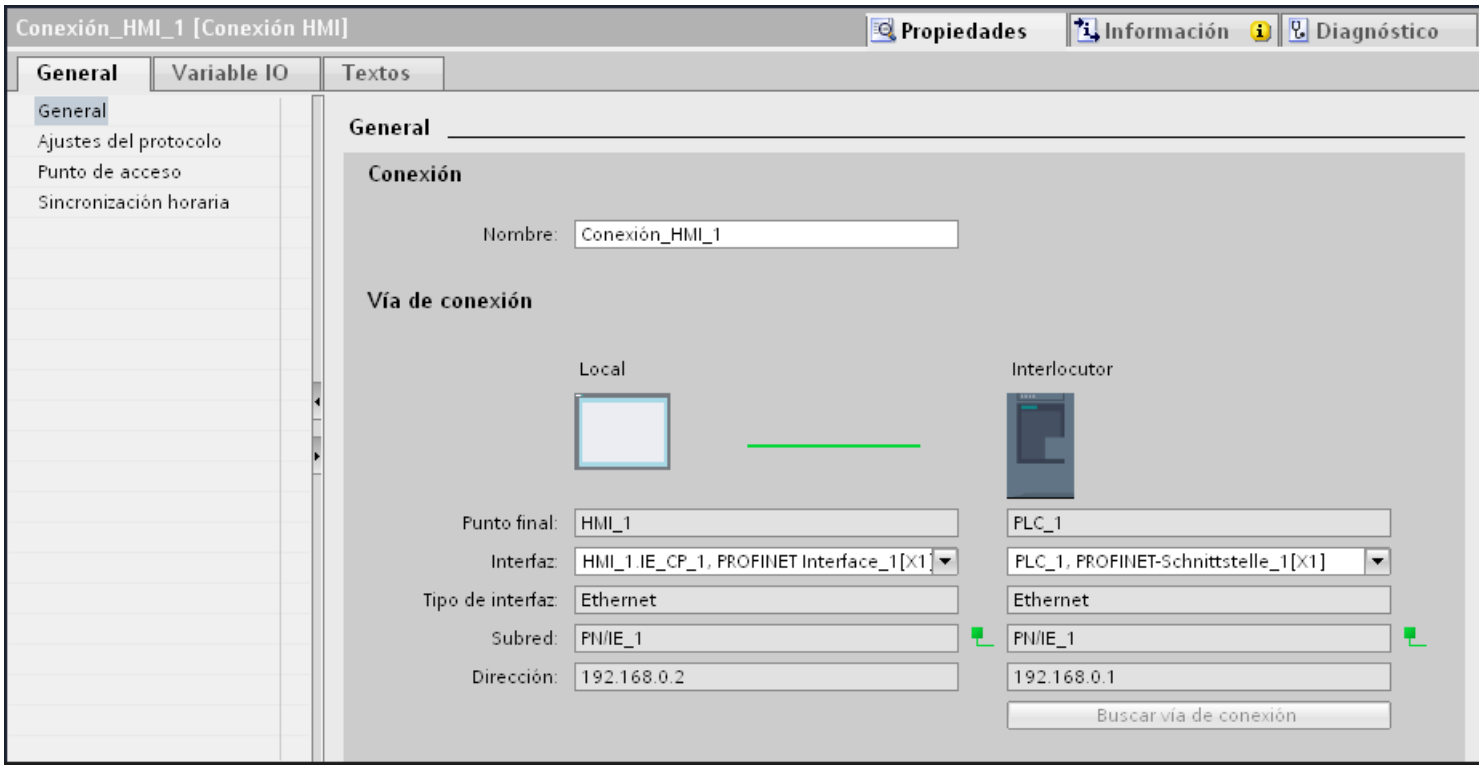
Parámetros PROFINET para la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



"Conexión"

Muestra el nombre de la conexión HMI.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros PROFINET. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.
- "Tipo de interfaz"
  Muestra el tipo de interfaz seleccionado. Esta área no es editable.
- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección IP seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

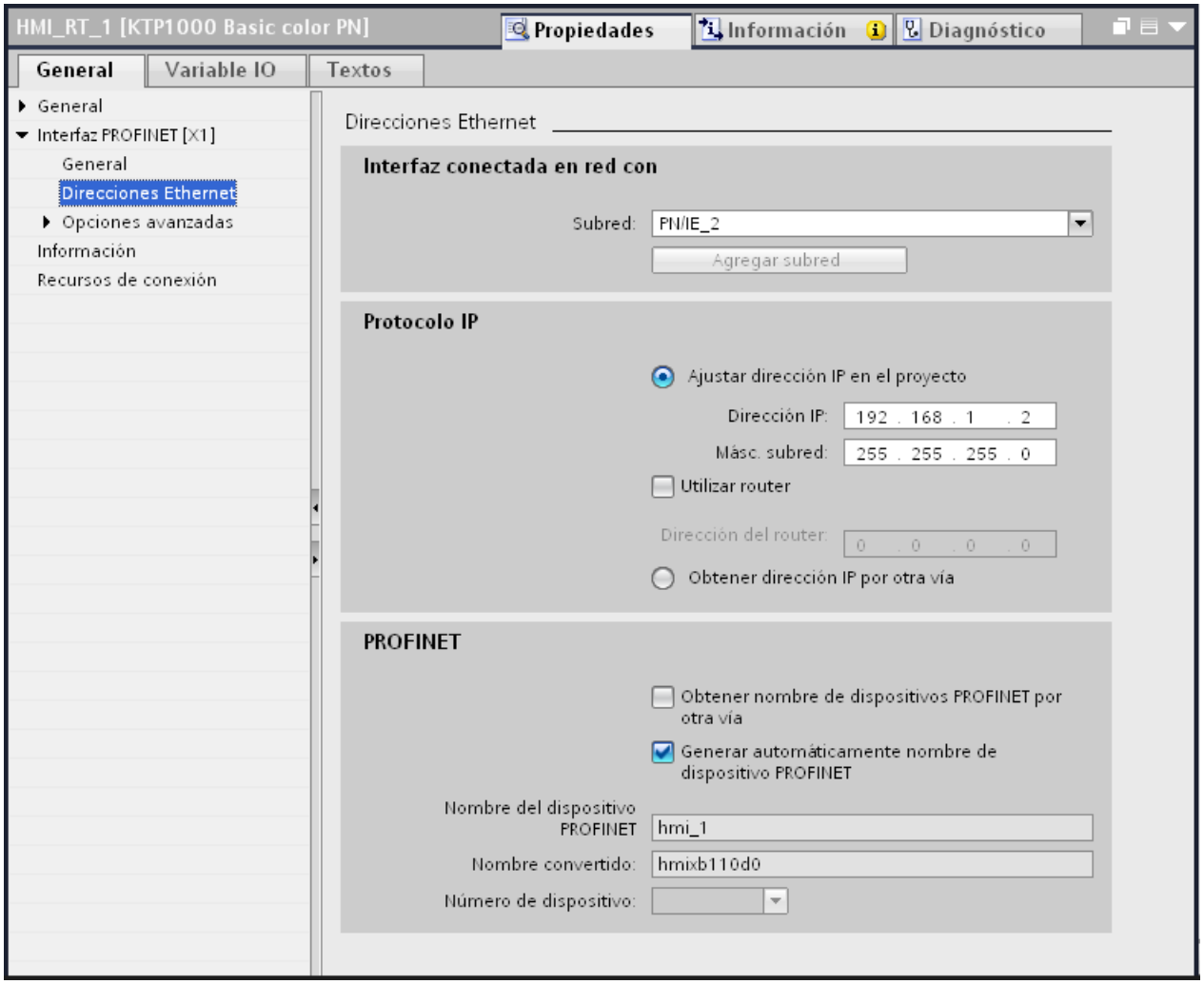
Parámetros PROFINET del panel de operador

Parámetros PROFINET del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros PROFINET del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Protocolo IP"

- "Ajustar la dirección IP en el proyecto"
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.

  Nota

  En los paneles de operador con el sistema operativo Windows CE 3.0 se realiza un reinicio automático.

  Paneles de operador con Windows CE 3.0:
  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Máscara de subred"
  En el área "Máscara de subred" se asignan los datos de la máscara de subred.
- "Utilizar router IP"
  Si utiliza un router IP, active "Utilizar router IP" e introduzca la dirección del router en el campo "Dirección del router".
- "Obtener la dirección IP de otro modo"
  Si la función "Obtener la dirección IP de otro modo" está activada, la dirección IP no se toma del proyecto. Introduzca la dirección IP directamente en el Control Panel del panel de operador.

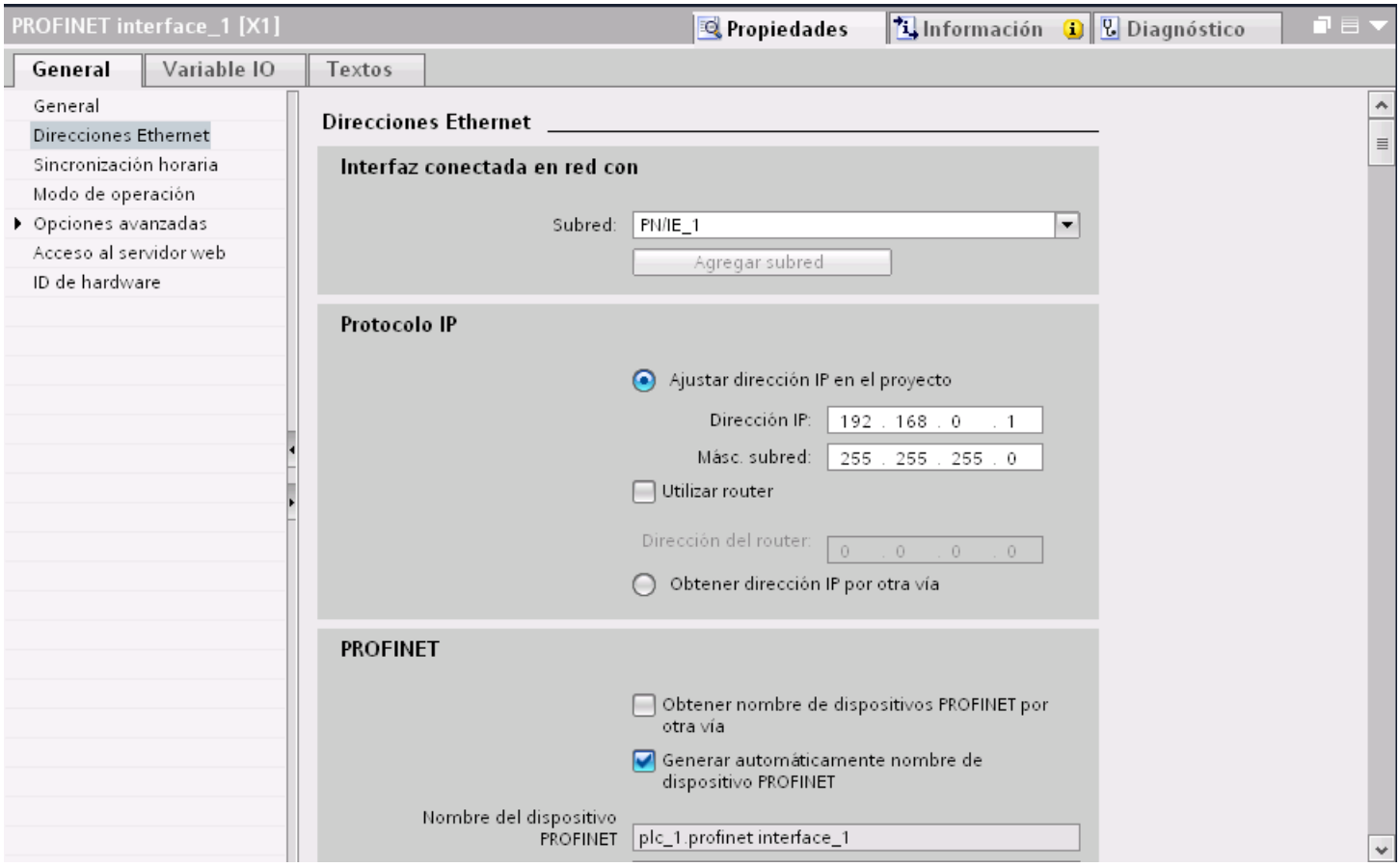
Parámetros PROFINET del controlador

Parámetros PROFINET del controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFINET del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el controlador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Protocolo IP"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección IP"
  En el área "Dirección IP" se asigna la dirección IP del panel de operador.
- "Máscara de subred"
  En el área "Máscara de subred" se asignan los datos de la máscara de subred.
  Si se utiliza un router IP, active "Utilizar router IP" e introduzca la dirección del router en el campo inferior.

Protección de la comunicación

Niveles de protección

Si desea proteger la comunicación del controlador y el panel de operador, puede asignar niveles de protección para la comunicación.

En una CPU SIMATIC S7-1500 es posible introducir varias contraseñas y, por tanto, establecer diferentes derechos de acceso para distintos grupos de usuarios.

Las contraseñas se introducen en una tabla, de modo que cada una tiene asignado exactamente un nivel de protección.

En la columna "Protección" se indica el efecto de la contraseña.

Para ajustar niveles de protección en el controlador SIMATIC S7-1200 hay que tener en cuenta diversos aspectos. Encontrará más información al respecto en: Niveles de protección S7-1200

Ejemplo

Para una CPU estándar (es decir, no una CPU F) se elige el nivel "Protección completa" durante la configuración del controlador.

A continuación se introduce una contraseña propia para cada nivel de protección situado encima en la tabla.

Para los usuarios que no conocen ninguna de las contraseñas, la CPU está completamente protegida. Ni siquiera son posibles accesos HMI.

Para los usuarios que conocen una de las contraseñas parametrizadas, la repercusión depende de la fila de la tabla en la que se encuentre la contraseña:

- La contraseña de la fila 1 (sin protección) hace como si la CPU no estuviera protegida. Los usuarios que conocen esta contraseña tienen un acceso ilimitado a la CPU.
- La contraseña de la fila 2 (protección de escritura) hace como si la CPU estuviera protegida contra escritura. A pesar de conocer esta contraseña, los usuarios solo tienen acceso de lectura a la CPU.
- La contraseña de la fila 3 (protección de escritura y lectura) hace como si la CPU estuviera protegida contra escritura y lectura, de modo que los usuarios que conocen esta contraseña solo disponen de acceso HMI.

Contraseña de acceso para la conexión HMI

Introducción

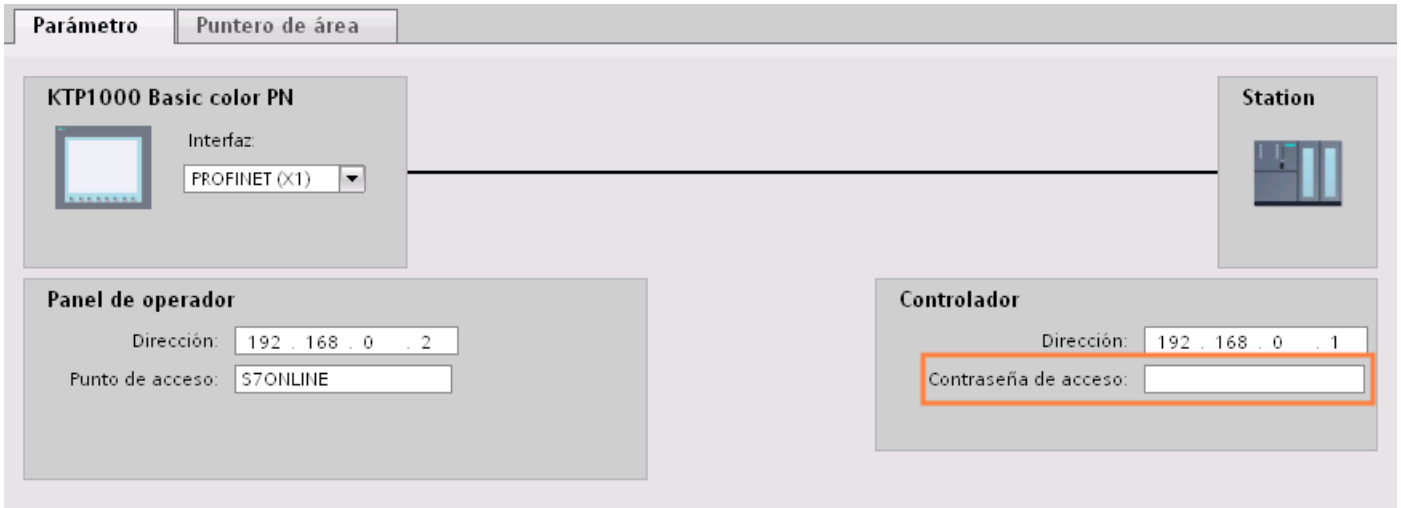
El acceso a un controlador se protege asignándole una contraseña.

Esta se asigna al configurar la contraseña.

A partir del nivel de protección "Protección completa" es imprescindible introducir la contraseña. Si no se introduce la contraseña o se introduce erróneamente, no se establece ninguna comunicación con el controlador.

Introducir contraseña de acceso

La contraseña de acceso para el controlador se asigna en el editor "Conexiones".



Definir opciones de puerto

Definición de opciones de puerto

Modificar los ajustes de conexión para el puerto PROFINET IO

En caso necesario, los ajustes de red para el puerto PROFINET IO se pueden modificar. La opción predeterminada es la definición automática de ajustes que, por norma general, permite comunicarse correctamente.

Posibilidades de ajuste de velocidad de transferencia/dúplex

Se pueden realizar los siguientes ajustes para "Velocidad de transferencia/dúplex" en función del dispositivo seleccionado:

- Ajuste automático
  Ajuste predeterminado recomendando para el puerto. La configuración de la transferencia con el puerto del interlocutor se "negocia" automáticamente. Con este ajuste también se activa automáticamente la opción "Activar autonegotiation", es decir, se puede utilizar cable cruzado o no cruzado para la conexión.
- TP/ITP con x Mbits/s. dúplex (semidúplex)
  Ajuste de la velocidad de transferencia y del modo dúplex/semidúplex. El efecto depende de la opción ajustada "Activar autonegotiation":
  - Autonegotiation activada
    Es posible utilizar tanto cable cruzado como no cruzado.
  - Autonegotiation desactivada
    Asegúrese de utilizar el cable correcto (cable cruzado o no cruzado). El puerto también se vigila con este ajuste.
- Desactivado
  En función del tipo de módulo, en la lista desplegable aparece la opción "desactivado". Así, por ejemplo, tiene la posibilidad de prohibir el acceso a un puerto no utilizado por motivos de seguridad. Con este ajuste no se generan eventos de diagnóstico.

Opción "Vigilar"

Con esta opción se activa y desactiva el diagnóstico de puerto. Ejemplos de diagnóstico de puerto: se vigila el estado del enlace, es decir, se genera el diagnóstico en caso de fallo del enlace y se vigila la reserva del sistema en los puertos de fibra óptica.

Opción "Activar autonegotiation"

El ajuste de autonegotiation sólo es modificable si se ha seleccionado un medio específico (p. ej. TP 100 Mbits/s dúplex). De las propiedades del módulo depende que un medio específico se pueda ajustar o no.

Si se ha desactivado la autonegotiation, se fuerza el ajuste fijo del puerto, de manera similar a como se requiere p. ej. en caso de arranque priorizado del dispositivo IO.

Hay que procurar realizar los mismos ajustes en el puerto del interlocutor, porque en esta opción los parámetros operativos de la red conectada no se reconocen y, en consecuencia, la velocidad de transferencia de datos y el modo de transferencia no se pueden ajustar óptimamente.

Nota

Cuando hay un puerto local interconectado, STEP 7 realiza el ajuste del puerto del interlocutor siempre y cuando soporte este ajuste. Si el puerto del interlocutor no soporta este ajuste, se genera un mensaje de error.

Normas de cableado con autonegotiation desactivada

Requisitos

Se han realizado los ajustes siguientes p. ej. para acelerar el tiempo de arranque del dispositivo IO para el puerto correspondiente:

- Velocidad de transferencia fija
- Autonegotiation incl. autocrossing desactivada

De ese modo, durante el arranque se prescinde del tiempo de negociación de la velocidad de transferencia.

Si ha desactivado Autonegotiation, debe observar las normas de cableado.

Normas de cableado con Autonegotiation desactivada

Los dispositivos PROFINET tienen los dos tipos de puertos siguientes:

Tipo de puerto	Dispositivos PROFINET	Observación
Switchport con asignación de pines cruzada	En dispositivos IO: puerto 2 En CPUs S7 con 2 puertos: puerto 1 y puerto 2	Asignación de pines cruzada significa que la asignación de pines de los puertos para envío y recepción se intercambian internamente entre los dispositivos PROFINET afectados.
Puerto de terminal con asignación de pines no cruzada	En dispositivos IO: puerto 1 En CPUs S7 con un puerto: puerto 1	-

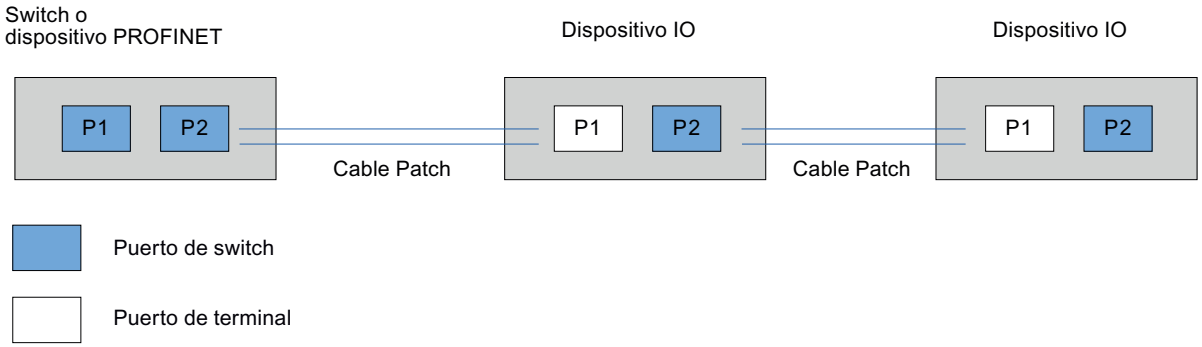
Validez de las normas de cableado

Las normas de cableado descritas en el siguiente apartado rigen exclusivamente para aquellos casos en los que se ha definido un ajuste de puerto fijo.

Normas para el cableado

Varios dispositivos IO pueden conectarse en línea con un tipo de cable (cable no cruzado). Para ello se conecta el puerto 2 del dispositivo IO (periferia descentralizada) con el puerto 1

del siguiente dispositivo IO. En el gráfico siguiente se representa un ejemplo con dos dispositivos IO.



Limitaciones en el puerto

Requisitos

Para poder trabajar con "boundaries" (limitaciones), el dispositivo correspondiente debe contar con más de un puerto. Si PROFINET no soporta limitaciones, éstas tampoco se visualizan.

Activar limitaciones

Se entiende por limitaciones los límites para transferir determinados frames en Ethernet. Es posible activar las siguientes limitaciones en un puerto:

- "Fin del registro de nodos accesibles"
  No se transfieren los frames DCP para el registro de nodos accesibles. Los nodos que están detrás de este puerto ya no se visualizan en "Nodos accesibles" del árbol del proyecto. La CPU ya no puede acceder a dichos nodos.
- "Fin de la detección de topología"
  No se transfieren los frames LLDP (Link Layer Discovery Protocol) para la detección de topología.
- "Fin del dominio Sync"
  No se transfieren frames Sync que se transmiten dentro de un dominio Sync para sincronizar los nodos.
  Si se opera p. ej. un dispositivo PROFINET con más de dos puertos en un anillo, es preciso impedir la inserción de frames Sync en el anillo activando una limitación Sync (en los puertos no incluidos en el anillo).
  Otro ejemplo: si se quieren utilizar varios dominios Sync, hay que configurar una limitación de dominio Sync para el puerto que está conectado con un dispositivo PROFINET del otro dominio Sync.

Restricciones

Hay que tener en cuenta las siguientes restricciones:

- Las casillas de verificación sólo están operativas si el puerto soporta la correspondiente característica.
- Si se ha determinado un puerto interlocutor para el puerto, las siguientes casillas de verificación no están operativas:
  - "Fin del registro de nodos accesibles"
  - "Fin de la detección de topología"
- Si la autonegotiation está desactivada, no está operativa ninguna casilla de verificación.

10.8.6.3 Comunicación vía PROFIBUS

Configurar una conexión HMI

Configurar una conexión HMI mediante PROFIBUS

Introducción

Una conexión HMI vía PROFIBUS entre paneles de operador y un SIMATIC S7 1500 se configura en el editor "Dispositivos y redes".

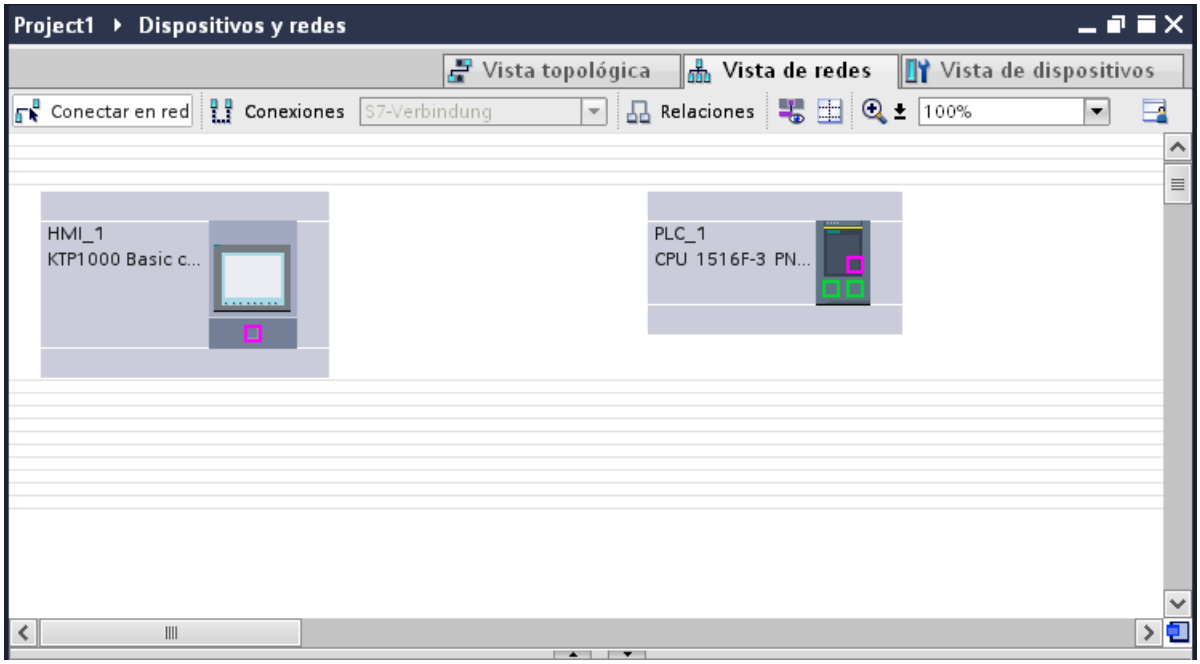
Requisitos

Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

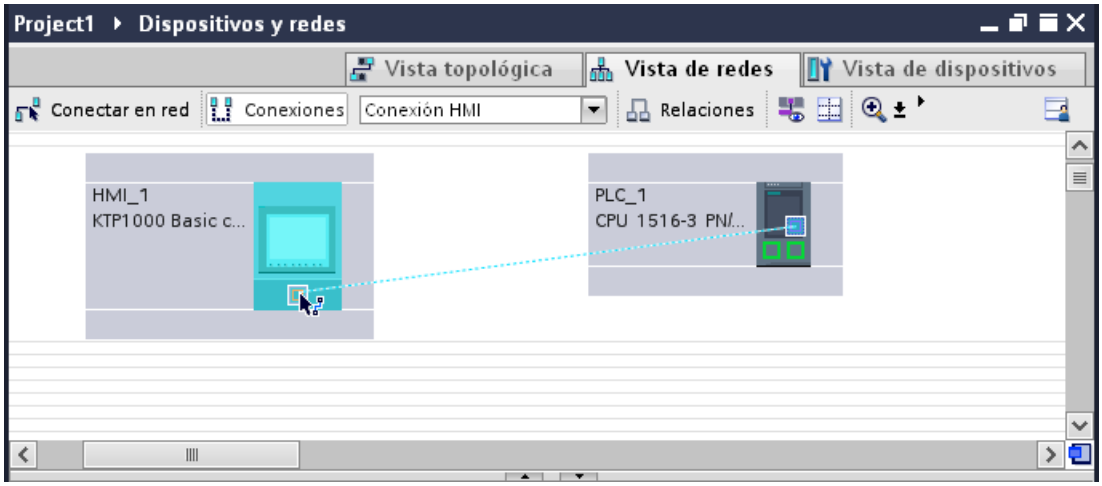
- panel de operador con interfaz MPI/DP
- SIMATIC S7 1500 con interfaz PROFIBUS

Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.



2. Haga clic en el botón "Conexiones".
   Los dispositivos disponibles para conectar se representan marcados en color.
3. Haga clic en la interfaz del panel de operador.
4. Seleccione el tipo de interfaz "PROFIBUS" en la ventana de inspección "Propiedades > General > HMI MPIDP > Parámetros".
5. Haga clic en la interfaz del controlador y arrastre una conexión hasta el panel de operador con la función Drag&Drop.



6. Haga clic en la línea de conexión.

7. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.
   En la ventana de inspección se representa la conexión gráficamente.
8. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros PROFINET en la ventana de inspección de acuerdo con los requisitos del proyecto.
   Encontrará más información al respecto en el capítulo "Auto-Hotspot".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión.

El nombre local de la conexión sólo se puede modificar en la tabla.

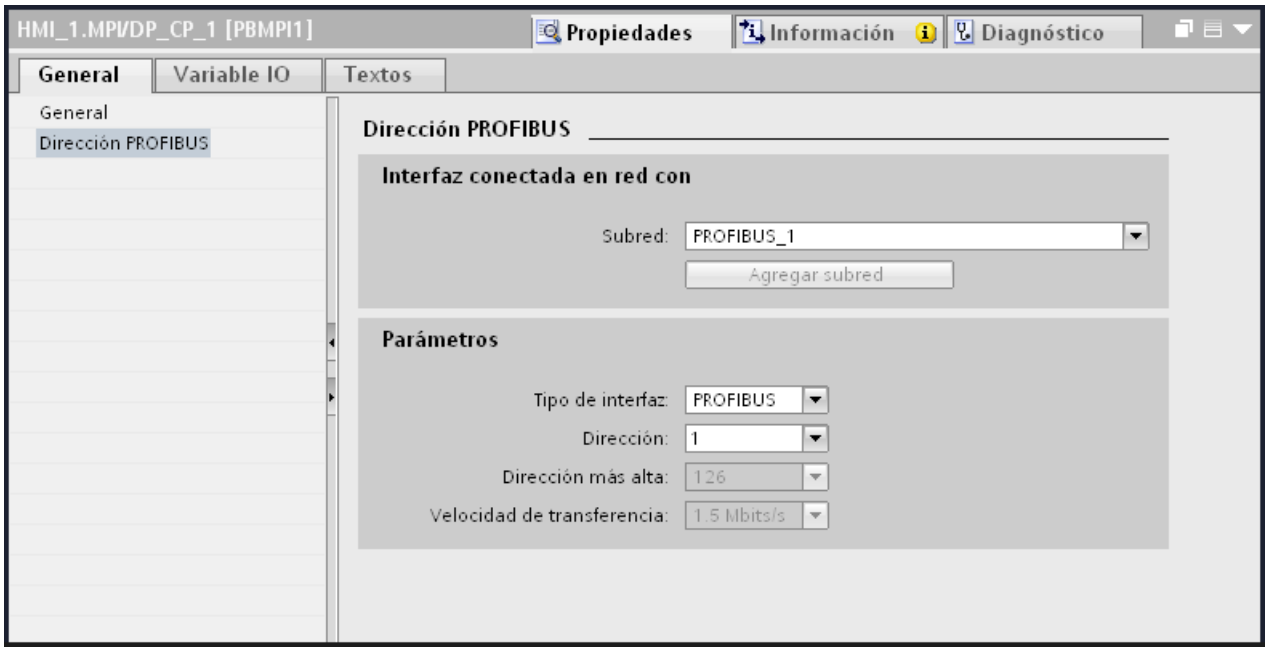
Resultado

Se ha creado una conexión HMI entre un panel de operador y un SIMATIC S7 1500 a través de PROFIBUS.

Comunicación vía PROFIBUS

Conexiones HMI mediante PROFIBUS

Si se han insertado en el proyecto un panel de operador y un SIMATIC S7 1500, interconecte las dos interfaces PROFIBUS en el editor "Dispositivos y redes".



Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFIBUS entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".

Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Parámetros PROFIBUS

Parámetros PROFIBUS de la conexión HMI

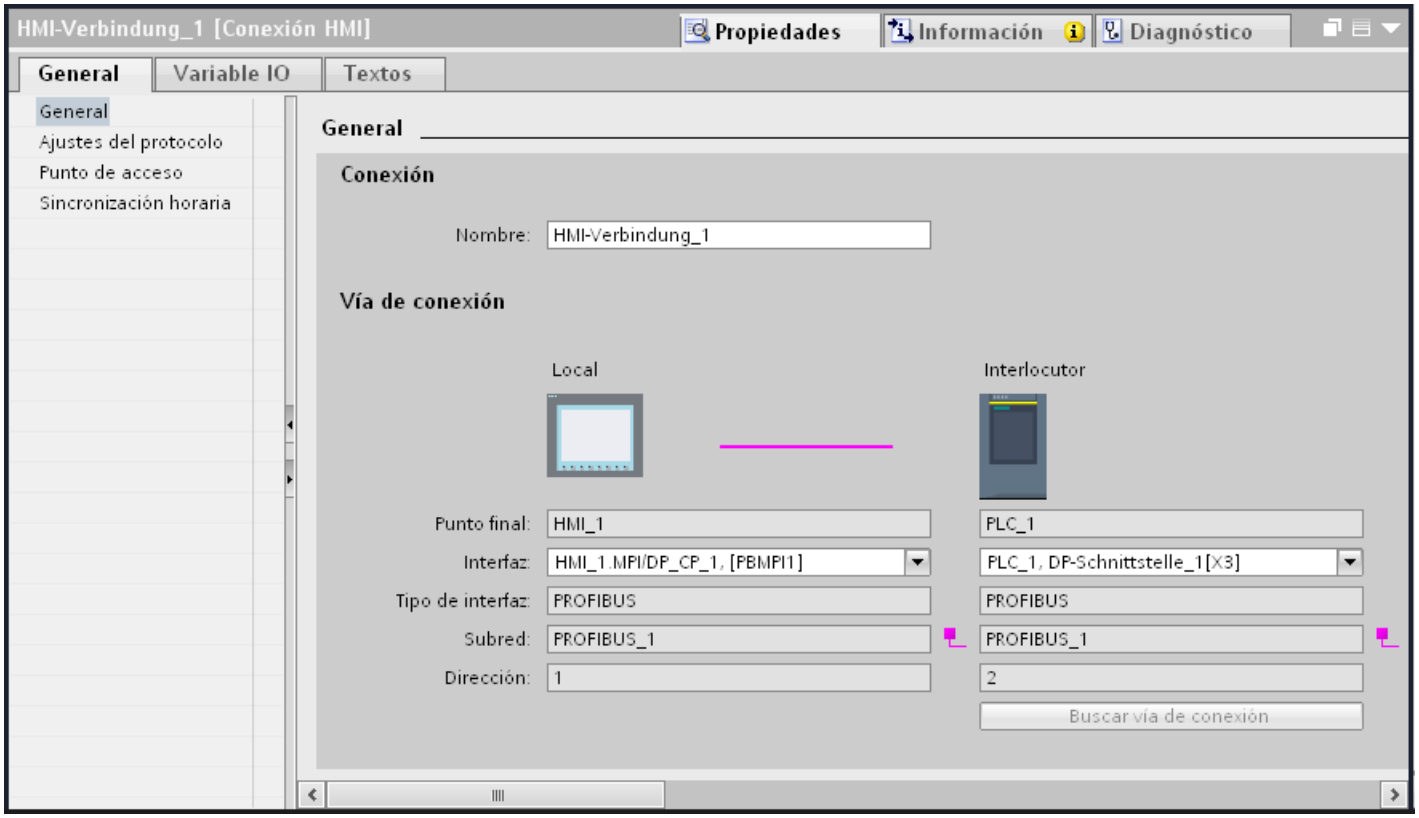
Parámetros PROFIBUS de la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



"Conexión"

Indica si los dispositivos ya están interconectados en red.

- Aparece cuando los dispositivos están interconectados en red.
- Aparece cuando los dispositivos no están interconectados en red.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros PROFIBUS. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.

- "Tipo de interfaz"
  Muestra el tipo de interfaz seleccionado. Esta área no es editable.
- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección PROFIBUS seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

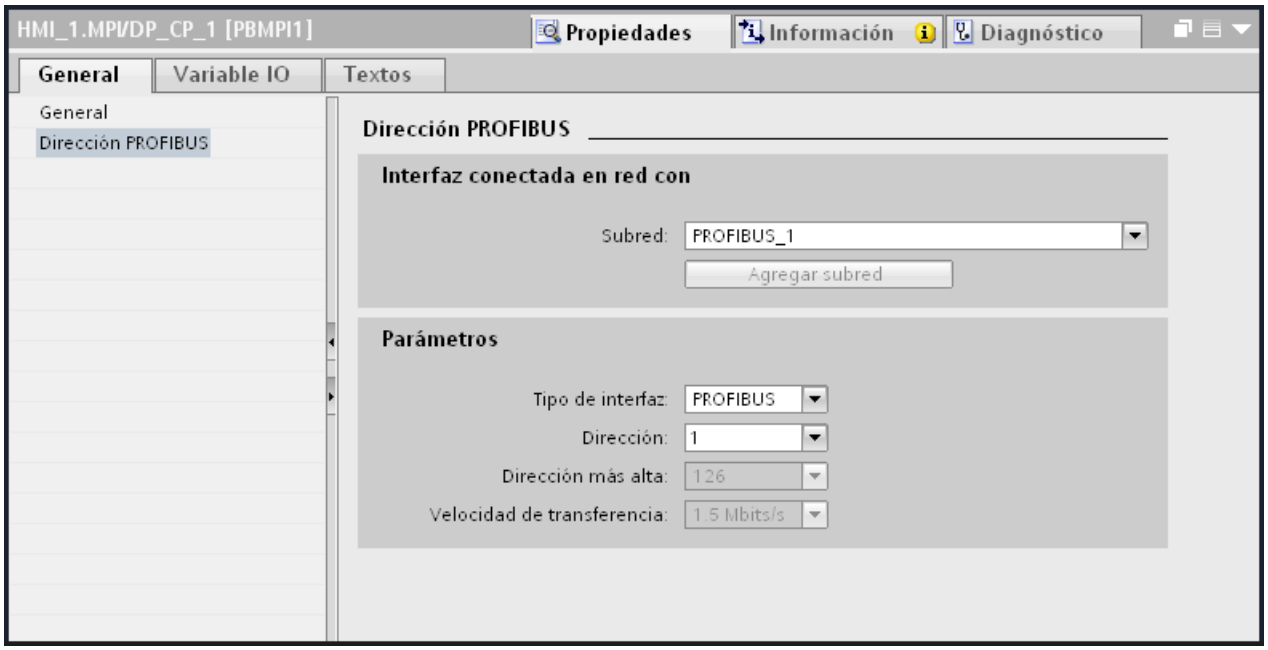
Parámetros PROFIBUS del panel de operador

Parámetros PROFIBUS del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros PROFINET del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En el área "Tipo de interfaz" se parametriza el tipo de interfaz. En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red PROFIBUS.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

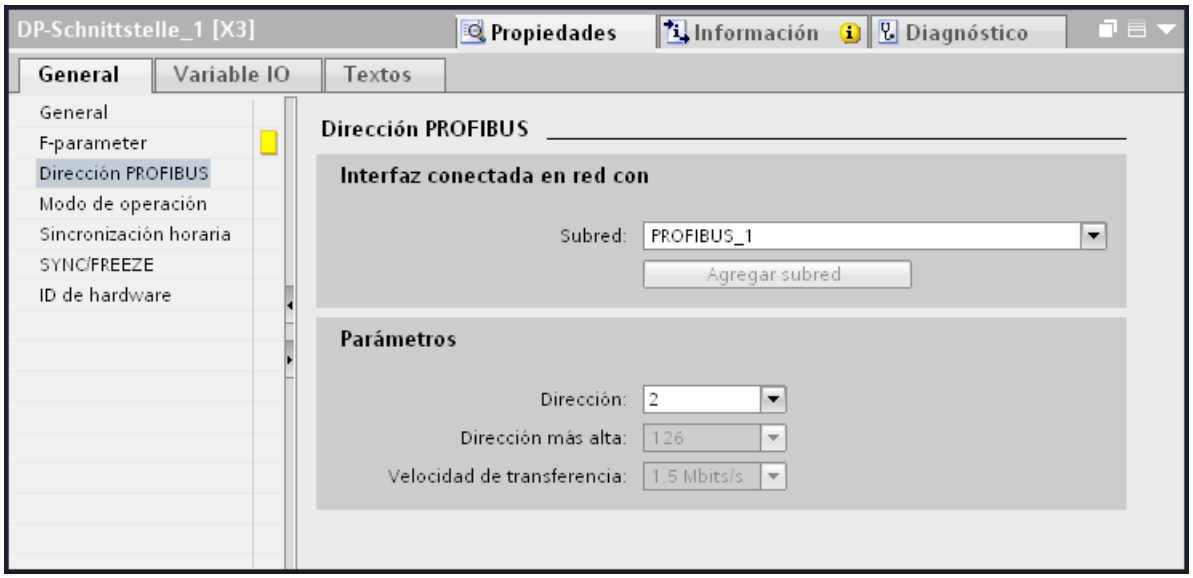
Parámetros PROFIBUS del controlador

Parámetros PROFIBUS del controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFIBUS del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el controlador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red PROFIBUS.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

Protección de la comunicación

Contraseña de acceso para la conexión HMI

Introducción

El acceso a un controlador se protege asignándole una contraseña.

Esta se asigna al configurar la contraseña.

A partir del nivel de protección "Protección completa" es imprescindible introducir la contraseña. Si no se introduce ninguna contraseña o la que se introduce es incorrecta, no se establecerá la comunicación con el controlador.

Asignar contraseña

La contraseña de acceso para el controlador se asigna en el editor "Conexiones".

Niveles de protección

Si desea proteger la comunicación del controlador y el panel de operador, puede asignar niveles de protección para la comunicación.

En una CPU SIMATIC S7-1500 es posible introducir varias contraseñas y, por tanto, establecer diferentes derechos de acceso para distintos grupos de usuarios.

Las contraseñas se introducen en una tabla, de modo que cada una tiene asignado exactamente un nivel de protección.

En la columna "Protección" se indica el efecto de la contraseña.

Para ajustar niveles de protección en el controlador SIMATIC S7-1200 hay que tener en cuenta diversos aspectos. Encontrará más información al respecto en: Auto-Hotspot

Ejemplo

Para una CPU estándar (es decir, no una CPU F) se elige el nivel "Protección completa" durante la configuración del controlador.

A continuación se introduce una contraseña propia para cada nivel de protección situado encima en la tabla.

Para los usuarios que no conocen ninguna de las contraseñas, la CPU está completamente protegida. Ni siquiera son posibles accesos HMI.

Para los usuarios que conocen una de las contraseñas parametrizadas, la repercusión depende de la fila de la tabla en la que se encuentre la contraseña:

- La contraseña de la fila 1 (sin protección) hace como si la CPU no estuviera protegida. Los usuarios que conocen esta contraseña tienen un acceso ilimitado a la CPU.
- La contraseña de la fila 2 (protección de escritura) hace como si la CPU estuviera protegida contra escritura. A pesar de conocer esta contraseña, los usuarios solo tienen acceso de lectura a la CPU.
- La contraseña de la fila 3 (protección de escritura y lectura) hace como si la CPU estuviera protegida contra escritura y lectura, de modo que los usuarios que conocen esta contraseña solo disponen de acceso HMI.

10.8.6.4 Intercambio de datos

Puntero de área

Generalidades sobre los punteros de área

Introducción

A través de un puntero de área se accede a un rango de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación.

Al evaluar los datos almacenados, el controlador y el panel de operador ejecutan acciones fijamente definidas.

Configurar punteros de área

Antes de utilizar un puntero de área, actívelos en "Conexiones > Punteros de área". Seguidamente, parametrícelos.

Encontrará más información sobre la configuración de punteros de área en:

Intercambio de datos mediante punteros de área (Página 8779)

Restricciones

Para el intercambio de datos mediante punteros de área en la comunicación con SIMATIC S7 1500 sólo pueden configurarse los siguientes tipos de datos:

- UInt y matriz de UInt
- Word y matriz de Word
- Int y matriz de Int
- "Array[0..15] of Bool" en el puntero de área "Coordinación"
- Date_And_Time
- DTL y LDT

Puntero de área "Número de imagen"

Función

Los paneles de operador depositan en el puntero de área "Número de imagen" información acerca de la imagen visualizada en el panel.

Ello permite transferir al controlador información acerca del contenido actual de la imagen en el panel de operador. En el controlador se pueden disparar determinadas reacciones, p. ej. la llamada de otra imagen.

Utilización

Antes de poder utilizar el puntero de área "Número de imagen", es necesario configurarlo y activarlo bajo "Comunicación > Conexiones". El puntero de área "Número de imagen" se puede crear únicamente en **un** controlador y depositarse allí **una sola vez**.

El número de imagen se transfiere siempre al controlador cuando se activa una nueva imagen o cuando el foco cambia de un objeto de imagen a otro dentro de una imagen.

Estructura

Este puntero de área es un área de datos de la memoria del controlador con una longitud fija de 5 palabras.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1ª palabra	Tipo de imagen actual															
2ª palabra	Número de imagen actual															
3ª palabra	Reservada															
4ª palabra	Número de campo actual															
5ª palabra	Reservada															

- Tipo de imagen actual
  "1" = imagen raíz
  "4" = ventana permanente
- Número de imagen actual
  1 a 32767
- Número de campo actual
  de 1 a 32767

Puntero de área "Fecha/hora"

Función

Este puntero de área sirve para transferir la fecha y la hora del panel de operador al controlador.

El controlador escribe la petición de control "41" ó "40" en la bandeja de peticiones.

Al evaluar la petición de control, el panel de operador escribe su fecha y hora actuales en el área de datos configurada en el puntero de área "Fecha/hora". Todos los datos se expresan en formato BCD.

Nota

Si ha configurado el puntero de área "Fecha/hora", no podrá utilizar el puntero de área "Fecha/hora PLC".

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

El área de datos "Fecha/hora" tiene la siguiente estructura:

Palabra de datos	Byte más significativo								Byte menos significativo								
	7							0	7							0	
n+0	Reservada								Hora (0 a 23)								Hora
n+1	Minuto (0 a 59)								Segundo (0 a 59)								
n+2	Reservada								Reservada								
n+3	Reservada								Día de la semana (1 a 7, 1=domingo)								Fecha
n+4	Día (1 a 31)								Mes (1 a 12)								
n+5	Año (80 a 99/0 a 29)								Reservada								

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Utilización de tipos de datos

Los tipos de datos "Date_And_Time, DTL" y "LDT" sólo pueden utilizarse con los punteros de área "Fecha/hora" y "Fecha/hora PLC".

El formato de datos utilizado del puntero de área "Fecha/hora" depende de la petición de control 40/41 utilizada.

Si no existe ninguna variable de controlador conectada a los punteros de área, o si existe una variable de controlador conectada con el tipo de datos "Array[0..5] of UInt/Word/Int", se aplica lo siguiente:

La estructura representada del puntero de área "Fecha/hora" se utiliza sólo con la petición de control 41.

Si se utiliza la petición de control 40, se utiliza el formato de datos "DATE_AND_TIME (en código BCD)" representado más abajo.

Si una variable de controlador con el tipo de datos "DATE_AND_TIME", "DTL" o "LDT" está conectada a los punteros de área "Fecha/hora" y "Fecha/hora PLC", en el puntero de área respectivo sólo se utiliza el formato de datos correspondiente.

Puntero de área "Fecha/hora PLC"

Función

Este puntero de área sirve para transferir la fecha y la hora del controlador al panel de operador. Este puntero de área se utiliza si el controlador es el maestro que determina la hora.

El controlador carga el área de datos del puntero de área. Todos los datos se expresan en formato BCD.

Según el ciclo de adquisición configurado, el panel de operador lee los datos cíclicamente y se sincroniza.

Nota

No configure un ciclo de adquisición demasiado breve para el puntero de área "Fecha/hora", puesto que ello afecta el rendimiento del panel de operador.
Recomendación: ciclo de adquisición = 1 minuto (si el proceso lo permite).

"Fecha/hora PLC" es un puntero de área global y solo se puede configurar una vez en cada proyecto.

Nota

Si ha configurado el puntero de área "Fecha/hora PLC", no podrá utilizar el puntero de área "Fecha/hora".

El área de datos "Fecha/hora" tiene la siguiente estructura:

Formato DATE_AND_TIME (codificado en BCD)

Palabra de datos	Byte más significativo			Byte menos significativo		
	7	.....	0	7	.....	0
n+0	Año (80 a 99/0 a 29)			Mes (1 a 12)		
n+1	Día (1 a 31)			Hora (0 a 23)		
n+2	Minuto (0 a 59)			Segundo (0 a 59)		
n+3	Reservada			Reservada		Día de la semana (1 a 7, 1=domingo)
n+4 1)	Reservada			Reservada		
n+5 1)	Reservada			Reservada		

1) Ambas palabras de datos deben existir en el área de datos para garantizar la concordancia con el formato de datos de WinCC flexible y evitar la lectura de información errónea.

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Utilización de tipos de datos

Los tipos de datos "Date_And_Time, DTL" y "LDT" sólo pueden utilizarse con los punteros de área "Fecha/hora" y "Fecha/hora PLC".

El formato de datos utilizado del puntero de área "Fecha/hora" depende de la petición de control 40/41 utilizada.

Si no existe ninguna variable de controlador conectada a los punteros de área, o si existe una variable de controlador conectada con el tipo de datos "Array[0..5] of UInt/Word/Int", se aplica lo siguiente:

La estructura representada del puntero de área "Fecha/hora" se utiliza sólo con la petición de control 41.

Si se utiliza la petición de control 40, se utiliza el formato de datos "DATE_AND_TIME (en código BCD)" representado más abajo.

Si una variable de controlador con el tipo de datos "DATE_AND_TIME", "DTL" o "LDT" está conectada a los punteros de área "Fecha/hora" y "Fecha/hora PLC", en el puntero de área respectivo sólo se utiliza el formato de datos correspondiente.

Puntero de área "Coordinación"

Función

El puntero de área "Coordinación" sirve para realizar las funciones siguientes:

- Detectar en el programa de control el arranque del panel de operador
- Detectar en el programa de control el modo de operación actual del panel de operador
- Detectar en el programa de control si el panel de operador está dispuesto para comunicarse

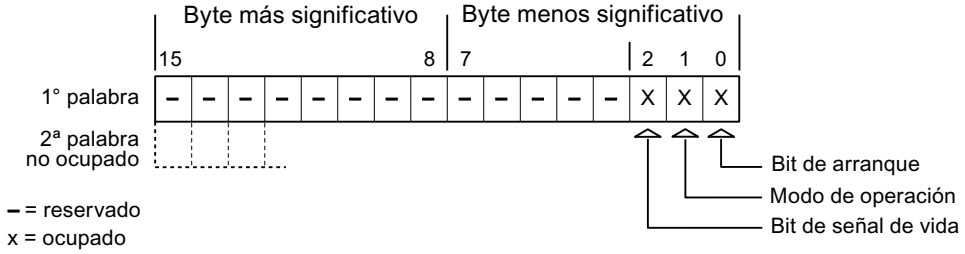
El puntero de área "Coordinación" tiene una longitud estándar de una palabra y no se puede modificar.

Utilización

Nota

Cada vez que el panel de operador actualiza el puntero de área, se escribe siempre toda el área de coordinación.
Por tanto, el programa de control no puede efectuar cambios en el área de coordinación.

Ocupación de los bits en el puntero de área "Coordinación"



Bit de arranque

Durante el arranque, el panel de operador pone el bit de arranque a "0" por breve tiempo. Después del arranque, el bit se pone a "1" de forma permanente.

Modo de operación

Cuando el usuario conmuta el panel de operador a modo "offline", el bit de modo de operación de pone a "1". En el servicio normal del panel de operador, el estado del bit de modo de operación es "0". Consultando este bit en el programa de control es posible averiguar el modo de operación actual del panel de operador.

Bit de señal de vida

El panel de operador invierte el estado del bit de señal de vida en intervalos de aprox. 1 segundo. Consultando este bit en el programa de control es posible averiguar si todavía existe una conexión con el panel de operador.

Puntero de área "Identificación del proyecto"

Función

Al iniciar runtime puede comprobarse si el panel de operador está conectado al controlador correcto. Esta comprobación es importante si se utilizan varios paneles de operador.

Para ello, el panel de operador compara un valor almacenado en el controlador con el valor indicado en la configuración. Así puede garantizarse la compatibilidad de los datos de configuración con los del programa de control. Si no hay compatibilidad, se muestra un aviso de sistema en el panel de operador y se detiene el runtime.

Utilización

Nota

Las conexiones HMI no pueden pasar a "online"

La conexión HMI en la que se utiliza el puntero de área "Identificador del proyecto" debe pasar a "online".

Para utilizar este puntero de área, especifique en la configuración lo siguiente:

- Versión de la configuración. El valor puede estar comprendido entre 1 y 255.
  La versión se indica en el editor "Configuración de runtime > General" en el área "Identificación".
- Dirección de datos del valor para la versión que está almacenada en el controlador:
  La dirección de datos se indica en el editor "Comunicación > Conexiones" en "Dirección".

Fallo de una conexión

Al fallar la conexión con un panel de operador en el que esté configurado el puntero de área "Identificador del proyecto", se desactivarán también todas las demás conexiones del dispositivo.

Para este comportamiento se deben cumplir los requisitos siguientes:

- En el proyecto se han configurado varias conexiones.
- En por lo menos una conexión se está utilizando el puntero de área "Identificador del proyecto".

Las causas siguientes pueden conmutar las conexiones al estado "offline":

- No se puede acceder al controlador.
- La conexión se ha conmutado a "offline" en el sistema de ingeniería.

Puntero de área "Petición de control"

Función

Desde la bandeja de peticiones de controlador se pueden lanzar peticiones de controlador al panel de operador y disparar con ellas acciones en el panel. Entre estas funciones se incluyen p. ej.:

- Mostrar imagen
- Ajustar la fecha y hora

Estructura de datos

En la primera palabra de la bandeja de peticiones de controlador figura el número de la petición. Dependiendo de la petición en cuestión, pueden transferirse hasta 3 parámetros como máximo.

Palabra	Byte más significativo	Byte menos significativo
n+0	0	Número de petición
n+1	Parámetro 1	
n+2	Parámetro 2	
n+3	Parámetro 3	

Si la primera palabra de la bandeja de peticiones de controlador es diferente de 0, el panel de operador evalúa la petición de controlador. Por este motivo hay que introducir primero los parámetros en la bandeja de peticiones de controlador y sólo entonces el número de la petición.

Una vez que el panel de operador ha aceptado la petición de controlador, la primera palabra se pone de nuevo a 0. Por lo general, la ejecución de la petición de control no se habrá completado todavía.

Peticiones de control

A continuación se indican todas las peticiones de control y sus respectivos parámetros. La columna "N.º" contiene el número de petición de control. Por regla general, las peticiones de control sólo pueden ser disparadas por el controlador si el panel de operador está en modo online.

N.º	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 1	Byte izquierdo: - Byte derecho: horas (0-23)
	Parámetro 2	Byte izquierdo: minutos (0-59) Byte derecho: segundos (0-59)
	Parámetro 3	-
15	Ajustar la fecha (codificada en BCD) [3) 4)]	
	Parámetro 1	Byte izquierdo: - Byte derecho: Día de la semana (1-7: Domingo-Sábado)
	Parámetro 2	Byte izquierdo: día (1-31) Byte derecho: mes (1-12)
	Parámetro 3	Byte izquierdo: año
23	Iniciar la sesión	
	Inicia la sesión del usuario "PLC User" y el número de grupo transferido en el parámetro 1 en el panel de operador. Para poder iniciar la sesión es necesario que el número de grupo transferido exista en el proyecto.	
	Parámetro 1	Número de grupo 1 - 255
	Parámetro 2, 3	-
24	Cerrar la sesión	

N.º	Función	
14	Ajustar la hora (codificada en BCD)	
	Cierra la sesión del usuario actual. (Esta función equivale a la función de sistema "CerrarSesión")	
	Parámetro 1, 2, 3	-
40	Transferir fecha/hora al controlador	
	(En formato S7 DATE_AND_TIME) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
41	Transferir fecha/hora al controlador	
	(En formato OP/MP) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
46	Actualizar variable	
	Hace que el panel de operador lea el valor actual de la variable del controlador cuyo identificador de actualización coincide con el valor transferido en el parámetro 1. (Esta función equivale a la función de sistema "ActualizarVariable")	
	Parámetro 1	1 - 100
49	Borrar búfer de avisos	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Warnings" del búfer de avisos.	
	Parámetro 1, 2, 3	-
50	Borrar búfer de avisos	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Errors" del búfer de avisos.	
	Parámetro 1, 2, 3	-
51	Selección de imagen [2]	
	Parámetro 1	Número de imagen
	Parámetro 2	-
	Parámetro 3	Número de campo
69	Leer registro del controlador [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	0: No sobrescribir el registro existente 1: Sobrescribir el registro existente
70	Escribir registro en el controlador [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	-

[1]	Sólo en paneles que soporten recetas.
[2]	En los paneles de operador OP 73, OP 77A y TP 177A la petición de control "Selección de imagen" se ejecuta incluso con el teclado de pantalla abierto.

3)	En el panel de operador KTP 600 BASIC PN se ignora el día de la semana.
4)	Si utiliza el puntero de área "Fecha/hora PLC", se ignora el día de la semana.

Puntero de área "Registro"

Puntero de área "Registro"

Función

Al transferir registros entre el panel de operador y el controlador, ambos interlocutores acceden alternativamente a las áreas de comunicación conjuntas del controlador.

Tipos de transferencia

Existen dos posibilidades para transferir registros entre el panel de operador y el controlador, a saber:

- Transferencia sin sincronización
- Transferencia con sincronización a través de la bandeja de datos

Los registros se transfieren siempre directamente. Por tanto, los valores de variables se leen o escriben directamente de/en la dirección configurada para la variable, sin desviarse al portapapeles.

Iniciar la transferencia de registros

Para iniciar la transferencia existen tres posibilidades:

- Inicio desde la vista de recetas
- Tareas de control
  El autómata también puede disparar la transferencia de registros.
- Activar funciones configuradas

Si la transferencia de registros se dispara mediante una función configurada o una orden de control, la vista de recetas seguirá pudiéndose controlar sin impedimentos desde el panel de operador. Los registros se transfieren en segundo plano.

Sin embargo, no es posible procesar simultáneamente varias peticiones de transferencia. En este caso, el panel de operador deniega una segunda transferencia con un aviso de sistema.

Transferencia sin sincronización

En la transferencia asíncrona de registros entre el panel de operador y el controlador no se coordinan las áreas de datos utilizadas conjuntamente. Por ello, no es necesario crear un área de datos en la configuración.

La transferencia asíncrona de registros resulta útil p. ej. en los casos siguientes:

- Las propiedades del sistema impiden que los interlocutores puedan sobrescribir los datos de forma incontrolada.
- El controlador no necesita información sobre el número de receta o de registro.
- La transferencia de registros se inicia por operación desde el panel de operador.

Leer valores

Al iniciar la transferencia para la lectura se leen los valores de las direcciones de control y se transfieren al panel de operador.

- Inicio por operación desde la vista de recetas:
  Los valores se cargan en el panel de operador. Éstos se pueden procesar posteriormente en el panel de operador (p. ej. se pueden modificar o guardar valores, etc.)
- Inicio por función o petición del controlador:
  Los valores se almacenan inmediatamente en el soporte de datos.

Escribir valores

Al iniciar la transferencia para escritura se escriben los valores en las direcciones de control.

- Inicio por operación desde la vista de recetas:
  Los valores actuales se escriben en el controlador.
- Inicio por función o petición del controlador:
  Los valores del soporte de datos se escriben en el controlador.

Transferencia con sincronización

En la transferencia sincronizada, ambos interlocutores activan determinados bits de estado en el área de datos utilizada conjuntamente. De este modo se impide que los datos se sobrescriban unos a otros de forma incontrolada en el programa de control.

Aplicación

La transferencia síncrona de registros resulta útil p. ej. en los casos siguientes:

- El controlador es el "interlocutor activo" en la transferencia de registros.
- En el controlador se evalúa información sobre los números de receta y de registro.
- La transferencia de registros se dispara mediante una petición del controlador.

Requisitos

Para que los registros puedan transferirse de forma sincronizada entre el panel de operador y el controlador, en el proceso de configuración deben cumplirse las siguientes condiciones:

- Se ha configurado un puntero de área: Editor "Comunicación > Conexiones" bajo "Punteros de área".
- En la receta está indicado el controlador con el que el panel de operador sincroniza la transferencia de registros:
  Editor "Recetas" en el área "General > Sincronización > Configuración" de la ventana de inspección, y selección "Transferencia de datos coordinada" activada.

Estructura del área de datos

El área de datos tiene una longitud fija de 5 palabras. El área de datos está estructurada de la siguiente manera:

	15		0
1. palabra	Número de receta actual (1-999)		
2. palabra	Número de registro actual (0-65.535)		
3. palabra	Reservada		
4. palabra	Estado (0, 2, 4, 12)		
5. palabra	Reservada		

- Estado
  La palabra de estado (palabra 4) acepta los siguientes valores:

Valor		Significado
Decimal	Binario	
0	0000 0000	Transferencia permitida, bandeja de datos libre
2	0000 0010	Transferencia en curso.
4	0000 0100	Transferencia finalizada sin fallos
12	0000 1100	Transferencia finalizada con fallos

Ejecución de la transferencia en caso de petición de control

El panel de operador o el autómata pueden iniciar la transferencia de registros entre sí.

Para este tipo de transferencia se dispone de las tareas de control n° 69 y n° 70.

N° 69: Leer registro del autómata ("SPS → DAT")

La orden de control n° 69 transfiere los registros del autómata al panel de operador. La orden de control está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	69
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	No sobrescribir el registro existente: 0 Sobrescribir el registro existente: 1	

N° 70: Escribir registro en el autómata ("DAT → SPS")

La orden de control n° 70 transfiere los registros del panel de operador al autómata. La petición del controlador está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	70
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	—	

Ejecución al leer del autómata mediante la orden de control "SPS → DAT" (n° 69)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la tarea, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la orden de control.	
4	• Si en la tarea se ha seleccionado "Sobrescribir", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia concluida". • Si en la tarea se ha seleccionado "No sobrescribir" y ya existe el registro, el panel de operador cancelará la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Ejecución al escribir en el autómata mediante la orden de control "DAT → SPS" (n° 70)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce en la bandeja de datos el número de receta y de registro indicados en la petición, así como el estado "Transferencia en curso".	Cancelación sin respuesta.
3	El panel de operador toma los valores del registro indicado en la tarea del portador de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Ejecución de la transferencia al lanzarla mediante una función configurada

Leer en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la función.	
4	• Si en la función "Sobrescribir" se ha seleccionado "Sí", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia concluida". • Si en la función "Sobrescribir" se ha seleccionado "No" y el registro ya existe, el panel de operador interrumpirá la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Escribir en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No

Paso	Acción	
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de control recoge los valores del registro indicado en la función del soporte de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Posibles causas de error en la transferencia de registros

Causas de error posibles

Si la transferencia de registros termina con errores, puede deberse a las siguientes causas:

- No se ha configurado ninguna dirección de variable en el autómata
- No es posible sobrescribir registros
- El número de receta no existe
- El número de registro no existe

Nota

El panel de operador es el único que puede activar la palabra de estado. El controlador sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el controlador sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos se ha activado a "Transferencia finalizada".
- El estado en la bandeja de datos se ha activado a "Transferencia finalizada con fallos".

Reacción a una interrupción debida a un fallo

El panel de operador reacciona a una interrupción de la transferencia de registros debida a un fallo de la siguiente forma:

- Inicio por el usuario desde la vista de recetas:
  Indicaciones en la barra de estado de la vista de recetas y salida de avisos de sistema
- Inicio mediante una función
  Emisión de avisos de sistema
- Inicio mediante una orden de control
  No hay respuesta en el panel de operador

Independientemente de ello, el usuario puede evaluar el estado de la transferencia consultando la palabra de estado en la bandeja de datos.

Curvas

Aspectos generales de las curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Auto-Hotspot

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Solicitud de curvas y transferencia de curvas

Área de solicitud de curvas

Si en el panel de operador se abre una imagen con una o varias curvas, el panel de operador activará los bits correspondientes en el área de solicitud de curvas. Una vez deseleccionada la imagen, el panel de operador desactiva los correspondientes bits en el área de solicitud de curvas.

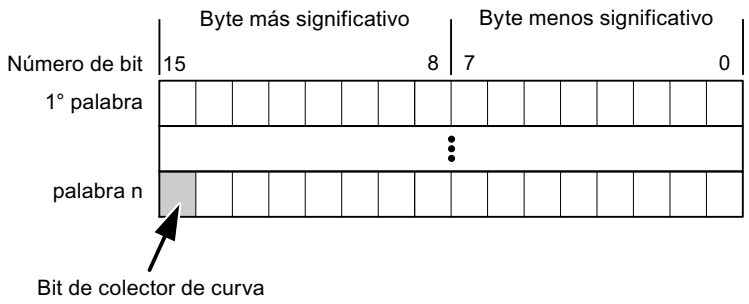
A través del área de solicitud de curvas es posible evaluar en el controlador qué curva se está representando actualmente en el panel de operador. Las curvas también se pueden disparar sin necesidad de evaluar el área de solicitud de curvas.

Área de transferencia de curvas 1

Esta área sirve para disparar curvas. En el programa de control se debe activar el bit asignado a la curva en el área de transferencia de curvas así como el bit de curvas de grupo. El bit de curvas de grupo es el último bit del área de transferencia de curvas.

El panel de operador detecta el disparo. El panel de operador lee un valor, o bien el búfer completo del controlador. A continuación, éste activa el bit de curvas y desactiva el bit de curvas de grupo.

La siguiente imagen muestra la estructura de un área de transferencia de curvas.



Mientras no se desactive el bit de curvas de grupo, el programa de control no podrá modificar el área de transferencia de curvas.

Área de transferencia de curvas 2

El área de transferencia de curvas 2 se requiere para las curvas que se configuran con el búfer alternativo. El área de transferencia de curvas 2 está estructurada del mismo modo que el área de transferencia de curvas 1.

Búfer alternativo

El búfer alternativo es un segundo búfer para la misma curva, que puede crearse en la configuración.

Mientras el panel de operador lee los valores del búfer 1, el controlador escribe en el búfer 2. Mientras el panel de operador lee el búfer 2, el controlador escribe en el búfer 1. Así se impide que se sobrescriban los valores de curva del controlador mientras el panel de operador lee la curva.

Tipos de datos admisibles para curvas

Para SIMATIC S7

Asigne un bit a cada curva en la configuración. Se admiten las variables del tipo de datos "Word" o "Int" y variables de matriz del tipo de datos "Word" o "Int".

Avisos

Aspectos generales de las curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. Dependiendo de la configuración, la lectura del valor se efectúa disparada por tiempo o por bit.

Encontrará más información al respecto en:

Auto-Hotspot

Nota

En los Basic Panels el valor se lee disparado por tiempo.

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración. Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Curvas disparadas por bit

Cuando se activa un bit de disparo en la variable Solicitud de curva, el panel de operador lee un valor de curva o un búfer de curvas completo. Este ajuste se define en la configuración. Las curvas disparadas por bit se utilizan por lo general para representar valores que se modifican rápidamente. Un ejemplo de ello es la presión de inyección en la fabricación de piezas de plástico.

Para activar curvas disparadas por bit, deben crearse durante la configuración variables externas en el editor "Variables HMI" y vincularse con áreas de curvas. El panel de operador y el controlador se comunicarán entonces a través de dichas áreas de curvas.

Se dispone de las siguientes áreas para las curvas:

- Área de solicitud de curvas
- Área de transferencia de curvas 1
- Área de transferencia de curvas 2 (sólo se requiere para el búfer alternativo)

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

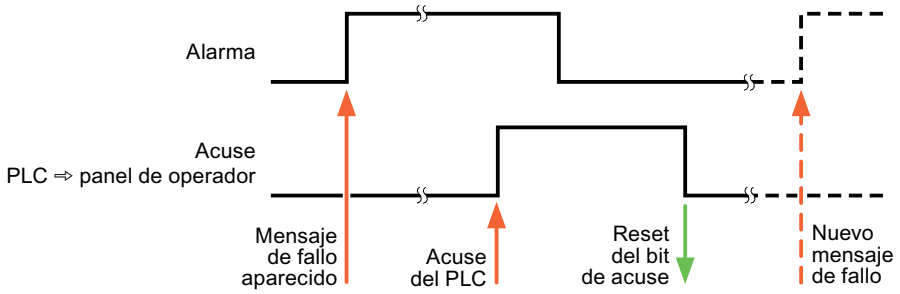
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

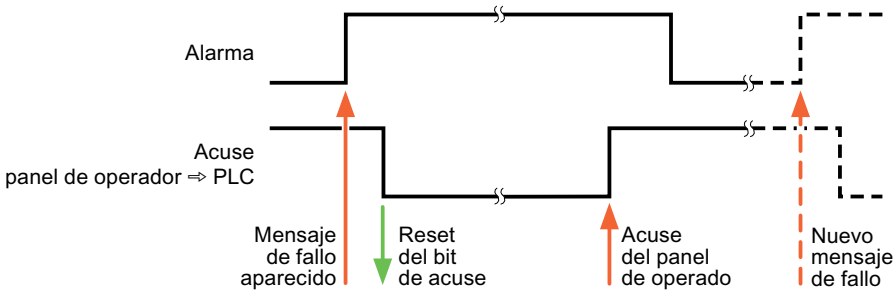
Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



10.8.6.5 Características de rendimiento de la comunicación

Dependencia del equipo SIMATIC S7 1500

Dependencia del equipo

Si se utilizan con el TIA Portal V13 dispositivos de una versión anterior del TIA Portal, puede que no sea posible configurar conexiones a determinados paneles de operador.

Basic Panels V11.0

Paneles de operador	SIMATIC S7-1500
KP300 Basic	no
KP400 Basic	no
KTP400 Basic PN	no
KTP600 Basic DP	no
KTP600 Basic PN	no
KTP1000 Basic DP	no
KTP1000 Basic PN	no
TP1500 Basic PN	no

Basic Panels V12.0

Paneles de operador	SIMATIC S7-1500
KP300 Basic	sí
KP400 Basic	sí
KTP400 Basic PN	sí
KTP600 Basic DP	sí
KTP600 Basic PN	sí
KTP1000 Basic DP	sí

Paneles de operador	SIMATIC S7-1500
KTP1000 Basic PN	sí
TP1500 Basic PN	sí

Basic Panels V13.0

Paneles de operador	SIMATIC S7-1500
KTP400 Basic	sí
KTP700 Basic	sí
KTP900 Basic	sí
KTP1200 Basic	sí

Tipos de datos admisibles para SIMATIC S7 1500

Tipos de datos admisibles para conexiones con SIMATIC S7 1500

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Longitud
BOOL	1 bit
BYTE	1 byte
WORD	2 bytes
DWORD	4 bytes
CHAR	1 byte
INT	2 bytes
DINT	4 bytes
REAL	4 bytes
TIME	4 bytes
DATE	2 bytes
TIME_OF_DAY	4 bytes
S5TIME	2 bytes
COUNTER	2 bytes
TEMPORIZADOR	2 bytes
DATE_AND_TIME	8 bytes
STRING	(2+n) bytes, n = 0 a 254
DTL	12 bytes
LDT	8 bytes
LINT	8 bytes
LREAL	8 bytes
LTIME	8 bytes

Tipo de datos	Longitud
LTIME_OF_DAY	8 bytes
SINT	1 byte
UDINT	4 bytes
UINT	2 bytes
ULINT	8 bytes
USINT	1 byte

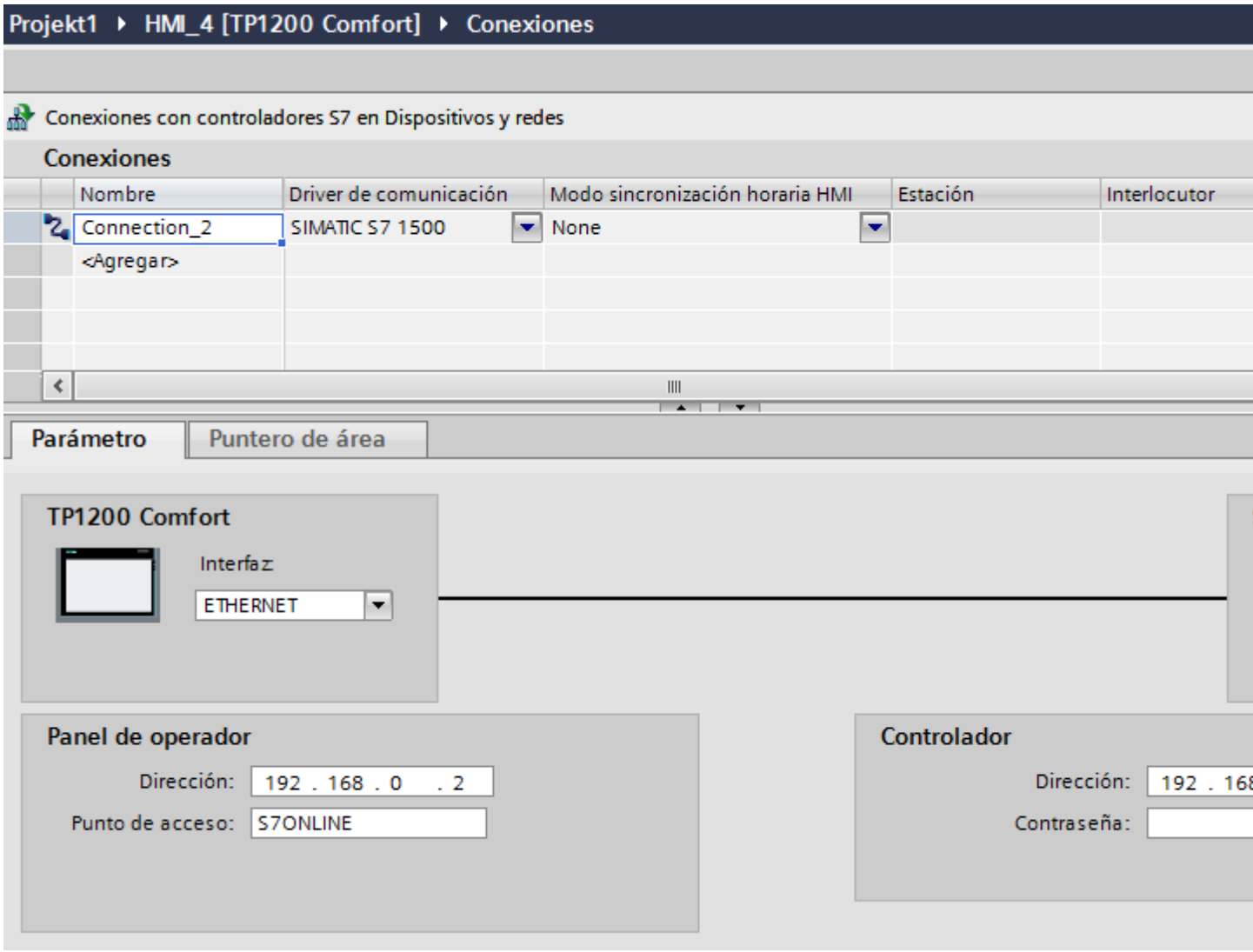
10.8.6.6 Configurar conexiones en el editor "Conexiones"

Parámetros de la conexión

Parámetros que deben ajustarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador", "Red" y "Controlador", según sea la interfaz utilizada.


Projekt1 ▸ HMI_4 [TP1200 Comfort] ▸ Conexiones
Conexiones con controladores S7 en Dispositivos y redes
Conexiones
Nombre
Driver de comunicación
Modo sincronización horaria HMI
Estación
Interlocutor
Connection_2
SIMATIC S7 1500
None
<Agregar>
Parámetro
Puntero de área
TP1200 Comfort
Interfaz:
ETHERNET
Panel de operador
Dirección: 192 . 168 . 0 . 2
Punto de acceso: S7ONLINE
Controlador
Dirección: 192 . 168
Contraseña:

Parámetros Ethernet

Parámetros del panel de operador

Ajuste en "Panel de operador" los parámetros para conectar el panel de operador a la red. Los cambios no se transfieren automáticamente al panel de operador. La configuración se debe modificar en el panel de control del panel de operador.

- "Interfaz"
  Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC.

  Nota

  Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

  Si activa "Obtener dirección IP por otra vía", en la siguiente carga se conserva en el Control Panel la dirección IP establecida previamente.

  Al transferir el proyecto se transfiere la dirección IP al panel de operador.
  La dirección IP del panel de operador se configura del siguiente modo:

  – Haga clic en el panel de operador.
  – Abra el editor "Configuración de dispositivos".
  – Haga clic en la interfaz Ethernet.
  – Asigne la dirección IP en la ventana de inspección en:
    "General > Interfaz PROFINET > Direcciones Ethernet"

- "Dirección"
  En el área "Dirección" se asigna la dirección IP del panel de operador.
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.

- "Punto de acceso"
  En "Punto de acceso" se define el punto de acceso para la interfaz PG/PC a través del cual se puede acceder al interlocutor.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección", determine la dirección IP del módulo S7 conectado al panel de operador.
- "Contraseña de acceso"
  En "Contraseña de acceso" se introduce una contraseña. La contraseña debe coincidir con la contraseña que haya introducido en el controlador.

Nota

La contraseña solo se necesita si ha guardado el nivel de protección "Protección completa" en el controlador.

Si ha guardado el nivel de protección "Protección completa" en el controlador y no introduce ninguna contraseña, no se establecerá ninguna conexión con el controlador.

Parámetros PROFIBUS

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red PROFIBUS.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

Nota

Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.

- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro).
  En el S7-200 hay que configurar un panel de operador como maestro.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros de la red

En "Red" se ajustan los parámetros de la red PROFIBUS en la que está integrado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "DP", "Universal" o "Estándar" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección PROFIBUS real más alta. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Número de maestros"
  En "Número de maestros" se ajusta el número de maestros en la red PROFIBUS. Este dato es necesario para calcular correctamente los parámetros del bus.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se determina la dirección PROFIBUS del módulo S7 (CPU, FM o CP) conectado al panel de operador.
- "Contraseña de acceso"
  En "Contraseña de acceso" se introduce una contraseña. La contraseña debe coincidir con la contraseña que haya introducido en el controlador.

Nota

La contraseña solo se necesita si ha guardado el nivel de protección "Protección completa" en el controlador.

Si ha guardado el nivel de protección "Protección completa" en el controlador y no introduce ninguna contraseña, no se establecerá ninguna conexión con el controlador.

10.8.7 Comunicación con SIMATIC S7 1200

10.8.7.1 Comunicación con SIMATIC S7 1200

Introducción

Este apartado describe la comunicación entre un panel de operador y el controlador SIMATIC S7 1200.

Para el controlador SIMATIC S7 1200 es posible configurar los siguientes canales de comunicación:

- PROFINET
- PROFIBUS

Conexión HMI para la comunicación

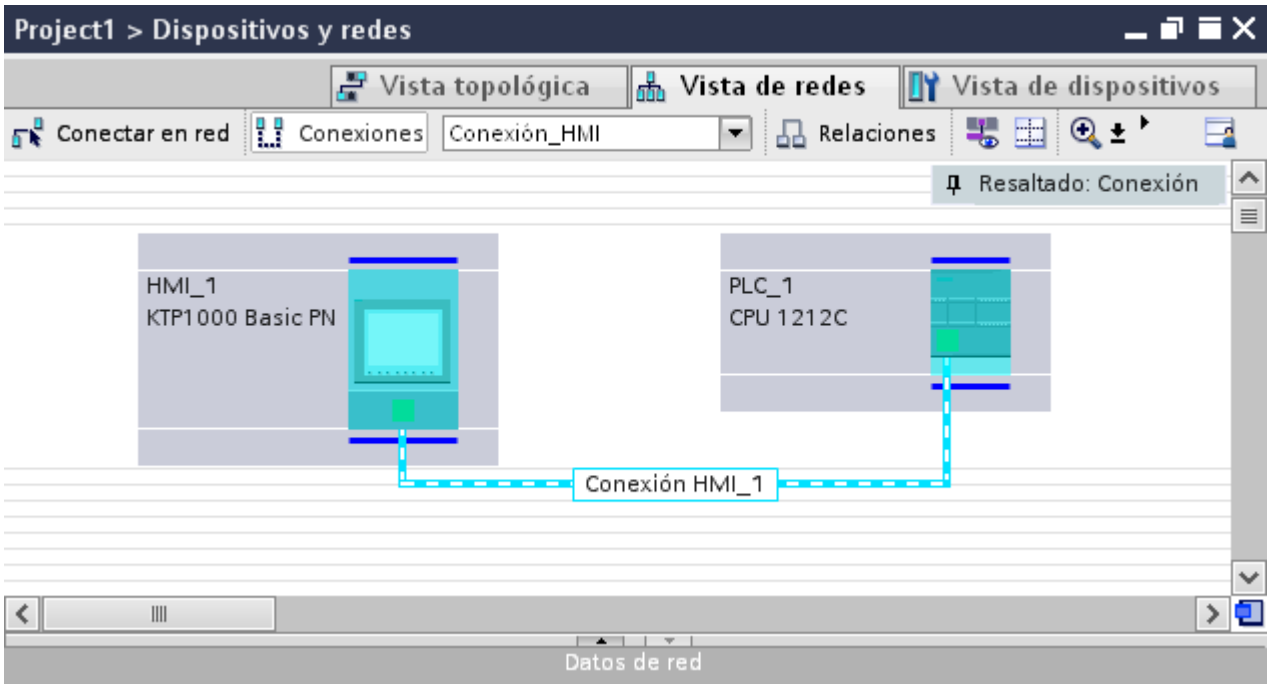
Las conexiones entre un panel de operador y un SIMATIC S7 1200 se configuran en el editor "Dispositivos y redes". Si se ha configurado un panel de operador con una conexión serie, hay que configurar un módulo de comunicación apto para PROFIBUS asociado al SIMATIC S7 1200.

10.8.7.2 Comunicación vía PROFINET

Comunicación vía PROFINET

Conexiones HMI mediante PROFINET

Si se han insertado en el proyecto un panel de operador y un SIMATIC S7 1200, interconecte las dos interfaces PROFINET en el editor "Dispositivos y redes".



También es posible conectar varios paneles de operador a un SIMATIC S7 1200 y varios SIMATIC S7 1200 a un panel de operador. El número máximo de interlocutores que pueden conectarse a un panel de operador depende del panel utilizado.

Encontrará más información al respecto en la documentación del panel de operador correspondiente.

Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFINET entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".

Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión vía PROFINET entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Configurar una conexión HMI mediante PROFINET

Introducción

Una conexión HMI vía PROFINET o Ethernet entre paneles de operador y un SIMATIC S7 1200 se configura en el editor "Dispositivos y redes".

⚠ PRECAUCIÓN
Comunicación vía Ethernet En una comunicación basada en Ethernet, el usuario final es responsable de la seguridad de su red de datos. Los ataques selectivos pueden provocar la sobrecarga del dispositivo y perjudicar su funcionalidad.

Requisitos

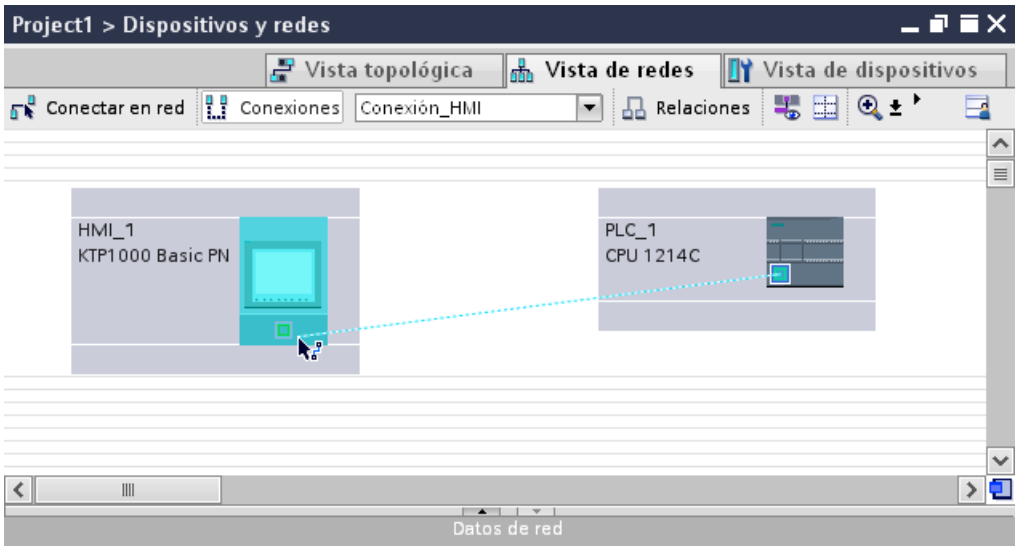
Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

- SIMATIC S7 1200
- panel de operador con interfaz PROFINET o Ethernet

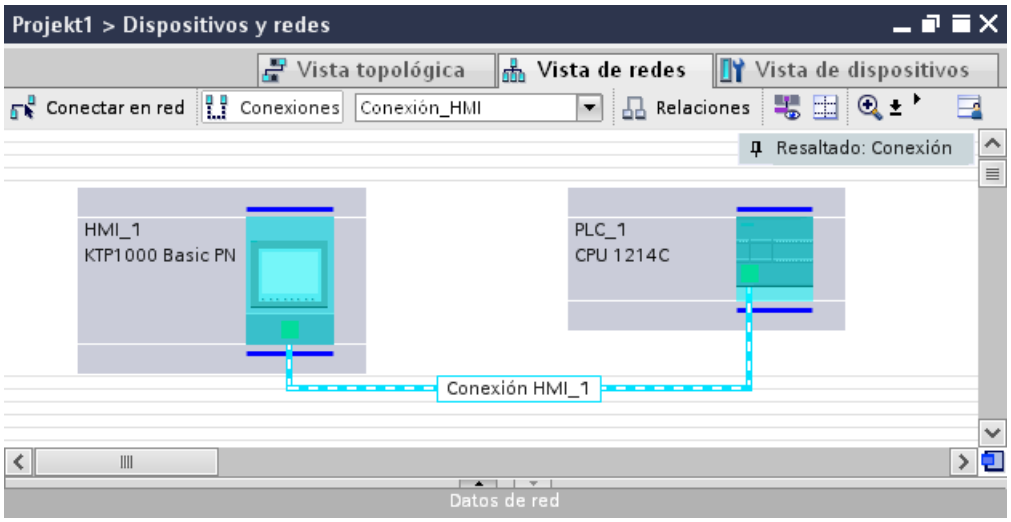
Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.
2. Haga clic en el botón "Conexiones" y elija el tipo "Conexión HMI".
   Los dispositivos disponibles para conectar se representan marcados en color.

3. Haga clic en la interfaz PROFINET del controlador y arrastre una conexión a la interfaz PROFINET o Ethernet del panel de operador utilizando la función Drag&Drop.



4. Haga clic en la línea de conexión.
5. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.



En la ventana de inspección se representa la conexión gráficamente.

6. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros PROFINET en la ventana de inspección de acuerdo con los requisitos del proyecto.
Encontrará más información al respecto en el capítulo "Parámetros PROFINET (Página 8847)".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión.

El nombre local de la conexión sólo se puede modificar en la tabla.

Resultado

Se ha creado una conexión entre un panel de operador y un SIMATIC S7 1200. Los parámetros de conexión "Dirección IP" y "Máscara de subred" están configurados.

Parámetros PROFINET

Parámetros PROFINET para la conexión HMI

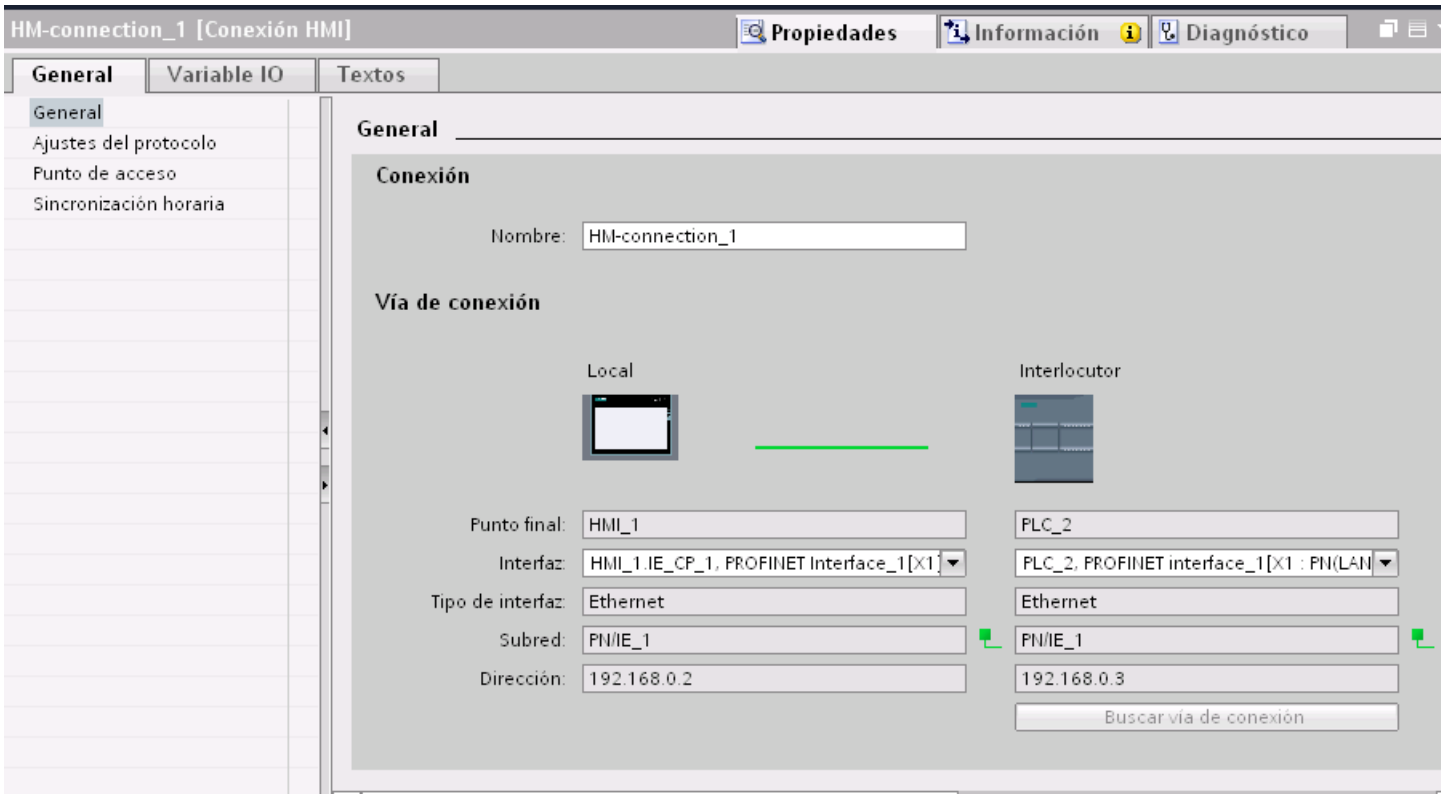
Parámetros PROFINET para la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



Conexión

En el área "Conexión" se muestra la conexión HMI que se ha creado entre los dispositivos.

El nombre de la conexión HMI se puede editar en esta área.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros PROFINET. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.
- "Tipo de interfaz"
  Muestra el tipo de interfaz seleccionado. Esta área no es editable.
- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección IP seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

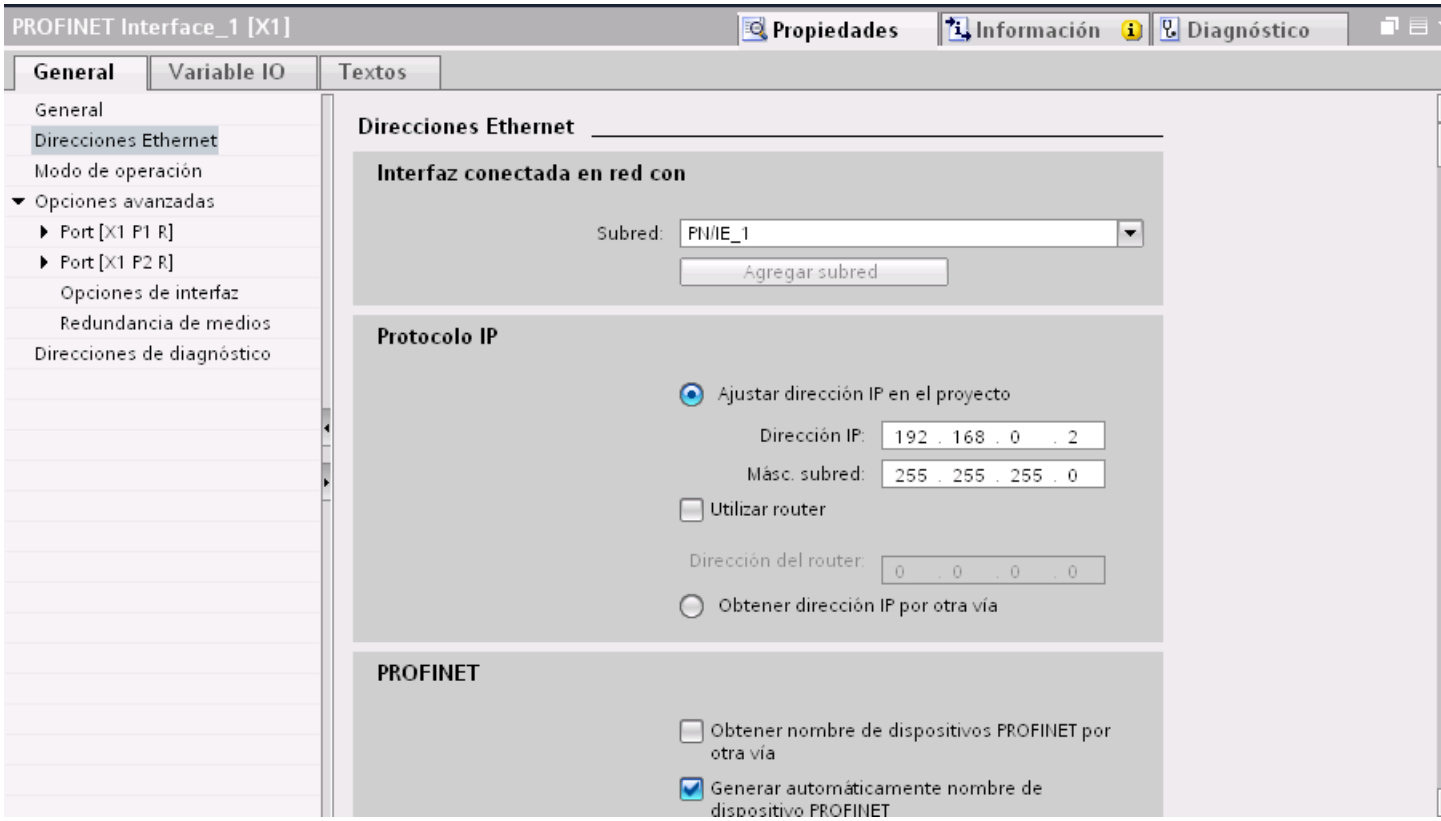
Parámetros PROFINET del panel de operador

Parámetros PROFINET del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros PROFINET del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Protocolo IP"

- "Ajustar la dirección IP en el proyecto"
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.

  Nota

  En los paneles de operador con el sistema operativo Windows CE 3.0 se realiza un reinicio automático.

  Paneles de operador con Windows CE 3.0:
  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Máscara de subred"
  En el área "Máscara de subred" se asignan los datos de la máscara de subred.
- "Utilizar router IP"
  Si utiliza un router IP, active "Utilizar router IP" e introduzca la dirección del router en el campo "Dirección del router".
- "Obtener la dirección IP de otro modo"
  Si la función "Obtener la dirección IP de otro modo" está activada, la dirección IP no se toma del proyecto. Introduzca la dirección IP directamente en el Control Panel del panel de operador.

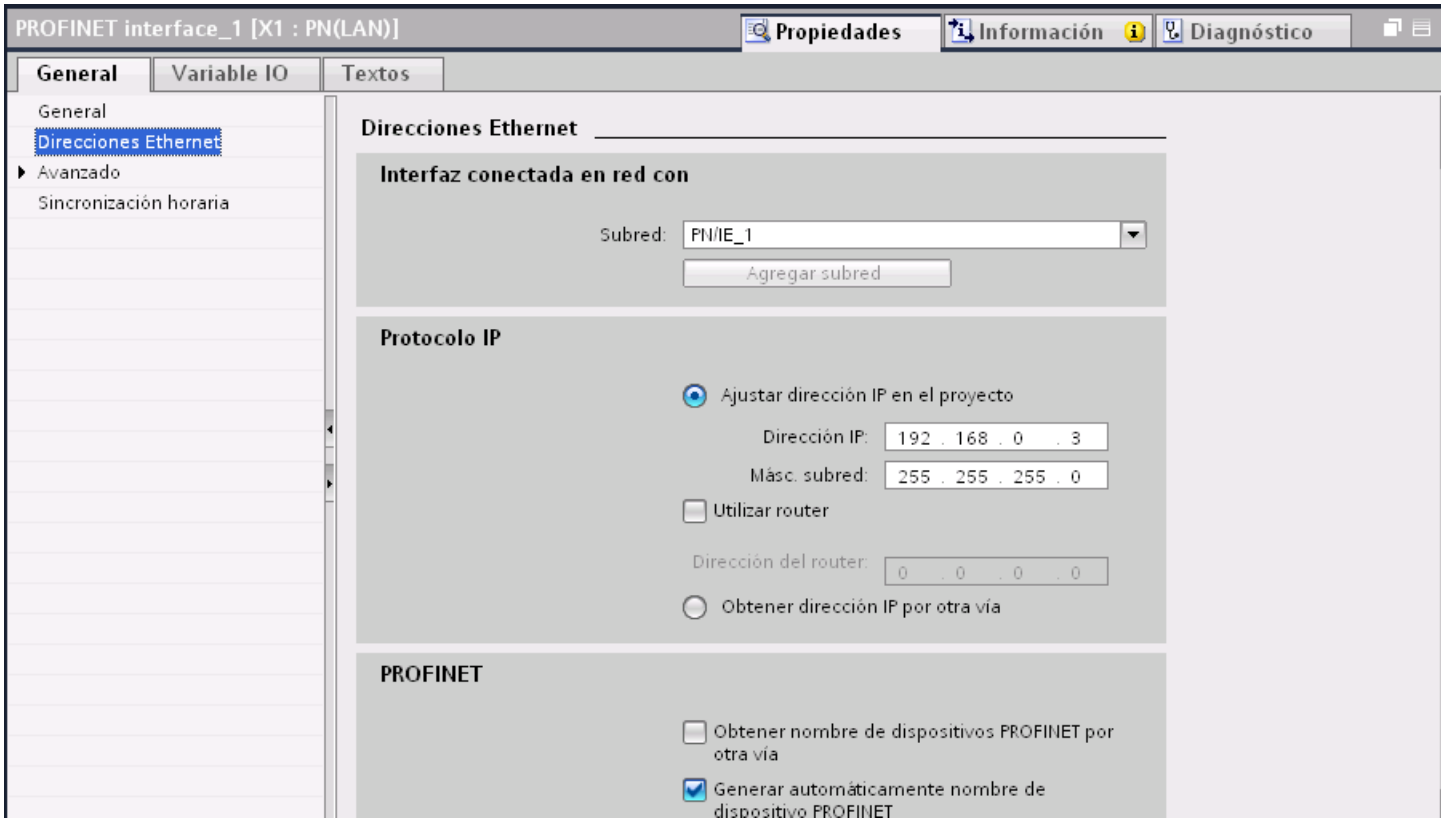
Parámetros PROFINET del controlador

Parámetros PROFINET del controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFINET del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el controlador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Protocolo IP"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección IP"
  En el área "Dirección IP" se asigna la dirección IP del panel de operador.
- "Máscara de subred"
  En el área "Máscara de subred" se asignan los datos de la máscara de subred.
  Si se utiliza un router IP, active "Utilizar router IP" e introduzca la dirección del router en el campo inferior.

Configuración de red de Industrial Ethernet

Reglas de la configuración de red

Las interfaces Ethernet de los dispositivos tienen una dirección IP predeterminada que puede modificarse.

Dirección IP

Los parámetros IP son visibles si el dispositivo apto para la comunicación soporta el protocolo TCP/IP.

La dirección IP consiste en 4 números decimales comprendidos en un rango entre 0 y 255. Los decimales están separados entre sí por un punto.

Ejemplo: 140.80.0.2

La dirección IP se compone de los datos siguientes:

- la dirección de la (sub)red
- la dirección del nodo (también se denomina host o nodo de red)

Máscara de subred

La máscara de subred separa las dos direcciones. Determina qué parte de la dirección IP direcciona la red y qué parte direcciona el nodo.

Los bits activados de la máscara de subred determinan la parte de la dirección IP correspondiente a la red.

Ejemplo:

Máscara de subred: 255.255.0.0 = 11111111.11111111.00000000.00000000

En el ejemplo superior de dirección IP, la máscara de subred tiene el significado siguiente:

Los 2 primeros bytes de la dirección IP determinan la subred, es decir, 140.80. Los 2 últimos bytes direccionan el nodo, es decir, 0.2.

Por lo general rige lo siguiente:

- La dirección de la red resulta de la combinación lógica Y de la dirección IP y la máscara de subred.
- La dirección del nodo resulta de la combinación lógica Y-NO de la dirección IP y la máscara de subred.

Relación entre la dirección IP y la máscara de subred predeterminada

Existe un acuerdo respecto de la asignación entre áreas de direcciones IP y "máscaras de subred predeterminadas". El primer decimal de la dirección IP (desde la izquierda) determina

la estructura de la máscara de subred predeterminada en cuanto al número de valores "1" (binarios), del siguiente modo:

Dirección IP (dec.)	Dirección IP (bin.)	Clase de dirección	Máscara de subred predeterminada
de 0 a 126	0xxxxxxx.xxxxxxxx....	A	255.0.0.0
de 128 a 191	10xxxxxx.xxxxxxxx...	B	255.255.0.0
de 192 a 223	110xxxxx.xxxxxxxx...	C	255.255.255.0

Nota

Rango de valores del primer decimal

En el primer decimal de la dirección IP también se acepta un valor entre 224 y 255 (clase de dirección D, etc.). Sin embargo, no es recomendable porque no se verifica la dirección de estos valores.

Enmascarar otras subredes

La máscara de subred permite seguir esctructurando una subred asignada a las clases de dirección A, B o C y crear subredes "privadas" poniendo a "1" otras posiciones de orden inferior de la máscara de subred. Por cada bit puesto a "1" se duplica el número de redes "privadas" y se divide en dos el número de nodos que contiene. Hacia fuera, la red sigue pareciendo una red individual.

Ejemplo:

En una subred de la clase de dirección B (p. ej. dirección IP 129.80.xxx.xxx), la máscara de subred predeterminada se modifica del siguiente modo:

Máscaras	Decimal	Binario
Máscara de subred predeterminada	255.255.0.0	11111111.11111111.00000000.00000000
Máscara de subred	255.255.128.0	11111111.11111111.10000000.00000000

Resultado:

Todos los nodos con direcciones comprendidas entre 129.80.001.xxx y 129.80.127.xxx están en una subred y todos los nodos con direcciones comprendidas entre 129.80.128.xxx y 129.80.255.xxx están en otra subred.

Transición de red (router)

Las transiciones de red (router) sirven para conectar las subredes. Si un datagrama IP debe enviarse a otra red, primero hay que transmitirlo a un router. Para que esto sea posible, en este caso hay que introducir la dirección del router en cada nodo de la subred.

La dirección IP de un nodo de la subred y la dirección de la transición de red (router) sólo pueden ser diferentes en las cifras que son "0" en la máscara de subred.

Protección de la comunicación

Niveles de protección

Si desea proteger la comunicación del controlador y el panel de operador, puede asignar niveles de protección para la comunicación.

En una CPU SIMATIC S7-1500 es posible introducir varias contraseñas y, por tanto, establecer diferentes derechos de acceso para distintos grupos de usuarios.

Las contraseñas se introducen en una tabla, de modo que cada una tiene asignado exactamente un nivel de protección.

En la columna "Protección" se indica el efecto de la contraseña.

Para ajustar niveles de protección en el controlador SIMATIC S7-1200 hay que tener en cuenta diversos aspectos. Encontrará más información al respecto en: Niveles de protección S7-1200

Ejemplo

Para una CPU estándar (es decir, no una CPU F) se elige el nivel "Protección completa" durante la configuración del controlador.

A continuación se introduce una contraseña propia para cada nivel de protección situado encima en la tabla.

Para los usuarios que no conocen ninguna de las contraseñas, la CPU está completamente protegida. Ni siquiera son posibles accesos HMI.

Para los usuarios que conocen una de las contraseñas parametrizadas, la repercusión depende de la fila de la tabla en la que se encuentre la contraseña:

- La contraseña de la fila 1 (sin protección) hace como si la CPU no estuviera protegida. Los usuarios que conocen esta contraseña tienen un acceso ilimitado a la CPU.
- La contraseña de la fila 2 (protección de escritura) hace como si la CPU estuviera protegida contra escritura. A pesar de conocer esta contraseña, los usuarios solo tienen acceso de lectura a la CPU.
- La contraseña de la fila 3 (protección de escritura y lectura) hace como si la CPU estuviera protegida contra escritura y lectura, de modo que los usuarios que conocen esta contraseña solo disponen de acceso HMI.

Contraseña de acceso para la conexión HMI

Introducción

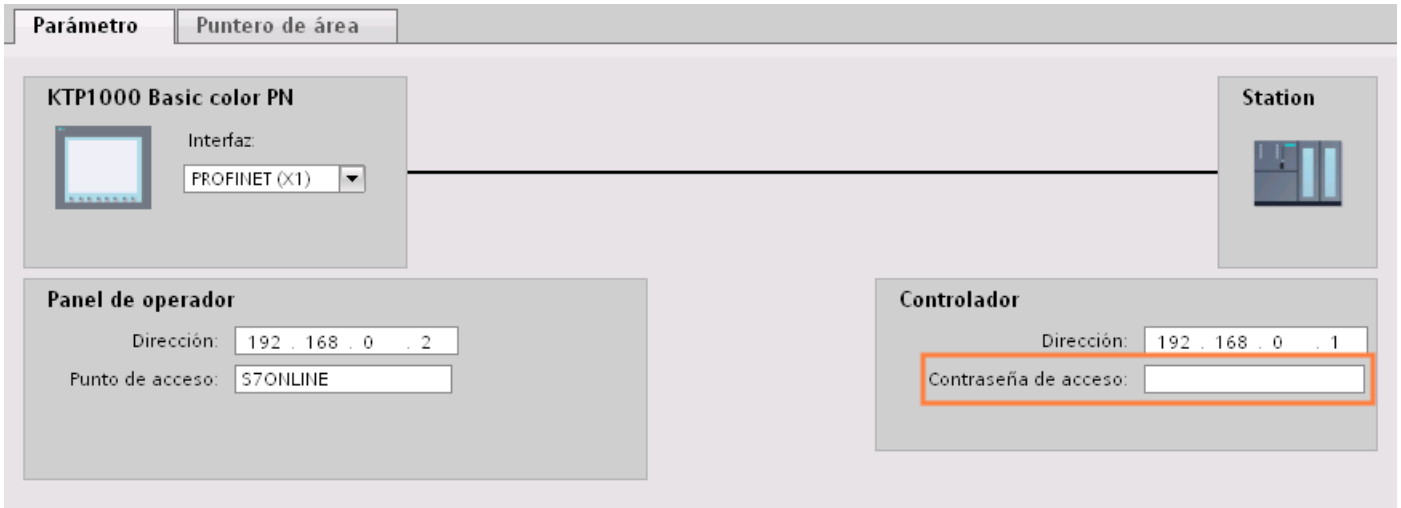
El acceso a un controlador se protege asignándole una contraseña.

Esta se asigna al configurar la contraseña.

A partir del nivel de protección "Protección completa" es imprescindible introducir la contraseña. Si no se introduce la contraseña o se introduce erróneamente, no se establece ninguna comunicación con el controlador.

Introducir contraseña de acceso

La contraseña de acceso para el controlador se asigna en el editor "Conexiones".



Definir opciones de puerto

Definición de opciones de puerto

Modificar los ajustes de conexión para el puerto PROFINET IO

En caso necesario, los ajustes de red para el puerto PROFINET IO se pueden modificar. La opción predeterminada es la definición automática de ajustes que, por norma general, permite comunicarse correctamente.

Posibilidades de ajuste de velocidad de transferencia/dúplex

Se pueden realizar los siguientes ajustes para "Velocidad de transferencia/dúplex" en función del dispositivo seleccionado:

- Ajuste automático
  Ajuste predeterminado recomendando para el puerto. La configuración de la transferencia con el puerto del interlocutor se "negocia" automáticamente. Con este ajuste también se activa automáticamente la opción "Activar autonegotiation", es decir, se puede utilizar cable cruzado o no cruzado para la conexión.
- TP/ITP con x Mbits/s. dúplex (semidúplex)
  Ajuste de la velocidad de transferencia y del modo dúplex/semidúplex. El efecto depende de la opción ajustada "Activar autonegotiation":
  - Autonegotiation activada
    Es posible utilizar tanto cable cruzado como no cruzado.
  - Autonegotiation desactivada
    Asegúrese de utilizar el cable correcto (cable cruzado o no cruzado). El puerto también se vigila con este ajuste.
- Desactivado
  En función del tipo de módulo, en la lista desplegable aparece la opción "desactivado". Así, por ejemplo, tiene la posibilidad de prohibir el acceso a un puerto no utilizado por motivos de seguridad. Con este ajuste no se generan eventos de diagnóstico.

Opción "Vigilar"

Con esta opción se activa y desactiva el diagnóstico de puerto. Ejemplos de diagnóstico de puerto: se vigila el estado del enlace, es decir, se genera el diagnóstico en caso de fallo del enlace y se vigila la reserva del sistema en los puertos de fibra óptica.

Opción "Activar autonegotiation"

El ajuste de autonegotiation sólo es modificable si se ha seleccionado un medio específico (p. ej. TP 100 Mbits/s dúplex). De las propiedades del módulo depende que un medio específico se pueda ajustar o no.

Si se ha desactivado la autonegotiation, se fuerza el ajuste fijo del puerto, de manera similar a como se requiere p. ej. en caso de arranque priorizado del dispositivo IO.

Hay que procurar realizar los mismos ajustes en el puerto del interlocutor, porque en esta opción los parámetros operativos de la red conectada no se reconocen y, en consecuencia, la velocidad de transferencia de datos y el modo de transferencia no se pueden ajustar óptimamente.

Nota

Cuando hay un puerto local interconectado, STEP 7 realiza el ajuste del puerto del interlocutor siempre y cuando soporte este ajuste. Si el puerto del interlocutor no soporta este ajuste, se genera un mensaje de error.

Normas de cableado con autonegotiation desactivada

Requisitos

Se han realizado los ajustes siguientes p. ej. para acelerar el tiempo de arranque del dispositivo IO para el puerto correspondiente:

- Velocidad de transferencia fija
- Autonegotiation incl. autocrossing desactivada

De ese modo, durante el arranque se prescinde del tiempo de negociación de la velocidad de transferencia.

Si ha desactivado Autonegotiation, debe observar las normas de cableado.

Normas de cableado con Autonegotiation desactivada

Los dispositivos PROFINET tienen los dos tipos de puertos siguientes:

Tipo de puerto	Dispositivos PROFINET	Observación
Switchport con asignación de pines cruzada	En dispositivos IO: puerto 2 En CPUs S7 con 2 puertos: puerto 1 y puerto 2	Asignación de pines cruzada significa que la asignación de pines de los puertos para envío y recepción se intercambian internamente entre los dispositivos PROFINET afectados.
Puerto de terminal con asignación de pines no cruzada	En dispositivos IO: puerto 1 En CPUs S7 con un puerto: puerto 1	-

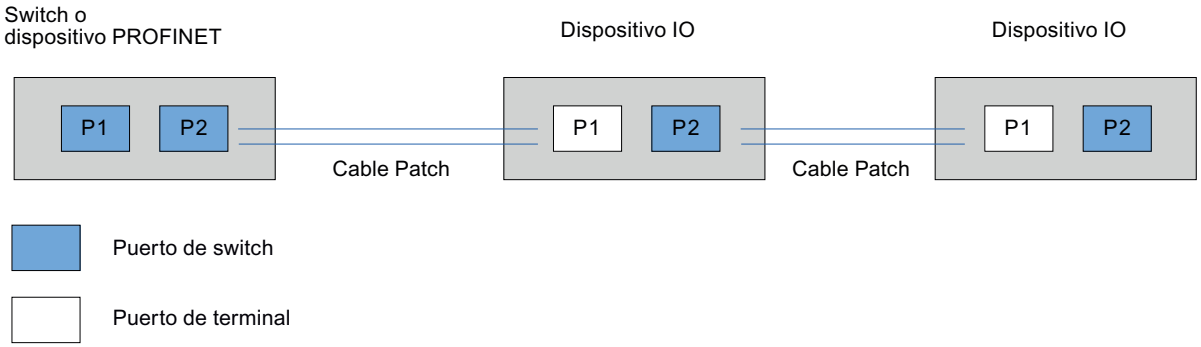
Validez de las normas de cableado

Las normas de cableado descritas en el siguiente apartado rigen exclusivamente para aquellos casos en los que se ha definido un ajuste de puerto fijo.

Normas para el cableado

Varios dispositivos IO pueden conectarse en línea con un tipo de cable (cable no cruzado). Para ello se conecta el puerto 2 del dispositivo IO (periferia descentralizada) con el puerto 1

del siguiente dispositivo IO. En el gráfico siguiente se representa un ejemplo con dos dispositivos IO.



Limitaciones en el puerto

Requisitos

Para poder trabajar con "boundaries" (limitaciones), el dispositivo correspondiente debe contar con más de un puerto. Si PROFINET no soporta limitaciones, éstas tampoco se visualizan.

Activar limitaciones

Se entiende por limitaciones los límites para transferir determinados frames en Ethernet. Es posible activar las siguientes limitaciones en un puerto:

- "Fin del registro de nodos accesibles"
  No se transfieren los frames DCP para el registro de nodos accesibles. Los nodos que están detrás de este puerto ya no se visualizan en "Nodos accesibles" del árbol del proyecto. La CPU ya no puede acceder a dichos nodos.
- "Fin de la detección de topología"
  No se transfieren los frames LLDP (Link Layer Discovery Protocol) para la detección de topología.
- "Fin del dominio Sync"
  No se transfieren frames Sync que se transmiten dentro de un dominio Sync para sincronizar los nodos.
  Si se opera p. ej. un dispositivo PROFINET con más de dos puertos en un anillo, es preciso impedir la inserción de frames Sync en el anillo activando una limitación Sync (en los puertos no incluidos en el anillo).
  Otro ejemplo: si se quieren utilizar varios dominios Sync, hay que configurar una limitación de dominio Sync para el puerto que está conectado con un dispositivo PROFINET del otro dominio Sync.

Restricciones

Hay que tener en cuenta las siguientes restricciones:

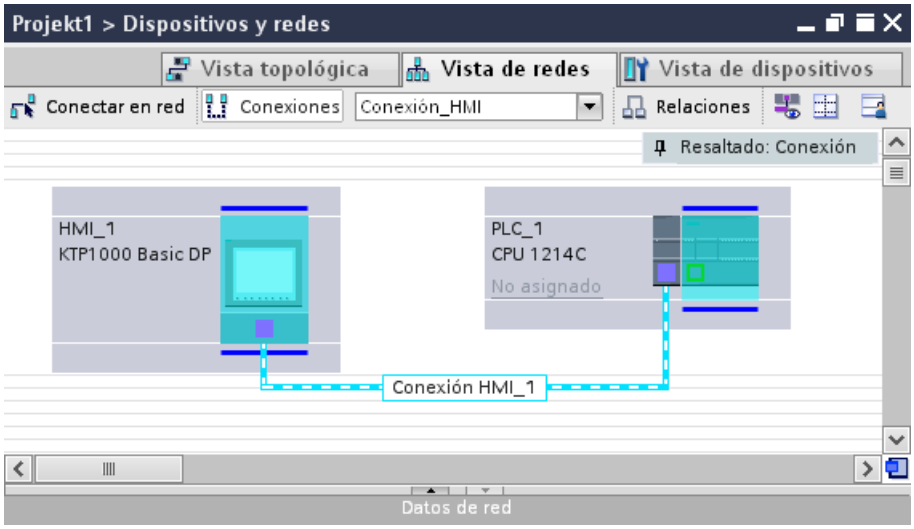
- Las casillas de verificación sólo están operativas si el puerto soporta la correspondiente característica.
- Si se ha determinado un puerto interlocutor para el puerto, las siguientes casillas de verificación no están operativas:
  - "Fin del registro de nodos accesibles"
  - "Fin de la detección de topología"
- Si la autonegotiation está desactivada, no está operativa ninguna casilla de verificación.

10.8.7.3 Comunicación vía PROFIBUS

Comunicación vía PROFIBUS

Conexiones HMI mediante PROFIBUS

Si desea conectar un SIMATIC S7 1200 con un panel de operador a través de PROFIBUS, primero hay que configurar un módulo de comunicación apto para PROFIBUS en un slot del controlador.



Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFIBUS entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".

Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Configurar una conexión HMI mediante PROFIBUS

Introducción

Una conexión HMI vía PROFIBUS entre paneles de operador y un SIMATIC S7 1200 se configura en el editor "Dispositivos y redes".

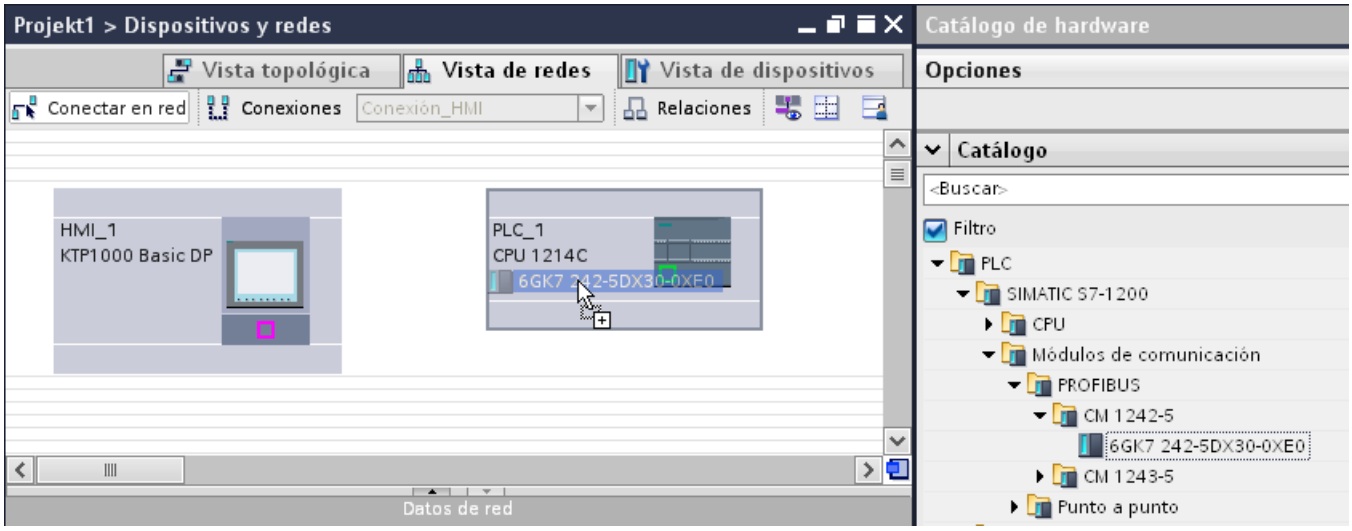
Requisitos

Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

- panel de operador con interfaz MPI/DP
- SIMATIC S7 1200

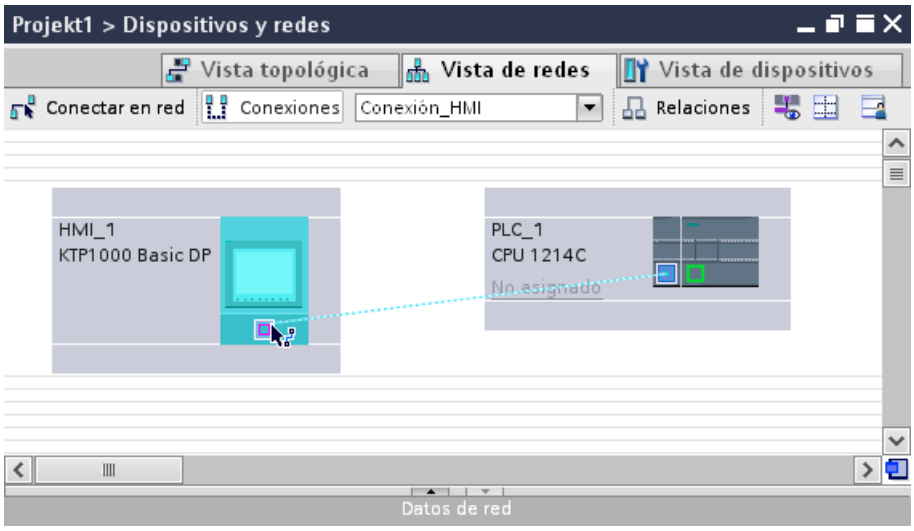
Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.
2. Haga clic en el botón "Conexiones".
   Los dispositivos disponibles para conectar se representan marcados en color.
3. Utilizando la función Drag&Drop, arrastre un módulo de comunicación apto para PROFIBUS del catálogo de hardware al controlador.



4. Haga clic en la interfaz del panel de operador.
5. Seleccione el tipo de interfaz "PROFIBUS" en la ventana de inspección "Propiedades > General > Dirección PROFIBUS/ Dirección MPI > Parámetros".

6. Haga clic en la interfaz del módulo de comunicación y arrastre una conexión hasta el panel de operador con la función Drag&Drop.



7. Haga clic en el nombre de la conexión.
   En la ventana de inspección se representa la conexión gráficamente.
8. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.
9. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros PROFIBUS en la ventana de inspección de acuerdo con los requisitos del proyecto.
   Encontrará más información al respecto en el capítulo "Parámetros PROFIBUS (Página 8861)".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión.

El nombre local de la conexión sólo se puede modificar en la tabla.

Resultado

Se ha creado una conexión HMI entre un panel de operador y un SIMATIC S7 1200 a través de PROFIBUS.

Parámetros PROFIBUS

Parámetros PROFIBUS de la conexión HMI

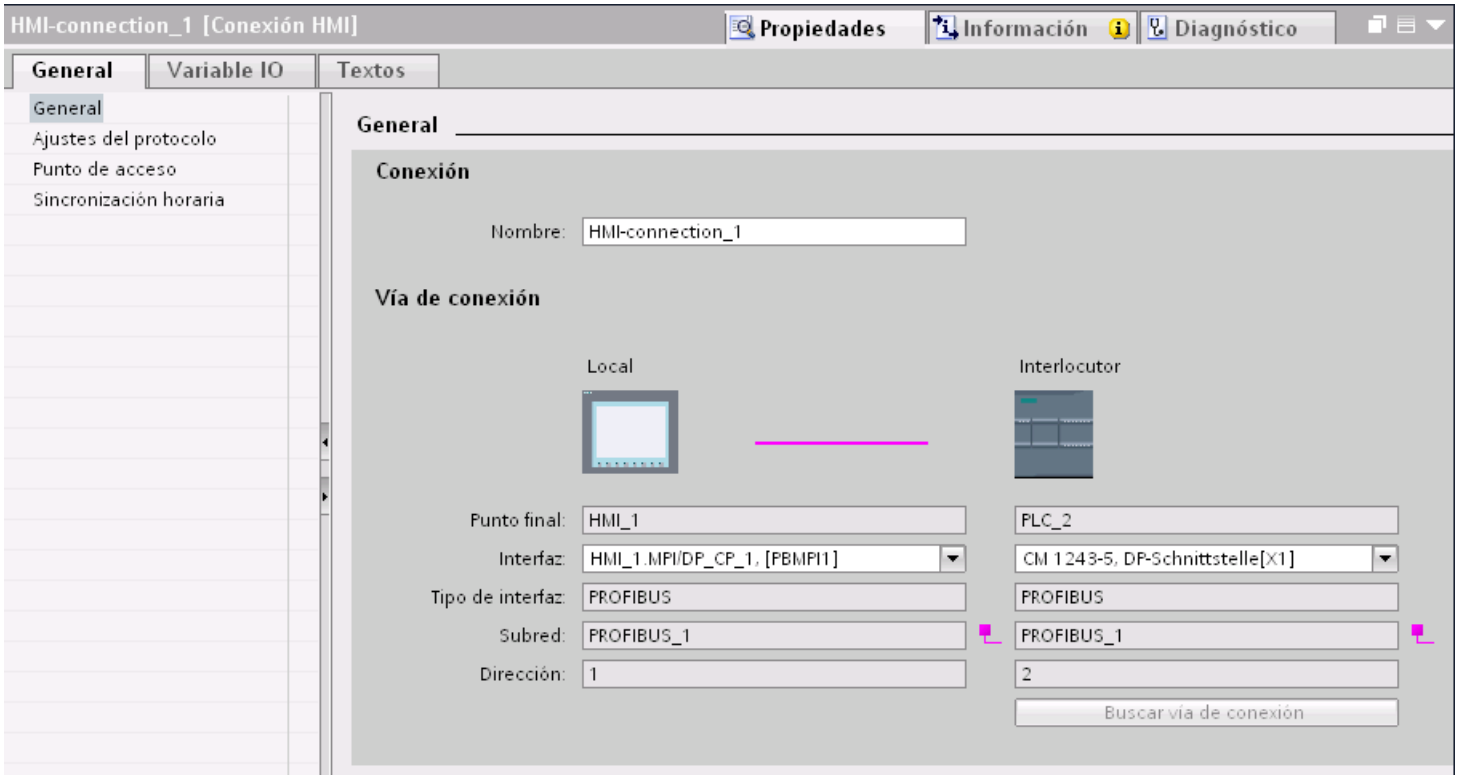
Parámetros PROFIBUS de la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



"Conexión"

En el área "Conexión" se muestra la conexión HMI que se ha creado entre los dispositivos.

El nombre de la conexión HMI se puede editar en esta área.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros PROFIBUS. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.
- "Tipo de interfaz"

Muestra el tipo de interfaz seleccionado. Esta área no es editable.

- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección PROFIBUS seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

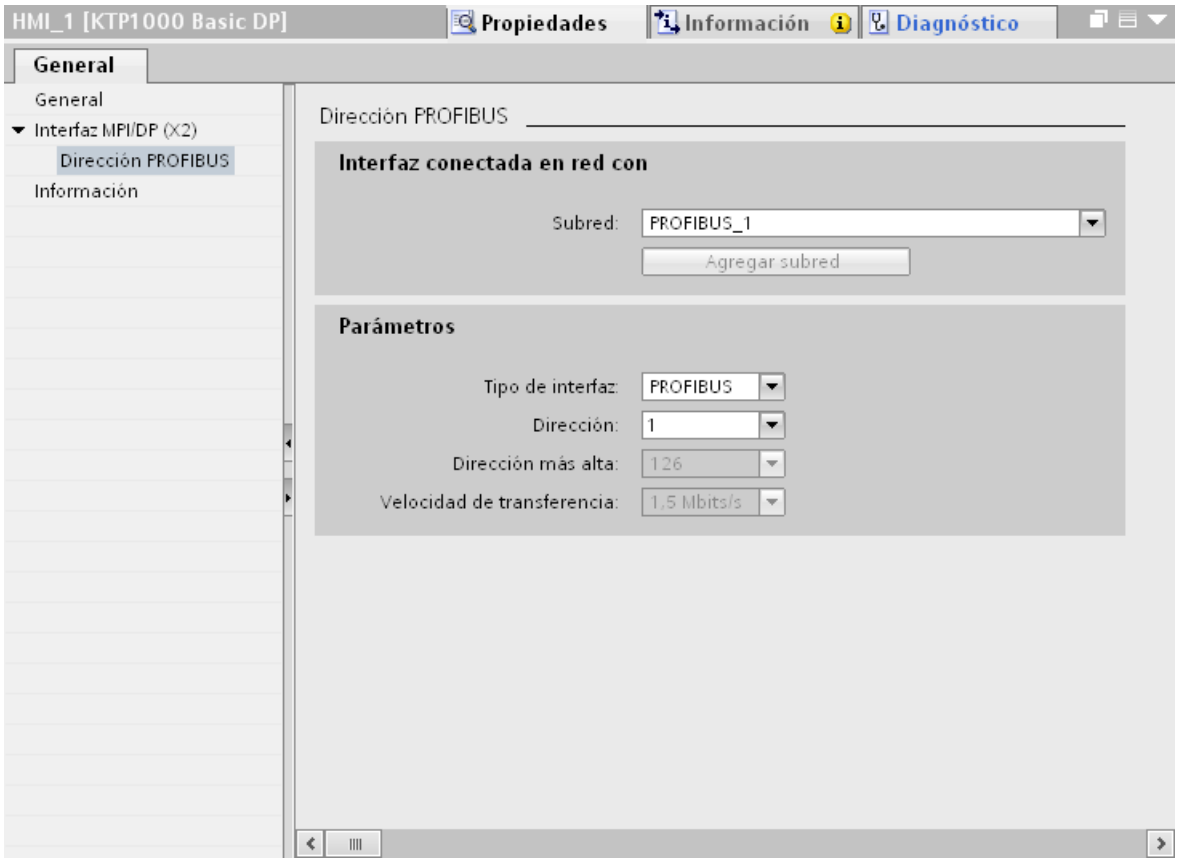
Parámetros PROFIBUS del panel de operador

Parámetros PROFIBUS del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros PROFIBUS del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red PROFIBUS.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

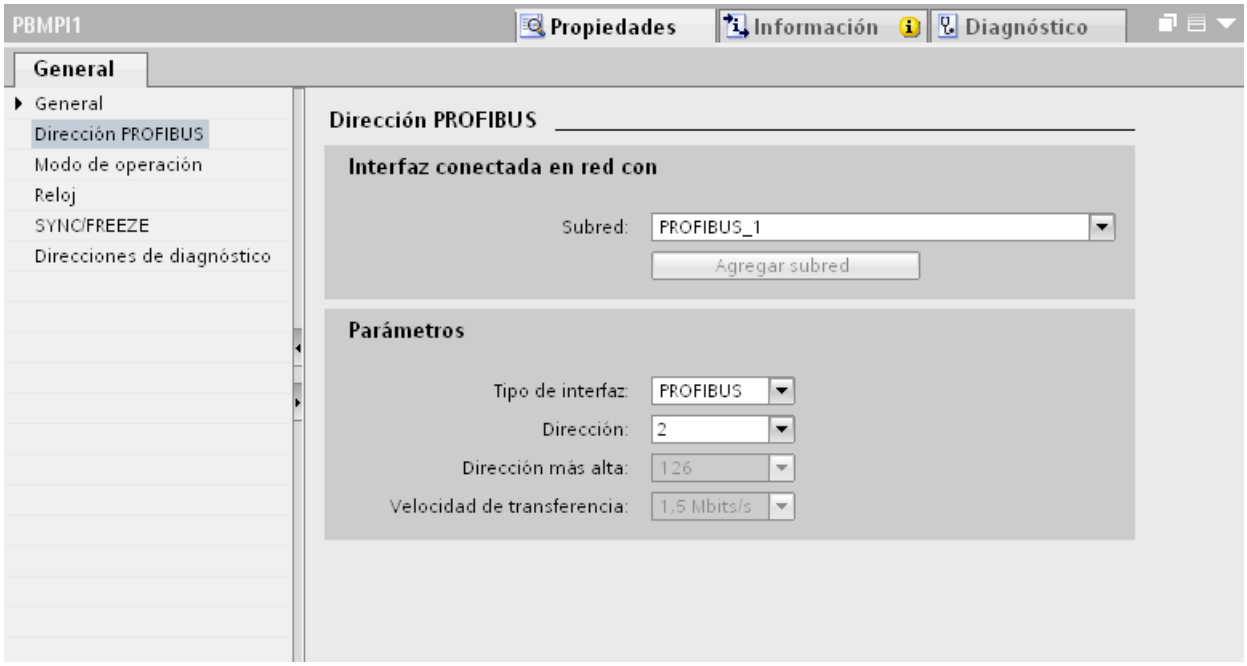
Parámetros PROFIBUS para el controlador

Parámetros PROFIBUS para el controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFIBUS del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el controlador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red PROFIBUS.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

Perfiles de bus en PROFIBUS

Introducción

Según sean los tipos de dispositivo conectados y los protocolos utilizados en PROFIBUS, hay disponibles diferentes perfiles. Los perfiles se diferencian por sus posibilidades de ajuste y por el cálculo de los parámetros de bus. A continuación se explican los perfiles.

Nodo con diferentes perfiles en la misma subred PROFIBUS

La subred PROFIBUS sólo funciona a la perfección si los parámetros de bus de todos los nodos tienen los mismos valores.

Perfiles y velocidades de transferencia

Perfiles	Velocidades de transferencia soportadas en Kbits/s
DP	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Estándar	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Universal	9,6 19,2 93,75 187,5 500 1500

Significado de los perfiles

Perfil	Significado
DP	Seleccione el perfil de bus "DP" si en la subred PROFIBUS sólo hay conectados dispositivos que cumplen con exigencias de la norma EN 50170 Volumen 2/3, Parte 8-2 PROFIBUS. El ajuste de los parámetros de bus está optimizado para dichos dispositivos. Entre ellos se incluyen dispositivos con interfaces de maestro y esclavo DP del SIMATIC S7 así como dispositivos de periferia descentralizada de otros fabricantes.
Estándar	En comparación con el perfil "DP", el perfil "Estándar" ofrece la posibilidad añadida de incluir nodos de otro proyecto o nodos no configurados aquí en el cálculo de los parámetros de bus. Seguidamente, los parámetros de bus se calculan siguiendo un algoritmo sencillo y no optimizado.
Universal	Seleccione el perfil de bus "Universal" cuando algunos nodos de la subred PROFIBUS utilicen el servicio PROFIBUS FMS. Entre ellos se incluyen, p. ej., los dispositivos siguientes: • CP 343-5 • dispositivos PROFIBUS FMS de otros fabricantes Igual que en el perfil "Estándar", en este caso también existe la posibilidad de considerar nodos adicionales al calcular los parámetros de bus.

Protección de la comunicación

Niveles de protección

Si desea proteger la comunicación del controlador y el panel de operador, puede asignar niveles de protección para la comunicación.

En una CPU SIMATIC S7-1500 es posible introducir varias contraseñas y, por tanto, establecer diferentes derechos de acceso para distintos grupos de usuarios.

Las contraseñas se introducen en una tabla, de modo que cada una tiene asignado exactamente un nivel de protección.

En la columna "Protección" se indica el efecto de la contraseña.

Para ajustar niveles de protección en el controlador SIMATIC S7-1200 hay que tener en cuenta diversos aspectos. Encontrará más información al respecto en: Auto-Hotspot

Ejemplo

Para una CPU estándar (es decir, no una CPU F) se elige el nivel "Protección completa" durante la configuración del controlador.

A continuación se introduce una contraseña propia para cada nivel de protección situado encima en la tabla.

Para los usuarios que no conocen ninguna de las contraseñas, la CPU está completamente protegida. Ni siquiera son posibles accesos HMI.

Para los usuarios que conocen una de las contraseñas parametrizadas, la repercusión depende de la fila de la tabla en la que se encuentre la contraseña:

- La contraseña de la fila 1 (sin protección) hace como si la CPU no estuviera protegida. Los usuarios que conocen esta contraseña tienen un acceso ilimitado a la CPU.
- La contraseña de la fila 2 (protección de escritura) hace como si la CPU estuviera protegida contra escritura. A pesar de conocer esta contraseña, los usuarios solo tienen acceso de lectura a la CPU.
- La contraseña de la fila 3 (protección de escritura y lectura) hace como si la CPU estuviera protegida contra escritura y lectura, de modo que los usuarios que conocen esta contraseña solo disponen de acceso HMI.

Contraseña de acceso para la conexión HMI

Introducción

El acceso a un controlador se protege asignándole una contraseña.

Esta se asigna al configurar la contraseña.

A partir del nivel de protección "Protección completa" es imprescindible introducir la contraseña. Si no se introduce ninguna contraseña o la que se introduce es incorrecta, no se establecerá la comunicación con el controlador.

Asignar contraseña

La contraseña de acceso para el controlador se asigna en el editor "Conexiones".

10.8.7.4 Intercambio de datos

Intercambio de datos mediante punteros de área

Generalidades sobre los punteros de área

Introducción

A través de un puntero de área se accede a un rango de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación.

Al evaluar los datos almacenados, el controlador y el panel de operador ejecutan acciones fijamente definidas.

Configurar punteros de área

Antes de utilizar un puntero de área, actívelos en "Conexiones > Punteros de área". Seguidamente, parametrícelos.

Encontrará más información sobre la configuración de punteros de área en:

Configurar punteros de área

Puntero de área "Fecha/hora"

Función

Este puntero de área sirve para transferir la fecha y la hora del panel de operador al controlador.

El controlador escribe la petición de control "41" en la bandeja de peticiones.

Al evaluar la petición de control, el panel de operador escribe su fecha y hora actuales en el área de datos configurada en el puntero de área "Fecha/hora".

Nota

Si ha configurado el puntero de área "Fecha/hora", no podrá utilizar el puntero de área "Fecha/hora PLC".

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

El área de datos "Fecha/hora" tiene la siguiente estructura:

Palabra de datos	Byte más significativo								Byte menos significativo								
	7							0	7							0	
n+0	Reservada								Hora (0 a 23)								Hora
n+1	Minuto (0 a 59)								Segundo (0 a 59)								
n+2	Reservada								Reservada								
n+3	Reservada								Día de la semana (1 a 7, 1=domingo)								Fecha
n+4	Día (1 a 31)								Mes (1 a 12)								
n+5	Año (80 a 99/0 a 29)								Reservada								

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Tipos de datos permitidos

Cuando configure el puntero de área "Fecha/hora", puede utilizar los siguientes tipos de datos:

- Word
- UInt
- DTL

Uso del tipo de datos "DTL"

Si se emplea el driver de comunicación S7 1200 utilice el tipo de datos "DTL". Una variable del tipo de datos "DTL" tiene una longitud de 12 bytes y guarda información sobre la fecha y la hora en una estructura predefinida.

El tipo de datos "DTL" tiene la estructura siguiente:

Byte	Componente	Tipo de datos	Rango de valores
0	Año	UINT	1970 a 2554
1			
2	Mes	USINT	0 a 12
3	Día	USINT	1 a 31
4	Día de la semana	USINT	1(domingo) a 7(sábado) El día de la semana no se tiene en cuenta al introducir valores.
5	Hora	USINT	0 a 23
6	Minuto	USINT	0 a 59
7	Segundo	USINT	0 a 59
8	Nanosegundos	UDINT	0 a 999 999 999
9			
10			
11			

El tipo de datos "DTL" soporta la indicación de la hora en el rango de nanosegundos. Como los Panels solo soportan la indicación de la hora en el rango de milisegundos, al utilizarla en los punteros de área se da la siguiente limitación:

Al transferir la indicación de la hora desde un Panel al controlador rige 1 milisegundo como unidad de tiempo mínima. El rango de microsegundos a nanosegundos del tipo de datos "DTL" se rellena con ceros.

Puntero de área "Fecha/hora PLC"

Función

Este puntero de área sirve para transferir la fecha y la hora del controlador al panel de operador. Este puntero de área se utiliza si el controlador es el maestro que determina la hora.

El controlador carga el área de datos del puntero de área.

Según el ciclo de adquisición configurado, el panel de operador lee los datos cíclicamente y se sincroniza.

Nota

No configure un ciclo de adquisición demasiado breve para el puntero de área "Fecha/hora PLC", puesto que ello afecta el rendimiento del panel de operador.
Recomendación: Ciclo de adquisición = 1 minuto (si el proceso lo permite).

"Fecha/hora PLC" es un puntero de área global y solo se puede configurar una vez en cada proyecto.

Nota

Si ha configurado el puntero de área "Fecha/hora PLC", no podrá utilizar el puntero de área "Fecha/hora".

El área de datos "Fecha/hora" tiene la siguiente estructura:

Formato DATE_AND_TIME (codificado en BCD)

Palabra de datos	Byte más significativo			Byte menos significativo		
	7	. . . . .	0	7	. . . . .	0
n+0	Año (80 a 99/0 a 29)			Mes (1 a 12)		
n+1	Día (1 a 31)			Hora (0 a 23)		
n+2	Minuto (0 a 59)			Segundo (0 a 59)		
n+3	Reservada			Reservada		Día de la semana (1 a 7, 1=domingo)
n+4 1)	Reservada			Reservada		
n+5 1)	Reservada			Reservada		

1) Ambas palabras de datos deben existir en el área de datos para garantizar la concordancia con el formato de datos de WinCC flexible y evitar la lectura de información errónea.

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Tipos de datos permitidos

Cuando configure el puntero de área "Fecha/hora PLC", puede utilizar los siguientes tipos de datos:

- DTL

Uso del tipo de datos "DTL"

Si se emplea el driver de comunicación S7 1200 utilice el tipo de datos "DTL". Una variable del tipo de datos "DTL" tiene una longitud de 12 bytes y guarda información sobre la fecha y la hora en una estructura predefinida.

El tipo de datos "DTL" tiene la estructura siguiente:

Byte	Componente	Tipo de datos	Rango de valores
0	Año	UINT	1970 a 2554
1			
2	Mes	USINT	0 a 12
3	Día	USINT	1 a 31
4	Día de la semana	USINT	1(domingo) a 7(sábado) El día de la semana no se tiene en cuenta al introducir valores.
5	Hora	USINT	0 a 23
6	Minuto	USINT	0 a 59
7	Segundo	USINT	0 a 59
8	Nanosegundos	UDINT	0 a 999 999 999
9			
10			
11			

Los paneles de operador no soportan el uso de nanosegundos. Durante el procesamiento en runtime se ignorarán los valores que estén en el rango de los nanosegundos.

El tipo de datos "DTL" soporta la indicación de la hora en el rango de nanosegundos. Como los Panels solo soportan la indicación de la hora en el rango de milisegundos, al utilizarla en los punteros de área se da la siguiente limitación:

Al transferir la indicación de la hora desde un controlador a un Panel se ignora el rango de microsegundos a nanosegundos. En el panel se posprocesa la indicación de la hora, milisegundos incluidos.

Puntero de área "Coordinación"

Función

El puntero de área "Coordinación" sirve para realizar las funciones siguientes:

- Detectar en el programa de control el arranque del panel de operador
- Detectar en el programa de control el modo de operación actual del panel de operador
- Detectar en el programa de control si el panel de operador está dispuesto para comunicarse

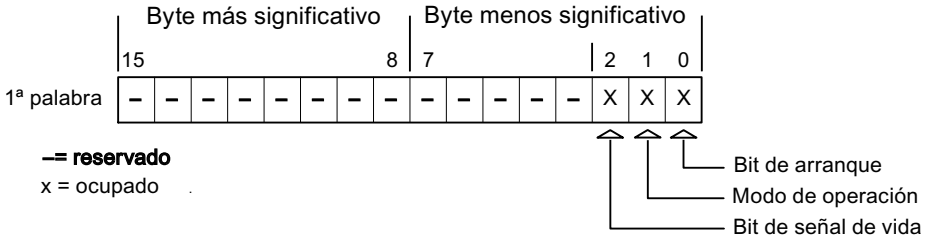
El puntero de área "Coordinación" tiene una longitud estándar de una palabra y no se puede modificar.

Utilización

Nota

Cada vez que el panel de operador actualiza el puntero de área, se escribe siempre toda el área de coordinación.
Por tanto, el programa de control no puede efectuar cambios en el área de coordinación.

Ocupación de los bits en el puntero de área "Coordinación"



Bit de arranque

Durante el arranque, el panel de operador pone el bit de arranque a "0" por breve tiempo. Después del arranque, el bit se pone a "1" de forma permanente.

Modo de operación

Cuando el usuario conmuta el panel de operador a modo "offline", el bit de modo de operación de pone a "1". En el servicio normal del panel de operador, el estado del bit de modo de operación es "0". Consultando este bit en el programa de control es posible averiguar el modo de operación actual del panel de operador.

Bit de señal de vida

El panel de operador invierte el estado del bit de señal de vida en intervalos de aprox. 1 segundo. Consultando este bit en el programa de control es posible averiguar si todavía existe una conexión con el panel de operador.

Procesamiento en el controlador

Para facilitar las posibilidades de evaluación en el programa de control, utilice una matriz Bool para este puntero de área si emplea el driver de comunicación SIMATIC S7 1200. Tendrá que reproducir toda la palabra de 16 bits del puntero de área. Por tanto, configure una variable del tipo de datos "Array [0 .. 15] of bool" en el programa de control.

Tipos de datos permitidos

Cuando configure el puntero de área "Coordinación", puede utilizar los siguientes tipos de datos:

- Word
- UInt
- Bool

Puntero de área "Número de imagen"

Función

Los paneles de operador depositan en el puntero de área "Número de imagen" información acerca de la imagen visualizada en el panel.

Ello permite transferir al controlador información acerca del contenido actual de la imagen en el panel de operador. En el controlador se puede disparar determinadas respuestas, p. ej., la llamada de otra imagen.

Utilización

Antes de poder utilizar el puntero de área "Número de imagen" es preciso configurarlo y activarlo bajo "Conexiones ▸ Punteros de área". El puntero de área "Número de imagen" se puede crear únicamente en **un** controlador y depositarse allí **una sola vez**.

El número de imagen se transfiere siempre al controlador cuando se activa una nueva imagen o cuando el foco cambia de un objeto de imagen a otro dentro de una imagen.

Estructura

Este puntero de área es un área de datos de la memoria del controlador con una longitud fija de 5 palabras.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1. palabra	Tipo de imagen actual															
2. palabra	Número de imagen actual															
3. palabra	Reservada															
4ª palabra	Número de campo actual															
5. palabra	Reservada															

- Tipo de imagen actual
  "1" = imagen raíz
  "4" = ventana permanente
- Número de imagen actual
  1 a 32767
- Número de campo actual
  1 a 32767

Nota

Dependencia del equipo

Las ventanas permanentes no están disponibles en Basic Panels.

Tipos de datos permitidos

Cuando configure el puntero de área "Número de imagen", puede utilizar los siguientes tipos de datos:

- Word
- UInt

Puntero de área "Identificación del proyecto"

Función

Al iniciar runtime puede comprobarse si el panel de operador está conectado al controlador correcto. Esta comprobación es importante si se utilizan varios paneles de operador.

Para ello, el panel de operador compara un valor almacenado en el controlador con el valor indicado en la configuración. Así puede garantizarse la compatibilidad de los datos de configuración con los del programa de control.

Si no hay coincidencia, se emite un aviso en correspondencia y el runtime no se inicia.

Utilización

Nota

Las conexiones HMI no pueden pasar a "online"

La conexión HMI en la que se utiliza el puntero de área "Identificador del proyecto" debe pasar a "online".

Para utilizar este puntero de área, especifique en la configuración lo siguiente:

- Versión de la configuración. El valor puede estar comprendido entre 1 y 255.
  La versión se indica en el editor "Configuración de runtime > General" en el área "Identificación".
- La variable de controlador o la matriz de variables que se ha configurado como rango de datos para el puntero de área.

Fallo de una conexión

Al fallar la conexión con un panel de operador en el que esté configurado el puntero de área "Identificador del proyecto", se desactivarán también todas las demás conexiones del dispositivo.

Para este comportamiento se deben cumplir los requisitos siguientes:

- En el proyecto se han configurado varias conexiones.
- En por lo menos una conexión se está utilizando el puntero de área "Identificador del proyecto".

Las causas siguientes pueden conmutar las conexiones al estado "offline":

- no se puede acceder al controlador.
- la conexión se ha conmutado a "offline" en el sistema de ingeniería.

Tipos de datos permitidos

Cuando configure el puntero de área "Identificación del proyecto", puede utilizar los siguientes tipos de datos:

- Word
- UInt

Puntero de área "Petición de control"

Función

Desde la bandeja de peticiones de control se pueden lanzar peticiones al panel de operador y con ello disparar acciones en el panel. A estas funciones pertenecen p. ej.:

- Mostrar imagen
- Ajustar la fecha y hora

Estructura de datos

En la primera palabra de la bandeja de peticiones de control figura el número de la petición. Dependiendo de la petición en cuestión, pueden transferirse hasta 3 parámetros como máximo.

Palabra	Byte más significativo	Byte menos significativo
n+0	0	Número de petición
n+1	Parámetro 1	
n+2	Parámetro 2	
n+3	Parámetro 3	

Si la primera palabra de la bandeja de peticiones de control es diferente de 0, el panel de operador evaluará la petición. Por este motivo hay que introducir primero los parámetros en la bandeja de peticiones de control y sólo entonces el número de la petición.

Cuando el panel de operador haya aceptado la petición de control, la primera palabra se pondrá de nuevo a 0. Por lo general, la ejecución de la petición de control no se habrá completado todavía.

Peticiones de control

A continuación se indican todas las peticiones de control y sus respectivos parámetros. La columna "N.º" contiene el número de petición de control. Por regla general, las peticiones de

control sólo pueden ser disparadas por el controlador si el panel de operador está en modo online.

N.°	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 1	Byte izquierdo: - Byte derecho: horas (0-23)
	Parámetro 2	Byte izquierdo: Minutos (0-59) Byte derecho: Segundos (0-59)
	Parámetro 3	-
15	Ajustar la fecha (codificada BCD)	
	Parámetro 1	Byte izquierdo: - Byte derecho: Día de la semana (1-7: Domingo-Sábado)
	Parámetro 2	Byte izquierdo: Día (1-31) Byte derecho: Mes (1-12)
	Parámetro 3	Byte izquierdo: Año
23	Iniciar la sesión	
	Inicia la sesión del usuario "PLC User" y el número de grupo transferido en el parámetro 1 en el panel de operador. Para poder iniciar la sesión es necesario que el número de grupo transferido exista en el proyecto.	
	Parámetro 1	Número de grupo 1 - 255
	Parámetro 2, 3	-
24	Cerrar la sesión	
	Cierra la sesión del usuario actual. (esta función equivale a la función de sistema "CerrarSesión")	
	Parámetro 1, 2, 3	-
40	Transferir fecha/hora al controlador	
	Entre dos peticiones deberán transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
41	Transferir fecha/hora al controlador	
	Entre dos peticiones deberán transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
46	Actualizar variable	
	Hace que el panel de operador lea el valor actual de la variable del controlador cuyo identificador de actualización coincide con el valor transferido en el parámetro 1. (Esta función equivale a la función de sistema "ActualizarVariable")	
	Parámetro 1	1 - 100
49	Borrar búfer de avisos de operación	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Warnings" del búfer de avisos.	
	Parámetro 1, 2, 3	-
50	Borrar búfer de alarmas	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Errors" del búfer de avisos.	

N.°	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 1, 2, 3	-
51	Selección de imagen	
	Parámetro 1	Número de imagen
	Parámetro 2	-
	Parámetro 3	Número de campo
69	Leer registro del controlador [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	0: No sobrescribir el registro existente 1: Sobrescribir el registro existente
70	Escribir registro en el controlador [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	-

[1]	Sólo en paneles que soporten recetas
[2]	En los paneles de operador OP 73, OP 77A y TP 177A la petición de control "Selección de imagen" se ejecuta incluso con el teclado de pantalla abierto.
[3]	En el panel de operador KTP 600 BASIC PN se ignora el día de la semana.

Tipos de datos permitidos

Cuando configure el puntero de área "Número de imagen", puede utilizar los siguientes tipos de datos:

- Word
- UInt

Puntero de área "Registro"

Puntero de área "Registro"

Función

Al transferir registros entre el panel de operador y el autómata, ambos interlocutores acceden alternativamente a las áreas de comunicación conjuntas del autómata.

Tipos de transferencia

Existen dos posibilidades para transferir registros entre el panel de operador y el autómata, a saber:

- Transferencia sin sincronización
- Transferencia con sincronización a través de la bandeja de datos

Los registros se transfieren siempre directamente. Por tanto, los valores de variables se leen o escriben directamente de/en la dirección configurada para la variable, sin desviarse al portapapeles.

Iniciar la transferencia de registros

Para iniciar la transferencia existen tres posibilidades:

- Inicio desde el visor de recetas
- Peticiones de control
  El controlador también puede disparar la transferencia de registros.
- Activar funciones configuradas

Si una petición de control lanza la transferencia de registros, los datos del visor de recetas también se actualizarán. Así pues, procure no utilizar el visor de recetas mientras se disparan peticiones de control para transferir registros. Si ya ha empezado a editar un registro y se lanza una petición de control para transferir registros, dicha petición se rechazará.

Tipos de datos permitidos

Cuando configure el puntero de área "Registro", puede utilizar los siguientes tipos de datos:

- Word
- UInt

Transferencia sin sincronización

En la transferencia asíncrona de registros entre el panel de operador y el autómata no se coordinan las áreas de datos utilizadas conjuntamente. Por ello, no es necesario crear un área de datos en la configuración.

La transferencia asíncrona de registros resulta útil p. ej. en los casos siguientes:

- Condicionado por el sistema, puede descartarse que los interlocutores sobrescriban los datos de forma incontrolada.
- El autómata no necesita información sobre el número de receta o el de registro.
- La transferencia de registros es iniciada por el usuario desde el panel de operador.

Leer valores

Al iniciar la transferencia de lectura se leen los valores de las direcciones de control y se transfieren al panel de operador.

- Inicio por el usuario desde la vista de recetas:
  Los valores se cargan en el panel de operador. Éstos se pueden procesar posteriormente en el panel de operador, p. ej. modificar o guardar los valores, etc.
- Inicio mediante una función u orden de control:
  Los valores se almacenan inmediatamente en el soporte de datos.

Escribir valores

Al iniciar la transferencia para escritura se escriben los valores en las direcciones de control.

- Inicio por operación desde la vista de recetas:
  Los valores actuales se escriben en el autómata.
- Inicio por función o petición del controlador:
  Los valores del soporte de datos se escriben en el autómata.

Ejecución de la transferencia al iniciarla desde la vista de recetas

Leer del autómata desde la vista de recetas

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta que ha de ser leído, así como el estado "Transferencia en curso" en la bandeja de datos y repone el número de registro a 0.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del autómata y los muestra en la vista de recetas. En las recetas con variables sincronizadas, los valores del autómata también se escriben en las variables.	
4	El panel de operador activa el estado "Transferencia finalizada".	
5	Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Escribir en el autómata desde la vista de recetas

Paso	Acción	
	Verificación: ¿Palabra de estado = 0?	
1	Sí	No
	El panel de operador introduce el número de receta y el número de registro que deben escribirse, así como como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.

Paso	Acción	
2	El panel de operador escribe los valores actuales en el autómata. En las recetas con variables sincronizadas, los valores modificados son sincronizados entre la vista de recetas y las variables, escribiéndose luego en el autómata.	
3	El panel de operador activa el estado "Transferencia concluida".	
4	Dado el caso, el programa de control puede evaluar entonces los datos transferidos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Nota

El panel de operador es el único que puede activar la palabra de estado. El autómata sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el autómata sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos está activado a "Transferencia finalizada".
- El estado en la bandeja de datos está activado a "Transferencia finalizada con fallos".

Ejecución de la transferencia en caso de petición de control

El panel de operador o el autómata pueden iniciar la transferencia de registros entre sí.

Para este tipo de transferencia se dispone de las órdenes de control n° 69 y n° 70.

N° 69: Leer registro del autómata ("SPS → DAT")

La orden de control n° 69 transfiere los registros del autómata al panel de operador. La orden de control está estructurada de la siguiente manera:

	Byte más significativo	Byte menos significativo
1ª palabra	0	69
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	No sobrescribir el registro existente: 0 Sobrescribir el registro existente: 1	

N° 70: Escribir registro en el autómata ("DAT → SPS")

La orden de control n° 70 transfiere los registros del panel de operador al autómata. La orden de control está estructurada de la siguiente manera:

	Byte más significativo	Byte menos significativo
1ª palabra	0	70
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	—	

Ejecución al leer del autómata mediante la orden de control "SPS → DAT" (n° 69)

Paso	Acción	
1	Prueba: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la orden, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador lee los valores del autómata y los almacena en el registro indicado en la orden de control.	
4	• Si en la orden se ha seleccionado "Sobrescribir", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia finalizada". • Si en la orden se ha seleccionado "No sobrescribir" y ya existe el registro, el panel de operador cancelará la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Ejecución al escribir en el autómata mediante la orden de control "DAT → SPS" (n° 70)

Paso	Acción	
1	Prueba: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la orden, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador toma los valores del registro indicado en la orden del portador de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Ejecución de la transferencia con inicio mediante una función configurada

Leer en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la función.	
4	• Si en la función "Sobrescribir" se ha seleccionado "Sí", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia finalizada". • Si en la función "Sobrescribir" se ha seleccionado "No" y el registro ya existe, el panel de operador interrumpirá la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Escribir en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de control recoge los valores del registro indicado en la función del soporte de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Posibles causas de error en la transferencia de registros

Causas de error posibles

Si la transferencia de registros termina con errores, puede deberse a las siguientes causas:

- No se ha configurado ninguna dirección de variable en el controlador
- No es posible sobrescribir registros
- El número de receta no existe
- El número de registro no existe

Nota

El panel de operador es el único que puede activar la palabra de estado. El controlador sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el controlador sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos se ha activado a "Transferencia finalizada".
- El estado en la bandeja de datos se ha activado a "Transferencia finalizada con fallos".

Reacción a una interrupción debida a un fallo

El panel de operador reacciona a una interrupción de la transferencia de registros debida a un fallo de la siguiente forma:

- Inicio por el usuario desde el visor de recetas:
  Indicaciones en la barra de estado del visor de recetas y salida de avisos de sistema
- Inicio mediante una función
  Emisión de avisos de sistema
- Inicio mediante una petición de control
  No hay respuesta en el panel de operador

Independientemente de ello, el usuario puede evaluar el estado de la transferencia consultando la palabra de estado en la bandeja de datos.

Nota

Dependencia del equipo

Las indicaciones en la barra de estado del visor de recetas no están disponibles en Basic Panels.

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos

Para configurar avisos tales como avisos de operador, alarmas y acuses se requieren varios pasos.

- Paso 1: Crear variables
- Paso 2: Configurar avisos
- Paso 3: configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- tipos de datos de la variable
- direccionamiento de la variable
- modo de contaje de las posiciones de bit

Tipos de datos

Para conexiones con drivers de comunicación SIMATIC se soportan los tipos de datos siguientes:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
Controladores SIMATIC S7.	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

Modo de contaje de las posiciones de bit

Para conexiones con drivers de comunicación SIMATIC rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte 0								Byte 1							
	Byte más significativo								Byte menos significativo							
En controladores SIMATIC S7	7							0	7							0
Configure en WinCC:	15							8	7							0

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

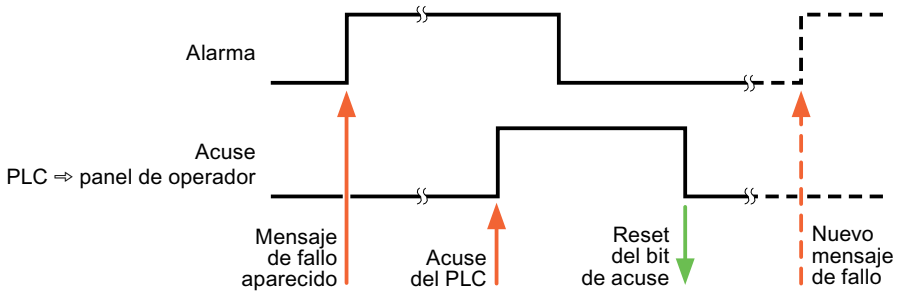
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

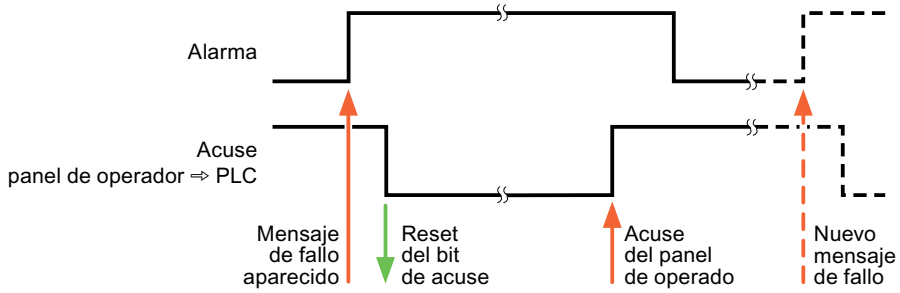
Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



10.8.7.5 Características de rendimiento de la comunicación

Dependencia del equipo SIMATIC S7 1200

Comunicación con el controlador SIMATIC S7 1200

Si utiliza con el TIA Portal V 12 SP1 dispositivos de una versión anterior del TIA Portal, puede ocurrir que no puedan realizarse conexiones integradas con determinados paneles de operador.

Basic Panels V11.0

Paneles de operador	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KP300 Basic	sí	sí	sí	sí	sí
KP400 Basic	sí	sí	sí	sí	sí
KTP400 Basic PN	sí	sí	sí	sí	sí
KTP600 Basic DP	sí	sí	sí	sí	sí
KTP600 Basic PN	sí	sí	sí	sí	sí
KTP1000 Basic DP	sí	sí	sí	sí	sí
KTP1000 Basic PN	sí	sí	sí	sí	sí
TP1500 Basic PN	sí	sí	sí	sí	sí

Basic Panels V12.0

Paneles de operador	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KP300 Basic	no	sí	sí	sí	sí
KP400 Basic	no	sí	sí	sí	sí
KTP400 Basic PN	no	sí	sí	sí	sí
KTP600 Basic DP	no	sí	sí	sí	sí
KTP600 Basic PN	no	sí	sí	sí	sí
KTP1000 Basic DP	no	sí	sí	sí	sí
KTP1000 Basic PN	no	sí	sí	sí	sí
TP1500 Basic PN	no	sí	sí	sí	sí

Basic Panels V13.0

Paneles de operador	SIMATIC S7-1200 (V1)	SIMATIC S7-1200 (V2)	SIMATIC S7-1200 (V2.2)	SIMATIC S7-1200 (V3)	SIMATIC S7-1200 (V4)
KTP700 Basic PN	no	sí	sí	sí	sí
KTP900 Basic PN	no	sí	sí	sí	sí

Tipos de datos admisibles para SIMATIC S7 1200

Tipos de datos admisibles para conexiones con SIMATIC S7 1200

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Longitud
BOOL	1 bit
SINT	1 byte
INT	2 bytes
DINT	4 bytes
USINT	1 byte
UINT	2 bytes
UDINT	4 bytes
REAL	4 bytes
LREAL	8 bytes
TIME	4 bytes
DATE	2 bytes
DTL	12 bytes
TIME_OF_DAY, TOD	4 bytes
STRING	(2+n) bytes, n = de 0 a 254
CHAR	1 byte
Array of CHAR	--
BYTE	1 byte
WORD	2 bytes
DWORD	4 bytes

10.8.7.6 Crear conexiones en el editor "Conexiones"

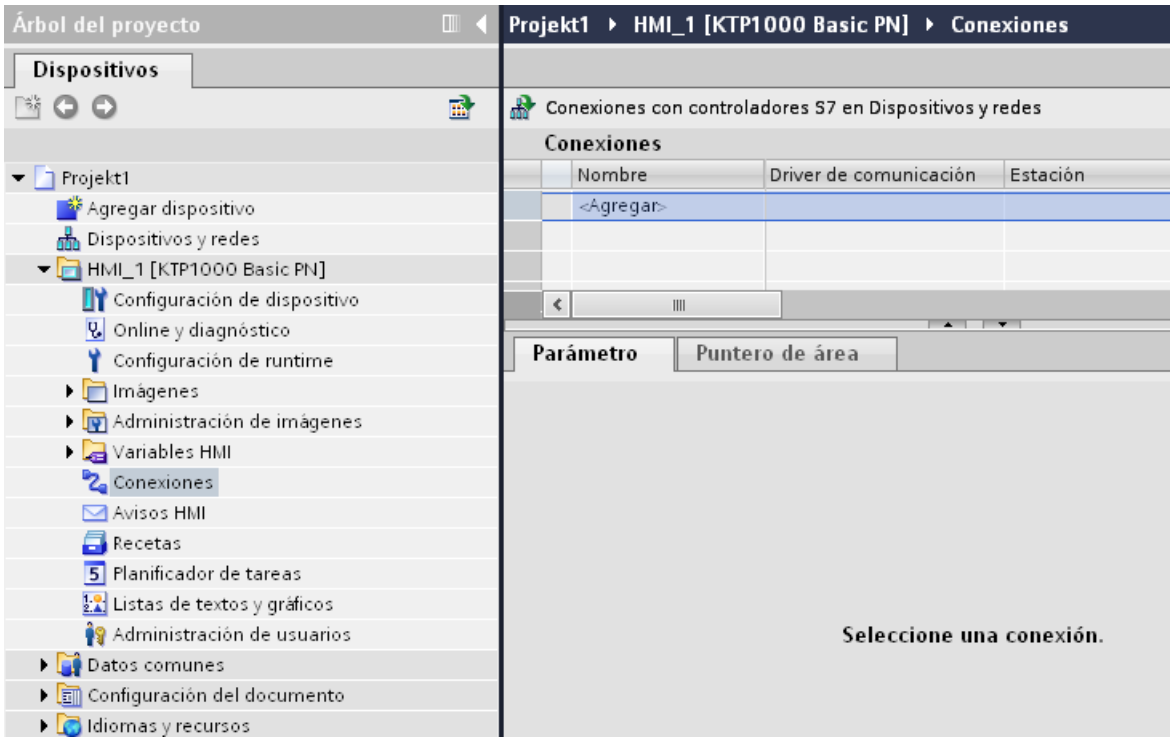
Crear una conexión PROFINET

Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador con interfaz PROFINET.

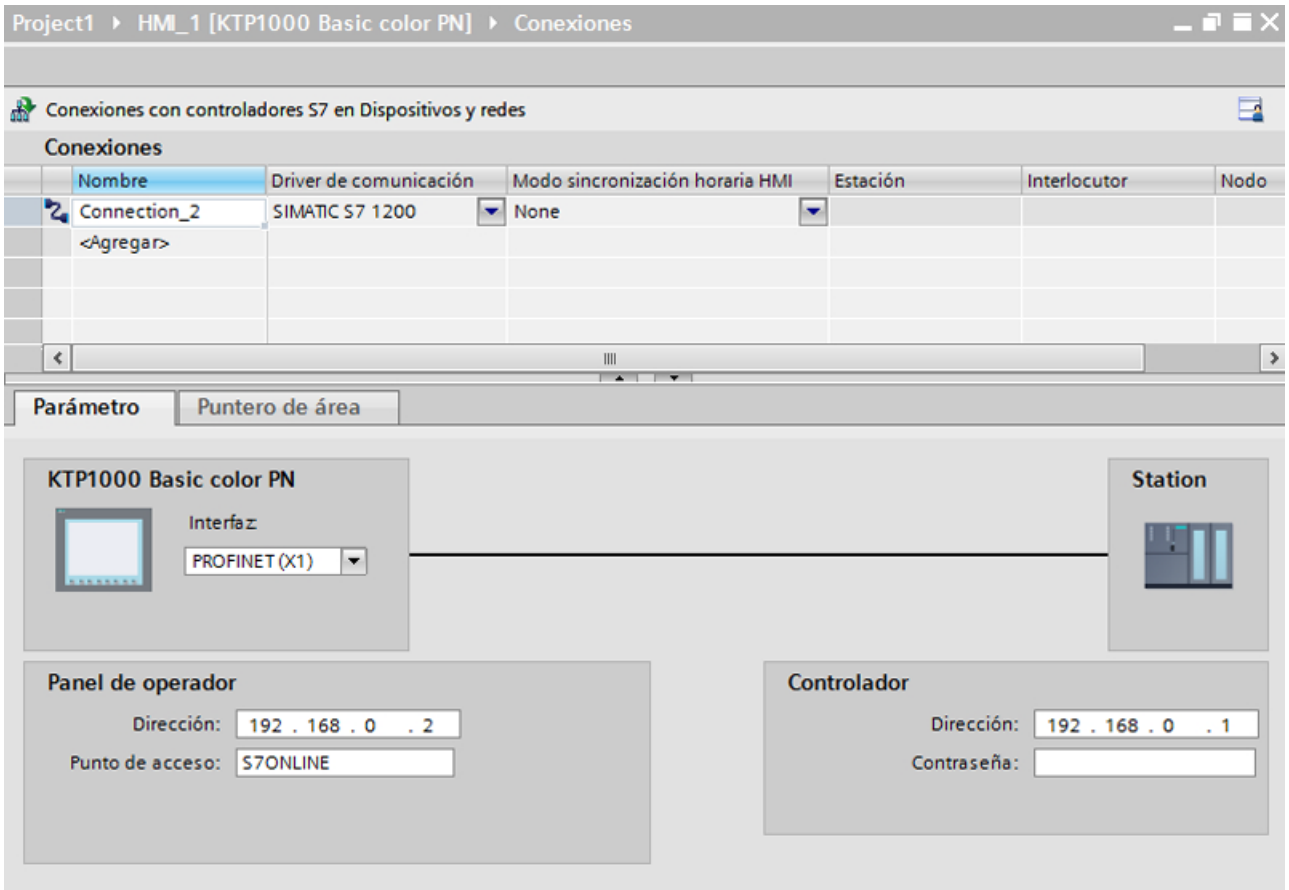
Procedimiento

1. Abra el editor "Conexiones" del panel de operador.
2. Haga doble clic en "<Agregar>".



3. Seleccione el driver "SIMATIC S7 1200" en la columna "Driver de comunicación".
4. Haga clic en el nombre de la conexión.

5. Seleccione una interfaz PROFINET del panel de operador en "Propiedades > Interfaz" de la ventana de inspección.



6. Ajuste las direcciones IP de los interlocutores en la ventana de inspección:
   - Panel de operador: "Parámetros > Panel de operador > Dirección"
   - Controlador: "Parámetros > Controlador > Dirección"

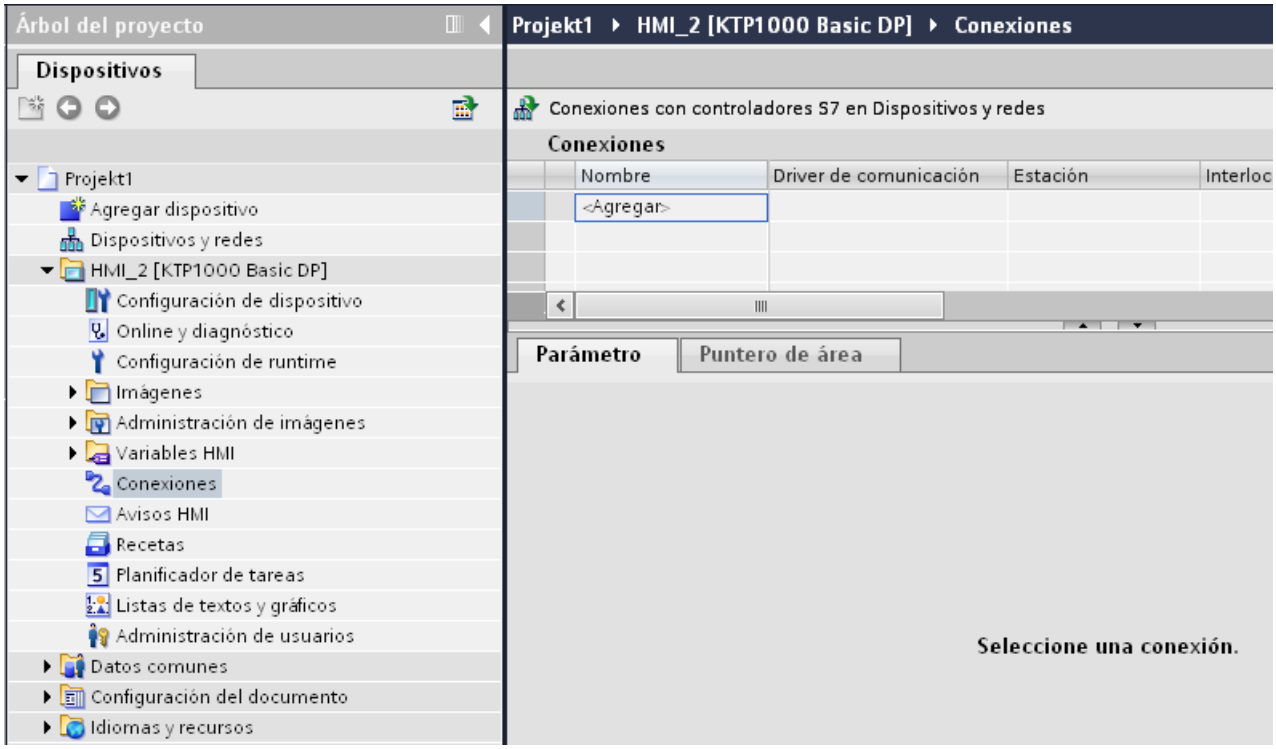
Crear una conexión PROFIBUS DP

Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador con interfaz PROFIBUS.

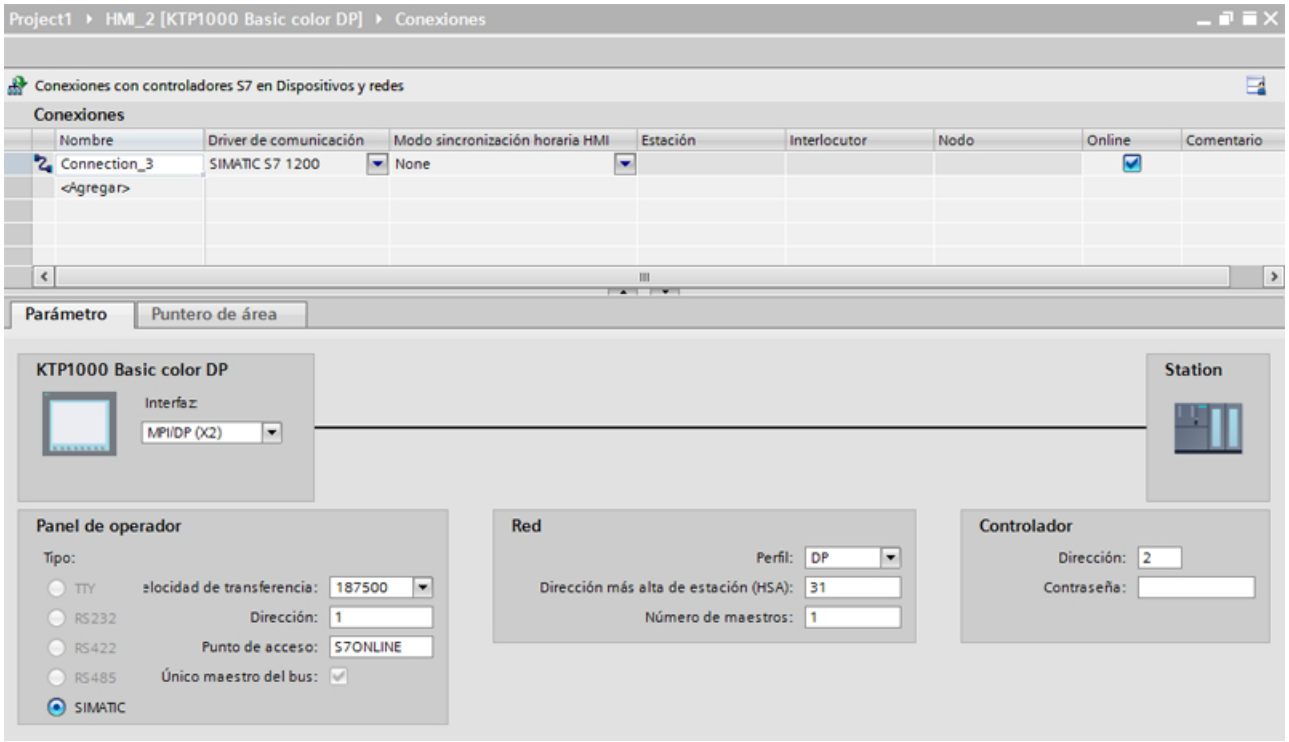
Procedimiento

1. Abra el editor "Conexiones" del panel de operador.
2. Haga doble clic en "<Agregar>".



3. Seleccione el driver "SIMATIC S7 1200" en la columna "Driver de comunicación".
4. Haga clic en el nombre de la conexión.
5. Seleccione la interfaz "MPI/DP" en "Parámetros" de la ventana de inspección.

6. Seleccione el perfil "DP" en "Parámetros > Red" de la ventana de inspección.



7. Ajuste las direcciones de los interlocutores en la ventana de inspección:
   - Panel de operador: "Parámetros > Panel de operador > Dirección"
   - Controlador: "Parámetros > Controlador > Dirección"

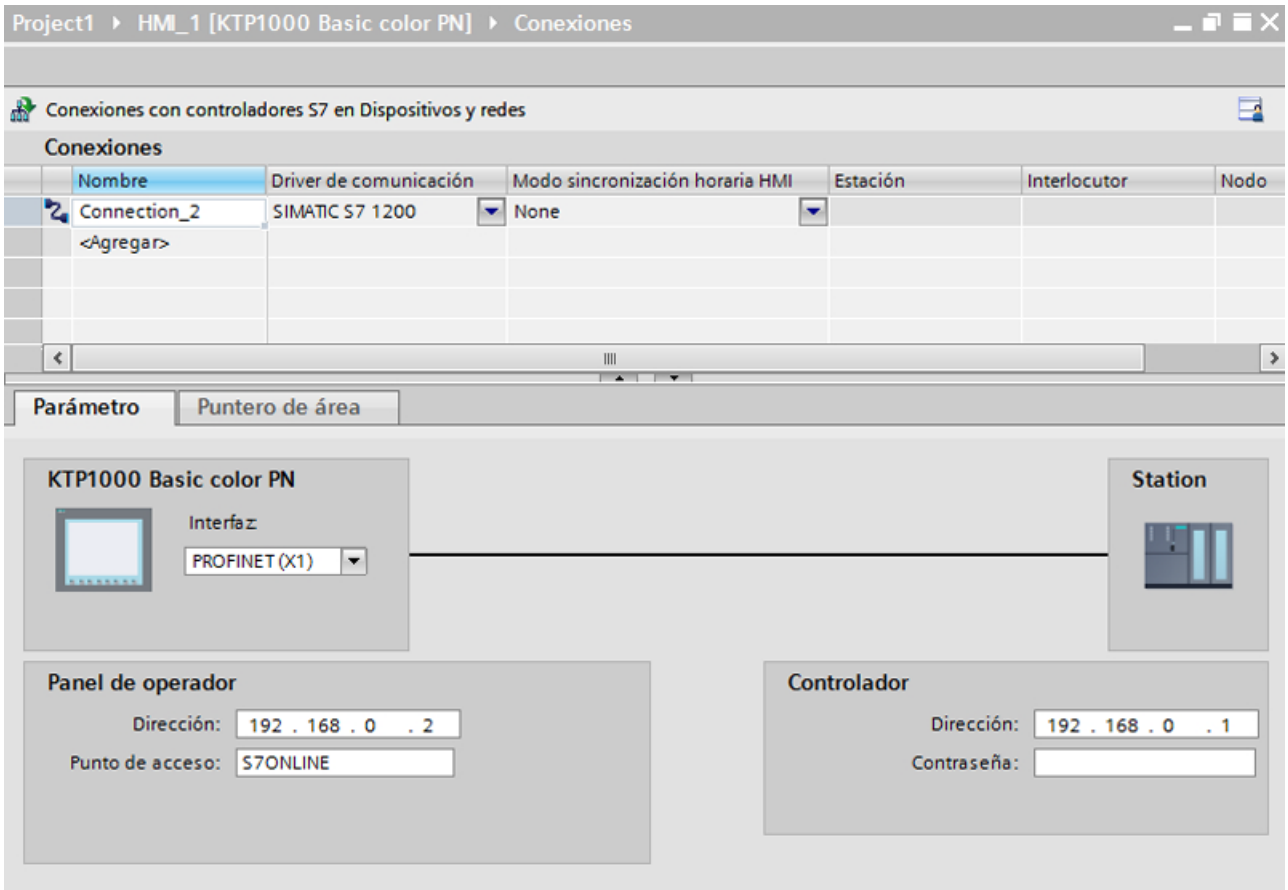
Parámetros de la conexión

Parámetros de la conexión (SIMATIC S7 1200)

Parámetros que deben configurarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador", "Red" y "Controlador", según sea la interfaz utilizada.


Project1 ▸ HMI_1 [KTP1000 Basic color PN] ▸ Conexiones
Conexiones con controladores S7 en Dispositivos y redes
Conexiones
Nombre
Driver de comunicación
Modo sincronización horaria HMI
Estación
Interlocutor
Nodo
Connection_2
SIMATIC S7 1200
None
<Agregar>
Parámetro
Puntero de área
KTP1000 Basic color PN
Interfaz:
PROFINET (X1)
Station
Panel de operador
Dirección: 192 . 168 . 0 . 2
Punto de acceso: S7ONLINE
Controlador
Dirección: 192 . 168 . 0 . 1
Contraseña:

Parámetros Ethernet

Parámetros del panel de operador

Ajuste en "Panel de operador" los parámetros para conectar el panel de operador a la red. Los cambios no se transfieren automáticamente al panel de operador. La configuración se debe modificar en el panel de control del panel de operador.

- "Interfaz"
  Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC.

  Nota

  Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

  Si activa "Obtener dirección IP por otra vía", en la siguiente carga se conserva en el Control Panel la dirección IP establecida previamente.

  Al transferir el proyecto se transfiere la dirección IP al panel de operador.
  La dirección IP del panel de operador se configura del siguiente modo:

  – Haga clic en el panel de operador.
  – Abra el editor "Configuración de dispositivos".
  – Haga clic en la interfaz Ethernet.
  – Asigne la dirección IP en la ventana de inspección en:
    "General > Interfaz PROFINET > Direcciones Ethernet"

- "Dirección"
  En el área "Dirección" se asigna la dirección IP del panel de operador.
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.

- "Punto de acceso"
  En "Punto de acceso" se define el punto de acceso para la interfaz PG/PC a través del cual se puede acceder al interlocutor.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección", determine la dirección IP del módulo S7 conectado al panel de operador.
- "Contraseña de acceso"
  En "Contraseña de acceso" se introduce una contraseña. La contraseña debe coincidir con la contraseña que haya introducido en el controlador.

Nota

La contraseña solo se necesita si ha guardado el nivel de protección "Protección completa" en el controlador.

Si ha guardado el nivel de protección "Protección completa" en el controlador y no introduce ninguna contraseña, no se establecerá ninguna conexión con el controlador.

Parámetros PROFIBUS

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red PROFIBUS.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

Nota

Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.

- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro).
  En el S7-200 hay que configurar un panel de operador como maestro.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros de la red

En "Red" se ajustan los parámetros de la red PROFIBUS en la que está integrado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "DP", "Universal" o "Estándar" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección PROFIBUS real más alta. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Número de maestros"
  En "Número de maestros" se ajusta el número de maestros en la red PROFIBUS. Este dato es necesario para calcular correctamente los parámetros del bus.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se determina la dirección PROFIBUS del módulo S7 (CPU, FM o CP) conectado al panel de operador.
- "Contraseña de acceso"
  En "Contraseña de acceso" se introduce una contraseña. La contraseña debe coincidir con la contraseña que haya introducido en el controlador.

Nota

La contraseña solo se necesita si ha guardado el nivel de protección "Protección completa" en el controlador.

Si ha guardado el nivel de protección "Protección completa" en el controlador y no introduce ninguna contraseña, no se establecerá ninguna conexión con el controlador.

10.8.8 Comunicación con SIMATIC S7 300/400

10.8.8.1 Comunicación con SIMATIC S7 300/400

Introducción

Este apartado describe la comunicación entre un panel de operador y los controladores S7 300 y S7 400. Ambos controladores se agrupan en el nombre SIMATIC S7 300/400.

Se puede configurar los siguientes canales de comunicación para el controlador SIMATIC S7 300/400:

- PROFINET
- PROFIBUS
- MPI

Conexión HMI para la comunicación

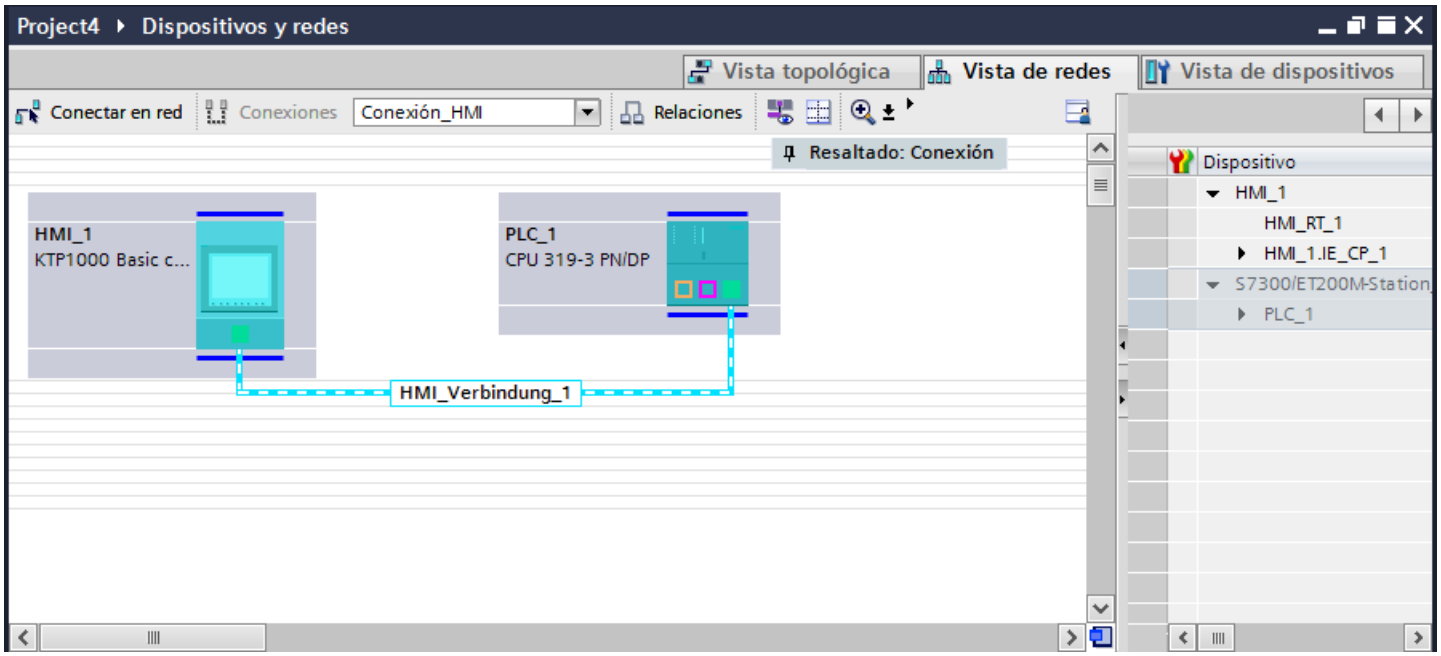
Las conexiones HMI entre un panel de operador y un SIMATIC S7 300/400 se configuran en el editor "Dispositivos y redes".

10.8.8.2 Comunicación vía PROFINET

Comunicación vía PROFINET

Conexiones HMI mediante PROFINET

Si se han insertado en el proyecto un panel de operador y un SIMATIC S7 300/400, interconecte las dos interfaces PROFINET en el editor "Dispositivos y redes".



También es posible conectar varios paneles de operador a un SIMATIC S7 300/400 y varios SIMATIC S7 300/400 a un panel de operador. El número máximo de interlocutores que pueden conectarse a un panel de operador depende del panel utilizado.

Encontrará más información al respecto en la documentación del panel de operador correspondiente.

Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFINET entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".

Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión vía PROFINET entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Configurar una conexión HMI mediante PROFINET

Introducción

Una conexión HMI vía PROFINET o Ethernet entre paneles de operador y un SIMATIC S7 300/400 se configura en el editor "Dispositivos y redes".

⚠ PRECAUCIÓN
Comunicación vía Ethernet En una comunicación basada en Ethernet, el usuario final es responsable de la seguridad de su red de datos. Los ataques selectivos pueden provocar la sobrecarga del dispositivo y perjudicar su funcionalidad.

Requisitos

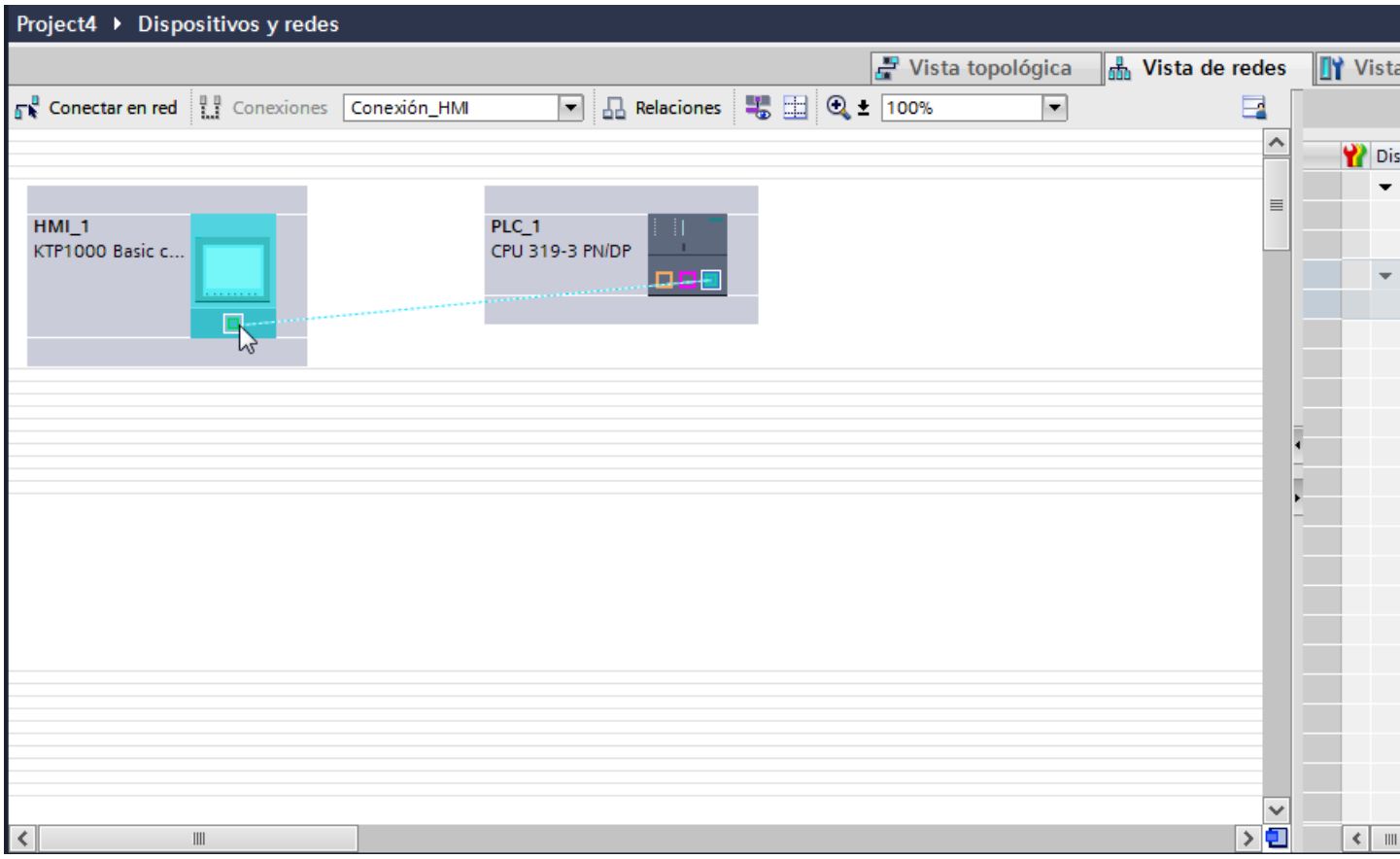
Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

- panel de operador con interfaz PROFINET o Ethernet
- SIMATIC S7 300/400 con interfaz PROFINET

Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.
2. Haga clic en el botón "Conexiones" y elija el tipo "Conexión HMI".
   Los dispositivos disponibles para conectar se representan marcados en color.

3. Haga clic en la interfaz PROFINET del controlador y arrastre una conexión a la interfaz PROFINET o Ethernet del panel de operador utilizando la función Drag&Drop.



4. Haga clic en la línea de conexión.
5. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.
   En la ventana de inspección se representa la conexión gráficamente.
6. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros PROFINET en la ventana de inspección de acuerdo con los requisitos del proyecto.
   Encontrará más información al respecto en el capítulo "Parámetros PROFINET (Página 8902)".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión.

El nombre local de la conexión sólo se puede modificar en la tabla.

Resultado

Se ha creado una conexión entre un panel de operador y un SIMATIC S7 300/400. Los parámetros de conexión "Dirección IP" y "Máscara de subred" están configurados.

Parámetros PROFINET

Parámetros PROFINET de la conexión HMI

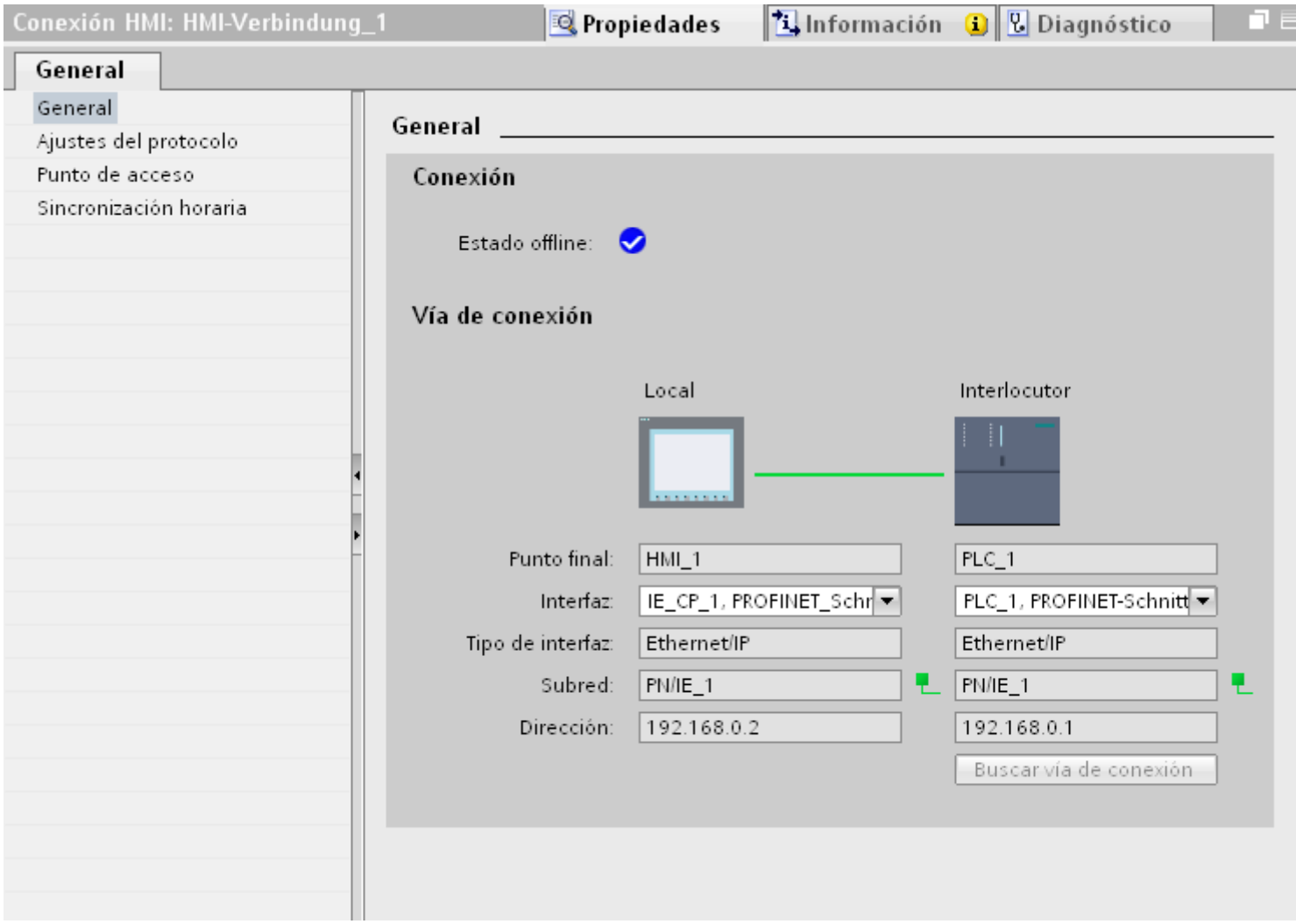
Parámetros PROFINET para la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



"Conexión"

Indica si los dispositivos ya están interconectados en red.

- Aparece cuando los dispositivos están interconectados en red.

- Aparece cuando los dispositivos no están interconectados en red.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros PROFINET. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.
- "Tipo de interfaz"
  Muestra el tipo de interfaz seleccionado. Esta área no es editable.
- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección IP seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

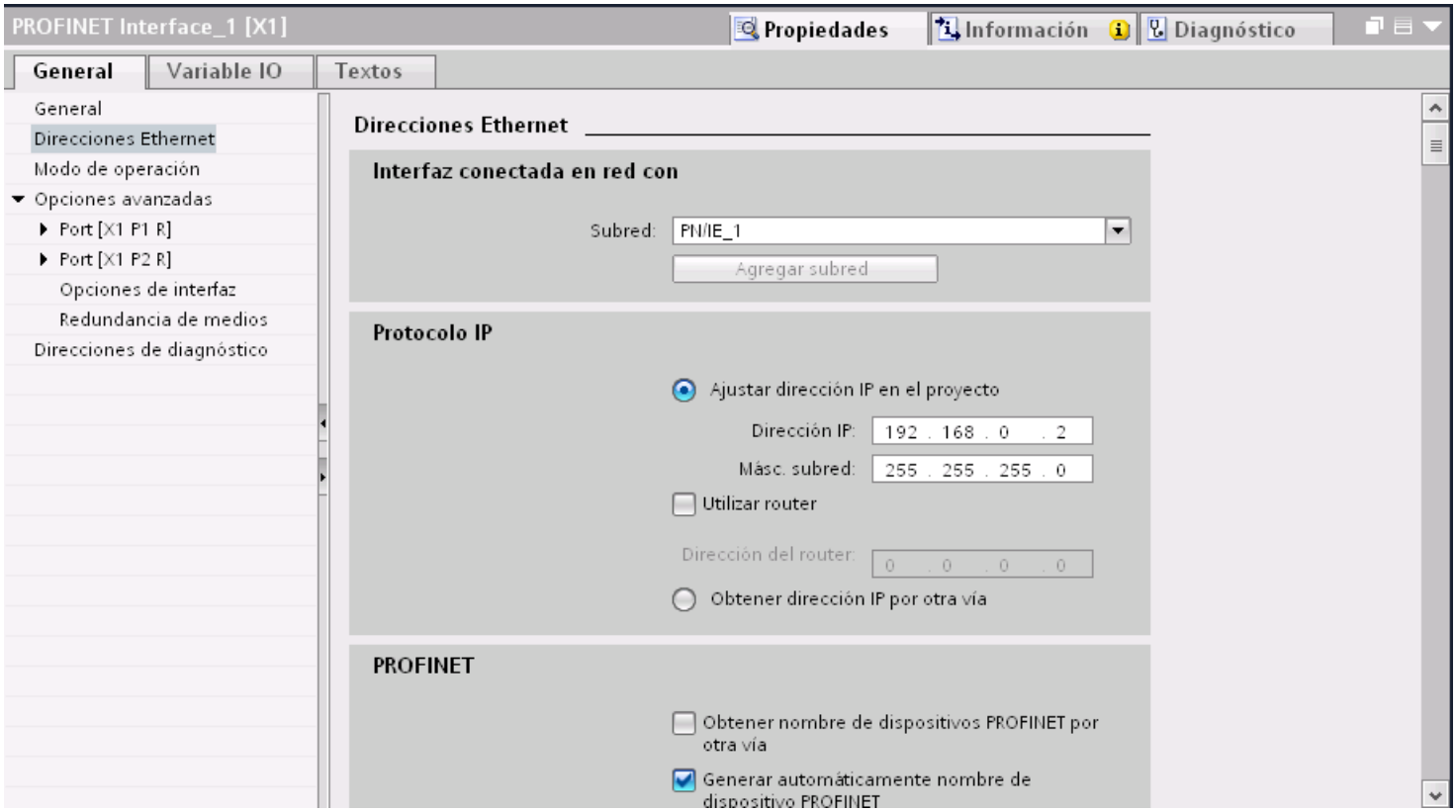
Parámetros PROFINET del panel de operador

Parámetros PROFINET del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros PROFINET del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Protocolo IP"

- "Ajustar la dirección IP en el proyecto"
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.

  Nota

  En los paneles de operador con el sistema operativo Windows CE 3.0 se realiza un reinicio automático.

  Paneles de operador con Windows CE 3.0:
  - OP 77B
  - TP 177B color PN/DP
  - TP 177B mono DP
  - OP 177B color PN/DP
  - OP 177B mono DP
  - Mobile Panel 177 PN
  - Mobile Panel 177 DP
  - TP 277 6"
  - OP 277 6"

- "Máscara de subred"
  En el área "Máscara de subred" se asignan los datos de la máscara de subred.
- "Utilizar router IP"
  Si utiliza un router IP, active "Utilizar router IP" e introduzca la dirección del router en el campo "Dirección del router".
- "Obtener la dirección IP de otro modo"
  Si la función "Obtener la dirección IP de otro modo" está activada, la dirección IP no se toma del proyecto. Introduzca la dirección IP directamente en el Control Panel del panel de operador.

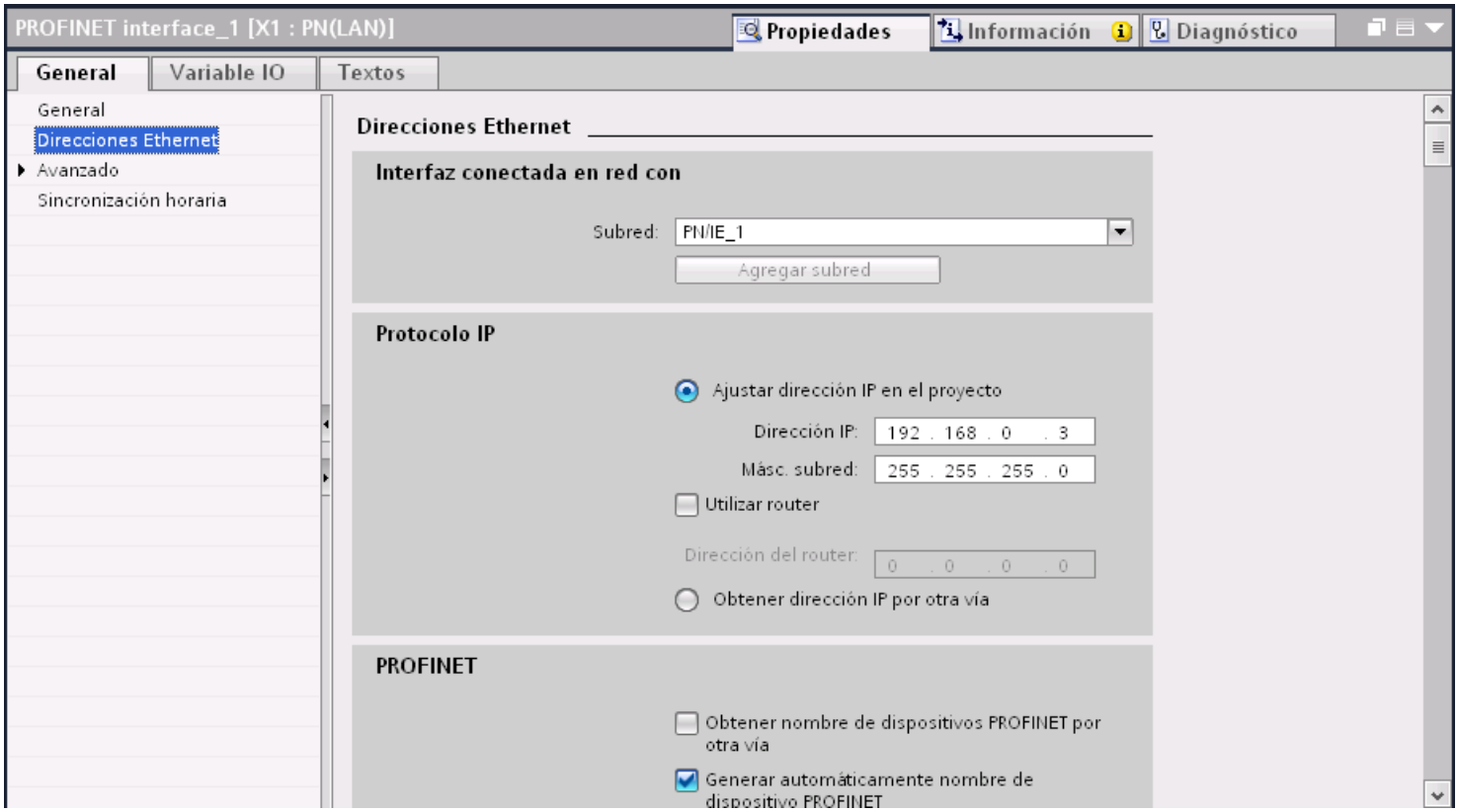
Parámetros PROFINET del controlador

Parámetros PROFINET del controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFINET del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el controlador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Protocolo IP"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección IP"
  En el área "Dirección IP" se asigna la dirección IP del panel de operador.
- "Máscara de subred"
  En el área "Máscara de subred" se asignan los datos de la máscara de subred.
  Si se utiliza un router IP, active "Utilizar router IP" e introduzca la dirección del router en el campo inferior.

Configuración de red de Industrial Ethernet

Reglas de la configuración de red

Las interfaces Ethernet de los dispositivos tienen una dirección IP predeterminada que puede modificarse.

Dirección IP

Los parámetros IP son visibles si el dispositivo apto para la comunicación soporta el protocolo TCP/IP.

La dirección IP consiste en 4 números decimales comprendidos en un rango entre 0 y 255. Los decimales están separados entre sí por un punto.

Ejemplo: 140.80.0.2

La dirección IP se compone de los datos siguientes:

- la dirección de la (sub)red
- la dirección del nodo (también se denomina host o nodo de red)

Máscara de subred

La máscara de subred separa las dos direcciones. Determina qué parte de la dirección IP direcciona la red y qué parte direcciona el nodo.

Los bits activados de la máscara de subred determinan la parte de la dirección IP correspondiente a la red.

Ejemplo:

Máscara de subred: 255.255.0.0 = 11111111.11111111.00000000.00000000

En el ejemplo superior de dirección IP, la máscara de subred tiene el significado siguiente:

Los 2 primeros bytes de la dirección IP determinan la subred, es decir, 140.80. Los 2 últimos bytes direccionan el nodo, es decir, 0.2.

Por lo general rige lo siguiente:

- La dirección de la red resulta de la combinación lógica Y de la dirección IP y la máscara de subred.
- La dirección del nodo resulta de la combinación lógica Y-NO de la dirección IP y la máscara de subred.

Relación entre la dirección IP y la máscara de subred predeterminada

Existe un acuerdo respecto de la asignación entre áreas de direcciones IP y "máscaras de subred predeterminadas". El primer decimal de la dirección IP (desde la izquierda) determina

la estructura de la máscara de subred predeterminada en cuanto al número de valores "1" (binarios), del siguiente modo:

Dirección IP (dec.)	Dirección IP (bin.)	Clase de dirección	Máscara de subred predeterminada
de 0 a 126	0xxxxxxx.xxxxxxxx....	A	255.0.0.0
de 128 a 191	10xxxxxx.xxxxxxxx...	B	255.255.0.0
de 192 a 223	110xxxxx.xxxxxxxx...	C	255.255.255.0

Nota

Rango de valores del primer decimal

En el primer decimal de la dirección IP también se acepta un valor entre 224 y 255 (clase de dirección D, etc.). Sin embargo, no es recomendable porque no se verifica la dirección de estos valores.

Enmascarar otras subredes

La máscara de subred permite seguir esctructurando una subred asignada a las clases de dirección A, B o C y crear subredes "privadas" poniendo a "1" otras posiciones de orden inferior de la máscara de subred. Por cada bit puesto a "1" se duplica el número de redes "privadas" y se divide en dos el número de nodos que contiene. Hacia fuera, la red sigue pareciendo una red individual.

Ejemplo:

En una subred de la clase de dirección B (p. ej. dirección IP 129.80.xxx.xxx), la máscara de subred predeterminada se modifica del siguiente modo:

Máscaras	Decimal	Binario
Máscara de subred predeterminada	255.255.0.0	11111111.11111111.00000000.00000000
Máscara de subred	255.255.128.0	11111111.11111111.10000000.00000000

Resultado:

Todos los nodos con direcciones comprendidas entre 129.80.001.xxx y 129.80.127.xxx están en una subred y todos los nodos con direcciones comprendidas entre 129.80.128.xxx y 129.80.255.xxx están en otra subred.

Transición de red (router)

Las transiciones de red (router) sirven para conectar las subredes. Si un datagrama IP debe enviarse a otra red, primero hay que transmitirlo a un router. Para que esto sea posible, en este caso hay que introducir la dirección del router en cada nodo de la subred.

La dirección IP de un nodo de la subred y la dirección de la transición de red (router) sólo pueden ser diferentes en las cifras que son "0" en la máscara de subred.

Definir opciones de puerto

Definición de opciones de puerto

Modificar los ajustes de conexión para el puerto PROFINET IO

En caso necesario, los ajustes de red para el puerto PROFINET IO se pueden modificar. La opción predeterminada es la definición automática de ajustes que, por norma general, permite comunicarse correctamente.

Posibilidades de ajuste de velocidad de transferencia/dúplex

Se pueden realizar los siguientes ajustes para "Velocidad de transferencia/dúplex" en función del dispositivo seleccionado:

- Ajuste automático
  Ajuste predeterminado recomendando para el puerto. La configuración de la transferencia con el puerto del interlocutor se "negocia" automáticamente. Con este ajuste también se activa automáticamente la opción "Activar autonegotiation", es decir, se puede utilizar cable cruzado o no cruzado para la conexión.
- TP/ITP con x Mbits/s. dúplex (semidúplex)
  Ajuste de la velocidad de transferencia y del modo dúplex/semidúplex. El efecto depende de la opción ajustada "Activar autonegotiation":
  - Autonegotiation activada
    Es posible utilizar tanto cable cruzado como no cruzado.
  - Autonegotiation desactivada
    Asegúrese de utilizar el cable correcto (cable cruzado o no cruzado). El puerto también se vigila con este ajuste.
- Desactivado
  En función del tipo de módulo, en la lista desplegable aparece la opción "desactivado". Así, por ejemplo, tiene la posibilidad de prohibir el acceso a un puerto no utilizado por motivos de seguridad. Con este ajuste no se generan eventos de diagnóstico.

Opción "Vigilar"

Con esta opción se activa y desactiva el diagnóstico de puerto. Ejemplos de diagnóstico de puerto: se vigila el estado del enlace, es decir, se genera el diagnóstico en caso de fallo del enlace y se vigila la reserva del sistema en los puertos de fibra óptica.

Opción "Activar autonegotiation"

El ajuste de autonegotiation sólo es modificable si se ha seleccionado un medio específico (p. ej. TP 100 Mbits/s dúplex). De las propiedades del módulo depende que un medio específico se pueda ajustar o no.

Si se ha desactivado la autonegotiation, se fuerza el ajuste fijo del puerto, de manera similar a como se requiere p. ej. en caso de arranque priorizado del dispositivo IO.

Hay que procurar realizar los mismos ajustes en el puerto del interlocutor, porque en esta opción los parámetros operativos de la red conectada no se reconocen y, en consecuencia, la velocidad de transferencia de datos y el modo de transferencia no se pueden ajustar óptimamente.

Nota

Cuando hay un puerto local interconectado, STEP 7 realiza el ajuste del puerto del interlocutor siempre y cuando soporte este ajuste. Si el puerto del interlocutor no soporta este ajuste, se genera un mensaje de error.

Normas de cableado con autonegotiation desactivada

Requisitos

Se han realizado los ajustes siguientes p. ej. para acelerar el tiempo de arranque del dispositivo IO para el puerto correspondiente:

- Velocidad de transferencia fija
- Autonegotiation incl. autocrossing desactivada

De ese modo, durante el arranque se prescinde del tiempo de negociación de la velocidad de transferencia.

Si ha desactivado Autonegotiation, debe observar las normas de cableado.

Normas de cableado con Autonegotiation desactivada

Los dispositivos PROFINET tienen los dos tipos de puertos siguientes:

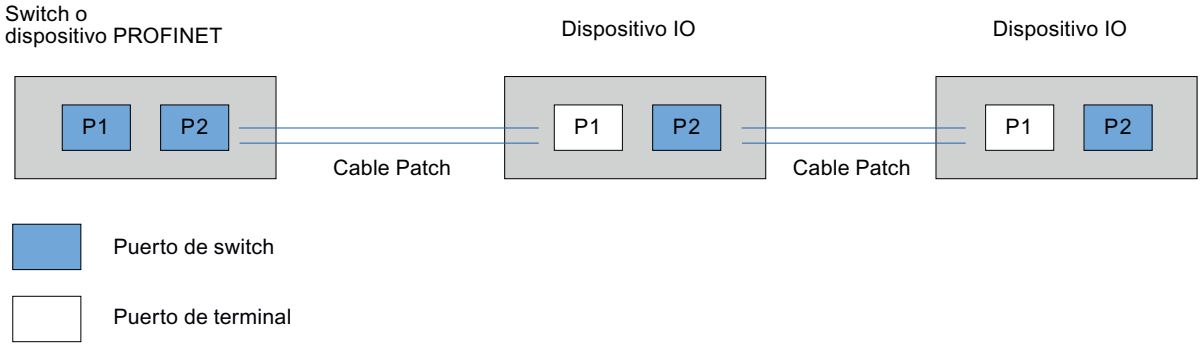
Tipo de puerto	Dispositivos PROFINET	Observación
Switchport con asignación de pines cruzada	En dispositivos IO: puerto 2 En CPUs S7 con 2 puertos: puerto 1 y puerto 2	Asignación de pines cruzada significa que la asignación de pines de los puertos para envío y recepción se intercambian internamente entre los dispositivos PROFINET afectados.
Puerto de terminal con asignación de pines no cruzada	En dispositivos IO: puerto 1 En CPUs S7 con un puerto: puerto 1	-

Validez de las normas de cableado

Las normas de cableado descritas en el siguiente apartado rigen exclusivamente para aquellos casos en los que se ha definido un ajuste de puerto fijo.

Normas para el cableado

Varios dispositivos IO pueden conectarse en línea con un tipo de cable (cable no cruzado). Para ello se conecta el puerto 2 del dispositivo IO (periferia descentralizada) con el puerto 1 del siguiente dispositivo IO. En el gráfico siguiente se representa un ejemplo con dos dispositivos IO.



Limitaciones en el puerto

Requisitos

Para poder trabajar con "boundaries" (limitaciones), el dispositivo correspondiente debe contar con más de un puerto. Si PROFINET no soporta limitaciones, éstas tampoco se visualizan.

Activar limitaciones

Se entiende por limitaciones los límites para transferir determinados frames en Ethernet. Es posible activar las siguientes limitaciones en un puerto:

- "Fin del registro de nodos accesibles"
  No se transfieren los frames DCP para el registro de nodos accesibles. Los nodos que están detrás de este puerto ya no se visualizan en "Nodos accesibles" del árbol del proyecto. La CPU ya no puede acceder a dichos nodos.
- "Fin de la detección de topología"
  No se transfieren los frames LLDP (Link Layer Discovery Protocol) para la detección de topología.
- "Fin del dominio Sync"
  No se transfieren frames Sync que se transmiten dentro de un dominio Sync para sincronizar los nodos.
  Si se opera p. ej. un dispositivo PROFINET con más de dos puertos en un anillo, es preciso impedir la inserción de frames Sync en el anillo activando una limitación Sync (en los puertos no incluidos en el anillo).
  Otro ejemplo: si se quieren utilizar varios dominios Sync, hay que configurar una limitación de dominio Sync para el puerto que está conectado con un dispositivo PROFINET del otro dominio Sync.

Restricciones

Hay que tener en cuenta las siguientes restricciones:

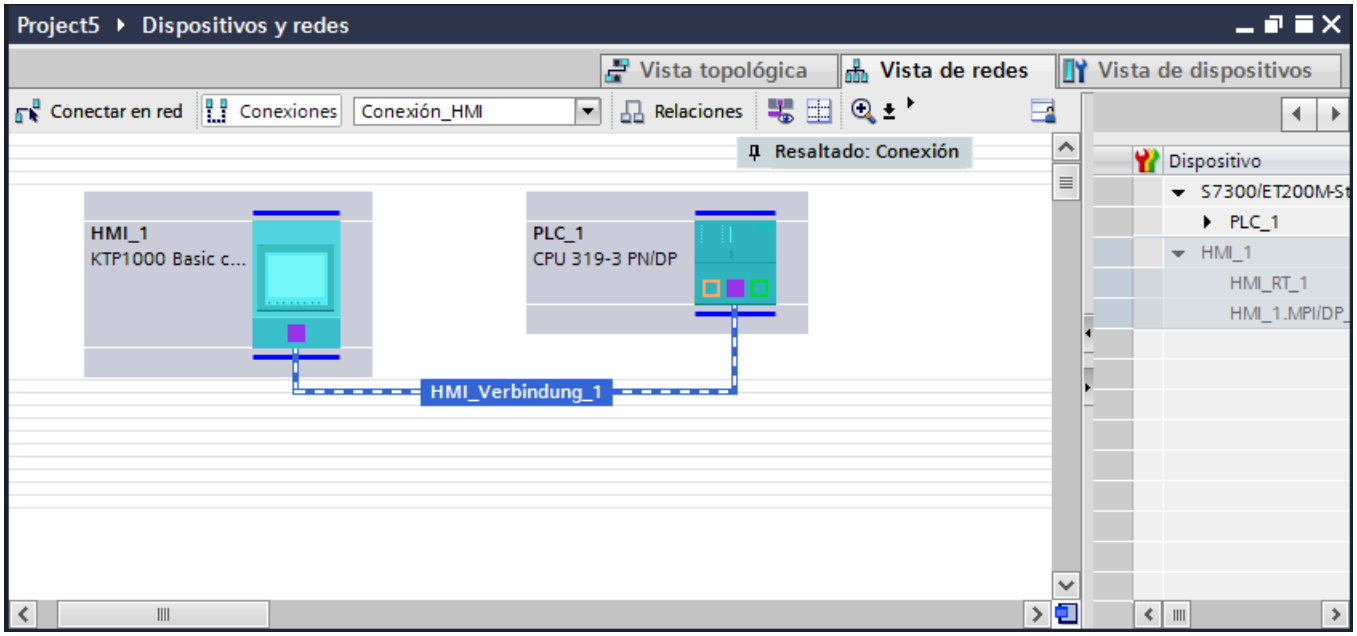
- Las casillas de verificación sólo están operativas si el puerto soporta la correspondiente característica.
- Si se ha determinado un puerto interlocutor para el puerto, las siguientes casillas de verificación no están operativas:
  - "Fin del registro de nodos accesibles"
  - "Fin de la detección de topología"
- Si la autonegotiation está desactivada, no está operativa ninguna casilla de verificación.

10.8.8.3 Comunicación vía PROFIBUS

Comunicación vía PROFIBUS

Conexiones HMI mediante PROFIBUS

Si se han insertado en el proyecto un panel de operador y un SIMATIC S7 300/400, interconecte las dos interfaces PROFIBUS en el editor "Dispositivos y redes".



Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFIBUS entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".

Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Configurar una conexión HMI mediante PROFIBUS

Introducción

Una conexión HMI vía PROFIBUS entre paneles de operador y un SIMATIC S7 300/400 se configura en el editor "Dispositivos y redes".

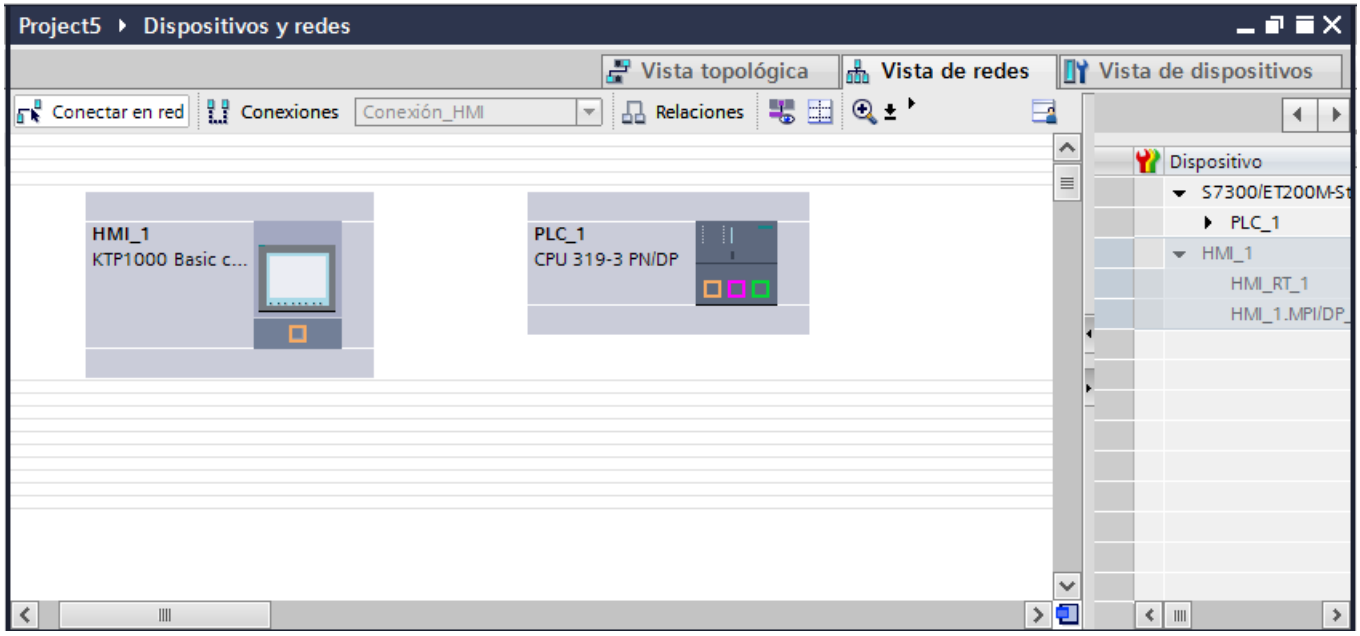
Requisitos

Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

- panel de operador con interfaz MPI/DP
- SIMATIC S7 300/400 con interfaz PROFIBUS

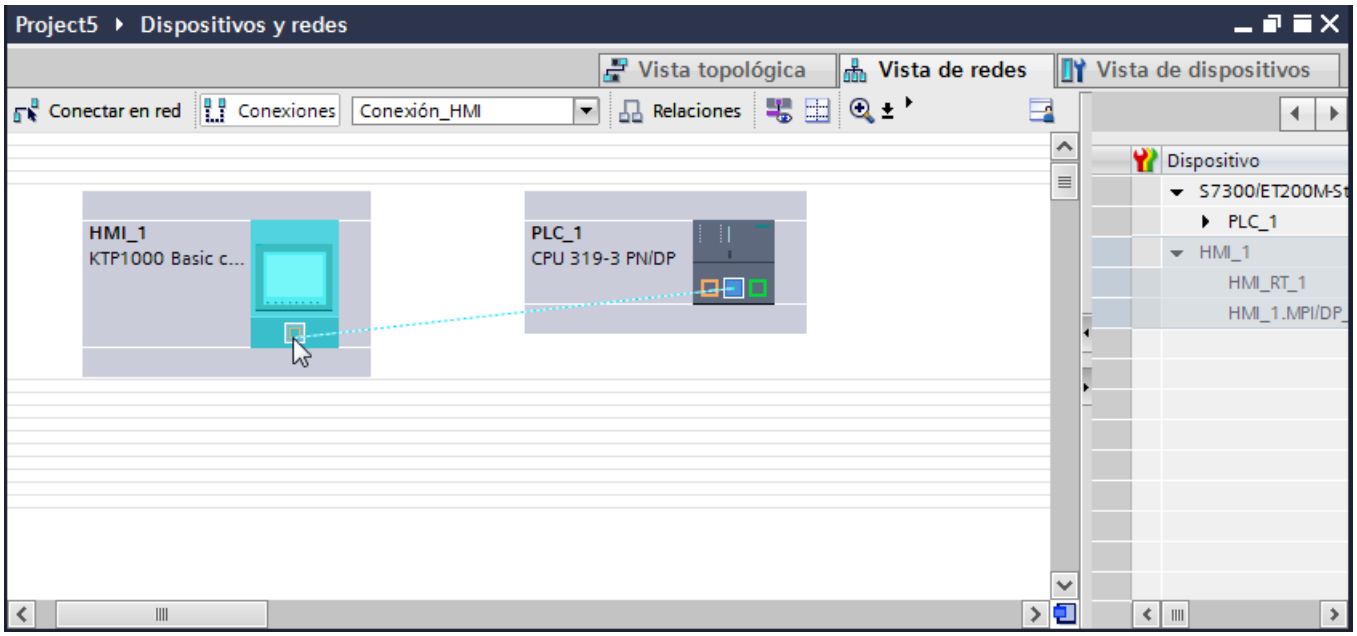
Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.



2. Haga clic en el botón "Conexiones".
   Los dispositivos disponibles para conectar se representan marcados en color.
3. Haga clic en la interfaz del panel de operador.
4. Seleccione el tipo de interfaz "PROFIBUS" en la ventana de inspección "Propiedades > General > HMI MPIDP > Parámetros".

5. Haga clic en la interfaz del controlador y arrastre una conexión hasta el panel de operador con la función Drag&Drop.



6. Haga clic en la línea de conexión.
7. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.
   En la ventana de inspección se representa la conexión gráficamente.
8. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros PROFINET en la ventana de inspección de acuerdo con los requisitos del proyecto.
   Encontrará más información al respecto en el capítulo "Parámetros PROFIBUS (Página 8915)".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión.

El nombre local de la conexión sólo se puede modificar en la tabla.

Resultado

Se ha creado una conexión HMI entre un panel de operador y un SIMATIC S7 300/400 a través de PROFIBUS.

Parámetros PROFIBUS

Parámetros PROFIBUS de la conexión HMI

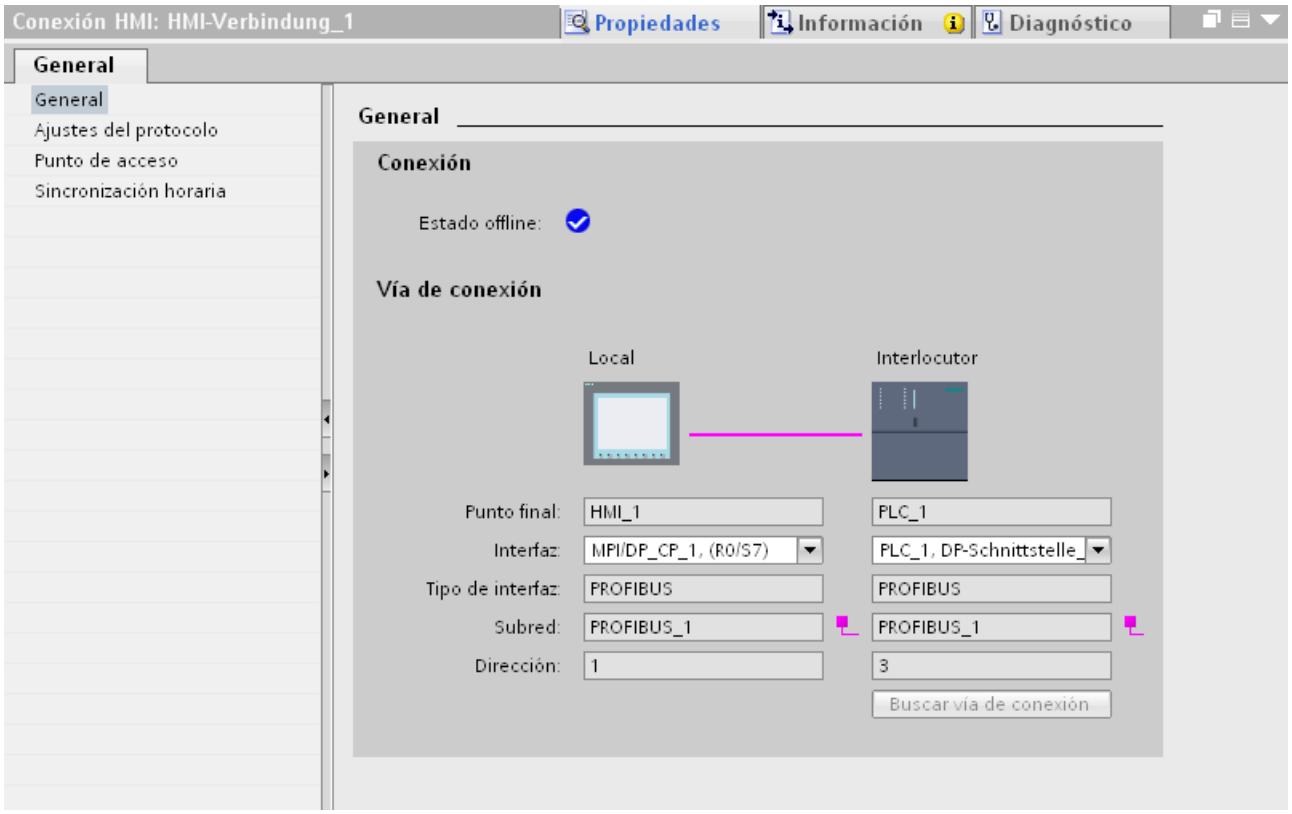
Parámetros PROFIBUS de la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



"Conexión"

Indica si los dispositivos ya están interconectados en red.

- Aparece cuando los dispositivos están interconectados en red.
- Aparece cuando los dispositivos no están interconectados en red.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros PROFIBUS. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.
- "Tipo de interfaz"
  Muestra el tipo de interfaz seleccionado. Esta área no es editable.
- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección PROFIBUS seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

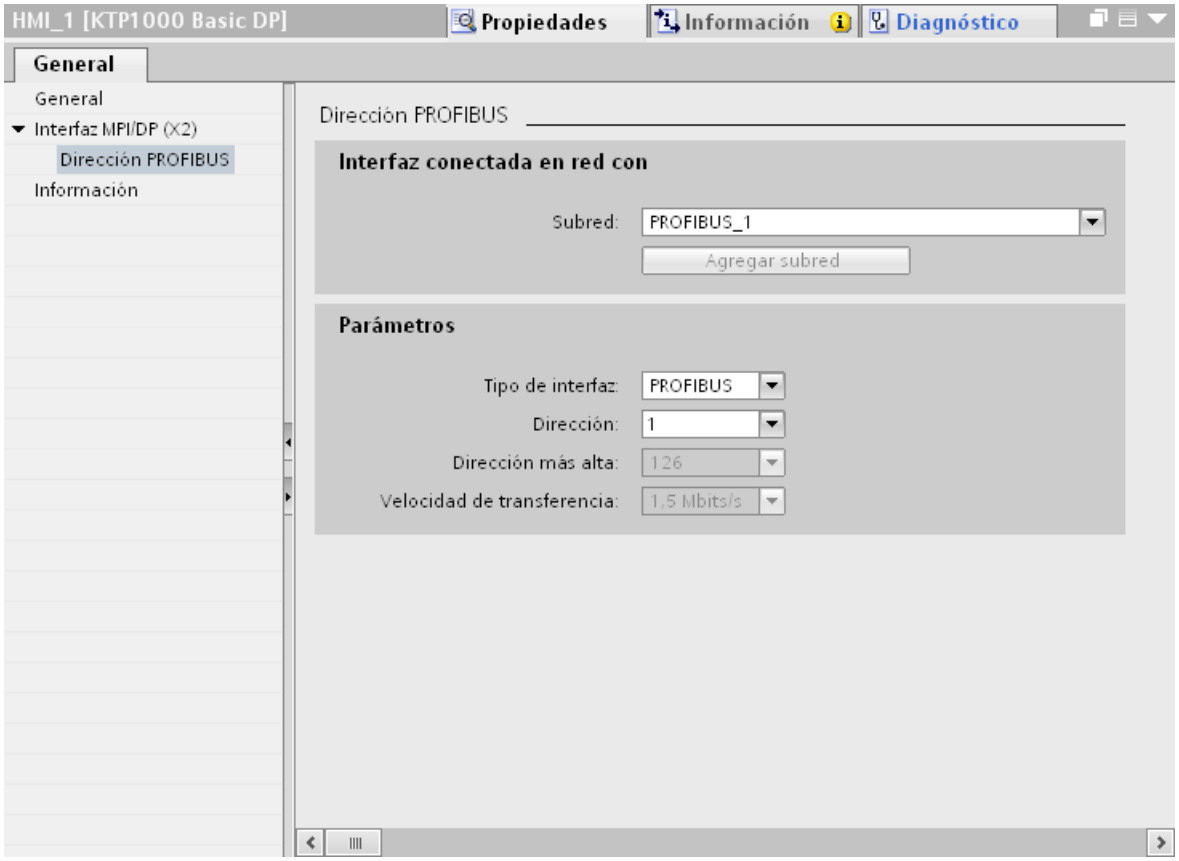
Parámetros PROFIBUS del panel de operador

Parámetros PROFIBUS del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros PROFINET del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En el área "Tipo de interfaz" se parametriza el tipo de interfaz. En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.

- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red PROFIBUS.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

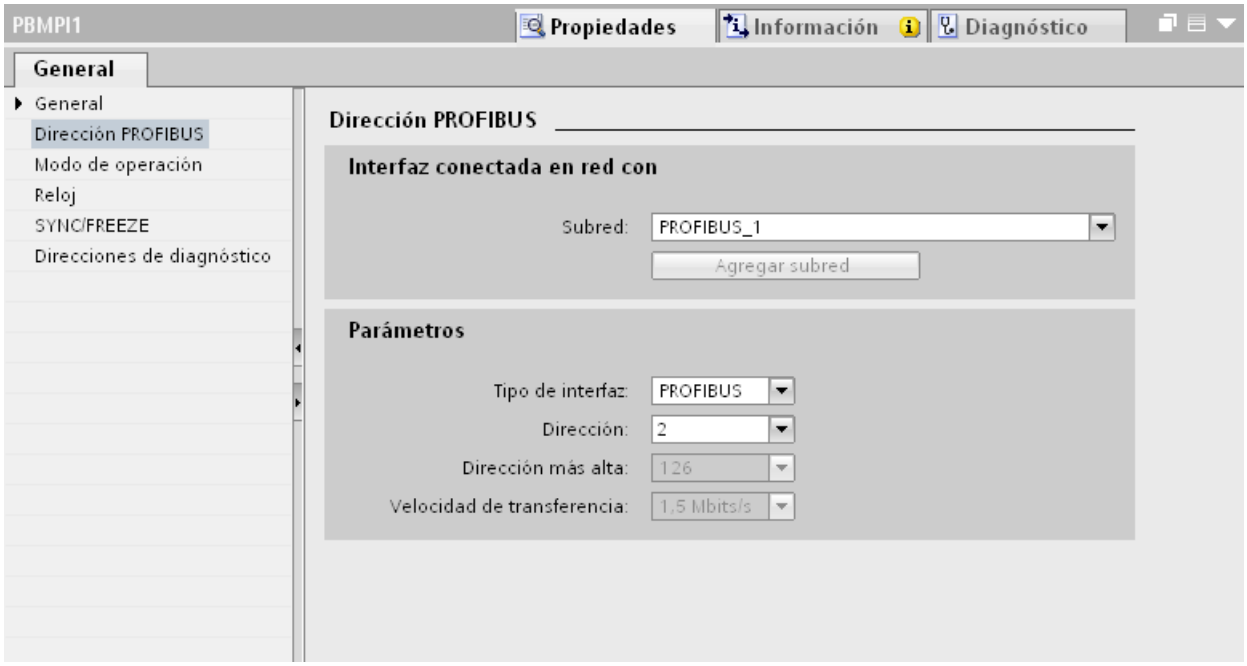
Parámetros PROFIBUS del controlador

Parámetros PROFIBUS del controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFIBUS del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el controlador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red PROFIBUS.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

Perfiles de bus en PROFIBUS

Introducción

Según sean los tipos de dispositivo conectados y los protocolos utilizados en PROFIBUS, hay disponibles diferentes perfiles. Los perfiles se diferencian por sus posibilidades de ajuste y por el cálculo de los parámetros de bus. A continuación se explican los perfiles.

Nodo con diferentes perfiles en la misma subred PROFIBUS

La subred PROFIBUS sólo funciona a la perfección si los parámetros de bus de todos los nodos tienen los mismos valores.

Perfiles y velocidades de transferencia

Perfiles	Velocidades de transferencia soportadas en Kbits/s
DP	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Estándar	9,6 19,2 45,45 93,75 187,5 500 1500 3000 6000 12000
Universal	9,6 19,2 93,75 187,5 500 1500

Significado de los perfiles

Perfil	Significado
DP	Seleccione el perfil de bus "DP" si en la subred PROFIBUS sólo hay conectados dispositivos que cumplen con exigencias de la norma EN 50170 Volumen 2/3, Parte 8-2 PROFIBUS. El ajuste de los parámetros de bus está optimizado para dichos dispositivos. Entre ellos se incluyen dispositivos con interfaces de maestro y esclavo DP del SIMATIC S7 así como dispositivos de periferia descentralizada de otros fabricantes.
Estándar	En comparación con el perfil "DP", el perfil "Estándar" ofrece la posibilidad añadida de incluir nodos de otro proyecto o nodos no configurados aquí en el cálculo de los parámetros de bus. Seguidamente, los parámetros de bus se calculan siguiendo un algoritmo sencillo y no optimizado.
Universal	Seleccione el perfil de bus "Universal" cuando algunos nodos de la subred PROFIBUS utilicen el servicio PROFIBUS FMS. Entre ellos se incluyen, p. ej., los dispositivos siguientes: • CP 343-5 • dispositivos PROFIBUS FMS de otros fabricantes Igual que en el perfil "Estándar", en este caso también existe la posibilidad de considerar nodos adicionales al calcular los parámetros de bus.

10.8.8.4 Comunicación vía MPI

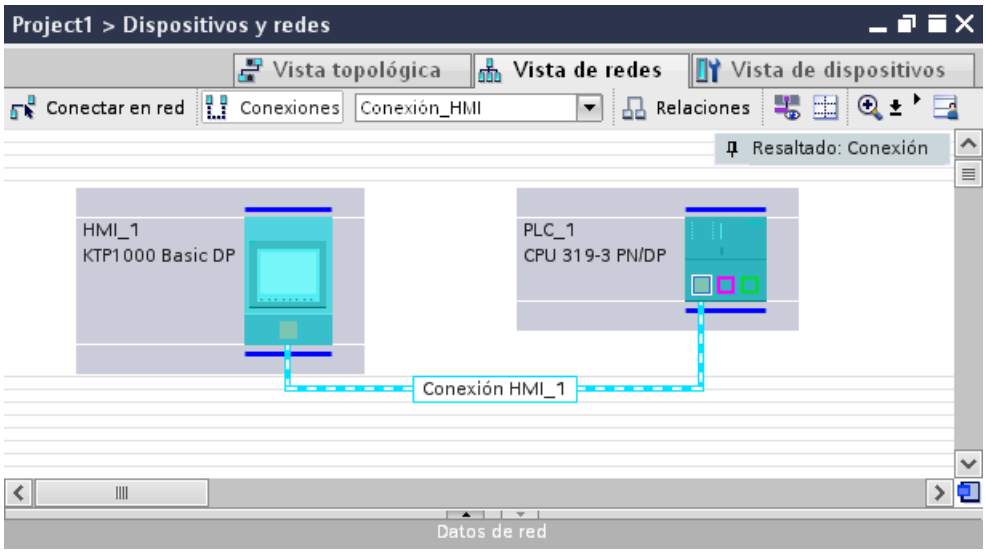
Comunicación vía MPI

Conexiones HMI mediante MPI

Si se han insertado en el proyecto un panel de operador y un SIMATIC S7 300/400, interconecte las dos interfaces MPI en el editor "Dispositivos y redes".

Conexión HMI en el editor "Dispositivos y redes"

La conexión HMI vía PROFIBUS entre el controlador y el panel de operador se configura en el editor "Dispositivos y redes".



Conexión en el editor "Conexiones"

También existe la posibilidad de configurar la conexión entre el controlador y el panel de operador en el editor "Conexiones" del panel de operador.

Configurar una conexión HMI mediante MPI

Introducción

Una conexión HMI vía MPI entre paneles de operador y un SIMATIC S7 300/400 se configura en el editor "Dispositivos y redes".

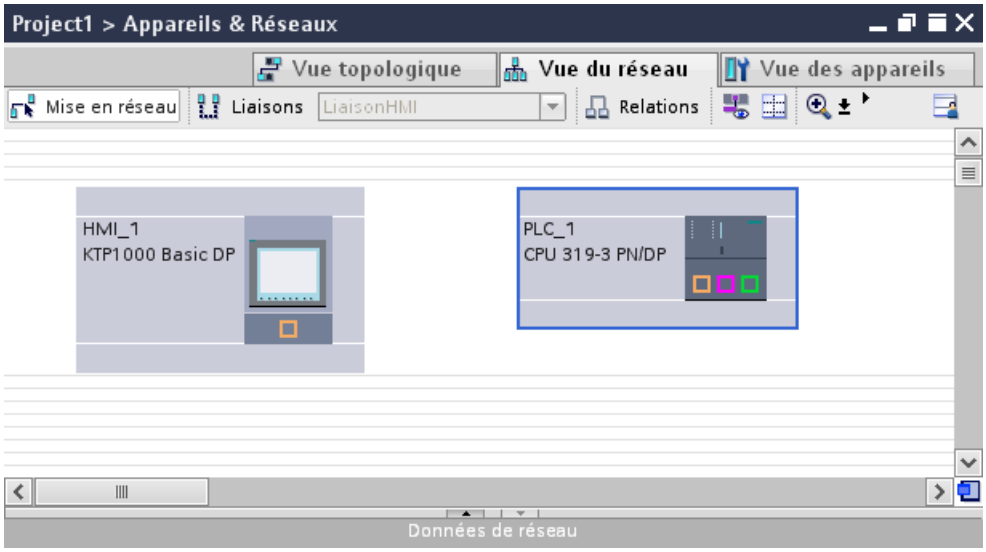
Requisitos

Los interlocutores siguientes están creados en el editor "Dispositivos y redes":

- panel de operador con interfaz MPI/DP
- SIMATIC S7 300/400 con interfaz MPI/DP

Procedimiento

1. Haga doble clic en la entrada "Dispositivos y redes" del árbol del proyecto.
   En la vista de redes se representan gráficamente los interlocutores existentes en el proyecto.



2. Haga clic en el botón "Conexiones".
   Los dispositivos disponibles para conectar se representan marcados en color.
3. Haga clic en la interfaz del controlador y arrastre una conexión hasta el panel de operador con la función Drag&Drop.

4. Haga clic en la línea de conexión.
   En la ventana de inspección se representa la conexión gráficamente.

5. Haga clic en "Resaltar conexión HMI" y seleccione la conexión HMI.
6. Haga clic en los interlocutores en la "Vista de redes" y modifique los parámetros MPI en la ventana de inspección de acuerdo con los requisitos del proyecto.
   Encontrará más información al respecto en el capítulo "Parámetros MPI (Página 8923)".

Nota

La conexión HMI creada se visualizará también en el área del editor en forma de tabla, en la ficha "Conexiones". En la tabla se controlan los parámetros de conexión, donde también se puede modificar el interlocutor. El nombre local de la conexión sólo se puede modificar en la tabla.

Resultado

Se ha creado una conexión HMI entre un panel de operador y un SIMATIC S7 300/400 a través de MPI.

Parámetros MPI

Parámetros MPI de la conexión HMI

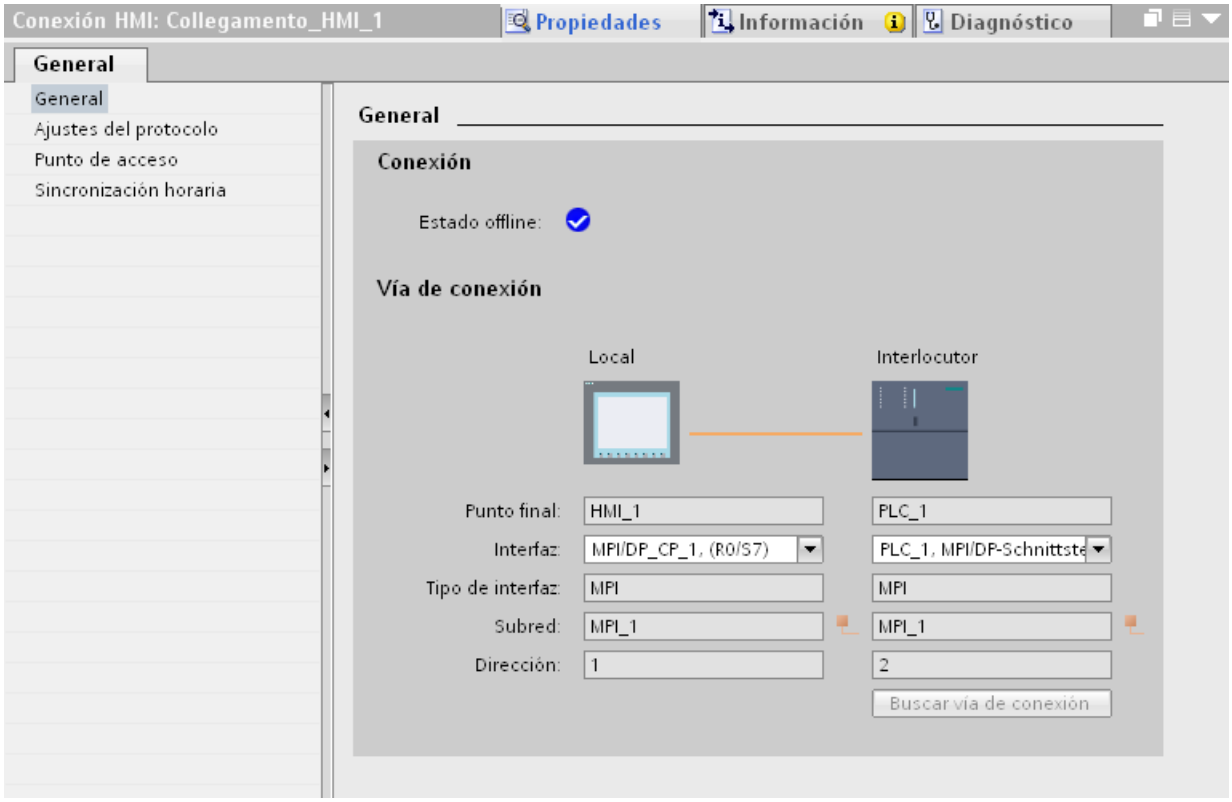
Parámetros MPI de la conexión HMI

En las propiedades de la conexión HMI encontrará una vista general de los parámetros configurados de una conexión HMI.

En esta ventana de inspección las posibilidades de realizar modificaciones son limitadas.

Visualizar y modificar los parámetros de la conexión HMI

1. Haga clic en la conexión HMI en el editor "Dispositivos y redes".
2. Modifique los parámetros de la conexión HMI en "Propiedades > General > General" de la ventana de inspección.



"Conexión"

Indica si los dispositivos ya están interconectados en red.

- Aparece cuando los dispositivos están interconectados en red.
- Aparece cuando los dispositivos no están interconectados en red.

"Vía de conexión"

En el área "Vía de conexión" se visualizan los interlocutores de la conexión HMI seleccionada y sus parámetros MPI. Algunas de las áreas visualizadas en este cuadro de diálogo no pueden editarse.

- "Punto final"
  Muestra el nombre del dispositivo. Esta área no es editable.
- "Interfaz"
  Muestra la interfaz seleccionada del dispositivo. En función del dispositivo es posible elegir entre varias interfaces.

- "Tipo de interfaz"
  Muestra el tipo de interfaz seleccionado. Esta área no es editable.
- "Subred"
  Muestra la subred seleccionada. Esta área no es editable.
- "Dirección"
  Muestra la dirección MPI seleccionada del dispositivo. Esta área no es editable.
- Botón "Buscar vía de conexión"
  Permite especificar las conexiones a posteriori.

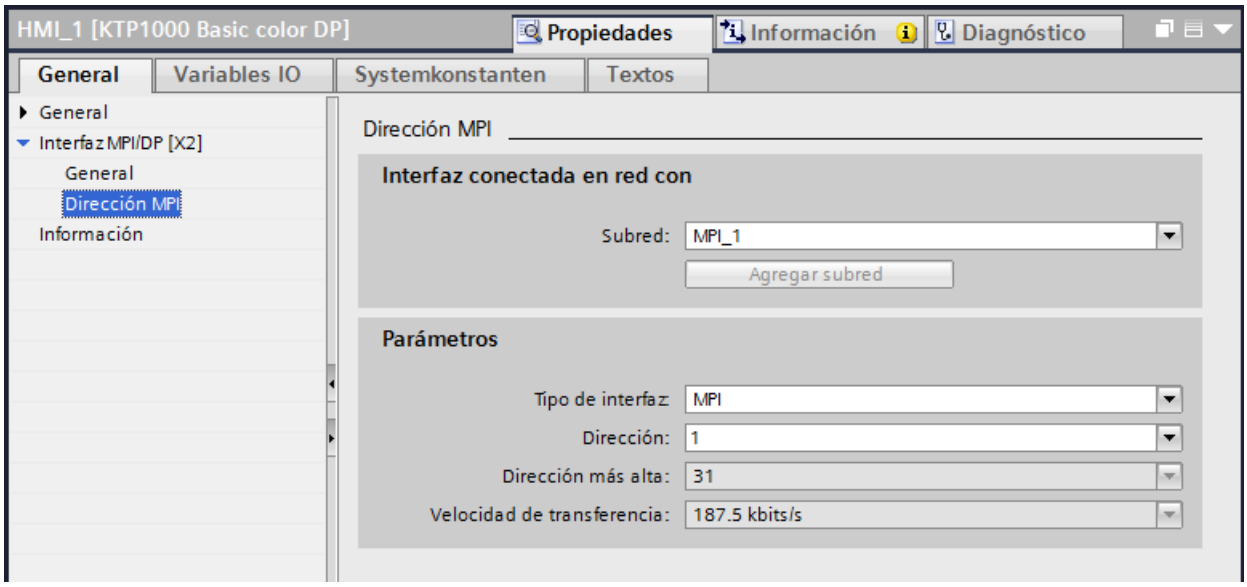
Parámetros MPI del panel de operador

Parámetros MPI del panel de operador

En las propiedades del panel de operador encontrará una vista general de los parámetros configurados del panel de operador.

Visualizar y modificar los parámetros MPI del panel de operador

1. Haga clic en el panel de operador en el editor "Dispositivos y redes".
2. Modifique los parámetros del panel de operador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Interfaz conectada en red con", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En el área "Tipo de interfaz" se parametriza el tipo de interfaz. En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección MPI del panel de operador. La dirección MPI debe ser unívoca en la red MPI.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red MPI.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

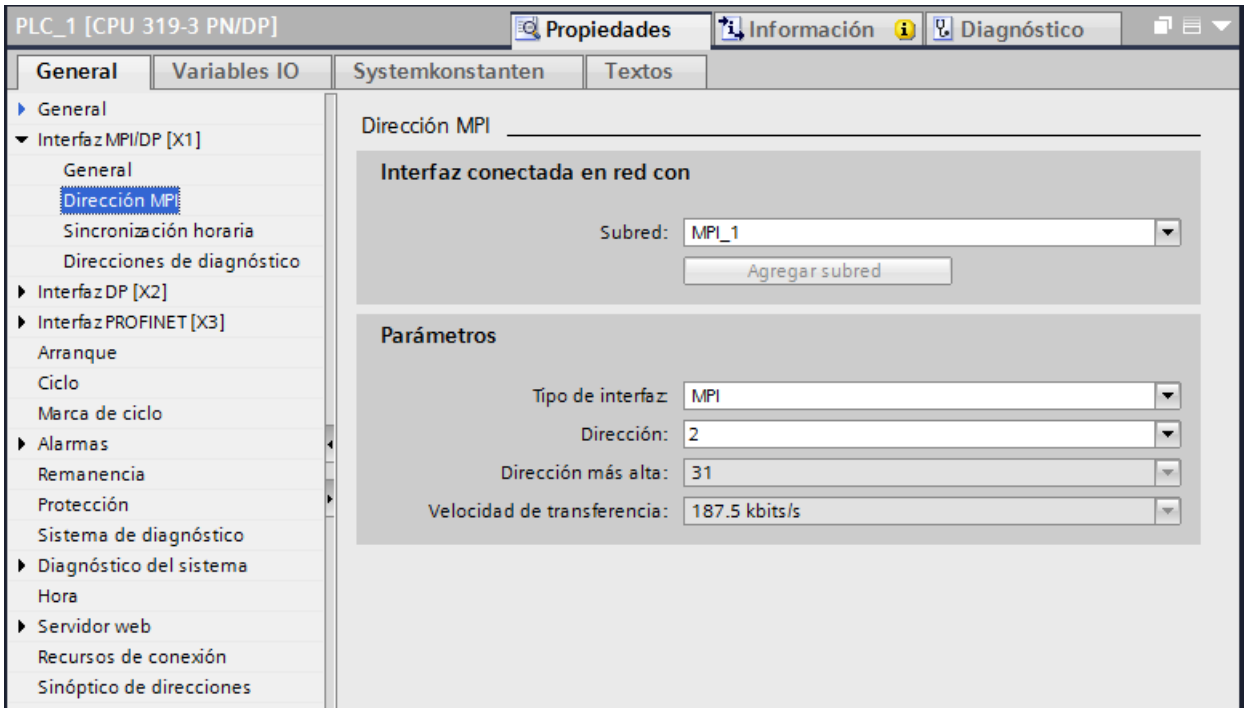
Parámetros MPI del controlador

Parámetros MPI del controlador

En las propiedades del controlador encontrará una vista general de los parámetros configurados.

Visualizar y modificar los parámetros PROFIBUS del controlador

1. Haga clic en el controlador en el editor "Dispositivos y redes".
2. Modifique los parámetros del controlador en "Propiedades > General > General" de la ventana de inspección.



"Interfaz conectada en red con"

En el área "Subred", seleccione la subred de la conexión HMI a través de la cual el panel de operador está conectado a la red. El botón "Agregar subred" permite crear una subred nueva.

"Parámetros"

- "Tipo de interfaz"
  En función del tipo de panel de operador habrá diferentes interfaces disponibles.
- "Dirección"
  En el área "Dirección" se parametriza la dirección MPI del panel de operador. La dirección MPI debe ser unívoca en la red MPI.
- "Dirección más alta"
  En el área "Dirección más alta" se muestra la dirección más alta de la red MPI.
- "Velocidad de transferencia"
  La "Velocidad de transferencia" queda determinada por el dispositivo más lento conectado a la red. Este ajuste es el mismo en toda la red.

Direccionamiento del controlador vía MPI

Introducción

Para que los interlocutores conectados a la red MPI puedan comunicarse entre sí, es preciso asignar una dirección MPI a cada uno de ellos.

Cada uno de los módulos S7 de comunicación integrados en el SIMATIC S7-300/400 tiene una dirección MPI unívoca. Sólo se puede utilizar una CPU por rack.

Nota

Los paneles de operador quedan inoperables si se asignan direcciones incorrectas.

Evite una asignación múltiple de las direcciones en el bus MPI.

Dirección MPI del interlocutor en un controlador SIMATIC S7-300

Para el direccionamiento se distingue entre los interlocutores con y sin dirección MPI propia.

- Tratándose de interlocutores con dirección MPI propia, indique sólo la dirección MPI.
- Tratándose de interlocutores sin dirección MPI propia, indique la dirección MPI del interlocutor a través del que se efectúa la conexión. Además, indique el slot y el rack del interlocutor sin dirección MPI propia.

Dirección MPI del interlocutor en un controlador SIMATIC S7-400

Los módulos S7 equipados con un conector MPI son los únicos que tienen una dirección MPI. Los módulos sin conector MPI se direccionan de forma indirecta:

- Dirección MPI del módulo conectado al panel de operador.
- Slot y rack en el que está insertado el módulo con el que se debe comunicar el panel de operador.

10.8.8.5 Intercambio de datos

Intercambio de datos mediante punteros de área

Generalidades sobre los punteros de área

Introducción

A través de un puntero de área se accede a un rango de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación.

Al evaluar los datos almacenados, el controlador y el panel de operador ejecutan acciones fijamente definidas.

Configurar punteros de área

Antes de utilizar un puntero de área, actívelos en "Conexiones > Punteros de área". Seguidamente, parametrícelos.

Encontrará más información sobre la configuración de punteros de área en:

Configurar un puntero de área (Página 8783)

Puntero de área "Número de imagen"

Función

Los paneles de operador depositan en el puntero de área "Número de imagen" información acerca de la imagen visualizada en el panel.

Ello permite transferir al controlador información acerca del contenido actual de la imagen en el panel de operador. En el controlador se pueden disparar determinadas reacciones, p. ej. la llamada de otra imagen.

Utilización

Antes de poder utilizar el puntero de área "Número de imagen", es necesario configurarlo y activarlo bajo "Comunicación > Conexiones". El puntero de área "Número de imagen" se puede crear únicamente en **un** controlador y depositarse allí **una sola vez**.

El número de imagen se transfiere siempre al controlador cuando se activa una nueva imagen o cuando el foco cambia de un objeto de imagen a otro dentro de una imagen.

Estructura

Este puntero de área es un área de datos de la memoria del controlador con una longitud fija de 5 palabras.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1ª palabra	Tipo de imagen actual															
2ª palabra	Número de imagen actual															
3ª palabra	Reservada															
4ª palabra	Número de campo actual															
5ª palabra	Reservada															

- Tipo de imagen actual
  "1" = imagen raíz
  "4" = ventana permanente
- Número de imagen actual
  1 a 32767
- Número de campo actual
  de 1 a 32767

Puntero de área "Fecha/hora"

Función

Este puntero de área sirve para transferir la fecha y la hora del panel de operador al controlador.

El controlador escribe la petición de control "41" ó "40" en la bandeja de peticiones.

Al evaluar la petición de control, el panel de operador escribe su fecha y hora actuales en el área de datos configurada en el puntero de área "Fecha/hora". Todos los datos se expresan en formato BCD.

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

Nota

Si ha configurado el puntero de área "Fecha/hora", no podrá utilizar el puntero de área "Fecha/hora PLC".

Nota

Si utiliza el puntero de área "Fecha/Hora", no es posible el direccionamiento simbólico.

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

El área de datos "Fecha/hora" tiene la siguiente estructura:

Palabra de datos	Byte más significativo								Byte menos significativo								
	7							0	7							0	
n+0	Reservada								Hora (0 a 23)								Hora
n+1	Minuto (0 a 59)								Segundo (0 a 59)								
n+2	Reservada								Reservada								
n+3	Reservada								Día de la semana (1 a 7, 1=domingo)								Fecha
n+4	Día (1 a 31)								Mes (1 a 12)								
n+5	Año (80 a 99/0 a 29)								Reservada								

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Puntero de área "Fecha/hora PLC"

Función

Este puntero de área sirve para transferir la fecha y la hora del controlador al panel de operador. Este puntero de área se utiliza si el controlador es el maestro que determina la hora.

El controlador carga el área de datos del puntero de área. Todos los datos se expresan en formato BCD.

Según el ciclo de adquisición configurado, el panel de operador lee los datos cíclicamente y se sincroniza.

Nota

No configure un ciclo de adquisición demasiado breve para el puntero de área "Fecha/hora", puesto que ello afecta el rendimiento del panel de operador.
Recomendación: Ciclo de adquisición = 1 minuto (si el proceso lo permite).

Si ha configurado el puntero de área "Fecha/hora PLC", no podrá utilizar el puntero de área "Fecha/hora".

"Fecha/hora PLC" es un puntero de área global y solo se puede configurar una vez en cada proyecto.

Nota

Si ha configurado el puntero de área "Fecha/hora PLC", no podrá utilizar el puntero de área "Fecha/hora".

El área de datos "Fecha/hora" tiene la siguiente estructura:

Formato DATE_AND_TIME (codificado en BCD)

Palabra de datos	Byte más significativo			Byte menos significativo		
	7	. . . . .	0	7	. . . . .	0
n+0	Año (80 a 99/0 a 29)			Mes (1 a 12)		
n+1	Día (1 a 31)			Hora (0 a 23)		
n+2	Minuto (0 a 59)			Segundo (0 a 59)		
n+3	Reservada			Reservada		Día de la semana (1 a 7, 1=domingo)
n+4 1)	Reservada			Reservada		
n+5 1)	Reservada			Reservada		

1) Ambas palabras de datos deben existir en el área de datos para garantizar la concordancia con el formato de datos de WinCC flexible y evitar la lectura de información errónea.

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Puntero de área "Coordinación"

Función

El puntero de área "Coordinación" sirve para realizar las funciones siguientes:

- Detectar en el programa de control el arranque del panel de operador
- Detectar en el programa de control el modo de operación actual del panel de operador
- Detectar en el programa de control si el panel de operador está dispuesto para comunicarse

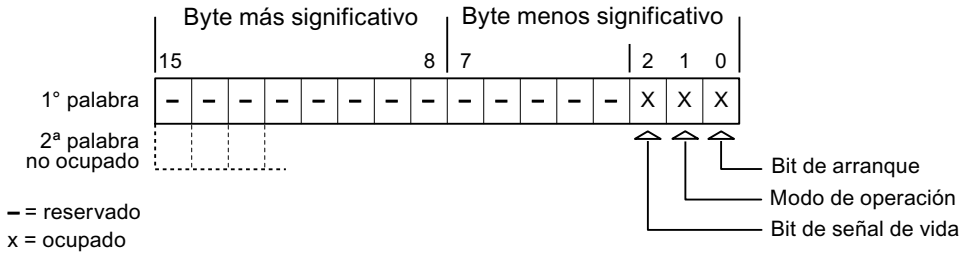
El puntero de área "Coordinación" tiene una longitud estándar de una palabra y no se puede modificar.

Utilización

Nota

Cada vez que el panel de operador actualiza el puntero de área, se escribe siempre toda el área de coordinación.
Por tanto, el programa de control no puede efectuar cambios en el área de coordinación.

Ocupación de los bits en el puntero de área "Coordinación"



Bit de arranque

Durante el arranque, el panel de operador pone el bit de arranque a "0" por breve tiempo. Después del arranque, el bit se pone a "1" de forma permanente.

Modo de operación

Cuando el usuario conmuta el panel de operador al modo offline, el bit de modo de operación de pone a "1". En el funcionamiento normal del panel de operador, el estado del bit de modo de operación es "0". Consultando este bit en el programa de control es posible averiguar el modo de operación actual del panel de operador.

Bit de vida

El panel de operador invierte el estado del bit de vida en intervalos de aprox. 1 segundo. Consultando este bit en el programa de control es posible averiguar si todavía existe una conexión con el panel de operador.

Puntero de área "Identificación del proyecto"

Función

Al iniciar runtime puede comprobarse si el panel de operador está conectado al controlador correcto. Esta comprobación es importante si se utilizan varios paneles de operador.

Para ello, el panel de operador compara un valor almacenado en el controlador con el valor indicado en la configuración. Así puede garantizarse la compatibilidad de los datos de configuración con los del programa de control. Si no hay compatibilidad, se muestra un aviso de sistema en el panel de operador y se detiene el runtime.

Utilización

Nota

Las conexiones HMI no pueden pasar a "online"

La conexión HMI en la que se utiliza el puntero de área "Identificador del proyecto" debe pasar a "online".

Para utilizar este puntero de área, especifique en la configuración lo siguiente:

- Versión de la configuración. El valor puede estar comprendido entre 1 y 255.
  La versión se indica en el editor "Configuración de runtime > General" en el área "Identificación".
- Dirección de datos del valor para la versión que está almacenada en el controlador:
  La dirección de datos se indica en el editor "Comunicación > Conexiones" en "Dirección".

Fallo de una conexión

Al fallar la conexión con un panel de operador en el que esté configurado el puntero de área "Identificador del proyecto", se desactivarán también todas las demás conexiones del dispositivo.

Para este comportamiento se deben cumplir los requisitos siguientes:

- En el proyecto se han configurado varias conexiones.
- En por lo menos una conexión se está utilizando el puntero de área "Identificador del proyecto".

Las causas siguientes pueden conmutar las conexiones al estado "offline":

- No se puede acceder al controlador.
- La conexión se ha conmutado a "offline" en el sistema de ingeniería.

Puntero de área "Petición de control"

Función

Desde la bandeja de peticiones de controlador se pueden lanzar peticiones de controlador al panel de operador y disparar con ellas acciones en el panel. Entre estas funciones se incluyen p. ej.:

- Mostrar imagen
- Ajustar la fecha y hora

Estructura de datos

En la primera palabra de la bandeja de peticiones de controlador figura el número de la petición. Dependiendo de la petición en cuestión, pueden transferirse hasta 3 parámetros como máximo.

Palabra	Byte más significativo	Byte menos significativo
n+0	0	Número de petición
n+1	Parámetro 1	
n+2	Parámetro 2	
n+3	Parámetro 3	

Si la primera palabra de la bandeja de peticiones de controlador es diferente de 0, el panel de operador evalúa la petición de controlador. Por este motivo hay que introducir primero los parámetros en la bandeja de peticiones de controlador y sólo entonces el número de la petición.

Una vez que el panel de operador ha aceptado la petición de controlador, la primera palabra se pone de nuevo a 0. Por lo general, la ejecución de la petición de control no se habrá completado todavía.

Peticiones de control

A continuación se indican todas las peticiones de control y sus respectivos parámetros. La columna "N.º" contiene el número de petición de control. Por regla general, las peticiones de control sólo pueden ser disparadas por el controlador si el panel de operador está en modo online.

N.º	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 1	Byte izquierdo: - Byte derecho: horas (0-23)
	Parámetro 2	Byte izquierdo: minutos (0-59) Byte derecho: segundos (0-59)
	Parámetro 3	-
15	Ajustar la fecha (codificada en BCD) [3) 4)]	
	Parámetro 1	Byte izquierdo: - Byte derecho: Día de la semana (1-7: Domingo-Sábado)
	Parámetro 2	Byte izquierdo: día (1-31) Byte derecho: mes (1-12)
	Parámetro 3	Byte izquierdo: año
23	Iniciar la sesión	
	Inicia la sesión del usuario "PLC User" y el número de grupo transferido en el parámetro 1 en el panel de operador. Para poder iniciar la sesión es necesario que el número de grupo transferido exista en el proyecto.	
	Parámetro 1	Número de grupo 1 - 255
	Parámetro 2, 3	-
24	Cerrar la sesión	

N.º	Función	
14	**Ajustar la hora (codificada en BCD)**	
	Cierra la sesión del usuario actual. (Esta función equivale a la función de sistema "CerrarSesión")	
	Parámetro 1, 2, 3	-
40	**Transferir fecha/hora al controlador**	
	(En formato S7 DATE_AND_TIME) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
41	**Transferir fecha/hora al controlador**	
	(En formato OP/MP) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
46	**Actualizar variable**	
	Hace que el panel de operador lea el valor actual de la variable del controlador cuyo identificador de actualización coincide con el valor transferido en el parámetro 1. (Esta función equivale a la función de sistema "ActualizarVariable")	
	Parámetro 1	1 - 100
49	**Borrar búfer de avisos**	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Warnings" del búfer de avisos.	
	Parámetro 1, 2, 3	-
50	**Borrar búfer de avisos**	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Errors" del búfer de avisos.	
	Parámetro 1, 2, 3	-
51	**Selección de imagen** [2]	
	Parámetro 1	Número de imagen
	Parámetro 2	-
	Parámetro 3	Número de campo
69	**Leer registro del controlador** [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	0: No sobrescribir el registro existente 1: Sobrescribir el registro existente
70	**Escribir registro en el controlador** [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	-

[1]	Sólo en paneles que soporten recetas.
[2]	En los paneles de operador OP 73, OP 77A y TP 177A la petición de control "Selección de imagen" se ejecuta incluso con el teclado de pantalla abierto.

3)	En el panel de operador KTP 600 BASIC PN se ignora el día de la semana.
4)	Si utiliza el puntero de área "Fecha/hora PLC", se ignora el día de la semana.

Puntero de área "Registro"

Puntero de área "Registro"

Función

Al transferir registros entre el panel de operador y el controlador, ambos interlocutores acceden alternativamente a las áreas de comunicación conjuntas del controlador.

Tipos de transferencia

Existen dos posibilidades para transferir registros entre el panel de operador y el controlador, a saber:

- Transferencia sin sincronización
- Transferencia con sincronización a través de la bandeja de datos

Los registros se transfieren siempre directamente. Por tanto, los valores de variables se leen o escriben directamente de/en la dirección configurada para la variable, sin desviarse al portapapeles.

Iniciar la transferencia de registros

Para iniciar la transferencia existen tres posibilidades:

- Inicio desde la vista de recetas
- Tareas de control
  El autómata también puede disparar la transferencia de registros.
- Activar funciones configuradas

Si la transferencia de registros se dispara mediante una función configurada o una orden de control, la vista de recetas seguirá pudiéndose controlar sin impedimentos desde el panel de operador. Los registros se transfieren en segundo plano.

Sin embargo, no es posible procesar simultáneamente varias peticiones de transferencia. En este caso, el panel de operador deniega una segunda transferencia con un aviso de sistema.

Transferencia sin sincronización

En la transferencia asíncrona de registros entre el panel de operador y el controlador no se coordinan las áreas de datos utilizadas conjuntamente. Por ello, no es necesario crear un área de datos en la configuración.

La transferencia asíncrona de registros resulta útil p. ej. en los casos siguientes:

- Las propiedades del sistema impiden que los interlocutores puedan sobrescribir los datos de forma incontrolada.
- El controlador no necesita información sobre el número de receta o de registro.
- La transferencia de registros se inicia por operación desde el panel de operador.

Leer valores

Al iniciar la transferencia para la lectura se leen los valores de las direcciones de control y se transfieren al panel de operador.

- Inicio por operación desde la vista de recetas:
  Los valores se cargan en el panel de operador. Éstos se pueden procesar posteriormente en el panel de operador (p. ej. se pueden modificar o guardar valores, etc.)
- Inicio por función o petición del controlador:
  Los valores se almacenan inmediatamente en el soporte de datos.

Escribir valores

Al iniciar la transferencia para escritura se escriben los valores en las direcciones de control.

- Inicio por operación desde la vista de recetas:
  Los valores actuales se escriben en el controlador.
- Inicio por función o petición del controlador:
  Los valores del soporte de datos se escriben en el controlador.

Transferencia con sincronización

En la transferencia sincronizada, ambos interlocutores activan determinados bits de estado en el área de datos utilizada conjuntamente. De este modo se impide que los datos se sobrescriban unos a otros de forma incontrolada en el programa de control.

Aplicación

La transferencia síncrona de registros resulta útil p. ej. en los casos siguientes:

- El controlador es el "interlocutor activo" en la transferencia de registros.
- En el controlador se evalúa información sobre los números de receta y de registro.
- La transferencia de registros se dispara mediante una petición del controlador.

Requisitos

Para que los registros puedan transferirse de forma sincronizada entre el panel de operador y el controlador, en el proceso de configuración deben cumplirse las siguientes condiciones:

- Se ha configurado un puntero de área: Editor "Comunicación > Conexiones" bajo "Punteros de área".
- En la receta está indicado el controlador con el que el panel de operador sincroniza la transferencia de registros:
  Editor "Recetas" en el área "General > Sincronización > Configuración" de la ventana de inspección, y selección "Transferencia de datos coordinada" activada.

Estructura del área de datos

El área de datos tiene una longitud fija de 5 palabras. El área de datos está estructurada de la siguiente manera:

	15		0
1. palabra	Número de receta actual (1-999)		
2. palabra	Número de registro actual (0-65.535)		
3. palabra	Reservada		
4. palabra	Estado (0, 2, 4, 12)		
5. palabra	Reservada		

- Estado
  La palabra de estado (palabra 4) acepta los siguientes valores:

Valor		Significado
Decimal	Binario	
0	0000 0000	Transferencia permitida, bandeja de datos libre
2	0000 0010	Transferencia en curso.
4	0000 0100	Transferencia finalizada sin fallos
12	0000 1100	Transferencia finalizada con fallos

Ejecución de la transferencia al iniciarla desde el visor de recetas

Leer del autómata desde la vista de recetas

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta que ha de ser leído, así como el estado "Transferencia en curso" en la bandeja de datos y repone el número de registro a 0.	Cancelación con aviso de sistema.

Paso	Acción	
3	El panel de operador lee los valores del autómata y los muestra en la vista de recetas. En las recetas con variables sincronizadas, los valores del autómata también se escriben en las variables.	
4	El panel de operador activa el estado "Transferencia finalizada".	
5	Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Escribir en el autómata desde la vista de recetas

Paso	Acción	
	Verificación: ¿Palabra de estado = 0?	
1	Sí	No
	El panel de operador introduce el número de receta y el número de registro que deben escribirse, así como como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
2	El panel de operador escribe los valores actuales en el autómata. En las recetas con variables sincronizadas, los valores modificados son sincronizados entre la vista de recetas y las variables, escribiéndose luego en el autómata.	
3	El panel de operador activa el estado "Transferencia concluida".	
4	Dado el caso, el programa de control puede evaluar entonces los datos transferidos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Nota

El panel de operador es el único que puede activar la palabra de estado. El autómata sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el autómata sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos está activado a "Transferencia finalizada".
- El estado en la bandeja de datos está activado a "Transferencia finalizada con fallos".

Ejecución de la transferencia en caso de petición de control

El panel de operador o el autómata pueden iniciar la transferencia de registros entre sí.

Para este tipo de transferencia se dispone de las tareas de control n° 69 y n° 70.

N° 69: Leer registro del autómata ("SPS → DAT")

La orden de control n° 69 transfiere los registros del autómata al panel de operador. La orden de control está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	69
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	No sobrescribir el registro existente: 0 Sobrescribir el registro existente: 1	

N° 70: Escribir registro en el autómata ("DAT → SPS")

La orden de control n° 70 transfiere los registros del panel de operador al autómata. La petición del controlador está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	70
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	—	

Ejecución al leer del autómata mediante la orden de control "SPS → DAT" (n° 69)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la tarea, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la orden de control.	
4	• Si en la tarea se ha seleccionado "Sobrescribir", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia concluida". • Si en la tarea se ha seleccionado "No sobrescribir" y ya existe el registro, el panel de operador cancelará la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Ejecución al escribir en el autómata mediante la orden de control "DAT → SPS" (n° 70)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce en la bandeja de datos el número de receta y de registro indicados en la petición, así como el estado "Transferencia en curso".	Cancelación sin respuesta.
3	El panel de operador toma los valores del registro indicado en la tarea del portador de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Ejecución de la transferencia con inicio mediante una función configurada

Leer en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la función.	
4	• Si en la función "Sobrescribir" se ha seleccionado "Sí", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia concluida". • Si en la función "Sobrescribir" se ha seleccionado "No" y el registro ya existe, el panel de operador interrumpirá la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Escribir en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No

Paso	Acción	
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de control recoge los valores del registro indicado en la función del soporte de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Posibles causas de error en la transferencia de registros

Causas de error posibles

Si la transferencia de registros termina con errores, puede deberse a las siguientes causas:

- No se ha configurado ninguna dirección de variable en el autómata
- No es posible sobrescribir registros
- El número de receta no existe
- El número de registro no existe

Nota

El panel de operador es el único que puede activar la palabra de estado. El controlador sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el controlador sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos se ha activado a "Transferencia finalizada".
- El estado en la bandeja de datos se ha activado a "Transferencia finalizada con fallos".

Reacción a una interrupción debida a un fallo

El panel de operador reacciona a una interrupción de la transferencia de registros debida a un fallo de la siguiente forma:

- Inicio por el usuario desde la vista de recetas:
  Indicaciones en la barra de estado de la vista de recetas y salida de avisos de sistema
- Inicio mediante una función
  Emisión de avisos de sistema
- Inicio mediante una orden de control
  No hay respuesta en el panel de operador

Independientemente de ello, el usuario puede evaluar el estado de la transferencia consultando la palabra de estado en la bandeja de datos.

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos

Para configurar avisos tales como avisos de operador, alarmas y acuses se requieren varios pasos.

- Paso 1: Crear variables
- Paso 2: Configurar avisos
- Paso 3: Configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- Tipos de datos de la variable
- Direccionamiento de la variable
- Modo de contaje de las posiciones de bit

Tipos de datos

Para conexiones con drivers de comunicación SIMATIC se soportan los tipos de datos siguientes:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
SIMATIC S7 300/400	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, COUNTER, TIME

Modo de contaje de las posiciones de bit

Para conexiones con drivers de comunicación SIMATIC rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte 0								Byte 1							
	Byte más significativo								Byte menos significativo							
En controladores SIMATIC S7	7							0	7							0
Configure en WinCC:	15							8	7							0

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

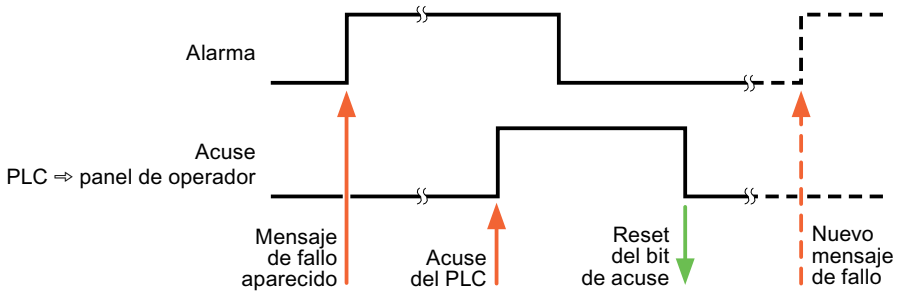
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

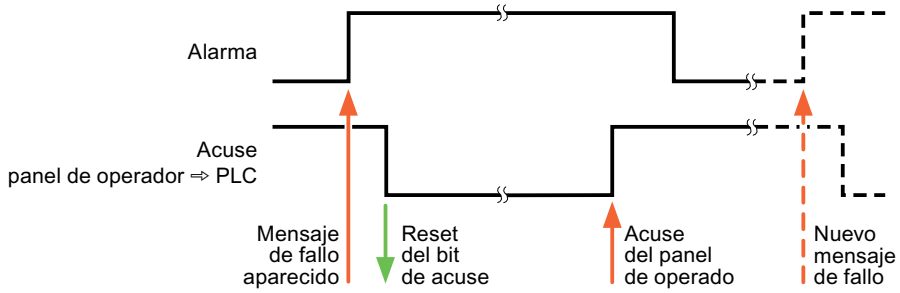
Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



10.8.8.6 Características de rendimiento de la comunicación

Tipos de datos admisibles para SIMATIC S7 300/400

Tipos de datos admisibles para conexiones con SIMATIC S7 300/400

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Longitud
BOOL	1 bit
BYTE	1 byte
WORD	2 bytes
DWORD	4 bytes
CHAR	1 byte
INT	2 bytes
DINT	4 bytes
REAL	4 bytes
TIME	4 bytes
DATE	2 bytes
TIME_OF_DAY, TOD	4 bytes
S5TIME	2 bytes
COUNTER	2 bytes
TEMPORIZADOR	2 bytes
DATE_AND_TIME	8 bytes
STRING	(2+n) bytes, n = de 0 a 254

10.8.8.7 Crear conexiones en el editor "Conexiones"

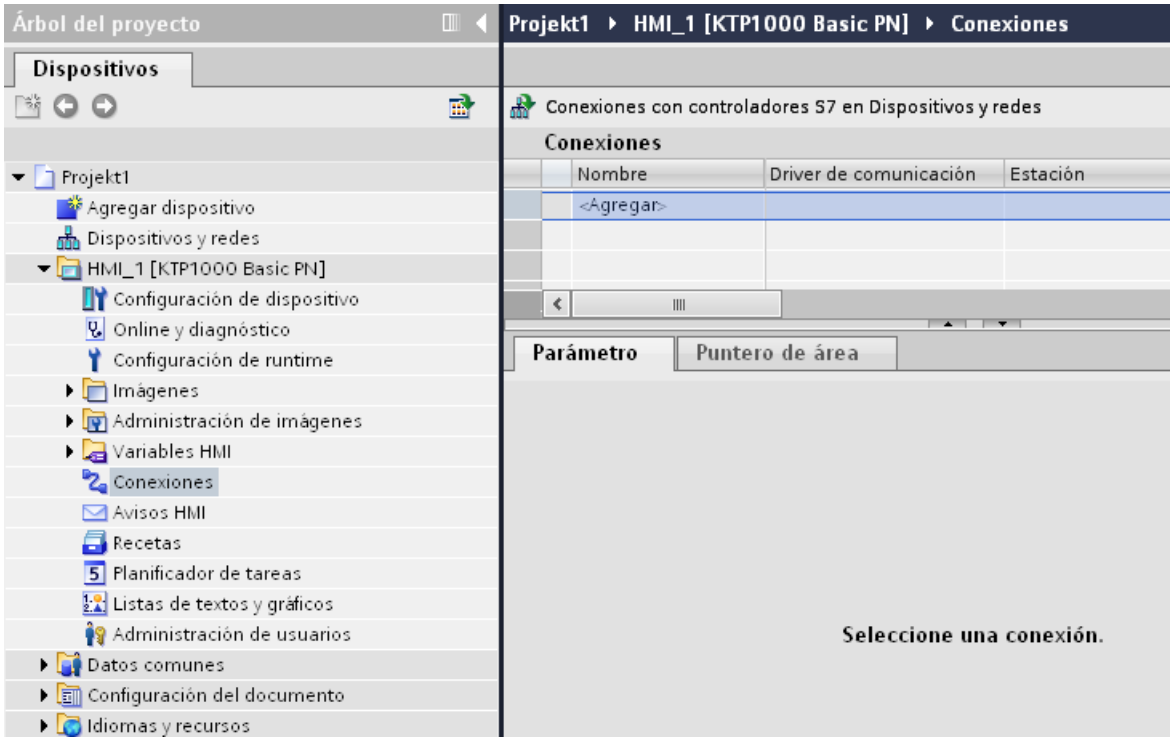
Crear una conexión PROFINET

Requisitos

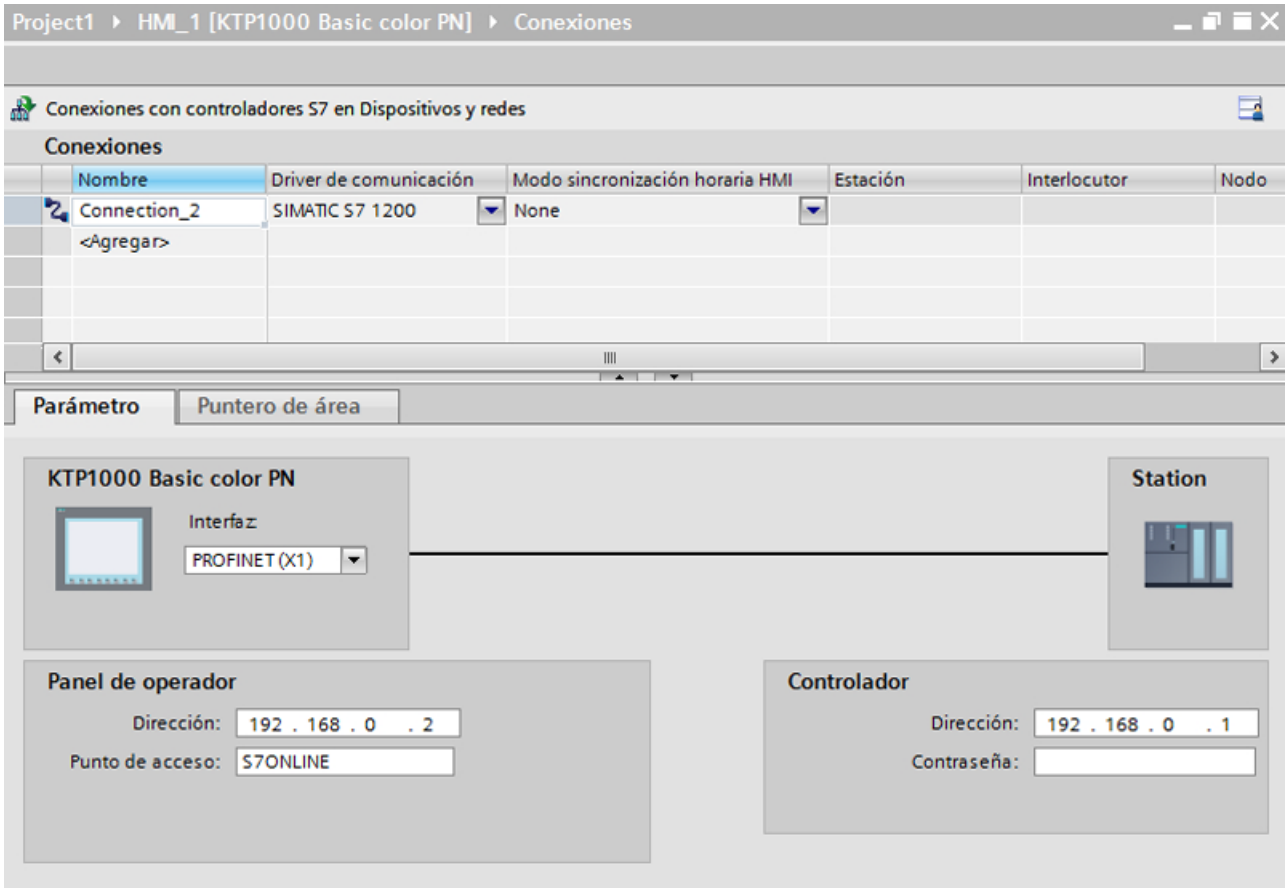
- Hay un proyecto abierto.
- Se ha creado un panel de operador con interfaz PROFINET.

Procedimiento

1. Abra el editor "Conexiones" del panel de operador.
2. Haga doble clic en "<Agregar>".



3. Seleccione el controlador en la columna "Drivers de comunicación".



4. Haga clic en el nombre de la conexión.
5. Seleccione una interfaz PROFINET del panel de operador en "Propiedades > Interfaz" de la ventana de inspección.
6. Ajuste las direcciones IP de los interlocutores en la ventana de inspección:
   - Panel de operador: "Parámetros > Panel de operador > Dirección"
   - Controlador: "Parámetros > Controlador > Dirección"

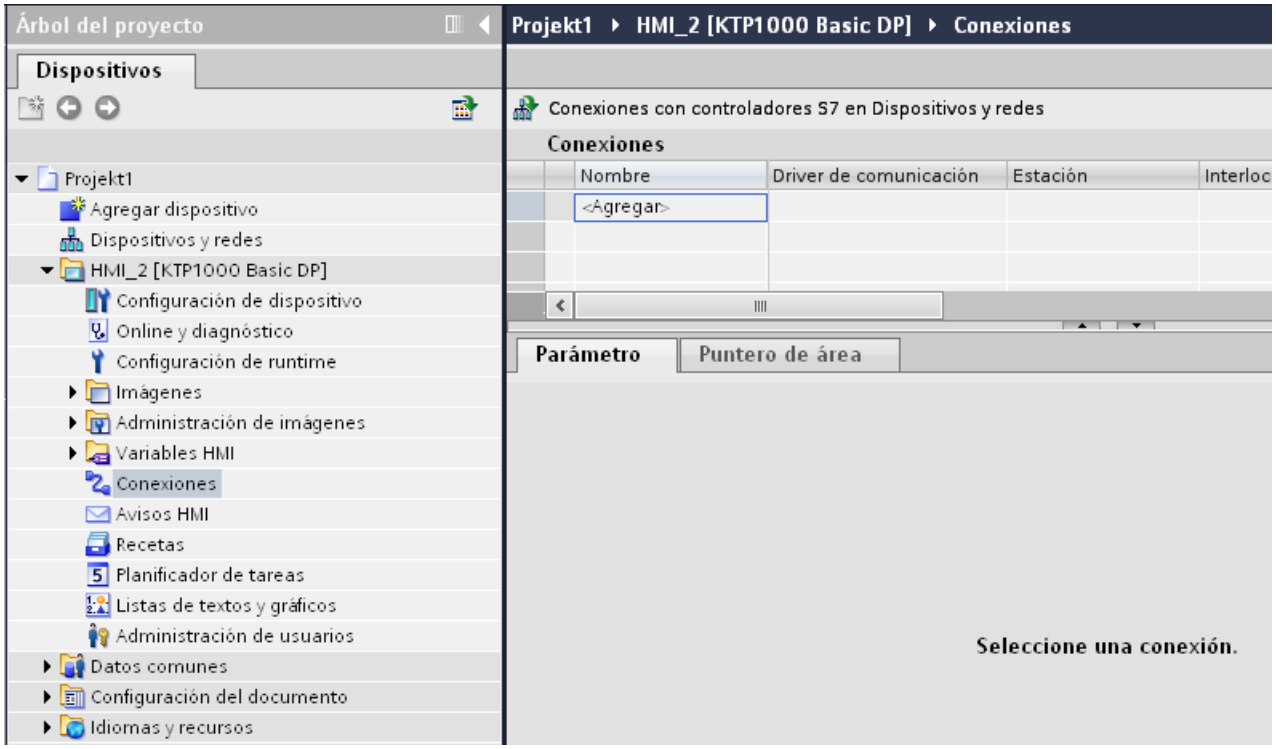
Crear conexión PROFIBUS

Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador con interfaz PROFIBUS.

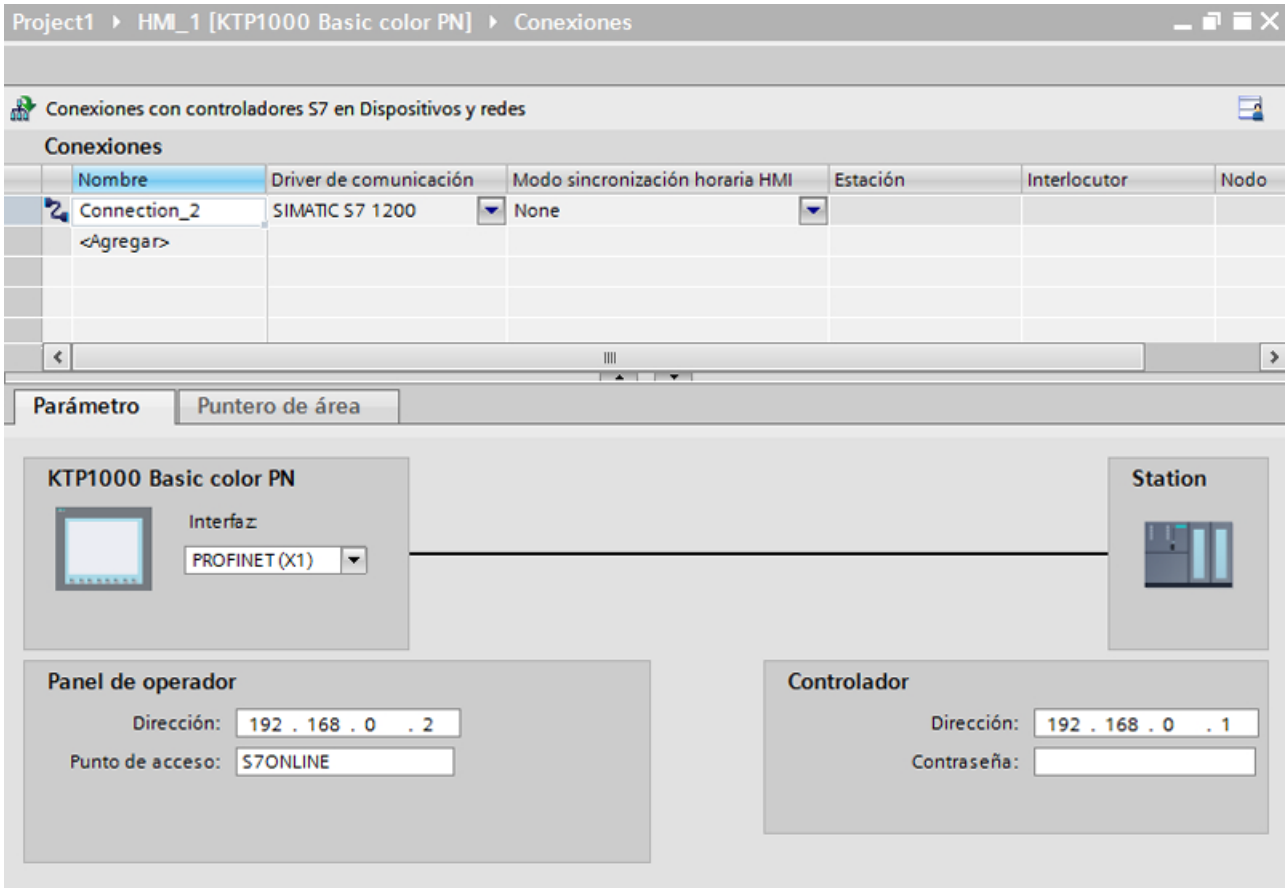
Procedimiento

1. Abra el editor "Conexiones" del panel de operador.
2. Haga doble clic en "<Agregar>".



3. Seleccione el controlador en la columna "Drivers de comunicación".
4. Haga clic en el nombre de la conexión.
5. Seleccione la interfaz "MPI/DP" en "Parámetros > Interfaz" de la ventana de inspección.

6. Seleccione el perfil "DP" en "Parámetros > Red" de la ventana de inspección.



7. Ajuste las direcciones de los interlocutores en la ventana de inspección:
   - Panel de operador: "Parámetros > Panel de operador > Dirección"
   - Controlador: "Parámetros > Controlador > Dirección"

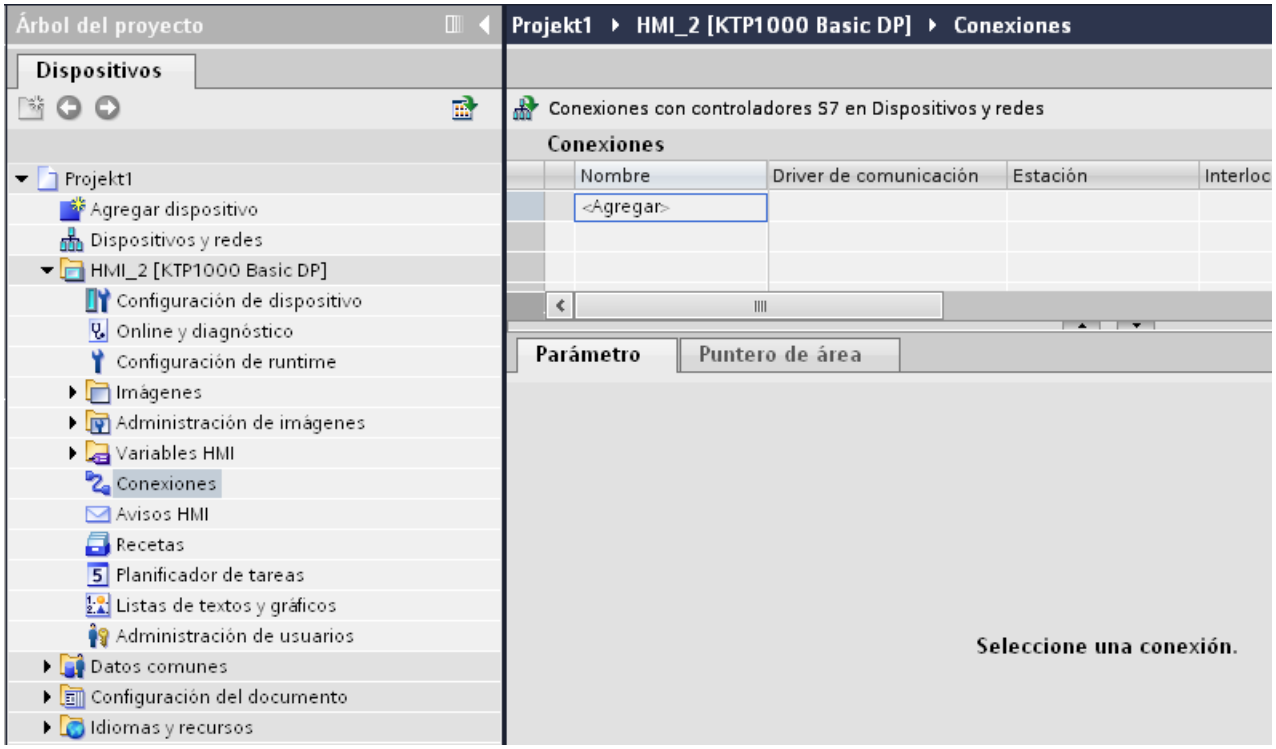
Crear una conexión MPI

Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador con interfaz MPI.

Procedimiento

1. Abra el editor "Conexiones" del panel de operador.
2. Haga doble clic en "<Agregar>".



3. Seleccione el controlador en la columna "Drivers de comunicación".
4. Haga clic en el nombre de la conexión.
5. Seleccione la interfaz "MPI/DP" en "Parámetros > Interfaz" de la ventana de inspección.

6. Seleccione el perfil "MPI" en "Parámetros > Red" de la ventana de inspección.

7. Ajuste las direcciones de los interlocutores en la ventana de inspección:
   - Panel de operador: "Parámetros > Panel de operador > Dirección"
   - Controlador: "Parámetros > Controlador > Dirección"

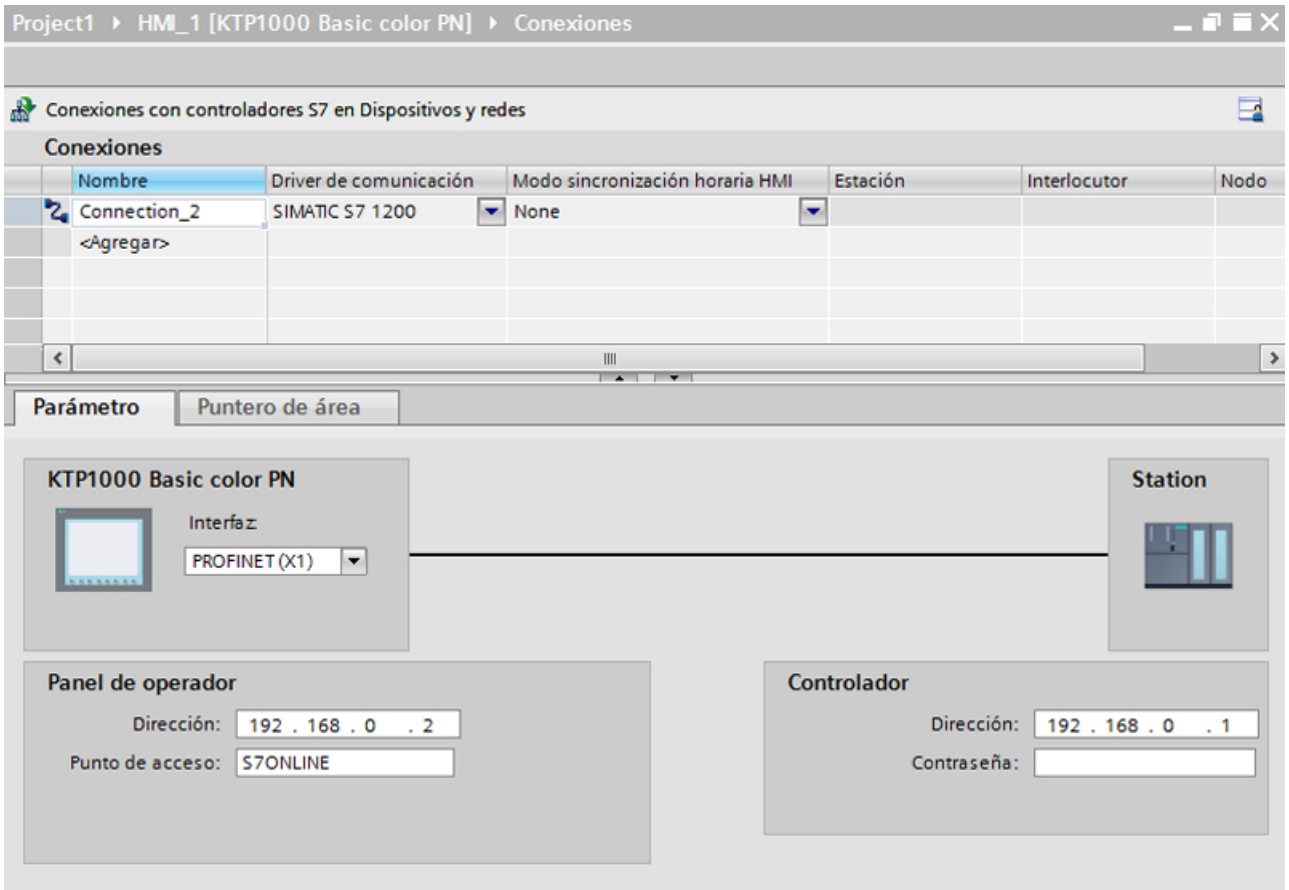
Parámetros de la conexión

Parámetros de la conexión (SIMATIC S7 300/400)

Parámetros que deben ajustarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador", "Red" y "Controlador", según sea la interfaz utilizada.


Project1 ▸ HMI_1 [KTP1000 Basic color PN] ▸ Conexiones
Conexiones con controladores S7 en Dispositivos y redes
Conexiones
Nombre
Driver de comunicación
Modo sincronización horaria HMI
Estación
Interlocutor
Nodo
Connection_2
SIMATIC S7 1200
None
<Agregar>
Parámetro
Puntero de área
KTP1000 Basic color PN
Interfaz:
PROFINET (X1)
Station
Panel de operador
Dirección: 192 . 168 . 0 . 2
Punto de acceso: S7ONLINE
Controlador
Dirección: 192 . 168 . 0 . 1
Contraseña:

Parámetros Ethernet

Parámetros del panel de operador

Ajuste en "Panel de operador" los parámetros para conectar el panel de operador a la red. Los cambios no se transfieren automáticamente al panel de operador. La configuración se debe modificar en el panel de control del panel de operador.

- "Interfaz"
  Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC.

  Nota

  Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

  Si activa "Obtener dirección IP por otra vía", en la siguiente carga se conserva en el Control Panel la dirección IP establecida previamente.

  Al transferir el proyecto se transfiere la dirección IP al panel de operador.
  La dirección IP del panel de operador se configura del siguiente modo:

  – Haga clic en el panel de operador.
  – Abra el editor "Configuración de dispositivos".
  – Haga clic en la interfaz Ethernet.
  – Asigne la dirección IP en la ventana de inspección en:
    "General > Interfaz PROFINET > Direcciones Ethernet"

- "Dirección"
  En el área "Dirección" se asigna la dirección IP del panel de operador.
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección", determine la dirección IP del módulo S7 conectado al panel de operador.
- "Slot de ampliación"
  Determina el número del slot de ampliación de la CPU que se debe direccionar.

- "Rack"
  Determina el número del rack de la CPU que se debe direccionar.
- "Funcionamiento cíclico"

  Nota

  El ajuste "Funcionamiento cíclico" no se puede configurar en el controlador SIMATIC S7 1200.

  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento.
  Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador.

Parámetros PROFIBUS

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red PROFIBUS.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro).
  En el S7-200 hay que configurar un panel de operador como maestro.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros de la red

En "Red" se ajustan los parámetros de la red PROFIBUS en la que está integrado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "DP", "Universal" o "Estándar" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección PROFIBUS real más alta. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Número de maestros"
  En "Número de maestros" se ajusta el número de maestros en la red PROFIBUS. Este dato es necesario para calcular correctamente los parámetros del bus.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se determina la dirección PROFIBUS del módulo S7 (CPU, FM o CP) conectado al panel de operador.
- "Funcionamiento cíclico"

  Nota

  El ajuste "Funcionamiento cíclico" no se puede configurar en el controlador SIMATIC S7 1200.

  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento. Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador. Este ajuste no es necesario para los SIMATIC S7-200.

Parámetros MPI

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red MPI.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección MPI del panel de operador. La dirección MPI debe ser unívoca en la red MPI.
- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro). Si sólo se han conectado esclavos en el panel de operador, hay que desactivar la función de seguridad "Único maestro del bus".
  En el S7-200 hay que configurar un panel de operador como maestro.

Parámetros de la red

En "Red" se ajustan los parámetros de la red MPI con la que está conectado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "MPI" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección MPI real más alta. Este ajuste debe ser el mismo en toda la red.
- "Número de maestros"
  Este dato no es necesario en MPI.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se determina la dirección MPI del módulo S7 (CPU, FM o CP) conectado al panel de operador.
- "Funcionamiento cíclico"
  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento. Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador. Este ajuste no es necesario para los SIMATIC S7-200.

Funcionamiento cíclico

Uso de la selección "Funcionamiento cíclico"

Si está activado el "Funcionamiento cíclico", al principio de la comunicación el panel de operador enviará un telegrama al controlador con la información de que determinadas variables se requieren constantemente.

A partir de entonces el controlador enviará los datos siempre en el mismo ciclo. De este modo el panel de operador evita tener que pedir los datos una y otra vez.

Si el funcionamiento cíclico está desactivado, el panel de operador enviará una petición especial para cada información que necesite.

Otras características:

- El funcionamiento cíclico da soporte al panel de operador en la transferencia de datos reduciendo su carga. Para reducir la carga del panel de operador se utilizan los recursos existentes en el controlador.
- El controlador soporta sólo un número determinado de servicios cíclicos. Cuando el controlador ya no dispone de recursos para los servicios cíclicos, entonces el panel de operador se vuelve a hacer cargo de la operación.
- Si el controlador no soporta el funcionamiento cíclico, entonces el panel de operador se encarga de formar el ciclo.
- Las variables de imagen no se incorporan al funcionamiento cíclico.
- El funcionamiento cíclico se configura únicamente al reiniciar el runtime.
- Cuando está activado el funcionamiento cíclico, dependiendo del controlador, se transfiere más de una petición del panel de operador al controlador.
- Cuando está desactivado el funcionamiento cíclico, se transfiere sólo una petición del panel de operador al controlador.

10.8.9 Comunicación con SIMATIC S7 200

10.8.9.1 Comunicación con SIMATIC S7 200

Introducción

Este apartado describe la comunicación entre un panel de operador y el controlador SIMATIC S7 200.

Se puede configurar los siguientes canales de comunicación para el controlador SIMATIC S7 200:

- PROFINET y Ethernet
- PROFIBUS
- MPI
- PPI

Conexión HMI para la comunicación

Las conexiones entre un panel de operador y un SIMATIC S7 200 se configuran en el editor "Conexiones" del panel de operador.

10.8.9.2 Crear una conexión con SIMATIC S7 200

Introducción

Una conexión con el controlador SIMATIC S7 200 se configura en el editor "Conexiones" del panel de operador. Las interfaces reciben nombres distintos en función del panel de operador.

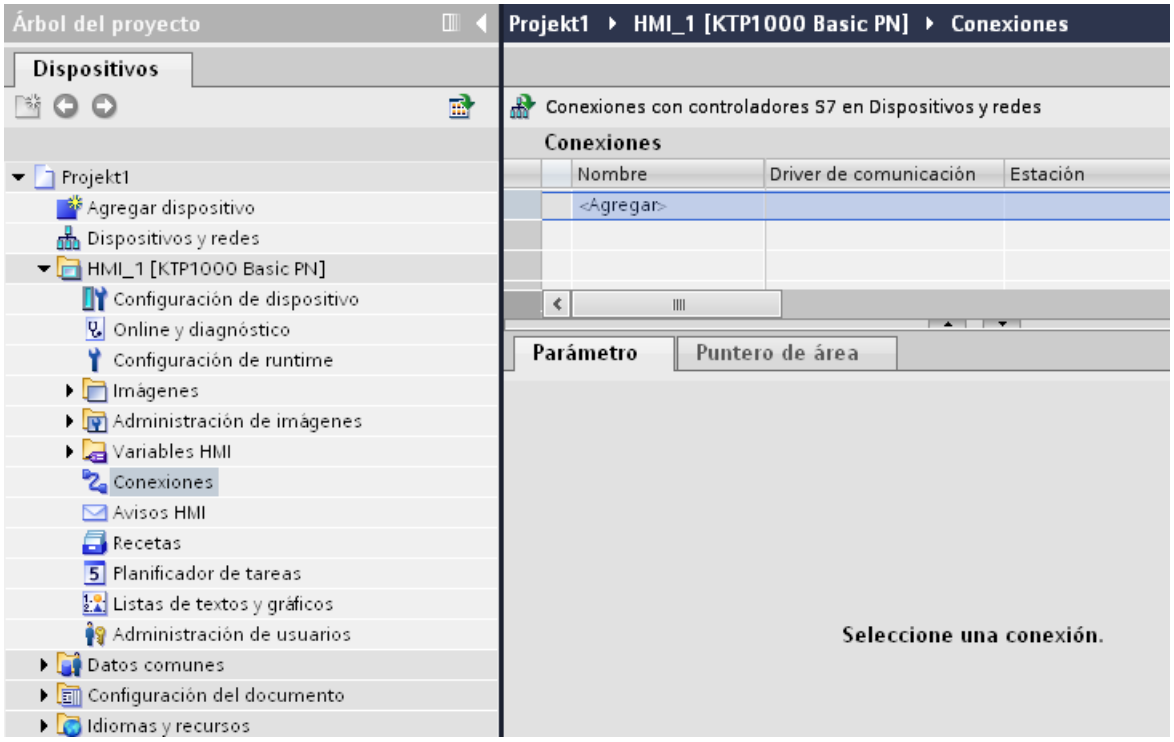
Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador.

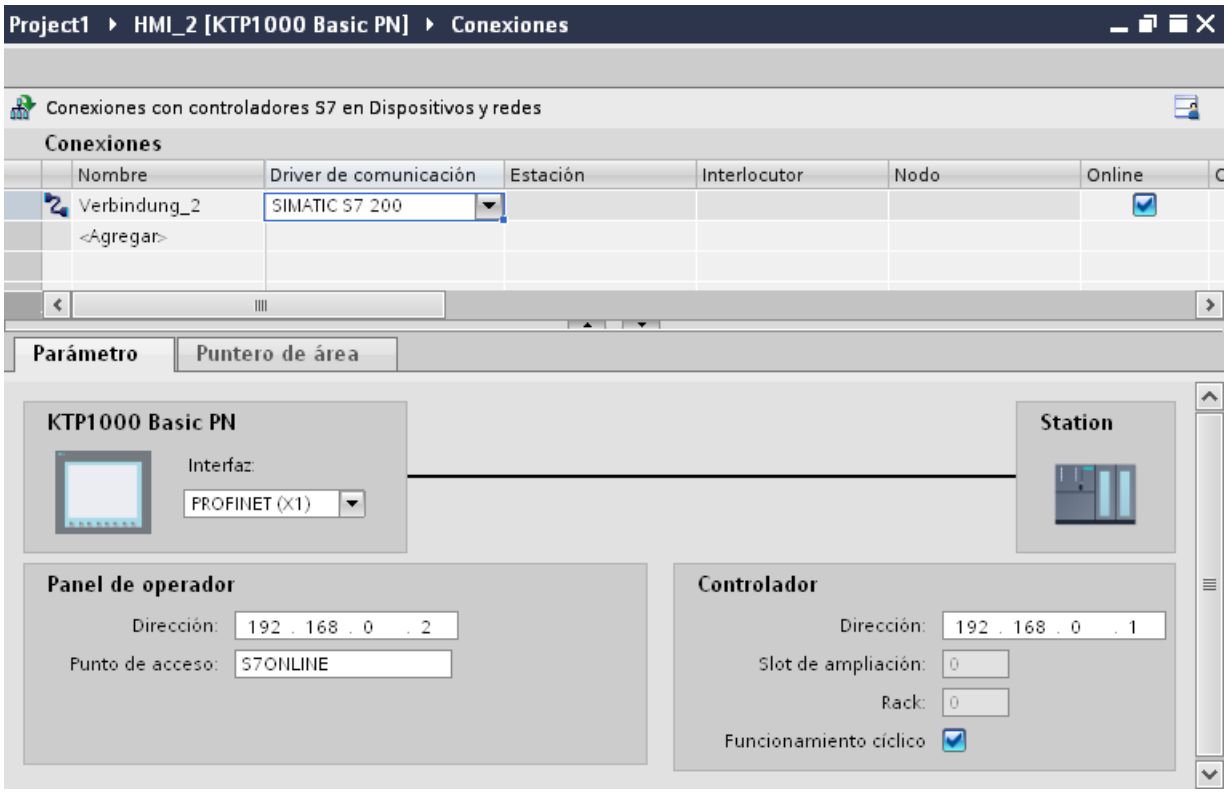
Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".

3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el driver "SIMATIC S7 200" en la columna "Driver de comunicación".
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.



Encontrará más información al respecto en el capítulo "Parámetros de la conexión (Página 8961)".

10.8.9.3 Parámetros de la conexión

Funcionamiento cíclico

Uso de la selección "Funcionamiento cíclico"

Si está activado el "Funcionamiento cíclico", al principio de la comunicación el panel de operador enviará un telegrama al controlador con la información de que determinadas variables se requieren constantemente.

A partir de entonces el controlador enviará los datos siempre en el mismo ciclo. De este modo el panel de operador evita tener que pedir los datos una y otra vez.

Si el funcionamiento cíclico está desactivado, el panel de operador enviará una petición especial para cada información que necesite.

Otras características:

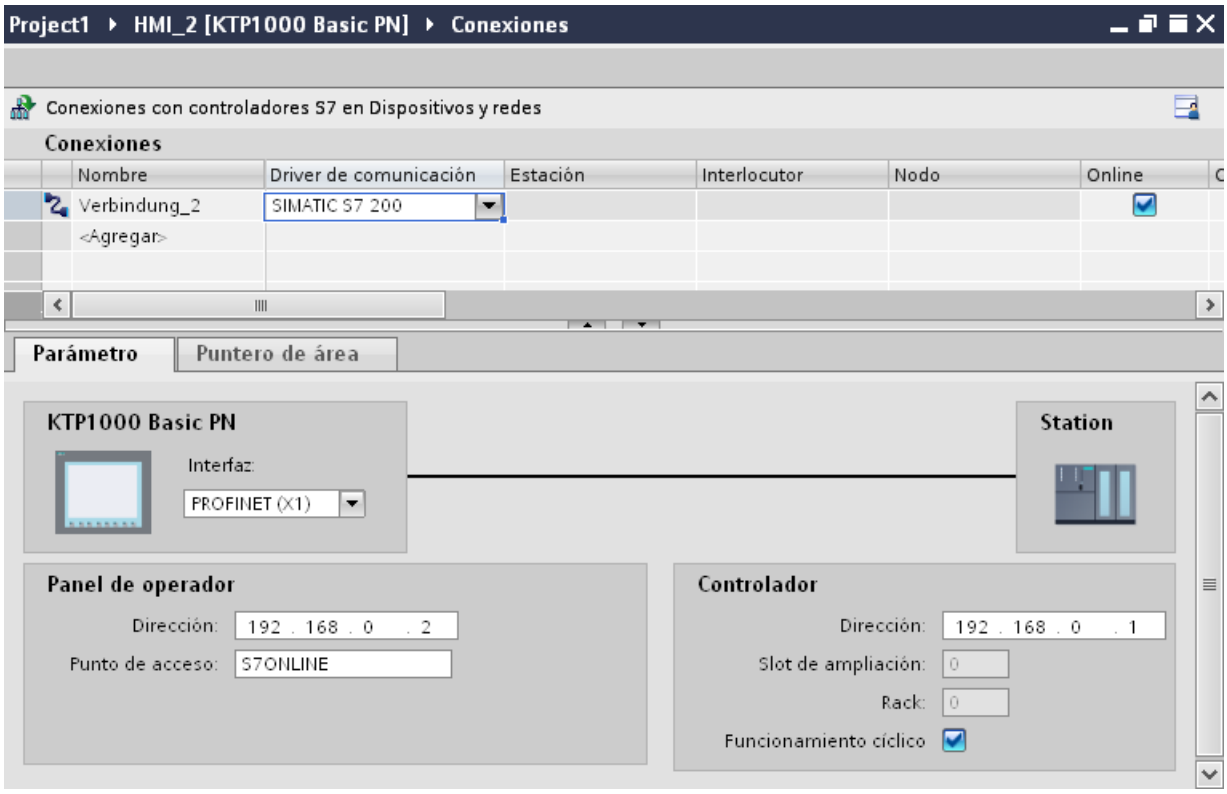
- El funcionamiento cíclico da soporte al panel de operador en la transferencia de datos reduciendo su carga. Para reducir la carga del panel de operador se utilizan los recursos existentes en el controlador.
- El controlador soporta sólo un número determinado de servicios cíclicos. Cuando el controlador ya no dispone de recursos para los servicios cíclicos, entonces el panel de operador se vuelve a hacer cargo de la operación.
- Si el controlador no soporta el funcionamiento cíclico, entonces el panel de operador se encarga de formar el ciclo.
- Las variables de imagen no se incorporan al funcionamiento cíclico.
- El funcionamiento cíclico se configura únicamente al reiniciar el runtime.
- Cuando está activado el funcionamiento cíclico, dependiendo del controlador, se transfiere más de una petición del panel de operador al controlador.
- Cuando está desactivado el funcionamiento cíclico, se transfiere sólo una petición del panel de operador al controlador.

Parámetros de la conexión (SIMATIC S7 200)

Parámetros que deben ajustarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador" y "Controlador", según sea la interfaz utilizada.


Project1 ▸ HMI_2 [KTP1000 Basic PN] ▸ Conexiones
Conexiones con controladores S7 en Dispositivos y redes
Conexiones
Nombre
Driver de comunicación
Estación
Interlocutor
Nodo
Online
Verbindung_2
SIMATIC S7 200
<Agregar>
Parámetro
Puntero de área
KTP1000 Basic PN
Interfaz:
PROFINET (X1)
Station
Panel de operador
Dirección:
192 . 168 . 0 . 2
Punto de acceso:
S7ONLINE
Controlador
Dirección:
192 . 168 . 0 . 1
Slot de ampliación:
0
Rack:
0
Funcionamiento cíclico

Parámetros Ethernet

Parámetros del panel de operador

Ajuste en "Panel de operador" los parámetros para conectar el panel de operador a la red. Los cambios no se transfieren automáticamente al panel de operador. La configuración se debe modificar en el panel de control del panel de operador.

- "Interfaz"
  Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC.

  Nota

  Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

  Si activa "Obtener dirección IP por otra vía", en la siguiente carga se conserva en el Control Panel la dirección IP establecida previamente.

  Al transferir el proyecto se transfiere la dirección IP al panel de operador.
  La dirección IP del panel de operador se configura del siguiente modo:

  – Haga clic en el panel de operador.
  – Abra el editor "Configuración de dispositivos".
  – Haga clic en la interfaz Ethernet.
  – Asigne la dirección IP en la ventana de inspección en:
    "General > Interfaz PROFINET > Direcciones Ethernet"

- "Dirección"
  En el área "Dirección" se asigna la dirección IP del panel de operador.
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección", determine la dirección IP del módulo S7 conectado al panel de operador.
- "Slot de ampliación"
  Determina el número del slot de ampliación de la CPU que se debe direccionar.

- "Rack"
  Determina el número del rack de la CPU que se debe direccionar.
- "Funcionamiento cíclico"

  Nota

  El ajuste "Funcionamiento cíclico" no se puede configurar en el controlador SIMATIC S7 1200.

  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento.
  Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador.

Parámetros PROFIBUS

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red PROFIBUS.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección PROFIBUS del panel de operador. La dirección PROFIBUS debe ser unívoca en la red PROFIBUS.
- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro).
  En el S7-200 hay que configurar un panel de operador como maestro.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros de la red

En "Red" se ajustan los parámetros de la red PROFIBUS en la que está integrado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "DP", "Universal" o "Estándar" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección PROFIBUS real más alta. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Número de maestros"
  En "Número de maestros" se ajusta el número de maestros en la red PROFIBUS. Este dato es necesario para calcular correctamente los parámetros del bus.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se determina la dirección PROFIBUS del módulo S7 (CPU, FM o CP) conectado al panel de operador.
- "Funcionamiento cíclico"

  Nota

  El ajuste "Funcionamiento cíclico" no se puede configurar en el controlador SIMATIC S7 1200.

  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento. Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador. Este ajuste no es necesario para los SIMATIC S7-200.

Parámetros MPI

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red MPI.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección MPI del panel de operador. La dirección MPI debe ser unívoca en la red MPI.
- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro). Si sólo se han conectado esclavos en el panel de operador, hay que desactivar la función de seguridad "Único maestro del bus".
  En el S7-200 hay que configurar un panel de operador como maestro.

Parámetros de la red

En "Red" se ajustan los parámetros de la red MPI con la que está conectado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "MPI" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección MPI real más alta. Este ajuste debe ser el mismo en toda la red.
- "Número de maestros"
  Este dato no es necesario en MPI.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se determina la dirección MPI del módulo S7 (CPU, FM o CP) conectado al panel de operador.
- "Funcionamiento cíclico"
  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento. Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador. Este ajuste no es necesario para los SIMATIC S7-200.

Parámetros PPI

Parámetros del panel de operador

En "Panel de operador" se ajustan una sola vez los parámetros del panel de operador dentro de la red. El cambio afecta a todos los interlocutores.

- "Tipo"
  Define la conexión física utilizada.
- "Interfaz"
  En "Interfaz" se selecciona la interfaz del panel de operador que lo conecta a la red PPI.
- "Velocidad de transferencia"
  En "Velocidad de transferencia" se ajusta la velocidad para transferir los datos en la red. La velocidad de transferencia queda determinada por el panel de operador más lento conectado a la red. Este ajuste debe ser el mismo en toda la red.

  Nota

  Si en el OP 73 o en el OP 77A se ajusta una velocidad de transferencia de 1,5 Mbaudios, la dirección más alta de estación debe ser menor o igual a 63.

- "Dirección"
  En "Dirección" se ajusta la dirección PPI del panel de operador. La dirección PPI debe ser unívoca en la red PPI.
- "Punto de acceso"
  En "Punto de acceso" se define el punto de acceso que permite acceder al interlocutor.
- "Único maestro del bus"
  Desactiva una función de seguridad adicional para proteger de averías en el bus cuando se acopla el panel de operador a la red. Una estación pasiva (esclavo) sólo puede enviar datos cuando se lo pide una estación activa (maestro). Si sólo se han conectado esclavos en el panel de operador, hay que desactivar la función de seguridad "Único maestro del bus".
  En el S7-200 hay que configurar un panel de operador como maestro.

Parámetros de la red

En "Red" se ajustan los parámetros de la red en la que está integrado el panel de operador.

- "Perfil"
  En "Perfil" se selecciona el perfil de red que debe utilizarse. Ajuste "PPI" en "Perfil". Este ajuste debe ser el mismo en toda la red.
- "Dirección más alta de estación"
  En "Dirección más alta de estación" se ajusta la dirección más alta de estación. Dicha dirección debe ser mayor o igual a la dirección MPI real más alta. Este ajuste debe ser el mismo en toda la red.
- "Número de maestros"
  Ajuste el número de maestros que están en la red a "1".

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección" se ajusta la dirección PPI del módulo S7 (CPU) conectado al panel de operador.
- "Funcionamiento cíclico"
  Este parámetro no es necesario para una comunicación vía PPI.

10.8.9.4 Intercambio de datos

Intercambio de datos mediante punteros de área

Generalidades sobre los punteros de área

Introducción

A través de un puntero de área se accede a un rango de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación.

Al evaluar los datos almacenados, el controlador y el panel de operador ejecutan acciones fijamente definidas.

Configurar punteros de área

Antes de utilizar un puntero de área, actívelos en "Conexiones > Punteros de área". Seguidamente, parametrícelos.

Encontrará más información sobre la configuración de punteros de área en:

Configurar un puntero de área (Página 8783)

Puntero de área "Número de imagen"

Función

Los paneles de operador depositan en el puntero de área "Número de imagen" información acerca de la imagen visualizada en el panel.

Ello permite transferir al controlador información acerca del contenido actual de la imagenen el panel de operador. En el controlador se pueden disparar determinadas reacciones, p. ej. la llamada de otra imagen.

Utilización

Antes de poder utilizar el puntero de área "Número de imagen", es necesario configurarlo y activarlo bajo "Comunicación > Conexiones". El puntero de área "Número de imagen" se puede crear únicamente en **un** controlador y depositarse allí **una sola vez**.

El número de imagen se transfiere siempre al controlador cuando se activa una nueva imagen o cuando el foco cambia de un objeto de imagen a otro dentro de una imagen.

Estructura

Este puntero de área es un área de datos de la memoria del controlador con una longitud fija de 5 palabras.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1ª palabra	Tipo de imagen actual															
2ª palabra	Número de imagen actual															
3ª palabra	Reservado															
4ª palabra	Número de campo actual															
5ª palabra	Reservada															

- Tipo de imagen actual
  "1" = imagen raíz
  "4" = ventana permanente
- Número de imagen actual
  1 a 32767
- Número de campo actual
  1 a 32767

Puntero de área "Fecha/hora"

Función

Este puntero de área sirve para transferir la fecha y la hora del panel de operador al controlador.

El controlador escribe la petición de control "41" en la bandeja de peticiones.

Al evaluar la petición de control, el panel de operador escribe su fecha y hora actuales en el área de datos configurada en el puntero de área "Fecha/hora". Todos los datos se expresan en formato BCD.

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

Nota

Si ha configurado el puntero de área "Fecha/hora", no podrá utilizar el puntero de área "Fecha/hora PLC".

Nota

Si utiliza el puntero de área "Fecha/Hora", no es posible el direccionamiento simbólico.

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

El área de datos "Fecha/hora" tiene la siguiente estructura:

Palabra de datos	Byte más significativo								Byte menos significativo								
	7							0	7							0	
n+0	Reservada								Hora (0 a 23)								Hora
n+1	Minuto (0 a 59)								Segundo (0 a 59)								
n+2	Reservada								Reservada								
n+3	Reservada								Día de la semana (1 a 7, 1=domingo)								Fecha
n+4	Día (1 a 31)								Mes (1 a 12)								
n+5	Año (80 a 99/0 a 29)								Reservada								

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Puntero de área "Fecha/hora PLC"

Función

Este puntero de área sirve para transferir la fecha y la hora del controlador al panel de operador. Este puntero de área se utiliza si el controlador es el maestro que determina la hora.

El controlador carga el área de datos del puntero de área. Todos los datos se expresan en formato BCD.

Según el ciclo de adquisición configurado, el panel de operador lee los datos cíclicamente y se sincroniza.

Nota

No configure un ciclo de adquisición demasiado breve para el puntero de área "Fecha/hora", puesto que ello afecta el rendimiento del panel de operador.
Recomendación: Ciclo de adquisición = 1 minuto (si el proceso lo permite).

"Fecha/hora PLC" es un puntero de área global y solo se puede configurar una vez en cada proyecto.

Nota

Si ha configurado el puntero de área "Fecha/hora PLC", no podrá utilizar el puntero de área "Fecha/hora".

El área de datos "Fecha/hora" tiene la siguiente estructura:

Formato DATE_AND_TIME (codificado en BCD)

Palabra de datos	Byte más significativo			Byte menos significativo		
	7	. . . . .	0	7	. . . . .	0
n+0	Año (80 a 99/0 a 29)			Mes (1 a 12)		
n+1	Día (1 a 31)			Hora (0 a 23)		
n+2	Minuto (0 a 59)			Segundo (0 a 59)		
n+3	Reservada			Reservada		Día de la semana (1 a 7, 1=domingo)
n+4 [1)]	Reservada			Reservada		
n+5 [1)]	Reservada			Reservada		

1) Ambas palabras de datos deben existir en el área de datos para garantizar la concordancia con el formato de datos de WinCC flexible y evitar la lectura de información errónea.

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Puntero de área "Coordinación"

Función

El puntero de área "Coordinación" sirve para realizar las funciones siguientes:

- Detectar en el programa de control el arranque del panel de operador
- Detectar en el programa de control el modo de operación actual del panel de operador
- Detectar en el programa de control si el panel de operador está dispuesto para comunicarse

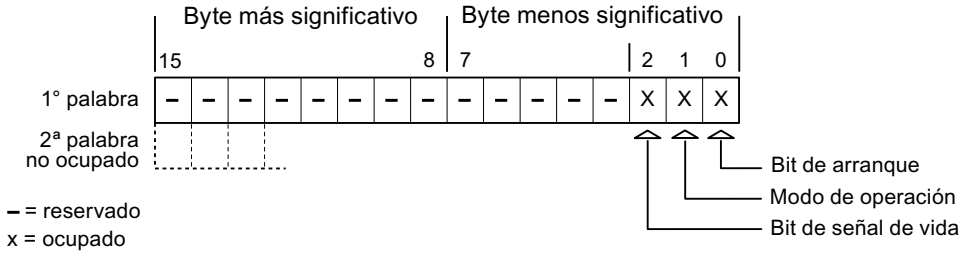
El puntero de área "Coordinación" tiene una longitud estándar de una palabra y no se puede modificar.

Utilización

Nota

Cada vez que el panel de operador actualiza el puntero de área, se escribe siempre toda el área de coordinación.
Por tanto, el programa de control no puede efectuar cambios en el área de coordinación.

Ocupación de los bits en el puntero de área "Coordinación"



Bit de arranque

Durante el arranque, el panel de operador pone el bit de arranque a "0" por breve tiempo. Después del arranque, el bit se pone a "1" de forma permanente.

Modo de operación

Cuando el usuario conmuta el panel de operador a modo "offline", el bit de modo de operación de pone a "1". En el servicio normal del panel de operador, el estado del bit de modo de operación es "0". Consultando este bit en el programa de control es posible averiguar el modo de operación actual del panel de operador.

Bit de señal de vida

El panel de operador invierte el estado del bit de señal de vida en intervalos de aprox. 1 segundo. Consultando este bit en el programa de control es posible averiguar si todavía existe una conexión con el panel de operador.

Puntero de área "Identificación del proyecto"

Función

Al iniciar runtime puede comprobarse si el panel de operador está conectado al controlador correcto. Esta comprobación es importante si se utilizan varios paneles de operador.

Para ello, el panel de operador compara un valor almacenado en el controlador con el valor indicado en la configuración. Así puede garantizarse la compatibilidad de los datos de configuración con los del programa de control. Si no hay compatibilidad, se muestra un aviso de sistema en el panel de operador y se detiene el runtime.

Utilización

Nota

Las conexiones HMI no pueden pasar a "online"

La conexión HMI en la que se utiliza el puntero de área "Identificador del proyecto" debe pasar a "online".

Para utilizar este puntero de área, especifique en la configuración lo siguiente:

- Versión de la configuración. El valor puede estar comprendido entre 1 y 255.
  La versión se indica en el editor "Configuración de runtime > General" en el área "Identificación".
- Dirección de datos del valor para la versión que está almacenada en el controlador:
  La dirección de datos se indica en el editor "Comunicación > Conexiones" en "Dirección".

Fallo de una conexión

Al fallar la conexión con un panel de operador en el que esté configurado el puntero de área "Identificador del proyecto", se desactivarán también todas las demás conexiones del dispositivo.

Para este comportamiento se deben cumplir los requisitos siguientes:

- En el proyecto se han configurado varias conexiones.
- En por lo menos una conexión se está utilizando el puntero de área "Identificador del proyecto".

Las causas siguientes pueden conmutar las conexiones al estado "offline":

- No se puede acceder al controlador.
- La conexión se ha conmutado a "offline" en el sistema de ingeniería.

Puntero de área "Petición de control"

Función

Desde la bandeja de peticiones de control se pueden lanzar peticiones al panel de operador y con ello disparar acciones en el panel. A estas funciones pertenecen p. ej.:

- Mostrar imagen
- Ajustar la fecha y hora

Estructura de datos

En la primera palabra de la bandeja de peticiones de control figura el número de la petición. Dependiendo de la petición en cuestión, pueden transferirse hasta 3 parámetros como máximo.

Palabra	Byte más significativo	Byte menos significativo
n+0	0	Número de petición
n+1	Parámetro 1	
n+2	Parámetro 2	
n+3	Parámetro 3	

Si la primera palabra de la bandeja de peticiones de control es diferente de 0, el panel de operador evaluará la petición. Por este motivo hay que introducir primero los parámetros en la bandeja de peticiones de control y sólo entonces el número de la petición.

Cuando el panel de operador haya aceptado la petición de control, la primera palabra se pondrá de nuevo a 0. Por lo general, la ejecución de la petición de control no se habrá completado todavía.

Peticiones de control

A continuación se indican todas las peticiones de control y sus respectivos parámetros. La columna "N.º" contiene el número de petición de control. Por regla general, las peticiones de control sólo pueden ser disparadas por el controlador si el panel de operador está en modo online.

N.º	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 1	Byte izquierdo: - Byte derecho: horas (0-23)
	Parámetro 2	Byte izquierdo: minutos (0-59) Byte derecho: segundos (0-59)
	Parámetro 3	-
15	Ajustar la fecha (codificada en BCD) [3)]	

N.º	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 1	Byte izquierdo: - Byte derecho: día de la semana (1-7: domingo-sábado)
	Parámetro 2	Byte izquierdo: Día (1-31) Byte derecho: mes (1-12)
	Parámetro 3	Byte izquierdo: año
23	Iniciar la sesión	
	Inicia la sesión del usuario "PLC User" y el número de grupo transferido en el parámetro 1 en el panel de operador. Para poder iniciar la sesión es necesario que el número de grupo transferido exista en el proyecto.	
	Parámetro 1	Número de grupo 1 - 255
	Parámetro 2, 3	-
24	Cerrar la sesión	
	Cierra la sesión del usuario actual. (Esta función equivale a la función de sistema "CerrarSesión")	
	Parámetro 1, 2, 3	-
40	Transferir fecha/hora al controlador	
	(En formato S7 DATE_AND_TIME) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
41	Transferir fecha/hora al controlador	
	(En formato OP/MP) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
46	Actualizar variable	
	Hace que el panel de operador lea el valor actual de la variable del controlador cuyo identificador de actualización coincide con el valor transferido en el parámetro 1. (Esta función equivale a la función de sistema "ActualizarVariable")	
	Parámetro 1	1 - 100
49	Borrar búfer de avisos	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Warnings" del búfer de avisos.	
	Parámetro 1, 2, 3	-
50	Borrar búfer de avisos	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Errors" del búfer de avisos.	
	Parámetro 1, 2, 3	-
51	Selección de imagen [2)]	
	Parámetro 1	Número de imagen
	Parámetro 2	-
	Parámetro 3	Número de campo
69	Leer registro del controlador [1)]	
	Parámetro 1	Número de receta (1-999)

N.º	Función	
14	Ajustar la hora (codificada en BCD)	
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	0: No sobrescribir el registro existente 1: Sobrescribir el registro existente
70	Escribir registro en el controlador [1]	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	-

[1]	Sólo en paneles que soporten recetas
[2]	En los paneles de operador OP 73, OP 77A y TP 177A la petición de control "Selección de imagen" se ejecuta incluso con el teclado de pantalla abierto.
[3]	En el panel de operador KTP 600 BASIC PN se ignora el día de la semana.

Puntero de área "Registro"

Puntero de área "Registro"

Función

Al transferir registros entre el panel de operador y el autómata, ambos interlocutores acceden alternativamente a las áreas de comunicación conjuntas del autómata.

Tipos de transferencia

Existen dos posibilidades para transferir registros entre el panel de operador y el autómata, a saber:

- Transferencia sin sincronización
- Transferencia con sincronización a través de la bandeja de datos

Los registros se transfieren siempre directamente. Por tanto, los valores de variables se leen o escriben directamente de/en la dirección configurada para la variable, sin desviarse al portapapeles.

Iniciar la transferencia de registros

Para iniciar la transferencia existen tres posibilidades:

- Inicio desde la vista de recetas
- Tareas de control
  El autómata también puede disparar la transferencia de registros.
- Activar funciones configuradas

Si la transferencia de registros se dispara mediante una función configurada o una orden de control, la vista de recetas seguirá pudiéndose controlar sin impedimentos desde el panel de operador. Los registros se transfieren en segundo plano.

Sin embargo, no es posible procesar simultáneamente varias peticiones de transferencia. En este caso, el panel de operador deniega una segunda transferencia con un aviso de sistema.

Ejecución de la transferencia al iniciarla desde el visor de recetas

Leer del controlador por operación en el visor de recetas

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta que ha de ser leído, así como el estado "Transferencia en curso" en la bandeja de datos y pone el número de registro a 0.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del controlador y los muestra en el visor de recetas. En las recetas con variables sincronizadas, los valores del controlador también se escriben en las variables.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Escribir en el controlador por operación en el visor de recetas

Paso	Acción	
	Verificación: ¿Palabra de estado = 0?	
1	Sí	No
	El panel de operador introduce el número de receta y el número de registro que deben escribirse, así como como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
2	El panel de operador escribe los valores actuales en el controlador. En las recetas con variables sincronizadas los valores modificados primero se sincronizan entre el visor de recetas y las variables, y luego se escriben en el controlador.	
3	El panel de operador activa el estado "Transferencia concluida".	
4	Dado el caso, el programa de control puede evaluar entonces los datos transferidos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Nota

La palabra de estado sólo se puede activar desde el panel de operador. El controlador sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el controlador sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos es "Transferencia concluida".
- El estado en la bandeja de datos es "Transferencia concluida con errores".

Ejecución de la transferencia en caso de petición de control

Tanto el panel de operador como el controlador pueden iniciar la transferencia de registros entre sí.

Para este tipo de transferencia se dispone de las peticiones de controlador n.° 69 y n.° 70.

N.° 69: Leer registro del controlador ("SPS → DAT")

La petición de controlador n.° 69 transfiere los registros del controlador al panel de operador. La petición del controlador está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	69
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	No sobrescribir el registro existente: 0 Sobrescribir el registro existente: 1	

N.° 70: Escribir registro en el controlador ("DAT → SPS")

La petición del controlador n.° 70 transfiere los registros del panel de operador al controlador. La petición del controlador está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	70
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	—	

Procedimiento al leer del controlador mediante petición de controlador "SPS → DAT" (n.° 69)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la tarea, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador lee los valores del controlador y los guarda en el registro que se haya indicado en la petición del controlador.	
4	• Si en la petición se ha seleccionado "Sobrescribir", un registro existente se sobrescribe sin previa consulta. El panel de operador activa el estado "Transferencia concluida". • Si en la petición se ha seleccionado "No sobrescribir" y ya existe el registro, el panel de operador cancelará la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Procedimiento al escribir en el controlador mediante la petición de controlador "DAT → SPS" (n.° 70)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la tarea, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador toma los valores del registro indicado en la petición del soporte de datos y los escribe en el controlador.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Ejecución de la transferencia al lanzarla mediante una función configurada

Leer del controlador mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.

Paso	Acción	
3	El panel de operador lee los valores del controlador y los almacena en el registro que se haya indicado en la función.	
4	• Si en la función "Sobrescribir" se ha seleccionado "Sí", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia concluida". • Si en la función "Sobrescribir" se ha seleccionado "No" y el registro ya existe, el panel de operador interrumpirá la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Escribir en el controlador mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de control recoge los valores del registro indicado en la función del soporte de datos y los escribe en el controlador.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Posibles causas de error en la transferencia de registros

Causas de error posibles

Si la transferencia de registros termina con errores, puede deberse a las siguientes causas:

- No se ha configurado ninguna dirección de variable en el controlador
- No es posible sobrescribir registros

- El número de receta no existe
- El número de registro no existe

Nota

La palabra de estado sólo se puede activar desde el panel de operador. El controlador sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el controlador sólo debe efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos es "Transferencia concluida".
- El estado en la bandeja de datos es "Transferencia concluida con errores".

Reacción a una interrupción debida a un error

El panel de operador reacciona a una interrupción de la transferencia de registros debida a un error de la siguiente forma:

- Inicio por operación en el visor de recetas:
  Indicaciones en la barra de estado del visor de recetas y salida de avisos de sistema
- Inicio mediante una función
  Emisión de avisos de sistema
- Inicio mediante una petición de controlador
  No hay respuesta en el panel de operador

Independientemente de ello, el usuario puede evaluar el estado de la transferencia consultando la palabra de estado en la bandeja de datos.

Transferencia sin sincronización

En la transferencia asíncrona de registros entre el panel de operador y el autómata no se coordinan las áreas de datos utilizadas conjuntamente. Por ello, no es necesario crear un área de datos en la configuración.

La transferencia asíncrona de registros resulta útil p. ej. en los casos siguientes:

- Condicionado por el sistema, puede descartarse que los interlocutores sobrescriban los datos de forma incontrolada.
- El autómata no necesita información sobre el número de receta o el de registro.
- La transferencia de registros es iniciada por el usuario desde el panel de operador.

Leer valores

Al iniciar la transferencia de lectura se leen los valores de las direcciones de control y se transfieren al panel de operador.

- Inicio por el usuario desde la vista de recetas:
  Los valores se cargan en el panel de operador. Éstos se pueden procesar posteriormente en el panel de operador, p. ej. modificar o guardar los valores, etc.
- Inicio mediante una función u orden de control:
  Los valores se almacenan inmediatamente en el soporte de datos.

Escribir valores

Al iniciar la transferencia para escritura se escriben los valores en las direcciones de control.

- Inicio por el usuario desde la vista de recetas:
  Los valores actuales se escriben en el autómata.
- Inicio mediante una función u orden de control:
  Los valores del soporte de datos se escriben en el autómata.

Transferencia con sincronización

En la transferencia sincronizada, ambos interlocutores activan determinados bits de estado en el área de datos utilizada conjuntamente. De este modo se impide que los datos se sobrescriban unos a otros de forma incontrolada en el programa de control.

Aplicación

La transferencia síncrona de registros resulta útil p. ej. en los casos siguientes:

- El controlador es el "interlocutor activo" en la transferencia de registros.
- En el controlador se evalúa información sobre los números de receta y de registro.
- La transferencia de registros se dispara mediante una petición de controlador.

Requisitos

Para que los registros puedan transferirse de forma sincronizada entre el panel de operador y el controlador, en el proceso de configuración deben cumplirse las siguientes condiciones:

- Se ha configurado un puntero de área: Editor "Comunicación > Conexiones" bajo "Punteros de área".
- En la receta está indicado el controlador con el que el panel de operador sincroniza la transferencia de registros:
  Editor "Recetas" en el área "General > Sincronización > Configuración" de la ventana de inspección, y selección "Transferencia de datos coordinada" activada.

Estructura del área de datos

El área de datos tiene una longitud fija de 5 palabras. El área de datos está estructurada de la siguiente manera:

	15		0
1. palabra	Número de receta actual (1-999)		
2. palabra	Número de registro actual (0-65.535)		
3. palabra	Reservada		
4. palabra	Estado (0, 2, 4, 12)		
5. palabra	Reservada		

- Estado
  La palabra de estado (palabra 4) acepta los siguientes valores:

Valor		Significado
Decimal	Binario	
0	0000 0000	Transferencia permitida, bandeja de datos libre
2	0000 0010	Transferencia en curso.
4	0000 0100	Transferencia finalizada sin fallos
12	0000 1100	Transferencia finalizada con fallos

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos

Para configurar avisos tales como avisos de operador, alarmas y acuses se requieren varios pasos.

- Paso 1: Crear variables
- Paso 2: Configurar avisos
- Paso 3: configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- tipos de datos de la variable
- direccionamiento de la variable
- modo de contaje de las posiciones de bit

Tipos de datos

Para conexiones con drivers de comunicación SIMATIC se soportan los tipos de datos siguientes:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
Controladores SIMATIC S7.	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

Modo de contaje de las posiciones de bit

Para conexiones con drivers de comunicación SIMATIC rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte 0								Byte 1							
	Byte más significativo								Byte menos significativo							
En controladores SIMATIC S7	7							0	7							0
Configure en WinCC:	15							8	7							0

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

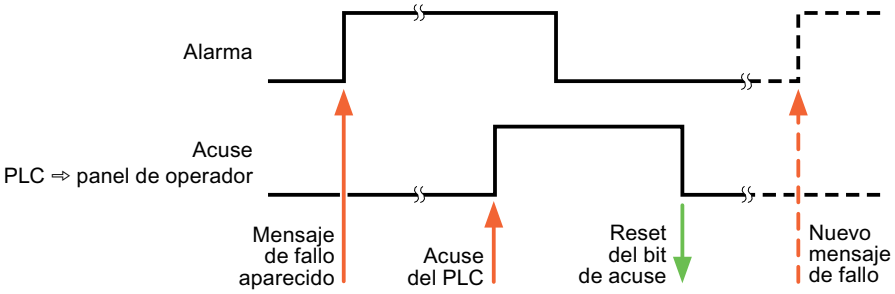
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el

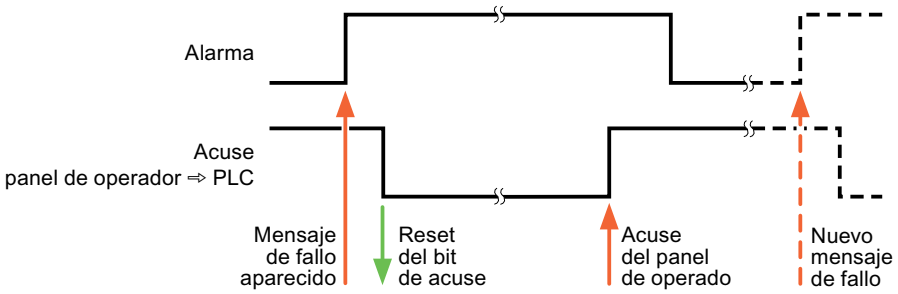
controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



10.8.9.5 Características de rendimiento de la comunicación

Tipos de datos admisibles para SIMATIC S7 200

Tipos de datos admisibles para conexiones con SIMATIC S7 200

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Longitud
Bool	1 bit
Byte	1 byte
Char	1 byte
Word	2 bytes
Int	2 bytes
DWord	4 bytes
DInt	4 bytes
Real	4 bytes
StringChar	--
Temporizador	2 bytes
Matriz	--

Nota

Interrupción de la conexión en la red PPI

Si se utilizan matrices en la configuración, con un tamaño de matriz de aprox. 1.000 bytes puede producirse una interrupción de la conexión.

Utilice matrices más pequeñas en la configuración.

10.8.10 Comunicación con SIMATIC LOGO!

10.8.10.1 Comunicación con SIMATIC LOGO!

Introducción

Este apartado describe la comunicación entre un panel de operador y el controlador SIMATIC LOGO!.

Se pueden configurar los siguientes canales de comunicación para el controlador SIMATIC LOGO!:

- PROFINET
- Ethernet

Conexión HMI para la comunicación

Las conexiones entre un panel de operador y un SIMATIC LOGO! se configuran en el editor "Conexiones" del panel de operador.

Intercambio de datos

El intercambio de datos con el controlador SIMATIC LOGO! es posible a través de variables.

No es posible el intercambio de datos a través del puntero de área.

10.8.10.2 Crear una conexión con SIMATIC LOGO!

Introducción

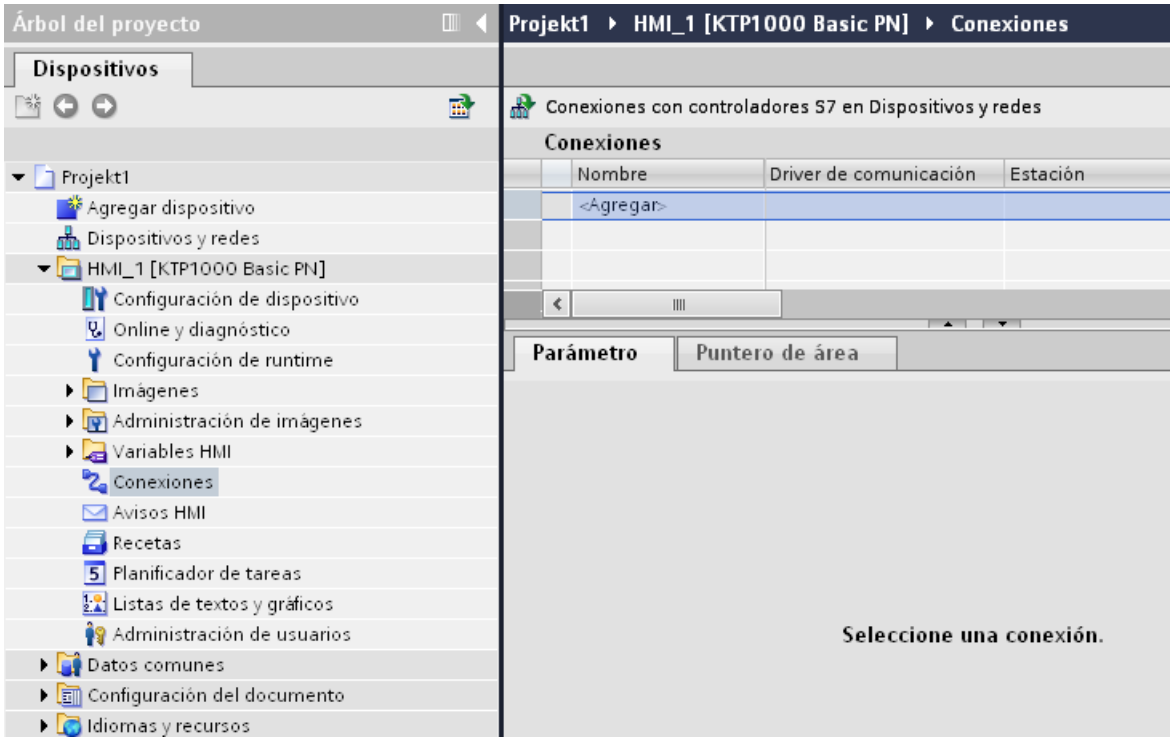
La conexión con el controlador SIMATIC LOGO! se configura en el editor "Conexiones" del panel de operador. Las interfaces reciben nombres distintos en función del panel de operador.

Requisitos

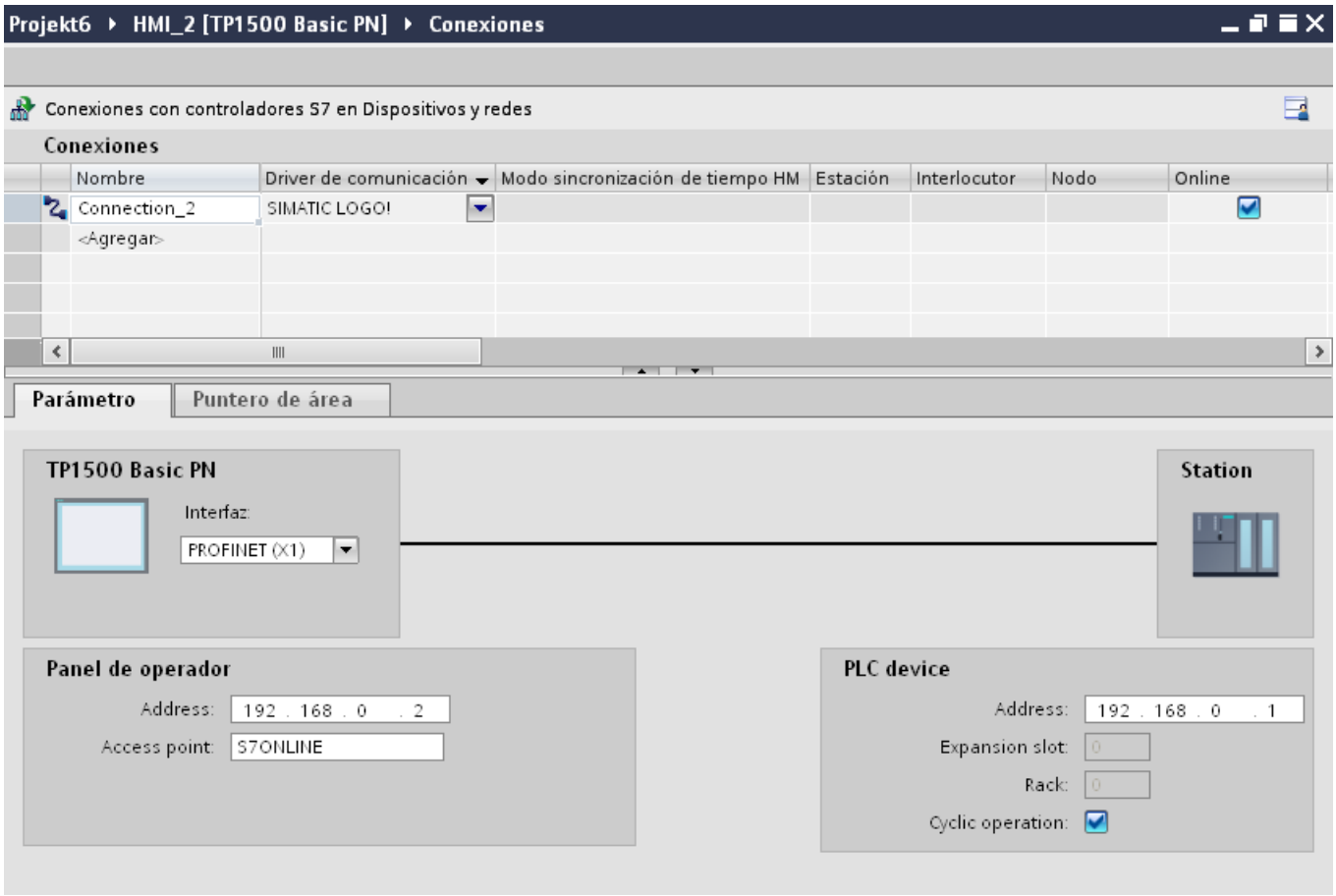
- Hay un proyecto abierto.
- Se ha creado un panel de operador.

Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".
3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el driver "SIMATIC LOGO!" en la columna "Driver de comunicación".
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.



Encontrará más información al respecto en el capítulo "Auto-Hotspot".

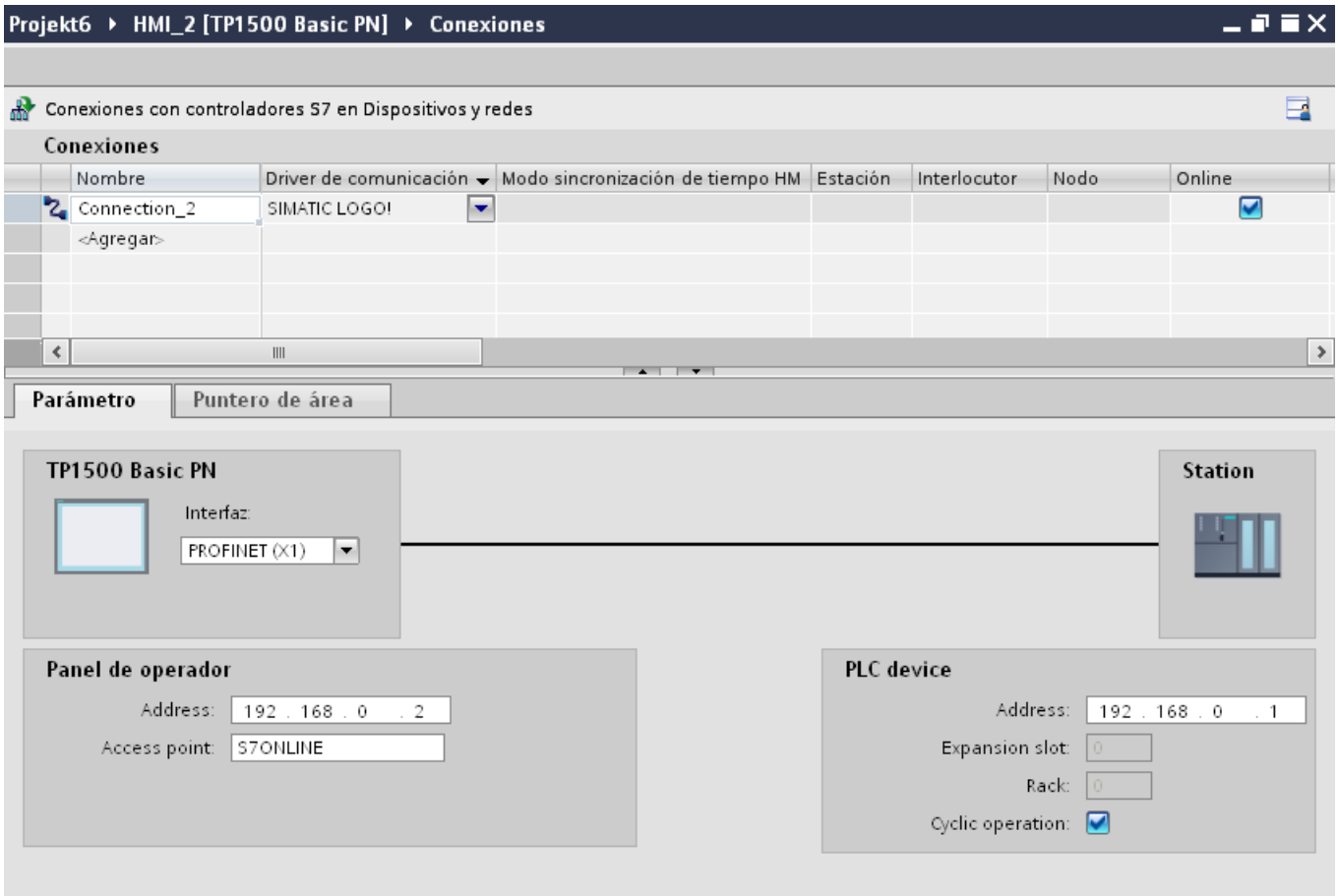
10.8.10.3 Parámetros de la conexión

Parámetros de la conexión

Parámetros que deben ajustarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador" y "Controlador", según sea la interfaz utilizada.


Projekt6 ▸ HMI_2 [TP1500 Basic PN] ▸ Conexiones
Conexiones con controladores S7 en Dispositivos y redes
Conexiones
Nombre
Driver de comunicación
Modo sincronización de tiempo HM
Estación
Interlocutor
Nodo
Online
Connection_2
SIMATIC LOGO!
<Agregar>
Parámetro
Puntero de área
TP1500 Basic PN
Interfaz:
PROFINET (X1)
Station
Panel de operador
Address: 192 . 168 . 0 . 2
Access point: S7ONLINE
PLC device
Address: 192 . 168 . 0 . 1
Expansion slot: 0
Rack: 0
Cyclic operation:

Parámetros Ethernet

Parámetros del panel de operador

Ajuste en "Panel de operador" los parámetros para conectar el panel de operador a la red. Los cambios no se transfieren automáticamente al panel de operador. La configuración se debe modificar en el panel de control del panel de operador.

- "Interfaz"
  Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC.

  Nota

  Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

  Si activa "Obtener dirección IP por otra vía", en la siguiente carga se conserva en el Control Panel la dirección IP establecida previamente.

  Al transferir el proyecto se transfiere la dirección IP al panel de operador.
  La dirección IP del panel de operador se configura del siguiente modo:

  – Haga clic en el panel de operador.
  – Abra el editor "Configuración de dispositivos".
  – Haga clic en la interfaz Ethernet.
  – Asigne la dirección IP en la ventana de inspección en:
    "General > Interfaz PROFINET > Direcciones Ethernet"

- "Dirección"
  En el área "Dirección" se asigna la dirección IP del panel de operador.
  Al transferir el proyecto de WinCC al panel de operador, esta dirección IP se configura directamente en el panel de operador.
- "Punto de acceso"
  El punto de acceso define un nombre de dispositivo lógico que permite acceder al interlocutor.

Parámetros del controlador

En "Controlador" se direcciona el módulo S7 que debe intercambiar datos con el panel de operador. Asigne un nombre a la conexión de cada interlocutor.

- "Dirección"
  En "Dirección", determine la dirección IP del módulo S7 conectado al panel de operador.
- "Slot de ampliación"
  Determina el número del slot de ampliación de la CPU que se debe direccionar.

- "Rack"
  Determina el número del rack de la CPU que se debe direccionar.
- "Funcionamiento cíclico"

  Nota

  El ajuste "Funcionamiento cíclico" no se puede configurar en el controlador SIMATIC S7 1200.

  Si se activa el funcionamiento cíclico, el controlador optimizará la transferencia de datos entre él mismo y el panel de operador. Ello permite aumentar el rendimiento.
  Desactive el funcionamiento cíclico si la comunicación se debe establecer con varios paneles de operador.

Funcionamiento cíclico

Uso de la selección "Funcionamiento cíclico"

Si está activado el "Funcionamiento cíclico", al principio de la comunicación el panel de operador enviará un telegrama al controlador con la información de que determinadas variables se requieren constantemente.

A partir de entonces el controlador enviará los datos siempre en el mismo ciclo. De este modo el panel de operador evita tener que pedir los datos una y otra vez.

Si el funcionamiento cíclico está desactivado, el panel de operador enviará una petición especial para cada información que necesite.

Otras características:

- El funcionamiento cíclico da soporte al panel de operador en la transferencia de datos reduciendo su carga. Para reducir la carga del panel de operador se utilizan los recursos existentes en el controlador.
- El controlador soporta sólo un número determinado de servicios cíclicos. Cuando el controlador ya no dispone de recursos para los servicios cíclicos, entonces el panel de operador se vuelve a hacer cargo de la operación.
- Si el controlador no soporta el funcionamiento cíclico, entonces el panel de operador se encarga de formar el ciclo.
- Las variables de imagen no se incorporan al funcionamiento cíclico.
- El funcionamiento cíclico se configura únicamente al reiniciar el runtime.
- Cuando está activado el funcionamiento cíclico, dependiendo del controlador, se transfiere más de una petición del panel de operador al controlador.
- Cuando está desactivado el funcionamiento cíclico, se transfiere sólo una petición del panel de operador al controlador.

10.8.10.4 Intercambio de datos

Curvas

Aspectos generales de las curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. Dependiendo de la configuración, la lectura del valor se efectúa disparada por tiempo o por bit.

Encontrará más información al respecto en:

Auto-Hotspot

Nota

En los Basic Panels el valor se lee disparado por tiempo.

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración. Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Curvas disparadas por bit

Cuando se activa un bit de disparo en la variable Solicitud de curva, el panel de operador lee un valor de curva o un búfer de curvas completo. Este ajuste se define en la configuración. Las curvas disparadas por bit se utilizan por lo general para representar valores que se modifican rápidamente. Un ejemplo de ello es la presión de inyección en la fabricación de piezas de plástico.

Para activar curvas disparadas por bit, deben crearse durante la configuración variables externas en el editor "Variables HMI" y vincularse con áreas de curvas. El panel de operador y el controlador se comunicarán entonces a través de dichas áreas de curvas.

Se dispone de las siguientes áreas para las curvas:

- Área de solicitud de curvas
- Área de transferencia de curvas 1
- Área de transferencia de curvas 2 (sólo se requiere para el búfer alternativo)

Solicitud de curvas y transferencia de curvas

Área de solicitud de curvas

Si en el panel de operador se abre una imagen con una o varias curvas, el panel de operador activará los bits correspondientes en el área de solicitud de curvas. Una vez deseleccionada la imagen, el panel de operador desactiva los correspondientes bits en el área de solicitud de curvas.

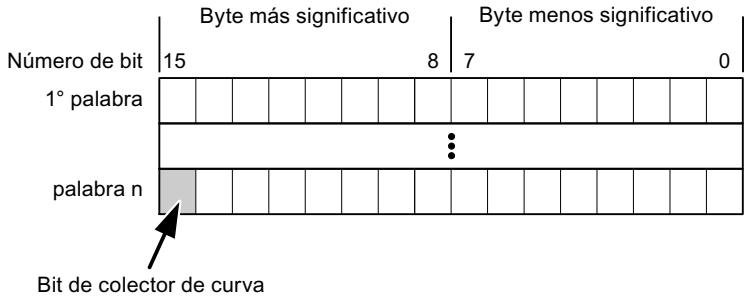
A través del área de solicitud de curvas es posible evaluar en el controlador qué curva se está representando actualmente en el panel de operador. Las curvas también se pueden disparar sin necesidad de evaluar el área de solicitud de curvas.

Área de transferencia de curvas 1

Esta área sirve para disparar curvas. En el programa de control se debe activar el bit asignado a la curva en el área de transferencia de curvas así como el bit de curvas de grupo. El bit de curvas de grupo es el último bit del área de transferencia de curvas.

El panel de operador detecta el disparo. El panel de operador lee un valor, o bien el búfer completo del controlador. A continuación, éste activa el bit de curvas y desactiva el bit de curvas de grupo.

La siguiente imagen muestra la estructura de un área de transferencia de curvas.



Mientras no se desactive el bit de curvas de grupo, el programa de control no podrá modificar el área de transferencia de curvas.

Área de transferencia de curvas 2

El área de transferencia de curvas 2 se requiere para las curvas que se configuran con el búfer alternativo. El área de transferencia de curvas 2 está estructurada del mismo modo que el área de transferencia de curvas 1.

Búfer alternativo

El búfer alternativo es un segundo búfer para la misma curva, que puede crearse en la configuración.

Mientras el panel de operador lee los valores del búfer 1, el controlador escribe en el búfer 2. Mientras el panel de operador lee el búfer 2, el controlador escribe en el búfer 1. Así se impide

que se sobrescriban los valores de curva del controlador mientras el panel de operador lee la curva.

Tipos de datos admisibles para curvas

Para SIMATIC S7

Asigne un bit a cada curva en la configuración. Se admiten las variables del tipo de datos "Word" o "Int" y variables de matriz del tipo de datos "Word" o "Int".

Avisos

Configurar avisos

Configurar avisos

Para configurar avisos tales como avisos de operador, alarmas y acuses se requieren varios pasos.

- Paso 1: Crear variables
- Paso 2: Configurar avisos
- Paso 3: configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Auto-Hotspot

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- tipos de datos de la variable
- direccionamiento de la variable
- modo de contaje de las posiciones de bit

Tipos de datos

Para conexiones con drivers de comunicación SIMATIC se soportan los tipos de datos siguientes:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
Controladores SIMATIC S7.	WORD, INT	BYTE, CHAR, WORD, INT, DWORD, DINT, REAL, TIMER

Modo de contaje de las posiciones de bit

Para conexiones con drivers de comunicación SIMATIC rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte 0								Byte 1							
	Byte más significativo								Byte menos significativo							
En controladores SIMATIC S7	7							0	7							0
Configure en WinCC:	15							8	7							0

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

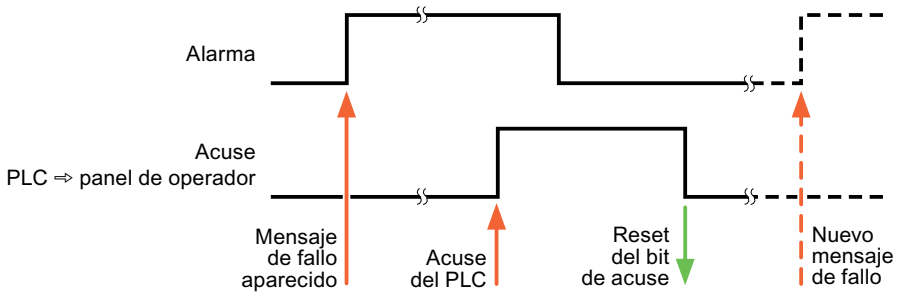
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

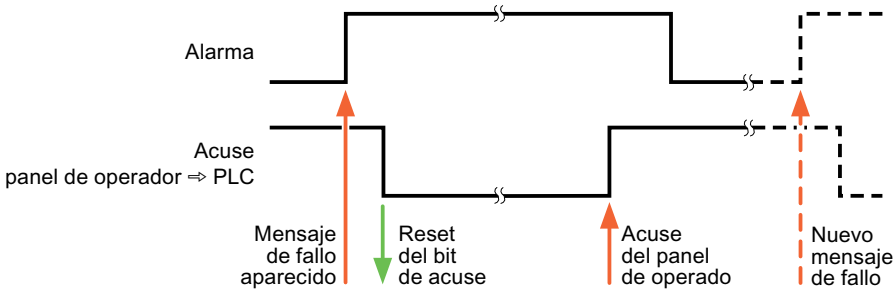
Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



10.8.10.5 Características de rendimiento de la comunicación

Tipos de datos admisibles para SIMATIC LOGO!

Tipos de datos admisibles para conexiones con SIMATIC LOGO!

Tipo de datos	Longitud
Bool	1 bit
Byte	1 byte

Tipo de datos	Longitud
Int	2 bytes
DInt	4 bytes
Word	2 bytes
DWord	4 bytes
Array	--

10.8.11 Comunicación con otros controladores

10.8.11.1 Comunicación con otros controladores

Introducción

La comunicación con otros controladores es la comunicación con controladores que no forman parte de la familia SIMATIC.

Estos controladores disponen de protocolos internos propios para el intercambio de datos. Los protocolos se registran en WinCC como drivers de comunicación.

Drivers de comunicación

WinCC soporta los siguientes drivers de comunicación (ya vienen instalados):

- Allen-Bradley
  - Allen-Bradley EtherNet/IP
  - Allen-Bradley DF1
- Mitsubishi
  - Mitsubishi MC TCP/IP
  - Mitsubishi FX
- Modicon Modbus
  - Modicon Modbus TCP/IP
  - Modicon Modbus RTU
- Omron
  - Omron Host Link

Drivers de comunicación en WinCC RT Professional

RT Professional soporta los siguientes drivers de comunicación:

- Allen-Bradley
  - Allen-Bradley EtherNet/IP
- Mitsubishi
  - Mitsubishi MC TCP/IP
- Modicon Modbus
  - Modicon Modbus TCP

Conexiones entre paneles de operador y otros controladores

Las conexiones entre paneles de operador y otros controladores se configuran en el editor "Conexiones" del panel de operador. Dichas conexiones forman parte de las no integradas.

10.8.11.2 Particularidades de la configuración

Particularidades del intercambio de datos

Al configurar conexiones con otros controladores existen particularidades respecto de la configuración de conexiones integradas.

Tenga en cuenta las particularidades siguientes durante la configuración:

- Direccionamiento de variables
- Tipos de datos admisibles
- Particularidades de la configuración de punteros de área
- Particularidades de la configuración de avisos
- Particularidades de la configuración de curvas

Encontrará más información sobre las particularidades de la configuración en el capítulo "Intercambio de datos" del driver de comunicación correspondiente.

10.8.11.3 Drivers de comunicación

Allen-Bradley

Drivers de comunicación Allen-Bradley

Introducción

Este apartado describe la comunicación entre un panel de operador y un controlador que utilizan drivers de comunicación Allen-Bradley.

Se soportan los siguientes drivers de comunicación:

- Allen-Bradley EtherNet/IP
- Allen-Bradley DF1

Intercambio de datos

El intercambio de datos se realiza mediante variables o punteros de área.

- Variables
  El controlador y el panel de operador intercambian los datos a través de valores de proceso. Configure variables que señalen a direcciones en el controlador. El panel de operador lee y muestra el valor de la dirección indicada. El operador también puede introducir un valor en el panel. Dicho valor se escribe luego en la dirección en el controlador.
- Punteros de área
  Los punteros de área sirven para intercambiar datos especiales y se configuran únicamente si se utiliza este tipo de datos.

Allen-Bradley EtherNet/IP

Configurar una conexión mediante Allen-Bradley EtherNet/IP

Introducción

Una conexión con un controlador que tenga un driver de comunicación Allen-Bradley EtherNet/IP se configura en el editor "Conexiones" del panel de operador.

Las interfaces Ethernet reciben nombres distintos en función del panel de operador.

Ejemplo: la interfaz PROFINET corresponde a la interfaz Ethernet

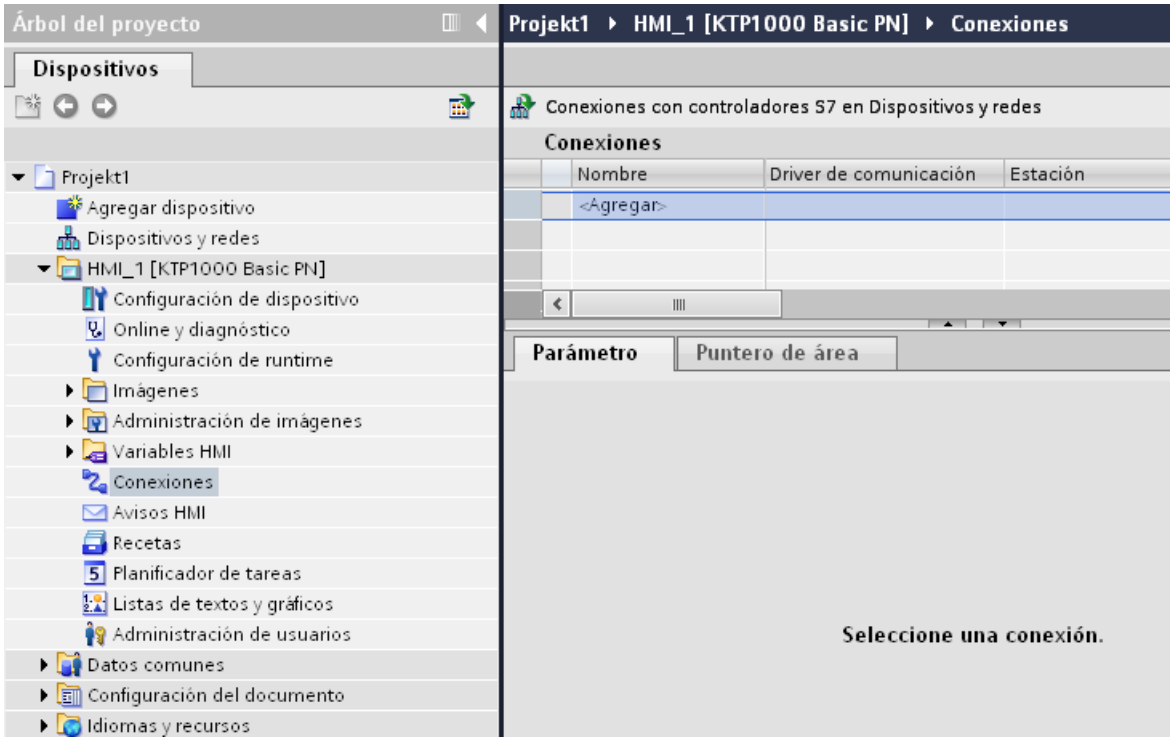
Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador.

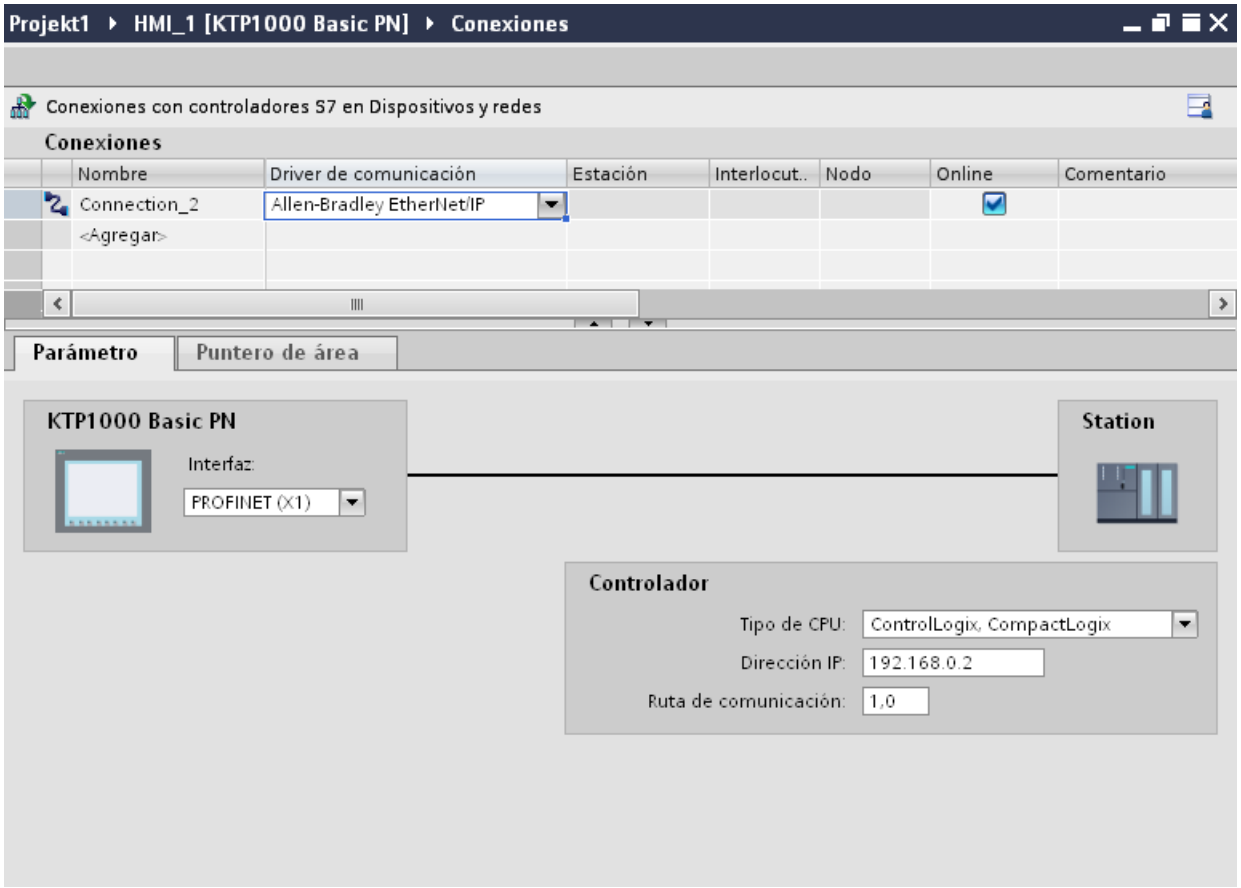
Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".

3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el driver "Allen-Bradley EtherNet/IP" en la columna "Driver de comunicación".



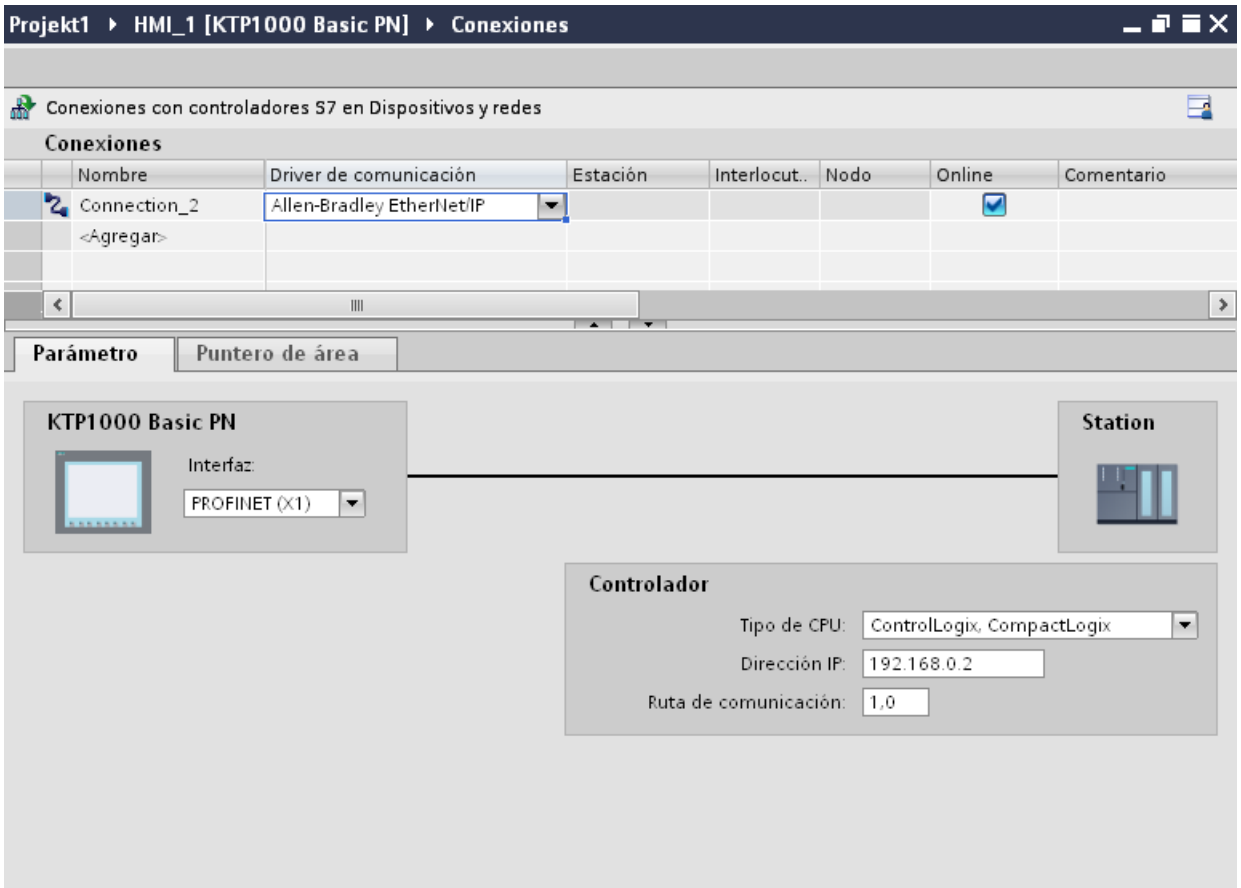
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Allen-Bradley EtherNet/IP)

Parámetros que deben configurarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existe el área "Controlador", según sea la interfaz utilizada.



Parámetros del panel de operador

En la ventana de inspección "Parámetros" sólo se puede seleccionar una interfaz para el panel de operador. En función del panel de operador habrá diferentes interfaces disponibles.

Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC. En la siguiente carga se transfiere la dirección IP al panel de operador.

Nota

Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

Si activa "Obtener dirección IP por otra vía", en la siguiente carga la dirección IP establecida previamente se conserva en el Control Panel.

La dirección IP del panel de operador se configura del siguiente modo:

1. Haga clic en el panel de operador.
2. Abra el editor "Configuración de dispositivos".

3. Haga clic en la interfaz Ethernet.
4. Asigne la dirección IP en la ventana de inspección en:
"General > Interfaz PROFINET > Direcciones Ethernet"

Parámetros del controlador

- Tipo de CPU
En "Tipo de CPU" se ajusta el tipo de CPU del controlador utilizado.
- Dirección IP
Ajuste la dirección IP o el nombre de host del módulo Ethernet/IP del controlador. En un Basic Panel sólo puede utilizarse la dirección IP.
- Ruta de comunicación
Ajuste la ruta CIP desde el módulo Ethernet hasta el controlador. Con ello se establece una conexión lógica entre el módulo Ethernet y el PLC, aunque ambos estén en redes CIP distintas.
Encontrará más información al respecto en el capítulo: Ejemplos: Ruta de comunicación

Conectar el panel de operador con el controlador

Conexiones mediante Allen-Bradley EtherNet/IP

Conexión

El panel de operador puede conectarse al controlador Allen-Bradley utilizando los componentes siguientes:

- Una red Ethernet existente en la que también se encuentran los controladores
- Un cable Ethernet cruzado (Cross-Over) directamente a la interfaz Ethernet de la CPU o del módulo de comunicación

El acoplamiento entre el panel de operador y un controlador Allen-Bradley se limita principalmente a la conexión física del panel de operador. No se requieren bloques especiales en el controlador para el acoplamiento.

Tipos de comunicación

Tipos de comunicación activados con Allen-Bradley EtherNet/IP

Los siguientes tipos de comunicación han sido comprobados y activados:

- Acoplamiento punto a punto a los controles activados
- Acoplamiento a varios puntos de un panel de operador (Allen-Bradly Ethernet/IP-Client) con hasta 4 autómatas, con los controles correspondientes activados. Se puede mezclar los tipos de CPU.

Acoplamiento

El acoplamiento a los siguientes autómatas está activado con Allen-Bradley EtherNet/IP:

- Tipo de CPU: "ControlLogix, Compact Logix"
  - ControlLogix
    556x(1756-L6x) con módulo Ethernet 1756-ENBT
  - Sistema Guard Logix ControlLogix
    556x(1756-L6xS) con módulo Ethernet 1756-ENBT
  - CompactLogix
  - 533xE(1769-L3xE) con interfaz Ethernet
  - 532xE(1769-L2xE) con interfaz Ethernet
  - 534x (1768-L4x) con módulo Ethernet 1768-ENBT
- Tipo de CPU: "SLC, MicroLogix"
  - MicroLogix 1100 (con interfaz Ethernet)
  - MicroLogix 1400 (con interfaz Ethernet)
  - SLC 5/05 (con interfaz Ethernet)

Características de rendimiento de la comunicación

Tipos de datos admisibles para Allen-Bradley EtherNet/IP

Tipos de datos admisibles

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de CPU: ControlLogix, CompactLogix

Tipo de datos	Longitud
Bool	1 bit
DInt	4 bytes
Int	2 bytes
Real	4 bytes
SInt	1 byte
String	de 1 a 80 caracteres
UDInt	4 bytes
UInt	2 bytes
USInt	1 byte

Matrices de tipos de datos admisibles

Dirección	Tipos de datos admisibles
Array	SInt, USInt, Int, UInt, DInt, UDInt, Real
Bits individuales de los tipos de datos básicos del controlador SInt, USInt, Int, UInt, DInt, UDInt	Bool*

* En algunos bits, tras modificar el bit indicado, se vuelve a escribir todo el valor en el controlador. No se comprueba si entre tanto se han modificado otros bits en el valor. Por este motivo, el controlador (o un equipo diferente) sólo puede acceder al valor en lectura.

Tipo de CPU: SLC, MicroLogix

Tipo de datos	Tipo de operando	Longitud
ASCII	A	de 0 a 80 caracteres
Bool	N, R, C, T, B, S, I, O	1 bit
DInt	N	4 bytes
Int	N, R, C, T, S	2 bytes
Real	N, F	4 bytes
String	ST	de 1 a 82 caracteres
UDInt	N	4 bytes
UInt	N, R, C, T, B, I, O	2 bytes

Matrices de tipos de datos admisibles

Dirección	Tipos de datos admisibles
Array	Int, UInt, DInt, UDInt, Real

Particularidades de los acoplamientos con Allen Bradley Ethernet/IP

Con el driver de comunicación Allen Bradley Ethernet/IP y el tipo de CPU SLC, MicroLogix las variables de matriz se pueden utilizar sólo para avisos de bit y curvas.

Nota

Los módulos de entrada/salida con 8 ó 16 puertos ocupan una palabra entera en el controlador.

Los módulos de entrada/salida con 24 ó 32 puertos ocupan dos palabras.

Si en el panel de operador están ocupados bits no existentes, éste no visualizará ningún aviso de error.

Al ajustar la configuración procure por tanto que en los módulos de entrada/salida con 8 y 24 puertos sólo se ocupen aquellos bits que tengan asignado un puerto.

Tipos de CPU soportados para Allen-Bradley EtherNet/IP

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Allen-Bradley EtherNet/IP.

- CompactLogix
  - 1769-L2xE con interfaz Ethernet integrada
  - 1769-L3xE con interfaz Ethernet integrada
  - 1768-L4x con módulo Ethernet 1768-ENBT
- ControlLogix
  - 1756-L6x con módulo Ethernet 1756-ENBT
- GuardLogix
  - 1756-L61S con módulo Ethernet 1756-ENBT
  - 1756-L62S con módulo Ethernet 1756-ENBT
  - 1756-L63S con módulo Ethernet 1756-ENBT
- MicroLogix
  - MicroLogix 1100 / 1400
- SLC50x
  - SLC5/05

Direccionamiento en el tipo de CPU C.Logix

Direccionamiento

Direccionamiento

En WinCC, una variable se identifica unívocamente mediante una dirección en el controlador. A este respecto, la dirección debe corresponder al nombre de la variable en el controlador. La dirección de una variable se indica con una secuencia de 128 caracteres como máximo.

Uso de caracteres para el direccionamiento

Los caracteres permitidos para las direcciones de variables son:

- Letras (a-z, A-Z)
- Números (0-9)
- Carácter de subrayado (_)

La dirección de una variable se compone del nombre de la misma, así como de cadenas de caracteres adicionales para identificar exactamente la variable en el controlador.

Un nombre de variable debe tener las propiedades siguientes:

- El nombre puede comenzar pero no terminar con un carácter de subrayado.
- No está permitido utilizar varios caracteres de subrayado y/o espacios en blanco consecutivos.
- La longitud máxima de la dirección es de 128 caracteres.

Nota

Los caracteres reservados para direccionar las variables no pueden utilizarse en los nombres del programa ni de las variables, ni tampoco en otra parte de la dirección.

Los caracteres reservados se indican a continuación:

Carácter reservado	Función
.	Separación entre elementos
:	Indicación de una variable del programa
,	Separación al direccionar matrices multidimensionales
/	Reservado para direccionar bits
[ ]	Direccionamiento de elementos de matriz o de matrices

Variables del controlador y del programa

El driver de comunicación Allen-Bradley EtherNet/IP permite acceder a variables del controlador (variables globales del proyecto) y/o a variables del programa (variables globales del programa).

Una variable del programa se declara separando mediante dos puntos el nombre del programa en el controlador y el nombre de la variable en sí. En cambio, las variables del controlador se declaran simplemente mediante su nombre.

Nota

Errores de direccionamiento

Si los nombre de variable y los tipos de datos no concuerdan, se producirán errores de direccionamiento.

Al efectuar el direccionamiento, asegúrese de que el nombre de la variable en el campo de dirección de WinCC concuerde con el nombre de la variable en el controlador. Asegúrese de que los tipos de datos de las variables en WinCC concuerden con los tipos de datos en el controlador.

Nota

Las variables específicas de módulos (p. ej. para datos de los módulos de entradas y salidas) no se pueden direccionar directamente. En cambio, utilice un alias en el controlador.

Ejemplo: Local:3:O.Data no se puede direccionar en WinCC.

Si para Local:3:O se ha definido el alias "MyOut" en el controlador, se podrá direccionar con WinCC a través de MyOut.Data.

Sintaxis del direccionamiento

Reglas de sintaxis para el direccionamiento

En las tablas siguientes se define la sintaxis de las diversas posibilidades de direccionamiento con Allen-Bradley EtherNet/IP.

Tabla 10-12 Acceso a matrices, tipos de datos básicos y elementos de estructuras

Tipos de datos	Tipo	Dirección
Tipos de datos básicos	Variable del controlador	Nombre de la variable
	Variable del programa	Nombre del programa:Nombre de la variable
Matrices	Variable del controlador	Variable de matriz
	Variable del programa	Nombre del programa:Variable de matriz
Bits	Variable del controlador	Nombre de la variable/Número de bit
	Variable del programa	Nombre del programa:Nombre de la variable/Número de bit
Elementos de estructuras	Variable del controlador	Variable de estructura.Elemento de estructura
	Variable del programa	Nombre del programa:Variable de estructura.Elemento de estructura

Nota

En los tipos de datos Bool, Real y String no se permite direccionar bits, puesto que esto provocaría errores de direccionamiento.

Descripción de la sintaxis

A continuación se describe la sintaxis de forma resumida.

```
(Nombre del programa:)Nombre de la variable([x(,y)(,z)]){.Nombre de
la variable([x(,y)(,z)])}(/Número de bit)
```

- La expresión "( )" significa que la expresión es opcional y que sólo puede existir una vez.
- La expresión "{ }" significa que la expresión es opcional y que puede existir un número cualquiera de veces.

La longitud de una dirección no puede exceder el límite superior de 128 caracteres.

Tipos de direccionamiento

Matrices

Una matriz es una estructura que contiene una cantidad de datos de un mismo tipo. En WinCC sólo se pueden crear matrices unidimensionales.

En la columna "Dirección" del editor de variables, introduzca el nombre de la matriz, indicando opcionalmente un elemento inicial. La longitud se define en el campo de entrada Elemento de matriz en el editor de variables. Si se exceden los límites de las matrices en el controlador (debido a una indexación errónea), se producirán errores de direccionamiento.

Estas matrices se deben declarar en el controlador como variables del programa o del controlador.

Las matrices bidimensionales o tridimensionales del controlador sólo se pueden direccionar en WinCC si pueden mapearse por áreas en matrices unidimensionales.

Nota

En todos los accesos de lectura y escritura se leen o escriben siempre todos los elementos de matriz de una variable. Si una variable de matriz está conectada con un controlador, cuando se produzca una modificación se transferirá básicamente todo el contenido. Por ello, el panel de operador y el controlador no pueden escribir valores en la misma variable de matriz al mismo tiempo. Al realizar una operación de escritura de un elemento, se escribe en el controlador la matriz entera y no sólo el elemento en cuestión.

Elementos de matriz

Los elementos de matrices unidimensionales, bidimensionales y tridimensionales en el controlador se indexan en el editor de variables mediante un índice con la sintaxis correspondiente. El direccionamiento de una matriz comienza con el elemento 0. Para direccionar los elementos se admiten matrices de todos los tipos básicos. Con esto se escribe o se lee sólo el elemento direccionado y no la matriz entera.

Bits y variables binarias

Está permitido acceder a bits individuales en todos los tipos de datos básicos, excepto Bool, Real y String. El direccionamiento de bits también se permite en los elementos de matrices y estructuras. Al direccionar bits y variables binarias en los tipos de datos básicos se ajusta en WinCC el tipo de datos Bool.

Los números de bits de un dígito se direccionan con "/x" ó "/0x" (x = número de bit). Los números de bit se indican como máximo con dos dígitos.

Nota

En el caso del tipo de datos "Bool" de los tipos de datos SInt, Int y DInt, se vuelve a escribir toda la variable en el controlador después de cambiar el bit indicado. No se comprueba si entretanto se han modificado otros bits de la variable. Por este motivo, el controlador sólo puede leer la variable indicada.

Estructuras

Las estructuras se utilizan para crear tipos de datos de usuario. En las estructuras se reúnen variables de distintos tipos de datos. Las estructuras pueden comprender tipos básicos, matrices y otras estructuras. En WinCC se direccionan sólo los elementos de estructuras y no las estructuras enteras.

Elementos de estructuras

Los elementos de estructuras se direccionan mediante el nombre de la estructura y el nombre del elemento deseado. La sintaxis de este direccionamiento se efectúa separando los componentes mediante un punto. Los elementos de estructuras pueden ser no sólo tipos de datos básicos, sino también matrices y otras estructuras. Sólo se permiten matrices unidimensionales como elementos de estructuras.

Nota

La profundidad de anidamiento de las estructuras se ve limitada sólo por la longitud máxima de 128 caracteres de la dirección.

Multiplexar direcciones

Multiplexar direcciones

Con el tipo de CPU CompactLogix, ControlLogix se puede multiplexar direcciones.

Para multiplexar las direcciones se requieren dos variables:

- "Tag_1" es una variable del tipo de datos "String" que contiene una dirección lógica como valor, p. ej. "HMI:Robot5.Block5".
  No obstante, el valor puede modificarse a una segunda dirección válida, p. ej. "HMI:Robot4.Block3".
- "Tag_2" es una variable para la que el driver de comunicación "Allen-Bradley EtherNet/IP" está configurado como conexión.
  Como dirección introduzca un nombre válido de una variable HMI entre corchetes.
  - p. ej.: "[Tag_1]"
  - La variable debe ser del tipo de datos String.
  - Los corchetes indican que la dirección se multiplexa.
  - La dirección es el valor actual de la variable "Tag_1".

  Nota

  Sólo es posible multiplexar direcciones Allen-Bradley EtherNet/IP enteras, pero no partes de ellas. "HMI:Robot[Tag_1].Block5" no es una dirección válida.

  También puede hacer clic en la columna "Dirección" en la flecha derecha. En el cuadro de diálogo siguiente, seleccione mediante la flecha en el borde izquierdo la entrada "Multiplex" en vez de "Constante". En la lista de selección de variables aparecerán entonces sólo las variables del tipo de datos "String".

  En las variables multiplexadas también se puede asociar una función al evento "Modificación de valor".

Ejemplos de direccionamiento

Tabla de ejemplos de direccionamiento

En la tabla siguiente se indican las variantes de direccionamiento básicas para las variables del autómata. Otras variantes de direccionamiento se obtienen creando combinaciones.

Tipo	Clase	Dirección
Generalidades	Variable del autómata	Nombre de la variable
	Variable del programa	Programa:Nombre de la variable
Matriz	Acceso a un elemento de una matriz bidimensional	Variable de matriz[Dim1,Dim2]
	Elemento de una matriz de estructura (unidimensional)	Variable de matriz[Dim1].Elemento de estructura
	Bit en elemento de matriz de tipo básico (bidimensional)	Variable de matriz[Dim1,Dim2]/Bit
Estructura	Matriz en estructura	Variable de estructura.Variable de matriz
	Bit en elemento de una matriz en subestructura	Variable de estructura.Estructura2.Variable de matriz [Elemento]/Bit

Nota

Si desea direccionar variables del programa, deberá anteponer a la dirección el nombre del programa en el autómata, separado con dos puntos.

Ejemplo: Nombre del programa:Variable de matriz[Dim1,Dim2]

Acceder a los elementos de matriz

Tipo	Dirección
Variable del autómata	Variable de matriz[Dim1]
	Variable de matriz[Dim1,Dim2]
	Variable de matriz[Dim1,Dim2,Dim3]
Variable del programa	Nombre del programa:Variable de matriz[Dim1]
	Nombre del programa:Variable de matriz[Dim1,Dim2]
	Nombre del programa:Variable de matriz[Dim1,Dim2,Dim3]

Ejemplos: Ruta de comunicación

Ejemplo 1:

Conexión con un autómata que se encuentre en el mismo bastidor Allen-Bradley.

1,0

Número	Significado
1	Representa una conexión backplane.
0	Representa el número de slot de la CPU.

Ejemplo 2:

Conexión con un autómata que se encuentre en un bastidor Allen-Bradley diferente. Dos bastidores Allen-Bradley están conectados con Ethernet.

1,2,2,190.130.3.101,1,5

Número	Significado
1	Conexión backplane
2	Representa el número de slot del segundo módulo Ethernet.
2	Representa una conexión con la red Ethernet.
190.130.3.101	Dirección IP de un bastidor AB diferente en la red – especialmente el tercer módulo Ethernet

Número	Significado
1	Conexión backplane
5	Número de slot de la CPU

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   - Interfaz
   - Parámetros de transferencia
   - Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Nota

Cuando el firmware del controlador CompactLogix es anterior a la versión 18, puede que en algunos casos tenga que reiniciar el panel de operador tras una transferencia del programa PLC.

O bien separe la conexión antes de transferir el programa PLC y vuelva a establecerla tras la transferencia del programa PLC.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Allen-Bradley DF1

Configurar una conexión mediante Allen-Bradley DF1

Introducción

Una conexión con un controlador que tenga un driver de comunicación Allen-Bradley DF1 se configura en el editor "Conexiones" del panel de operador.

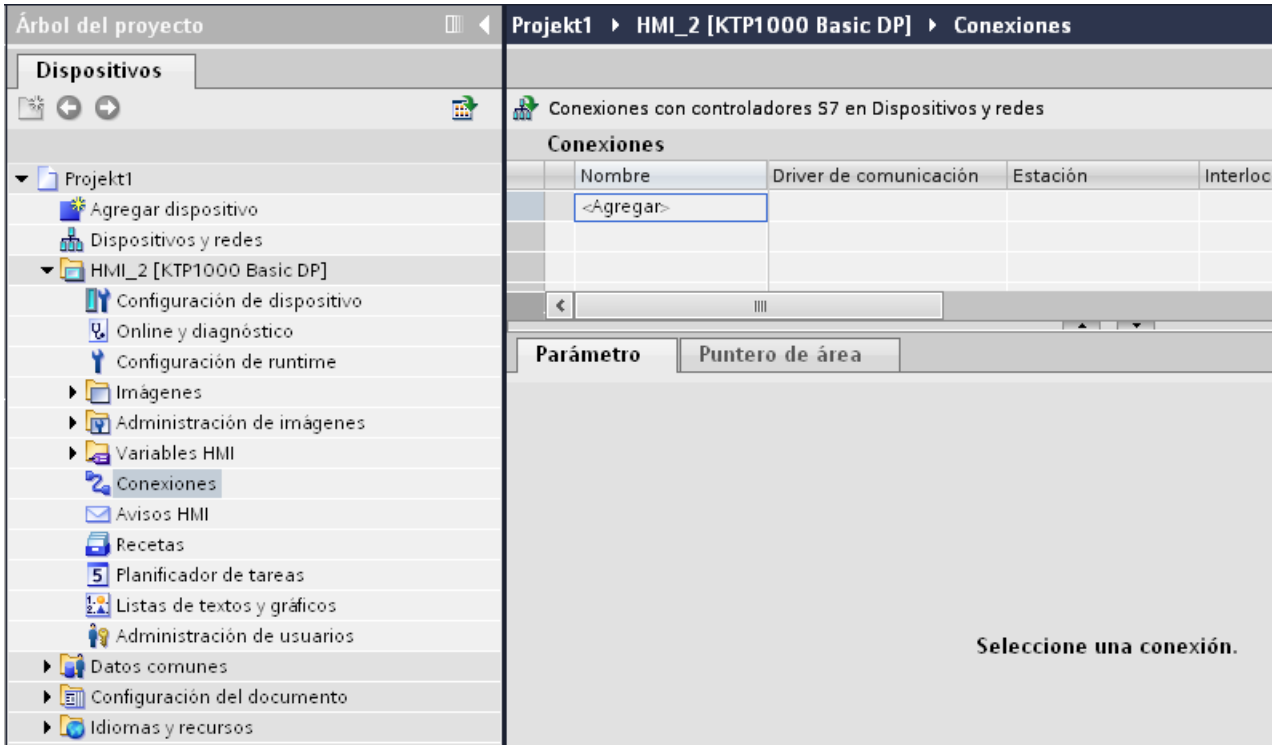
Las interfaces reciben nombres distintos en función del panel de operador.

Requisitos

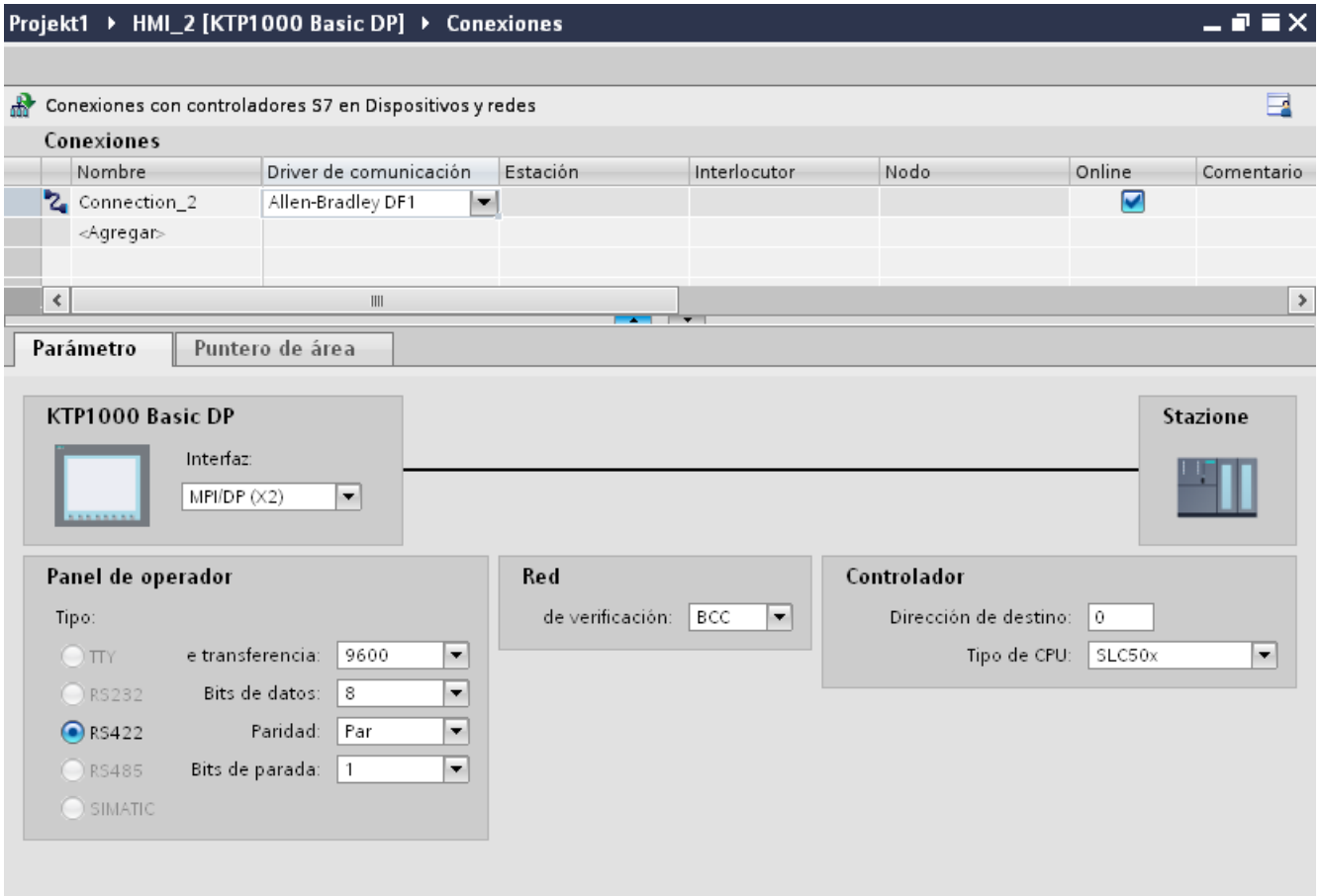
- Hay un proyecto abierto.
- Se ha creado un panel de operador.

Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".
3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el driver "Allen-Bradley DF1" en la columna "Driver de comunicación".



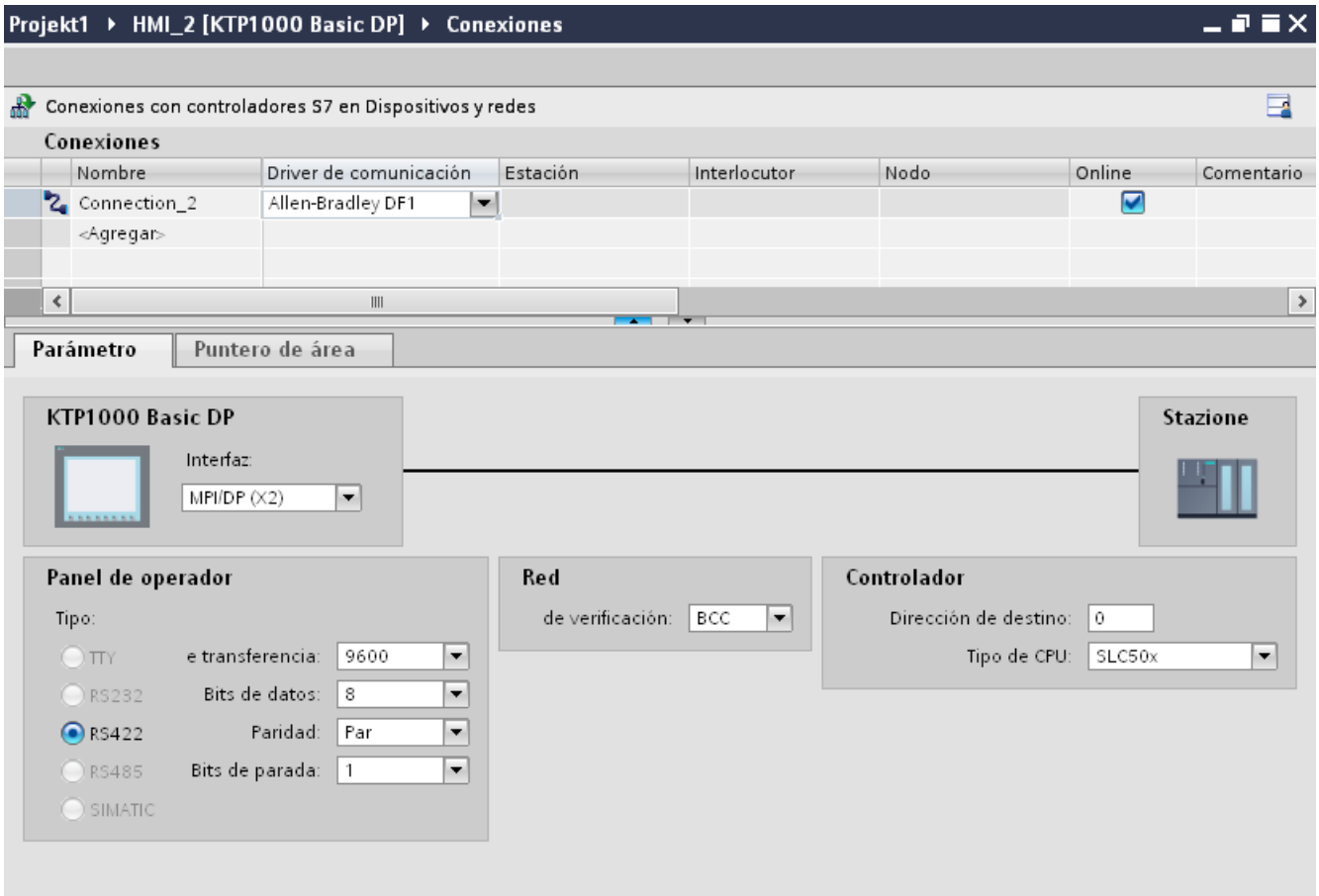
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Allen-Bradley DF1)

Parámetros que deben configurarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador", "Red" y "Controlador", según sea la interfaz utilizada.



Parámetros del panel de operador

- Interfaz
  En "Interfaz" se selecciona la interfaz del panel de operador a la que está conectado el controlador.
  Encontrará más información en el manual de producto del panel de operador.
- Tipo
  Define la conexión física utilizada.

  Nota

  Cuando se utiliza la interfaz IF1B, hay que conmutar además los datos de recepción RS 485 y la señal RTS mediante el interruptor 4 DIL de la parte posterior de los paneles de operador.

- Velocidad de transferencia
  En "Velocidad de transferencia" se selecciona la velocidad de transferencia entre el panel de operador y el controlador.
- Bits de datos
  En "Bits de datos" hay que elegir entre "7 bits" u "8 bits".

- Paridad
  En "Paridad" hay que elegir entre "Ninguna", "Par" e "Impar".
- Bits de parada
  En "Bits de parada" hay que elegir entre 1 y 2 bits.

Parámetros de la red

- Suma de verificación
  En "Suma de verificación" se elige el procedimiento para determinar el código de error: "BCC" o "CRC".

Parámetros del controlador

- Dirección de destino
  En "Dirección de destino" se elige la dirección del controlador. En un acoplamiento DF1 punto a punto se ajusta la dirección 0.
- Tipo de CPU
  En "Tipo de CPU" se ajusta el tipo de CPU del controlador utilizado.

Nota

Parametrice el driver DF1 FULL-DUPLEX en la CPU del modo indicado a continuación: "NO HANDSHAKING" en "Control Line" y "AUTO-DETECT" en "Embedded Responses".

Conectar el panel de operador con el controlador

Conexiones mediante Allen-Bradley DF1

Conexión

La conexión estará establecida cuando los parámetros de la interfaz del autómata y del panel de operador se hayan realizado unos sobre otros. No se requieren bloques especiales para el acoplamiento en el controlador.

Nota

La empresa Rockwell ofrece una gran variedad de adaptadores de comunicación para integrar "nodos DF1" en las redes DH485, DH y DH+. De estos acoplamientos están autorizados los directos y los que utilizan los módulos KF2 y KF3. Todos los demás no han sido probados en el sistema por SIEMENS AG y, por lo tanto, no están autorizados.

Interlocutores para Allen-Bradley DF1

Controladores acoplables

Los drivers de comunicación indicados se encuentran a disposición para los siguientes controladores Allen-Bradley: :

Controlador	DF1 (punto a punto) RS 232	DF1 (punto a punto) RS 422	DF1 (multipunto) a través del módulo KF2 a DH+ LAN RS 232/RS 422	DF1 (multipunto) a través del módulo KF3 a DH485 LAN RS 232
SLC500	–	–	–	X
SLC501	–	–	–	X
SLC502	–	–	–	X
SLC503	X	–	–	X
SLC504	X	–	X	X
SLC505	X	–	–	X
MicroLogix	X	–	–	X
PLC-5 [1)]	X	X	X	–

[1)] Para PLC-5 sólo están liberados los siguientes procesadores: PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60 y PLC-5/80.

Tipos de comunicación

Controlador con driver de comunicación Allen-Bradley DF1

Este apartado describe la comunicación entre un panel de operador y los siguientes controladores Bradley:

- SLC500
- SLC501
- SLC502
- SLC503
- SLC504
- SLC505
- PLC5
- MicroLogix

El acoplamiento tiene lugar en todos estos controladores vía protocolos propios de los controladores Allen Bradley DF1, Allen Bradley DH485 y Allen Bradley DH+.

Aquí siempre se puede utilizar el driver de comunicación Allen-Bradley DF1, cuyo protocolo, en la comunicación multipunto con los módulos de comunicación KF2 (Allen Bradley DH+) y

KF3(Allen Bradley DH485), se convierte en uno de los otros dos protocolos de los controladores.

Tipos de comunicación autorizados con Allen-Bradley DF1

Los siguientes tipos de comunicación han sido comprobados y activados:

- HMI (Allen Bradley DF1)
  Acoplamiento punto a punto
- HMI (Allen Bradley DF1)
  Vía módulo KF2 a Allen Bradley DH+ (comunicación con hasta 4 controladores)
- HMI (Allen Bradley DF1)
  Vía módulo KF3 a Allen Bradley DH485 (comunicación con hasta 4 controladores)

Controladores acoplables

El driver de comunicación Allen Bradley DF1 está a disposición para los siguientes controladores Allen-Bradley:

Controlador	DF1 (punto a punto) RS 232	DF1 (punto a punto) RS 422	DF1 (multipunto) a través del módulo KF2 a DH+ LAN RS 232/RS 422	DF1 (multipunto) a través del módulo KF3 a DH485 LAN RS 232 [2]
SLC500	–	–	–	X
SLC501	–	–	–	X
SLC502	–	–	–	X
SLC503	X [2]	–	–	X
SLC504	X [2]	–	X	X
SLC505	X [2]	–	–	X
MicroLogix	X [2]	–	–	X
PLC-5 [1]	X	X	X	–

[1] Para PLC-5 sólo están liberados los siguientes procesadores: PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60 y PLC-5/80.

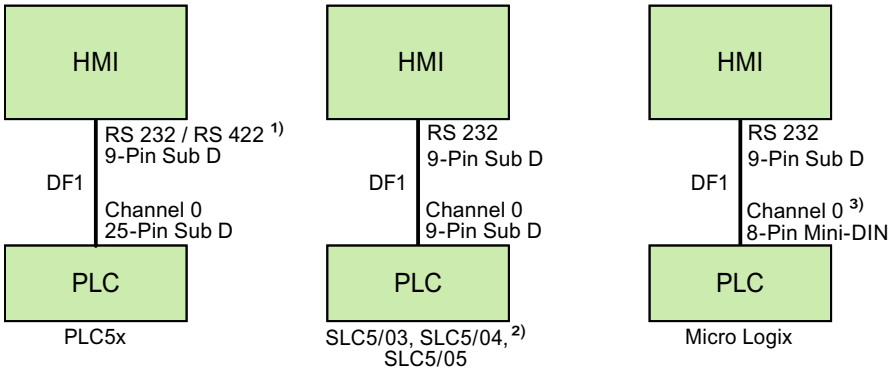
[2] Para paneles de operador que sólo tienen una interfaz RS 422/485 y el interlocutor es una interfaz RS 232, el convertidor RS 422/232 está probado y liberado.

Referencia: 6AV6 671-8XE00-0AX0

Protocolo DF1 con acoplamiento multipunto

Acoplamiento punto a punto con el protocolo DF1

El protocolo DF1 permite establecer únicamente acoplamientos punto a punto.



1) En Panel PC y PC sólo es posible RS 232.

2) No es posible realizar un acoplamiento punto a punto con los controladores SLC500, SLC501 y SLC502 mediante DF1.

3) En Micro Logix ML1500 LRP también es posible Channel 1 (9 pines Sub D).

Cable de conexión

Interfaz HMI Panel utilizada	Para conexión con PLC5x	Para conexión con SLC5/03, SLC5/04, SLC5/05	Para conexión con MicroLogix
RS 232 de 9 pines	Cable Allen-Bradley 1784-CP10	Cable Allen-Bradley 1747-CP3	Cable Allen-Bradley 1761-CBL-PM02
RS 422 de 9 pines	Cable de conexión de 9 pines RS 422 Sub D	—	—

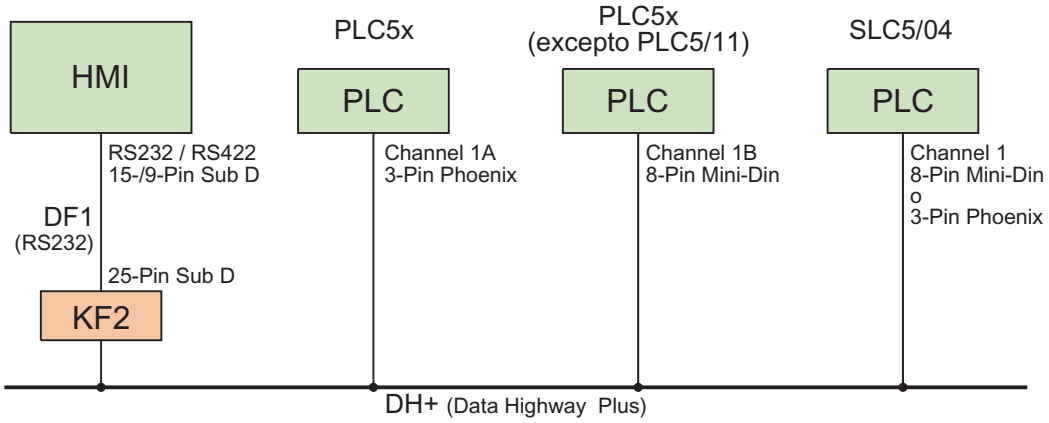
En el manual de producto correspondiente encontrará información sobre la interfaz que debe utilizarse en el panel de operador.

Encontrará la asignación de pines de los cables en el capítulo "Cables de conexión para Allen-Bradley".

Protocolo DF1 con acoplamiento multipunto mediante el módulo KF2

Protocolo DF1 con acoplamiento multipunto mediante el módulo KF2 a DH+ LAN

Empleando un protocolo de interfaz de módulo KF2 se puede realizar un acoplamiento con los controladores en DH+ LAN (Data Highway Plus Local Area Network).



Cable de conexión

Interfaz HMI Panel utilizada	Para la conexión con el módulo de interfaz KF2
RS 232 de 9 pines	Cable Allen-Bradley 1784-CP10 y adaptador de 25 pines hembra/hembra
RS 422 de 9 pines	Cable de conexión RS 422 Sub D de 9 pines y adaptador de 25 pines hembra/hembra

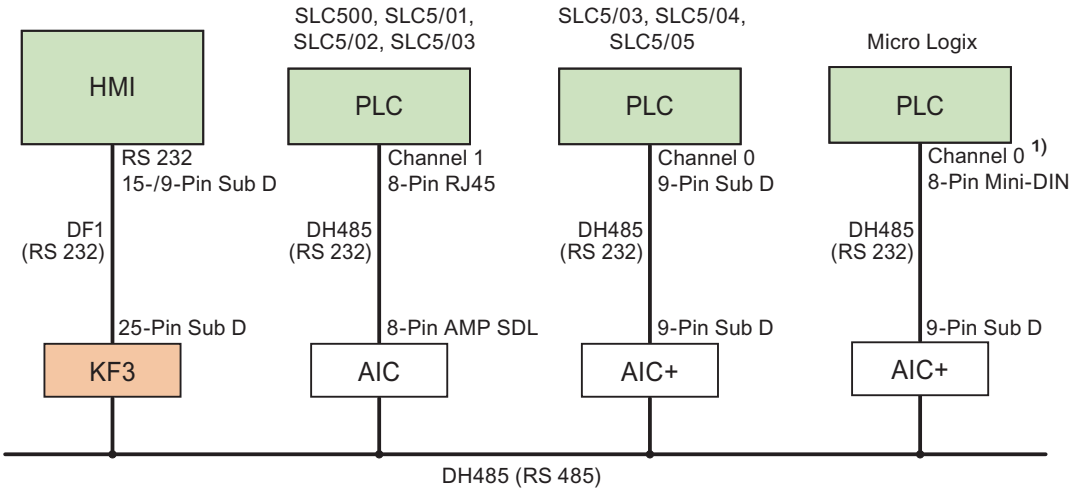
Consulte el cableado de los controladores al bus de datos DH+ en la documentación de Allen-Bradley.

En el manual de producto correspondiente encontrará información sobre la interfaz que debe utilizarse en el panel de operador.

Encontrará la asignación de pines de los cables en el capítulo "Cables de conexión para Allen-Bradley".

Protocolo DF1 con acoplamiento multipunto mediante el módulo KF3

Protocolo DF1 con acoplamiento multipunto mediante el módulo KF3 a DH485 LAN



1) En Micro Logix ML1500 LRP también es posible Channel 1 (9 pines Sub D).

Cable de conexión

Interfaz HMI Panel utilizada	Para la conexión con el módulo de interfaz KF3
RS 232 de 9 pines	Cable Allen-Bradley 1784-CP10 y adaptador de 25 pines hembra/hembra

En el manual de producto correspondiente encontrará información sobre la interfaz que debe utilizarse en el panel de operador.

Encontrará la asignación de pines de los cables en el capítulo "Cables de conexión para Allen-Bradley".

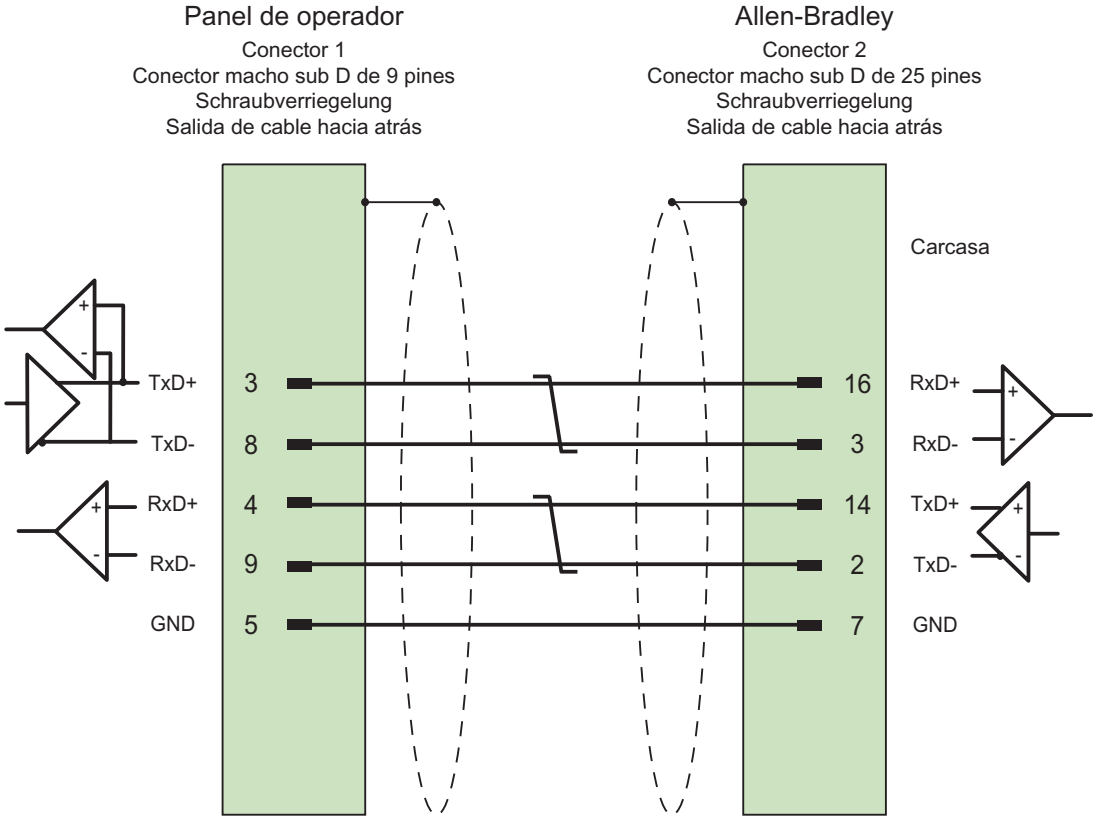
Cables de conexión para Allen-Bradley DF1

Cable de conexión RS 422 Sub D de 9 polos para Allen-Bradley

Cable de conexión RS 422 Sub D de 9 polos

Para conexión con panel de operador (RS 422, conector Sub D de 9 pines) - PLC5x, KF2, KF3.

Para la conexión con KF2 y KF3 se requiere adicionalmente un adaptador (Gender Changer) de 25 pines, hembra / hembra.



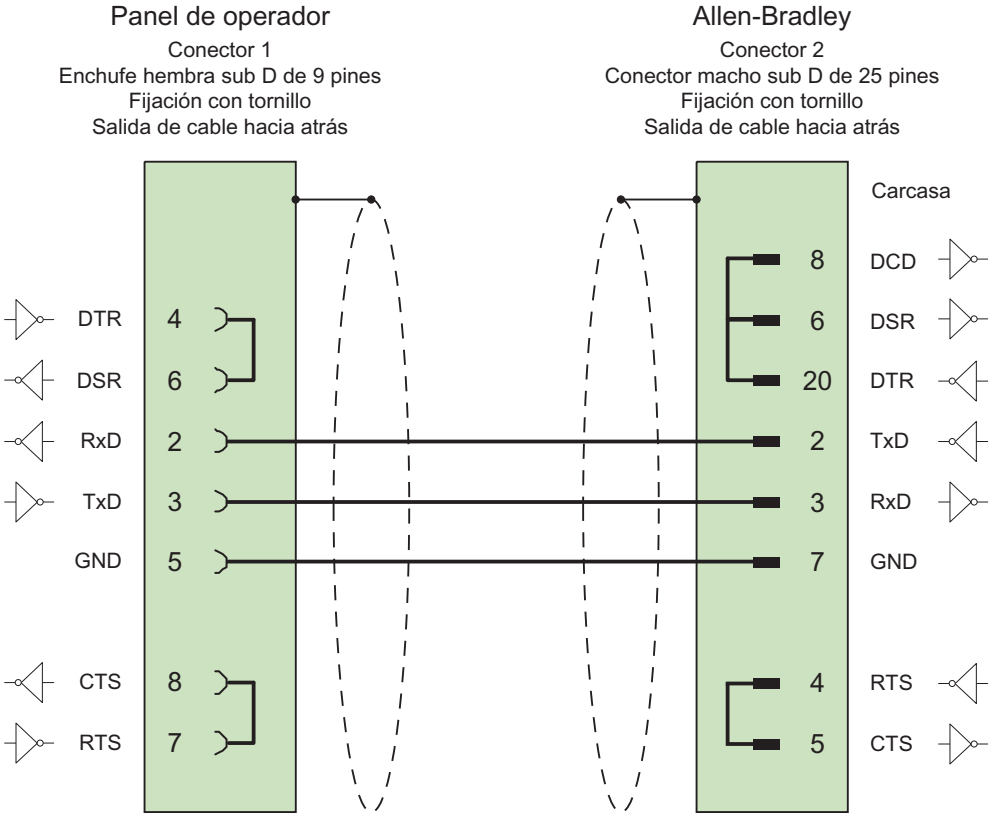
Blindaje conectado a la carcasa en ambas partes y con gran superficie, y contactos de pantalla conectados
Cable: 3 x 2 x 0,14 mm^2, blindado,
longitud máx. 60 m

Cable de conexión 1784-CP10, RS 232, para Allen-Bradley

Cable Allen-Bradley 1784-CP10

Para la conexión con panel de operador (RS 232, conector Sub D de 9 pines) - PLC5x, KF2, KF3

Para la conexión con KF2 y KF3 se requiere adicionalmente un adaptador (Gender Changer) de 25 pines, hembra / hembra.



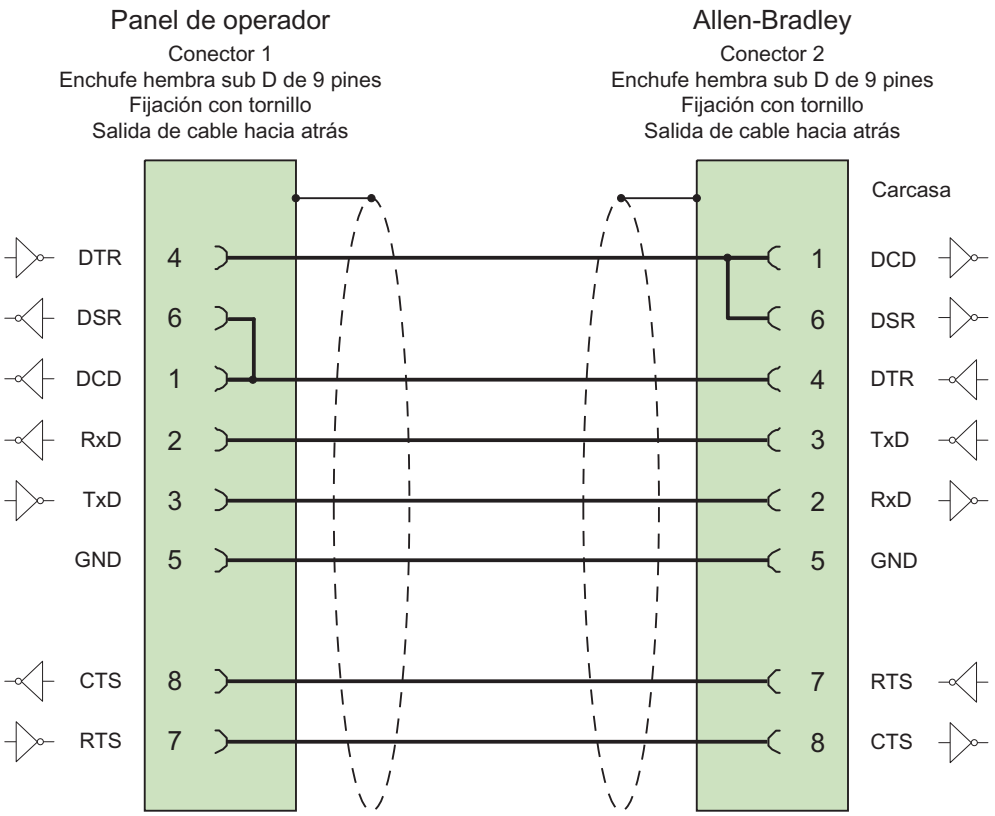
Blindaje conectado a la carcasa en ambos lados y con gran superficie

longitud máx. 15 m

Cable de conexión 1747-CP3, RS 232, para Allen-Bradley

Cable Allen-Bradley 1747-CP3

Para conexión con panel de operador (RS 232, conector Sub D de 9 pines) - SLC503, SLC504, SLC505 (Channel 0), AIC+



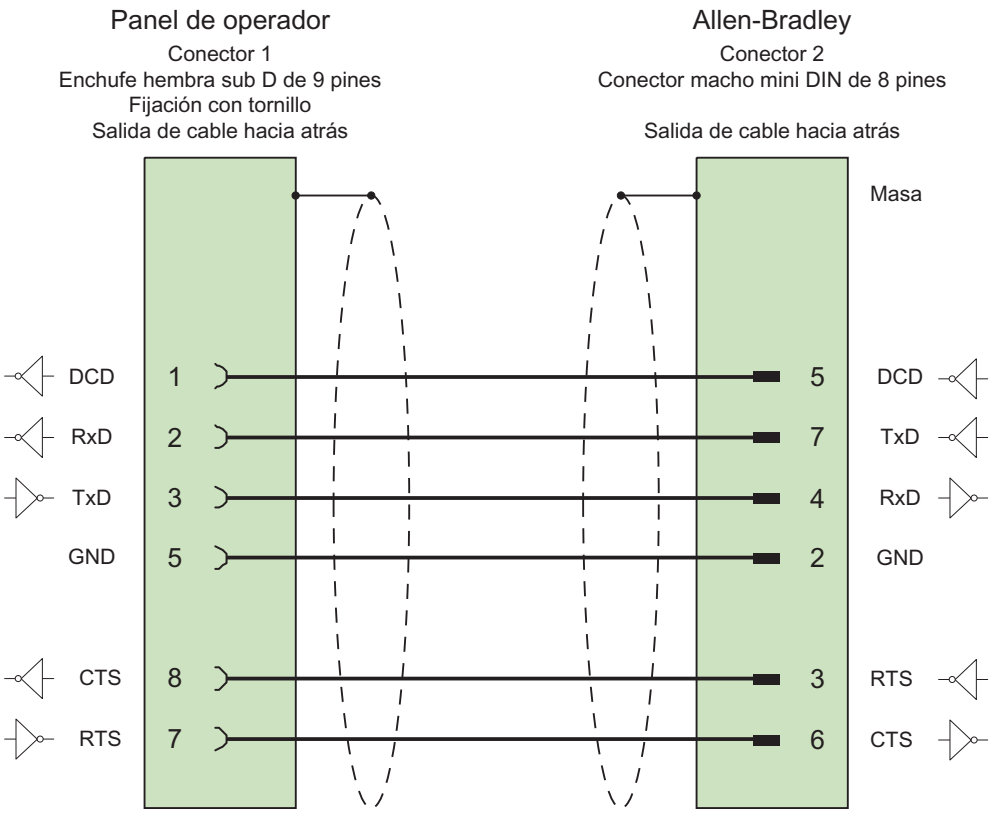
Blindaje conectado a la carcasa en ambos lados y con gran superficie

longitud máx. 3 m

Cable de conexión 1761-CBL-PM02, RS 232, para Allen-Bradley

Cable Allen-Bradley 1761-CBL-PM02

Para conexión con panel de operador (RS 232, conector Sub D de 9 pines) – Micro Logix, AIC +



Blindaje conectado a la carcasa en ambos lados y con gran superficie

longitud máx. 15 m

Características de rendimiento de la comunicación

Tipos de datos admisibles para Allen-Bradley DF1

Tipos de datos admisibles para Allen-Bradley DF1

En la tabla se indican los tipos de datos de usuario que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Tipo de operando	Longitud
ASCII	A [1)]	de 1 a 80 caracteres
Bool	N, R, C, T, B, S, I, O	1 bit
Int	N, R, C, T, S	2 bytes
DInt	N, D [2)]	4 bytes
UInt	N, R, C, T, B, I, O, D [2)]	2 bytes

Tipo de datos	Tipo de operando	Longitud
UDInt	N, D [2]	4 bytes
Real	N, F [1]	4 bytes

[1] Seleccionable en función del tipo de CPU elegido.

[2] Sólo en el tipo de CPU PLC5

Abreviaturas

En WinCC se abrevian los formatos de los tipos de datos del siguiente modo:

- UNSIGNED INT = UInt
- UNSIGNED LONG = UDInt
- SIGNED INT = Int
- SIGNED LONG = DInt

Particularidades de los acoplamientos con Allen-Bradley DF1

Con Allen Bradley DF1 las variables de matriz se pueden utilizar en WinCC sólo para avisos de bit y curvas.

Nota

Los módulos de entrada/salida con 8 ó 16 puertos ocupan una palabra entera en el controlador.

Los módulos de entrada/salida con 24 ó 32 puertos ocupan dos palabras.

Si en el panel de operador están ocupados bits no existentes, éste no visualizará ningún aviso de error.

Al ajustar la configuración procure por tanto que en los módulos de entrada/salida con 8 y 24 puertos sólo se ocupen aquellos bits que tengan asignado un puerto.

Tipos de CPU soportados para Allen-Bradley DF1

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Allen-Bradley DF1.

- SLC
  - SLC500
  - SLC501
  - SLC502
  - SLC503
  - SLC504
  - SLC505
- MicroLogix
  - MicroLogix 1x00
  - MicroLogix 1100 / 1400
- PLC 5
  - PLC-5/11
  - PLC-5/20
  - PLC-5/40
  - PLC-5/60
  - PLC-5/80

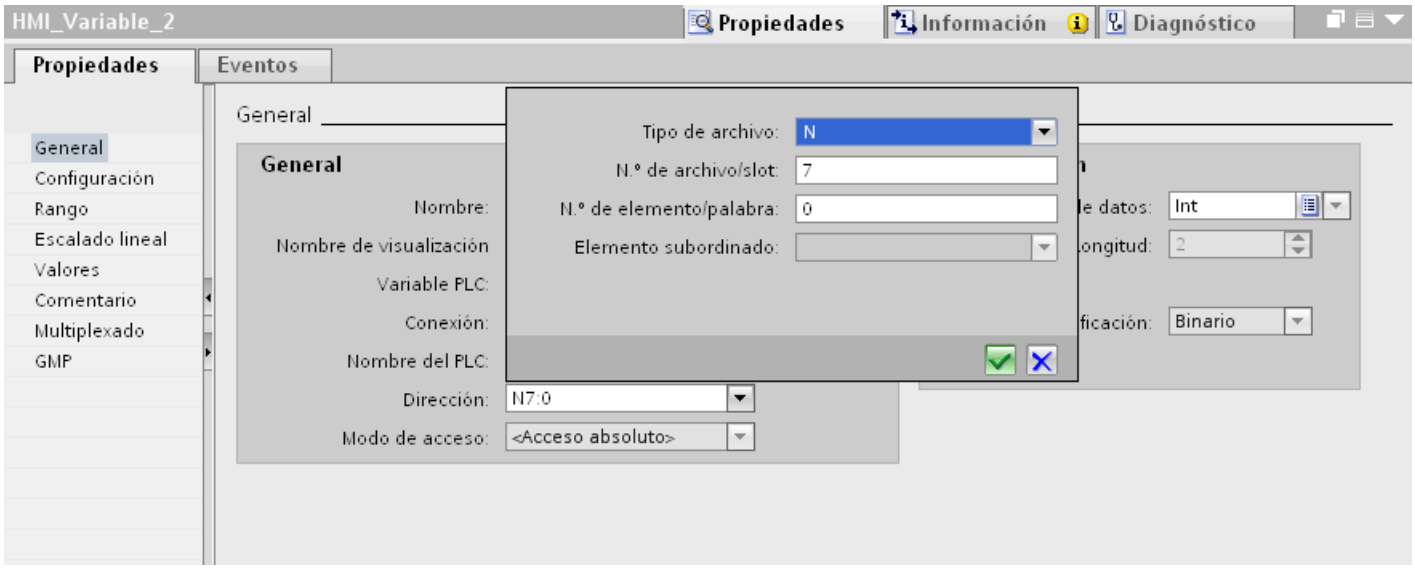
Direccionamiento

Direccionamiento

El direccionamiento se introduce en el driver de comunicación Allen-Bradley DF1 en el siguiente orden:

- Tipo de operando
- Número de archivo
- Número de elemento

- Elemento subordinado
- Número de bit



La dirección aparece entonces en el siguiente formato sin espacios vacíos:

- Tipo de archivo número de archivo: Número de elemento. Elemento subordinado
- P. ej., T8:2.ACC

Tipo de operando

Para "Tipo de operando" se dispone de las opciones siguientes:

- I
- O
- S
- B
- T
- C
- R
- N
- A
- D sólo en el tipo de CPU PLC5

Número de archivo

En "Número de archivo" se selecciona la cantidad entre dos valores límite:

- Límite inferior
- Límite superior

Los valores límite dependen del tipo de archivo seleccionado.

Elemento subordinado

Se puede seleccionar un elemento subordinado si se ha seleccionado uno de los siguientes tipos de archivo:

- R
- C
- T

Áreas de direcciones para Allen-Bradley DF1

MicroLogix

Áreas de direcciones	Tipos de datos					
	Bool	Int	UInt	DInt	UDInt	Real
N	N7:0/0 - N255:255/15	N7:0 - N255:255	N7:0 - N255:255	N7:0 - N255:254	N7:0 - N255:254	N7:0 - N255:254
F	--	--	--	--	F8:0 - F255:255	--
R	R6:0.EN - R255:255.ER - R255:255.DN - R255:255.FD - R255:255.IN - R255:255.EU - R255:255.EM - R255:255.UL	R6:0.LEN - R255:255.POS	R6:0.LEN - R255:255.POS	--	--	--
C	C5:0.CU - C255:255.CD - C255:255.DN - C255:255.OV - C255:255.UN	C5:0.PRE - C255:255.ACC	C5:0.PRE - C255:255.ACC	--	--	--
T	T4:0.DN - T255:255.TT - T255:255.EN	T4:0.PRE - T255:255.ACC	T4:0.PRE - T255:255.ACC	--	--	--
B	B3:0/0 - B255:255/15	--	B3:0 - B255:255	--	--	--
S	S2:0/0 - S2:65/15	S2:0 - S2:65	--	--	--	--
I	I0:0/0 - I38:255/15	--	I0:0 - I38:255	--	--	--
O	O0:0/0 - O38:255/15	--	O0:0/0 - O38:255	--	--	--

SLC500

Áreas de direcciones	Tipos de datos					
	Bool	Int	UInt	DInt	UDInt	Real
N	N7:0/0 - N255:255/15	N7:0 - N255:255	N7:0 - N255:255	N7:0 - N255:254	N7:0 - N255:254	N7:0 - N255:254
R	R6:0.EN - R255:255.ER - R255:255.DN - R255:255.FD - R255:255.IN - R255:255.EU - R255:255.EM - R255:255.UL	R6:0.LEN - R255:255.POS	R6:0.LEN - R255:255.POS	--	--	--
C	C5:0.CU - C255:255.CD - C255:255.DN - C255:255.OV - C255:255.UN	C5:0.PRE - C255:255.ACC	C5:0.PRE - C255:255.ACC	--	--	--
T	T4:0.DN - T255:255.TT - T255:255.EN	T4:0.PRE - T255:255.ACC	T4:0.PRE - T255:255.ACC	--	--	--
B	B3:0/0 - B255:255/15	--	B3:0 - B255:255	--	--	--
S	S2:0/0 - S2:15/15	S2:0 - S2:15	--	--	--	--
I	I0:0/0 - I38:255/15	--	I0:0 - I38:255	--	--	--
O	O0:0/0 - O38:255/15	--	O0:0 - O38:255	--	--	--

SLC501/502

Áreas de direcciones	Tipos de datos					
	Bool	Int	UInt	DInt	UDInt	Real
N	N7:0/0 - N255:255/15	N7:0 - N255:255	N7:0 - N255:255	N7:0 - N255:254	N7:0 - N255:254	N7:0 - N255:254
R	R6:0.EN - R255:255.ER - R255:255.DN - R255:255.FD - R255:255.IN - R255:255.EU - R255:255.EM - R255:255.UL	R6:0.LEN - R255:255.POS	R6:0.LEN - R255:255.POS	--	--	--

Áreas de direcciones	Tipos de datos					
	Bool	Int	UInt	DInt	UDInt	Real
C	C5:0.CU - C255:255.CD - C255:255.DN - C255:255.OV - C255:255.UN	C5:0.PRE - C255:255.ACC	C5:0.PRE - C255:255.ACC	--	--	--
T	T4:0.DN - T255:255.TT - T255:255.EN	T4:0.PRE - T255:255.ACC	T4:0.PRE - T255:255.ACC	--	--	--
B	B3:0/0 - B255:255/15	--	B3:0 - B255:255	--	--	--
S	S2:0/0 - S2:32/15	S2:0 - S2:32	--	--	--	--
I	I0:0/0 - I38:255/15	--	I0:0 - I38:255	--	--	--
O	O0:0/0 - O38:255/15	--	O0:0 - O38:255	--	--	--

PLC5

Áreas de direcciones	Tipos de datos						
	Bool	Int	UInt	DInt	UDInt	Real	ASCII
N	N3:0/0 - N999:999/15	N3:0 - N999:999	N3:0 - N999:999	N3:0 - N999:999	N3:0 - N999:998	N3:0 - N999:998	--
F	--	--	--	--	--	F3:0 - F999:999	--
A	--	--	--	--	--	--	A3:0 - A999:999
R	R3:0.EN - R999:999.ER - R999:999.DN - R999:999.FD - R999:999.IN - R999:999.EU - R999:999.EM - R999:999.UL	R3:0.LEN - R999:999.POS	R3:0.LEN - R999:999.POS	--	--	--	--

Áreas de direcciones	Tipos de datos						
	Bool	Int	UInt	DInt	UDInt	Real	ASCII
C	C3:0.CU - C999:999.CD - C999:999.DN - C999:999.OV - C999:999.UN	C3:0.PRE - C999:999.ACC	C3:0.PRE - C999:999.ACC	--	--	--	--
T	T3:0.DN - T999:999.TT - T999:999.EN	T3:0.PRE - T999:999.ACC	T3:0.PRE - T999:999.ACC	--	--	--	--
B	B3:0/0 - B999:999/15	--	B3:0 - B999:999	--	--	--	--
S	S2:0/0 - S2:127/15	S2:0 - S2:127	--	--	--	--	--
I	I1:0/0 - I1:277/17	--	I1:0 - I1:277	--	--	--	--
O	O0:0/0 - O0:277/17	--	O0:0 - O0:277	--	--	--	--
D	D3:0/0 - D999:999/15	D3:0 - D999:999	D3:0 - D999:999	--	D3:0 - D999:998	--	--

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   - Interfaz
   - Parámetros de transferencia
   - Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún

tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Intercambio de datos

Punteros de área con Allen-Bradley

Punteros de área en las conexiones mediante drivers de comunicación de Allen-Bradley

A través de un puntero de área se accede a un rango de datos del controlador.

Encontrará más información sobre los punteros de área y su configuración en el capítulo "Intercambio de datos mediante punteros de área".

Particularidades de las conexiones mediante Allen-Bradley EtherNet/IP

Puede configurar los siguientes punteros de área

Puntero de área	Allen-Bradley EtherNet/IP	Allen-Bradley DF1
Número de imagen	sí	sí
Fecha/hora	sí	sí
Fecha/hora PLC	sí	sí
Coordinación	sí	sí
Identificador del proyecto	sí	sí
Petición de control	sí	sí
Registro	sí	sí

Limitaciones de Allen-Bradley Ethernet/IP

La configuración de punteros de área tiene las siguientes limitaciones.

Tipo de CPU	Tipos de datos	Tipos de archivo
ControlLogix, CompactLogix	Int, UInt	--
SLC, MicroLogix	Int, UInt	N, B

Limitaciones de Allen-Bradley DF1

La configuración de punteros de área tiene las siguientes limitaciones.

Tipo de CPU	Tipos de datos	Tipos de archivo
MicroLogix	--	N, O, I, B
SLC50x	--	N, O, I, B
PLC5	--	N, O, I, B

Consulte también

Intercambio de datos mediante punteros de área (Página 9118)

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos con conexiones no integradas

Para configurar avisos tales como advertencias, mensajes de error y acuses se requieren varios pasos.

- Paso 1: crear variables
- Paso 2: configurar avisos
- Paso 3: configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- Tipos de datos de la variable
- Direccionamiento de la variable
- Modo de contaje de las posiciones de bit

Restricciones

Como variable de disparo de avisos de bit sólo se admiten las variables en el "Tipo de archivo" "N", "O", "I", "S" y "B". Las variables sólo se soportan para los tipos de datos "Int" y "UInt".

Tipos de datos

Para las conexiones con drivers de comunicación Allen-Bradley se soportan los tipos de datos siguientes:

Drivers de comunicación	Controlador	Tipos de datos admisibles	
		Avisos de bit	Avisos analógicos
Allen-Bradley DF1	SLC500, SLC501, SLC502, SLC503, SLC504, SLC505, PLC5, MicroLogix	Int, UInt	Int, UInt, Long, ULong, Real
Allen-Bradley EtherNet/IP	ControlLogix, CompactLogix, SLC, Micrologix	Int, UInt	SInt, USInt, Int, UInt, DInt, UDInt, Real

Modo de contaje de las posiciones de bit

Para conexiones con drivers de comunicación Allen-Bradley rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte izquierdo								Byte derecho							
En controladores Allen-Bradley	15							8	7							0
Configure en WinCC:	15							8	7							0

Consulte también

El sistema de avisos de WinCC (Página 8501)

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

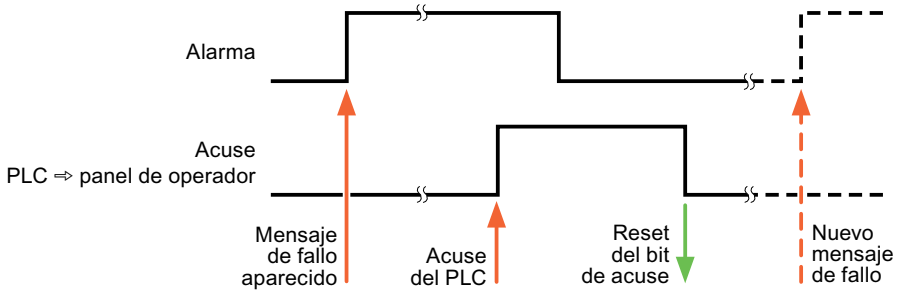
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el

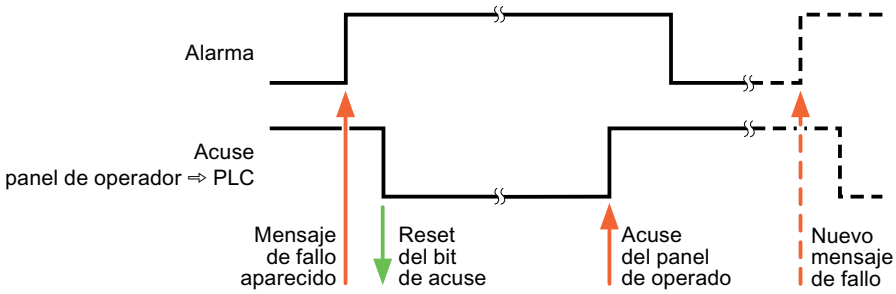
controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



Mitsubishi

Drivers de comunicación Mitsubishi

Introducción

Este apartado describe la comunicación entre un panel de operador y un controlador que utilizan drivers de comunicación Mitsubishi.

Se soportan los siguientes drivers de comunicación:

- Mitsubishi MC TCPI/IP
- Mitsubishi FX

Intercambio de datos

El intercambio de datos se realiza mediante variables o punteros de área.

- Variables
  El controlador y el panel de operador intercambian los datos a través de valores de proceso. Configure variables que señalen a direcciones en el controlador. El panel de operador lee y muestra el valor de la dirección indicada. El operador también puede introducir un valor en el panel. Dicho valor se escribe luego en la dirección en el controlador.
- Punteros de área
  Los punteros de área sirven para intercambiar datos especiales y se configuran únicamente si se utiliza este tipo de datos.

Mitsubishi MC TCP/IP

Configurar una conexión mediante Mitsubishi MC TCPI/IP

Introducción

Una conexión con un controlador que tenga un driver de comunicación Mitsubishi MC TCPI/IP se configura en el editor "Conexiones" del panel de operador.

Las interfaces Ethernet reciben nombres distintos en función del panel de operador.

Ejemplo: la interfaz PROFINET corresponde a la interfaz Ethernet

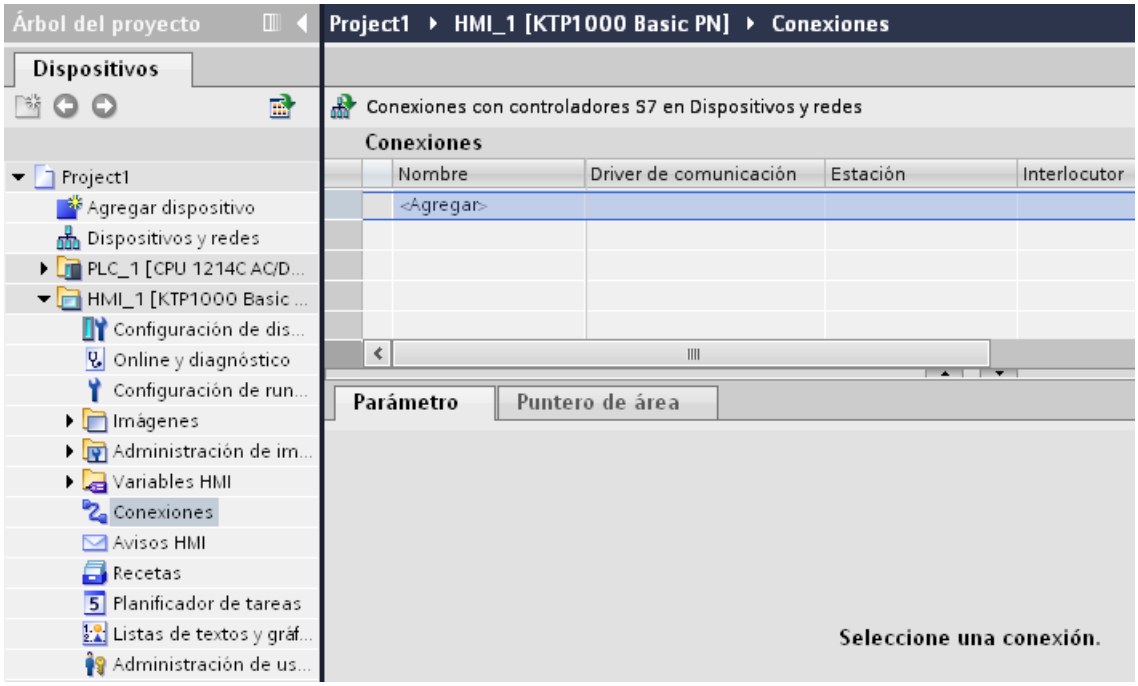
Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador.

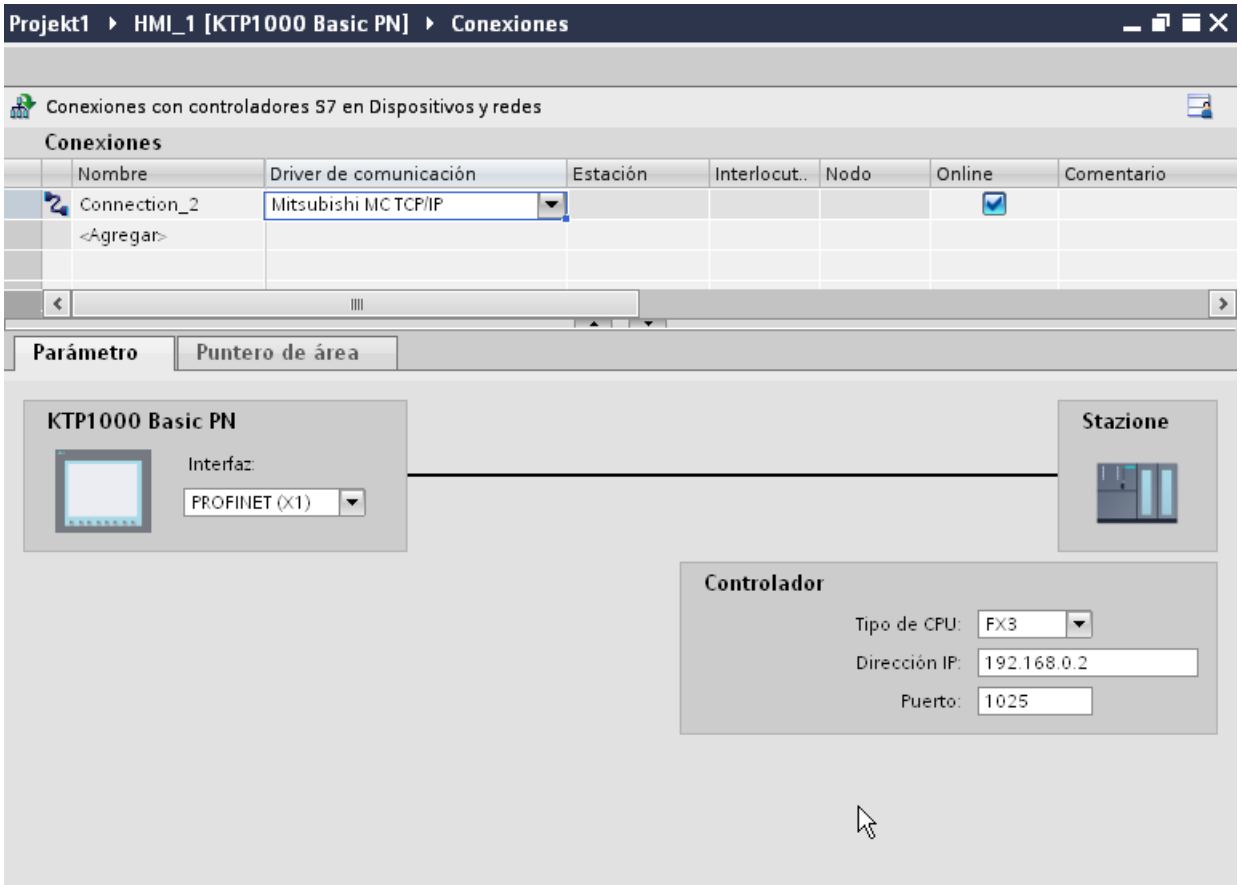
Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".

3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el driver "Mitsubishi MC TCPI/IP" en la columna "Driver de comunicación".



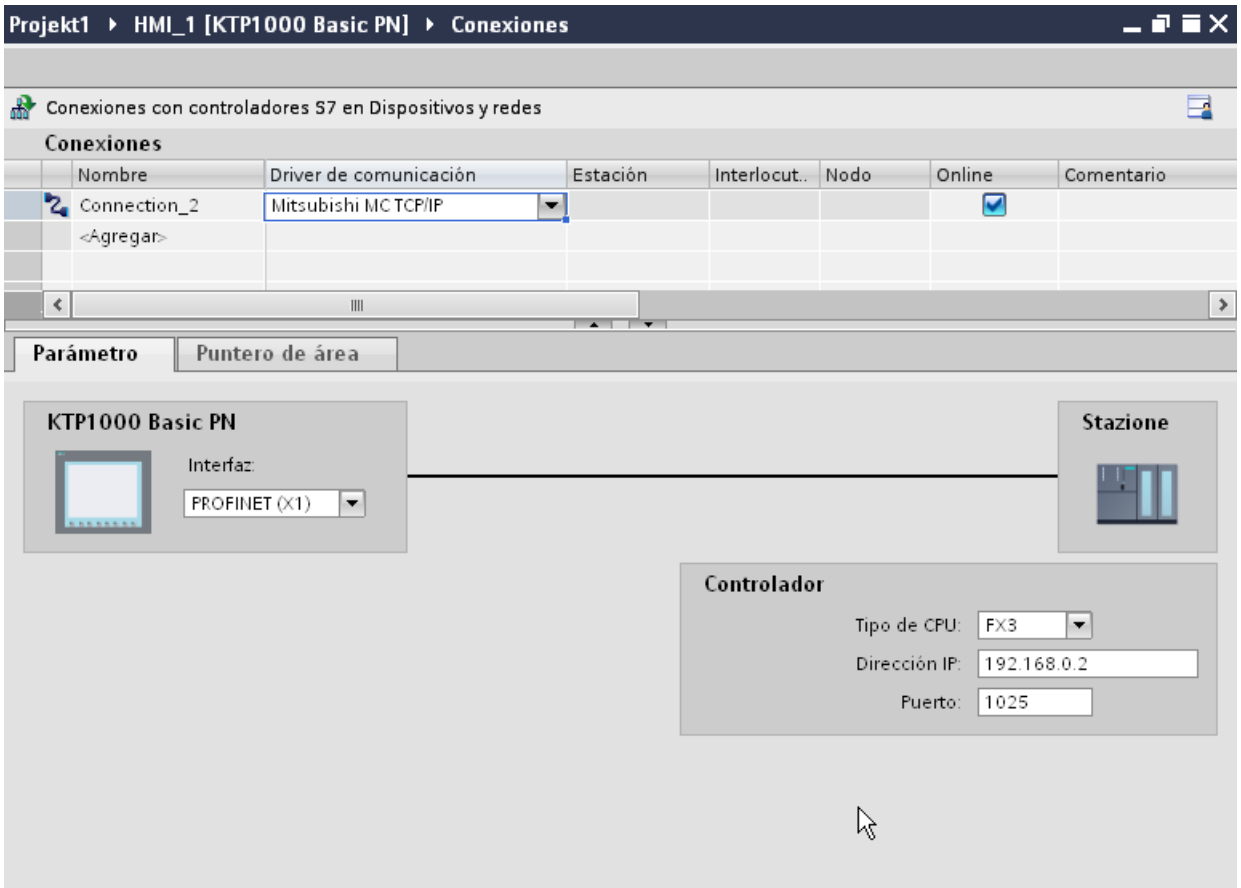
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Mitsubishi MC TCP/IP)

Parámetros que deben ajustarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador" y "Controlador", según sea la interfaz utilizada.



Parámetros del panel de operador

En la ventana de inspección "Parámetros" sólo se puede seleccionar una interfaz para el panel de operador. En función del panel de operador habrá diferentes interfaces disponibles.

Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC. Al transferir el proyecto se transfiere la dirección IP al panel de operador.

Nota

Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

Si activa "Obtener dirección IP por otra vía", en la siguiente carga la dirección IP establecida previamente se conserva en el Control Panel.

La dirección IP del panel de operador se configura del siguiente modo:

1. Haga clic en el panel de operador.
2. Abra el editor "Configuración de dispositivos".

3. Haga clic en la interfaz Ethernet.
4. Asigne la dirección IP en la ventana de inspección en:
"General > Interfaz PROFINET > Direcciones Ethernet"

Parámetros del controlador

- Tipo de CPU
En "Tipo de CPU" se especifica el tipo de controlador al que está conectado el panel de operador.
Son posibles las siguientes entradas:
–FX3
–Q
Si se selecciona el tipo de CPU FX3, se utilizará el protocolo MC de Mitsubishi "1E" y con el tipo de CPU "Q", se utilizará "3E".
Siempre se utiliza la variante de protocolo "Código binario".

Nota

Si se modifica el tipo de CPU en una conexión configurada, hay que revisar las variables que tienen las siguientes propiedades:

- Operandos que no existen para el nuevo tipo de CPU, como "W", "B", "F".
- Entradas y salidas con direccionamiento diferente (hexadecimal/octal).
- Direcciones mayores que el área de direcciones admitido del nuevo tipo de CPU.

- Dirección IP
Ajuste la dirección IP o el nombre de host del módulo Ethernet/IP del controlador. En un Basic Panel sólo puede utilizarse la dirección IP.
- Puerto
Ajuste el número de puerto del módulo del controlador.

Conectar el panel de operador con el controlador

Conexiones mediante Mitsubishi MC TCP/IP

Conexión

El panel de operador puede conectarse al controlador Mitsubishi utilizando los componentes siguientes:

- Una red Ethernet existente en la que también se encuentran los controladores
- Un cable Ethernet cruzado (Cross-Over) directamente a la interfaz Ethernet de la CPU o del módulo de comunicación

El acoplamiento entre el panel de operador y un controlador Mitsubishi se limita principalmente a la conexión física del panel de operador. No se requieren bloques especiales para el acoplamiento en el controlador.

Conecte el panel de operador de uno o más controladores de las series Q y/o FX3. El panel de operador se conecta a través de las siguientes interfaces:

- Interfaz de comunicación OnBoard
- Modulo de comunicación adecuado liberado hacia el controlador

Nota

Comportamiento de timeout en TCP/IP (Ethernet)

Debido a que se utiliza el protocolo TCP/IP, las conexiones interrumpidas se detectan a lo sumo al cabo de aprox. un minuto. Si no se solicitan variables (p. ej. si la imagen actual no contiene una variable de salida), la interrupción de la conexión no se detectará con seguridad.

Configure un puntero de área "Coordinación" para cada controlador. Este ajuste garantiza que, aun en el caso descrito, la interrupción de la conexión se detecte al cabo de aprox. dos minutos.

Tipos de comunicación

Tipos de comunicación habilitados

- Sólo válido para Mitsubishi FX(protocolo PG):
  el acoplamiento punto a punto de un panel de operador a una CPU Mitsubishi FX activada a través de Mitsubishi FX ha sido comprobado y activado por Siemens AG.
- Sólo válido para Mitsubishi MC TCP/IP:
  los siguientes tipos de comunicación han sido comprobados y activados:
  - acoplamiento punto a punto a los controles activados
  - acoplamiento a varios puntos de un panel de operador con hasta 4 controles, con los controles correspondientes activados. Se puede mezclar los tipos de CPU (FX3 y Q).

Nota

el panel de control es el cliente y el control tiene que trabajar como servidor.

Controladores acoplables

Se puede realizar acoplamientos para los siguientes controladores de Mitsubishi:

Controlador	Mitsubishi FX (Protocolo PG)	Mitsubishi MC TCP/IP
MELSEC FX1n, FX2n	sí	no
MELSEC FX3U, FX3UC, FX3G con módulo de comunicación FX3U-ENET	no	sí
MELSEC Sistema Q • Serie Q con el módulo de comunicación QJ71E71-100 • CPU QnUDEH con interfaz Ethernet integrada	no	sí

Parametrización de los módulos de comunicación

Controladores FX3

Procedimiento

1. Inicie FX-Configurator.
2. Seleccione el módulo.
3. Haga los siguientes ajustes en el cuadro de diálogo "Operational settings":
   - Communication data code:
     Binary code
   - Initial timing:
     Always wait for OPEN
   - IP address:
     Dirección IP
   - Send frame setting:
     Ethernet(V2.0)
   - TCP Existence confirmation setting:
     Use the Ping

4. Haga los siguientes ajustes en el cuadro de diálogo "Open settings":
   - Protocol:
     TCP
   - Open system:
     Unpassive
   - Fixed buffer:
     Receive
   - Fixed buffer communication procedure:
     Procedure exist(MC)
   - Pairing open
     Disable
   - Existence confirmation
     No confirm
   - Host station Port No. (DEC)
     Número de puerto

   Nota

   El número de puerto elegido en el módulo de comunicación debe coincidir con el de WinCC. Para cada panel de operador conectado debe parametrizarse una conexión con un número de puerto.

   El número de puerto debe indicarse en valores decimales.

5. Confirme la configuración estándar de los cuadros de diálogo siguientes.

Los parámetros "N.º de red" y "N.º de estación" no son relevantes para el acoplamiento y pueden seleccionarse como se prefiera.

Controladores Q

Procedimiento

1. Haga clic en "Editar parámetros de red".
2. Elija el tipo de red:
   - Ethernet
     Ni el número de red ni los números de grupo/estación se evalúan y por lo tanto se pueden asignar libremente.

3. Haga los siguientes ajustes en el cuadro de diálogo "Operational settings":
   - Communication data code:
     Binary code
   - Initial timing:
     Always wait for OPEN
   - IP address:
     Dirección IP
   - Send frame setting:
     Ethernet(V2.0)
   - Permitir la escritura durante RUN
4. Haga los siguientes ajustes en el cuadro de diálogo "Open settings":
   - Protocol:
     TCP
   - Open system:
     Unpassive
   - Pairing open
     Disable
   - Existence confirmation
     No confirm
   - Host station Port No. (HEX)
     Port-Nummer

Nota

El número de puerto elegido en el módulo de comunicación debe coincidir con el de WinCC. Para cada panel de operador conectado debe parametrizarse una conexión con un número de puerto.

El número de puerto debe indicarse en valores hexadecimales.

Puerto Ethernet interno de la CPU Q0xUDEH

Procedimiento

1. Haga los siguientes ajustes en el cuadro de diálogo "Internal Ethernet Port":
   - IP address:
     Dirección IP
   - Communication data code:
     Binary code
   - Conectar cambios online
2. Haga los siguientes ajustes en el cuadro de diálogo "Open settings":
   - Protocol:
     TCP
   - Open system:
     MC-Protocol
   - Host station Port No. (HEX)
     Número de puerto

   Nota

   El número de puerto elegido en el módulo de comunicación debe coincidir con el de WinCC. Para cada panel de operador conectado debe parametrizarse una conexión con un número de puerto.

Características de rendimiento de la comunicación

Tipos de datos admisibles para Mitsubishi MC TCPI/IP

Tipos de datos admisibles

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Tipo de operando	Longitud
4 bit block	M, X, Y, B, F	1 byte
8 bit block	M, X, Y B, F	1 byte
12 bit block	M, X, Y B, F	2 bytes
16 bit block	M, X, Y B, F	2 bytes
20 bit block	M, X, Y B, F	4 bytes
24 bit block	M, X, Y B, F	4 bytes
28 bit block	M, X, Y B, F	4 bytes
32 bit block	M, X, Y B, F	4 bytes
Bool	M, D, X, Y B, F	1 bit

Tipo de datos	Tipo de operando	Longitud
DInt	D, W	4 bytes
DWord	D, C, W	4 bytes
Int	D, W	2 bytes
Real 1)	D, W	4 bytes
String 1)	D	de 1 a 80 caracteres
Word	D, T, C, W	2 bytes

1) Los tipos de datos "String" y "Real" no están disponibles en todas las CPU.
2) Los tipos de operando B, F y W sólo están disponibles para el tipo de CPU "Q".

Nota

Nota relativa a los accesos de escritura:

Sólo es posible escribir variables cuando se ha seleccionado "Conectar cambios online" o "Permitir escribir en RUN" en la parametrización de los módulos de comunicación Mitsubishi.

En el tipo de datos "Bool" del tipo de operando "D" toda la palabra se vuelve a escribir en el controlador después de modificar el bit indicado. No se comprueba si entretanto se han modificado otros bits en la palabra. Por este motivo, el controlador sólo puede leer la palabra indicada.

Nota

Los elementos de matriz en campos E/S no están permitidos para una comunicación con un controlador Mitsubishi.

Tipos de CPU soportados para Mitsubishi MC TCPI/IP

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Mitsubishi MC TCP/IP.

- FX3 series
  - FX 3G / FX 3G with communication modul FX3U-ENET
  - FX 3U / FX 3U with communication modul FX3U-ENET
  - FX 3UC / FX 3UC with communication modul FX3U-ENET
- Q series
  - Q-Series with QJ71E71-100 communication module
- iQ series / QnUD
  - QnUDEHCPU with built in ethernet module

Direcciones en Mitsubishi MC TCP/IP

Áreas de direcciones en las conexiones mediante Mitsubishi MC TCP/IP

Para las CPU de las diferentes series existen límites distintos en cuanto a las áreas de direcciones que pueden consultarse en los manuales de MITSUBISHI Computerlink.

Ejemplos de los límites de áreas de direcciones en función del formato de comunicación y CPU:

Denominación	Tipo de operando	Dirección máx. FX3	Dirección máx. Serie Q
Output/Input	Y/X	Octal X/Y 0 - 777	HEX X/Y 0 - 7FF
Marca	M	M0 - M3071 y M8000 - M8255	M/L/S 0 - 8191
Registro de datos	D	D0 - 7999 D8000 - D8255	D0 - 8191 D9000 - D9255 se convierte en SD1000 - SD1255
Counter	C	C0 - 255	C0 - 1023
Temporizador	T	T0 - 255	T0 - 2047
Registro de enlace	W	--	Hex: W0 - FFF
Marca de enlace	B	--	Hex: B0 - FFF
Marca de error	F	--	F0 - 2047

Áreas de direcciones para Mitsubishi MC TCP/IP

FX0

Áreas de direcciones	Tipos de datos												
	Bool	Word	DWord	Real	String	4 bit block	8 bit block	12 bit block	16 bit block	20 bit block	24 bit block	28 bit block	32 bit block
M	M0 - M9999	--	--	--	--	M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
D	D0.0 - D999.15	D0 - D999	D0 - D998	D0 - D998	D0 - D998	--	--	--	--	--	--	--	--
T		T0 - T255	--	--	--	--	--	--	--	--	--	--	--
C-16-Bit	--	C-16-Bit 0 - C-16-Bit 199	--	--	--	--	--	--	--	--	--	--	--

Áreas de direcciones	Tipos de datos												
	Bool	Word	DWord	Real	String	4 bit block	8 bit block	12 bit block	16 bit block	20 bit block	24 bit block	28 bit block	32 bit block
C-32-Bit	--	--	C-32-Bit 200 - C-32-Bit 255	--	--	--	--	--	--	--	--	--	--
X	X0 - X255	--	--	--	--	X0 - X252	X0 - X248	X0 - X244	X0 - X240	X0 - X236	X0 - X232	X0 - X228	X0 - X224
Y	Y0 - X255	--	--	--	--	Y0 - Y252	Y0 - Y248	Y0 - Y244	Y0 - Y240	Y0 - Y236	Y0 - Y232	Y0 - Y228	Y0 - Y224

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   – Interfaz
   – Parámetros de transferencia
   – Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Mitsubishi FX

Configurar una conexión mediante Mitsubishi FX

Introducción

Una conexión con un controlador que tenga un driver de comunicación Mitsubishi FX se configura en el editor "Conexiones" del panel de operador.

El protocolo Mitsubishi FX también se denomina protocolo Mitsubishi PG.

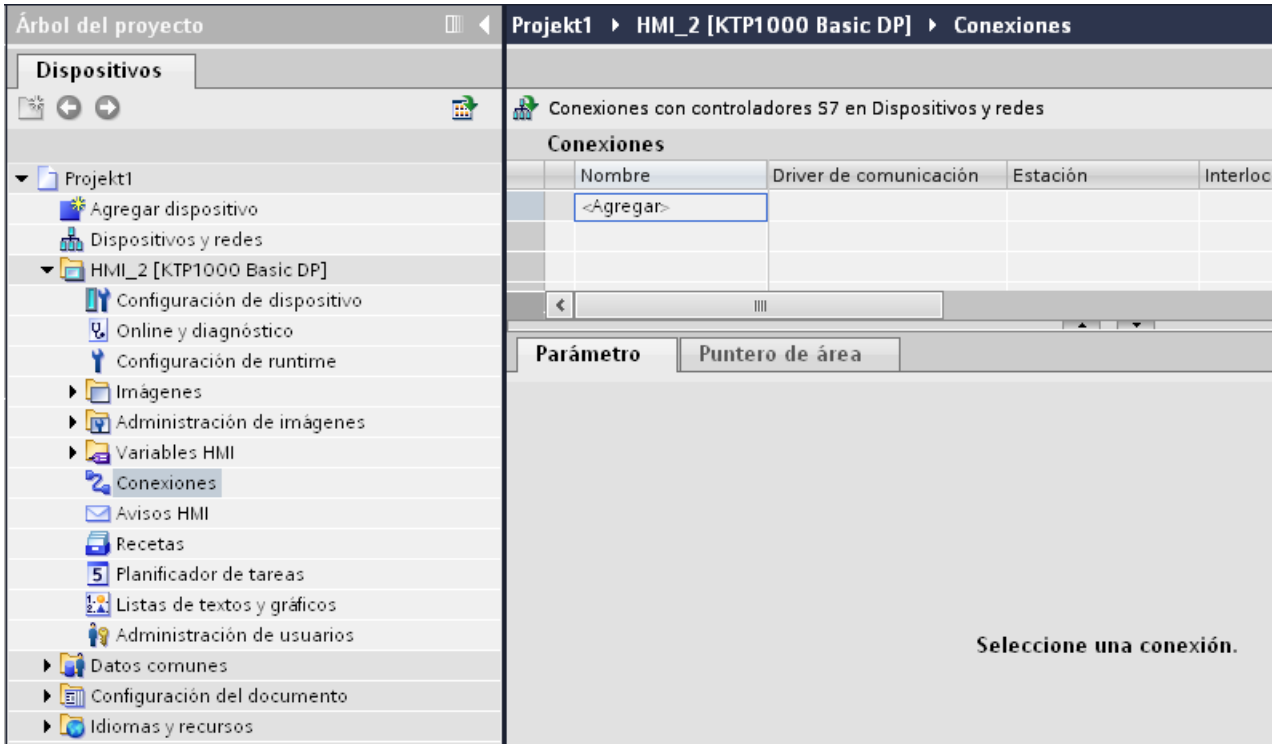
Las interfaces reciben nombres distintos en función del panel de operador.

Requisitos

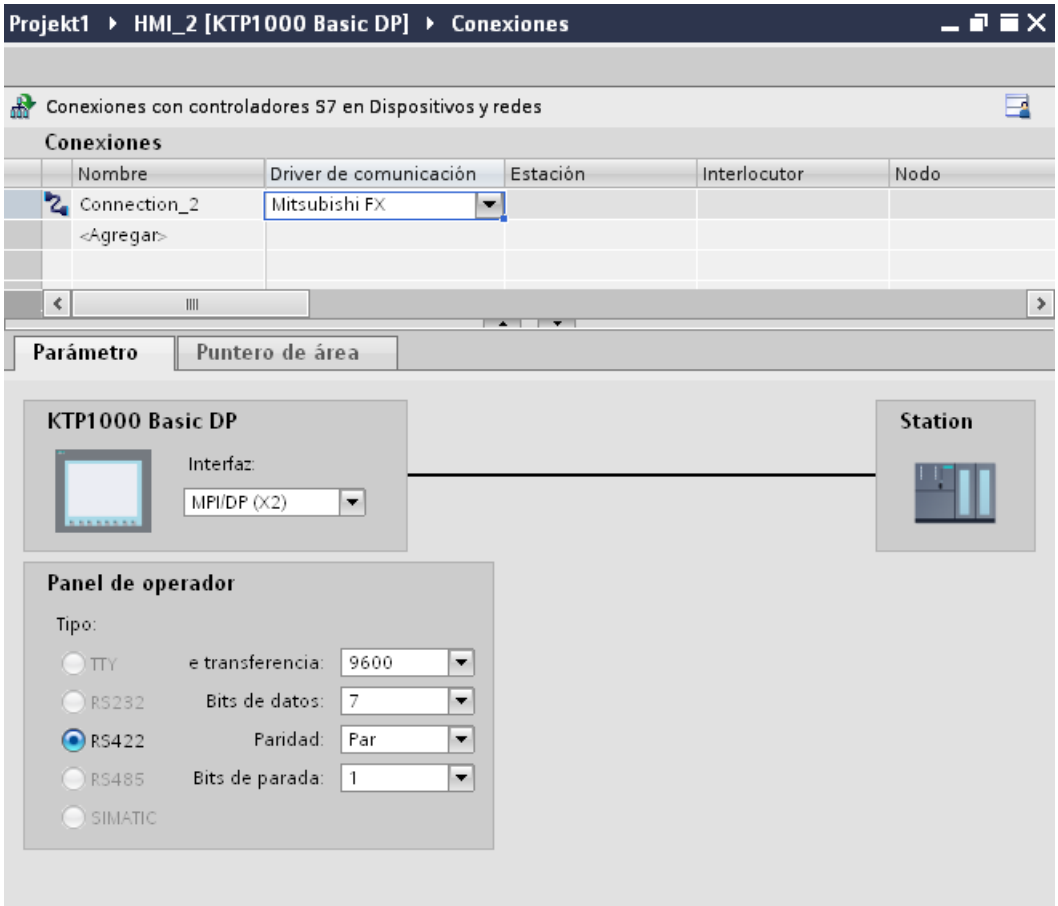
- Hay un proyecto abierto.
- Se ha creado un panel de operador.

Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".
3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el controlador "Mitsubishi FX" en la columna "Driver de comunicación".



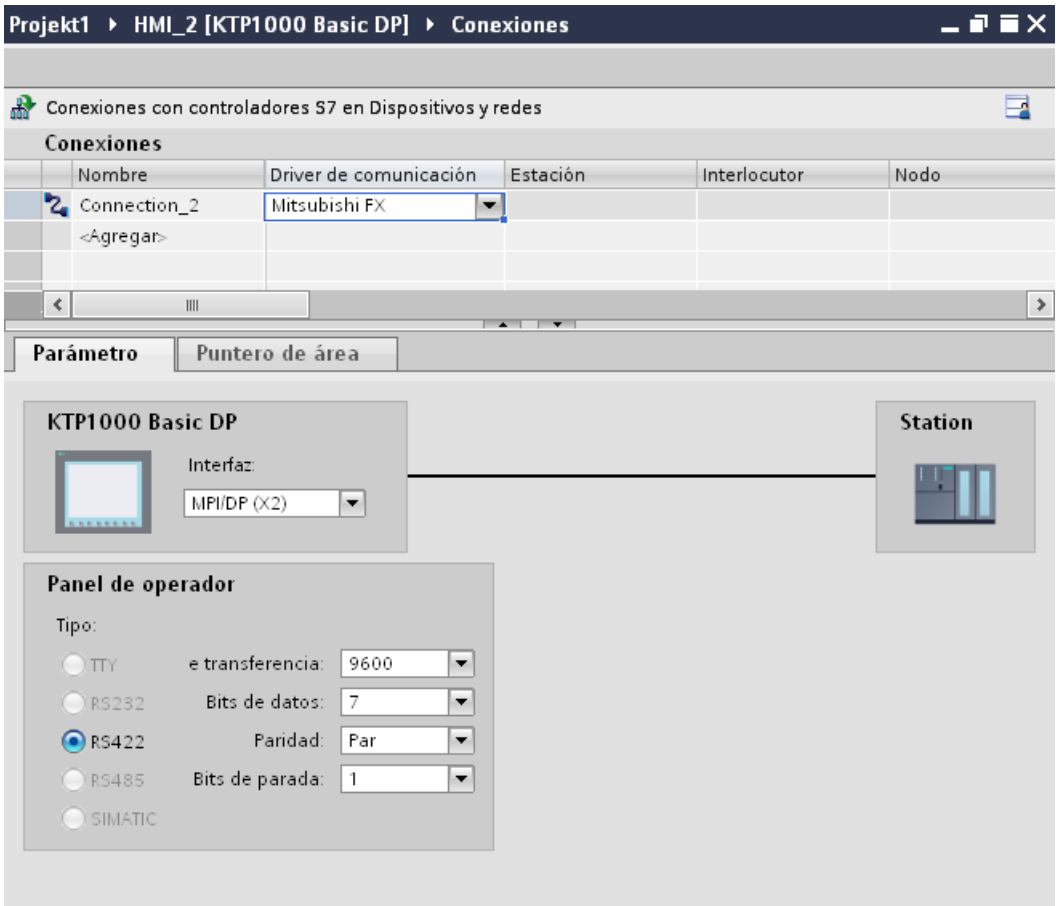
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Mitsubishi FX)

Parámetros que deben configurarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador", "Red" y "Controlador", según sea la interfaz utilizada.



Parámetros del panel de operador

En la ventana de inspección "Parámetros" se puede seleccionar una interfaz para el panel de operador. En función del panel de operador habrá diferentes interfaces disponibles.

- "Tipo"
  Define la conexión física utilizada.

  Nota

  Cuando se deba utilizar la interfaz IF1B, hay que conmutar además los datos de recepción RS422 y la señal RTS mediante el interruptor 4 DIL de la parte posterior del panel de operador.

Parámetros del controlador

- Velocidad de transferencia: En "Velocidad de transferencia" se selecciona la velocidad de transferencia entre el panel de operador y el controlador. Seleccione la velocidad de transferencia 9600.
- Bits de datos: Seleccione "7 bits" en "Bits de datos".

- Paridad: Seleccione "Par" en "Paridad".
- Bits de parada: Seleccione "1 bit" en "Bits de parada".

Conectar el panel de operador con el controlador

Tipos de comunicación

Tipos de comunicación habilitados

- Sólo válido para Mitsubishi FX(protocolo PG):
  el acoplamiento punto a punto de un panel de operador a una CPU Mitsubishi FX activada a través de Mitsubishi FX (Protocolo PG: Protocol for access to the program and memory elements of the FX series PC CPU version V1.21 and after) ha sido comprobado y activado por Siemens AG.
- Sólo válido para Mitsubishi MC TCP/IP:
  los siguientes tipos de comunicación han sido comprobados y activados:
  – acoplamiento punto a punto a los controles activados
  – acoplamiento a varios puntos de un panel de operador con hasta 4 controles, con los controles correspondientes activados. Se puede mezclar los tipos de CPU (FX3 y Q).

  Nota

  El panel de control es el cliente y el control tiene que trabajar como servidor.

Controladores acoplables

Se puede realizar acoplamientos para los siguientes controladores de Mitsubishi:

Controlador	Mitsubishi FX (Protocolo PG)	Mitsubishi MC TCP/IP
MELSEC FX1n, FX2n	sí	no
MELSEC FX3U, FX3UC, FX3G con módulo de comunicación FX3U-ENET	no	sí
MELSEC Sistema Q • Serie Q con el módulo de comunicación QJ71E71-100 • CPU QnUDEH con interfaz Ethernet integrada	no	sí

Conexiones mediante Mitsubishi FX

Conexión

Conecte el panel de operador a la interfaz de programación de la CPU (RS 422) (consulte la documentación del controlador).

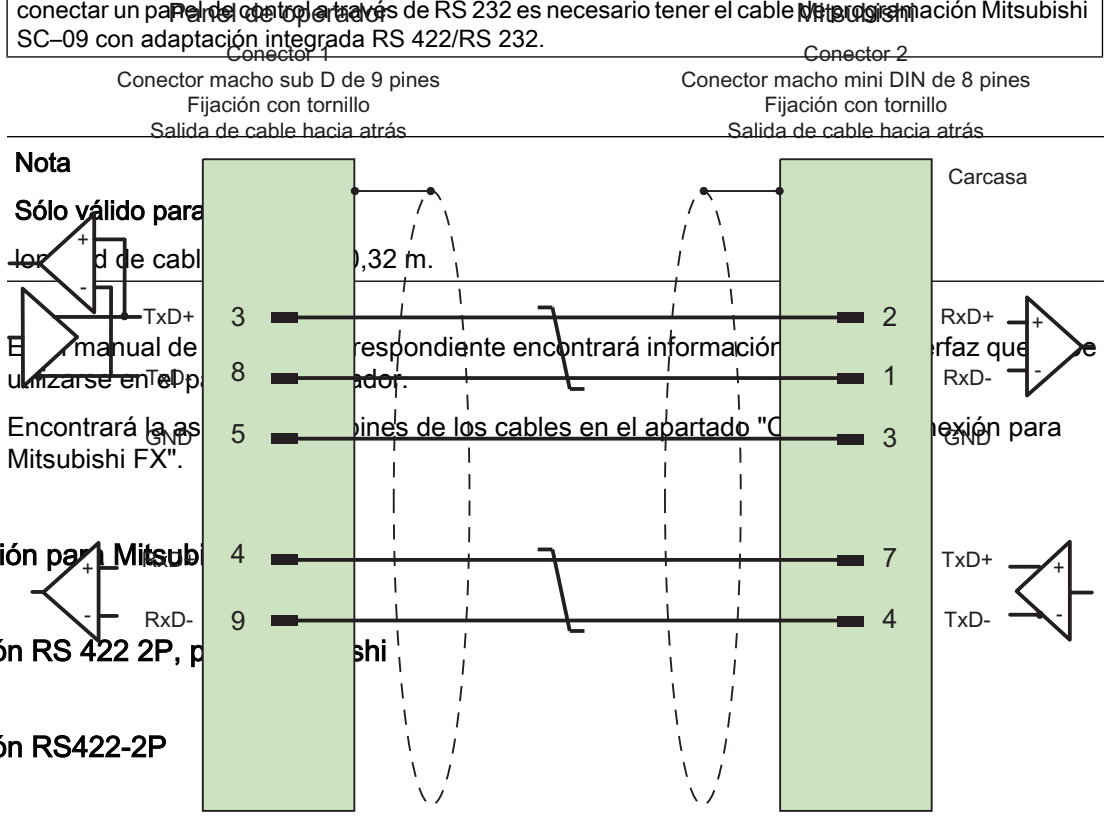
El acoplamiento entre el panel de operador y el control de Mitsubishi se limita principalmente a ajustar los parámetros de la interfaz. No se requieren bloques especiales para el acoplamiento en el controlador.

Cable de conexión

Para conectar el panel de operador al controlador están disponibles los siguientes cables de conexión:

Interfaz en el panel de operador o adaptador	Controlador Mitsubishi Electric mediante protocolo FX
	FX1n, Fx2n, Mini DIN, de 8 polos
RS 232, de 9 polos	Mitsubishi SC-09 [1]
RS 422, de 9 polos	Cable de conexión RS422-2P

[1] Ya que los controladores de Mitsubishi se comunican de forma estándar a través de RS 422, para conectar un panel del control a través de RS 232 es necesario tener el cable de programación Mitsubishi SC–09 con adaptación integrada RS 422/RS 232.



Nota

Sólo válido para

lon d de cabl),32 m.

E manual de respondiente encontrará información rfaz que e utilizarse en el p dor.

Encontrará la as pines de los cables en el apartado "C nexión para Mitsubishi FX".

Cables de conexión par Mitsub

Cable de conexión RS 422 2P, p shi

Cable de conexión RS422-2P

Blindaje conectado a la carcasa en ambos lados y con gran superficie
Cable: 3 x 2 x 0,14 mm^2, blindado,
longitud máx. 500 m

Características de rendimiento de la comunicación

Tipos de datos admisibles para Mitsubishi FX

Tipos de datos permitidos

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Tipo de operando	Longitud
4 bit block	M, X, Y	1 byte
8 bit block	M, X, Y	1 byte
12 bit block	M, X, Y	2 bytes
16 bit block	M, X, Y	2 bytes
20 bit block	M, X, Y	4 bytes
24 bit block	M, X, Y	4 bytes
28 bit block	M, X, Y	4 bytes
32 bit block	M, X, Y	4 bytes
Bool	D, M, X, Y	1 bit
DWord	D, C 32 bits	4 bytes
Real	D	4 bytes
String	D	de 1 a 50 caracteres
Word	D, T, C 16 bits	2 bytes

Nota

Nota relativa a los accesos de escritura:

En el tipo de datos "Bool" del tipo de operando "D" toda la palabra se vuelve a escribir en el controlador después de modificar el bit indicado. No se comprueba si entretanto se han modificado otros bits en la palabra. Por este motivo, el controlador sólo puede leer la palabra indicada.

Nota

Los elementos de matriz en campos E/S no están permitidos para una comunicación con un controlador Mitsubishi.

Tipos de CPU soportados para Mitsubishi FX

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Mitsubishi FX.

- FX1 series
  - FX1n
- FX2 series
  - FX2n

Áreas de direcciones para Mitsubishi FX

FX3

Áreas de direcciones	Tipos de datos														
	Bool	Int	Word	DInt	DWord	Real	String	4 bit block	8 bit block	12 bit block	16 bit block	20 bit block	24 bit block	28 bit block	32 bit block
M	M0 - M9999	--	--	--	--	--	--	M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
D	D0.0 - D9999.15	D0 - D9999	D0 - D9999	D0 - D9999	D0 - D9999	D0 - D9999	D0 - D9999	--	--	--	--	--	--	--	--
T	--	--	T0 - T999	--	--	--	--	--	--	--	--	--	--	--	--
C	--	C0 - C999	C0 - C999	C0 - C998	C0 - C998	--	--	--	--	--	--	--	--	--	--
X	X0 - X777	--	--	--	--	--	--	X0 - X774	X0 - X770	X0 - X764	X0 - X760	X0 - X754	X0 - X750	X0 - X744	X0 - X740
Y	Y0 - Y777	--	--	--	--	--	--	Y0 - Y774	Y0 - Y770	Y0 - Y764	Y0 - Y760	Y0 - Y754	Y0 - Y750	Y0 - Y744	Y0 - Y740

Q

Áreas de direcciones	Tipos de datos														
	Bool	Int	Word	DInt	DWord	Real	String	4 bit block	8 bit block	12 bit block	16 bit block	20 bit block	24 bit block	28 bit block	32 bit block
M	M0 - M9999	--	--	--	--	--	--	M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
F	F0 - F9999	--	--	--	--	--	--	F0 - F9996	F0 - F9992	F0 - F9988	F0 - F9984	F0 - F9980	F0 - F9976	F0 - F9972	F0 - F9968
B	B0 - BFFFF	--	--	--	--	--	--	B0 - BFFFC	B0 - BFFF8	B0 - BFFF4	B0 - BFFF0	B0 - BFFEC	B0 - BFFE8	B0 - BFFE4	B0 - BFFE0
D	D0.0 - D65534.15	D0 - D65534	D0 - D65534	D0 - D65533	D0 - D65533	D0 - D65533	D0 - D65534	--	--	--	--	--	--	--	--
T	--	--	T0 - T2047	--	--	--	--	--	--	--	--	--	--	--	--
C		C0 - C2047	C0 - C2047	C0 - C2046	C0 - C2046	--	--	--	--	--	--	--	--	--	--
W	--	W0 - WFFFF	W0 - WFFFF	W0 - WFFFE	W0 - WFFFE	W0 - WFFFE	--	--	--	--	--	--	--	--	--
X	X0 - XFFFF	--	--	--	--	--	--	X0 - XFFFC	X0 - XFFF8	X0 - XFFF4	X0 - XFFF0	X0 - XFFEC	X0 - XFFE8	X0 - XFFE4	X0 - XFFE0
Y	Y0 - YFFFF	--	--	--	--	--	--	Y0 - YFFFC	Y0 - YFFF8	Y0 - YFFF4	Y0 - YFFF0	Y0 - YFFEC	Y0 - YFFE8	Y0 - YFFE4	Y0 - YFFE0

FX0

Áreas de direcciones	Tipos de datos												
	Bool	Word	DWord	Real	String	4 bit block	8 bit block	12 bit block	16 bit block	20 bit block	24 bit block	28 bit block	32 bit block
M	M0 - M9999					M0 - M9996	M0 - M9992	M0 - M9988	M0 - M9984	M0 - M9980	M0 - M9976	M0 - M9972	M0 - M9968
D	D0.0 - D999.15	D0 - D999	D0 - D998	D0 - D998	D0 - D998								
T		T0 - T255											
C-16-Bit		C-16-Bit 0 - C-16-Bit 199											
C-32-Bit			C-32-Bit 200 - C-32-Bit 255										
X	X0 - X255					X0 - X252	X0 - X248	X0 - X244	X0 - X240	X0 - X236	X0 - X232	X0 - X228	X0 - X224
Y	Y0 - X255					Y0 - Y252	Y0 - Y248	Y0 - Y244	Y0 - Y240	Y0 - Y236	Y0 - Y232	Y0 - Y228	Y0 - Y224

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   – Interfaz
   – Parámetros de transferencia
   – Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Intercambio de datos

Punteros de área con Mitsubishi

Punteros de área en las conexiones mediante drivers de comunicación Mitsubishi

A través de un puntero de área se accede a un área de datos del controlador.

Encontrará más información sobre los punteros de área y su configuración en el capítulo "Intercambio de datos mediante punteros de área".

Particularidades de las conexiones mediante drivers de comunicación Mitsubishi

Puede configurar los siguientes punteros de área

Puntero de área	Mitsubishi MC TCP/IP	Mitsubishi FX
Número de imagen	Sí	Sí
Fecha/hora	Sí	Sí
Fecha/hora PLC	Sí	Sí
Coordinación	Sí	Sí
Identificación del proyecto	Sí	Sí
Petición del controlador	Sí	Sí
Registro	Sí	Sí

Limitaciones de Mitsubishi MC TCP/IP

La configuración de punteros de área tiene las siguientes limitaciones.

Tipo de CPU	Tipos de datos	Tipo de operando
FX3	Int, Word	D
Q	Int, Word	D

Restricciones de Mitsubishi FX

El tipo de operando D se puede utilizar para configurar punteros de área.

Consulte también

Intercambio de datos mediante punteros de área (Página 9118)

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos con conexiones no integradas

Para configurar avisos tales como advertencias, mensajes de error y acuses se requieren varios pasos.

- Paso 1: crear variables
- Paso 2: configurar avisos
- Paso 3: configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- tipos de datos de la variable
- direccionamiento de la variable
- modo de contaje de las posiciones de bit

Tipos de datos

Para las conexiones con drivers de comunicación Mitsubishi se soportan los siguientes tipos de datos:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
FX1n, FX2n, serie FX3, serie Q, serie iQ	Word, Int [1]	4 bit block, 8 bit block, 12 bit block, 16 bit block, 20 bit block, 24 bit block, 28 bit block, 32 bit block, Word, DWord, Int [1], DInt [1], Real,
[1] No en el driver de comunicación Mitsubishi FX		

Modo de contaje de las posiciones de bit

Para las conexiones con drivers de comunicación Mitsubishi rige el siguiente modo de contaje:

Modo de contaje de las posiciones de bit	Byte izquierdo								Byte derecho							
En controladores Mitsubishi	15							8	7							0
Configure en WinCC:	15							8	7							0

Restricciones para avisos

- Mitsubishi MC TCP/IP
  Como variables de disparo para avisos de bit sólo pueden crearse variables del tipo de operando "D" y de los tipos de datos "Word" e "Int". Para avisos de bit se pueden usar las siguientes variables de matriz (tipo de operando: "D"; tipos de datos: utilizar "ARRAY [x..y] of Word" o "ARRAY [x..y] of Int").
- Mitsubishi FX
  Como variables de disparo para avisos de bit sólo pueden crearse variables del tipo de operando "D" y del tipo de operando "Word". Para avisos de bit es posible utilizar variables de matriz (tipo de operando: "D" tipos de datos: "ARRAY [x..y] of Word").

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

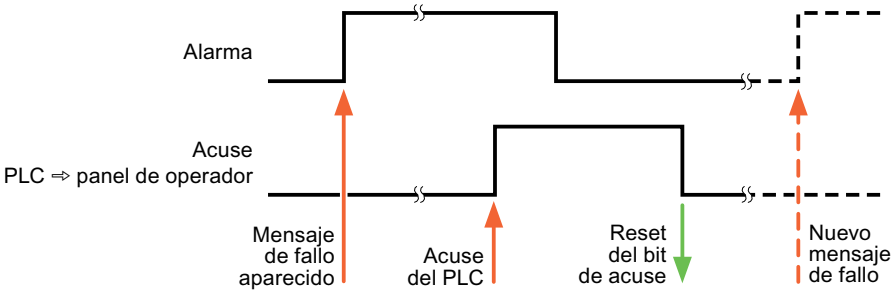
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el

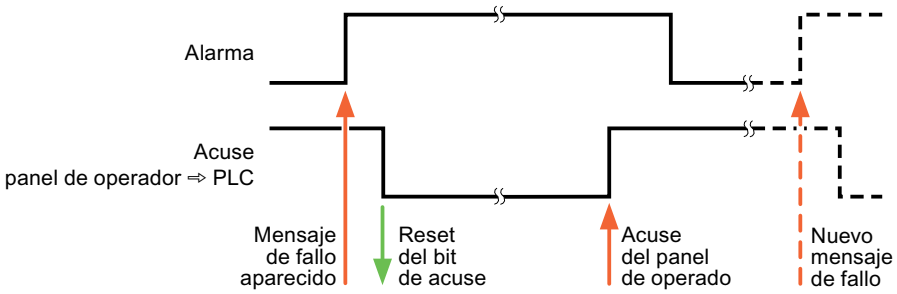
controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



Modicon Modbus

Drivers de comunicación Modicon Modbus

Introducción

Este apartado describe la comunicación entre un panel de operador y controladores que utilizan drivers de comunicación Modicon Modbus.

Se soportan los siguientes drivers de comunicación:

- Modicon Modbus TCP/IP
- Modicon Modbus RTU

Intercambio de datos

El intercambio de datos se realiza mediante variables o punteros de área.

- Variables
  El controlador y el panel de operador intercambian los datos a través de valores de proceso. Cree variables en la configuración que señalen a direcciones en el controlador. El panel de operador lee y muestra el valor de la dirección indicada. El operador también puede introducir un valor en el panel de operador. Dicho valor se escribe luego en la dirección en el controlador.
- Punteros de área
  Los punteros de área sirven para intercambiar datos especiales y se configuran únicamente si se utiliza este tipo de datos.

Modicon Modbus TCP/IP

Configurar la conexión mediante Modicon Modbus TCP/IP

Introducción

Una conexión con un controlador que tenga un driver de comunicación Modicon Modbus TCP/IP se configura en el editor "Conexiones" del panel de operador.

Las interfaces Ethernet reciben nombres distintos en función del panel de operador.

Ejemplo: la interfaz PROFINET corresponde a la interfaz Ethernet

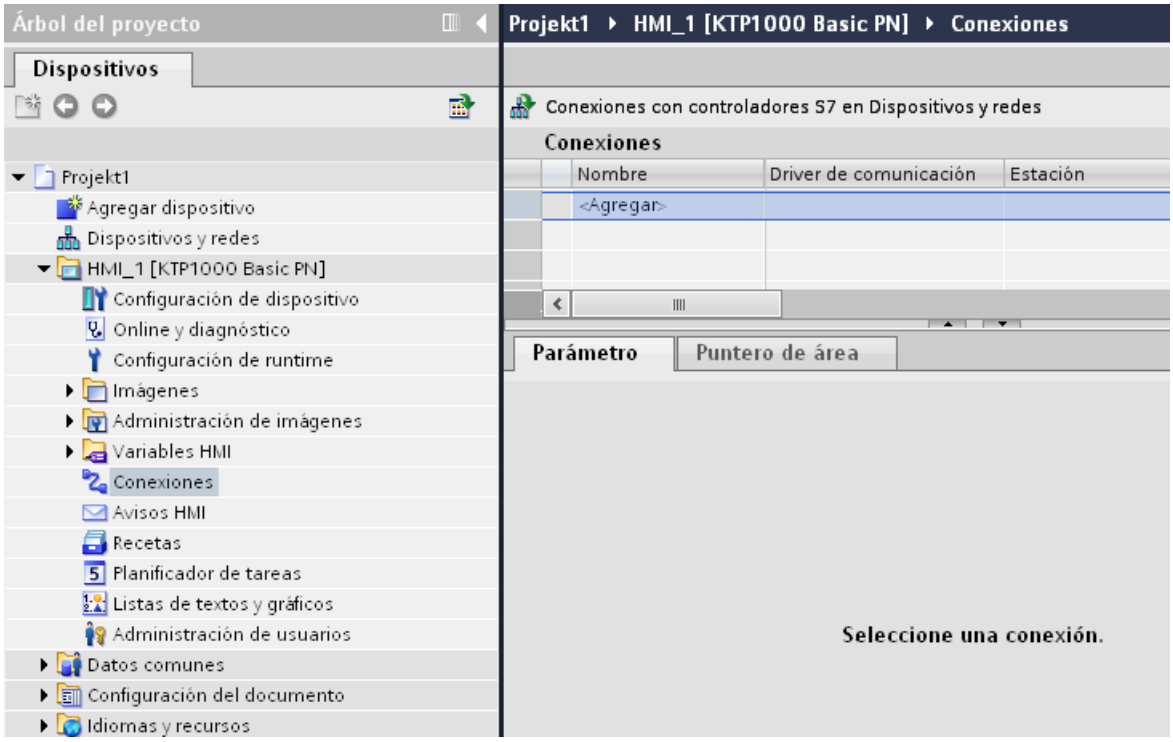
Requisitos

- Hay un proyecto abierto.
- Se ha creado un panel de operador.

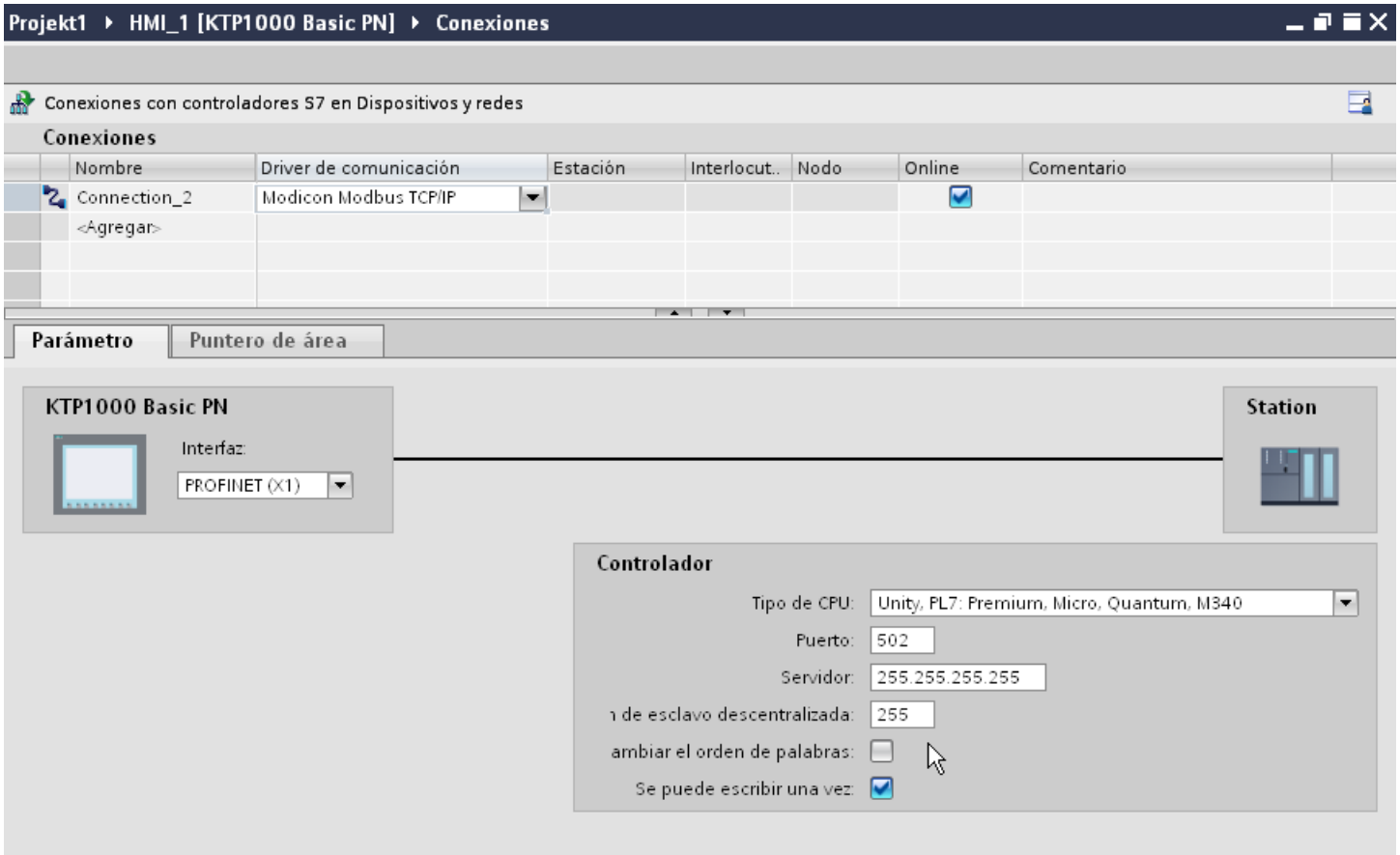
Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".

3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el controlador "Modicon Modbus TCPI" en la columna "Driver de comunicación".



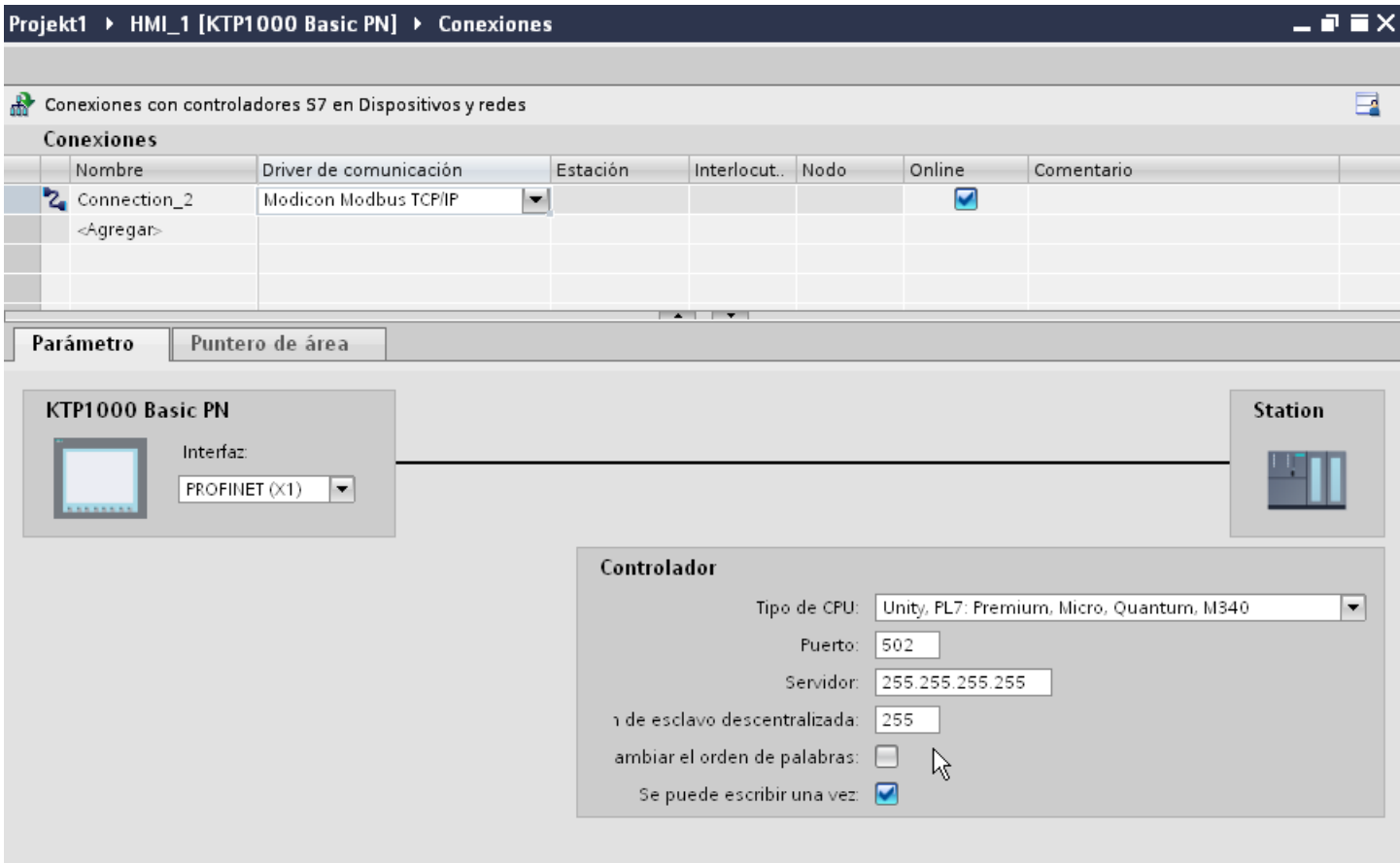
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Modicon Modbus TCP/IP)

Parámetros que deben ajustarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador" y "Controlador", según sea la interfaz utilizada.



Parámetros del panel de operador

En la ventana de inspección "Parámetros" sólo se puede seleccionar una interfaz para el panel de operador. En función del panel de operador habrá diferentes interfaces disponibles.

Si al configurar está conectado directamente con el panel de operador, es posible establecer la dirección IP del panel de operador en WinCC. Al transferir el proyecto se transfiere la dirección IP al panel de operador.

Nota

Si ya ha configurado la dirección IP en el Control Panel del panel de operador, en la siguiente carga la dirección IP se sobrescribe en el Control Panel.

Si activa "Obtener dirección IP por otra vía", en la siguiente carga la dirección IP establecida previamente se conserva en el Control Panel.

La dirección IP del panel de operador se configura del siguiente modo:

1. Haga clic en el panel de operador.
2. Abra el editor "Configuración de dispositivos".

3. Haga clic en la interfaz Ethernet.
4. Asigne la dirección IP en la ventana de inspección en:
"General > Interfaz PROFINET > Direcciones Ethernet"

Parámetros del controlador

- "Tipo de CPU"
En "Tipo de CPU" se especifica el controlador Modicon al que está conectado el panel de operador.
- "Puerto"
En "Puerto" se ajusta el puerto que se utiliza para la conexión TCP/IP. El puerto que utilizan los controladores Modicon es el 502.
- "Servidor"
En "Servidor" se ajusta la dirección IP o el nombre de host del controlador. En un Basic Panel sólo puede utilizarse la dirección IP.
- "Dirección de esclavo descentralizada"
Solo si se utiliza un bridge, en "Dirección de esclavo descentralizada" se ajusta la dirección de esclavo que tiene el controlador remoto.
Si no se utiliza ningún bridge, debe mantenerse el valor predeterminado 255 (o bien 0).
- "Change word order"
El parámetro "Change word order" sólo afecta al orden de palabras en la representación de los valores de 32 bits. El ajuste afecta a los tipos de datos Double, Double+/- y Float. No se puede modificar el orden de los bytes.
  - "Change word order" no activado
  El "Most Significant Byte" se envía en primer lugar.
  En las palabras dobles, la "Least Significant Word" se envía antes que la "Most Significant Word".
  Este ajuste se ha probado en el sistema para todos los controladores autorizados.
  - "Change word order" activado
  El "Most Significant Byte" se envía en primer lugar.
  En las palabras dobles, la "Most Significant Word" se envía antes que la "Least Significant Word".

  Nota

  Este ajuste debe utilizarse para los instrumentos de medición multifuncionales de SIEMENS SENTRON PAC3200 y PAC4200 y también puede emplearse para controladores de otros fabricantes.

- "Use single write"
Si se deselecciona esta función, para escribir en el controlador sólo se utilizarán los códigos de función 15H y 16H.
Si esta función permanece seleccionada, se utilizarán los códigos de función 05H, 06H, 15H y 16H.

Conectar el panel de operador con el controlador

Conexiones mediante Modicon Modbus TCP/IP

Conexión

El panel de operador puede conectarse al controlador Modicon Modbus utilizando los componentes siguientes:

- Una red Ethernet existente en la que también se encuentran los controladores
- Un cable Ethernet cruzado (Cross-Over) directamente a la interfaz Ethernet de la CPU o del módulo de comunicación

El acoplamiento entre el panel de operador y un controlador Modicon Modbus se limita principalmente a la conexión física del panel de operador. No se requieren bloques especiales para el acoplamiento en el controlador.

Nota

Comportamiento de timeout en TCP/IP (Ethernet)

Debido a que se utiliza el protocolo TCP/IP, las conexiones interrumpidas se detectan a lo sumo al cabo de aprox. un minuto. Si no se solicitan variables (p. ej. si la imagen actual no contiene una variable de salida), la interrupción de la conexión no se detectará con seguridad.

Configure un puntero de área "Coordinación" para cada controlador. Este ajuste garantiza que, aun en el caso descrito, la interrupción de la conexión se detecte al cabo de aprox. dos minutos.

Tipos de comunicación

Tipos de comunicación habilitados

Los siguientes tipos de comunicación han sido comprobados y activados:

- Acoplamiento punto a punto:
- Acoplamiento de varios puntos de un panel de operador (cliente Modbus TCP/IP) con un número de hasta 4 controladores con cuatro acoplamientos distintos. Es posible la mezcla de los tipos de CPU.
  Además, son posibles los siguientes acoplamientos:
  - Acoplamiento a la interfaz de Ethernet de la CPU de TSX Unity Quantum.
  - Acoplamiento a través de los módulos de comunicación para Ethernet 140 NOE 771 01 para las series TSX Quantum y TSX Unity Quantum
  - Acoplamiento a través de la interfaz Ethernet del adaptador de CPU 171 CCC 980 30 de la serie Momentum
  - Acoplamiento a la interfaz de Ethernet de la CPU de TSX Unity Premium.
  - Acoplamiento a través de los módulos de conexión de Ethernet TCP/IP TSX ETY 110 para las series TSX Premium y TSX Unity Premium
  - Acoplamiento a través de los módulos de conexión de Ethernet TCP/IP TSX ETY 410 para la serie Micro
  - Acoplamiento a través de Ethernet TCP/IP-Modbus Plus Bridge 174 CEV 200 40 a la interfaz Modbus Plus de Compact, de TSX Quantum y de TSX Unity Quantum

A través de TCP/IP-Modbus Plus Bridge 174 CEV 200 40 se puede acceder a los controladores con su dirección de esclavo remoto mediante la interfaz Ethernet de este bridge.

Nota

La integración del panel de operador en una red Modbus a través de un Bridge no es posible. El panel de operador es el maestro Modbus.

Restricciones

El acoplamiento del panel de operador a los controladores de otros fabricantes que ofrecen una interfaz Modbus TCP/IP no está probado por el sistema y por tanto no está autorizado.

No obstante, si emplea otro controlador, tenga en cuenta las siguientes indicaciones:

- Emplee los siguientes tipos de CPU, dado que estos funcionan sin offset de direccionamiento y con el cómputo de bits habitual.
  - Unity, PL7: Premium, Micro, Quantum, M340
- Los siguientes códigos de función se utilizan para las respectivas áreas de datos:

Códigos de función de lectura		Área de direcciones	
01	ReadCoilStatus	0x / %M	DIGITAL_OUT
02	ReadInputStatus	1x / %I	DIGITAL_IN
03	ReadHoldingRegisters	4x / %MW	USERDATA
04	ReadInputRegisters	3x / %IW	ANALOG_IN
20 (14Hex)	ReadGeneralReference	6x / –	EXTENDEDMEMORY (No en todas las CPU)

Códigos de función de escritura		Área de direcciones	
06 [1]	PresetSingleRegister	4x / %MW	USERDATA Single
16 (10Hex)	PresetMultipleRegisters	4x / %MW	USERDATA Multiple
05 [1]	ForceSingelCoil	0x / %M	DIGITAL_OUT con BIT
15 (0FHex)	ForceMultipleCoils	0x / %M	DIGITAL_OUT con 16 BIT GROUP
21 (15Hex)	WriteGeneralReference	6x / –	EXTENDEDMEMORY (No en todas las CPU)

[1] Se puede seleccionar su uso a través de "Use single write".

Controladores acoplables

Es posible realizar acoplamientos para los siguientes controladores Modicon Modbus:

Controlador Modicon Modbus	Protocolo soportado	
Modicon Modbus RTU [2]	Modicon Modbus TCP/IP	
TSX Compact	x	x[1]
TSX Quantum	x	x
Momentum	x	x
Premium	-	x
Micro	-	x
M340 20x0 (sin 2010)	-	x

[1] Sólo a través de Ethernet TCP/IP-Modbus Plus Bridge

[2] El controlador ha probado y autorizado la comunicación vía RS 232. En los HMI que sólo poseen una interfaz RS 422/485, se ha probado y autorizado el convertidor RS 422/232 con el número de pedido 6AV6 671-8XE00-0AX0.

Características de rendimiento de la comunicación

Tipos de datos admisibles para Modicon Modbus TCP

Tipos de datos permitidos

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Nota

Si cambia el driver de comunicación de Modicon Modbus RTU a Modicon Modbus TCP/IP, entonces la secuencia de caracteres del tipo de datos "String" puede ser diferente.

Tipos de datos admisibles para el tipo de CPU "Unity, PLC: Premium, Micro, Quantum M340"

Tipo de datos	Tipo de operando	Longitud
+/- Double	%MW	4 bytes
+/- Int	%MW, %IW	2 bytes
16 bit group	%MW, %I	2 bytes
ASCII	%MW	de 0 a 80 caracteres
Bit	%MW, %IW, %M, %I	1 bit
Double	%MW	4 bytes
Float	%MW	4 bytes
Int	%MW, %IW	2 bytes

Nota

En los siguientes tipos de CPU no se soportan las áreas "%I" y "%IW":

- Premium
- Micro
- M340

Tipos de datos admisibles para el tipo de CPU "Concept, ProWORX: Compact, Quantum, Momentum"

Tipo de datos	Tipo de operando	Longitud
+/- Double	4x, 6x	4 bytes
+/- Int	3x, 4x, 6x	2 bytes
16 bit group	0x, 1x	2 bytes

Tipo de datos	Tipo de operando	Longitud
ASCII	4x, 6x	de 0 a 80 caracteres
Bit	0x, 1x, 3x, 4x, 6x	1 bit
Double	4x, 6x	4 bytes
Float	4x, 6x	4 bytes
Int	3x, 4x, 6x	2 bytes

Modo de contaje de bits

El modo habitual de contar bits "16 LSB - 1 MSB" de los siguientes tipos de CPU sólo se utiliza en el editor "Variables HMI" al estar seleccionado el tipo de datos "Bit":

- Concept, ProWORX: Compact, Quantum, Momentum

Asignación de las posiciones de los bits:

	Byte izquierdo								Byte derecho							
Modo de contaje en variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Representación de "Signed"

El comodín "+/-" representa los tipos de datos "Signed Int" y "Signed Double".

Tipos de CPU soportados para Modicon Modbus TCP/IP

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Modicon Modbus TCP/IP.

- Compact
- Momentum
- Quantum
  - Concept Quantum
  - Unity Quantum
- Micro
- Premium
- Modicon M340
  - 20x0 (excepto 2010)

Áreas de direcciones para Modicon Modbus TCP/IP

UnityPl7

Áreas de direcciones	Tipos de datos							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
%1	%I0 - %I65535	%I65535 %I0 - %I65520	--	--	--	--	--	--
%M	%M0 - %M65535	%M65535 %M0 - %M65520	--	--	--	--	--	--
%IW	%IW0.0 - %IW65535.15	--	%IW0 - %IW65535	%IW0 - %IW65535	--	--	--	--
%MW	%MW0.0 - %MW65535.15	--	%MW0 - %MW65535	%MW0 - %MW65535	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65535

ConceptProWORX

Áreas de direcciones	Tipos de datos							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
0x	0x1 - 0x65535	0x1 - 0x65520	--	--	--	--	--	--
1x	1x100001 - 1x165535	1x100001 - 1x165520	--	--	--	--	--	--
3x	3x300001.1 - 3x365535.16	--	3x300001 - 3x365535	3x300001 - 3x365535	--	--	--	--
4x	4x400001.1 - 4x465535.16	--	4x400001 - 4x465535	4x400001 - 4x465535	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465535
6x	6x60000.1:1 - 6x69999.16:10	--	6x60000:1 - 6x69999:10	6x60000:1 - 6x69999:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69999:10

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   - Interfaz
   - Parámetros de transferencia
   - Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Modicon Modbus RTU

Configurar la conexión mediante Modicon Modbus RTU

Introducción

Una conexión con un controlador que tenga un driver de comunicación Modicon Modbus RTU se configura en el editor "Conexiones" del panel de operador.

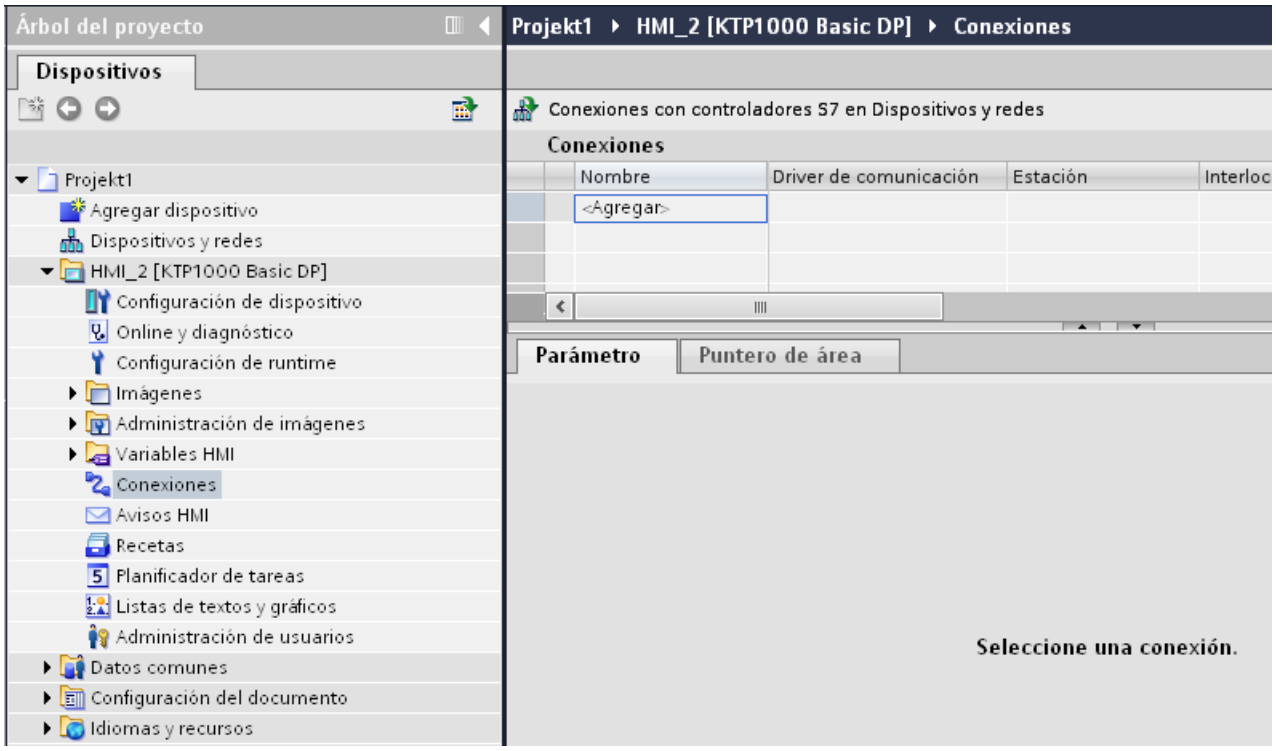
Las interfaces reciben nombres distintos en función del panel de operador.

Requisitos

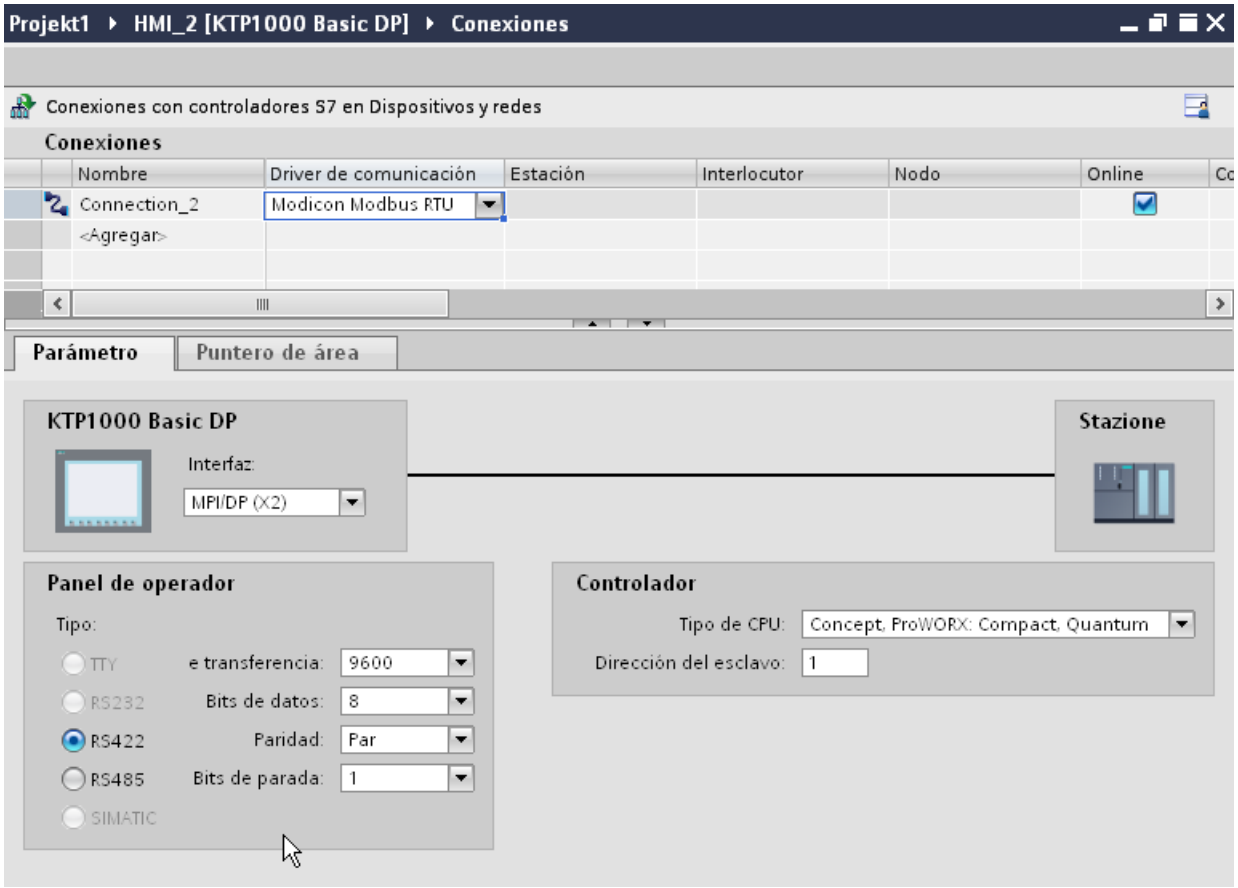
- Hay un proyecto abierto.
- Se ha creado un panel de operador.

Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".
3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el driver "Modicon Modbus RTU" en la columna "Driver de comunicación".



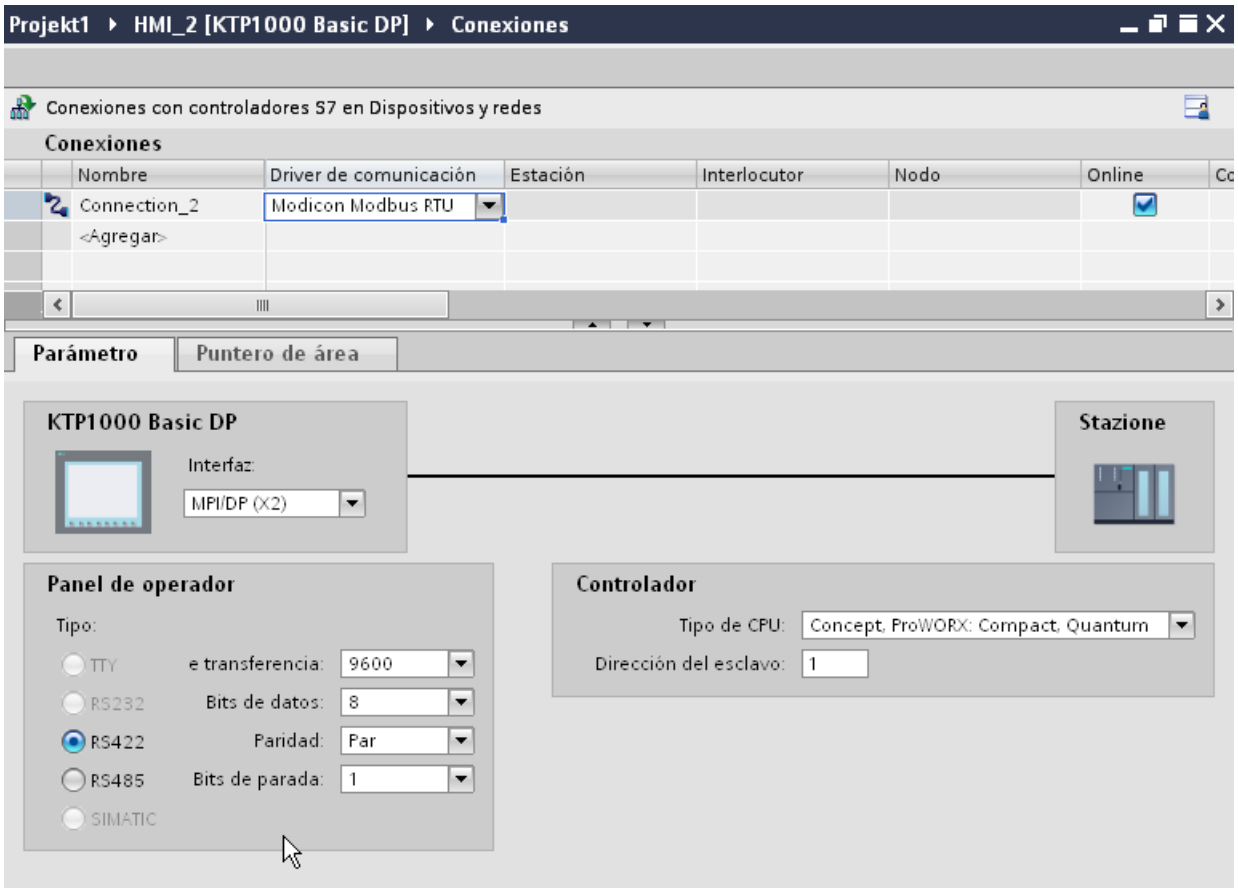
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Modicon Modbus RTU)

Parámetros que deben configurarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador" y "Controlador", según sea la interfaz utilizada.


Projekt1 ▸ HMI_2 [KTP1000 Basic DP] ▸ Conexiones
Conexiones con controladores S7 en Dispositivos y redes
Conexiones
Nombre
Driver de comunicación
Estación
Interlocutor
Nodo
Online
Connection_2
Modicon Modbus RTU
<Agregar>
Parámetro
Puntero de área
KTP1000 Basic DP
Interfaz:
MPI/DP (X2)
Stazione
Panel de operador
Tipo:
TTY
RS232
RS422
RS485
SIMATIC
e transferencia:
9600
Bits de datos:
8
Paridad:
Par
Bits de parada:
1
Controlador
Tipo de CPU:
Concept, ProWORX: Compact, Quantum
Dirección del esclavo:
1

Parámetros del panel de operador

En la ventana de inspección "Parámetros" se puede seleccionar una interfaz para el panel de operador. En función del panel de operador habrá diferentes interfaces disponibles.

- Tipo
  Sólo RS 232 ha sido probado en el sistema.
  No se proporciona ninguna garantía para RS 485.

  Nota

  RS 422 sólo está autorizado en combinación con el convertidor RS 422-RS 232.

  Referencia: 6AV6 671-8XE00-0AX0

  Nota

  Cuando se deba utilizar la interfaz IF1B, hay que conmutar además los datos de recepción RS422 mediante el interruptor 4 DIL de la parte posterior del panel de operador.

- Velocidad de transferencia
  En "Velocidad de transferencia" se ajusta la velocidad de transferencia entre el panel de operador y el controlador Modicon. La comunicación es posible con una velocidad de transferencia de 19200 y 9600 baudios.
  Para determinados paneles de operador se puede seleccionar la velocidad de transferencia de 4.800 baudios.
- Bits de datos
  En "Bits de datos" sólo se puede elegir "8".
- Paridad
  En "Paridad" hay que elegir entre "Ninguna", "Par" e "Impar".
- Bits de parada
  En "Bits de parada" hay que elegir entre 1 y 2.

Parámetros del controlador

- Tipo de CPU
  En "Tipo de CPU" se especifica el controlador Modicon al que está conectado el panel de operador.
  Se puede elegir entre las siguientes CPU:
  – Concept, ProWORX: Compact, Quantum
- Dirección del esclavo
  En "Dirección del esclavo" se especifica la dirección del esclavo que tiene la CPU.

Conectar el panel de operador con el controlador

Conexiones mediante Modicon Modbus RTU

Conexión

Conecte el panel de operador a la interfaz Modicon Modbus RTU del esclavo Modicon Modbus RTU.

El acoplamiento entre el panel de operador y Modicon se limita principalmente a la conexión física del panel de operador. No se requieren bloques especiales para el acoplamiento en el controlador.

Cable de conexión

Para conectar el panel de operador a Modicon Modbus están disponibles los siguientes cables de conexión:

Interfaz en el panel de operador	Controlador Modicon		
	directamente vía interfaz Modbus (RS232) con conector Sub D macho de 9 polos	vía MB Bridge (RS 232)	directamente vía interfaz Modbus (RS232) con conector RJ45 de 8 pines
RS 232, 9 polos	PP1	PP1	PP2

Encontrará la asignación de pines de los cables en el apartado "Cables de conexión para Modicon Modbus RTU".

Tipos de comunicación

Tipos de comunicación habilitados

Los siguientes tipos de comunicación han sido comprobados y activados:

- Acoplamiento punto a punto sólo vía interfaz RS-232.
- Acoplamiento a varios puntos de un panel de operador (Modbus maestro) con hasta 4 controles: el panel de operador debe estar conectado con un Modbus Plus Bridge o una CPU Compact, Momentum o una CPU TSX Quantum que esté configurada como Modbus Plus Bridge.

- Los siguientes controladores se conectan mediante la conexión Modbus Plus al primer controlador. Se puede acceder a los controladores bajo su dirección a través de la funcionalidad de Bridge del primer controlador.

Nota

No se puede integrar el panel de operador en una red Modbus, ya que el panel de operador es el modbus-maestro.

- La integración del panel de operador en una red Modbus Plus a través del "Modo Bridge" de una Compact, Momentum o Quantum (comunicación lógica punto a punto del panel de operador con una Compact, Momentum o Quantum).

Restricciones

El acoplamiento del panel de operador a controladores de otros fabricantes que ofrecen una interfaz Modicon Modbus no se ha probado en el sistema y por ello no está autorizado.

Si de todas maneras quiere usar otro controlador, proceda como sigue:

- este controlador funciona sólo con variables con el modo de contaje habitual para los controladores Modicon desde la izquierda (Bit1 = bit de mayor valor) hacia la derecha (Bit16 = bit de menor valor del tipo de datos INT).
- el offset de dirección mostrado durante la configuración se sustrae a nivel de protocolo en el telegrama. P. ej., está en el Holding Register 4x del offset "40001". Así se pasa de la dirección configurada "40006" en el telegrama a la dirección "5". La conversión de la dirección contenida en el telegrama (p. ej. "5") al área de direcciones específica de los controladores se realiza de forma diferente en los diferentes controladores no Modicon.
- se espera un telegrama de respuesta sin "ExceptionCode" en un plazo de 500 ms.
- los siguientes códigos de función se utilizan para las respectivas áreas de datos:

Códigos de función de lectura		Área de direcciones	
01	ReadCoilStatus	0x	DIGITAL_OUT
02	ReadInputStatus	1x	DIGITAL_IN
03	ReadHoldingRegisters	4x	USERDATA
04	ReadInputRegisters	3x	ANALOG_IN
20 (14Hex)	ReadGeneralReference	6x	EXTENDEDMEMORY (No en todas las CPU)

Códigos de función de escritura		Área de direcciones	
06	PresetSingleRegister	4x	USERDATA Single
16 (10Hex)	PresetMultipleRegisters	4x	USERDATA Multiple
05	ForceSingelCoil	0x	DIGITAL_OUT con tipo de datos Bit
15 (0FHex)	ForceMultipleCoils	0x	DIGITAL_OUT con tipo de datos 16 bit group
21 (15Hex)	WriteGeneralReference	6x	EXTENDEDMEMORY (No en todas las CPU)

Controladores acoplables

Es posible realizar acoplamientos para los siguientes controladores Modicon Modbus:

Controlador Modicon Modbus	Protocolo soportado	
Modicon Modbus RTU [2]	Modicon Modbus TCP/IP	
TSX Compact	x	x [1]
TSX Quantum	x	x
Momentum	x	x
Premium	-	x
Micro	-	x
M340 20x0 (sin 2010)	-	x

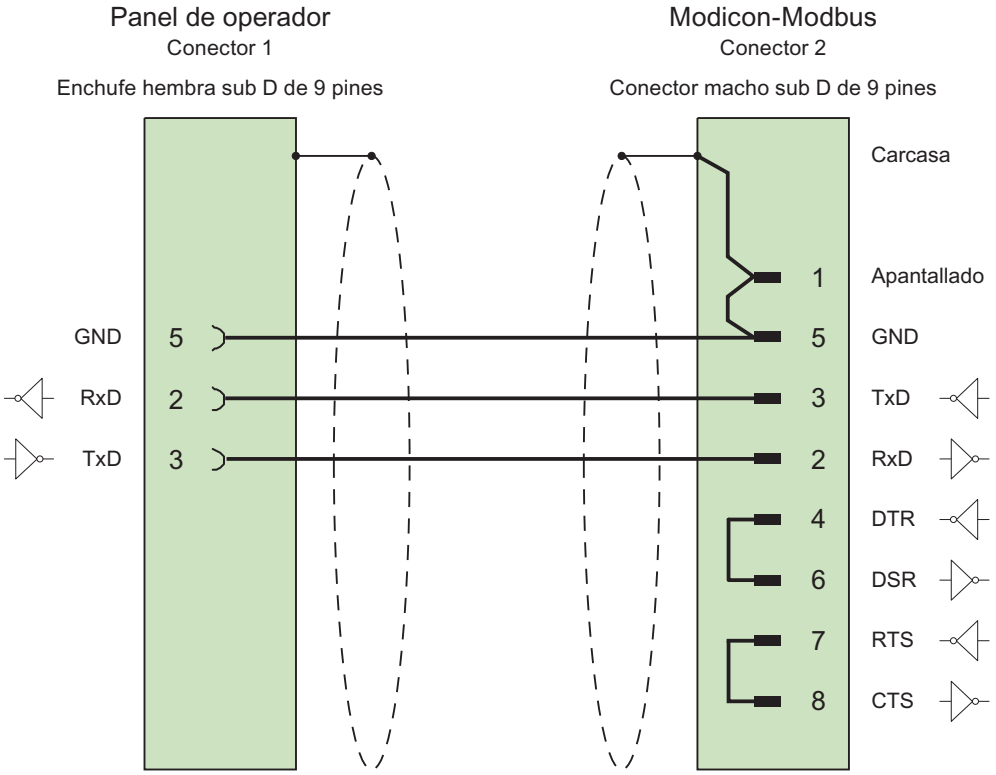
[1] Sólo a través de Ethernet TCP/IP-Modbus Plus Bridge

[2] El controlador ha probado y autorizado la comunicación vía RS 232. Para los paneles de operador que sólo tienen una interfaz RS 422/485 se ha probado y autorizado el convertidor RS 422/232 con la referencia 6AV6 671-8XE00-0AX0.

Cables de conexión para Modicon Modbus RTU

Cable de conexión PP1, RS 232, para Modicon

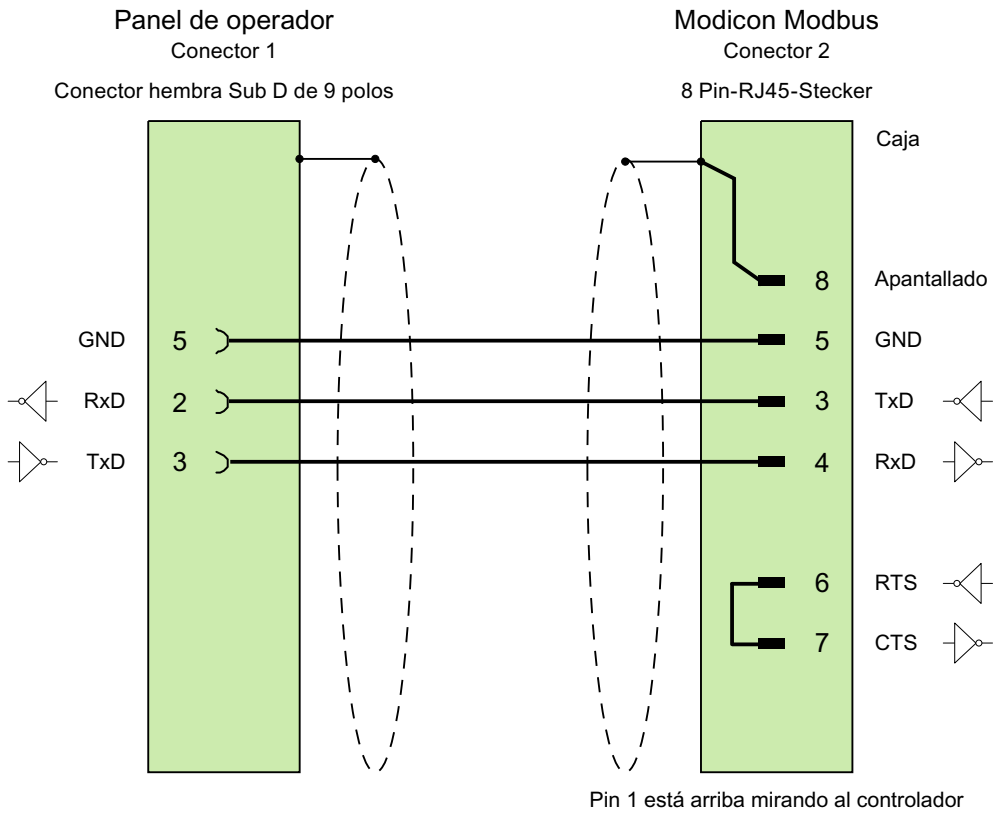
Cable punto a punto 1: PLC > PC ...



Cable: 3 x 0,14 mm^2, blindado,
longitud máx. 15 m

Cable de conexión PP2, RS 232, para Modicon

Cable punto a punto 2: PLC (TSX Compact) > PC...



Cable: 3 x 0,14 mm², blindado,
longitud máx. 15 m

Características de rendimiento de la comunicación

Tipos de datos admisibles para Modicon Modbus RTU

Tipos de datos admisibles

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Tipo de operando	Longitud
+/- Double	4x, 6x	4 bytes
+/- Int	3x, 4x, 6x	2 bytes
16 bit group	0x, 1x	2 bytes
ASCII	4x, 6x	de 0 a 80 caracteres
Bit [1)]	0x, 1x, 3x, 4x, 6x	1 bit
Double	4x, 6x	4 bytes
Float	4x, 6x	4 bytes
Int	3x, 4x, 6x	2 bytes

[1)] Nota relativa a los accesos de escritura:

En el tipo de datos "Bit" con los tipos de operando "4x" y "6x", se vuelve a escribir la palabra completa en el controlador tras modificar el bit indicado. No se comprueba si entretanto se han modificado otros bits en la palabra. Por este motivo, el controlador sólo puede leer la palabra indicada.

El modo de contaje habitual (16 LSB - 1 MSB) de los siguientes tipos de CPU sólo se utiliza en el editor "Variables HMI" al estar seleccionado el tipo de datos "Bit":

- Concept ProWORX: Compact, Quantum

Asignación de las posiciones de los bits:

	Byte izquierdo								Byte derecho							
Modo de contaje en variables	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Representación de "Signed"

El comodín "+/-" representa los tipos de datos "Signed Int" y "Signed Double".

Tipos de CPU soportados para Modicon Modbus RTU

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Modicon Modbus TCP/IP.

- Compact
- Momentum
- Quantum

Áreas de direcciones para Modicon Modbus RTU

UnityPl7

Áreas de direcciones	Tipos de datos							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
%1	%I0 - %I65535	%I65535 %I0 - %I65520	--	--	--	--	--	--
%M	%M0 - %M65535	%M65535 %M0 - %M65520	--	--	--	--	--	--
%IW	%IW0.0 - %IW65535.15	--	%IW0 - %IW65535	%IW0 - %IW65535	--	--	--	--
%MW	%MW0.0 - %MW65535.15	--	%MW0 - %MW65535	%MW0 - %MW65535	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65534	%MW0 - %MW65535

ConceptProWORX

Áreas de direcciones	Tipos de datos							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
0x	0x1 - 0x65535	0x1 - 0x65520	--	--	--	--	--	--
1x	1x100001 - 1x165535	1x100001 - 1x165520	--	--	--	--	--	--
3x	3x300001.1 - 3x365535.16	--	3x300001 - 3x365535	3x300001 - 3x365535	--	--	--	--

Áreas de direcciones	Tipos de datos							
	Bool	16 Bit Group	Int	+/- Int	DInt	+/- DInt	Float	ASCII
4x	4x400001.1 - 4x465535.16	--	4x400001 - 4x465535	4x400001 - 4x465535	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465534	4x400001 - 4x465535
6x	6x60000.1:1 - 6x69999.16:10	--	6x60000:1 - 6x69999:10	6x60000:1 - 6x69999:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69998:10	6x60000:1 - 6x69999:10

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   - Interfaz
   - Parámetros de transferencia
   - Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Intercambio de datos

Punteros de área con Modicon Modbus

Punteros de área en conexiones vía driver de comunicación Modicon Modbus

A través de un puntero de área se accede a un rango de datos del controlador.

Encontrará más información sobre los punteros de área y su configuración en el capítulo "Intercambio de datos mediante punteros de área (Página 9118)".

Particularidades de las conexiones mediante el driver de comunicación Modicon Modbus

Puede configurar los siguientes punteros de área

Puntero de área	Modicon Modbus TCP/IP	Modicon Modbus RTU
Número de imagen	sí	sí
Fecha/hora	sí	sí
Fecha/hora PLC	sí	sí
Coordinación	sí	sí
Identificación del proyecto	sí	sí
Petición de control	sí	sí
Registro	sí	sí

Limitaciones Modicon Modbus TCP/IP

La configuración de punteros de área tiene las siguientes restricciones.

Tipo de CPU	Tipos de datos	Tipos de archivo
Concept, ProWORX: Compact, Quantum, Momentum	+/- Int, Int	4x, 6x
Unity, PL7: Premium, Micro, Quantum, M340	+/- Int, Int	%MW

Restricciones de Modicon Modbus RTU

La configuración de punteros de área tiene las siguientes limitaciones.

Tipo de CPU	Tipos de datos	Tipos de archivo
Concept, ProWORX: Compact, Quantum, Momentum	+/- Int, Int	4x, 6x

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos con conexiones no integradas

Para configurar avisos tales como advertencias, mensajes de error y acuses se requieren varios pasos.

- Paso 1: Crear variables
- Paso 2: Configurar avisos
- Paso 3: Configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- Tipos de datos de la variable
- Direccionamiento de la variable
- Modo de contaje de las posiciones de bit

Tipos de datos

Para las conexiones con drivers de comunicación Modicon Modbus se soportan los tipos de datos siguientes:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
Todas las series Modicon	Int, +/-Int	16 Bit Group, Int, +/-Int, Double, +/-Double, Float

Las matrices y variables de matriz pueden utilizarse para los avisos de bit.

Modo de contaje de las posiciones de bit

Para las conexiones con drivers de comunicación Modicon Modbus rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte izquierdo								Byte derecho							
Configure en WinCC:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

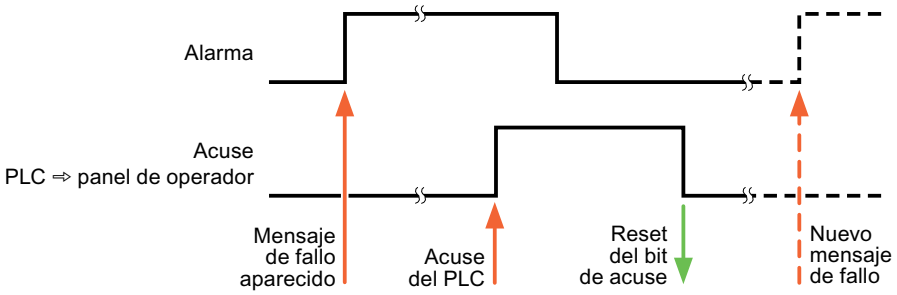
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

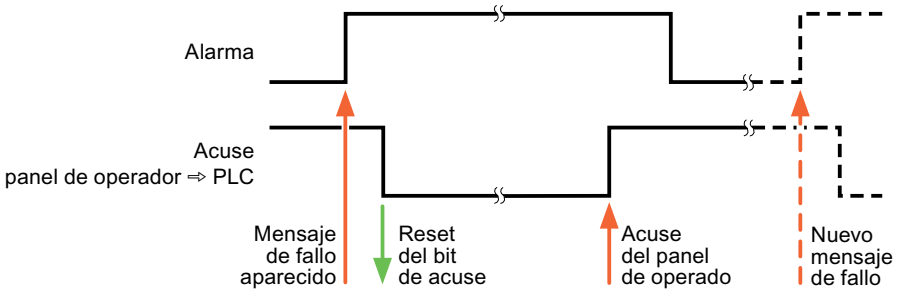
Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



Omron

Drivers de comunicación Omron

Introducción

Este apartado describe la comunicación entre un panel de operador y un controlador que utilizan drivers de comunicación Omron.

Se soportan los siguientes drivers de comunicación:

- Omron Host Link

Intercambio de datos

El intercambio de datos se realiza mediante variables o punteros de área.

- Variables
  El controlador y el panel de operador intercambian los datos a través de valores de proceso. Cree variables en la configuración que señalen a direcciones en el controlador. El panel de operador lee y muestra el valor de la dirección indicada. El operador también puede introducir un valor en el panel de operador. Dicho valor se escribe luego en la dirección en el controlador.
- Punteros de área
  Los punteros de área sirven para intercambiar datos especiales y se configuran únicamente si se utiliza este tipo de datos.

Omron Hostlink

Configurar una conexión mediante Omron Host Link

Introducción

Una conexión con un controlador que tenga un driver de comunicación Omron Host Link se configura en el editor "Conexiones" del panel de operador.

Nota

Establecimiento de conexión con Omron Host Link

Si ha configurado una conexión a través de Omron, al iniciar runtime no se establece automáticamente una conexión.

En la imagen inicial de runtime debe estar configurada una variable que se encuentre en el área de memoria válida del cotrolador.

De lo contrario, la conexión sólo se establece cuando se selecciona una imagen correspondiente.

Al iniciar runtime se llama dicha variable y con ello se establece una conexión.

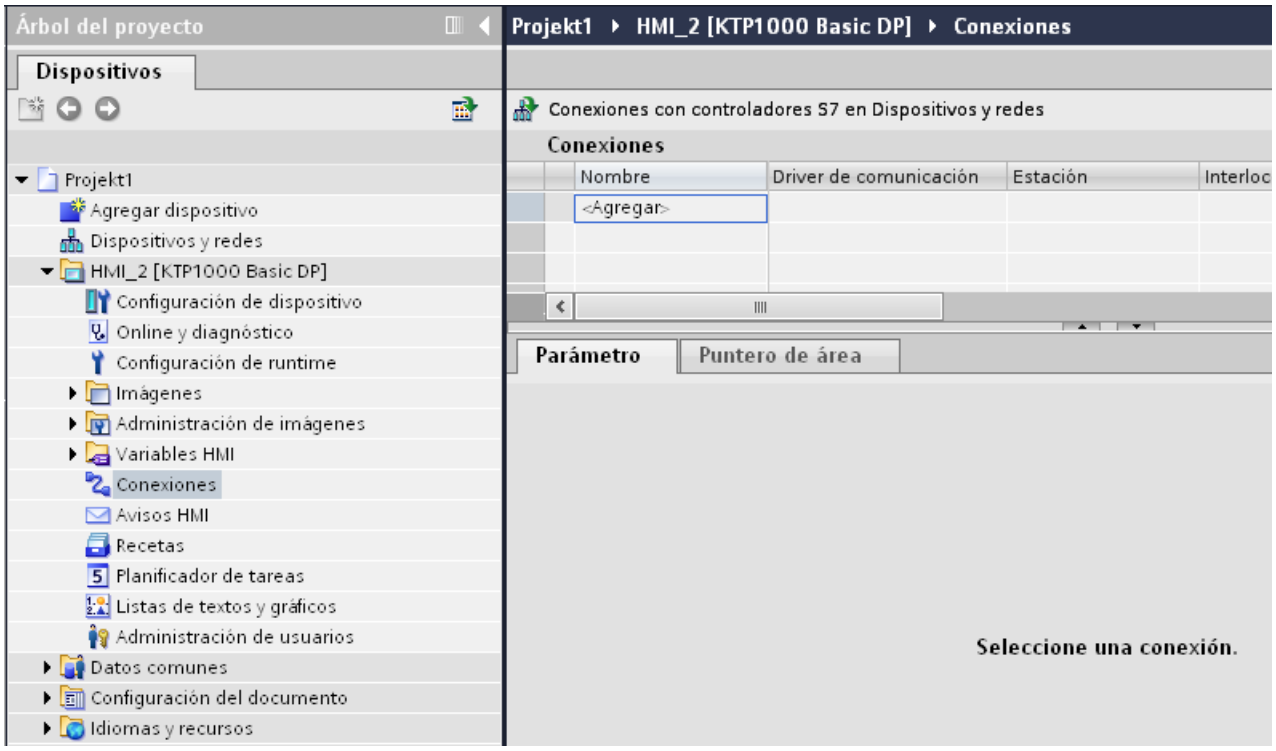
Las interfaces reciben nombres distintos en función del panel de operador.

Requisitos

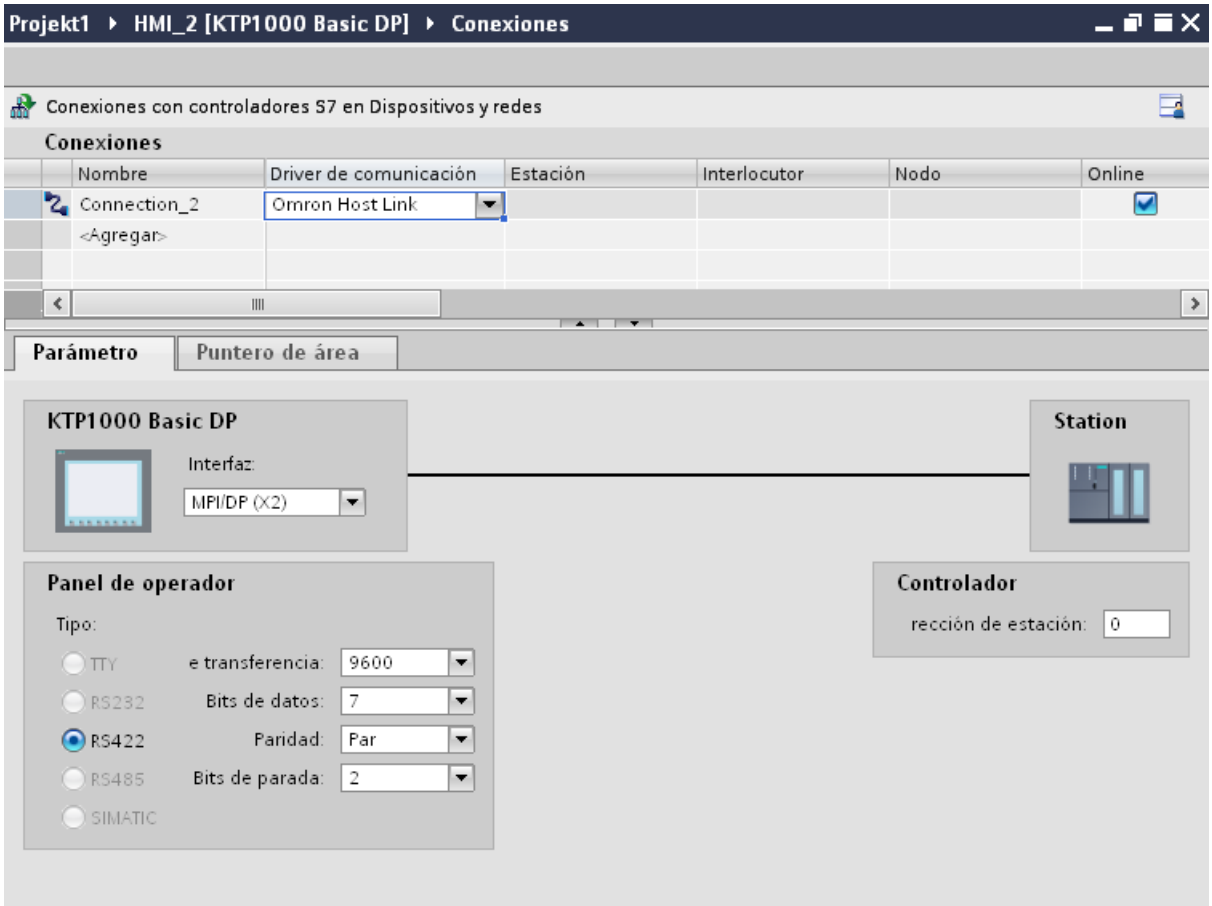
- Hay un proyecto abierto.
- Se ha creado un panel de operador.

Procedimiento

1. Haga doble clic en el panel de operador que aparece en "Dispositivos" del árbol del proyecto.
2. Haga doble clic en la entrada "Conexiones".
3. Haga doble clic en "<Agregar>" en el editor "Conexiones".



4. Seleccione el controlador "Omron Host Link" en la columna "Driver de comunicación".



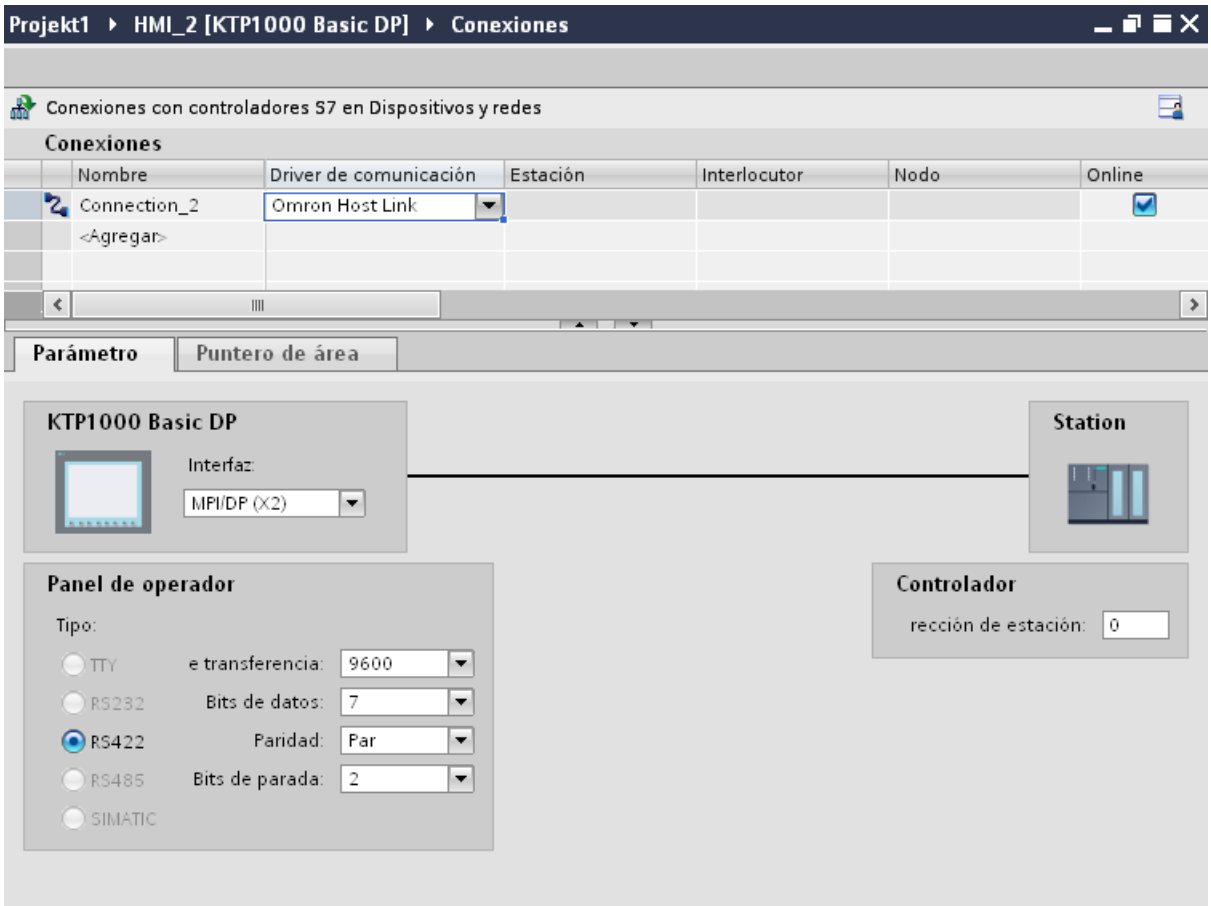
5. Seleccione todos los parámetros de conexión necesarios para la interfaz en "Parámetros" de la ventana de inspección.

Parámetros de la conexión (Omron Hostlink)

Parámetros que deben configurarse

Para ajustar los parámetros de la conexión, como direcciones y perfiles, haga clic en la conexión creada en el editor "Conexiones".

En "Parámetros" de la ventana de inspección se representan esquemáticamente los interlocutores. Para ajustar los parámetros existen las áreas "Panel de operador" y "Controlador", según sea la interfaz utilizada.



Parámetros del panel de operador

En la ventana de inspección "Parámetros" se puede seleccionar una interfaz para el panel de operador. En función del panel de operador habrá diferentes interfaces disponibles.

- Tipo
  Define la conexión física utilizada.
- Velocidad de transferencia
  En "Velocidad de transferencia" se ajusta la velocidad para transferir del panel de operador a OMRON. La comunicación es posible con una velocidad de transferencia de 19200 y 9600 baudios.
- Bits de datos
  En "Bits de datos" hay que elegir entre "7 bits" u "8 bits".
- Paridad
  En "Paridad" hay que elegir entre "Ninguna", "Par" e "Impar".
- Bits de parada
  En "Bits de parada" hay que elegir entre 1 y 2 bits.

Parámetros del controlador

- Dirección de estación
  En "Dirección de estación" se introduce el número de estación del controlador conectado.

Conectar el panel de operador con el controlador

Conexiones mediante Omron Host Link

Conexión

El acoplamiento entre el panel de operador y un controlador OMRON se limita principalmente a la conexión física del panel de operador. No se requieren bloques especiales para el acoplamiento en el controlador.

Cable de conexión

Para conectar el panel de operador a un controlador Omron están disponibles los siguientes cables de conexión:

Interfaz en el panel de operador	Controlador Omron			
	RS232, 9 polos	RS232 puerto periférico	RS422, 9 polos	RS422, bornes/ conectores
RS232, 9 polos	PP1	Cable de programación (cable estándar de la empresa Omron)	—	—
RS232 vía convertidor	—	—	—	Cable multipunto 1
RS422, 9 polos	—	—	PP2	Cable multipunto 2

En el manual de producto correspondiente encontrará información sobre la interfaz que debe utilizarse en el panel de operador.

Tipos de comunicación

Tipos de comunicación habilitados

El acoplamiento de un panel de operador a una CPU OMRON mediante el protocolo Omron Host Link a través de RS232 y RS 422 ha sido probado en el sistema y autorizado por Siemens AG.

Esto afecta a los siguientes tipos de CPU:

- CP1x (CP1L, CP1H, CP1E)
- CJ1x(CJ1M, CJ1H, CJ1G)
- CJ2H

- CS1x(CS1G, CS1H, CS1D)
- CPM2C

Nota

Para los Basic Panels, TP 177A y OP 77A sólo los siguientes tipos de CPU han sido probados y autorizados:

- CP1x (CP1L, CP1H, CP1E)
- CJ1x(CJ1M, CJ1H, CJ1G)

Acoplamiento multipunto

Un acoplamiento multipunto a hasta 4 controladores OMRON habilitados en un conjunto multidrop a cuatro hilos RS422 puede realizarse con módulos de comunicación en los controladores y ha sido probado en el sistema y autorizado por Siemens AG.

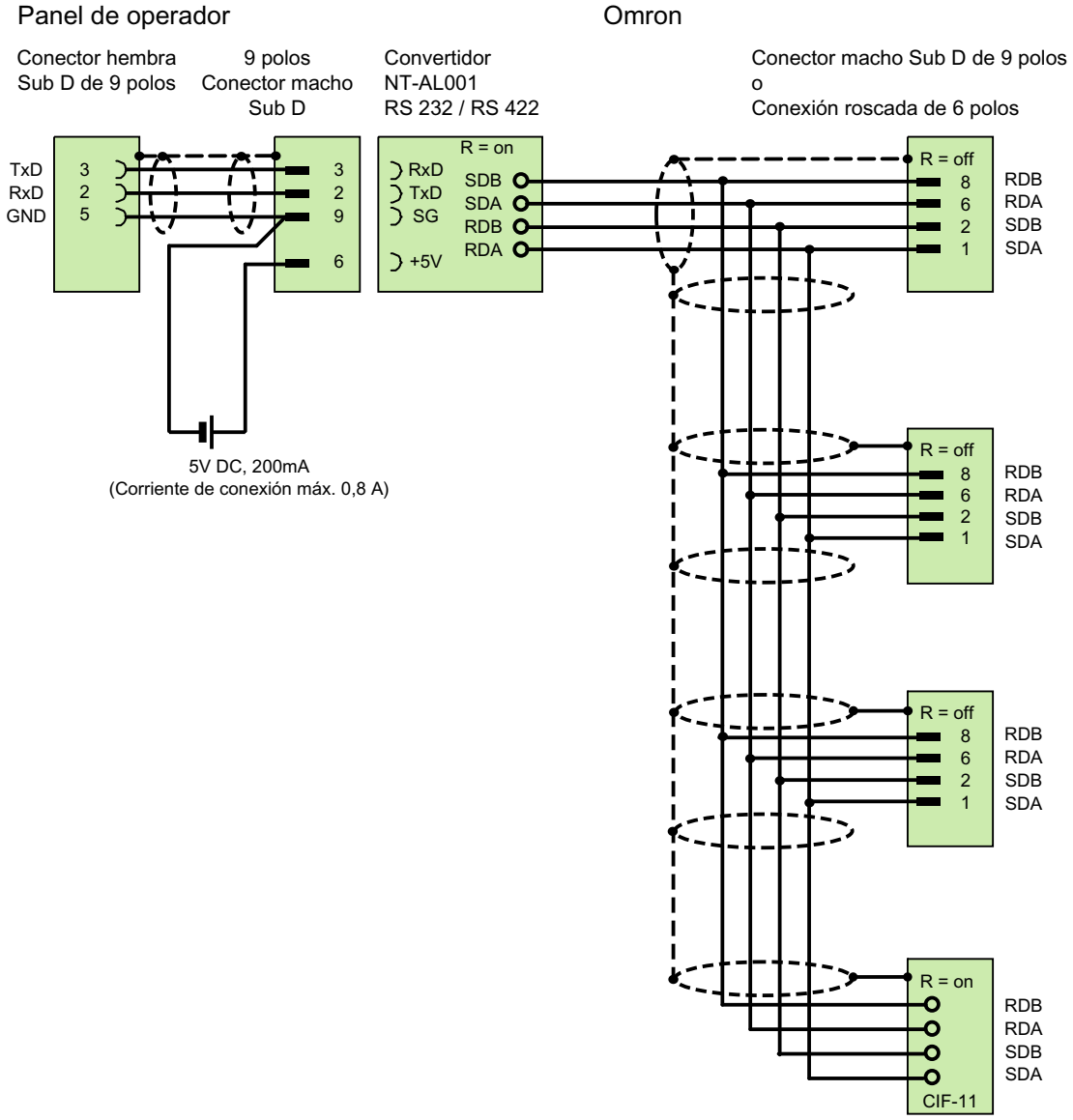
Nota

El panel de operador solo puede utilizarse como maestro. En el conjunto multidrop a cuatro hilos RS422 solo puede existir un maestro.

Cables de conexión

Cable de conexión MP1, RS 232, vía convertidor, para Omron

Cable multipunto 1: MP/TP/PC > PLC

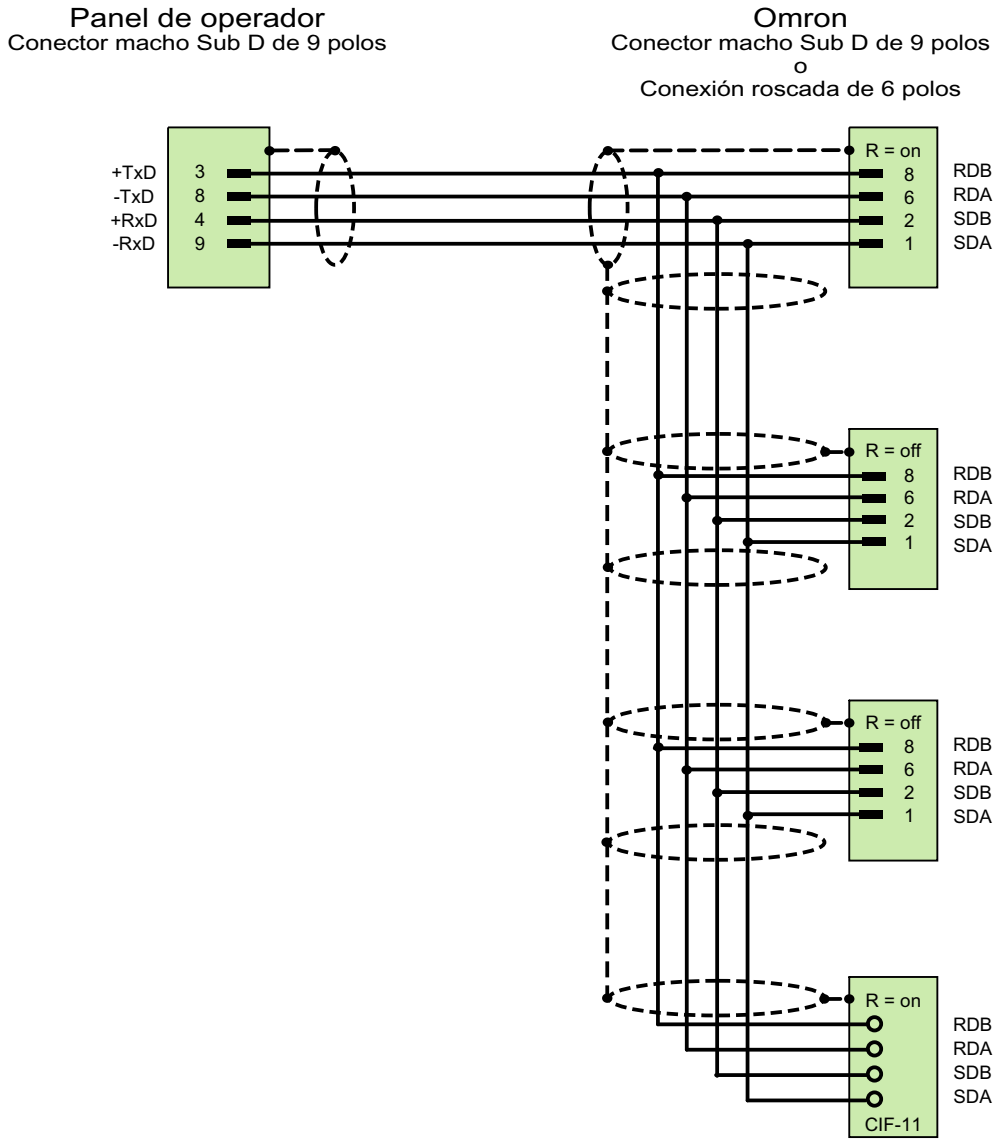


[1] Corriente al conectar: máx. 0,8 A

protegido, longitud máx. 500 m

Cable de conexión MP2, RS 422, para Omron

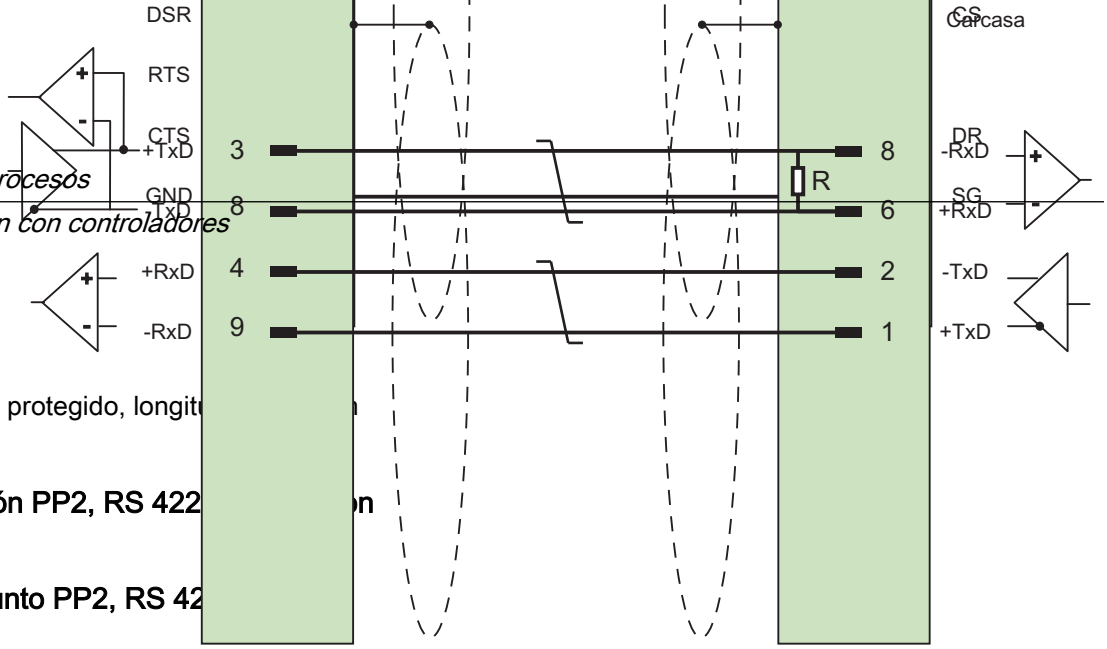
Cable multipunto 2: RS422, MP/TP/PC > PLC_



protegido, longitud máx. 500 m

Cable de conexión PP1, RS 232, para Omron

Cable punto a punto PP1, PC/TP/OP - PLC



Resistencia 220 ohmios / > 150 mW,
p. ej. modelo 0207, insertar

protegido, longitu

Cable de conexión PP2, RS 422 n

Cable punto a punto PP2, RS 42

Protegido, longitud máx. 500 m

Características de rendimiento de la comunicación

Tipos de datos admisibles para Omron Host Link

Tipos de datos admisibles

En la tabla se indican los tipos de datos que pueden utilizarse para la configuración de variables y punteros de área.

Tipo de datos	Tipo de operando	Longitud
Bool	I/O, HR, AR, LR, DM, T/C Bit, CPU Status	1 bit
Byte	Tipo de CPU	1 byte
DInt	HR, AR, LR, DM	4 bytes
Int	I/O, HR, AR, LR, DM, T/C Val	2 bytes
Real	HR, DM	4 bytes
String	HR, AR, LR, DM	de 0 a 80 caracteres
UDInt	HR, AR, LR, DM	4 bytes
UInt	I/O, HR, AR, LR, DM, T/C Val	2 bytes

Nota

Sólo es posible leer o escribir de forma fiable en todos los rangos de datos del controlador OMRON con el modo de operación "STOP" o "MONITOR".

Con "I/O" se denomina el área IR/IS o el área CIO según la serie de controlador. Los tipos de operando "LR", "HR" y "AR" no están disponibles en todas las series de controladores.

Nota

Nota relativa a los accesos de escritura:

En el tipo de datos "Bool" con los tipos de operando "I/O", "H/R", "A/R", "L/R" y "DM", se vuelve a escribir la palabra completa en el controlador tras modificar el bit indicado . No se comprueba si entretanto se han modificado otros bits en la palabra. Por este motivo, el controlador sólo puede leer la palabra indicada.

Tipo de operando de PLC antiguo	Tipo de operando PLC CS y CJ
Estado de CPU	Estado de CPU
I/O	CIO
HR	H Range 0-511
AR	A
LR	n/a 1)
DM	D
T/C	T/C
Tipo de CPU	Tipo de CPU

1) Recibirá un aviso de error si en el área LR lee o escribe en los siguientes PLC:
- CS
- CJ
- CP

Tipos de CPU soportados para Omron Host Link

Tipos de CPU

Los siguientes tipos de CPU se soportan al configurar el driver de comunicación Omron Host Link.

- CP1
  - CP1L
  - CP1H
  - CP1E
- CJ1
  - CJ1M
  - CJ1H
  - CJ1G
- CJ2
  - CJ2H

- CS1
  - CS1G
  - CS1H
  - CS1D
- CPM
  - CPM2C

Direccionamiento en Omron Host Link

Direccionamiento de controladores con Omron Host Link

En los controladores de las series CS, CP y CJ, los temporizadores 0-4095 se direccionan con T/C 0-2047.

Los contadores 0-4095 deben direccionarse con un offset de 2048 (T/C 2048-4095 equivalen a los contadores 0-2047). Los contadores y temporizadores con direcciones > 2047 no pueden direccionarse mediante Host Link.

Los contadores y temporizadores con direcciones > 2047 no pueden direccionarse mediante Host Link.

Ejemplo:

Si desea direccionar el contador C20 hay que direccionar T/C 20+2048 = T/C 2068.

Áreas de direcciones para Omron Hostlink

Omron

Áreas de direcciones	Tipos de datos							
	Bool	Byte	UInt	Int	UDInt	DInt	Real	String
I/O	I/O 0.0 - I/O 9999.15	--	I/O 0 - I/O 9999	I/O 0 - I/O 9999	--	--	--	--
HR	HR 0.0 - HR 9999.15		HR 0 - HR 9999	HR 0 - HR 9999	HR 0 - HR 9998	HR 0 - HR 9998	HR 0 - HR 9999	HR 0 - HR 9999
AR	AR 0.0 - AR 9999.15		AR 0 - AR 9999	AR 0 - AR 9999	AR 0 - AR 9998	AR 0 - AR 9998		AR 0 - AR 9999
LR	LR 0.0 - LR 9999.15		LR 0 - LR 9999	LR 0 - LR 9999	LR 0 - LR 9998	LR 0 - LR 9998		LR 0 - LR 9999
DM	DM 0.0 - DM 9999.15		DM 0 - DM 9999	DM 0 - DM 9999	DM 0 - DM 9998	DM 0 - DM 9998	DM 0 - DM 9999	DM 0 - DM 9999
T/C Bit	T/C Bit 0 - T/C Bit 4095							

Áreas de direcciones	Tipos de datos							
	Bool	Byte	UInt	Int	UDInt	DInt	Real	String
T/C Val			T/C Val 0 - T/C Val 4095	T/C Val 0 - T/C Val 4095				
Estado de CPU	RUN, MONITOR							
Tipo de CPU		Tipo de CPU						

Poner en marcha componentes

Transferir el proyecto al panel de operador

1. Ponga el panel de operador en "Modo de transferencia".
2. Ajuste todos los parámetros necesarios para la transferencia:
   - Interfaz
   - Parámetros de transferencia
   - Memoria de destino
3. Inicie la transferencia del proyecto.
   El proyecto se generará automáticamente.
   En una ventana se registran los diferentes pasos de generación y transferencia.

Interconexión del autómata y el panel de operador

1. Conecte el autómata con el panel de operador mediante un cable de conexión adecuado.
2. En el panel de operador se visualizará el aviso "Se ha establecido la conexión con el autómata".

Optimizar la configuración

Ciclo de adquisición y tiempo de actualización

Los ciclos de adquisición indicados en el software de configuración para los "punteros de área" y los ciclos de adquisición de las variables son factores decisivos para los tiempos de actualización que realmente pueden alcanzarse.

El tiempo de actualización es el ciclo de adquisición más el tiempo de transferencia más el tiempo de procesamiento.

Para obtener tiempos de actualización óptimos, tenga en cuenta los siguientes aspectos en la configuración:

- Las distintas áreas de datos deben configurarse con el mínimo tamaño posible y con el máximo necesario.
- Los ciclos de adquisición demasiado pequeños merman innecesariamente el rendimiento total. Configure el ciclo de adquisición conforme a la velocidad de modificación de los valores de proceso. Así, por ejemplo, la curva de temperatura de un horno es considerablemente más lenta que la curva de velocidad de un accionamiento eléctrico. El valor aproximativo para el ciclo de adquisición es de aprox. 1 hora.
- Disponga las variables de un aviso o una imagen en un área de datos sin dejar huecos.
- Para que las modificaciones puedan detectarse con seguridad en el controlador, deberán aplicarse al menos durante el ciclo real de adquisición

Imágenes

En las imágenes, la velocidad de actualización que puede alcanzarse realmente depende del tipo y número de los datos a representar.

Configure sólo ciclos de adquisición cortos sólo para aquellos objetos que realmente deban actualizarse rápidamente. De este modo, se reducen los tiempos de actualización.

Curvas

Si en las curvas disparadas por bit se activa el bit de grupo en el "Área de transferencia de curvas", el panel de operador actualizará siempre todas las curvas cuyo bit se haya definido en esta área. Después el panel vuelve a desactivar los bits.

Sólo después de que el panel de operador haya desactivado todos los bits, podrá activarse de nuevo el bit de grupo en el programa de control.

Peticiones de controlador

En caso de que se hayan enviado muchas peticiones de control en intervalos cortos, la comunicación entre el panel de operador y el controlador puede verse sobrecargada.

Si el panel de operador registra el valor 0 en la primera palabra de datos del buzón de peticiones, significa que éste ha aceptado la petición de control. El panel de operador procesa la petición, por lo que requerirá un cierto tiempo. Si inmediatamente después se vuelve a registrar una nueva petición de control en el buzón de peticiones, puede transcurrir algún tiempo hasta que el panel de operador ejecute la siguiente. La siguiente petición de control no se aceptará hasta que el equipo no disponga de la suficiente potencia de cálculo.

Intercambio de datos

Punteros de área con Omron

Punteros de área en conexiones mediante drivers de comunicación Omron

A través de un puntero de área se accede a un rango de datos del controlador.

Encontrará más información sobre los punteros de área y su configuración en el capítulo:

"Intercambio de datos mediante punteros de área".

Particularidades de las conexiones mediante Omron Host Link

Los punteros de área sólo pueden crearse en los "Tipos de archivo" "DM", "I/O", "HR", "AR" y "LR".

Consulte también

Intercambio de datos mediante punteros de área (Página 9118)

Curvas

Curvas

Una curva es la representación gráfica de uno o varios valores del controlador. En los Basic Panels el valor se lee disparado por tiempo.

Encontrará más información al respecto en:

Configurar el visor de curvas para los valores del controlador (Página 8499)

Curvas disparadas por tiempo

El panel de operador lee los valores de curva cíclicamente con la base de tiempo que se haya definido en la configuración.

Las curvas disparadas por tiempo se adecuan para procesos continuos, p. ej. la temperatura de servicio de un motor.

Avisos

Configurar avisos

Configurar avisos con conexiones no integradas

Para configurar avisos como advertencias, avisos de error y acuse se requieren varios pasos.

- Paso 1: crear variables
- Paso 2: Configurar avisos
- Paso 3: configurar el acuse

Se puede encontrar más información al respecto en el capítulo:

Trabajar con avisos (Página 8510)

Particularidades de la configuración de avisos

Si se configuran conexiones entre paneles de operador y controladores de otros fabricantes, hay que tener en cuenta las particularidades siguientes durante la configuración:

- Tipos de datos de la variable
- Direccionamiento de la variable
- Modo de contaje de las posiciones de bit

Tipos de datos

Para las conexiones con drivers de comunicación Omron se soportan los siguientes tipos de datos:

Controlador	Tipos de datos admisibles	
	Avisos de bit	Avisos analógicos
CP1, CJ1, CJ2, CS1, CPM	Uint, int	UInt, Int, UDInt, DInt

Modo de contaje de las posiciones de bit

Para las conexiones con drivers de comunicación Omron rige el modo de contaje siguiente:

Modo de contaje de las posiciones de bit	Byte izquierdo								Byte derecho							
En controladores Omron	15							8	7							0
Configure en WinCC:	15							8	7							0

Como variable de disparo para avisos de bit sólo se admiten las variables de los "Tipos de archivo" "DM", "I/O", "HR", "AR" y "LR".

Configurar avisos de bit

Utilice matrices para los avisos de bit y adjunte usted mismo los distintos avisos a un bit de la variable de matriz, respectivamente, y no a los distintos subelementos.

Para los avisos de bit y las matrices sólo se admiten variables de las áreas "DM", "I/O", "HR", "AR" y "LR", así como de los tipos de datos "Int" e "UInt".

Acuse de avisos

Procedimiento

Configure en el controlador las variables que correspondan para acusar una alarma. Estas variables deben asignarse a un aviso en el editor "Avisos de bit". La asignación se efectúa bajo "Propiedades > Acuse".

Se distinguen los siguientes tipos de acuse:

- Acuse desde el controlador
- Acuse en el panel de operador

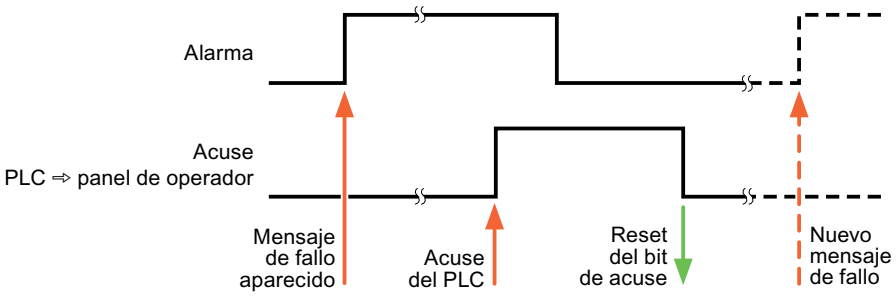
Acuse desde el controlador

En "Variable de acuse de PLC" se configuran la variable o la variable de matriz y el número de bit que el panel de operador utiliza para detectar un acuse por parte del controlador.

Un bit activado en la variable origina el acuse del bit de alarma asignado en el panel de operador. Así, un bit activado en la variable cumple la misma función que el acuse desde el panel de operador (p. ej. al pulsar la tecla "ACK").

El bit de acuse debe encontrarse en la misma variable que el bit de la alarma.

Vuelva a desactivar el bit de acuse antes de volver a activar el bit en el área de alarmas. La siguiente figura muestra el diagrama de impulsos.



Acuse en el panel de operador

En "Variable de acuse HMI" se configuran la variable o la variable de matriz y el número de bit que se escribirá en el controlador después del acuse por parte del panel de operador. Si se utilizan variables de array, hay que tener en cuenta que su longitud no supere las 6 palabras.

Para que, al activar el bit de acuse asignado de un aviso de bit de acuse obligatorio se genere siempre un cambio de señal, el panel de operador desactiva el bit de acuse asignado al aviso y escribe la variable de acuse en el controlador tan pronto como se detecta un aviso de acuse obligatorio. Puesto que el panel de operador debe procesar las operaciones, transcurre algún tiempo desde que se detecta el aviso hasta que se escribe la variable de acuse en el controlador.

Cuando se acusa en el panel de operador un aviso de bit de acuse obligatorio, se activa el respectivo bit en la variable de acuse asignada. Luego, el panel de operador escribe en el

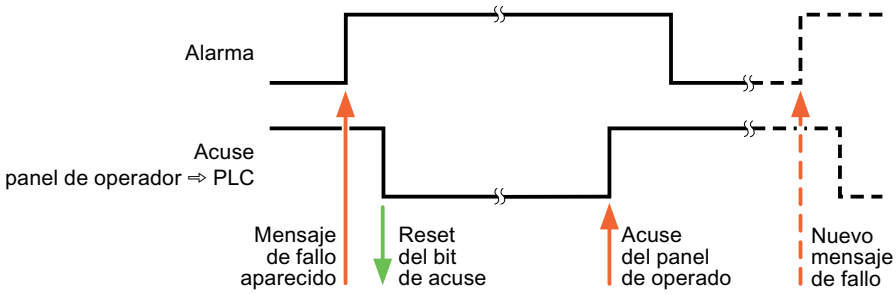
controlador la variable de acuse completa. De este modo, el controlador puede detectar que se ha acusado una alarma determinada en el panel de operador.

Nota

Todos los bits de aviso acusados desde el último inicio de runtime se conservan en la variable de acuse hasta detectarse una nueva aparición de los avisos de bit correspondientes.

El controlador sólo debería leer esta área, puesto que el panel de operador sobrescribe el área entera la próxima vez que se escribe en la variable de acuse.

La siguiente figura muestra el diagrama de impulsos.



Communication Service Packages

10.8.11.4 Intercambio de datos mediante punteros de área

Generalidades sobre los punteros de área

Introducción

A través de un puntero de área se accede a un rango de datos del controlador. El controlador y el panel de operador escriben y leen alternativamente en dichos rangos durante la comunicación.

Al evaluar los datos almacenados, el controlador y el panel de operador ejecutan acciones fijamente definidas.

Configurar punteros de área

Antes de utilizar un puntero de área, actívelos en "Conexiones ▸ Punteros de área". Seguidamente, parametrícelos.

Parámetro | Puntero de área

Activo	Nombre de visualiza...	Variable PLC	Tipo de acceso	Direcci...	Longitud	Modo de adquisi...	Ciclo de adquisici...	Comentar
	Coordinación	<No definido>	<acceso simbólico>		1	Cíclico continuo	<No definido>	
	Fecha/hora	<No definido>	<acceso simbólico>		6	Cíclico continuo	<No definido>	
	Petición del controla...	<No definido>	<acceso simbólico>		4	Cíclico continuo	<No definido>	
	Registro	<No definido>	<acceso simbólico>		5	Cíclico continuo	<No definido>	

Punteros de área generales del panel de operador

Conexión	Nombre de visualiza...	Variable PLC	Tipo de acceso	Dirección	Longitud	Modo de adquisi...	Ciclo de adquisic...	Comenta
<No definido>	Identificación del pro...	<No definido>	<acceso simbólico>		1	Cíclico continuo	<No definido>	
<No definido>	Número de imagen	<No definido>	<acceso simbólico>		5	Cíclico continuo	<No definido>	
<No definido>	Fecha/hora PLC	<No definido>	<acceso simbólico>		6	Cíclico continuo	<No definido>	

- Activo
  Activa el puntero de área.
- Nombre de visualización
  Nombre del puntero de área predeterminado por WinCC.
- Variable de controlador
  Aquí se selecciona la variable de controlador o la matriz de variables que se ha configurado como rango de datos para el puntero de área.
- Dirección
  A causa del direccionamiento simbólico, en este campo no aparecerá ninguna dirección.
- Longitud
  La longitud del puntero de área viene dada por WinCC.
- Ciclo de adquisición
  Determine aquí el ciclo de adquisición para punteros de área que serán leídos por el panel de operador. Tenga en cuenta que un tiempo de adquisición muy corto podría repercutir negativamente en el rendimiento del panel de operador.
- Comentario
  Introduzca un comentario p. ej. relativo a la utilización del puntero de área.

Acceder a los rangos de datos

Acceder a los rangos de datos

La tabla siguiente muestra cómo el panel de operador y el controlador acceden a los distintos rangos de datos, bien sea leyendo (R) o escribiendo (W) en ellos.

Rango de datos	Necesario para	Panel de operador	Controlador
Número de imagen	Evaluación desde el controlador cuya imagen está abierta en ese momento.	W	R
Registro	Transferencia de registros con sincronización	R/W	R/W
Fecha/hora	Transferencia de la fecha y hora del panel de operador al controlador	W	R
Fecha/hora PLC	Transferencia de la fecha y hora del controlador al panel de operador	R	W
Coordinación	Consultar el estado del panel de operador en el programa de control	W	R
Identificación del proyecto	Runtime comprueba si la identificación del proyecto de WinCC y el proyecto del controlador son coherentes.	R	W
Petición de control	Activación de funciones en el panel de operador a través del programa de control	R/W	R/W

Puntero de área "Número de imagen"

Función

Los paneles de operador depositan en el puntero de área "Número de imagen" información acerca de la imagen visualizada en el panel.

Ello permite transferir al controlador información acerca del contenido actual de la imagen en el panel de operador. En el controlador se pueden disparar determinadas reacciones, p. ej. la llamada de otra imagen.

Utilización

Antes de poder utilizar el puntero de área "Número de imagen", es necesario configurarlo y activarlo bajo "Comunicación > Conexiones". El puntero de área "Número de imagen" se puede crear únicamente en **un** controlador y depositarse allí **una sola vez**.

El número de imagen se transfiere siempre al controlador cuando se activa una nueva imagen o cuando el foco cambia de un objeto de imagen a otro dentro de una imagen.

Estructura

Este puntero de área es un área de datos de la memoria del controlador con una longitud fija de 5 palabras.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1. Palabra	Tipo de imagen actual															
2. Palabra	Número de imagen actual															
3. Palabra	Reservada															
4ª palabra	Número de campo actual															
5. Palabra	Reservada															

- Tipo de imagen actual
  "1" = imagen raíz
  "4" = ventana permanente
- Número de imagen actual
  1 a 32767
- Número de campo actual
  1 a 32767

Puntero de área "Fecha/hora"

Función

Este puntero de área sirve para transferir la fecha y la hora del panel de operador al controlador.

El controlador escribe la petición de control "41" en la bandeja de peticiones.

Al evaluar la petición de control, el panel de operador escribe su fecha y hora actuales en el área de datos configurada en el puntero de área "Fecha/hora". Todos los datos se expresan en formato BCD.

Si hay varias conexiones configuradas en un proyecto y en una de ellas se debe utilizar el puntero de área "Fecha / hora", éste deberá activarse para cada conexión configurada.

Nota

Si ha configurado el puntero de área "Fecha/hora", no podrá utilizar el puntero de área "Fecha/hora PLC".

El área de datos "Fecha/hora" tiene la siguiente estructura:

Palabra de datos	Byte izquierdo								Byte derecho								
	15							8	7							0	
n+0	Reservada								Hora (0 a 23)								Hora
n+1	Minuto (0 a 59)								Segundo (0 a 59)								
n+2	Reservada								Reservada								

Palabra de datos	Byte izquierdo								Byte derecho								
	15							8	7							0	
n+3	Reservada								Día de la semana (1 a 7, 1=domingo)								Fecha
n+4	Día (1 a 31)								Mes (1 a 12)								
n+5	Año (80 a 99/0 a 29)								Reservada								

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Puntero de área "Fecha/hora PLC"

Función

Este puntero de área sirve para transferir la fecha y la hora del controlador al panel de operador. Este puntero de área se utiliza si el controlador es el maestro que determina la hora.

El controlador carga el área de datos del puntero de área. Todos los datos se expresan en formato BCD.

Según el ciclo de adquisición configurado, el panel de operador lee los datos cíclicamente y se sincroniza.

Nota

No configure un ciclo de adquisición demasiado breve para el puntero de área "Fecha/hora", puesto que ello afecta el rendimiento del panel de operador.
Recomendación: Ciclo de adquisición = 1 minuto (si el proceso permite dicho ciclo).

"Fecha/hora PLC" es un puntero de área global y solo se puede configurar una vez en cada proyecto.

Nota

Si ha configurado el puntero de área "Fecha/hora PLC", no podrá utilizar el puntero de área "Fecha/hora".

El área de datos "Fecha/hora PLC" tiene la siguiente estructura:

Palabra de datos	Byte izquierdo			Byte derecho		
	15	. . . . .	8	7	. . . . .	0
n+0	Año (80 a 99/0 a 29)			Mes (1 a 12)		
n+1	Día (1 a 31)			Hora (0 a 23)		

Palabra de datos	Byte izquierdo			Byte derecho		
	15	. . . . .	8	7	. . . . .	0
n+2	Minuto (0 a 59)			Segundo (0 a 59)		
n+3	Reservada			Reservada		Día de la semana (1 a 7, 1=domingo)
n+4 [1]	Reservada			Reservada		
n+5 [1]	Reservada			Reservada		

1) Ambas palabras de datos deben existir en el área de datos para garantizar la concordancia del formato de datos de WinCC y evitar la lectura de información errónea.

Nota

Al introducir el año, tenga en cuenta que los valores 80 a 99 corresponden a los años 1980 a 1999, en tanto que los valores 0 a 29 equivalen a los años 2000 a 2029.

Puntero de área "Coordinación"

Función

El puntero de área "Coordinación" sirve para realizar las funciones siguientes:

- Detectar en el programa de control el arranque del panel de operador
- Detectar en el programa de control el modo de operación actual del panel de operador
- Detectar en el programa de control si el panel de operador está dispuesto para comunicarse

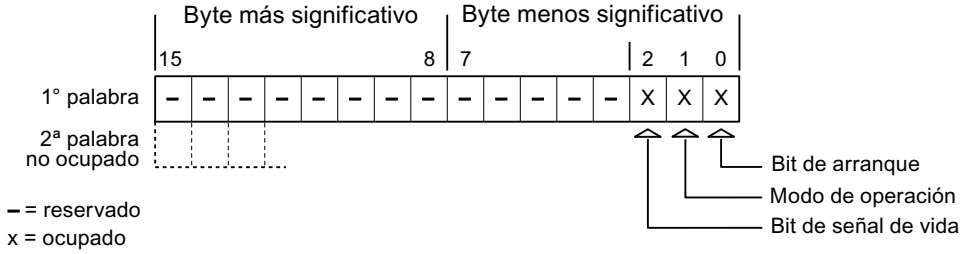
El puntero de área "Coordinación" tiene una longitud de una palabra.

Utilización

Nota

Cada vez que el panel de operador actualiza el puntero de área, se escribe siempre toda el área de coordinación.
Por tanto, el programa de control no puede efectuar cambios en el área de coordinación.

Ocupación de los bits en el puntero de área "Coordinación"



Bit de arranque

Durante el arranque, el panel de operador pone el bit de arranque a "0" por breve tiempo. Después del arranque, el bit se pone a "1" de forma permanente.

Modo de operación

Cuando el usuario conmuta el panel de operador a modo "offline", el bit de modo de operación de pone a "1". En el servicio normal del panel de operador, el estado del bit de modo de operación es "0". Consultando este bit en el programa de control es posible averiguar el modo de operación actual del panel de operador.

Bit de señal de vida

El panel de operador invierte el estado del bit de señal de vida en intervalos de aprox. 1 segundo. Consultando este bit en el programa de control es posible averiguar si todavía existe una conexión con el panel de operador.

Puntero de área "Identificación de proyecto"

Función

Al iniciar runtime puede comprobarse si el panel de operador está conectado al autómata correcto. Esta comprobación es importante si se utilizan varios paneles de operador.

Para ello, el panel de operador compara un número almacenado en el autómata con el valor indicado en la configuración. Así puede garantizarse la compatibilidad de los datos de configuración con los del programa de control. Si no hay compatibilidad, se muestra un aviso de sistema en el panel de operador y se detiene el Runtime.

Utilización

Para utilizar este puntero de área, especifique en la configuración lo siguiente:

- Indicación de la versión de la configuración. El valor puede estar comprendido entre 1 y 255.
  La versión se indica en el editor "Configuración de runtime > General" en el área "Identificación".
- Dirección de datos del valor para la versión que está almacenada en el autómata:
  la dirección se indica en el editor "Comunicación > Conexiones".

Fallo de una conexión

Al fallar la conexión con un panel de operador en el que esté configurado el puntero de área "Identificación de proyecto", se desactivarán también todas las demás conexiones del proyecto.

Para este comportamiento se deben cumplir los requisitos siguientes:

- En el proyecto se han configurado varias conexiones.
- En por lo menos una conexión se está utilizando el puntero de área "Identificación de proyecto".

Las causas siguientes pueden conmutar las conexiones al estado "offline":

- No se puede acceder al autómata.
- La conexión se ha conmutado a "offline" en el sistema de ingeniería.

Puntero de área "Tarea de control"

Función

Desde la bandeja de peticiones de control se pueden lanzar peticiones al panel de operador y con ello disparar acciones en el panel. A estas funciones pertenecen p. ej.:

- Mostrar imagen
- Ajustar la fecha y hora

Estructura de datos

En la primera palabra de la bandeja de peticiones de control figura el número de la petición. Dependiendo de la petición en cuestión, pueden transferirse hasta 3 parámetros como máximo.

Palabra	Byte izquierdo	Byte derecho
n+0	0	Número de petición
n+1	Parámetro 1	

Palabra	Byte izquierdo	Byte derecho
n+2	Parámetro 2	
n+3	Parámetro 3	

Si la primera palabra de la bandeja de peticiones de control es diferente de 0, el panel de operador evaluará la petición de control. Por este motivo hay que introducir primero los parámetros en la bandeja de peticiones de control y sólo entonces el número de la petición.

Cuando el panel de operador haya aceptado la petición de control, la primera palabra se pondrá de nuevo a 0. Por lo general, la ejecución de la petición de control no se habrá completado todavía.

Peticiones de control

A continuación se indican todas las peticiones de control y sus respectivos parámetros. La columna "N.º" contiene el número de petición de control. Por regla general, las peticiones de control sólo pueden ser disparadas por el controlador si el panel de operador está en modo online.

Nota

No todos los paneles de operador soportan las peticiones de control.

N.°	Función	
14	**Ajustar la hora (codificada en BCD)**	
	Parámetro 1	Byte izquierdo: - Byte derecho: horas (0-23)
	Parámetro 2	Byte izquierdo: Minutos (0-59) Byte derecho: Segundos (0-59)
	Parámetro 3	-
15	**Ajustar la fecha (codificada BCD)**	
	Parámetro 1	Byte izquierdo: - Byte derecho: Día de la semana (1-7: Domingo-Sábado)
	Parámetro 2	Byte izquierdo: Día (1-31) Byte derecho: Mes (1-12)
	Parámetro 3	Byte izquierdo: Año
23	**Iniciar la sesión**	
	Inicia la sesión del usuario con el nombre "PLC User" y el número de grupo transferido en el parámetro 1 en el panel de operador. Para poder iniciar la sesión es necesario que el número de grupo transferido exista en el proyecto.	
	Parámetro 1	Número de grupo 1 - 255
	Parámetro 2, 3	-
24	**Cerrar la sesión**	
	Cierra la sesión del usuario actual. (Esta función equivale a la función de sistema "CerrarSesión")	
	Parámetro 1, 2, 3	-
40	**Transferir fecha/hora al controlador**	

N.°	Función	
14	**Ajustar la hora (codificada en BCD)**	
	(En formato S7 DATE_AND_TIME) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
41	**Transferir fecha/hora al controlador**	
	(En formato OP/MP) Entre dos peticiones deben transcurrir por lo menos 5 segundos para evitar que se sobrecargue el panel de operador.	
	Parámetro 1, 2, 3	-
46	**Actualizar variable**	
	Hace que el panel de operador lea el valor actual de la variable del controlador cuyo identificador de actualización coincide con el valor transferido en el parámetro 1. (Esta función equivale a la función de sistema "ActualizarVariable")	
	Parámetro 1	1 - 100
49	**Borrar búfer de avisos**	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Warnings" del búfer de avisos.	
	Parámetro 1, 2, 3	-
50	**Borrar búfer de avisos**	
	Borra todos los avisos analógicos y todos los avisos de bit de la categoría "Errors" del búfer de avisos.	
	Parámetro 1, 2, 3	-
51	**Selección de imagen** [1]	
	Parámetro 1	Número de imagen
	Parámetro 2	-
	Parámetro 3	Número de campo
69	**Leer registro del controlador**	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	0: No sobrescribir el registro existente 1: Sobrescribir el registro existente
70	**Escribir registro en el controlador**	
	Parámetro 1	Número de receta (1-999)
	Parámetro 2	Número de registro (1-65535)
	Parámetro 3	-

[1] En los paneles de operador OP 73, OP 77A y TP 177A la petición de control "Selección de imagen" se ejecuta incluso con el teclado de pantalla abierto.

Puntero de área "Registro"

Puntero de área "Registro"

Función

Al transferir registros entre el panel de operador y el autómata, ambos interlocutores acceden alternativamente a las áreas de comunicación conjuntas del autómata.

Tipos de transferencia

Existen dos posibilidades para transferir registros entre el panel de operador y el autómata, a saber:

- Transferencia sin sincronización
- Transferencia con sincronización a través de la bandeja de datos

Los registros se transfieren siempre directamente. Por tanto, los valores de variables se leen o escriben directamente de/en la dirección configurada para la variable, sin desviarse al portapapeles.

Iniciar la transferencia de registros

Para iniciar la transferencia existen tres posibilidades:

- Inicio desde la vista de recetas
- Tareas de control
  El autómata también puede disparar la transferencia de registros.
- Activar funciones configuradas

Si la transferencia de registros se dispara mediante una función configurada o una orden de control, la vista de recetas seguirá pudiéndose controlar sin impedimentos desde el panel de operador. Los registros se transfieren en segundo plano.

Sin embargo, no es posible procesar simultáneamente varias peticiones de transferencia. En este caso, el panel de operador deniega una segunda transferencia con un aviso de sistema.

Transferencia sin sincronización

En la transferencia asíncrona de registros entre el panel de operador y el autómata no se coordinan las áreas de datos utilizadas conjuntamente. Por ello, no es necesario crear un área de datos en la configuración.

La transferencia asíncrona de registros resulta útil p. ej. en los casos siguientes:

- Condicionado por el sistema, puede descartarse que los interlocutores sobrescriban los datos de forma incontrolada.
- El autómata no necesita información sobre el número de receta o el de registro.
- La transferencia de registros es iniciada por el usuario desde el panel de operador.

Leer valores

Al iniciar la transferencia de lectura se leen los valores de las direcciones de control y se transfieren al panel de operador.

- Inicio por el usuario desde la vista de recetas:
  Los valores se cargan en el panel de operador. Éstos se pueden procesar posteriormente en el panel de operador, p. ej. modificar o guardar los valores, etc.
- Inicio mediante una función u orden de control:
  Los valores se almacenan inmediatamente en el soporte de datos.

Escribir valores

Al iniciar la transferencia para escritura se escriben los valores en las direcciones de control.

- Inicio por el usuario desde la vista de recetas:
  Los valores actuales se escriben en el autómata.
- Inicio mediante una función u orden de control:
  Los valores del soporte de datos se escriben en el autómata.

Transferencia con sincronización

En la transferencia sincronizada, ambos interlocutores activan determinados bits de estado en el área de datos utilizada conjuntamente. De este modo se impide que los datos se sobrescriban unos a otros de forma incontrolada en el programa de control.

Aplicación

La transferencia síncrona de registros resulta útil p. ej. en los casos siguientes:

- El controlador es el "interlocutor activo" en la transferencia de registros.
- En el controlador se evalúa información sobre los números de receta y de registro.
- La transferencia de registros se dispara mediante una petición del controlador.

Requisitos

Para que los registros puedan transferirse de forma sincronizada entre el panel de operador y el controlador, en el proceso de configuración deben cumplirse las siguientes condiciones:

- Se ha configurado un puntero de área: Editor "Comunicación > Conexiones" bajo "Punteros de área".
- En la receta está indicado el controlador con el que el panel de operador sincroniza la transferencia de registros:
  Editor "Recetas" en el área "General > Sincronización > Configuración" de la ventana de inspección, y selección "Transferencia de datos coordinada" activada.

Estructura del área de datos

El área de datos tiene una longitud fija de 5 palabras. El área de datos está estructurada de la siguiente manera:

	15		0
1. palabra	Número de receta actual (1-999)		
2. palabra	Número de registro actual (0-65.535)		
3. palabra	Reservada		
4. palabra	Estado (0, 2, 4, 12)		
5. palabra	Reservada		

- Estado
  La palabra de estado (palabra 4) acepta los siguientes valores:

Valor		Significado
Decimal	Binario	
0	0000 0000	Transferencia permitida, bandeja de datos libre
2	0000 0010	Transferencia en curso.
4	0000 0100	Transferencia finalizada sin fallos
12	0000 1100	Transferencia finalizada con fallos

Ejecución de la transferencia al iniciarla desde la vista de recetas

Leer del autómata desde la vista de recetas

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta que ha de ser leído, así como el estado "Transferencia en curso" en la bandeja de datos y repone el número de registro a 0.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del autómata y los muestra en la vista de recetas. En las recetas con variables sincronizadas, los valores del autómata también se escriben en las variables.	
4	El panel de operador activa el estado "Transferencia finalizada".	
5	Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Escribir en el autómata desde la vista de recetas

Paso	Acción	
	Verificación: ¿Palabra de estado = 0?	
1	Sí	No
	El panel de operador introduce el número de receta y el número de registro que deben escribirse, así como como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
2	El panel de operador escribe los valores actuales en el autómata. En las recetas con variables sincronizadas, los valores modificados son sincronizados entre la vista de recetas y las variables, escribiéndose luego en el autómata.	
3	El panel de operador activa el estado "Transferencia concluida".	
4	Dado el caso, el programa de control puede evaluar entonces los datos transferidos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Nota

El panel de operador es el único que puede activar la palabra de estado. El autómata sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el autómata sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos está activado a "Transferencia finalizada".
- El estado en la bandeja de datos está activado a "Transferencia finalizada con fallos".

Ejecución de la transferencia en caso de tarea de control

El panel de operador o el autómata pueden iniciar la transferencia de registros entre sí.

Para este tipo de transferencia se dispone de las tareas de control n° 69 y n° 70.

N° 69: Leer registro del autómata ("SPS → DAT")

La orden de control n° 69 transfiere los registros del autómata al panel de operador. La orden de control está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	69
2ª palabra	Número de receta (1-999)	

	Byte izquierdo (LB)	Byte derecho (RB)
3ª palabra	Número de registro (1-65.535)	
4ª palabra	No sobrescribir el registro existente: 0 Sobrescribir el registro existente: 1	

N° 70: Escribir registro en el autómata ("DAT → SPS")

La orden de control n° 70 transfiere los registros del panel de operador al autómata. La orden de control está estructurada de la siguiente manera:

	Byte izquierdo (LB)	Byte derecho (RB)
1ª palabra	0	70
2ª palabra	Número de receta (1-999)	
3ª palabra	Número de registro (1-65.535)	
4ª palabra	—	

Ejecución al leer del autómata mediante la orden de control "SPS → DAT" (n° 69)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la tarea, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la orden de control.	
4	• Si en la tarea se ha seleccionado "Sobrescribir", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia finalizada". • Si en la tarea se ha seleccionado "No sobrescribir" y ya existe el registro, el panel de operador cancelará la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Ejecución al escribir en el autómata mediante la orden de control "DAT → SPS" (n° 70)

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la tarea, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación sin respuesta.
3	El panel de operador toma los valores del registro indicado en la tarea del portador de datos y los escribe en el autómata.	

Paso	Acción	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Ejecución de la transferencia con inicio mediante una función configurada

Leer en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de operador lee los valores del autómata y los almacena en el registro que se haya indicado en la función.	
4	• Si en la función "Sobrescribir" se ha seleccionado "Sí", se sobrescribirá un registro existente sin previa consulta. El panel de operador activa el estado "Transferencia finalizada". • Si en la función "Sobrescribir" se ha seleccionado "No" y el registro ya existe, el panel de operador interrumpirá la operación e introducirá 0000 1100 en la palabra de estado de la bandeja de datos.	
5	Para permitir una nueva transferencia, el programa de control debe volver a poner la palabra de estado a 0.	

Escribir en el autómata mediante una función configurada

Paso	Acción	
1	Verificación: ¿Palabra de estado = 0?	
	Sí	No
2	El panel de operador introduce el número de receta y de registro indicados en la función, así como el estado "Transferencia en curso" en la bandeja de datos.	Cancelación con aviso de sistema.
3	El panel de control recoge los valores del registro indicado en la función del soporte de datos y los escribe en el autómata.	
4	El panel de operador activa el estado "Transferencia concluida".	
5	El programa de control puede evaluar entonces los datos transferidos. Para permitir una nueva transferencia, el programa de control debe reponer la palabra de estado a 0.	

Posibles causas de error en la transferencia de registros

Causas de error posibles

Si la transferencia de registros termina con errores, puede deberse a las siguientes causas:

- No se ha configurado ninguna dirección de variable en el autómata
- No es posible sobrescribir registros
- El número de receta no existe
- El número de registro no existe

Nota

El panel de operador es el único que puede activar la palabra de estado. El autómata sólo puede desactivar (poner a "0") la palabra de estado.

Nota

Para garantizar la coherencia de los datos, la evaluación de los números de receta y de registro en el autómata sólo deberá efectuarse si se cumple una de las condiciones siguientes:

- El estado en la bandeja de datos se ha activado a "Transferencia finalizada".
- El estado en la bandeja de datos se ha activado a "Transferencia finalizada con fallos".

Reacción a una interrupción debida a un fallo

El panel de operador reacciona a una interrupción de la transferencia de registros debida a un fallo de la siguiente forma:

- Inicio por el usuario desde la vista de recetas:
  Indicaciones en la barra de estado de la vista de recetas y salida de avisos de sistema
- Inicio mediante una función
  Emisión de avisos de sistema
- Inicio mediante una orden de control
  No hay respuesta en el panel de operador

Independientemente de ello, el usuario puede evaluar el estado de la transferencia consultando la palabra de estado en la bandeja de datos.

10.9 Utilizar funciones extendidas

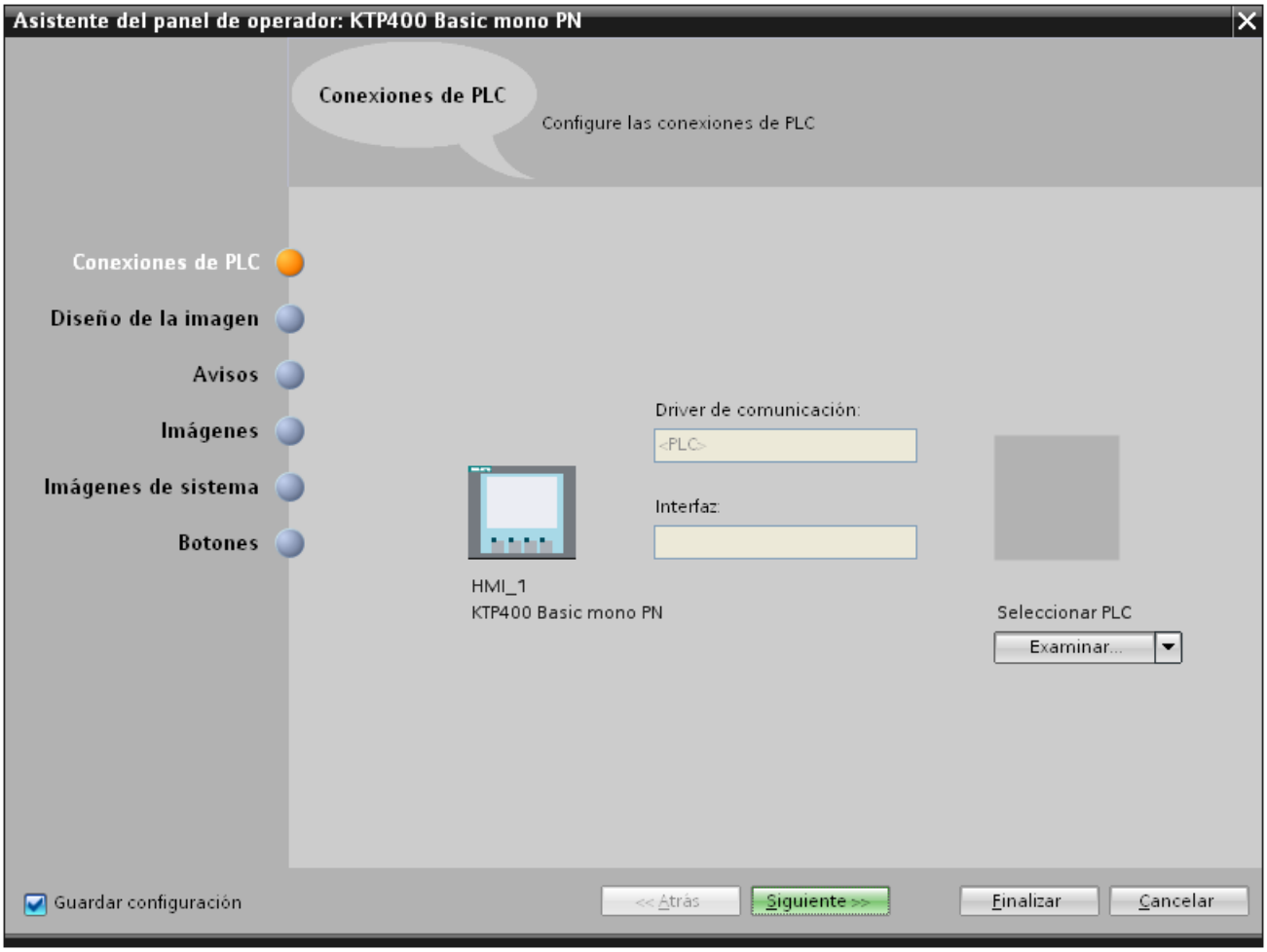
10.9.1 Conceptos básicos del asistente de paneles de operador

Introducción

Cuando se crea un nuevo panel de operador en el proyecto, se inicia automáticamente el asistente de paneles de operador.

Asistente de paneles de operador

El asistente de paneles de operador guía al usuario paso a paso por los diversos cuadros de diálogo y le ayuda a crear un dispositivo. Además, permite definir opciones de configuración básicas para el panel de operador, como puede ser la representación de la imagen o la conexión con el PLC.



10.9.2 Trabajar con librerías

10.9.2.1 Principios básicos de las librerías

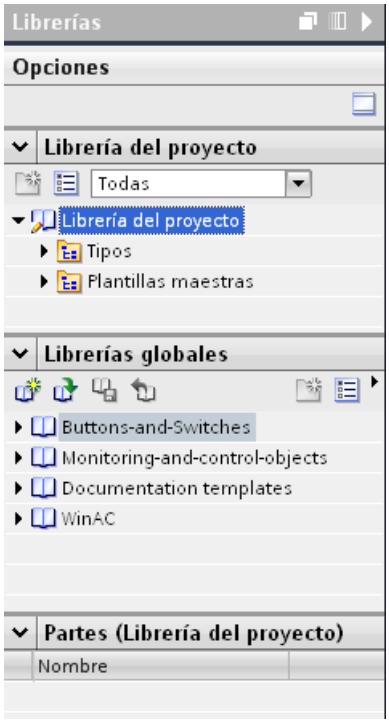
Introducción

En las librerías se depositan de forma centralizada los objetos que se necesitan con frecuencia. Un objeto depositado en la librería sólo tiene que configurarse una vez. Posteriormente puede utilizarse las veces que se quiera. Los objetos de la librería amplían el número de objetos de imagen disponibles y aumentan la efectividad de la configuración gracias a la utilización múltiple de objetos acabados.

Junto con WinCC se suministran librerías completas, p. ej., de "Motores" o "Válvulas". No obstante, también es posible definir objetos de librerías propios.

Las librerías se administran en la Task Card "Librerías" o en la vista de librería. Tiene a su disposición las siguientes librerías:

- Librería del proyecto
- Librerías globales



Nota

En la Task Card "Herramientas" de la paleta "Gráficos" hay una librería de símbolos.

Librería del proyecto

Cada proyecto contiene exactamente una librería. Los objetos de la librería del proyecto se guardan junto con los datos del mismo, estando sólo disponibles en el proyecto en el que se ha creado la librería. Si el proyecto se transfiere a otro PC, también se transfiere la librería correspondiente.

Para poder aprovechar un objeto de la librería del proyecto en otros proyectos, desplace o copie el objeto a una librería global.

Librerías globales

Las librerías globales se guardan de forma independiente de los datos del proyecto en un archivo propio con la extensión *.al12.

Un proyecto puede acceder a varias librerías globales. Varios proyectos pueden utilizar simultáneamente una misma librería global.

Si un objeto de la librería se modifica en un proyecto, la librería se modificará también en los demás proyectos que hayan abierto esta librería.

Objetos de librerías

Una librería puede contener todos los objetos de WinCC. Ejemplos:

- Panel de operador completo
- Imágenes
- Objetos de manejo y visualización, incluidas las variables y funciones
- Gráficos
- Variables
- Avisos
- Listas de textos y gráficos
- Faceplates
- Tipos de datos de usuario

Consulte también

Plantillas maestras y tipos (Página 9139)

10.9.2.2 Sinopsis de la vista de librerías

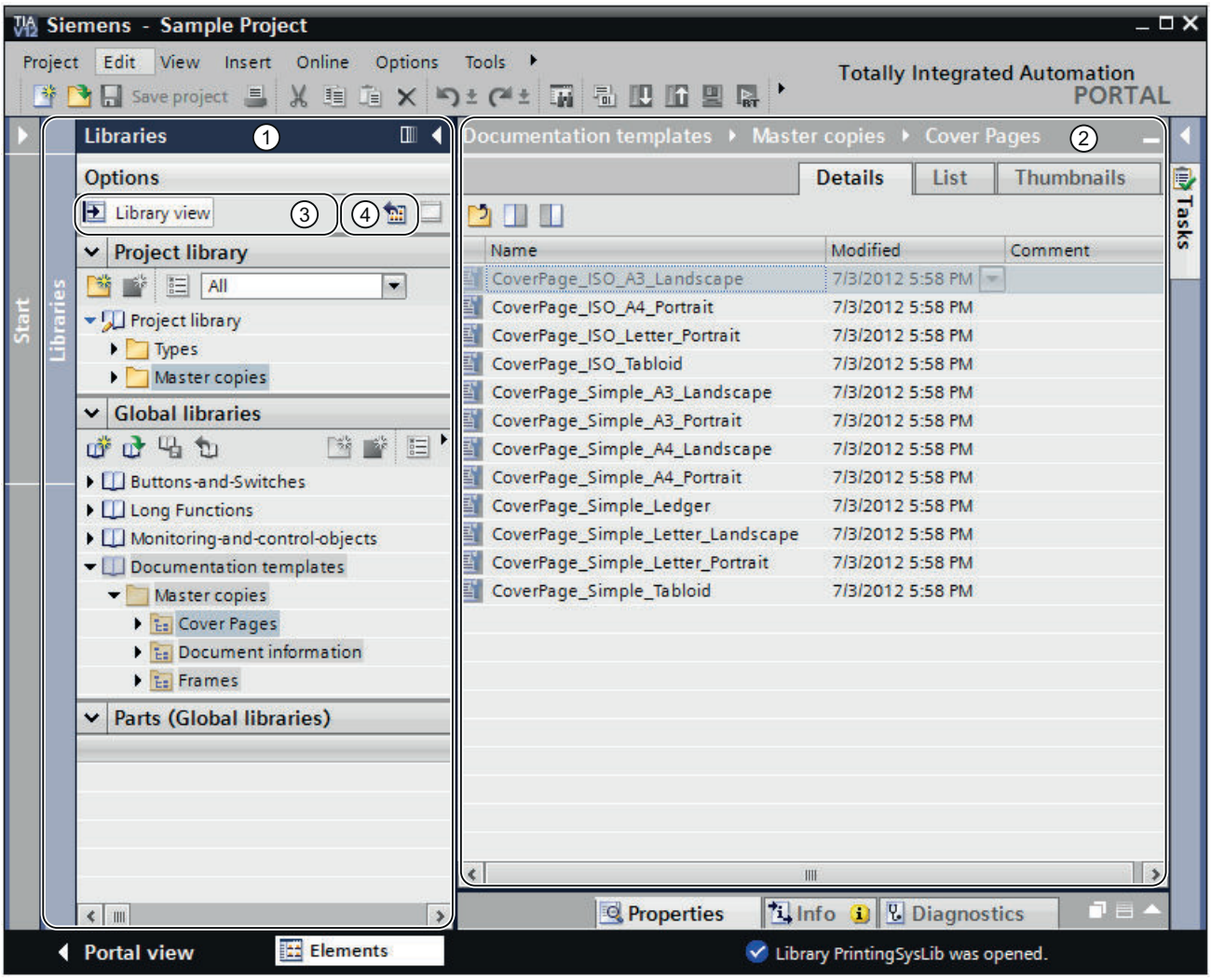
Función de la vista de librería

La vista de librería unifica las funciones de la Task Card "Librerías" y las de la ventana general. En la vista de librería se visualizan los elementos de una librería en diferentes vistas y, además,

propiedades adicionales de los distintos elementos en la vista detallada. Aparte de esto, en la vista de librería se editan y versionan los tipos.

Estructura de la vista de librería

La figura siguiente muestra los componentes de la vista de librería:



1 Árbol de librerías
2 Vista general de librerías
3 Botón "Vista de librería"
4 Botón "Abrir o cerrar vista general de librerías"

Árbol de librerías

Salvo pequeñas diferencias, el árbol de librerías es igual que la Task Card "Librerías". Al contrario que en la Task Card, no hay una paleta "Elementos", ya que los elementos se

muestran en la vista general de librerías. Además, en el árbol de librerías se puede cerrar la vista de librería o abrir y cerrar la vista general de librerías.

Vista general de librerías

La vista general de librerías se corresponde con la ventana general y muestra los elementos del objeto seleccionado en el árbol de librerías. Los elementos pueden visualizarse en tres vistas diferentes. Además, en la vista general de librerías pueden realizarse, p. ej., las siguientes acciones:

- Copiar elementos
- Desplazar elementos
- Versionar tipos
- Editar faceplates y tipos de datos de usuario HMI
- Editar instancias de tipos

10.9.2.3 Plantillas maestras y tipos

Introducción

Tanto la "Librería de proyecto" como la "Librería global" tienen las carpetas "Plantillas maestras" y "Tipos". Los objetos de librería se pueden crear o utilizar como plantilla maestra o como tipo.

Plantillas maestras

Para crear copias de los objetos de librería independientes las unas de las otras, utilice las plantillas maestras.

Tipos

Desde los objetos de la carpeta "Tipos" se crean instancias y se utilizan en su proyecto. Las instancias están unidas a su tipo respectivo.

Administrar los objetos de librería

Los objetos de librería se copian y desplazan a otra librería. Las plantillas maestras sólo se copian a la carpeta "Plantillas maestras" o a cualquier otra subcarpeta de "Plantillas maestras". Los tipos sólo se pueden insertar igualmente en la carpeta "Tipos" o en cualquier otra subcarpeta de "Tipos".

Consulte también

Principios básicos de las librerías (Página 9134)

Estado de las versiones de tipo (Página 9154)

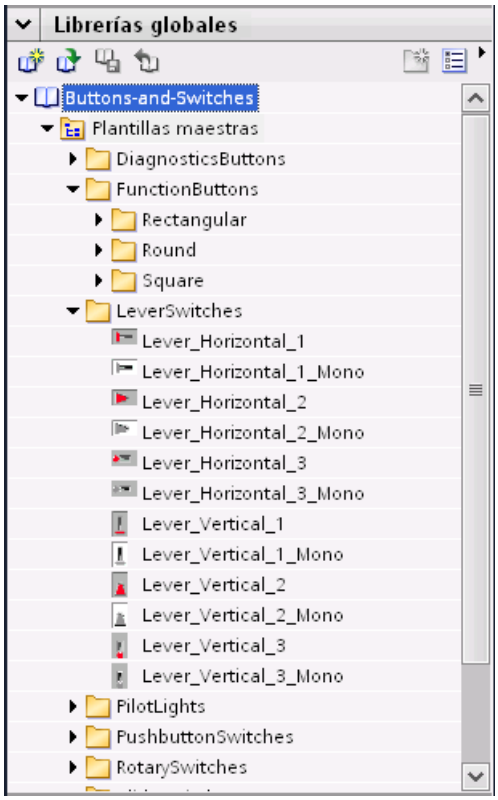
10.9.2.4 Librerías en WinCC

Introducción

WinCC se suministra con un gran número de librerías. Clasificados por temas en carpetas, se pueden encontrar objetos gráficos prediseñados, que pueden utilizarse en las imágenes para el manejo y la observación de la instalación.

Librería global "Buttons and Switches"

Las librerías "Buttons and Switches" ofrecen una amplia selección de interruptores y botones.



Las carpetas clasifican los interruptores o botones en categorías. La carpeta "DiagnosticsButtons" contiene, por ejemplo, el objeto "Indicador del diagnóstico de sistema". El objeto "Indicador del diagnóstico de sistema" se utiliza para realizar diagnósticos de sistema en la instalación.

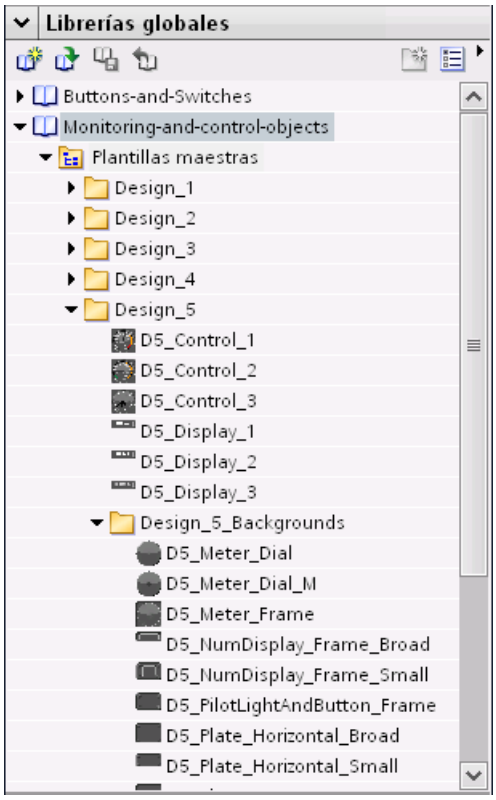
Nota

Los objetos de la carpeta "DiagnosticsButtons" sólo pueden utilizarse en Comfort Panels.

Los objetos con la denominación "Switch" en el nombre del objeto o en el nombre de carpeta correspondiente no pueden utilizarse en runtime Professional.

Librería global "Monitoring and Control objects"

La librería "Monitoring and Control objects" ofrece objetos de control y visualización más o menos complejos en diversos diseños, así como lámparas de control, botones e interruptores correspondientes.



Además, en la carpeta "Design_Backgrounds" hay visores de gráficos adecuados a los diseños, que pueden utilizarse como fondos de objeto para la ampliación específica de cliente del volumen de la librería.

Nota

Los objetos con la denominación "Switch" en el nombre del objeto no pueden utilizarse en Runtime Professional. Lo mismo rige para el objeto "D5_Display_3" con el campo de fecha y hora contenido en el mismo.

10.9.2.5 Administrar librerías

Sinopsis de la administración de librerías

Función de la administración de librerías

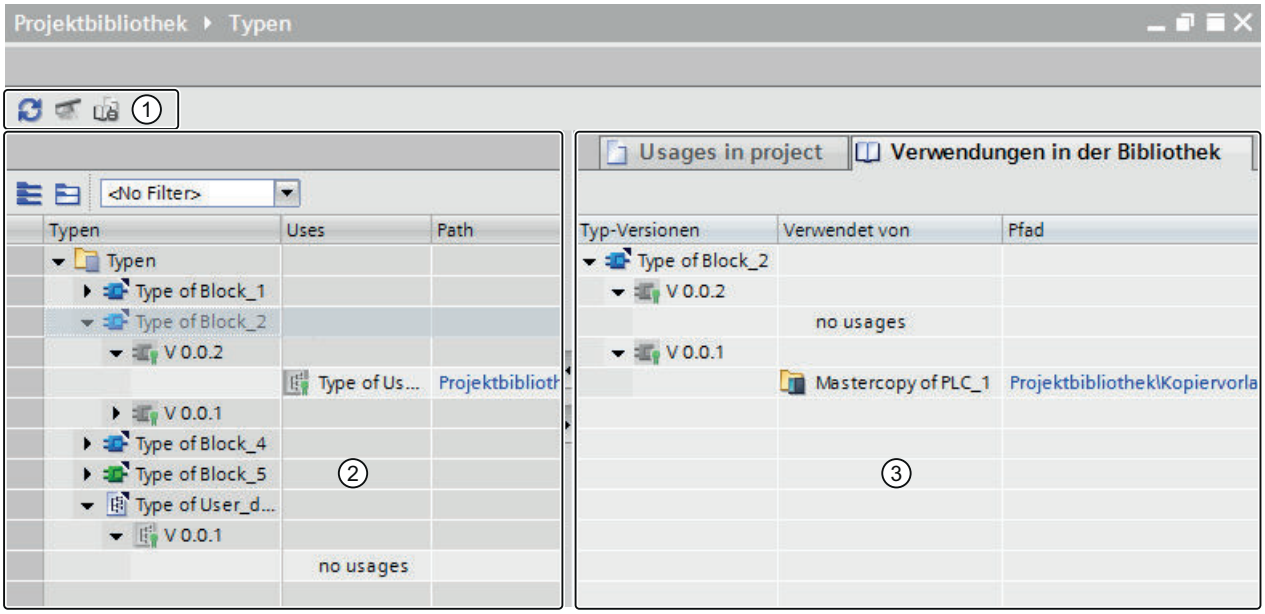
Las plantillas maestras y los tipos con dependencias de otros elementos de librerías están sujetos a algunas restricciones de funciones. Por ejemplo, no es posible borrarlos mientras existan dependencias. De este modo se impide que se inutilicen otros elementos de la librería. La administración de librerías permite identificar dependencias y proporciona una visión global del progreso del trabajo.

La administración de librerías ofrece las siguientes funciones:

- Visualización de las relaciones de los tipos y las plantillas maestras
  Si un tipo se referencia a otros tipos o a plantillas maestras, las relaciones se muestran en la administración de librerías. Asimismo se muestran los elementos de librerías a los que hace referencia un tipo o una plantilla maestra.
- Indicación de las ubicaciones de tipos en el proyecto
- Filtrado de todos los tipos que se encuentran en el estado "en test" o "en proceso"

Estructura de la administración de librerías

La figura siguiente muestra los componentes de la administración de librerías:



① Barra de herramientas de la administración de librerías
② Área "Tipos"
③ Área "Usos"

Barra de herramientas de la administración de librerías

La barra de herramientas de la administración de librerías permite realizar las siguientes acciones:

- Actualizar la vista
  Si se ha modificado el proyecto, actualice la vista de la administración de librerías.
- Limpiar librería
  Es posible limpiar la librería del proyecto y las librerías globales. Al limpiar una librería se borran todos los tipos y todas las versiones de tipos que no están enlazados con una instancia del proyecto.
- Armonizar proyecto
  Al armonizar un proyecto, los nombres y las estructuras de rutas de utilización de tipos en el proyecto se adaptan a los correspondientes nombres y estructuras de rutas de los tipos dentro de una librería.

Área "Tipos"

En el área "Tipos" se muestra el contenido de la carpeta que se ha seleccionado en la vista de librería. Mediante los botones de la barra de herramientas del área "Tipos" es posible abrir o cerrar todos los tipos. Además es posible filtrar la vista con la lista desplegable "Filtro" y ver exclusivamente todos los tipos con el estado "en test" o "en proceso". Con cada tipo se muestran también los tipos a los que hace referencia.

Área "Usos"

En el área "Usos" se muestra una relación de las ubicaciones de los tipos y plantillas maestras seleccionados. El área "Usos" se divide en dos fichas:

- Ficha "Usos en el proyecto"
  En la ficha "Usos en el proyecto" se muestran las instancias de versiones de tipo, así como su correspondiente ubicación en el proyecto. Si se selecciona una instancia, en la ventana de inspección pueden mostrarse las referencias cruzadas de la instancia en el proyecto.
- Ficha "Usos en la librería"
  En la ficha "Usos en la librería" se muestran todas las ubicaciones dentro de la librería en las que se utiliza un tipo o una plantilla maestra.

Abrir la administración de librerías

Procedimiento

Para abrir la administración de librerías, proceda del siguiente modo:

1. Abra la vista de librería.
2. Seleccione un tipo o una carpeta cualquiera que contenga tipos.
3. Elija el comando "Administración de librerías" del menú contextual.

Resultado

Se abre la administración de librerías y se muestran los tipos junto con sus versiones.

Filtrar tipos en la administración de librerías

Introducción

La función de filtro de la administración de librerías ofrece la posibilidad de limitar los tipos mostrados. Están disponibles los siguientes filtros:

- Visualización de todos los tipos que tienen una versión en estado "en edición".
- Visualización de todos los tipos que no tienen instancias en el proyecto
- Visualización de todos los tipos que tienen más de una versión
- Visualización de todos los tipos habilitados

Requisitos

Se ha creado un tipo como mínimo.

Filtrar por todos los tipos en estado "en edición"

1. Seleccione la carpeta "Tipos" en la librería del proyecto.
2. Seleccione la entrada "Administración de librerías" en el menú contextual de la carpeta "Tipos".
   Se abre la administración de librerías.
3. Seleccione la entrada "Cambios pendientes" en la lista desplegable "Filtro".
   En el área "Tipos" solo se muestran los tipos que se encuentran en el estado "en edición".

Filtrar por todos los tipos que no tienen instancias en el proyecto

1. Seleccione la carpeta "Tipos" en la librería del proyecto.
2. Seleccione la entrada "Administración de librerías" en el menú contextual de la carpeta "Tipos".
   Se abre la administración de librerías.
3. Seleccione la entrada "Sin instancias en el proyecto" en la lista desplegable "Filtro".
   ¿Se traduce "not used in Project" como "Sin instancias en el proyecto"?
   En el área "Tipos" solo se muestran los tipos que se encuentran en el estado "en edición".

Filtrar por tipos con varias versiones

1. Seleccione la carpeta "Tipos" en la librería del proyecto.
2. Seleccione la entrada "Administración de librerías" en el menú contextual de la carpeta "Tipos".
   Se abre la administración de librerías.

3. Seleccione la entrada "Varias versiones" en la lista desplegable "Filtro".
4. En el área "Tipos" solo se muestran los tipos que tienen más de una versión.

Filtrar por todos los tipos habilitados

1. Seleccione la carpeta "Tipos" en la librería del proyecto.
2. Seleccione la entrada "Administración de librerías" en el menú contextual de la carpeta "Tipos".
   Se abre la administración de librerías.
3. Seleccione la entrada "Tipos habilitados" en la lista desplegable "Filtro".
   En el área "Tipos" solo se muestran los tipos que tienen únicamente versiones habilitadas.

Consulte también

Abrir la administración de librerías (Página 9141)

Crear librería global

Introducción

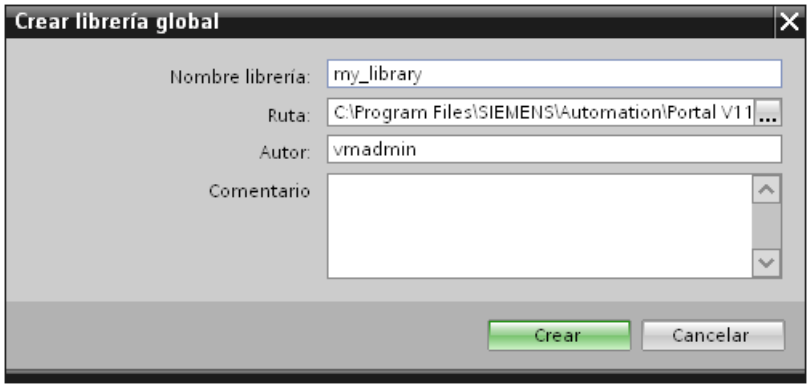
En las librerías se depositan los objetos configurados que se utilizarán varias veces en la configuración. Para utilizar objetos en varios proyectos se crea una librería global.

Requisitos

- Hay un proyecto abierto.
- Está abierta la Task Card "Librerías" o la "Vista de librería".

Procedimiento

1. En "Librerías globales", haga clic en el símbolo .
   Se abre el cuadro de diálogo "Crear librería global nueva".



2. Introduzca un nombre.

3. Seleccione la ruta en la que debe guardarse la nueva librería.
4. Haga clic en "Crear".

Resultado

La librería nueva se mostrará en la paleta "Librerías globales". La librería global contiene las carpetas "Tipos", "Plantillas maestras" y "Datos comunes". Los "Datos comunes" incluyen los informes para la librería global.

En la ubicación de la librería global se crea una carpeta con el nombre de la librería global en el sistema de archivos. El archivo de librería en sí tiene la extensión ".al12".

Guardar una librería global

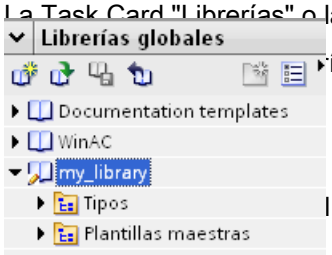
Introducción

Una librería global se guarda como archivo independiente en el disco duro del equipo. El archivo contiene los objetos de la librería global, incluidos los objetos referenciados. Así, p. ej., la referencia de una variable que se ha configurado en un campo E/S también se guarda en la librería.

Cuando cierre su proyecto o WinCC sin guardar, WinCC le solicitará que guarde las librerías globales. Durante la configuración también es posible guardar la librería global sin guardar el proyecto entero.

Requisitos

- Deberá estar abierto un proyecto con una librería como mínimo.
- La Task Card "Librerías" o la vista de librería está abierta.
- ría.



Procedimiento

1. l que desea guardar.
2. En la paleta "Librería global", haga clic en el símbolo .

Otra posibilidad es utilizar el comando "Guardar librerías globales" del menú contextual.

Si desea guardar la librería global en otra carpeta, seleccione "Guardar como" en el menú contextual. Seleccione la ruta en la que desea guardar la librería nueva e introduzca un nombre de archivo.

Resultado

Las librerías globales se guardan con el nombre actual o con el nuevo.

Abrir una librería global

Introducción

Las librerías globales se almacenan en WinCC en archivos independientes. Es posible utilizar una librería global en cada proyecto.

Requisitos

- Hay una librería global guardada.
- Hay un proyecto abierto.
- La Task Card "Librerías" o la vista de librería está abierta.

Procedimiento

1. En la paleta "Librerías globales", haga clic en el símbolo .
   Se abre el cuadro de diálogo "Abrir librería global".
2. Seleccione la ruta en la que está guardada la librería.
3. Haga clic en "Abrir".

Nota

Para poder acceder a una librería global desde varios proyectos, abra la librería global en modo de solo lectura. Si una librería global no se abre en modo de solo lectura, se bloqueará el acceso desde los proyectos.

Resultado

WinCC muestra la librería global abierta en la paleta "Librerías globales".

Mostrar los informes de las librerías globales

Al actualizar las librerías globales se crean informes que especifican todos los cambios realizados en la librería global. Los informes se guardan junto con la librería global y siempre están disponibles una vez que se abre la librería global.

Procedimiento

Para abrir los informes de una librería global, proceda del siguiente modo:

1. Abra la librería global en la Task Card "Librerías" o en la vista de librería.
2. Abra la carpeta subordinada "Datos comunes > Informes".
3. Haga doble clic en el informe deseado.
   El informe se abre en el área de trabajo.

Actualizar un proyecto con el contenido de una librería

Introducción

Después de editar varios tipos en la librería del proyecto, es necesario actualizar todas las instancias del proyecto a la versión más reciente de los tipos.

Requisitos

Está abierta la Task Card "Librerías" o la vista de librería.

Procedimiento

1. Seleccione la librería del proyecto.
2. Elija "Actualizar > Proyecto" en el menú contextual. Se abre un cuadro de diálogo.
3. Seleccione todo el proyecto o algunos dispositivos para la actualización.
4. Para borrar de la librería del proyecto todas las versiones anteriores de los tipos actualizados, active la casilla de verificación "Borrar de la librería del proyecto todas las versiones no utilizadas".
5. Confirme con "Aceptar".

Resultado

En el proyecto, todas las instancias de tipos se actualizarán a la versión más reciente de los tipos seleccionados en la librería del proyecto.

En el árbol del proyecto, bajo "Datos comunes", encontrará un informe del proceso de actualización.

Actualizar una librería con los contenidos de otra librería

Existen las siguientes posibilidades para actualizar librerías:

- Actualización de una librería global con tipos de otra librería global
- Actualización de la librería del proyecto con tipos de una librería global

Los siguientes elementos pueden seleccionarse como origen de la actualización:

- Una librería completa
- Diferentes carpetas dentro de una librería
- Distintos tipos

Requisitos

Para actualizar una librería global, ábrala con derechos de escritura.

Requisitos

Está abierta la Task Card "Librerías" o la vista de librería.

Procedimiento

Para actualizar una librería con contenidos de otra librería, proceda del siguiente modo:

1. Seleccione la librería entera, una carpeta de la librería o tipos concretos.
2. Haga clic en el origen con el botón derecho del ratón y elija el comando "Actualizar > Librería" del menú contextual.
   Se abre el cuadro de diálogo "Actualizar librería".
3. Seleccione qué clase de librería desea actualizar:
   – Elija "Actualizar librería del proyecto" para actualizar la librería del proyecto con tipos de una librería global.
   – Elija "Actualizar librería global" si desea actualizar una librería global.
4. Opcionalmente: Seleccione en la lista desplegable la librería global que desee actualizar.
5. Para borrar de la librería del proyecto todas las versiones anteriores de los tipos actualizados, active la casilla de verificación "Borrar de la librería del proyecto todas las versiones no utilizadas".
6. Confirme con "Aceptar".

Resultado

- Los tipos aún no existentes en la librería de destino se agregarán a ella con todas sus versiones. Los tipos ya existentes en la librería de destino se complementan con las versiones más recientes.
  Si en la librería de destino ya existe una versión actual de un tipo, la versión actual se copia igualmente desde la librería de origen y se le asigna automáticamente un número de versión más reciente.
- Se crea un informe para el proceso de actualización que contiene todas las modificaciones realizadas en la librería de destino.
  Si ha actualizado la librería del proyecto, encontrará el informe en "Datos comunes > Informes" en el árbol del proyecto.
  Si ha actualizado una librería global, encontrará el informe bajo la librería global en la carpeta "Datos comunes > Informes" del árbol del proyecto.

10.9.2.6 Administrar objetos en una librería

Mostrar objetos de librería

Introducción

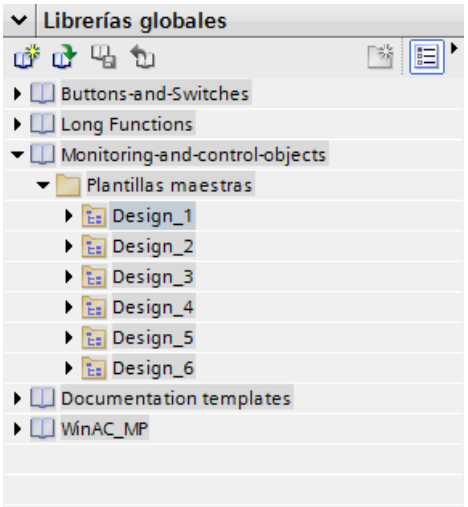
Las librerías se muestran en la paleta correspondiente en forma de carpetas de archivos. Los elementos incluidos en la librería se muestran en la carpeta de archivos y en la paleta "Elementos".

Requisitos

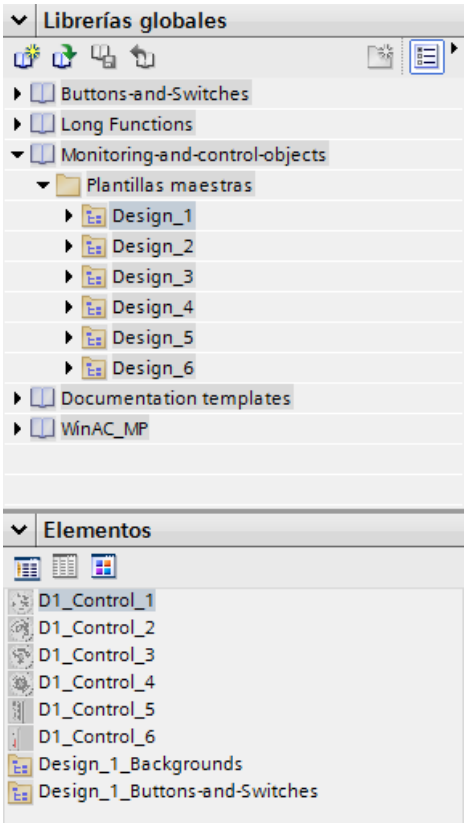
- Se ha creado como mínimo un objeto de librería en una librería.
- La Task Card "Librerías" está abierta.

Procedimiento

1. Seleccione la librería en la paleta correspondiente cuyos objetos desee visualizar.



2. Haga clic en el símbolo .
   Los objetos de librería incluidos se mostrarán en la paleta "Elementos".



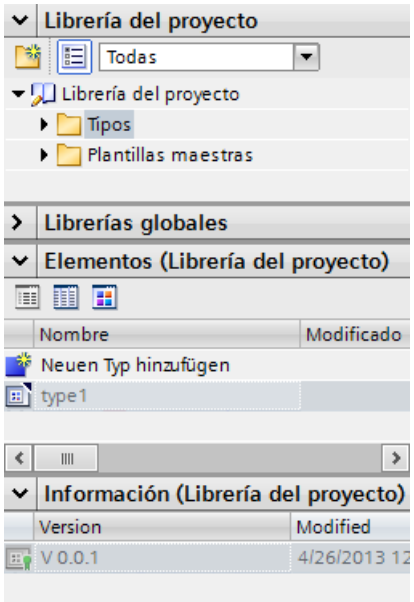
3. Haga clic en uno de los siguientes símbolos:

Símbolo	Descripción
	Vista del elemento en modo detallado
	Vista del elemento en modo de lista
	Vista del elemento en modo sinóptico con símbolos

Si se asignan a la librería varios objetos con una selección múltiple, en la paleta "Elementos" sólo se muestra uno de los objetos. Los diferentes componentes de este elemento se mostrarán en la paleta "Partes".

Mostrar partes de los objetos de una librería

1. Seleccione en la paleta correspondiente la librería de la que desee ver los componentes de un elemento.
2. Haga clic en el símbolo .
3. Los objetos de librería incluidos se mostrarán en la paleta "Elementos".
4. Seleccione el elemento.
   En la paleta "Partes" se verán los objetos que componen el elemento.



Resultado

Los objetos de librería se visualizarán conforme a la configuración efectuada. Se visualizarán los componentes de los faceplates.

Depositar un objeto en una librería

Introducción

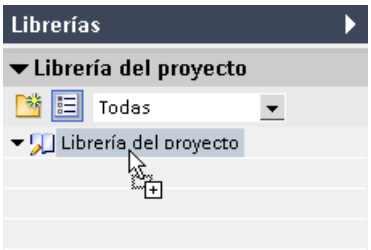
Todos los objetos de WinCC, tales como imágenes, variables, objetos gráficos o avisos, se pueden depositar en librerías. Con la función Drag&Drop se arrastra el objeto en cuestión a la librería desde el área de trabajo, la ventana del proyecto o la vista detallada. Si ha clasificado la librería en categorías, también podrá insertar un objeto directamente en una categoría determinada.

Requisitos

- El editor "Imágenes" está abierto.
- Se ha creado un objeto de imagen en el área de trabajo de la imagen.
- Se muestran las librerías creadas.

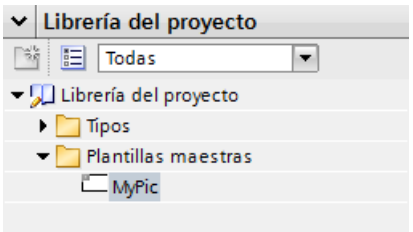
Procedimiento

1. Seleccione el objeto en el área de trabajo del editor "Imágenes".
2. Con la función Drag&Drop, arrastre el objeto desde el área de trabajo hasta la librería deseada.
   El puntero adoptará la forma de una cruz y se visualizará junto con el icono del objeto.



Resultado

El objeto se depositará en la librería y se podrá utilizar varias veces durante la configuración.



Insertar un objeto de librería

Introducción

Al insertar un objeto de librería, el nombre del mismo se compone por defecto de la denominación del tipo de objeto y de un número correlativo.

Si el objeto insertado ya existe, es posible sustituirlo o guardarlo con otro nombre.

Los objetos de librería no asistidos por el panel de operador no pueden insertarse.

Nota

Si se inserta de la librería una imagen con una plantilla enlazada, también se copiará la plantilla. Si ya existe una plantilla adecuada, no se utilizará.

Requisitos

- La Task Card "Librerías" está abierta.
- El editor en el que desea insertar el objeto de librería está abierto.

Procedimiento

1. Seleccione un objeto de la librería.
2. Utilizando la función Drag&Drop, desplace el objeto de librería hasta la posición del área de trabajo en la que desea insertarlo.
   Se inserta el objeto de librería.

Resultado

Cuando el objeto está en la carpeta "Plantillas maestras" hay una copia independiente de dicho objeto de librería insertada en el editor.

Cuando el objeto está en la carpeta "Tipos", hay una instancia del objeto de librería insertada en el editor.

10.9.2.7 Utilizar tipos y sus versiones

Estado de las versiones de tipo

Introducción

Según cuál sea su punto de aplicación, la versión de un tipo puede tener diferentes estados.

Versión habilitada

El estado "versión habilitada" está disponible para todos los tipos sin importar la ubicación.

Para poder editar una versión habilitada es necesario crear primero una nueva versión de prueba o una fase de trabajo.

Las versiones de tipo habilitadas de scripts e imágenes pueden abrirse y visualizarse en su instancia.

Versión "en edición"

El estado "en edición" es válido para los siguientes tipos:

- Faceplates
- Datos del usuario
- Estilos.

Si se crea un tipo nuevo o una versión nueva de un tipo habilitado, el tipo cambiará al estado "en edición".

Los tipos con el estado "en edición" pueden editarse en la vista de librería sin que exista relación con una instancia del proyecto. La compatibilidad del tipo se comprueba mediante una prueba de coherencia al realizar la habilitación.

Versión "en test"

Solo tienen el estado "en test" las versiones de scripts e imágenes HMI.

Si se crea una nueva versión de un tipo, el tipo cambiará al estado "en test".

Una versión "en test" está enlazada con una instancia en el proyecto. Únicamente es posible poner una sola versión de cada tipo en el estado "en test".

Una versión en test solo puede estar enlazada con una sola instancia en el proyecto. Por ello, no es posible copiar una instancia en el portapapeles, duplicarla o crear otro tipo a partir de la instancia mientras esta tenga el estado "en test".

Consulte también

Generar script como tipo (Página 9155)

Generar imagen como tipo (Página 9156)

Crear una nueva versión de un tipo (Página 9157)

Plantillas maestras y tipos (Página 9137)

Generar script como tipo

Requisitos

- Hay un proyecto abierto.
- Se ha creado y abierto un panel de operador.
- El árbol del proyecto está abierto.
- La Task Card "Librerías" está abierta.

Procedimiento

1. Abra en el árbol del proyecto el editor "Scripts".
2. Cree un nuevo script.
3. Seleccione el script en el árbol del proyecto.
4. Arrastre el script mediante arrastrar y soltar a una librería de la Task Card "Librerías". Se abre un cuadro de diálogo.
5. Introduzca un nombre.
6. Introduzca un comentario.

Resultado

Se ha creado una versión de tipo en la librería a partir de un script. El tipo creado se guardará en la librería como versión habilitada. En el proyecto se utilizará una instancia del tipo.

Para modificar el script, cree una nueva versión de este.

Consulte también

Estado de las versiones de tipo (Página 9152)

Generar imagen como tipo

Requisitos

Hay un proyecto abierto.

Se ha creado y abierto un panel de operador.

El árbol del proyecto está abierto.

La Task Card "Librerías" está abierta.

Procedimiento

1. Abra en el árbol del proyecto el editor "Imágenes".
2. Cree una nueva imagen.
3. Seleccione la imagen en el árbol del proyecto.
4. Arrastre la imagen mediante arrastrar y soltar a una librería de la Task Card "Librerías". Se abre un cuadro de diálogo.
5. Introduzca un nombre.
6. Introduzca un comentario.

Resultado

Se ha creado un tipo en la librería a partir de una imagen.

El tipo creado se guardará en la librería como versión habilitada. En el proyecto se utilizará una instancia del tipo.

Para modificar la imagen, cree una nueva versión de esta.

Consulte también

Estado de las versiones de tipo (Página 9152)

Generar estilo como tipo

Introducción

Para definir un nuevo estilo, añada un nuevo tipo a la librería del proyecto.

Adición de un estilo nuevo

1. Abra "Librerías" en la Task Card.
2. Elija el comando "Añadir nuevo tipo" en la opción "Tipos" del menú contextual de la librería del proyecto. Se abre un cuadro de diálogo.
3. Elija "Estilo HMI".

Resultado

El nuevo estilo se creará y aparecerá con el nombre seleccionado en la librería del proyecto. El tipo de estilo HMI recibe el estado "en edición" y la versión 0.0.1.

Crear una nueva versión de un tipo

Principio

Cuando se crea una nueva versión de un tipo, el punto de aplicación del tipo determina el estado que adquirirá la nueva versión creada.

Requisitos

La Task Card "Librerías" está abierta.

Se ha creado y habilitado un tipo.

Procedimiento

1. Seleccione el tipo habilitado.
2. Seleccione "Editar tipo" en el menú contextual.

Resultado para tipos de faceplates, datos de usuario y estilos

Se crea una versión nueva del tipo.

La versión tendrá el estado "en edición". Se abre la vista de librerías.

Resultado para tipos de scripts e imágenes

Se abre un cuadro de diálogo.

Después de seleccionar los ajustes en el cuadro de diálogo, la versión cambia al estado "en test". La instancia utilizada en el proyecto cambia al estado "en test". Se abre la vista de librerías.

Consulte también

Estado de las versiones de tipo (Página 9152)

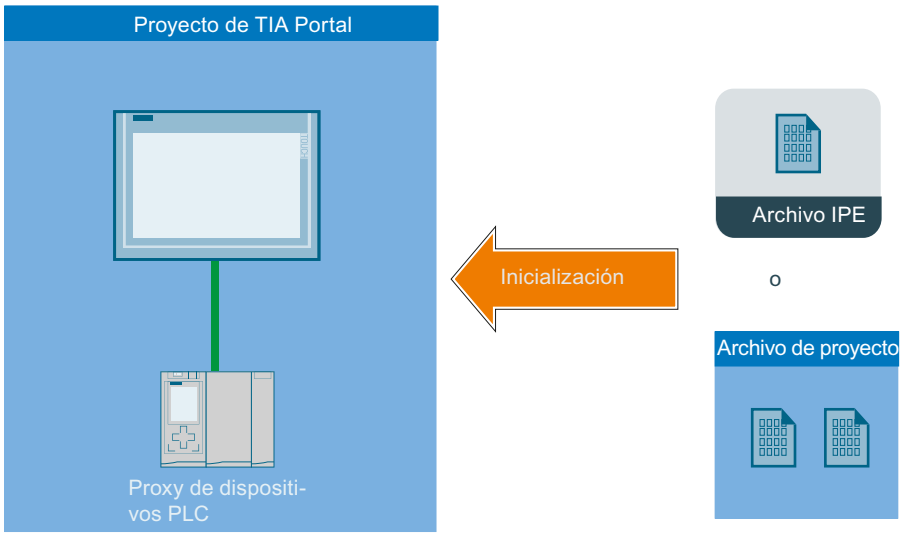
10.9.3 Utilizar datos de control de otros proyectos

10.9.3.1 Principios básicos de la ingeniería interproyecto (IPE)

¿Qué es la IPE?

Con la función IPE (ingeniería interproyecto) puede leer los datos de control de otros proyectos del TIA Portal y utilizarlos para la configuración.

Con IPE puede implementar y agrupar la comunicación de la configuración HMI y la programación PLC en distintos proyectos del TIA Portal.



Puede intercambiar datos de control con IPE de dos maneras:

- Inicialización de proxys de dispositivos mediante archivo IPE (Página 9159)
- Inicialización de proxys de dispositivos mediante archivo de proyecto (Página 9165)

10.9.3.2 Datos de control

Introducción

En el proyecto de origen del controlador se puede elegir qué datos de control se guardarán como datos de proxy de dispositivo.

Para ello se generan datos de proxy de un controlador configurado.

¿Qué datos del proxy del dispositivo pueden intercambiarse?

Mediante IPE se pueden intercambiar los siguientes datos del proxy del dispositivo:

- Bloques de programa
- Objetos tecnológicos
- Variables PLC
- Avisos de PLC

La siguiente información se intercambia automáticamente:

- Interfaces del controlador
- Procesadores de comunicación y módulos de comunicaciones configurados en el controlador

10.9.3.3 Inicializar proxy de dispositivos mediante archivo IPE

Inicialización de proxys de dispositivos mediante archivo IPE

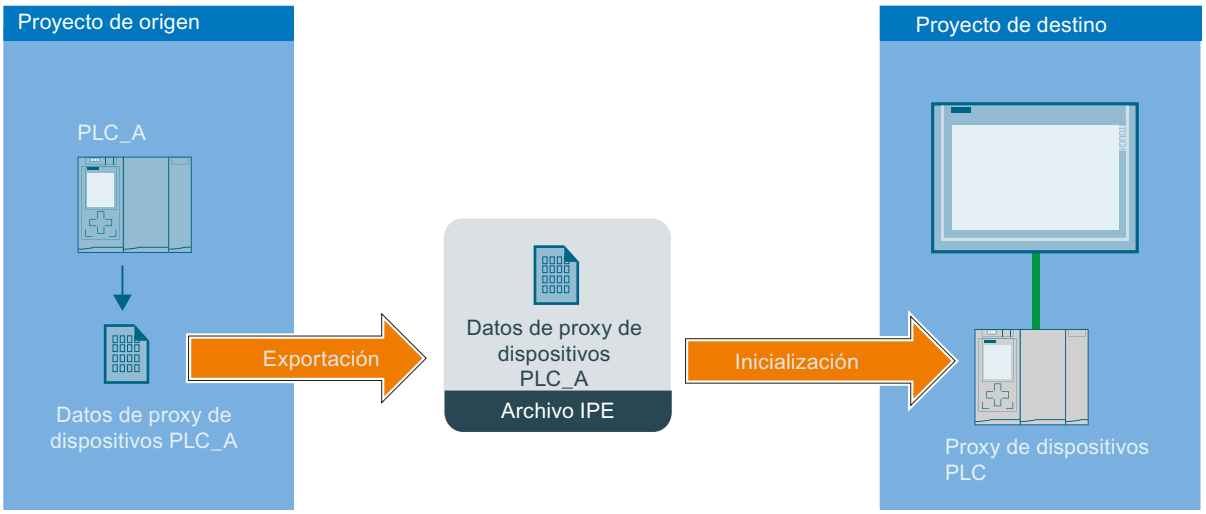
¿Cómo funciona la inicialización de un proxy de dispositivos mediante un archivo IPE?

Se pueden intercambiar datos de control entre dos o más proyectos con ayuda de los archivos IPE.

Los archivos IPE se pueden enviar, p. ej., por correo electrónico, pues el volumen de datos es reducido.

Puede generar para cada controlador varios archivos IPE con diferentes datos de proxy de dispositivos.

En el proyecto de destino, un proxy de dispositivos solo puede llenarse con un registro de proxy de dispositivos.



- Desde el proyecto de origen se exporta un archivo IPE.
  El archivo IPE contiene datos de proxy de dispositivos del controlador "PLC_A".
- En el proyecto de destino se crea un proxy de dispositivos.
- En el proyecto de destino se inicializa entonces el proxy de dispositivos con sus correspondientes datos mediante el archivo IPE.
- Tras la inicialización, todos los datos de proxy de dispositivos procedentes del archivo IPE estarán contenidos en dicho proxy.
- Si se realizan modificaciones en el proyecto de origen, el proxy de dispositivos puede ser actualizado en el proyecto de destino por el PLC_A mediante el nuevo archivo IPE generado.

Inicializar proxy de dispositivos mediante archivo IPE

Introducción

En el TIA Portal se inicializan los proxys de dispositivos en un cuadro de diálogo del proxy de dispositivos.

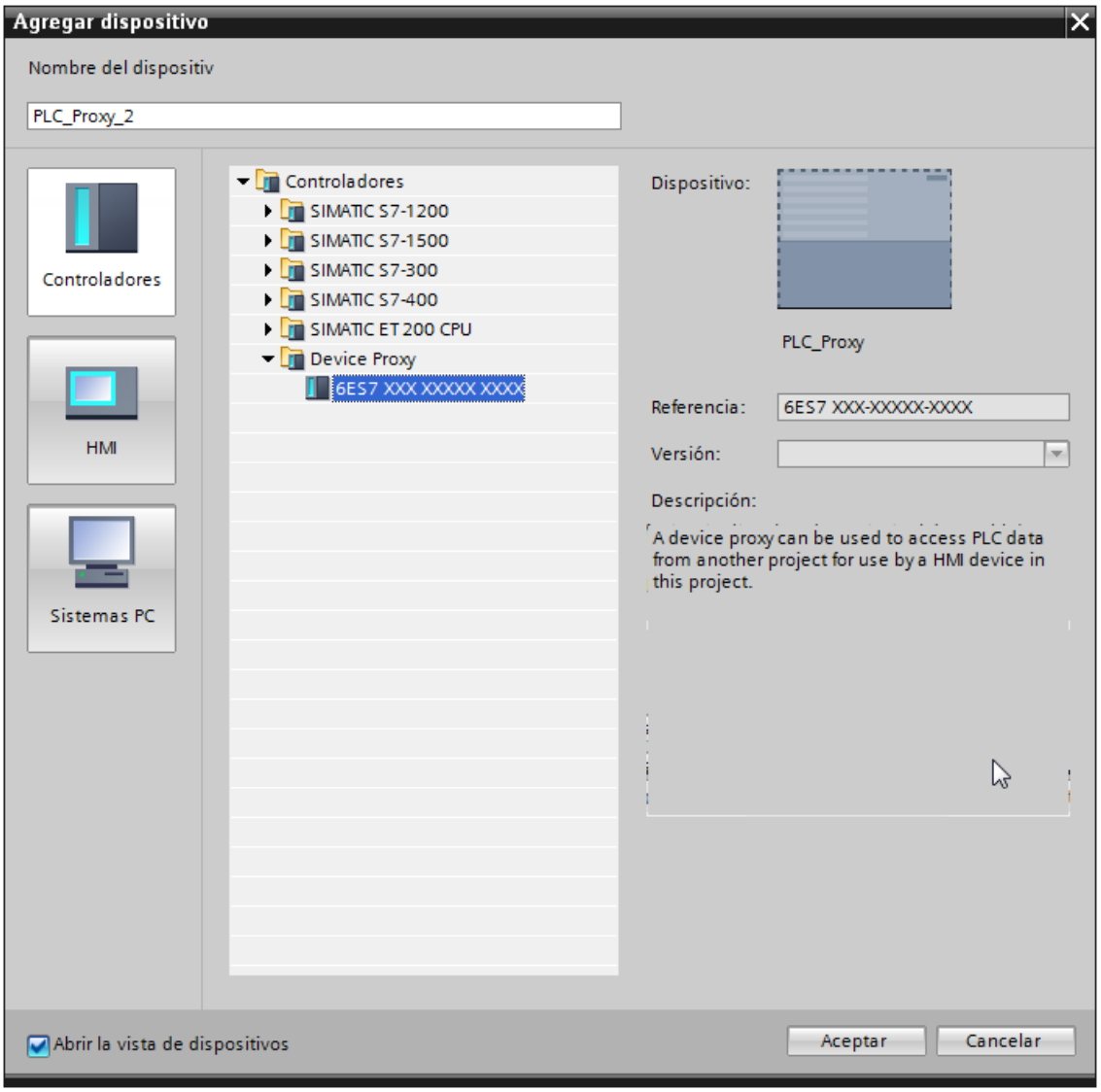
En el archivo IPE están contenidos los datos de proxy de dispositivos procedentes del proyecto de origen.

Requisitos

Debe existir un archivo IPE.

Procedimiento

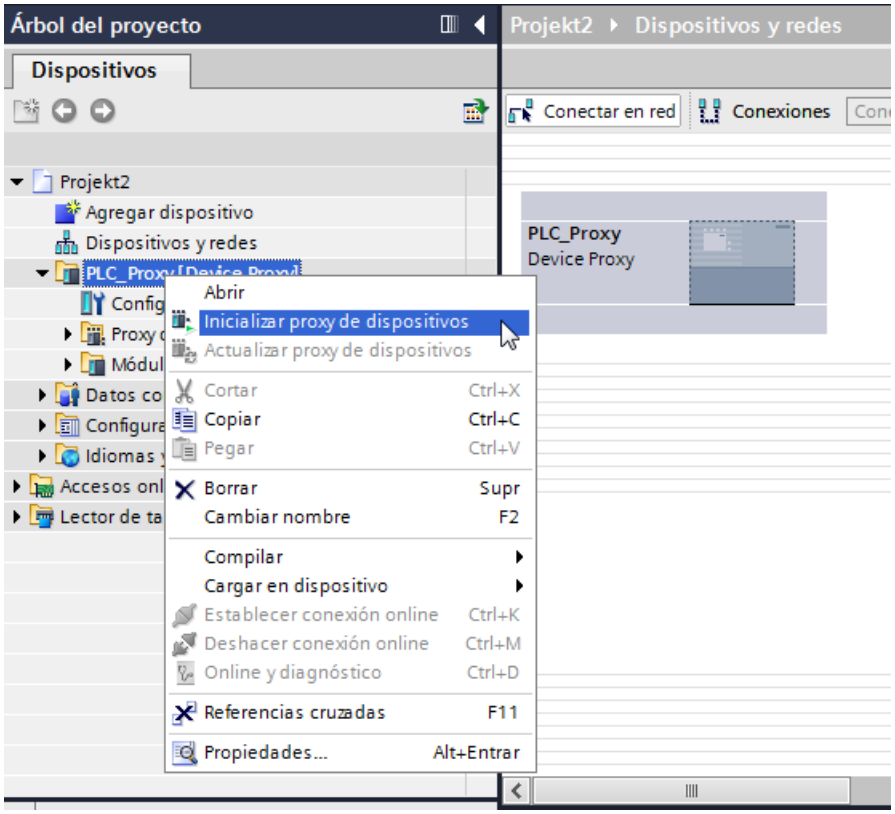
1. Haga doble clic en "Agregar dispositivo" en el árbol del proyecto.
2. Seleccione el proxy de dispositivos en "Controlador".



En el editor "Dispositivos y redes" se crea un nuevo dispositivo.

3. Seleccione el proxy de dispositivos en el árbol del proyecto.

4. Seleccione "Inicializar proxy de dispositivos" en el menú contextual.



5. Seleccione el archivo IPE y haga clic en "Abrir".
   Se abre el cuadro de diálogo "Inicializar proxy de dispositivos".

Nota

En el proyecto de destino no es posible elegir los datos de proxy de dispositivos antes de la inicialización.

Se aplican todos los datos de proxy de dispositivos que están presentes en el archivo IPE.

Si desea seleccionar datos de proxy de dispositivos antes de una inicialización, inicialice un archivo de proyecto: Inicializar proxy de dispositivos mediante archivo de proyecto (Página 9165)

6. Haga clic en "Aceptar".
   Comenzará la inicialización del proxy de dispositivos.

Resultado

Tras la correcta inicialización, los datos de proxy de dispositivos procedentes del archivo IPE estarán guardados en dicho proxy.

Ahora puede configurar una conexión HMI con el proxy de dispositivos y, p. ej., conectar variables PLC del proxy de dispositivos con variables HMI.

Actualizar proxy de dispositivos mediante archivo IPE

Introducción

Si en el proyecto de origen del proxy de dispositivos se han modificado los datos de dicho proxy, estos datos se pueden actualizar en el proyecto de destino.

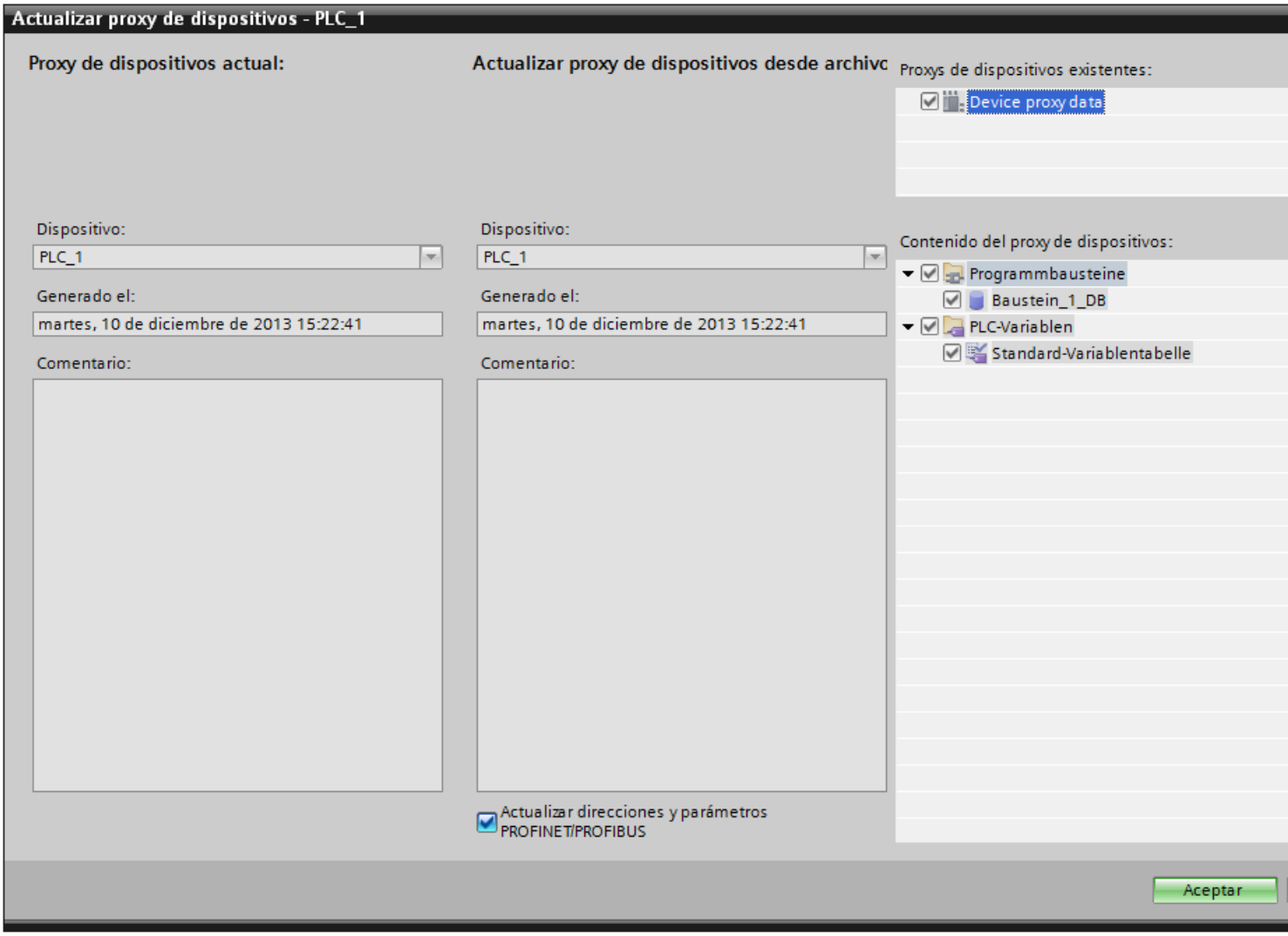
Requisitos

- Nuevo archivo IPE con los datos de proxy de dispositivos procedentes del proyecto de origen.
- En el proyecto de destino ya se ha inicializado un proxy de dispositivos con un archivo IPE.

Procedimiento

1. Haga clic en el proxy de dispositivos en el árbol del proyecto.
2. Seleccione "Actualizar proxy de dispositivos" en el menú contextual.

3. Seleccione el archivo IPE.



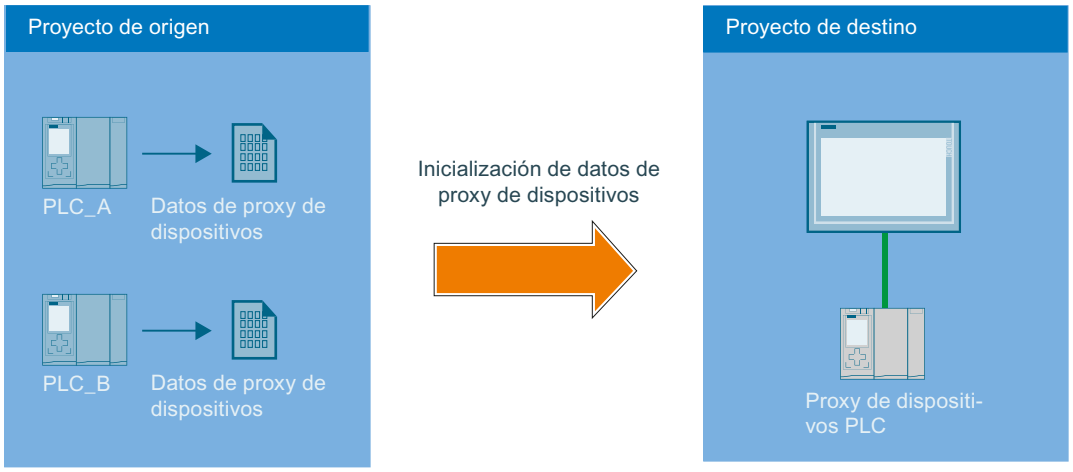
4. Seleccione si desea actualizar las direcciones PROFINET o PROFIBUS.
5. Haga clic en "Aceptar".

10.9.3.4 Inicializar proxy de dispositivos mediante archivo de proyecto

Inicialización de proxys de dispositivos mediante archivo de proyecto

¿Cómo funciona la inicialización mediante un archivo de proyecto?

Puede aplicar los datos de control a su proyecto del TIA Portal mediante un archivo de proyecto:



- En el proyecto de origen existen, p. ej., dos controladores.
- En el proyecto de origen, los dos controladores crean objetos de datos de proxy de dispositivos.
- El proyecto de origen se guarda.
- En el proyecto de destino se crea un proxy de dispositivos.
- Con la entrada para inicializar el proxy de dispositivos en el menú contextual de dispositivo se selecciona primero el proyecto de origen (*.ap13).
- Para la selección están disponibles los controladores "PLC_A" y "PLC_B" y sus datos de proxy de dispositivos ya creados.
- Tras la selección de datos de proxy de dispositivos de "PLC_A", los datos se guardan en el PLC_Proxy del proyecto de destino.
- Las modificaciones del proyecto de origen pueden aplicarse con una actualización mediante el archivo de proyecto.

Inicializar proxy de dispositivos mediante archivo de proyecto

Introducción

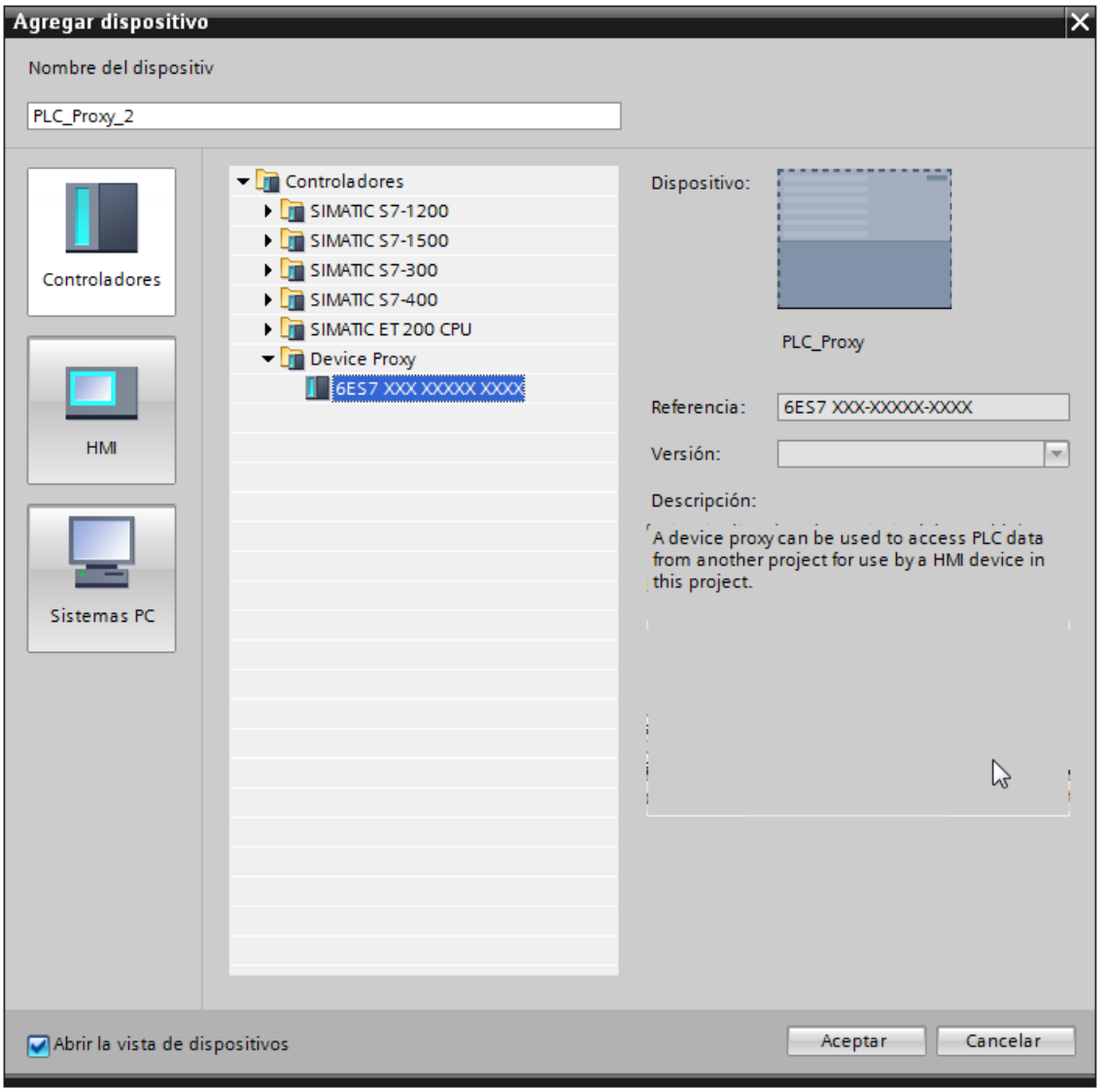
Los proxys de dispositivos se inicializan con un archivo de proyecto.

Requisitos

Existe el archivo de proyecto (*.ap13).

Procedimiento

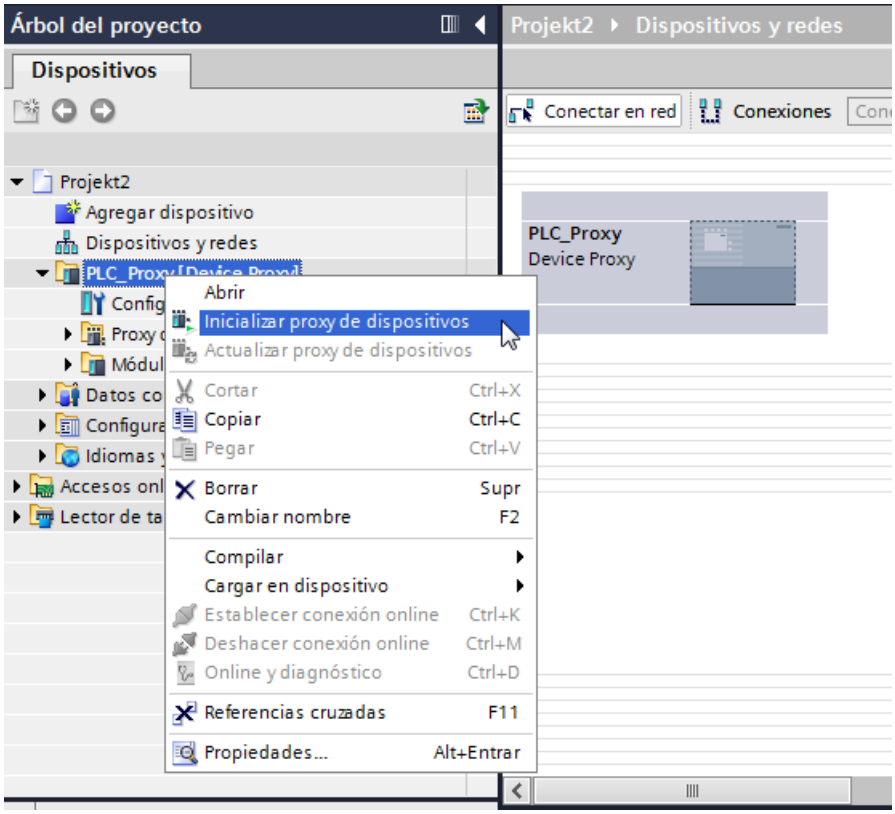
1. Haga doble clic en "Agregar dispositivo" en el árbol del proyecto.
2. Seleccione el proxy del dispositivo en "Controlador".



En el editor "Dispositivos y redes" se crea un nuevo dispositivo.

3. Seleccione el proxy del dispositivo en el árbol del proyecto.

4. Seleccione "Inicializar proxy del dispositivo" en el menú contextual.



5. Seleccione en el cuadro de diálogo "Abrir fuente de datos de proxy de dispositivos" la siguiente entrada:
   "Proyectos de TIA Portal (*.ap13)"
6. Seleccione un archivo de proyecto y haga clic en "Abrir".
   Se abre el cuadro de diálogo "Inicializar proxy del dispositivo".
7. Seleccione en un PLC disponible un objeto Datos de proxy del dispositivo ya creado para inicializar el proxy.
8. Haga clic en "Aceptar".

Resultado

Tras la inicialización, los datos de control del archivo de proyecto que contiene el objeto Datos del proxy del dispositivos seleccionado se guardan en el proxy.

Ahora podrá configurar una conexión HMI con el proxy del dispositivo y, p. ej., conectar variables PLC del proxy del dispositivo con variables HMI.

Inicialización mediante proyectos Classic

Si importa datos de control de proyectos previos al TIA Portal V11, encontrará más información en la FAQ correspondiente.

ID del artículo: 73502293 (http://support.automation.siemens.com/WW/view/de/73502293)

Actualizar proxy de dispositivos mediante archivo de proyecto

Introducción

Si en el proyecto de origen del proxy de dispositivos se han modificado los datos de control, puede actualizar el proxy de dispositivos en su proyecto del TIA Portal.

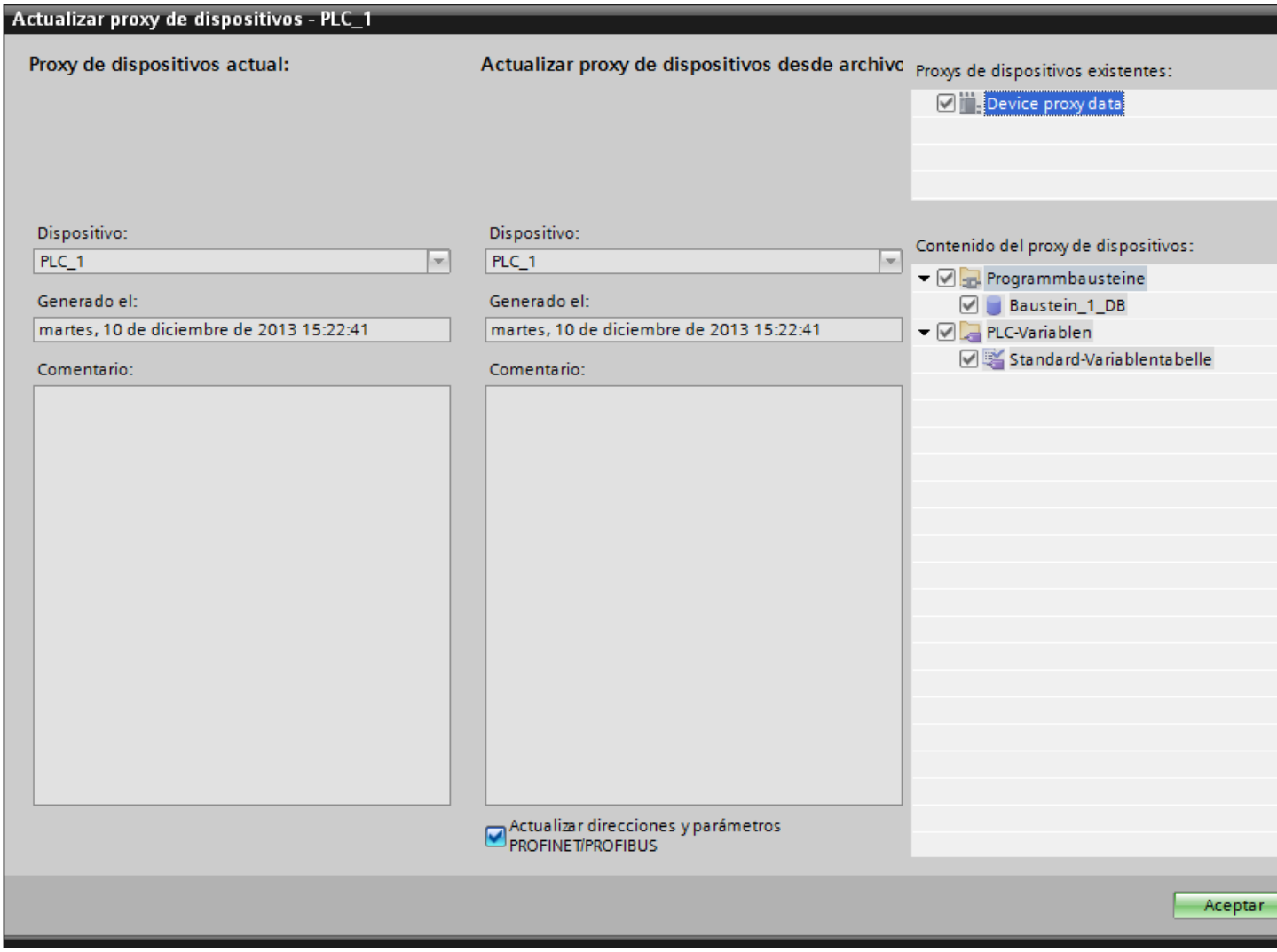
Requisitos

- El archivo de proyecto se ha generado a partir del proyecto de origen del proxy de dispositivos.
- En el proyecto de destino existe un proxy de dispositivos ya inicializado.

Procedimiento

1. Haga clic en el proxy de dispositivos en el árbol del proyecto.
2. Seleccione "Actualizar proxy de dispositivos" en el menú contextual.

3. Seleccione el archivo de proyecto.
   Se abre el cuadro de diálogo "Actualizar proxy de dispositivos".



4. Seleccione un dispositivo.

Nota

Si ha creado en el proyecto de origen varios objetos de datos de proxy de dispositivos, debe seleccionar uno de ellos.

Nota

Los proxys de dispositivos ya inicializados no se pueden sobrescribir con datos de control de otro controlador.

5. Seleccione si desea actualizar las direcciones PROFINET o PROFIBUS.
6. Haga clic en "Aceptar".

10.9.4 Importar y exportar datos de proyecto

10.9.4.1 Importación y exportación de datos de proyecto

Introducción

WinCC ofrece la posibilidad de intercambiar datos de proyecto entre distintos proyectos o copiarlos a aplicaciones externas.

Exportación e importación entre proyectos

Datos de proyecto que pueden exportarse desde un proyecto e importarse a otro:

- Registros de receta
- Avisos
- Variables
- Listas de textos
- Textos del proyecto

La posibilidad de exportar e importar estos datos reduce el esfuerzo necesario. En vez de crear registros nuevos, utilice los datos que ya ha creado en proyectos anteriores.

Edición del archivo de exportación

Dependiendo del editor se encuentran disponibles los siguientes formatos de archivo para la exportación o importación:

- *.xlsx para avisos, variables, textos de proyecto y listas de textos
- *.csv para registros de recetas

El archivo de importación se puede editar, p. ej., en Excel.

Formato de archivo XLSX

El formato XLSX es un formato de archivo para hojas Excel que se basa en el formato Open-XML. Los archivos XLSX están optimizados para Microsoft Excel 2007.

El archivo XLSX permite organizar las columnas como se desee.

Formato de archivo CSV

CSV es el acrónimo inglés de "comma separated value" (valor separado por comas). En este formato, las columnas de la tabla que contienen el nombre y el valor de la entrada aparecen separadas por un punto y coma. Cada fila de la tabla se finaliza con un salto de línea. Los archivos CSV también pueden abrirse en Excel para poder editarlos.

Importación de datos de proyecto

Al importar los datos de proyecto se crean los objetos en el mismo.

Al importar archivos se comprueba la sintaxis del archivo de importación. No se comprueba la corrección de los valores importados ni la dependencia entre los valores importados.

Si los datos importados presentan fallos, éstos se notifican mediante un aviso cuando se compila el proyecto.

Copiar en formato Excel

Todos los editores de tablas permiten copiar los contenidos en formato Excel en la memoria caché del PC. A continuación pueden pegarse los datos de proyecto en formato Excel directamente en cualquier aplicación fuera del TIA Portal. Para ello debe usarse el correspondiente comando del menú contextual del área de trabajo:

- Al seleccionar el comando en el menú contextual del encabezado, se copia la fila entera en la memoria caché.
- Al seleccionar el comando en el menú contextual de una celda, se copia solo el contenido de la celda en la memoria caché.
- Si se seleccionan varias celdas y se activa el comando, se copian siempre todos los datos seleccionados en la memoria caché.

Este intercambio de datos solo es posible como exportación.

Consulte también

Exportar avisos (Página 9175)

Exportar variables (Página 9182)

Exportar listas de textos (Página 9188)

10.9.4.2 Importar y exportar recetas

Exportar recetas

Introducción

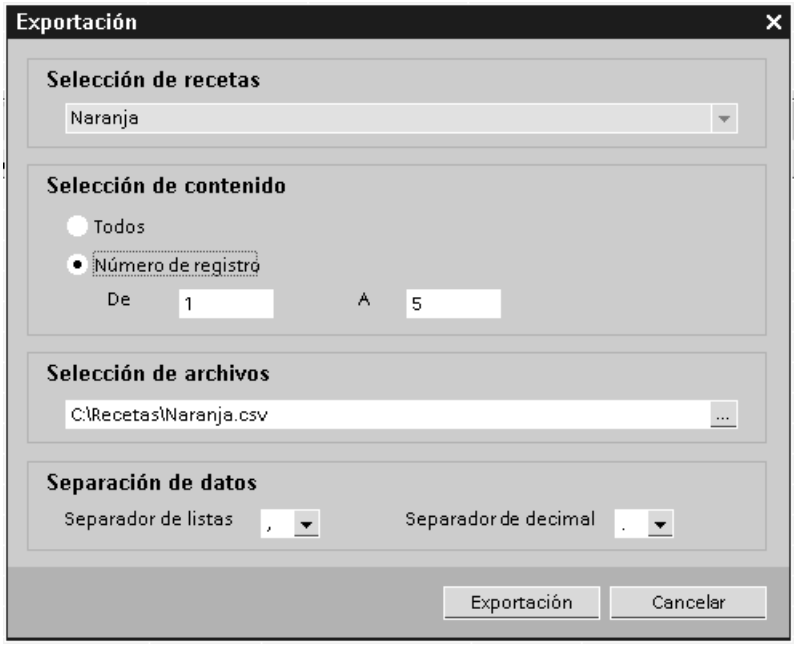
WinCC ofrece una función para la exportación de registros de recetas.

Requisitos

- El proyecto de WinCC está abierto para realizar la exportación.
- En un proyecto se han creado recetas.
- El editor "Recetas" está abierto.

Exportar recetas

1. En el editor "Recetas" seleccione la receta cuyos registros desee exportar.
2. Haga clic en el símbolo .
   Se abre el cuadro de diálogo "Exportación".



   En "Selección de recetas" se mostrará la receta seleccionada.
3. En "Selección de contenido" determine si deben exportarse todos los registros o sólo los seleccionados.
4. En "Selección de archivos" determine en qué archivo deben guardarse los datos de receta.
5. En "Separación de datos" defina el separador de listas y de decimales.
6. Haga clic en "Exportación".
   Se iniciará la exportación.

Resultado

Los datos exportados se han escrito en un archivo CSV. El archivo CSV se guarda en el directorio indicado.

Consulte también

Exportar avisos (Página 9175)

Importación y exportación de datos de proyecto (Página 9168)

Exportar variables (Página 9182)

Exportar listas de textos (Página 9188)

Importar recetas

Introducción

Las recetas se identifican mediante su nombre. Así pues, estos nombres deben ser unívocos. Para comprobar la estructura de los datos del archivo de importación, ábralo en un editor de textos sencillo.

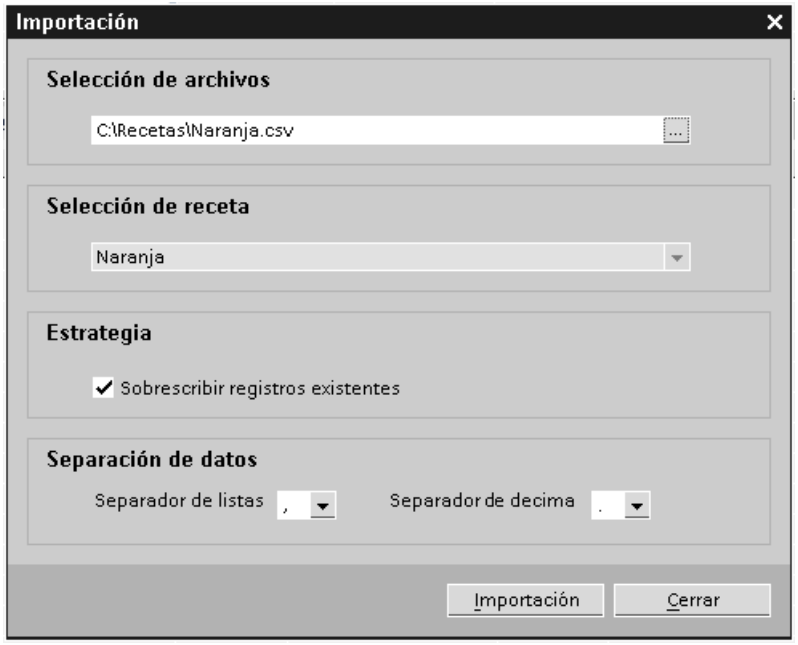
Al realizar la importación, seleccione si deben sobrescribirse los registros homónimos existentes.

Requisitos

- Se ha creado un archivo CSV con una receta como mínimo.
- El proyecto de WinCC está abierto para realizar la importación.
- El editor "Recetas" está abierto con una receta como mínimo.

Importar una receta

1. En el editor "Recetas" seleccione la receta a la que deben importarse los registros.
2. Haga clic en el símbolo .
   Se abre el cuadro de diálogo "Importación".



   En "Selección de receta" se mostrará la receta seleccionada.
3. En "Selección de archivos" seleccione el archivo que desea importar.
4. En "Estrategia" determine si deben sobrescribirse los registros homónimos existentes.

5. En "Separación de datos" seleccione el separador de listas y de decimales que se utilizó en el archivo CSV.
6. Haga clic en "Importación".
   Se iniciará la importación.

Resultado

Los registros se crearán en la receta seleccionada. En función del ajuste de la "Estrategia", los registros homónimos existentes del archivo CSV se sobrescribirán.

Si se desactiva la opción "Sobrescribir registros existentes", también se importan los registros homónimos del archivo CSV.

Formato de los datos de recetas

Introducción

En este capítulo se describe la estructura requerida para el archivo de importación de recetas. El archivo que contiene los datos de las recetas debe existir en el formato "*.csv". :

Estructura de los datos de receta

La estructura del archivo de importación está predeterminada y es obligatoria. El ejemplo siguiente muestra la estructura de una receta que contiene dos elementos con dos registros cada uno:

```
List separator=<separador de listas>Decimal symbol=<separador de
decimales><separador de listas><salto de línea>
<nombre de la receta><separador de listas><separador de
listas><salto de línea>
LANGID_<ID del idioma><separador de listas>
<nombre de visualización del elemento de receta 1><separador de
listas>
<nombre de visualización del elemento de receta 2><salto de línea>
<número de la receta><separador de listas>
<número de registro de receta 1><separador de listas>
<número de registro de receta 2><salto de línea>
<variable del elemento de receta 1><separador de listas>
<registro de receta 1 valor 1><separador de listas>
<registro de receta 2 valor 1><salto de línea>
<variable del elemento de receta 2><separador de listas>
<registro de receta 1 valor 2><separador de listas>
<registro de receta 2 valor 2><salto de línea>
```

ID del idioma

Utilice las "IDs de idiomas de Windows" en formato decimal, p. ej. "1033" para inglés. Encontrará más información en la documentación del sistema operativo Windows.

Consulte también

Exportar recetas (Página 9169)

10.9.4.3 Importar y exportar avisos

Exportar avisos

Introducción

WinCC incorpora una función de exportación para avisos.

Requisitos

- El proyecto de WinCC está abierto para realizar la exportación.
- Se han creado avisos en el proyecto.
- El editor "Avisos HMI" está abierto.

Exportar avisos

1. En "Avisos de bit" o "Avisos analógicos", haga clic en el botón .
   Se abre el cuadro de diálogo "Exportación".

2. Haga clic en el botón "..." e indique en qué archivo se guardarán los datos.
3. Indique si desea exportar "Avisos de bit" o "Avisos analógicos".
4. Haga clic en "Exportación". Se iniciará la exportación.

Resultado

Los datos exportados se han escrito en un archivo xlsx. El archivo xlsx se guarda en la carpeta indicada.

Consulte también

Importación de avisos (Página 9176)

Formato de los datos de avisos analógicos (Página 9177)

Formato de los datos de avisos de bit (Página 9180)

Exportar recetas (Página 9169)

Exportar variables (Página 9182)

Importación y exportación de datos de proyecto (Página 9168)

Importación de avisos

Introducción

Los avisos se identifican mediante su número. Los números de aviso deben ser unívocos en los tipos de aviso "Aviso analógico" y "Aviso de bit". Por este motivo, los avisos con números de aviso idénticos se sobrescriben. Se crea un aviso nuevo con un número de aviso no existente.

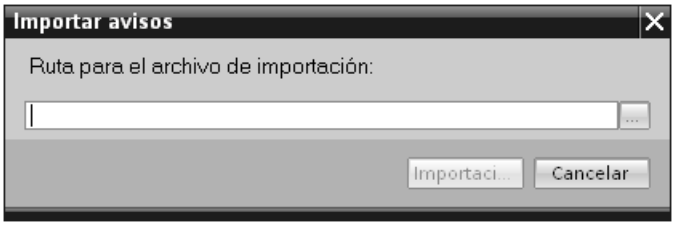
Si un archivo xlsx contiene entradas de lista vacías de avisos ya existentes, se ignorarán durante la importación. Las entradas del aviso existente se conservan y no se sustituyen por entradas vacías.

Requisitos

- Se ha creado un archivo xlsx con avisos.
- El archivo xlsx tiene la estructura exigida.
- El proyecto de WinCC está abierto para realizar la importación.
- El editor "Avisos HMI" está abierto.

Importar avisos

1. En "Avisos de bit" o "Avisos analógicos", haga clic en el botón . Se abre el cuadro de diálogo "Importación".



2. Haga clic en el botón "..." y seleccione el archivo que desee importar.
3. Haga clic en el botón "Importar". Se iniciará la importación. El progreso de la importación se indica en una barra.

Resultado

Se han creado en WinCC los avisos correspondientes a los datos importados, incluidos sus textos. En la ventana de resultados se muestran avisos relativos a la importación. En el directorio de origen de los archivos de importación se deposita un archivo de informe. El nombre del archivo de registro es idéntico al del archivo de importación y tiene la extensión "*.xml".

Cuando se importan los datos, es preciso comprobar si hay enlaces con objetos, por ejemplo, parámetros dinámicos como variables.

- Cuando hay un objeto del mismo nombre, se utiliza el objeto existente.
- Si no existe un objeto homónimo, debe crearse un objeto con el nombre correspondiente, o bien crearse un nuevo enlace.

Nota

Al importar archivos xlsx se comprueba la sintaxis del archivo de importación. No se comprueban sin embargo el significado de las propiedades ni las dependencias entre las propiedades. Es posible asignar a un aviso una variable de disparo del tipo equivocado p. ej. String. Al compilar se notifica un error.

Consulte también

Exportar avisos (Página 9173)

Formato de los datos de avisos analógicos

Introducción

En este capítulo se describe la estructura requerida para el archivo de importación de avisos analógicos. El archivo que contiene los datos de avisos analógicos debe existir en el formato "*.xlsx".

Estructura de los datos de aviso

En Microsoft Excel, el archivo de importación consta de hojas de tablas:

- Analog alarms (avisos analógicos)
- Limits (límites)

Cada aviso obtiene una fila propia en el archivo de importación. El archivo de importación que contiene los avisos analógicos debe tener la estructura siguiente:

Ejemplo de hoja de tabla "Analog alarms"

	A	B	C	D	E
1	ID	Name	Event text [en-US], Alarm text	FieldInfo [Alarm text]	Class
2	1	Analog_alarm_1	AA1 Error-AC with maximum text length: abcdefghijklmnopqrstuvwxyz		Errors
3	2	Analog_alarm_2	AA2 Warning-AC this text should be bold		Warnings
4	3	Analog_alarm_3	AA3 SDm-AC this text should be italic		SDm
5	4	Analog_alarm_4	AA4 SDo-AC <u>this text should be underlined</u>		SDo
6	5	Analog_alarm_5	AA5 SystemAcknowledgement-AC <blink>this text should be flashing<		System_Acknowle
7	25	Analog_alarm_25	Internal AA23 switchDT: Deadband mode in case of violation - value HL		AADT-Internal
8	26	Analog_alarm_26	Internal AA23 switchDT: Deadband mode in case of violation - percent		AADT-Internal
9	31	Analog_alarm_31	Internal AA23 switchDT: Low limit violation static		AADT-Internal
10	32	Analog_alarm_32	Internal AA23 switchDT: High limit violation static		AADT-Internal
11	33	Analog_alarm_33	Internal AA23 switchDT: Low limit violation dynamic		AADT-Internal
12	34	Analog_alarm_34	Internal AA23 switchDT: High limit violation dynamic		AADT-Internal
13	35	Analog_alarm_35	Internal AA23 switchDT: delay 3 seconds High limit violation		AADT-Internal
14	42	Analog_alarm_40	Internal AA23 switchDT: delay 3 seconds Low limit violation LLV		AADT-Internal
15	23	Analog_alarm_41	AA23 DT in event text: <field ref="0" /> Bool, <field ref="1" /> Byte, <fi	<ref id = 0; type = AlarmTag; T	ACAT
16	24	Analog_alarm_42	AA24 DT in event text: <field ref="0" /> Timer, <field ref="1" /> Counte	<ref id = 0; type = AlarmTag; T	ACAT

Tabla 10-13 Significado de las entradas

Entrada de lista	Significado
ID	El número sirve para referenciar un aviso. El número de aviso es unívoco. Por este motivo, los avisos con números de aviso idénticos se sobrescriben durante la importación. Se crea un aviso nuevo con un número de aviso no existente.
Name	Nombre del aviso analógico
Event text [de-DE], Alarm text	Muestra el texto del aviso. El nombre del campo contiene un código de idioma. Para la importación es necesario que los textos de aviso vayan provistos de un código de idioma. Si el texto del aviso incluye un parámetro dinámico, al texto se le añade una expresión con una ID de referencia. Ejemplo: text <field ref="0" />. La ID permite asignar el parámetro dinámico a un texto de aviso.
FeldInfo	Especifica si el texto del aviso contiene parámetros dinámicos. Los ajustes están separados por un punto y coma (";"). Ejemplo de parámetros dinámicos: Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;> Lista de textos: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;>
Class	La pertenencia a una categoría (clase de aviso) determina si el aviso se debe acusar o no. Además, sirve para controlar la visualización del aviso en el panel de operador. La clase de aviso (categoría) determina asimismo si y dónde debe archivarse el aviso correspondiente.
Group	Especifica la pertenencia a un grupo de avisos. Si un aviso pertenece a un grupo, sólo se podrá acusar explícitamente junto con los demás avisos de ese mismo grupo.
Trigger tag	Especifica la variable que se supervisa para indicar un rebase de límite.

Entrada de lista	Significado
Delay time value	Especifica el tiempo de retardo. El aviso se activa tan solo cuando se infringe el límite durante el tiempo de retardo indicado.
Delay time unit	Especifica la unidad de tiempo del retardo.
Report	Activa la generación de un informe del respectivo aviso en una impresora. True o "1" = informe activado False o "0" = informe desactivado Además, la generación de informes debe activarse globalmente en el proyecto.
Info text [de-DE], Info text	El tooltip es una propiedad opcional de un aviso y puede contener información adicional acerca de dicho aviso. Este tooltip se visualiza en una ventana independiente del panel de operador cuando el operador pulsa la tecla de ayuda (<HELP>). El nombre del campo contiene un código de idioma.

Ejemplo de hoja de tabla "Limits"

	A	B	C	D	E	F	G
1	Alarm ID	Limit type	Limit value	Limit mode	Deadband mo	Deadband valu	Deadband in percent
2	1	Constant	0	Upper limit	Off	0	False
3	2	Constant	1	Upper limit	Off	0	False
4	3	Constant	2	Upper limit	Off	0	False
5	4	Constant	3	Upper limit	Off	0	False
6	5	Constant	4	Upper limit	Off	0	False
7	25	Constant	50	Upper limit	On both	5	False
8	26	Constant	50	Upper limit	On both	10	True
9	31	Constant	50	Lower limit	Off	0	False
10	32	Constant	50	Upper limit	Off	0	False
11	33	Tag	AASDTdyn	Lower limit	Off	0	False
12	34	Tag	AASDT1dyn	Upper limit	Off	0	False
13	35	Constant	50	Upper limit	Off	0	False
14	36	Constant	50	Lower limit	On both	5	False

Tabla 10-14 Significado de las entradas

Entrada de lista	Significado
Alarm ID	Número de aviso El número sirve para referenciar un aviso. El número de aviso es unívoco. Por este motivo, los avisos con números de aviso idénticos se sobrescriben durante la importación. Se crea un aviso nuevo con un número de aviso no existente.
Limit mode	Disparador Especifica el método con el que se supervisa el límite.
Limit type	Especifica el límite que se supervisa. Como valor límite puede determinarse tanto una variable como una constante.
Limit value	Valor límite Especifica la variable o constante que se supervisa para indicar un rebase de límite.

Entrada de lista	Significado
Deadband mode	Modo de histéresis Especifica si se aplica una histéresis y en qué casos. Al "Desaparecer" Al "Aparecer" Al "Aparecer" y "Desaparecer"
Deadband in percent	0 = el valor indicado para "Histéresis" se considera un valor absoluto. 1 = El valor indicado para "Histéresis" está referido en porcentajes del límite.
Deadband mode	Histéresis Especifica una constante como valor de la histéresis.

Nota

"No Value" en la tabla

Las entradas de la tabla que tienen el valor "No Value" borran los valores correspondientes de un aviso existente del mismo nombre.

Consulte también

Exportar avisos (Página 9173)

Formato de los datos de avisos de bit

Introducción

En este capítulo se describe la estructura requerida para el archivo de importación de avisos de bit. El archivo que contiene los avisos de bit debe existir en el formato "*.xlsx".

Estructura de los datos de aviso

El archivo de importación consta en Microsoft Excel de la hoja "Discrete alarms" (avisos de bit). Cada aviso obtiene una fila propia en el archivo de importación. El archivo de importación que contiene los avisos de bit debe tener la estructura siguiente:

Ejemplo de hoja de tabla "Discrete alarms"

	A	B	C	D	E	
1	ID	Name	Event text [en-US], Alarm text	FieldInfo [Alarm text]	Class	Trig
2	1	Discrete_alarm_1	DA1 Error-AC with maximum text length: abcdefghijklmnopqrstuvwxyz äöü\na		Errors	HM
3	2	Discrete_alarm_2	DA2 Warning-AC this text should be bold		Warnings	HM
4	3	Discrete_alarm_3	DA3 SDm-AC this text should be italic		SDm	HM
5	4	Discrete_alarm_4	DA4 SDo-AC <u>this text should be underlined</u>		SDo	HM
6	5	Discrete_alarm_5	DA5 SystemAcknowledgement-AC <blink>this text should be flashing</blink		System_Ackn	HM
7	6	Discrete_alarm_6	DA6 SystemNoAcknowledgement-AC mixed test: Bold, Italic, <		System_No_A	HM
8	7	Discrete_alarm_7	DA7 DT in event text: <field ref="0" /> Integer, <field ref="1" /> Real, <field re	<ref id = 0; type = AlarmTag; Tag = PL	ACAT	HM
9	8	Discrete_alarm_8	DA8 DT in event text: <field ref="0" /> S5Time, <field ref="1" /> Timer, <field r	<ref id = 0; type = AlarmTag; Tag = PL	ACAT	HM
10	11	Discrete_alarm_9	DA11 DT in event text: <field ref="0" /> Int, <field ref="1" /> Real, <field ref="2	<ref id = 0; type = AlarmTag; Tag = In	ACAT	HM
11	12	Discrete_alarm_10	DA12 DT in event text: <field ref="0" /> UDInt, <field ref="1" /> UInt;	<ref id = 0; type = AlarmTag; Tag = U	ACAT	HM
12	13	Discrete_alarm_11	DA13 Textformat: Integer: <field ref="0" /> decimal, <field ref="1" /> binary, <	<ref id = 0; type = AlarmTag; Tag = H	ACAT	HM

Tabla 10-15 Significado de las entradas

Entrada de lista	Significado
ID	El número sirve para referenciar un aviso. El número de aviso es unívoco. Por este motivo, los avisos con números de aviso idénticos se sobrescriben durante la importación. Se crea un aviso nuevo con un número de aviso no existente.
Name	Nombre del aviso analógico
Event text [de-DE], Alarm text	Muestra el texto del aviso. El nombre del campo contiene un código de idioma. Para la importación es necesario que los textos de aviso vayan provistos de un código de idioma. Si el texto del aviso incluye un parámetro dinámico, al texto se le añade una expresión con una ID de referencia. Ejemplo: text <field ref="0" />. La ID permite asignar el parámetro dinámico a un texto de aviso.
FeldInfo	Especifica si el texto del aviso contiene parámetros dinámicos. Los ajustes están separados por un punto y coma (";"). Ejemplo de parámetros dinámicos: Variable: <ref id = 0; type = AlarmTag; Tag = Tag1; DisplayType = Decimal; Length = 5;> Lista de textos: <ref id = 1; type = CommonTextList; TextList = Textlist1; Tag = tag 2; Length = 5;>
Class	La pertenencia a una categoría (clase de aviso) determina si el aviso se debe acusar o no. Además, sirve para controlar la visualización del aviso en el panel de operador. La clase de aviso (categoría) determina asimismo si y dónde debe archivarse el aviso correspondiente.
Group	Especifica la pertenencia a un grupo de avisos. Si un aviso pertenece a un grupo, sólo se podrá acusar explícitamente junto con los demás avisos de ese mismo grupo.
Trigger tag	Especifica la variable que contiene el bit con el que se activa el aviso.
Trigger bit	Especifica el número del bit con el que se activa el aviso.
Acknowledge tag	Especifica la variable que contiene el bit que el usuario activa al acusar. Sólo está disponible si la clase de aviso (categoría) seleccionada exige que se acuse el aviso.
Acknowledgment bit	Especifica el número del bit que se activa cuando el usuario acusa el aviso.

Entrada de lista	Significado
PLC acknowledgement tag	Especifica la variable que contiene el bit con el que el programa de control acusa el aviso. Sólo está disponible si la clase de aviso (categoría) seleccionada exige que se acuse el aviso.
PLC acknowledgment bit	Especifica el número del bit con el que el programa de control acusa el aviso.
Delay time value	Especifica el tiempo de retardo. El aviso se activa tan solo cuando se infringe el límite durante el tiempo de retardo indicado.
Delay time unit	Especifica la unidad de tiempo del retardo.
Report	Activa la generación de un informe del respectivo aviso en una impresora. True o "1" = informe activado False o "0" = informe desactivado Además, la generación de informes debe activarse globalmente en el proyecto.
Info text [de-DE], Info text	El tooltip es una propiedad opcional de un aviso y puede contener información adicional acerca de dicho aviso. Este tooltip se visualiza en una ventana independiente del panel de operador cuando el operador pulsa la tecla de ayuda (<HELP>). El nombre del campo contiene un código de idioma.

Nota

"No Value" en la tabla

Las entradas de la tabla que tienen el valor "No Value" borran los valores correspondientes de un aviso existente del mismo nombre.

Consulte también

Exportar avisos (Página 9173)

10.9.4.4 Importar y exportar variables

Exportar variables

Introducción

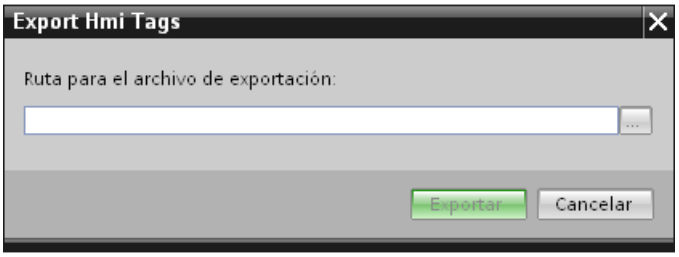
WinCC incorpora una función de exportación para variables.

Requisitos

- El proyecto de WinCC está abierto para realizar la exportación.
- Se han creado variables en el proyecto.
- El editor "Variables HMI" está abierto.

Exportar variables

1. En la ficha "Variables HMI", haga clic en el botón .
   Se abre el cuadro de diálogo "Exportación".



2. Haga clic en el botón "..." e indique en qué archivo se guardarán los datos.
3. Haga clic en "Exportación". Se iniciará la exportación.

Resultado

Los datos exportados se han escrito en un archivo xlsx. El archivo xlsx se guarda en la carpeta indicada.

Consulte también

Importar variables (Página 9183)

Exportar avisos (Página 9173)

Importación y exportación de datos de proyecto (Página 9168)

Exportar listas de textos (Página 9188)

Exportar recetas (Página 9169)

Formato de los datos de variables (Página 9185)

Importar variables

Introducción

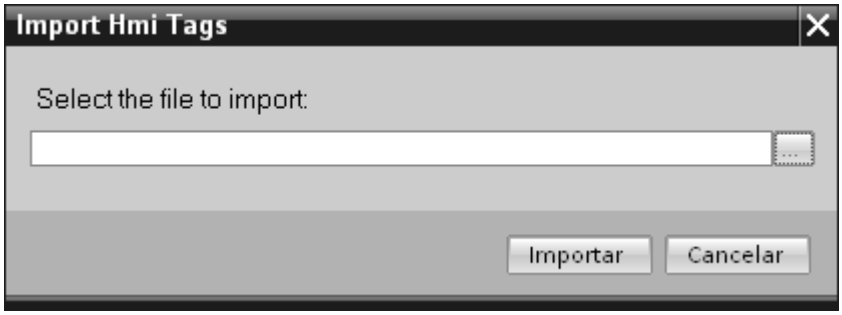
Las variables se identifican mediante el nombre de las variables. Cuando el nombre de la variable ya existe en el proyecto, la variable existente se sobrescribe con los datos del archivo xlsx. Si la variable no existe todavía, se crea una variable nueva.

Requisitos

- Se ha creado un archivo xlsx con variables.
- El archivo xlsx tiene la estructura exigida.
- El proyecto de WinCC está abierto para realizar la importación.

Importar variables

1. Haga clic en "Variables HMI" en el árbol del proyecto.
2. Haga doble clic en "Mostrar todas las variables". Se abrirá el editor "Variables HMI".
3. Haga clic en el botón . Se abre el cuadro de diálogo "Importación".



4. Haga clic en el botón "..." y seleccione el archivo que desee importar.
5. Haga clic en el botón "Importar". Se iniciará la importación.

Resultado

Las variables correspondientes se han creado en WinCC. En la ventana de resultados se muestran avisos relativos a la importación. En el directorio de origen de los archivos de importación se deposita un archivo de informe. El nombre del archivo de registro es idéntico al del archivo de importación y tiene la extensión "*.xml".

Cuando se importan los datos, es preciso comprobar si hay enlaces con objetos, por ejemplo, parámetros dinámicos como variables.

- Cuando hay un objeto del mismo nombre, se utiliza el objeto existente.
- Si no existe un objeto homónimo, debe crearse un objeto con el nombre correspondiente, o bien crearse un nuevo enlace.

Nota

Al importar archivos xlsx se comprueba la sintaxis del archivo de importación. No se comprueban sin embargo el significado de las propiedades ni las dependencias entre las propiedades. Es posible asignar a una variable una variable de disparo del tipo equivocado, p. ej. String. Al compilar se notifica un error.

Consulte también

Exportar variables (Página 9180)

Formato de los datos de variables

Introducción

En este capítulo se describe la estructura que debe tener el archivo de los datos de las variables para la importación de éstas. El archivo de los datos de las variables debe existir en el formato "*.xlsx".

Estructura de los datos de las variables

En Microsoft Excel, el archivo de importación consta de hojas de tablas:

- HMI Tags (variables HMI)
- Multiplexing (variables multiplex)

Cada variable aparece en una fila propia en el archivo de importación. El archivo de importación que contiene los datos de las variables deberá tener la estructura siguiente:

Ejemplo de hoja de tabla "HMI Tags"

	A	B	C	D	E	F	G	H
1	Name	Path	Connection	PLC tag	DataType	Length	Address	Access
2	HMI_Int	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
3	Mux_Tag_1	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
4	Mux_Tag_2	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
5	Mux_11	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
6	Mux_21	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
7	Mux_13	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
8	Mux_12	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
9	Mux_23	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
10	Mux_22	Internal Tags	<No Value>	<No Value>	Int	2	<No Value>	<No Va
11	Mux_Tag_1_Index	Internal Tags	<No Value>	<No Value>	UInt	2	<No Value>	<No Va
12	Mux_Tag_12	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Va
13	Mux_Tag_11	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Va
14	Mux_Tag_13	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Va
15	HMI_UDInt	Internal Tags	<No Value>	<No Value>	UDInt	4	<No Value>	<No Va
16	Gauge_Process	Default tag table	<No Value>	<No Value>	Int	2	<No Value>	<No Va
17	Button_Tag_4	Default tag table	<No Value>	<No Value>	Int	2	<No Value>	<No Va
18	HMI_USInt	Internal Tags	<No Value>	<No Value>	USInt	1	<No Value>	<No Va
19	Data_block_2_PLC_DateTime_2	Default tag table	HMI_connection_1	Data_block_2.PLC_I	Date_And_Time	8	%DB28.DBX598.(	<absol

Tabla 10-16 Significado de las entradas

Entrada de lista	Significado
Name	Indica el nombre configurado de una variable HMI.
Path	Indica en qué carpeta de la navegación de proyecto se almacena la variable. La estructura de carpeta se representa mediante "\": "nombre de carpeta 1\nombre de carpeta 2\nombre de variable".

Entrada de lista	Significado
PLC Tag	Indica si la variable está conectada con una variable PLC.
Connection	Indica el nombre de la conexión con el PLC.
Data type	Indica el tipo de datos de la variable. Los tipos de datos permitidos dependen del driver de comunicación utilizado. Para más información sobre los tipos de datos permitidos para los distintos drivers de comunicación, consulte el capítulo "Comunicación" de la documentación.
Length	Indica la longitud de la variable. La entrada sólo es útil para tipos de datos con longitud dinámica, como los de tipo String. En otros tipos de datos la entrada se queda vacía.
Address	Indica la dirección de la variable en el PLC. La dirección de la variable debe escribirse exactamente igual a como se representa en WinCC; por ejemplo "%DB1.DBW0". Esta dirección permanece vacía en el caso de las variables internas.
Multiplexing	Indica si se aplica el multiplexado.
Index tag	Muestra el nombre de la variable indexada para el multiplexado. En runtime se lee primero el valor de la variable indexada. A continuación se accede a la variable que está en la posición correspondiente de la lista de variables.
StartValue	Indica el valor inicial de la variable.
ID tag	Mediante la ID de actualización se actualiza el valor de una variable con ayuda de una función o de una tarea PLC. La ID de actualización debe ser unívoca en el panel de operador.
Coding	Muestra el método de codificación.
DiplayName [de_DE]	Muestra el nombre de visualización de una variable HMI. El nombre del campo contiene un código de idioma. El nombre del campo contiene un código de idioma. Para la importación es necesario que los nombres de visualización estén provistos de un código de idioma. Al realizar la importación el texto se importa en el idioma de proyecto correspondiente.
Acquisition mode	Indica el modo de adquisición de la variable.
Acquisition cycle	Indica el ciclo de adquisición de la variable. El ciclo de adquisición debe escribirse exactamente igual a como se representa en WinCC. Este valor no depende del idioma, por lo que debe indicarse de forma idéntica en todos los idiomas. El valor estándar es "1 s". Si el tipo de adquisición de la variable es "Bajo demanda", el ciclo de adquisición es indefinido. Los ciclos de adquisición definidos por el usuario deben crearse previamente, ya que de lo contrario no se importará el archivo.
High High Limit type	Indica si el límite "High high" se va a vigilar con una constante, con una variable o si no se va a vigilar en absoluto.
High High Limit	Muestra el límite "High High".
High Limit type	Indica si el límite "High" se va a vigilar con una constante, con una variable o si no se va a vigilar en absoluto.
High Limit	Muestra el límite "High".
Low Limit type	Indica si el límite "Low" se va a vigilar con una constante, con una variable o si no se va a vigilar en absoluto.
Low Limit	Muestra el límite "Low".
Low Low Limit type	Indica si el límite "Low Low" se va a vigilar con una constante, con una variable o si no se va a vigilar en absoluto.
Low Low Limit	Muestra el límite "Low Low".

Entrada de lista	Significado
Linear scaling	Indica si está activada la escala lineal. Esta entrada sólo puede utilizarse para variables externas.
End value PLC	Indica el valor final de la variable PLC.
Start value PLC	Indica el valor inicial de la variable PLC.
End value HMI	Indica el valor final de la variable HMI.
Start value HMI	Indica el valor inicial de la variable HMI.

Ejemplo de hoja de tabla "Multiplexing"

	A	B	C
1	HMI Tag name	Multiplex Tag	Index
2	Mux_Tag_1	Mux_11	0
3	Mux_Tag_1	Mux_12	1
4	Mux_Tag_1	Mux_13	2
5	Mux_Tag_2	Mux_21	0
6	Mux_Tag_2	Mux_22	1
7	Mux_Tag_2	Mux_23	2
8	Mux_Tag_12	HMI_Array_Mux2	-1
9	Mux_Tag_11	HMI_Array_Mux1	-1
10	Mux_Tag_13	HMI_Array_Mux3	-1

Tabla 10-17 Significado de las entradas

Entrada de lista	Significado
Name	Indica el nombre configurado de una variable HMI que utiliza un direccionamiento indirecto. La variable HMI debe estar presente en la hoja "HMI Tags".
Index	Muestra el valor del que depende la elección de la variable.
Multiplex Tag	Muestra la variable perteneciente al valor de índice de la lista de variables.

Nota

"No Value" en la tabla

Las entradas de la tabla que tienen el valor "No Value" borran los valores correspondientes de una variable existente del mismo nombre.

Consulte también

Exportar variables (Página 9180)

10.9.4.5 Importar y exportar listas de textos

Exportar listas de textos

Introducción

WinCC incorpora una función de exportación para listas de textos.

Requisitos

- El proyecto de WinCC está abierto para realizar la exportación.
- Se han creado listas de textos en el proyecto.
- El editor "Listas de textos y gráficos" está abierto.

Exportar listas de textos

1. En la ficha "Listas de textos", haga clic en el botón .
   Se abre el cuadro de diálogo "Exportación".

2. Haga clic en el botón "..." e indique en qué archivo se guardarán los datos.
3. De forma estándar, los textos se exportan en todos los idiomas del proyecto definidos.
   Si no desea exportar algún idioma, desactívelo en el cuadro de diálogo.
4. Haga clic en "Exportación". Se iniciará la exportación.

Resultado

Los datos exportados se han escrito en un archivo xlsx. El archivo xlsx se guarda en la carpeta indicada.

Consulte también

Importar listas de textos (Página 9189)

Importación y exportación de datos de proyecto (Página 9168)

Exportar variables (Página 9180)

Exportar recetas (Página 9169)

Formato de los datos para las listas de textos (Página 9190)

Importar listas de textos

Introducción

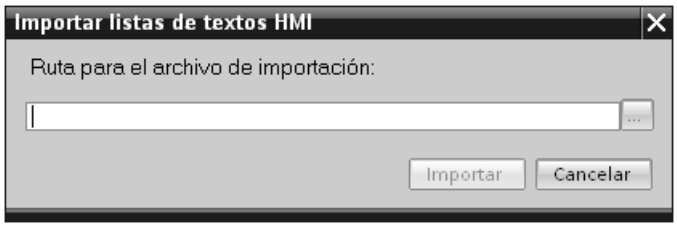
A continuación, importe las listas de textos desde un archivo xlsx a WinCC.

Requisitos

- Se ha creado un archivo xlsx con listas de textos.
- El archivo xlsx tiene la estructura exigida.
- El proyecto de WinCC está abierto para realizar la importación.
- El editor "Listas de textos y gráficos" está abierto.

Importar listas de textos

1. En la ficha "Listas de textos", haga clic en el botón .
   Se abre el cuadro de diálogo "Importación".



2. En "Selección de archivos" seleccione el archivo que desea importar.
3. Haga clic en el botón "Importar". Se iniciará la importación.

Resultado

Se han importado las listas de textos. Se han creado en WinCC las correspondientes listas de textos. En la ventana de resultados se muestran avisos relativos a la importación. En el directorio de origen de los archivos de importación se deposita un archivo de informe. El nombre del archivo de registro es idéntico al del archivo de importación y tiene la extensión "*.xml".

Cuando se importan los datos, es preciso comprobar si hay enlaces con objetos, por ejemplo, parámetros dinámicos como variables.

- Cuando hay un objeto del mismo nombre, se utiliza el objeto existente.
- Si no existe un objeto homónimo, debe crearse un objeto con el nombre correspondiente, o bien crearse un nuevo enlace.

Consulte también

Exportar listas de textos (Página 9186)

Formato de los datos para las listas de textos

Introducción

En este capítulo se describe la estructura que debe tener el archivo de las listas de textos para la importación. El archivo de los datos de las listas de textos debe existir en el formato "*.xlsx".

Estructura de los datos de las variables

El archivo de importación consta en Microsoft Excel de dos hojas:

- TextList (listas de textos)
- TextListEntry (entrada de listas de textos)

Cada lista de textos obtiene una fila propia en el archivo de importación. El archivo de importación que contiene los datos debe tener la estructura siguiente:

Ejemplo de hoja de tabla "TextList"

	A	B	C
1	Name	ListRange	Comment
2	TLValue/Range	Decimal	
3	TLBit	Bit	
4	TLBitnumber	Binary	
5	TLall_1	Decimal	
6	TLall_2	Decimal	

Tabla 10-18 Significado de las entradas

Entrada de lista	Significado
Name	Muestra el nombre de la lista de textos.
ListRange	Indica el área de la lista de textos: número = número de bit (0-31) rango = valor/área bit = bit (0;1)
Comment	Comentario cualquiera acerca de la lista de textos. Es posible introducir 500 caracteres como máximo.

Ejemplo de hoja de tabla "TextListEntry"

	A	B	C	D	E	F
1	Name	Parent	DefaultEntry	Value	Text	FieldInfos
2	Text_list_entry_1	TLValue/Range	TRUE	0 - 1	Default entry TLValue/Range ->	
3	Text_list_entry_2	TLValue/Range		2 - 3	TLValue/Range = 2-3 ->	
4	Text_list_entry_3	TLValue/Range		1	TLValue Range - single value = 1 ->	
5	Text_list_entry_1	TLBit		0	TLBit = 0 ->	
6	Text_list_entry_2	TLBit		1	TLBit = 1 ->	
7	Text_list_entry_1	TLBitnumber	TRUE	0	Default entry TLBitnumber; ->	
8	Text_list_entry_2	TLBitnumber		0	TLBitnumber - Bitnumber 0 is set ->	
9	Text_list_entry_3	TLBitnumber		1	TLBitnumber - Bitnumber 1 is set ->	
10	Text_list_entry_4	TLBitnumber		2	TLBitnumber - Bitnumber 2 is set ->	
11	Text_list_entry_5	TLBitnumber		3	TLBitnumber - Bitnumber 3 is set ->	
12	Text_list_entry_1	TL1	TRUE	0 - 1	Default entry TL1	
13	Text_list_entry_2	TL1		1 - 3	TL1 Value between 1 - 3 ->	
14	Text_list_entry_3	TL1		4 - 6	TL1 Value between 4 - 6 ->	
15	Text_list_entry_4	TL1		7	TL1 Single value = 7 ->	
16	Text_list_entry_1	TL2	TRUE	0 - 1	<field ref="0" /> ->	<ref id = 0; type = CommonTextList; TextList = TL1; Tag = HMI_TL1control; Lengt
17	Text_list_entry_2	TL2		1	TL2 Single value = 1 ->	
18	Text_list_entry_3	TL2		2 - 3	TL2 Range between 2 - 3 ->	
19	Text_list_entry_1	TLMultilined	TRUE	0 - 1	Default entry TLMultilined; last row	
20	Text_list_entry_2	TLMultilined		0 - 3	TLMultilined Value between 0-3\nwith test of"\n"	
21	Text_list_entry_1	TLall_1	TRUE	0 - 1	Default entry TLall_1	
22	Text_list_entry_1	TLall_2	TRUE	0 - 1	Default entry TLall_2	

Tabla 10-19 Significado de las entradas

Entrada de lista	Significado
Name	Muestra el nombre de la entrada de la lista de textos.
Parent	Muestra el nombre de la lista de textos correspondiente.
DefaultEntry	Indica si la entrada de la lista de textos es una entrada estándar. La entrada estándar aparecerá siempre que la variable adopte un valor no definido.
Value	Indica los valores o rangos de valores de números enteros de una variable asignados a la entrada de texto de la lista de textos.

Entrada de lista	Significado
Text	Muestra la entrada de la lista de textos. El nombre del campo contiene un código de idioma. Para la importación es necesario que las entradas de la lista de textos estén provistas de un código de idioma. Si la entrada de la lista de textos incluye un parámetro dinámico, al texto se le añade una expresión con una ID de referencia. Ejemplo: text <field ref="0" />. A través de la ID se asigna una entrada de la lista de textos al parámetro dinámico.
FeldInfo	Indica si la lista de textos contiene parámetros dinámicos. Los ajustes están separados por puntos y coma (";") Ejemplo de parámetros dinámicos: Variable: <ref id = 0; type = CommonTagDisplayFormat; Tag = tag 1; DisplayType = Decimal; DisplayFormat = 9;> Lista de textos: <ref id = 1; type = CommonTextList; TextList = Textliste_1; Tag = tag 2; Length = 5;> Variable de controlador: <ref id = 0; type = CommonControlTagDisplayFormat; DisplayType = Decimal; DisplayFormat = 9;>

Consulte también

Exportar listas de textos (Página 9186)

10.9.4.6 Exportar e importar textos del proyecto

Exportar textos del proyecto

Se exportan textos del proyecto para traducirlos. Los datos se exportan a un archivo Office Open XML con la extensión ".xlsx". Este archivo puede editarse p. ej. con Microsoft Excel.

El archivo se puede intercambiar con los traductores y, tras ser traducido, se puede volver a importar directamente en el proyecto.

Requisitos

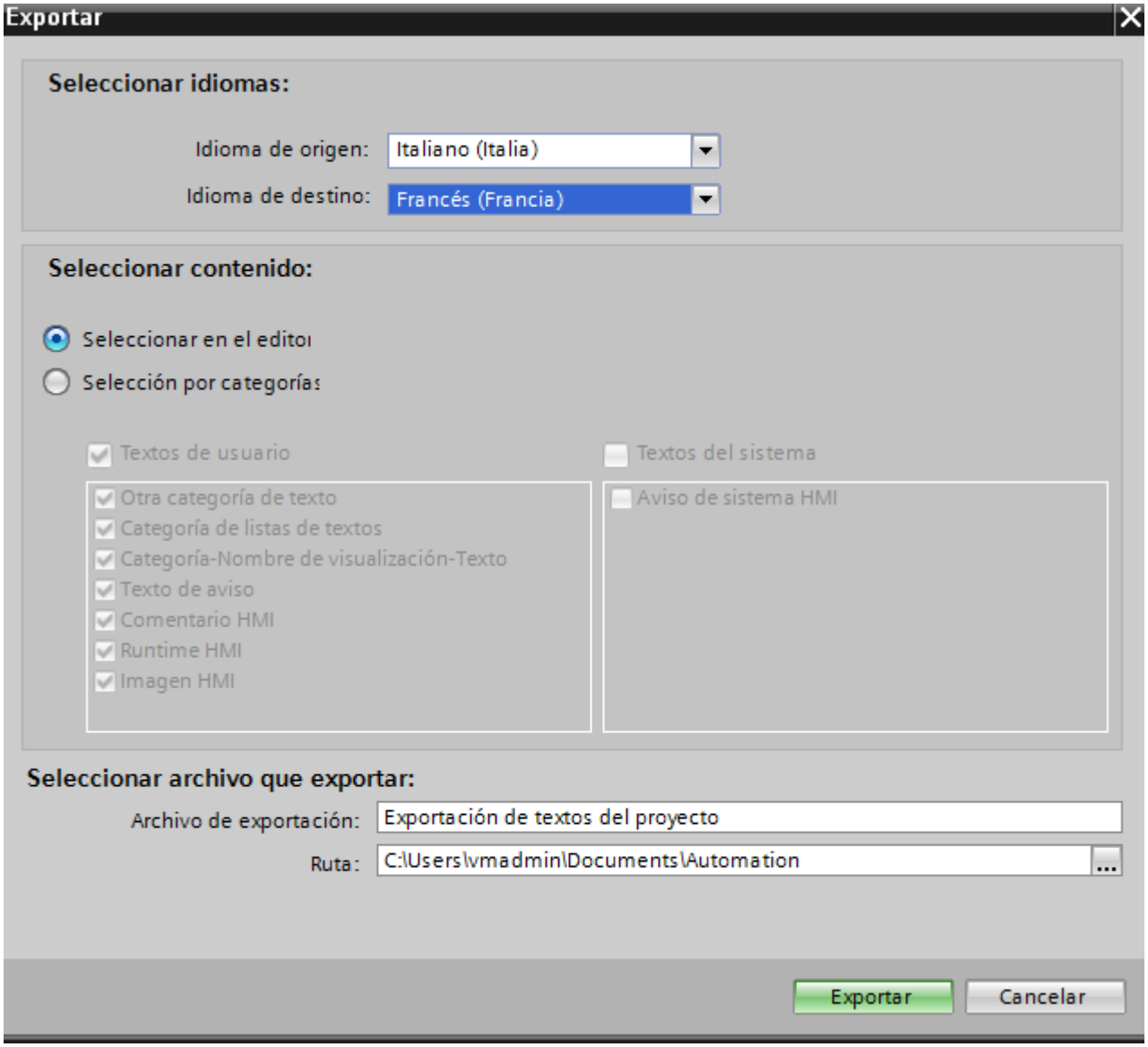
- En el editor "Idiomas del proyecto" están activados por lo menos dos idiomas, p. ej. el italiano y el francés.

Exportar textos del proyecto

Para exportar textos del proyecto individuales, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto". Se abre el editor "Textos del proyecto".
3. Seleccione los textos que desee exportar.

4. Haga clic en el botón . Se abrirá el cuadro de diálogo "Exportación".



5. Elija en la lista desplegable "Idioma de origen" el idioma a partir del cual desea efectuar la traducción, p. ej. el italiano.
6. Elija en la lista desplegable "Idioma de destino" el idioma al que desea traducir el texto, p. ej. el francés.
7. Indique en el campo de entrada "Archivo de exportación" una ruta y un nombre para el archivo de exportación.
8. Haga clic en "Exportación".

Resultado

Los textos seleccionados en el editor "Textos del proyecto" se han escrito en un archivo xlsx. El archivo xlsx se guarda en la carpeta indicada.

También es posible seleccionar y exportar todos los textos del proyecto por categorías. Dependiendo de los textos que desee exportar, active en el cuadro de diálogo "Exportación" las opciones "Textos de usuario" o "Textos del sistema". En este caso la exportación se puede limitar adicionalmente por categorías.

Nota

Los textos del proyecto integrados en objetos de librería no pueden exportarse.

Consulte también

Importar textos del proyecto (Página 9194)

Importar textos del proyecto

El archivo xlsx se edita o bien se envía a un traductor. Una vez finalizada la traducción se importan los textos traducidos. Dentro del proyecto, los idiomas extranjeros se importan al objeto correspondiente.

Requisitos

- En el editor "Idiomas del proyecto" están activados por lo menos dos idiomas, p. ej. italiano y francés.

Importar textos del proyecto

Para importar un archivo con textos del proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto". Se abre el editor "Textos del proyecto".
3. Haga clic en el botón . Se abrirá el cuadro de diálogo "Importación".
4. Seleccione en el campo "Archivo de importación" la ruta y el nombre del archivo de importación.
5. Si ha realizado modificaciones en el idioma de origen del archivo de exportación y desea que las entradas correspondientes del proyecto se sobrescriban con estas modificaciones, active la casilla de verificación "Importar idioma de origen".
6. Haga clic en "Importar".

Resultado

Se han importado los textos del proyecto.

Consulte también

Exportar textos del proyecto (Página 9190)

10.9.5 Utilizar referencias cruzadas

10.9.5.1 Información general sobre referencias cruzadas

Introducción

La lista de referencias cruzadas ofrece una visión de conjunto del uso de objetos dentro del proyecto.

Uso de las referencias cruzadas

La lista de referencias cruzadas ofrece las ventajas siguientes:

- Al crear y modificar el proyecto se mantiene siempre una vista general de los objetos, variables, avisos, etc. utilizados.
- Desde las referencias cruzadas se puede saltar directamente a la ubicación correspondiente de los objetos.
- Al buscar errores se obtiene la información siguiente, p. ej.:
  - Qué objetos se utilizan en cada imagen y en cuál.
  - Qué avisos y recetas se visualizan en cada visor y en cuál.
  - Qué variable se utiliza en cada aviso u objeto y en cuál.
- Como parte de la documentación del proyecto, las referencias cruzadas ofrecen una amplia visión de conjunto de todos los objetos, avisos, recetas, variables, imágenes, etc. utilizados.

10.9.5.2 Mostrar la lista de referencias cruzadas

Introducción

La lista de referencias cruzadas proporciona información sobre el uso de objetos. La navegación del proyecto incluye referencias cruzadas a paneles de operador, carpetas y todos los editores. Por su parte, la vista detallada permite elegir también objetos concretos de los editores.

Requisitos

Se ha creado un proyecto.

Se han creado varios objetos.

Procedimiento

1. Seleccione la entrada deseada en la navegación del proyecto o en la vista detallada.
2. En el menú contextual, elija el comando "Referencias cruzadas". La lista de referencias cruzadas se abre en el área de trabajo.
3. Para ver en qué lugar se utilizan los objetos visualizados en la lista de referencias cruzadas, abra la ficha "Utilizado por".
4. Para ver quién utiliza los objetos visualizados en la lista de referencias cruzadas, abra la ficha "Utilizado".
5. Para ordenar las entradas de la columna "Objeto" en orden ascendente o descendente, haga clic en el título de la columna correspondiente.
6. Para acceder a la ubicación del objeto correspondiente, haga clic en el enlace que se visualiza.

Resultado

En el área de trabajo aparece la lista de referencias cruzadas correspondiente al objeto seleccionado.

10.9.5.3 Estructura de la lista de referencias cruzadas

Vistas de la lista de referencias cruzadas

Existen dos vistas de la lista de referencias cruzadas, que se diferencian por los objetos que aparecen en la primera columna:

- Utilizado por:
  Visualización de los objetos referenciados. Aquí se muestra en qué lugares se utiliza el objetos.
- Utilizado:
  Visualización de los objetos a referenciar. Aquí se muestran los usuarios del objeto.

Los tooltips asignados ofrecen más información sobre los objetos en cuestión.

Estructura de la lista de referencias cruzadas

Columna	Contenido/significado
Objeto	Nombre del objeto que utiliza los objetos subordinados o que es utilizado por éstos.
Número	Número de usos
Ubicación	Ubicación correspondiente, p. ej. objeto o evento
Propiedad	Función de los objetos referenciados, p. ej. variable para registro o valor de proceso
Conectado con	Variable de PLC con la que está conectado el objeto.
Tipo	Tipo del objeto
Ruta	Ruta del objeto

En función de los productos instalados se muestran columnas adicionales o distintas para las referencias cruzadas.

Ajustes en la lista de referencias cruzadas

Los ajustes siguientes se definen para la lista de referencias cruzadas mediante símbolos de la barra de herramientas:

- Actualizar la lista de referencias cruzadas
  Actualiza la lista actual de referencias cruzadas.
- Definir la configuración de la lista de referencias cruzadas
  Aquí se determina si deben mostrarse todos los objetos utilizados, todos los no utilizados, todos los definidos o todos los no definidos. Si la opción "Objetos no definidos" está activada, también se mostrarán referencias a objetos ya borrados.
- Reducir las entradas
  Reduce las entradas de la lista actual de referencias cruzadas cerrando los objetos subordinados.
- Ampliar las entradas
  Amplía las entradas de la lista actual de referencias cruzadas abriendo los objetos subordinados.

Clasificación en la lista de referencias cruzadas

Las entradas de la columna "Objeto" se pueden clasificar en orden ascendente o descendente. Para ello, haga clic en el título de la columna.

10.9.5.4 Mostrar referencias cruzadas en la ventana de inspección

Introducción

La información de las referencias cruzadas relativas a un objeto seleccionado se visualiza en la ventana de inspección "Información > Referencias cruzadas". La información de las referencias cruzadas se visualizan en la ventana de inspección en forma de tabla.

Requisitos

Se ha creado un proyecto.

Se han creado varios objetos.

Procedimiento

1. Seleccione un objeto en una imagen o en un editor tabular.
2. Seleccione "Información de referencia cruzada" en el menú contextual. En la ventana de inspección se muestran las referencias cruzadas.

Resultado

Para cada objeto seleccionado se muestran los puntos en los que se utiliza dicho objeto, así como los otros objetos que lo utilizan.

La tabla siguiente muestra la información detallada que aparece en la ficha "Información > Referencia cruzada":

Columna	Significado
Objeto	Nombre del objeto que utiliza los objetos subordinados o que es utilizado por éstos.
Número	Número de usos
Ubicación	Ubicación correspondiente, p. ej. objeto o evento
Propiedad	Función de los objetos referenciados, p. ej. variable para registro o valor de proceso
Conectado con	Variable de PLC con la que está conectado el objeto.
Tipo	Tipo del objeto
Ruta	Ruta del objeto

En función de los productos instalados se muestran columnas adicionales o distintas para las referencias cruzadas.

10.9.5.5 Reasignar variables en imágenes

Introducción

En el cuadro de diálogo "Modificar referencias a objeto" se pueden modificar referencias de variables en las propiedades de objetos de imagen.

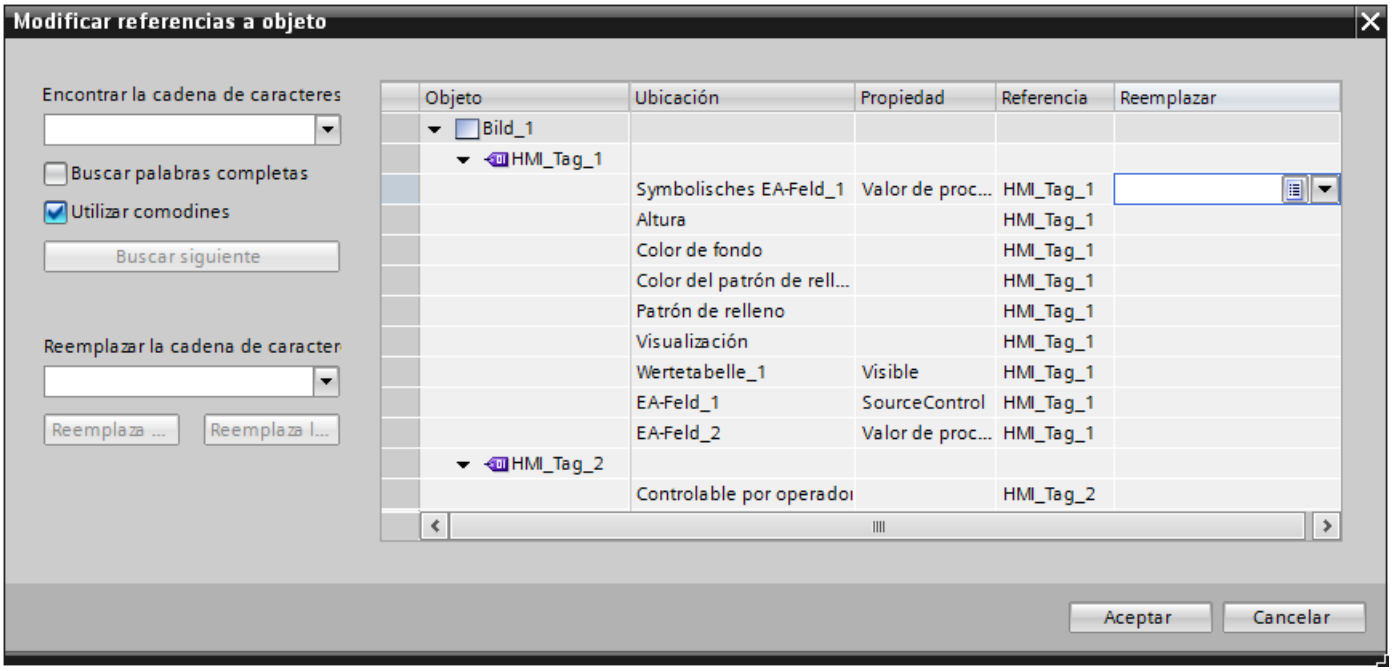
Esta función permite sustituir un gran número de variables en diversos objetos de imagen.

En los nombres de variables se puede buscar y reemplazar un carácter determinado o bien varios caracteres.

El cuadro de diálogo "Modificar referencias a objeto" se abre mediante el respectivo comando del menú contextual. Este cuadro de diálogo se puede abrir para uno o varios objetos de imagen o para una imagen completa.

Reemplazo de una variable

1. Seleccione los objetos que desee modificar de la imagen que contiene una variable.
2. Seleccione el comando "Modificar referencias a objeto" en el menú contextual o bien en el menú "Edición", .
   Se abre el cuadro de diálogo "Modificar referencias a objeto". En la columna "Objeto" se pueden ver los objetos de imagen seleccionados y las variables utilizadas.



3. Introduzca en el campo de entrada "Encontrar la cadena de caracteres en la referencia" el nombre o la parte del nombre que desee encontrar.
4. Con el botón "Buscar siguiente" se inicia la búsqueda. Cuando se encuentra una referencia, esta se marca en la columna "Referencia".
   Haciendo clic otra vez en el botón "Buscar siguiente" continúa la búsqueda hasta examinar toda la lista.
   A continuación, la búsqueda vuelve a comenzar desde el principio.
5. Introduzca en el campo de entrada "Reemplazar con" el nombre o la parte del nombre con el que desee reemplazar el nombre encontrado.
6. Haga clic en el botón "Reemplazar" para sustituir la variable seleccionada actualmente. Con el botón "Reemplazar todo" se sustituyen todas las referencias encontradas. Esto permite sustituir un gran número de referencias por otra referencia.

7. Para seleccionar una variable de la lista de todas las variables creadas en el proyecto, haga clic en el botón de la columna "Reemplazar".
8. Para seleccionar una variable de la lista de objetos o crear una nueva variable, haga clic en el botón de la columna "Reemplazar".
9. Confirme la entrada realizada con "Aceptar".

La reasignación de las variables se transfiere al proyecto.

Opciones adicionales para la búsqueda

La búsqueda puede optimizarse activando las siguientes opciones:

- Utilizar comodines
  Introduzca * para un número cualquiera de caracteres. Ejemplo: Desea buscar todas las variables que empiezan con "HMI". Introduzca "HMI*" en el campo de búsqueda. Introduzca ? para un único carácter.
- Buscar palabras completas
  Para buscar exclusivamente nombres completos, active la opción "Buscar palabras completas".

Sustitución de matrices y variables definidas por el usuario

Las variables cuyos nombres contienen los caracteres especiales. [ ] se representan entre comillas.

Las variables de tipo Array (matrices) y las variables de tipos de datos definidos por el usuario (UDT) contienen caracteres especiales para separar la variable de sus elementos. Sin embargo, no se representan entre comillas. Para reemplazar una matriz o un UDT por una variable HMI con caracteres especiales, introduzca el nombre de la variable HMI entre comillas en el campo de entrada "Reemplazar". Si se introduce un nombre sin comillas, se creará una referencia a una matriz o UDT inexistente en lugar de remitir a la variable HMI con caracteres especiales deseada.

Corrección de entradas

Si en el campo de entrada "Reemplazar con" se introduce un texto erróneo, las referencias existentes se reemplazarán por referencias a variables inexistentes.

Si detecta una entrada errónea antes de confirmar con "Aceptar", puede deshacer el error. Para ello, seleccione la columna "Reemplazar" y pulse la tecla "Suprimir". Se borrarán todas las entradas erróneas de la columna "Referencia".

Si ha confirmado el cuadro de diálogo con "Aceptar", las propiedades afectadas de los objetos de imagen se marcarán en rojo. En el cuadro de diálogo "Modificar referencias cruzadas" se marcan en rojo las variables inexistentes.

Si la nueva variable todavía no existe, deberá crearla posteriormente.

10.9.6 Gestionar idiomas

10.9.6.1 Conceptos lingüísticos en WinCC

Idioma de la interfaz de usuario e idiomas del proyecto

En WinCC se distingue entre dos niveles de idiomas:

- Idioma de la interfaz de usuario
  Durante la configuración, los textos de los menús y los cuadros de diálogo de WinCC aparecen en el idioma de la interfaz de usuario. El idioma de la interfaz también afecta a la rotulación de los objetos de control, así como a los parámetros de las funciones del sistema, la Ayuda en pantalla, etc.
- Idiomas del proyecto
  Los idiomas del proyecto son todos los idiomas en los que un proyecto debe editarse posteriormente. Los idiomas del proyecto se utilizan para crear un proyecto en varios idiomas.

Ambos niveles de idiomas son completamente independientes entre sí. Así, por ejemplo, es posible crear proyectos en inglés con una interfaz de usuario en alemán y viceversa.

Idiomas del proyecto

Dentro de los idiomas del proyecto se distingue entre los idiomas siguientes:

- Idioma de referencia
  El idioma de referencia es el idioma en el que se configura el proyecto en primer lugar. Durante la configuración, uno de los idiomas del proyecto se selecciona como idioma de referencia. El idioma de referencia se utiliza de idioma base para las traducciones. En primer lugar, se crean todos los textos del proyecto en el idioma de referencia y, posteriormente, se traducen. Durante la traducción pueden mostrarse simultáneamente los textos del idioma de referencia.
- Idioma de edición
  El idioma de edición es el idioma en el que se crean las traducciones de los textos. Tras crear el proyecto en el idioma de referencia, los textos se traducen a los demás idiomas del proyecto. Elija uno de los idiomas del proyecto como idioma de edición y edite los textos para el idioma correspondiente. El idioma de edición puede cambiarse en todo momento.

  Nota

  Al cambiar el idioma del proyecto también se cambia el idioma del teclado. En algunos idiomas (p. ej. el español), el sistema operativo impide cambiar el idioma del teclado. En tal caso, el teclado pasará a inglés.

- Idiomas de runtime
  Los idiomas de runtime son los idiomas del proyecto que se transfieren al panel de operador. En función de los requisitos del proyecto se decide qué idiomas del proyecto se transfieren al panel de operador.
  Para que el usuario pueda cambiar entre estos idiomas en runtime, ponga a su disposición los elementos de manejo correspondientes.

Consulte también

Configuración de idiomas en el sistema operativo (Página 9202)

Configuración en el sistema operativo para los idiomas asiáticos (Página 9203)

Seleccionar el idioma de la interfaz de usuario (Página 9204)

Editor "Gráficos" (Página 9215)

Idiomas de runtime (Página 9219)

Ejemplo: Configurar un botón para el cambio de idioma (Página 9226)

10.9.6.2 Configuración de idiomas en el sistema operativo

Introducción

Los ajustes realizados en el sistema operativo del PC de configuración afectan a las siguientes áreas de la administración de idiomas de WinCC:

- Selección de los idiomas del proyecto
- Representación de fechas, horas, importes y cifras en función del idioma
- Representación de caracteres ASCII

Selección del idioma del proyecto

Un idioma sólo está disponible como idioma del proyecto si se ha instalado previamente en el sistema operativo.

Representación de fechas, horas, importes y cifras en función del idioma

Mediante WinCC, en el campo de fecha y hora se define de forma fija la representación de fechas y horas para el idioma del proyecto y de runtime establecido.

Para que las fechas, horas y cifras se representen de forma correcta en el idioma de edición seleccionado, hay que ajustar este idioma en el Panel de control, bajo Configuración regional.

Representación de caracteres ASCII

En los campos de salida de texto, la representación de los caracteres ASCII a partir de 128 depende del idioma configurado y del sistema operativo utilizado.

Si los mismos caracteres especiales deben representarse en diversos PCs, el sistema operativo y la configuración regional utilizados deberán ser iguales en esos PCs.

Consulte también

Conceptos lingüísticos en WinCC (Página 9199)

Configuración en el sistema operativo para los idiomas asiáticos (Página 9203)

10.9.6.3 Configuración en el sistema operativo para los idiomas asiáticos

Configuración en sistemas operativos occidentales

Si desea introducir caracteres asiáticos deberá activar el soporte de dichos idiomas en el sistema operativo.

Para configurar textos con caracteres asiáticos, Windows dispone del Input Method Editor (IME). Sin este editor únicamente se podrán visualizar los textos asiáticos, pero no editar. Encontrará más información acerca de Input Method Editor en la documentación de Windows. Para introducir caracteres asiáticos en la configuración, conmute en el "Input Method Editor" al método de entrada asiático.

Para visualizar los textos del proyecto en función del idioma (p. ej. textos de avisos) con caracteres asiáticos, cambie el idioma del sistema operativo.

Configuración en sistemas operativos asiáticos

Si realiza la configuración en un sistema operativo asiático, deberá cambiar al idioma inglés si desea introducir caracteres ASCII (p. ej. para nombres de objetos). Puesto que el idioma inglés está comprendido en la instalación básica del sistema operativo, no es necesario que instale un idioma adicional.

Activar localización

1. Abra el Panel de control.
2. Seleccione el menú "Configuración regional y de idioma".
3. En la ficha "Idiomas", active la casilla de verificación "Instalar archivos para los idiomas de Asia oriental".
4. En el área "Servicios de texto e idiomas del dispositivo de entrada", haga clic en el botón "Detalles". Se abre el cuadro de diálogo "Servicios de texto e idiomas del dispositivo de entrada".
5. En la ficha "Configuración", agregue el idioma deseado en el área "Servicios instalados".
6. En la ficha "Avanzado" seleccione el idioma del sistema operativo en el área "Idioma para programas que no soportan Unicode".

Consulte también

Conceptos lingüísticos en WinCC (Página 9199)

Configuración de idiomas en el sistema operativo (Página 9200)

10.9.6.4 Ajustar idiomas del proyecto

Seleccionar el idioma de la interfaz de usuario

Introducción

El idioma de la interfaz de usuario se utiliza para mostrar opciones de menú, barras de título, textos de ayuda, textos de cuadros de diálogo y otras denominaciones existentes en la interfaz de usuario de WinCC.

Durante la configuración puede alternarse entre los distintos idiomas de interfaz de usuario instalados. Incluso después de cambiar el idioma de la interfaz, la designación de los objetos de control se mantiene en el idioma que se ha configurado al añadir el objeto.

Procedimiento

1. Elija el comando "Herramientas > Configuración".
   Se abre el diálogo "Configuración".
2. Seleccione en "General > Configuración general" el idioma de interfaz de usuario que desee.

Resultado

WinCC utilizará como idioma de la interfaz de usuario el idioma seleccionado.

Consulte también

Activar idiomas del proyecto (Página 9204)

Seleccionar idioma de referencia e idioma de edición (Página 9205)

Conceptos lingüísticos en WinCC (Página 9199)

Activar idiomas del proyecto

Introducción

En el editor "Idiomas del proyecto" puede configurar los idiomas del proyecto. Determine qué idioma del proyecto será el idioma de referencia y qué idioma será el idioma de edición.

Activar idiomas del proyecto

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Idiomas del proyecto".
   En el área de trabajo se visualizan los posibles idiomas del proyecto.
3. Active los idiomas que desee que estén disponibles como idiomas del proyecto.

Nota

Copiar objetos configurados en varios idiomas

Al copiar un objeto configurado en varios idiomas a otro proyecto sólo se copiarán los textos en los idiomas que estén activados en el proyecto de destino. Active en el proyecto de destino todos los idiomas de proyecto cuyos textos desee adoptar al copiar en el proyecto de destino.

Desactivar idiomas del proyecto

1. Desactive los idiomas que no desee que estén disponibles como idiomas del proyecto.

ATENCIÓN
Si desactiva un idioma del proyecto, se eliminarán todos los textos y gráficos del proyecto actual que se hayan creado en dicho idioma.

Consulte también

Seleccionar el idioma de la interfaz de usuario (Página 9202)

Seleccionar idioma de referencia e idioma de edición (Página 9205)

Seleccionar idioma de referencia e idioma de edición

Introducción

En el editor "Idiomas del proyecto" puede configurar los idiomas del proyecto. Determine qué idioma del proyecto será el idioma de referencia y qué idioma será el idioma de edición. El idioma de edición se puede modificar en cualquier momento.

Requisitos

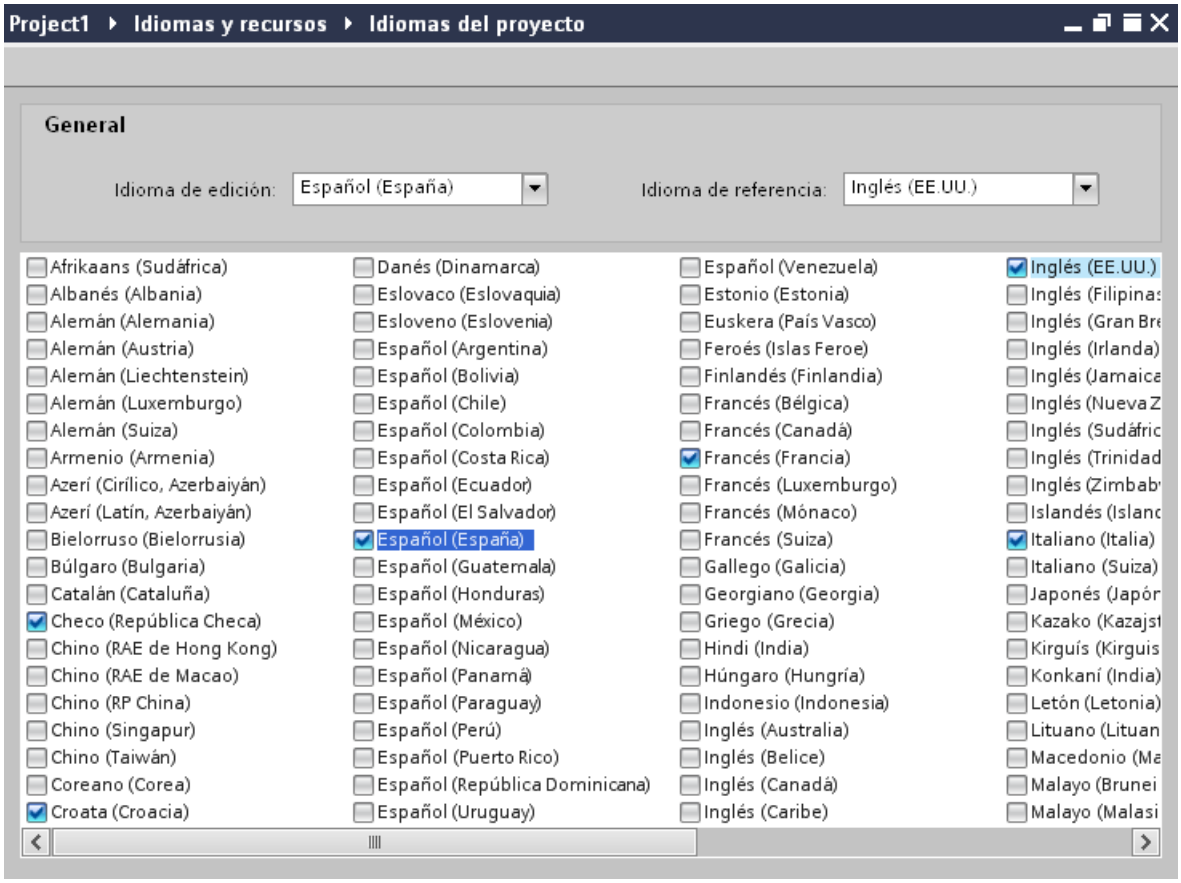
El editor "Idiomas del proyecto" está abierto.

Se activan varios idiomas de proyecto.

Seleccionar idioma de referencia e idioma de edición

1. En el área "General > Idioma de edición", haga clic en el símbolo de flecha de la lista de selección.
2. Haga clic en el idioma deseado de la lista de selección, p. ej. alemán.
3. En el área "General > Idioma de referencia", haga clic en el símbolo de flecha de la lista de selección.
4. Haga clic en el idioma deseado de la lista de selección, p. ej. inglés.

El idioma seleccionado se mostrará en el cuadro de lista.



Resultado

Se ha seleccionado el idioma de edición y el idioma de referencia.

Si cambia el idioma de edición, todos los textos que se introduzcan en adelante se guardarán en el nuevo idioma de edición.

Consulte también

Seleccionar el idioma de la interfaz de usuario (Página 9202)

Activar idiomas del proyecto (Página 9202)

10.9.6.5 Crear un proyecto en varios idiomas

Trabajar con varios idiomas

Configuración multilingüe en WinCC

WinCC permite configurar proyectos también en varios idiomas. Existen varios motivos para querer crear un proyecto en varios idiomas:

- Un proyecto se utiliza en varios países.
  El proyecto se crea en varios idiomas. Al poner en servicio el panel de operador sólo se le transfiere a éste el idioma utilizado por los operadores en el emplazamiento en cuestión.
- Los operadores de una instalación hablan idiomas distintos.
  Ejemplo: un panel de operador se utiliza en China pero el personal de servicio sólo entiende inglés.

Traducir textos del proyecto

Con WinCC es posible introducir directamente textos del proyecto en varios idiomas en diferentes editores, p. ej. en el editor "Textos del proyecto". Asimismo, WinCC pone a su disposición posibilidades de exportación e importación para traducir la configuración. Esto es especialmente útil cuando se configuran y traducen proyectos con un elevado porcentaje de texto.

Administración de idiomas y traducción en WinCC

Los editores siguientes de WinCC sirven para administrar idiomas y traducir textos:

Editor	Descripción
Idiomas del proyecto	Selección de los idiomas del proyecto, del idioma de edición y del idioma de referencia.
Idiomas y fuentes	Selección de los idiomas de runtime y de las fuentes utilizadas en el panel de operador.
Textos del proyecto	Administración centralizada de los textos configurados en todos los idiomas del proyecto.
Gráficos	Juego de gráficos para la administración de los gráficos y de sus variantes idiomáticas.

Consulte también

Principios básicos de los textos del proyecto (Página 9208)

Traducir textos directamente (Página 9209)

Traducir textos con ayuda de textos de referencia (Página 9211)

Exportar textos del proyecto (Página 9212)

Importar textos del proyecto (Página 9214)

Principios básicos de los textos del proyecto

Textos en distintos idiomas en el proyecto

Los textos que se visualizan durante la ejecución del proceso en los visualizadores se introducen generalmente en el idioma en el que se ha programado la solución de automatización. Asimismo, los comentarios y nombres de objetos se introducen sólo en ese idioma.

Si un operador no domina dicho idioma, requiere una traducción de todos los textos relevantes para el usuario en su lengua materna. Por tanto, todos los textos pueden traducirse a un idioma cualquiera. De esta manera se garantiza que toda persona que deba trabajar posteriormente con los textos del proyecto pueda visualizarlos en el idioma deseado.

Textos de usuario y textos del sistema

Para mejorar la comprensión se distingue entre textos de usuario y textos del sistema:

- Los textos de usuario son textos que han sido creados por el usuario.
- Los textos del sistema son textos que se crean automáticamente en el proyecto en función de la configuración.

Los textos del proyecto se gestionan en el editor de textos del proyecto. Este editor se encuentra en el árbol del proyecto en "Idiomas y recursos > Textos del proyecto".

Ejemplo de textos del proyecto en varios idiomas

Por ejemplo, los siguientes tipos de textos pueden gestionarse en varios idiomas:

- Textos visualizados
- Textos de aviso
- Comentarios recogidos en tablas
- Rotulaciones de objetos de imagen
- Listas de textos

Traducir textos

Existen dos procedimientos diferentes para traducir textos.

- Traducir textos directamente
  Las traducciones para los distintos idiomas del proyecto se pueden introducir directamente en el editor "Textos del proyecto".
- Traducir textos con ayuda de textos de referencia
  Si el número de textos no es muy elevado, se puede cambiar el idioma de edición. Mientras se muestran los textos del idioma de referencia, los textos nuevos se introducen en el idioma de edición.

Consulte también

Trabajar con varios idiomas (Página 9205)

Traducir textos directamente (Página 9209)

Traducir textos con ayuda de textos de referencia (Página 9211)

Exportar textos del proyecto (Página 9212)

Importar textos del proyecto (Página 9214)

Traducir textos directamente

Traducir textos

Si se utilizan varios idiomas en el proyecto, es posible traducir textos sueltos directamente. Tan pronto como se cambia el idioma de la interfaz de usuario, los textos traducidos están disponibles en el idioma en cuestión.

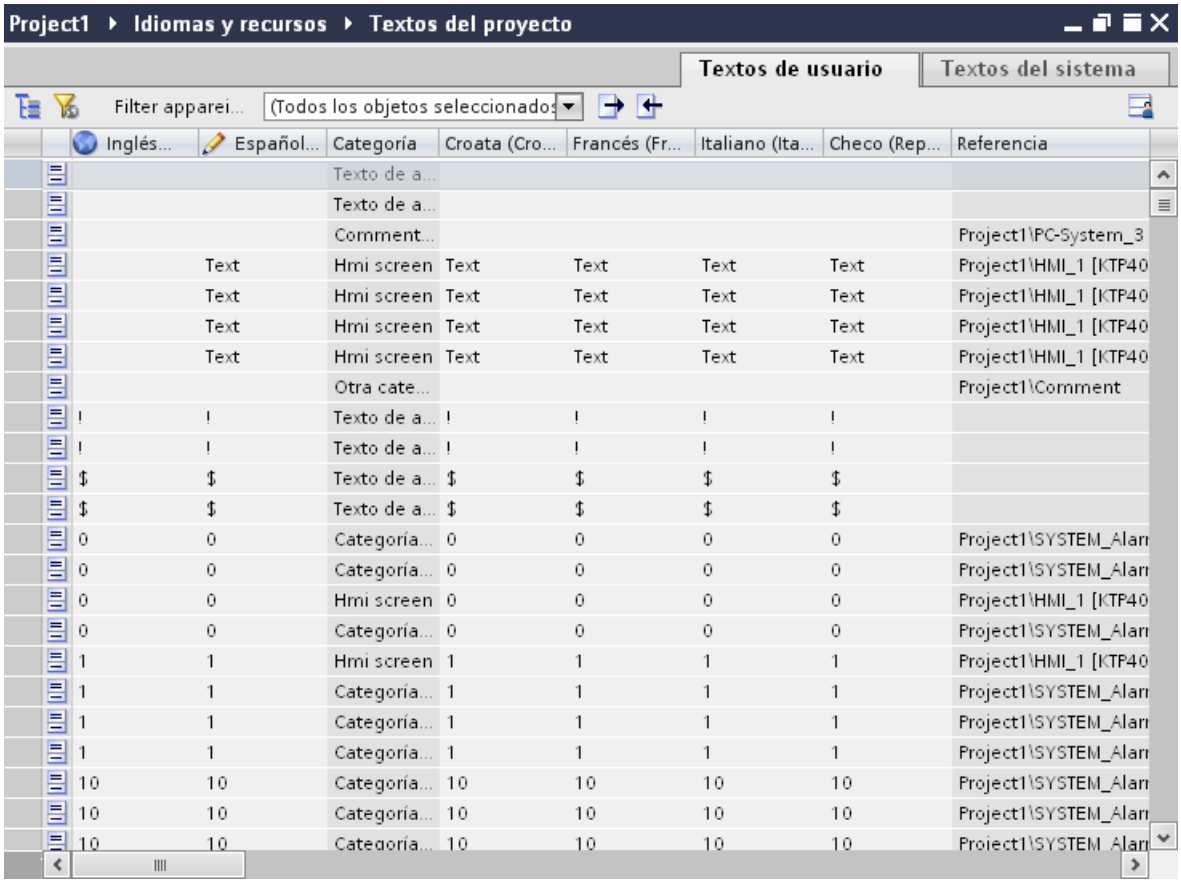
Requisitos

- La vista del proyecto está abierta.
- Hay un proyecto abierto.
- Se han seleccionado por lo menos dos idiomas de proyecto adicionales.

Procedimiento

Para traducir textos individuales, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto".
   En el área de trabajo aparece una lista con los textos del proyecto. Cada idioma del proyecto aparece en una columna propia.



3. Para agrupar textos idénticos y traducirlos de una sola vez, haga clic en el botón de la barra de herramientas.
4. Para ocultar textos sin traducir, haga clic en el botón de la barra de herramientas.
5. Haga clic en una columna vacía e introduzca la traducción.

Resultado

Se han traducido textos individuales en el editor "Textos del proyecto". Los textos se muestran en runtime en el idioma de runtime configurado en cada caso.

Consulte también

Trabajar con varios idiomas (Página 9205)

Exportar textos del proyecto (Página 9212)

Principios básicos de los textos del proyecto (Página 9206)

Importar textos del proyecto (Página 9214)

Traducir textos con ayuda de textos de referencia

Introducción

Tras cambiar el idioma de edición, todos los textos contenidos en los campos de entrada se visualizan en el nuevo idioma de edición. Si los textos no están traducidos todavía a ese idioma, los campos de entrada aparecen vacíos o se rellenan con los valores estándar.

Si se vuelve a introducir texto en un campo de entrada, el texto se guardará en el idioma de edición actual. Posteriormente, los textos de ese campo de entrada existirán en dos idiomas del proyecto, a saber: en el idioma de edición anterior y en el idioma de edición actual. De esta manera se pueden crear textos en varios idiomas del proyecto.

En un campo de entrada es posible visualizar las traducciones existentes en otros idiomas del proyecto. Éstas sirven de comparación para la entrada de textos en el idioma de edición actual y se denominan idioma de referencia.

Requisitos

Para un campo de entrada existe por lo menos una traducción a un idioma del proyecto adicional.

Procedimiento

Para visualizar la traducción del contenido de un campo de entrada en un idioma de referencia, proceda del siguiente modo:

1. Seleccione la opción "Tareas > Idiomas y recursos" en la Task Card.
2. Seleccione un idioma de referencia de la lista desplegable "Idioma de referencia".

Resultado

El idioma de referencia está predeterminado. Si hace clic en un campo de texto, en la Task Card "Tareas > Idioma de referencia" se visualizarán las traducciones existentes en otros idiomas del proyecto.

Consulte también

Trabajar con varios idiomas (Página 9205)

Exportar textos del proyecto (Página 9212)

Principios básicos de los textos del proyecto (Página 9206)

Importar textos del proyecto (Página 9214)

Exportar textos del proyecto

Se exportan textos del proyecto para traducirlos. Los datos se exportan a un archivo Office Open XML con la extensión ".xlsx". Este archivo puede editarse p. ej. con Microsoft Excel.

El archivo se puede intercambiar con los traductores y, tras ser traducido, se puede volver a importar directamente en el proyecto.

Requisitos

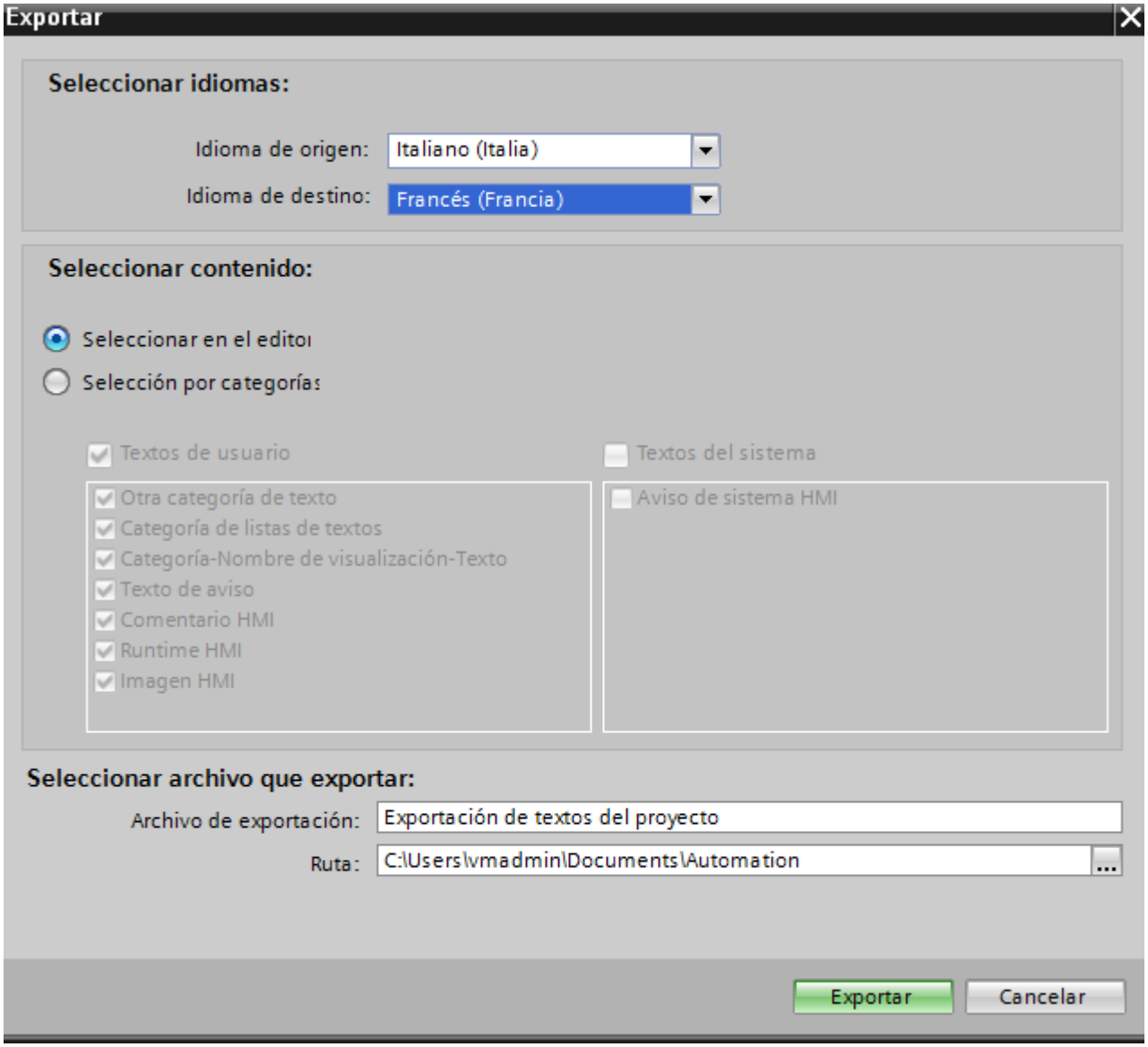
- En el editor "Idiomas del proyecto" están activados por lo menos dos idiomas, p. ej. el italiano y el francés.

Exportar textos del proyecto

Para exportar textos del proyecto individuales, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto". Se abre el editor "Textos del proyecto".
3. Seleccione los textos que desee exportar.

4. Haga clic en el botón . Se abrirá el cuadro de diálogo "Exportación".



5. Elija en la lista desplegable "Idioma de origen" el idioma a partir del cual desea efectuar la traducción, p. ej. el italiano.
6. Elija en la lista desplegable "Idioma de destino" el idioma al que desea traducir el texto, p. ej. el francés.
7. Indique en el campo de entrada "Archivo de exportación" una ruta y un nombre para el archivo de exportación.
8. Haga clic en "Exportación".

Resultado

Los textos seleccionados en el editor "Textos del proyecto" se han escrito en un archivo xlsx. El archivo xlsx se guarda en la carpeta indicada.

También es posible seleccionar y exportar todos los textos del proyecto por categorías. Dependiendo de los textos que desee exportar, active en el cuadro de diálogo "Exportación" las opciones "Textos de usuario" o "Textos del sistema". En este caso la exportación se puede limitar adicionalmente por categorías.

Nota

Los textos del proyecto integrados en objetos de librería no pueden exportarse.

Consulte también

Trabajar con varios idiomas (Página 9205)

Traducir textos con ayuda de textos de referencia (Página 9209)

Traducir textos directamente (Página 9207)

Principios básicos de los textos del proyecto (Página 9206)

Importar textos del proyecto (Página 9214)

Importar textos del proyecto

El archivo xlsx se edita o bien se envía a un traductor. Una vez finalizada la traducción se importan los textos traducidos. Dentro del proyecto, los idiomas extranjeros se importan al objeto correspondiente.

Requisitos

- En el editor "Idiomas del proyecto" están activados por lo menos dos idiomas, p. ej. italiano y francés.

Importar textos del proyecto

Para importar un archivo con textos del proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de "Idiomas y recursos".
   Se visualizan los elementos subordinados.
2. Haga doble clic en "Textos del proyecto". Se abre el editor "Textos del proyecto".
3. Haga clic en el botón . Se abrirá el cuadro de diálogo "Importación".
4. Seleccione en el campo "Archivo de importación" la ruta y el nombre del archivo de importación.
5. Si ha realizado modificaciones en el idioma de origen del archivo de exportación y desea que las entradas correspondientes del proyecto se sobrescriban con estas modificaciones, active la casilla de verificación "Importar idioma de origen".
6. Haga clic en "Importar".

Resultado

Se han importado los textos del proyecto.

Consulte también

Exportar textos del proyecto (Página 9210)

Trabajar con varios idiomas (Página 9205)

Principios básicos de los textos del proyecto (Página 9206)

Traducir textos directamente (Página 9207)

Traducir textos con ayuda de textos de referencia (Página 9209)

10.9.6.6 Utilizar gráficos específicos del idioma

Editor "Gráficos"

Introducción

En el editor "Gráficos" se administran los objetos gráficos configurados en varias versiones idiomáticas. Los proyectos multilingües exigen en parte también variantes idiomáticas de los gráficos, por ejemplo, cuando:

- los gráficos contienen texto,
- en los gráficos hay circunstancias culturales relevantes.

Abrir el editor "Gráficos"

Haga doble clic en "Idiomas y recursos > Gráficos" en el árbol del proyecto.

Área de trabajo

El área de trabajo visualiza mediante una tabla todos los objetos gráficos configurados. Para cada idioma del proyecto se crea una columna propia en la tabla. Las columnas de la tabla contienen las variantes gráficas de un idioma.

Asimismo, para cada gráfico puede definirse un gráfico estándar que se visualice siempre que no haya ningún gráfico específico para un idioma del proyecto.

Vista preliminar

Aquí puede visualizarse una vista preliminar de las representaciones de los gráficos en diferentes paneles de operador.

Consulte también

Agregar un gráfico externo a la colección de gráficos (Página 9217)

Agregar un gráfico a la colección de gráficos (Página 9216)

Conceptos lingüísticos en WinCC (Página 9199)

Agregar un gráfico a la colección de gráficos

Introducción

Con el editor "Gráficos" se importan los gráficos que se utilizarán en objetos de imagen en el editor "Imágenes". Asimismo, se gestionan las variantes idiomáticas de los gráficos. Una vista preliminar muestra las representaciones de un gráfico en diferentes paneles de operador.

Requisito

- Las variantes idiomáticas de un gráfico deben estar disponibles.
- En el editor "Idiomas del proyecto" hay varios idiomas activados.
- El editor "Gráficos" está abierto.

Insertar un gráfico

1. Haga clic en "Agregar" en la tabla "Gráficos". Se abre un cuadro de diálogo.
2. Seleccione el archivo gráfico deseado.
3. En la ventana de diálogo, haga clic en "Abrir".
   El gráfico se insertará en el proyecto y se visualizará en el editor "Gráficos" en todas las celdas de esa fila.
4. Haga clic en la celda correspondiente de un idioma para el que exista una variante idiomática de ese gráfico.
5. Elija la entrada "Insertar gráfico" del menú contextual. Se abre un cuadro de diálogo
6. Seleccione el archivo gráfico deseado y haga clic en "Abrir".
   En el lugar del gráfico del idioma de referencia se insertará la variante idiomática en la tabla.
7. Por último, inserte en la columna "Gráfico estándar" un gráfico que deba visualizarse en runtime en los idiomas para los que no exista ningún gráfico específico del idioma.

Con la función Drag&Drop se puede arrastrar un gráfico desde el Explorador de Windows hasta el punto deseado de la tabla "Gráficos".

Visualizar gráficos en la vista preliminar del panel de operador

1. Haga clic en un gráfico de la tabla.
2. En la ventana de inspección "Propiedades > Configuración de gráficos > Vista preliminar del panel de operador" seleccione el panel de operador que desee.
   El gráfico se visualizará en la vista preliminar tal y como se representará en runtime en el panel de operador seleccionado.

Resultado

Los gráficos insertados estarán disponibles en el editor "Gráficos". En la edición aparecerá el gráfico asignado al idioma de edición pertinente. En todos los idiomas de edición para los que no se ha importado ninguna imagen aparecerá la imagen estándar.

En runtime se visualizarán las imágenes asignadas al idioma de runtime correspondiente. En todos los idiomas de runtime para los que no se ha importado ninguna imagen aparecerá la imagen estándar.

Nota

Si desactiva un idioma del proyecto, se eliminarán todos los gráficos del proyecto actual que se hayan creado en dicho idioma.

Consulte también

Editor "Gráficos" (Página 9213)

Agregar un gráfico externo a la colección de gráficos

Introducción

Para poder visualizar los gráficos creados con un programa externo en las imágenes, depositelos previamente en el juego de gráficos del proyecto de WinCC.

Requisitos

- En el editor "Idiomas del proyecto" hay varios idiomas activados.
- El editor "Gráficos" está abierto.
- Hay un gráfico en el editor "Gráficos".

Crear e insertar un gráfico nuevo como objeto OLE

1. Haga clic en "Agregar" en la tabla "Gráficos". Se abre un cuadro de diálogo.
2. Desplácese hasta la carpeta en la que esté guardado el gráfico.

3. Haga clic en "Abrir" en el cuadro de diálogo.
   El gráfico se insertará en el proyecto y se verá en todas las celdas de esa fila en el editor "Gráficos".
4. Haga clic en la celda correspondiente de un idioma para la que exista una variante específica de idioma de ese gráfico.
5. En el menú contextual, seleccione la opción "Insertar objeto". Se abre el cuadro de diálogo "Insertar objeto".

   Nota

   También se abre el cuadro de diálogo "Ejecutando aplicación externa...". El cuadro de diálogo no se cerrará hasta que se cierre la aplicación externa.

6. Seleccione en el cuadro de diálogo "Insertar objeto > Crear nuevo" y un tipo de objeto.
7. Haga clic en "Aceptar". Se abre el programa de gráficos correspondiente.
8. Cuando haya creado el gráfico, cierre el programa de gráficos.
   El gráfico se guarda en el formato estándar del programa de gráficos y se muestra en la colección.

Insertar el gráfico creado en WinCC

1. Haga clic en la celda correspondiente de un idioma para la que exista una variante específica de idioma de ese gráfico.
2. En el menú contextual, seleccione la opción "Insertar objeto". Se abre el cuadro de diálogo "Insertar objeto".

   Nota

   También se abre el cuadro de diálogo "Ejecutando aplicación externa...". El cuadro de diálogo no se cerrará hasta que se cierre la aplicación externa.

3. En el cuadro de diálogo "Insertar objeto", seleccione la opción "Crear de archivo".
4. Haga clic en el botón de comando "Examinar".
5. Desplácese hasta el gráfico que ha creado y selecciónelo.

   Nota

   Para importar archivos gráficos tenga en cuenta las restricciones siguientes en cuanto al tamaño:

   *.bmp, *.tif, *.emf, *.wmf ≤4 Mbytes

   *.jpg, *.jpeg, *.ico, *.gif "*≤1 Mbyte

Resultado

Los objetos OLE insertados están disponibles en el editor "Gráficos".

En el editor "Imágenes" se muestra la variante gráfica del idioma de edición seleccionado actualmente. En todos los idiomas de edición para los que no se ha importado ninguna imagen, se muestra el gráfico predeterminado.

En runtime se muestra en cada caso la variante gráfica del idioma de runtime actual. En todos los idiomas de runtime para los que no se ha insertado ninguna imagen, se muestra el gráfico predeterminado.

Los objetos OLE depositados en la colección de gráficos se abren haciendo doble clic en ellos con el programa gráfico correspondiente para editarlos.

Consulte también

Editor "Gráficos" (Página 9213)

10.9.6.7 Idiomas de runtime

Idiomas de runtime

Utilizar varios idiomas de runtime

Es posible determinar qué idiomas del proyecto se utilizarán en runtime en un panel de operador determinado. El número de idiomas de runtime que pueden estar disponibles simultáneamente en el panel de operador depende del tipo de equipo. Para que el usuario pueda cambiar de idioma en runtime, debe configurarse un elemento de mando pertinente.

Al inicio de runtime se visualizará el proyecto en el idioma configurado en último lugar. En el primer inicio de runtime aparece el idioma con el número más bajo en "Secuencia de cambio de idioma".

Ajustes en la configuración

En el editor "Idiomas y fuentes" se determina

- qué idiomas del proyecto están disponibles como idiomas de runtime para el tipo de panel de operador en cuestión.
- en qué orden se cambian los idiomas al efectuar un cambio de idioma.

Consulte también

Métodos para cambiar de idioma (Página 9220)

Activar idioma de runtime (Página 9220)

Definir secuencia del idioma de runtime a la hora de cambiar el idioma (Página 9222)

Ajustar la fuente estándar para un idioma de runtime (Página 9224)

Seleccionar el idioma de archivación (Página 9225)

Conceptos lingüísticos en WinCC (Página 9199)

Métodos para cambiar de idioma

Introducción

Para que pueda haber varios idiomas de runtime disponibles en el panel de operador, es preciso configurar un cambio de idioma. Sólo de este modo el operador podrá elegir entre diferentes idiomas de runtime.

Métodos para cambiar de idioma

Pueden configurarse los métodos siguientes para cambiar de idioma:

- Selección directa de idioma
  Cada idioma se configura mediante un botón propio. En este caso, cree un botón para cada idioma de runtime.
- Cambio de idioma
  El operador cambia de idioma mediante un botón.

En ambos métodos, los títulos de los botones deben traducirse a todos los idiomas utilizados. Asimismo, puede configurarse un campo de salida que muestre el idioma seleccionado en ese momento.

Consulte también

Idiomas de runtime (Página 9217)

Seleccionar el idioma de archivación (Página 9225)

Activar idioma de runtime (Página 9220)

Definir secuencia del idioma de runtime a la hora de cambiar el idioma (Página 9222)

Ajustar la fuente estándar para un idioma de runtime (Página 9224)

Activar idioma de runtime

Introducción

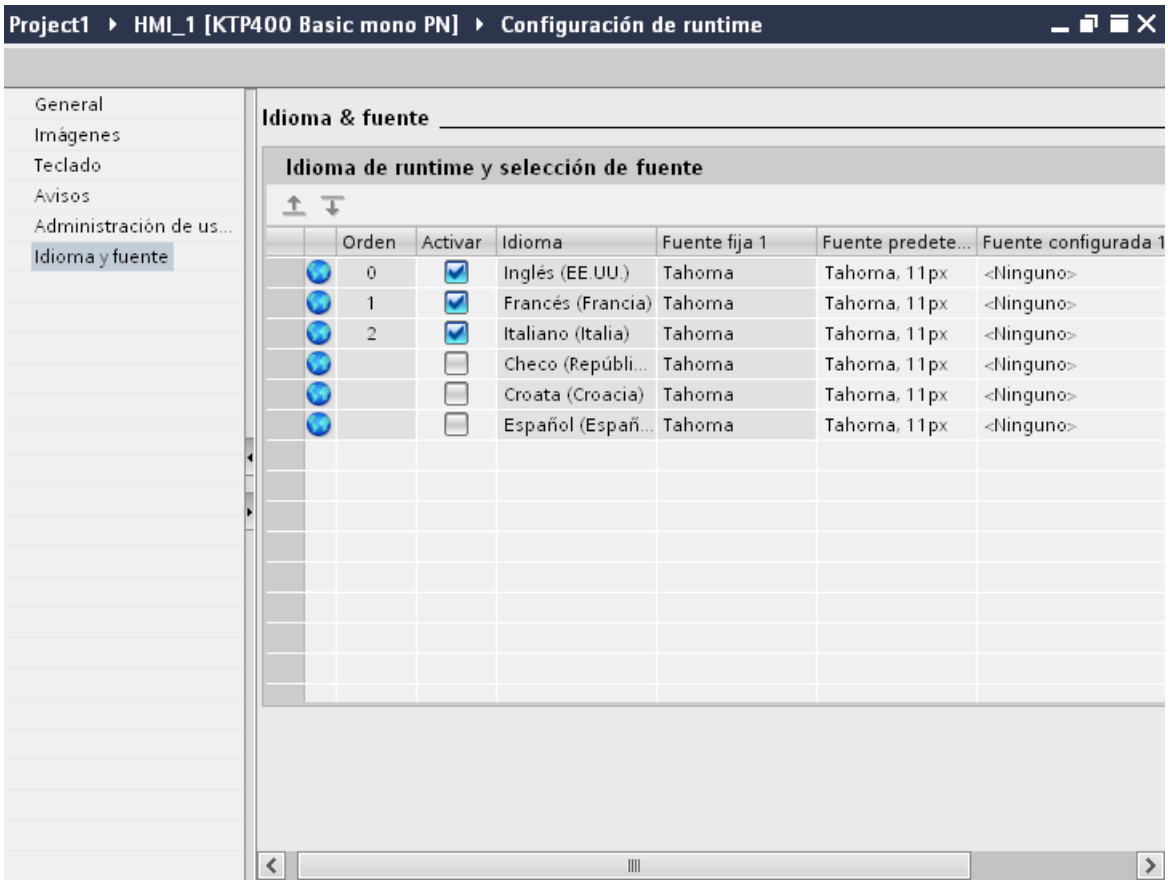
El editor "Idioma y fuente" muestra todos los idiomas disponibles en el proyecto. Aquí se elige qué idiomas del proyecto deben ser idiomas de runtime en el panel de operador.

Requisitos

En el editor "Idiomas del proyecto" hay varios idiomas activados.

Procedimiento

1. Haga doble clic en "Configuración de runtime" en el árbol del proyecto.
2. Haga clic en "Idioma y fuente".
3. Active los siguientes idiomas:
   - Inglés
   - Italiano
   - Francés



Resultado

Ha activado tres idiomas de runtime. En la columna "Secuencia" se asignará automáticamente un número a cada idioma. Los idiomas de runtime activados se cargarán en el panel de operador con el proyecto compilado.

Si selecciona más idiomas de los que pueden transferirse al panel de operador, esto se indicará marcando en color el fondo de la tabla.

Consulte también

Idiomas de runtime (Página 9217)

Seleccionar el idioma de archivación (Página 9225)

Definir secuencia del idioma de runtime a la hora de cambiar el idioma (Página 9222)

Métodos para cambiar de idioma (Página 9218)

Definir secuencia del idioma de runtime a la hora de cambiar el idioma

Introducción

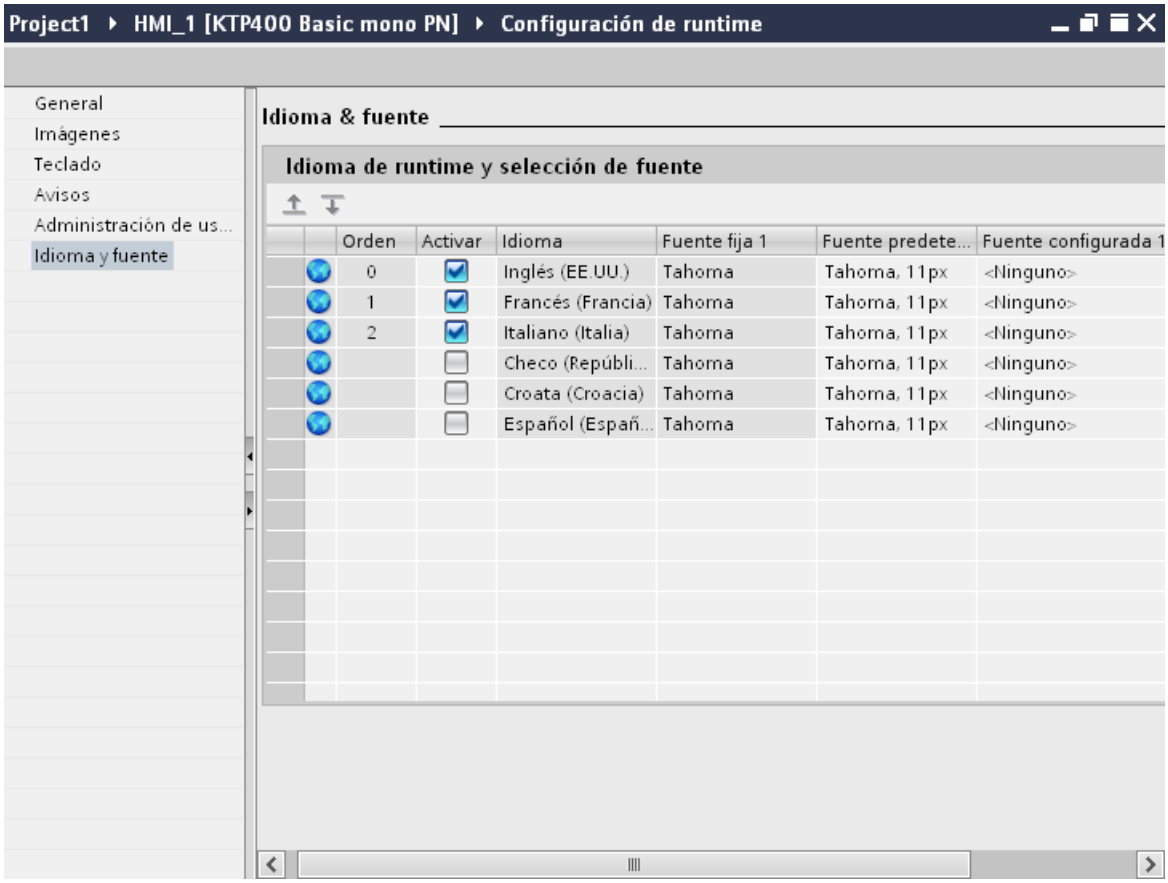
Se determina el orden en el que se cambiarán los idiomas de runtime cuando se realice un cambio de idioma. En el primer inicio de runtime aparece el proyecto en el idioma con el número más bajo en la columna "Orden".

Requisitos

- En el editor "Idiomas del proyecto" hay varios idiomas activados.
- En el editor está abierto "Idioma y fuente" y hay tres idiomas de runtime activados en el siguiente orden:
  1. Inglés
  2. Italiano
  3. Francés

Procedimiento

1. Seleccione el idioma de runtime "Italiano".
2. Haga clic en el botón . El idioma de runtime "Italiano" se desplazará hacia abajo. En la columna "Orden", el número cambiará automáticamente a "2".



Resultado

Se ha cambiado el orden de los idiomas de runtime. Al iniciar runtime por primera vez, el proyecto aparece en el idioma con el número más bajo, en este caso el chino. Si se efectúa un cambio de idioma, el idioma cambia en el orden de los números asignados.

Consulte también

Idiomas de runtime (Página 9217)

Seleccionar el idioma de archivación (Página 9225)

Activar idioma de runtime (Página 9218)

Ajustar la fuente estándar para un idioma de runtime (Página 9224)

Métodos para cambiar de idioma (Página 9218)

Ajustar la fuente estándar para un idioma de runtime

Introducción

El editor "Idioma y fuente" permite determinar para cada idioma de runtime con qué fuente se visualizarán los textos en el panel de operador. La fuente predeterminada se utilizará para todos los textos en los que no pueda definirse ningún tipo de fuente, como son los textos de los cuadros de diálogo.

En este caso, WinCC ofrece sólo las fuentes soportadas por el panel de operador.

Requisitos

- En el editor "Idiomas del proyecto" hay varios idiomas activados.
- En el editor "Idioma y fuente" hay tres idiomas de runtime activados.
  1. Chino
  2. Alemán
  3. Francés

Procedimiento

1. Haga doble clic en "Configuración de runtime" en el árbol del proyecto.
2. Haga clic en "Idioma y fuente". La tabla muestra los idiomas de runtime activados y las fuentes.
3. Haga clic en la fila correspondiente a "Francés" en la columna "Fuente estándar".
4. Seleccione la fuente que se utilizará por defecto cuando no pueda seleccionar ninguna fuente para un texto.

Resultado

Los textos del proyecto del idioma de runtime "Francés" se visualizarán con el tipo de fuente elegido en el panel de operador.

Al transferir, estos tipos de fuente también se transfieren al panel de operador.

La fuente predeterminada también se utilizará para representar los cuadros de diálogo del sistema operativo del panel de operador. Si los textos o los títulos de los cuadros de diálogo no se visualizan por completo, seleccione una fuente más pequeña como fuente predeterminada.

Consulte también

Idiomas de runtime (Página 9217)

Seleccionar el idioma de archivación (Página 9225)

Definir secuencia del idioma de runtime a la hora de cambiar el idioma (Página 9220)

Métodos para cambiar de idioma (Página 9218)

Seleccionar el idioma de archivación

Introducción

En el editor "Configuración de runtime > General" se selecciona el idioma en el que deben escribirse los ficheros en runtime.

Requisitos

- En el editor "Idiomas del proyecto" están activados los idiomas que utiliza en su proyecto, p. ej. "Alemán" e "Inglés".

Procedimiento

1. Haga doble clic en "Configuración de runtime" en la navegación del proyecto.
2. Haga clic en "Idioma y fuente".
3. Active los idiomas de runtime, p. ej., "Alemán" e "Inglés".
4. Defina el orden del proceso.
   - 1 Alemán
   - 2 Inglés
5. Haga clic en "Configuración de runtime > General".
6. Seleccione en "Ficheros > Idioma de archivación" el idioma "Alemán".

Resultado

Después de cargar, el proyecto se inicia en el idioma de runtime "Alemán". Los ficheros se escriben en alemán. En runtime, el operador cambia el idioma de runtime a inglés. No obstante, los ficheros se siguen escribiendo en alemán.

Consulte también

Idiomas de runtime (Página 9217)

Ajustar la fuente estándar para un idioma de runtime (Página 9222)

Definir secuencia del idioma de runtime a la hora de cambiar el idioma (Página 9220)

Métodos para cambiar de idioma (Página 9218)

Activar idioma de runtime (Página 9218)

10.9.6.8 Ejemplo de un proyecto multilingüe

Ejemplo: Configurar un botón para el cambio de idioma

Introducción

En este ejemplo se configurará un botón con el que podrá cambiar entre varios idiomas en runtime.

Requisitos

- Debe haber realizado el ejemplo "Configurar la etiqueta multilingüe de un botón".
- La imagen "Screen_1" está abierta.
- El botón se selecciona en la imagen.

Procedimiento

1. En la ventana de inspección haga clic en "Propiedades > Eventos > Pulsar".
2. En la tabla haga clic en "Agregar función".
3. Seleccione la función de sistema "EstablecerIdioma" y el ajuste "Conmutar".

Resultado

Se ha asignado la función "AjustarIdioma" al botón. Si pulsa el botón en runtime, se cambia el idioma de runtime. La secuencia de los idiomas de runtime se aplica tal como se ha predeterminado en el editor "Idioma y fuente" mediante la numeración de los idiomas.

Consulte también

Ejemplo: Configurar la etiqueta multilingüe de un botón (Página 9226)

Ejemplo: Configurar un botón para el cambio de idioma por cada idioma de runtime (Página 9227)

Conceptos lingüísticos en WinCC (Página 9199)

Ejemplo: Configurar la etiqueta multilingüe de un botón

Introducción

En este ejemplo se configurará un botón que en español se titula "Cambiar idioma" y en inglés "Switch Language".

Requisitos

- En el editor "Idiomas del proyecto" se activan los idiomas "Alemán" e "Inglés".
- El alemán se ha seleccionado como idioma de edición y como idioma de referencia.
- La imagen "Screen_1" está creada y abierta.
- La ventana de inspección está abierta.

Procedimiento

1. Arrastre un botón con la función de Drag&Drop desde la Task Card "Herramientas" hasta la imagen.
   El botón se inserta en la imagen.
2. En la ventana de inspección abra "Propiedades > Propiedades > General".
3. En "Texto > Off" seleccione la etiqueta "Sprache umschalten".
4. Confirme la entrada con <Intro>. El botón ya tiene etiqueta.
5. Abra la Task Card "Tareas".
6. Seleccione "Idiomas y recursos > Idioma de edición" y elija el idioma "Inglés".
7. Introduzca la etiqueta "Switch Language" en la ventana de inspección "Propiedades > Propiedades > General > Texto > Off".

Resultado

La etiqueta del botón ya está configurada en español y en inglés. En runtime se visualizará la etiqueta que corresponde al idioma de runtime actual.

Consulte también

Ejemplo: Configurar un botón para el cambio de idioma (Página 9224)

Ejemplo: Configurar un botón para el cambio de idioma por cada idioma de runtime (Página 9227)

Ejemplo: Configurar un botón para el cambio de idioma por cada idioma de runtime

Introducción

En este ejemplo se configurará un botón que en español se llama "Cambiar idioma" y en inglés "Switch Language".

Requisitos

- En el editor "Idiomas del proyecto" están activados los siguientes idiomas
  - Alemán
  - Inglés
  - Italiano
- Todos los idiomas están activados en el editor "Configuración de runtime > Idioma y fuente" como idioma de runtime.
- La imagen "Screen_1" está creada y abierta.
- En la imagen se han creado tres botones:
  - Botón_1 con la etiqueta "Deutsch"
  - Botón_2 con la etiqueta "English"
  - Botón_3 con la etiqueta "Italiano"
- La ventana de inspección está abierta.

Procedimiento

1. Seleccione el "botón_1".
2. En la ventana de inspección haga clic en "Propiedades > Eventos > Pulsar".
3. En la tabla haga clic en "<Agregar función>".
4. Seleccione la función de sistema "AjustarIdioma".
5. Haga clic en el campo "Cambiar".
6. Haga clic en el botón .
7. Seleccione "Idioma de runtime". El campo se marca en rojo.
8. En la lista desplegable seleccione "Alemán".
9. Repita los pasos 1 a 8 para los otros dos botones y seleccione el idioma de runtime correspondiente.

Resultado

Ha configurado tres botones para el cambio de idioma en runtime. Cada botón cambia a otro idioma de runtime. Si en runtime hace clic, por ejemplo, en el botón "Inglés", puede cambiar el idioma de runtime a inglés.

Consulte también

Ejemplo: Configurar un botón para el cambio de idioma (Página 9224)

Ejemplo: Configurar la etiqueta multilingüe de un botón (Página 9224)

10.9.7 Sustituir dispositivo

10.9.7.1 Principios básicos

Introducción

Al cambiar los paneles es posible utilizar configuraciones ya existentes para sus nuevos paneles y optimizarlas con pequeños ajustes manuales.

Todos los datos que configure permanecen en los datos de configuración. Por lo tanto, no es necesario copiar objetos individuales de un panel para copiarlos en otro.

Principio

Al cambiar un panel de operador se aplican los siguientes principios:

- Sólo están disponibles las funciones soportadas por el nuevo dispositivo. Sólo se visualizan los datos de configuración soportados por el nuevo dispositivo.
  Esto afecta, por ejemplo, a
  - las recetas
  - los objetos disponibles en imágenes,
  - las funciones de sistema disponibles,
  - los protocolos de comunicación disponibles.
- En el nuevo panel el número de objetos soportados, por ejemplo, imágenes o variables, puede ser limitado. Si los objetos existentes superan las limitaciones del nuevo dispositivo, los objetos se visualizan por completo. En todo caso, los objetos se identifican mediante colores en los diferentes editores. Al compilar los datos de configuración se notifica un error.
  Por lo tanto, al cambiar a un panel de rendimiento inferior es necesario realizar pequeños ajustes manuales.
  Ejemplo: Limitación de cantidad de conexiones
  Si en el nuevo dispositivo se soportan menos conexiones de las configuradas, todas las conexiones se marcan en color como no válidas. Borre las conexiones que sobrepasen el número permitido.

Nota

Si se cambia un panel, como puede ser un Panel, y selecciona una estación PC como nuevo panel, WinCC Runtime Advanced se desplaza automáticamente debajo de la estación PC en el árbol del proyecto.

Consulte también

Funciones dependientes del panel (Página 9230)

Posibilidades para la adaptación de las imágenes (Página 9233)

Asignación de teclas cuando se sustituye un dispositivo (Página 9231)

Engineering System (Página 9341)

Basic Panel (Página 9342)

10.9.7.2 Funciones dependientes del panel

Funciones dependientes del panel

Funciones dependientes del panel

Las funciones dependientes del panel se implementan del siguiente modo:

- Colores
  Si cambia de un panel con visor de color a uno que tenga una gama de colores menor, el color se cambia automáticamente.
  Si el usuario cambia manualmente el color y vuelve a cambiar a un panel con una gama de colores mayor, se mantiene la gama de colores reducida.
- Fuentes
  Si en un panel de operador no existe una fuente configurada, se sustituye por una fuente similar o por la fuente predeterminada configurada. La fuente predeterminada depende del dispositivo seleccionado.
- Juegos de caracteres con distintos tamaños de fuente
  Evite utilizar muchos tamaños de fuente distintos cuando realice la configuración en los siguientes paneles de operador:
  - OP 73
  - OP 77A
  - TP 177A

  Por cada fuente se carga en el dispositivo un juego de caracteres. Al compilar, compruebe en la ventana de inspección la cantidad de memoria del dispositivo que está ocupada por juegos de caracteres.

- Tamaño de fuente
  Para mostrar el texto en los dispositivos, utilice fuentes de Windows pequeñas. Si utiliza fuentes de Windows grandes, es posible que, en función del tamaño de la pantalla, no se pueda mostrar el texto completo.
  Si utiliza para los dispositivos OP 77A y TP 177A fuentes mayores o iguales a 28 píxeles, el rendimiento se resentirá.
  En los idiomas asiáticos, el número de caracteres es notablemente mayor. Por eso, si se utilizan tamaños de fuente distintos, el consumo de memoria aumenta significativamente en todos los dispositivos.
  Para lograr una configuración de alto rendimiento, utilice la misma fuente para todos los textos grandes de la totalidad del proyecto.
- Imágenes y objetos de imagen
  Si, al cambiar un panel, el nuevo soporta una resolución distinta a la del panel anterior, existen diversas posibilidades para adaptar las imágenes.
  El menú "Herramientas > Configuración > Visualización > Adaptaciones de tamaño de imágenes" permite adaptar el tamaño de las imágenes al nuevo dispositivo.

Consulte también

Principios básicos (Página 9227)

Asignación de teclas cuando se sustituye un dispositivo

Introducción

Como los dispositivos disponibles están equipados con diferentes teclas de función, cuando se sustituye el dispositivo las funciones configuradas en él se representan en las teclas de función disponibles en el nuevo panel de operador.

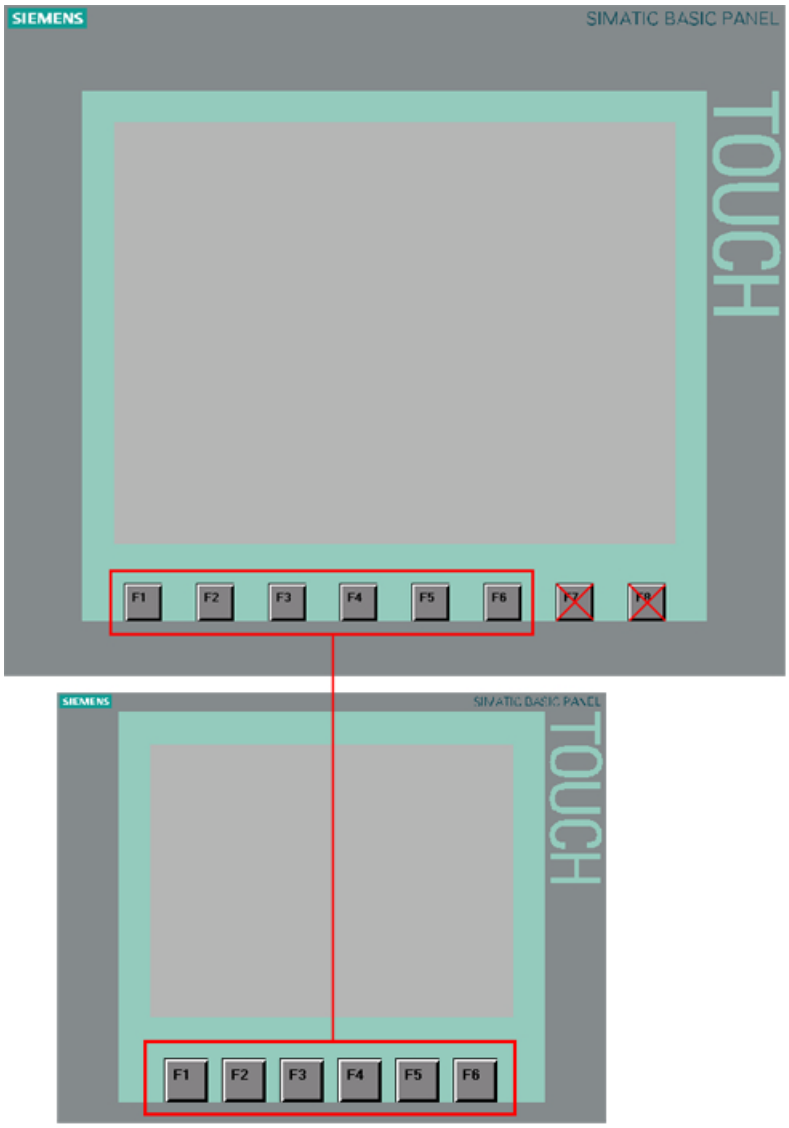
Representación de las teclas de función

Las teclas de función ubicadas debajo de la pantalla se representan de izquierda a derecha en el nuevo dispositivo. Si el nuevo dispositivo tiene menos teclas, las teclas que faltan no se representan.

Ejemplo: Cambio del KTP1000 Basic por el KTP600 Basic

Se ha configurado una función en el KTP1000 Basic para la tecla F2. Al cambiar a un KTP600 Basic esta función también se activa al pulsar la tecla F2.

Sin embargo, si se ha utilizado la tecla F7 en un KTP1000 Basic, esta función ya no está disponible después de cambiar a un KTP600 Basic.



Representación de la teclas de control y de las teclas de cursor

Las teclas siguientes se representan sólo en las mismas teclas del nuevo dispositivo:

- HELP
- ESC
- ACK
- ENTER
- PAGE UP
- PAGE DOWN
- CURSOR UP
- CURSOR DOWN

Consulte también

Principios básicos (Página 9227)

Posibilidades para la adaptación de las imágenes (Página 9233)

10.9.7.3 Adaptar imágenes al nuevo dispositivo

Posibilidades para la adaptación de las imágenes

Introducción

Antes de sustituir un dispositivo seleccione la adaptación del tamaño de las imágenes. La adaptación del tamaño resulta importante sobre todo cuando se sustituyen paneles con diferentes resoluciones de pantalla.

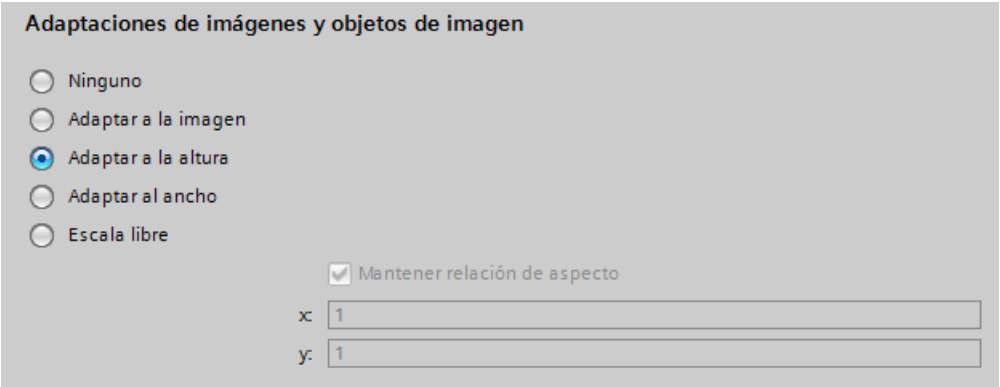
En objetos como los visores de gráficos o los campos de texto la adaptación de los objetos puede depender del contenido de los mismos.

Nota

Si se sustituye un dispositivo con una pantalla en formato horizontal (paisaje) por un dispositivo con formato vertical, los objetos se representan distorsionados. Así, por ejemplo, al producirse un cambio en el formato de la pantalla puede que la rotulación de los objetos se vea cortada o que el contenido no esté ajustado al objeto. Por lo tanto, es recomendable adaptar siempre las imágenes al nuevo dispositivo una vez sustituido.

Adaptación de las imágenes cuando se sustituye un dispositivo

El menú "Herramientas > Configuración > Visualización > Adaptaciones de imágenes y objetos de imagen" permite adaptar el tamaño de las imágenes al nuevo dispositivo.



Elija uno de los ajustes siguientes:

Ninguno (ajuste estándar)

Las imágenes no se escalan. Los objetos mantienen su posición y su tamaño en la imagen. Utilice este ajuste como un primer test para comprobar el posible resultado de la sustitución y que no se hayan producido pérdidas de redondeo en el proceso de sustitución.

Cuando la pantalla del nuevo dispositivo es más pequeña, esta opción puede hacer que haya objetos fuera del área configurable.

Adaptación a imagen

La posición y el tamaño de objeto se adaptan al nuevo tamaño de pantalla. La adaptación se realiza en los ejes X y Y. Así, los gráficos y los tamaños de fuente se adaptan según corresponda.

Adaptar a la altura

La relación de aspecto se mantiene y las imágenes se adaptan a la altura del nuevo dispositivo.

Utilice esta opción si, por ejemplo, cambia un panel con el formato de pantalla 4:3 por un panel con pantalla ancha.

Adaptar al ancho

La relación de aspecto se mantiene y las imágenes se adaptan al ancho del nuevo dispositivo.

Utilice esta opción si, por ejemplo, cambia un panel con el formato de pantalla ancha por uno con pantalla 4:3.

Escala libre

Aquí es posible seleccionar un factor de escala libre para la adaptación de la imagen. Este factor puede definirse tanto para el eje x como para el eje y.

Si se utiliza un factor inferior a 1 en la escala libre, pueden producirse distorsiones en los objetos. Por ejemplo, puede que la rotulación de los objetos aparezca cortada o que el contenido no esté ajustado al objeto.

Por lo tanto, es recomendable adaptar siempre las imágenes al nuevo dispositivo una vez sustituido.

Nota

En los objetos con relación de aspecto fija, como es un indicador gauge o un círculo, no se produce una adaptación de la proporción de página. Los objetos se representan en el nuevo dispositivo con la misma relación de aspecto que había antes del cambio.

Consulte también

Definir la posición de los objetos de imagen (Página 9238)

Adaptar objetos al contenido (Página 9235)

Principios básicos (Página 9227)

Asignación de teclas cuando se sustituye un dispositivo (Página 9229)

Adaptar objetos al contenido

Introducción

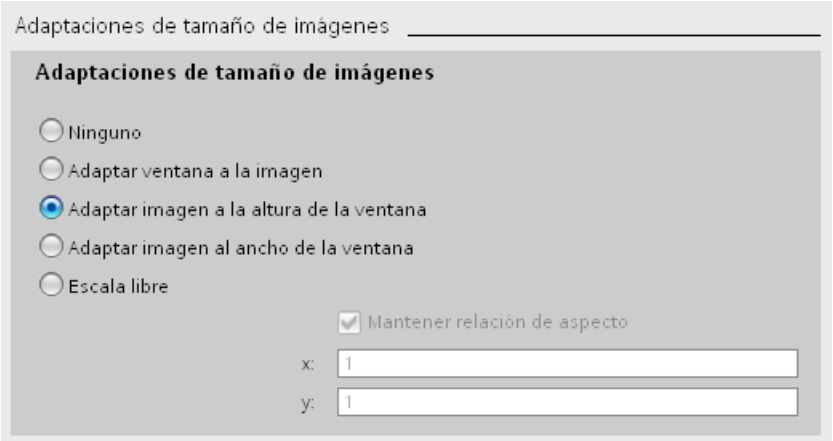
En algunos objetos la ventana de inspección permite establecer que el tamaño de los mismos se adapte al contenido de cada momento; por ejemplo:

- Campo de texto: Adaptar tamaño al texto contenido.
- Campo E/S: Adaptar tamaño al texto contenido.
- Campo E/S simbólico: Adaptar tamaño al texto contenido o a la lista de textos.
- Visor de gráficos: Adaptar tamaño al gráfico contenido.

Adaptación de tamaño para los objetos de texto y los objetos gráficos

El menú "Herramientas > Configuración > Visualización > Adaptaciones de tamaño de imágenes > Adaptar objetos al contenido" permite desactivar la adaptación automática del tamaño de los diversos objetos. De esta forma los objetos se escalan tal como se ha definido en "Herramientas > Configuración > Visualización > Adaptaciones de tamaño de imágenes.

Seleccione los objetos en los que desee desactivar la adaptación automática del tamaño.



- Si se ha activado "Desactivar la adaptación de tamaño para objetos de texto", en las propiedades de los objetos de texto se ignora la adaptación automática del tamaño.
  Si, por ejemplo, se ha activado la opción "Adaptar imagen a la altura de la ventana", el campo de texto se escala como los demás objetos, conforme a la altura del nuevo panel.
- Si se ha activado "Desactivar adaptación de tamaño para objetos gráficos", en las propiedades de los objetos gráficos se ignora la adaptación automática del tamaño.
  Si, por ejemplo, se ha activado la opción "Adaptar imagen al ancho de la ventana", el visor de gráficos se escala como los demás objetos, conforme a la altura del nuevo panel.

Nota

Los ajustes no tienen efecto alguno en los objetos de imagen cuyo tamaño no puede cambiarse, como son los indicadores de aviso o los objetos de imagen con una proporción de página fija.

Las funciones para desactivar la adaptación de tamaño para objetos de texto y desactivar la adaptación de tamaño para objetos gráficos no tienen efecto si:

- Se ha activado la opción "Adaptaciones de imágenes y objetos de imágenes > Ninguna".

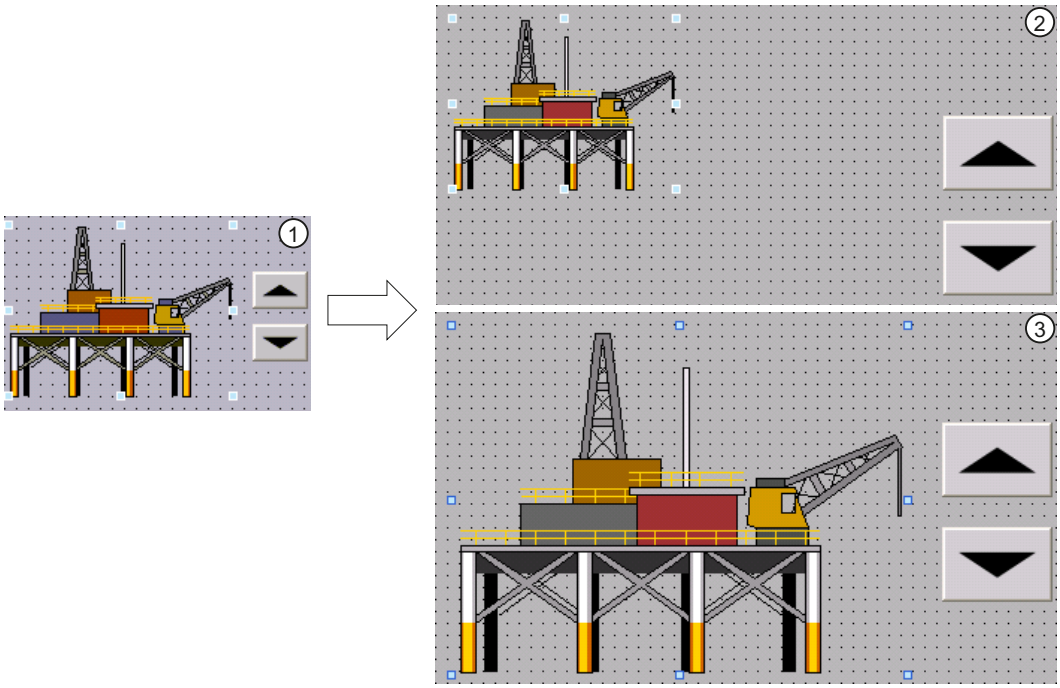
- Se ha activado la opción "Adaptar imagen a ancho y altura de la ventana" y el nuevo panel tiene la misma resolución que el actual.

- Se ha activado la opción "Adaptar imagen a la altura de la ventana" y el nuevo panel tiene la misma resolución que el actual.

- Se ha activado la opción "Adaptar imagen al ancho de la ventana" y el nuevo panel tiene la misma resolución que el actual.

Ejemplo

La siguiente figura muestra los efectos que se derivan de la adaptación automática del tamaño tomando como ejemplo un objeto gráfico en el que están alineados dos botones:



① Situación de partida:

- Dos botones están alineados a un objeto gráfico.
- En las propiedades del objeto gráfico, en "Representación > Adaptación del tamaño", está activada la opción "Adaptar tamaño objeto al gráfico".

② Posibilidad 1: Después de cambiar de panel de operador se deben conservar las propiedades originales del objeto gráfico.

- En la pestaña "Adaptación del tamaño de los objetos" de la configuración, desactive la opción "Impedir la adaptación de tamaño de objetos gráficos".

Efecto: el objeto gráfico conserva su tamaño original tras cambiar de panel de operador. Se pierde la alineación con los botones.

③ Posibilidad 2: Después de cambiar de panel de operador, el objeto gráfico debe ubicarse con relación a la nueva resolución de pantalla.

- En la pestaña "Adaptación del tamaño de los objetos" de los ajustes, active la opción "Impedir la adaptación de tamaño de objetos gráficos".

En las propiedades del objeto gráfico se activa automáticamente la opción "Adaptar gráfico a tamaño objeto". Los dos botones estarán correctamente alineados al objeto gráfico incluso después de cambiar de panel de operador.

Consulte también

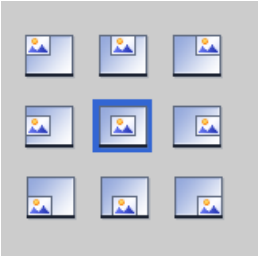
Definir la posición de los objetos de imagen (Página 9238)

Posibilidades para la adaptación de las imágenes (Página 9231)

Definir la posición de los objetos de imagen

Introducción

Existen varias formas de adaptar la posición de los objetos de imagen al nuevo dispositivo.



Seleccionar posición

El menú "Herramientas > Configuración > Visualización > Adaptaciones de tamaño de imágenes > Seleccionar posición" permite adaptar la posición de los objetos de las imágenes al nuevo panel.

Ejemplo

La siguiente opción alinea los objetos en el borde izquierdo superior.

La siguiente opción centra los objetos en el centro de la pantalla.

Consulte también

Adaptar objetos al contenido (Página 9233)

Posibilidades para la adaptación de las imágenes (Página 9231)

10.9.8 Copiar entre paneles de operador y editores

10.9.8.1 Principios básicos

Principios básicos

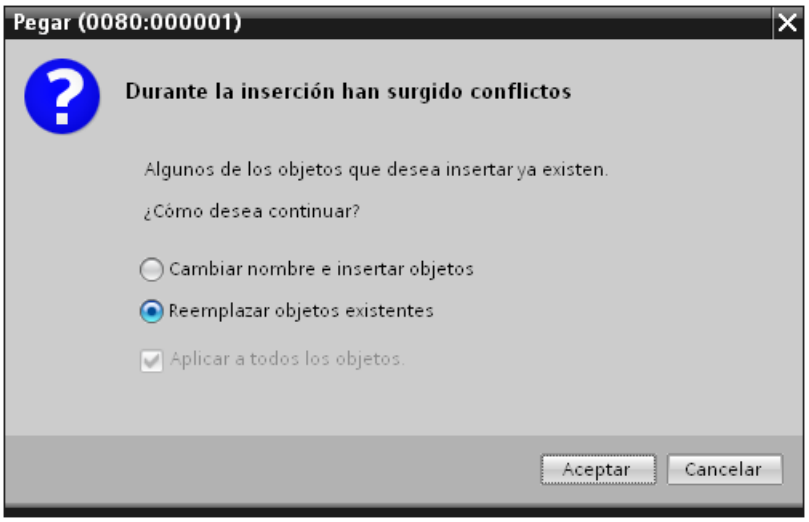
Copiar y pegar dentro de un panel de operador

Se copian y pegan objetos (p. ej. objetos de visualización) dentro de un panel de operador. Si el objeto ya está creado en el editor, al pegar se adjunta automáticamente un número al nombre de objeto, de acuerdo con la siguiente lógica:

- "<Nombre de objeto>_1" pasa a denominarse "<Nombre de objeto>_2".
- "<Nombre de objeto>_2" pasa a denominarse "<Nombre de objeto>_3".

Copiar y pegar entre paneles de operador

También es posible copiar y pegar de un panel de operador a otro. Si ya hay un objeto con el mismo nombre, existen las siguientes posibilidades:



Nota

Excepción a esta regla básica

Esta regla básica no se aplica al copiar y pegar una categoría generada a partir de una categoría válida para todo el proyecto. Si la categoría copiada ya existe en el panel de operador de destino dentro del mismo proyecto, no se ejecuta el comando "Pegar".

Copiar carpetas definidas por el usuario

En la navegación del proyecto se crea una carpeta definida por el usuario para editores, p. ej. para variables HMI, imágenes, etc.

Se copia una carpeta definida por el usuario y se pega en otro panel de operador. Los objetos contenidos en una carpeta definida por el usuario pueden salvar las limitaciones existentes para otro panel de operador, p. ej. en lo referente al número de imágenes soportadas. Después de pegar se muestran todos los objetos. Al compilar los datos del proyecto se emite un error.

No se pueden copiar en ningún caso carpetas del sistema.

Consulte también

Objetos y funciones no soportados (Página 9240)

Objetos y funciones no soportados

Introducción

Al copiar un objeto, todas las propiedades y configuraciones del mismo se aplican en el panel de operador de destino.

Objetos no soportados

Los objetos que no se soportan en el panel de operador de destino no pueden pegarse.

Nota

Si se copia una imagen que contiene objetos no compatibles con el panel de operador de destino, los objetos se mantienen en segundo plano. Si se vuelve a copiar la imagen y el nuevo panel soporta los objetos, éstos se muestran de nuevo.

Objetos no válidos

Los siguientes objetos dejan de ser válidos después de pegarlos en el panel de operador de destino:

- Los objetos referenciados que no existen en el panel de operador de destino.
- Los objetos con una configuración que no se soporta en el panel de operador de destino.
- Funciones de sistema configuradas en objetos, las cuales no se soportan en el panel de operador de destino.

Los objetos no válidos se identifican mediante una marca de color. Seleccione un objeto soportado o cree un objeto nuevo. Si se conserva un objeto no válido, al compilar los datos de proyecto se notifica un error.

Colores y tipos de fuente

Los paneles de operador soportan los colores y los tipos de fuente en diferente medida. Al realizar la operación "Pegar", los colores y tipos de fuente no soportados se sustituyen por colores y tipos de fuente soportados. Si vuelve a pegar el mismo objeto en el panel de operador de origen, la configuración de origen vuelve a estar activa.

Consulte también

Principios básicos (Página 9237)

10.9.8.2 Copiar y pegar

Copiar imágenes

Introducción

Es posible copiar una o más imágenes de la carpeta "Imágenes" y pegarlas en la carpeta "Imágenes" de otro panel de operador.

Tipo y tamaño de la pantalla

En los paneles de operador con teclas, las teclas disponibles se visualizan automáticamente en la imagen. Al copiar una imagen entre los paneles de operador, las teclas se muestran o se ocultan. Las funciones configuradas en las teclas de función no se aplican.

Si en el panel de operador de destino hay menos espacio para la imagen que en el panel de operador de origen, es posible adaptar el tamaño de los objetos existentes y las distancias entre dichos objetos.

Adaptación automática del tamaño de los objetos

1. Elija el comando "Herramientas > Configuración > Visualización > Adaptaciones de tamaño de imágenes".
2. Active, por ejemplo, la opción "Adaptar a la altura".

Consulte también

Copiar recetas en un mismo panel de operador (Página 9242)

Copiar objetos con objetos enlazados (Página 9242)

Objetos enlazados copiados automáticamente (Página 9243)

Copiar recetas en un mismo panel de operador

Editor "Recetas"

Es posible copiar recetas, elementos de receta y registros de receta dentro de la tabla respectiva. Se copia un elemento de receta a otra receta.

Sólo WinCC Runtime Professional: Se copia un elemento de vista de receta a otra vista de receta. Si ya existe un elemento de vista de receta con el mismo nombre, aparece un cuadro de diálogo de conflicto. Se puede escoger entre sustituir el elemento de receta o renombrarlo. Es posible copiar elementos de receta en la primera fila vacía del editor "Vistas de recetas" en la ficha "Elementos".

Es posible copiar un registro de receta a otra receta si la otra receta contiene el mismo número de elementos de receta. Si los tipos de datos son diferentes, el valor se copia en el registro de destino pero recibe una marca de error.

Editor "Variables"

Con la función Drag&Drop se puede arrastrar una variable hasta un elemento de receta de la columna "Variable". La variable se asocia al elemento de receta. Si ya hay una variable asociada, se visualizará un mensaje de error.

Editor "Imágenes"

Si arrastra una receta hasta una imagen con la función Arrastrar y soltar, se crea un nuevo visor de recetas que se asocia a la receta.

Consulte también

Copiar imágenes (Página 9239)

Copiar objetos con objetos enlazados (Página 9242)

Objetos enlazados copiados automáticamente (Página 9243)

Copiar objetos con objetos enlazados

Introducción

Un objeto se enlaza con otro objeto, por ejemplo, en los siguientes casos:

- Ha definido una variable como variable de disparo para un aviso.
  El aviso es el objeto. La variable es el objeto enlazado.
- Ha definido una conexión para una variable externa.
  La variable es el objeto. La conexión es el objeto enlazado.

El objeto se pega siempre por completo al copiar y pegar. Si el objeto enlazado se pega o no, depende del comando utilizado para pegar.

Inserción simple

El objeto enlazado no se copiará. El enlace del objeto se aplica y se incluye en el panel de operador de destino, tal como se indica a continuación:

- Si hay un objeto del mismo nombre, el objeto existente se utiliza con su configuración.
- Si no hay un objeto del mismo nombre, se visualiza el nombre del objeto. El objeto deja de ser válido.

En algunos casos, los objetos enlazados se copian y se pegan automáticamente durante la operación de pegado simple.

Inserción avanzada

Si en el menú contextual selecciona el comando "Inserción avanzada" puede determinar que los objetos enlazados se inserten también. Si en el panel de operador de destino hay objetos del mismo nombre, para cada objeto debe definir si es preciso sobrescribir el objeto existente.

Consulte también

Copiar imágenes (Página 9239)

Copiar recetas en un mismo panel de operador (Página 9240)

Objetos enlazados copiados automáticamente

Copiar objetos enlazados

La tabla siguiente muestra los objetos en los que los objetos enlazados también se pegan automáticamente en el pegado simple.

Objeto	Objeto enlazado
Imagen	Plantilla
Campo E/S simbólico	Lista de textos
Campo E/S gráfico	Lista de gráficos
Visor de gráficos	Gráfico
Variable	Aviso
	Ciclo
Elemento de receta	Lista de textos
Planificador de tareas	Disparador

Consulte también

Copiar imágenes (Página 9239)

Copiar recetas en un mismo panel de operador (Página 9240)

Drag&Drop desde la vista detallada

Introducción

La eficacia de su configuración puede aumentar con unos pasos muy sencillos. A continuación, verá algunos ejemplos para realizar una configuración eficaz.

Insertar objetos en una imagen a partir de la vista detallada

Es posible arrastrar objetos de diversos editores a otros editores desde la vista detallada.

Insertar campo E/S simbólico

1. Abra una imagen.
2. En en el árbol del proyecto haga clic en el editor "Listas de textos y gráficos". En la vista detallada aparecerán todas las listas de textos y gráficos disponibles.
3. En la vista detallada haga clic en una lista de textos, por ejemplo "Textlist1".
4. Con la función de Drag&Drop arrastre una lista de textos desde la vista detallada hasta una imagen. Se ha creado un campo E/S simbólico que está conectado con la lista de textos "Textlist1".

Insertar campo E/S gráfico

1. Abra una imagen.
2. En en el árbol del proyecto haga clic en el editor "Listas de textos y gráficos". En la vista detallada aparecerán todas las listas de textos y gráficos disponibles.
3. En la vista detallada haga clic en una lista de gráficos, por ejemplo "Graficlist1".
4. Con la función de Drag&Drop arrastre una lista de gráficos desde la vista detallada hasta una imagen. Se ha creado un campo E/S gráfico que está conectado con la lista de gráficos "Graficlist1".

Insertar campo E/S

1. Abra una imagen.
2. En el árbol del proyecto haga clic en el editor "Variables HMI". En la vista detallada aparecerán todas las variables HMI disponibles.
3. En la vista detallada haga clic en una variable HMI, por ejemplo "Tag1".
4. Con la función de Drag&Drop arrastre una variable HMI desde la vista detallada hasta una imagen. Se ha creado un campo E/S que está conectado con la variable HMI "Tag1".

10.9.8.3 Copia entre distinas versiones de RT y ES

Introducción

Es posible copiar e insertar datos de proyecto, p. ej. imágenes, objetos o variables, entre proyectos con diferentes versiones de WinCC.

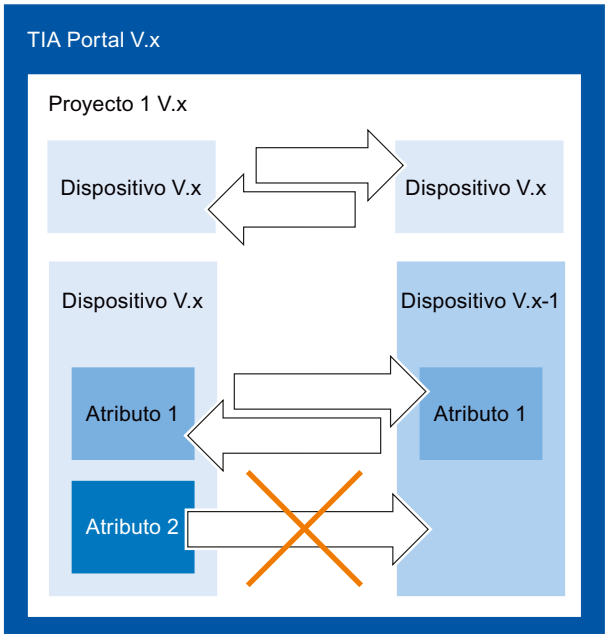
Si hace copias entre diferentes versiones de WinCC, se conservan todas las configuraciones que estén soportadas también en la versión de destino. Las configuraciones no soportadas en la versión de destino se marcan en color como no válidas.

Copiar entre paneles de operador de distintas versiones de dispositivos

WinCC soporta todas las configuraciones procedentes de una versión anterior de WinCC.

Al copiar entre versiones de ES distintas, se aplica lo siguiente:

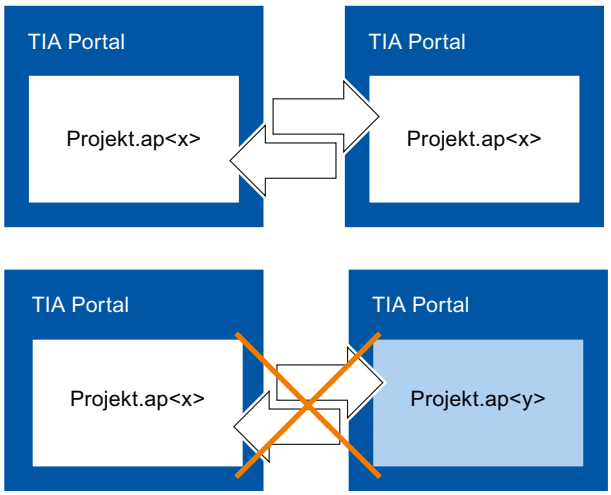
- Se mantienen todas las configuraciones soportadas en las correspondientes versiones de RT.
- En el caso de las configuraciones soportadas exclusivamente por la versión de WinCC del proyecto de destino, se definen ajustes estándar.
- Las configuraciones no admitidas por la correspondiente versión de RT se marcan en color como no válidas o no se muestran.
  Si se vuelve a copiar un objeto al panel de operador de origen sin modificaciones, todas las propiedades y ajustes definidos en el panel de operador de origen volverán a estar activos.



- El panel de operador debe ser válido para la versión actual de Runtime.

Copiar entre versiones de ES distintas

Para copiar entre dos proyectos TIA, abra una segunda instancia de su TIA Portal. Solo podrá copiar entre proyectos creados con la misma versión de ES. La versión de ES de un proyecto se identifica por la extensión de archivo *.ap<número_versión>.



10.9.9 Utilizar la compatibilidad de versiones de WinCC

10.9.9.1 Principios básicos de la compatibilidad de versiones

Introducción

Los proyectos existentes se editan como sigue con WinCC:

- Editar, compilar y cargar los proyectos existentes con el volumen de funciones de la versión anterior de WinCC. Estos proyectos se pueden seguir editando a continuación con la versión anterior de WinCC.
- Actualizar los proyectos existentes y utilizar las funciones de la versión actual de WinCC.

Nota

Funciones de WinCC

Si edita un proyecto en una versión anterior de WinCC, sólo dispondrá de las funciones y paneles de operador de esta versión anterior de WinCC.

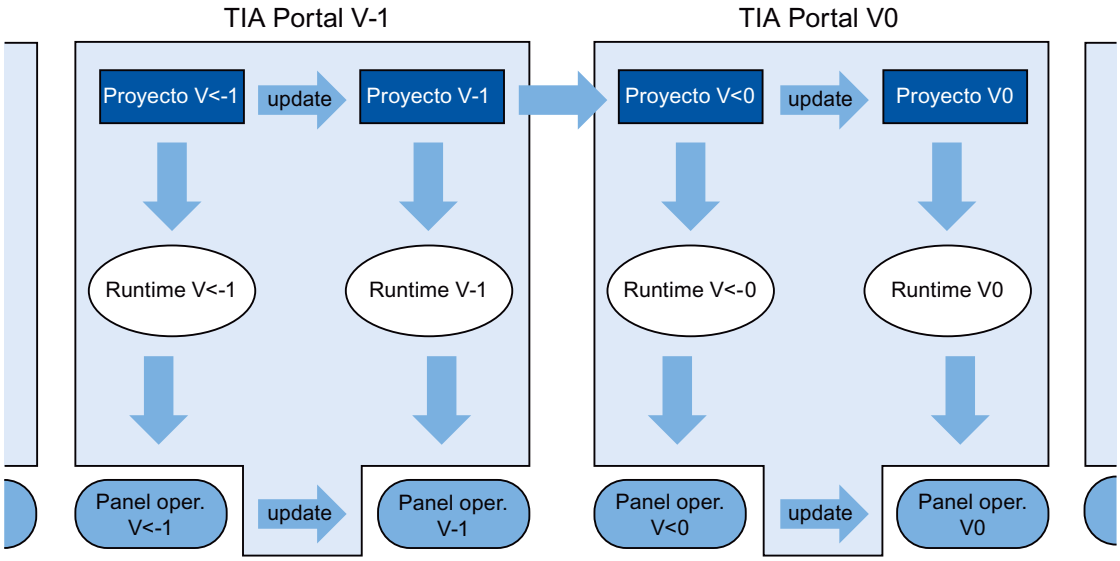
Versiones de WinCC

En WinCC se trabaja con distintos tipos de versiones:

- Versión de WinCC
  La versión de WinCC instalada en el PC de configuración, p. ej., WinCC V12.
- Versión de proyecto
  Los proyectos se crean con la versión de WinCC instalada en el PC de configuración. Si edita p. ej. un proyecto de WinCC de una versión anterior con la versión actual de WinCC, se mostrará en el árbol de proyecto el identificador de la versión detrás del nombre del proyecto.
- Versión de runtime
  En WinCC puede configurar paneles de operador con distintas versiones de runtime. Solo es necesario definir una vez la versión de runtime para un panel de operador. La versión del panel debe concordar con la versión de runtime.
- Versiones de panel
  Dependiendo del panel de operador utilizado, la imagen se compone del sistema operativo y / o del software runtime. WinCC proporciona para cada panel de operador diferentes imágenes, que se cargan en él según sea necesario dependiendo de la configuración. La versión de panel se corresponde con una determinada imagen. La versión de panel debe concordar con la configuración.

Compatibilidad con versiones de WinCC, versiones de runtime y versiones de panel

La siguiente figura muestra la interacción de las versiones en el TIA Portal:



Creación de proyectos

Si crea un proyecto nuevo en WinCC, ábralo y edítelo con la versión de WinCC con la que lo ha creado.

Guardar

Para volver a guardar un proyecto de una versión anterior de WinCC para esta versión, proceda de la forma habitual. En cuanto actualice manualmente el proyecto a su versión de WinCC, ya no podrá abrirlo en la versión anterior de WinCC.

Para almacenar un proyecto de una versión anterior de WinCC en la versión actual, actualice el proyecto a su versión de WinCC. A partir de entonces, este proyecto ya no podrá editarse con la versión anterior de WinCC.

Compilación, simulación y carga

Si utiliza un proyecto en una versión anterior de WinCC, con su versión actual podrá crear datos de Runtime de esta versión anterior. De este modo también podrá cargar paneles de operador que ya no sean compatibles con su versión de WinCC.

Copia dentro de proyectos con diferentes versiones de WinCC

Si también dispone de objetos y configuraciones en la versión de destino, cópielos según sea necesario mediante el portapapeles o utilizando el método de arrastrar y soltar.

Abrir, editar y guardar proyectos de una versión anterior de WinCC

Se pueden abrir y editar proyectos de versiones anteriores de WinCC según se necesite. Al hacerlo, únicamente se utilizan las funciones de esa versión anterior de WinCC. Tras su edición, el proyecto se puede volver a modificar y guardar en la versión anterior de WinCC.

Compilar, cargar y simular proyectos con versiones anteriores de WinCC

Se pueden compilar, cargar y simular proyectos de versiones anteriores de WinCC según se necesite. Para ello, su actual versión de WinCC proporciona los runtimes y las versiones de panel de la correspondiente versión de WinCC.

10.9.9.2 Edición de proyectos de una versión anterior de WinCC

Introducción

WinCC ofrece la posibilidad de editar proyectos de una versión anterior de WinCC. Mientras edita un proyecto de una versión anterior de WinCC, sólo dispondrá de las funciones de esta versión. Para utilizar las funciones de su versión actual de WinCC para este proyecto, actualice el proyecto a su versión de WinCC.

Nota

Cuando actualice un proyecto a su versión de WinCC, ya no podrá abrirlo ni editarlo en la versión anterior de WinCC.

Requisitos

- Se ha creado un proyecto de una versión anterior de WinCC.
- En el PC de configuración está instalada la versión actual de WinCC.

Procedimiento

Para editar un proyecto de una versión anterior de WinCC, proceda como sigue:

1. Abra el proyecto.
2. Edite el proyecto con las funciones de la versión anterior de WinCC.
3. Guarde el proyecto.
4. Compile el proyecto.
5. Cargue y simule el proyecto.
6. Abra el proyecto en caso necesario con la versión anterior de WinCC y siga editándolo allí.

Resultado

Los datos de proyecto modificados pueden seguirse editando en otro PC de configuración con la versión anterior de WinCC. El proyecto de Runtime se ha creado y cargado en la correspondiente versión de Runtime.

10.9.9.3 Actualización del proyecto

Introducción

Si la versión del proyecto es más antigua que la versión de WinCC, en el árbol de proyecto se mostrará el identificador de la versión. Su versión de WinCC contiene también la versión anterior, con la que pueden editarse los proyectos si es necesario. Para utilizar las funciones y opciones de su versión de WinCC en un proyecto, actualice el proyecto a su versión de WinCC y, a continuación, modifique sus versiones de dispositivos o de runtime como corresponda a la nueva versión del proyecto.

Nota

Compatibilidad de versiones de WinCC

Cuando actualice un proyecto a su versión de WinCC, ya no podrá editarlo en la versión anterior.

Requisitos

- La versión del proyecto es predecesora de su versión de WinCC.
- Usted dispone de derechos de escritura para su unidad de proyecto.
- La capacidad de memoria de su unidad de proyecto es suficiente para otro proyecto más de este tamaño.

Procedimiento

Para actualizar un proyecto a su versión de WinCC, proceda como sigue:

1. Seleccione el proyecto en el árbol del proyecto.
2. Elija el comando "Actualizar proyecto" del menú contextual del proyecto.
   Se abre un cuadro de diálogo.
3. Haga clic en "Confirmar".
   El proyecto se cierra y se muestra un indicador de progreso.

Cuando el proyecto está actualizado, aparece un mensaje.

Resultado

- En la unidad de proyecto se ha almacenado el proyecto en la versión anterior de WinCC con la correspondiente extensión de archivo.
- En la unidad de proyecto se muestra el proyecto con la versión actual de WinCC y la correspondiente extensión de archivo.
- El proyecto se muestra en el árbol de proyecto sin el identificador de versión de WinCC.

Para utilizar las funciones de la nueva versión de WinCC, modifique en el paso siguiente sus versiones de dispositivos o de runtime como corresponda a la nueva versión del proyecto.

10.9.9.4 Actualizar una librería global

Introducción

Para poder editar objetos de una librería global en un proyecto, la librería global debe tener la misma versión de producto que el proyecto. Puede actualizar toda librería global de una versión anterior a la versión actual del producto. Al abrir la librería global se muestra el requerimiento correspondiente.

Requisitos

- La versión de la librería global es predecesora de su versión de WinCC.
- Usted dispone de derechos de escritura en la unidad del proyecto.
- Todos los tipos de la librería están habilitados.

Nota

Actualización de una librería de usuario

Si desea utilizar una librería de usuario de una versión anterior de WinCC, deberá actualizarla. Asegúrese de que todos los tipos de la librería estén habilitados. Para la actualización, ninguno de los tipos contenidos en la librería debe tener el estado "en proceso".

Procedimiento

Para actualizar una librería global del TIA Portal V12.x o anterior, haga lo siguiente:

1. Abra la librería global.
   Se abrirá el cuadro de diálogo "Actualizar una librería global".
2. Haga clic en "Aceptar".

Se creará y actualizará una copia de la librería global. Se añadirá "_V13" al nombre de la copia de la librería global. Se abre la librería global.

Resultado

La librería global se ha guardado con la correspondiente extensión de archivo.

Consulte también

Principios básicos de la compatibilidad de versiones (Página 9244)

Administrar librerías (Página 9140)

10.9.9.5 Cambio entre distintas versiones de dispositivos

Selección de la versión de panel

Si se configura un nuevo panel de operador, WinCC selecciona automáticamente la versión de panel más reciente.

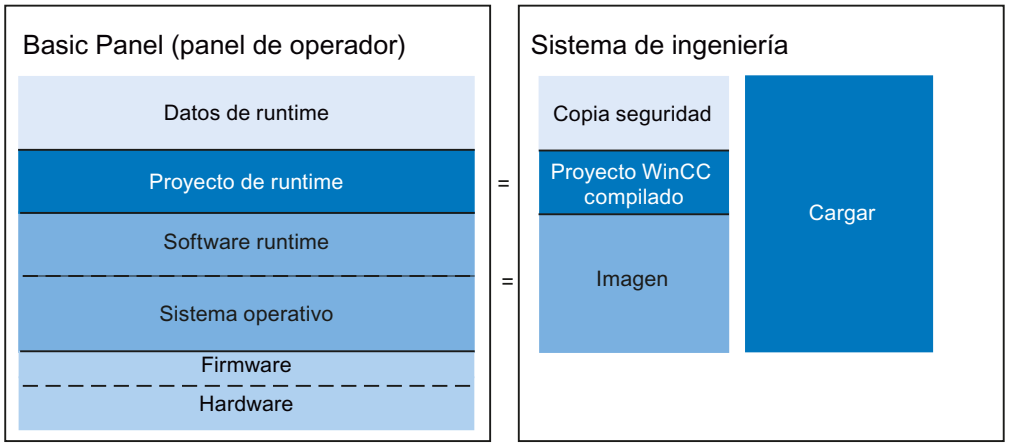
Si en un panel de operador desea utilizar una versión de panel distinta de la configurada en WinCC, transfiera una imagen al panel de operador. WinCC proporciona las imágenes requeridas de los paneles de operador soportados.

Encontrará información relativa a las versiones de panel utilizadas en WinCC en la sección de preguntas frecuentes de Internet, en el artículo con el n.º de identificación (ID) 21742389.

ATENCIÓN
El cambio de versión del panel borra todos los datos del panel de operador. Al cambiar la versión del panel se borran datos en el sistema de destino. Por tanto, haga una copia de seguridad de los datos de runtime existentes y cambie después la versión del panel.

Componentes del panel de operador

La siguiente figura muestra los componentes de software de un panel de operador:



10.9.9.6 Cambio de versión del dispositivo

Introducción

Dependiendo de la versión de runtime deseada, seleccione la versión del panel adecuada para su configuración.

Nota

Selección de versiones de panel

La selección ofrecida de versiones de panel depende de la versión del proyecto.

Requisitos

- Se ha creado y abierto un proyecto.
- El proyecto contiene un panel de operador.

Procedimiento

Para modificar la versión de panel, proceda del siguiente modo:

1. Haga doble clic en "Dispositivos y redes" del árbol del proyecto.
   Se abre el editor.
2. Seleccione el panel de operador deseado en la vista de dispositivos.
3. Seleccione la entrada "Modificar dispositivo/versión" en el menú contextual del panel de operador.
   Se abre un cuadro de diálogo.
4. Seleccione el panel de operador deseado.

5. Elija en "Versión" la versión de panel deseada.
6. Confirme la selección con "Aceptar".

Resultado

Ha modificado la versión de panel del dispositivo en el proyecto WinCC.

ATENCIÓN
El cambio de versión del panel borra todos los datos del panel de operador durante la carga.
Al cambiar la versión del panel y compilar y cargar el proyecto, se borrarán datos en el panel de operador. Por tanto, guarde los datos de runtime existentes antes de la carga.

10.9.10 Mostrar datos de una tarjeta de memoria

10.9.10.1 Principios básicos

Introducción

WinCC le ofrece la posibilidad de mostrar datos de su tarjeta de memoria. Se pueden utilizar tanto tarjetas de memoria del panel de operador como de la CPU.

Existen las posibilidades siguientes:

Mostrar copia de seguridad (Página 9254)

Cambiar nombre y borrar copia de seguridad (Página 9255)

Auto-Hotspot

Auto-Hotspot

Auto-Hotspot

Consulte también

Mostrar copia de seguridad (Página 9254)

Cambiar nombre y borrar copia de seguridad (Página 9255)

10.9.10.2 Trabajo con copias de seguridad

Mostrar copia de seguridad

Introducción

Si almacena en una tarjeta de memoria la copia de seguridad de un Basic Panel, podrá visualizar la copia de seguridad en el TIA Portal.

Requisitos

- WinCC está instalado.
- Existe una tarjeta de memoria con una copia de seguridad.
- El lector de tarjetas está conectado con el PC de configuración.
- La vista del proyecto está abierta.

Copia de seguridad en la tarjeta de memoria del lector de tarjetas

1. Coloque la tarjeta de memoria en el lector de tarjetas.
2. Abra el "SIMATIC Card Reader" en el árbol del proyecto.
3. Seleccione la unidad del lector de tarjetas.
   Se muestra la carpeta "Online Card Data".
4. Abra la carpeta "Online Card Data".
5. Haga clic en la copia de seguridad y abra el menú contextual.
6. Elija "Propiedades".

Copia de seguridad en la tarjeta de memoria del PLC

Si se ha almacenado la copia de seguridad en la tarjeta de memoria del PLC, proceda como sigue:

1. Conecte el PLC con el PC de configuración.
2. Haga clic en el PLC dentro del árbol del proyecto.
3. En el menú contextual, elija "Establecer conexión online".
   Se establece una conexión con el PLC.
   Cuando el PLC esté conectado, aparecerá la carpeta "Online Card Data".

4. Abra la carpeta "Online Card Data".

Nota

Acceso a un PLC protegido por contraseña

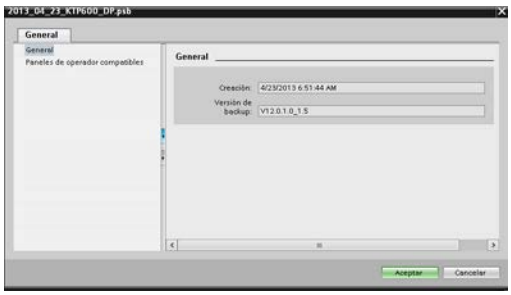
Si accede a un PLC protegido por contraseña, se le pedirá que introduzca la contraseña.

Para poder visualizar los datos de la tarjeta de memoria, necesita al menos acceso de lectura.

5. Haga clic en la copia de seguridad y abra el menú contextual.
6. Elija "Propiedades".

Resultado

Se muestran las propiedades de la copia de seguridad en un cuadro de diálogo aparte.



Consulte también

Principios básicos (Página 9251)

Cambiar nombre y borrar copia de seguridad (Página 9255)

Cambiar nombre y borrar copia de seguridad

Introducción

Es posible cambiar el nombre y borrar la copia de seguridad de una tarjeta de memoria en el árbol del proyecto del TIA Portal.

Requisitos

- WinCC está instalado.
- El lector de tarjetas está conectado con el PC de configuración.
  O bien el PLC está conectado online con el PC de configuración.
- Existe una tarjeta de memoria con una copia de seguridad.

- La vista del proyecto está abierta.
- La copia de seguridad se muestra en el árbol del proyecto.

Nota

Acceso a un PLC protegido por contraseña

Si accede a un PLC protegido por contraseña, se le pedirá que introduzca la contraseña.

Para borrar o cambiar el nombre de los datos de la tarjeta de memoria se necesita acceso de escritura.

Procedimiento

1. Haga clic en la copia de seguridad dentro del árbol del proyecto.
2. Abra el menú contextual.
3. Para cambiar el nombre del archivo, seleccione "Cambiar nombre".
4. Introduzca un nuevo nombre.
5. Para borrar el archivo, seleccione "Borrar".

Resultado

Se cambia el nombre o se borra el archivo de copia de seguridad.

Consulte también

Principios básicos (Página 9251)

Mostrar copia de seguridad (Página 9252)

10.10 Compilar y cargar

10.10.1 Compilar y cargar proyectos

10.10.1.1 Resumen de cómo compilar y cargar proyectos

Resumen

Ya cuando se configura un proyecto en WinCC, dicho proyecto se va compilando en el fondo de forma continua. De esta manera se reduce considerablemente el tiempo necesario para realizar la compilación definitiva. Debido al inicio de la compilación se genera un archivo que se puede ejecutar en el panel de operador correspondiente.

Si se producen errores durante la compilación, WinCC le ayuda a encontrarlos y eliminarlos.

Tras eliminar los errores, el proyecto compilado se carga en los paneles de operador en los que debe utilizarse.

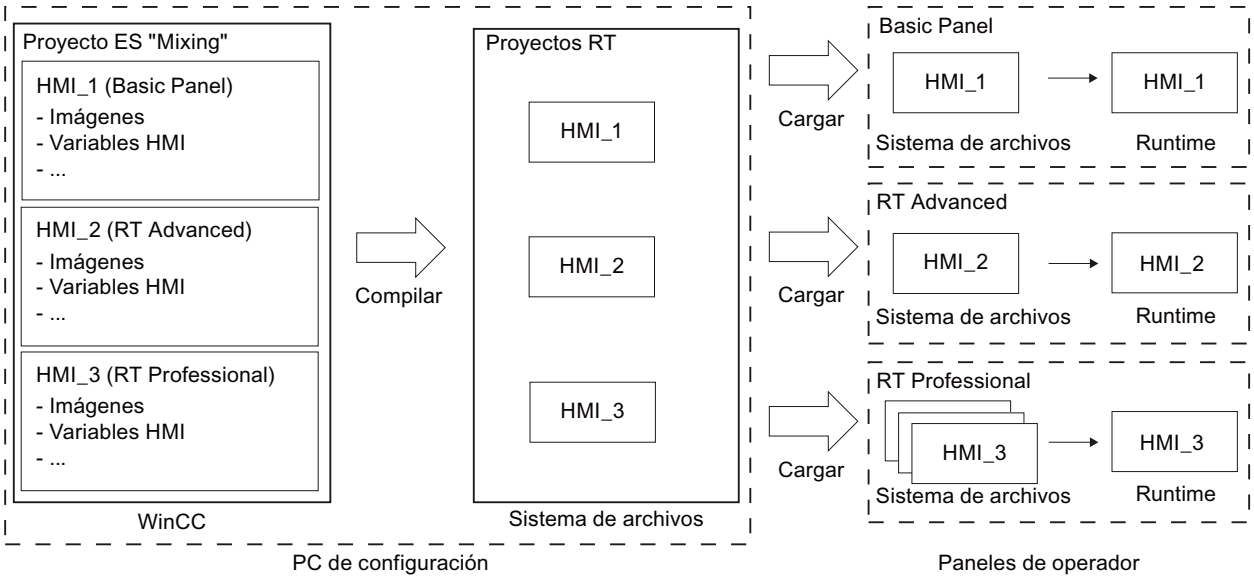
Si en el proyecto se utilizan variables HMI que están vinculadas a variables de controlador, antes de compilar el panel de operador HMI compile todos los bloques S7 modificados con el comando "Compilar > Software" del menú contextual.

Explicación de conceptos

En el contexto de las tareas de compilación y carga el concepto "proyecto" es polisémico. Por un lado "proyecto" designa el proyecto de WinCC del PC de configuración. Por otro lado, "proyecto" se utiliza para aludir al proyecto de runtime que se crea mediante compilación a partir de los datos de configuración de un panel de operador y que se carga a continuación en el panel de operador.

- Proyecto de WinCC: Contiene los datos de configuración de uno o varios paneles de operador
- Proyecto runtime: Contiene los datos de configuración compilados de un panel de operador

La imagen siguiente muestra la relación entre el proyecto WinCC y el proyecto runtime tomando como ejemplo el proceso de "compilación y carga":



Versión de runtime

La versión de Runtime depende de la imagen del panel de operador configurado. La versión de Runtime del proyecto compilado se muestra en la ventana de inspección bajo "Información".

10.10.1.2 Compilar un proyecto

Introducción

Ya cuando se configura un proyecto en WinCC, los cambios realizados en el proyecto se van compilando en el fondo de forma continua. Si se carga un proyecto, este se compila de forma automática. De este modo se garantiza que siempre se cargue la versión más reciente del proyecto.

Durante la compilación, WinCC comprueba la coherencia del proyecto. En la ventana de inspección se listan todos los puntos defectuosos del proyecto. Pulsando la entrada de la ventana de inspección se pasa directamente a la causa del error. Los errores encontrados se deben verificar y eliminar.

Alcance de la compilación

Nada más comenzar la configuración de un panel de operador, los datos de configuración se compilan de forma continua en segundo plano. Si un proyecto se compila de forma manual, sólo se compilan en segundo plano los cambios realizados en la configuración desde la última compilación.

Por ejemplo, para comprobar la coherencia de los datos configurados, es posible iniciar manualmente una compilación completa del proyecto.

Requisitos

- Hay un proyecto abierto.

Procedimiento

Para compilar un proyecto, proceda del siguiente modo:

1. Si desea compilar varios paneles de operador al mismo tiempo, seleccione todos los paneles de operador deseados con la opción de selección múltiple de la navegación del proyecto.
2. Compile el proyecto:
   - Para compilar únicamente modificaciones en el proyecto, seleccione el comando "Compilar > Software (solo modificaciones)" en el menú contextual del panel de operador.
   - Para compilar el proyecto por completo, seleccione en el menú contextual el comando "Compilar > Software (compilar bloques por completo)".

Resultado

Se compilan los datos de configuración de todos los paneles de operador seleccionados. Si se producen errores durante la compilación, aparecerán en la ventana de inspección.

10.10.1.3 Cargar proyectos

Resumen de cómo cargar proyectos

Resumen

Cuando se transfiere un proyecto a uno o varios paneles de operador, la parte del proyecto que se ha modificado desde la última transferencia se compila automáticamente antes de la carga. De este modo se garantiza que siempre se transfiera la versión actual del proyecto.

Carga de un proyecto en un panel de operador

Antes de realizar la carga se ejecutan los siguientes pasos:

1. Los ajustes de la carga se verifican. Si un proyecto se carga por 1.ª vez en un panel de operador, se abre automáticamente el cuadro de diálogo "Carga avanzada". En este cuadro de diálogo se define el protocolo y la interfaz o la ruta de destino del proyecto, independientemente del runtime utilizado en el panel de operador.
   El cuadro de diálogo "Carga avanzada" puede abrirse en cualquier momento a través del comando de menú "Online > Carga avanzada en dispositivo".
   Se abre el cuadro de diálogo "Vista preliminar: Cargar".
2. El proyecto se compila. Las advertencias y los errores que se producen durante la compilación se visualizan en la ventana de inspección y en el cuadro de diálogo "Vista preliminar: Cargar",
3. que muestra la información siguiente, separada por paneles de operador:
   – Los diferentes pasos de la carga
   – Si la imagen del panel de operador de destino no concuerda con la imagen de la configuración, se consulta si se desea cambiar la imagen ahora.

ATENCIÓN
El cambio de la imagen borra todos los datos del panel de operador. Al cambiar la imagen del panel de operador se borran datos del sistema de destino. Por tanto, en caso necesario, realice primero una copia de seguridad de los siguientes datos: • Administración de usuarios • Recetas

   – Los ajustes predeterminados que serán efectivos al cargar. Adapte estos ajustes para el proceso de carga en función de sus necesidades.
   – Advertencias aparecidas (opcional). El proyecto se puede cargar a pesar de las "Advertencias". No obstante, es posible que la funcionalidad en runtime quede restringida.
   – Errores aparecidos (opcional). El proyecto no se puede cargar. Elimine primero los errores y cargue de nuevo el proyecto.
     Si hace doble clic en el error visualizado dentro de la ventana de inspección, WinCC abre la configuración defectuosa en el editor correspondiente. Corrija los errores y cargue de nuevo el proyecto.

Si en el proyecto se utilizan variables HMI que están vinculadas a variables de controlador, antes de compilar el panel de operador HMI compile todos los bloques S7 modificados con el comando "Compilar > Software" del menú contextual.

Carga a través del routing S7

Los ajustes del routing S7 se configuran en el controlador correspondiente en el editor "Dispositivos y redes". Los ajustes dependen del dispositivo configurado.

El routing S7 soporta los protocolos siguientes:

- MPI/PROFIBUS
- Ethernet

Transferencia de Addons de runtime

Los proyectos pueden contener Addons de runtime en forma de controles o CSP (Communication Support Packages). Estos Addons de runtime se transfieren automáticamente con el proyecto.

Cargar un proyecto

Introducción

Para ejecutar un proyecto en un panel de operador, cargue el proyecto en este último. Al realizar la carga, determine en particular si los datos existentes en el panel de operador se sobrescriben como "Administración de usuarios" o "Datos de receta".

Si el panel de operador soporta PROFINET, el nombre del panel de operador incluido en el árbol del proyecto se utiliza como nombre de dispositivo para la comunicación PROFINET. El nombre se escribe al realizar el proceso de escritura en el panel de operador. Si el panel de operador contenía ya un nombre de dispositivo para la comunicación PROFINET, este se sobrescribe.

Por lo general en un panel de operador solo puede haber un proyecto en runtime. De manera predeterminada un panel de operador está configurado de tal modo que el runtime en curso finaliza automáticamente al realizar la carga. De lo contrario, es preciso finalizar el runtime en el panel de operador de forma manual.

Si la imagen del panel de operador de destino no concuerda con la imagen de la configuración, se consulta si se desea cambiar la imagen ahora.

ATENCIÓN
El cambio de la imagen borra todos los datos del panel de operador. Al cambiar la imagen del panel de operador se borran datos del sistema de destino. Por tanto, en caso necesario, realice primero una copia de seguridad de los siguientes datos: • Administración de usuarios • Recetas

Para obtener más información sobre la configuración de la transferencia, consulte la documentación del panel de operador que esté utilizando.

Nota

Finalización automática de runtime

Si en el panel de operador está activada la transferencia automática y en el PC de configuración se inicia una transferencia, el proyecto activo se cierra automáticamente.

El panel de operador cambia entonces automáticamente al modo de operación "Transfer".

Desactive la transferencia automática tras la fase de puesta en servicio para que el panel de operador no pase por error al modo de transferencia.

El modo de transferencia puede provocar reacciones imprevistas en la instalación.

Para bloquear el acceso a los ajustes de transferencia, y evitar así modificaciones no autorizadas, asigne una contraseña en el Control Panel.

Requisitos

- En el proyecto se ha creado un panel de operador.
- El panel de operador está conectado con el PC de configuración.
- En el panel de operador se ha ajustado el modo de transferencia.

Procedimiento

Para cargar un proyecto, proceda del siguiente modo:

1. Si desea cargar un proyecto en varios paneles de operador al mismo tiempo, seleccione en la navegación del proyecto todos los paneles de operador que desee con la opción de selección múltiple.
2. En el menú contextual de un panel de operador, elija el comando "Cargar en dispositivo > Software".
3. Cuando se abre el cuadro de diálogo "Carga avanzada" configure los "ajustes de carga". Asegúrese de que los ajustes de carga coinciden con los ajustes de transferencia del panel de operador.
   - Seleccione el protocolo que vaya a utilizar, como puede ser Ethernet o HTTP.
   - Configure los parámetros de interfaz correspondientes en el PC de configuración.
   - En caso necesario ajuste las opciones específicas de la interfaz o del protocolo en el panel de operador.
   - Haga clic en "Cargar".

   El cuadro de diálogo "Carga avanzada" puede abrirse en cualquier momento a través del comando de menú "Online > Carga avanzada en dispositivo".
   Se abre el cuadro de diálogo "Vista preliminar: Cargar". El proyecto se compila simultáneamente. El resultado se muestra en el cuadro de diálogo "Vista preliminar: Cargar".

4. Compruebe los ajustes predeterminados que aparecen y modifíquelos en caso necesario.
5. Haga clic en "Cargar".

Resultado

El proyecto se cargará en los paneles de operador seleccionados. Si ya existe un proyecto, éste se sustituye. Dependiendo de los ajustes efectuados en el cuadro de diálogo "Vista preliminar: Carga", se sustituyen los datos de la administración de usuarios y/o de las recetas.

Durante la carga puede ver qué archivos se están transfiriendo.

Si se producen errores o advertencias durante la carga, los avisos correspondientes se emiten en la ventana de inspección, en "Información > Cargar".

Una vez realizada la carga correctamente, el proyecto puede ejecutarse en el panel de operador.

Nota

Si se interrumpe la transferencia, WinCC se asegura automáticamente de que no se pierdan datos y de que los datos existentes en el panel de operador solo se borren una vez completada la transferencia.

Consulte también

Guardar y restablecer los datos del panel de operador (Página 9273)

Actualizar el sistema operativo del panel de operador (Página 9275)

Mensajes de error al cargar proyectos (Página 9277)

Adaptar el proyecto a otro panel de operador (Página 9278)

Establecer una conexión con el panel de operador (Página 9280)

10.10.1.4 Inicio de runtime

Iniciar el runtime en el panel de operador

Introducción

El proyecto se puede iniciar en runtime en cuanto se ha cargado en el panel de operador. El proyecto se guarda en el panel de operador en un archivo con la extensión siguiente:

- Basic Panels así como OP 73, OP 77A y TP 177A: "*.srt"

Al iniciar el proyecto en runtime, se hacen efectivos los ajustes establecidos para el proyecto en la "Configuración de runtime" del panel de operador.

Los programas para iniciar proyectos en el panel de operador se encuentran en el directorio de instalación de runtime.

Nota

Finalización automática de runtime

Si en el panel de operador está activada la transferencia automática y en el PC de configuración se inicia una transferencia, el proyecto activo se cierra automáticamente.

El panel de operador cambia entonces automáticamente al modo de operación "Transfer".

Una vez concluida la fase de puesta en servicio, desactive la transferencia automática para que el panel de operador no pase por error al modo de transferencia.

El modo de transferencia puede provocar reacciones imprevistas en la instalación.

Para bloquear el acceso a los ajustes de transferencia, y evitar así modificaciones no autorizadas, asigne una contraseña en el Control Panel.

Requisitos

El panel de operador tiene instalado WinCC Runtime.

Procedimiento

En un dispositivo se inicia la aplicación "RT Loader". En función de la configuración, el proyecto cargado se inicia automáticamente tras un tiempo de retardo.

Si el proyecto no se inicia automáticamente:

1. Para seleccionar el archivo de proyecto, haga clic en "Configuración" e introduzca la ruta del archivo de proyecto en "Archivo de configuración".
2. Haga clic en "Aceptar" y a continuación en "Inicio".

10.10.2 Simular proyectos

10.10.2.1 Principios básicos de la simulación

Introducción

Con el simulador se prueba el comportamiento de la configuración en el PC de configuración. De este modo se detectan errores de configuración lógicos de antemano y antes de pasar al modo productivo.

El simulador se puede iniciar las siguientes maneras:

- En el menú contextual del panel de operador o de una imagen: "Iniciar simulación"
- Comando de menú "Online > Simulación > [Iniciar|Con simulador de variables|Con depurador de scripts]"
- En la vista de portal, en "Visualización > Simular dispositivo"

Requisitos

En el PC de configuración está instalado el componente de simulación/runtime.

Campo de aplicación

Con el simulador pueden probarse las funciones siguientes del sistema de manejo y visualización, p. ej.:

- Comprobación de niveles de valores límite y emisiones de avisos
- Continuidad de alarmas
- Simulaciones de alarmas configuradas
- Advertencias configuradas
- Avisos de error configurados
- Control de indicadores de estado
- Interconexión y diseño de imágenes

Consulte también

Simular un proyecto (Página 9266)

10.10.2.2 Simular un proyecto

Introducción

El proyecto se simula de uno de los dos modos siguientes:

- Sin un controlador conectado
  Los valores de los punteros de área y las variables se modifican en un simulador de variables que lee WinCC Runtime durante la simulación.
- Con un controlador conectado sin proceso activo
  El proyecto se simula ejecutándolo directamente en el runtime. Las variables y los punteros de área se activan. De este modo, en runtime se consigue una simulación auténtica del panel de operador configurado.

Nota

Restricciones en la simulación

Las funciones de sistema siguientes no pueden simularse:

- CalibrarPantallaTáctil

No es posible simular el Media Player. En la ventana de simulación aparece una imagen estática en lugar del Media Player.

En los paneles de operador con Windows CE no es posible acceder a los archivos a través de scripts.

Requisitos

- Simulación sin un controlador conectado: Las variables están creadas
- Simulación con un controlador conectado sin proceso activo: Se ha creado un proyecto con variables y punteros de área

Procedimiento

Para simular un proyecto a través del simulador de variables, proceda del siguiente modo:

1. Abra el proyecto en el PC de configuración.
2. Seleccione el comando de menú "Online > Simulación > Con simulador de variables".
   Al simular el proyecto por primera vez, el simulador se inicia con una nueva tabla vacía. Al mismo tiempo, el proyecto se abre en runtime.
   Para cambiar entre el simulador de variables y runtime se utiliza la combinación de teclas <Alt+Tab>.
3. Para simular un valor de proceso, seleccione la "variable" deseada en el simulador de variables.
   La selección contiene todas las variables configuradas. Se puede simular un máximo de 300 variables a la vez.
4. Seleccione el tipo de simulación deseado en la columna "Simulación".
5. En las columnas correspondientes, modifique los valores de las variables y los punteros de área.

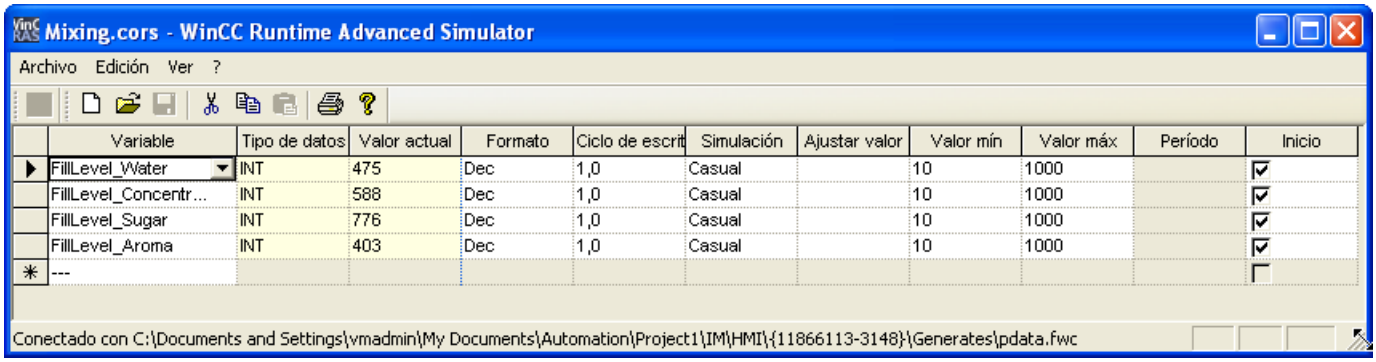
6. Para iniciar la simulación para dicha variable, active la casilla de verificación "Inicio".
7. Para guardar la simulación, seleccione el comando de menú "Archivo > Guardar" e introduzca un nombre correspondiente, p. ej. "Mixing".
   El archivo recibe la extensión "*.cors".

Resultado

Los valores de proceso se simulan en runtime. En función del tipo de simulación los valores de variable, por ejemplo, se crean de forma aleatoria o se incrementan.

Para especificar valores de variables, modifique el tipo de simulación en "<Visualización>" e introduzca el valor deseado en "Ajustar valor".

La siguiente imagen muestra un simulador de variables con cuatro variables cuyos valores se determinan de forma aleatoria en el rango de valores de 10 a 1000:



Variable	Tipo de datos	Valor actual	Formato	Ciclo de escrit	Simulación	Ajustar valor	Valor mín	Valor máx	Período	Inicio
FillLevel_Water	INT	475	Dec	1,0	Casual		10	1000		☑
FillLevel_Concentr...	INT	588	Dec	1,0	Casual		10	1000		☑
FillLevel_Sugar	INT	776	Dec	1,0	Casual		10	1000		☑
FillLevel_Aroma	INT	403	Dec	1,0	Casual		10	1000		☑
---										☐

Administrar los datos de simulación

Si se han guardado los datos de una simulación anterior, se puede abrir el archivo posteriormente y volver a simular el proyecto. Para ello es imprescindible que las variables y los punteros de área contenidos en el simulador de variables sigan estando en el proyecto.

Para abrir un archivo de simulación, proceda del siguiente modo:

1. Seleccione el comando de menú "Online > Simular runtime > Con simulador de variables".
2. Seleccione el comando de menú "Archivo > Abrir" en el simulador de variables.
3. Seleccione el archivo de simulación correspondiente y haga clic en "Abrir".
   El simulador aplica los datos guardados.

Activar y desactivar variables

Para que el paso de la configuración offline a la configuración online se produzca de forma óptima, inicie y detenga la simulación de cada variable individualmente. Para ello, active "Inicio" en la fila correspondiente.

Cuando una variable está activada, los valores de la simulación se calculan y transfieren al simulador de WinCC.

Borrar variable

Para borrar una variable del simulador de variables, proceda del siguiente modo:

1. Seleccione la celda con el nombre de la variable.
2. Seleccione el comando de menú "Edición > Cortar".
   La variable se elimina de la tabla.

Consulte también

Principios básicos de la simulación (Página 9262)

Trabajar con el simulador de variables (Página 9268)

10.10.2.3 Trabajar con el simulador de variables

Explicaciones del simulador de variables

El simulador de variables contiene las columnas siguientes:

Columna	Descripción
Variable	Define la variable para la simulación.
Tipo de datos	Muestra el tipo de datos de la variable seleccionada.
Valor act.	Muestra el valor simulado de las variables introducidas.
Formato	Define el formato seleccionado en el que se simularán los valores de las variables: • Decimal (1, 2, 3, 4, ...) • Hexadecimal (03CE, 01F3, ...) • Binario (0 y 1)
Ciclo de escritura	Define el intervalo de tiempo seleccionado en el que se simularán los valores actuales de las variables. Si aquí se introduce el valor "2", por ejemplo, cada 2 segundos se mostrará el valor actual de la variable.
Simulación	Muestra el modo en el que se procesarán los valores de las variables durante la simulación.
Establecer valor	Define el valor seleccionado para la variable correspondiente. La simulación se inicia con el valor indicado.
minWert maxWert	Define el rango de valores de la variable. Para ello hay que definir un valor mínimo y uno máximo. El valor predeterminado es -32768 para el valor mínimo y 32767 para el valor máximo.
Período	Contiene el período en el que se repite el valor de la variable en los tipos de simulación "Incrementar" y "Decrementar".
Inicio	Inicia la simulación de la variable tras realizar las entradas anteriores.

Tipos de simulación

El simulador dispone de seis tipos de simulación diferentes. En esta simulación, las variables configuradas se abastecen con valores basados en la realidad.

Tipo de simulación	Descripción
Seno	Modifica el valor de la variable en forma de curva sinusoidal. La visualización se efectúa como función periódica no lineal.
Aleatorio	Prepara valores generados casualmente. El valor de la variable se modifica mediante una función aleatoria.
Incremento	Aumenta el valor de la variable continuamente hasta un valor máximo especificado. Tras alcanzar el máximo vuelve a empezar por el mínimo. La evolución de los valores corresponde a una curva positiva con dientes de sierra.
Decremento	Reduce el valor de la variable continuamente hasta un valor mínimo especificado. Tras alcanzar el mínimo vuelve a empezar por el máximo. La evolución de los valores corresponde a una curva negativa con dientes de sierra.
Desplazar bit	Desplaza un bit activado continuamente una posición. La posición anterior se resetea. De este modo se pueden probar, por ejemplo, los avisos de un panel de operador.
<Vista>	El valor actual de la variable se visualiza estáticamente.

Ejemplo: simular variables con el tipo de simulación "Desplazar bit"

Para simular las variables con el tipo de simulación "Desplazar bit", proceda del siguiente modo:

1. Abra el proyecto que debe simularse.
2. Elija el comando de menú "Online > Simular runtime > Con simulador de variables". Se abre el simulador de variables.
3. En la columna "Variable" selecciona una variable del proyecto.
4. Seleccione "Bin" en la columna "Formato".
5. En la columna "Ciclo de escritura", introduzca el valor "1".
6. Seleccione el tipo de simulación "Desplazar bit" en la columna "Simulación".
7. En la columna "Establecer valor", introduzca el valor "1".
8. Active la variable con la casilla de verificación "Inicio".

Resultado

El simulador prueba la variable seleccionada por bits del siguiente modo:

Valores de simulación	Byte para avisos
Valor inicial establecido	00000001
1. Valor de simulación	00000010
2. Valor de simulación	00000100
3. Valor de simulación	00001000
....	...

En runtime verá si para cada valor se emite el aviso deseado, p. ej.

Consulte también

Simular un proyecto (Página 9264)

10.10.2.4 Restricciones en la simulación

Avisos con parámetros dinámicos

Si se utilizan variables o listas de textos como variables externas para avisos, los parámetros dinámicos de los avisos no se mostrarán.

Para la simulación de avisos en el simulador de variables solo pueden usarse variables internas.

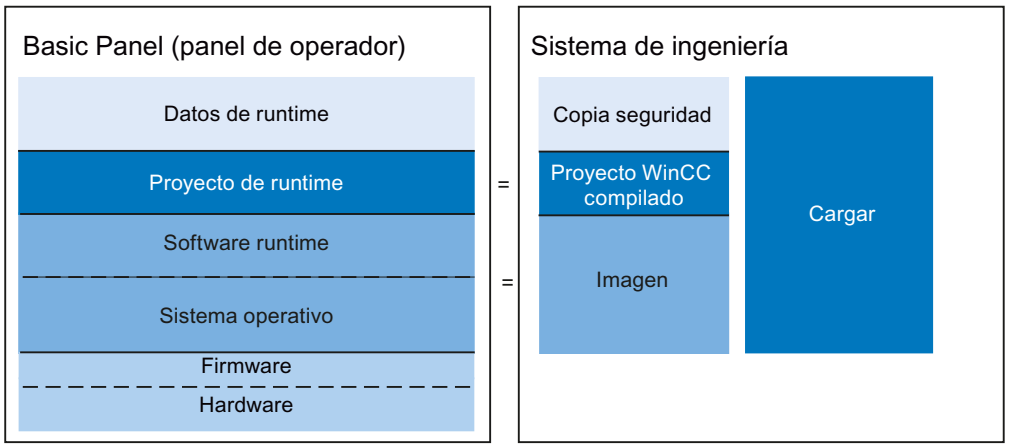
Para simular parámetros dinámicos, utilice PLCSim.

10.10.3 Mantenimiento del panel de operador

10.10.3.1 Resumen del mantenimiento del panel de operador (Basic Panels)

Componentes

La figura siguiente muestra los componentes de software de un panel de operador y su relación con el sistema de ingeniería.



Datos runtime

Los datos de runtime se generan durante el funcionamiento de la instalación y se guardan en el panel de operador. Entre ellos están, p. ej., las recetas y los datos para la administración

de usuarios. Estos datos se sobrescriben durante la carga. Si es necesario, guarde estos datos antes de cargar un proyecto runtime.

Proyecto de Runtime

El proyecto de Runtime contiene los datos de configuración compilados para un panel de operador. Cargue el proyecto de Runtime en el panel de operador desde WinCC.

Software runtime y sistema operativo

El software runtime y el sistema operativo de un panel de operador constituyen la imagen. Para un panel de operador existen diferentes imágenes. Todas las imágenes de un panel de operador están disponibles en WinCC. Dependiendo de la configuración, cargue si es necesario la imagen adecuada junto con el proyecto de runtime en el panel de operador.

Firmware y hardware

En el estado de suministro del panel de operador, el firmware y el hardware ya se encuentran configurados.

10.10.3.2 ProSave

Introducción

Durante la instalación de WinCC, la herramienta de servicio "ProSave" se instala de forma predeterminada. Las funciones de ProSave se abren en WinCC a través del menú "Online > Mantenimiento de dispositivos HMI".

Funcionalidad

ProSave incorpora todas las funciones necesarias para transferir archivos al panel de operador.

- Crear una copia de seguridad y restablecer los datos guardados
- Actualización del sistema operativo
- Configuración de la comunicación

Consulte también

Crear una copia de seguridad de los datos del panel de operador (Página 9272)

Actualizar el sistema operativo (Página 9274)

Resumen del mantenimiento del panel de operador (Basic Panels) (Página 9268)

10.10.3.3 Crear una copia de seguridad de los datos del panel de operador

Introducción

Si se realizan copias de seguridad de los datos de un panel de operador con regularidad, los tiempos de inactividad se reducen a un mínimo, p. ej. al sustituir un dispositivo. Los datos guardados se transfieren al panel de operador y con ello se restablece el estado original.

Crear una copia de seguridad con WinCC

Si un panel de operador está conectado a un PC de configuración, WinCC permite crear una copia de seguridad de los datos y restaurar dicha copia desde este PC.

Alcance de la copia de seguridad

El tipo de panel de operador determina de qué datos se hace copia de seguridad y cuáles se restauran:

- Copia de seguridad completa.
  Según el panel de operador: Runtime, firmware, sistema operativo, configuración, recetas, administración de usuarios, datos de ajuste
- Sólo recetas
- Sólo administración de usuarios
- Recetas como archivo CSV

Al realizar la copia de seguridad de los datos de un panel de operador se crea un archivo comprimido con la extensión "*.psb".

En principio se pueden crear copias de seguridad de datos en cualquier soporte de memoria. Si el panel de operador está integrado en una red, se pueden crear copias de seguridad de los datos también en un servidor.

Nota

Alcance de la copia de seguridad

Al realizar la copia de seguridad se crea una copia de la memoria flash. Si pretende crear una copia de seguridad completa y restablecer también todos los datos para los paneles de operador, tenga en cuenta lo siguiente:

- En caso de restablecer todos los datos, los datos contenidos anteriormente en el panel de operador, incluyendo el sistema operativo, se borrarán sin excepción y de forma irreversible.
- Si se interrumpe la restauración de los datos, ejecute el comando "Restablecer configuración de fábrica". A continuación, vuelva a iniciar la restauración de los datos.

Consulte también

Guardar y restablecer los datos del panel de operador (Página 9273)

ProSave (Página 9269)

Resumen del mantenimiento del panel de operador (Basic Panels) (Página 9268)

10.10.3.4 Guardar y restablecer los datos del panel de operador

Nota

Utilice el restablecimiento de datos del proyecto sólo en los paneles de operador que se hayan configurado con el mismo software de configuración.

Requisitos

- El panel de operador está conectado con el PC de configuración.
- El panel de operador está seleccionado en la navegación del proyecto.
- Si se utiliza un servidor para crear copias de seguridad: El PC de configuración tiene acceso al servidor.

Copia de seguridad de datos del panel de operador

Para crear una copia de seguridad de los datos del panel de operador, proceda del siguiente modo:

1. En el menú "Online > Mantenimiento de paneles de operador HMI", seleccione el comando "Copia de seguridad".
   Se abre el cuadro de diálogo "Crear una copia de seguridad".
2. Seleccione el tipo de interfaz PG/PC y el dispositivo de destino y haga clic en "Crear".
   Se abre el cuadro de diálogo "SIMATIC ProSave".
3. En "Clase de datos", seleccione los datos del panel de operador que desee incluir en la copia de seguridad.
4. Introduzca el nombre del archivo de seguridad en "Guardar como".
5. Haga clic en "Inicio de Backup".

Se inicia la copia de seguridad. Dependiendo de la conexión seleccionada, este proceso puede tardar varios minutos.

Restaurar los datos del panel de operador

Para restaurar los datos del panel de operador, proceda del siguiente modo:

1. En el menú "Online > Mantenimiento de paneles de operador HMI", seleccione el comando "Restaurar".
   Se abre el cuadro de diálogo "Restaurar copia de seguridad".
2. Seleccione el tipo de interfaz PG/PC y el dispositivo de destino y haga clic en "Cargar".
   Se abre el cuadro de diálogo "SIMATIC ProSave".
3. Introduzca el nombre del archivo de seguridad en "Abrir ...".
   En "Acerca del archivo" se muestra información sobre el archivo de seguridad seleccionado.
4. Haga clic en "Inicio de Restore".

Los datos se comenzarán a restablecer. Dependiendo de la conexión seleccionada, este proceso puede tardar varios minutos.

Backup/Restore con el cuadro de diálogo "Backup/Restore" en el Control Panel del panel de operador

La función vía "Backup/Restore" está habilitada para MMC, tarjetas de memoria SD y soportes de memoria USB.

Consulte también

Crear una copia de seguridad de los datos del panel de operador (Página 9270)

Resumen del mantenimiento del panel de operador (Basic Panels) (Página 9268)

10.10.3.5 Actualizar el sistema operativo

Introducción

Si la versión de la imagen de un panel de operador no es compatible con la configuración, al cargar el proyecto se actualizará automáticamente tras la pregunta de confirmación correspondiente. A continuación, el proceso de carga continúa. Si rechaza la actualización de la imagen, la carga se cancela. Con la actualización de la imagen se actualiza también el sistema operativo del panel de operador.

Actualización de la imagen

Para actualizar la imagen, conecte el panel de operador con el PC de configuración. En la medida de lo posible, utilice para esta conexión la interfaz con el ancho de banda más alto, como puede ser Ethernet. Si se utiliza una conexión serie, la actualización de la imagen puede requerir hasta una hora.

"Restablecer configuración de fábrica"

Si el sistema operativo del panel de operador deja de funcionar, actualice el sistema operativo y restablezca la configuración de fábrica de dicho dispositivo.

Nota

Para restablecer las opciones de fábrica a través de Ethernet, necesita lo siguiente:

- Dirección MAC del panel de operador
- Dirección IP disponible
- Interfaz PG/PC ajustada en Ethernet TCP/IP del PC de configuración.

La interfaz PG/PC se configura a través del Panel de control del PC de configuración. En el campo "Punto de acceso de la aplicación" seleccione la entrada "S7ONLINE (STEP7) -> TCP/IP".

Consulte también

Actualizar el sistema operativo del panel de operador (Página 9275)

ProSave (Página 9269)

Resumen del mantenimiento del panel de operador (Basic Panels) (Página 9268)

10.10.3.6 Actualizar el sistema operativo del panel de operador

Si se actualiza el sistema operativo, también se actualiza el software runtime en el panel de operador y se cambia la versión del panel de operador.

ATENCIÓN

La actualización del sistema operativo borra todos los datos del panel de operador

Al actualizar el sistema operativo se borran todos los datos existentes en el sistema de destino. Por tanto, realice antes una copia de seguridad de los siguientes datos:

- Administración de usuarios
- Recetas

Requisitos

- El panel de operador está conectado con el PC de configuración o con el PC con ProSave.
- El panel de operador está seleccionado en la navegación del proyecto.

Actualizar el sistema operativo

Para actualizar el sistema operativo, proceda del modo siguiente:

1. En el menú "Online > Mantenimiento de paneles de operador", seleccione el comando "Actualizar sistema operativo".
2. En el PC de configuración, seleccione el comando "Actualizar sistema operativo" en el menú "Online > Mantenimiento de dispositivos HMI" de WinCC.
   Se abre el cuadro de diálogo "SIMATIC ProSave [OS-Update]". La ruta con la imagen ya está ajustada de forma predeterminada.
3. En caso necesario, seleccione otra ruta para la imagen que desee transferir al panel de operador.
4. Haga clic en "Update OS".

Se inicia la actualización. Dependiendo de la conexión seleccionada, este proceso puede tardar varios minutos.

Restablecer la configuración de fábrica del panel de operador

Para restablecer la configuración de fábrica del panel de operador, proceda del siguiente modo:

1. Desconecte la alimentación del panel de operador.
2. En el PC de configuración, seleccione el comando "Actualizar sistema operativo" en el menú "Online > Mantenimiento de dispositivos HMI" de WinCC.
   Se abre el cuadro de diálogo "SIMATIC ProSave [OS-Update]". La ruta con la imagen ya está ajustada de forma predeterminada.
3. En caso necesario, seleccione otra ruta para la imagen que desee transferir al panel de operador.
4. Active la casilla de verificación "Restablecer configuración de fábrica".
5. Haga clic en "Update OS".
6. Para iniciar "Restablecer configuración de fábrica", vuelva a conectar la alimentación del panel de operador.

Este proceso puede durar varios minutos.

Resultado

El sistema operativo del panel de operador se actualizará.

Consulte también

Actualizar el sistema operativo (Página 9272)

Resumen del mantenimiento del panel de operador (Basic Panels) (Página 9268)

10.10.4 Referencia

10.10.4.1 Mensajes de error al cargar proyectos

Posibles problemas de carga

Mientras se carga un proyecto en un panel de operador, en la ventana de resultados se visualizan avisos de estado que indican el progreso de la carga.

Si al cargar el proyecto en el panel de operador se producen problemas, la causa es una de las siguientes en la mayoría de los casos:

- Ajustes incorrectos para la carga en el panel de operador;
- Tipo de panel de operador incorrecto en el proyecto;
- El panel de operador no está conectado al PC de configuración.

A continuación se listan los casos más frecuentes que provocan la interrupción de la carga con la posible causa y la solución.

La carga serie se interrumpe

Posibles soluciones: Seleccione una velocidad de transferencia inferior.

La carga se interrumpe debido a un conflicto de compatibilidades

Causa posible	Solución
El PC de configuración está conectado a un dispositivo incorrecto, p. ej. un controlador	Compruebe el cableado. Corrija los parámetros de comunicación

El proyecto no se carga

Causa posible	Solución
La conexión con el panel de operador no se puede establecer (aviso en la ventana de resultados).	Compruebe la conexión física entre el PC de configuración y el panel de operador. Compruebe si el panel de operador se encuentra en modo de transferencia. Excepción: control remoto
El driver de comunicación estándar no se ha introducido en el Administrador de dispositivos de Windows	Compruebe el estado del puerto COM en la ventana de propiedades del Administrador de dispositivos.

La carga con la interfaz MPI/DP no funciona

Causa posible	Solución
En el CP está ajustado el "modo configurado", p. ej. si se utiliza el CD de SIMATIC NET.	Ajuste el "Modo PG" con la aplicación "Ajustar equipo PC" para el CP. Compruebe los parámetros de red "Velocidad de transferencia" y "Dirección MPI". Cargue el proyecto en el CP desde WinCC. Ajuste nuevamente el "modo configurado" del CP.
El punto de acceso "S7ONLINE" no se ha asignado a un dispositivo físico, como CP5611 (MPI), en el Panel PG/PC. La causa puede ser la instalación del "CD 7/2001 de SIMATIC".	Ajuste el punto de acceso "S7ONLINE" en el dispositivo que desee con la aplicación "Panel PG/PC" o "Ajustar equipo PC" . Compruebe los parámetros de red "Velocidad de transferencia" y "Dirección MPI". Cargue el proyecto en el panel de operador desde WinCC. Vuelva a asignar el punto de acceso "S7ONLINE" al dispositivo original.

La configuración es demasiado compleja.

Causa posible	Solución
La configuración contiene demasiados objetos u opciones diferentes para el panel de operador seleccionado.	Quite todos los objetos de un tipo, p. ej. todos los visores de gráficos.

10.10.4.2 Adaptar el proyecto a otro panel de operador

Introducción

Al transferir un proyecto de WinCC a un panel de operador, WinCC comprueba si el panel de operador coincide con el tipo utilizado en el proyecto. Si los dos tipos no coinciden, recibirá un aviso durante el proceso de carga.

El proceso de carga se cancelará.

Adaptar el proyecto al panel de operador

Para cargar el proyecto al panel de operador conectado, hay que adaptarlo en correspondencia.

- Inserte un panel de operador nuevo en el árbol del proyecto. En la selección de paneles de operador, elija el tipo adecuado.
- Copie los componentes configurados del panel de operador actual al nuevo.
  Muchos componentes se copian directamente en la navegación del proyecto y en la vista detallada.
  Así, p. ej. copie toda la carpeta "Imágenes" a la carpeta del mismo nombre del panel de operador nuevo utilizando el menú contextual.
- Para las entradas del árbol del proyecto que no disponen del comando "Copiar" en el menú contextual, copie el contenido desde la vista detallada.
- Seleccione, p. ej., la entrada "Recetas" del árbol del proyecto. Las recetas aparecerán en la vista detallada.
- Seleccione las recetas en la vista detallada y arrástrelas hasta la entrada "Recetas" del panel de operador nuevo con la función Drag&Drop. Se copiarán las recetas. La vista detallada admite una selección múltiple.
- Vuelva a configurar los componentes que no pueden copiarse, p. ej. conexiones, punteros de área y avisos.
- Guarde el proyecto de vez en cuando.
- Compile el proyecto por completo.
- Si la compilación termina sin errores, cargue el proyecto en el panel de operador.

Vinculación de referencias

Si se copian objetos, también se copiarán las referencias a objetos conectados. Tras copiar los objetos conectados, las referencias volverán a enlazarse entre sí.

Ejemplo:

Se copia una imagen cuyos objetos están enlazados con variables. Tras insertar la imagen en el panel de operador nuevo, los nombres de variable están registrados en los diferentes objetos. Los nombres de variables están marcados en rojo, puesto que las referencias están abiertas. Si, a continuación, se copian las variables y se insertan en el panel de operador nuevo, se cerrarán las referencias abiertas. La marca roja de los nombres de variable se eliminará.

Para completar las referencias a objetos conectados en el controlador, primero debe configurarse una conexión con el controlador.

Uso del área de información

Si se compila el proyecto para el panel de operador nuevo, se mostrarán errores y advertencias en la ficha "Información" de la ventana de inspección. Con el comando "Ir a" del menú contextual se accede directamente al lugar correspondiente para corregir el error o la advertencia.

Procese la lista de errores y advertencias en orden sucesivo.

Si la compilación termina sin errores, cargue el proyecto en el panel de operador.

10.10.4.3 Establecer una conexión con el panel de operador

Introducción

Para cargar un proyecto de WinCC en un panel de operador, la conexión entre el PC de configuración y el panel de operador debe estar establecida y bien configurada. Si la conexión no se establece durante la carga, se cancela el proceso de carga.

Establecer una conexión entre el PC de configuración y el panel de operador

1. Compruebe la conexión de cables entre el panel de operador y el PC de configuración.
2. Abra en WinCC el editor "Dispositivos y redes" y active la vista de red.
3. En la vista de red, seleccione la subred y compruebe sus ajustes.
4. En la vista de red o de dispositivos, seleccione la interfaz del panel de operador y compruebe los parámetros de conexión en la ventana de inspección.
5. Conecte el panel de operador y pulse el botón "Control Panel" en el Loader.
   Se abrirá el Control Panel.
6. Pulse dos veces en "Transfer" en el Control Panel.
   Se abre el cuadro de diálogo "Transfer Settings".
7. Compruebe los ajustes y pulse seguidamente en "Advanced".
   Se abre el cuadro de diálogo de configuración del protocolo.
   *: El título del cuadro de diálogo depende del protocolo utilizado, como puede ser "Configuración PROFIBUS".
8. Compruebe los ajustes avanzados y cierre el cuadro de diálogo con "OK".

Ajustes importantes

Cuando compruebe la configuración de la conexión, ponga especial atención en los parámetros siguientes:

- Direcciones de red y de estación
- Velocidad de transferencia ajustada
- Maestro del bus; por lo general solo se admite un maestro.

Si se utiliza un adaptador ajustable para la conexión, compruebe también sus ajustes, p. ej. velocidad de transferencia y maestro del bus.

10.11 Manejar en runtime

10.11.1 Principios básicos

10.11.1.1 Visión de conjunto

Fase de configuración y fase de control del proceso

Los paneles de operador se utilizan para realizar tareas de manejo y visualización en la automatización de los procesos y de la producción. Las imágenes de la instalaciones visualizadas en los paneles de operador permiten representar los procesos de forma más clara. El proyecto contenido en el panel de operador (que contiene, entre otros, las imágenes de la instalación) se crea en la fase de configuración.

Para utilizar el proyecto en la fase de control del proceso, debe transferirse al panel de operador. Otro requisito para la fase de control del proceso es que el panel de operador esté acoplado online a un controlador. Seguidamente, es posible controlar los procesos de trabajo que se están ejecutando (manejo y visualización).

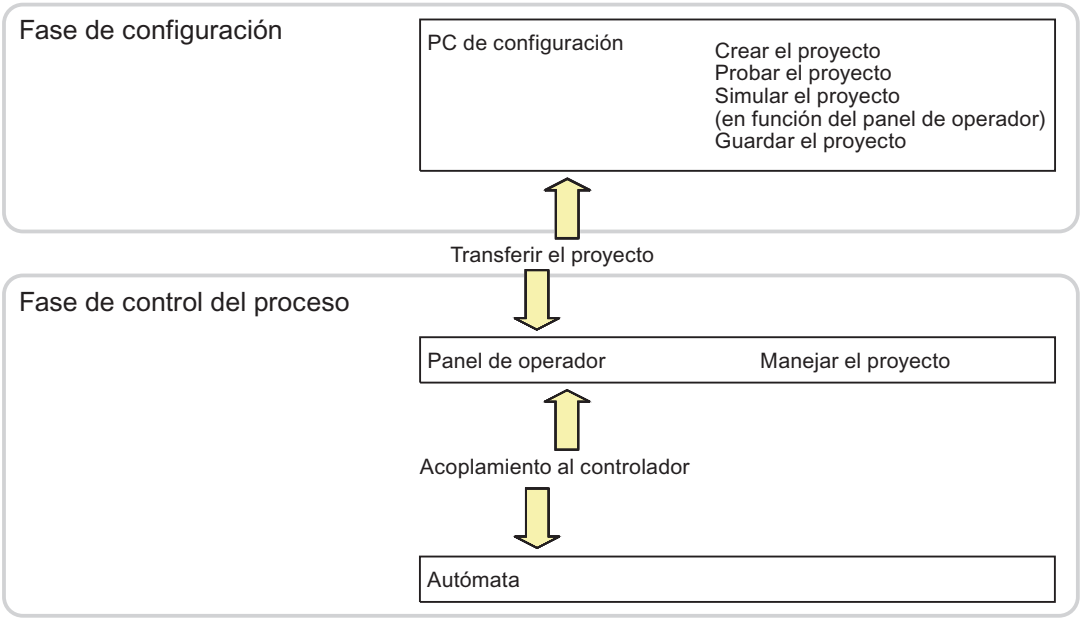


Figura 10-3 Fase de configuración y fase de control del proceso

Cargar el proyecto en el panel de operador

Existen diversos procedimientos para cargar un proyecto en el panel de operador, a saber:

- Cargar desde el PC de configuración
- Restablecimiento vía ProSave desde un PC
  En este caso, la copia de seguridad del proyecto se carga en el panel de operador desde un PC. En dicho PC no debe estar instalado el software de configuración.

Estos procedimientos se pueden seleccionar tanto en la primera puesta en marcha como al volver a utilizar un proyecto.

Primera y siguiente puesta en marcha

- En la primera puesta en marcha, el panel de operador no contiene ningún proyecto.
  El panel de operador tiene este estado también tras actualizar el sistema operativo.
- En la siguiente puesta en marcha se sustituye un proyecto existente en el panel de operador.

10.11.1.2 Variables en Runtime

Definición

Las variables corresponden a posiciones de memoria definidas en el panel de operador. En ellas se escriben valores y de ellas se leen valores. Esto puede realizarse desde el controlador, o bien desde el panel de operador.

10.11.1.3 Funciones de sistema en runtime

Utilización

Las funciones de sistema tienen la finalidad siguiente en runtime:

- Controlar el proceso
- Utilizar las propiedades del panel de operador
- Configurar los ajustes del sistema online en el panel de operador

En WinCC, cada función de sistema está asociada a un objeto y un evento. En cuanto aparece el evento, se lanza la función de sistema.

Funciones del sistema

Las funciones de sistema son funciones predefinidas con las que se realizan muchas tareas en runtime, p. ej.:

- Cálculos (p. ej. aumentar el valor de una variable en una cantidad fija o variable)
- Funciones de archivación (p. ej. iniciar un fichero de valores de proceso)

- Configuración (p. ej. cambiar de controlador o activar un bit en el controlador)
- Avisos (p. ej. tras cambiar de usuario)

Eventos

Los eventos que se pueden definir como disparador para la ejecución de una función de sistema, dependen del objeto en cuestión y de la función seleccionada.

Así, por ejemplo, los eventos "Cambio de valor", "Rebase por defecto del límite inferior" y "Rebase por exceso del límite superior" pertenecen al objeto "Variable". Los eventos "Aparecido" y "Desaparecido" pertenecen al objeto "Imagen".

10.11.2 Poner en marcha proyectos

10.11.2.1 Configuración del software runtime

En el software de configuración WinCC se realizan los ajustes siguientes en el software runtime:

Visualización en el sistema de destino

En WinCC se configura cómo debe representarse en runtime en proyecto generado. En los Basic Panels, la resolución de pantalla es fija. Si la imagen es más grande que la resolución de pantalla ajustada, aparecen barras de desplazamiento.

Para desactivar la barra de tareas, seleccione el comando "Configuración > Barra de tareas y menú Inicio" del menú Inicio. En el cuadro de diálogo "Propiedades de Barra de tareas y menú Inicio", desactive "Siempre visible" y "Ocultar automáticamente".

Fuente de los cuadros de diálogo

Los textos de los cuadros de diálogo se representan en la fuente estándar. La fuente estándar se determina en el editor "Configuración del panel", en "Idioma & fuente".

Bloquear la conmutación de programas

La conmutación de programas se puede bloquear para impedir que el usuario abra otras aplicaciones en runtime.

En el editor "Configuración del panel" haga clic en "Ventana" y active "Bloquear cambio de programa". Además, desactive la barra de tareas de Windows.

Nota

Finalizar runtime

En caso de bloquear la conmutación de programas, es absolutamente necesario vincular en el proyecto la función de sistema "PararRuntime" con una tecla de función o con un botón, por ejemplo. De lo contrario no se cerrarán ni el runtime ni Windows.

Protector de pantalla

La mayoría de los monitores ya no requieren protector de pantalla, incluso es perjudicial. Estos monitores se desconectan cuando la señal de vídeo no cambia durante un tiempo configurable. Un protector de pantalla convencional impide esta desconexión que prolonga la vida útil del monitor.

Nota

Protectores de pantalla autorizados

Si desea utilizar un protector de pantalla, tenga en cuenta que runtime solo admite los protectores de pantalla estándar incluidos en el volumen de suministro de Windows.

Asegúrese de que el PC en el que funciona el software Runtime tiene la zona horaria correcta. La zona horaria se configura en Windows con "Inicio > Configuración > Panel de control > Fecha y hora" (ficha "Zona horaria").

10.11.2.2 Cargar un proyecto

Visión de conjunto

Para cargar el proyecto hay varios escenarios posibles:

- El software Runtime está instalado en el mismo sistema que el software de configuración.
- El software Runtime está instalado en un sistema distinto que el software de configuración. El proyecto se carga en el sistema de destino desde el PC de configuración. Los paneles de operador se conectan con el PC de configuración para la transferencia. Además, es imprescindible que el modo de transferencia de los paneles de operador coincida con el de WinCC.

  Nota

  En función de la configuración, se realizarán consultas de seguridad durante la carga. Los datos de recetas y la lista de contraseñas existentes en el panel de operador se sobrescribirán previa consulta.

Software de configuración y software runtime en el mismo sistema

Si el software de configuración y el software Runtime están en el mismo sistema, proceda del siguiente modo:

1. Cree el proyecto y compílelo a continuación.
2. Inicie el Runtime directamente desde el software de configuración que se está ejecutando. Elija el comando "Iniciar runtime" del menú "Online".
3. En caso de haber configurado una comunicación con el controlador, puede proceder a probar y utilizar el proyecto con el controlador.

Software de configuración y software runtime en distintos sistemas

Si el software de configuración y el software Runtime están en dos sistemas distintos, proceda del siguiente modo:

1. Cree el proyecto y compílelo a continuación. Encontrará más información al respecto en "Compilar un proyecto".
2. Si desea cargar el archivo por cable:
   Conecte el panel de operador con el PC de configuración utilizando un cable estándar adecuado al tipo de transferencia deseado y conecte el panel de operador.
3. Ponga el panel de operador en modo de transferencia.
   Para iniciar el modo de transferencia, pulse el botón "Transfer" en el Loader. Alternativamente, asocie la función de sistema "AjustarModoDeOperacion" a un objeto de manejo.
4. Cargue el proyecto en el sistema de destino desde el PC de configuración. Encontrará más información al respecto en "Cargar un proyecto".

Nota

Si el panel de operador es un PC, es posible transferir el archivo generado sin necesidad del Loader, p. ej., vía Ethernet. En el PC haga doble clic en el archivo e inicie así el Runtime.

10.11.2.3 Iniciar runtime en la estación de ingeniería

Introducción

Mientras se configura en WinCC, se puede iniciar simultáneamente el proyecto en runtime en la estación de ingeniería. Sin embargo, esta configuración online sufre algunas restricciones.

Mientras el runtime se ejecuta en la estación de ingeniería, el proyecto no se compila en el fondo. Si, tras finalizar el runtime, se carga el proyecto en un panel de operador, la parte del proyecto que se ha modificado entretanto se compilará automáticamente. Otra posibilidad es iniciar la compilación manualmente.

Al iniciar el proyecto en runtime serán efectivos los ajustes que se han almacenado en el editor "Configuración" para el proyecto del panel de operador.

Requisitos

En la estación de ingeniería debe haber un proyecto abierto.

Procedimiento

Para iniciar el runtime en la estación de ingeniería proceda del siguiente modo:

1. Seleccione el panel de operador deseado en el árbol del proyecto.
2. Elija el comando de menú "Online > Iniciar runtime".
3. Si, tras iniciar el runtime, se modifica el proyecto en la estación de ingeniería, seleccione el comando "Compilar > Software" del menú contextual del panel de operador.

En el runtime de la estación de ingeniería aparecerá el proyecto actualizado.

10.11.2.4 Iniciar el runtime en el panel de operador

Introducción

El proyecto se puede iniciar en runtime en cuanto se ha cargado en el panel de operador. El proyecto se guarda en el panel de operador en un archivo con la extensión siguiente:

- Basic Panels así como OP 73, OP 77A y TP 177A: "*.srt"

Al iniciar el proyecto en runtime, se hacen efectivos los ajustes establecidos para el proyecto en la "Configuración de runtime" del panel de operador.

Los programas para iniciar proyectos en el panel de operador se encuentran en el directorio de instalación de runtime.

Nota

Finalización automática de runtime

Si en el panel de operador está activada la transferencia automática y en el PC de configuración se inicia una transferencia, el proyecto activo se cierra automáticamente.

El panel de operador cambia entonces automáticamente al modo de operación "Transfer".

Una vez concluida la fase de puesta en servicio, desactive la transferencia automática para que el panel de operador no pase por error al modo de transferencia.

El modo de transferencia puede provocar reacciones imprevistas en la instalación.

Para bloquear el acceso a los ajustes de transferencia, y evitar así modificaciones no autorizadas, asigne una contraseña en el Control Panel.

Requisitos

El panel de operador tiene instalado WinCC Runtime.

Procedimiento

En un dispositivo se inicia la aplicación "RT Loader". En función de la configuración, el proyecto cargado se inicia automáticamente tras un tiempo de retardo.

Si el proyecto no se inicia automáticamente:

1. Para seleccionar el archivo de proyecto, haga clic en "Configuración" e introduzca la ruta del archivo de proyecto en "Archivo de configuración".
2. Haga clic en "Aceptar" y a continuación en "Inicio".

10.11.2.5 Comprobar el proyecto

Introducción

Existen las posibilidades siguientes para probar un proyecto de WinCC:

Probar el proyecto en el PC de configuración.

- Simulador
  El simulador se utiliza para probar proyectos de WinCC con variables internas y variables de proceso. Encontrará más información al respecto en "Simular un proyecto".
  El simulador permite probar lo siguiente:
  - Comprobar una configuración sin estar conectado el controlador
  - Comprobar una configuración estando conectado el controlador pero sin un proceso en curso
  - Implementar un proyecto para fines de demostración

Probar el proyecto en el panel de operador

- Probar el proyecto offline en el panel de operador
  Probar offline significa que, durante el test, no hay ninguna conexión de comunicación entre el panel de operador y el controlador. Si bien se utiliza el panel de operador, no se transfieren datos al controlador ni viceversa. Para ajustar el modo de operación "Offline" en el panel de operador, asocie la función de sistema "AjustarModoDeOperacion" a un objeto de manejo.
- Probar el proyecto online en el panel de operador
  Probar online significa que, durante el test, hay una conexión de comunicación entre el panel de operador y el controlador. La instalación se opera con el panel de operador de acuerdo con la configuración. Para ajustar el modo de operación "Online" en el panel de operador, asocie la función de sistema "AjustarModoDeOperacion" a un objeto de manejo.

Procedimiento

Para simular un proyecto en el PC de configuración sin conexión a un controlador, proceda del siguiente modo:

1. Cree un proyecto tal y como debe funcionar posteriormente con el controlador conectado.
2. Guarde y compile el proyecto.
3. Inicie el simulador directamente desde el software de configuración que se está ejecutando. Elija el comando de menú "Online > Simular runtime > Con simulador de variables". Cuando simule el proyecto por primera vez, el simulador se iniciará con una nueva tabla de simulación vacía. Si ya ha creado una tabla de simulación para el proyecto, ésta se abrirá.
   La tabla de simulación "*.six" contiene todos los ajustes necesarios para la simulación. Encontrará más información al respecto en "Trabajar con el simulador de variables".
4. Ahora, en la tabla de simulación, manipule las variables y punteros de área del proyecto. Pase de la tabla de simulación al runtime con las teclas <Alt + Tab>.
   Guarde los ajustes para la simulación con el comando de menú "Archivo > Guardar". Introduzca un nombre adecuado para el archivo. El archivo recibe automáticamente la extensión "*.six".

10.11.2.6 Salir del proyecto

Introducción

En la configuración se define la manera de salir del Runtime.

Procedimiento

Para salir del runtime, proceda del siguiente modo:

1. Cuando el runtime funcione, salga del runtime utilizando el símbolo para cerrar o el administrador de tareas.
2. Cuando el runtime funciones, pulse el botón previsto para salir del runtime. La salida del runtime se configura por separado.

10.11.2.7 Guardar y restaurar los datos del panel de operador

Introducción

Realice copias de seguridad de los datos de un panel de operador en intervalos regulares.

Si un panel de operador está conectado a una estación de ingeniería, podrá guardar y restaurar los datos del panel a partir de esta última con WinCC.

Si WinCC no está instalado en un PC, puede crear cómodamente una copia de seguridad central utilizando ProSave.

Requisitos

- El panel de operador debe estar conectado con la estación de ingeniería o el PC en el que está instalado ProSave.
- El panel de operador cuyos datos deben guardarse o restaurarse debe estar seleccionado en el árbol del proyecto.
- En las propiedades del panel de operador deben estar bien ajustada la configuración de carga.
- Si se utiliza un medio de almacenamiento separado, p. ej. un servidor de datos: el panel de operador está conectado con el medio de almacenamiento.

Procedimiento

Para crear una copia de seguridad de los datos, proceda del siguiente modo:

1. En el menú "Online > Mantenimiento de paneles de operador", seleccione el comando "Copia de seguridad".
2. Elija el ámbito de la copia de seguridad: "Copia de seguridad completa", "Recetas" o "Administración de usuarios".
3. Haga clic en el botón "...", seleccione la ubicación en el cuadro de diálogo "Seleccionar archivo de copia de seguridad" e introduzca un nombre.
4. Haga clic en "Aceptar".

Se inicia la copia de seguridad. La copia de seguridad puede durar varios minutos.

Procedimiento al restaurar los datos

1. En el menú "Online > Mantenimiento de paneles de operador", seleccione el comando "Restaurar...".
2. Haga clic en "..." y seleccione la ubicación y el archivo en el cuadro de diálogo "Abrir". En el área "Contenido" se indica de qué panel de operador proviene la copia de seguridad, así como su tamaño.
3. Haga clic en "Aceptar".

Los datos se comenzarán a restaurar. Este proceso puede durar varios minutos.

10.11.3 Idiomas de runtime

10.11.3.1 Idiomas de runtime

Utilizar varios idiomas de runtime

Es posible determinar qué idiomas del proyecto se utilizarán en runtime en un panel de operador determinado. El número de idiomas de runtime que pueden estar disponibles

simultáneamente en el panel de operador depende del tipo de equipo. Para que el usuario pueda cambiar de idioma en runtime, debe configurarse un elemento de control pertinente.

Al inicio de runtime se visualizará el proyecto en el idioma configurado en último lugar. En el primer inicio de runtime aparece el idioma con el número más bajo en "Orden del idioma".

Ajustes en la configuración

En el editor "Idioma y fuente" se determina lo siguiente:

- Los idiomas del proyecto que están disponibles como idiomas de runtime para el panel de operador.
- El orden en el que se cambian los idiomas al efectuar un cambio de idioma.

10.11.3.2 Ajustar idioma de runtime

Introducción

El editor "Idioma y fuente" muestra todos los idiomas disponibles en el proyecto. Determine qué idiomas del proyecto serán idiomas de runtime en el panel de operador. Asimismo, se determina el orden en el que se cambiarán los idiomas cuando se realice un cambio de idioma.

Requisitos

En el editor "Idiomas del proyecto" hay varios idiomas activados.

Procedimiento

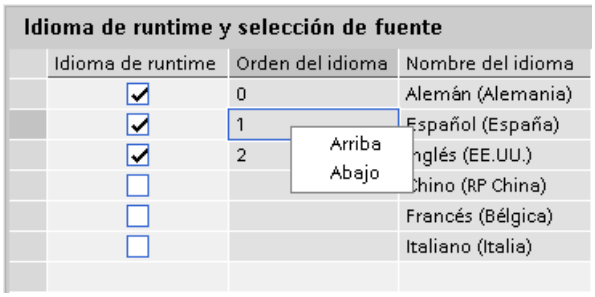
1. Abra el editor "Idioma y fuente" en "Configuración del dispositivo".
2. En la columna "Idioma de runtime", active el idioma que se utilizará la primera vez que se inicie el runtime.
   En la columna "Orden del idioma" se asignará el número "0" al idioma seleccionado.
3. En la columna "Idioma de runtime", active el idioma que se activará como segundo idioma al realizar un cambio de idioma.
   En la columna "Orden del idioma" se asignará el número "1" al idioma seleccionado.

Idioma & fuente

Idioma de runtime y selección de fuente

Idioma de runtime	Orden del idioma	Nombre del idioma	Fuente fija 0	Fuente predeterminada	Fuente configurada
☑	0	Alemán (Alemania)	Tahoma	Tahoma, 11px	<None>
☑	1	Español (España)	Tahoma	Tahoma, 11px	<None>
☑	2	Inglés (EE.UU.)	Tahoma	Tahoma, 11px	<None>
☐		Chino (RP China)	SimSun	SimSun, 16px	<None>
☐		Francés (Bélgica)	Tahoma	Tahoma, 11px	<None>
☐		Italiano (Italia)	Tahoma	Tahoma, 11px	Arial

4. Active otros idiomas en el orden en el que se activarán al cambiar de idioma.
   Si selecciona más idiomas de los que pueden cargarse en el panel de operador, esto se indicará marcando en color el fondo de la tabla.
5. Si desea cambiar el orden de un idioma, seleccione la fila deseada y elija el comando "Move up" o "Move down" del menú contextual.



Resultado

Los idiomas de runtime activados se cargarán en el panel de operador con el proyecto compilado.

En el primer inicio de runtime aparece el proyecto en idioma con el número más bajo en "Orden del idioma".

Si se ha configurado un cambio de idioma con la función de sistema "AjustarIdioma", el cambio de idioma se produce en el orden de los números asignados.

10.11.3.3 Ajustar fuente para un idioma de runtime

Introducción

En el editor "Idioma y fuente" se determina para cada idioma de runtime con qué fuente se visualizarán los textos de este idioma en el panel de operador. La fuente predeterminada se utilizará para todos los textos en los que no pueda definirse ningún tipo de fuente.

En este caso, WinCC ofrece sólo las fuentes soportadas por el panel de operador.

La fuente predeterminada también se utilizará para representar los cuadros de diálogo del sistema operativo del panel de operador. Si los textos o los títulos de los cuadros de diálogo no se visualizan por completo, seleccione una fuente más pequeña como fuente predeterminada.

Requisitos

En el editor "Idiomas del proyecto" hay varios idiomas activados.

Procedimiento

1. Abra el editor "Idioma y fuente" en "Configuración del dispositivo".
2. En la columna "Idiomas de runtime", active los idiomas que deben visualizarse en el panel de operador.
   En la columna "Fuente fija 0", WinCC muestra las fuentes utilizadas por defecto en runtime.
3. En la columna "Fuente configurada 0" defina otra fuente para cada uno de los idiomas disponibles durante la configuración.
   Al transferir, estos tipos de fuente también se transfieren al panel de operador.
4. En la columna "Fuente predeterminada" seleccione la fuente que se utilizará por defecto cuando no pueda seleccionar ninguna fuente para un texto.

Resultado

Los textos del proyecto del idioma seleccionado se visualizarán con el tipo de fuente elegido en el panel de operador.

10.11.3.4 Configurar cambio de idioma

Introducción

Para que pueda haber varios idiomas de runtime disponibles en el panel de operador, es preciso configurar un cambio de idioma. Sólo de este modo el usuario podrá elegir entre diferentes idiomas de runtime.

Métodos para cambiar de idioma

Pueden configurarse los métodos siguientes para cambiar de idioma:

- Selección directa de idioma
  Cada idioma se configura mediante un botón propio. En este caso, cree un botón para cada idioma de runtime. Configure en cada botón la función de sistema "AjustarIdioma" con el número o la abreviatura del idioma como parámetro.
- Cambio de idioma
  El operador cambia de idioma mediante un botón.
  Configure en este botón la función de sistema "AjustarIdioma" con el parámetro "Conmutar". El idioma se activará en el orden definido en el editor "Idioma y fuente".

En ambos métodos, los títulos de los botones deben traducirse a todos los idiomas utilizados. Asimismo, puede configurarse un campo de salida que muestre el idioma seleccionado en ese momento.

10.11.3.5 Particularidades de los idiomas asiáticos y orientales en runtime

Introducción

Para que los proyectos creados en idiomas asiáticos funcionen correctamente en runtime es preciso tener en cuenta ciertas particularidades.

Nota

Durante la configuración sólo pueden utilizarse las fuentes asiáticas soportadas por el PC de configuración.

Memoria necesaria para los caracteres asiáticos

Como es natural, el espacio de memoria necesario para los caracteres asiáticos es mayor que en el caso de los idiomas occidentales. Por tanto, es necesario tener en cuenta los mensajes de error correspondientes al compilar el proyecto.

Introducción de caracteres orientales y asiáticos (no ANSI)

En los Basic Panels no es posible introducir caracteres orientales ni asiáticos en runtime.

Interpretación de caracteres asiáticos

Si se utilizan Sm@rtAccess y Sm@rtService sólo se podrán usar caracteres conocidos en el panel de operador. Para poder utilizar caracteres asiáticos, éstos se deberán configurar en el sistema de ingeniería. Los caracteres configurados adicionalmente requieren más espacio en el panel de operador. Tenga en cuenta la memoria disponible en el panel de operador.

Tamaño de fuente para los caracteres asiáticos

Si crea un proyecto para idiomas asiáticos utilice para la representación en runtime un tamaño de fuente de como mínimo 10 puntos. Si utiliza una fuente más pequeña los caracteres asiáticos serán ilegibles. Esto es aplicable también a la fuente predeterminada en el editor "Idioma y fuente".

Longitud de los campos de texto en idiomas asiáticos

Si edita proyectos multilingües con idiomas asiáticos, tenga en cuenta la longitud de los campos de texto. Dependiendo del tipo y tamaño de la fuente utilizada puede ocurrir que los campos no se visualicen en su totalidad.

1. En la ventana de inspección del campo de texto seleccione "Propiedades > Representación".
2. Desactive la opción "Adaptar automáticamente" en "Adaptación del tamaño".
3. Compruebe si es correcta la visualización en runtime.

10.11.4 Manejar el proyecto

10.11.4.1 Principios básicos

Visión de conjunto del manejo de un proyecto

Todos los paneles de operador Basic están equipados con una pantalla táctil. Algunos paneles de operador Basic poseen teclas de función. Con la pantalla táctil y las teclas de función se maneja el Control Panel o el proyecto que se está ejecutando en el panel de operador.

> ⚠ PELIGRO
>
> **Operación incorrecta**
>
> Un proyecto puede contener acciones de operador que exigen amplios conocimientos de la instalación por parte del operador.
>
> Asegúrese de que la instalación sólo pueda ser operada por personal técnico especializado.

Manejo de la pantalla táctil

> ATENCIÓN
>
> **Deterioro de la pantalla táctil**
>
> Los objetos puntiagudos o afilados pueden dañar la superficie de plástico de la pantalla táctil.
>
> Maneje la pantalla táctil únicamente con el dedo o con un lápiz apropiado.
>
> **Activación de acciones no intencionadas**
>
> En caso de tocar simultáneamente varios objetos de manejo pueden activarse acciones no intencionadas.
>
> No toque nunca más de un objeto a la vez en la pantalla.

Los objetos de manejo son representaciones sensibles al contacto que aparecen en la pantalla del panel de operador.

Básicamente, se manejan del mismo modo las teclas mecánicas. Los objetos de manejo se manejan con el dedo.

> **Nota**
>
> En cuanto el panel de operador detecta que se ha tocado un objeto, reacciona con una respuesta óptica.
>
> La respuesta óptica no depende de la comunicación con el controlador. Por tanto, la respuesta no es ningún indicio de que la acción deseada se llevará a cabo realmente.

Ejemplos de los objetos de manejo:

- Botones
  Los botones pueden adoptar los estados siguientes:

tocado"

cado":

- Botones invisibles
  De forma estándar, el foco de los botones invisibles no aparece marcado al seleccionarlos. En este caso no hay respuesta óptica.
  No obstante, el ingeniero puede configurar los botones invisibles de manera que sus contornos puedan verse en forma de línea al tocarlos. Los contornos permanecerán visibles hasta que se seleccione un objeto de manejo diferente.
- Campos de E/S
  Al tocar un campo E/S, como respuesta óptica aparece un teclado de pantalla, p. ej. para introducir una contraseña.
  Dependiendo del panel de operador y del objeto configurado se visualizan distintos teclados de pantalla que permiten introducir valores numéricos o alfanuméricos.
  Al finalizar la entrada, el teclado de pantalla se oculta automáticamente.

Manejo de las teclas de función

Las teclas de función pueden tener una asignación global o local:

- Teclas de función con asignación global de funciones
  Una tecla de función con asignación global activará en el panel de operador o en el controlador siempre la misma acción, independientemente de la imagen que esté abierta en ese momento. Una acción tal puede ser p. ej. la activación de una imagen o el cierre de una ventana de avisos.
- Teclas de función de asignación local
  Una tecla de función de asignación local es específica de la imagen en que se utiliza y, por consiguiente, sólo tiene efecto en la imagen activa.
  La función de una tecla de función puede variar de imagen a imagen.

En una misma imagen una tecla de función puede tener una sola asignación: o global o local. Si hay asignación local y global, tiene preferencia la local.

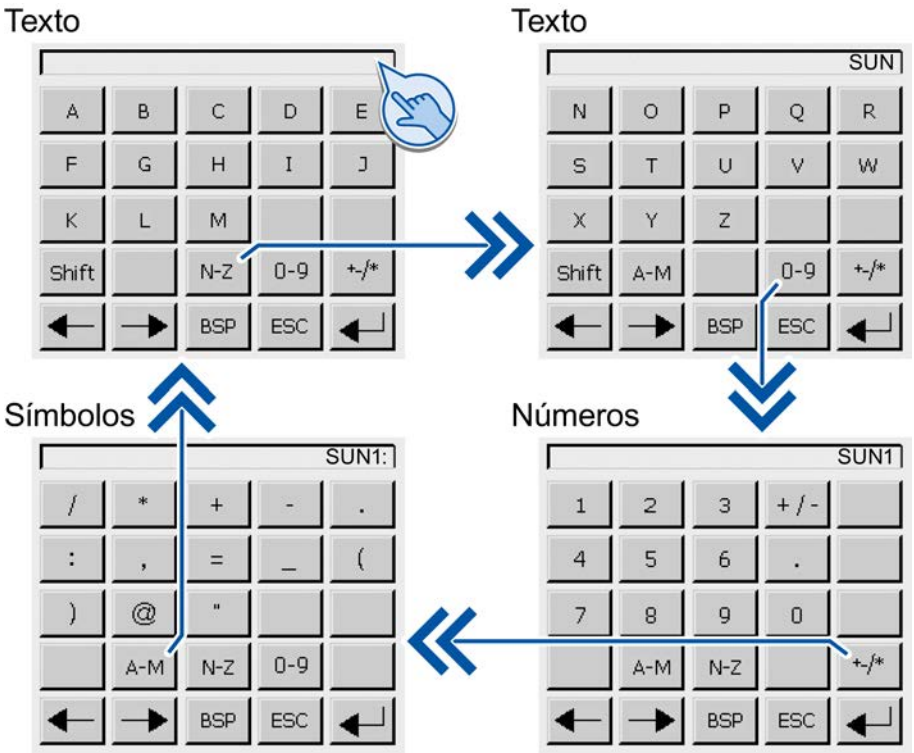
Funciones generales del teclado de pantalla

Las teclas siguientes están disponibles en el teclado de pantalla de todos los paneles de operador Basic:

←	Cursor hacia la izquierda
→	Cursor hacia la derecha
BSP	Borrar un carácter
ESC	Cancelar la entrada
↵	Confirmar la entrada
Help	Mostrar un texto de ayuda. Esta tecla sólo aparece si se ha configurado un texto de ayuda para el objeto de manejo.

Introducir datos en el KTP400 Basic

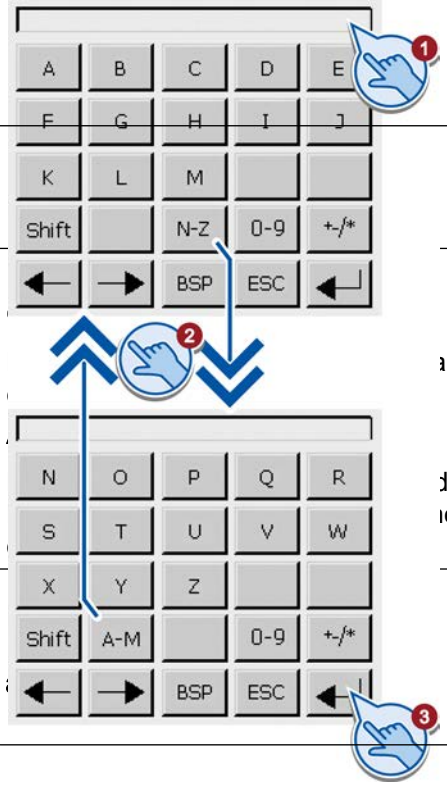
Dado que su display es menor, el teclado de pantalla y el concepto de entrada del KTP400 Basic se distinguen de los de los restantes paneles de operador Basic.



Si en la pantalla táctil del panel de operador toca un objeto que exija una entrada, aparecerá un teclado de pantalla.

El teclado de pantalla del KTP400 posee cuatro vistas. Con las teclas situadas en la cuarta línea del teclado de pantalla puede cambiar la vista al introducir valores.

	Cambia a la vista
	Entrada de texto, caracteres "A" a "M"
	Entrada de texto, caracteres "N" a "Z"
	Entrada de números, "0" a "9", con o sin signo y decimales
Shift	Entrada de caracteres especiales
	Entrada de texto, conmutación a minúsculas

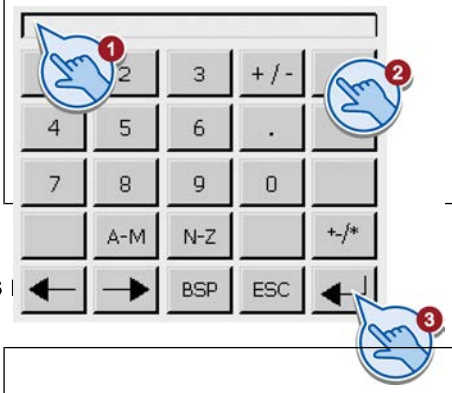


, la orden de control 51 "Selección de imagen" no tiene

do alfanumérico es en un solo idioma.
o tiene efecto alguno sobre la asignación del teclado

Introducir valores

	1. Toque el objeto de control deseado en la imagen. Se abre el teclado de pantalla alfanumérico. 2. Introduzca el valor. Dependiendo de la configuración, al hacerlo, el panel de operador emitirá una señal acústica. Con las teclas <N-Z> y <A-M> cambie la vista del teclado de pantalla, si es necesario. Utilice la tecla <Mayus> para introducir minúsculas. 3. Confirme la entrada con <Intro> o rechácela con <Esc>. En ambos casos se cerrará el teclado de pantalla.



Introducir valores

	1. Toque el objeto de control deseado en la imagen. Se abrirá el teclado de pantalla numérico. 2. Introduzca el valor. Dependiendo de la configuración, al hacerlo, el panel de operador emitirá una señal acústica. Para introducir números en formato hexadecimal cambie la vista del teclado de pantalla con las teclas <N-Z> y <A-M>. 3. Confirme la entrada con <Intro> o rechácela con <Esc>. En ambos casos se cerrará el teclado de pantalla.

Comprobar los límites de los valores numéricos

Para las variables se pueden configurar valores límite. Si introduce un valor que se encuentre fuera de estos límites, el valor no se aplicará. Si hay configurada una ventana de avisos se visualizará un mensaje de sistema y volverá a aparecer el valor original.

Decimales de los valores numéricos

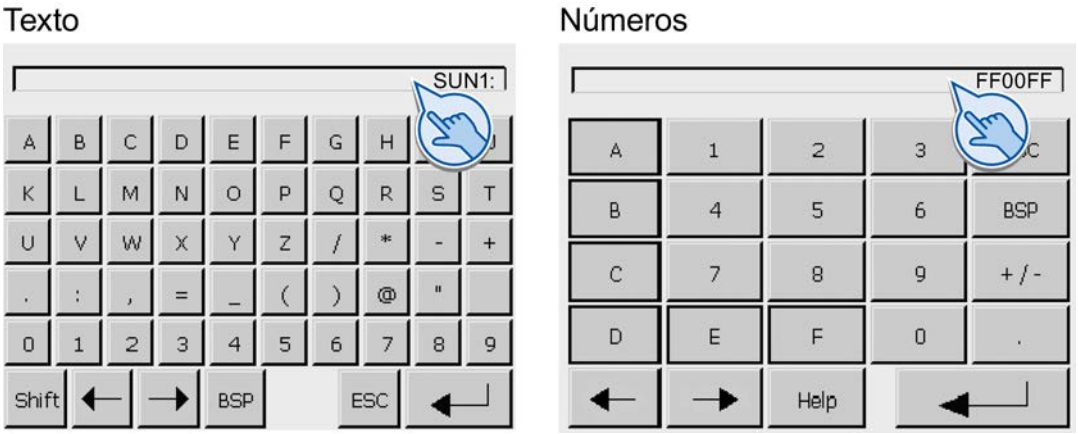
El ingeniero de configuración puede determinar el número de decimales que se deben introducir en un campo de entrada numérico. En este caso, al introducir un valor en un campo ES se comprobará el número de decimales.

- Los decimales que sobren se ignorarán.
- Los decimales que falten se rellenarán con "0".

Introducir datos en el KTP600, KTP1000, TP1500 Basic

Teclado de pantalla alfanumérico

Si en la pantalla táctil del panel de operador toca un objeto que exija una entrada, aparecerá un teclado de pantalla.



Nota

Orden de control sin efecto

Estando abierto el teclado de pantalla, la orden de control 51 "Selección de imagen" no tiene efecto.

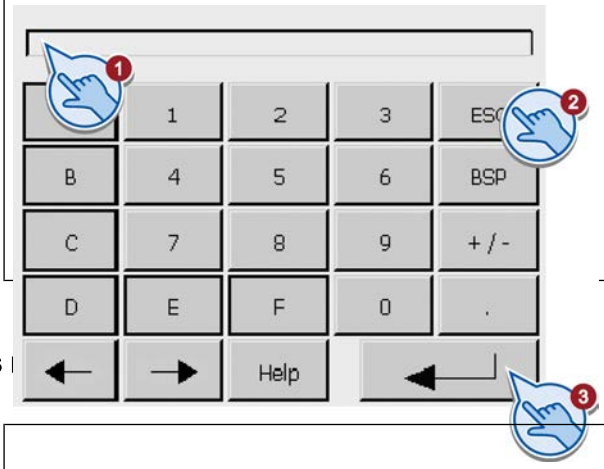
Asignación de teclas

La asignación de las teclas del teclado alfanumérico es en un solo idioma.
El cambio de idioma en el proyecto no tiene efecto alguno sobre la asignación del teclado de pantalla alfanumérico.

Introducir valores

	1. Toque el objeto de control deseado en la imagen. Se abre el teclado de pantalla alfanumérico. 2. Introduzca el valor. Dependiendo de la configuración, al hacerlo, el panel de operador emitirá una señal acústica. Utilice la tecla <Mayus> para introducir minúsculas. 3. Confirme la entrada con <Intro> o rechácela con <Esc>. En ambos casos se cerrará el teclado de pantalla.

Introducir valores



	1. Toque el objeto de control deseado en la imagen. Se abrirá el teclado de pantalla numérico. 2. Introduzca el valor. Dependiendo de la configuración, al hacerlo, el panel de operador emitirá una señal acústica. 3. Confirme la entrada con <Intro> o rechácela con <Esc>. En ambos casos se cerrará el teclado de pantalla.

Comprobar los límites de los valores numéricos

Para las variables se pueden configurar valores límite. Si introduce un valor que se encuentre fuera de estos límites, el valor no se aplicará. Si hay configurada una ventana de avisos se visualizará un mensaje de sistema y volverá a aparecer el valor original.

Decimales de los valores numéricos

El ingeniero de configuración puede determinar el número de decimales que se deben introducir en un campo de entrada numérico. En este caso, al introducir un valor en un campo ES se comprobará el número de decimales.

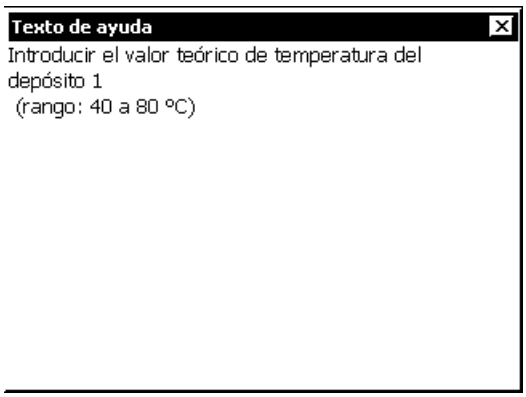
- Los decimales que sobren se ignorarán.
- Los decimales que falten se rellenarán con "0".

Mostrar tooltip

Utilización

Los tooltips configurados por el ingeniero proporcionan información adicional y consejos de manejo. El ingeniero puede configurar tooltips para imágenes y objetos de manejo.

El tooltip de un campo E/S puede contener p. ej. información acerca del valor que debe introducirse.



Procedimiento

Para abrir el tooltip de los objetos de manejo, proceda del siguiente modo:

1. Toque el objeto de manejo deseado.
   Se abrirá el teclado de pantalla. La representación de la tecla Help indica si se ha configurado un tooltip para el objeto de manejo o para la imagen actual.
2. Pulse la tecla Help en el teclado de pantalla.
   Se mostrará el tooltip del objeto de manejo. Si no existe ningún tooltip para el objeto de manejo seleccionado, se mostrará el tooltip de la imagen actual (si se ha configurado). Para desplazar el contenido en tooltips largos, utilice los botones ▼ y ▲.

   Nota

   Conmutar entre los tooltips mostrados

   El ingeniero puede configurar un tooltip para un campo E/S y para la imagen correspondiente. Para conmutar entre ambos tooltips, toque la ventana del tooltip.

3. Cierre el tooltip mostrado con el botón ☒.

Procedimiento alternativo

Dependiendo de la configuración, también es posible acceder a los tooltips a través de un objeto de manejo previsto para ello.

Ajustar el idioma del proyecto

Introducción

El proyecto del panel de operador puede ser multilingüe. Para cambiar el idioma ajustado en el panel de operador durante el funcionamiento, es preciso haber configurado un objeto a este efecto.

El proyecto se inicia siempre con el últimoidioma de configuración que se ha ajustado.

Requisitos

- El idioma deseado para el proyecto deberá estar disponible en el panel de operador.
- La función para cambiar de idioma se deberá haber vinculado a un objeto de manejo (p. ej. un botón) durante la configuración.

Seleccionar el idioma

Es posible conmutar en todo momento entre los idiomas. Inmediatamente después de ejecutarse la función de cambio de idioma, se mostrarán los objetos dependientes del idioma en el nuevo idioma.

El cambio de idioma se puede realizar de distintas maneras:

- Un objeto de manejo configurado conmuta a una selección de idiomas.
- Un objeto de manejo configurado permite seleccionar directamente el idioma deseado.

10.11.4.2 Manejar objetos

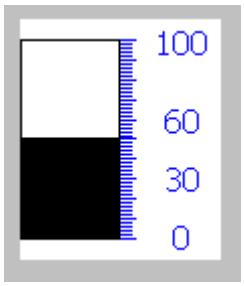
Barra

Utilización

La barra es un objeto de visualización dinámico. La barra representa un valor del controlador en forma de superficie rectangular. La barra permite apreciar fácilmente p. ej.:

- La distancia del valor actual hasta los valores límite configurados
- Si se ha alcanzado un valor teórico predeterminado

Con la barra se representa p. ej. el nivel de llenado o el número de piezas.



Representación

La representación de la barra depende de la configuración.

- La barra puede estar rotulada con una escala de valores.
- Los valores límite configurados se pueden señalar con líneas.
- El rebase por exceso o por defecto de un valor límite se puede señalar con un cambio de color.

Campo de fecha y hora

Visión de conjunto

Utilización

Un campo de fecha y hora puede tener las funciones siguientes en runtime:

- Indicación de hora y fecha
- Introducción y emisión combinadas. En este caso es posible editar los valores emitidos y cambiar así la fecha o la hora.

15/07/2009 15:35:07

Representación

La representación del campo de fecha y hora depende del idioma configurado en el panel de operador.

La fecha se puede visualizar en formato largo (p.ej. martes, 31 de diciembre de 2003) o corto (31/12/2003).

Manejo

Dependiendo de la configuración, el campo de fecha y hora se puede manejar como se indica a continuación:

- Manejo estándar: cambiar la fecha y la hora.

Comportamiento de manejo

Si el operador no respeta la sintaxis al introducir valores en runtime, o bien si introduce valores no válidos, estas introducciones no se adoptarán. En su lugar se visualizará de nuevo el valor original en el campo de fecha y hora (más el espacio de tiempo transcurrido entretanto) y se emitirá un aviso del sistema en el panel de operador.

Manejo con pantalla táctil y con teclado

Manejo con pantalla táctil

Para operar el campo de fecha y hora proceda del siguiente modo:

1. Toque el campo de fecha y hora en la pantalla táctil del panel de operador. Se abrirá automáticamente el teclado de pantalla.
2. Introduzca el valor deseado utilizando el teclado de pantalla.
3. Confirme la entrada en el panel de operador con <Intro> o rechácela con <Esc>. Una vez confirmada o rechazada la entrada, el teclado de pantalla se ocultará automáticamente.

Manejo con teclado

Active el campo de fecha y hora, p. ej., con uno o varios TAB, de acuerdo con el orden de tabulación configurado. La selección se señalizará con un cambio de color del contenido del campo.

Procedimiento

Para operar el campo de fecha y hora proceda del siguiente modo:

1. Posicione el cursor con las teclas de cursor e introduzca el valor deseado.
2. Pulse ENTER. El objeto pasa al modo de edición especial. Ahora sólo hay un carácter del campo marcado.
   - Desplácese por una tabla de caracteres utilizando las teclas de cursor ▲ / ▼.
   - Las teclas de cursor ◀ / ▶ le permiten cambiar a la posición siguiente o anterior de la entrada.
3. Confirme la entrada con ENTER o rechácela con ESC.

Campo E/S

Visión de conjunto

Utilización

En un campo E/S se introducen valores numéricos o alfanuméricos. Un valor numérico es, p. ej., el número 80 como consigna de una temperatura. Un valor alfanumérico es, p. ej., el texto "Servicio" como nombre de usuario.

Representación

La representación del campo E/S depende de la configuración.

- Campo E/S numérico
  Para introducir números en modo decimal hexadecimal o binario.
- Campo E/S alfanumérico
  Para introducir secuencias de caracteres.
- Campo E/S para fecha y hora
  Para entrar fechas de calendario o datos de tiempo. El formato depende del idioma del proyecto configurado.
- Campo E/S para introducir la contraseña
  Para introducir una contraseña de forma oculta. La secuencia de caracteres introducida se representa con asteriscos (*).

Manejo

Dependiendo de la configuración, el campo E/S se puede manejar como se indica a continuación:

- Manejo estándar: introduzca un valor en el campo E/S.
- Evento: al accionar el campo E/S (p. ej. activándolo) se dispara un evento. El evento puede estar asociado al procesamiento de una lista de funciones.

Comportamiento de manejo

Comprobar los límites de los valores numéricos

Para las variables se pueden configurar valores límite. Si se introduce un valor que está fuera de estos límites, no se aplicará, p. ej. 80 con un valor límite 78. En este caso, se emitirá un aviso de sistema en el panel de operador, siempre que se haya configurado una ventana de avisos. El valor original se visualizará de nuevo.

Decimales de los valores numéricos

El ingeniero puede definir el número de decimales de un campo de entrada numérico. Al introducir un valor en uno de estos campos E/S, se verificará el número de decimales.

- Si se introducen demasiados decimales, se ignorarán.
- Si se introducen demasiado pocos decimales, los que faltan se rellenarán con "0".

Entrada oculta

Durante la entrada oculta se muestra un "*" para cada carácter. El formato de datos del valor de entrada no es reconocible.

Comportamiento al conmutar entre los campos de entrada

Si al manejar una imagen se cambia de un campo de entrada a otro, visualizándose debido a ello el teclado de la pantalla, el evento "Abandonar campo" no se disparará de inmediato para el campo anterior, sino apenas después de haber cerrado el teclado de pantalla.

Manejo con pantalla táctil y con teclado

Manejo con pantalla táctil

Para manejar el campo E/S, proceda del siguiente modo:

1. Toque el campo E/S en la pantalla táctil del panel de operador. Se abrirá automáticamente el teclado de pantalla.
2. Introduzca el valor deseado utilizando el teclado de pantalla.
3. Confirme la entrada en el panel de operador con <Intro> o rechácela con <Esc>.

Una vez confirmada o rechazada la entrada, el teclado de pantalla se ocultará automáticamente.

Manejo con teclado

Active el campo E/S, p. ej., con uno o varios TAB, de acuerdo con el orden de tabulación configurado. La selección se señaliza con un cambio de color del contenido del campo.

Procedimiento

Para manejar el campo E/S, proceda del siguiente modo:

1. Posicione el cursor con SHIFT y una tecla de cursor.
2. La marca del contenido del campo se eliminará. Introduzca el valor deseado.

3. Pulse ENTER. El objeto pasa al modo de edición especial. Ahora sólo hay un carácter del campo marcado.
   – Desplácese por una tabla de caracteres utilizando las teclas de cursor ▲ / ▼.
   – Las teclas de cursor ▶ / ◀ le permiten cambiar a la posición siguiente o anterior de la entrada.
4. Confirme la entrada con ENTER o rechácela con ESC.

Nota

Para entrar los caracteres decimales de la "A" a la "F" para A-Z res numéricos, conmute las teclas de entrada a la asignación de letras con la tecla A-Z.

Visor de gráficos

Utilización

Este visor sirve para visualizar gráficos.

Visualización

La representación del gráfico depende de la configuración. Por ejemplo, el visor se adapta automáticamente al tamaño del gráfico.

Nota

Si se utilizan mapas de bits en WinCC con el ajuste "Color transparente", la representación en los paneles de operador del tipo Panel requerirá un rendimiento elevado. Para mejorar dicho rendimiento, desactive el ajuste "Color transparente" en las propiedades del objeto gráfico en cuestión. Esta limitación es aplicable sobre todo si los mapas de bits se utilizan como imágenes de fondo.

Manejo

El visor de gráficos sirve sólo para visualizar y no puede manejarse.

Campo E/S gráfico

Visión de conjunto

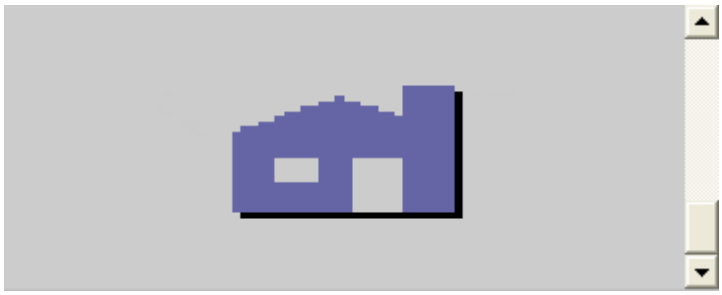
Utilización

Un campo E/S gráfico puede tener las funciones siguientes en runtime:

- Emisión de los registros de una lista de gráficos
- Introducción y emisión combinadas

Ejemplo de utilización como campo de salida:

Para indicar el estado de una válvula en runtime, en el campo E/S gráfico se representa una válvula cerrada o abierta.



Manejo

Dependiendo de la configuración, el campo E/S gráfico se puede manejar como se indica a continuación:

- Manejo estándar: Seleccione un registro en la lista de gráficos.
- Evento: al accionar el campo E/S gráfico (p. ej. activándolo) se dispara un evento. El evento puede estar asociado al procesamiento de una lista de funciones.

Comportamiento de manejo

Si en el campo E/S gráfico se visualiza un cactus, significa que en el proyecto no se ha definido ningún gráfico que se deba emitir para un valor determinado.

La activación se visualizará en el panel de operador mediante un cambio de color del contenido del campo E/S gráfico.

El marco en estilo tridimensional se representa gráficamente sólo en un campo de salida.

Manejo con pantalla táctil y con teclado

Manejo con pantalla táctil

Toque el campo E/S gráfico en la pantalla táctil del panel de operador. Se activará el modo de selección.

Seleccione el gráfico deseado con la barra de desplazamiento.

Aplique el gráfico deseado tocándolo o rechace la selección tocando otro objeto de imagen.

Manejo con teclado

Cómo manejar un campo E/S gráfico en el panel con teclado:

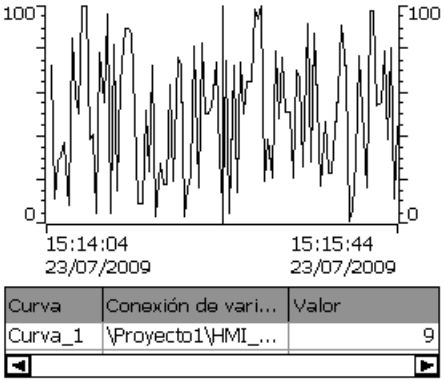
Paso		Procedimiento	
1	Seleccionar el campo E/S gráfico	p. ej. ▶	El campo E/S gráfico queda marcado.
2	Activar el modo de selección	ENTER	Se activará el modo de selección.
3	Seleccionar la entrada	▲ ▼ ◀ ▶	Mueve el cursor por filas.
4	Aplicar la selección o	ENTER	La entrada seleccionada se aplicará. Se cerrará el modo de selección.
	Cancelar selección	ESC	El valor original se reconstruirá de nuevo.

Visor de curvas

Visión de conjunto

Utilización

El visor de curvas es un objeto de visualización dinámico. En el visor de curvas se pueden representar de forma continua los datos de proceso actuales que se están ejecutando y, si es que el panel de operador lo soporta, visualizar los datos de proceso de un fichero.



Representación

La representación del visor de curvas depende de la configuración. En un visor de curvas pueden representarse varias curvas a la vez, para que el usuario pueda comparar diferentes transcursos de procesos, por ejemplo. Cuando el valor de proceso representado supera o no llega a los valores límite configurados, la violación de valor límite puede visualizarse por un cambio de color de la curva.

Asimismo, una regla facilita la lectura de valores de proceso en el visor de curvas. La regla muestra el valor de curva correspondiente a un valor X.

Manejo

Dependiendo de la configuración, es posible realizar las siguientes acciones:

- Aumentar el intervalo de tiempo representado.
- Reducir el intervalo de tiempo representado.
- Retroceder un ancho de visualización.
- Avanzar un ancho de visualización.
- Detener y reanudar la visualización de las curvas.

Elementos de control

Los botones tienen las funciones siguientes:

Elemento de control	Función
	Retrocede hasta el principio del registro de la curva. Allí se visualizan los valores iniciales con los que ha empezado el registro de curvas.
	Aumenta el intervalo de tiempo representado
	Reduce el intervalo de tiempo representado
	Desplaza la regla hacia atrás (hacia la izquierda).
	Desplaza la regla hacia adelante (hacia la derecha).
	Retrocede un ancho de visualización (hacia la izquierda).
	Avanza un ancho de visualización (hacia la derecha).
	Muestra u oculta la regla. La regla muestra el valor Y correspondiente a un valor X.
	Detiene o reinicia el registro de la curva

Manejo con pantalla táctil y con teclado

Procedimiento

Toque el elemento de control deseado del visor de curvas en la pantalla táctil del panel de operador.

Procedimiento

Active el visor de curvas, p. ej. con TAB siguiendo el orden de tabulación configurado.

La tabla siguiente muestra las posibles combinaciones de teclas:

Teclas	Función
CTRL ENTER	Retrocede hasta el principio del registro de la curva. Allí se visualizan los valores iniciales con los que ha comenzado el registro.
CTRL + +	Aumenta el intervalo de tiempo representado.
CTRL + –	Reduce el intervalo de tiempo representado.
CTRL + ALT + ◀	Desplaza la regla hacia atrás (hacia la izquierda).
CTRL + ALT + ▶	Desplaza la regla hacia adelante (hacia la derecha).
SHIFT + ◀	Retrocede un ancho de visualización (hacia la izquierda).
SHIFT + ▶	Avanza un ancho de visualización (hacia la derecha).

Botón

Visión de conjunto

Utilización

Un botón es una tecla virtual en la pantalla del panel de operador que puede estar ocupada con una o varias funciones.

Representación

La representación depende del tipo de botón.

- Botón con texto: el texto representado en el botón proporciona información sobre el estado de éste último.
- Botón con gráfico: el gráfico representado en el botón proporciona información sobre el estado éste último.
- Oculto: el botón no se visualiza en runtime.

Manejo

Dependiendo de la configuración, el botón se puede manejar como se indica a continuación:

- Manejo estándar: haga clic en el botón.
- Evento: al accionar el botón (p. ej. haciendo clic en él) se dispara un evento. El evento puede estar asociado al procesamiento de una lista de funciones.

Comportamiento de manejo

Tras manejar el botón puede aparecer una confirmación óptica. De todas formas, tenga en cuenta que la confirmación óptica sólo muestra el manejo realizado, y no si se han ejecutado realmente las funciones configuradas.

Manejo con pantalla táctil y con teclado

Procedimiento

Toque el botón en la pantalla táctil del panel de operador.

Procedimiento

Cómo manejar un botón en el panel con teclado:

1. Seleccione el botón con una tecla de cursor, p. ej. ▶.
2. Pulse a continuación la tecla ENTER o la tecla ␣.

Interruptor

Visión de conjunto

Utilización

El interruptor es un objeto de manejo y visualización con los estados "Pulsado" y "Soltado". Los interruptores pueden señalizar el estado de una parte de la instalación que no puede verse desde el panel de operador, p. ej. un motor. Al mismo tiempo, permiten modificar el estado de la parte de la instalación afectada en el panel de operador.

Representación

La representación depende del tipo de interruptor.

- Interruptor: el interruptor tiene un conmutador cuya posición indica el estado de aquel.
- Interruptor con texto: el texto representado en el interruptor proporciona información sobre el estado de éste último.
- Interruptor con gráfico: el gráfico representado en el interruptor proporciona información sobre el estado de éste último.

Manejo

Dependiendo de la configuración, el interruptor se puede manejar como se indica a continuación:

- Manejo estándar: haga clic en el interruptor.
- Evento: al accionar el interruptor (p. ej. haciendo clic en él) se dispara un evento. El evento puede estar asociado al procesamiento de una lista de funciones.

Comportamiento en caso de manejo

Un interruptor tiene dos estados estables: cuando se activa el interruptor, pasa al otro estado. Mantiene este estado hasta el próximo manejo.

Manejo con pantalla táctil y con teclado

Manejo con pantalla táctil

El manejo con pantalla táctil del interruptor es diferente según el tipo de interruptor:

- Si se visualiza un deslizador para el interruptor:
  Arrastre el deslizador por la pantalla táctil del panel de operador hasta la nueva posición o haga doble clic en el área del deslizador.
- Si sólo se visualiza un texto o un gráfico para el interruptor:
  Toque el interruptor en la pantalla táctil del panel de operador.

Manejo con teclado

Cómo manejar un interruptor en el panel con teclado:

- Seleccione el interruptor con una tecla de cursor, p. ej. ▶.
- Pulse a continuación la tecla ENTER o la tecla ␣.

Campo E/S simbólico

Visión de conjunto

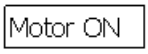
Utilización

Un campo E/S simbólico puede tener las funciones siguientes en runtime:

- Emisión de los registros de una lista de textos
- Introducción y emisión combinadas

Ejemplo de utilización como campo de entrada y salida combinado:

Para controlar un motor en runtime, seleccione en la lista uno de los textos "Motor ON" o "Motor OFF". El motor arrancará o parará conforme a la selección efectuada. El campo E/S simbólico muestra el estado actual del motor.



Manejo

Dependiendo de la configuración, el campo E/S simbólico se puede manejar como se indica a continuación:

- Manejo estándar: seleccione un registro en la lista de textos.
- Evento: al accionar el campo E/S simbólico (p. ej. activándolo) se dispara un evento. El evento puede estar asociado al procesamiento de una lista de funciones.

Comportamiento de manejo

Si el campo E/S simbólico contiene una línea de texto vacía en la lista desplegable, significa que en el proyecto no se ha definido ningún registro. La activación se visualizará en el panel de operador mediante un cambio de color del contenido del campo E/S simbólico.

Manejo con pantalla táctil y con teclado

Manejo con pantalla táctil

Toque el campo E/S simbólico en la pantalla táctil del panel de operador. Las entradas predefinidas aparecerán en la lista de selección.

Si la lista de selección tiene una barra de desplazamiento: toque la barra de desplazamiento en la pantalla táctil del panel de operador. Mueva la barra de desplazamiento en el sentido deseado tocando continuamente la pantalla táctil.

Seleccione la entrada deseada y aplique el valor correspondiente de la variable tocando dicha entrada en la pantalla táctil. La lista de selección se cerrará y la entrada se visualizará. El campo E/S simbólico sigue estando resaltado.

Manejo con teclado

Cómo manejar un campo E/S simbólico en el panel con teclado:

Paso		Procedimiento	
1	Seleccionar el campo E/S simbólico	p. ej. ▶	El campo E/S simbólico queda marcado.
2	Abrir la lista de selección	ENTER	La lista de selección se abrirá.
3	Seleccionar la entrada	▲ ◀ ▼ ▶	Mueve el cursor por filas.
4	Aplicar la selección O	ENTER	La entrada seleccionada se aplicará. El campo de selección se cerrará.
	Cancelar la selección	ESC	El valor original se reconstruirá de nuevo. El campo de selección se cerrará.

10.11.4.3 Seguridad en el proyecto

Visión de conjunto

Estructura del sistema de seguridad

El ingeniero puede proteger el proyecto mediante un sistema de seguridad.

El sistema de seguridad se basa en autorizaciones, grupos de usuarios y usuarios.

Si desea manejar un objeto con protección de acceso en el proyecto, deberá iniciar la sesión previamente en el panel de operador. A este efecto se visualiza un cuadro de diálogo de inicio de sesión en el que deberá introducir el nombre de usuario y la contraseña. Tras iniciar la sesión podrá manejar los objetos para los que disponga de las autorizaciones necesarias.

El ingeniero también puede configurar el acceso al cuadro de diálogo de inicio de sesión mediante un objeto de manejo.

Asimismo, puede configurar un objeto de manejo para cerrar la sesión. Tras cerrar la sesión ya no será posible manejar los objetos con protección de acceso, sino que deberá iniciar la sesión de nuevo.

Grupos de usuarios y permisos

El ingeniero crea los grupos de usuarios conforme al proyecto. Los grupos "Administradores" y "Usuarios" están contenidos por defecto en todo proyecto. Los grupos de usuarios tienen asignados permisos. En el proyecto está definido para cada objeto y para cada función, qué permiso es necesario para su manejo.

Usuarios y contraseñas

Cada usuario está asignado a un solo grupo de usuarios.

Las personas siguientes pueden crear usuarios y asignarles contraseñas:

- El ingeniero al crear la configuración
- El administrador en el panel de operador
- Un usuario autorizado para gestionar usuarios en el panel de operador

Independientemente del grupo de usuarios, todo usuario puede modificar su propia contraseña.

Tiempos de cierre de sesión

Para cada usuario se puede configurar un tiempo de cierre de sesión en el sistema. Si el tiempo transcurrido entre dos acciones cualquiera del usuario (p. ej. introducir un valor o cambiar de imagen) es superior al tiempo de cierre de sesión, el usuario será desconectado automáticamente. Si el usuario desea seguir manejando objetos con protección de acceso, deberá iniciar la sesión de nuevo.

Crear una copia de seguridad y restablecer datos

Los datos de los usuarios se codifican y se guardan en el panel de operador asegurados contra fallos de red.

Es posible crear una copia de seguridad de los datos de los usuarios, las contraseñas, las asignaciones a grupos y los tiempos de cierre de sesión creados en el panel de operador y

restablecer dichos datos posteriormente. De este modo evita tener que volver a introducir los datos en otro panel de operador.

Nota

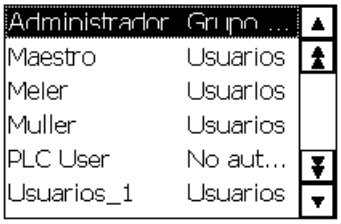
Los datos actuales de los usuarios se sobrescriben en los casos siguientes:

- Cuando el proyecto se cargue de nuevo (dependiendo de la configuración)
- Cuando se restablezca un proyecto del que se haya creado un backup
- Cuando se importe la gestión de usuarios mediante un objeto de manejo. Los datos de los usuarios y sus respectivas contraseñas que se carguen de nuevo o que se restablezcan tendrán validez inmediata.

Visor de usuarios simple

Utilización

En paneles de operador con pantalla pequeña, el visor de usuarios simple se utiliza para visualizar los usuarios.



Nota

El objeto "Visor de usuarios simple" no se puede dinamizar mediante un script.

Representación

La representación depende de las autorizaciones.

- En el visor de usuarios del administrador o de un usuario autorizado para gestionar usuarios se muestran todos los usuarios existentes en el panel de operador.
- Un usuario que no está autorizado para gestionar usuarios sólo podrá apreciar su propia entrada de usuario.

Manejo

Dependiendo de la configuración, es posible realizar las siguientes acciones:

- Gestionar usuarios (p.ej. crear o borrar usuarios)
- Modificar los datos de los usuarios existentes.
- Exportar o importar datos de usuarios.

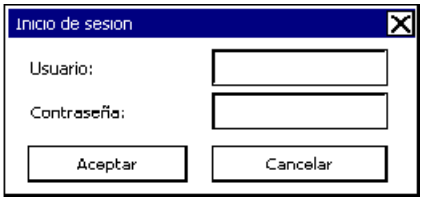
Nota

Un panel de operador puede ser utilizado por 100 usuarios y un "PLC User" como máximo. Este límite no es aplicable a los PCs. El número máximo de usuarios de un PC está limitado por la memoria física del mismo.

Iniciar una sesión de usuario

Cuadro de diálogo de inicio de sesión

Para iniciar la sesión en el sistema de seguridad del panel de operador, utilice el cuadro de diálogo de inicio de sesión. Introduzca su nombre de usuario y su contraseña en el cuadro de diálogo de inicio de sesión.



El cuadro de diálogo de inicio sesión se abre en los casos siguientes:

- Al desear manejar un objeto con protección de acceso.
- Al manejar un objeto configurado para visualizar el cuadro de diálogo de inicio de sesión.
- Al activar la entrada "<ENTER>" en el visor de usuarios simple.
- Al activar una entrada vacía en el visor de usuarios avanzado.
- Dependiendo de la configuración, el cuadro de diálogo de inicio de sesión se visualiza automáticamente al abrir un proyecto.

Requisitos

El cuadro de diálogo de inicio de sesión deberá estar abierto.

Procedimiento en paneles con pantalla táctil

Proceda del siguiente modo:

1. Introduzca su nombre de usuario y su contraseña.
   A este efecto, toque el campo de entrada en cuestión. El teclado de pantalla alfanumérico se mostrará.
2. Confirme el inicio de sesión pulsando el botón "Aceptar".

Procedimiento en paneles con teclado

Proceda del siguiente modo:

1. Pulsando la tecla TAB, seleccione el campo de entrada "Usuario" en el cuadro de diálogo de inicio de sesión.
2. Introduzca el nombre de usuario utilizando las teclas del sistema.
   Para introducir letras, conmute el teclado numérico mediante la tecla A-Z a la asignación de letras.
3. Pulsando la tecla TAB, seleccione el campo de entrada "Contraseña".
4. Introduzca la contraseña utilizando las teclas del sistema.
5. Confirme los ajustes efectuados haciendo clic en "Aceptar".

Nota

Al introducir el nombre de usuario no se distingue entre mayúsculas y minúsculas.

Al introducir la contraseña sí se distingue entre mayúsculas y minúsculas.

Resultado

Tras iniciar la sesión en el sistema de seguridad, podrá ejecutar en el panel de operador las funciones con protección de acceso para las que tenga la autorización necesaria.

Si introduce una contraseña incorrecta se visualizará un aviso de error (si se ha configurado una ventana de avisos).

Cerrar una sesión de usuario

Requisito

Debe haber iniciado una sesión en el sistema de seguridad del panel de operador.

Procedimiento

Existen dos posibilidades de cerrar una sesión:

- Accione el objeto de manejo configurado para cerrar la sesión.
- Si no maneja el proyecto y se excede el tiempo de cierre de sesión, se cerrará automáticamente su sesión.

Si introduce una contraseña incorrecta, también se cerrará automáticamente su sesión.

Resultado

Su sesión se cerrará en el proyecto. Para manejar un objeto con protección de acceso deberá iniciar la sesión de nuevo.

Crear un usuario

Requisitos

- El visor de usuarios está abierto.
- Debe estar autorizado para gestionar usuarios o tener derechos de administrador.
- Se ha creado un grupo de usuario.

Nota

En runtime debe asignarse un usuario a un grupo de usuarios. El grupo de usuarios se crea en el sistema de ingeniería. La denominación del grupo de usuarios depende del idioma.

Nota

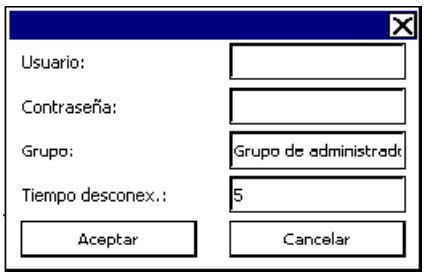
En la contraseña no puede utilizar los siguientes caracteres:

- Espacio
- Caracteres especiales * ? . % / \ ' "

Crear usuarios en el visor de usuarios simple

Proceda del siguiente modo:

1. En el visor de usuarios, toque la entrada "<Nuevo usuario>". Se abrirá un cuadro de diálogo.



2. Introduzca el nombre de usuario deseado y una contraseña.
   A este efecto, toque el campo de entrada en cuestión. El teclado de pantalla alfanumérico se visualizará.
3. Haga clic en el campo de entrada del grupo. Se abrirá un cuadro de diálogo.
4. Asigne el usuario a un grupo. Utilice los botones ▲ y ▼ para desplazarse por la lista.

5. Toque el registro deseado en la lista de selección.
   El registro seleccionado se aplicará en el campo de entrada.
6. Toque el campo de entrada "Tiempo de cierre de sesión". Se abrirá el teclado de pantalla.
7. Introduzca un valor comprendido entre 0 y 60 minutos para el tiempo de cierre de sesión.
   Si introduce el valor 0, significa que la sesión no debe cerrarse automáticamente.
8. Confirme los ajustes efectuados haciendo clic en "Aceptar".

Resultado

Se habrá creado un nuevo usuario.

Modificar un usuario

Requisitos

El visor de usuarios está abierto.

Sus derechos determinan los datos que puede modificar:

- Si tiene derechos de administrador o si está autorizado para gestionar usuarios, podrá modificar en el visor de usuarios los datos de todos los usuarios existentes en el panel de operador:
  - Nombre del usuario
  - Asignación a grupos
  - Contraseña
  - Tiempo de cierre de sesión
- Si no está autorizado para gestionar usuarios, sólo podrá modificar sus propios datos de usuario:
  - Contraseña
  - Tiempo de cierre de sesión (si se ha previsto en la configuración)

Nota

Para el usuario "Admin" sólo es posible modificar el tiempo de cierre de sesión y la contraseña.

Para el usuario "PLC_User" sólo es posible modificar el tiempo de cierre de sesión. Este usuario se necesita para iniciar la sesión a través del controlador.

Nota

Las modificaciones en el visor de usuarios de runtime son efectivas inmediatamente. Las modificaciones en runtime no se actualizan en el sistema de ingeniería.

Al cargar la administración de usuarios en el panel de operador, se sobrescriben todas las modificaciones del visor de usuarios.

Modificar los datos de usuarios en el visor de usuarios simple

Proceda del siguiente modo:

1. En el visor de usuarios, toque el usuario cuyos datos desea modificar.
2. Para modificar los datos, proceda de la misma manera que al crear un usuario.

Modificar los datos de usuarios en el visor de usuarios avanzado

Proceda del siguiente modo:

1. En el visor de usuarios, toque los datos del usuario que desea modificar.
2. Para modificar los datos, proceda de la misma manera que al crear un usuario.

Resultado

Los datos del usuario se habrán modificado.

Borrar un usuario

Requisitos

- Hay una imagen abierta que contiene el visor de usuarios.
- Si desea borrar usuarios deberá tener derechos de administrador, o bien la autorización para gestionar usuarios.

Nota

Las modificaciones en el visor de usuarios de runtime son efectivas inmediatamente. Las modificaciones en runtime no se actualizan en el sistema de ingeniería.

Al cargar la administración de usuarios en el panel de operador, se sobrescriben todas las modificaciones del visor de usuarios.

Procedimiento en paneles con pantalla táctil

Proceda del siguiente modo:

1. En el visor de usuarios, toque el usuario que desea borrar.
2. Borre el nombre de usuario.

Procedimiento en paneles con teclado

Proceda del siguiente modo:

1. Seleccione el visor de usuarios con la tecla TAB o con las teclas de cursor.
2. Seleccione el usuario en el visor de usuarios con las teclas de cursor.
3. Para borrar el usuario pulse la tecla INS DEL.

Resultado

El usuario se borrará y ya no podrá iniciar la sesión en el proyecto.

10.11.4.4 Manejar avisos

Visión de conjunto

Avisos

Los avisos indican en el panel de operador eventos y estados que se presentan en la instalación, en el proceso o bien en el panel de operador. Al ocurrir un estado, éste se notifica.

En el caso de los avisos pueden ocurrir los siguientes eventos:

- Aparecer
- Desaparecer
- Acusar

El ingeniero determina qué avisos deben ser acusados por el usuario.

Un aviso puede contener las informaciones siguientes:

- Fecha
- Hora
- Texto de aviso
- Texto de evento
- Ubicación del fallo
- Estado
- Categoría
- Número de aviso
- Grupo de avisos
- Diagnosticable

Categorías

Los avisos están asignados a distintas categorías. La selección depende del panel de operador.

- "Warnings"
  Por lo general, los avisos de esta categoría indican estados en la instalación, p. ej. "Motor encendido". Los avisos de esta categoría no deben acusarse.
- "Errors"
  Los avisos de esta categoría siempre deben ser acusados. Por lo general, las alarmas indican estados críticos en la instalación, p. ej. " Temperatura del motor demasiado elevada".

- "System"
  Los avisos de sistema indican estados o eventos del panel de operador.
  Los avisos del sistema informan p. ej. acerca de operaciones erróneas o de averías en la comunicación.
- "Diagnosis Events"
  Los avisos de diagnóstico SIMATIC muestran los estados y eventos de los controladores SIMATIC S7.

Nota

Dependencia del equipo

Los avisos de diagnóstico no están disponibles para Basic Panels.

- Categorías en STEP 7
  Las categorías configuradas en STEP 7 están disponibles también en el panel de operador.

  Nota

  Dependencia del equipo

  Las categorías de STEP 7 no están disponibles para Basic Panels.

- Categorías personalizadas
  Las propiedades de esta categoría se definen durante la configuración.

Búfer de avisos

Los eventos de avisos se guardan en un búfer interno. El tamaño de este búfer de avisos depende del tipo de panel de operador.

Informe de avisos

Si está activada la generación de informes de avisos, los eventos de aviso se imprimirán directamente en la impresora conectada.

Para cada aviso se puede configurar por separado si debe imprimirse o no. La impresión de un aviso de este tipo se inicia cuando ocurren los eventos "Aparecer" y "Desaparecer".

En caso de que se deban imprimir avisos de la categoría "Sistema", habrá que hacerlo desde el búfer de avisos correspondiente. En este caso se imprimirá todo el contenido del búfer de avisos. Para ello se tiene que haber configurado en el proyecto un objeto de manejo que inicie la impresión.

Nota

Dependencia del equipo

Los informes de avisos no están disponibles para Basic Panels.

Fichero de avisos

En caso de haber configurado un fichero de avisos, los eventos de aviso también se guardarán en este fichero. La capacidad del fichero está limitada por el medio de almacenamiento y por los límites del sistema.

Nota

Dependencia del equipo

Los ficheros de avisos no están disponibles para Basic Panels.

Visor de avisos

En el visor de avisos se visualizan los avisos o eventos de avisos seleccionados del búfer o del fichero de avisos. En la configuración se define si es necesario acusar los avisos que van apareciendo. La visualización se puede configurar de manera que aparezcan sólo los avisos que contengan en el texto una cadena de caracteres determinada.

Ventana de avisos

La ventana de avisos (si está configurada) muestra todos los avisos pendientes o por acusar de la categoría pertinente. La ventana de avisos se abre cuando entra un aviso nuevo.

El orden de los avisos visualizados puede configurarse. Se puede elegir entre visualizar primero el aviso actual o el más antiguo. Asimismo, en la ventana de avisos puede visualizarse el lugar exacto del fallo con fecha y hora del evento de aviso. La visualización se puede configurar de manera que aparezcan sólo los avisos que contengan en el texto una cadena de caracteres determinada.

Indicador de avisos

El indicador de avisos es un símbolo gráfico que se visualiza en la pantalla cuando aparece un aviso de la categoría determinada.

El indicador de avisos puede tener dos estados:

- Intermitente: Hay como mínimo un aviso pendiente de acuse.
- Estático: Los avisos se han acusado, pero al menos uno de ellos no ha desaparecido todavía. El número que aparece indica la cantidad de avisos que todavía están pendientes.

Visor de avisos simple, ventana de avisos

Utilización

En el visor de avisos simple se muestran los avisos o eventos de avisos seleccionados del búfer o del fichero de avisos. La apariencia y el manejo de la ventana de avisos simple son similares a los del visor de avisos simple.

Nota

El objeto "Visor de avisos simple" no se puede dinamizar mediante un script.

En la ventana de propiedades del grupo "Animaciones" del sistema de ingeniería se puede p. ej. dinamizar la visibilidad de un objeto. El objeto "Visor de avisos simple" no soporta ninguna animación en runtime. Si ha configurado una animación y comprueba, p. ej., la coherencia del proyecto, se mostrará un mensaje de error en la ventana de resultados.

Representación

Dependiendo de la configuración, en el visor de avisos se muestran diversas columnas con información acerca de un aviso o de un evento de aviso.

Con objeto de distinguir las diferentes categorías, la primera columna del visor de avisos contiene un símbolo:

Símbolo	Categoría
!	"Errors"
(vacío)	"Warnings"
(en función de la configuración)	Categorías personalizadas
$	"System"

Manejo

Dependiendo de la configuración, es posible realizar las siguientes acciones:

- Acusar avisos
- Editar avisos

Elementos de control

Los botones tienen las funciones siguientes:

Botón	Función
!	Acusar un aviso
↲	Editar un aviso
?	Mostrar el texto de ayuda de un aviso
▶	Mostrar el texto completo del aviso seleccionado en una ventana independiente (es decir, en la ventana de avisos). En la ventana de avisos pueden mostrarse los textos de avisos que excedan el espacio disponible en el visor de avisos. Cierre la ventana de avisos con ✕.
▲	Retrocede un aviso hacia arriba
⇞	Retrocede una página hacia arriba en el visor de avisos
⇟	Avanza una página hacia abajo en el visor de avisos
▼	Avanza un aviso hacia abajo

Representación de los elementos de control

La representación de los botones de manejo en el visor de avisos simple depende del tamaño configurado. Por ello, compruebe en el panel de operador si están disponibles todos los botones necesarios.

Detectar los avisos pendientes

Introducción

La existencia de avisos de acuse obligatorio se reconoce por lo siguiente:

- En paneles de operador con teclado: se ilumina el LED correspondiente a la tecla ACK.
- Dependiendo de la configuración: en la pantalla se visualiza un indicador de avisos.

De la configuración depende si un aviso se debe acusar o no. La obligación de acuse se ve determinada por la clase a la que pertenece un aviso.

LED de la tecla "ACK"

En los paneles de operador con teclado, la tecla ACK incorpora un LED. El LED se enciende si hay avisos de acuse obligatorio que no se hayan acusado todavía.

El LED se apagará tras haberse acusado todos los avisos de acuse obligatorio.

Indicador de avisos

El indicador de avisos es un símbolo gráfico que, dependiendo de la configuración, puede indicar avisos pendientes o que deban acusarse.

Figura 10-4 Indicador de avisos con tres avisos pendientes

Representación

El indicador de avisos puede tener dos estados:

- Intermitente:
  Mientras haya avisos sin acusar, el indicador de avisos seguirá parpadeando. El número que aparece indica la cantidad de avisos que todavía están pendientes. El ingeniero puede configurar funciones que deban ejecutarse al manejar el indicador de avisos.
- Estático: Los avisos se han acusado, pero al menos uno de ellos no ha desaparecido todavía.

Mostrar cuadros de diálogo

El indicador de avisos queda oculto p. ej. por el cuadro de diálogo de inicio de sesión, por la ayuda así como por las ventanas de avisos. El indicador de avisos se visualiza de nuevo cuando se cierran dichos cuadros.

Mostrar el texto de ayuda de un aviso

Procedimiento en paneles con pantalla táctil

Para visualizar el texto de ayuda, proceda del siguiente modo:

1. Seleccione el aviso deseado en la ventana de avisos o en el visor de avisos.
   El aviso se aparecerá seleccionado.
2. Toque el botón ? en el visor de avisos simple, o bien el botón en el visor de avisos avanzado.
   Si se ha configurado un texto de ayuda para el aviso, se visualizará dicho texto.
3. Pulse el botón ☒ para cerrar la ventana en la que se visualiza el texto de ayuda.

Procedimiento en paneles con teclado

Para visualizar el texto de ayuda, proceda del siguiente modo:

1. Seleccione el aviso deseado en el visor de avisos.
2. Pulse la tecla HELP.
   Si se ha configurado un texto de ayuda para el aviso, se visualizará dicho texto.
3. Cierre el texto de ayuda pulsando la tecla HELP.

Acusar un aviso

Requisitos

El aviso de acuse obligatoria se deberá visualizar en la ventana de avisos o en el visor de avisos.

Procedimiento en paneles con pantalla táctil

Para acusar un aviso, proceda del siguiente modo:

1. Seleccione el aviso deseado en la ventana de avisos o en el visor de avisos.
   El aviso quedará seleccionado.
2. Toque el botón ! en el visor de avisos simple, o bien el botón en el visor de avisos avanzado.

Procedimiento en paneles con teclado

En el visor de avisos o en la ventana de avisos existe un orden de tabulación que permite activar mediante el teclado los elementos de control y el aviso seleccionado de último.

Para acusar un aviso, proceda del siguiente modo:

1. Pulsando la tecla TAB, seleccione el visor de avisos o la ventana de avisos deseada.
2. Seleccione el aviso deseado. Utilice para ello las teclas HOME, END, ▲ o ▼.
3. Pulse la tecla ACK.

Manejo alternativo

Dependiendo de la configuración, el aviso también se puede acusar mediante una tecla de función.

Resultado

El aviso se acusará. Si el aviso pertenece a un grupo, se acusarán todos los avisos que conforman dicho grupo.

Editar un aviso

Introducción

El ingeniero puede configurar funciones adicionales para cada aviso. Estas funciones se ejecutan en el momento de editar el aviso.

Nota

Al editar un aviso no acusado todavía, éste se acusará automáticamente.

Requisitos

El aviso a editar se deberá visualizar en la ventana de avisos o en el visor de avisos.

Procedimiento en paneles con pantalla táctil

Para editar un aviso, proceda del siguiente modo:

1. Seleccione el aviso deseado en la ventana de avisos o en el visor de avisos. El aviso se aparecerá seleccionado.
2. Toque el botón ↲ en el visor de avisos simple, o bien el botón en el visor de avisos avanzado.

Procedimiento en paneles con teclado

Para editar un aviso, proceda del siguiente modo:

1. Pulsando la tecla TAB, seleccione el visor de avisos o la ventana de avisos deseada.
2. Seleccione el aviso deseado. Utilice para ello las teclas HOME, END, ▲ o ▼.
3. Pulse la tecla TAB tantas veces hasta que aparezca seleccionado el botón ↲ en el visor de avisos simple, o bien el botón en el visor de avisos avanzado.
4. Realice la acción pulsando la tecla ENTER.

Resultado

Se ejecutarán las funciones adicionales para el aviso. Para más información al respecto, consulte la documentación de su instalación.

10.11.4.5 Manejar recetas

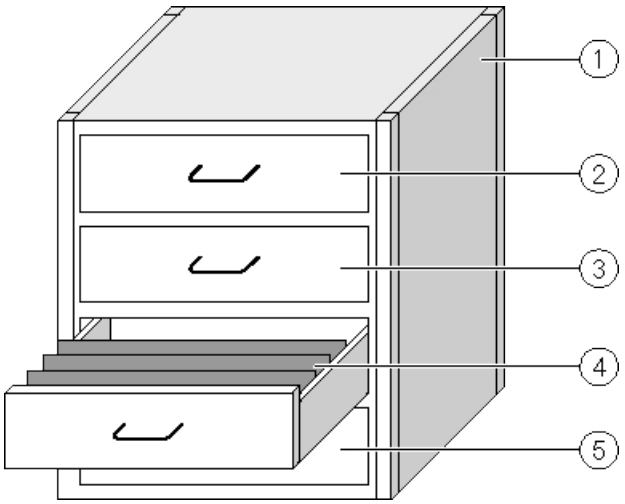
Estructura de una receta

Recetas

La colección de recetas para fabricar una gama de productos es comparable con un armario archivador. Una receta para fabricar un producto equivale a un cajón de dicho armario.

Ejemplo:

En una planta de fabricación de bebidas se utilizan diversas recetas para los distintos sabores. Por ejemplo, existen sendas recetas para las bebidas con sabor a naranja, uva, manzana y cereza.



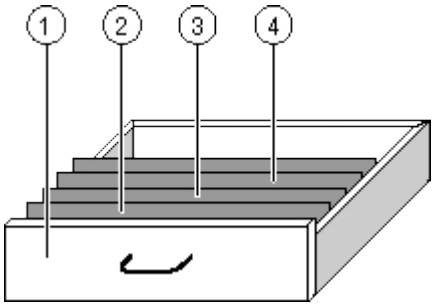
① Armario archivador	Colección de recetas	Recetas de una planta de zumos de fruta
② Cajón	Receta	Bebidas con sabor a naranja
③ Cajón	Receta	Bebidas con sabor a uva
④ Cajón	Receta	Bebidas con sabor a manzana
⑤ Cajón	Receta	Bebidas con sabor a cereza

Registros de receta

Los cajones del armario archivador contienen carpetas colgantes. Estas carpetas representan los registros necesarios para fabricar las diversas variantes del producto.

Ejemplo:

Las variantes de la bebida con sabor a manzana son p. ej. refresco, zumo y néctar.



①	Cajón	Receta	Variantes de la bebida con sabor a manzana
②	Carpeta colgante	Registro de receta	Refresco de manzana
③	Carpeta colgante	Registro de receta	Néctar de manzana
④	Carpeta colgante	Registro de receta	Zumo de manzana

Elementos

En la imagen que muestra el armario archivador, todas las carpetas colgantes contienen un número idéntico de hojas. Cada hoja de una carpeta colgante representa un elemento del registro de receta. Todos los registros de una receta contienen idénticos elementos. No obstante, los registros se diferencian en el valor de los elementos individuales.

Ejemplo:

Todas las bebidas contienen idénticos ingredientes, a saber: agua, concentrado, azúcar y aroma. No obstante, los registros correspondientes a las variantes "Refresco", "Zumo" y "Néctar" se diferencian en la cantidad de azúcar utilizada para su fabricación.

Recetas en el proyecto

Resumen

Los componentes siguientes interaccionan al utilizarse recetas en un proyecto:

- Memoria de recetas del panel de operador
  Las recetas se guardan en forma de registros en la memoria de recetas del panel de operador.
- Visor de recetas / imagen de receta
  En el panel de operador, las recetas de visualizan y se editan en el visor de recetas o en una imagen de receta.
  - Los registros de recetas se visualizan y se editan en el visor de recetas desde la memoria interna del panel de operador.
  - Los valores de las variables de una receta se visualizan y se editan en la imagen de receta.

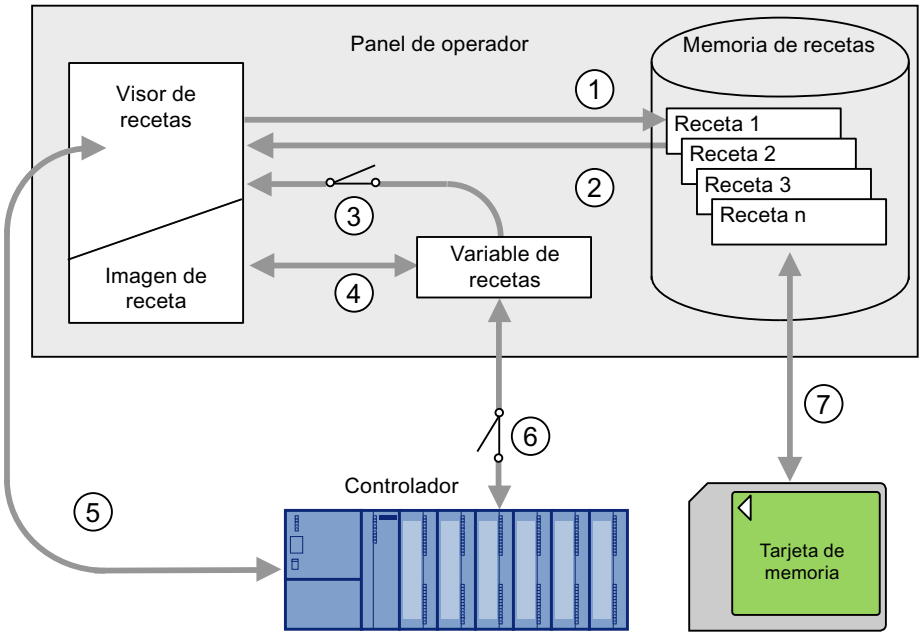
  Nota

  Una misma variable de receta puede estar configurada en diversas recetas. Si modifica el valor de una variable de receta, el valor de esa variable cambiará debido a la sincronización también en todas las recetas.

- Variables de receta
  Las variables de receta contienen datos de recetas. Al editar una receta en una imagen de receta, los valores de la receta se guardan en variables. La configuración determina cuándo se deben intercambiar los valores de las variables de receta con el controlador.

Flujo de datos

La figura siguiente muestra el flujo de datos en un proyecto con recetas:



① Editar, guardar o borrar un registro de receta.
② Visualizar un registro de receta.
③ Sincronizar (o no) variables de receta.
④ Visualizar y editar variables de receta en la imagen de receta.
⑤ Escribir registros del visor de recetas en el controlador, o bien leer registros del controlador y visualizarlos en el visor de recetas.
⑥ Las variables de receta pueden estar online u offline con el controlador.
⑦ Exportar o importar un registro de receta a / de la tarjeta de memoria.

Visor de recetas simple

Visualización

El visor de recetas simple comprende tres áreas de visualización, a saber:

- Lista de recetas
- Lista de registros
- Lista de elementos

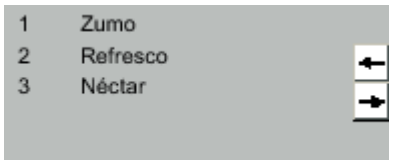


Figura 10-5 Visor de recetas simple (ejemplo con la lista de registros)

El visor de recetas simple representa cada una de estas áreas por separado en el panel de operador. Todas estas áreas de visualización se manejan mediante un menú contextual.

Por defecto se visualiza primero el visor de recetas simple con la lista de recetas.

Manejo

Dependiendo de la configuración, el visor de recetas simple se puede manejar como se indica a continuación:

- Crear, modificar, copiar o borrar registros de recetas
- Leer un registro de recetas del controlador o transferirlo al controlador

Elemento de control del visor de recetas simple

Para manejar el visor de recetas simple, conmute entre las áreas de visualización y los menús contextuales.

La tabla siguiente muestra las posibilidades de manejo del área de visualización.

	Tecla	Función
Tocar una entrada	ENTER	Abre la siguiente área de visualización subordinada, es decir, la lista de registros o la lista de elementos.
←	ESC	Abre el área de visualización anterior.
→	▶	Abre el menú contextual del área de visualización.

La tabla siguiente muestra las posibilidades de manejo del menú contextual.

	Tecla	Función
→	ESC	Cierra el menú. Abre el área de visualización.
Tocar el comando de menú	Introducir el número del comando de menú	Ejecuta el comando de menú.

Menús contextuales del visor de recetas simple

En cada área de visualización puede llamarse un menú contextual con el botón → o la tecla ▶. Dicho menú contextual contiene los comandos disponibles para el área de visualización actual. Cada comando tiene un número asignado. Si se introduce el número del comando, se ejecutará dicho comando. Alternativamente, en algunos comandos se pueden utilizar las teclas de sistema.

El alcance depende del panel de operador.

- Lista de recetas

N.º	Comando	Teclas	Función
0	Nuevo	SHIFT + INS DEL	Permite crear un nuevo registro para la receta seleccionada. Si se ha configurado un valor inicial, éste se visualizará en el campo de entrada.
1	Visualizar el texto de ayuda	HELP	Visualiza el texto de ayuda configurado para el visor de recetas simple.
2	Abrir	ENTER	Abre la lista de registros de la receta seleccionada.

- Lista de registros

	Comando	Teclas	Función
	Nuevo	SHIFT + INS DEL	Permite crear un nuevo registro de receta. Si se ha configurado un valor inicial, éste se visualizará en el campo de entrada.
	Borrar	INS DEL	Borra el registro seleccionado.
	Guardar como		Guarda el registro seleccionado con un nombre diferente. El nombre se introduce en un cuadro de diálogo.
	Cambiar nombre		Permite cambiar el nombre del registro seleccionado. El nombre se introduce en un cuadro de diálogo.
	Abrir	ENTER	Abre la lista de elementos del registro seleccionado.
	Atrás	ESC	Abre la lista de recetas.
	Al controlador		Los valores visualizados del registro seleccionado se transfieren del panel de operador al controlador.
	Del controlador		Los valores de receta del controlador se visualizan en el visor de recetas del panel de operador.
	Visualizar el texto de ayuda	HELP	Visualiza el texto de ayuda configurado para el visor de recetas simple.

- Lista de elementos

	Comando	Teclas	Función
	Guardar		Permite guardar el registro seleccionado.
	Al controlador		Los valores visualizados del registro seleccionado se transfieren del panel de operador al controlador.
	Del controlador		Los valores de receta del controlador se visualizan en el visor de recetas del panel de operador.
	Guardar como		Guarda el registro seleccionado con un nombre diferente. El nombre se introduce en un cuadro de diálogo.
	Visualizar el texto de ayuda	HELP	Visualiza el texto de ayuda configurado para el visor de recetas simple.
	Cambiar nombre		Permite cambiar el nombre del registro seleccionado. El nombre se introduce en un cuadro de diálogo.
	Atrás	ESC	Abre la lista de registros.

Manejar el visor de recetas simple utilizando el ratón o el panel táctil

1. Seleccione la receta deseada en el visor de recetas.
2. Pulse el botón →.
   Se abrirá el menú contextual.
3. Elija el comando de menú deseado.
   Se ejecutará el comando de menú deseado.
4. Alternativamente, puede abrir la receta deseada en el visor de recetas.
   Se visualizará la lista de registros.
5. Abra el registro deseado. Alternativamente, pulse el botón → para abrir el menú contextual y elija un comando de menú.
   Se ejecutará el comando de menú deseado.

Manejar el visor de recetas simple utilizando el teclado

1. Pulse la tecla TAB hasta que aparezca destacado el visor de recetas simple.
2. Seleccione la receta deseada utilizando las teclas con flecha.
3. Pulse la tecla ▶.
   Se abrirá el menú contextual.
4. Pulse la tecla de cursor ▼ hasta que aparezca seleccionado el comando de menú deseado.
5. Confirme el comando de menú pulsando la tecla ENTER.
6. Alternativamente, puede pulsar el número correspondiente al comando de menú deseado.
   Se ejecutará el comando de menú deseado.

Crear un registro de receta

Introducción

Los registros de receta se crean en la lista de recetas, o bien en la lista de registros. A continuación, los valores del nuevo registro se introducen en la lista de elementos y se guarda el registro.

Requisito

Se visualiza una imagen con un visor de recetas simple.

Procedimiento

Para crear un registro de receta, proceda del siguiente modo:

1. Si la lista de recetas contiene varias recetas: Seleccione la receta para la que desea crear un nuevo registro de receta.
2. Abra el menú de la lista de recetas.
3. Elija el comando de menú "0 Nuevo".
   Se crea un registro nuevo.
   Se abrirá la lista de elementos del nuevo registro.
4. Introduzca los valores para los elementos del registro.
   Dependiendo de la configuración, las variables del registro pueden tener valores predeterminados.
5. Abra el menú de la lista de elementos y elija el comando "0 Guardar".
6. Introduzca un nombre para el nuevo registro.
7. Confirme las introducciones efectuadas.
   Si cambia el nuevo número de registro por uno ya existente, se sobrescribirá el registro existente.

Resultado

El nuevo registro de receta se habrá guardado en la receta seleccionada.

Editar un registro de receta

Introducción

Los valores de los registros de recetas se editan y se guardan en un visor de recetas.

Sincronización con el controlador

Si desea visualizar los valores de receta actuales del controlador en el visor de recetas simple, deberá leer primero los valores actuales del controlador en la lista de elementos con el comando de menú "2 desde el controlador".

Los valores modificados en el visor de recetas no tendrán efecto en el controlador hasta haberse transferido el registro modificado al controlador con el comando de menú "1 hacia el controlador".

Requisitos

Se deberá visualizar una imagen con un visor de recetas.

Procedimiento

Para editar un registro de receta, proceda del siguiente modo:

1. Si la lista de recetas contiene varias recetas: Seleccione la receta que contiene el registro deseado.
2. Abra la lista de registros.
3. Seleccione el registro de receta que desea modificar.
4. Abra la lista de elementos.
5. Modifique los valores de los elementos según desee.
6. Guarde los cambios efectuados con el comando de menú "0 Guardar".

Resultado

El registro de receta modificado se guardará en la receta seleccionada.

Borrar un registro de receta

Introducción

Es posible borrar los registros que no se necesiten más.

Requisitos

Se visualiza una imagen con un visor de recetas simple.

Procedimiento en paneles con pantalla táctil

Para borrar un registro de receta nuevo, proceda del siguiente modo:

1. Si la lista de recetas contiene varias recetas: Seleccione la receta que contiene el registro deseado.
2. Abra la lista de registros.
3. Seleccione el registro que desea borrar.
4. Abra el menú.
5. Elija el comando de menú "1 Borrar".

Procedimiento en paneles con teclado

Para borrar un registro de receta nuevo, proceda del siguiente modo:

1. Si la lista de recetas contiene varias recetas: Seleccione la receta que contiene el registro deseado.
2. Abra la lista de registros.
3. Seleccione el registro que desea borrar.
4. Pulse la tecla INS DEL.

Resultado

El registro se habrá borrado.

Leer un registro de receta del controlador

Introducción

Los valores de los elementos de receta se intercambian con el controlador a través de variables.

En el proyecto que se está ejecutando es posible modificar directamente en la instalación los valores depositados también en las recetas en el panel de operador. Éste es el caso p. ej. cuando una válvula se abre directamente en la instalación más de lo especificado en la receta. En este caso, es posible que los valores de las variables guardadas en el panel de operador ya no concuerden con los valores del controlador.

Para sincronizar los valores de recetas, lea los valores del controlador y visualícelos en el visor de recetas.

Requisitos

Se visualiza una imagen con un visor de recetas simple.

Procedimiento

Para transferir un registro de receta al controlador, proceda del siguiente modo:

1. Si la lista de recetas contiene varias recetas: Seleccione la receta que contiene el registro deseado.
2. Abra la lista de elementos del registro de receta en el que desea aplicar los valores del controlador.
3. Abra el menú.
4. Elija el comando de menú "2 desde el controlador".
   Los valores se leerán del controlador.
5. Si desea guardar los valores visualizados en el panel de operador, elija el comando de menú "0 Guardar".

Resultado

Los valores se habrán leído del controlador, se visualizarán en el panel de operador y quedarán almacenados en el registro de receta seleccionado.

Nota

Basic Panels

En los Basic Panels, el comando de menú "desde el controlador" también puede estar configurado para la lista de registros. En este caso también puede seleccionar el comando de menú "desde el controlador" en la lista de registros.

Transferir un registro de receta al controlador

Introducción

Para que un registro de receta modificado tenga efecto en el proyecto es preciso transferir los valores al controlador.

Los valores visualizados en el visor de recetas son los que se transfieren al controlador.

Requisitos

Se visualiza una imagen con un visor de recetas simple.

Procedimiento

Para transferir un registro de receta al controlador, proceda del siguiente modo:

1. Si la lista de recetas contiene varias recetas: Seleccione la receta que contiene el registro deseado.
2. Abra la lista de elementos del registro de receta cuyos valores desea transferir al controlador.
3. Abra el menú.
4. Elija el comando de menú "1 hacia el controlador".

Resultado

Los valores del registro de receta se habrán transferido al controlador y tendrán efecto en el proceso.

Nota

Basic Panels

En los Basic Panels, el comando de menú "hacia el controlador" también puede estar configurado para la lista de registros. En este caso también puede seleccionar el comando de menú "hacia el controlador" en la lista de registros.

10.12 Características

10.12.1 Engineering System

Engineering System

Las tablas siguientes ayudarán al usuario a determinar si su proyecto se encuentra dentro de las características aplicables al sistema de ingeniería.

Además de los límites indicados, también se deberán tener en cuenta las restricciones debidas a la memoria principal disponible. WinCC utiliza un máximo de 2 GB de la memoria principal para el sistema operativo. Aún así, se recomienda una memoria principal de más de 2 GB si en el PC se ejecutan simultáneamente muchas aplicaciones que requieran gran cantidad de memoria.

Límites del sistema para un proyecto

En función de si utiliza un sistema operativo de 32 o 64 bits, existen distintos límites para el sistema de ingeniería.

	Sistema operativo de 32 bits	Sistema operativo de 64 bits
Número de paneles de operador en el proyecto	20	40
Número de variables HMI [1)]	80.000	260.000
Número de variables de fichero	8.000	20.000
Número de bloques (faceplates, tipos de datos de usuario) [3)]	10.000	20.000
Número de imágenes	3.000	6.000
Número de objetos de imagen por imagen	3.200	3.200
Número de objetos de imagen	320.000	640.000
Número de avisos [2) 3)]	20.000	60.000
Número de textos [3)]	300.000	600.000
Número de listas de textos y listas de gráficos [3)]	10.000	20.000
Número de entradas por lista de textos	3.000	3.000
Número de idiomas	32	32
Número de bibliotecas globales [3)]	20	20
Número de objetos en la biblioteca del proyecto [3)]	300.000	800.000

1) Incluidas las variables de fichero.
2) Con una media de 3 textos y un parámetro dinámico
3) Incluidos los objetos configurados en el área "Programación PLC"

Límites del sistema WinCC para un dispositivo HMI

	Sistema operativo de 32 bits	Sistema operativo de 64 bits
Número de variables HMI [1]	80.000	260.000
Número de variables de fichero	8.000	20.000
Número de ficheros	500	1.000
Número de imágenes	1000	2.000
Número de objetos de imagen por imagen	3.200	3.200
Número de objetos de imagen	320.000	640.000
Número de listas de funciones	30.000	60.000
Número de animaciones y scripts locales	50.000	100.000
Número de funciones definidas por el usuario	1.000	2.000
Número de tareas	500	1.000
Número de avisos [2]	20.000	60.000
Número de recetas	1.000	2.000
Número de elementos de receta	10.000	20.000
Número de textos	300.000	600.000
Número de listas de textos y listas de gráficos	1.000	2.000
Número de entradas por lista de textos	5.000	10.000
Número de usuarios	200	200
Número de informes	300	600

1) Incluidas las variables de fichero.
2) Con una media de 3 textos y un parámetro dinámico

Límites del sistema durante la migración

Es posible migrar proyectos que se encuentran en uno o más puntos por encima de los límites indicados del sistema.

Si durante la migración surge un proyecto cuyos límites rebasan los límites indicados del sistema, aparece un aviso que informa sobre esta situación. Para garantizar un funcionamiento seguro en WinCC, después de la migración es preciso adaptar el proyecto de forma que se mantengan los límites indicados del sistema.

10.12.2 Basic Panel

Basic Panel

La tabla siguiente le ayudará a determinar si su proyecto se encuentra dentro de las características aplicables al panel de operador.

Los valores máximos indicados no son aditivos. El funcionamiento de las configuraciones que agotan todos los límites del sistema no puede garantizarse en los paneles.

Además de los límites indicados, también se deberán tener en cuenta las limitaciones debidas a la memoria disponible para la configuración.

Variables

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de variables del proyecto	250	500	250 (mono) 500 (color)	500	500	500
Número de PowerTags	--	--	--	--	--	--
Número de elementos por matriz	100	100	100	100	100	100
Número de variables locales	--	--	--	--	--	--
Número de estructuras	--	--	--	--	--	--
Número de elementos de estructura	--	--	--	--	--	--

Avisos

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de categorías de avisos	32	32	32	32	32	32
Número de avisos de bit	200	200	200	200	200	200
Número de avisos analógicos	15	15	15	15	15	15
Longitud de un aviso (en caracteres)	80	80	80	80	80	80
Número de valores de proceso por aviso	8	8	8	8	8	8
Tamaño del búfer de avisos	256	256	256	256	256	256
Número de eventos de aviso pendientes	64	64	64	64	64	64

Imágenes

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de imágenes	50	50	50	50	50	50
Número de campos por imagen	30	30	30	30	30	30
Número de variables por imagen	30	30	30	30	30	30
Número de objetos complejos por imagen [1)]	5	5	5	5	5	5
Número de elementos de matriz por imagen [2)]	100	100	100	100	100	100

1) Son objetos complejos: barra, deslizador, librería de símbolos, reloj y todos los objetos del área Controles.

2) Los elementos de matriz contenidos en recetas también cuentan.

Recetas

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de recetas	5	5	5	5	5	5
Número de elementos por receta[1)]	20	20	20	20	20	20
Longitud de los datos útiles (en bytes) por registro	--	--	--	--	--	--
Número de registros por receta	20	20	20	20	20	20
Memoria reservada para los registros en la memoria flash interna	40 kB	40 kB	40 kB	40 kB	40 kB	40 kB

1) En caso de utilizar matrices, cada elemento de matriz cuenta como un elemento de receta.

Ficheros

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de ficheros	--	--	--	--	--	--
Número de entradas por fichero (incl. todos los segmentos) [1)]	--	--	--	--	--	--
Número de segmentos de fichero	--	--	--	--	--	--
Disparador cíclico para archivar variables	--	--	--	--	--	--
Número de variables archivables por fichero	--	--	--	--	--	--

1) Con el método de archivación "Fichero circular segmentado", el número de entradas es aplicable a todos los ficheros siguientes. El producto resultante de multiplicar el número de ficheros de continuación por el número de registros por fichero de continuación no debe rebasar el límite del sistema

Curvas

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de curvas	25	25	25	25	25	25

Listas de textos y gráficos

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de listas de gráficos	100	100	100	100	100	100
Número de listas de textos	150	150	150	150	150	150
Número de entradas por lista de textos o gráficos	30	30	30	30	30	30
Número de objetos gráficos	500	500	500	500	500	500
Número de elementos de texto	500	500	500	500	500	500

Scripts

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de scripts	--	--	--	--	--	--

Comunicación

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de conexiones	4	4	4	4	4	4
Número de conexiones basadas en "SIMATIC HMI HTTP"	--	--	--	--	--	--

Sistema de ayuda

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Longitud de un texto de ayuda (en caracteres)	500	500	500	500	500	500

Idiomas

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de idiomas de runtime	5	5	5	5	5	5

Planificador de tareas

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Tareas disparadas por tiempo [1)]	--	--	--	--	--	--

1) Las tareas disparadas por eventos no son relevantes para los límites del sistema

Administración de usuarios

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Número de grupos de usuarios	50	50	50	50	50	50
Número de autorizaciones	32	32	32	32	32	32
Número de usuarios	50	50	50	50	50	50

Proyecto

	KP300 Basic	KP400 Basic	KTP400 Basic	KTP600 Basic	KTP1000 Basic	TP1500 Basic
Tamaño del archivo de proyecto "*.srt"	1024 kB	1024 kB	1024 kB	1024 kB	1024 kB	1024 kB

Consulte también

Manual S7-1200 (http://support.automation.siemens.com/WW/view/en/36932465/0/es)

10.12.3 Basic Panel de 2.ª generación

Basic Panel de 2.ª generación

La tabla siguiente le ayudará a determinar si su proyecto se encuentra dentro de las características aplicables al panel de operador.

Los valores máximos indicados no son aditivos. El funcionamiento de las configuraciones que agotan todos los límites del sistema no puede garantizarse en los paneles.

Además de los límites indicados, también se deberán tener en cuenta las limitaciones debidas a la memoria disponible para la configuración.

Variables

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de variables del proyecto	800	800	800	800
Número de PowerTags	--	--	--	--
Número de elementos por matriz	100	100	100	100
Número de variables locales	--	--	--	--
Número de estructuras	--	--	--	--
Número de elementos de estructura	--	--	--	--

Avisos

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de categorías de avisos	32	32	32	32
Número de avisos de bit	1000	1000	100	100
Número de avisos analógicos	25	25	25	25
Longitud de un aviso (en caracteres)	80	80	80	80
Número de valores de proceso por aviso	8	8	8	8
Tamaño del búfer de avisos	256	256	256	256
Número de eventos de aviso pendientes	64	64	64	64

Imágenes

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de imágenes	250	250	250	250
Número de campos por imagen	100	100	100	100
Número de variables por imagen	100	100	100	100
Número de objetos complejos por imagen [1)]	150	150	150	150
Número de visores de recetas por imagen	10	10	10	10
Número de visores de curvas por imagen	8	8	8	8
Número de visores de avisos por imagen	20	20	20	20
Número de visores de usuarios por imagen	1	1	1	1
Número de visores de diagnóstico del sistema por imagen	150	150	150	150
Número de elementos de matriz por imagen [2)]	100	100	100	100

1) Son objetos complejos: barra, deslizador, librería de símbolos, reloj y todos los objetos del área Controles.

2) Los elementos de matriz contenidos en recetas también cuentan.

Recetas

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de recetas	50	50	50	50
Número de elementos por receta[1)]	100	100	100	100
Longitud en KB de los datos útiles por registro	32	32	32	32
Número de registros por receta	100	100	100	100
Memoria reservada para los registros en la memoria flash interna	256 kB	256 kB	256 kB	256 kB

1) En caso de utilizar matrices, cada elemento de matriz cuenta como un elemento de receta.

Ficheros

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de ficheros	2	2	2	2
Número de entradas por fichero (incl. todos los segmentos) [1)]	10	10	10	10
Número de segmentos de fichero	10000	10000	10000	10000
Disparador cíclico para archivar variables	400	400	400	400
Número de variables archivables por fichero	1 s	1 s	1 s	1 s

1) Con el método de archivación "Fichero circular segmentado", el número de entradas es aplicable a todos los ficheros siguientes. El producto resultante de multiplicar el número de ficheros de continuación por el número de registros por fichero de continuación no debe rebasar el límite del sistema

Curvas

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de curvas	25	25	25	25

Listas de textos y gráficos

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de listas de gráficos	100	100	100	100
Número de listas de textos	300	300	300	300
Número de entradas por lista de textos o gráficos	100	100	100	100
Número de objetos gráficos	1000	1000	1000	1000
Número de elementos de texto	2500	2500	2500	2500

Scripts

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de scripts	--	--	--	--

Comunicación

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de conexiones	4	4	4	4
Número de conexiones basadas en "SIMATIC HMI HTTP"	--	--	--	--

Sistema de ayuda

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Longitud de un texto de ayuda (en caracteres)	500	500	500	500

Idiomas

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de idiomas de runtime	10	10	10	10

Planificador de tareas

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Tareas disparadas por tiempo [1)]	--	--	--	--

1) Las tareas disparadas por eventos no son relevantes para los límites del sistema

Administración de usuarios

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Número de grupos de usuarios	50	50	50	50
Número de autorizaciones	32	32	32	32
Número de usuarios	50	50	50	50

Proyecto

	KTP400 Basic	KTP700 Basic	KTP900 Basic	KTP1200 Basic
Tamaño del archivo de proyecto "*.srt"	10 MB	10 MB	10 MB	10 MB

10.12.4 Datos técnicos generales

10.12.4.1 Caracteres utilizables

Introducción

La siguiente tabla indica las limitaciones que deben tenerse en cuenta al adjudicar nombres y contraseñas.

Caracteres utilizables

Nombres / contraseñas	Limitación
Nombres de objetos	No utilice los caracteres ? " / \ * < > %
Nombres y contraseñas en el visor de usuarios	No utilice los caracteres ? " / \ § & $ %
Nombres de ficheros de avisos	En las ubicaciones "Archivo - RDB", "Archivo - CSV (ASCII)" y "Archivo - TXT (Unicode)", no utilice los caracteres \ / * ? : " < > |
	En la ubicación "Base de datos", utilice solo los caracteres a-z A-Z 0-9 _ @ # $ Los caracteres _ @ # $ no deben usarse en el primer carácter del nombre
Nombres de variables HMI	Los nombres de variables HMI no pueden comenzar con el carácter @
Nombres de imágenes	No utilice los caracteres especiales ? " / \ * < >
Nombres de AuditTrails	No utilice caracteres especiales
Nombres de conexiones	No utilice espacios en blanco ni caracteres especiales
Nombres de funciones VB	No utilice espacios en blanco, caracteres especiales ni palabras clave VB
Nombres de usuarios y grupos de usuarios de Windows al utilizar SIMATIC Logon	No utilice los caracteres especiales / \

10.12.4.2 Impresoras recomendadas

Impresoras recomendadas

Encontrará la lista actual de impresoras recomendadas para los paneles de operador en la siguiente dirección de Internet:

Enlace a la lista de impresoras actual (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=0&lang=es&referer=%2fWW%2f&func=cslib.csinfo&siteid=csius&caller=view&extranet=standard&viewreg=WW&nodeid0=10805558&objaction=csopen)

Nota

Todos los paneles de operador excepto PC y Panel PC soportan como máximo una impresora vía conexión USB, aunque físicamente existan varias conexiones.

Consulte también

Lista de impresoras (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=0&lang=es&referer=%2fWW%2f&func=cslib.csinfo&siteid=csius&caller=view&extranet=standard&viewreg=WW&nodeid0=10805558&objaction=csopen)

10.12.4.3 Memoria necesaria para recetas en Basic Panels

Introducción

El siguiente cálculo de la memoria necesaria para recetas es válido para Basic Panels y para los paneles de operador OP 77A y TP 177A.

Limitaciones

La memoria disponible en el panel de operador para las recetas es de 39 Kbytes. No se permite exceder este espacio de memoria. El espacio de memoria total para las recetas se calcula de la manera siguiente: suma de todas las recetas + receta que requiere el mayor espacio de memoria.

El espacio de memoria máximo disponible para cada receta no puede exceder los 19 Kbytes.

Calcular la memoria necesaria

La memoria necesaria para cada receta (en Kbytes) se calcula sumando los adendos D1 + D2 + D3.

Siendo:

- D1 = número de registros x M

  Siendo M (tamaño de un registro):
  M = 1 x número de elementos de un byte + 2 x número de elementos de 2 bytes + 4 x número de elementos de 4 bytes + 8 x número de elementos de 8 bytes + K

  Siendo K (tamaño de los elementos de cadena):
  K = número de elementos de cadena x (tamaño de la cadena + 1) x 2

- D2 - tamaño del registro

  D2 = 4 + cantidad de idiomas x 8 + número de idiomas x (4 + 4 x número de registros + (longitud del nombre de registro + 1) x 2 x número de registros) + 8 + 8 x número de registros

  O, expresado de otra forma:

  D2 = 12 + 8 x número de registros + número de idiomas x (12 + número de registros x (4 + (longitud del nombre de registro +1) x 2))

- D3 Memoria utilizada en común
  D3 = 14 + número de elementos

Nota

El cálculo de las matrices y elementos individuales se puede realizar de la manera descrita arriba.

10.13 Cambio a WinCC en el TIA Portal

10.13.1 Resumen de la migración a WinCC V12

Resumen del apartado "Cambio a WinCC en el TIA Portal"

SIMATIC WinCC ofrece una serie de cambios funcionales en el TIA Portal. Algunas funciones presentan diferencias con respecto a las funciones que el usuario ya conoce en entornos como WinCC V7 o WinCC flexible.

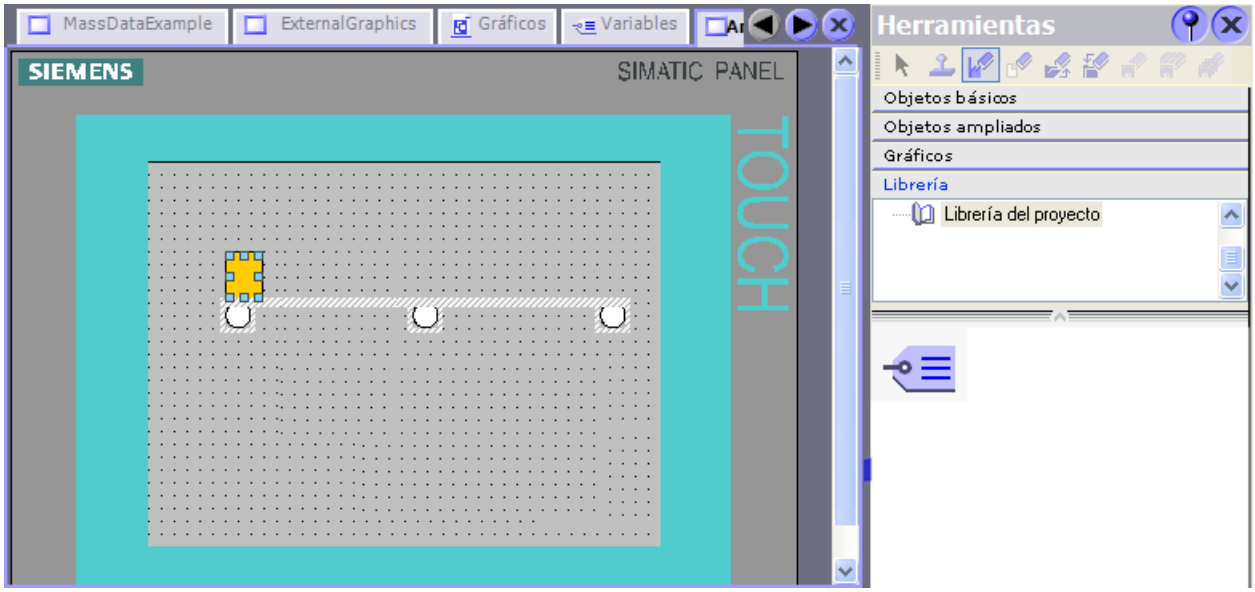
Este documento ofrece una vista general de las funciones y procedimientos especiales disponibles en SIMATIC WinCC en el TIA Portal.

Se trata de funciones y procedimientos que presentan cambios esenciales con respecto a las versiones WinCC V7 y WinCC flexible o que tienen un nombre diferente.

10.13.2 Librerías

Librerías en WinCC flexible

Las librerías son colecciones de objetos de imagen ya configurados. Amplían la cantidad de objetos de imagen disponibles y aumentan la efectividad durante la configuración, ya que los objetos de la librería pueden utilizarse una y otra vez sin necesidad de volver a configurarlos.



En WinCC flexible se pueden crear dos tipos de librerías:

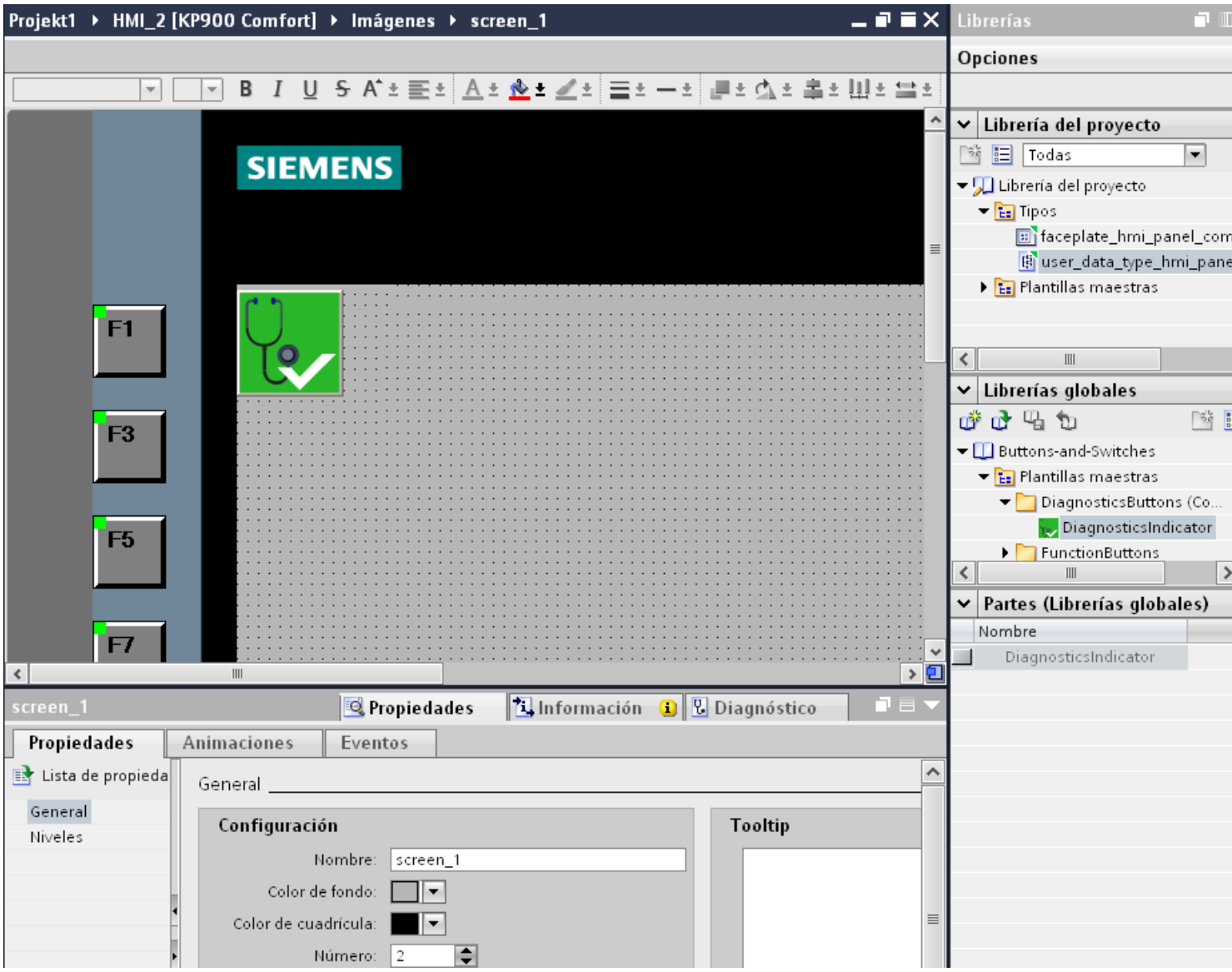
- Librería del proyecto
- Librería global

Una librería puede contener todos los objetos de WinCC flexible, tales como imágenes, variables, objetos gráficos o avisos.

¿Cómo se configuran librerías con WinCC en el TIA Portal?

Con WinCC también se configuran la "Librería del proyecto" y la "Librería global".

En comparación con WinCC flexible, no es posible depositar funciones de sistema en las librerías.



Tanto la "Librería de proyecto" como la "Librería global" contienen las carpetas "Plantillas maestras" y "Tipos". Los objetos de la librería se pueden crear o utilizar como copia maestra o como tipo.

- Copias maestras
  Para crear copias de los objetos de la librería independientes entre sí, utilice las copias maestras.
- Tipos
  A partir de los objetos de la carpeta "Tipos" se crean instancias, que son las que se utilizan en el proyecto. Las instancias están vinculadas a su tipo respectivo. Los cambios de una instancia también se realizan en las demás instancias. En la Task Card "Librerías", los tipos aparecen marcados con un triángulo verde.
- Administrar los objetos de librería
  Los objetos de librería sólo se copian y mueven dentro de la misma librería.

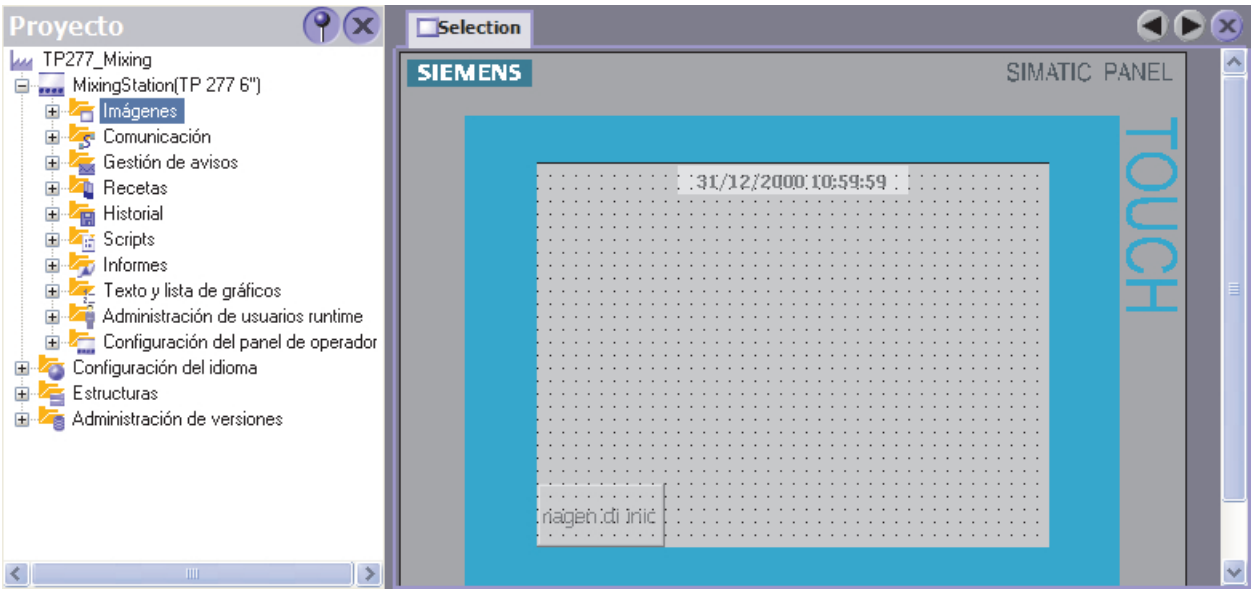
Encontrará más información al respecto en:

Librerías en WinCC (Página 9138)

10.13.3 Imágenes y plantillas

Imágenes y plantillas en WinCC flexible

WinCC flexible permite crear imágenes para manejar y visualizar máquinas e instalaciones. Para crear imágenes se dispone de objetos predefinidos para reproducir la instalación, visualizar los distintos procesos y especificar valores de proceso.



A cada panel de operador le corresponde en el proyecto una plantilla, en la que se pueden configurar las teclas de función y los objetos del proyecto de forma centralizada.

Todas las imágenes que se basan en la plantilla contienen las teclas de función y los objetos configurados en ella. Si se modifica un objeto o la asignación de una tecla de función en la plantilla, el objeto o tecla también se modifica en todas las imágenes que se basan en la plantilla.

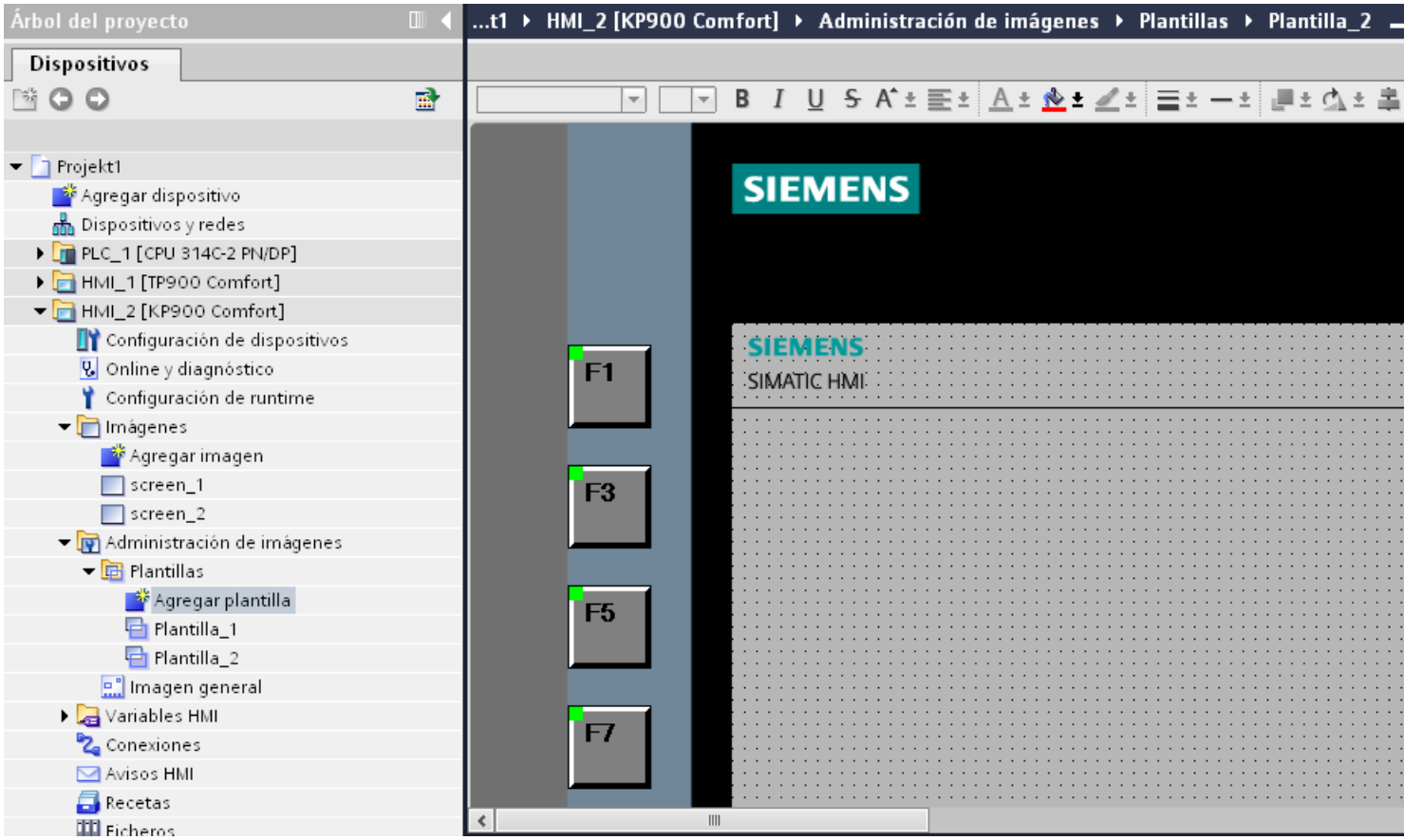
¿Cómo se configuran imágenes y plantillas con WinCC en el TIA Portal?

En WinCC, además de "Imágenes" también se configuran "Plantillas" y una "Imagen general".

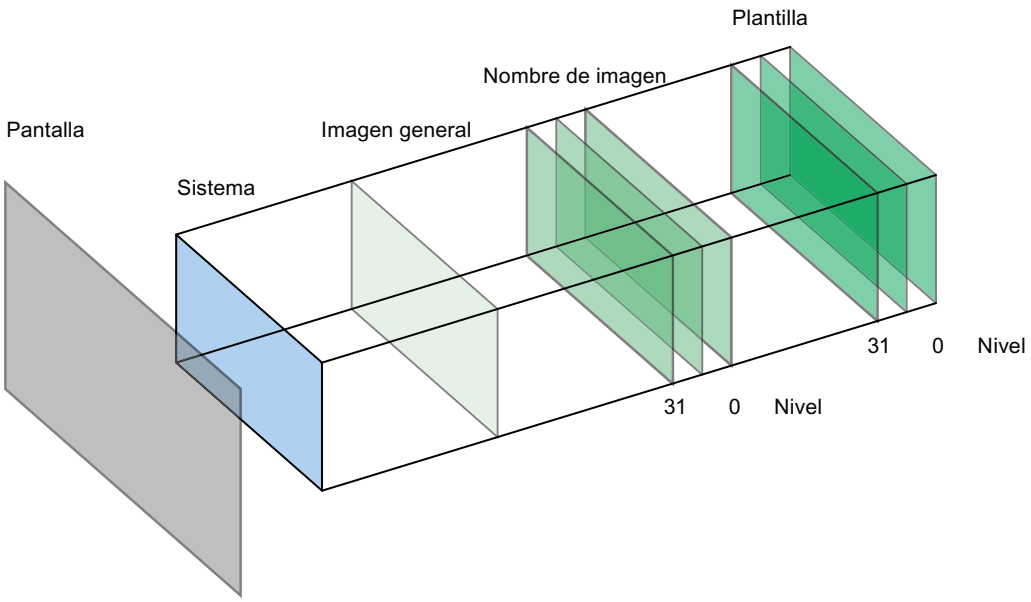
En la plantilla se definen las funciones y objetos que se aplican a todas las imágenes que se basan en dicha plantilla. En WinCC se pueden crear varias plantillas.

En la "Imagen general" se definen elementos para todas las imágenes de un panel de operador, independientes de la plantilla utilizada. Como objetos generales se dispone de los objetos "Ventana de avisos" e "Indicador de avisos". En los paneles de operador con teclas de función, éstas se asignan en el editor "Imagen general".

Para Comfort Panels también se puede configurar una "Ventana de diagnóstico del sistema" en la imagen general.



Para la visualización de las imágenes en runtime se aplica el siguiente orden (exceptuando los Controls):



Encontrará más información al respecto en:

Principios básicos de las imágenes (Página 8307)

Utilizar funciones tecnológicas 11

11.1 Contaje, medición y lectura de posición

11.1.1 Contaje, medición y lectura de posición (S7-1500)

11.1.1.1 Principios básicos de contaje, medición y lectura de posición

Sinopsis de los módulos tecnológicos y sus características

Módulos tecnológicos

La siguiente tabla resume las características de los módulos tecnológicos de contaje, medición y lectura de posición:

Tabla 11-1 Módulos tecnológicos de contaje, medición y lectura de posición

Característica	S7-1500		ET 200SP	
	TM Count 2x24V	TM PosInput 2	TM Count 1x24V	TM PosInput 1
Número de canales	2	2	1	1
Frecuencia de señal máxima	200 kHz	1 MHz	200 kHz	1 MHz
Frecuencia de señal máxima en encóders incrementales con evaluación cuádruple	800 kHz	4 MHz	800 kHz	4 MHz
Rango máximo de valores de contaje/posición	32 Bit	32 Bit/31 Bit	32 Bit	32 Bit/31 Bit
Conexión de encóder incremental y de impulsos RS422/TTL	—	X	—	X
Conexión de encóder incremental y de impulsos 24 V	X	—	X	—
Conexión de encóder absoluto SSI	—	X	—	X
Alimentación de encóder de 5 V	—	X	—	—
Alimentación de encóder de 24 V	X	X	X	X
Número de entradas digitales por canal	3	2	3	2

Característica	S7-1500		ET 200SP	
	TM Count 2x24V	TM PosInput 2	TM Count 1x24V	TM PosInput 1
Número de salidas digitales por canal	2	2	2	2
Control de puerta	X	X	X	X
Función Capture	X	X	X	X
Sincronización	X	X	X	X
Funciones de comparación	X	X	X	X
Medición de frecuencia, velocidad y periodo	X	X	X	X
Soporte de modo isócrono	X	X	X	X
Soporte de alarmas de diagnóstico	X	X	X	X
Soporte de alarmas de proceso	X	X	X	X
Filtro parametrizable para señales de contaje y entradas digitales	X	X	X	X

Sinopsis de las aplicaciones posibles

Introducción

El módulo tecnológico se configura y parametriza con el software de configuración.

Las funciones del módulo tecnológico se controlan a través del programa de usuario.

Entorno del sistema

Los módulos tecnológicos pueden utilizarse en los siguientes entornos de sistema:

Aplicaciones posibles	Componentes necesarios	Software de configuración	En el programa de usuario
Operación centralizada y descentralizada en un sistema S7-1500 con el correspondiente sistema ET 200	• Sistema de automatización S7-1500 • Sistema de periferia descentralizada ET 200 • Módulo tecnológico	STEP 7 (TIA Portal): • Configuración del dispositivo con configuración hardware (HWCN) • Ajuste de parámetros con objeto tecnológico High_Speed_Counter	Funciones de contaje y medición: Instrucción High_Speed_Counter para el objeto tecnológico Lectura de posición con encóder absoluto SSI: Acceso directo a la interfaz de control y realimentación del módulo tecnológico en los datos IO
		STEP 7 (TIA Portal): Configuración del dispositivo con configuración hardware (HWCN) en el modo de operación "Lectura de posición para Motion Control"	Control mediante el objeto tecnológico
Operación descentralizada en un sistema S7-300/400	• Sistema de automatización S7-300/400 • Sistema de periferia descentralizada ET 200 • Módulo tecnológico	STEP 7 (TIA Portal): Configuración del dispositivo y ajuste de parámetros con configuración hardware (HWCN) STEP 7: Configuración del dispositivo y ajuste de parámetros con HSP	Acceso directo a la interfaz de control y realimentación del módulo tecnológico en los datos IO
Operación descentralizada en un sistema de otro fabricante	• Sistema de automatización de otro fabricante • Sistema de periferia descentralizada ET 200 • Módulo tecnológico	Software de configuración de otro fabricante: Configuración del dispositivo y ajuste de parámetros con el archivo GSD	Acceso directo a la interfaz de control y realimentación del módulo tecnológico en los datos IO

Posibilidades de la parametrización

En un sistema S7-1500 tiene dos alternativas para parametrizar y controlar las funciones del módulo tecnológico:

- Configuración mediante el objeto tecnológico High_Speed_Counter y control mediante la instrucción High_Speed_Counter correspondiente
  El acceso a la interfaz de control y realimentación del módulo tecnológico se produce a través del objeto tecnológico.
- Ajuste de parámetros mediante HWCN
  El acceso a la interfaz de control y realimentación del módulo tecnológico se realiza mediante el acceso directo a los datos IO.

Configuración mediante objeto tecnológico

Para el uso centralizado y descentralizado en un sistema S7-1500 se recomienda la configuración cómoda y que admite gráficos por medio de un objeto tecnológico High_Speed_Counter. Encontrará una descripción a fondo de esta configuración a partir del apartado Objeto tecnológico High_Speed_Counter (Página 9408).

En la configuración del equipo, determine "Configurar canal mediante objeto tecnológico" para el módulo tecnológico. ver el apartado Modo de funcionamiento (Página 9433).

En los parámetros básicos del objeto tecnológico puede encontrar la asignación para el módulo tecnológico y el canal de contaje: ver el apartado Parámetros básicos (Página 9413).

Ajuste de parámetros mediante HWCN

Encontrará más indicaciones para el ajuste de parámetros mediante HWCN en la ayuda contextual de los parámetros en STEP 7 (TIA Portal). Encontrará una descripción de la interfaz de control y realimentación en los siguientes apartados:

Asignación de la interfaz de control (Página 9460)

Asignación de la interfaz de realimentación (Página 9463)

Captura de las señales de contaje

Contaje con encóder incremental o generador de impulsos

Contar significa registrar y sumar eventos. Los contadores de los módulos tecnológicos capturan señales de impulsos y señales incrementales, y las evalúan según corresponda. El sentido de contaje puede predefinirse a través de señales de encóder o de impulsos adecuadas o a través del programa de usuario.

Los procesos de contaje pueden controlarse con las entradas digitales del módulo tecnológico. Puede conmutar las salidas digitales con exactitud en los valores de contaje definidos, independientemente del programa de usuario.

Es posible parametrizar el comportamiento de los contadores con ayuda de las funciones que se describen a continuación.

Límites de contaje

Los límites de contaje definen el rango utilizado por los valores de contaje. Los límites de contaje son parametrizables y pueden modificarse en tiempo de ejecución a través del programa de usuario.

El límite de contaje máximo ajustable es 2147483647 (2^{31}–1). El límite de contaje mínimo ajustable es –2147483648 (–2^{31}).

Es posible parametrizar el comportamiento del contador en los límites de contaje:

- Continuar o finalizar los procedimientos de contaje al rebasarse un límite de contaje (cierre de puerta automático).
- Cambiar el valor de contaje por el valor de arranque o por el límite de contaje opuesto al rebasarse un límite de contaje.

Valor de arranque

Como valor de arranque se puede parametrizar un valor cualquiera comprendido entre los límites de contaje. El valor de arranque puede modificarse en tiempo de ejecución a través del programa de usuario.

Según la parametrización, el módulo tecnológico puede cambiar el valor de contaje actual por el valor de arranque al sincronizar, durante la función de Capture, al rebasarse un límite de contaje o al abrirse la puerta.

Control de puerta

La apertura y el cierre de la puerta de hardware (puerta HW) y la puerta de software (puerta SW) definen el intervalo temporal en el que se capturan las señales de contaje.

El control de la puerta HW se realiza de forma externa a través de una entrada digital del módulo tecnológico. El control de la puerta SW se realiza a través del programa de usuario. La puerta HW se puede activar con la parametrización. La puerta SW (bit en la interfaz de control de los datos IO cíclicos) no se puede desactivar.

Capture

Es posible parametrizar el flanco de una señal de referencia externa que dispare el almacenamiento del valor de contaje o del valor de posición actuales como valor Capture. Las siguientes señales externas pueden desencadenar la función de Capture.

- Flanco ascendente o descendente de una entrada digital
- Ambos flancos de una entrada digital
- Flanco ascendente de la señal N en la entrada del encóder (con encóders incrementales o generadores de impulsos)

En el caso de los encóders incrementales y los encóders de impulsos, es posible parametrizar si a continuación de la función de Capture se seguirá contando con el valor de contaje actual o con el valor de arranque.

Sincronización

Se puede parametrizar el flanco de una señal de referencia externa que cargue el contador con el valor de arranque predefinido. Las siguientes señales externas pueden cargar el contador con el valor de arranque:

- Flanco ascendente o descendente de una entrada digital
- Flanco ascendente de la señal N en la entrada del encóder
- Flanco ascendente de la señal N en la entrada del encóder en función del nivel de la entrada digital asignada

Histéresis

Se puede especificar una histéresis para los valores de comparación, dentro de la cual se impide que vuelva a conmutarse una salida digital. Un encóder puede permanecer en una posición determinada en torno a la cual oscila el valor de contaje con leves movimientos. Si

en este rango de oscilación hay un valor de referencia o un límite de contaje, la salida digital correspondiente se activa y desactiva con la frecuencia conveniente sin aplicar una histéresis. La histéresis impide esta conmutación no deseada.

Lectura de posición con encóder absoluto SSI

Descripción

Se pueden utilizar los módulos tecnológicos TM PosInput con un encóder absoluto SSI para la lectura de posición. El módulo tecnológico lee el valor de posición a través de una interfaz serie síncrona del encóder absoluto SSI y lo pone a disposición del controlador.

Puede conmutar las salidas digitales del módulo tecnológico con exactitud en los valores de posición definidos, independientemente del programa de usuario. La lectura de posición con un encóder absoluto SSI se lleva a cabo sin control de puerta. Por motivos del sistema, la sincronización no es posible con un encóder absoluto SSI.

Para ello, en el ajuste de parámetros (HWCN) del módulo tecnológico en STEP 7 (TIA Portal), seleccione el tipo de señal "Encóder absoluto (SSI)".

Conversión Gray-Binario

Se soportan encóders absolutos SSI con codificación Gray y Binaria.

Capture

Puede parametrizar uno o ambos flancos de una entrada digital que desencadenen el almacenamiento del valor de posición actual como valor de Capture.

Histéresis

Se puede especificar una histéresis para los valores de comparación, dentro de la cual se impide que vuelva a conmutarse una salida digital. Un encóder puede permanecer en una posición determinada en torno a la cual oscila el valor de posición con leves movimientos. Si en este rango de oscilación hay un valor de comparación o un límite, la salida digital correspondiente se activa y desactiva con la frecuencia conveniente sin aplicar una histéresis. La histéresis impide esta conmutación no deseada.

Rango para valor de posición

Puede definir para el encóder absoluto SSI una longitud de telegrama de entre 10 y 40 bits. Los números de bit parametrizables del LSB y del MSB del valor de posición en el telegrama definen el rango de valores. El módulo tecnológico puede leer un valor de posición con una longitud máxima de 31 bits y transmitírselo al control. El valor de posición se trata como valor positivo sin signo y puede adoptar valores entre "0" y "$2^{(MSB-LSB+1)-1}$".

Telegrama SSI completo

En lugar de una magnitud pueden devolverse los 32 bits menos significativos del actual telegrama SSI no procesado. De esta manera, además del valor de posición, obtiene bits específicos de encóder adicionales, p. ej., bits de error. Si el telegrama SSI es más corto que 32 bits, se devuelven el telegrama SSI completo, justificado a la derecha, y los bits superiores no usados con "0" en la interfaz de realimentación.

Comportamiento en los límites de contaje

Rebase de un límite de contaje

El límite de contaje superior se rebasa por exceso si el valor de contaje actual coincide con el límite superior de contaje y se produce un nuevo impulso de contaje en sentido ascendente. El límite inferior de contaje se rebasa por exceso si el valor de contaje actual coincide con el límite inferior de contaje y se produce un nuevo impulso de contaje en sentido descendente.

En caso de rebase, en la interfaz de realimentación se activa el bit de estado correspondiente:

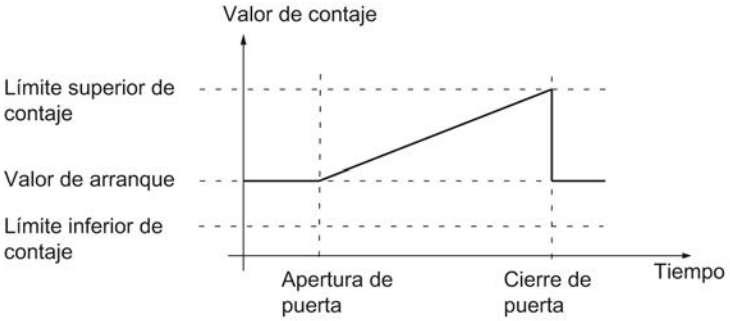
Límite de contaje rebasado	Bit de estado
Límite superior de contaje	EVENT_OFLW se activa
Límite inferior de contaje	EVENT_UFLW se activa

Puede restablecer los bits de estado con RES_EVENT .

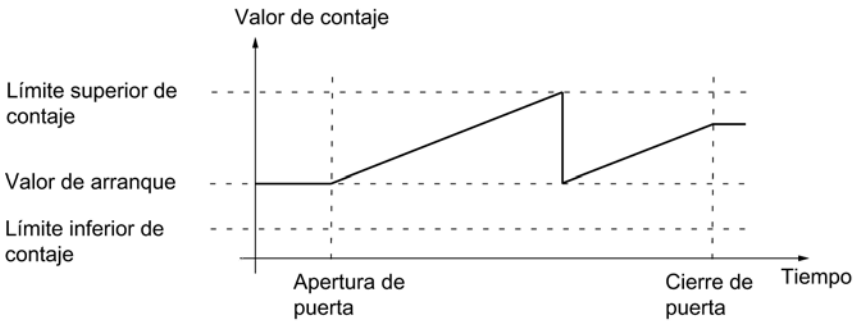
Para el rebase por exceso de un límite de contaje puede parametrizar si se sigue contando con el valor de contaje y con cuál de ellos.

Ejemplos

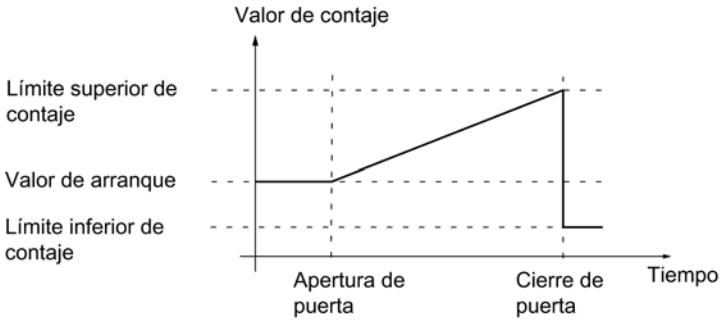
La siguiente figura muestra un ejemplo de finalización del proceso de contaje (cierre de puerta automática) tras un rebase por exceso y el establecimiento del contador al valor de arranque:



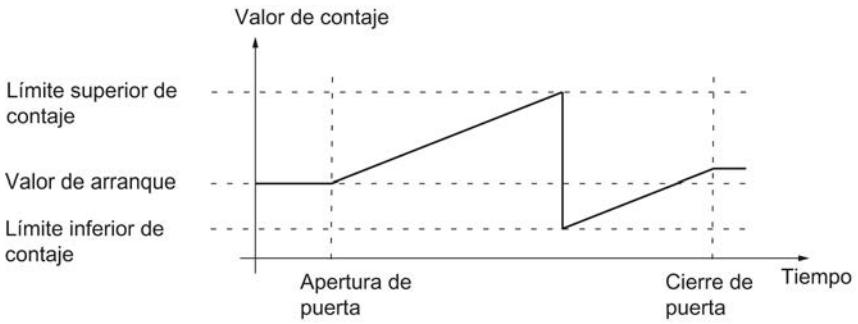
La siguiente figura muestra un ejemplo de continuación del proceso de contaje tras un rebase por exceso y el establecimiento del contador al valor de arranque:



La siguiente figura muestra un ejemplo de finalización del proceso de contaje tras un rebase por exceso y el establecimiento del contador al otro límite de contaje:



La siguiente figura muestra un ejemplo de continuación del proceso de contaje tras un rebase por exceso y el establecimiento del contador al otro límite de contaje:



Control de puerta con encóder incremental o generador de impulsos

Muchas aplicaciones requieren que el contaje comience o se detenga en función de otros eventos. El contaje se inicia o se detiene por medio de la función de puerta.

Por cada canal, los módulos tecnológicos poseen dos puertas, que determinan la puerta interna resultante:

- Puerta software (puerta SW)
- Puerta hardware (puerta HW)

Puerta SW

La puerta SW se abre y cierra con el bit de control (Página 9460) SW_GATE. El estado de la puerta SW se indica mediante el bit de realimentación (Página 9463) STS_SW_GATE .

Puerta HW

La puerta HW es opcional. La puerta HW se abre y cierra mediante señales en las entradas digitales parametrizadas del canal.

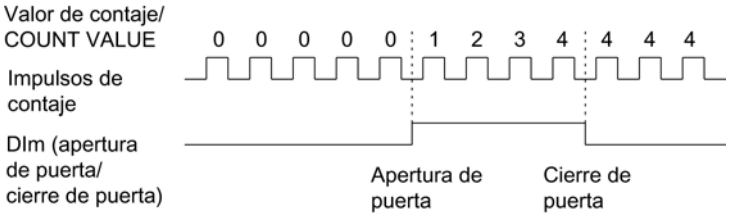
Nota

Los filtros de entrada parametrizados retrasan la señal de control de la entrada digital.

El estado de una entrada digital DIm se indica mediante el bit de realimentación (Página 9463) STS_DIm correspondiente.

Apertura y cierre controlados por nivel de la puerta HW con una entrada digital

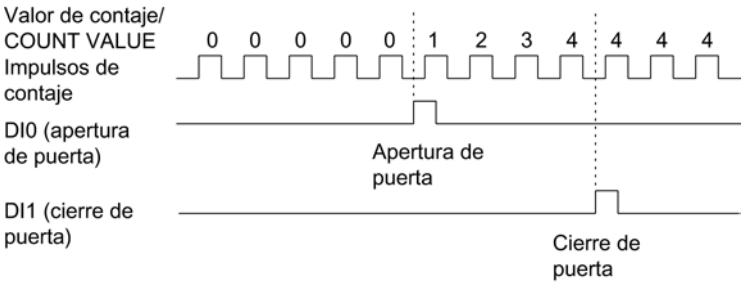
La siguiente figura muestra un ejemplo de apertura y cierre controlados por nivel con una entrada digital. La entrada digital se parametriza de tal modo que esté activa con nivel alto:



Mientras la entrada digital está activa, la puerta HW está abierta y se cuentan los impulsos de contaje. Si la entrada digital se desactiva, la puerta HW se cierra. Los impulsos de contaje se ignoran y el valor de contaje se mantiene constante.

Apertura y cierre controlados por flanco de la puerta HW con dos entradas digitales

La siguiente figura muestra un ejemplo de apertura y cierre con dos entradas digitales. Las dos entradas digitales se parametrizan de modo que el flanco ascendente abra la puerta HW:



La puerta HW se abre con un flanco parametrizado en la entrada digital, que está parametrizada para la apertura. La puerta HW se cierra con un flanco parametrizado en la entrada digital, que está parametrizada para el cierre.

Puerta interna

Puerta interna

La puerta interna está abierta cuando la puerta SW está abierta y la puerta HW está abierta o no se ha parametrizado. El estado de la puerta interna se indica mediante el bit de realimentación (Página 9463) STS_GATE.

Cuando la puerta interna está abierta, se inicia el contaje. Cuando la puerta interna está cerrada, el resto de los impulsos de contaje se ignoran y el contaje se detiene.

La puerta SW debe estar abierta para controlar un proceso de contaje solo con la puerta HW. Si no se parametriza ninguna puerta HW, se considerará que está siempre abierta. En ese caso, la puerta interna debe abrirse y cerrarse únicamente mediante la puerta SW.

Puerta hardware	Puerta software	Puerta interna
Abierta/no parametrizada	Abierta	Abierta
Abierta/no parametrizada	Cerrada	Cerrada
Cerrada	Abierta	Cerrada
Cerrada	Cerrada	Cerrada

Durante la parametrización del comportamiento del contador debe establecerse si, tras abrir la puerta interna, el proceso de contaje debe empezar con el valor de arranque o con el valor de contaje actual.

Además, la puerta interna puede cerrarse automáticamente al rebasar por exceso un valor de contaje. Para continuar el contaje es necesario que se cierre y se vuelva a abrir la puerta software o la puerta hardware.

Comportamiento del contador ante la apertura de puerta

Dispone de las siguientes posibilidades de parametrización para el comportamiento del contador en la apertura de la puerta:

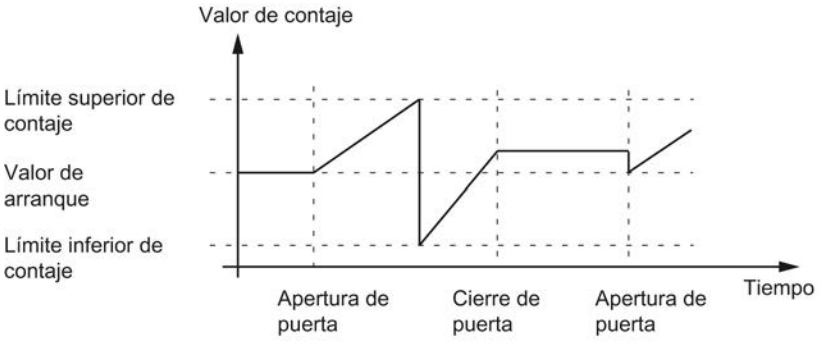
- Ajustar el contador al valor de arranque.
- Continuar con el valor de contaje actual.

Ajuste del contador al valor de arranque

Con esta parametrización se aplican los siguientes comportamientos de contador:

Cada proceso de contaje comienza tras la apertura de la puerta interna con el valor de arranque.

La siguiente imagen muestra un ejemplo de continuación del proceso de contaje tras poner el contador al valor de arranque:

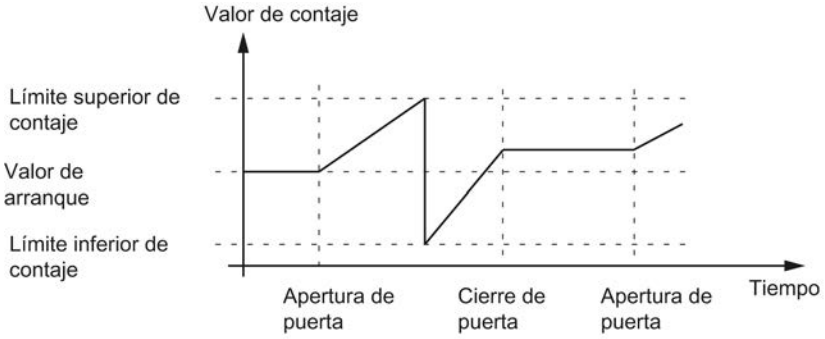


Continuar con el valor de contaje actual

Con esta parametrización se aplican los siguientes comportamientos de contador:

Cada proceso de contaje comienza tras la nueva apertura de la puerta interna con el valor de contaje actual.

La siguiente figura muestra un ejemplo de continuación del proceso de contaje con el valor de contaje actual:



Capture

Capture con encóder incremental o generador de impulsos

Descripción

Con la función "Capture" se guarda el valor de contaje actual a través de una señal de referencia externa. La función de Capture se puede parametrizar para las siguientes señales de referencia:

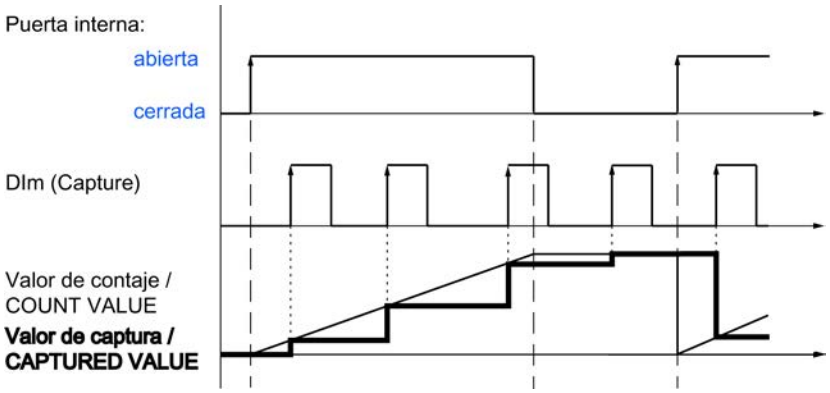
- Flanco ascendente o descendente en una entrada digital
- Flanco ascendente y descendente en una entrada digital
- Flanco ascendente de la señal N en la entrada del encóder

Funcionamiento

El valor de Capture es siempre exactamente el valor de contaje en el instante del flanco correspondiente (retardado el tiempo de filtrado de entrada parametrizado). La función de Capture actúa independientemente del estado de la puerta interna. Si la puerta está cerrada, se guarda el valor de contaje sin cambios.

La siguiente figura muestra un ejemplo de la función de Capture con la siguiente parametrización:

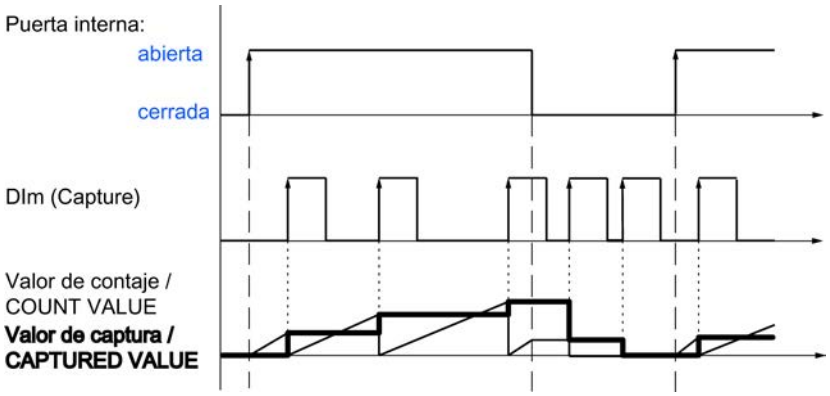
- Valor de arranque = 0
- Evento de Capture con flanco ascendente en la entrada digital parametrizada
- Poner el contador al valor de arranque con la apertura de puerta
- Continuar el contaje tras evento de Capture



La siguiente figura muestra otro ejemplo de la función de Capture con la siguiente parametrización:

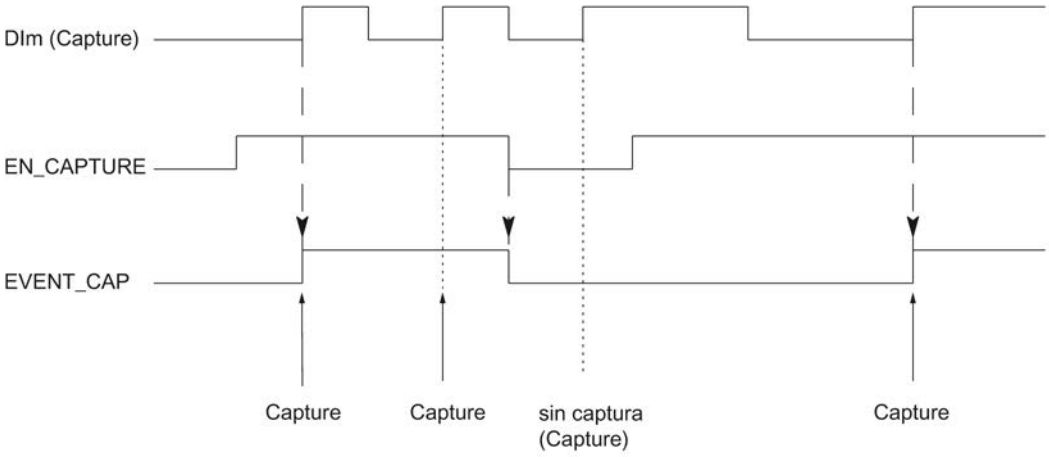
- Valor de arranque = 0
- Evento de Capture con flanco ascendente en la entrada digital parametrizada

- Poner el contador al valor de arranque con la apertura de puerta
- Tras el evento de Capture, establecer el valor de contaje al valor de arranque y continuar el contaje



Con el bit de control (Página 9460) EN_CAPTURE se habilita la función de Capture. El bit de realimentación (Página 9463) EVENT_CAP indica que se ha guardado un valor de contaje como valor Capture en la interfaz de realimentación. Cuando restablece EN_CAPTURE, se restablece EVENT_CAP. El estado de una entrada digital se indica mediante el bit de realimentación (Página 9463) STS_DIm correspondiente.

La siguiente figura muestra un ejemplo de los bits EN_CAPTURE y EVENT_CAP al utilizar la función de Capture mediante un flanco ascendente en una entrada digital:



Nota

Los filtros de entrada parametrizados retrasan la señal de control de la entrada digital correspondiente.

La función de Capture no tiene ninguna influencia en el bit de realimentación STS_CNT ni en los LED UP y DN.

Alarma de proceso

Se puede parametrizar una alarma de proceso para la función de Capture. Si las alarmas de proceso se disparan más rápido de lo que el sistema puede acusarlas, pueden perderse alarmas de proceso. Puede establecerse un aviso de alarma de proceso perdida por alarma de diagnóstico.

Capture con encóder absoluto SSI

Descripción

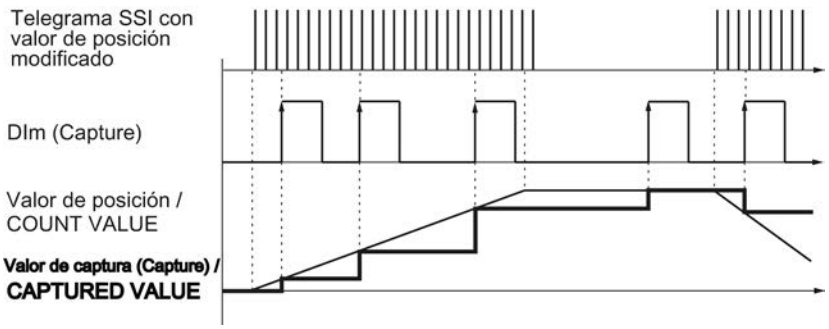
Con la función "Capture" se guarda el valor de posición actual a través de una señal de referencia externa. La función de Capture se puede parametrizar para las siguientes señales de referencia:

- Flanco ascendente o descendente en una entrada digital
- Flanco ascendente y descendente en una entrada digital

Funcionamiento

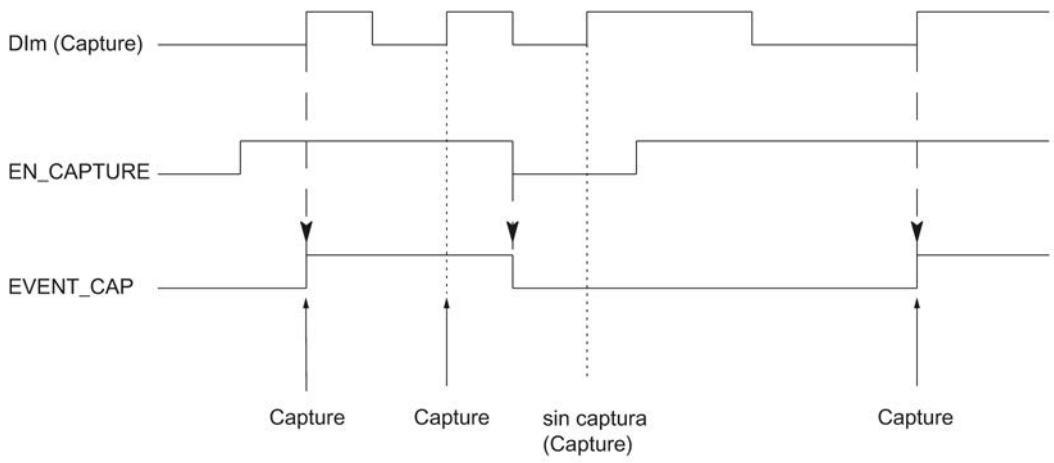
En el instante del flanco correspondiente, el valor de posición del último telegrama SSI válido se guarda en el valor Capture.

La siguiente figura muestra un ejemplo de evento de Capture mediante un flanco ascendente en la entrada digital parametrizada:



Con el bit de control (Página 9460) EN_CAPTURE se habilita la función de Capture. El bit de realimentación (Página 9463) EVENT_CAP indica que se ha guardado un valor de posición como valor de Capture en la interfaz de realimentación. Cuando restablece EN_CAPTURE, se restablece EVENT_CAP. El estado de una entrada digital se indica mediante el bit de realimentación (Página 9463) STS_DIm correspondiente.

La siguiente figura muestra un ejemplo de los bits EN_CAPTURE y EVENT_CAP al utilizar la función de Capture mediante un flanco ascendente en una entrada digital:



Nota

Los filtros de entrada parametrizados retrasan la señal de control de la entrada digital correspondiente.

Alarma de proceso

Se puede parametrizar una alarma de proceso para la función de Capture. Si las alarmas de proceso se disparan más rápido de lo que el sistema puede acusarlas, pueden perderse alarmas de proceso. Puede establecerse un aviso de alarma de proceso perdida por alarma de diagnóstico.

Sincronización

Descripción

Con la función "Sincronización" se pone el contador al valor de arranque predefinido mediante una señal de referencia externa. La sincronización se puede parametrizar para las siguientes señales de referencia:

- Flancos ascendentes o descendentes en una entrada digital
- Flanco ascendente de la señal N en la entrada del encóder
- Flanco ascendente de la señal N en la entrada del encóder en función del nivel de una entrada digital

Funcionamiento

La sincronización siempre se realiza en el instante preciso de la señal de referencia. La sincronización actúa independientemente del estado de la puerta interna.

Con el bit de control (Página 9460) EN_SYNC_UP se habilita la sincronización para el contaje en sentido ascendente. Con el bit de control (Página 9460) EN_SYNC_DN se habilita la sincronización para el contaje en sentido descendente. El bit de realimentación (Página 9463) EVENT_SYNC indica que se ha producido una sincronización. Al restablecer EN_SYNC_UP o EN_SYNC_DN, se restablece también EVENT_SYNC.

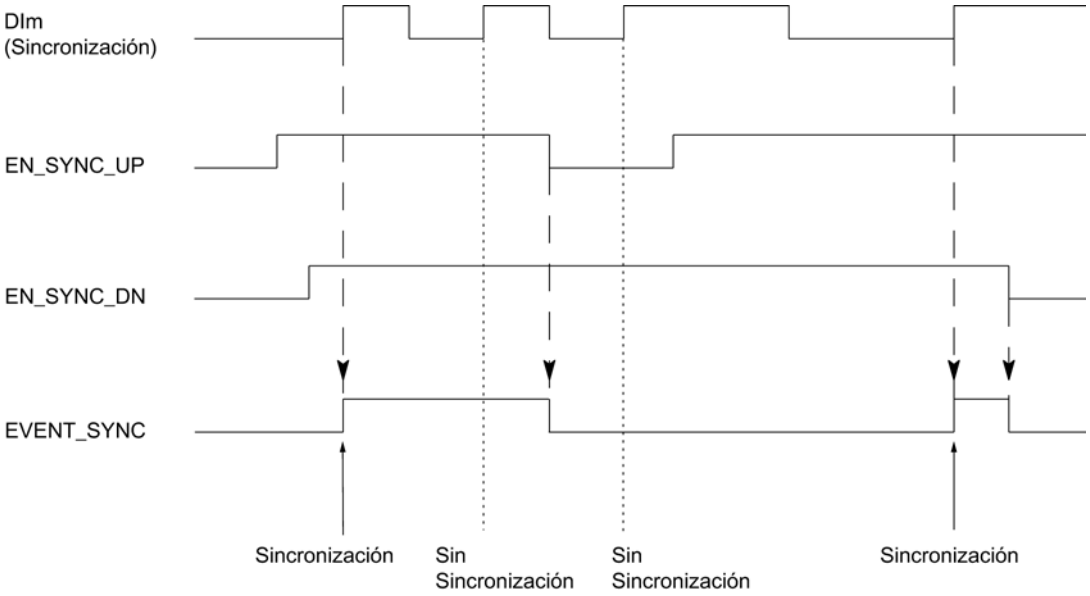
Nota

Los filtros de entrada parametrizados retrasan la señal de control de la entrada digital correspondiente.

La sincronización no tiene ninguna influencia en el bit de realimentación (Página 9463) STS_CNT ni en los LED UP y DN.

Sincronización única

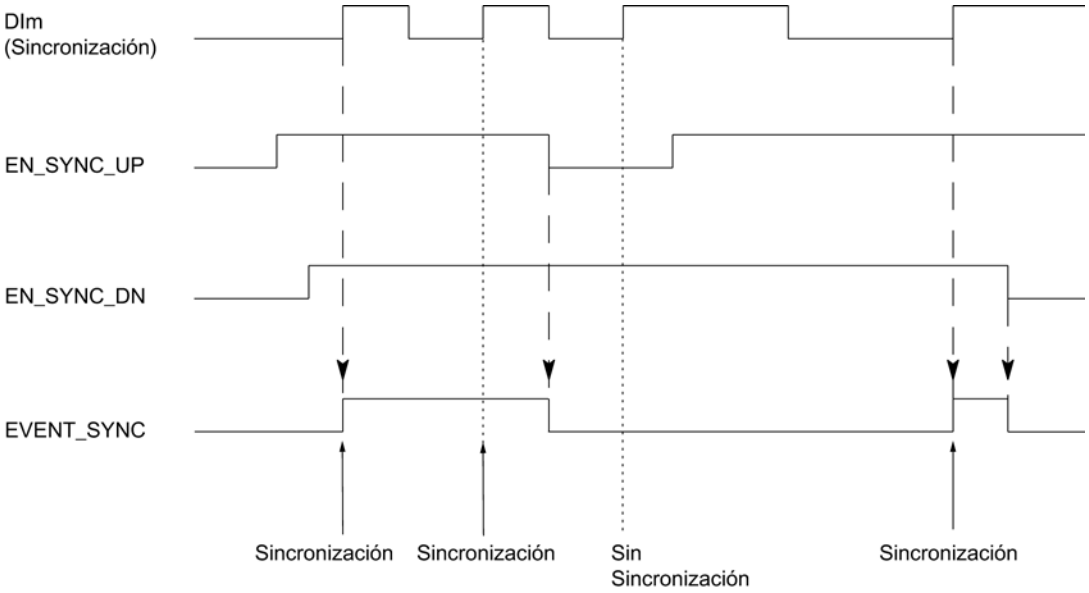
La siguiente figura muestra un ejemplo de los bits EN_SYNC_UP, EN_SYNC_DN y EVENT_SYNC con una única sincronización mediante un flanco en una entrada digital en el caso de los impulsos de contaje en sentido ascendente:



Tras la habilitación de la sincronización para el contaje en sentido ascendente, el contador se sincroniza con el primer flanco ascendente en la entrada digital parametrizada. Solo tras desactivar y volver a activar el bit de control (Página 9460) EN_SYNC_UP, el contador puede volver a sincronizarse.

Sincronización periódica

La siguiente figura muestra un ejemplo de los bits EN_SYNC_UP, EN_SYNC_DN y EVENT_SYNC con una sincronización periódica mediante un flanco en una entrada digital en el caso de los impulsos de contaje en sentido ascendente:



Mientras esté habilitada la sincronización para el contaje en sentido ascendente, el contador se sincroniza con cada flanco ascendente en la entrada digital parametrizada.

Alarma de proceso

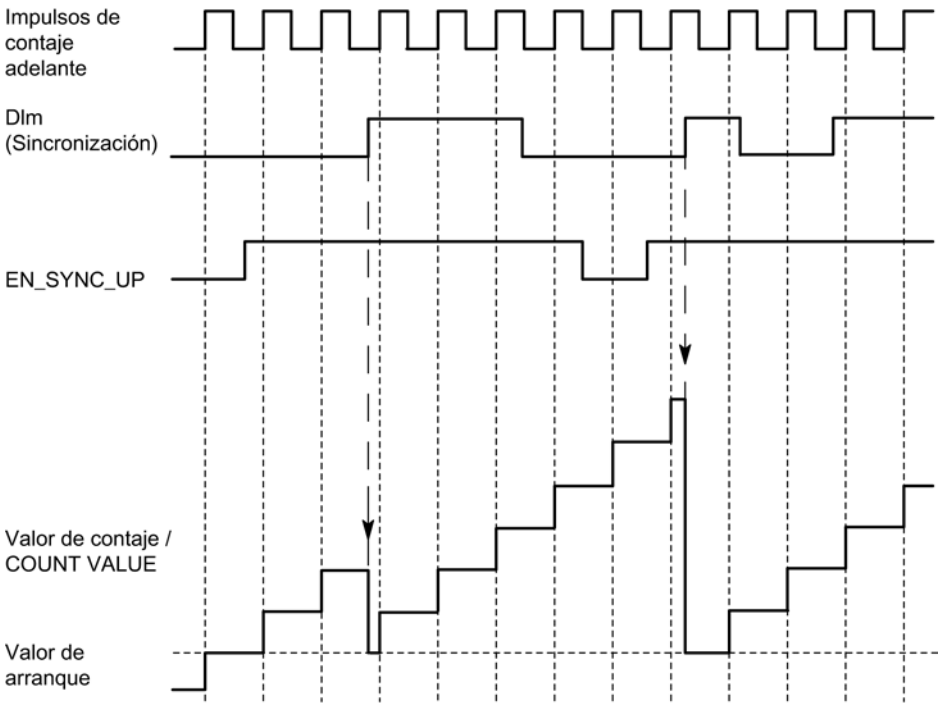
Se puede parametrizar una alarma de proceso para la sincronización. Si las alarmas llegan más rápido de lo que pueden ser acusadas por el sistema, se pierden alarmas de proceso. Puede establecerse un aviso de alarma de proceso perdida por alarma de diagnóstico.

Sincronización mediante entrada digital

Se puede activar la sincronización en una entrada digital mediante flancos.

Sincronización única

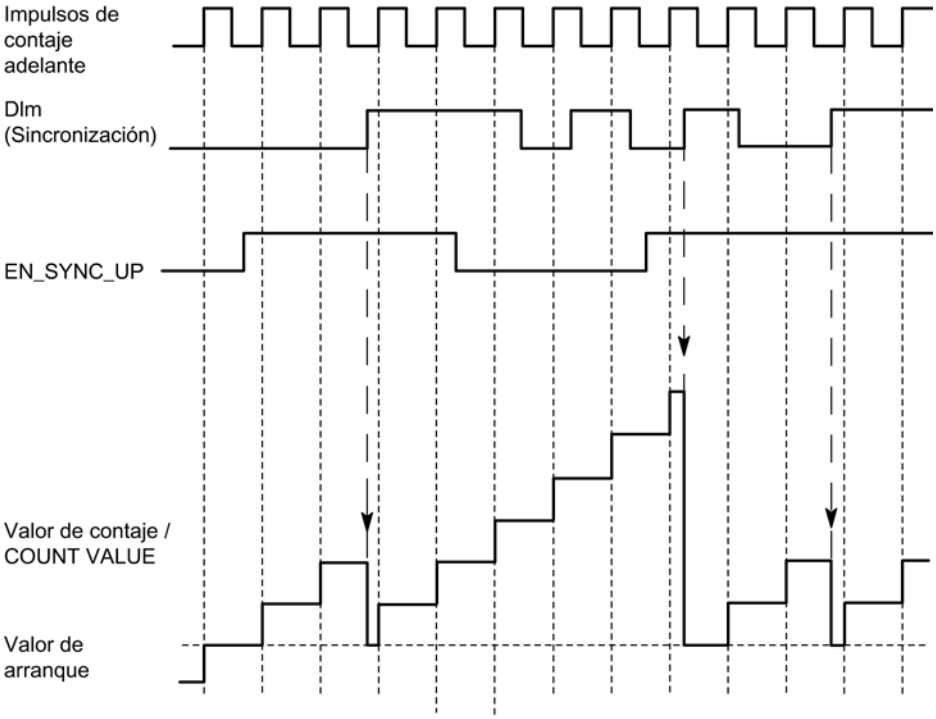
La siguiente figura muestra un ejemplo de sincronización única mediante un flanco en una entrada digital:



Tras la habilitación de la sincronización para el contaje en sentido ascendente, el contador se sincroniza con el primer flanco ascendente en la entrada digital parametrizada. Hasta que el bit de control (Página 9460) EN_SYNC_UP se desactiva y se vuelve activar, se ignora un nuevo flanco ascendente en la entrada digital. A continuación puede sincronizarse de nuevo el contador.

Sincronización periódica

La siguiente figura muestra un ejemplo de sincronización periódica mediante un flanco en una entrada digital:



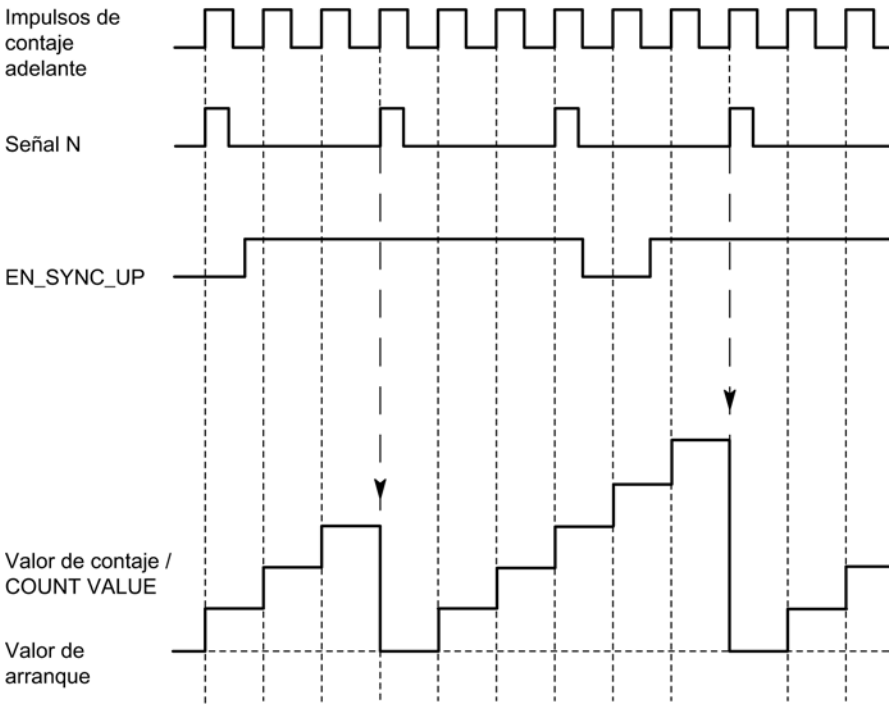
Mientras esté habilitada la sincronización para el contaje en sentido ascendente, el contador se sincroniza con cada flanco ascendente en la entrada digital parametrizada.

Sincronización en caso de señal N

La sincronización en caso de señal N en la entrada de encóder se puede disparar dependiente o independientemente del estado de una entrada digital.

Sincronización única

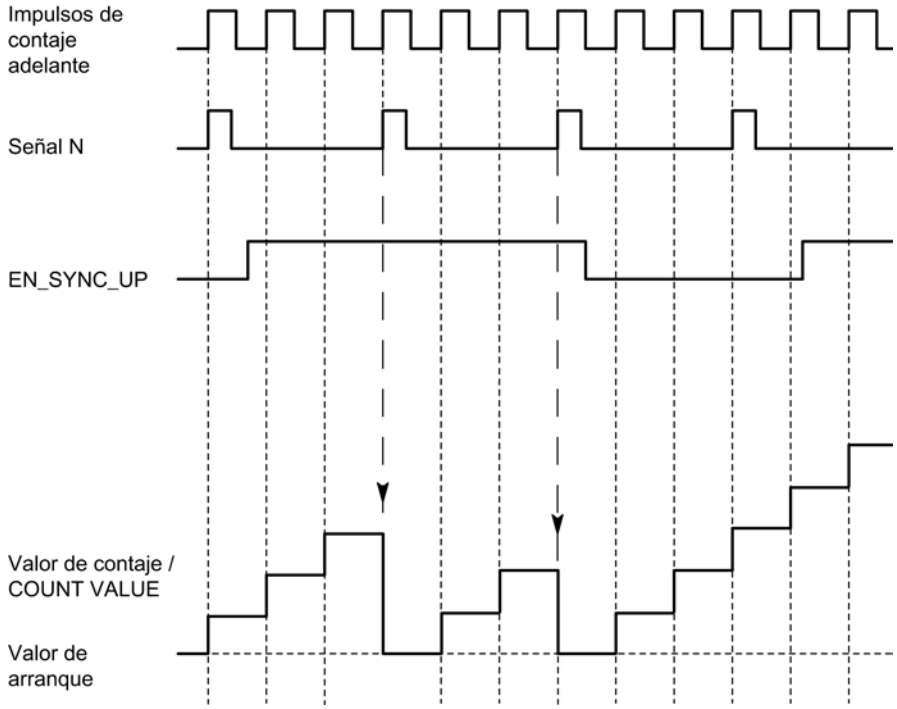
La siguiente figura muestra un ejemplo de sincronización única en caso de señal N (independientemente de una entrada digital):



Tras la habilitación de la sincronización para el contaje en sentido ascendente, el contador se sincroniza con la primera señal N. Tras desactivar y volver a activar el bit de control (Página 9460) EN_SYNC_UP , el contador puede volver a sincronizarse.

Sincronización periódica

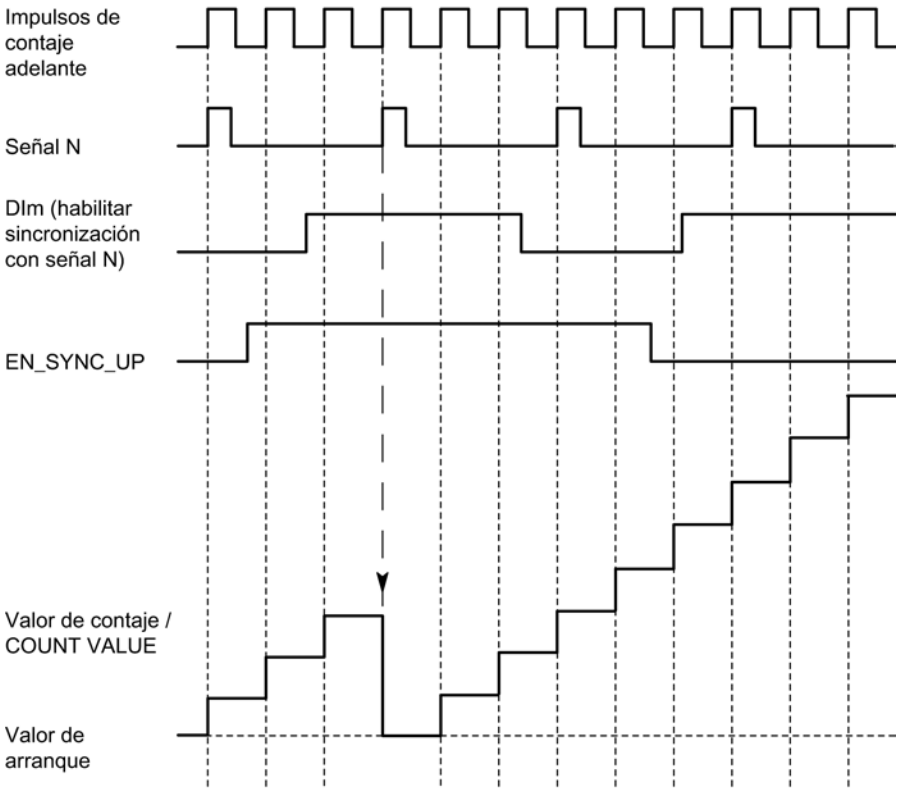
La siguiente figura muestra un ejemplo de sincronización periódica en caso de señal N:



Mientras esté habilitada la sincronización para el contaje en sentido ascendente, el contador se sincroniza con cada señal N.

Habilitación mediante una entrada digital

La siguiente figura muestra un ejemplo de sincronización periódica en caso de señal N en función del estado de una entrada digital:



Mientras esté habilitada la sincronización para el contaje en sentido ascendente y la entrada digital correspondiente esté activa, el contador se sincroniza con cada señal N. Si no se cumple una de las dos condiciones, el contador no se sincroniza con la señal N.

Valores de comparación

Valores de comparación y salidas

Descripción

Es posible definir dos valores de comparación que pueden controlar las dos salidas digitales del canal, independientemente del programa de usuario:

- Valor de comparación 0 para salida digital DQ0
- Valor de comparación 1 para salida digital DQ1

En función del modo de operación y el encóder utilizado, establezca como valores de comparación dos valores de posición, valores de contaje o valores medidos. El valor de comparación 1 debe ser superior al valor de comparación 0. Los valores de comparación son

parametrizables y pueden modificarse en tiempo de ejecución a través del programa de usuario.

Conmutación de salidas digitales desde el programa de usuario

Con los bits de control (Página 9460) TM_CTRL_DQ0 y TM_CTRL_DQ1 se controla la utilización de las salidas digitales.

Si TM_CTRL_DQm está establecido en 0, se puede controlar la salida digital correspondiente con el bit de control SET_DQm desde el programa de usuario independientemente de la función tecnológica configurada. Si TM_CTRL_DQm está establecido en 1, se habilita la función tecnológica para controlar la salida digital correspondiente.

El estado de una salida digital se notifica mediante el bit de realimentación STS_DQm correspondiente.

Conmutar en los valores de comparación en el modo de operación Contaje

Los valores de comparación se comparan con el valor de contaje o posición actual. Si el valor de contaje o posición de la condición de comparación parametrizada se cumple y la función tecnológica de la salida digital correspondiente está habilitada, se activa la salida digital. Cuando se parametriza "Entre los valores de comparación 0 y 1" para la salida digital DQ1, los dos valores de comparación repercuten en DQ1.

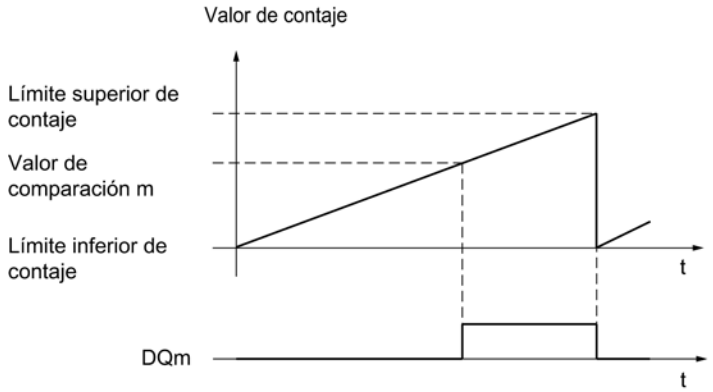
Conmutar en los valores de comparación con encóder incremental o generador de impulsos

La conmutación de una salida digital puede configurarse para que dependa de los siguientes eventos de comparación:

Ajuste entre el valor de comparación y el límite superior de contaje

La salida digital correspondiente se establece en 1 cuando:

Valor de comparación <= valor de contaje <= límite superior de contaje

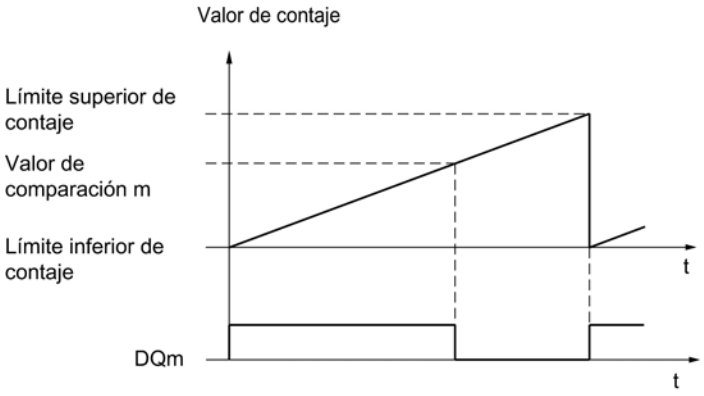


El evento de comparación es independiente del sentido de contaje.

Ajuste entre el valor de comparación y el límite inferior de contaje

La salida digital correspondiente se establece en 1 cuando:

Límite inferior de contaje <= valor de contaje <= valor de comparación



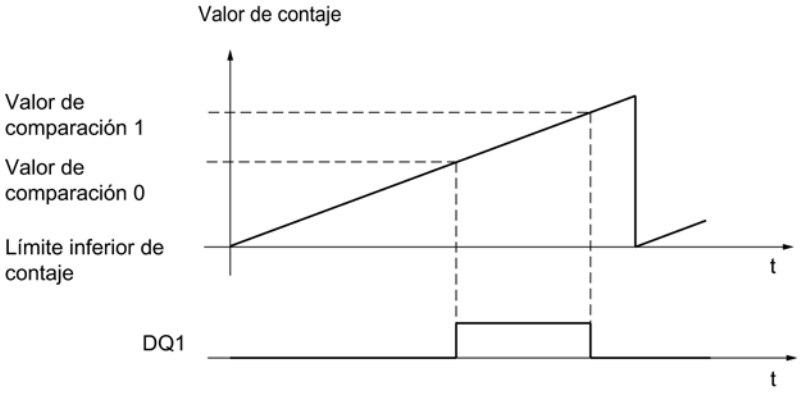
El evento de comparación es independiente del sentido de contaje.

Ajuste entre los valores de comparación 0 y 1

El evento de comparación puede parametrizarse para la salida digital DQ1 cuando para la salida digital DQ0 se ha parametrizado "Utilizada por el programa de usuario".

DQ1 se establece en 1 cuando:

Valor de comparación 0 <= valor de contaje <= valor de comparación 1



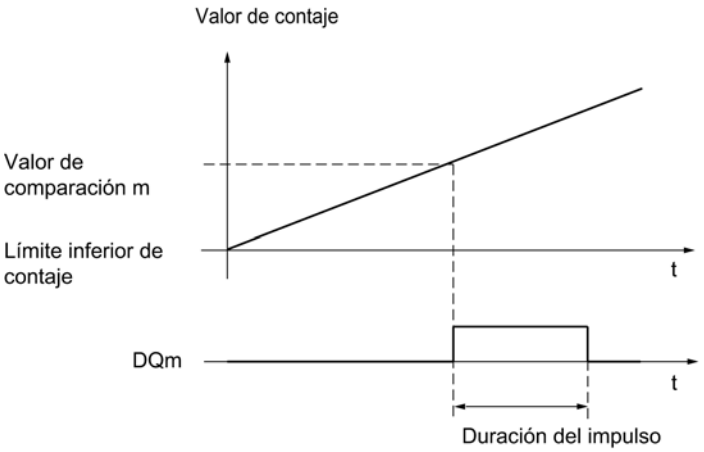
El evento de comparación es independiente del sentido de contaje.

Ajuste con valor de comparación para una duración del impulso

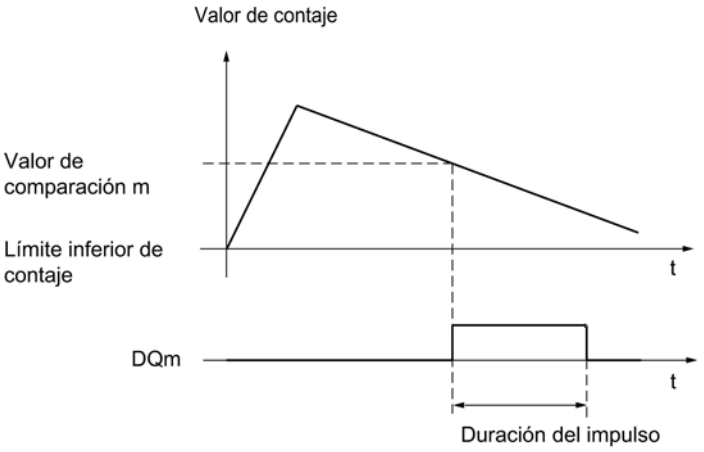
La salida digital correspondiente se ajusta a 1 durante un tiempo definido si se cumplen las siguientes condiciones:

- Valor de contaje = valor de comparación
- Sentido de contaje actual = sentido de contaje parametrizado para el evento de comparación

La siguiente figura muestra un ejemplo del evento de comparación al contar en sentido ascendente:

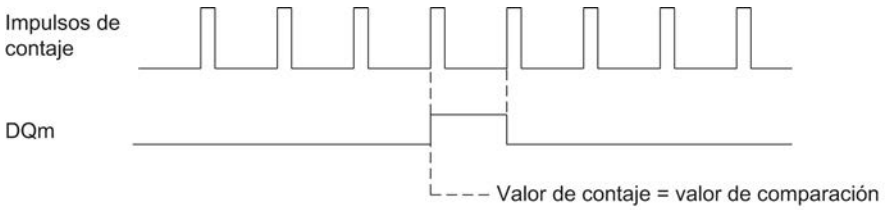


La siguiente figura muestra un ejemplo del evento de comparación al contar en sentido descendente:



Para repetir el evento de comparación, debe cambiarse el valor de contaje de modo que coincida de nuevo con el valor de comparación correspondiente.

Cuando se define "0" como duración del impulso y el valor de contaje coincide con el valor de comparación, la salida digital correspondiente se establece en 1 hasta el siguiente impulso de contaje:



Nota

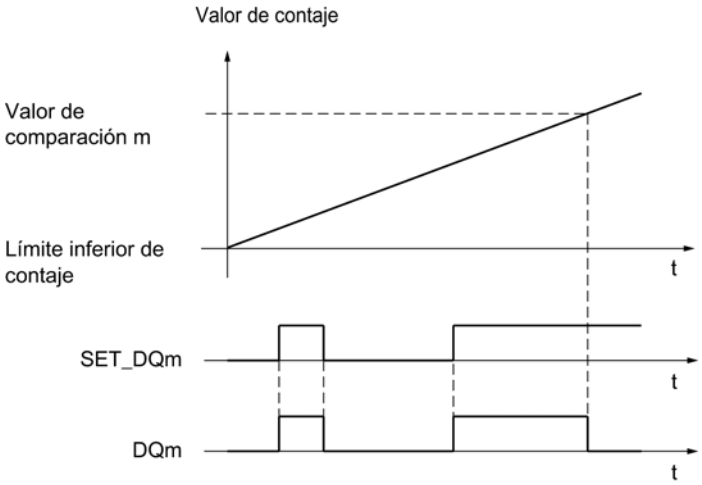
Este evento de comparación conmuta la salida digital correspondiente cuando el valor de comparación se alcanza mediante un impulso de contaje. Al establecer el valor de contaje, p. ej. por medio de la sincronización, la salida digital no se conmuta.

Ajuste mediante el programa de usuario hasta el valor de comparación

Puede establecer cada salida digital en 1 (flanco) ajustando el bit de control (Página 9460) SET_DQm. La correspondiente salida digital se establece en 0 si se produce uno de los siguientes eventos:

- Coincidencia del valor de contaje y el valor de comparación en el sentido parametrizado del evento de comparación
- Desactivación del correspondiente bit de control SET_DQm

La siguiente figura muestra un ejemplo del evento de comparación al contar en sentido ascendente:



Al establecer el bit de control SET_DQm a 0 puede desconectar la salida digital antes de que el valor de contaje coincida con el de comparación.

Nota

Si se alcanza el valor de comparación en el sentido de contaje parametrizado, el bit de realimentación EVENT_CMPm se establece independientemente del estado del bit de control SET_DQm.

El evento de comparación conmuta una salida digital cuando el valor de comparación correspondiente se alcanza mediante un impulso de contaje. Al establecer el valor de contaje, p. ej. por medio de la sincronización, la salida digital no se conmuta.

Conmutar en los valores de comparación con encóder absoluto SSI

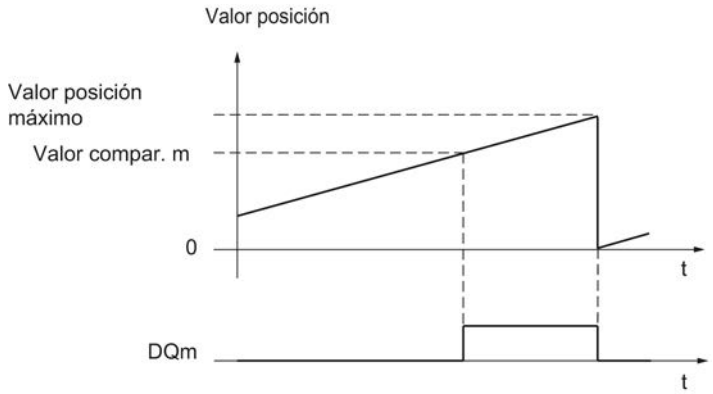
La conmutación de una salida digital puede configurarse para que dependa de los siguientes eventos de comparación:

Ajuste entre el valor de comparación y el límite superior

El límite superior coincide con el valor de posición máximo.

La salida digital correspondiente se establece en 1 cuando:

Valor de comparación <= valor de posición <= valor de posición máximo



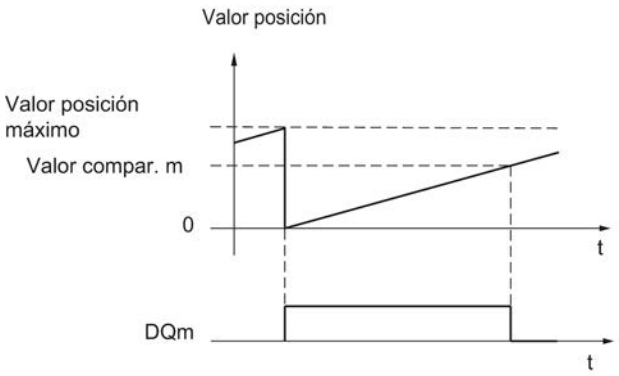
El evento de comparación es independiente del sentido del cambio del valor de posición. El valor de posición máximo depende de la resolución del encóder absoluto SSI.

Ajuste entre el valor de comparación y el límite inferior

El límite inferior coincide con el valor de posición "0".

La salida digital correspondiente se establece en 1 cuando:

0 <= Valor de posición = valor de comparación



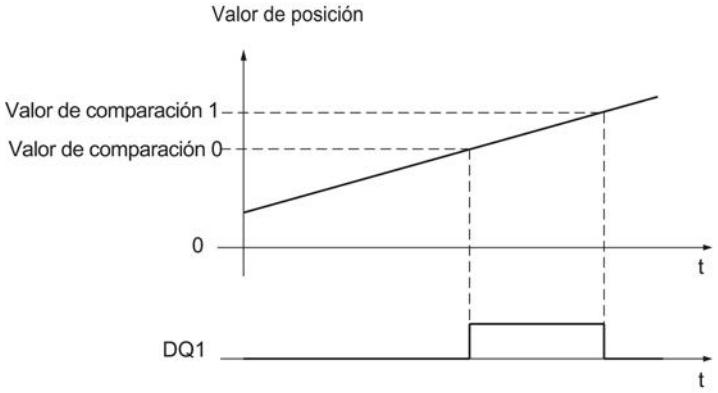
El evento de comparación es independiente del sentido del cambio del valor de posición.

Ajuste entre los valores de comparación 0 y 1

El evento de comparación puede parametrizarse para la salida digital DQ1 cuando para la salida digital DQ0 se ha parametrizado "Utilizada por el programa de usuario".

DQ1 se establece en 1 cuando:

Valor de comparación 0 <= valor de posición <= valor de comparación 1



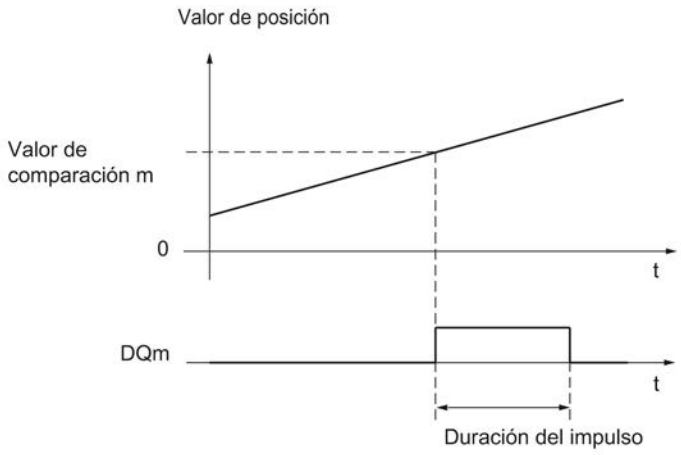
El evento de comparación es independiente del sentido del cambio del valor de posición.

Ajuste con valor de comparación para una duración del impulso

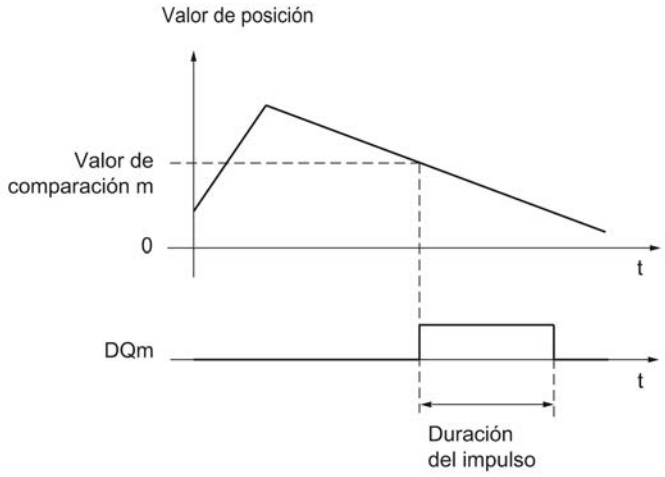
La salida digital correspondiente se ajusta a 1 durante un tiempo definido si se cumplen las siguientes condiciones:

- Coincidencia del valor de posición y del valor de comparación o cruce del valor de comparación
- Sentido actual del cambio de valor de posición = sentido parametrizado para el evento de comparación

La siguiente figura muestra un ejemplo del evento de comparación en sentido ascendente:



La siguiente figura muestra un ejemplo del evento de comparación en sentido descendente:



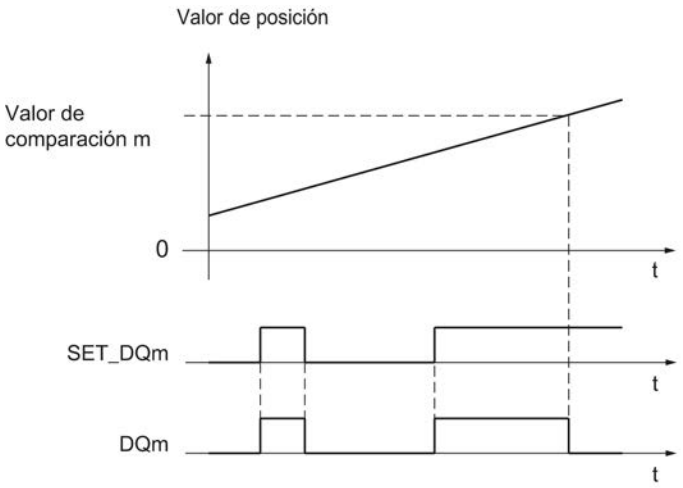
Para repetir el evento de comparación, debe cambiarse el valor de posición de modo que coincida de nuevo con el valor de comparación correspondiente o lo cruce.

Ajuste mediante el programa de usuario hasta el valor de comparación

Puede establecer cada salida digital en 1 (flanco) ajustando el bit de control (Página 9460) SET_DQm. La correspondiente salida digital se establece en 0 si se produce uno de los siguientes eventos:

- Coincidencia del valor de posición y del valor de comparación o cruce del valor de comparación en el sentido parametrizado del evento de comparación
- Desactivación del correspondiente bit de control SET_DQm

La siguiente figura muestra un ejemplo del evento de comparación en sentido ascendente:



Al establecer el bit de control SET_DQm a 0 puede desconectar la salida digital antes de que el valor de posición coincida con el de comparación o lo cruce.

Nota

Si se alcanza o se cruza el valor de comparación en el sentido parametrizado, el bit de realimentación EVENT_CMPm se establece independientemente del estado del bit de control SET_DQm.

Conmutar en los valores de comparación en el modo de operación Medición

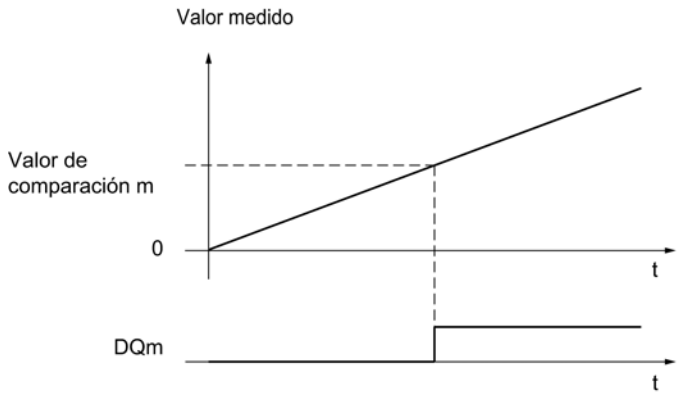
Los valores de comparación se comparan con el valor medido actual. Si el valor medido de la condición de comparación parametrizada se cumple y la función tecnológica de la salida digital correspondiente está habilitada, se activa la salida digital. Cuando se parametriza "Entre los valores de comparación 0 y 1" o "No comprendido entre valor de comparación 0 y 1" para la salida digital DQ1, los dos valores de comparación repercuten en DQ1.

La conmutación de una salida digital puede configurarse para que dependa de los siguientes eventos de comparación:

Ajuste por encima del valor de comparación

La salida digital correspondiente se establece en 1, cuando:

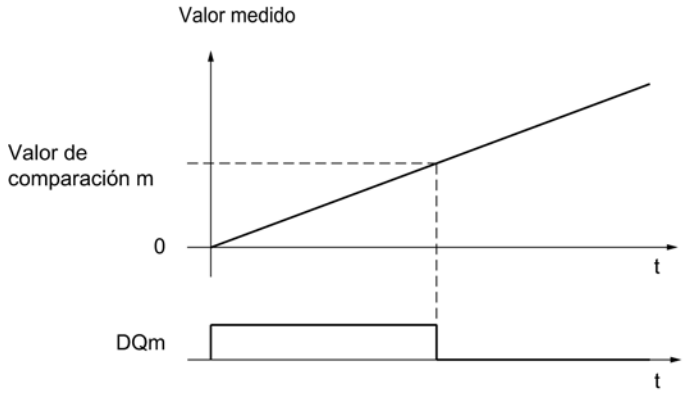
Valor de medida >= valor de referencia



Ajuste por debajo del valor de comparación

La salida digital correspondiente se establece en 1, cuando:

Valor de medida <= valor de referencia

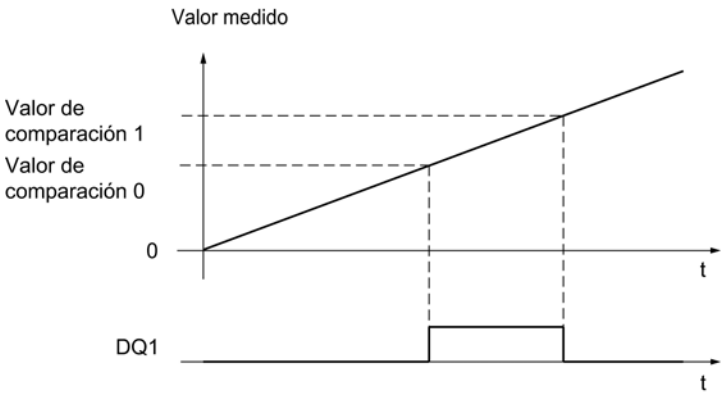


Ajuste entre los valores de comparación 0 y 1

El evento de comparación puede parametrizarse para la salida digital DQ1 cuando se ha parametrizado "Utilizada por el programa de usuario" para la salida digital DQ0.

DQ1 se establece en 1 cuando:

Valor de comparación 0 <= valor medido <= valor de comparación 1

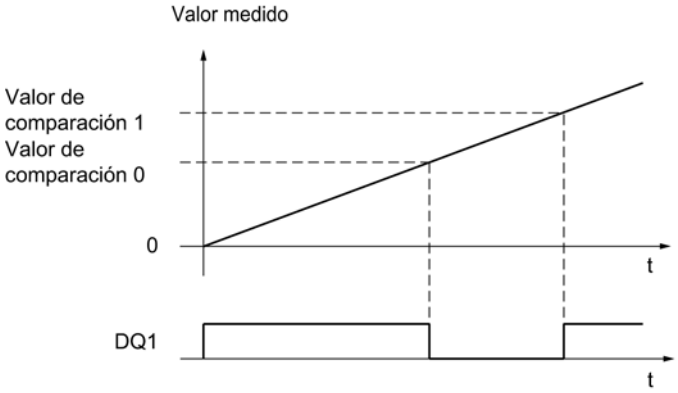


Sin ajuste entre los valores de comparación 0 y 1

El evento de comparación puede parametrizarse para la salida digital DQ1 cuando se ha parametrizado "Utilizada por el programa de usuario" para la salida digital DQ0.

DQ1 se establece en 1 cuando:

Valor de comparación 1 <= valor medido <= valor de comparación 0



Determinación de medidas

Sinopsis de las funciones de medición

Existen las siguientes funciones de medición:

Tipo de medición (Página 9393)	Descripción
Medición de frecuencia	A partir del cronograma de los impulsos de contaje o cambios de los valores de posición se calcula la frecuencia media en un intervalo de medición, y dicha frecuencia se devuelve como número en coma flotante en hercios.
Medición del período	A partir del cronograma de los impulsos de contaje o los cambios de los valores de posición se calcula el período medio en un intervalo de medición, y dicho período medio se devuelve como número en coma flotante en segundos.
Medición de velocidad	A partir del cronograma de los impulsos de contaje o los cambios de los valores de posición y otros parámetros se calcula la velocidad media en un intervalo de medición, y dicha velocidad media se devuelve en la unidad parametrizada.

Los valores medidos y de contaje están disponibles simultáneamente en la interfaz de realimentación.

Tiempo de actualización

Puede parametrizar como tiempo de actualización el intervalo con el que el módulo tecnológico actualiza cíclicamente los valores medidos. A través de tiempos de actualización más dilatados se pueden filtrar magnitudes inestables y se puede aumentar la precisión de medida.

Control de puerta en encóders incrementales y generadores de impulsos

La apertura y el cierre de la puerta interna definen la ventana de tiempo en la que se capturan los impulsos de contaje. El tiempo de actualización es asíncrono al abrir la puerta, es decir, el tiempo de actualización no se inicia con la apertura de la puerta. Después del cierre de la puerta interna se devuelve el último valor medido.

Determinación de medidas con encóder incremental o generador de impulsos

Rangos de medición

Rangos de medición

Las funciones de medición tienen los siguientes límites de rango de medida:

Tipo de medición	Límite inferior del rango de medida	Límite superior del rango de medida
Medición de frecuencia	0,04 Hz	800 kHz* / 4 MHz**
Medición del período	1,25 µs* / 0,25 µs**	25 s
Medición de velocidad	En función del número parametrizado de "Incrementos por unidad" y "Base de tiempo para medición de velocidad"	

* Válido para encóder incremental de 24 V y Evaluación de la señal "cuádruple".

** Válido para encóder incremental RS422 y Evaluación de la señal "cuádruple".

Todos los valores medidos se devuelven como valores con signo. El signo indica si el valor de contaje ha aumentado o disminuido durante el intervalo.

Intervalo de medida

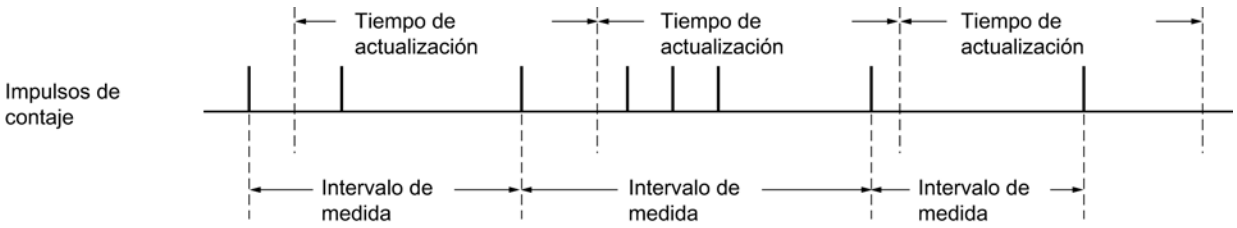
Principio de medición

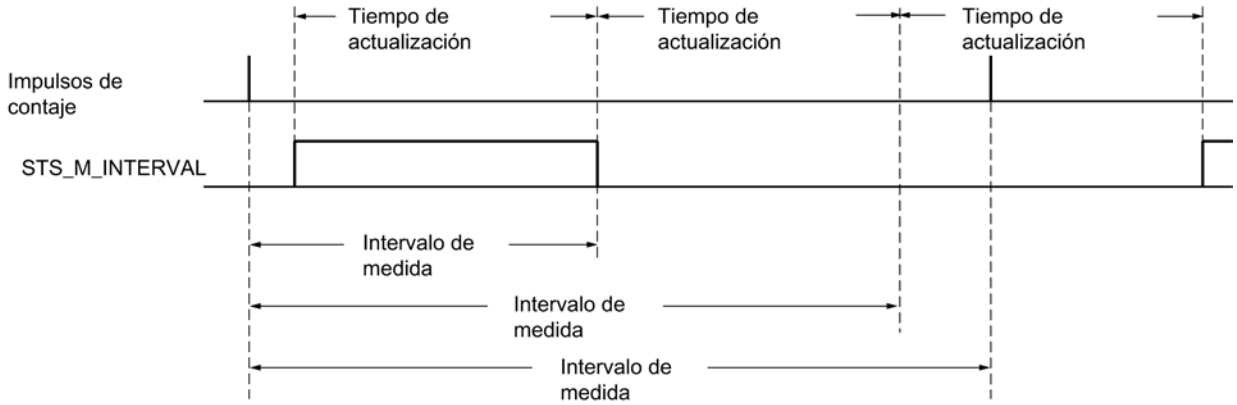
El módulo tecnológico asigna un valor de tiempo a cada impulso de contaje. El intervalo de medida se define como el intervalo que transcurre entre el último impulso de contaje antes y durante el tiempo de actualización precedente. Para calcular una magnitud se evalúan el intervalo de medida y el número de impulsos del intervalo de medida.

Cuando no se produce ningún impulso de contaje en el tiempo de actualización, se produce una adaptación dinámica del intervalo de medida. En este caso, se adopta un impulso al final del tiempo de actualización y el intervalo de medida se calcula desde ahí hasta el último impulso producido. A continuación, el número de impulsos es 1.

El bit de realimentación STS_M_INTERVAL notifica si se ha producido un impulso de contaje en el intervalo de medida precedente. De ese modo, se puede distinguir entre un impulso de contaje adoptado y uno efectivo.

Las siguientes figuras muestran el principio de medición y la adaptación dinámica del intervalo de medida:





Tipos de medición

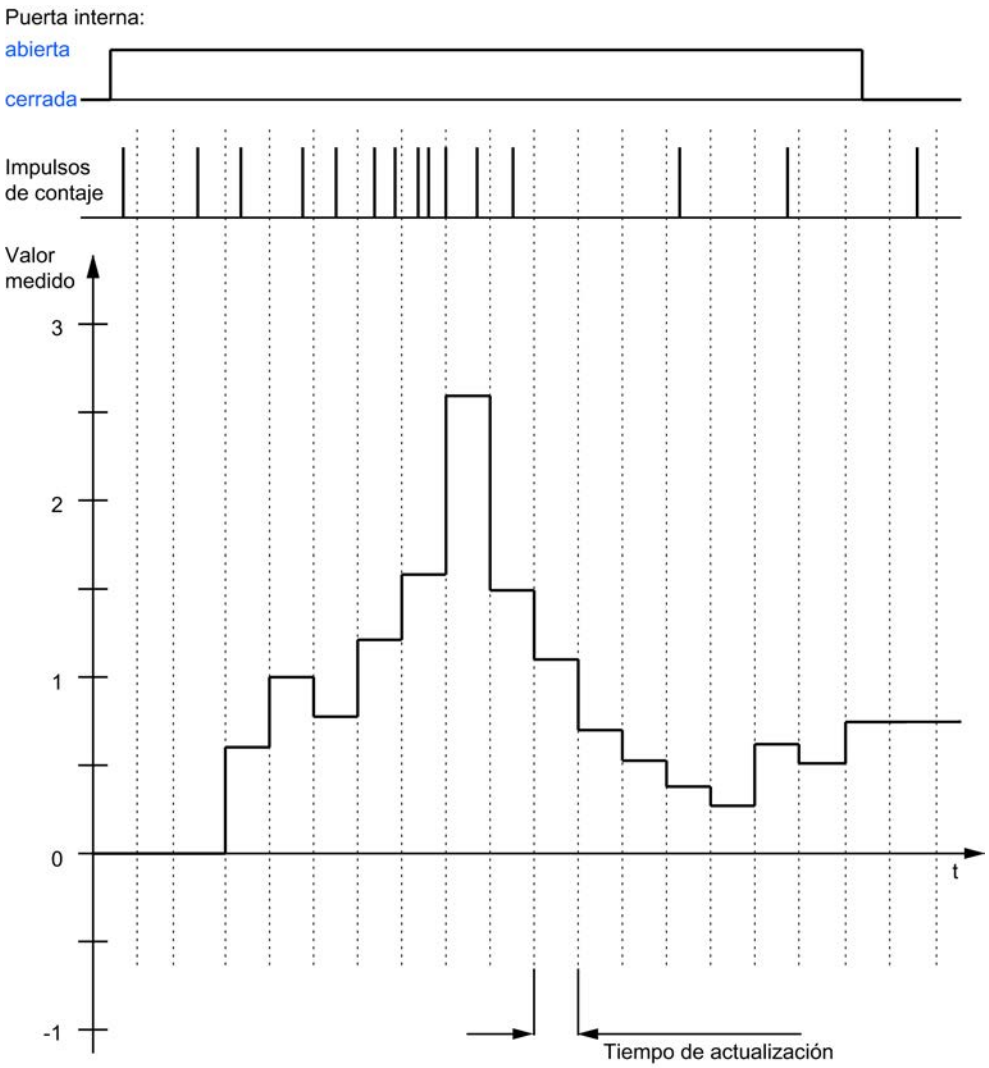
Medición de frecuencia

En el tiempo que transcurre hasta el primer valor de medida existente se devuelve el valor "0".

El proceso de medición empieza con el primer impulso captado tras la apertura de la puerta interna. El primer valor medido no se calcula hasta después del segundo impulso.

Cada vez que transcurre el tiempo de actualización, el valor medido se actualiza en la interfaz de realimentación (Página 9463). Cuando la puerta interna está cerrada, el proceso de medición se detiene y el valor medido ya no se actualiza.

La siguiente figura muestra un ejemplo de la medición de la frecuencia con un tiempo de actualización de 1 s:



Medición del período

En la medición del período, el valor inverso de la frecuencia se emite como valor medido.

En el tiempo que transcurre hasta el primer valor de medida existente se devuelve el valor "25 s".

Medición de velocidad

Para la medición de la velocidad, la frecuencia normalizada se emite como valor medido. La normalización se parametriza a través de la base de tiempo y el número de incrementos que el encóder emite por unidad.

Ejemplo:

Su encóder emite 4000 incrementos por metro. La velocidad debe medirse en metros por minuto.

En este caso, debe parametrizar 4000 Incrementos por unidad y una base de tiempo de un minuto.

Determinación de medidas con encóder absoluto SSI

Rangos de medición

Rangos de medición para el encóder absoluto SSI

Las funciones de medición tienen los siguientes límites de rango de medida:

Tipo de medición	Límite inferior del rango de medida	Límite superior del rango de medida
Medición de frecuencia	0,04 Hz	4 MHz
Medición del período	0,25 µs	25 s
Medición de velocidad	En función del número parametrizado de "Incrementos por unidad" y "Base de tiempo para medición de velocidad"	

Todos los valores medidos se devuelven como valores con signo. El signo indica si el valor de posición ha aumentado o disminuido durante el intervalo.

Intervalo de medida

Principio de medición

El módulo tecnológico asigna un valor de tiempo a cada telegrama SSI. El intervalo de medida se define como el intervalo que transcurre entre el último telegrama SSI con un cambio del valor de posición antes y durante el tiempo de actualización precedente. Para el cálculo de una magnitud se evalúan el intervalo de medición y el cambio total del valor de posición en el intervalo de medición. El cambio total del valor de posición en un intervalo de medición coincide con el número de incrementos del encóder en el mismo intervalo de medición.

Cuando no se produce ningún cambio del valor de posición en el tiempo de actualización, se produce una adaptación dinámica del intervalo de medición. En este caso, se adopta un cambio del valor de posición al final del tiempo de actualización y el intervalo de medición se calcula desde ahí hasta el último telegrama SSI con un cambio del valor de posición. Entonces, el cambio del valor de posición es 1.

El bit de realimentación STS_M_INTERVAL notifica si se ha producido un cambio del valor de posición en el intervalo de medición precedente. De ese modo, se puede distinguir entre un cambio del valor de posición adoptado y uno efectivo. Si el módulo tecnológico no puede calcular ningún valor medido debido a un rebase de los límites del rango de medida, no se establece el bit de realimentación STS_M_INTERVAL.

Tipos de medición

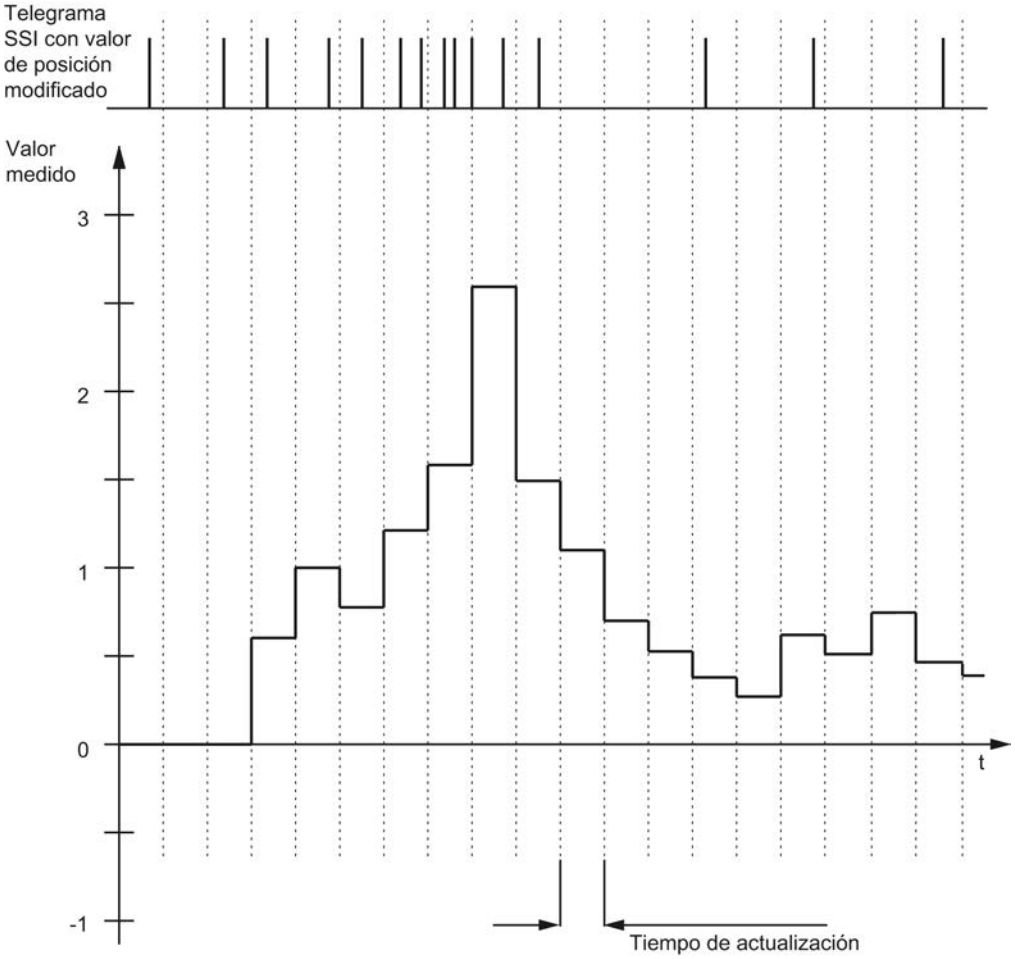
Medición de frecuencia

En el tiempo que transcurre hasta el primer valor medido existente se devuelve el valor "0,0".

El proceso de medición empieza con el primer cambio del valor de posición captado. El primer valor medido no se calcula hasta después del segundo cambio del valor de posición captado.

Cada vez que transcurre el tiempo de actualización, el valor medido se actualiza en la interfaz de realimentación (Página 9463).

La siguiente figura muestra un ejemplo de la medición de la frecuencia con un tiempo de actualización de 1 s:



Medición del período

En la medición del período, el valor inverso de la frecuencia se emite como valor medido.

En el tiempo que transcurre hasta el primer valor de medida existente se devuelve el valor "25 s".

Medición de velocidad

Para la medición de la velocidad, la frecuencia normalizada se emite como valor medido. La normalización se parametriza a través de la base de tiempo y el número de incrementos que el encóder emite por unidad.

Ejemplo:

Su encóder absoluto SSI funciona con una resolución de 12 bits por vuelta y emite 4096 incrementos por vuelta. La velocidad debe medirse en vueltas por minuto.

En este caso, debe parametrizar 4096 Incrementos por unidad y una base de tiempo de un minuto.

Histéresis

Histéresis con encóder incremental o generador de impulsos

Descripción

Con la histéresis se puede definir un rango para los valores de comparación en el que las salidas digitales no deban conmutarse de nuevo antes de que el valor de contaje haya salido una vez de este rango.

Con leves movimientos del encóder, el valor de contaje puede oscilar alrededor de un valor concreto. Si en el rango de oscilación hay un valor de comparación o un límite de contaje, la salida digital correspondiente se activa y desactiva con la frecuencia conveniente sin aplicar una histéresis. La histéresis evita esta conmutación no deseada, así como alarmas de proceso parametrizadas cuando tiene lugar un evento de comparación.

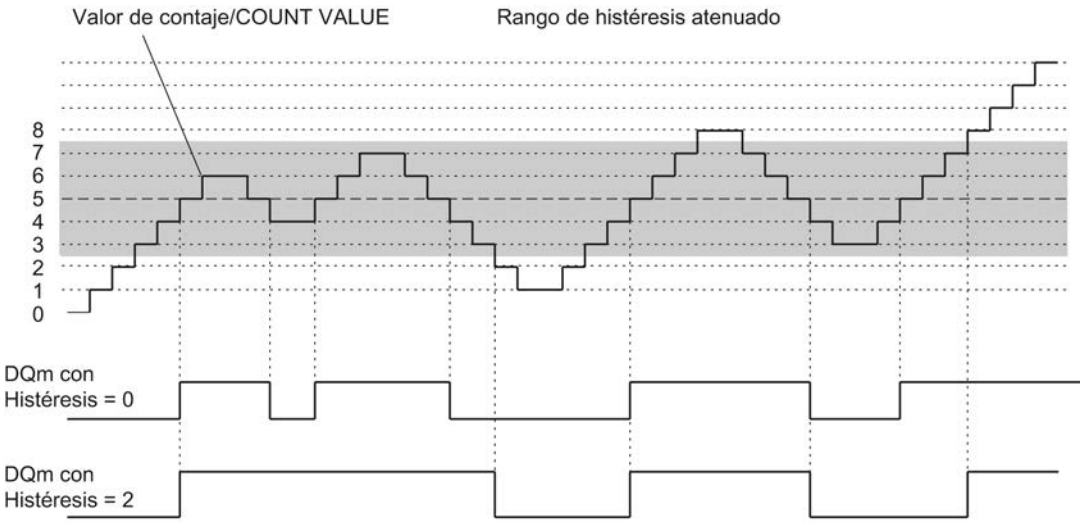
La histéresis se activa si el correspondiente valor de comparación se alcanza con un impulso de contaje. Si el valor de contaje se ajusta al valor de arranque durante una histéresis activa, la histéresis se desactiva.

El rango de histéresis finaliza, independientemente del valor de histéresis, en el límite inferior o superior de contaje.

Funcionamiento

La siguiente figura muestra un ejemplo de histéresis con la siguiente parametrización:

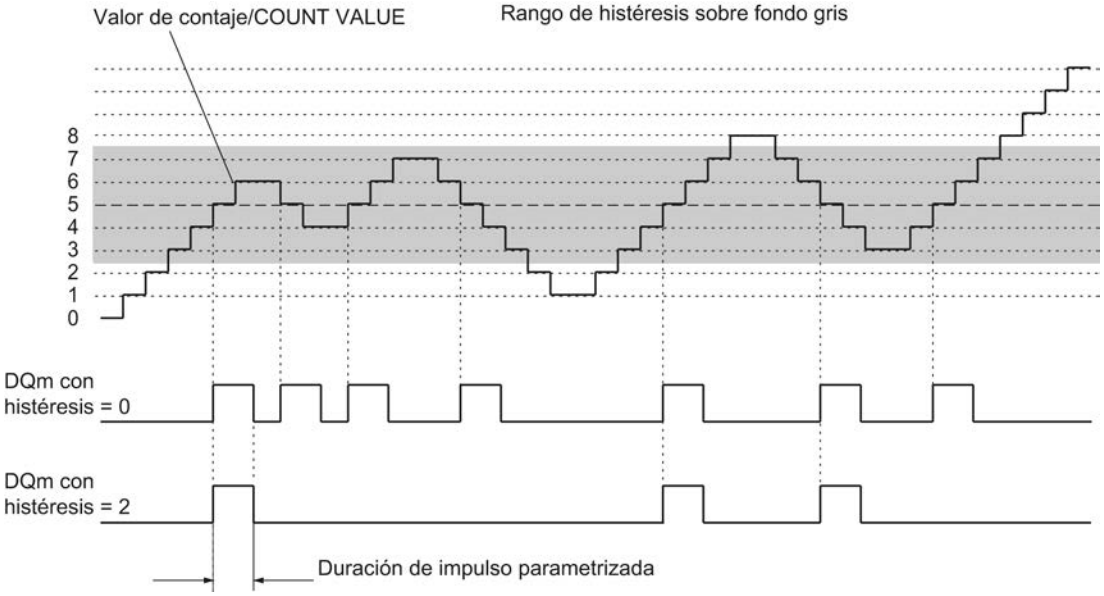
- Ajuste de una salida digital entre el valor de comparación y el límite de contaje superior
- Valor de comparación = 5
- Histéresis = 0 o 2



La histéresis se activa al alcanzar el valor de contaje 5. Con la histéresis activada el resultado de la comparación permanece invariable. La histéresis se desactiva al alcanzar los valores de contaje 2 u 8.

La siguiente figura muestra un ejemplo de histéresis con la siguiente parametrización:

- Ajuste con valor de comparación para una duración del impulso
- Valor de comparación = 5
- Comparación en ambos sentidos de contaje
- Histéresis = 0 o 2



Histéresis con encóder absoluto SSI

Descripción

Con la histéresis se puede definir un rango para los valores de comparación en el que las salidas digitales no deban conmutarse de nuevo antes de que el valor de posición haya salido una vez de este rango.

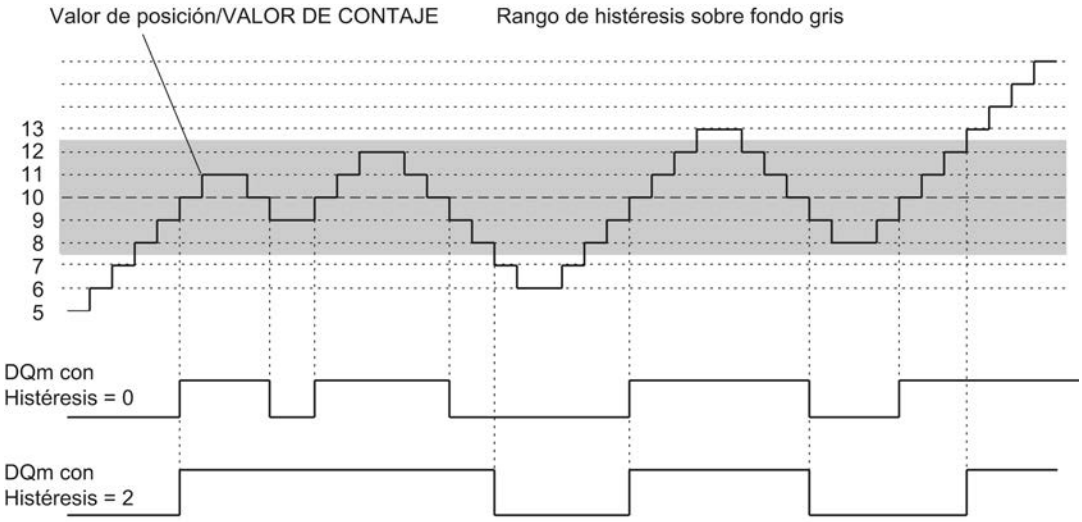
Con leves movimientos del encóder, el valor de posición puede oscilar alrededor de un valor concreto. Si en el rango de oscilación hay un valor de comparación, "0" o el valor de posición máximo, la salida digital correspondiente se activa y desactiva con la frecuencia conveniente sin aplicar una histéresis. La histéresis evita esta conmutación no deseada, así como alarmas de proceso parametrizadas cuando tiene lugar un evento de comparación.

El rango de histéresis finaliza, independientemente del valor de la histéresis, en "0" y en el valor de posición máximo.

Funcionamiento

La siguiente figura muestra un ejemplo de histéresis con la siguiente parametrización:

- Ajuste de una salida digital entre el valor de comparación y el límite superior
- Valor de comparación = 10
- Histéresis = 0 o 2

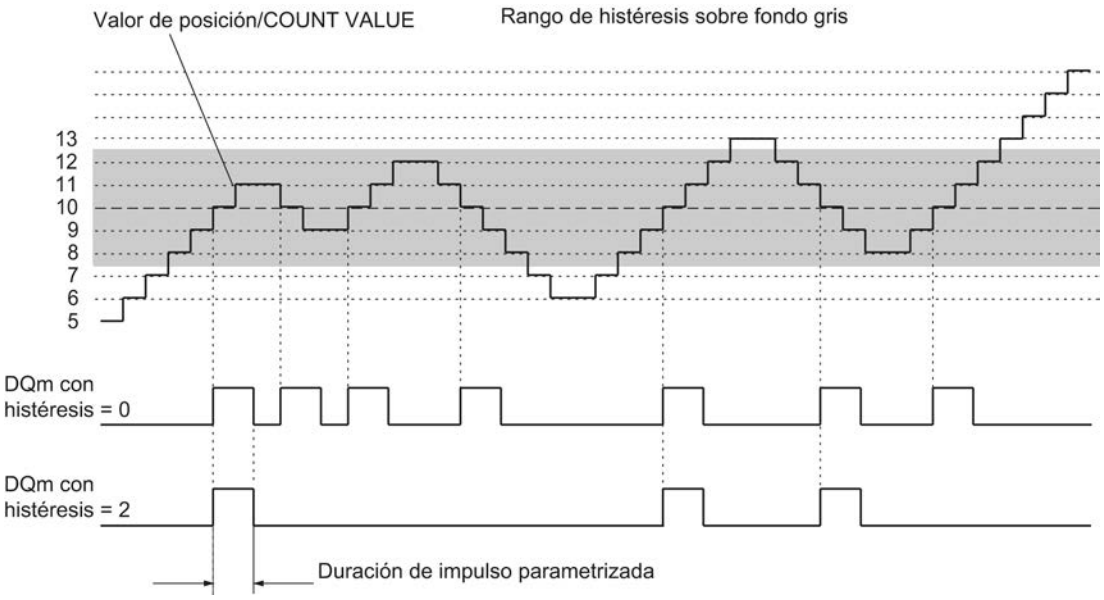


La histéresis se activa al alcanzarse el valor de posición 10. Con la histéresis activada el resultado de la comparación permanece invariable. La histéresis se desactiva al alcanzarse los valores de posición 7 o bien 13.

La siguiente figura muestra un ejemplo de histéresis con la siguiente parametrización:

- Ajuste con valor de comparación para una duración del impulso
- Valor de comparación = 10

- Comparación en ambos sentidos de los cambios del valor de posición
- Histéresis = 0 o 2



Alarmas

Alarma de proceso

El módulo tecnológico puede disparar una alarma de proceso en la CPU, entre otros, cuando tiene lugar un evento de comparación, cuando se rebasan el límite superior o inferior, cuando el contador pasa por cero y/o cuando cambia el sentido de contaje (inversión de sentido). Es posible definir qué eventos deben generar una alarma de proceso durante el funcionamiento.

Alarma de diagnóstico

El módulo tecnológico puede disparar alarmas de diagnóstico si se producen errores. Debe habilitar las alarmas de diagnóstico para determinados errores en la configuración del dispositivo. Infórmese con el manual de producto del módulo tecnológico sobre los eventos que pueden disparar una alarma de diagnóstico durante el funcionamiento.

Lectura de posición para Motion Control

Descripción

Se puede utilizar el módulo tecnológico con S7-1500 Motion Control para la lectura de posición.

Para ello, en la Configuración del dispositivo del módulo tecnológico en STEP 7 (TIA Portal) se selecciona el modo de funcionamiento "Lectura de posición para Motion Control". Así se

reducen las posibilidades de parametrización a los parámetros necesarios. El modo de funcionamiento se aplica automáticamente a todos los canales del módulo tecnológico.

Si se emplea un encóder incremental o un generador de impulsos, la lectura de posición se basa en la función de contaje del módulo tecnológico. Con un encóder absoluto SSI, el valor absoluto se lee a través de una interfaz serie síncrona, se acondiciona de acuerdo con la parametrización y se pone a disposición de S7-1500 Motion Control.

Señales del encóder

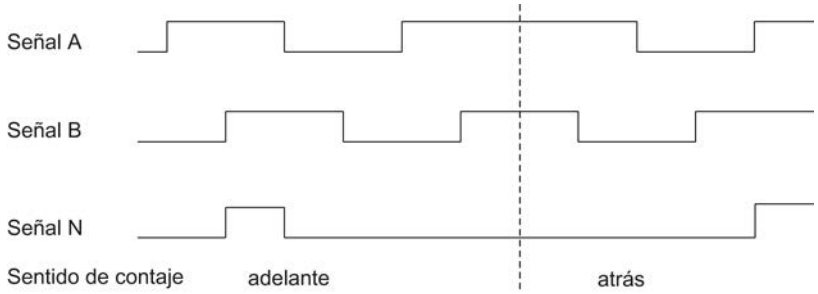
Señales de contaje de 24 V y TTL

Señales de contaje de encóders incrementales de 24 V y TTL

El encóder incremental de 24 V suministra al módulo tecnológico las señales de 24 V A, B y N. Las señales A y B están desfasadas 90° entre sí. Puede conectar también el encóder incremental sin señal N.

En un encóder incremental de 24 V se utilizan las señales A y B para contar. La señal N sirve, en la parametrización correspondiente, para poner el contador al valor de arranque o guardar el valor de contaje actual en el valor de Capture.

La siguiente figura muestra un ejemplo del cronograma de las señales de un encóder incremental de 24 V:



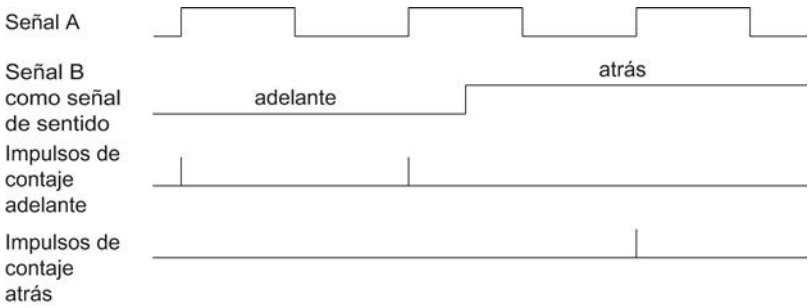
El módulo tecnológico detecta el sentido de contaje por la secuencia de los flancos en las señales A y B. Puede definir una inversión del sentido de contaje.

Señales de contaje de generadores de impulsos de 24 V y TTL sin/con señal de sentido

El encóder, por ejemplo un detector de proximidad (BERO) o una fotocélula, solo proporciona una señal de contaje, que debe conectarse a la conexión A del contador.

Adicionalmente puede conectar a la conexión B del contador una señal para detectar el sentido. Si el encóder no proporciona la señal correspondiente, se puede predefinir el sentido de contaje con el programa de usuario a través de la interfaz de control.

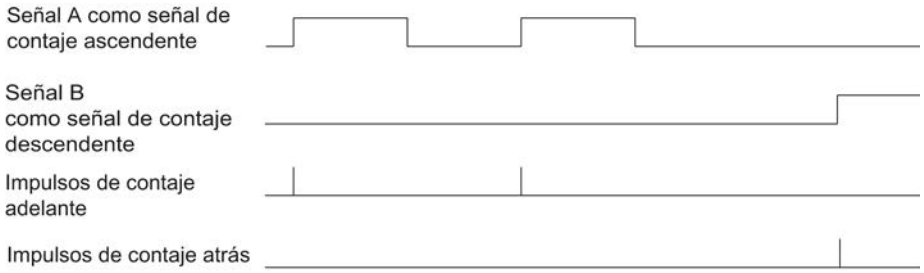
La siguiente figura muestra un ejemplo del cronograma de las señales de un encóder de impulsos de 24 V con señal de sentido y los impulsos de contaje resultantes:



Señales de contaje de generadores de impulsos de 24 V y TTL con señal de contaje ascendente/descendente

La señal de contaje ascendente se conecta al conector A. La señal de contaje descendente se conecta al conector B.

La siguiente figura muestra un ejemplo del cronograma de las señales de un generador de impulsos con señal de contaje ascendente/descendente y los impulsos de contaje resultantes:



Sensor tipo PNP/NPN para señales de contaje de 24 V

En las entradas de contaje puede conectar los siguientes encóders o sensores:

- Sensor tipo PNP (en fuente):
  Las entradas A, B y N se conmutan a 24VDC .
- Sensor tipo NPN (en sumidero):
  Las entradas A, B y N se conmutan a masa M .
- Tipo push/pull (tipo PNP y NPN):
  Las entradas A, B y N se conmutan de forma alternativa a 24VDC y a masa M .

Vigilancia de las señales de encóder

El módulo tecnológico vigila la posible rotura de hilo en las señales de encóders de 24 V tipo push/pull. El módulo tecnológico vigila posibles tensiones erróneas en las señales TTL.

Si se habilitan la alarma de diagnóstico en la configuración del dispositivo, el módulo tecnológico dispara una alarma de diagnóstico en caso de error en las señales del encóder.

Señales de contaje RS422

Señales de contaje de encóders incrementales RS422

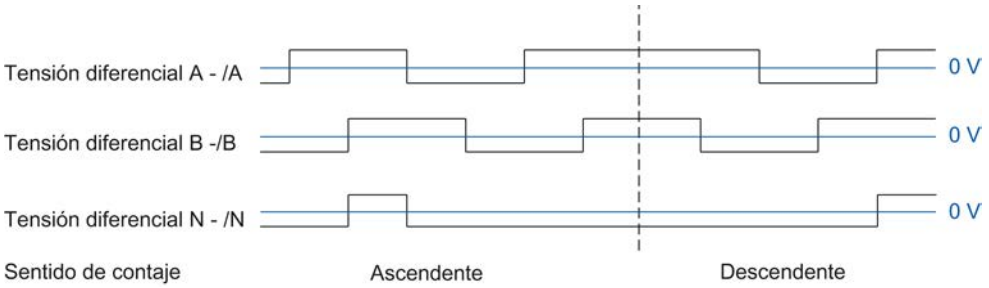
El encóder incremental RS422 proporciona al módulo tecnológico las siguientes señales diferenciales:

- A y /A
- B y /B
- N y /N

La información de señal en señales RS422 se codifica en su tensión diferencial entre A y /A, B y /B o N y /N. Las señales A y B están desfasadas 90° entre sí. Puede conectar también el encóder incremental sin señal N.

En un encóder incremental RS422 se emplean las señales A y B para contar. La señal N sirve, en la parametrización correspondiente, para poner el contador al valor de arranque o guardar el valor de contaje actual como valor de Capture.

La siguiente figura muestra un ejemplo del cronograma de las señales de un encóder incremental RS422:



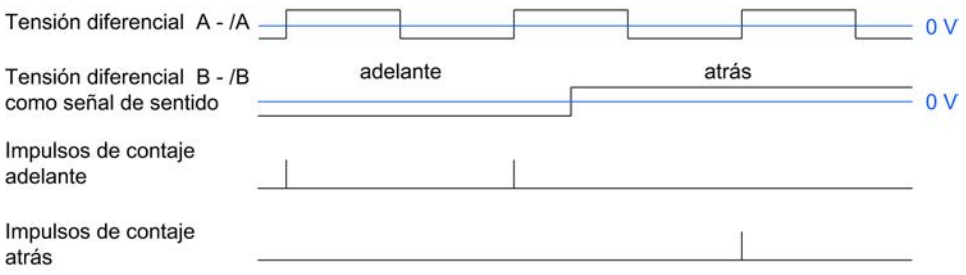
El módulo tecnológico detecta el sentido de contaje por la secuencia de los flancos en las señales A y B. Puede definir una inversión del sentido de contaje.

Señales de contaje de encóders de impulsos RS422 sin/con señal de sentido

El encóder, por ejemplo una fotocélula, solo proporciona una señal de contaje, que debe conectarse a la conexión A.

Adicionalmente puede conectar a la conexión B una señal para detectar el sentido. Si el encóder no proporciona la señal correspondiente, se puede predefinir el sentido de contaje con el programa de usuario a través de la interfaz de control.

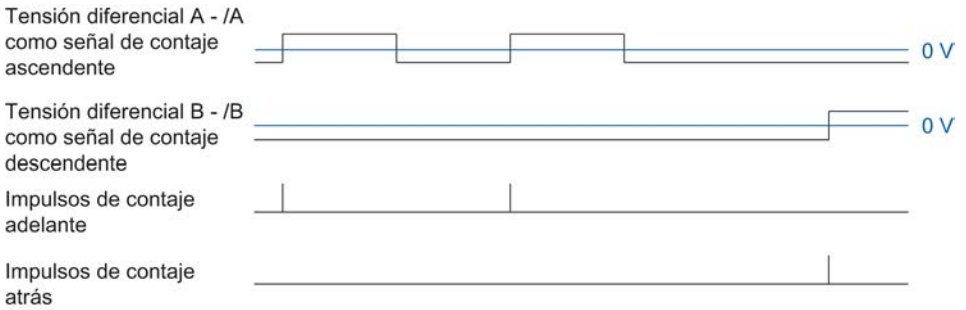
La siguiente figura muestra un ejemplo del cronograma de las señales de un generador de impulsos RS422 con señal de sentido y los impulsos de contaje resultantes:



Señales de contaje de encóders de impulsos RS422 con señal de contaje ascendente/descendente

La señal de contaje ascendente se conecta a las conexiones A. La señal de contaje descendente se conecta a las conexiones B.

La siguiente figura muestra un ejemplo del cronograma de las señales de un generador de impulsos RS422 con señal de contaje ascendente/descendente y los impulsos de contaje resultantes:



Vigilancia de las señales de encóder

El módulo tecnológico vigila posibles roturas de hilo, cortocircuitos y tensiones erróneas en las señales RS422.

Si se habilitan la alarma de diagnóstico en la configuración del dispositivo, el módulo tecnológico dispara una alarma de diagnóstico en caso de error en las señales del encóder.

Señales SSI

Señales de encóders absolutos SSI

El encóder absoluto SSI y el módulo tecnológico se comunican a través de las señales de datos SSI DAT y /DAT (D) y las señales en contrafase SSI CLK y /CLK (C). SSI utiliza el estándar de señal RS422. La información de señal está codificada en la tensión diferencial correspondiente entre C y /C, así como D y /D.

Vigilancia de las señales de encóder y los telegramas SSI

El módulo tecnológico vigila posibles roturas de hilo, cortocircuitos o tensiones erróneas en las señales de un encóder absoluto SSI. Además, el módulo tecnológico vigila posibles errores en los telegramas SSI.

Si se habilitan las alarmas de diagnóstico en la configuración de dispositivo, el módulo tecnológico dispara una alarma de diagnóstico en caso de error en las señales del encóder o el telegrama SSI.

Evaluación de señales incrementales

Vista general

El contador del módulo tecnológico cuenta los flancos de las señales de encóder A y B. En los encóders incrementales con señales A y B desfasadas entre sí, puede elegir entre la evaluación simple y la múltiple para aumentar la resolución.

Es posible parametrizar las siguientes evaluaciones de la señal:

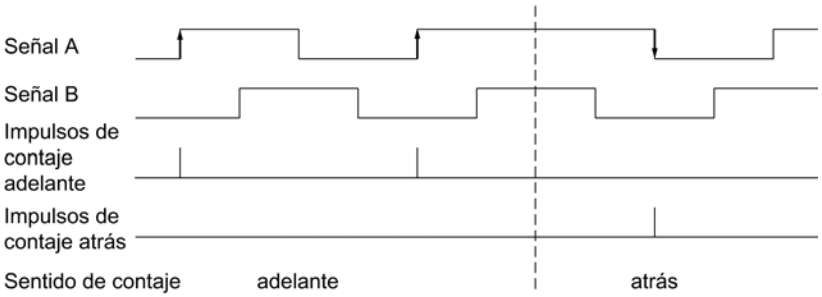
- Evaluación simple (Página 9405)
- Doble evaluación (Página 9406)
- Evaluación cuádruple (Página 9406)

Evaluación simple

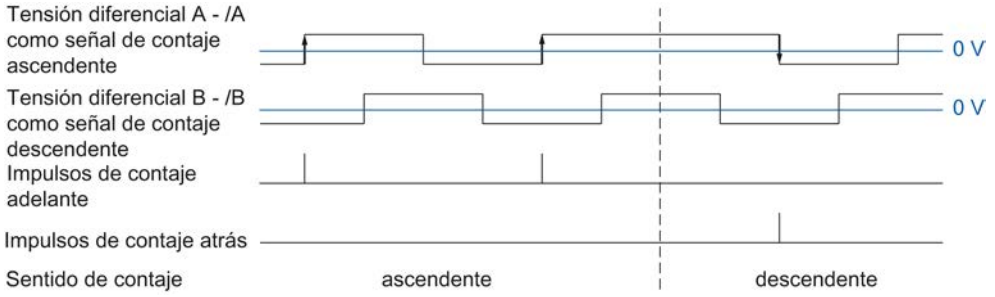
En la evaluación simple se evalúan el flanco ascendente y descendente en la señal A durante un nivel bajo en la señal B.

Los impulsos de contaje en sentido ascendente se generan con un flanco ascendente en la señal A durante un nivel bajo en la señal B. Los impulsos de contaje en sentido descendente se generan con un flanco descendente en la señal A durante un nivel bajo en la señal B.

La figura siguiente muestra un ejemplo de evaluación simple de señales de contaje de 24 V y TTL:



La figura siguiente muestra un ejemplo de evaluación simple de las señales de contaje RS422:

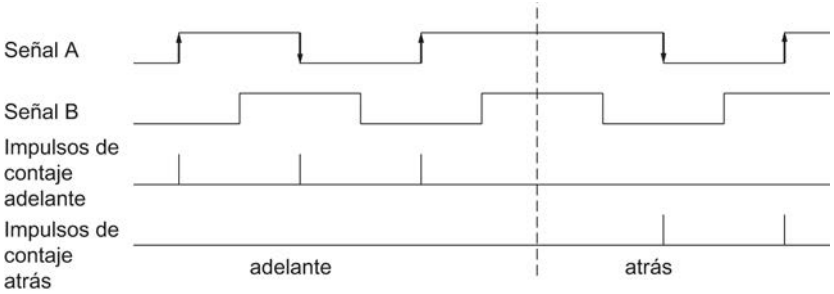


Evaluación doble

En la evaluación doble se evalúan los flancos ascendentes y descendentes de la señal A.

En función del sentido de los flancos de la señal A y del nivel de la señal B entretanto, se crean impulsos de contaje en sentido ascendente o descendente.

La figura siguiente muestra un ejemplo de evaluación doble de señales de contaje de 24 V y TTL:



La figura siguiente muestra un ejemplo de evaluación doble de las señales de contaje RS422:

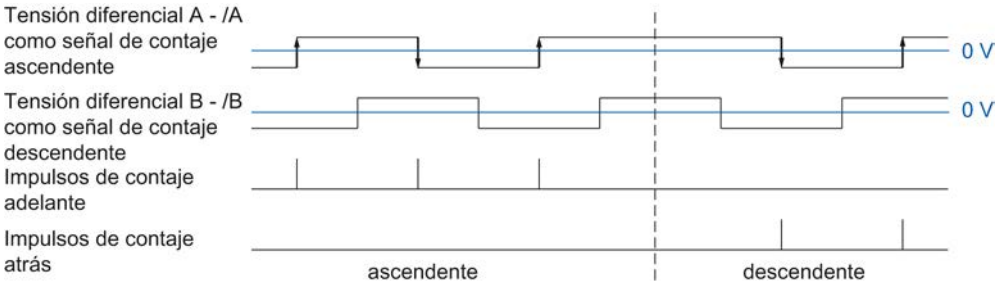


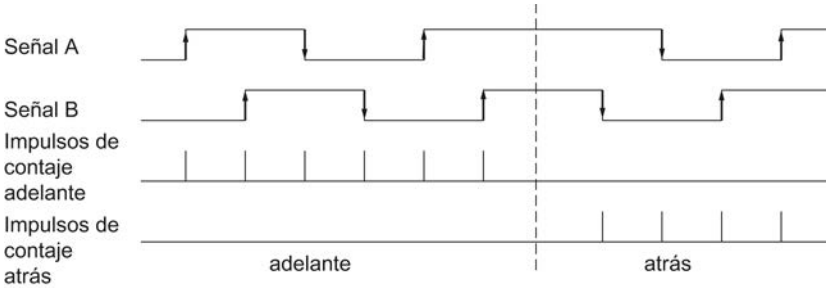
Figura 11-1 Doble

Evaluación cuádruple

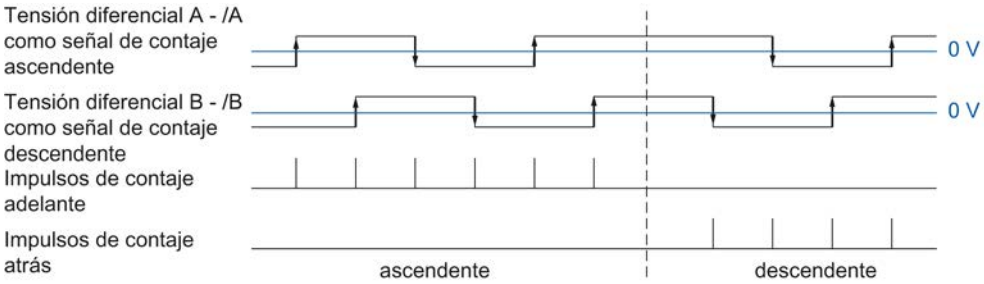
En la evaluación cuádruple se evalúan los flancos ascendentes y descendentes en las señales A y B.

En función del sentido de los flancos de una señal y del nivel entretanto de la otra, se crean los impulsos de contaje en sentido ascendente o descendente.

La figura siguiente muestra un ejemplo de evaluación cuádruple de señales de contaje de 24 V y TTL:



La figura siguiente muestra un ejemplo de evaluación cuádruple de las señales de contaje RS422:



Modo isócrono

El módulo tecnológico admite la función de sistema "Modo isócrono" en funcionamiento descentralizado. Esta función de sistema permite capturar valores de posición, valores de contaje y valores medidos en un determinado ciclo del sistema.

En el modo isócrono, el ciclo del programa de usuario, la transferencia de las señales de entrada y el procesamiento en el módulo tecnológico se sincronizan entre sí. Las señales de salida se conmutan inmediatamente si la condición de comparación correspondiente se cumple. Un cambio de estado en una entrada digital causa de inmediato la reacción prevista del módulo tecnológico y el cambio del bit de estado de la entrada digital en la interfaz de realimentación.

Procesamiento de los datos

Los datos transferidos al módulo tecnológico en el ciclo de bus actual a través de la interfaz de control se hacen efectivos al editarlos en el marco del ciclo interno del módulo tecnológico. En el instante T_i se capturan el valor de posición o de contaje, el valor medido y los bits de estado, y se proporcionan en la interfaz de realimentación para la recogida en el ciclo de bus actual.

En el modo isócrono siempre existe coherencia de datos para todos los bytes en la interfaz de realimentación.

11.1.1.2 Utilización de High_Speed_Counter

Objeto tecnológico High_Speed_Counter

STEP 7 (TIA Portal) ofrece soporte para la configuración de las funciones de contaje y medición de los módulos tecnológicos mediante la función "Objetos tecnológicos" del árbol del proyecto ("Proyecto > S7-1500 PLC > Objetos tecnológicos"):

- En STEP 7 (TIA Portal) configure el objeto tecnológico High_Speed_Counter con los ajustes para las funciones de contaje y medición.
- En el programa de usuario programe la instrucción correspondiente High_Speed_Counter. Esta instrucción aplica la alimentación de la interfaz de control y realimentación del módulo tecnológico.

El objeto tecnológico High_Speed_Counter corresponde al DB de instancia de la instrucción High_Speed_Counter. La configuración de las funciones de contaje y medición se guarda en el objeto tecnológico. El objeto tecnológico se encuentra en la carpeta "PLC > Objetos tecnológicos". A través del DB de instancia se materializa la instrucción.

El objeto tecnológico High_Speed_Counter puede utilizarse igualmente para los módulos tecnológicos de los sistemas S7-1500 y ET 200SP .

Modo de operación Contaje

Al utilizar el objeto tecnológico High_Speed_Counter se determina automáticamente el modo de operación Contaje. El valor medido está disponible simultáneamente.

Modo de operación Medición

Para el modo de operación Medición no hay ningún objeto tecnológico disponible. La parametrización del modo de operación Medición se realiza a través del ajuste de parámetros (HWCN) (Página 9439) del módulo tecnológico. El control del módulo tecnológico se realiza por medio del acceso directo a la interfaz de control y realimentación (Página 9460).

Lectura de posición

- La parametrización del encóder absoluto SSI (Página 9446) se realiza a través del ajuste de parámetros (HWCN) (Página 9439) del módulo tecnológico. Para la lectura de posición con un encóder absoluto SSI no hay ningún objeto tecnológico disponible.
- La parametrización de las señales de encóder para la lectura de posición para Motion Control (Página 9434) se realiza por medio de la configuración del dispositivo del módulo tecnológico. El resto de las configuraciones de esta aplicación se lleva a cabo a través del objeto tecnológico "TO_ExternalEncoder" de S7-1500 Motion Control .

Sinopsis de los pasos de configuración

Introducción

La siguiente sinopsis muestra el procedimiento básico de configuración de las funciones de contaje y medición del módulo tecnológico a través del objeto tecnológico High_Speed_Counter.

Requisitos

Para utilizar el objeto tecnológico High_Speed_Counter, en STEP 7 (TIA Portal) debe haberse creado un proyecto con una CPU S7-1500.

Procedimiento

Siga el orden que se recomienda a continuación:

Paso	Descripción
1	Configuración del módulo tecnológico y definición de la utilización con objeto tecnológico (Página 9430)
2	Agregar objeto tecnológico (Página 9409)
3	Trabajando con el diálogo de configuración (Página 9411)
4	Llamar la instrucción en el programa de usuario (Página 9426)
5	Carga en la CPU
6	Puesta en servicio del objeto tecnológico (Página 9427)
7	Diagnóstico del objeto tecnológico (Página 9428)

Agregar objeto tecnológico

Agregar objeto tecnológico en el árbol del proyecto

Al agregar un objeto tecnológico, se genera un DB de instancia de la instrucción para este objeto tecnológico. En él se guarda la configuración del objeto tecnológico.

Requisitos

Hay un proyecto creado con una CPU S7-1500.

Procedimiento

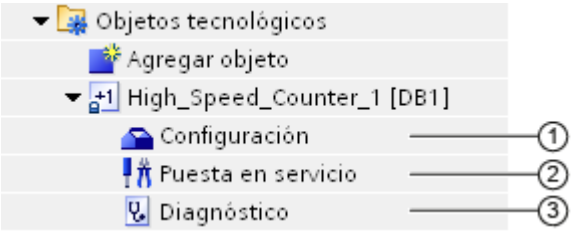
Para agregar un objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Objetos tecnológicos".
3. Haga doble clic en "Agregar objeto".
   Se abre el cuadro de diálogo "Agregar objeto".

4. Seleccione la tecnología "Contaje y medición".
5. Seleccione el objeto "High_Speed_Counter".
6. En el campo de entrada "Nombre" introduzca un nombre personalizado para el objeto tecnológico.
7. Seleccione la opción "Manual" si desea modificar el número de bloque de datos propuesto para el DB de instancia.
8. Haga clic en "Más información" para guardar información propia sobre el objeto tecnológico.
9. Confirme con "OK".

Resultado

El nuevo objeto tecnológico se genera y se guarda en la carpeta "Objetos tecnológicos" del árbol del proyecto.



	Objeto	Descripción
①	Configuración (Página 9411)	En el cuadro de diálogo de configuración: • Asignación del módulo tecnológico y del canal • Ajuste del parámetro del objeto tecnológico para las funciones de contaje y medición Si modifica la configuración del objeto tecnológico, a continuación deberá cargar en la CPU el objeto tecnológico **y** la configuración hardware.
②	Puesta en servicio (Página 9427)	Puesta en servicio y prueba funcional del objeto tecnológico: Simulación de parámetros de la instrucción High_Speed_Counter y observación de su repercusión
③	Diagnóstico (Página 9428)	Vigilancia de las funciones de contaje y medida

Configuración de High_Speed_Counter

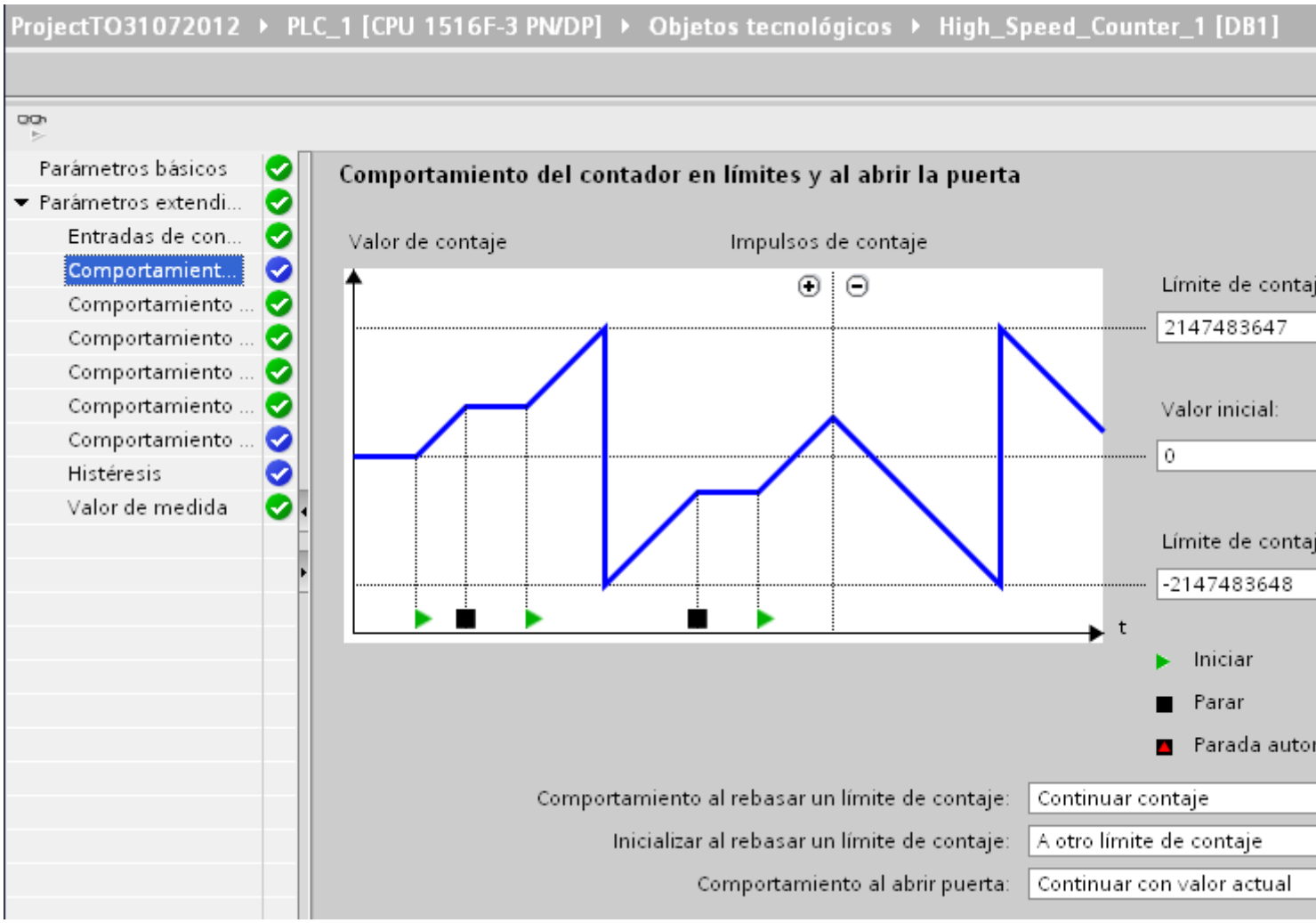
Trabajando con el diálogo de configuración

Las propiedades del objeto tecnológico se configuran en la ventana de configuración. Para abrir la ventana de configuración del objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Configuración".

La configuración se divide en las siguientes categorías:

- **Parámetros básicos**
  Los parámetros básicos contienen la selección del módulo tecnológico y el número del canal de contaje para el que se configura el objeto tecnológico.
- **Parámetros avanzados**
  Los parámetros avanzados contienen los parámetros para adaptar las funciones de contaje y medición y ajustar el comportamiento de las entradas y salidas digitales.



Símbolos de la ventana de configuración

Los símbolos que aparecen en la navegación local de la configuración muestran más detalles sobre el estado de la configuración:

✔	**La configuración contiene valores predeterminados y está completa.** La configuración contiene exclusivamente valores predeterminados. Con ellos es posible utilizar el objeto tecnológico sin más modificaciones.
✔	**La configuración contiene valores definidos por el usuario y está completa** Todos los campos de entrada de la configuración contienen valores válidos y al menos un valor predeterminado ha sido modificado.
✖	**La configuración es errónea o está incompleta** Al menos un campo de entrada o una lista desplegable contiene un valor no válido. El campo o la lista desplegable en cuestión se marca en rojo. Al hacer clic, el roll out con el aviso de error indica la causa del error.

Parámetros básicos

"Parámetros básicos" permite establecer la conexión entre el objeto tecnológico High_Speed_Counter y el módulo tecnológico.

Módulo

Seleccione el módulo tecnológico en el diálogo siguiente. Puede seleccionar todos los módulos tecnológicos (centralizados o descentralizados) configurados en la CPU S7-1500 para la aplicación de un objeto tecnológico de "contaje y medición".

Tras la selección del módulo tecnológico, puede abrir la configuración del dispositivo correspondiente al módulo tecnológico haciendo clic en el botón "Configuración del dispositivo".

El ajuste de parámetros del módulo tecnológico necesario para utilizar el objeto tecnológico se realiza a través de "Parámetros avanzados" del objeto tecnológico.

Canal

En el caso de un módulo tecnológico con varios canales de contaje, seleccione también el número del canal de contaje para el que es válido el objeto tecnológico High_Speed_Counter.

Calibración de los valores de parámetro

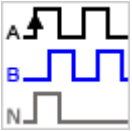
Si tras la asignación del canal de contaje al objeto tecnológico hay una incoherencia entre los valores de los parámetros de "Parámetro" (HWCN) y del objeto tecnológico, aparece el botón "Sobrescribir parámetros del módulo tecnológico". Al hacer clic en el botón se sobrescriben (previa consulta) en STEP 7 (TIA Portal) los valores de los parámetros de "Parámetro" (HWCN) con los valores de los parámetros del objeto tecnológico. Los valores de parámetros actuales del objeto tecnológico se muestran en "Parámetro" (HWCN) (read only).

Entradas de contaje

Especificar señales de entrada/tipo de encóder

Tipo de señal

Puede escoger entre los siguientes tipos de señal (Página 9399):

Símbolo	Tipo de señal	Significado
	Encóder incremental (A, B desfasado)	Está conectado un encóder incremental con las señales desfasadas A y B.
	Encóder incremental (A, B, N)	Un encóder incremental está conectado con las señales A y B desfasadas entre sí, así como con la señal nula N.
	Impulso (A) y sentido (B)	Está conectado un encóder de impulsos (señal A) con señal de sentido (señal B).
	Impulso (A)	Está conectado un encóder de impulsos (señal A) sin señal de sentido. Puede indicarse el sentido de contaje mediante la interfaz de control (Página 9460).
	Contador ascendente (A), contador descendente (B)	Están conectadas señales para el contaje en sentido ascendente (señal A) y sentido descendente (señal B).

Invertir sentido

Es posible invertir el sentido de contaje para adaptarlo al proceso.

La inversión del sentido solo puede parametrizarse y es efectiva si se ha seleccionado el tipo de señal "Encóder incremental (A, B desfasado)" o "Encóder incremental (A, B, N)".

Otros parámetros

Evaluación de la señal

Al parametrizar la Evaluación de la señal (Página 9403) se establece qué flancos de las señales se contarán.

Puede escoger entre las siguientes opciones:

Símbolo	Evaluación de la señal	Significado
	Simple (Página 9403) (predeterminado)	Se evalúan los flancos de la señal A durante un nivel bajo de la señal B.
	Doble (Página 9404)	Se evalúa cada flanco de la señal A.
	Cuádruple (Página 9404)	Se evalúa cada flanco de las señales A y B.

El parámetro puede parametrizarse con los siguientes tipos de señal:

- Encóder incremental (A, B desfasado)
- Encóder incremental (A, B, N)

Frecuencia de filtrado

Al parametrizar la frecuencia de filtrado se suprimen las interferencias en las entradas de contaje A, B y N.

La frecuencia de filtrado seleccionada se refiere a una proporción impulso/pausa comprendida entre 40:60 y 60:40. De esto resulta una determinada duración mínima de impulso y pausa. Los cambios de señal con una duración menor que la duración mínima de impulso/pausa se suprimen.

Puede escoger entre las siguientes frecuencias de filtro:

Tabla 11-2 Frecuencia de filtrado y respectiva duración mínima de impulso y pausa

Frecuencia de filtrado	Duración mínima de impulso y pausa
100 Hz	4,0 ms
200 Hz	2,0 ms
500 Hz	800 µs
1 kHz	400 µs
2 kHz	200 µs
5 kHz	80 µs
10 kHz	40 µs
20 kHz	20 µs
50 kHz	8,0 µs
100 kHz	4,0 µs
200 kHz (predeterminado en TM Count)	2,0 µs

Frecuencia de filtrado	Duración mínima de impulso y pausa
500 kHz*	0,8 µs
1 MHz* (predeterminado en TM PosInput)	0,4 µs

* Solo en TM PosInput

Tipo de sensor

Al parametrizar el tipo de sensor se define para TM Count el modo de conmutación de las entradas de contaje.

Puede escoger entre las siguientes opciones:

Tipo de sensor	Significado
Sensor tipo PNP (en fuente) (predeterminado)	El encóder o sensor conmuta las entradas A, B y N a 24VDC.
Sensor tipo NPN (en sumidero)	El encóder o sensor conmuta las entradas A, B y N a M.
Sensor tipo push-pull (tipo PNP y NPN)	El encóder o sensor conmuta las entradas A, B y N alternativamente a M y 24VDC.

Al utilizar encóders incrementales, la selección típica es "Salida push-pull". Cuando se utilizan sensores de 2 hilos, p. ej. barreras fotoeléctricas o detectores de proximidad, se debe seleccionar el cableado "Sensor tipo PNP (en fuente)" o "Sensor tipo NPN (en sumidero)" según corresponda.

Para saber si el encóder incremental es de tipo push/pull, consulte la hoja de datos del encóder.

Nota

Si utiliza un encóder de tipo push/pull, puede vigilar posibles roturas de hilo en las señales del encóder.

Estándar de interfaz

Con este parámetro puede definir para el TM PosInput si el encóder emite señales simétricas (RS422) o asimétricas (TTL).

Puede escoger entre las siguientes opciones:

Estándar de interfaz	Significado
RS422, simétrico	El encóder emite señales simétricas conforme al estándar RS422 (Página 9401).
TTL (5 V), asimétrico	El encóder emite señales asimétricas de 5 V conforme al estándar TTL (Página 9399).

Nota

El estándar RS422 brinda una mayor inmunidad a perturbaciones que el estándar TTL. Si el encóder incremental o el generador de impulsos dominan los estándares RS422 **y** TTL, se recomienda el estándar RS422.

Comportamiento con señal N

Este parámetro define cuál será la reacción en caso de señal N.

Puede escoger entre las siguientes opciones:

Opción	Significado
Sin reacción en caso de señal N (predeterminado)	La señal N no afecta al contador.
Sincronización en caso de señal N (Página 9375)	En caso de señal N, el contador se ajusta al valor de arranque. Si se selecciona para una entrada digital la función "Habilitar sincronización con señal N", la sincronización dependerá del nivel detectado en la entrada digital.
Captura (Capture) con señal N (Página 9368)	El valor de contaje se guarda en el valor de captura (Capture) en caso de señal N.

Nota

El comportamiento con señal N solo puede seleccionarse si se ha elegido el tipo de señal (Página 9412) "Encóder incremental (A, B, N)".

Nota

Si se selecciona " Sincronización en caso de señal N", puede elegirse para una entrada digital (Página 9419) la función "Habilitar sincronización con señal N".

Frecuencia

Con estos parámetros se define la frecuencia de los siguientes eventos:

- Sincronización en caso de señal N
- Sincronización como función de una entrada digital

Puede escoger entre las siguientes opciones:

Opción	Significado
Único (predeterminado)	El contador solo se ajusta con la primera señal N o el primer flanco parametrizado de la entrada digital.
Periódico	El contador se ajusta con cada señal N o cada flanco parametrizado de la entrada digital.

Comportamiento del contador

Límites de contaje y valor de arranque

Límite de contaje superior

Con la parametrización del límite de contaje superior se limita el rango de contaje. Puede especificarse un valor hasta 2147483647 (2^{31}-1). Debe introducirse un valor que esté por encima del límite de contaje inferior.

El ajuste predeterminado es "2147483647".

Límite de contaje inferior

Con la parametrización del límite de contaje inferior se limita el rango de contaje. Puede especificarse un valor hasta -2147483648 (-2^{31}). Debe introducirse un valor que esté por debajo del límite de contaje superior.

El ajuste predeterminado es "-2147483648".

Valor de arranque

Con la parametrización del valor de arranque se establece con qué valor de contaje se empieza y se continúa ante determinados eventos. Debe introducir un valor igual a los límites de contaje o comprendido entre ellos.

El ajuste predeterminado es "0".

Más información

Encontrará más información en Comportamiento en los límites de contaje (Página 9363) y Comportamiento del contador ante la apertura de puerta (Página 9367).

Comportamiento del contador en los límites y ante la apertura de puerta

Comportamiento al rebasar un límite de contaje

Puede parametrizarse el comportamiento siguiente al rebasar un límite de contaje (Página 9363):

Comportamiento	Significado
Parar contaje	Después de rebasar por exceso un límite de contaje, el contaje se cancela y la puerta interna se cierra. Para reiniciar el contaje debe cerrar y abrir de nuevo la puerta SW o la puerta HW, en su caso.
Continuar contaje (predeterminado)	En función del resto de la parametrización, se prosigue el contaje con el valor de arranque o con el límite de contaje opuesto en cada caso.

Inicializar al rebasar un límite de contaje

Puede ajustarse el contador a los siguientes valores al rebasar un límite de contaje:

Restablecimiento del valor	Significado
A valor de arranque	El valor de contaje se fija al valor de arranque.
A otro límite de contaje (predeterminado)	El valor de contaje se fija al límite de contaje opuesto.

Comportamiento ante apertura de puerta

Puede parametrizarse el siguiente comportamiento ante apertura de puerta (Página 9367):

Comportamiento	Significado
Poner a valor de arranque	Al abrirse la puerta, el valor de contaje se ajusta al valor de arranque.
Continuar con valor actual (predeterminado)	Al abrirse la puerta se sigue contando desde el último valor de contaje.

Comportamiento de una DI

Función de una DI

Ajuste de la función de la DI

Al parametrizar una entrada digital se define la función que dicha entrada digital desencadenará al conmutarse.

Puede escoger entre las siguientes opciones:

Función de una entrada digital	Significado
Apertura/cierre de puerta (controlados por nivel)	El nivel en la correspondiente entrada digital abre y cierra la puerta HW (Página 9365).
Apertura de puerta (controlada por flanco)	El flanco parametrizado en la correspondiente entrada digital abre la puerta HW (Página 9365).
Cierre de puerta (controlado por flanco)	El flanco parametrizado en la correspondiente entrada digital cierra la puerta HW (Página 9365).
Sincronización (Página 9371)	El flanco parametrizado en la correspondiente entrada digital pone el contador al valor de arranque.
Habilitar sincronización con señal N	El nivel activo en la correspondiente entrada digital habilita la sincronización del contador en caso de señal N (Página 9375).
Captura (Capture) (Página 9368)	El flanco parametrizado en la entrada digital correspondiente guarda el valor de contaje actual como valor de Capture.
Entrada digital sin función	La entrada digital en cuestión no tiene asignada función alguna. La CPU puede leer el estado lógico de la entrada digital a través de la interfaz de realimentación (Página 9463).

Nota

Cada función, excepto "Entrada digital sin función", solo puede utilizarse una vez y ya no puede volver a seleccionarse en el resto de las entradas digitales.

Tiempo de filtrado

Al parametrizar el tiempo de filtrado se suprimen las interferencias en las entradas digitales. Las señales con una duración de impulso inferior al tiempo de filtrado parametrizado se suprimen.

Puede escoger entre los siguientes tiempos de filtrado:

- Ninguno
- 0,05 ms
- 0,1 ms (predeterminado)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Nota

Si selecciona la opción "Ninguno" o "0,05 ms", debe utilizar cables apantallados para la conexión de las entradas digitales.

Nota

El tiempo de filtrado solo puede parametrizarse en "Comportamiento DI0" para todas las entradas digitales en conjunto.

Opciones de función

Algunas funciones necesitan otros parámetros con los que pueden detallar el comportamiento. Se pueden parametrizar para la función correspondiente.

Selección de nivel

Con este parámetro se determina el nivel con el que se activa la entrada digital.

Puede escoger entre las siguientes opciones:

Nivel	Significado
Activo con nivel alto (predeterminado)	La correspondiente entrada digital está activa cuando está ajustada.
Activo con nivel bajo	La correspondiente entrada digital está activa cuando no está ajustada.

El parámetro puede parametrizarse en las siguientes funciones de una entrada digital:

- Apertura/cierre de puerta (controlados por nivel)
- Habilitar sincronización con señal N

Selección de flancos

Este parámetro determina con qué flanco de una entrada digital se dispara la función parametrizada.

Independientemente de la función seleccionada, puede elegir entre las siguientes opciones:

- Con flanco ascendente (predeterminado)
- Con flanco descendente
- Con flanco ascendente y descendente

El parámetro puede parametrizarse en las siguientes funciones de una entrada digital:

- Apertura de puerta (controlada por flanco)
- Cierre de puerta (controlado por flanco)

- Sincronización
- Capture

"Con flanco ascendente y descendente" solo puede parametrizarse para la función "Capture".

Comportamiento del valor de contaje tras captura (Capture)

Puede parametrizarse la siguiente reacción del contador después de un evento Captura (Capture) (Página 9368):

	Significado
Continuar contaje (predeterminado)	El proceso de contaje continúa sin cambios después de que se guarde el valor de contaje actual como valor de Capture.
Poner al valor inicial y seguir contando	El proceso de contaje continúa con el valor de arranque tras guardarse el valor de contaje actual como valor de captura (Capture).

Consulte también

Función de una DI (Página 9417)

Función de una DQ

Activar salida

Al parametrizar una salida digital se define la condición (Página 9379) que provocará la conmutación de la salida digital.

Puede escoger entre las siguientes opciones:

Función de una salida digital	Significado
Entre valor de comparación y límite de contaje superior (predeterminado)	La salida digital correspondiente está activa si el valor de contaje se encuentra entre el valor de comparación y el límite de contaje superior.
Entre valor de referencia y límite de contaje inferior	La salida digital correspondiente está activa si el valor de contaje se encuentra entre el valor de comparación y el límite de contaje inferior.
Entre los valores de comparación 0 y 1	La salida digital DQ1 está activa si el valor de contaje se encuentra entre el valor de comparación 0 y el valor de comparación 1.
Con valor de comparación por la duración del impulso	La salida digital correspondiente está únicamente activa para el tiempo y el sentido de contaje parametrizados si el valor de contaje coincide con el valor de comparación.

Función de una salida digital	Significado
Tras comando Set de CPU hasta valor referencia	Cuando se produce un comando Set de la CPU, la salida digital correspondiente está activa hasta que el valor de contaje coincide con el de comparación.
Utilizada por el programa de usuario	La salida digital (Página 9378) en cuestión puede conmutarse desde la CPU a través de la interfaz de control (Página 9460).

Nota

Entonces es posible seleccionar la función "Entre valor de comparación 0 y 1" si se ha seleccionado para la salida digital DQ0 la función "Utilizada por el programa de usuario".

Nota

Las funciones "Con valor de comparación por la duración del impulso" y "Tras comando Set de CPU hasta valor referencia" conmutan la salida digital correspondiente solo cuando el valor de comparación se alcanza mediante un impulso de contaje. Al establecer el valor de contaje, p. ej. por medio de la sincronización, la salida digital no se conmuta.

Valor de comparación 0

Al parametrizar el valor de comparación (Página 9378) se determina con qué valor de contaje se conmutará la salida digital DQ0 de acuerdo con el evento de comparación seleccionado.

Debe introducirse un valor mayor o igual que el límite de contaje inferior y menor que el valor de comparación 1.

El ajuste predeterminado es "0".

Valor de comparación 1

Al parametrizar el valor de comparación (Página 9378) se determina con qué valor de contaje se conmutará la salida digital DQ1 de acuerdo con el evento de comparación seleccionado.

Debe introducirse un valor mayor que el valor de comparación 0 y menor o igual que el límite de contaje superior.

El ajuste predeterminado es "10".

Sentido de contaje

Con este parámetro se define para qué sentido de contaje es válida la función seleccionada.

Puede escoger entre las siguientes opciones:

Sentido de contaje	Significado
En ambos sentidos (predeterminado)	La salida digital correspondiente se compara y se conmuta independientemente del sentido de contaje.
Ascendente	La salida digital en cuestión solo se compara y se conmuta si el contador cuenta hacia delante.
Descendente	La salida digital correspondiente solo se compara y se conmuta si el contador cuenta hacia atrás.

El parámetro puede parametrizarse con las siguientes funciones:

- Con valor de comparación por la duración del impulso
- Tras comando Set de CPU hasta valor referencia

Duración del impulso

Al parametrizar la duración de impulso para la función "Con valor de comparación por la duración del impulso", se determina durante cuántos milisegundos estará activa la correspondiente salida digital.

Cuando introduce "0" y el valor de contaje coincide con el valor de comparación correspondiente, la salida digital está activa hasta el siguiente impulso de contaje.

Se admiten valores entre 0,0 y 6553,5.

El ajuste predeterminado es "500,0" y corresponde a una duración de impulso de 0,5 s.

Histéresis

Ajustar rango de histéresis

Con la parametrización de la histéresis (Página 9395) se define un rango para los valores de comparación. En el rango de histéresis las salidas digitales no pueden conmutarse de nuevo hasta que el valor de contaje haya salido una vez de este rango.

El rango de histéresis finaliza, independientemente del valor de histéresis, en el límite inferior o superior de contaje. Si se introduce "0" se desactiva la histéresis. Se admiten valores entre 0 y 255. El ajuste predeterminado es "0".

La histéresis puede parametrizarse únicamente en el modo de operación Contaje.

Especificar valor de medida

Magnitud

Con este parámetro se define qué magnitud (Página 9391) debe suministrar el módulo tecnológico.

Puede escoger entre las siguientes opciones:

Magnitud	Significado
Frecuencia (predeterminado)	La magnitud es una frecuencia. La unidad es Hz.
Duración del período	La magnitud es una duración del período. La unidad es s.
Velocidad	La magnitud es una velocidad. La base de tiempo para la medición de la velocidad y los Incrementos por unidad se parametrizan por separado.

Tiempo de actualización

Con la parametrización del tiempo de actualización (Página 9390) en milisegundos se define el intervalo de tiempo que transcurre entre dos actualizaciones del valor medido.

El tiempo de actualización y el tipo de señal (Página 9412) influyen en la precisión de la medición. Con tiempos de actualización de al menos 100 ms la influencia del Tipo de señal es insignificante.

Con tiempos de actualización inferiores a 100 ms se alcanza la máxima precisión de medición con los siguientes tipos de señales:

- Encóder incremental (A, B desfasado) con Evaluación de señal "Simple"
- Encóder incremental (A, B, N) con Evaluación de señal "Simple"
- Impulso (A) y sentido (B)
- Impulso (A)

En otros tipos de señales la precisión de medición depende del encóder utilizado y de la línea.

Cuando se introduce "0", el valor medido se actualiza en cada ciclo. Puede introducir un máximo de tres decimales. Se admiten valores entre 0,000 y 25000,000. El ajuste predeterminado es "10,000".

Base de tiempo para medición de velocidad

Con este parámetro se define la base de tiempo con la que debe ponerse a disposición la velocidad.

Puede escoger entre las siguientes opciones:

- 1 ms
- 10 ms
- 100 ms
- 1 s
- 60 s/1 min
- El ajuste predeterminado es "60 s/1 min".

Incrementos por unidad

Con este parámetro se define el número de impulsos de contaje por unidad relevante.

El número de impulsos de contaje depende de la Evaluación de señal parametrizada. Se admiten valores entre 1 y 65535.

Ejemplo:

Su encóder emite 4000 incrementos por metro. La velocidad debe medirse en metros por minuto. Como Evaluación de señal se ha parametrizado "doble".

En este caso, parametrice lo siguiente:

- Incrementos por unidad: 8000
- Base de tiempo para medición de velocidad: 60 s/1 min

Programación de High_Speed_Counter

Instrucción High_Speed_Counter

High_Speed_Counter

La instrucción High_Speed_Counter corresponde al objeto tecnológico High_Speed_Counter y acepta la alimentación de la interfaz de control y realimentación del módulo tecnológico.

La instrucción High_Speed_Counter constituye así la interfaz de software entre la CPU y el módulo tecnológico. Se ejecuta de modo cíclico desde el programa de usuario para compensar los datos de entrada y salida.

La instrucción High_Speed_Counter puede utilizarse de igual modo para los módulos tecnológicos de los sistemas S7-1500 y ET 200SP.

Más información

Descripción High_Speed_Counter (Página 5123)

Parámetro de entrada High_Speed_Counter (Página 5127)

Parámetro de salida High_Speed_Counter (Página 5128)

Parámetro ErrorID (Página 5129)

Variables estáticas High_Speed_Counter (Página 5130)

Llamar la instrucción en el programa de usuario

La instrucción High_Speed_Counter puede llamarse una vez por contador en el ciclo o alternativamente en un programa controlado por tiempo. No se permite realizar la llamada en un programa de interrupción controlado por eventos.

Procedimiento

Para llamar a la instrucción en el programa de usuario, proceda del siguiente modo:

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Bloques de programa".

3. Haga clic en el OB para la ejecución cíclica del programa.
   El bloque se abre en el área de trabajo.
4. En la ventana "Instrucciones" abra el grupo "Tecnología" y la carpeta "Contaje y medición".
   La carpeta contiene las instrucciones.
5. Seleccione una instrucción, arrástrela al OB y suéltela.
   Se abre el cuadro de diálogo "Opciones de llamada".
6. En la lista "Nombre" seleccione un objeto tecnológico o introduzca el nombre del nuevo objeto tecnológico.

Resultado

Si el objeto tecnológico no existe todavía, se agregará. La instrucción se agregará en el OB. El objeto tecnológico está asignado a esta llamada de la instrucción.

Puesta en servicio de High_Speed_Counter

Puesta en servicio del objeto tecnológico

El editor de puesta en servicio le ayudará a poner en marcha el objeto tecnológico y probar su funcionalidad. En el modo online de la CPU/IM puede modificar determinados parámetros de la instrucción High_Speed_Counter y observar el efecto de los cambios.

Requisitos

- Hay una conexión online entre STEP 7 (TIA Portal) y la CPU.
- La CPU se encuentra en el estado operativo RUN.
- La instrucción High_Speed_Counter correspondiente se ejecuta de modo cíclico desde el programa de usuario.
- Los parámetros del objeto tecnológico no se sobrescriben del programa de usuario.

Procedimiento

Para abrir el editor de puesta en servicio de un objeto tecnológico, haga lo siguiente:

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico High_Speed_Counter en el árbol del proyecto.
3. Haga doble clic en el objeto "Puesta en servicio".
   Se muestran las funciones de puesta en servicio del objeto tecnológico High_Speed_Counter.
4. Haga clic en el botón "Iniciar indicador".
   Se cargan y se muestran los parámetros (valores online) del objeto tecnológico High_Speed_Counter.

Modo online

En el modo online se pueden modificar los siguientes parámetros y, de este modo, probar el funcionamiento del objeto tecnológico:

- Nuevo valor de contaje (NewCountValue)
- Nuevo límite de contaje superior (NewUpperLimit)
- Nuevo límite de contaje inferior (NewLowerLimit)
- Nuevo valor de comparación 0 (NewReferenceValue0)
- Nuevo valor de comparación 1 (NewReferenceValue1)
- Nuevo valor de arranque (NewStartValue)
- Iniciar y parar contador (SwGate)
- Habilitar Capture (CaptureEnable)
- Habilitar sincronización (SyncEnable)
- Confirmación de los estados de error notificados (ErrorAck)
- Restablecimiento de flags de estado (EventAck)

Los valores modificados se activan al salir del campo de entrada o mediante ENTER o el establecimiento adicional de la variable Set correspondiente como nuevo parámetro del objeto tecnológico.

Diagnóstico de High_Speed_Counter

Observación de valores de contaje, valores medidos, DI y DQ

Con las funciones de diagnóstico se vigilan las funciones de contaje y medida.

Requisitos

- Hay una conexión online entre STEP 7 (TIA Portal) y la CPU.
- La CPU se encuentra en el estado operativo RUN.

Procedimiento

Para abrir el editor de visualización para las funciones de diagnóstico, haga lo siguiente:

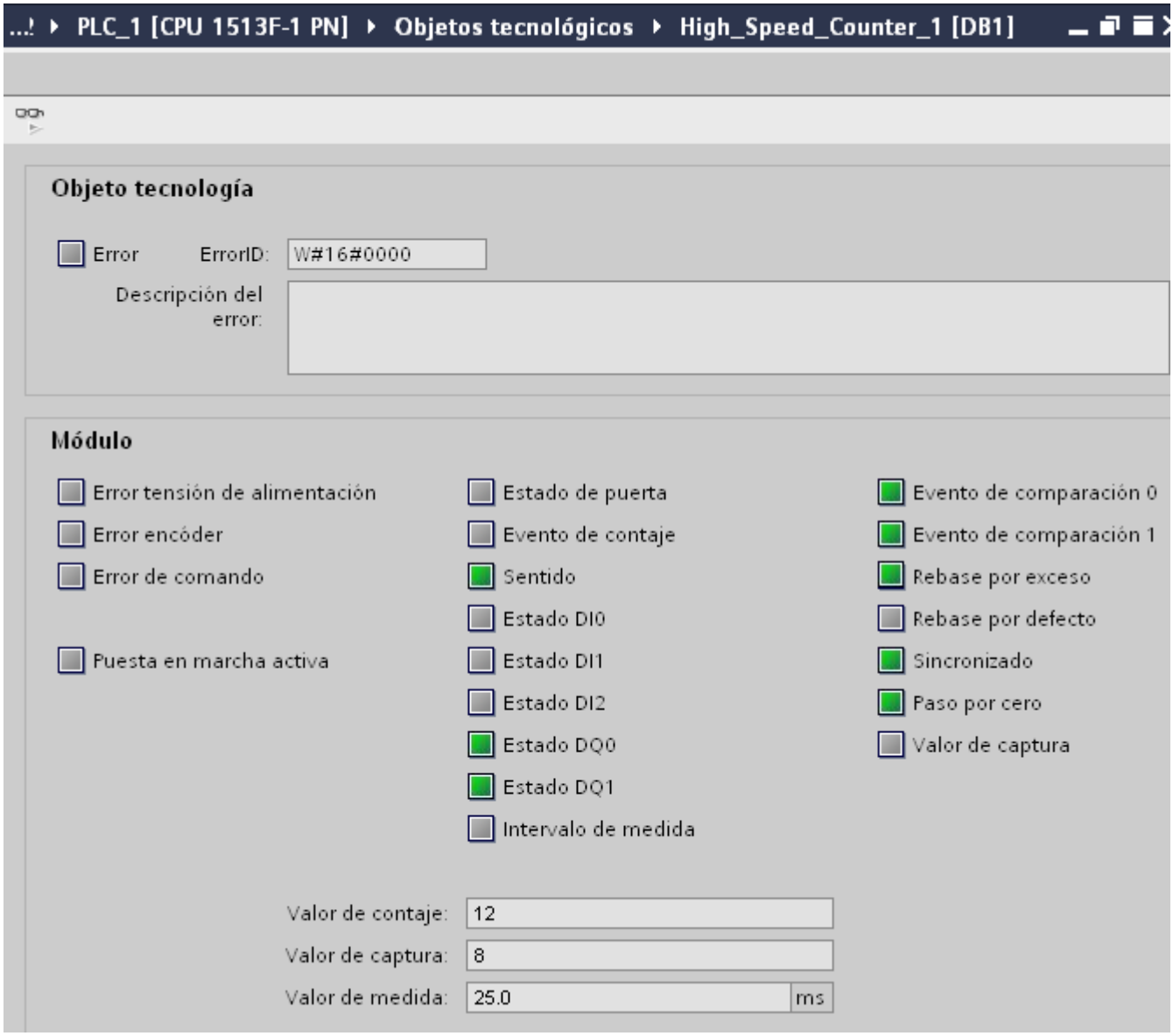
1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico High_Speed_Counter en el árbol del proyecto.
3. Haga doble clic en el objeto "Diagnóstico".

Indicador

Se leen y se muestran los siguientes valores del objeto tecnológico:

- Indicación de eventos/información de diagnóstico
- Estados lógicos de las entradas digitales y las salidas digitales
- Valor de contaje
- Valor Capture
- Valor medido

Encontrará más información acerca de las indicaciones de estado en la ayuda contextual de cada evento en STEP 7 (TIA Portal). Si la CPU está en STOP, no se actualiza la indicación de estado.



11.1.1.3 Utilización del módulo tecnológico

Configuración del módulo

Agregar un módulo tecnológico a la configuración de hardware

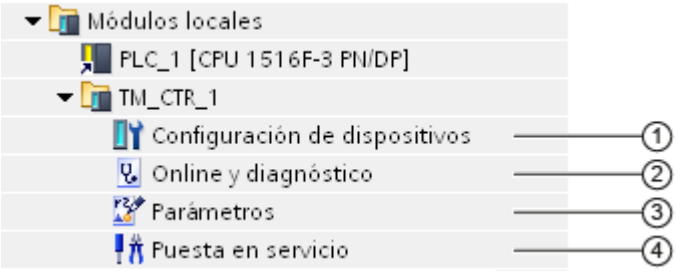
Procedimiento

Para poder configurar un módulo tecnológico, debe haber creado previamente un proyecto donde pueda guardar la configuración.

1. Abra la configuración de equipo de la CPU o el IM.
2. Seleccione un rack.
3. Seleccione el módulo tecnológico en el catálogo de módulos:
   "TM > Contaje o Lectura de recorrido > Módulo tecnológico > Referencia".
4. Coloque el módulo tecnológico arrastrándolo al slot deseado del rack y soltándolo.

Resultado

En el árbol del proyecto, el nuevo módulo tecnológico aparecerá en "Módulos locales" o "Periferia descentralizada" con los siguientes objetos: Haga doble clic para pasar al editor deseado.



	Objeto	Descripción
①	Configuración del equipo (Página 9431)	En la ventana de inspección (por canal): • Ajuste de la reacción a STOP de la CPU • Habilitación de la alarma de diagnóstico • Ajuste del modo de funcionamiento (Página 9433) • Habilitar alarma de proceso (Página 9436) • Ajuste de las direcciones del módulo
②	Online y diagnóstico (Página 9457)	• Diagnóstico de hardware • Obtención de información a través del módulo tecnológico • Actualización del firmware

	Objeto	Descripción
③	Parámetro (Página 9438)	Visualización y, dado el caso, ajuste de los parámetros para las funciones de contaje y medición en HWCN, si la parametrización y el control no se llevan a cabo por medio del objeto tecnológico High_Speed_Counter .
④	Puesta en servicio (Página 9457)	Puesta en servicio y prueba funcional del módulo tecnológico: Simulación de los valores de contaje y DI en el modo online de la CPU o el IM y observación del efecto de los cambios.

Parámetros básicos

Reacción a STOP de la CPU

Reacción a STOP de la CPU

La reacción del módulo tecnológico en caso de STOP de la CPU se ajusta para cada canal en los parámetros básicos de la configuración del equipo.

Tabla 11-3 Reacción del módulo tecnológico en caso de STOP de la CPU en función de la parametrización (por canal)

Parámetros básicos	Reacción a STOP de la CPU
Continuar	El módulo tecnológico sigue operando con plena funcionalidad. Se procesan los impulsos de contaje entrantes o se lee la posición real. Las salidas digitales continúan conmutándose con arreglo a la parametrización.
Aplicar valor sustitutivo	Hasta la siguiente transición de STOP a RUN de la CPU, el módulo tecnológico emite en las salidas digitales los valores de sustitución parametrizados. Después de una transición de STOP a RUN, el módulo tecnológico pasa a su estado de arranque: el valor de contaje se ajusta al valor de arranque (con encóders incrementales o generadores de impulsos) y las salidas digitales se conmutan según la parametrización.
Mantener último valor	Hasta la siguiente transición de STOP a RUN de la CPU, el módulo tecnológico emite en las salidas digitales los valores que eran válidos en el momento de la transición a STOP. Después de una transición de STOP a RUN, el módulo tecnológico pasa a su estado de arranque: el valor de contaje se ajusta al valor de arranque (con encóders incrementales o generadores de impulsos) y las salidas digitales se conmutan según la parametrización.

Valor sustitutivo de DQ0

Con este parámetro se define el valor que debe emitir el módulo tecnológico en caso de STOP de la CPU en la salida digital DQ0 para el comportamiento "Aplicar valor sustitutivo".

Valor sustitutivo de DQ1

Con este parámetro se define el valor que debe emitir el módulo tecnológico en caso de STOP de la CPU en la salida digital DQ1 para el comportamiento "Aplicar valor sustitutivo".

Alarmas de diagnóstico

El módulo tecnológico puede disparar alarmas de diagnóstico adicionales si se activa la habilitación de las alarmas de diagnóstico en los parámetros básicos. Las alarmas de diagnóstico se procesan en un OB de alarma.

Habilitar alarma de diagnóstico en caso de rotura de hilo

Este parámetro permite especificar para los siguientes encóders si se dispara una alarma de diagnóstico en caso de rotura de hilo de las señales utilizadas:

- Encóder tipo push/pull de 24 V (Página 9399)
- Encóder RS422 (Página 9401) (se vigila también si se producen cortocircuitos o tensiones erróneas)
- Encóder absoluto SSI (Página 9402) (se vigila también si se producen cortocircuitos o tensiones erróneas)

Nota

Si utiliza un encóder con otro tipo de sensor o estándar de interfaz, no puede detectarse una rotura de hilo.

Habilitación de otras alarmas de diagnóstico

Este parámetro define si se disparan alarmas de diagnóstico con otros errores.

Infórmese con el manual de producto del módulo tecnológico sobre los errores que pueden disparar una alarma de diagnóstico durante el funcionamiento.

Modo de funcionamiento

Selección del modo de funcionamiento para el canal

Con este ajuste se define cómo debe llevarse a cabo la parametrización y el control de las funciones de contaje y medida del canal.

Modo de funcionamiento	Descripción
Configurar canal mediante objeto tecnológico	La parametrización del canal se produce a través del objeto tecnológico High_Speed_Counter. El acceso a la interfaz de control y realimentación del módulo tecnológico aplica la instrucción High_Speed_Counter correspondiente en el programa de usuario. El emparejamiento de módulo tecnológico/canal y objeto tecnológico se define al comienzo de la parametrización del objeto tecnológico: ver Parámetros básicos Objeto tecnológico (Página 9411) Al utilizar el objeto tecnológico High_Speed_Counter se determina automáticamente el modo de operación Contaje.
Lectura de posición para Motion Control	El módulo tecnológico es utilizado para la lectura de posición por un controlador Motion Control de nivel superior. En este caso, se encuentran disponibles todos los canales del módulo tecnológico para Motion Control. La parametrización se efectúa a través de la configuración de equipo del módulo tecnológico. La parametrización de las señales del encóder se realiza por medio de los parámetros del módulo (Página 9434).
Funcionamiento manual	La parametrización del canal se lleva a cabo por medio del ajuste de parámetros (HWCN) (Página 9439) del módulo tecnológico. A través del programa de usuario tiene acceso directo a la interfaz de control y realimentación del módulo tecnológico.

Selección del modo de operación para el canal

Con el parámetro "Funcionamiento manual" se define para qué función se utiliza el canal del módulo tecnológico. Las posibilidades de ajuste de "Parámetro" (HWCN) se adaptan en función de dicha elección.

Modo de operación	Descripción
Contaje (Página 9360)/Lectura de posición (Página 9362)	La tarea principal del módulo tecnológico es el contaje o la lectura de posición. Las funciones de comparación y alarmas de proceso trabajan con el valor de contaje o de posición. El valor medido está disponible simultáneamente.
Medición (Página 9389)	La tarea principal del módulo tecnológico es la medición. Las funciones de comparación (Página 9445) y las alarmas de proceso para eventos de comparación trabajan con el valor de medida. El valor de contaje está disponible simultáneamente.

Parámetros del módulo (lectura de posición para Motion Control)

En el modo de funcionamiento "Lectura de posición para Motion Control", los parámetros para las señales de encóder del canal se ajustan en "Parámetros del módulo". Los parámetros dependen del encóder utilizado.

Parámetros del módulo para encóders incrementales y de impulsos

Cuando se utiliza un encóder incremental o uno de impulsos, se ajustan los siguientes parámetros para las señales de encóder del canal.

- Tipo de señal
- Invertir sentido
- Evaluación de la señal
- Frecuencia de filtrado
- Tipo de sensor (en TM Count)
- Estándar de interfaz (en TM PosInput)
- Selección de señal para marca de referencia 0
- Incrementos por unidad y
- Velocidad de referencia

Encontrará explicaciones sobre los primeros seis parámetros en el apartado Entradas de contaje (Página 9412).

Parámetros del módulo para encóder absoluto SSI

Si utiliza un TM PosInput con un encóder absoluto SSI, ajuste los siguientes parámetros para las señales de encóder del canal.

- Tipo de señal
- Invertir sentido
- Longitud de telegrama
- Tipo de código
- Velocidad de transferencia
- Tiempo monoestable
- Paridad
- Número de bit del LSB del valor de posición
- Número de bit del MSB del valor de posición
- Incrementos por unidad y
- Velocidad de referencia

Encontrará explicaciones sobre los primeros nueve parámetros en el apartado Parametrización de la lectura de posición con encóder absoluto SSI (Página 9446).

Selección señal marca de referencia 0

Con este parámetro se define con qué señal de referencia externa se guarda una nueva marca de referencia para la posición del encóder al utilizar un S7-1500 Motion Control .

Puede escoger entre las siguientes opciones:

Opción	Significado
DI0	Si se produce un flanco ascendente en la entrada digital DI0, el valor de contaje actual se guarda como nueva marca de referencia para la posición del encóder.
Señal N del encóder incremental	Si se produce un flanco ascendente en la señal N del encóder incremental, el valor de contaje actual se guarda como nueva marca de referencia para la posición del encóder.

Incrementos por unidad

En caso de utilizar un encóder incremental o de impulsos, con este parámetro se define el número de impulsos de contaje por vuelta del encóder. En caso de utilizar un encóder absoluto SSI, con este parámetro se define el número de incrementos que proporciona el encóder con cada vuelta.

El número de impulsos de contaje depende de la Evaluación de señal parametrizada. Se admiten valores entre 1 y 65535.

Ejemplo de encóder incremental o de impulsos:

Su encóder incremental o de impulsos proporciona 2048 incrementos por vuelta. En este caso, dependiendo de la Evaluación de la señal, debe parametrizar lo siguiente:

Evaluación de la señal	Incrementos por unidad
Simple	2048
Doble	4096
Cuádruple	8192

Velocidad de referencia

Al utilizar un S7-1500 Motion Control, el encóder transfiere el valor real de la velocidad como valor porcentual según la velocidad de referencia. Con este parámetro se define la velocidad en rpm que debe corresponder al valor 100 % . La velocidad de referencia debe ser idéntica a la del ajuste del control.

Se admiten valores entre 6,00 y 210000,00. El ajuste predeterminado es "3000,00".

Otras configuraciones

El resto de las configuraciones se lleva a cabo a través del objeto tecnológico "TO_ExternalEncoder" de S7-1500 Motion Control . Infórmese con el manual de funciones

S7-1500 Motion Control (http://support.automation.siemens.com/WW/view/es/59381279) del resto de opciones de configuración y puesta en servicio de la lectura de posición.

Alarmas de proceso

En los parámetros básicos del módulo tecnológico se puede definir para cada canal qué eventos generarán una alarma de proceso durante el funcionamiento.

En un sistema S7-1500, se adjudica un nombre de evento adecuado para cada alarma de proceso habilitada y se asigna un OB de alarma de proceso correspondiente a cada alarma de proceso. Cuando se dispara una alarma de proceso se inicia el OB correspondiente para evaluar los datos de las alarmas de proceso.

Una alarma de proceso se dispara cuando se cumple la condición para la modificación del correspondiente bit de estado o de evento en la interfaz de realimentación.

Alarma de proceso perdida

Si se produce un evento que debe provocar una alarma de proceso pero todavía no se ha procesado un evento igual ocurrido anteriormente, no se dispara una nueva alarma de proceso. La alarma de proceso se pierde. Dependiendo de la parametrización, esto puede conducir a la alarma de diagnóstico "Alarma de proceso perdida".

Alarmas de proceso activables

Alarma de proceso	Disponible en el modo de operación Contaje si se utiliza		Disponible en el modo de operación Medición si se utiliza		Disponible en el modo de funcionamiento o Lectura de posición para Motion Control	Descripción	Número de EventType
	Encóder incremental o generador de impulsos	Encóder absoluto SSI	Encóder incremental o generador de impulsos	Encóder absoluto SSI			
Nuevo valor de Capture existente	Sí	Sí	No	No	No	Alarma de proceso al guardar el valor de contaje o de posición actual como valor Capture.	8
Sincronización del contador por señal externa	Sí	No	Sí	No	No	Alarma de proceso al sincronizar el contador con la señal N o el flanco DI	9
Apertura de puerta	Sí	No	Sí	No	No	Alarma de proceso al abrir la puerta interna	1
Cierre de puerta	Sí	No	Sí	No	No	Alarma de proceso al cerrar la puerta interna	2
Rebase por exceso (límite superior excedido)	Referido al valor de contaje	No	Referido al valor de contaje	No	No	Alarma de proceso cuando el valor de contaje rebasa el límite superior de contaje	3

Alarma de proceso	Disponible en el modo de operación Contaje si se utiliza		Disponible en el modo de operación Medición si se utiliza		Disponible en el modo de funcionamiento o Lectura de posición para Motion Control	Descripción	Número de EventType
	Encóder incremental o generador de impulsos	Encóder absoluto SSI	Encóder incremental o generador de impulsos	Encóder absoluto SSI			
Rebase por defecto (límite inferior excedido)	Referido al valor de contaje	No	Referido al valor de contaje	No	No	Alarma de proceso cuando el valor de contaje rebasa el límite inferior de contaje	4
Inversión de sentido*	Referido al valor de contaje	Referido al valor de posición	Referido al valor de contaje	Referido al valor de posición	No	Alarma de proceso al invertir el sentido del valor de contaje o de posición	10
Paso por cero	Referido al valor de contaje	Referido al valor de posición	Referido al valor de contaje	Referido al valor de posición	No	Alarma de proceso en el paso por cero del valor de contaje o de posición	7
Evento de comparación producido para DQ0	Referido al valor de contaje	Referido al valor de posición	Referido al valor medido	Referido al valor medido	No	Alarma de proceso cuando se produce un evento de comparación para DQ0 debido a la condición de comparación seleccionada. Sin alarma de proceso si en un encóder incremental o generador de impulsos la modificación del valor de contaje no ha sido provocada por un impulso de contaje.	5
Evento de comparación producido para DQ1	Referido al valor de contaje	Referido al valor de posición	Referido al valor medido	Referido al valor medido	No	Alarma de proceso cuando se produce un evento de comparación para DQ1 debido a la condición de comparación seleccionada. Sin alarma de proceso si en un encóder incremental o generador de impulsos la modificación del valor de contaje no ha sido provocada por un impulso de contaje.	6

* El bit de realimentación STS_DIR está preajustado con "0". Si el valor de contaje o de posición se modifica por primera vez directamente después de activar el módulo tecnológico en sentido descendente, no se dispara ninguna alarma de proceso.

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de proceso.

Parametrizar módulo

Posibilidades de la parametrización

Modo de operación Contaje: posibilidades de la parametrización

En el caso de las funciones de contaje, tiene dos alternativas para parametrizar y controlar las funciones del módulo tecnológico:

- Configuración de un objeto tecnológico High_Speed_Counter y control por medio de la instrucción High_Speed_Counter correspondiente

  Nota

  Configuración mediante objeto tecnológico

  Se recomienda la configuración cómoda y que admite gráficos por medio del objeto tecnológico High_Speed_Counter. Encontrará una descripción a fondo de esta configuración a partir del apartado Auto-Hotspot.

  Al iniciar la parametrización del objeto tecnológico (Página 9411), asigne al objeto tecnológico el módulo tecnológico configurado y el canal de contaje.

- Ajuste de parámetros mediante HWCN (Página 9439) y control por medio de la interfaz de control y realimentación del módulo tecnológico

  Nota

  Ajuste de parámetros mediante HWCN

  Encontrará más indicaciones para el ajuste de parámetros mediante HWCN en la ayuda contextual de los parámetros en STEP 7 (TIA Portal). Encontrará una descripción de la interfaz de control y realimentación en los siguientes apartados:

  Asignación de la interfaz de control (Página 9460)

  Asignación de la interfaz de realimentación (Página 9463)

Modo de operación Medición

Para el modo de operación Medición no hay ningún objeto tecnológico disponible. La parametrización (Página 9439) del modo de operación Medición se realiza a través del ajuste de parámetros (HWCN) (Página 9439) del módulo tecnológico.

El control del módulo tecnológico se realiza por medio del acceso directo a la interfaz de control y realimentación.

Lectura de posición con encóder absoluto SSI

Para la lectura de posición con el encóder absoluto SSI no hay ningún objeto tecnológico disponible.

- En el modo de funcionamiento "Funcionamiento manual" realice la parametrización (Página 9446) del encóder absoluto SSI, las entradas digitales y las salidas digitales por medio del ajuste de parámetros (HWCN) (Página 9439) del módulo tecnológico.
- En el modo de funcionamiento "Lectura de posición para Motion Control" realice la parametrización del encóder absoluto SSI mediante los parámetros del módulo (Página 9432) al configurar el dispositivo del módulo tecnológico.

Las realimentaciones del módulo tecnológico se realizan a través de la interfaz de realimentación.

Apertura del ajuste de parámetros (HWCN)

Apertura mediante el árbol del proyecto

Proceda de la siguiente manera:

1. Abra la carpeta "Módulos locales" o "Periferia descentralizada" en el árbol del proyecto.
2. Abra el módulo tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Parámetro".

Apertura mediante la vista de dispositivos

Proceda de la siguiente manera:

1. Abra la configuración del equipo de la CPU o el IM.
2. Seleccione la vista de dispositivos.
3. Haga clic con el botón derecho del ratón en el módulo tecnológico y seleccione "Parámetro".

Modo de operación Medición

Sinopsis

En el modo de operación Medición, parametrice lo siguiente:

- Entradas de contaje
  - Especificar señales de entrada/tipo de encóder (Página 9440)
  - Otros parámetros (Página 9440)
- Especificar valor de medida (Página 9422)
- Comportamiento de una DI
  - Función de una DI (Página 9443)
  - Opciones de función (Página 9444)

- Comportamiento de una DQ
  - Función de una DQ (Página 9445)
- Comportamiento del contador (Página 9416)

Entradas de contaje: Especificar señales de entrada/tipo de encóder (modo de operación Medición)

Tipo de señal

Puede escoger entre los siguientes tipos de señal:

Tipo de señal	Significado
Encóder incremental (A, B desfasado)	Está conectado un encóder incremental con las señales desfasadas A y B.
Encóder incremental (A, B, N)	Un encóder incremental está conectado con las señales A y B desfasadas entre sí, así como con la señal nula N.
Impulso (A) y sentido (B)	Está conectado un encóder de impulsos (señal A) con señal de sentido (señal B).
Impulso (A)	Está conectado un encóder de impulsos (señal A) sin señal de sentido. Puede indicarse el sentido de contaje mediante la interfaz de control (Página 9460).
Contador ascendente (A), contador descendente (B)	Están conectadas señales para el contaje en sentido ascendente (señal A) y sentido descendente (señal B).
Encóder absoluto (SSI)	Está conectado un encóder absoluto SSI (solo al utilizar TM PosInput): ver Parametrización de la lectura de posición con encóder absoluto SSI (Página 9446)

Invertir sentido

Es posible invertir el sentido de contaje o los valores de posición para adaptarlos al proceso.

La inversión puede parametrizarse y hacerse efectiva en los siguientes tipos de señal:

- Encóder incremental (A, B desfasado)
- Encóder incremental (A, B, N)
- Encóder absoluto (SSI)

Entradas de contaje: Otros parámetros (modo de operación Medición)

Evaluación de la señal

Al parametrizar la Evaluación de la señal (Página 9403) se establece qué flancos de las señales se contarán.

Puede escoger entre las siguientes opciones:

Evaluación de la señal	Significado
Simple (predeterminado)	Se evalúan los flancos de la señal A durante un nivel bajo de la señal B.
Doble	Se evalúa cada flanco de la señal A.
Cuádruple	Se evalúa cada flanco de las señales A y B.

El parámetro puede parametrizarse con los siguientes tipos de señal:

- Encóder incremental (A, B desfasado)
- Encóder incremental (A, B, N)

Frecuencia de filtrado

Al parametrizar la frecuencia de filtrado se suprimen las interferencias en las entradas de contaje A, B y N.

La frecuencia de filtrado seleccionada se refiere a una proporción impulso/pausa comprendida entre 40:60 y 60:40. De esto resulta una determinada duración mínima de impulso y pausa. Los cambios de señal con una duración menor que la duración mínima de impulso/pausa se suprimen.

Puede escoger entre las siguientes frecuencias de filtro:

Tabla 11-4 Frecuencia de filtrado y respectiva duración mínima de impulso y pausa

Frecuencia de filtrado	Duración mínima de impulso y pausa
100 Hz	4,0 ms
200 Hz	2,0 ms
500 Hz	800 µs
1 kHz	400 µs
2 kHz	200 µs
5 kHz	80 µs
10 kHz	40 µs
20 kHz	20 µs
50 kHz	8,0 µs
100 kHz	4,0 µs
200 kHz (predeterminado en TM Count)	2,0 µs
500 kHz*	0,8 µs
1 MHz* (predeterminado en TM PosInput)	0,4 µs

* Solo en TM PosInput

Tipo de sensor

Al parametrizar el tipo de sensor se define para TM Count el modo de conmutación de las entradas de contaje.

Puede escoger entre las siguientes opciones:

Tipo de sensor	Significado
Sensor tipo PNP (en fuente) (predeterminado)	El encóder o sensor activa las entradas A, B y N a 24VDC.
Sensor tipo NPN (en sumidero)	El encóder o sensor activa las entradas A, B y N a M.
Sensor tipo push-pull (tipo PNP y NPN)	El encóder o sensor activa las entradas A, B y N alternativamente a M y 24VDC.

Al utilizar encóders incrementales, la selección típica es "Salida push-pull". Cuando se utilizan sensores de 2 hilos, p. ej. barreras fotoeléctricas o detectores de proximidad, se debe seleccionar el cableado "Sensor tipo PNP (en fuente)" o "Sensor tipo NPN (en sumidero)" según corresponda.

Para saber si el encóder incremental es de tipo push/pull, consulte la hoja de datos del encóder.

Nota

Si utiliza un encóder incremental de tipo push/pull, puede vigilar posibles roturas de hilo en las señales del encóder.

Comportamiento con señal N

Este parámetro define cuál será la reacción en caso de señal N.

Puede escoger entre las siguientes opciones:

Opción	Significado
Sin reacción en caso de señal N (predeterminado)	La señal N no afecta al contador.
Sincronización en caso de señal N (Página 9375)	En caso de señal N, el contador se ajusta al valor de arranque. Si se selecciona para una entrada digital la función "Habilitar sincronización con señal N", la sincronización dependerá del nivel detectado en la entrada digital.

Nota

El comportamiento con señal N solo puede seleccionarse si se ha elegido el tipo de señal (Página 9438) "Encóder incremental (A, B, N)".

Nota

Si se selecciona " Sincronización en caso de señal N", puede elegirse para una entrada digital (Página 9443) la función "Habilitar sincronización con señal N".

Frecuencia

Con estos parámetros se define la frecuencia de los siguientes eventos:

- Sincronización en caso de señal N
- Sincronización como función de una entrada digital

Puede escoger entre las siguientes opciones:

Opción	Significado
Único (predeterminado)	El contador solo se ajusta con la primera señal N o el primer flanco parametrizado de la entrada digital.
Periódico	El contador se ajusta con cada señal N o cada flanco parametrizado de la entrada digital.

Comportamiento de una DI: Función de una DI (modo de operación Medición)

Ajuste de la función de la DI

Al parametrizar una entrada digital se define la función que dicha entrada digital desencadenará al conmutarse.

Puede escoger entre las siguientes opciones:

Función de una entrada digital	Significado
Apertura/cierre de puerta (controlados por nivel)	El nivel en la correspondiente entrada digital abre y cierra la puerta HW (Página 9365).
Apertura de puerta (controlada por flanco)	El flanco parametrizado en la correspondiente entrada digital abre la puerta HW (Página 9365).
Cierre de puerta (controlado por flanco)	El flanco parametrizado en la correspondiente entrada digital cierra la puerta HW (Página 9365).
Sincronización (Página 9371)	El flanco parametrizado en la correspondiente entrada digital pone el contador al valor de arranque.
Habilitar sincronización con señal N	El nivel activo en la correspondiente entrada digital habilita la sincronización del contador en caso de señal N (Página 9375).
Entrada digital sin función	La entrada digital en cuestión no tiene asignada función alguna. La CPU puede leer el estado lógico de la entrada digital a través de la interfaz de realimentación (Página 9463).

Nota

Cada función, excepto "Entrada digital sin función", solo puede utilizarse una vez y ya no puede volver a seleccionarse en el resto de las entradas digitales.

Tiempo de filtrado

Al parametrizar el tiempo de filtrado se suprimen las interferencias en las entradas digitales. Las señales con una duración de impulso inferior al tiempo de filtrado parametrizado se suprimen.

Puede escoger entre los siguientes tiempos de filtrado:

- Ninguno
- 0,05 ms
- 0,1 ms (predeterminado)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Nota

Si selecciona la opción "Ninguno" o "0,05 ms", debe utilizar cables apantallados para la conexión de las entradas digitales.

Nota

El tiempo de filtrado solo puede parametrizarse en "Comportamiento DI0" para todas las entradas digitales en conjunto.

Comportamiento de una DI: Opciones de función (modo de operación Medición)

Algunas funciones necesitan otros parámetros con los que pueden detallar el comportamiento. Se pueden parametrizar para la función correspondiente.

Selección de nivel

Con este parámetro se determina el nivel con el que se activa la entrada digital.

Puede escoger entre las siguientes opciones:

Nivel	Significado
Activo con nivel alto (predeterminado)	La correspondiente entrada digital está activa cuando está ajustada.
Activo con nivel bajo	La correspondiente entrada digital está activa cuando no está ajustada.

El parámetro puede parametrizarse en las siguientes funciones de una entrada digital:

- Apertura/cierre de puerta (controlados por nivel)
- Habilitar sincronización con señal N

Selección de flancos

Este parámetro determina con qué flanco de una entrada digital se dispara la función parametrizada.

Puede escoger entre las siguientes opciones:

- Con flanco ascendente (predeterminado)
- Con flanco descendente

El parámetro puede parametrizarse en las siguientes funciones de una entrada digital:

- Apertura de puerta (controlada por flanco)
- Cierre de puerta (controlado por flanco)
- Sincronización

Función de una DQ (modo de medición)

Activar salida

Al parametrizar una salida digital se define la condición (Página 9386) que provocará la conmutación de la entrada digital.

Puede escoger entre las siguientes opciones:

Función de una salida digital	Significado
Valor de medida >= valor de comparación (predeterminado)	La salida digital en cuestión está activa si el valor medido es mayor o igual que el valor de comparación.
Valor de medida <= valor de comparación	La salida digital en cuestión está activa si el valor medido es menor o igual que el valor de comparación.
Entre los valores de comparación 0 y 1	La salida digital DQ1 está activa si el valor medido se encuentra entre el valor de comparación 0 y el valor de comparación 1.
No comprendido entre valores de comparación 0 y 1	La salida digital DQ1 está activa si el valor medido es menor que el valor de comparación 0 o mayor que el valor de comparación 1.
Utilizada por el programa de usuario	La salida digital en cuestión puede conmutarse desde la CPU a través de la interfaz de control (Página 9460).

Nota

Se pueden seleccionar las funciones "Entre valor de comparación 0 y 1" y "No comprendido entre valor de comparación 0 y 1" si se ha seleccionado para la salida digital DQ0 la función "Utilizada por el programa de usuario".

Valor de comparación 0

Al parametrizar el valor de comparación (Página 9393) se determina con qué valor medido se conmutará la salida digital DQ0 de acuerdo con el evento de comparación seleccionado.

Debe introducirse un valor inferior al valor de comparación 1. Puede introducir un máximo de seis decimales.

El ajuste predeterminado es "0.000000". La unidad del valor de comparación depende de la magnitud.

Valor de comparación 1

Al parametrizar el valor de comparación (Página 9393) se determina con qué valor medido se conmutará la salida digital DQ1 de acuerdo con el evento de comparación seleccionado.

Debe introducirse un valor superior al valor de comparación 0. Puede introducir un máximo de seis decimales.

El ajuste predeterminado es "10.000000". La unidad del valor de comparación depende de la magnitud.

Parametrización de la lectura de posición con encóder absoluto SSI

Sinopsis

Si utiliza un encóder absoluto SSI, parametrice lo siguiente:

- Entradas de contaje
  - Especificar señales de entrada/tipo de encóder (Página 9447)
  - Otros parámetros (Página 9447)
- Comportamiento de una DI
  - Función de una DI en el modo de operación Contaje (Página 9449)
  - Función de una DI en el modo de operación Medición (Página 9450)
  - Opciones de función (Página 9451)
- Comportamiento de una DQ
  - Función de una DQ en el modo de operación Contaje (Página 9451)
  - Función de una DQ en el modo de operación Medición (Página 9453)

- Histéresis (Página 9453)
- Especificar valor de medida (Página 9453)

Encontrará dos ejemplos de la estructura del telegrama SSI en el módulo tecnológico en Ejemplos de estructuras de telegrama (Página 9455).

Entradas de contaje: Especificar señales de entrada/tipo de encóder (SSI)

Tipo de señal

Si hay un encóder absoluto SSI (Página 9362) conectado, seleccione el tipo de señal "Encóder absoluto (SSI)".

Invertir sentido

Es posible invertir los valores suministrados por el encóder absoluto SSI para adaptarlos al proceso.

Entradas de contaje: Otros parámetros (SSI)

Longitud de telegrama

Con la parametrización de la longitud de telegrama se define el número de bits de un telegrama SSI. Para saber la longitud de telegrama del encóder absoluto SSI, consulte la hoja de datos del encóder. Un bit de paridad presente no cuenta en la longitud de telegrama.

Se permite una longitud de telegrama de entre 10 y 40 bits. El ajuste predeterminado es "13 Bit".

Tipo de código

Con la parametrización del tipo de código se define si el encóder proporciona códigos binarios naturales o códigos Gray.

Puede escoger entre las siguientes opciones:

Tipo de código	Significado
Gray (predeterminado)	El valor de posición emitido en código Gray por el encóder absoluto SSI se convierte a código binario natural.
Binario	El valor emitido por el encóder absoluto SSI no se convierte.

Velocidad de transferencia

Al parametrizar la velocidad de transferencia se define la velocidad de la transferencia de datos del encóder absoluto SSI al módulo tecnológico. Puede escoger, entre varias opciones, entre 125 kHz y 2 MHz. El ajuste predeterminado es "125 kHz".

La velocidad de transferencia máxima depende de la longitud de línea y de los datos técnicos del encóder absoluto SSI. Encontrará más información en la descripción del encóder.

Tiempo monoestable

Al parametrizar el tiempo monoestable se define el tiempo de pausa entre dos telegramas SSI.

El tiempo monoestable parametrizado debe ser mayor que el tiempo monoestable del encóder absoluto SSI utilizado. Este valor figura en los datos técnicos del encóder absoluto SSI.

Puede escoger entre las siguientes opciones:

- Automático (predeterminado)
- 16 µs
- 32 µs
- 48 µs
- 64 µs

Nota

Si utiliza la función de sistema "Modo isócrono", la opción "Automático" corresponde a un tiempo monoestable de 64 µs.

Paridad

Al parametrizar la paridad se define si el encóder absoluto SSI transferirá un bit de paridad.

Si, p. ej., un encóder de 25 bits está parametrizado con paridad, el módulo tecnológico lee 26 bits. El bit que sigue al LSB (least significant bit) en SSI se interpreta como un bit de paridad. Los errores de paridad se notifican en la interfaz de realimentación (Página 9463) a través del bit ENC_ERROR.

Número de bit del LSB del valor de posición

Con este parámetro se define el número de bit del LSB (Least significant bit) del valor de posición en el telegrama del encóder absoluto SSI. Así se limita la resolución del encóder absoluto SSI. El valor debe ser inferior al número de bit del MSB del valor de posición. La diferencia entre los números de bit del MSB y del LSB del valor de posición debe ser menor que 31.

El ajuste predeterminado es "0".

Número de bit del MSB del valor de posición

Con este parámetro se define el número de bit del MSB (Most significant bit) del valor de posición en el telegrama del encóder absoluto SSI. Así se limita la resolución del encóder absoluto SSI. El valor debe ser inferior a la longitud del telegrama y superior al número de bit del LSB del valor de posición. La diferencia entre los números de bit del MSB y del LSB del valor de posición debe ser menor que 31.

El ajuste predeterminado es "12".

Nota

Cuando se ha seleccionado "Gray" como tipo de código, solo se convierte en código binario natural el rango de LSB a MSB del valor de posición.

Comportamiento de una DI: Función de una DI (SSI, modo de operación Contaje)

Ajuste de la función de la DI

Al parametrizar una entrada digital se define la función que dicha entrada digital desencadenará al conmutarse.

Puede escoger entre las siguientes opciones:

Función de una entrada digital	Significado
Capture (Página 9370)	El flanco parametrizado en la entrada digital correspondiente guarda el valor de posición actual como valor de Capture. La función puede utilizarse solo para una entrada digital.
Entrada digital sin función	La entrada digital en cuestión no tiene asignada función alguna. La CPU puede leer el estado lógico de la entrada digital a través de la interfaz de realimentación (Página 9463).

Tiempo de filtrado

Al parametrizar el tiempo de filtrado se suprimen las interferencias en las entradas digitales. Las señales con una duración de impulso inferior al tiempo de filtrado parametrizado se suprimen.

Puede escoger entre los siguientes tiempos de filtrado:

- Ninguno
- 0,05 ms
- 0,1 ms (predeterminado)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Nota

Si selecciona la opción "Ninguno" o "0,05 ms", debe utilizar cables apantallados para la conexión de las entradas digitales.

Nota

El tiempo de filtrado solo puede parametrizarse en "Comportamiento DI0" para todas las entradas digitales en conjunto.

Comportamiento de una DI: Función de una DI (SSI, modo de operación Medición)

Ajuste de la función de la DI

La entrada digital correspondiente tiene asignado "Entrada digital sin función". La CPU puede leer el estado lógico de la entrada digital a través de la interfaz de realimentación (Página 9463).

Tiempo de filtrado

Al parametrizar el tiempo de filtrado se suprimen las interferencias en las entradas digitales. Las señales con una duración de impulso inferior al tiempo de filtrado parametrizado se suprimen.

Puede escoger entre los siguientes tiempos de filtrado:

- Ninguno
- 0,05 ms
- 0,1 ms (predeterminado)
- 0,4 ms
- 0,8 ms
- 1,6 ms
- 3,2 ms
- 12,8 ms
- 20 ms

Nota

Si selecciona la opción "Ninguno" o "0,05 ms", debe utilizar cables apantallados para la conexión de las entradas digitales.

Nota

El tiempo de filtrado solo puede parametrizarse en "Comportamiento DI0" para todas las entradas digitales en conjunto.

Comportamiento de una DI: Opciones de función (SSI)

Selección de flancos

Este parámetro determina para la función "Captura (Capture)" con qué flanco de una entrada digital se dispara la función parametrizada.

Puede escoger entre las siguientes opciones:

- Con flanco ascendente (predeterminado)
- Con flanco descendente
- Con flanco ascendente y descendente

Comportamiento de una DQ: Función de una DQ (SSI, modo de operación Contaje)

Activar salida

Al parametrizar una salida digital se define la condición (Página 9383) que provocará la conmutación de la salida digital.

Puede escoger entre las siguientes opciones:

Función de una salida digital	Significado
Entre valor de comparación y límite de contaje superior (predeterminado)	La salida digital correspondiente está activa si el valor de posición se encuentra entre el valor de comparación y el límite superior.
Entre valor de referencia y límite de contaje inferior	La salida digital correspondiente está activa si el valor de posición se encuentra entre "0" y el valor de comparación.
Entre los valores de comparación 0 y 1	La salida digital DQ1 está activa si el valor de posición se encuentra entre el valor de comparación 0 y el valor de comparación 1.
Con valor de comparación por la duración del impulso	La salida digital correspondiente está únicamente activa para el tiempo y el sentido del cambio de valor de posición parametrizados si el valor de posición coincide con el valor de comparación o si lo ha rebasado por exceso o por defecto.

Función de una salida digital	Significado
Tras comando Set de CPU hasta valor referencia	Cuando se produce un comando Set de la CPU, la salida digital correspondiente está activa para el sentido parametrizado del cambio de valor de posición hasta que el valor de posición coincide con el de comparación o hasta que lo ha rebasado por exceso o por defecto.
Utilizada por el programa de usuario	La salida digital en cuestión puede conmutarse desde la CPU a través de la interfaz de control (Página 9460).

Nota

Entonces es posible seleccionar la función "Entre valor de comparación 0 y 1" si se ha seleccionado para la salida digital DQ0 la función "Utilizada por el programa de usuario".

Valor de comparación 0

Al parametrizar el valor de comparación se determina con qué valor de posición se conmutará la salida digital DQ0 de acuerdo con el evento de comparación seleccionado.

Debe introducirse un valor inferior al valor de comparación 1.

El ajuste predeterminado es "0".

Valor de comparación 1

Al parametrizar el valor de comparación se determina con qué valor de posición se conmutará la salida digital DQ1 de acuerdo con el evento de comparación seleccionado.

Debe introducirse un valor superior al valor de comparación 0.

El ajuste predeterminado es "10".

Sentido de contaje

Con este parámetro se define para qué sentido del cambio del valor de posición es válida la función seleccionada.

Puede escoger entre las siguientes opciones:

Sentido del cambio del valor de posición	Significado
En ambos sentidos (predeterminado)	La comparación y la conmutación de la salida digital correspondiente tienen lugar independientemente de si el valor de posición aumenta o disminuye.
Ascendente	La salida digital correspondiente solo se compara y se conmuta si el valor de posición aumenta.
Descendente	La salida digital correspondiente solo se compara y se conmuta si el valor de posición disminuye.

El parámetro puede parametrizarse con las siguientes funciones:

- Con valor de comparación por la duración del impulso
- Tras comando Set de CPU hasta valor referencia

Duración del impulso

Al parametrizar la duración de impulso para la función "Con valor de comparación por la duración del impulso", se determina durante cuántos milisegundos estará activa la correspondiente salida digital.

Se admiten valores entre 0,5 y 6553,5 s.

El ajuste predeterminado es "500,0" y corresponde a una duración de impulso de 0,5 s.

Comportamiento de una DQ: Función de una DQ (SSI, modo de operación Medición)

En el modo de operación Medición se definen los parámetros descritos en el apartado Función de una DQ (modo de medición) (Página 9443) para las salidas digitales del canal.

Histéresis (SSI)

Ajustar rango de histéresis

Con la parametrización de la histéresis (Página 9397) se define un rango para los valores de comparación. En el rango de histéresis las salidas digitales no pueden conmutarse de nuevo hasta que el valor de posición haya salido una vez de este rango.

Si se introduce "0" se desactiva la histéresis. Se admiten valores entre 0 y 255. El ajuste predeterminado es "0".

La histéresis puede parametrizarse únicamente en el modo de operación Contaje.

Especificación del valor medido (SSI)

Magnitud

Con este parámetro se define qué magnitud (Página 9394) debe suministrar el módulo tecnológico.

Puede escoger entre las siguientes opciones:

Magnitud	Significado
Frecuencia (predeterminado)	La magnitud es una frecuencia. La unidad es Hz. Se miden los cambios incrementales del valor de posición.
Duración del período	La magnitud es una duración del período que coincide con el valor inverso de la frecuencia. La unidad es s.

Magnitud	Significado
Velocidad	La magnitud es una velocidad. La base de tiempo para la medición de la velocidad y los Incrementos por unidad se parametrizan por separado.
Telegrama SSI completo	En lugar de una magnitud se devuelven los primeros 32 bits del telegrama SSI. Se suministran bits adicionales que no pertenecen a la verdadera información de posición. No se tiene en cuenta una inversión del sentido parametrizada.

Tiempo de actualización

Con la parametrización del tiempo de actualización (Página 9393) en milisegundos se define el intervalo de tiempo que transcurre entre dos actualizaciones del valor medido.

El tiempo de actualización afecta a la precisión de la medición.

Cuando se introduce "0", el valor medido se actualiza en cada ciclo. Puede introducir un máximo de tres decimales. Se admiten valores entre 0,000 y 25000,000. El ajuste predeterminado es "10,000".

Base de tiempo para medición de velocidad

Con este parámetro se define la base de tiempo con la que debe ponerse a disposición la velocidad.

Puede escoger entre las siguientes opciones:

- 1 ms
- 10 ms
- 100 ms
- 1 s
- 60 s/1 min

El ajuste predeterminado es "60 s/1 min".

Incrementos por unidad

Con este parámetro se define el número de incrementos que proporciona el encóder absoluto SSI por vuelta.

Se admiten valores entre 1 y 65535.

Ejemplo:

Su encóder funciona con una resolución de 12 bits por vuelta y emite 4096 incrementos por vuelta. La velocidad debe medirse en vueltas por minuto.

En este caso, parametrice lo siguiente:

- Incrementos por unidad: 4096
- Base de tiempo para medición de velocidad: 60 s/1 min

Ejemplos de estructuras de telegrama

Ejemplo 1

En este ejemplo, el encóder absoluto SSI tiene la siguiente especificación:

- El telegrama SSI tiene una longitud de 25 bits.
- El MSB del valor de posición es el bit 24.
- El LSB del valor de posición es el bit 0.
- El valor de posición tiene codificación Gray.
- No existe un bit de paridad.

El telegrama presenta la siguiente estructura:

Ciclo:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
Bit:	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	MG	SG	SG
	MSB																								LSB

Valor de posición

MG Bit Multiturn con codificación Gray

SG Bit Singleturn con codificación Gray

Telegrama SSI completo

Si en lugar de una magnitud desea recibir el telegrama SSI no procesado, este se devuelve justificado a la derecha en la interfaz de realimentación:

Ciclo:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG

MG Bit Multiturn con codificación Gray

SG Bit Singleturn con codificación Gray

Valor de realimentación del valor de posición

El módulo tecnológico convierte el valor de posición suministrado en codificación Gray a código binario y lo devuelve justificado a la derecha en la interfaz de realimentación:

Ciclo:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD	SD

MD Bit Multiturn como código binario

SD Bit Singleturn como código binario

Ejemplo 2

En este ejemplo, el encóder absoluto SSI tiene la siguiente especificación:

- El telegrama SSI tiene una longitud de 34 bits.
- El MSB del valor de posición es el bit 33.
- El LSB del valor de posición es el bit 6.
- El valor de posición tiene codificación Gray.
- El telegrama SSI tiene dos bits especiales.
- Existe un bit de paridad. El bit de paridad no cuenta en la longitud de telegrama.

El telegrama presenta la siguiente estructura:

Ciclo:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
Bit:	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	*	3	2	1	0
	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	MG	SG	SG	SG	SG	SG	S1	S2	P	0	0	0	0
	MSB																											LSB							

Valor de posición

MG Bit Multiturn con codificación Gray

SG Bit Singleturn con codificación Gray

S1 Bit de estado 1

S2 Bit de estado 2

P Bit de paridad

Telegrama SSI completo

Si en lugar de una magnitud desea recibir el telegrama SSI no procesado, el módulo tecnológico lee como máximo los 32 menos significativos del encóder. El módulo tecnológico devuelve el bit que sigue al LSB como bit de paridad. Por ello, en este ejemplo el módulo tecnológico lee los 31 bits menos significativos del telegrama SSI.

El telegrama SSI completo devuelto presenta la siguiente estructura:

Ciclo:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MG	MG	MG	MG	MG	MG	MG	MG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	SG	MG	SG	SG	P	S1	S2	0	0	0	0

MG Bit Multiturn con codificación Gray

SG Bit Singleturn con codificación Gray

Valor de posición

El módulo tecnológico convierte el valor de posición suministrado en codificación Gray a código binario y lo devuelve justificado a la derecha en la interfaz de realimentación:

Ciclo:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	SD	SD	SD	SD	SD	SD	SD	MD	SD	SD	SD	SD	SD	SD	SD	SD	SD

MD Bit Multiturn como código binario

SD Bit Singleturn como código binario

Módulo Online y diagnóstico

Indicación y evaluación del diagnóstico

Por medio de la vista online y de diagnóstico puede realizar un diagnóstico del hardware. Además, puede:

- Recibir información a través del módulo tecnológico (p. ej. la versión de Firmware y el número de serie).
- Si es necesario, realice una actualización del firmware.

Procedimiento

Para abrir el editor de visualización para las funciones de diagnóstico, haga lo siguiente:

1. Abra la carpeta "Módulos locales" o "Periferia descentralizada" en el árbol del proyecto.
2. Abra el módulo tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Online y diagnóstico".
4. Haga clic en la visualización deseada en el árbol de diagnóstico.

Más información

Para más información sobre los avisos de diagnóstico y sobre posibles remedios, consulte el manual de producto del módulo tecnológico.

Nota

Lectura de posición para Motion Control

En el modo de funcionamiento "Lectura de posición para Motion Control", el diagnóstico de canal no se encuentra disponible para el módulo tecnológico.

Poner en servicio el módulo

El editor de puesta en servicio le ayudará a poner en servicio el módulo tecnológico y realizar su prueba funcional.

Puede simular señales de contaje y el control de los estados de las entradas digitales sin cableado del proceso. De este modo, puede observar y comprobar el comportamiento del programa de usuario durante el proceso de contaje.

Nota

Las funciones parametrizadas de las salidas digitales y las alarmas de proceso parametrizadas se disparan en función del valor de contaje simulado.

Nota

En el modo de funcionamiento "Lectura de posición para Motion Control" no se utiliza el editor de puesta en servicio.

Consulte también

Asignación de la interfaz de realimentación (Página 9463)

Observación de valores de contaje, valores medidos, DI y DQ

Requisitos

- Hay una conexión online entre STEP 7 (TIA Portal) y la CPU.
- La CPU se encuentra en el estado operativo RUN.

Procedimiento

Para abrir el editor de "Puesta en servicio" de un módulo tecnológico, haga lo siguiente:

1. Abra la carpeta "Módulos locales" o "Periferia descentralizada" en el árbol del proyecto.
2. Abra el módulo tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Puesta en servicio".

El editor de puesta en servicio conmuta automáticamente al modo online. En el lado derecho se muestra en la Task Card "Test" el panel de control de la CPU, que permite controlar el estado operativo de la CPU.

Observar valores

Los siguientes valores se leen cíclicamente en el módulo tecnológico y se muestran en "Observar valores":

- Valor de contaje
- Valor medido
- Valor de captura (Capture)
- Código de error

Los siguientes valores se leen una vez en el módulo tecnológico y se muestran en "Observar valores":

- Valor de comparación 0
- Valor de comparación 1
- Límite de contaje superior
- Límite de contaje inferior
- Valor de arranque

En la columna "Formato de visualización", el formato de los valores mostrados puede alternarse entre decimal y hexadecimal. Para el valor medido se aplica además el formato de visualización "Número en coma flotante".

Observar flags de estado

La información de estado del módulo tecnológico se lee y se muestra online. Si la CPU está en STOP, no se actualiza la indicación de estado.

Observar DI y DQ

Los estados lógicos de las entradas y salidas digitales del módulo tecnológico se muestran en "Observar/forzar DI & DQ". En la columna "Comentario" se muestra la función parametrizada actual. La consola de manejo situada por encima de la tabla permite modificar manualmente los valores y simular así las funciones de contaje y medición: ver Controlar manualmente valores de contaje, DI y DQ (Página 9459)

Controlar manualmente valores de contaje, DI y DQ

En el marco de un control manual puede modificar online el estado de las entradas digitales del módulo tecnológico sin cableado del proceso y especificar un valor de contaje. Mediante una función de simulación y la especificación de una velocidad de simulación, se puede observar el efecto de las funciones de medición y contaje. Las salidas digitales se conmutan en función del valor de contaje simulado y las alarmas de proceso correspondientes se disparan.

Requisitos

- Hay una conexión online entre STEP 7 (TIA Portal) y la CPU.
- La CPU se encuentra en el estado operativo RUN.
- La ventana de puesta en servicio "Observar/forzar DI & DQ" para el módulo tecnológico está abierta. Se muestran los valores actuales del módulo tecnológico.

Control manual

> ⚠ ADVERTENCIA
>
> **Advertencia de daños personales y materiales**
>
> Si el control manual no se ejecuta correctamente, existe peligro de lesiones graves y daños materiales en máquinas e instalaciones:
>
> - Compruebe todas las entradas antes de cambiar al control manual y antes de aplicar los valores modificados durante el control manual.
> - El control manual está reservado exclusivamente a personal especializado.

Procedimientos básicos: control manual de valores de contaje, DI y DQ

1. Para activar el control manual, pulse el botón "Activar" en el panel de control.
2. Introduzca los valores nuevos en la columna vacía "Valor de control".
   La casilla de verificación situada a la derecha se activará automáticamente. El triángulo amarillo de advertencia indica que este valor todavía no se ha transferido al módulo tecnológico.
3. Introduzca un valor para la "Velocidad de simulación". Para un TM Count los límites del rango de valores son de ±800000,00. Para un TM PosInput, los límites del rango de valores son de ±4000000,00.
   El valor determina la velocidad de las señales de contaje simuladas.
4. Para aceptar los valores modificados con casilla de verificación activada, pulse el botón "Modificar".
   El triángulo amarillo de advertencia se apagará. Los valores modificados se actualizarán en el módulo tecnológico.
5. Para controlar la simulación de la función de contaje o medida, utilice los botones "Iniciar" y "Parar".
6. Observe los indicadores en "Observar valores", "Observar flags de estado" y "Observar/forzar DI & DQ" y compruebe las reacciones de su programa de usuario.
7. Adapte si es necesario los valores de la columna "Valor de control" repitiendo los pasos 2 a 4.

Interfaz de control y realimentación

Encontrará información sobre el uso de la interfaces de control y realimentación en el resumen de las posibles aplicaciones (Página 9358).

Asignación de la interfaz de control

El programa de usuario gobierna el comportamiento del módulo tecnológico a través de la interfaz de control.

Interfaz de control por canal

La siguiente tabla muestra la asignación de la interfaz de control:

Offset respecto a la dirección inicial	Parámetro	Significado				
Bytes 0 … 3	Slot 0	Valor de carga (el significado del valor se especifica en LD_SLOT_0)				
Bytes 4 … 7	Slot 1	Valor de carga (el significado del valor se especifica en LD_SLOT_1)				
Byte 8	LD_SLOT_0*	Especifica el significado del valor en Slot 0				
		Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	
		0	0	0	0	Ninguna acción, estado de reposo
		0	0	0	1	Cargar valor de contaje (con encóders incrementales o generadores de impulsos)
		0	0	1	0	Reservado
		0	0	1	1	Cargar valor de arranque (con encóders incrementales o generadores de impulsos)
		0	1	0	0	Cargar valor de comparación 0
		0	1	0	1	Cargar valor de comparación 1
		0	1	1	0	Cargar límite de contaje inferior (con encóders incrementales o generadores de impulsos)
		0	1	1	1	Cargar límite de contaje superior (con encóders incrementales o generadores de impulsos)
		1	0	0	0	Reservado
		a				
		1	1	1	1	
	LD_SLOT_1*	Especifica el significado del valor en Slot 1				
		Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	
		0	0	0	0	Ninguna acción, estado de reposo
		0	0	0	1	Cargar valor de contaje (con encóders incrementales o generadores de impulsos)
		0	0	1	0	Reservado
		0	0	1	1	Cargar valor de arranque (con encóders incrementales o generadores de impulsos)
		0	1	0	0	Cargar valor de comparación 0
		0	1	0	1	Cargar valor de comparación 1
		0	1	1	0	Cargar límite de contaje inferior (con encóders incrementales o generadores de impulsos)
		0	1	1	1	Cargar límite de contaje superior (con encóders incrementales o generadores de impulsos)
		1	0	0	0	Reservado
		a				
		1	1	1	1	
Byte 9	EN_CAPTURE	Bit 7: Habilitación de la función Capture				
	EN_SYNC_DN	Bit 6: Habilitación sincronización descendente (con encóders incrementales o generadores de impulsos)				
	EN_SYNC_UP	Bit 5: Habilitación sincronización ascendente (con encóders incrementales o generadores de impulsos)				

Offset respecto a la dirección inicial	Parámetro	Significado
	SET_DQ1	Bit 4: Activación DQ1
	SET_DQ0	Bit 3: Activación DQ0
	TM_CTRL_DQ1	Bit 2: Habilitación de la función tecnológica DQ1
	TM_CTRL_DQ0	Bit 1: Habilitación de la función tecnológica DQ0
	SW_GATE	Bit 0: Puerta de software (con encóders incrementales o generadores de impulsos)
Byte 10	SET_DIR	Bit 7: Sentido de contaje (con encóders sin señal de sentido)
	–	Bits 2 a 6: Reservados; los bits deben estar ajustados a 0
	RES_EVENT	Bit 1: Desactivar eventos guardados
	RES_ERROR	Bit 0: Desactivar estados de fallo guardados
Byte 11	–	Bits 0 a 7: Reservados; los bits deben estar ajustados a 0

* Si a través de LD_SLOT_0 y LD_SLOT_1 se cargan valores simultáneamente, se aplica primero el valor de Slot 0 y a continuación el valor de Slot 1 de forma interna. De ese modo, pueden producirse estados intermedios inesperados.

Aclaraciones sobre los bits de control

Bit de control	Explicaciones
EN_CAPTURE	Este bit habilita la función de Capture. El restablecimiento del bit desactiva un EVENT_CAP activado en la interfaz de realimentación.
EN_SYNC_DN	Con este bit se habilita la sincronización del contador al contar en sentido descendente si se utiliza un encóder incremental o un generador de impulsos. El restablecimiento del bit desactiva un EVENT_SYNC activado en la interfaz de realimentación.
EN_SYNC_UP	Con este bit se habilita la sincronización del contador al contar en sentido ascendente si se utiliza un encóder incremental o un generador de impulsos. El restablecimiento del bit desactiva un EVENT_SYNC activado en la interfaz de realimentación.
RES_EVENT	Con este bit se inicia la desactivación de los eventos guardados en los bits de realimentación EVENT_ZERO, EVENT_OFLW, EVENT_UFLW, EVENT_CMP0, EVENT_CMP1 .
RES_ERROR	Con este bit se inicia la desactivación de los estados de error guardados LD_ERROR y ENC_ERROR .
SET_DIR	Con este bit se predefine el sentido de contaje con el tipo de señal "Impulso (A)". 0 significa: ascendente 1 significa: descendente
SET_DQ0	Con este bit se activa la salida digital DQ0 cuando TM_CTRL_DQ0 está establecido en 0. Con la función "Tras comando Set de CPU hasta valor referencia", SET_DQ0 actúa independientemente de TM_CTRL_DQ0, siempre y cuando el valor de contaje no coincida con el valor de comparación.
SET_DQ1	Con este bit se activa la salida digital DQ1 cuando TM_CTRL_DQ1 está establecido en 0. Con la función "Tras comando Set de CPU hasta valor referencia", SET_DQ1 actúa independientemente de TM_CTRL_DQ1, siempre y cuando el valor de contaje no coincida con el valor de comparación.

Bit de control	Explicaciones
SW_GATE	Con este bit se abre y cierra la puerta de software si se utiliza un encóder incremental o un generador de impulsos. La puerta software, junto con la puerta hardware, forma la puerta interna. El módulo tecnológico solo efectúa el contaje si la puerta interna está abierta. 0 significa: puerta de software cerrada 1 significa: puerta de software abierta
TM_CTRL_DQ0	Con este bit se habilita la función tecnológica de la salida digital DQ0. 0 significa: SET_DQ0 determina el estado de DQ0 1 significa: la función parametrizada determina el estado de DQ0
TM_CTRL_DQ1	Con este bit se habilita la función tecnológica de la salida digital DQ1. 0 significa: SET_DQ1 determina el estado de DQ1 1 significa: la función parametrizada determina el estado de DQ1

Asignación de la interfaz de realimentación

A través de la interfaz de realimentación, el programa de usuario recibe valores actuales e información de estado del módulo tecnológico.

Interfaz de realimentación por canal

La siguiente tabla muestra la asignación de la interfaz de realimentación:

Offset respecto a la dirección inicial	Parámetro	Significado
Bytes 0 … 3	COUNT VALUE	Valor de contaje o valor de posición actuales
Bytes 4 … 7	CAPTURED VALUE	Último valor de Capture captado
Bytes 8 … 11	MEASURED VALUE	Valor medido actual o telegrama SSI completo
Byte 12	–	Bits 3 a 7: reservado; establecido en 0
	LD_ERROR	Bit 2: error al cargar a través de interfaz de realimentación
	ENC_ERROR	Bit 1: Señal de encóder o telegrama SSI incorrectos
	POWER_ERROR	Bit 0: Módulo S7-1500: tensión de alimentación L+ incorrecta/módulo ET 200SP: tensión de alimentación demasiado baja L+
Byte 13	–	Bits 6 a 7: reservado; establecido en 0
	STS_SW_GATE	Bit 5: Estado puerta SW (con encóders incrementales o generadores de impulsos)
	STS_READY	Bit 4: módulo tecnológico arrancado y parametrizado
	LD_STS_SLOT_1	Bit 3: solicitud de carga para Slot 1 detectada y ejecutada (conmutada)
	LD_STS_SLOT_0	Bit 2: solicitud de carga para Slot 0 detectada y ejecutada (conmutada)
	RES_EVENT_ACK	Bit 1: reset de los bits de evento activo
	–	Bit 0: reservado; establecido en 0
Byte 14	STS_DI2 (TM Count)	Bit 7: TM Count: estado DI2/TM PosInput: reservado; establecido en 0
	STS_DI1	Bit 6: Estado DI1
	STS_DI0	Bit 5: Estado DI0
	STS_DQ1	Bit 4: Estado DQ1
	STS_DQ0	Bit 3: Estado DQ0

Offset respecto a la dirección inicial	Parámetro	Significado
	STS_GATE	Bit 2: Estado puerta interna (con encóders incrementales o generadores de impulsos)
	STS_CNT	Bit 1: Impulso de contaje o modificación del valor de posición registrado en los últimos 0,5 s aprox.
	STS_DIR	Bit 0: Sentido del último cambio del valor de contaje o posición
Byte 15	STS_M_INTERVAL	Bit 7: Impulso de contaje o modificación del valor de posición registrado en el último intervalo de medida
	EVENT_CAP	Bit 6: se ha producido un evento de Capture
	EVENT_SYNC	Bit 5: Se lleva a cabo la sincronización (con encóders incrementales o generadores de impulsos)
	EVENT_CMP1	Bit 4: Se ha producido un evento de comparación para DQ1
	EVENT_CMP0	Bit 3: Se ha producido un evento de comparación para DQ0
	EVENT_OFLW	Bit 2: Se ha producido un rebase por exceso
	EVENT_UFLW	Bit 1: Se ha producido un rebase por defecto
	EVENT_ZERO	Bit 0: paso por cero producido

Aclaraciones sobre los bits de realimentación

Bit de realimentación	Explicaciones
ENC_ERROR	Este bit indica que, con el módulo tecnológico respectivo, se ha producido uno de los siguientes errores en las señales del encóder (con memoria): TM Count: • Rotura de hilo de entrada digital A, B o N (con encóder tipo push-pull) • Transición ilegal de las señales A/B (en encóders incrementales) TM PosInput: • Transición ilegal de las señales A/B (en encóders incrementales) • Error RS422/TTL • Error en el encóder SSI o en el telegrama SSI (con encóders absolutos SSI) Si se han habilitado las alarmas de diagnóstico, en caso de error en las señales del encóder, se dispara la alarma de diagnóstico correspondiente. Infórmese con el manual de producto del módulo tecnológico respectivo sobre el significado de las alarmas de diagnóstico. El bit se desactiva después de que se haya acusado el error con RES_ERROR .
EVENT_CAP	Este bit muestra que se ha producido un evento de Capture y que se había guardado un valor de contaje en CAPTURED VALUE . Este estado se desactiva reiniciando EN_CAPTURE .
EVENT_CMP0	Este bit muestra el estado almacenado de que se ha producido un evento de comparación para la salida digital DQ0. Este estado se desactiva acusando RES_EVENT. Al ajustar el valor de contaje al valor de arranque, no se activa el bit EVENT_CMP0.
EVENT_CMP1	Este bit muestra el estado almacenado de que se ha producido un evento de comparación para la salida digital DQ1. Este estado se desactiva acusando RES_EVENT. Al ajustar el valor de contaje al valor de arranque, no se activa el bit EVENT_CMP1.
EVENT_OFLW	Este bit indica el estado almacenado de que el valor de contaje tenía un rebase por exceso. Este estado se desactiva acusando RES_EVENT.

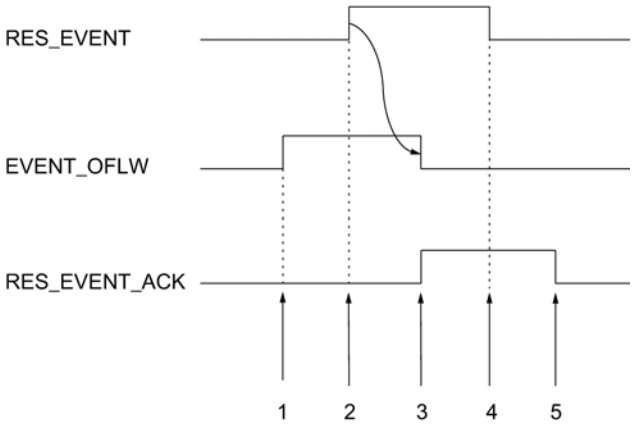
Bit de realimentación	Explicaciones
EVENT_SYNC	Este bit indica el estado almacenado de que el contador se ha cargado con el valor de arranque por una señal de referencia externa (sincronización) si se utiliza un encóder incremental o un generador de impulsos. Este estado se activa restableciendo EN_SYNC_UP o EN_SYNC_DN .
EVENT_UFLW	Este bit indica el estado almacenado de que el valor de contaje tenía un rebase por defecto. Este estado se desactiva acusando RES_EVENT.
EVENT_ZERO	Este bit indica el estado almacenado de que el valor de contaje o el valor de posición tenía un paso por cero. Este estado se desactiva acusando RES_EVENT.
LD_ERROR	Este bit indica que se ha producido un error al cargar a través de la interfaz de control (con memoria). No se han adoptado los valores de carga. Si se utiliza un encóder incremental o un generador de impulsos, no se cumple una de las siguientes condiciones: • Límite inferior de contaje <= valor de contaje <= límite superior de contaje • límite inferior de contaje <= valor de arranque <= límite superior de contaje • Límite inferior de contaje <= valor de comparación 0/1 <= límite superior de contaje • Valor de comparación 0 < valor de comparación 1 Si se utiliza un encóder absoluto SSI, no se cumple una de las siguientes condiciones: • 0 <= valor de posición <= valor de posición máximo • 0 <= valor de comparación 0/1 <= valor de posición máximo • Valor de comparación 0 < valor de comparación 1 El bit se desactiva después de que se haya acusado el error con RES_ERROR .
LD_STS_SLOT_0	Este bit indica con un cambio de estado (conmutación) que la solicitud de carga para Slot 0 (LD_SLOT_0) ha sido detectada y ejecutada.
LD_STS_SLOT_1	Este bit indica con un cambio de estado (conmutación) que la solicitud de carga para Slot 1 (LD_SLOT_1) ha sido detectada y ejecutada.
POWER_ERROR	Este bit muestra para un módulo tecnológico S7-1500 que la tensión de alimentación L+ no existe, que es demasiado baja o que el conector frontal no está conectado. Este bit muestra para un módulo tecnológico ET 200SP que la tensión de alimentación L+ es demasiado baja. Si la tensión de alimentación L+ vuelve a estar disponible en la magnitud necesaria, POWER_ERROR se ajusta automáticamente a 0.
RES_EVENT_ACK	Este bit indica que la desactivación de los bits de evento EVENT_SYNC, EVENT_CMP0, EVENT_CMP1, EVENT_OFLW, EVENT_UFLW, EVENT_ZERO está activa.
STS_CNT	Este bit indica que dentro de los últimos 0,5 s aprox. se ha producido al menos un impulso de contaje o un cambio del valor de posición.
STS_DI0	Este bit indica el estado de la entrada digital DI0.
STS_DI1	Este bit indica el estado de la entrada digital DI1.
STS_DI2	Este bit indica el estado de la entrada digital DI2 de TM Count .
STS_DIR	Este bit muestra el sentido de contaje del último impulso de contaje o el sentido del último cambio del valor de posición. 0 significa: descendente 1 significa: ascendente
STS_DQ0	Este bit indica el estado de la salida digital DQ0.
STS_DQ1	Este bit indica el estado de la salida digital DQ1.
STS_GATE	Este bit muestra el estado de la puerta interna si se utiliza un encóder incremental o un generador de impulsos. 0 significa: puerta cerrada 1 significa: puerta abierta
STS_M_INTERVAL	Este bit indica que en el intervalo de medida precedente se ha capturado al menos un impulso de contaje o un cambio del valor de posición.

Bit de realimentación	Explicaciones
STS_READY	Este bit indica que el módulo tecnológico emite datos de usuario válidos. El módulo tecnológico está arrancado y parametrizado.
STS_SW_GATE	Este bit indica el estado de la puerta SW. 0 significa: puerta cerrada 1 significa: puerta abierta

Método de acuse completo

Los bits guardados se acusan conforme al método de acuse completo.

La siguiente figura muestra un ejemplo del proceso del método de acuse completo en caso de rebase por exceso:



① El bit de realimentación EVENT_OFLW se activa como evento con memoria en caso de rebase por exceso.

② Establezca el bit de control RES_EVENT para impulsar el restablecimiento de EVENT_OFLW .

③ El bit de realimentación RES_EVENT_ACK se activa si se ha detectado la desactivación de EVENT_OFLW.

④ Se restablece el bit de control RES_EVENT .

⑤ El bit de realimentación RES_EVENT_ACK se restablece.

11.1.2 Contaje y medición (S7-300, S7-400)

11.1.2.1 Contaje y medición

Principios básicos de contaje y medición

¿En qué consisten las tareas de contaje y medición?

Contaje y medición

Contar significa registrar y sumar eventos. Los módulos de contaje registran las señales de encóder (impulsos) y las evalúan según corresponda.

En el modo de medición, los impulsos se detectan de forma precisa en el tiempo y, a partir de ellos, se calculan los valores medidos correspondientes (por ejemplo, la frecuencia).

Conceptos básicos sobre la operación de contaje

Rangos y límites de contaje

En los dos rangos de contaje "0 a +32 bits" y "-31 a +31 bits", el módulo cuenta dentro de límites de contaje diferentes. En los respectivos límites de contaje se detecta un rebase por exceso o por defecto.

En el rango de contaje "-31 a +31 bits", el estado del contador se representa como complemento a 2.

Rango de contaje		Rebase por exceso	Rebase por defecto
0 a +32 bits [1]	0 a 4.294.967.295 0 a FFFF FFFFH	Al cambiar el estado del contador de 4 294 967 295 a 0	Al cambiar el estado del contador de 0 a 4 294 967 295
-31 a +31 bits	–2 147 483 648 a 2 147 483 647 8000 0000H a 7FFF FFFFH	Al cambiar el estado del contador de +2 147 483 647 a –2 147 483 648	Al cambiar el estado del contador de –2 147 483 648 a 2 147 483 647
[1] Este rango de contaje sólo está disponible con FM x50-1. Los valores sólo se pueden predeterminar y evaluar en formato hexadecimal.			

Valor de carga

El valor de carga es el valor del contador a partir del cual el módulo comienza el contaje.

Durante el funcionamiento es posible predeterminar para el módulo un valor de carga LOAD_VAL que sobrescribe el valor inicial.

El valor de carga se puede predeterminar **directamente**. En este caso, el módulo lo adoptará directamente como nuevo valor de contaje y lo cargará de manera preparatoria.

El valor de carga se puede cargar de forma sólo **preparatoria**. Un valor de carga cargado de forma preparatoria será adoptado por el módulo como nuevo valor de contaje en los eventos siguientes:

- En todos los modos de contaje
  - Al iniciar el contaje por interrupción de la puerta SW o HW (en la función de puerta de interrupción al iniciar la puerta SW o HW se sigue contando desde el estado actual del contador y no se adopta el valor de carga).
  - Sincronización (a través del DI correspondiente)
  - Congelar/redisparar (a través del DI correspondiente)
- Además, en los modos de contaje "Contaje único" y "Contaje periódico"
  - Cuando se alcance el límite de contaje superior parametrizado con el sentido de contaje principal adelante.
  - Cuando se alcance 0 en el sentido de contaje principal atrás.

Control de puerta

La puerta de hardware (puerta HW) y la puerta de software (puerta SW) sirven para controlar, es decir, iniciar y detener, los procesos de contaje del módulo.

Sentido principal de contaje

Si se parametriza un sentido principal de contaje (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño, ajustando para ello un límite de contaje superior. El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado. Ello permite p. ej. crear aplicaciones de contaje incrementales o decrementales. El límite de contaje principal parametrizado no tiene efecto en la evaluación del sentido cuando se detectan los impulsos de contaje.

Valores iniciales tras realizar la parametrización

Valor	Sentido principal de contaje	Valor inicial
Valor de carga	Ninguno	0
	Adelante	0
	Atrás	Límite de contaje superior parametrizado
Valor de contaje	Ninguno	0
	Adelante	0
	Atrás	Límite de contaje superior parametrizado

Valor	Sentido principal de contaje	Valor inicial
Valores de comparación 1 y 2	Ninguno Adelante Atrás	0 0 Límite de contaje superior parametrizado
Valor latch	Ninguno Adelante Atrás	0 0 Límite de contaje superior parametrizado

Comandos en los modos de contaje

Es posible aplicar comandos al contaje del módulo durante el funcionamiento:

Comando	Descripción
Abrir y cerrar puerta	El contaje comienza cuando se abre una puerta y se detiene al cerrarse la misma.
Activar contador [1]	El contador se pone en el valor de carga mediante diferentes señales (DI, señal de paso por cero).
Congelar/redisparar [2]	Guardar el estado del contador y cargar el contador con el valor de carga al ocurrir un flanco ascendente en DI-Start.
Congelar [2]	Guardar el estado del contador al ocurrir un flanco ascendente en DI-Start.
Medir tiempos entre dos flancos [3]	Medir tiempos entre dos flancos inmediatamente consecutivos en la entrada digital DI-Start.
[1] sólo con FM 350-1 y FM 450-1 [2] sólo con FM 350-1, ET 200S 1Count24V y ET 200S 1Count5V [3] sólo con FM 350-1	

Modo isócrono

En modo isócrono, el módulo adopta bits de control y elementos de control de la interfaz de control en cada ciclo PROFIBUS/PROFINET y devuelve su respuesta a ello en ese mismo ciclo.

El módulo transfiere en cada ciclo PROFIBUS/PROFINET la lectura del contador o el valor de congelación tal y como eran en el instante T_i, y los bits de estado tal y como eran en el instante T_i.

Un estado del contador influido por señales de entrada de hardware sólo se transfiere durante el propio ciclo cuando la señal de entrada ha aparecido antes del instante T_i.

Modos de contaje

Descripción general de los modos de contaje

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el módulo. La tabla siguiente ofrece una visión de conjunto de los modos de contaje.

Modo de operación	Descripción
Contaje sin fin (Página 9470) (con o sin puerta SW o puerta HW)	El módulo cuenta en modo sin fin a partir del estado actual del contador.
Contaje único (Página 9471) (con puerta SW o con puerta HW)	El módulo comienza a contar a partir del valor de carga al abrirse la puerta y se detiene en el límite de contaje.
Contaje periódico (Página 9472) (con puerta SW o con puerta HW)	El módulo cuenta al abrirse la puerta entre el valor de carga y el límite de contaje.

Contaje sin fin

Descripción

En este modo de operación, el módulo cuenta continuamente a partir del estado del contador:

- Cuando el contador alcanza el límite superior contando hacia adelante y recibe un impulso de contaje adicional, salta al límite inferior de contaje y sigue contando desde allí sin perder impulsos.
- Cuando el contador alcanza el límite inferior contando hacia atrás y recibe un impulso de contaje adicional, salta al límite de contaje superior y sigue contando desde allí sin perder impulsos.

Rango de contaje de ±31 bits:

- El límite de contaje superior está ajustado a +2 147 483 647 (2^{31} - 1).
- El límite de contaje inferior está ajustado a -2 147 483 648 (-2^{31}).

Rango de contaje de 32 bits:

- El límite de contaje superior está ajustado a +4.294.967.295 (2^{32} - 1).
- El límite de contaje inferior está ajustado a 0 (cero).

Comportamiento en los límites de contaje

Cuando el contador alcanza el límite superior o inferior y recibe un impulso de contaje adicional, el contador se ajusta al límite de contaje opuesto. En la interfaz de realimentación y, en dado caso, en el DB se activa el bit de estado correspondiente.

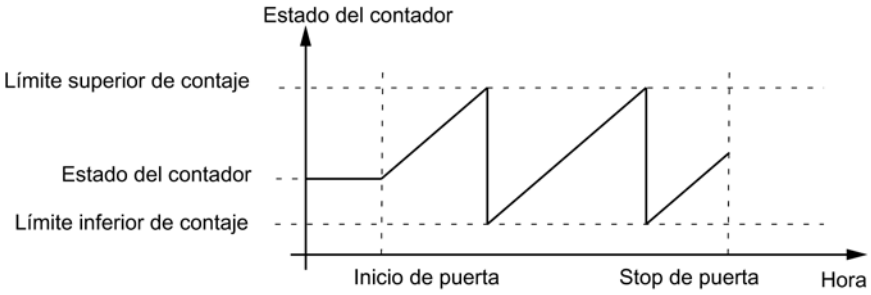
Límite de contaje alcanzado	Bit de estado
Límite superior de contaje	STS_OFLW se activa
Límite inferior	STS_UFLW se activa

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Existen las siguientes posibilidades:

- Sin puerta (estándar)
- Puerta SW (Página 9490)
- Puerta HW (Página 9490), con control de nivel o de flanco

La figura siguiente muestra un ejemplo de contaje sin fin con control de puerta.



Contaje único

Descripción

En este modo de operación, el módulo cuenta una vez hasta un límite de contaje en el sentido de contaje principal parametrizado y detiene luego el contaje automáticamente.

Comportamiento en los límites de contaje

Cuando el contador alcanza el límite superior o inferior y recibe un impulso de contaje adicional, el contador se ajusta

- en el caso del contaje sin sentido principal de contaje (Página 9473) al límite de contaje opuesto.,
- al valor de carga si se está contando con sentido de contaje principal (Página 9473).

A continuación, la puerta se cierra y el contaje finaliza, aunque el parámetro/el bit de control SW_GATE siga activado o la puerta HW continúe abierta. En la interfaz de notificación/en el DB se activa el bit de estado correspondiente:

Límite de contaje alcanzado	Bit de estado
Límite superior de contaje	STS_OFLW se activa
Límite inferior de contaje	STS_UFLW se activa

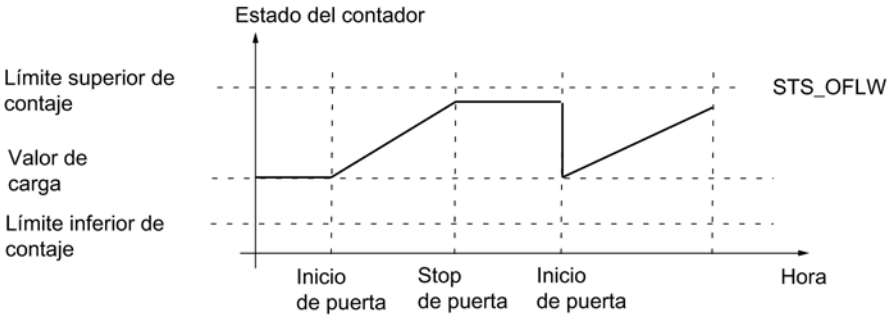
Para volver a iniciar el contador es preciso volver a activar el parámetro/el bit de control SW_GATE o volver a abrir la puerta HW. El contaje continuará entonces a partir del valor de carga.

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Para ello se dispone de varias posibilidades, a saber:

- Puerta SW (Página 9490)
- Puerta HW (Página 9490), con control de nivel o de flanco

La figura siguiente muestra un ejemplo de contaje único con valor de carga y control de puerta.



Contaje periódico

Descripción

En este modo de operación, el módulo cuenta periódicamente con la puerta abierta, sin detener el contaje en los límites.

Comportamiento en los límites de contaje

Si el contador

- ha alcanzado uno de los límites al contar sin sentido de contaje principal (Página 9473),
- ha alcanzado el límite superior al contar con sentido de contaje principal adelante (Página 9473),
- ha alcanzado el límite inferior al contar con sentido de contaje principal atrás (Página 9473),

y recibe un impulso de contaje adicional, el contador se ajustará al valor de carga a partir del cual se sigue contando. En la interfaz de notificación/en el DB se activa el bit de estado correspondiente.

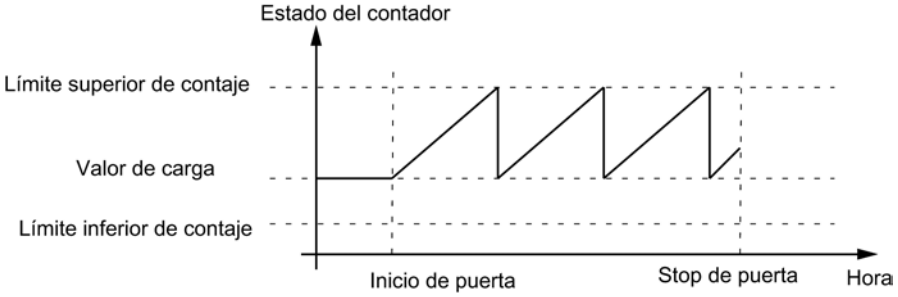
Límite de contaje alcanzado	Bit de estado
Límite superior de contaje	STS_OFLW se activa
Límite inferior de contaje	STS_UFLW se activa

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Para ello se dispone de varias posibilidades, a saber:

- Puerta SW (Página 9490)
- Puerta HW (Página 9490), con control de nivel o de flanco

La figura siguiente muestra un ejemplo de contaje periódico con valor de carga y control de puerta.



Sentido principal de contaje

El sentido principal de contaje sólo puede seleccionarse en los modos de operación "Contaje único", "Contaje periódico" y "Dosificación".

Es posible parametrizar el comportamiento siguiente:

- Ninguno (sin sentido principal de contaje)
- Adelante
- Atrás

Contaje único sin sentido principal de contaje

En el modo de operación “Contaje único" sin sentido principal de contaje, el módulo cuenta desde el valor de carga – tras abrirse la puerta – adelante o atrás, hasta excederse uno de los límites de contaje.

Al excederse uno de los límites de contaje se

- cierra la puerta,
- activa el bit STS_OFLW o STS_UFLW en la interfaz de notificación,
- activa el contador hasta el límite de contaje opuesto.

Los límites de contaje están ajustados fijamente al rango de contaje máximo.

El bit STS_ZERO o STS_ND se activa si el estado del contador es cero (pero no en el caso de FM 350-2).

Para reiniciar el contaje es preciso abrir la puerta de nuevo.

Contaje periódico sin sentido de contaje principal

En el modo de operación “Contaje periódico" sin sentido principal de contaje, el módulo cuenta desde el valor de carga – tras abrirse la puerta – adelante o atrás, hasta excederse uno de los límites de contaje.

Al excederse uno de los límites de contaje se

- activa el bit STS_OFLW o STS_UFLW en la interfaz de notificación,
- ajusta el contador al valor de carga a partir del cual se sigue contando.

Los límites de contaje están ajustados fijamente al rango de contaje máximo.

El bit STS_ZERO o STS_ND se activa si el estado del contador es cero (pero no en el caso de FM 350-2).

El contaje continúa hasta que se cierre la puerta.

Contaje único con sentido principal de contaje adelante

En los modos des operación "Contaje único" y "Dosificación" con sentido principal de contaje adelante, el módulo cuenta desde el valor de carga – tras abrirse la puerta – adelante o atrás, hasta excederse el límite de contaje superior.

Al excederse el límite de contaje superior se

- cierra la puerta,
- activa el bit STS_OFLW en la interfaz de notificación,
- carga en el contador el valor de carga.

El límite de contaje superior se puede parametrizar, el valor de carga tiene su valor inicial y se puede modificar.

Para reiniciar el contaje es preciso abrir la puerta de nuevo.

Contaje periódico con sentido de contaje principal adelante

En el modo de operación "Contaje periódico" con sentido de contaje principal adelante, el módulo cuenta desde el valor de carga – tras abrirse la puerta – adelante o atrás, hasta excederse el límite de contaje superior.

Al excederse el límite de contaje superior se

- activa el bit STS_OFLW en la interfaz de notificación,
- ajusta el contador al valor de carga a partir del cual se sigue contando.

El límite de contaje superior se puede parametrizar, el valor de carga tiene su valor inicial y se puede modificar.

El contaje continúa hasta que se cierre la puerta.

Contaje único con sentido principal de contaje atrás

En los modos des operación "Contaje único" y "Dosificación" con sentido principal de contaje atrás, el módulo cuenta desde el valor de carga – tras abrirse la puerta – adelante o atrás, hasta excederse el límite de contaje inferior.

Al excederse el límite de contaje inferior se

- cierra la puerta,
- activa el bit STS_UFLW en la interfaz de notificación,
- carga en el contador el valor de carga.

El límite de contaje inferior se establece de forma fija a 0, el valor de carga tiene su valor inicial y se puede modificar.

Para reiniciar el contaje es preciso abrir la puerta de nuevo.

Contaje periódico con sentido de contaje principal atrás

En el modo de operación "Contaje periódico" con sentido principal de contaje atrás, el módulo cuenta desde el valor de carga – tras abrirse la puerta – adelante o atrás, hasta excederse el límite de contaje inferior.

Al excederse el límite de contaje inferior se

- activa el bit STS_UFLW en la interfaz de notificación,
- ajusta el contador al valor de carga a partir del cual se sigue contando.

El límite de contaje inferior se establece de forma fija a 0, el valor de carga tiene su valor inicial y se puede modificar.

El contaje continúa hasta que se cierre la puerta.

Conceptos básicos sobre la operación de medición

Rangos de medida

En los modos de medición existen los siguientes rangos de medida para los grupos:

Modo de operación	Rango de medida
Medición de frecuencia	0,1 Hz [1] a 650.000 Hz (según el módulo)
Medición de velocidad	hasta 25.000 /min
Medición del período	hasta 120 s

[1] El límite inferior de 0,1 Hz está condicionado por el tiempo de actualización máxima de 10 s. La resolución del valor es 0,001 Hz.

Tiempo de actualización/integración

El módulo actualiza los valores medidos cíclicamente. Para calcular el tiempo de actualización/integración predetermine el parámetro "Tiempo de medición" como factor n.

Cálculo del tiempo de actualización/integración:

Condiciones marco		Tiempo de actualización/ integración	Rango de valores de n	
			$n_{mín}$	$n_{máx}$
Modo no isócrono	T_{DP} cualquiera	n × 10 ms	1	1000 [3]
Funcionamiento isócrono[2]	T_{DP} < 10 ms	n × T_{DP}	(10 ms/T_{DP} [ms]) + 1 [1]	1000 [3]
	T_{DP} ≥ 10 ms	n × T_{DP}	1	10000 [3] ms/T_{DP} [ms] [1]

[1] Las posiciones decimales que surgen con la división entre T_{DP} se suprimen.

Estos límites no se pueden infringir. En caso de exceder estos límites, el módulo generará un error de parametrización y no cambiará al modo isócrono.

[2] no con FM 350-2

[3] en la medición del período: 120000

Una vez transcurrido cada tiempo de actualización/integración, el valor medido de la interfaz de realimentación se actualiza y el bit STS_CMP1 o STS_COMP1 se activa.

Principio de medición

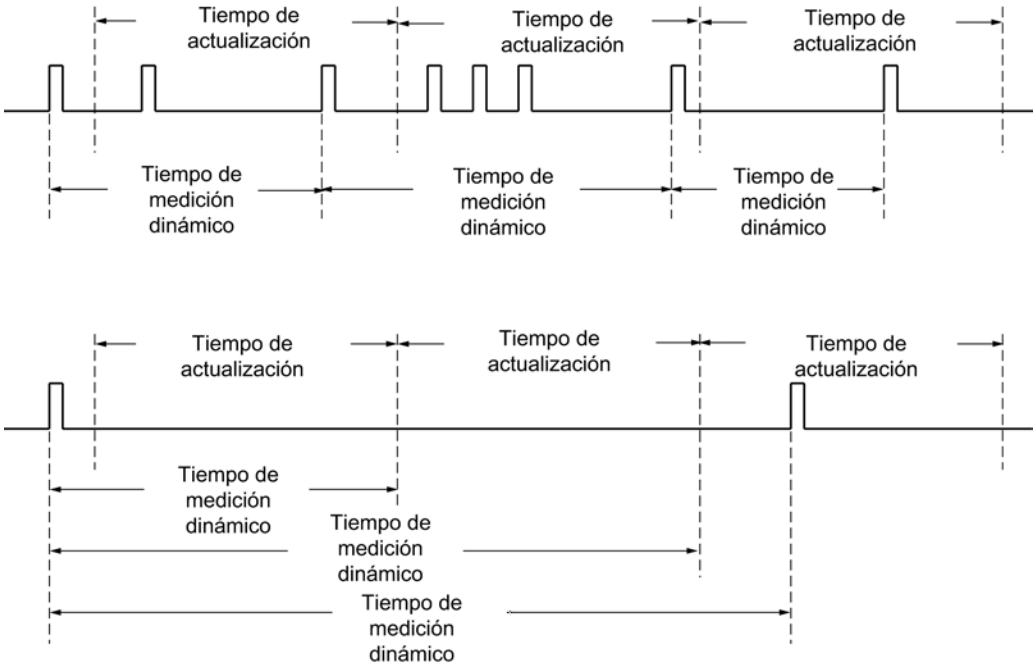
El módulo cuenta cada flanco ascendente de un impulso y le asigna un valor de tiempo en µs.

El tiempo de medición dinámico está definido como la diferencia entre dos valores de tiempo.

En una secuencia de impulsos con uno o varios impulsos por intervalo de tiempo de actualización vale lo siguiente:

Tiempo de medición dinámico =	valor de tiempo del último impulso en el intervalo actual del tiempo de actualización menos Valor de tiempo del último impulso en el intervalo de tiempo de actualización precedente.

Si tras calcular el tiempo de medición dinámico aparecen intervalos de actualización sin impulsos, entonces el tiempo de medición se prolonga en estos intervalos. Si el valor "1 impulso por tiempo de medición dinámico" es inferior al último valor medido, entonces este valor se emite como nuevo valor. La figura muestra el principio de medición.



Procedimiento de medición

El módulo mide de forma continua. En la parametrización se preselecciona un tiempo de actualización.

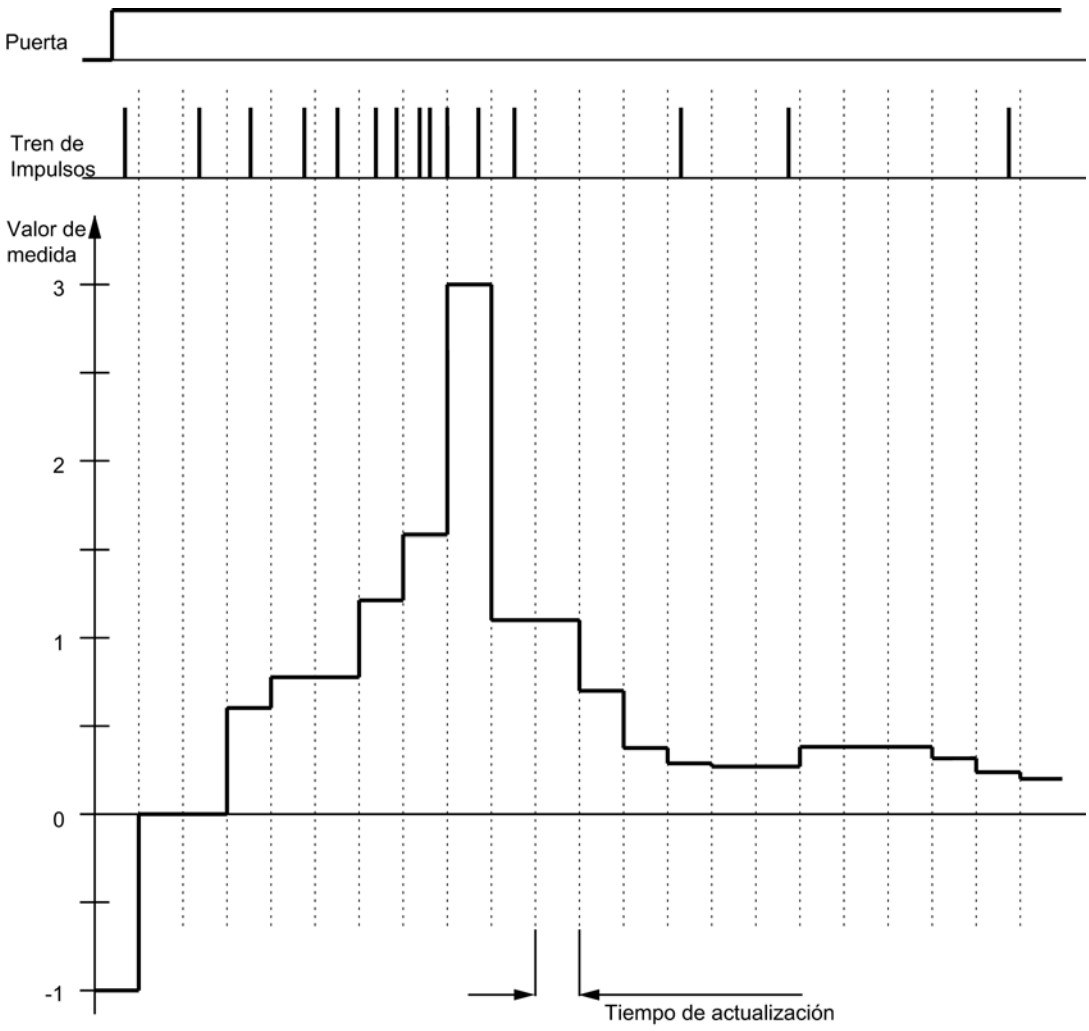
En el tiempo que transcurre hasta finalizar el primer tiempo de actualización se retorna el valor "-1". El primer tiempo de actualización comienza al abrise la puerta.

La medición continua comienza tras abrirse la puerta con el primer impulso de la secuencia de impulsos a medir. El primer valor medido no se calcula hasta después del segundo impulso.

Cada vez que finaliza el tiempo de actualización se retorna un valor medido en la interfaz de realimentación (frecuencia, duración del periodo o velocidad). El final de una medición se notifica mediante los bits de estado STS_CMP1 o STS_COMP1 (en FM 350-2). Este bit se resetea mediante el principio de realimentación completo con los bits RES_ZERO y STS_RES_ZERO o RES_STS y RES_STS_A.

Si el sentido de rotación se invierte durante un tiempo de actualización, el valor medido para este período es incierto. Evaluando los bits de realimentación para el sentido es posible reaccionar a posibles irregularidades del proceso.

La figura siguiente explica el principio de la medición continua con el ejemplo de una medición de frecuencia.



Vigilancia de valores límite

Los valores límite cumplen todas y cada una de las condiciones siguientes:

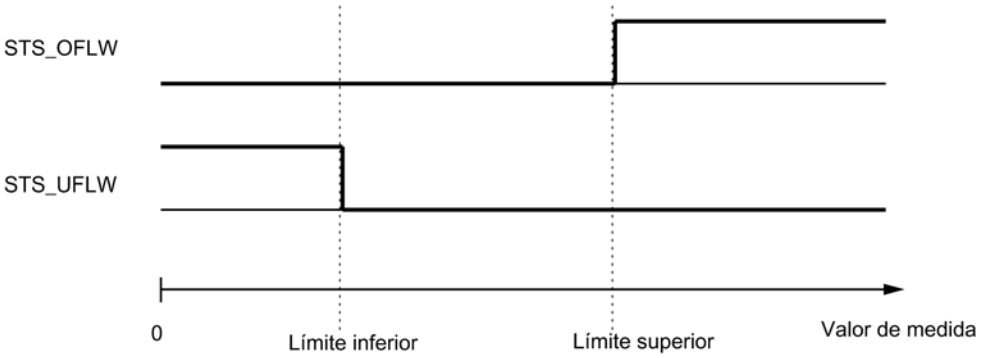
- Límite inferior ≥ límite inferior del rango de medida no alcanzado
- Límite superior ≥ límite superior del rango de medida no alcanzado
- Límite superior ≥ (límite inferior +1)

Cada vez que finaliza un tiempo de actualización se compara el valor medido (frecuencia, velocidad o duración del período) con los valores límite parametrizados superior e inferior. La comparación da el siguiente resultado:

Si el valor medido es...	...se activa el siguiente bit en el estado:
mayor que el límite superior,	STS_OFLW
menor que el límite inferior,	STS_UFLW

Además se puede generar una alarma de proceso.

La figura siguiente muestra el estado del bit de estado para la vigilancia de límites en los modos de medición.



Los bits STS_OFLW y STS_UFLW se tienen que desactivar con los bits RES_ZERO y STS_RES_ZERO o RES_STS mediante el principio de acuse completo. Si después del acuse el valor medido se sigue encontrando o se encuentra nuevamente fuera de los límites, se vuelve a activar el bit de estado.

Con la parametrización adecuada también se puede conmutar la salida correspondiente DOn con la vigilancia de límites.

Control de puerta

La puerta de hardware (puerta HW) y la puerta de software (puerta SW) sirven para controlar, es decir, iniciar y detener, los procesos de medición.

Comandos en los modos de medición

Es posible influir durante el funcionamiento en el proceso de medición del módulo con comandos:

Denominación	Descripción
Abrir y cerrar puerta	El proceso de medición se inicia al abrir una puerta y se finaliza al cerrarla.

Modo isócrono

En el modo isócrono, el módulo toma en cada ciclo PROFIBUS/PROFINET las señales de control de la interfaz de control en el instante T_o. De este modo todos los controles se ejecutan de forma síncrona y se hacen efectivos en el instante T_o. La respuesta al controlador se devuelve ya en el mismo ciclo PROFIBUS/PROFINET.

El módulo devuelve un valor medido y los bits de estado en cada ciclo PROFIBUS/PROFINET en el instante T_i.

La medición empieza y termina siempre en el instante T_i.

Nota

Como el tiempo de actualización en el modo no isócrono se indica en múltiplos enteros de 10 ms, pero en el modo isócrono en múltiplos enteros del tiempo de ciclo de bus, al cambiar entre uno y otro modo de operación hay que adaptar siempre el parámetro Tiempo de actualización si se desea mantener el tiempo de actualización real.

Modos de medición

Descripción general de los modos de medición

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el módulo. La tabla siguiente ofrece una visión de conjunto de los modos de medición.

Modo de operación	Descripción
Medición de frecuencia (Página 9480)	El módulo cuenta los impulsos que llegan en un tiempo de medición dinámico [1] y calcula a partir de ahí la frecuencia.
Medición de velocidad de rotación (Página 9482)	El módulo cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de medición dinámico [1] y calcula a partir de ello la velocidad con los impulsos por vuelta del encóder.
Medición del periodo (Página 9483)	El módulo indica el tiempo de medición dinámico [1] como duración del período. Si el período es inferior al tiempo de actualización, se crea un valor medio para el período.

[1] El tiempo de medición dinámico significa: Si no se detectan 2 flancos en la señal de medición antes de transcurrir el tiempo de medición parametrizado, se calcula un valor medido aproximado con el tiempo de medición dinámico. En este caso se supone que, al final del tiempo de actualización, aparece un impulso. Con ello se obtiene un valor medido continuo.

Nota

El FM 450-1 no soporta ningún modo de medición.

Medición de frecuencia

Descripción

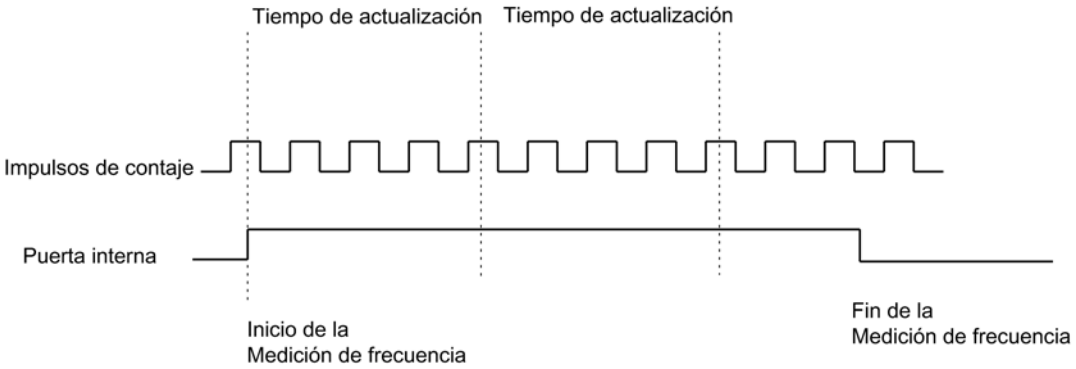
En este modo de operación, el módulo cuenta los impulsos que aparecen en un tiempo de medición dinámico o en un tiempo de integración especificado. La frecuencia se calcula a partir de aquí en combinación con las marcas de tiempo de los flancos.

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Para ello se dispone de varias posibilidades, a saber:

- Puerta SW (Página 9490)
- Puerta HW (Página 9490), con control de nivel o de flanco

La figura siguiente muestra un ejemplo de medición de frecuencia con control de puerta.



Rangos de medición posibles con información de errores

La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones de frecuencia continuas.

Frecuencia f	Error absoluto	
	FM 350-1 ET 200S 1Count5V	ET 200S 1Count24V
0,1 Hz	± 0,001 Hz	
1 Hz	± 0,001 Hz	
10 Hz	± 0,003 Hz	
100 Hz	± 0,02 Hz	
1 000 Hz	± 0,18 Hz	
10 000 Hz	± 1,8 Hz	
100.000 Hz	± 18 Hz	
500 000 Hz	± 90 Hz	–

La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones de frecuencia con tiempo de integración.

Tiempo de integración	f_{min} ± error absoluto	$f_{máx}$ ± error absoluto	
		ET 200S 1Count24V	ET 200S 1Count5V
10 s	0,1 Hz ± 0,001 Hz	100.000 Hz ± 18 Hz	500.000 Hz ± 90 Hz
1 s	1 Hz ± 0,001 Hz	100.000 Hz ± 11 Hz	500.000 Hz ± 55 Hz
0,1 s	10 Hz ± 0,002 Hz	100.000 Hz ± 10 Hz	500.000 Hz ± 52 Hz
0,01 s	100 Hz ± 0,013 Hz	100.000 Hz ± 13 Hz	500.000 Hz ± 63 Hz

Valores modificables durante el funcionamiento

Durante el funcionamiento es posible modificar los siguientes valores:

- Valor límite inferior
- Valor límite superior
- Tiempo de actualización (tiempo de medición) [1)]
- Función de la salida digital [1)]

[1)] no con FM 350-2

Resultado

El final de una medición de frecuencia se notifica con el bit de estado STS_CMP1 o STS_COMP1 (con FM 350-2). El valor de la frecuencia medida se pone a disposición con la unidad 10^{-3} Hz. El valor de frecuencia medido puede leerse en la interfaz de notificación.

Medición de velocidad

Descripción

En este modo de operación, el módulo cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de medición dinámico o en un tiempo de integración especificado y mide el intervalo entre los flancos. A partir ahí calcula la velocidad con los impulsos por vuelta del encóder.

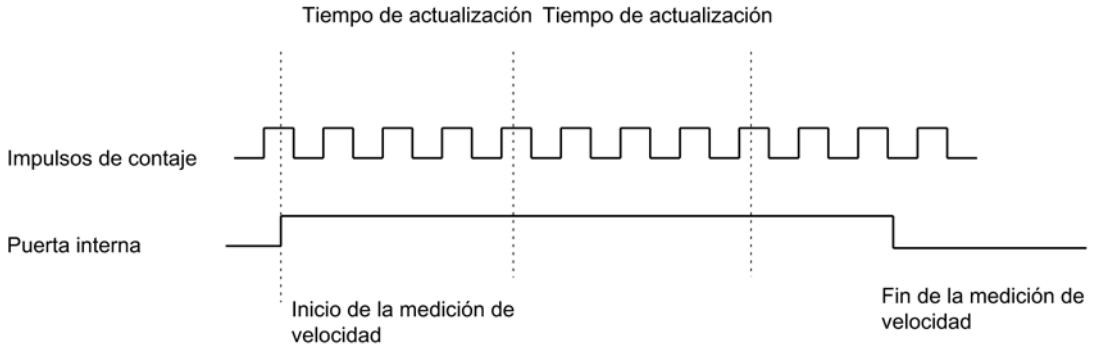
Para el modo "Medición de velocidad" es preciso parametrizar adicionalmente los impulsos por vuelta del encóder.

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Para ello se dispone de varias posibilidades, a saber:

- Puerta SW (Página 9490)
- Puerta HW (Página 9490), con control de nivel o de flanco

La figura siguiente muestra un ejemplo de medición de velocidad con control de puerta.



Rangos de medición posibles con información de errores

La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones de velocidad continuas cuando el número de impulsos por vuelta del encóder = 60.

Velocidad n	Error absoluto
1/min	± 0,04 /min
10 /min	± 0,04 /min
100 /min	± 0,05 /min
1.000 /min	± 0,21 /min
10.000 /min	± 1,82 /min
25.000 /min	± 4,50 /min

La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones de velocidad con tiempo de integración cuando el número de impulsos por vuelta del encóder = 60.

Tiempo de integración	$n_{mín}$ ± error absoluto	$n_{máx}$ ± error absoluto
10 s	1 /min ± 0,03 /min	25.000 /min ± 4,5 /min
1 s	1 /min ± 0,03 /min	25.000 /min ± 2,8 /min
0,1 s	10 /min ± 0,03 /min	25.000 /min ± 2,6 /min
0,01 s	100 /min ± 0,04 /min	25.000 /min ± 3,2 /min

Valores modificables durante el funcionamiento

Durante el funcionamiento es posible modificar los siguientes valores:

- Valor límite inferior
- Valor límite superior
- Tiempo de actualización (tiempo de medición) [1)]
- Función de la salida digital [1)]

[1)] no con FM 350-2

Resultado

El final de una medición de velocidad se notifica con el bit de estado STS_CMP1 o STS_COMP1 (con FM 350-2). El valor de la velocidad medida se pone a disposición con la unidad 10^{-3} /min. La velocidad medida puede leerse en la interfaz de notificación.

Medición del período

Descripción

En este modo de operación, el grupo indica el tiempo de medición dinámico como período. Si el período es inferior al tiempo de actualización, se crea un valor medio para el período.

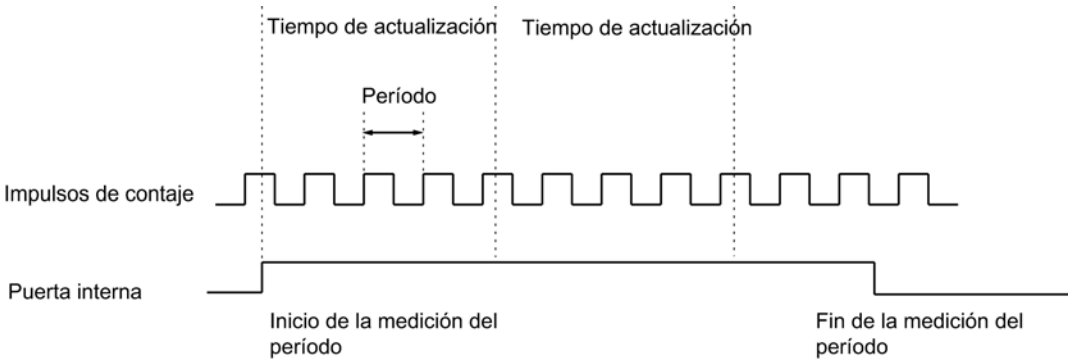
En el modo de operación "Medición del período con tiempo de integración", el 1Count24V mide el tiempo entre dos flancos ascendentes de la señal de contaje contando los impulsos de una frecuencia interna de referencia (16 MHz) de precisión de cuarzo dentro de un tiempo de integración especificado.

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Para ello se dispone de varias posibilidades, a saber:

- Puerta SW (Página 9490)
- Puerta HW (Página 9490), con control de nivel o de flanco

La figura siguiente muestra un ejemplo de medición del período con control de puerta.



Rangos de medición posibles con información de errores

Medición continua del período:

- En mediciones continuas del período con una **resolución de 1 µs**, el error absoluto es de 0,1% del valor medido.
- La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones del período continuas con una **resolución de 1/16 µs**.

Período $T_{mín}$ ± error absoluto
1/16 µs × (160 ± 1)
1/16 µs × (1.600 ± 1)
1/16 µs × (16.000 ± 3)
1/16 µs × (160.000 ± 20)
1/16 µs × (1.600.000 ± 160)
1/16 µs × (16.000.000 ± 1.600)
1/16 µs × (160.000.000 ± 16.000)
1/16 µs × (1.600.000.000 ± 160.000)

Mediciones de período con tiempo de integración:

- La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones de período con tiempo de integración con una **resolución de 1 µs**.

Tiempo de integración	T_{min} ± error absoluto	T ± error absoluto
100 s	1 µs × (10 ± 0)	1 µs × (100.000.000 ± 10.000)
10 s	1 µs × (10 ± 0)	1 µs × (10.000.000 ± 1.000)
1 s	1 µs × (10 ± 0)	1 µs × (1.000.000 ± 100)
0,1 s	1 µs × (10 ± 0)	1 µs × (100.000 ± 10)
0,01 s	1 µs × (10 ± 0)	1 µs × (10.000 ± 1)

- La tabla siguiente muestra los rangos de medida y el error absoluto para las mediciones de período con tiempo de integración con una **resolución de 1/16 µs**.

Tiempo de integración	T_{min} ± error absoluto	T ± error absoluto
100 s	1/16 µs × (160 ± 0)	1/16 µs × (1.600.000.000 ± 160.000)
10 s	1/16 µs × (160 ± 0)	1/16 µs × (160.000.000 ± 16.000)
1 s	1/16 µs × (160 ± 0)	1/16 µs × (16.000.000 ± 1.600)
0,1 s	1/16 µs × (160 ± 0)	1/16 µs × (1.600.000 ± 160)
0,01 s	1/16 µs × (160 ± 0)	1/16 µs × (160.000 ± 16)

Valores modificables durante el funcionamiento

Durante el funcionamiento es posible modificar los siguientes valores:

- Valor límite inferior
- Valor límite superior
- Tiempo de actualización (tiempo de medición) [1)]
- Función de la salida digital [1)]

[1)] no con FM 350-2

Resultado

El final de una medición del período se notifica con el bit de estado STS_CMP1 o STS_COMP1 (con FM 350-2). El valor del período medido se indica en 1 µs o 1/16 µs. El período medido puede leerse en la interfaz de notificación.

Dosificación (sólo con FM 350-2)

Descripción

En este modo de operación se agrupan cuatro canales de contaje del FM 350-2 en un solo canal de dosificación. Cuando la puerta está abierta, el FM 350-2 cuenta una única vez en el sentido principal de contaje:

- Entre 0 y el límite de contaje superior parametrizado en el sentido principal de contaje adelante.
  Si se parametriza como sentido principal de contaje el hacia adelante, el valor inicial es 0 y hay que definir el valor final.
- Entre el límite de contaje superior parametrizado y 0 en el sentido principal de contaje atrás.
  Si se parametriza como sentido principal de contaje el hacia atrás, hay que definir el valor inicial y el final es 0.

Seleccionar el control de puerta

En este modo de operación se puede seleccionar el control de puerta. Para ello se dispone de varias posibilidades, a saber:

- Puerta SW (Página 9490)
- Puerta SW (Página 9490) y puerta HW (Página 9490)
  Se puede añadir una puerta hardware a la puerta software. Las dos puertas actúan conjuntamente como una operación lógica AND, lo que significa que el FM 350-2 sólo cuenta cuando ambas están abiertas.

Comportamiento en los límites de contaje, puerta software

Sentido principal de contaje adelante: Cuando el contador ha alcanzado el valor "Límite superior de contaje – 1" y recibe otro impulso de contaje, el contador se pone a 0, la puerta interna se cierra y el contaje se termina, aun cuando el bit SW_GATE0, 4 todavía esté activo. El bit de estado correspondiente STS_OFLW0, 4 se activa. Así, el límite de contaje superior en sí no se alcanza jamás.

Sentido principal de contaje atrás: Cuando el contador ha alcanzado el valor "1" y recibe otro impulso de contaje, el contador se pone en el límite de contaje superior, la puerta interna se cierra y se termina el contaje, aunque el bit SW_GATE0, 4 esté todavía activo. El bit de estado correspondiente STS_UFLW0, 4 se activa. De este modo jamás se llega al valor "0".

Para volver a iniciar el contador, es necesario desactivar el bit SW_GATE0, 4 y activarlo de nuevo.

Comportamiento en los límites de contaje, puerta hardware

Sentido principal de contaje adelante: Cuando el contador ha alcanzado el valor "Límite superior de contaje – 1" y recibe otro impulso de contaje, el contador se pone a 0, la puerta interna se cierra y el contaje se termina, aun cuando el bit SW_GATE0, 4 y la entrada I0, I4 todavía estén activos. Así, el límite de contaje superior en sí no se alcanza jamás. El bit de estado correspondiente STS_OFLW0, 4 se activa.

Sentido principal de contaje atrás: Cuando el contador ha alcanzado el valor "1" y recibe otro impulso de contaje, el contador se establece el límite de contaje superior, la puerta interna se cierra y el contaje se termina, aun cuando el bit SW_GATE0, 4 y la entrada I0, I4 todavía estén activos. De este modo jamás se llega al valor "0". El bit de estado correspondiente STS_UFLW0, 4 se activa.

Para volver a iniciar el contador, es necesario desactivar la entrada I0, I4 y activarla de nuevo. Sólo es posible un nuevo contaje con la puerta hardware.

Valores de comparación

Para cada canal de dosificación del FM 350-2 es posible parametrizar cuatro valores de comparación dentro del rango de contaje parametrizado. También es posible definir por parametrización si y bajo qué condiciones se activará una salida digital y/o se disparará una alarma de proceso en relación con un valor de comparación. Para ello se pueden establecer las condiciones siguientes:

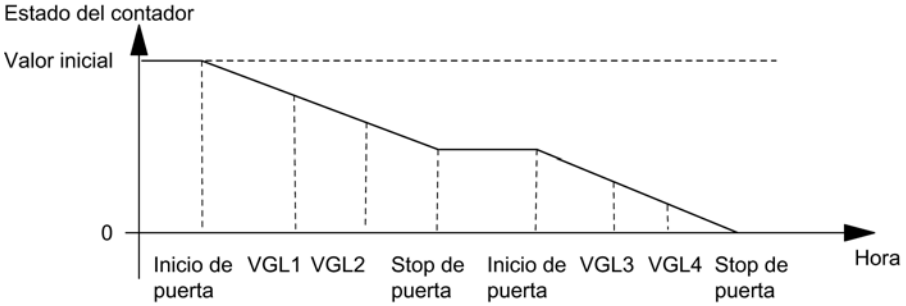
- Cuando la lectura actual del contador coincide con un valor de comparación, se dispara una alarma de proceso.
- Cuando la lectura actual del contador es mayor o igual a un valor de comparación, se activa una salida digital.
- Cuando la lectura actual del contador es menor o igual a un valor de comparación, se activa una salida digital.

La salida digital sólo se activa si ha habilitado la salida correspondiente con el bit CTRL_DQ0…7.

Nota

Las salidas sólo se pueden activar si la CPU está en RUN y la puerta está abierta.

La figura siguiente muestra un ejemplo de dosificación en sentido principal de contaje atrás con valores de comparación.



Valor de carga con anticipación

Dentro del área de contaje parametrizada, es posible definir de forma anticipada un valor de carga con la CPU en modo RUN. El contador emplea este valor como nuevo valor inicial cuando se dan los eventos siguientes:

- Cuando se alcanza el límite de contaje superior en el sentido de contaje principal adelante.
- Al alcanzar el valor 0 cuando el sentido del contaje es hacia atrás.
- Interrupción del contaje mediante una puerta software o hardware (cuando el contaje se interrumpe no se utiliza el valor de carga).

En estos casos, el valor de carga es el nuevo valor inicial a partir del cual comienzan la siguiente dosificación y todas las demás. El comportamiento parametrizado de las salidas y las alarmas se mantiene igual.

Valor de carga directo

La lectura actual del contador se puede modificar con la CPU en modo RUN. El contador tomará la nueva lectura del contador directamente como valor de dosificación actual.

Rango de valores para valores de carga

El rango de valores para los valores de carga depende del sentido principal de contaje parametrizado. Consiste en:

- Desde 0 hasta el límite superior de contaje – 2 en el sentido principal de contaje adelante
- Desde el valor inicial hasta 2 en el sentido de contaje principal atrás

Control de puerta

Funciones de puerta

Contaje con funciones de puerta

Muchas aplicaciones requieren que el contaje comience o se detenga en un instante determinado, dependiendo de otros eventos. En los módulos de contaje, este tipo de inicio o detención del contaje se consigue mediante una función de puerta. Cuando se abre la puerta, los impulsos pasan al contador y comienza el contaje. Cuando la puerta se cierra, los impulsos de contaje ya no pueden llegar al contador y el contaje se detiene.

Puerta software y puerta hardware

Los módulos de contaje poseen dos puertas:

- Una puerta de software (Página 9490) (puerta SW) que se controla mediante el parámetro de entrada/bit de control SW_GATE.
- Una puerta de hardware (Página 9490) (puerta HW), que se controla a través de las señales correspondientes en las entradas digitales del módulo.

Tenga en cuenta también las informaciones sobre el funcionamiento de la puerta en el modo isócrono (Página 9492).

Puerta interna

La puerta interna es la combinación lógica AND de una puerta hardware y una puerta software. El estado de la puerta interna se indica mediante el bit STS_GATE.

Si no se ha parametrizado ninguna puerta hardware, sólo es relevante el ajuste de la puerta software. El contaje se activa, interrumpe, continúa y cancela mediante la puerta interna. La puerta interna también puede cerrarse con eventos que dependan de la lectura del contador en los modos de operación "Contaje único" y "Dosificación".

Puerta hardware	Puerta software	Puerta interna	Proceso de contaje
Abierta	Abierta	Abierta	Activo
Abierta	Cerrada	Cerrada	Inactivo
Cerrada	Abierta	Cerrada	Inactivo
Cerrada	Cerrada	Cerrada	Inactivo

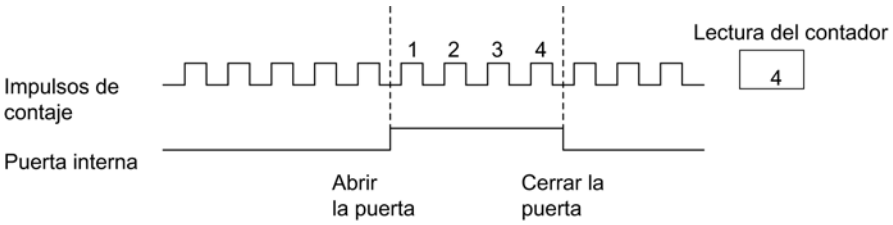
Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cuando se cancela (Página 9491), el contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Cuando se interrumpe (Página 9491), el contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Ejemplo

Al activar la señal de puerta, la puerta se abre y los impulsos se cuentan. Cuando la señal de puerta se retira, ésta se cierra y el contador deja de registrar los impulsos de contaje. La lectura del contador se mantiene constante.

La siguiente figura muestra la apertura y el cierre de una puerta y el contaje de los impulsos.



Finalizar el contaje con la función de parada por puerta

Además, en los módulos FM x50-1 el contaje se puede finalizar independientemente de las señales aplicadas o del estado de la puerta SW con la función de parada por puerta. A este efecto, active el parámetro de entrada GATE_STP.

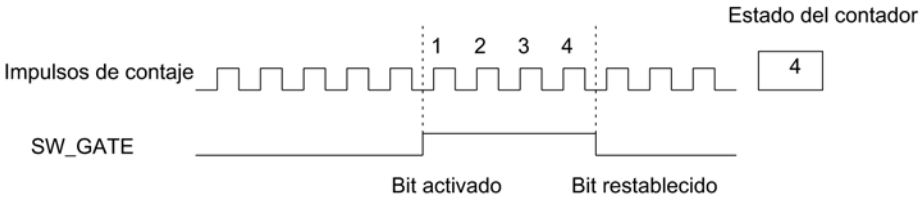
Si desactiva este parámetro, entonces solo podrá abrir la puerta con un flanco ascendente en la entrada digital "DI-Start" (puerta HW) o volviendo a activar el parámetro de entrada SW_GATE (puerta SW).

Puerta SW

La puerta SW se abre y se cierra con el parámetro de entrada/bit de control SW_GATE.

Apertura y cierre de la puerta SW

La figura siguiente muestra la apertura y el cierre de la puerta SW. No es posible parametrizar ninguna puerta HW.



La puerta SW:

- se abre activando el parámetro de entrada/bit de control SW_GATE.
- se cierra activando el parámetro de entrada/bit de control SW_GATE.

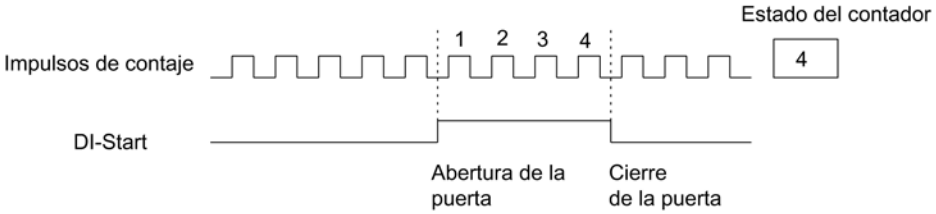
Una puerta cerrada puede volver a abrirse volviendo a activar el parámetro de entrada SW_GATE.

Puerta HW

La puerta HW se abre y se cierra aplicando o suprimiendo las señales pertinentes en las entradas digitales correspondientes.

Al mismo tiempo, la puerta SW también tiene que estar abierta.

Abrir y cerrar la puerta HW de forma controlada por nivel



Mientras esté activada la entrada digital, los impulsos de contaje pueden alcanzar el contador y se cuentan. Si se desactiva la entrada digital, se cerrará la puerta. Los impulsos de contaje ya no se cuentan, el contador se para.

Si la puerta se cierra por un rebase por exceso o por defecto, primero hay que desactivar la entrada digital y después reactivarla para volver a abrir la puerta.

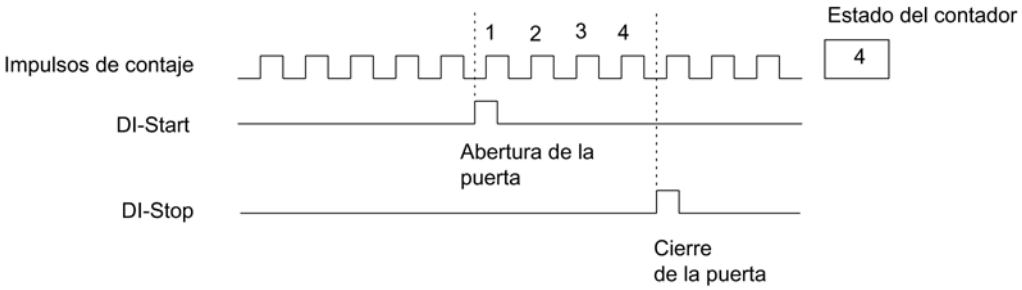
La puerta HW controlada por nivel actúa con el primer flanco ascendente en la entrada digital después de la parametrización.

ET 200S 1Count24V y 1Count5V:

El 1Count24V/1Count5V posee una puerta de hardware (puerta HW) que está controlada por la entrada digital del módulo. La puerta HW se parametriza como función de la entrada digital (función DI "Puerta HW"). Se abre cuando hay un flanco ascendente en la entrada digital y se cierra cuando hay un flanco descendente.

Abrir y cerrar la puerta HW de forma controlada por flanco

Abrir y cerrar la puerta HW de forma controlada por flanco sólo es posible en el FM x50-1.



Con la puerta controlada por flanco, la puerta HW se abre con un flanco ascendente en la entrada digital "DI-Start". La puerta se cierra con un flanco ascendente en la entrada digital "DI-Stop".

Si hubiera simultáneamente un flanco ascendente en ambas entradas entonces se cerrará la puerta si estaba abierta o bien permanecerá cerrada. si estaba cerrada. Si la entrada digital "DI-Stop" está activada, entonces un flanco ascendente en la entrada digital "DI-Start" no puede abrir la puerta.

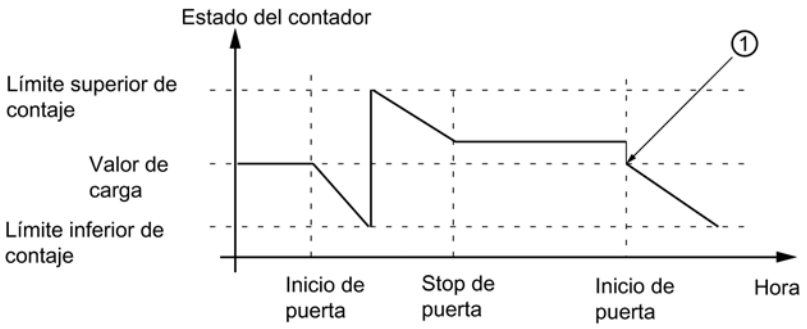
Función de puerta de cancelación e interrupción

Al parametrizar la función de puerta se puede especificar si la puerta debe cancelar o interrumpir el contaje.

Función de puerta de cancelación

En la función de puerta de cancelación el contaje comienza tras cerrar y reiniciar la puerta con el valor de carga (instante ① en la figura siguiente).

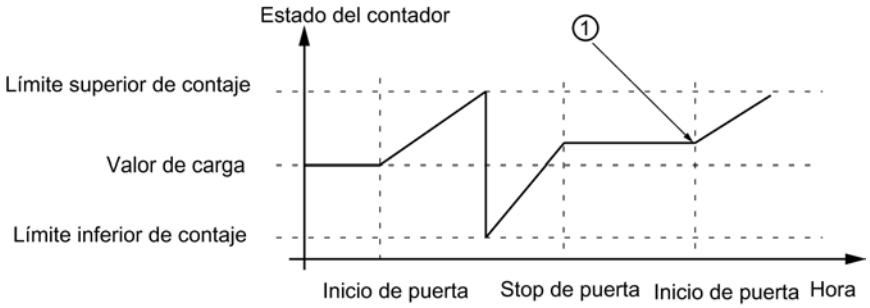
La figura siguiente muestra un ejemplo de contaje sin fin hacia atrás con control de puerta de cancelación.



Función de puerta de interrupción

En la función de puerta de interrupción el contaje continúa tras cerrar y reiniciar la puerta con el último valor de contaje actual (instante ① en la figura siguiente).

La figura siguiente muestra un ejemplo de contaje sin fin hacia adelante con control de puerta de interrupción.

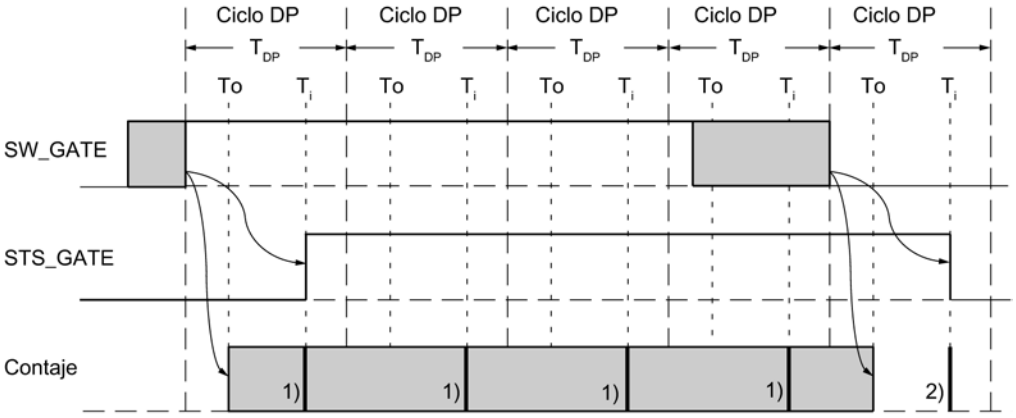


Control de puerta en modo isócrono

En el modo isócrono el control de puerta funciona de la siguiente manera:

Control de puerta con puerta SW

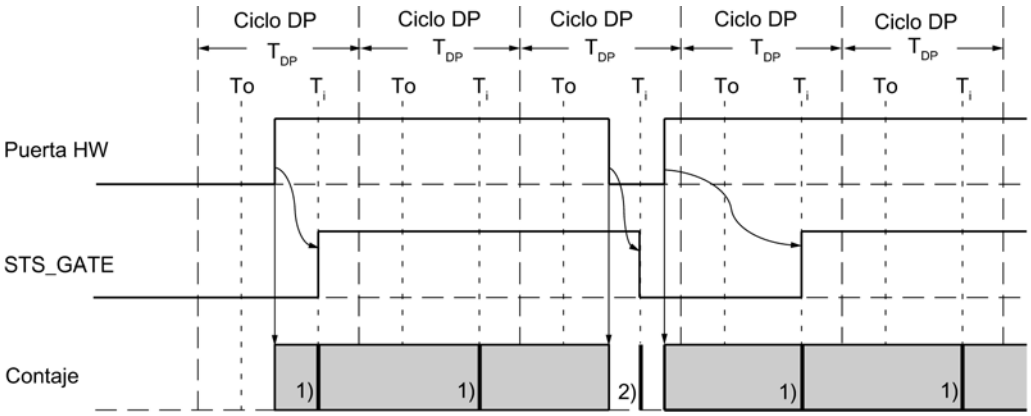
El control de la puerta SW se realiza activando y desactivando el bit de control SW_GATE. El contaje empieza y finaliza entonces en el instante T_o del siguiente ciclo PROFIBUS DP tras cambiar el bit de control:



1) Prepación del valor de contaje actual
2) Preparación del valor de contaje que estaba activo al finalizar el contaje

Control de puerta con puerta HW

En el control de la puerta HW el contaje empieza y termina inmediatamente al abrirse y cerrarse la puerta HW:



1) Prepación del valor de contaje actual
2) Preparación del valor de contaje que estaba activo al finalizar el contaje

Señales del encóder

Encoders compatibles

El módulo de contador puede procesar señales de contaje rectangulares generadas por encóders incrementales o incrementales de posición.

- Los **encóders incrementales** hacen un barrido de retícula y de este modo generan impulsos eléctricos rectangulares. Las diferencias residen en la amplitud del impulso y en la cantidad de las señales. Los encóders incrementales proporcionan dos pistas desfasadas en 90°.
- Los **encóders incrementales de posición**, como las fotocélulas o los detectores de proximidad (BERO), sólo proporcionan una señal rectangular con un nivel de tensión específico.

Señales diferenciales de 5 V

Señales de contaje de los encóders incrementales de 5 V

El encóder incremental de 5 V proporciona al módulo las siguientes señales diferenciales según RS 422:

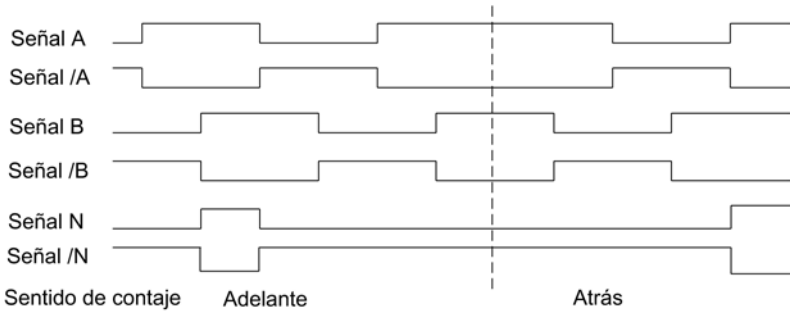
- A y /A
- B y /B
- N y /N

Las señales /A, /B y /N son las señales invertidas de A, B y N. Las señales A y B están desfasadas a 90° respectivamente.

En un encóder incremental de 5 V se utilizan las pistas A y B para contar. La pista N se utiliza para poner el contador en el valor de carga siempre y cuando haya sido parametrizado así.

Los encóders que suministran para cada señal la señal inversa se denominan encóders simétricos.

La siguiente figura muestra el cronograma de las señales de encóder de un encóder incremental de 5 V.



El módulo detecta el sentido de contaje por la relación entre las señales A y B.

Cambiar el sentido de contaje

Con el parámetro "Sentido de contaje normal" o "Sentido de contaje invertido" puede cambiar el sentido de contaje sin necesidad de cambiar el cableado.

Vigilancia de las señales del encóder

El módulo vigila si los cables están conectados y si hay una rotura de hilo o un cortocircuito.

Mediante la correspondiente parametrización puede definir cuáles de los tres pares de señales deben vigilarse. Por ello no deben conectarse señales que no se utilicen si en la parametrización se ha desactivado el diagnóstico del correspondiente par de señales (vigilancia).

Si se notifican fallos para las tres señales, puede que el encóder esté defectuoso, que la alimentación de encóder presente un cortocircuito o que no haya ningún encóder conectado.

Si el módulo detecta un fallo después de la parametrización, éste se registra en los registros de diagnóstico DS0 y DS1. Con la correspondiente parametrización, ese hecho puede provocar una alarma de diagnóstico.

Señales de 24 V

Señales de contaje de los encóders incrementales de 24 V

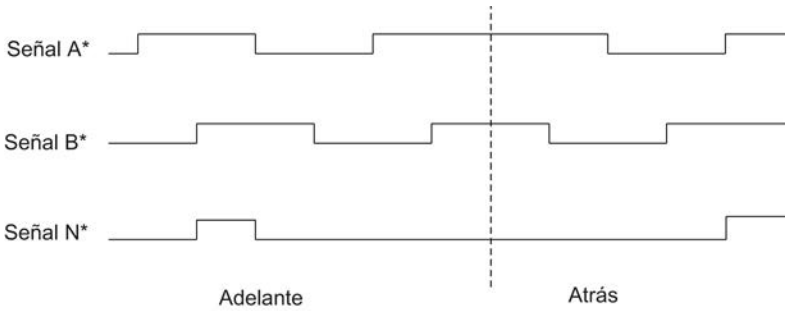
El encóder incremental de 24 V suministra al módulo las señales de 24 V A, B y N. Las señales A* y B* están desfasadas 90° respectivamente.

Las señales con una tensión de 24 V se identifican con un *.

En un encóder incremental de 24 V se utilizan las pistas A y B para contar. La pista N* se utiliza para poner el contador en el valor de carga siempre y cuando haya sido parametrizado así.

Los encóders que no suministran señales inversas se denominan encóders asimétricos.

La siguiente figura muestra el cronograma de las señales de encóder de un encóder incremental de 24 V.



El módulo detecta el sentido de contaje por la relación entre las señales A* y B*.

En las entradas de las señales de encóder de 24 V es posible parametrizar si se desean conectar al contador salidas tipo PNP o contracontactos o bien salidas tipo NPN. Encontrará más información al respecto en la descripción del encóder.

Modificar el sentido de contaje

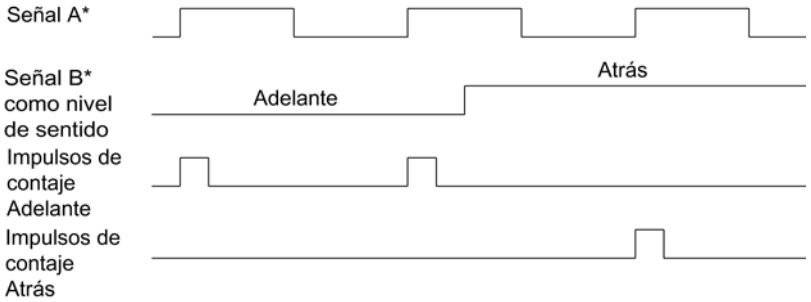
Con el parámetro "Sentido de contaje normal" o "Sentido de contaje inversa" puede cambiar el sentido de contaje sin necesidad de cambiar el cableado.

Señales de contaje de encóders de impulsos de 24 V sin/con nivel de sentido

El encóder, por ejemplo un detector de proximidad (BERO) o una barrera luminosa, sólo proporciona una señal de contaje, que debe conectarse a la conexión A* del conector frontal.

Adicionalmente puede conectar a la conexión B* del contador en cuestión una señal para detectar el sentido. Si el encóder no proporciona la señal correspondiente, también puede generar y conectar la correspondiente identificación dentro del programa de usuario o puede utilizar una señal de proceso equivalente.

La siguiente figura muestra el cronograma de las señales de encóder de un encóder incremental de posición de 24 V con nivel de sentido y los impulsos de contaje resultantes:



Vigilancia de las señales de encóder

Para las señales de contaje de 24 V no se vigila rotura de hilo ni cortocircuito.

Interpretación de la señal

Vista general

El contador del módulo de contador cuenta los flancos de las señales. Normalmente, se evalúa el flanco en A (A*) (evaluación individual). Para alcanzar mayor resolución, puede escoger una de las siguientes opciones:

- Evaluación simple (Página 9497)
- Doble evaluación (Página 9497)
- Evaluación cuádruple (Página 9497)

Sólo es posible una evaluación múltiple

- para encóders incrementales de 5 V con señales A y B desfasadas 90°
- para encóders incrementales de 24 V con señales A* y B* desfasadas 90°

Ajuste predeterminado

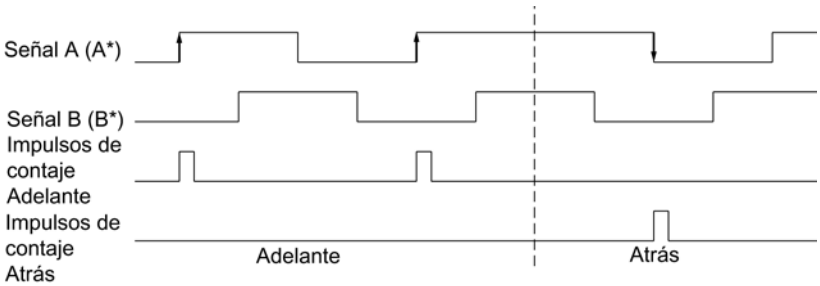
El ajuste predeterminado es "Evaluación simple".

Evaluación simple

Evaluación simple significa que sólo se evalúa un flanco de A.

- Los impulsos de contaje adelante se registran con un flanco ascendente en A y nivel bajo en B.
- Los impulsos de contaje atrás se registran con un flanco descendente de A y nivel bajo en B.

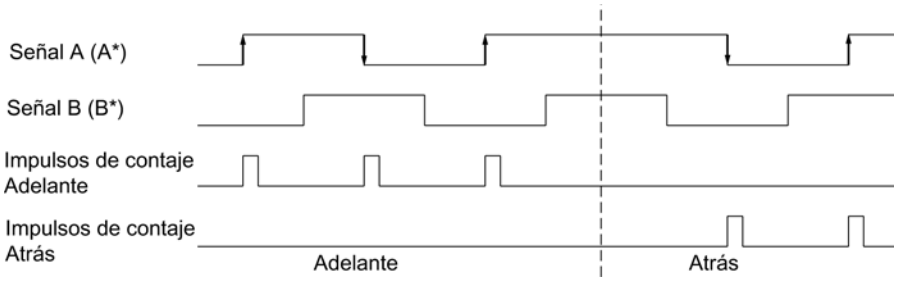
La figura siguiente muestra la evaluación simple de las señales.



Evaluación doble

Evaluación doble significa que se evalúan los flancos creciente y decreciente de la señal A. En función del nivel de la señal B, se generan impulsos de contaje adelante o atrás.

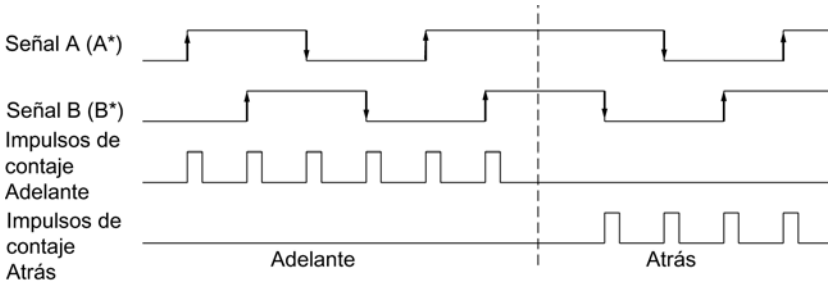
La figura siguiente muestra la evaluación doble de las señales.



Evaluación cuádruple

Evaluación cuádruple significa que se evalúan los flancos creciente y decreciente de A y B. En función de los niveles de las señales A y B, se generan impulsos de contaje adelante o atrás.

La figura siguiente muestra la evaluación cuádruple de las señales.



Utilizar el FM 350-1

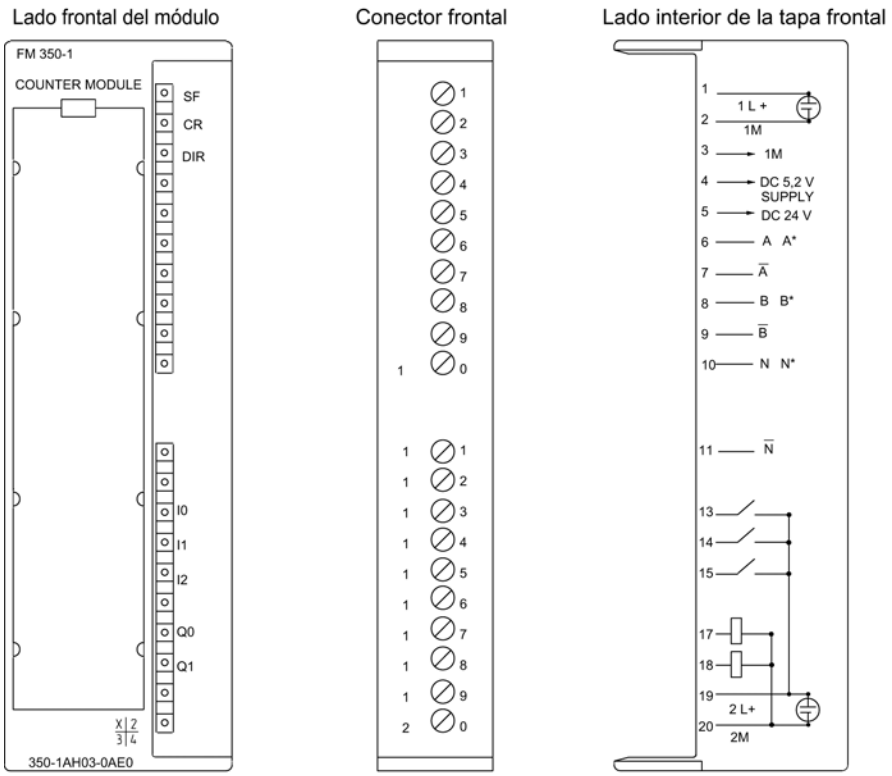
Asignación de pines del conector frontal del FM 350-1

Conector frontal

En el conector frontal de 20 pines se conectan:

- Señales de contaje
- Entradas digitales
- Salidas digitales
- Alimentación del encóder
- Tensión auxiliar y de carga

La figura siguiente muestra el lado frontal del módulo, el conector frontal y el lado interior de la tapa frontal con la asignación de terminales.



Asignación de terminales del conector frontal

La tabla siguiente muestra la asignación de terminales del conector frontal del FM 350-1.

Terminal	Nombre	Entrada/ salida	Función			
Tensión auxiliar						
1	1L+	Entrada	Tensión auxiliar de 24 V			
2	1M	Entrada	Tierra de la tensión auxiliar			
Alimentación del encóder y señales de contaje			**Encóder de 5 V RS 422, simétrico**	**Encóder de 24 V asimétrico**	**Generador de impulsos de 24 V con nivel de sentido**	**Detector de proximidad de 24 V**
3	1M	Salida	Tierra de la alimentación del encóder			
4	5,2 V DC	Salida	Alimentación de 5,2 V del encóder			
5	24 V DC	Salida	Alimentación de 24 V del encóder			
6	A A*	Entrada	Señal del encóder A	Señal del encóder A*		
7	/A	Entrada	Señal del encóder /A	—		
8	B B*	Entrada	Señal del encóder B	Señal del encóder B*	Señal de sentido	—
9	/B	Entrada	Señal del encóder /B	—		
10	N N*	Entrada	Señal del encóder N	Señal del encóder N*	—	
11	/N	Entrada	Señal del encóder / N	—		

Terminal	Nombre	Entrada/ salida	Función
12	—	—	—
Entradas y salidas digitales			
13	I0	Entrada	Entrada digital DI-Start
14	I1	Entrada	Entrada digital DI-Stop
15	I2	Entrada	Entrada digital DI-Set
16	—	—	—
17	Q0	Salida	Salida digital DO0
18	Q1	Salida	Salida digital DO1
Tensión de carga			
19	2L+	Entrada	Tensión de carga de 24 V
20	2M	Entrada	Tierra de la tensión de carga para las entradas y salidas digitales

Nota

Las conexiones para las entradas de contaje (alimentación del encóder, señales del encóder) están unidas galvánicamente con masa de la CPU. Por eso debe conectar el borne 2 (1M) con la masa de la CPU con una baja impedancia. Si falta esta conexión, podría ocurrir un fallo o una avería en el FM 350-1.

Si los encóders se alimentan externamente, también se deberá conectar la masa de dicha tensión externa con masa de la CPU.

Encontrará más información sobre la estructura y el cableado en el manual de producto (http://support.automation.siemens.com/WW/view/es/1086726) del módulo de contaje FM 350-1.

Tensión auxiliar 1L+/1M

Para la alimentación de tensión de los encóders de 5 V y 24 V debe conectar una tensión continua de 24 V en las conexiones 1L+ y 1M.

Un diodo integrado protege el módulo contra inversión de polaridad de la tensión auxiliar.

El módulo vigila si está conectada la tensión auxiliar.

Alimentación de encóder de 5,2 V DC

El módulo genera una tensión de 5,2 V a partir de la tensión auxiliar 1L+/1M con una intensidad máxima de 300 mA, disponible en el terminal DC5.2V para la alimentación de un encóder de 5 V a prueba de cortocircuitos. Se comprueba si hay cortocircuitos en la alimentación del encóder.

Alimentación de encóder de 24 V DC.

Para la alimentación de tensión de 24 V de un encóder, se pone a disposición la tensión 1L+/1M en la salida DC24V a prueba de cortocircuitos. Se comprueba si hay cortocircuitos en la alimentación del encóder.

Señales de encóder de 5 V A y /A, B y /B, N y /N

Al conector frontal se pueden conectar encóders incrementales con señales diferenciales de 5 V conforme a RS 422, es decir, encóders incrementales con las señales diferenciales A y /A, B y /B, N y /N.

Las señales A y /A, B y /B, N y /N se conectan a través de los terminales correspondientes.

Las señales N y /N sólo se conectan si se desea poner el contador a la marca cero del encóder.

Las entradas no disponen de aislamiento galvánico con el bus de la S7-300.

Señales de encóder de 24 V A*, B* y N*

Las señales de 24 V se identifican con las letras A*, B* y N*.

Se pueden conectar tres tipos de encóders diferentes, a saber:

- Encóders incrementales con señales de 24 V:
  Las señales A*, B* y N* se conectan a través de los terminales correspondientes.
- Generador de impulsos sin nivel de sentido:
  La señal se conecta al terminal A*.
- Generadores de impulsos con nivel de sentido:
  La señal de contaje se conecta al terminal A*. El nivel de sentido se conecta al terminal B*.

Las entradas no disponen de aislamiento galvánico con el bus de la S7-300.

Filtro de entrada para señales de encóder de 24 V

Para eliminar fallos, puede parametrizar filtros de entrada (componentes RC) con un tiempo de filtrado unitario para las entradas de 24 V A*, B* y N*. Se dispone de los dos filtros de entrada siguientes:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Frecuencia de contaje máxima	200 kHz	20 kHz
Ancho de impulso mínimo de las señales de contaje	2,5 µs	25 µs

Entradas digitales DI-Start, DI-Stop y DI-Set

Las entradas digitales DI-Start y DI-Stop se pueden utilizar para el control de puerta del contador. El control de puerta puede realizarse con control de nivel o de flanco.

La entrada digital DI-Set sirve para ajustar el contador al valor de carga.

Las entradas digitales funcionan con una tensión nominal de 24 V.

Las entradas digitales están separadas galvánicamente del bus posterior del S7-300 y de las entradas del contador.

Filtro de entrada para entradas digitales

Con objeto de suprimir interferencias, se puede parametrizar un filtro de entrada (elementos RC) con un tiempo de filtrado uniforme para las entradas digitales DI-Start, DI-Stop y DI-Set. Se dispone de los dos filtros de entrada siguientes:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Ancho de impulso mínimo de las señales de entrada	2,5 µs	25 µs

Salidas digitales DO0 y DO1

El FM 350-1 dispone de las salidas digitales DO0 y DO1 para disparar directamente procesos de control.

Las salidas digitales se alimentan con la tensión de carga 2L+.

Las salidas digitales están separadas galvánicamente del bus posterior del S7-300 y de las entradas del contador.

Las salidas digitales son salidas tipo PNP y pueden recibir una corriente de carga de 0,5 A. Están protegidas contra sobrecarga y cortocircuito.

Nota

Se pueden conectar directamente relés y contactores sin circuitería externa.

El comportamiento temporal de las salidas digitales varía en función de la parametrización.

Tensión de carga 2L+/2M

Para alimentar las salidas digitales DQ0 y DQ1, el módulo se debe abastecer con una tensión de carga de 24 V a través de los bornes 2L+ y 2M.

Un diodo integrado protege el módulo contra inversión de polaridad de la tensión de carga.

El FM 350-1 no vigila la tensión de carga 2L+/2M.

Configuración del FM 350-1

Parametrización del FM 350-1

Modos de operación

Seleccionar el modo de operación

Descripción

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el módulo de contaje.

Modo de operación	Descripción
Contaje sin fin (Página 9468) (con o sin puerta)	El FM 350-1 cuenta en modo sin fin a partir del estado actual del contador.
Contaje único (Página 9469) (con puerta HW o SW)	El FM 350-1 comienza a contar a partir del valor de carga al abrirse la puerta y se detiene en el límite de contaje.
Contaje periódico (Página 9470) (con puerta HW o SW)	El FM 350-1 comienza a contar al abrirse la puerta a partir del valor de carga entre los límites de contaje.
Medición de la frecuencia (Página 9478) (con puerta HW o SW)	El FM 350-1 cuenta los impulsos que llegan en un tiempo de medición dinámico.
Medición de la velocidad (Página 9480) (con puerta HW o SW)	El FM 350-1 cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de medición dinámico y calcula a partir de ello la velocidad con los impulsos por vuelta del encóder.
Medición del período (Página 9481) (con puerta HW o SW)	El FM 350-1 indica el tiempo de medición dinámico como duración del período. Si el período es inferior al tiempo de actualización, se crea un valor medio para el período.

El ajuste predeterminado es el modo de operación "Contaje sin fin".

Determinar el límite superior de contaje

Descripción

Si se parametriza un límite superior de contaje en combinación con un sentido principal de contaje (Página 9508) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño. El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El límite superior de contaje puede adoptar un valor > 1 hasta el límite superior del rango de contaje.

Definir el tiempo de medición

Descripción

Si se ha seleccionado un modo de medición, el módulo actualiza los valores medidos de forma cíclica. El tiempo de actualización es el producto del tiempo de medición x 10 ms. El tiempo de medición se predefine como factor n.

Si desea obtener una descripción detallada sobre el rango de valores para el tiempo de medición, consulte el apartado "Conceptos básicos sobre la operación de medición (Página 9474)".

Seleccionar el control de puerta

Descripción

Al preseleccionar un control de puerta se elige cómo se realizará el contaje en el modo de operación seleccionado.

- Sin puerta
  Todas las puertas están abiertas.
- Puerta SW (Página 9488)
  La puerta SW se abre y se cierra con el parámetro de entrada/bit de control SW_GATE.
- Puerta HW (Página 9488), con control de nivel o de flanco
  La puerta HW se abre y se cierra con las señales correspondientes en las entradas digitales
- Congelar (Página 9504)
  Con flancos en la entrada digital "DI-Start" se pueden almacenar (congelar) estados del contador.
- Congelar/redisparar (Página 9506)
  Con flancos en la entrada digital "DI-Start" se pueden almacenar (congelar) estados del contador. Después de la operación de guardado, el contador se establece al valor de carga.

El ajuste predeterminado es "Sin puerta".

Congelar

Descripción

Seleccionando el control de puerta "Congelar", es posible guardar (congelar) estados del contador con flancos en la entrada digital "DI-Start". Así, el estado del contador no cambia.

Requisitos

Para "congelar" es preciso utilizar la puerta SW (Página 9488) y seleccionar el flanco para congelar (Página 9509).

El intervalo mínimo de los flancos de congelación es de 1 ms. Si el intervalo entre dos flancos es menor, se pueden perder valores.

Funcionamiento

El estado del contador y el valor de congelación están ajustados a sus valores iniciales. El valor inicial es 0 con un modo de operación sin sentido principal de contaje y con sentido principal de contaje "Adelante". Con el sentido principal de contaje "Atrás", el valor inicial es el límite de contaje superior parametrizado.

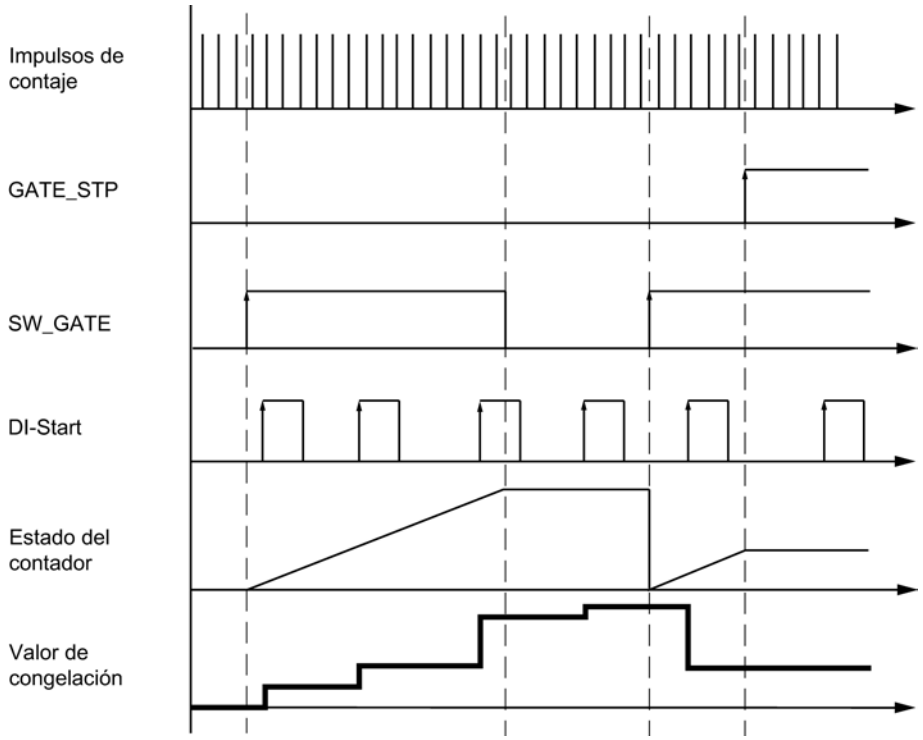
La función de contaje se inicia cuando se abre la puerta SW. El contador comienza con el valor de carga.

El valor de congelación siempre equivale al estado del contador en el instante en que aparece el flanco.

El estado de "DI-Start" se indica en el DB con el bit de estado STS_STA.

El valor de congelación se indica en el DB con LATCH_LOAD.

La figura siguiente muestra un ejemplo de "Congelar" con el valor de carga = 0 y un flanco ascendente en "DI-Start".



Interrupción y finalización

Si cierra la puerta SW reseteando SW_GATE, el efecto es cancelante.
Con flancos en "DI-Start" se puede guardar nuevamente el estado no modificado del contador incluso con la puerta SW cerrada.

Si la puerta SW se cierra con GATE_STP de la instrucción CNT_CTL1, se cancelará el contaje. En tal caso ya no se podrán almacenar estados del contador con "DI-Start".

Alarma de proceso al "congelar"

El almacenamiento de valores de contaje mediante "Congelar" también puede provocar una alarma de proceso. Esto puede hacer necesario un intervalo mayor entre los flancos. Si las alarmas llegan más rápido de lo que pueden ser acusadas por el sistema, se pierden alarmas de proceso. Esto se notifica mediante una alarma de diagnóstico.

Congelar/redisparar

Descripción

Seleccionando el control de puerta "Congelar/redisparar", es posible guardar (congelar) estados del contador con flancos en la entrada digital "DI-Start". Cada vez que se almacenan estados, el contador se inicializa con el valor de carga y sigue contando desde el valor de carga (redisparar).

Requisitos

Para "Congelar/redisparar" es preciso utilizar la puerta SW (Página 9488) y seleccionar el flanco para congelar (Página 9509).

El intervalo mínimo de los flancos de congelación es de 1 ms. Si el intervalo entre dos flancos es menor, se pueden perder valores.

Funcionamiento

Al abrirse la puerta SW se habilita la función de contaje.

El estado del contador y el valor de congelación están ajustados a sus valores iniciales. Permanecen invariables al abrirse la puerta SW. El valor inicial es 0 con un modo de operación sin sentido normal de contaje y con sentido normal de contaje "Adelante". En sentido principal de contaje "Atrás", el valor inicial es el límite de contaje superior parametrizado.

El contaje desde el valor de carga no comienza hasta que se produce el primer flanco en "DI-Start".

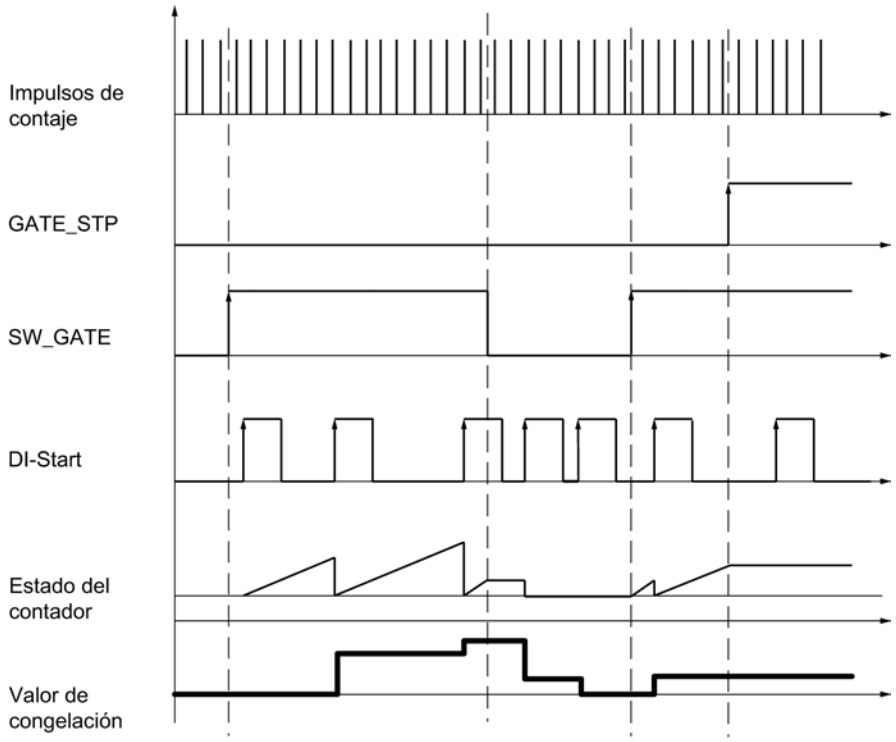
Con cada nuevo flanco en "DI-Start" se vuelve a cargar el valor de carga.

El valor de congelación siempre equivale al estado del contador en el instante en que aparece el flanco.

El estado de "DI-Start" se indica en el DB con el bit de estado STS_STA.

El valor de congelación se indica en el DB con LATCH_LOAD.

La siguiente figura muestra un ejemplo de "Congelar/redisparar" con valor de carga = 0 y flanco ascendente en "DI-Start".



Una aplicación especial del control de puerta "Congelar/redisparar" es la medición de tiempos entre dos flancos (Página 9519) con ayuda de la base de tiempo interna 1 MHz (Página 9512).

Interrupción y finalización

Si cierra la puerta SW reseteando SW_GATE, el efecto es de interrupción.
Con flancos en "DI-Start" se puede guardar nuevamente el estado no modificado del contador incluso con la puerta SW cerrada.

Si la puerta SW se cierra con GATE_STP de la instrucción CNT_CTL1, se cancelará el contaje. En tal caso ya no se podrán almacenar estados del contador con "DI-Start".

Alarma de proceso con "Congelar/redisparar"

El almacenamiento de valores de contaje mediante "Congelar/redisparar" también puede provocar una alarma de proceso. Esto puede hacer necesario un intervalo mayor entre los flancos. Si las alarmas llegan más rápido de lo que pueden ser acusadas por el sistema, se pierden alarmas de proceso. Esto se notifica mediante una alarma de diagnóstico.

Seleccionar el rango de contaje

Descripción

El módulo incorpora un registro de contaje de 32 bits. El ajuste “Rango de contaje” permite seleccionar si el módulo debe contar sólo en el rango positivo, o bien, si el bit 32 debe interpretarse como bit de signo, permitiendo de este modo la representación de números

negativos. El rango de contaje sólo se puede seleccionar si no se ha parametrizado un sentido de contaje principal.

Seleccione el rango de contaje deseado.

Encontrará más información en "Rangos de contaje y límites de contaje (Página 9465)".

Rebase del límite superior, rebase por defecto del límite inferior y paso por cero

En los límites de contaje, si se produce un rebase por exceso o por defecto se activa un bit en el DB de la instrucción CNT_CTL1.

Al producirse un paso por cero en el rango de contaje de -31 a +31 bits, también se activa un bit en el DB.

Al producirse un paso por cero en el rango de contaje de 0 a +32 bits se indica además un rebase por exceso, o bien un rebase por defecto (dependiendo del sentido de contaje).

Evento	Bit de estado en el DB
Rebase del límite superior	STS_OFLW se activa.
Rebase del límite inferior	STS_UFLW se activa.
Paso por cero	STS_ZERO se activa.

Disparo de alarmas de proceso

También puede notificar los eventos rebase por exceso, rebase por defecto y paso por cero con alarmas de proceso.

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Selección del sentido principal de contaje

Descripción

Si se parametriza un sentido principal de contaje (Página 9471) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño, ajustando para ello un límite superior de contaje (Página 9501). El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El sentido principal de contaje sólo puede seleccionarse en los modos de operación "Contaje único" y "Contaje periódico".

Se puede seleccionar el siguiente comportamiento:

- Ninguno (sin sentido principal de contaje)
- Adelante
- Atrás

Seleccionar flanco para congelar

Elección de los flancos

Es posible seleccionar el flanco con el que los estados del contador se deben almacenar (congelar) en la entrada digital "DI-Start". Es posible parametrizar el siguiente comportamiento:

- Congelar con flanco ascendente en "DI-Start"
- Congelar con flanco descendente en "DI-Start"
- Congelar con los dos flancos "DI-Start"

Seleccionar la resolución del período

Descripción

El valor del período medido se indica en 1 µs o 1/16 µs.

Seleccione la unidad deseada para la resolución.

Rangos de medición posibles con información de error

- Con una **resolución de 1 µs** el error absoluto es de 0,1 ‰ del valor medido.
- La siguiente tabla muestra los rangos de medida y errores para una **resolución de 1/16 µs**:

Período $T_{mín}$ ± error absoluto
1/16 µs × (160 ± 1)
1/16 µs × (1 600 ± 1)
1/16 µs × (16 000 ± 3)
1/16 µs × (160 000 ± 20)
A partir de un período de 100 ms el error absoluto es de 0,1 ‰ del valor medido.

Encóder

Seleccionar el tipo de señal del encóder

Seleccionar tipo de señal

1. Seleccione el tipo de señal del encóder utilizado.
   Dependiendo de su selección y del modo de operación se bloquearán aquellas opciones no permitidas para ese determinado encóder. Las opciones bloqueadas se identifican con letra gris en la correspondiente ventana de selección.
2. Seleccione las opciones que desee utilizar con el encóder seleccionado.

Es posible selecciona un tipo de señal de uno de los encóders siguientes:

- Encóder: incremental de 5 V (Página 9510) (señal diferencial de 5 V)
- Encóder: incremental de 24 V (Página 9511) (pista A+B de 24 V desfasada)
- Encóder: impulso y sentido de 24 V (Página 9511)
- Encóder: detector de proximidad de 24 V (Página 9512)
- Base de tiempo interna 1 MHz (Página 9512)

Encóder: incremental de 5 V

Encóder incremental de 5 V

Seleccione el tipo de señal "5 V Señal diferencial" si utiliza un encóder incremental de 5 V con señales diferenciales A, /A, B, /B y N, /N.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal (Página 9494)	Simple	Sólo se evalúa un flanco de señal A.
	Doble	Se evalúan ambos flancos de la señal A.
	Cuádruple	Se evalúan ambos flancos de las señales A y B.
Sentido de contaje (Página 9492)	Normal	Para adaptar el sentido de contaje al proceso
	Invertido	Para adaptar el sentido de contaje al proceso
Frecuencia máx. de contaje	–	La frecuencia de contaje máxima para este encóder es 500 kHz.
Entradas del encóder	–	Sin relevancia
Vigilancia (Página 9492)	A+B+N	Se vigila cortocircuito y rotura de hilo en todos los pares de señal.
	A+B	Se vigilan los pares de señal A y B.
	A	Sólo se vigila el par de señal A.
	Ninguna	No se vigila ningún par de señal.

Encóder: incremental de 24 V

Encóder incremental de 24 V

Seleccione el tipo de señal "24 V pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A*, B* y N*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal (Página 9494)	Simple	Sólo se evalúa un flanco de señal A.
	Doble	Se evalúan ambos flancos de la señal A.
	Cuádruple	Se evalúan ambos flancos de las señales A y B.
Sentido de contaje (Página 9493)	Normal	Para adaptar el sentido de contaje al proceso
	Invertido	Para adaptar el sentido de contaje al proceso
Frecuencia máx. de contaje (Página 9513)	200 kHz	En las entradas se aplica un filtro con un retardo de 1 µs.
	20 kHz	En las entradas se aplica un filtro con un retardo de 15 µs.
Entradas del encóder (Página 9512)	Salida tipo NPN	El encóder conmuta a 0 V (masa).
	Salida tipo PNP/ simétrica	El encóder conmuta a 24 V.
Vigilancia	–	Sin relevancia

Encóder: impulso y sentido de 24 V

Encóder incremental de posición de 24 V

Seleccione el tipo de señal "24 V Impulso y sentido" si desea utilizar un encóder incremental de posición de 24 V con la señal de contaje A* y la señal de sentido B*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal	–	Sin relevancia
Sentido de contaje	–	Sin relevancia
Frecuencia máx. de contaje (Página 9513)	200 kHz	En las entradas se aplica un filtro con un retardo de 1 µs.
	20 kHz	En las entradas se aplica un filtro con un retardo de 15 µs.
Entradas del encóder (Página 9512)	Salida tipo NPN	El encóder conmuta a 0 V (masa).
	Salida tipo PNP/ simétrica	El encóder conmuta a 24 V.
Vigilancia	–	Sin relevancia

Encóder: detector de proximidad de 24 V

Detector de proximidad de 24 V

Seleccione el tipo de señal "24 V Detector de proximidad" si desea utilizar un encóder incremental de posición de 24 V con la señal de contaje A*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal	–	Sin relevancia
Sentido de contaje	–	Sin relevancia
Frecuencia máx. de contaje (Página 9513)	200 kHz	En las entradas se aplica un filtro con un retardo de 1 µs.
	20 kHz	En las entradas se aplica un filtro con un retardo de 15 µs.
Entradas del encóder (Página 9512)	Salida tipo NPN	El encóder conmuta a 0 V (masa).
	Salida tipo PNP/ simétrica	El encóder conmuta a 24 V.
Vigilancia	–	Sin relevancia

Base de tiempo interna 1 MHz

Base de tiempo interna 1 MHz

Seleccione el tipo de señal "Base de tiempo interna 1 MHz" si, en lugar de un encóder externo, desea utilizar la base de tiempo interna de 1 MHz.

Esta selección permite medir p. ej. tiempos entre dos flancos inmediatamente consecutivos en la entrada digital "DI-Start".

Consulte también

Medir los tiempos entre dos flancos (Página 9519)

Parametrización de las entradas del encóder

En la parametrización ha seleccionado "24 V Impulso y sentido" en Tipo de señal (Página 9508).

Parametrización de las entradas del encóder

En la parametrización de las entradas del encóder el sentido de contaje se define tal como se indica a continuación. La tabla muestra el cambio del sentido de contaje en función de la parametrización.

Parametrización	Borne de conexión B*	Sentido de contaje
Salida tipo NPN	Sin conexión	Atrás
	Cortocircuitado a masa	Adelante
Salida tipo PNP/simétrica	Sin conexión	Adelante
	24 V conectado	Atrás

Con esta señales de contaje no es posible invertir el sentido mediante la inversión de la señal B*.

Nota

Si las señales de contaje son oscilantes, en este tipo de evaluación el valor de contaje puede "desaparecer" en el flanco, ya que todas las señales se suman.

Filtro de entrada para entradas de contaje de 24 V

Filtro de entrada para entradas de contaje de 24 V

Para eliminar fallos, puede parametrizar filtros de entrada con un tiempo de filtrado unitario para las entradas de contaje de 24 V A*, B* y N*. Existen los siguientes filtros de entrada:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Frecuencia de contaje máxima	200 kHz	20 kHz
Ancho de impulso mínimo de las señales de contaje	2,5 µs	25 µs

Introducir impulsos por vuelta del encóder

Impulsos por vuelta del encóder

Para el modo Medición de velocidad (Página 9480) es preciso parametrizar adicionalmente los impulsos por vuelta del encóder.

El valor para el número de impulsos por vuelta del encóder se encuentra en la placa de características o en los datos técnicos del encóder.

Entradas

Seleccionar señales de puerta

Control de la puerta HW

Para el control de la puerta HW se pueden seleccionar las siguientes opciones:

- Puerta de hardware controlada por nivel (con nivel en "DI-Start")
- Puerta de hardware controlada por flanco (con flanco en "DI-Start", "DI-Stop")

Encontrará más información en el apartado "Puerta de hardware (Página 9488)".

Seleccionar el ancho de impulso mínimo

Ancho de impulso mínimo

Para eliminar fallos, es posible parametrizar filtros de entrada (componentes RC) con un tiempo de filtrado unitario para las entradas digitales.

Se encuentran disponibles dos filtros de entrada con los siguientes datos:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Ancho de impulso mínimo de las señales de entrada	2,5 µs	25 µs

Inicializar contador (DI-Set)

Inicializar contador

Resumen

Si se desea iniciar el contaje a partir de un determinado valor (el valor de carga), se deberá parametrizar la señal con la que se vaya a poner el contador al valor de carga. Es posible inicializar el contador de la manera siguiente:

- con el parámetro de entrada L_DIRECT o L_PREPAR de la instrucción CNT_CTL1
- con una señal externa, bien sea a través de la entrada "DI-Set", o bien a través de "DI-Set" en combinación con la señal de paso por cero del encóder

Valor de carga

Como valor de carga se puede predeterminar un número cualquiera comprendido en el rango de contaje.

El valor de carga se interpreta conforme al rango de contaje seleccionado. Si, por ejemplo, se predetermina FFFF FFFF H como valor de carga, este valor se interpretará como 4.294.967.295 en el rango de contaje "0 a +32 bits", o bien como -1 en el rango de contaje "-31 a +31 bits".

El valor de carga se introduce en el DB de la instrucción CNT_CTL1 y se transfiere al módulo con la CNT_CTL1.

Los rangos de valores siguientes están permitidos para el valor de carga:

Rango de valores para valores de carga	Sentido principal de contaje		
	Ninguno	Adelante	Atrás
Límite inferior	Límite de contaje inferior máx.	$-2^{31} + 1$	2
Límite superior	Límite de contaje superior máx.	Límite de contaje superior parametrizado - 2	$2^{31} - 1$

Ajustar el contador mediante una señal externa

Es posible seleccionar dos señales externas distintas para ajustar el contador al valor de carga:

- Sólo "DI-Set" (Página 9516)
- "DI-Set" y señal de paso por cero (Página 9517) del encóder

La señal de paso por cero del encóder se utiliza para sincronizar el contador a un estado específico del mismo en un determinado punto del proceso. Gracias a ello se logra una mayor precisión en el contaje.

El contador se ajusta independientemente del modo de contaje.

Tras haber ajustado el contador mediante una señal externa, se activa el bit STS_SYNC en el DB.

Nota

La sincronización de un contador con la señal de paso por cero sólo tiene sentido si está abierta la puerta.

Si se ha habilitado sólo un sentido de contaje al ajustar el contador mediante una señal externa, tenga en cuenta que – cuando se cierre la puerta – se guardará (congelará) el sentido de contaje actual. Ello permite sincronizar el contador en el sentido de contaje contrario al sentido habilitado.

Inicializar el contador con "DI-Set"

Ajustar el contador con "DI-Set"

El valor de carga se puede cargar en el contador con un flanco ascendente en la entrada "DI-Set".

Las variables ENSET_UP y ENSET_DN en el DB de la instrucción CNT_CTL1 y la parametrización permiten definir el comportamiento del FM 350-1 ante un flanco ascendente en la entrada "DI-Set".

Parámetro	Comportamiento del FM 350-1
ENSET_UP activado	El contador se activa sólo al contar hacia delante.
ENSET_DN activado	El contador se activa sólo al contar atrás.
ENSET_UP y ENSET_DN activados	El contador se activa al contar hacia delante y hacia atrás.
Parametrización "Ajuste único del contador"	El contador sólo se ajusta al ocurrir el primer flanco ascendente en la entrada "DI-Set". Si desea volver a activar el contador, primero es preciso activar de nuevo ENSET_UP o ENSET_DN. Entonces, el contador se activará de nuevo con el siguiente flanco ascendente en la entrada "DI-Set".
Parametrización "Ajuste múltiple del contador"	El contador se activa con cada flanco ascendente de "DI-Set" mientras que estén activadas ENSET_UP y/o ENSET_DN.

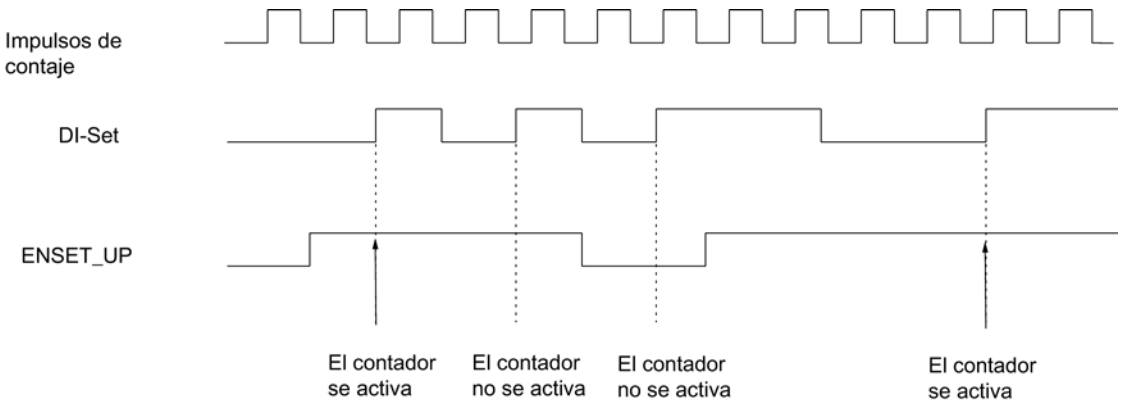
Nota

En todo caso, deberá activar una de las variables ENSET_UP y/o ENSET_DN para que el contador se pueda activar mediante la entrada digital "DI-Set".

Activación única del contador con "DI-Set"

La figura siguiente muestra una activación individual del contador con la entrada digital "DI-Set". En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se activa al contar hacia delante.

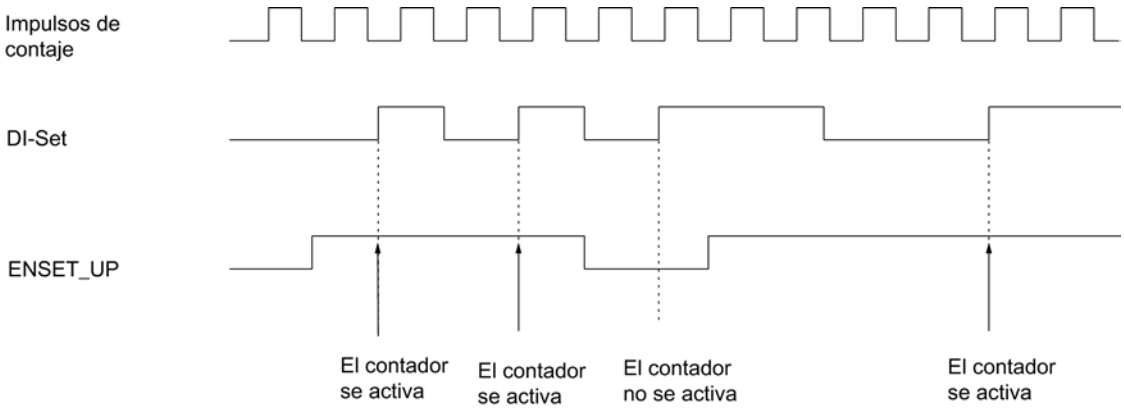
El contador se activa con el primer flanco ascendente que ocurra en la entrada digital "DI-Set", siempre y cuando esté activada ENSET_UP. Si se desea volver a activar el contador, en primer lugar es preciso desactivar la variable ENSET_UP y activarla luego de nuevo. El siguiente flanco ascendente en la entrada digital "DI-Set" implicará una nueva activación del contador.



Activación múltiple del contador con "DI-Set"

La figura siguiente muestra una activación múltiple del contador con la entrada digital "DI-Set". En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se activa al contar hacia delante.

El contador se activa en cada flanco ascendente que ocurra en "DI-Set", siempre y cuando esté activada ENSET_UP. Si desactiva ENSET_UP, el contador no se activará con "DI-Set". Sólo cuando se haya vuelto a activar ENSET_UP, el contador se activará en el siguiente flanco ascendente que ocurra en la entrada "DI-Set".



Inicializar el contador con "DI-Set" y la señal de paso por cero

Inicializar el contador con "DI-Set" y la señal de paso por cero

Si la inicialización del contador se parametriza con la señal de paso por cero del encóder ("Evaluar señal de paso por cero para el ajuste"), el contador se pone al valor de carga con el flanco ascendente de la señal de paso por cero.

La inicialización se realiza sólo si, en el instante del flanco ascendente de la señal de paso por cero, está activa además la entrada "DI-Set".

Las variables ENSET_UP y ENSET_DN en el DB de la instrucción CNT_CTL1 y la parametrización permiten definir el comportamiento del FM 350-1 ante un flanco ascendente de la señal de paso por cero:

Parámetros de entrada	Comportamiento del FM 350-1
ENSET_UP activado	El contador se pone al valor de inicialización sólo al contar hacia delante.
ENSET_DN activado	El contador se pone al valor de inicialización sólo al contar atrás.
ENSET_UP y ENSET_DN inicializados	El contador se inicializa al contar hacia delante y hacia atrás.
Parametrización "Ajuste único del contador"	El contador sólo se inicializa al ocurrir el primer flanco ascendente en la señal de paso por cero. Si desea volver a inicializar el contador, deberá inicializar primero de nuevo ENSET_UP o ENSET_DN (evaluación de flancos). Entonces, el contador se inicializará de nuevo con el siguiente flanco ascendente en la señal de paso por cero.
Parametrización "Inicialiazción repetida del contador"	El contador se inicializa en cada flanco ascendente de la señal de paso por cero mientras que estén activadas ENSET_UP y/o ENSET_DN.

Nota

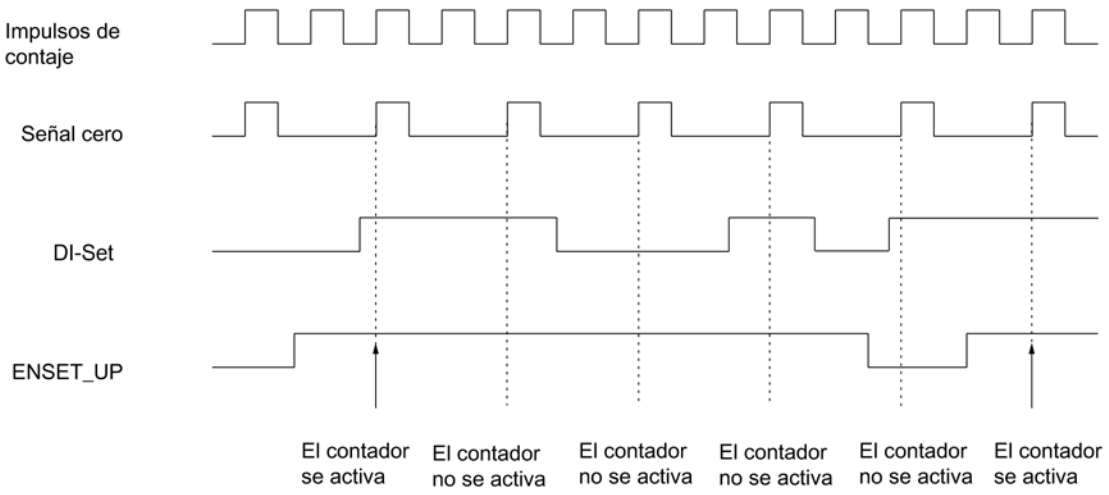
En todo caso, es preciso activar una de las variables ENSET_UP y/o ENSET_DN y la entrada "DI-Set" para que el contador se pueda inicializar con la señal de paso por cero.

Ajuste único del contador mediante "DI-Set" y la señal de paso por cero

La figura siguiente muestra una inicialización único del contador con la señal de paso por cero. En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se inicializa al contar hacia delante.

El contador se activa con el primer flanco ascendente de la señal de paso por cero, siempre y cuando estén activadas ENSET_UP y "DI-Set".

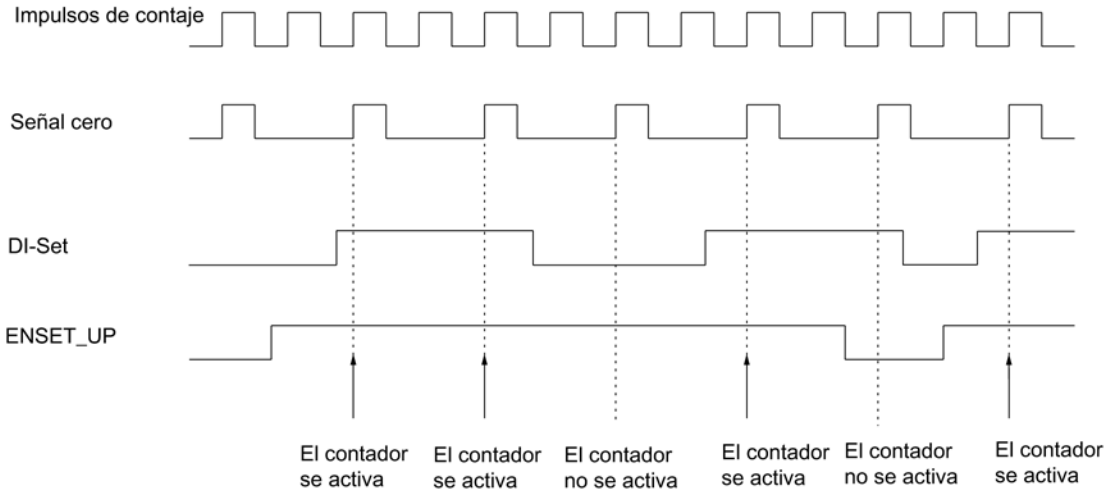
Si desea volver a activar el contador, es preciso desactivar primero la variable ENSET_UP y activarla luego de nuevo. Si "DI-Set" no está activada, la inicialización se efectúa con la primera señal de paso por cero tras activar "DI-Set". Si "DI-Set" está activada, la activación se efectúa con la siguiente señal de paso por cero.



Inicialización repetida del contador con "DI-Set" y la señal de paso por cero

La figura siguiente muestra una inicialización repetida del contador con la señal de paso por cero. En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se inicializa al contar hacia delante.

El contador se activa en cada flanco ascendente de la señal de paso por cero, siempre y cuando estén activadas ENSET_UP y "DI-Set".



Medir los tiempos entre dos flancos

Introducción

Es posible medir tiempos entre dos flancos inmediatamente consecutivos en la entrada digital "DI Start".

Requisito

Para la medición deben cumplirse los siguientes requisitos:

- No conecte ningún encóder al FM 350-1.
- Parametrice un modo de contaje (Página 9501) cualquiera como modo de operación.
- Parametrice el control de puerta "Congelar/redisparar" (Página 9502) como modo de operación.
- Para el encóder, parametrice el tipo de señal "Base de tiempo interna 1MHz" (Página 9508).

Elección de los flancos

Para medir el tiempo entre dos flancos inmediatamente consecutivos ...	... parametrice
Flancos ascendentes en "DI-Start"	Latch con flanco ascendente
Flancos descendentes en "DI-Start"	Latch con flanco descendente
Cualquier flanco en "DI-Start"	Latch con ambos flancos

Véase también: Seleccionar flanco para congelar (Página 9507)

Funcionamiento

El FM 350-1 utiliza una base de tiempo interna de 1 MHz para medir el tiempo. La medición del tiempo comienza al ocurrir el primer flanco en "DI-Start". Con cada flanco adicional en "DI-Start", en la interfaz de notificación se guarda siempre el tiempo transcurrido (en µs) desde el último flanco como valor de congelación LATCH_LOAD.

Salidas

Definir el comportamiento de las salidas para los modos de contaje

Valores de comparación y salidas

Descripción

En el FM 350-1 se pueden guardar dos valores de comparación (valor de comparación 1 y 2) para cada contador. Estos valores están asignados a las dos salidas digitales (valor de comparación 1: DO0, valor de comparación 2: DO1). Los valores de comparación se comparan con el estado de contaje actual. Cuando el estado de contaje alcanza un valor de comparación, puede activarse la salida correspondiente.

En "Conmutar en valores de comparación (Página 9526)" los dos valores de comparación se asignan al DO1.

Valores de comparación 1 y 2

Los dos valores de comparación se introducen en el DB de la instrucción CNT_CTL1 (CMP_V1, CMP_V2) y se transfieren al FM 350-1 activando los bits T_CMP_V1 o T_CMP_V2 respectivamente. El contaje no se ve afectado por este hecho.

Los valores de comparación deben estar dentro de los límites del rango de contaje en cuestión. El valor de comparación se interpreta conforme al rango de contaje seleccionado. Si, por ejemplo, se predetermina FFFF FFFF H como valor de comparación, este valor se interpretará como 4 294 967 295 en el rango de contaje de "0 a +32 bits", o bien como -1 en el rango de contaje de "-31 a +31 bits".

Para los valores de comparación se permiten los rangos de valores siguientes:

Rango de los valores de comparación	Sentido principal de contaje		
	Ninguno	Adelante	Atrás
Límite inferior	Límite de contaje inferior máx.	-2^{31}	1
Límite superior	Límite de contaje superior máx.	Límite de contaje superior parametrizado -1	$2^{31} - 1$

Habilitación de las salidas

Antes de poder controlar las salidas es preciso haberlas habilitado activando los bits correspondientes en el DB. Si desactiva uno de esos bits, la correspondiente salida se desactiva inmediatamente, aunque haya parametrizado para ella una duración de impulsos.

Para la habilitación de las salidas pueden activarse los siguientes bits:

Salida	...es habilitada por el bit
DO0	CTRL_DO0
DO1	CTRL_DO1

Activar y desactivar las salidas

Si se parametriza la opción "No se compara" para el comportamiento de una salida, las salidas habilitadas se pueden activar y desactivar con los bits correspondientes en el DB.

La activación y desactivación de las salidas se produce con los bits siguientes:

Salida	...se activa con	...se desactiva con
DO0	SET_DO0 = 1	SET_DO0 = 0
DO1	SET_DO1 = 1	SET_DO1 = 0

Comportamiento de las salidas en los modos de contaje

Definir el comportamiento de las salidas

Comportamiento de las salidas DO0 y DO1

Seleccione el comportamiento deseado de la salida correspondiente. Active para ello la casilla de verificación correspondiente:

- Valor sustitutivo "1" en caso de STOP de la CPU (Página 9523)
- No se compara (Página 9523)
- Activar si la lectura del contador >= valor de comparación (Página 9523)
- Activar si la lectura del contador <= valor de comparación (Página 9524)
- Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante (Página 9525)
- Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás (Página 9525)
- Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás (Página 9524)
- Conmutar en los valores de comparación (Página 9526) (sólo posible para la salida DO1)

Nota

Para activar las opciones, es preciso introducir los valores de comparación 1 y 2 en el DB con los datos de contaje, así como transferirlos al módulo con la instrucción CNT_CTL1; además, es preciso activar los bits CTRL_DO0 y CTRL_DO1 en el DB para habilitar las salidas. Véase también "Valores de comparación y salidas (Página 9518)".

Ajuste predeterminado de las salidas

En el ajuste predeterminado las salidas están desactivadas.

Comportamiento de las salidas digitales en modo isócrono

En modo isócrono, las salidas DO0 y DO1 conmutan inmediatamente después de cumplirse las condiciones de comparación. Por tanto, son independientes del ciclo PROFIBUS DP.

Excepción:

Si se ha parametrizado "No se compara" para el comportamiento de las salidas digitales y las salidas se accionan después de la habilitación con CTRL_DO0 o CTRL_DO1 con las señales de control SET_DO0 o SET_DO1, éstas se activan y desactivan en el instante T_o.

Valor sustitutivo "1" en caso de STOP de la CPU

Valor sustitutivo "1" en caso de STOP de la CPU

Si en los parámetros básicos del módulo ha seleccionado la opción "Valor sustitutivo" en "Reacción a STOP de la CPU" y la CPU se para, el módulo transmite a las salidas digitales los valores sustitutivos parametrizados. Active la casilla de verificación correspondiente a DO0; a continuación se activa la salida DO0 en el caso de parada de la CPU.

Los valores sustitutivos permanecen válidos tras pasar la CPU de STOP a RUN, hasta el siguiente manejo de las salidas digitales.

Si cambia el parámetro básico "Reacción a STOP de la CPU", las salidas se desactivan.

⚠ PRECAUCIÓN
Se pueden producir daños materiales. Los valores sustitutivos parametrizados se aplican siempre que la CPU pasa a STOP, aunque la puerta esté cerrada e incluso a salidas no habilitadas. Hay que parametrizar exclusivamente valores sustitutivos cuya aplicación no pueda ocasionar estados peligrosos.

No se compara

No se compara

La salida permanece desactivada y no se ve afectada por los eventos valor de comparación, paso por cero, rebase por exceso o por defecto. La salida DOx se puede utilizar puramente como salida digital. Si está habilitada, se puede activar y desactivar mediante el bit SET_DOx.

Activar si la lectura del contador >= valor de comparación

Activar si la lectura del contador >= valor de comparación

La salida se pone a "1" si el contador se encuentra en el rango comprendido entre el valor de comparación y el rebase por exceso.

La salida se pone a "1" si el contador se ajusta a un valor comprendido entre el valor de comparación y el rebase por exceso.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área.

Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9527).

Consulte también

Efecto de la histéresis (Página 9528)

Activar si la lectura del contador <= valor de comparación

Activar si la lectura del contador <= valor de comparación

La salida se pone a "1" si el contador se encuentra en el rango comprendido entre el valor de comparación y el rebase por defecto.

La salida se pone a "1" si el contador se ajusta a un valor comprendido entre el valor de comparación y el rebase por defecto.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área.

Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9527).

Consulte también

Efecto de la histéresis (Página 9528)

Activar al alcanzar el valor de comparación por la duración del impulso hacia delante/atrás

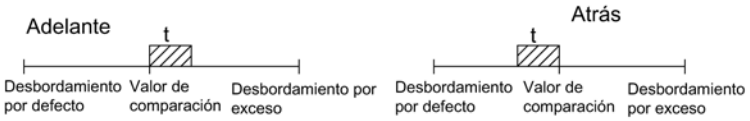
Activar al alcanzar el valor de comparación por la duración del impulso hacia delante/atrás

Mientras dura el impulso e independientemente del sentido de contaje, la salida se pone a "1" cuando se alcanza el valor de comparación.

Requisito:

- Sentido principal de contaje: "Ninguno"

La superficie sombreada del gráfico siguiente que la salida está activada en esta área (t = duración del impulso).



Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante

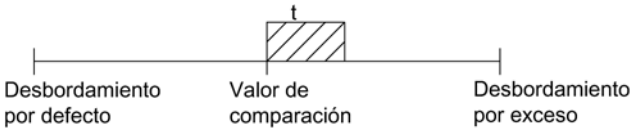
Activar al alcanzarse el valor de comparación por la duración del impulso hacia adelante

Mientras dura el impulso, la salida se pone a "1" cuando el contador alcanza el valor de comparación al contar hacia delante.

Requisito alternativo:

- Sentido principal de contaje: "Ninguno"
- Sentido principal de contaje: "Adelante"

La superficie sombreada del gráfico siguiente que la salida está activada en esta área (t = duración del impulso).



Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9527).

Consulte también

Efecto de la histéresis (Página 9528)

Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás

Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás

Mientras dura el impulso, la salida se pone a "1" cuando el contador alcanza el valor de comparación al contar hacia atrás.

Requisito alternativo:

- Sentido principal de contaje: "Ninguno"
- Sentido principal de contaje: "Hacia atrás"

La superficie sombreada del gráfico siguiente significa que la salida está activada en esta área (t = duración del impulso).

Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9527).

Consulte también

Efecto de la histéresis (Página 9528)

Conmutar en los valores de comparación

Conmutar en los valores de comparación

La salida DO1 conmuta en dos valores de comparación si se cumplen las condiciones siguientes:

- Para la salida DO0 se ha parametrizado el comportamiento "No se compara".
- Para la salida DO1 se ha parametrizado el comportamiento "Conmutar en valores de comparación".
- Los valores de comparación CMP_V1 y CMP_V2 se han cargado.
- Se ha habilitado la salida DO1 con CRTL_DO1.

La tabla siguiente muestra cuándo está activada o desactivada la salida DO1:

Valores de comparación CMP_V1 y CMP_V2	DO1 está conectada cuando	DO1 está desconectada cuando
CMP_V1 < CMP_V2	CMP_V1 ≤ Estado del contador ≤ CMP_V2	Estado del contador < CMP_V1 o Estado del contador > CMP_V2
CMP_V1 = CMP_V2	CMP_V1 = estado del contador = CMP_V2	CMP_V1 ≠ Estado del contador ≠ CMP_V2
CMP_V1 > CMP_V2	Estado del contador < CMP_V2 o Estado del contador > CMP_V1	CMP_V2 ≤ Estado del contador ≤ CMP_V1

El bit de estado STS_COMP2 indica el resultado de la comparación.

El bit de estado STS_COMP2 sólo se puede acusar y, por tanto, desactivar, cuando ya no se cumpla la condición de comparación.

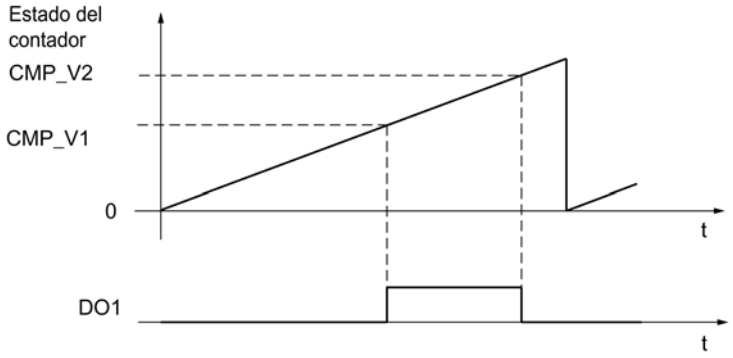
El estado de la salida DO1 se indica mediante el bit de estado STS_CMP2.

En este comportamiento de salida no hay histéresis.

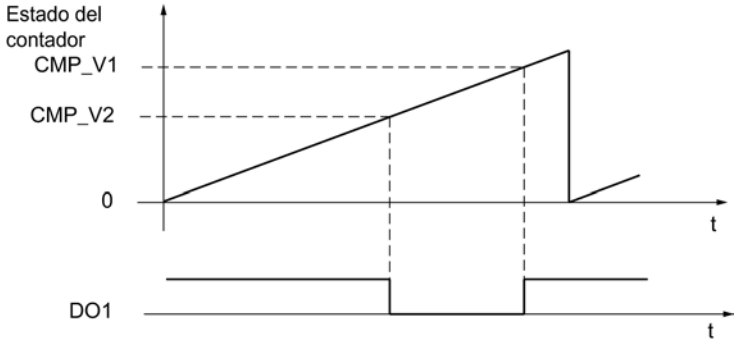
En este comportamiento de salida, el bit de control SET_DO1 no puede controlar la salida DO1.

Las siguientes figuras muestran el estado de la salida DO1 dependiendo del estado del contador y de los valores de comparación:

- Al iniciarse el contaje es CMP_V2 > CMP_V1



- Al iniciarse el contaje es CMP_V2 < CMP_V1



Condiciones marco para el comportamiento de las salidas

Condiciones marco para el comportamiento de las salidas digitales

Al parametrizar el comportamiento de las salidas digitales debe observar las siguientes condiciones marco.

Si...	Entonces...
... desea parametrizar una salida "Activar si el estado de contador >= valor de comparación o <= valor de comparación",	... deberá asegurarse de que el intervalo de tiempo entre estos dos eventos sea mayor que el tiempo de conmutación mínimo de las salidas (tiempo de conmutación: 300 µs); de lo contrario, se perderán los impulsos de control de las salidas. Si el estado del contador alcanza nuevamente el valor de comparación mientras la salida sigue activa, no se disparará un nuevo impulso. No se podrá emitir un nuevo impulso hasta que la salida deje de estar activada.
... desea parametrizar una salida "Activar si el estado de contador >= valor de comparación",	... no habilite ninguna alarma de proceso al "Alcanzar el valor de comparación correspondiente hacia delante o hacia atrás".
... desea parametrizar una salida "Activar si el estado de contador <= valor de comparación",	... no habilite ninguna alarma de proceso al "Alcanzar el valor de comparación correspondiente hacia delante o hacia atrás".

Si...	Entonces...
... desea parametrizar una salida "Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante"	... no habilite una alarma de proceso al "Alcanzar el valor de comparación correspondiente hacia atrás".
... desea parametrizar una salida "Activar al alcanzarse el valor de comparación por la duración del impulso en sentido descendente"	... no podrá habilitar una alarma de proceso al "Alcanzar el valor de comparación correspondiente en sentido ascendente".

Efecto de la histéresis

Funcionamiento de la histéresis con la parametrización "Activar si el valor del contador >= valor de comparación o <= valor de comparación"

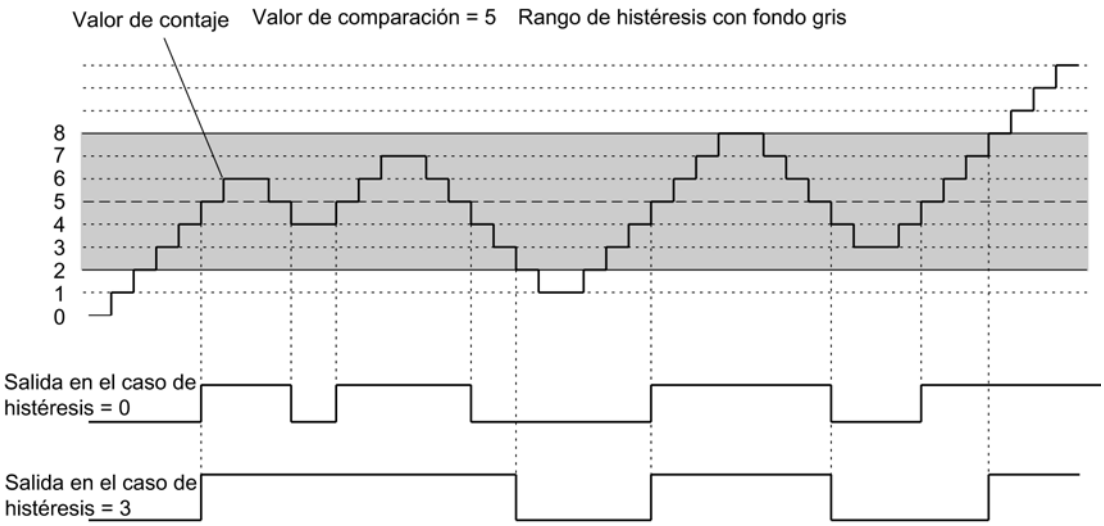
La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (= desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo el valor de comparación = 5.

La parametrización en este ejemplo es:

- Sentido principal de contaje: "Adelante"
- Comportamiento de la salida: "Activar si el estado del contador >= valor de comparación"

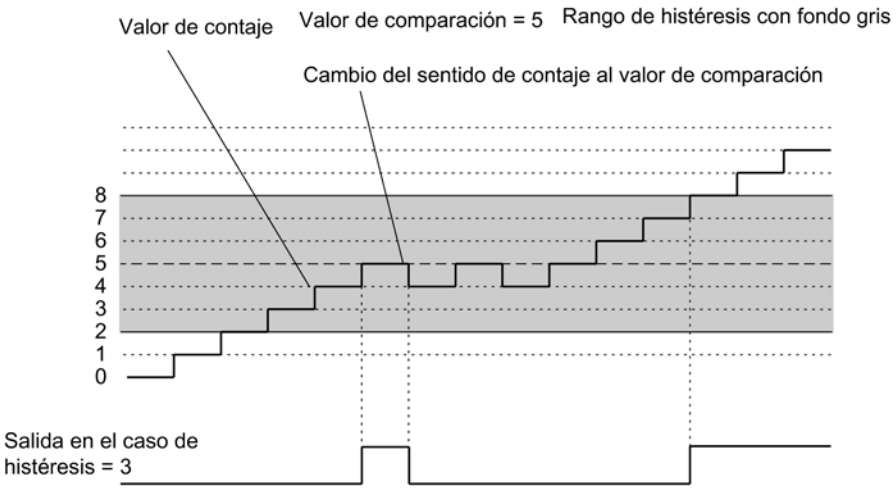
Al alcanzarse la condición de comparación (estado del contador = 5) se activa la histéresis. Con la histéresis activada el resultado de la comparación permanece invariable.

Cuando el valor de contaje abandona el rango de histéresis, en el ejemplo con el valor de contaje 2 u 8, la histéresis se desactiva. El comparador vuelve a conmutar según sus condiciones de comparación, en el ejemplo con el valor de contaje 5.



Nota

Si el estado del contador es igual al valor de comparación y la histéresis está activa, el FM 350-1 desactiva la salida cuando se invierte el sentido de contaje en el valor de comparación (véase la figura siguiente).



Funcionamiento de la histéresis con la parametrización "Activar al alcanzarse el valor de comparación en sentido ascendente/descendente por la duración del impulso"

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (= desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo el valor de comparación = 5.

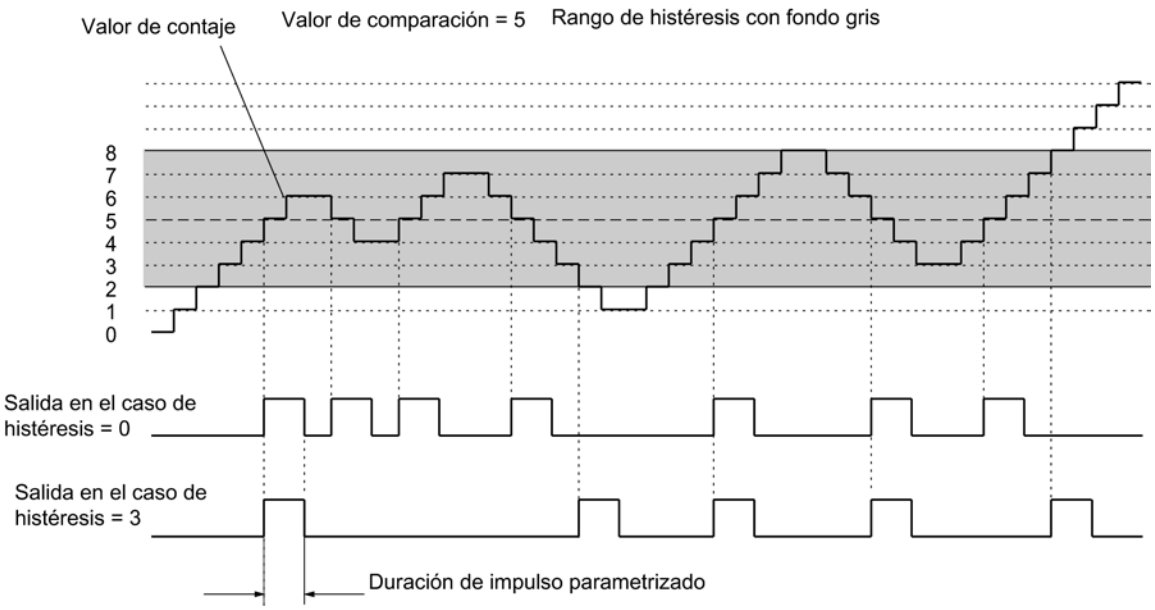
La parametrización en este ejemplo es:

- Sentido principal de contaje: "Ninguno"
- Comportamiento de la salida: "Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante"
- Duración del impulso > 0

Al alcanzarse la condición de comparación (estado del contador = 5) se activa la histéresis y se emite un impulso por la salida con la duración parametrizada.

Si el valor de contaje abandona el rango de histéresis, la histéresis se desactiva.

Cuando se activa la histéresis, el FM 350-1 almacena el sentido de contaje. Si se abandona el rango de histéresis en sentido contrario al almacenado, se emite un impulso.



Definir el comportamiento de las salidas para los modos de medición

Valores límite y salidas

Valores límite inferior y superior

Es posible determinar un valor límite superior y un valor límite inferior para medir la frecuencia, la velocidad o el período. Si se infringen dichos límites, se activará la salida digital DO0. Estos valores límite se pueden parametrizar y modificar a través de la interfaz de control.

Los siguientes rangos de valores están permitidos para los valores límite inferior y superior:

Modo de medición	Valor límite inferior (LI):	Valor límite superior
Medición de frecuencia (encóder de 5 V)	0 a 499 999 999 × 10^{-3} Hz	LI+1 a 500 000 000 × 10^{-3} Hz
Medición de frecuencia (encóder de 24 V)	0 a 199 999 999 × 10^{-3} Hz	LI+1 a 200 000 000 × 10^{-3} Hz
Medición de velocidad	0 a 24 999 999 × 10^{-3} /min	LI+1 a 25 000 000 × 10^{-3}/min
Medición del período (resolución 1 µs)	0 a 119 999 999 µs	LI+1 a 120 000 000 µs
Medición del período (resolución 1/16 µs)	0 a 1 919 999 999 µs	LI+1 a 1 920 000 000 µs

Habilitación de las salidas

Antes de poder controlar las salidas es preciso haberlas habilitado activando los bits correspondientes en el DB. Si se desactiva uno de estos bits, la salida correspondiente se desconectará de inmediato.

Salida	...es habilitada por el bit
DO0	CTRL_DO0
DO1	CTRL_DO1

Comportamiento de las salidas en los modos de medición

Comportamiento de las salidas digitales

Salida digital DO0

Para la salida digital DO0 se puede programar una de 4 respuestas posibles cuando se alcancen los valores límite. En la tabla siguiente puede ver las diferentes posibilidades.

Parametrización de la salida digital DO0	Determinar el comportamiento de la salida digital DO0	Instante de conmutación	
		Modo isócrono	Modo no isócrono
No se compara	No afectado por la vigilancia de valor límite. Si la salida DO0 ya se ha activado, se puede desactivar modificando el parámetro a "No se compara". La salida DO0 se puede utilizar libremente como salida digital. Además, es posible activarla y desactivarla con la señal de control SET_DO0 si se ha habilitado con la señal de control CTRL_DO0.	En el instante T_o	Inmediatamente después de activar o desactivar la salida
Fuera de los valores límite	DO0 se activa en los dos casos siguientes: • Valor medido < valor límite inferior • Valor medido > valor límite superior	Al final del tiempo de actualización en el instante T_i	Al final del tiempo de actualización
Rebase por defecto del valor límite inferior	DO0 se activa si • Valor medido < valor límite inferior		
Rebase por exceso del valor límite superior	DO0 se activa si • Valor medido > valor límite superior		

Seleccione el comportamiento deseado de la salida DO0. Active para ello la casilla de verificación correspondiente.

Salida digital DO1

Una vez habilitada, la salida DO1 se puede utilizar libremente como salida digital, siendo posible activarla y desactivarla con la señal de control SET_DO1.

DO1 no se ve afectada por la vigilancia de valor límite.

En modo isócrono, DO1 conmuta al final del tiempo de actualización en el instante T_o.

Definir la duración del impulso

Duración del impulso

Es posible definir una duración del impulso para adaptarse a los actuadores utilizados en el proceso (contactores, elementos finales de regulación, etc.). La duración del impulso indica cuánto tiempo están activas las salidas DO0 y DO1 al alcanzarse un valor de comparación.

Si se ha parametrizado el "sentido principal de contaje adelante" o "atrás", la duración del impulso sólo tendrá efecto en el sentido principal de contaje.

Si no se ha parametrizado un sentido principal de contaje, la duración del impulso puede tener efecto en ambos sentidos.

La duración del impulso comienza cuando se activa la salida. La imprecisión de la duración del impulso es inferior a 1 ms.

Para la duración del impulso se puede predefinir un valor comprendido entre 0 y 500 ms. Este valor es válido para ambas salidas.

Si la duración del impulso = 0 ms, la salida se activará al alcanzarse el valor de comparación y se desactivará con el siguiente impulso de contaje.

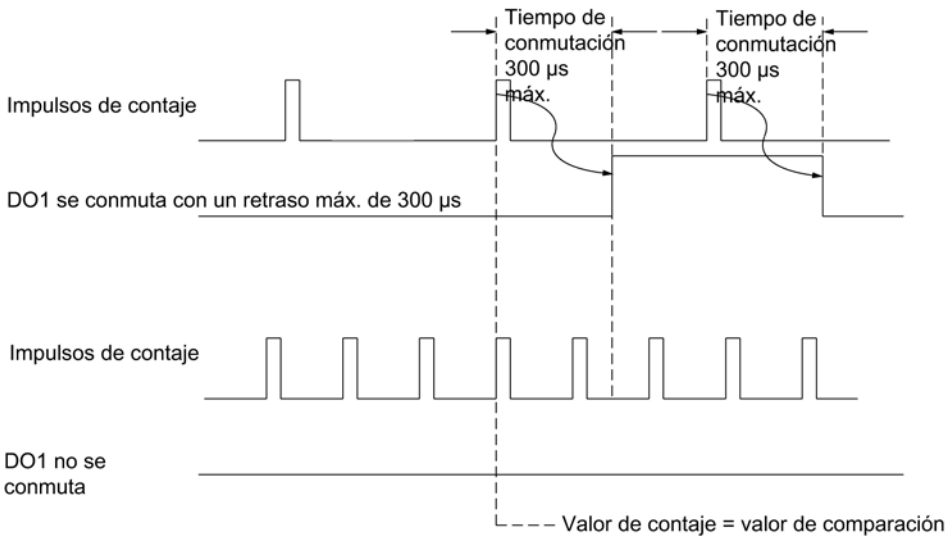
El valor predeterminado para la duración de impulsos es 0.

Nota

Si se predetermina una duración del impulso = 0 ms, la salida permanecerá activada hasta que el estado del contador sea igual al valor de comparación (véase la mitad superior de la figura siguiente).

Debido a los tiempos de conmutación de las salidas digitales (máx. 300 µs) se podrían perder impulsos de control en las salidas si el intervalo de tiempo entre los impulsos de contaje es menor que los tiempos de conmutación de las salidas digitales (véase la mitad inferior de la figura siguiente).

Por tanto, asegúrese de que el intervalo de tiempo entre los impulsos de contaje sea mayor que los tiempos de conmutación de las salidas digitales.



Definir histéresis

Funcionamiento básico de la histéresis

Un encóder puede pararse en una posición determinada y "oscilar" alrededor de dicha posición. Esto hace que el estado del contador fluctúe alrededor de un valor determinado. Si hay un valor de comparación en este rango de fluctuación, por ejemplo, la salida correspondiente se activa y desactiva de acuerdo con el ritmo de las fluctuaciones. Para evitar estas conmutaciones cuando se producen pequeñas oscilaciones, el FM 350-1 cuenta con una histéresis parametrizable.

Para la histéresis se pueden parametrizar valores comprendidos entre 0 y 255.

Dependiendo del valor parametrizado, la histéresis actúa tal como se indica a continuación:

Histéresis	Efecto
Valor de histéresis n = 0, 1	La histéresis no tiene efecto (desactivada). La salida reaccionará al más mínimo cambio del estado del contador.
2 ≤ valor de histéresis n ≤ 255	La histéresis tiene efecto. La salida reacciona sólo cuando el estado del contador se ha distanciado n unidades del valor de comparación.

La histéresis también actúa en caso de rebase por exceso y por defecto.

Habilitar alarmas de proceso

Habilitar la activación de una alarma de proceso

Introducción

En el FM 350-1 se puede definir qué eventos deben generar una alarma de proceso durante el funcionamiento.

¿Qué es una alarma de proceso?

Si se debe reaccionar a un evento determinado, independientemente del ciclo de la CPU, el FM 350-1 puede disparar una alarma de proceso. En ese caso la CPU interrumpe el programa cíclico y procesa el OB de alarma de proceso.

Habilitar la alarma de proceso

La habilitación de las alarmas de proceso tiene lugar en dos pasos:

1. En la configuración del hardware las alarmas del módulo se habilitan en los parámetros básicos y se especifica si el módulo debe activar una alarma de diagnóstico y/o una alarma de proceso.
2. En "Habilitar alarma de proceso" es preciso seleccionar los eventos correspondientes:
   - Eventos para la activación de una alarma de proceso en los modos de contaje (Página 9535)
   - Eventos para la activación de una alarma de proceso en los modos de medición (Página 9535)

OB de alarma de proceso, OB 40

Si ocurre una alarma de proceso, el programa de usuario se interrumpe, los datos se transfieren del módulo a la información de arranque del OB 40 (Página 9535) y se llama al OB 40. La alarma de proceso se acusa saliendo del OB 40.

La CPU pasa a modo STOP si no se ha programado el OB 40. Si a continuación se pasa a RUN, se borran las condiciones para la alarma de proceso.

Alarma de proceso perdida

Si ocurre un evento que deba disparar una alarma de proceso, y el mismo evento anterior no se ha acusado todavía, no se dispararán más alarmas de proceso; la alarma de proceso se perderá. Esta situación puede conducir a la alarma de diagnóstico "Alarma de proceso perdida".

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de proceso.

Eventos para la activación de una alarma de proceso en los modos de contaje

¿Qué eventos pueden disparar una alarma de proceso en los modos de contaje?

En los modos de contaje, los eventos siguientes pueden disparar una alarma de proceso mientras el FM 350-1 está en funcionamiento:

- Apertura de la puerta (en los modos de operación con puerta HW o SW)
- Cierre de la puerta (en los modos de operación con puerta HW o SW)
- Rebase del límite superior
- Rebase del límite inferior
- Paso por cero
- Alcance del valor de comparación 1 ó 2 en sentido de contaje adelante
- Alcance del valor de comparación 1 ó 2 en sentido de contaje atrás
- Activar contador (DI-Set)
- Congelar

Active la casilla de verificación para los eventos correspondientes.

Es posible seleccionar un número cualquiera de eventos para disparar alarmas de proceso. Para las alarmas de proceso que se disparan al alcanzar un valor de comparación hay que tener en cuenta las condiciones marco para el comportamiento de las salidas (Página 9525).

Eventos para la activación de una alarma de proceso en los modos de medición

¿Qué eventos pueden disparar una alarma de proceso en los modos de medición?

En los modos de medición, los eventos siguientes pueden disparar una alarma de proceso mientras el FM 350-1 está en funcionamiento:

- Apertura de la puerta (en los modos de operación con puerta HW o SW)
- Cierre de la puerta (en los modos de operación con puerta HW o SW)
- Rebase por defecto del valor límite inferior
- Rebase por exceso del valor límite superior
- Fin de la medición

Active la casilla de verificación para los eventos correspondientes.

Es posible seleccionar un número cualquiera de eventos para disparar alarmas de proceso.

Información de arranque del OB 40

Información de arranque del OB 40

La variable temporal OB40_POINT_ADDR se escribe en la información de arranque del OB 40.

La variable OB40_POINT_ADDR (byte 8 a 11) abarca 4 bytes. En los bytes 8 y 9 se introducen las informaciones acerca del evento que ha disparado la alarma de proceso.

La tabla muestra qué bits se activan según el tipo de alarma. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Byte	Bit	Significado: Alarma en caso de ...
8	0	Apertura de la puerta
	1	Cierre de la puerta
	2	Rebase del límite superior (en el modo de contaje)
		Valor medido fuera de los valores límite (en modo de medición)
	3	Rebase por defecto del límite inferior (en el modo de contaje)
		Fin de medición (en modo de medición)
	4	Alcance del valor de comparación 1 en sentido de contaje adelante
	5	Alcance del valor de comparación 1 en sentido de contaje atrás
	6	Alcance del valor de comparación 2 en sentido de contaje adelante
	7	Alcance del valor de comparación 2 en sentido de contaje atrás
9	0	Paso por cero
	5	Activar contador (DI-Set) (sincronización)
	7	Congelar

Interfaz de control y realimentación del FM 350-1

Resumen

Si se desea utilizar el FM 350-1 sin instrucciones, el FM 350-1 se controla y se observa directamente a través de la interfaz de control y realimentación (interfaz de datos de usuario).

La interfaz de datos de usuario comienza en la dirección inicial del módulo y tiene una longitud de 16 bytes.

Con los comandos de carga se puede leer la interfaz de realimentación.

Los comandos de transferencia permiten escribir en la interfaz de aviso de control.

No se permite un modo mixto con comandos de carga/transferencia y programación con instrucciones.

Interfaz de control de los modos de contaje

Interfaz de control de los modos de contaje

El parámetro LOAD_VAL (byte 0 a 3) tiene dos significados:

- Si se activa el bit L_DIRECT o L_PREPAR, LOAD_VAL se interpreta como valor de carga.
- Si se activa el bit C_DOPARA, en el byte 0 se puede definir el comportamiento de las salidas DO0 y DO1; los bytes 1 y 2 se interpretan como histéresis y duración del impulso.

La siguiente tabla muestra la asignación de la interfaz de control (las salidas) del FM 350-1 para los modos de contaje.

Offset de la dirección inicial	Parámetro	Significado				
Byte 0 a 3	LOAD_VAL	Valor de carga; carga directa y anticipativa con el bit L_DIRECT Valor de carga; carga anticipativa con el bit L_PREPAR				
Byte 0		Comportamiento de las salidas DO0 y DO1, de la histéresis y de la duración de impulso; definir con el bit C_DOPARA				
		Bit 3	**Bit 2**	**Bit 1**	**Bit 0**	**Comportamiento de la salida DO0**
		x	0	0	0	No se compara
		x	0	0	1	Activar si la lectura del contador >= valor de comparación
		x	0	1	0	Activar si la lectura del contador <= valor de comparación
		x	0	1	1	Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás
		x	1	0	0	Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante
		x	1	0	1	Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás
		Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Comportamiento de la salida DO1**
		x	0	0	0	No se compara
		x	0	0	1	Activar si la lectura del contador >= valor de comparación
		x	0	1	0	Activar si la lectura del contador <= valor de comparación
		x	0	1	1	Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás
		x	1	0	0	Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante
		x	1	0	1	Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás
		x	1	1	0	Conmutar en valores de comparación
		x = irrelevante				
Byte 1		Histéresis (rango de valores 0 a 255)				
Byte 2		Duración del impulso (rango de valores 0 a 250)				
Byte 3		Reserva = 0				
Bytes 4 a 7	CMP_V1	Valor de comparación 1; cargar con el bit T_CMP_V1				
Byte 8 a 11	CMP_V2	Valor de comparación 2; cargar con el bit T_CMP_V2				

Offset de la dirección inicial	Parámetro	Significado
Byte 12	– NEUSTQ – – OT_ERR_A – – –	Bit 7: Reserva = 0 Bit 6: Acuse de rearranque completo Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: Acuse de error de operador Bit 2: reserva = 0 Bit 1: reserva = 0 Bit 0: reserva = 0
Byte 13	– – – – SW_GATE GATE_STP ENSET_DN ENSET_UP	Bit 7: reserva = 0 Bit 6: reserva = 0 Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: Bit de control puerta SW Bit 2: Parada general de puerta Bit 1: Habilitación sincronización descendente Bit 0: Habilitación sincronización ascendente
Byte 14	– – – – SET_DO1 SET_DO0 CTRL_DO1 CTRL_DO0	Bit 7: reserva = 0 Bit 6: reserva = 0 Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: Bit de control DO1 Bit 2: Bit de control DO0 Bit 1: Habilitación DO1 Bit 0: Habilitación DO0
Byte 15	– C_DOPARA[1)] RES_ZERO RES_SYNC T_CMP_V2[2)] T_CMP_V1[2)] L_PREPAR[2)] L_DIRECT[2)]	Bit 7: reserva = 0 Bit 6: Función DO0/DO1, modificar histéresis o duración del impulso Bit 5: Desactivar bits de estado de paso por cero, rebase por exceso, rebase por defecto y comparador Bit 4: Desactivar bit de estado para sincronización Bit 3: Cargar valor de comparación 2 Bit 2: Cargar valor de comparación 1 Bit 1: Carga anticipativa del contador Bit 0: Carga directa y anticipativa del contador
[1)] Este bit no se puede activar al mismo tiempo que el bit 0, 1, 2 ó 3 del byte 15. [2)] Este bit no se puede activar al mismo tiempo que el bit 6 del byte 15.		

Aclaraciones sobre los bits de control de los modos de contaje

Bit de control	Aclaraciones
C_DOPARA	Con este bit se modifica la función y el comportamiento de DO0 y DO1, la histéresis y la duración del impulso. Los valores del byte 0 a 2 se aceptan como nueva función, histéresis y duración del impulso de DO0, DO1. Si no se desean modificar los valores, hay que transferir los valores antiguos.
CTRL_DO0	Habilitación DO0 Con este bit se habilita la salida DO0.
CTRL_DO1	Habilitación DO1 Con este bit se habilita la salida DO1.
ENSET_DN	Con este bit se habilita la carga del contador al contar en sentido ascendente.
ENSET_UP	Con este bit se habilita la carga del contador al contar en sentido descendente.
GATE_STP	Con este bit se cierra la puerta interna.
L_DIRECT	Con este bit se carga el contador de forma directa y anticipativa.
L_PREPAR	Con este bit se carga el contador de forma anticipativa.
NEUSTQ	Con este bit se acusa un arranque del FM 350-1. Después del arranque, el FM 350-1 no detecta si se produce un control o una entrada de datos hasta que se ha activado este bit. El bit NEUSTQ se activa con la instrucción CNT_CNTL1 en cuanto se ha activado la señal de realimentación FM_NEUST y la señal de realimentación es FM_NEUSTQ = 0 . Se desactiva con la instrucción CNT_CNTL1 cuando el bit FM_NEUST ha sido desactivado por el FM 350-1 y el bit FM_NEUSTQ ha sido activado por el FM 350-1. Si no se utiliza la instrucción CNT_CNTL1, es preciso realizar la coordinación del rearranque a través del programa de usuario.
OT_ERR_A	Con este bit se acusa un error de operador. Si necesita información más detallada sobre el error de operador, deberá leerla en la interfaz de realimentación antes de acusar el error. El aviso de error ya no es válido después del acuse.
RES_SYNC	Con este bit se desactiva el bit de realimentación STS_SYNC y, con ello, se acusa la carga del contador con la entrada de sincronización DI-Set.
RES_ZERO	Con este bit se desactivan los bits de realimentación STS_ZERO, STS_OFLW, STS_UFLW, STS_COMP1 y STS_COMP2.
SET_DO0	Este activa y desactiva la salida digital DO0 si para la salida se ha parametrizado el comportamiento "No se compara" y si se ha activado el bit de habilitación CRTL_DO0.
SET_DO1	Este activa y desactiva la salida digital DO1 si para la salida se ha parametrizado el comportamiento "No se compara" y si se ha activado el bit de habilitación CRTL_DO1.
SW_GATE	Con este bit se abre y cierra la puerta SW.
T_CMP_V1	Con este bit se carga el valor del byte 4 a 7 en el valor de comparación 1.
T_CMP_V2	Con este bit se carga el valor del byte 8 a 11 en el valor de comparación 2.

Interfaz de realimentación de los modos de contaje

Interfaz de realimentación de los modos de contaje

La siguiente tabla muestra la asignación de la interfaz de realimentación (las entradas) del FM 350-1 para los modos de contaje.

Offset de la dirección inicial	Parámetro	Significado
Byte 0 a 3	LATCH_LOAD	Valor de carga legible o valor de contaje almacenado con la función de congelación en la entrada digital
Bytes 4 a 7	ACT_CNTV	Estado del contador
Bytes 8 a 9	DA_ERR_W	Error de datos
Byte 10	OT_ERR_B	Error de operador
Byte 11	PARA	Bit 7: Parametrización realizada
	FM_NEUST	Bit 6: Solicitud de rearranque completo
	FM_NEUSTQ	Bit 5: Acuse de rearranque completo realizado
	DATA_ERR	Bit 4: Error de datos
	OT_ERR	Bit 3: Error de operador
	DIAG	Bit 2: Evento de diagnóstico
	–	Bit 1: reserva = 0
	–	Bit 0: reserva = 0
Byte 12	–	reserva = 0
Byte 13	STS_SW_GATE	Bit 7: Estado de la puerta SW
	STS_GATE	Bit 6: Estado de la puerta
	STS_SYNC	Bit 5: Sincronización
	STS_UFLW	Bit 4: Rebase del límite inferior
	STS_OFLW	Bit 3: Rebase del límite superior
	STS_ZERO	Bit 2: Paso por cero
	STS_DIR	Bit 1: Bit de sentido
	STS_RUN	Bit 0: Contador en funcionamiento

Offset de la dirección inicial	Parámetro	Significado
Byte 14	STS_COMP2	Bit 7: Estado remanente del comparador 2
	STS_COMP1	Bit 6: Estado remanente del comparador 1
	STS_CMP2	Bit 5: Estado de la salida DO1
	STS_CMP1	Bit 4: Estado de la salida DO0
	STS_STP	Bit 3: Estado de la entrada digital DI-Stop
	STS_STA	Bit 2: Estado de la entrada digital DI-Start
	STS_LATCH	Bit 1: Nuevo valor de congelación en modo isócrono
	STS_SET	Bit 0: Estado de la entrada digital DI-Set
Byte 15	–	Bit 7: reserva = 0
	STS_C_DOPARA	Bit 6: Función DO0/DO1, modificar histéresis o duración del impulso
	STS_RES_ZERO	Bit 5: Desactivar bit de estado para paso por cero, rebase por exceso, rebase por defecto o comparador
	STS_RES_SYNC	Bit 4: Desactivar bit de estado Sincronización
	STS_T_CMP_V2	Bit 3: Cargar valor de comparación 2
	STS_T_CMP_V1	Bit 2: Cargar valor de comparación 1
	STS_L_PREPAR	Bit 1: Carga anticipativa del contador
	STS_L_DIRECT	Bit 0: Carga directa y anticipativa del contador

Aclaraciones sobre los bits de realimentación de los modos de contaje

Bit de realimentación	Significado
DATA_ERR	Este bit indica que se ha registrado un error de datos (error de parametrización) en la interfaz de realimentación.
DIAG	El bit se activa cuando se ha actualizado el registro de diagnóstico DS1 para notificar un evento de diagnóstico. Una vez leído el registro DS1 se vuelve a desactivar el bit. Si no hay ninguna alarma de diagnóstico habilitada, este bit se puede utilizar como bit de inicio para la instrucción DIAG_INF integrada en el OB 1.
FM_NEUST	El FM 350-1 activa este bit cada vez que ejecuta un rearranque completo o cada vez que detecta un arranque del sistema, independientemente de si el sistema arranca de forma manual o automática. Con el siguiente flanco ascendente en el bit NEUSTQ se desactiva el bit FM_NEUST. A continuación, el FM 350-1 puede controlarse, lo que permite la lectura y la transferencia de datos desde el mismo.
FM_NEUSTQ	El FM 350-1 borra este bit cada vez que ejecuta un rearranque completo o cada vez que detecta un arranque del sistema, independientemente de si el sistema arranca de forma manual o automática. Se activa cuando se desactiva el bit FM_NEUST.
OT_ERR	Este bit se activa cuando se registra un error de operaror en la interfaz de realimentación. Se desactiva cuando se activa el bit OT_ERR_A. Mientras el bit OT_ERR permanece activo, no puede notificarse otro error de operador.
PARA	Este bit se activa si el módulo ha sido parametrizado sin errores. El registro de parámetros no contiene errores en el módulo. No obstante, este bit sólo se activa si se ha activado el bit FM_NEUSTQ. A partir de este momento los valores de la interfaz de realimentación son válidos y actuales.
STS_C_DOPARA	Bit de acuse para la modificación simultánea del comportamiento de DO0 y DO1, de la histéresis y de la duración del impulso. Si no desea que los valores se modifiquen, debe transferir los valores antiguos.

Bit de realimentación	Significado
STS_CMP1	Estado de la salida DO0
STS_CMP2	Estado de la salida DO1
STS_T_CMP_V1	Bit de acuse de la carga del valor de comparación 1
STS_T_CMP_V2	Bit de acuse de la carga del valor de comparación 2
STS_COMP1	Este bit indica el estado almacenado de que la salida DO0 estaba activada. Esto también ocurre en el caso de que la salida DO0 no se haya habilitado con CTRL_DO0. El estado almacenado se desactiva realizando un acuse con RES_ZERO.
STS_COMP2	Este bit indica el estado almacenado de que la salida DO1 estaba activada. Esto también ocurre en el caso de que la salida DO1 no se haya habilitado con CTRL_DO1. El estado almacenado se desactiva realizando un acuse con RES_ZERO.
STS_DIR	Este bit indica el sentido de contaje del contador: 0 = ascendente (LED DIR apagado) 1 = descendente (LED DIR encendido)
STS_GATE	Este bit indica el estado de la puerta. 0 = Puerta cerrada 1 = Puerta abierta
STS_LATCH	Este bit muestra en modo isócrono si se ha almacenado por lo menos un nuevo valor de congelación entre el penúltimo Ti y el último Ti. Si el bit está activado, LATCH_LOAD contiene el último valor de congelación. Si no se ha almacenado ningún otro valor de congelación, el bit no está activado. En el modo no isócrono el bit no está activado.
STS_L_DIRECT	Bit de acuse de la carga directa y anticipativa del contador y del valor de carga.
STS_L_PREPAR	Bit de acuse de la carga anticipativa del valor de carga.
STS_OFLW	Este bit indica que se ha producido un rebase por exceso. El estado almacenado se desactiva realizando un acuse con RES_ZERO.
STS_RES_SYNC	Desactivación del bit de realimentación STS_SYNC.
STS_RES_ZERO	Bit de acuse para la desactivación de los estados almacenados en los bits de realimentación STS_ZERO, STS_OFLW, STS_UFLW, STS_COMP1 y STS_COMP2
STS_RUN	Este bit equivale al bit 2^0 del contador. 0 = LED CR apagado 1 = LED CR encendido
STS_SET	Estado de la entrada digital DI-Set
STS_STA	Estado de la entrada digital DI-Start
STS_STP	Estado de la entrada digital DI-Stop
STS_UFLW	Este bit indica que se ha producido un rebase por defecto. El estado almacenado se desactiva realizando un acuse con RES_ZERO.
STS_SYNC	Este bit indica el estado almacenado de que el contador ha sido cargado por un evento en el DI-Set (sincronización). El estado almacenado se desactiva realizando un acuse con RES_SYNC.
STS_ZERO	Este bit indica el estado almacenado de que la lectura del contador tenía un paso por cero. El estado almacenado se desactiva realizando un acuse con RES_ZERO.

Interfaz de control de los modos de medición

Interfaz de control de los modos de medición

El parámetro LOAD_VAL (byte 0 a 3) tiene dos significados:

- Si se activa el bit L_PREPAR, LOAD_VAL se interpreta como límite inferior.
- Si se define el bit C_DOPARA, en el byte 0 puede definirse el comportamiento de la salida DO0.

La siguiente tabla muestra la asignación de la interfaz de control (las salidas) del FM 350-1 para los modos de medición.

Offset respecto a la dirección inicial	Parámetros	Asignación			
Byte 0 a 3	LOAD_VAL	Cargar límite inferior con el bit L_PREPAR			
Byte 0		Definir el comportamiento de la salida DO0 con el bit C_DOPARA			
		Bits 2 a 7:	Bit 1	Bit 0	Comportamiento de la salida DO0
		irrelevante	0	0	No se compara
		irrelevante	0	1	Fuera de los valores límite
		irrelevante	1	0	Rebase por defecto del límite inferior
		irrelevante	1	1	Rebase por exceso del límite superior
Byte 1		Reserva = 0			
Byte 2		Reserva = 0			
Byte 3		Reserva = 0			
Bytes 4 a 7	CMP_V1	Cargar límite superior con el bit T_CMP_V1			
Bytes 8 a 9	CMP_V2	Tiempo de actualización; cargar con el bit T_CMP_V2			
Byte 10 a 11	–	–			
Byte 12	– NEUSTQ – – OT_ERR_A – – –	Bit 7: reserva = 0 Bit 6: Acuse de rearranque completo Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: Acuse de error de operador Bit 2: reserva = 0 Bit 1: reserva = 0 Bit 0: reserva = 0			
Byte 13	– – – – SW_GATE GATE_STP – –	Bit 7: reserva = 0 Bit 6: reserva = 0 Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: Bit de control puerta SW Bit 2: Parada general de puerta Bit 1: reserva = 0 Bit 0: reserva = 0			

Offset respecto a la dirección inicial	Parámetros	Asignación
Byte 14	– – – – SET_DO1 SET_DO0 CTRL_DO1 CTRL_DO0	Bit 7: reserva = 0 Bit 6: reserva = 0 Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: Bit de control DO1 Bit 2: Bit de control DO0 Bit 1: Habilitación DO1 Bit 0: Habilitación DO0
Byte 15	– C_DOPARA[1)] RES_ZERO – T_CMP_V2[2)] T_CMP_V1[2)] L_PREPAR[2)] –	Bit 7: reserva = 0 Bit 6: Modificar función DO0 Bit 5: Desactivar bits de estado para rebase por exceso, rebase por defecto, fin de medición Bit 4: reserva = 0 Bit 3: Modificar tiempo de actualización Bit 2: Cargar límite superior Bit 1: Cargar límite inferior Bit 0: –
[1)] Este bit no se puede activar al mismo tiempo que el bit 1, 2 ó 3 del byte 15. [2)] Este bit no se puede activar al mismo tiempo que el bit 6 del byte 15.		

Aclaraciones sobre los bits de control de los modos de medición

Bit de control	Significado
C_DOPARA	Con este bit se modifica la función y el comportamiento de DO0. Los valores del byte 0 se aceptan como nueva función de DO0. Si no desea que los valores se modifiquen, debe transferir los valores antiguos.
CTRL_DO0	Habilitación DO0 Con este bit se habilita la salida DO0.
CTRL_DO1	Habilitación DO1 Con este bit se habilita la salida DO1
GATE_STP	Con este bit se cierra la puerta interna.
L_PREPAR	Con este bit se carga el límite inferior.
NEUSTQ	Con este bit se acusa un arranque del FM 350-1. Después del arranque, el FM 350-1 no detecta si se produce un control o una entrada de datos hasta que se ha activado este bit. El bit NEUSTQ se activa con la instrucción CNT_CNTL1 en cuanto se ha activado la señal de realimentación FM_NEUST y la señal de realimentación es FM_NEUSTQ = 0 . Se desactiva con la instrucción CNT_CNTL1 cuando el bit FM_NEUST ha sido desactivado por el FM 350-1 y el bit FM_NEUSTQ ha sido activado por el FM 350-1. Si no se utiliza la instrucción CNT_CNTL1, es preciso realizar la coordinación del rearranque a través del programa de usuario.
OT_ERR_A	Con este bit se acusa un error de operador. Si necesita información más detallada sobre el error de operador, deberá leerla en la interfaz de realimentación antes de acusar el error. El aviso de error ya no es válido después del acuse.

Bit de control	Significado
RES_ZERO	Con este bit se desactivan los bits de realimentación STS_OFLW, STS_UFLW y STS_COMP1.
SET_DO0	Este activa y desactiva la salida digital DO0 si para la salida se ha parametrizado el comportamiento "No se compara" y si se ha activado el bit de habilitación CRTL_DO0.
SET_DO1	Este activa y desactiva la salida digital DO1 si para la salida se ha parametrizado el comportamiento "No se compara" y si se ha activado el bit de habilitación CRTL_DO1.
SW_GATE	Con este bit se abre y cierra la puerta SW.
T_CMP_V1	Con este bit se carga el límite superior.
T_CMP_V2	Con este bit se carga el tiempo de actualización.

Interfaz de realimentación de los modos de medición

Interfaz de realimentación de los modos de medición

La siguiente tabla muestra la asignación de la interfaz de realimentación (las entradas) del FM 350-1 para los modos de medición.

Offset respecto a la dirección inicial	Parámetros	Asignación
Byte 0 a 3	LATCH_LOAD	Valor de medida
Bytes 4 a 7	ACT_CNTV	Estado del contador
Bytes 8 a 9	DA_ERR_W	Error de datos
Byte 10	OT_ERR_B	Error de operador
Byte 11	PARA	Bit 7: Parametrización realizada
	FM_NEUST	Bit 6: Solicitud de rearranque completo
	FM_NEUSTQ	Bit 5: Acuse de rearranque completo realizado
	DATA_ERR	Bit 4: Error de datos
	OT_ERR	Bit 3: Error de operador
	DIAG	Bit 2: Evento de diagnóstico
	–	Bit 1: reserva = 0
	–	Bit 0: reserva = 0
Byte 12		reserva = 0
Byte 13	–	Bit 7: reserva = 0
	STS_GATE	Bit 6: Estado de la puerta
	–	Bit 5: reserva = 0
	STS_UFLW	Bit 4: Rebase por defecto
	STS_OFLW	Bit 3: Rebase por exceso
	STS_COMP1	Bit 2: Final de medición
	STS_DIR	Bit 1: Bit de sentido
	STS_RUN	Bit 0: Contador en funcionamiento

Offset respecto a la dirección inicial	Parámetros	Asignación
Byte 14	–	Bit 7: reserva = 0
	–	Bit 6: reserva = 0
	STS_CMP2	Bit 5: Estado de la salida DO1
	STS_CMP1	Bit 4: Estado de la salida DO0
	STS_STP	Bit 3: Estado de la entrada digital DI-Stop
	STS_STA	Bit 2: Estado de la entrada digital DI-Start
	–	Bit 1: reserva = 0
	STS_SET	Bit 0: Estado de la entrada digital DI-Set
Byte 15	–	Bit 7: reserva = 0
	STS_C_DOPARA	Bit 6: Modificar función DO0
	STS_RES_ZERO	Bit 5: Desactivar bit de estado Fin de medición
	–	Bit 4: reserva = 0
	STS_T_CMP_V2	Bit 3: Modificar tiempo de actualización
	STS_T_CMP_V1	Bit 2: Cargar límite superior
	STS_L_PREPAR	Bit 1: Cargar límite inferior
	–	Bit 0: reserva = 0

Aclaraciones sobre los bits de realimentación de los modos de medición

Bit de realimentación	Significado
DATA_ERR	Este bit indica que se ha registrado un error de datos en la interfaz de realimentación.
DIAG	El bit se activa cuando se ha actualizado el registro de diagnóstico DS1 para notificar un evento de diagnóstico. Una vez leído el registro DS1 se vuelve a desactivar el bit. Si no hay ninguna alarma de diagnóstico habilitada, este bit se puede utilizar como bit de inicio para la instrucción DIAG_INF integrada en el OB 1.
FM_NEUST	El FM 350-1 activa este bit cada vez que ejecuta un rearranque completo o cada vez que detecta un arranque del sistema, independientemente de si el sistema arranca de forma manual o automática. Con el siguiente flanco ascendente en el bit NEUSTQ se desactiva el bit FM_NEUST. A continuación, el FM 350-1 puede controlarse, lo que permite la lectura y la transferencia de datos desde el mismo.
FM_NEUSTQ	El FM 350-1 borra este bit cada vez que ejecuta un rearranque completo o cada vez que detecta un arranque del sistema, independientemente de si el sistema arranca de forma manual o automática. Se activa cuando se desactiva el bit FM_NEUST.
OT_ERR	Este bit se activa cuando se registra un error de operador en la interfaz de realimentación. Se desactiva cuando se activa el bit OT_ERR_A. Mientras el bit OT_ERR permanece activo, no puede notificarse otro error de operador.
PARA	Este bit se activa si el módulo ha sido parametrizado sin errores. El registro de parámetros no contiene errores en el módulo. No obstante, este bit sólo se activa si se ha activado el bit FM_NEUSTQ. A partir de este momento los valores de la interfaz de realimentación son válidos y actuales.
STS_C_DOPARA	Bit de acuse para la modificación simultánea del comportamiento de DO0 y DO1, de la histéresis y de la duración del impulso. Si no desea que los valores se modifiquen, debe transferir los valores antiguos.
STS_CMP1	Estado de la salida DO0
STS_CMP2	Estado de la salida DO1

Bit de realimentación	Significado
STS_CMP_T_VAL1	Bit de acuse de la carga del límite superior
STS_CMP_T_VAL2	Bit de acuse para la carga del tiempo de actualización
STS_DIR	Este bit indica el sentido de contaje del contador: 0 = ascendente (LED DIR apagado) 1 = descendente (LED DIR encendido)
STS_GATE	Este bit indica el estado de la puerta. 0 = Puerta cerrada 1 = Puerta abierta
STS_L_PREPAR	Bit de acuse de la carga del límite inferior
STS_OFLW	Este bit indica el estado almacenado en el que un valor medido es mayor que el límite superior. El estado almacenado se desactiva realizando un acuse con RES_ZERO.
STS_RES_ZERO	Bit de acuse para la desactivación de los estados almacenados en los bits de realimentación STS_OFLW, STS_UFLW y STS_COMP1
STS_RUN	Este bit equivale al bit 2^0 del contador. 0 = LED CR apagado 1 = LED CR encendido
STS_SET	Estado de la entrada digital DI-Set
STS_STA	Estado de la entrada digital DI-Start
STS_STP	Estado de la entrada digital DI-Stop
STS_UFLW	Este bit indica el estado almacenado en el que un valor medido es menor que el límite inferior. El estado almacenado se desactiva realizando un acuse con RES_ZERO.

Uso del FM 350-1 desde el programa de usuario

Inclusión del FM 350-1 en un programa de usuario

Para integrar el FM 350-1 en un programa de usuario pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

Las instrucciones CNT_CTRL, CNT_CTL1 y CNT_CTL2 se activan en el programa de usuario de la CPU. Éstas permiten la comunicación entre la CPU y el FM 350-1.

Todos los datos pertenecientes a un canal del módulo se encuentran en el DB de la instrucción CNT_CTRL, CNT_CTL1 o CNT_CTL2 (DB de contaje (Página 7393)). Inserte en el proyecto un bloque de datos del tipo CNT_CHANTYPE1 mediante "Bloques de programa > Agregar bloque".

Además, para el FM 350-1 se puede utilizar también la instrucción DIAG_INF, que permite transferir información de diagnóstico al DB de contaje.

La instrucción CNT_CTL2 sólo se utiliza en el modo isócrono.

Consulte también

CNT_CTRL (Página 7388)

DIAG_INF (Página 7390)

CNT_CTL1 (Página 7390)
CNT_CTL2 (Página 7393)

FM 350-1: Online y diagnóstico

Información actual sobre el módulo

Diagnóstico estándar

Indicación de fallos mediante el LED de error de grupo del FM 350-1

¿Dónde se indica el fallo?

Si el LED rojo de error de grupo se enciende, significa que se ha producido un fallo en el módulo (fallo interno) o en las conexiones (fallo externo).

¿Qué fallos se indican?

La tabla siguiente muestra los tipos de fallos indicados por el LED de error de grupo.

Tipo de fallo	Causa del fallo	Solución
Fallos internos	Fallo en el test de la EPROM	Sustituir el módulo
	Fallo en test de RAM	Sustituir el módulo
	Disparo del temporizador de vigilancia (watchdog)	Sustituir el módulo
	Alarma de proceso perdida	Aumentar el intervalo de tiempo entre las causas de las alarmas
	Falta la parametrización del módulo	Definir y transferir la parametrización
Fallos externos	Conector codificador en posición incorrecta	Corregir la posición del conector codificador
	Tensión auxiliar 1L+/1M no conectada o cortocircuito en la alimentación del encóder DC24V	Corregir la conexión
	Cortocircuito o sobrecarga en la alimentación del encóder DC5.2V	Corregir la conexión
	Fallos en las señales del encóder de 5 V (rotura de hilo, cortocircuito, falta de línea)	Corregir la conexión
	La parametrización del módulo no concuerda con la posición del conector codificador	Corregir y transferir la parametrización, o bien modificar la posición del conector codificador

Disparo de una alarma de diagnóstico

Todos los fallos, exceptuando el del test de la EPROM, pueden disparar una alarma de diagnóstico si ésta se ha habilitado en los parámetros básicos de la parametrización. En los registros de diagnóstico DS0 y DS1 (Página 9550) se puede consultar qué fallo ha ocasionado que se encienda el LED.

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

¿Qué es una alarma de diagnóstico?

Si se quiere que un programa de usuario responda a un error interno o externo, se puede parametrizar una alarma de diagnóstico que interrumpa el programa cíclico de la CPU y llame al OB de alarma de diagnóstico.

¿Qué eventos pueden provocar una alarma de diagnóstico?

Los siguientes eventos pueden provocar una alarma de diagnóstico:

- Tensión auxiliar externa 1L+/1M defectuosa
- Alimentación de 5,2 V DC del encóder cortocircuitada o sobrecargada
- Falta parametrización del módulo
- Error en la parametrización del módulo
- La vigilancia de tiempo (watchdog) ha saltado
- Defecto en RAM
- Alarma de proceso perdida
- Señal A defectuosa (rotura de hilo, cortocircuito, falta un cable)
- Señal B defectuosa (rotura de hilo, cortocircuito, falta un cable)
- Señal N defectuosa (rotura de hilo, cortocircuito, falta un cable)
- Conector codificador en posición incorrecta

Habilitar la alarma de diagnóstico

En la configuración del hardware las alarmas del módulo se habilitan en los parámetros básicos y se especifica si el módulo debe activar una alarma de diagnóstico y/o una alarma de proceso.

Respuestas a una alarma de diagnóstico

Cuando se produce un evento que puede disparar una alarma de diagnóstico ocurre lo siguiente:

- La información de diagnóstico se registra en los registros de diagnóstico DS0 y DS1 (Página 9550) del módulo.
- El LED de error de grupo se enciende.
  Una vez eliminado el error, se apaga el LED de error de grupo.
- Se llama al OB de alarma de diagnóstico (por ejemplo, OB 82).
- El registro de diagnóstico DS0 se graba en la información de arranque del OB de alarma de diagnóstico.
- El contaje prosigue sin cambios si se ha programado un OB de alarma de diagnóstico. La CPU pasa a modo STOP si no hay OB de alarma de diagnóstico.

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de diagnóstico.

Registro de diagnóstico DS0 y DS1

Registro de diagnóstico DS0 y DS1

La información que indica qué evento ha provocado una alarma de diagnóstico se guarda en los registros de diagnóstico DS0 y DS1. El registro de diagnóstico DS0 consta de cuatro bytes; DS1 consta de 16 bytes, siendo los cuatro primeros idénticos a DS0.

Leer registro del módulo

El registro de diagnóstico DS0 (Página 9551) se transfiere automáticamente a la información de arranque en la llamada del OB de diagnóstico. Allí, los cuatro bytes se guardan en los datos locales (bytes 8 a 11) del OB de alarma de diagnóstico.

El registro de diagnóstico DS1 (Página 9551) (y, por lo tanto, también el contenido del DS0) puede leerse con la instrucción DIAG_INF del módulo. Esa lectura sólo tiene sentido si en el DS0 se notifica un fallo en un canal.

Asignación del registro de diagnóstico DS0

Asignación del registro de diagnóstico DS0 y de la información de arranque

La tabla siguiente muestra la asignación del registro de diagnóstico DS0 en la información de arranque. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Byte	Bit	Significado	Nota	Nº de evento
0	0	Fallo en módulo	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Error interno	Se activa para todos los fallos internos: • Fallo en test de RAM • La vigilancia de tiempo (watchdog) ha saltado • Alarma de proceso perdida	8:x:01
	2	Error externo	Se activa para todos los errores externos: • Tensión auxiliar 1L+/1M no conectada o alimentación del encóder de 5,2 V DC cortocircuitada • Alimentación de 5,2 V DC del encóder cortocircuitada o sobrecargada • Señales de 5 V defectuosas • Error de parametrización	8:x:02
	3	Fallo en un canal	Para obtener más detalles, consulte DS1, byte 4 (Página 9551)	8:x:03
	4	Defecto de la tensión auxiliar externa	Comprobar tensión	8:x.04
	6	Falta parametrización	Realizar la parametrización	8:x:06
	7	Error de parametrización	Para más información, consulte el capítulo Errores de datos (Página 9552)	8:x:07
1	0...3	Clase de tipo	Siempre tiene asignado un 8	
	4	Información de canal	Siempre tiene asignado un 1	
2	3	La vigilancia de tiempo (watchdog) ha saltado	Módulo defectuoso o fuertes interferencias	8:x:33
3	3	Defecto en RAM	Módulo defectuoso o fuertes interferencias	8:x:43
	6	Alarma de proceso perdida	Comprobar la configuración. Se ha detectado un evento y no se ha podido notificar porque el mismo evento aún no se ha sido acusado por la CPU.	8:x:46

Asignación del registro de diagnóstico DS1

Asignación del registro de diagnóstico DS1

El registro de diagnóstico DS1 se compone de 16 bytes. Los primeros 4 bytes son idénticos a los del registro de diagnóstico DS0. La siguiente tabla muestra la asignación de los bytes restantes. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Este registro es introducido por la instrucción DIAG_INF en el DB de la CNT_CTL1 a partir de la DW54.

Byte	Bit	Significado	Nota	Nº de evento
4	0...6	Tipo de canal	Siempre con un 76H asignado	
	7	Otros tipos de canal	Siempre tiene asignado un 0	
5	0...7	Longitud de la información de diagnóstico	Siempre con un 10H asignado	
6	0...7	Número de canales	Siempre tiene asignado un 1	
7	0	Vector de error de canal	En caso de un error de canal siempre con un 1 asignado	
8	0	Señal A errónea		8:x:B0
	1	Señal B errónea		8:x:B1
	2	Señal N errónea		8:x:B2
	4	Fallo en alimentación de 5,2 V del encóder		8:x:B4
	5...7	Reservado		
9 ... 15		Reservado		

Error de datos

¿Cuándo se producen errores de datos?

Cuando se transfieren nuevos parámetros al módulo, el FM 350-1 comprueba dichos parámetros. Si se producen errores durante esta comprobación, el módulo notifica estos errores.

¿Dónde se indican los errores de datos?

La instrucción CNT_CTL1 registra los errores de datos con sus números de error en el DB de CNT_CTL1. La variable DA_ERR_W permite acceder a esta palabra de datos en el programa de usuario.

¿Qué errores de datos existen?

En la tabla siguiente se muestran los errores de datos, junto con el número de error y el significado.

Nº	Significado
0	No hay error
200	Posición incorrecta del adaptador del margen o falta dicho adaptador
201	La posición del adaptador no es compatible con el encóder parametrizado
202	Valor erróneo para el diagnostico de la pareja de señales
203	Valor erróneo para evaluación de las señales
204	Valor erróneo para filtro de entrada de las señales de contaje de 24 V

Nº	Significado
205	Valor erróneo para el filtro de entrada de las entradas digitales
206	Inversión del sentido no admisible
207	Comportamiento de DO0 parametrizado incorrectamente
208	Comportamiento de DO1 parametrizado incorrectamente
209	Rango excedido en la duración del impulso
211	Se ha seleccionado un modo de operación incorrecto
212	No se ha indicado ninguna puerta o se han indicado ambas
213	Sentido principal de contaje parametrizado incorrectamente
214	Rango excedido en el límite superior de contaje
215	Al parametrizar la alarma de proceso "Alcanzar el valor de comparación adelante o atrás" se ha definido un sentido de contaje distinto del especificado en la parametrización del comportamiento de las salidas "Activar al alcanzarse el valor de comparación por la duración del impulso adelante o atrás". Estos sentidos de contaje deben coincidir.
216	Las alarmas por control de puerta sólo son posibles en los modos de operación con control de puerta.
217	En el comportamiento "Activar si estado del contador >= valor de comparación" o "Activar si estado del contador <= valor de comparación", no se soporta una alarma al "alcanzarse los valores de comparación".
218	Alarma en caso de paso por cero no permitida
219	Codificación del ajuste "Congelar" incorrecta
220	Control de puerta parametrizado incorrectamente
221	Rango excedido en valor límite inferior o valor de carga
222	Rango excedido en valor límite superior o valor de comparación 1
223	Rango excedido en tiempo de actualización o valor de comparación 2
224	Rango excedido en impulsos por vuelta del encóder

¿Cómo se acusan los errores de datos?

Corrija los valores de los parámetros de acuerdo con los valores predeterminados. Retransfiera el juego de parámetros corregido al FM 350-1. Éste vuelve a comprobar los parámetros y borra los errores de datos del DB.

Errores de operador

¿Cuándo aparecen los errores de operador?

Los errores de operador aparecen cuando se controla el módulo incorrectamente por especificar las señales de control incorrectamente.

¿Dónde se indican los errores de operador?

La instrucción CNT_CTL1 registra los números de errores de operador en el DB. Activando el parámetro de salida OT_ERR la instrucción CNT_CTL1 muestra que se ha producido un error de operador. La variable OT_ERR_B permite acceder a este byte de datos en el programa.

¿Qué errores de operador existen?

En la tabla siguiente se muestran los errores de operador, junto con el número de error y el significado.

Nº	Significado
0	No hay error
1	El modo de operación no se puede iniciar con puerta SW
2	El modo de operación no se puede cancelar
4	Solo se soporta con la CPU en STOP
5	Solo se puede activar el bit de control para la parametrización
6	Tarea no permitida
10	Rango excedido en valor límite inferior o valor de carga
11	Rango excedido en valor límite superior o valor de comparación 1
12	Rango excedido en tiempo de actualización o valor de comparación 2
20	Comportamiento de DO0 parametrizado incorrectamente
21	Comportamiento de DO1 parametrizado incorrectamente
22	Rango excedido en la duración del impulso
90	Véase CNT_CTL2 (Página 7393)
91	Véase CNT_CTL2 (Página 7393)

¿Cómo se acusan los errores de operador?

Acuse el error con el parámetro de entrada OT_ERR_A de la instrucción CNT_CTL1 o CNT_CTL2.

Uso del FM 350-2

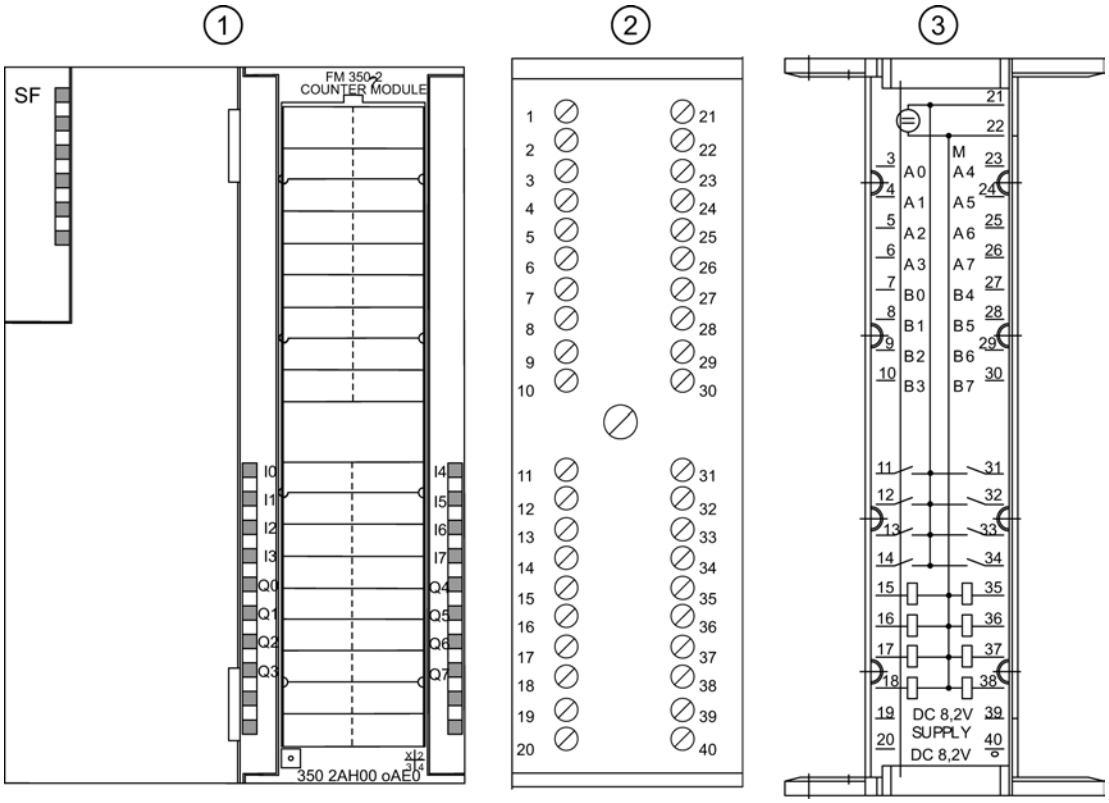
Asignación de conectores del conector frontal del FM 350-2

Conector frontal

En el conector frontal de 40 polos se conectan:

- Señales de contaje
- Entradas digitales
- Salidas digitales
- Alimentación del encóder y tensión del módulo

La figura siguiente muestra el lado frontal del módulo, el conector frontal y el lado interior de la tapa frontal con la asignación de terminales.



① Frontal del módulo
② Conector frontal
③ Interior de la puerta frontal

Asignación del conector frontal

La tabla siguiente muestra la asignación de terminales del conector frontal del FM 350-2.

Terminal	Nombre	Entrada/salida	Función
1	-	-	No conectado
2	-	-	No conectado
3	A0	Entrada	Entrada de contaje NAMUR / BERO, canal 0
4	A1	Entrada	Entrada de contaje NAMUR / BERO, canal 1
5	A2	Entrada	Entrada de contaje NAMUR / BERO, canal 2
6	A3	Entrada	Entrada de contaje NAMUR / BERO, canal 3
7	B0	Entrada	Entrada de dirección BERO, canal 0
8	B1	Entrada	Entrada de dirección BERO, canal 1
9	B2	Entrada	Entrada de dirección BERO, canal 2
10	B3	Entrada	Entrada de dirección BERO, canal 3
11	I0	Entrada	Entrada de puerta hardware BERO, canal 0
12	I1	Entrada	Entrada de puerta hardware BERO, canal 1
13	I2	Entrada	Entrada de puerta hardware BERO, canal 2

Terminal	Nombre	Entrada/salida	Función
14	I3	Entrada	Entrada de puerta hardware BERO, canal 3
15	Q0	Salida	Salida digital 0,5 A, canal 0
16	Q1	Salida	Salida digital 0,5 A, canal 1
17	Q2	Salida	Salida digital 0,5 A, canal 2
18	Q3	Salida	Salida digital 0,5 A, canal 3
19	P8V2	Salida	Alimentación para detector NAMUR 8,2 V
20	P8V2	Salida	Alimentación para detector NAMUR 8,2 V
21	L+	Entrada	Alimentación del módulo de 24 V
22	M	Entrada	Masa del módulo
23	A4	Entrada	Entrada de contaje NAMUR / BERO, canal 4
24	A5	Entrada	Entrada de contaje NAMUR / BERO, canal 5
25	A6	Entrada	Entrada de contaje NAMUR / BERO, canal 6
26	A7	Entrada	Entrada de contaje NAMUR / BERO, canal 7
27	B4	Entrada	Entrada de dirección BERO, canal 4
28	B5	Entrada	Entrada de dirección BERO, canal 5
29	B6	Entrada	Entrada de dirección BERO, canal 6
30	B7	Entrada	Entrada de dirección BERO, canal 7
31	I4	Entrada	Entrada de puerta hardware BERO, canal 4
32	I5	Entrada	Entrada de puerta hardware BERO, canal 5
33	I6	Entrada	Entrada de puerta hardware BERO, canal 6
34	I7	Entrada	Entrada de puerta hardware BERO, canal 7
35	Q4	Salida	Salida digital 0,5 A, canal 4
36	Q5	Salida	Salida digital 0,5 A, canal 5
37	Q6	Salida	Salida digital 0,5 A, canal 6
38	Q7	Salida	Salida digital 0,5 A, canal 7
39	P8V2	Salida	Alimentación para detector NAMUR 8,2 V
40	P8V2	Salida	Alimentación para detector NAMUR 8,2 V

Nota

Los circuitos para las entradas de contaje (alimentación del encóder, señales del encóder) están aislados de la tierra de la CPU.

Todas las entradas están unidas galvánicamente y aisladas del S7-300.

Encontrará más información sobre la estructura y el cableado en el manual de puesta en servicio y montaje (http://support.automation.siemens.com/WW/view/es/1105178) del módulo de contaje FM 350-2.

Alimentación de tensión de 24 V

Conecte un tensión continua de 24 V en los terminales L+ y M para alimentar el FM 350-2.

Alimentación del encóder 8,2 V DC

A partir de los 24 V de alimentación, el módulo genera una tensión de 8,2 V (máx. 200 mA). Esta tensión está disponible en los terminales P8V2 (pines 19, 20, 39 y 40) para la alimentación de los detectores NAMUR y es resistente a cortocircuitos.

Se controla que la alimentación del encóder se mantenga a 8,2 V.

Señales de encóder A0 a A7 y B0 a B7

Se pueden conectar cuatro tipos diferentes de encóders:

- Sensores NAMUR conforme a la DIN 19234 (con diagnóstico):
  Las señales se conectan a los terminales A0 a A7.
- Encóder incremental de 24 V:
  Las señales comprendidas entre A0/B0 y A7/B7 se conectan a los terminales con la identificación correspondiente.
- Encóder incremental de posición de 24 V con nivel de sentido:
  Las señales de contaje se conectan a los terminales A0 a A7. Los niveles de sentido se conectan a los terminales A0 a A7.
- Detectores de proximidad de 24 V:
  Las señales se conectan a los terminales A0 a A7.

Nota

Para alimentar los detectores de 24 V es preciso utilizar una fuente de alimentación externa de 24 V DC.

Entradas digitales de I0 a I7 (puertas hardware)

Para el control de puerta del contador se pueden usar las entradas digitales comprendidas entre I0 y I7.

A cada canal de contaje le corresponde una entrada digital para iniciar y detener el contador correspondiente.

Las entradas digitales funcionan con una tensión nominal de 24 V.

Salidas digitales de Q0 a Q7

El FM 350-2 dispone de salidas digitales de Q0 a Q7 para lanzar directamente procesos de control.

Hay disponible una salida digital para cada contador.

Las salidas digitales se alimentan con la tensión de alimentación de 24 V del FM 350-2.

Las salidas digitales son salidas tipo PNP y pueden trabajar con una corriente de carga de 0,5 A. Están protegidas contra sobrecarga y cortocircuito.

Nota

Se pueden conectar directamente relés y contactores sin circuitería externa.

Configurar FM 350-2

Parametrizar FM 350-2

Especificar canales

Seleccionar ajustes de canal

Ajuste del canal

En este caso es posible ajustar la función que tienen los canales del módulo:

- Contador individual: En el grupo seleccionado cada canal funciona por separado como contador.
- Contador de dosificación: 4 canales de contaje se agrupan respectivamente en un canal de dosificación. Desde este canal pueden controlarse hasta 4 unidades de dosificación.

Se dispone de las siguientes posibilidades de selección:

- Canal 0..7 como contador individual
- Canal 0 como contador de dosificación, canal 4..7 como contador individual
- Canal 0..3 como contador individual, canal 4 como contador de dosificación
- Canal 0 como contador de dosificación y canal 4 como contador de dosificación

Habilitar entrada de contador para sensores NAMUR

Sensores NAMUR

En este campo es posible habilitar las entradas de contador en las que hay conectado un sensor NAMUR.

Dispone de las siguientes posibilidades de selección:

- Entrada de contador canal 0..3 NAMUR
  y/o
- Entrada de contador canal 4..7 NAMUR

Véase también: "Señales NAMUR (Página 9559)" y "Habilitar vigilancia de hardware para sensores NAMUR (Página 9565)"

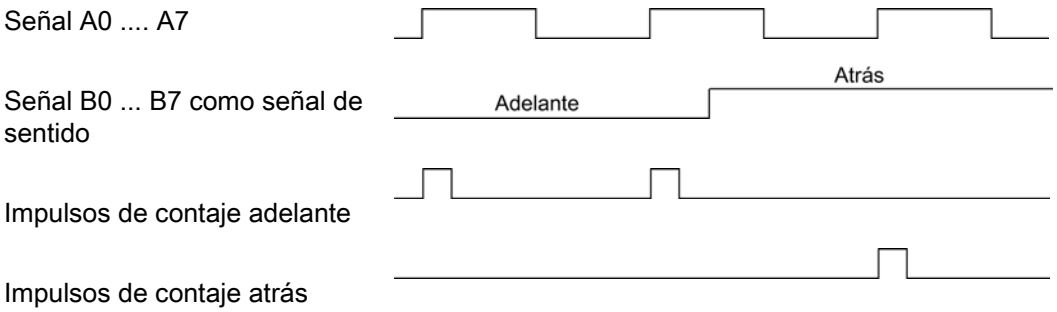
Señales NAMUR

Sensores NAMUR

El detector proporciona una señal de contaje conforme con la norma DIN 19234 que se ha de conectar a los terminales A0 a A7 del conector frontal.

Se puede conectar una señal para la detección del sentido (24 V) a los terminales B0 a B7 del contador pertinente. Si el detector no proporciona una señal pertinente en el S7 se puede generar y conectar también una identificación, o bien se puede usar una señal de proceso equivalente.

La figura siguiente muestra la evolución temporal de las señales del detector NAMUR con nivel de sentido.



Es posible habilitar las entradas A0 … A3, A4 … A7 o bien todas ellas para conectar los sensores NAMUR.

¿Cómo se vigilan las señales?

Si se ha parametrizado el FM 350-2 convenientemente, el módulo vigila cualquier rotura de hilo o cortocircuito que ocurra en A0 … A7. Con ello, indirectamente se vigilará también la alimentación del detector de 8,2 V. Esta vigilancia sólo es posible en los sensores NAMUR.

En caso de rotura de hilo o cortocircuito, se encenderá el LED de error de grupo; adicionalmente se puede definir también una alarma de diagnóstico.

⚠ PRECAUCIÓN

Se pueden producir daños materiales.

La conexión de un encóder o un detector en un canal del FM 350-2 que había sido parametrizado para un sensor NAMUR puede dañar el módulo.

Conecte un sensor NAMUR sólo en aquellos canales del FM 350-2 que hayan sido parametrizados para conectar sensores NAMUR.

Seleccionar asignación de la interfaz de notificación

Asignación de la interfaz de notificación bytes 8 a 15

Este campo permite seleccionar la forma en la que se asignan los bytes 8 a 15 de la interfaz de notificación.

Para las variables USER_STAT_WORD_0 a USER_STAT_WORD_3 existen las siguientes posibilidades de selección:

- Tipo de datos
  - WORD
  - DWORD
- Canal
  - Canal 0 a 7
- Valor
  - Valor de contaje
  - Valor medido
- Word
  - Se puede depositar tanto la palabra baja como la palabra alta de un valor o bien ambas. Es posible acceder a estos valores de forma directa.

Representación de los valores de contaje y medidos de todos los canales en los datos de usuario avanzados

Representación de todos los valores de contaje y medidos en la imagen de datos de usuario

Bajo determinados requisitos es posible representar todos los valores de contaje y medidos (uso de los datos de usuarioavanzados).

Para utilizar los datos de usuario avanzados rigen las siguientes condiciones marco:

- Proyecte el módulo "6ES7350-2AH01-0AE0 E" desde el catálogo de módulos.
- La casilla de verificación "Utilizar interfaz de datos de usuario avanzada" está activada.
- Los datos de usuario avanzados sólo pueden utilizarse de forma descentralizada después de un IM 153-2 V5.0 (número de pedido 6ES7153-2BA01-0XB0 V5.0) o superior.
- Para ello hay disponibles 40 bytes de datos de entrada.
- El ciclo de actualización de los datos de usuario avanzados es diferente al de los 16 bytes de datos de entrada. Dependiendo del tiempo de ciclo PROFIBUS DP, el ciclo de actualización de los datos de usuario avanzados es de entre 100 y 200 ms.

La estructura de los primeros 16 bytes en los datos de usuario es como hasta ahora y debe consultarse en el modo de operación correspondiente. Los 24 bytes restantes se definen por el modo de operación.

Dependiendo del modo de operación, en los datos de usuario avanzados se depositan los valores (valor de contaje o medido) de los canales 2 a 7. La asignación por defecto de estos

valores es "0". Mientras no se haya registrado ningún valor válido, los datos de entrada tienen asignado el valor 0.

Estructura de los datos de usuario avanzados en función del modo de operación:

Byte	Modos de contaje	Modos de medición	Modos de dosificación (dos contadores de dosificación)
0..15	Como hasta ahora: estructura modificable mediante parametrización	Como hasta ahora: estructura modificable mediante parametrización	Como hasta ahora: estructura modificable mediante parametrización
16..19	Valor de contaje, canal 2	Valor medido, canal 2	0
20..23	Valor de contaje, canal 3	Valor medido, canal 3	0
24..27	Valor de contaje, canal 4	Valor medido, canal 4	Contador de dosificación, canal 4
28..31	Valor de contaje, canal 5	Valor medido, canal 5	0
32..35	Valor de contaje, canal 6	Valor medido, canal 6	0
36..39	Valor de contaje, canal 7	Valor medido, canal 7	0

Copiar canales

Copiar canales

Este parámetro permite copiar un canal en uno o más canales.

1. Seleccione el canal cuyos parámetros desee copiar (origen del canal).
2. Active los canales en los que desee copiar los parámetros (objetivo del canal).
3. Pulse el botón "Copiar".

Modos de operación del canal n

Seleccionar el modo de operación

Descripción

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el módulo de contaje.

Modo de operación	Descripción
Contaje sin fin (Página 9468) (con o sin puerta)	El módulo de contaje cuenta en modo sin fin a partir del estado actual del contador.
Contaje único (Página 9469) (con puerta HW o SW)	El módulo de contaje cuenta al abrir la puerta desde cero hasta el límite de contaje superior parametrizado o bien desde el límite de contaje superior parametrizado hasta cero.

Modo de operación	Descripción
Contaje periódico (Página 9470) (con puerta SW o HW)	El módulo de contaje cuenta al abrir la puerta entre cero y el límite de contaje superior parametrizado.
Medición de la frecuencia (Página 9478) (con puerta SW o HW)	El módulo de contaje registra los impulsos que llegan en una ventana de tiempo parametrizada de n x 10 ms y determina con ello la frecuencia actual.
Medición de la velocidad (Página 9480) (con puerta HW o SW)	El módulo de contaje registra los impulsos que llegan en una ventana de tiempo parametrizada de n x 10 ms y determina la velocidad a partir del número de impulsos parametrizado por vuelta del encóder.
Medición del período (Página 9481) (con puerta HW o SW)	El módulo de contaje registra los impulsos que llegan en una ventana de tiempo parametrizada de n × 10 ms y determina con ello la duración del período.
Dosificación (Página 9484) (con puerta HW o SW)	En este modo de operación se agrupan cuatro canales de contaje en un solo canal de dosificación. Desde el punto de vista funcional, este modo de operación es idéntico al modo de operación "Contaje único".

El ajuste predeterminado es el modo de operación "Contaje sin fin".

Determinar el límite superior de contaje

Descripción

Si se parametriza un límite superior de contaje en combinación con un sentido principal de contaje (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño. El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El límite superior de contaje puede adoptar un valor > 1 hasta el límite superior del rango de contaje.

Definir el tiempo de medición

Descripción

Si se ha seleccionado un modo de medición, el módulo actualiza los valores medidos de forma cíclica. El tiempo de integración es el producto del tiempo de medición × 10 ms. El tiempo de medición se predefine como factor n.

Si desea obtener una descripción detallada sobre el rango de valores para el tiempo de medición, consulte el apartado "Conceptos básicos sobre la operación de medición (Página 9474)".

Uso de la puerta HW

Uso de la puerta HW

Se puede seleccionar si el canal debe contar con o sin puerta HW.

Active la casilla de verificación "Utilizar puerta HW" para contar con la puerta HW.

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Selección del sentido principal de contaje

Descripción

Si se parametriza un sentido principal de contaje (Página 9471) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño, ajustando para ello un límite de contaje superior. El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El sentido principal de contaje sólo puede seleccionarse en los modos de operación "Contaje único", "Contaje periódico" y "Dosificación".

Se puede seleccionar el siguiente comportamiento:

- Adelante
- Atrás

Encóders del canal n

Seleccionar el tipo de señal del encóder

Seleccionar tipo de señal

1. Seleccione el tipo de señal del encóder utilizado.
   Dependiendo de su selección y del modo de operación se bloquearán aquellas opciones no permitidas para ese determinado encóder. Las opciones bloqueadas se identifican con letra gris en la correspondiente ventana de selección.
2. Seleccione las opciones que desee utilizar con el encóder seleccionado.

Es posible selecciona un tipo de señal de uno de los encóders siguientes:

- Encóder: incremental de 24 V (Página 9564) (24 V pista A+B desfasado)
- Encóder: impulso y sentido de 24 V (Página 9564)

Encóder: incremental de 24 V

Encóder incremental de 24 V

Seleccione el tipo de señal "24 V pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A* y B*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal (Página 9494)	Simple	Sólo se evalúa un flanco de señal A.
	Doble	Se evalúan ambos flancos de la señal A.
	Cuádruple	Se evalúan ambos flancos de las señales A y B.
Sentido de contaje (Página 9564)	Normal	Para adaptar el sentido de contaje al proceso
	Invertido	Para adaptar el sentido de contaje al proceso

Encóder: impulso y sentido de 24 V

Encóder incremental de posición de 24 V

Seleccione el tipo de señal "24 V Impulso y sentido" si desea utilizar un encóder incremental de posición de 24 V con la señal de contaje A* y la señal de sentido B*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal	–	Sin relevancia
Sentido de contaje (Página 9564)	Invertido	Para adaptar el sentido de contaje al proceso
	Normal	Para adaptar el sentido de contaje al proceso

Seleccionar el sentido de contaje

Modificar el sentido de contaje

Con el parámetro "Sentido de contaje normal" o "Sentido de contaje inverso" puede cambiar el sentido de contaje sin necesidad de cambiar el cableado.

Si se emplea la parametrización "Sentido de contaje principal atrás (Página 9471)", el sentido de contaje **no** se invierte automáticamente.

Introducir impulsos por vuelta del encóder

Impulsos por vuelta del encóder

Para el modo Medición de velocidad (Página 9480) es preciso parametrizar adicionalmente los impulsos por vuelta del encóder.

El valor para el número de impulsos por vuelta del encóder se encuentra en la placa de características o en los datos técnicos del encóder.

Definir histéresis

Funcionamiento básico de la histéresis

Un encóder puede pararse en una posición determinada y "oscilar" alrededor de dicha posición. Esto hace que el estado del contador fluctúe alrededor de un valor determinado. Si hay un valor de comparación en este rango de fluctuación, por ejemplo, la salida correspondiente se activa y desactiva de acuerdo con el ritmo de las fluctuaciones. Para evitar estas conmutaciones cuando se producen pequeñas oscilaciones, el FM 350-2 cuenta con una histéresis parametrizable.

Para la histéresis se pueden parametrizar valores comprendidos entre 0 y 255.

Dependiendo del valor parametrizado, la histéresis actúa tal como se indica a continuación:

Histéresis	Efecto
Valor de histéresis n = 0, 1	La histéresis no tiene efecto (desactivada). La salida reaccionará al más mínimo cambio del estado del contador.
2 ≤ valor de histéresis n ≤ 255	La histéresis tiene efecto. La salida reacciona sólo cuando el estado del contador se ha distanciado n unidades del valor de comparación.

La histéresis también actúa en caso de rebase por exceso y por defecto.

Habilitar vigilancia de hardware para sensores NAMUR

¿Cómo se vigilan los sensores NAMUR?

El FM 350-2 puede vigilar si en A0 … A7 hay una rotura de hilo o un cortocircuito. Con ello, indirectamente se vigilará también la alimentación del detector de 8,2 V.

Active la casilla de verificación "Vigilancia hardware para sensores NAMUR", si desea que el FM 350-2 supervise las señales del sensor.

Salidas del canal n

Valores de comparación y salidas

Descripción

Es posible depositar ocho valores de comparación en el FM 350-2 y asignarlos a las ocho salidas digitales (valor de comparación 0: Q0, Valor de comparación 1: Q1, etc.). La salida correspondiente se puede activar en función del estado del contador y del valor de comparación.

En el modo de operación "Dosificación (Página 9484)" para cada canal de dosificación del FM 350-2 es posible parametrizar cuatro valores de comparación.

Valores de comparación

Los valores de comparación se introducen durante la parametrización.

En el modo RUN de la CPU, los valores de comparación pueden introducirse en el DB de la instrucción CNT2_WR/CNT2WRPN (CMP_VAL0…7) y transferirse al FM 350-2 utilizando las peticiones 30...37 ó 60...62. El contaje no se ve afectado por este hecho.

Los valores de comparación han de estar dentro de los rangos de contaje de los distintos modos operativos; no se soportan los límites del rango de contaje.

No obstante, si en un modo de operación con sentido principal de contaje se fija el valor de comparación en el valor inicial o final, las salidas digitales mostrarán el comportamiento siguiente:

Comportamiento parametrizado de las salidas digitales	Reacción de la salida digital asignada
Activar si la lectura del contador >= valor de comparación	La salida está siempre activada, independientemente de la lectura del contador.
Activar si la lectura del contador <= valor de comparación	La salida está siempre desactivada, independientemente de la lectura del contador.

Habilitación de las salidas

Antes de poder controlar las salidas es preciso haberlas habilitado activando los bits correspondientes en el DB de contador. Si se desactiva uno de estos bits, la salida correspondiente se desconectará de inmediato. Los bits se transfieren entre el bloque de datos y el módulo con la instrucción CNT2_CTR.

Salida	...habilitada por
Q0…7	CTRL_DQ0…7

Desactivar las salidas

Las salidas se desactivan independientemente de la parametrización cuando se dan los eventos siguientes:

- Activación de la vigilancia de tiempo (watchdog) del módulo (error interno)
- Supresión de los bits de habilitación (CTRL_DQ0...7 para Q0...7 en el DB)

Control de las salidas

Tras su habilitación con CTRL_DQ0…7 las salidas se pueden activar y desactivar con las señales de control SET_DQ0…7, independientemente de conmutarlas a través de los comparadores.

En la relación entre el control y la activación hay que tener en cuenta que el control tiene prioridad sobre la activación por los comparadores. Esto significa:

- si no se prevé para la salida una activación por el comparador, la salida puede emplearse como una salida digital.
- Si se va a utilizar una función de comparación para las salidas, éstas se pueden seguir controlando con SET_DQ0…7. De este modo es posible simular el efecto de las funciones de comparación mediante el programa de control:
  - Con el flanco ascendente de SET_DQ0…7 se activa la salida.
  - Un flanco descendente de SET_DQ0…7 desactiva la salida de nuevo.

  Obsérvese que los comparadores continúan activos y que pueden activar o desactivar la salida si hay un cambio en el resultado de la comparación.

Nota

Una salida activada con SET_DQ0…7 no se desactiva con el comparador.

Comportamiento de las salidas en los modos de contaje

Definir el comportamiento de las salidas

Comportamiento de las salidas DQ0 a DQ7

Seleccione el comportamiento deseado de la salida correspondiente. Active para ello la casilla de verificación correspondiente.

- No se compara (Página 9568)
- Activar si la lectura del contador >= valor de comparación (Página 9568)
- Activar si la lectura del contador <= valor de comparación (Página 9569)

Nota

Para activar las opciones, es preciso introducir los valores de los valores de comparación en el DB con los datos de contaje, así como transferirlos al módulo con la instrucción CNT2_WR; además, es preciso activar los bits CNT2WRPN y CTRL_DQ0…7 en el DB para habilitar las salidas. Véase también "Valores de comparación y salidas (Página 9564)".

Nota

En los modos de operación "Medición de frecuencia", "Medición de velocidad" y "Medición del período" no es posible parametrizar las salidas.

Ajuste predeterminado de las salidas

En el ajuste predeterminado las salidas están desactivadas.

No se compara

No se compara

La salida permanece desactivada y no se ve afectada por los eventos valor de comparación, paso por cero, salto del valor final al inicial (rebase por exceso), o viceversa (rebase por defecto).

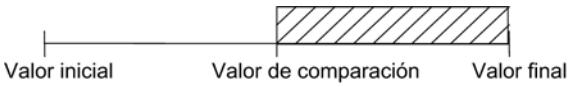
Activar si la lectura del contador >= valor de comparación

Activar si la lectura del contador >= valor de comparación

La salida se pone a "1" si el contador se encuentra en el rango comprendido entre el valor de comparación y el valor final.

La salida se pone a "1" si el contador se ajusta a un valor comprendido entre el valor de comparación y el valor final.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área.



Condición:

Si desea parametrizar una salida "Activar si el estado de contador >= valor de comparación", se aplica la siguiente condición marco: El tiempo que transcurre entre que se alcanza el valor de comparación y el valor de inicio o fin tiene que ser mayor que el tiempo de conmutación mínimo de las salidas (tiempo de conmutación: 300 µs). De lo contrario, se perderán los impulsos de control de las salidas.

Consulte también

Efecto de la histéresis (Página 9569)

Activar si la lectura del contador <= valor de comparación

Activar si la lectura del contador <= valor de comparación

La salida se pone a "1" si el contador se encuentra en el rango comprendido entre el valor de comparación y el valor inicial.

La salida se pone a "1" si el contador se ajusta a un valor comprendido entre el valor de comparación y el valor inicial.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área.

Condición:

Si desea parametrizar una salida "Activar si el estado de contador <= valor de comparación", se aplica la siguiente condición marco: El tiempo que transcurre entre que se alcanza el valor de comparación y el valor de inicio o fin tiene que ser mayor que el tiempo de conmutación mínimo de las salidas (tiempo de conmutación: 300 µs). De lo contrario, se perderán los impulsos de control en las salidas.

Consulte también

Efecto de la histéresis (Página 9569)

Efecto de la histéresis

Funcionamiento de la histéresis con la parametrización "Activar si el valor del contador >= valor de comparación o <= valor de comparación"

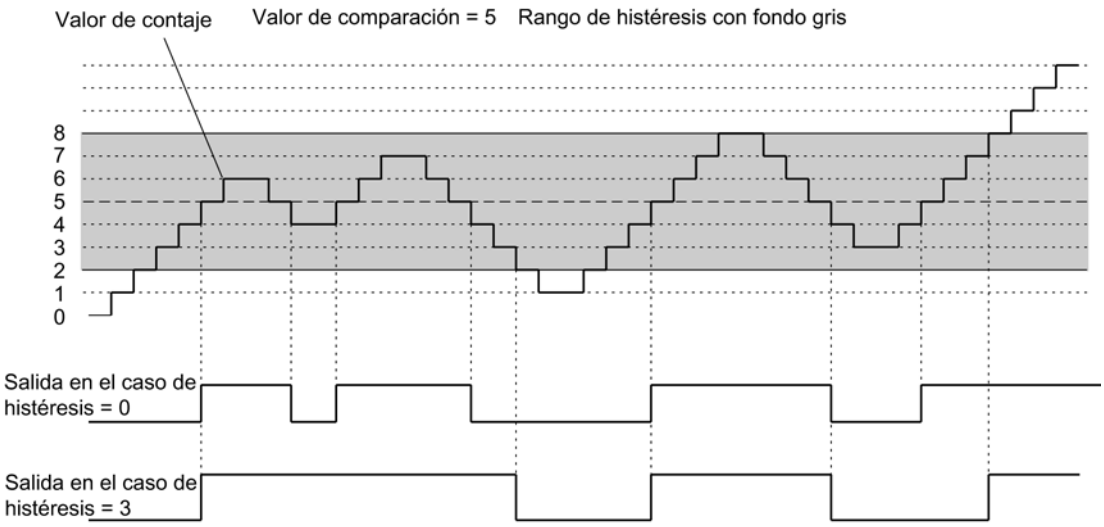
La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (= desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo el valor de comparación = 5.

La parametrización en este ejemplo es:

- Sentido principal de contaje: "Adelante"
- Comportamiento de la salida: "Activar si el estado del contador >= valor de comparación"

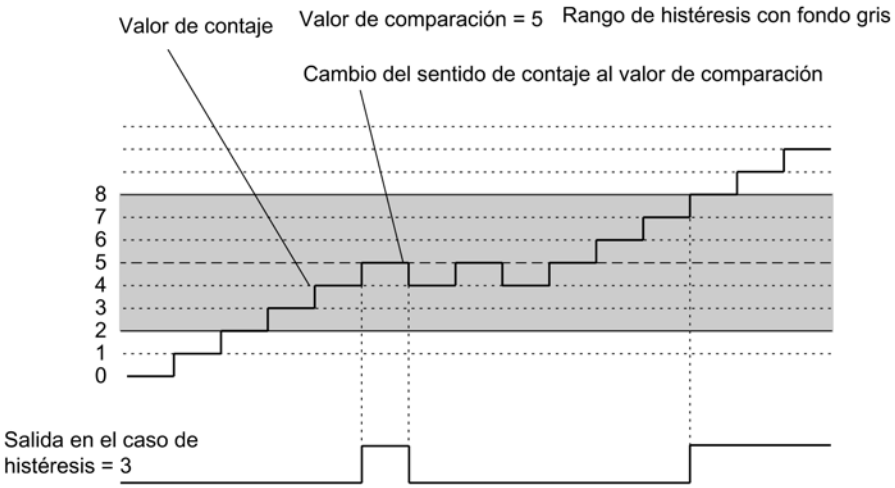
Al alcanzarse la condición de comparación (estado del contador = 5) se activa la histéresis. Con la histéresis activada el resultado de la comparación permanece invariable.

Cuando el valor de contaje abandona el rango de histéresis, en el ejemplo con el valor de contaje 2 u 8, la histéresis se desactiva. El comparador vuelve a conmutar según sus condiciones de comparación, en el ejemplo con el valor de contaje 5.



Nota

Si el estado del contador es igual al valor de comparación y la histéresis está activa, el FM 350-2 desactiva la salida cuando se invierte el sentido de contaje en el valor de comparación (véase la figura siguiente).



Habilitar valor sustitutivo "1" en caso de STOP de la CPU

Valor sustitutivo "1" en caso de STOP de la CPU

Si en los parámetros básicos del módulo ha seleccionado la opción "Valor sustitutivo" en "Reacción a STOP de la CPU" y la CPU se para, el módulo transmite a las salidas digitales los valores sustitutivos parametrizados. Active la casilla de verificación correspondiente a DQ0; a continuación se activa la salida Q0 en el caso de parada de la CPU.

En el modo de operación "Dosificación" es posible ajustar hasta cuatro valores sustitutivos, mientras que en los otros modos de contaje puede ajustarse un valor sustitutivo.

Los valores sustitutivos permanecen válidos tras pasar la CPU de STOP a RUN, hasta el siguiente manejo de las salidas digitales.

Si cambia el parámetro básico "Reacción a STOP de la CPU", las salidas se desactivan.

⚠ PRECAUCIÓN
Se pueden producir daños materiales. Los valores sustitutivos parametrizados se aplican siempre que la CPU pasa a STOP, aunque la puerta esté cerrada e incluso a salidas no habilitadas. Hay que parametrizar exclusivamente valores sustitutivos cuya aplicación no pueda ocasionar estados peligrosos.

Habilitación de alarmas de proceso del canal n

Habilitar la activación de una alarma de proceso

Introducción

En el FM 350-2 se puede definir qué eventos deben generar una alarma de proceso durante el funcionamiento.

¿Qué es una alarma de proceso?

Si se debe reaccionar a un evento determinado, independientemente del ciclo de la CPU, el FM 350-2 puede disparar una alarma de proceso. En ese caso la CPU interrumpe el programa cíclico y procesa el OB de alarma de proceso.

Habilitar la alarma de proceso

La habilitación de las alarmas de proceso tiene lugar en dos pasos:

1. En la configuración del hardware las alarmas del módulo se habilitan en los parámetros básicos y se especifica si el módulo debe activar una alarma de diagnóstico y/o una alarma de proceso.
2. En "Habilitar alarma de proceso" es preciso seleccionar los eventos correspondientes:
   - Eventos para la activación de una alarma de proceso en los modos de contaje (Página 9572)
   - Eventos para la activación de una alarma de proceso en los modos de medición (Página 9573)

OB de alarma de proceso, por ejemplo, OB 40

Si ocurre una alarma de proceso, el programa de usuario se interrumpe, los datos se transfieren del módulo a la información de arranque del OB 40 (Página 9573) y se llama al OB 40. La alarma de proceso se acusa saliendo del OB 40.

Si no se ha programado ningún OB de alarma de proceso, la CPU pasará a STOP. Si a continuación se pasa a RUN, se borran las condiciones para la alarma de proceso.

Alarma de proceso perdida

Si se produce un evento que debe provocar una alarma de proceso pero todavía no se ha acusado un evento igual ocurrido anteriormente, no se activa una nueva alarma de proceso y ésta se pierde. Esta situación puede conducir a la alarma de diagnóstico "Alarma de proceso perdida".

Si entre dos eventos que deben disparar una alarma de proceso transcurre un tiempo < 2 ms la segunda alarma de proceso se perderá sin que se haya podido disparar una alarma de diagnóstico.

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de proceso.

Eventos para la activación de una alarma de proceso en los modos de contaje

¿Qué eventos pueden disparar una alarma de proceso en los modos de contaje?

En los modos de contaje, los eventos siguientes pueden disparar una alarma de proceso mientras el FM 350-2 está en funcionamiento:

- Abrir la puerta HW
- Cerrar la puerta HW

- Rebase por exceso/Rebase por defecto
- Alcanzar el valor de comparación
  Registre el valor de comparación correspondiente en el campo de entrada. Los valores de comparación han de estar dentro de los rangos de contaje de los distintos modos operativos; no se soportan los límites del rango de contaje.

Active la casilla de verificación para los eventos correspondientes.

Independientemente del modo de operación parametrizado se pueden seleccionar varios eventos para el disparo de una alarma de proceso. Para las alarmas de proceso disparadas cuando se alcanza un valor de comparación es preciso tener en cuenta las condiciones marco (para "Activar si estado del contador >= valor de comparación (Página 9566)" o "Activar si estado del contador <= valor de comparación (Página 9567)").

Eventos para la activación de una alarma de proceso en los modos de medición

¿Qué eventos pueden disparar una alarma de proceso en los modos de medición?

En los modos de medición, los eventos siguientes pueden disparar una alarma de proceso mientras el FM 350-2 está en funcionamiento:

- Abrir la puerta HW
- Cerrar la puerta HW
- Fin de la medición
- Rebase por defecto del valor límite inferior
  Registre el límite inferior correspondiente en el campo de entrada. Si desea obtener información sobre el rango de valores para los límites inferiores, consulte el apartado "Conceptos básicos sobre la operación de medición (Página 9474)".
- Rebase por exceso del valor límite superior
  Registre el límite superior correspondiente en el campo de entrada. Si desea obtener información sobre el rango de valores para los límites superiores, consulte el apartado "Conceptos básicos sobre la operación de medición (Página 9474)".

Active la casilla de verificación para los eventos correspondientes.

Independientemente del modo de operación parametrizado se pueden seleccionar varios eventos para el disparo de una alarma de proceso.

Información de arranque del OB 40

Información de arranque del OB 40

La variable temporal OB40_POINT_ADDR se escribe en la información de arranque del OB 40.

La variable OB40_POINT_ADDR (byte 8 a 11) abarca 4 bytes. En estos cuatro bytes se guarda información sobre el evento que ha provocado la alarma de proceso.

La siguiente tabla muestra qué bits se activan según el tipo de alarma. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Modo de operación: contaje sin fin, contaje único, contaje periódico

Canal	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	8					Valor de comparación alcanzado	Rebase por exceso/ Rebase por defecto	Cierre de la puerta HW	Puerta HW abierta
1		Valor de comparación alcanzado	Rebase por exceso/ Rebase por defecto	Cierre de la puerta HW	Puerta HW abierta				
2 a 7	9 a 11	Véase byte 8							

Modo de operación: medición de frecuencia, medición de velocidad, medición del período

Canal	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	8					Límite superior rebasado por exceso/límite inferior rebasado por defecto	Fin de la medición	Cierre de la puerta HW	Puerta HW abierta
1		Límite superior rebasado por exceso/límite inferior rebasado por defecto	Fin de la medición	Cierre de la puerta HW	Puerta HW abierta				
2 a 7	9 a 11	Véase byte 8							

Modo de operación: dosificación

Canal	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	8	–	Rebase por exceso/ Rebase por defecto	Valor de comparación 4 alcanzado	Valor de comparación 3 alcanzado	Valor de comparación 2 alcanzado	Valor de comparación 1 alcanzado	Cierre de la puerta HW	Puerta HW abierta
4	10	Véase byte 8							

Poner en marcha FM 350-2

Uso del FM 350-2 desde el programa de usuario

Integración del FM 350-2 en un programa de usuario

Para integrar el FM 350-2 en un programa de usuario pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

Las instrucciones CNT2WRPN, CNT2RDPN y CNT2_CTR se activan en el programa de usuario de la CPU. Éstas llevan a cabo la comunicación entre la CPU y el FM 350-2.

Todos los datos pertenecientes a un canal del módulo se encuentran en el DB de la instrucción CNT2_CTR, CNT2WRPN o CNT2RDPN (DB de contaje (Página 7407)). Inserte en su proyecto un bloque de datos del tipo CNT2_CHANTYPE mediante "Bloques de programa > Agregar bloque".

Además, para el FM 350-2 se puede utilizar también la instrucción DIAG_RD, que permite transferir información de diagnóstico al DB de contaje.

Consulte también

CNT2WRPN (Página 7404)

CNT2RDPN (Página 7405)

CNT2_CTR (Página 7405)

DIAG_RD (Página 7406)

FM 350-2: Online y diagnóstico

Información actual sobre el módulo

Diagnóstico estándar

Indicación de fallos mediante el LED de error de grupo del FM 350-2

¿Dónde se indica el fallo?

Si el LED de error de grupo rojo se ilumina significa que ha ocurrido un error en el módulo (error interno), en los terminales de los cables (error externo), o bien que la parametrización es defectuosa.

¿Qué errores se indican?

La tabla siguiente muestra los tipos de fallos indicados por el LED de error de grupo.

Tipo de fallo	Causa del fallo	Solución
Errores internos	Activación de la vigilancia de tiempo (Watchdog)	Sustituir el módulo
	Alarma de proceso perdida	Acuse mediante el procesamiento de la alarma de proceso
Errores externos	Falta la parametrización del módulo	Definir y transferir la parametrización
	La parametrización contiene fallos	
Fallos en el canal externo	Cortocircuito o sobrecarga en la alimentación del encóder	Corregir la conexión
	Fallos en las señales de un detector NAMUR (rotura de hilo, cortocircuito, falta de línea)	Corregir la conexión

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

¿Qué es una alarma de diagnóstico?

Si se quiere que un programa de usuario responda a un fallo interno o externo, se puede parametrizar una alarma de diagnóstico que interrumpa el programa cíclico de la CPU y llame al OB de alarma de diagnóstico (OB 82).

¿Qué eventos pueden provocar una alarma de diagnóstico?

Los siguientes eventos pueden provocar una alarma de diagnóstico:

- Error en un canal parametrizado
- Falta parametrización del módulo
- Error en la parametrización del módulo
- La vigilancia de tiempo (watchdog) ha saltado
- Alarma de proceso perdida
- La vigilancia de la línea de señales del sensor NAMUR indica errores
- Fallo en la alimentación de 8,2 V de un sensor NAMUR

Habilitar la alarma de diagnóstico

En la configuración del hardware las alarmas del módulo se habilitan en los parámetros básicos y se especifica si el módulo debe activar una alarma de diagnóstico y/o una alarma de proceso.

Respuestas a una alarma de diagnóstico

Cuando se produce un evento que puede disparar una alarma de diagnóstico ocurre lo siguiente:

- La información de diagnóstico se registra en los registros de diagnóstico DS0 y DS1 (Página 9577) del módulo.
- El LED de error de grupo se enciende.
  Una vez eliminado el error, se apaga el LED de error de grupo.
- Se llama al OB de alarma de diagnóstico (OB 82).
- El registro de diagnóstico DS0 se graba en la información de arranque del OB de alarma de diagnóstico.
- El contaje prosigue sin cambios si se ha programado una alarma del OB 82. La CPU pasa a modo STOP si no hay un OB 82.

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de diagnóstico.

Registro de diagnóstico DS0 y DS1

Registro de diagnóstico DS0 y DS1

La información que indica qué evento ha provocado una alarma de diagnóstico se guarda en los registros de diagnóstico DS0 y DS1. El registro de diagnóstico DS0 consta de cuatro bytes; DS1 consta de 16 bytes, siendo los cuatro primeros idénticos a DS0.

Leer registro del módulo

El registro de diagnóstico DS0 (Página 9577) se transfiere automáticamente a la información de arranque en la llamada del OB de diagnóstico. Allí, los cuatro bytes se guardan en los datos locales (bytes 8 a 11) del OB 82.

El registro de diagnóstico DS1 (Página 9578) (y, por lo tanto, también el contenido del DS0) puede leerse con la instrucción DIAG_RD del módulo. Esa lectura sólo tiene sentido si en el DS0 se notifica un fallo en un canal.

Asignación del registro de diagnóstico DS0

Asignación del registro de diagnóstico DS0 y de la información de arranque

La tabla siguiente muestra la asignación del registro de diagnóstico DS0 en la información de arranque. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Byte	Bit	Significado	Nota	Nº de evento
0	0	Fallo en módulo	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Error interno	Se activa para todos los errores internos	8:x:01

Byte	Bit	Significado	Nota	Nº de evento
	2	Error externo	Se activa para todos los errores externos	8:x:02
	3	Fallo en un canal	Si desea obtener más información, consulte DS 1, byte 4 (Página 9578)	8:x:03
	6	Falta parametrización	Realizar la parametrización	8:x:06
	7	Error de parametrización	Para más información, consulte el capítulo Errores de datos (Página 9579)	8:x:07
1	0 ... 3	Clase de tipo	Siempre tiene asignado un 8	
	4	Información de canal	Siempre tiene asignado un 1	
2	3	La vigilancia de tiempo (watchdog) ha saltado	Módulo defectuoso o fuertes interferencias	8:x:33
3	6	Alarma de proceso perdida	Comprobar la configuración. Se ha detectado un evento y no se ha podido notificar porque el mismo evento aún no se ha sido acusado por la CPU	8:x:46

Asignación del registro de diagnóstico DS1

Asignación del registro de diagnóstico DS1

El registro de diagnóstico DS1 se compone de 16 bytes. Los 4 primeros bytes son idénticos a los del registro de diagnóstico DS0. La siguiente tabla muestra la asignación de los bytes restantes. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero. Este registro es introducido por la instrucción DIAG_RD en el DB de la instrucción CNT2_CTR a partir de la DW212.

Byte	Bit	Significado	Nota	Nº de evento
4	0 ... 6	Tipo de canal	Siempre con un 76H asignado	
	7	Otros tipos de canal	Siempre tiene asignado un 0	
5	0 ... 7	Longitud de la información de diagnóstico	Siempre tiene asignado un 8	
6	0 ... 7	Número de canales	Siempre tiene asignado un 8	
7	0	Vector de error de canal	Bit de canal	
8	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 0)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96
9	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 1)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96
10	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 2)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96
11	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 3)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96
12	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 4)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96
13	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 5)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96
14	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 6)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96

Byte	Bit	Significado	Nota	Nº de evento
15	4	Fallo en alimentación de 8,2 V del encóder	Sensores NAMUR	8:x:94
(canal 7)	6	Línea de señales del sensor NAMUR	Cortocircuito/rotura de hilo	8:x:96

Errores de datos

¿Dónde se producen errores de datos?

Cuando se transfieren peticiones al módulo desde la programadora con la instrucción CNT2WRPN (Página 7404), las peticiones se comprueban. Si se producen errores durante esta comprobación, el módulo notifica estos errores.

El módulo no soporta peticiones con errores.

¿Dónde se indican los errores de datos?

Los errores de datos se muestran en la interfaz de parametrización.

Si se detecta un error cuando se están comprobando las peticiones, en el DB de contador se activará el bit CHECKBACK_SIGNALS, DATA_ERR = 1.

¿Cómo se acusan los errores de datos?

Corrija las peticiones según lo preestablecido. Retransfiera las peticiones corregidas al FM 350-2.

Búfer de diagnóstico del FM 350-2

Las entradas del búfer de diagnóstico del FM 350-2 se muestran en la interfaz de parametrización.

Códigos de error en el búfer de diagnóstico

Códigos de error

A continuación, figuran los números de error con sus correspondientes descripciones del error. En el búfer de diagnóstico se introduce un error con una ID de error de las siguientes características:

16#Fx83 yyzz

Significado de los dígitos:

- yy = identificador para la clase de error, p, ej. 04 para un error de parametrización de canal.
- zz = número de error, p. ej. 18 para el error "sentido principal de contaje no válido"

Error de parametrización de canal	Número de error	Descripción
04	01	No está permitida la petición porque el canal está ocupado para dosificación
04	18	Sentido principal de contaje no válido
04	19	Rango de contaje erróneo
04	1A	Valor de comparación fuera del rango de contaje disponible
04	1B	Comportamiento de salida no válido
04	1C	Imposible habilitar la alarma de proceso
04	1D	Parametrización de la vigilancia de hardware no válida
04	1E	Parametrización de la puerta HW no válida
04	1F	No es posible la selección de señal
04	20	Selección de señal errónea
04	21	Valor de histéresis demasiado grande
04	22	Valor sustitutivo no válido
04	23	Reacción no válida al cerrar una puerta
04	24	Parámetro reservado diferente de 0
04	25	Parámetro reservado diferente de 0
04	26	Parámetro reservado diferente de 0
04	27	Selección de encóder errónea
04	28	Diagnóstico de la entrada de contaje sólo posible para sensor NAMUR
04	29	Valor de carga fuera del rango de contaje válido
04	2B	Imposible ampliar el rango de contaje
04	2C	Valor de comparación 1 fuera del rango de contaje
04	2D	Valor de comparación 2 fuera del rango de contaje
04	2E	Valor de comparación 3 fuera del rango de contaje
04	2F	Valor de comparación 4 fuera del rango de contaje
04	30	Valor límite inferior para la frecuencia o la velocidad no válido
04	31	Valor límite superior para la frecuencia o la velocidad no válido
04	32	Valor límite inferior >= valor límite superior
04	33	Tiempo de integración no válido
04	34	Impulsos por vuelta no válidos
04	35	Valor límite inferior para la duración del período no válido
04	36	Valor límite superior para la duración del período no válido

Error de parametrización de módulo	Número de error	Descripción
05	0A	Generación de alarma no válida
05	0B	Selección de alarma no válida
05	0C	Reacción no válida con output disable activado
05	0D	Parámetro reservado diferente de 0
05	18	Indicación de fecha y hora no válida

Error de parametrización de módulo	Número de error	Descripción
05	19	Selección de señal de encóder no válida
05	1A	Modo de operación de canal no válido
05	1B	Distribución de CHECKBACK_SIGNALS.USER_STAT en bloque de datos no válida

Error de datos	Número de error	Descripción
06	18	Sentido principal de contaje no válido
06	19	Rango de contaje erróneo
06	1A	Valor de comparación fuera del rango de contaje disponible
06	1B	Comportamiento de salida no válido
06	1C	Imposible habilitar la alarma de proceso
06	1D	Parametrización de la vigilancia de hardware no válida
06	1E	Parametrización de la puerta HW no válida
06	1F	No es posible la selección de señal
06	20	Selección de señal errónea
06	21	Valor de histéresis demasiado grande
06	22	Valor sustitutivo no válido
06	23	Reacción no válida al cerrar una puerta
06	24	Parámetro reservado diferente de 0
06	25	Parámetro reservado diferente de 0
06	26	Parámetro reservado diferente de 0
06	27	Selección de encóder errónea
06	28	Diagnóstico de la entrada de contaje sólo posible para sensor NAMUR
06	29	Valor de carga fuera del rango de contaje válido
06	2B	Imposible ampliar el rango de contaje
06	2C	Valor de comparación 1 fuera del rango de contaje
06	2D	Valor de comparación 2 fuera del rango de contaje
06	2E	Valor de comparación 3 fuera del rango de contaje
06	2F	Valor de comparación 4 fuera del rango de contaje
06	30	Valor límite inferior para la frecuencia o la velocidad no válido
06	31	Valor límite superior para la frecuencia o la velocidad no válido
06	32	Valor límite inferior >= valor límite superior
06	33	Tiempo de integración no válido
06	34	Impulsos por vuelta no válidos
06	35	Valor límite inferior para la duración del período no válido
06	36	Valor límite superior para la duración del período no válido

Aviso	Número de error	Descripción
0F	01	Fin del arranque del módulo
0F	02	Ningún parámetro en el módulo
0F	03	Módulo está parametrizado por defecto
0F	04	Módulo está parametrizado
0F	05	Modo PG ajustado
0F	06	Modo PG deseleccionado
0F	07	Modo PG ha fallado
0F	08	Modo PG está ajustado, no se permite ninguna petición de la CPU
0F	09	Secuencia de petición errónea

Error de diagnóstico	Número de error	Descripción
80	04	Falta tensión auxiliar externa
80	05	Falta conector frontal
80	06	Módulo no parametrizado
80	07	Parámetro erróneo en el módulo
80	30	Falta módulo personalizable o es incorrecto
80	31	Fallo de comunicación
80	32	Estado operativo STOP
80	33	La vigilancia de tiempo ha reaccionado
80	34	Falla la alimentación interna del módulo
80	35	Pila 1 agotada
80	36	Falla todo respaldo
80	41	Falla el procesador
80	42	Error EPROM
80	43	Error de RAM
80	44	Error ADU/DAU
80	45	Fallo de fusible
80	46	Se ha perdido una alarma de proceso
80	90	Señal de encóder A en canal errónea
80	91	Señal de encóder B en canal errónea
80	92	Señal de encóder N en canal errónea
80	93	Valor erróneo entre los canales
80	94	Alimentación de encóders 5,2 V o NAMUR en canal errónea
80	95	Fallo en alimentación de 24 V del encóder
80	96	Sensor NAMUR cortocircuito/rotura de hilo en canal

Uso del FM 450-1

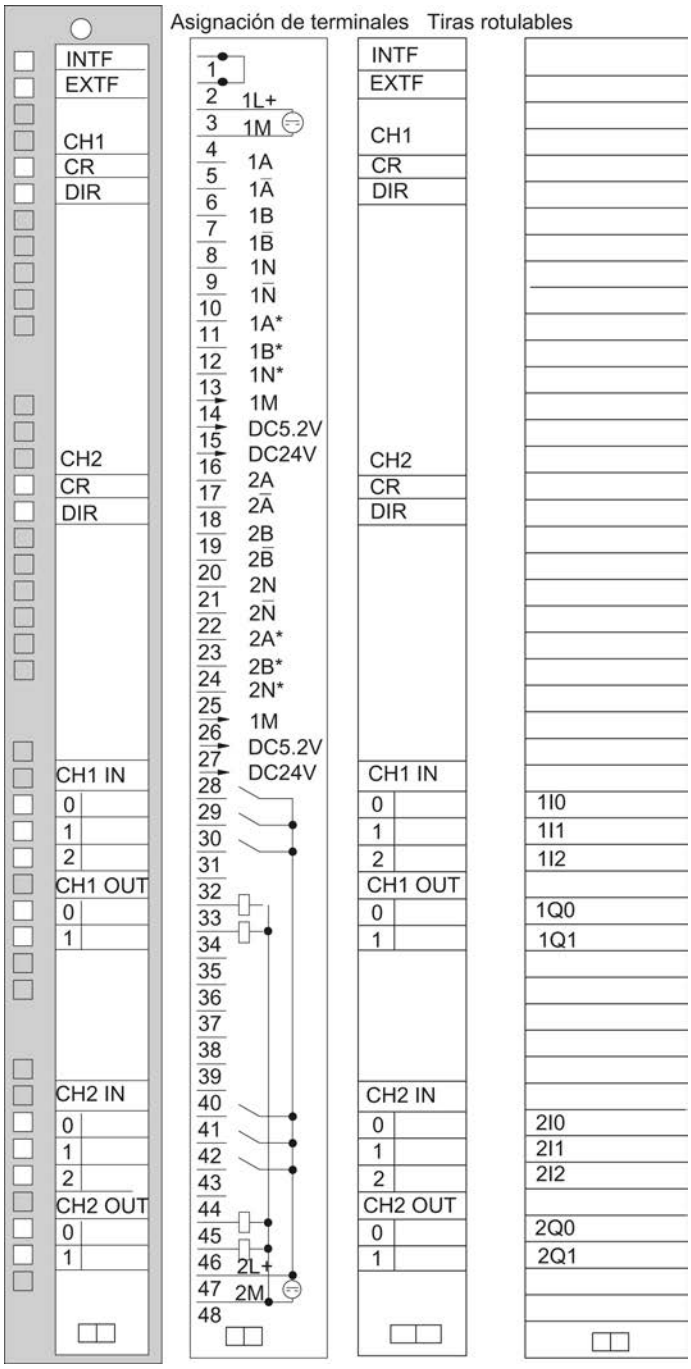
Asignación de conectores del conector frontal del FM 450-1

Conector frontal

En el conector frontal de 48 polos se conectan:

- Señales de contaje
- Entradas digitales
- Salidas digitales
- Alimentación del encóder
- Tensión auxiliar y de carga

La figura siguiente muestra la parte delantera del conector frontal, la tira con la indicación de la asignación de terminales y las tiras rotulables.


Asignación de terminales
Tiras rotulables
INTF
EXTF
CH1
CR
DIR
CH2
CR
DIR
CH1 IN
0
1
2
CH1 OUT
0
1
CH2 IN
0
1
2
CH2 OUT
0
1
1L+
1M
1A
1Ā
1B
1B̄
1N
1N̄
1A*
1B*
1N*
1M
DC5.2V
DC24V
2A
2Ā
2B
2B̄
2N
2N̄
2A*
2B*
2N*
1M
DC5.2V
DC24V
2L+
2M
1I0
1I1
1I2
1Q0
1Q1
2I0
2I1
2I2
2Q0
2Q1

Asignación de terminales del conector frontal

La tabla siguiente muestra la asignación de terminales del conector frontal del FM 450-1.

Terminal	Nombre	Entrada/ salida	Función			
			Encóder de 5 V RS 422, simétrico	Encóder de 24 V asimétrico	Generador de impulsos de 24 V con nivel de sentido	Detector de proximidad de 24 V
1			-			
2			-			
3	1L+	Entrada	Tensión auxiliar de 24 V para alimentación del encóder			
4	1M	Entrada	Masa tensión auxiliar para alimentación del encóder			
Contador 1						
5	1 A	Entrada	Señal del encóder A	-		
6	1 /A	Entrada	Señal del encóder / A	-		
7	1 B	Entrada	Señal del encóder B	-		
8	1 /B	Entrada	Señal del encóder / B	-		
9	1 N	Entrada	Señal del encóder N	-		
10	1 /N	Entrada	Señal del encóder / N	-		
11	1 A*	Entrada	-	Señal del encóder A*		
12	1 B*	Entrada	-	Señal del encóder B*	Señal de sentido	-
13	1 N*	Entrada	-	Señal del encóder N*	-	
14	1M	Salida	Masa alimentación del encóder			
15	5,2 V DC	Salida	Alimentación de 5,2 V del encóder	-		
16	24 V DC	Salida	-	Alimentación del encóder 24 V		
Contador 2						
17	2 A	Entrada	Señal del encóder A	-		
18	2 /A	Entrada	Señal del encóder / A	-		
19	2 B	Entrada	Señal del encóder B	-		
20	2 /B	Entrada	Señal del encóder / B	-		
21	2 N	Entrada	Señal del encóder N	-		
22	2 /N	Entrada	Señal del encóder / N	-		
23	2 A*	Entrada	-	Señal del encóder A*		

Terminal	Nombre	Entrada/ salida	Función			
			Encóder de 5 V RS 422, simétrico	Encóder de 24 V asimétrico	Generador de impulsos de 24 V con nivel de sentido	Detector de proximidad de 24 V
24	2 B*	Entrada	-	Señal del encóder B*, señal de sentido		
25	2 N*	Entrada	-	Señal del encóder N*		
26	1M	Salida	Masa alimentación del encóder			
27	5,2 V DC	Salida	Alimentación de 5,2 V del encóder	-		
28	24 V DC	Salida	-	Alimentación del encóder 24 V		
Contador 1						
29	1I0	Entrada	Entrada digital 1I0			
30	1I1	Entrada	Entrada digital 1I1			
31	1I2	Entrada	Entrada digital 1I2 (activar contador)			
32			-			
33	1Q0	Salida	Salida digital 1Q0			
34	1Q1	Salida	Salida digital 1Q1			
35			-			
36			-			
37			-			
38			-			
39			-			
40			-			
Contador 2						
41	2I0	Entrada	Entrada digital 2I0			
42	2I1	Entrada	Entrada digital 2I1			
43	2I2	Entrada	Entrada digital 2I2 (activar contador)			
44			-			
45	2Q0	Salida	Salida digital 2Q0			
46	2Q1	Salida	Salida digital 2Q1			
47	2L+	Entrada	Tensión de carga 24 V para las entradas y salidas digitales			
48	2M	Entrada	Masa tensión de carga para las entradas y salidas digitales			

Nota

Las conexiones para las entradas de contaje (alimentación del encóder, señales del encóder) están unidas galvánicamente con masa de la CPU. Por eso debe conectar el borne 4 (1M) con la masa de la CPU con una baja impedancia.

Si los encóders se alimentan externamente, también se deberá conectar la masa de dicha tensión externa con masa de la CPU.

Encontrará más información sobre la estructura y el cableado en el manual de producto (http://support.automation.siemens.com/WW/view/es/1118412) del módulo de contaje FM 450-1.

Tensión auxiliar 1L+, 1M

Para la alimentación de tensión de los encóders de 5 V y 24 V debe conectar una tensión continua de 24 V en las conexiones 1L+ y 1M.

Un diodo integrado protege el módulo contra inversión de polaridad de la tensión auxiliar.

El módulo vigila si está conectada la tensión auxiliar.

Alimentación de encóder de 5,2 V DC

El módulo genera una tensión de 5,2 V a partir de la tensión auxiliar 1L+/1M con una intensidad máxima de 300 mA por canal de contaje, disponible en el conector "DC5.2V" para la alimentación de un encóder de 5 V a prueba de cortocircuitos.

Alimentación de encóder de 24V DC

Para la alimentación de tensión de 24 V de un encóder, se pone a disposición la tensión 1L+/1M en la salida DC24V a prueba de cortocircuitos. Se comprueba si hay cortocircuitos en la alimentación del encóder.

Señales de encóder de 5 V A y /A, B y /B, N y /N

A cada contador se pueden conectar encóders incrementales con señales diferenciales de 5 V conforme a RS 422, es decir, encóders incrementales con las señales diferenciales A y /A, B y /B, N y /N.

Las señales A y /A, B y /B, N y /N se conectan a través de los terminales correspondientes.

Las señales N y /N sólo se conectan si se desea poner el contador a la marca cero del encóder.

Las entradas no disponen de un aislamiento galvánico con el bus de la S7-400.

Señales de encóder de 24 V A*, B* y N*

Las señales de 24 V se identifican con las letras A*, B* y N*.

Puede conectar tres tipos de encóder diferentes a cada contador:

- Encóders incrementales con señales de 24 V:
  Las señales A*, B* y N* se conectan a través de los terminales correspondientes.
- Generador de impulsos sin nivel de sentido:
  La señal se conecta en la conexión A*.
- Generador de impulsos con nivel de sentido:
  La señal de contaje se conecta al terminal A*. El nivel de sentido se conecta al terminal B*.

Las entradas no disponen de un aislamiento galvánico con el bus de la S7-400.

Filtro de entrada para señales de encóder de 24 V

Para eliminar fallos, puede parametrizar filtros de entrada (componentes RC) con un tiempo de filtrado unitario para las entradas de 24 V A*, B* y N*. Tiene a su disposición los dos filtros de entrada siguientes para cada contador:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Frecuencia de contaje máxima	200 kHz	20 kHz
Ancho de impulso mínimo de las señales de contaje	2,5 µs	25 µs

Entradas digitales

Las entradas digitales 1I0 y 1I1 se pueden utilizar para controlar las puertas del contador 1. Las entradas digitales 2I0 y 2I1 se pueden utilizar para controlar las puertas del contador 2. Las puertas se pueden utilizar tanto con control por nivel como con control por flanco.

La entrada digital 1I2 sirve para ajustar el contador 1 al valor de carga.
La entrada digital 2I2 sirve para ajustar el contador 2 al valor de carga.

Las entradas digitales funcionan con una tensión nominal de 24 V.

Las entradas digitales están separadas galvánicamente del bus posterior del S7-400 y de las entradas del contador.

Filtro de entrada para entradas digitales

Para eliminar fallos, puede parametrizar filtros de entrada (componentes RC) con un tiempo de filtrado unitario para las entradas digitales 1I0, 1I1 y 1I2 ó bien 2I0, 2I1 y 2I2 Se dispone de los dos filtros de entrada siguientes:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Ancho de impulso mínimo de las señales de entrada	2,5 µs	25 µs

Salidas digitales

Para iniciar directamente procesos de control, el FM 450-1 dispone de las salidas digitales 1Q0 y 1Q1 (para el contador 1) o bien 2Q0 y 2Q1 (para el contador 2).

Las salidas digitales se alimentan con la tensión de carga 2L+.

Las salidas digitales están separadas galvánicamente del bus posterior del S7-400 y de las entradas del contador.

Las salidas digitales son salidas tipo PNP y pueden recibir una corriente de carga de 0,5 A. Están protegidas contra sobrecarga y cortocircuito.

Nota

Se pueden conectar directamente relés y contactores sin necesidad de circuitería externa.

El comportamiento temporal de las salidas digitales varía en función de la parametrización.

Tensión de carga 2L+/2M

Para la alimentación de las salidas digitales 1Q0 y 1Q1 ó bien 2Q0 y 2Q1 es preciso conectar una tensión de carga de 24 V en los bornes 2L+ y 2M del módulo.

Un diodo integrado protege el módulo contra inversión de polaridad de la tensión de carga.

El FM 450-1 no vigila la tensión de carga 2L+/2M.

Configurar FM 450-1

Parametrizar FM 450-1

Especificar canal

Copiar canal

Este parámetro permite copiar un canal en otro canal.

1. Seleccione el canal cuyos parámetros desee copiar (origen del canal).
2. Active el canal en el que desee copiar los parámetros (objetivo del canal).
3. Pulse el botón "Copiar".

Modos de operación del canal n

Seleccionar el modo de operación

Descripción

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el módulo de contaje.

Modo de operación	Descripción
Contaje sin fin (Página 9468) (con o sin puerta)	El FM 450-1 cuenta en modo sin fin a partir del estado actual del contador.
Contaje único (Página 9469) (con puerta HW o SW)	El FM 450-1 comienza a contar a partir del valor de carga al abrirse la puerta y se detiene en el límite de contaje.
Contaje periódico (Página 9470) (con puerta HW o SW)	El FM 450-1 comienza a contar al abrirse la puerta a partir del valor de carga entre los límites de contaje.

El ajuste predeterminado es el modo de operación "Contaje sin fin".

Seleccionar el control de puerta

Descripción

Al preseleccionar un control de puerta se elige cómo se realizará el contaje en el modo de operación seleccionado.

- Sin puerta
  Todas las puertas están abiertas.
- Puerta SW (Página 9488)
  La puerta SW se abre y se cierra con el parámetro de entrada/bit de control SW_GATE.
- Puerta HW (Página 9488), con control de nivel o de flanco
  La puerta HW se abre y se cierra con las señales correspondientes en las entradas digitales

El ajuste predeterminado es "Sin puerta".

Seleccionar el rango de contaje

Descripción

El módulo incorpora un registro de contaje de 32 bits. El ajuste “Rango de contaje” permite seleccionar si el módulo debe contar sólo en el rango positivo, o bien, si el bit 32 debe interpretarse como bit de signo, permitiendo de este modo la representación de números negativos.

Seleccione el rango de contaje deseado.

Encontrará más información en "Rangos de contaje y límites de contaje (Página 9465)".

Rebase del límite superior, rebase por defecto del límite inferior y paso por cero

En los límites de contaje, si se produce un rebase por exceso o por defecto se activa un bit en el DB de la instrucción CNT_CTRL.

Al producirse un paso por cero en el rango de contaje de -31 a +31 bits, también se activa un bit en el DB.

Al producirse un paso por cero en el rango de contaje de 0 a +32 bits se indica además un rebase por exceso, o bien un rebase por defecto (dependiendo del sentido de contaje).

Evento	Bit de estado en el DB
Rebase del límite superior	STS_OFLW se activa.
Rebase del límite inferior	STS_UFLW se activa.
Paso por cero	STS_ZERO se activa.

Disparo de alarmas de proceso

También puede notificar los eventos rebase por exceso, rebase por defecto y paso por cero con alarmas de proceso.

Encóders del canal n

Seleccionar el tipo de señal del encóder

Seleccionar tipo de señal

1. Seleccione el tipo de señal del encóder utilizado.
   Dependiendo de su selección y del modo de operación se bloquearán aquellas opciones no permitidas para ese determinado encóder. Las opciones bloqueadas se identifican con letra gris en la correspondiente ventana de selección.
2. Seleccione las opciones que desee utilizar con el encóder seleccionado.

Es posible selecciona un tipo de señal de uno de los encóders siguientes:

- Encóder: incremental de 5 V (Página 9591) (5 V señal diferencial)
- Sensor: incremental de 24 V (Página 9592) (24 V pista A+B desfasado)
- Encóder: 24 V Impulso y sentido (Página 9592)
- Encóder: 24 V Detector de proximidad (Página 9593)

Encóder: incremental de 5 V

Encóder incremental de 5 V

Seleccione el tipo de señal "5 V Señal diferencial" si utiliza un encóder incremental de 5 V con señales diferenciales A, /A, B, /B y N, /N.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal (Página 9494)	Simple	Sólo se evalúa un flanco de señal A.
	Doble	Se evalúan ambos flancos de la señal A.
	Cuádruple	Se evalúan ambos flancos de las señales A y B.
Sentido de contaje (Página 9492)	Normal	Para adaptar el sentido de contaje al proceso
	Invertido	Para adaptar el sentido de contaje al proceso
Frecuencia máx. de contaje	–	La frecuencia de contaje máxima para este encóder es 500 kHz.
Entradas del encóder	–	Sin relevancia
Vigilancia (Página 9492)	A+B+N	Se vigila cortocircuito y rotura de hilo en todos los pares de señal.
	A+B	Se vigilan los pares de señal A y B.
	A	Sólo se vigila el par de señal A.
	Ninguna	No se vigila ningún par de señal.

Encóder: incremental de 24 V

Encóder incremental de 24 V

Seleccione el tipo de señal "24 V pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A*, B* y N*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal (Página 9494)	Simple	Sólo se evalúa un flanco de señal A.
	Doble	Se evalúan ambos flancos de la señal A.
	Cuádruple	Se evalúan ambos flancos de las señales A y B.
Sentido de contaje (Página 9493)	Normal	Para adaptar el sentido de contaje al proceso
	Invertido	Para adaptar el sentido de contaje al proceso
Frecuencia máx. de contaje (Página 9594)	200 kHz	En las entradas se aplica un filtro con un retardo de 1 µs.
	20 kHz	En las entradas se aplica un filtro con un retardo de 15 µs.
Entradas del encóder (Página 9593)	Salida tipo NPN	El encóder conmuta a 0 V (masa).
	Salida tipo PNP/ simétrica	El encóder conmuta a 24 V.
Vigilancia	–	Sin relevancia

Encóder: impulso y sentido de 24 V

Encóder incremental de posición de 24 V

Seleccione el tipo de señal "Impulso y sentido de 24 V" si desea utilizar un encóder incremental de posición de 24 V con la señal de contaje A* y la señal de sentido B*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal	–	Sin relevancia
Sentido de contaje	–	Sin relevancia
Frecuencia máx. de contaje (Página 9594)	200 kHz	En las entradas se aplica un filtro con un retardo de 1 µs.
	20 kHz	En las entradas se aplica un filtro con un retardo de 15 µs.
Entradas del encóder (Página 9593)	Salida tipo NPN	El encóder conmuta a 0 V (masa).
	Salida tipo PNP/ simétrica	El encóder conmuta a 24 V.
Vigilancia	–	Sin relevancia

Encóder: detector de proximidad de 24 V

Detector de proximidad de 24 V

Seleccione el tipo de señal "Detector de proximidad de 24 V" si desea utilizar un encóder incremental de posición de 24V con la señal de contaje A*.

Para este encóder se pueden seleccionar las siguientes opciones:

Parámetro	Opción	Significado
Interpretación de la señal	–	Sin relevancia
Sentido de contaje	–	Sin relevancia
Frecuencia máx. de contaje (Página 9594)	200 kHz	En las entradas se aplica un filtro con un retardo de 1 µs.
	20 kHz	En las entradas se aplica un filtro con un retardo de 15 µs.
Entradas del encóder (Página 9593)	Salida tipo NPN	El encóder conmuta a 0 V (masa).
	Salida tipo PNP/ simétrica	El encóder conmuta a 24 V.
Vigilancia	–	Sin relevancia

Parametrización de las entradas de encóder

Parametrización de las entradas de encóder

El sentido de contaje está predeterminado por la parametrización de las entradas del encóder. La tabla muestra el cambio del sentido de contaje en función de la parametrización.

Parametrización	Borne de conexión B*	Sentido de contaje
Salida tipo NPN	Sin conexión	Atrás
	Cortocircuitado a masa	Adelante

Parametrización	Borne de conexión B*	Sentido de contaje
Salida tipo PNP/simétrica	Sin conexión	Adelante
	24 V conectado	Atrás

En la parametrización seleccione como encóder "24 V Impulso y sentido".

Con estas señales de contaje no es posible invertir el sentido mediante inversión de la señal B*.

Nota

Si las señales de contaje son oscilantes, en este tipo de evaluación el valor de contaje puede "desaparecer" en el flanco, ya que todas las señales se suman.

Filtro de entrada para entradas de contaje de 24 V

Filtro de entrada para entradas de contaje de 24 V

Para suprimir interferencias puede parametrizar filtros de entrada con un tiempo de filtrado uniforme para las entradas de contaje de 24 V A*, B* y N* y para las entradas digitales. Existen los siguientes filtros de entrada:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Frecuencia de contaje máxima	200 kHz	20 kHz
Ancho de impulso mínimo de las señales de contaje	2,5 µs	25 µs

Entradas del canal n

Seleccionar señales de puerta

Control de la puerta HW

Para el control de la puerta HW se pueden seleccionar las siguientes opciones:

- Puerta de hardware controlada por nivel (con nivel en la entrada 1I0 o 2I0)
- Puerta de hardware controlada por flanco (para abrir con flanco en la entrada 1I0 o 2I0); para cerrar con flanco en la entrada 1I1 o 2I1.

Encontrará más información en el apartado "Puerta de hardware (Página 9488)".

Seleccionar el ancho de impulso mínimo

Ancho de impulso mínimo

Para eliminar fallos, es posible parametrizar filtros de entrada (componentes RC) con un tiempo de filtrado unitario para las entradas de 24 V A*, B* y N* y para las entradas digitales.

Se encuentran disponibles dos filtros de entrada con los siguientes datos:

Características	Filtro de entrada 1 (predeterminado)	Filtro de entrada 2
Retardo de entrada típico	1 µs	15 µs
Frecuencia de contaje máxima	200 kHz	20 kHz
Ancho de impulso mínimo de las señales de contaje	2,5 µs	25 µs

Inicializar contador

Inicializar contador

Resumen

Si desea iniciar el contaje a partir de un determinado valor (el valor de carga), deberá parametrizar la señal con la que se debe ajustar el contador al valor de carga. El contador se puede ajustar de las maneras siguientes:

- Con el parámetro de entrada L_DIRECT de la instrucción CNT_CTRL
- Con una señal externa, bien sea a través de la entrada 1I2 (2I2), o bien a través de la entrada en combinación con la señal de paso por cero del encóder

Valor de carga

Como valor de carga se puede predeterminar un número cualquiera comprendido en el rango de contaje.

El valor de carga se interpreta conforme al rango de contaje seleccionado. Si, por ejemplo, se predetermina FFFF FFFF H como valor de carga, este valor se interpretará como 4.294.967.295 en el rango de contaje "0 a +32 bits", o bien como -1 en el rango de contaje "-31 a +31 bits".

Introduzca el valor de varga en el DB de la instrucción CNT_CTRL.

Los rangos de valores siguientes están permitidos para el valor de carga:

Rango de los valores de carga	
Límite inferior	Límite de contaje inferior máx.
Límite superior	Límite de contaje superior máx.

Ajustar el contador mediante una señal externa

Es posible seleccionar dos señales externas distintas para ajustar el contador al valor de carga:

- Sólo entrada 1I2 (2I2) (Página 9596)
- Entrada 1I2 (2I2) y señal de paso por cero (Página 9598) del encóder

La marca cero del encóder se utiliza si desea sincronizar el contador a un estado específico del mismo en un determinado punto del proceso. Gracias a ello se logra una mayor precisión en el contaje.

El contador se ajusta independientemente del modo de contaje.

Tras haber ajustado el contador mediante una señal externa, se activa el bit STS_SYNC en el DB.

Nota

La sincronización de un contador con la marca cero sólo tiene sentido si está abierta la puerta.

Si ha habilitado sólo un sentido de contaje al ajustar el contador mediante una señal externa, tenga en cuenta que – cuando se cierre la puerta – se guardará (congelará) el sentido de contaje actual. Ello permite sincronizar el contador en el sentido de contaje contrario al sentido habilitado.

Activar contador con entrada 1I2 (2I2)

Activar contador con entrada 1I2 (2I2)

El valor de carga se puede cargar en el contador con un flanco ascendente en la entrada 1I2 (2I2) (DI-Set).

Las variables ENSET_UP y ENSET_DN en el DB de la instrucción CNT_CTRL y la parametrización permiten definir el comportamiento del FM 450-1 ante un flanco ascendente en la entrada 1I2 (2I2).

Parámetros	Comportamiento del FM 450-1
ENSET_UP activado	El contador se activa sólo al contar hacia delante.
ENSET_DN activado	El contador se activa sólo al contar atrás.
ENSET_UP y ENSET_DN activados	El contador se activa al contar hacia delante y hacia atrás.
Parametrización "Ajuste único del contador"	El contador sólo se ajusta al ocurrir el primer flanco ascendente en la entrada 1I2 (2I2). Si desea volver a activar el contador, primero es preciso activar de nuevo ENSET_UP o ENSET_DN. Entonces, el contador se ajustará de nuevo con el siguiente flanco ascendente en la entrada 1I2 (2I2).
Parametrización "Ajuste múltiple del contador"	El contador se ajusta en cada flanco ascendente de la entrada 1I2 (2I2) mientras estén activadas ENSET_UP y/o ENSET_DN.

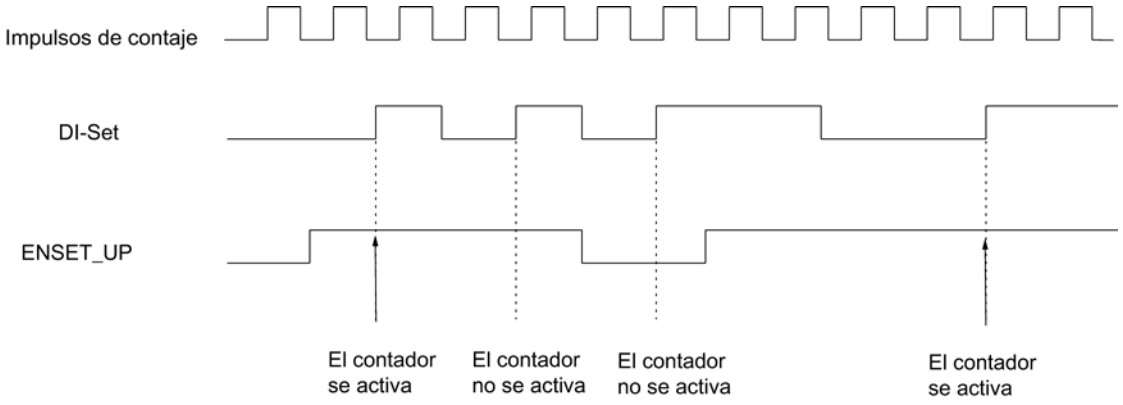
Nota

En todo caso, deberá activar una de las variables ENSET_UP y/o ENSET_DN para que el contador se pueda ajustar con la entrada 1I2 (2I2).

Activación única con la entrada 1I2 (2I2)

La figura siguiente muestra un ajuste único del contador 1 con la entrada 1I2. Para el contador 2 se aplican condiciones similares con la entrada 2I2. En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se activa al contar hacia delante.

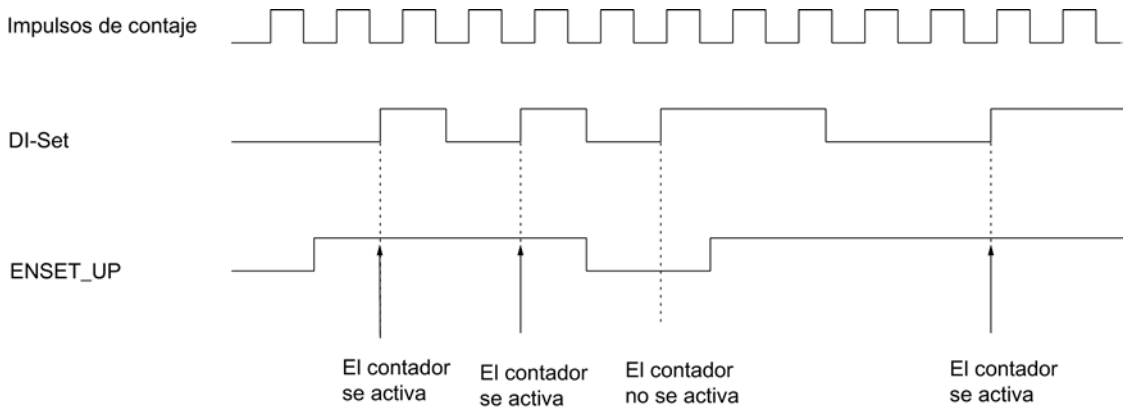
El contador 1 se ajusta en el primer flanco ascendente de la entrada 1I2 mientras esté activada ENSET_UP. Si se desea volver a activar el contador, en primer lugar es preciso desactivar la variable ENSET_UP y activarla luego de nuevo. El siguiente flanco ascendente en la entrada 1I2 implica un nuevo ajuste del contador.



Activación múltiple con la entrada 1I2 (2I2)

La figura siguiente muestra un ajuste múltiple del contador 1 con la entrada 1I2. Para el contador 2 se aplican condiciones similares con la entrada 2I2. En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se activa al contar hacia delante.

El contador 1 se activa en cada primer flanco ascendente de la entrada 1I2 mientras esté activada ENSET_UP. Si desactiva ENSET_UP, el contador no se activará con un flanco ascendente en la entrada 1I2. Sólo cuando se haya vuelto a activar ENSET_UP, el contador se activará en el siguiente flanco ascendente que ocurra en la entrada 1I2.



Activar contador con entrada 1I2 (2I2) y señal de paso por cero

Activar contador con entrada 1I2 (2I2) y señal de paso por cero

Si la activción del contador se parametriza con la señal de paso por cero del encóder ("Evaluar señal de paso por cero para el ajuste"), el contador se activa al valor de carga con el flanco ascendente de la señal de paso por cero.

La activación se realiza sólo si, en el instante del flanco ascendente de la señal de paso por cero, está activa además la entrada 1I2 (2I2).

Las variables ENSET_UP y ENSET_DN en el DB de la instrucción CNT_CTRL y la parametrización permiten definir el comportamiento del FM 450-1 ante un flanco ascendente de la señal de paso por cero:

Parámetros de entrada	Comportamiento del FM 450-1
ENSET_UP activado	El contador se activa sólo al contar hacia delante.
ENSET_DN activado	El contador se activa sólo al contar atrás.
ENSET_UP y ENSET_DN activados	El contador se activa al contar hacia delante y hacia atrás.
Parametrización "Ajuste único del contador"	El contador sólo se ajusta al ocurrir el primer flanco ascendente en la marca cero. Si desea volver a activar el contador, deberá activar primero de nuevo ENSET_UP o ENSET_DN (evaluación de flancos). Entonces, el contador se ajustará de nuevo con el siguiente flanco ascendente en la marca cero.
Parametrización "Ajuste múltiple del contador"	El contador se activa en cada flanco ascendente de la señal de paso por cero mientras que estén activadas ENSET_UP y/o ENSET_DN.

Nota

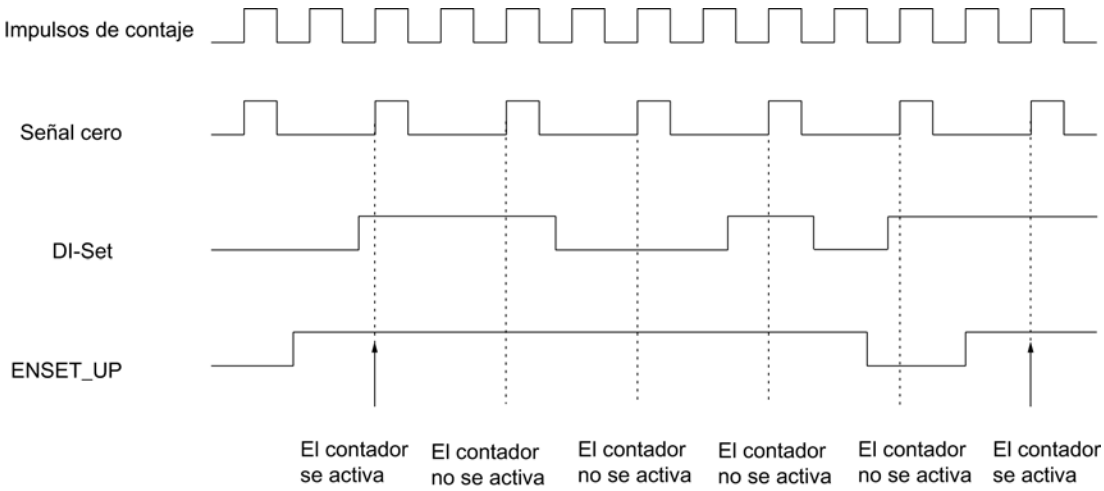
En todo caso, es preciso activar una de las variables ENSET_UP y/o ENSET_DN y la entrada 1I2 (2I2) para que el contador se pueda activar con la señal de paso por cero.

Activación única con entrada 1I2 (2I2) y señal de paso por cero

La figura siguiente muestra un ajuste único del contador 1 con la señal de paso por cero. Para el contador 2 se aplican condiciones similares con la entrada 2I2. En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se activa al contar hacia delante.

El contador 1 se activa con el primer flanco ascendente de la señal de paso por cero, siempre y cuando estén activadas ENSET_UP y la entrada 1I2.

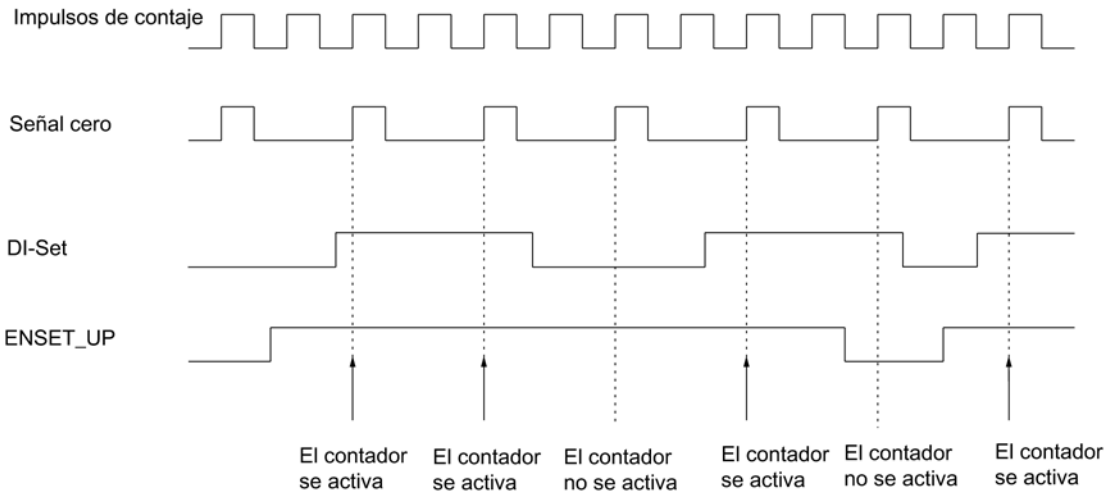
Si desea volver a activar el contador 1, en primer lugar es preciso desactivar la variable ENSET_UP y luego activarla de nuevo. Si no se ha activado la entrada 1I2, la activación del contador se realiza con la primera señal de paso por cero después de activar la entrada 1I2. Si la entrada 1I2 está activada, la activación se efectuará con la siguiente señal de paso por cero.



Activación múltiple con entrada 1I2 (2I2) y señal de paso por cero

La figura siguiente muestra una activación múltiple del contador 1 con la señal de paso por cero. Para el contador 2 se aplican condiciones similares con la entrada 2I2. En el caso ilustrado sólo está activada la variable ENSET_UP. Por tanto, el contador se activa al contar hacia delante.

El contador se activa en cada flanco ascendente de la señal de paso por cero, siempre y cuando estén activadas ENSET_UP y la entrada 1I2.



Salidas del canal n

Definir duración del impulso

Duración del impulso

Es posible definir una duración del impulso para adaptarse a los actuadores utilizados en el proceso (contactores, elementos finales de regulación, etc.). La duración del impulso indica cuánto tiempo están activas las salidas Q0 y Q1 al alcanzarse un valor de comparación.

Con la duración de impulsos se define el tiempo mínimo que debe estar activada la salida. Este ajuste sólo es efectivo si preselecciona el correspondiente comportamiento de la salida. La duración de impulsos no es efectiva si la salida debe permanecer activada entre el valor de comparación y rebase por exceso o por defecto.

La duración del impulso comienza cuando se activa la salida. La imprecisión de la duración del impulso es inferior a 1 ms.

Para la duración del impulso se puede predefinir un valor comprendido entre 0 y 500 ms. Este valor es válido para ambas salidas.

Si la duración del impulso = 0 ms, la salida se activará al alcanzarse el valor de comparación y se desactivará con el siguiente impulso de contaje.

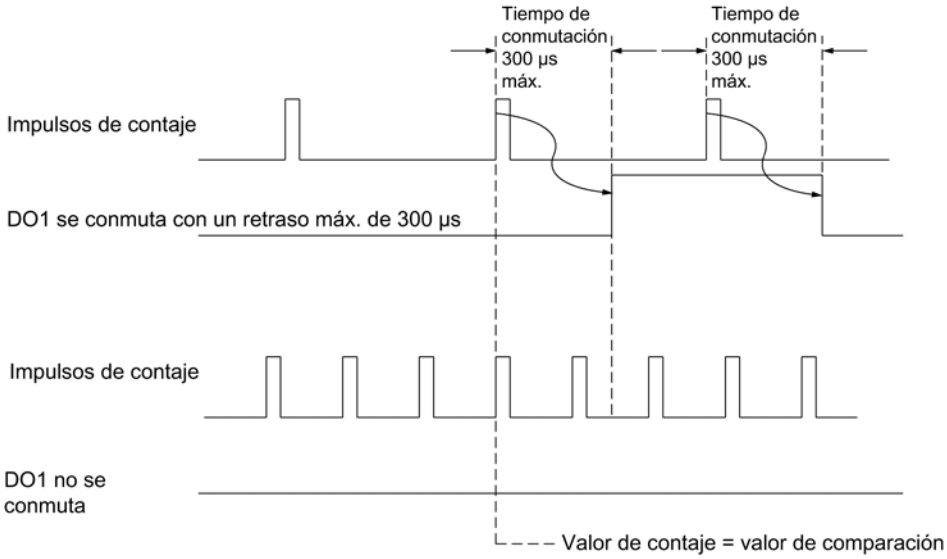
El valor predeterminado para la duración de impulsos es 0.

Nota

Si se predetermina una duración del impulso = 0 ms, la salida permanecerá activada hasta que el estado del contador sea igual al valor de comparación (véase la parte superior de la figura siguiente).

Debido a los tiempos de conmutación de las salidas digitales (máx. 300 µs) se podrían perder impulsos de control en las salidas si el intervalo de tiempo entre los impulsos de contaje es menor que los tiempos de conmutación de las salidas digitales (véase la parte inferior de la figura siguiente).

Por tanto, asegúrese de que el intervalo de tiempo entre los impulsos de contaje sea mayor que los tiempos de conmutación de las salidas digitales.



Definir el comportamiento de las salidas

Valores de comparación y salidas

Descripción

En el FM 450-1 es posible definir para cada contador dos valores de comparación (valor de comparación 1 y 2) y asignarlos a las dos salidas digitales (valor de comparación 1: 1Q0 ó 2Q0, valor de comparación 2: 1Q1 ó 2Q1). Los valores de comparación se comparan con el estado de contaje actual. Cuando el estado de contaje alcanza un valor de comparación, puede activarse la salida correspondiente.

Valores de comparación 1 y 2

Los dos valores de comparación se introducen en el DB de la instrucción CNT_CTRL (CMP_V1, CMP_V2) y se transfieren al FM 450-1 activando los bits T_CMP_V1 o T_CMP_V2 respectivamente. El contaje no se ve afectado por este hecho.

Los valores de comparación deben estar dentro de los límites del rango de contaje en cuestión. El valor de comparación se interpreta conforme al rango de contaje seleccionado. Si, por ejemplo, se predetermina FFFF FFFF H como valor de comparación, este valor se interpretará como 4 294 967 295 en el rango de contaje de "0 a +32 bits", o bien como -1 en el rango de contaje de "-31 a +31 bits".

Para los valores de comparación se permiten los rangos de valores siguientes:

Rango de los valores de comparación	
Límite inferior	Límite de contaje inferior máx.
Límite superior	Límite de contaje superior máx.

Habilitar las salidas

Antes de poder controlar las salidas es preciso haberlas habilitado activando los bits correspondientes en el DB. Si se desactiva uno de estos bits, la salida correspondiente se desconectará inmediatamente, incluso si se ha parametrizado una duración de impulso para la misma.

Para la habilitación de las salidas pueden activarse los siguientes bits:

Salida	...es habilitada por el bit
Q0	CTRL_DQ0
Q1	CTRL_DQ1

Comportamiento de las salidas en los modos de contaje

Definir el comportamiento de las salidas

Comportamiento de las salidas Q0 y Q1

Seleccione el comportamiento deseado de la salida correspondiente. Active para ello la casilla de verificación correspondiente.

- Sin comparación (Página 9603)
- Activar si la lectura del contador >= valor de comparación (Página 9603)
- Activar si la lectura del contador <= valor de comparación (Página 9603)
- Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante (Página 9604)

- Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás (Página 9605)
- Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás (Página 9604)

Nota

Para activar las opciones, es preciso introducir los valores de comparación 1 y 2 en el DB con los datos de contaje, así como transferirlos al módulo con la instrucción CNT_CTRL; además, es preciso activar los bits CTRL_DQ0 y CTRL_DQ1 en el DB para habilitar las salidas. Véase también "Valores de comparación y salidas (Página 9599)".

Sin comparación

Sin comparación

La salida permanece desactivada y no se ve afectada por los eventos valor de comparación, paso por cero, rebase por exceso o por defecto. La salida Qx se puede utilizar puramente como salida digital.

Activar si la lectura del contador >= valor de comparación

Activar si la lectura del contador >= valor de comparación

La salida se pone a “1” si el contador se encuentra en el rango comprendido entre el valor de comparación y el desbordamiento por exceso.

La salida se pone a “1” si el contador se ajusta a un valor comprendido entre el valor de comparación y el desbordamiento por exceso.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área.

Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9605).

Activar si la lectura del contador <= valor de comparación

Activar si la lectura del contador <= valor de comparación

La salida se pone a “1” si el contador se encuentra en el rango comprendido entre el valor de comparación y el desbordamiento por defecto.

La salida se pone a “1” si el contador se ajusta a un valor comprendido entre el valor de comparación y el desbordamiento por defecto.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área.

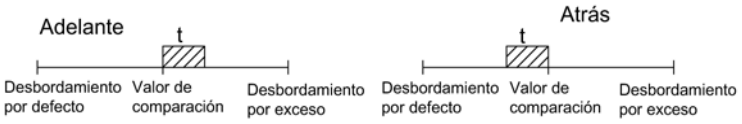
Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9605).

Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás

Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás

Mientras dura el impulso e independientemente del sentido de contaje, la salida se pone a "1" cuando se alcanza el valor de comparación.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área (t = duración del impulso).

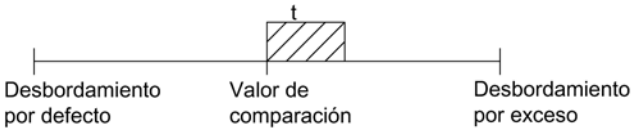


Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante

Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante

Mientras dura el impulso, la salida se pone a "1" cuando el contador alcanza el valor de comparación al contar hacia delante.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área (t = duración del impulso).



Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9605).

Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás

Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás

Mientras dura el impulso, la salida se pone a "1" cuando el contador alcanza el valor de comparación al contar hacia atrás.

La superficie sombreada del gráfico siguiente que la salida está activada en esta área (t = duración del impulso).

Tenga en cuenta las condiciones marco para el comportamiento de las salidas (Página 9605).

Condiciones marco para el comportamiento de las salidas

Condiciones marginales para el comportamiento de las salidas digitales

Al parametrizar el comportamiento de las salidas digitales debe observar las siguientes condiciones marco.

Si..	Entonces...
... desea parametrizar una salida "Activar si el estado de contador >= valor de comparación o <= valor de comparación",	... deberá asegurarse de que el intervalo de tiempo entre estos dos eventos sea mayor que el tiempo de conmutación mínimo de las salidas (tiempo de conmutación: 300 µs); de lo contrario, se perderán los impulsos de control de las salidas. Si el estado del contador alcanza nuevamente el valor de comparación mientras la salida sigue activa, no se disparará un nuevo impulso. No se podrá emitir un nuevo impulso hasta que la salida deje de estar activada.
... desea parametrizar una salida "Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante"	... no podrá habilitar una alarma de proceso al "Alcanzar el valor de comparación correspondiente hacia atrás".
... desea parametrizar una salida "Activar al alcanzarse el valor de comparación por la duración del impulso en sentido descendente"	... no podrá habilitar una alarma de proceso al "Alcanzar el valor de comparación correspondiente en sentido ascendente".

Habilitación de alarmas de proceso del canal n

Habilitar la activación de una alarma de proceso

Introducción

En el contador de FM 450-1 se puede definir qué eventos deben generar una alarma de proceso durante el funcionamiento.

¿Qué es una alarma de proceso?

Si se debe reaccionar a un evento determinado, independientemente del ciclo de la CPU, cada contador del FM 450-1 puede disparar una alarma de proceso. En consecuencia de ello, la CPU interrumpe el programa cíclico y procesa el OB de alarma de proceso.

Habilitar la alarma de proceso

La habilitación de las alarmas de proceso tiene lugar en dos pasos:

1. En la configuración del hardware las alarmas del módulo se habilitan en los parámetros básicos y se especifica si el módulo debe activar una alarma de diagnóstico y/o una alarma de proceso.
2. En "Habilitar alarma de proceso" es preciso seleccionar los eventos correspondientes (Página 9607):

OB de alarma de proceso, OB 4x

Si ocurre una alarma de proceso, el programa de usuario se interrumpe, los datos se transfieren del módulo a la información de arranque del OB 4x (Página 9607) y se llama al OB 4x. Cerrando el OB 4x se anula la alarma de proceso.

La CPU pasa a modo STOP si no se ha programado el OB 4x. Al conmutarse luego nuevamente a modo RUN, se habrán borrado las peticiones de alarma de proceso.

Alarma de proceso perdida

Si ocurre un evento que deba disparar una alarma de proceso, y el mismo evento anterior no se ha acusado todavía, no se dispararán más alarmas de proceso; la alarma de proceso se perderá. Esta situación puede conducir a la alarma de diagnóstico "Alarma de proceso perdida".

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de proceso.

Eventos para la activación de una alarma de proceso

¿Qué eventos pueden disparar una alarma de proceso?

Durante el funcionamiento del FM 450-1 los siguientes eventos pueden activar una alarma de proceso (con independencia para cada uno de los contadores):

- Apertura de la puerta (en los modos de operación con puerta HW o SW)
- Cierre de la puerta (en los modos de operación con puerta HW o SW)
- Rebase del límite superior
- Rebase del límite inferior
- Paso por cero

- Alcance del valor de comparación 1 ó 2 en sentido de contaje adelante
- Alcance del valor de comparación 1 ó 2 en sentido de contaje atrás
- Activar contador (DI-Set)

Active la casilla de verificación para los eventos correspondientes.

Es posible seleccionar un número cualquiera de eventos para disparar alarmas de proceso. Para las alarmas de proceso que se disparan al alcanzar un valor de comparación hay que tener en cuenta las condiciones marco para el comportamiento de las salidas (Página 9603).

Información de arranque del OB 4x

Información de arranque del OB 4x

La variable temporal OB4x_POINT_ADDR se escribe en la información de arranque del OB 4x.

La variable OB4x_POINT_ADDR (byte 8 a 11) abarca 4 bytes. En estos cuatro bytes se guarda información sobre el evento que ha provocado la alarma de proceso.

La tabla muestra qué bits se activan según el tipo de alarma. Los bits no mencionados son irrelevantes y adoptan el valor cero.

Byte	Bit	Significado: Alarma en caso de ...
		Contador 1
8	0	Apertura de la puerta
	1	Cierre de la puerta
	2	Rebase del límite superior
	3	Rebase del límite inferior
	4	Alcance del valor de comparación 1 en sentido de contaje adelante
	5	Alcance del valor de comparación 1 en sentido de contaje atrás
	6	Alcance del valor de comparación 2 en sentido de contaje adelante
	7	Alcance del valor de comparación 2 en sentido de contaje atrás
9	0	Paso por cero
	5	Activar contador (DI-Set) (sincronización)
		Contador 2
10	0	Apertura de la puerta
	1	Cierre de la puerta
	2	Rebase del límite superior
	3	Rebase del límite inferior
	4	Alcance del valor de comparación 1 en sentido de contaje adelante
	5	Alcance del valor de comparación 1 en sentido de contaje atrás
	6	Alcance del valor de comparación 2 en sentido de contaje adelante
	7	Alcance del valor de comparación 2 en sentido de contaje atrás
11	0	Paso por cero
	5	Activar contador (DI-Set) (sincronización)

Uso del FM 450-1 desde el programa de usuario

Inclusión del FM 450-1 en un programa de usuario

Para integrar el FM 450-1 en un programa de usuario pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

La instrucción CNT_CTRL se activa en el programa de usuario de la CPU. Ésta permite la comunicación entre la CPU y el FM 450-1.

Todos los datos pertenecientes a un canal del módulo se encuentran en el DB de la instrucción CNT_CTRL (DB de contaje (Página 7393)). Inserte en el proyecto un bloque de datos del tipo CNT_CHANTYPE1 mediante "Bloques de programa > Agregar bloque".

Además, para el FM 450-1 se puede utilizar también la instrucción DIAG_INF, que permite transferir información de diagnóstico al DB de contaje.

Consulte también

CNT_CTRL (Página 7388)

DIAG_INF (Página 7390)

FM 450-1: Online y diagnóstico

Información actual sobre el módulo

Diagnóstico estándar

Indicadores de fallos mediante los LEDs de diagnóstico del FM 450-1

¿Dónde se indica el fallo?

Los fallos se indican a través de los dos LEDs de diagnóstico rojos:

- El LED de diagnóstico INTF indica que hay un error interno en el módulo.
- El LED de diagnóstico EXTF indica que hay un error externo en los terminales de cable.

¿Qué fallos se indican?

La tabla siguiente muestra los tipos de errores indicados por los LEDs de diagnóstico.

Tipo de error/LED	Causa del fallo	Solución
Fallos internos LED de diagnóstico INTF	Fallo en el test de la EPROM	Sustituir módulos
	Fallo en el TEST de la RAM	Sustituir el módulo
	Disparo del temporizador de vigilancia (watchdog)	Sustituir el módulo

Tipo de error/LED	Causa del fallo	Solución
	Alarma de proceso perdida	Se acusa procesando la alarma de proceso.
Fallos externos LED de diagnóstico EXTF	Tensión auxiliar 1L+/1M no conectada o cortocircuito en la alimentación del encóder DC24V	Corregir la conexión
	Cortocircuito o sobrecarga en la alimentación del encóder DC5.2V	Corregir la conexión
	Fallos en las señales del encóder de 5 V (rotura de hilo, cortocircuito, falta de línea)	Corregir la conexión

Activación de una alarma de diagnóstico

Todos los fallos, exceptuando el del test de la EPROM, pueden disparar una alarma de diagnóstico si ésta se ha habilitado en los parámetros básicos de la parametrización. En los registros de diagnóstico DS0 y DS1 (Página 9611) puede ver qué error ha provocado que se encienda el LED.

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

¿Qué es una alarma de diagnóstico?

Si se quiere que un programa de usuario responda a un fallo interno o externo, se puede parametrizar una alarma de diagnóstico que interrumpa el programa cíclico de la CPU y llame al OB de alarma de diagnóstico (OB 82).

¿Qué eventos pueden disparar una alarma de diagnóstico?

Los siguientes eventos pueden provocar una alarma de diagnóstico:

- Tensión auxiliar externa 1L+/1M defectuosa
- Alimentación de 5,2 VDC del encóder cortocircuitada o sobrecargada
- Falta parametrización del módulo
- Error en la parametrización del módulo
- La vigilancia de tiempo (watchdog) se ha disparado
- RAM defectuosa
- Alarma de proceso perdida
- Fallo en señal A (rotura de hilo, cortocircuito, falta cable)
- Fallo en señal B (rotura de hilo, cortocircuito, falta cable)
- Fallo en señal N (rotura de hilo, cortocircuito, falta cable)

Habilitación de la alarma de diagnóstico

En la configuración del hardware las alarmas del módulo se habilitan en los parámetros básicos y se especifica si el módulo debe activar una alarma de diagnóstico y/o una alarma de proceso.

Reacciones a una alarma de diagnóstico

Cuando se produce un evento que puede disparar una alarma de diagnóstico ocurre lo siguiente:

- La información de diagnóstico se registra en los registros de diagnóstico DS0 y DS1 (Página 9611) del módulo.
- El LED de error de grupo se enciende.
  Una vez eliminado el fallo o error, se apaga el LED de error de grupo.
- Se llama al OB de alarma de diagnóstico (OB 82).
- El registro de diagnóstico DS0 se graba en la información de arranque del OB de alarma de diagnóstico.
- El contaje prosigue sin cambios si se ha programado una alarma del OB 82. La CPU pasa a modo STOP si no hay un OB 82.

Ajuste predeterminado

En el ajuste predeterminado no se han habilitado alarmas de diagnóstico.

Registros de diagnóstico DS0 y DS1

Registro de diagnóstico DS0 y DS1

Para saber cuál ha sido el evento que ha disparado una alarma de diagnóstico se consultan los registros de diagnóstico DS0 y DS1. El registro de diagnóstico DS0 consta de cuatro bytes; DS1 consta de 16 bytes, siendo los cuatro primeros idénticos a DS0.

Leer el registro desde el módulo

El registro de diagnóstico DS0 (Página 9611) se transfiere automáticamente a la información de arranque en la llamada del OB de diagnóstico. Allí, los cuatro bytes se guardan en los datos locales (bytes 8 a 11) del OB 82.

El registro de diagnóstico DS1 (Página 9612) (y, por lo tanto, también el contenido del DS0) puede leerse con la instrucción DIAG_INF (Página 7390) del módulo. Esto sólo tiene sentido cuando se notifica un fallo en un canal en el DS0.

Asignación del registro de diagnóstico DS0

Asignación del registro de diagnóstico DS0 y de la información de arranque

La tabla siguiente muestra la asignación del registro de diagnóstico DS0 en la información de arranque. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Byte	Bit	Significado	Observación	Nº de evento
0	0	Módulo averiado	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Fallo interno	Se activa en todos los fallos internos: • Fallo en test de RAM • La vigilancia de tiempo (watchdog) ha saltado • Alarma de proceso perdida	8:x:01
	2	Fallo externo	Se activa en todos los fallos externos: • Tensión auxiliar 1L+/1M no conectada o alimentación del encóder de 5,2 VDC cortocircuitada • Alimentación de 5,2 V DC del encoder cortocircuitada o sobrecargada • Señales de 5 V defectuosas • Error de parametrización	8:x:02
	3	Fallo en un canal	Siempre tiene el valor 1 cuando hay un error interno o externo.	8:x:03
	4	Fallo en la tensión auxiliar externa	Comprobar tensión	8:x.04
	7	Error de parametrización	Para más información, consulte el capítulo Errores de datos (Página 9613)	8:x:07
1	0...3	Clase de tipo	Siempre asignado con un 8	
	4	Información de canal	Siempre tiene asignado un 1	
2	3	La vigilancia de tiempo (watchdog) ha saltado	Módulo defectuoso o fuertes interferencias	8:x:33
3	3	Defecto en RAM	Módulo defectuoso o fuertes interferencias	8:x:43
	6	Alarma de proceso perdida	Comprobar la configuración. Se ha detectado un evento y no se ha podido notificar porque el mismo evento aún no se ha sido acusado por la CPU	8:x:46

Asignación del registro de diagnóstico DS1

Asignación del registro de diagnóstico DS1

El registro de diagnóstico DS1 se compone de 12 bytes. Los 4 primeros bytes son idénticos a los del registro de diagnóstico DS0. La siguiente tabla muestra la asignación de los bytes restantes. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Este registro es introducido por la instrucción DIAG_INF (Página 7390) en el DB de la instrucción CNT_CTRL a partir de la DW54.

Byte	Bit	Significado	Nota	Nº de evento
4	0...6	Tipo de canal	Siempre con un 76H asignado	
	7	Más tipos de canal	Siempre asignado con un 0	
5	0...7	Longitud de la información de diagnóstico	Siempre tiene asignado un CH	
6	0...7	Cantidad de canales	Siempre tiene asignado un 2	
7	0…1	Vector de fallo de canal	Bit 0= 1: Fallo en canal 1 Bit 1= 1: Fallo en canal 2	
8	0	Señal A errónea canal 1		8:x:B0
	1	Señal B errónea canal 1		8:x:B1
	2	Señal N errónea canal 1		8:x:B2
	4	Alimentación del encóder 5,2 V errónea canal 1		8:x:B4
	5...7	Reservado		
9		Reservado		
10	0	Señal A errónea canal 2		8:x:B0
	1	Señal B errónea canal 2		8:x:B1
	2	Señal N errónea canal 2		8:x:B2
	4	Alimentación del encóder 5,2 V errónea canal 2		8:x:B4
	5...7	Reservado		
11		Reservado		

Error de datos

¿Cuándo se producen errores de datos?

Cuando se transfieren nuevos parámetros al módulo, el FM 450-1 comprueba dichos parámetros. Si se producen errores durante esta comprobación, el módulo notifica estos errores.

¿Dónde se indican los errores de datos?

La instrucción CNT_CTRL (Página 7388) registra los errores de datos con sus números de error en el DB de la instrucción CNT_CTRL. La variable DA_ERR_W permite acceder a esta palabra de datos en el programa de usuario.

¿Qué errores de datos existen?

En la tabla siguiente se muestran los errores de datos, junto con el número de error y el significado.

Nº	Significado
0	No hay error.
201	Valor para selección del receptor incorrecto
202	Valor erróneo para el diagnostico de la pareja de señales
203	Valor erróneo para evaluación de las señales
204	Valor erróneo para filtro de entrada de las señales de contaje de 24 V
205	Valor erróneo para el filtro de entrada de las entradas digitales
206	Inversión del sentido no admisible
207	Comportamiento de 1Q0 (2Q0) parametrizado incorrectamente
208	Comportamiento de 1Q1 (2Q1) parametrizado incorrectamente
209	Rango excedido en la duración del impulso
211	Se ha seleccionado un modo de operación incorrecto
212	No se ha indicado ninguna puerta o se han indicado ambas
215	Al parametrizar la alarma de proceso "Alcanzar el valor de comparación adelante o atrás" se ha definido un sentido de contaje distinto del especificado en la parametrización del comportamiento de las salidas "Activar al alcanzarse el valor de comparación por la duración del impulso adelante o atrás". Estos sentidos de contaje deben coincidir.
216	Las alarmas por control de puerta sólo son posibles en los modos de operación con control de puerta.
217	En el comportamiento "Activar si estado del contador >= valor de comparación" o "Activar si estado del contador <= valor de comparación", no se soporta una alarma al "alcanzarse los valores de comparación".

¿Cómo se acusan los errores de datos?

Corrija los valores de los parámetros de acuerdo con los valores predeterminados. Retransfiera el juego de parámetros corregido al FM 450-1. Éste vuelve a comprobar los parámetros y borra los errores de datos del DB.

Errores de control

¿Cuándo aparecen los errores de operador?

Los errores de operador aparecen cuando se controla el módulo incorrectamente a causa de una preselección incorrecta de las señales de control.

¿Dónde se indican los errores de operador?

La instrucción CNT_CTRL (Página 7388) registra los números de errores de operador en el DB. Activando el parámetro de salida OT_ERR la instrucción CNT_CTRL muestra que se ha producido un error de operador. La variable OT_ERR_B permite acceder a esta palabra de datos en el programa.

¿Qué errores de operador existen?

En la tabla siguiente se muestran los errores de operador, junto con el número de error y el significado.

Nº	Significado
0	No hay error
1	El modo de operación no se puede iniciar con puerta SW
2	El modo de operación no se puede cancelar
4	Solo se soporta con la CPU en STOP

¿Cómo se acusan los errores de operador?

Acuse el error con el parámetro de entrada OT_ERR_A de la instrucción CNT_CTRL (Página 7388).

Uso del ET 200S 1Count24V

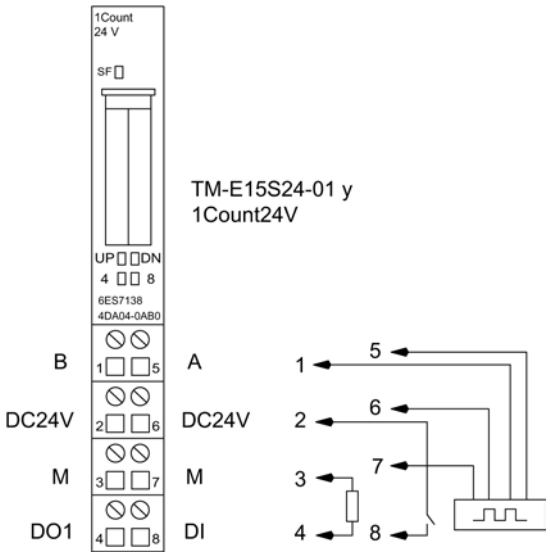
Asignación de conectores 1Count24V

Reglas de cableado

Los cables (terminales 1 y 5 y terminales 2 y 8) deben estar apantallados. La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contacto (véase el anexo de las instrucciones de servicio Sistema de periferia descentralizada ET 200S (http://support.automation.siemens.com/WW/view/es/1144348)).

Asignación de terminales del 1Count24V

La tabla siguiente muestra la asignación de conectores para el 1Count24V.

Vista	Asignación de terminales	Observaciones
		B: Entrada de sentido o pista B A: entrada de impulsos o pista A DC24V: alimentación del encóder M: tierra DI: entrada digital DO1: salida digital

Conexión del generador de impulsos

La tabla siguiente muestra las conexiones de un generador de impulsos en el módulo del terminal y los posibles sentidos de contaje.

Tipo de encóder	Conexión	Sentido de contaje
Encóder incremental de posición sin nivel de sentido	Impulsos de contaje 24 V a terminal 5 (A)	Adelante
Encóder incremental de posición con nivel de sentido	Impulsos de contaje 24 V a terminal 5 (A) y sentido 24 V a terminal 1 (B)	adelante, atrás
Encóder incremental de posición con 2 pistas con desfase de 90º	Pista A terminal 5 (A) y pista B terminal 1 (B)	adelante, atrás

Configurar y parametrizar 1Count24V

En servicio

Seleccionar modo de operación

Modo de operación

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el 1Count24V.

Modo de operación	Descripción
Contaje (Página 9617)	Preselección para los modos de contaje
Medición (Página 9641)	Preselección para los modos de medición
Lectura del recorrido (Página 9658)	El 1Count24V cuenta sin fin en este modo de operación a partir del valor de carga: • Cuando el 1Count24V alcanza el límite superior de contaje en la cuenta ascendente, y entonces aparece otro impulso de contaje, salta hasta el límite inferior de contaje y continúa contando desde ese valor sin perder impulsos. • Cuando el 1Count24V alcanza el límite inferior de contaje en la cuenta descendente, y entonces aparece otro impulso, salta hasta el límite superior de contaje y continúa contando desde ese valor sin perder impulsos.
Modo Fast (Página 9670) (a partir de la versión de firmware 2.0)	El 1Count24V cuenta sin fin en este modo de operación a partir del valor de inicio: • Cuando el 1Count24V alcanza el valor máximo representable con 25 bits (todos los bits activados) al contar en sentido ascendente y cuando llega un nuevo impulso de contaje, el valor de contaje salta a "0" y continúa contando desde ahí sin que se pierda ningún impulso. • Cuando el 1Count24V alcanza el valor "0" al contar en sentido descendente y cuando llega un nuevo impulso de contaje, el valor de contaje salta al valor máximo representable con 25 bits (todos los bits activados) y continúa contando desde éste sin que se pierda ningún impulso. Este modo de operación es adecuado para la lectura del recorrido en ciclos isócronos especialmente breves.

Parámetros para contaje

Activar diagnóstico colectivo

Diagnóstico colectivo

Si activa el diagnóstico colectivo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar la reacción en caso de STOP de la CPU/del maestro

Ajuste del comportamiento en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count24V en caso de fallo del controlador de nivel superior.

Parámetro	Estado del 1Count24V en caso de STOP de la CPU/ del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO 1	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual sigue activo. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.
DO 1 Aplicar valor sustitutivo	El modo de operación actual se cancela. La puerta se cierra y el valor sustitutivo parametrizado para la salida digital se aplica directamente. El valor de comparación 1 und 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros. Si para la salida se parametriza "Impulso al alcanzar el valor de comparación", para la duración del impulso sólo está disponible el valor sustitutivo = 1.	Los parámetros modificados son aceptados y tienen efecto inmediato.
DO 1 Mantener el último valor	El modo de operación actual se cancela. La puerta se cierra y el estado de la salida digital se conserva. Los valores de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/ el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count24V.

Abandonar el estado parametrizado

¿En qué condiciones abandona el 1Count24V el estado parametrizado?

La CPU o el maestro debe estar en modo RUN y la interfaz de control (Página 9637) debe modificarse.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count24V
- POWER ON o inserción del módulo de potencia correspondiente

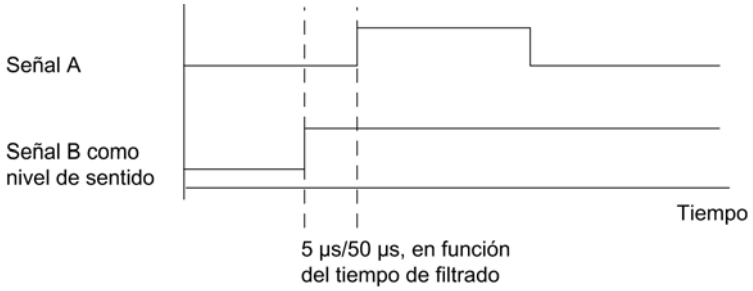
Interpretación de la señal

Seleccionar evaluación de contaje y sentido

Evaluación de señal A*, B*

La evaluación de la señal mediante A*, B* permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Impulso y sentido (Página 9621)
  En los encóders incrementales de posición de 24 V con nivel de sentido debe garantizarse que entre la señal de sentido B* y la señal de contaje A* se observe un intervalo de como mínimo 5 µs/50 µs, según el filtro de entrada parametrizado.
  La figura siguiente muestra el intervalo de tiempo entre la señal de sentido y la señal de contaje.



- Encóder rotativo (Página 9494)
  Si conecta un encóder rotativo de 24 V con dos pistas desfasadas 90 grados en la entrada de contaje y sentido, es posible parametrizar una evaluación simple (Página 9495) en todos los modos de contaje. Además se puede parametrizar una evaluación doble (Página 9495) o cuádruple (Página 9495).
  En todos los modos de evaluación se puede invertir el sentido de contaje (Página 9622) en la entrada de sentido B*.
  En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/ simétrica o salida tipo NPN).

Nota

Si con el 1Count24V ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Nota

La indicación 100 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A* o B*. Así, en la evaluación doble se obtienen como máximo 200 kHz en los impulsos de contaje o máximo 400 kHz en la evaluación cuádruple.

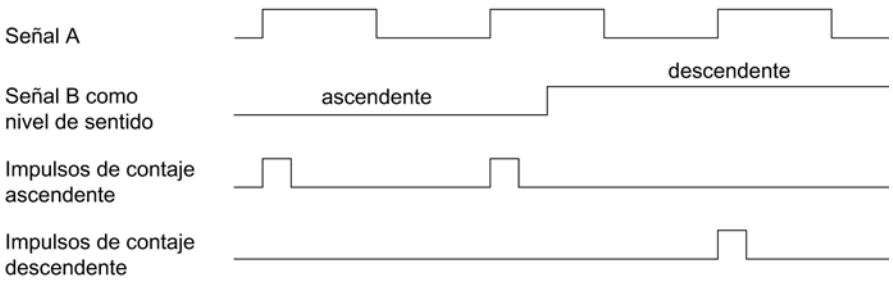
Evaluación "Impulso y sentido"

Impulso y sentido

En este modo de valuación se utiliza como indicador de sentido el nivel en la entrada de sentido B*.

Una entrada no cableada equivale al sentido de contaje "ascendente", si en el parámetro Evaluación de señal ha elegido "Impulso y sentido".

La figura siguiente muestra las señales de un encóder incremental de posición de 24 V con nivel de sentido.



Seleccionar filtro para señales de entrada

Filtro de entrada

Para filtrar las señales en la entrada de contaje A*, la entrada de sentido B* y la entrada digital DI, se puede activar un filtro en función de la duración mínima del impulso o bien de la frecuencia máxima de señales.

Para cada entrada puede seleccionarse un ancho de impulso mínimo de 2,5 µs o 25 µs.

Seleccionar entradas del encóder

Entradas del encóder

En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/simétrica o salida tipo NPN).

Nota

Si se ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Comportamiento de las salidas en los modos de contaje

Valores de comparación y salidas

Introducción

En el 1Count24V se pueden almacenar dos valores de comparación que estén asignados a las salidas digitales. Dependiendo del estado del contador y de los valores de comparación se pueden activar las salidas.

Salidas digitales

El 1Count24V tiene una salida digital "verdadera" y una salida digital "virtual" que sólo existe como bit de estado en la interfaz de notificación.

Ambas salidas pueden parametrizarse (parámetros "Función DO1", "Función DO2").

La función y el comportamiento de las salidas digitales se puede cambiar durante el funcionamiento. La nueva función actúa inmediatamente.

Controlar las salidas simultáneamente con los comparadores

Si se va a utilizar una función de comparación para las salidas, estas se pueden seguir controlando con SET_DO1 o SET_DO2. De esta manera, se puede simular el efecto de la comparación mediante el programa de control:

- Con el flanco ascendente de SET_DO1 o SET_DO2 se activa la salida.
  Si se ha seleccionado la función "Impulso al alcanzarse el valor de comparación", se emitirá un único impulso de la duración especificada. Si la duración del impulso = 0, SET_DO1 y SET_DO2 quedan sin efecto.
  El bit de control SET_DO1 no está permitido en el caso de seleccionar el comportamiento de salida "Conmutación en los valores de comparación".
- Un flanco descendente de SET_DO1 o SET_DO2 desactiva la salida de nuevo.

Obsérvese que los comparadores continúan activos y que pueden activar o desactivar la salida si hay un cambio en el resultado de la comparación.

Nota

Una salida activada con SET_DO1 o SET_DO2 no se desactiva con el comparador.

Cargar los valores de comparación

Los valores de comparación se transfieren al 1Count24V. El contaje no se ve afectado por este hecho.

Rango válido para los dos valores de comparación

Sentido principal de contaje: ninguno	Sentido principal de contaje: adelante	Sentido principal de contaje: atrás
Límite inferior de contaje hasta Límite superior de contaje	–2147483648 hasta Límite superior de contaje -1	1 hasta 2.147.483.647

Cambiar la función y el comportamiento de las salidas digitales

Las funciones y el comportamiento de las salidas se pueden cambiar durante el funcionamiento mediante la interfaz de control (Página 9637). Al hacerlo, el módulo 1Count24V desactiva las salidas y aplica los valores de la siguiente manera:

- **Función de las salidas digitales DO1 y DO2:** Si se cambia la función de manera que se cumpla la condición de comparación, la salida no cambiará hasta después del siguiente impulso de contaje. No obstante, si la histéresis (Página 9627) está activada, el módulo 1Count24V no cambia la salida.
- **Histéresis:** Una histéresis (Página 9627) se mantiene activa incluso después de una modificación. La próxima vez que se alcance el valor de comparación, se aplicará el nuevo rango de la histéresis.
- **Duración del impulso:** La nueva duración del impulso (Página 9630) tiene efecto con el siguiente impulso.

Seleccionar el comportamiento de las salidas

Comportamiento de las salidas DO1 y DO2

En la lista desplegable, seleccione el comportamiento deseado de la salida correspondiente.

- Salida (Página 9624)
- Conexión con estado del contador ≥ valor de comparación (Página 9624)
- Conexión con estado del contador ≤ valor de comparación (Página 9624)
- Impulso al alcanzar el valor de comparación (Página 9625)
- Conmutar en los valores de comparación (Página 9625)(sólo DO1)

Ajustar o modificar la función y el comportamiento de la salida digital DO1

Si quiere ajustar o cambiar el comportamiento de la DO1, asegúrese de que se consideran todas las dependencias parametrizadas, pues de lo contrario puede producirse un error de parametrización o un error de carga.

Condiciones marginales:

Si se parametriza "Conmutar al alcanzar los valores de comparación" para la DO1 hay que hacer lo siguiente:

- ajustar la histéresis = 0 y
- además parametrizar también la "Función DO2" como "Salida".

Salida

Salida

Los bits de control (Página 9637)SET_DO1 y SET_DO2 permiten activar y desactivar las salidas.

Para ello, es preciso que los bits de control (Página 9637)CTRL_DO1 o CTRL_DO2 estén activados.

Los bits de estado STS_DO1 y STS_DO2 de la interfaz de realimentación (Página 9639) permiten consultar el estado de las salidas.

Los bits de estado (Página 9639)STS_CMP1 y STS_CMP2 indican que la salida correspondiente está o estaba conectada. Estos bits de estado conservan su estado hasta ser acusados. Si la salida permanece conectada, el correspondiente bit se vuelve a activar inmediatamente. Estos bits de estado se activan también si los bits de control SET_DO1 o SET_DO2 se accionan cuando las salidas DO1 o DO2 no están habilitadas.

Modo isócrono

En el modo isócrono la salida DO1 conmuta en el instante T_o. El estado de la salida virtual DO2 se notifica en el instante T_i.

Conexión si el estado del contador ≤ valor de comparación o estado del contador ≥ valor de comparación

Estado del contador ≤ valor de comparación y estado del contador ≥ valor de comparación

Cuando las condiciones de comparación se cumplen, el comparador correspondiente activa la salida. El estado de la salida (Página 9639) se indica con STS_DO1 y STS_DO2.

Para ello, es preciso que los bits de control (Página 9637)CTRL_DO1 o CTRL_DO2 estén activados.

El bit de estado (Página 9639)STS_CMP1 o STS_CMP2 indica el resultado de la comparación. Sólo podrá acusar y por tanto desactivar estos bits cuando dejen de cumplirse las condiciones de comparación.

Modo isócrono

También en el modo isócrono la salida DO1 se activa inmediatamente al cumplirse la condición de comparación y, por consiguiente, no depende del ciclo de bus. El estado de la salida virtual DO2 se notifica en el instante T_i.

Impulso al alcanzar el valor de comparación

Impulso al alcanzar el valor de comparación

Cuando el estado del comparador alcanza el valor de comparación, el comparador activa la correspondiente salida digital durante el tiempo parametrizado para el impulso.

Para ello, es preciso que los bits de control (Página 9637)CTRL_DO1 o CTRL_DO2 estén activados.

Los bits de estado (Página 9639)STS_DO1 y STS_DO2 tienen siempre el estado de la correspondiente salida digital.

El resultado de la comparación se indica mediante el bit de estado (Página 9639)STS_CMP1 o STS_CMP2 y sólo se podrá desactivar mediante acuse cuando haya expirado la duración del impulso.

Si se ha parametrizado el sentido principal de contaje, el comparador conmutará sólo cuando se alcance el valor de comparación en el sentido principal de contaje.

Si no se ha parametrizado el sentido principal de contaje, el comparador conmutará sólo cuando se alcance el valor de comparación en cualquiera de los dos sentidos de contaje.

Si la salida digital ha sido activada mediante el bit de control (Página 9637)SET_DO1 o SET_DO2, se desactivará cuando haya transcurrido la duración del impulso.

Modo isócrono

También en el modo isócrono la salida DO1 se activa inmediatamente al cumplirse la condición de comparación y, por consiguiente, no depende del ciclo de bus. El estado de la salida virtual DO2 se notifica en el instante T_i.

Conmutar en los valores de comparación

Conmutar en los valores de comparación

El comparador activa la salida cuando se cumplen las siguientes condiciones:

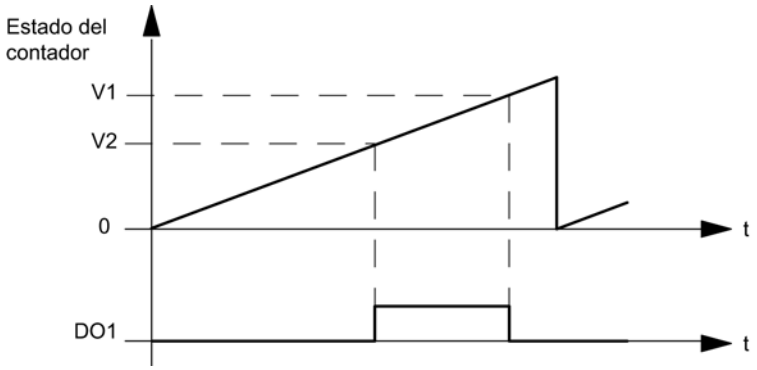
- Los dos valores de comparación deben cargarse mediante la función de carga (Página 9637)CMP_VAL1 y CMP_VAL2 y
- una vez cargados los valores de comparación, hay que habilitar la salida DO1 (Página 9637) con CRTL_DO1.

La tabla siguiente muestra cuándo está activada o desactivada la salida DO1:

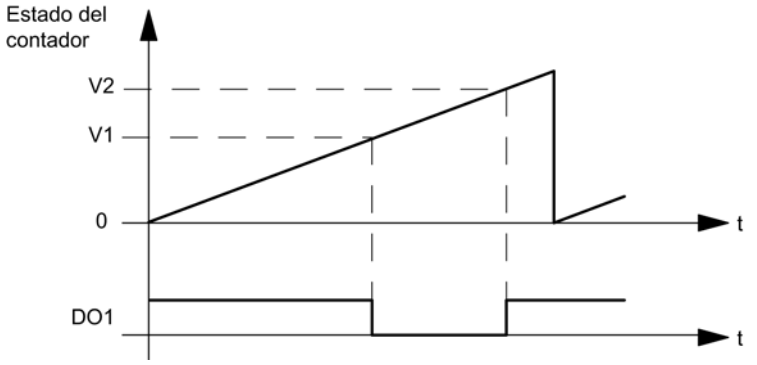
	DO1 está conectada cuando	DO1 está desconectada cuando
V2 < V1 (v. figura inferior)	V2 ≤ estado del contador ≤ V1	V2 > estado del contador o estado del contador > V1
V2 = V1	V2 = estado del contador = V1	V2 ≠ estado del contador ≠ V1
V2 > V1 (v. figura inferior)	V1 > estado del contador o estado del contador > V2	V1 ≤ estado del contador ≤ V2

Las siguientes figuras muestran el estado de la salida DO1 dependiendo del estado del contador y de los valores de comparación:

- V2 < V1 al comenzar el contaje



- V2 > V1 al comenzar el contaje



El bit de estado (Página 9639)STS_CMP1 indica el resultado de la comparación. Sólo podrá acusar y por tanto desactivar este bit cuando deje de cumplirse la condición de comparación.

No hay ninguna histéresis cuando la salida tiene este comportamiento.

En este comportamiento de salida, el bit de control (Página 9637)SET_DO1 no puede controlar la salida DO1.

Modo isócrono

También en el modo isócrono la salida DO1 se activa inmediatamente al cumplirse la condición de comparación y, por consiguiente, no depende del ciclo de bus.

Seleccionar valor sustitutivo DO1

Valor sustitutivo DO1

Si se ha seleccionado el parámetro de conmutar valor sustitutivo DO1 en "Comportamiento en STOP de la CPU/del maestro (Página 9615)", seleccione aquí el valor sustitutivo que debe enviarse a la salida digital DO1.

Activar diagnóstico DO1

Diagnóstico DO1

El "diagnóstico DO1" (rotura de hilo, cortocircuito) sólo es posible con anchos de impulso > 90 ms en la salida digital DO1.

Active la casilla de verificación para el diagnóstico de la salida DO1.

Definir histéresis

Histéresis

Un encóder puede permanecer en una posición particular y fluctuar entonces alrededor de esta posición. Esto hace que el estado del contador fluctúe alrededor de un valor determinado. Si hay un valor de comparación en este rango de fluctuación, por ejemplo, la salida correspondiente se activa y desactiva de acuerdo con el ritmo de las fluctuaciones. Para evitar estas conmutaciones cuando se producen pequeñas oscilaciones, el 1Count24V cuenta con una histéresis parametrizable. Se puede parametrizar un rango entre 0 y 255 (0 significa: histéresis desconectada).

La histéresis también actúa en caso de desbordamiento por exceso y por defecto.

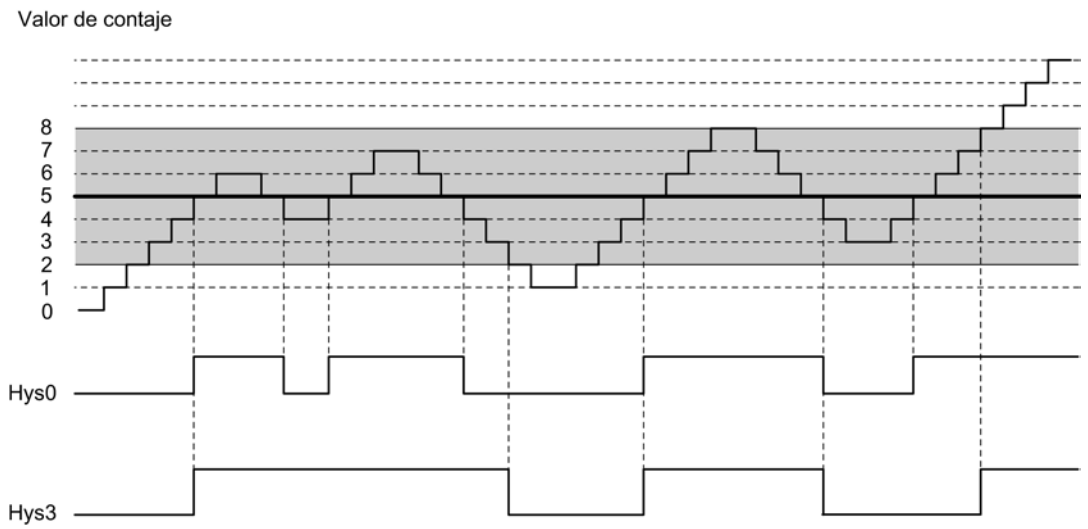
Funcionamiento de la histéresis con el estado del contador ≤ valor de comparación y estado del contador ≥ valor de comparación

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (= desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo, el valor de comparación = 5.

El contador está parametrizado con "Sentido principal de contaje adelante" y "la salida con "Conexión si estado del contador ≥ valor de comparación".

Al alcanzarse la condición de comparación, se activa la histéresis. Cuando la histéresis está activada, el resultado de la comparación permanece invariable.

Si el valor de contaje abandona el rango de histéresis, la histéresis se desactiva. El comparador se conecta de nuevo conforme a sus condiciones de comparación.



Nota

Si se produce un cambio del sentido de contaje en el valor de comparación cuando la histéresis está activada, la salida se desactiva.

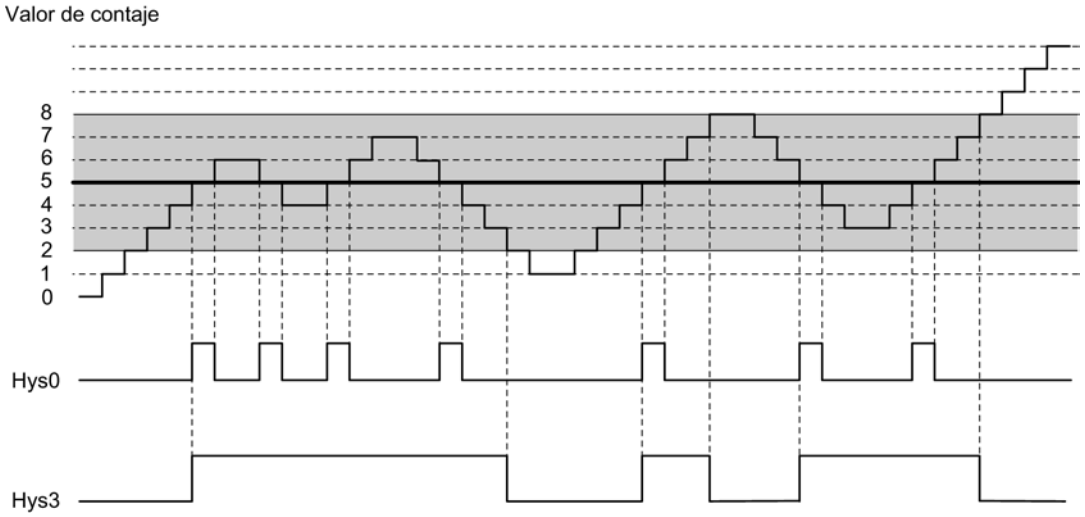
Funcionamiento de la histéresis cuando se alcanza el valor de comparación y la duración del impulso = 0

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo, el valor de comparación = 5.

El contador está parametrizado con "Impulso al alcanzarse el valor de comparación", "Sin sentido principal de contaje" y "Duración del impulso = 0".

Cuando se cumplen las condiciones de comparación se activa la histéresis. Cuando la histéresis está activada, el resultado de la comparación permanece invariable.

Cuando el valor de contaje abandona el rango de histéresis, la histéresis se desactiva. El comparador borra el resultado de la comparación.



Funcionamiento de la histéresis cuando se alcanza el valor de comparación, salida de la duración del impulso

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo, el valor de comparación = 5.

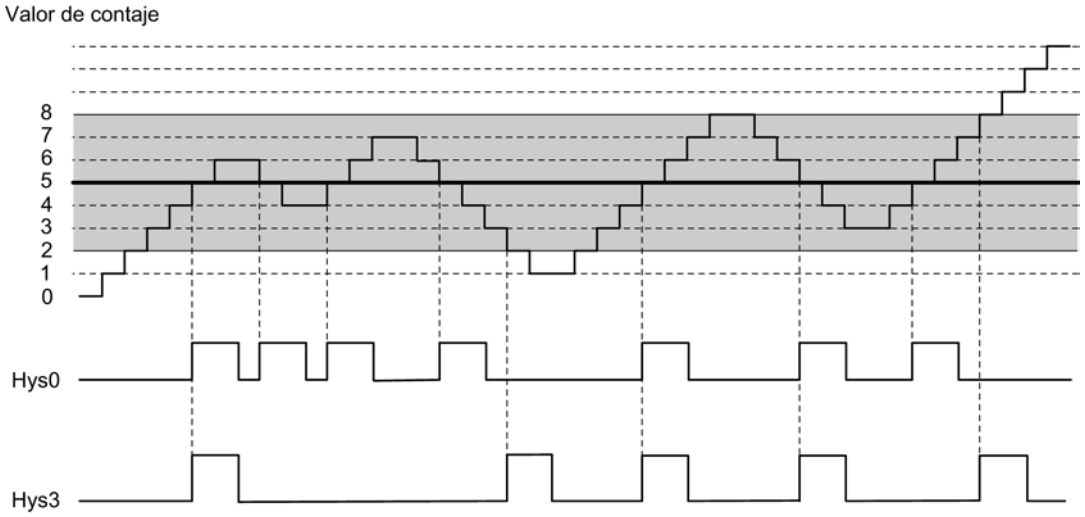
El contador está parametrizado con "Impulso al alcanzarse el valor de comparación", "Sin sentido principal de contaje" y "Duración del impulso > 0".

Cuando las condiciones de comparación se cumplen, la histéresis se activa y se transfiere a la salida un impulso de la duración parametrizada.

Cuando el valor de contaje abandona el rango de histéresis, la histéresis se desactiva.

Cuando se activa la histéresis, el 1Count24V almacena el sentido de contaje.

Si el rango de histéresis se abandona en un sentido diferente del almacenado, se transfiere un impulso a la salida.



Definir duración del impulso

Duración del impulso

La duración del impulso comienza al activarse la salida digital correspondiente. La imprecisión de la duración del impulso es inferior a 2 ms.

La duración del impulso puede ser utilizada para conseguir una adaptación a los actuadores utilizados. La duración del impulso indica durante cuánto tiempo ha de estar activada la salida. La duración del impulso puede ser preseleccionada en intervalos de 2 ms entre 0 ms y 510 ms.

Si la duración del impulso es 0, la salida quedará activada hasta que la condición de comparación deje de cumplirse. Observe que los tiempos de impulsos de contaje deben ser mayores que los tiempos mínimos de conexión de la salida digital.

Modo isócrono

También en el modo isócrono la salida DO1 se activa inmediatamente al cumplirse la condición de comparación y, por consiguiente, no depende del ciclo de bus. El estado de la salida virtual DO2 se notifica en el instante T_i.

Seleccionar modo de contaje

Descripción

Al preseleccionar un modo de contaje se elige la funcionalidad con la que se utilizará el 1Count24V.

Modo de operación	Descripción
Contaje sin fin (Página 9468) (con o sin puerta)	El 1Count24V cuenta en modo sin fin a partir del estado actual del contador.
Contaje único (Página 9469) (con puerta HW o SW)	El 1Count24V comienza a contar a partir del valor de carga al abrirse la puerta y se detiene en el límite de contaje.
Contaje periódico (Página 9470) (con puerta HW o SW)	El 1Count24V comienza a contar al abrirse la puerta a partir del valor de carga entre los límites de contaje.

El ajuste predeterminado es el modo de operación "Contaje sin fin".

Selección del sentido principal de contaje

Descripción

Si se parametriza un sentido principal de contaje (Página 9471) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño, ajustando para ello un límite superior de contaje (Página 9636). El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El sentido principal de contaje sólo puede seleccionarse en los modos de operación "Contaje único" y "Contaje periódico".

Se puede seleccionar el siguiente comportamiento:

- Ninguno (sin sentido principal de contaje)
- Adelante
- Atrás

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Función DI

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital.
- Congelar/redisparar con flanco ascendente (Página 9632)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador. Después de la operación de guardado, el contador se establece al valor de carga.

- Sincronización con flanco ascendente (Página 9633)
  Los flancos ascendentes de la entrada digital permiten establecer el contador al valor de carga.
- Congelar con flanco ascendente (Página 9635)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador.

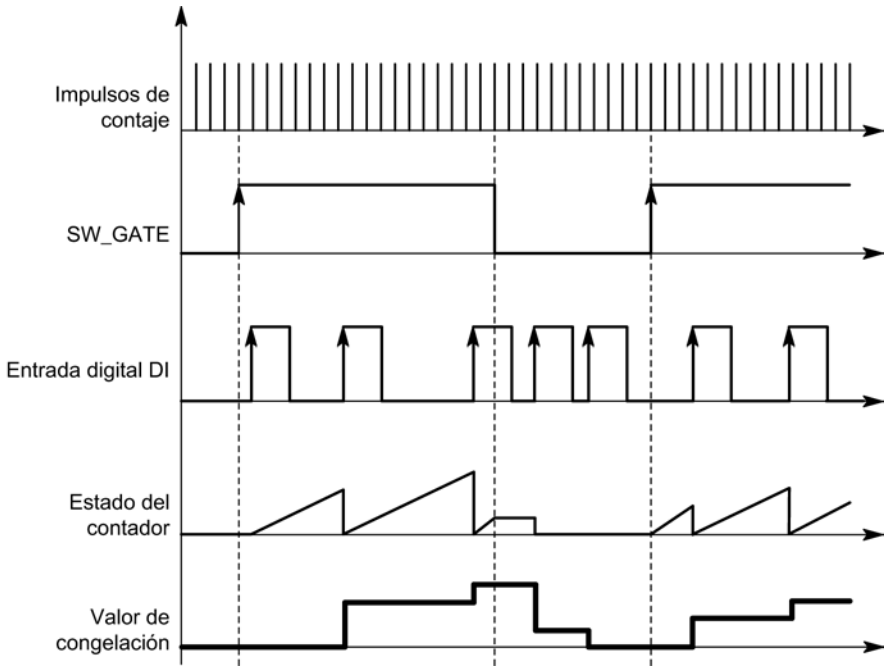
El ajuste predeterminado es "Entrada".

Congelar/redisparar con flanco ascendente

Congelar/redisparar

Con esta función se almacena el estado actual interno del contador del 1Count24V y se redispara el contaje cuando hay un flanco ascendente en la entrada digital. Esto significa que se almacena el estado actual interno del contador en el instante del flanco ascendente (valor de congelación) y, entonces, se carga nuevamente el módulo 1Count24V con el valor de carga y se sigue contando desde allí.

La siguiente figura muestra la función de congelar/redisparar con valor de carga = 0.



Para poder ejecutar la función, el modo de contaje tiene que estar habilitado con la puerta SW. Se inicia con el primer flanco ascendente de la entrada digital.

En la interfaz de notificación (Página 9639) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el estado de la señal de congelación y redisparo.

El valor de congelación se pone por defecto al estado de RESET.

- "0" en el sentido principal de contaje "Ninguno" o "Adelante"
- "Límite de contaje superior parametrizado" en el sentido principal de contaje atrás.

No cambia cuando se abre la puerta SW.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

Si se cierra la puerta SW, sólo se interrumpe el contaje; es decir, cuando se vuelve a abrir la puerta SW, continúa el contaje. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

También en el modo isócrono se congela y redispara el contaje con cada flanco de la entrada digital. En la interfaz de realimentación se muestra el estado del contador válido en el instante del último flanco antes de T_i.

Interfaz de notificación avanzada

Si el módulo 1Count24V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de notificación (Página 9639).

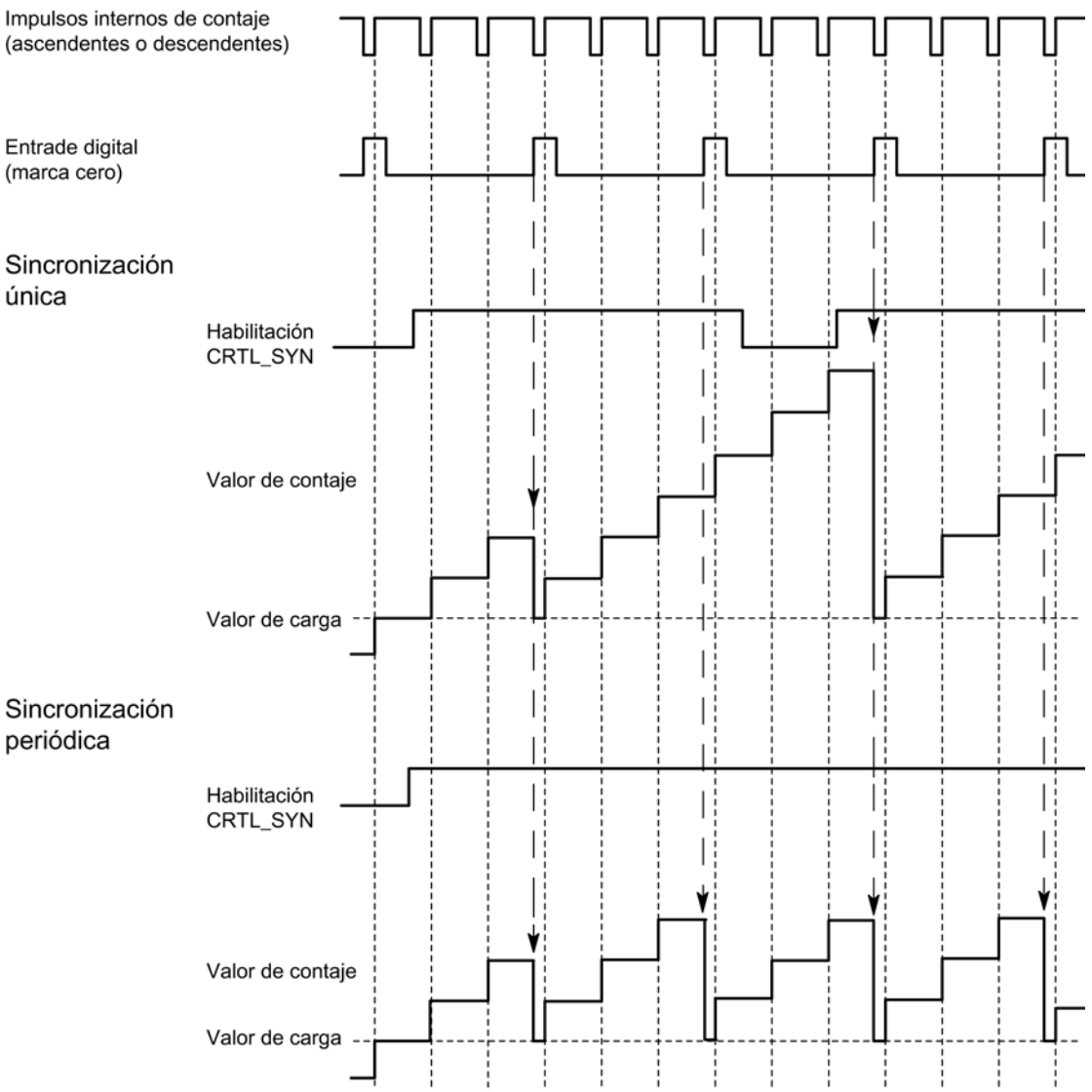
Sincronización con flanco ascendente

Sincronización

Si se ha parametrizado "Sincronización con flanco ascendente", el flanco ascendente de una señal de referencia en la entrada sirve para poner el 1Count24V al valor de carga.

Se puede seleccionar entre la sincronización única y la periódica (parámetro "Sincronización").

La figura siguiente muestra los estados del contador con una sincronización única y periódica.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control (Página 9637) "Habilitación de la sincronización" (CTRL_SYN) debe estar activado.
- Sincronización única: si el bit de habilitación está activado, el primer flanco carga el 1Count24V con el valor de carga.
- Sincronización periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count24V con el valor de carga.
- Una vez realizada la sincronización correctamente, el bit de realimentación (Página 9639)STS_SYN queda activado. Debe desactivarse con el bit de control (Página 9637)RES_STS.

- Como señal de referencia se puede utilizar la señal de un interruptor libre de rebotes o la marca cero de un encóder rotativo.
- El bit de realimentación (Página 9639)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación (Página 9639) activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

Congelar con flanco ascendente

Congelar

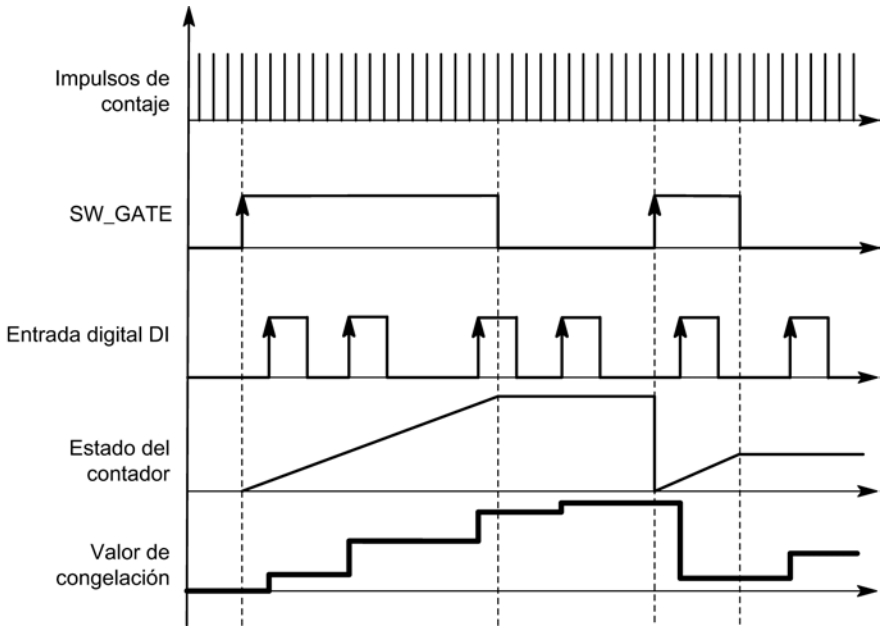
El estado del contador y el valor de congelación reciben sus estados de RESET.

- "0" en el sentido principal de contaje "Ninguno" o "Adelante"
- "Límite de contaje superior parametrizado" en el sentido principal de contaje atrás.

La función de contaje se inicia cuando se abre la puerta SW. El 1Count24V comienza a contar a partir del valor de carga.

- "0" en el sentido principal de contaje "Ninguno" o "Adelante"
- "Límite de contaje superior parametrizado" en el sentido principal de contaje atrás.

La siguiente figura muestra la función de congelar con valor de carga = 0.



El valor de congelación equivale siempre al valor de contaje en el instante del flanco ascendente de la entrada digital DI.

En la interfaz de notificación (Página 9639) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el nivel de la señal de congelación.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

En el modo isócrono, se muestra en la interfaz de realimentación el estado del contador que se congeló en el instante del último flanco ascendente antes de T_i.

Si se cierra la puerta SW, actúa como parametrizada, de cancelación o de interrupción. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

Interfaz de notificación ampliada

Si el módulo 1Count24V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de notificación (Página 9639).

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9639)STS_DI indica el nivel de la entrada digital.

Seleccionar sincronización

Sincronización

Si se ha seleccionado "Sincronización con flanco ascendente" en "Función DI (Página 9628)", es posible elegir entre la sincronización única y la periódica.

- Una vez: si el bit de habilitación está activado, el primer flanco carga el 1Count24V con el valor de carga.
- Periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count24V con el valor de carga.

Determinar el límite superior de contaje

Descripción

Si se parametriza un límite superior de contaje en combinación con un sentido principal de contaje (Página 9471) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño. El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El límite superior de contaje puede adoptar un valor > 1 hasta el límite superior del rango de contaje.

Interfaz de notificación y control para los modos de contaje

Interfaz de control de los modos de contaje

Nota

Para el 1Count24V, los datos siguientes de la interfaz de control son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del 1Count24V para los modos de contaje.

Dirección		Asignación			
Bytes 0 a 3		**Valor de carga directo, preliminar, valor de comparación 1 ó 2**			
	Byte 0	**Comportamiento de DO1, DO2 del 1Count24V**			
		Bit 2	Bit 1	Bit 0	Función DO1
		0	0	0	Salida
		0	0	1	Conexión con estado del contador ≥ valor de comparación
		0	1	0	Conexión con estado del contador ≤ valor de comparación
		0	1	1	Impulso al alcanzarse el valor de comparación
		1	0	0	Conmutar en los valores de comparación
		1	0	1	bloqueado
		1	1	0	bloqueado
		1	1	1	bloqueado
			Bit 5	Bit 4	Función DO2
			0	0	Salida
			0	1	Conexión con estado del contador ≥ valor de comparación
			1	0	Conexión con estado del contador ≤ valor de comparación
			1	1	Impulso al alcanzarse el valor de comparación
		Bits 3, 6 y 7: reservado = 0			
	Bytes 1 a 3	Byte 1	Histéresis DO1, DO2 (rango de valores 0 ... 255)		
		Byte 2	Duración del impulso DO1, DO2 (rango 0 a 255)		
		Byte 3	reservado = 0		
					Denominación
Byte 4		Bit 7	Acuse del error		EXTF_ACK
		Bit 6	Habilitación de DO2		CTRL_DO2
		Bit 5	Bit de control DO2		SET_DO2
		Bit 4	Habilitación de DO1		CTRL_DO1

Dirección	Asignación		
	Bit 3	Bit de control DO1	SET_DO1
	Bit 2	Inicio de la desactivación del bit de estado	RES_STS
	Bit 1	Habilitación de la sincronización	CTRL_SYN
	Bit 0	Bit de control puerta SW	SW_GATE
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	Cambio de función y comportamiento de DO1, DO2	C_DOPARAM
	Bit 3	Carga del valor de comparación 2	CMP_VAL2
	Bit 2	Carga del valor de comparación 1	CMP_VAL1
	Bit 1	Carga preliminar del contador	LOAD_PREPARE
	Bit 0	Carga directa del contador	LOAD_VAL
Bytes 6 a 7	reservado = 0 [1]		

[1] Falta en la interfaz de datos de usuario avanzada

Significado de los bits de control

Bit	Significado
C_DOPARAM	Cambio de función y comportamiento de DO1, DO2 Los valores de los bytes 0 a 2 son aceptados como nueva función, histéresis, y duración de impulso de DO1, DO2. Esto puede conllevar el siguiente error: Las condiciones de conexión/desconexión al alcanzarse los valores de comparación no se cumplen.
CMP_VAL1	Carga del valor de comparación 1 El valor de los bytes 0 a 3 se transfiere al valor de comparación 1 con el bit de control "Carga del valor de comparación 1" (CMP_VAL1).
CMP_VAL2	Carga del valor de comparación 2 El valor de los bytes 0 a 3 se transfiere al valor de comparación 2 con el bit de control "Carga del valor de comparación 2" (CMP_VAL2).
CTRL_DO1	Habilitación de DO1 Este bit habilita la salida DO1
CTRL_DO2	Habilitación de DO2 Este bit habilita la salida DO2.
CTRL_SYN	Este bit habilita la sincronización
EXTF_ACK	Acuse del error Los bits de error deben ser acusados mediante el bit de control EXTF_ACK una vez eliminada la causa.
LOAD_PREPARE	Carga preliminar del contador El valor de los bytes 0 a 3 se acepta como valor de carga.
LOAD_VAL	El valor de los bytes de 0 a 3 se carga directamente como nuevo valor de contaje
RES_STS	Inicio de la desactivación del bit de estado Los bits de estado se desactivan mediante el proceso de realimentación entre el bit RES_STS y el bit RES_STS_A.
SET_DO1	Bit de control DO1 Conectar y desconectar la salida digital DO1 cuando se ha activado CRTL_DO1.

Bit	Significado
SET_DO2	Bit de control DO2 Conectar y desconectar la salida digital DO2 cuando se ha activado CRTL_DO2.
SW_GATE	Bit de control puerta SW La puerta SW se abre/cierra mediante la interfaz de control con el bit SW_GATE.

Interfaz de notificación de los modos de contaje

Nota

Para el 1Count24V, los datos siguientes de la interfaz de notificación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de notificación

La siguiente tabla muestra la asignación de la interfaz de notificación (entradas) del 1Count24V para los modos de contaje.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor de contaje o valor de contaje almacenado con función de congelación en la entrada digital		
Byte 4	Bit 7	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 6	Cortocircuito / rotura de hilo / sobretemperatura	ERR_DO1
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	Desactivación de los bits de estado en curso	RES_STS_A
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Byte 5	Bit 7	Estado de sentido descendente	STS_C_DN
	Bit 6	Estado de sentido ascendente	STS_C_UP
	Bit 5	reservado = 0	
	Bit 4	Estado de DO2	STS_DO2
	Bit 3	Estado de DO1	STS_DO1
	Bit 2	reservado = 0	
	Bit 1	Estado de DI	STS_DI
	Bit 0	Estado de la puerta interna	STS_GATE
Byte 6	Bit 7	Paso por cero en el rango de contaje sin sentido principal de contaje	STS_ND
	Bit 6	Límite inferior	STS_UFLW
	Bit 5	Límite superior	STS_OFLW

Dirección	Asignación		Denominación
	Bit 4	Estado del comparador 2	STS_CMP2
	Bit 3	Estado del comparador 1	STS_CMP1
	Bit 2	reservado = 0	
	Bit 1	reservado = 0	
	Bit 0	Estado de la sincronización	STS_SYN
Byte 7	reservado = 0		
Bytes 8 a 11	Valor de contaje [1]		

[1] Interfaz de datos de usuario avanzada

Nota sobre los bits de realimentación

Bit	Significado
ERR_24V	Cortocircuito en la alimentación del encóder El bit de error tiene que ser acusado con el bit de control (Página 9634)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_DO1	Cortocircuito / rotura de hilo / sobretemperatura por sobrecarga en la salida DO1 El bit de error tiene que ser acusado con el bit de control (Página 9634)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_LOAD	Error en función de carga Los bits de control (Página 9634)LOAD_VAL, LOAD_PREPARE, CMP_VAL1, CMP_VAL2 y C_DOPARAM no pueden estar activados al mismo tiempo durante la transferencia. Esto tiene como consecuencia la activación del bit de estado ERR_LOAD, similar a cargar un valor incorrecto (que no se acepta).
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
RES_STS_A	Desactivación de los bits de estado en curso
STS_C_DN	Estado de sentido descendente
STS_C_UP	Estado de sentido ascendente
STS_CMP1	Estado del comparador 1 El bit de estado STS_CMP1 indica que la salida está o estaba activada. Debe acusarse con el bit de control (Página 9634)RES_STS. Si el bit de estado se acusa cuando la salida todavía está activada, el bit se vuelve a activar inmediatamente. Este bit también se activa cuando se activa el bit de control SET_DO1 sin estar habilitada DO1.
STS_CMP2	Estado del comparador 2 El bit de estado STS_CMP2 indica que la salida está o estaba activada. Debe acusarse con el bit de control (Página 9634)RES_STS. Si el bit de estado se acusa cuando la salida todavía está activada, el bit se vuelve a activar inmediatamente. Este bit también se activa cuando se acciona el bit de control SET_DO2 sin que esté habilitada DO2.
STS_DI	Estado de DI El estado de la DI se indica en todos los modos de operación con el bit STS_DI en la interfaz de notificación.
STS_DO1	Estado de DO1 El bit de estado STS_DO1 muestra el estado de la salida digital DO1.
STS_DO2	Estado de DO2 El bit de estado STS_DO2 indica el estado de la salida digital virtual DO2.
STS_GATE	Estado de la puerta interna Contaje en curso.

Bit	Significado
STS_LOAD	Función de carga en curso
STS_ND	Paso por cero en el rango de contaje sin sentido principal de contaje. El bit debe desactivarse con el bit de control (Página 9634)RES_STS.
STS_OFLW STS_UFLW	Límite superior de contaje excedido Límite inferior de contaje excedido Ambos bits deben ser desactivados.
STS_SYN	Estado de la sincronización Al terminar la sincronización se activa el bit STS_SYN. Debe desactivarse con el bit de control (Página 9634)RES_STS.

Parámetros para medición

Activar diagnóstico colectivo

Diagnóstico colectivo

Si activa el diagnóstico colectivo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar la reacción en caso de STOP de la CPU/del maestro

Ajuste del comportamiento en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count24V en caso de fallo del controlador de nivel superior.

Parámetro	Estado del 1Count24V en caso de STOP de la CPU/del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO 1	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual sigue activo. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.

Parámetro	Estado del 1Count24V en caso de STOP de la CPU/ del maestro	¿Qué ocurre en el caso de una reparametrización?
DO 1 Aplicar valor sustitutivo	El modo de operación actual se cancela. La puerta se cierra y el valor sustitutivo parametrizado para la salida digital se aplica directamente. El valor de comparación 1 und 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros. Si para la salida se parametriza "Impulso al alcanzar el valor de comparación", para la duración del impulso sólo está disponible el valor sustitutivo = 1.	Los parámetros modificados son aceptados y tienen efecto inmediato.
DO 1 Mantener el último valor	El modo de operación actual se cancela. La puerta se cierra y el estado de la salida digital se conserva. Los valores de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/ el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count24V.

Abandonar el estado parametrizado

¿En qué condiciones abandona el 1Count24V el estado parametrizado?

La CPU o el maestro debe estar en modo RUN y la interfaz de control (Página 9655) debe modificarse.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count24V
- POWER ON o inserción del módulo de potencia correspondiente

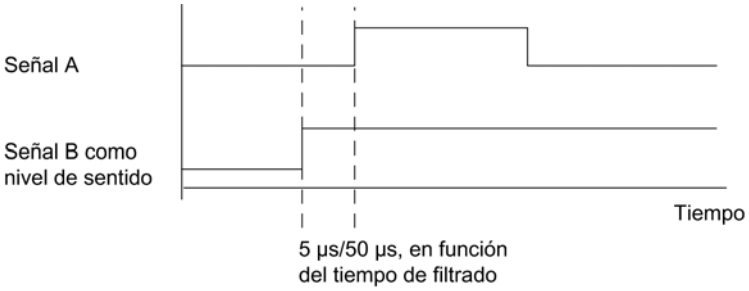
Interpretación de la señal

Seleccionar evaluación de contaje y sentido

Evaluación de señal A*, B*

La evaluación de la señal mediante A*, B* permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Impulso y sentido (Página 9643)
  En los encóders incrementales de posición de 24 V con nivel de sentido debe garantizarse que entre la señal de sentido B* y la señal de contaje A* se observe un intervalo de como mínimo 5 µs/50 µs, según el filtro de entrada parametrizado.
  La figura siguiente muestra el intervalo de tiempo entre la señal de sentido y la señal de contaje.



- Encóder rotativo
  Si conecta un encóder rotativo de 24 V con dos pistas desfasadas 90 grados en la entrada de contaje y sentido, es posible parametrizar una evaluación simple (Página 9495) en todos los modos de medición.
  En la entrada de sentido B* es posible invertir elsentido de contaje (Página 9649).
  En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/ simétrica o salida tipo NPN).

Nota

Si con el 1Count24V ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Nota

La indicación 100 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A* o B*.

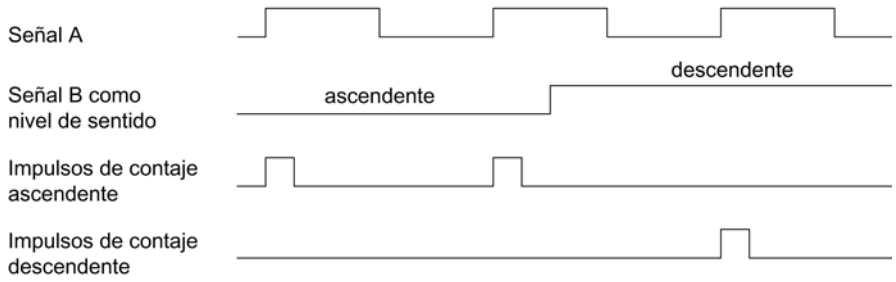
Evaluación "Impulso y sentido"

Impulso y sentido

En este modo de valuación se utiliza como indicador de sentido el nivel en la entrada de sentido B*.

Una entrada no cableada equivale al sentido de contaje "ascendente", si en el parámetro Evaluación de señal ha elegido "Impulso y sentido".

La figura siguiente muestra las señales de un encóder incremental de posición de 24 V con nivel de sentido.



Método de medición

Resumen

Método de medición

En el parámetro "Método de medición" es posible elegir entre los siguientes métodos:

- con tiempo de integración (Página 9644)
- continuo (Página 9645)

Medición con tiempo de integración

Desarrollo de la medición con tiempo de integración

La medición se lleva a cabo durante el tiempo de integración parametrizado. Al transcurrir el tiempo de integración se actualiza el valor medido.

El final de una medición se indica con el bit de estado (Página 9657)STS_CMP1. Este bit se desactiva con el bit de control (Página 9655)RES_STS en la interfaz de control.

Si no se han producido al menos dos flancos ascendentes dentro del tiempo de integración parametrizado, se devuelve 0 como valor medido.

Se devuelve el valor -1 s hasta finalizar el primer tiempo de integración.

El tiempo de integración para la siguiente medición se puede cambiar durante el funcionamiento.

Inversión del sentido de giro

Si el sentido de rotación se invierte durante un tiempo de integración, el valor medido para este período es incierto. Al evaluar los bits de realimentación (Página 9657)STS_C_UP y

STS_C_DN (evaluación de sentido), se puede responder a una posible irregularidad del proceso.

Consulte también

Tiempo de integración y tiempo de actualización en el modo isócrono (Página 9648)

Medición continua

Principio de medición

El módulo 1Count24V cuenta todos los flancos ascendentes de un impulso y les asigna un valor de tiempo en µs.

El tiempo de actualización indica en qué espacio de tiempo se actualiza el valor medido del módulo en la interfaz de realimentación.

En una secuencia de impulsos con uno o varios impulsos por tiempo de actualización rige lo siguiente:

Tiempo de medición dinámico =	valor de tiempo del último impulso en el intervalo actual del tiempo de actualización menos valor de tiempo del último impulso en el intervalo anterior del tiempo de actualización

Una vez transcurrido el tiempo de actualización, se calcula y aplica un nuevo valor medido con el tiempo de medición dinámico.

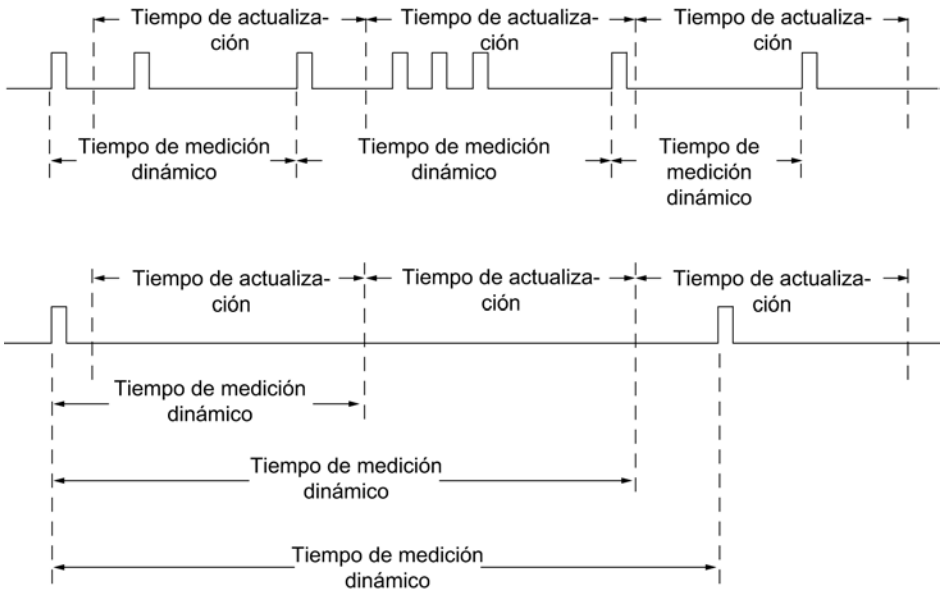
Si el tiempo de actualización actual no contiene ningún impulso, el tiempo de medición dinámico resultante es el siguiente:

Tiempo de medición dinámico =	valor de tiempo del tiempo de actualización actual transcurrido menos el valor de tiempo del último impulso

Una vez transcurrido un tiempo de actualización, se calcula un valor medido aproximado con el tiempo de medición dinámico suponiendo que haya aparecido un impulso al final del tiempo de actualización.

Si en la medición de frecuencia y de velocidad el valor medido aproximado "1 impulso por tiempo de medición dinámico" es menor que el último valor medido, este valor medido aproximado se emite como nuevo valor medido. En la medición del período, el tiempo de medición dinámico se emite como duración aproximada del período cuando el tiempo de medición dinámico es mayor que la última duración medida del período.

La figura muestra el principio de medición.



El 1Count24V mide de forma continua. Durante la parametrización se introduce un tiempo de actualización.

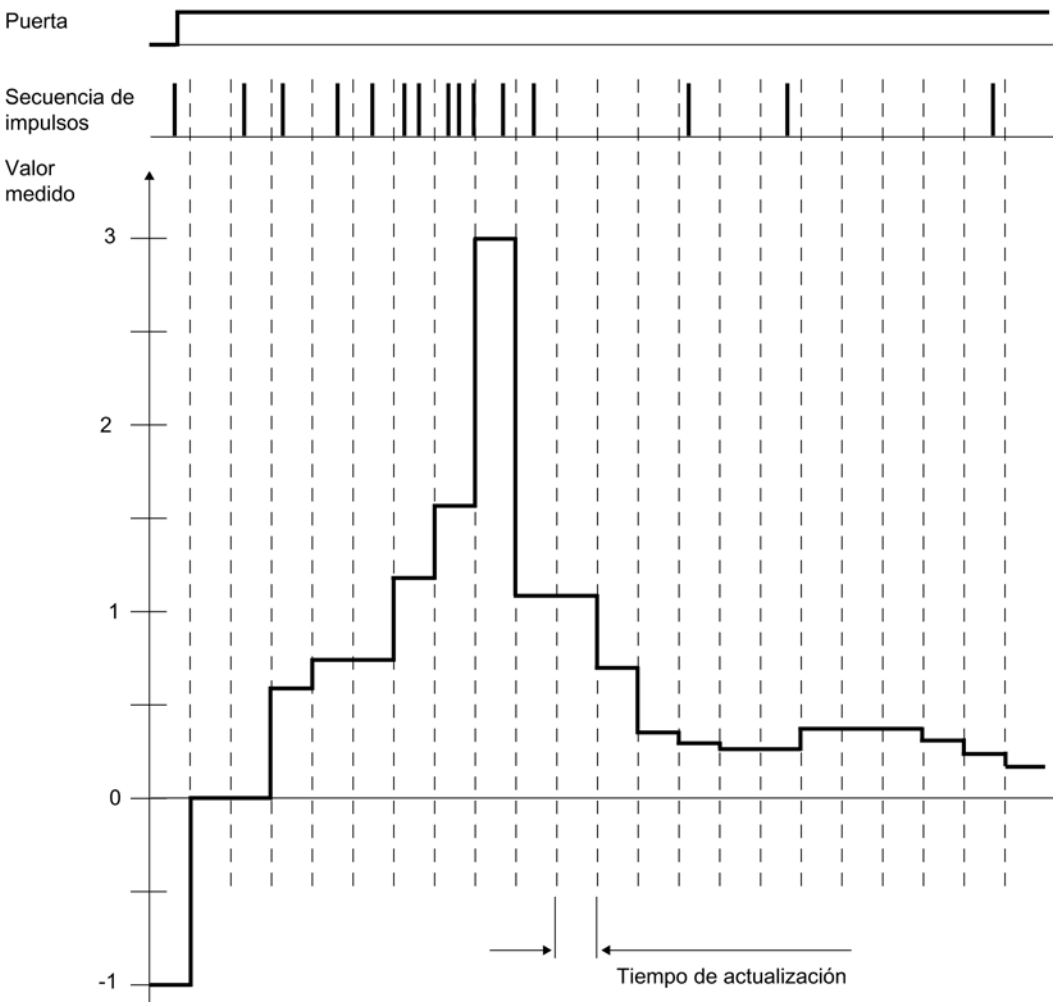
El valor "–1" se devuelve antes de finalizar el primer tiempo de actualización transcurrido.

La medición continua empieza tras abrir la puerta con el primer impulso de la secuencia de impulsos que debe medirse. El primer valor medido no puede calcularse hasta después del segundo impulso.

Cada vez que transcurre el tiempo de actualización, en la interfaz de realimentación se emite un valor medido (frecuencia, duración del período o velocidad). El final de una medición se indica con el bit de estado (Página 9657)STS_CMP1. Este bit se desactiva con los bits RES_STS y RES_STS_A según el principio de realimentación completo.

Si el sentido de rotación se invierte durante un tiempo de actualización, el valor medido para este período es incierto. Evaluando los bits de realimentación (Página 9657)STS_C_DN y STS_C_UP (evaluación del sentido) se puede reaccionar a posibles irregularidades del proceso.

La figura siguiente explica el principio de la medición continua con el ejemplo de una medición de frecuencia.



Control de puerta

Para controlar el módulo 1Count24V se han de usar las funciones de puerta.

Modo isócrono

En el modo isócrono, el 1Count24V acepta en cada ciclo de bus bits de control y valores de control de la interfaz de control y notifica la respuesta durante el mismo ciclo.

El 1Count24V transfiere en cada ciclo un valor medido y los bits de estado tal y como eran válidos en el instante T_i.

La medición empieza y acaba en cada caso en el instante T_i.

Consulte también

Tiempo de integración y tiempo de actualización en el modo isócrono (Página 9648)

Tiempo de integración y tiempo de actualización en el modo isócrono

Tiempo de integración y tiempo de actualización en el modo isócrono

Si el tiempo de integración/actualización dura varios ciclos T_{DP}, es posible reconocer el nuevo valor medido en el programa de usuario por el bit de estado (Página 9657)STS_CMP1 (medición finalizada) de la interfaz de realimentación. Esto permite una vigilancia del proceso de medición o una sincronización con el proceso de medición. Sin embargo, el acuse de este aviso dura 4 ciclos T_{DP}. En este caso el tiempo de integración/actualización mínimo será de (4 x T_{DP}).

Si la aplicación puede tolerar un jitter del tiempo de integración/actualización de un T_{DP} y un valor medido constante durante varios ciclos, no es necesario evaluar constantemente el bit de estado STS_CMP1. Entonces serán posibles tiempos de integración/actualización de (1 x T_{DP}) a (3 x T_{DP}).

Debido a una pérdida del sincronismo en el último ciclo T_{DP} del tiempo de integración se prolonga el tiempo de integración/actualización otro ciclo T_{DP}. El valor medido no se ve falsificado por ello.

Nota

No deben excederse los límites del rango del tiempo de integración/actualización (consulte las tablas de cada uno de los modos de medición).

Una violación de los límites del rango de valores provoca un error de parametrización, con lo cual el 1Count24V no cambia al modo isócrono.

Nota

Al cambiar la configuración del modo no isócrono al modo isócrono y viceversa, siempre deberá adaptarse el parámetro Tiempo de integración/actualización si se desea mantener la duración del tiempo de integración/actualización.

Seleccionar filtro para señales de entrada

Filtro de entrada

Para filtrar las señales en la entrada de contaje A*, la entrada de sentido B* y la entrada digital DI, se puede activar un filtro en función de la duración mínima del impulso o bien de la frecuencia máxima de señales.

Para cada entrada puede seleccionarse un ancho de impulso mínimo de 2,5 µs o 25 µs.

Seleccionar entradas del encóder

Entradas del encóder

En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/simétrica o salida tipo NPN).

Nota

Si se ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Comportamiento de la salida en los modos de medición

Seleccionar comportamiento de la salida DO1

Comportamiento de la salida en los modos de medición

La salida digital del 1Count24V es parametrizable.

Para la medición de frecuencia, velocidad o período se puede indicar un límite superior y un límite inferior, respectivamente, que al ser excedido active la salida digital DO1. Estos valores límite son parametrizables y pueden ser modificados con la función de carga.

La función y el comportamiento de la salida digital se puede modificar durante el funcionamiento. La nueva función actúa de inmediato, en el modo isócrono siempre en el instante T_i.

En la lista desplegable se pueden seleccionar las siguientes funciones:

- Salida (Página 9650)
- Fuera de los valores límite (Página 9650) (vigilancia de valor límite)
- Rebase por defecto del valor límite inferior (Página 9650) (vigilancia de valor límite)
- Rebase por exceso del valor límite superior (Página 9650) (vigilancia de valor límite)

Salida

Salida

Para conectar o desconectar la salida, es necesario habilitarla con el bit de control (Página 9655)CTRL_DO1.

Con el bit de control (Página 9655)SET_DO1 se conecta y desconecta la salida.

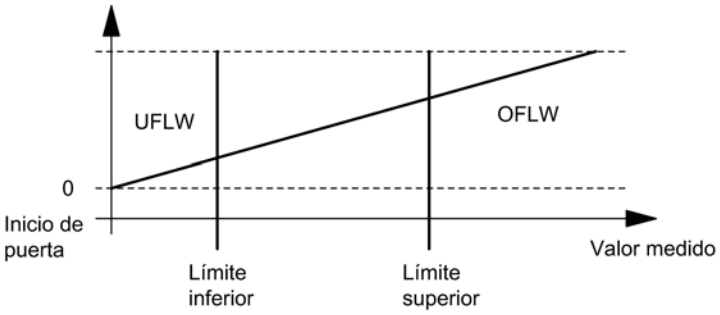
Con el bit de estado (Página 9657)STS_DO1 de la interfaz de realimentación se puede consultar el estado de la salida.

En el modo isócrono la salida conmuta en el instante T_o.

Vigilancia de valores límite

Vigilancia de valor límite

La figura siguiente muestra el principio de la vigilancia de valor límite.



Una vez transcurrido el tiempo de integración, el valor medido (frecuencia, velocidad o período) se compara con los valores límite parametrizados.

Si el valor medido actual está por debajo del valor límite inferior parametrizado (valor medido < valor límite inferior), el bit STS_UFLW = 1 se activa en la interfaz de notificación (Página 9657).

Si el valor medido actual está por encima del valor límite superior parametrizado (valor medido < valor límite superior), el bit STS_OFLW = 1 se activa en la interfaz de realimentación (Página 9657).

Estos bits tienen que ser acusados con el bit de control (Página 9655)RES_STS.

Si el valor medido sigue fuera de los valores límite o vuelve a estarlo después del realimentación, se vuelve a activar el correspondiente bit de estado.

Poniendo el límite inferior a 0, se desconecta la vigilancia dinámica de desbordamiento por defecto del valor límite inferior.

La tabla siguiente muestra que, dependiendo de la parametrización, la salida digital DO1 habilitada se puede activar con la vigilancia de valor límite:

Parámetro "Función de DO1"	DO1 se activa cuando ...
Fuera de los valores límite	Valor medido < valor límite inferior O Valor medido > valor límite superior
Rebase por defecto del valor límite inferior	Valor medido < valor límite inferior
Rebase por exceso del valor límite superior	Valor medido > valor límite superior

En el modo isócrono la salida conmuta al terminar la medición en el instante T_i.

Seleccionar valor sustitutivo DO1

Valor sustitutivo DO1

Si se ha seleccionado el parámetro de conmutar valor sustitutivo DO1 en "Comportamiento en STOP de la CPU/del maestro (Página 9638)", seleccione aquí el valor sustitutivo que debe enviarse a la salida digital DO1.

Activar diagnóstico DO1

Diagnóstico DO1

El "diagnóstico DO1" (rotura de hilo, cortocircuito) sólo es posible con anchos de impulso > 90 ms en la salida digital DO1.

Active la casilla de verificación para el diagnóstico de la salida DO1.

Seleccionar modo de medición

Descripción

Al preseleccionar un modo de medición se elige la funcionalidad con la que se utilizará el 1Count24V.

	Descripción	
Modo de operación	**Medición con tiempo de integración**	**Medición continua**
Medición de la frecuencia (Página 9478) (con puerta HW o SW)	El 1Count24V cuenta los impulsos que llegan en un tiempo de integración predefinido.	El 1Count24V cuenta los impulsos que llegan en un tiempo de medición dinámico.
Medición de la velocidad (Página 9480) (con puerta HW o SW)	El 1Count24V cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de integración predefinido y calcula a partir de ello la velocidad con los impulsos por vuelta del encóder.	El 1Count24V cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de medición dinámico y calcula a partir de ello la velocidad con los impulsos por vuelta del encóder.
Medición del período (Página 9481) (con puerta HW o SW)	El 1Count24V mide el tiempo entre dos flancos ascendentes de la señal de contaje contando los impulsos de una frecuencia interna de referencia (16 MHz) de precisión de cuarzo dentro del tiempo de integración dado.	El 1Count24V indica el tiempo de medición dinámico como duración del período. Si el período es inferior al tiempo de actualización, se crea un valor medio para el período.

El ajuste predeterminado es el modo de operación "Medición de frecuencia".

Seleccionar la resolución del período

Descripción

El valor del período medido se indica en 1 µs o 1/16 µs.

Seleccione la unidad deseada para la resolución.

Rangos de medición posibles con información de errores

- Medición del período con tiempo de integración
  - Rangos de medición y errores a una resolución de 1 µs:

Tiempo de integración	$T_{mín}$ ± error absoluto	$T_{mín}$ ± error absoluto
100 s	1 µs* (10 ± 0)	1 µs* (100.000.000 ± 10.000)
10 s	1 µs* (10 ± 0)	1 µs* (10.000.000 ± 1.000)
1 s	1 µs* (10 ± 0)	1 µs* (1.000.000 ± 100)
0,1 s	1 µs* (10 ± 0)	1 µs* (100.000 ± 10)
0,01 s	1 µs* (10 ± 0)	1 µs* (10.000 ± 1)

  - Rangos de medición y errores en una resolución de 1/16 µs:

Tiempo de integración	$T_{mín}$ ± error absoluto	$T_{mín}$ ± error absoluto
100 s	1/16 µs* (160 ± 0)	1/16 µs* (1.600.000.000 ± 160.000)
10 s	1/16 µs* (160 ± 0)	1/16 µs* (160.000.000 ± 16.000)
1 s	1/16 µs* (160 ± 0)	1/16 µs* (16.000.000 ± 1.600)
0,1 s	1/16 µs* (160 ± 0)	1/16 µs* (1.600.000 ± 160)
0,01 s	1/16 µs* (160 ± 0)	1/16 µs* (160.000 ± 16)

- Medición continua del período:
  - Rangos de medición y errores a una resolución de 1 µs:

Período $T_{mín}$ ± error absoluto	Período $T_{mín}$ ± error absoluto
1 µs* (10 ± 0)	1 µs* (100 000 ± 10)
1 µs* (100 ± 0)	1 µs* (1 000 000 ± 100)
1 µs* (1 000 ± 0)	1 µs* (10 000 000 ± 1 002)
1 µs* (10 000 ± 1)	1 µs* (100 000 000 ± 10 020)

  - Rangos de medición y errores en una resolución de 1/16 µs:

Período $T_{mín}$ ± error absoluto	Período $T_{mín}$ ± error absoluto
1/16 µs* (160 ± 1)	1/16 µs* (1 600 000 ± 160)
1/16 µs* (1 600 ± 1)	1/16 µs* (16 000 000 ± 1 600)
1/16 µs* (16 000 ± 3)	1/16 µs* (160 000 000 ± 16 000)
1/16 µs* (160 000 ± 20)	1/16 µs* (1 600 000 000 ± 160 000)

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital.

El ajuste predeterminado es "Entrada".

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9657)STS_DI indica el nivel de la entrada digital.

Definir valor límite inferior y superior

Valores límite inferior y superior

Para la medición de frecuencia, velocidad o período se puede indicar un límite superior y un límite inferior, respectivamente, que al ser excedido active la salida digital DO1 (Página 9646). Estos valores límite se pueden parametrizar y modificar a través de la función de carga.

Los siguientes rangos de valores están permitidos para los valores límite inferior y superior:

Modo de medición	Valor límite inferior (LI):	Valor límite superior
Medición de frecuencia	0 a 99.999.999 × 10^{-3} Hz	LI+1 a 100.000.000 × 10^{-3} Hz
Medición de velocidad	0 a 24.999.999 × 10^{-3} /min	LI+1 a 25.000.000 × 10^{-3}/min
Medición del período (resolución 1 µs)	0 a 119.999.999 µs	LI+1 a 120.000.000 µs
Medición del período (resolución 1/16 µs)	0 a 1.919.999.999 µs	LI+1 a 1.920.000.000 µs

Definir el tiempo de medición

Descripción

Si se ha seleccionado un modo de medición, el módulo actualiza los valores medidos de forma cíclica. El tiempo de actualización/integración es el producto del tiempo de medición x 10 ms. El tiempo de medición se predefine como factor n.

Si desea obtener una descripción detallada sobre el rango de valores para el tiempo de medición, consulte el apartado "Conceptos básicos sobre la operación de medición (Página 9474)".

Introducir impulsos por vuelta del encóder

Impulsos por vuelta del encóder

Para el modo Medición de velocidad (Página 9480) es preciso parametrizar adicionalmente los impulsos por vuelta del encóder.

El valor para el número de impulsos por vuelta del encóder se encuentra en la placa de características o en los datos técnicos del encóder.

Interfaz de notificación y control para los modos de medición

Interfaz de control de los modos de medición

Nota

Para el 1Count24V, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del 1Count24V para los modos de medición.

Dirección	Asignación			
Bytes 0 a 3	**Valor límite inferior o superior**			
	Función de DO1			
	Byte 0	Bit 1	Bit 0	Función DO1
		0	0	Salida
		0	1	Fuera de los valores límite
		1	0	Rebase por defecto del valor límite inferior
		1	1	Rebase por exceso del valor límite superior
	Bytes 1 a 3	reservado = 0		
	Tiempo de integración			
	Byte 0, 1	Tiempo de integración [n × 10 ms] (Rango de valores 1 … 1.000 / 12.000)		
	Byte 2, 3	reservado = 0		

Dirección	Asignación		
			Denominación
Byte 4	Bit 7	Acuse del error	EXTF_ACK
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	Habilitación de DO1	CTRL_DO1
	Bit 3	Bit de control DO1	SET_DO1
	Bit 2	Inicio de la desactivación del bit de estado	RES_STS
	Bit 1	reservado = 0	
	Bit 0	Bit de control puerta SW	SW_GATE
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	Modificar la función de DO1	C_DOPARAM
	Bit 3	reservado = 0	
	Bit 2	Modificar el tiempo de integración	C_INTTIME
	Bit 1	Cargar valor límite superior	LOAD_PREPARE
	Bit 0	Cargar valor límite inferior	LOAD_VAL
Bytes 6 a 7	reservado = 0 [1]		

[1] Falta en la interfaz de datos de usuario avanzada

Significado de los bits de control

Bit	Significado
C_DOPARAM	Modificar la función de DO1 El valor del byte 0 es aceptado como nueva función de DO1.
C_INTTIME	Modificar el tiempo de integración El valor de los bytes 0 y 1 es aceptado como nuevo tiempo de integración para la siguiente medición.
CTRL_DO1	Habilitación de DO1 Este bit habilita la salida DO1
EXTF_ACK	Acuse del error Los bits de error deben ser acusados mediante el bit de control EXTF_ACK una vez eliminada la causa.
LOAD_PREPARE	Cargar valor límite superior El valor de los bytes 0 a 3 es aceptado como nuevo límite superior.
LOAD_VAL	Cargar valor límite inferior El valor de los bytes 0 a 3 es aceptado como nuevo valor límite inferior.
RES_STS	Inicio de la desactivación del bit de estado Los bits de estado se desactivan mediante el proceso de realimentación entre el bit RES_STS y el bit RES_STS_A.
SET_DO1	Bit de control DO1 Conectar y desconectar la salida digital DO1 cuando se ha activado CRTL_DO1.
SW_GATE	Bit de control puerta SW La puerta SW se abre/cierra mediante la interfaz de control con el bit SW_GATE.

Interfaz de realimentación de los modos de medición

Nota

Para el 1Count24V, los datos siguientes de la interfaz de realimentación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 1Count24V para los modos de medición.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor medido		
Byte 4	Bit 7	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 6	Cortocircuito / rotura de hilo / sobretemperatura	ERR_DO1
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	Desactivación de los bits de estado en curso	RES_STS_A
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Byte 5	Bit 7	Estado de sentido descendente	STS_C_DN
	Bit 6	Estado de sentido ascendente	STS_C_UP
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	Estado de DO1	STS_DO1
	Bit 2	reservado = 0	
	Bit 1	Estado de DI	STS_DI
	Bit 0	Estado de la puerta interna	STS_GATE
Byte 6	Bit 7	reservado = 0	
	Bit 6	Valor límite inferior del rango de medida	STS_UFLW
	Bit 5	Valor límite superior del rango de medida	STS_OFLW
	Bit 4	reservado = 0	
	Bit 3	Fin de la medición	STS_CMP1
	Bit 2	reservado = 0	
	Bit 1	reservado = 0	
	Bit 0	reservado = 0	
Byte 7	reservado = 0		
Bytes 8 a 11	Valor de contaje [1]		

[1] Interfaz de datos de usuario avanzada

Nota sobre los bits de realimentación

Bit	Significado
ERR_24V	Cortocircuito en la alimentación del encóder El bit de error tiene que ser acusado con el bit de control (Página 9652)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_DO1	Cortocircuito/rotura de hilo/sobretemperatura en la salida DO1 El bit de error tiene que ser acusado con el bit de control (Página 9652)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_LOAD	Error en función de carga Los bits de control (Página 9652)LOAD_VAL, LOAD_PREPARE, C_DOPARAM y C_INTTIME no pueden estar activados al mismo tiempo durante la transferencia. Esto tiene como consecuencia la activación del bit de estado ERR_LOAD, similar a cargar un valor incorrecto (que no se acepta).
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
RES_STS_A	Desactivación de los bits de estado en curso
STS_C_DN	Estado de sentido descendente
STS_C_UP	Estado de sentido ascendente
STS_CMP1	Fin de la medición El valor medido es actualizado cada vez que termina un intervalo (tiempo de actualización/integración). **Medición con tiempo de integración:** El fin de una medición (al terminar el intervalo) es indicado con el bit de estado STS_CMP1. **Medición continua:** Al finalizar el tiempo de actualización, el fin de la medición se indica con el bit de estado STS_CMP1, siempre que se emita un valor medido. Si se emite un valor medido aproximado, el bit se mantiene en "0". Este bit se desactiva con el bit de control (Página 9652)RES_STS en la interfaz de control.
STS_DI	Estado de DI El estado de la DI se indica en todos los modos de operación con el bit STS_DI en la interfaz de realimentación.
STS_DO1	Estado de DO1
STS_GATE	Estado de la puerta interna: Se mide.
STS_LOAD	Función de carga en curso
STS_OFLW STS_UFLW	Valor límite superior rebasado Valor límite inferior no alcanzado Ambos bits deben ser desactivados.

Parámetros para la lectura del recorrido

Activar diagnóstico colectivo

Diagnóstico colectivo

Si activa el diagnóstico colectivo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar la reacción en caso de STOP de la CPU/del maestro

Ajuste de la reacción en caso de STOP de la CPU/del maestro

Es posible parametrizar el comportamiento del 1Count24V en caso de fallo del controlador principal.

Parámetro	Estado del 1Count24V en caso de STOP de la CPU/ del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO 1	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual continúa. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/ el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count24V.

Abandonar el estado parametrizado

¿En qué condiciones abandona el 1Count24V el estado parametrizado?

La CPU o el maestro debe estar en modo RUN y la interfaz de control (Página 9667) debe modificarse.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/del maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count24V
- POWER ON o inserción del módulo de potencia correspondiente

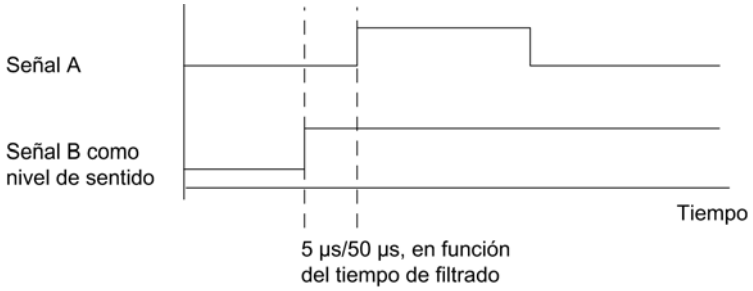
Interpretación de la señal

Seleccionar evaluación de contaje y sentido

Evaluación de señal A*, B*

La evaluación de la señal mediante A*, B* permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Impulso y sentido (Página 9661)
  En los encóders incrementales de posición de 24 V con nivel de sentido debe garantizarse que entre la señal de sentido B* y la señal de contaje A* se observe un intervalo de como mínimo 5 µs/50 µs, según el filtro de entrada parametrizado.
  La figura siguiente muestra el intervalo de tiempo entre la señal de sentido y la señal de contaje.



- Encóder rotativo (Página 9494)
  Si conecta un encóder rotativo de 24 V con dos pistas desfasadas 90 grados en la entrada de contaje y sentido, es posible parametrizar una evaluación simple (Página 9495) en todos los modos de contaje. Además se puede parametrizar una evaluación doble (Página 9495) o cuádruple (Página 9495).
  En todos los modos de evaluación se puede invertir el sentido de contaje (Página 9662) en la entrada de sentido B*.
  En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/ simétrica o salida tipo NPN).

Nota

Si con el 1Count24V ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Nota

La indicación 100 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A* o B*. Así, en la evaluación doble se obtienen como máximo 200 kHz en los impulsos de contaje o máximo 400 kHz en la evaluación cuádruple.

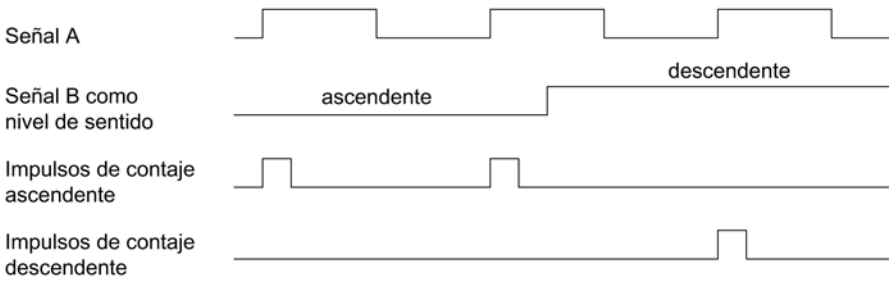
Evaluación "Impulso y sentido"

Impulso y sentido

En este modo de valuación se utiliza como indicador de sentido el nivel en la entrada de sentido B*.

Una entrada no cableada equivale al sentido de contaje "ascendente", si en el parámetro Evaluación de señal ha elegido "Impulso y sentido".

La figura siguiente muestra las señales de un encóder incremental de posición de 24 V con nivel de sentido.



Seleccionar filtro para señales de entrada

Filtro de entrada

Para filtrar las señales en la entrada de contaje A*, la entrada de sentido B* y la entrada digital DI, se puede activar un filtro en función de la duración mínima del impulso o bien de la frecuencia máxima de señales.

Para cada entrada puede seleccionarse un ancho de impulso mínimo de 2,5 µs o 25 µs.

Seleccionar entradas del encóder

Entradas del encóder

En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/simétrica o salida tipo NPN).

Nota

Si se ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Función DI

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital.
- Congelar/redisparar con flanco ascendente (Página 9663)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador. Después de la operación de guardado, el contador se establece al valor de carga.
- Sincronización con flanco ascendente (Página 9664)
  Los flancos ascendentes de la entrada digital permiten establecer el contador al valor de carga.
- Congelar con flanco ascendente (Página 9665)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador.

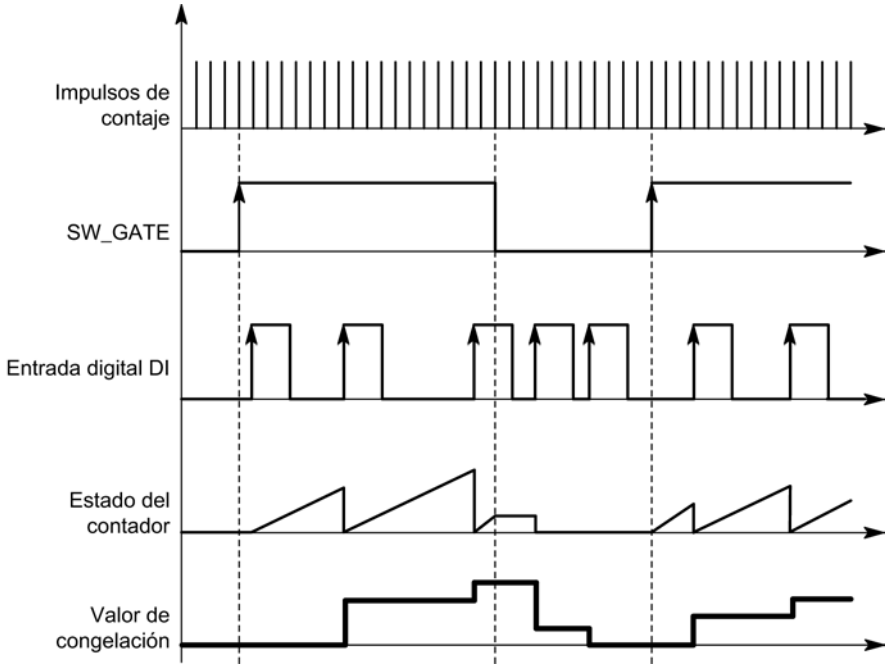
El ajuste predeterminado es "Entrada".

Congelar/redisparar con flanco ascendente

Congelar/redisparar

Con esta función se almacena el estado actual interno del contador del 1Count24V y se redispara el contaje cuando hay un flanco ascendente en la entrada digital. Esto significa que se almacena el estado actual interno del contador en el momento del flanco ascendente (valor de congelación) y, entonces, se carga nuevamente el módulo 1Count24V con el valor de carga y se sigue contando desde allí.

La siguiente figura muestra la función de congelar/redisparar con el valor de carga = 0.



Para poder ejecutar la función, el modo de contaje tiene que estar habilitado con la puerta SW. Se inicia con el primer flanco positivo de la entrada digital.

En la interfaz de realimentación (Página 9668) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el estado de la señal de congelación y redisparo.

El valor de congelación se pone por defecto al estado de RESET ("0"). No cambia cuando se abre la puerta SW.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

Si se cierra la puerta SW, sólo se interrumpe el contaje; es decir, cuando se vuelve a abrir la puerta SW, continúa el contaje. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

También en el modo isócrono se congela y redispara el contaje con cada flanco de la entrada digital. En la interfaz de realimentación se muestra el estado del contador válido en el instante del último flanco antes de T_i.

Interfaz de realimentación ampliada

Si el módulo 1Count24V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces de datos de usuario más anchas, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de realimentación (Página 9668).

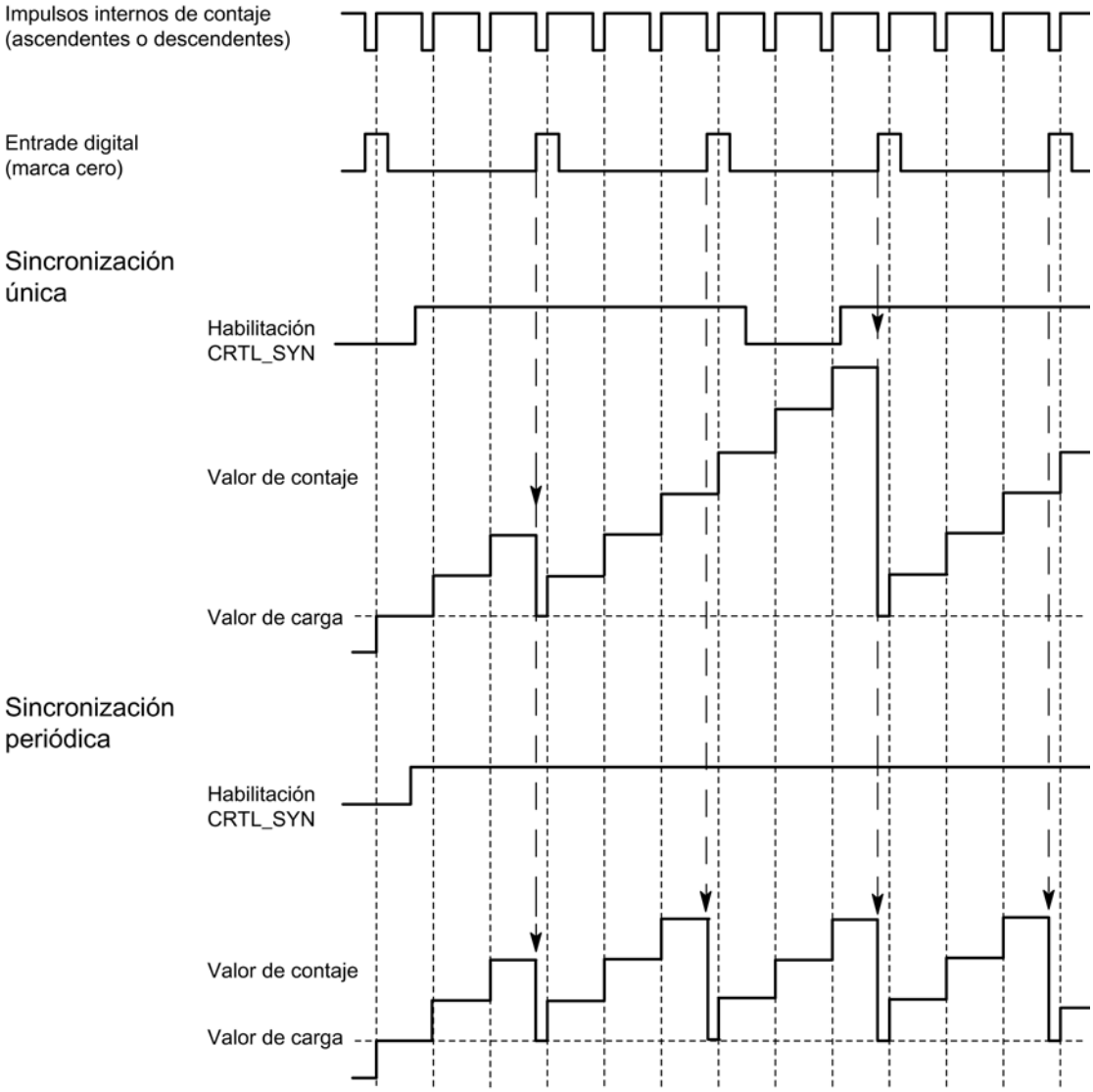
Sincronización con flanco ascendente

Sincronización

Si se ha parametrizado "Sincronización con flanco ascendente", el flanco ascendente de una señal de referencia en la entrada sirve para poner el 1Count24V al valor de carga.

Se puede seleccionar entre la sincronización única y la periódica (parámetro "Sincronización").

La figura siguiente muestra los estados del contador con una sincronización única y periódica.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control (Página 9667) "Habilitación de la sincronización" (CTRL_SYN) debe estar activado.
- Sincronización única: si el bit de habilitación está activado, el primer flanco carga el 1Count24V con el valor de carga.
- Sincronización periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count24V con el valor de carga.
- Una vez realizada la sincronización correctamente, el bit de realimentación (Página 9668)STS_SYN queda activado. Debe desactivarse con el bit de control (Página 9667)RES_STS.
- Como señal de referencia se puede utilizar la señal de un interruptor libre de rebotes o la marca cero de un encóder rotativo.
- El bit de realimentación (Página 9668)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación (Página 9668) activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

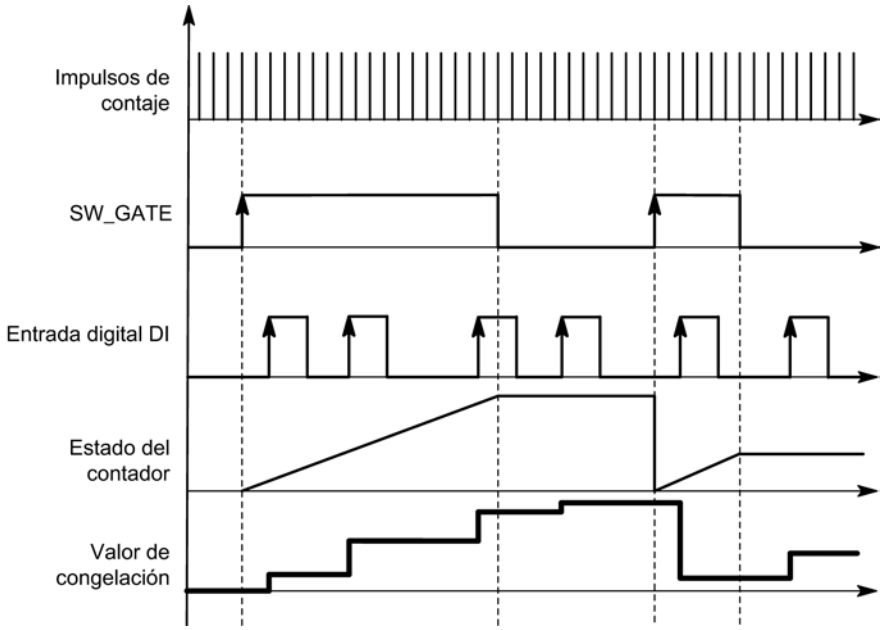
Congelar con flanco ascendente

Congelar

El estado del contador y el valor de congelación reciben sus estados de RESET ("0"):

La función de contaje se inicia cuando se abre la puerta SW. El 1Count24V comienza a contar a partir del valor de carga ("0").

La siguiente figura muestra la función de congelar con valor de carga = 0.



El valor de congelación equivale siempre al valor de contaje en el instante del flanco ascendente de la entrada digital DI.

En la interfaz de realimentación (Página 9668) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el nivel de la señal de congelación.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

En el modo isócrono, se muestra en la interfaz de realimentación el estado del contador que se congeló en el instante del último flanco ascendente antes de T_i.

Si se cierra la puerta SW, actúa como parametrizada, de cancelación o de interrupción. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

Interfaz de realimentación ampliada

Si el módulo 1Count24V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de realimentación (Página 9668).

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9668)STS_DI indica el nivel de la entrada digital.

Seleccionar sincronización

Sincronización

Si se ha seleccionado "Sincronización con flanco ascendente" en "Función DI (Página 9659)", es posible elegir entre la sincronización única y la periódica.

- Una vez: si el bit de habilitación está activado, el primer flanco carga el 1Count24V con el valor de carga.
- Periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count24V con el valor de carga.

Interfaz de realimentación y control para la lectura del recorrido

Interfaz de control para la lectura del recorrido

Nota

Para el 1Count24V, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (las salidas) del 1Count24V para la lectura del recorrido.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor de carga directo, preliminar		
Byte 4	Bit 7	Acuse del error	EXTF_ACK
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	Inicio de la desactivación del bit de estado	RES_STS
	Bit 1	Habilitación de la sincronización	CTRL_SYN
	Bit 0	Bit de control puerta SW	SW_GATE
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	

Dirección	Asignación		Denominación
	Bit 2	reservado = 0	
	Bit 1	Carga preliminar del contador	LOAD_PREPARE
	Bit 0	Carga directa del contador	LOAD_VAL
Bytes 6 a 7	reservado = 0 [1]		

[1] Falta en la interfaz de datos de usuario avanzada

Significado de los bits de control

Bit	Significado
CTRL_SYN	Este bit habilita la sincronización
EXTF_ACK	Acuse del error Los bits de error deben ser acusados mediante el bit de control EXTF_ACK una vez eliminada la causa.
LOAD_PREPARE	Carga preliminar del contador El valor de los bytes 0 a 3 se acepta como valor de carga
LOAD_VAL	El valor de los bytes 0 a 3 se carga directamente como nuevo valor de contaje.
RES_STS	Inicio de la desactivación del bit de estado Los bits de estado se desactivan mediante el proceso de realimentación entre el bit RES_STS y el bit RES_STS_A.
SW_GATE	La puerta SW se abre/cierra mediante la interfaz de control con el bit SW_GATE.

Interfaz de realimentación para la lectura del recorrido

Nota

Para el 1Count24V, los datos siguientes de la interfaz de realimentación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 1Count24V para la lectura del recorrido.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor de contaje o valor de contaje almacenado con función de congelación en la entrada digital		
Byte 4	Bit 7	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 6	reservado = 0	
	Bit 5	Error de parametrización	ERR_PARA

Dirección	Asignación		Denominación
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	Desactivación de los bits de estado en curso	RES_STS_A
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Byte 5	Bit 7	Estado de sentido descendente	STS_C_DN
	Bit 6	Estado de sentido ascendente	STS_C_UP
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	reservado = 0	
	Bit 1	Estado de DI	STS_DI
	Bit 0	Estado de la puerta interna	STS_GATE
Byte 6	Bit 7	Paso por cero	STS_ND
	Bit 6	Límite inferior de contaje	STS_UFLW
	Bit 5	Límite superior de contaje	STS_OFLW
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	reservado = 0	
	Bit 1	reservado = 0	
	Bit 0	Estado de la sincronización	STS_SYN
Byte 7	reservado = 0		
Bytes 8 a 11	Valor de contaje [1]		

[1] Interfaz de datos de usuario avanzada

Nota sobre los bits de realimentación

Bit	Significado
ERR_24V	Cortocircuito en la alimentación del encóder El bit de error tiene que ser acusado con el bit de control (Página 9664)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_LOAD	Error en función de carga Los bits de control (Página 9664)LOAD_VAL y LOAD_PREPARE no pueden estar activados al mismo tiempo durante la transferencia. Esto tiene como consecuencia la activación del bit de estado ERR_LOAD, similar a cargar un valor incorrecto (que no se acepta).
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
RES_STS_A	Desactivación de los bits de estado en curso
STS_C_DN	Estado de sentido descendente
STS_C_UP	Estado de sentido ascendente
STS_DI	Estado de DI El estado de la DI se indica en todos los modos de operación con el bit STS_DI en la interfaz de realimentación.
STS_GATE	Estado de la puerta interna: Contaje en curso.

Bit	Significado
STS_LOAD	Función de carga en curso
STS_ND	Paso por cero en el rango de contaje sin sentido principal de contaje. El bit debe desactivarse con el bit de control (Página 9664)RES_STS.
STS_OFLW STS_UFLW	Límite superior de contaje rebasado Límite inferior de contaje rebasado Ambos bits deben ser desactivados.
STS_SYN	Estado de la sincronización: Al terminar la sincronización se activa el bit STS_SYN. Debe desactivarse con el bit de control (Página 9664)RES_STS.

Parametrizar para el modo Fast

Seleccionar el comportamiento en caso de STOP de la CPU/del maestro

Ajuste del comportamiento en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count24V en caso de fallo del controlador principal.

Parámetro	Estado del 1Count24V en caso de STOP de la CPU/ del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO 1	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual continúa. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/ el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count24V.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/del maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP

- Al insertar el 1Count24V
- POWER ON o inserción del módulo de potencia correspondiente

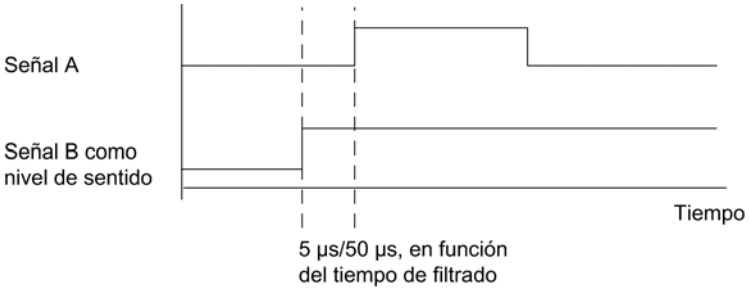
Evaluación de la señal

Seleccionar evaluación de contaje y sentido

Evaluación de señal A*, B*

La evaluación de la señal mediante A*, B* permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Impulso y sentido (Página 9672)
  En los encóders incrementales de posición de 24 V con nivel de sentido debe garantizarse que entre la señal de sentido B* y la señal de contaje A* se observe un intervalo de como mínimo 5 µs/50 µs, según el filtro de entrada parametrizado.
  La figura siguiente muestra el intervalo de tiempo entre la señal de sentido y la señal de contaje.



- Encóder rotativo (Página 9494)
  Si conecta un encóder rotativo de 24 V con dos pistas desfasadas 90 grados en la entrada de contaje y sentido, es posible parametrizar una evaluación simple (Página 9495) en todos los modos de contaje. Además se puede parametrizar una evaluación doble (Página 9495) o cuádruple (Página 9495).
  En todos los modos de evaluación se puede invertir el sentido de contaje (Página 9673) en la entrada de sentido B*.
  En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/ simétrica o salida tipo NPN).

Nota

Si con el 1Count24V ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Nota

La indicación 100 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A* o B*. Así, en la evaluación doble se obtienen como máximo 200 kHz en los impulsos de contaje o máximo 400 kHz en la evaluación cuádruple.

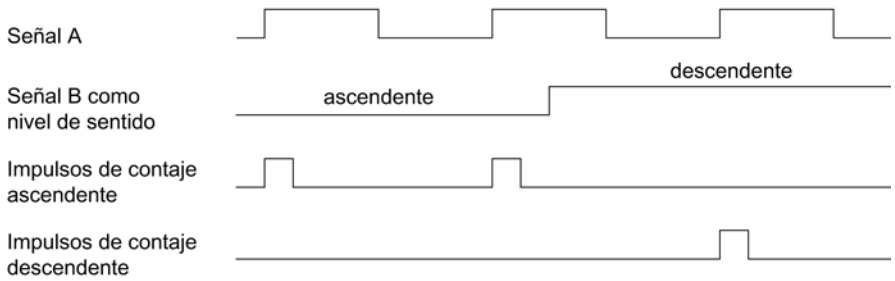
Evaluación "Impulso y sentido"

Impulso y sentido

En este modo de valuación se utiliza como indicador de sentido el nivel en la entrada de sentido B*.

Una entrada no cableada equivale al sentido de contaje "ascendente", si en el parámetro Evaluación de señal ha elegido "Impulso y sentido".

La figura siguiente muestra las señales de un encóder incremental de posición de 24 V con nivel de sentido.



Seleccionar filtro para señales de entrada

Filtro de entrada

Para filtrar las señales en la entrada de contaje A*, la entrada de sentido B* y la entrada digital DI, se puede activar un filtro en función de la duración mínima del impulso o bien de la frecuencia máxima de señales.

Para cada entrada puede seleccionarse un ancho de impulso mínimo de 2,5 µs o 25 µs.

Seleccionar entradas del encóder

Entradas del encóder

En las entradas de encóder es posible conectar diferentes encóders (salida tipo PNP/simétrica o salida tipo NPN).

Nota

Si se ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también encóders con salida tipo NPN.

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Función DI

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital
- Sincronización con flanco ascendente (Página 9673)
  Los flancos ascendentes de la entrada digital permiten establecer el contador al valor de carga.

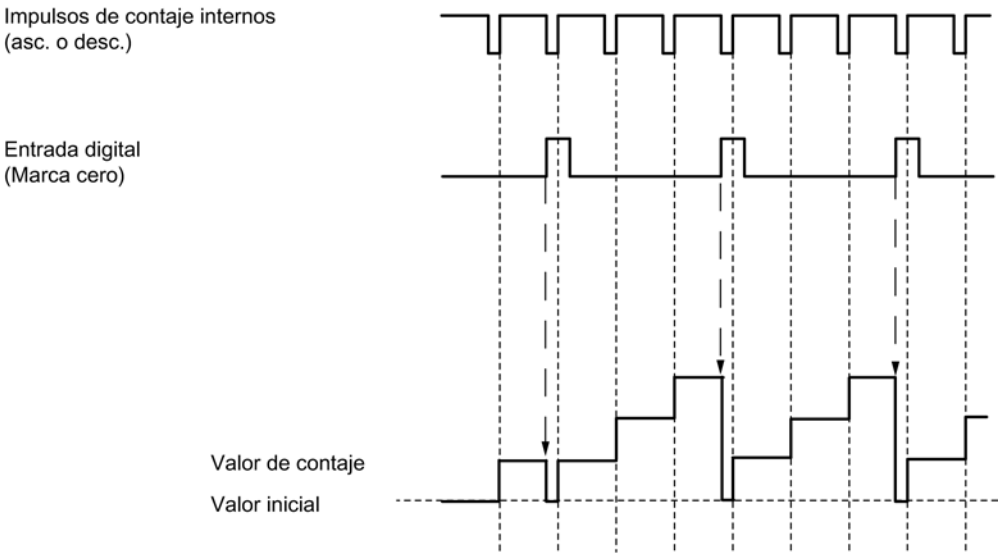
El ajuste predeterminado es "Entrada".

Sincronización con flanco ascendente

Sincronización

Si se ha parametrizado "Sincronización con flanco ascendente", el flanco ascendente de una señal de referencia en la entrada sirve para poner el 1Count24V al valor de inicio (= valor de carga).

La figura siguiente muestra los estados del contador con una sincronización con flanco ascendente.



Se aplican las siguientes condiciones:

- El modo de operación "Fast Mode" debe estar activo (puerta HW).
  - Con la sincronización activada, el primer flanco y cualquier flanco posterior cargan el 1Count24V con el valor inicial (= valor de carga).
- Como señal de referencia puede utilizarse la señal de un interruptor sin rebotes o la marca cero de un encóder rotativo.
- El bit de realimentación STS_DI indica el nivel de la señal de referencia.

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9675)STS_DI indica el nivel de la entrada digital.

Definir valor de carga

Valor de carga

Es posible asignar un valor de carga al 1Count24V.

Dicho valor de carga se aplica directamente como valor inicial.

Rango de valores admisible: -16.777.216 a +16.777.215

Interfaz de realimentación para el modo Fast

Nota

Para el 1Count24V, los datos siguientes de la interfaz de realimentación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 1Count24V para el modo Fast.

Dirección	Asignación		Denominación
Bytes 0 a 3	Bit 31	Señal de vida	LZ
	Bit 30	Modo isócrono registrado	STS_TIC
	Bit 29	Error de parametrización	ERR_PARA
	Bit 28	Error de grupo	EXTF
	Bit 27	Estado de DI	STS_DI
	Bit 26	Estado de sentido ascendente / descendente	STS_DIR
	Bit 25	Estado de la puerta interna	STS_GATE
	Bits 0 a 24	Valor de contaje	

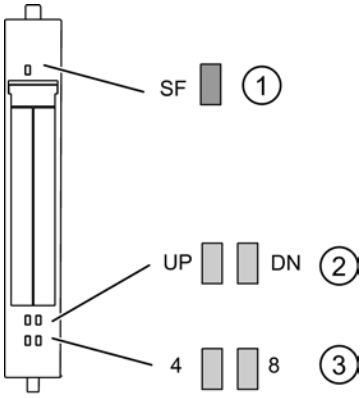
Significado de los bits de realimentación

Bit	Significado
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
EXTF	Error de grupo Causa posible: • Cortocircuito, alimentación de encóders EXTF se desactiva cuando se eliminan las causas del error.
LZ	La señal de vida se invierte en cada actualización de la interfaz de realimentación, es decir, que se invierte el último valor enviado.
STS_DI	El bit muestra el estado de la entrada digital DI.
STS_DIR	Estado de sentido; • cuando cambia el valor del encóder de posiciones elevadas a posiciones más bajas (incl. paso por cero) → "1 " • cuando cambia el valor del encóder de posiciones bajas a posiciones más elevadas (incl. paso por cero) → "0 "
STS_GATE	Estado de la puerta interna: Contaje en curso.
STS_TIC	El modo isócrono (si está parametrizado) se ha registrado.

Asignación del 1Count24V

Diagnóstico con indicadores LED

Indicador LED del 1Count24V



① Error de grupo (rojo)
② Indicador de estado del sentido de contaje (verde)
③ Indicador de estado de entrada digital/salida digital (verde)

Indicadores de estado y error mediante los LEDs del 1Count24V

La tabla siguiente muestra los indicadores de estado y error del 1Count24V.

Evento (LED)					Causa	Solución
SF	UP	DN	4	8		
ON					No hay parametrización o el módulo enchufado es incorrecto. Hay un aviso de diagnóstico.	Compruebe la parametrización. Evalúe el aviso de diagnóstico:
	ON				Estado del bit menos significativo del contador, si el contador cuenta adelante.	
		ON			Estado del bit invertido menos significativo del contador, si el contador cuenta atrás.	
			ON		DO (control directo, salida del comparador) activada.	
				ON	DI (puerta HW, sincronización, latch) activada.	

Tipos de error

En el manual de producto del módulo de interfaz utilizado en la estación ET 200S encontrará información sobre la estructura del diagnóstico de canal.

Tipos de error del 1Count24V

La tabla siguiente muestra los tipos de error del 1Count24V.

Tipo de error		Significado	Solución
1_D	00001: cortocircuito	Cortocircuito de la alimentación del encóder o actuador.	Compruebe el cableado del encóder. Corregir el cableado del proceso.
5_D	00101: Sobretemperatura	Salida digital sobrecargada.	Corregir el cableado del proceso.
6_D	00110: Rotura de hilo	Cable del actuador interrumpido.	Corregir el cableado del proceso.
9_D	01001: Error	Ha aparecido un error interno en el módulo	Sustituir el módulo.
		Tensión de carga insuficiente del módulo de potencia.	Corregir el cableado del proceso. Compruebe la tensión de carga.
16_D	10000: Error de parametrización	Módulo no parametrizado.	Corregir la parametrización.

Utilizar ET 200S 1Count5V

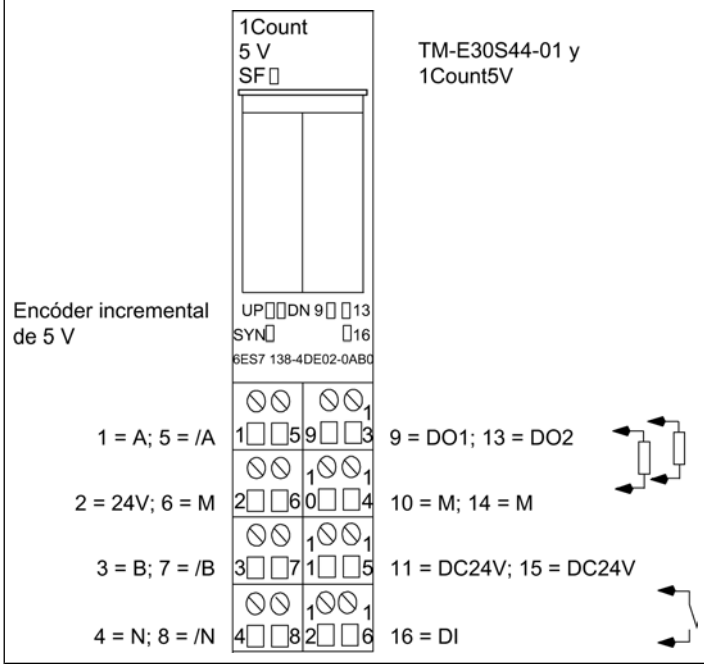
Asignación de conectores del 1Count5V

Reglas de cableado

Los cables (terminales 1 a 8 y terminales 15 y 16) deben estar apantallados. La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contacto (véase el anexo de las instrucciones de servicio Sistema de periferia descentralizada ET 200S (http://support.automation.siemens.com/WW/view/es/1144348)).

Asignación de terminales del 1Count5V

La tabla siguiente muestra la asignación de conectores para el 1Count5V.

Vista	Asignación de terminales	Observaciones
		A, /A: pista A B, /B: pista B N, /N: pista N DC24V: alimentación del encóder M: tierra DI: entrada digital DO1: salida digital DO2: salida digital

Configurar y parametrizar 1Count5V

En servicio

Seleccionar modo de operación

Modo de operación

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el 1Count5V.

Modo de operación	Descripción
Contaje (Página 9679)	Preselección para los modos de contaje
Medición (Página 9702)	Preselección para los modos de medición

Modo de operación	Descripción
Lectura del recorrido (Página 9718)	El 1Count5V cuenta sin fin en este modo de operación a partir del valor de carga: • Cuando el 1Count5V alcanza el límite superior de contaje en la cuenta ascendente, y entonces aparece otro impulso de contaje, salta hasta el límite inferior de contaje y continúa contando desde ese valor sin perder impulsos. • Cuando el 1Count5V alcanza el límite inferior de contaje en la cuenta descendente, y entonces aparece otro impulso, salta hasta el límite superior de contaje y continúa contando desde ese valor sin perder impulsos.
Modo Fast (Página 9730) (a partir de la versión de firmware 2.0)	El 1Count5V cuenta sin fin en este modo de operación a partir del valor de inicio: • Cuando el 1Count5V alcanza el valor máximo representable con 25 bits (todos los bits activados) al contar en sentido ascendente, al llegar un nuevo impulso de contaje el valor de contaje salta a "0" y continúa contando desde ahí sin que se pierda ningún impulso. • Cuando el 1Count5V alcanza el valor "0" al contar en sentido descendente, al llegar un nuevo impulso de contaje el valor de contaje salta al valor máximo representable con 25 bits (todos los bits activados) y continúa contando desde éste sin que se pierda ningún impulso. Este modo de operación es adecuado para la lectura del recorrido en ciclos isócronos especialmente breves.

Parámetros para contaje

Activar diagnóstico colectivo

Diagnóstico colectivo

Si activa el diagnóstico colectivo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar el comportamiento en caso de STOP de la CPU/del maestro

Ajuste del comportamiento en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count5V en caso de fallo del controlador principal.

Parámetro	Estado del 1Count5V en caso de STOP de la CPU/del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual continúa. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.
DO Aplicar valor sustitutivo	El modo de operación actual se cancela. La puerta se cierra y el valor sustitutivo parametrizado para la salida digital se aplica directamente. Los valores de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros. Si para la salida se parametriza "Impulso al alcanzar el valor de comparación", sólo estará disponible el valor sustitutivo = 1 para la duración del impulso.	Los parámetros modificados son aceptados y tienen efecto inmediato.
DO Mantener el último valor	El modo de operación actual se cancela. La puerta se cierra y el estado de la salida digital se conserva. Los valores de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count5V.

Abandonar el estado parametrizado

¿En qué condiciones abandona el 1Count5V el estado parametrizado?

La CPU o el maestro debe estar en modo RUN y la interfaz de control (Página 9698) debe modificarse.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/del maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count5V
- POWER ON o inserción del módulo de potencia correspondiente

Seleccionar evaluación de señal

Evaluación de señal A, B

La evaluación de la señal mediante A, B permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Encóder rotativo (Página 9494)
  El 1Count5V puede contar los flancos de las señales. Normalmente, sólo se evalúan los flancos en A (evaluación simple). Para obtener una resolución más alta, en la parametrización (parámetro "Evaluación de señal") se puede seleccionar si las señales deben evaluarse de forma simple (Página 9495), doble (Página 9495) o cuádruple (Página 9495).
  La evaluación múltiple sólo es posible con encóders incrementales asimétricos con señales A y B desfasadas 90 grados.

  Nota

  La indicación 500 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A o B. Así, en la evaluación doble se obtienen como máximo 1 MHz en los impulsos de contaje o máximo 2 MHz en la evaluación cuádruple.

Activar diagnóstico de los pares de señales

Vigilancia de los pares de señales

Los pares de señales A y /A, B y /B, N y /N pueden vigilarse.

Active para ello la casilla de verificación correspondiente.

Nota

El "Diagnóstico N, /N" no se puede activar hasta que se ha activado el "diagnóstico A, /A y B, /B".

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Comportamiento de las salidas en los modos de contaje

Valores de comparación y salidas

Introducción

En el 1Count5V se pueden almacenar dos valores de comparación que estén asignados a las salidas digitales. Dependiendo del estado del contador y de los valores de comparación se pueden activar las salidas.

Salidas digitales

El 1Count5V dispone de dos salidas digitales.

Ambas salidas pueden parametrizarse (parámetros "Función DO1" y "Función DO2").

La función y el comportamiento de las salidas digitales se puede cambiar durante el funcionamiento.

Controlar las salidas simultáneamente con los comparadores

Si se va a utilizar una función de comparación para las salidas, estas se pueden seguir controlando con SET_DO1 o SET_DO2. De esta manera, se puede simular el efecto de la comparación mediante el programa de control:

- Con el flanco ascendente de SET_DO1 o SET_DO2 se activa la salida.
  Si se ha seleccionado la función "Impulso al alcanzarse el valor de comparación", se emitirá un único impulso de la duración especificada. Si la duración del impulso = 0, la salida puede activarse con SET_DO1 o SET_DO2, mientras el valor de contaje se encuentre en el valor de comparación o la histéresis esté activa.
  El bit de control SET_DO1 no está permitido en el caso de seleccionar el comportamiento de salida "Conmutación en los valores de comparación".
- Un flanco descendente de SET_DO1 o SET_DO2 desactiva la salida de nuevo.

Obsérvese que los comparadores continúan activos y que pueden activar o desactivar la salida si hay un cambio en el resultado de la comparación.

Nota

Una salida activada con SET_DO1 o SET_DO2 no se desactiva en el valor de comparación (con el comparador).

Cargar los valores de comparación

Los valores de comparación se transfieren al 1Count5V. El contaje no se ve afectado por este hecho.

Rango válido para los dos valores de comparación

Sentido principal de contaje: Ninguno	Sentido principal de contaje: Adelante	Sentido principal de contaje: Atrás
Límite inferior de contaje hasta Límite superior de contaje	-2147483648 hasta Límite superior de contaje -1	1 hasta 2147483647

Cambiar la función y el comportamiento de las salidas digitales

Las funciones y el comportamiento de las salidas se pueden cambiar durante el funcionamiento mediante la interfaz de control (Página 9698). Al hacerlo, el módulo 1Count5V desactiva las salidas y aplica los valores de la siguiente manera:

- **Función de las salidas digitales DO1 y DO2:** Si se cambia la función de manera que se cumpla la condición de comparación, la salida no cambiará hasta después del siguiente impulso de contaje. No obstante, si la histéresis (Página 9687) está activada, el módulo 1Count5V no cambia la salida.
- **Histéresis:** Una histéresis (Página 9687) se mantiene activa incluso después de una modificación. La próxima vez que se alcance el valor de comparación, se aplicará el nuevo rango de la histéresis.
- **Duración del impulso:** La nueva duración del impulso (Página 9690) tiene efecto con el siguiente impulso.

Seleccionar el comportamiento de las salidas

Comportamiento de las salidas DO1 y DO2

En la lista desplegable, seleccione el comportamiento deseado de la salida correspondiente.

- Salida (Página 9684)
- Conexión con estado del contador ≥ valor de comparación (Página 9684)
- Conexión con estado del contador ≤ valor de comparación (Página 9684)

- Impulso al alcanzar el valor de comparación (Página 9685)
- Conmutar en los valores de comparación (Página 9685)(sólo DO1)

Ajustar o modificar la función y el comportamiento de la salida digital DO1

Si quiere ajustar o cambiar el comportamiento de la DO1, asegúrese de que se consideran todas las dependencias parametrizadas, pues de lo contrario puede producirse un error de parametrización o un error de carga.

Condiciones marginales:

Si se parametriza "Conmutar al alcanzar los valores de comparación" para la DO1 hay que hacer lo siguiente:

- ajustar la histéresis = 0 y
- además parametrizar también la "Función DO2" como "Salida".

Salida

Salida

Los bits de control (Página 9698)SET_DO1 y SET_DO2 permiten activar y desactivar las salidas.

Para ello, es preciso que los bits de control (Página 9698)CTRL_DO1 o CTRL_DO2 estén activados.

Los bits de estado STS_DO1 y STS_DO2 de la interfaz de realimentación (Página 9700) permiten consultar el estado de las salidas.

Los bits de estado (Página 9700)STS_CMP1 y STS_CMP2 indican que la salida correspondiente está o estaba conectada. Estos bits de estado conservan su estado hasta ser acusados. Si la salida permanece conectada, el correspondiente bit se vuelve a activar inmediatamente. Estos bits de estado se activan también si los bits de control SET_DO1 o SET_DO2 se accionan cuando las salidas DO1 o DO2 no están habilitadas.

Modo isócrono

En el modo isócrono las salidas DO1 y DO2 conmutan en el instante T_o.

Conexión si el estado del contador ≤ valor de comparación o estado del contador ≥ valor de comparación

Estado del contador ≤ valor de comparación y estado del contador ≥ valor de comparación

Cuando las condiciones de comparación se cumplen, el comparador correspondiente activa la salida. El estado de la salida (Página 9700) se indica con STS_DO1 y STS_DO2.

Para ello, es preciso que los bits de control (Página 9698)CTRL_DO1 o CTRL_DO2 estén activados.

El bit de estado (Página 9700)STS_CMP1 o STS_CMP2 indica el resultado de la comparación. Sólo podrá acusar y por tanto desactivar estos bits cuando dejen de cumplirse las condiciones de comparación.

Modo isócrono

También en el modo isócrono las salidas DO1 y DO2 conmutan inmediatamente al cumplirse la condición de comparación y, por consiguiente, no dependen del ciclo de bus.

Impulso al alcanzar el valor de comparación

Impulso al alcanzar el valor de comparación

Cuando el estado del comparador alcanza el valor de comparación, el comparador activa la correspondiente salida digital durante el tiempo parametrizado para el impulso.

Para ello, es preciso que los bits de control (Página 9698)CTRL_DO1 o CTRL_DO2 estén activados.

Los bits de estado (Página 9700)STS_DO1 y STS_DO2 tienen siempre el estado de la correspondiente salida digital.

El resultado de la comparación se indica mediante el bit de estado (Página 9700)STS_CMP1 o STS_CMP2 y sólo se podrá desactivar mediante acuse cuando haya expirado la duración del impulso.

Si se ha parametrizado el sentido principal de contaje, el comparador conmutará sólo cuando se alcance el valor de comparación en el sentido principal de contaje.

Si no se ha parametrizado el sentido principal de contaje, el comparador conmutará sólo cuando se alcance el valor de comparación en cualquiera de los dos sentidos de contaje.

Si la salida digital ha sido activada mediante el bit de control (Página 9698)SET_DO1 o SET_DO2, se desactivará cuando haya transcurrido la duración del impulso.

Modo isócrono

También en el modo isócrono las salidas DO1 y DO2 conmutan inmediatamente al cumplirse la condición de comparación y, por consiguiente, no dependen del ciclo de bus.

Conmutar en los valores de comparación

Conmutar en los valores de comparación

El comparador activa la salida cuando se cumplen las siguientes condiciones:

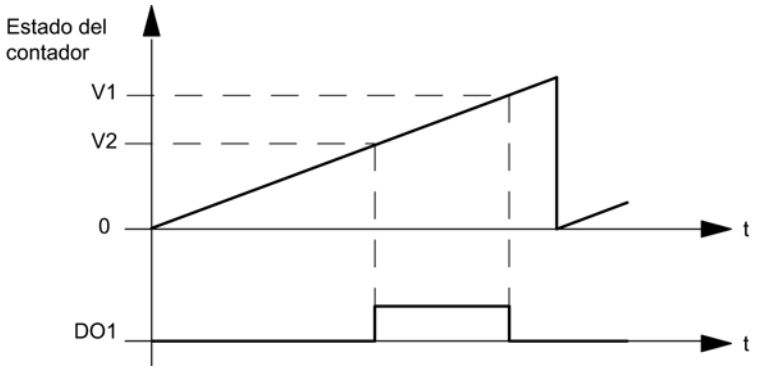
- Los dos valores de comparación deben cargarse mediante la función de carga (Página 9698)CMP_VAL1 y CMP_VAL2 y
- una vez cargados los valores de comparación, hay que habilitar la salida DO1 (Página 9698) con CRTL_DO1.

La tabla siguiente muestra cuándo está activada o desactivada la salida DO1:

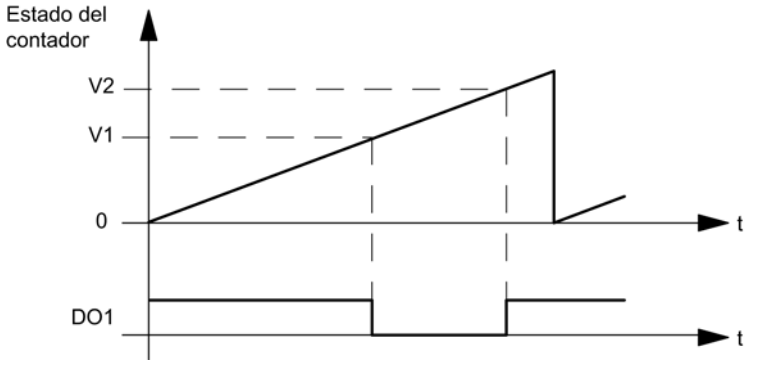
	DO1 está conectada cuando	DO1 está desconectada cuando
V2 < V1 (v. figura inferior)	V2 ≤ estado del contador ≤ V1	V2 > estado del contador o estado del contador > V1
V2 = V1	V2 = estado del contador = V1	V2 ≠ estado del contador ≠ V1
V2 > V1 (v. figura inferior)	V1 > estado del contador o estado del contador > V2	V1 ≤ estado del contador ≤ V2

Las siguientes figuras muestran el estado de la salida DO1 dependiendo del estado del contador y de los valores de comparación:

- V2 < V1 al comenzar el contaje



- V2 > V1 al comenzar el contaje



El bit de estado (Página 9700)STS_CMP1 indica el resultado de la comparación. Sólo podrá acusar y por tanto desactivar este bit cuando deje de cumplirse la condición de comparación.

No hay ninguna histéresis cuando la salida tiene este comportamiento.

En este comportamiento de salida, el bit de control (Página 9698)SET_DO1 no puede controlar la salida DO1.

Modo isócrono

También en el modo isócrono la salida DO1 se activa inmediatamente al cumplirse la condición de comparación y, por consiguiente, no depende del ciclo de bus.

Seleccionar valor sustitutivo de DOx

Valor sustitutivo DO1 y valor sustitutivo DO2

Si se ha seleccionado el parámetro de conmutar valor sustitutivo DO en "Comportamiento en caso de STOP de la CPU/del maestro (Página 9677)", seleccione el valor sustitutivo que debe enviarse a la salida digital DO1 o DO2.

Activar diagnóstico DOx

Diagnóstico DO1 y diagnóstico DO2

El "diagnóstico DO1" o "diagnóstico DO2" (rotura de hilo, cortocircuito) sólo es posible con anchos de impulso > 90 ms en la salida digital DO1 o DO2.

Active la respectiva casilla de verificación para el diagnóstico de la salida DO1 o DO2.

Definir histéresis

Histéresis

Un encóder puede permanecer en una posición particular y fluctuar entonces alrededor de esta posición. Esto hace que el estado del contador fluctúe alrededor de un valor determinado. Si hay un valor de comparación en este rango de fluctuación, por ejemplo, la salida correspondiente se activa y desactiva de acuerdo con el ritmo de las fluctuaciones. Para evitar estas conmutaciones cuando se producen pequeñas oscilaciones, el 1Count5V cuenta con una histéresis parametrizable. Se puede parametrizar un rango entre 0 y 255 (0 significa: histéresis desconectada).

La histéresis también actúa en caso de desbordamiento por exceso y por defecto.

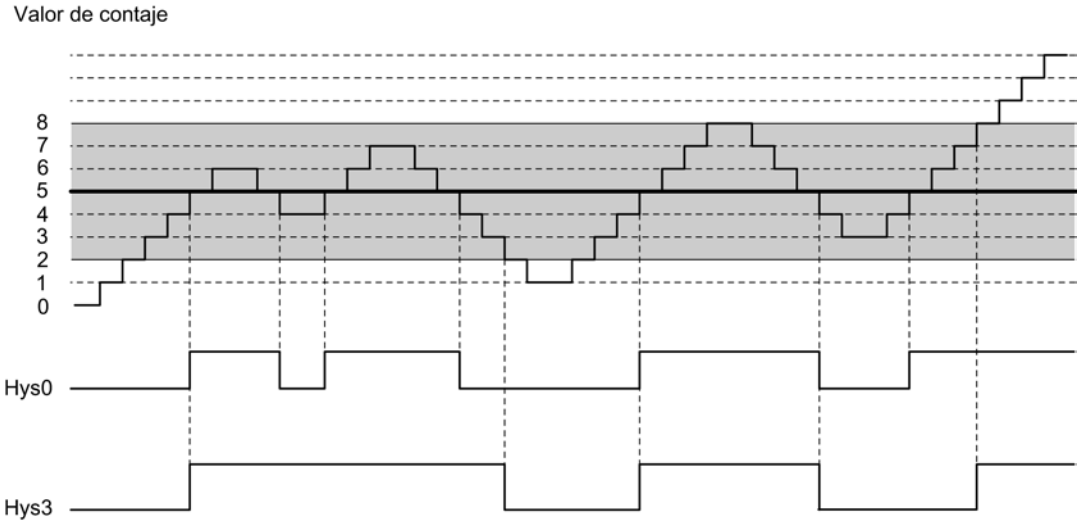
Funcionamiento de la histéresis con el estado del contador ≤ valor de comparación y estado del contador ≥ valor de comparación

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (= desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo, el valor de comparación = 5.

El contador está parametrizado con "Sentido principal de contaje adelante" y "la salida con "Conexión si estado del contador ≥ valor de comparación".

Al alcanzarse la condición de comparación, se activa la histéresis. Cuando la histéresis está activada, el resultado de la comparación permanece invariable.

Si el valor de contaje abandona el rango de histéresis, la histéresis se desactiva. El comparador se conecta de nuevo conforme a sus condiciones de comparación.



Nota

Si se produce un cambio del sentido de contaje en el valor de comparación cuando la histéresis está activa, la salida se desactiva.

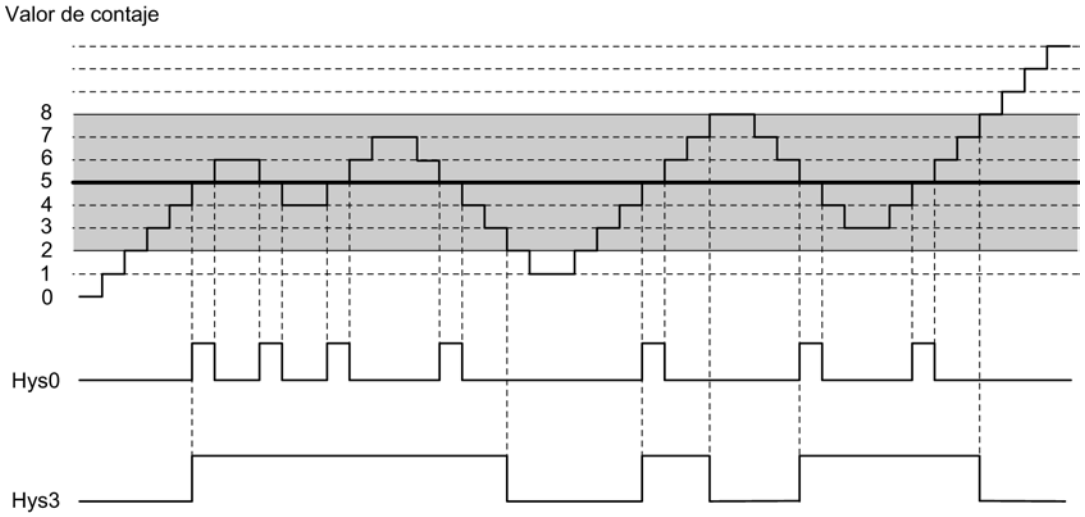
Funcionamiento de la histéresis cuando se alcanza el valor de comparación y la duración del impulso = 0

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo, el valor de comparación = 5.

El contador está parametrizado con "Impulso al alcanzarse el valor de comparación", "Sin sentido principal de contaje" y "Duración del impulso = 0".

Cuando se cumplen las condiciones de comparación se activa la histéresis. Cuando la histéresis está activada, el resultado de la comparación permanece invariable.

Si el valor de contaje abandona el rango de histéresis, la histéresis se desactiva. El comparador borra el resultado de la comparación.



Funcionamiento de la histéresis cuando se alcanza el valor de comparación, salida de la duración del impulso

La figura siguiente muestra un ejemplo del efecto de la histéresis. Muestra las diferencias de comportamiento de una salida cuando se parametriza la histéresis a 0 (desconectada) en contraposición a cuando se parametriza a 3. En el ejemplo, el valor de comparación = 5.

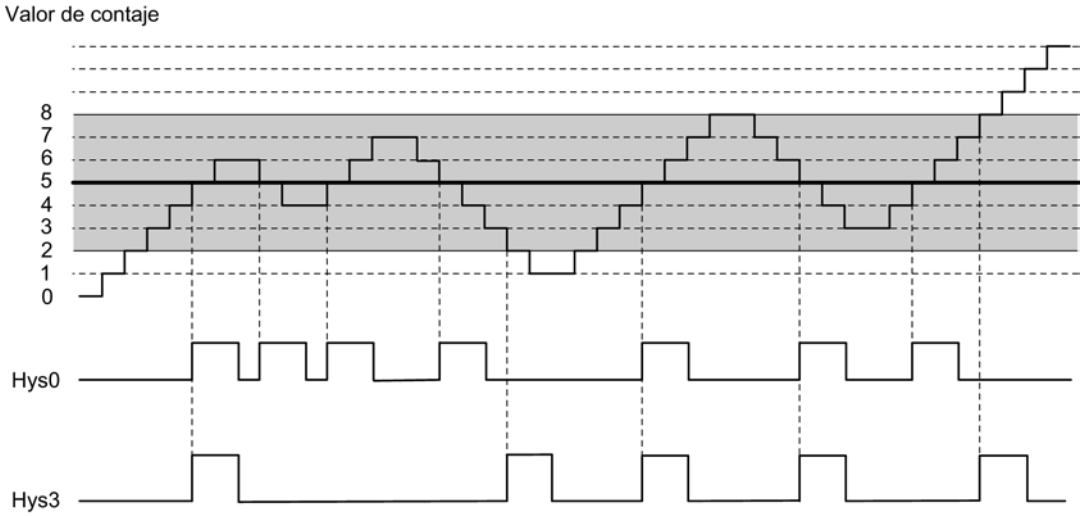
El contador está parametrizado con "Impulso al alcanzarse el valor de comparación", "Sin sentido principal de contaje" y "Duración del impulso > 0".

Cuando las condiciones de comparación se cumplen, la histéresis se activa y se transfiere a la salida un impulso de la duración parametrizada.

Cuando el valor de contaje abandona el rango de histéresis, la histéresis se desactiva.

Cuando se activa la histéresis, el 1Count5V almacena el sentido de contaje.

Si se abandona el rango de histéresis en sentido contrario al almacenado, se emite un impulso.



Definir duración del impulso

Duración del impulso

La duración del impulso comienza al activarse la salida digital correspondiente. La imprecisión de la duración del impulso es inferior a 2 ms.

La duración del impulso puede ser utilizada para conseguir una adaptación a los actuadores utilizados. La duración del impulso indica durante cuánto tiempo ha de estar activada la salida. La duración del impulso puede ser preseleccionada en intervalos de 2 ms entre 0 ms y 510 ms.

Si la duración del impulso es 0, la salida quedará activada hasta que la condición de comparación deje de cumplirse. Observe que los tiempos de impulsos de contaje deben ser mayores que los tiempos mínimos de conexión de la salida digital.

Modo isócrono

También en el modo isócrono las salidas DO1 y DO2 conmutan inmediatamente al cumplirse la condición de comparación y, por consiguiente, no dependen del ciclo de bus.

Seleccionar modo de contaje

Descripción

Al preseleccionar un modo de contaje se elige la funcionalidad con la que se utilizará el 1Count5V.

Modo de operación	Descripción
Contaje sin fin (Página 9468) (con o sin puerta)	El 1Count5V cuenta en modo sin fin a partir del estado actual del contador.
Contaje único (Página 9469) (con puerta HW o SW)	El 1Count5V comienza a contar a partir del valor de carga al abrirse la puerta y se detiene en el límite de contaje.
Contaje periódico (Página 9470) (con puerta HW o SW)	El 1Count5V comienza a contar al abrirse la puerta a partir del valor de carga entre los límites de contaje.

El ajuste predeterminado es el modo de operación "Contaje sin fin".

Selección del sentido principal de contaje

Descripción

Si se parametriza un sentido principal de contaje (Página 9471) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño, ajustando para ello un límite superior de contaje (Página 9697). El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El sentido principal de contaje sólo puede seleccionarse en los modos de operación "Contaje único" y "Contaje periódico".

Se puede seleccionar el siguiente comportamiento:

- Ninguno (sin sentido principal de contaje)
- Adelante
- Atrás

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Función DI

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital
- Congelar/redisparar con flanco ascendente (Página 9692)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador. Después de la operación de guardado, el contador se establece al valor de carga.

- Sincronización con flanco ascendente (Página 9693)
  Los flancos ascendentes de la entrada digital permiten establecer el contador al valor de carga.
- Congelar con flanco ascendente (Página 9695)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador.
- Habilitar HW para sincronización (Página 9696)
  La entrada digital sirve como habilitación HW. Cuando está activada la habilitación HW, el contador se activa en el valor de carga con la señal de paso por cero del encóder.

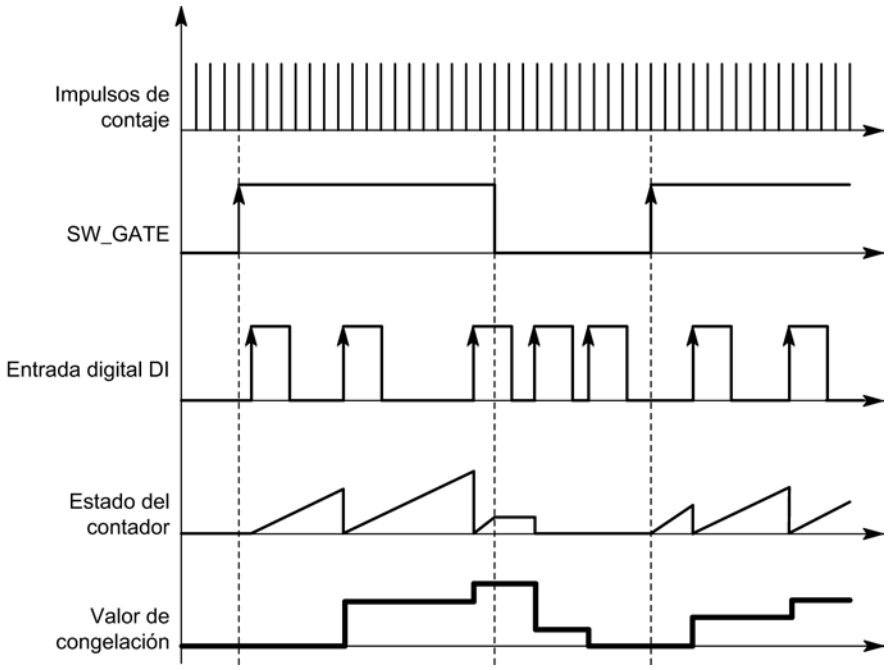
El ajuste predeterminado es "Entrada".

Congelar/redisparar con flanco ascendente

Congelar/redisparar

Con esta función se almacena el estado actual interno del contador del 1Count5V y se redispara el contaje cuando hay un flanco ascendente en la entrada digital. Esto significa que se almacena el estado actual interno del contador en el momento del flanco ascendente (valor de congelación) y, entonces, se carga nuevamente el módulo 1Count5V con el valor de carga y se sigue contando desde allí.

La siguiente figura muestra la función de congelar/redisparar con valor de carga = 0.



Para poder ejecutar la función, el modo de contaje tiene que estar habilitado con la puerta SW. Se inicia con el primer flanco positivo de la entrada digital.

En la interfaz de realimentación (Página 9700) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el estado de la señal de congelación y redisparo.

El valor de congelación se pone por defecto al estado de RESET.

- "0" en el sentido principal de contaje "Ninguno" o "Adelante"
- "Límite de contaje superior parametrizado" en el sentido principal de contaje atrás.

No cambia cuando se abre la puerta SW.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

Si se cierra la puerta SW, sólo se interrumpe el contaje; es decir, cuando se vuelve a abrir la puerta SW, continúa el contaje. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

También en el modo isócrono se congela y redispara el contaje con cada flanco de la entrada digital. En la interfaz de realimentación se muestra el estado del contador válido en el instante del último flanco antes de T_i.

Interfaz de realimentación ampliada

Si el módulo 1Count5V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de realimentación (Página 9700).

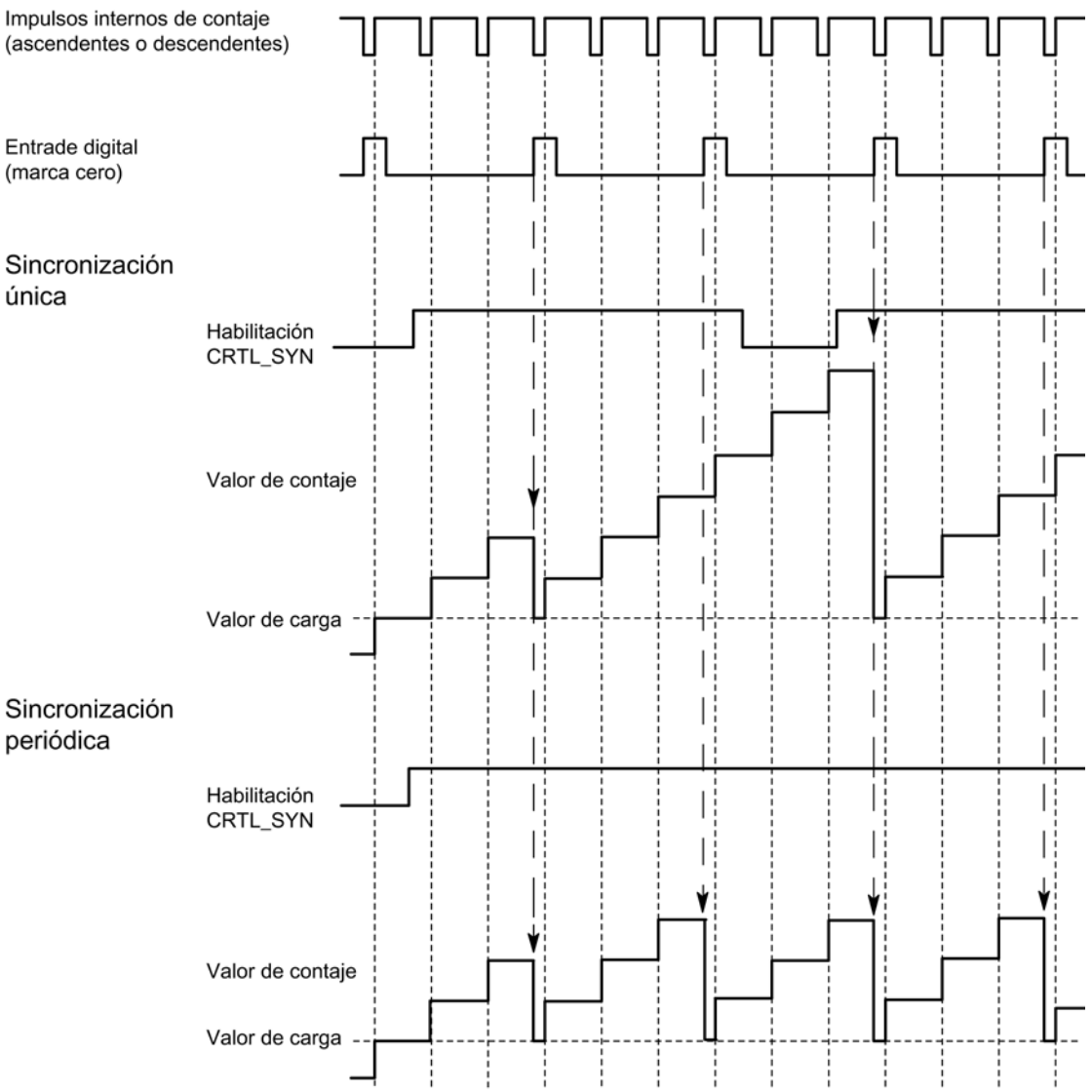
Sincronización con flanco ascendente

Sincronización

Si se ha parametrizado "Sincronización con flanco ascendente", el flanco ascendente de una señal de referencia en la entrada sirve para poner el 1Count5V al valor de carga.

Se puede seleccionar entre la sincronización única y la periódica (parámetro "Sincronización").

La figura siguiente muestra los estados del contador con una sincronización única y periódica.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control (Página 9698) "Habilitación de la sincronización" (CTRL_SYN) debe estar activado.
- Sincronización única: si el bit de habilitación está activado, el primer flanco carga el 1Count5V con el valor de carga.
- Sincronización periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count5V con el valor de carga.
- Tras una sincronización correcta, el bit de realimentación (Página 9700)STS_SYN se activa y el LED SYN se enciende. El bit de control (Página 9698)RES_STS desactiva el bit de realimentación y apaga el LED.
- La señal de un interruptor libre de rebotes puede servir de señal de referencia.
- El bit de realimentación (Página 9700)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación (Página 9700) activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

Congelar con flanco ascendente

Congelar

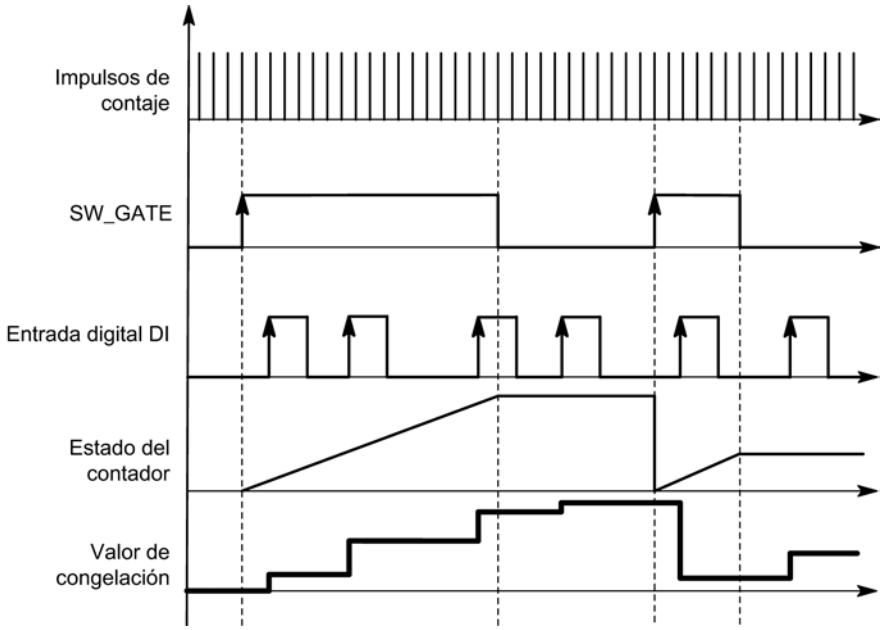
El estado del contador y el valor de congelación reciben sus estados de RESET.

- "0" en el sentido principal de contaje "Ninguno" o "Adelante"
- "Límite de contaje superior parametrizado" en el sentido principal de contaje atrás.

La función de contaje se inicia cuando se abre la puerta SW. El 1Count5V comienza a contar a partir del valor de carga.

- "0" en el sentido principal de contaje "Ninguno" o "Adelante"
- "Límite de contaje superior parametrizado" en el sentido principal de contaje atrás.

La siguiente figura muestra la función de congelar con el valor de carga = 0.



El valor de congelación equivale siempre al valor de contaje en el instante del flanco ascendente de la entrada digital DI.

En la interfaz de realimentación (Página 9700) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el nivel de la señal de congelación.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

En el modo isócrono, se muestra en la interfaz de realimentación el estado del contador que se congeló en el instante del último flanco ascendente antes de T_i.

Si se cierra la puerta SW, actúa como parametrizada, de cancelación o de interrupción. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

Interfaz de realimentación ampliada

Si el módulo 1Count5V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de realimentación (Página 9700).

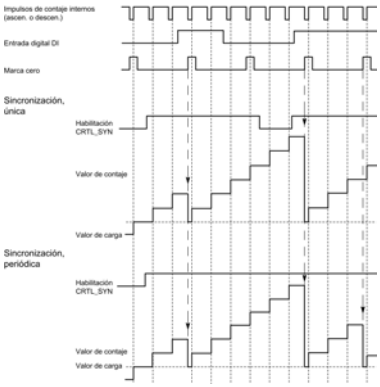
Habilitar HW para sincronización

Sincronización con DI y señal de paso por cero

Si se ha parametrizado la habilitación HW para sincronización, el DI sirve como habilitación HW. Cuando está activada la habilitación HW, se carga el valor de carga en el 1Count5V con la señal cero del encóder.

Se puede elegir entre la sincronización única y la periódica (parámetro "Sincronización (Página 9697)").

La figura siguiente muestra los estados del contador con una sincronización única y periódica.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control (Página 9698) "Habilitación de la sincronización" (CTRL_SYN) debe estar activado.
- Sincronización única: si el bit de habilitación está activado y la habilitación HW está presente, la primera señal cero carga el 1Count5V con el valor de carga.
- Sincronización periódica: Si el bit de habilitación está activado y la habilitación HW está presente, la primera señal de paso por cero y cada señal de paso por cero subsiguiente cargan el 1Count5V con el valor de carga.
- Tras una sincronización correcta, el bit de realimentación (Página 9700)STS_SYN se activa y el LED SYN se enciende. El bit de control (Página 9698)RES_STS desactiva el bit de realimentación y apaga el LED.

- La señal de un interruptor libre de rebotes puede servir de señal de referencia para la habilitación HW.
- El bit de realimentación (Página 9700)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación (Página 9700) activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9700)STS_DI indica el nivel de la entrada digital.

Seleccionar sincronización

Sincronización

Si se ha seleccionado "Sincronización con flanco ascendente" en "Función DI (Página 9688)", es posible elegir entre la sincronización única y la periódica.

- Una vez: si el bit de habilitación está activado, el primer flanco carga el 1Count5V con el valor de carga.
- Periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count5V con el valor de carga.

Determinar el límite superior de contaje

Descripción

Si se parametriza un límite superior de contaje en combinación con un sentido principal de contaje (Página 9471) (adelante o atrás) es posible limitar el rango de contaje máximo a un rango más pequeño. El rango de contaje parametrizado está comprendido entonces entre 0 y el límite superior de contaje ajustado.

El límite superior de contaje puede adoptar un valor > 1 hasta el límite superior del rango de contaje.

Interfaz de control y realimentación para los modos de contaje

Interfaz de control de los modos de contaje

Nota

Para el 1Count5V, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del 1Count5V para los modos de contaje.

Dirección		Asignación			
Bytes 0 a 3		**Valor de carga directo, preliminar, valor de comparación 1 ó 2**			
	Byte 0	**Comportamiento de DO1, DO2 del 1Count5V**			
		Bit 2	Bit 1	Bit 0	Función DO1
		0	0	0	Salida
		0	0	1	Conexión con estado del contador ≥ valor de comparación
		0	1	0	Conexión con estado del contador ≤ valor de comparación
		0	1	1	Impulso al alcanzarse el valor de comparación
		1	0	0	Conmutación en los valores de comparación
		1	0	1	bloqueado
		1	1	0	bloqueado
		1	1	1	bloqueado
			Bit 5	Bit 4	Función DO2
			0	0	Salida
			0	1	Conexión con estado del contador ≥ valor de comparación
			1	0	Conexión con estado del contador ≤ valor de comparación
			1	1	Impulso al alcanzarse el valor de comparación
		Bits 3, 6 y 7: Reservado = 0			
	Bytes 1 a 3	Byte 1	Histéresis DO1, DO2 (rango 0 … 255)		
		Byte 2	Duración del impulso DO1, DO2 (rango 0 a 255)		
		Byte 3	reservado = 0		
					Denominación
Byte 4		Bit 7	Acuse del error		EXTF_ACK
		Bit 6	Habilitación de DO2		CTRL_DO2
		Bit 5	Bit de control DO2		SET_DO2
		Bit 4	Habilitación de DO1		CTRL_DO1

Dirección	Asignación		
	Bit 3	Bit de control DO1	SET_DO1
	Bit 2	Inicio de la desactivación del bit de estado	RES_STS
	Bit 1	Habilitación de la sincronización	CTRL_SYN
	Bit 0	Bit de control puerta SW	SW_GATE
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	Cambio de función y comportamiento de DO1, DO2	C_DOPARAM
	Bit 3	Carga del valor de comparación 2	CMP_VAL2
	Bit 2	Carga del valor de comparación 1	CMP_VAL1
	Bit 1	Carga preliminar del contador	LOAD_PREPARE
	Bit 0	Carga directa del contador	LOAD_VAL
Bytes 6 a 7	reservado = 0 [1]		

[1] Falta en la interfaz de datos de usuario avanzada

Significado de los bits de control

Bit	Significado
C_DOPARAM	Cambio de función y comportamiento de DO1, DO2 Los valores de los bytes 0 a 2 son aceptados como nueva función, histéresis, y duración de impulso de DO1, DO2. Esto puede conllevar el siguiente error: Las condiciones de conexión al alcanzarse los valores de comparación no se cumplen.
CMP_VAL1	Carga del valor de comparación 1 El valor de los bytes 0 a 3 se transfiere al valor de comparación 1 con el bit de control "Carga del valor de comparación 1" (CMP_VAL1).
CMP_VAL2	Carga del valor de comparación 2 El valor de los bytes 0 a 3 se transfiere al valor de comparación 2 con el bit de control "Carga del valor de comparación 2" (CMP_VAL2).
CTRL_DO1	Habilitación de DO1 Este bit habilita la salida DO1
CTRL_DO2	Habilitación de DO2 Este bit habilita la salida DO2
CTRL_SYN	Este bit habilita la sincronización
EXTF_ACK	Acuse del error Los bits de error deben ser acusados mediante el bit de control EXTF_ACK una vez eliminada la causa.
LOAD_PREPARE	Carga preliminar del contador El valor de los bytes 0 a 3 se acepta como valor de carga
LOAD_VAL	El valor de los bytes de 0 a 3 se carga directamente como nuevo valor de contaje
RES_STS	Inicio de la desactivación del bit de estado Los bits de estado se desactivan mediante el proceso de realimentación entre el bit RES_STS y el bit RES_STS_A.
SET_DO1	Bit de control DO1 Conectar y desconectar la salida digital DO1 cuando se ha activado CRTL_DO1.

Bit	Significado
SET_DO2	Bit de control DO2 Conectar y desconectar la salida digital DO2 cuando se ha activado CRTL_DO2.
SW_GATE	Bit de control puerta SW La puerta SW se abre/cierra mediante la interfaz de control con el bit SW_GATE.

Interfaz de notificación de los modos de contaje

Nota

Para el 1Count5V, los datos siguientes de la interfaz de notificación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de notificación

La siguiente tabla muestra la asignación de la interfaz de notificación (entradas) del 1Count5V para los modos de contaje.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor de contaje o valor de contaje almacenado con función de congelación en la entrada digital		
Byte 4	Bit 7	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 6	cortocircuito / rotura de hilo / sobretemperatura	ERR_DO1
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	Cortocircuito/rotura de hilo/sobretemperatura	ERR_DO2
	Bit 3	cortocircuito / rotura de hilo / señal del encóder	ERR_ENCODER
	Bit 2	Desactivación de los bits de estado en curso	RES_STS_A
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Byte 5	Bit 7	Estado de sentido descendente	STS_C_DN
	Bit 6	Estado de sentido ascendente	STS_C_UP
	Bit 5	reservado = 0	
	Bit 4	Estado de DO2	STS_DO2
	Bit 3	Estado de DO1	STS_DO1
	Bit 2	reservado = 0	
	Bit 1	Estado de DI	STS_DI
	Bit 0	Estado de la puerta interna	STS_GATE
Byte 6	Bit 7	Paso por cero en el rango de contaje sin sentido principal de contaje	STS_ND
	Bit 6	Límite inferior	STS_UFLW
	Bit 5	Límite superior	STS_OFLW

Dirección	Asignación		Denominación
	Bit 4	Estado del comparador 2	STS_CMP2
	Bit 3	Estado del comparador 1	STS_CMP1
	Bit 2	reservado = 0	
	Bit 1	reservado = 0	
	Bit 0	Estado de la sincronización	STS_SYN
Byte 7	reservado = 0		
Bytes 8 a 11	Valor de contaje [1]		

[1] Interfaz de datos de usuario avanzada

Nota sobre los bits de realimentación

Bit	Significado
ERR_24V	Cortocircuito en la alimentación del encóder El bit de error tiene que ser acusado mediante el bit de control (Página 9695)EXTF_ACK. Aviso de diagnóstico, si está parametrizado.
ERR_DO1	Cortocircuito / rotura de hilo / sobretemperatura por sobrecarga en la salida DO1 El bit de error tiene que ser acusado mediante el bit de control (Página 9695)EXTF_ACK. Aviso de diagnóstico, si está parametrizado.
ERR_DO2	Cortocircuito/rotura de hilo/sobretemperatura en la salida DO2 El bit de error tiene que ser acusado mediante el bit de control (Página 9695)EXTF_ACK. Aviso de diagnóstico, si está parametrizado.
ERR_ENCODER	Cortocircuito/rotura de hilo señal del encóder 5 V El bit de error tiene que ser acusado mediante el bit de control (Página 9695)EXTF_ACK. Aviso de diagnóstico, si está parametrizado.
ERR_LOAD	Error en función de carga Los bits de control (Página 9695)LOAD_VAL, LOAD_PREPARE, CMP_VAL1, CMP_VAL2 y C_DOPARAM no pueden estar activados al mismo tiempo durante la transferencia. Esto tiene como consecuencia la activación del bit de estado ERR_LOAD, similar a cargar un valor incorrecto (que no se acepta).
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
RES_STS_A	Desactivación de los bits de estado en curso
STS_C_DN	Estado de sentido descendente
STS_C_UP	Estado de sentido ascendente
STS_CMP1	Estado del comparador 1 El bit de estado STS_CMP1 indica que la salida está o estaba activada. Debe acusarse con el bit de control (Página 9695)RES_STS. Si el bit de estado se acusa cuando la salida todavía está activada, el bit se vuelve a activar inmediatamente. Este bit también se activa cuando se activa el bit de control SET_DO1 sin estar habilitada DO1.
STS_CMP2	Estado del comparador 2 El bit de estado STS_CMP2 indica que la salida está o estaba activada. Debe acusarse con el bit de control (Página 9695)RES_STS. Si el bit de estado se acusa cuando la salida todavía está activada, el bit se vuelve a activar inmediatamente. Este bit también se activa cuando se acciona el bit de control SET_DO2 sin que esté habilitada DO2.

Bit	Significado
STS_DI	Estado de DI El estado de la DI se indica en todos los modos de operación con el bit STS_DI en la interfaz de notificación.
STS_DO1	Estado de DO1 El bit de estado STS_DO1 muestra el estado de la salida digital DO1.
STS_DO2	Estado de DO2 El bit de estado STS_DO2 muestra el estado de la salida digital DO2.
STS_GATE	Estado de la puerta interna: Contaje en curso.
STS_LOAD	Función de carga en curso
STS_ND	Paso por cero en el rango de contaje sin sentido principal de contaje. El bit debe desactivarse con el bit de control (Página 9695)RES_STS.
STS_OFLW STS_UFLW	Límite superior de contaje excedido Límite inferior de contaje excedido Ambos bits deben ser desactivados.
STS_SYN	Estado de la sincronización Al terminar la sincronización se activa el bit STS_SYN. Debe desactivarse con el bit de control (Página 9695)RES_STS.

Parámetros para medición

Activar diagnóstico colectivo

Diagnóstico colectivo

Si activa el diagnóstico colectivo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar el comportamiento en caso de STOP de la CPU/del maestro

Ajuste del comportamiento en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count5V en caso de fallo del controlador principal.

Parámetro	Estado del 1Count5V en caso de STOP de la CPU/del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual continúa. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.
DO Aplicar valor sustitutivo	El modo de operación actual se cancela. La puerta se cierra y el valor sustitutivo parametrizado para la salida digital se aplica directamente. Los valores de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros. Si para la salida se parametriza "Impulso al alcanzar el valor de comparación", sólo estará disponible el valor sustitutivo = 1 para la duración del impulso.	Los parámetros modificados son aceptados y tienen efecto inmediato.
DO Mantener el último valor	El modo de operación actual se cancela. La puerta se cierra y el estado de la salida digital se conserva. Los valores de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count5V.

Abandonar el estado parametrizado

¿En qué condiciones abandona el 1Count5V el estado parametrizado?

La CPU o el maestro debe estar en modo RUN y la interfaz de control (Página 9714) debe modificarse.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/del maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count5V
- POWER ON o inserción del módulo de potencia correspondiente

Método de medición

Resumen

Método de medición

En el parámetro "Método de medición" es posible elegir entre los siguientes métodos:

- con tiempo de integración (Página 9704)
- continuo (Página 9705)

Medición con tiempo de integración

Desarrollo de la medición con tiempo de integración

La medición se lleva a cabo durante el tiempo de integración parametrizado. Al transcurrir el tiempo de integración se actualiza el valor medido.

El final de una medición se indica con el bit de estado (Página 9716)STS_CMP1. Este bit se desactiva con el bit de control (Página 9714)RES_STS en la interfaz de control.

Si no se han producido al menos dos flancos ascendentes dentro del tiempo de integración parametrizado, se devuelve 0 como valor medido.

Se devuelve el valor -1 s hasta finalizar el primer tiempo de integración.

El tiempo de integración para la siguiente medición se puede cambiar durante el funcionamiento.

Inversión del sentido de giro

Si el sentido de rotación se invierte durante un tiempo de integración, el valor medido para este período es incierto. Al evaluar los bits de realimentación (Página 9716)STS_C_UP y STS_C_DN (evaluación de sentido), se puede responder a una posible irregularidad del proceso.

Consulte también

Tiempo de integración y tiempo de actualización en el modo isócrono (Página 9708)

Medición continua

Principio de medición

El módulo 1Count5V cuenta todos los flancos ascendentes de un impulso y les asigna un valor de tiempo en µs.

El tiempo de actualización indica en qué espacio de tiempo se actualiza el valor medido del módulo en la interfaz de notificación.

En una secuencia de impulsos con uno o varios impulsos por tiempo de actualización rige lo siguiente:

Tiempo de medición dinámico =	valor de tiempo del último impulso en el intervalo actual del tiempo de actualización menos valor de tiempo del último impulso en el intervalo anterior del tiempo de actualización

Una vez transcurrido el tiempo de actualización, se calcula y aplica un nuevo valor medido con el tiempo de medición dinámico.

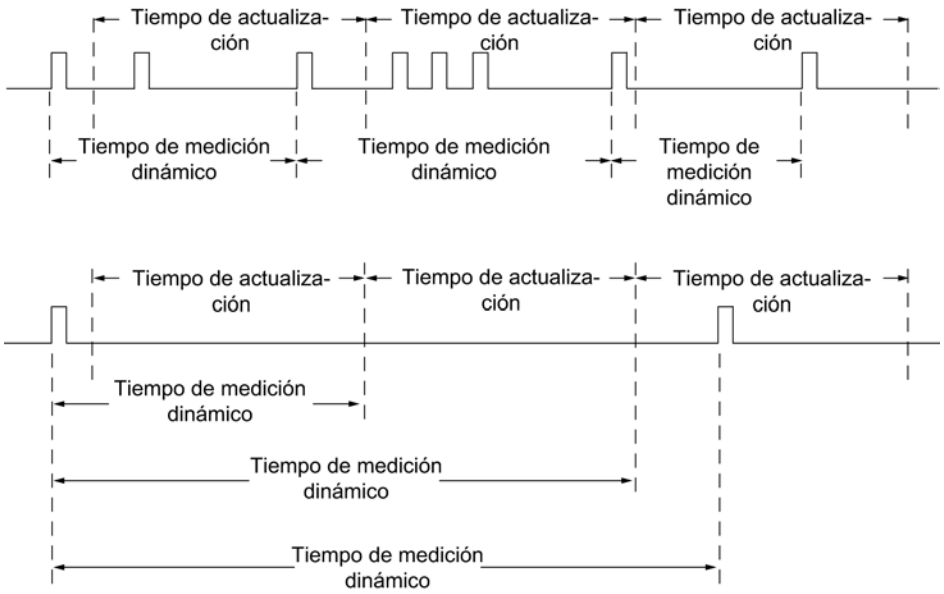
Si el tiempo de actualización actual no contiene ningún impulso, el tiempo de medición dinámico resultante es el siguiente:

Tiempo de medición dinámico =	valor de tiempo del tiempo de actualización actual transcurrido menos el valor de tiempo del último impulso

Una vez transcurrido un tiempo de actualización, se calcula un valor medido aproximado con el tiempo de medición dinámico suponiendo que haya aparecido un impulso al final del tiempo de actualización.

Si en la medición de frecuencia y de velocidad el valor medido aproximado "1 impulso por tiempo de medición dinámico" es menor que el último valor medido, este valor medido aproximado se emite como nuevo valor medido. En la medición del período, el tiempo de medición dinámico se emite como duración aproximada del período cuando el tiempo de medición dinámico es mayor que la última duración medida del período.

La figura muestra el principio de medición.



El 1Count5V mide de forma continua. Durante la parametrización se introduce un tiempo de actualización.

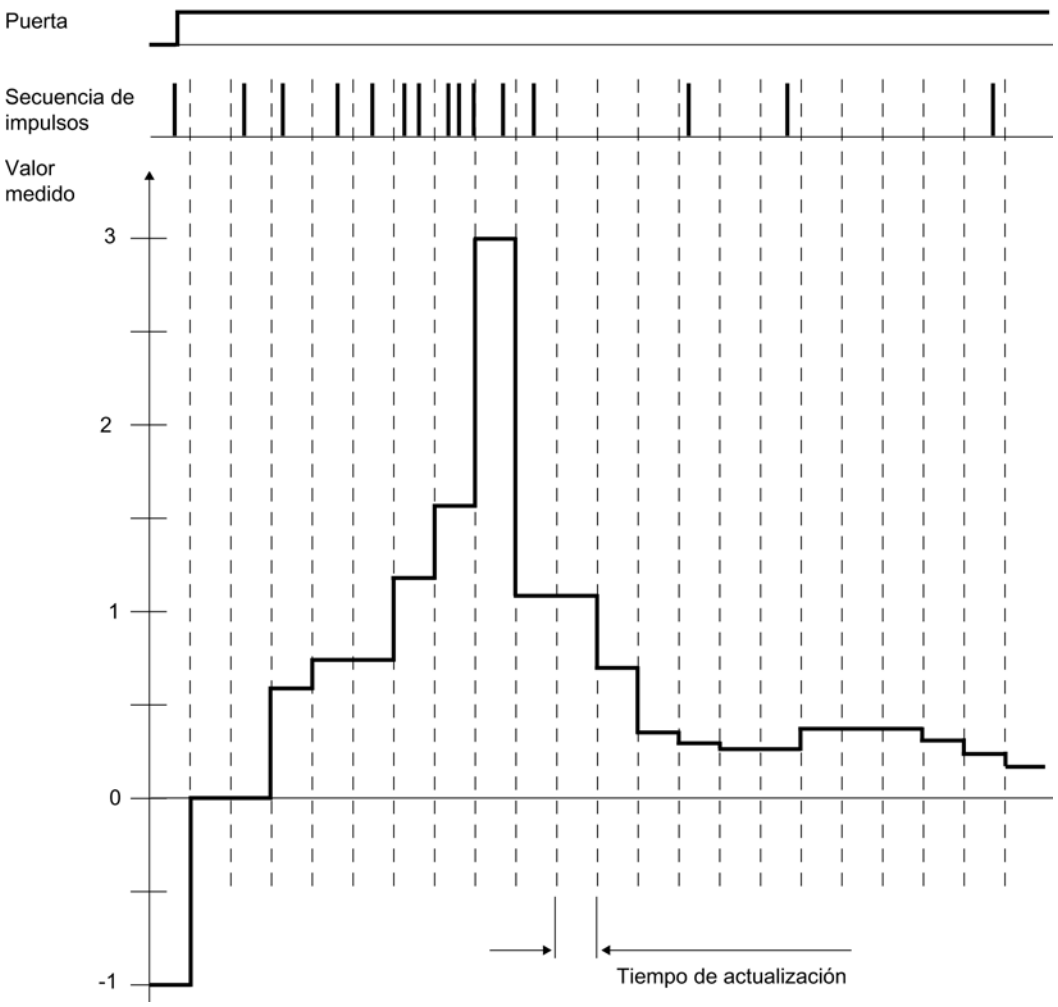
El valor "–1" se devuelve antes de finalizar el primer tiempo de actualización transcurrido.

La medición continua empieza tras abrir la puerta con el primer impulso de la secuencia de impulsos que debe medirse. El primer valor medido no puede calcularse hasta después del segundo impulso.

Cada vez que transcurre el tiempo de actualización, en la interfaz de notificación se emite un valor medido (frecuencia, duración del período o velocidad). El final de una medición se indica con el bit de estado (Página 9716)STS_CMP1. Este bit se desactiva con los bits RES_STS y RES_STS_A según el principio de realimentación completo.

Si el sentido de rotación se invierte durante un tiempo de actualización, el valor medido para este período es incierto. Evaluando los bits de realimentación (Página 9716)STS_C_DN y STS_C_UP (evaluación del sentido) se puede reaccionar a posibles irregularidades del proceso.

La figura siguiente explica el principio de la medición continua con el ejemplo de una medición de frecuencia.



Control de puerta

Para controlar el módulo 1Count5V se deben usar las funciones de puerta.

Modo isócrono

En el modo isócrono, el 1Count5V acepta en cada ciclo de bus bits de control y valores de control de la interfaz de control y notifica la respuesta durante el mismo ciclo.

El 1Count5V transfiere en cada ciclo un valor medido y los bits de estado tal y como eran válidos en el instante T_i.

La medición empieza y acaba en cada caso en el instante T_i.

Consulte también

Tiempo de integración y tiempo de actualización en el modo isócrono (Página 9708)

Tiempo de integración y tiempo de actualización en el modo isócrono

Tiempo de integración y tiempo de actualización en el modo isócrono

Si el tiempo de integración/actualización dura varios ciclos T_{DP}, es posible reconocer el nuevo valor medido en el programa de usuario por el bit de estado (Página 9716)STS_CMP1 (medición finalizada) de la interfaz de notificación. Esto permite una vigilancia del proceso de medición o una sincronización con el proceso de medición. Sin embargo, el acuse de este aviso dura 4 ciclos T_{DP}. En este caso el tiempo de integración/actualización mínimo será de (4 x T_{DP}).

Si la aplicación puede tolerar un jitter del tiempo de integración/actualización de un T_{DP} y un valor medido constante durante varios ciclos, no es necesario evaluar constantemente el bit de estado STS_CMP1. Entonces serán posibles tiempos de integración/actualización de (1 x T_{DP}) a (3 x T_{DP}).

Debido a una pérdida del sincronismo en el último ciclo T_{DP} del tiempo de integración se prolonga el tiempo de integración/actualización otro ciclo T_{DP}. El valor medido no se ve falsificado por ello.

Nota

No deben excederse los límites del rango del tiempo de integración/actualización (consulte las tablas de cada uno de los modos de medición).

Una violación de los límites del rango de valores provoca un error de parametrización, con lo cual el 1Count5V no cambia al modo isócrono.

Nota

Al cambiar la configuración del modo no isócrono al modo isócrono y viceversa, siempre deberá adaptarse el parámetro Tiempo de integración/actualización si se desea mantener la duración del tiempo de integración/actualización.

Activar diagnóstico de los pares de señales

Vigilancia de los pares de señales

Los pares de señales A y /A, y B y /B pueden vigilarse.

Active para ello la casilla de verificación correspondiente.

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Comportamiento de las salidas en los modos de medición

Seleccionar comportamiento de la salida DO1

Comportamiento de la salida en los modos de medición

La salida digital del 1Count5V es parametrizable.

Para la medición de frecuencia, velocidad o período se puede indicar un límite superior y un límite inferior, respectivamente, que al ser excedido active la salida digital DO1. Estos valores límite son parametrizables y pueden ser modificados con la función de carga.

La función y el comportamiento de la salida digital se puede modificar durante el funcionamiento. La nueva función actúa de inmediato, en el modo isócrono siempre en el instante T_i.

En la lista desplegable se pueden seleccionar las siguientes funciones.

- Salida (Página 9709)
- Fuera de los valores límite (Página 9710) (vigilancia de valor límite)
- Rebase por defecto del valor límite inferior (Página 9710) (vigilancia de valor límite)
- Rebase por exceso del valor límite superior (Página 9710) (vigilancia de valor límite)

Salida

Salida

Para conectar o desconectar las salidas, es necesario habilitarlas con los bits de control (Página 9714)CTRL_DO1 o CTRL_DO2.

Los bits de control (Página 9714)SET_DO1 o SET_DO2 permiten activar y desactivar las salidas.

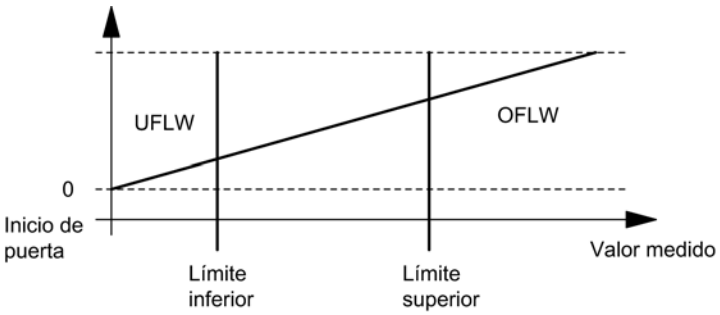
Con el bit de estado (Página 9716)STS_DO1 o STS_DO2 de la interfaz de notificación se puede consultar el estado de la salida.

En el modo isócrono las salidas conmutan en el instante T_o.

Vigilancia de valores límite

Vigilancia de valor límite

La figura siguiente muestra el principio de la vigilancia de valor límite.



Una vez transcurrido el tiempo de integración, el valor medido (frecuencia, velocidad o período) se compara con los valores límite parametrizados.

Si el valor medido actual está por debajo del valor límite inferior parametrizado (valor medido < valor límite inferior), el bit STS_UFLW = 1 se activa en la interfaz de notificación (Página 9716).

Si el valor medido actual está por encima del valor límite superior parametrizado (valor medido < valor límite superior), el bit STS_OFLW = 1 se activa en la interfaz de realimentación (Página 9716).

Estos bits tienen que ser acusados con el bit de control (Página 9714)RES_STS.

Si el valor medido sigue fuera de los valores límite o vuelve a estarlo después del realimentación, se vuelve a activar el correspondiente bit de estado.

Poniendo el límite inferior a 0, se desconecta la vigilancia dinámica de desbordamiento por defecto del valor límite inferior.

La tabla siguiente muestra que, dependiendo de la parametrización, la salida digital DO1 habilitada se puede activar con la vigilancia de valor límite:

Parámetro "Función de DO1"	DO1 se activa cuando ...
Fuera de los valores límite	Valor medido < valor límite inferior O Valor medido > valor límite superior
Rebase por defecto del valor límite inferior	Valor medido < valor límite inferior
Rebase por exceso del valor límite superior	Valor medido > valor límite superior

En el modo isócrono la salida conmuta al terminar la medición en el instante T_i.

Seleccionar valor sustitutivo de DOx

Valor sustitutivo DO1 y valor sustitutivo DO2

Si se ha seleccionado el parámetro de conmutar valor sustitutivo DO en "Comportamiento en caso de STOP de la CPU/del maestro (Página 9700)", seleccione el valor sustitutivo que debe enviarse a la salida digital DO1 o DO2.

Activar diagnóstico DOx

Diagnóstico DO1 y diagnóstico DO2

El "diagnóstico DO1" o "diagnóstico DO2" (rotura de hilo, cortocircuito) sólo es posible con anchos de impulso > 90 ms en la salida digital DO1 o DO2.

Active la respectiva casilla de verificación para el diagnóstico de la salida DO1 o DO2.

Seleccionar modo de medición

Descripción

Al preseleccionar un modo de medición se elige la funcionalidad con la que se utilizará el 1Count5V.

	Descripción	
Modo de operación	Medición con tiempo de integración	Medición continua
Medición de la frecuencia (Página 9478) (con puerta HW o SW)	El 1Count5V cuenta los impulsos que llegan en un tiempo de integración predefinido.	El 1Count5V cuenta los impulsos que llegan en un tiempo de medición dinámico.
Medición de la velocidad (Página 9480) (con puerta HW o SW)	El 1Count5V cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de integración predefinido y calcula a partir de ello la velocidad con los impulsos por vuelta del encóder.	El 1Count5V cuenta los impulsos que llegan de un encóder de velocidad en un tiempo de medición dinámico y calcula a partir de ello la velocidad con los impulsos por vuelta del encóder.
Medición del período (Página 9481) (con puerta HW o SW)	El 1Count5V mide el tiempo entre dos flancos ascendentes de la señal de contaje contando los impulsos de una frecuencia interna de referencia (16 MHz) de precisión de cuarzo dentro del tiempo de integración dado.	El 1Count5V indica el tiempo de medición dinámico como duración del período. Si el período es inferior al tiempo de actualización, se crea un valor medio para el período.

El ajuste predeterminado es el modo de operación "Medición de la frecuencia".

Seleccionar la resolución del período

Descripción

El valor del período medido se indica en 1 µs o 1/16 µs.

Seleccione la unidad deseada para la resolución.

Rangos de medición posibles con información de errores

- Medición del período con tiempo de integración
  - Rangos de medición y errores a una resolución de 1 µs:

Tiempo de integración	$T_{mín}$ ± error absoluto	$T_{mín}$ ± error absoluto
100 s	1 µs* (10 ± 0)	1 µs* (100.000.000 ± 10.000)
10 s	1 µs* (10 ± 0)	1 µs* (10.000.000 ± 1.000)
1 s	1 µs* (10 ± 0)	1 µs* (1.000.000 ± 100)
0,1 s	1 µs* (10 ± 0)	1 µs* (100.000 ± 10)
0,01 s	1 µs* (10 ± 0)	1 µs* (10.000 ± 1)

  - Rangos de medición y errores en una resolución de 1/16 µs:

Tiempo de integración	$T_{mín}$ ± error absoluto	$T_{mín}$ ± error absoluto
100 s	1/16 µs* (160 ± 0)	1/16 µs* (1.600.000.000 ± 160.000)
10 s	1/16 µs* (160 ± 0)	1/16 µs* (160.000.000 ± 16.000)
1 s	1/16 µs* (160 ± 0)	1/16 µs* (16.000.000 ± 1.600)
0,1 s	1/16 µs* (160 ± 0)	1/16 µs* (1.600.000 ± 160)
0,01 s	1/16 µs* (160 ± 0)	1/16 µs* (160.000 ± 16)

- Medición continua del período:
  - Rangos de medición y errores a una resolución de 1 µs:

Período $T_{mín}$ ± error absoluto	Período $T_{mín}$ ± error absoluto
1 µs* (10 ± 0)	1 µs* (100 000 ± 10)
1 µs* (100 ± 0)	1 µs* (1 000 000 ± 100)
1 µs* (1 000 ± 0)	1 µs* (10 000 000 ± 1 002)
1 µs* (10 000 ± 1)	1 µs* (100 000 000 ± 10 020)

  - Rangos de medición y errores en una resolución de 1/16 µs:

Período $T_{mín}$ ± error absoluto	Período $T_{mín}$ ± error absoluto
1/16 µs* (160 ± 1)	1/16 µs* (1 600 000 ± 160)
1/16 µs* (1 600 ± 1)	1/16 µs* (16 000 000 ± 1 600)
1/16 µs* (16 000 ± 3)	1/16 µs* (160 000 000 ± 16 000)
1/16 µs* (160 000 ± 20)	1/16 µs* (1 600 000 000 ± 160 000)

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital.

El ajuste predeterminado es "Entrada".

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9716)STS_DI indica el nivel de la entrada digital.

Definir valor límite inferior y superior

Valores límite inferior y superior

Para la medición de frecuencia, velocidad o período se puede indicar un límite superior y un límite inferior, respectivamente, que al ser excedido active la salida digital DO1 (Página 9706). Estos valores límite se pueden parametrizar y modificar a través de la función de carga.

Los siguientes rangos de valores están permitidos para los valores límite inferior y superior:

Modo de medición	Valor límite inferior (LI):	Valor límite superior
Medición de frecuencia	0 a 99.999.999 × 10^{-3} Hz	LI+1 a 100.000.000 × 10^{-3} Hz
Medición de velocidad	0 a 24.999.999 × 10^{-3} /min	LI+1 a 25.000.000 × 10^{-3}/min
Medición del período (resolución 1 µs)	0 a 119.999.999 µs	LI+1 a 120.000.000 µs
Medición del período (resolución 1/16 µs)	0 a 1.919.999.999 µs	LI+1 a 1.920.000.000 µs

Definir el tiempo de medición

Descripción

Si se ha seleccionado un modo de medición, el módulo actualiza los valores medidos de forma cíclica. El tiempo de actualización/integración es el producto del tiempo de medición x 10 ms. El tiempo de medición se predefine como factor n.

Si desea obtener una descripción detallada sobre el rango de valores para el tiempo de medición, consulte el apartado "Conceptos básicos sobre la operación de medición (Página 9474)".

Introducir impulsos por vuelta del encóder

Impulsos por vuelta del encóder

Para el modo Medición de velocidad (Página 9480) es preciso parametrizar adicionalmente los impulsos por vuelta del encóder.

El valor para el número de impulsos por vuelta del encóder se encuentra en la placa de características o en los datos técnicos del encóder.

Interfaz de notificación y control para los modos de medición

Interfaz de control de los modos de medición

Nota

Para el 1Count5V, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del 1Count5V para los modos de medición.

Dirección	Asignación			
Bytes 0 a 3	**Valor límite inferior o superior**			
	Función de DO1			
	Byte 0	Bit 1	Bit 0	Función DO1
		0	0	Salida
		0	1	Fuera de los límites
		1	0	Rebase por defecto del valor límite inferior
		1	1	Rebase por exceso del valor límite superior
	Bytes 1 a 3	reservado = 0		
	Tiempo de integración/actualización			
	Byte 0, 1	Tiempo de integración [n × 10 ms] (Rango de valores 1 … 1.000 / 12.000)		
	Byte 2, 3	reservado = 0		

Dirección	Asignación		
			Denominación
Byte 4	Bit 7	Acuse del error	EXTF_ACK
	Bit 6	Habilitación de DO2	CTRL_DO2
	Bit 5	Bit de control DO2	SET_DO2
	Bit 4	Habilitación de DO1	CTRL_DO1
	Bit 3	Bit de control DO1	SET_DO1
	Bit 2	Inicio de la desactivación del bit de estado	RES_STS
	Bit 1	reservado = 0	
	Bit 0	Bit de control puerta SW	SW_GATE
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	Modificar la función de DO1	C_DOPARAM
	Bit 3	reservado = 0	
	Bit 2	Modificar el tiempo de integración	C_INTTIME
	Bit 1	Cargar valor límite superior	LOAD_PREPARE
	Bit 0	Cargar valor límite inferior	LOAD_VAL
Bytes 6 a 7	reservado = 0 [1]		

[1] Falta en la interfaz de datos de usuario avanzada

Significado de los bits de control

Bit	Significado
C_DOPARAM	Modificar la función de DO1 El valor del byte 0 es aceptado como nueva función de DO1.
C_INTTIME	Modificar el tiempo de integración El valor de los bytes 0 y 1 es aceptado como nuevo tiempo de integración para la siguiente medición.
CTRL_DO1	Habilitación de DO1 Este bit habilita la salida DO1
CTRL_DO2	Habilitación de DO2 Este bit habilita la salida DO2
EXTF_ACK	Acuse del error Los bits de error deben ser acusados mediante el bit de control EXTF_ACK una vez eliminada la causa.
LOAD_PREPARE	Cargar valor límite superior El valor de los bytes 0 a 3 es aceptado como nuevo límite superior.
LOAD_VAL	Cargar valor límite inferior El valor de los bytes 0 a 3 es aceptado como nuevo valor límite inferior.
RES_STS	Inicio de la desactivación del bit de estado Los bits de estado se desactivan mediante el proceso de realimentación entre el bit RES_STS y el bit RES_STS_A.
SET_DO1	Bit de control DO1 Conectar y desconectar la salida digital DO1 cuando se ha activado CRTL_DO1.

Bit	Significado
SET_DO2	Bit de control DO2 Conectar y desconectar la salida digital DO2 cuando se ha activado CRTL_DO2.
SW_GATE	Bit de control puerta SW La puerta SW se abre/cierra mediante la interfaz de control con el bit SW_GATE.

Interfaz de notificación de los modos de medición

Nota

Para el 1Count5V, los datos siguientes de la interfaz de notificación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de notificación

La siguiente tabla muestra la asignación de la interfaz de notificación (entradas) del 1Count5V para los modos de medición.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor medido		
Byte 4	Bit 7	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 6	Cortocircuito/rotura de hilo/sobretemperatura	ERR_DO1
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	Cortocircuito/rotura de hilo/sobretemperatura	ERR_DO2
	Bit 3	Cortocircuito/rotura de hilo/señal del encóder	ERR_ENCODER
	Bit 2	Desactivación de los bits de estado en curso	RES_STS_A
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Byte 5	Bit 7	Estado de sentido descendente	STS_C_DN
	Bit 6	Estado de sentido ascendente	STS_C_UP
	Bit 5	reservado = 0	
	Bit 4	Estado de DO2	STS_DO2
	Bit 3	Estado de DO1	STS_DO1
	Bit 2	reservado = 0	
	Bit 1	Estado de DI	STS_DI
	Bit 0	Estado de la puerta interna	STS_GATE
Byte 6	Bit 70	reservado = 0	
	Bit 6	Valor límite inferior del rango de medida	STS_UFLW
	Bit 5	Valor límite superior del rango de medida	STS_OFLW
	Bit 4	reservado = 0	

Dirección	Asignación		Denominación
	Bit 3	Fin de la medición	STS_CMP1
	Bit 2	reservado = 0	
	Bit 1	reservado = 0	
	Bit 0	reservado = 0	
Byte 7	reservado = 0		
Bytes 8 a 11	Valor de contaje [1]		

[1] Interfaz de datos de usuario avanzada

Nota sobre los bits de realimentación

Bit	Significado
ERR_24V	Cortocircuito en la alimentación del encóder El bit de error tiene que ser acusado con el bit de control (Página 9711)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_DO1	Cortocircuito/rotura de hilo/sobretemperatura en la salida DO1 El bit de error tiene que ser acusado con el bit de control (Página 9711)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_DO2	Cortocircuito/rotura de hilo/sobretemperatura en la salida DO2 El bit de error tiene que ser acusado con el bit de control (Página 9711)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_ENCODER	Cortocircuito / rotura de hilo señal del encóder El bit de error tiene que ser acusado con el bit de control (Página 9711)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_LOAD	Error en función de carga Los bits de control (Página 9711)LOAD_VAL, LOAD_PREPARE, C_DOPARAM y C_INTTIME no pueden estar activados al mismo tiempo durante la transferencia. Esto tiene como consecuencia la activación del bit de estado ERR_LOAD, similar a cargar un valor incorrecto (que no se acepta).
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
RES_STS_A	Desactivación de los bits de estado en curso
STS_C_DN	Estado de sentido descendente
STS_C_UP	Estado de sentido ascendente
STS_CMP1	Fin de la medición El valor medido es actualizado cada vez que termina un intervalo (tiempo de actualización/integración). **Medición con tiempo de integración:** El fin de una medición (al terminar el intervalo) es indicado con el bit de estado STS_CMP1. **Medición continua:** Al finalizar el tiempo de actualización, el fin de la medición se indica con el bit de estado STS_CMP1, siempre que se emita un valor medido. Si se emite un valor medido aproximado, el bit se mantiene en "0". Este bit se desactiva con el bit de control (Página 9711)RES_STS en la interfaz de control.
STS_DI	Estado de DI El estado de la DI se indica en todos los modos de operación con el bit STS_DI en la interfaz de notificación.
STS_DO1	Estado de DO1

Bit	Significado
STS_DO2	Estado de DO2
STS_GATE	Estado de la puerta interna: Se mide.
STS_LOAD	Función de carga en curso
STS_OFLW STS_UFLW	Valor límite superior rebasado Valor límite inferior no alcanzado Ambos bits deben ser desactivados.

Parámetros para la lectura del recorrido

Activar diagnóstico colectivo

Diagnóstico colectivo

Si activa el diagnóstico colectivo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar la reacción en caso de STOP de la CPU/del maestro

Ajuste de la reacción en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count5V en caso de fallo del controlador principal.

Parámetros	Estado del 1Count5V en caso de STOP de la CPU/ del maestro	¿Qué ocurre en el caso de una reparametrización?
Desconectar DO	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El valor límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual continúa. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/ el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count5V.

Abandonar el estado parametrizado

¿En qué condiciones abandona el 1Count5V el estado parametrizado?

La CPU o el maestro debe estar en modo RUN y la interfaz de control (Página 9726) debe modificarse.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count5V
- POWER ON o inserción del módulo de potencia correspondiente

Seleccionar evaluación de señal

Evaluación de señal A, B

La evaluación de la señal mediante A, B permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Encóder rotativo (Página 9494)
  El 1Count5V puede contar los flancos de las señales. Normalmente, sólo se evalúan los flancos en A (evaluación simple). Para obtener una resolución más alta, en la parametrización (parámetro "Evaluación de señal") se puede seleccionar si las señales deben evaluarse de forma simple (Página 9495), doble (Página 9495) o cuádruple (Página 9495).
  La evaluación múltiple sólo es posible con encóders incrementales asimétricos con señales A y B desfasadas 90 grados.

  Nota

  La indicación 500 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A o B. Así, en la evaluación doble se obtienen como máximo 1 MHz en los impulsos de contaje o máximo 2 MHz en la evaluación cuádruple.

Activar diagnóstico de los pares de señales

Vigilancia de los pares de señales

Los pares de señales A y /A, B y /B, N y /N pueden vigilarse.

Active para ello la casilla de verificación correspondiente.

Nota

El "Diagnóstico N, /N" no se puede activar hasta que se ha activado el "diagnóstico A, /A y B, /B".

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Función DI

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital
- Congelar/redisparar con flanco ascendente (Página 9721)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador. Después de la operación de guardado, el contador se establece al valor de carga.
- Sincronización con flanco ascendente (Página 9722)
  Los flancos ascendentes de la entrada digital permiten establecer el contador al valor de carga.
- Congelar con flanco ascendente (Página 9724)
  Los flancos ascendentes de la entrada digital permiten almacenar (congelar) estados del contador.
- Habilitar HW para sincronización (Página 9725)
  La entrada digital sirve como habilitación HW. Cuando está activada la habilitación HW, el contador se activa en el valor de carga con la señal de paso por cero del encóder.

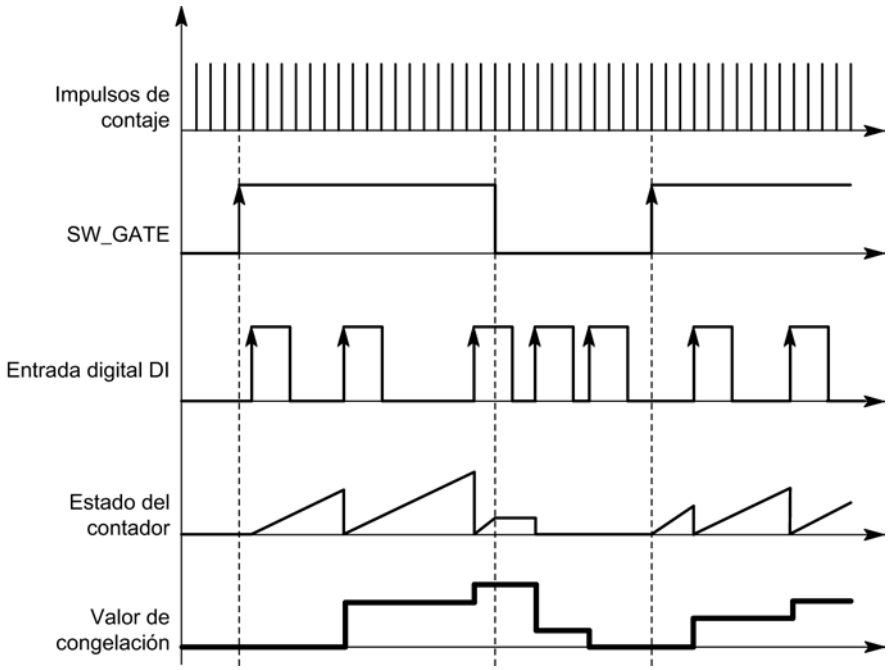
El ajuste predeterminado es "Entrada".

Congelar/redisparar con flanco ascendente

Congelar/redisparar

Con esta función se almacena el estado actual interno del contador del 1Count5V y se redispara el contaje cuando hay un flanco ascendente en la entrada digital. Esto significa que se almacena el estado actual interno del contador en el momento del flanco ascendente (valor de congelación) y, entonces, se carga nuevamente el módulo 1Count5V con el valor de carga y se sigue contando desde allí.

La siguiente figura muestra la función de congelar/redisparar con valor de carga = 0.



Para poder ejecutar la función, el modo de contaje tiene que estar habilitado con la puerta SW. Se inicia con el primer flanco ascendente de la entrada digital.

En la interfaz de notificación (Página 9728) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el estado de la señal de congelación y redisparo.

El valor de congelación se pone por defecto al estado de RESET ("0"). No cambia cuando se abre la puerta SW.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

Si se cierra la puerta SW, sólo se interrumpe el contaje; es decir, cuando se vuelve a abrir la puerta SW, continúa el contaje. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

También en el modo isócrono se congela y redispara el contaje con cada flanco de la entrada digital. En la interfaz de realimentación se muestra el estado del contador válido en el instante del último flanco antes de T_i.

Interfaz de notificación ampliada

Si el módulo 1Count5V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de notificación (Página 9728).

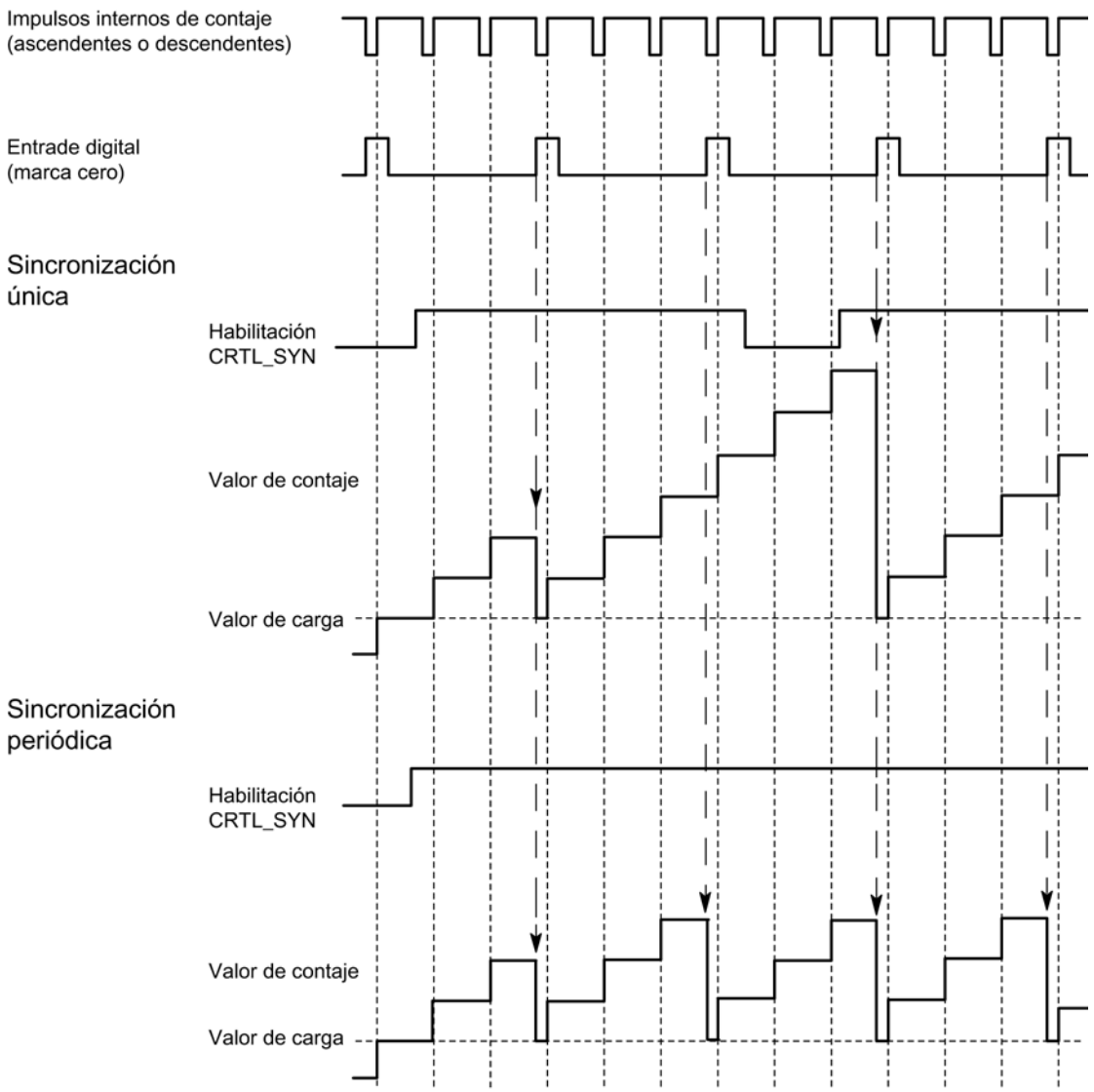
Sincronización con flanco ascendente

Sincronización

Si se ha parametrizado "Sincronización con flanco ascendente", el flanco ascendente de una señal de referencia en la entrada sirve para poner el 1Count5V al valor de carga.

Se puede seleccionar entre la sincronización única y la periódica (parámetro "Sincronización").

La figura siguiente muestra los estados del contador con una sincronización única y periódica.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control (Página 9726) "Habilitación de la sincronización" (CTRL_SYN) debe estar activado.
- Sincronización única: si el bit de habilitación está activado, el primer flanco carga el 1Count5V con el valor de carga.
- Sincronización periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count5V con el valor de carga.
- Tras una sincronización correcta, el bit de realimentación (Página 9728)STS_SYN se activa y el LED SYN se enciende. El bit de control (Página 9726)RES_STS desactiva el bit de realimentación y apaga el LED.
- La señal de un interruptor libre de rebotes puede servir de señal de referencia.
- El bit de realimentación (Página 9728)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación (Página 9728) activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

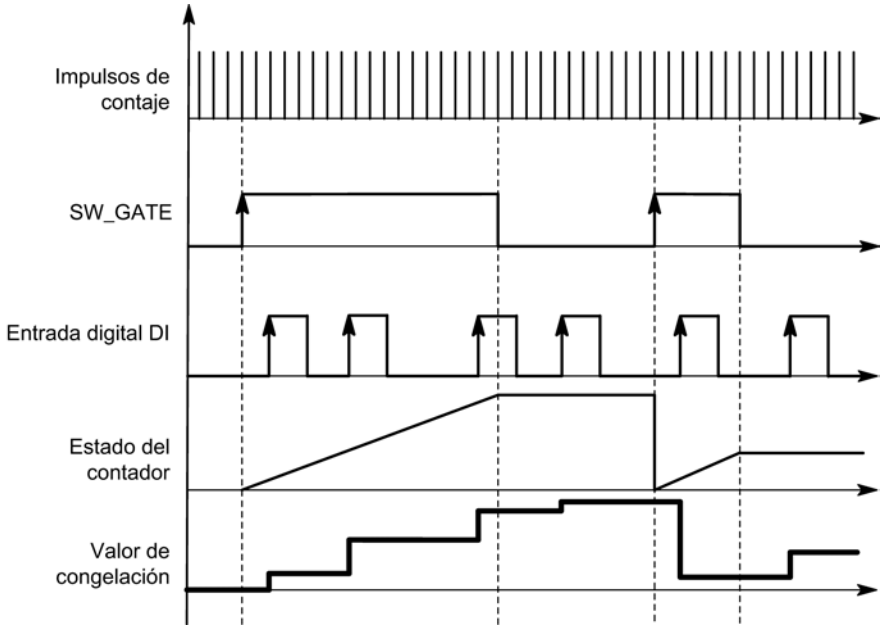
Congelar con flanco ascendente

Congelar

El estado del contador y el valor de congelación reciben sus estados de RESET ("0").

La función de contaje se inicia cuando se abre la puerta SW. El 1Count5V comienza a contar a partir del valor de carga ("0").

La siguiente figura muestra la función de congelar con valor de carga = 0.



El valor de congelación equivale siempre al valor de contaje en el instante del flanco ascendente de la entrada digital DI.

En la interfaz de notificación (Página 9728) se indica el estado almacenado del contador en vez del estado actual del contador. El bit STS_DI indica el nivel de la señal de congelación.

La carga directa del contador no provoca que cambie el estado almacenado que indica el contador.

En el modo isócrono, se muestra en la interfaz de notificación el estado del contador que se congeló en el instante del último flanco ascendente antes de T_i.

Si se cierra la puerta SW, actúa como parametrizada, de cancelación o de interrupción. La entrada digital DI permanece activa incluso cuando la puerta SW está cerrada.

Interfaz de notificación ampliada

Si el módulo 1Count5V está enchufado detrás de un IM 151 que soporta la lectura y escritura de interfaces más amplias de datos de usuario, el valor actual de contaje puede leerse desde los bytes 8 a 11 de la interfaz de notificación (Página 9728).

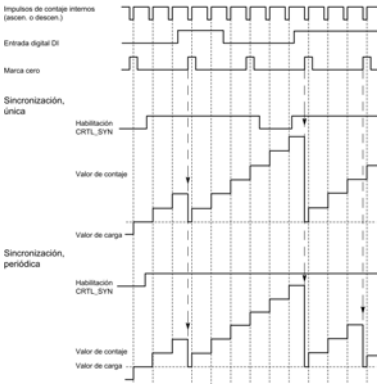
Habilitar HW para sincronización

Sincronización con DI y señal de paso por cero

Si se ha parametrizado la habilitación HW para sincronización, el DI sirve como habilitación HW. Cuando está activada la habilitación HW, se carga el valor de carga en el 1Count5V con la señal cero del encóder.

Se puede elegir entre la sincronización única y la periódica (parámetro "Sincronización (Página 9726)").

La figura siguiente muestra los estados del contador con una sincronización única y periódica.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control (Página 9726) "Habilitación de la sincronización" (CTRL_SYN) debe estar activado.
- Sincronización única: si el bit de habilitación está activado y la habilitación HW está presente, la primera señal cero carga el 1Count5V con el valor de carga.
- Sincronización periódica: Si el bit de habilitación está activado y la habilitación HW está presente, la primera señal de paso por cero y cada señal de paso por cero subsiguiente cargan el 1Count5V con el valor de carga.
- Tras una sincronización correcta, el bit de realimentación (Página 9728)STS_SYN se activa y el LED SYN se enciende. El bit de control (Página 9726)RES_STS desactiva el bit de realimentación y apaga el LED.
- La señal de un interruptor libre de rebotes puede servir de señal de referencia para la habilitación HW.
- El bit de realimentación (Página 9728)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación (Página 9728) activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9728)STS_DI indica el nivel de la entrada digital.

Seleccionar sincronización

Sincronización

Si se ha seleccionado "Sincronización con flanco ascendente" en "Función DI (Página 9717)", es posible elegir entre la sincronización única y la periódica.

- Una vez: si el bit de habilitación está activado, el primer flanco carga el 1Count5V con el valor de carga.
- Periódica: si el bit de habilitación está activado, el primer flanco y cada flanco subsiguiente cargan el 1Count5V con el valor de carga.

Interfaz de notificación y control para la lectura del recorrido

Interfaz de control para la lectura del recorrido

Nota

Para el 1Count5V, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (las salidas) del 1Count5V para la lectura del recorrido.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor de carga directo, preliminar		
Byte 4	Bit 7	Acuse del error	EXTF_ACK
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	Inicio de la desactivación del bit de estado	RES_STS
	Bit 1	Habilitación de la sincronización	CTRL_SYN
	Bit 0	Bit de control puerta SW	SW_GATE
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	reservado = 0	
	Bit 1	Carga preliminar del contador	LOAD_PREPARE
	Bit 0	Carga directa del contador	LOAD_VAL
Bytes 6 a 7	reservado = 0 [1]		

[1] Falta en la interfaz de datos de usuario avanzada

Significado de los bits de control

Bit	Significado
CTRL_SYN	Este bit habilita la sincronización
EXTF_ACK	Acuse del error Los bits de error deben ser acusados mediante el bit de control EXTF_ACK una vez eliminada la causa.
LOAD_PREPARE	Carga preliminar del contador El valor de los bytes 0 a 3 se acepta como valor de carga
LOAD_VAL	El valor de los bytes 0 a 3 se carga directamente como nuevo valor de contaje.
RES_STS	Inicio de la desactivación del bit de estado Los bits de estado se desactivan mediante el proceso de realimentación entre el bit RES_STS y el bit RES_STS_A.
SW_GATE	Bit de control puerta SW La puerta SW se abre/cierra mediante la interfaz de control con el bit SW_GATE.

Interfaz de notificación para la lectura del recorrido

Nota

Para el 1Count5V, los datos siguientes de la interfaz de notificación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7
- Bytes 8 ... 11 (interfaz de datos de usuario avanzada)

Asignación de la interfaz de notificación

La siguiente tabla muestra la asignación de la interfaz de notificación (entradas) del 1Count5V para la lectura del recorrido.

Dirección	Asignación		Denominación
Bytes 0 a 3	Valor de contaje o valor de contaje almacenado con función de congelación en la entrada digital		
Byte 4	Bit 7	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 6	reservado = 0	
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	Desactivación de los bits de estado en curso	RES_STS_A
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Byte 5	Bit 7	Estado de sentido descendente	STS_C_DN
	Bit 6	Estado de sentido ascendente	STS_C_UP
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	reservado = 0	
	Bit 1	Estado de DI	STS_DI
	Bit 0	Estado de la puerta interna	STS_GATE
Byte 6	Bit 7	Paso por cero	STS_ND
	Bit 6	Límite inferior de contaje	STS_UFLW
	Bit 5	Límite superior de contaje	STS_OFLW
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	reservado = 0	
	Bit 1	reservado = 0	
	Bit 0	Estado de la sincronización	STS_SYN

Dirección	Asignación	Denominación
Byte 7	reservado = 0	
Bytes 8 a 11	Valor de contaje [1]	

[1] Interfaz de datos de usuario avanzada

Nota sobre los bits de realimentación

Bit	Significado
ERR_24V	Cortocircuito en la alimentación del encóder El bit de error tiene que ser acusado con el bit de control (Página 9723)EXTF_ACK. Aviso de diagnóstico, si se ha parametrizado.
ERR_LOAD	Error en función de carga Los bits de control (Página 9723)LOAD_VAL y LOAD_PREPARE no pueden estar activados al mismo tiempo durante la transferencia. Esto tiene como consecuencia la activación del bit de estado ERR_LOAD, similar a cargar un valor incorrecto (que no se acepta).
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
RES_STS_A	Desactivación de los bits de estado en curso
STS_C_DN	Estado de sentido descendente
STS_C_UP	Estado de sentido ascendente
STS_DI	Estado de DI El estado de la DI se indica en todos los modos de operación con el bit STS_DI en la interfaz de notificación.
STS_GATE	Estado de la puerta interna: Contaje en curso.
STS_LOAD	Función de carga en curso
STS_ND	Paso por cero en el rango de contaje sin sentido principal de contaje. El bit debe desactivarse con el bit de control (Página 9723)RES_STS.
STS_OFLW STS_UFLW	Límite superior de contaje rebasado Límite inferior de contaje rebasado Ambos bits deben ser desactivados.
STS_SYN	Estado de la sincronización Al terminar la sincronización se activa el bit STS_SYN. Debe desactivarse con el bit de control (Página 9723)RES_STS.

Parametrizar para el modo Fast

Seleccionar el comportamiento en caso de STOP de la CPU/del maestro

Ajuste del comportamiento en caso de STOP de la CPU/del maestro

Es posible parametrizar la reacción del 1Count5V en caso de fallo del controlador principal.

Parámetro	Estado del 1Count5V en caso de STOP de la CPU/ del maestro	¿Qué ocurre en el caso de una nueva parametrización?
Desactivar DO	El modo de operación actual se cancela. La puerta se cierra y la salida digital se bloquea. El valor de comparación 1 y 2 y el valor de carga se resetean. El límite superior e inferior, la función y el comportamiento de las salidas digitales y el tiempo de integración se toman de los parámetros.	Los parámetros modificados son aceptados y tienen efecto inmediato.
Continuar modo de operación[1)]	El modo de operación actual continúa. La puerta y la salida digital conservan sus estados.	La puerta se cierra. El modo de operación actual se cancela. La salida digital se bloquea. Los parámetros modificados son aceptados y tienen efecto inmediato.

1) Si el modo de operación seleccionado debe continuar al cambiar la CPU/el maestro de STOP a RUN (arranque), la CPU/ el maestro no puede borrar las salidas.
Posible solución: En la parte del programa de usuario que se procesa durante el arranque active el bit de control SW_GATE y transfiera los valores al 1Count5V.

Reparametrización automática

Una reparametrización de la estación ET 200S mediante la CPU/del maestro DP tiene lugar en los siguientes casos:

- POWER ON de la CPU/del maestro DP
- POWER ON de la IM 151-1 / IM 151-1 FO
- Después de un fallo en la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ maestro DP
- Al insertar el 1Count5V
- POWER ON o inserción del módulo de potencia correspondiente

Seleccionar evaluación de señal

Evaluación de señal A, B

La evaluación de la señal mediante A, B permite contar direccionalmente. Son posibles diferentes modos de evaluación dependiendo de lo que se parametrice:

- Encóder rotativo (Página 9494)
  El 1Count5V puede contar los flancos de las señales. Normalmente, sólo se evalúan los flancos en A (evaluación simple). Para obtener una resolución más alta, en la parametrización (parámetro "Evaluación de señal") se puede seleccionar si las señales deben evaluarse de forma simple (Página 9495), doble (Página 9495) o cuádruple (Página 9495).
  La evaluación múltiple sólo es posible con encóders incrementales asimétricos con señales A y B desfasadas 90 grados.

Nota

La indicación 500 kHz en la frecuencia de contaje se refiere a la frecuencia máxima de las señales A o B. Así, en la evaluación doble se obtienen como máximo 1 MHz en los impulsos de contaje o máximo 2 MHz en la evaluación cuádruple.

Activar diagnóstico de los pares de señales

Vigilancia de los pares de señales

Los pares de señales A y /A, B y /B, N y /N pueden vigilarse.

Active para ello la casilla de verificación correspondiente.

Nota

El "Diagnóstico N, /N" no se puede activar hasta que se ha activado el "diagnóstico A, /A y B, /B".

Seleccionar el sentido de contaje

Sentido de contaje

Para adaptar el sentido de contaje al proceso es posible seleccionar lo siguiente.

- Sentido de contaje normal
- Sentido de contaje invertido

Selección de la función de puerta

Descripción

Al parametrizar las puertas HW y SW se puede definir si la puerta interna debe interrumpir o cancelar el contaje.

- Cancelar contaje (Página 9489): El contaje comienza de nuevo desde el principio después de una reactivación de la señal de la puerta.
- Interrumpir contaje (Página 9489): El contaje se reanuda desde el último valor actual de contaje después de una reactivación de la señal de la puerta.

Función DI

Seleccionar función DI

Descripción

La "función DI" permite seleccionar la función para la entrada digital:

- Entrada
  La entrada digital funciona como entrada "normal".
- Puerta HW (Página 9488), controlada por nivel
  La puerta HW se abre y se cierra con la señal correspondiente de la entrada digital
- Sincronización con flanco ascendente (Página 9732)
  Los flancos ascendentes de la entrada digital permiten establecer el contador al valor de carga.
- Habilitar HW para sincronización (Página 9733)
  La entrada digital sirve como habilitación HW. Cuando está activada la habilitación HW, el contador se activa en el valor de carga con la señal de paso por cero del encóder.

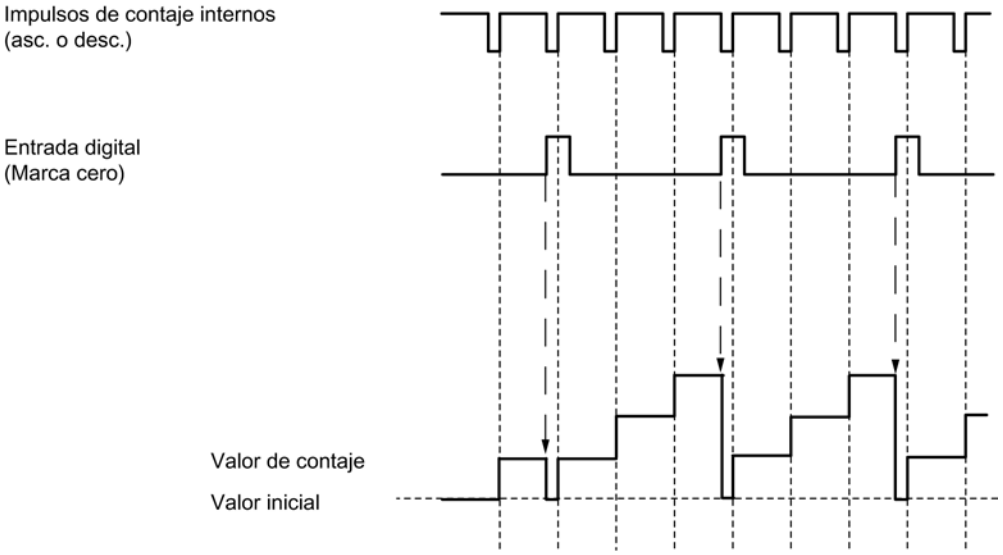
El ajuste predeterminado es "Entrada".

Sincronización con flanco ascendente

Sincronización

Si se ha parametrizado "Sincronización con flanco ascendente", el flanco ascendente de una señal de referencia en la entrada sirve para poner el 1Count5V al valor de inicio (= valor de carga).

La figura siguiente muestra los estados del contador con una sincronización con flanco ascendente.



Se aplican las siguientes condiciones:

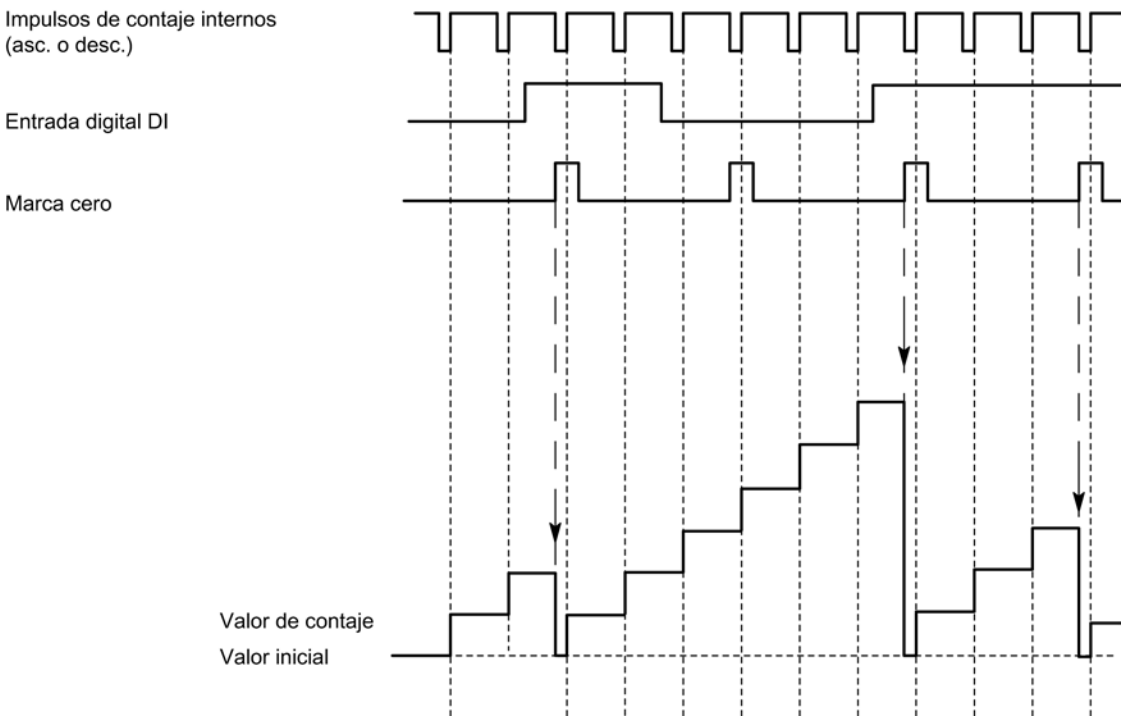
- El modo de operación "Fast Mode" debe estar activo (puerta HW).
  - Con la sincronización activada, el primer flanco y cualquier flanco posterior cargan el 1Count5V con el valor inicial (= valor de carga).
- Como señal de referencia puede utilizarse la señal de un interruptor sin rebotes o la señal de paso por cero de un encóder rotativo.
- El bit de realimentación STS_DI indica el nivel de la señal de referencia.

Habilitar HW para sincronización

Sincronización con DI y señal de paso por cero

Si se ha parametrizado la habilitación HW para sincronización, el DI sirve como habilitación HW. Cuando está activada la habilitación HW, el valor inicial (= valor de carga) se carga en el 1Count5V con la señal de paso por cero del encóder.

La figura siguiente muestra los estados del contador con habilitación HW en el DI y sincronización con flanco ascendente.



Se aplican las siguientes condiciones:

- El modo de contaje tiene que haber sido iniciado mediante la puerta SW.
- El bit de control "Habilitación de la sincronización" (CTRL_SYN) está activado.
- Tras una sincronización correcta, el bit de realimentación (Página 9735)STS_SYN se activa y el LED SYN se enciende. El bit de control RES_STS desactiva el bit de realimentación y apaga el LED.
- La señal de un interruptor libre de rebotes puede servir de señal de referencia para la habilitación HW.
- El bit de realimentación (Página 9735)STS_DI indica el nivel de la señal de referencia.

En el modo isócrono, el bit de realimentación activado STS_SYN indica que el flanco ascendente de la entrada digital se encontraba entre el instante T_i del ciclo actual y el instante T_i del ciclo pasado.

Seleccionar señal de entrada de la puerta HW

Señal de entrada puerta HW

Al parametrizar "Función DI = puerta HW", el nivel de la entrada digital se puede invertir con el parámetro "Señal de entrada puerta HW".

El bit de realimentación (Página 9735)STS_DI indica el nivel de la entrada digital.

Definir el valor de carga

Valor de carga

Es posible especificar un valor de carga para el 1Count5V.

Dicho valor de carga se toma directamente como valor inicial.

Rango de valores admisible: -16.777.216 a +16.777.215

Interfaz de realimentación para el Fast Mode

Nota

Para el 1Count5V, los datos siguientes de la interfaz de realimentación son datos relacionados, es decir, coherentes:

- Bytes 0 … 3

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 1Count5V para el modo Fast.

Dirección	Asignación		Denominación
Bytes 0 a 3	Bit 31	Señal de vida	LZ
	Bit 30	Modo isócrono registrado	STS_TIC
	Bit 29	Error de parametrización	ERR_PARA
	Bit 28	Error de grupo • Cortocircuito, alimentación de encóders • Cortocircuito / rotura de hilo señal del encóder	EXTF
	Bit 27	Estado de DI	STS_DI
	Bit 26	Estado de sentido ascendente / descendente	STS_DIR
	Bit 25	Estado de la puerta interna	STS_GATE
	Bits 0 a 24	Valor de contaje	

Significado de los bits de realimentación

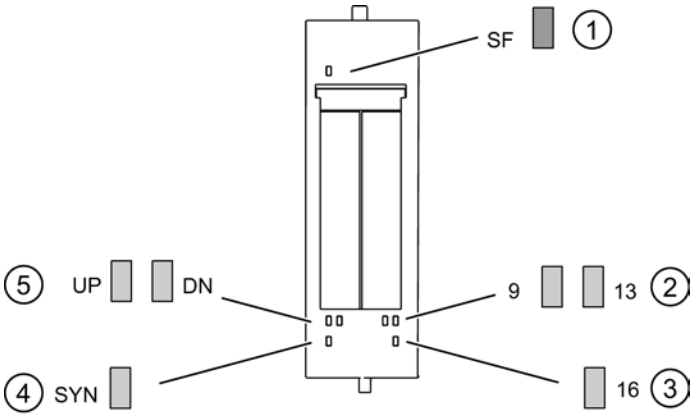
Bit	Significado
ERR_PARA	En la parametrización del módulo hay parámetros incorrectos.
EXTF	Error de grupo Causas posibles: • Cortocircuito, alimentación de encóders • Cortocircuito o rotura de hilo señal del encóder EXTF se desactiva cuando se eliminan las causas del error.

Bit	Significado
LZ	La señal de vida se invierte en cada actualización de la interfaz de realimentación, es decir, que se invierte el último valor enviado.
STS_DI	El bit muestra el estado de la entrada digital DI.
STS_DIR	Estado de sentido; • cuando cambia el valor del encóder de posiciones elevadas a posiciones más bajas (incl. paso por cero) → "1 " • cuando cambia el valor del encóder de posiciones bajas a posiciones más elevadas (incl. paso por cero) → "0 "
STS_GATE	Estado de la puerta interna: Contaje en curso.
STS_TIC	El modo isócrono (si está parametrizado) se ha registrado.

Diagnóstico del 1Count5V

Diagnóstico con indicadores LED

Indicador LED del 1Count5V



① Error de grupo (rojo)
② Indicador de estado de la salida digital (verde)
③ Indicador de estado de la entrada digital (verde)
④ Indicador de estado de la sincronización (verde)
⑤ Indicador de estado del sentido de contaje (verde)

Indicadores de estado y error mediante los LEDs del 1Count5V

La siguiente tabla muestra los indicadores de estado y error del 1Count5V.

Evento (LED)							Causa	Solución
SF	UP	DN	SYN	9	13	16		
ON							No hay parametrización Hay un aviso de diagnóstico.	Compruebe la parametrización. Evalúe el aviso de diagnóstico.
	ON						Estado del bit menos significativo del contador, si el contador cuenta adelante.	
		ON					Estado del bit invertido menos significativo del contador, si el contador cuenta atrás.	
			ON				Sincronización ejecutada (sólo en los modos de contaje; imagen del bit de realimentación STS_SYN).	
				ON			DO 1 activada	
					ON		DO 2 activada	
						ON	DI activada	

Tipos de error

En el manual de producto del módulo de interfaz utilizado en la estación ET 200S encontrará información sobre la estructura del diagnóstico de canal.

Tipos de error del 1Count5V

La tabla siguiente muestra los tipos de error del 1Count5V.

Tipo de error		Significado	Solución
1_D	00001: cortocircuito	Cortocircuito de la alimentación del encóder o actuador.	Compruebe el cableado del encóder. Corregir el cableado del proceso.
5_D	00101: Sobretemperatura	Salida digital sobrecargada.	Corregir el cableado del proceso.
6_D	00110: Rotura de hilo	Cable del actuador interrumpido.	Corregir el cableado del proceso.
9_D	01001: Error	Ha aparecido un error interno en el módulo.	Sustituir el módulo.
16_D	10000: Error de parametrización	Módulo no parametrizado.	Corregir la parametrización.
26_D	11010: error externo	Rotura de hilo/cortocircuito de las señales de 5 V del encóder: A, /A, B, /B, N, /N,	Corregir la parametrización

11.2 Leva electrónica (S7-300, S7-400)

11.2.1 Control por levas

11.2.1.1 Conceptos básicos del control por levas

¿Qué es el control por levas?

Control por levas

Los secuenciadores de levas (levas electrónicas) se aplican para controlar funciones dependientes de la posición o del tiempo. Son superiores a los secuenciadores de levas mecánicos sobre todo por su gran flexibilidad, p. ej. se pueden realizar modificaciones por software durante el funcionamiento.

Una leva de salida genera señales de conmutación en función de la posición del eje o de un encóder externo. Se diferencia entre leva de recorrido y leva de tiempo.

Una leva de salida se define mediante:

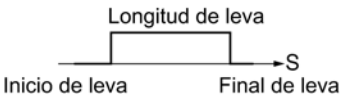
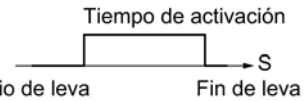
- Posición inicial y posición final (leva de recorrido (Página 9738))
- Posición inicial y duración de conexión (leva de tiempo (Página 9738))

Características de los tipos de levas

Tipos de levas

Cualquier leva se puede parametrizar como leva de recorrido o como leva de tiempo.

La tabla siguiente compara las características de los dos tipos de levas.

	Leva de recorrido	Leva de tiempo
Representación		
Parametrización	Se parametriza: • Inicio de leva • Fin de leva • Sentido de actuación • Tiempo de anticipación	Se parametriza: • Inicio de leva • Tiempo de activación • Sentido de actuación • Tiempo de anticipación

	Leva de recorrido	Leva de tiempo
Sentido de actuación	Existen dos sentidos de actuación posibles: • positivo: la leva se activa al inicio, cuando el eje se desplaza en dirección ascendente respecto al valor real. • negativo: La leva se activa al final, cuando el eje se desplaza en dirección descendente respecto al valor real. Es posible también activar simultáneamente los dos sentidos de actuación.	Existen dos sentidos de actuación posibles: • positivo: la leva se activa al inicio, cuando el eje se desplaza en sentido ascendente respecto al valor real. • Negativo: La leva se activa a su inicio, cuando el eje se desplaza en dirección descendente respecto al valor real. Es posible también activar simultáneamente los dos sentidos de actuación.
Activación	La leva se activa: • al inicio de la leva cuando el sentido del movimiento del eje es positivo y se ha ajustado el sentido de actuación positivo. • al final de la leva cuando el sentido del movimiento del eje es negativo y se ha ajustado el sentido de actuación en negativo. • cuando el valor real se encuentra dentro del rango de la leva.	La leva se activa: • al principio de la leva cuando el sentido del movimiento del eje es positivo y se ha ajustado el sentido de actuación positivo, o cuando el sentido de movimiento del eje es negativo y se ha ajustado el sentido de actuación negativo Tras la conexión, se agota todo el tiempo de conexión de la leva. Incluso si tras la activación de la leva se modifica el sentido del movimiento del eje. Si durante el tiempo de conexión de una leva cuyo inicio se ha vuelto a rebasar, la leva no se vuelve a disparar.
Desactivación	La leva se desactiva si • se ha rebasado el recorrido parametrizado, • el sentido de actuación se detecta como contrario al sentido del movimiento y no se ha parametrizado histéresis, • el valor real no se encuentra en el rango de la leva.	La leva se desconecta cuando el tiempo parametrizado ha transcurrido.
Longitud del recorrido	La longitud del recorrido de la leva se define por el inicio y el final de la leva. El principio y el fin de la leva pertenecen a la parte activa de la leva.	La longitud del recorrido de la leva depende de la velocidad a la que el eje se desplaza durante el tiempo de conexión de la leva.
Duración de conexión	La duración de conexión de la leva depende de la velocidad a la que el eje rebase la longitud del recorrido de la leva.	La duración de conexión de la leva se parametriza con el tiempo de conexión.

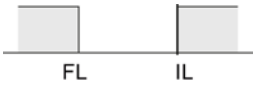
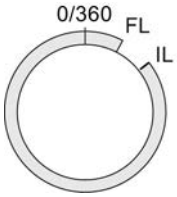
Detección del sentido

El sentido del movimiento del eje se determina del modo siguiente:

- Con cada impulso del encóder incremental.
- Con cada telegrama sin errores del encóder SSI.

Leva invertida

En "Modificar flancos de leva" puede ocurrir que durante un breve tiempo surja una leva invertida si el nuevo principio de leva es mayor que el antiguo inicio de leva. La tabla siguientes muestra el efecto de una leva inversa con un eje lineal y uno rotativo.

Leva invertida en un eje lineal	Leva invertida en un eje rotativo
	
El principio de leva es mayor que el fin de leva	El principio de leva es más positivo que el fin de leva
Con ambos tipos de eje debe observarse una distancia mínima de 4 pulsos entre el principio de leva y el fin de leva.	

Pistas normales y resultados de la pista

Pistas de levas

Con las 32 pistas es posible controlar un máximo de 32 procesos de conexión distintos. Las pistas se pueden evaluar con ayuda de las señales de realimentación.

Cada una de las primeras 13 pistas (de la 0 a la 12) tiene asignada una salida digital (Q0 a Q12) del FM 352, capaz, por ejemplo, de controlar directamente un contactor conectado. En el FM 452 cada una de las primeras 16 pistas (de la 0 a la 15) tiene asignada una salida digital (Q0 a Q15).

Resultado de pista

Hay disponible un máximo de 128 levas a las que se puede asignar cualquier pista.

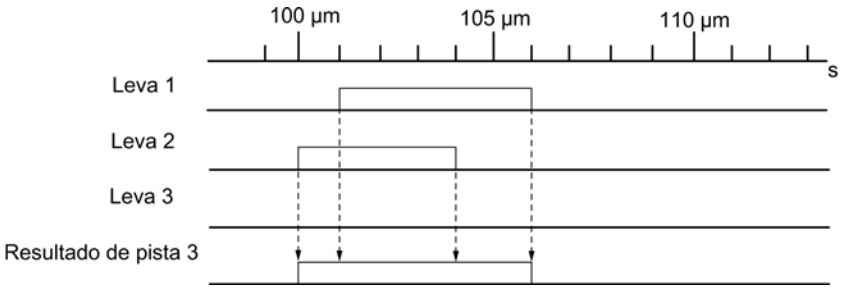
Cada pista puede estar asignada a varias levas. El resultado de la pista es el resultado de la operación lógica OR de todos los valores de leva de esta pista.

Ejemplo de un resultado de pista

Al parametrizar se definen las levas siguientes para la pista 3:

Leva	Inicio de leva	Fin de leva
1	101 µm	106 µm
2	100 µm	104 µm

La figura siguiente muestra el resultado de la pista.



Habilitación de pista

Para que los resultados de las pistas comprendidas 0 a 12 se puedan aplicar como señal de pista en las salidas digitales de Q0 a Q12 del FM 352 es preciso habilitar las pistas utilizadas. En el FM 452 se dispone de las pistas 0 a 15.

Habilitación externa de una pista

La habilitación externa de la pista 3 se puede parametrizar en los datos máquina. La señal de pista 3 se combina lógicamente con la entrada digital I3 mediante la operación lógica AND antes de que pueda conectar la salida digital Q3 del FM 352.

Así pues, la salida digital Q3 se activa cuando se cumplen las condiciones siguientes:

- La pista correspondiente está habilitada.
- Queda por lo menos una leva activa en esta pista (resultado de pista = 1).
- La entrada digital I3 correspondiente se ha activado mediante un evento externo.

En el FM 452 se dispone de las pistas 3 a 10 para una habilitación externa.

Activación de las señales de pista

Las señales de pista entre 0 y 12 (correspondientes a las salidas digitales entre Q0 y Q12) se pueden activar con el control de levas o la CPU. En el FM 452 es posible activar las señales de pista 0 a 15.

Pistas especiales

Definición

Es posible parametrizar como pistas especiales las pistas entre 0...2:

- Pista 0 o 1: Pista de leva de contaje
- Pista 2: Pista de leva de freno
  Para que la pista se pueda activar, se evalúa la entrada I0.

Requisito

Para trabajar con pistas especiales se tienen que cumplir los requisitos siguientes:

- Las levas están parametrizadas en la pista.
- El procesamiento de levas está activado.
- La pista correspondiente está habilitada.
- La pista está parametrizada como pista especial.

Pista de leva de contaje

Una pista de leva de contaje cuenta el cambio de estado de los resultados de pista en esa pista.

Es preciso indicar un valor de contaje superior para la pista de la leva de contaje e iniciar la función de contaje.

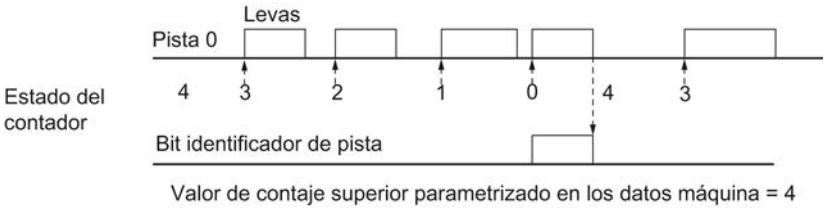
Todo flanco ascendente del resultado de pista disminuye en 1 el valor de contaje de la pista correspondiente.

Mientras el valor de contaje de la pista de leva de contaje no sea 0, el bit indicador de la pista se mantiene en 0.

Si el valor de contaje alcanza el valor 0, el bit indicador de la pista se activa y, con la parametrización correspondiente, la señal de la pista (Página 9745) se activa.

Con el siguiente flanco descendente del resultado de pista (todas las levas de esta pista están desactivadas) el bit identificador de la pista se desactiva de nuevo y el contador regresa de nuevo al valor de contaje predeterminado.

La siguiente figura muestra la conmutación de una pista de levas de contaje.



Pista de leva de freno

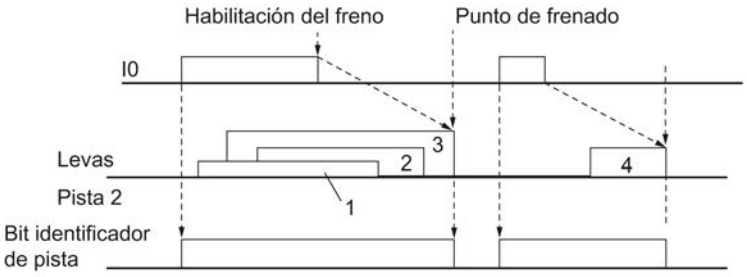
Para emplear la pista 2 como pista de leva de freno la entrada digital I0 tiene que estar interconectada.

Con el flanco ascendente de la señal en I0 el bit identificador de la pista se activa de inmediato.

El bit identificador de la pista se desactiva de nuevo si

- en I0 no hay más señales "1" y posteriormente
- el flanco descendente del resultado de pista de la pista 2 se detecta.

La siguiente figura muestra la conmutación de una pista de levas de freno.



1 a 4 identifican 4 levas que se aplican a la pista de leva de freno

En el ejemplo el bit identificador de la pista se desactiva mediante los flancos descendentes de las levas 3 o 4.

Histéresis

Definición

Una vibración mecánica en el eje puede provocar cambios en el valor real de la posición. Si el valor real de la posición fluctúa en un flanco de una leva o dentro de una leva activa con un solo sentido de actuación, esta leva se activará y desactivará de forma permanente. La histéresis impide esta activación.

Una histéresis configurada depende del valor real y rige para todas las levas. Se activa en cuanto se detecta un cambio de sentido. Una histéresis también es efectiva cuando no hay ninguna leva en la posición actual del eje.

Reglas para el rango de histéresis

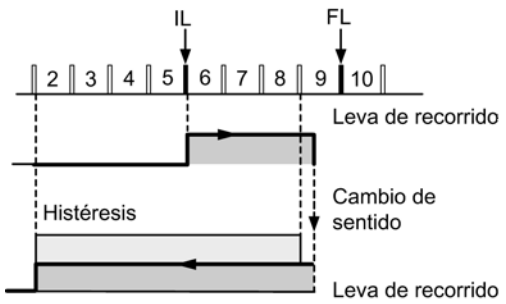
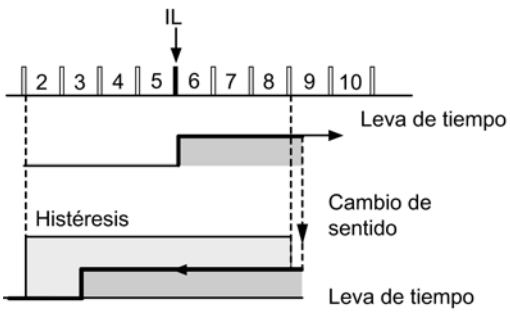
En el rango de histéresis se aplican las reglas siguientes:

- La histéresis siempre pasa a activa con un cambio de sentido.
- Dentro de la histéresis la indicación del valor real se mantiene constante.
- Dentro de la histéresis el sentido no se define de nuevo.
- Dentro de la histéresis una leva de recorrido ni se activa ni se desactiva.

- Dentro de la histéresis una leva de tiempo no se activa y una leva de recorrido activada se desactiva en cuanto transcurre el tiempo de conexión parametrizado (no al final de la histéresis).
- Tras abandonar el rango de histéresis el FM x52 determina:
  - el valor real de la posición,
  - el sentido de movimiento actual del eje,
  - los estados actuales de todas las levas.
- El rango de histéresis rige para todas las levas.

Inversión de sentido en una leva con histéresis

En la tabla siguiente se muestra la respuesta de una leva cuando se produce un cambio de sentido. Para ello es preciso diferenciar entre la respuesta de una leva de recorrido y una leva de recorrido. El sentido de actuación de la leva es **positivo**.

Leva de recorrido	Leva de tiempo
	
La histéresis se activa tras la detección del cambio de sentido. La leva se desactiva en cuanto se sale de la histéresis.	La leva se mantiene siempre activada durante el tiempo de conexión parametrizado.
Leva Histéresis	

Ajuste dinámico

Tarea

El ajuste dinámico sirve para compensar los retardos de los elementos de conexión conectados.

Tiempo de anticipación

El período de retardo se puede parametrizar como tiempo de anticipación que se puede asignar a cada leva. Es posible especificar un tiempo de anticipación por leva. El tiempo de anticipación se aplica al inicio y al fin de la leva.

Recorrido de anticipación

El recorrido de anticipación de una leva se calcula continuamente en función de la velocidad actual y del tiempo de anticipación. Toda la leva se desplaza ese recorrido en la dirección del valor real. El rango parametrizado se denomina "rango estático" y el rango calculado atendiendo al tiempo de anticipación se llama "rango dinámico".

Recorrido de anticipación = tiempo de anticipación × velocidad actual del eje

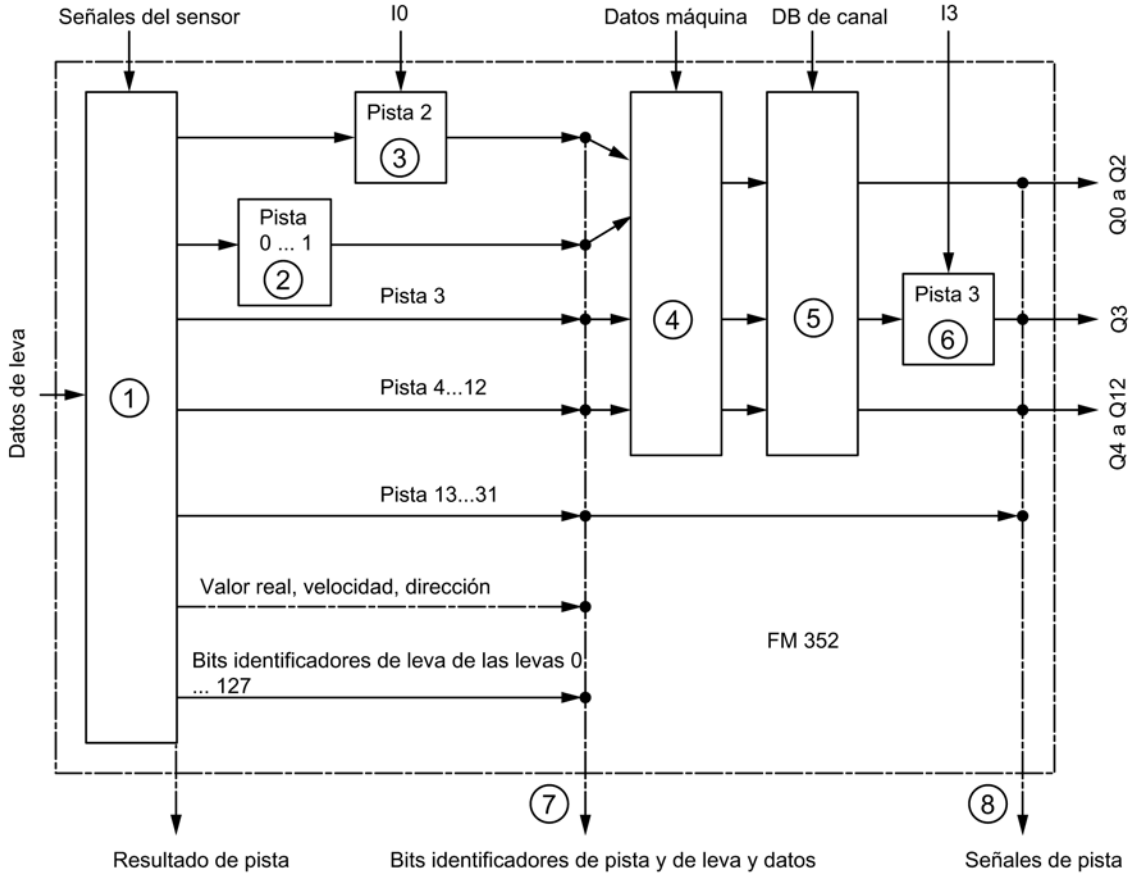
El cálculo de los recorridos de anticipación de todas las levas se realiza dentro de 1/4 del tiempo de anticipación más largo parametrizado en el FM x52.

Si se parametriza un tiempo de anticipación muy prolongado para una leva, se reduce la dinámica del procesamiento de levas.

Interfaces de la leva

Vista general

Tomando como ejemplo el FM 352, la figura siguiente muestra las principales interfaces de la leva electrónica para hacer más clara la relación entre datos, entradas y salidas. Las diferencias respecto al FM 452 se muestran en la tabla que se incluye a continuación.



Nº	Descripción
①	En el procesamiento de levas del FM x52 los bits identificadores de leva se calculan a partir de las condiciones de conexión y el valor real actual. Por otra parte, los resultados de pista resultan a partir de la asignación de las levas a las pistas.
②	Si se ha parametrizado la pista 0 ó 1 como pista de leva de contaje, el resultado de pista de la leva (punto 1) se combina lógicamente con el resultado de contaje del bit identificador de la pista. Por lo demás, el bit identificador de pista es igual al resultado de pista.
③	Si se ha parametrizado la pista 2 como pista de leva de freno, el resultado de pista de la leva (Punto 1) se combina lógicamente con la entrada I0 del bit identificador de la pista. Por lo demás, el bit identificador de pista es igual al resultado de pista.
④	Con los datos máquina es posible controlar si los bits identificadores de pista calculados hasta el instante se transfieren a las pistas de 0 a 12 (en el FM 352) ó 0 a 15 (en el FM 452) de la leva electrónica o, con independencia de ello, si se activan directamente con la habilitación de pista (TRACK_EN).
⑤	Las señales de las pistas 0 a 12 (ó 0 a 15) se habilitan a través de TRACK_EN y la función de contaje, a través de CNTC0_EN / CNTC1_EN.
⑥	FM 352: La señal de pista de la pista 3 se puede combinar lógicamente con la entrada digital I3 mediante la operación lógica AND, si se ha parametrizado esta posibilidad en los datos máquina (EN_IN_I3).
	FM 452: Las señales de pista de las pistas 3 hasta 10 se pueden combinar lógicamente con las entradas digitales I3 hasta I10 mediante la operación lógica AND, si se ha parametrizado esta posibilidad en los datos máquina (EN_IN_I3 a EN_IN_I10).
⑦	Aquí (esto es, antes de combinarse con los datos máquina y los de canal) todos los bits identificadores de pista y de leva se pueden leer con la petición ACTPOS_EN o CAMOUT_EN. Para las pistas de 3 a 31 el bit identificador es igual al resultado de pista (punto 1).
⑧	FM 352: Tras la conexión con los datos máquina (por ejemplo, tipo de eje, final de carrera por software, recorrido por vuelta de encóder, ...) y los datos de canal, las señales de pista de las pistas de 0 a 12 están disponibles en las señales de realimentación. Las señales de las pistas 13 a 31 son idénticas a los bits identificadores de pista del punto 7. Las señales de las pistas 0 a 12 también están disponibles en las entradas digitales Q0 a Q12.
	FM 452: Tras la combinación lógica con los datos máquina y los datos de canal las señales de pista de las pistas 0 a 15 están disponibles en las señales de realimentación. Las señales de las pistas 16 a 31 son idénticas a los bits identificadores de pista del punto 7. Las señales de las pistas 0 a 15 también están disponibles en las entradas digitales Q0 a Q15.

Señales del encóder

Encóders compatibles

Encóders incrementales conectables

Se soportan encóders incrementales (Página 9748) de dos impulsos con un desfase de 90º con o sin señal de paso por cero:

- Encóder con señales de salida asimétricas con nivel de 24 V
  - Frecuencia límite = 50 kHz
  - Longitud del cable: máx. 100 m, apantallado
- Encóder con señales de salida simétricas con un puerto diferencial de 5 V según RS 422.
  - Frecuencia límite = 1 MHz
  - Con una alimentación de tensión de 5 V: Longitud del cable: máx. 32 m, apantallada
  - Con una alimentación de tensión de 24 V: Longitud del cable: máx. 100 m, apantallado

Nota

Si el encóder de 5 V no emite ninguna señal de paso por cero y se ha activado la comprobación de rotura de hilo, es preciso interconectar externamente las entradas de señal de paso por cero N y /N para que estas entradas presenten un nivel distinto (p. ej. N a 5 V, /N a la masa).

Detectores de proximidad conectables

Los siguientes detectores de proximidad (Página 9750) pueden conectarse al FM x52:

- Detectores de proximidad con nivel de 24 V (interruptores de proximidad)
- Frecuencia límite = 50 kHz
- Longitud del cable máx. 100 m

Encóders absolutos conectables

Se soportan encóders absolutos (Página 9750) con interfaz serie. La transferencia de la información de recorrido se realiza de forma síncrona según el protocolo SSI (**i**nterfaz **s**erial**s**íncrona). El FM x52 sólo soporta el código GRAY. Según la disposición de los bits de datos en el telegrama resultan los formatos de datos "abeto", "medio abeto" y "alineación derecha".

Entre los encóders absolutos se distingue entre

- Encóders monovuelta
  Los encóders monovuelta reflejan el rango completo de medida en una vuelta del encóder.
- Encóders multivuelta
  Los encóders multivuelta reflejan el rango completo de medida en varias vueltas del encóder.

Tipo de encóder	Longitud/tipo de telegrama
Encóder monovuelta	13 bits medio abeto
Encóder monovuelta	13 bits alineación derecha
Encóder monovuelta	25 bits alineación derecha
Encóder multivuelta	25 bits abeto
Encóder multivuelta	25 bits alineación derecha
Escuchar	Abeto
Escuchar	alineación derecha
Ajuste especial: Encóder multivuelta como encóder monovuelta	25 bits medio abeto

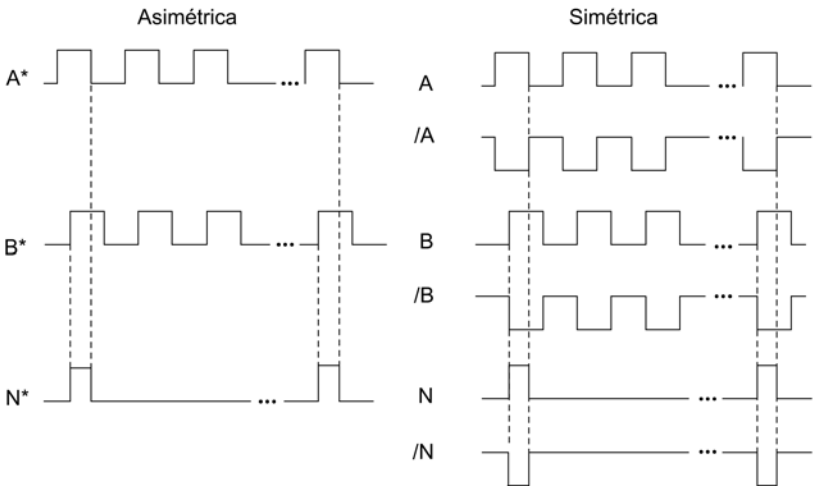
La velocidad de la transferencia de datos depende de la longitud del cable:

- Máx. 125 kHz con 320 m longitud de cable apantallado
- Máx. 250 kHz con 160 m longitud de cable apantallado
- Máx. 500 kHz con 60 m longitud de cable apantallado
- Máx. 1 MHz con 20 m longitud de cable apantallado

Encóder incremental

Formas de señal

La figura siguiente muestra las formas de señal de los encóders con señales de salida asimétricas y simétricas.

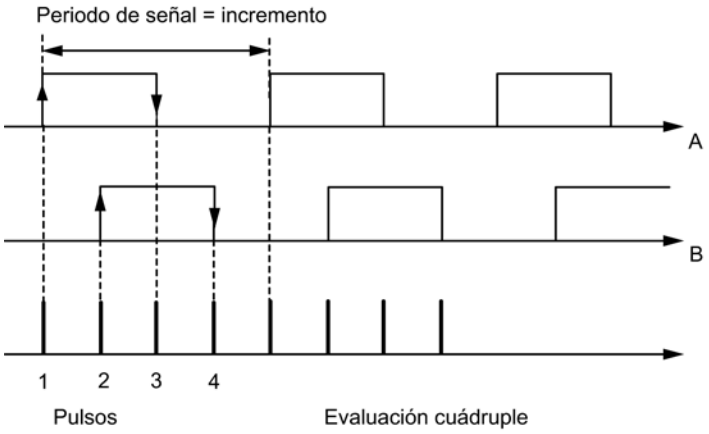


Evaluación de señales

Incrementos

Un incremento identifica un período de señal de las dos señales A y B de un encóder. Este valor se indica en las especificaciones técnicas de un encóder y/o en su placa de características.

La figura siguiente muestra la relación de incrementos e impulsos.



Impulsos

El FM x52 evalúa los 4 flancos de las señales A y B de cada incremento (evaluación cuádruple).

1 incremento (dado por el encóder) = 4 impulsos (evaluación FM)

Tiempos de respuesta

El FM x52 tiene los tiempos de respuesta siguientes para encóders incrementales conectados:

- Tiempo mínimo de respuesta = tiempo de ciclo de leva + tiempo de conexión de los elementos conectados
- Tiempo máximo de respuesta = 2 × tiempo de ciclo de leva + tiempo de conexión de los elementos conectados

Ejemplo de tiempos de respuesta

Ejemplo del tiempo de respuesta mínimo y máximo con un alcance de 16 levas:

- Tiempo de ciclo de la leva: aprox. 20 µs
- Tiempo de conexión del hardware: aprox. 150 µs

Tiempo de respuesta mínimo = 20 µs + 150 µs = 170 µs

Tiempo de respuesta máximo = 2 × 20 µs + 150 µs = 190 µs

Nota

El tiempo de respuesta se puede compensar mediante una parametrización adecuada de las levas o bien mediante el ajuste dinámico.

Imprecisión

La imprecisión es la diferencia entre el tiempo de respuesta máximo y mínimo. En los encóders incrementales:

Imprecisión = tiempo de ciclo de la leva

Nota

Si es posible discriminar el tiempo de conexión de hardware en el FM x52 y el tiempo de conexión de los elementos conectados, se puede asegurar una activación segura de la leva cuando es más larga que el recorrido que realiza dentro del tiempo de ciclo de la leva.

Detectores de proximidad

Definición

Los detectores de proximidad son interruptores simples sin información de sentido, los cuales emiten impulsos. El sentido se indica con los datos máquina de "sentido del detector de proximidad".

⚠ PRECAUCIÓN

Se pueden producir daños materiales.

Una indicación de sentido incorrecta puede causar fallos muy graves en la instalación (p. ej. causados por un control erróneo de grupos eléctricos).

Compruebe la indicación de sentido en la puesta en servicio y cada vez que cambie un detector de proximidad.

Interpretación de la señal

En un detector de proximidad, se cuenta el flanco ascendente de la señal A*.

Encóders absolutos (SSI)

Evaluación de impulsos del encóder absoluto

1 incremento (dado por el encóder) = 1 impulso (evaluación FM)

Escuchar

"Escuchar" significa: Un encóder absoluto se utiliza paralelamente en dos módulos (p.ej. FM x51 y FM x52). El módulo de posicionamiento FM x51 es el módulo maestro y envía impulsos al encóder absoluto; la leva electrónica FM x52 es esclavo y escucha las señales del telegrama SSI.

Ajuste "Incrementos/Vuelta del encóder" y "Número de vueltas" en la configuración del maestro. La "velocidad de transferencia" no es relevante. En "Longitud del telegrama" seleccione, según el tipo de encóder, "Escuchar" o "Escuchar en alineación derecha".

Tiempos de respuesta

El FM x52 tiene los siguientes tiempos de respuesta para encóders absolutos:

- Tiempo mínimo de respuesta = tiempo de ejecución de telegramas + tiempo de ciclo de levas + tiempo de conexión de los elementos conectados
- Tiempo máximo de respuesta = 2 × tiempo de ejecución de telegramas + tiempo monoestable + 2 × tiempo de ciclo de leva + tiempo de conexión de los elementos conectados
- En el caso de encóders absolutos programables:
  Tiempo máximo de respuesta = tiempo de ejecución de telegramas + tiempo monoestable + 2 × tiempo de ciclo de leva + tiempo de conexión de los elementos conectados + 1/ frecuencia de pasos máx.

Tiempo monoestable

Para el tiempo monoestable rigen los valores límite siguientes:

- Tiempo monoestable mínimo: > 15 µs
- Tiempo monoestable máximo: < 64 µs

Los encóders con valores fuera de los límites que aquí se indican no son admisibles.

Tiempos de ejecución de telegramas

Los tiempos de ejecución de telegramas dependen de la velocidad de transferencia:

Velocidad de transferencia	Tiempo de ejecución de telegramas a 13 bits	Tiempo de ejecución de telegramas a 25 bits
0,125 MHz	112 µs	208 µs
0,250 MHz	56 µs	104 µs
0,500 MHz	28 µs	52 µs
1,000 MHz	14 µs	26 µs

Ejemplo de tiempos de respuesta

Este ejemplo muestra cómo calcular el tiempo de respuesta máximo y mínimo. En el ejemplo no se emplea ningún encóder programable.

- Tiempo de ciclo de la leva: aprox. 20 µs con máx. 16 levas
- Tiempo de conexión del hardware: aprox. 150 µs
- Tiempo de ejecución de telegramas: 26 µs a una velocidad de transferencia de 1 MHz (telegramas de 25 bits)
- Tiempo monoestable: 20 µs (según el encóder: normalmente 20 - 40 µs)

Tiempo de respuesta mínimo = 26 µs + 20 µs + 150 µs = 196 µs

Tiempo de respuesta máximo = 2 × 26 µs + 20 µs + 2 × 20 µs + 150 µs = 262 µs

Nota

El tiempo de respuesta se puede compensar mediante una parametrización adecuada de las levas o bien mediante el ajuste dinámico.

Imprecisión

La imprecisión es la diferencia entre el tiempo de respuesta máximo y mínimo.

En el caso de un encóder absoluto es:

- Imprecisión = tiempo de ciclo de leva + tiempo de ejecución de telegrama + tiempo monoestable

En el caso de los encóders absolutos programables es:

- Imprecisión = tiempo de ciclo de leva + tiempo de ejecución de telegrama + tiempo monoestable
  + 1/frecuencia de pasos máx.

Nota

Si es posible discriminar el tiempo de conexión de hardware en el FM x52 y el tiempo de conexión de los elementos conectados, se puede asegurar una activación segura de la leva cuando es más larga que el recorrido que realiza dentro del tiempo de ciclo de la leva.

11.2.1.2 Funciones de FM x52

Escribir y activar los datos máquina

Escribir y activar los datos máquina

Con los datos máquina se adapta el FM x52 al eje y al encóder.

El programa de usuario permite modificar los parámetros durante el funcionamiento.

Todos los parámetros se depositan en el DB de parámetros:

- Los datos máquina se encuentran en el DB de parámetros, en las direcciones que van del 3.1 al 104.0.

El número del DB de parámetros se tiene que registrar en el DB de canal correspondiente en cada caso.

Los parámetros pueden introducirse con el editor DB, o bien introducirse cómodamente en los cuadros de diálogo "Sistema de medición", "Eje", "Encóder" y "Pistas" y se escriben con la función "Exportar" en el DB de parámetros.

Los parámetros de un DB de parámetros existente pueden importarse a los cuadros de diálogo con la función "Importar".

Primera parametrización

Si el módulo todavía no contiene ningún dato máquina (señal de realimentación PARA = 0), proceda del siguiente modo para una primera parametrización sin el cuadro de diálogo de parametrización:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Cargue el DB de parámetros en la CPU.
3. Active el bit de inicio siguiente en el DB de canal:
   - Escribir datos máquina (MDWR_EN = 1)
4. Llame las instrucciones CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) en el programa de usuario cíclico.

Modificar datos máquina

Para modificar datos máquina existentes (señal de realimentación PARA = 1) mediante el programa de usuario proceda del siguiente modo:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Active los bits de inicio en el DB de canal:
   - Escribir datos máquina (MDWR_EN = 1)
   - Activar datos máquina (MD_EN = 1)
3. Llame las instrucciones CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) en el programa de usuario cíclico.
4. Compruebe si los datos de leva hasta el instante son compatibles con los datos máquina modificados.
5. En cualquier caso, escriba de nuevo los datos de las levas parametrizadas, modificados o no (CAM1WR_EN … CAM8WR_EN).

Si se definen los bits de inicio al mismo tiempo, la instrucción CAM_CTRL o CAM_CTRL_452 se encarga de que las peticiones se procesen en la secuencia correcta.

De lo contrario, se deberán cambiar siempre los datos máquina en la siguiente secuencia:

1. Escribir datos máquina
2. Activar datos máquina

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentario
35.0	MDWR_EN	BOOL	FALSE	1 = Escribir datos máquina
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina

Nota

Si se han modificado parámetros relevantes para la sincronización, al activar los datos máquina se borra la sincronización. Además, las funciones se restauran y todos los datos máquina y los datos de leva del módulo se borran.

Son parámetros relevantes para la sincronización:

- Tipo de eje
- Final del eje rotativo
- Tipo de señal
- Recorrido por vuelta
- Incrementos por vuelta
- Cantidad vueltas
- Coordenada del punto de referencia
- Reajuste del encóder absoluto
- Forma de redisparar el punto de referencia
- Adaptación del sentido
- Alcance
- Inicio y fin del final de carrera por software

Leer datos máquina

Leer datos máquina

Para leer los datos máquina actuales del módulo proceda del modo siguiente:

1. Active el bit de inicio siguiente en el DB de canal:
   – Leer datos máquina (MDRD_EN = 1)
2. Llame las instrucciones CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) en el programa de usuario cíclico.

De este modo se depositan en la CPU los datos máquina del DB de parámetros.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
37.1	MDRD_EN	BOOL	FALSE	1 = Leer datos máquina

Escribir datos de leva

Escribir datos de leva

Con los datos de leva se define el modo y el funcionamiento de las levas así como su asignación a las pistas.

Los datos de leva se encuentran en el DB de parámetros a partir de la dirección 108.0. Se encuentran agrupados en paquetes de 16 levas cada uno.

Los datos de la leva están activos inmediatamente después de su escritura.

Para escribir los datos de leva sin cuadro de diálogo de parametrización proceda del siguiente modo:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Cargue el DB de parámetros en la CPU.
3. Active los bits de inicio en el DB de canal (CAM1WR_EN … CAM8WR_EN)
4. Llame las instrucciones CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) en el programa de usuario cíclico.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentarios
35.3	CAM1WR_EN	BOOL	FALSE	1 = Escribir datos de leva 1 (levas de 0 a 15)
35.4	CAM2WR_EN	BOOL	FALSE	1 = Escribir datos de leva 2 (levas de 16 a 31)
35.5	CAM3WR_EN	BOOL	FALSE	1 = Escribir datos de leva 3 (levas de 32 a 47)
35.6	CAM4WR_EN	BOOL	FALSE	1 = Escribir datos de leva 4 (levas de 48 a 63)
35.7	CAM5WR_EN	BOOL	FALSE	1 = Escribir datos de leva 5 (levas de 64 a 79)
36.0	CAM6WR_EN	BOOL	FALSE	1 = Escribir datos de leva 6 (levas de 80 a 95)
36.1	CAM7WR_EN	BOOL	FALSE	1 = Escribir datos de leva 7 (levas de 96 a 111)
36.2	CAM8WR_EN	BOOL	FALSE	1 = Escribir datos de leva 8 (levas de 112 a 127)

Leer los datos de leva

Leer datos de leva

Para leer los datos de leva actuales del módulo proceda del modo siguiente:

1. Active el bit de inicio siguiente en el DB de canal:
   – Leer los datos de leva (CAM1RD_EN … CAM8RD_EN)
2. Llame las instrucciones CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) en el programa de usuario cíclico.

De este modo se depositan en la CPU los datos de leva del DB de parámetros.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
37.2	CAM1RD_EN	BOOL	FALSE	1 = Leer datos de leva 1 (levas de 0 a 15)
37.3	CAM2RD_EN	BOOL	FALSE	1 = Leer datos de leva 2 (levas de 16 a 31)
37.4	CAM3RD_EN	BOOL	FALSE	1 = Leer datos de leva 3 (levas de 32 a 47)
37.5	CAM4RD_EN	BOOL	FALSE	1 = Leer datos de leva 4 (levas de 48 a 63)
37.6	CAM5RD_EN	BOOL	FALSE	1 = Leer datos de leva 5 (levas de 64 a 79)
37.7	CAM6RD_EN	BOOL	FALSE	1 = Leer datos de leva 6 (levas de 80 a 95)
38.0	CAM7RD_EN	BOOL	FALSE	1 = Leer datos de leva 7 (levas de 96 a 111)
38.1	CAM8RD_EN	BOOL	FALSE	1 = Leer datos de leva 8 (levas de 112 a 127)

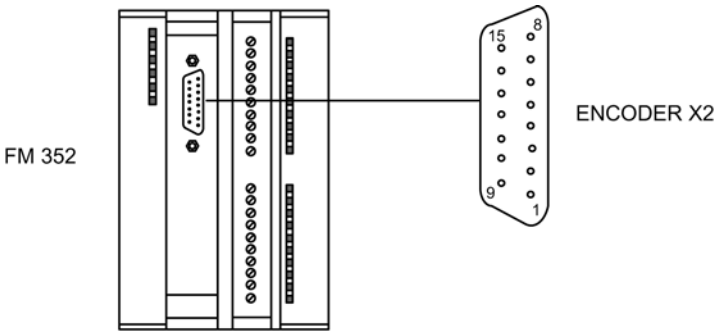
11.2.1.3 Uso del FM 352

Asignación de terminales de los conectores frontales del FM 352

Descripción de la interfaz del encoder

Posición del conector hembra sub-D

En la figura siguiente se muestra la posición de montaje y el nombre del conector hembra del módulo. En el conector hembra sub-D se puede conectar un detector de proximidad, un encóder incremental o un encóder absoluto (SSI).



Asignación de la interfaz del encóder (X2)

Pin	Nombre	Detector de proximidad	Encoder incremental	Encóder absoluto
1	A*	Señal de encoder A (24 V)		---
2	CLS	---	---	Pulso de desplazamiento SSI
3	/CLS	---	---	Pulso de desplazamiento SSI inverso
4	B*	---	Señal de encoder B (24 V)	---
5	DC 24 V	Alimentación del sensor 24 V		
6	DC 5.2 V	---	Alimentación del sensor 5,2 V	
7	M	Masa		
8	N*	---	Señal de marca cero (24 V)	---
9	RE	---	Salida tipo PNP/NPN 2)	---
10	N	---	Señal de marca cero (5 V)	---
11	/N	---	Señal de marca cero invertida (5 V)	---
12	/B / /CLI 1)	---	Señal del sensor B (5 V) invertida	Pulso de desplazamiento SSI inverso
13	B / CLI 1)	---	Señal del sensor B (5 V)	Pulso de desplazamiento SSI

Pin	Nombre	Detector de proximidad	Encoder incremental	Encóder absoluto
14	/A / /DAT	---	Señal de encoder A (5 V) invertida	Datos SSI invertidos
15	A / DAT	---	Señal del sensor A (5 V)	Datos SSI

[1] En modo de escucha

[2] **Nota:** si se conecta un encóder incremental de otro fabricante en contrafase (salida tipo PNP/NPN), debe tenerse en cuenta lo siguiente:

- Salida tipo PNP: conecte RE (9) con la masa (7).
- Salida tipo NPN: conecte RE (9) con +24 V (5).

Asignación del conector frontal

Conector frontal

En el conector frontal de 20 polos se conectan:

- las salidas digitales y los elementos de conexión (entradas);
- la tensión de alimentación de los encóders y de las salidas digitales.

Asignación de terminales del conector frontal (X1)

La tabla siguiente contiene la asignación de terminales del conector frontal del FM 352.

Terminal	Nombre	Función
1	L+	Alimentación del encoder DC 24 V y salidas digitales
2	M	Masa de la alimentación del encoder y salidas digitales
3	I 0	Habilitación del freno
4	I 1	Medición de longitud/Detección de bordes/Ajustar valor real al vuelo
5	I 2	Sensor del punto de referencia
6	I 3	Habilitación de la señal de pista 3
7	Q 0	Salida digital 0
8	Q 1	Salida digital 1
9	Q 2	Salida digital 2
10	Q 3	Salida digital 3
11	Q 4	Salida digital 4
12	Q 5	Salida digital 5
13	Q 6	Salida digital 6
14	Q 7	Salida digital 7
15	Q 8	Salida digital 8
16	Q 9	Salida digital 9
17	Q 10	Salida digital 10
18	Q 11	Salida digital 11

Terminal	Nombre	Función
19	Q 12	Salida digital 12
20	–	–

Encontrará más información sobre el diseño y el cableado en las instrucciones de servicio (http://support.automation.siemens.com/WW/view/es/2103044) de la leva electrónica FM 352.

Tensión auxiliar para el encoder y las salidas digitales (L+, M)

La tensión auxiliar DC 24 V del encoder y las salidas digitales se supervisa:

- por si hay una rotura de hilo en la línea de 24 V
- por si se corta la alimentación.

En la interfaz del encóder (conector hembra sub-D X2) se proporcionan DC 24 V y DC 5,2 V para los diferentes tipos de encóder.

Alimentaciones de carga

La alimentación de la carga DC debe satisfacer los siguientes requisitos:

Como alimentación de la carga sólo debe utilizarse una pequeña tensión DC ≤ 60 V separada con seguridad de la red. La separación segura puede estar efectuada según los requisitos que figuran, entre otros, en:

- VDE 0100 Parte 410/HD 384-4-41/IEC 364-4-41
  (como pequeña tensión funcional con separación segura),
- VDE 0805/EN 60950/IEC 950
  (como pequeña tensión de seguridad SELV) o bien
- VDE 0106 Parte 101.

4 entradas digitales (de I 0 a I 3)

En las 4 entradas digitales es posible conectar conectores libres de rebotes (de 24 V tipo P) o sensores sin contacto (detectores de proximidad de 2 ó 3 hilos).

Las entradas digitales no se supervisan si presentan cortocircuito o rotura de hilos y carecen de aislamiento galvánico respecto de la masa del módulo.

13 salidas digitales (de Q 0 a Q 12)

Por medio de 13 salidas digitales se indica el estado correspondiente (on/off) de las pistas entre 0 y 12. Las salidas digitales carecen de aislamiento galvánico respecto a la masa del módulo.

Son posibles las cargas siguientes:

- Tensión de trabajo 24 V
- Carga de corriente 0,5 A/resistente al cortocircuito

El estado de cada salida se indica en el LED correspondiente.

Funciones de control de levas

Influencia de las funciones en el comportamiento de conmutación de las levas de tiempo

Cambios en el valor real

Es posible obviar una leva de tiempo con funciones que modifiquen el valor real:

- Ajustar valor real
- Ajustar valor real al vuelo
- Decalaje de señal cero
- Redisparar el punto de referencia

Activación de una leva de tiempo

Si se obvia el principio de una leva de tiempo mediante una de las funciones arriba mencionadas, esta leva se activará siempre y cuando el sentido real del movimiento del eje se corresponda con el sentido de actuación parametrizado para la leva. El tiempo de conexión parametrizado se cuenta.

Nota

En posición de parada, el sentido del movimiento depende de las fluctuaciones del valor real.

Si desea tener en cuenta un sentido de movimiento definido también en posición de reposo, deberá parametrizar una histéresis mayor que las fluctuaciones del valor real en reposo.

Si el eje no está en movimiento, se conserva el último sentido de movimiento calculado.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

Las funciones con modificación del valor real en un eje rotativo pueden conducir a una activación no deseada de las levas de tiempo.

En cualquier caso, con un eje rotativo ajuste las levas de tiempo como "no válidas" a fin de influir en el valor real mediante las funciones arriba indicadas.

Ajustar valor real / Ajustar valor real al vuelo / Cancelar preselección del valor real

Definición

Las funciones "Ajustar valor real", "Ajustar valor real al vuelo" permiten asignar una nueva coordenada al estado actual del encóder. Así, el sistema de coordenadas se desplaza en el valor: $REAL_{nuevo}$ - $REAL_{actual}$

Donde:

- $REAL_{nuevo}$ es el valor especificado
- $REAL_{actual}$ es el valor real en el instante de la ejecución

Calcular coordenadas nuevas

Todas las posiciones especificadas en el sistema de coordenadas desplazado se calculan con la fórmula siguiente:

$$Coordenada_{nueva} = Coordenada_{antigua} + (REAL_{nuevo} - REAL_{actual})$$

Requisitos

- El eje tiene que estar sincronizado.
- En "Ajustar valor real al vuelo": La entrada digital I1 tiene que estar interconectada.

Desarrollo de la función

1. Introduzca las coordenadas para el valor real o para el valor real al vuelo en el DB de canal.
   - Eje lineal:
     El valor real indicado se debe seleccionar de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admitido tras llamar la función.
     El valor de decalaje que resulta de $(REAL_{nuevo} - REAL_{actual})$ tiene que ser menor o igual al valor de la zona de desplazamiento admisible (máximo 100 m o 1000 m, respectivamente).
   - Eje rotativo:
     Para el valor real indicado rige:
     0 ≤ Valor real < Final del eje rotativo
2. Active los correspondientes bits de inicio en el DB de canal.
3. Llame la instrucción CAM_CTRL (Página 7453).
   Se ejecuta de inmediato "Ajustar valor real".
   "Ajustar valor real al vuelo" se ejecuta con el flanco ascendente siguiente en la entrada digital I1. El bit FVAL_DONE se activa.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.4	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.5	FVAL_EN	BOOL	FALSE	1 = Ajustar valor real al vuelo
90.0	AVAL	DINT	L#0	Coordenada para valor real
94.0	FVAL	DINT	L#0	Coordenada para valor real al vuelo
25.5	FVAL_DONE	BOOL	FALSE	1 = Ejecutado Ajustar valor real al vuelo

Efectos de la función

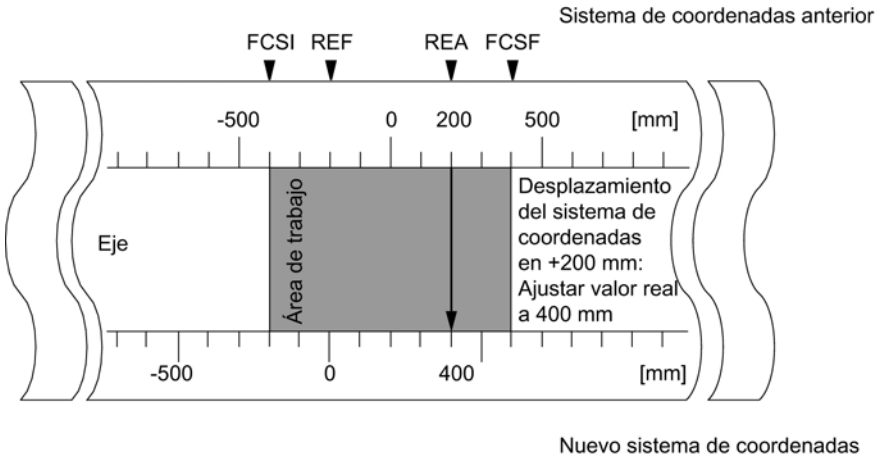
En el ejemplo "Ajustar valor real" a 400 mm (en la posición 200 mm) se muestra cómo esta función provoca un decalaje del sistema de coordenadas. De ello resultan los siguientes efectos:

- La posición de la zona de trabajo **no** se desplaza físicamente.
- A cada uno de los puntos (como, p.ej. a los finales de carrera por software) se les asignan nuevos valores de coordenadas.
- Las levas conservan sus valores de coordenadas y, por lo tanto, se encuentran en otra posición física.
- En el caso del eje sincronizado y procesamiento de levas habilitado, el valor real de la posición puede obviar flancos de levas o levas completas mediante esta función.
- El cambio de estado de la leva que normalmente dispararía una alarma se podría perder.

Nota

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9757).

La tabla siguiente muestra el decalaje del sistema de coordenadas con "Ajustar valor real" / "Ajustar valor real al vuelo".

Ajustar valor real	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	-200	200	400
	-200	0	400	600

Deshacer la función

Con la función "Cancelar preselección del valor real" se deshace el decalaje de coordenadas establecido respectivamente con "Ajustar valor real" o "Ajustar valor real al vuelo".

Un "Ajustar valor real al vuelo" iniciado no se puede borrar antes de su ejecución mediante un flanco ascendente en la entrada I1. Pero sí puede sobrescribirse con un nuevo "Ajustar valor real al vuelo".

Con un arranque del módulo se inicializan estas funciones.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentario
35.2	AVALREM_EN	BOOL	FALSE	1: Deshacer Ajustar valor real

Posibles fuentes de error

"Ajustar valor real al vuelo" y "Redisparar punto de referencia" no deben ejecutarse de forma simultánea.

Con la función "Ajustar valor real al vuelo" puede notificarse un error de operador si esta función implica que un final de carrera por software excede la zona de desplazamiento admisible con un flanco ascendente en I1. Este error de operador se notifica con una alarma de diagnóstico y queda registrado en el búfer de diagnóstico.

Ejecutar un decalaje de señal cero

Definición

Con la función "Decalaje de señal cero" se desplaza en el valor introducido la señal cero del sistema de coordenadas. El signo define la dirección del decalaje.

Calcular nueva coordenada

Todos los valores del sistema de coordenadas desplazado se calculan con la fórmula siguiente:

$\text{Coordenada}_{\text{nueva}} = \text{coordenada}_{\text{antigua}} + (\text{DSC}_{\text{nueva}} - \text{DSC}_{\text{antigua}})$

$\text{DSC}_{\text{antigua}}$ indica un decalaje anterior de señal cero. Si antes de la llamada no ha habido activo ningún decalaje de señal cero, introduzca para $\text{DSC}_{\text{antiguo}}$ el valor 0.

Con esta fórmula es posible calcular qué valores de coordenadas adoptan, por ejemplo, los finales de carrera por software.

Desarrollo de la función

1. Introduzca el valor del decalaje de señal cero en el DB de canal.
   - Eje lineal:
     El decalaje de señal cero se tiene que escoger de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admisible tras la llamada de la función.
   - Eje rotativo:
     Para el decalaje de señal cero rige:
     Valor del decalaje de señal cero ≤ Fin del eje rotativo.
2. Active el bit de inicio correspondiente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.6	ZOFF_EN	BOOL	FALSE	1 = Activar decalaje de señal cero
86.0	ZOFF	DINT	L#0	Decalaje de señal cero

Efectos en un eje lineal

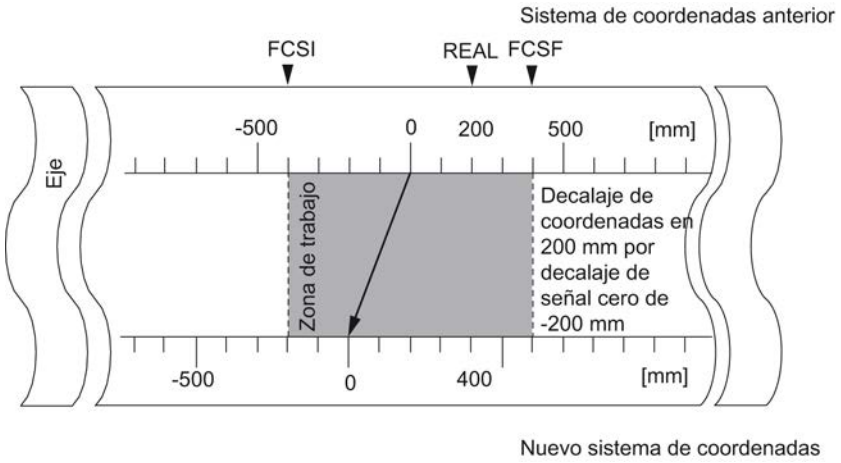
En el ejemplo de un decalaje de señal cero de -200 mm se ve cómo esta función desplaza el sistema de coordenadas en sentido positivo. De ello resultan los siguientes efectos:

- La zona de trabajo **no** se desplaza físicamente.
- A cada uno de los puntos (como, p.ej. a los finales de carrera por software) se les asignan nuevos valores de coordenadas.
- Las levas conservan sus valores de coordenadas y, de este modo, se encuentran en otra posición física.
- En el caso del eje sincronizado y procesamiento de levas habilitado, el valor real de la posición puede obviar flancos de levas o levas completas mediante esta función.
- El cambio de estado de la leva que normalmente dispararía una alarma, se puede perder.

Nota

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9757).

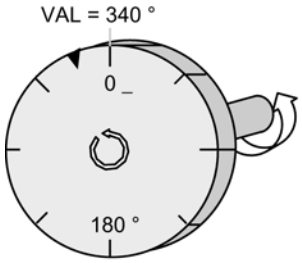
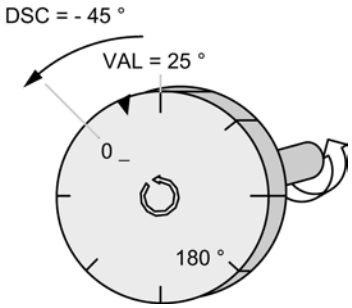
La tabla siguiente muestra el desplazamiento del sistema de coordenadas mediante el decalaje de señal cero.

Decalaje de señal cero	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	-200	200	400
	-200	0	400	600

Efectos en un eje rotativo

En el ejemplo de un decalaje de señal cero en -45° se ve cómo esta función gira el sistema de coordenadas:

La tabla siguiente muestra el giro del sistema de coordenadas mediante el decalaje de señal cero.

Herramienta con REAL = 340°	Herramienta con REAL = 25°
	

La señal cero gira -45 grados. Todos los puntos del eje adquieren nuevos valores de coordenada.

Considerando un **$DSC_{antiguo}$ = 0** resulta un nuevo valor de 385°

Como al final del eje rotativo en un sentido de giro positivo el valor real empieza de nuevo en 0 el valor real de 25º se calcula a partir de:

$Coordenada_{nueva}$ = $coordenada_{antigua}$ - (DSC_{nueva} - $DSC_{antigua}$) -final de eje rotativo

El valor **Final del eje rotativo** sólo tiene que restarse si

$Coordenada_{antigua}$ - **(DSC_{nueva} - $DSC_{antigua}$)** es mayor que Final del eje rotativo.

Pérdida de la sincronización

Si se pierde la sincronización a causa de un error o se reactiva mediante "Redisparar punto de referencia" **se mantiene** activo un decalaje de señal cero.

Deshacer función

Al definir un decalaje de señal cero de 0 se recupera el decalaje de señal cero existente.

Definir punto de referencia

Definición

Con la función "Definir punto de referencia" se sincroniza el eje. Esta función desplaza la zona de trabajo. Todos los decalajes generados por decalaje de señal cero o con "Ajustar valor real" se conservan.

La función proyecta la zona de trabajo sobre el eje. Así, con distintas entradas de valores la zona de trabajo puede encontrarse en posiciones diferentes dentro del rango físico del eje.

Requisitos

Es preciso que el procesamiento de levas esté desactivado.

Desarrollo de la función

1. Introduzca el valor de las coordenadas del punto de referencia en el DB de canal.
   - Eje lineal:
     La coordenada del punto de referencia no puede encontrarse fuera de los finales de carrera por software. Esto rige también para la coordenada del punto de referencia de un sistema de coordenadas desplazado.
   - Eje rotativo:
     Para la coordenada del punto de referencia rige:
     0 ≤ Coordenada del punto de referencia < Final del eje rotativo
2. Active el bit de inicio correspondiente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.3	REFPT_EN	BOOL	FALSE	1 = Definir coordenada del punto de referencia
98.0	REFPT	DINT	L#0	Coordenada del punto de referencia
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado

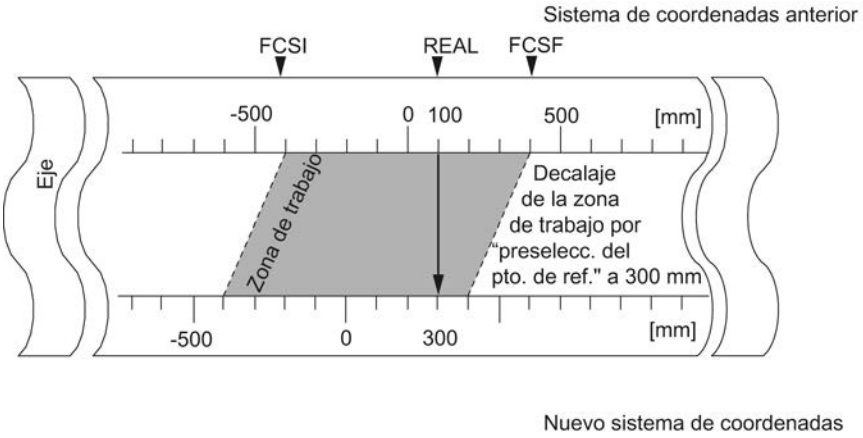
Efectos de la función

En el ejemplo "Definir punto de referencia" a 300 mm se ve cómo esta función desplaza la zona de trabajo en el eje.

De ello resultan los efectos siguientes:

- La posición real se fija en el valor de la coordenada del punto de referencia.
- La zona de trabajo se desplaza físicamente en el eje.
- Cada uno de los puntos conserva sus coordenadas originales pero se encuentran en posiciones físicas nuevas.
- El bit SYNC en las señales de realimentación está activado.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Definir punto de referencia".

Definir punto de referencia	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	-200	100	400
	-400	-200	300	400

Particularidades del encóder absoluto

Esta función es necesaria para un reajuste del encóder absoluto (Página 9790).

Modificar flancos de leva

Definición

Con la función "Modificar flancos de leva" es posible modificar durante el funcionamiento el principio de leva y, en el caso de las levas de recorrido, también el final de leva de una leva individual.

Requisitos

La leva que se quiera modificar tiene que ser válida.

Desarrollo de la función

1. Escriba el número de leva en el DB de canal.
2. En una leva de recorrido:
   Escriba el inicio de leva y el fin de leva en el DB de canal.
   En una leva de recorrido:
   Introduzca el valor del inicio de leva en el DB de canal.
3. Active el bit de inicio correspondiente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.7	CH01CAM_EN	BOOL	FALSE	1 = Escribir función para flanco de leva (1 leva)
102.0	CAM_NO	INT	0	Número de leva
104.0	CAM_START	DINT	L#0	Inicio de leva
108.0	CAM_END	DINT	L#0	Fin de leva

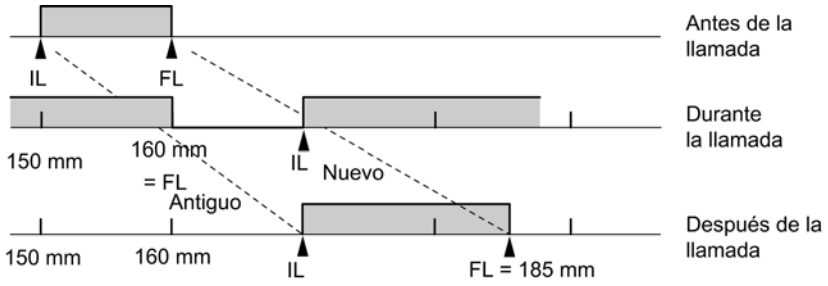
Efectos de la función

El FM x52 desplaza primero el flanco de activación y luego el de desactivación de la leva. Este proceso es independiente de la dirección en la que se desplace la leva.

Caso especial:

Con el proceso que se ha descrito anteriormente puede ocurrir que durante un breve tiempo surja una leva invertida (Página 9735) si el nuevo principio de leva es mayor que el antiguo fin de leva.

La figura siguiente muestra la modificación de los flancos de levas en pasos individuales.



Nota

Si para esta leva se ha habilitado una alarma de proceso, al detectar la leva invertida el FM x52 puede disparar una o dos alarmas de proceso, según se haya parametrizado.

Con la modificación del flanco de activación y/o de desactivación se pueden producir saltos de flancos de leva o de toda una leva.

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9757).

Los cambios de estado de la leva que normalmente dispararían una alarma de proceso, se pueden perder.

Leer los valores modificados

Los valores modificados se pueden leer con una de las peticiones CAM1RD_EN a CAM8RD_EN.

Deshacer la función

Los valores modificados se pierden al reiniciar el módulo.

Ejecutar "Cambio rápido de levas"

Definición

Con la función "Cambio rápido de levas" es posible modificar simultáneamente los datos de hasta 16 levas durante el funcionamiento.

Requisito

Las levas que se quieran modificar tienen que ser válidas.

Desarrollo de la función

1. Introduzca el número de levas que hay que modificar en el DB de canal.
2. Introduzca el número de levas que hay que modificar en el DB de canal.
3. Active los bits de inicio para los cambios deseados.
4. Introduzca los nuevos valores en el DB de canal.
5. Repita los pasos de 2 a 4 para cada leva que tenga que modificarse.
6. Active el bit de inicio correspondiente en el DB de canal

Datos utilizados en el DB de canal

Dirección Absoluta	Nombre	Tipo	valor inicial	Comentarios
37.0	CH16CAM_EN	BOOL	FALSE	1 = Función para un cambio rápido de levas (16 levas)
176.0	C_QTY	BYTE	B#16#0	Número de levas a modificar
177.0	DIS_CHECK	BOOL	FALSE	1 = Desactivar la comprobación de datos

Dirección relativa	Nombre	Tipo	valor inicial	Comentarios
+0.0	CAM_NO	BYTE	B#16#0	Número de la leva a modificar
+1.0	C_EFFDIR	BOOL	FALSE	1 = Cambio del sentido de actuación de la leva
+1.1	C_CBEGIN	BOOL	FALSE	1 = Cambio del inicio de la leva al valor CBEGIN
+1.2	C_CEND	BOOL	FALSE	1 = Cambio del final de leva/tiempo de conexión al valor CEND
+1.3	C_LTIME	BOOL	FALSE	1 = Cambio del tiempo de anticipación al valor LTIME
+1.4	CAM_OFF	BOOL	FALSE	1 = Desactivación de la leva durante el cambio de leva

Dirección relativa	Nombre	Tipo	valor inicial	Comentarios
+1.5	EFFDIR_P	BOOL	FALSE	1 = Sentido de actuación positivo (+)
+1.6	EFFDIR_M	BOOL	FALSE	1 = Sentido de actuación negativo (-)
+2.0	CBEGIN	DINT	L#0	Nuevo inicio de leva
+6.0	CEND	DINT	L#0	Nuevo fin de leva / Nuevo tiempo de conexión
+10.0	LTIME	INT	L#0	Nuevo tiempo de anticipación

Desactivar leva al modificar

Una modificación coherente del inicio y del final de la leva sólo es posible si en el instante de la modificación se ha desactivado la leva (CAM_OFF).

Comprobación de datos por parte del módulo

Con el parámetro DIS_CHECK (DB de canal) se define si se tiene que desactivar la comprobación de los datos transferidos a través del FM x52. Si se desactiva la comprobación de datos es preciso asegurarse de que sólo se transfieren valores admisibles. Si introduce valores no admisibles sin comprobación se puede producir un comportamiento impredecible del módulo.

- FALSE: El módulo comprueba todos los datos que se transfieren
- TRUE: La comprobación de los datos en referencia a los parámetros de la leva está desactivada. De este modo los datos que hay que modificar se incorporan más rápidamente en el FM x52.
  Independientemente de esta función siempre se comprueba si
  - el eje está parametrizado,
  - la cantidad de levas para modificar (C_QTY) es admisible,
  - la leva (número de leva) que tiene que modificarse es válida.

Sólo cuando todos los datos han sido comprobados y no se han detectado problemas, se activan en el módulo.

En caso de error todos los datos se rechazan.

Efectos de la función

Nota

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9757).

Leer los valores modificados

Los valores modificados se pueden leer con una de las peticiones de CAM1RD_EN a CAM8RD_EN.

Deshacer la función

Los valores modificados se pierden con el reinicio del módulo.

Ejecutar "Medición de longitud" y "Captura de bordes"

Definición

Con las funciones "Medición de longitud" y "Captura de bordes" es posible calcular la longitud de una pieza de trabajo.

La medición de la longitud y la captura de bordes están y se mantienen activos hasta que se desactivan de nuevo o se seleccionan otros métodos de medición. Si se seleccionan ambos métodos de medición a la vez, la instrucción CAM_CTRL (Página 7453) activa la medición de longitud.

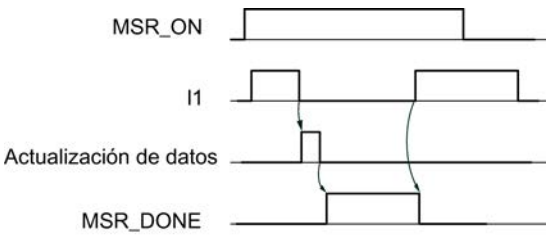
Requisitos

En la entrada I1 tiene que estar conectado un conector **libre de rebotes**.

Desarrollo de las funciones

Según el tipo de medición, el FM 352 actualiza los datos del módulo en instantes distintos. El FM 352 notifica cada actualización en un parámetro en la interfaz de realimentación.

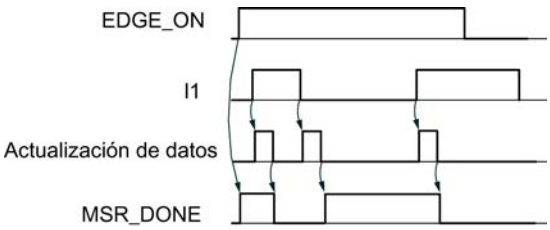
Medición de longitud



1. Active el selector de función de "Medición de longitud"
2. El flanco ascendente de la entrada I1 inicia la medición de longitud.
3. El flanco descendente de la entrada I1 pone fin a una medición en curso. El FM 352 actualiza los datos Valor inicial, Valor final y Longitud.
4. Con el parámetro activado MSR_DONE el FM 352 comunica la actualización de los datos. El parámetro indica que la medición ha finalizado. Los resultados de la medición se pueden leer.
5. El inicio de otra medición con el flanco ascendente de I1 desactiva el parámetro MSR_DONE.

Si durante una medición de longitud se desactiva la función, el FM 352 no actualiza los datos. El parámetro MSR_DONE se mantiene desactivado.

Captura de bordes



1. Dado el caso, introduzca un valor para la separación mínima de bordes en el DB de parámetros. Escriba y active los datos máquina.
2. Active el selector de función de "Captura de bordes". Se activa el parámetro MSR_DONE.
3. El flanco ascendente de la entrada I1 inicia la captura de bordes. Los resultados de la medición se actualizan y se pueden leer, el valor inicial de la medición se registra; el valor final y la longitud pasan a ser -1.
4. Tras la actualización, el FM 352 notifica el cambio desactivando el parámetro MSR_DONE.
5. El flanco descendente de la entrada I1 pone fin a una medición en curso. El FM 352 actualiza los datos del valor final de la medición y la longitud.
6. Tras la actualización, el FM 352 notifica el cambio activando el parámetro MSR_DONE. Los resultados de la medición se pueden leer.
7. El inicio de otra medición con el flanco ascendente de I1 desactiva el parámetro MSR_DONE.

Si durante una captura de bordes se desactiva la función, el FM 352 no actualiza los datos. El parámetro MSR_DONE se mantiene desactivado.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
25.1	MSR_DONE	BOOL	FALSE	1 = Medición de longitud cancelada
34.0	EDGE_ON	BOOL	FALSE	1 = Captura de bordes activada
34.2	MSR_ON	BOOL	FALSE	1 = Medición de longitud activada
38.2	MSRRD_EN	BOOL	FALSE	1 = Leer valores medidos
112.0	BEG_VAL	DINT	L#0	Valor inicial
116.0	END_VAL	DINT	L#0	Valor final
120.0	LEN_VAL	DINT	L#0	Longitud

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
4.0	EDGEDIST	DINT	L#0	**Distancia mínima entre bordes en la captura de bordes** Rango de valores admisible: • 0 ... 1 000 000 000 µm

En la captura de bordes, con la distancia mínima entre bordes se define un rango tras la detección del inicio de la medición. Si el final del proceso de medición se encuentra dentro de este rango, la medición se desestima.

El inicio de la medición se notifica en cuanto se ha recorrido la distancia mínima entre bordes.

Condiciones específicas para una medición de longitud

- La distancia entre el flanco de activación y el flanco de desactivación en la entrada I1 tiene que ser tal que su programa en la CPU pueda evaluar correctamente el resultado de la medición antes de iniciar otra.
- La distancia mínima entre el flanco ascendente y descendente en la entrada I1, al igual que entre el flanco descendente y el siguiente ascendente en la entrada I1, tiene que ser mayor que 2 ms.

Medición con errores

En caso de una medición de longitud/captura de bordes con errores, el FM 352 devuelve el valor -1 para la longitud.

Una "Medición de longitud" o "Captura de bordes" debería ejecutar un máximo de 126 cambios a cero en un sentido. Un cambio a cero es el cambio del eje rotativo del valor final del eje rotativo a 0 y a la inversa. En cuanto el FM 352 detecta más de 126 pasos por cero en un sentido, notifica una "Medición de longitud" o una "Captura de bordes" errónea, aunque luego haya detectado nuevamente pasos por cero en el otro sentido.

Una medición también es errónea si

- en un eje rotativo la longitud medida es superior a 2^{31} ,
- el FM 352 simultáneamente el flanco de activación y el de desactivación (p. ej. por rebotes del conector).

Este evento se puede comunicar como alarma de proceso.

Desplazamiento del sistema de coordenadas durante una medición de longitud

Los desplazamientos de coordenadas afectan a la longitud medida bajo las condiciones siguientes:

- Se utiliza un encóder incremental o un detector de proximidad, o bien el FM 352 en el modo de simulación.
- Se ejecuta "Definir punto de referencia" o "Redisparar punto de referencia" durante una medición de longitud.

Ejemplo

Es posible emplear la influencia arriba indicada de la longitud medida del modo que sigue:

Se dispone de un sistema en el que durante una medición de longitud siempre se produce un deslizamiento.

Con el "redisparo del punto de referencia" es posible corregir este deslizamiento de forma que se emitan los valores medidos de longitud correctos.

Redisparar punto de referencia

Definición

Con la función "Redisparar punto de referencia" es posible sincronizar el eje en función de un evento externo repetitivo.

La función se mantiene activa hasta que se desactiva de nuevo.

Requisitos

- Se emplea un encóder incremental o un detector de proximidad.
- El evento externo puede ser una señal de paso por cero de un encóder incremental o un sensor del punto de referencia en la entrada I2.

Desarrollo de la función

1. Introduzca el valor de la coordenada del punto de referencia en el DB de parámetros.
2. Introduzca el tipo de "Redisparar el punto de referencia" en el DB de parámetros.
   Para ello, existen las posibilidades siguientes:
   - Sólo se evalúa la señal de paso por cero del encóder (RETR_TYPE = 7).
   - Sólo se evalúa el sensor del punto de referencia (RETR_TYPE = 6).
   - Se evalúa la señal de paso por cero
     en sentido positivo: Se evalúa el primer flanco ascendente de la señal de paso por cero tras abandonar el sensor del punto de referencia en sentido positivo (RETR_TYPE = 0).
     en sentido negativo: Se evalúa el primer flanco descendente de la señal de paso por cero tras abandonar el sensor del punto de referencia en sentido negativo (RETR_TYPE = 1).
3. Escriba y active los datos máquina.
4. Active el selector de función del DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
34.3	REFTR_ON	BOOL	FALSE	1 = Redisparar el punto de referencia
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	Coordenada del punto de referencia
52.0	RETR_TYPE	DINT	L#0	Forma de "Redisparar el punto de referencia"

Efectos de la función

- El FM x52 evalúa la señal de paso por cero y el sensor del punto de referencia en función del sentido del movimiento del eje.
  - En caso de un sentido positivo se evalúan los flancos ascendentes.
  - En caso de un sentido de movimiento negativo se evalúan los flancos descendentes.
- La posición real se fija en el valor de la coordenada del punto de referencia.
- La zona de trabajo se desplaza físicamente en el eje.
- Cada uno de los puntos conserva su valor original, pero se encuentran en posiciones físicas nuevas.
- Los cambios de estado de la leva que normalmente dispararían una alarma, se pueden perder.
- El bit SYNC de las señales de realimentación se activa.

Nota

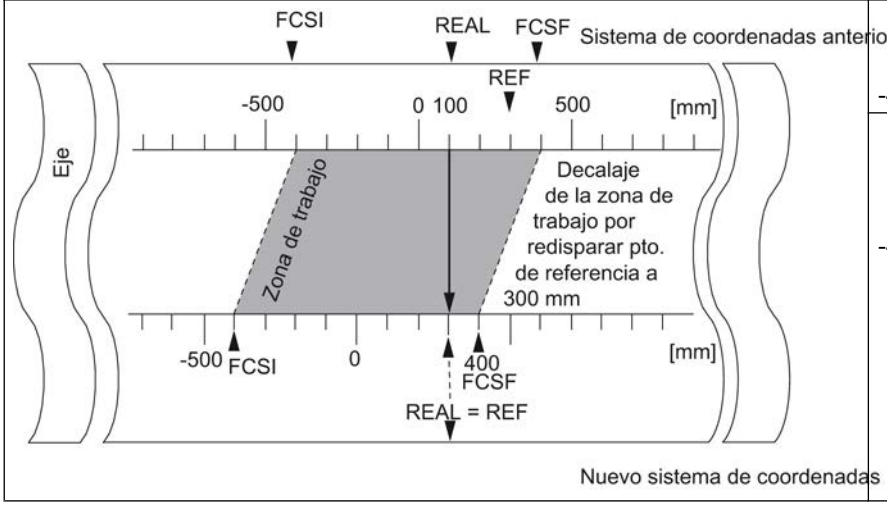
Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9757).

Ejemplo

Para este ejemplo rige:

- Del sensor del punto de referencia y la señal de paso por cero se evalúan los flancos ascendentes (sentido positivo del eje).
- La coordenada del punto de referencia tiene el valor 300 mm.
- En el instante de la ejecución no hay activo ningún decalaje de señal cero.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Redisparar punto de referencia".

Redisparar punto de referencia	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	300	100	400
	-400	300	300	400

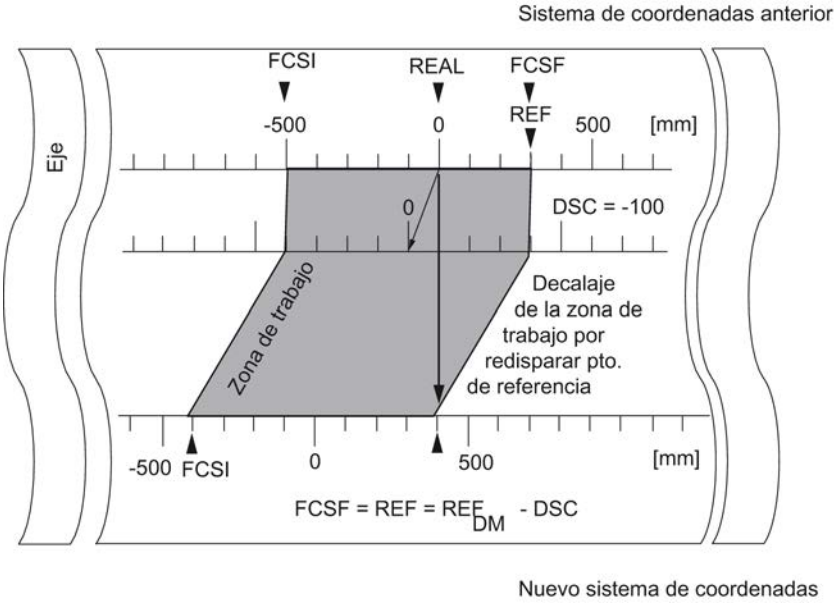
Consideración de un decalaje de señal cero

Cuando hay un decalaje de señal cero activo, se tiene en cuenta en la función "Redisparar punto de referencia". Esto significa que la coordenada del punto de referencia que se activará, se calculará conforme a la fórmula siguiente:

Ref = Ref_{DM} - Decalaje de señal cero

Ref_{DM} es el valor depositado en los datos máquina a modo de coordenada del punto de referencia.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Redisparar punto de referencia" con decalaje de señal cero.

Redisparar punto de referencia	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-500	300	0	300
	-400	400	100	400
	-400	400	400	400

Desconectar el final de carrera por software

Definición

La función "Desconectar final de carrera por software" desactiva la vigilancia del final de carrera por software de un eje lineal.

La función se mantiene activa hasta que se desactiva de nuevo. Entonces se vuelven a activar los finales de carrera parametrizados originalmente.

Desarrollo de la función

Active el selector de función en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentarios
34.4	SSW_OFF	BOOL	FALSE	1 = Final de carrera por software desconectado

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	Valor inicial	Comentarios
64.0	SSW_STRT	DINT	L# -1000000000	Final de carrera por software Inicio
68.0	SSW_END	DINT	L#1000000000	Final de carrera por software Fin

Efectos de la función

- Simulación
  - Si en el modo de simulación se rebasa un final de carrera por software se detiene el modo de simulación.
  - Si desactiva entonces la vigilancia del final de carrera por software, el modo de simulación prosigue. El eje se desplaza en la dirección indicada.
- Decalaje de señal cero con vigilancia desactivada
  Si se define un decalaje de señal cero en el que los finales de carrera por software se encuentran todavía dentro de los límites de la zona de desplazamiento, el valor real puede encontrarse fuera del rango de contaje admisible.
- Las levas que se encuentran fuera del final de carrera por software parametrizado, se pueden activar.

⚠ PRECAUCIÓN
Se pueden producir daños materiales. Si limita la zona de desplazamiento con finales de carrera por software y protege con ello su instalación, al desconectar los finales de carrera por software se pueden producir daños materiales. Por ello, al planificar la instalación cuide que el accionamiento pueda desplazarse por toda la zona de desplazamiento física.

Ejecutar la "Simulación"

Definición

La función "Simulación" brinda la posibilidad de activar la leva electrónica sin un encóder conectado.

Desarrollo de la función

1. Introduzca la velocidad de simulación en el DB de parámetros.
2. Escriba y active los datos máquina.
3. En el DB de canal seleccione como sentido de la simulación el sentido positivo o negativo.
4. Active el selector de función del DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.2	DIR_M	BOOL	FALSE	1 = Simulación en el sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Simulación en el sentido positivo
34.1	SIM_ON	BOOL	FALSE	1 = Simulación activa

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
84.0	SIM_SPD	DINT	L#0	Velocidad de simulación

Efectos al activar la simulación

- Las señales de encóder dejan de evaluarse.
- Todas las supervisiones referidas a la entrada del encóder se desconectan.
- Los errores que se hayan notificado con respecto al encóder, se desactivan.
- El FM x52 simula el movimiento de un eje a una velocidad de simulación constante.
- El procesamiento de levas se desactiva con la activación de la simulación. Sin embargo, es posible activarlo de nuevo. La sincronización se mantiene.
- El valor real de la posición cambia a partir del valor real actual, dependiendo de la velocidad y del sentido de la simulación.

Efectos al desactivar la simulación

- El procesamiento de levas se desactiva.
- En caso de un encóder incremental o un detector de proximidad, la sincronización se borra. Entonces como valor real se define el valor de la coordenada del punto de referencia.
- En el caso de un encóder absoluto se notifica el valor real de la posición, que se corresponde con la posición actual del encóder. Las señales del encóder se evalúan de nuevo tal como se han parametrizado en los datos máquina.

Valores límite

Los límites máximo y mínimo de la velocidad de simulación dependen de la resolución (Página 9795).

Velocidad

Entre la velocidad definida y la velocidad real (Página 9795) a la que funciona el módulo, puede haber diferencias.

Leer "Valores de contaje de las pistas de leva de contaje"

Definición

La función "Valores de contaje de las pistas de leva de contaje" permite leer los valores de contaje actuales.

Desarrollo de la función

1. Defina las pistas de la leva de contaje y los valores de contaje superiores en los datos máquina.
2. Escriba y active los datos máquina.
3. Habilite la función de contaje.
4. El valor de contaje se fijará en el valor de contaje superior.
5. Con cada flanco ascendente del resultado de la pista el valor de contaje se reducirá en 1.
6. Active el bit de inicio en el DB de canal para la lectura de los valores de contaje.
7. Los valores de contaje de las dos pistas se escriben en el DB de canal. Para una pista no parametrizada como pista de leva de contaje se emite 0.
8. Si el valor de contaje alcanza el valor 0, el bit identificador de la pista de leva de contaje se pone a 1.
9. Con el siguiente flanco descendente del resultado de pista, el bit identificador vuelve a 0 y el contador se pone al valor de contaje superior.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.5	CNTC0_EN	BOOL	FALSE	1 = Habilitar la función de contaje de la pista de leva de contaje 0
15.6	CNTC1_EN	BOOL	FALSE	1 = Habilitar la función de contaje de la pista de leva de contaje 1
38.3	CNTTRC_EN	BOOL	FALSE	1 = Leer valores de contaje de la pista de leva de contaje
124.0	CNT_TRC0	INT	0	Valor de contaje actual para la pista de leva de contaje 0
126.0	CNT_TRC1	INT	0	Valor de contaje actual para la pista de leva de contaje 1

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
99.0	SPEC_TRC0	BOOL	FALSE	1 = Pista 0 es pista de leva de contaje
99.1	SPEC_TRC1	BOOL	FALSE	1 = Pista 1 es pista de leva de contaje
100.0	CNT_LIM0	DINT	L#2	Valor de contaje superior para la pista de leva de contaje 0
104.0	CNT_LIM1	DINT	L#2	Valor de contaje superior para la pista de leva de contaje 1

Leer "Datos de posición y pista"

Definición

Con la función "Datos de posición y pista" se lee la posición, la velocidad y el bit identificador de pista en ese instante. Los bits identificadores de pista se capturan antes de combinarlos con los datos máquina y de canal.

El algoritmo de cálculo implantado en el FM x52 calcula cambios de velocidad mayores que 1 impulso por 4 ms. La velocidad que se muestra, por lo tanto, incluye imprecisión y no es conveniente para fines de cálculo. La velocidad interna que se emplea para dinamizar las levas es más precisa.

Desarrollo de la función

1. Active el bit de inicio en el DB de canal.
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
38.4	ACTPOS_EN	BOOL	FALSE	1 = Leer Datos de posición y pista
128.0	ACTPOS	DINT	L#0	Posición actual
132.0	ACTSPD	DINT	L#0	Velocidad actual
136.0	TRACK_ID	DWORD	DW#16#0	Bits identificadores de pista de las pistas 0 a 31

Leer "Datos de encóder"

Definición

Con la función "Datos de encóder" se leen los datos actuales del encóder así como el valor del reajuste del encóder absoluto.

Requisitos

Es posible leer el valor para el reajuste del encóder absoluto (Página 9790) después de ejecutar la función "Definir punto de referencia (Página 9762)".

Desarrollo de la función

1. Active el bit de inicio en el DB de canal.
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
38.5	ENCVAL_EN	BOOL	FALSE	1 = Leer valores del encóder
140.0	ENCVAL	DINT	L#0	Valor de encóder / Lectura del contador (representación interna)
144.0	ZEROVAL	DINT	L#0	Lectura del contador en la última señal de paso por cero (representación interna)
148.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto

Leer los "Datos de leva y pista"

Definición

Con la función "Datos de leva y pista" se leen los bits identificadores de leva y de pista actuales así como la posición en ese instante. Los bits identificadores de pista se capturan antes de combinarlos con los datos máquina y de canal.

Desarrollo de la función

1. Introduzca como identificador de tipo un 1 en el parámetro FM_TYPE del DB de canal. Con ello es posible leer los datos de leva y de pista de 24 bytes.
   Si introduce 0 como identificador de tipo sólo se leerán los bits identificadores de la leva (16 byte).
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
12.0	FM_TYPE	BOOL	FALSE	0 = FM 352 a V4.0 1 = FM 352 a partir de V5.0 y FM 452
38.6	CAMOUT_EN	BOOL	FALSE	1 = Leer datos de leva y pista
152.0	CAM_00_31	DWORD	DW#16#0	Bits identificadores de la leva para las levas de 0 a 31
156.0	CAM_32_63	DWORD	DW#16#0	Bits identificadores de leva para las levas de 32 a 63
160.0	CAM_64_95	DWORD	DW#16#0	Bits identificadores de la leva para las levas de 64 a 95
164.0	CAM_96_127	DWORD	DW#16#0	Bits identificadores de la leva para las levas de 96 a 127
168.0	TRACK_ID1	DWORD	DW#16#0	Bits identificadores de pista de las pistas 0 a 31
172.0	ACTPOS1	DINT	L#0	Posición actual

Activar "Señales de control de la leva electrónica"

Definición

Con la función "Señales de control de la leva electrónica" se habilita el procesamiento de levas así como las pistas.

Desarrollo de la función

1. Active los bits que desee en el DB de canal.
2. Con cada llamada de la instrucción CAM_CTRL (Página 7453) los datos se transfieren al módulo.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.4	CAM_EN	BOOL	FALSE	1 = Habilitar el procesamiento de levas
16.0	TRACK_EN	WORD	W#16#0	Habilitar pistas de leva de 0 a 12 Bit 0 = Pista 0

Efectos

El procesamiento de levas se inicia o se detiene según la habilitación.

Los bits identificadores de pista de las pistas habilitadas se transfieren a las señales de pista y a las salidas digitales.

Consultar las "Señales de realimentación de la leva electrónica"

Definición

Con la función "Señales de respuesta de la leva electrónica" se informa del estado actual de la leva y de las señales de pista. La coherencia entre la posición indicada y las señales de pista no está asegurada.

Desarrollo de la función

Con cada llamada de la instrucción CAM_CTRL (Página 7453) los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
23.4	CAM_ACT	BOOL	FALSE	1 = Procesamiento de levas activo
26.0	ACT_POS	DINT	L#0	Posición actual del eje
30.0	TRACK_OUT	DWORD	DW#16#0	Señales de pista actuales de las pistas 0 a 31 Bit 0 = Pista 0

Consultar "Señales de realimentación del diagnóstico"

Definición

Con la función "Señales de realimentación del diagnóstico" se recibe información sobre los eventos de diagnóstico que se han producido.

Desarrollo de la función

1. Cuando el módulo realiza una nueva entrada en el búfer de diagnóstico, activa el bit DIAG en la interfaz de realimentación. Cada aparición de un error (Página 9826) provoca una entrada en el búfer de diagnóstico.
2. Cuando el módulo detecta una petición de escritura con datos erróneos, activa el bit DATA_ERR en la interfaz de realimentación. La causa del error se registra en el búfer de diagnóstico.
3. La instrucción CAM_CTRL (Página 7453) coloca la interfaz de realimentación en el DB de canal.
4. Una vez que la instrucción CAM_DIAG (Página 7455) haya leído el búfer de diagnóstico, el módulo restablece el bit DIAG a 0.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
22.2	DIAG	BOOL	FALSE	1 = Búfer de diagnóstico modificado
22.4	DATA_ERR	BOOL	FALSE	1 = Error de datos

Configurar FM 352

Parametrizar el FM 352

Sistema de medición

Seleccionar sistema de medidas

Seleccionar sistema de medidas

Para la entrada y salida de datos es posible elegir entre los siguientes sistemas de medida:

- Milímetros (ajuste predeterminado)
- Inch
- Grados
- Impulsos

Nota

Si se modifica el sistema de medida en el cuadro de diálogo "Sistema de medida", los valores se convierten al nuevo sistema. Aquí se pueden producir errores de redondeo.

Si se modifica el sistema de medida mediante el dato máquina, los valores no se convierten automáticamente.

Si el sistema de medida cambia de o a "impulsos", el procesamiento de levas se desactivará y el eje dejará de estar sincronizado.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentario
8.0	UNITS	DINT	L#1	**Sistema de medida** Valores permitidos: • 1 = 10^{-3} mm • 2 = 10^{-4} pulgadas • 3 = 10^{-4} grados • 4 = 10^{-2} grados • 5 = Impulsos • 6 = 10^{-3} grados

Sistema de medida estándar en la ayuda en pantalla

Para la indicación de valores límite se utiliza el sistema de medida mm. Por ello, para determinar los valores límite en los otros sistemas de medida, realice la conversión siguiente:

Para convertir de		calcule
mm → pulgadas		Valor límite (inch) = valor límite (mm) × 0,1 [1]
mm → grados	10^{-4} (4 decimales)	Valor límite (grados) = valor límite (mm) × 0,1
	10^{-3} (3 decimales)	Valor límite (grados) = valor límite (mm) × 1
	10^{-2} (2 decimales)	Valor límite (grados) = valor límite (mm) × 10
mm → impulsos		Valor límite (impulsos) = valor límite (mm) x 1000

[1] En el valor máximo el número de decimales repercute en el número de dígitos que preceden a la coma. En el sistema de medida Inch se utilizan 4 decimales, por lo tanto se pueden introducir como máximo 100 000,0000 inch. El sistema de medida Milímetros utiliza 3 decimales y por lo tanto se pueden introducir como máximo 1 000 000,000 mm.

Eje

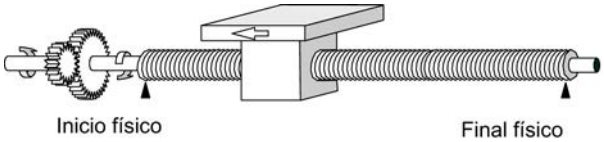
Definir tipo de eje

Seleccionar tipo de eje

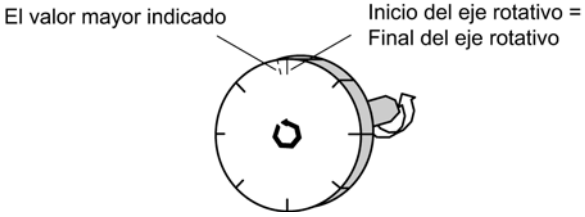
Tipos de eje

La leva electrónica soporta los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento física limitada.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
12.0	AXIS_TYPE	DINT	L#0	**Tipo de eje** Valores permitidos: • 0 = Eje lineal • 1 = Eje rotativo

Definir final de carrera por software

Final de carrera por software

Los finales de carrera por software existen sólo para el eje lineal.

El área limitada por el final de carrera por software es la zona de trabajo. Los límites de la zona de trabajo pueden ser vigilados por el FM x52.

El inicio del final de carrera por software (FCSI) debe ser siempre menor que el fin del final de carrera por software (FCSF).

Los finales de carrera por software están activos cuando el FM x52 está sincronizado.

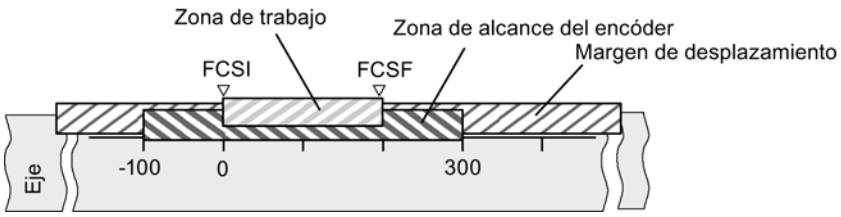
- Si se utiliza un encóder incremental o un detector de proximidad:
  Después de cada arranque del FM x52 el eje no está sincronizado. Sólo tras una sincronización se vigilan los finales de carrera por software parametrizados.
- Si se utiliza un encóder absoluto (SSI):
  El FM x52 está sincronizado después de haber recibido un telegrama completo y sin errores. A partir de este instante los finales de carrera por software son vigilados. El encóder absoluto utilizado tiene que abarcar por lo menos la zona de trabajo, incluido el final de carrera por software.

Relación entre la zona de trabajo, del encóder y la zona de desplazamiento

- La zona de trabajo es el área que el usuario define para sus tareas mediante los finales de carrera por software.
- El zona del encóder es el área que el encóder cubre claramente. En el caso de un eje lineal del módulo, el área del encóder se define de forma simétrica por encima de la zona de trabajo. Es decir, el módulo desplaza la zona del encóder de tal modo que las distancias entre los finales de carrera por software y los finales de la zona del encóder son iguales en cada caso (véase la figura en la parte de abajo).
- La zona de desplazamiento es el rango de valores que puede procesar el FM x52. La zona de desplazamiento depende de la resolución.

Se debe considerar: Zona de desplazamiento ≥ Rango del encóder ≥ Zona de trabajo

La figura siguiente muestra la relación entre los finales de carrera por software, la zona de trabajo, el rango del encóder y la zona de desplazamiento.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
64.0	SSW_STRT	DINT	L# -100.000.000	**Final de carrera por software Inicio**
68.0	SSW_END	DINT	L# 100.000.000	**Final de carrera por software Fin** Rango de valores admisible: • -1 000 000 000 µm a 1 000 000 000 µm

Definir final del eje rotativo

Final del eje rotativo

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar. Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo (0).

El valor más grande que se muestra en el caso de un eje rotativo es:

Final del eje rotativo [µm] - Resolución [µm/impulso] × 1 [impulso]

Ejemplo: Final del eje rotativo 1000 mm

La visualización pasa:

- Con un sentido de giro positivo, de 999 mm a 0 mm.
- con un sentido de giro negativo, de 0 mm a 999 mm.

Eje rotativo con encóders absolutos

En el caso de un eje rotativo con un encóder absoluto, el área del eje rotativo (0 hasta el final del eje rotativo) tiene que coincidir exactamente con el total de pasos del encóder absoluto.

$$\text{Fin de eje rotativo[µm]} = \text{número de vueltas(encoder)} * \frac{\text{Pasos(encoder)[pul]}}{\text{Vuelta}} * \text{RES}[\frac{\text{µm}}{\text{Pulso}}]$$

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
16.0	ENDROTAX	DINT	L#100000	**Final del eje rotativo** Rango de valores admisible: • 1 µm a +1 000 000 000 µm

Definir sincronización

Seleccionar la "Forma de redisparar el punto de referencia"

Redisparar el punto de referencia

Con la forma de "Redisparar el punto de referencia (Página 9771)" se definen las condiciones de sincronización del eje para el funcionamiento con un encóder incremental o un detector de proximidad.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
52.0	RETR_TYPE	DINT	L#0	**Forma de "Redisparar el punto de referencia"** Valores permitidos: • 0 = sensor del punto de referencia y señal de paso por cero en sentido + • 1 = sensor del punto de referencia y señal de paso por cero en sentido - • 6 = sólo sensor del punto de referencia • 7 = sólo señal de paso por cero

Definir la coordenada del punto de referencia

Coordenada del punto de referencia

- **Encóder incremental y detector de proximidad:**
  Con el selector de función "Redisparar punto de referencia" y un evento de sincronización, definido por la forma de "Redisparar el punto de referencia" (Página 9786), se asigna a este evento la coordenada del punto de referencia.
- **Encóder absoluto (SSI)**
  Un eje parametrizado con un encóder absoluto siempre está sincronizado después de la transmisión del primer telegrama SSI sin errores.
  Consulte al respecto también la descripción del reajuste del encóder absoluto (Página 9790), en la que se explica la interacción de dicho ajuste con el resto de datos.

El valor de la coordenada del punto de referencia se encuentra dentro de la zona de trabajo:

- **Eje lineal**
  Incluido el final de carrera por software.
- **Eje rotativo**
  Mayor o igual que 0 y menor que el valor "Final del eje rotativo" (0 ≤ coordenada del punto de referencia < final del eje rotativo).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1.000.000.000 µm a +1.000.000.000 µm

Definir reajuste del encóder absoluto

Calcular reajuste del encóder absoluto

Definición

El reajuste del encóder absoluto sólo es necesario para un encóder absoluto (SSI).

Con el reajuste del encóder absoluto y las coordenadas del punto de referencia el rango de valores del sensor se reproduce claramente en el sistema de coordenadas del eje. Se calcula el valor del encóder que equivale a la coordenada del punto de referencia en el eje. El valor tiene que ser menor que el total de pasos del encóder absoluto.

Encontrará más información al respecto en "Procedimiento de reajuste del encóder absoluto (Página 9791)", así como en un ejemplo (Página 9792).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
48.0	ENC_ADJ	DINT	L#0	**Reajuste del encóder absoluto** Rango de valores admisible: • 0 a (2^{25} - 1)

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
98.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1.000.000.000 µm a +1.000.000.000 µm

Procedimiento para un reajuste del encóder absoluto

Determinar el reajuste correcto del encóder absoluto

Tras la primera parametrización hay que llevar a cabo otros pasos para que se establezca una relación correcta entre el encóder y el sistema de coordenadas. El proceso se representa utilizando el cuadro de diálogo de parametrización.

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Fin del final de carrera por software".
2. Llame el ajuste "Definir punto de referencia (Página 9762)" con la coordenada del punto definido en 1.
   El FM x52 determina entonces un valor de encóder para la coordenada del punto de referencia registrada en el eje en el DB de canal (REFPT): el reajuste del encóder absoluto. Este valor se puede leer con el ajuste "Datos de encóder (Página 9778)".
3. Registre el valor leído en el campo "Reajuste del encóder absoluto".
4. Guarde la parametrización en el DB de parámetros correspondiente con la función de exportación.
5. Cargue los datos de HWCN en la CPU.
6. Para activar los datos, rearranque la CPU.

Nota

Esta sincronización se realiza una vez en la puesta en marcha. Tras una parametrización, el FM x52 queda sincronizado durante el arranque en cuanto se recibe un telegrama sin errores del encóder.

Alternativa: Reajuste mecánico de un encóder

También se puede obtener una relación correcta entre el sistema de coordenadas y el encóder del modo siguiente:

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Inicio del final de carrera por software".
2. Introduzca este valor de coordenadas en los datos máquina a modo de "coordenada del punto de referencia".
3. Lea el valor de encóder que se muestra en esta posición con el ajuste "Datos de encóder (Página 9778)".
4. Introduzca este valor como reajuste del encóder absoluto en los datos máquina.

Tras la parametrización se mostrará siempre un valor real correcto.

En lugar del paso 3 y 4 también es posible resetear a cero el codificador a través de la opción "Reset", si está disponible. A continuación, introduzca el valor "0" como reajuste del encóder absoluto en los datos máquina.

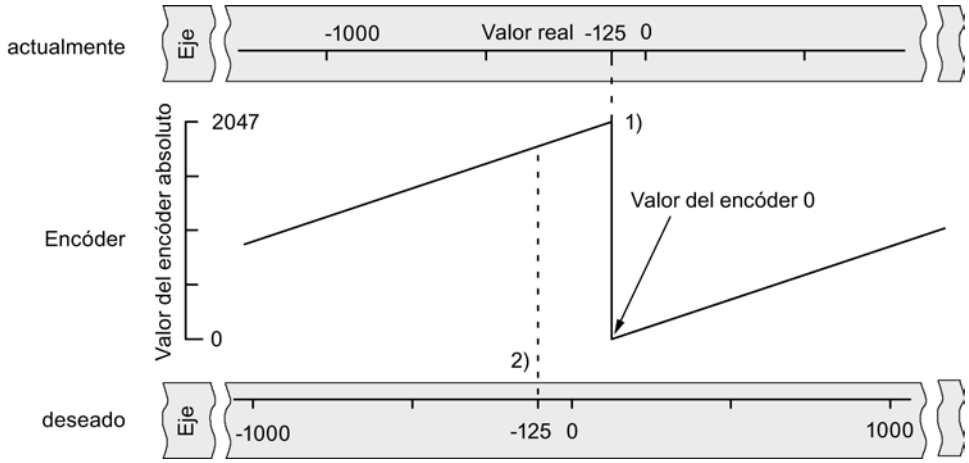
Ejemplo para un reajuste del encóder absoluto

Ejemplo para un reajuste del encóder absoluto

Para este ejemplo se parte de las premisas siguientes:

- Coordenada del punto de referencia = -125 mm
- Área de trabajo de SSW_STRT = -1.000 mm a SSW_END = 1.000 mm
- Reajuste del encóder absoluto = 0
- Área del encóder = 2.048 incrementos con una resolución de 1 mm/impulso
- El encóder absoluto utilizado no permite un ajuste mecánico exacto y tampoco ofrece la posibilidad de definir el valor del encóder.

La figura siguiente muestra un ejemplo de reajuste del encóder absoluto.

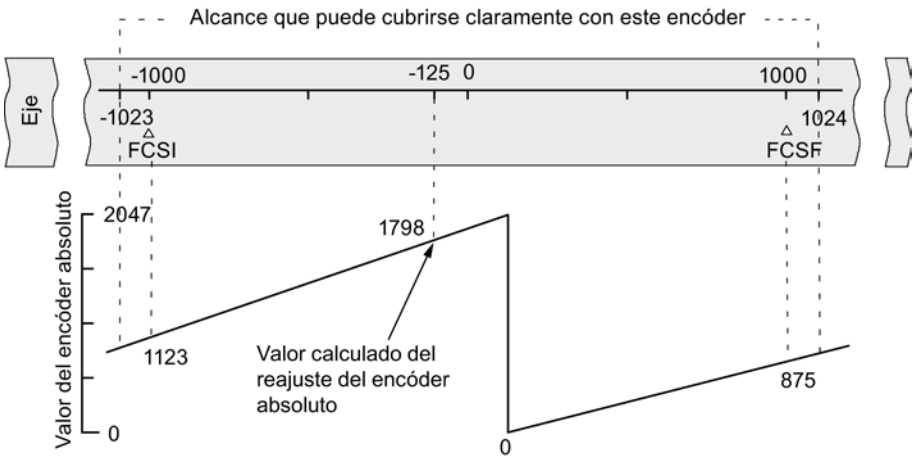


1) Asignación del sistema de coordenadas a los valores del sensor con el reajuste de encóder absoluto definido. El valor del encóder 0 equivale al valor real -125.
2) Asignación deseada del sistema de coordenadas con respecto al encóder. En esta posición, la coordenada tiene que ser -125.

Resultado tras "Definir el punto de referencia"

Tras "Definir el punto de referencia", la relación es la siguiente:

La coordenada del punto de referencia en el eje (-125) se asigna al valor del encóder calculado (1798) a partir del reajuste del encóder absoluto. La figura siguiente muestra esta asignación.



El encóder devuelve 2.048 valores unívocos. La zona de trabajo está definida por el final de carrera por software. Sin embargo, debido a la resolución seleccionada de 1 mm por impulso, el encóder puede abarcar una zona de trabajo mayor que la prevista por el final de carrera por software.

Con la resolución definida, la zona de trabajo está ya cubierta con 2.001 valores. Por ello en el ejemplo quedan 47 impulsos "sobrantes" dispuestos simétricamente alrededor de la zona de trabajo.

Definir histéresis

Histéresis

El rango de valores de la histéresis (Página 9740) depende de la resolución (Página 9799).

El valor máximo de entrada es como sigue:

- En el caso de ejes lineales: Valor de entrada máximo < ¼ de la zona de trabajo
- En el caso de los ejes rotativos: Valor de entrada máximo < ¼ del área del eje rotativo

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
80.0	HYS	DINT	L#0	**Histéresis** Rango de valores admisible: • 0 … 65.535 [Imp] × resolución [$^{\mu m}/_{Imp}$]

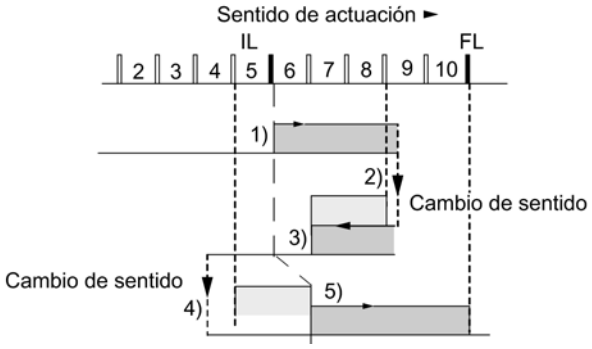
Leva de recorrido con histéresis

Una leva de recorrido se acciona cuando se cumplen las siguientes condiciones:

- el valor real detectado se encuentra dentro de la leva de recorrido
- la histéresis no está activada.

Según la posición en la que tiene lugar el cambio de sentido, puede variar la posición de un punto de cambio.

La siguiente figura muestra el comportamiento de conmutación de una leva de recorrido en el caso de una histéresis.



	Leva
	Histéresis
1)	La leva se activa porque el principio de leva se rebasa en sentido de actuación positivo.
2)	A pesar del cambio de sentido, la leva permanece activada debido a la histéresis.
3)	La leva se desactiva al final de la histéresis.
4)	La leva permanece desactivada tras el cambio de sentido.
5)	Tras el final de la histéresis la leva se activa nuevamente.

Nota

Las levas de recorrido que son más cortas que la histéresis no pueden ser acortadas por la histéresis en caso de un cambio de sentido.

Levas de tiempo con histéresis

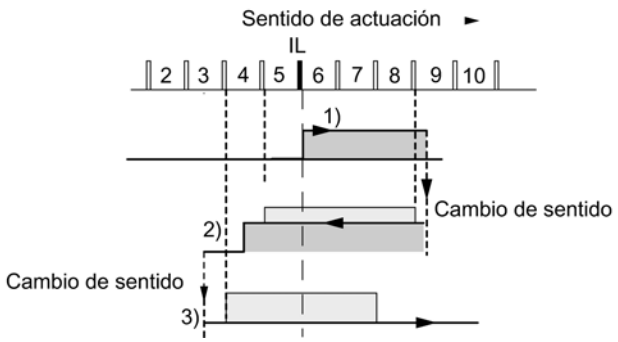
La leva de tiempo se acciona cuando se cumplen las siguientes condiciones:

- El principio de leva se rebasa en el sentido de actuación.
- la histéresis no está activada.

Nota

Si el área comprendida entre el punto de inversión y el principio de leva de tiempo es menor que la histéresis la leva de tiempo quedará oculta por la histéresis.

La siguiente figura muestra el comportamiento de conmutación de una leva de tiepo en el caso de una histéresis. Se representa una leva de tiempo que **no** se activa de nuevo.



	Leva
	Histéresis
1)	La leva se activa porque el principio de leva se rebasa en sentido de actuación positivo.
2)	Tras el cambio de sentido y hasta que ha transcurrido el tiempo de conexión la leva se mantiene activada, independientemente de la histéresis.
3)	El principio de leva queda oculto por la histéresis, la leva no se activa.

Definir velocidad de simulación

Velocidad de simulación

Este dato máquina define la velocidad de la "simulación". Tenga en cuenta a este respecto la información incluida en "Influencia de los ajustes en el comportamiento de conmutación de las levas de tiempo (Página 9757)".

La velocidad de simulación real V_{sim} puede diferir de la velocidad de simulación introducida $V_{sim, V}$ y se calcula conforme a la fórmula siguiente:

$$V_{Sim} = \frac{6*10exp7*\ RES}{\text{Entero}\left(\frac{6*10exp7*\ RES}{V_{Sim,\ V}}\right)}$$

En esta fórmula:

- V_{sim}: es la velocidad de simulación definida por el FM 352. Unidad: $^{\mu m}/_{min}$
- $V_{sim, V}$: es la velocidad de simulación que el usuario define en los datos máquina. Unidad: $^{\mu m}/_{min}$
- RES: Resolución calculada a partir de los datos del encóder. Unidad: $^{\mu m}/_{Impulso}$
- Entero ( ): a partir de esta expresión, sólo se tiene en cuenta el valor previo al decimal para otros cálculos. Esta expresión tiene que encontrarse dentro del rango de 2 ... 65.536 en todos los cálculos.

Las diferentes relaciones (véase la fórmula anterior) provocan cambios inesperados en la velocidad de simulación real.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
84.0	SIM_SPD	DINT	L#0	**Velocidad de simulación** Rango de valores admisible: • 0 = Parada • 5×10^8 = Ajuste máximo del módulo Dentro de este rango la velocidad de simulación depende de la resolución (Página 9799): 1000 × resolución ≤ velocidad de simulación ≤ 3×10^7 × resolución

Definir distancia mínima entre bordes

Distancia mínima entre bordes

En una captura de bordes, con este dato máquina se define un área tras detectar el inicio de la medición. Si el final del proceso de medición se encuentra dentro de este rango, la medición se desestima.

El inicio de la medición se notifica en cuanto se ha recorrido la distancia mínima entre bordes.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
4.0	EDGEDIST	DINT	L#0	**Distancia mínima entre bordes** Rango de valores admisible: • 0 ... 1.000.000.000 µm

Encóder

Seleccionar tipo de señal

Tipo de señal

El encóder da información de recorrido al módulo, el cual la evalúa y la convierte en un valor real con la resolución.

Es posible seleccionar un tipo de señal de uno de los encóders (Página 9744) siguientes:

- Señal diferencial de 5 V (Página 9745)
  Seleccione el tipo de señal "5 V Señal diferencial" si utiliza un encóder incremental de 5 V con señales diferenciales A, /A, B, /B y N, /N.
- 24 V Pista A+B desfasado (Página 9745)
  Seleccione el tipo de señal "24 V Pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A*, B* y N*.
- SSI absoluto (Página 9747)
  Seleccione el tipo de señal "SSI absoluto" si desea utilizar un encóder absoluto con una interfaz en serie.
- 24 V Detector de proximidad (Página 9747)
  Seleccione el tipo de señal "Detector de proximidad de 24 V" si desea utilizar un detector de 24V con la señal A*.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
20.0	ENC_TYPE	DINT	L#1	**Tipo de señal** Valores permitidos: • 1 = 5 V incremental • 2 = 24 V incremental • 3 = SSI longitud del telegrama 13 bits • 4 = SSI longitud del telegrama 25 bits • 5 = Escuchar • 6 = 24 V Detector de proximidad ascendente (positivo) • 7 = 24 V Detector de proximidad descendente (negativo) • 8 = SSI longitud del telegrama 13 bits (alineación derecha) • 9 = SSI longitud del telegrama 25 bits (alineación derecha) • 10 = Escuchar (alineación derecha)

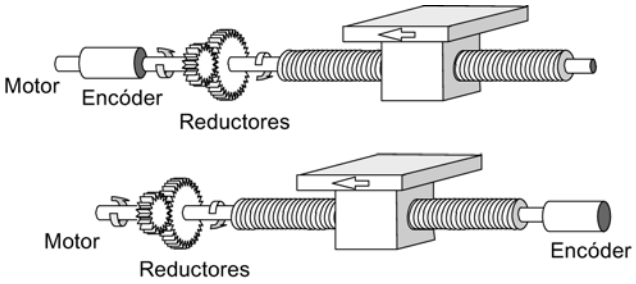
Definir resolución

Definir el recorrido por vuelta del encóder

recorrido por vuelta del encóder

Con el dato máquina "Recorrido por vuelta del encóder" se indica al FM x52 qué recorrido realiza el sistema de accionamiento en cada vuelta.

El valor "Recorrido por vuelta del encóder" depende de la estructura del eje y del modo en que está colocado el encóder. Debe tener en cuenta todos los elementos de transmisión, como acoplamientos o engranajes. La figura siguiente muestra dos disposiciones de encóder distintas a modo de ejemplo.



La relación entre los datos máquina "Recorrido por vuelta del encóder" e "Incrementos por vuelta del encóder" se describe en el apartado Resolución (Página 9799).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
24.0	DISP_REV	DINT	L#80000	**Recorrido por vuelta del encóder** Rango de valores admisible: • 1 µm a 1.000.000.000 µm

Definir los incrementos por vuelta del encóder

Incrementos por vuelta del encóder

El dato máquina "Incrementos por vuelta del encóder" indica la cantidad de incrementos que proporciona un encóder en cada vuelta. A partir de este valor y del dato máquina "Recorrido por vuelta del encóder", el FM x52 calcula la resolución.

Encóder incremental

Como entrada (Página 9745) es posible indicar cualquier valor dentro del rango de valores. El módulo evalúa los incrementos en múltiplos de 4.

Detector de proximidad

Es posible indicar cualquier valor dentro del rango de valores.

Encóder absoluto

Los límites varían según el tipo de encóder:

Tipo de encóder	Longitud/tipo de telegrama	Valores posibles	utilizable como eje lineal
Encóder monovuelta	13 bits medio abeto	64 ... 8192 en potencias de 2	
Encóder monovuelta	13 bits alineación derecha	64 ... 8192 todos los valores	X
Encóder monovuelta	25 bits alineación derecha	64 ... 2^{25} todos los valores	X
Encóder multivuelta	25 bits abeto	64 ... 8192 en potencias de 2	

Tipo de encóder	Longitud/tipo de telegrama	Valores posibles	utilizable como eje lineal
Encóder multivuelta	25 bits alineación derecha	64 ... 2^{24} todos los valores	
Escuchar	Abeto	64 ... 8192 en potencias de 2	
Escuchar	alineación derecha	64 ... 2^{25} todos los valores	X
Ajuste especial: Encóder multivuelta como encóder monovuelta	25 bits medio abeto	64 ... 8192 en potencias de 2	

Nota

La cantidad de impulsos de un encóder se calcula a partir de los dos datos "Incrementos por vuelta del encóder" multiplicado por "Número de vueltas" (véase el apartado "Resolución (Página 9799)").

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
32.0	INC_REV	DINT	L#500	**Incrementos por vuelta del encóder** Rango de valores admisible: • 1 a 2^{25} Atención: En el sistema de medida "Impulsos" esta entrada carece de importancia.

Resolución

Resolución

La resolución es una medida para la precisión del procesamiento de levas. Determina también la zona de desplazamiento máxima posible.

Para calcular la resolución (RES) se procede tal como se indica en la tabla siguiente:

	Encóder incremental	Encóder absoluto/detector de proximidad
Valores de entrada	• Recorrido por vuelta • Incrementos por vuelta del encóder: – Evaluación de impulsos: Cuádruple – 1 Incremento = 4 Impulsos	• Recorrido por vuelta del encóder • Incrementos por vuelta del encóder. – 1 Incremento = 1 Impulso
Cálculo	$RES = \frac{\frac{\text{Trayecto}}{\text{Vuelta del encóder}}}{\frac{\text{Impulsos}}{\text{Vuelta del encóder}}}$	

Nota

En el sistema de medida "Impulsos" la resolución siempre tiene el valor 1.

Todos los datos de posición se redondean al múltiplo entero de la resolución. De este modo es posible diferenciar entre los valores introducidos y los utilizados.

Rango de valores de la resolución

Es preciso convertir el rango de valores admitidos de la resolución al sistema de medida subyacente. Con la elección de los dos valores "Recorrido por vuelta" e "Incrementos por vuelta" la resolución se define dentro de este rango de valores.

El sistema de medida seleccionado determina el margen de valores de la resolución:

Sistema de medición	Datos en ...	Rango de valores de la resolución
mm	10^{-3} mm	$0,1 \times 10^{-3}$ mm .... 1000×10^{-3} mm/impulso
pulgadas	10^{-4} pulgadas	$0,1 \times 10^{-4}$ pulgadas .... 1000×10^{-4} pulgadas/impulso
grados	10^{-4} grados 10^{-3} grados 10^{-2} grados	$0,1 \times 10^{-4}$ grados .... 1000×10^{-4} grados/impulso $0,1 \times 10^{-3}$ grados .... 1000×10^{-3} grados/impulso $0,1 \times 10^{-2}$ grados .... 1000×10^{-2} grados/impulso
Impulsos	1 impulso	1

Ejemplo

- Un encóder incremental tiene los datos siguientes:
  - Incrementos por vuelta: 5000
  - Recorrido por vuelta del encóder: 1000 mm
  - 1 incremento = 4 impulsos

  De ello resulta la resolución (evaluación cuádruple):

$$\text{Resolución} = \frac{1000\ \text{mm}}{5.000\ \text{incrementos}} = 0{,}2000\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{impulsos}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impulso}}$$

- Un encóder SSI tiene los datos siguientes:
  - Incrementos por vuelta: 4096
  - Recorrido por vuelta del encóder: 1000 mm
  - 1 incremento = 1 impulso

  De ello resulta la resolución:

$$\text{Resolución} = \frac{1000\ \text{mm}}{4096\ \text{incrementos}} = 0{,}2441\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2441\ \frac{\text{mm}}{\text{Impulso}}$$

Dependencia entre la zona de desplazamiento y la resolución

La zona de desplazamiento está limitada por la representación numérica en el FM x52. Esta representación numérica varía en función de la resolución. Por ello, al definir valores hay que procurar mantenerse siempre dentro de los límites admitidos.

La siguiente tabla muestra la zona de desplazamiento máxima:

Resolución (RES) dentro de la zona	Zona de desplazamiento máxima
0,1 µm/impulso ≦ RES < 1 µm/impulso	-10^8 µm a 10^8 µm (-100 m a +100 m)
1 µm/impulso ≦RES ≦ 1000 µm/impulso	-10^9 µm a 10^9 µm (-1000 m a +1000 m)

Dependencia de la velocidad de la resolución

La velocidad que se muestra puede oscilar dentro de los márgenes que se indican según sea la resolución (los datos se expresan en el sistema de medida mm):

- de 1 µm/min a 90 m/min a una resolución < 1 µm/impulso
- de 1 µm/min a 900 m/min a una resolución ≥ 1 µm/impulso

La velocidad es calculada y suavizada por el módulo cada 4 ms.

Tiene por lo menos una imprecisión de un impulso/4 ms y no resulta adecuada para fines de regulación.

Seleccionar sentido de contaje

Dirección de contaje

Con el dato máquina "Dirección de contaje" se adapta la dirección de la captura del recorrido al sentido del movimiento del eje. Para ello, tenga en cuenta todos los sentidos de giro de los elementos de transmisión (como p. ej., acoplamientos y engranajes).

- normal = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real ascendentes
- invertida = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real descendentes

Nota:
No se permite un tiempo de anticipación en combinación con un encóder absoluto (SSI) y con sentido de contaje invertido.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	Valor inicial	Comentario
59.0	CNT_DIR	BOOL	FALSE	**Sentido de contaje** Valores permitidos: • 0 = normal • 1 = invertida

Seleccionar vigilancia

Vigilancia

Rotura de hilo

Con la activación de la vigilancia, el FM x52 vigila, en el caso de un encóder incremental, las señales A, /A, B, /B, N y /N. La vigilancia detecta:

- Rotura de hilo
- Cortocircuito de los distintos cables
  En los encóders incrementales sin marcas cero, como alternativa tiene que
  - desconectar la vigilancia de rotura de hilo
  - interconectar externamente las señales N y /N (Página 9744)
- Distancia de flancos de los impulsos de contaje
- Corte de la alimentación del encóder

Error del telegrama

La monitorización de errores de telegrama en los encóders absolutos (SSI) no puede desactivarse. Supervisa:

- el telegrama para ver si hay errores del bit de inicio y del bit de parada
- el tiempo monoestable del encóder conectado

Impulsos erróneos (encóder incremental)

Entre dos marcas cero consecutivas, un encóder incremental siempre tiene que proporcionar el mismo número de incrementos.

El FM x52 comprueba si la marca cero de un encóder incremental aparece en el estado adecuado del encóder.

En el caso de encóders sin marca cero es preciso desactivar la vigilancia de los impulsos de error.

Igualmente es preciso desactivar la vigilancia de rotura de hilo o bien interconectar las entradas de marca cero N y /N de forma externa.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	Valor inicial	Comentarios
				Vigilancias
63.0	MON_WIRE	BOOL	TRUE	1 = Rotura de hilo
63.1	MON_FRAME	BOOL	TRUE	1 = Error de telegrama (tiene que ser siempre 1)
63.2	MON_PULSE	BOOL	TRUE	1 = Impulsos de error

Seleccionar datos para el encóder absoluto (SSI)

Número de vueltas

Número de vueltas del encóder

El dato máquina "Número de vueltas del encóder" sólo existe en encóders absolutos. Con él se indica el número de vueltas posibles con este encóder. Encontrará más información en el apartado "Encóder absoluto (SSI) (Página 9747)".

- **Encóder monovuelta**
  Sólo es posible el valor 1.
- **Encóder multivuelta**
  - Encóder Multivuelta / Escuchar (abeto): 2 ... 4096 en potencias de 2
  - Encóder multivuelta / Escuchar (alineación derecha): 1 ... 4096 todos los valores
- **Escala lineal**
  También es posible conectar una escala lineal. Introduzca para ello el valor 1.

El **total de pasos del encóder** no es un dato máquina. Se calcula del modo siguiente:

Total de pasos = Incremento por vuelta x Número de vueltas

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentario
36.0	NO_REV	DINT	L#1024	**Número de vueltas del encóder** Rango de valores admisible: • 1 (encóder monovuelta) • 1 a 4096 (encóder multivuelta)

Velocidad de transferencia

Velocidad de transferencia

Con el dato máquina "Velocidad de transferencia" se define la velocidad de la transferencia de datos del encóder SSI al FM x52.

La velocidad de transferencia máxima depende de la velocidad de transferencia:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 63 m → 500 kHz
- 20 m → 1000 kHz

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
40.0	BAUDRATE	DINT	L#0	**Velocidad de transferencia** Valores permitidos: • 0 = 125 kHz • 1 = 250 kHz • 2 = 500 kHz • 3 = 1000 kHz

Longitud de telegrama

Longitud de telegrama

Con la "longitud de telegrama" se define el marco de impulsos que emite el FM x52.

Seleccionando "Escuchar", se desactiva el reloj del FM x52. Entonces el FM x52 puede escuchar también otros telegramas SSI con un marco de telegramas de 13 ó 25 bits. La velocidad de la transferencia se basa en la velocidad de reloj del módulo maestro.

Datos utilizados en el DB de parámetros

La longitud de telegrama seleccionada aquí se dispone en el DB de parámetro con el dato ENC_TYPE (Página 9793).

Sentido del detector de proximidad

Sentido del detector de proximidad

Con el sentido del detector es posible determinar el sentido de un detector de proximidad conectado:

- Positivo: corresponde al sentido de avance (impulsos de contaje ascendentes)
- Negativo: corresponde al sentido de retroceso (impulsos de contaje descendentes)

Pistas

Seleccionar el modo de control de las salidas de pista

Control de las salidas de pista

Con el dato máquina "Control de las salidas de pista" se define el modo de control de las salidas de pista de las pistas 0 a 12. El control es es posible mediante:

- Leva electrónica: Las señales de pista son conectadas y desconectadas por el procesamiento de levas del FM 352.
- CPU: Las señales de pista reproducen directamente los valores correspondientes de las habilitaciones de pistas en el DB de canal.

De este modo las salidas de pista se pueden controlar directamente desde su programa.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
90.0	TRACK_OUT	WORD	W#16#0	**Control de las salidas de pista** Valores permitidos: • 0 = Leva electrónica • 1 = CPU Número de bit = Número de pista Los bits entre 13 y 15 tienen que ser 0.

Seleccionar pistas especiales

Pistas especiales

Es posible parametrizar las pistas 0, 1 y 2 como pistas especiales.

- Pista 0 y pista 1 como pista de leva de contaje (Página 9739)
- Pista 2 como pista de leva de freno (Página 9739)

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
				Pistas especiales
99.0	SPEC_TRC0	BOOL	FALSE	1 = Pista 0 es pista de leva de contaje
99.1	SPEC_TRC1	BOOL	FALSE	1 = Pista 1 es pista de leva de contaje
99.2	SPEC_TRC2	BOOL	FALSE	1 = Pista 2 es pista de leva de freno

Definir valor de contaje superior para la pista de leva de contaje

Valor de contaje superior pista de leva de contaje

Con este dato máquina se define el valor de contaje superior para la pista de leva de contaje parametrizada.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CNT_LIM0	DINT	L#2	**Valor de contaje superior para pista de leva de contaje**(Pista 0)
104.0	CNT_LIM1	DINT	L#2	**Valor de contaje superior para pista de leva de contaje**(Pista 1) Rango de valores admisible: • 2 … 65535

Seleccionar entrada de habilitación

Entrada de habilitación

La señal de pista Q3 se activa cuando se cumplen todas las condiciones siguientes:

- La pista está habilitada.
- La entrada de habilitación externa correspondiente I3 está activada.
- El resultado de la pista es 1.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
95.0	EN_IN_I3	BOOL	FALSE	**Entrada de habilitación** 1 = La señal de pista Pista 3 está combinada con la operación lógica AND con la entrada de habilitación I3 Los bits entre 95.1 y 95.7 tienen que ser 0.

Leva

Alcance

Alcance

El alcance determina el número máximo de levas parametrizables y, con ello, el tiempo de ciclo de la leva.

Alcance	Tiempo de ciclo de la leva
16 levas	20,48 µs
32 levas	40,96 µs
64 levas	81,92 µs
128 levas	163,84 µs

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
76.0	C_QTY	DINT	UDT3: L#0 UDT4: L#1 UDT5: L#2 UDT6: L#3	**Alcance** 0 = máx. 16 levas 1 = máx. 32 levas 2 = máx. 64 levas 3 = máx. 128 levas

Datos de leva

Datos de leva

Datos de leva

Los datos de leva describen las propiedades de una leva, la asignación de cada una de ellas a una pista y el comportamiento de conmutación de la leva.

Los datos de leva que se listan a continuación se ajustan por separado para cada leva.

- Validez (Página 9808)
- Número de pista (Página 9808)
- Tipo de leva (Página 9809)
- Principio y final de la leva para levas de recorrido (Página 9809)
- Inicio de leva y tiempo de conexión para levas de tiempo (Página 9810)
- Tiempo de anticipación (Página 9811)
- Sentido de actuación (Página 9813)
- Alarma de proceso (Página 9815)

Por lo general, para el procesamiento de levas se aplica lo siguiente:

- Sólo las levas ajustadas como "válidas" son interpretadas y procesadas por el módulo.
- Las levas 0 ... 7 soportan alarmas de proceso
- La cantidad de levas parametrizables depende del alcance.

Validez de una leva

Leva "válida"

Es preciso activar como válidas todas las levas que deben ser procesadas por la leva electrónica.

Nota

Configure siempre las levas no utilizadas como "no válidas" (CAMVALID = FALSE).

Datos utilizados en el DB de parámetros

Dirección relativa	Nombre	Tipo	valor inicial	Comentarios
+0.0	CAMVALID	BOOL	FALSE	1 = Leva "válida"

Definir número de pista para una leva

Número de pista

Al escoger el número de pista se define para cada leva la pista en la que dicha leva tiene que tener efecto.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+1.0	TRACK_NO	BYTE	B#16#0	**Número de pista** Rango de valores admisible: • 0 a 31

Seleccionar tipo de leva

Tipo de leva

Seleccione el tipo de leva deseado.

- Leva de recorrido
- Leva de tiempo

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+0.3	CAM_TYPE	BOOL	FALSE	**Tipo de leva** Valores permitidos: • 0 = Leva de recorrido • 1 = Leva de recorrido

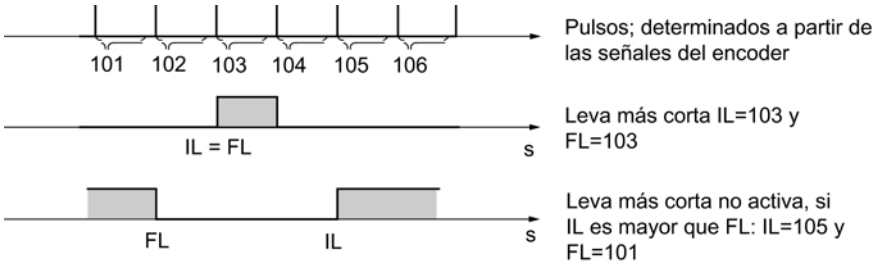
Definir inicio y final de la leva para levas de recorrido

Inicio de leva (IL) / Fin de leva (FL) en levas de recorrido

El inicio y el final de leva definen la longitud de leva en una leva de recorrido.

Longitud mínima de una leva de recorrido

La siguiente figura muestra la leva más corta con sentido de movimiento positivo del eje.



Por regla general, rige lo siguiente:

- La parte no activa de una leva siempre tiene que mantener una distancia mínima de 4 impulsos entre el fin de leva (FL) y el inicio de leva (IL).
- En el caso de FL = IL la leva se activa por un impulso.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios para levas de recorrido
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Inicio de leva (IL)** **Fin de leva (FL)** Rango de valores admisible: • -1.000.000.000 µm a 1 000 000 000 µm

Definir principio de leva y tiempo de conexión para levas de tiempo

Inicio de leva (IL) / Final de leva (FL) ≙ Tiempo de conexión en levas de tiempo

En el caso de una leva de tiempo es preciso definir un inicio de leva y, en lugar del fin de leva, un tiempo de conexión. El tiempo de conexión se define en pasos de 100 µs. El tiempo empieza a transcurrir a partir de la conexión de la leva.

Para la definición de estos tiempos rigen las condiciones siguientes:

- 0 µs: Una leva con el tiempo de conexión 0 µs no se activa nunca
- 0 µs < t ≤ 400 µs: El FM x52 ajusta un tiempo de leva mínimo de unos 330 µs.
- t > 400 µs: El FM x52 calcula el tiempo de conexión real t_{real} a partir del tiempo de conexión especificado t_{def} según la fórmula siguiente:

$$t_{real} = \text{número entero} \left(\frac{t_{def}}{\text{Tiempo de ciclo de la leva}} \right) * \text{Tiempo de ciclo de la leva}$$

El error máximo siempre es menor que el tiempo de ciclo de la leva.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios para levas de tiempo
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Inicio de leva (IL)** **Final de leva (FL) ≙ Tiempo de conexión** Rango admisible de valores para inicio de leva: • -1.000.000.000 µm a 1 000 000 000 µm Rango admisible de valores para tiempo de conexión: • (0 a 13421) × 100 µs con un máximo de 16 levas • (0 a 26843) × 100 µs con un máximo de 32 levas • (0 a 53686) × 100 µs con un máximo de 64 levas • (0 a 65535) × 100 µs con un máximo de 128 levas

Definir tiempo de anticipación

Tiempo de anticipación

Los retardos de los elementos de maniobra conectados se pueden compensar definiendo un tiempo de anticipación. El tiempo de anticipación se define en pasos de 100 µs. Es posible conceder un tiempo de anticipación por leva. El tiempo de anticipación es válido para el inicio y el final de la leva.

Recorrido de anticipación

El recorrido de anticipación de una leva se calcula continuamente en función de la velocidad actual y del tiempo de anticipación. Toda la leva se desplaza ese recorrido en la dirección del valor real. El área parametrizada se denomina "área estática" y el área calculada atendiendo al tiempo de anticipación se denomina "área dinámica".

Recorrido de anticipación = tiempo de anticipación × velocidad actual

El cálculo de los recorridos de anticipación de todas las levas se realiza dentro de ¼ del tiempo de anticipación más largo parametrizado en el FM x52. Si parametriza un tiempo de anticipación muy prolongado para una leva, el ajuste dinámico se calcula con menos frecuencia.

Tiempo de anticipación real

El tiempo de anticipación real se calcula del modo siguiente:

1. Calcule el tiempo de ciclo de la leva: Es el tiempo en el que el FM x52 ha procesado por completo todas las levas. Depende del alcance parametrizado.
2. Calcule el tiempo de anticipación real con la fórmula siguiente:

$$\text{Tiempo de disponibilidad}_{real} = \text{número entero}\left(\frac{\text{Tiempo de disponibilidad}_{D}}{\text{Tiempo de ciclo de la leva} * 4}\right) * \text{Tiempo de ciclo de la leva} * 4$$

   Significados:
   - Tiempo de anticipación$_{real}$es el tiempo de anticipación ajustado por el FM x52.
   - Tiempo de anticipación $_D$ es el tiempo de anticipación preseleccionado.
   - Número entero ( ) significa que para el cálculo de lo comprendido entre paréntesis sólo se tiene en cuenta el valor anterior al decimal.

El error máximo del tiempo de anticipación$_{real}$ siempre es menor que el tiempo de ciclo de la leva × 4.

Ejemplo:

Se especifican los valores siguientes:

- Alcance: máximo 32 levas
- Tiempo de ciclo de la leva: 40,96 µs
- Tiempo de anticipación$_D$ = 1000 µs

Resultado: Se obtiene un tiempo de anticipación real de 983 µs.

Nota

El tiempo de anticipación real siempre es menor que el tiempo de anticipación parametrizado. Puede ser 0 aunque el tiempo de anticipación parametrizado sea ≥100 µs.

El recorrido de anticipación en un eje rotativo tiene que ser menor que el área del eje rotativo y la parte no activa de la leva. Esto tiene que estar garantizado para todas las velocidades.

Nota

No se soporta un tiempo de anticipación en relación con un encóder absoluto (SSI) ni con una dirección de contaje parametrizada como invertida.

Datos utilizados en el DB de parámetros

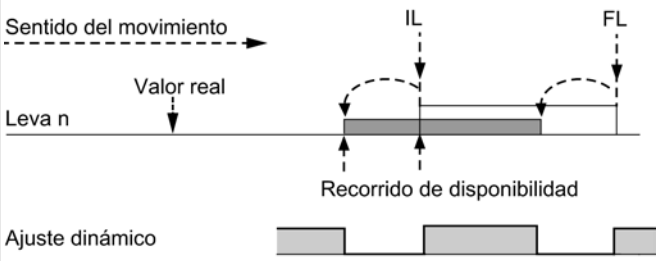
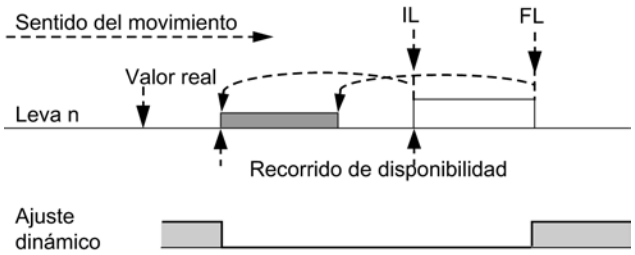
Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+ 10.0	LTIME	INT	0	**Tiempo de anticipación** Rango de valores admisible: • (0 a 53686) × 100 µs con un máx. de 16 levas • (0 a 65535) × 100 µs con un máx. 32, 64 o 128 levas

Ajuste dinámico de una leva

Para el rango de la leva se distinguen dos casos:

1. El rango estático y el rango dinámico de la leva están superpuestos.
2. El rango estático y el rango dinámico de la leva no están superpuestos.

La tabla siguiente muestra para ambos casos el cálculo del ajuste dinámico.

Ajuste dinámico	Descripción
	Cuando el rango dinámico se superpone al rango estático de la leva, rige lo siguiente: • Cuando se alcanza el área dinámica de la leva, ésta se activa. Simultáneamente se bloquea el cálculo de un nuevo ajuste dinámico. • Si el valor real alcanza el área estática de la leva, se vuelve a habilitar el cálculo de un ajuste dinámico nuevo y un cambio de velocidad actúa al final de la leva. • Si la leva se desactiva al final del área dinámica, el ajuste dinámico vuelve a estar bloqueado hasta el final del área estática de la leva.
	Cuando el área dinámica no se superpone al área estática de la leva, rige lo siguiente: • Cuando se alcanza el rango dinámico de la leva, ésta se activa. Simultáneamente se bloquea el cálculo de un nuevo ajuste dinámico. • Al final del área estática de la leva se vuelve a habilitar el ajuste dinámico.
Área dinámica Área estática Es posible un **nuevo** ajuste dinámico	

Nota

Modificando la dirección de giro se vuelve a habilitar la determinación del ajuste dinámico.

Selección sentido de actuación

Sentido de actuación

Existen dos sentidos de actuación posibles:

- Positivo: la leva se activa al inicio, cuando el eje se desplaza en sentido ascendente respecto al valor real.
- Negativo: la leva se activa al final, cuando el eje se desplaza en sentido descendente respecto al valor real.

Es posible también ajustar simultáneamente ambos sentidos de actuación.

Si se selecciona "Ninguno" la leva queda sin efecto.

Comportamiento de conmutación de la leva dependiendo del sentido de actuación

La siguiente tabla muestra el comportamiento de actuación de las levas de recorrido y de tiempo dependiendo del sentido de actuación. Excepto en el ejemplo 5 se parte siempre de un sentido de actuación positivo.

Nº	Descripción	Leva de recorrido	Leva de tiempo
1	Una leva se rebasa en el sentido de actuación	IL FL	IL Tiempo de activación
2	Una leva se rebasa en el sentido contrario al de actuación	IL FL	IL Tiempo de activación
3	Una leva se desplaza en el sentido de actuación; mientras está activada, cambia el sentido del movimiento del eje	IL FL	IL Tiempo de activación x t1 t2 Tiempo de activación x = t1 + t2
4	Una leva se desplaza en sentido opuesto al de actuación; en la leva el sentido de movimiento del eje cambia al sentido de actuación	IL FL	La leva no se activa
5	Una leva se desplaza y se abandona en un sentido cualquiera; el sentido de actuación está ajustado para **ambas direcciones**	IL FL	IL Tiempo de activación x t1 t2 t3 Tiempo de activación x = t1 + t2 + t3

Leva parametrizada
Leva activada

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+0.1	EFFDIR_P	BOOL	TRUE	1 = Sentido de actuación positivo (+)
+0.2	EFFDIR_M	BOOL	TRUE	1 = Sentido de actuación negativo (-)

Seleccionar alarma de proceso para levas

Alarma de proceso al activar/desactivar levas

Para las levas 0 a 7 se puede definir si, al conectar, desconectar o al conectar y desconectar la leva, debe generarse una alarma de proceso.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+0.4	PI_SW_ON	BOOL	FALSE	1 = Alarma de proceso al activar
+0.5	PI_SW_OFF	BOOL	FALSE	1 = Alarma de proceso al desactivar

Habilitar alarma de proceso

Habilitar las alarmas de proceso

Habilitar las alarmas de proceso

Este dato máquina permite habilitar alarmas de proceso.

Es posible ajustar cuándo deben generarse las alarmas de proceso:

- Leva activa / inactiva
  En los datos de leva (Página 9812) es posible establecer si se tienen que generar alarmas de proceso al activar y/o desactivar las levas 0 a 7.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
3.2	PI_CAM	BOOL	FALSE	1 = Habilitar la alarma de proceso: Leva activa / inactiva

Evaluación de una alarma de proceso

Cuando el FM 352 dispara una alarma de proceso, en las variables OB4x_POINT_ADDR del OB de alarma de procesos se encuentra disponible la información sobre alarmas.

La tabla siguiente muestra el formato del tipo de datos OB4x_POINT_ADDR:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	Leva	0	0
2	Leva 7 activa	Leva 7 inactiva	Leva 6 activa	Leva 6 inactiva	Leva 5 activa	Leva 5 inactiva	Leva 4 activa	Leva 4 inactiva
3	Leva 3 activa	Leva 3 inactiva	Leva 2 activa	Leva 2 inactiva	Leva 1 activa	Leva 1 inactiva	Leva 0 activa	Leva 0 inactiva

La causa de la alarma se puede consultar en el byte 1.

Leva: Evalúe los bytes 2 y 3 según la tabla.

Alarmas de proceso perdidas

Si una alarma de proceso todavía no se ha terminado de procesar en el OB de alarma de proceso, el módulo memoriza todos los eventos siguientes de alarma de proceso. Si vuelve a aparecer un evento antes de que la alarma de proceso se haya podido disparar, el módulo disparará la alarma de diagnóstico "Alarma de proceso perdida".

Interfaz de control y realimentación del FM 352

Aplicación

En aplicaciones especiales o en el nivel de las alarmas es preciso un acceso especialmente rápido a las señales de respuesta y de control. A estos datos se accede directamente a través de las áreas de entrada y salida del módulo.

Para coordinar el arranque cada vez que arranca el módulo (p. ej. tras insertar el módulo o después de una transición STOP-RUN de la CPU), la instrucción CAM_CTRL (Página 7453) debe llamarse hasta que se muestre el final del arranque a través de RETVAL = 0.

Nota

Para acceder directamente a los datos del FM 352 sólo se deben emplear los datos no internos aquí descritos del modo indicado. De lo contrario, el acceso de su programa de usuario al módulo puede sufrir anomalías.

Leer señales de respuesta mediante acceso directo

Las direcciones de byte se indican de forma relativa con respecto a la dirección de salida del módulo. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7461).

En AWL se accede a los datos con los comandos PEB (leer 1 byte) y PED (leer 4 bytes).

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	PARA	interno	interno	DATA_ERR	interno	DIAG	interno	interno
Byte 1	0	0	0	CAM_ACT	0	0	0	0
Byte 2	interno							
Byte 3	0	0	FVAL_DONE	HYS	GO_P	GO_M	MSR_DONE	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								
Byte 8	TRACK_OUT							
Byte 9								
Byte 10								
Byte 11								

Escribir las señales de control mediante acceso directo

Las direcciones de byte se indican de forma relativa respecto a la dirección de entrada del módulo. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7461).

En AWL se accede a los datos con los comandos PAB (escribir 1 byte) y PAW (escribir 2 bytes).

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	interno							
Byte 1	0	CNTC1_EN	CNTC0_EN	CAM_EN	DIR_P	DIR_M	0	0
Byte 2	TRACK_EN							
Byte 3								

Ejemplo: Valor real de posición ACT_POS

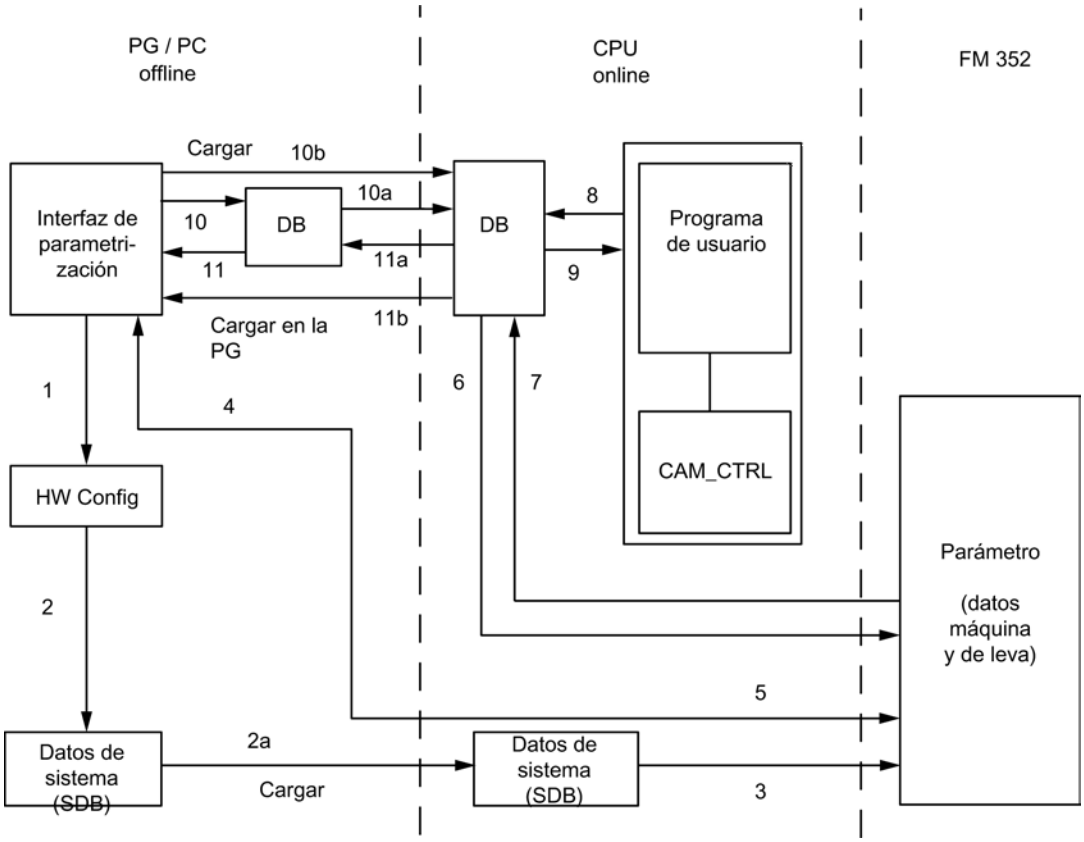
La dirección inicial del módulo es 512.

```
AWL
L PED 516      //Leer valor real de posición actual (ACT_POS) con acceso directo:
               //Dirección inicial del módulo + 4
```

Vías de transferencia de parámetros

Se entiende por parámetros los datos máquina y los datos de leva que siguen a continuación.

La figura siguiente muestra las vías de transferencia de parámetros al FM 352.



1 Guardar los parámetros en la interfaz de parametrización.

2 Guardar, compilar y cargar la configuración de hardware en la CPU.

3 La CPU escribe los parámetros en el módulo en parametrizaciones de sistema.

4 Cargar los parámetros del módulo en la programadora con el comando "Cargar sistema de destino en PG".

5 Cargar los parámetros en el módulo desde la interfaz de parametrización con el comando "Cargar sistema de destino".

6 Escribir los parámetros en el módulo con peticiones del programa de usuario.

7 Leer los parámetros desde el módulo con peticiones del programa de usuario.

8 Almacenar parámetros del programa de usuario en el DB Online.

9 Leer parámetros del DB Online en el programa de usuario.

10 Exportar parámetros desde la interfaz de parametrización hasta el DB (DB Offline o DB Online); posteriormente, también debe cargarse un DB Offline en la CPU.

11 Importar parámetros de un DB Online o DB Offline a la interfaz de parametrización.

Casos de aplicación de la transferencia de parámetros

Caso de aplicación	Pasos
Los parámetros se editan con la interfaz de parametrización. A continuación, el módulo se tiene que parametrizar automáticamente en el arranque.	Ejecute los pasos 1, 2 y 3.
Los parámetros se modifican durante la puesta en servicio en el modo Test en la interfaz de parametrización.	Ejecute los pasos 4 y 5
Los parámetros modificados durante la puesta en servicio se tienen que cargar automáticamente durante el arranque.	Ejecute los pasos 1, 2 y 3.
Los parámetros se crean con la interfaz de parametrización. En el arranque, el módulo sólo se debe parametrizar con el programa de usuario mediante bloques de datos.	Ejecute los pasos 10 y 6.
Para crear cómodamente datos de reserva para recetas.	Ejecute el paso 10.
Los parámetros se crean con la interfaz de parametrización. Éstos tienen que estar a disposición del programa de usuario para modificaciones provisionales.	Ejecute los pasos 1, 2 y 3 para la parametrización automática.
	Ejecute los pasos 10 y 7 para el acceso por medio del programa de usuario.
Los parámetros existentes se modifican (exclusivamente) con el programa de usuario.	Ejecute los pasos 7, 9, 8 y 6.
Para ver los datos modificados por el programa de usuario con la interfaz de parametrización.	Ejecute el paso 11.
Los parámetros modificados por el programa de usuario se tienen que cargar automáticamente también en el arranque.	Ejecute los pasos 6, 11, 1, 2 y 3.

Puesta en servicio del FM 352

Uso del FM 352 desde el programa de usuario

Inclusión del FM 352 en un programa de usuario

Para integrar el FM 352 en un programa de usuario pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

Las instrucciones CAM_INIT y CAM_CTRL se activan en el programa de usuario de la CPU. Éstas realizan la comunicación entre la CPU y el FM 352. El DB de canal (Página 7461) se crea automáticamente como DB de instancia en la instrucción CAM_CTRL, si se integra la instrucción CAM_CTRL en el proyecto.

Todos los datos de parámetros pertenecientes a un canal del módulo (datos máquina, datos de leva) se encuentran en el DB de parámetros (Página 7469) del módulo. Inserte un bloque de datos del tipo CAM_PxxxTYPE en el proyecto mediante "Bloques de programa > Agregar bloque". Dependiendo del alcance se requieren los siguientes tipos de DB:

- Hasta 16 levas: CAM_P016TYPE
- Hasta 32 levas: CAM_P032TYPE

- Hasta 64 levas: CAM_P064TYPE
- Hasta 128 levas: CAM_P128TYPE

Además, para el FM 352 se puede utilizar también la instrucción CAM_DIAG, que permite transferir informaciones de diagnóstico al DB de diagnóstico. El DB de diagnóstico (Página 7471) se crea automáticamente como DB de instancia en la instrucción CAM_DIAG, si se integra la instrucción CAM_DIAG en el proyecto.

Puesta en servicio del módulo

Incluso sin programa de usuario es posible realizar un primer test del módulo con el cuadro de diálogo "Puesta en servicio". Tenga en cuenta particularmente la advertencia siguiente.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

Para evitar daños personales y materiales observe los puntos siguientes:

- Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Sólo así puede garantizarse que en caso de un fallo del software o del equipo sea posible desconectar la instalación de modo seguro.
- Instale finales de carrera por hardware que actúen directamente en las etapas de potencia de todos los accionamientos.
- Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.
- Si controla y vigila paralelamente el FM 352 desde el programa y desde el cuadro de diálogo "Puesta en servicio" pueden producirse conflictos cuyos efectos no están definidos. Por lo tanto, conmute siempre la CPU al estado STOP cuando trabaje con el cuadro de diálogo "Puesta en servicio" o bien desactive el programa.

Consulte también

CAM_INIT (Página 7453)

CAM_CTRL (Página 7453)

CAM_DIAG (Página 7455)

FM 352: Diagnóstico online

Información actual sobre el módulo

Diagnóstico estándar

Indicador de errores mediante el LED de error general del FM 352

¿Dónde se indica el error?

Si el LED rojo de error general se enciende, significa que se ha producido un error o bien en el módulo (error interno), o bien en las conexiones de cable (error externo). El LED se ilumina también en caso de errores de aparición breve, por lo menos, durante 3 s.

¿Qué errores se indican?

La tabla siguiente contiene los tipos de error que se indican mediante el LED de error general.

LED	Significado	Aclaraciones
SF	Error de grupo de fallos internos y externos	Este LED indica los siguientes estados de error del FM 352: • Alarma de proceso perdida • Vigilancia de tiempo transcurrida • FM 352 no configurado • FM 352 parametrizado con errores • Falta tensión auxiliar externa de 24 V • Falta conector frontal • Rotura de hilo Encoder • Fallo operativo • Error de telegrama encóder absoluto • Pulso erróneo encoder incremental o falta marca cero Para eliminar el error, ver Clases de error (Página 9826).

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

Procesamiento de las alarmas

El FM 352 puede disparar alarmas de proceso y de diagnóstico. Estas alarmas se procesan en un OB de alarma. Cuando se dispara una alarma sin que se encuentre cargado el OB correspondiente, la CPU pasa a STOP.

Para habilitar las alarmas de diagnóstico se procede del modo siguiente:

1. Seleccione el módulo en HWCN
2. Habilite la alarma de diagnóstico en Edición > Propiedades del objeto > Parámetros básicos.
3. Guardar y compilar la configuración.
4. Cargue la configuración en la CPU

Visión de conjunto de las alarmas de diagnóstico

Los eventos y errores siguientes disparan una alarma de diagnóstico:

- Fallo operativo
- Datos máquina erróneos
- Datos de levas erróneos
- Error de diagnóstico

Estos errores se describen con detalle en el apartado "Clases de error (Página 9826)".

Reacción del módulo en caso de error con alarma de diagnóstico

- El procesamiento de levas se desactiva.
- Con las alarmas de diagnóstico siguientes se borra la sincronización:
  - Falta conector frontal, falta tensión auxiliar externa
  - se ha detectado un error de señal cero, error de línea (señales de encóder de 5 V)
  - se ha rebasado la zona de desplazamiento (se notifica con un error de operador)
  - Ajustar valor real no se puede ejecutar (se notifica con un error de operador)
- Las señales de control dejan de procesarse excepto en un caso.
  Excepción:
  En caso de desbordamiento del final de carrera por software todavía es posible un cambio de sentido en el modo de simulación.
- Los selectores de función y las peticiones se continúan procesando.

El módulo detecta un fallo ("entrante")

Una alarma de diagnóstico es "entrante" cuando hay por lo menos un error en cola de espera. Si no se han eliminado todos los errores, los errores restantes, se notificarán de nuevo como "entrantes".

Secuencia:

1. El FM 352 detecta uno o varios fallos y dispara una alarma de diagnóstico. El LED de error se enciende. El error se registra en el búfer de diagnóstico.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Es posible evaluar la información de inicio del OB 82.

4. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
5. Encontrará más información si llama la instrucción CAM_DIAG (Página 7455).

El módulo detecta el paso al estado libre de fallos ("saliente")

Una alarma sólo es "saliente" cuando se ha eliminado el último error del módulo.

Secuencia:

1. El FM 352 detecta que todos los fallos han sido eliminados y dispara una alarma de diagnóstico. El LED de error está apagado. El búfer de diagnóstico no se modifica.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
4. Evalúe el bit OB82_MDL_DEFECT.
   Si este bit es "0", no hay errores en el módulo. Su evaluación puede finalizar aquí.

Alarmas de diagnóstico según el estado de la CPU

- En estado STOP de la CPU las alarmas de diagnóstico están bloqueadas por el FM 352.
- Si con la CPU en estado STOP no se eliminan todos los fallos pendientes, el FM 352 vuelve a notificar como "entrantes" los errores todavía no eliminados tras la transición al estado RUN.
- Si se eliminan todos los errores pendientes con la CPU en estado STOP, el estado libre de errores del FM 352no se notificará con una alarma de diagnóstico tras la transición al estado RUN.

Evaluación de errores en el programa de usuario

Reacción de errores en el programa de usuario

En el programa de usuario es posible reaccionar de forma precisa a los errores que aparecen. Para ello se dispone de los siguientes recursos:

- Los valores de retorno RETVAL de las instrucciones integradas como indicador de grupo de los errores que han aparecido durante el transcurso de las instrucciones.
- Los bits de error _ERR de las peticiones como indicador de grupo de errores que se han producido durante la ejecución de una petición.
- El bit de error DATA_ERR como indicador de grupo de un error que el FM 352 ha detectado durante una petición de escritura.
- La identificación de errores en JOB_ERR de la causa de error en la comunicación entre la instrucción y el FM 352.
- La instrucción CAM_DIAG (Página 7455) para leer el búfer de diagnóstico del FM 352. Aquí es posible averiguar las causas de error de las peticiones y los eventos asíncronos (fallos operativos y errores de diagnóstico).
- Alarma de diagnóstico para una respuesta rápida a eventos.

Búfer de diagnóstico del módulo

Eventos de diagnóstico

El búfer de diagnóstico del módulo contiene como máximo 4 eventos de diagnóstico y está organizado como búfer anular.

Un evento de diagnóstico se escribe en el búfer cuando se detecta un aviso (de error) entrante. Esto puede ser un aviso, un error isócrono (error de datos) o incluso un error asíncrono (error de servicio y error de diagnóstico). De una causa de un error pueden surgir también varias entradas como errores consiguientes. Los avisos salientes no generan entradas en el búfer de diagnóstico.

Para cada evento de diagnóstico se indican:

- Estado (siempre entrante)
- Error interno
- Error externo
- Clase de error
- Número de error.
- Número de canal
- Número de levas (si hay errores de datos de levas)

Cuando se escribe un evento de diagnóstico en el búfer de diagnóstico, la señal de realimentación DIAG = 1 se activa.

El búfer de diagnóstico puede transferirse como un todo con la instrucción CAM_DIAG (Página 7455) a un bloque de datos (DB de diagnóstico) o a través de un cuadro de diálogo de evaluación de errores del software de configuración. Si se lee el búfer de diagnóstico, el módulo activa la señal de realimentación DIAG = 0.

Nota

Si el búfer de diagnóstico se lee al mismo tiempo con la instrucción CAM_DIAG y el cuadro de diálogo de evaluación de errores, puede ocurrir que uno de los eventos de diagnóstico que se ha producido no sea detectado por el programa.

Datos y estructura del DB de diagnóstico

Datos y estructura del DB de diagnóstico

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

La tabla siguiente muestra la estructura del DB de diagnóstico.

Dirección	Nombre	Tipo	valor inicial	Comentarios
0.0	MOD_ADDR (registrar)	INT	0	Dirección del módulo
256.0	JOB_ERR	INT	0	Error de comunicación
258.0	JOBBUSY	BOOL	FALSE	1 = Petición activa
258.1	DIAGRD_EN	BOOL	FALSE	1 = Leer el búfer de diagnóstico necesariamente
260.0	DIAG_CNT	INT	0	Número de entradas válidas de la lista
262.0	DIAG[1]	STRUCT		Entrada más reciente de datos de diagnóstico
272.0	DIAG[2]	STRUCT		Segunda entrada de datos de diagnóstico
282.0	DIAG[3]	STRUCT		Tercera entrada de datos de diagnóstico
292.0	DIAG[4]	STRUCT		Última entrada de datos de diagnóstico

La tabla siguiente muestra la estructura de una entrada de diagnóstico DIAG[n].

Dirección	Nombre	Tipo	valor inicial	Comentarios
+0.0	STATE	BOOL	FALSE	0 = Evento saliente 1 = Evento entrante
+0.1	INTF	BOOL	FALSE	1 = Error interno
+0.2	EXTF	BOOL	FALSE	1 = Error externo
+2.0	FCL	INT	0	Clase de error 1: Fallo operativo 4: Fallo de datos 5: Fallo de datos máquina 7: Fallo de datos de leva 15: Avisos 128: Error de diagnóstico
+4.0	FNO	INT	0	Número de error 0 a 255
+6.0	CH_NO	INT	0	Número de canal (siempre 1)
+8.0	CAMNO	INT	0	Número de leva de 0 a 127 con clase de error = error de datos de leva

Lista de los avisos JOB_ERR

Avisos JOB_ERR

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A0	32928	-32608	Acuse negativo al leer el módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2 [1)]	32930	-32606	Error de protocolo en la capa 2 (transferencia de datos dentro de PROFINET / PROFIBUS DP interrumpida, p. ej. por rotura de hilo, ausencia de conector de cierre, error de parametrización, etc.)

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A3 [1]	32931	-32605	Error de protocolo en User-Interface/User (transferencia de datos dentro de PROFINET / PROFIBUS DP interrumpida, p. ej. por rotura de hilo, ausencia de conector de cierre, error de parametrización, etc.)
80A4 [1]	32932	-32604	Anomalía en la comunicación del bus K
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo teórico.
80C0 [1]	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1 [1]	32961	-32575	Los datos de una tarea de escritura semejante todavía no se han procesado en el módulo.
80C2 [1]	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3 [1]	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados de instante.
80C4 [1]	32964	-32572	Error de comunicación
80C5 [1]	32965	-32571	La periferia descentralizada no está disponible.
80C6 [1]	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (Petición de escritura)
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (Petición de escritura)
853A	34106	-31430	DB de parámetros no presente. (Petición de escritura)
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras un error. (Petición de escritura)
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (Petición de lectura)
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (Petición de lectura)
873A	34618	-30918	DB de parámetros no presente. (Petición de lectura)
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras un error. (Petición de lectura)

[1] Los errores 80A2 a 80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario. Los avisos con la estructura 7xxx indican situaciones temporales de la comunicación.

Clases de error

Fallo operativo

Clase de error 1: Fallo operativo

Los fallos operativos se detectan de forma asíncrona respecto a un control.

La tabla siguiente muestra los fallos operativos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Rebase del inicio del final de carrera por software**		Sí
2	**Rebase del fin del final de carrera por software**		Sí
3	**Rebase del inicio de la zona de desplazamiento**		Sí
4	**Rebase del fin de la zona de desplazamiento**		Sí
13	**Ajustar valor real al vuelo no ejecutable**		Sí
	Causa	Los finales de carrera por software, tras ajustar valor real al vuelo, se encuentran fuera de la zona de desplazamiento (-100 m...+100 m o -1.000 m...+1.000 m). El decalaje resultante de Ajustar valor real / Ajustar valor real al vuelo es mayor que ±100 m o ±1.000 m.	
	Efecto	Eje no sincronizado.	

Error de datos

Clase de error 4: Error de datos

Los errores de datos se detectan de forma síncrona respecto a un control/mando.

La tabla siguiente muestra los errores de datos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
10	**Decalaje de señal cero erróneo**		No
	Causa	El decalaje de señal cero es mayor que ± 100 m o ± 1.000 m. Los finales de carrera por software, tras ajustar el decalaje de señal cero, se encuentran fuera de la zona de desplazamiento (-100 m...+100 m o -1.000 m...+1.000 m). Eje rotativo: El valor del decalaje de señal cero es mayor que el final del eje rotativo.	
11	**Indicación del valor real errónea**		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
12	**Punto de referencia erróneo**		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
20	**Activar datos máquina no admisible**		no
	Causa	En el módulo no hay datos máquina nuevos (sin errores)	
21	**Ajustar valor real al vuelo no admisible**		no
	Causa	Se ha intentado ejecutar "Ajustar valor real al vuelo" teniendo "Redisparar punto de referencia" activado.	
27	**Ajuste codificado en bits no permitido**		no

Nº	Significado		Alarma de diagnóstico
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0. Se ha intentado seleccionar simultáneamente" Medición de longitud" y "Captura de bordes".	
28	Redisparar el punto de referencia no admisible		no
	Causa	Se ha intentado ejecutar "Redisparar el punto de referencia" estando "Ajustar valor real al vuelo" pendiente. Se ha intentado ejecutar "Redisparar el punto de referenciao" con un encóder SSI.	
29	Comando codificado en bits no permitido		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
30	Tiempo de anticipación erróneo		no
31	Número de leva erróneo		no
	Causa	La leva no es válida. El número de leva no se encuentra en el rango 0..127.	
32	Inicio de leva erróneo		no
	Causa	El inicio de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: El principio de leva es < 0 o mayor que el final del eje rotativo.	
33	Fin de leva erróneo / Tiempo de activación erróneo		no
	Causa	El fin de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: El final de leva es < 0 o mayor que el final del eje rotativo. La leva no está activada por lo menos en la duración de un impulso. En una leva invertida no hay una distancia mínima de 4 impulsos entre inicio de leva y fin de leva.	
34	No es posible cancelar la activación del valor real		no
	Causa	Con un encóder SSI y un eje líneal, tras la ejecución del ajuste el valor real de posición se encontraría fuera de la zona de trabajo.	
35	Indicación del valor real errónea al Ajustar valor real / Ajustar valor real al vuelo		no
	Causa	La indicación del valor real se encuentra fuera del margen numérico admisible de ± 100 m ó ± 1.000 m. Tras la ejecución del ajuste, los finales de carrera por software se encontrarían fuera de la zona de desplazamiento (-100 m...+1000 m ó -1.000 m...+1.000 m). El decalaje resultante de Ajustar valor real / Ajustar valor al vuelo sería mayor que ±100 m ó ±1.000 m.	
107	Eje no parametrizado		no
	Causa	En el eje no hay ningún dato máquina disponible. En el eje no hay ningún dato máquina activado.	
108	Eje no sincronizado		no
	Causa	Uno de los ajustes "Ajustar valor real" o "Ajustar valor real al vuelo" se ha activado a pesar de que el eje no está sincronizado.	
109	Procesamiento de levas en curso.		no
110	Número erróneo de levas a modificar		no

Error de datos máquina

Clase de error 5: Error de datos máquina

La alarma de diagnóstico sólo se disparará con un módulo de datos del sistema (SDB) con errores.

La tabla siguiente muestra errores de datos máquina posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
5	Error en el ajuste de la alarma de proceso		Sí
	Causa	Ha intentado seleccionar una alarma de proceso que el módulo no soporta.	
6	Distancia mínima entre bordes errónea		Sí
	Causa	Ha indicado como distancia mínima entre bordes un valor < 0 ó > 10^9 µm.	
8	Tipo de eje erróneo		Sí
	Causa	Como tipo de eje no ha indicado ni 0 ni 1.	
9	Final erróneo del eje rotativo		Sí
	Causa	El valor del final del eje rotativo se encuentra fuera del rango admitido de 1 a 10^9 µm o respectivamente de 1 a 10^8 µm (según sea la resolución).	
10	Tipo de encóder erróneo		Sí
	Causa	El valor del tipo de encóder se encuentra fuera del rango admisible de 1 a 10	
11	Recorrido erróneo por vuelta		Sí
	Causa	El valor del recorrido/vuelta se encuentra fuera del margen admisible de 1 a 10^9 µm (independientemente de la resolución).	
13	Número erróneo de incrementos/vuelta del encóder(véase el apartado "Incrementos por vuelta del encóder (Página 9795)")		Sí
14	Número erróneo de vueltas (véase el apartado ""Número de vueltas de encóder (Página 9800)")		Sí
15	Velocidad de transferencia errónea		Sí
	Causa	Para la velocidad de transferencia ha indicado un valor fuera del rango admisible de 0 a 3.	
16	Coordinada del punto de referencia errónea		Sí
	Causa	La coordenada se encuentra fuera del margen de -100 m a +100 m o, respectivamente, de -1000 m a +1000 m, según la resolución. Eje lineal: La coordenada se encuentra fuera de la zona de trabajo. Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
17	Reajuste erróneo del encóder absoluto		Sí
	Causa	Encóder SSI de recorrido: El valor del reajuste del encóder absoluto no se encuentra dentro del rango del encóder (Incrementos por vuelta × Cantidad de vueltas - 1).	
18	Forma errónea para redisparar el punto de referencia		Sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0, 1, 6 y 7.	
19	Ajuste erróneo del sentido		Sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0 y 1.	
20	Supervisión de hardware no posible		Sí
	Causa	Ha puesto en "FALSE" la vigilancia de errores de telegrama en el DB de parámetros.	

Nº	Significado		Alarma de diagnóstico
21	**Inicio erróneo del final de carrera por software**		Sí
	Causa	Eje lineal: El inicio del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución). Eje lineal: El inicio del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es menor que -100 m ó -1000 m (según la resolución).	
22	**Fin erróneo del final de carrera por software**		Sí
	Causa	Eje lineal: El fin del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución) o es menor que el inicio del final de carrera por software. El fin del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es mayor que +100 m ó +1000 m (según la resolución).	
144	**Alcance erróneo**		Sí
	Causa	Para el alcance ha indicado un valor distinto de 0 a 3.	
145	**Histéresis errónea**		Sí
	Causa	La histéresis se encuentra fuera del rango 0..65535 × resolución. El decalaje de señal cero es mayor que ¼ × zona de trabajo o ¼ × área de eje rotativo.	
146	**Velocidad de simulación errónea**		Sí
	Causa	La velocidad de simulación se encuentra fuera del rango 1000 × RES hasta 3×10^7 × RES o es mayor que 5×10^8 µm/min. La velocidad de simulación no puede ajustarse internamente.	
147	**Pista errónea**		Sí
	Causa	Se ha seleccionado el control de una pista fuera de 0..15 (bit 0..15).	
148	**Selección errónea de entradas de habilitación**		Sí
	Causa	Ha querido habilitar una pista fuera de 3..10 (bit 0..7) mediante una señal externa.	
149	**Selección errónea de una pista especial**		Sí
	Causa	Ha querido definir una pista fuera de 0, 1 y 2 (bit 0, 1 y 2) a modo de pista especial	
150	**Valor de contaje superior erróneo pista 0**		Sí
	Causa	Ha indicado un valor de contaje < 2 ó > 65535 como valor de contaje superior.	
151	**Valor de contaje superior erróneo pista 1**		Sí
	Causa	Ha indicado un valor de contaje < 2 ó > 65535 como valor de contaje superior.	
200	**Resolución errónea**		Sí
	Causa	Ha indicado una resolución < 0,1 µm/impulso ó >1000 µm/impulso. Ha indicado un recorrido/vuelta y una cantidad de impulsos/vuelta que dan una resolución < 0,1 ó > 1000.	
201	**El encóder de recorrido no se adapta a la zona de trabajo / área del eje rotativo**		Sí
	Causa	Encóder SSI de recorrido y eje rotativo: El encóder de recorrido no cubre exactamente el área del eje rotativo. Eje lineal: El encóder de recorrido no cubre ni siquiera la zona de trabajo (finales de carrera por software incluidos).	

Error de datos de leva

Clase de error 7: Error de datos de leva

La alarma de diagnóstico sólo se disparará con un bloque de datos del sistema (SDB) con errores.

La tabla siguiente muestra los errores de datos de leva posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Alarma de proceso no admisible**		Sí
	Causa	Ha querido indicar una alarma de proceso con una leva con un número > 7.	
2	**Número de pista erróneo**		Sí
	Causa	El número de pista se encuentra fuera del margen 0 a 31.	
3	**Inicio de leva erróneo**		Sí
	Causa	El inicio de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: El principio de leva es < 0 o mayor que el final del eje rotativo.	
4	**Final de leva erróneo**		Sí
	Causa	El fin de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). La leva no tiene una longitud de por lo menos 1 impulso. Eje rotativo: El final de leva es < 0 o mayor que el final del eje rotativo. En una leva invertida no hay una distancia mínima de 4 impulsos entre inicio de leva y fin de leva.	
5	**Tiempo de activación erróneo**		Sí
	Causa	El tiempo de conexión es < 0µs. El valor máximo (Página 9807) depende del alcance.	
6	**Tiempo de anticipación erróneo**[1]		Sí
	Causa	El tiempo de anticipación es < 0 µs. El valor máximo (Página 9808) depende del alcance.	
50	**Demasiados juegos de levas**		Sí
	Causa	Ha querido indicar más juegos de levas de los posibles con este alcance.	
51	**Eje en funcionamiento**		Sí
	Causa	Ha querido indicar juegos de levas con la leva electrónica activada.	
52	**Eje no parametrizado**		Sí
	Causa	Ha querido indicar datos de levas a pesar de que todavía no hay datos máquina activos.	

[1] El mensaje de error también puede aparecer si ha parametrizado el sentido de contaje "invertido" en combinación con un encóder absoluto (SSI).

Avisos

Clase de error 15: Avisos

La tabla siguiente muestra los avisos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Inicio de parametrización**		no
	Causa	El módulo ha detectado una parametrización mediante un bloque de datos del sistema.	
2	**Final de la parametrización**		no
	Causa	El módulo ha procesado sin problemas la parametrización mediante un bloque de datos del sistema.	

Error de diagnóstico

Clase de error 128: Error de diagnóstico

La tabla siguiente muestra los errores de datos posibles y sus causas, así como los efectos y las soluciones posibles.

N.º	Significado		Alarma de diagnóstico
4	**Falta tensión auxiliar externa**		Sí
	Causa	La tensión auxiliar externa de 24 V no se ha conectado o ha caído, falta el conector frontal	
	Efecto	Consulte "Activación de alarmas de diagnóstico (Página 9818)": • El procesamiento de levas se desactiva • Desconexión de las salidas de pista: • Borrado de la sincronización, con encóders incrementales • El FM 352 no está parametrizado (señal de realimentación PARA = 0).	
	Solución	Comprobar que la conexión de 24 V sea correcta (si la conexión de 24 V es correcta, el módulo está averiado).	
51	**Vigilancia de tiempo transcurrida**		Sí
	Causa	• Interferencias fuertes en el FM 352 • Error en el FM 352	
	Efecto	• El módulo se resetea • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, este vuelve a estar listo para funcionar. • El módulo notifica el transcurso de la vigilancia del tiempo (watchdog) con "entrante" y "saliente"	
	Solución	• Eliminación de interferencias • Consulte al departamento comercial correspondiente cuando las circunstancias exactas que han dado lugar al error son de mayor entidad. • Cambio del FM 352	

N.º	Significado		Alarma de diagnóstico
52	**Caída de la alimentación interna del módulo**		Sí
	Causa	Error en el FM 352	
	Efecto	• El módulo se resetea • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, este vuelve a estar listo para funcionar.	
	Solución	Cambio del FM 352	
70	**Alarma de proceso perdida**		Sí
	Causa	El FM 352 ha detectado un evento de alarma de proceso, que no puede ser notificado porque el mismo evento todavía no ha sido procesado por el programa de usuario o la CPU.	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización, con encóders incrementales	
	Solución	• Incorporar el OB 40 en el programa de usuario • Comprobar la conexión de bus del módulo • Desactivar la alarma de proceso • Adapte el hardware y software a las necesidades de su proceso (p. ej. CPU más rápida, optimizar el programa de usuario)	
144	**Rotura de hilo Encóder**		Sí
	Causa	• El cable del encóder no está enchufado o está cortado • Encóder sin señales de cuadratura • Asignación de pines errónea • Longitud de cable excesiva • Cortocircuito de las señales del encóder	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización, con encóders incrementales	
	Solución	• Comprobar el cable del encóder • Guardar los datos técnicos del encóder • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador. • Guardar las especificaciones técnicas del módulo	
145	**Error de telegrama Encóder absoluto**		Sí
	Causa	El tráfico de telegramas entre el FM 352 y el encóder absoluto (SSI) tiene errores o está interrumpido: • El cable del encóder no está enchufado o está cortado • Tipo de encóder erróneo • Encóder mal configurado (encóders programables) • Longitud de telegrama mal indicada • El encóder proporciona valores erróneos (encóder defectuoso) • Interferencia en el cable del sistema de medición • Velocidad de transferencia escogida demasiado alta	

N.º	Significado		Alarma de diagnóstico
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • el último valor real correcto se mantiene sin cambios hasta el final de la siguiente transferencia SSI correcta	
	Solución	• Comprobar el cable del encóder • Comprobar el encóder • Comprobar el tráfico de telegramas entre el encóder y el FM 352	
146	Impulsos erróneos (encóder incremental)		Sí
	Causa	• Vigilancia del encóder ha detectado impulsos erróneos • La cantidad de incrementos por vuelta está mal introducida • Encóder defectuoso: no proporciona la cantidad de impulsos indicada • Marca cero errónea o no presente • Interferencias en el cable del encóder	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización	
	Solución	• Indicar correctamente el número de incrementos/vuelta. • Comprobar el encóder y su cable • Cumplir las normas de apantallamiento y de conexión a tierra • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador.	

11.2.1.4 Uso del FM 452

Asignación de terminales del conector frontal del FM 452

Conector frontal

En el conector frontal de 48 polos se conectan:

- el encóder;
- las salidas y las entradas digitales;
- las tensiones de alimentación de los encóders y de las salidas digitales.

Asignación de terminales del conector frontal (X1)

La tabla siguiente contiene la asignación de terminales del conector frontal del FM 452.

Terminal	Nombre	Detector de proximidad	Encóder incremental	Encóder absoluto
1		–		
2		–		

Terminal	Nombre	Detector de proximidad	Encóder incremental	Encóder absoluto
3	1L+	Tensión auxiliar DC 24 V		
4	A/DAT	–	Señal de encoder A (5 V)	Datos SSI
5	/A / /DAT	–	Señal de encóder A (5 V) invertida	Datos SSI invertidos
6	B / CLI [1]	–	Señal de encoder B (5 V)	Pulso de desplazamiento SSI entrada [1]
7	/B / /CLI [1]	–	Señal de encoder B (5 V) invertida	Pulso de desplazamiento SSI entrada invertida [1]
8	N	–	Señal de marca cero (5 V)	–
9	/N	–	Señal de marca cero invertida (5 V)	–
10	CLS [2]	–	–	Pulso de desplazamiento SSI salida
11	/CLS [2]	–	–	Pulso de desplazamiento SSI salida inverso
12	A*	Señal de encóder A (24 V)		–
13	B*	–	Señal de encóder B (24 V)	–
14	N*	–	Señal de marca cero (24 V)	–
15	Q0	Salida digital 0		
16	Q1	Salida digital 1		
17	Q2	Salida digital 2		
18	Q3	Salida digital 3		
19	Q4	Salida digital 4		
20	Q5	Salida digital 5		
21	Q6	Salida digital 6		
22	Q7	Salida digital 7		
23	DC 5.2 V	–	Alimentación del encoder (5,2 V)	
24	DC 24 V	Alimentación del encoder (24 V)		
25	M [3]	Masa encoder		
26	2L+	Tensión auxiliar DC 24 V		
27	RE	–	Salida tipo PNP/NPN [4]	–
28	Q8	Salida digital 8		
29	Q9	Salida digital 9		
30	Q10	Salida digital 10		
31	Q11	Salida digital 11		
32	Q12	Salida digital 12		
33	Q13	Salida digital 13		
34	Q14	Salida digital 14		
35	Q15	Salida digital 15		
36	3L+	Tensión auxiliar DC 24 V		
37	I0	Habilitación de pista de leva de freno		
38	I1	Medición de longitud/Detección de bordes/Ajustar valor real al vuelo		
39	I2	Sensor del punto de referencia		

Terminal	Nombre	Detector de proximidad	Encóder incremental	Encóder absoluto
40	I3	Habilitación de la señal de pista 3		
41	I4	Habilitación de la señal de pista 4		
42	I5	Habilitación de la señal de pista 5		
43	I6	Habilitación de la señal de pista 6		
44	I7	Habilitación de la señal de pista 7		
45	I8	Habilitación de la señal de pista 8		
46	I9	Habilitación de la señal de pista 9		
47	I10	Habilitación de la señal de pista 10		
48	M [3]	Masa para tensiones auxiliares		

[1] En modo de escucha

[2] En modo maestro

[3] Las conexiones de masa están cableadas en el módulo

[4] **Nota:** si se conecta un encóder incremental de otro fabricante en contrafase (salida tipo PNP/NPN), debe tenerse en cuenta lo siguiente:

- Salida tipo PNP: conecte RE (27) con la masa (25).
- Salida tipo NPN: conecte RE (27) con +24 V (24).

Encontrará más información sobre el diseño y el cableado en las instrucciones de servicio (http://support.automation.siemens.com/WW/view/es/1123332) de la leva electrónica FM 452.

Tensión auxiliar para encóders y salidas digitales (1L+, 2L+, 3L+)

La tensión auxiliar DC 24 V del encóder y las salidas digitales se vigila:

- por si hay una rotura de hilo en la línea de 24 V;
- por si se corta la alimentación.

En el conector frontal (X1) se proporcionan DC 24 V (borne 24) y DC 5,2 V (borne 23) para los diferentes tipos de encóder.

Alimentaciones de carga

La alimentación de la carga DC debe satisfacer los siguientes requisitos:

Como alimentación de la carga sólo debe utilizarse una pequeña tensión DC ≤ 60 V separada con seguridad de la red. La separación segura puede estar efectuada según los requisitos que figuran, entre otros, en:

- VDE 0100 Parte 410/HD 384-4-41/IEC 364-4-41
  (como pequeña tensión funcional con separación segura),
- VDE 0805/EN 60950/IEC 950
  (como pequeña tensión de seguridad SELV) o bien
- VDE 0106 Parte 101.

11 entradas digitales (I 0 a I 10)

A través de las 11 entradas digitales es posible conectar interruptores libres de rebotes (de 24 V tipo fuente) o sensores sin contacto (detectores de proximidad de 2 ó 3 hilos).

No se supervisa el cortocircuito o rotura de hilo en las entradas digitales y carecen de aislamiento galvánico respecto a la masa del módulo.

El estado de cada entrada se indica en el LED correspondiente.

16 salidas digitales (de Q 0 a Q 15)

Por medio de 16 salidas digitales se indica el estado correspondiente (on/off) de las pistas entre 0 y 15. Las salidas digitales carecen de aislamiento galvánico respecto a la masa del módulo.

Son posibles las cargas siguientes:

- Tensión de trabajo 24 V
- Carga de corriente 0,5 A/resistente al cortocircuito

El estado de cada salida se indica en el LED correspondiente.

Funciones para controlar las levas

Influencia de las funciones en el comportamiento de conmutación de las levas de tiempo

Cambios en el valor real

Es posible obviar una leva de tiempo con funciones que modifiquen el valor real:

- Ajustar valor real
- Ajustar valor real al vuelo
- Decalaje de señal cero
- Redisparar el punto de referencia

Activación de una leva de tiempo

Si se obvia el principio de una leva de tiempo mediante una de las funciones arriba mencionadas, esta leva se activará siempre y cuando el sentido real del movimiento del eje

se corresponda con el sentido de actuación parametrizado para la leva. El tiempo de conexión parametrizado se cuenta.

Nota

En posición de parada, el sentido del movimiento depende de las fluctuaciones del valor real.

Si desea tener en cuenta un sentido de movimiento definido también en posición de reposo, deberá parametrizar una histéresis mayor que las fluctuaciones del valor real en reposo.

Si el eje no está en movimiento, se conserva el último sentido de movimiento calculado.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

Las funciones con modificación del valor real en un eje rotativo pueden conducir a una activación no deseada de las levas de tiempo.

En cualquier caso, con un eje rotativo ajuste las levas de tiempo como "no válidas" a fin de influir en el valor real mediante las funciones arriba indicadas.

Ajustar valor real / Ajustar valor real al vuelo / Cancelar preselección del valor real

Definición

Las funciones "Ajustar valor real" y "Ajustar valor real al vuelo" permiten asignar una nueva coordenada al estado actual del sensor. Así, el sistema de coordenadas se desplaza en el valor: $REAL_{nuevo}$ - $REAL_{actual}$

Donde:

- $REAL_{nuevo}$ es el valor predefinido
- $REAL_{actual}$ es el valor real en el momento de la ejecución

Calcular coordenadas nuevas

Todas las posiciones preseleccionadas en el sistema de coordenadas desplazado se calculan con la fórmula siguiente:

$$\text{Coordenada}_{nueva} = \text{Coordenada}_{antigua} + (REAL_{nueva} - REAL_{actual})$$

Requisitos

- El eje tiene que estar sincronizado.
- En "Ajustar valor real al vuelo": La entrada digital I1 tiene que estar interconectada.

Desarrollo de la función

1. Introduzca las coordenadas para el valor real o para el valor real al vuelo en el DB de canal.
   - Eje lineal:
     El valor real indicado se debe seleccionar de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admisible tras llamar la función.
     El valor de desplazamiento que resulta de **($REAL_{nuevo}$ - $REAL_{actual}$)** tiene que ser menor o igual al valor de la zona de desplazamiento admisible (máximo 100 m o 1000 m, respectivamente).
   - Eje rotativo:
     Para el valor real indicado rige:
     0 ≤ Valor real < Final del eje rotativo
2. Active los correspondientes bits de inicio en el DB de canal.
3. Llame la instrucción CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457).
   Se ejecuta de inmediato "Ajustar valor real".
   "Ajustar valor real al vuelo" se ejecuta con el flanco ascendente siguiente en la entrada digital I1. El bit FVAL_DONE se activa.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentario
36.4	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.5	FVAL_EN	BOOL	FALSE	1 = Ajustar valor real al vuelo
90.0	AVAL	DINT	L#0	Coordenada para valor real
94.0	FVAL	DINT	L#0	Coordenada para valor real al vuelo
25.5	FVAL_DONE	BOOL	FALSE	1 = Ejecutado Ajustar valor real al vuelo

Efectos de la función

En el ejemplo "Ajustar valor real" a 400 mm (en la posición 200 mm) se muestra cómo esta función provoca un decalaje del sistema de coordenadas. De ello resultan los siguientes efectos:

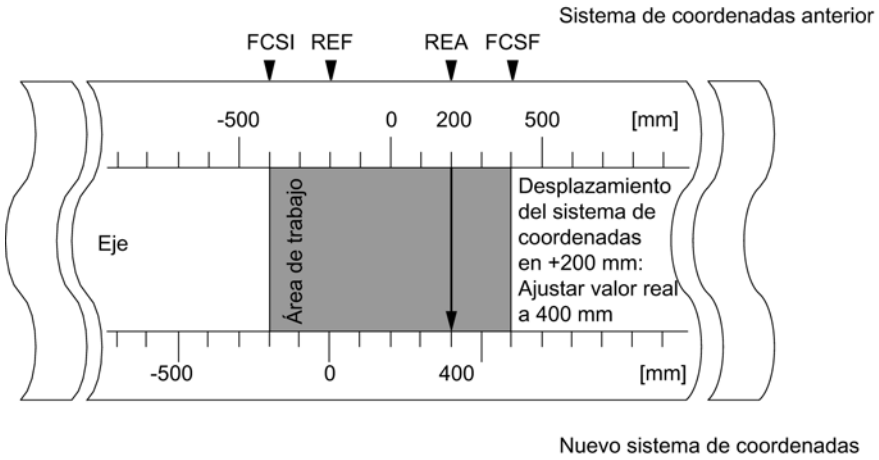
- La posición de la zona de trabajo **no** se desplaza físicamente.
- A cada uno de los puntos (como, p.ej. a los finales de carrera por software) se les asignan nuevos valores de coordenadas.
- Las levas conservan sus valores de coordenadas y, por lo tanto, se encuentran en otra posición física.

- En el caso del eje sincronizado y procesamiento de levas habilitado, el valor real de la posición puede obviar flancos de levas o levas completas mediante esta función.
- El cambio de estado de la leva que normalmente dispararía una alarma se podría perder.

Nota

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9834).

La tabla siguiente muestra el desplazamiento del sistema de coordenadas con "Ajustar valor real" / "Ajustar valor real al vuelo".

Ajustar valor real	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	-200	200	400
	-200	0	400	600

Deshacer la función

Con la función "Cancelar preselección del valor real" se deshace el decalaje de coordenadas establecido respectivamente con "Ajustar valor real" o "Ajustar valor real al vuelo".

Un "Ajustar valor real al vuelo" iniciado no se puede borrar antes de su ejecución mediante un flanco ascendente en la entrada I1. Pero sí puede sobrescribirse con un nuevo "Ajustar valor real al vuelo".

Con un arranque del módulo se inicializan estas funciones.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentario
35.2	AVALREM_EN	BOOL	FALSE	1: Deshacer Ajustar valor real

Posibles fuentes de error

"Ajustar valor real al vuelo" y "Redisparar punto de referencia" no deben ejecutarse de forma simultánea.

Con la función "Ajustar valor real al vuelo" puede notificarse un error de operador si esta función implica que un final de carrera por software excede el margen de desplazamiento admisible con un flanco ascendente en I1. Este error de operador se notifica con una alarma de diagnóstico y queda registrado en el búfer de diagnóstico.

Ejecutar un decalaje de señal cero

Definición

Con la función "Decalaje de señal cero" se desplaza en el valor introducido la señal cero del sistema de coordenadas. El signo define la dirección del decalaje.

Calcular nueva coordenada

Todos los valores del sistema de coordenadas desplazado se calculan con la fórmula siguiente:

$\text{Coordenada}_{nueva} = \text{coordenada}_{antigua} + (\text{DSC}_{nueva} - \text{DSC}_{antigua})$

$\text{DSC}_{antigua}$ indica un decalaje anterior de señal cero. Si antes de la llamada no ha habido activo ningún decalaje de señal cero, introduzca para $\text{DSC}_{antiguo}$ el valor 0.

Con esta fórmula es posible calcular qué valores de coordenadas adoptan, por ejemplo, los finales de carrera por software.

Desarrollo de la función

1. Introduzca el valor del decalaje de señal cero en el DB de canal.
   - Eje lineal:
     El decalaje de señal cero se tiene que escoger de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admisible tras la llamada de la función.
   - Eje rotativo:
     Para el decalaje de señal cero rige:
     Valor del decalaje de señal cero ≤ Fin del eje rotativo.
2. Active el bit de inicio correspondiente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.6	ZOFF_EN	BOOL	FALSE	1 = Activar decalaje de señal cero
86.0	ZOFF	DINT	L#0	Decalaje de señal cero

Efectos en un eje lineal

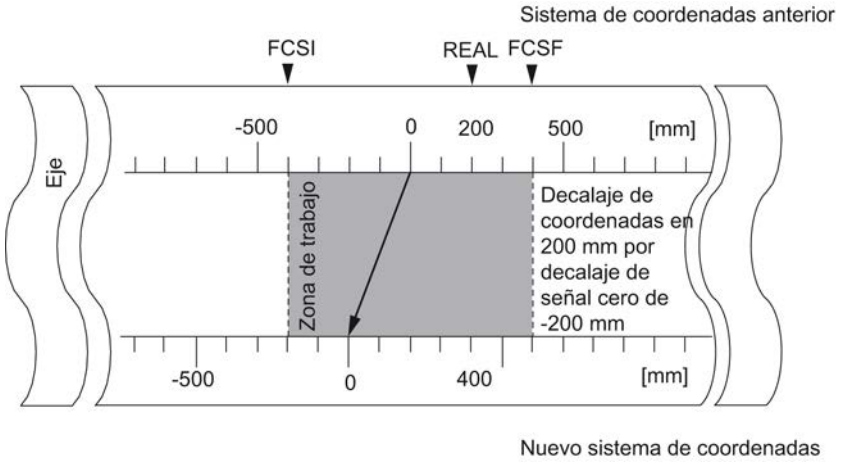
En el ejemplo de un decalaje de señal cero de -200 mm se ve cómo esta función desplaza el sistema de coordenadas en sentido positivo. De ello resultan los siguientes efectos:

- La zona de trabajo **no** se desplaza físicamente.
- A cada uno de los puntos (como, p.ej. a los finales de carrera por software) se les asignan nuevos valores de coordenadas.
- Las levas conservan sus valores de coordenadas y, de este modo, se encuentran en otra posición física.
- En el caso del eje sincronizado y procesamiento de levas habilitado, el valor real de la posición puede obviar flancos de levas o levas completas mediante esta función.
- El cambio de estado de la leva que normalmente dispararía una alarma, se puede perder.

Nota

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9834).

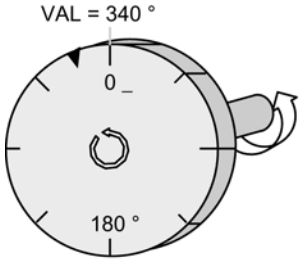
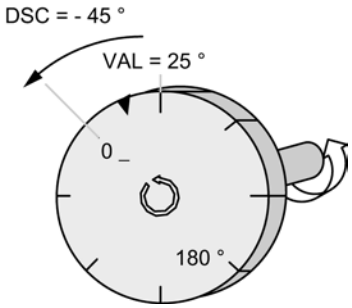
La tabla siguiente muestra el desplazamiento del sistema de coordenadas mediante el decalaje de señal cero.

Decalaje de señal cero	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	-200	200	400
	-200	0	400	600

Efectos en un eje rotativo

En el ejemplo de un decalaje de señal cero en -45° se ve cómo esta función gira el sistema de coordenadas:

La tabla siguiente muestra el giro del sistema de coordenadas mediante el decalaje de señal cero.

Herramienta con REAL = 340°	Herramienta con REAL = 25°
	

La señal cero gira -45 grados. Todos los puntos del eje adquieren nuevos valores de coordenada.

Considerando un **$DSC_{antiguo}$ = 0** resulta un nuevo valor de 385°

Como al final del eje rotativo en un sentido de giro positivo el valor real empieza de nuevo en 0 el valor real de 25º se calcula a partir de:

$Coordenada_{nueva}$ = $coordenada_{antigua}$ - (DSC_{nueva} - $DSC_{antigua}$) -final de eje rotativo

El valor **Final del eje rotativo** sólo tiene que restarse si

$Coordenada_{antigua}$ - **(DSC_{nueva} - $DSC_{antigua}$)** es mayor que Final del eje rotativo.

Pérdida de la sincronización

Si se pierde la sincronización a causa de un error o se reactiva mediante "Redisparar punto de referencia" **se mantiene** activo un decalaje de señal cero.

Deshacer función

Al definir un decalaje de señal cero de 0 se recupera el decalaje de señal cero existente.

Definir punto de referencia

Definición

Con la función "Definir punto de referencia" se sincroniza el eje. Esta función desplaza la zona de trabajo. Todos los decalajes generados por decalaje de señal cero o con "Ajustar valor real" se conservan.

La función proyecta la zona de trabajo sobre el eje. Así, con distintas entradas de valores la zona de trabajo puede encontrarse en posiciones diferentes dentro del rango físico del eje.

Requisitos

Es preciso que el procesamiento de levas esté desactivado.

Desarrollo de la función

1. Introduzca el valor de las coordenadas del punto de referencia en el DB de canal.
   - Eje lineal:
     La coordenada del punto de referencia no puede encontrarse fuera de los finales de carrera por software. Esto rige también para la coordenada del punto de referencia de un sistema de coordenadas desplazado.
   - Eje rotativo:
     Para la coordenada del punto de referencia rige:
     0 ≤ Coordenada del punto de referencia < Final del eje rotativo
2. Active el bit de inicio correspondiente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.3	REFPT_EN	BOOL	FALSE	1 = Definir coordenada del punto de referencia
98.0	REFPT	DINT	L#0	Coordenada del punto de referencia
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado

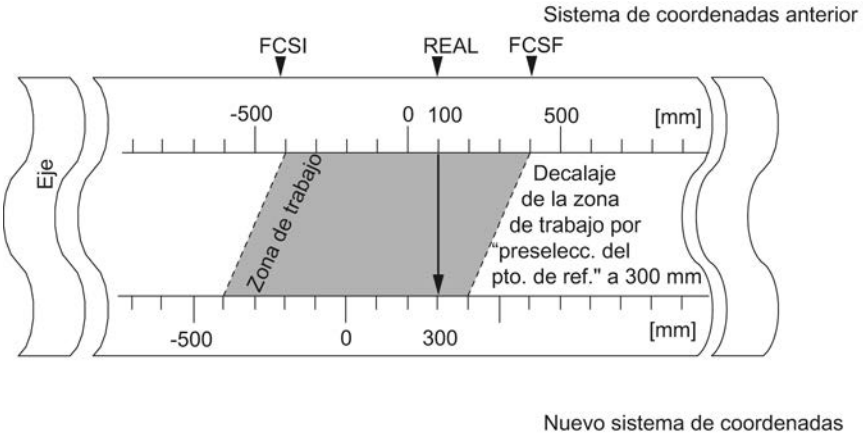
Efectos de la función

En el ejemplo "Definir punto de referencia" a 300 mm se ve cómo esta función desplaza la zona de trabajo en el eje.

De ello resultan los efectos siguientes:

- La posición real se fija en el valor de la coordenada del punto de referencia.
- La zona de trabajo se desplaza físicamente en el eje.
- Cada uno de los puntos conserva sus coordenadas originales pero se encuentran en posiciones físicas nuevas.
- El bit SYNC en las señales de realimentación está activado.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Definir punto de referencia".

Definir punto de referencia	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	-200	100	400
	-400	-200	300	400

Particularidades del encóder absoluto

Esta función es necesaria para un reajuste del encóder absoluto (Página 9868).

Modificar flancos de leva

Definición

Con la función "Modificar flancos de leva" es posible modificar durante el funcionamiento el principio de leva y, en el caso de las levas de recorrido, también el final de leva de una leva individual.

Requisitos

La leva que se quiera modificar tiene que ser válida.

Desarrollo de la función

1. Escriba el número de leva en el DB de canal.
2. En una leva de recorrido:
   Escriba el inicio de leva y el fin de leva en el DB de canal.
   En una leva de recorrido:
   Introduzca el valor del inicio de leva en el DB de canal.
3. Active el bit de inicio correspondiente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.7	CH01CAM_EN	BOOL	FALSE	1 = Escribir función para flanco de leva (1 leva)
102.0	CAM_NO	INT	0	Número de leva
104.0	CAM_START	DINT	L#0	Inicio de leva
108.0	CAM_END	DINT	L#0	Fin de leva

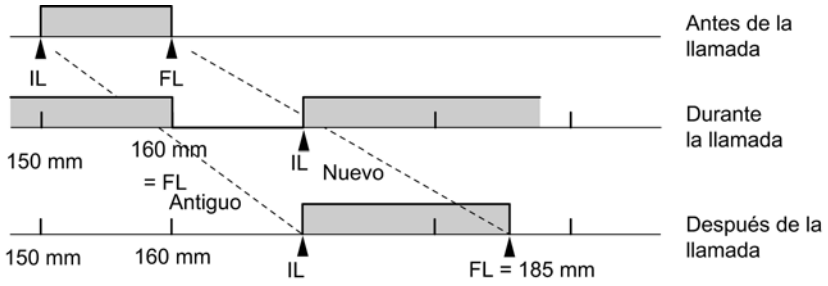
Efectos de la función

El FM x52 desplaza primero el flanco de activación y luego el de desactivación de la leva. Este proceso es independiente de la dirección en la que se desplace la leva.

Caso especial:

Con el proceso que se ha descrito anteriormente puede ocurrir que durante un breve tiempo surja una leva invertida (Página 9735) si el nuevo principio de leva es mayor que el antiguo fin de leva.

La figura siguiente muestra la modificación de los flancos de levas en pasos individuales.



Nota

Si para esta leva se ha habilitado una alarma de proceso, al detectar la leva invertida el FM x52 puede disparar una o dos alarmas de proceso, según se haya parametrizado.

Con la modificación del flanco de activación y/o de desactivación se pueden producir saltos de flancos de leva o de toda una leva.

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9834).

Los cambios de estado de la leva que normalmente dispararían una alarma de proceso, se pueden perder.

Leer los valores modificados

Los valores modificados se pueden leer con una de las peticiones CAM1RD_EN a CAM8RD_EN.

Deshacer la función

Los valores modificados se pierden al reiniciar el módulo.

Ejecutar "Cambio rápido de levas"

Definición

Con la función "Cambio rápido de levas" es posible modificar simultáneamente los datos de hasta 16 levas durante el funcionamiento.

Requisito

Las levas que se quieran modificar tienen que ser válidas.

Desarrollo de la función

1. Introduzca el número de levas que hay que modificar en el DB de canal.
2. Introduzca el número de levas que hay que modificar en el DB de canal.
3. Active los bits de inicio para los cambios deseados.
4. Introduzca los nuevos valores en el DB de canal.
5. Repita los pasos de 2 a 4 para cada leva que tenga que modificarse.
6. Active el bit de inicio correspondiente en el DB de canal

Datos utilizados en el DB de canal

Dirección Absoluta	Nombre	Tipo	valor inicial	Comentarios
37.0	CH16CAM_EN	BOOL	FALSE	1 = Función para un cambio rápido de levas (16 levas)
176.0	C_QTY	BYTE	B#16#0	Número de levas a modificar
177.0	DIS_CHECK	BOOL	FALSE	1 = Desactivar la comprobación de datos

Dirección relativa	Nombre	Tipo	valor inicial	Comentarios
+0.0	CAM_NO	BYTE	B#16#0	Número de la leva a modificar
+1.0	C_EFFDIR	BOOL	FALSE	1 = Cambio del sentido de actuación de la leva
+1.1	C_CBEGIN	BOOL	FALSE	1 = Cambio del inicio de la leva al valor CBEGIN
+1.2	C_CEND	BOOL	FALSE	1 = Cambio del final de leva/tiempo de conexión al valor CEND
+1.3	C_LTIME	BOOL	FALSE	1 = Cambio del tiempo de anticipación al valor LTIME
+1.4	CAM_OFF	BOOL	FALSE	1 = Desactivación de la leva durante el cambio de leva

Dirección relativa	Nombre	Tipo	valor inicial	Comentarios
+1.5	EFFDIR_P	BOOL	FALSE	1 = Sentido de actuación positivo (+)
+1.6	EFFDIR_M	BOOL	FALSE	1 = Sentido de actuación negativo (-)
+2.0	CBEGIN	DINT	L#0	Nuevo inicio de leva
+6.0	CEND	DINT	L#0	Nuevo fin de leva / Nuevo tiempo de conexión
+10.0	LTIME	INT	L#0	Nuevo tiempo de anticipación

Desactivar leva al modificar

Una modificación coherente del inicio y del final de la leva sólo es posible si en el instante de la modificación se ha desactivado la leva (CAM_OFF).

Comprobación de datos por parte del módulo

Con el parámetro DIS_CHECK (DB de canal) se define si se tiene que desactivar la comprobación de los datos transferidos a través del FM x52. Si se desactiva la comprobación de datos es preciso asegurarse de que sólo se transfieren valores admisibles. Si introduce valores no admisibles sin comprobación se puede producir un comportamiento impredecible del módulo.

- FALSE: El módulo comprueba todos los datos que se transfieren
- TRUE: La comprobación de los datos en referencia a los parámetros de la leva está desactivada. De este modo los datos que hay que modificar se incorporan más rápidamente en el FM x52.
  Independientemente de esta función siempre se comprueba si
  - el eje está parametrizado,
  - la cantidad de levas para modificar (C_QTY) es admisible,
  - la leva (número de leva) que tiene que modificarse es válida.

Sólo cuando todos los datos han sido comprobados y no se han detectado problemas, se activan en el módulo.

En caso de error todos los datos se rechazan.

Efectos de la función

Nota

Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9834).

Leer los valores modificados

Los valores modificados se pueden leer con una de las peticiones de CAM1RD_EN a CAM8RD_EN.

Deshacer la función

Los valores modificados se pierden con el reinicio del módulo.

Ejecutar "Medición de longitud" y "Captura de bordes"

Definición

Con las funciones "Medición de longitud" y "Captura de bordes" es posible calcular la longitud de una pieza de trabajo.

La medición de la longitud y la captura de bordes están y se mantienen activos hasta que se desactivan de nuevo o se seleccionan otros métodos de medición. Si se seleccionan ambos métodos de medición a la vez, la instrucción CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) activa la medición de longitud.

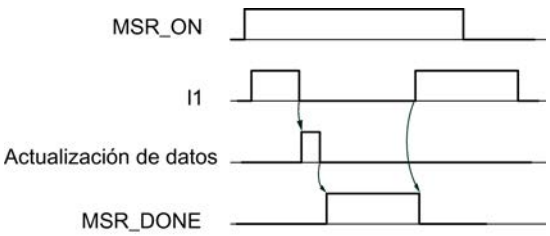
Requisitos

En la entrada I1 tiene que estar conectado un conector **libre de rebotes**.

Proceso de las funciones

Según el tipo de medición, el FM 452 actualiza los datos del módulo en instantes distintos. El FM 452 notifica cada actualización en un parámetro en la interfaz de realimentación.

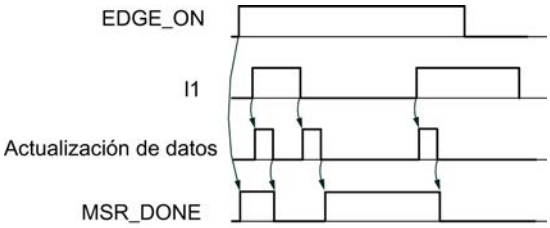
Medición de longitud



1. Active el selector de función de "Medición de longitud"
2. El flanco ascendente de la entrada I1 inicia la medición de longitud.
3. El flanco descendente de la entrada I1 pone fin a una medición en curso. El FM 452 actualiza los datos Valor inicial, Valor final y Longitud.
4. Con el parámetro activado MSR_DONE el FM 452 comunica la actualización de los datos. El parámetro indica que la medición ha finalizado. Los resultados de la medición se pueden leer.
5. El inicio de otra medición con el flanco ascendente de I1 desactiva el parámetro MSR_DONE.

Si durante una medición de longitud se desactiva la función, el FM 452 no actualiza los datos. El parámetro MSR_DONE se mantiene desactivado.

Captura de bordes



1. Dado el caso, introduzca un valor para la separación mínima de bordes en el DB de parámetros. Escriba y active los datos máquina.
2. Active el selector de función de "Captura de bordes". Se activa el parámetro MSR_DONE.
3. El flanco ascendente de la entrada I1 inicia la captura de bordes. Los resultados de la medición se actualizan y se pueden leer, el valor inicial de la medición se registra; el valor final y la longitud pasan a ser -1.
4. Tras la actualización, el FM 452 notifica el cambio desactivando el parámetro MSR_DONE.
5. El flanco descendente de la entrada I1 pone fin a una medición en curso. El FM 452 actualiza los datos del valor final de la medición y la longitud.
6. Tras la actualización, el FM 452 notifica el cambio activando el parámetro MSR_DONE. Los resultados de la medición se pueden leer.
7. El inicio de otra medición con el flanco ascendente de I1 desactiva el parámetro MSR_DONE.

Si durante una captura de bordes se desactiva la función, el FM 452 no actualiza los datos. El parámetro MSR_DONE se mantiene desactivado.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
25.1	MSR_DONE	BOOL	FALSE	1 = Medición de longitud cancelada
34.0	EDGE_ON	BOOL	FALSE	1 = Captura de bordes activada
34.2	MSR_ON	BOOL	FALSE	1 = Medición de longitud activada
38.2	MSRRD_EN	BOOL	FALSE	1 = Leer valores medidos
112.0	BEG_VAL	DINT	L#0	Valor inicial
116.0	END_VAL	DINT	L#0	Valor final
120.0	LEN_VAL	DINT	L#0	Longitud

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
4.0	EDGEDIST	DINT	L#0	**Distancia mínima entre bordes en la captura de bordes** Rango de valores admisible: • 0 ... 1 000 000 000 µm

En la captura de bordes, con la distancia mínima entre bordes se define un rango tras la detección del inicio de la medición. Si el final del proceso de medición se encuentra dentro de este rango, la medición se desestima.

El inicio de la medición se notifica en cuanto se ha recorrido la distancia mínima entre bordes.

Condiciones específicas para una medición de longitud

- La distancia entre el flanco de activación y el flanco de desactivación en la entrada I1 tiene que ser tal que su programa en la CPU pueda evaluar correctamente el resultado de la medición antes de iniciar otra.
- La distancia mínima entre el flanco ascendente y descendente en la entrada I1, al igual que entre el flanco descendente y el siguiente ascendente en la entrada I1, tiene que ser mayor que 2 ms.

Alarmas de proceso

El inicio y el fin de una medición pueden notificarse con alarmas de proceso en el FM 452.

Medición con errores

En caso de una medición de longitud/captura de bordes con errores, el FM 452 devuelve el valor -1 para la longitud.

Una "Medición de longitud" o "Captura de bordes" debería ejecutar un máximo de 126 cambios a cero en un sentido. Un cambio a cero es el cambio del eje rotativo del valor final del eje rotativo a 0 y a la inversa. En cuanto el FM 452 detecta más de 126 pasos por cero en un sentido, notifica una "Medición de longitud" o una "Captura de bordes" errónea, aunque luego haya detectado nuevamente pasos por cero en el otro sentido.

Una medición también es errónea si

- en un eje rotativo la longitud medida es superior a 2^{31} ,
- el FM 452 simultáneamente el flanco de activación y el de desactivación (p. ej. por rebotes del conector).

Este evento se puede comunicar como alarma de proceso.

Desplazamiento del sistema de coordenadas durante una medición de longitud

Los desplazamientos de coordenadas afectan a la longitud medida bajo las condiciones siguientes:

- Se utiliza un encóder incremental o un detector de proximidad, o bien el FM 452 en el modo de simulación.
- Se ejecuta "Definir punto de referencia" o "Redisparar punto de referencia" durante una medición de longitud.

Ejemplo

Es posible emplear la influencia arriba indicada de la longitud medida del modo que sigue:

Se dispone de un sistema en el que durante una medición de longitud siempre se produce un deslizamiento.

Con el "Redisparo del punto de referencia" es posible corregir este deslizamiento de forma que se emitan los valores medidos de longitud correctos.

Redisparar punto de referencia

Definición

Con la función "Redisparar punto de referencia" es posible sincronizar el eje en función de un evento externo repetitivo.

La función se mantiene activa hasta que se desactiva de nuevo.

Requisitos

- Se emplea un encóder incremental o un detector de proximidad.
- El evento externo puede ser una señal de paso por cero de un encóder incremental o un sensor del punto de referencia en la entrada I2.

Desarrollo de la función

1. Introduzca el valor de la coordenada del punto de referencia en el DB de parámetros.
2. Introduzca el tipo de "Redisparar el punto de referencia" en el DB de parámetros.
   Para ello, existen las posibilidades siguientes:
   - Sólo se evalúa la señal de paso por cero del encóder (RETR_TYPE = 7).
   - Sólo se evalúa el sensor del punto de referencia (RETR_TYPE = 6).
   - Se evalúa la señal de paso por cero
     en sentido positivo: Se evalúa el primer flanco ascendente de la señal de paso por cero tras abandonar el sensor del punto de referencia en sentido positivo (RETR_TYPE = 0).
     en sentido negativo: Se evalúa el primer flanco descendente de la señal de paso por cero tras abandonar el sensor del punto de referencia en sentido negativo (RETR_TYPE = 1).
3. Escriba y active los datos máquina.
4. Active el selector de función del DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
34.3	REFTR_ON	BOOL	FALSE	1 = Redisparar el punto de referencia
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	Coordenada del punto de referencia
52.0	RETR_TYPE	DINT	L#0	Forma de "Redisparar el punto de referencia"

Efectos de la función

- El FM x52 evalúa la señal de paso por cero y el sensor del punto de referencia en función del sentido del movimiento del eje.
  - En caso de un sentido positivo se evalúan los flancos ascendentes.
  - En caso de un sentido de movimiento negativo se evalúan los flancos descendentes.
- La posición real se fija en el valor de la coordenada del punto de referencia.
- La zona de trabajo se desplaza físicamente en el eje.
- Cada uno de los puntos conserva su valor original, pero se encuentran en posiciones físicas nuevas.
- Los cambios de estado de la leva que normalmente dispararían una alarma, se pueden perder.
- El bit SYNC de las señales de realimentación se activa.

Nota

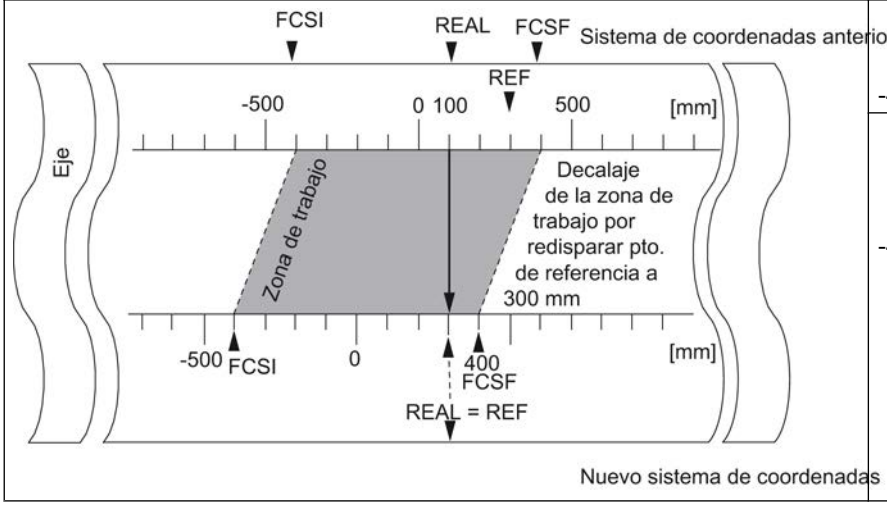
Tenga en cuenta el comportamiento de conmutación con las levas de tiempo (Página 9834).

Ejemplo

Para este ejemplo rige:

- Del sensor del punto de referencia y la señal de paso por cero se evalúan los flancos ascendentes (sentido positivo del eje).
- La coordenada del punto de referencia tiene el valor 300 mm.
- En el instante de la ejecución no hay activo ningún decalaje de señal cero.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Redisparar punto de referencia".

Redisparar punto de referencia	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-400	300	100	400
	-400	300	300	400

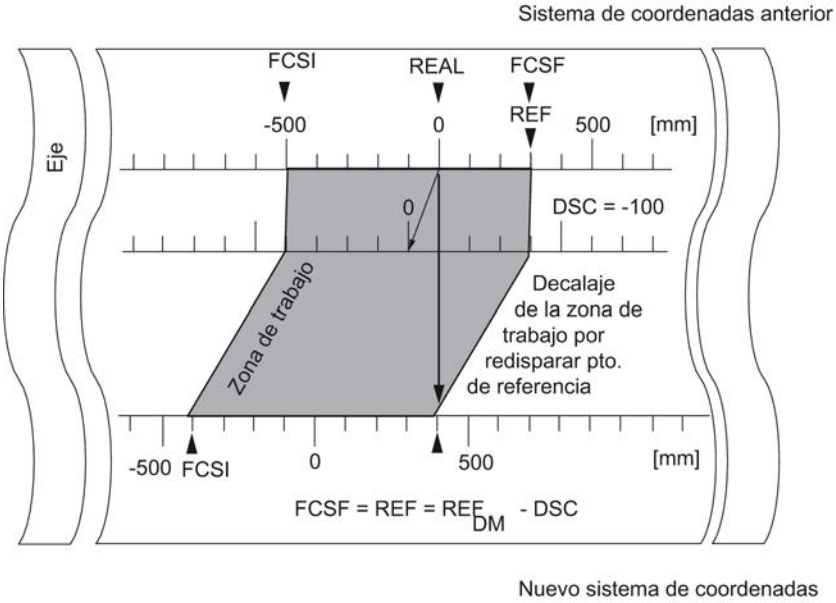
Consideración de un decalaje de señal cero

Cuando hay un decalaje de señal cero activo, se tiene en cuenta en la función "Redisparar punto de referencia". Esto significa que la coordenada del punto de referencia que se activará, se calculará conforme a la fórmula siguiente:

Ref = Ref_{DM} - Decalaje de señal cero

Ref_{DM} es el valor depositado en los datos máquina a modo de coordenada del punto de referencia.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Redisparar punto de referencia" con decalaje de señal cero.

Redisparar punto de referencia	FCSI [mm]	REF [mm]	REAL [mm]	FCSF [mm]
	-500	300	0	300
	-400	400	100	400
	-400	400	400	400

Desconectar el final de carrera por software

Definición

La función "Desconectar final de carrera por software" desactiva la vigilancia del final de carrera por software de un eje lineal.

La función se mantiene activa hasta que se desactiva de nuevo. Entonces se vuelven a activar los finales de carrera parametrizados originalmente.

Desarrollo de la función

Active el selector de función en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	Valor inicial	Comentarios
34.4	SSW_OFF	BOOL	FALSE	1 = Final de carrera por software desconectado

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	Valor inicial	Comentarios
64.0	SSW_STRT	DINT	L# -1000000000	Final de carrera por software Inicio
68.0	SSW_END	DINT	L#1000000000	Final de carrera por software Fin

Efectos de la función

- Simulación
  - Si en el modo de simulación se rebasa un final de carrera por software se detiene el modo de simulación.
  - Si desactiva entonces la vigilancia del final de carrera por software, el modo de simulación prosigue. El eje se desplaza en la dirección indicada.
- Decalaje de señal cero con vigilancia desactivada
  Si se define un decalaje de señal cero en el que los finales de carrera por software se encuentran todavía dentro de los límites de la zona de desplazamiento, el valor real puede encontrarse fuera del rango de contaje admisible.
- Las levas que se encuentran fuera del final de carrera por software parametrizado, se pueden activar.

⚠ PRECAUCIÓN
Se pueden producir daños materiales. Si limita la zona de desplazamiento con finales de carrera por software y protege con ello su instalación, al desconectar los finales de carrera por software se pueden producir daños materiales. Por ello, al planificar la instalación cuide que el accionamiento pueda desplazarse por toda la zona de desplazamiento física.

Ejecutar la "Simulación"

Definición

La función "Simulación" brinda la posibilidad de activar la leva electrónica sin un encóder conectado.

Desarrollo de la función

1. Introduzca la velocidad de simulación en el DB de parámetros.
2. Escriba y active los datos máquina.
3. En el DB de canal seleccione como sentido de la simulación el sentido positivo o negativo.
4. Active el selector de función del DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.2	DIR_M	BOOL	FALSE	1 = Simulación en el sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Simulación en el sentido positivo
34.1	SIM_ON	BOOL	FALSE	1 = Simulación activa

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
84.0	SIM_SPD	DINT	L#0	Velocidad de simulación

Efectos al activar la simulación

- Las señales de encóder dejan de evaluarse.
- Todas las supervisiones referidas a la entrada del encóder se desconectan.
- Los errores que se hayan notificado con respecto al encóder, se desactivan.
- El FM x52 simula el movimiento de un eje a una velocidad de simulación constante.
- El procesamiento de levas se desactiva con la activación de la simulación. Sin embargo, es posible activarlo de nuevo. La sincronización se mantiene.
- El valor real de la posición cambia a partir del valor real actual, dependiendo de la velocidad y del sentido de la simulación.

Efectos al desactivar la simulación

- El procesamiento de levas se desactiva.
- En caso de un encóder incremental o un detector de proximidad, la sincronización se borra. Entonces como valor real se define el valor de la coordenada del punto de referencia.
- En el caso de un encóder absoluto se notifica el valor real de la posición, que se corresponde con la posición actual del encóder. Las señales del encóder se evalúan de nuevo tal como se han parametrizado en los datos máquina.

Valores límite

Los límites máximo y mínimo de la velocidad de simulación dependen de la resolución (Página 9873).

Velocidad

Entre la velocidad definida y la velocidad real (Página 9873) a la que funciona el módulo, puede haber diferencias.

Leer "Valores de contaje de las pistas de leva de contaje"

Definición

La función "Valores de contaje de las pistas de leva de contaje" permite leer los valores de contaje actuales.

Desarrollo de la función

1. Defina las pistas de la leva de contaje y los valores de contaje superiores en los datos máquina.
2. Escriba y active los datos máquina.
3. Habilite la función de contaje.
4. El valor de contaje se fijará en el valor de contaje superior.
5. Con cada flanco ascendente del resultado de la pista el valor de contaje se reducirá en 1.
6. Active el bit de inicio en el DB de canal para la lectura de los valores de contaje.
7. Los valores de contaje de las dos pistas se escriben en el DB de canal. Para una pista no parametrizada como pista de leva de contaje se emite 0.
8. Si el valor de contaje alcanza el valor 0, el bit identificador de la pista de leva de contaje se pone a 1.
9. Con el siguiente flanco descendente del resultado de pista, el bit identificador vuelve a 0 y el contador se pone al valor de contaje superior.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.5	CNTC0_EN	BOOL	FALSE	1 = Habilitar la función de contaje de la pista de leva de contaje 0
15.6	CNTC1_EN	BOOL	FALSE	1 = Habilitar la función de contaje de la pista de leva de contaje 1
38.3	CNTTRC_EN	BOOL	FALSE	1 = Leer valores de contaje de la pista de leva de contaje
124.0	CNT_TRC0	INT	0	Valor de contaje actual para la pista de leva de contaje 0
126.0	CNT_TRC1	INT	0	Valor de contaje actual para la pista de leva de contaje 1

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
99.0	SPEC_TRC0	BOOL	FALSE	1 = Pista 0 es pista de leva de contaje
99.1	SPEC_TRC1	BOOL	FALSE	1 = Pista 1 es pista de leva de contaje
100.0	CNT_LIM0	DINT	L#2	Valor de contaje superior para la pista de leva de contaje 0
104.0	CNT_LIM1	DINT	L#2	Valor de contaje superior para la pista de leva de contaje 1

Leer "Datos de posición y pista"

Definición

Con la función "Datos de posición y pista" se lee la posición, la velocidad y el bit identificador de pista en ese instante. Los bits identificadores de pista se capturan antes de combinarlos con los datos máquina y de canal.

El algoritmo de cálculo implantado en el FM x52 calcula cambios de velocidad mayores que 1 impulso por 4 ms. La velocidad que se muestra, por lo tanto, incluye imprecisión y no es conveniente para fines de cálculo. La velocidad interna que se emplea para dinamizar las levas es más precisa.

Desarrollo de la función

1. Active el bit de inicio en el DB de canal.
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
38.4	ACTPOS_EN	BOOL	FALSE	1 = Leer Datos de posición y pista
128.0	ACTPOS	DINT	L#0	Posición actual
132.0	ACTSPD	DINT	L#0	Velocidad actual
136.0	TRACK_ID	DWORD	DW#16#0	Bits identificadores de pista de las pistas 0 a 31

Leer "Datos de encóder"

Definición

Con la función "Datos de encóder" se leen los datos actuales del encóder así como el valor del reajuste del encóder absoluto.

Requisitos

Es posible leer el valor para el reajuste del encóder absoluto (Página 9868) después de ejecutar la función "Definir punto de referencia (Página 9840)".

Desarrollo de la función

1. Active el bit de inicio en el DB de canal.
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
38.5	ENCVAL_EN	BOOL	FALSE	1 = Leer valores del encóder
140.0	ENCVAL	DINT	L#0	Valor de encóder / Lectura del contador (representación interna)
144.0	ZEROVAL	DINT	L#0	Lectura del contador en la última señal de paso por cero (representación interna)
148.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto

Leer los "Datos de leva y pista"

Definición

Con la función "Datos de leva y pista" se leen los bits identificadores de leva y de pista actuales así como la posición en ese instante. Los bits identificadores de pista se capturan antes de combinarlos con los datos máquina y de canal.

Desarrollo de la función

1. Introduzca como identificador de tipo un 1 en el parámetro FM_TYPE del DB de canal. Con ello es posible leer los datos de leva y de pista de 24 bytes.
   Si introduce 0 como identificador de tipo sólo se leerán los bits identificadores de la leva (16 byte).
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
12.0	FM_TYPE	BOOL	FALSE	0 = FM 352 a V4.0 1 = FM 352 a partir de V5.0 y FM 452
38.6	CAMOUT_EN	BOOL	FALSE	1 = Leer datos de leva y pista
152.0	CAM_00_31	DWORD	DW#16#0	Bits identificadores de la leva para las levas de 0 a 31
156.0	CAM_32_63	DWORD	DW#16#0	Bits identificadores de leva para las levas de 32 a 63
160.0	CAM_64_95	DWORD	DW#16#0	Bits identificadores de la leva para las levas de 64 a 95
164.0	CAM_96_127	DWORD	DW#16#0	Bits identificadores de la leva para las levas de 96 a 127
168.0	TRACK_ID1	DWORD	DW#16#0	Bits identificadores de pista de las pistas 0 a 31
172.0	ACTPOS1	DINT	L#0	Posición actual

Activar "Señales de control de la leva electrónica"

Definición

Con la función "Señales de control de la leva electrónica" se habilita el procesamiento de levas así como las pistas.

Proceso de la función

1. Active los bits que desee en el DB de canal.
2. Con cada llamada de la instrucción CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) los datos se transfieren al módulo.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.4	CAM_EN	BOOL	FALSE	1 = Habilitar el procesamiento de levas
16.0	TRACK_EN	WORD	W#16#0	Habilitar pistas de leva de 0 a 12 Bit 0 = Pista 0

Efectos

El procesamiento de levas se inicia o se detiene según la habilitación.

Los bits identificadores de pista de las pistas habilitadas se transfieren a las señales de pista y a las salidas digitales.

Consultar "Señales de realimentación de la leva electrónica"

Definición

Con la función "Señales de respuesta de la leva electrónica" se informa del estado actual de la leva y de las señales de pista. La coherencia entre la posición indicada y las señales de pista no está asegurada.

Proceso de la función

Con cada llamada de la instrucción CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
23.4	CAM_ACT	BOOL	FALSE	1 = Procesamiento de levas activo
26.0	ACT_POS	DINT	L#0	Posición actual del eje
30.0	TRACK_OUT	DWORD	DW#16#0	Señales de pista actuales de las pistas 0 a 31 Bit 0 = Pista 0

Consultar "Señales de realimentación del diagnóstico"

Definición

Con el ajuste "Señales de realimentación del diagnóstico" se recibe información sobre los eventos de diagnóstico que se han producido.

Proceso de ajuste

1. Cuando el módulo realiza una nueva entrada en el búfer de diagnóstico, activa el bit DIAG en la interfaz de realimentación. Cada aparición de un error (Página 9906) provoca una entrada en el búfer de diagnóstico.
2. Cuando el módulo detecta una petición de escritura con datos erróneos, activa el bit DATA_ERR en la interfaz de realimentación. La causa del error se registra en el búfer de diagnóstico.
3. La instrucción CAM_CTRL (Página 7453) o CAM_CTRL_452 (Página 7457) coloca la interfaz de realimentación en el DB de canal.
4. Una vez que la instrucción CAM_DIAG (Página 7455) o CAM_DIAG_452 (Página 7458) haya leído el búfer de diagnóstico, el módulo restablece el bit DIAG a 0.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
22.2	DIAG	BOOL	FALSE	1 = Búfer de diagnóstico modificado
22.4	DATA_ERR	BOOL	FALSE	1 = Error de datos

Configurar FM 452

Parametrizar el FM 452

Sistema de medición

Seleccionar sistema de medidas

Seleccionar sistema de medidas

Para la entrada y salida de datos es posible elegir entre los siguientes sistemas de medida:

- Milímetros (ajuste predeterminado)
- Inch
- Grados
- Impulsos

Nota

Si se modifica el sistema de medida en el cuadro de diálogo "Sistema de medida", los valores se convierten al nuevo sistema. Aquí se pueden producir errores de redondeo.

Si se modifica el sistema de medida mediante el dato máquina, los valores no se convierten automáticamente.

Si el sistema de medida cambia de o a "impulsos", el procesamiento de levas se desactivará y el eje dejará de estar sincronizado.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentario
8.0	UNITS	DINT	L#1	**Sistema de medida** Valores permitidos: • 1 = 10^{-3} mm • 2 = 10^{-4} pulgadas • 3 = 10^{-4} grados • 4 = 10^{-2} grados • 5 = Impulsos • 6 = 10^{-3} grados

Sistema de medida estándar en la ayuda en pantalla

Para la indicación de valores límite se utiliza el sistema de medida mm. Por ello, para determinar los valores límite en los otros sistemas de medida, realice la conversión siguiente:

Para convertir de		calcule
mm → pulgadas		Valor límite (inch) = valor límite (mm) × 0,1 [1]
mm → grados	10^{-4} (4 decimales)	Valor límite (grados) = valor límite (mm) × 0,1
	10^{-3} (3 decimales)	Valor límite (grados) = valor límite (mm) × 1
	10^{-2} (2 decimales)	Valor límite (grados) = valor límite (mm) × 10
mm → impulsos		Valor límite (impulsos) = valor límite (mm) x 1000

[1] En el valor máximo el número de decimales repercute en el número de dígitos que preceden a la coma. En el sistema de medida Inch se utilizan 4 decimales, por lo tanto se pueden introducir como máximo 100 000,0000 inch. El sistema de medida Milímetros utiliza 3 decimales y por lo tanto se pueden introducir como máximo 1 000 000,000 mm.

Eje

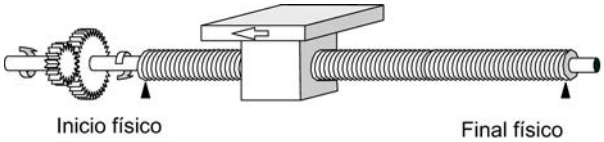
Definir tipo de eje

Seleccionar tipo de eje

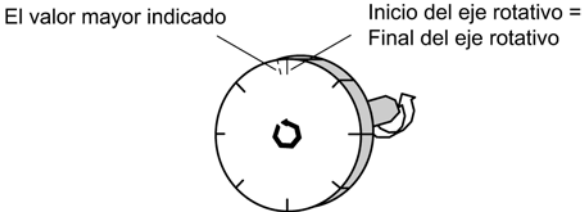
Tipos de eje

La leva electrónica soporta los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento física limitada.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
12.0	AXIS_TYPE	DINT	L#0	**Tipo de eje** Valores permitidos: • 0 = Eje lineal • 1 = Eje rotativo

Definir final de carrera por software

Final de carrera por software

Los finales de carrera por software existen sólo para el eje lineal.

El área limitada por el final de carrera por software es la zona de trabajo. Los límites de la zona de trabajo pueden ser vigilados por el FM x52.

El inicio del final de carrera por software (FCSI) debe ser siempre menor que el fin del final de carrera por software (FCSF).

Los finales de carrera por software están activos cuando el FM x52 está sincronizado.

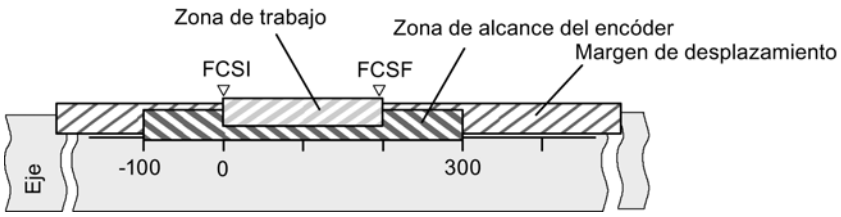
- Si se utiliza un encóder incremental o un detector de proximidad:
  Después de cada arranque del FM x52 el eje no está sincronizado. Sólo tras una sincronización se vigilan los finales de carrera por software parametrizados.
- Si se utiliza un encóder absoluto (SSI):
  El FM x52 está sincronizado después de haber recibido un telegrama completo y sin errores. A partir de este instante los finales de carrera por software son vigilados. El encóder absoluto utilizado tiene que abarcar por lo menos la zona de trabajo, incluido el final de carrera por software.

Relación entre la zona de trabajo, del encóder y la zona de desplazamiento

- La zona de trabajo es el área que el usuario define para sus tareas mediante los finales de carrera por software.
- El zona del encóder es el área que el encóder cubre claramente. En el caso de un eje lineal del módulo, el área del encóder se define de forma simétrica por encima de la zona de trabajo. Es decir, el módulo desplaza la zona del encóder de tal modo que las distancias entre los finales de carrera por software y los finales de la zona del encóder son iguales en cada caso (véase la figura en la parte de abajo).
- La zona de desplazamiento es el rango de valores que puede procesar el FM x52. La zona de desplazamiento depende de la resolución.

Se debe considerar: Zona de desplazamiento ≥ Rango del encóder ≥ Zona de trabajo

La figura siguiente muestra la relación entre los finales de carrera por software, la zona de trabajo, el rango del encóder y la zona de desplazamiento.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
64.0	SSW_STRT	DINT	L# -100.000.000	**Final de carrera por software Inicio**
68.0	SSW_END	DINT	L# 100.000.000	**Final de carrera por software Fin** Rango de valores admisible: • -1 000 000 000 µm a 1 000 000 000 µm

Definir final del eje rotativo

Final del eje rotativo

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar. Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo (0).

El valor más grande que se muestra en el caso de un eje rotativo es:

Final del eje rotativo [µm] - Resolución [µm/impulso] × 1 [impulso]

Ejemplo: Final del eje rotativo 1000 mm

La visualización pasa:

- Con un sentido de giro positivo, de 999 mm a 0 mm.
- con un sentido de giro negativo, de 0 mm a 999 mm.

Eje rotativo con encóders absolutos

En el caso de un eje rotativo con un encóder absoluto, el área del eje rotativo (0 hasta el final del eje rotativo) tiene que coincidir exactamente con el total de pasos del encóder absoluto.

$$\text{Fin de eje rotativo[µm]} = \text{número de vueltas(encoder)} * \frac{\text{Pasos(encoder)[pul]}}{\text{Vuelta}} * \text{RES}[\frac{\text{µm}}{\text{Pulso}}]$$

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
16.0	ENDROTAX	DINT	L#100000	**Final del eje rotativo** Rango de valores admisible: • 1 µm a +1 000 000 000 µm

Definir sincronización

Seleccionar la "Forma de redisparar el punto de referencia"

Redisparar el punto de referencia

Con la forma de "Redisparar el punto de referencia (Página 9849)" se definen las condiciones de sincronización del eje para el funcionamiento con un encóder incremental o un detector de proximidad.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
52.0	RETR_TYPE	DINT	L#0	**Forma de "Redisparar el punto de referencia"** Valores permitidos: • 0 = sensor del punto de referencia y señal de paso por cero en sentido + • 1 = sensor del punto de referencia y señal de paso por cero en sentido - • 6 = sólo sensor del punto de referencia • 7 = sólo señal de paso por cero

Definir la coordenada del punto de referencia

Coordenada del punto de referencia

- **Encóder incremental y detector de proximidad:**
  Con el selector de función "Redisparar punto de referencia" y un evento de sincronización, definido por la forma de "Redisparar el punto de referencia" (Página 9864), se asigna a este evento la coordenada del punto de referencia.
- **Encóder absoluto (SSI)**
  Un eje parametrizado con un encóder absoluto siempre está sincronizado después de la transmisión del primer telegrama SSI sin errores.
  Consulte al respecto también la descripción del reajuste del encóder absoluto (Página 9868), en la que se explica la interacción de dicho ajuste con el resto de datos.

El valor de la coordenada del punto de referencia se encuentra dentro de la zona de trabajo:

- **Eje lineal**
  Incluido el final de carrera por software.
- **Eje rotativo**
  Mayor o igual que 0 y menor que el valor "Final del eje rotativo" (0 ≤ coordenada del punto de referencia < final del eje rotativo).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1.000.000.000 µm a +1.000.000.000 µm

Definir reajuste del encóder absoluto

Calcular reajuste del encóder absoluto

Definición

El reajuste del encóder absoluto sólo es necesario para un encóder absoluto (SSI).

Con el reajuste del encóder absoluto y las coordenadas del punto de referencia el rango de valores del sensor se reproduce claramente en el sistema de coordenadas del eje. Se calcula el valor del encóder que equivale a la coordenada del punto de referencia en el eje. El valor tiene que ser menor que el total de pasos del encóder absoluto.

Encontrará más información al respecto en "Procedimiento de reajuste del encóder absoluto (Página 9869)", así como en un ejemplo (Página 9870).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
48.0	ENC_ADJ	DINT	L#0	**Reajuste del encóder absoluto** Rango de valores admisible: • 0 a (2^{25} - 1)

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
98.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1.000.000.000 µm a +1.000.000.000 µm

Procedimiento para un reajuste del encóder absoluto

Determinar el reajuste correcto del encóder absoluto

Tras la primera parametrización hay que llevar a cabo otros pasos para que se establezca una relación correcta entre el encóder y el sistema de coordenadas. El proceso se representa utilizando el cuadro de diálogo de parametrización.

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Fin del final de carrera por software".
2. Llame el ajuste "Definir punto de referencia (Página 9840)" con la coordenada del punto definido en 1.
   El FM x52 determina entonces un valor de encóder para la coordenada del punto de referencia registrada en el eje en el DB de canal (REFPT): el reajuste del encóder absoluto. Este valor se puede leer con el ajuste "Datos de encóder (Página 9856)".
3. Registre el valor leído en el campo "Reajuste del encóder absoluto".
4. Guarde la parametrización en el DB de parámetros correspondiente con la función de exportación.
5. Cargue los datos de HWCN en la CPU.
6. Para activar los datos, rearranque la CPU.

Nota

Esta sincronización se realiza una vez en la puesta en marcha. Tras una parametrización, el FM x52 queda sincronizado durante el arranque en cuanto se recibe un telegrama sin errores del encóder.

Alternativa: Reajuste mecánico de un encóder

También se puede obtener una relación correcta entre el sistema de coordenadas y el encóder del modo siguiente:

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Inicio del final de carrera por software".
2. Introduzca este valor de coordenadas en los datos máquina a modo de "coordenada del punto de referencia".
3. Lea el valor de encóder que se muestra en esta posición con el ajuste "Datos de encóder (Página 9856)".
4. Introduzca este valor como reajuste del encóder absoluto en los datos máquina.

Tras la parametrización se mostrará siempre un valor real correcto.

En lugar del paso 3 y 4 también es posible resetear a cero el codificador a través de la opción "Reset", si está disponible. A continuación, introduzca el valor "0" como reajuste del encóder absoluto en los datos máquina.

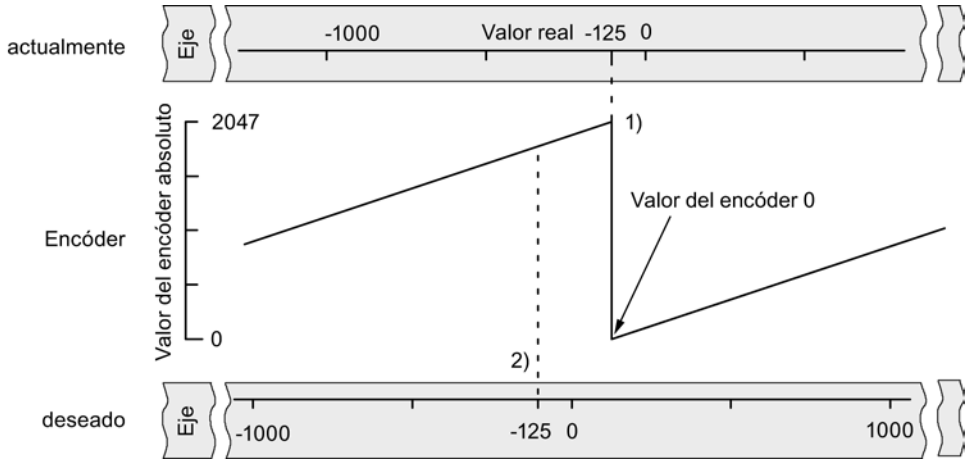
Ejemplo para un reajuste del encóder absoluto

Ejemplo para un reajuste del encóder absoluto

Para este ejemplo se parte de las premisas siguientes:

- Coordenada del punto de referencia = -125 mm
- Área de trabajo de SSW_STRT = -1.000 mm a SSW_END = 1.000 mm
- Reajuste del encóder absoluto = 0
- Área del encóder = 2.048 incrementos con una resolución de 1 mm/impulso
- El encóder absoluto utilizado no permite un ajuste mecánico exacto y tampoco ofrece la posibilidad de definir el valor del encóder.

La figura siguiente muestra un ejemplo de reajuste del encóder absoluto.

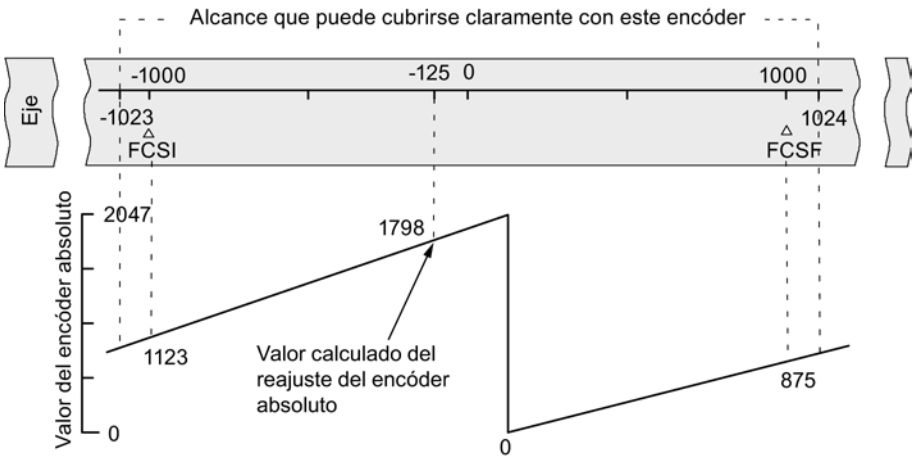


1) Asignación del sistema de coordenadas a los valores del sensor con el reajuste de encóder absoluto definido. El valor del encóder 0 equivale al valor real -125.
2) Asignación deseada del sistema de coordenadas con respecto al encóder. En esta posición, la coordenada tiene que ser -125.

Resultado tras "Definir el punto de referencia"

Tras "Definir el punto de referencia", la relación es la siguiente:

La coordenada del punto de referencia en el eje (-125) se asigna al valor del encóder calculado (1798) a partir del reajuste del encóder absoluto. La figura siguiente muestra esta asignación.



El encóder devuelve 2.048 valores unívocos. La zona de trabajo está definida por el final de carrera por software. Sin embargo, debido a la resolución seleccionada de 1 mm por impulso, el encóder puede abarcar una zona de trabajo mayor que la prevista por el final de carrera por software.

Con la resolución definida, la zona de trabajo está ya cubierta con 2.001 valores. Por ello en el ejemplo quedan 47 impulsos "sobrantes" dispuestos simétricamente alrededor de la zona de trabajo.

Definir histéresis

Histéresis

El rango de valores de la histéresis (Página 9740) depende de la resolución (Página 9877).

El valor máximo de entrada es como sigue:

- En el caso de ejes lineales: Valor de entrada máximo < ¼ de la zona de trabajo
- En el caso de los ejes rotativos: Valor de entrada máximo < ¼ del área del eje rotativo

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
80.0	HYS	DINT	L#0	**Histéresis** Rango de valores admisible: • 0 … 65.535 [Imp] × resolución [$^{\mu m}/_{Imp}$]

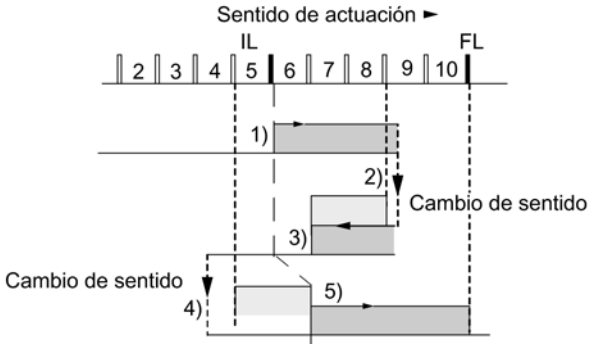
Leva de recorrido con histéresis

Una leva de recorrido se acciona cuando se cumplen las siguientes condiciones:

- el valor real detectado se encuentra dentro de la leva de recorrido
- la histéresis no está activada.

Según la posición en la que tiene lugar el cambio de sentido, puede variar la posición de un punto de cambio.

La siguiente figura muestra el comportamiento de conmutación de una leva de recorrido en el caso de una histéresis.



	Leva
	Histéresis
1)	La leva se activa porque el principio de leva se rebasa en sentido de actuación positivo.
2)	A pesar del cambio de sentido, la leva permanece activada debido a la histéresis.
3)	La leva se desactiva al final de la histéresis.
4)	La leva permanece desactivada tras el cambio de sentido.
5)	Tras el final de la histéresis la leva se activa nuevamente.

Nota

Las levas de recorrido que son más cortas que la histéresis no pueden ser acortadas por la histéresis en caso de un cambio de sentido.

Levas de tiempo con histéresis

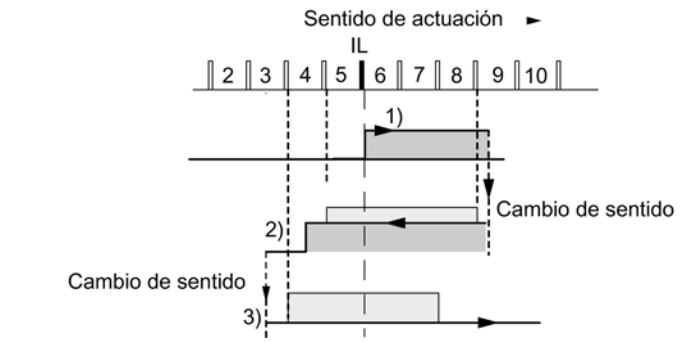
La leva de tiempo se acciona cuando se cumplen las siguientes condiciones:

- El principio de leva se rebasa en el sentido de actuación.
- la histéresis no está activada.

Nota

Si el área comprendida entre el punto de inversión y el principio de leva de tiempo es menor que la histéresis la leva de tiempo quedará oculta por la histéresis.

La siguiente figura muestra el comportamiento de conmutación de una leva de tiepo en el caso de una histéresis. Se representa una leva de tiempo que **no** se activa de nuevo.



	Leva
	Histéresis
1)	La leva se activa porque el principio de leva se rebasa en sentido de actuación positivo.
2)	Tras el cambio de sentido y hasta que ha transcurrido el tiempo de conexión la leva se mantiene activada, independientemente de la histéresis.
3)	El principio de leva queda oculto por la histéresis, la leva no se activa.

Definir velocidad de simulación

Velocidad de simulación

Este dato máquina define la velocidad de la "simulación". Tenga en cuenta a este respecto la información incluida en "Influencia de los ajustes en el comportamiento de conmutación de las levas de tiempo (Página 9834)".

La velocidad de simulación real V_{sim} puede diferir de la velocidad de simulación introducida $V_{sim, V}$ y se calcula conforme a la fórmula siguiente:

$$V_{Sim} = \frac{6*10exp7*\ RES}{\text{Entero}\left(\frac{6*10exp7*\ RES}{V_{Sim,\ V}}\right)}$$

En esta fórmula:

- V_{sim}: es la velocidad de simulación definida por el FM 352. Unidad: $^{\mu m}/_{min}$
- $V_{sim, V}$: es la velocidad de simulación que el usuario define en los datos máquina. Unidad: $^{\mu m}/_{min}$
- RES: Resolución calculada a partir de los datos del encóder. Unidad: $^{\mu m}/_{Impulso}$
- Entero ( ): a partir de esta expresión, sólo se tiene en cuenta el valor previo al decimal para otros cálculos. Esta expresión tiene que encontrarse dentro del rango de 2 ... 65.536 en todos los cálculos.

Las diferentes relaciones (véase la fórmula anterior) provocan cambios inesperados en la velocidad de simulación real.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
84.0	SIM_SPD	DINT	L#0	**Velocidad de simulación** Rango de valores admisible: • 0 = Parada • 5×10^8 = Ajuste máximo del módulo Dentro de este rango la velocidad de simulación depende de la resolución (Página 9877): 1000 × resolución ≤ velocidad de simulación ≤ 3×10^7 × resolución

Definir distancia mínima entre bordes

Distancia mínima entre bordes

En una captura de bordes, con este dato máquina se define un área tras detectar el inicio de la medición. Si el final del proceso de medición se encuentra dentro de este rango, la medición se desestima.

El inicio de la medición se notifica en cuanto se ha recorrido la distancia mínima entre bordes.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
4.0	EDGEDIST	DINT	L#0	**Distancia mínima entre bordes** Rango de valores admisible: • 0 ... 1.000.000.000 µm

Encóder

Seleccionar tipo de señal

Tipo de señal

El encóder da información de recorrido al módulo, el cual la evalúa y la convierte en un valor real con la resolución.

Es posible seleccionar un tipo de señal de uno de los encóders (Página 9744) siguientes:

- Señal diferencial de 5 V (Página 9745)
  Seleccione el tipo de señal "5 V Señal diferencial" si utiliza un encóder incremental de 5 V con señales diferenciales A, /A, B, /B y N, /N.
- 24 V Pista A+B desfasado (Página 9745)
  Seleccione el tipo de señal "24 V Pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A*, B* y N*.
- SSI absoluto (Página 9747)
  Seleccione el tipo de señal "SSI absoluto" si desea utilizar un encóder absoluto con una interfaz en serie.
- 24 V Detector de proximidad (Página 9747)
  Seleccione el tipo de señal "Detector de proximidad de 24 V" si desea utilizar un detector de 24V con la señal A*.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
20.0	ENC_TYPE	DINT	L#1	**Tipo de señal** Valores permitidos: • 1 = 5 V incremental • 2 = 24 V incremental • 3 = SSI longitud del telegrama 13 bits • 4 = SSI longitud del telegrama 25 bits • 5 = Escuchar • 6 = 24 V Detector de proximidad ascendente (positivo) • 7 = 24 V Detector de proximidad descendente (negativo) • 8 = SSI longitud del telegrama 13 bits (alineación derecha) • 9 = SSI longitud del telegrama 25 bits (alineación derecha) • 10 = Escuchar (alineación derecha)

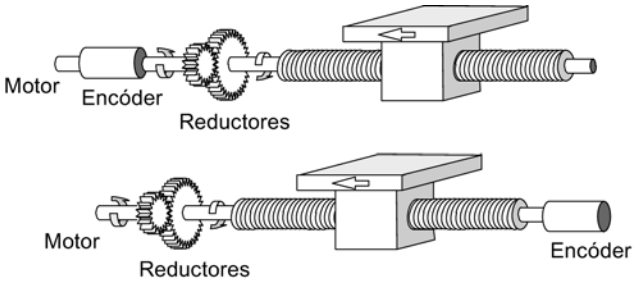
Definir resolución

Definir el recorrido por vuelta del encóder

recorrido por vuelta del encóder

Con el dato máquina "Recorrido por vuelta del encóder" se indica al FM x52 qué recorrido realiza el sistema de accionamiento en cada vuelta.

El valor "Recorrido por vuelta del encóder" depende de la estructura del eje y del modo en que está colocado el encóder. Debe tener en cuenta todos los elementos de transmisión, como acoplamientos o engranajes. La figura siguiente muestra dos disposiciones de encóder distintas a modo de ejemplo.



La relación entre los datos máquina "Recorrido por vuelta del encóder" e "Incrementos por vuelta del encóder" se describe en el apartado Resolución (Página 9877).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
24.0	DISP_REV	DINT	L#80000	**Recorrido por vuelta del encóder** Rango de valores admisible: • 1 µm a 1.000.000.000 µm

Definir los incrementos por vuelta del encóder

Incrementos por vuelta del encóder

El dato máquina "Incrementos por vuelta del encóder" indica la cantidad de incrementos que proporciona un encóder en cada vuelta. A partir de este valor y del dato máquina "Recorrido por vuelta del encóder", el FM x52 calcula la resolución.

Encóder incremental

Como entrada (Página 9745) es posible indicar cualquier valor dentro del rango de valores. El módulo evalúa los incrementos en múltiplos de 4.

Detector de proximidad

Es posible indicar cualquier valor dentro del rango de valores.

Encóder absoluto

Los límites varían según el tipo de encóder:

Tipo de encóder	Longitud/tipo de telegrama	Valores posibles	utilizable como eje lineal
Encóder monovuelta	13 bits medio abeto	64 ... 8192 en potencias de 2	
Encóder monovuelta	13 bits alineación derecha	64 ... 8192 todos los valores	X
Encóder monovuelta	25 bits alineación derecha	64 ... 2^{25} todos los valores	X
Encóder multivuelta	25 bits abeto	64 ... 8192 en potencias de 2	

Tipo de encóder	Longitud/tipo de telegrama	Valores posibles	utilizable como eje lineal
Encóder multivuelta	25 bits alineación derecha	64 ... 2^{24} todos los valores	
Escuchar	Abeto	64 ... 8192 en potencias de 2	
Escuchar	alineación derecha	64 ... 2^{25} todos los valores	X
Ajuste especial: Encóder multivuelta como encóder monovuelta	25 bits medio abeto	64 ... 8192 en potencias de 2	

Nota

La cantidad de impulsos de un encóder se calcula a partir de los dos datos "Incrementos por vuelta del encóder" multiplicado por "Número de vueltas" (véase el apartado "Resolución (Página 9877)").

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
32.0	INC_REV	DINT	L#500	**Incrementos por vuelta del encóder** Rango de valores admisible: • 1 a 2^{25} Atención: En el sistema de medida "Impulsos" esta entrada carece de importancia.

Resolución

Resolución

La resolución es una medida para la precisión del procesamiento de levas. Determina también la zona de desplazamiento máxima posible.

Para calcular la resolución (RES) se procede tal como se indica en la tabla siguiente:

	Encóder incremental	Encóder absoluto/detector de proximidad
Valores de entrada	• Recorrido por vuelta • Incrementos por vuelta del encóder: – Evaluación de impulsos: Cuádruple – 1 Incremento = 4 Impulsos	• Recorrido por vuelta del encóder • Incrementos por vuelta del encóder. – 1 Incremento = 1 Impulso
Cálculo	$RES = \frac{\frac{\text{Trayecto}}{\text{Vuelta del encóder}}}{\frac{\text{Impulsos}}{\text{Vuelta del encóder}}}$	

Nota

En el sistema de medida "Impulsos" la resolución siempre tiene el valor 1.

Todos los datos de posición se redondean al múltiplo entero de la resolución. De este modo es posible diferenciar entre los valores introducidos y los utilizados.

Rango de valores de la resolución

Es preciso convertir el rango de valores admitidos de la resolución al sistema de medida subyacente. Con la elección de los dos valores "Recorrido por vuelta" e "Incrementos por vuelta" la resolución se define dentro de este rango de valores.

El sistema de medida seleccionado determina el margen de valores de la resolución:

Sistema de medición	Datos en ...	Rango de valores de la resolución
mm	10^{-3} mm	$0,1 \times 10^{-3}$ mm .... 1000×10^{-3} mm/impulso
pulgadas	10^{-4} pulgadas	$0,1 \times 10^{-4}$ pulgadas .... 1000×10^{-4} pulgadas/impulso
grados	10^{-4} grados 10^{-3} grados 10^{-2} grados	$0,1 \times 10^{-4}$ grados .... 1000×10^{-4} grados/impulso $0,1 \times 10^{-3}$ grados .... 1000×10^{-3} grados/impulso $0,1 \times 10^{-2}$ grados .... 1000×10^{-2} grados/impulso
Impulsos	1 impulso	1

Ejemplo

- Un encóder incremental tiene los datos siguientes:
  - Incrementos por vuelta: 5000
  - Recorrido por vuelta del encóder: 1000 mm
  - 1 incremento = 4 impulsos

  De ello resulta la resolución (evaluación cuádruple):

$$\text{Resolución} = \frac{1000\ \text{mm}}{5.000\ \text{incrementos}} = 0{,}2000\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{impulsos}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impulso}}$$

- Un encóder SSI tiene los datos siguientes:
  - Incrementos por vuelta: 4096
  - Recorrido por vuelta del encóder: 1000 mm
  - 1 incremento = 1 impulso

  De ello resulta la resolución:

$$\text{Resolución} = \frac{1000\ \text{mm}}{4096\ \text{incrementos}} = 0{,}2441\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2441\ \frac{\text{mm}}{\text{Impulso}}$$

Dependencia entre la zona de desplazamiento y la resolución

La zona de desplazamiento está limitada por la representación numérica en el FM x52. Esta representación numérica varía en función de la resolución. Por ello, al definir valores hay que procurar mantenerse siempre dentro de los límites admitidos.

La siguiente tabla muestra la zona de desplazamiento máxima:

Resolución (RES) dentro de la zona	Zona de desplazamiento máxima
0,1 µm/impulso ≦ RES < 1 µm/impulso	-10^8 µm a 10^8 µm (-100 m a +100 m)
1 µm/impulso ≦RES ≦ 1000 µm/impulso	-10^9 µm a 10^9 µm (-1000 m a +1000 m)

Dependencia de la velocidad de la resolución

La velocidad que se muestra puede oscilar dentro de los márgenes que se indican según sea la resolución (los datos se expresan en el sistema de medida mm):

- de 1 µm/min a 90 m/min a una resolución < 1 µm/impulso
- de 1 µm/min a 900 m/min a una resolución ≥ 1 µm/impulso

La velocidad es calculada y suavizada por el módulo cada 4 ms.

Tiene por lo menos una imprecisión de un impulso/4 ms y no resulta adecuada para fines de regulación.

Seleccionar sentido de contaje

Dirección de contaje

Con el dato máquina "Dirección de contaje" se adapta la dirección de la captura del recorrido al sentido del movimiento del eje. Para ello, tenga en cuenta todos los sentidos de giro de los elementos de transmisión (como p. ej., acoplamientos y engranajes).

- normal = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real ascendentes
- invertida = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real descendentes

Nota:
No se permite un tiempo de anticipación en combinación con un encóder absoluto (SSI) y con sentido de contaje invertido.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	Valor inicial	Comentario
59.0	CNT_DIR	BOOL	FALSE	**Sentido de contaje** Valores permitidos: • 0 = normal • 1 = invertida

Seleccionar vigilancia

Vigilancia

Rotura de hilo

Con la activación de la vigilancia, el FM x52 vigila, en el caso de un encóder incremental, las señales A, /A, B, /B, N y /N. La vigilancia detecta:

- Rotura de hilo
- Cortocircuito de los distintos cables
  En los encóders incrementales sin marcas cero, como alternativa tiene que
  - desconectar la vigilancia de rotura de hilo
  - interconectar externamente las señales N y /N (Página 9744)
- Distancia de flancos de los impulsos de contaje
- Corte de la alimentación del encóder

Error del telegrama

La monitorización de errores de telegrama en los encóders absolutos (SSI) no puede desactivarse. Supervisa:

- el telegrama para ver si hay errores del bit de inicio y del bit de parada
- el tiempo monoestable del encóder conectado

Impulsos erróneos (encóder incremental)

Entre dos marcas cero consecutivas, un encóder incremental siempre tiene que proporcionar el mismo número de incrementos.

El FM x52 comprueba si la marca cero de un encóder incremental aparece en el estado adecuado del encóder.

En el caso de encóders sin marca cero es preciso desactivar la vigilancia de los impulsos de error.

Igualmente es preciso desactivar la vigilancia de rotura de hilo o bien interconectar las entradas de marca cero N y /N de forma externa.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	Valor inicial	Comentarios
				Vigilancias
63.0	MON_WIRE	BOOL	TRUE	1 = Rotura de hilo
63.1	MON_FRAME	BOOL	TRUE	1 = Error de telegrama (tiene que ser siempre 1)
63.2	MON_PULSE	BOOL	TRUE	1 = Impulsos de error

Seleccionar datos para el encóder absoluto (SSI)

Número de vueltas

Número de vueltas del encóder

El dato máquina "Número de vueltas del encóder" sólo existe en encóders absolutos. Con él se indica el número de vueltas posibles con este encóder. Encontrará más información en el apartado "Encóder absoluto (SSI) (Página 9747)".

- **Encóder monovuelta**
  Sólo es posible el valor 1.
- **Encóder multivuelta**
  - Encóder Multivuelta / Escuchar (abeto): 2 ... 4096 en potencias de 2
  - Encóder multivuelta / Escuchar (alineación derecha): 1 ... 4096 todos los valores
- **Escala lineal**
  También es posible conectar una escala lineal. Introduzca para ello el valor 1.

El **total de pasos del encóder** no es un dato máquina. Se calcula del modo siguiente:

Total de pasos = Incremento por vuelta x Número de vueltas

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentario
36.0	NO_REV	DINT	L#1024	**Número de vueltas del encóder** Rango de valores admisible: • 1 (encóder monovuelta) • 1 a 4096 (encóder multivuelta)

Velocidad de transferencia

Velocidad de transferencia

Con el dato máquina "Velocidad de transferencia" se define la velocidad de la transferencia de datos del encóder SSI al FM x52.

La velocidad de transferencia máxima depende de la velocidad de transferencia:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 63 m → 500 kHz
- 20 m → 1000 kHz

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
40.0	BAUDRATE	DINT	L#0	**Velocidad de transferencia** Valores permitidos: • 0 = 125 kHz • 1 = 250 kHz • 2 = 500 kHz • 3 = 1000 kHz

Longitud de telegrama

Longitud de telegrama

Con la "longitud de telegrama" se define el marco de impulsos que emite el FM x52.

Seleccionando "Escuchar", se desactiva el reloj del FM x52. Entonces el FM x52 puede escuchar también otros telegramas SSI con un marco de telegramas de 13 ó 25 bits. La velocidad de la transferencia se basa en la velocidad de reloj del módulo maestro.

Datos utilizados en el DB de parámetros

La longitud de telegrama seleccionada aquí se dispone en el DB de parámetro con el dato ENC_TYPE (Página 9871).

Sentido del detector de proximidad

Sentido del detector de proximidad

Con el sentido del detector es posible determinar el sentido de un detector de proximidad conectado:

- Positivo: corresponde al sentido de avance (impulsos de contaje ascendentes)
- Negativo: corresponde al sentido de retroceso (impulsos de contaje descendentes)

Pistas

Seleccionar el modo de control de las salidas de pista

Control de las salidas de pista

Con el dato máquina "Control de las salidas de pista" se define el modo de control de las salidas de pista de las pistas 0 a 15. El control es posible mediante:

- Leva electrónica: Las señales de pista son activadas y desactivadas por el procesamiento de levas del FM 452.
- CPU: Las señales de pista reproducen directamente los valores correspondientes de las habilitaciones de pistas en el DB de canal.

De este modo las salidas de pista se pueden controlar directamente desde su programa.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
90.0	TRACK_OUT	WORD	W#16#0	**Control de las salidas de pista** Valores permitidos: • 0 = Leva electrónica • 1 = CPU Número de bit = Número de pista

Seleccionar pistas especiales

Pistas especiales

Es posible parametrizar las pistas 0, 1 y 2 como pistas especiales.

- Pista 0 y pista 1 como pista de leva de contaje (Página 9739)
- Pista 2 como pista de leva de freno (Página 9739)

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
				Pistas especiales
99.0	SPEC_TRC0	BOOL	FALSE	1 = Pista 0 es pista de leva de contaje
99.1	SPEC_TRC1	BOOL	FALSE	1 = Pista 1 es pista de leva de contaje
99.2	SPEC_TRC2	BOOL	FALSE	1 = Pista 2 es pista de leva de freno

Definir valor de contaje superior para la pista de leva de contaje

Valor de contaje superior pista de leva de contaje

Con este dato máquina se define el valor de contaje superior para la pista de leva de contaje parametrizada.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CNT_LIM0	DINT	L#2	Valor de contaje superior para pista de leva de contaje(Pista 0)
104.0	CNT_LIM1	DINT	L#2	Valor de contaje superior para pista de leva de contaje(Pista 1) Rango de valores admisible: • 2 … 65535

Seleccionar entrada de habilitación

Entrada de habilitación

La señal de pista Q3 a Q10 se activa cuando se cumplen todas las condiciones siguientes:

- La pista está habilitada.
- La entrada de habilitación externa correspondiente I3 a I10 está activada.
- El resultado de pista es = 1.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
				Entrada de habilitación
95.0	EN_IN_I3	BOOL	FALSE	1 = La señal de pista Pista 3 está combinada con la entrada de habilitación I3 con la operación lógica AND
…	…	…	…	…
95.7	EN_IN_I10	BOOL	FALSE	1 = La señal de pista Pista 10 está combinada con la entrada de habilitación I10 con la operación lógica AND

Leva

Alcance

Alcance

El alcance determina el número máximo de levas parametrizables y, con ello, el tiempo de ciclo de la leva.

Alcance	Tiempo de ciclo de la leva
16 levas	20,48 µs
32 levas	40,96 µs
64 levas	81,92 µs
128 levas	163,84 µs

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
				Alcance
76.0	C_QTY	DINT	UDT3: L#0	0 = máx. 16 levas
			UDT4: L#1	1 = máx. 32 levas
			UDT5: L#2	2 = máx. 64 levas
			UDT6: L#3	3 = máx. 128 levas

Datos de leva

Datos de leva

Datos de leva

Los datos de leva describen las propiedades de una leva, la asignación de cada una de ellas a una pista y el comportamiento de conmutación de la leva.

Los datos de leva que se listan a continuación se ajustan por separado para cada leva.

- Validez (Página 9886)
- Número de pista (Página 9886)
- Tipo de leva (Página 9887)
- Principio y final de la leva para levas de recorrido (Página 9887)
- Inicio de leva y tiempo de conexión para levas de tiempo (Página 9888)
- Tiempo de anticipación (Página 9889)
- Sentido de actuación (Página 9891)
- Alarma de proceso (Página 9893)

Por lo general, para el procesamiento de levas se aplica lo siguiente:

- Sólo las levas ajustadas como "válidas" son interpretadas y procesadas por el módulo.
- Las levas 0 ... 7 soportan alarmas de proceso
- La cantidad de levas parametrizables depende del alcance.

Validez de una leva

Leva "válida"

Es preciso activar como válidas todas las levas que deben ser procesadas por la leva electrónica.

Nota

Configure siempre las levas no utilizadas como "no válidas" (CAMVALID = FALSE).

Datos utilizados en el DB de parámetros

Dirección relativa	Nombre	Tipo	valor inicial	Comentarios
+0.0	CAMVALID	BOOL	FALSE	1 = Leva "válida"

Definir número de pista para una leva

Número de pista

Al escoger el número de pista se define para cada leva la pista en la que dicha leva tiene que tener efecto.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+1.0	TRACK_NO	BYTE	B#16#0	**Número de pista** Rango de valores admisible: • 0 a 31

Seleccionar tipo de leva

Tipo de leva

Seleccione el tipo de leva deseado.

- Leva de recorrido
- Leva de tiempo

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+0.3	CAM_TYPE	BOOL	FALSE	**Tipo de leva** Valores permitidos: • 0 = Leva de recorrido • 1 = Leva de recorrido

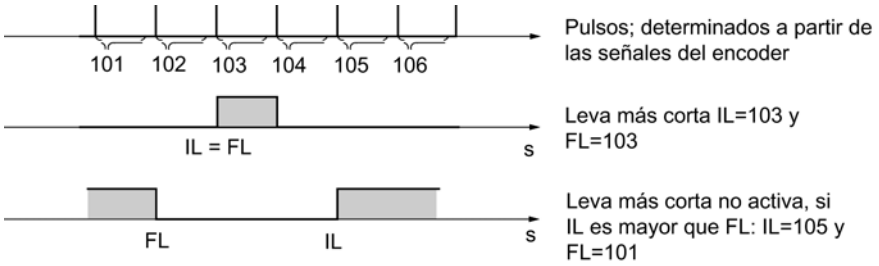
Definir inicio y final de la leva para levas de recorrido

Inicio de leva (IL) / Fin de leva (FL) en levas de recorrido

El inicio y el final de leva definen la longitud de leva en una leva de recorrido.

Longitud mínima de una leva de recorrido

La siguiente figura muestra la leva más corta con sentido de movimiento positivo del eje.



Por regla general, rige lo siguiente:

- La parte no activa de una leva siempre tiene que mantener una distancia mínima de 4 impulsos entre el fin de leva (FL) y el inicio de leva (IL).
- En el caso de FL = IL la leva se activa por un impulso.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios para levas de recorrido
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Inicio de leva (IL)** **Fin de leva (FL)** Rango de valores admisible: • -1.000.000.000 µm a 1 000 000 000 µm

Definir principio de leva y tiempo de conexión para levas de tiempo

Inicio de leva (IL) / Final de leva (FL) ≙ Tiempo de conexión en levas de tiempo

En el caso de una leva de tiempo es preciso definir un inicio de leva y, en lugar del fin de leva, un tiempo de conexión. El tiempo de conexión se define en pasos de 100 µs. El tiempo empieza a transcurrir a partir de la conexión de la leva.

Para la definición de estos tiempos rigen las condiciones siguientes:

- 0 µs: Una leva con el tiempo de conexión 0 µs no se activa nunca
- 0 µs < t ≤ 400 µs: El FM x52 ajusta un tiempo de leva mínimo de unos 330 µs.
- t > 400 µs: El FM x52 calcula el tiempo de conexión real t_{real} a partir del tiempo de conexión especificado t_{def} según la fórmula siguiente:

$$t_{real} = \text{número entero} \left(\frac{t_{def}}{\text{Tiempo de ciclo de la leva}} \right) * \text{Tiempo de ciclo de la leva}$$

El error máximo siempre es menor que el tiempo de ciclo de la leva.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios para levas de tiempo
+2.0 +6.0	CBEGIN CEND	DINT DINT	L#-100000000 L#100000000	**Inicio de leva (IL)** **Final de leva (FL) ≙ Tiempo de conexión** Rango admisible de valores para inicio de leva: • -1.000.000.000 µm a 1 000 000 000 µm Rango admisible de valores para tiempo de conexión: • (0 a 13421) × 100 µs con un máximo de 16 levas • (0 a 26843) × 100 µs con un máximo de 32 levas • (0 a 53686) × 100 µs con un máximo de 64 levas • (0 a 65535) × 100 µs con un máximo de 128 levas

Definir tiempo de anticipación

Tiempo de anticipación

Los retardos de los elementos de maniobra conectados se pueden compensar definiendo un tiempo de anticipación. El tiempo de anticipación se define en pasos de 100 µs. Es posible conceder un tiempo de anticipación por leva. El tiempo de anticipación es válido para el inicio y el final de la leva.

Recorrido de anticipación

El recorrido de anticipación de una leva se calcula continuamente en función de la velocidad actual y del tiempo de anticipación. Toda la leva se desplaza ese recorrido en la dirección del valor real. El área parametrizada se denomina "área estática" y el área calculada atendiendo al tiempo de anticipación se denomina "área dinámica".

Recorrido de anticipación = tiempo de anticipación × velocidad actual

El cálculo de los recorridos de anticipación de todas las levas se realiza dentro de ¼ del tiempo de anticipación más largo parametrizado en el FM x52. Si parametriza un tiempo de anticipación muy prolongado para una leva, el ajuste dinámico se calcula con menos frecuencia.

Tiempo de anticipación real

El tiempo de anticipación real se calcula del modo siguiente:

1. Calcule el tiempo de ciclo de la leva: Es el tiempo en el que el FM x52 ha procesado por completo todas las levas. Depende del alcance parametrizado.
2. Calcule el tiempo de anticipación real con la fórmula siguiente:

$$\text{Tiempo de disponibilidad}_{real} = \text{número entero}\left(\frac{\text{Tiempo de disponibilidad}_{D}}{\text{Tiempo de ciclo de la leva} * 4}\right) * \text{Tiempo de ciclo de la leva} * 4$$

   Significados:
   - Tiempo de anticipación$_{real}$es el tiempo de anticipación ajustado por el FM x52.
   - Tiempo de anticipación $_D$ es el tiempo de anticipación preseleccionado.
   - Número entero ( ) significa que para el cálculo de lo comprendido entre paréntesis sólo se tiene en cuenta el valor anterior al decimal.

El error máximo del tiempo de anticipación$_{real}$ siempre es menor que el tiempo de ciclo de la leva × 4.

Ejemplo:

Se especifican los valores siguientes:

- Alcance: máximo 32 levas
- Tiempo de ciclo de la leva: 40,96 µs
- Tiempo de anticipación$_D$ = 1000 µs

Resultado: Se obtiene un tiempo de anticipación real de 983 µs.

Nota

El tiempo de anticipación real siempre es menor que el tiempo de anticipación parametrizado. Puede ser 0 aunque el tiempo de anticipación parametrizado sea ≥100 µs.

El recorrido de anticipación en un eje rotativo tiene que ser menor que el área del eje rotativo y la parte no activa de la leva. Esto tiene que estar garantizado para todas las velocidades.

Nota

No se soporta un tiempo de anticipación en relación con un encóder absoluto (SSI) ni con una dirección de contaje parametrizada como invertida.

Datos utilizados en el DB de parámetros

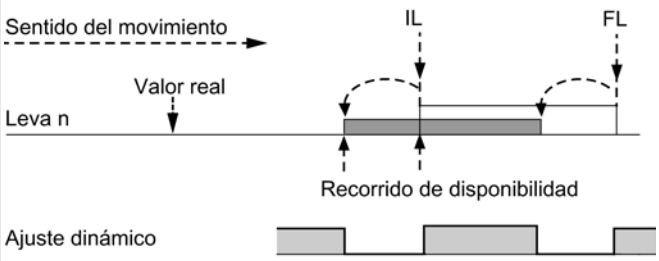
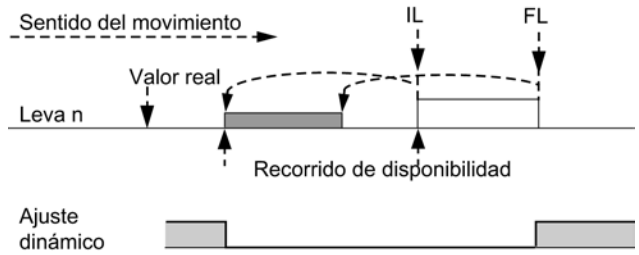
Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+ 10.0	LTIME	INT	0	**Tiempo de anticipación** Rango de valores admisible: • (0 a 53686) × 100 µs con un máx. de 16 levas • (0 a 65535) × 100 µs con un máx. 32, 64 o 128 levas

Ajuste dinámico de una leva

Para el rango de la leva se distinguen dos casos:

1. El rango estático y el rango dinámico de la leva están superpuestos.
2. El rango estático y el rango dinámico de la leva no están superpuestos.

La tabla siguiente muestra para ambos casos el cálculo del ajuste dinámico.

Ajuste dinámico	Descripción
	Cuando el rango dinámico se superpone al rango estático de la leva, rige lo siguiente: • Cuando se alcanza el área dinámica de la leva, ésta se activa. Simultáneamente se bloquea el cálculo de un nuevo ajuste dinámico. • Si el valor real alcanza el área estática de la leva, se vuelve a habilitar el cálculo de un ajuste dinámico nuevo y un cambio de velocidad actúa al final de la leva. • Si la leva se desactiva al final del área dinámica, el ajuste dinámico vuelve a estar bloqueado hasta el final del área estática de la leva.
	Cuando el área dinámica no se superpone al área estática de la leva, rige lo siguiente: • Cuando se alcanza el rango dinámico de la leva, ésta se activa. Simultáneamente se bloquea el cálculo de un nuevo ajuste dinámico. • Al final del área estática de la leva se vuelve a habilitar el ajuste dinámico.
Área dinámica Área estática Es posible un **nuevo** ajuste dinámico	

Nota

Modificando la dirección de giro se vuelve a habilitar la determinación del ajuste dinámico.

Selección sentido de actuación

Sentido de actuación

Existen dos sentidos de actuación posibles:

- Positivo: la leva se activa al inicio, cuando el eje se desplaza en sentido ascendente respecto al valor real.
- Negativo: la leva se activa al final, cuando el eje se desplaza en sentido descendente respecto al valor real.

Es posible también ajustar simultáneamente ambos sentidos de actuación.

Si se selecciona "Ninguno" la leva queda sin efecto.

Comportamiento de conmutación de la leva dependiendo del sentido de actuación

La siguiente tabla muestra el comportamiento de actuación de las levas de recorrido y de tiempo dependiendo del sentido de actuación. Excepto en el ejemplo 5 se parte siempre de un sentido de actuación positivo.

Nº	Descripción	Leva de recorrido	Leva de tiempo
1	Una leva se rebasa en el sentido de actuación	IL FL	IL Tiempo de activación
2	Una leva se rebasa en el sentido contrario al de actuación	IL FL	IL Tiempo de activación
3	Una leva se desplaza en el sentido de actuación; mientras está activada, cambia el sentido del movimiento del eje	IL FL	IL Tiempo de activación x t1 t2 Tiempo de activación x = t1 + t2
4	Una leva se desplaza en sentido opuesto al de actuación; en la leva el sentido de movimiento del eje cambia al sentido de actuación	IL FL	La leva no se activa
5	Una leva se desplaza y se abandona en un sentido cualquiera; el sentido de actuación está ajustado para **ambas direcciones**	IL FL	IL Tiempo de activación x t1 t2 t3 Tiempo de activación x = t1 + t2 + t3

Leva parametrizada
Leva activada

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+0.1	EFFDIR_P	BOOL	TRUE	1 = Sentido de actuación positivo (+)
+0.2	EFFDIR_M	BOOL	TRUE	1 = Sentido de actuación negativo (-)

Seleccionar alarma de proceso para levas

Alarma de proceso al activar/desactivar levas

Para las levas 0 a 7 se puede definir si, al conectar, desconectar o al conectar y desconectar la leva, debe generarse una alarma de proceso.

Datos utilizados en el DB de parámetros

Dirección relativo	Nombre	Tipo	valor inicial	Comentarios
+0.4	PI_SW_ON	BOOL	FALSE	1 = Alarma de proceso al activar
+0.5	PI_SW_OFF	BOOL	FALSE	1 = Alarma de proceso al desactivar

Habilitar alarma de proceso

Habilitar las alarmas de proceso

Habilitar las alarmas de proceso

Los datos máquina permiten habilitar alarmas de proceso.

Es posible ajustar cuándo deben generarse las alarmas de proceso:

- Comienzo de la medición
  En el ajuste "Captura de bordes (Página 9846)", un flanco ascendente de la entrada digital I1 puede iniciar una alarma de proceso.
- Fin de la medición
  Tanto en el ajuste "Captura de bordes (Página 9846)" como en el ajuste "Medición de longitud (Página 9846)", un flanco ascendente de la entrada digital I1 puede iniciar una alarma de proceso.
- Leva activa / inactiva
  En los datos de leva (Página 9890) es posible establecer si se tienen que generar alarmas de proceso al activar y/o desactivar las levas 0 a 7.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
3.1	PI_MEND	BOOL	FALSE	1 = Habilitar la alarma de proceso: Fin de la medición
3.2	PI_CAM	BOOL	FALSE	1 = Habilitar la alarma de proceso: Leva activa / inactiva
3.5	PI_MSTRT	BOOL	FALSE	1 = Habilitar la alarma de proceso: Comienzo de la medición

Evaluación de una alarma de proceso

Cuando el FM 452 dispara una alarma de proceso, en las variables OB4x_POINT_ADDR del OB de alarma de procesos se encuentra disponible la información sobre alarmas.

La tabla siguiente muestra el formato del tipo de datos OB4x_POINT_ADDR:

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	0	0
1	0	0	Comienzo de la medición	0	0	Leva	Fin de la medición	0
2	Leva 7 activa	Leva 7 inactiva	Leva 6 activa	Leva 6 inactiva	Leva 5 activa	Leva 5 inactiva	Leva 4 activa	Leva 4 inactiva
3	Leva 3 activa	Leva 3 inactiva	Leva 2 activa	Leva 2 inactiva	Leva 1 activa	Leva 1 inactiva	Leva 0 activa	Leva 0 inactiva

La causa de la alarma se puede consultar en el byte 1.

- Leva:
  Evalúe los bytes 2 y 3 según la tabla.
- Comienzo / Final de la medición:
  Con la instrucción CAM_MSRM_452 (Página 7460) es posible leer los valores de medición actuales del módulo.

Alarmas de proceso perdidas

Si una alarma de proceso todavía no se ha terminado de procesar en el OB de alarma de proceso, el módulo memoriza todos los eventos siguientes de alarma de proceso. Si vuelve a aparecer un evento antes de que la alarma de proceso se haya podido disparar, el módulo disparará la alarma de diagnóstico "Alarma de proceso perdida".

Interfaz de control y realimentación del FM 452

Aplicación

En aplicaciones especiales o en el nivel de las alarmas es preciso un acceso especialmente rápido a las señales de respuesta y de control. Estos datos se alcanzan directamente por medio de las áreas de entrada y salida del módulo.

Para coordinar el arranque cada vez que arranca el módulo (p. ej. tras insertar el módulo o después de una transición STOP-RUN de la CPU), la instrucción CAM_CTRL (Página 7453) debe llamarse hasta que se muestre el final del arranque a través de RETVAL = 0.

Nota

Para acceder directamente a los datos del FM 452 sólo se deben emplear los datos no internos aquí descritos del modo indicado. De lo contrario, el acceso de su programa de usuario al módulo puede sufrir anomalías.

Leer señales de respuesta mediante acceso directo

Las direcciones de byte se indican de forma relativa con respecto a la dirección de salida del módulo. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7461).

En AWL se accede a los datos con los comandos PEB (leer 1 byte) y PED (leer 4 bytes).

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	PARA	interno	interno	DATA_ERR	interno	DIAG	interno	interno
Byte 1	0	0	0	CAM_ACT	0	0	0	0
Byte 2	interno							
Byte 3	0	0	FVAL_DONE	HYS	GO_P	GO_M	MSR_DONE	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								
Byte 8	TRACK_OUT							
Byte 9								
Byte 10								
Byte 11								

Escribir las señales de control mediante acceso directo

Las direcciones de byte se indican de forma relativa respecto a la dirección de entrada del módulo. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7461).

En AWL se accede a los datos con los comandos PAB (escribir 1 byte) y PAW (escribir 2 bytes).

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	interno							
Byte 1	0	CNTC1_EN	CNTC0_EN	CAM_EN	DIR_P	DIR_M	0	0

Dirección	Número de bit
Byte 2	TRACK_EN
Byte 3	

Ejemplo: Valor real de posición ACT_POS

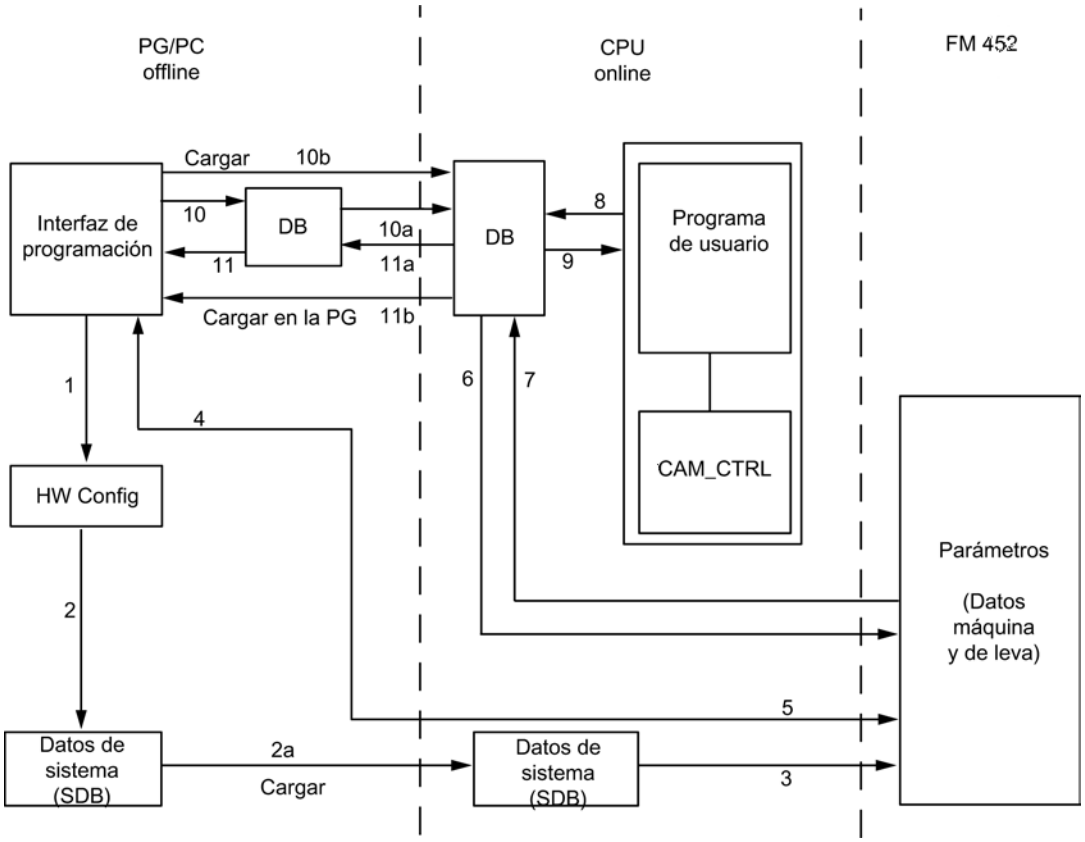
La dirección inicial del módulo es 512.

```
AWL
L PED 516      //Leer el valor real de posición (ACT_POS) con acceso directo:
               //Dirección inicial del módulo + 4
```

Vías de transferencia de parámetros

Se entiende por parámetros los datos máquina y los datos de leva que siguen a continuación.

La figura siguiente muestra las vías de transferencia de parámetros al FM 452.



1 Guardar los parámetros en la interfaz de parametrización.

2 Guardar, compilar y cargar la configuración de hardware en la CPU.

3 La CPU escribe los parámetros en el módulo en parametrizaciones de sistema.

4 Cargar los parámetros del módulo en la programadora con el comando "Cargar sistema de destino en PG".

5 Cargar los parámetros en el módulo desde la interfaz de parametrización con el comando "Cargar sistema de destino".

6 Escribir los parámetros en el módulo con tareas del programa de usuario.

7 Leer los parámetros desde el módulo con tareas del programa de usuario.

8 Almacenar parámetros del programa de usuario en el DB Online.

9 Leer parámetros del DB Online en el programa de usuario.

10 Exportar parámetros desde la interfaz de parametrización hasta el DB (DB Offline o DB Online); posteriormente, también debe cargarse un DB Offline en la CPU.

11 Importar parámetros de un DB Online o DB Offline a la interfaz de parametrización.

Casos de aplicación de la transferencia de parámetros

Caso de aplicación	Pasos
Los parámetros se editan con la interfaz de parametrización. A continuación, el módulo se tiene que parametrizar automáticamente en el arranque.	Ejecute los pasos 1, 2 y 3.
Los parámetros se modifican durante la puesta en servicio en el modo Test en la interfaz de parametrización.	Ejecute los pasos 4 y 5
Los parámetros modificados durante la puesta en servicio se tienen que cargar automáticamente durante el arranque.	Ejecute los pasos 1, 2 y 3.
Los parámetros se crean con la interfaz de parametrización. En el arranque, el módulo sólo se debe parametrizar con el programa de usuario mediante bloques de datos.	Ejecute los pasos 10 y 6.
Para crear cómodamente datos de reserva para recetas.	Ejecute el paso 10.
Los parámetros se crean con la interfaz de parametrización. Éstos tienen que estar a disposición del programa de usuario para modificaciones provisionales.	Ejecute los pasos 1, 2 y 3 para la parametrización automática.
	Ejecute los pasos 10 y 7 para el acceso por medio del programa de usuario.
Los parámetros existentes se modifican (exclusivamente) con el programa de usuario.	Ejecute los pasos 7, 9, 8 y 6.
Para ver los datos modificados por el programa de usuario con la interfaz de parametrización.	Ejecute el paso 11.
Los parámetros modificados por el programa de usuario se tienen que cargar automáticamente también en el arranque.	Ejecute los pasos 6, 11, 1, 2 y 3.

Puesta en servicio del FM 452

Uso del FM 452 desde el programa de usuario

Inclusión del FM 452 en un programa de usuario

Para integrar el FM 452 en un programa de usuario pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

Las instrucciones CAM_INIT y CAM_CTRL o CAM_CTRL_452 se activan en el programa de usuario de la CPU. Éstas realizan la comunicación entre la CPU y el FM 452.

Si integra la instrucción CAM_CTRL en el proyecto, el DB de canal (Página 7461) se creará automáticamente como DB de instancia en la instrucción CAM_CTRL.
Si integra la instrucción CAM_CTRL_452 en el proyecto, entonces deberá insertar en su programa como DB de canal un bloque de datos del tipo CAM_CHANTYPE mediante "Bloques de programa > Agregar bloque".

Todos los datos de parámetros pertenecientes a un canal del módulo (datos máquina, datos leva) se encuentran en el DB de parámetros (Página 7469) del módulo. Inserte un bloque de datos del tipo CAM_PxxxTYPE en el proyecto mediante "Bloques de programa > Agregar bloque". Dependiendo del alcance se requieren los siguientes tipos de DB:

- Hasta 16 levas: CAM_P016TYPE
- Hasta 32 levas: CAM_P032TYPE
- Hasta 64 levas: CAM_P064TYPE
- Hasta 128 levas: CAM_P128TYPE

La instrucción CAM_MSRM_452 se utiliza para la evaluación inmediata de los datos de medición de la medición de longitud o de la captura de bordes de un FM 452 en el OB de la alarma de proceso.

Además, para el FM 452 se puede utilizar también la instrucción CAM_DIAG o CAM_DIAG_452, que permite transferir informaciones de diagnóstico al DB de diagnóstico.

Si integra la instrucción CAM_DIAG en el proyecto, el DB de diagnóstico (Página 7471) se creará automáticamente como DB de instancia en la instrucción CAM_DIAG.
Si integra la instrucción CAM_DIAG_452 en el proyecto, entonces deberá insertar en el programa como DB de diagnóstico un bloque de datos del tipo CAM_DIAGTYPE mediante "Bloques de programa > Agregar bloque".

Puesta en servicio del módulo

Incluso sin programa de usuario es posible realizar un primer test del módulo con el cuadro de diálogo "Puesta en servicio". Tenga en cuenta particularmente la advertencia siguiente.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

Para evitar daños personales y materiales observe los puntos siguientes:

- Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Sólo así puede garantizarse que en caso de un fallo del software o del equipo sea posible desconectar la instalación de modo seguro.
- Instale finales de carrera por hardware que actúen directamente en las etapas de potencia de todos los accionamientos.
- Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.
- Si controla y vigila paralelamente el FM 452 desde el programa y desde el cuadro de diálogo "Puesta en servicio" pueden producirse conflictos cuyos efectos no están definidos. Por lo tanto, conmute siempre la CPU al estado STOP cuando trabaje con el cuadro de diálogo "Puesta en servicio" o bien desactive el programa.

Consulte también

CAM_INIT (Página 7453)

CAM_CTRL (Página 7453)

CAM_DIAG (Página 7455)

CAM_CTRL_452 (Página 7457)

CAM_DIAG_452 (Página 7458)
CAM_MSRM_452 (Página 7460)

FM 452: Diagnóstico online

Información actual sobre el módulo

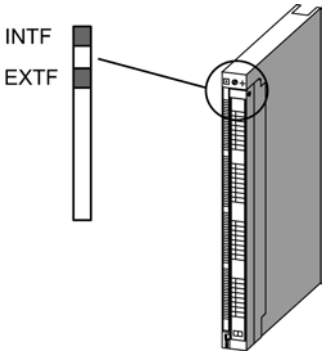
Diagnóstico estándar

Indicación de errores mediante los diodos de error del FM 452

¿Dónde se indica el error?

Los diodos rojos de error del módulo indican diferentes estados de error. Los LED se encienden también en caso de errores que aparecen brevemente, por lo menos, durante 3 s.

En la figura siguiente se muestra la posición de los diodos de error en el FM 452.



¿Qué errores se indican?

La tabla siguiente contiene los tipos de error que se indican mediante los diodos de error.

LED	Significado	Explicación
INTF	Error de grupo para errores internos	Este diodo indica los siguientes estados de error del FM 452: • Alarma de proceso perdida • Vigilancia de tiempo transcurrida • FM 452 no configurado • FM 452 parametrizado con errores Para eliminar el error, ver Clases de error (Página 9906).
EXTF	Error de grupo para errores externos	Este LED indica los siguientes estados de error: • Falta tensión auxiliar externa de 24 V • Falta conector frontal • Rotura de hilo encóder • Fallo operativo • Error de telegrama encóder absoluto • Error en señal encóder incremental o falta marca cero
INTF + EXTF	Módulo defectuoso	Todas las salidas se desconectan. El módulo debe ser sustituido.

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

Procesamiento de las alarmas

El FM 452 puede disparar alarmas de proceso y de diagnóstico. Estas alarmas se procesan en un OB de alarma. Cuando se dispara una alarma sin que se encuentre cargado el OB correspondiente, la CPU pasa a STOP.

Para habilitar las alarmas de diagnóstico se procede del modo siguiente:

1. Seleccione el módulo en HWCN
2. Habilite la alarma de diagnóstico en Edición > Propiedades del objeto > Parámetros básicos.
3. Guardar y compilar la configuración.
4. Cargue la configuración en la CPU

Visión de conjunto de las alarmas de diagnóstico

Los eventos y errores siguientes disparan una alarma de diagnóstico:

- Fallo operativo
- Datos máquina erróneos

- Datos de levas erróneos
- Error de diagnóstico

Estos errores se describen con detalle en el apartado "Clases de error (Página 9906)".

Reacción del módulo en caso de error con alarma de diagnóstico

- El procesamiento de levas se desactiva.
- Con las alarmas de diagnóstico siguientes se borra la sincronización:
  - Falta conector frontal, falta tensión auxiliar externa
  - se ha detectado un error de señal cero, error de línea (señales de encóder de 5 V)
  - se ha rebasado la zona de desplazamiento (se notifica con un error de operador)
  - Ajustar valor real no se puede ejecutar (se notifica con un error de operador)
- Las señales de control dejan de procesarse excepto en un caso.
  Excepción:
  En caso de desbordamiento del final de carrera por software todavía es posible un cambio de sentido en el modo de simulación.
- Los selectores de función y las tareas se continúan procesando.

El módulo detecta un fallo ("entrante")

Una alarma de diagnóstica es "entrante" cuando por lo menos hay un error en cola de espera. Si no se han solventado todos los fallos los demás que queden en cola de espera, se notificarán de nuevo como "entrantes".

Secuencia:

1. El FM 452 detecta uno o varios fallos y dispara una alarma de diagnóstico. El LED de error se enciende. El error se registra en el búfer de diagnóstico.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Es posible evaluar la información de inicio del OB 82.
4. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
5. Encontrará más información si llama la instrucción CAM_DIAG (Página 7455) o CAM_DIAG_452 (Página 7458).

El módulo detecta el paso al estado libre de fallos ("saliente")

Una alarma sólo es "saliente" cuando se ha eliminado el último error del módulo.

Secuencia:

1. El FM 452 detecta que todos los fallos han sido eliminados y dispara una alarma de diagnóstico. El LED de error está apagado. El búfer de diagnóstico no se modifica.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.

3. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
4. Evalúe el bit OB82_MDL_DEFECT.
   Si este bit es "0", no hay errores en el módulo. Su evaluación puede finalizar aquí.

Alarmas de diagnóstico según el estado de la CPU

- En estado STOP de la CPU las alarmas de diagnóstico están bloqueadas por el FM 452.
- Si con la CPU en estado STOP no se eliminan todos los fallos pendientes, el FM 452 vuelve a notificar como "entrantes" los errores todavía no eliminados tras la transición al estado RUN.
- Si se eliminan todos los errores pendientes con la CPU en estado STOP, el estado libre de errores del FM 452no se notificará con una alarma de diagnóstico tras la transición al estado RUN.

Evaluación de errores en el programa de usuario

Reacción de errores en el programa de usuario

En el programa de usuario es posible reaccionar de forma precisa a los errores que aparecen. Para ello se dispone de los siguientes recursos:

- Los valores de retorno RET_VAL de las instrucciones integradas como indicador de grupo de los errores que han aparecido durante el transcurso de las instrucciones.
- Los bits de error _ERR de las peticiones como indicador de grupo de errores que se han producido durante la ejecución de una petición.
- El bit de error DATA_ERR como indicador de grupo de un error que el FM 452 ha detectado durante una petición de escritura.
- La identificación de errores en JOB_ERR de la causa de error en la comunicación entre la instrucción y el FM 452.
- La instrucción CAM_DIAG (Página 7455) o CAM_DIAG_452 (Página 7458) para leer el búfer de diagnóstico del FM 452. Aquí es posible averiguar las causas de error de las peticiones y los eventos asíncronos (fallos operativos y errores de diagnóstico).
- Alarma de diagnóstico para una respuesta rápida a eventos.

Búfer de diagnóstico del módulo

Eventos de diagnóstico

El búfer de diagnóstico del módulo contiene como máximo 4 eventos de diagnóstico y está organizado como búfer anular.

Un evento de diagnóstico se escribe en el búfer cuando se detecta un aviso (de error) entrante. Esto puede ser un aviso, un error isócrono (error de datos) o incluso un error asíncrono (error de servicio y error de diagnóstico). De una causa de un error pueden surgir también varias entradas como errores consiguientes. Los avisos salientes no generan entradas en el búfer de diagnóstico.

Para cada evento de diagnóstico se indican:

- Estado (siempre entrante)
- Error interno
- Error externo
- Clase de error
- Número de error
- Número de canal
- Número de levas (si hay errores de datos de levas)

Cuando se escribe un evento de diagnóstico en el búfer de diagnóstico, la señal de realimentación DIAG = 1 se activa.

El búfer de diagnóstico puede transferirse como un todo con la instrucción CAM_DIAG (Página 7455) o CAM_DIAG_452 (Página 7458) a un bloque de datos (DB de diagnóstico) o a través de un cuadro de diálogo de evaluación de errores del software de configuración. Si se lee el búfer de diagnóstico, el módulo activa la señal de realimentación DIAG = 0.

Nota

Si el búfer de diagnóstico se lee al mismo tiempo con la instrucción CAM_DIAG o CAM_DIAG_452 y el cuadro de diálogo de evaluación de errores, puede ocurrir que uno de los eventos de diagnóstico que se ha producido no sea detectado por el programa.

Datos y estructura del DB de diagnóstico

Datos y estructura del DB de diagnóstico

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

La tabla siguiente muestra la estructura del DB de diagnóstico.

Dirección	Nombre	Tipo	valor inicial	Comentarios
0.0	MOD_ADDR (registrar)	INT	0	Dirección del módulo
256.0	JOB_ERR	INT	0	Error de comunicación
258.0	JOBBUSY	BOOL	FALSE	1 = Petición activa
258.1	DIAGRD_EN	BOOL	FALSE	1 = Leer el búfer de diagnóstico necesariamente
260.0	DIAG_CNT	INT	0	Número de entradas válidas de la lista
262.0	DIAG[1]	STRUCT		Entrada más reciente de datos de diagnóstico

Dirección	Nombre	Tipo	valor inicial	Comentarios
272.0	DIAG[2]	STRUCT		Segunda entrada de datos de diagnóstico
282.0	DIAG[3]	STRUCT		Tercera entrada de datos de diagnóstico
292.0	DIAG[4]	STRUCT		Última entrada de datos de diagnóstico

La tabla siguiente muestra la estructura de una entrada de diagnóstico DIAG[n].

Dirección	Nombre	Tipo	valor inicial	Comentarios
+0.0	STATE	BOOL	FALSE	0 = Evento saliente 1 = Evento entrante
+0.1	INTF	BOOL	FALSE	1 = Error interno
+0.2	EXTF	BOOL	FALSE	1 = Error externo
+2.0	FCL	INT	0	Clase de error 1: Fallo operativo 4: Fallo de datos 5: Fallo de datos máquina 7: Fallo de datos de leva 15: Avisos 128: Error de diagnóstico
+4.0	FNO	INT	0	Número de error 0 a 255
+6.0	CH_NO	INT	0	Número de canal (siempre 1)
+8.0	CAMNO	INT	0	Número de leva de 0 a 127 con clase de error = error de datos de leva

Lista de los avisos JOB_ERR

Avisos JOB_ERR

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A0	32928	-32608	Acuse negativo al leer el módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2 1)	32930	-32606	Error de protocolo en la capa 2 (transferencia de datos dentro de PROFINET / PROFIBUS DP interrumpida, p. ej. por rotura de hilo, ausencia de conector de cierre, error de parametrización, etc.)
80A3 1)	32931	-32605	Error de protocolo en User-Interface/User (transferencia de datos dentro de PROFINET / PROFIBUS DP interrumpida, p. ej. por rotura de hilo, ausencia de conector de cierre, error de parametrización, etc.)
80A4 1)	32932	-32604	Anomalía en la comunicación del bus K
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo teórico.
80C0 1)	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80C1 [1]	32961	-32575	Los datos de una tarea de escritura semejante todavía no se han procesado en el módulo.
80C2 [1]	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3 [1]	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados de instante.
80C4 [1]	32964	-32572	Error de comunicación
80C5 [1]	32965	-32571	La periferia descentralizada no está disponible.
80C6 [1]	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (Petición de escritura)
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (Petición de escritura)
853A	34106	-31430	DB de parámetros no presente. (Petición de escritura)
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras un error. (Petición de escritura)
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (Petición de lectura)
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (Petición de lectura)
873A	34618	-30918	DB de parámetros no presente. (Petición de lectura)
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras un error. (Petición de lectura)

[1] Los errores 80A2 a 80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario. Los avisos con la estructura 7xxx indican situaciones temporales de la comunicación.

Clases de error

Fallo operativo

Clase de error 1: Fallo operativo

Los fallos operativos se detectan de forma asíncrona respecto a un control.

La tabla siguiente muestra los fallos operativos posibles y sus causas.

Nº	Significado	Alarma de diagnóstico
1	Rebase del inicio del final de carrera por software	Sí
2	Rebase del fin del final de carrera por software	Sí
3	Rebase del inicio de la zona de desplazamiento	Sí
4	Rebase del fin de la zona de desplazamiento	Sí
13	Ajustar valor real al vuelo no ejecutable	Sí

Nº	Significado		Alarma de diagnóstico
	Causa	Los finales de carrera por software, tras ajustar valor real al vuelo, se encuentran fuera de la zona de desplazamiento (-100 m...+100 m o -1.000 m...+1.000 m). El decalaje resultante de Ajustar valor real / Ajustar valor real al vuelo es mayor que ±100 m o ±1.000 m.	
	Efecto	Eje no sincronizado.	

Error de datos

Clase de error 4: Error de datos

Los errores de datos se detectan de forma síncrona respecto a un control/mando.

La tabla siguiente muestra los errores de datos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
10	Decalaje de señal cero erróneo		No
	Causa	El decalaje de señal cero es mayor que ± 100 m o ± 1.000 m. Los finales de carrera por software, tras ajustar el decalaje de señal cero, se encuentran fuera de la zona de desplazamiento (-100 m...+100 m o -1.000 m...+1.000 m). Eje rotativo: El valor del decalaje de señal cero es mayor que el final del eje rotativo.	
11	Indicación del valor real errónea		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
12	Punto de referencia erróneo		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
20	Activar datos máquina no admisible		no
	Causa	En el módulo no hay datos máquina nuevos (sin errores)	
21	Ajustar valor real al vuelo no admisible		no
	Causa	Se ha intentado ejecutar "Ajustar valor real al vuelo" teniendo "Redisparar punto de referencia" activado.	
27	Ajuste codificado en bits no permitido		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0. Se ha intentado seleccionar simultáneamente" Medición de longitud" y "Captura de bordes".	
28	Redisparar el punto de referencia no admisible		no

Nº	Significado		Alarma de diagnóstico
	Causa	Se ha intentado ejecutar "Redisparar el punto de referencia" estando "Ajustar valor real al vuelo" pendiente. Se ha intentado ejecutar "Redisparar el punto de referenciao" con un encóder SSI.	
29	**Comando codificado en bits no permitido**		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
30	**Tiempo de anticipación erróneo**		no
31	**Número de leva erróneo**		no
	Causa	La leva no es válida. El número de leva no se encuentra en el rango 0..127.	
32	**Inicio de leva erróneo**		no
	Causa	El inicio de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: El principio de leva es < 0 o mayor que el final del eje rotativo.	
33	**Fin de leva erróneo / Tiempo de activación erróneo**		no
	Causa	El fin de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: El final de leva es < 0 o mayor que el final del eje rotativo. La leva no está activada por lo menos en la duración de un impulso. En una leva invertida no hay una distancia mínima de 4 impulsos entre inicio de leva y fin de leva.	
34	**No es posible cancelar la activación del valor real**		no
	Causa	Con un encóder SSI y un eje líneal, tras la ejecución del ajuste el valor real de posición se encontraría fuera de la zona de trabajo.	
35	**Indicación del valor real errónea al Ajustar valor real / Ajustar valor real al vuelo**		no
	Causa	La indicación del valor real se encuentra fuera del margen numérico admisible de ± 100 m ó ± 1.000 m. Tras la ejecución del ajuste, los finales de carrera por software se encontrarían fuera de la zona de desplazamiento (-100 m...+1000 m ó -1.000 m...+1.000 m). El decalaje resultante de Ajustar valor real / Ajustar valor al vuelo sería mayor que ±100 m ó ±1.000 m.	
107	**Eje no parametrizado**		no
	Causa	En el eje no hay ningún dato máquina disponible. En el eje no hay ningún dato máquina activado.	
108	**Eje no sincronizado**		no
	Causa	Uno de los ajustes "Ajustar valor real" o "Ajustar valor real al vuelo" se ha activado a pesar de que el eje no está sincronizado.	
109	**Procesamiento de levas en curso.**		no
110	**Número erróneo de levas a modificar**		no

Error de datos máquina

Clase de error 5: Error de datos máquina

La alarma de diagnóstico sólo se disparará con un módulo de datos del sistema (SDB) con errores.

La tabla siguiente muestra errores de datos máquina posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
5	**Error en el ajuste de la alarma de proceso**		Sí
	Causa	Ha intentado seleccionar una alarma de proceso que el módulo no soporta.	
6	**Distancia mínima entre bordes errónea**		Sí
	Causa	Ha indicado como distancia mínima entre bordes un valor < 0 ó > 10^9 µm.	
8	**Tipo de eje erróneo**		Sí
	Causa	Como tipo de eje no ha indicado ni 0 ni 1.	
9	**Final erróneo del eje rotativo**		Sí
	Causa	El valor del final del eje rotativo se encuentra fuera del rango admitido de 1 a 10^9 µm o respectivamente de 1 a 10^8 µm (según sea la resolución).	
10	**Tipo de encóder erróneo**		Sí
	Causa	El valor del tipo de encóder se encuentra fuera del rango admisible de 1 a 10	
11	**Recorrido erróneo por vuelta**		Sí
	Causa	El valor del recorrido/vuelta se encuentra fuera del margen admisible de 1 a 10^9 µm (independientemente de la resolución).	
13	**Número erróneo de incrementos/vuelta del encóder**(véase el apartado "Incrementos por vuelta del encóder (Página 9873)")		Sí
14	**Número erróneo de vueltas** (véase el apartado ""Número de vueltas de encóder (Página 9878)")		Sí
15	**Velocidad de transferencia errónea**		Sí
	Causa	Para la velocidad de transferencia ha indicado un valor fuera del rango admisible de 0 a 3.	
16	**Coordinada del punto de referencia errónea**		Sí
	Causa	La coordenada se encuentra fuera del margen de -100 m a +100 m o, respectivamente, de -1000 m a +1000 m, según la resolución. Eje lineal: La coordenada se encuentra fuera de la zona de trabajo. Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
17	**Reajuste erróneo del encóder absoluto**		Sí
	Causa	Encóder SSI de recorrido: El valor del reajuste del encóder absoluto no se encuentra dentro del rango del encóder (Incrementos por vuelta × Cantidad de vueltas - 1).	
18	**Forma errónea para redisparar el punto de referencia**		Sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0, 1, 6 y 7.	
19	**Ajuste erróneo del sentido**		Sí
	Causa	Ha indicado un valor fuera de los valores admisibles 0 y 1.	
20	**Supervisión de hardware no posible**		Sí
	Causa	Ha puesto en "FALSE" la vigilancia de errores de telegrama en el DB de parámetros.	

Nº	Significado		Alarma de diagnóstico
21	**Inicio erróneo del final de carrera por software**		Sí
	Causa	Eje lineal: El inicio del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución). Eje lineal: El inicio del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es menor que -100 m ó -1000 m (según la resolución).	
22	**Fin erróneo del final de carrera por software**		Sí
	Causa	Eje lineal: El fin del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m, dependiendo de la resolución) o es menor que el inicio del final de carrera por software. El fin del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es mayor que +100 m ó +1000 m (según la resolución).	
144	**Alcance erróneo**		Sí
	Causa	Para el alcance ha indicado un valor distinto de 0 a 3.	
145	**Histéresis errónea**		Sí
	Causa	La histéresis se encuentra fuera del rango 0..65535 × resolución. El decalaje de señal cero es mayor que ¼ × zona de trabajo o ¼ × área de eje rotativo.	
146	**Velocidad de simulación errónea**		Sí
	Causa	La velocidad de simulación se encuentra fuera del rango 1000 × RES hasta 3×10^7 × RES o es mayor que 5×10^8 µm/min. La velocidad de simulación no puede ajustarse internamente.	
147	**Pista errónea**		Sí
	Causa	Se ha seleccionado el control de una pista fuera de 0..15 (bit 0..15).	
148	**Selección errónea de entradas de habilitación**		Sí
	Causa	Ha querido habilitar una pista fuera de 3..10 (bit 0..7) mediante una señal externa.	
149	**Selección errónea de una pista especial**		Sí
	Causa	Ha querido definir una pista fuera de 0, 1 y 2 (bit 0, 1 y 2) a modo de pista especial	
150	**Valor de contaje superior erróneo pista 0**		Sí
	Causa	Ha indicado un valor de contaje < 2 ó > 65535 como valor de contaje superior.	
151	**Valor de contaje superior erróneo pista 1**		Sí
	Causa	Ha indicado un valor de contaje < 2 ó > 65535 como valor de contaje superior.	
200	**Resolución errónea**		Sí
	Causa	Ha indicado una resolución < 0,1 µm/impulso ó >1000 µm/impulso. Ha indicado un recorrido/vuelta y una cantidad de impulsos/vuelta que dan una resolución < 0,1 ó > 1000.	
201	**El encóder de recorrido no se adapta a la zona de trabajo / área del eje rotativo**		Sí
	Causa	Encóder SSI de recorrido y eje rotativo: El encóder de recorrido no cubre exactamente el área del eje rotativo. Eje lineal: El encóder de recorrido no cubre ni siquiera la zona de trabajo (finales de carrera por software incluidos).	

Error de datos de leva

Clase de error 7: Error de datos de leva

La alarma de diagnóstico sólo se disparará con un bloque de datos del sistema (SDB) con errores.

La tabla siguiente muestra los errores de datos de leva posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Alarma de proceso no admisible**		Sí
	Causa	Ha querido indicar una alarma de proceso con una leva con un número > 7.	
2	**Número de pista erróneo**		Sí
	Causa	El número de pista se encuentra fuera del margen 0 a 31.	
3	**Inicio de leva erróneo**		Sí
	Causa	El inicio de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). Eje rotativo: El principio de leva es < 0 o mayor que el final del eje rotativo.	
4	**Final de leva erróneo**		Sí
	Causa	El fin de leva se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó -1000 m...+1000 m). La leva no tiene una longitud de por lo menos 1 impulso. Eje rotativo: El final de leva es < 0 o mayor que el final del eje rotativo. En una leva invertida no hay una distancia mínima de 4 impulsos entre inicio de leva y fin de leva.	
5	**Tiempo de activación erróneo**		Sí
	Causa	El tiempo de conexión es < 0µs. El valor máximo (Página 9885) depende del alcance.	
6	**Tiempo de anticipación erróneo**[1]		Sí
	Causa	El tiempo de anticipación es < 0 µs. El valor máximo (Página 9886) depende del alcance.	
50	**Demasiados juegos de levas**		Sí
	Causa	Ha querido indicar más juegos de levas de los posibles con este alcance.	
51	**Eje en funcionamiento**		Sí
	Causa	Ha querido indicar juegos de levas con la leva electrónica activada.	
52	**Eje no parametrizado**		Sí
	Causa	Ha querido indicar datos de levas a pesar de que todavía no hay datos máquina activos.	

[1] El mensaje de error también puede aparecer si ha parametrizado el sentido de contaje "invertido" en combinación con un encóder absoluto (SSI).

Avisos

Clase de error 15: Avisos

La tabla siguiente muestra los avisos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Inicio de parametrización**		no
	Causa	El módulo ha detectado una parametrización mediante un bloque de datos del sistema.	
2	**Final de la parametrización**		no
	Causa	El módulo ha procesado sin problemas la parametrización mediante un bloque de datos del sistema.	

Error de diagnóstico

Clase de error 128: Error de diagnóstico

La tabla siguiente muestra los errores de datos posibles y sus causas, así como los efectos y las soluciones posibles.

Nº	Significado		Alarma de diagnóstico
4	**Falta tensión auxiliar externa**		Sí
	Causa	La tensión auxiliar externa de 24 V no se ha conectado o ha caído, falta el conector frontal	
	Efecto	Consulte "Activación de alarmas de diagnóstico (Página 9898)": • El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización, con encóders incrementales • El FM 452 no está parametrizado (señal de realimentación PARA = 0).	
	Solución	Comprobar que la conexión de 24 V sea correcta (si la conexión de 24 V es correcta, el módulo está averiado).	
5	**Falta conector frontal**		Sí
	Causa	El conector frontal no está insertado	
	Efecto	• Falta tensión auxiliar externa de 24 V • El módulo no está listo para funcionar	
	Solución	Enchufar el conector frontal	
51	**Vigilancia de tiempo transcurrida**		Sí
	Causa	• Interferencias fuertes en el FM 452 • Error en el FM 452	

Nº	Significado		Alarma de diagnóstico
	Efecto	• El módulo se resetea • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, este vuelve a estar listo para funcionar. • El módulo notifica el transcurso de la vigilancia del tiempo (watchdog) con "entrante" y "saliente"	
	Solución	• Eliminación de interferencias • Consulte al departamento comercial correspondiente cuando las circunstancias exactas que han dado lugar al error son de mayor entidad. • Cambio del FM 452	
52	**Caída de la alimentación interna del módulo**		Sí
	Causa	Error en el FM 452	
	Efecto	• El módulo se resetea • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, este vuelve a estar listo para funcionar.	
	Solución	Cambio del FM 452	
70	**Alarma de proceso perdida**		Sí
	Causa	El FM 452 ha detectado un evento de alarma de proceso, que no puede ser notificado porque el mismo evento todavía no ha sido procesado por el programa de usuario o la CPU.	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización, con encóders incrementales	
	Solución	• Incorporar el OB 40 en el programa de usuario • Comprobar la conexión de bus del módulo • Desactivar la alarma de proceso • Adapte el hardware y software a las necesidades de su proceso (p. ej. CPU más rápida, optimizar el programa de usuario)	
144	**Rotura de hilo Encóder**		Sí
	Causa	• El cable del encóder no está enchufado o está cortado • Encóder sin señales de cuadratura • Asignación de pines errónea • Longitud de cable excesiva • Cortocircuito de las señales del encóder	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización, con encóders incrementales	
	Solución	• Comprobar el cable del encóder • Guardar los datos técnicos del encóder • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador. • Guardar las especificaciones técnicas del módulo	
145	**Error de telegrama Encóder absoluto**		Sí

Nº	Significado		Alarma de diagnóstico
	Causa	El tráfico de telegramas entre el FM 452 y el encóder absoluto (SSI) tiene errores o está interrumpido: • El cable del encóder no está enchufado o está cortado • Tipo de encóder erróneo • Encóder mal configurado (encóders programables) • Longitud de telegrama mal indicada • El encóder proporciona valores erróneos (encóder defectuoso) • Interferencia en el cable del sistema de medición • Velocidad de transferencia escogida demasiado alta	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • el último valor real correcto se mantiene sin cambios hasta el final de la siguiente transferencia SSI correcta	
	Solución	• Comprobar el cable del encóder • Comprobar el encóder • Comprobar el tráfico de telegramas entre el encóder y el FM 452	
146	Impulsos erróneos (encóder incremental)		Sí
	Causa	• Vigilancia del encóder ha detectado impulsos erróneos • La cantidad de incrementos por vuelta está mal introducida • Encóder defectuoso: no proporciona la cantidad de impulsos indicada • Marca cero errónea o no presente • Interferencias en el cable del encóder	
	Efecto	• El procesamiento de levas se desactivará • Desconexión de las salidas de pista: • Borrado de la sincronización	
	Solución	• Indicar correctamente el número de incrementos/vuelta. • Comprobar el encóder y su cable • Cumplir las normas de apantallamiento y de conexión a tierra • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador.	

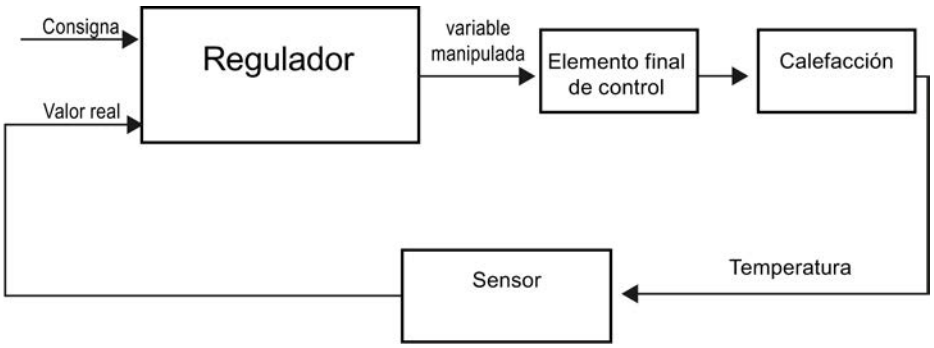
11.3 Regulación PID

11.3.1 Principios básicos de la regulación

11.3.1.1 Lazo de regulación y elementos finales de control

Lazo de regulación

Un ejemplo sencillo de un lazo de regulación es la regulación de la temperatura ambiente mediante una calefacción. La temperatura ambiente se mide con un sensor y se transmite a un regulador. Este compara la temperatura ambiente actual con una consigna y calcula un valor de salida (variable manipulada) para controlar la calefacción.



Un regulador PID ajustado correctamente alcanza la consigna lo más rápido posible y luego la mantiene constante. Tras un cambio del valor de salida el valor real suele cambiar de forma retardada. Este comportamiento debe compensarse con el regulador.

Elementos finales de control

El elemento final de control es un componente del lazo de regulación y se ve influenciado por el regulador. De esta manera se modifica un flujo másico o energético.

La tabla siguiente muestra los campos de aplicación de los elementos finales de control

Aplicación para	Elemento final de control
Flujo másico de líquidos y gases	Válvula, compuerta, corredera
Flujo másico sólido p. j. carga a granel	Válvula de descarga, cinta transportadora, canal vibratorio
Flujo eléctrico	Conmutador, contactor, relé, tiristor
	Resistor variable, transformador de regulación, transistor

Los elementos finales de control se diferencian del modo siguiente:

- Elementos finales de control proporcionales con señal de posicionamiento constante
  Los ángulos de apertura, ángulos de rotación o posiciones se adoptan proporcionalmente al valor de salida. Dentro del rango de ajuste el valor de salida incide en el proceso de manera análoga.
  A los elementos finales de control de este grupo pertenecen los accionamientos neumáticos cargados por resorte, así como los accionamientos motorizados con realimentación de posición en los que se conforma un lazo de regulación de posicionamiento.
  El valor de salida es generado por un regulador continuo, p. ej. PID_Compact.
- Elementos finales de control proporcionales mediante señal con modulación de ancho de impulsos
  En estos elementos finales de control (actuadores) se emite un impulso con una frecuencia equivalente al tiempo de muestreo y cuya longitud es proporcional al valor de salida. El elemento final de control, p. ej. una resistencia calefactora o un grupo frigorífico, se conecta de forma sincronizada con el ciclo por un tiempo variable en función del valor de salida.
  La señal de posicionamiento puede adoptar unipolarmente los estados "Conectado" o "Desconectado" o bien representar bipolarmente los valores "Abierto/Cerrado", "Adelante/Atrás", "Acelerar/Decelerar" etc.
  El valor de salida es generado por un regulador de dos puntos, p. ej. PID_Compact con modulación de ancho de impulsos.
- Elementos finales de control de acción integral con señal de salida de tres puntos
  Frecuentemente los elementos finales de control se accionan con motores cuya duración de conexión es proporcional al trayecto de ajuste del órgano de regulación. A este grupo pertenecen p. ej. válvulas, compuertas y correderas. A pesar de que su diseño varía considerablemente, estos elementos finales de control se asemejan en cuanto a que tienen el efecto de una acción I en la entrada del sistema regulado.
  El valor de salida es generado por un regulador paso a paso, p. ej. PID_3Step.

11.3.1.2 Sistemas regulados

Las propiedades de un sistema regulado están definidas por aspectos tecnológicos y mecánicos que dejan un margen de influencia muy reducido. Para que la regulación arroje buenos resultados es imprescindible elegir un tipo de regulador apropiado, que pueda ser adaptado óptimamente a la rapidez de respuesta del sistema regulado. Por lo tanto, es esencial conocer a fondo los datos característicos y el tipo del sistema regulado para poder configurar la acción P, I y D del regulador.

Tipos de sistema regulado

Los sistemas regulados se clasifican en función de su rapidez de respuesta a cambios bruscos del valor de salida.

Se distinguen los siguientes sistemas regulados:

- Sistemas regulados con compensación
  - Sistemas regulados tipo P
  - Sistemas regulados tipo PT1
  - Sistemas regulados tipo PT2
- Sistemas regulados sin compensación
- Sistemas regulados con y sin tiempo muerto

Sistemas regulados con compensación

Sistemas regulados tipo P

En los sistemas proporcionales el valor real sigue al valor de salida casi instantáneamente. La relación entre el valor real y el valor de salida se expresa mediante la ganancia proporcional Gain del sistema regulado.

Ejemplos:

- Válvula deslizante en un sistema de tuberías
- Divisor de tensión
- Reducción en sistemas hidráulicos

Sistemas regulados tipo PT1

En un sistema regulado tipo PT1 el valor real varía en un principio de manera proporcional al cambio del valor de salida. A medida que transcurre el tiempo, el valor real cambia más lentamente (con retardo) hasta alcanzar un valor final.

Ejemplos:

- Sistemas de amortiguación por resorte
- Carga de componentes RC
- Depósito de agua calentado con vapor.

Frecuentemente rigen las mismas constantes de tiempo para procesos de calefacción y refrigeración o curvas de carga y descarga. Si las constantes de tiempo son divergentes, la regulación se torna mucho más complicada.

Sistemas regulados tipo PT2

En un sistema regulado tipo PT2 el valor real en un principio no cambia ante un escalón del valor de salida, pero luego aumenta en rampa de subida hasta finalmente aproximarse a la consigna con una rampa de bajada. El sistema regulado presenta un comportamiento de transferencia proporcional con retardo de 2º grado.

Ejemplos:

- Regulación de presión
- Regulación de caudal
- Regulación de temperatura

Sistemas regulados sin compensación

Los sistemas regulados sin compensación presentan un comportamiento integral. El valor real persigue un valor infinitamente grande.

Ejemplo:

- Entrada de un líquido determinado a un depósito

Sistemas regulados con tiempo muerto

Un tiempo muerto es siempre el tiempo de ejecución o de transporte hasta que un cambio en la entrada del sistema pueda ser medido en la salida del mismo.

En los sistemas regulados con tiempo muerto el valor real varía con retardo en función del tiempo muerto.

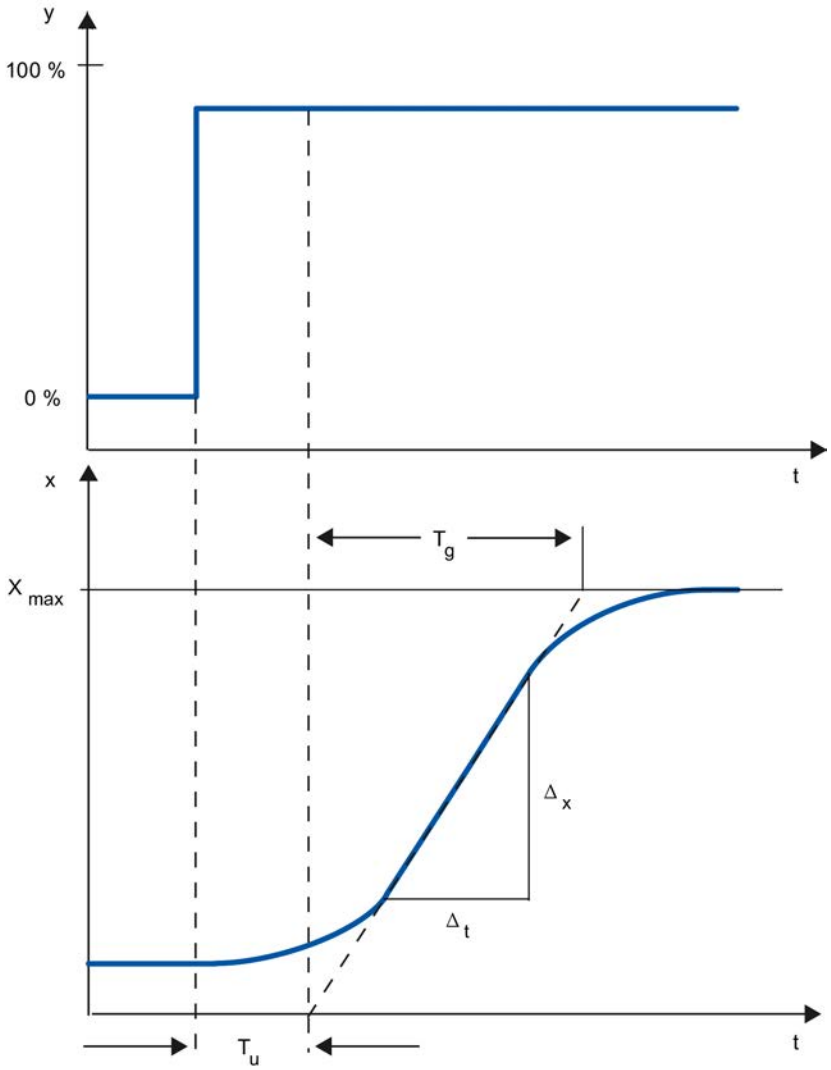
Ejemplo:

Cinta transportadora

11.3.1.3 Valores característicos del sistema regulado

Determinar el comportamiento de tiempo a partir de la respuesta indicial

La respuesta temporal del sistema regulado se puede determinar a partir de la evolución temporal del valor real x tras un cambio brusco del valor de salida y. En la mayoría de los casos se trata de sistemas regulados con compensación.



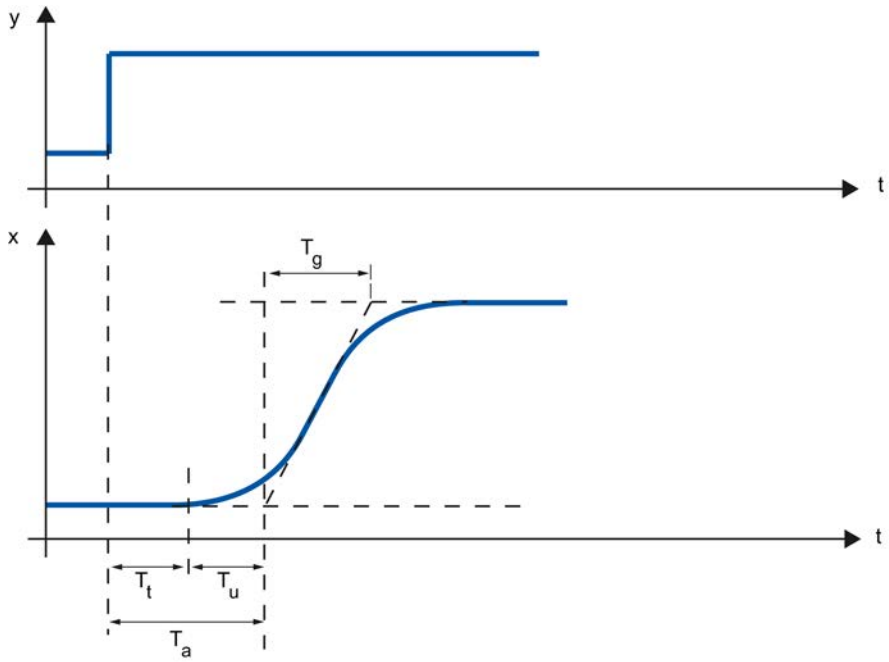
El comportamiento de tiempo se puede definir aproximadamente con las magnitudes tiempo de retardo T_u, tiempo de compensación T_g y valor máximo X_{max}. Las magnitudes se determinan posicionando tangentes en el valor máximo y en el punto de inflexión de la respuesta indicial. En muchos casos no es posible aceptar la respuesta transitoria hasta el valor máximo debido a que el valor real no puede rebasar ciertos valores. Luego se utiliza la velocidad de aumento v_{max} para identificar el sistema regulado ($v_{max} = \Delta_x/\Delta_t$).

La regulabilidad del sistema regulado se puede estimar a partir de la relación T_u/T_g o bien $T_u \times v_{max}/X_{max}$ Rige lo siguiente:

Tipo de sistema	T_u / T_g	Regulabilidad del sistema regulado
I	< 0,1	Altamente regulable
II	0,1 hasta 0,3	Todavía regulable
III	> 0,3	Difícil de regular

Incidencia del tiempo muerto en la regulabilidad del sistema regulado

Un sistema regulado con tiempo muerto y compensación presenta la respuesta siguiente ante un escalón del valor de salida



T_t	Tiempo muerto
T_u	Tiempo de retardo
T_g	Tiempo de compensación
y	Valor de salida
x	Valor real

La regulabilidad de un sistema regulado con tiempo muerto está determinada por la relación existente entre T_t y T_g. El valor de T_t debe ser pequeño en relación a T_g. Rige:

$T_t/T_g \leq 1$

Velocidad de reacción de sistemas regulados

Los sistemas regulados se pueden catalogar en función de los valores siguientes:

T_u < 0,5 min, T_g < 5 min = sistema regulado rápido

T_u > 0,5 min, T_g > 5 min = sistema regulado lento

Valores característicos de algunos sistemas regulados

Magnitud física	Sistema regulado	Tiempo de retardo T_u	Tiempo de compensación T_g	Velocidad de aumento v_{max}
Temperatura	Horno pequeño calentado eléctricamente	0,5 hasta 1 min	5 hasta 15 min	hasta 60 K/min
	Horno de recocer grande calentado eléctricamente	1 hasta 5 min	10 hasta 20 min	hasta 20 K/min
	Horno de recocer grande calentado por gas	0,2 hasta 5 min	3 hasta 60 min	1 hasta 30 K/min
	Columna de destilación fraccionada	1 hasta 7 min	40 hasta 60 min	0,1 hasta 0,5 °C/s
	Autoclave (2,5 m^3)	0,5 hasta 0,7 min	10 hasta 20 min	ninguna indicación
	Autoclave de alta presión	12 hasta 15 min	200 hasta 300 min	ninguna indicación
	Recalentador de vapor	30 s hasta 2,5 min	1 hasta 4 min	2 °C/s
	Máquinas de moldeo por inyección	0,5 hasta 3 min	3 hasta 30 min	5 hasta 20 K/min
	Extrusora	1 hasta 6 min	5 hasta 60 min	
	Máquinas de envasado y embalaje	0,5 hasta 4 min	3 hasta 40 min	2 hasta 35 K/min
	Calefacción de locales	1 hasta 5 min	10 hasta 60 min	1 °C/min
Caudal	Tubería con gas	0 hasta 5 s	0,2 hasta 10 s	no aplicable
	Tubería con líquido	Sin	Sin	
Presión	Tubería de gas	Sin	0,1 s	no aplicable
	Caldera de tambor calentada con gas o fuel oil	Sin	150 s	no aplicable
	Caldera de tambor con molinos batidores	1 hasta 2 min	2 hasta 5 min	no aplicable
Nivel del depósito (nivel)	Caldera de tambor	0,6 hasta 1 min	ninguna indicación	0,1 hasta 0,3 cm/s
Velocidad	Accionamiento eléctrico pequeño	Sin	0,2 hasta 10 s	no aplicable
	Accionamiento eléctrico grande	Sin	5 hasta 40 s	no aplicable
	Turbina de gas	Sin	ninguna indicación	50 rpm
Tensión eléctrica	Generadores pequeños	Sin	1 hasta 5 s	no aplicable
	Generadores grandes	Sin	5 hasta 10 s	no aplicable

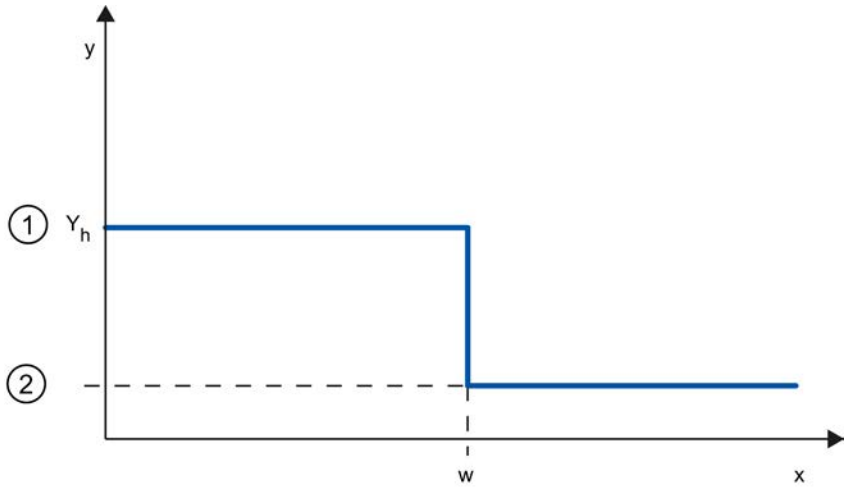
11.3.1.4 Regulador de impulsos

Regulador de dos puntos sin realimentación

Los reguladores de dos puntos tienen como función de conmutación el estado "ON" y "OFF". Esto corresponde al 100 % o bien al 0 % de la potencia. Este comportamiento origina una oscilación continua del valor real x en torno a la consigna w.

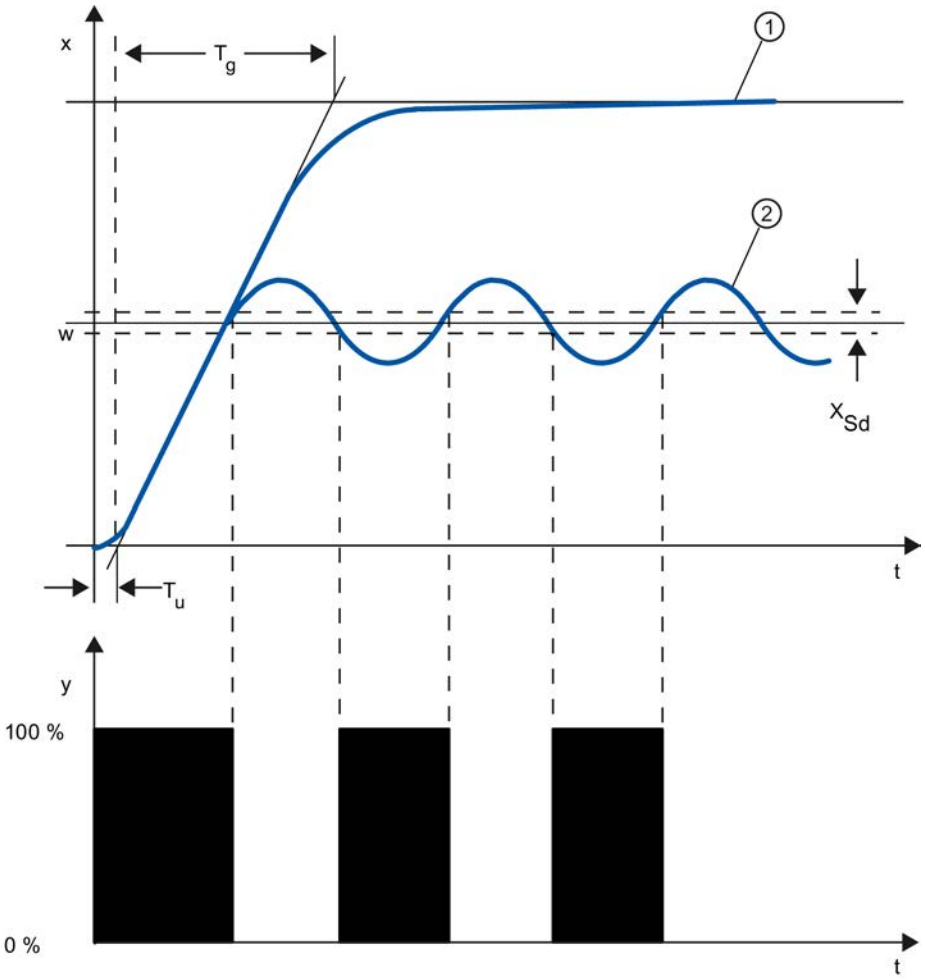
La amplitud y la duración de periodo de la oscilación aumentan con la relación existente entre el tiempo de retardo T_u y el tiempo de compensación T_g presente en el sistema regulado. Estos reguladores se utilizan principalmente en regulaciones de temperatura simples (p. ej. hornos eléctricos de calefacción directa) o como señalizadores de límite.

El gráfico siguiente muestra la curva característica de un regulador de dos puntos



①	ON
②	OFF
Y_h	Rango
w	Consigna

El gráfico siguiente muestra la función de regulación de un regulador de dos puntos



①	Respuesta transitoria sin regulador
②	Respuesta transitoria con regulador de dos puntos
T_u	Tiempo de retardo
T_g	Tiempo de compensación
X_{Sd}	Ganancia diferencial

Regulador de dos puntos con realimentación

El comportamiento de los reguladores de dos puntos en sistemas regulados que presentan grandes tiempos de retardo (p. ej. hornos en los que la capacidad útil está separada de la calefacción) se puede mejorar con realimentaciones electrónicas.

Con la realimentación se eleva la frecuencia de conmutación del regulador, con lo que se reduce la amplitud del valor real. Además, se pueden mejorar considerablemente los resultados de la regulación en el funcionamiento dinámico. El límite de la frecuencia de conmutación está definido por la etapa de salida. La frecuencia de conmutación de elementos finales de control mecánicos como relés y contactores no debe ser mayor de 1 a 5 conmutaciones por minuto. Para las salidas de corriente y tensión binarias con tiristores o triacs postconectados se pueden elegir altas frecuencias de conmutación ubicadas muy por encima de la frecuencia de corte del sistema regulado.

Dado que los impulsos de conmutación en la salida del sistema regulado ya no se pueden determinar, se obtienen resultados comparables a los de los reguladores continuos.

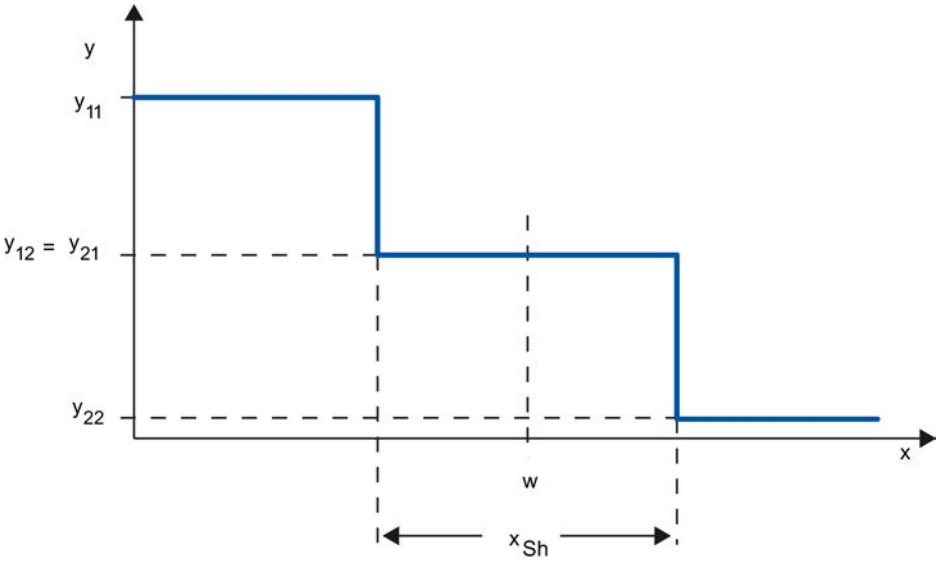
El valor de salida se genera por modulación del ancho de impulsos del valor de salida de un regulador continuo.

Los reguladores de dos puntos con realimentación se utilizan para regular la temperatura en hornos, máquinas para procesar plástico, textiles, papel, goma y alimentos, así como en enfriadores y calefactores.

Regulador de tres puntos

Los reguladores de tres puntos se utilizan en calefacción/refrigeración. Estos reguladores tienen dos puntos de conmutación como salida. Los resultados de la regulación se optimizan con estructuras de realimentación electrónicas. Los campos de aplicación de estos reguladores son las cámaras térmicas, frigoríficas y climatizadas, así como el precalentamiento de herramientas de máquinas transformadoras de plástico.

El gráfico siguiente muestra la curva característica de un regulador de tres puntos

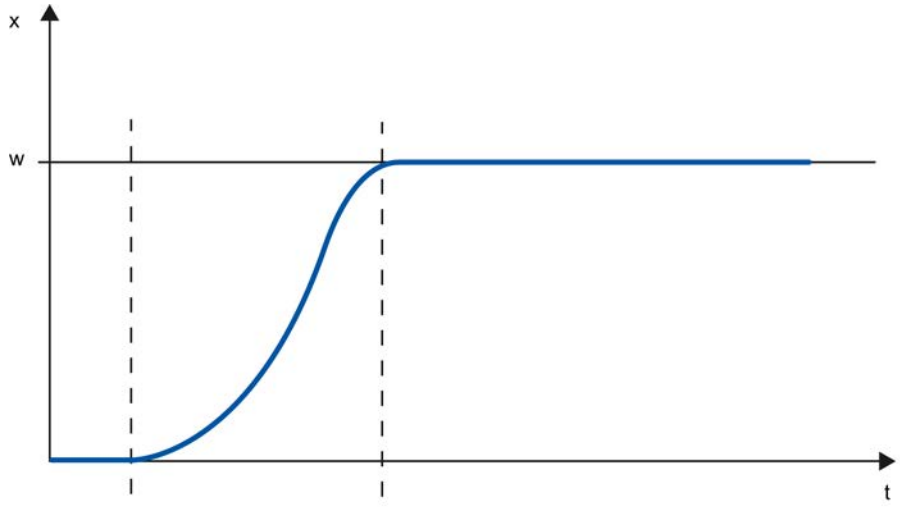


y	Valor de salida, p. ej. y11 = 100 % calefacción y12 = 0 % calefacción y21 = 0 % refrigeración y22 = 100 % refrigeración
x	Unidad física del valor real, p. ej. la temperatura en °C
w	Consigna
x_{Sh}	Distancia entre los puntos de conmutación 1 y 2

11.3.1.5 Respuesta de regulación y a perturbaciones

Respuesta de regulación

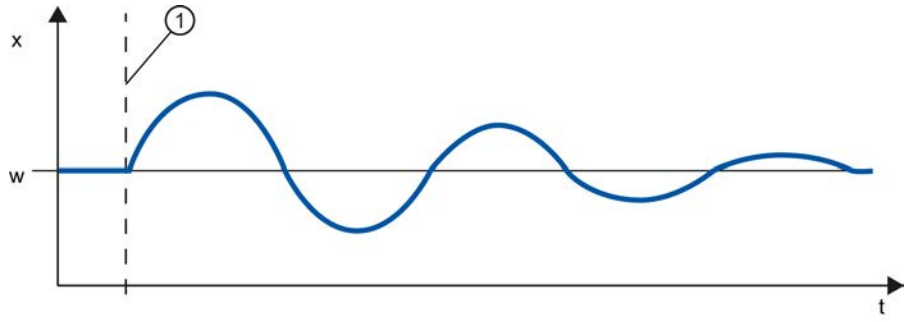
El valor real debe seguir un cambio de la consigna lo más rápido posible. La respuesta de regulación mejora cuanto más corto sea el tiempo necesario para alcanzar la nueva consigna y menor la fluctuación del valor real.



x	Valor real
w	Consigna

Respuesta a perturbaciones

La consigna se ve afectada por magnitudes perturbadoras. El regulador debe eliminar lo más rápido posible el error de regulación resultante. La respuesta a perturbaciones mejora cuanto más corto sea el tiempo necesario para alcanzar la nueva consigna consigna y menor la fluctuación del valor real.



x	Valor real
w	Consigna
①	Incidencia de una magnitud perturbadora

Una magnitud perturbadora se compensa con reguladores con acción I. La calidad de la regulación no se reduce por una magnitud perturbadora que incida constantemente, ya que el error de regulación es relativamente constante. Una magnitud perturbadora dinámica afecta en mayor medida a la calidad de la regulación porque el error de regulación fluctúa. Sólo a través de la acción I, que se forma lentamente, se vuelve a reducir el error de regulación.

Una magnitud perturbadora medible puede entrar en el lazo de regulación. Esto puede acelerar considerablemente la reacción del regulador.

11.3.1.6 Respuesta de regulación con diferentes estructuras de realimentación

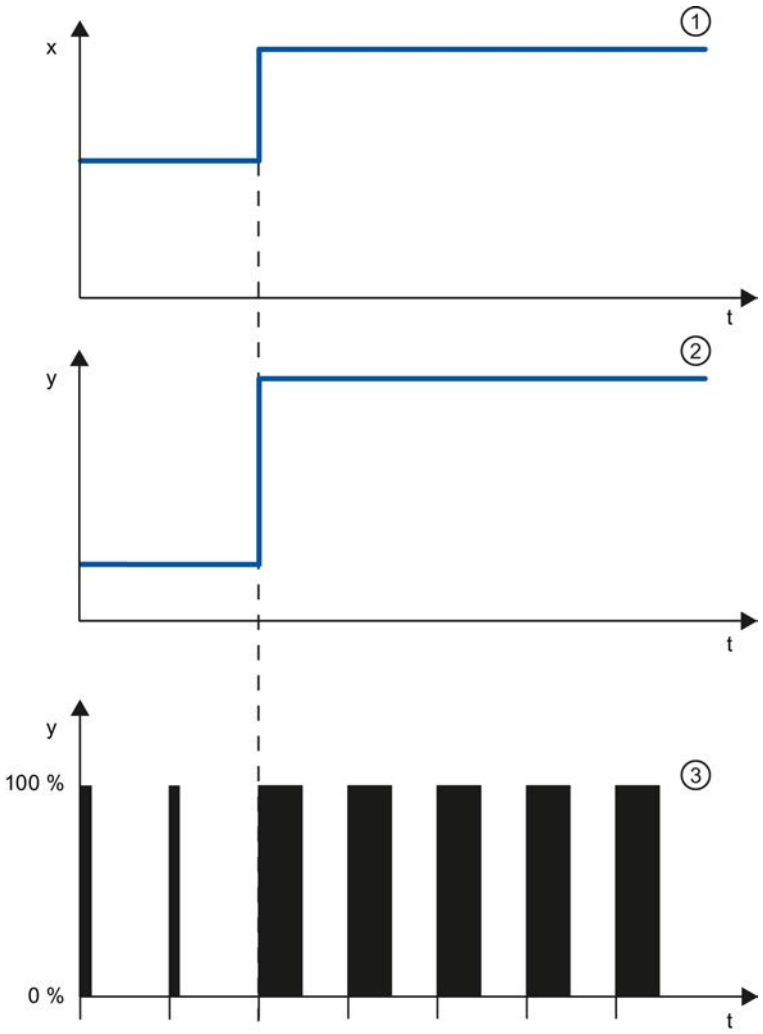
Comportamientos de regulador

Cuanto mayor sea la precisión de adaptación del regulador al comportamiento de tiempo del sistema regulado, mayor será la precisión de ajuste de la consigna, de modo que la respuesta a magnitudes perturbadores será óptima.

La inclusión en la realimentación puede tener un comportamiento proporcional (P), proporcional diferencial (PD), proporcional integral (PI) o proporcional integral diferencial (PID).

Si se le aplica una función de escalón al error de regulación, se generarán distintas respuestas indiciales en función del regulador.

Respuesta indicial de un regulador P



① Error de regulación
② Valor de salida de un regulador continuo
③ Valor de salida de un regulador de impulsos

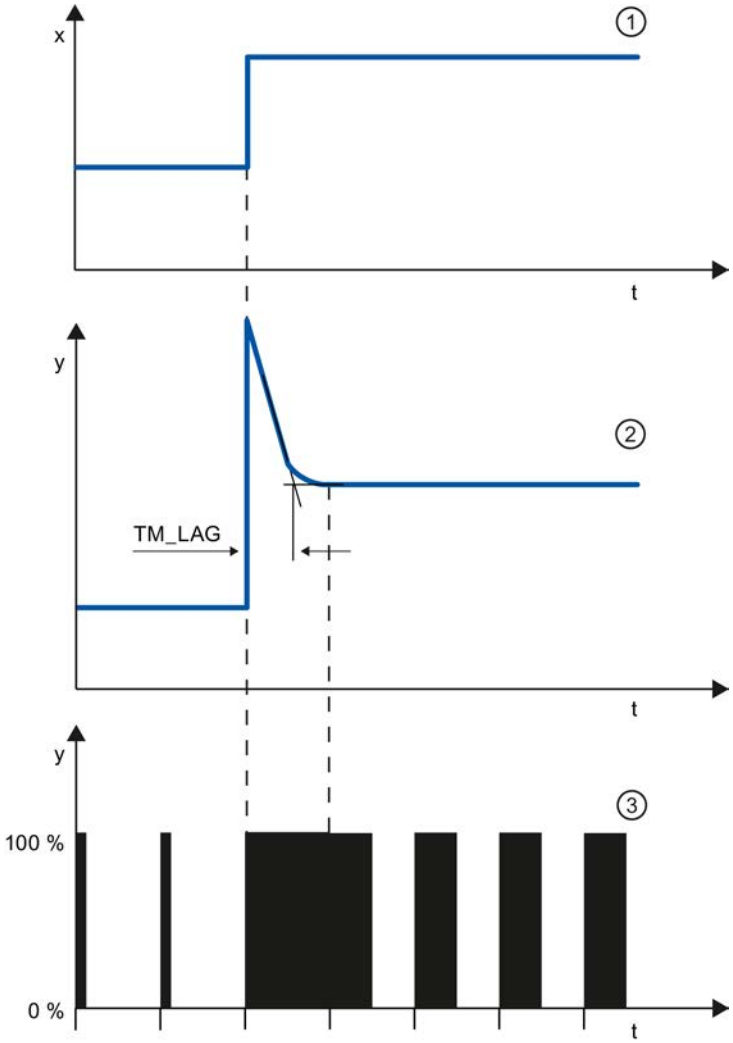
Fórmula para reguladores P

El valor de salida y el error de regulación son directamente proporcionales, es decir:

Valor de salida = ganancia proporcional × error de regulación

y = GAIN × x

Respuesta indicial de un regulador PD



①	Error de regulación
②	Valor de salida de un regulador continuo
③	Valor de salida de un regulador de impulsos
TM_LAG	Retardo de la acción D

Fórmula para reguladores PD

Para la respuesta indicial del regulador PD en el dominio temporal rige lo siguiente:

$$y = GAIN \cdot X_W \cdot \left(1 + \frac{TD}{TM_LAG} \cdot e^{\frac{-t}{TM_LAG}} \right)$$

t = duración desde el escalón del error de regulación

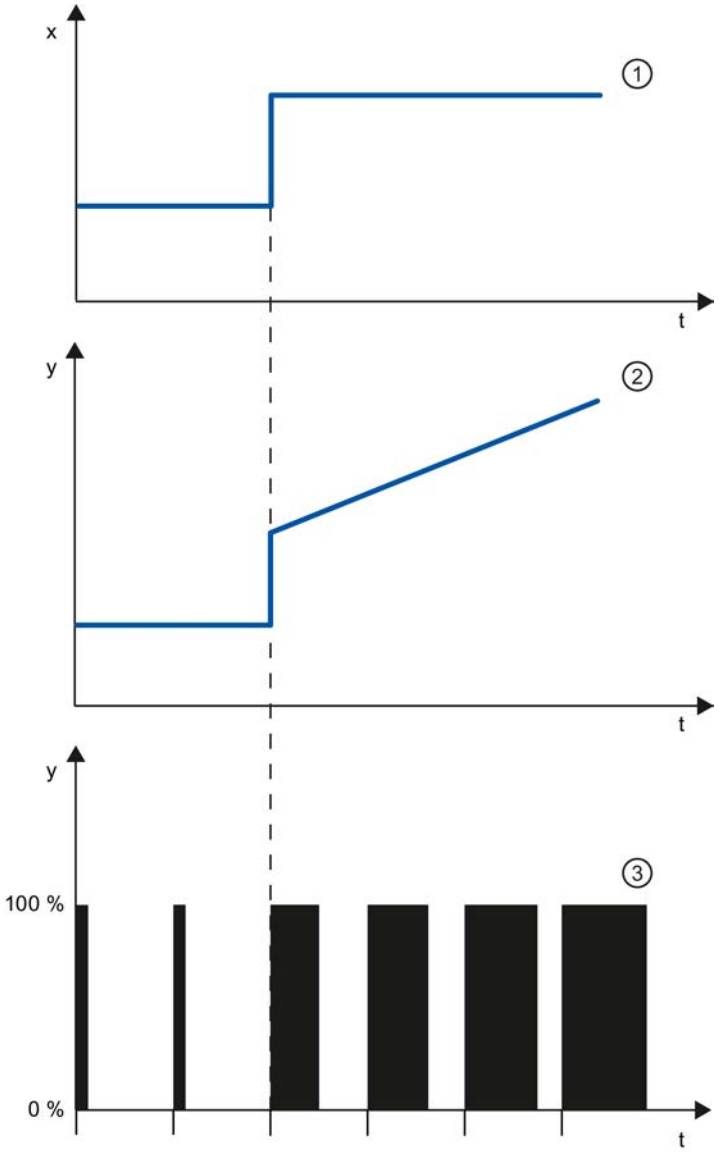
La acción D genera un valor de salida en función de la velocidad con la que cambia el valor real. Una acción D pura no es apta para la regulación, pues sólo un cambio del valor real

puede generar un cambio del valor de salida. Si el valor real permanece constante, el valor de salida tampoco cambia.

La acción D, en combinación con una acción P, mejora la respuesta a perturbaciones. Las perturbaciones no se eliminan por completo. El comportamiento beneficioso es el bueno y dinámico. Durante el desplazamiento y al cambiar la consigna se alcanza una transición bien amortiguada y exenta de vibraciones.

Un regulador con acción D no es apropiado si el sistema regulado tiene magnitudes de medición pulsantes, p. ej. en la regulación de presión o de caudal.

Respuesta indicial de un regulador PI



① Error de regulación

② Valor de salida de un regulador continuo

③ Valor de salida de un regulador de impulsos

Una acción I en el regulador totaliza el error de regulación en el tiempo. De este modo el regulador sigue reajustándose hasta eliminar el error de regulación. En un regulador P puro se presenta un error de regulación permanente. Éste se puede eliminar con una acción I en el regulador.

Dependiendo de los requisitos que deba cumplir la respuesta de regulación, en la práctica resulta idónea una combinación de las acciones P, I y D. El comportamiento de tiempo de las distintas acciones se puede describir con los siguientes parámetros del regulador: ganancia proporcional GAIN, tiempo de acción integral TI (acción I) y tiempo de anticipación TD (acción D).

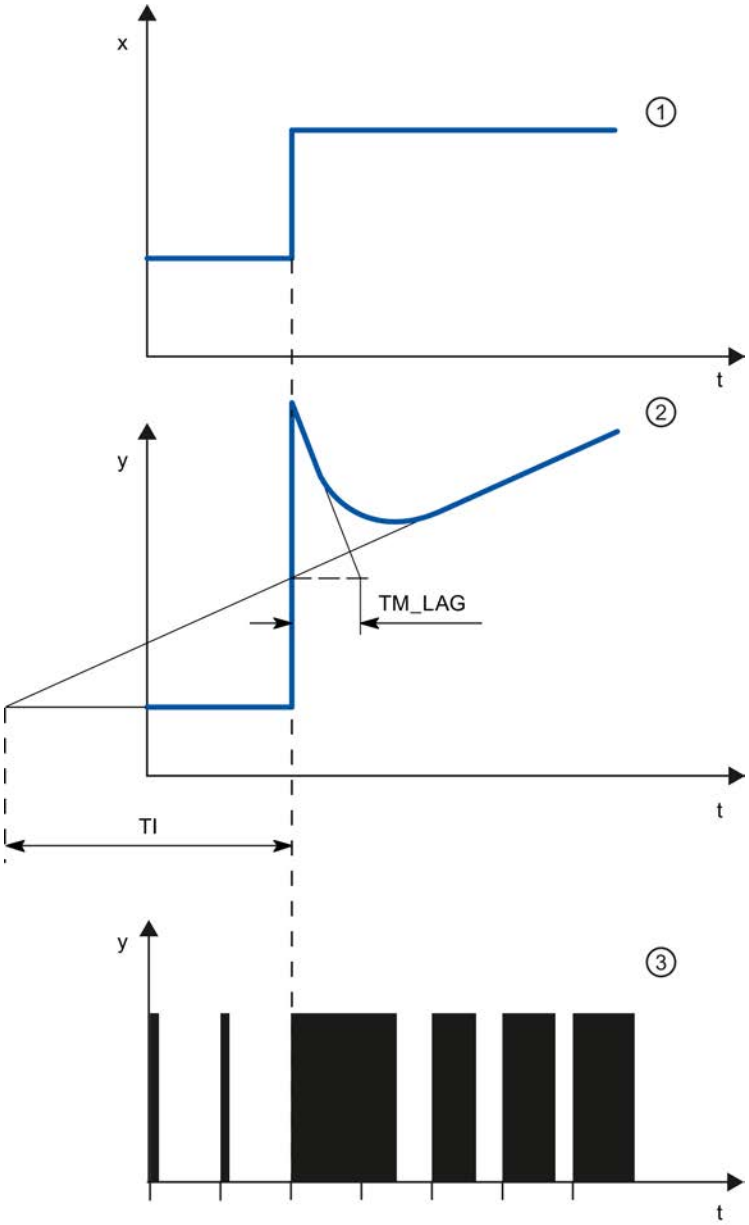
Fórmula para reguladores PI

Para la respuesta indicial del regulador PI en el dominio temporal rige lo siguiente:

$$y = GAIN \cdot X_W \cdot \left(1 + \frac{1}{TI \cdot t}\right)$$

t = duración desde el escalón del error de regulación

Respuesta indicial de un regulador PID



①	Error de regulación
②	Valor de salida de un regulador continuo
③	Valor de salida de un regulador de impulsos
TM_LAG	Retardo de la acción D
T_i	Tiempo de integración

Fórmula para reguladores PID

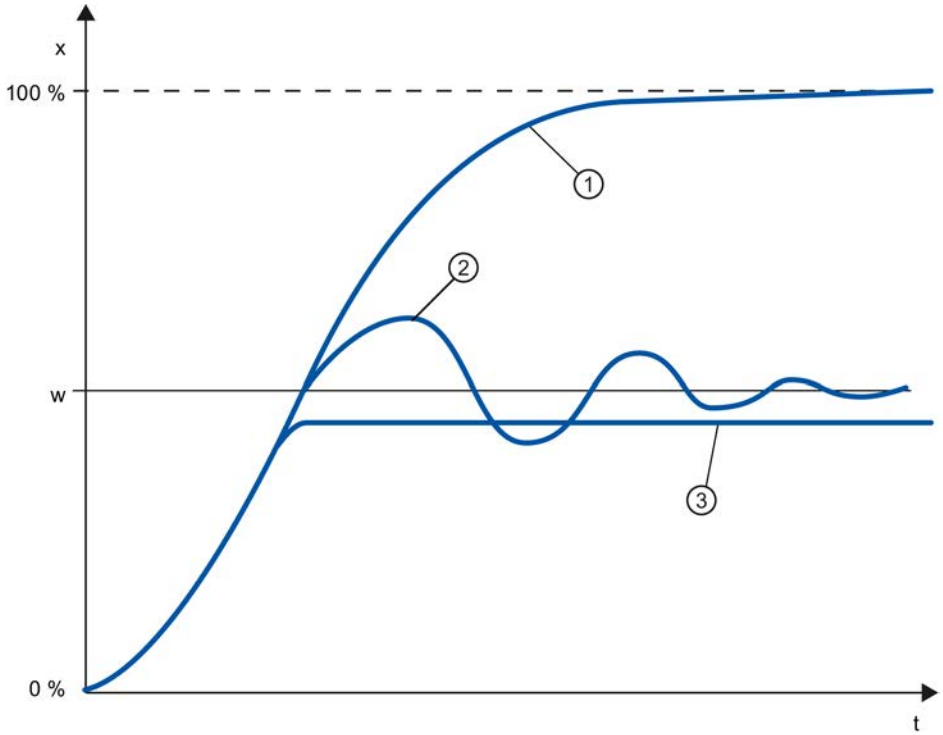
Para la respuesta indicial del regulador PID en el dominio temporal rige lo siguiente:

$$y = GAIN \cdot X_w \cdot \left(1 + \frac{1}{TI \cdot t} + \frac{TD}{TM_LAG} \cdot e^{\frac{-t}{TM_LAG}}\right)$$

t = duración desde el escalón del error de regulación

Comportamiento de un sistema regulado con distintas estructuras de regulación

La mayoría de regulaciones que se aplican en la técnica de procesos se puede controlar con un regulador con comportamiento PI. En sistemas regulados lentos con grandes tiempos de retardo, p. ej. regulación de temperatura, el resultado de la regulación se puede mejorar con un regulador con comportamiento PID.



①	Sin regulador
②	Regulador PID
③	Regulador PD
w	Consigna
x	Valor real

Los reguladores con comportamiento PI y PID ofrecen la ventaja de que tras la fase de estabilización el valor real no muestra ninguna desviación frente a la consigna. Durante el desplazamiento el valor real oscila sobre la consigna.

11.3.1.7 Elección de la estructura del regulador con un sistema regulado definido

Selección de las estructuras de regulador apropiadas

Un resultado de regulación óptimo sólo se puede alcanzar si se elige una estructura de regulador que sea afín al sistema regulado, y que se pueda adaptar al mismo dentro de determinados límites.

La tabla siguiente ofrece una vista general de las estructuras de regulador adecuadas en función del sistema regulado.

Sistema regulado		Estructura de regulador			
		P	PD	PI	PID
	Sólo con tiempo muerto	No apta	No apta	Apta	No apta
	PT1 con tiempo muerto	No apta	No apta	Muy apta	Muy apta
	PT2 con tiempo muerto	No apta	Apta con limitaciones	Muy apta	Muy apta
	Alto grado	No apta	No apta	Apta con limitaciones	Muy apta
	Sin compensación	Muy apta	Muy apta	Muy apta	Muy apta

La tabla siguiente ofrece una vista general de las estructuras de regulador adecuadas en función de la magnitud física.

Magnitud física	Estructura de regulador			
	P	PD	PI	PID
	Error de regulación permanente		Error de regulación no permanente	
Temperatura	Para exigencias reducidas y sistemas P con T_u/Tg < 0,1	Muy apta	Estructuras de regulador idóneas para exigencias elevadas (con excepción de reguladores especiales adaptados)	
Presión	Apta, si el tiempo de retardo no es significativo	No apta	Estructuras de regulador idóneas para exigencias elevadas (con excepción de reguladores especiales adaptados)	
Caudal	No apta, porque el rango GAIN requerido suele ser demasiado grande	No apta	Válido, pero el regulador I solo suele ser mejor	Casi no es necesaria

11.3.1.8 Ajuste de los parámetros PID

Fórmula empírica para ajustar los parámetros

Estructura de regulador	Ajuste
P	GAIN ≈ v_{max} × T_u [°C]
PI	GAIN ≈ 1,2 × v_{max} × T_u [°C]
PD	GAIN ≈ 0,83 × v_{max} × T_u [°C] TD ≈ 0,25 × v_{max} × T_u [min] TM_LAG ≈ 0,5 × TD[min]
PID	GAIN ≈ 0,83 × v_{max} × T_u [°C] TI ≈ 2 × T_u [min] TD ≈ 0,4 × T_u [min] TM_LAG ≈ 0,5 × TD[min]
PD/PID	GAIN ≈ 0,4 × v_{max} × T_u [°C] TI ≈ 2 × T_u [min] TD ≈ 0,4 × T_u [min] TM_LAG ≈ 0,5 × TD[min]

En vez de $v_{max} = \Delta_x / \Delta_t$ se puede utilizar X_{max} / T_g.

En los reguladores con estructura PID el ajuste del tiempo de acción integral está acoplado por lo general al ajuste del tiempo de anticipación.

La relación TI / TD es de 4 a 5 y por lo tanto óptima para la mayoría de sistemas regulados.

La inobservancia del tiempo de anticipación TD no es grave en los reguladores PD.

En los reguladores PI o PID ocurren oscilaciones en la regulación si el tiempo de acción integral TI se ha ajustado por debajo de la mitad del nivel requerido.

Un tiempo de acción integral demasiado grande retarda la eliminación de perturbaciones. No se puede pretender que los lazos de regulación trabajen óptimamente después de haber ajustado por primera vez los parámetros. La experiencia ha demostrado que es necesario efectuar un reajuste de parámetros si se trata de un sistema "difícil de regular" con T_u / T_g > 0,3.

11.3.2 Configuración de un regulador por software

11.3.2.1 Vista general de los reguladores de software

Para la configuración de un regulador por software, se requiere una instrucción con el algoritmo de regulación y un objeto tecnológico. El objeto tecnológico para un regulador de software equivale al DB de instancia de la instrucción. En el objeto tecnológico, se guarda la configuración del regulador. A diferencia de los DB de instancia de otras instrucciones, los objetos tecnológicos no se guardan junto a los recursos del programa, sino en CPU > Objetos tecnológicos.

Objetos tecnológicos e instrucciones

CPU	Librería	Instrucción	Objeto tecnológico	Descripción
S7-1200	Compact PID	PID_Compact V1.X	PID_Compact V1.X	Regulador PID universal con optimización integrada
S7-1200		PID_3Step V1.X	PID_3Step V1.X	Regulador PID para válvulas con optimización integrada
S7-1500 S7-1200 V4.x		PID_Compact V2.X	PID_Compact V2.X	Regulador PID universal con optimización integrada
S7-1500 S7-1200 V4.x		PID_3Step V2.X	PID_3Step V2.X	Regulador PID para válvulas con optimización integrada
S7-1500/300/400	Funciones básicas de PID	CONT_C	CONT_C	Regulador continuo
S7-1500/300/400		CONT_S	CONT_S	Regulador paso a paso para elementos finales de control de acción integrativa
S7-1500/300/400		PULSEGEN	-	Generador de impulsos para elementos finales de control de acción proporcional
S7-1500/300/400		TCONT_CP	TCONT_CP	Regulador de temperatura continuo con generador de impulsos
S7-1500/300/400		TCONT_S	TCONT_S	Regulador de temperatura para accionadores integrables
S7-300/400	PID Self Tuner	TUN_EC	TUN_EC	Optimización de un regulador continuo
S7-300/400		TUN_ES	TUN_ES	Optimización de un regulador paso a paso
S7-300/400	Standard PID Control (paquete opcional PID Professional)	PID_CP	PID_CP	Regulador continuo con generador de impulso
S7-300/400		PID_ES	PID_ES	Regulador paso a paso para elementos finales de control de acción integrativa
S7-300/400		LP_SCHED	-	Distribuir llamadas del regulador
S7-300/400	Modular PID Control (paquete opcional PID Professional)	A_DEAD_B	-	Filtrar señales perturbadoras del error de regulación
S7-300/400		CRP_IN	-	Escalar señal de entrada analógica
S7-300/400		CRP_OUT	-	Escalar señal de salida analógica
S7-300/400		DEAD_T	-	Emitir señal de entrada retardada
S7-300/400		DEADBAND	-	Inhibir pequeñas oscilaciones del valor real
S7-300/400		DIF	-	Diferenciar señal de entrada en el tiempo
S7-300/400		ERR_MON		Vigilar error de regulación
S7-300/400		INTEG	-	Integrar señal de entrada en el tiempo
S7-300/400		LAG1ST	-	Elemento de retardo de primer orden
S7-300/400		LAG2ND	-	Elemento de retardo de segundo orden
S7-300/400		LIMALARM	-	Notificar valores límite
S7-300/400		LIMITER	-	Limitar la variable manipulada
S7-300/400		LMNGEN_C	-	Determinar variable manipulada para regulador continuo
S7-300/400		LMNGEN_S	-	Determinar variable manipulada para regulador paso a paso
S7-300/400		NONLIN	-	Linealizar señal del encóder

CPU	Librería	Instrucción	Objeto tecnológico	Descripción
S7-300/400		NORM	-	Normalizar el valor real físicamente
S7-300/400		OVERRIDE	-	Aplicar variables manipuladas de 2 reguladores PID en 1 actuador
S7-300/400		PARA_CTL	-	Conmutar juegos de parámetros
S7-300/400		PID	-	Algoritmo PID
S7-300/400		PUSLEGEN_M	-	Generar impulsos para actuadores de acción proporcional
S7-300/400		RMP_SOAK	-	Especificar consignas según perfil
S7-300/400		ROC_LIM	-	Limitar la velocidad de cambio
S7-300/400		SCALE_M	-	Escalar valor real
S7-300/400		SP_GEN	-	Especificar consigna manualmente
S7-300/400		SPLT_RAN	-	Dividir rangos de variable manipulada
S7-300/400		SWITCH	-	Aplicar valores analógicos
S7-300/400		LP_SCHED_M	-	Distribuir llamadas del regulador

11.3.2.2 Pasos para la configuración de un regulador por software

Todos los reguladores de software se proyectan según el mismo esquema:

Paso	Descripción
1	Agregar objeto tecnológico (Página 9937)
2	Configurar objeto tecnológico (Página 9938)
3	Llamar la instrucción en el programa de usuario (Página 9939)
4	Cargar objeto tecnológico en el dispositivo (Página 9940)
5	Poner en marcha el regulador de software (Página 9942)
6	Guardar los parámetros PID optimizados en el proyecto (Página 9942)
7	Comparar valores (Página 9943)
8	Mostrar instancias de un objeto tecnológico (Página 9964)

11.3.2.3 Agregar objetos tecnológicos

Agregar objeto tecnológico en el árbol del proyecto

Al agregar un objeto tecnológico, se genera un DB de instancia de la instrucción para este objeto tecnológico. En él se guarda la configuración del objeto tecnológico.

Requisitos

Hay un proyecto creado con una CPU.

Procedimiento

Para agregar un objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Objetos tecnológicos".
3. Haga doble clic en "Agregar objeto".
   Se abre el cuadro de diálogo "Agregar objeto".
4. Haga clic en el botón "Regulador PID".
   Se muestran todos los reguladores PID disponibles para esta CPU.
5. Seleccione la instrucción para el objeto tecnológico, p. ej., PID_Compact.
6. En el campo de entrada "Nombre", introduzca un nombre personalizado para el objeto tecnológico.
7. Seleccione la opción "manual" si desea modificar el número de bloque de datos propuesto para el DB de instancia.
8. Haga clic en "Más información" para guardar información propia sobre el objeto tecnológico.
9. Confirme con "OK".

Resultado

El nuevo objeto tecnológico se genera y se guarda en la capeta "Objetos tecnológicos" del árbol del proyecto. El objeto tecnológico se utiliza, cuando la instrucción para este objeto tecnológico se llama en un OB de alarma cíclica.

Nota

Existe la posibilidad de activar la casilla de verificación "Agregar nuevo y abrir" situada en la parte inferior del diálogo. De ese modo, la configuración del objeto tecnológico se abrirá una vez agregado.

11.3.2.4 Configurar objetos tecnológicos

Puede configurar las propiedades de un objeto tecnológico en una CPU S7-1200 de dos maneras.

- En la ventana de inspección del editor de programación
- En el editor de configuración

Puede configurar las características de un objeto tecnológico en una CPU S7-300/400 sólo en el editor de configuración.

Ventana de inspección del editor de programación

En la ventana de inspección del editor de programación, sólo es posible configurar los parámetros que son necesarios para el funcionamiento.

También en el modo en línea, se muestran los valores fuera de línea de los parámetros. Los valores en línea se pueden modificar sólo en la ventana de puesta en servicio.

Para abrir la ventana de inspección del objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta "Bloques de programa" en el árbol del proyecto.
2. Haga doble clic en el bloque (OB de alarma cíclica) en el que llama la instrucción del regulador de software.
   El bloque se abre en el área de trabajo.
3. Haga clic en la instrucción del regulador de software.
4. En la ventana de inspección, seleccione una tras otra las fichas "Propiedades" y "Configuración".

Ventana de configuración

Para cada objeto tecnológico, hay una ventana de configuración específica, en la que se configuran todas las características.

Para abrir la ventana de configuración de un objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Configuración".

Símbolos

Los símbolos que aparecen en la navegación local de la configuración y de la ventana de inspección muestran más detalles sobre la integridad de la configuración:

	La configuración contiene valores predeterminados y está completa. La configuración sólo contiene valores predeterminados. Con ellos es posible utilizar el objeto tecnológico sin más modificaciones.
	La configuración contiene valores definidos por el usuario y está completa Todos los campos de entrada de la configuración contienen valores válidos y se ha modificado como mínimo un valor predeterminado.
	La configuración es incompleta o errónea Como mínimo un campo de entrada o una lista desplegable está vacío o contiene un valor no válido. El campo o la lista desplegable en cuestión se marca en rojo. Al hacer clic, el roll out con el aviso de error le indica la causa del error.

La características de un objeto tecnológico se describen con detalle en el capítulo para el objeto tecnológico.

11.3.2.5 Llamar la instrucción en el programa de usuario

La instrucción del regulador de software se debe llamar en el OB de alarma cíclica. El tiempo de muestreo del regulador de software se determina a partir del intervalo transcurrido entre las llamadas en el OB de alarma cíclica.

Requisitos

El OB de alarma cíclica está creado y el tiempo de ciclo del OB de alarma cíclica está configurado correctamente.

Procedimiento

Para llamar la instrucción en el programa de usuario, proceda del siguiente modo:

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Árbol del proyecto".
3. Haga doble clic en el OB de alarma cíclica.
   El bloque se abre en el área de trabajo.
4. Abra, en la ventana "Instrucciones", el grupo "Tecnología" y la carpeta "PID Control".
   La carpeta contiene todas las instrucciones para los reguladores del software que se pueden configurar en la CPU.
5. Seleccione una instrucción y arrástrela al OB de alarma cíclica correspondiente mediante Drag & Drop.
   Se abre el cuadro de diálogo "Opciones de llamada".
6. En la lista "Nombre" seleccione un objeto tecnológico o introduzca el nombre del nuevo objeto tecnológico.

Resultado

Si el objeto tecnológico no existe todavía, éste se agrega. La instrucción se agrega en el OB de alarma cíclica. El objeto tecnológico está asignado a esta llamada de la instrucción.

11.3.2.6 Cargar objetos tecnológicos en el dispositivo

Para poder utilizar una configuración nueva o modificada en el modo online es necesario cargarla en la CPU. Al cargar datos remanentes, hay que tener en cuenta las siguientes particularidades:

- Software (solo modificaciones)
  - S7-1200, S7-1500:
    Los datos remanentes se conservan.
  - S7-300/400:
    Los datos remanentes se actualizan de inmediato. La CPU no pasa a Stop.
- Cargar e inicializar programa PLC en el dispositivo
  - S7-1200, S7-1500:
    Los datos remanentes se actualizan con la siguiente transición de Stop a RUN. El programa PLC solo puede cargarse completo.
  - S7-300/400:
    Los datos remanentes se actualizan con la siguiente transición de Stop a RUN.

Cargar datos remanentes en una CPU S7-1200 o S7-1500

Nota

La carga e inicialización del programa PLC con la instalación en marcha puede causar importantes daños materiales y personales en caso de fallos de funcionamiento o errores de programa.

Asegúrese de que no puedan darse situaciones de peligro antes de cargar e inicializar el programa PLC.

Para cargar datos remanentes, proceda del siguiente modo:

1. Seleccione la entrada de la CPU en el árbol del proyecto.
2. Seleccione el comando "Cargar e inicializar programa PLC en el dispositivo" en el menú "Online".
   - Si hasta ahora no se ha establecido ninguna conexión online, se abre el cuadro de diálogo "Carga avanzada". Ajuste en este caso todos los parámetros necesarios para la conexión y haga clic en "Cargar".
   - Una vez definida la conexión online, los datos de proyecto se compilan si es necesario y se abre el cuadro de diálogo "Cargar vista preliminar". En este cuadro de diálogo se muestran avisos y se indican acciones necesarias para la carga.
3. Compruebe los avisos.
   Cuando la carga pueda efectuarse, el botón "Cargar" se activa.
4. Haga clic en "Cargar".
   El programa PLC completo se carga y se abre el cuadro de diálogo "Cargar resultados". Este cuadro de diálogo muestra el estado y las acciones tras el proceso de carga.
5. Si los módulos deben volver a arrancarse directamente tras la carga, active la casilla de verificación "Arrancar todos".
6. Cierre el cuadro de diálogo "Cargar resultados" con "Finalizar".

Resultado

El programa PLC completo se carga en el dispositivo. Se borran los bloques que solo existen online en el dispositivo. Cargando todos los bloques implicados y borrando los bloques no necesarios en el dispositivo, se evitan incoherencias entre los bloques en el programa de usuario.

Los avisos de la ventana de inspección (en "Información > General") indican si el proceso de carga ha finalizado correctamente.

11.3.2.7 Poner en marcha el regulador de software

Procedimiento

Para abrir la zona de trabajo "Puesta en servicio" de un objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Puesta en servicio".

Las funciones para la puesta en servicio son específicas para cada regulador y se describen allí.

11.3.2.8 Guardar los parámetros PID optimizados en el proyecto

El regulador de software se optimiza en la CPU. De este modo, los valores ya no coinciden con su proyecto en el DB de instancia en la CPU.

Para actualizar los parámetros PID en el proyecto con los parámetros PID optimizados en la CPU, proceda del siguiente modo:

Requisitos

- Hay establecida una conexión online con la CPU y esta se encuentra en el estado operativo "RUN".
- Las funciones de la ventana de puesta en servicio se han habilitado con el botón "Iniciar".

Procedimiento

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Objetos tecnológicos".
3. Abra un objeto tecnológico.
4. Haga doble clic en "Puesta en servicio".
5. Haga clic en el símbolo "Cargar parámetros PID".
6. Guarde el proyecto.

Resultado

Los parámetros PID efectivos en ese momento se guardan en los datos del proyecto. Al cargar nuevamente los datos de proyecto en la CPU, se utilizan los parámetros optimizados.

11.3.2.9 Comparar valores

Visualización de comparación y condiciones

La función "Comparar valores" ofrece las siguientes opciones:

- Comparación de los valores de arranque configurados del proyecto con los valores de arranque de la CPU y los valores actuales
- Procesamiento directo de los valores actuales y de los valores de arranque del proyecto
- Detección y visualización inmediatas de errores de introducción con ayuda para su corrección
- Copia de seguridad de los valores actuales en el proyecto
- Transferencia de los valores de arranque del proyecto a la CPU como valores actuales

Símbolos y elementos de control

Están disponibles los siguientes símbolos y elementos de control:

Símbolo	Función
	El valor de arranque en la CPU es igual al valor de arranque configurado en el proyecto
	El valor de arranque en la CPU es diferente al valor de arranque configurado en el proyecto
	No se puede realizar la comparación entre valor de arranque en la CPU y valor de arranque configurado en el proyecto
	Por lo menos uno de los dos valores de comparación es incorrecto desde el punto de vista tecnológico o sintáctico.
	Los valores actuales se transmiten al proyecto offline
	Los valores de arranque actualizados en el proyecto se transfieren a la CPU (inicializar valores de ajuste)
	Se abre el cuadro de diálogo "Comparar valores"

Condiciones

La función "Comparar valores" está disponible de forma ilimitada para S7-1200 y S7-1500.

Para S7-300 y S7-400 rige la siguiente limitación:

En el modo de observación una S7-300/S7-400 no puede transferir los valores de arranque a la CPU. Estos valores no pueden mostrarse online con "Comparar valores".

Los valores actuales del objeto tecnológico se muestran y se pueden modificar directamente.

Comparar valores

A continuación se muestra el procedimiento con el ejemplo de los "parámetros PID".

Requisitos

- Está configurado un proyecto con un regulador de software.
- El proyecto está cargado en la CPU.
- En el navegador de proyecto está abierto el diálogo de configuración.

Procedimiento

1. Abra el regulador de software deseado en el árbol del proyecto.
2. Haga doble clic en el objeto "Configuración".
3. Navegue al cuadro de diálogo "Parámetros PID" en la ventana de configuración.
4. Haga clic en el símbolo para activar el modo de observación.
   Se muestran los símbolos y elementos de control (Página 9940) de la función "Comparar valores" detrás de los parámetros.
5. Haga clic en el campo de entrada del parámetro deseado y modifique manualmente los valores de parámetro mediante entrada directa.
   - Si el fondo del campo de entrada es gris, los valores solo son legibles, no modificables.
   - Para modificar los valores en el cuadro de diálogo "Parámetros PID", active primero la entrada manual haciendo clic en la casilla de verificación "Activar entrada manual".
6. Haga clic en el símbolo para abrir el cuadro de diálogo de los valores de arranque.
   Este cuadro de diálogo muestra dos valores del parámetro:
   - Valor de arranque en la CPU: En la parte superior se muestra el valor de arranque en la CPU.
   - Valor de arranque en el proyecto: En la parte inferior se muestra el valor de arranque configurado en el proyecto.
7. Introduzca el valor deseado en el campo de entrada para el proyecto.

Detección

Se detecta la entrada de valores incorrectos. En ese caso, se ofrece ayuda para la corrección.

Si introduce un valor sintácticamente erróneo, debajo del parámetro se abre una ventana desplegable con el correspondiente aviso de error. No se aplica el valor medido incorrecto.

Si introduce un valor tecnológicamente incorrecto, se abre un cuadro de diálogo en el que se notifica el error y se muestra información para la corrección:

- Haciendo clic en "No" puede adoptar la corrección y corregir su entrada.
- Haciendo clic en "OK" se aplica el valor incorrecto.

ATENCIÓN
Mal funcionamiento del regulador Los valores tecnológicamente incorrectos pueden producir un mal funcionamiento del regulador.

Copia de seguridad de los valores actuales

Haciendo clic en el símbolo se transfieren los valores actuales del regulador a los valores de arranque de su proyecto configurado.

Transferir valores de proyecto a la CPU

Haciendo clic en el símbolo se transfieren los valores configurados de su proyecto a la CPU.

PRECAUCIÓN

Evitar daños personales y materiales

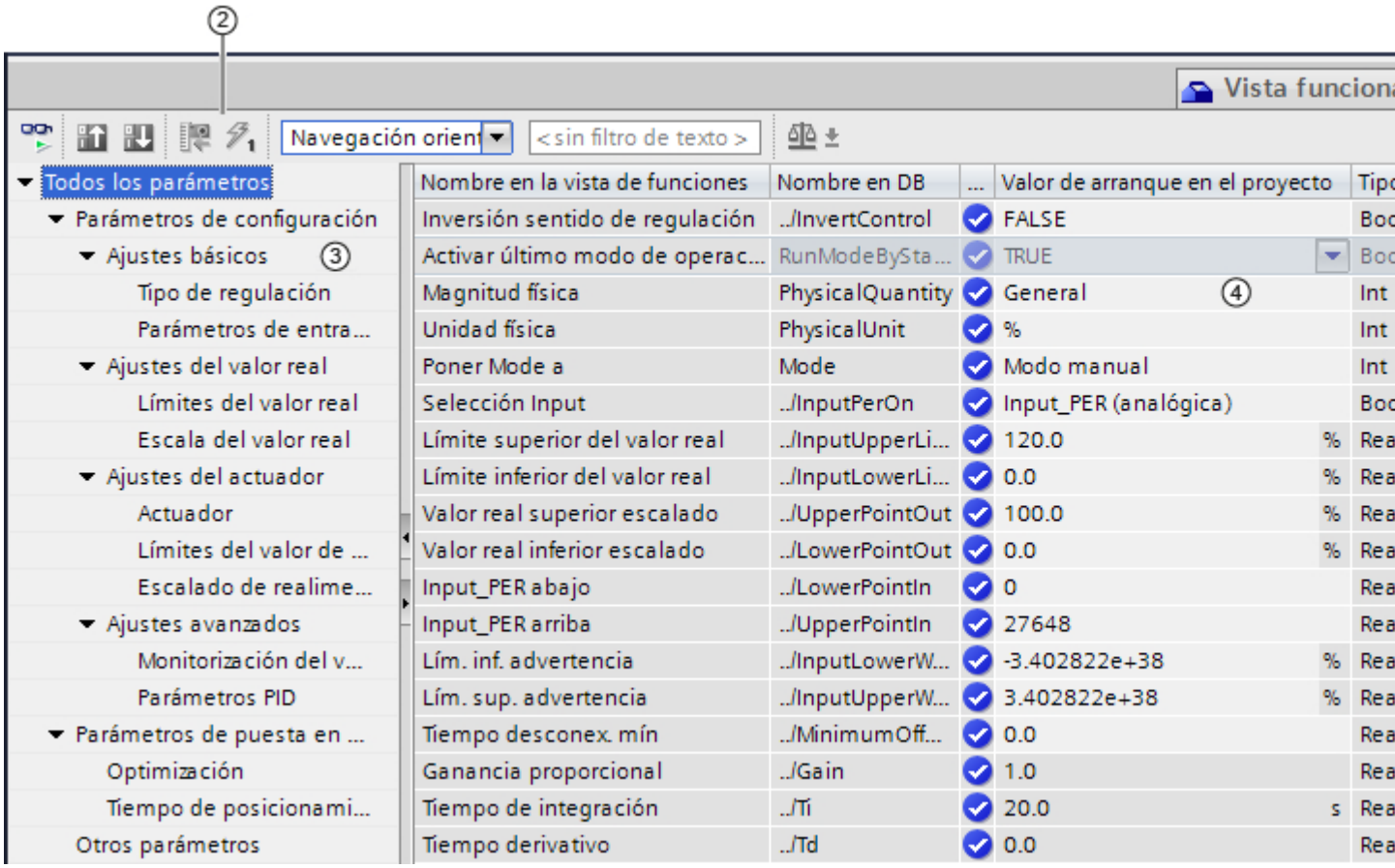
La carga e inicialización del programa de usuario en la instalación en marcha puede causar importantes daños materiales y personales en caso de fallos de funcionamiento o errores de programa.

Asegúrese de que no puedan darse situaciones de peligro antes de cargar e inicializar el programa de usuario.

11.3.2.10 Vista de parámetros

Introducción a la vista de parámetros

La vista de parámetros le ofrece una visión de conjunto de todos los parámetros relevantes de un objeto tecnológico. Proporciona una vista general de los ajustes de los parámetros, de manera que usted puede modificarlos cómodamente en el modo offline y online.



① Ficha "Vista de parámetros"

② Barra de herramientas (Página 9948)

③ Navegación (Página 9949)

④ Tabla de parámetros (Página 9949)

Funcionalidad

Para analizar los parámetros de los objetos tecnológicos, y para poderlos observar y forzar de manera selectiva, hay disponibles las siguientes funciones.

Funciones de indicación:

- Indicación de los valores de parámetros en modo offline y online
- Indicación de la información de estado de los parámetros

- Indicación de desviaciones de los valores y posibilidad de corregirlas directamente
- Indicación de errores de configuración
- Indicación de variaciones de valores como consecuencia de dependencias de parámetros
- Indicación de todos los valores almacenados de un parámetro: valor de arranque en la CPU, valor de arranque en el proyecto, valor de observación
- Indicación de la comparación de parámetros de los valores almacenados de un parámetro

Funciones de manejo:

- Navegación, para cambiar rápidamente entre los parámetros y las estructuras de parámetros.
- Filtro de texto, para encontrar más rápidamente determinados parámetros.
- Función de ordenación, para adaptar el orden de parámetros y grupos de parámetros en función de las necesidades.
- Función de almacenamiento, para guardar ajustes estructurales de la vista de parámetros.
- Observar y forzar online valores de parámetro.
- Guardar instantánea de valores de parámetros de la CPU para reproducir situaciones transitorias y reaccionar a ellas.
- Aplicar instantánea de valores de parámetros como valores de arranque.
- Cargar en la CPU valores de arranque modificados.
- Funciones de comparación, para comparar entre sí valores de parámetros.

Validez

La vista de parámetros está disponible para los siguientes objetos tecnológicos:

- PID_Compact
- PID_3Step
- CONT_C (solo S7-1500)
- CONT_S (solo S7-1500)
- TCONT_CP (solo S7-1500)
- TCONT_S (solo S7-1500)
- TO_Axis_PTO (S7-1200 Motion Control)
- TO_Positioning_Axis (S7-1200 Motion Control)
- TO_CommandTable_PTO (S7-1200 Motion Control)
- TO_CommandTable (S7-1200 Motion Control)

Estructura de la vista de parámetros

Barra de herramientas

En la barra de herramientas de la vista de parámetros pueden seleccionarse las siguientes funciones:

Símbolo	Función	Explicación
	Observar todos	Inicia la observación de los parámetros visibles en la vista de parámetros activa (modo online).
	Crear instantánea de los valores de observación y aplicar valores de ajuste de esa instantánea como valores de arranque	Aplica los valores de observación actuales a la columna "Instantánea" y actualiza los valores de arranque en el proyecto. Solo en modo online con PID_Compact y PID_3Step.
	Inicializar valores de ajuste	Transfiere a la CPU los valores de arranque actualizados en el proyecto. Solo en modo online con PID_Compact y PID_3Step.
	Crear instantánea de los valores de observación	Aplica los valores de observación actuales a la columna "Instantánea". Solo en modo online.
	Forzar una vez e inmediatamente todos los parámetros seleccionados	Este comando se ejecuta una sola vez y lo antes posible, sin referencia a un punto determinado del programa de usuario. Solo en modo online.
Navegación orient	Seleccionar la estructura de navegación	Cambia entre la navegación orientada a funciones y la navegación orientada a datos.
Navegación orient	Filtro de texto...	Tras introducir una cadena de caracteres: indicación de todos los parámetros que contienen la cadena de caracteres introducida en una de las columnas visibles actualmente.
	Seleccionar valores de comparación	Selección de los valores de parámetros que deben compararse entre sí en modo online (valor de arranque en el proyecto, valor de arranque en la CPU, instantánea) Solo en modo online.
	Memorizar disposición	Guarda los ajustes de visualización de la vista de parámetros realizados (p. ej., estructura de navegación seleccionada, columnas activadas de la tabla, etc.).

Navegación

Dentro de la ficha "Vista de parámetros" pueden seleccionarse como alternativa las siguientes estructuras de navegación:

Navegación		Explicación
Navegación orientada a funciones	Todos los parámetros Parámetros de configuración Parámetros de puesta en servicio Otros parámetros	En la navegación orientada a funciones, la estructura de los parámetros se basa en la estructura de los cuadros de diálogo de configuración (ficha "Vista de funciones"), puesta en marcha y diagnóstico. El último grupo "Otros parámetros" contiene los restantes parámetros del objeto tecnológico.
Navegación orientada a datos	Todos los parámetros Input Output InOut Static Otros parámetros	En la navegación orientada a datos, la estructura de los parámetros se basa en la estructura del DB de instancia. El último grupo "Otros parámetros" contiene los parámetros que no están incluidos en el DB de instancia.

Con la lista desplegable "Seleccionar la estructura de navegación" puede cambiar la estructura de navegación.

Tabla de parámetros

La tabla siguiente muestra el significado de las distintas columnas de la tabla de parámetros. Las columnas pueden mostrarse u ocultarse según convenga.

- Columna "Offline" = X: la columna es visible en modo offline.
- Columna "Online" = X: la columna es visible en modo online (conexión online con la CPU).

Columna	Explicación	Offline	Online
Nombre en la vista de funciones	Nombre del parámetro en la vista de funciones. Este campo de visualización está vacío en el caso de parámetros que no se configuran por medio del objeto tecnológico.	X	X
Nombre completo en DB	Ruta completa del parámetro en el DB de instancia. Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Nombre en DB	Nombre del parámetro en el DB de instancia. Si el parámetro es parte de una estructura o UDT, se añade el prefijo ". ./". Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Estado de la configuración	Indicación de la integridad de la configuración mediante símbolos de estado ver Estado de la configuración (offline) (Página 9957)	X	
Resultado de la comparación	Resultado de la función "Comparar valores". Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. ver Comparar valores (Página 9940)		X

Columna	Explicación	Offline	Online
Valor de arranque en el proyecto	Valor de arranque configurado en el proyecto. Indicación de errores en caso de valores mal introducidos desde el punto de vista sintáctico o tecnológico.	X	X
Valor predeterminado	Valor predeterminado del parámetro. Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Instantánea	Instantánea de los valores actuales en la CPU (valores de observación). Indicación de errores en caso de valores incorrectos desde el punto de vista tecnológico.	X	X
Valor de arranque en la CPU	Valor de arranque en la CPU. Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. Indicación de errores en caso de valores incorrectos desde el punto de vista tecnológico.		X
Valor de observación	Valor actual en la CPU. Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. Indicación de errores en caso de valores incorrectos desde el punto de vista tecnológico.		X
Valor de forzado	Valor con que debe modificarse el valor de observación. Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. Indicación de errores en caso de valores mal introducidos desde el punto de vista sintáctico o tecnológico.		X
Selección del valor de forzado	Selección de los valores de forzado que deben transferirse por medio del botón "Forzar una vez e inmediatamente todos los parámetros seleccionados". Esta columna aparece junto con la columna "Valor de forzado".		X
Valor mínimo	Valor tecnológico más bajo del parámetro. Si el valor mínimo depende de otros parámetros, entonces viene determinado: • Offline: por los valores de arranque en el proyecto. • Online: por los valores de observación.	X	X
Valor máximo	Valor tecnológico más alto del parámetro. Si el valor máximo depende de otros parámetros, entonces viene determinado: • Offline: por los valores de arranque en el proyecto. • Online: por los valores de observación.	X	X
Valor de ajuste	Identifica el parámetro como valor de ajuste. Estos parámetros pueden inicializarse online.	X	X
Tipo de datos	Tipo de datos del parámetro. Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Remanencia	Identifica el valor como remanente. Los valores de los parámetros remanentes se conservan incluso después de desconectar la tensión de alimentación.	X	X
Accesible desde HMI	Indica si HMI puede acceder a este parámetro durante el tiempo de ejecución.	X	X
Visible en HMI	Indica si el parámetro es visible por defecto en la lista de selección de HMI.	X	X
Comentario	Descripción abreviada del parámetro.	X	X

Abrir la vista de parámetros

Requisitos

El objeto tecnológico se ha agregado al navegador de proyectos, lo que significa que se ha generado el correspondiente DB de instancia de la instrucción.

Procedimiento

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Configuración".
4. Seleccione en la esquina superior derecha la ficha "Vista de parámetros".

Resultado

Se abre la vista de parámetros. En la tabla de parámetros cada parámetro mostrado viene está representado por una fila de la tabla.

Las propiedades visualizables de los parámetros (columnas de la tabla) dependen de si la vista de parámetros está en modo offline u online.

Además, es posible mostrar y ocultar las distintas columnas de la tabla de manera selectiva.

Consulte también

Ajuste predeterminado de la vista de parámetros (Página 9951)

Ajuste predeterminado de la vista de parámetros

Ajustes predeterminados

Para trabajar de forma efectiva con la vista de parámetros, puede adaptar la representación de los parámetros y guardar los ajustes realizados.

Pueden efectuarse y guardarse los siguientes ajustes:

- Mostrar y ocultar columnas
- Modificar el ancho de las columnas
- Modificar el orden de las columnas
- Cambiar la navegación
- Seleccionar un grupo de parámetros en la navegación
- Seleccionar valores de comparación

Mostrar y ocultar columnas

Para mostrar y ocultar columnas en la tabla de parámetros, proceda del siguiente modo:

1. Sitúe el puntero del ratón en el encabezado de la tabla de parámetros.
2. En el menú contextual, seleccione el comando "Mostrar/ocultar".
   Se muestra la selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación de la columna.
4. Para ocultar una columna, desactive la casilla de verificación de la columna.

o bien

1. Sitúe el puntero del ratón en el encabezado de la tabla de parámetros.
2. En el menú contextual, seleccione el comando "Mostrar todas las columnas" si desea mostrar todas las columnas del modo offline u online.

Algunas columnas solo pueden mostrarse en modo online: ver Tabla de parámetros (Página 9946).

Modificar el ancho de las columnas

Para adaptar el ancho de una columna al contenido de manera que pueda leerse todo el texto de las filas, proceda del siguiente modo:

1. Sitúe el puntero del ratón en el encabezado de la tabla de parámetros, a la derecha junto a la columna que se desea optimizar, hasta que el puntero del ratón adopte la forma de una cruz.
2. Haga doble clic en este punto.

o bien

1. Abra el menú contextual del encabezado de la tabla de parámetros.
2. Haga clic en
   - "Optimizar ancho de columna" o
   - "Optimizar ancho de todas las columnas".

Situando brevemente el puntero del ratón sobre el campo en cuestión, aparece el contenido completo de los distintos campos en las columnas de ancho insuficiente.

Modificar el orden de las columnas

Las columnas de la tabla de parámetros pueden situarse donde se desee.

Para modificar el orden de las columnas, proceda del siguiente modo:

1. Haga clic sobre el encabezado de la columna, arrástrelo y suéltelo en el lugar deseado. Cuando suelte el botón del ratón, la columna se anclará en la nueva posición.

Cambiar la navegación

Para cambiar la estructura de visualización de los parámetros, proceda del siguiente modo:

1. Seleccione la navegación que desee en la lista desplegable "Seleccionar la estructura de navegación":
   - Navegación orientada a datos
   - Navegación orientada a funciones

Ver también Navegación (Página 9946).

Seleccionar un grupo de parámetros en la navegación

Dentro de la navegación elegida puede optar entre la visualización "Todos los parámetros" o la visualización de un grupo de parámetros subordinado deseado.

1. Haga clic sobre el grupo de parámetros deseado en la navegación.
   En la tabla de parámetros se muestran solo los parámetros del grupo de parámetros.

Seleccionar valores de comparación (online)

Para ajustar los valores de comparación para la función "Comparar valores", proceda del siguiente modo:

1. Seleccione en la lista desplegable "Seleccionar valores de comparación" los valores de comparación que desee:
   - Valor de arranque en el proyecto / valor de arranque en la CPU
   - Valor de arranque en el proyecto / instantánea
   - Valor de arranque en la CPU / instantánea

Por defecto está ajustada la opción "Valor de arranque en el proyecto / Valor de arranque en la CPU".

Guardar el ajuste predeterminado de la vista de parámetros

Para guardar las modificaciones anteriores de la vista de parámetros, proceda del siguiente modo:

1. Adapte la vista de parámetros en función de sus necesidades.
2. Haga clic en el botón "Memorizar disposición" situado en la parte superior derecha de la vista de parámetros.

Trabajar con la vista de parámetros

Sinopsis

La siguiente tabla ofrece una vista general de las funciones de la vista de parámetros que se describen a continuación en los modos online y offline.

- Columna "Offline" = X: esta función está disponible en modo offline.
- Columna "Online" = X: esta función está disponible en modo online.

Función/acción	Offline	Online
Filtrar la tabla de parámetros (Página 9954)	X	X
Ordenar la tabla de parámetros (Página 9955)	X	X
Aplicar datos de parámetros en otros editores (Página 9955)	X	X
Mostrar errores (Página 9956)	X	X
Editar valores de arranque en el proyecto (Página 9956)	X	X
Estado de la configuración (offline) (Página 9957)	X	
Observar online los valores en la vista de parámetros (Página 9958)		X
Crear instantánea de los valores de observación (Página 9959)		X
Forzar valores (Página 9960)		X
Comparar valores (Página 9961)		X
Aplicar valores como valores de arranque desde el programa online (Página 9962)		X
Inicializar los valores de ajuste en el programa online (Página 9963)		X

Filtrar la tabla de parámetros

Los parámetros de la tabla de parámetros pueden filtrarse de las siguientes maneras:

- Con el filtro de texto
- Con los subgrupos de la navegación

Los dos métodos de filtrado pueden utilizarse simultáneamente.

Con el filtro de texto

Puede filtrarse por los textos que son visibles en la tabla de parámetros. Esto significa que solo se puede filtrar por los textos de las líneas de parámetros y columnas mostradas.

1. Introduzca en el campo de entrada "Filtro de texto..." la cadena de caracteres por la que desee filtrar.
   La tabla de parámetros solo muestra aquellos parámetros que contienen la cadena de caracteres.

El filtro de texto se restablece:

- Seleccionando otro grupo de parámetros en la navegación.
- Cambiando entre la navegación orientada a datos y a funciones.

Con los subgrupos de la navegación

1. Haga clic sobre el grupo de parámetros deseado en la navegación, p. ej., "Static".
   En la tabla de parámetros se muestran solo los parámetros "Static". En algunos grupos de la navegación puede seleccionar otros subgrupos.
2. Haga clic sobre "Todos los parámetros" en la navegación si desea volver a mostrar todos los parámetros.

Ordenar la tabla de parámetros

Los valores de los parámetros están dispuestos por filas. La tabla de parámetros puede ordenarse según cualquiera de las columnas mostradas.

- En las columnas con valores numéricos se ordena según la altura del valor numérico.
- En las columnas con texto se ordena alfabéticamente.

Ordenar por columnas

1. Sitúe el puntero del ratón en el encabezado de la columna deseada.
   El fondo de esta celda se marca en azul.
2. Haga clic sobre el encabezado de la columna.

Resultado

Toda la tabla de parámetros se ordena según la columna marcada. En el encabezado de la columna aparece un triángulo con la punta hacia arriba.

Haciendo clic de nuevo sobre el encabezado de la columna, el orden cambia de la siguiente manera:

- Símbolo "▲": la tabla de parámetros se ordena de manera ascendente.
- Símbolo "▼": la tabla de parámetros se ordena de manera descendente.
- Ningún símbolo: se anula la ordenación. La tabla de parámetros adopta la vista predeterminada.

En la ordenación se ignora el prefijo "../" de la columna "Nombre en DB".

Aplicar datos de parámetros en otros editores

Una vez seleccionada toda una fila de parámetros de la tabla de parámetros, con

- arrastrar y soltar
- <Ctrl+C>/<Ctrl+V>
- copiar/pegar por medio del menú contextual

pueden aplicarse parámetros en los siguientes editores del TIA Portal:

- en el editor de programas;
- en la tabla de observación;
- en la tabla de señales para Trace.

El parámetro se añade con el nombre completo: ver indicación en la columna "Nombre completo en DB".

Mostrar errores

Señalización de errores

Los errores de parametrización que conllevan errores de compilación (p. ej., rebase de límite), se indican en la vista de parámetros.

Cada vez que se introduce un valor en la vista de parámetros, se comprueba y muestra de inmediato la corrección tecnológica y sintáctica.

Los valores erróneos se indican con:

- Un símbolo de error rojo en las columnas "Estado de la configuración" (modo offline) o "Resultado de la comparación" (modo online, en función del tipo de comparación elegido)

y/o

- El campo de la tabla con el fondo rojo
  Al hacer clic sobre el campo erróneo, se muestra el mensaje de error desplegable que indica el rango de valores admisible o la sintaxis requerida (formato)

Errores de compilación

Desde el mensaje de error del compilador puede abrirse directamente la vista de parámetros (navegación orientada a funciones) con el parámetro causante del error para aquellos parámetros que no se muestren en el cuadro de diálogo de configuración.

Editar valores de arranque en el proyecto

Con la vista de parámetros pueden editarse los valores de arranque en el proyecto en los modos offline y online:

- Los valores se modifican en la columna "Valor de arranque en el proyecto" de la tabla de parámetros.
- En la columna "Estado de la configuración" de la tabla de parámetros se muestra el progreso de la configuración mediante los símbolos de estado conocidos del cuadro de diálogo de configuración del objeto tecnológico.

Condiciones

- Si hay otros parámetros que dependen del parámetro cuyo valor de arranque se ha modificado, el valor de arranque del parámetro dependiente se adapta igualmente.
- Si un parámetro de un objeto tecnológico no es editable, tampoco lo es en la vista de parámetros. La posibilidad de edición de un parámetro puede depender también de los valores de otros parámetros.

Definir nuevos valores de arranque

Para especificar valores de arranque para parámetros en la vista de parámetros, proceda del siguiente modo:

1. Abra la vista de parámetros del objeto tecnológico.
2. Introduzca en la columna "Valor de arranque en el proyecto" los valores de arranque deseados. El valor debe ser conforme al tipo de datos del parámetro y no debe rebasar el rango de valores del parámetro.
   En las columnas "Valor máximo" y "Valor mínimo" se muestran los valores límite del rango de valores.

En la columna "Estado de la configuración" se muestra el "progreso" de la configuración mediante símbolos de colores.

Ver también Estado de la configuración (offline) (Página 9957)

Una vez adaptados los valores de arranque y cargado el objeto tecnológico en la CPU, los parámetros adoptan durante el arranque el valor definido, siempre y cuando estos parámetros estén declarados como remanentes (columna "Remanencia").

Señalización de errores

Cuando se introduce un valor de arranque, se comprueba y muestra de inmediato la corrección tecnológica y sintáctica:

Los valores de arranque erróneos se indican con

- Un símbolo de error rojo en las columnas "Estado de la configuración" (modo offline) o "Resultado de la comparación" (modo online, en función del tipo de comparación elegido)

y/o

- Un fondo rojo en el campo "Valor de arranque en el proyecto"
  Al hacer clic sobre el campo erróneo, se muestra el mensaje de error desplegable que indica el rango de valores admisible o la sintaxis requerida (formato)

Corregir valores de arranque erróneos

1. Corrija los valores de arranque erróneos con ayuda de la información del mensaje de error desplegable.
   El símbolo de error rojo, el fondo rojo del campo y el mensaje de error desplegable dejarán de mostrarse.

El proyecto solo puede compilarse correctamente con valores de arranque correctos.

Estado de la configuración (offline)

El estado de la configuración se indica mediante símbolos:

- En la columna "Estado de la configuración" en la tabla de parámetros
- En la estructura de navegación de la navegación orientada a funciones o la navegación orientada a datos

Símbolo en la columna "Estado de la configuración"

Símbolo	Significado
	El valor de arranque del parámetro coincide con el valor predeterminado y es válido. El usuario todavía no ha definido ningún valor de arranque.
	El valor de arranque del parámetro contiene un valor definido por el usuario. El valor de arranque difiere del valor predeterminado. El valor de arranque no contiene errores y es válido.
	El valor de arranque del parámetro no es válido (error sintáctico o tecnológico). El campo de entrada tiene fondo rojo. Al hacer clic, el roll out con el aviso de error indica la causa del error.
	Solo con S7-1200 Motion Control: El valor de arranque del parámetro es válido, pero contiene advertencias. El campo de entrada tiene fondo amarillo.

Símbolo en la navegación

Los símbolos en la navegación muestran el "progreso" de la configuración de la misma manera que en el cuadro de diálogo de configuración del objeto tecnológico.

Ver Configurar objetos tecnológicos (Página 9935)

Observar online los valores en la vista de parámetros

Es posible observar directamente en la vista de parámetros los valores que los parámetros del objeto tecnológico van adoptando actualmente en la CPU (valores de observación).

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.

Procedimiento

1. Inicie la observación haciendo clic en el símbolo .
   En cuanto la vista de parámetros esté online, se mostrarán también las siguientes columnas:
   - Resultado de la comparación
   - Valor de arranque en la CPU
   - Valor de observación
   - Valor de forzado
   - Selección del valor de forzado

   La columna "Valor de observación" muestra los valores actuales de los parámetros en la CPU.
   Significado de las demás columnas: ver Tabla de parámetros (Página 9946)
2. La observación se finaliza haciendo clic de nuevo en el símbolo .

Indicación

Todas las columnas disponibles únicamente online tienen fondo naranja:

- Los valores de las celdas naranja claro se pueden modificar.
- Los valores de las celdas con fondo naranja oscuro no admiten modificaciones.

Crear instantánea de los valores de observación

Es posible guardar los valores actuales del objeto tecnológico en la CPU (valores de observación) y mostrarlos en la vista de parámetros.

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.

Procedimiento

Para mostrar los valores actuales de los parámetros, proceda del siguiente modo:

1. Haga clic en la vista de parámetros sobre el símbolo "Crear instantánea de los valores de observación".

Resultado

Los valores de observación actuales se aplican una vez en la columna "Instantánea" de la tabla de parámetros.

Los valores "congelados" así pueden analizarse mientras que los valores de observación de la columna "Valores de observación" se siguen actualizando.

Forzar valores

Con la vista de parámetros puede forzar los valores del objeto tecnológico en la CPU.

Puede asignar valores al parámetro (valor de forzado) una sola vez y forzarlos de inmediato. Al ejecutar esta función la orden de forzado se lleva a cabo lo antes posible, sin referencia a un punto determinado del programa de usuario.

PELIGRO

Peligro en el forzado:

Una modificación de los valores de los parámetros con la instalación en marcha puede causar importantes daños materiales y personales en caso de fallos de funcionamiento o errores de programa.

Asegúrese de que no puedan darse situaciones de peligro antes de ejecutar la función "Forzar".

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.
- El parámetro puede forzarse (el campo correspondiente en la columna "Valor de forzado" tiene fondo naranja claro).

Procedimiento

Para forzar parámetros inmediatamente, proceda del siguiente modo:

1. Introduzca los valores de forzado deseados en la columna "Valores de forzado" de la tabla de parámetros.
2. Compruebe si se ha activado la casilla de verificación para el forzado en la columna "Selección del valor de forzado".
   Los valores de forzado y las casillas de verificación correspondientes de los parámetros dependientes se adaptan también automáticamente.
3. Haga clic en el símbolo "Forzar una vez e inmediatamente todos los parámetros seleccionados".

Los parámetros seleccionados se fuerzan una vez e inmediatamente con los valores especificados y pueden observarse en la columna "Valores de observación". Las casillas de verificación para el forzado en la columna "Selección del valor de forzado" se desactivan automáticamente una vez ejecutada la orden de forzado.

Señalización de errores

Cuando se introduce un valor de forzado, se comprueba y muestra de inmediato la corrección tecnológica y sintáctica:

Los valores de forzado erróneos se indican con

- Un fondo rojo en el campo "Valor de forzado"

y

- Al hacer clic sobre el campo erróneo, se muestra el mensaje de error desplegable que indica el rango de valores admisible o la sintaxis requerida (formato)

Valores de forzado erróneos

- Los valores de forzado erróneos desde el punto de vista tecnológico pueden transferirse.
- Los valores de forzado erróneos desde el punto de vista sintáctico **no** pueden transferirse.

Comparar valores

Las funciones de comparación permiten comparar los siguientes valores almacenados de un parámetro:

- Valor de arranque en el proyecto
- Valor de arranque en la CPU
- Instantánea

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").

- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.

Procedimiento

Para comparar los valores de arranque de los diversos sistemas de destino, proceda del siguiente modo:

1. Haga clic en el símbolo "Seleccionar valores de comparación".
   Se abre una lista de selección con las opciones de comparación:
   – Valor de arranque en el proyecto - valor de arranque en la CPU (ajuste predeterminado)
   – Valor de arranque en el proyecto - instantánea
   – Valor de arranque en la CPU - instantánea
2. Elija la opción de comparación que desee.
   La opción de comparación seleccionada se ejecuta de la siguiente manera:
   – En las celdas del encabezado de las dos columnas seleccionadas para la comparación aparece un símbolo con una balanza.
   – En la columna "Resultado de la comparación" se muestra mediante símbolos el resultado de la comparación de las columnas seleccionadas.

Símbolo en la columna "Resultado de la comparación"

Símbolo	Significado
	Los valores de comparación son iguales y no contienen errores.
	Los valores de comparación son distintos y no contienen errores.
	Por lo menos uno de los dos valores de comparación es incorrecto desde el punto de vista tecnológico o sintáctico.
	No se ha podido llevar a cabo la comparación. Por lo menos uno de los dos valores de comparación no está disponible (p. ej., instantánea).

Símbolo en la navegación

Los símbolos se muestran de la misma manera en la navegación cuando el resultado de la comparación se cumple como mínimo para uno de los parámetros de debajo de la estructura de navegación mostrada.

Aplicar valores como valores de arranque desde el programa online

Para aplicar al proyecto en un paso valores optimizados de la CPU como valores de arranque, genere una instantánea de los valores de observación. A continuación, los valores de la instantánea marcados como "Valor de ajuste" se aplicarán como valores de arranque en el proyecto.

Requisitos

- El objeto tecnológico es de tipo "PID_Compact" o "PID_3Step".
- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.

Procedimiento

Para aplicar valores optimizados de la CPU, proceda del siguiente modo:

1. Haga clic en el símbolo "Crear instantánea de los valores de observación y aplicar valores de ajuste de esa instantánea como valores de arranque".

Resultado

Los valores de observación actuales se aplican a la columna "Instantánea" y sus valores de ajuste se copian como valores de arranque nuevos en la columna "Valor de arranque en el proyecto".

Nota

Aplicar valores de parámetros individuales

También es posible aplicar los valores de determinados parámetros que no estén marcados como valor de ajuste de la columna "Instantánea" a la columna "Valores de arranque en el proyecto". Para ello copie los valores con ayuda de los comandos "Copiar"·y "Pegar" del menú contextual y péguelos en la columna "Valor de arranque en el proyecto".

Inicializar los valores de ajuste en el programa online

Todos los parámetros marcados en la vista de parámetros como "Valor de ajuste" se pueden inicializar en la CPU con nuevos valores en un solo paso. Para ello se deben cargar los valores de arranque del proyecto en la CPU. La CPU se mantiene en el estado operativo "RUN".

Para evitar una pérdida de datos en la CPU al arrancar en frío o al reiniciar (arranque en caliente), es necesario cargar también el objeto tecnológico en la CPU.

> ⚠ PELIGRO
>
> **Peligro al modificar valores de parámetros**
>
> Una modificación de los valores de los parámetros con la instalación en marcha puede causar importantes daños materiales y personales en caso de fallos de funcionamiento o errores de programa.
>
> Asegúrese de que no puedan darse situaciones de peligro antes de reinicializar los valores de ajuste.

Requisitos

- El objeto tecnológico es de tipo "PID_Compact" o "PID_3Step".
- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.
- Los parámetros marcados como "Valor de ajuste" disponen de un "Valor de arranque en el proyecto" sin errores tecnológicos o sintácticos.

Procedimiento

Para inicializar todos los valores de ajuste, proceda del siguiente modo:

1. Introduzca los valores deseados en la columna "Valor de arranque en el proyecto". Compruebe que los valores de arranque no contengan errores tecnológicos o sintácticos.
2. Haga clic en el símbolo "Inicializar valores de ajuste".

Resultado

Los valores de ajuste de la CPU se inicializan con los valores de arranque del proyecto.

11.3.2.11 Mostrar el DB de instancia de un objeto tecnológico

Para cada objeto tecnológico, se crea un DB de instancia, en el que están guardados los parámetros y las variables estáticas.

Procedimiento

Para mostrar el DB de instancia de un objeto tecnológico, proceda del siguiente modo:

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Objetos tecnológicos".
3. Marque un objeto tecnológico.
4. Seleccione la orden "Abrir en el editor DP" en el menú contextual.

11.3.3 Regulación PID (S7-1200, S7-1500)

11.3.3.1 Utilizar PID_Compact

Objeto tecnológico PID_Compact

El objeto tecnológico PID_Compact ofrece un regulador PID continuo con optimización integrada. Asimismo, también es posible configurar un regulador de impulsos. Es posible elegir entre el modo manual y el automático.

PID-Compact registra de forma continua el valor real medido dentro de un lazo de regulación y lo compara con la consigna deseada. A partir del error de regulación resultante, la instrucción PID_Compact calcula un valor de salida, con el que el valor real se iguala con la consigna con la máxima rapidez y estabilidad. En los reguladores PID, el valor de salida se compone de tres acciones:

- Acción **P**
  La acción P del valor de salida aumenta proporcionalmente al error de regulación.
- Acción **I**
  La acción I del valor de salida aumenta hasta que se compensa el error de regulación.
- Acción **D**
  La acción D aumenta con una velocidad de variación creciente del error de regulación. El valor real se iguala lo más rápidamente posible con la consigna. Si la velocidad de variación del error de regulación vuelve a reducirse, también lo hace la acción D.

La instrucción PID_Compact calcula los parámetros P, I y D para su sistema regulado de forma autónoma durante la optimización inicial. Los parámetros pueden optimizarse aún más a través de una optimización fina. No es necesario determinar los parámetros manualmente.

Información adicional

- Vista general de los reguladores de software (Página 9932)
- Agregar objetos tecnológicos (Página 9934)
- Configurar objetos tecnológicos (Página 9935)
- Configurar PID_Compact V2 (Página 9966)
- Configurar PID_Compact V1 (Página 9981)

PID_Compact V2

Configurar PID_Compact V2

Ajustes básicos

Introducción

Configure las propiedades siguientes del objeto tecnológico PID_Compact en el área de ajustes básicos de la ventana de inspección o de configuración.

- Magnitud física
- Sentido de regulación
- Comportamiento en arranque después de un reset
- Consigna (sólo en la ventana de inspección)
- Valor real (sólo en la ventana de inspección)
- Valor de salida (sólo en la ventana de inspección)

Consigna, valor real y valor de salida

La consigna, el valor real y el valor de salida sólo pueden configurarse en la ventana de inspección del editor de programación. Para cada salida digital elija la fuente:

- DB de instancia
  Se utiliza el valor que se encuentra almacenado en el DB de instancia.
  El valor debe actualizarse en el DB de instancia del programa de usuario.
  En la instrucción no puede haber ningún valor.
  Modificación posible desde HMI.
- Instrucción
  Se utiliza el valor que se conecta con la instrucción.
  Cada vez que se llama la instrucción, el valor se escribe en el DB de instancia.
  No se puede modificar desde HMI.

Tipo de regulación

Magnitud física

En el grupo "Tipo de regulación", seleccione la magnitud física y la unidad para la consigna, el valor real y la magnitud perturbadora. La consigna, el valor real y la magnitud perturbadora se mostrarán en dicha unidad.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal.

PID_Compact no funciona con ganancia proporcional negativa. Para reducir el valor real con un valor de salida más elevado, active la casilla de verificación "Inversión sentido de regulación".

Ejemplos

- Al abrir una válvula de escape se reduce el nivel de llenado de un recipiente.
- Si se aumenta la potencia de refrigeración, disminuye la temperatura.

Comportamiento en arranque

1. Para cambiar inmediatamente al modo de operación "Inactivo" tras el rearranque de la CPU, desactive la casilla de verificación "Activar Mode tras rearrancar la CPU".
   Para cambiar inmediatamente, tras el rearranque de la CPU, al modo de operación almacenado en Mode, active la casilla de verificación "Activar Mode tras rearrancar la CPU".
2. En la lista desplegable "Poner Mode a", seleccione el modo de operación que debe activarse tras una carga completa en el dispositivo.
   Tras una carga completa en el dispositivo, PID_Compact arranca en el modo de operación seleccionado. Con cada rearranque, PID_Compact arranca en el último modo de operación almacenado en Mode.

Ejemplo

Se ha activado la casilla de verificación "Activar Mode tras rearrancar la CPU" y se ha seleccionado la entrada "Optimización inicial" en la lista "Poner Mode a". Tras una carga completa en el dispositivo, PID_Compact arranca en el modo de operación "Optimización inicial". Si la optimización inicial aún está activa y se rearranca la CPU, PID_Compact arranca de nuevo en el modo de operación "Optimización inicial". Si la optimización inicial ha finalizado correctamente, el modo automático aún está activo y se rearranca la CPU, PID_Compact arranca en "Modo automático".

Consigna

Procedimiento

Para especificar una consigna fija, proceda del siguiente modo:

1. Seleccione "DB de instancia".
2. Introduzca una consigna, como puede ser 80 °C.
3. En caso necesario, elimine una entrada de la instrucción.

Para especificar una consigna variable, proceda del siguiente modo:

1. Seleccione "Instrucción".
2. Indique el nombre de la variable REAL en la que está almacenada la consigna.
   Es posible asignar varios valores controlados por programa a la variable REAL para, por ejemplo, modificar la consigna de tiempo de forma controlada.

Valor real

Si utiliza directamente el valor de la entrada analógica, PID_Compact escala el valor de la entrada analógica al tamaño físico.

Si se desea acondicionar el valor de la entrada analógica, es preciso escribir un programa propio para su acondicionamiento. Por ejemplo, el valor real no es directamente proporcional al valor de la entrada analógica. El valor real acondicionado debe estar en formato de coma flotante.

Procedimiento

Para utilizar directamente el valor de la entrada analógica, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input_PER".
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica.

Para utilizar el valor real acondicionado en el formato de coma flotante, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input".
2. Seleccione "Instrucción" como fuente.
3. Indique el nombre de la variable en la que está almacenado el valor real acondicionado.

Valor de salida

PID_Compact ofrece tres valores de salida. El valor de salida que se utilice dependerá del actuador.

- Output_PER
  El actuador se activa a través de una salida analógica y se controla con una señal continua, como puede ser 0 a 10 V, 4 a 20 mA.
- Output
  El valor de salida debe acondicionarse mediante el programa de usuario, por ejemplo, porque el actuador muestra un comportamiento no lineal.
- Output_PWM
  El actuador se controla a través de una salida digital. Una modulación de ancho de impulsos permite formar tiempos de conexión y desconexión variables.

Procedimiento

Para utilizar el valor de salida analógico, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output_PER" (analógica).
2. Seleccione "Instrucción".
3. Introduzca la dirección de la salida analógica.

Para acondicionar el valor de salida mediante el programa de usuario, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output".
2. Seleccione "DB de instancia".
   El valor de salida calculado se guarda en DB de instancia.
3. Para acondicionar el valor de salida utilice el parámetro de salida Output.
4. Transfiera el valor de salida acondicionado hasta el actuador a través de una salida digital o analógica de la CPU.

Para utilizar el valor de salida digital, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output_PWM".
2. Seleccione "Instrucción".
3. Introduzca la dirección de la salida digital.

Ajustes del valor real

Escalar valor real

Si en los ajustes básicos ha configurado el uso de Input_PER, deberá convertir el valor de la entrada analógica a la magnitud física del valor real. En el cuadro de visualización Input_PER se muestra la configuración actual.

Si el valor real es directamente proporcional al valor de la entrada analógica, Input_PER se escalará a partir de un par de valores inferior y superior.

Procedimiento

Para escalar el valor real, haga lo siguiente:

1. En los campos de entrada "Valor real inferior escalado" y "abajo", introduzca el par de valores inferior.
2. En los campos de entrada "Valor real superior escalado" y "arriba", introduzca el par de valores superior.

La configuración hardware contiene pares de valores predeterminados. Para utilizar los pares de valores contenidos en la configuración hardware, haga lo siguiente:

1. Seleccione en el editor de programación la instrucción PID_Compact.
2. En los ajustes básicos, interconecte Input_PER con una entrada analógica.
3. En los ajustes del valor real, haga clic en el botón "Ajuste automático".

Los valores existentes se sobrescribirán con los valores de la configuración hardware.

Límites del valor real

Como valores límite para el sistema regulado defina los límites superior e inferior absolutos del valor real de modo que sean razonables. Tan pronto como dichos valores se rebasen por exceso o defecto, se producirá un error (ErrorBits = 0001h). La optimización se cancela cuando se rebasan los valores reales. Configure en los ajustes de los valores de salida cómo debe reaccionar PID_Compact en caso de error en el modo automático.

Ajustes avanzados

Monitorización del valor real

Configure un límite de advertencia inferior y uno superior para el valor real en la ventana de configuración "Monitorización del valor real". Si durante el funcionamiento se rebasa uno de los límites de advertencia por defecto o por exceso, se muestra una advertencia en la instrucción "PID_Compact".

- En el parámetro de salida "InputWarning_H", si se ha rebasado por exceso el límite superior de advertencia
- En el parámetro de salida "InputWarning_L", si se ha rebasado por defecto el límite inferior de advertencia

Los límites de advertencia deben encontrarse dentro de los límites superior e inferior del valor real.

Si no introduce ningún valor, se utilizan los límites superior e inferior del valor real.

Ejemplo

Límite superior del valor real = 98 °C; límite superior de advertencia = 90 °C

Límite inferior de advertencia = 10 °C; límite inferior del valor real = 0 °C

PID_Compact se comporta del modo siguiente:

Valor real	InputWarning_H	InputWarning_L	ErrorBits	Modo de operación
> 98 °C	TRUE	FALSE	0001h	Inactivo o valor de salida sustitutivo con monitorización de errores
≤ 98 °C y > 90 °C	TRUE	FALSE	0000h	Modo automático
≤ 90 °C y ≥ 10 °C	FALSE	FALSE	0000h	Modo automático
< 10°C y ≥ 0 °C	FALSE	TRUE	0000h	Modo automático
< 0 °C	FALSE	TRUE	0001h	Inactivo o valor de salida sustitutivo con monitorización de errores

Configure en los ajustes de los valores de salida cómo debe reaccionar PID_Compact si se rebasa el límite superior o inferior del valor real.

Consulte también

Parámetros State y Mode V2 (Página 5152)

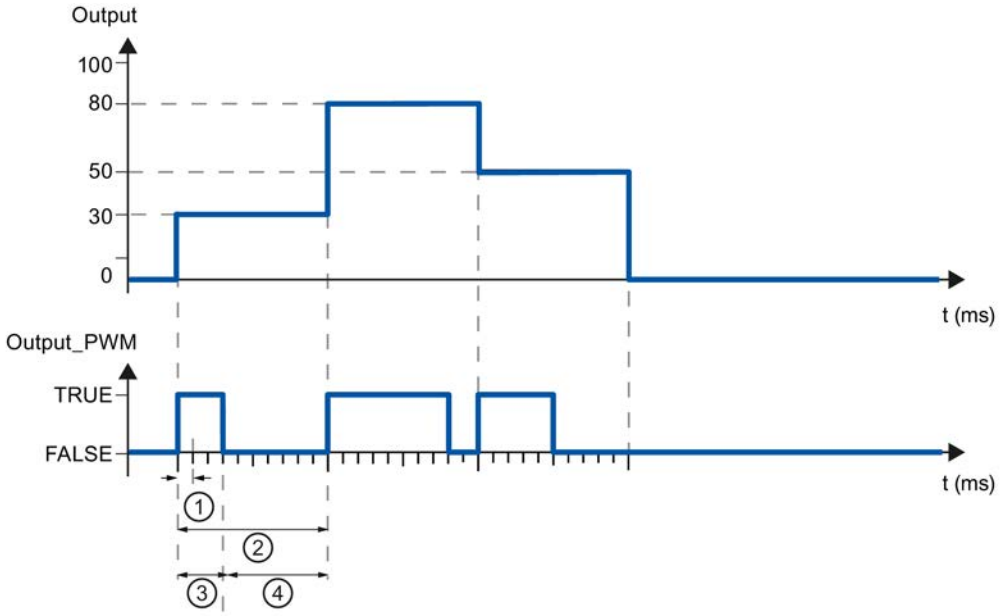
Limitaciones PWM

Una modulación de ancho de impulsos transforma el valor del parámetro de salida Output en una secuencia de impulsos que se emite en el parámetro de salida Output_PWM. Output se

calcula en el tiempo de muestreo del algoritmo PID, Output_PWM se emite en el tiempo de muestreo PID_Compact.

El tiempo de muestreo del algoritmo PID se determina durante la optimización inicial o la optimización fina. Al ajustar los parámetros PID manualmente, debe configurarse también ahí el tiempo de muestreo del algoritmo PID. El tiempo de muestreo PID_Compact equivale al tiempo de ciclo del OB invocante.

La duración del impulso es proporcional al valor de Output y es siempre un múltiplo entero del tiempo de muestreo PID_Compact.



① Tiempo de muestreo PID_Compact
② Tiempo de muestreo algoritmo PID
③ Duración del impulso
④ Duración de la pausa

"Tiempo conexión mín." y "Tiempo de desconex. mín." se redondean a un múltiplo entero del tiempo de muestreo PID_Compact.

Un impulso o una pausa nunca son más cortas que los tiempos de conexión o desconexión mínimos. Las imprecisiones que se producen se suman y se compensan en el siguiente ciclo.

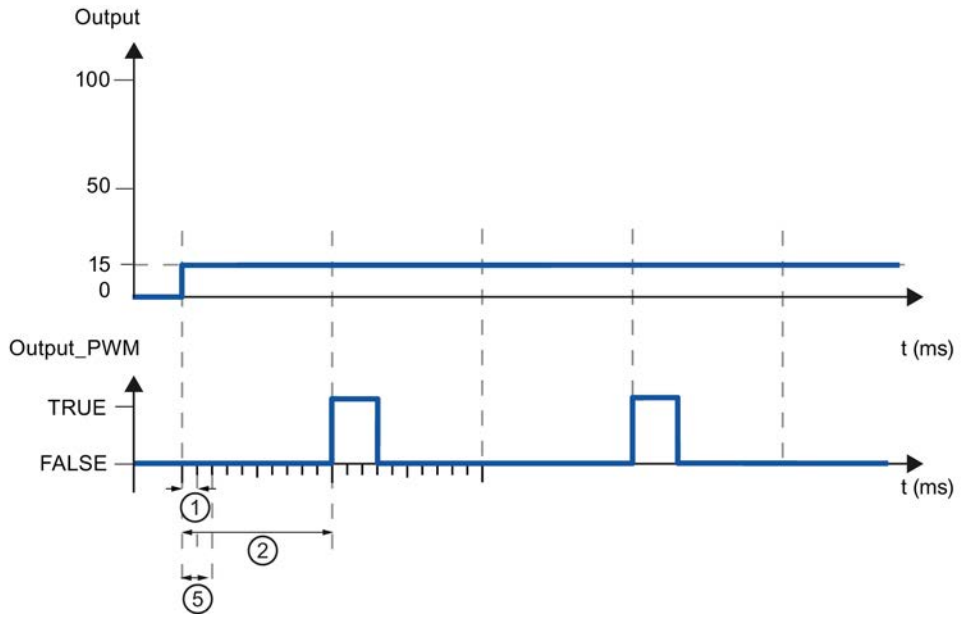
Ejemplo

Tiempo de muestreo PID_Compact = 100 ms

Tiempo de muestreo del algoritmo PID = 1000 ms

Tiempo conexión mín. = 200 ms

Output es constante del 15%. El mínimo impulso que PID_Compact puede emitir es del 20 % En el primer ciclo no se emite ningún impulso. En el segundo ciclo, el impulso no emitido del primer ciclo se agrega al impulso del segundo ciclo.



① Tiempo de muestreo PID_Compact

② Tiempo de muestreo algoritmo PID

⑤ Tiempo conexión mín.

Para reducir la frecuencia de conmutación y proteger el actuador, prolongue los tiempos de conexión y desconexión mínimos.

Si se utiliza "Output" o "Output_PER", los tiempos mínimos de conexión y desconexión deben configurarse con el valor 0.0.

Nota

Los tiempos mínimos de conexión y desconexión actúan solo en el parámetro de salida Output_PWM y no se utilizan para posibles generadores de impulsos integrados en la CPU.

Valor de salida

Límites del valor de salida

Configure en tantos por ciento los límites absolutos del valor de salida en la ventana de configuración "Límites del valor de salida". Los límites del valor de salida absolutos no se rebasan ni por exceso ni por defecto ni en el modo manual ni en el automático. Si en el modo manual se especifica un valor de salida fuera de los límites, el valor efectivo se limitará en la CPU a los límites configurados.

Los límites del valor de salida deben ser compatibles con el sentido de regulación.

Los valores válidos para los límites del valor de salida dependen del Output utilizado.

Output	de -100.0 a 100.0%
Output_PER	de -100.0 a 100.0%
Output_PWM	de -0.0 a 100.0%

Comportamiento en caso de error

ATENCIÓN
Su instalación puede sufrir daños. Si en caso de fallo se emite "Valor actual para la duración del error" o "Valor de salida sustitutivo mientras dure el error", PID_Compact se mantiene en modo automático. De este modo, los límites del valor real pueden rebasarse y su instalación puede sufrir daños. Configure un comportamiento en caso de error para su sistema regulado que proteja su instalación de daños.

PID_Compact está preajustado de manera que, en caso de error, la regulación permanece activa en la mayoría de los casos. Si en el modo de regulación se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación. Compruebe en tal caso el parámetro Errorbits y elimine la causa de error.

En caso de error, PID_Compact emite un valor de salida configurable:

- Cero (inactivo)
  PID_Compact emite 0.0 como valor de salida para todos los errores y conmuta al modo de operación "Inactivo". El regulador no se reactiva hasta que no se detecta un flanco descendente en Reset o un flanco ascendente en ModeActivate.
- Valor actual mientras dure el error
  Si se producen los siguientes errores en el **modo automático**, PID_Compact regresa a dicho modo en cuanto los errores en cuestión dejan de existir.
  Si se producen uno o varios de los errores siguientes, PID_Compact permanece en modo automático:
  - 0001h: El parámetro "Input" se encuentra fuera de los límites del valor real.
  - 0800h: Error en tiempo de muestreo
  - 40000h: Valor no válido en el parámetro Disturbance.

  Si en el **modo automático** se producen uno o varios de los errores siguientes, PID_Compact pasa al modo "Valor de salida sustitutivo con monitorización de errores" y emite el último valor de salida válido:
  - 0002h: Valor no válido en el parámetro Input_PER.
  - 0200h: Valor no válido en el parámetro Input.
  - 0400h: Error al calcular el valor de salida.
  - 1000h: Valor no válido en el parámetro Setpoint.

  Si se produce un error en **modo manual**, PID_Compact sigue utilizando como valor de salida el valor manual. Si el valor manual no es válido, se utiliza el valor de salida sustitutivo. Si el valor manual y el valor de salida sustitutivo no son válidos, se utiliza el límite inferior del valor de salida.
  Si durante una **optimización inicial u optimización fina** aparece el siguiente error, PID_Compact se mantiene en el modo de operación activo:
  - 0020h: La optimización inicial no está permitida durante la optimización fina.

  Con todos los demás errores, PID_Compact interrumpe la optimización y cambia al modo desde el que se inició la optimización.
  En cuanto no aparezcan errores, PID_Compact pasa de nuevo al modo automático.
- Valor de salida sustitutivo mientras dure el error
  PID_Compact emite el valor de salida sustitutivo.
  Si se produce el siguiente error, PID_Compact se mantiene en el modo "Valor de salida sustitutivo con monitorización de errores" y emite el límite inferior del valor de salida:
  - 20000h: Valor no válido en la variable SubstituteOutput.

  Con el resto de los errores, PID_Compact se comporta del modo descrito en "Valor actual mientras dure el error".

Consulte también

Parámetros State y Mode V2 (Página 5152)

Parámetros PID

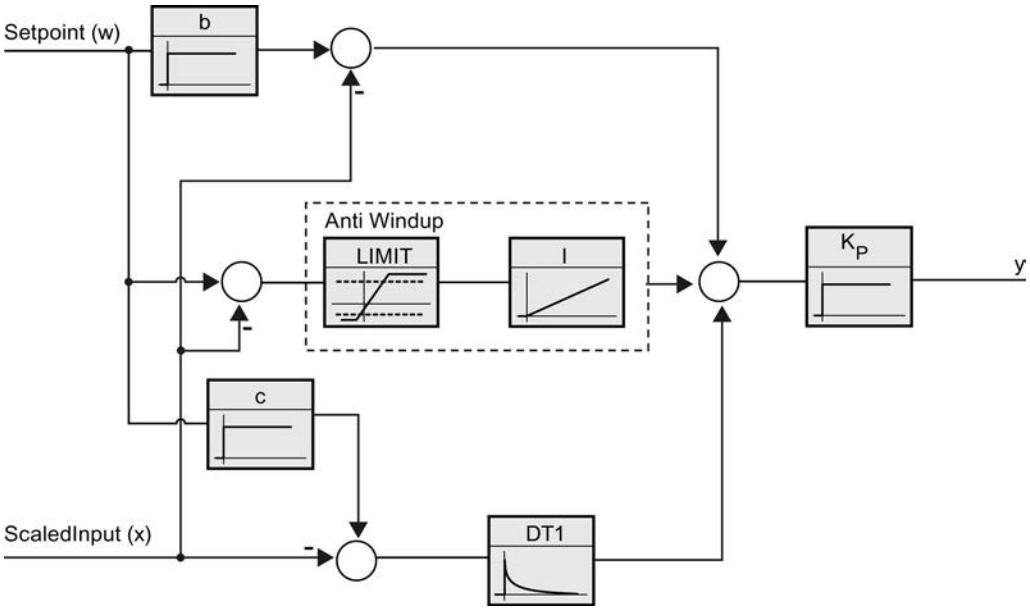
En la ventana de configuración "Parámetros PID" se visualizan los parámetros PID. Durante la optimización los parámetros PID se adaptan al sistema regulado. No es necesario introducir los parámetros PID de forma manual.

El algoritmo PID funciona de acuerdo con la siguiente fórmula:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
y	Valor de salida del algoritmo PID
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada T1 = a ×T_D)
T_D	Tiempo derivativo
c	Ponderación de la acción D

El gráfico siguiente muestra cómo entran los parámetros en el algoritmo PID.



Todos los parámetros PID son remanentes. Si introduce manualmente los parámetros PID, debe cargar PID_Compact por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Ganancia proporcional

El valor indica la ganancia proporcional del regulador. PID_Compact no funciona con una ganancia proporcional negativa. El sentido de regulación se invierte en Ajustes básicos > Tipo de regulación.

Tiempo de integración

El tiempo de integración determina el comportamiento temporal de la acción I. La desconexión de la acción I se realiza con el tiempo de integración = 0,0.

Tiempo derivativo

El tiempo de la acción derivada determina el comportamiento temporal de la acción D. La desconexión de la acción D se realiza con el tiempo derivativo = 0,0.

Coeficiente para el retardo de la acción derivada

El efecto de la acción D se retrasa mediante el coeficiente de retardo de la acción derivada.

Retardo de la acción derivada = Tiempo derivativo x coeficiente de retardo de la acción derivada

- 0.0: la acción D solo surte efecto para un ciclo y, por ello, casi no es efectiva.
- 0.5: Este valor se ha acreditado en la práctica para sistemas regulados con **una** constante de tiempo dominante.
- > 1.0: Cuanto mayor sea el coeficiente, más se retrasará el efecto de la acción D.

Ponderación de la acción P

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción P.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción P es totalmente efectiva cuando se cambia la consigna
- 0.0: La acción P no actúa al cambiar la consigna

Si se produce una modificación del valor real, la acción P es totalmente efectiva.

Ponderación de la acción D

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción D.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción D es totalmente efectiva cuando se cambia la consigna.
- 0.0: La acción D no actúa al cambiar la consigna

Si se produce una modificación del valor real, la acción D es totalmente efectiva.

Tiempo de muestreo algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de ciclo. Todas las demás funciones de PID_Compact se ejecutan con cada llamada.

Si utiliza Output_PWM, la precisión de la señal de salida se determina mediante el comportamiento del tiempo de muestreo del algoritmo PID al tiempo de ciclo del OB. El tiempo de muestreo del algoritmo PID equivale a la duración del período de la modulación del ancho de pulso. El tiempo de ciclo debería ser de al menos 10 veces el tiempo de muestreo del algoritmo PID.

Regla para la optimización

En la lista desplegable "Estructura del regulador" seleccione si van a calcular los parámetros PI o PID.

- PID
  Se calcula durante la optimización inicial y la optimización fina del parámetro PID.
- PI
  Se calcula durante la optimización inicial y la optimización fina del parámetro PI.
- Definido por el usuario
  Si se han ajustado diferentes estructuras del regulador para la optimización inicial y la optimización fina en un programa de usuario, en la lista desplegable aparece "Definido por el usuario".

Poner en servicio PID_Compact V2

Optimización inicial

La optimización inicial determina el comportamiento del proceso a un escalón del valor de salida y busca el punto de inflexión. Los parámetros PID óptimos se calculan a partir de la pendiente máxima y el tiempo muerto del sistema regulado. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina.

Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. Esto tiene más probabilidades de suceder en los modos "Inactivo" o "Manual". Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

Requisitos

- La instrucción "PID_Compact" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- Reset = FALSE

- PID_Compact se encuentra en el modo de operación "Modo manual", "Inactivo" o "Modo automático".
- La consigna y el real se encuentran dentro de los límites configurados (véase la configuración "Monitorización del valor real").
- La diferencia entre la consigna y el valor real es mayor del 30% de la diferencia entre el límite superior e inferior del valor real.
- La diferencia entre la consigna y el valor real es superior al 50% de la consigna.

Procedimiento

Para poder realizar una optimización inicial, proceda del siguiente modo:

1. Haga doble clic en la navegación del proyecto en la entrada "PID_Compact" > Puesta en servicio".
2. En la lista desplegable "Modo de optimización" seleccione la entrada "Optimización inicial".
3. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia la optimización inicial.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los fallos ocurridos. La barra de progreso muestra el progreso del paso actual.

   Nota

   Haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si la optimización inicial ha finalizado sin ningún mensaje de error, significa que los parámetros PID se han optimizado. PID_Compact cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si no es posible realizar la optimización inicial, PID_Compact se comporta del modo configurado en Comportamiento en caso de error.

Consulte también

Parámetros State y Mode V2 (Página 5152)

Optimización fina

La optimización fina genera una oscilación constante y limitada del valor real. Los parámetros PID se optimizan para el punto de operación a partir de la amplitud y la frecuencia. A partir de los resultados se vuelven a calcular todos los parámetros PID. Los parámetros PID

existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina.

PID_Compact intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan solo una mínima influencia sobre la optimización fina. Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

Requisitos

- La instrucción "PID_Compact" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- Reset = FALSE
- La consigna y el valor real se encuentran dentro de los límites configurados.
- El lazo de regulación es estacionario en el punto de operación. El punto de operación se ha alcanzado cuando el valor real coincide con la consigna.
- No se esperan perturbaciones.
- PID_Compact se encuentra en el modo de operación inactivo, automático o manual.

El proceso depende de la situación de inicio

La optimización fina puede iniciarse desde los modos de operación "Inactivo", "Automático" o "Manual". Al inicio, la optimización fina se desarrolla del modo siguiente:

- Modo automático
  Si desea mejorar los parámetros PID existentes mediante optimización, inicie la optimización fina desde el modo automático.
  PID_Compact regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se cumplen las condiciones para la optimización fina. Sólo entonces comienza la optimización fina.
- Inactivo o manual
  Si se cumplen las condiciones para optimización inicial, ésta se inicia. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina. Si no es posible realizar la optimización inicial, PID_Compact se comporta del modo configurado en Comportamiento en caso de error.
  Si el valor real para una optimización inicial se encuentra ya muy cerca de la consigna, se intenta alcanzar la consigna con el valor de salida máximo o mínimo, lo que puede ocasionar una sobreoscilación muy alta.

Procedimiento

Para poder realizar una optimización fina, proceda del siguiente modo:

1. En la lista desplegable "Modo de optimización" seleccione la entrada "Optimización fina".
2. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia el proceso de optimización fina.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los fallos ocurridos. La barra de progreso muestra el progreso del paso actual.

Nota

En el grupo "Modo de optimización" haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si durante la optimización fina no se producen errores, significa que los parámetros PID se han optimizado. PID_Compact cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si se han producido errores durante la "optimización fina", PID_Compact se comporta del modo configurado en Comportamiento en caso de error.

Consulte también

Parámetros State y Mode V2 (Página 5152)

Modo de operación "Modo manual"

A continuación se describe cómo utilizar el modo de operación "Modo manual" en la ventana de puesta en servicio del objeto tecnológico "PID_Compact". El modo manual es posible aunque haya un error pendiente.

Requisitos

- La instrucción "PID_Compact" se llama en un OB de alarma cíclica.
- Hay establecida una conexión online con la CPU y ésta se encuentra en el estado operativo "RUN".

Procedimiento

Utilice "Modo manual" en la ventana de puesta en servicio si desea probar el sistema regulado especificando un valor manual. Para especificar un valor manual, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
2. Active la casilla de verificación "Modo manual" en el área "Estado online del regulador". PID_Compact funciona en modo manual. El último valor de salida actual permanece activo.
3. Introduzca el valor manual en la unidad % en el campo "Output".
4. Haga clic en el símbolo .

Resultado

El valor manual se escribe en la CPU y es efectivo de forma inmediata.

Desactive la casilla de verificación "Modo manual" si desea que el regulador PID vuelva a especificar el valor de salida. El cambio al modo automático se efectúa sin discontinuidad.

Consulte también

Parámetros State y Mode V2 (Página 5152)

PID_Compact V1

Configurar PID_Compact V1

Ajustes básicos

Introducción

Configure las propiedades siguientes del objeto tecnológico PID_Compact en el área de ajustes básicos de la ventana de inspección o de configuración.

- Magnitud física
- Sentido de regulación
- Comportamiento en arranque después de un reset
- Consigna (sólo en la ventana de inspección)
- Valor real (sólo en la ventana de inspección)
- Valor de salida (sólo en la ventana de inspección)

Consigna, valor real y valor de salida

La consigna, el valor real y el valor de salida sólo pueden configurarse en la ventana de inspección del editor de programación. Para cada salida digital elija la fuente:

- DB de instancia
  Se utiliza el valor que se encuentra almacenado en el DB de instancia.
  El valor debe actualizarse en el DB de instancia del programa de usuario.
  En la instrucción no puede haber ningún valor.
  Modificación posible desde HMI.
- Instrucción
  Se utiliza el valor que se conecta con la instrucción.
  Cada vez que se llama la instrucción, el valor se escribe en el DB de instancia.
  No se puede modificar desde HMI.

Tipo de regulación

Magnitud física

En el grupo "Tipo de regulación" seleccione la magnitud física y la unidad de la consigna y el valor real. La consigna y el valor real se muestran en esta unidad.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal.

PID_Compact no funciona con ganancia proporcional negativa. Para reducir el valor real con un valor de salida más elevado, active la casilla de verificación "Inversión sentido de regulación".

Ejemplos

- Al abrir una válvula de escape se reduce el nivel de un recipiente.
- Si se aumenta la potencia de refrigeración, disminuye la temperatura.

Comportamiento en arranque tras un reset

Para cambiar inmediatamente al último modo de operación activo tras el rearranque de la CPU, active la casilla de verificación "Activar último modo de operación después del rearranque de la CPU".

Si la casilla de verificación está desactivada, PID_Compact se queda en el modo de operación "Inactivo".

Consigna

Procedimiento

Para especificar una consigna fija, proceda del siguiente modo:

1. Seleccione "DB de instancia".
2. Introduzca una consigna, como puede ser 80 °C.
3. En caso necesario, elimine una entrada de la instrucción.

Para especificar una consigna variable, proceda del siguiente modo:

1. Seleccione "Instrucción".
2. Indique el nombre de la variable REAL en la que está almacenada la consigna.
   Es posible asignar varios valores controlados por programa a la variable REAL para, por ejemplo, modificar la consigna de tiempo de forma controlada.

Valor real

Si utiliza directamente el valor de la entrada analógica, PID_Compact escala el valor de la entrada analógica al tamaño físico.

Si se desea acondicionar el valor de la entrada analógica, es preciso escribir un programa propio para su acondicionamiento. Por ejemplo, el valor real no es directamente proporcional al valor de la entrada analógica. El valor real acondicionado debe estar en formato de coma flotante.

Procedimiento

Para utilizar directamente el valor de la entrada analógica, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input_PER".
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica.

Para utilizar el valor real acondicionado en el formato de coma flotante, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input".
2. Seleccione "Instrucción" como fuente.
3. Indique el nombre de la variable en la que está almacenado el valor real acondicionado.

Valor de salida

PID_Compact ofrece tres valores de salida. El valor de salida que se utilice dependerá del actuador.

- Output_PER
  El actuador se activa a través de una salida analógica y se controla con una señal continua, como puede ser 0 a 10 V, 4 a 20 mA.
- Output
  El valor de salida debe acondicionarse mediante el programa de usuario, por ejemplo, porque el actuador muestra un comportamiento no lineal.
- Output_PWM
  El actuador se controla a través de una salida digital. Una modulación de ancho de impulsos permite formar tiempos de conexión y desconexión variables.

Procedimiento

Para utilizar el valor de salida analógico, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output_PER" (analógica).
2. Seleccione "Instrucción".
3. Introduzca la dirección de la salida analógica.

Para acondicionar el valor de salida mediante el programa de usuario, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output".
2. Seleccione "DB de instancia".
   El valor de salida calculado se guarda en DB de instancia.
3. Para acondicionar el valor de salida utilice el parámetro de salida Output.
4. Transfiera el valor de salida acondicionado hasta el actuador a través de una salida digital o analógica de la CPU.

Para utilizar el valor de salida digital, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output_PWM".
2. Seleccione "Instrucción".
3. Introduzca la dirección de la salida digital.

Ajuste del valor real

Configure la normalización del valor real en la ventana de configuración "Ajustes del valor real" y fije los límites absolutos del valor real.

Escalar valor real

Si en los ajustes básicos se ha elegido el uso de Input_PER, el valor de la entrada analógica debe convertirse en el tamaño físico del valor real. En el cuadro de visualización Input_PER se muestra la configuración actual.

Si el valor real es directamente proporcional al valor de la entrada analógica, Input_PER se escala a partir de una pareja de valores inferiores y superiores.

1. En los campos de entrada "Valor real inferior escalado" y "Abajo" introduzca la pareja de valores inferiores.
2. En los campos de entrada "Valor real inferior escalado" y "Abajo" introduzca la pareja de valores inferiores.

La configuración de hardware incluye una configuración predeterminada para los pares de valores. Para utilizar los pares de valores a partir de la configuración de hardware, proceda del siguiente modo:

1. Seleccione la instrucción PID_Compact en el editor de programación.
2. En los ajustes básicos, interconecte Input_PER con una entrada analógica.
3. En la configuración del valor real, haga clic en el botón "Ajuste automático".

Los valores existentes se sobrescriben con los valores de la configuración de hardware.

Monitorizar el valor real

Defina los límites absolutos superior e inferior del valor real. En cuanto dichos valores se rebasen por exceso o por defecto durante el funcionamiento, la regulación se desconectará y el valor de salida se ajustará al 0%. Como límites, se deben introducir valores razonables para el sistema regulado. Durante la optimización es importante disponer de valores límite razonables para obtener los parámetros PID óptimos.

La preasignación del "Límite superior del valor real" es 120%. En la entrada de periferia, el valor real puede encontrarse como máximo un 18% por encima del rango normalizado (margen de saturación). No se comunica ningún error más debido al rebasamiento por exceso del "Límite superior del valor real". Únicamente se detectan la rotura de hilo y el cortocircuito y PID_Compact pasa al modo de operación "Inactivo".

⚠ ADVERTENCIA
Si ajusta valores muy altos (z. B. $-3{,}4*10^{38}$...$+3{,}4*10^{38}$) como límites del valor real, se desactiva la monitorización del valor real. Esto puede dar lugar a daños en la instalación.

Consulte también

Monitorización del valor real (Página 9986)

Limitaciones PWM (Página 9986)

Límites del valor de salida (Página 9988)

Parámetros PID (Página 9989)

Ajustes avanzados

Monitorización del valor real

Configure un límite de advertencia inferior y uno superior para el valor real en la ventana de configuración "Monitorización del valor real". Si durante el funcionamiento se rebasa uno de los límites de advertencia por defecto o por exceso, se muestra una advertencia en la instrucción "PID_Compact".

- En el parámetro de salida "InputWarning_H", si se ha rebasado por exceso el límite superior de advertencia
- En el parámetro de salida "InputWarning_L", si se ha rebasado por defecto el límite inferior de advertencia

Los límites de advertencia deben encontrarse dentro de los límites superior e inferior del valor real.

Si no introduce ningún valor, se utilizan los límites superior e inferior del valor real.

Ejemplo

Límite superior del valor real = 98 °C; límite superior de advertencia = 90 °C

Límite inferior de advertencia = 10 °C; límite inferior del valor real = 0 °C

PID_Compact se comporta del modo siguiente:

Valor real	InputWarning_H	InputWarning_L	Modo de operación
> 98 °C	TRUE	FALSE	Inactivo
≤ 98 °C y > 90 °C	TRUE	FALSE	Modo automático
≤ 90 °C y ≥ 10 °C	FALSE	FALSE	Modo automático
< 10°C y ≥ 0 °C	FALSE	TRUE	Modo automático
< 0 °C	FALSE	TRUE	Inactivo

Consulte también

Ajuste del valor real (Página 9981)

Limitaciones PWM (Página 9986)

Límites del valor de salida (Página 9988)

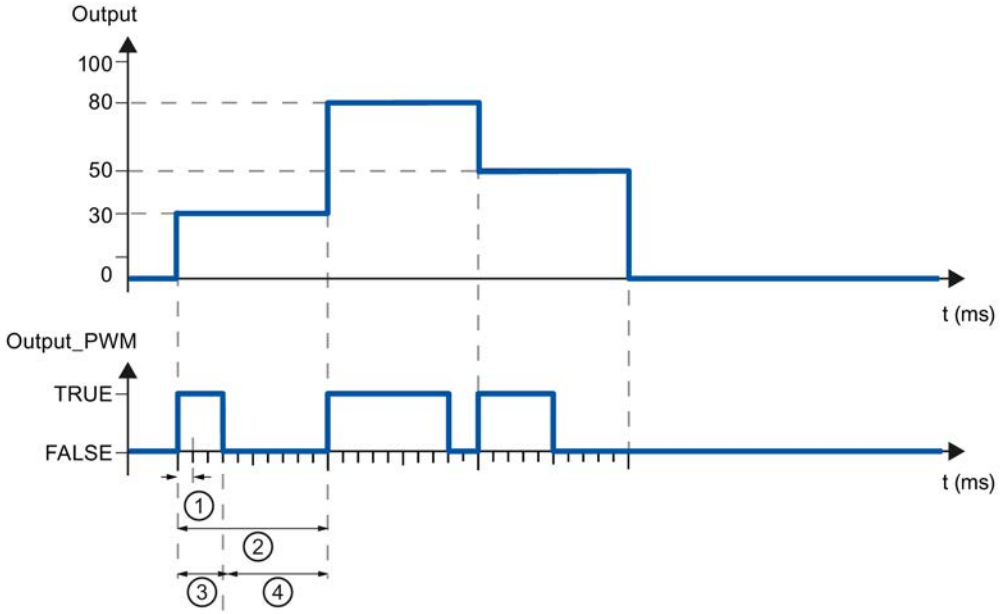
Parámetros PID (Página 9989)

Limitaciones PWM

Una modulación de ancho de impulsos transforma el valor del parámetro de salida Output en una secuencia de impulsos que se emite en el parámetro de salida Output_PWM. Output se calcula en el tiempo de muestreo del algoritmo PID, Output_PWM se emite en el tiempo de muestreo PID_Compact.

El tiempo de muestreo del algoritmo PID se determina durante la optimización inicial o la optimización fina. Al ajustar los parámetros PID manualmente, debe configurarse también ahí el tiempo de muestreo del algoritmo PID. El tiempo de muestreo PID_Compact equivale al tiempo de ciclo del OB invocante.

La duración del impulso es proporcional al valor de Output y es siempre un múltiplo entero del tiempo de muestreo PID_Compact.



① Tiempo de muestreo PID_Compact
② Tiempo de muestreo algoritmo PID
③ Duración del impulso
④ Duración de la pausa

"Tiempo conexión mín." y "Tiempo de desconex. mín." se redondean a un múltiplo entero del tiempo de muestreo PID_Compact.

Un impulso o una pausa nunca son más cortas que los tiempos de conexión o desconexión mínimos. Las imprecisiones que se producen se suman y se compensan en el siguiente ciclo.

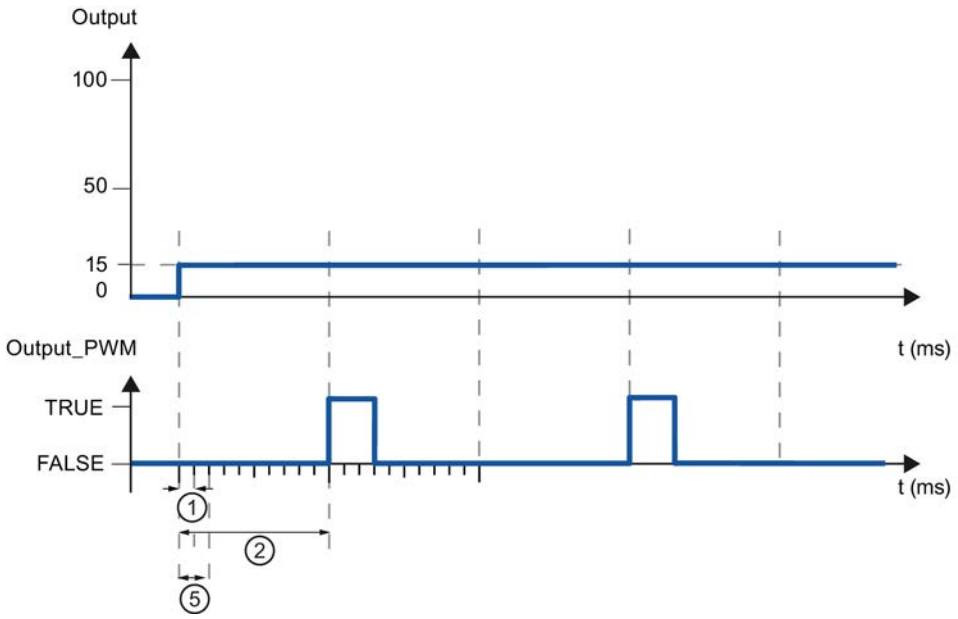
Ejemplo

Tiempo de muestreo PID_Compact = 100 ms

Tiempo de muestreo del algoritmo PID = 1000 ms

Tiempo conexión mín. = 200 ms

Output es constante del 15%. El mínimo impulso que PID_Compact puede emitir es del 20 % En el primer ciclo no se emite ningún impulso. En el segundo ciclo, el impulso no emitido del primer ciclo se agrega al impulso del segundo ciclo.



① Tiempo de muestreo PID_Compact

② Tiempo de muestreo algoritmo PID

⑤ Tiempo conexión mín.

Para reducir la frecuencia de conmutación y proteger el actuador, prolongue los tiempos de conexión y desconexión mínimos.

Si se utiliza "Output" o "Output_PER", los tiempos mínimos de conexión y desconexión deben configurarse con el valor 0.0.

Nota

Los tiempos mínimos de conexión y desconexión actúan solo en el parámetro de salida Output_PWM y no se utilizan para posibles generadores de impulsos integrados en la CPU.

Consulte también

Ajuste del valor real (Página 9981)

Monitorización del valor real (Página 9983)

Límites del valor de salida (Página 9988)

Parámetros PID (Página 9989)

Límites del valor de salida

Configure en tantos por ciento los límites absolutos del valor de salida en la ventana de configuración "Límites del valor de salida". Los límites absolutos del valor de salida no se rebasan ni por exceso ni por defecto ni en el modo manual ni en el automático. Si en el modo manual se especifica un valor de salida fuera de los límites, el valor efectivo se limitará en la CPU a los límites configurados.

Los valores válidos para los límites del valor de salida dependen del Output utilizado.

Output	-100,0 a 100,0
Output_PER	-100,0 a 100,0
Output_PWM	0,0 a 100,0

En caso de error, PID_Compact ajusta el valor de salida a 0.0. Así pues, 0.0 debe encontrarse siempre dentro de los límites del valor de salida. Para obtener un límite inferior del valor de salida mayor que 0.0, en el programa de usuario debe sumarse un offset a Output y Output_PER.

Consulte también

Ajuste del valor real (Página 9981)

Monitorización del valor real (Página 9983)

Limitaciones PWM (Página 9983)

Parámetros PID (Página 9989)

Parámetros PID

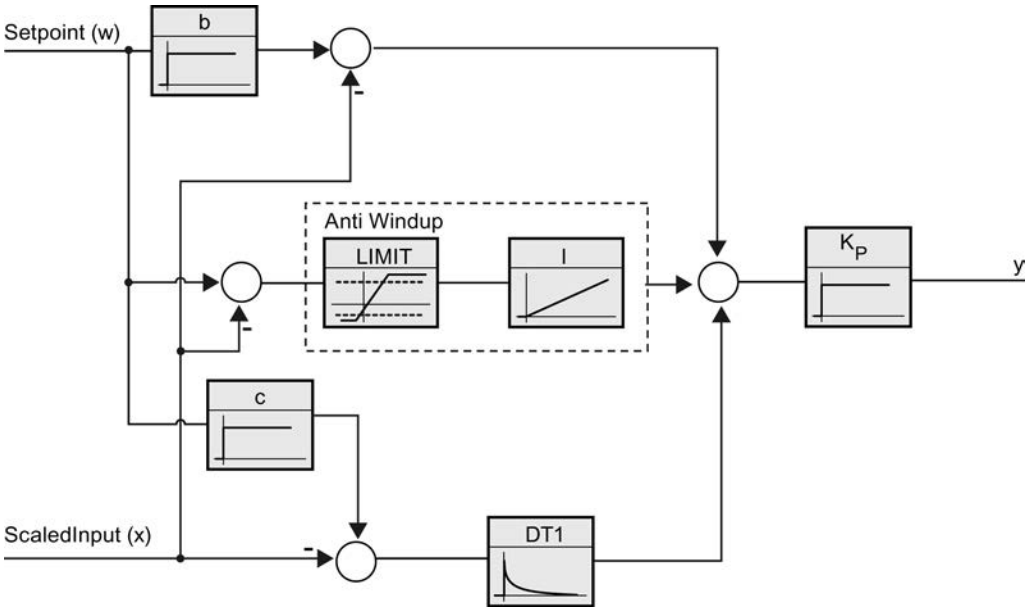
En la ventana de configuración "Parámetros PID" se visualizan los parámetros PID. Durante la optimización los parámetros PID se adaptan al sistema regulado. No es necesario introducir los parámetros PID de forma manual.

El algoritmo PID funciona de acuerdo con la siguiente fórmula:

$$y = K_p \left[(b \cdot w - x) + \frac{1}{T_I \cdot s} (w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1} (c \cdot w - x) \right]$$

Símbolo	Descripción
y	Valor de salida del algoritmo PID
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada T1 = a ×T_D)
T_D	Tiempo derivativo
c	Ponderación de la acción D

El gráfico siguiente muestra cómo entran los parámetros en el algoritmo PID.



Todos los parámetros PID son remanentes. Si introduce manualmente los parámetros PID, debe cargar PID_Compact por completo.

Auto-Hotspot

Ganancia proporcional

El valor indica la ganancia proporcional del regulador. PID_Compact no funciona con una ganancia proporcional negativa. El sentido de regulación se invierte en Ajustes básicos > Tipo de regulación.

Tiempo de integración

El tiempo de integración determina el comportamiento temporal de la acción I. La desconexión de la acción I se realiza con el tiempo de integración = 0,0.

Tiempo derivativo

El tiempo de la acción derivada determina el comportamiento temporal de la acción D. La desconexión de la acción D se realiza con el tiempo derivativo = 0,0.

Coeficiente para el retardo de la acción derivada

El efecto de la acción D se retrasa mediante el coeficiente de retardo de la acción derivada.

Retardo de la acción derivada = Tiempo derivativo x coeficiente de retardo de la acción derivada

- 0.0: la acción D solo surte efecto para un ciclo y, por ello, casi no es efectiva.
- 0.5: Este valor se ha acreditado en la práctica para sistemas regulados con una constante de tiempo dominante.
- > 1.0: Cuanto mayor sea el coeficiente, más se retrasará el efecto de la acción D.

Ponderación de la acción P

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción P.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción P es totalmente efectiva cuando se cambia la consigna
- 0.0: La acción P no actúa al cambiar la consigna

Si se produce una modificación del valor real, la acción P es totalmente efectiva.

Ponderación de la acción D

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción D.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción D es totalmente efectiva cuando se cambia la consigna.
- 0.0: La acción D no actúa al cambiar la consigna

Si se produce una modificación del valor real, la acción D es totalmente efectiva.

Tiempo de muestreo algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de ciclo. Todas las demás funciones de PID_Compact se ejecutan con cada llamada.

Si utiliza Output_PWM, la precisión de la señal de salida se determina mediante el comportamiento del tiempo de muestreo del algoritmo PID al tiempo de ciclo del OB. El tiempo de muestreo del algoritmo PID equivale a la duración del período de la modulación del ancho de pulso. El tiempo de ciclo debería ser de al menos 10 veces el tiempo de muestreo del algoritmo PID.

Regla para la optimización

En la lista desplegable "Estructura del regulador" seleccione si van a calcular los parámetros PI o PID.

- PID
  Se calcula durante la optimización inicial y la optimización fina del parámetro PID.
- PI
  Se calcula durante la optimización inicial y la optimización fina del parámetro PI.
- Definido por el usuario
  Si se han ajustado diferentes estructuras del regulador para la optimización inicial y la optimización fina en un programa de usuario, en la lista desplegable aparece "Definido por el usuario".

Consulte también

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Poner en servicio PID_Compact V1

Puesta en servicio

La ventana de puesta en servicio ayuda durante la puesta en servicio del regulador PID. En el visor de curvas se puede observar los valores de la consigna, valor real y valor de salida a lo largo del eje de tiempo. En la ventana de puesta en servicio se soportan las siguientes funciones:

- Optimización inicial del regulador
- Optimización fina del regulador
  Utilice la optimización fina si desea realizar un ajuste de precisión del parámetro PID.
- Observación de la regulación en curso en la ventana de curvas
- Probar el sistema regulado especificando un valor de salida manual

Para todas las funciones es preciso realizar una conexión online a la CPU.

Principios básicos del manejo

- Seleccione el tiempo de actualización que desee en la lista desplegable "Tiempo de actualización".
  Todos los valores de la ventana de puesta en servicio se actualizan en el tiempo de actualización seleccionado.
- Haga clic en el botón "Inicio" en el grupo Medición si desea utilizar las funciones de puesta en servicio.
  Se inicia el registro de los valores. En el visor de curvas se registran los valores actuales de consigna, valor real y valor de salida. Se habilita el manejo de la ventana de puesta en servicio.
- Haga clic en el botón "Parar" si desea finalizar las funciones de puesta en servicio.
  Los valores mostrados en el visor de curvas pueden continuar analizándose.

Al cerrar la ventana de puesta en servicio finaliza el registro en el visor de curvas y los valores registrados se borran.

Consulte también

Optimización inicial (Página 9993)

Optimización fina (Página 9994)

Modo de operación "Modo manual" (Página 9996)

Optimización inicial

La optimización inicial determina el comportamiento del proceso a un escalón del valor de salida y busca el punto de inflexión. A partir de la inclinación máxima y del tiempo muerto del sistema regulado se calculan los parámetros PID óptimos.

Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

Requisitos

- La instrucción "PID_Compact" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- PID_Compact se encuentra en el modo de operación "Inactivo" o "Manual".
- La consigna no puede modificarse durante la optimización. De lo contrario se desactiva PID_Compact.
- La consigna y el real se encuentran dentro de los límites configurados (véase la configuración "Monitorización del valor real").
- La diferencia entre la consigna y el valor real es mayor del 30% de la diferencia entre el límite superior e inferior del valor real.
- La diferencia entre la consigna y el valor real es superior al 50% de la consigna.

Procedimiento

Para poder realizar una optimización inicial, proceda del siguiente modo:

1. Haga doble clic en la navegación del proyecto en la entrada "PID_Compact" > Puesta en servicio".
2. En la lista desplegable "Modo de optimización" seleccione la entrada "Optimización inicial".
3. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia la optimización inicial.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los fallos ocurridos. La barra de progreso muestra el progreso del paso actual.

Nota

Haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si la optimización inicial ha finalizado sin ningún mensaje de error, significa que los parámetros PID se han optimizado. PID_Compact cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si no es preciso realizar la optimización inicial, PID_Compact cambia al modo de operación "Inactivo".

Consulte también

Parámetros State y sRet.i_Mode V1 (Página 5171)

Puesta en servicio (Página 9989)

Optimización fina (Página 9994)

Modo de operación "Modo manual" (Página 9996)

Optimización fina

La optimización fina genera una oscilación constante y limitada del valor real. Los parámetros PID se optimizan para el punto de operación a partir de la amplitud y la frecuencia. A partir de los resultados se vuelven a calcular todos los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial.

PID_Compact intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan sólo una mínima influencia sobre la

optimización fina. Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

Requisitos

- La instrucción "PID_Compact" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- La consigna y el real se encuentran dentro de los límites configurados (véase la configuración "Monitorización del valor real").
- El lazo de regulación es estacionario en el punto de operación. El punto de operación se ha alcanzado cuando el valor real coincide con la consigna.
- No se esperan perturbaciones.
- La consigna no puede modificarse durante la optimización.
- PID_Compact se encuentra en el modo de operación inactivo, automático o manual.

El proceso depende de la situación de inicio

La optimización fina puede iniciarse desde los modos de operación "Inactivo", "Automático" o "Manual". La optimización fina se realiza en el momento del inicio tal como se indica a continuación:

- Modo automático
  Si desea mejorar los parámetros PID existentes mediante optimización, inicie la optimización fina a partir del modo automático.
  PID_Comact regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina.
- Inactivo o manual
  Si se cumplen las condiciones para optimización inicial, ésta se inicia. Con los parámetros PID calculados la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina. Si no es posible realizar la optimización inicial, PID_Compact cambia al modo de operación "Inactivo".
  Si el valor real para una optimización inicial se encuentra ya muy cerca de la consigna, se intenta alcanzar la consigna con el valor de salida máximo o mínimo. lo que puede ocasionar una sobreoscilación muy alta.

Procedimiento

Para poder realizar una optimización fina, proceda del siguiente modo:

1. En la lista desplegable "Modo de optimización" seleccione la entrada "Optimización fina".
2. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia el proceso de optimización fina.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los fallos ocurridos. La barra de progreso muestra el progreso del paso actual.

Nota

En el grupo "Modo de optimización" haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si la optimización fina se ha completado sin ningún aviso de error, significa que los parámetros PID se han optimizado. PID_Compact cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si se produjeran errores durante la optimización fina, PID_Compact cambia al modo de operación "Inactivo".

Consulte también

Parámetros State y sRet.i_Mode V1 (Página 5171)

Puesta en servicio (Página 9989)

Optimización inicial (Página 9990)

Modo de operación "Modo manual" (Página 9996)

Modo de operación "Modo manual"

A continuación se describe cómo utilizar el modo de operación "Modo manual" en la ventana de puesta en servicio del objeto tecnológico "PID Compact".

Requisitos

- La instrucción "PID_Compact" se llama en un OB de alarma cíclica.
- Hay establecida una conexión online con la CPU y ésta se encuentra en el estado operativo "RUN".
- Las funciones de la ventana de puesta en marcha se han habilitado con el botón "Medición on".

Procedimiento

Utilice "Modo manual" en la ventana de puesta en servicio si desea probar el sistema regulado especificando un valor manual. Para especificar un valor manual, proceda del siguiente modo:

1. Active la casilla de verificación "Modo manual" en el área "Estado online del regulador". PID_Compact funciona en modo manual. El último valor de salida actual permanece activo.
2. Introduzca el valor manual en la unidad % en el campo "Output".
3. Haga clic en el botón .

Resultado

El valor manual se escribe en la CPU y es efectivo de forma inmediata.

Nota

PID_Compact sigue monitorizando el valor real. Cuando se exceden los límites del valor real, se desactiva PID_Compact.

Desactive la casilla de verificación "Modo manual" si desea que el regulador PID vuelva a especificar el valor de salida. El cambio al modo automático se efectúa sin discontinuidad.

Consulte también

Parámetros State y sRet.i_Mode V1 (Página 5171)

Puesta en servicio (Página 9989)

Optimización inicial (Página 9990)

Optimización fina (Página 9991)

11.3.3.2 Utilizar PID_3Step

Objeto tecnológico PID_3Step

El objeto tecnológico PID_3Step proporciona un regulador PID con optimización para válvulas o actuadores con comportamiento integrador.

Es posible configurar los siguientes reguladores:

- Regulador paso a paso de tres puntos con realimentación de posición
- Regulador paso a paso de tres puntos sin realimentación de posición
- Regulador de válvula con valor de salida analógico

PID_3Step registra de forma continua el valor real medido dentro de un lazo de regulación y lo compara con la consigna. A partir del error de regulación resultante, PID_3Step calcula un valor de salida, con el que el valor real se iguala con la consigna con la máxima rapidez y estabilidad. En los reguladores PID, el valor de salida se compone de tres acciones:

- Acción **P**
  La acción P del valor de salida aumenta proporcionalmente al error de regulación.
- Acción **I**
  La acción I del valor de salida aumenta hasta que se compensa el error de regulación.
- Acción **D**
  La acción D aumenta con una velocidad de variación creciente del error de regulación. El valor real se igual lo más rápidamente posible con la consigna. Si la velocidad de variación del error de regulación vuelve a reducirse, también lo hace la acción D.

La instrucción PID_3Step calcula los parámetros P, I y D para su sistema regulado de forma autónoma durante la optimización inicial. Los parámetros pueden optimizarse aún más a través de una optimización fina. No es necesario determinar los parámetros manualmente.

Información adicional

- Vista general de los reguladores de software (Página 9932)
- Agregar objetos tecnológicos (Página 9934)
- Configurar objetos tecnológicos (Página 9935)
- Configuración de PID_3Step V2 (Página 9998)
- Configuración de PID_3Step V1 (Página 10016)

PID_3Step V2

Configuración de PID_3Step V2

Ajustes básicos

Introducción

Configure las propiedades siguientes del objeto tecnológico PID_3Step en el área de ajustes básicos de la ventana de inspección o de configuración.

- Magnitud física
- Sentido de regulación
- Comportamiento en arranque tras un reset
- Consigna (sólo en la ventana de inspección)

- Valor real (sólo en la ventana de inspección)
- Valor de salida (sólo en la ventana de inspección)
- Realimentación de posición (sólo en la ventana de inspección)

Consigna, valor real, valor de salida y realimentación de posición

La consigna, el valor real, el valor de salida y la realimentación de posición sólo pueden configurarse en la ventana de inspección del editor de programación. Para cada valor elija el origen:

- DB de instancia
  Se utiliza el valor que se encuentra almacenado en el DB de instancia.
  El valor debe actualizarse en el DB de instancia del programa de usuario.
  En la instrucción no puede haber ningún valor.
  Modificación posible desde HMI.
- Instrucción
  Se utiliza el valor que se interconecta a la instrucción.
  Cada vez que se llama la instrucción, el valor se escribe en el DB de instancia.
  No se puede modificar desde HMI.

Tipo de regulación

Magnitud física

En el grupo "Tipo de regulación", seleccione la magnitud física y la unidad para la consigna, el valor real y la magnitud perturbadora. La consigna, el valor real y la magnitud perturbadora se mostrarán en dicha unidad.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal.

PID_3Step no funciona con ganancia proporcional negativa. Para reducir el valor real con un valor de salida más elevado, active la casilla de verificación "Inversión sentido de regulación".

Ejemplos

- Al abrir una válvula de escape se reduce el nivel de llenado de un recipiente.
- Si se aumenta la potencia de refrigeración, disminuye la temperatura.

Comportamiento en arranque

1. Para cambiar inmediatamente al modo de operación "Inactivo" tras el rearranque de la CPU, desactive la casilla de verificación "Activar Mode tras rearrancar la CPU".
   Para cambiar inmediatamente, tras el rearranque de la CPU, al modo de operación almacenado en Mode, active la casilla de verificación "Activar Mode tras rearrancar la CPU".
2. En la lista desplegable "Poner Mode a", seleccione el modo de operación que debe activarse tras una carga completa en el dispositivo.
   Tras una carga completa en el dispositivo, PID_3Step arranca en el modo de operación seleccionado. Con cada rearranque, PID_3Step arranca en el último modo de operación almacenado en Mode.

Ejemplo

Se ha activado la casilla de verificación "Activar Mode tras rearrancar la CPU" y se ha seleccionado la entrada "Optimización inicial" en la lista "Poner Mode a". Tras una carga completa en el dispositivo, PID_3Step arranca en el modo de operación "Optimización inicial". Si la optimización inicial aún está activa y se rearranca la CPU, PID_3Step arranca de nuevo en el modo de operación "Optimización inicial". Si la optimización inicial ha finalizado correctamente, el modo automático aún está activo y se rearranca la CPU, PID_3Step arranca en "Modo automático".

Consigna

Procedimiento

Para especificar una consigna fija, proceda del siguiente modo:

1. Seleccione "DB de instancia".
2. Introduzca una consigna, como puede ser 80 °C.
3. En caso necesario, elimine una entrada de la instrucción.

Para especificar una consigna variable, proceda del siguiente modo:

1. Seleccione "Instrucción".
2. Indique el nombre de la variable REAL en la que está almacenada la consigna.
   Es posible asignar varios valores de forma controlada por programa a la variable REAL para, por ejemplo, modificar la consigna de tiempo de forma controlada.

Valor real

Si utiliza directamente el valor de la entrada analógica, PID_3Step escala el valor de la entrada analógica al tamaño físico.

Si desea acondicionar primero el valor de la entrada analógica, es preciso escribir un programa propio para su acondicionamiento. Por ejemplo, el valor real no es directamente proporcional al valor de la entrada analógica. El valor real acondicionado debe estar en formato de coma flotante.

Procedimiento

Para utilizar directamente el valor de la entrada analógica, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input_PER".
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica.

Para utilizar el valor real acondicionado en el formato de coma flotante, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input".
2. Seleccione "Instrucción" como fuente.
3. Indique el nombre de la variable en la que está almacenado el valor real acondicionado.

Realimentación de posición

La configuración de la realimentación de posición depende del elemento final de control utilizado.

- Actuador sin realimentación de posición
- Elemento final de control con señales de tope digitales
- Actuador con realimentación de posición analógica
- Actuador con realimentación de posición analógico y señales de tope

Elemento final de control sin realimentación de posición

Para configurar PID_3Step para un elemento final de control sin realimentación de posición, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "No Feedback".

Elemento final de control con señales de tope digitales

Para configurar PID_3Step para un elemento final de control con señales de tope, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "No Feedback".
2. Active la casilla de verificación "Señal tope del actuador".
3. Seleccione "Instrucción" como fuente para Actuator_H y Actuator_L.
4. Introduzca las direcciones de las entradas digitales para Actuator_H y Actuator_L.

Elemento final de control con realimentación de posición analógica

Para configurar PID_3Step para un elemento final de control con realimentación de posición analógica, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "Feedback" o "Feedback_PER".
   - En Feedback_PER utilice el valor de la entrada analógica. El escalado de Feedback_PER se configura en la configuración del elemento final de control.
   - En Feedback procese el valor de la entrada analógica a través de su programa de usuario.
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica o la variable de su programa de usuario.

Elemento final de control con realimentación de posición analógica y señales de tope

Para configurar PID_3Step para un elemento final de control con realimentación de posición analógica y señales de tope, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "Feedback" o "Feedback_PER".
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica o la variable de su programa de usuario.
4. Active la casilla de verificación "Señal tope del actuador".
5. Seleccione "Instrucción" como fuente para Actuator_H y Actuator_L.
6. Introduzca las direcciones de las entradas digitales para Actuator_H y Actuator_L.

Valor de salida

PID_3Step ofrece un valor de salida analógico (Output_PER) y valores de salida digitales (Output_UP, Output_DN). El valor de salida que se utilice dependerá del actuador.

- Output_PER
  El actuador se activa a través de una salida analógica y se controla con una señal continua, como puede ser 0 a 10 V, 4 a 20 mA.
- Output_UP, Output_DN
  El actuador se controla a través de dos salidas digitales.

Procedimiento

Para utilizar el valor de salida analógico, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada Output (analógica).
2. Seleccione "Instrucción".
3. Introduzca la dirección de la salida analógica.

Para utilizar el valor de salida digital, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output (digital)".
2. Seleccione "Instrucción" para Output_UP y Output_DN.
3. Introduzca las direcciones de las salidas digitales.

Para acondicionar el valor de salida mediante el programa de usuario, proceda del siguiente modo:

1. En la lista desplegable "Output" seleccione la entrada adecuada al actuador.
2. Seleccione "Instrucción".
3. Indique el nombre de la variable que utilizará para acondicionar el valor de salida.
4. Transfiera el valor de salida acondicionado hasta el actuador a través de una salida analógica o digital de la CPU.

Ajustes del valor real

Escalar valor real

Si en los ajustes básicos ha configurado el uso de Input_PER, deberá convertir el valor de la entrada analógica a la magnitud física del valor real. En el cuadro de visualización Input_PER se muestra la configuración actual.

Si el valor real es directamente proporcional al valor de la entrada analógica, Input_PER se escalará a partir de un par de valores inferior y superior.

Procedimiento

Para escalar el valor real, haga lo siguiente:

1. En los campos de entrada "Valor real inferior escalado" y "abajo", introduzca el par de valores inferior.
2. En los campos de entrada "Valor real superior escalado" y "arriba", introduzca el par de valores superior.

La configuración hardware contiene pares de valores predeterminados. Para utilizar los pares de valores contenidos en la configuración hardware, haga lo siguiente:

1. Marque la instrucción PID_3Step en el editor de programación.
2. En los ajustes básicos, interconecte Input_PER con una entrada analógica.
3. En los ajustes del valor real, haga clic en el botón "Ajuste automático".

Los valores existentes se sobrescribirán con los valores de la configuración hardware.

Límites del valor real

Como valores límite para el sistema regulado defina los límites superior e inferior absolutos del valor real de modo que sean razonables. Tan pronto como dichos valores se rebasen por exceso o defecto, se producirá un error (ErrorBits = 0001h). La optimización se cancela cuando se rebasan los valores reales. Defina en los ajustes del actuador cómo debe reaccionar PID_3Step en caso de error en el modo automático.

Ajustes del actuador

Elemento final de control (actuador)

Tiempos específicos del actuador

Para proteger el elemento final de control frente a daños, ajuste el tiempo de posicionamiento del motor y los tiempos mínimos de conexión y desconexión. Los datos figuran en la hoja de datos del elemento final de control.

El tiempo de posicionamiento del motor es el tiempo en segundos que requiere el motor para desplazar el elemento final de control del estado cerrado al estado abierto. El tiempo de posicionamiento del motor puede calcularse durante la puesta en marcha.

El tiempo de posicionamiento del motor es remanente. Si modifica manualmente el tiempo de posicionamiento del motor, debe cargar PID_3Step por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Si se utiliza Output_UP y Output_DN, debe reducirse la frecuencia de conmutación con los tiempos mínimos de conexión y desconexión.

En el modo automático se acumulan los tiempos calculados de conexión y desconexión y estos no surten efecto hasta que la suma sea igual o superior al tiempo mínimo de conexión o desconexión.

En el modo manual, Manual_UP = TRUE o Manual_DN = TRUE permiten controlar el elemento final de control al menos en el tiempo mínimo de conexión o desconexión.

Comportamiento en caso de error

PID_3Step está preajustado de manera que, en caso de error, la regulación permanece activa en la mayoría de los casos. Si en el modo de regulación se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación. Compruebe en tal caso el parámetro Errorbits y elimine la causa de error.

ATENCIÓN
Su instalación puede sufrir daños. Si en caso de error se emite "Valor actual para la duración del error" o "Valor de salida sustitutivo mientras dure el error", PID_3Step se mantiene en modo automático también cuando se rebasan los límites del valor real. De este modo, su instalación puede sufrir daños. Configure un comportamiento en caso de error para su sistema regulado que proteja su instalación de daños.

En caso de error, PID_3Step emite un valor de salida configurable:

- Valor actual
  PID_3Step está desconectado y ya no modifica la posición del elemento final de control.
- Valor actual para la duración del error
  Las funciones de regulación de PID_3Step están desactivadas y ya no se modifica la posición del elemento final de control.
  Si se producen los siguientes errores en el modo automático, PID_3Step regresa a dicho modo en cuanto los errores en cuestión dejan de existir.
  - 0002h: Valor no válido en el parámetro Input_PER.
  - 0200h: Valor no válido en el parámetro Input.
  - 0400h: Error al calcular el valor de salida.
  - 1000h: Valor no válido en el parámetro Setpoint.
  - 2000h: Valor no válido en el parámetro Feedback_PER.
  - 4000h: Valor no válido en el parámetro Feedback.
  - 8000h: Error en la realimentación digital de posición.
  - 20000h: Valor no válido en la variable SavePosition.

  Si se producen uno o varios de los errores siguientes, PID_3Step permanece en modo automático:
  - 0001h: El parámetro Input se encuentra fuera de los límites del valor real.
  - 0800h: Error en tiempo de muestreo
  - 40000h: Valor no válido en el parámetro Disturbance.

  Si aparece un error en el modo manual, PID_3Step permanece en dicho modo.
  Si durante la optimización o la medición del tiempo de posicionamiento se produce un error, PID_3Step cambia al modo de operación en que se había comenzado la optimización o la medición del tiempo de posicionamiento. La optimización no se cancela con los siguientes errores únicamente:
  - 0020h: La optimización inicial no está permitida durante la optimización fina.
- Valor de salida sustitutivo
  PID_3Step desplaza el elemento final de control (actuador) al valor de salida sustitutivo y se desconecta.
- Valor de salida sustitutivo mientras dure el error
  PID_3Step desplaza el elemento final de control (actuador) al valor de salida sustitutivo. Una vez alcanzado el valor de salida sustitutivo, PID_3Step se comporta como se describe en "Valor actual mientras dure el error".

El valor de salida sustitutivo se introduce en "%".

En los actuadores sin realimentación de posición analógica, solo es posible aproximar de forma exacta los valores de salida sustitutivos 0% y 100%. Un valor de salida sustitutivo diferente del 0% o del 100% se aproxima a través de una realimentación de posición simulada internamente. Sin embargo, con este procedimiento el valor de salida sustitutivo no podrá aproximarse nunca de forma exacta.

En los actuadores con realimentación de posición analógica, es posible aproximar de forma exacta todos los valores de salida sustitutivos.

Escalar realimentación de posición

Escalar realimentación de posición

Si con el ajuste básico se ha elegido el uso de Feedback_PER, el valor de la entrada analógica debe convertirse en %. En el cuadro de visualización "Feedback" se muestra el ajuste actual.

Feedback_PER se escala a partir de una pareja de valores inferiores y superiores.

1. En los campos de entrada "Tope inferior" y "Abajo" introduzca la pareja de valores inferiores.
2. En los campos de entrada "Tope superior" y "Arriba" introduzca la pareja de valores superiores.

El valor de "Tope inferior" debe ser inferior al de "Tope superior"; por su parte, "Abajo" debe ser inferior a "Arriba".

Los valores válidos para el "tope superior" y el "tope inferior" dependen de:

- Sin Feedback, Feedback, Feedback_PER
- Output (analógico), Output (digital)

Output	Feedback	Tope inferior	Tope superior
Output (digital)	Sin Feedback	No ajustable (0.0 %)	No ajustable (100.0 %)
Output (digital)	Feedback	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (digital)	Feedback_PER	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (analógico)	Sin Feedback	No ajustable (0.0 %)	No ajustable (100.0 %)
Output (analógico)	Feedback	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (analógico)	Feedback_PER	-100.0 % o 0.0 %	0.0 % o +100.0 %

Límites del valor de salida

Limitar el valor de salida

Los límites del valor de salida pueden rebasarse por exceso o por defecto durante la medición del tiempo de posicionamiento y en el modo = 10. En los demás modos de operación, el valor de salida se limita a estos valores.

Introduzca en los campos de entrada "Lím. sup. v. de salida" y "Lím. inf. v. de salida" los límites absolutos del valor de salida. Los límites del valor de salida deben encontrarse dentro del "Tope inferior" o el "Tope superior".

Si no existe ningún Feedback y está ajustado Output (digital), no se puede limitar el valor de salida. Output_UP y Output_DN se restablecen entonces si se cumple Actuator_H = TRUE o Actuator_L = TRUE. Si tampoco existen señales de tope, Output_UP y Output_DN se restablecen después de un tiempo de desplazamiento del 150% del tiempo de posicionamiento del motor.

Ajustes avanzados

Monitorización del valor real

Configure un límite de advertencia inferior y uno superior para el valor real en la ventana de configuración "Monitorización del valor real". Si durante el funcionamiento se rebasa uno de los límites de advertencia por defecto o por exceso, se muestra una advertencia en la instrucción "PID_3Step".

- En el parámetro de salida "InputWarning_H", si se ha rebasado por exceso el límite superior de advertencia.
- En el parámetro de salida "InputWarning_L", si se ha rebasado por defecto el límite inferior de advertencia.

Los límites de advertencia deben encontrarse dentro de los límites superior e inferior del valor real.

Si no introduce ningún valor, se utilizan los límites superior e inferior del valor real.

Ejemplo

Límite superior del valor real = 98 °C; límite superior de advertencia = 90 °C

Límite inferior de advertencia = 10 °C; límite inferior del valor real = 0 °C

PID_3Step se comporta del modo siguiente:

Valor real	InputWarning_H	InputWarning_L	ErrorBits	Modo de operación
> 98 °C	TRUE	FALSE	0001h	Según configuración
≤ 98 °C y > 90 °C	TRUE	FALSE	0000h	Modo automático
≤ 90 °C y ≥ 10 °C	FALSE	FALSE	0000h	Modo automático
< 10°C y ≥ 0 °C	FALSE	TRUE	0000h	Modo automático
< 0 °C	FALSE	TRUE	0001h	Según configuración

Configure en los ajustes del actuador cómo debe reaccionar PID_3Step si se rebasa el límite superior o inferior del valor real.

Parámetros PID

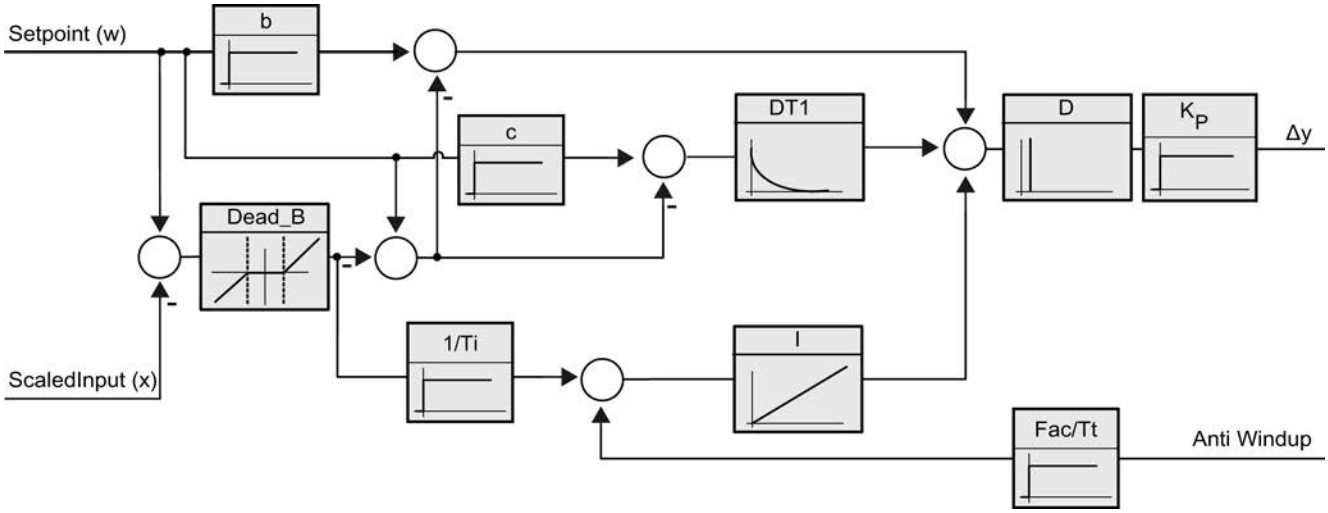
En la ventana de configuración "Parámetros PID" se visualizan los parámetros PID. Durante la optimización los parámetros PID se adaptan al sistema regulado. No es necesario introducir los parámetros PID de forma manual.

El algoritmo PID funciona de acuerdo con la siguiente fórmula:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
Δy	Valor de salida del algoritmo PID
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada T1 = a × T_D)
T_D	Tiempo derivativo
c	Ponderación de la acción D

El gráfico siguiente muestra cómo entran los parámetros en el algoritmo PID.



Todos los parámetros PID son remanentes. Si introduce manualmente los parámetros PID, debe cargar PID_3Step por completo.

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Ganancia proporcional

El valor indica la ganancia proporcional del regulador. PID_3Step no funciona con una ganancia proporcional negativa. El sentido de regulación se invierte en Ajustes básicos > Tipo de regulación.

Tiempo de integración

El tiempo de integración determina el comportamiento temporal de la acción I. La desconexión de la acción I se realiza con el tiempo de integración = 0,0.

Tiempo derivativo

El tiempo de la acción derivada determina el comportamiento temporal de la acción D. La desconexión de la acción D se realiza con el tiempo derivativo = 0,0.

Coeficiente para el retardo de la acción derivada

El efecto de la acción D se retrasa mediante el coeficiente de retardo de la acción derivada.

Retardo de la acción derivada = Tiempo derivativo x coeficiente de retardo de la acción derivada

- 0.0: la acción D solo surte efecto para un ciclo y, por ello, casi no es efectiva.
- 0.5: Este valor se ha acreditado en la práctica para sistemas regulados con **una** constante de tiempo dominante.
- > 1.0: Cuanto mayor sea el coeficiente, más se retrasará el efecto de la acción D.

Ponderación de la acción P

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción P.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción P es totalmente efectiva cuando se cambia la consigna
- 0.0: la acción P no actúa al cambiar la consigna.

Si se produce una modificación del valor real, la acción P es totalmente efectiva.

Ponderación de la acción D

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción D.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción D es totalmente efectiva cuando se cambia la consigna.
- 0.0: la acción D no actúa al cambiar la consigna.

Si se produce una modificación del valor real, la acción D es totalmente efectiva.

Tiempo de muestreo algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de muestreo PID_3Step. Todas las demás funciones de PID_3Step se ejecutan con cada llamada.

Ancho de zona muerta

La zona muerta suprime los niveles del ruido en el estado estabilizado del regulador. El ancho de zona muerta indica el tamaño de la zona muerta. Con un ancho de zona muerta de 0.0, la zona muerta se encuentra desactivada.

Puesta en servicio de PID_3Step V2

Optimización inicial

La optimización inicial determina el comportamiento del proceso a un impulso del valor de salida y busca el punto de inflexión. A partir de la inclinación máxima y del tiempo muerto del sistema regulado se calculan los parámetros PID óptimos. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina.

Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. Esto tiene más probabilidades de suceder en los modos "Inactivo" o "Manual". Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

La consigna se congela durante la optimización inicial.

Requisitos

- La instrucción "PID_3Step" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- Reset = FALSE
- El tiempo de posicionamiento del motor está configurado o medido.
- PID_3Step se encuentra en el modo de operación "Inactivo", "Modo manual" o "Modo automático".
- La consigna y el valor real se encuentran dentro de los límites configurados (véase la configuración "Configuración del valor real").

Procedimiento

Para poder realizar una optimización inicial, proceda del siguiente modo:

1. Haga doble clic en la navegación del proyecto en la entrada "PID_3Step > Puesta en servicio".
2. En la lista desplegable "Modo de optimización" del área de trabajo "Optimización" seleccione la entrada "Optimización inicial".
3. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia la optimización inicial.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los fallos ocurridos. La barra de progreso muestra el progreso del paso actual.

Nota

Haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si la optimización inicial ha finalizado sin ningún mensaje de error, significa que los parámetros PID se han optimizado. PID_3Step cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si no es posible realizar la optimización inicial, PID_3Step se comporta del modo configurado en Comportamiento en caso de error.

Optimización fina

La optimización fina genera una oscilación constante y limitada del valor real. Los parámetros PID se optimizan para el punto de operación a partir de la amplitud y la frecuencia. A partir de los resultados se vuelven a calcular todos los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial. Para obtener los mejores parámetros PID, debe efectuarse una optimización inicial y una optimización fina.

PID_3Step intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan sólo una mínima influencia sobre la optimización fina. Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

La consigna se congela durante la optimización fina.

Requisitos

- La instrucción "PID_3Step" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- Reset = FALSE
- El tiempo de posicionamiento del motor está configurado o medido.
- La consigna y el valor real se encuentran dentro de los límites configurados (véase la configuración "Configuración del valor real").
- El lazo de regulación es estacionario en el punto de operación. El punto de operación se ha alcanzado cuando el valor real coincide con la consigna.
- No se esperan perturbaciones.
- PID_3Step se encuentra en el modo de operación inactivo, automático o manual.

El proceso depende de la situación de inicio

La optimización fina se desarrolla del modo siguiente en el momento del inicio:

- Modo automático
  Si desea mejorar los parámetros PID existentes mediante optimización, inicie la optimización fina desde el modo automático.
  PID_3Step regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina.
- Inactivo o manual
  Se realiza primero una optimización inicial. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina.

Procedimiento

Para poder realizar una optimización fina, proceda del siguiente modo:

1. En la lista desplegable "Modo de optimización" seleccione la entrada "Optimización fina".
2. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia el proceso de optimización fina.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los errores ocurridos. La barra de progreso muestra el progreso del paso actual.

Nota

En el grupo "Modo de optimización" haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si durante la optimización fina no se producen errores, significa que los parámetros PID se han optimizado. PID_3Step cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si se han producido errores durante la optimización fina, PID_3Step se comporta del modo configurado en Comportamiento en caso de error.

Realizar la puesta en servicio con parámetros PID manuales

Requisitos

- La instrucción "PID_3Step" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- Reset = FALSE
- El tiempo de posicionamiento del motor está configurado o medido.
- PID_3Step se encuentra en el modo de operación "Inactivo".
- La consigna y el valor real se encuentran dentro de los límites configurados (véase la configuración "Configuración del valor real").

Procedimiento

Para poner en marcha PID_3Step con parámetros PID manuales, proceda del siguiente modo:

1. Haga doble clic en la navegación del proyecto en la entrada "PID_3Step" > Configuración".
2. En la ventana de configuración haga clic en "Ajustes avanzados > Parámetros PID".
3. Active la casilla de verificación "Activar entrada manual".
4. Introduzca los parámetros PID.
5. Haga doble clic en la navegación del proyecto en la entrada "PID_3Step > Puesta en servicio".
6. Establezca una conexión online con la CPU.
7. Cargue los parámetros PID en la CPU.
8. Haga clic en el símbolo "Start PID_3Step".

Resultado

PID_3Step cambia al modo automático y realiza la regulación con los parámetros PID actuales.

Consulte también

Parámetros PID (Página 10004)

Medir el tiempo de posicionamiento del motor

Introducción

PID_3Step necesita un tiempo de posicionamiento del motor lo más preciso posible para alcanzar un buen resultado de regulación. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento.

Si utiliza actuadores con realimentación de posición o con señales de tope, es posible medir el tiempo de posicionamiento del motor durante la puesta en servicio. Los límites del valor de salida no se tienen en cuenta durante la medición del tiempo de posicionamiento del motor. El actuador puede desplazarse hasta el tope superior o inferior.

Si no se dispone de una realimentación de posición ni de señales de tope, no es posible medir el tiempo de posicionamiento del motor.

Actuadores con realimentación de posición analógica

Para medir el tiempo de posicionamiento del motor con realimentación de posición, proceda del siguiente modo:

Requisitos

- En la configuración básica se ha seleccionado Feedback o Feedback_PER y la señal se interconecta.
- Se ha establecido una conexión online con la CPU.

1. Active la casilla de verificación "Utilizar realimentación de posición".
2. En el campo de entrada "Posición de destino" indique hacia dónde desea mover el elemento final de control.
   La realimentación de posición actual (posición de inicio) se muestra. La diferencia entre "Posición de destino" y "Realimentación de posición" debe ser al menos el 50% del rango admisible para el valor de salida.
3. Haga clic en el símbolo "Start".

Resultado

El elemento final de control se desplaza hasta la posición de destino desde la posición de inicio. La medición del tiempo se inicia de inmediato y finaliza en cuanto el elemento final de control ha alcanzado la posición de destino. El tiempo de posicionamiento del motor se calcula según la fórmula:

Tiempo de posicionamiento del motor = (Límite superior del valor de salida - Límite inferior del valor de salida) x Tiempo de medición /VALOR ABSOLUTO(Posición de destino - Posición de inicio).

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se almacena en el bloque de datos de instancia de la CPU y se muestra en el campo "Tiempo de actuación medido" Si ha finalizado la medición del tiempo de posicionamiento y ActivateRecoverMode = TRUE, PID_3Step pasa al modo desde el que

se inició la medición. Si ha finalizado la medición del tiempo de posicionamiento y ActivateRecoverMode = FALSE, PID_3Step pasa al modo de operación "Inactivo".

Nota

Para adoptar el tiempo de posicionamiento del motor medido en el proyecto, haga clic en el icono "Cargar tiempo pos. medido".

Actuadores con señales de tope

Para medir el tiempo de posicionamiento de los actuadores con señales de tope, proceda del siguiente modo:

Requisitos

- En la configuración básica la casilla de verificación "Señales de tope" está activada y los actuadores H y L están interconectados.
- Se ha establecido una conexión online con la CPU.

Para medir el tiempo de posicionamiento del motor con señales de tope, proceda del siguiente modo:

1. Active la casilla de verificación "Utilizar señales de tope del actuador".
2. Seleccione el sentido en el que desee mover el elemento final de control.
   - Abrir - Cerrar - Abrir
     El elemento final de control se mueve primero hasta el tope superior, después, hasta el tope inferior y, de nuevo, hasta el tope superior.
   - Cerrar - Abrir - Cerrar
     El elemento final de control se mueve primero hasta el tope inferior, después, hasta el tope superior y, de nuevo, hasta el tope inferior.
3. Haga clic en el símbolo "Start".

Resultado

El elemento final de control se mueve en el sentido seleccionado. La medición del tiempo se inicia cuando el elemento final de control ha alcanzado el primer tope y finaliza cuando el elemento final de control alcanza este tope por segunda vez. El tiempo medido dividido entre dos da lugar al tiempo de posicionamiento del motor.

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se almacena en el bloque de datos de instancia de la CPU y se muestra en el campo "Tiempo de actuación medido" Si ha finalizado la medición del tiempo de posicionamiento y ActivateRecoverMode = TRUE, PID_3Step pasa al modo desde el que se inició la medición. Si ha finalizado la medición del tiempo de posicionamiento y ActivateRecoverMode = FALSE, PID_3Step pasa al modo de operación "Inactivo".

Cancelar medición del tiempo de posicionamiento

Una vez cancelada la medición del tiempo de posicionamiento con el botón Stop, PID_3Step cambia al modo de operación "Inactivo".

PID_3Step V1

Configuración de PID_3Step V1

Ajustes básicos

Introducción

Configure las propiedades siguientes del objeto tecnológico PID_3Step en el área de ajustes básicos de la ventana de inspección o de configuración.

- Magnitud física
- Sentido de regulación
- Comportamiento en arranque tras un reset
- Consigna (sólo en la ventana de inspección)
- Valor real (sólo en la ventana de inspección)
- Valor de salida (sólo en la ventana de inspección)
- Realimentación de posición (sólo en la ventana de inspección)

Consigna, valor real, valor de salida y realimentación de posición

La consigna, el valor real, el valor de salida y la realimentación de posición sólo pueden configurarse en la ventana de inspección del editor de programación. Para cada valor elija el origen:

- DB de instancia
  Se utiliza el valor que se encuentra almacenado en el DB de instancia.
  El valor debe actualizarse en el DB de instancia del programa de usuario.
  En la instrucción no puede haber ningún valor.
  Modificación posible desde HMI.
- Instrucción
  Se utiliza el valor que se interconecta a la instrucción.
  Cada vez que se llama la instrucción, el valor se escribe en el DB de instancia.
  No se puede modificar desde HMI.

Tipo de regulación

Magnitud física

En el grupo "Tipo de regulación" seleccione la magnitud física y la unidad de la consigna y el valor real. La consigna y el valor real se muestran en esta unidad.

Sentido de regulación

Por lo general, un aumento del valor de salida debe lograr un aumento del valor real. En este caso, se habla de un sentido de regulación normal.

PID_3Step no funciona con ganancia proporcional negativa. Para reducir el valor real con un valor de salida más elevado, active la casilla de verificación "Inversión sentido de regulación".

Ejemplos

- Al abrir una válvula de escape se reduce el nivel de llenado de un recipiente.
- Si se aumenta la potencia de refrigeración, disminuye la temperatura.

Comportamiento en arranque tras un reset

Para cambiar inmediatamente al último modo de operación activo tras un rearranque completo de la CPU, active la casilla de verificación "Activar último modo de operación después del rearranque de la CPU".

Si la casilla de verificación está desactivada, PID_3Step se queda en el modo de operación "Inactivo".

Consigna

Procedimiento

Para especificar una consigna fija, proceda del siguiente modo:

1. Seleccione "DB de instancia".
2. Introduzca una consigna, como puede ser 80 °C.
3. En caso necesario, elimine una entrada de la instrucción.

Para especificar una consigna variable, proceda del siguiente modo:

1. Seleccione "Instrucción".
2. Indique el nombre de la variable REAL en la que está almacenada la consigna.
   Es posible asignar varios valores de forma controlada por programa a la variable REAL para, por ejemplo, modificar la consigna de tiempo de forma controlada.

Valor real

Si utiliza directamente el valor de la entrada analógica, PID_3Step escala el valor de la entrada analógica al tamaño físico.

Si desea acondicionar primero el valor de la entrada analógica, es preciso escribir un programa propio para su acondicionamiento. Por ejemplo, el valor real no es directamente proporcional al valor de la entrada analógica. El valor real acondicionado debe estar en formato de coma flotante.

Procedimiento

Para utilizar directamente el valor de la entrada analógica, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input_PER".
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica.

Para utilizar el valor real acondicionado en el formato de coma flotante, proceda del siguiente modo:

1. En la lista desplegable "Input", seleccione la entrada "Input".
2. Seleccione "Instrucción" como fuente.
3. Indique el nombre de la variable en la que está almacenado el valor real acondicionado.

Realimentación de posición

La configuración de la realimentación de posición depende del elemento final de control utilizado.

- Actuador sin realimentación de posición
- Elemento final de control con señales de tope digitales
- Actuador con realimentación de posición analógica
- Actuador con realimentación de posición analógico y señales de tope

Elemento final de control sin realimentación de posición

Para configurar PID_3Step para un elemento final de control sin realimentación de posición, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "No Feedback".

Elemento final de control con señales de tope digitales

Para configurar PID_3Step para un elemento final de control con señales de tope, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "No Feedback".
2. Active la casilla de verificación "Señal tope del actuador".
3. Seleccione "Instrucción" como fuente para Actuator_H y Actuator_L.
4. Introduzca las direcciones de las entradas digitales para Actuator_H y Actuator_L.

Elemento final de control con realimentación de posición analógica

Para configurar PID_3Step para un elemento final de control con realimentación de posición analógica, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "Feedback" o "Feedback_PER".
   - En Feedback_PER utilice el valor de la entrada analógica. El escalado de Feedback_PER se configura en la configuración del elemento final de control.
   - En Feedback procese el valor de la entrada analógica a través de su programa de usuario.
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica o la variable de su programa de usuario.

Elemento final de control con realimentación de posición analógica y señales de tope

Para configurar PID_3Step para un elemento final de control con realimentación de posición analógica y señales de tope, proceda del siguiente modo:

1. En la lista desplegable "Feedback", seleccione la entrada "Feedback" o "Feedback_PER".
2. Seleccione "Instrucción" como fuente.
3. Introduzca la dirección de la entrada analógica o la variable de su programa de usuario.
4. Active la casilla de verificación "Señal tope del actuador".
5. Seleccione "Instrucción" como fuente para Actuator_H y Actuator_L.
6. Introduzca las direcciones de las entradas digitales para Actuator_H y Actuator_L.

Valor de salida

PID_3Step ofrece un valor de salida analógico (Output_PER) y valores de salida digitales (Output_UP, Output_DN). El valor de salida que se utilice dependerá del actuador.

- Output_PER
  El actuador se activa a través de una salida analógica y se controla con una señal continua, como puede ser 0 a 10 V, 4 a 20 mA.
- Output_UP, Output_DN
  El actuador se controla a través de dos salidas digitales.

Procedimiento

Para utilizar el valor de salida analógico, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada Output (analógica).
2. Seleccione "Instrucción".
3. Introduzca la dirección de la salida analógica.

Para utilizar el valor de salida digital, proceda del siguiente modo:

1. En la lista desplegable "Output", seleccione la entrada "Output (digital)".
2. Seleccione "Instrucción" para Output_UP y Output_DN.
3. Introduzca las direcciones de las salidas digitales.

Para acondicionar el valor de salida mediante el programa de usuario, proceda del siguiente modo:

1. En la lista desplegable "Output" seleccione la entrada adecuada al actuador.
2. Seleccione "Instrucción".
3. Indique el nombre de la variable que utilizará para acondicionar el valor de salida.
4. Transfiera el valor de salida acondicionado hasta el actuador a través de una salida analógica o digital de la CPU.

Ajuste del valor real

Configure la normalización del valor real en la ventana de configuración "Ajustes del valor real" y fije los límites absolutos del valor real.

Escalar valor real

Si en los ajustes básicos se ha elegido el uso de Input_PER, el valor de la entrada analógica debe convertirse en el tamaño físico del valor real. En el cuadro de visualización Input_PER se muestra la configuración actual.

Si el valor real es directamente proporcional al valor de la entrada analógica, Input_PER se escala a partir de una pareja de valores inferiores y superiores.

1. En los campos de entrada "Valor real inferior escalado" y "Abajo" introduzca la pareja de valores inferior.
2. En los campos de entrada "Valor real superior escalado" y "arriba", introduzca el par de valores superior.

La configuración hardware incluye una configuración predeterminada de las parejas de valores. Para utilizar los pares de valores a partir de la configuración hardware, proceda del siguiente modo:

1. Marque la instrucción PID_3Step en el editor de programación.
2. En los ajustes básicos, interconecte Input_PER con una entrada analógica.
3. En los ajustes del valor real, haga clic en el botón "Ajuste automático".
   Los valores existentes se sobrescriben con los valores de la configuración hardware.

Monitorizar el valor real

Defina los límites absolutos superior e inferior del valor real. Como límites, se deben introducir valores razonables para el sistema regulado. Durante la optimización es importante disponer de valores límite razonables para obtener los parámetros PID óptimos. La preasignación del "Límite superior del valor real" es 120%. En la entrada de periferia, el valor real puede encontrarse como máximo un 18% por encima del rango normalizado (margen de saturación). No se comunica ningún error más con este ajuste debido al rebase por exceso del "Límite

superior del valor real". Únicamente se detectan la rotura de hilo y el cortocircuito y PID_3Step se comporta según la reacción configurada en caso de error.

ATENCIÓN
Su instalación puede sufrir daños. Si ajusta valores muy altos (z. B. $-3{,}4*10^{38}$...$+3{,}4*10^{38}$) como límites del valor real, se desactiva la monitorización del valor real. Esto puede dar lugar a daños en la instalación. Configure límites del valor real razonables para su sistema regulado.

Ajuste del actuador

Tiempos específicos del actuador

Para proteger el elemento final de control frente a daños, ajuste el tiempo de posicionamiento del motor y los tiempos mínimos de conexión y desconexión. Los datos figuran en la hoja de datos del elemento final de control.

El tiempo de posicionamiento del motor es el tiempo en segundos que requiere el motor para desplazar el elemento final de control del estado cerrado al estado abierto. El elemento final de control se mueve en un sentido a como máximo el 110% del tiempo de posicionamiento del motor. El tiempo de posicionamiento del motor puede medirse durante la puesta en servicio.

Si se utiliza Output_UP y Output_DN, debe reducirse la frecuencia de conmutación con los tiempos mínimos de conexión y desconexión.

En el modo automático se acumulan los tiempos calculados de conexión y desconexión y estos no surten efecto hasta que la suma sea igual o superior al tiempo mínimo de conexión o desconexión.

En el modo manual, un flanco ascendente en Manual_UP o Manual_DN permite controlar el elemento final de control al menos en el tiempo mínimo de conexión o desconexión.

Comportamiento en caso de error

PID_3Step está preajustado de manera que, en caso de error, la regulación permanece activa en la mayoría de los casos. Si en el modo de regulación se producen errores con frecuencia, este ajuste empeorará el comportamiento de regulación. Compruebe en tal caso el parámetro Errorbits y elimine la causa de error.

En caso de error, PID_3Step emite un valor de salida configurable:

- Valor actual
  PID_3Step está desconectado y ya no modifica la posición del elemento final de control.
- Valor actual para la duración del error
  Las funciones de regulación de PID_3Step están desactivadas y ya no se modifica la posición del elemento final de control.
  Si se producen los siguientes errores en el modo automático, PID_3Step regresa a dicho modo en cuanto los errores en cuestión dejan de existir.
  - 0002h: Valor no válido en el parámetro Input_PER.
  - 0200h: Valor no válido en el parámetro Input.
  - 0800h: Error en tiempo de muestreo
  - 1000h: Valor no válido en el parámetro Setpoint.
  - 2000h: Valor no válido en el parámetro Feedback_PER.
  - 4000h: Valor no válido en el parámetro Feedback.
  - 8000h: Error en la realimentación digital de posición.

  Si aparece alguno de los errores siguientes en el modo manual, PID_3Step permanece en dicho modo.
  Si se produce un error durante la optimización o durante la medición del tiempo de posicionamiento, PID_3Step se desconecta.
- Valor de salida sustitutivo
  PID_3Step desplaza el elemento final de control (actuador) al valor de salida sustitutivo y se desconecta.
- Valor de salida sustitutivo mientras dure el error
  PID_3Step desplaza el elemento final de control (actuador) al valor de salida sustitutivo. Una vez alcanzado el valor de salida sustitutivo, PID_3Step se comporta como se describe en "Valor actual mientras dure el error".

El valor de salida sustitutivo se introduce en "%".

En los actuadores sin realimentación de posición analógica, solo es posible aproximar de forma exacta los valores de salida sustitutivos 0% y 100%. Para poder alcanzar el tope superior e inferior, el elemento final de control se desplaza en un sentido con un 110% del tiempo de posicionamiento del motor. Las señales de tope tienen prioridad. Un valor de salida sustitutivo diferente del 0% o del 100% se aproxima a través de una realimentación de posición simulada internamente. Sin embargo, con este procedimiento el valor de salida sustitutivo no podrá aproximarse nunca de forma exacta.

En los actuadores con realimentación de posición analógica, es posible aproximar de forma exacta todos los valores de salida sustitutivos.

Escalar realimentación de posición

Si con el ajuste básico se ha elegido el uso de Feedback_PER, el valor de la entrada analógica debe convertirse en %. En el cuadro de visualización "Feedback" se muestra el ajuste actual.

Feedback_PER se escala a partir de una pareja de valores inferiores y superiores.

1. En los campos de entrada "Tope inferior" y "Abajo" introduzca la pareja de valores inferiores.
2. En los campos de entrada "Tope superior" y "Arriba" introduzca la pareja de valores superiores.

El valor de "Tope inferior" debe ser inferior al de "Tope superior"; por su parte, "Abajo" debe ser inferior a "Arriba".

Los valores válidos para el "Tope superior" y el "Tope inferior" dependen de:

- Sin Feedback, Feedback, Feedback_PER
- Output (analógico), Output (digital)

Output	Feedback	Tope inferior	Tope superior
Output (digital)	Sin Feedback	No ajustable (0.0 %)	No ajustable (100.0 %)
Output (digital)	Feedback	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (digital)	Feedback_PER	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (analógico)	Sin Feedback	No ajustable (0.0 %)	No ajustable (100.0 %)
Output (analógico)	Feedback	-100.0 % o 0.0 %	0.0 % o +100.0 %
Output (analógico)	Feedback_PER	-100.0 % o 0.0 %	0.0 % o +100.0 %

Limitar el valor de salida

Los límites del valor de salida solo pueden rebasarse por exceso o por defecto durante la medición del tiempo de posicionamiento. En los demás modos de operación, el valor de salida se limita a estos valores.

Introduzca en los campos de entrada "Lím. sup. v. de salida" y "Lím. inf. v. de salida" los límites absolutos del valor de salida. Los límites del valor de salida deben encontrarse dentro del "Tope inferior" o el "Tope superior".

Si no hay ningún Feedback y se ha ajustado Output (digital), el valor de salida no se puede limitar. Las salidas digitales deben desactivarse en Actuator_H = TRUE o Actuator_L = TRUE o después de un tiempo de desplazamiento del 110% del tiempo de posicionamiento del motor.

Ajustes avanzados

Monitorización del valor real

Configure un límite de advertencia inferior y uno superior para el valor real en la ventana de configuración "Monitorización del valor real". Si durante el funcionamiento se rebasa uno de los límites de advertencia por defecto o por exceso, se muestra una advertencia en la instrucción "PID_3Step".

- En el parámetro de salida "InputWarning_H", si se ha rebasado por exceso el límite superior de advertencia.
- En el parámetro de salida "InputWarning_L", si se ha rebasado por defecto el límite inferior de advertencia.

Los límites de advertencia deben encontrarse dentro de los límites superior e inferior del valor real.

Si no introduce ningún valor, se utilizan los límites superior e inferior del valor real.

Ejemplo

Límite superior del valor real = 98 °C; límite superior de advertencia = 90 °C

Límite inferior de advertencia = 10 °C; límite inferior del valor real = 0 °C

PID_3Step se comporta del modo siguiente:

Valor real	InputWarning_H	InputWarning_L	Modo de operación
> 98 °C	TRUE	FALSE	Inactivo
≤ 98 °C y > 90 °C	TRUE	FALSE	Modo automático
≤ 90 °C y ≥ 10 °C	FALSE	FALSE	Modo automático
< 10°C y ≥ 0 °C	FALSE	TRUE	Modo automático
< 0 °C	FALSE	TRUE	Inactivo

Parámetros PID

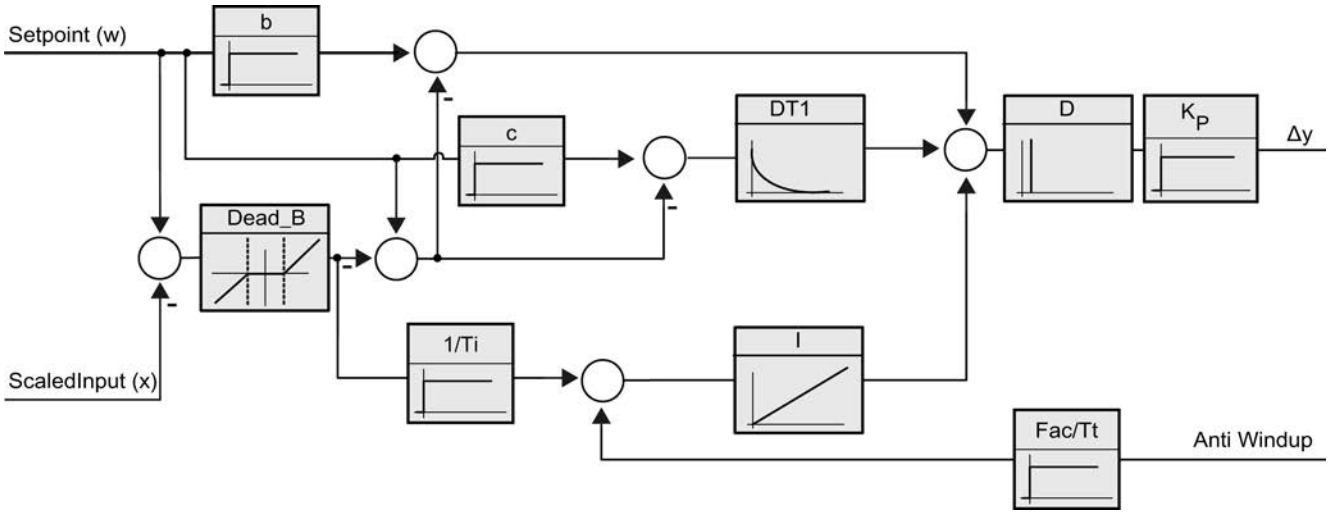
En la ventana de configuración "Parámetros PID" se visualizan los parámetros PID. Durante la optimización los parámetros PID se adaptan al sistema regulado. No es necesario introducir los parámetros PID de forma manual.

El algoritmo PID funciona de acuerdo con la siguiente fórmula:

$$\Delta y = K_p \cdot s \cdot \left[(b \cdot w - x) + \frac{1}{T_I \cdot s}(w - x) + \frac{T_D \cdot s}{a \cdot T_D \cdot s + 1}(c \cdot w - x) \right]$$

Símbolo	Descripción
Δy	Valor de salida del algoritmo PID
K_p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T_I	Tiempo de integración
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada T1 = a × T_D)
T_D	Tiempo derivativo
c	Ponderación de la acción D

El gráfico siguiente muestra cómo entran los parámetros en el algoritmo PID.



Todos los parámetros PID son remanentes. Si introduce manualmente los parámetros PID, debe cargar PID_3Step por completo.

Auto-Hotspot

Ganancia proporcional

El valor indica la ganancia proporcional del regulador. PID_3Step no funciona con una ganancia proporcional negativa. El sentido de regulación se invierte en Ajustes básicos > Tipo de regulación.

Tiempo de integración

El tiempo de integración determina el comportamiento temporal de la acción I. La desconexión de la acción I se realiza con el tiempo de integración = 0,0.

Tiempo derivativo

El tiempo de la acción derivada determina el comportamiento temporal de la acción D. La desconexión de la acción D se realiza con el tiempo derivativo = 0,0.

Coeficiente para el retardo de la acción derivada

El efecto de la acción D se retrasa mediante el coeficiente de retardo de la acción derivada.

Retardo de la acción derivada = Tiempo derivativo x coeficiente de retardo de la acción derivada

- 0.0: la acción D solo surte efecto para un ciclo y, por ello, casi no es efectiva.
- 0.5: Este valor se ha acreditado en la práctica para sistemas regulados con una constante de tiempo dominante.
- > 1.0: Cuanto mayor sea el coeficiente, más se retrasará el efecto de la acción D.

Ponderación de la acción P

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción P.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción P es totalmente efectiva cuando se cambia la consigna
- 0.0: la acción P no actúa al cambiar la consigna.

Si se produce una modificación del valor real, la acción P es totalmente efectiva.

Ponderación de la acción D

En el caso de que se produzcan cambios en la consigna, es posible atenuar la acción D.

Se recomiendan valores comprendidos entre 0.0 y 1.0.

- 1.0: La acción D es totalmente efectiva cuando se cambia la consigna.
- 0.0: la acción D no actúa al cambiar la consigna.

Si se produce una modificación del valor real, la acción D es totalmente efectiva.

Tiempo de muestreo algoritmo PID

Dado que el sistema regulado necesita cierto tiempo para responder a un cambio del valor de salida, no es razonable calcular este valor en cada ciclo. El tiempo de muestreo del algoritmo PID es el tiempo entre dos cálculos del valor de salida. Este se determina durante la optimización y se redondea a un múltiplo del tiempo de muestreo PID_3Step. Todas las demás funciones de PID_3Step se ejecutan con cada llamada.

Ancho de zona muerta

La zona muerta suprime los niveles del ruido en el estado estabilizado del regulador. El ancho de zona muerta indica el tamaño de la zona muerta. Con un ancho de zona muerta de 0.0, la zona muerta se encuentra desactivada.

Consulte también

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Puesta en servicio de PID_3Step V1

Puesta en servicio

En la zona de trabajo "Optimización" es posible observar la consigna, el valor real y el valor de salida en función del tiempo. En el registrador de curvas se soportan las siguientes funciones de puesta en servicio:

- Optimización inicial del regulador
- Optimización fina del regulador
- Observación de la regulación en curso en la ventana de curvas

Para todas las funciones es preciso realizar una conexión online a la CPU.

Principios básicos del manejo

- Seleccione el tiempo de actualización que desee en la lista desplegable "Tiempo de actualización".
  Todos los valores del área de trabajo de optimización se actualizan en el tiempo de actualización seleccionado.
- Haga clic en el botón "Inicio" en el grupo Medición si desea utilizar las funciones de puesta en servicio.
  Se inicia el registro de los valores. En el visor de curvas se registran los valores actuales de consigna, valor real y valor de salida. Se habilita el manejo de la ventana de puesta en servicio.
- Haga clic en el botón "Parar" si desea finalizar las funciones de puesta en servicio.
  Los valores mostrados en el visor de curvas pueden continuar analizándose.
- Al cerrar la ventana de puesta en servicio finaliza el registro en el visor de curvas y los valores registrados se borran.

Optimización inicial

La optimización inicial determina el comportamiento del proceso a un impulso del valor de salida y busca el punto de inflexión. A partir de la inclinación máxima y del tiempo muerto del sistema regulado se calculan los parámetros PID óptimos.

Cuanto más estable es el valor real, con mayor facilidad y precisión se pueden calcular los parámetros PID. Un ruido del valor real es aceptable siempre que la subida del valor real sea considerablemente mayor que el ruido. Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

La consigna se congela durante la optimización inicial.

Requisitos

- La instrucción "PID_3Step" se llama en un OB de alarma cíclica.
- ManualEnable = FALSE
- PID_3Step se encuentra en el modo de operación "Inactivo" o "Manual".
- La consigna y el valor real se encuentran dentro de los límites configurados (véase la configuración "Configuración del valor real").

Procedimiento

Para poder realizar una optimización inicial, proceda del siguiente modo:

1. Haga doble clic en la navegación del proyecto en la entrada "PID_3Step > Puesta en servicio".
2. En la lista desplegable "Modo de optimización" del área de trabajo "Optimización" seleccione la entrada "Optimización inicial".
3. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia la optimización inicial.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los fallos ocurridos. La barra de progreso muestra el progreso del paso actual.

Nota

Haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si la optimización inicial ha finalizado sin ningún mensaje de error, significa que los parámetros PID se han optimizado. PID_3Step cambia al modo automático y utiliza los parámetros optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si no es posible realizar la optimización inicial, PID_3Step cambia al modo de operación "Inactivo".

Optimización fina

La optimización fina genera una oscilación constante y limitada del valor real. Los parámetros PID se optimizan para el punto de operación a partir de la amplitud y la frecuencia. A partir de los resultados se vuelven a calcular todos los parámetros PID. Los parámetros PID existentes después de la optimización fina muestran en su mayoría un comportamiento de guía y ante fallos mucho mejor que los parámetros PID de la optimización inicial.

PID_3Step intenta generar automáticamente una oscilación que es mayor que el ruido del valor real. La estabilidad del valor real ejerce tan sólo una mínima influencia sobre la optimización fina. Es preciso realizar una copia de seguridad de los parámetros PID antes de volver a calcularlos.

La consigna se congela durante la optimización fina.

Requisitos

- La instrucción PID_3Step se llama en un OB se alarma cíclica.
- ManualEnable = FALSE

- El tiempo de posicionamiento del motor está configurado o medido.
- La consigna y el valor real se encuentran dentro de los límites configurados (véase la configuración "Configuración del valor real").
- El lazo de regulación es estacionario en el punto de operación. El punto de operación se ha alcanzado cuando el valor real coincide con la consigna.
- No se esperan perturbaciones.
- PID_3Step se encuentra en el modo de operación inactivo, automático o manual.

El proceso depende de la situación de inicio

La optimización fina se desarrolla del modo siguiente en el momento del inicio:

- Modo automático
  Si desea mejorar los parámetros PID existentes mediatne optimización, inicie la optimización fina desde el modo automático.
  PID_3Step regula con los parámetros PID existentes hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina.
- Inactivo o manual
  Se realiza primero una optimización inicial. Con los parámetros PID calculados, la regulación se realiza hasta que el lazo de regulación es estacionario y se han cumplido las condiciones para una optimización fina. Sólo entonces comienza la optimización fina.

Procedimiento

Para poder realizar una optimización fina, proceda del siguiente modo:

1. En la lista desplegable "Modo de optimización" seleccione la entrada "Optimización fina".
2. Haga clic en el símbolo "Start".
   - Se establece una conexión online.
   - Se inicia el registro de los valores.
   - Se inicia el proceso de optimización fina.
   - En el campo "Estado" se muestran los pasos actuales y, de haberlos, los errores ocurridos. La barra de progreso muestra el progreso del paso actual.

   Nota

   En el grupo "Modo de optimización" haga clic en el símbolo "Stop" cuando la barra de progreso haya alcanzado el 100% y se deba presuponer un bloqueo de la optimización. Compruebe la configuración del objeto tecnológico y, si procede, vuelva a iniciar la optimización.

Resultado

Si la optimización fina ha finalizado sin ningún mensaje de error, significa que los parámetros PID se han optimizado. PID_3Step cambia al modo automático y utiliza los parámetros

optimizados. Los parámetros PID optimizados se conservan al desconectar (Power OFF) y al rearrancar por completo la CPU.

Si se produjeran errores durante la optimización fina, PID_3Step cambia al modo de operación "Inactivo".

Realizar la puesta en servicio con parámetros PID manuales

Procedimiento

Para poner en marcha PID_3Step con parámetros PID manuales, proceda del siguiente modo:

1. Haga doble clic en la navegación del proyecto en la entrada "PID_3Step" > Configuración".
2. En la ventana de configuración haga clic en "Ajustes avanzados > Parámetros PID".
3. Active la casilla de verificación "Activar entrada manual".
4. Introduzca los parámetros PID.
5. Haga doble clic en la navegación del proyecto en la entrada "PID_3Step" > Puesta en servicio".
6. Establezca una conexión online con la CPU.
7. Cargue los parámetros PID en la CPU.
8. Haga clic en el símbolo "Activar regulador".

Resultado

PID_3Step cambia al modo automático y realiza la regulación con los parámetros PID actuales.

Medir el tiempo de posicionamiento del motor

Introducción

PID_3Step necesita un tiempo de posicionamiento del motor lo más preciso posible para alcanzar un buen resultado de regulación. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento.

Si utiliza actuadores con realimentación de posición o con señales de tope, es posible medir el tiempo de posicionamiento del motor durante la puesta en servicio. Los límites del valor de salida no se tienen en cuenta durante la medición del tiempo de posicionamiento del motor. El actuador puede desplazarse hasta el tope superior o inferior.

Si no se dispone de una realimentación de posición ni de señales de tope, no es posible medir el tiempo de posicionamiento del motor.

Actuadores con realimentación de posición analógica

Para medir el tiempo de posicionamiento del motor con realimentación de posición, proceda del siguiente modo:

Requisitos

- En la configuración básica se ha seleccionado Feedback o Feedback_PER y la señal se interconecta.
- Se ha establecido una conexión online con la CPU.

1. Active la casilla de verificación "Utilizar realimentación de posición".
2. En el campo de entrada "Posición de destino" indique hacia dónde desea mover el elemento final de control.
   La realimentación de posición actual (posición de inicio) se muestra. La diferencia entre "Posición de destino" y "Realimentación de posición" debe ser al menos el 50% del rango admisible para el valor de salida.
3. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se desplaza hasta la posición de destino desde la posición de inicio. La medición del tiempo se inicia de inmediato y finaliza en cuanto el elemento final de control ha alcanzado la posición de destino. El tiempo de posicionamiento del motor se calcula según la fórmula:

Tiempo de posicionamiento del motor = (Límite superior del valor de salida - Límite inferior del valor de salida) x Tiempo de medición /VALOR ABSOLUTO(Posición de destino - Posición de inicio).

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se almacena en el bloque de datos de instancia de la CPU y se muestra en el campo "Tiempo de actuación medido" Una vez finalizada la medición del tiempo de posicionamiento, PID_3Step cambia al modo de operación "Inactivo".

Nota

Para adoptar el tiempo de posicionamiento del motor medido en el proyecto, haga clic en el icono "Cargar tiempo pos. medido".

Actuadores con señales de tope

Para medir el tiempo de posicionamiento de los actuadores con señales de tope, proceda del siguiente modo:

Requisitos

- En la configuración básica la casilla de verificación "Señales de tope" está activada y los actuadores H y L están interconectados.
- Se ha establecido una conexión online con la CPU.

Para medir el tiempo de posicionamiento del motor con señales de tope, proceda del siguiente modo:

1. Active la casilla de verificación "Utilizar señales de tope del actuador".
2. Seleccione el sentido en el que desee mover el elemento final de control.
   – Abrir - Cerrar - Abrir
     El elemento final de control se mueve primero hasta el tope superior, después, hasta el tope inferior y, de nuevo, hasta el tope superior.
   – Cerrar - Abrir - Cerrar
     El elemento final de control se mueve primero hasta el tope inferior, después, hasta el tope superior y, de nuevo, hasta el tope inferior.
3. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se mueve en el sentido seleccionado. La medición del tiempo se inicia cuando el elemento final de control ha alcanzado el primer tope y finaliza cuando el elemento final de control alcanza este tope por segunda vez. El tiempo medido dividido entre dos da lugar al tiempo de posicionamiento del motor.

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se almacena en el bloque de datos de instancia de la CPU y se muestra en el campo "Tiempo de actuación medido" Una vez finalizada la medición del tiempo de posicionamiento, PID_3Step cambia al modo de operación "Inactivo".

Cancelar medición del tiempo de posicionamiento

Una vez finalizada la medición del tiempo de posicionamiento, PID_3Step cambia de inmediato al modo de operación "Inactivo". El elemento final de control deja de moverse. Es posible volver a activar PID-3Step en el registrador de curvas.

11.3.4 Regulación PID (S7-300, S7-400, S7-1500)

11.3.4.1 Utilizar funciones básicas PID

CONT_C

Objeto tecnológico CONT_C

El objeto tecnológico CONT_C proporciona un regulador PID continuo para los modos automático y manual. Equivale al bloque de datos de instancia de la instrucción CONT_C. La instrucción PULSEGEN permite configurar un regulador de impulsos.

Los componentes proporcional, integrativo (INT) y diferencial (DIF) están conectados en paralelo y se pueden conectar o desconectar de forma independiente. Con ello se pueden parametrizar los reguladores P, PI, PD y PID.

S7-1500
Todos los parámetros y variables del objeto tecnológico son remanentes y, al efectuarse la carga en el dispositivo, solo pueden modificarse si se carga CONT_C por completo.

Consulte también

Vista general de los reguladores de software (Página 9932)

Agregar objetos tecnológicos (Página 9934)

Configurar objetos tecnológicos (Página 9935)

CONT_C (Página 7078)

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Configurar el error de regulación CONT_C

Utilizar el valor real de periferia

Para utilizar el valor real en el formato de periferia en el parámetro de entrada PV_PER, proceda del siguiente modo:

1. Active la casilla de verificación "Activar periferia".
2. Si existe el valor real como magnitud física, introduzca el factor y el offset para la normalización en porcentajes.
   De este modo, el valor real se calcula según la siguiente fórmula:
   PV = PV_PER × PV_FAC + PV_OFF

Utilizar el valor real interno

Para utilizar el valor real en el formato de coma flotante en el parámetro de entrada PV_IN, proceda del siguiente modo:

1. Desactive la casilla de verificación "Activar periferia".

Error de regulación

Ajuste un ancho de zona muerta según los siguientes requisitos:

- La señal del valor real tiene ruido.
- La ganancia del regulador es alta.
- La acción D está activada.

El nivel de ruido del valor real provoca, en este caso, fuertes variaciones de la variable manipulada. La zona muerta suprime el nivel de ruido, cuando el estado del regulador es estacionario. El ancho de zona muerta indica el tamaño de la zona muerta. Con un ancho de zona muerta de 0.0, la zona muerta se encuentra desactivada.

Consulte también

Funcionamiento de CONT_C (Página 7079)

Configurar el algoritmo de regulación CONT_C

General

Para averiguar qué acciones del algoritmo de regulación están activadas, proceda del siguiente modo:

1. Seleccione una entrada de la lista "Estructura de regulador".
   Sólo puede introducir los parámetros necesarios para la estructura de regulación seleccionada.

Acción P

1. Si la estructura de regulador contiene una acción P, introduzca la "Ganancia proporcional".

Acción I

1. Si la estructura de regulador contiene una acción I, introduzca la "Tiempo de integración".
2. Para asignar un valor de inicialización a la acción I, active el botón de opción "Inicializar acción I" e introduzca el valor de inicialización.
3. Para ajustar la acción I permanentemente a este valor de inicialización, active el botón de opción "Detener acción I".

Acción D

1. Si la estructura del regulador contiene una acción D, introduzca el tiempo derivativo, la ponderación de la acción D y el tiempo de retardo.

Consulte también

Funcionamiento de CONT_C (Página 7079)

Configurar la variable manipulada CONT_C

General

Puede instroducir CONT_C en el modo manual o en el modo automático.

1. Para fijar una variable manipulado manual, active el botón de opción "Activar modo manual".
   Puede especificar una variable manipulado manual en el parámetro de entrada MAN.

Límites de la variable manipulada

La variable manipulada se limita hacia arriba o hacia abajo, para que sólo pueda aceptar valores válidos. No puede desconectar la limitación. Si se sobrepasan los límites, esto se muestra mediante los parámetros de salida QLMN_HLM y QLMN_LLM.

1. Introduzca un valor para el límite superior y el límite inferior de la variable manipulada.
   Si la variable manipulada es una magnitud física, las unidades para el límite superior y el límite inferior de la variable manipulada deben coincidir.

Normalización

La variable manipulada se puede normalizar para la salida como valor en coma flotante y valor de periferia mediante un factor y un offset, según la siguiente fórmula.

Variable manipulada normalizada = variable manipulada × factor + offset

Están preseleccionados un factor de 1.0 y un offset de 0.0.

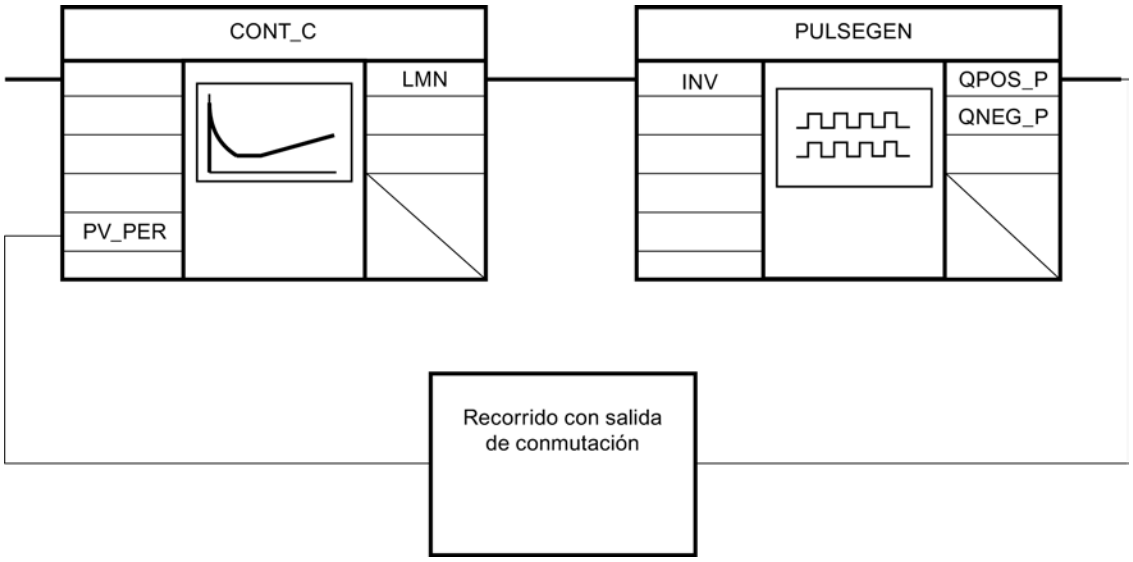
1. Introduzca un valor para factor y offset.

Consulte también

Funcionamiento de CONT_C (Página 7079)

Programar regulador de impulsos

Con el regulador continuo CONT_C y el formador de impulsos PULSEGEN, es posible realizar un regulador de valor fijo con salida de conmutación para actuadores proporcionales. La figura siguiente muestra el proceso de señales del lazo de regulación.



El regulador continuo CONT_C forma la variable manipulada LMN que el formador de impulsos PULSEGEN convierte en señales de pausa de impulso QPOS_P o QNEG_P.

Consulte también

PULSEGEN (Página 7088)

Poner en marcha CONT_C

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. Introduzca parámetros PID nuevos en los campos "P", "I", "D" y "Tiempo de retardo".
3. Haga clic en el grupo "Optimización" en el símbolo "Enviar parámetros a la CPU".
4. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .
5. Introduzca otra consigna y haga clic en el grupo "Valores actuales", en el símbolo .

6. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabaja con los nuevos parámetros PID y regula según la nueva consigna.
7. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
8. Repita los pasos del 2 hasta el 6, hasta que esté satisfecho/a con el resultado de la regulación.

CONT_S

Objeto tecnológico CONT_S

El objeto tecnológico CONT_S pone a disposición un regulador paso a paso para elementos finales de control con comportamiento integral y sirve para regular los procesos técnicos de temperatura con señales binarias de salida de variable manipulada. El objeto tecnológico corresponde al bloque de datos de instancia de la instrucción CONT_S. El modo de trabajo se basa en el algoritmo de regulación PI del regulador de muestreo. El regulador paso a paso opera sin realimentación de posición. Es posible elegir entre el modo manual y el automático.

S7-1500
Todos los parámetros y variables del objeto tecnológico son remanentes y, al efectuarse la carga en el dispositivo, solo pueden modificarse si se carga CONT_S por completo.

Consulte también

Vista general de los reguladores de software (Página 9932)

Agregar objetos tecnológicos (Página 9934)

Configurar objetos tecnológicos (Página 9935)

CONT_S (Página 7084)

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Configurar el error de regulación CONT_S

Utilizar el valor real de periferia

Para utilizar el valor real en el formato de periferia en el parámetro de entrada PV_PER, proceda del siguiente modo:

1. Active la casilla de verificación "Activar periferia".
2. Si existe el valor real como magnitud física, introduzca el factor y el offset para la normalización en porcentajes.
   De este modo, el valor real se calcula según la siguiente fórmula:
   PV = PV_PER × PV_FAC + PV_OFF

Utilizar el valor real interno

Para utilizar el valor real en el formato de coma flotante en el parámetro de entrada PV_IN, proceda del siguiente modo:

1. Desactive la casilla de verificación "Activar periferia".

Error de regulación

Ajuste un ancho de zona muerta según los siguientes requisitos:

- La señal del valor real tiene ruido.
- La ganancia del regulador es alta.
- La acción D está activada.

La componente de ruido del valor real provoca, en este caso, fuertes variaciones de la variable manipulada. La zona muerta suprime la componente de ruido cuando el estado del regulador es estacionario. El ancho de zona muerta indica el tamaño de la zona muerta. Con un ancho de zona muerta de 0.0, la zona muerta se encuentra desactivada.

Consulte también

Funcionamiento de CONT_S (Página 7084)

Configurar el algoritmo de regulación CONT_S

Algoritmo PI

1. Introduzca la "Ganancia roporcional" para la acción P.
2. Introduzca el tiempo de integración para la rapidez de respuesta de la acción I.
   Con un tiempo de integración de 0.0, la acción I se encuentra desactivada.

Consulte también

Funcionamiento de CONT_S (Página 7084)

Configurar la variable manipulada CONT_S

General

Puede introducir CONT_S en el modo manual o en el modo automático.

1. Para fijar una variable manipulado manual, active el botón de opción "Activar modo manual".
   Especifique una variable manipulada manual en los parámetros de entrada LMNUP y LMNDN.

Generador de impulsos

1. Introduzca la duración mínima de impulso y la duración mínima de pausa.
   Los valores deben ser mayores o iguales al tiempo de ciclo en el parámetro de entrada CYCLE. De esta manera, se reduce la frecuencia de maniobra.
2. Introduzca el tiempo de posicionamiento del motor.
   El valor debe ser mayor o igual al tiempo de ciclo en el parámetro de entrada CYCLE.

Consulte también

Funcionamiento de CONT_S (Página 7084)

Poner en marcha CONT_S

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. Introduzca una nueva ganancia proporcional y un nuevo tiempo de integración en los campos "P" e "I".
3. Haga clic en el grupo "Optimización" en el símbolo "Enviar parámetros a la CPU".
4. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .
5. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
6. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabaja con los nuevos parámetros y regula según la nueva consigna.

7. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
8. Repita los pasos del 2 hasta el 6, hasta que esté satisfecho/a con el resultado de la regulación.

TCONT_CP

Objeto tecnológico TCONT_CP

El objeto tecnológico TCONT_CP proporciona un regulador de temperatura continuo con generador de impulsos. Equivale al bloque de datos instancia de la instrucción TCONT_CP. El modo de trabajo se basa en el algoritmo de regulación PID del regulador de muestreo. Es posible elegir entre el modo manual y el automático.

La instrucción TCONT_CP calcula los parámetros P, I y D para el sistema regulado de forma autónoma durante la "optimización inicial". Los parámetros pueden optimizarse aún más a través de una "optimización fina". Los parámetros PID también pueden determinarse manualmente.

S7-1500
Todos los parámetros y variables del objeto tecnológico son remanentes y, al efectuarse la carga en el dispositivo, solo pueden modificarse si se carga TCONT_CP por completo.

Consulte también

Vista general de los reguladores de software (Página 9932)

Agregar objetos tecnológicos (Página 9934)

Configurar objetos tecnológicos (Página 9935)

TCONT_CP (Página 7098)

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Configurar TCONT_CP

Error de regulación

Utilizar el valor real de periferia

Para utilizar el parámetro de entrada PV_PER, proceda del siguiente modo:

1. Seleccione la entrada "Periferia" en la lista "Origen".
2. Seleccione el "Tipo de sensor ".
   Según el tipo de sensor, el valor real se normaliza según diferentes fórmulas.
   - Termopares
     estándar; PT100/NI100
     PV = 0.1 × PV_PER × PV_FAC + PV_OFFS
   - Climatiz.;
     PT100/NI100
     PV = 0.01 × PV_PER × PV_FAC + PV_OFFS
   - Intensidad/tensión
     PV = 100/27648 × PV_PER × PV_FAC + PV_OFFS
3. Introduzca factor y offset para la normalización del valor real de periferia.

Utilizar el valor real interno

Para utilizar el parámetro de entrada PV_IN, proceda del siguiente modo:

1. Seleccione la entrada "Interno" en la lista "Origen".

Error de regulación

Ajuste un ancho de zona muerta según los siguientes requisitos:

- La señal del valor real tiene ruido.
- La ganancia del regulador es alta.
- La acción D está activada.

La componente de ruido del valor real provoca, en este caso, fuertes variaciones de la variable manipulada. La zona muerta suprime la componente de ruido cuando el estado del regulador es estacionario. El ancho de zona muerta indica el tamaño de la zona muerta. Con un ancho de zona muerta de 0.0, la zona muerta se encuentra desactivada.

Consulte también

Funcionamiento de TCONT_CP (Página 7099)

Algoritmo de regulación

General

1. Introduzca el "Tiempo de muestreo del algoritmo PID".
   El tiempo de muestreo del regulador no debería exceder un 10 % del tiempo de integración del regulador (TI) calculado.
2. Si la estructura de regulador contiene una acción P, introduzca la "Ganancia proporcional".
   Una ganancia proporcional negativa invierte el sentido de regulación.

Acción P

En caso de modificaciones de la consigna, la acción P puede provocar sobreoscilaciones. Mediante la ponderación de la acción P se puede elegir con qué intensidad debe actuar la acción P al cambiar la consigna. La atenuación de la acción P se logra mediante una compensación en la acción I.

1. Para atenuar la acción P en caso de modificaciones de la consigna, introduzca la "Ponderación de la acción P".
   - 1.0: Acción P completamente efectiva en caso de modificación de la consigna
   - 0.0: Acción P inefectiva en caso de modificación de la consigna

Acción I

En caso de limitación de la variable manipulada, la acción I se detiene. En caso de un error de regulación, que mueve la acción I en dirección al área manipulada interna, la acción I vuelve a activarse.

1. Si la estructura de regulador contiene una acción I, introduzca el "Tiempo de integración".
   Con un tiempo de integración de 0.0, la acción I se encuentra desactivada.
2. Para asignar un valor de inicialización a la acción I, active el botón de opción "Inicializar acción I" e introduzca el "Valor de inicialización".
   En caso de reinicio o COM_RST = TRUE, la acción I se pone en este valor.

Acción D

1. Si la estructura del regulador contiene una acción D, introduzca el tiempo derivativo (TD) y el coeficiente DT1 (D_F).
   Si está conectada la acción D, deberá cumplirse la siguiente ecuación:
   TD = 0.5 × CYCLE× D_F.
   De ahí se calcula el tiempo de retardo según la fórmula:
   Tiempo de retardo = TD/D_F

Parametrizar el regulador PD con el punto de operación

1. Introduzca el tiempo de integración 0.0.
2. Active el botón de opción "Inicializar acción I".
3. Introduzca el punto de operación como valor de inicialización.

Parametrizar el regulador P con el punto de operación

1. Parametrice un regulador PD con el punto de operación.
2. Introduzca el tiempo derivativo 0.0.
   La acción D se desactiva.

Zona de regulación

La zona de regulación limita el rango del error de regulación. Si el error de regulación se encuentra fuera de este rango, se utilizan los límites de la variable manipulada.

Al entrar en la zona de regulación, la acción D conectada provoca una reducción muy rápida de la magnitud manipulada. Por lo tanto, la zona de regulación sólo es útil, cuando la acción D está conectada. Sin zona de regulación, principalmente sólo la acción P que va disminuyendo reduciría la magnitud manipulada. La zona de regulación provoca un régimen transitorio rápido sin sobreoscilaciones ni posteriores infraoscilaciones, en caso de que la magnitud manipulada mínima o máxima emitida esté a mucha distancia de la magnitud manipulada necesaria para el nuevo punto de operación.

1. Active el botón de opción "Activar" en el grupo "Zona de regulación".
2. Introduzca en el campo de entrada "Ancho" un valor que el valor real pueda desviarse de la consigna por encima o por debajo.

Consulte también

Funcionamiento de TCONT_CP (Página 7099)

Variable manipulada del regulador continuo

Límites de la variable manipulada

La variable manipulada se limita hacia arriba o hacia abajo, para que sólo pueda aceptar valores válidos. No puede desconectar la limitación. Si se sobrepasan los límites, esto se muestra mediante los parámetros de salida QLMN_HLM y QLMN_LLM.

1. Introduzca un valor para el límite superior y el límite inferior de la variable manipulada.

Normalización

La variable manipulada se puede normalizar para la salida como valor de punto flotante y valor de periferia mediante un factor y un offset, según la siguiente fórmula.

Variable manipulada normalizada = variable manipulada × factor + offset

Hay preasignado un factor de 1.0 y un offset de 0.0.

1. Introduzca un valor para factor y offset.

Generador de impulsos

Para un regulador continuo, el generador de impulsos debe estar desconectado.

1. Desactive el botón de opción "Activar" en el grupo "Generador de impulsos".

Consulte también

Funcionamiento de TCONT_CP (Página 7099)

Variable manipulada del regulador de impulsos

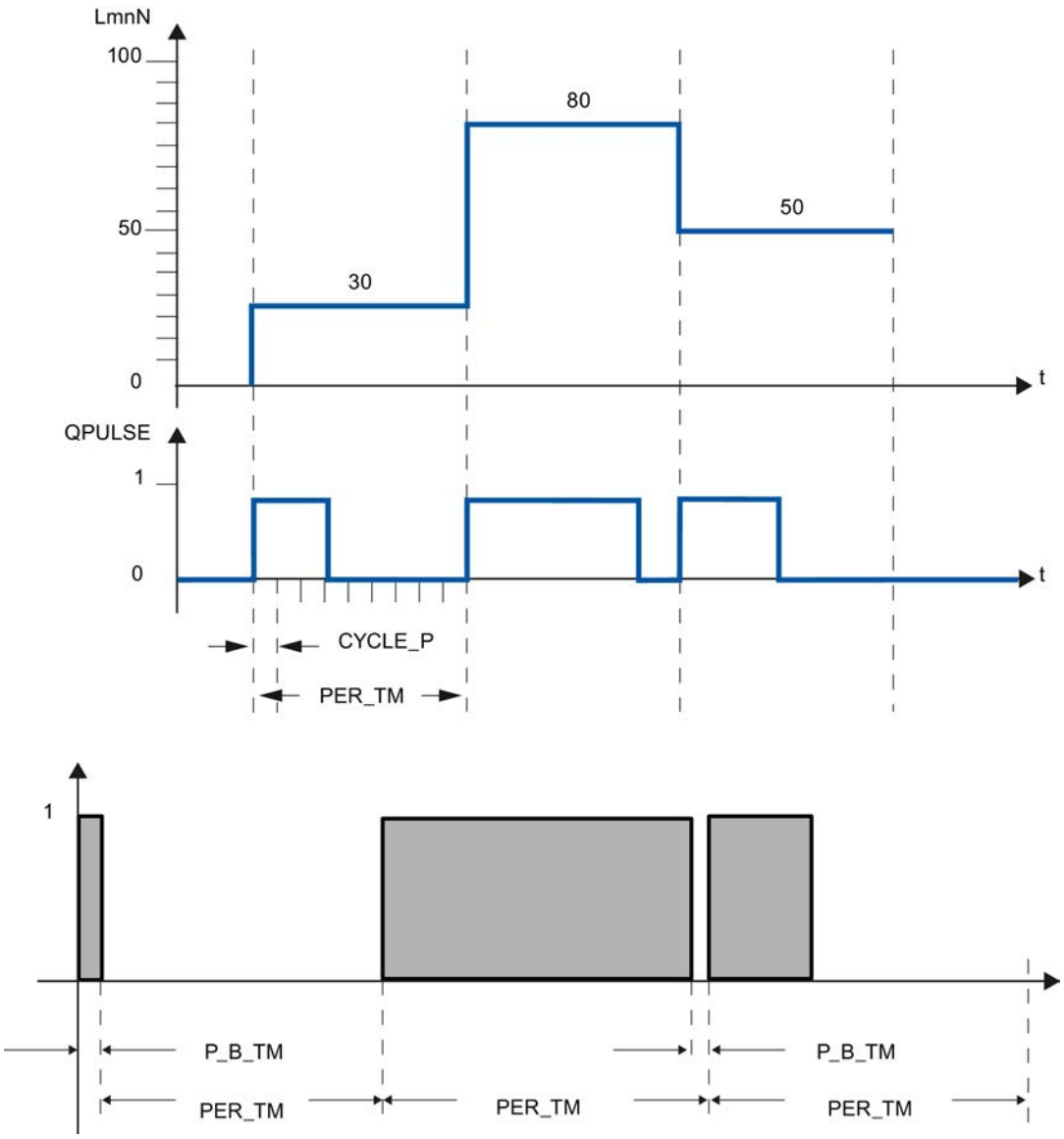
Generador de impulsos

La variable manipulada análogo (LmnN) se puede emitir con modulación de ancho de impulsos en el parámetro de salida QPULSE como un intervalor de impulsos.

Para utilizar el generador de impulsos, proceda del siguiente modo:

1. Active el botón de opción "Activar" en el grupo "Generador de impulsos".
2. Introduzca el "Tiempo de muestreo del generador de impulsos", la "Duración mínima de impulso/de pausa" y la "Duración del período".

Los siguientes gráficos aclaran la relación entre el "Tiempo de muestreo del generador de impulsos" (CYCLE_P), la "Duración mínima de impulso/de pausa" (P_B_TM) y la "Duración del período" (PER_TM):



Tiempo de muestreo del generador de impulsos

El tiempo de muestreo del generador de impulsos debe coincidir con el período del OB de alarma cíclica invocante. La duración del impulso generado es simpre un múltiplo entero de este valor. Para una resolución exacta y adecuada de la variable manipulada, debería darse la siguiente relación:
CYCLE_P ≤ PER_TM/50

Duración mínima de impulso/de pausa

Mediante la duración mínima de impulso/de pausa se evitan los tiempos de conexión o desconexión cortos en el elemento final de control. Un impulso menor que P_B_TM se suprime.

Se recomienda los valores P_B_TM ≤ 0.1 × PER_TM.

Duración del período

La duración del período no debería exceder un 20 % del tiempo de integración del regulador (TI) calculado:
PER_TM ≤ TI/5

Ejemplo de los efectos de los parámetros CYCLE_P, CYCLE y PER_TM:

Duración del período PER_TM = 10 s

Tiempo de muestreo del algoritmo PID CYCLE = 1 s

Tiempo de muestreo del generador de impulsos CYCLE_P = 100 ms.

Cada segundo se calcula una nueva variable manipulada, cada 100 ms se produce la comparación de la variable manipulada con la longitud de impulso o de pausa emitida hasta entonces.

- Si se emite un impulso, existen 2 posibilidades:
  - La variable manipulada calculada es mayor que la longitud de impulso/PER_TM existente hasta entonces. En tal caso, el impulso se prolonga.
  - La variable manipulada calculada es menor o igual que la longitud de impulso/PER_TM existente hasta entonces. En tal caso, no se emiten más señales de impulso.
- Si no se emite ningún impulso, también caben 2 posibilidades:
  - El valor (100 % - variable manipulada calculada) es mayor que la longitud de pausa/ PER_TM existente hasta entonces. En tal caso, la pausa se prolonga.
  - El valor (100 % - variable manipulada calculada) es menor o igual que la longitud de pausa/ PER_TM existente hasta entonces. En tal caso se emite una señal de impulso.

Consulte también

Funcionamiento de TCONT_CP (Página 7099)

Funcionamiento del generador de impulsos (Página 7108)

Poner en marcha TCONT_CP

Optimización de TCONT_CP

Posibilidades de utilización

La optimización del regulador puede utilizarse exclusivamente en los procesos de refrigeración o en los procesos de calefacción del tipo de sistema I. Pero también para los sistemas de orden superior del tipo II o III, puede utilizar el bloque.

Los parámetros PI/PID se calculan y ajustan automáticamente. El diseño de regulador se ha concebido con un comportamiento óptimo frente a perturbaciones. Los parámetros "de efecto

brusco" resultantes conducirían a sobreoscilaciones de entre el 10 % y el 40 % de la amplitud del escalón, en caso de escalones de consigna.

Fases de la optimización del regulador

Durante la optimización del regulador se ejecutan fases individuales, que se indican en el parámetro PHASE .

PHASE = 0

No se produce ninguna optimización. TCONT_CP funciona en modo automático o manual.

Mientras PHASE = 0, el sistema regulado debe cumplir con los requisitos para una optimización.

Al final de la optimización, TCONT_CP vuelve a cambiar en PHASE = 0.

PHASE = 1

TCONT_CP se encuentra en disposición de optimización. PHASE = 1 sólo puede iniciarse, si se cumplen las condiciones para una optimización.

Mientras PHASE = 1 se calculan los siguientes valores:

- Ruido del valor real NOISE_PV
- Subida inicial PVDT0
- Valor medio de la variable manipulada
- Tiempo de muestreo algoritmo PID CYCLE
- Tiempo de muestreo del generador de impulsos CYCLE_P

PHASE = 2

En la fase 2 se busca el punto de inversión del valor real con una variable manipulada constante. El procedimiento impide que el punto de inversión se detecte demasiado pronto debido al ruido de PV:

En el regulador de impulsos se toma el promedio de PV mediante N ciclos de impulsos y, seguidamente, se pone a disposición de la parte del regulador. En la parte del regulador se realiza otra promediación: al principio, dicha promediación está inactiva, es decir, sólo se toma el promedio de 1 ciclo cada vez. En cuanto el ruido excede una medida determinada, se duplica el número de ciclos.

Se determinan el período y la amplitud del ruido. La búsqueda del punto de inversión no se cancela y se sale de la fase 2 hasta que el gradiente es siempre menor que el ascenso máximo durante el período estimado. Sin embargo, TU y T_P_INF se calculan en el punto de inversión real.

De todas formas, la optimización no finaliza hasta que se cumplen las dos condiciones siguientes:

1. El valor real está más alejado del punto de inversión que 2*NOISE_PV.
2. El valor real ha excedido el punto de inversión en un 20 %.

Nota

En caso de excitación mediante escalón de consigna, la optimización finaliza a muy tardar cuando el valor real ha pasado el 75 % del escalón de consigna (SP_INT-PV0) (véase abajo).

PHASE = 3, 4, 5

Las fases 3, 4 y 5 duran cada una 1 ciclo.

En la fase 3, se guardan los parámetros PI/PID válidos antes de la optimización y se calculan los parámetros del proceso.

En la fase 4, se calculan los nuevos parámetros PI/PID.

En la fase 5, se calcula la nueva variable manipulada y se fija según el sistema regulado.

PHASE = 7

El tipo de sistema se comprueba en la fase 7, ya que TCONT_CP siempre cambia al modo automático tras la optmización. El modo automático se inicia con LMN = LMN0 + 0.75*TUN_DLMN como variable manipulada. La comprobación del tipo de sistema tiene lugar **en modo automático** con los parámetros del regulador que se acaban de calcular y finaliza, a lo sumo, 0,35*TA (tiempo de compensación) tras el punto de inversión. Si el orden de proceso se diferencia considerablemente del valor estimado, vuelven a calcularse los parámetros del regulador y aumenta en 1 el STATUS_D, de lo contrario, los parámetros del regulador se mantienen sin cambios.

La operación de optimización está terminada y TCONT_CP vuelve a encontrarse en PHASE = 0. El parámetro STATUS_H indica si la optimización ha finalizado correctamente.

Cancelación anticipada de la optimización

En las fases 1, 2 ó 3 se puede cancelar la optimización sin que se calculen parámetros nuevos, restableciendo TUN_ON = FALSE. El regulador se inicia en modo automático con LMN = LMN0 + TUN_DLMN. Si el regulador se encontraba en modo manual antes de la optimización, se emitirá la variable manipulada manual antigua.

Si en la fase 4, 5 ó 7 se interrumpe la optimización por medio de TUN_ON = FALSE, se conservan los parámetros del regulador ya calculados.

Condiciones para una optimización

Régimen transitorio

El proceso debe presentar un régimen transitorio estable, con retardo y asintótico.

Después de un escalón de la magnitud manipulada, el valor real debe pasar a un estado estacionario. Con ello se excluyen procesos que presentan un comportamiento oscilante

incluso sin regulación, así como sistemas regulados sin compensación (integrador en el proceso regulado).

⚠ ADVERTENCIA

Pueden producirse daños materiales considerables, lesiones físicas graves o incluso la muerte.

Durante una optimización, el parámetro MAN_ON no tiene efecto alguno. Por lo tanto, la variable manipulada o el valor real pueden tomar valores no deseados, o también extremos.

La variable manipulada se especifica mediante optimización. Para interrumpir la optimización, primero hay ajustar TUN_ON = FALSE. De este modo, MAN_ON vuelve a tener efecto.

Consolidación de un estado inicial cuasi-estacionario (fase 0)

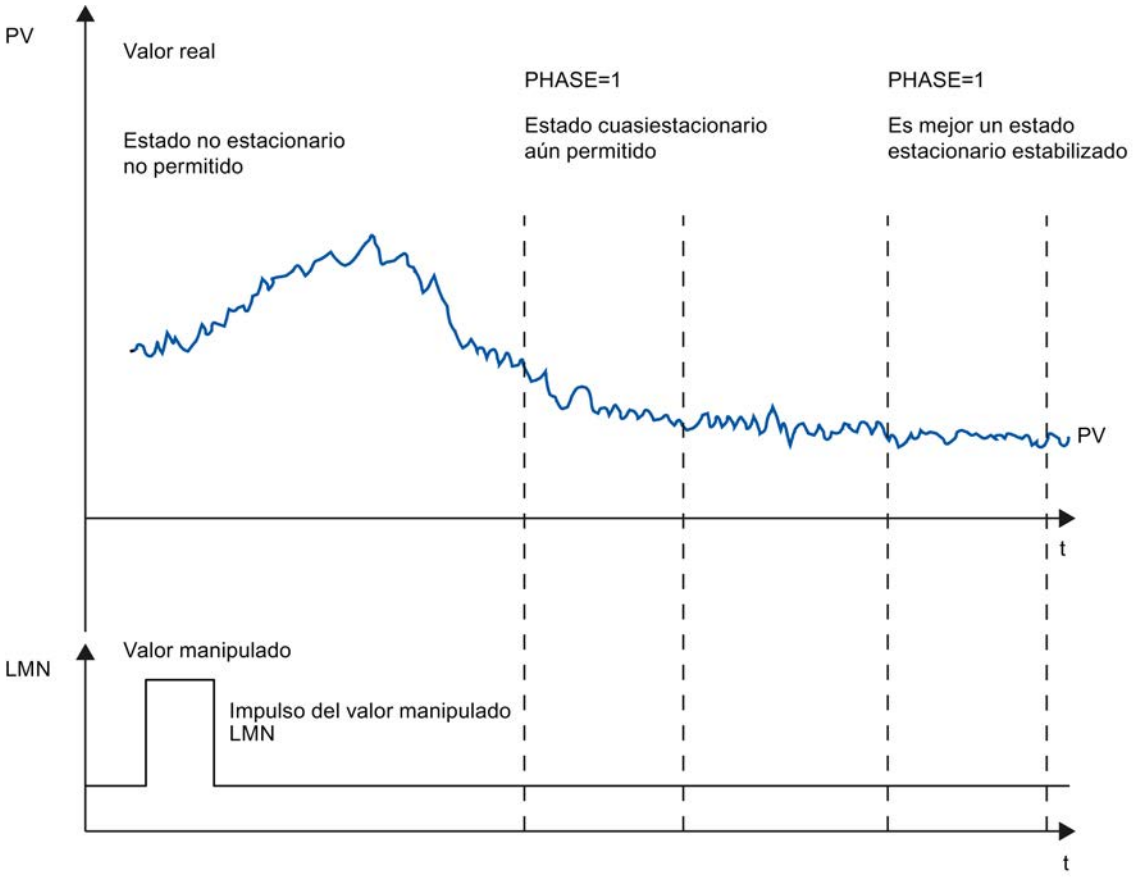
Si la magnitud regulada presentara oscilaciones de baja frecuencia, por ejemplo, debido a unos parámetros del regulador incorrectos, habrá que controlar el regulador manualmente antes de iniciar la optimización y esperar a que desaparezcan dichas oscilaciones. Como alternativa, también se puede conmutar a un regulador PI de efecto "suave" (ganancia del lazo pequeña, tiempo de integración largo).

Ahora deberá esperar hasta que se alcance el estado estacionario, es decir, hasta que el valor real y la variable manipulada sean estacionarios. También está permitido un régimen transitorio asintótico o un desplazamiento lento del valor real (estado cuasi-estacionario, consulte la figura siguiente). La magnitud manipulada debe ser constante u oscilar alrededor de un valor medio constante.

Nota

Evite modificar la magnitud manipulada justo antes del inicio de la optimización. La magnitud manipulada también se puede modificar de manera involuntaria mediante el establecimiento de las condiciones de ensayo (p. ej., cerrar una puerta abierta). Si este fuera el caso, deberá esperar como mínimo a que el valor real experimente de nuevo un régimen transitorio de forma asintótica a un estado estacionario No obstante, conseguirá mejores parámetros de regulación, si espera hasta que el proceso de régimen transitorio haya finalizado completamente.

En la siguiente figura, se muestra el régimen transitorio al estado estacionario:



Linealidad y zona de trabajo

El proceso debe presentar un comportamiento lineal en la zona de trabajo. Un comportamiento no lineal se produce, p. ej., con un cambio de un estado de agregación. La optimización debe tener lugar en una parte lineal de la zona de trabajo.

Esto significa que, tanto para la optimización como para el modo de regulación normal, los efectos no lineales dentro de dicha zona de trabajo deben ser despreciablemente pequeños. Sin embargo, es posible optimizar de nuevo el proceso cambiando el punto de operación si la optimización vuelve a ejecutarse en un entorno reducido del nuevo punto de operación y si durante la optimización no se pasa por la no linealidad.

Si se conocen determinadas no linealidades estáticas (p. ej. características de válvulas), es aconsejable compensarlas previamente con una línea poligonal para linealizar el comportamiento del proceso.

Perturbaciones en procesos de temperatura

Las perturbaciones, como la transferencia de calor a zonas colindantes, no deben influir excesivamente en el proceso global de temperatura. Así, p. ej., al optimizar las zonas de una extrusionadora, todas las zonas deben calentarse simultáneamente.

Posibilidades de la optimización

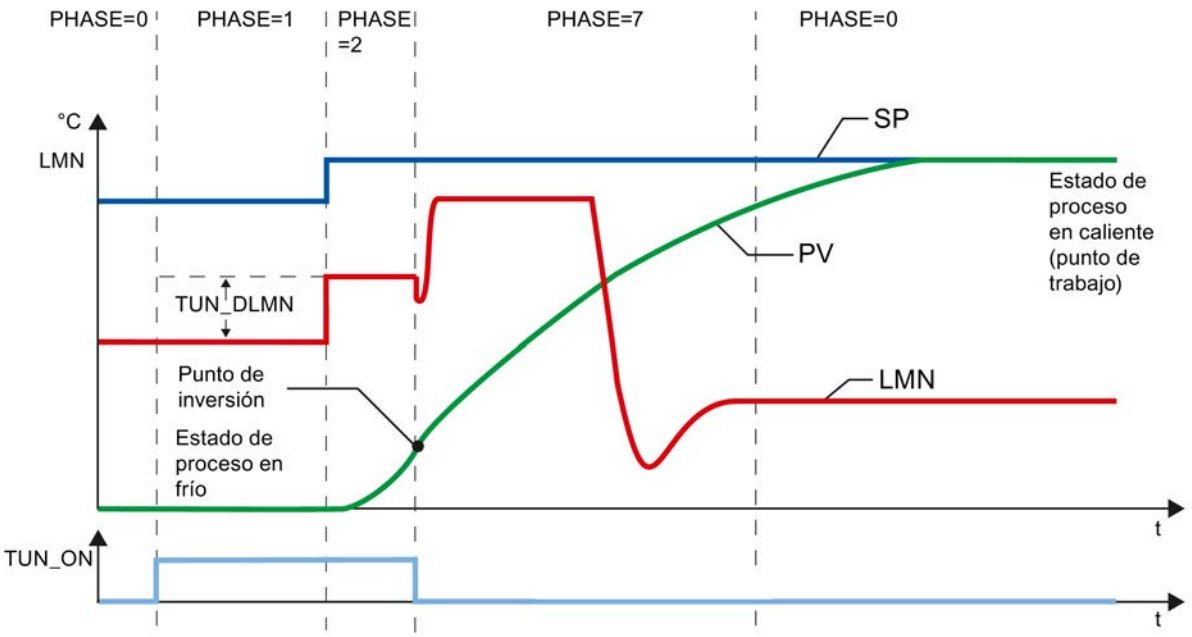
Existen las siguientes posibilidades de optimización:

- Optimización inicial
- Optimización fina
- Optimización fina manual en modo de regulación

Optimización inicial

Durante esta optimización, el punto de operación del estado frío se pone en marcha a través de un escalón de consigna.

Con TUN_ON = TRUE, cree la disposición de optimización. El regulador cambia de PHASE = 0 a PHASE = 1.



La variable manipulada de la optimización (LMN0 + TUN_DLMN) se aplica mediante modificación de la consigna (transición fase 1 -> 2). La consigna no es efectiva hasta que se alcanza el punto de inversión (sólo entonces se conmutará al modo Automático).

El delta de la excitación manipulada (TUN_DLMN) se define de acuerdo con la modificación admisible del valor real bajo responsabilidad propia. El signo de TUN_DLMN debe estar en concordancia con la modificación prevista del valor real (observar el sentido de regulación).

El escalón de consigna y TUN_DLMN deben estar coordinados. Un TUN_DLMN demasiado grande conlleva el peligro de que el punto de inversión no esté dentro del 75 % del escalón de consigna.

Sin embargo, TUN_DLMN debe ser lo suficientemente grande como para que el valor real alcance como mínimo el 22 % del escalón de consigna. En caso contrario, el procedimiento permanece en modo Optimización (fase 2).

Solución: reduzca la consigna durante la búsqueda del punto de inversión.

Nota

En procesos con un fuerte retardo, es aconsejable definir para la optimización una consigna de destino un poco más baja que el punto de operación deseado y observar con detenimiento los bits de estado y el PV (peligro de sobreoscilaciones).

Optimización únicamente en el rango lineal:

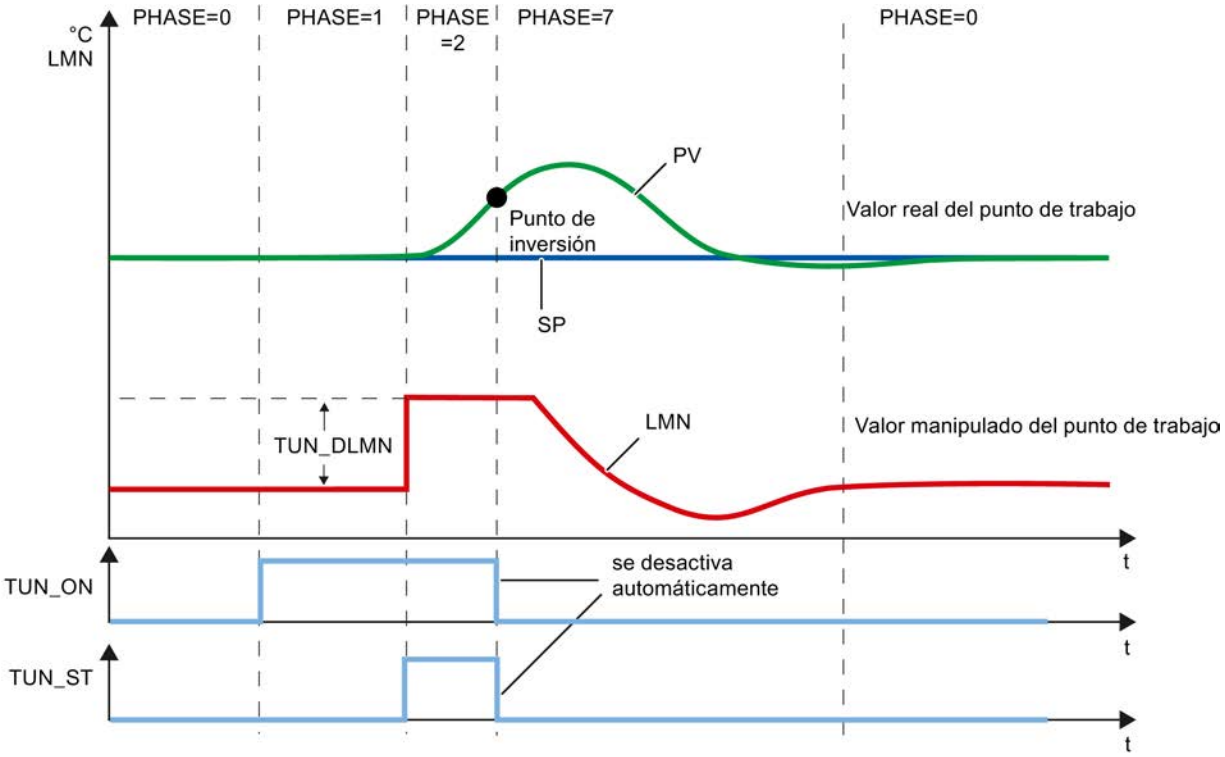
Determinados sistemas regulados (p. ej. calderas de fusión de zinc o magnesio) pasan por un rango no lineal poco antes del punto de operación (modificación del estado de agregación).

Si se selecciona con habilidad el escalón de consigna es posible limitar la optimización al rango lineal. Si el valor real ha pasado el 75 % del escalón de consigna (SP_INT-PV0) se finaliza la optimización.

Paralelamente, TUN_DLMN debería reducirse hasta que el punto de inversión se encuentre con seguridad antes de alcanzar el 75 % del escalón de consigna.

Optimización fina

Durante esta optimización, el proceso se excita, en caso de una consigna constante, mediante un salto de la variable manipulada.



La variable manipulada de la optimización (LMN0 + TUN_DLMN) se aplica mediante activación del bit de inicio TUN_ST (transición fase 1 -> 2). Si se modifica la consigna, la nueva consigna

no es efectiva hasta que no se alcanza el punto de inversión (sólo entonces se conmuta al modo Automático).

El delta de la excitación manipulada (TUN_DLMN) se define de acuerdo con la modificación admisible del valor real bajo responsabilidad propia. El signo de TUN_DLMN debe estar en concordancia con la modificación prevista del valor real (observar el sentido de regulación).

ATENCIÓN
Si la excitación se realiza mediante TUN_ST no se produce ninguna desconexión de seguridad al 75 %. La optimización finaliza cuando se alcanza el punto de inversión. Sin embargo, en procesos con ruidos es posible que se sobrepase claramente el punto de inversión.

Optimización fina manual en modo de regulación

Para conseguir un comportamiento de consigna sin sobreoscilaciones, pueden tomarse las medidas siguientes:

- Adaptar la zona de regulación
- Optimizar el comportamiento de referencia
- Amortiguar los parámetros de regulación
- Modificar los parámetros de regulación

Resultado de la optimización

La cifra de la izquierda de STATUS_H muestra el estado de optimización

STATUS_H	Resultado
0	Se han encontrado parámetros del regulador predeterminados, ninguno o ninguno nuevo.
10000	Se han encontrado parámetros del regulador adecuados
2xxxx	Se han encontrado parámetros del regulador superiores a los valores estimados; compruebe el comportamiento de regulación, lea el aviso de diagnóstico STATUS_H y repita la optimización del regulador.
3xxxx	Se han producido errores de operador; lea el aviso de diagnóstico STATUS_H y repita la optimización del regulador.

En la fase 1 ya se verificaron los tiempos de muestreo CYCLE y CYCLE_P.

Los siguientes parámetros de regulación se actualizan en TCONT_CP:

- P (ganancia proporcional GAIN)
- I (tiempo de integración TI)
- D (tiempo de acción derivativa TD)
- Ponderación de la acción P PFAC_SP
- Coeficiente DT1 (D_F)

- Zona de regulación on/off CONZ_ON
- Ancho de la zona de regulación CON_ZONE

La zona de regulación sólo se activa si el tipo de sistema es adecuado (I y II) y si el regulador PID está activado (CONZ_ON = TRUE).

En función de PID_ON se regula con el regulador PI o PID. Los parámetros antiguos del regulador se guardan y pueden volver a activarse mediante UNDO_PAR. Además, se guarda un juego de parámetros PI y PID en las estructuras PI_CON y PID_CON. Mediante LOAD_PID y la activación correspondiente de PID_ON, también es posible cambiar más adelante entre los parámetros optimizados PI o PID.

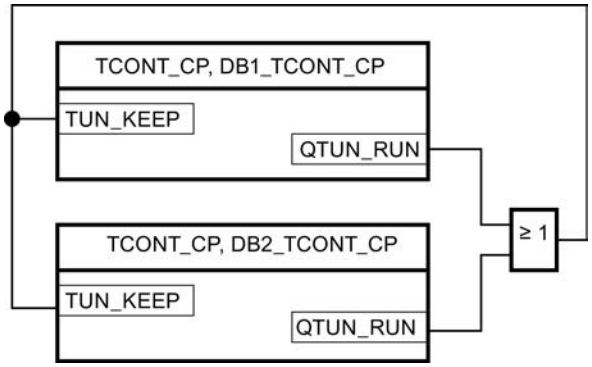
Optimización paralela de canales del regulador

Zonas colindantes (fuerte acoplamiento térmico)

Si hay dos o más reguladores que regulan la temperatura, p. ej. en una placa (es decir, dos calefacciones y dos valores reales medidos con un fuerte acoplamiento térmico), proceda del siguiente modo:

1. Combine lógicamente con la operación O las dos salidas QTUN_RUN.
2. Interconecte cada una de las dos entradas TUN_KEEP con una de las salidas del elemento O.
3. Inicie los dos reguladores predefiniendo simultáneamente un escalón de consigna o activando simultáneamente TUN_ST.

La figura siguiente representa la optimización paralela de canales del regulador:



Ventaja:

Ambos reguladores emiten LMN0 + TUN_DLMN hasta que los dos han salido de la fase 2. De este modo se impide que el regulador que finaliza primero la optimización falsifique el resultado de optimización del otro regulador al modificar su magnitud manipulada.

ATENCIÓN
Cuando se alcanza el 75 % del escalón de consigna se sale de la fase 2, con lo que se desactiva la salida QTUN_RUN. Sin embargo, el modo Automático no empezará hasta que TUN_KEEP sea 0.

Zonas colindantes (acoplamiento térmico débil)

Por norma general, debería optimizarse del mismo modo que se regula posteriormente. Si en modo productivo las zonas se desplazan conjuntamente en paralelo, de modo que las diferencias de temperatura entre las distintas zonas se mantienen iguales, durante la optimización debería aumentarse el nivel de temperatura de las zonas colindantes en correspondencia.

Las diferencias de temperatura al principio del ensayo no son relevantes, porque se compensan mediante una calefacción inicial adecuada (→ subida inicial = 0).

Síntomas de error y soluciones posibles

Interceptar errores de operador

Error de operador	STAUTS y medida	Comentario
Activación simultánea de TUN_ON y escalón de consigna o TUN_ST	Transición a la fase 1 pero sin inicio de optimización. • SP_INT = $SP_{antiguo}$ o • TUN_ST = FALSE	La modificación de la consigna se anula. De este modo se impide que el regulador se regule a la nueva consigna y abandone innecesariamente el punto de operación estacionario.
TUN_DLMN < 5 % efectivo (fin de la fase 1)	STATUS_H = 30002 • Transición a la fase 0 • TUN_ON = FALSE • SP = $SP_{antiguo}$	Cancelación de la optimización. La modificación de la consigna se anula. De este modo se impide que el regulador se regule a la nueva consigna y abandone innecesariamente el punto de operación estacionario.

Punto de inversión no alcanzado (sólo en caso de excitación mediante escalón de consigna)

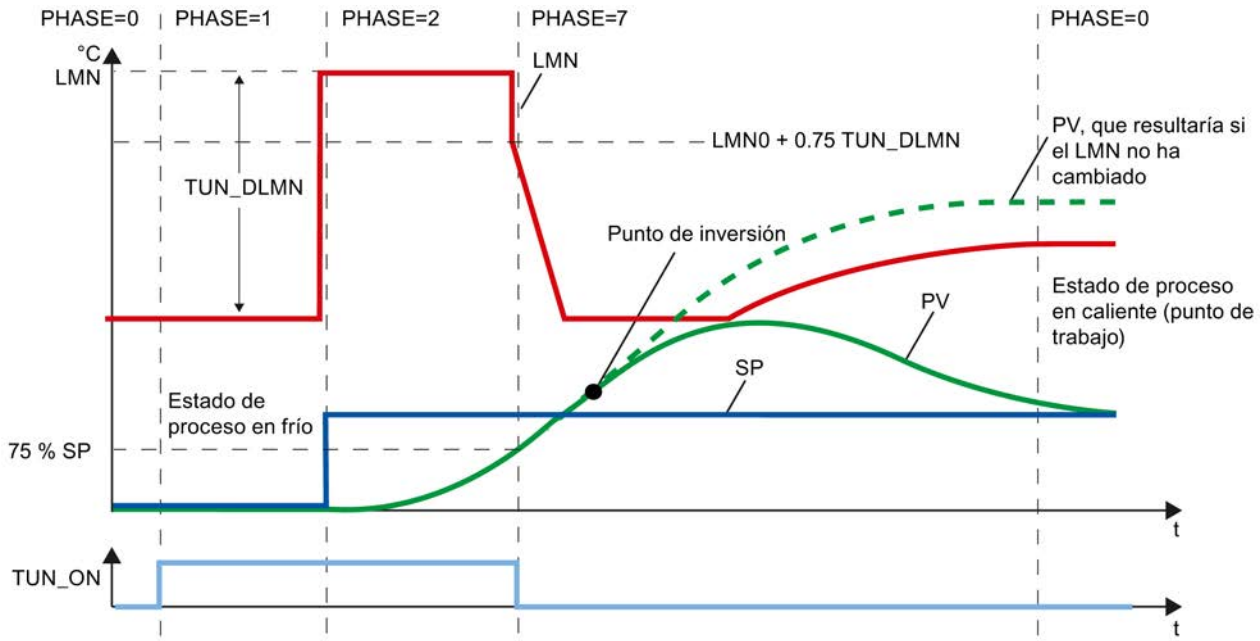
La optimización finaliza como muy tarde cuando el valor real ha pasado el 75 % del escalón de consigna (SP_INT-PV0). Esto se notifica mediante "Punto de inversión no alcanzado" en STATUS_H (2xx2x).

Siempre rige la consigna que está ajustada en cada momento. Así pues, si se reduce la consigna, es posible que la optimización finalice anticipadamente.

En casos de procesos de temperatura típicos, suele bastar con cancelar la optimización al 75 % del escalón de consigna para impedir sobreoscilaciones. Sin embargo, hay que tomar precauciones, especialmente en procesos con un retardo considerable(TU/TA > 0.1, tipo III). Si la excitación manipulada es demasiado fuerte en comparación con el escalón de consigna, el valor real tendrá muchas sobreoscilaciones (hasta el factor 3).

Si, en procesos de orden superior, el punto de inversión sigue muy alejado una vez alcanzado el 75 % del escalón de consigna, hay una sobreoscilación clara. Además, los parámetros del regulados son demasiado rigurosos. En este caso, atenúe los parámetros del regulador o repita el ensayo.

La figura siguiente representa la sobreoscilación del valor real en caso de excitación demasiado fuerte (tipo de sistema III):



En procesos de temperatura típicos, una cancelación justo ante de alcanzar el punto de inversión no es crítica respecto de los parámetros del regulador.

Si repite el ensayo, reduzca TUN_DLMN o aumente el escalón de consigna.

Principio: la variable manipulada de optimización debe concordar con el escalón de consigna.

Error de estimación con tiempo de retardo u orden

El tiempo de retardo (STATUS_H = 2x1xx o 2x3xx) o el orden (STATUS_H = 21xxx o 22xxx) no ha podido registrarse correctamente. Se sigue trabajando con un valor estimado que puede provocar parámetros no óptimos del regulador.

Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

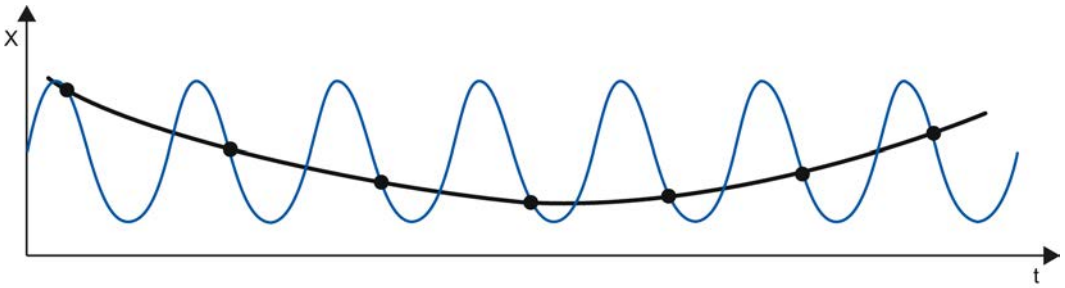
Nota

El caso especial de un proceso PT1 puro también se señala con STATUS_H = 2x1xx (TU <= 3*CYCLE). En este caso no es necesario repetir el ensayo. Atenúe los parámetros del regulador si la regulación oscila.

Calidad de las señales medidas (ruido medido, perturbaciones de baja frecuencia)

El resultado de la optimización puede verse entorpecido por ruidos medidos o perturbaciones de baja frecuencia. En tal caso, considere lo siguiente:

- En caso de ruidos medidos, seleccione una frecuencia de muestreo más bien alta que baja. Dentro de un período de ruido, el valor real debe muestrearse como mínimo dos veces. En el modo Impulso sirve de ayuda la filtración integrada del valor medio. De todas formas, requiere que el valor PV se transfiera a la instrucción en el ciclo de impulso rápido. La medida de ruido no debería exceder un 5 % del cambio de señal útil.
- Las perturbaciones de alta frecuencia ya no pueden eliminarse por filtración mediante TCONT_CP. Dichas perturbaciones ya deberían haberse filtrado en el sensor de medición para impedir el efecto aliasing.
  La figura siguiente representa el efecto aliasing con un tiempo de muestreo demasiado grande:



- En caso de perturbaciones de baja frecuencia es relativamente fácil garantizar una tasa de muestreo lo suficientemente elevada. Por otro lado, el TCONT_CP debe generar una señal medida homogénea a causa de un intervalo elevado de la filtración de valores medios. Una filtración de valores medios debe abarcar como mínimo dos períodos de ruido. De esta forma, dentro del bloque se generan rápidamente tiempos de muestreo más grandes, de modo que se merma la precisión de la optimización. Una precisión suficiente se garantiza con un mínimo de 40 períodos de ruido hasta el punto de inversión.
  Medidas posibles al repetir el ensayo:
  aumentar TUN_DLMN.

Sobreoscilación

En las situaciones siguientes pueden producirse sobreoscilaciones:

Situación	Causa	Solución
Fin de la optimización	• Excitación mediante un cambio demasiado brusco de la variable manipulada en comparación con el escalón de consigna (véase arriba). • Regulador PI activado mediante PID_ON = FALSE.	• Aumentar el escalón de consigna o reducir el escalón de la variable manipulada • Si el proceso admite un regulador PID, inicie la optimización con PID_ON = TRUE.
Optimización en fase 7	Primero se han determinado parámetros más suaves para el regulador (tipo de sistema III) que pueden provocar sobreoscilaciones en la fase 7.	-
Modo de regulación	Regulador PI y con PFAC_SP = 1.0 para el tipo de sistema I.	Si el proceso admite un regulador PID, inicie la optimización con PID_ON = TRUE.

Realizar una optimización inicial

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar los parámetros PID óptimos en la primera puesta en servicio, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización inicial".
   TCONT_CP está preparado para la optimización.
3. En el campo "Escalón de la variable manipulada" indique el valor en el que debe aumentarse la variable manipulada.
4. En el campo "Consigna", indique una consigna. El escalón de la variable manipulada se hace efectivo sólo con una nueva consigna.
5. Haga clic en el símbolo "Iniciar optimización".
   Se inicia la optimización inicial. Se muestra el estado de la optimización.

Realizar optimización fina

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar los parámetros PID óptimos en el punto de operación, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización fina".
   TCONT_CP está disponible para la optimización.
3. En el campo "Escalón de la variable manipulada" indique el valor en el que debe aumentarse la variable manipulada.
4. Haga clic en el símbolo "Iniciar optimización".
   Se inicia la optimización fina. Se muestra el estado de la optimización.

Cancelación de la optimización inicial o fina

Para cerrar una optimización inicial o fina, haga clic en el símbolo "Detener optimización".

Si los parámetros PID no se han calculado y almacenado todavía, TCONT_CP se inicia en el modo automático con LMN = LMN0 + TUN_DLMN. Si el regulador se encontraba en modo manual antes de la optimización, se emitirá la variable manipulada manual antigua.

Si los parámetros PID calculados ya están almacenados, TCONT_CP se inicia en el modo automático y funciona entonces con los parámetros PID calculados hasta la fecha.

Optimización fina manual en modo de regulación

Para conseguir un comportamiento de consigna sin sobreoscilaciones pueden tomarse las medidas siguientes:

Adaptar la zona de regulación

En la optimización se determina una zona de regulación CON_ZONE del TCONT_CP y se activa con un tipo de sistema adecuado (I y II) y un regulador PID (CONZ_ON = TRUE). En

el modo de regulación es posible modificar la zona de regulación o desactivarla completamente (con CONZ_ON = FALSE).

Nota

Si se activa la zona de regulación en procesos de orden superior (tipo III), no suele suponer ninguna ventaja, porque en este caso la zona de regulación es mayor que el rango de regulación asequible con el 100 % de la magnitud manipulada. Tampoco es ventajoso activar la zona de regulación para regulador PI.

Antes de activar a mano la zona de regulación, asegúrese de no ajustar un ancho de zona de regulación demasiado pequeño. Si el ancho de la zona de regulación ajustado fuera insuficiente, se producirán oscilaciones en las curvas de la magnitud manipulada y del valor real.

Atenuar continuamente la respuesta de regulación con PFAC_SP

La respuesta de regulación se atenúa con el parámetro PFAC_SP. Dicho parámetro determina qué proporción de la acción P será efectiva en casos de escalones de consigna.

PFAC_SP se predetermina a 0.8 mediante la optimización, independientemente del tipo de sistema; seguidamente puede modificarse el valor. Para limitar las sobreoscilaciones a un 2 % aproximadamente en caso de escalones de consigna (si los demás parámetros del regulador son correctos), son suficientes los valores siguientes para PFAC_SP:

	Tipo de sistema I	Tipo de sistema II	Tipo de sistema III
	Proceso de temperatura típico	Rango de transición	Proceso de temperatura de orden superior
PI	0.8	0.82	0.8
PID	0.6	0.75	0.96

Ajuste el factor predeterminado (0.8), especialmente en los casos siguientes:

- Tipo de sistema I con PID (0.8 -> 0.6): los escalones de consigna dentro de la zona de regulación provocan todavía un 18 % de sobreoscilaciones con PFAC_SP = 0.8.
- Tipo de sistema III con PID (0.8 -> 0.96): los escalones de consigna con PFAC_SP = 0.8 se amortiguan demasiado. Se regala un tiempo de ajuste claro.

Amortiguar los parámetros de regulación

Si en el lazo de regulación cerrado se producen oscilaciones o si hay sobreoscilaciones después de escalones de consigna, es posible girar hacia atrás la ganancia del regulador GAIN (p. ej. al 80 % del valor original) y aumentar el tiempo de acción integral TI (p. ej. al 150 % del valor original). Si la magnitud manipulada analógica del regulador continuo se convierte en señales manipuladas binarias con un generador de impulsos, es posible que aparezcan pequeñas oscilaciones continuas debido a los efectos de cuantificación. Dichas oscilaciones se eliminan aumentando la zona muerta del regulador DEADB_W.

Modificar los parámetros de regulación

Para modificar los parámetros del regulador, proceda del siguiente modo:

1. Guarde los parámetros actuales con SAVE_PAR.
2. Modifique los parámetros.
3. Pruebe el comportamiento de regulación.

Si los parámetros nuevos fueran peores que los antiguos, vuelva a ajustar los parámetros antiguos con UNDO_PAR.

Realizar una optimización fina manual

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Manual".
3. Introduzca los nuevos parámetros PID.
4. Haga clic en el grupo "Optimización" en el símbolo "Enviar parámetros a la CPU".
5. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .
6. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
7. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabaja con los nuevos parámetros PID y regula según la nueva consigna.
8. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
9. Repita los pasos del 3 hasta el 8, hasta que esté satisfecho/a con el resultado de la regulación.

TCONT_S

Objeto tecnológico TCONT_S

El objeto tecnológico TCONT_S pone a disposición un regulador paso a paso para elementos finales de control con comportamiento integrador y sirve para regular los procesos técnicos de temperatura con señales binarias de salida de variable manipulada. El objeto tecnológico corresponde al bloque de datos de instancia de la instrucción TCONT_S. El modo de trabajo se basa en el algoritmo de regulación PI del regulador de muestreo. El regulador paso a paso opera sin realimentación de posición. Es posible elegir entre el modo manual y el automático.

S7-1500
Todos los parámetros y variables del objeto tecnológico son remanentes y, al efectuarse la carga en el dispositivo, solo pueden modificarse si se carga TCONT_S por completo.

Consulte también

Vista general de los reguladores de software (Página 9932)

Agregar objetos tecnológicos (Página 9934)

Configurar objetos tecnológicos (Página 9935)

TCONT_S (Página 7120)

Cargar objetos tecnológicos en el dispositivo (Página 9937)

Configurar el error de regulación de TCONT_S

Utilizar el valor real de periferia

Para utilizar el parámetro de entrada PV_PER, proceda del siguiente modo:

1. Seleccione la entrada "Periferia" en la lista "Origen".
2. Seleccione el "Tipo de sensor".
   Según el tipo de sensor, el valor real se normaliza según diferentes fórmulas.
   - Termopares
     estándar; PT100/NI100
     PV = 0.1 × PV_PER × PV_FAC + PV_OFFS
   - Climatiz.;
     PT100/NI100
     PV = 0.01 × PV_PER × PV_FAC + PV_OFFS
   - Intensidad/tensión
     PV = 100/27648 × PV_PER × PV_FAC + PV_OFFS
3. Introduzca factor y offset para la normalización del valor real de periferia.

Utilizar el valor real interno

Para utilizar el parámetro de entrada PV_IN, proceda del siguiente modo:

1. Seleccione la entrada "Interno" en la lista "Origen".

Error de regulación

Ajuste un ancho de zona muerta según los siguientes requisitos:

- La señal del valor real tiene ruido.
- La ganancia del regulador es alta.
- La acción D está activada.

La componente de ruido del valor real provoca, en este caso, fuertes variaciones de la variable manipulada. La zona muerta suprime la componente de ruido cuando el estado del regulador es estacionario. El ancho de zona muerta indica el tamaño de la zona muerta. Con un ancho de zona muerta de 0.0, la zona muerta se encuentra desactivada.

Consulte también

Funcionamiento de TCONT_S (Página 7121)

Configurar el algoritmo de regulación TCONT_S

General

1. Introduzca el "Tiempo de muestreo del algoritmo PID".
   El tiempo de muestreo del regulador no debería exceder un 10 % del tiempo de integración del regulador (TI) calculado.
2. Si la estructura de regulador contiene una acción P, introduzca la "Ganancia proporcional".
   Una ganancia proporcional negativa invierte el sentido de regulación.

Acción P

En caso de modificaciones de la consigna, la acción P puede provocar sobreoscilaciones. Mediante la ponderación de la acción P se puede elegir con qué intensidad debe actuar la acción P al cambiar la consigna. La atenuación de la acción P se logra mediante una compensación en la acción I.

1. Para atenuar la acción P en caso de modificaciones de la consigna, introduzca la "Ponderación de la acción P".
   - 1.0: Acción P completamente efectiva en caso de modificación de la consigna
   - 0.0: Acción P inefectiva en caso de modificación de la consigna

Acción I

1. Si la estructura de regulador contiene una acción I, introduzca el "Tiempo de integración".
   Con un tiempo de integración de 0.0, la acción I se encuentra desactivada.

Consulte también

Funcionamiento de TCONT_S (Página 7121)

Configurar la variable manipulada TCONT_S

Generador de impulsos

1. Introduzca la duración mínima de impulso y la duración mínima de pausa.
   Los valores deben ser mayores o iguales al tiempo de ciclo en el parámetro de entrada CYCLE. De esta manera, se reduce la frecuencia de maniobra.
2. Introduzca el tiempo de posicionamiento del motor.
   El valor debe ser mayor o igual al tiempo de ciclo en el parámetro de entrada CYCLE.

Consulte también

Funcionamiento de TCONT_S (Página 7121)

Poner en marcha TCONT_S

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. Introduzca los parámetros PID nuevos en los campos "P", "I", "D" y ponderación de la acción P.
3. Haga clic en el grupo "Optimización" en el símbolo "Enviar parámetros a la CPU".
4. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .
5. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
6. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabaja con los nuevos parámetros y regula según la nueva consigna.
7. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
8. Repita los pasos del 2 hasta el 6, hasta que esté satisfecho/a con el resultado de la regulación.

11.3.5 Regulación PID (S7-300, S7-400)

11.3.5.1 Utilizar control PID estándar

Utilizar PID_CP

Objeto tecnológico PID_CP

El objeto tecnológico PID_CP ofrece un regulador PID continuo para los modos automático y manual. Equivale al bloque de datos de instancia de la instrucción PID_CP. Es posible configurar los siguientes reguladores:

- Regulación de valor fijo con regulador P, PI, PD y PID continuo
- Regulación de valor fijo con control anticipativo
- Regulación en cascada
- Regulación de relación de dos lazos
- Regulación de mezcla

Se configuran las funciones siguientes del objeto tecnológico PID_CP:

- Normalización de la señal de valor real
- Supresión de oscilaciones de alta frecuencia en la rama del valor real mediante filtrado (retardando) la señal del valor real.
- Linealización de funciones cuadráticas del valor real (regulación de caudal con sensores de presión diferencial)
- Llamada de "funciones propias" en la rama del valor real
- Vigilancia de rebase por exceso de los límites de alarma y advertencia del valor real
- Vigilancia de la velocidad de cambio del valor real
- Modificación de la consigna por generador ramp soak
- Normalización de una señal de consigna externa en las regulaciones de relación
- Llamada de "funciones propias" en la rama de la consigna
- Limitación de la velocidad de cambio de la consigna
- Limitación de los valores absolutos de la consigna
- Filtrado de la señal de error de regulación por una zona muerta
- Vigilancia de rebase por exceso de los límites de alarma y advertencia del error de regulación
- Acciones P, I y D activables individualmente
- Efecto P y D opcional en la rama de realimentación del regulador.
- Control anticipativo
- Control de la variable manipulada en modo manual desde una programadora o un dispositivo HMI

- Llamada de "funciones propias" en la rama de la variable manipulada
- Limitación de la velocidad de cambio de la variable manipulada
- Limitación de los valores absolutos de la variable manipulada
- Normalización de la variable manipulada
- Modulación de ancho de impulsos para reguladores de impulsos

Cálculo de error de regulación

Valor real

Normalización (PV_NORM)

Mediante la casilla de verificación "Activar periferia" se especifica la magnitud de entrada analógica que se normaliza a la unidad física del valor real.

- Activada: la magnitud de entrada es el valor real de periferia (PV_PER)
- Desactivada: la magnitud de entrada es el valor real interno (PV_IN)

Para definir la recta de normalización de forma unívoca, defina los cuatro parámetros siguientes:

- entrada arriba (NM_PIHR)
- entrada abajo (NM_PILR)
- salida arriba (NM_PVHR)
- salida abajo (NM_PVLR)

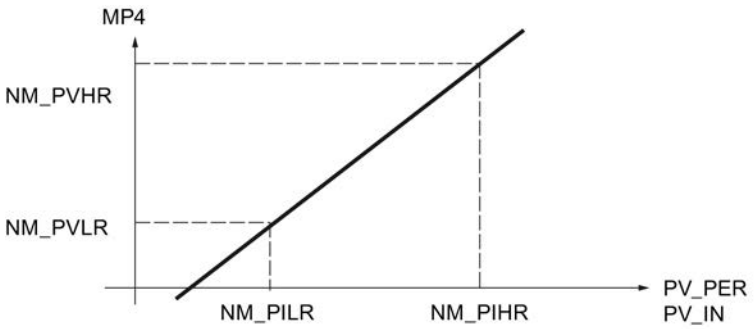
Los parámetros se prestablecen según el tipo de sensor.

El valor real se calcula con la fórmula siguiente a partir del respectivo valor de entrada PV_PER o PV_IN:

MP4 = (PV_PER - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR

O bien:

MP4 = (PV_IN - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR



El valor real se emite en la variable estática MP4.

No se limitan valores ni se comprueban parámetros. Si se introduce el mismo valor para la entrada arriba y la entrada abajo, según las fórmulas anteriores se puede producir una división por cero. Esto no se considera en la instrucción.

Funciones de filtro

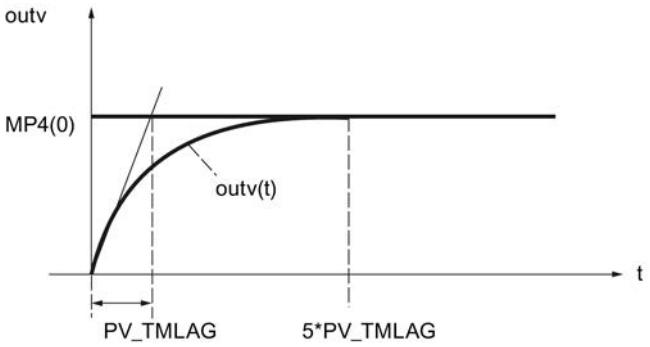
Filtrar magnitud regulada (LAG1ST)

Para filtrar señales de proceso con oscilaciones rápidas superpuestas (Noise) se utiliza un elemento de retardo de primer orden en la rama del valor real. Esta función atenúa en mayor o menor grado la magnitud regulada analógica en función de la constante de tiempo ajustada (PV_TMLAG). De ese modo se suprimen de forma efectiva las señales perturbadoras. Sin embargo, con esta medida también se aumenta en general la constante de tiempo de todo el lazo de regulación, de modo que la regulación es más lenta.

El efecto atenuador se consigue con un algoritmo de retardo de 1er orden.

La respuesta al escalón en el rango de tiempo es:

$outv(t) = MP4(0)\ (1 - e^{-t/PV_TMLAG})$



MP4(0)	la altura del escalón del valor real en la entrada
outv(t)	la magnitud de salida
PV_TMLAG	la constante de tiempo de retardo
t	Tiempo

Filtrado del valor real

Si el filtrado del valor real (LAG1STON = FALSE) no está activado, la entrada de periferia (PV_PER) o la entrada interna (PV_IN) se aplican sin retardo en la rama del valor real.

Tiempo de retardo

El parámetro de entrada PV_TMLAG determina el tiempo de retardo para el valor real. Todo el rango de valores es admisible.

Si PV_TMLAG ≤ 0.5 * CYCLE ya no actúa ningún retardo.

Se requiere un tiempo de muestreo (CYCLE) de menos de una quinta parte del tiempo de retardo para obtener un comportamiento de retardo aproximadamente equivalente al comportamiento analógico.

Calcular raíz cuadrada (SQRT)

La función "Cálculo de la raíz cuadrada" (SQRT) permite linealizar la magnitud regulada. Si el valor real proporcionado por un encóder está presente como magnitud física en relación cuadrática con la magnitud de proceso medida, el desarrollo de la magnitud regulada debe linealizarse antes de su procesamiento en el regulador.

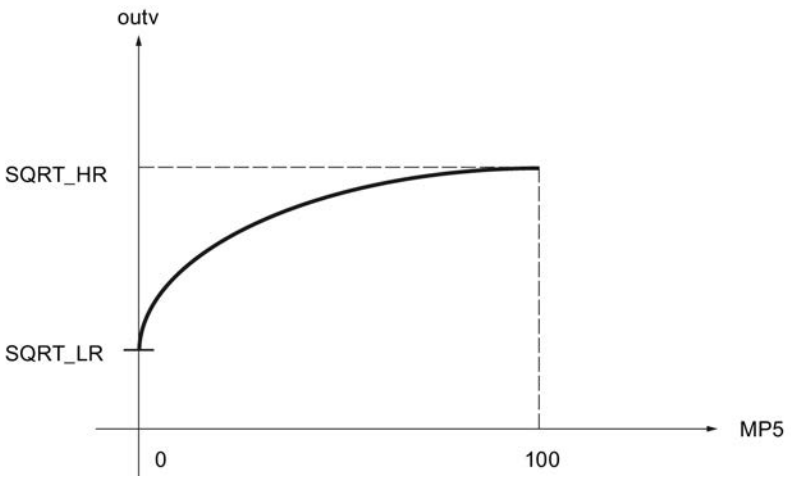
La señal de medición siempre debe linealizarse extrayendo la raíz cuadrada si se realizan mediciones de caudal con diafragmas o tubos de Venturi, etc. La presión diferencial medida (presión efectiva) es así proporcional al cuadrado del caudal.

A través de la señal de entrada SQRT_ON = TRUE se activa la función de raíz cuadrada en la rama de valor real. El algoritmo de la función de raíz cuadrada es el siguiente:

outv = SQRT (MP5) × (SQRT_HR – SQRT_LR) / 10.0 + SQRT_LR

outv es el valor de salida normalizado de la función de raíz cuadrada (caudal físico).

Esta fórmula presupone que el valor de entrada de la raíz está normalizado a un rango numérico entre 0 y 100. En la normalización del valor real deben parametrizarse el "Rango de medida superior" y el "Rango de medida inferior" a 100.0 y 0.0, respectivamente. La raíz de este valor da como resultado un rango de medida entre 0 y 10. Este rango se normaliza mediante el límite superior (SQRT_HR) y el límite inferior del rango de medida físico (SQRT_LR).



Ejemplo de normalización

El valor de entrada PV_IN del regulador es la presión diferencial en mbar:

Principio del rango de medida NM_PILR	Fin del rango de medida NM_PIHR	Ejemplo de valores de PV_IN
20,0 mbar	200,0 mbar	150,0 mbar

Con la función de normalización PV_NORM se calcula la presión diferencial normalizada, seleccionando NM_PVHR = 100.0 y NM_PVLR = 0.0:

MP4 = (PV_IN - NM_PILR) * (NM_PVHR - NM_PVLR) /
(NM_PIHR - NM_PILR) + NM_PVLR
= (PV_IN - 20.0 mbar) * (100.0 - 0.0) /
(200.0 mbar - 20.0 mbar) + 0.0
= (PV_IN - 20.0 mbar) * 100 / 180.0 mbar

Valor inicial de MP4	Valor final de MP4	Ejemplo de valores de MP4
0.0	100.0	72.222

En este ejemplo no se utiliza ningún filtrado y, por lo tanto, rige lo siguiente: MP5 = MP4.

Para la raíz de la presión diferencial normalizada MP5 resulta:

Valor inicial tras la raíz	Valor final tras la raíz	Ejemplo de valores de SQRT(MP5)
0.0	10.0	8.498

Para el valor de salida normalizado outv de la función de raíz (caudal físico) sigue con SQRT_HR = 20000.0 m^3/h und SQRT_LR = 0.0 m3/h:

outv = SQRT(MP5) * (SQRT_HR - SQRT_LR) / 10.0 + SQRT_LR
= SQRT(MP5) * (20000.0 m^3/h - 0.0 m3/h) / 10.0 + 0.0 m3/h
= 2000.0 m^3/h * SQRT(MP5)

Principio del rango de medida outv	Fin del rango de medida outv	Ejemplo de valores de outv
0,0 m^3/h	20000,0 m^3/h	16996,732 m^3/h

Llamada de FC en la rama de valor real (PVFC)

La inserción de una función específica de usuario en la rama de valor real permite someter la magnitud regulada a un tratamiento de señal antes de su procesamiento en el regulador (p. ej. un retardo de señal o linealización).

Llamar una función

Para insertar una función específica de usuario en la rama del valor real, active "Activar llamada de FC" (PVFC_ON). En el campo de entrada "Número de FC" (PVFC_NBR) se indica el número de la función utilizada. El regulador ejecuta una llamada de la función. En ella no se suministran valores a los parámetros de entrada y salida de la función de usuario. La transferencia de datos debe programarla el usuario en su función.

El valor de salida de la función puede observarse en la variable estática MP6.

Nota

No se comprueba si una función está disponible. Si la función no está disponible, la CPU pasa a STOP con un error interno de sistema.

Alarma

Vigilar los límites del valor real (PV_ALARM)

Si se rebasan por exceso o defecto determinados valores de las magnitudes del proceso (como velocidad, presión, nivel de llenado, temperatura, etc.), en las regulaciones pueden producirse estados no permitidos de proceso o instalación. En estos casos se aplica la vigilancia de los límites del valor real (PV_ALARM) para controlar si el valor real rebasa el rango de servicio admisible por exceso o por defecto. Los rebases de límite se registran y notifican a fin de activar una respuesta adecuada.

Nota

La vigilancia de los límites del valor real no puede desactivarse. El rebase por exceso de los límites configurados se notifica siempre.

La vigilancia de los límites del valor real (PV_ALARM) vigila la magnitud regulada PV(t) en función de cuatro límites predefinibles en dos zonas de tolerancia. Cuando los límites se

alcanzan o rebasan, la función notifica "Advertencia" en el primer límite y "Alarma" en el segundo.

Valores límite

Los valores numéricos de los límites se ajustan en los parámetros de entrada para "Advertencia" y "Alarma". Si el valor real (PV) rebasa estos límites por exceso o defecto, se activan los bits de aviso de salida correspondientes para la alarma superior (QPVH_ALM), la advertencia superior (QPVH_WRN), la advertencia inferior (QPVL_WRN) y la alarma inferior (QPVL_ALM).

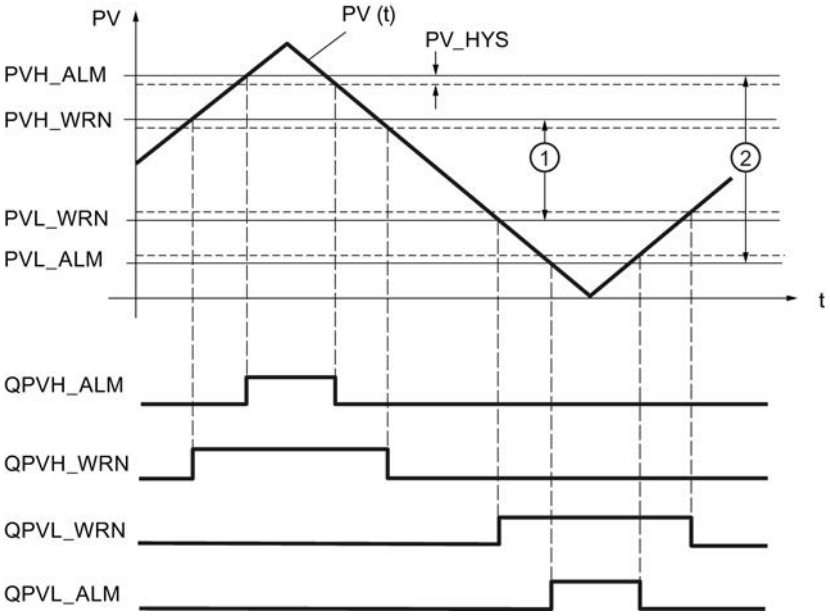
Durante el rearranque se desactivan los bits de aviso.

Los bits de aviso se activan del modo siguiente:

QPVH_ALM	QPVH_WRN	QPVL_WRN	QPVL_ALM	si:	y:
TRUE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_ALM PV ≥ PVH_ALM - PV_HYS
FALSE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_WRN PV ≥ PVH_WRN - PV_HYS
FALSE	FALSE	TRUE	FALSE	PV ↘ PV ↗	PV ≤ PVL_WRN PV ≤ PVL_WRN + PV_HYS
FALSE	FALSE	TRUE	TRUE	PV ↘ PV ↗	PV ≤ PVL_ALM PV ≤ PVL_ALM + PV_HYS

Para que el bloque funcione de manera eficiente debe aplicarse lo siguiente:

alarma superior < advertencia superior < advertencia inferior < alarma inferior



①	1. banda de tolerancia
②	2. banda de tolerancia

Histéresis

Para impedir que los bits de aviso "centelleen" con cambios muy pequeños de la magnitud de entrada o con errores de redondeo, se ajusta una histéresis (diferencia de desactivación) PV_HYS. La magnitud regulada debe superar la histéresis antes de que los bits de aviso se desactiven.

Vigilar la velocidad de cambio del valor real (ROCALARM)

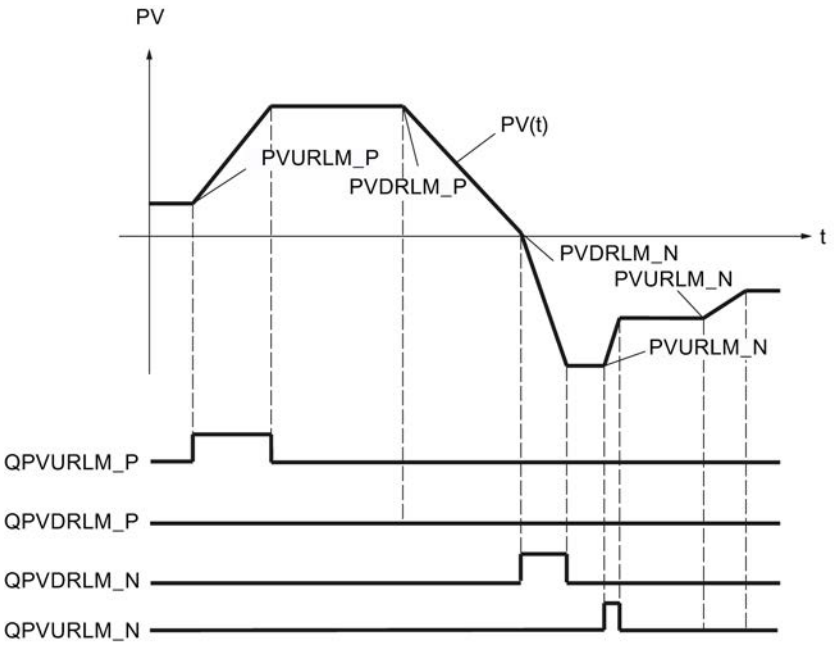
Si el cambio de la magnitud de proceso (p. ej. velocidad, presión, nivel de llenado, temperatura, etc.) es demasiado elevada, en las regulaciones pueden producirse estados no permitidos de proceso o instalación. En estos casos se aplica la vigilancia de la velocidad de cambio del valor real (ROCALARM) para controlar si el valor real rebasa por exceso o por defecto una velocidad de cambio admisible. Los rebases de límite se registran y notifican a fin de activar una respuesta adecuada.

Nota

La vigilancia de la velocidad de cambio del valor real (ROCALARM) no puede desactivarse. El rebase por exceso de los límites configurados se notifica siempre.

La función ROCALARM vigila la magnitud regulada PV(t) respecto a límites admisibles de velocidad de cambio, los cuales pueden ajustarse de forma diferente según el signo correspondiente.

Los valores numéricos de las pendientes límite se ajustan en los parámetros de entrada "Rampa de subida máxima" y "Rampa de bajada máxima" en el rango positivo y negativo de la magnitud regulada. Las pendientes se refieren a la subida o bajada en porcentaje por segundo. Si la velocidad de cambio de la magnitud regulada supera estos límites, se activan los bits de aviso de salida correspondientes QPVURLMP ... QPVDRLMN.



La identificación de los parámetros de las rampas se realiza según el siguiente esquema:

Parámetro	Cambio de PV
Rampa de subida máxima en el rango positivo (PVURLM_P)	PV > 0 y |PV| ascendente
Rampa de bajada máxima en el rango positivo (PVDRLM_P)	PV > 0 y |PV| descendente
Rampa de subida máxima en el rango negativo (PVURLM_N)	PV < 0 y |PV| ascendente
Rampa de bajada máxima en el rango negativo (PVDRLM_N)	PV < 0 y |PV| descendente

Consigna

Definir la consigna

Seleccione un origen para la consigna. Las posibilidades son las siguientes:

- Consigna interna
  El parámetro de entrada SP_INT se utiliza como consigna
- Generador de consigna
  El parámetro de entrada SP_INT se utiliza como consigna y puede aumentarse o disminuirse mediante las variables estáticas SPUP y SPDN.

- Generador ramp soak
  El parámetro de entrada SP_INT se utiliza como consigna y puede modificarse siguiendo un perfil predefinido.
- Consigna externa
  En los reguladores de mezcla, relación y cascada se utiliza el parámetro de entrada SP_EXT como consigna. La consigna externa puede normalizarse.

En el parámetro de salida SP se muestra la consigna efectiva.

Generador de consigna (SP_GEN)

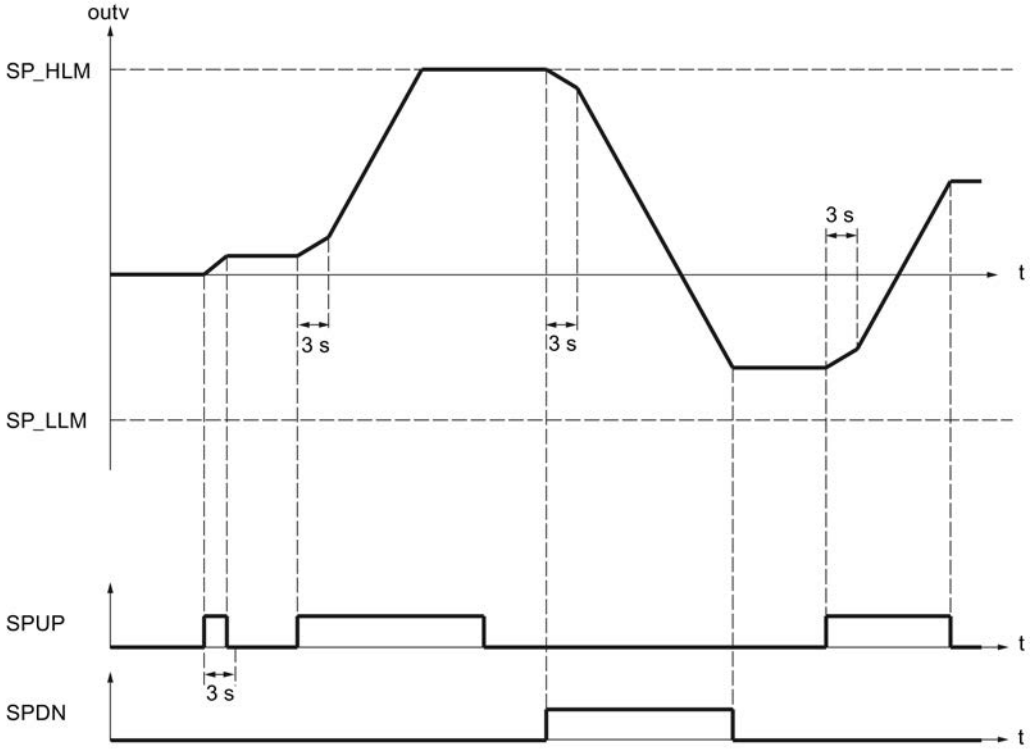
La consigna interna SP_INT se aumenta o disminuye progresivamente utilizando los interruptores SPUP y SPDN. El valor modificado puede observarse en MP1. Se tienen en cuenta los límites configurados de la consigna.

Repercusión de los interruptores SPUP y SPDN

Para poder realizar un cambio graduado finamente, el regulador debe tener un tiempo de muestreo de 100 ms como máximo.

La velocidad de cambio de la magnitud de salida depende de la duración de pulsación de los interruptores SPUP y SPDN así como de los límites ajustados de acuerdo con la relación siguiente:

- durante los primeros 3 segundos después de activar SPUP o SPDN rige:
  doutv/dt = (SP_HLM - SP_LLM) / 100 s
- a continuación
  doutv/dt = (SP_HLM - SP_LLM) / 10 s



Con un tiempo de muestreo de 100 ms y un rango de consigna de -100,0 a 100,0, la consigna cambia en 0,2 por ciclo en los tres primeros segundos, por ejemplo. Si SPUP permanece en TRUE durante más de 3 segundos, la velocidad de cambio aumenta a diez veces el valor, en este caso a 2 por ciclo.

Arranque y funcionamiento del generador de consigna

- Durante el rearranque la salida outv se pone a 0.0.
- Al activar el generador de consigna (SPGEN_ON = TRUE), en la salida outv se muestra primero el valor de señal SPFC_IN. Por este motivo, la conmutación al generador de consigna desde otro modo de operación se produce siempre suavemente. Mientras los interruptores SPUP y SPDN (teclas arriba y abajo) no se activan, SPFC_IN permanece en la salida.

Parámetros de la función SP_GEN

El parámetro de salida outv es un parámetro implícito. Este puede observarse en la herramienta de configuración mediante el punto de medición MP1.

Parámetro	Significado	Rango admisible
SPFC_IN	Consigna de la entrada de FC	Rango técnico
SP_INT	Consigna interna	Rango técnico

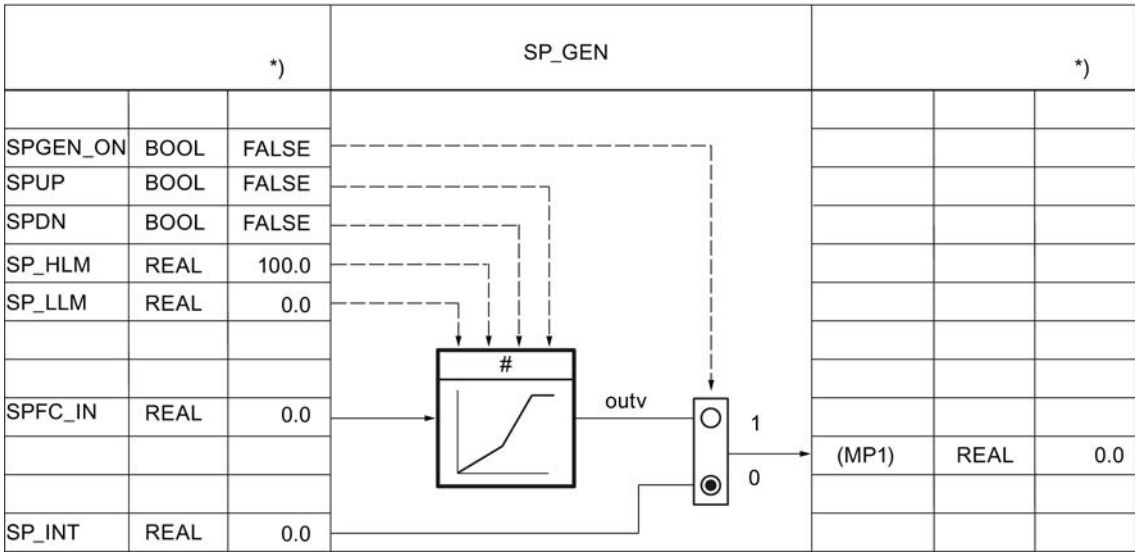


Figura 11-2 Esquema funcional y parámetros del generador de consigna

Generador ramp soak (RMP_SOAK)

Si la consigna SP_INT debe cambiar por sí misma en función del tiempo, p. ej. en la regulación de procesos conforme a un programa de temperatura controlado por tiempo, es posible

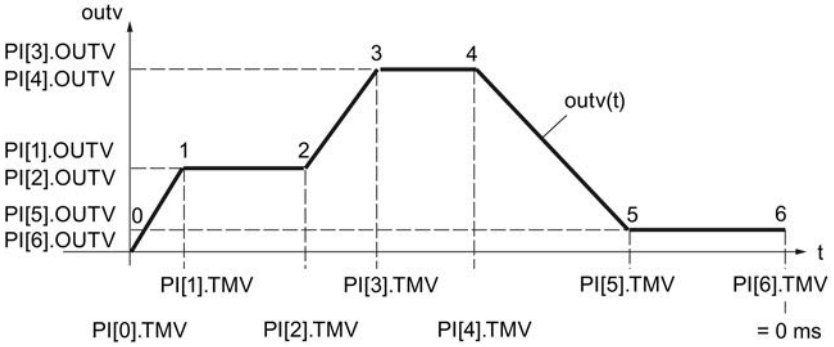
hacerlo configurando un perfil y activando el generador ramp soak. El cambio de la consigna se calcula a partir de segmentos con un máximo de 256 nodos.

Nota

La función del bloque no comprueba si existe un DB global con el número "DB_NBR" ni tampoco si el parámetro "Número de nodos" NBR_PTS coincide con la longitud del DB. Si la parametrización es incorrecta, la CPU cambia al estado **STOP** con un error de sistema interno.

Funcionamiento del generador ramp soak

El generador ramp soak proporciona una función de la magnitud de salida outv que se ejecuta de acuerdo con una programación determinada. El desarrollo del perfil se define con una serie de nodos de interpolación en un bloque de datos global RMP_SOAK con los valores de tiempo PI[i].TMV y los correspondientes valores de salida PI[i].OUTV. La figura siguiente muestra un perfil con 6 nodos.

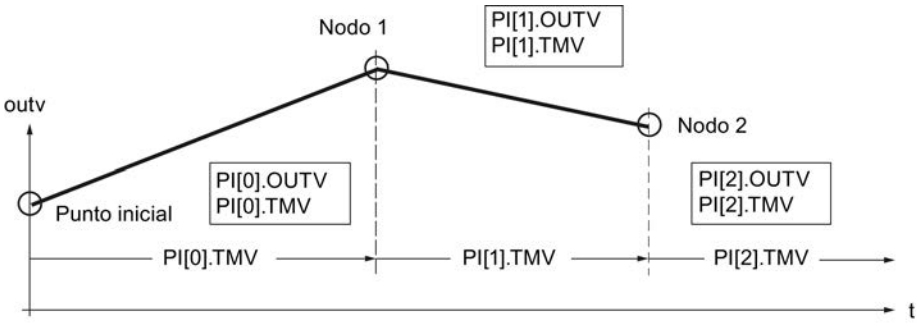


Para el último nodo n, el valor de tiempo PI[n].TMV = 0 ms. El tiempo de procesamiento de un perfil se calcula partiendo del valor inicial y descendiendo hasta cero.

PI[i].TMV indica los intervalos entre nodo y nodo. Entre los nodos se interpola de acuerdo con la función siguiente:

$$outv(t) = PI[n + 1].\ OUTV - \frac{RS_TM}{PI[n].TMV}\ (PI[n + 1].OUTV - PI[n].OUTV)$$

$0 \leq n \leq (NBR_PTS - 1)$



Nota

Nota

Al interpolar la curva de evolución entre los nodos, es posible que el valor de salida se mantenga temporalmente fijo si resulta que el tiempo de muestreo CYCLE es muy pequeño con respecto al tiempo entre los nodos PI[n].TMV. El generador ramp soak no puede linealizar curvas planas debido a la precisión de cálculo de la CPU. Si la curva es demasiado plana, el valor de salida permanece por un tiempo en el nodo correspondiente y, pasado cierto tiempo, se integra en el próximo nodo con la pendiente mínima.

Solución: reduzca el tiempo entre los nodos insertando más nodos. De este modo, la curva emitida se acercará a la curva plana deseada en forma de trapecio.

Activar el generador ramp soak

Con el cambio RMPSK_ON de FALSE a TRUE se activa el generador ramp soak. Una vez alcanzado al último nodo se termina el perfil. Cuando el operador reinicia la función, es necesario poner primero RMPSK_ON a 'FALSE' y a continuación a 'TRUE'.

En el **rearranque**, la salida outv se pone a 0.0 y se determina el tiempo total o el tiempo residual total. En la transición al modo normal, la curva de evolución se procesa inmediatamente desde el punto inicial de acuerdo con el modo de operación ajustado. Si no se desea que esto suceda, el parámetro RMPSK_ON debe ponerse a FALSE en el OB de rearranque.

Funciones del generador ramp soak

La instrucción dispone de las funciones siguientes:

- Activar generador ramp soak para ejecución única
- Predeterminar salida con valor fijo
- Activar repetición cíclica
- Detener generador ramp soak
- Continuar generador ramp soak
- Actualizar tiempo total de procesamiento y tiempo sobrante total

Para ajustar el modo de operación deseado se aplica la siguiente tabla para la valencia de las entradas de control:

Modo de operación	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Señal de salida OUTV
Activar generador ramp soak	TRUE	FALSE	FALSE	**)	FALSE	**)	OUTV(t) El valor final se mantiene tras el final de procesamiento.
Especificar salida	TRUE	TRUE	**)	**)	**)	**)	DF_OUTV
Activar repetición cíclica	TRUE	FALSE	FALSE	**)	TRUE	**)	OUTV(t) Tras final: inicio automático

Modo de operación	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Señal de salida OUTV
Detener generador ramp soak	TRUE	FALSE	TRUE	FALSE	**)	**)	se mantiene el valor actual de OUTV(t) *)
Continuar generador ramp soak	TRUE	FALSE	TRUE	TRUE	**)	**)	OUTV (antiguo) *)
			FALSE				Se sigue con los nuevos valores predeterminados.
Actualizar el tiempo total y el tiempo residual total	**)	**)	**)	**)	**)	FALSE	No influye en OUTV
						TRUE	No influye en OUTV
*) Hasta el próximo nodo la curva no tiene la forma parametrizada por el usuario. **) El modo de operación ajustado en cada caso se ejecuta independientemente de la valencia de las señales de control en los campos marcados con **)							

Especificar la salida del generador ramp soak

Si se activa esta función (DFRMP_ON = TRUE), el valor de salida del generador ramp soak se pone al valor de señal SP_INT o al valor de salida SP_GEN.

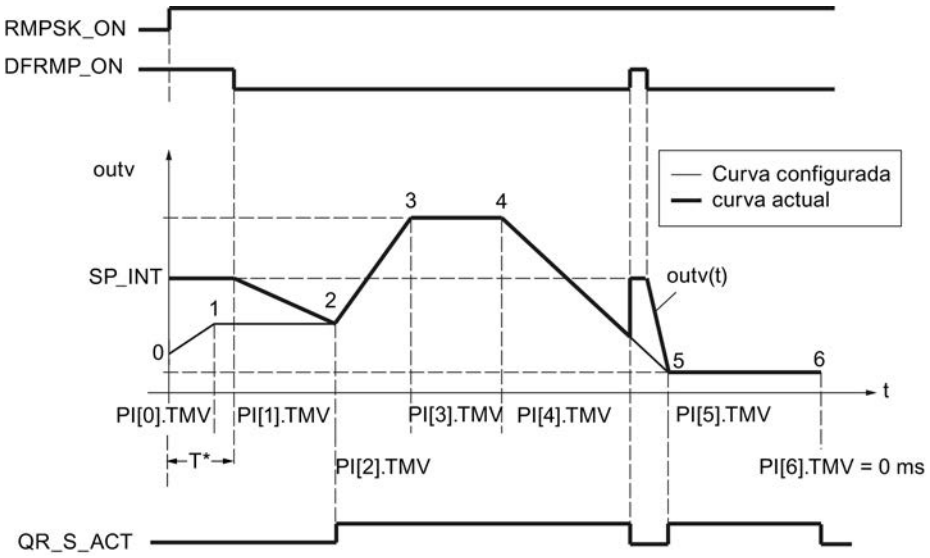
Nota

La asignación previa con la consigna interna solo tiene efecto si el generador ramp soak está activado (RMPSK_ON = TRUE).

Tras la conmutación DFRMP_ON = FALSE, outv se lleva linealmente de la consigna ajustada (p. ej. SP_INT) al valor de salida del número de nodo actual PI[NBR_ATMS].OUTV.

El procesamiento interno de tiempo también continúa con una consigna fija aplicada directamente (RMPSK_ON = TRUE y DFRMP_ON = TRUE).

Al iniciar la curva de evolución RMPSK_ON = TRUE se emite la consigna fija SP_INT hasta que DFRMP_ON cambia de TRUE a FALSE después del tiempo T*.



En este momento ha transcurrido el tiempo PI[0].TMV y parte del tiempo PI[1].TMV. OUTV se lleva de SP_INT a PI[2].OUTV, es decir, a la consigna del nodo 2.

La curva configurada se emite solo a partir del nodo 2. La coincidencia entre la curva configurada y la actual se indica con QR_S_ACT = TRUE. Con un flanco ascendente en DFRMP_ON durante el procesamiento de la curva, el valor emitido OUTV salta a SP_INT sin retardo.

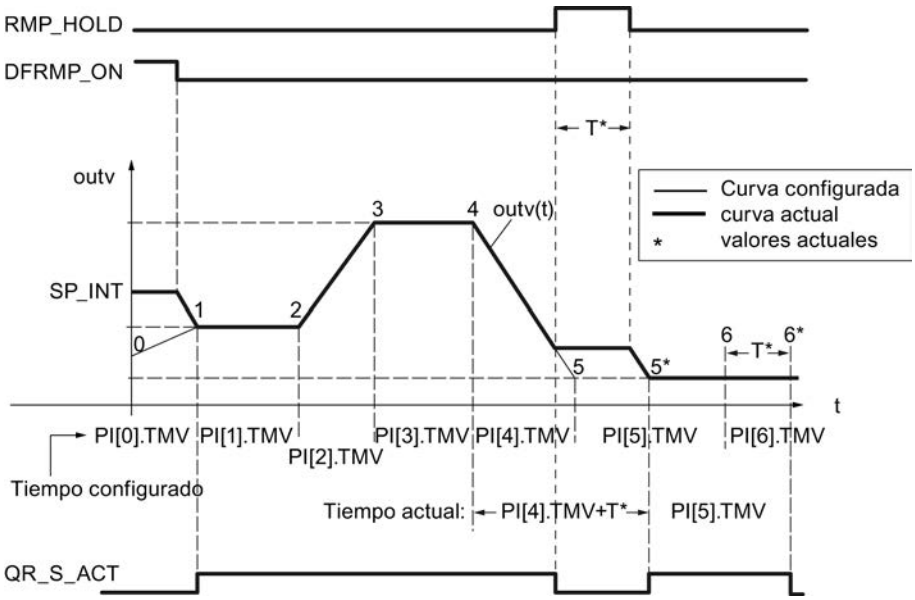
Activar repetición cíclica

Cuando el modo de operación 'Repetición cíclica' (CYC_ON = TRUE) está activado, el generador ramp soak vuelve automáticamente al punto inicial tras emitir el último valor de nodo y se ejecuta de nuevo.

Entre el último nodo y el punto inicial no se interpola. Para una transición suave se debe aplicar lo siguiente: PI[NBR_PTS].OUTV = PI[0].OUTV.

Detener generador ramp soak

Con RMP_HOLD = TRUE se congela el valor de la magnitud de salida (incluido el procesamiento de tiempo). Cuando RMP_HOLD = FALSE se desactiva, PI[x].TMV continúa en el punto de interrupción.



El tiempo de procesamiento de la curva se prolonga por el tiempo T*. La curva presenta el desarrollo configurado desde el nodo hasta el flanco ascendente en RMP_HOLD y desde el nodo 5* hasta el nodo 6*, es decir, la señal de salida QR_S_ACT tiene el valor TRUE.

Si el interruptor CONT_ON está activado, el generador ramp soak detenido continúa en el punto predeterminado TM_CONT con el procesamiento de la curva de desplazamiento.

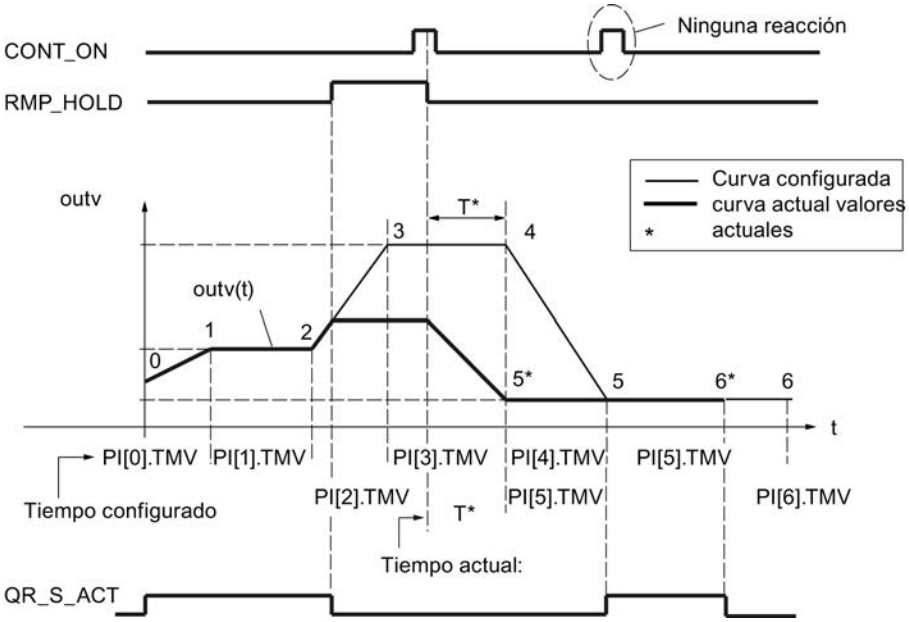
Continuar generador ramp soak

Con CONT_ON se continúa una curva detenida en un punto definido. Si el generador ramp soak no está detenido, CONT_ON no tiene efecto.

Si la entrada de control para continuar CONT_ON = TRUE está activada, el generador ramp soak detenido continúa en el nodo TM_SNBR. TM_CONT determina el tiempo residual que requiere el generador hasta que se alcanza el nodo TM_SNBR.

La figura siguiente muestra la influencia de RMP_HOLD y CONT_ON con la configuración siguiente:

Continuar en el nodo 5 (TM_SNBR = 5) y continuar en T* s: TM_CONT = T*



Los nodos 3 y 4 se suprimen para este ciclo de procesamiento del generador ramp soak. Después del cambio de señal RMP_HOLD de TRUE a FALSE, el desarrollo configurado de la curva solo se vuelve a alcanzar a partir del nodo 5.

La salida QR_S_ACT solo se activa cuando el generador ramp soak procesa la curva de evolución parametrizada por el usuario.

Actualizar el tiempo total y el tiempo residual total

En cada ciclo se actualizan el número actual del nodo NBR_ATMS, el tiempo residual actual hasta alcanzar el nodo RS_TM, el tiempo total T_TM y el tiempo residual total hasta alcanzar el final de la curva RT_TM.

En caso de modificaciones online de PI[n].TMV, cambian el tiempo total y el tiempo residual total de la curva de evolución. Ya que el cálculo de T_TM y RT_TM prolonga considerablemente el tiempo de procesamiento del bloque de función en muchos nodos, este solo se realiza después del rearranque o si TUPDT_ON = TRUE. Los intervalos PI[0 ... NBR_PTS].TMV entre los distintos nodos se suman y se indican en la salida del tiempo total T_TM y del tiempo residual total RT_TM.

Tenga en cuenta que el cálculo de los tiempos totales requiere mucho tiempo de ejecución de la CPU.

Procedimiento

Para configurar el generador ramp soak, proceda del siguiente modo:

1. En la lista "Origen de la consigna" seleccione la entrada "Generador ramp soak".
2. Cree un bloque de datos global RMP_SOAK.
3. Defina los parámetros necesarios para el perfil.
4. Introduzca el número del bloque de datos RMP_SOAK en el campo de entrada "Número de DB".
5. Active o desactive las demás funciones del generador ramp soak.
6. Si ha activado la función "Continuar generador ramp soak", introduzca el nodo y el tiempo en que deba continuar el generador.

Consulte también

Bloque de datos RMP_SOAK (Página 7197)

Normalización

Normalización de una consigna externa

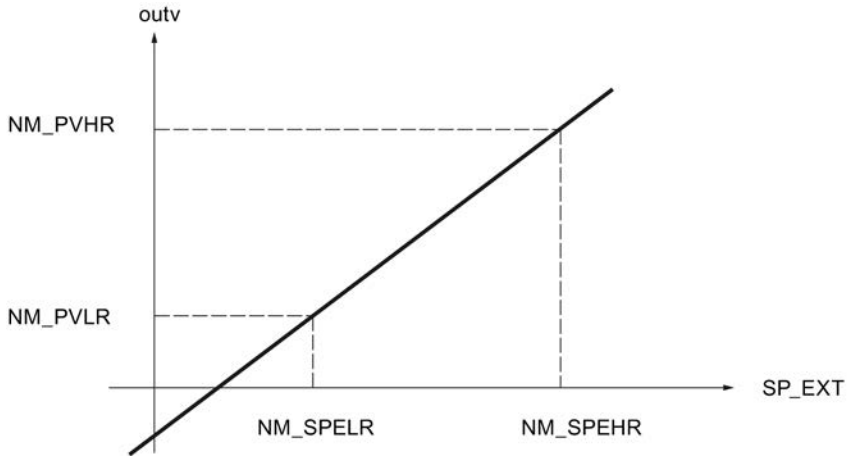
Si la consigna externa no está presente en la unidad física del valor real (p. ej. en % en una cascada de regulador), es necesario normalizar ese valor y su rango de ajuste a la unidad física del valor real. Para ello se utiliza la función "Normalización en la rama de consigna".

La función SP_NORM normaliza una magnitud de entrada analógica. La consigna externa analógica se convierte a la magnitud de salida outv con la recta de normalización. El valor de salida OUTV está accesible en la herramienta de configuración a través del punto de medición MP2.

El valor de salida de la función es efectivo cuando la entrada de control es SPEXT_ON = TRUE. Para definir la recta de normalización de forma unívoca, defina los cuatro parámetros siguientes:

- Entrada arriba: SP_EXT: NM_SPEHR
- Entrada abajo: SP_EXT: NM_SPELR

- Salida arriba: NM_PVHR (este valor se introduce en la función de normalización del valor real.)
- Salida abajo: NM_PVLR (este valor se introduce en la función de normalización del valor real.)



El valor de salida outv se calcula con la fórmula siguiente a partir del correspondiente valor de entrada SP_EXT:

outv = (SP_EXT – NM_SPELR) x (NM_PVHR – NM_PVLR) / (NM_SPEHR – NM_SPELR) + NM_PVLR

Para el caso especial de una función de raíz aplicada en la rama de valor real rigen como límite superior e inferior del valor de salida outv los valores de normalización de la función de raíz (SQRT_HR y SQRT_LR).

En ese caso, el valor de salida outv de la función de normalización se calcula con la fórmula siguiente a partir del correspondiente valor de entrada SP_EXT:

outv = (SP_EXT – NM_SPELR) x (SQRT_HR – SQRT_LR) / (NM_SPEHR – NM_SPELR) + SQRT_LR

Dentro de la función no se limitan los valores ni se comprueban los parámetros. Si introduce el mismo valor para NM_SPEHR y NM_SPELR, según las fórmulas anteriores se puede producir una división entre cero. Esto no se considera en la instrucción.

Factor de mezcla/relación

En una regulación de relación, las consignas de los lazos de regulación se adaptan unas a otras de forma que la relación ajustada se corresponde con la relación efectiva de los valores reales.

En una regulación de mezcla se ajusta la proporción deseada de cada uno de los componentes. La suma de los factores de mezcla FAC debe dar 1.

Procedimiento

Para configurar la normalización, proceda del siguiente modo:

1. Introduzca los valores de la normalización en los campos de entrada "Entrada arriba" y "Entrada abajo".
2. Introduzca un factor de mezcla/relación (FAC).

Límites

Llamada de FC en la rama de consigna

La inserción de una función específica de usuario en la rama de consigna permite someter una referencia predeterminada de forma externa a un procesamiento de señal (p. ej. un retardo de señal o linealización) antes de aplicarla en el regulador.

Si se activa la llamada de FC en la rama de consigna, se llama una función específica de usuario. El regulador ejecuta una llamada de la FC. En ella no se suministran valores a los parámetros de entrada y salida de la función de usuario ya existentes. Por ello, la transmisión de datos debe programarse manualmente con S7-AWL en la FC de usuario. La consigna acondicionada se observa en MP3.

En ese caso, el valor de SPFC_ON determina si en el canal de la consigna se inserta en este punto una función programada libremente en forma de FC estándar (p. ej. una curva característica) o si la consigna continúa procesándose sin ninguna influencia de ese tipo.

Nota

La instrucción no comprueba si hay una función. Si la función no está disponible, la CPU pasa a STOP con un error interno de sistema.

Procedimiento

Para llamar una función en la rama de la consigna, proceda del siguiente modo:

1. Active la casilla de verificación "Activar llamada de FC".
2. En el campo de entrada "FC", indique el número de la función que se va a llamar.

Limitación de la velocidad de cambio de la consigna

Las funciones de rampa en la rama de la consigna se utilizan en aquellos casos en que el proceso no soporta cambios bruscos de la señal manipulada, ya que un cambio abrupto de la consigna generalmente provoca también un escalón en la variable manipulada del regulador. Estos cambios abruptos de la variable manipulada deben evitarse p. ej. en aquellos casos en que entre el motor regulado y la carga que debe accionarse hay interpuesto un engranaje, y una elevación muy rápida del número de revoluciones del motor derivaría en una sobrecarga del engranaje.

La función SP_ROC limita la velocidad de cambio de las consignas procesadas en el regulador, ya sea por separado para la rampa de subida y bajada, como por separado para los rangos positivo y negativo de la referencia.

En las cuatro entradas SPURLM_P, SPDRLM_P, SPURLM_N y SPDRLM_N se introducen las limitaciones correspondiente para los aumentos de la función de rampa en los rangos positivo y negativo de la referencia. Las rampas de subida se refieren a la subida o bajada por segundo. Los cambios de consigna más rápidos se retardan a estas velocidades límite.

Si p. ej. SPURLM_P se parametriza a 10.0 [rango de valores tec./s], en cada ciclo de muestreo se suman los siguientes valores al "valor antiguo" de outv mientras inv sea > outv:

Tiempo de muestreo	1 s	$outv_{antiguo}$ + 10
	100 ms	$outv_{antiguo}$ + 1
	10 ms	$outv_{antiguo}$ + 0.1

La siguiente figura muestra el funcionamiento del procesamiento de la señal a partir de un ejemplo. Las funciones de escalón en la entrada inv(t) se traducen en funciones de rampa en la salida outv(t).

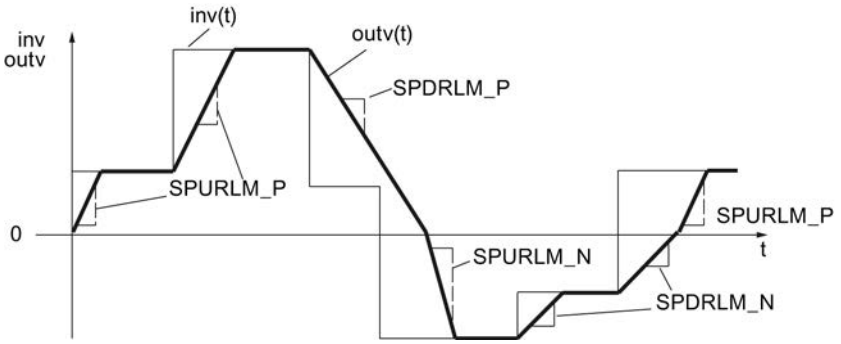


Figura 11-3 Limitación de la velocidad de cambio de la referencia SP(t)

No se notifica el alcance de los límites de la pendiente.

Procedimiento

Para limitar la velocidad de cambio, proceda del siguiente modo:

1. Active la casilla de verificación "Limitar la velocidad de cambio".
2. Defina los siguientes valores tanto para el rango positivo como para el negativo:
   - Rampa de subida de consigna
   - Rampa de bajada de consigna

Limitación del valor absoluto de la referencia

El rango de ajuste de la referencia determina el rango dentro del cual puede oscilar la referencia, es decir, el margen en el que se mueve el proceso dentro de los valores de estado admisibles.

Por ello, para evitar estados de proceso críticos o no permitidos, Standard PID Control limita el rango de ajuste de la referencia en la rama de consigna por arriba y por abajo (SP_LIMIT).

La función SP_LIMIT limita la consigna SP al valor inferior y superior predeterminable SP_LLM y SP_HLM mientras la magnitud de entrada INV esté fuera de dichas limitaciones. Dado que

la función no puede desactivarse, en la configuración siempre hay que tener en cuenta la indicación de los límites inferior y superior.

Los valores numéricos de los límites se ajustan en los parámetros de entrada para la limitación inferior y superior. Si la magnitud de entrada inv(t) rebasa los límites, se emiten las indicaciones correspondientes a través de las salidas de aviso.

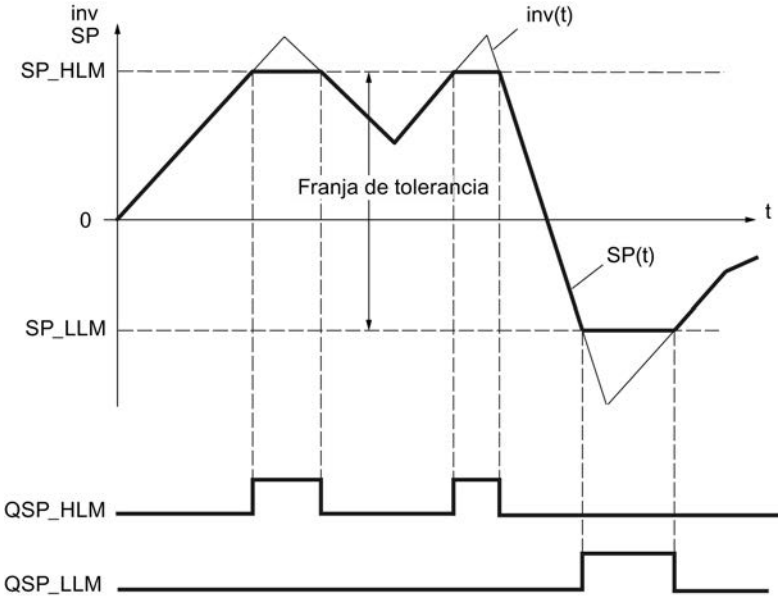


Figura 11-4 Limitación de valor absoluto de la referencia SP(t)

Durante el rearranque se desactivan los bits de aviso.

Los bits de aviso se activan del modo siguiente:

SP =PVH_ALM	QSP_HLM	QSP_LLM	si:
SP_HLM	TRUE	FALSE	inv ≥ SP_HLM
SP_LLM	FALSE	TRUE	inv ≤ SP_LLM
inv	FALSE	FALSE	SP_HLM < inv < SP_LLM

Procedimiento

Para limitar el valor absoluto de la referencia, proceda del siguiente modo:

1. En el campo de entrada "Límite superior", indique el límite superior absoluto de la consigna (SP_HLM).
2. En el campo de entrada "Límite inferior", indique el límite inferior absoluto de la consigna (SP_LLM).

Error de regulación

Zona muerta

En caso de superposición de la magnitud regulada o la referencia con una señal perturbadora de mayor frecuencia, en un regulador con ajuste óptimo el ruido también es efectivo en la salida del regulador. En una amplificación elevada del regulador y con derivación activada, por ejemplo, esto puede provocar importantes oscilaciones en la magnitud manipulada. Debido al aumento de la frecuencia de conmutación (regulador paso a paso), eso deriva en un desgaste elevado del actuador.

En estado forzado del regulador, la función reduce el ruido en la señal del error de regulación, disminuyendo así la oscilación no deseada de la salida de regulador.

La zona muerta puede desactivarse.

Filtrado de señal por función de zona muerta

La función DEADBAND suprime pequeñas oscilaciones de la magnitud de entrada dentro de un rango parametrizable alrededor de un punto cero definido. Fuera de esta franja de oscilaciones, el error de regulación ER aumenta o cae proporcionalmente a la magnitud de entrada. El ancho de la zona no efectiva (zona muerta) puede definirse con el parámetro DEADB_W. El ancho de zona muerta solo debe tener valores positivos.

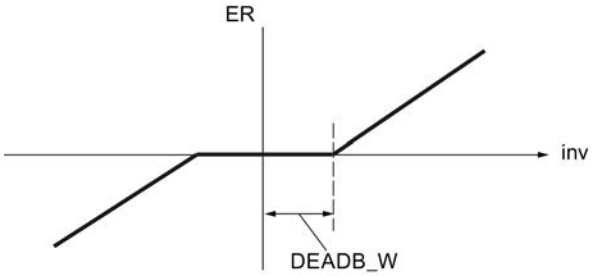
Si la magnitud de entrada se encuentra dentro de la zona muerta, se deposita el valor 0 (error de regulación = 0) en la salida. Solo una vez que la magnitud de entrada abandona este rango de insensibilidad, la salida aumenta o cae la parte correspondiente a los mismos valores de la magnitud de entrada inv. El resultado es una falsificación de la señal transmitida, también fuera de la zona muerta. Sin embargo, esto se tolera para evitar escalones en los límites de la zona muerta (véase la siguiente figura). La falsificación corresponde al valor DEADB_W, por lo que es fácilmente controlable.

La función DEADBAND trabaja según las siguientes funciones:

(ER) = inv + DEADB_W para inv < - DEADB_W

(ER) = 0 para -DEADB_W ≤ inv ≤ + DEADB_W

(ER) = inv - DEADB_W para inv > + DEADB_W



Las repercusiones del filtrado de señales se observan en el parámetro de salida ER.

Alarma

Las desviaciones excesivas de la magnitud regulada respecto a la consigna ajustada pueden provocar estados de proceso no deseados. En estos casos se utiliza la función ER_ALARM para vigilar que el error de regulación no rebase el rango de servicio permitido por exceso ni por defecto. ER_ALARM registra y notifica los rebases de límite ocurridos, para permitir una respuesta adecuada.

Nota

La vigilancia de los límites del error de regulación no puede desactivarse. El rebase por exceso de los límites configurados se notifica siempre.

La función ER_ALARM vigila el valor del error de regulación ER(t) en función de cuatro límites predefinibles en dos zonas de tolerancia. Cuando los límites se alcanzan o rebasan, la función notifica primero "Advertencia" y a continuación "Alarma" en el segundo límite.

Valores límite

Los valores numéricos de los límites se ajustan en los parámetros de entrada para "Advertencia" y "Alarma". Si el error de regulación (ER) rebasa estos límites por exceso o defecto, se activan los bits de aviso de salida correspondientes QERN_ALM ... QERP_ALM.

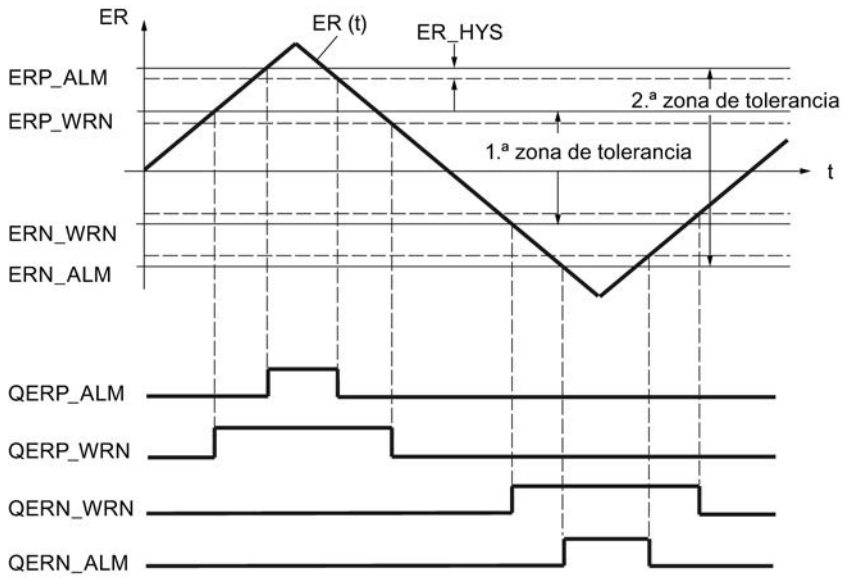
Durante el rearranque se desactivan los bits de aviso.

Los bits de aviso se activan del modo siguiente:

QERP_ALM	QERP_WRN	QERN_WRN	QERN_ALM	si:	y:
TRUE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_ALM ER ≥ ERP_ALM - ER_HYS
FALSE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_WRN ER ≥ ERP_WRN - ER_HYS
FALSE	FALSE	TRUE	FALSE	ER ↘ ER ↗	ER ≤ ERP_WRN ER ≤ ERP_WRN + ER_HYS
FALSE	FALSE	TRUE	TRUE	ER ↘ ER ↗	ER ≤ ERP_ALM ER ≤ ERP_ALM + ER_HYS

Para que el bloque funcione de manera eficiente debe aplicarse lo siguiente:

alarma superior < advertencia superior < advertencia inferior < alarma inferior



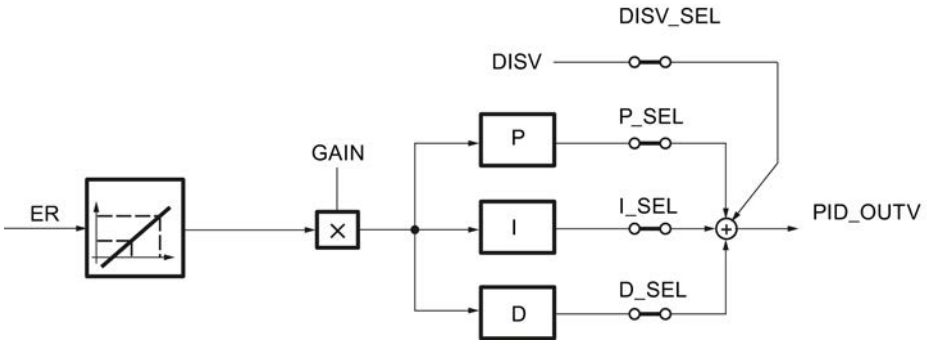
Histéresis

Para evitar que los bits de aviso "centelleen" con cambios muy pequeños de la magnitud de entrada o con errores de redondeo, se ajusta una histéresis (diferencia de desactivación) ER_HYS. El error de regulación debe superar esta histéresis antes de que los bits de aviso se desactiven.

Regulador

Algoritmo de regulación

En el ciclo del tiempo de muestreo configurado se calcula la magnitud manipulada del regulador continuo a partir del error de regulación en el algoritmo de posición PID. El regulador tiene una estructura puramente paralela. Las acciones proporcional, integral o diferencial se pueden desconectar individualmente.

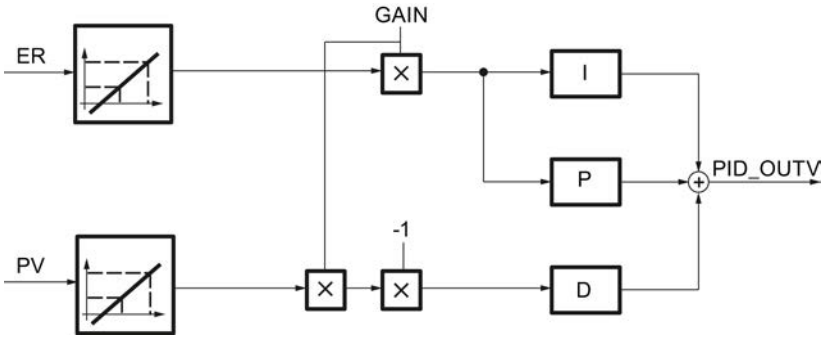


Control anticipativo

A la señal de salida del regulador PID_OUTV se le puede aplicar adicionalmente una magnitud perturbadora DISV. La activación o desactivación se realiza en la ventana PID de la herramienta de configuración a través del conmutador de estructura DISV_SEL o mediante "Magnitud perturbadora ON".

Acción P y D en la rama de realimentación

En la estructura paralela, cada acción del algoritmo regulador recibe el error de regulación como señal de entrada. La magnitud manipulada es modificada directamente por la acción P y la acción D con escalones de la consigna. Sin embargo, otra estructura del regulador en la que la formación de la acción P y la acción D se traslada a la realimentación, garantiza una evolución suave de la magnitud manipulada en caso de variaciones del escalón de la referencia. En esta estructura la acción I procesa el error de regulación como señal de entrada, en la acción P y en la acción D solo se conecta la magnitud de regulación negativa (factor = -1).



Normalización de las magnitudes de entrada ER y PV

Las magnitudes de entrada ER y PV del regulador PID se normalizan antes del procesamiento del regulador en el rango de 0 a 100, conforme a las fórmulas siguientes:

- con función de raíz desactivada (SQRT_ON = FALSE):
  - $ER_{normalizado}$ = ER * 100.0 / (NM_PVHR – NM_PVLR)
  - $PV_{normalizado}$ = (PV - NM_PVLR) * 100.0 / (NM_PVHR – NM_PVLR)
- con función de raíz activada (SQRT_ON = TRUE):
  - $ER_{normalizado}$ = ER * 100.0 / (SQRT_HR – SQRT_LR)
  - $PV_{normalizado}$ = (PV - SQRT_LR) * 100.0 / (SQRT_HR – SQRT_LR)

La normalización se lleva a cabo para que la ganancia proporcional GAIN del regulador PID pueda introducirse sin dimensiones. De este modo, si cambia el límite superior e inferior del rango de medida físico (p. ej. de bar a mbar), no es necesario cambiar el factor de ganancia.

Las magnitudes de entrada normalizadas $ER_{normalizado}$ y $PV_{normalizado}$ no pueden observarse.

Consulte también

Estructuras de regulador (Página 10090)

Acción P (Página 10093)

Acción D (Página 10096)

Acción I (Página 10094)

Estructuras de regulador

Uso y parametrización del regulador PID

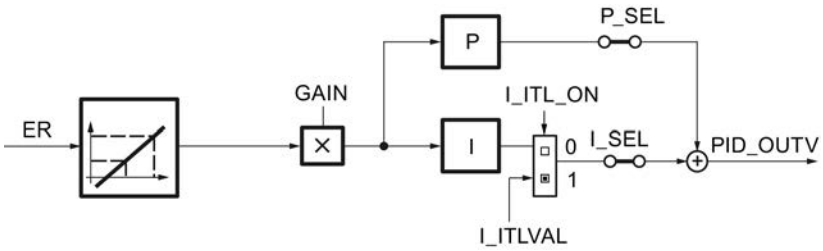
La mayor parte de los procesos regulados que se dan en la industria del procesamiento y los procesos puede regularse con la función de regulador PI/PID de Standard PID Control. Solo en casos especiales se requieren métodos y medidas adicionales para dominar la regulación en cuestión.

No obstante, un gran problema práctico suele ser la parametrización del regulador PI/PID, es decir, encontrar los valores de ajuste "correctos" para los parámetros del regulador. La calidad de esta parametrización es determinante para la efectividad de la regulación PID, y requiere una gran experiencia práctica, conocimientos especializados o bien mucho tiempo.

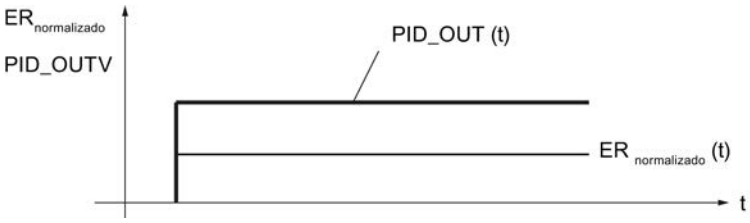
Regulación P

En el regulador P están desconectadas las acciones I y D. (I_SEL y D_SEL = FALSE). Eso significa que con un error de regulación ER = 0, la señal de salida también es OUTV = 0. Para ajustar un punto de trabajo ≠ 0, es decir, un valor numérico para la señal de salida con un error de regulación cero, hay que activar la rama I (véase la figura siguiente).

En la acción I se puede predeterminar un punto de trabajo ≠ 0 para el regulador P ajustando el valor de inicialización I_ITLVAL en correspondencia. Para ello hay que activar los conmutadores 'I_ITL_ON' y 'I_SEL' = TRUE.



La respuesta indicial del regulador P en el rango de tiempo es:
PID_OUTV (t) = I_ITLVAL + GAIN * ER $_{normalizado}$ (t)



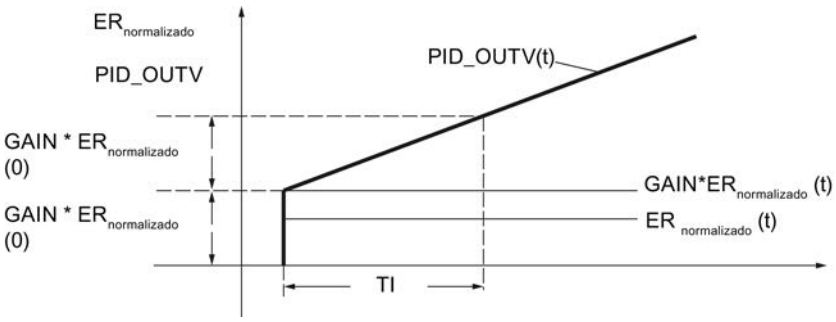
PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
I_ITLVAL	es el punto de trabajo del regulador P
GAIN	es la ganancia de regulador
$ER_{normalizado}(t)$	es el error de regulación normalizado

Regulación PI

En el regulador PI está desconectada la acción D. (D_SEL = FALSE). Un regulador PI modifica la magnitud de salida PID_OUTV con la acción I hasta que el error de regulación es ER = 0. Sin embargo, esto solo es aplicable cuando la magnitud de salida no rebasa los límites del rango manipulado.

La respuesta indicial en el rango de tiempo es:

$$PID_OUTV(t) = GAIN * ER_{normalizado}(0)\left(1 + \frac{1}{TI} * t\right)$$



PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
GAIN	es la ganancia de regulador
$ER_{normalizado}(0)$	es la altura del escalón del error de regulación normalizado
TI	Constante de tiempo de integración

Regulación PD

En el regulador PD está desconectada la acción I (I–SEL = FALSE). Eso significa que con un error de regulación ER = 0, la señal de salida también es OUTV = 0. Para ajustar un punto de trabajo ≠ 0, es decir, un valor numérico para la señal de salida con un error de regulación cero, hay que activar la rama I (véase también el diagrama de bloques de la regulación P).

En la acción I se puede predeterminar un punto de trabajo ≠ 0 para el regulador PD ajustando el valor de inicialización I_ITLVAL en correspondencia. Para ello hay que activar los conmutadores 'I_ITL_ON' e 'I_SEL' TRUE .

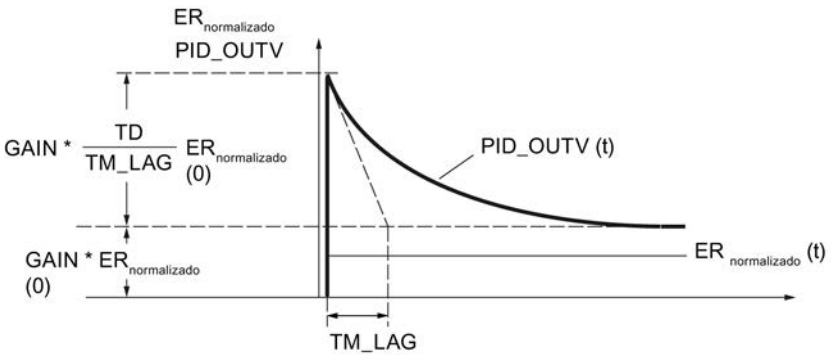
El regulador PD reproduce la magnitud de entrada ER(t) proporcionalmente sobre la señal de salida y añade la acción D creada por la derivación de ER(t), que se calcula con el doble de precisión según la regla del trapecio (aproximación de Padé). El comportamiento de tiempo se determina mediante la constante derivativa (tiempo de derivación) TD.

Para el filtrado de señales y la supresión de señales perturbadoras se ha integrado un retardo de 1er orden (tiempo ajustable: TM_LAG) en el algoritmo para formar la acción D.

Generalmente es suficiente un valor reducido para TM_LAG para conseguir el resultado deseado. Si se parametriza TM_LAG ≤ CYCLE/2 el retardo está desactivado.

La respuesta indicial en el rango de tiempo es:

$$PID_OUTV(t) = GAIN * ER_{normalizado}(0)\left(1 + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}}\right)$$



PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
GAIN	es la ganancia de regulador
$ER_{normalizado}(0)$	es la altura del escalón del error de regulación normalizado
TD	Tiempo derivativo
TM_LAG	Tiempo de retardo

Regulación PID

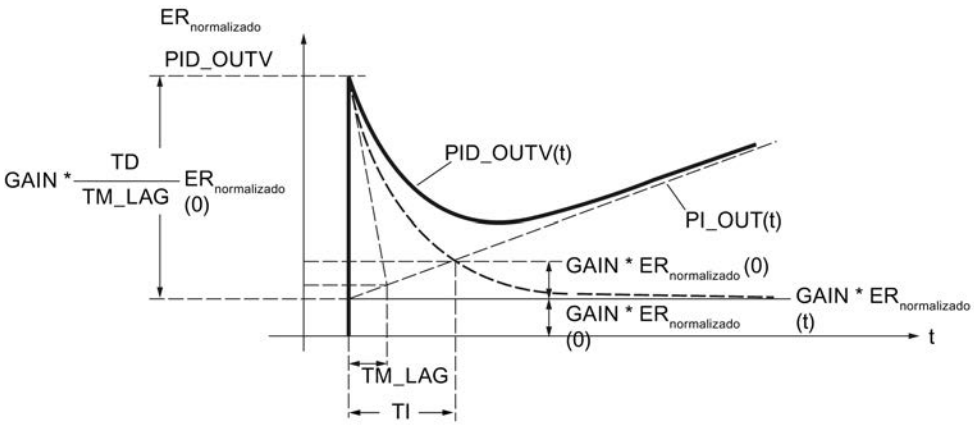
En el regulador PID están activadas las acciones P, I y D (P–SEL = TRUE, I–SEL = TRUE, D_SEL = TRUE). Un regulador PID modifica la magnitud de salida PID_OUTV mediante la acción I hasta que el error de regulación es ER = 0. Sin embargo, esto solo es aplicable cuando la magnitud de salida no rebasa los límites del rango manipulado. Si se rebasan los límites de la variable manipulada, la acción I mantiene el valor alcanzado en el límite en (Anti-Reset-Windup).

El regulador PID reproduce la magnitud de entrada normalizada $ER_{normalizado}(t)$ proporcionalmente sobre la señal de salida y añade las acciones creadas por la derivación e integración de $ER_{normalizado}(t)$, que se calculan con el doble de precisión según la regla de trapecio (aproximación Padé). La respuesta de tiempo se determina por el tiempo de derivación (tiempo de acción derivativa) TD y el tiempo de integración (tiempo de acción integral) TI.

Para el filtrado y la supresión de señales perturbadoras se ha integrado un retardo de 1er orden (constante de tiempo ajustable: TM_LAG) en el algoritmo para formar la acción D. Generalmente es suficiente un valor reducido para TM_LAG para conseguir el resultado deseado. Si se parametriza TM_LAG ≤ CYCLE/2 el retardo está desactivado.

La respuesta indicial en el rango de tiempo es:

$$PID_OUTV(t) = GAIN * ER_{normalizado}(0)\left(1 + \frac{1}{TI} * t + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}}\right)$$



PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
GAIN	es la ganancia de regulador
$ER_{normalizado}(0)$	es la altura del escalón del error de regulación normalizado
TI	Constante de tiempo de integración
TD	Tiempo derivativo
TM_LAG	Tiempo de retardo

Nota

Si TD cambia, conviene adaptar TM_LAG de la forma correspondiente.

Recomendación: 5 ≤ (TM / TM_LAG) ≤ 10

Consulte también

Algoritmo de regulación (Página 10085)

Acción P (Página 10093)

Acción D (Página 10096)

Acción I (Página 10094)

Acción P

Ganancia proporcional

El signo de la ganancia proporcional determina el sentido de acción del regulador continuo.

- Ganancia proporcional positiva
  error de regulación ascendente → variable manipulada ascendente
- Ganancia proporcional negativa
  error de regulación ascendente → variable manipulada descendente

Aplicar la acción P en rama de realimentación

Si la acción P está aplicada en la rama de realimentación, solo el valor real tendrá efecto sobre la acción P. Por ello, la magnitud manipulada – igual que en el caso de escalón de la consigna – en un cambio de automático a manual no realiza ningún escalón por la acción P: la conmutación es suave.

Consulte también

Algoritmo de regulación (Página 10085)

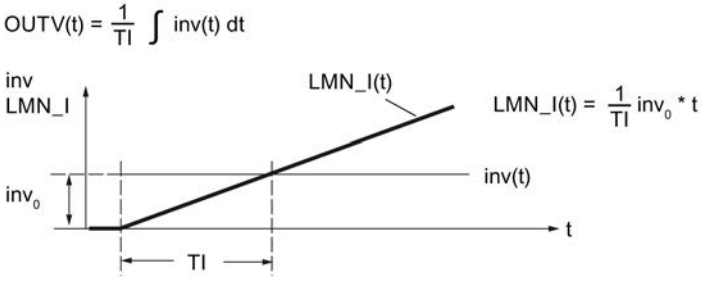
Acción I

Función de la acción I

La acción I hace que, reajustando el punto de operación, el error de regulación pueda convertirse en 0 para cualquier valor de la magnitud manipulada.

La acción I genera una señal de salida cuya velocidad de cambio es proporcional al cambio del valor absoluto de la magnitud de entrada. La respuesta temporal viene determinada por el tiempo integrativo TI.

La respuesta a un escalón de entrada inv_0 es:



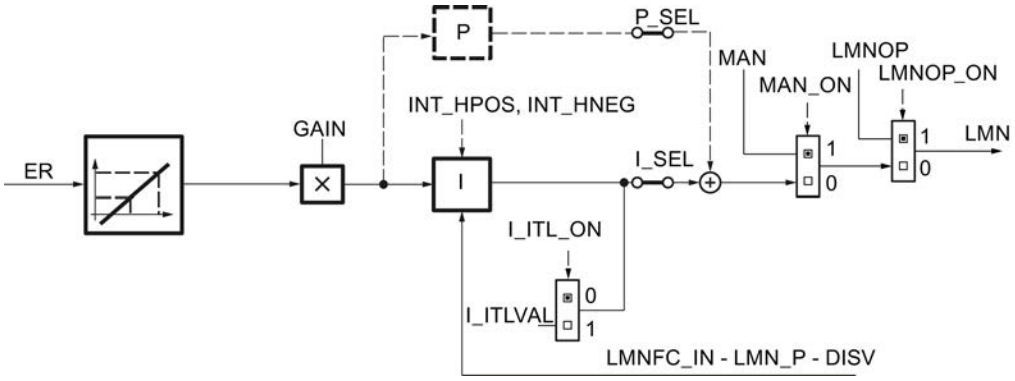
LMN_I(t)	la magnitud de salida de la acción I
inv_0	la altura del escalón en la entrada de la acción I
TI	Tiempo integrativo

La acción I se observa en la variable estática LMN_I.

En estado desactivado (I_SEL = FALSE) la acción I, es decir, la memoria interna y la variable estática LMN_I, se pone a cero.

La salida y la memoria de la acción I están limitados por los límites absolutos de la variable manipulada LMN_HLM y LMN_LLM (Anti Reset Wind–up).

En caso de influir en la señal manipulada en modo manual, es decir, cuando MAN_ON, LMNOP_ON o CAS_ON = TRUE, el valor de la memoria interna de la acción I se corrige según el valor LMNFC_IN - LMN_P - DISV.



Conmutación suave de manual a automático

En el modo manual, las acciones I y D del regulador se corrigen de tal forma que el regulador comienza la conmutación al modo automático con una magnitud manipulada adecuada.

- Conmutación suave activada (SMOO_CHG = TRUE)
  La acción I se activa en modo manual de modo que la magnitud manipulada permanece invariable en principio durante la conmutación de manual a automático.
  La acción D se pone a cero en modo manual. La conmutación a automático se lleva a cabo sin escalón de la magnitud manipulada.
  Si hay un error de regulación, este se regula lentamente.
- Conmutación suave desactivada (SMOO_CHG = FALSE)
  Durante la conmutación de manual a automático y partiendo de la posición actual, la magnitud manipulada realiza un escalón cuya altura queda determinada por la acción P. La altura del escalón de la acción P equivale al cambio de la magnitud manipulada cuando hay un escalón de consiga del valor real actual a la consigna actual.
  En modo manual, la acción D se pone a un valor que equivale al error de regulación existente. La conmutación a automático se realiza con escalón de magnitud manipulada; el error de regulación se regula más rápido.
  Sin embargo, si la acción P está aplicada en la rama de realimentación, solo el valor real tendrá efecto sobre la acción P. Por ello, en una conmutación de manual a automático, la magnitud manipulada no realiza ningún escalón por la acción P; la conmutación es suave.
  Lo mismo es válido para la acción D siempre que esta
  se haya aplicado en la rama de realimentación (DFDB_SEL = TRUE).
  El error de regulación existente se regula más rápido. Esto debe ser así, por ejemplo, en sistemas de regulación de temperatura.

Rangos admisibles para el tiempo integrativo TI y CYCLE

Debido a la precisión limitada de los números REAL calculados en la CPU, es posible que se produzca el efecto siguiente durante la integración: si se ha elegido un tiempo de muestreo CYCLE de la regulación demasiado pequeño en comparación con el tiempo integrativo TI y si el valor de entrada inv de la acción I es pequeño en comparación con su valor de salida OUTV, la acción I no se activa y permanece en su valor de salida momentáneo.

Este efecto puede evitarse si se considera la siguiente regla de dimensionado en la configuración:

$CYCLE > 10^{-4} \times TI$

De este modo, la acción I reacciona incluso a cambios de los valores de entrada situados en el rango de una millonésima parte por mil de la magnitud de salida actual:

$inv > 10^{-10} \times OUTV$

Para que el comportamiento de transferencia del algoritmo del integrador equivalga al comportamiento analógico, el tiempo de muestreo debería ser menor que el 20 % del tiempo integrativo ajustado o bien TI debería tener como mínimo cinco veces el valor del tiempo de muestreo seleccionado:

$CYCLE < 0{,}2 \times TI$

El algoritmo admite para el tiempo de muestreo valores hasta $CYCLE \leq 0{,}5 * TI$.

Inicializar la acción I

Cuando se activa la inicialización de la acción I, el valor de inicialización se aplica a la variable estática LMN_I en caso de rearranque de la CPU o si COM_RST = TRUE. Al cambiar al modo normal reseteando la inicialización, la acción I comienza la integración de su magnitud de entrada a partir del valor de inicialización.

Detener la acción I

Mediante las variables estáticas INT_HPOS y INT_HNEG es posible bloquear la acción I en sentido positivo y negativo, respectivamente. Esto puede ser útil en cascadas. Si, p. ej., la magnitud manipulada del regulador secundario se aproxima al límite superior, existe la posibilidad de impedir que siga aumentando la magnitud manipulada del regulador piloto mediante su acción I. Este comportamiento se programa con las instrucciones siguientes, p. ej., en AWL:

```
A    "regulador secundario".QLMN_HLM
=    "regulador piloto".INT_HPOS
A    "regulador secundario".QLMN_ LLM
=    "regulador piloto".INT_HNEG
```

Consulte también

Algoritmo de regulación (Página 10085)

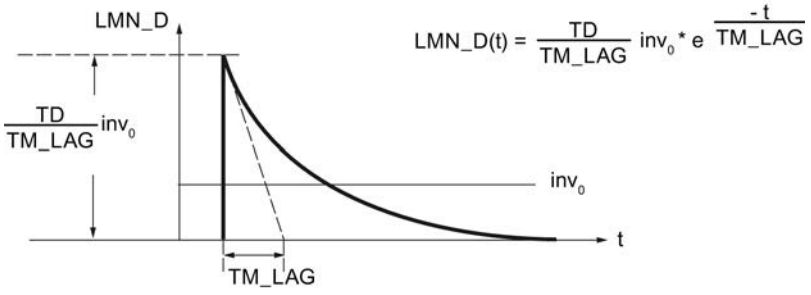
Estructuras de regulador (Página 10087)

Acción D

Función de la acción D

La acción D genera una señal de salida cuya magnitud cambia proporcionalmente a la velocidad de cambio de la magnitud de entrada. La respuesta temporal se determina mediante el tiempo derivativo TD y el tiempo de retardo TM_LAG.

La respuesta a un escalón de entrada inv_0 es:



LMN_D(t)	la magnitud de salida de la acción D
inv_0	la altura del escalón en la entrada de la acción D
TD	Tiempo derivativo
TM_LAG	Tiempo de retardo

La acción D se observa en la variable estática LMN_D.

En estado desactivado (D_SEL = FALSE) la acción D, es decir, la memoria interna y la variable estática LMN_D, se pone a cero.

Aplicar la acción D en la rama de realimentación

Si la acción D está aplicada en la rama de realimentación, solo el valor real tendrá efecto sobre la acción D. Por ello, en una conmutación de manual a automático, la magnitud manipulada – igual que en el caso de un escalón de la consigna – no realiza ningún escalón por la acción D; la conmutación es suave.

Valores admisibles para el tiempo derivativo y el tiempo de retardo

Para que la acción D pueda calcularse correctamente, deben observarse las condiciones siguientes para el tiempo derivativo y el tiempo de retardo:

- TD ≥ CYCLE y
- TM_LAG ≥ 0,5 × CYCLE

Si TD < CYCLE, la acción D trabaja como si TD tuviera el valor CYCLE.

Si TM_LAG < 0,5 × CYCLE, la acción D trabaja sin retardo. Un escalón de entrada se multiplica por el factor TD/CYCLE y este valor se devuelve en la salida como "impulso de aguja". Esto significa que en el próximo ciclo de procesamiento, LMN_D se pondrá de nuevo a cero.

Coeficiente DT1

El coeficiente DT1 se calcula del modo siguiente:

Coeficiente DT1 = TD/TM_LAG

Tiempo de retardo

Para el filtrado de señales y la supresión de señales perturbadoras se ha integrado un retardo de 1er orden (constante de tiempo ajustable: TM_LAG) en el algoritmo para formar la acción D.

La mayoría de las veces es suficiente con un valor pequeño como tiempo de retardo para alcanzar el resultado deseado. Si se parametriza un tiempo de retardo inferior a la mitad del tiempo de muestreo del regulador, el retardo está desactivado.

Procedimiento

Para configurar la acción D, proceda del siguiente modo:

1. Para aplicar la acción D en la rama de realimentación, active la casilla de verificación correspondiente.
2. Introduzca el tiempo derivativo en segundos.
3. Introduzca el tiempo de retardo en segundos.

Consulte también

Algoritmo de regulación (Página 10085)

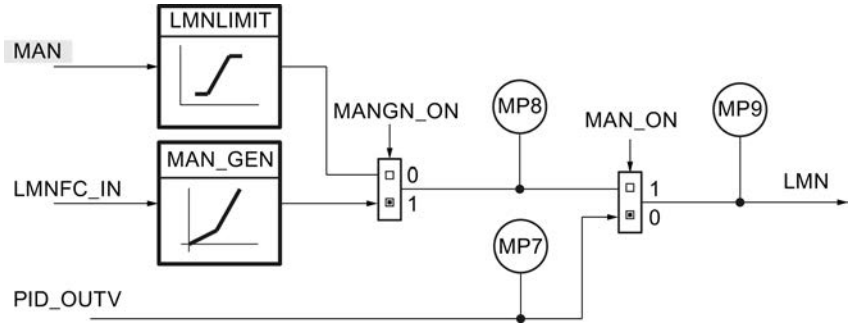
Variable manipulada

Modo manual

Además del modo de operación "Automático" con la salida del algoritmo PID (PID_OUTV) aplicada directamente en la salida, existen dos modos de operación en los que es posible influir manualmente en la señal manipulada: "Modo manual sin manejo del interruptor" y "Modo manual mediante interruptor de más/menos" (MAN_GEN).

A través del parámetro MAN se puede aplicar una influencia externa sobre la magnitud manipulada, o puede venir dada por el programa de usuario. El valor de entrada MAN se limita a los límites de magnitud manipulada LMN_HLM (superior) y LMN_LLM (inferior).

La estructura de la generación y aplicación del valor manual puede verse en la figura siguiente. Al activar MAN_GEN desde otro modo de operación, se aplica la variable manipulada actualmente efectiva en la salida de MAN_GEN. Así, la conmutación al generador del valor manual siempre es suave.



Activar el modo automático

Si se ha desactivado el modo manual (MAN_ON = FALSE)), la variable manipulada del algoritmo PID se aplica directamente a la salida. En el modo manual (MAN_ON = TRUE), la acción I del regulador se controla de tal forma que el regulador comienza la conmutación al modo automático con una magnitud manipulada adecuada. La salida del algoritmo PID se encuentra en el punto de medición MP7.

En el modo automático, el valor manual MAN sigue a la magnitud manipulada (menos la acción D). Por ello, en la conmutación al modo manual la magnitud manipulada permanece en el último valor calculado. Puede modificarse mediante intervención manual.

1. Desactive la casilla de verificación "Activar modo manual".

Activar el modo manual con valor manual

En este modo de operación (MANGN_ON = FALSE y MAN_ON = TRUE) se introduce la variable manipulada manual como valor absoluto en la entrada MAN. La variable manipulada manual se indica en el punto de medición MP8.

1. Active la casilla de verificación "Activar modo manual".
2. Seleccione la opción "Valor manual".

Activar el generador de valor manual

El valor manual MAN se aumenta o reduce progresivamente utilizando los interruptores MANUP y MANDN. El valor modificado puede observarse en MP8. Se tienen en cuenta los límites configurados de la variable manipulada.

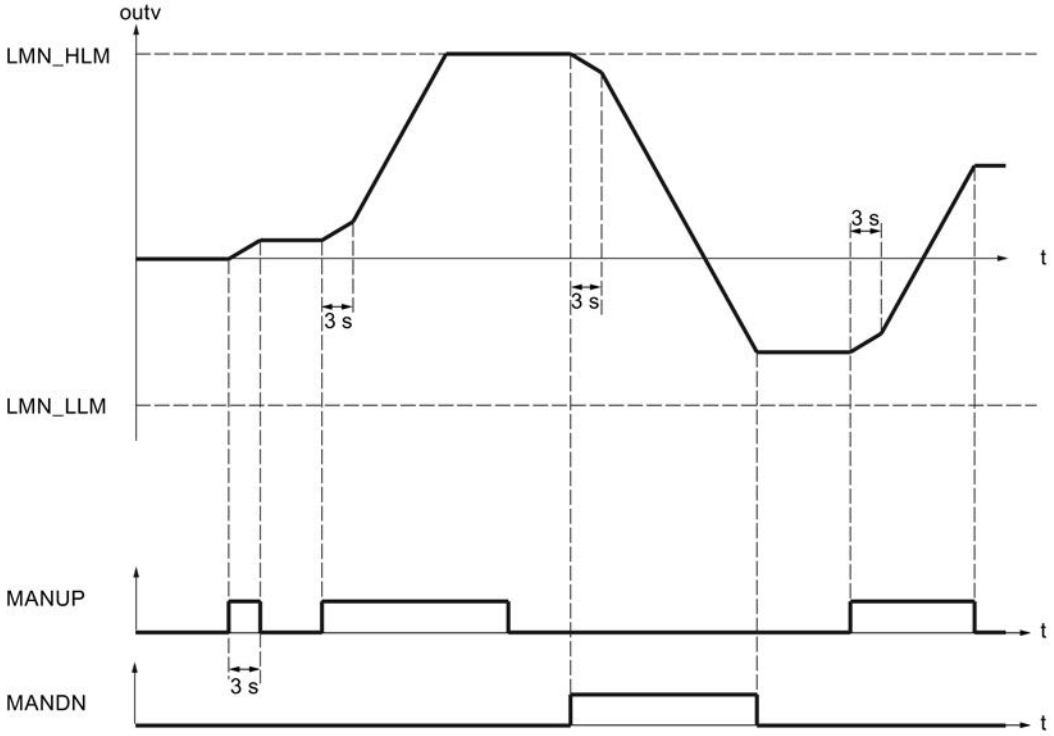
1. Active la casilla de verificación "Activar modo manual".
2. Seleccione la opción "Generador del valor manual".

Repercusión de los interruptores MANUP y MANDN

Para poder realizar un cambio graduado finamente, el regulador debe tener un tiempo de muestreo de 100 ms como máximo.

La velocidad de cambio de la variable manipulada depende de la duración de pulsación de los interruptores MANUP y MANDN así como de los límites ajustados de acuerdo con la relación siguiente:

- durante los primeros 3 segundos después de activar MANUP o MANDN rige:
  doutv/dt = (LMN_HLM - LMN_LLM) / 100 s
- a continuación
  doutv/dt = (LMN_HLM - LMN_LLM) / 10 s



Con un tiempo de muestreo de 100 ms y un rango de la variable manipulada de -100,0 a 100,0, la variable manipulada cambia en 0,2 por ciclo en los tres primeros segundos, por ejemplo. Si MANUP se mantiene en TRUE durante más de 3 segundos, la velocidad de cambio aumenta a diez veces el valor, en este caso a 2 por ciclo.

El generador del valor manual se activa suavemente.

Modos de operación

La tabla siguiente muestra los modos de operación posibles del regulador continuo con los valores necesarios de los interruptores.

Modo de operación	MAN_ON	MANGN_ON
Modo automático	FALSE	no aplicable
Modo manual con valor manual	TRUE	FALSE
Modo manual con generador del valor manual	TRUE	TRUE

Límites

Llamar una función

Insertando una función específica de usuario en la rama de la magnitud manipulada se puede someter la magnitud manipulada PID_OUTV generada en el regulador a un procesamiento de señal (p. ej. un retardo) antes de la aplicación en la salida del regulador.

El regulador ejecuta una llamada de la función. En ella no se suministran valores a los parámetros de entrada y salida de la función de usuario. La transferencia de datos debe programarla el usuario en su función.

Nota

No se comprueba si hay una función disponible. Si la función no está disponible, la CPU pasa a STOP con un error interno de sistema.

1. Active la casilla de verificación "Activar llamada de FC".
2. Introduzca el número de la función en el campo de entrada "FC".

Limitar la velocidad de cambio

Las funciones de rampa en la rama de la magnitud manipulada se utilizan cuando el proceso no soporta cambios de la señal de entrada del proceso en forma de esalón. Los cambios abruptos de la variable manipulada deben evitarse p. ej. en aquellos casos en que entre el motor regulado y la carga que debe accionarse hay interpuesto un engranaje, y un aumento muy rápido del número de revoluciones del motor derivaría en una sobrecarga del engranaje.

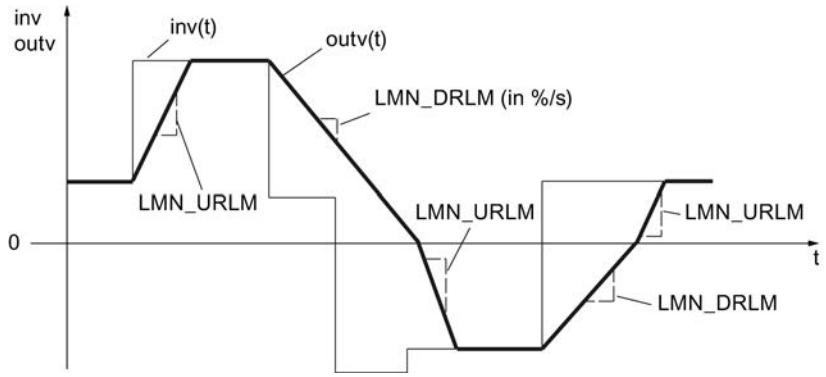
La velocidad de cambio de las variables manipuladas en la salida del regulador se limita por separado para la rampa de subida y bajada de la magnitud manipulada. Para todo el rango de valores es posible parametrizar dos rampas con valores ascendentes o descendentes partiendo desde el punto cero. La función está activada si LMNRC_ON = TRUE.

En las dos variables estáticas LMN_URLM y LMN_DRLM se introducen las limitaciones correspondientes para las pendientes de la función de rampa en los rangos positivo y negativo de la magnitud manipulada. Las pendientes se refieren a la subida o bajada en porcentajes por segundo. Los cambios más rápidos de la variable manipulada se retardan a estas velocidades límite.

Si p. ej. 'LMN_URLM' se parametriza a 10,0 [%], en cada ciclo de muestreo se suman los siguientes valores al "valor antiguo" de outv mientras |inv| sea > |outv|:

Tiempo de muestreo	1 s	→ $outv_{antiguo}$ + 10 %
	100 ms	→ $outv_{antiguo}$ + 1 %
	10 ms	→ $outv_{antiguo}$ + 0,1 %

La figura siguiente muestra el funcionamiento del procesamiento de las señales. Las funciones de escalón en la entrada inv (t) se traducen en funciones de rampa en la salida outv (t).



No se notifica el alcance de los límites de la pendiente.

La identificación de los parámetros de las rampas se realiza según el siguiente esquema:

Parámetro	Rampa
LMN_URLM	|outv| ascendente
LMN_DRLM	|outv| descendente

1. Active la casilla de verificación "Limitar la velocidad de cambio".
2. Introduzca la rampa de subida máxima de la variable manipulada en el campo de entrada "Rampa de subida máxima".
   Los cambios más rápidos de la variable manipulada se retardan a ese valor.
3. Introduzca la rampa de bajada máxima de la variable manipulada en el campo de entrada "Rampa de bajada máxima".
   Los cambios más rápidos de la variable manipulada se retardan a ese valor.

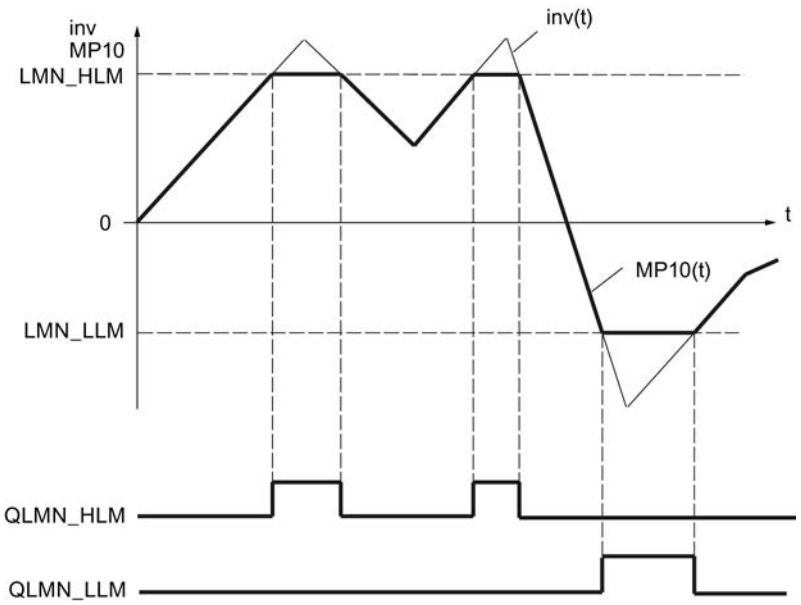
Limitar el valor absoluto de la magnitud manipulada

El rango de trabajo, es decir, el rango dentro del cual puede moverse el actuador sin abandonar las variables manipuladas admisibles, está determinado por el rango de ajuste de la variable manipulada. Dado que, dependiendo del tipo de actuador, los límites de las variables manipuladas permitidas no suelen cubrirse con el límite 0 % o 100 % del rango de ajuste, es posible configurar el límite superior e inferior de la variable manipulada.

Para evitar estados no permitidos en el proceso en cuestión, en la rama de la variable manipulada de la regulación se limita el rango de ajuste de la magnitud manipulada por arriba y abajo.

La magnitud manipulada LMN(t) se limita a los límites superior (LMN_HLM) e inferior (LMN_LLM) predefinibles. Sin embargo, la magnitud de entrada inv debe estar fuera de estos límites. Dado que la función no puede desactivarse, en la configuración siempre hay que tener en cuenta la indicación de los límites inferior y superior.

Si la magnitud de entrada inv(t) rebasa los límites, se emiten las indicaciones correspondientes a través de las salidas de aviso.



Arranque y funcionamiento

- En caso de rearranque, todas las salidas de aviso se ponen a cero.
- La limitación trabaja según las siguientes relaciones:

LMN =	QLMN_HLM	QLMN_LLM	si:
LMN_HLM	TRUE	FALSE	inv ≥ LMN_HLM
LMN_LLM	FALSE	TRUE	inv ≤ LMN_LLM
inv	FALSE	FALSE	LMN_HLM < inv < LMN_LLM

En la salida, es decir, en el parámetro LMN o en el punto de medición MP10, se muestra la variable manipulada efectiva de la regulación.

Procedimiento

Para limitar el valor absoluto de la magnitud manipulada, proceda del siguiente modo:

1. Introduzca el valor más alto permitido en el campo de entrada "Límite superior".
2. Introduzca el valor más bajo permitido en el campo de entrada "Límite inferior".

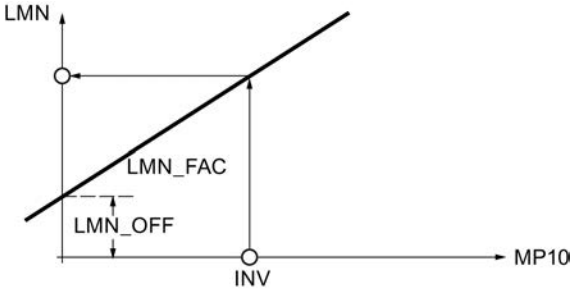
Normalización

Si a la entrada del proceso regulado se requiere una magnitud física como magnitud manipulada, los valores en coma flotante deben desnormalizarse entre 0 y 100 % al rango físico (p. ej. 150 a 3000 rpm) de la magnitud manipulada.

Para definir la recta de normalización de forma unívoca deben definirse dos parámetros:

- el factor para la pendiente (LMN_FAC)
- el offset de la recta de normalización en el punto cero (LMN_OFF)

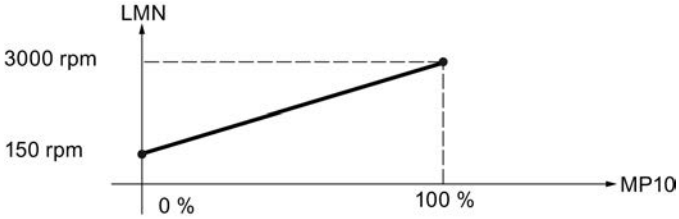
Valores porcentuales internos (en formato REAL) ⇒ valores físicos externos



El valor de normalización se calcula con la función siguiente a partir del correspondiente valor de entrada MP10:

LMN = MP10 * LMN_FAC + LMN_OFF

Ejemplo



$$LMNFAC = \frac{(3000 - 150)\ rpm}{(100 - 0)\%} = 28{,}5\ \frac{rpm}{\%}$$

LMNOFF = 150 rpm

Dentro de la función no se limitan valores ni se comprueban parámetros.

Realizar una normalización

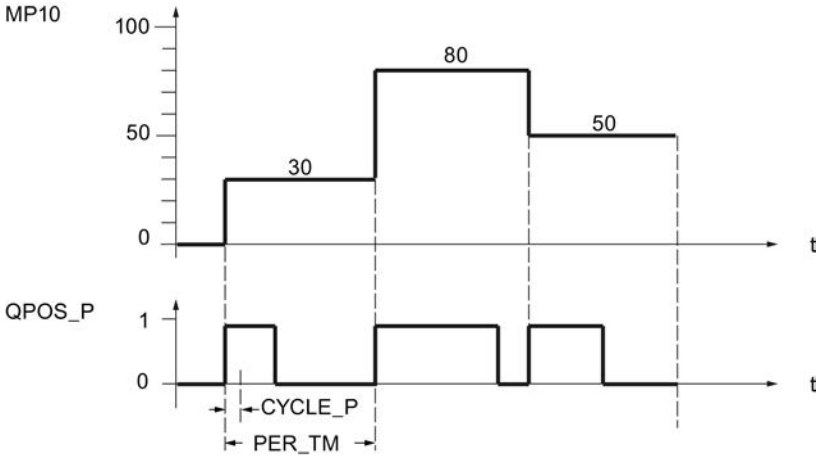
1. En el campo de entrada "Factor", introduzca la pendiente de la recta de normalización.
2. En el campo de entrada "Offset", introduzca el offset de la recta de normalización hacia el punto cero.

Generador de impulsos

Aplicación

El generador de impulsos transforma la magnitud de entrada "Variable manipulada del regulador PID en el punto de medición MP10" mediante modulación de ancho de impulsos en

un tren de impulsos con duración de período constante. La duración de un impulso por período es proporcional a la magnitud de entrada.



Una magnitud de entrada del 30% y 10 llamadas del generador de impulsos por duración de período significan, por tanto:

- "uno" en la salida QPOS para las tres primeras llamadas (30% de 10 llamadas),
- "cero" en la salida QPOS para otras siete llamadas (70% de 10 llamadas).

Modos de operación del regulador con salida de impulsos

Dependiendo de la parametrización del generador de impulsos es posible configurar reguladores PID con comportamiento de tres puntos o con salida de dos puntos bipolar o unipolar. La siguiente tabla muestra el ajuste de las combinaciones de interruptores para los posibles modos de operación.

Modo de operación	Interruptor		
	MAN_ON	STEP3_ON	ST2BI_ON
Regulación de tres puntos	FALSE	TRUE	cualquiera
Regulación de dos puntos con rango de ajuste bipolar (de –100 % a 100 %)	FALSE	FALSE	TRUE
Regulación de dos puntos con rango de ajuste unipolar (de 0 % a 100 %)	FALSE	FALSE	FALSE
Modo manual	TRUE	cualquiera	cualquiera

Regulación de tres puntos

En el modo de operación "Regulación de tres puntos" se pueden generar tres estados de la señal manipulada, por ejemplo, según actuador y proceso: más – apagado – menos, adelante – parado – atrás, calentar – apagado – enfriar, etc. Según los requisitos del proceso que se

desea regular se asignan los valores de estado de las señales de salida binarias QPOS_P y QNEG_P a los respectivos estados operativos del actuador. La tabla muestra dos ejemplos.

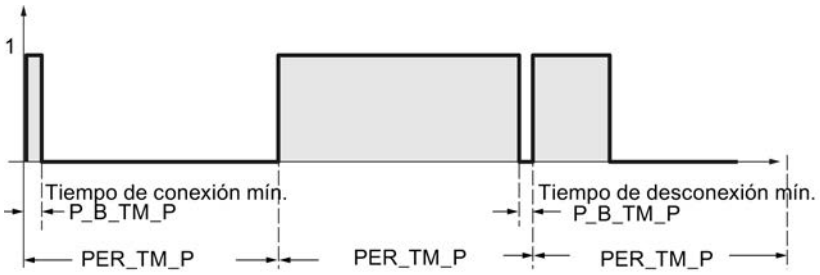
	calentar adelante	apagado parado	enfriar atrás
QPOS_P	TRUE	FALSE	FALSE
QNEG_P	FALSE	FALSE	TRUE

El dimensionado de la duración mínima de impulso o pausa P_B_TM_P puede evitar tiempos de conexión o desconexión muy cortos que merman la vida útil de los elementos de conmutación y los dispositivos de control. Para ello se sobrepone un valor de reacción para la salida de impulsos a la característica de salida proporcional mediante la que se calcula la duración de impulso.

Nota

Se suprimen valores absolutos pequeños de la magnitud de entrada "Variable manipulada del regulador PID en el punto de medición MP10" que podrían generar una duración de impulso menor que P_B_TM_P. Para los valores de entrada grandes, que generarían una duración de impulso mayor que PER_TM_P – P_B_TM_P, se ajusta la duración de impulso a 100% o –100%.

Se recomiendan valores de ajuste P_B_TM_P ≤ 0,1 * PER_TM_P.

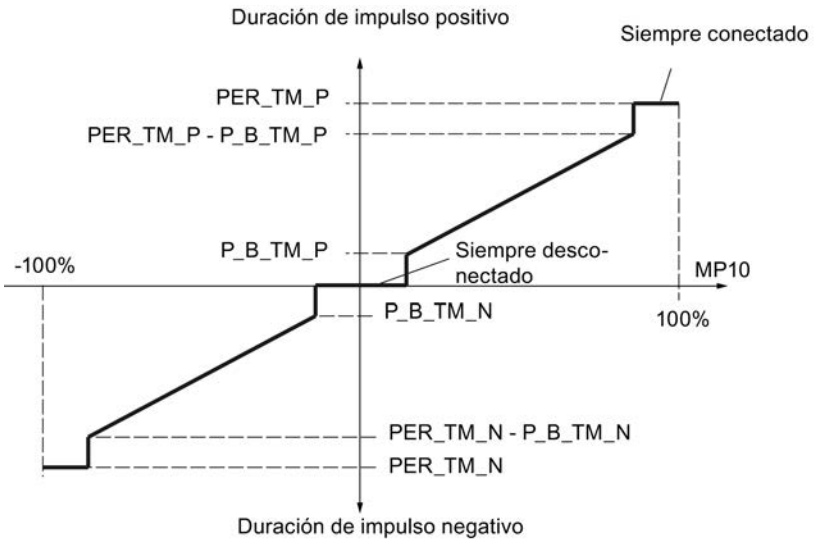


La duración de los impulsos positivos o negativos se calcula multiplicando la magnitud de entrada "Variable manipulada del regulador PID en el punto de medición MP10" (en %) por el período:

$$\text{Duración de impulso} = \frac{MP10}{100} * PER_TM_P[s]$$

Debido a la supresión de la duración mínima de impulso o pausa, se forman puntos de inflexión en el área inicial y final de la característica de conversión.

Lo dicho anteriormente también es válido para P_B_TM_N y PER_TM_N.



Regulación de tres puntos asimétrica

A través del factor de relación RATIOFAC puede modificarse la relación entre la duración de impulsos positivos y negativos. En un proceso térmico, ello permite tener en cuenta p. ej. constantes temporales de proceso distintas para calentar y para enfriar.

Si se desea acortar la duración de impulso en la salida de impulso negativo con respecto al impulso positivo y si el valor absoluto "Variable manipulada del regulador PID en el punto de medición MP10" es idéntico, se debe introducir un factor de relación menor que 1:

Impulso pos. > impulso neg.: RATIOFAC < 1

Duración de impulso negativo: $\frac{MP10}{100} * PER_TM_N * RATIOFAC$

Duración de impulso positivo: $\frac{MP10}{100} * PER_TM_N$

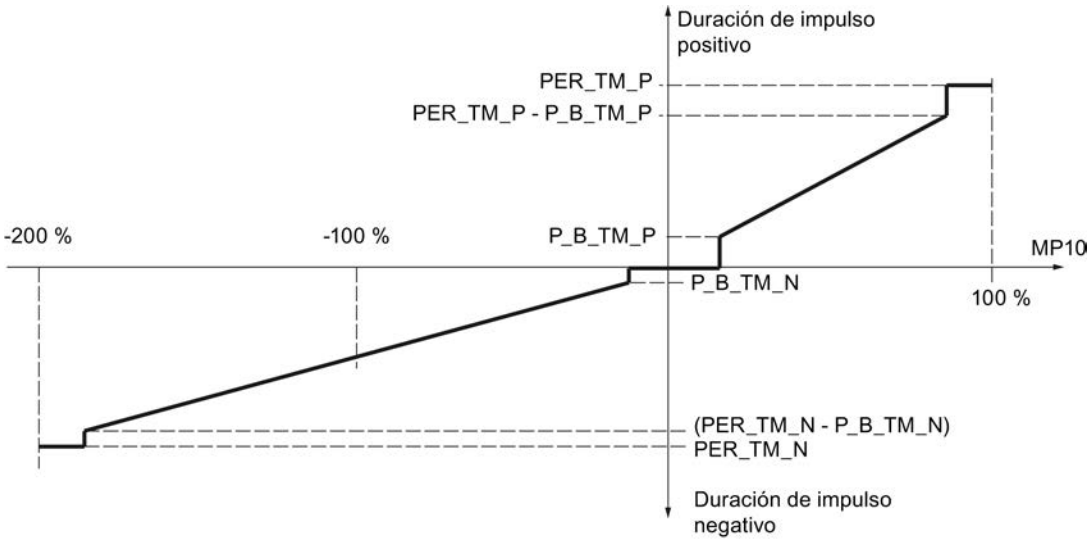


Figura 11-5 Característica asimétrica del regulador de tres puntos (factor de relación = 0,5)

Si, por el contrario, se desea acortar la duración de impulso en la salida de impulso positivo con respecto al impulso negativo y si el valor absoluto |MP10| es idéntico, se debe introducir un factor de relación mayor que 1:

Impulso pos. < impulso neg.: RATIOFAC > 1

Duración de impulso negativo: $\frac{MP10}{100} * PER_TM_N$

Duración de impulso positivo: $\frac{MP10 * PER_TM_P}{100 * RATIOFAC}$

En lo que respecta al cálculo, esto significa que con RATIOFAC < 1 el valor de reacción para impulsos negativos se debe multiplicar por el factor de relación y que con RATIOFAC > 1 el valor de reacción para impulsos positivos se debe dividir por el factor de relación.

Nota

En la regulación de tres puntos asimétrica RATIOFAC ≠ 1, los límites de la variable manipulada deben ajustarse de acuerdo con las fórmulas siguientes:

RATIOFAC < 1:

LMN_HLM = 100

LMN_LLM = –100 * (1 / RATIOFAC)

RATIOFAC > 1:

LMN_HLM = 100 * RATIOFAC

LMN_LLM = –100

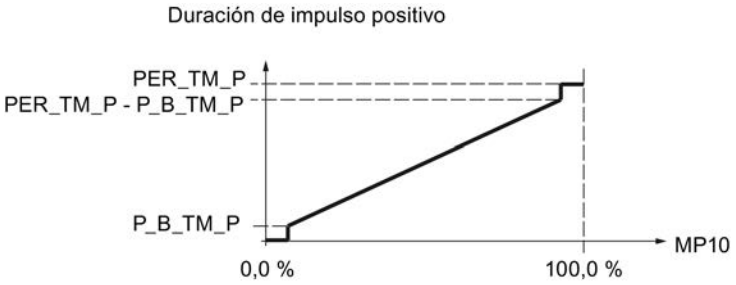
Ejemplos:

RATIOFAC	=	1	RATIOFAC	=	0.5	RATIOFAC	=	2.0
LMN_HLM	=	100	LMN_HLM	=	100	LMN_HLM	=	200
LMN_LLM	=	–100	LMN_LLM	=	–200	LMN_LLM	=	–100

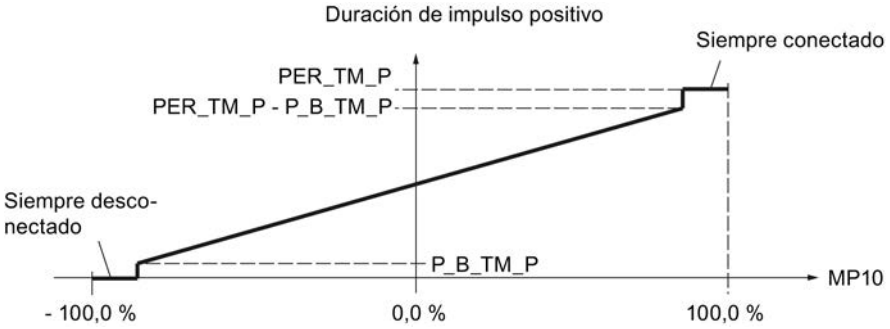
Regulación de dos puntos

En la regulación de dos puntos solo se conecta la salida de impulsos positiva QPOS_P al actuador de conexión/desconexión en cuestión. Dependiendo del rango de ajuste utilizado, MP10 = -100,0% a +100,0% o MP10 = 0,0% a 100,0%, el regulador de dos puntos tiene un rango de ajuste bipolar o unipolar.

En el modo de operación unipolar, MP10 solo debe adoptar valores entre 0,0% y 100,0%.



En el modo de operación bipolar, MP10 solo debe adoptar valores entre -100,0% y -100,0%.



En QNEG_P está disponible la señal de salida negada para el caso en que la interconexión del regulador de dos puntos en el lazo de regulación exige una señal binaria invertida lógicamente para los impulsos de la señal manipulada.

	ON	OFF
QPOS_P	TRUE	FALSE
QNEG_P	FALSE	TRUE

Tiempo de muestreo del regulador CYCLE y ancho de impulso CYCLE_P

Si ha activado la etapa de formación de impulsos (PULSE_ON = TRUE), será necesario especificar el ciclo del OB de alarma cíclica invocante en la entrada CYCLE_P. La duración del impulso generado siempre asciende a un múltiplo entero de este valor.

En la entrada CYCLE se especifica el tiempo de muestreo para las demás funciones de regulación del PID_CP. PID_CP determina la reducción del ciclo y procesa las funciones de regulación con el tiempo de muestreo CYCLE.

CYCLE debe ser un múltiplo entero de CYCLE_P. Si no se cumple esta condición, PID_CP redondea el tiempo de muestreo para las funciones de regulación a un múltiplo entero de CYCLE_P. Las funciones dependientes del tiempo (p. ej. filtrado, integración, diferenciación) no trabajan con toda la corrección en este caso.

CYCLE puede ser menor que la duración de período PER_TM_P o PER_TM_N. Esto es aconsejable si, por un lado, se desea una duración de período lo más grande posible para proteger los actuadores y, por otro lado, el tiempo de muestreo debe ser pequeño porque el sistema regulado es rápido.

Un valor adecuado para el tiempo de muestreo CYCLE es, al igual que en el regulador continuo sin etapa de formación de impulsos, que CYCLE no debe ser menor que aprox. el 10% de las constantes de tiempo dominantes en el sistema regulado.

Ejemplo de los efectos de los parámetros CYCLE_P, CYCLE y PER_TM_P o PER_TM_N:

- PER_TM_P = 10 s, CYCLE = 1 s, CYCLE_P = 100 ms.

Cada segundo se calcula una nueva variable manipulada, cada 100 ms se produce la comparación de la variable manipulada con la longitud de impulso o pausa emitida hasta entonces.

Cuando se emite un impulso y la variable manipulada calculada es mayor que la longitud de impulso actual / PER_TM_P, se prolonga el impulso. En caso contrario, no se emiten más señales de impulso. Si no se emite ningún impulso y (100% - variable manipulada calculada) es mayor que la longitud de pausa actual / PER_TM_P, se prolonga la pausa. En caso contrario se emite una señal de impulso.

Debido a un procedimiento especial de formación de impulsos, un aumento o una reducción de la magnitud manipulada durante el período provoca una prolongación o reducción del impulso emitido, respectivamente. Si, en este caso (CYCLE < PER_TM_P), la duración de período se ha parametrizado tan grande que provocaría una oscilación del valor real, la duración de período efectiva de PID_CP se reduce automáticamente a un valor adecuado.

Precisión de la formación de impulsos

La duración de período PER_TM no es idéntica al ciclo de procesamiento del generador de impulsos. Una duración de período se compone de varios ciclos de procesamiento del generador de impulsos. El número de llamadas del generador de impulsos por duración de período es una medida para la precisión del ancho de impulso.

Cuanto menor sea el tiempo de muestreo CYCLE_P en comparación con la duración de período PER_TM_P (o PER_TM_N), más exacta será la modulación de ancho de impulsos. Para una regulación con una precisión suficiente debe cumplirse la condición siguiente:

$$\text{CYCLE_P} \leq \frac{\text{PER_TM}}{20 \ldots 50}$$

Realización de tiempos de muestreo muy cortos

En un sistema regulado rápido son necesarios tiempos de muestreo muy cortos (p. ej. 10 ms). A causa del tiempo de ejecución del programa, en este caso no es aconsejable procesar los

elementos de la regulación en el mismo OB de alarma cíclica que el cálculo de la salida de impulsos. Es mejor trasladar el procesamiento de las funciones de regulación al OB 1 o a un OB de alarma cíclica más lento. El procesamiento de la función de regulación en el OB 1 solo tiene sentido si el tiempo de ciclo del OB 1 es claramente inferior al tiempo de muestreo CYCLE del regulador.

Mediante el parámetro SELECT se especifica la parte del programa que debe procesarse. La siguiente tabla ofrece una visión de conjunto de la parametrización del parámetro de entrada SELECT:

SELECT	Funcionalidad utilizada en el bloque	Método subyacente
0	Elemento de regulación y salida de impulsos	Elemento de regulación y salida de impulsos en el mismo bloque
1	Llamada en el OB 1 (elemento de regulación)	Elemento de regulación en el OB 1, salida de impulsos en el OB de alarma cíclica rápido
2	Llamada en el OB de alarma cíclica (salida de impulsos)	
3	Llamada en el OB de alarma cíclica lento (elemento de regulación)	Elemento de regulación en el OB de alarma cíclica lento, salida de impulsos en el OB de alarma cíclica rápido
2	Llamada en el OB de alarma cíclica rápido (salida de impulsos)	

A continuación se explican con detalle los métodos citados en la tabla superior para conseguir anchos de impulso muy cortos.

- Función de regulación en el OB 1, salida de impulsos en el OB de alarma cíclica
  Cuando se llama con SELECT = 2 se calcula la salida de impulsos y se comprueba si el tiempo de muestreo parametrizado en CYCLE ha transcurrido desde el último procesamiento del elemento de regulación. Si dicho tiempo de muestreo ha transcurrido, PID_CP escribe la variable QC_ACT en el DB de instancia con el valor TRUE.
  Cuando se llama con SELECT = 1 se evalúa la variable QC_ACT en el DB de instancia: Si QC_ACT tiene el valor FALSE, el bloque finaliza inmediatamente; así pues, ha necesitado un tiempo de ejecución muy corto. Si QC_ACT tiene el valor TRUE, el elemento de regulación se ejecuta una vez y, seguidamente, QC_ACT vuelve a ponerse a cero. Este procedimiento hace que el tiempo de muestreo para la función de regulación no pueda respetarse con exactitud. Oscila alrededor del tiempo de ejecución del OB 1 (incluidas todas las interrupciones). Por consiguiente, este procedimiento solo es adecuado si el tiempo de ejecución del OB 1 es pequeño en comparación con el tiempo de muestreo CYCLE.
- Función de regulación en el OB de alarma cíclica lento, salida de impulsos en el OB de alarma cíclica rápido
  Si se llama con SELECT = 2 se calcula siempre la salida de impulsos.
  Si se llama con SELECT = 3 se calcula siempre el elemento de regulación.

Nota

Para no tener que programar dos veces la llamada de la instrucción PID_CP con los muchos operandos formales que contiene, es recomendable programarla una sola vez en una FC que también tenga como parámetro de entrada un parámetro SELECT. Este parámetro de entrada se interconecta con PID_CP.SELECT. En tal caso, en el OB 1 o en el OB de alarma cíclica solo debe llamarse dicha FC.

Este procedimiento facilita los cambios y ahorra memoria de programa.

Puesta en servicio de PID_CP

Realizar una optimización manual

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- MAN_ON = FALSE

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Medición activada".
   Si aún no existe ninguna conexión online, ésta se establece. Los valores actuales de consigna, valor real y variable manipulada se graban.
2. Introduzca parámetros PID nuevos en los campos "P", "I", "D" y "Tiempo de retardo".
3. Haga clic en el grupo "Optimización" en el símbolo "Enviar parámetros a la CPU".
4. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .

5. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
   Si la velocidad de cambio de la consigna está limitada, se conmuta suavemente a la nueva consigna.
6. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabaja con los nuevos parámetros PID y regula según la nueva consigna.
7. Compruebe la calidad de los parámetros PID basándose en la evolución de las curvas.
8. Repita los pasos del 3 hasta el 8, hasta que esté satisfecho con el resultado de la regulación.

Trabajar con el visor de curvas

El visor de curvas permite representar gráficamente los valores de consigna, real y manipulado a lo largo del tiempo. Los valores del visor de curvas se actualizan en el tiempo de actualización seleccionado. El registro de los valores de las curvas comienza al hacer clic en el botón "Medición on" y finaliza al hacer clic en el botón "Medición off".

Elementos del visor de curvas

① Selección del modo de visualización
② Ventana de curvas
③ Área para mover y escalar los ejes
④ Regla
⑤ Leyenda con los valores de las curvas en la regla

Selección del modo de visualización

Para la representación del registro de curvas pueden seleccionarse los siguientes modos de visualización:

- Strip (visualización continuada)
  Los valores de curva nuevos se registran en la ventana de curvas derecha. Los valores de curva transcurridos se van deslizando a la izquierda. No es posible mover el eje de tiempo.
- Scope (visualización cambiante por áreas)
  Los valores de curva nuevos se registran de izquierda a derecha dentro de la ventana de curvas. Cuando se alcanza el margen derecho de la ventana de curvas, el área de observación se traslada un ancho de ventana a la derecha. El eje de tiempo puede moverse dentro de los límites del área de observación.
- Sweep (visualización en rotación)
  Los valores nuevos se muestran en la ventana de curvas en forma de visualización en rotación. Los valores se registran de izquierda a derecha. En la posición de escritura se sobrescriben los valores de la última rotación. No es posible mover el eje de tiempo.
- Static (visualización estática por áreas)
  La escritura de las curvas se interrumpe: los nuevos valores se registran en segundo plano. El eje de tiempo puede moverse a lo largo de todo el periodo registrado hasta ahora.

La relación entre el área visible y el área registrada puede consultarse en el visor de relación:

El ancho del visor de relación equivale a toda la duración del registro. La parte amarilla del visor representa la parte del registro visible en la ventana de curvas. La parte gris representa la parte del registro **no** visible en la ventana de curvas.

En el modo de visualización "Static" es posible mover el área visible del visor de relación con el ratón a lo largo del eje.

Ventana de curvas

En la ventana de curvas se muestran las curvas correspondientes a los valores de consigna, real y manipulado. Además de la distinción por colores, las curvas se identifican mediante símbolos (ver leyenda).

Mover y escalar los ejes

Los ejes correspondientes a los valores de consigna, real y manipulado, así como al eje de tiempo, pueden moverse y escalarse individualmente. Los botones derecho e izquierdo del ratón tienen asignadas funciones cambiantes. Puede utilizar los siguientes botones y acciones de ratón:

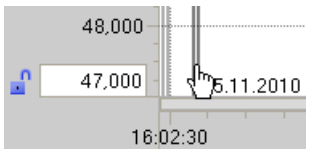
	Mover el eje de la consigna, del valor real o de la variable manipulada hacia arriba y abajo Si el valor de escala superior o inferior están bloqueados, el eje no puede moverse.
	Mover el eje de tiempo a derecha e izquierda Si el valor de escala izquierdo o derecho están bloqueados, el eje no puede moverse.

	Expansión y contracción comunes de los ejes de los valores de consigna, real y manipulado • Si ninguno de los valores de escala está bloqueado, la escala de los ejes se estira o encoge simétricamente. • Si uno de los valores de escala está bloqueado, se conserva el valor bloqueado al expandir o contraer la escala.
	Expandir y contraer el eje de tiempo • Si ninguno de los valores de escala está bloqueado, la escala del eje se estira o se encoge simétricamente. • Si uno de los valores de escala está bloqueado, se conserva el valor bloqueado al expandir o contraer la escala.
	Expansión y contracción de los ejes de consigna, valor real y variable manipulada El valor de escala inferior no se modifica al expandir o contraer la escala.
	Expansión y contracción de los ejes de consigna, valor real y variable manipulada. El valor de escala superior no se modifica al expandir o contraer la escala.
	Expansión y contracción del eje de tiempo. El valor de escala derecho no se modifica al expandir o contraer la escala.
	Expansión y contracción del eje de tiempo. El valor de escala izquierdo no se modifica al expandir o contraer la escala.
50,000	Entrada de un valor de escala.
50,000	El símbolo de un candado permite bloquear el valor de escala actual. Sólo es posible bloquear un valor de cada eje.
	Haciendo doble clic en la ventana de curvas se optimiza la escala y la posición de los ejes de los valores de consigna, real y manipulado en la ventana de curvas.
	Haciendo doble clic en el área del eje se restablecen la posición y la escala predeterminadas del eje.

Trabajar con reglas

Utilice una o más reglas para analizar valores discretos de las curvas.

Las posiciones de estacionamiento para reglas verticales se encuentran en el margen izquierdo del área de curva, para las reglas horizontales, en el margen superior.



Mueva el ratón hasta el margen izquierdo del área de curvas y observe cómo cambia el puntero del ratón. Arrastre ahora la regla vertical hasta la posición de la curva de medición que desea analizar.

Los valores de la curva situados junto a la regla vertical se muestran en ésta sangrados a la izquierda. El valor de tiempo de la posición de la regla se muestra en la base de ésta.

Los valores de la curva mostrados en la regla activa se muestran en la leyenda. Si se arrastran varias reglas al área de curvas, estará activa la última regla arrastrada. La regla activa se reconoce por el símbolo en color.

Sin embargo, los valores de las reglas horizontales no pueden representarse.

Las reglas no activas pueden volver a activarse haciendo clic.

Las reglas que ya no se necesitan pueden retirarse con Alt + clic.

Puesta en servicio con el PID Self Tuner

La instrucción PID_CP se pone en servicio con el PID Self-Tuner y se determinan los parámetros PID óptimos.

Procedimiento

Para poner en servicio un regulador continuo, proceda del siguiente modo:

1. Llame la instrucción PID_CP en un OB de alarma cíclica e interconecte los parámetros de entrada y salida.
2. Llame la instrucción TUN_EC en el mismo OB de alarma cíclica.
3. Interconecte la instrucción TUN_EC con PID_CP del siguiente modo:
   - TUN_EC.PV con PID_CP.PV
   - TUN_EC.LMN con PID_CP.LMN
   - TUN_EC.SP_OUT con PID_CP.SP_IN o PID_CP.SP_EXT
   - TUN_EC.GAIN con PID_CP.GAIN
   - TUN_EC.TI con PID_CP.TI
   - TUN_EC.TD con PID_CP.TD
   - TUN_EC.TM_LAG con PID_CP.TM_LAG
   - Si se utiliza un regulador de tres puntos, TUN_EC.RATIOFAC con PID_CP.RATIOFAC
   - TUN_EC.QMAN_ON con PID_CP.MAN_ON
4. Haga clic en el símbolo en la instrucción TUN_EC.
   Se abre el editor de puesta en servicio para TUN_EC. El funcionamiento del editor de puesta en servicio se describe en el PID Self-Tuner.

Utilizar PID_ES

Objeto tecnológico PID_ES

El objeto tecnológico PID_ES proporciona un regulador paso a paso para los modos automático y manual. Equivale al bloque de datos de instancia de la instrucción PID_ES. Es posible configurar los siguientes reguladores:

- Regulación de valor fijo con regulador P, PI, PD y PID paso a paso
- Regulador paso a paso con y sin realimentación de posición
- Regulación de valor fijo con control anticipativo
- Regulación en cascada con regulador paso a paso en el lazo subsiguiente
- Regulación de relación de dos lazos
- Regulación de mezcla

Se configuran las funciones siguientes del objeto tecnológico PID_ES:

- Normalización de la señal de valor real
- Supresión de oscilaciones de alta frecuencia en la rama del valor real mediante filtrado (retardando) de la señal del valor real.
- Linealización de funciones cuadráticas del valor real (regulación de caudal con sensores de presión diferencial)
- Llamada de "funciones propias" en la rama del valor real
- Vigilancia de rebase por exceso de los límites de alarma y advertencia del valor real
- Vigilancia de la velocidad de cambio del valor real
- Modificación de la consigna por el generador ramp soak
- Normalización de una señal de consigna externa en las regulaciones de relación
- Llamada de "funciones propias" en la rama de la consigna
- Limitación de la velocidad de cambio de la consigna
- Limitación de los valores absolutos de la consigna
- Filtrado de la señal de error de regulación por una zona muerta
- Vigilancia de rebase por exceso de los límites de alarma y advertencia del error de regulación
- Acciones P, I y D activables individualmente
- Efecto P y D opcional en la rama de realimentación del regulador.
- Control anticipativo
- Aplicación de la realimentación de posición
- Control de la variable manipulada en modo manual desde una programadora o un dispositivo HMI
- Limitación de los valores absolutos de la variable manipulada
- Consideración de tiempos específicos de los actuadores al generar los impulsos

Cálculo de error de regulación

Valor real

Normalización (PV_NORM)

Mediante la casilla de verificación "Activar periferia" se especifica la magnitud de entrada analógica que se normaliza a la unidad física del valor real.

- Activada: la magnitud de entrada es el valor real de periferia (PV_PER)
- Desactivada: la magnitud de entrada es el valor real interno (PV_IN)

Para definir la recta de normalización de forma unívoca, defina los cuatro parámetros siguientes:

- entrada arriba (NM_PIHR)
- entrada abajo (NM_PILR)
- salida arriba (NM_PVHR)
- salida abajo (NM_PVLR)

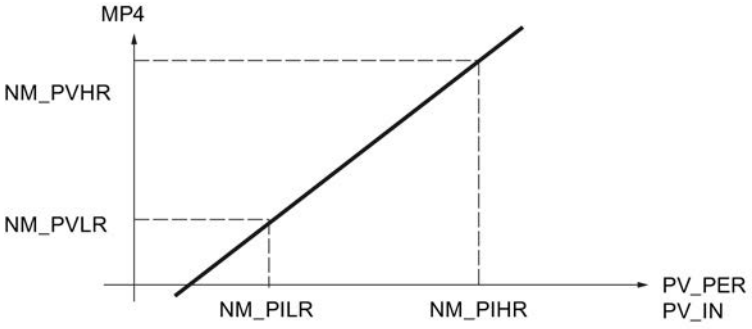
Los parámetros se prestablecen según el tipo de sensor.

El valor real se calcula con la fórmula siguiente a partir del respectivo valor de entrada PV_PER o PV_IN:

MP4 = (PV_PER - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR

O bien:

MP4 = (PV_IN - NM_PILR) × (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR



El valor real se emite en la variable estática MP4.

No se limitan valores ni se comprueban parámetros. Si se introduce el mismo valor para la entrada arriba y la entrada abajo, según las fórmulas anteriores se puede producir una división por cero. Esto no se considera en la instrucción.

Funciones de filtro

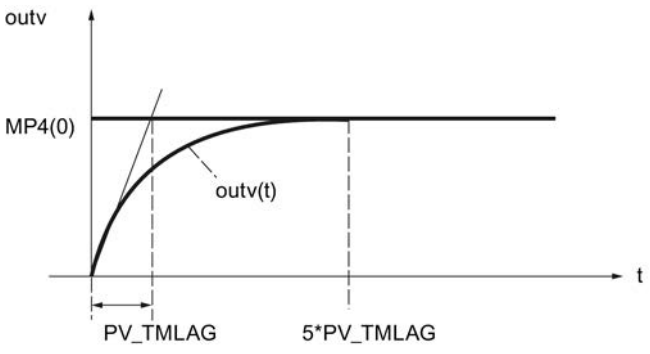
Filtrar magnitud regulada (LAG1ST)

Para filtrar señales de proceso con oscilaciones rápidas superpuestas (Noise) se utiliza un elemento de retardo de primer orden en la rama del valor real. Esta función atenúa en mayor o menor grado la magnitud regulada analógica en función de la constante de tiempo ajustada (PV_TMLAG). De ese modo se suprimen de forma efectiva las señales perturbadoras. Sin embargo, con esta medida también se aumenta en general la constante de tiempo de todo el lazo de regulación, de modo que la regulación es más lenta.

El efecto atenuador se consigue con un algoritmo de retardo de 1er orden.

La respuesta al escalón en el rango de tiempo es:

$outv(t) = MP4(0)\ (1 - e^{-t/PV_TMLAG})$



MP4(0)	la altura del escalón del valor real en la entrada
outv(t)	la magnitud de salida
PV_TMLAG	la constante de tiempo de retardo
t	Tiempo

Filtrado del valor real

Si el filtrado del valor real (LAG1STON = FALSE) no está activado, la entrada de periferia (PV_PER) o la entrada interna (PV_IN) se aplican sin retardo en la rama del valor real.

Tiempo de retardo

El parámetro de entrada PV_TMLAG determina el tiempo de retardo para el valor real. Todo el rango de valores es admisible.

Si PV_TMLAG ≤ 0.5 * CYCLE ya no actúa ningún retardo.

Se requiere un tiempo de muestreo (CYCLE) de menos de una quinta parte del tiempo de retardo para obtener un comportamiento de retardo aproximadamente equivalente al comportamiento analógico.

Calcular raíz cuadrada (SQRT)

La función "Cálculo de la raíz cuadrada" (SQRT) permite linealizar la magnitud regulada. Si el valor real proporcionado por un encóder está presente como magnitud física en relación cuadrática con la magnitud de proceso medida, el desarrollo de la magnitud regulada debe linealizarse antes de su procesamiento en el regulador.

La señal de medición siempre debe linealizarse extrayendo la raíz cuadrada si se realizan mediciones de caudal con diafragmas o tubos de Venturi, etc. La presión diferencial medida (presión efectiva) es así proporcional al cuadrado del caudal.

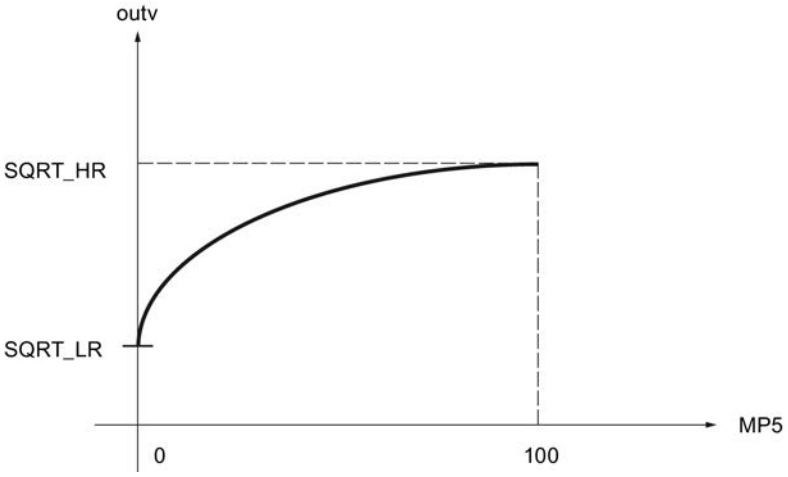
A través de la señal de entrada SQRT_ON = TRUE se activa la función de raíz cuadrada en la rama de valor real. El algoritmo de la función de raíz cuadrada es el siguiente:

outv = SQRT (MP5) × (SQRT_HR – SQRT_LR) / 10.0 + SQRT_LR

outv es el valor de salida normalizado de la función de raíz cuadrada (caudal físico).

Esta fórmula presupone que el valor de entrada de la raíz está normalizado a un rango numérico entre 0 y 100. En la normalización del valor real deben parametrizarse el "Rango

de medida superior" y el "Rango de medida inferior" a 100.0 y 0.0, respectivamente. La raíz de este valor da como resultado un rango de medida entre 0 y 10. Este rango se normaliza mediante el límite superior (SQRT_HR) y el límite inferior del rango de medida físico (SQRT_LR).



Ejemplo de normalización

El valor de entrada PV_IN del regulador es la presión diferencial en mbar:

Principio del rango de medida NM_PILR	Fin del rango de medida NM_PIHR	Ejemplo de valores de PV_IN
20,0 mbar	200,0 mbar	150,0 mbar

Con la función de normalización PV_NORM se calcula la presión diferencial normalizada, seleccionando NM_PVHR = 100.0 y NM_PVLR = 0.0:

MP4 = (PV_IN - NM_PILR) * (NM_PVHR - NM_PVLR) / (NM_PIHR - NM_PILR) + NM_PVLR
= (PV_IN - 20.0 mbar) * (100.0 - 0.0) / (200.0 mbar - 20.0 mbar) + 0.0
= (PV_IN - 20.0 mbar) * 100 / 180.0 mbar

Valor inicial de MP4	Valor final de MP4	Ejemplo de valores de MP4
0.0	100.0	72.222

En este ejemplo no se utiliza ningún filtrado y, por lo tanto, rige lo siguiente: MP5 = MP4.

Para la raíz de la presión diferencial normalizada MP5 resulta:

Valor inicial tras la raíz	Valor final tras la raíz	Ejemplo de valores de SQRT(MP5)
0.0	10.0	8.498

Para el valor de salida normalizado outv de la función de raíz (caudal físico) sigue con SQRT_HR = 20000.0 m^3/h und SQRT_LR = 0.0 m3/h:

outv = SQRT(MP5) * (SQRT_HR - SQRT_LR) / 10.0 + SQRT_LR
= SQRT(MP5) * (20000.0 m^3/h - 0.0 m3/h) / 10.0 + 0.0 m3/h
= 2000.0 m^3/h * SQRT(MP5)

Principio del rango de medida outv	Fin del rango de medida outv	Ejemplo de valores de outv
0,0 m^3/h	20000,0 m^3/h	16996,732 m^3/h

Llamada de FC en la rama de valor real (PVFC)

La inserción de una función específica de usuario en la rama de valor real permite someter la magnitud regulada a un tratamiento de señal antes de su procesamiento en el regulador (p. ej. un retardo de señal o linealización).

Llamar una función

Para insertar una función específica de usuario en la rama del valor real, active "Activar llamada de FC" (PVFC_ON). En el campo de entrada "Número de FC" (PVFC_NBR) se indica el número de la función utilizada. El regulador ejecuta una llamada de la función. En ella no se suministran valores a los parámetros de entrada y salida de la función de usuario. La transferencia de datos debe programarla el usuario en su función.

El valor de salida de la función puede observarse en la variable estática MP6.

Nota

No se comprueba si una función está disponible. Si la función no está disponible, la CPU pasa a STOP con un error interno de sistema.

Alarma

Vigilar los límites del valor real (PV_ALARM)

Si se rebasan por exceso o defecto determinados valores de las magnitudes del proceso (como velocidad, presión, nivel de llenado, temperatura, etc.), en las regulaciones pueden producirse estados no permitidos de proceso o instalación. En estos casos se aplica la vigilancia de los límites del valor real (PV_ALARM) para controlar si el valor real rebasa el

rango de servicio admisible por exceso o por defecto. Los rebases de límite se registran y notifican a fin de activar una respuesta adecuada.

Nota

La vigilancia de los límites del valor real no puede desactivarse. El rebase por exceso de los límites configurados se notifica siempre.

La vigilancia de los límites del valor real (PV_ALARM) vigila la magnitud regulada PV(t) en función de cuatro límites predefinibles en dos zonas de tolerancia. Cuando los límites se alcanzan o rebasan, la función notifica "Advertencia" en el primer límite y "Alarma" en el segundo.

Valores límite

Los valores numéricos de los límites se ajustan en los parámetros de entrada para "Advertencia" y "Alarma". Si el valor real (PV) rebasa estos límites por exceso o defecto, se activan los bits de aviso de salida correspondientes para la alarma superior (QPVH_ALM), la advertencia superior (QPVH_WRN), la advertencia inferior (QPVL_WRN) y la alarma inferior (QPVL_ALM).

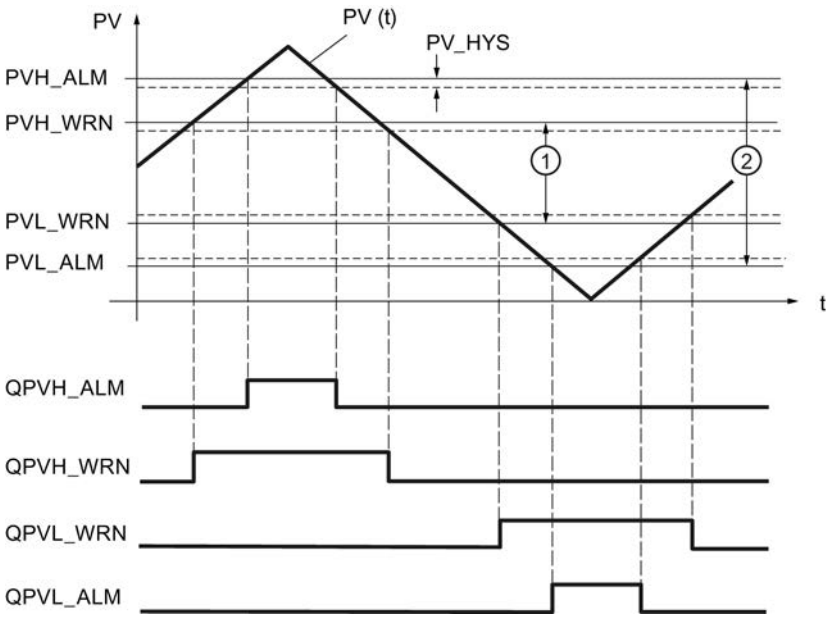
Durante el rearranque se desactivan los bits de aviso.

Los bits de aviso se activan del modo siguiente:

QPVH_ALM	QPVH_WRN	QPVL_WRN	QPVL_ALM	si:	y:
TRUE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_ALM PV ≥ PVH_ALM - PV_HYS
FALSE	TRUE	FALSE	FALSE	PV ↗ PV ↘	PV ≥ PVH_WRN PV ≥ PVH_WRN - PV_HYS
FALSE	FALSE	TRUE	FALSE	PV ↘ PV ↗	PV ≤ PVL_WRN PV ≤ PVL_WRN + PV_HYS
FALSE	FALSE	TRUE	TRUE	PV ↘ PV ↗	PV ≤ PVL_ALM PV ≤ PVL_ALM + PV_HYS

Para que el bloque funcione de manera eficiente debe aplicarse lo siguiente:

alarma superior < advertencia superior < advertencia inferior < alarma inferior



①	1. banda de tolerancia
②	2. banda de tolerancia

Histéresis

Para impedir que los bits de aviso "centelleen" con cambios muy pequeños de la magnitud de entrada o con errores de redondeo, se ajusta una histéresis (diferencia de desactivación) PV_HYS. La magnitud regulada debe superar la histéresis antes de que los bits de aviso se desactiven.

Vigilar la velocidad de cambio del valor real (ROCALARM)

Si el cambio de la magnitud de proceso (p. ej. velocidad, presión, nivel de llenado, temperatura, etc.) es demasiado elevada, en las regulaciones pueden producirse estados no permitidos de proceso o instalación. En estos casos se aplica la vigilancia de la velocidad de cambio del valor real (ROCALARM) para controlar si el valor real rebasa por exceso o por defecto una velocidad de cambio admisible. Los rebases de límite se registran y notifican a fin de activar una respuesta adecuada.

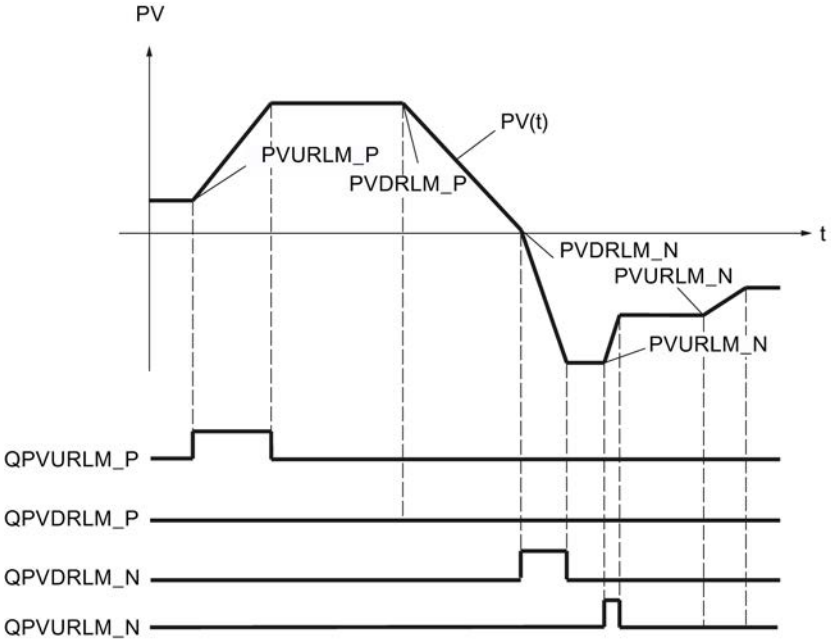
Nota

La vigilancia de la velocidad de cambio del valor real (ROCALARM) no puede desactivarse. El rebase por exceso de los límites configurados se notifica siempre.

La función ROCALARM vigila la magnitud regulada PV(t) respecto a límites admisibles de velocidad de cambio, los cuales pueden ajustarse de forma diferente según el signo correspondiente.

Los valores numéricos de las pendientes límite se ajustan en los parámetros de entrada "Rampa de subida máxima" y "Rampa de bajada máxima" en el rango positivo y negativo de

la magnitud regulada. Las pendientes se refieren a la subida o bajada en porcentaje por segundo. Si la velocidad de cambio de la magnitud regulada supera estos límites, se activan los bits de aviso de salida correspondientes QPVURLMP ... QPVDRLMN.



La identificación de los parámetros de las rampas se realiza según el siguiente esquema:

Parámetro	Cambio de PV
Rampa de subida máxima en el rango positivo (PVURLM_P)	PV > 0 y |PV| ascendente
Rampa de bajada máxima en el rango positivo (PVDRLM_P)	PV > 0 y |PV| descendente
Rampa de subida máxima en el rango negativo (PVURLM_N)	PV < 0 y |PV| ascendente
Rampa de bajada máxima en el rango negativo (PVDRLM_N)	PV < 0 y |PV| descendente

Consigna

Definir la consigna

Seleccione un origen para la consigna. Las posibilidades son las siguientes:

- Consigna interna
  El parámetro de entrada SP_INT se utiliza como consigna
- Generador de consigna
  El parámetro de entrada SP_INT se utiliza como consigna y puede aumentarse o disminuirse mediante las variables estáticas SPUP y SPDN.

- Generador ramp soak
  El parámetro de entrada SP_INT se utiliza como consigna y puede modificarse siguiendo un perfil predefinido.
- Consigna externa
  En los reguladores de mezcla, relación y cascada se utiliza el parámetro de entrada SP_EXT como consigna. La consigna externa puede normalizarse.

En el parámetro de salida SP se muestra la consigna efectiva.

Consulte también

Generador ramp soak (RMP_SOAK) (Página 10127)

Generador de consigna (SP_GEN)

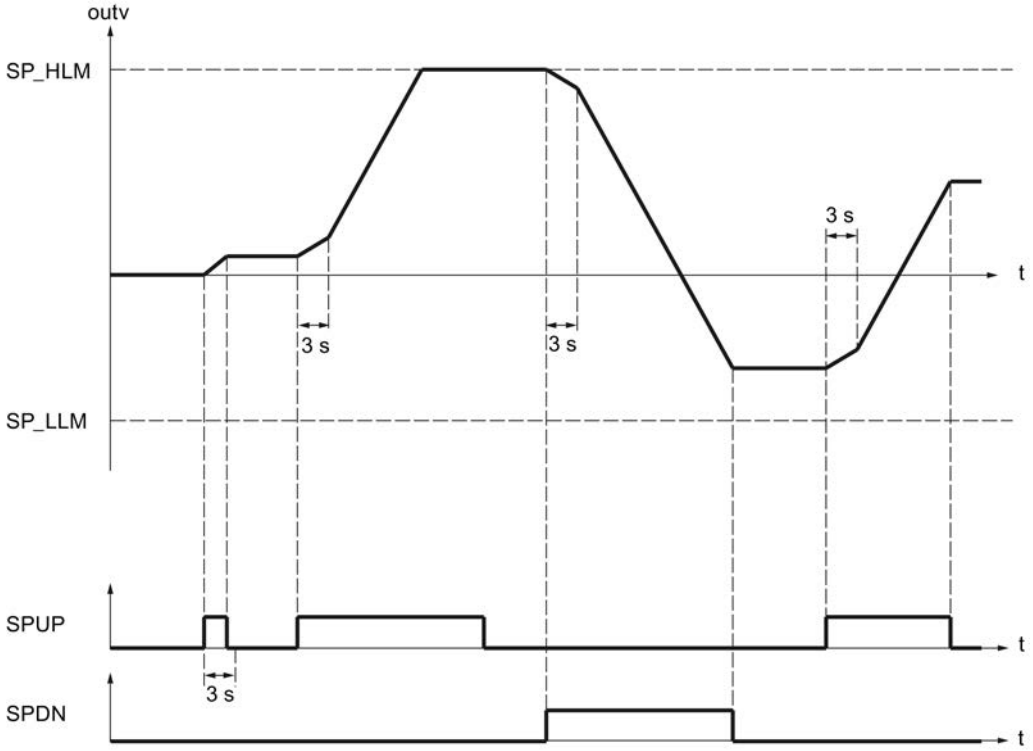
La consigna interna SP_INT se aumenta o disminuye progresivamente utilizando los interruptores SPUP y SPDN. El valor modificado puede observarse en MP1. Se tienen en cuenta los límites configurados de la consigna.

Repercusión de los interruptores SPUP y SPDN

Para poder realizar un cambio graduado finamente, el regulador debe tener un tiempo de muestreo de 100 ms como máximo.

La velocidad de cambio de la magnitud de salida depende de la duración de pulsación de los interruptores SPUP y SPDN así como de los límites ajustados de acuerdo con la relación siguiente:

- durante los primeros 3 segundos después de activar SPUP o SPDN rige:
  doutv/dt = (SP_HLM - SP_LLM) / 100 s
- a continuación
  doutv/dt = (SP_HLM - SP_LLM) / 10 s



Con un tiempo de muestreo de 100 ms y un rango de consigna de -100,0 a 100,0, la consigna cambia en 0,2 por ciclo en los tres primeros segundos, por ejemplo. Si SPUP permanece en TRUE durante más de 3 segundos, la velocidad de cambio aumenta a diez veces el valor, en este caso a 2 por ciclo.

Arranque y funcionamiento del generador de consigna

- Durante el rearranque la salida outv se pone a 0.0.
- Al activar el generador de consigna (SPGEN_ON = TRUE), en la salida outv se muestra primero el valor de señal SPFC_IN. Por este motivo, la conmutación al generador de consigna desde otro modo de operación se produce siempre suavemente. Mientras los interruptores SPUP y SPDN (teclas arriba y abajo) no se activan, SPFC_IN permanece en la salida.

Parámetros de la función SP_GEN

El parámetro de salida outv es un parámetro implícito. Este puede observarse en la herramienta de configuración mediante el punto de medición MP1.

Parámetro	Significado	Rango admisible
SPFC_IN	Consigna de la entrada de FC	Rango técnico
SP_INT	Consigna interna	Rango técnico

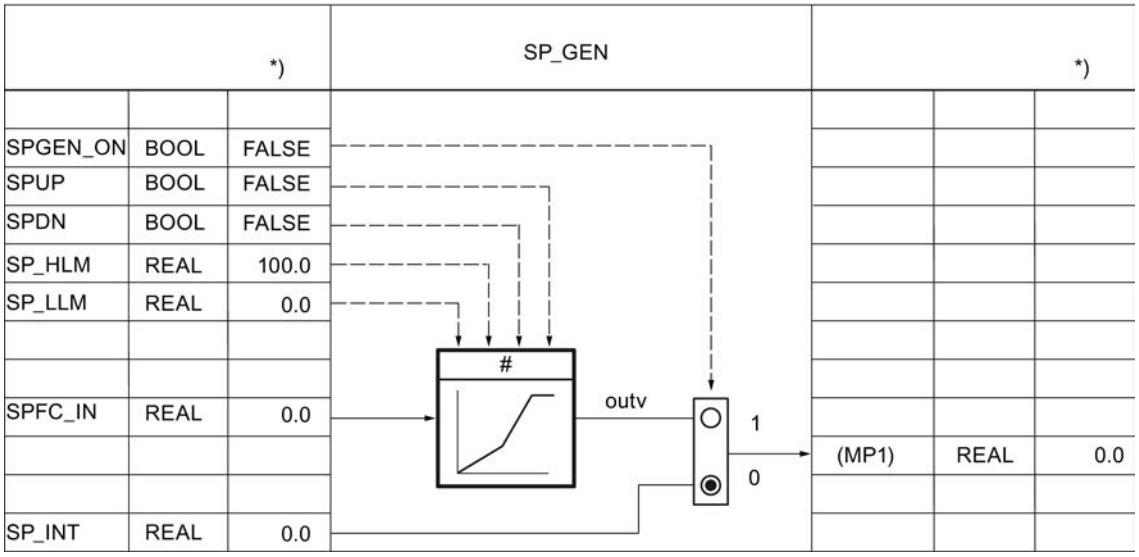


Figura 11-6 Esquema funcional y parámetros del generador de consigna

Generador ramp soak (RMP_SOAK)

Si la consigna SP_INT debe cambiar por sí misma en función del tiempo, p. ej. en la regulación de procesos conforme a un programa de temperatura controlado por tiempo, es posible hacerlo configurando un perfil y activando el generador ramp soak. El cambio de la consigna se calcula a partir de segmentos con un máximo de 256 nodos.

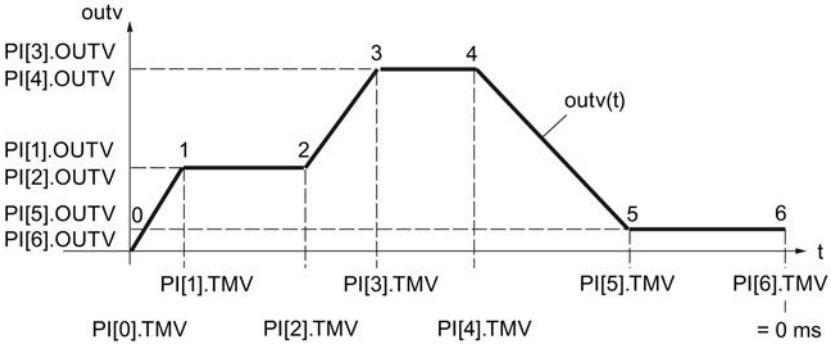
Nota

La función del bloque no comprueba si existe un DB global con el número "DB_NBR" ni tampoco si el parámetro "Número de nodos" NBR_PTS coincide con la longitud del DB. Si la parametrización es incorrecta, la CPU cambia al estado **STOP** con un error de sistema interno.

Funcionamiento del generador ramp soak

El generador ramp soak proporciona una función de la magnitud de salida outv que se ejecuta de acuerdo con una programación determinada. El desarrollo del perfil se define con una serie de nodos de interpolación en un bloque de datos global RMP_SOAK con los valores de tiempo

PI[i].TMV y los correspondientes valores de salida PI[i].OUTV. La figura siguiente muestra un perfil con 6 nodos.

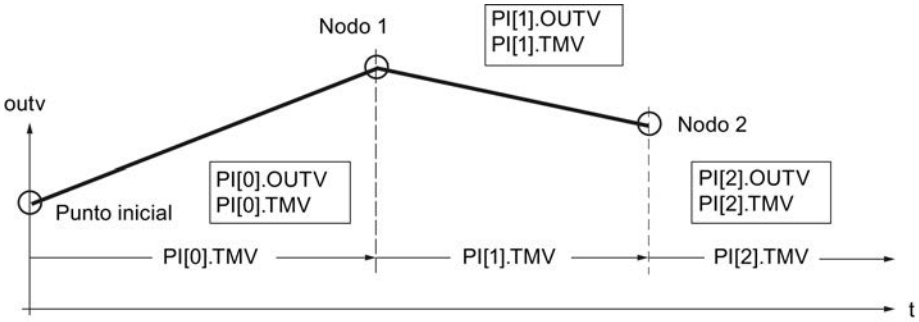


Para el último nodo n, el valor de tiempo PI[n].TMV = 0 ms. El tiempo de procesamiento de un perfil se calcula partiendo del valor inicial y descendiendo hasta cero.

PI[i].TMV indica los intervalos entre nodo y nodo. Entre los nodos se interpola de acuerdo con la función siguiente:

$$outv(t) = PI[n + 1].\ OUTV - \frac{RS_TM}{PI[n].TMV}\ (PI[n + 1].OUTV - PI[n].OUTV)$$

$0 \leq n \leq (NBR_PTS - 1)$



Nota

Nota

Al interpolar la curva de evolución entre los nodos, es posible que el valor de salida se mantenga temporalmente fijo si resulta que el tiempo de muestreo CYCLE es muy pequeño con respecto al tiempo entre los nodos PI[n].TMV. El generador ramp soak no puede linealizar curvas planas debido a la precisión de cálculo de la CPU. Si la curva es demasiado plana, el valor de salida permanece por un tiempo en el nodo correspondiente y, pasado cierto tiempo, se integra en el próximo nodo con la pendiente mínima.

Solución: reduzca el tiempo entre los nodos insertando más nodos. De este modo, la curva emitida se acercará a la curva plana deseada en forma de trapecio.

Activar el generador ramp soak

Con el cambio RMPSK_ON de FALSE a TRUE se activa el generador ramp soak. Una vez alcanzado al último nodo se termina el perfil. Cuando el operador reinicia la función, es necesario poner primero RMPSK_ON a 'FALSE' y a continuación a 'TRUE'.

En el **rearranque**, la salida outv se pone a 0.0 y se determina el tiempo total o el tiempo residual total. En la transición al modo normal, la curva de evolución se procesa inmediatamente desde el punto inicial de acuerdo con el modo de operación ajustado. Si no se desea que esto suceda, el parámetro RMPSK_ON debe ponerse a FALSE en el OB de rearranque.

Funciones del generador ramp soak

La instrucción dispone de las funciones siguientes:

- Activar generador ramp soak para ejecución única
- Predeterminar salida con valor fijo
- Activar repetición cíclica
- Detener generador ramp soak
- Continuar generador ramp soak
- Actualizar tiempo total de procesamiento y tiempo sobrante total

Para ajustar el modo de operación deseado se aplica la siguiente tabla para la valencia de las entradas de control:

Modo de operación	RMPSK_ ON	DFOUT_ ON	RMP_ HOLD	CONT_ ON	CYC_ ON	TUPDT_ ON	Señal de salida OUTV
Activar generador ramp soak	TRUE	FALSE	FALSE	**)	FALSE	**)	OUTV(t) El valor final se mantiene tras el final de procesamiento.
Especificar salida	TRUE	TRUE	**)	**)	**)	**)	DF_OUTV
Activar repetición cíclica	TRUE	FALSE	FALSE	**)	TRUE	**)	OUTV(t) Tras final: inicio automático
Detener generador ramp soak	TRUE	FALSE	TRUE	FALSE	**)	**)	se mantiene el valor actual de OUTV(t) *)
Continuar generador ramp soak	TRUE	FALSE	TRUE	TRUE	**)	**)	OUTV (antiguo) *)
			FALSE				Se sigue con los nuevos valores predeterminados.
Actualizar el tiempo total y el tiempo residual total	**)	**)	**)	**)	**)	FALSE	No influye en OUTV
						TRUE	No influye en OUTV
*) Hasta el próximo nodo la curva no tiene la forma parametrizada por el usuario. **) El modo de operación ajustado en cada caso se ejecuta independientemente de la valencia de las señales de control en los campos marcados con **)							

Especificar la salida del generador ramp soak

Si se activa esta función (DFRMP_ON = TRUE), el valor de salida del generador ramp soak se pone al valor de señal SP_INT o al valor de salida SP_GEN.

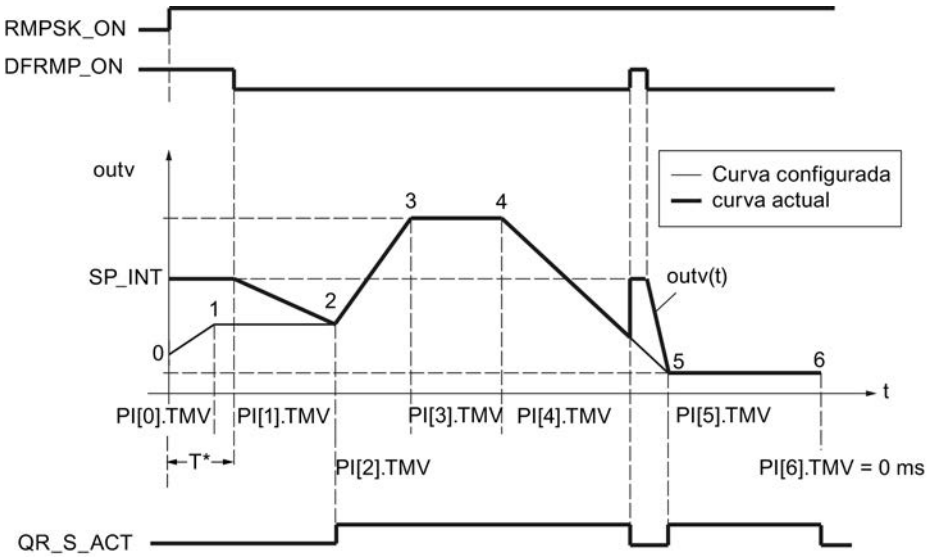
Nota

La asignación previa con la consigna interna solo tiene efecto si el generador ramp soak está activado (RMPSK_ON = TRUE).

Tras la conmutación DFRMP_ON = FALSE, outv se lleva linealmente de la consigna ajustada (p. ej. SP_INT) al valor de salida del número de nodo actual PI[NBR_ATMS].OUTV.

El procesamiento interno de tiempo también continúa con una consigna fija aplicada directamente (RMPSK_ON = TRUE y DFRMP_ON = TRUE).

Al iniciar la curva de evolución RMPSK_ON = TRUE se emite la consigna fija SP_INT hasta que DFRMP_ON cambia de TRUE a FALSE después del tiempo T*.



En este momento ha transcurrido el tiempo PI[0].TMV y parte del tiempo PI[1].TMV. OUTV se lleva de SP_INT a PI[2].OUTV, es decir, a la consigna del nodo 2.

La curva configurada se emite solo a partir del nodo 2. La coincidencia entre la curva configurada y la actual se indica con QR_S_ACT = TRUE. Con un flanco ascendente en DFRMP_ON durante el procesamiento de la curva, el valor emitido OUTV salta a SP_INT sin retardo.

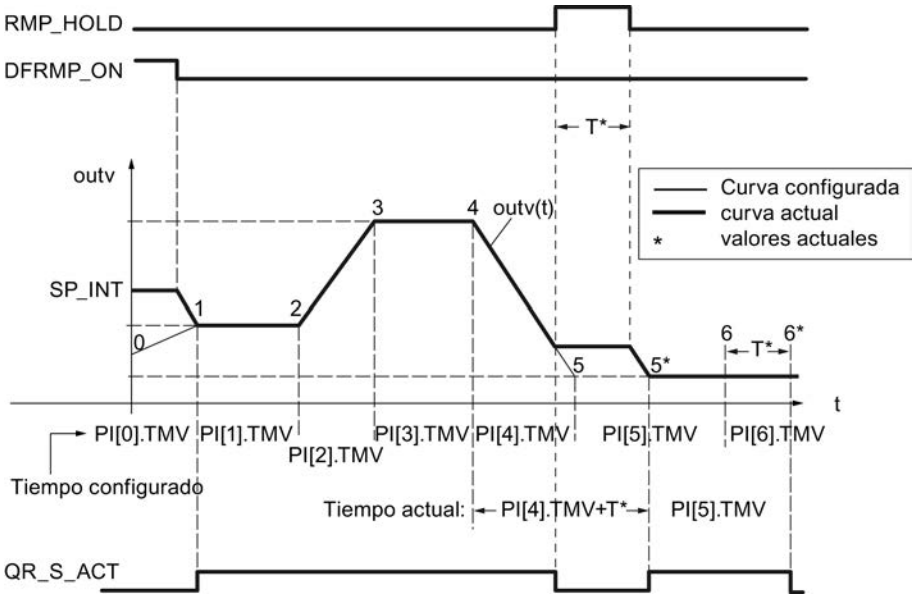
Activar repetición cíclica

Cuando el modo de operación 'Repetición cíclica' (CYC_ON = TRUE) está activado, el generador ramp soak vuelve automáticamente al punto inicial tras emitir el último valor de nodo y se ejecuta de nuevo.

Entre el último nodo y el punto inicial no se interpola. Para una transición suave se debe aplicar lo siguiente: PI[NBR_PTS].OUTV = PI[0].OUTV.

Detener generador ramp soak

Con RMP_HOLD = TRUE se congela el valor de la magnitud de salida (incluido el procesamiento de tiempo). Cuando RMP_HOLD = FALSE se desactiva, PI[x].TMV continúa en el punto de interrupción.



El tiempo de procesamiento de la curva se prolonga por el tiempo T*. La curva presenta el desarrollo configurado desde el nodo hasta el flanco ascendente en RMP_HOLD y desde el nodo 5* hasta el nodo 6*, es decir, la señal de salida QR_S_ACT tiene el valor TRUE.

Si el interruptor CONT_ON está activado, el generador ramp soak detenido continúa en el punto predeterminado TM_CONT con el procesamiento de la curva de desplazamiento.

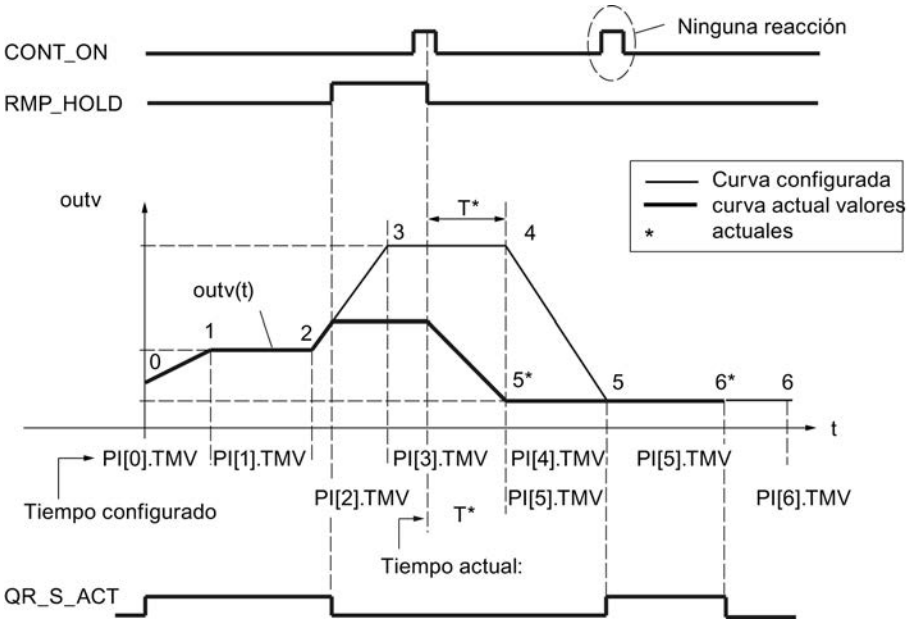
Continuar generador ramp soak

Con CONT_ON se continúa una curva detenida en un punto definido. Si el generador ramp soak no está detenido, CONT_ON no tiene efecto.

Si la entrada de control para continuar CONT_ON = TRUE está activada, el generador ramp soak detenido continúa en el nodo TM_SNBR. TM_CONT determina el tiempo residual que requiere el generador hasta que se alcanza el nodo TM_SNBR.

La figura siguiente muestra la influencia de RMP_HOLD y CONT_ON con la configuración siguiente:

Continuar en el nodo 5 (TM_SNBR = 5) y continuar en T* s: TM_CONT = T*



Los nodos 3 y 4 se suprimen para este ciclo de procesamiento del generador ramp soak. Después del cambio de señal RMP_HOLD de TRUE a FALSE, el desarrollo configurado de la curva solo se vuelve a alcanzar a partir del nodo 5.

La salida QR_S_ACT solo se activa cuando el generador ramp soak procesa la curva de evolución parametrizada por el usuario.

Actualizar el tiempo total y el tiempo residual total

En cada ciclo se actualizan el número actual del nodo NBR_ATMS, el tiempo residual actual hasta alcanzar el nodo RS_TM, el tiempo total T_TM y el tiempo residual total hasta alcanzar el final de la curva RT_TM.

En caso de modificaciones online de PI[n].TMV, cambian el tiempo total y el tiempo residual total de la curva de evolución. Ya que el cálculo de T_TM y RT_TM prolonga considerablemente el tiempo de procesamiento del bloque de función en muchos nodos, este solo se realiza después del rearranque o si TUPDT_ON = TRUE. Los intervalos PI[0 ... NBR_PTS].TMV entre los distintos nodos se suman y se indican en la salida del tiempo total T_TM y del tiempo residual total RT_TM.

Tenga en cuenta que el cálculo de los tiempos totales requiere mucho tiempo de ejecución de la CPU.

Procedimiento

Para configurar el generador ramp soak, proceda del siguiente modo:

1. En la lista "Origen de la consigna" seleccione la entrada "Generador ramp soak".
2. Cree un bloque de datos global RMP_SOAK.
3. Defina los parámetros necesarios para el perfil.
4. Introduzca el número del bloque de datos RMP_SOAK en el campo de entrada "Número de DB".

5. Active o desactive las demás funciones del generador ramp soak.
6. Si ha activado la función "Continuar generador ramp soak", introduzca el nodo y el tiempo en que deba continuar el generador.

Consulte también

Bloque de datos RMP_SOAK (Página 7225)

Normalización

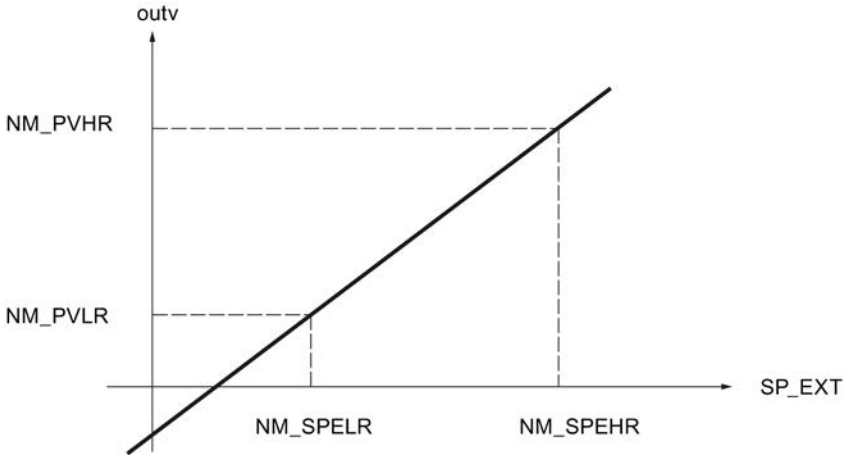
Normalización de una consigna externa

Si la consigna externa no está presente en la unidad física del valor real (p. ej. en % en una cascada de regulador), es necesario normalizar ese valor y su rango de ajuste a la unidad física del valor real. Para ello se utiliza la función "Normalización en la rama de consigna".

La función SP_NORM normaliza una magnitud de entrada analógica. La consigna externa analógica se convierte a la magnitud de salida outv con la recta de normalización. El valor de salida OUTV está accesible en la herramienta de configuración a través del punto de medición MP2.

El valor de salida de la función es efectivo cuando la entrada de control es SPEXT_ON = TRUE. Para definir la recta de normalización de forma unívoca, defina los cuatro parámetros siguientes:

- Entrada arriba: SP_EXT: NM_SPEHR
- Entrada abajo: SP_EXT: NM_SPELR
- Salida arriba: NM_PVHR (este valor se introduce en la función de normalización del valor real.)
- Salida abajo: NM_PVLR (este valor se introduce en la función de normalización del valor real.)



El valor de salida outv se calcula con la fórmula siguiente a partir del correspondiente valor de entrada SP_EXT:

outv = (SP_EXT – NM_SPELR) x (NM_PVHR – NM_PVLR) / (NM_SPEHR – NM_SPELR) + NM_PVLR

Para el caso especial de una función de raíz aplicada en la rama de valor real rigen como límite superior e inferior del valor de salida outv los valores de normalización de la función de raíz (SQRT_HR y SQRT_LR).

En ese caso, el valor de salida outv de la función de normalización se calcula con la fórmula siguiente a partir del correspondiente valor de entrada SP_EXT:

outv = (SP_EXT – NM_SPELR) x (SQRT_HR – SQRT_LR) / (NM_SPEHR – NM_SPELR) + SQRT_LR

Dentro de la función no se limitan los valores ni se comprueban los parámetros. Si introduce el mismo valor para NM_SPEHR y NM_SPELR, según las fórmulas anteriores se puede producir una división entre cero. Esto no se considera en la instrucción.

Factor de mezcla/relación

En una regulación de relación, las consignas de los lazos de regulación se adaptan unas a otras de forma que la relación ajustada se corresponde con la relación efectiva de los valores reales.

En una regulación de mezcla se ajusta la proporción deseada de cada uno de los componentes. La suma de los factores de mezcla FAC debe dar 1.

Procedimiento

Para configurar la normalización, proceda del siguiente modo:

1. Introduzca los valores de la normalización en los campos de entrada "Entrada arriba" y "Entrada abajo".
2. Introduzca un factor de mezcla/relación (FAC).

Límites

Llamada de FC en la rama de consigna

La inserción de una función específica de usuario en la rama de consigna permite someter una referencia predeterminada de forma externa a un procesamiento de señal (p. ej. un retardo de señal o linealización) antes de aplicarla en el regulador.

Si se activa la llamada de FC en la rama de consigna, se llama una función específica de usuario. El regulador ejecuta una llamada de la FC. En ella no se suministran valores a los parámetros de entrada y salida de la función de usuario ya existentes. Por ello, la transmisión de datos debe programarse manualmente con S7-AWL en la FC de usuario. La consigna acondicionada se observa en MP3.

En ese caso, el valor de SPFC_ON determina si en el canal de la consigna se inserta en este punto una función programada libremente en forma de FC estándar (p. ej. una curva característica) o si la consigna continúa procesándose sin ninguna influencia de ese tipo.

Nota

La instrucción no comprueba si hay una función. Si la función no está disponible, la CPU pasa a STOP con un error interno de sistema.

Procedimiento

Para llamar una función en la rama de la consigna, proceda del siguiente modo:

1. Active la casilla de verificación "Activar llamada de FC".
2. En el campo de entrada "FC", indique el número de la función que se va a llamar.

Limitación de la velocidad de cambio de la consigna

Las funciones de rampa en la rama de la consigna se utilizan en aquellos casos en que el proceso no soporta cambios bruscos de la señal manipulada, ya que un cambio abrupto de la consigna generalmente provoca también un escalón en la variable manipulada del regulador. Estos cambios abruptos de la variable manipulada deben evitarse p. ej. en aquellos casos en que entre el motor regulado y la carga que debe accionarse hay interpuesto un engranaje, y una elevación muy rápida del número de revoluciones del motor derivaría en una sobrecarga del engranaje.

La función SP_ROC limita la velocidad de cambio de las consignas procesadas en el regulador, ya sea por separado para la rampa de subida y bajada, como por separado para los rangos positivo y negativo de la referencia.

En las cuatro entradas SPURLM_P, SPDRLM_P, SPURLM_N y SPDRLM_N se introducen las limitaciones correspondiente para los aumentos de la función de rampa en los rangos positivo y negativo de la referencia. Las rampas de subida se refieren a la subida o bajada por segundo. Los cambios de consigna más rápidos se retardan a estas velocidades límite.

Si p. ej. SPURLM_P se parametriza a 10.0 [rango de valores tec./s], en cada ciclo de muestreo se suman los siguientes valores al "valor antiguo" de outv mientras inv sea > outv:

Tiempo de muestreo	1 s	$outv_{antiguo}$ + 10
	100 ms	$outv_{antiguo}$ + 1
	10 ms	$outv_{antiguo}$ + 0.1

La siguiente figura muestra el funcionamiento del procesamiento de la señal a partir de un ejemplo. Las funciones de escalón en la entrada inv(t) se traducen en funciones de rampa en la salida outv(t).

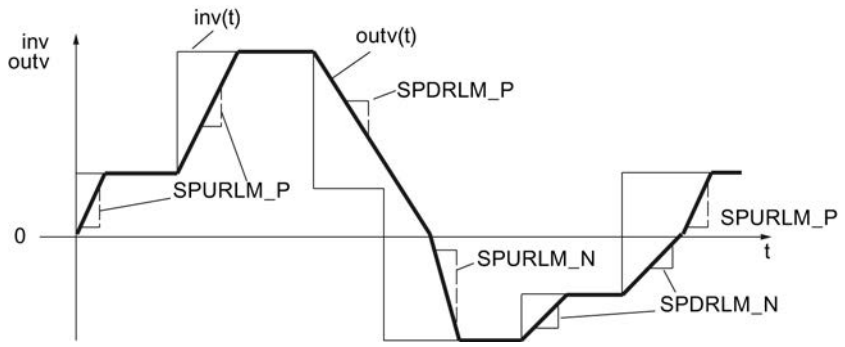


Figura 11-7 Limitación de la velocidad de cambio de la referencia SP(t)

No se notifica el alcance de los límites de la pendiente.

Procedimiento

Para limitar la velocidad de cambio, proceda del siguiente modo:

1. Active la casilla de verificación "Limitar la velocidad de cambio".
2. Defina los siguientes valores tanto para el rango positivo como para el negativo:
   - Rampa de subida de consigna
   - Rampa de bajada de consigna

Limitación del valor absoluto de la referencia

El rango de ajuste de la referencia determina el rango dentro del cual puede oscilar la referencia, es decir, el margen en el que se mueve el proceso dentro de los valores de estado admisibles.

Por ello, para evitar estados de proceso críticos o no permitidos, Standard PID Control limita el rango de ajuste de la referencia en la rama de consigna por arriba y por abajo (SP_LIMIT).

La función SP_LIMIT limita la consigna SP al valor inferior y superior predeterminable SP_LLM y SP_HLM mientras la magnitud de entrada INV esté fuera de dichas limitaciones. Dado que la función no puede desactivarse, en la configuración siempre hay que tener en cuenta la indicación de los límites inferior y superior.

Los valores numéricos de los límites se ajustan en los parámetros de entrada para la limitación inferior y superior. Si la magnitud de entrada inv(t) rebasa los límites, se emiten las indicaciones correspondientes a través de las salidas de aviso.

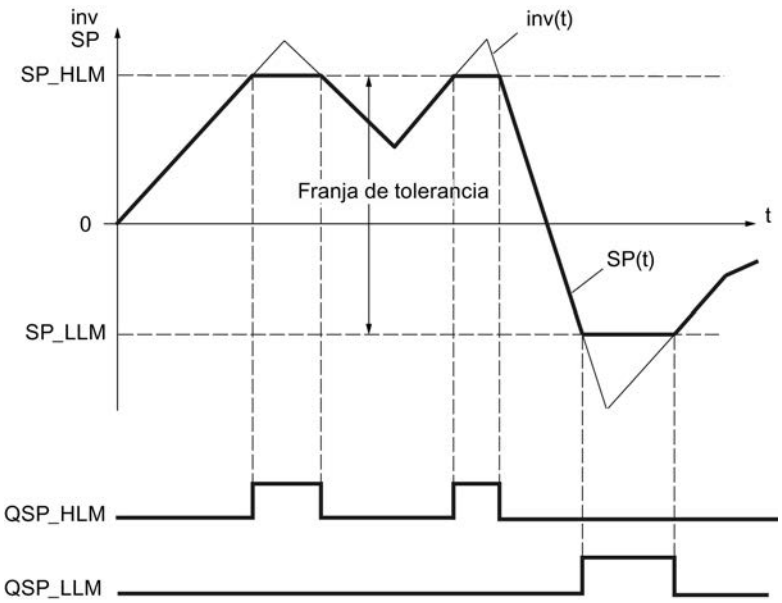


Figura 11-8 Limitación de valor absoluto de la referencia SP(t)

Durante el rearranque se desactivan los bits de aviso.

Los bits de aviso se activan del modo siguiente:

SP =PVH_ALM	QSP_HLM	QSP_LLM	si:
SP_HLM	TRUE	FALSE	inv ≥ SP_HLM
SP_LLM	FALSE	TRUE	inv ≤ SP_LLM
inv	FALSE	FALSE	SP_HLM < inv < SP_LLM

Procedimiento

Para limitar el valor absoluto de la referencia, proceda del siguiente modo:

1. En el campo de entrada "Límite superior", indique el límite superior absoluto de la consigna (SP_HLM).
2. En el campo de entrada "Límite inferior", indique el límite inferior absoluto de la consigna (SP_LLM).

Error de regulación

Zona muerta

En caso de superposición de la magnitud regulada o la referencia con una señal perturbadora de mayor frecuencia, en un regulador con ajuste óptimo el ruido también es efectivo en la salida del regulador. En una amplificación elevada del regulador y con derivación activada, por ejemplo, esto puede provocar importantes oscilaciones en la magnitud manipulada.

Debido al aumento de la frecuencia de conmutación (regulador paso a paso), eso deriva en un desgaste elevado del actuador.

En estado forzado del regulador, la función reduce el ruido en la señal del error de regulación, disminuyendo así la oscilación no deseada de la salida de regulador.

La zona muerta puede desactivarse.

Filtrado de señal por función de zona muerta

La función DEADBAND suprime pequeñas oscilaciones de la magnitud de entrada dentro de un rango parametrizable alrededor de un punto cero definido. Fuera de esta franja de oscilaciones, el error de regulación ER aumenta o cae proporcionalmente a la magnitud de entrada. El ancho de la zona no efectiva (zona muerta) puede definirse con el parámetro DEADB_W. El ancho de zona muerta solo debe tener valores positivos.

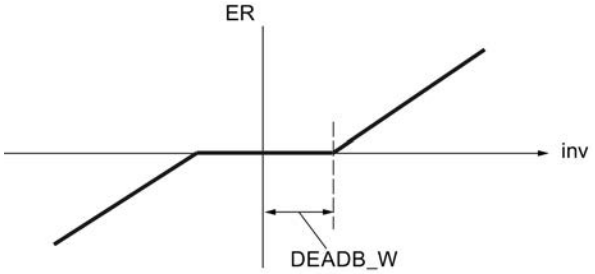
Si la magnitud de entrada se encuentra dentro de la zona muerta, se deposita el valor 0 (error de regulación = 0) en la salida. Solo una vez que la magnitud de entrada abandona este rango de insensibilidad, la salida aumenta o cae la parte correspondiente a los mismos valores de la magnitud de entrada inv. El resultado es una falsificación de la señal transmitida, también fuera de la zona muerta. Sin embargo, esto se tolera para evitar escalones en los límites de la zona muerta (véase la siguiente figura). La falsificación corresponde al valor DEADB_W, por lo que es fácilmente controlable.

La función DEADBAND trabaja según las siguientes funciones:

(ER) = inv + DEADB_W para inv < - DEADB_W

(ER) = 0 para -DEADB_W ≤ inv ≤ + DEADB_W

(ER) = inv - DEADB_W para inv > + DEADB_W



Las repercusiones del filtrado de señales se observan en el parámetro de salida ER.

Alarma

Las desviaciones excesivas de la magnitud regulada respecto a la consigna ajustada pueden provocar estados de proceso no deseados. En estos casos se utiliza la función ER_ALARM para vigilar que el error de regulación no rebase el rango de servicio permitido por exceso ni

por defecto. ER_ALARM registra y notifica los rebases de límite ocurridos, para permitir una respuesta adecuada.

Nota

La vigilancia de los límites del error de regulación no puede desactivarse. El rebase por exceso de los límites configurados se notifica siempre.

La función ER_ALARM vigila el valor del error de regulación ER(t) en función de cuatro límites predefinibles en dos zonas de tolerancia. Cuando los límites se alcanzan o rebasan, la función notifica primero "Advertencia" y a continuación "Alarma" en el segundo límite.

Valores límite

Los valores numéricos de los límites se ajustan en los parámetros de entrada para "Advertencia" y "Alarma". Si el error de regulación (ER) rebasa estos límites por exceso o defecto, se activan los bits de aviso de salida correspondientes QERN_ALM ... QERP_ALM.

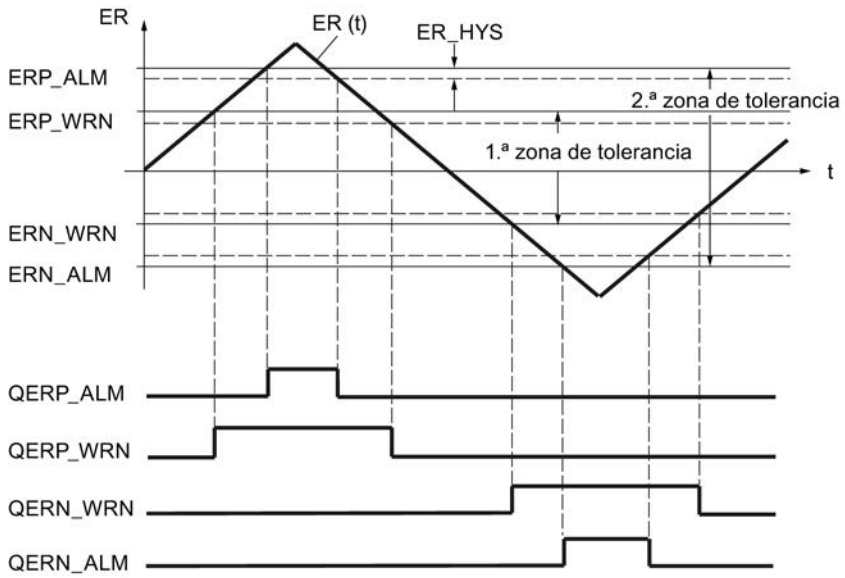
Durante el rearranque se desactivan los bits de aviso.

Los bits de aviso se activan del modo siguiente:

QERP_ALM	QERP_WRN	QERN_WRN	QERN_ALM	si:	y:
TRUE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_ALM ER ≥ ERP_ALM - ER_HYS
FALSE	TRUE	FALSE	FALSE	ER ↗ ER ↘	ER ≥ ERP_WRN ER ≥ ERP_WRN - ER_HYS
FALSE	FALSE	TRUE	FALSE	ER ↘ ER ↗	ER ≤ ERP_WRN ER ≤ ERP_WRN + ER_HYS
FALSE	FALSE	TRUE	TRUE	ER ↘ ER ↗	ER ≤ ERP_ALM ER ≤ ERP_ALM + ER_HYS

Para que el bloque funcione de manera eficiente debe aplicarse lo siguiente:

alarma superior < advertencia superior < advertencia inferior < alarma inferior



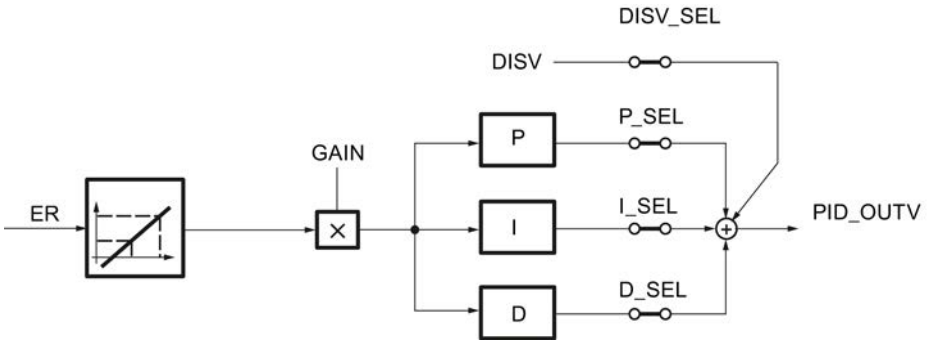
Histéresis

Para evitar que los bits de aviso "centelleen" con cambios muy pequeños de la magnitud de entrada o con errores de redondeo, se ajusta una histéresis (diferencia de desactivación) ER_HYS. El error de regulación debe superar esta histéresis antes de que los bits de aviso se desactiven.

Regulador

Algoritmo de regulación

En el ciclo del tiempo de muestreo configurado se calcula la magnitud manipulada del regulador continuo a partir del error de regulación en el algoritmo de posición PID. El regulador tiene una estructura puramente paralela. Las acciones proporcional, integral o diferencial se pueden desconectar individualmente.

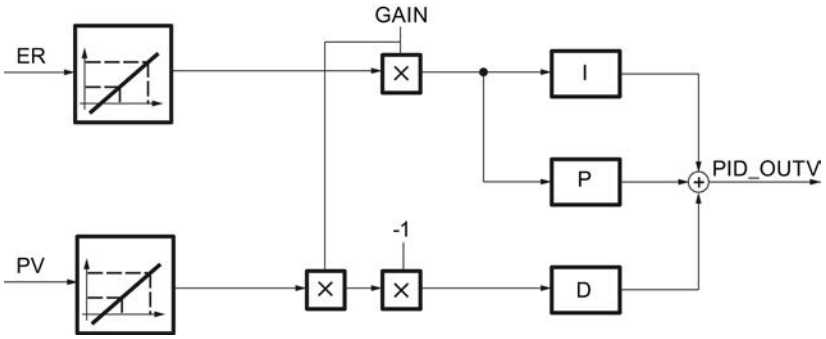


Control anticipativo

A la señal de salida del regulador PID_OUTV se le puede aplicar adicionalmente una magnitud perturbadora DISV. La activación o desactivación se realiza en la ventana PID de la herramienta de configuración a través del conmutador de estructura DISV_SEL o mediante "Magnitud perturbadora ON".

Acción P y D en la rama de realimentación

En la estructura paralela, cada acción del algoritmo regulador recibe el error de regulación como señal de entrada. La magnitud manipulada es modificada directamente por la acción P y la acción D con escalones de la consigna. Sin embargo, otra estructura del regulador en la que la formación de la acción P y la acción D se traslada a la realimentación, garantiza una evolución suave de la magnitud manipulada en caso de variaciones del escalón de la referencia. En esta estructura la acción I procesa el error de regulación como señal de entrada, en la acción P y en la acción D solo se conecta la magnitud de regulación negativa (factor = -1).



Normalización de las magnitudes de entrada ER y PV

Las magnitudes de entrada ER y PV del regulador PID se normalizan antes del procesamiento del regulador en el rango de 0 a 100, conforme a las fórmulas siguientes:

- con función de raíz desactivada (SQRT_ON = FALSE):
  - $ER_{normalizado}$ = ER * 100.0 / (NM_PVHR – NM_PVLR)
  - $PV_{normalizado}$ = (PV - NM_PVLR) * 100.0 / (NM_PVHR – NM_PVLR)
- con función de raíz activada (SQRT_ON = TRUE):
  - $ER_{normalizado}$ = ER * 100.0 / (SQRT_HR – SQRT_LR)
  - $PV_{normalizado}$ = (PV - SQRT_LR) * 100.0 / (SQRT_HR – SQRT_LR)

La normalización se lleva a cabo para que la ganancia proporcional GAIN del regulador PID pueda introducirse sin dimensiones. De este modo, si cambia el límite superior e inferior del rango de medida físico (p. ej. de bar a mbar), no es necesario cambiar el factor de ganancia.

Las magnitudes de entrada normalizadas $ER_{normalizado}$ y $PV_{normalizado}$ no pueden observarse.

Consulte también

Estructuras de regulador (Página 10142)

Acción P (Página 10145)

Acción D (Página 10149)

Acción I (Página 10146)

Estructuras de regulador

Uso y parametrización del regulador PID

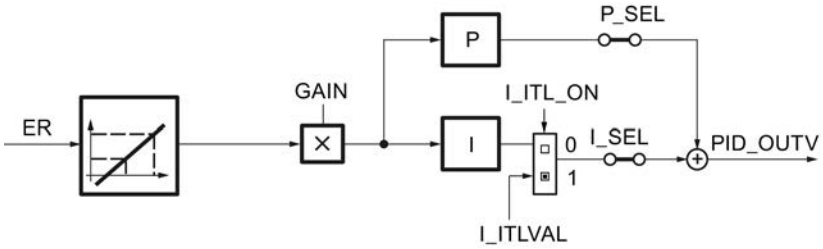
La mayor parte de los procesos regulados que se dan en la industria del procesamiento y los procesos puede regularse con la función de regulador PI/PID de Standard PID Control. Solo en casos especiales se requieren métodos y medidas adicionales para dominar la regulación en cuestión.

No obstante, un gran problema práctico suele ser la parametrización del regulador PI/PID, es decir, encontrar los valores de ajuste "correctos" para los parámetros del regulador. La calidad de esta parametrización es determinante para la efectividad de la regulación PID, y requiere una gran experiencia práctica, conocimientos especializados o bien mucho tiempo.

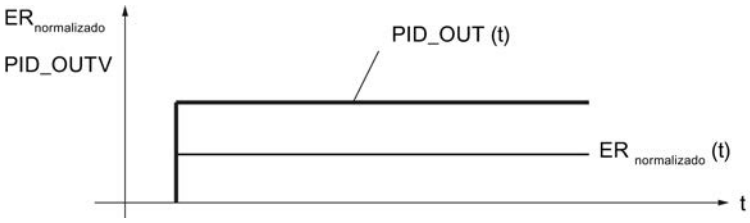
Regulación P

En el regulador P están desconectadas las acciones I y D. (I_SEL y D_SEL = FALSE). Eso significa que con un error de regulación ER = 0, la señal de salida también es OUTV = 0. Para ajustar un punto de trabajo ≠ 0, es decir, un valor numérico para la señal de salida con un error de regulación cero, hay que activar la rama I (véase la figura siguiente).

En la acción I se puede predeterminar un punto de trabajo ≠ 0 para el regulador P ajustando el valor de inicialización I_ITLVAL en correspondencia. Para ello hay que activar los conmutadores 'I_ITL_ON' y 'I_SEL' = TRUE.



La respuesta indicial del regulador P en el rango de tiempo es:
PID_OUTV (t) = I_ITLVAL + GAIN * ER $_{normalizado}$ (t)



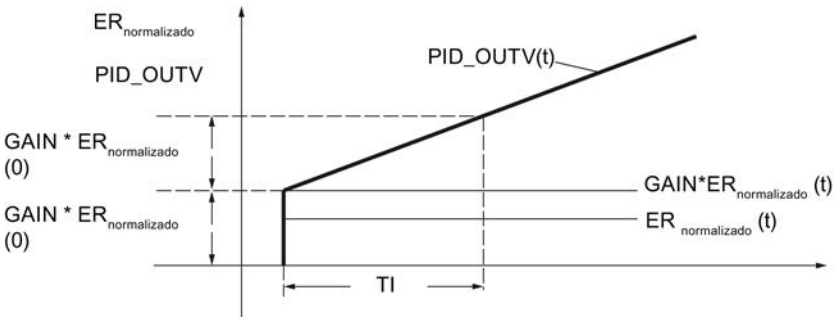
PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
I_ITLVAL	es el punto de trabajo del regulador P
GAIN	es la ganancia de regulador
$ER_{normalizado}(t)$	es el error de regulación normalizado

Regulación PI

En el regulador PI está desconectada la acción D. (D_SEL = FALSE). Un regulador PI modifica la magnitud de salida PID_OUTV con la acción I hasta que el error de regulación es ER = 0. Sin embargo, esto solo es aplicable cuando la magnitud de salida no rebasa los límites del rango manipulado.

La respuesta indicial en el rango de tiempo es:

$$PID_OUTV(t) = GAIN * ER_{normalizado}(0)\left(1 + \frac{1}{TI} * t\right)$$



PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
GAIN	es la ganancia de regulador
$ER_{normalizado}(0)$	es la altura del escalón del error de regulación normalizado
TI	Constante de tiempo de integración

Regulación PD

En el regulador PD está desconectada la acción I (I–SEL = FALSE). Eso significa que con un error de regulación ER = 0, la señal de salida también es OUTV = 0. Para ajustar un punto de trabajo ≠ 0, es decir, un valor numérico para la señal de salida con un error de regulación cero, hay que activar la rama I (véase también el diagrama de bloques de la regulación P).

En la acción I se puede predeterminar un punto de trabajo ≠ 0 para el regulador PD ajustando el valor de inicialización I_ITLVAL en correspondencia. Para ello hay que activar los conmutadores 'I_ITL_ON' e 'I_SEL' TRUE .

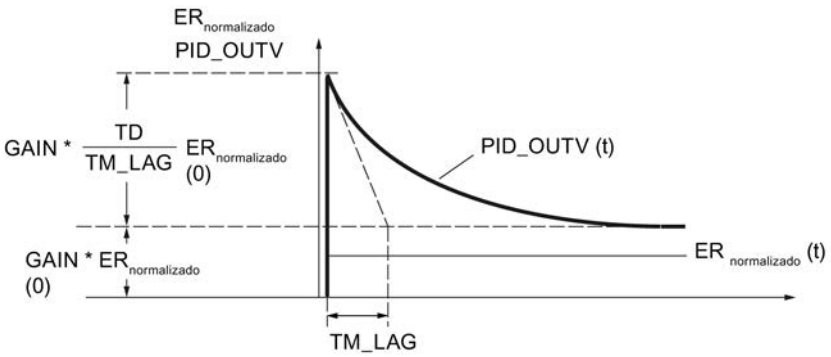
El regulador PD reproduce la magnitud de entrada ER(t) proporcionalmente sobre la señal de salida y añade la acción D creada por la derivación de ER(t), que se calcula con el doble de precisión según la regla del trapecio (aproximación de Padé). El comportamiento de tiempo se determina mediante la constante derivativa (tiempo de derivación) TD.

Para el filtrado de señales y la supresión de señales perturbadoras se ha integrado un retardo de 1er orden (tiempo ajustable: TM_LAG) en el algoritmo para formar la acción D.

Generalmente es suficiente un valor reducido para TM_LAG para conseguir el resultado deseado. Si se parametriza TM_LAG ≤ CYCLE/2 el retardo está desactivado.

La respuesta indicial en el rango de tiempo es:

$$PID_OUTV(t) = GAIN * ER_{normalizado}(0)\left(1 + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}}\right)$$



PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
GAIN	es la ganancia de regulador
$ER_{normalizado}(0)$	es la altura del escalón del error de regulación normalizado
TD	Tiempo derivativo
TM_LAG	Tiempo de retardo

Regulación PID

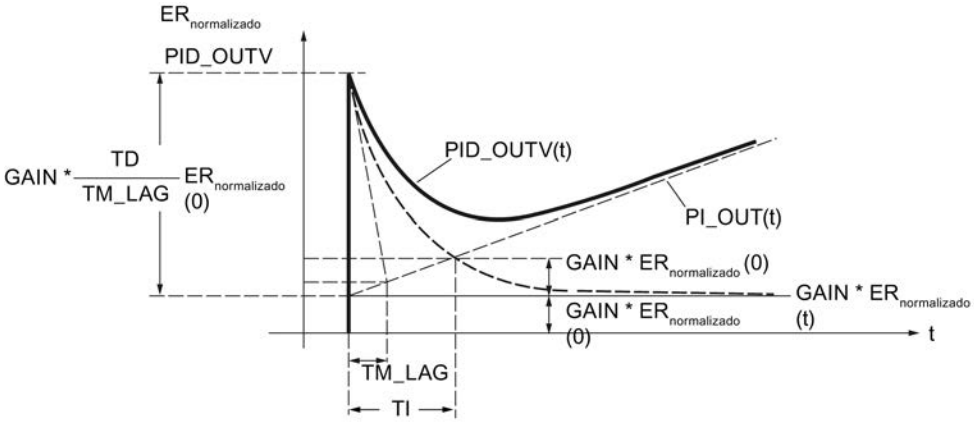
En el regulador PID están activadas las acciones P, I y D (P–SEL = TRUE, I–SEL = TRUE, D_SEL = TRUE). Un regulador PID modifica la magnitud de salida PID_OUTV mediante la acción I hasta que el error de regulación es ER = 0. Sin embargo, esto solo es aplicable cuando la magnitud de salida no rebasa los límites del rango manipulado. Si se rebasan los límites de la variable manipulada, la acción I mantiene el valor alcanzado en el límite en (Anti-Reset-Windup).

El regulador PID reproduce la magnitud de entrada normalizada $ER_{normalizado}(t)$ proporcionalmente sobre la señal de salida y añade las acciones creadas por la derivación e integración de $ER_{normalizado}(t)$, que se calculan con el doble de precisión según la regla de trapecio (aproximación Padé). La respuesta de tiempo se determina por el tiempo de derivación (tiempo de acción derivativa) TD y el tiempo de integración (tiempo de acción integral) TI.

Para el filtrado y la supresión de señales perturbadoras se ha integrado un retardo de 1er orden (constante de tiempo ajustable: TM_LAG) en el algoritmo para formar la acción D. Generalmente es suficiente un valor reducido para TM_LAG para conseguir el resultado deseado. Si se parametriza TM_LAG ≤ CYCLE/2 el retardo está desactivado.

La respuesta indicial en el rango de tiempo es:

$$PID_OUTV(t) = GAIN * ER_{normalizado}(0)\left(1 + \frac{1}{TI} * t + \frac{TD}{TM_LAG} * e^{-\frac{t}{TM_LAG}}\right)$$



PID_OUTV(t)	es la magnitud manipulada en modo automático del regulador
GAIN	es la ganancia de regulador
$ER_{normalizado}(0)$	es la altura del escalón del error de regulación normalizado
TI	Constante de tiempo de integración
TD	Tiempo derivativo
TM_LAG	Tiempo de retardo

Nota

Si TD cambia, conviene adaptar TM_LAG de la forma correspondiente.

Recomendación: 5 ≤ (TM / TM_LAG) ≤ 10

Consulte también

Algoritmo de regulación (Página 10137)

Acción P (Página 10145)

Acción D (Página 10149)

Acción P

Ganancia proporcional

El signo de la ganancia proporcional determina el sentido de acción del regulador continuo.

- Ganancia proporcional positiva
  error de regulación ascendente → variable manipulada ascendente
- Ganancia proporcional negativa
  error de regulación ascendente → variable manipulada descendente

Aplicar la acción P en rama de realimentación

Si la acción P está aplicada en la rama de realimentación, solo el valor real tendrá efecto sobre la acción P. Por ello, la magnitud manipulada – igual que en el caso de escalón de la consigna – en un cambio de automático a manual no realiza ningún escalón por la acción P: la conmutación es suave.

Consulte también

Algoritmo de regulación (Página 10137)

Acción I

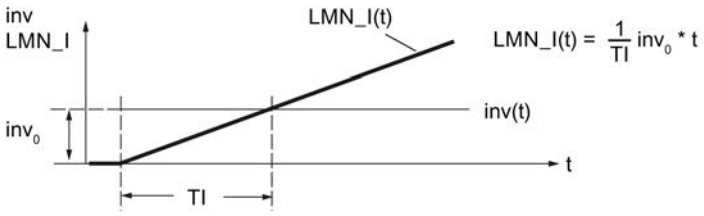
Función de la acción I

La acción I hace que, reajustando el punto de operación, el error de regulación pueda convertirse en 0 para cualquier valor de la magnitud manipulada.

La acción I genera una señal de salida cuya velocidad de cambio es proporcional al cambio del valor absoluto de la magnitud de entrada. La respuesta temporal viene determinada por el tiempo integrativo TI.

La función de transferencia en el rango de tiempo es:

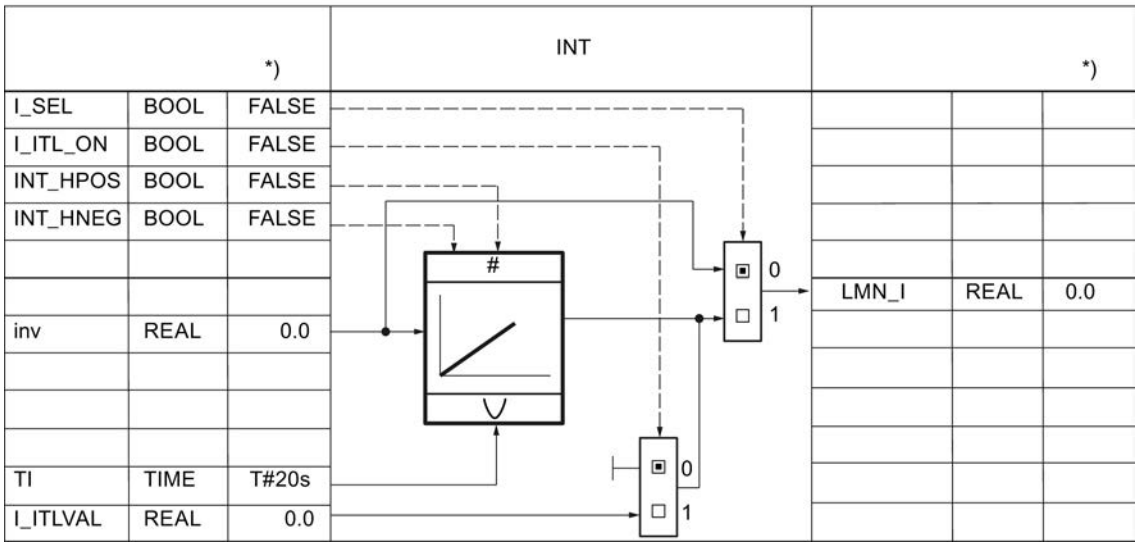
$$OUTV(t) = \frac{1}{TI} \int inv(t)\, dt$$



LMN_I(t)	la magnitud de salida de la acción I
inv_0	la altura del escalón en la entrada de la acción I
TI	Tiempo integrativo

La acción I se observa en la variable estática LMN_I.

Parámetro	Significado	Rango admisible
TI	Tiempo integrativo	≥ 5 * CYCLE
I_ITLVAL	Valor de inicialización para la acción I	-100,0 a +100,0 [%]

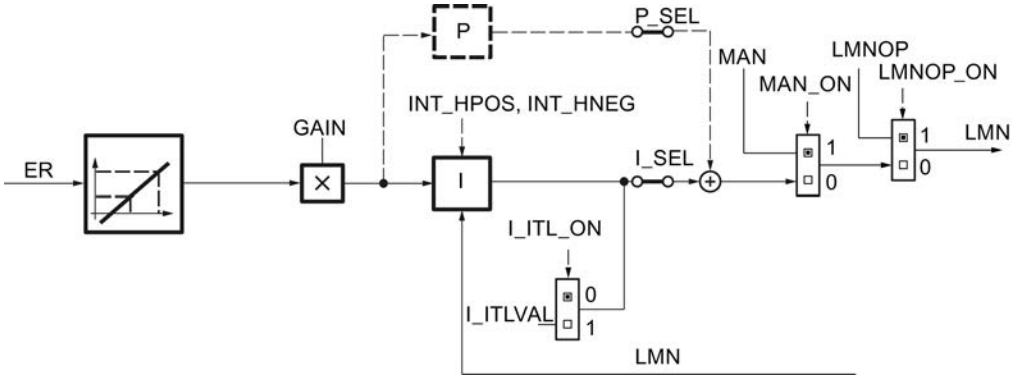


*) Ajuste predeterminado al crear el objeto tecnológico

En estado desactivado (I_SEL = FALSE) la acción I, es decir, la memoria interna y la variable estática LMN_I, se pone a cero.

La salida y la memoria de la acción I están limitados por los límites absolutos de la variable manipulada LMN_HLM y LMN_LLM (Anti Reset Wind–up).

En caso de influir en la señal manipulada en modo manual, es decir, cuando MAN_ON, LMNOP_ON o CAS_ON = TRUE, el valor de la memoria interna de la acción I se corrige según el valor LMN.



Conmutación de modo manual a automático

La acción I se ajusta en modo manual de modo que durante la conmutación de manual a automático y partiendo de la posición actual, la magnitud manipulada realiza un escalón cuya altura queda determinada por la acción P. La altura del escalón de la acción P equivale al cambio de la magnitud manipulada cuando hay un escalón de consiga del valor real actual a la consigna actual.

Sin embargo, si la acción P está aplicada en la rama de realimentación, solo el valor real tendrá efecto en la acción P. Por ello, en una conmutación de manual a automático, la magnitud manipulada no realiza ningún escalón por la acción P; la conmutación es suave.

Rangos admisibles para el tiempo integrativo TI y CYCLE

Debido a la precisión limitada de los números REAL calculados en la CPU, es posible que se produzca el efecto siguiente durante la integración: si se ha elegido un tiempo de muestreo CYCLE de la regulación demasiado pequeño en comparación con el tiempo integrativo TI y si el valor de entrada inv de la acción I es pequeño en comparación con su valor de salida OUTV, la acción I no se activa y permanece en su valor de salida momentáneo.

Este efecto puede evitarse si se considera la siguiente regla de dimensionado en la configuración:

$CYCLE > 10^{-4} \times TI$

De este modo, la acción I reacciona incluso a cambios de los valores de entrada situados en el rango de una millonésima parte por mil de la magnitud de salida actual:

$inv > 10^{-10} \times OUTV$

Para que el comportamiento de transferencia del algoritmo del integrador equivalga al comportamiento analógico, el tiempo de muestreo debería ser menor que el 20 % del tiempo integrativo ajustado o bien TI debería tener como mínimo cinco veces el valor del tiempo de muestreo seleccionado:

$CYCLE < 0{,}2 \times TI$

El algoritmo admite para el tiempo de muestreo valores hasta $CYCLE \leq 0{,}5 * TI$.

Inicializar la acción I

Cuando se activa la inicialización de la acción I, el valor de inicialización se aplica a la variable estática LMN_I en caso de rearranque de la CPU o si COM_RST = TRUE. Al cambiar al modo normal reseteando la inicialización, la acción I comienza la integración de su magnitud de entrada a partir del valor de inicialización.

Detener la acción I

Mediante las variables estáticas INT_HPOS y INT_HNEG es posible bloquear la acción I en sentido positivo y negativo, respectivamente. Esto puede ser útil en cascadas. Si, p. ej., la magnitud manipulada del regulador secundario se aproxima al límite superior, existe la posibilidad de impedir que siga aumentando la magnitud manipulada del regulador piloto mediante su acción I. Este comportamiento se programa con las instrucciones siguientes, p. ej., en AWL:

```
A    "regulador secundario".QLMN_HLM
=    "regulador piloto".INT_HPOS
A    "regulador secundario".QLMN_ LLM
=    "regulador piloto".INT_HNEG
```

Consulte también

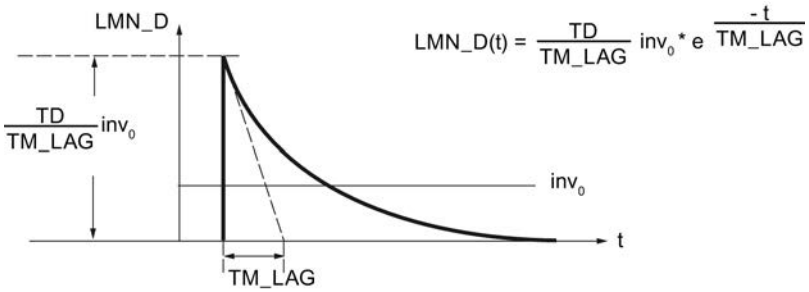
Algoritmo de regulación (Página 10137)

Acción D

Función de la acción D

La acción D genera una señal de salida cuya magnitud cambia proporcionalmente a la velocidad de cambio de la magnitud de entrada. La respuesta temporal se determina mediante el tiempo derivativo TD y el tiempo de retardo TM_LAG.

La respuesta a un escalón de entrada inv_0 es:



LMN_D(t)	la magnitud de salida de la acción D
inv_0	la altura del escalón en la entrada de la acción D
TD	Tiempo derivativo
TM_LAG	Tiempo de retardo

La acción D se observa en la variable estática LMN_D.

En estado desactivado (D_SEL = FALSE) la acción D, es decir, la memoria interna y la variable estática LMN_D, se pone a cero.

Aplicar la acción D en la rama de realimentación

Si la acción D está aplicada en la rama de realimentación, solo el valor real tendrá efecto sobre la acción D. Por ello, en una conmutación de manual a automático, la magnitud manipulada – igual que en el caso de un escalón de la consigna – no realiza ningún escalón por la acción D; la conmutación es suave.

Valores admisibles para el tiempo derivativo y el tiempo de retardo

Para que la acción D pueda calcularse correctamente, deben observarse las condiciones siguientes para el tiempo derivativo y el tiempo de retardo:

- TD ≥ CYCLE y
- TM_LAG ≥ 0,5 × CYCLE

Si TD < CYCLE, la acción D trabaja como si TD tuviera el valor CYCLE.

Si TM_LAG < 0,5 × CYCLE, la acción D trabaja sin retardo. Un escalón de entrada se multiplica por el factor TD/CYCLE y este valor se devuelve en la salida como "impulso de aguja". Esto significa que en el próximo ciclo de procesamiento, LMN_D se pondrá de nuevo a cero.

Coeficiente DT1

El coeficiente DT1 se calcula del modo siguiente:

Coeficiente DT1 = TD/TM_LAG

Tiempo de retardo

Para el filtrado de señales y la supresión de señales perturbadoras se ha integrado un retardo de 1er orden (constante de tiempo ajustable: TM_LAG) en el algoritmo para formar la acción D.

La mayoría de las veces es suficiente con un valor pequeño como tiempo de retardo para alcanzar el resultado deseado. Si se parametriza un tiempo de retardo inferior a la mitad del tiempo de muestreo del regulador, el retardo está desactivado.

Procedimiento

Para configurar la acción D, proceda del siguiente modo:

1. Para aplicar la acción D en la rama de realimentación, active la casilla de verificación correspondiente.
2. Introduzca el tiempo derivativo en segundos.
3. Introduzca el tiempo de retardo en segundos.

Consulte también

Algoritmo de regulación (Página 10137)

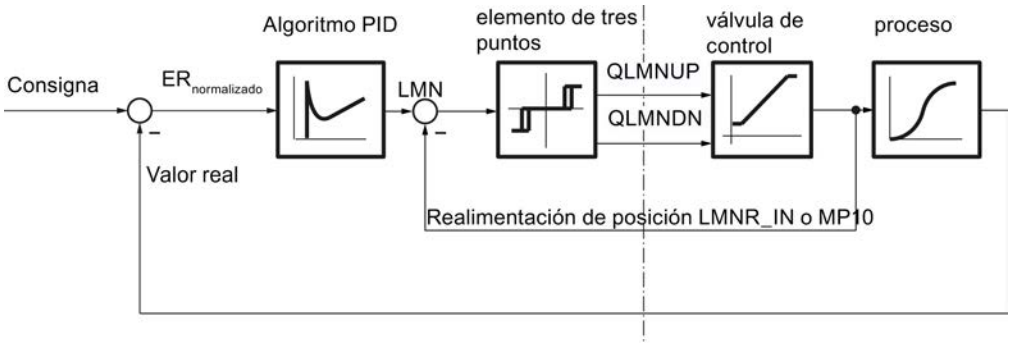
Variable manipulada

Regulador paso a paso con realimentación de posición

Esquema funcional del regulador paso a paso con realimentación de posición en el lazo de regulación

El funcionamiento del regulador paso a paso con realimentación de posición se basa en el algoritmo de regulación PID y se completa con los elementos funcionales que generan las señales de salida binarias a partir de la señal manipulada analógica del regulador.

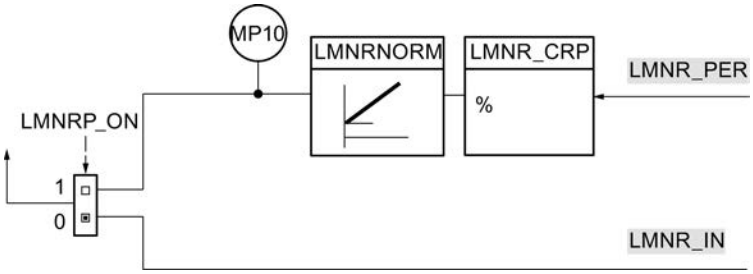
El elemento de tres puntos transforma las diferencias entre la magnitud manipulada y la realimentación de posición en impulsos positivos o negativos de la señal de salida (según sea el signo), que se dan en un accionamiento de válvula motriz, por ejemplo. En la práctica, se trata de una regulación en cascada con un lazo de regulación de posición subordinado.



Seleccionar la realimentación de posición

Para aplicar la realimentación de posición en el comparador en la rama de la variable manipulada del regulador paso a paso hay disponibles diferentes entradas con el correspondiente procesamiento de señal, en función del formato del valor que debe procesarse. Con LMNR_PER se pueden aplicar señales en el formato de la periferia SIMATIC y con LMNR_IN señales en formato de coma flotante.

En el punto de medición MP10 se deposita el valor interno correspondiente en el rango porcentual.



Si la realimentación de posición es suministrada por un módulo de entradas analógicas, el valor numérico existente en LMNR_PER de la realimentación de posición debe convertirse a un valor en coma flotante normalizado a porcentajes.

No se comprueba si se produce un desbordamiento positivo o negativo, si se alcanza el rango de tolerancia por encima o debajo ni si hay rotura de hilo.

La tabla siguiente ofrece una visión de conjunto de los rangos y valores numéricos antes y después del procesamiento.

Valor de periferia LMNR_PER	Valor de salida en %
32767	118,515
27648	100,000
1	0,003617
0	0,000
- 1	- 0,003617
- 27648	- 100,000
- 32768	- 118,519

Procedimiento

Nota

La realimentación de posición no debe activarse ni desactivarse en modo online. Esto significa que LMNR_ON no debe modificarse online.

Para activar la realimentación de posición, proceda del siguiente modo:

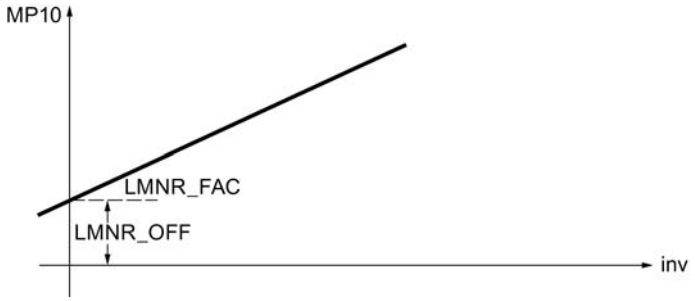
1. Active la casilla de verificación "Activar" en el área "Seleccionar realimentación de posición".
2. Elija la opción "Realimentación de posición en periferia" para aplicar señales en el formato de periferia SIMATIC.
   Elija la opción "Realimentación de posición" para aplicar señales en el formato de coma flotante.

Normalizar la realimentación de posición

Si la realimentación de posición está disponible en forma de magnitud física (p. ej. de 240 a 800 mm o en un ángulo de 0° a 60°), la realimentación de posición en formato de coma flotante (%) debe normalizarse a la magnitud física en el valor interno en coma flotante necesario para el procesamiento posterior.

Para definir la recta de normalización de forma unívoca deben definirse los parámetros siguientes:

- el factor (para la pendiente): LMNR_FAC
- el offset de la recta de normalización en el punto cero: LMNR_OFF



La realimentación de posición normalizada se calcula de acuerdo con la fórmula siguiente de inv (LMNR_PER) y se emite en MP10:

MP10 = inv * LMNR_FAC + LMNR_OFF

La normalización se hace efectiva cuando la realimentación de posición en periferia está activada (LMNRP_ON = TRUE). Dentro de la función no se limitan los valores. No tiene lugar una comprobación de los parámetros.

Procedimiento

Para normalizar la señal de la realimentación de posición, proceda del siguiente modo:

1. En el campo de entrada "Factor", introduzca la pendiente de la recta de normalización.
2. En el campo de entrada "Offset", introduzca el offset de la recta de normalización hacia el punto cero.

Regulador paso a paso sin realimentación de posición

Estructura y función del regulador paso a paso

El regulador paso a paso sin realimentación de posición se compone de dos elementos: el elemento de regulador PD que trabaja con señales continuas y un segundo elemento en el que se generan las señales manipuladas binarias a partir de la diferencia entre la acción PD y la realimentación.

La acción I del regulador paso a paso sin realimentación de posición se calcula en la realimentación. La señal de realimentación comparada con la salida del regulador PD se forma integrando los dos elementos siguientes de la señal:

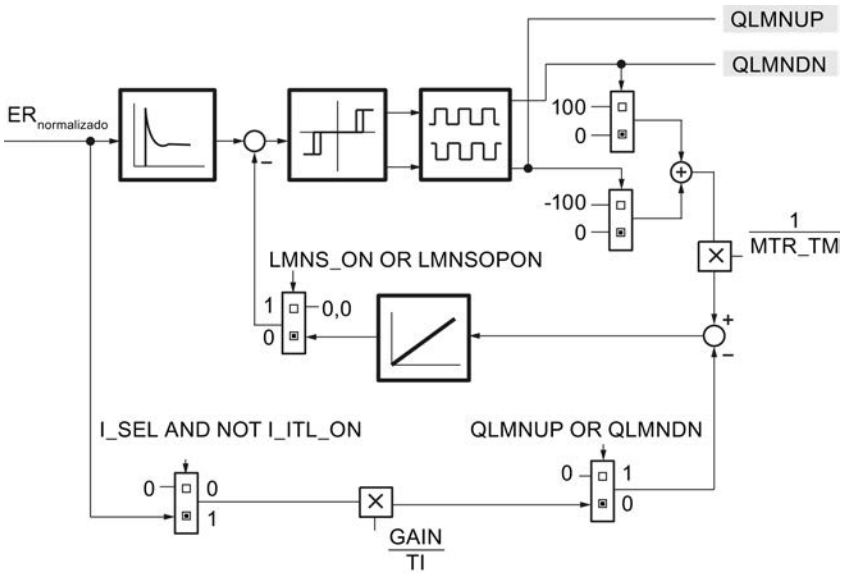
- Componente de señal para la realimentación de posición simulada
  ± 100/MTR_TM
- Componente de señal para la acción I
  $ER_{normalizado}$*GAIN/TI

Así pues, la señal de realimentación es la diferencia entre la realimentación de posición simulada y la acción I.

El elemento de tres puntos transforma las diferencias entre la magnitud manipulada y la señal de realimentación en impulsos positivos o negativos de la señal de salida (según sea el signo), que se dan en un accionamiento de válvula motriz, por ejemplo.

En estado estabilizado, la salida de la acción I y la acción PD se convierten en cero. Puesto que la entrada del elemento de tres puntos también se convierte en cero, las señales manipuladas binarias QLMNUP y QLMNDN permanecen en FALSE.

La acción I del algoritmo PID está desactivada. En el regulador paso a paso sin realimentación de posición no está previsto predeterminar o congelar la acción I. Aquí, el modo manual a través del parámetro MAN se suprime debido a la falta de información sobre la posición del actuador.



La activación de las señales de tope del actuador (LMNR_HS/LMNR_LS) provoca el bloqueo de la variable manipulada LMNUP o LMNDN.

Si el actuador no suministra señales de tope, hay que activar los parámetros de entrada LMNR_HS y LMNR_LS = FALSE.

Nota

Si no hay señales de tope, el regulador no puede detectar si se ha alcanzado un tope mecánico. Existe la posibilidad de que el regulador emita señales para abrir la válvula, por ejemplo, aunque ya se encuentre en el tope superior.

Modo manual

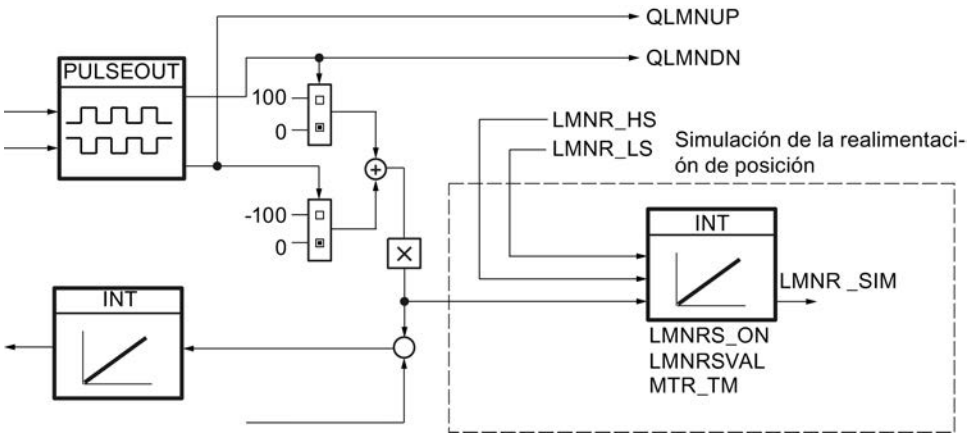
Debido a la falta de información sobre la posición del actuador, en el regulador paso a paso sin realimentación de posición se suprime el modo manual a través del parámetro MAN o el generador de la variable manipulada manual MAN_GEN. Solo es posible el modo manual mediante conmutación directa de las señales manipuladas (LMNS_ON = TRUE).

Simulación de la realimentación de posición

Si no se dispone de una realimentación de posición como magnitud mensurable, también puede simularse (LMNRS_ON = TRUE).

La simulación de la realimentación de posición se realiza mediante un integrador usando el tiempo de posicionamiento del motor MTR_TM como constante de tiempo integrativo. En el estado LMNRS_ON = FALSE se emite el valor inicial del parámetro LMNRSVAL en la salida del integrador LMNR_ SIM. Después de conmutar a LMNRS_ON = TRUE, la simulación arranca con este valor inicial.

Si LMNR_HS = TRUE, la integración se detiene hacia arriba, si LMNR_LS = TRUE, se detiene hacia abajo. No se sincroniza la realimentación de posición simulada en las posiciones finales.



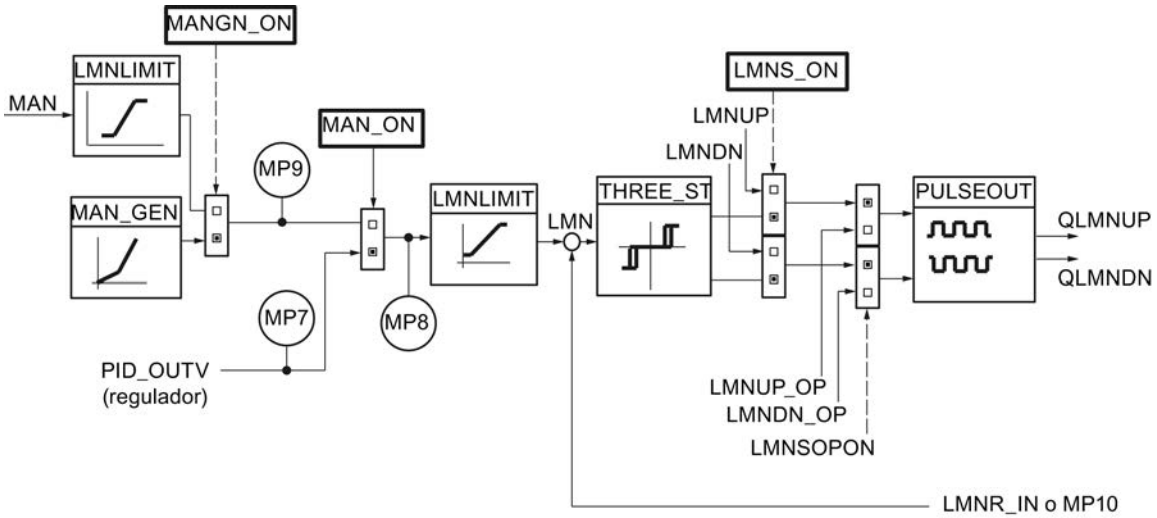
Nota

La realimentación de posición solamente se simula. No tiene que coincidir con la posición real del actuador. Si existe una realimentación de posición real, se debería usar sin falta.

Modo manual

Modo manual

La estructura de la generación y aplicación del valor manual puede verse en la figura siguiente.



En el regulador paso a paso existen tres modos de operación en los que es posible influir manualmente en la variable manipulada:

- Valor manual a través del parámetro de entrada MAN
  A través del parámetro MAN (-100,0 % a 100,0 %) se puede aplicar una influencia externa de la magnitud manipulada como valor absoluto, o puede predeterminarse desde el programa de usuario.
  Solo es posible si la realimentación de posición está activada.
- Modo manual mediante interruptor de más/menos del generador del valor manual (MANUP, MANDN),
  Si se activa MAN_GEN desde otro modo de operación, se aplica la variable manipulada MP9. Así, la conmutación al generador del valor manual siempre es suave. La variable manipulada manual puede aumentarse o reducirse dentro de los límites LMN_HLM y LMN_LLM.
  Solo es posible si la realimentación de posición está activada.
- Modo manual por aplicación directa de las señales manipuladas
  El modo manual por aplicación directa de las señales manipuladas LMNUP y LMNDN siempre tiene preferencia.
  Cuando se retrocede LMNS_ON = FALSE se aplica "con suavidad" un modo de operación cualquiera activado.

Modo automático

Si se ha desactivado el modo manual (MAN_ON = FALSE), la variable manipulada del algoritmo PID se aplica al elemento de tres puntos. La conmutación de modo manual a automático provoca un escalón de la variable manipulada LMN. No obstante, esto no tiene ningún efecto negativo, ya que el proceso se acciona en forma de rampa a través del actuador de acción integrativa. La salida del algoritmo PID se encuentra en el punto de medición MP7.

Corrección del valor manual en el modo automático

Si la realimentación de posición en periferia está activada (LMNRP_ON = TRUE), el parámetro de entrada MAN se corrige según la posición del actuador (MP10).

Si la realimentación de posición está activada (LMNRP_ON = FALSE), el parámetro de entrada MAN se corrige según la posición del actuador (LMNR_IN).

Al conmutar al modo manual, la magnitud manipulada permanece en el valor correspondiente a la posición del actuador. A continuación puede modificarse mediante intervención manual.

Activar el modo manual con valor manual

En este modo de operación (MANGN_ON = FALSE y MAN_ON = TRUE) se introduce la variable manipulada manual como valor absoluto en la entrada MAN. La variable manipulada manual se indica en el punto de medición MP8.

1. Active la casilla de verificación "Activar modo manual".
2. Seleccione la opción "Valor manual".

Activar el generador de valor manual

El valor manual MAN se aumenta o reduce progresivamente utilizando los interruptores MANUP y MANDN. El valor modificado puede observarse en MP8. Se tienen en cuenta los límites configurados de la variable manipulada.

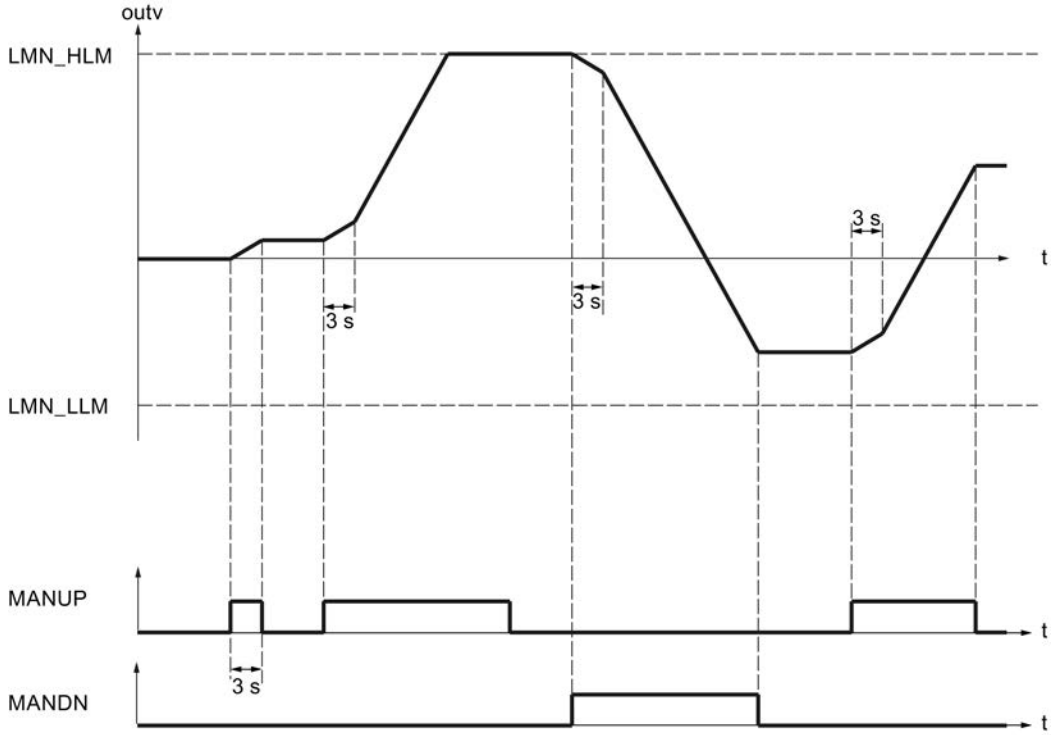
1. Active la casilla de verificación "Activar modo manual".
2. Seleccione la opción "Generador del valor manual".

Repercusión de los interruptores MANUP y MANDN

Para poder realizar un cambio graduado finamente, el regulador debe tener un tiempo de muestreo de 100 ms como máximo.

La velocidad de cambio de la variable manipulada depende de la duración de pulsación de los interruptores MANUP y MANDN así como de los límites ajustados de acuerdo con la relación siguiente:

- durante los primeros 3 segundos después de activar MANUP o MANDN rige:
  doutv/dt = (LMN_HLM - LMN_LLM) / 100 s
- a continuación
  doutv/dt = (LMN_HLM - LMN_LLM) / 10 s



Con un tiempo de muestreo de 100 ms y un rango de la variable manipulada de -100,0 a 100,0, la variable manipulada cambia en 0,2 por ciclo en los tres primeros segundos, por ejemplo. Si MANUP se mantiene en TRUE durante más de 3 segundos, la velocidad de cambio aumenta a diez veces el valor, en este caso a 2 por ciclo.

El generador del valor manual se activa suavemente.

Activación de los modos de operación

La tabla siguiente muestra los modos de operación posibles del regulador paso a paso con los valores necesarios para los interruptores que determinan la estructura.

Tabla 11-5 Modos de operación del regulador paso a paso

	MAN_ON	MNGN_ON	LMNS_ON
Modo automático	FALSE	cualquiera	FALSE
Modo manual sin control por interruptor	TRUE	FALSE	FALSE
Modo manual mediante generador de valor manual	TRUE	TRUE	FALSE
Modo manual mediante señales manipuladas	cualquiera	cualquiera	TRUE

Límites

En el regulador paso a paso con realimentación de posición es posible limitar la variable manipulada.

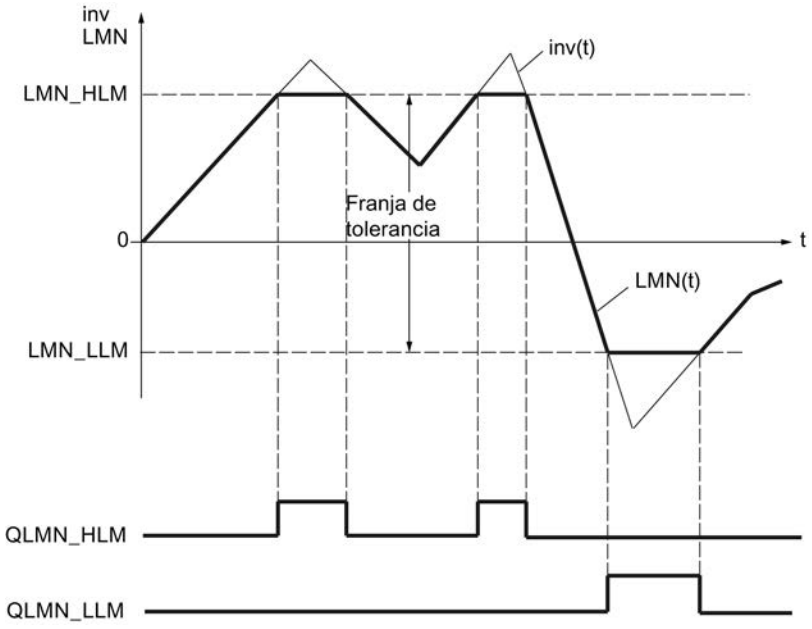
Limitar el valor absoluto de la magnitud manipulada

El rango de trabajo, es decir, el rango dentro del cual puede moverse el actuador sin abandonar las variables manipuladas admisibles, está determinado por el rango de ajuste de la variable manipulada. Dado que, dependiendo del tipo de actuador, los límites de las variables manipuladas permitidas no suelen cubrirse con el límite 0 % o 100 % del rango de ajuste, es posible configurar el límite superior e inferior de la variable manipulada.

Para evitar estados no permitidos en el proceso en cuestión, en la rama de la variable manipulada de la regulación se limita el rango de ajuste de la magnitud manipulada por arriba y abajo.

La función LMNLIMIT limita la magnitud manipulada LMN(t) a los valores inferior y superior LMN_HLM y LMN_LLM. Estos valores pueden predeterminarse. Sin embargo, la magnitud de entrada inv debe estar fuera de estos límites. Dado que la función no puede desactivarse, en la configuración siempre hay que tener en cuenta la indicación del límite inferior y superior.

Los valores numéricos de los límites (en porcentaje) se ajustan en los parámetros de entrada para la limitación inferior y superior. Si la magnitud de entrada inv(t) rebasa los límites, se muestran las indicaciones correspondientes a través de las salidas de aviso.



Arranque y funcionamiento

- En caso de rearranque, todas las salidas de aviso se ponen a cero.
- La limitación trabaja según las siguientes relaciones:

LMN =	QLMN_HLM	QLMN_LLM	si:
LMN_HLM	TRUE	FALSE	inv ≥ LMN_HLM
LMN_LLM	FALSE	TRUE	inv ≤ LMN_LLM
inv	FALSE	FALSE	LMN_HLM < inv < LMN_LLM

En la salida, es decir, en el parámetro LMN o en el punto de medición MP10, se muestra la variable manipulada efectiva de la regulación.

Procedimiento

Para limitar el valor absoluto de la magnitud manipulada, proceda del siguiente modo:

1. Introduzca el valor más alto permitido en el campo de entrada "Límite superior".
2. Introduzca el valor más bajo permitido en el campo de entrada "Límite inferior".

Generador de impulsos

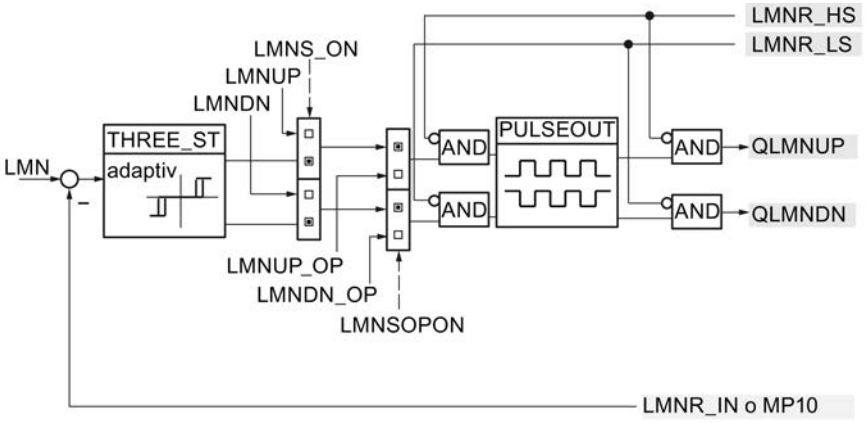
El generador de impulsos se encarga de que se mantenga el tiempo de conexión y desconexión mínimo con el fin de proteger los actuadores. La activación de las señales de tope del actuador (LMNR_HS/LMNR_LS) provoca el bloqueo de la variable manipulada LMNUP o LMNDN. Esto sirve tanto para el modo automático como manual.

Si están activados ambos interuptores de señal para el control de señal de la variable manipulada (LMNUP = LMNDN = TRUE o LMNUP_OP = LMNDN_OP = TRUE), las salidas QLMNUP y QLMNDN siempre son FALSE.

El cambio directo de "Señal de variable manipulada Abrir" (QLMNUP = TRUE, QLMNDN = FALSE) a "Señal de variable manipulada Cerrar" (QLMNUP = FALSE, QLMNDN = TRUE) no es posible. El generador de impulsos inserta un ciclo con QLMNUP = QLMNDN = FALSE.

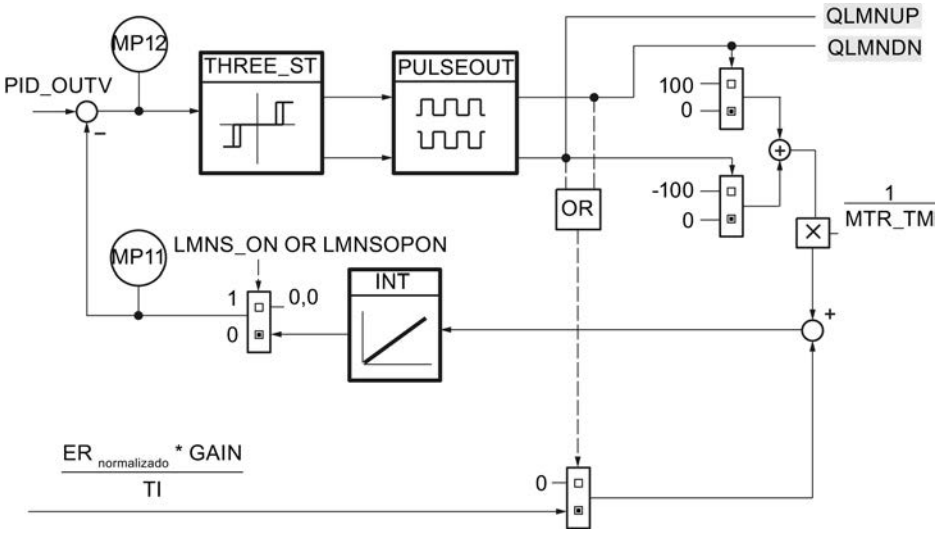
Señal de entrada INV en el regulador paso a paso con realimentación de posición

La diferencia entre la variable manipulada LMN y la realimentación de posición LMNR se aplica en el elemento de tres puntos con histéresis THREE_ST.



Señal de entrada INV en el regulador paso a paso sin realimentación de posición

La diferencia entre la acción PD del regulador y la magnitud de realimentación (MP11) se aplica al elemento de tres puntos con histéresis THREE_ST.



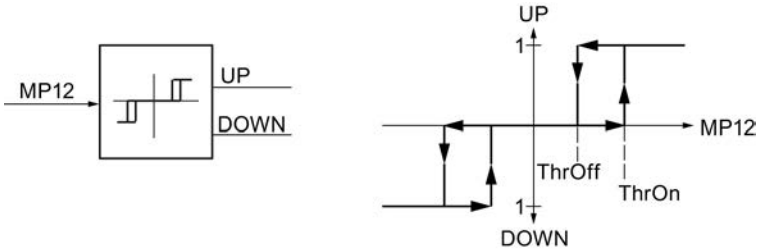
Influencia del tiempo de posicionamiento del motor en el elemento de tres puntos

El elemento de tres puntos genera dos señales binarias que son TRUE o FALSE según sean la magnitud y el signo del valor diferencial en la entrada.

El elemento de tres puntos THREE_ST reacciona a la señal de entrada INV de acuerdo con las relaciones siguientes y adopta en cada caso una de las tres combinaciones posibles de sus señales de salida UP/DOWN:

UP	DOWN	Combinación de entrada
TRUE	FALSE	|INV| ≥ ThrOn o INV > ThrOff y $UP_{antiguo}$ = TRUE
FALSE	TRUE	|INV| ≤ -ThrOn o INV < -ThrOff y $DOWN_{antiguo}$ = TRUE
FALSE	FALSE	|INV| ≤ ThrOff

ThrOn = umbral de conexión, ThrOff = umbral de desconexión



El umbral de desconexión ThrOff debe ser mayor que el cambio de la realimentación de posición o que el cambio de posición tras la duración de un impulso. Este valor depende del tiempo de posicionamiento del motor MTR_TM y se calcula del siguiente modo:

$$\text{ThrOff} = 0{,}5 * \frac{110}{\text{MTR_TM}} * \text{CYCLE}$$

MTR_TM debe ser un múltiplo entero de CYCLE.

Nota

Un tiempo de posicionamiento del motor demasiado elevado (10% por encima del tiempo de posicionamiento real) provoca una conexión y desconexión continuas de las señales de variable manipulada QLMNUP y QLMNDN.

Adaptación del umbral de conexión ThrOn

Con el fin de reducir la frecuencia de conmutación al regular errores de regulación mayores, el umbral de conexión ThrOn se adapta automáticamente durante el funcionamiento, mientras que ThrOff permanece constante. ThrOn está limitado a:

$$\text{Min ThrOn} = \frac{100}{\text{MTR_TM}} * \text{MAX (PULSE_TM, CYCLE)}$$

$$\text{Max ThrOn} = 10$$

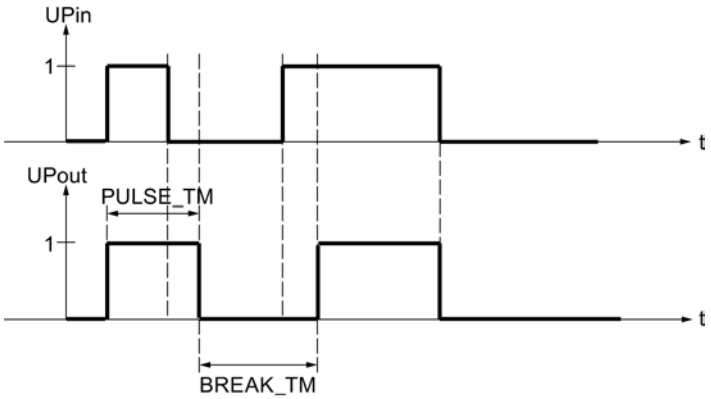
La adaptación del umbral de conexión está desactivada en reguladores P, D o PD puros. En este caso es:

ThrOn = Min ThrOn

Tiempo de conexión y desconexión mínimo

El generador de impulsos procura que se mantenga el tiempo de conexión y desconexión mínimo al conectar y desconectar los impulsos de salida, respectivamente.

Por este motivo, para proteger el actuador es posible parametrizar un tiempo de conexión mínimo PULSE_TM y un tiempo de desconexión mínimo BREAK_TM. La duración de los impulsos de salida QLMNUP y QLMNDN es siempre mayor que el tiempo de conexión mínimo y la pausa entre dos impulsos es siempre mayor que el tiempo de desconexión mínimo. La figura siguiente muestra el funcionamiento siguiendo el ejemplo de la señal UP.



Procedimiento

Para configurar el generador de impulsos, proceda del siguiente modo:

1. Introduzca el tiempo de posicionamiento del motor en el campo de entrada "Tiempo pos. motor".
2. Introduzca un "Tiempo mín. conexión".
3. Introduzca un "Tiempo mín. desconexión".

Puesta en servicio de PID_ES

Realizar una optimización manual

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- MAN_ON = FALSE

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Medición activada".
   Si aún no existe ninguna conexión online, ésta se establece. Los valores actuales de consigna, valor real y variable manipulada se graban.
2. Introduzca parámetros PID nuevos en los campos "P", "I", "D" y "Tiempo de retardo".
3. Haga clic en el grupo "Optimización" en el símbolo "Enviar parámetros a la CPU".
4. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .
5. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
   Si la velocidad de cambio de la consigna está limitada, se conmuta suavemente a la nueva consigna.
6. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabaja con los nuevos parámetros PID y regula según la nueva consigna.
7. Compruebe la calidad de los parámetros PID basándose en la evolución de las curvas.
8. Repita los pasos del 3 hasta el 8, hasta que esté satisfecho con el resultado de la regulación.

Consulte también

Trabajar con el visor de curvas (Página 10163)

Trabajar con el visor de curvas

El visor de curvas permite representar gráficamente los valores de consigna, real y manipulado a lo largo del tiempo. Los valores del visor de curvas se actualizan en el tiempo de actualización seleccionado. El registro de los valores de las curvas comienza al hacer clic en el botón "Medición on" y finaliza al hacer clic en el botón "Medición off".

Elementos del visor de curvas

① Selección del modo de visualización
② Ventana de curvas
③ Área para mover y escalar los ejes
④ Regla
⑤ Leyenda con los valores de las curvas en la regla

Selección del modo de visualización

Para la representación del registro de curvas pueden seleccionarse los siguientes modos de visualización:

- Strip (visualización continuada)
  Los valores de curva nuevos se registran en la ventana de curvas derecha. Los valores de curva transcurridos se van deslizando a la izquierda. No es posible mover el eje de tiempo.
- Scope (visualización cambiante por áreas)
  Los valores de curva nuevos se registran de izquierda a derecha dentro de la ventana de curvas. Cuando se alcanza el margen derecho de la ventana de curvas, el área de observación se traslada un ancho de ventana a la derecha. El eje de tiempo puede moverse dentro de los límites del área de observación.
- Sweep (visualización en rotación)
  Los valores nuevos se muestran en la ventana de curvas en forma de visualización en rotación. Los valores se registran de izquierda a derecha. En la posición de escritura se sobrescriben los valores de la última rotación. No es posible mover el eje de tiempo.
- Static (visualización estática por áreas)
  La escritura de las curvas se interrumpe: los nuevos valores se registran en segundo plano. El eje de tiempo puede moverse a lo largo de todo el periodo registrado hasta ahora.

La relación entre el área visible y el área registrada puede consultarse en el visor de relación:

El ancho del visor de relación equivale a toda la duración del registro. La parte amarilla del visor representa la parte del registro visible en la ventana de curvas. La parte gris representa la parte del registro **no** visible en la ventana de curvas.

En el modo de visualización "Static" es posible mover el área visible del visor de relación con el ratón a lo largo del eje.

Ventana de curvas

En la ventana de curvas se muestran las curvas correspondientes a los valores de consigna, real y manipulado. Además de la distinción por colores, las curvas se identifican mediante símbolos (ver leyenda).

Mover y escalar los ejes

Los ejes correspondientes a los valores de consigna, real y manipulado, así como al eje de tiempo, pueden moverse y escalarse individualmente. Los botones derecho e izquierdo del ratón tienen asignadas funciones cambiantes. Puede utilizar los siguientes botones y acciones de ratón:

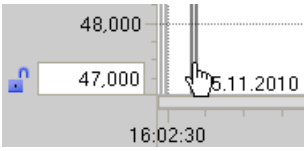
	Mover el eje de la consigna, del valor real o de la variable manipulada hacia arriba y abajo Si el valor de escala superior o inferior están bloqueados, el eje no puede moverse.
	Mover el eje de tiempo a derecha e izquierda Si el valor de escala izquierdo o derecho están bloqueados, el eje no puede moverse.
	Expansión y contracción comunes de los ejes de los valores de consigna, real y manipulado • Si ninguno de los valores de escala está bloqueado, la escala de los ejes se estira o encoge simétricamente. • Si uno de los valores de escala está bloqueado, se conserva el valor bloqueado al expandir o contraer la escala.
	Expandir y contraer el eje de tiempo • Si ninguno de los valores de escala está bloqueado, la escala del eje se estira o se encoge simétricamente. • Si uno de los valores de escala está bloqueado, se conserva el valor bloqueado al expandir o contraer la escala.
	Expansión y contracción de los ejes de consigna, valor real y variable manipulada El valor de escala inferior no se modifica al expandir o contraer la escala.
	Expansión y contracción de los ejes de consigna, valor real y variable manipulada. El valor de escala superior no se modifica al expandir o contraer la escala.
	Expansión y contracción del eje de tiempo. El valor de escala derecho no se modifica al expandir o contraer la escala.
	Expansión y contracción del eje de tiempo. El valor de escala izquierdo no se modifica al expandir o contraer la escala.
50,000	Entrada de un valor de escala.

	El símbolo de un candado permite bloquear el valor de escala actual. Sólo es posible bloquear un valor de cada eje.
	Haciendo doble clic en la ventana de curvas se optimiza la escala y la posición de los ejes de los valores de consigna, real y manipulado en la ventana de curvas.
	Haciendo doble clic en el área del eje se restablecen la posición y la escala predeterminadas del eje.

Trabajar con reglas

Utilice una o más reglas para analizar valores discretos de las curvas.

Las posiciones de estacionamiento para reglas verticales se encuentran en el margen izquierdo del área de curva, para las reglas horizontales, en el margen superior.



Mueva el ratón hasta el margen izquierdo del área de curvas y observe cómo cambia el puntero del ratón. Arrastre ahora la regla vertical hasta la posición de la curva de medición que desea analizar.

Los valores de la curva situados junto a la regla vertical se muestran en ésta sangrados a la izquierda. El valor de tiempo de la posición de la regla se muestra en la base de ésta.

Los valores de la curva mostrados en la regla activa se muestran en la leyenda. Si se arrastran varias reglas al área de curvas, estará activa la última regla arrastrada. La regla activa se reconoce por el símbolo en color.

Sin embargo, los valores de las reglas horizontales no pueden representarse.

Las reglas no activas pueden volver a activarse haciendo clic.

Las reglas que ya no se necesitan pueden retirarse con Alt + clic.

Puesta en servicio con el PID Self Tuner

La instrucción PID_ES se pone en servicio con el PID Self-Tuner y se determinan los parámetros PID óptimos.

Procedimiento

Para poner en servicio un regulador paso a paso, proceda del siguiente modo:

1. Llame la instrucción PID_ES en un OB de alarma cíclica e interconecte los parámetros de entrada y salida.
2. Llame la instrucción TUN_ES en el mismo OB de alarma cíclica.

3. Interconecte la instrucción TUN_ES con PID_ES del siguiente modo:
   - TUN_ES.PV con PID_ES.PV
   - Si se utiliza una realimentación de posición, TUN_ES.LMNR con PID_ES.LMNR_IN
   - TUN_ES.C_LMNUP y TUN_ES.C_LMNDN con PID_ES.QLMN_UP y PID_ES.QLMNDN
   - TUN_ES.LMNR_HS con PID_ES.LMNR_HS
   - TUN_ES.LMNR_ON con PID_ES.LMNR_ON
   - TUN_ES.LMNS_ON con PID_ES.LMNS_ON
   - TUN_ES.SP_OUT con PID_ES.SP_IN o PID_ES.SP_EXT
   - TUN_ES.GAIN con PID_ES.GAIN
   - TUN_ES.TI con PID_ES.TI
   - TUN_ES.TD con PID_ES.TD
   - TUN_ES.TM_LAG con PID_ES.TM_LAG
   - TUN_ES.MTR_TM con PID_ES.MTR_TM
   - Si se utiliza una zona muerta, TUN_ES.DEADB_W con PID_ES.DEADB_W
   - TUN_ES.QLMNUP y TUN_ES.QLMNDN con PID_ES.LMNUP y PID_ES.LMNDN
4. Haga clic en el símbolo en la instrucción TUN_ES.
   Se abre el editor de puesta en servicio para TUN_ES. El funcionamiento del editor de puesta en servicio se describe en el PID Self-Tuner.

11.3.5.2 Utilizar Modular PID Control

Volumen de funciones del Modular PID Control

En muchas tareas de regulación no solo es importante el clásico regulador PID como elemento que influye en el proceso, sino que también se ponen elevadas exigencias al procesamiento de las señales.

Modular PID Control contiene instrucciones con funciones para acondicionar el valor real y la consigna, así como para calcular y retocar la variable manipulada.

Son posibles los reguladores siguientes:

- Reguladores continuos
- Reguladores paso a paso para actuadores de acción integrativa
- Reguladores de impulsos

Interconectando las diferentes instrucciones se consiguen reguladores para diferentes aplicaciones:

- Regulador de valor fijo
- Regulador en cascada
- Regulador de relación
- Regulador de mezcla

- Regulador de rango partido
- Regulador de corrección
- Regulador con procesamiento de magnitudes perturbadoras
- Regulador multivariable

Se pueden regular tanto sistemas regulados lentos (temperatura, nivel, etc.) como sistemas regulados muy rápidos (caudal, presión, etc.).

La instrucción LP_SCHED_M permite llamar muchos lazos de regulación de forma equidistante pero con diferentes tiempos de muestreo. El tiempo de muestreo del lazo de regulación varía según sea la inercia del correspondiente sistema regulado.

De este modo se garantiza una carga homogénea de la CPU.

Para poner en servicio y optimizar los parámetros PID se utiliza el PID Self-Tuner.

Pasos básicos para crear un FB de regulación

Procedimiento básico para programar un regulador

Un regulador se programa, p. ej., con AWL como bloque de función (FB de regulación). El FB de regulación debe tener como mínimo los parámetros siguientes:

- Parámetro de entrada COM_RST
- Parámetro de entrada CYCLE
- Parámetro de entrada para consigna
- Parámetro de entrada para valor real
- Parámetro de salida para la variable manipulada
- Parámetro de entrada/salida para la variable manipulada manual (regulador continuo)

Adicionalmente, es posible crear los parámetros siguientes según sean los requisitos de regulación:

- Parámetro de entrada para realimentación de posición (regulador paso a paso)
- Parámetro de entrada para una magnitud perturbadora
- Parámetro de salida para el valor real acondicionado
- Parámetro de salida para la consigna acondicionada
- Parámetro de salida para el rebase por exceso de límites

El detalle con el que se configuren las entradas y salidas dependerá de las particularidades específicas, por ejemplo, si el valor real tiene formato de periferia o en coma flotante, qué realimentación de posición hay, etc.

En el FB de regulación se llaman las instrucciones de Modular PID Control y se interconectan los parámetros de entrada y salida. El FB de regulación creado se llama, p. ej. en el OB 100 o en un OB de alarma cíclica.

Para programar un regulador con Modular PID Control deben seguirse los pasos siguientes:

Paso	Descripción
1.º	Acondicionar consigna (Página 10169)
2.º	Acondicionamiento del valor real (Página 10171)
3.º	Determinar y acondicionar el error de regulación (Página 10173)
4.º	Control anticipativo (Página 10175) (opcional)
5.	Calcular variable manipulada • Crear un regulador continuo (Página 10178) • Crear un regulador de impulsos (Página 10179) • Crear un regulador paso a paso (Página 10183)
6.	Acondicionamiento de la realimentación de posición (Página 10184)

Consulte también

Acondicionar consigna (Página 10169)

Acondicionamiento del valor real (Página 10171)

Determinar y acondicionar el error de regulación (Página 10173)

Control anticipativo (Página 10175)

Crear un regulador continuo (Página 10178)

Crear un regulador de impulsos (Página 10179)

Crear un regulador paso a paso (Página 10183)

Acondicionamiento de la realimentación de posición (Página 10184)

Programar el FB de regulación

Acondicionar consigna

Instrucciones para acondicionar la consigna

Se pueden utilizar las siguientes instrucciones.

- SP_GEN: especificar consigna manualmente
- ROC_LIM: limitar velocidad de cambio
- LAG1ST: filtrar consigna
- NONLIN: linealizar consigna
- LIMITER: limitar consigna
- RMP_SOAK: especificar consigna según programación
- SCALE_M: escalar consigna
- NORM: normalizar consigna

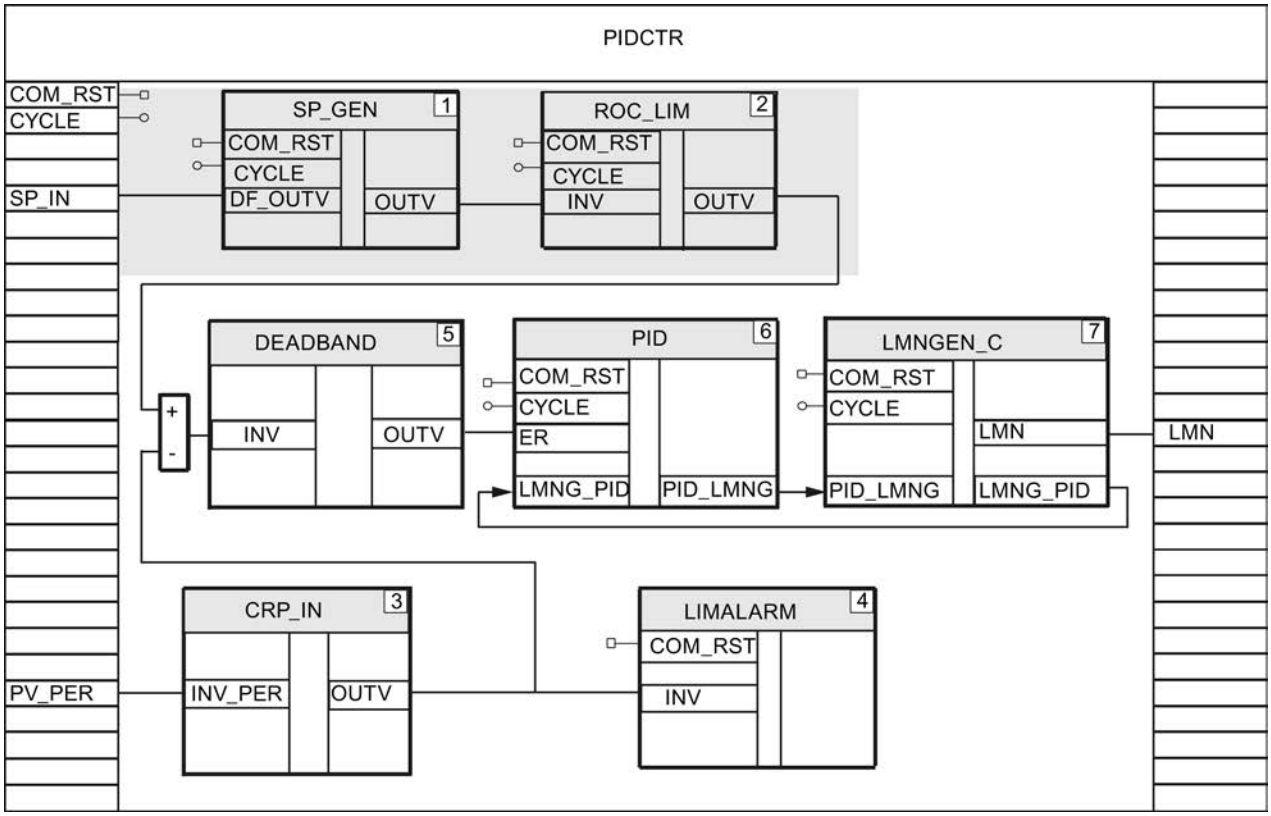
Ejemplo de acondicionamiento de la consigna

Una consigna especificada debe poder aumentarse o reducirse manualmente. Para evitar cambios de la consigna en forma de escalón, se limita la velocidad de cambio.

Para acondicionar la consigna tal como se describe en el ejemplo, proceda del siguiente modo:

1. Declare en el FB de regulación, en este caso PIDCTR, un parámetro de entrada SP_IN del tipo de datos "Real".
2. Interconecte SP_IN con una variable que contenga la consigna deseada.
3. En el FB de regulación llame la instrucción SP_GEN como multiinstancia e interconecte SP_IN con SP_GEN.DF_OUTV.
4. Parametrice el límite superior e inferior de la consigna en la estructura SP_GEN_Instance de la multiinstancia.
5. En el FB de regulación llame la instrucción ROC_LIM como multiinstancia e interconecte SP_GEN.OUTV con ROC_LIM.INV mediante una variable temporal.
6. Parametrice las velocidades de cambio máximas en la estructura ROC_LIM_Instance de la multiinstancia.
7. Para calcular el error de regulación, reste el valor real acondicionado de ROC_LIM.OUTV o interconecte ROC_LIM.OUTV con ERR_MON.SP.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR.



Consulte también

Acondicionamiento del valor real (Página 10171)
Descripción de SP_GEN (Página 7312)
Descripción de ROC_LIM (Página 7304)
Descripción de LAG1ST (Página 7253)
Descripción de NONLIN (Página 7277)
Descripción de LIMITER (Página 7263)
Descripción de RMP_SOAK (Página 7297)
Descripción de SCALE_M (Página 7311)
Descripción NORM (Página 7280)

Acondicionamiento del valor real

Instrucciones para el acondicionamiento del valor real

Se pueden utilizar las siguientes instrucciones.

- CRP_IN: escalar valor de periferia
- LAG1ST: filtrar valor real
- LIMITER: limitar valor real
- LIMALARM: vigilar valor real
- NONLIN: linealizar valor real
- SCALE_M: escalar valor real
- NORM: normalizar valor real

Ejemplo de acondicionamiento del valor real

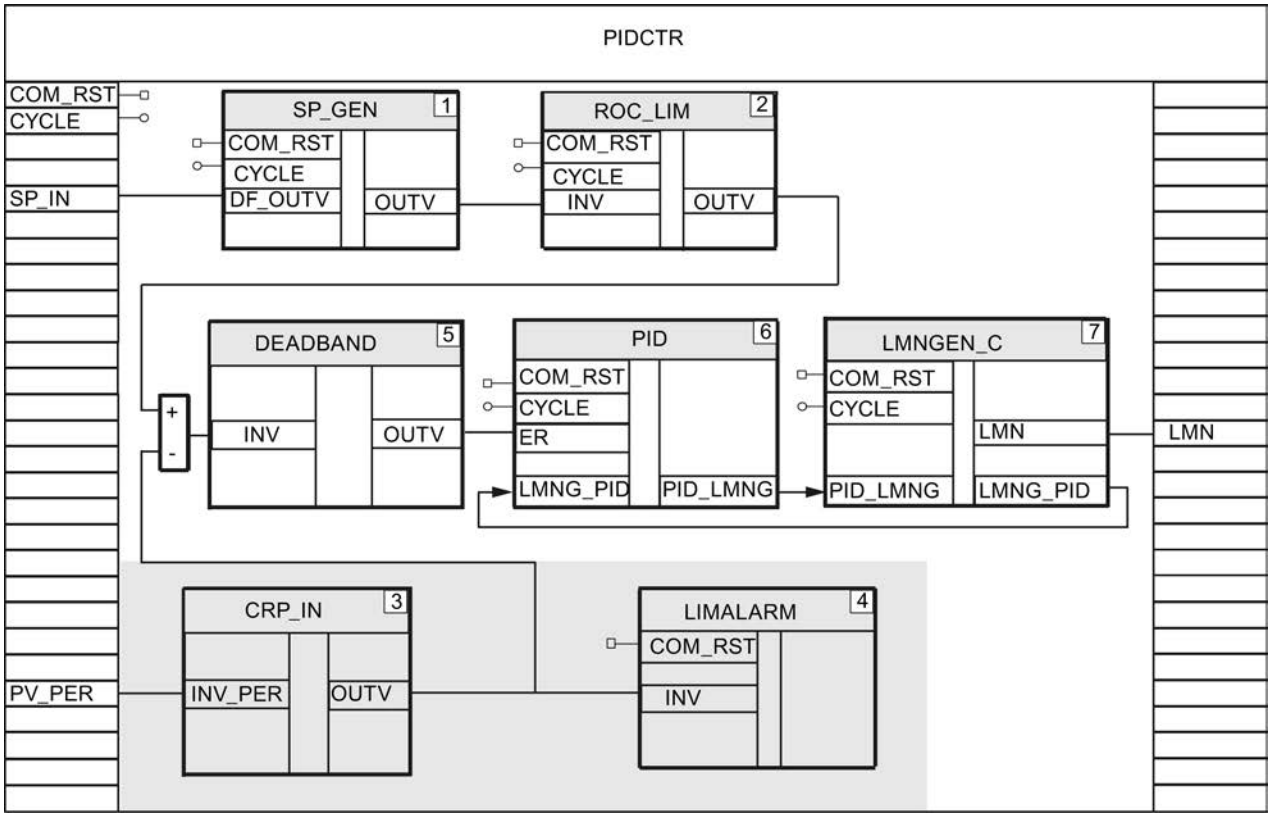
El valor real tiene formato de periferia y debe vigilarse el mantenimiento de los límites de valor real.

Para acondicionar el valor real tal como se describe en el ejemplo, proceda del siguiente modo:

1. Declare en el FB de regulación, en este caso PIDCTR, un parámetro de entrada PV_PER del tipo de datos "Word".
2. Interconecte PV_PER con la entrada de periferia en la que se mide el valor real.
3. En el FB de regulación llame la instrucción CRP_IN como multiinstancia, p. ej. con el nombre CRP_IN_Instance_PV.
4. Interconecte PV_PER con CRP_IN.INV_PER.
5. Parametrice el factor y offset para el escalado en la estructura CRP_IN_Instance_PV de la multiinstancia.

6. En el FB de regulación llame la instrucción LIMALARM como multiinstancia e interconecte CRP_IN.OUTV con LIMALARM.INV mediante una variable temporal.
7. Parametrice los límites del valor real en la estructura LIMALARM_Instance de la multiinstancia.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR.



Consulte también

Acondicionar consigna (Página 10166)

Determinar y acondicionar el error de regulación (Página 10173)

Descripción de CRP_IN (Página 7238)

Descripción de LAG1ST (Página 7253)

Descripción de LIMITER (Página 7263)

Descripción de LIMALARM (Página 7260)

Descripción de NONLIN (Página 7277)

Descripción de SCALE_M (Página 7311)

Descripción NORM (Página 7280)

Determinar y acondicionar el error de regulación

Determinar y limitar el error de regulación

El error de regulación se determina de dos formas: programando la resta del valor real de la consigna o utilizando la instrucción ERR_MON. En la instrucción ERR_MON ya se incluye una vigilancia del error de regulación.

Instrucciones para el acondicionamiento del error de regulación

Se pueden utilizar las siguientes instrucciones.

- A_DEAD_B: conducir el error de regulación por la zona muerta adaptativa
- DEADBAND: conducir el error de regulación por la zona muerta

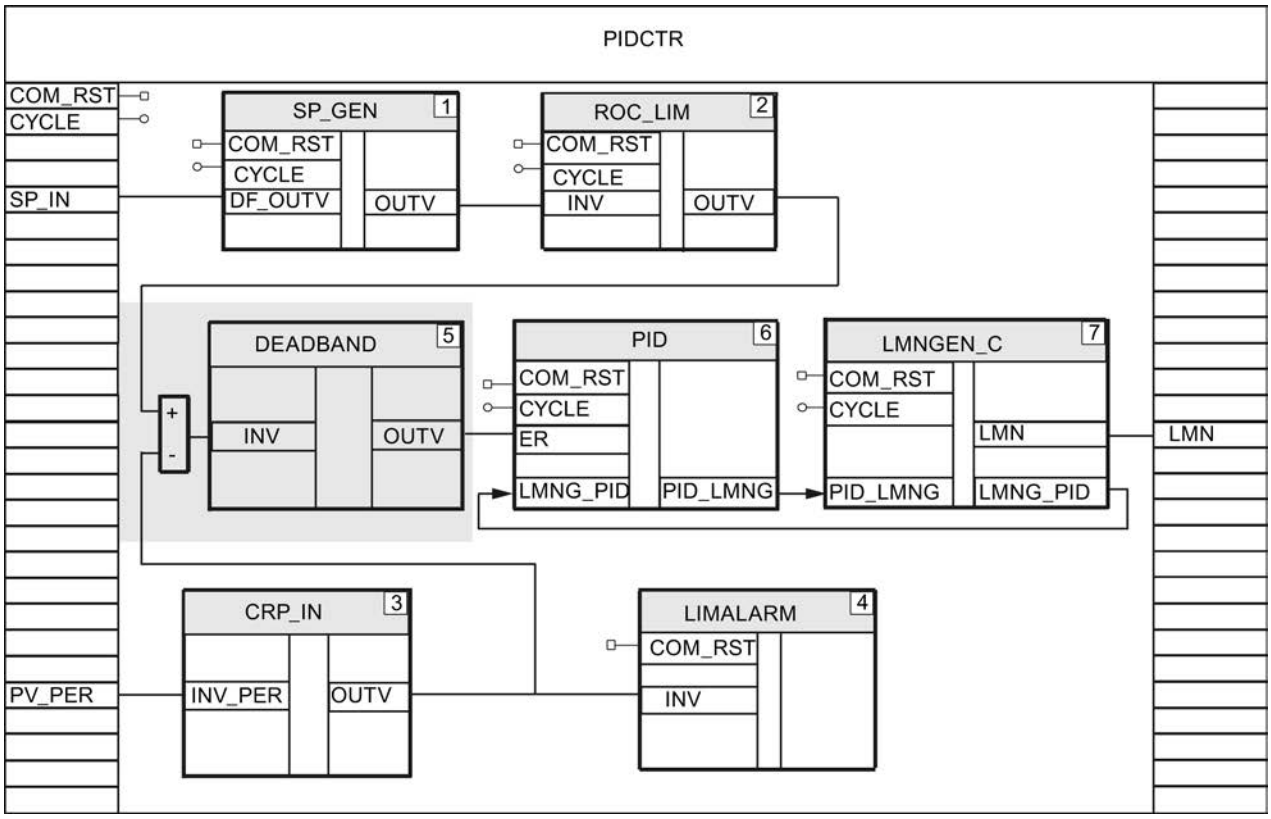
Ejemplo de la resta

El error de regulación se calcula restando y se conduce por una zona muerta.

Para determinar el error de regulación tal como se describe en el ejemplo, proceda del siguiente modo:

1. Llame la instrucción DEADBAND como multiinstancia e interconecte DEADBAND.INV con la diferencia entre consigna y valor real, p. ej. ROC_LIM.OUTV - CRP_IN.OUTV.
2. Parametrice el ancho de la zona muerta y un offset de la zona muerta en la estructura DEADBAND_Instance de la multiinstancia.
   Un DEADBAND.OUTV mostrará el error de regulación acondicionado.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR.



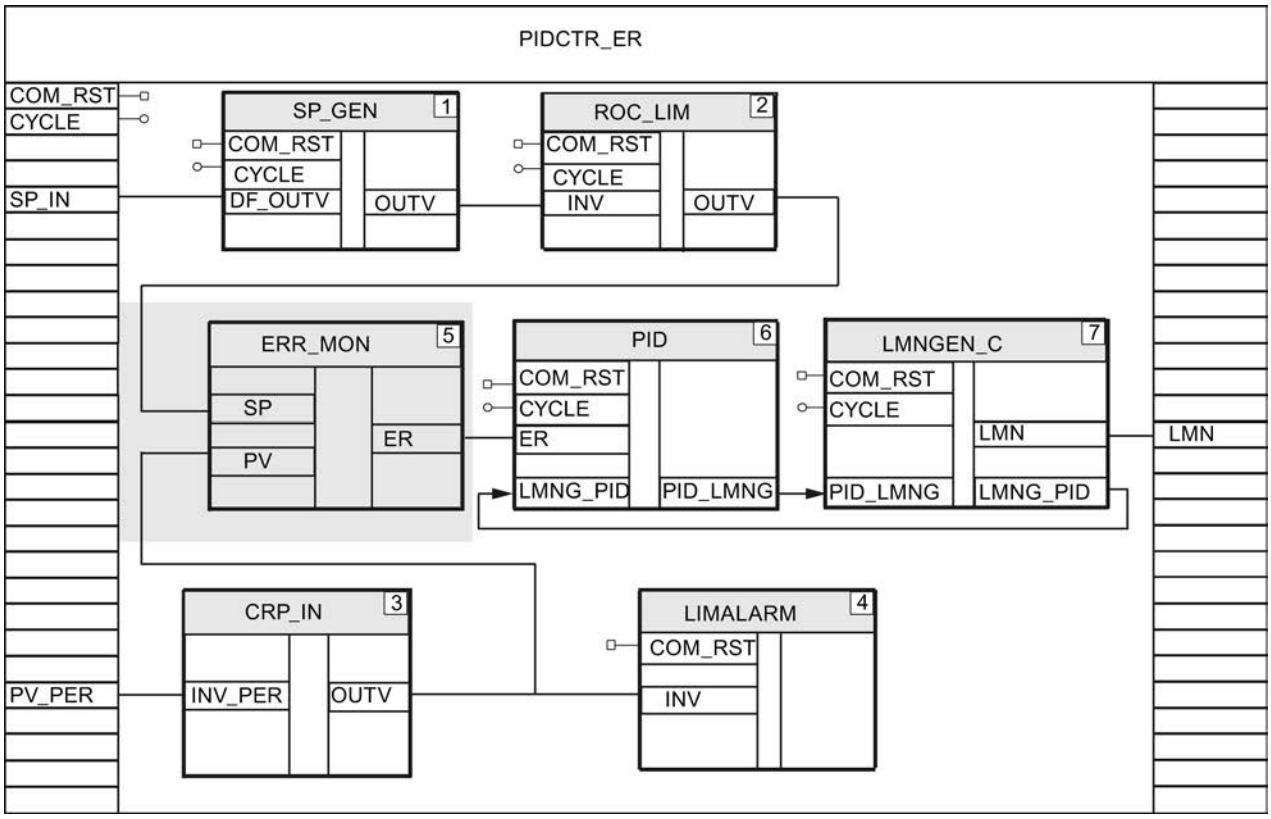
Ejemplo del uso de ERR_MON

El error de regulación se determina y vigila con la instrucción ERR_MON.

Para determinar el error de regulación tal como se describe en el ejemplo, proceda del siguiente modo:

1. Llame la instrucción ERR_MON como multiinstancia.
2. Utilizando variables temporales, interconecte la consigna acondicionada con ERR_MON.SP, p. ej. ROC_LIM.OUTV, y el valor real acondicionado con ERR_MON.PV, p. ej. CRP_IN.OUTV.
   En ERR_MON.ER se indica el error de regulación.
3. Parametrice la vigilancia del error de regulación en la estructura ERR_MON_Instance de la multiinstancia.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR_ER.



Consulte también

Acondicionamiento del valor real (Página 10168)

Control anticipativo (Página 10175)

Descripción de A_DEAD_B (Página 7234)

Descripción de DEADBAND (Página 7243)

Descripción de ERR_MON (Página 7247)

Control anticipativo

Instrucciones para el acondicionamiento de la magnitud perturbadora

Se pueden utilizar las siguientes instrucciones.

- CRP_IN: escalar valor de periferia
- DIF: reaccionar a cambios rápidos
- LAG1ST: filtrar perturbaciones
- NON_LIN: linealizar magnitud perturbadora
- SCALE_M: escalar magnitud perturbadora
- NORM: normalizar magnitud perturbadora

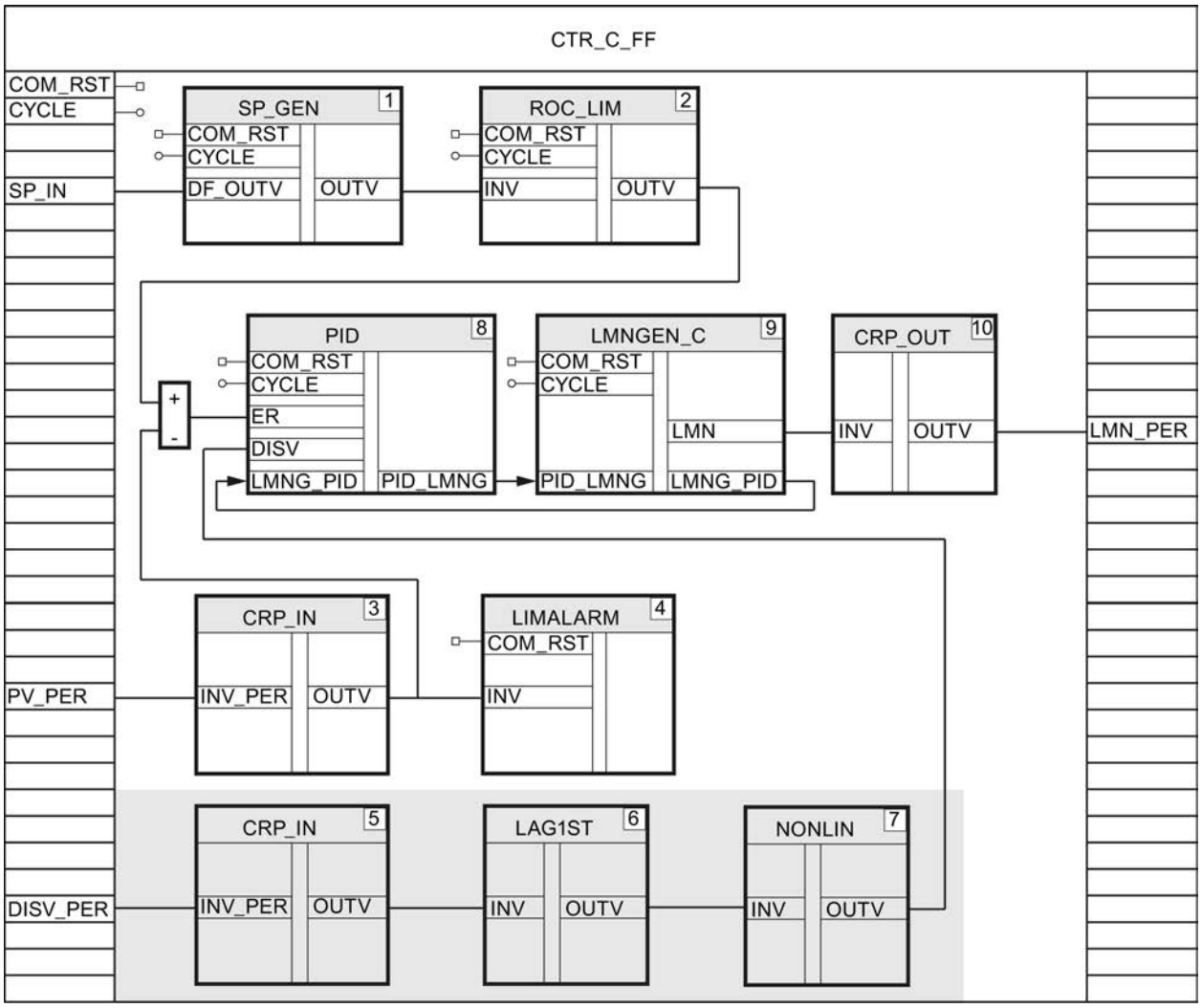
Ejemplo de acondicionamiento de la magnitud perturbadora

La magnitud perturbadora tiene formato de periferia, la señal debe filtrarse y linealizarse por medio de una curva característica.

Para acondicionar la magnitud perturbadora tal como se describe en el ejemplo, proceda del siguiente modo:

1. Declare en el FB de regulación un parámetro de entrada DISV_PER del tipo de datos "Word".
2. Interconecte DISV_PER con la entrada de periferia en la que se mide la magnitud perturbadora.
3. En el FB de regulación llame la instrucción CRP_IN como multiinstancia, p. ej. con el nombre CRP_IN_Instance_DISV.
4. Interconecte DISV_PER con CRP_IN.INV_PER.
5. Parametrice el factor y offset de la escala en la estructura CRP_IN_Instance_DISV de la multiinstancia.
6. En el FB de regulación llame la instrucción LAG1ST como multiinstancia e interconecte CRP_IN.OUTV con LAG1ST.INV mediante una variable temporal.
7. Parametrice el tiempo de retardo en la estructura LAG1ST_Instance de la multiinstancia.
8. En el FB de regulación llame la instrucción NONLIN como multiinstancia e interconecte LAG1ST.OUTV con NONLIN.INV mediante una variable temporal.
9. Cree un bloque de datos global DB_NBR_NONLIN en el que se guardarán los nodos de la curva característica.
10. Parametrice el número del bloque de datos global con la curva característica en la estructura NONLIN_Instance de la multiinstancia.
11. En NONLIN.OUTV se indica la magnitud perturbadora acondicionada.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación CTR_C_FF.



Consulte también

Determinar y acondicionar el error de regulación (Página 10170)

Crear un regulador continuo (Página 10178)

Crear un regulador de impulsos (Página 10179)

Crear un regulador paso a paso (Página 10183)

Descripción de CRP_IN (Página 7238)

Descripción de DIF (Página 7245)

Descripción de LAG1ST (Página 7253)

Descripción de NONLIN (Página 7277)

Descripción de SCALE_M (Página 7311)

Descripción NORM (Página 7280)

Crear un regulador continuo

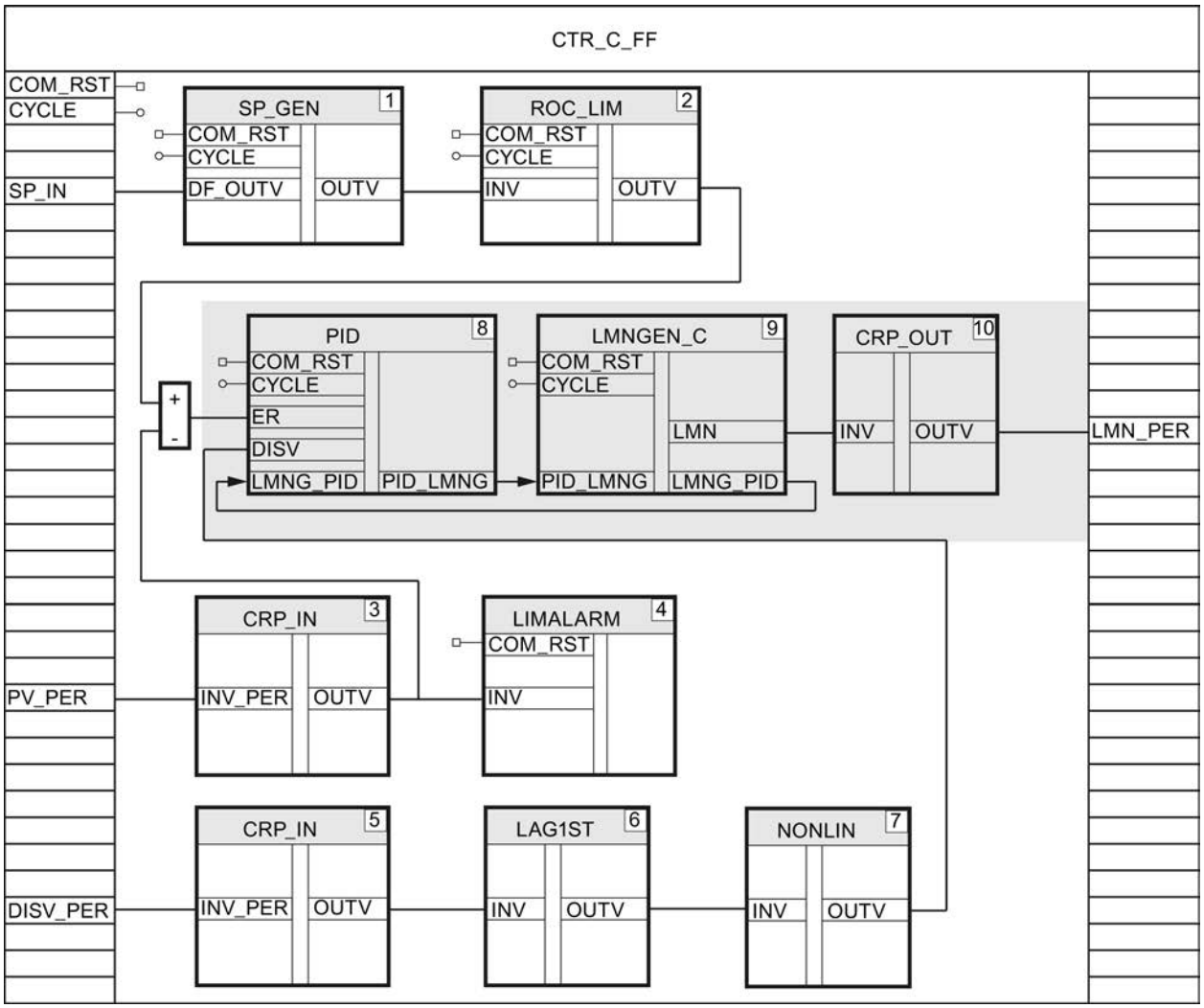
Para un regulador continuo con variable manipulada analógica se necesitan las instrucciones PID, LMNGEN_C y CRP_OUT.

Procedimiento

Para programar un regulador continuo con variable manipulada analógica, proceda del siguiente modo:

1. Declare en el FB de regulación un parámetro de entrada/salida MAN del tipo de datos "Real".
   MAN se necesita para poner en servicio el regulador con el PID Self-Tuner.
2. En el FB de regulación llame las instrucciones PID, LMNGEN_C y CRP_OUT como multiinstancia.
3. Declare en el FB de regulación un parámetro de entrada/salida MAN del tipo de datos "Real" e interconecte MAN con LMNGEN_C.MAN.
4. En el DB de regulación, parametrice el comportamiento del algoritmo regulador en la estructura PID_Instance, p. ej. si se utilizará la acción D.
5. En la estructura LMNGEN_C_Instance de la multiinstancia parametrice p. ej. los límites de la variable manipulada y de la velocidad de cambio de la variable manipulada.
6. En la estructura CRP_OUT_Instance de la multiinstancia parametrice el factor y el offset para la conversión del valor en coma flotante al formato de periferia.
7. Utilizando una variable temporal, interconecte PID.ER con el error de regulación, p. ej. ROC_LIM.OUTV - CRP_IN.OUTV o ERR_MON.ER.
8. En caso de control anticipativo, interconecte PID.DISV con la señal acondicionada de la magnitud perturbadora, p. ej. NONLIN.OUTV, utilizando una variable temporal.
9. Utilizando la instrucción BLKMOV, interconecte el parámetro de salida PID.PID_LMNG con el parámetro de entrada LMNGEN_C.PID_LMNG y el parámetro de salida LMNGEN_C.LMNG_PID con el parámetro de entrada PID.LMNG_PID.
10. Interconecte LMNGEN_C.LMN con CRP_OUT.INV utilizando una variable temporal.
11. Declare en el FB de regulación un parámetro de salida LMN_PER del tipo de datos "Word" e interconecte CRP_OUT.OUTV con LMN_PER.
12. Interconecte LMN_PER con la salida analógica de la CPU que controla el actuador.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación CTR_C_FF .



Consulte también

Control anticipativo (Página 10172)

Descripción de PID (Página 7285)

Descripción de LMNGEN_C (Página 7264)

Descripción de CRP_OUT (Página 7240)

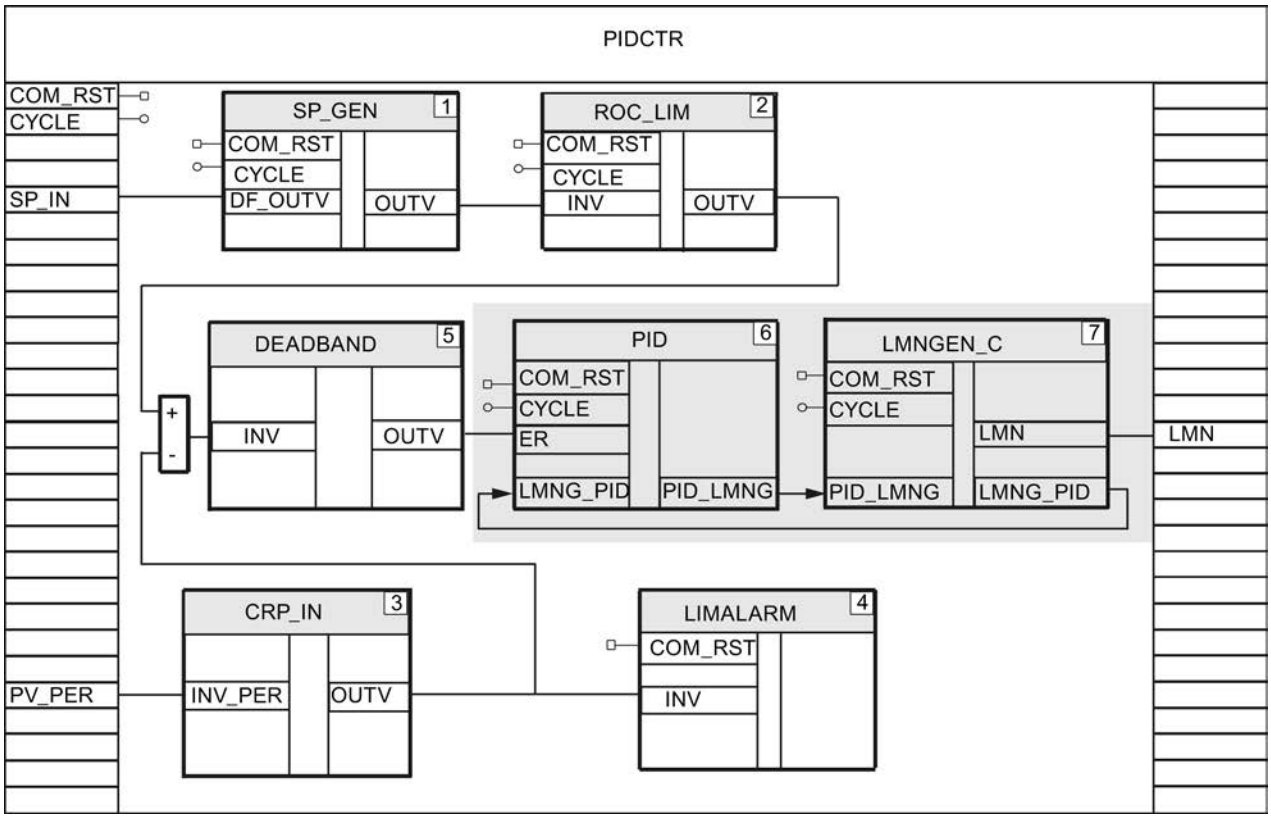
Crear un regulador de impulsos

Para un regulador de impulsos, genere una variable manipulada continua en formato de coma flotante con las instrucciones PID y LMNGEN_C. A partir de esta variable manipulada continua, genere un tren de impulsos con la instrucción PULSEGEN_M mediante la modulación de ancho de impulsos.

Generar una variable manipulada continua en formato de coma flotante

1. Declare en el FB de regulación, en este caso PIDCTR, un parámetro de salida LMN del tipo de datos "Real".
2. En el FB de regulación llame las instrucciones PID y LMNGEN_C y como multiinstancia.
3. Interconecte y parametrice PID y LMNGEN_C tal como se describe en el regulador continuo.
4. Interconecte LMNGEN_C.LMN con el parámetro de salida LMN del FB de regulación.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR.



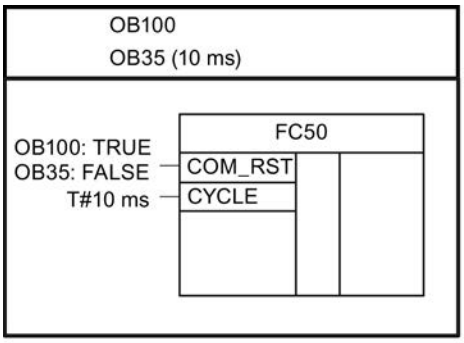
Generar un tren de impulsos

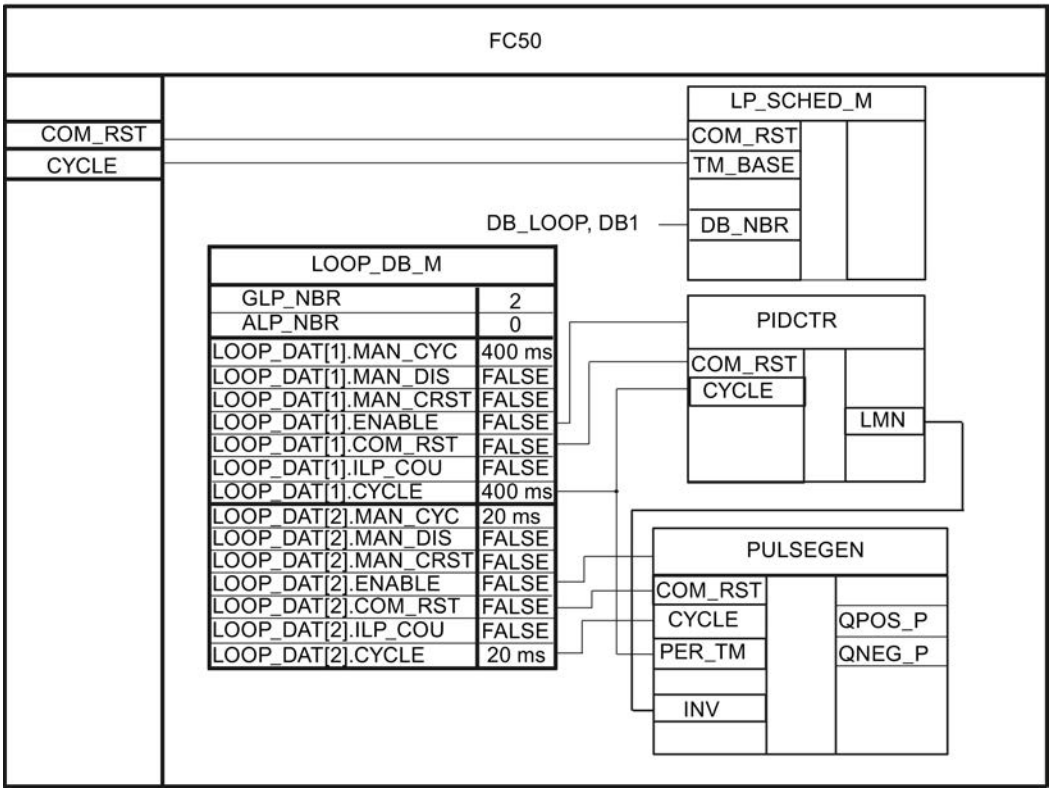
Para generar un tren de impulsos a partir de una variable manipulada continua, proceda del siguiente modo:

1. Cree una función, p. ej. FC50, con los parámetros de entrada COM_RST y CYCLE.
2. Cree un bloque de datos global LOOP_DB_M en el que se parametrizarán los tiempos de ciclo para la llamada del FB de regulación y del generador de impulsos PULSEGEN_M. La relación entre el tiempo de ciclo para la llamada del FB de regulación y PULSEGEN_M debería ser ≥ 20.
3. Llame la instrucción LP_SCHED_M en la función.
4. Interconecte LP_SCHED_M.DB_NBR con el bloque de datos global LOOP_DB_M.

5. Llame el FB de regulación, p. ej. PIDCTR, si LOOP_DB_M.LOOP_DAT[1].ENABLE = TRUE.
6. Interconecte LOOP_DB_M.LOOP_DAT[1].COM_RST y LOOP_DB_M.LOOP_DAT[1].CYCLE con los parámetros de entrada COM_RST y CYCLE del FB regulador.
7. Defina LOOP_DB_M.LOOP_DAT[1].ENABLE = FALSE.
8. Llame la instrucción PULSEGEN_M cuando LOOP_DB_M.LOOP_DAT[2].ENABLE = TRUE.
9. Interconecte LOOP_DB_M.LOOP_DAT[2].COM_RST y LOOP_DB_M.LOOP_DAT[2].CYCLE con PULSEGEN_M.COM_RST y PULSEGEN_M.CYCLE.
10. Interconecte PULSEGEN_M.INV con el parámetro de salida LMN del FB de regulación.
11. Interconecte PULSEGEN_M.PER_TM con el parámetro de entrada CYCLE del FB de regulación.
12. Defina LOOP_DB_M.LOOP_DAT[2].ENABLE = FALSE.

13. Interconecte PULSEGEN_M.QPOS_P y, dado el caso, PULSEGEN_M.QNEG_P con las salidas digitales de la CPU que controlan el actuador.
14. Llame la función en el OB 100 y en el OB de alarma cíclica.





Consulte también

Control anticipativo (Página 10172)

Descripción de PID (Página 7285)

Descripción de LMNGEN_C (Página 7264)

Descripción de PULSEGEN_M (Página 7289)

Descripción de LP_SCHED_M (Página 7318)

Crear un regulador paso a paso

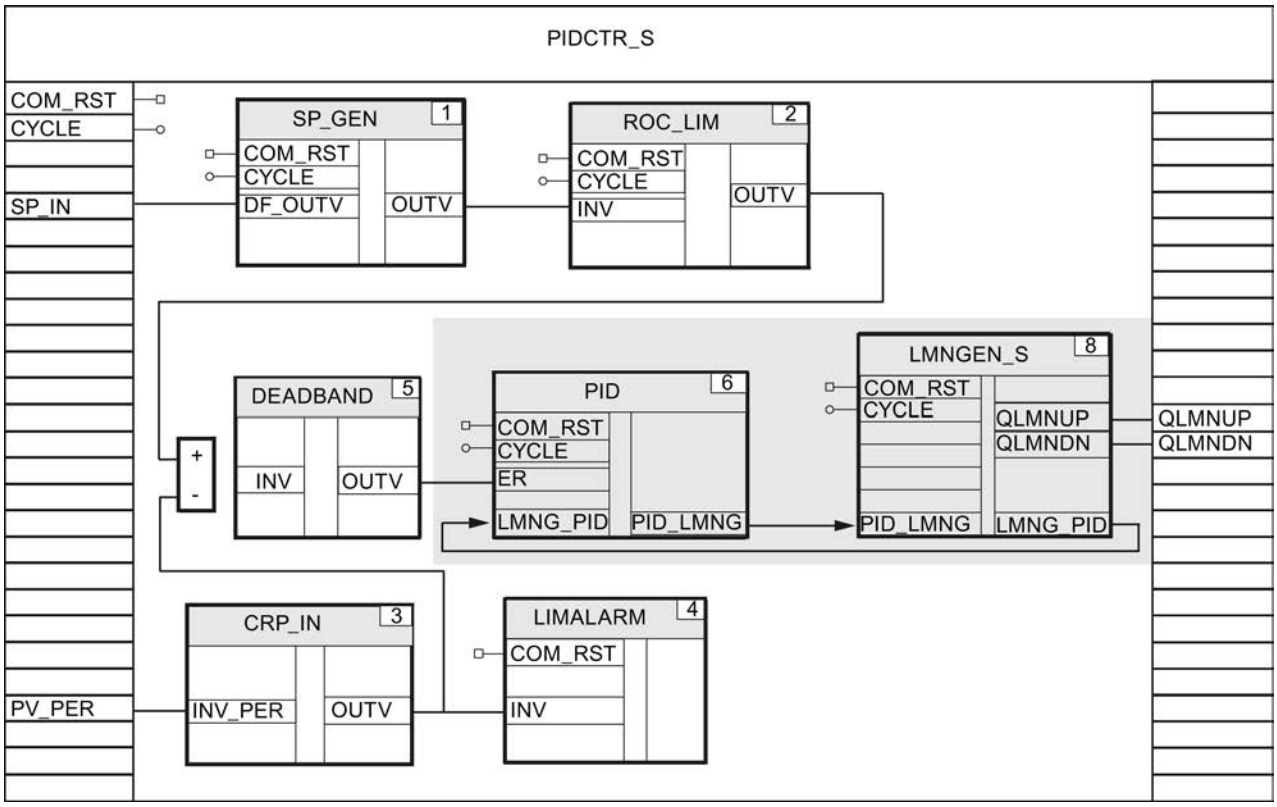
Para un regulador paso a paso se necesitan las instrucciones PID y LMNGEN_S.

Procedimiento

Para programar un regulador paso a paso, proceda del siguiente modo:

1. En el FB de regulación llame las instrucciones PID y LMNGEN_S y como multiinstancia.
2. Parametrice el comportamiento del algoritmo regulador en la estructura PID_Instance de la multiinstancia, p. ej. si se utilizará la acción D.
3. En la estructura LMNGEN_S_Instance de la multiinstancia, parametrice p. ej. que no deben utilizarse realimentaciones de posición, LMNR_ON = FALSE.
4. Utilizando una variable temporal, interconecte PID.ER con el error de regulación, p. ej. DEADBAND.OUTV o ERR_MON.ER.
5. En caso de control anticipativo, interconecte PID.DISV con la señal acondicionada de la magnitud perturbadora, p. ej. CRP_IN.OUTV, utilizando una variable temporal.
6. Utilizando la instrucción BLKMOV, interconecte el parámetro de salida PID.PID_LMNG con el parámetro de entrada LMNGEN_S.PID_LMNG y el parámetro de salida LMNGEN_S.LMNG_PID con el parámetro de entrada PID.LMNG_PID.
7. Declare en el FB de regulación los parámetros de salida QLMNUP y QLMNDN del tipo de datos "Bool".
8. Interconecte LNMGEN_S.QLMNUP y LMNGEN_S. QLMNDN con estos parámetros de salida.
9. Interconecte los parámetros de salida del FB de regulación con las salidas digitales de la CPU que controlan el actuador.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR_S.



Consulte también

Control anticipativo (Página 10172)

Acondicionamiento de la realimentación de posición (Página 10184)

Descripción de PID (Página 7285)

Descripción de LMNGEN_S (Página 7269)

Acondicionamiento de la realimentación de posición

La configuración de la realimentación de posición depende del actuador utilizado.

- Actuador con realimentación de posición analógica
- Actuador con señales de tope digitales

Requisitos

Debe haberse programado un regulador paso a paso.

Crear un regulador paso a paso con realimentación de posición analógica

Para acondicionar la realimentación de posición analógica se recomiendan las instrucciones siguientes.

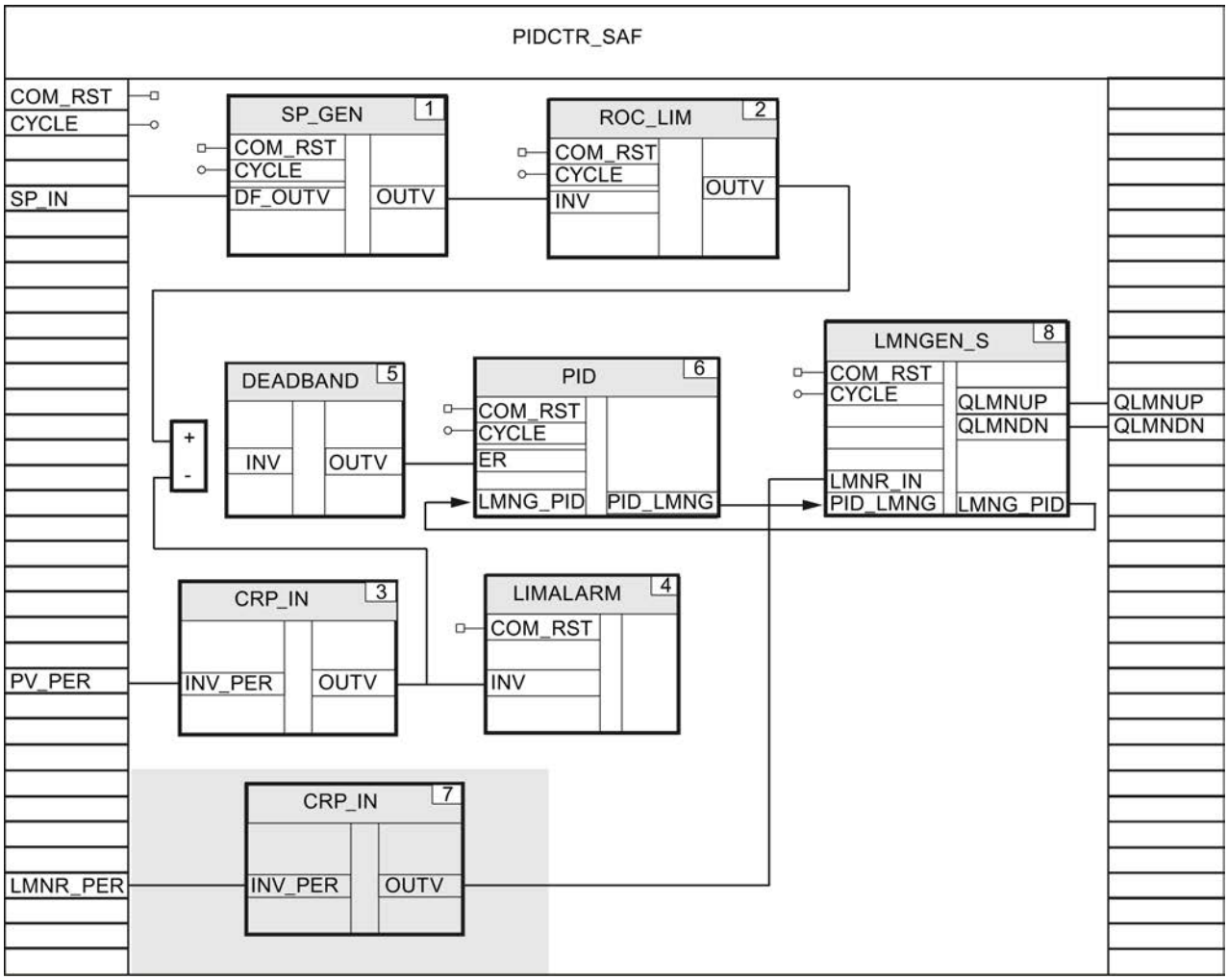
- CRP_IN: escalar valor de periferia
- NONLIN: linealizar realimentación de posición
- SCALE_M: escalar realimentación de posición
- NORM: normalizar realimentación de posición

Ejemplo de una realimentación de posición analógica

Un actuador proporciona una realimentación de posición en formato de periferia que debe tenerse en cuenta al calcular la variable manipulada.

1. Declare en el FB de regulación un parámetro de entrada LMNR_PER del tipo de datos "Word".
2. Interconecte LMNR_PER con la entrada de periferia.
3. En el FB de regulación llame la instrucción CRP_IN como multiinstancia, p. ej. con el nombre CRP_IN_Instance_LMNR.
4. Interconecte LMNR_PER con CRP_IN.INV_PER.
5. Parametrice el factor y offset de la escala en la estructura CRP_IN_Instance_LMNR de la multiinstancia.
6. Interconecte CRP_IN.OUTV con LMNGEN_S.LMNR_IN utilizando una variable temporal.
7. En la estructura LMNGEN_S_Instance de la multiinstancia active la realimentación de posición con LMNR_ON = TRUE.

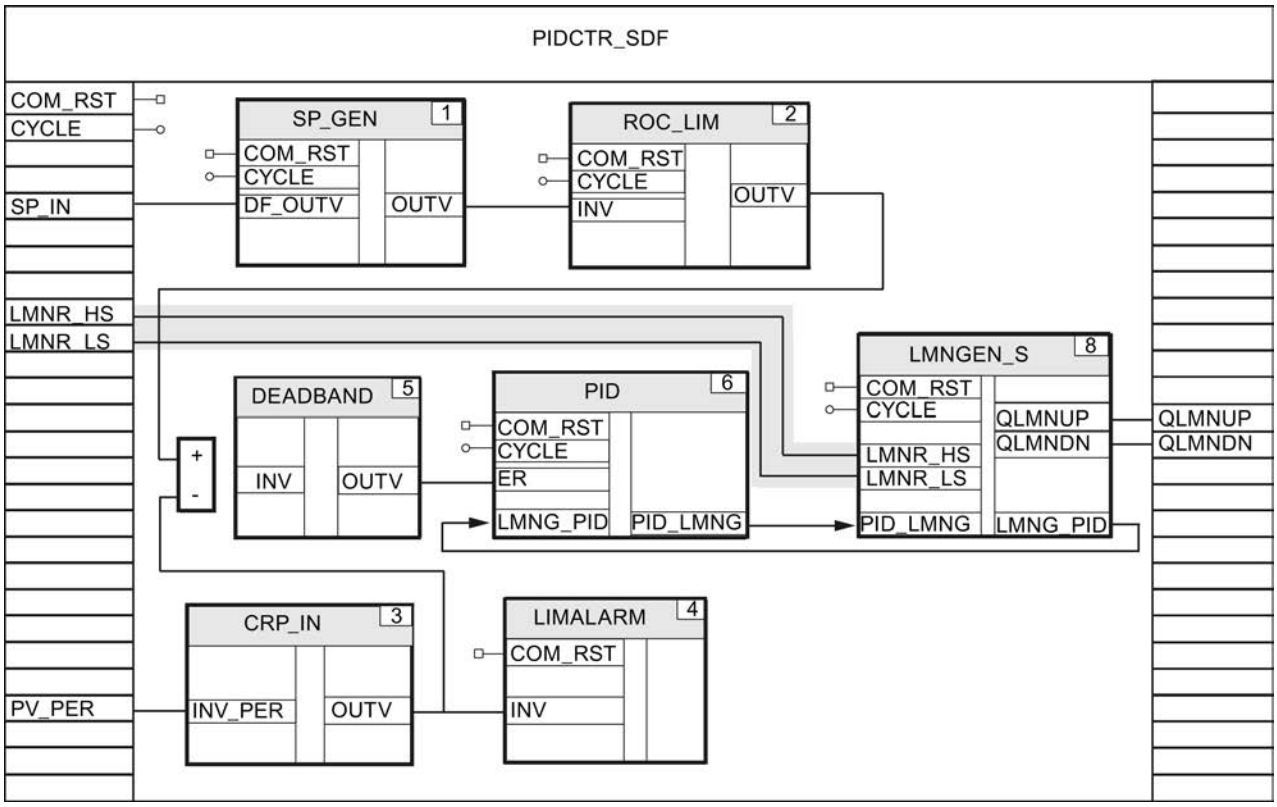
La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR_SAF.



Crear un regulador paso a paso con realimentación de posición digital

1. Declare en el FB de regulación los parámetros de entrada LMNR_HS y LMNR_LS del tipo de datos "Bool".
2. Interconecte los parámetros de entrada LMNR_HS y LMNR_LS con LMNGEN_S.LMNR_HS y LMNGEN_S.LMNR_LS.
3. En la estructura LMNGEN_S_Instance active la realimentación de posición con LMNR_ON = TRUE.

La figura siguiente muestra cómo se llaman e interconectan las instrucciones del Modular PID Control en el FB de regulación PIDCTR_SDF.



Consulte también

Crear un regulador paso a paso (Página 10180)

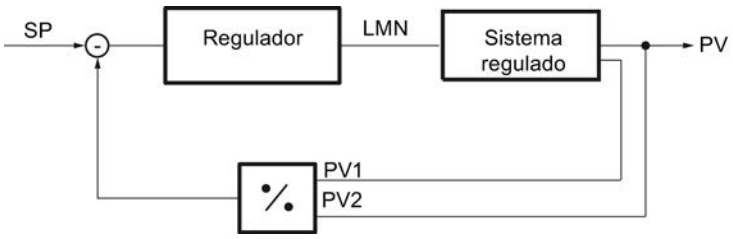
Descripción de CRP_IN (Página 7238)

Descripción de NONLIN (Página 7277)

Estructuras de regulador complejas

Regulador de relación monobucle

La figura siguiente muestra un lazo de regulación típico para un regulador de relación monobucle.



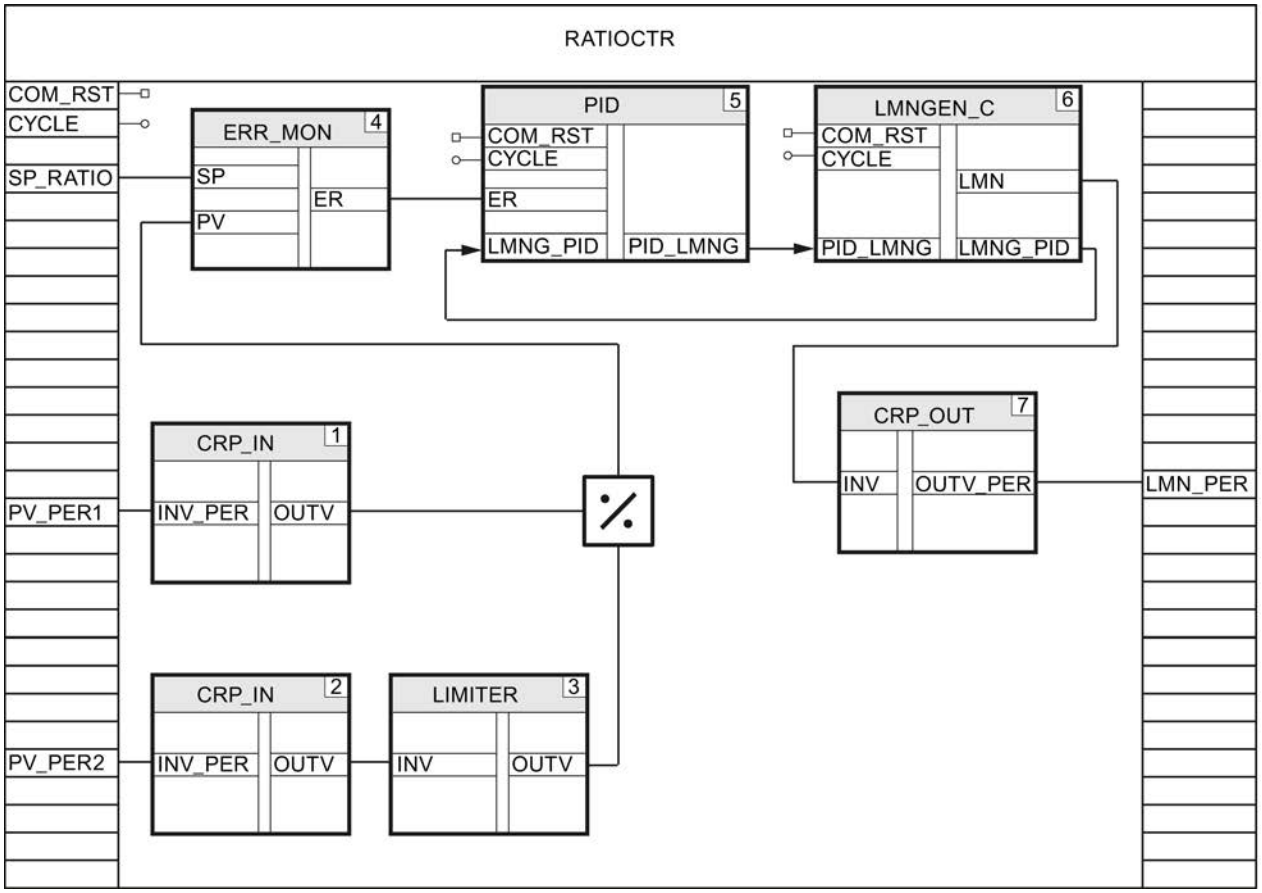
La relación entre dos valores reales se regula en una consigna ajustada con un algoritmo PID.

Procedimiento

Para crear un regulador de relación monobucle, proceda del siguiente modo:

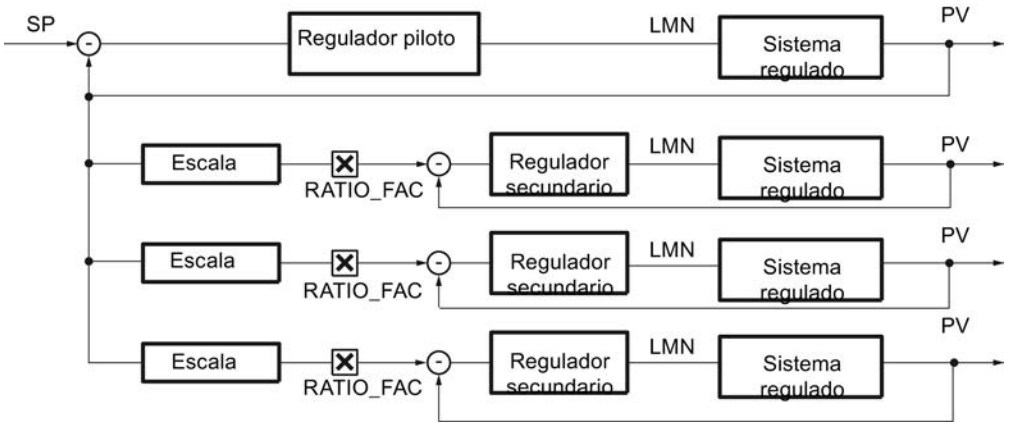
1. En el FB de regulación programe el acondicionamiento de los valores reales tal como se representa abajo.
2. Para evitar una división entre cero, hay que limitar el rango de valores del valor real PV_PER2 a valores > 0 con la instrucción LIMITER.

La figura siguiente muestra un FB de regulación, en este caso RATIOCTR, para un regulador de relación monobucle con variable manipulada continua.



Regulador de relación multibucle

La figura siguiente muestra un lazo de regulación típico para un regulador de relación multibucle.



Programar un lazo de regulación

Para programar este ejemplo con Modular PID Control, proceda del siguiente modo:

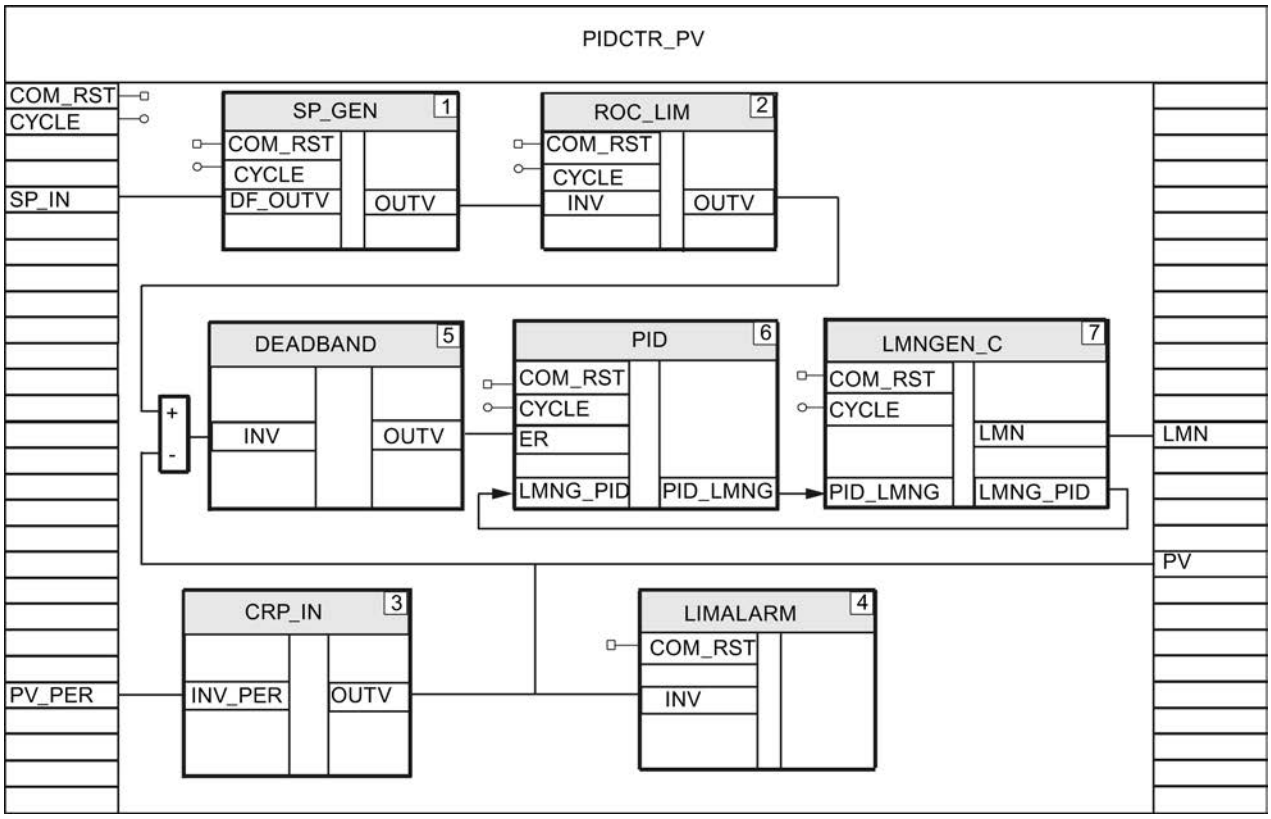
1. Programe un FB de regulación para el regulador piloto, p. ej. PIDCTR_PV.
2. Programe uno o varios FB de regulación para los reguladores secundarios, p. ej. RB_CTR_S.
3. Programe una función con la que se llamen e interconecten los cuatro reguladores.

Programar el regulador piloto

Para programar el regulador piloto, proceda del siguiente modo:

1. Programe el regulador piloto de acuerdo con sus necesidades, tal como se describe en el capítulo Programar el FB de regulación (Página 10166).
2. Declare en el FB de regulación, en este caso PIDCTR_PV, un parámetro de salida PV del tipo de datos "Real" en el que se muestre el valor real acondicionado del regulador piloto.

La figura siguiente muestra la interconexión para el regulador piloto siguiendo el ejemplo del FB de regulación PIDCTR_PV.

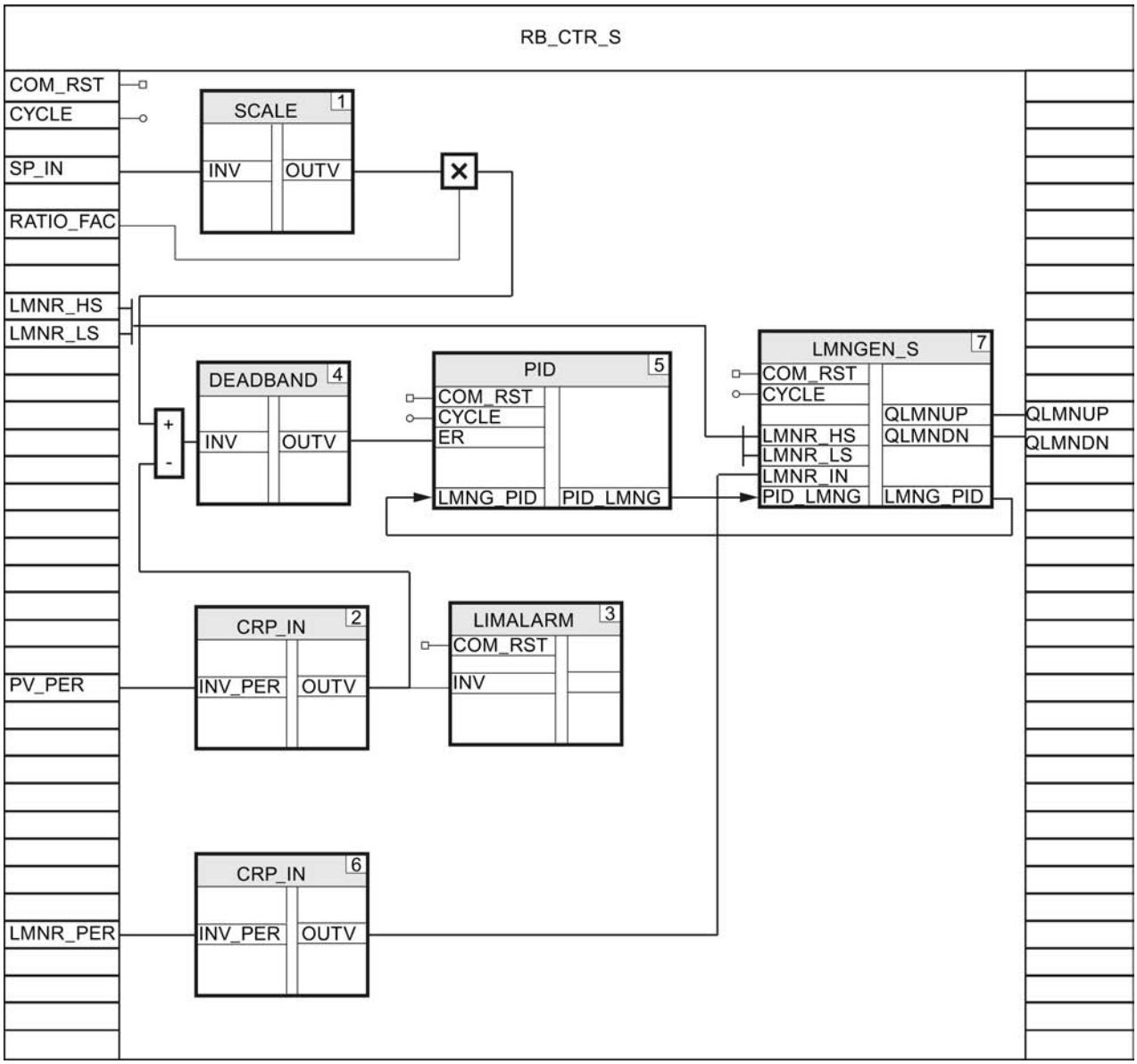


Programar los reguladores secundarios

Para programar los reguladores secundarios, proceda del siguiente modo:

1. Programe los reguladores secundarios de acuerdo con sus necesidades, tal como se describe en el capítulo Programar el FB de regulación (Página 10166).
2. Declare en el FB de regulación, en este caso RB_CTR_S, un parámetro de entrada RATIO_FAC del tipo de datos "Int". Con este valor se multiplica la consigna acondicionada, p. ej. SCALE.OUTV, del regulador secundario.

La figura siguiente muestra la interconexión para el regulador secundario siguiendo el ejemplo del FB de regulación RB_CTR_S.



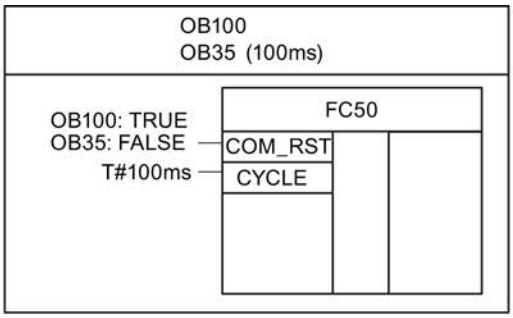
Llamar e interconectar reguladores

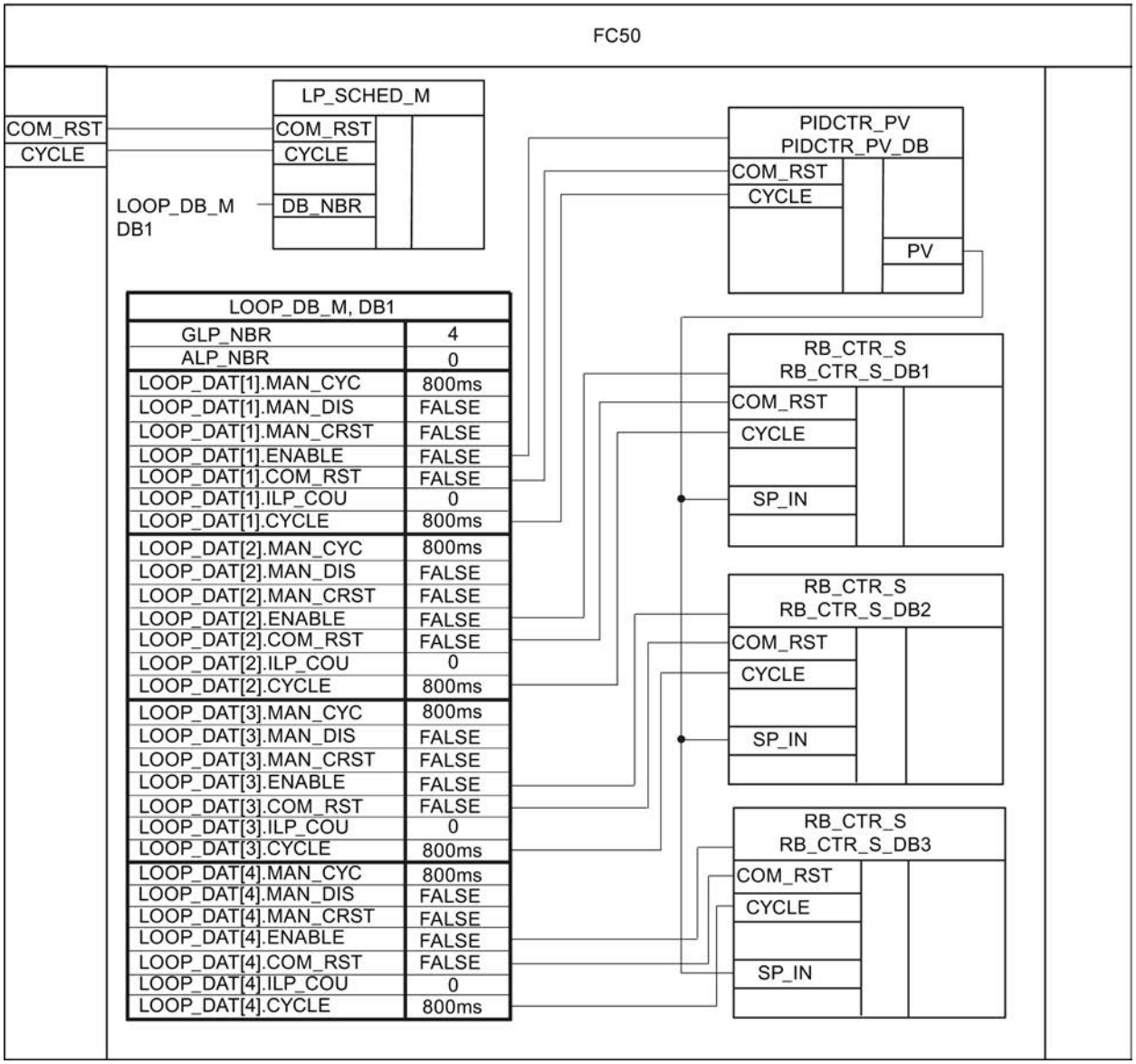
Para llamar e interconectar los reguladores piloto y secundarios, proceda del siguiente modo:

1. Cree una función, p. ej. FC50, con los parámetros de entrada COM_RST y CYCLE.
2. Llame la instrucción LP_SCHED_M en la función con el bloque de datos de instancia LP_SCHED_M_DB.
3. Cree un bloque de datos global DB_LOOP_M en el que se parametrizarán el número de reguladores y los tiempos de ciclo para la llamada del regulador piloto y de los reguladores secundarios.
   Los tiempos de ciclo de los reguladores secundarios deben ser iguales o inferiores al tiempo de ciclo del regulador piloto.

4. Llame el FB de regulación para el regulador piloto, en este caso PIDCTR_PV, si LOOP_DB_M.LOOP_DAT[1].ENABLE = TRUE.
5. Interconecte LOOP_DB_M.LOOP_DAT[1].COM_RST y LOOP_DB_M.LOOP_DAT[1].CYCLE con los parámetros de entrada COM_RST y CYCLE del regulador piloto.
6. Defina LOOP_DB_M.LOOP_DAT[1].ENABLE = FALSE.
7. Llame el FB de regulación para el primer regulador secundario, en este caso RB_CTR_S, si LOOP_DB_M.LOOP_DAT[2].ENABLE = TRUE.
8. Interconecte LOOP_DB_M.LOOP_DAT[2].COM_RST y LOOP_DB_M.LOOP_DAT[2].CYCLE con COM_RST y CYCLE del regulador secundario.
9. Interconecte el parámetro de salida PV del regulador piloto con el parámetro de entrada SP_IN del regulador secundario.
10. Defina LOOP_DB_M.LOOP_DAT[2].ENABLE = FALSE.
11. Repita los cuatro últimos pasos para los dos reguladores secundarios restantes. Incremente el contador de la matriz para cada regulador.
12. Interconecte las variables manipuladas con las salidas de la CPU.
13. Llame la función en el OB 100 y en el OB de alarma cíclica.

La figura siguiente muestra las llamadas y la interconexión de los reguladores siguiendo el ejemplo de la función FC50.

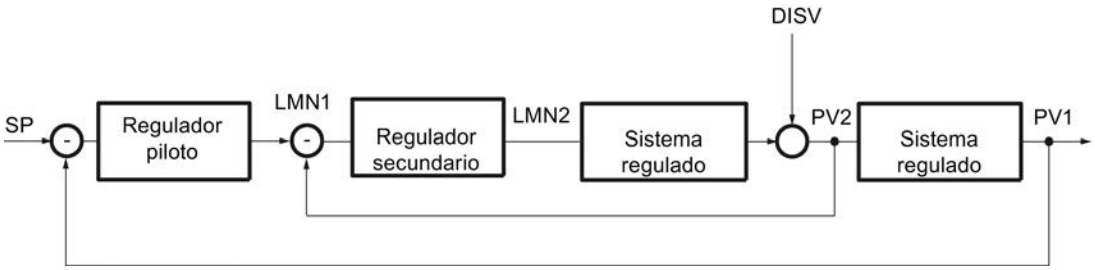




LOOP_DB_M, DB1	
GLP_NBR	4
ALP_NBR	0
LOOP_DAT[1].MAN_CYC	800ms
LOOP_DAT[1].MAN_DIS	FALSE
LOOP_DAT[1].MAN_CRST	FALSE
LOOP_DAT[1].ENABLE	FALSE
LOOP_DAT[1].COM_RST	FALSE
LOOP_DAT[1].ILP_COU	0
LOOP_DAT[1].CYCLE	800ms
LOOP_DAT[2].MAN_CYC	800ms
LOOP_DAT[2].MAN_DIS	FALSE
LOOP_DAT[2].MAN_CRST	FALSE
LOOP_DAT[2].ENABLE	FALSE
LOOP_DAT[2].COM_RST	FALSE
LOOP_DAT[2].ILP_COU	0
LOOP_DAT[2].CYCLE	800ms
LOOP_DAT[3].MAN_CYC	800ms
LOOP_DAT[3].MAN_DIS	FALSE
LOOP_DAT[3].MAN_CRST	FALSE
LOOP_DAT[3].ENABLE	FALSE
LOOP_DAT[3].COM_RST	FALSE
LOOP_DAT[3].ILP_COU	0
LOOP_DAT[3].CYCLE	800ms
LOOP_DAT[4].MAN_CYC	800ms
LOOP_DAT[4].MAN_DIS	FALSE
LOOP_DAT[4].MAN_CRST	FALSE
LOOP_DAT[4].ENABLE	FALSE
LOOP_DAT[4].COM_RST	FALSE
LOOP_DAT[4].ILP_COU	0
LOOP_DAT[4].CYCLE	800ms

Regulador en cascada

La figura siguiente muestra un lazo de regulación típico de un regulador en cascada.

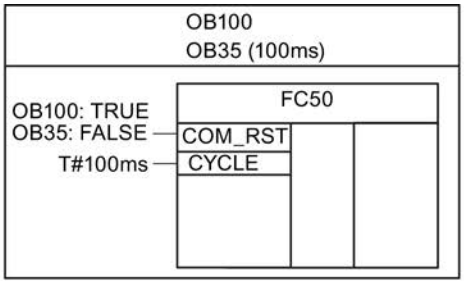


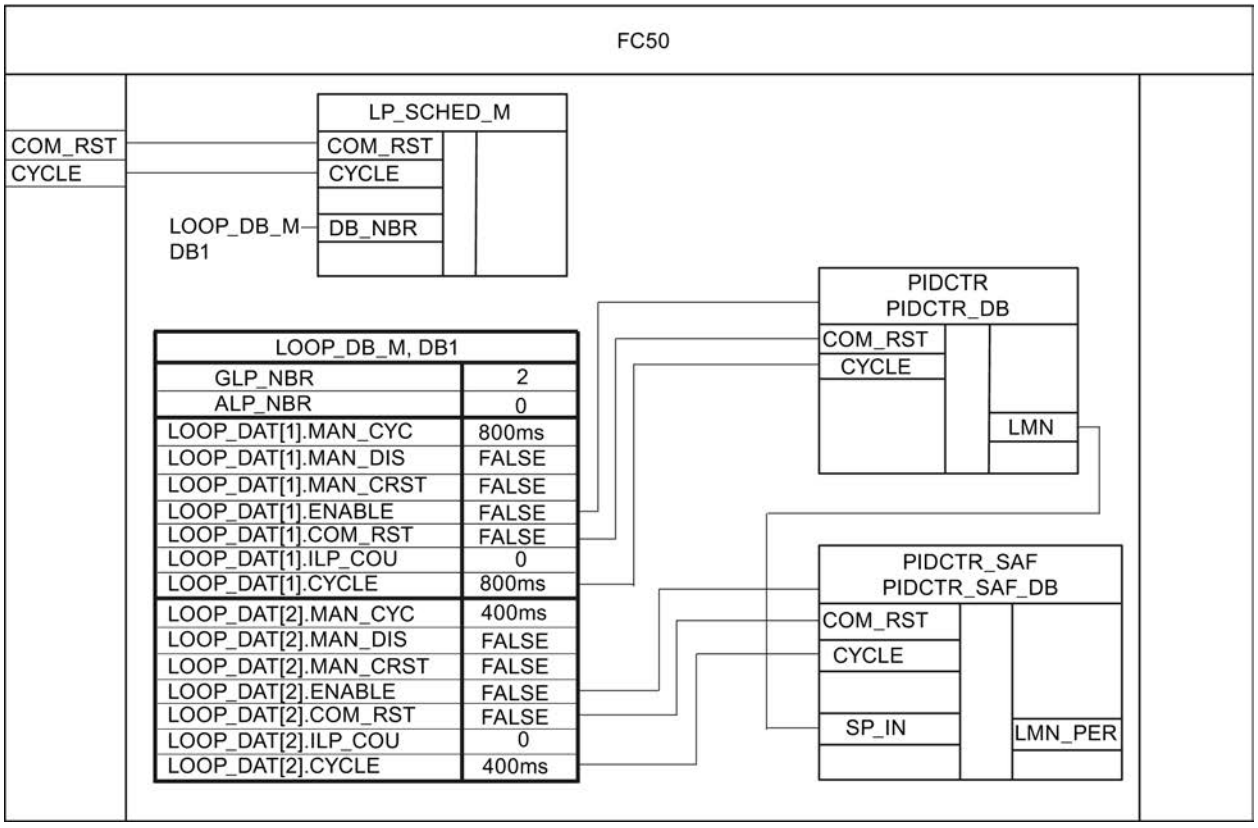
Programar un lazo de regulación

Para programar este ejemplo con Modular PID Control, proceda del siguiente modo:

1. Programe un FB de regulación para el regulador piloto, p. ej. PIDCTR.
2. Programe un FB de regulación para el regulador secundario, p. ej. PIDCTR_SAF.
3. Programe una función con la que se llamen e interconecten los dos reguladores.
4. Llame la instrucción LP_SCHED_M en la función con el bloque de datos de instancia LP_SCHED_M_DB.
5. Cree un bloque de datos global LOOP_DB_M en el que se parametrizarán el número de reguladores y los tiempos de ciclo para la llamada del regulador piloto y de los reguladores secundarios.
   El tiempo de ciclo del regulador secundario debe ser igual o inferior al tiempo de ciclo del regulador piloto.
6. Llame el FB de regulación para el regulador piloto si LOOP_DB_M.LOOP_DAT[1].ENABLE = TRUE.
7. Interconecte LOOP_DB_M.LOOP_DAT[1].COM_RST y LOOP_DB_M.LOOP_DAT[1].CYCLE con los parámetros de entrada COM_RST y CYCLE del regulador piloto.
8. Defina LOOP_DB_M.LOOP_DAT[1].ENABLE = FALSE.
9. Llame el FB de regulación para el regulador secundario si LOOP_DB_M.LOOP_DAT[2].ENABLE = TRUE.
10. Interconecte LOOP_DB_M.LOOP_DAT[2].COM_RST y LOOP_DB_M.LOOP_DAT[2].CYCLE con COM_RST y CYCLE del regulador secundario.
11. Interconecte el parámetro de salida LMN del regulador piloto con el parámetro de entrada SP_IN del regulador secundario.
12. Defina LOOP_DB_M.LOOP_DAT[2].ENABLE = FALSE.
13. Interconecte la variable manipulada del regulador secundario con la salida analógica de la CPU que controla el actuador.
14. Llame la función en el OB 100 y en el OB de alarma cíclica.

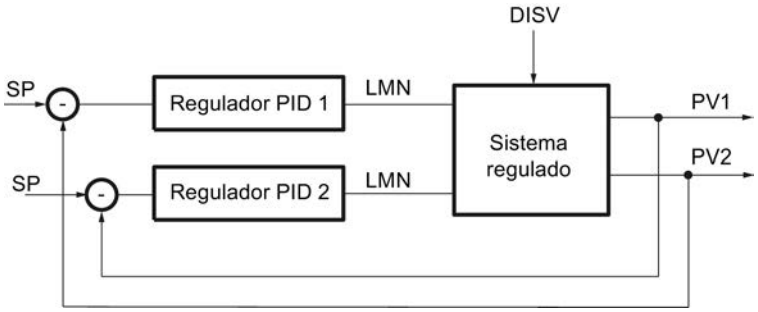
La figura siguiente muestra las llamadas y la interconexión de los reguladores siguiendo el ejemplo de la función FC50.





Regulador multivariable

La figura siguiente muestra un lazo de regulación típico para un regulador multivariable. Dos reguladores PID con salida continua se aplican al sistema regulado.

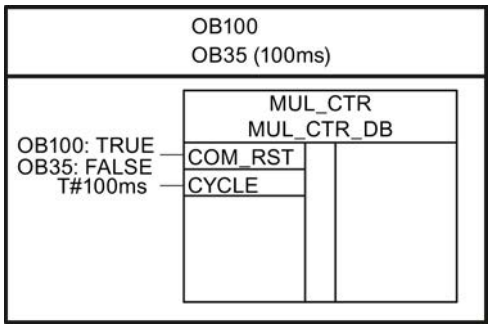


Programar un lazo de regulación

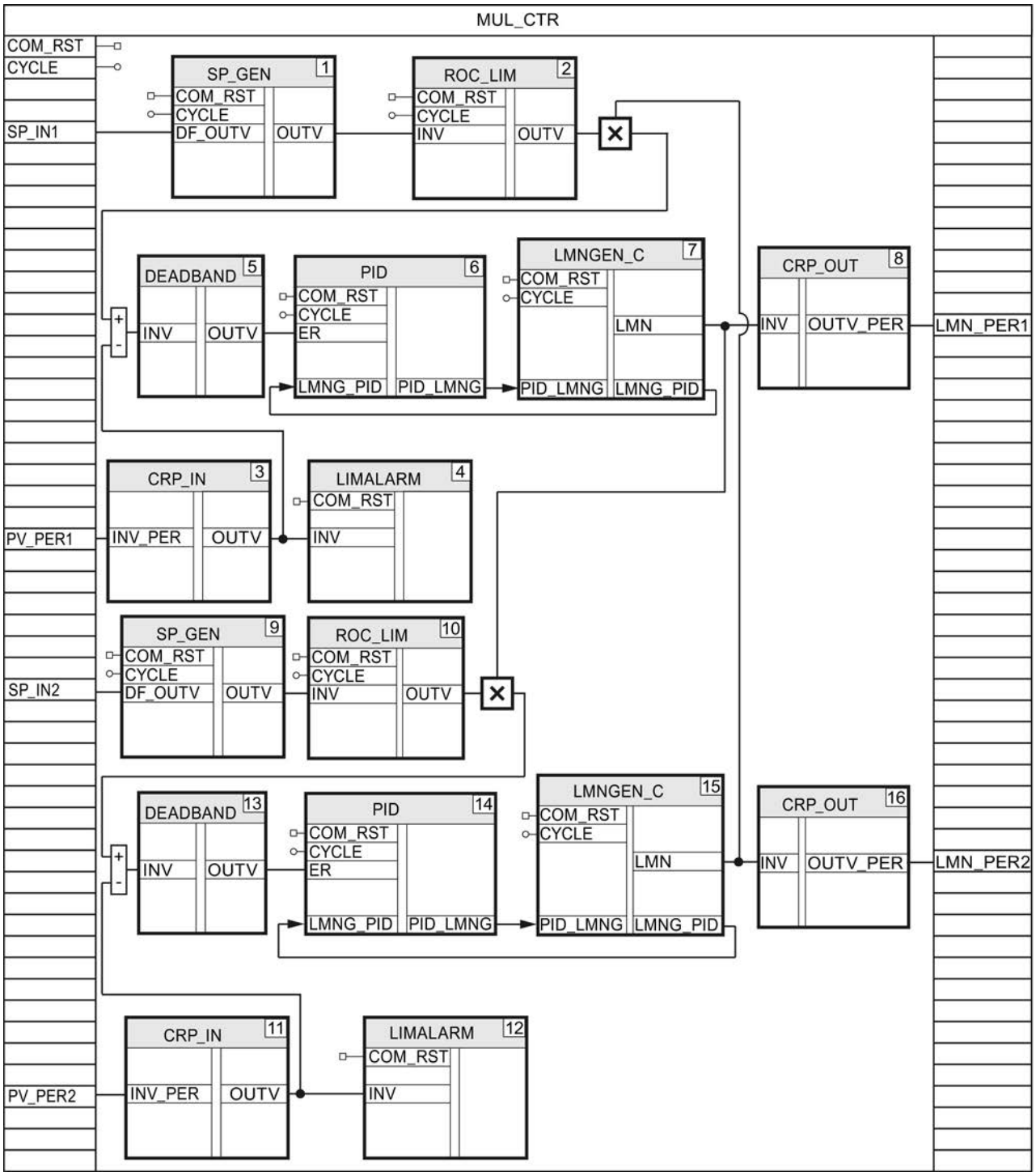
Para programar este ejemplo con Modular PID Control, proceda del siguiente modo:

1. Declare en el FB de regulación, en este caso MUL_CTR, los parámetros de entrada siguientes:
   - COM_RST del tipo de datos "Bool"
   - CYCLE del tipo de datos "Time"
   - SP_IN1 y SP_IN2del tipo de datos "Real"
   - PV_PER1 y PV_PER2 del tipo de datos "Word"
2. Declare en el FB de regulación los parámetros de salida LMN_PER1 y LMN_PER2 del tipo de datos "Word".
3. Programe para ambos reguladores PID el acondicionamiento de la consigna y el valor real así como el cálculo de la magnitud manipulada.
4. Para calcular el error de regulación del primer regulador PID, reste el valor real acondicionado PV_PER1 del producto de la consigna SP_IN1 y la variable manipulada del segundo regulador PID.
5. Para calcular el error de regulación del segundo regulador PID, reste el valor real acondicionado PV_PER2 del producto de la consigna SP_IN2 y la variable manipulada del primer regulador PID.
6. Interconecte las variables manipuladas LMN_PER1 y LMN_PER2 con dos salidas analógicas que controlen los actuadores.
7. Llame el FB de regulación en el OB 100 y en el OB de alarma cíclica.

La figura siguiente muestra la llamada siguiendo el ejemplo del FB de regulación MUL_CTR.



La figura siguiente muestra la interconexión de los reguladores siguiendo el ejemplo del FB de regulación MUL_CTR.


MUL_CTR
COM_RST
CYCLE
SP_IN1
SP_GEN
1
COM_RST
CYCLE
DF_OUTV
OUTV
ROC_LIM
2
COM_RST
CYCLE
INV
OUTV
DEADBAND
5
INV
OUTV
PID
6
COM_RST
CYCLE
ER
LMNG_PID
PID_LMNG
LMNGEN_C
7
COM_RST
CYCLE
LMN
PID_LMNG
LMNG_PID
CRP_OUT
8
INV
OUTV_PER
LMN_PER1
CRP_IN
3
INV_PER
OUTV
PV_PER1
LIMALARM
4
COM_RST
INV
SP_GEN
9
COM_RST
CYCLE
DF_OUTV
OUTV
SP_IN2
ROC_LIM
10
COM_RST
CYCLE
INV
OUTV
DEADBAND
13
INV
OUTV
PID
14
COM_RST
CYCLE
ER
LMNG_PID
PID_LMNG
LMNGEN_C
15
COM_RST
CYCLE
LMN
PID_LMNG
LMNG_PID
CRP_OUT
16
INV
OUTV_PER
LMN_PER2
CRP_IN
11
INV_PER
OUTV
PV_PER2
LIMALARM
12
COM_RST
INV

Poner en servicio el FB de regulación

Poner en servicio el regulador continuo

El FB de regulación se pone en servicio con el PID Self-Tuner y se determinan los parámetros PID óptimos.

Procedimiento

Para poner en marcha un regulador continuo o un regulador de impulsos, proceda del siguiente modo:

1. Llame el FB de regulación en un OB de alarma cíclica e interconecte los parámetros de entrada y salida.
2. Llame la instrucción TUN_EC en el mismo OB de alarma cíclica.
3. Interconecte la instrucción TUN_EC con el FB de regulación del siguiente modo:
   - TUN_EC.PV con el valor real acondicionado, p. ej. la variable estática CRP_IN_Instance_PV.OUTV o el parámetro de salida PV
   - TUN_EC.LMN con la variable manipulada en formato Real, p. ej. la variable estática LMNGEN_C_Instance.LMN
   - TUN_EC.MAN con el parámetro de entrada/salida para el valor manual, p. ej. MAN.
   - TUN_EC.SP_OUT con el parámetro de entrada para la consigna, p. ej. SP_IN
   - TUN_EC.GAIN con la ganancia proporcional, p. ej. PID_Instance.GAIN
   - TUN_EC.TI con el tiempo integrativo, p. ej. PID_Instance.TI
   - TUN_EC.TD con el tiempo derivativo, p. ej. PID_Instance.TD
   - TUN_EC.TM_LAG con el tiempo de retardo, p. ej. PID_Instance.TM_LAG
4. Haga clic en el símbolo en la instrucción TUN_EC.
   Se abre el editor de puesta en servicio para TUN_EC. El funcionamiento del editor de puesta en servicio se describe en el PID Self-Tuner.

Poner en servicio el regulador paso a paso

El FB de regulación se pone en servicio con el PID Self-Tuner y se determinan los parámetros PID óptimos.

Procedimiento

Para poner en servicio un regulador paso a paso, proceda del siguiente modo:

1. Llame el FB de regulación en un OB de alarma cíclica e interconecte los parámetros de entrada y salida.
2. Llame la instrucción TUN_ES en el mismo OB de alarma cíclica.

3. Interconecte la instrucción TUN_ES con el FB de regulación del siguiente modo:
   - TUN_ES.PV con el valor real acondicionado, p. ej. la variable estática CRP_IN_Instance _PV.OUTV o el parámetro de salida PV
   - TUN_ES.LMNR_IN con la realimentación de posición acondicionada, p. ej. LMNGEN_S_instance.LMNR_IN
   - TUN_ES.C_LMNUP y TUN_ES.C_LMNDN con las señales de la variable manipulada, p. ej. los parámetros de salida QLMN_UP y QLMNDN
   - TUN_ES.LMNR_HS con la señal del tope superior, p. ej. LMNGEN_S_Instance.LMNR_HS
   - TUN_ES.LMNR_ON con el conmutador de la realimentación de posición, p. ej. LMNGEN_S_Instance.LMNR_ON
   - TUN_ES.SP_OUT con el parámetro de entrada para la consigna, p. ej. SP_IN
   - TUN_ES.GAIN con la ganancia proporcional, p. ej. PID_Instance.GAIN
   - TUN_ES.TI con el tiempo integrativo, p. ej. PID_Instance.TI
   - TUN_ES.TD con el tiempo derivativo, p. ej. PID_Instance.TD
   - TUN_ES.TM_LAG con el tiempo de retardo, p. ej. PID_Instance.TM_LAG
   - TUN_ES.MTR_TM con el tiempo de posicionamiento del motor, p. ej. LMNGEN_S_Instance.MTR_TM
   - Si se utiliza una zona muerta, TUN_ES.DEADB_W con el ancho de la zona muerta, p. ej. DEADBAND_Instance.OUTV
   - TUN_ES.QLMNUP y TUN_ES.QLMNDN con las señales de la variable manipulada LMNGEN_S_Instance.LMNUP y LMNGEN_S_Instance.LMNDN
4. Haga clic en el símbolo en la instrucción TUN_ES.
   Se abre el editor de puesta en servicio para TUN_ES. El funcionamiento del editor de puesta en servicio se describe en el PID Self-Tuner.

11.3.5.3 Utilizar PID Self Tuner

Optimizar el regulador continuo y el regulador de impulsos

Interconectar TUN_EC con el regulador

La instrucción TUN_EC permite crear un regulador continuo o un regulador de impulsos. Según el regulador, TUN_EC se activa con una de las siguientes instrucciones:

Regulador	Instrucción
Reguladores de impulsos FM 355 C, FM 355 S	PID_FM
Reguladores de impulsos FM 455 C, FM 455 S	PID_FM
Funciones básicas PID	CONT_C Con o sin PULSEGEN

Regulador	Instrucción
Standard PID Control (paquete opcional PID Professional)	PID_CP
Modular PID Control (paquete opcional PID Professional)	PID LMNGEN_C

Interconecte los parámetros de entrada y salida de la instrucción TUN_EC con la instrucción del regulador como se describe a continuación.

TUN_EC	PID_FM	CONT_C	PID_CP
PV	PV	PV	PV
LMN	LMN	LMN	LMN
NORM_FAC	--	--	(NM_PVHR-NM_PVLR)*0.01
WRITE_DIS	Operación lógica O con LOAD_OP y LOAD_PAR	--	--
MAN_OUT	LMN_RE	MAN	MAN
SP_OUT	SP_RE	SP_INT	SP_INT
GAIN	GAIN	GAIN	GAIN
TI	TI	TI	TI
TD	TD	TD	TD
TM_LAG	TM_LAG	TM_LAG	TM_LAG
RATIOFAC	--	--	RATIOFAC
PHASE	--	--	PHASE
QMAN_ON	LMN_REON	MAN_ON	MAN_ON
QI_SEL	--	I_SEL	I_SEL
QD_SEL	--	D_SEL	D_SEL
QWRITE	LOAD_PAR y LOAD_OP	--	--

Consulte también

Funcionamiento de TUN_EC (Página 7133)

TUN_EC (Página 7129)

Posibilidades de optimización

Optimización inicial

La optimización de los parámetros PID se realiza durante el calentamiento. Para ello se lleva a cabo una identificación del proceso seguida de un diseño del regulador.

La optimización puede solicitarse con ADAPT1ST.

Si se ajusta ADAPT1ST = TRUE, se ajustará ADAPT_ON = TRUE y la optimización inicial se llevará a cabo obligatoriamente. En la optimización inicial, el escalón de la variable manipulada se indica en el parámetro LHLM_TUN.

El parámetro ADAPT1ST solo es necesario si, debido a errores graves durante la adaptación, se han obtenido parámetros de proceso tan absurdos que el escalón de la variable manipulada calculado automáticamente no sirve para una nueva adaptación.

Con ADAPT1ST = TRUE se ejecuta la rutina de inicialización de la instrucción TUN_EC. Se llevan a cabo los siguientes ajustes predeterminados:

- Se conmuta al regulador PI. Para ello se transfieren al regulador los parámetros de PI_CON. Si los parámetros PI_CON.GAIN y PI_CON.TI tienen ambos el valor 0.0, se les asignarán los valores predeterminados PI_CON.GAIN = 1.5 y PI_CON.TI = 3600.0 s.
- PROCESS.GAIN = 999.0
- RATIOFAC = 1.0
- STATUS_H = 0; STATUS_D = 0; STATUS_C = 0

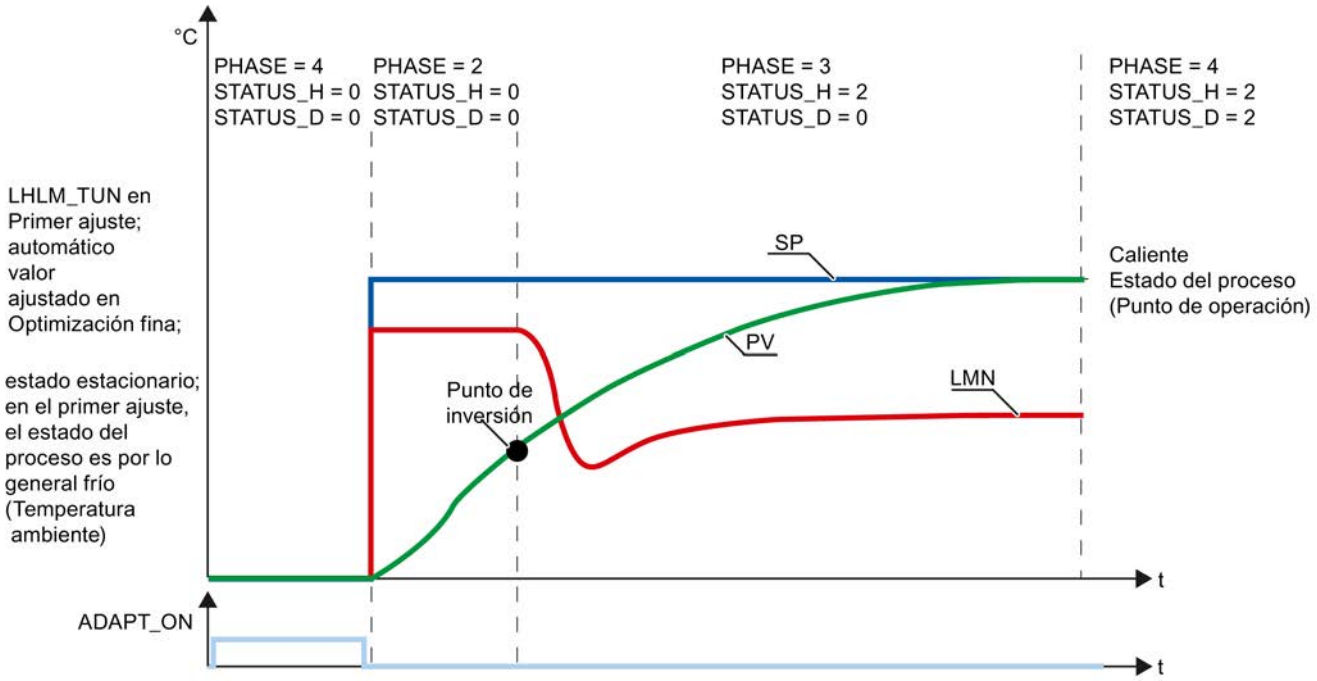
Nota

A menos que desee hacerlo expresamente, en caso de rearranque completo (en caliente) de la CPU **no** llame TUN_EC como es habitual con otras instrucciones, pues de lo contrario se perderán todos los parámetros de proceso.

La optimización se inicia con ADAPT1ST = TRUE, seguido de un escalón de la variable manipulada ≥ MIN_STEP en sentido positivo.

El escalón de la variable manipulada en la optimización inicial suele saltar desde la consigna del proceso en frío (temperatura ambiente) hasta la proximidad del punto de operación. También puede usarse como estado inicial cualquier otra consigna, a condición de que el proceso se encuentre en un estado estacionario. Durante la optimización no pueden especificarse más escalones de la variable manipulada.

Evolución de curva en la optimización inicial



Optimización fina

La optimización fina solo puede realizarse después de una optimización inicial concluida correctamente. Con ADAPT_ON = TRUE, TUN_EC decide si se llevará a cabo una optimización inicial o una optimización fina. Si ya se dispone de información acerca del proceso, TUN_EC calcula automáticamente un escalón de la variable manipulada y lleva a cabo una optimización fina. Si no existe información acerca del proceso, se realiza una optimización inicial.

Iniciar optimización

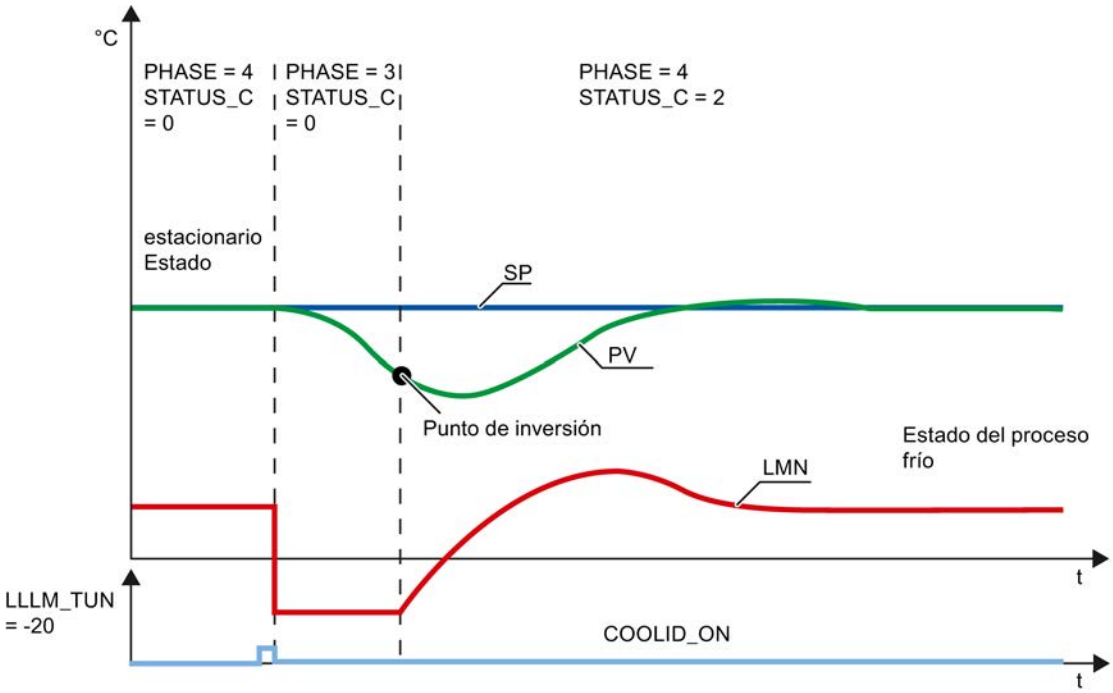
La optimización puede iniciarse tanto desde el modo automático como desde el modo manual.

- **Iniciar la optimización desde el modo automático**
  En el modo automático, active ADAPT_ON o ajuste ADAPT1ST = TRUE y especifique un salto de consigna ≥ MIN_STEP.
- **Iniciar la optimización desde el modo manual**
  En modo manual, active ADAPT_ON o ajuste ADAPT1ST = TRUE y especifique un escalón de consigna ≥ MIN_STEP. Cuando, a continuación, se resetee MAN_ON = FALSE , el escalón de consigna se hará efectivo y la optimización se iniciará.

Optimización de refrigeración

La optimización de refrigeración solo es posible si la optimización inicial ha concluido correctamente. Se inicia en la PHASE 4. Al ajustar COOLID_ON = TRUE, el regulador pasa a modo manual y se emite LLLM_TUN: MAN_OUT = LLLM_TUN. TUN_EC pasa a PHASE = 3, y se vuelve a desactivar inmediatamente COOLID_ON. Tan pronto como se encuentra un punto de inflexión en la curva del valor real, TUN_EC vuelve al modo de regulación (PHASE = 4). TUN_EC calcula un factor de relación RATIOFAC, cuyo valor se transmite al generador de impulsos del regulador.

Evolución de curva en la optimización de refrigeración



Nota

- El bit de inicio COOLID_ON puede activarse al mismo tiempo con ADAPT1ST o ADAPT_ON, así como durante una optimización de los parámetros del regulador. De este modo, una vez finalizada la optimización de los parámetros del regulador para el calentamiento, se inicia de inmediato la optimización de refrigeración.
- En una instalación con varias zonas de temperatura acopladas térmicamente (p. ej. máquinas de transformación de plásticos), la optimización de refrigeración no debe iniciarse hasta que todas (!) las zonas no hayan finalizado la optimización inicial, ya que de lo contrario se falsearían los resultados de la optimización inicial.

Cancelar optimización

Para interrumpir la optimización antes del escalón de consigna, es necesario ajustar ADAPT_ON = FALSE . Para interrumpir la optimización después del salto de consigna (PHASE 2 o 3) o interrumpir la optimización de refrigeración, debe pasarse a modo manual (MAN_ON = TRUE).

Al ajustar ADAPT_ON, ADAPT1ST o COOLID_ON durante una optimización en curso (PHASE 2 o 3) la optimización no se interrumpe, sino que se lleva a cabo hasta el final. Dado que no se desactivan ADAPT_ON, ADAPT1ST ni COOLID_ON, en cuanto se cumplen las condiciones de inicio se inicia una nueva optimización.

Fases de la optimización

La optimización del regulador se lleva a cabo en varias fases, que se indican en el parámetro PHASE .

PHASE = 0

Al crear un DB de instancia para TUN_EC, el parámetro PHASE tiene preasignado el valor cero y MAN_ON el valor TRUE. Por eso, después del primer ciclo se cambia al modo manual (PHASE = 7). Desde el modo manual se puede iniciar directamente la optimización inicial (ADAPT_ON = TRUE) o cambiar al modo automático. En el primer cambio al modo automático MAN_ON = FALSE, se adoptan los parámetros de PI_CON. Los valores predeterminados para GAIN y TI son: GAIN = 1.5, TI = 1h.

PHASE = 2

En cuanto se especifica un escalón de consigna ≥ MIN_STEP en sentido positivo hacia el punto de operación del estado caliente del proceso, en la optimización inicial se preasigna a MAN_OUT el valor LHLM_TUN, y en la optimización fina se preasigna a MAN_OUT un valor calculado por TUN_EC. Además se activa QMAN_ON = TRUE. Ambos valores se transmiten al regulador PID. El regulador PID se controla entonces en modo manual. MIN_STEP debería ser superior al 10 % del rango de trabajo de consigna y real.

PHASE = 3

Si se detecta el punto de inversión de la respuesta al salto (STATUS_H = 2) o si el valor real ha alcanzado el 70 % de la intensidad del salto de la consigna (STATUS_H = 3), se diseña un regulador PI de ajuste cuidado. La ganancia del regulador GAIN se limita al rango entre 0.2 y 20.0. El regulador funciona a partir de ese momento como regulador PI e intenta regular el proceso a un estado estacionario. TUN_EC guarda los nuevos parámetros de regulador PI en GAIN, TI y PI_CON. Los valores anteriores del regulador PI se guardan en PI_CON_OLD. Si el tiempo necesario para alcanzar el estado estacionario es muy prolongado (respuesta transitoria muy lenta en lazos de regulación de temperatura), ya puede iniciarse el diseño del regulador con los datos actuales al aproximarse al estado estacionario activando STEADY = TRUE. El diseño del regulador también se puede iniciar de nuevo con los datos actuales en un momento posterior, en PHASE = 3 o PHASE = 4, mediante la activación de STEADY = TRUE. Con frecuencia, así se mejora un poco más el diseño del regulador.

En la optimización inicial, los rebases transitorios o la imposibilidad de encontrar el punto de inflexión pueden deberse a que se ha seleccionado una altura demasiado grande del escalón de la variable manipulada LHLM_TUN, y no deben atribuirse necesariamente a un mal ajuste del regulador. En la siguiente optimización inicial, asigne a LHLM_TUN un valor aprox. un 20% menor.

Si TUN_EC ha detectado el estado estacionario o ha transcurrido desde el escalón de la variable manipulada el tiempo 30 × TI (TI: tiempo de integración del regulador PI ajustado en PHASE = 3), se realiza un diseño mejorado del regulador, y se cambia a PHASE = 4. TUN_ES guarda los nuevos parámetros de regulador PI/PID en GAIN, TI, TD y PI_CON o PID_CON. Los valores anteriores del regulador PID se guardan en PID_CON_OLD. A ser posible se diseñan un regulador PID y un regulador PI. Para PID_ON = TRUE, se activa el regulador PID, o en su caso el regulador PI. GAIN se limita de manera que la ganancia del lazo abierto (producto de GAIN por la ganancia del proceso) sea inferior a 80.0.

PHASE = 4

En esta fase el regulador regula con sus parámetros optimizados. Desde este estado se puede iniciar una optimización de refrigeración, una optimización fina o una optimización inicial.

Es posible alternar en cualquier momento sin discontinuidad entre el regulador PI y el regulador PID.

Consulte también

Funcionamiento de TUN_EC (Página 7133)

Realizar una optimización inicial

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- TUN_EC está interconectado con un regulador continuo o un regulador de impulsos.
- MAN_ON = FALSE
- El regulador está en modo automático.
- El proceso está en estado estacionario.

Procedimiento

Para determinar los parámetros PID óptimos en la primera puesta en servicio, haga lo siguiente:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización inicial".
   TUN_EC está disponible para la optimización.
3. Seleccione una estructura del regulador.
4. Introduzca en el campo "Variable manipulada" una variable manipulada, que se utilizará hasta que se pase a la PHASE = 3.
5. En el campo "Consigna", indique una consigna. El escalón de consigna debe ser ≥ MIN_STEP. La variable manipulada no será efectiva hasta que no exista una nueva consigna.
6. Haga clic en el símbolo "Iniciar optimización".
   La consigna y la variable manipulada se transferirán al regulador. Se iniciará la optimización inicial. Se mostrará el estado de la optimización.

Consulte también

Funcionamiento de TUN_EC (Página 7133)

Realizar optimización fina

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- TUN_EC está interconectado con un regulador continuo o un regulador de impulsos.
- MAN_ON = FALSE
- El regulador está en modo automático.
- El proceso está en estado estacionario.

Procedimiento

Para determinar los parámetros PID óptimos en el punto de operación, haga lo siguiente:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización fina".
   TUN_EC está disponible para la optimización.
3. Seleccione una estructura del regulador.
4. En el campo "Consigna", indique una consigna. El escalón de consigna debe ser ≥ MIN_STEP.
5. Haga clic en el símbolo "Iniciar optimización".
   Se enviará la nueva consigna al regulador. TUN-EC calculará un escalón de la variable manipulada. Se iniciará la optimización fina. Se mostrará el estado de la optimización.

Consulte también

Funcionamiento de TUN_EC (Página 7133)

Efectuar una optimización de la refrigeración

Durante la optimización de refrigeración se calcula el factor de relación para reguladores con dos actuadores de acción contrapuesta (p. ej. calentar/enfriar).

Requisitos

- La optimización inicial se ha llevado a cabo correctamente.
- La instrucción y el objeto tecnológico están cargados en la CPU.
- TUN_EC está interconectado con un regulador continuo o un regulador de impulsos.
- MAN_ON = FALSE
- El regulador está en modo automático.
- El proceso está en estado estacionario.

Procedimiento

Para obtener el factor de relación, haga lo siguiente:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización de la refrigeración".
   TUN_EC está disponible para la optimización.
3. Seleccione una estructura del regulador.
4. En el campo "Variable manipulada", introduzca una variable manipulada, que se utilizará hasta que se encuentre el punto de inflexión.
5. Haga clic en el símbolo "Iniciar optimización".
   Se inicia la optimización de refrigeración. Se mostrará el estado de la optimización.

Realizar una optimización manual

Si, después de una optimización inicial, se producen oscilaciones en el lazo de regulación cerrado, o existen rebases transitorios después de escalones de consigna, pueden modificarse manualmente los parámetros PID. P. ej., puede restablecer la ganancia del regulador GAIN al 80% del valor inicial, o aumentar el tiempo de integración TI hasta el 150% del valor ajustado inicialmente. Si se usa un regulador de impulsos, pueden producirse pequeñas oscilaciones permanentes que se eliminan aumentando la zona muerta DEADB_W.

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo", seleccione la entrada "Manual".
3. Seleccione una estructura del regulador.
4. Introduzca parámetros PID nuevos en los campos "P", "I" y "D".
5. En el grupo "Optimización", haga clic en el símbolo "Enviar parámetros al regulador".
6. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna".
7. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
8. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabajará con los nuevos parámetros PID y regulará de acuerdo con la nueva consigna.

9. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
10. Repita los pasos del 3 al 9 hasta obtener una regulación satisfactoria.

Consulte también

Funcionamiento de TUN_EC (Página 7133)

Optimizar el regulador paso a paso

Interconectar TUN_ES con el regulador

La instrucción TUN_ES permite optimizar un regulador paso a paso. Según el regulador, TUN_ES se activa con una de las siguientes instrucciones:

Regulador	Instrucción
Regulador paso a paso FM 355 S	PID_FM
Regulador paso a paso FM 455 S	PID_FM
Funciones básicas PID	CONT_S TCONT_S
Standard PID Control (paquete opcional)	PID_ES
Modular PID Control (paquete opcional)	PID LMNGEN_S

Interconecte los parámetros de entrada y salida de la instrucción TUN_ES con la instrucción del regulador del modo que se describe a continuación.

TUN_ES	PID_FM	CONT_S	TCONT_S	PID_ES
PV	PV	PV	PV	PV
LMNR	LMN_A	--	--	LMNR_IN
NORM_FAC	--	--	--	(NM_PVHR-NM_PVLR)*0.01
C_LMNUP	QLMNUP	QLMNUP	QLMNUP	QLMNUP
C_LMNDN	QLMNDN	QLMNDN	QLMNDN	QLMNDN
LMNR_HS	QLMNR_HS	LMNR_HS	LMNR_HS	LMNR_HS
LMNR_ON	QLMNR_ON	--	--	LMNR_ON
WRITE_DIS	Operación lógica O con LOAD_OP y LOAD_PAR	--	--	--
MAN_OUT	LMN_RE	--	--	MAN
SP_OUT	SP_RE	SP_INT	SP_INT	SP_INT o SP_EXT
GAIN	GAIN	GAIN	GAIN	GAIN
TI	TI	TI	TI	TI
TD	TD	--	--	TD
TM_LAG	TM_LAG	--	--	TM_LAG

TUN_ES	PID_FM	CONT_S	TCONT_S	PID_ES
MTR_TM	MTR_TM	MTR_TM	MTR_TM	MTR_TM
DEADB_W	DEADB_W	DEADB_W	DEADB_W	DEADB_W
PHASE	--	--	--	PHASE
QMAN_ON	LMN_REON	--	--	MAN_ON
QLMNS_ON	LMNSOPON	LMNS_ON	LMNS_ON	LMNS_ON
QLMNUP	LMNUP_OP	LMNUP	LMNUP	LMNUP
QLMNDN	LMNDN_OP	LMNDN	LMNDN	LMNDN
QI_SEL	--	--	--	I_SEL
QD_SEL	--	--	--	D_SEL
QWRITE	LOAD_PAR y LOAD_OP	--	--	--

Consulte también

Funcionamiento TUN_ES (Página 7152)

TUN_ES (Página 7149)

Posibilidades de la optimización

Optimización inicial

La optimización de los parámetros PID se realiza durante el calentamiento. Para ello se lleva a cabo una identificación del proceso seguida de un diseño del regulador.

La optimización inicial puede solicitarse con ADAPT1ST.

Si se ajusta ADAPT1ST = TRUE, entonces ADAPT_ON = TRUE y se cierra la válvula (PHASE = 1).

Con ADAPT1ST = TRUE se ejecuta la rutina de inicialización de la instrucción TUN_ES. Se llevan a cabo los siguientes ajustes predeterminados:

- Se conmuta al regulador PI. Para ello se transfieren al regulador los parámetros de PI_CON. Si los parámetros PI_CON.GAIN y PI_CON.TI tienen ambos el valor 0.0, se les asignarán los valores predeterminados PI_CON.GAIN = 1.5 y PI_CON.TI = 3600.0 s.
- PROCESS.GAIN = 999.0
- STATUS_H = 0; STATUS_D = 0

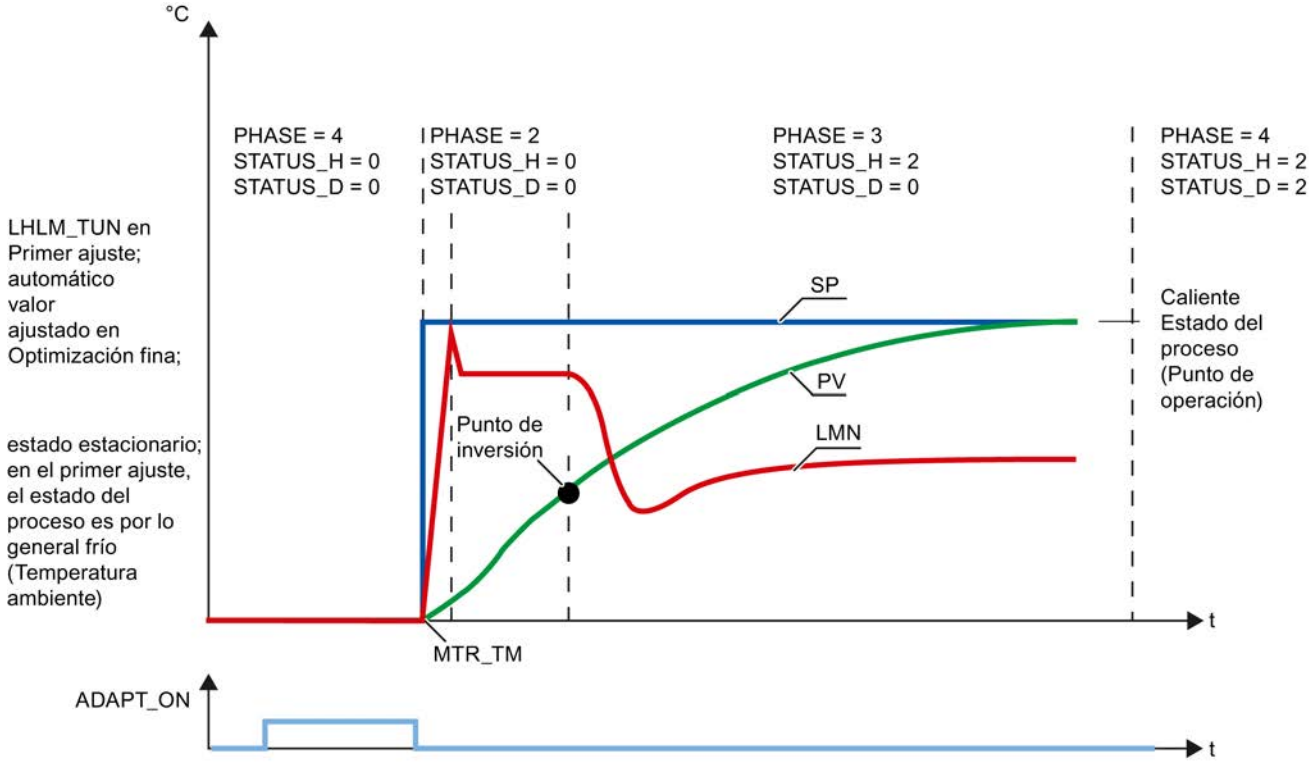
Nota

A menos que desee hacerlo expresamente, en caso de rearranque completo (en caliente) de la CPU **no** llame TUN_ES como es habitual con otras instrucciones, pues de lo contrario se perderán todos los parámetros de proceso.

Después de un escalón de consigna ≥ MIN_STEP se mide el tiempo de posicionamiento del motor y a continuación se lleva a cabo una optimización inicial. En la optimización inicial, TUN_ES utiliza para la excitación del proceso el parámetro LHLM_TUN, que determina el escalón de la variable manipulada. En los reguladores paso a paso con realimentación de

posición, se avanza directamente hasta el valor LHLM_TUN. En los reguladores paso a paso sin realimentación de posición, la servoválvula se abre por completo y se vuelve a cerrar de nuevo hasta que transcurre la proporción especificada (100 - LHLM_TUN) del tiempo de posicionamiento del motor.

Optimización inicial con medición del tiempo de posicionamiento del motor para reguladores paso a paso sin realimentación de posición



Optimización fina

La optimización solo se puede iniciar con ADAPT_ON en los reguladores paso a paso con realimentación de posición. A partir de los valores de proceso anteriores, TUN_ES decide si se realizará una optimización inicial sin medición del tiempo de posicionamiento del motor o una optimización fina. Con ADAPT_ON = TRUE seguido de un escalón de consigna ≥ MIN_STEP en sentido positivo, se inicia la optimización. En ese momento, TUN_ES pasa inmediatamente a la PHASE = 2 y realiza un escalón de consigna a LHLM_TUN en el caso de la optimización inicial, o a una variable manipulada calculada automáticamente, en el caso de la optimización fina.

El escalón de la variable manipulada en la optimización inicial suele saltar desde la consigna del proceso en frío (temperatura ambiente) hasta la proximidad del punto de operación. También puede usarse como estado inicial cualquier otra consigna, a condición de que se trate de un estado estacionario. Durante la optimización no pueden especificarse más escalones de la variable manipulada.

Iniciar optimización

La optimización puede iniciarse tanto desde el modo automático como desde el modo manual.

- **Iniciar la optimización desde el modo automático**
  En el modo automático, active ADAPT_ON o ajuste ADAPT1ST = TRUE y especifique un salto de consigna ≥ MIN_STEP.
- **Iniciar la optimización desde el modo manual**
  En modo manual, active ADAPT_ON o ajuste ADAPT1ST = TRUE y especifique un escalón de consigna ≥ MIN_STEP. Cuando, a continuación, se resetee MAN_ON = FALSE , el escalón de consigna se hará efectivo y la optimización se iniciará.

Cancelar optimización

Para interrumpir la optimización antes del escalón de consigna, es necesario ajustar ADAPT_ON = FALSE . Para interrumpir la optimización después del escalón de consigna (PHASE 2 o 3) o interrumpir la optimización de refrigeración, debe pasarse a modo manual (MAN_ON = TRUE = LMNS_ON = TRUE).

Al ajustar ADAPT_ON o ADAPT1ST durante una optimización (PHASE 2 o 3) la optimización no se interrumpe, sino que se lleva a cabo hasta el final. Dado que no se desactivan ADAPT_ON ni ADAPT1ST, en cuanto se cumplen las condiciones de inicio se inicia una nueva optimización.

Fases de la optimización

La optimización del regulador se lleva a cabo en varias fases, que se indican en el parámetro PHASE .

PHASE = 0

Al crear un DB de instancia para TUN_ES, el parámetro PHASE tiene preasignado el valor cero y LMNS_ON el valor TRUE. Por eso, después del primer ciclo se cambia al modo manual (PHASE = 7). Desde el modo manual se puede iniciar directamente la optimización inicial (ADAPT_ON = TRUE) o cambiar al modo automático. En el primer cambio al modo automático MAN_ON = FALSE, se adoptan los parámetros de PI_CON. Los valores predeterminados para GAIN y TI son: GAIN = 1.5, TI = 1h.

PHASE = 1

Una vez que la optimización ha comenzado con ADAPT1ST = TRUE, TUN_ES cierra la servoválvula (variable manipulada = 0 o LMNS_ON = LMNDN = TRUE) y espera hasta un escalón positivo de la consigna ≥ MIN_STEP.

PHASE = 2

En el momento en que se especifica un escalón de consigna ≥ MIN_STEP en sentido positivo hacia el punto de operación del estado caliente del proceso, en la optimización inicial se asigna a MAN_OUT el valor LHLM_TUN, y en la optimización fina se asigna a MAN_OUT un valor calculado por TUN_ES. MIN_STEP debe ser mayor que el 10% del rango de trabajo entre consigna y valor real.

En la regulación paso a paso con realimentación de posición (LMNR_ON=TRUE), se asigna a QMAN_ON el valor TRUE. El regulador regula la válvula al valor de MAN_OUT y TUN_ES calcula el tiempo de posicionamiento del motor MTR_TM.

En la regulación paso a paso sin realimentación de posición, QLMNS_ON y QLMNUP se ajustan a TRUE y la válvula se mueve hasta el tope superior. Al alcanzar el tope superior (LMNR_HS =TRUE), TUN_ES calcula el tiempo de posicionamiento del motor y lo transmite al regulador. A continuación la válvula se cierra hasta el valor de MAN_OUT (QLMNDN = TRUE).

PHASE = 3

Si se detecta el punto de inversión de la respuesta al salto (STATUS_H = 2) o si el valor real ha alcanzado el 70 % de la intensidad del salto de la consigna (STATUS_H = 3), se diseña un regulador PI de ajuste cuidado. La ganancia del regulador GAIN se limita al rango entre 0.2 y 20.0. El regulador funciona a partir de ese momento como regulador PI e intenta regular el proceso a un estado estacionario. TUN_ES guarda los nuevos parámetros de regulador PI en GAIN, TI y PI_CON. Los valores anteriores del regulador PI se guardan en PI_CON_OLD. Si el tiempo necesario para alcanzar el estado estacionario es muy prolongado (respuesta transitoria muy lenta en lazos de regulación de temperatura), ya puede iniciarse el diseño del regulador con los datos actuales al aproximarse al estado estacionario activando STEADY = TRUE. El diseño del regulador también se puede iniciar de nuevo con los datos actuales en un momento posterior, en PHASE = 3 o PHASE = 4, mediante la activación de STEADY = TRUE. Con frecuencia, así se mejora un poco más el diseño del regulador.

En la optimización inicial, los rebases transitorios o la imposibilidad de encontrar el punto de inflexión pueden deberse a que se ha seleccionado una altura demasiado grande del escalón de la variable manipulada LHLM_TUN, y no deben atribuirse necesariamente a un mal ajuste del regulador. En la siguiente optimización inicial, asigne a LHLM_TUN un valor aprox. un 20% menor.

Si TUN_ES ha detectado el estado estacionario o ha transcurrido desde el escalón de la variable manipulada el tiempo 30 × TI (TI: tiempo de integración del regulador PI ajustado en PHASE = 3), se realiza un diseño mejorado del regulador, y se cambia a PHASE = 4. TUN_ES guarda los nuevos parámetros de regulador PI/PID en GAIN, TI, TD y PI_CON o PID_CON. Los valores anteriores del regulador PID se guardan en PID_CON_OLD. A ser posible se diseñan un regulador PID y un regulador PI. Si PID_ON = TRUE, se activa el regulador PID, de lo contrario el regulador PI. En procesos complejos, TUN_ES diseña siempre un regulador PI independientemente de PID_ON. GAINse limita de tal manera que la ganancia del lazo abierto (producto de GAIN por la ganancia del proceso) sea inferior a 40.0.

PHASE = 4

En esta fase el regulador regula con sus parámetros optimizados. Desde este estado se puede iniciar una optimización fina o una optimización inicial.

Es posible alternar en cualquier momento sin discontinuidad entre el regulador PI y el regulador PID.

Consulte también

Campo de aplicación de TUN_ES (Página 7150)

Funcionamiento TUN_ES (Página 7152)

Realizar una optimización inicial

La optimización inicial de los parámetros del regulador se realiza durante el calentamiento. Para ello se lleva a cabo una identificación del proceso seguida de un diseño del regulador.

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- TUN_ES está interconectado con un regulador paso a paso.
- MAN_ON = FALSE
- El regulador está en modo automático.
- El proceso está en estado estacionario.

Procedimiento

Para determinar los parámetros PID óptimos en la primera puesta en servicio, haga lo siguiente:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización inicial".
   TUN_ES está disponible para la optimización.
3. Seleccione una estructura del regulador.
4. Introduzca en el campo "Variable manipulada" una variable manipulada, que se utilizará hasta que se pase a la PHASE = 3.
5. Introduzca una nueva consigna. El escalón de consigna debe ser ≥ MIN_STEP.
6. Haga clic en el símbolo "Iniciar optimización".
   Se iniciará la optimización inicial. Se mostrará el estado de la optimización.

Consulte también

Funcionamiento TUN_ES (Página 7152)

Realizar optimización fina

En los reguladores paso a paso con realimentación de posición puede iniciarse una optimización fina. Si cambia el punto de operación o el comportamiento del proceso, el regulador puede reoptimizarse online durante un salto positivo de la consigna.

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- TUN_ES está interconectado con un regulador paso a paso.
- MAN_ON = FALSE
- LMNR_ON = TRUE en la instrucción TUN_ES y en el regulador.
- El regulador está en modo automático.
- El proceso está en estado estacionario.

Procedimiento

Para determinar los parámetros PID óptimos en la primera puesta en servicio, haga lo siguiente:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo" seleccione la entrada "Optimización fina".
   TUN_ES está disponible para la optimización.
3. Seleccione una estructura del regulador.
4. Introduzca una nueva consigna. El escalón de consigna debe ser ≥ MIN_STEP.
5. Haga clic en el símbolo "Iniciar optimización".
   Se iniciará la optimización fina. TUN_ES determinará la variable manipulada. Se mostrará el estado de la optimización.

Realizar una optimización manual

Si, después de una optimización inicial, se producen oscilaciones en el lazo de regulación cerrado, o existen rebases transitorios después de escalones de consigna, pueden modificarse manualmente los parámetros PID. P. ej., puede restablecer la ganancia del regulador GAIN al 80% del valor inicial, o aumentar el tiempo de integración TI hasta el 150% del valor ajustado inicialmente.

Requisitos

- La instrucción y el objeto tecnológico están cargados en la CPU.
- TUN_ES está interconectado con un regulador paso a paso.
- MAN_ON = FALSE
- El regulador está en modo automático.
- El proceso está en estado estacionario.

Procedimiento

Para determinar manualmente los parámetros PID óptimos, proceda del siguiente modo:

1. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
2. En la lista "Modo", seleccione la entrada "Manual".
3. Seleccione una estructura del regulador.
4. Introduzca parámetros PID nuevos en los campos "P", "I" y "D".
5. En el grupo "Optimización", haga clic en el símbolo "Enviar parámetros al regulador".
6. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna".
7. Introduzca una nueva consigna y haga clic en el grupo "Valores actuales", en el símbolo .
8. En caso necesario, desactive la casilla de verificación "Modo manual".
   El regulador trabajará con los nuevos parámetros PID y regulará de acuerdo con la nueva consigna.
9. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
10. Repita los pasos del 3 al 9 hasta obtener una regulación satisfactoria.

Consulte también

Funcionamiento TUN_ES (Página 7152)

11.3.5.4 Fundamentos de los módulos de regulación

Entradas

Introducción

En las entradas analógicas se pueden conectar diferentes tipos de sensores. Las señales de entrada de los sensores se procesan a continuación según las necesidades.

Las entradas digitales permiten poner el módulo en diferentes modos de operación.

Los reguladores K y S tienen la misma estructura en cuanto a entradas analógicas y digitales.

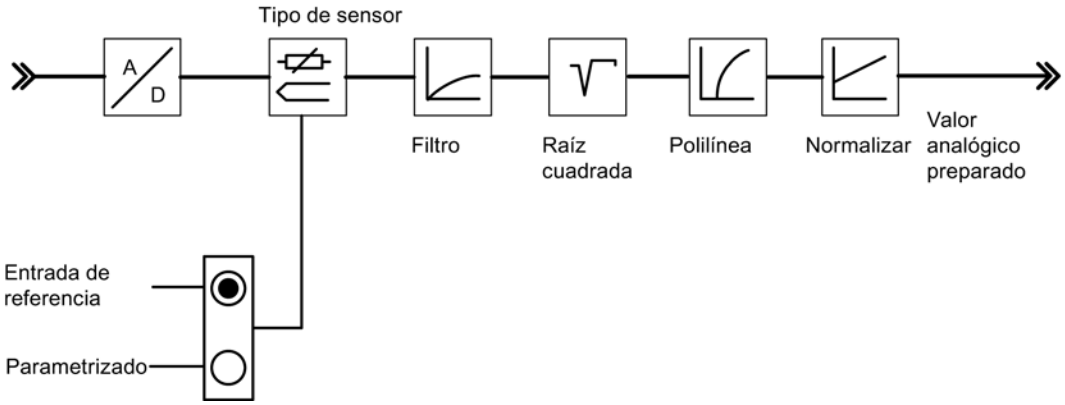
Número de entradas

Entrada	FM 355	FM 355-2	FM 455
Entradas analógicas con acondicionamiento de los valores analógicos	4	4	16
Entrada de unión fría para la compensación de termopares	1	1	1
Entradas digitales	8	8	16

Entradas analógicas

Bloques de función de una entrada analógica

La figura siguiente muestra el esquema para el acondicionamiento de los valores analógicos:



Adaptación a sensores

Las entradas analógicas se pueden adaptar a diferentes sensores mediante su parametrización. Son posibles los siguientes ajustes:

- La entrada analógica no se procesa (p. ej. entrada no utilizada)
  Con este ajuste, todos los parámetros PID se ajustan a cero, el canal de regulación no se procesa y no se calcula ninguna variable manipulada.
- Sensor de intensidad 0 a 20 mA
- Sensor de intensidad 4 a 20 mA
- Sensor de tensión 0 a 10 V
- Pt 100, –200 ... 850 °C
- Pt 100, –200 ... 556 °C (resolución doble)
- Pt 100, –200 ... 130 °C (resolución cuádruple)

- Termopares tipo B, J, K, R y S (entrada analógica ajustada a ±80 mV)
- Termopar libre (entrada analógica ajustada a ±80 mV)

Las entradas analógicas se parametrizan en la pantalla "Entrada analógica".

Cambio Celsius/Fahrenheit

Las temperaturas se pueden medir tanto en °C como en °F.

Adaptación a la frecuencia de red

Para suprimir interferencias durante la medición de señales analógicas se puede ajustar el procesamiento de la señal de entrada a la frecuencia de red. Son posibles los siguientes ajustes:

- Funcionamiento a 50 Hz
- Funcionamiento a 60 Hz

Unión fría

Si se ha ajustado un termopar como sensor en una entrada analógica, se puede conectar en la entrada de unión fría del módulo un Pt 100 para compensar la temperatura de referencia en termopares. De forma alternativa, se puede parametrizar una temperatura de unión frío fija.

En el FM 355 se puede parametrizar opcionalmente una compensación interna para los termopares J, K y E. Un sensor de temperatura interno mide la temperatura de la unión fría directamente en el módulo.

Si se utiliza la entrada de unión fría, el tiempo de muestreo de cada regulador se prolonga en el tiempo de conversión de la entrada de unión fría.

Acondicionamiento de valores analógicos

El acondicionamiento de los valores analógicos ofrece diferentes posibilidades parametrizables para acondicionar las señales de entrada. La siguiente tabla ofrece una visión de conjunto de estos parámetros y los valores ajustables.

Parámetro	Valores ajustables	Observaciones
Resolución	• 12 bits • 14 bits	Tiempo de conversión 20 ms (50 Hz) Tiempo de conversión $16^2/_3$ ms (60 Hz) Tiempo de conversión 100 ms
Filtro	• On/Off • Constante de tiempo en s	Filtro de 1er orden cuya respuesta temporal está determinada por la constante de tiempo
Raíz cuadrada	• On/Off	Para linealizar señales de encóder en las que el valor real existe como magnitud física en relación cuadrática con la magnitud medida del proceso

Parámetro	Valores ajustables	Observaciones
Polilínea	• On/Off • 13 puntos de apoyo seleccionables en – mA con entrada de intensidad – mV con entrada de tensión	Para linealizar curvas características de encóder
Normalización	• Valor superior • Valor inferior	Para la conversión de la señal de entrada en otra unidad física mediante interpolación lineal entre el valor inicial (abajo) y el valor final (arriba)

Nota

Normalización/polilínea: La conversión de la unidad mA o mV en una unidad física se realiza mediante polilínea o (si esta no está activada) mediante normalización. La polilínea se utiliza para linealizar un termopar libre o para cualquier linealización.

Copiar parámetros

Los FMs ofrecen en la entrada analógica la posibilidad de transferir parámetros de una entrada a otra o varias entradas. Todos los parámetros ajustados del canal de salida se transfieren a los canales de destino.

Entradas digitales

Modos de operación

Las entradas digitales sirven para cambiar los modos de operación de canales de regulación individuales.

El sentido de acción de las entradas digitales es parametrizable. Para cada entrada digital individual son posibles los siguientes ajustes:

- High activo
- Low activo o abierto

Se pueden seleccionar los siguientes modos de operación:

- Conmutación a la variable manipulada especificada con el FB PID_FM
- Cambio a modo de seguimiento (variable manipulada especificada mediante una entrada analógica)
- Conmutación a variable manipulada de seguridad

Con el regulador paso a paso se pueden definir además las siguientes señales a través de entradas digitales:

- Respuesta: Actuador en tope superior
- Respuesta: Actuador en tope inferior

Optimización paralela

La función "Optimización paralela" sólo está disponible para el FM 355-2.

Zonas vecinas (fuete acoplamiento térmico)

Si dos o más reguladores de un módulo regulan la temperatura p. ej. de una placa (p. ej. dos calefacciones y dos valores reales medidos con fuerte acoplamiento de calor), existe la posibilidad de realizar una optimización paralela para estos reguladores. Proceda del siguiente modo:

1. Defina los canales conexos en la zona A o la zona B.
2. Inicie la optimización en uno de los canales. La optimización de los canales conexos se inicia entonces automáticamente.

Nota

Sin embargo, en caso de excitación por escalón de consigna el usuario debe garantizar la activación simultánea de los escalones de consigna. Con el asistente, los parámetros PID_ON y TUN_DLMN/TUN_CLMN sólo se pueden especificar para un canal. Los parámetros para el resto de canales tienen que predefinirse con los correspondientes DBs de instancia del FB FMT_PID antes de iniciar la optimización.

Cancelación de la optimización

Si aparece un error en uno de los canales (fase 0 o final de la fase 1), se cancela la optimización de todos los canales.

El usuario puede restablecer la optimización poniendo TUN_ON=FALSE en alguno de los canales implicados.

Si un canal pasa al modo de seguridad, también se desactiva la optimización de los canales implicados.

En el asistente sólo se muestra la información de estado del canal seleccionado. Encontrará las causas de error de los canales restantes en los correspondientes DBs de instancia.

Ventaja

Los reguladores implicados emiten su correspondiente magnitud manipulada de optimización hasta que todos los reguladores hayan abandonado la fase 2. De esta forma se evita que el regulador que primero finalice la optimización falsifique el resultado de optimización del resto de reguladores al modificar su magnitud manipulada.

ATENCIÓN
El hecho de alcanzar el 75% del escalón de consigna en uno de los canales implicados no finaliza la optimización (peligro de sobreoscilación). El modo automático se inicia cuando todos los reguladores implicados han finalizado su optimización.

Zonas vecinas (débil acoplamiento térmico)

En general se aplica la norma de que habría que optimizar tal y como se vaya a regular después. Si en el modo de producción las zonas se recorren juntas en paralelo, de forma que las diferencias de temperatura entre las zonas se mantienen iguales, durante la optimización también debería aumentarse correspondientemente el nivel de temperatura de las zonas vecinas.

Las diferencias de temperatura al inicio del intento no son relevantes ya que se compensan con una calefacción inicial correspondiente (rampa inicial = 0).

Error de regulación

Copiar canales

En un regulador los FMs ofrecen la posibilidad de transferir un canal de regulación a uno o varios canales. Todos los ajustes del canal de salida se transfieren a los canales de destino.

Tipos de reguladores

En todos los módulos de regulación el error de regulación tiene la misma estructura.

1. La señal del valor real se acondiciona para obtener el valor real efectivo.
2. La señal de la consigna se acondiciona para obtener la consigna efectiva.
3. La consigna efectiva se resta del valor real efectivo.
4. El error de regulación restante se agrega al algoritmo de regulación.

Para la consigna y el valor real se pueden seleccionar los orígenes de regulación, para utilizar entonces el módulo de regulación como:

- Regulador de valor fijo y en cascada
- Regulador de tres componentes
- Regulador de relación y de mezcla

Regulador de valor fijo y en cascada

En el regulador de seguimiento de una regulación en cascada se selecciona como consigna la variable manipulada de un regulador piloto.

Regulador de tres componentes

El regulador de tres componentes dispone de tres entradas para el valor real. Los valores reales A, B y C se suman y dan como resultado el flujo total PV en una regulación de mezcla.

- Regulador de mezcla
  Un regulador de mezcla siempre es un regulador de seguimiento. El correspondiente regulador piloto es un regulador de tres componentes.
  La magnitud manipulada del regulador piloto se conecta a través de la entrada Valor real D. El factor de mezcla se define a través de la entrada de consigna del regulador.

Regulador de relación

Un regulador de relación siempre es un regulador de seguimiento. El correspondiente regulador piloto es un regulador de consigna fija.

Como valor real D se selecciona el valor real del regulador piloto. El factor de proporción se define a través de la entrada de consigna. Si se ha seleccionado una salida de regulador como factor de relación, la consigna se convierte (normaliza) al rango de valores "barrera inferior ... barrera superior" con ayuda de la barrera inferior y superior de "0... 100%".

Regulador de mezcla

Un regulador de mezcla siempre es un regulador de seguimiento. El correspondiente regulador piloto es un regulador de tres componentes.

Como valor real D se selecciona la magnitud manipulada del regulador piloto. El factor de mezcla se define a través de la entrada de consigna del regulador.

La magnitud manipulada del regulador del flujo total se indica en el rango de valores 0% a 100%. El regulador de seguimiento convierte esta magnitud del valor real D al rango de valores del valor real A. El rango de valores del valor real A son los valores de normalización "arriba" y "abajo" de la entrada analógica seleccionada.

Si la magnitud manipulada de un regulador de seguimiento llega al límite o si el aumento de la consigna de un regulador de seguimiento está limitado por la función rampa de la rama de consigna, la acción I del regulador piloto se bloquea en función del sentido (Anti-Reset-Windup) hasta que se haya solucionado la causa de la limitación en el regulador de seguimiento.

Selección de señal para consigna, valor real, entrada D y magnitud perturbadora

Selección de señal

Para la consigna, valor real, valor de la entrada D (entrada de acción D) y magnitud perturbadora de cada canal de regulación se puede seleccionar entre diferentes orígenes de

regulación. La siguiente tabla muestra una visión de conjunto sobre las posibilidades de selección de señal.

Tabla 11-6 Selección de señal para consigna, valor real, entrada D y magnitud perturbadora

Valores afectados	Origen de señal seleccionable
Consigna	• Un valor predefinido por el programa de usuario a través del bloque de función • El valor analógico acondicionado de una entrada analógica • La variable manipulada (LMN, LMN_A o LMN_B) de otro canal de regulación (en conexión en cascada de reguladores)
Valores reales A, B y C	• Cero • El valor analógico acondicionado de una entrada analógica (los valores reales B y C se pueden valorar adicionalmente por factores)
Valor real D	• Cero • El valor analógico acondicionado de una entrada analógica • Variable manipulada de otro canal de regulación
Valor para la entrada D (sólo relevante en reguladores PD o PID)	• El error de regulación tras la zona muerta del propio canal de regulación • El valor analógico acondicionado de una entrada analógica • El valor real efectivo negado del propio canal de regulación
Magnitud perturbadora	• Cero • El valor analógico acondicionado de una entrada analógica

Acondicionamiento de la consigna

El acondicionamiento de la consigna para obtener una consigna efectiva se puede variar con las siguientes funciones:

Origen de la consigna

La consigna puede venir dada por:

- Una instrucción
- Una entrada analógica
- Una variable manipulada acondicionada de otro canal de regulación del módulo
  Este origen de la consigna se selecciona en un regulador de mezcla o regulador de seguimiento en una regulación en cascada.
- Una variable manipulada A acondicionada de otro canal de regulación del módulo (sólo FM 355-2)
- Una variable manipulada B acondicionada de otro canal de regulación del módulo (sólo FM 355-2)

Consigna de seguridad

En caso de fallo de la CPU, el módulo puede regular en una consigna de seguridad.

Para ello debe introducirse una consigna de seguridad y ajustarse como reacción en caso de parada de la CPU "Consigna = Consigna de seguridad".

Respuesta a fallo de la CPU

En caso de fallo o parada de la CPU el módulo de regulación puede regular en las consignas siguientes:

- Consigna = última consigna válida
  Si como selección de consigna se ha ajustado "por instrucción", la consigna permanece constante en el último valor indicado tras el fallo de la CPU.
  Si la consigna viene determinada por la variable manipulada de un regulador del FM o por una entrada analógica, se modifica en función del valor seleccionado.
- Consigna = consigna de seguridad
  Si se ha indicado una consigna de seguridad, el módulo regula a esta consigna.

Respuesta a arranque del módulo (sólo FM 355 y FM 455)

El módulo de regulación puede regular también a las siguientes consignas en caso de arranque del módulo:

- Consigna = última consigna válida
- Consigna = consigna de seguridad

Rampa

Seleccionando un tiempo de arranque del valor inicial físico al valor final se puede limitar la velocidad de modificación de la consigna.

Límites de la consigna

Si la consigna viene dada por el bloque de función o es un valor analógico acondicionado de una entrada analógica, se limita a un límite inferior y superior especificables.

Si en el regulador de relación se ha seleccionado como consigna la variable manipulada acondicionada de otro canal de regulación, este valor actúa como factor para la multiplicación del valor real D. La consigna aplicada a la entrada en % se normaliza en este caso con los límites inferior y superior.

Si en el regulador de valor fijo o en cascada se utiliza como consigna la variable manipulada de otro canal de regulación, la consigna se normaliza a un valor físico con las constantes de normalización del canal de valor real seleccionado.

Multiplicar

Con el tipo de regulador de relación, el valor real A se utiliza como magnitud de regulación y el valor real D, como magnitud de relación. La entrada de consigna sirve como factor de relación. Multiplicándolo por el valor real D y sumando un offset ajustable, se obtiene una consigna efectiva. Si el valor real D se desconecta, entonces sólo se suma el offset a la consigna.

Acondicionamiento del valor real

El acondicionamiento del valor real para obtener el valor real efectivo depende de la estructura del regulador.

Valor real efectivo con regulador de valor fijo o en cascada

El valor real efectivo es igual que el valor real A. El origen del valor real A puede ser una entrada analógica. Los valores reales B, C y D no se tienen en cuenta.

Valor real efectivo en regulador de tres componentes

El valor real efectivo se calcula sumando los valores reales A, B y C, así como un offset ajustable. Los valores reales B y C se pueden valorar adicionalmente por factores. El valor real D no se tiene en cuenta.

Valor real efectivo en un regulador de relación

Para un regulador de relación seleccione los siguientes valores reales

- Valor real A: La señal acondicionada de una entrada analógica.
- Valor real D: El valor real del regulador piloto.

El factor de proporción se predetermina como consigna.

Si se ha seleccionado una salida de regulador como factor de proporción FAC, con ayuda de la barrera inferior y superior de "0... 100%" la consigna se convierte (normaliza) al rango de valores "barrera inferior ... barrera superior".

Valor real efectivo en reguladores de mezcla

Para un regulador de mezcla seleccione los siguientes valores reales

- Valores reales A, B y C: Señales acondicionadas de las respectivas entradas analógicas.
- Valor real D: El valor real del regulador piloto.

Los factores de proporción de cada uno de los sistemas regulados se definen como consigna.

La magnitud manipulada LMN del regulador del flujo total se indica en el rango de valores "0 ... 100%". El regulador de seguimiento convierte (normaliza) esta magnitud de la entrada Valor real D al rango de valores "barrera inferior ... barrera superior" del valor real A.

La magnitud manipulada del regulador piloto se aplica a través de la entrada "valor real D" del regulador de seguimiento. El factor de proporción FAC se define a través de la entrada de consigna del regulador.

Alarma

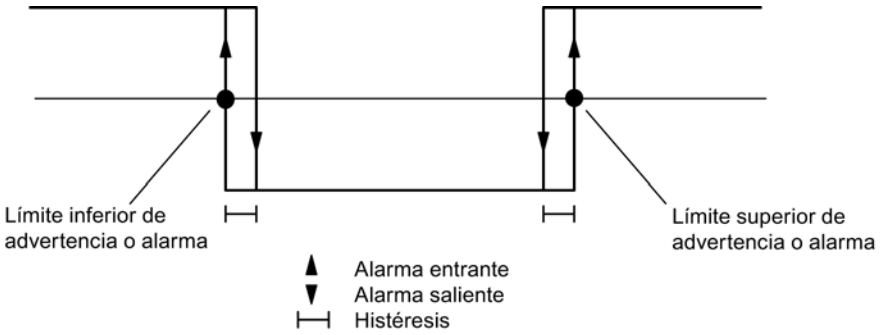
Vigilancia de límites

En el módulo de regulación se lleva a cabo una vigilancia de límites. Esta permite vigilar

- el error de regulación
- o el valor real efectivo

dentro de un límite de advertencia superior e inferior y un límite de alarma superior e inferior. Además, se puede ajustar una histéresis para estos límites.

La figura siguiente muestra la histéresis del límite de advertencia y de alarma:



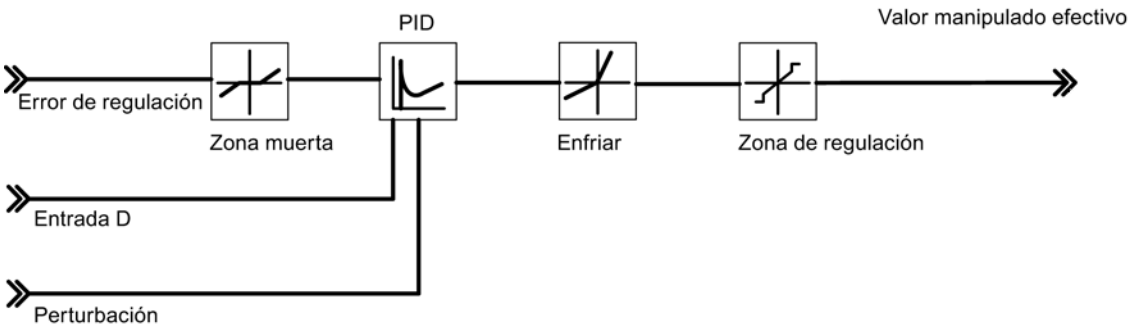
Algoritmo de regulación

Algoritmo de regulación FM 355-2

Componentes del algoritmo de regulación

El regulador continuo (FM 355-2 C) y el regulador de impulsos (FM 355-2 S) tienen la misma estructura de algoritmo de regulación. En el regulador paso a paso (FM 355-2 S) no se pueden seleccionar los botones Enfriar ni Zona de regulación.

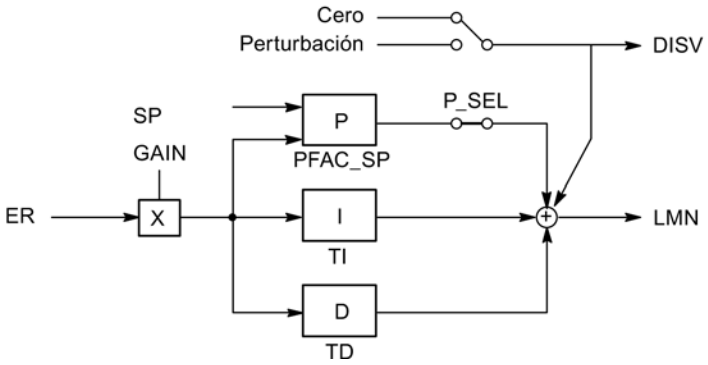
La figura siguiente muestra el diagrama de bloques del algoritmo de regulación para reguladores continuos y reguladores de impulsos:



Algoritmo de regulación: PID en estructura paralela

En el ciclo del tiempo de muestreo configurado se calcula la magnitud manipulada del regulador a partir del error de regulación del algoritmo de posición PID. El regulador tiene una estructura puramente paralela. La acción proporcional, integral o diferencial se puede desconectar individualmente. En el caso de la acción integral y diferencial esto se realiza poniendo a cero los correspondientes parámetros TI o TD.

La figura siguiente muestra el algoritmo de regulación del FM 355-2 (estructura en paralelo):



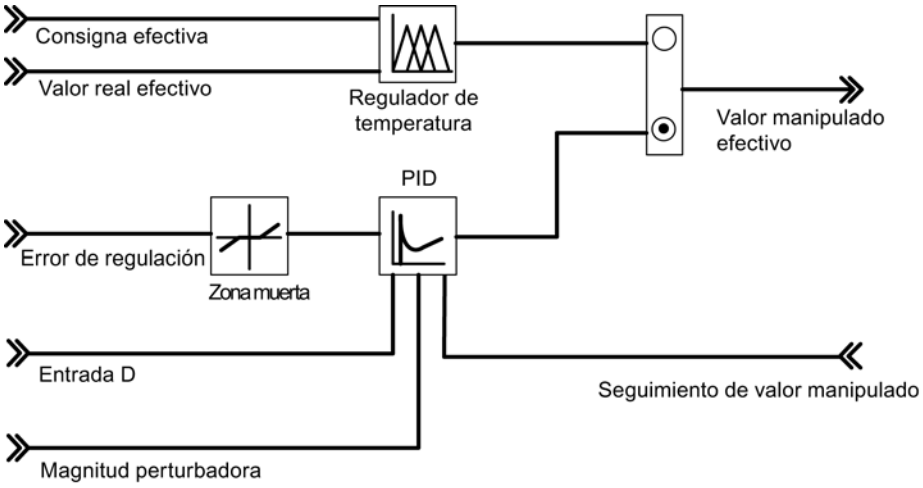
Algoritmo de regulación FM 355 y FM 455

Componentes del algoritmo de regulación

En el algoritmo de regulación se pueden seleccionar los siguientes modos de operación:

- Regulador de temperatura (regulador fuzzy con autoajuste)
- Regulador PID

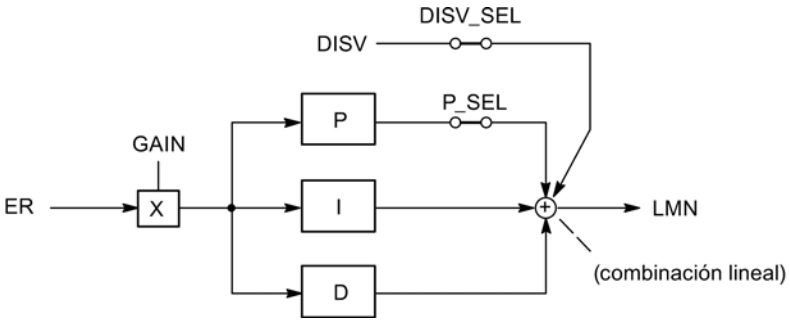
El regulador K y el regulador S tienen la misma estructura del algoritmo de regulación (véase la figura que sigue a continuación). La figura siguiente muestra el diagrama de bloques del algoritmo de regulación:



Algoritmo de regulación y estructura del regulador

En el ciclo del tiempo de muestreo configurado se calcula la magnitud manipulada del regulador continuo a partir del error de regulación del algoritmo de posición PID. El regulador tiene una estructura puramente paralela (véase figura inferior). La acción proporcional, integral o diferencial se puede desconectar individualmente. En el caso de las acciones integral y diferencial, esto se realiza poniendo a cero los correspondientes parámetros TI o TD.

La figura siguiente muestra el algoritmo de regulación del FM 355 (estructura en paralelo):



Seleccionar el modo de operación

Selección del modo de operación

Si como tipo de regulador se ha seleccionado la entrada "regulador PID", se puede elegir entre los diferentes modos de operación:

- P, para materializar un regulador P
- PI, para materializar un regulador PI
- PD, para materializar un regulador PD
- PID, para materializar un regulador PID

- I, para materializar un regulador I
- ID, para materializar un regulador ID
- D, para materializar un regulador D

Conmutación suave de manual a automático

Si ha seleccionado la conmutación suave de manual a automático (no con reguladores paso a paso), el integrador en modo manual se va corrigiendo de tal forma que la magnitud manipulada no da ningún salto por la acción P y D en la conmutación de manual a automático. Cualquier error de regulación se corrige siempre lentamente con la acción I. Si no se ha seleccionado la conmutación suave de manual a automático, en la conmutación de manual a automático la magnitud manipulada da un salto partiendo del valor manual actual, y el salto se corresponde con el error de regulación actual. Cualquier error de regulación se corrige rápidamente.

Nota

En el caso del regulador paso a paso, la conmutación al modo automático siempre se efectúa de forma brusca. El error de regulación existente produce a través de GAIN un cambio de la magnitud manipulada interna en forma de escalón. Como la acción del elemento final de control es integral se obtiene tan solo un control del proceso en forma de rampa.

Control anticipativo

A la señal de salida del regulador se le puede aplicar adicionalmente una magnitud perturbadora DISV. La activación o desactivación se realiza en la ventana "Error de regulación" de la herramienta de configuración a través del interruptor "Selección de señal magnitud manipulada regulador".

Zona muerta

Finalidad de la zona muerta

Existe una zona muerta antepuesta al regulador PID. Con el regulador estabilizado, la zona muerta suprime el nivel de ruido en la señal del error de regulación que puede producirse por superposición de la magnitud de regulación o consigna con una señal interferente de alta frecuencia. De esta forma se evita la oscilación no deseada de la salida del regulador.

Ancho de la zona muerta

El ancho de la zona muerta es ajustable. Si el error de regulación se encuentra dentro del ancho de zona muerta ajustado, se emite el valor 0 (error de regulación = 0) en la salida de la zona muerta. Sólo cuando la magnitud de entrada abandona el rango de sensibilidad, la salida cambia con los mismos valores que la magnitud de entrada.

De esto resulta una falsificación de la señal transmitida incluso fuera de la zona muerta. Sin embargo, esto se tolera para evitar escalones en los límites de la zona muerta. La falsificación se corresponde con el valor del ancho de la zona muerta y es, por ello, fácil de controlar.

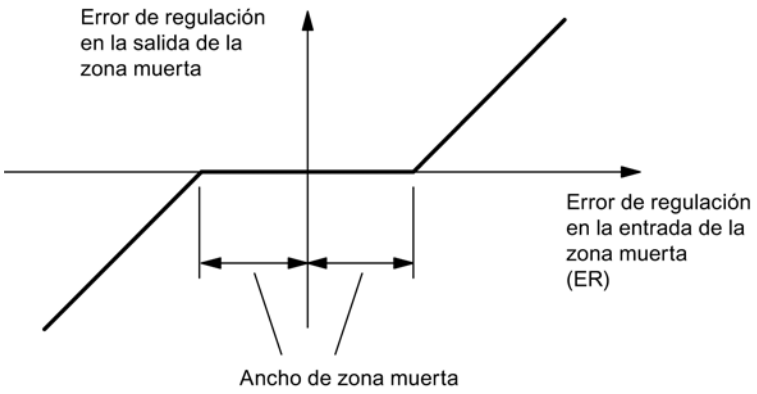
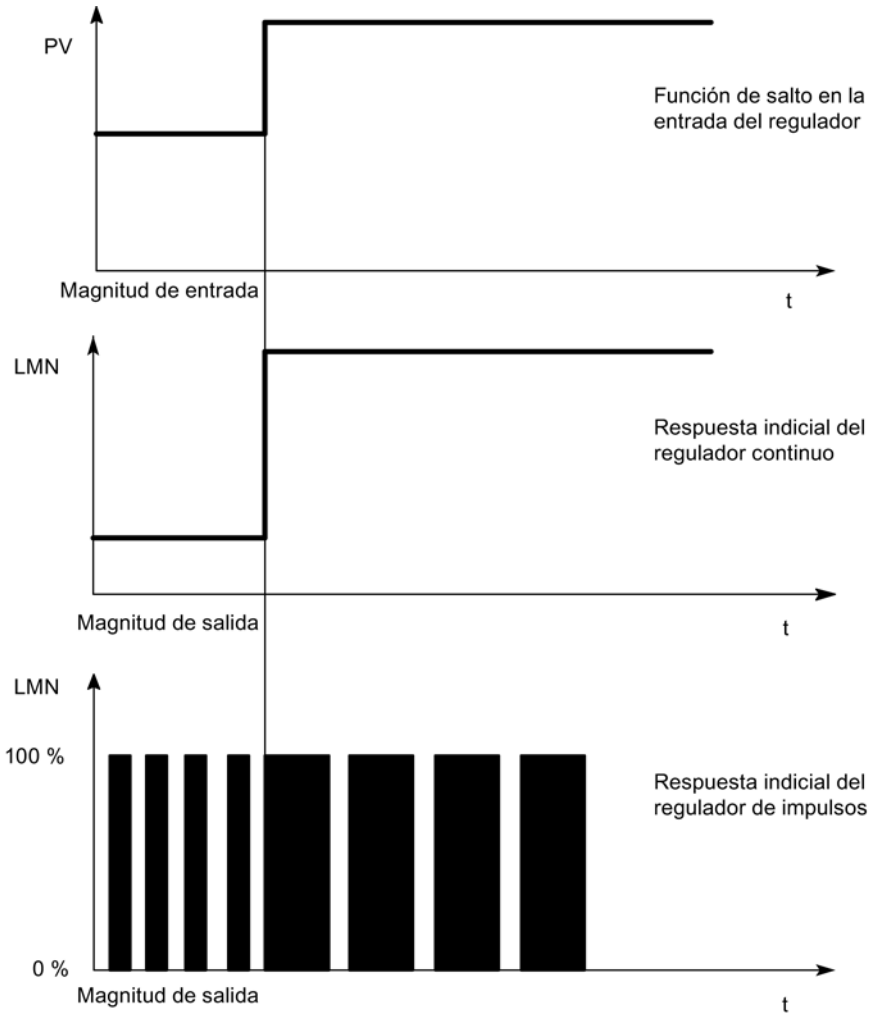


Figura 11-9 Zona muerta

Regulación P

Regulador P

La figura siguiente muestra la respuesta indicial de un regulador P:



Fórmula para reguladores P

Las magnitudes de salida y de entrada son directamente proporcionales, es decir:

Modificación de la magnitud de salida = ganancia proporcional x modificación de la magnitud de entrada

LMN = GAIN x ER

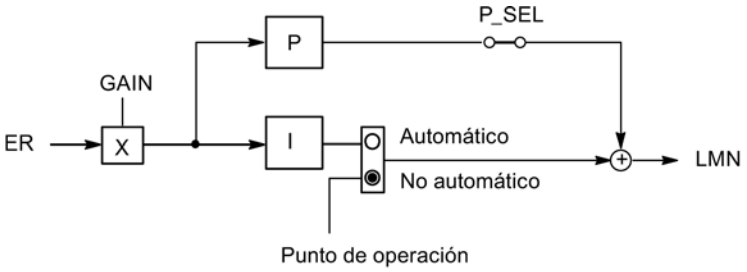
Regulación P

En el regulador P están desconectadas las acciones I y D. Esto quiere decir que en caso de error de regulación ER = 0, la magnitud manipulada también es = 0. Si se desea ajustar un

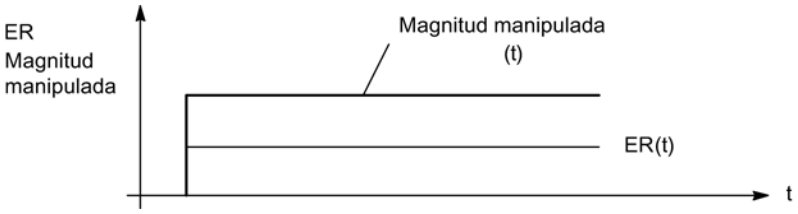
punto de operación ≠0, es decir, un valor numérico para la magnitud manipulada en caso de error de regulación 0, esto se puede realizar a través del punto de operación:

- Punto de operación automático: El regulador pone el punto de operación a la magnitud manipulada (manual) actual automáticamente durante la conmutación de manual a automático.
- Punto de operación no automático: Se puede parametrizar el punto de operación.
- Ejemplo: Punto de operación AP = 5% da una magnitud manipulada del 5% en caso de error de regulación ER = 0.

La figura siguiente muestra un regulador P con ajuste del punto de operación a través del componente I:



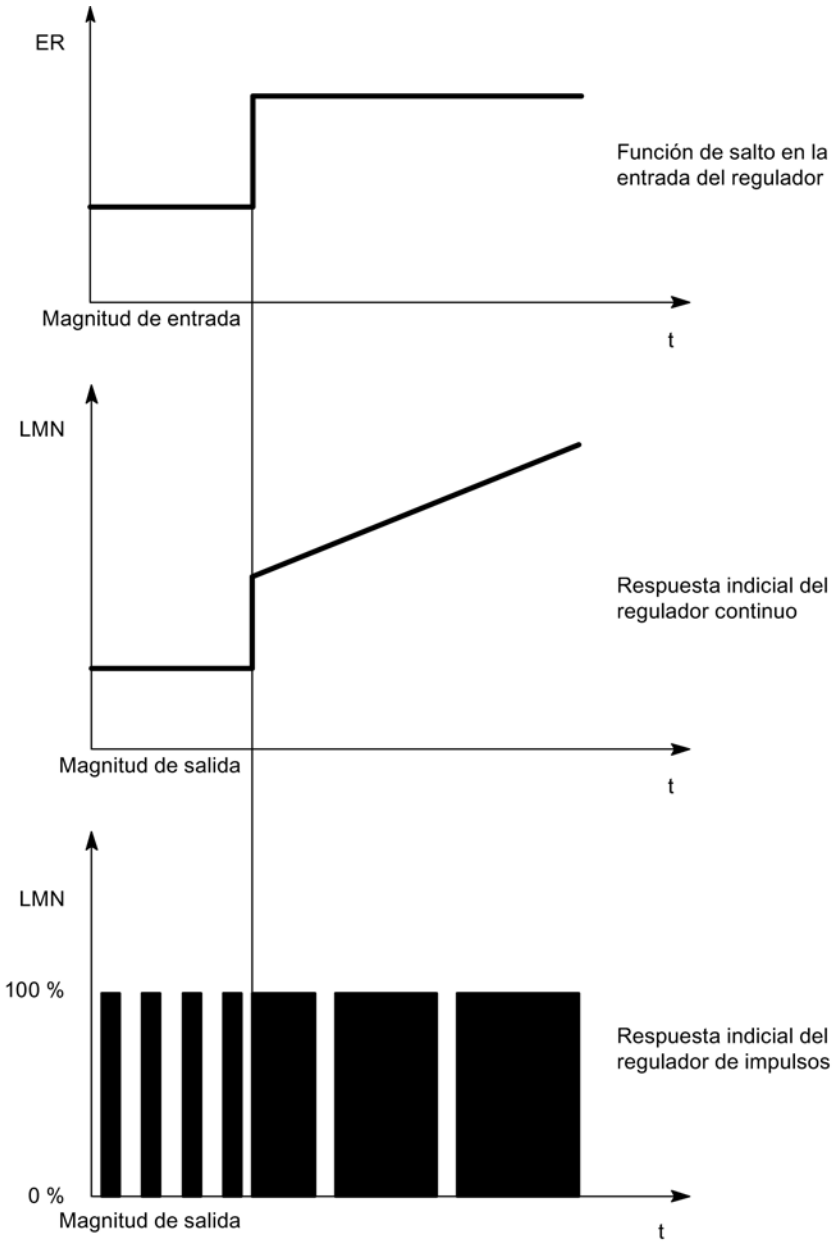
La figura siguiente muestra la respuesta indicial del regulador P:



Regulación PI

Regulador PI

La figura siguiente muestra la respuesta indicial de un regulador PI:



Los reguladores I tienen como magnitud de salida la integral de la magnitud de entrada, es decir, el regulador suma la diferencia de la consigna en el tiempo. Esto quiere decir que el regulador va corrigiendo hasta que se elimine la diferencia con la consigna. En la práctica, según las exigencias en cuanto a la respuesta del regulador, lo ideal es una combinación de los diferentes temporizadores. La respuesta de tiempo de cada uno de los componentes se

puede describir con los parámetros de regulación de ganancia proporcional GAIN, tiempo de acción integral TI (acción I) y tiempo de la acción derivada TD (acción D).

Fórmula para reguladores PI

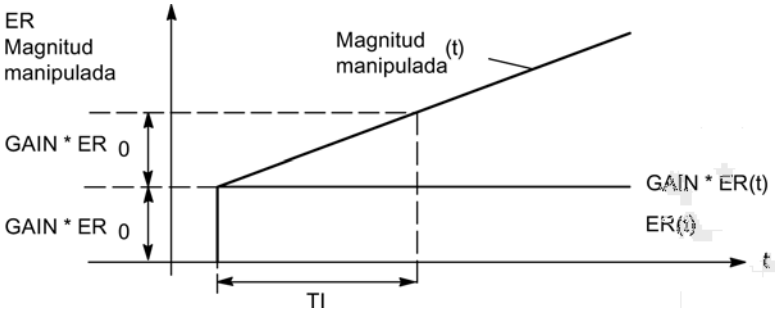
Para la respuesta indicial del regulador PI en el rango temporal rige lo siguiente:

$$LMN = GAIN \times ER \times \left(1 + \frac{t}{TI}\right)$$

t = tiempo desde el escalón de la magnitud de entrada

Regulación PI

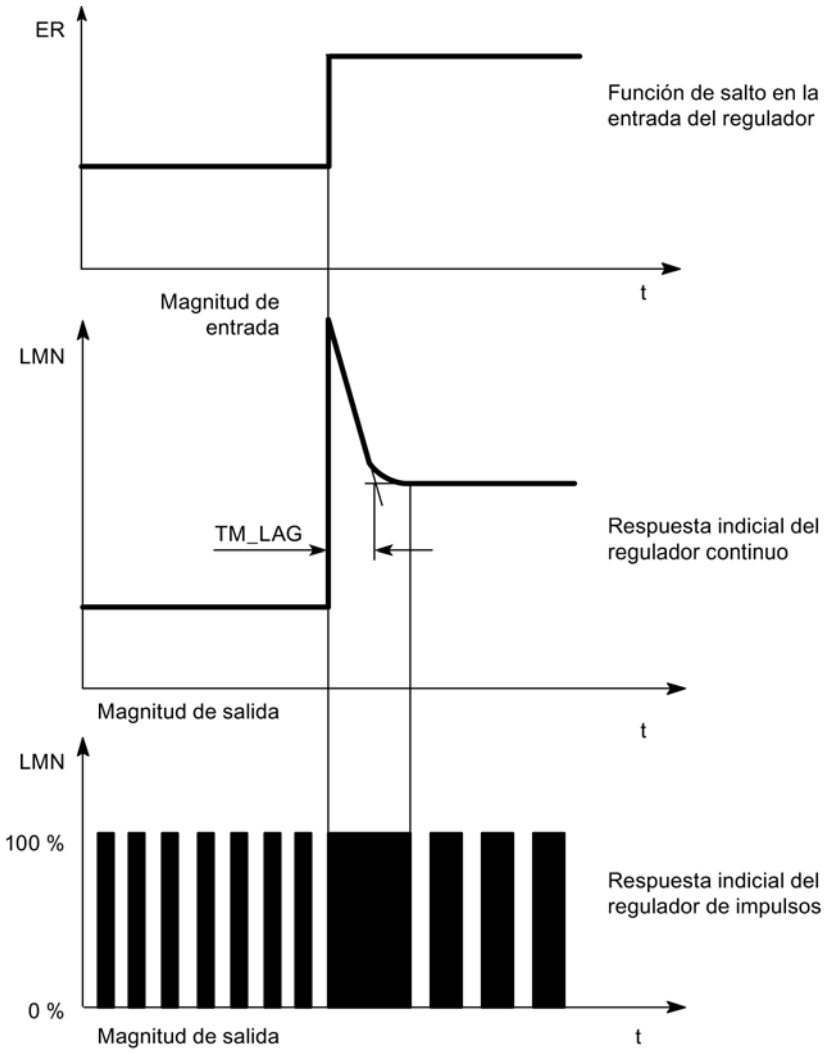
En el regulador PI está desconectada la acción D (TD=0.0). Un regulador PI modifica la magnitud de salida con la acción I hasta que el error de regulación sea ER = 0. Sin embargo, esto sólo es aplicable cuando la magnitud de salida no rebasa los límites de la zona regulada. Si se rebasan los límites de la variable manipulada, la acción I mantiene el valor alcanzado en el límite en (Anti-Reset-Windup).



Regulación PD

Regulador PD

La figura siguiente muestra la respuesta indicial de un regulador PD:



Los reguladores D no son adecuados para la regulación si funcionan solos, ya que dejan de dar órdenes para el elemento final de control cuando la magnitud de entrada regresa a un valor estático.

En combinación con reguladores P, la acción D se utiliza para generar un impulso de control adecuado según la velocidad de modificación de la magnitud de regulación. Si en el sistema regulado actúa una perturbación z, el regulador PD cambia a otro error de regulación debido al cambio del grado de regulación. Las perturbaciones no se eliminan por completo. El comportamiento beneficioso es el bueno y dinámico. Durante la aproximación y la modificación de las consignas se alcanza una transición bien amortiguada y sin vibraciones. Sin embargo,

un regulador con acción D no es apropiado si el sistema regulado tiene magnitudes de medición pulsantes, p. ej. en la regulación de presión o de caudal.

Fórmula para reguladores PD

Para la respuesta indicial del regulador PD en el dominio temporal rige lo siguiente:

$$LMN = GAIN \times ER \times \left(1 + \frac{TD}{TM_LAG} \times e^{\frac{-t}{TM_LAG}}\right)$$

t = tiempo desde el escalón de la magnitud de entrada

Regulación PD

En el regulador PD está desconectada la acción I (TI=0.0). Esto quiere decir que en caso de error de regulación ER = 0, la señal de salida también es = 0. Si se desea ajustar un punto de operación ≠0, es decir, un valor numérico para la magnitud manipulada en caso de error de regulación 0, esto se puede realizar a través del punto de operación:

- Punto de operación automático: El regulador pone el punto de operación a la magnitud manipulada (manual) actual automáticamente durante la conmutación de manual a automático.
- Punto de operación no automático: Se puede parametrizar el punto de operación.

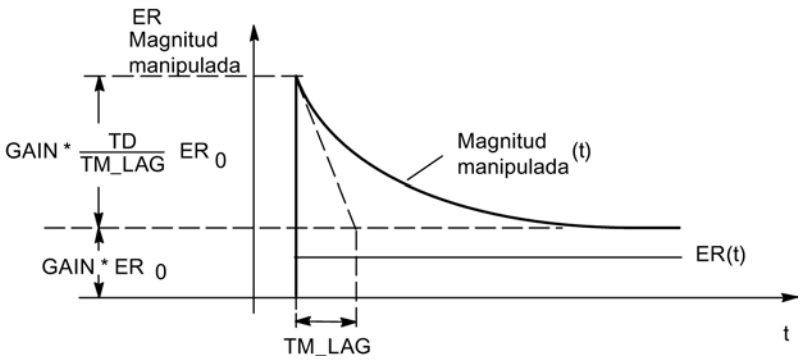
El regulador PD reproduce la magnitud de entrada ER(t) proporcionalmente sobre la señal de salida y añade la acción D creada por la diferenciación de ER(t), que se calcula con el doble de precisión según la regla del trapecio (aproximación de Padé). La respuesta de tiempo (fuerza de la acción D o superficie de regulación) se determina por el tiempo de la acción derivada TD (tiempo de acción derivada).

Para el filtrado de la señal y la supresión de señales perturbadoras, la acción D se realiza mediante un retardo de primer orden (constante de tiempo ajustable: TM_LAG). La mayoría de las veces es suficiente con un valor pequeño para TM_LAG, para alcanzar el resultado deseado.

Cuanto mayor seleccione el factor derivativo D_F,

- menor será la constante de tiempo efectiva TD/D_F del retardo y
- mayor será la variable manipulada máxima al inicio y
- mejor la respuesta del regulador y
- mayor también la sensibilidad a ruidos.

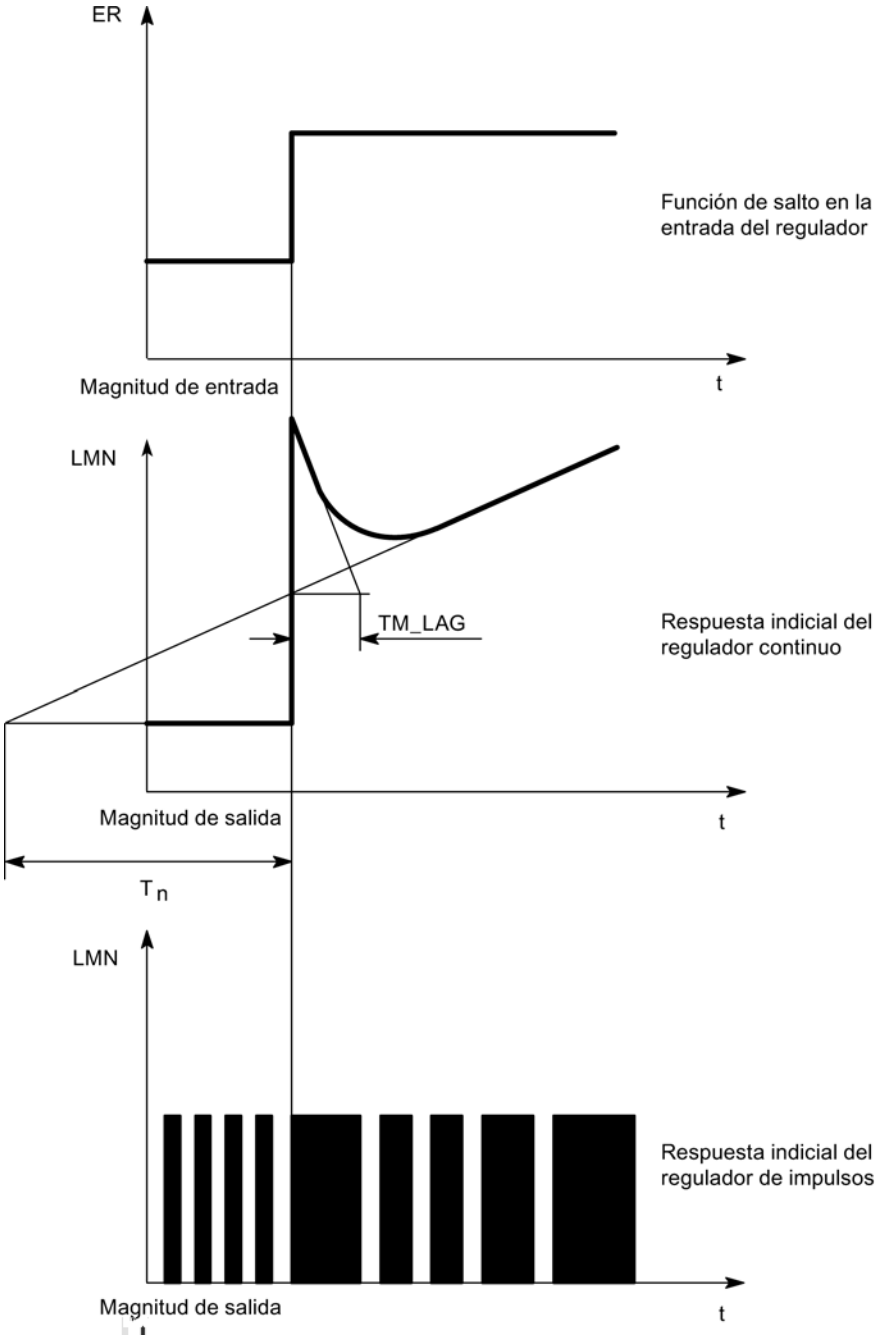
D_F se limita al rango de valores entre 5,0 y 10,0.


ER
Magnitud
manipulada
GAIN * TD / TM_LAG ER 0
Magnitud
manipulada (t)
GAIN * ER 0
ER(t)
TM_LAG
t

Regulación PID

Regulador PID

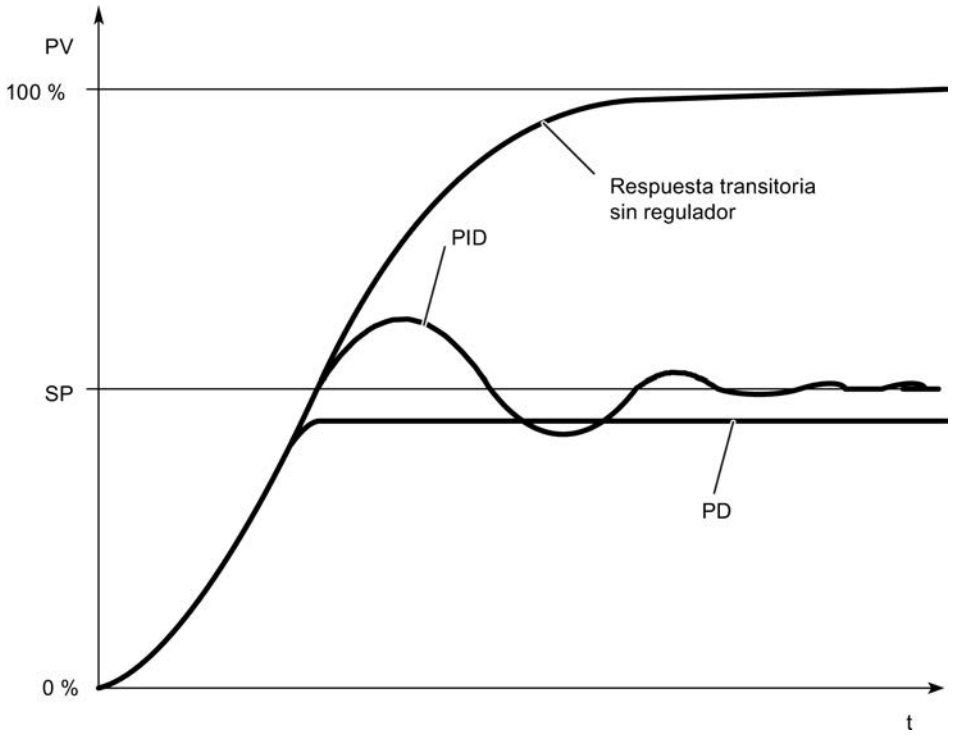
La figura siguiente muestra la respuesta indicial de un regulador PID:



La mayoría de regulaciones que se aplican en la técnica de procesos se puede controlar con un regulador con comportamiento PI. En sistemas regulados lentos con grandes tiempos de

retardo, p. ej. regulaciones de temperatura, el resultado de la regulación se puede mejorar con un regulador con comportamiento PID.

La figura siguiente muestra la respuesta indicial con distintos comportamientos del regulador:



Los reguladores con comportamiento PI y PID tienen la ventaja de que tras la fase de estabilización la magnitud de regulación no presenta ninguna diferencia respecto a la consigna. La magnitud regulada oscila sobre la consigna durante la aproximación.

Fórmula para reguladores PID

Para la respuesta indicial del regulador PI en el dominio temporal rige lo siguiente:

$$LMN = GAIN \times ER \times \left(1 + \frac{t}{TI} + \frac{TD}{TM_LAG} \times e^{\frac{-t}{TM_LAG}}\right)$$

t = tiempo desde el escalón de la magnitud de entrada

Regulación PID

En el regulador PID están conectadas la acción P, I y D. Un regulador PID modifica la magnitud de salida con la acción I hasta que el error de regulación sea ER = 0. Sin embargo, esto sólo es aplicable cuando la magnitud de salida no rebasa los límites de la zona regulada. Si se rebasan los límites de la variable manipulada, la acción I mantiene el valor alcanzado en el límite en (Anti-Reset-Windup).

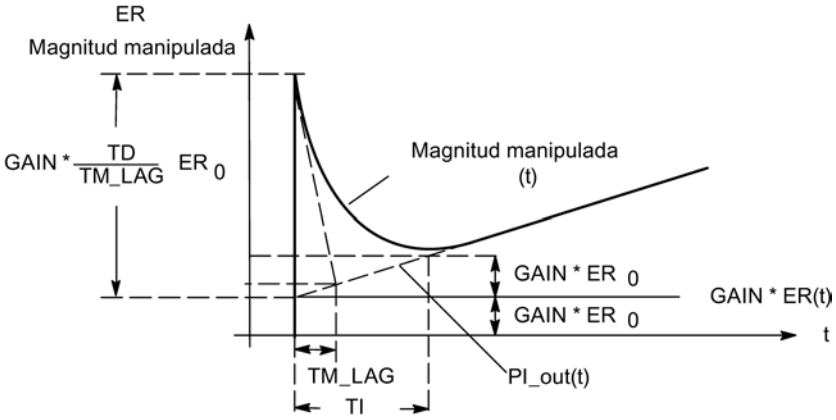
El regulador PID reproduce la magnitud de entrada ER(t) proporcionalmente sobre la señal de salida y añade los componentes creados por la diferenciación e integración de ER(t). La respuesta de tiempo se determina por el tiempo de la acción derivada TD (tiempo de la acción derivada) y el tiempo de integración TI (tiempo de acción integral).

Para el filtrado de la señal y la supresión de señales interferentes, la acción D se realiza a través de un retardo de primer orden.

Cuanto mayor seleccione el factor derivativo D_F,

- menor será la constante de tiempo efectiva TD/D_F del retardo y
- mayor será la variable manipulada máxima al inicio y
- mejor la respuesta del regulador y
- mayor también la sensibilidad a ruidos.

D_F se limita al rango de valores entre 5,0 y 10,0.



Regulación I

Regulador I

Para realizar una regulación I se puede desconectar la acción P. Esto también es posible con el parámetro P_SEL del módulo de función PID_FM.

Acción P

Atenuación de la acción P en caso de modificaciones de la consigna

Para evitar sobreoscilaciones del valor real o desviaciones importantes de la variable manipulada, la acción P se puede atenuar con el parámetro Factor proporcional en caso de modificaciones de la consigna (PFAC_SP). Con PFAC_SP se puede seleccionar de forma

continua entre 0.0 y 1.0 cómo debe actuar la acción P en caso de modificaciones de la consigna:

- PFAC_SP=1.0: La acción P es totalmente efectiva cuando se cambia la consigna
- PFAC_SP=0,0: La acción P no es efectiva cuando se cambia la consigna.

En el FM355-2 existen dos posibilidades de limitar la velocidad de una modificación de la consigna:

- Activar rampa (> 0,0 s)
- Factor de modificación de la consigna < 1,0

Utilice sólo una de las dos limitaciones. Si se activan simultáneamente ambas limitaciones, en caso de un escalón de la consigna aparece una modificación de la variable manipulada inversa a la modificación de la consigna (respuesta indicial).

Particularidades en el regulador paso a paso

Un valor de PFAC_SP < 1,0 puede reducir las sobreoscilaciones si el tiempo de posicionamiento del motor MTR_TM es corto frente al tiempo de estabilización TA y la relación TU/TA es <0,2. Si MTR_TM alcanza el 20% de TA, ya sólo se podrá alcanzar una ligera mejora.

Inversión del efecto del regulador

La inversión, es decir, el cambio del regulador de la asignación

- error de regulación ascendente = magnitud manipulada ascendente
  a
- error de regulación ascendente = magnitud manipulada descendente

se consigue ajustando una ganancia proporcional negativa (GAIN). El signo de este valor de parámetro determina el sentido de acción del regulador.

Acción D

Acción D en la rama de realimentación

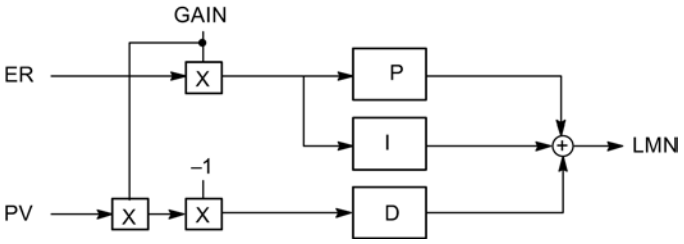
Para evitar desviaciones en forma de impulso en la acción D de la magnitud manipulada en caso de modificaciones de la consigna, se puede colocar la acción D en la rama de realimentación.

En esta estructura sólo se conecta a la acción D la magnitud de regulación negativa (factor = -1). En la acción D, la conmutación a la realimentación se lleva a cabo en la ventana "Error de regulación" con el interruptor "Entrada D regulador" seleccionando como señal de entrada el valor real efectivo negado.

Nota

Si se coloca la acción D en realimentación, también debe reducirse PFAC_SP, de lo contrario, aumenta la sobreoscilación del valor real.

La figura siguiente muestra el algoritmo de regulación con acción D en la rama de realimentación:



Acción P/D

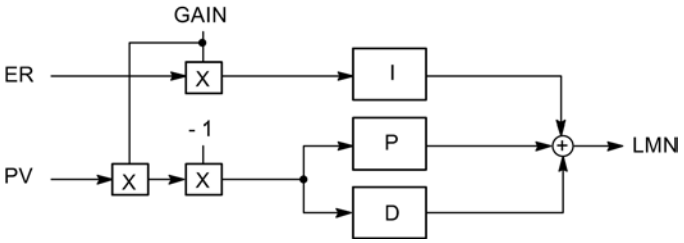
Acción P/D en la rama de realimentación

En la estructura paralela, cada acción del algoritmo de regulación recibe el error de regulación como señal de entrada. En esta estructura los escalones de consigna actúan directamente en el regulador. La magnitud manipulada se modifica inmediatamente a través de la acción P y D por escalones de la consigna.

Sin embargo, otra estructura del regulador en la que la formación de la acción P y D se traslada a la realimentación, garantiza un transcurso suave de la magnitud manipulada en caso de variaciones del escalón de la consigna (véase la siguiente figura).

En esta estructura la acción I procesa el error de regulación como señal de entrada, en la acción P y D sólo se conecta la magnitud de regulación **negativa** (factor = -1). En la acción D, la conmutación a la realimentación se lleva a cabo en la ventana "Error de regulación" con el interruptor "Entrada D regulador" seleccionando como señal de entrada el valor real efectivo negado.

La figura siguiente muestra el algoritmo de regulación con acción D en la rama de realimentación:



Módulo de regulación de temperatura por lógica difusa

Regulador de temperatura

El regulador de temperatura es un regulador fuzzy con autoajuste que, tras identificar el sistema regulado, trabaja con parámetros de regulación determinados por él mismo.

En los reguladores de temperatura son posibles los siguientes ajustes:

- Regulador de enfriamiento
- Regulador de calefacción
- Agresividad
  Con el parámetro Agresividad se puede modificar la velocidad del comportamiento transitorio.

Valores posibles para la agresividad	
-1 ≤ agresividad < 0	Comportamiento transitorio más lento que determinado mediante identificación
Agresividad = 0	Comportamiento de estabilización como determinado mediante identificación
0 < agresividad ≤ 1	Comportamiento transitorio más rápido que determinado mediante identificación

Salida del regulador

Diferencias de los reguladores

Básicamente los reguladores FM 355, FM 355-2 y FM 455 pueden dividirse en dos tipos: en reguladores paso a paso y reguladores continuos. Los reguladores paso a paso van acompañados de una S complementaria, a saber, FM 355 S, FM 355-2 S y FM 455 S, mientras que los reguladores continuos van acompañados de una C, a saber, FM 355 C, FM 355-2 C y FM 455 C. El restante funcionamiento de los módulos de regulación no se diferencia, un FM 355 S y un FM 355 C poseen algoritmos idénticos. Las diferencias se encuentran en la salida del regulador.

Regulador paso a paso

El regulador continuo paso a paso posee salidas digitales. De este modo, una válvula de control sólo puede activarse a través de las señales "abierta" o "cerrada". Por ejemplo, si una válvula de control se regula al 50% de la capacidad, esto sólo puede conseguirse si se efectúa un ajuste igual de largo de "abierta" y "cerrada".

Encontrará más información acerca de los reguladores paso a paso en los puntos siguientes:

- Salidas digitales del regulador (Página 10244)
- Regulador paso a paso con realimentación de posición (Página 10246)
- Regulador paso a paso con realimentación de posición (Página 10246)
- Regulador de impulsos (Página 10246)

Regulador continuo

El regulador continuo paso a paso posee salidas analógicas. Así pues, una válvula de control no sólo puede estar "abierta" o "cerrada", sino que también puede abrirse en determinadas proporciones. Por ejemplo, si una válvula de control se regula al 50% de la capacidad, el módulo de regulación emite el valor "50%" y la válvula se abre a la mitad.

Encontrará más información acerca de los reguladores continuos en los puntos siguientes:

- Salidas analógicas del regulador (Página 10244)
- Regulador continuo (Página 10245)

Salidas analógicas del regulador

Salidas analógicas

Para cada salida analógica se pueden parametrizar las siguientes funciones:

- Origen de la señal
- Tipo de señal

Selección del origen para la señal en las salidas analógicas

Con la selección del origen de la señal se puede determinar qué valor de señal se emite en la correspondiente salida analógica.

Se pueden parametrizar los siguientes orígenes de la señal:

- El valor cero
- El valor analógico âcondicionado de una de las entradas analógicas
- La variable manipulada A de uno de los canales de regulación
- La variable manipulada B de uno de los canales de regulación

El FM 355 y el FM 355-2 tienen cuatro entradas analógicas y cuatro canales de regulación cada uno.

El FM 455 tiene 16 entradas analógicas y 16 canales de regulación.

Tipo de señal en las salidas analógicas

Para cada salida analógica se puede determinar el tipo de señal.

Se pueden parametrizar los siguientes tipos de señal:

- Salida de intensidad 0 ... 20 mA
- Salida de intensidad 4 ... 20 mA
- Salida de tensión 0 ... 10 V
- Salida de tensión -10 ... 10 V

Salidas digitales del regulador

Salidas digitales del regulador

Las salidas digitales del regulador sirven para controlar elementos finales de control integradores o no integradores.

Para cada salida digital se pueden parametrizar las siguientes funciones:

- Tipo de salida
- Selección de señal
- Procesamiento de señal

Tipo de salida de la señal en la salida digital

Para cada salida digital se puede determinar el tipo de señal.

Los siguientes tipos de señales están disponibles:

- Regulador paso a paso sin entrada de realimentación de posición
- Regulador paso a paso con entrada de realimentación de posición
- Regulador de impulsos

Selección de señal

Según el tipo de señal en la salida, se pueden parametrizar diferentes valores para la selección de la señal:

- Variable manipulada externa (Página 10247)
- Modo de seguimiento (Página 10247)
- Entrada de realimentación de posición (Página 10248)
- Variable manipulada de seguridad (Página 10249)

Procesamiento de señal

Según el tipo de señal en la salida, se pueden parametrizar diferentes valores para el procesamiento de la señal:

- Límite señal de salida (Página 10250)
- Rango partido (Página 10250)
- Formador de impulsos (Página 10253)

Regulador continuo

Tipo de señal en las salidas digitales

Los siguientes parámetros sólo se pueden ajustar en un módulo analógico.

Parámetros ajustables

- Activar variable manipulada externa (Página 10247)
- Activar modo de seguimiento (Página 10247)
- Ajustar variable manipulada de seguridad (Página 10249)

- Limitar la variable manipulada (Página 10250)
- Controlar dos actuadores (Página 10250)
- Seleccionar señal de salida y tipo de señal (Página 10253)

Regulador paso a paso con realimentación de posición

Tipo de señal en las salidas digitales

Los siguientes parámetros sólo se pueden ajustar en un módulo digital.

Parámetros ajustables

- Activar variable manipulada externa (Página 10247)
- Ajustar variable manipulada de seguridad (Página 10249)
- Formador de impulsos (Página 10253)

Regulador paso a paso con realimentación de posición

Tipo de señal en las salidas digitales

Los siguientes parámetros sólo se pueden ajustar en un módulo digital.

Parámetros ajustables

- Activar variable manipulada externa (Página 10247)
- Activar modo de seguimiento (Página 10247)
- Ajustar variable manipulada de seguridad (Página 10249)
- Limitar la variable manipulada (Página 10250)
- Ajustar realimentación de posición (Página 10248)
- Formador de impulsos (Página 10253)

Regulador de impulsos

Tipo de señal en las salidas digitales

Los siguientes parámetros sólo se pueden ajustar en un módulo digital.

Parámetros ajustables

- Activar variable manipulada externa (Página 10247)
- Activar modo de seguimiento (Página 10247)

- Ajustar variable manipulada de seguridad (Página 10249)
- Limitar la variable manipulada (Página 10250)
- Preparar señal analógica para la conversión a una señal digital (Página 10250)
- Convertir señal analógica en señal digital (Página 10214)

Funciones de la salida del regulador

Variable manipulada externa

Resumen

En la salida del regulador se han implantado varias posibilidades de interconexión para la variable manipulada, la entrada de corrección y la variable manipulada de seguridad (conmutación de variable manipulada).

Para evitar que la variable manipulada pueda aceptar valores no admisibles para el proceso, se ha previsto una limitación.

Activar variable manipulada externa

La conmutación entre la variable manipulada externa y la variable manipulada efectiva del regulador se realiza de manera alternativa por

- un valor binario del bloque de función
- una señal que se produce por combinación lógica con O de un valor binario del bloque de función y una entrada digital

Modo de seguimiento

Resumen

La corrección de la variable manipulada evita un salto de la variable manipulada durante la conmutación de manual a automático.

En la conmutación de manual a automático la variable manipulada permanece inalterada. La corrección de la variable manipulada no está activa si se realiza un regulador P puro con punto de operación fijo (en la máscara Regulador PID no está seleccionada la opción "automático").

Entrada de corrección

Existen los siguientes ajustes alternativos:

- La entrada de corrección tiene el valor cero
- La entrada de corrección es el valor analógico acondicionado de una entrada analógica

Activar seguimiento

La conmutación entre variable ipulada y entrada de corrección se realiza alternativamente por

- un valor binario del bloque de función
- una señal que se produce por combinación lógica con O de un valor binario del bloque de función y una entrada digital

Realimentación de posición

Resumen

La entrada de realimentación de posición sólo está disponible en reguladores S.

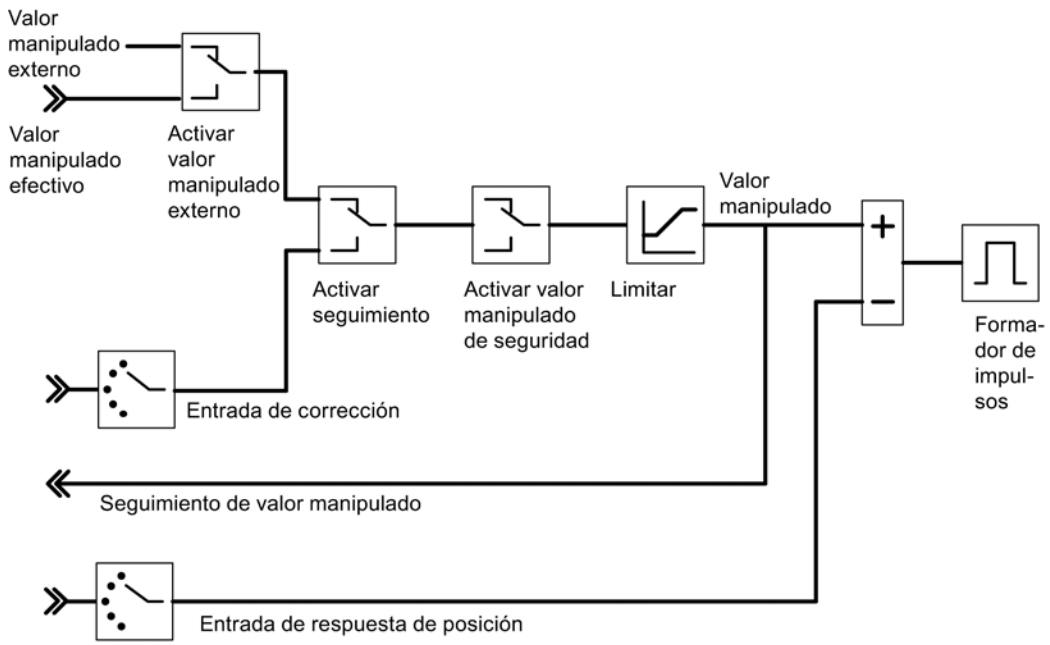
Entrada de realimentación de posición

Existen los siguientes ajustes alternativos:

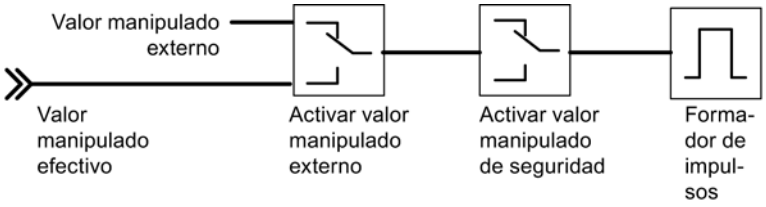
- La entrada de realimentación de posición tiene el valor cero
- La entrada de realimentación de posición es el valor analógico acondicionado de una entrada analógica

salida del regulador paso a paso

La figura siguiente muestra el esquema de la salida del regulador S (modo de operación Regulador paso a paso con realimentación de posición)



La figura siguiente muestra el esquema de la salida del regulador S (modo de operación Regulador paso a paso sin realimentación de posición)



En el regulador paso a paso sin realimentación de posición analógica, la variable manipulada externa y la variable manipulada de seguridad actúan de la siguiente manera:

Si se asigna un valor entre 40,0% y 60,0%, entonces no se activa ninguna salida binaria y el elemento final de control permanece sin cambios.

Si se establece un valor > 60,0%, se devuelve "Señal de posicionamiento alta" hasta que se active la realimentación "Actuador en tope superior".

Si se establece un valor < 40,0%, se emite "Señal de posicionamiento baja" hasta que la realimentación accede a "Actuador en tope inferior".

Variable manipulada de seguridad

Resumen

En la salida del regulador se han implantado varias posibilidades de interconexión para la variable manipulada, la entrada de corrección y la variable manipulada de seguridad (conmutación de variable manipulada).

Conmutar a variable manipulada de seguridad

- Determinación de la variable manipulada de seguridad
- Reacción alternativa del módulo de función al arrancar:
  - El módulo de función pasa a lazo cerrado.
  - Como variable manipulada se devuelve la variable manipulada de seguridad
- La conmutación a la variable manipulada de seguridad se realiza de forma alternativa por
  - un valor binario del bloque de función
  - una señal que se produce por combinación lógica con O de un valor binario del bloque de función y una entrada digital

- Respuesta a fallo del transductor de medida del valor real A:
  - Con el ajuste "Lazo cerrado", el modo de operación del regulador permanece sin cambios
  - Con el ajuste "Variable manipulada = variable manipulada de seguridad" se conmuta a la variable manipulada de seguridad
- Reacción en caso de fallo del transductor de medida de una entrada analógica:
  - Con el ajuste "Lazo cerrado", el modo de operación del regulador permanece sin cambios
  - Con el ajuste "Variable manipulada = variable manipulada de seguridad" se conmuta a la variable manipulada de seguridad

Limitación de la variable manipulada

Resumen

Para evitar que la variable manipulada pueda aceptar valores no admisibles para el proceso, se ha previsto una limitación.

Limitación de la variable manipulada

- Limitación superior e inferior (no puede desactivarse)

Rango partido

Resumen

La función de rango partido genera a partir de la variable manipulada como señal de entrada dos señales de salida normalizables: variable manipulada A y variable manipulada B. De esta forma se pueden controlar p. ej. dos válvulas con una variable manipulada.

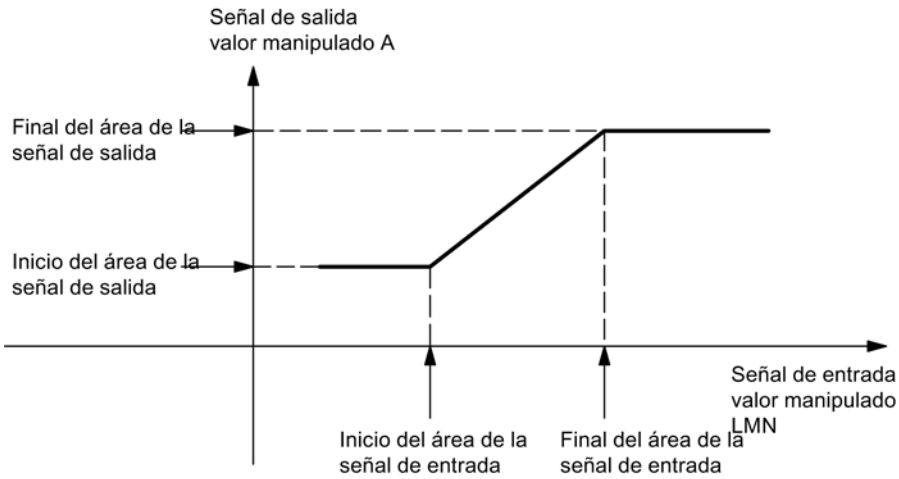
Formación de los valores de rango partido

- On/Off (sólo reguladores K)
- Valor inicial y valor final señal de entrada
- Valor inicial y valor final señal de salida

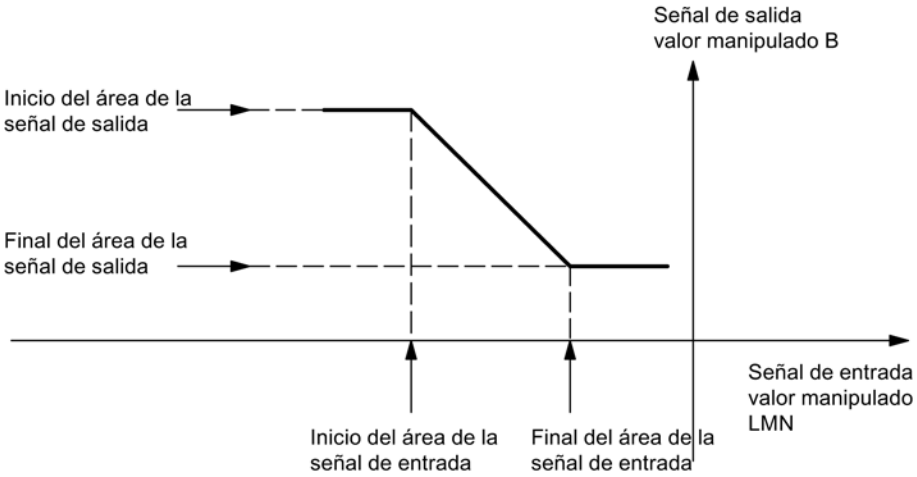
Rango partido

Con la función de rango partido se pueden controlar dos válvulas de control con una magnitud manipulada. La función de rango partido genera a partir de la variable manipulada LMN como señal de entrada las dos señales de salida variable manipulada A y variable manipulada B.

La figura siguiente muestra el efecto de los parámetros para la salida de la variable manipulada A:



La figura siguiente muestra el efecto de los parámetros para la salida de la variable manipulada B:



El inicio del área de la señal de entrada tiene que ser inferior que el final del área de la señal de entrada.

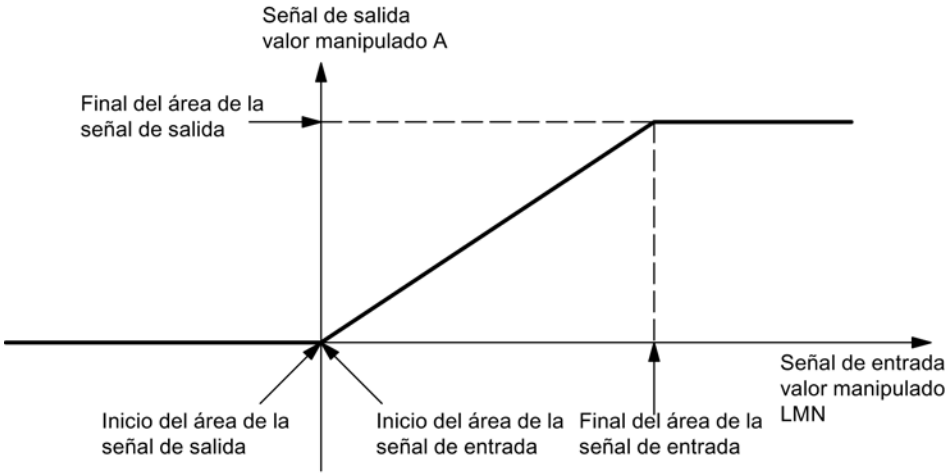
Rango partido/formador de impulsos

La función de rango partido es la preparación de la señal analógica para la conversión en una señal binaria.

En un **regulador de dos puntos** (p. ej. un regulador de calefacción) sólo es relevante la variable manipulada A. La conversión de la variable manipulada en la variable manipulada A se representa en la siguiente figura. La conversión en una señal de salida binaria se realiza de manera que, en la salida digital correspondiente, la relación entre el ancho de impulso y la duración del periodo se corresponda con la variable manipulada A.

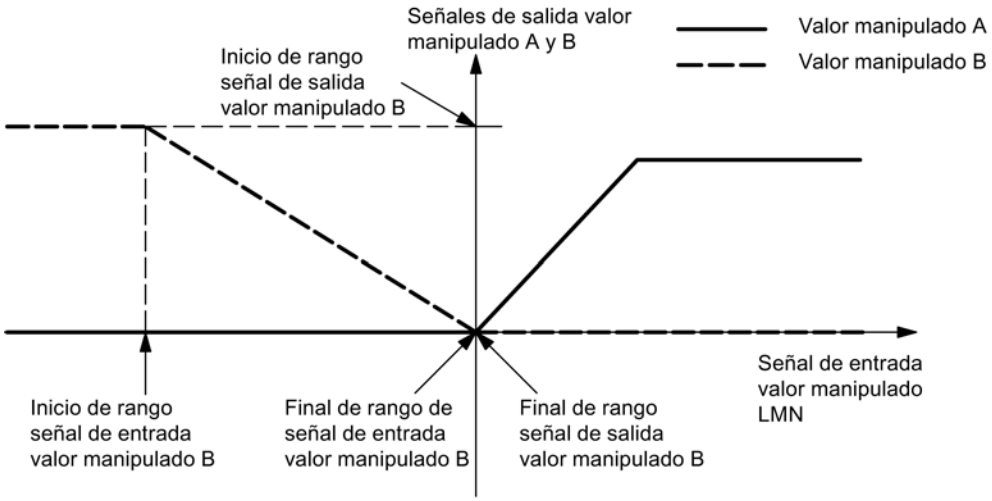
Por ejemplo, una variable manipulada A del 40% con una duración del periodo de 60 segundos da un ancho de impulso de 24 segundos y una longitud de pausa de 36 segundos.

La figura siguiente muestra la función de rango partido en un regulador de dos puntos:



En un **regulador de tres puntos** (p. ej. un regulador de calefacción y enfriamiento) se aplican las afirmaciones anteriores para la variable manipulada A. La segunda señal para el control del enfriamiento se genera a través de la variable manipulada B. La conversión de la variable manipulada en las variables manipuladas A y B se representa en la figura inferior. La conversión en una señal de salida binaria se realiza de manera que, en las salidas digitales correspondientes, la relación entre el ancho de impulso y la duración del periodo se corresponda con las variables manipuladas A o B.

La figura siguiente muestra la función de rango partido en un regulador de tres puntos:



Formador de impulsos del regulador paso a paso

Resumen

El formador de impulsos sólo está disponible para reguladores paso a paso (FM 355 S, FM 355-2 S y FM 455). Existen las siguientes posibilidades de ajuste.

Formador de impulsos (sólo regulador S)

- Tiempo de posicionamiento del motor
- Duración mínima del impulso
- Duración mínima de la pausa

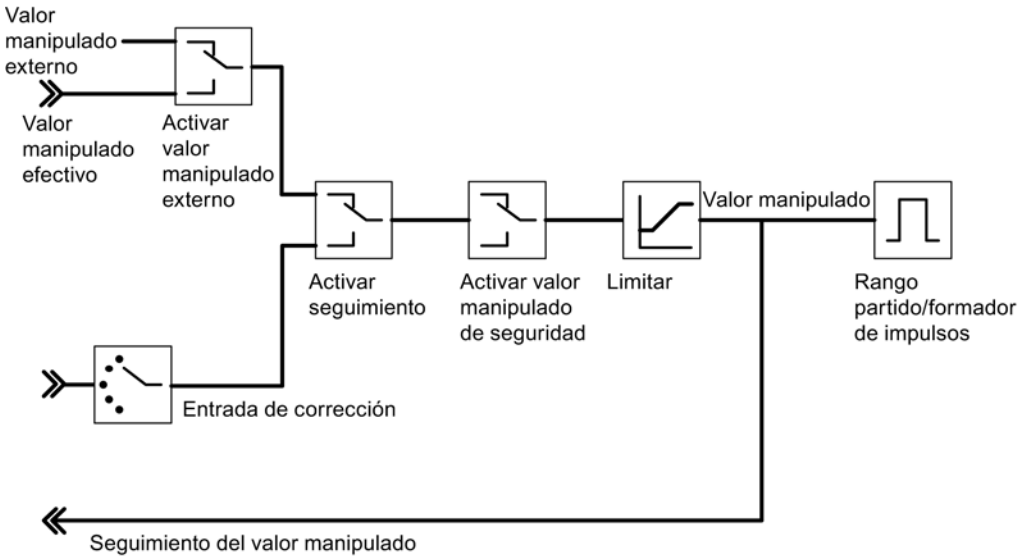
Seleccionar la señal de salida y el tipo de señal

Salida analógica

En la salida analógica se puede seleccionar para cada canal qué señal debe emitirse. La mayoría de las veces es la variable manipulada A de un regulador. Sin embargo, también se puede seleccionar la variable manipulada B de un regulador o un valor de entrada analógica. Este último se puede utilizar para linealizar un valor analógico. De esta forma se puede linealizar p. ej. la señal proporcionada por un termopar y convertirla en 0 a 10 V.

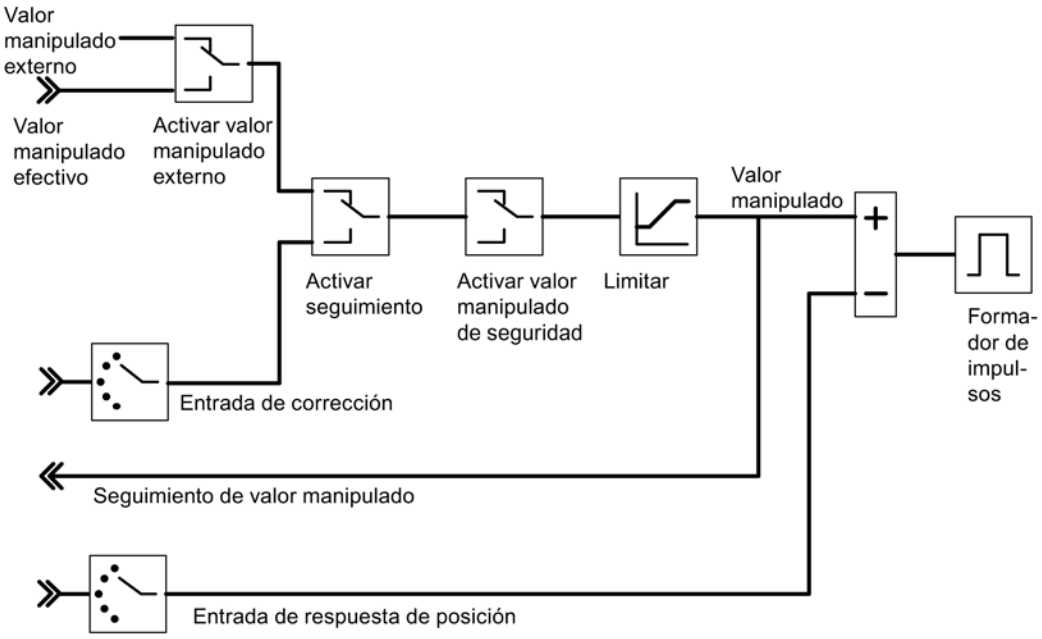
Salida del regulador de impulso

La figura siguiente muestra la salida del regulador S (modo de operación Regulador de impulsos)

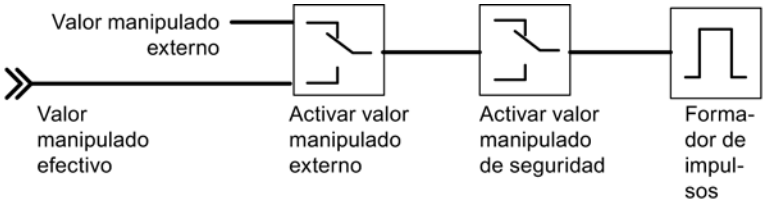


Salida del regulador paso a paso

La figura siguiente muestra la salida del regulador S (modo de operación Regulador paso a paso con realimentación de posición):



La figura siguiente muestra la salida del regulador S (modo de operación Regulador paso a paso sin realimentación de posición):



En el regulador paso a paso sin realimentación de posición analógica, la variable manipulada externa y la variable manipulada de seguridad actúan de la siguiente manera:

Si se asigna un valor entre el 40,0% y el 60,0%, no se activa ninguna salida binaria y el actuador permanece sin cambios.

Si se establece un valor > 60,0%, se devuelve "Señal de posicionamiento alta" hasta que se activa la realimentación "Actuador en tope superior".

Si se establece un valor < 40,0%, se emite "Señal de posicionamiento baja" hasta que la realimentación accede a "Actuador en tope inferior".

11.3.5.5 Uso del FM 355

Configuración del FM 355

Configuración hardware

Procedimiento

Para poder configurar el hardware debe haber creado previamente un proyecto donde pueda guardar la parametrización. A continuación se explican únicamente los pasos más importantes.

1. Inicie el TIA Portal y abra la configuración de dispositivos.
2. Seleccione un rack.
3. Seleccione un módulo de regulación del catálogo de módulos.
4. Arrastre el módulo de regulación hasta la posición correspondiente del rack.
5. Anote la dirección de entrada del módulo a partir de la tabla de configuración, p. ej. 272.
   El valor leído se muestra en formato decimal.

Parametrización del FM 355

Parámetros del módulo

Procedimiento

Al parametrizar se configuran los parámetros del módulo.

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros". En la ventana de inspección, seleccione la ficha "Propiedades".
   Resultado: Las propiedades se muestran en la ventana de inspección.
2. Parametrice los parámetros básicos del módulo y los parámetros restantes.
3. Parametrice los otros parámetros de bloque en la vista de configuración.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".
5. Compile el proyecto con "Edición > Compilar".
6. Cargue los datos de parametrización a la CPU en estado STOP mediante "Online > Cargar en dispositivo..."
   Resultado: Los datos se encuentran en la memoria de la CPU y se transfieren directamente desde allí hasta el módulo.
7. Efectúe un rearranque de la CPU.

Parámetros de la entrada analógica

Evaluación de la entrada analógica

Procedimiento

Para seleccionar el tipo de sensor, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue por la vista de edición hasta "Entrada analógica > Entrada [nº] > Evaluación de la entrada analógica".
3. Defina los siguientes parámetros:
   - Resolución entrada analógica
   - Tipo de sensor entrada analógica
     La opción "Vigilancia de rotura de hilo" se habilita para diversos tipos de sensor. La opción se puede activar según se requiera.
   - Temperatura de referencia
4. Guarde los parámetros introducidos con "Proyecto > Guardar".

Funciones de filtro

Procedimiento

Para establecer las funciones de filtro, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Funciones de filtro".
3. Establezca los siguientes parámetros:
   - Para filtrar en función de una constante de tiempo, active la opción "Filtro de señal". Si la opción "Filtro de señal" está seleccionada, se puede establecer la constante de tiempo del filtro.
   - Constante de tiempo del filtro
     Se puede activar solamente si la opción "Filtro de señal" se encuentra activada.
   - Para linealizar de la señal del encóder active la opción "Filtro de raíz cuadrada".
   - Para linealizar la curva característica de la señal del encóder, active la opción "Filtro de polígono"
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Filtro de señal

Filtro 1. Orden cuya rapidez de respuesta se establece mediante la constante de tiempo.

Filtro de raíz cuadrada

Sirve para linealizar señales de encóder, en las que el valor real existe como magnitud físca, que se encuentra en relación cuadrática con la magnitud de proceso medida.

Filtro de polígono

Sirve para linealizar de características de encóders.

Los 13 nodos disponibles pueden seleccionarse en:

- mA en la entrada de corriente
- mV en la entrada de tensión

Normalización

Procedimiento

Para convertir la señal de entrada en una unidad física, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Entrada [nº] > Normalizar".
3. Establezca los siguientes parámetros:
   - Valor superior
   - Valor inferior
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Normalización

Para convertir la señal de entrada en otra unidad física mediante interpolación lineal entre el valor inicial (valor inferior) y el valor final (valor superior).

Error de regulación

Realizar un regulador de valor fijo o en cascada

Regulador de valor fijo o en cascada

En las estructuras de regulación del regulador de valor fijo o en cascada, el valor real efectivo es idéntico al valor real A.

Seleccionar tipo de regulador

Para realizar un regulador de valor fijo o en cascada, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de configuración hasta "Entrada analógica > Canal [nº]".
3. Seleccione en el área "Tipo de regulador" la opción "Regulador de valor fijo o en cascada".
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para acondicionar la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Establezca, en las distintas áreas, los parámetros correspondientes:
   - Origen de consigna
     Seleccione el origen de la consigna.
   - Consigna de seguridad
     Ajuste aquí la consigna de seguridad. Asimismo, establezca la reacción en caso de un fallo de la CPU y el comportamiento en el arranque.
   - Rampa de consigna
     Establezca el tiempo de subida de rampa.
   - Límite de consigna
     Seleccione aquí el límite superior y el límite inferior de consigna.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Seleccionar el origen de la señal para el valor real A

Para seleccionar el origen de la señal para el valor real A, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione el origen correspondiente en el área "Variable manipulada". Están disponibles las siguientes opciones:
   - Cero
   - Entrada analógica acondicionada
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Realizar un regulador de tres componetes

Regulador de tres componentes

En la estructura del regulador de tres componentes, el valor real efectivo se forma a partir de la suma de los tres valores reales A, B y C y mediante la adición de un offset ajustable. Los valores reales B y C se pueden evaluar adicionalmente mediante factores.

Seleccionar el tipo de regulador

Para realizar un regulador de tres componentes, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº]".
3. Seleccione en el área "Tipo de regulador" la opción "Regulador de tres componentes".
4. Guarde los parámetros introducidos con "Proyecto > Guardar".

Acondicionamiento de la consigna

Para ajustar cómo debe acondicionarse la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione, en el área "Origen de consigna", de dónde debe proceder la consigna.
4. Introduzca, en el área "Consigna de seguridad", un valor que debe servir como valor de seguridad, en caso de un fallo de la CPU y para el arranque.
5. Seleccione el comportamiento del módulo de regulación, que debe ser válido como "Reacción a fallo de la CPU" y para el "Comportamiento en arranque". Para ello, se dispone de las siguientes opciones:
   - Consigna = última consigna válida: al producirse el evento, se asigna la última consigna válida a la consigna. La última consigna válida se mantiene.
   - Consigna = consigna de seguridad: al producirse el evento, se asigna la consigna de seguridad a la consigna. Así se puede realizar un funcionamiento de emergencia protegido.
6. Establezca, en el área "Rampa de consigna", el tiempo de subida que debe respetarse, en caso de una modificación de la consigna, con el fin de reducir una sobreoscilación.
7. Establezca en el área "Límite de consigna" tanto un límite superior como también uno inferior para la consigna.
8. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Seleccionar el origen de la señal para el valor real A, B y C

Para establecer los orígenes de la señal para los valores reales A, B y C, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione, en el área "Variable manipulada", el origen de la señal correspondiente para cada valor real. Puede seleccionar si no utiliza ninguna señal ("cero") como origen o si utiliza una de las entradas analógicas acondicionadas.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Introducir factores para los valores reales B y C así como offset

Para establecer los orígenes de la señal para los valores reales A, B y C, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. En el área "Sumar los valores reales", establezca el factor de ponderación necesario para los valores reales B y C.
4. Establezca el "Offset" necesario que debe sumarse a los valores reales.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Realizar una regulación de relación o de mezcla

Seleccionar el tipo de regulador

Para realizar un regulador de relación o de mezcla, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº]".
3. Seleccione en el área "Tipo de regulador" la opción "Regulador de relación o regulador de mezcla".
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para ajustar cómo debe acondicionarse la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione en el área "Origen de consigna", de dónde debe proceder la consigna.
4. Introduzca, en el área "Consigna de seguridad", un valor que debe servir como valor de seguridad, en caso de un fallo de la CPU y para el arranque.

5. Seleccione el comportamiento del módulo de regulación, que debe ser válido como "Reacción a fallo de la CPU" y para el "Comportamiento en arranque". Para ello, dispone de la selección siguiente:
   - Consigna = última consigna válida: al producirse el evento, se asigna la última consigna válida a la consigna. Por consiguiente, a última consigna válida se mantiene.
   - Consigna = consigna de seguridad: al aparecer el evento se asigna la consigna de seguridad a la consigna. Así se puede realizar un funcionamiento de emergencia seguro.
6. Establezca, en el área "Rampa de consigna", el tiempo de subida que debe respetarse, en caso de una modificación de la consigna, con el fin de reducir una sobreoscilación.
7. Establezca en el área "Límite de consigna" tanto un límite superior como también uno inferior para la consigna.
8. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Seleccionar el origen de la señal para los valores reales A y D

Para establecer los orígenes de la señal para los valores reales A y D, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione, en el área "Variable manipulada", el origen de la señal correspondiente para cada valor real. Puede especificar no utilizar ninguna señal ("cero") como origen o utilizar una de las entradas analógicas acondicionadas, para el valor real D puede seleccionar también el valor manipulado de uno de los reguladores.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Introducir multiplicador

Para establecer los orígenes de la señal para los valores reales A y D, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione el offset deseado en el área "Multiplicar".
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Para el regulador de relación, se utiliza el valor real A como magnitud regulada y el valor real D como magnitud de relación. La entrada de consigna funciona como factor de relación. Ésta se acondiciona para obtener una consigna efectiva mediante multiplicación por el valor real D y adición de un offset ajustable. Si el valor real D se desactiva, entonces sólo se suma el offset a la consigna.

Ajustar la generación de alarma

Procedimiento

Para ajustar la generación de alarma, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Entrada analógica > Canal [nº] > Error de regulación".
3. Seleccione las opciones necesarias en el área "Generar alarma":
   - Seleccione, en "Avisador de valor límite de", para qué señal se debe fijar la alarma.
   - Bajo "Alarma límite superior", establezca a partir de qué valor se activa la alarma al alcanzarse el límite superior.
   - Bajo "Advertencia límite superior", establezca a partir de que valor se activa una advertencia, cuando se produce una aproximación al límite superior.
   - Bajo "Advertencia límite inferior", establezca a partir de qué valor activa una advertencia, cuando se produce una aproximación al límite inferior.
   - Bajo "Alarma límite inferior", establezca a partir de qué valor se activa la alarma al alcanzarse el límite inferior.
   - Para evitar que centelleen los indicadores de vigilancia, se puede parametrizar una histéresis bajo "Histéresis".
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Algoritmo de regulación

Realizar una regulación P

Procedimiento

Para realizar una regulación P, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Tipo", la opción "Regulador PID".
4. Seleccione en el área "General", bajo "Modo de operación", la opción "P".
   Todas las opciones que son parametrizables para una regulación P se activan; todas las opciones restantes se desactivan.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Realizar una regulación PI

Procedimiento

Para realizar una regulación PI, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Tipo", la opción "Regulador PID".
4. Seleccione en el área "General", bajo "Modo de operación", la opción "PI".
   Todas las opciones que son parametrizables para una regulación PI se activan; todas las opciones restantes se desactivan.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Crear regulación I

Procedimiento

Para realizar una regulación I, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Tipo", la opción "Regulador PID".
4. Seleccione en el área "General", bajo "Modo de operación", la opción "I".
   Todas las opciones que son parametrizables para una regulación I se activan; todas las opciones restantes se desactivan.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Crear una regulación PID

Procedimiento

Para realizar una regulación PID, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Tipo", la opción "Regulador PID".
4. Seleccione en el área "General", bajo "Modo de operación", la opción "PID".
   Todas las opciones que son parametrizables para una regulación PID se activan; todas las opciones restantes se desactivan.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Crear una regulación PD

Procedimiento

Para realizar una regulación PD, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Tipo", la opción "Regulador PID".
4. Seleccione en el área "General", bajo "Modo de operación", la opción "PD".
   Todas las opciones que son parametrizables para una regulación PD se activan; todas las opciones restantes se desactivan.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Realizar una regulación de temperatura por lógica difusa

Procedimiento

Para llevar a cabo una regulación de temperatura por lógica difusa, proceda del modo siguiente:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Tipo", la opción "Módulo de regulación de temperatura por lógica difusa".
   Todas las opciones que son parametrizables para una regulación de temperatura por lógica difusa se activan; todas las opciones restantes se desactivan.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Suprimir nivel de ruido del error de regulación

Procedimiento

Para suprimir el nivel de ruido del error de regulación, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Establezca, en el área "General", el ajuste de "Ancho de zona muerta".
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Control anticipativo

Procedimiento

Para aplicar adicionalmente una perturbación, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. En el área "General", en la lista "Fuente de perturbación", seleccione la fuente de la perturbación.
   Seleccione la entrada "Cero" para no aplicar ninguna perturbación adicionalmente.
4. Introduzca, bajo "Factor de perturbación", un valor, con el que desea ponderar la perturbación.
   Un valor < 0 debilita la perturbación, un valor > 0 aumenta la perturbación. Con un valor de 1, la perturbación no cambia.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Poner la acción P por rama de realimentación

Procedimiento

Para realizar un regulador de valor fijo o en cascada, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en la "Puesta en servicio".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione en el área "Acción P" la opción "Por rama de realimentación".
   Si la opción está desactivada, la acción P está puesta en rama de realimentación.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Debilitar la acción D en caso de cambios de consigna

Seleccionar el tipo de regulador

Para realizar un regulador de valor fijo o en cascada, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Seleccione, en el área "Acción D", con la opción "Acción D", el origen para la acción D.
4. Establezca el factor para la acción D en el área "Factor derivativo".
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Invertir el efecto del regulador

Principio

La inversión del regulador de la asignación "error de regulación creciente = magnitud manipulada creciente" a "error de regulación creciente = magnitud manipulada decreciente", se logra mediante el ajuste de una ganancia proporcional negativa (GAIN).

Procedimiento

Para invertir el efecto del regulador, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Algoritmo de regulación".
3. Establezca, en el área "General", una ganancia proporcional con signo invertido.
   Si se indica una ganancia proporcional positiva y el efecto del regulador debe invertirse, establezca una ganancia proporcional negativa.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Salida del regulador

Crear un regulador continuo

Un regulador continuo sólo puede realizarse con un módulo analógico.

Procedimiento

1. Conectar una variable manipulada externo (Página 10268)
2. Ajustar la variable manipulada de seguridad (Página 10269)
3. Conectar el modo de seguimiento (Página 10268)
4. Limitar la variable manipulada (Página 10269)
5. Controlar dos actuadores - Rango partido (Página 10270)
6. Seleccionar la señal de salida y el tipo de señal (Página 10271)

Regulador paso a paso sin realimentación de posición

Un regulador paso a paso sin realimentación de posición sólo puede realizarse con un módulo digital.

Procedimiento

1. Seleccionar regulador paso a paso sin realimentación de posición
2. Conectar una variable manipulada externo (Página 10268)

3. Ajustar la variable manipulada de seguridad (Página 10269)
4. Formador de impulsos del regulador paso a paso (Página 10270)

Crear un regulador paso a paso con realimentación de posición

Un regulador paso a paso con realimentación de posición sólo puede realizarsese con un módulo digital.

Procedimiento

1. Seleccionar regulador paso a paso con realimentación de posición
2. Conectar una variable manipulada externo (Página 10268)
3. Conectar el modo de seguimiento (Página 10268)
4. Ajustar la variable manipulada de seguridad (Página 10269)
5. Limitar la variable manipulada (Página 10269)
6. Ajustar la realimentación de posición (Página 10268)
7. Formador de impulsos del regulador paso a paso (Página 10270)

Crear regulador de impulsos

Un regulador de impulsos sólo puede realizarse con un módulo digital.

Procedimiento

1. Seleccionar salida del regulador de impulsos
2. Conectar una variable manipulada externo (Página 10268)
3. Conectar el modo de seguimiento (Página 10268)
4. Ajustar la variable manipulada de seguridad (Página 10269)
5. Limitar la variable manipulada (Página 10269)
6. Preparar señal analógica para la conversión en señal digital
7. Convertir la señal analógica en señal digital (Página 10271)

Parametrizar las funciones de la salida del regulaldor

Conectar una variable manipulada externo

Procedimiento

Para conectar la variable manipulada externa, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Salida del regulador".
3. En el área "Variable manipulada externa", seleccione la opción con la cual debe activarse la variable manipulada externa. Para ello, dispone de las siguientes opciones:
   – Activación del bloque de función.
   – Activación del bloque de función o de una de las entradas digitales.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Conectar el modo de seguimiento

Procedimiento

Para realizar un regulador de valor fijo o en cascada, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Salida del regulador".
3. En el área "Modo de seguimiento", seleccione la opción mediante la cual se debe activar la variable manipulada externa. Para ello, dispone de las siguientes opciones:
   – Activación del bloque de función.
   – Activación del bloque de función o de una de las entradas digitales.
4. En el área "Modo de seguimiento", seleccione el origen de la señal para el modo de seguimiento. Para ello, dispone de las siguientes opciones:
   – Cero, en este caso, no se utilizar ningún origen.
   – Una de las entradas analógicas.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Ajustar la realimentación de posición

Procedimiento

Para ajustar la realimentación de posición, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Salida digital > Salida [nº]".

3. En el área "Entrada de realimentación de posición", seleccione la opción "Origen", para seleccionar la señal para la realimentación de posición.
   – Cero: ninguna realimentación de posición.
   – Una de las entradas analógicas acondicionadas.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Ajustar la variable manipulada de seguridad

Procedimiento

Para ajustar la variable manipulada de seguridad, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Salida del regulador".
3. En el área "Variable manipulada de seguridad", establezca el valor que debe utilizarse como variable manipulada de seguridad.
4. Bajo "Activación", seleccione la opción mediante la cual se debe activar la variable manipulada externa. Para ello, dispone de las siguientes opciones:
   – Activación del bloque de función.
   – Activación del bloque de función o de una de las entradas digitales.
5. Bajo "Reacción al arrancar el módulo", seleccione la opción que debe llevarse a cabo en el arranque del módulo. Dispone de las siguientes opciones para su elección:
   – Lazo cerrado: en el arranque, el módulo pasa inmediatamente al modo de regulación.
   – Variable manipulada de seguridad: primero se regula a la variable manipulada de seguridad.
6. Bajo "Reacción si falla el transductor valor real A", seleccione la opción que debe llevarse a cabo, en caso de fallo del transductor de medida del valor real A.
7. Bajo "Reacción a fallo transductor entrada", seleccione la opción que debe llevarse a cabo, en caso de fallo del transductor de medida de una entrada.
8. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Limitar la variable manipulada

Procedimiento

Para limitar la variable manipulada, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Salida del regulador".

3. En el área "Límite de señal de salida", establezca los siguientes valores:
   - Valor superior: representa el valor límite superior para la señal de salida.
   - Valor inferior: representa el valor límite inferior para la señal de salida.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Controlar dos actuadores - Rango partido

Procedimiento

Para activar dos actuadores, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Regulador > Canal [nº] > Salida del regulador".
3. En el área "Rango partido", seleccione la opción "Activar", para activar dos elementos finales de control.
4. Establezca los siguientes valores tanto para las señales de entrada y salida A como también para las señales de entrada y salida B:
   - Señal de entrada inferior.
   - Señal de entrada superior.
   - Señal de salida inferior.
   - Señal de salida superior.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Formador de impulsos del regulador paso a paso

Procedimiento

Para parametrizar el formador de impulsos, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Salida digital > Salida [nº]".
3. En el área "formador de impulsos", establezca los siguientes parámetros:
   - Tiempo de posicionamiento del motor.
   - Duración mínima de impulso.
   - Duración mínima de pausa.
4. Seleccione la opción adecuada de la que se debe tomar el valor para el valor superior y el inferior:
   - Del bloque de función.
   - Del bloque de función o de una de las entradas digitales.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Convertir la señal analógica en señal digital

Procedimiento

Para convertir una señal analógica en una señal digital, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Salida digital > Canal [nº]".
3. En el área "formador de impulsos", establezca los siguientes parámetros:
   – Duración mínima de impulso.
   – Duración mínima de pausa.
4. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Seleccionar la señal de salida y el tipo de señal

Procedimiento

Para seleccionar la señal de salida y el tipo de señal, proceda del siguiente modo:

1. Haga doble clic en la vista de dispositivos, bajo el módulo del "Ajuste de parámetros".
2. Navegue en la vista de edición hasta "Salida analógica > Salida [nº]".
3. Bajo la opción "Origen", seleccione qué señal se debe utilizar como señal de salida. Dispone de las siguientes opciones:
   – Cero: ningún origen para la señal de salida.
   – Una de las entradas analógicas.
   – Uno de los valores manipulados para el regulador A o B.
4. Bajo la opción "Tipo de señal", seleccione el tipo en el que debe emitirse la señal de salida.
5. Guarde los parámetros introducidos mediante "Proyecto > Guardar".

Programación del FM 355

Puesta en servicio del FM 355

Crear y configurar DBs de instancia

Procedimiento

Para satisfacer los requisitos para la puesta en marcha, proceda de la manera siguiente:

1. Cree un programa de usuario que efectúe cíclicamente una llamada a la instrucción PID_FM en un OB de alarma cíclica.
2. En el programa del usuario, ponga READ_VAR cíclicamente a TRUE.

3. Genere para cada canal del regulador un DB de instancia individual de la instrucciónPID_FM.
4. Para cada DB de instancia, introduzca la dirección de módulo en el parámetro PID_FM.MOD_ADDR.
   Utilice como dirección de módulo del FM 355 la dirección inicial especificada al configurar el hardware.
5. Introduzca en el parámetro PID_FM.CHANNEL el número del canal de regulación (1 ... 4).
6. Si utiliza la instrucción PID_PAR, introduzca en el parámetro PID_PAR.CHANNEL el número del canal del regulador.
7. Compruebe que esté activado el "Tipo de sensor Entrada analógica" del canal en cuestión. Si no se procesa la entrada analógica, todos los parámetros PID se ajustan a cero, y no se calcula ninguna variable manipulada.

Nota

DBs multiinstancia

Si utiliza DBs multiinstancia en su proyecto, los FM se deben poner en servicio a través de una tabla de observación.

Consulte también

Descripción de PID_FM (Página 7484)

Descripción de PID_PAR (Página 7512)

Puesta en servicio del regulador

Introducción

Los parámetros PID se pueden optimizar del modo siguiente:

- Regulador PID/PI (regulador paso a paso, regulador continuo, regulador de impulsos): Manual
- Regulador de temperatura por lógica difusa: Ciclo continuo
- Regulador paso a paso: Automáticamente con la instrucción TUN_ES
- Regulador continuo: Automáticamente con la instrucción TUN_EC
- Regulador de impulsos: Automáticamente con la instrucción TUN_EC

Procedimiento

Para optimizar un regulador, proceda del siguiente modo:

1. Coloque la CPU en el estado RUN.
2. Si utiliza un regulador paso a paso, mida el tiempo de posicionamiento del motor.

3. Optimice los parámetros PID con alguna de las posibilidades citadas anteriormente.
4. Observe el lazo de regulación con el registrador de curvas.

Consulte también

Medir el tiempo de posicionamiento del motor (Página 10280)

Descripción de TUN_ES (Página 7149)

Descripción de TUN_EC (Página 7129)

Optimizar los parámetros PID manualmente

Requisitos

En la parametrización del algoritmo de regulación ha elegido el "Regulador PID".

Procedimiento

Para ajustar manualmente un regulador PI/PID, proceda del modo siguiente:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. Introduzca los valores correspondientes para la acción P, la acción I, la acción D y el retardo.
4. En el grupo "Optimización", haga clic en el símbolo "Enviar parámetros PID al FM".
5. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna".
6. Introduzca una nueva consigna.
7. Desactive en caso necesario la casilla de verificación "Modo manual" y, en el grupo "Valores actuales", haga clic en el símbolo .
   El regulador trabaja con los nuevos parámetros y regula según la nueva consigna.
8. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
9. Repita los pasos del 3 hasta el 8, hasta que esté satisfecho/a con el resultado de la regulación.

Identificación del sistema por lógica difusa

Requisitos del proceso para un regulador de temperatura por lógica difusa

Requisitos

Para conseguir una regulación óptima con la ayuda del regulador de temperatura por lógica difusa, el proceso debe satisfacer los requisitos siguientes:

- En caso de calefacciones de baños, el líquido calefactor debe estar bien mezclado.
- En caso de regulaciones tipo baño en baño, ambos líquidos deben estar bien agitados. Simultáneamente, se debe proporcionar una buena transferencia de calor entre todos los medios caloportadores. Si se emplean materiales que son malos caloportadores, para conseguir un buen transporte del calor se deben utilizar superficies de transferencia de gran tamaño.
- En caso de regulación de la temperatura del aire ambiental, se debe proporcionar una buena mezcla (p. ej., mediante ventiladores).
- La ganancia del sistema regulado no debe superar un factor 3.
- El tiempo de retardo debe equivaler como máximo al 3% del tiempo de compensación.
- Durante el periodo de muestreo del regulador, y con la salida de la variable manipulada en el valor máximo, la temperatura que se desea regular debe variar como máximo un 1‰ de la temperatura máxima preajustada.

Clasificación de los sistemas regulados

Un sistema regulado, o bien un proceso que se desea regular, se caracteriza por parámetros como, p. ej., la potencia de calefacción, la masa que se debe calentar o la capacidad térmica del medio que se desea calentar. En lo que se refiere al regulador por lógica difusa, se efectúa una distinción entre sistemas de temperatura regulada "críticos" y "no críticos" del modo siguiente: La regulación es tanto más crítica

- cuanto mayor es la potencia de calefacción,
- cuanto mayor es la capacidad térmica de la calefacción,
- cuanto menor es la capacidad térmica del medio que se desea calentar,
- cuanto mayor es la resistencia a la transmisión de calor,
- cuanto menor es la superficie de transferencia de calor.

Tras el seguimiento automático de un salto en la variable manipulada por parte del sistema regulado, éste reacciona con una respuesta a un escalón. La clasificación de los sistemas regulados también se puede llevar a cabo en función de esta respuesta a un escalón: La regulación es tanto más crítica cuanto mayor es el cociente t_u / t_g y cuanto mayor es la ganancia del sistema regulado. Si t_u / t_g < 1/10, el sistema regulado no es crítico.

Desarrollo de la identificación del sistema por lógica difusa

Desarrollo de la optimización de parámetros con un regulador de temperatura por lógica difusa

La optimización de parámetros se basa en un regulador por lógica difusa autooptimizante.

La identificación del sistema regulado comienza con una fase de observación, durante la cual la potencia de calefacción emitida es nula. Esta fase de observación dura lo siguiente:

- Fase de observación de un regulador continuo: aprox. 1 minuto
- Fase de observación de un regulador paso a paso: aprox. 1 minuto + 1/2 x tiempo de desplazamiento del actuador

Este tiempo se emplea para poder detectar tendencias de temperatura en el medio calefactor. A continuación se emite el 100% de la potencia máxima de calefacción; esto se aprecia en el parámetro de salida LMN del DB de instancia de la instrucción PID_FM.

El área del primer 4% de incremento de la temperatura del rango de consigna se utiliza para la identificación, ya que tanto en el 1% como en el 4% del incremento de temperatura se obtiene información sobre el comportamiento del proceso a partir del tiempo transcurrido.

La identificación concluye cuando la potencia de calefacción pasa a ser inferior al 100%. Este comportamiento se puede utilizar, p. ej., para desconectar la identificación a través del programa de usuario. Tras la fase de identificación, el regulador sigue funcionando con los parámetros obtenidos.

Si la divergencia provocada por el salto de consigna no se compensa y la potencia de calefacción permanece a cero de manera continua, significa que la identificación ha resultado fallida, es decir, el regulador no puede regular el sistema regulado al que está conectado.

Comportamiento del regulador para distintos sistemas regulados

Si el sistema regulado es "no crítico", no hay problemas en la identificación ni en la regulación.

La identificación de un sistema regulado "demasiado crítico" queda cancelada. Si un sistema regulado se identifica como "crítico", su regulación se efectúa con mucha "precaución", es decir, despacio.

Efectuar la identificación de sistema por lógica difusa.

Requisitos

En la parametrización del algoritmo de regulación ha elegido el "Regulador por lógica difusa".

Si es posible, asegúrese de que antes de la identificación el sistema regulado se haya estabilizado (ningún proceso activo de calefacción o refrigeración) o varíe con lentitud y de forma monótona. Se puede utilizar como criterio al respecto que el comportamiento de la variación de temperatura durante un minuto se pueda aproximar por una recta. En procesos rápidos, este requisito resulta especialmente actual.

Dado que el valor emitido para la variable manipulada es nulo durante aprox. un minuto al principio de la identificación del regulador, la temperatura que se desea regular debe ser cercana a la temperatura ambiente.

Iniciar la identificación del sistema por lógica difusa

Para optimizar un regulador por lógica difusa, proceda del modo siguiente:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. En el grupo "Optimización" introduzca una nueva consigna (SP_OP).
   La nueva consigna debe cumplir las condiciones siguientes:
   – Salto de consigna > 5%
   – Nueva consigna > valor real + límite superior de la consigna x 0,12
4. En el grupo "Optimización", haga clic en el símbolo "Identificación del sistema por lógica difusa".

Resultado

La nueva consigna (SP_OP), SP_OP_ON = TRUE y FUZID_ON = TRUE se envían al FM. La identificación del sistema por lógica difusa comienza. El parámetro IDSTATUS de la instrucción CH_DIAG proporciona información sobre el estado de la identificación. En el cuadro de visualización de estado puede ver si hay una identificación del sistema por lógica difusa en estado activo o en ejecución.

Desactivar la identificación del sistema por lógica difusa

Mientras el bit FUZID_ON = TRUE, cada vez que se produzca un salto de consigna suficientemente grande se iniciará la identificación siguiente.

Para desconectar el modo de optimización, proceda del modo siguiente:

1. Inserte FUZID_ON en una tabla de observación.
2. Defina FUZID_ON = FALSE.

Nota

La desconexión del modo de optimización FUZID_ON = FALSE antes de terminar la identificación no provoca que esta se detenga. Una vez iniciada la identificación, esta sigue adelante pase lo que pase, excepto si se produce un salto de consigna negativo.

Repetir la identificación del sistema por lógica difusa

Cada vez que se produce un salto de consigna ≥12% se efectúa una nueva identificación, a no ser que esta se desconecte de nuevo con FUZID_ON = FALSE.

También existe la posibilidad de iniciar de nuevo la identificación disminuyendo y, a continuación, aumentando de nuevo la consigna en un valor >12%.

Cancelar la identificación del sistema por lógica difusa

La identificación se puede cancelar en los casos siguientes:

- Por parte del regulador, si se identifica un sistema regulado "crítico". Tras la cancelación, el regulador se encuentra en estado de error. Este estado se manifiesta en que el regulador desactiva la variable manipulada de forma permanente. Este estado no varía aunque se desconecte la optimización. El inicio de una nueva identificación anula el estado de error. Se puede obtener información sobre el estado de la identificación a través del parámetro IDSTATUS del FB CH_DIAG. Más detalles al respecto al final de este capítulo, en la sección "Información sobre el estado del regulador".
- Por parte del operador, generando un salto de consigna negativo, siempre y cuando la consigna rebase por defecto la condición 2 (véase más arriba, "¿Cómo iniciar la identificación?").

Para cancelar la optimización, proceda del modo siguiente:

1. Introduzca una nueva consigna que sea inferior al valor real actual y que cumpla las mismas condiciones que al principio de la identificación del sistema por lógica difusa.
2. En el grupo "Optimización", haga clic en el símbolo "Identificación del sistema por lógica difusa".

Consulte también

Parámetro IDSTATUS (Página 7510)

Cargar los parámetros optimizados en el dispositivo

Introducción

Para impedir que los parámetros del FM se sobrescriban con los datos del sistema en el siguiente arranque de la CPU, se puede proceder del modo siguiente:

- Tras cada arranque del FM , cargue en este con LOAD_PAR=TRUE los parámetros PID optimizados del DB de instancia del PID_FM.
- Cargue los parámetros PID optimizados en el SDB.

Nota

Los juegos de parámetros PI, PID y SAVE no se pueden guardar en el SDB.

Procedimiento

Para cargar los parámetros en el SDB, proceda del siguiente modo:

1. Haga clic en el símbolo "Transfiera datos online al proyecto offline".
   Los parámetros PID se guardan en el ajuste de parámetros del FM.
2. Pare la CPU.
3. Haga clic en el símbolo "Cargar en dispositivo".

Trabajar con el registrador de curvas

Observar

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Puesta en servicio".
2. Navegue en la vista de configuración hasta "Registrador de curvas".
3. Seleccione el DB de instancia que desee observar. Tras la selección se muestra el canal al que está asignado el DB de instancia.
4. Defina el tiempo de actualización con el que debe registrarse el registrador de curvas.
5. Haga clic en "Medición activada" para iniciar la medición.
6. Haga clic en "Medición desactivada" para finalizar la medición.

Procedimiento

Para adaptar la escala, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Puesta en servicio".
2. Navegue en la vista de configuración hasta "Registrador de curvas".
3. Para modificar el eje de tiempo:
   - Haga clic en el botón de flecha junto a la fecha, para abrir un calendario. Seleccione una fecha en el calendario. Para bloquear la selección, haga clic en el candado azul. Esta posibilidad existe tanto para la fecha inicial como para la fecha final.
   - Haga clic con el puntero del ratón en la escala de tiempo. El puntero del ratón se representa como flecha a la izquierda/derecha. Mantenga el puntero del ratón presionado y arrástrelo hacia la izquierda o derecha.
     Si el tiempo inicial o final está bloqueado, se modifica la escala del tiempo.
     Si no está bloqueado el tiempo inicial ni el final, se modifica el período indicado; la escala se mantiene igual.
4. Para modificar el eje de tiempo:
   - Haga clic en el campo de entrada para el valor superior o el valor inferior mostrado. Introduzca ahí el valor deseado. Para bloquear la selección, haga clic en el candado azul. Esta posibilidad existe tanto para el valor superior como también para el inferior.
   - Haga clic con el puntero del ratón en la escala de valores. El puntero del ratón se representa como flecha arriba/abajo. Mantenga el puntero del ratón presionado y arrástrelo hacia arriba o abajo.
     Si el valor superior o inferior está bloqueado, se modifica la escala del eje de valores.
     Si no está bloqueado el valor superior ni el valor inferior, se modifica el rango indicado; la escala se mantiene igual.

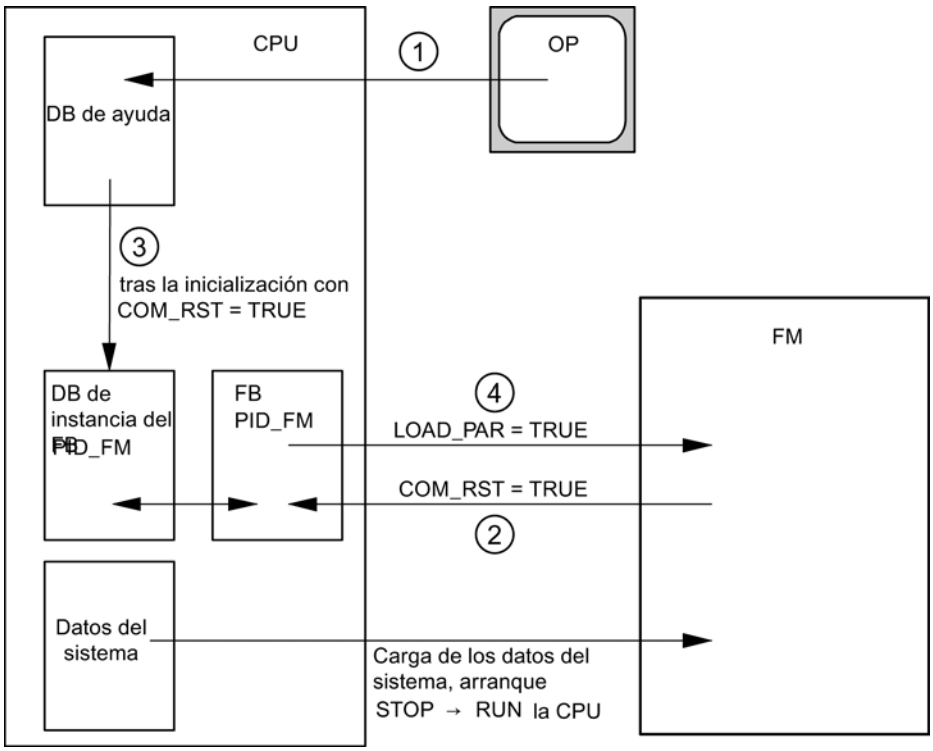
Modificar los parámetros a través de la HMI

Procedimiento

Si desea modificar los parámetros del regulador de la instrucción PID_FM mediante el manejo a través de la HMI, proceda del modo siguiente:

1. Desde la HMI, escriba en la CPU (véase ①) en un DB auxiliar los parámetros que se deben modificar.
2. Ponga PID_FM.COM_RST a TRUE.
   La instrucción ejecuta una inicialización. Los parámetros se sincronizan entre el FM y la CPU (véase ②).
3. Transfiera los parámetros que se deben modificar desde el DB auxiliar hasta el DB de instancia de la instrucción PID_FM (véase ③).
4. Ponga PID_FM.LOAD_PAR a TRUE.
   Los parámetros modificados se transfieren al FM (véase ④).

Resulta necesario depositar los parámetros en un DB auxiliar porque, tras el arranque de la CPU, PID_FM lee con COM_RST = TRUE aquellos parámetros del módulo que la CPU ha transferido previamente desde los datos del sistema al FM.



Consulte también

Funcionamiento de PID_FM (Página 7485)

Medir el tiempo de posicionamiento del motor

Introducción

Para conseguir un buen resultado de la regulación, el tiempo de posicionamiento del motor debe ser lo más preciso posible. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento.

Si utiliza actuadores con realimentación de posición o con señales de tope, es posible medir el tiempo de posicionamiento del motor durante la puesta en servicio. Si no se dispone de una realimentación de posición ni de señales de tope, no es posible medir el tiempo de posicionamiento del motor.

Requisitos

- El valor parametrizado del tiempo de posicionamiento del motor debe equivaler como mínimo a la mitad del tiempo de posicionamiento real. El actuador se desplaza en un sentido durante el doble del tiempo de posicionamiento como máximo.
- En los actuadores con realimentación de posición analógica, la realimentación de posición debe estar interconectada.
- En los actuadores con señales de tope, las señales de tope deben estar interconectadas.

Actuadores con realimentación de posición analógica

Para medir el tiempo de posicionamiento del motor con realimentación de posición, proceda del siguiente modo:

1. Elija el DB instancia para el que se debe medir el tiempo de posicionamiento del motor.
2. Active la casilla de verificación "Utilizar realimentación de posición".
3. En el campo de entrada "Posición de destino" indique hacia dónde desea mover el elemento final de control.
   La realimentación de posición actual (posición de inicio) se muestra. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se desplaza hasta la posición de destino desde la posición de inicio. La medición del tiempo se inicia de inmediato y finaliza en cuanto el elemento final de control ha alcanzado la posición de destino. El tiempo de posicionamiento del motor se calcula según la fórmula:

Tiempo de posicionamiento del motor = (Límite superior del valor de salida - Límite inferior del valor de salida) x Tiempo de medición /VALOR ABSOLUTO(Posición de destino - Posición de inicio).

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se muestra en el campo "Tiempo de actuación medido". Una vez finalizada la medición del tiempo de posicionamiento se conecta el modo manual de las

señales de posicionamiento (LMNSOPON = TRUE). El elemento final de control deja de moverse.

Actuadores con señales de tope

Para medir el tiempo de posicionamiento del motor con señales de tope, proceda del siguiente modo:

1. Elija el DB instancia para el que se debe medir el tiempo de posicionamiento del motor.
2. Active la casilla de verificación "Utilizar señales de tope del actuador".
3. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se mueve primero hasta el tope inferior, después, hasta el tope superior y, de nuevo, hasta el tope inferior. La medición del tiempo se inicia cuando el actuador ha alcanzado el tope inferior y finaliza cuando el actuador alcanza por segunda vez el tope inferior. El tiempo medido dividido entre dos da lugar al tiempo de posicionamiento del motor.

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se muestra en el campo "Tiempo de actuación medido". Una vez finalizada la medición del tiempo de posicionamiento se conecta el modo manual de las señales de posicionamiento (LMNSOPON = TRUE). El elemento final de control deja de moverse.

Cargar el tiempo de ajuste medido

1. Para guardar en el proyecto el tiempo de posicionamiento del motor medido, haga clic en el símbolo "Transfiere datos online al proyecto offline".
   El tiempo de ajuste medido se guarda en el ajuste de parámetros del FM.
2. Para que los cambios surtan efecto en el arranque siguiente de la CPU, haga clic en el icono "Cargar en dispositivo".

Cancelar medición del tiempo de posicionamiento

Si cancela la medición del tiempo de posicionamiento, el modo manual de las señales de posicionamiento se conecta (LMNSOPON = TRUE) y el modo manual con especificación de la variable manipulada se desconecta (LMNOP_ON = FALSE). El elemento final de control deja de moverse.

Consulte también

Parámetros de entrada/salida de PID_FM (Página 7491)

FM 355: Diagnóstico a través de indicadores LED

Indicación de errores mediante el LED de error de grupo

Introducción

Si el LED rojo de error de grupo se enciende, significa que se ha producido un error en el módulo (error interno) o en las conexiones (error externo).

Si el LED amarillo parpadea, el firmware se borra. Este estado sólo puede aparecer si existe un defecto en el hardware o si el proceso de carga del firmware se cancela.

¿Qué errores se muestran?

Cuando se ilumina el LED de error de grupo se señalan los errores siguientes:

Tipo de error	Aviso de diagnóstico	Causa posible	Solución
Errores internos	Módulo defectuoso	Error de hardware	Cambie el módulo
	Vigilancia de tiempo (watch dog) activado	Error de hardware	Cambie el módulo
	Contenido EEPROM no válido	Fallo de la tensión de alimentación al parametrizar	Vuelva a parametrizar el módulo
Errores externos	Parámetro erróneo en el módulo	Se han transferido parámetros incorrectos al módulo	Vuelva a parametrizar el módulo
	Error en las entradas o las salidas analógicas	Error de hardware en la entrada analógica	Cambie el módulo
		Rotura de hilo en la entrada analógica	Solucione la rotura de hilo
		Rebase por defecto del rango de medición en la entrada analógica	Revise la señal de medición
		Rebase por exceso del rango de medición en la entrada analógica	Revise la señal de medición
		Rotura de hilo en la salida analógica	Solucione la rotura de hilo
		Cortocircuito en la salida analógica	Solucione el cortocircuito
	Falta tensión auxiliar externa	Fallo en la alimentación 24 V	Restablezca la alimentación de 24 V

Alarma de diagnóstico en caso de error

Todos los errores pueden disparar una alarma de diagnóstico si ésta se ha habilitado en la pantalla de parametrización correspondiente. En los registros de diagnóstico DS0 y DS1 puede ver qué error ha provocado que se encienda el LED. La asignación de los registros de diagnóstico DS0 y DS1 está descrita en el siguiente apartado.

Disparo de alarmas de diagnóstico

¿Qué es una alarma de diagnóstico?

Si se quiere que un programa de usuario responda a un error interno o externo, se puede parametrizar una alarma de diagnóstico que interrumpa el programa cíclico de la CPU y llame al OB de alarma de diagnóstico (OB 82).

¿Qué eventos pueden disparar una alarma de diagnóstico?

La siguiente lista muestra qué eventos pueden disparar una alarma de diagnóstico:

- Módulo no parametrizado o errores en la parametrización
- Módulo defectuoso
- Rotura de hilo en entradas analógicas (sólo 4 a 20 mA)
- Rebase por exceso y por defectos en entradas analógicas
- Rotura de carga y cortocircuito en salidas analógicas

Ajuste por defecto

La alarma de diagnóstico está bloqueada de forma predeterminada.

Habilitación de la alarma de diagnóstico

La pantalla "Parámetros básicos" permite bloquear o habilitar la alarma de diagnóstico para el módulo.

Respuestas a un evento que dispara una alarma

Cuando se produce un evento que puede disparar una alarma de diagnóstico ocurre lo siguiente:

- La información de diagnóstico se registra en los registros de diagnóstico DS0 y DS1 del módulo.
- El LED de error de grupo se enciende.
- Se llama al OB de alarma de diagnóstico (OB 82).
- El registro de diagnóstico DS0 se graba en la información de arranque del OB de alarma de diagnóstico.
- Si no hay ningún defecto de hardware, el módulo sigue regulando.

La CPU pasa a modo STOP si no se ha programado el OB 82.

Registro de diagnóstico DS0 y DS1

La información que indica qué evento ha provocado una alarma de diagnóstico se guarda en los registros de diagnóstico DS0 y DS1. El registro de diagnóstico DS0 consta de cuatro bytes; DS1 consta de 16 bytes, siendo los cuatro primeros idénticos a DS0.

Registros de diagnóstico DS0 y DS1

Leer el registro desde el módulo

El registro de diagnóstico DS0 se transfiere automáticamente a la información de arranque en la llamada del OB de diagnóstico. Allí, estos cuatro bytes se guardan en los datos locales (bytes 8-11) del OB 82.

El registro de diagnóstico DS1 se puede leer (y en consecuencia también el contenido de DS0) desde el módulo con la instrucción "CH_DIAG". Esta lectura sólo tiene sentido si en el DS0 se notifica un error en un canal.

La instrucción "CH_DIAG" debe llamarse en el mismo OB que el FB "PID_FM". Esto se consigue con la medida siguiente: Ajuste un bit mientras el OB 82 se procesa. Consulte este bit en el OB, en el que también se llama el FB "PID_FM" y llame la instrucción "CH_DIAG" en el bit definido.

¿Cómo aparecen los textos de diagnóstico en el búfer de diagnóstico?

Si se desea registrar el aviso de diagnóstico en el búfer de diagnóstico, se debe llamar la instrucción "WR_REC" "Introducir avisos específicos de usuario en el búfer de diagnóstico" dentro del programa de usuario. En cada caso, el número de evento del aviso de diagnóstico se especifica en el parámetro de entrada EVENTN. La alarma se introduce en el búfer de diagnóstico con x=1 como entrante y con x=0 como saliente. El búfer de diagnóstico contiene los textos de diagnóstico correspondientes en la columna de "Significado", así como el momento en que se produjo la entrada.

Asignación del registro de diagnóstico DS0 en la información de arranque

La tabla siguiente muestra la asignación del registro de diagnóstico DS0 en la información de arranque. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Tabla 11-7 Asignación del registro de diagnóstico DS0

Byte	Bit	Significado	Nota	Nº de evento
0	0	Avería en módulo	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Error interno	Se activa en todos los errores internos: • Vigilancia de tiempo activada • Contenido de la EEPROM no válido; el módulo arranca sin regular y espera a que la CPU realice una nueva parametrización • Error EPROM • Error ADC/DAC • Error de hardware en la entrada analógica	8:x:01

Byte	Bit	Significado	Nota	Nº de evento
	2	Error externo	Se activa en todos los errores externos: • Falta tensión auxiliar externa • Error de parametrización • Rotura de hilo en la entrada analógica (sólo en el margen 4 a 20 mA) • Rebase por defecto del rango de medición en la entrada analógica • Rebase del rango de medición en la entrada analógica • Rotura de hilo en la salida analógica • Cortocircuito en la salida analógica	8:x:02
	3	Fallo en un canal	Consultar el DS1, a partir del byte 7, para más detalles	8:x:03
	4	Falta tensión auxiliar externa	Fallo en la alimentación 24 V del FM 355	8:x:04
	6	Contenido EEPROM no válido	Fallo de la tensión de alimentación durante un proceso de escritura en la EEPROM. El módulo arranca con los parámetros predeterminados.	8:x:03
	7	Error de parametrización	El módulo no puede utilizar un parámetro. Motivo: Parámetro desconocido o combinación de parámetros no permitida. Consulte el menú **Sistema de destino > Mostrar errores de parametrización**	8:x:07
1	0 ... 3	Clase de módulo	Siempre tiene asignado un 8	–
	4	Diagnóstico específico de canal	Se activa cuando el módulo puede proporcionar información del canal adicional y hay un error de canal (consulte DS1 bytes 7 a 12)	–
2	3	Vigilancia de tiempo (watch dog) activado	Error de hardware	8:x:33
3	2	Error EPROM	Módulo defectuoso	8:x:42
	4	Error ADC/DAC	Módulo defectuoso	8:x:44

Asignación del registro de diagnóstico DS1

El registro de diagnóstico DS1 se compone de 16 bytes. Los 4 primeros bytes son idénticos a los del registro de diagnóstico DS0. La siguiente tabla muestra la asignación de los bytes restantes. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Tabla 11-8 Asignación de los bytes del 4 a 12 del registro de diagnóstico DS1

Byte	Bit	Significado	Nota	Nº de evento
4	0 ... 7	Tipo de canal	Siempre tiene asignado 75H	–
5	0 ... 7	Longitud de la información de diagnóstico	Siempre tiene asignado un 8	–
6	0 ... 7	Cantidad de canales	Siempre tiene asignado un 5 (4 reguladores + 1 canal de referencia)	–
7	0 ... 7	Vector de error de canal	Cada bit tiene asignado un canal	–

Byte	Bit	Significado	Nota		Nº de evento
8	0	Error de hardware en la entrada analógica	Diagnóstico específico del canal 1		8:x:B0
	1	sin usar			8:x:B1
	2	Rotura de hilo en la entrada analógica (sólo en el margen 4 a 20 mA)			8:x:B2
	3	sin usar			8:x:B3
	4	Rebase por defecto del rango de medición en la entrada analógica			8:x:B4
	5	Rebase del rango de medición en la entrada analógica			8:x:B5
	6	Rotura de hilo en la salida analógica		Sólo en la salida de intensidad del regulador K	8:x:B6
	7	Cortocircuito en la salida analógica		Sólo en la salida de tensión del regulador K	8:x:B7
9	0 ... 7	Véase byte 8	Diagnóstico específico del canal 2		véase arriba
10	0 ... 7	véase byte 8	Diagnóstico específico del canal 3		véase arriba
11	0 ... 7	véase byte 8	Diagnóstico específico del canal 4		véase arriba
12	0 ... 5	véase byte 8	Diagnóstico para el canal de referencia		véase arriba

Información adicional

Para obtener más información sobre el parámetro "RET_REALU", consulte Lista de los avisos RET_REALU (Página 10287) y el área de instrucciones avanzadas del sistema de ayuda.

Fallo del transductor de medida

Averías en los transductores de medida

El módulo de regulación puede detectar los siguientes fallos en el transductor de medida:

- Rebase por defecto del rango de medición
- Rebase del margen de medición
- Rotura de hilo (no se muestra en todos los márgenes de medición)

Si se produce una de estas averías en el registro de diagnóstico DS0 se define el bit de error de grupo "Error externo" y, en el registro de diagnóstico DS1, bits de error específicos del canal (consulte las tablas en la sección anterior). Cuando desaparecen estas averías los bits correspondientes se desactivan.

La siguiente tabla muestra los límites de los diferentes márgenes de medición en los que los bits de error se activan o desactivan:

Rango de medida	Bit de error Rebase por defecto del rango de medición en ...	Bit de error Rebase del rango de medición en ...	Bit de error Mostrar rotura de hilo
	DS1: Byte 10 a 26, Bit 4	DS1: Byte 10 a 26, Bit 5	DS1: Byte 10 a 26, Bit 2
0 a 20 mA	< –3,5 mA	> 23,5 mA	–
4 a 20 mA	Bit de error = 1 a < 3,6 mA Bit de error = 0 a > 3,8 mA	> 22,8 mA	Bit de error = 1 a < 3,6 mA Bit de error = 0 a > 3,8 mA
0 a 10 V	< –1,175 V	> 11,75 V	–
Pt 100 (–200 a 850 °C) (–328 a 1562 °F)	< 30,82 mV	> 650,46 mV	–
Pt 100 (–200 a 556 °C) (–328 a 1032 °F)	< 30,82 mV	> 499,06 mV	–
Pt 100 (–200 a 130 °C) (–328 a 264 °F)	< 30,82 mV	> 254,12 mV	–
Termoelemento tipo B	< 0 mV	> 13,81 mV	–
Termoelemento tipo J	< –8,1 mV	> 69,54 mV	–
Termoelemento tipo K	< –6,45 mV	> 54,88 mV	–
Termoelemento tipo R	< –0,23 mV	> 21,11 mV	–
Termoelemento tipo S	< –0,24 mV	> 18,7 mV	–
Termopar libre	< Valor de entrada inferior de la línea poligonal	> Valor de entrada superior de la línea poligonal	–

Lista de los avisos RET_REALU

Avisos RET_REALU

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
7000	28672	-32624	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.
7001	28673	-32624	Primera llamada con REQ=1: Transferencia de datos lanzada; BUSY tiene el valor 1.
7002	28674	-32624	Llamada intermedia (REQ irrelevante). Transferencia de datos ya activa; BUSY tiene el valor 1.
8090	32912	-32624	La dirección base lógica indicada no es válida: No existe ninguna asignación en el SDB1/SDB2x, o no se trata de una dirección base.
80A0	32928	-32608	Acuse negativo al leer el módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2	32930	-32606	Error de protocolo para capa 2

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A3	32931	-32605	Error de protocolo en User-Interface / User
80A4	32932	-32604	Fallo de comunicación en el bus K
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo teórico.
80C0	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1	32961	-32575	Los datos de una petición de escritura semejante todavía no se han procesado en el módulo.
80C2	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados en estos momentos.
80C4	32964	-32572	Error de comunicación
80C5	32965	-32571	La periferia descentralizada no está disponible.
80C6	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (petición de escritura).
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (petición de escritura).
853A	34106	-31430	DB de parámetros no presente. (petición de escritura).
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras aparecer un error. (petición de escritura).
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (petición de lectura).
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU. No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (petición de lectura).
873A	34618	-30918	DB de parámetros no presente. (petición de lectura).
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras aparecer un error. (petición de lectura).
80ff	33023	-32513	Indicación de índice incorrecta en el bloque FMT_PAR
Los errores 80A2..80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario. Los avisos con la estructura 7xxx indican situaciones temporales de la comunicación.			

Proyectos de ejemplo de FM 355

11.3.5.6 Utilización del FM 355-2

Configuración del FM 355-2

Configurar el hardware

Procedimiento

Para poder configurar el hardware debe haber creado previamente un proyecto donde pueda guardar la parametrización. A continuación se explican únicamente los pasos más importantes.

1. Inicie el TIA portal y abra la configuración de dispositivos.
2. Seleccione un rack.
3. Seleccione un módulo de regulación del catálogo de módulos.
4. Arrastre el módulo de regulación hasta la posición correspondiente del rack.
5. Anote la dirección de entrada del módulo a partir de la tabla de configuración, p. ej. 272.
   El valor leído se muestra en formato decimal.

Parametrización del FM 355-2

Parámetros del módulo

Procedimiento

A través de la parametrización se configuran los parámetros del módulo.

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
   Seleccione en la ventana de inspección la pestaña "Propiedades".
   Resultado: Las propiedades se muestran en la ventana de inspección.
2. Configure los parámetros básicos del módulo y el resto de parámetros.
3. Configure los demás parámetros del bloque en la vista de configuración.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".
5. Compile el proyecto con "Editar > Compilar".
6. Cargue los datos de parametrización en el estado de STOP de la CPU con "Online > Cargar en dispositivo..."
   Resultado: Los datos se encuentran en la memoria de la CPU y se transfieren desde ésta directamente al módulo.
7. Arranque la CPU.

Parámetros de la entrada analógica

Evaluación de la entrada analógica

Procedimiento

Para seleccionar un tipo de sensor proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Evaluación de la entrada analógica".
3. Defina los siguientes parámetros:
   - Tipo de sensor entrada analógica
     La opción "Vigilancia de rotura de hilo" se habilita para diversos tipos de sensor. Se puede activar la opción según requerimiento.
   - Temperatura de referencia
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Funciones de filtro

Procedimiento

Para definir las funciones de filtro proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Funciones de filtro".
3. Defina los siguientes parámetros:
   - Active la opción "Filtro de señal" para filtrar por una constante de tiempo.
     Si la opción "Filtro de señal" se encuentra seleccionada se puede activar la constante de tiempo del filtro.
   - Constante de tiempo del filtro
     Se puede activar solamente si la opción "Filtro de señal" se encuentra activada.
   - Active la opción "Filtro de raíz cuadrada" para linealizar la señal del sensor.
   - Active la opción "Filtro de polígono" para linealizar la característica de la señal del sensor.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Filtro de señal

Filtro de primer orden cuya respuesta temporal viene determinada por la constante de tiempo.

Filtro de raíz cuadrada

Su finalidad es linealizar señales de sensores que están en relación cuadrada con la magnitud de proceso medida, en las cuales el valor real obedece a una magnitud física.

Filtro de polígono

Su finalidad es linealizar características de sensores.

Los 13 nodos seleccionables se pueden seleccionar en:

- mA en entrada de corriente
- mV en entrada de tensión

Normalización

Procedimiento

Para convertir la señal de entrada en una unidad física proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Normalización".
3. Defina los siguientes parámetros:
   - Valor superior
   - Valor inferior
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Normalización

Para convertir la señal de entrada en otra unidad física mediante interpolación lineal entre el valor inicial (valor inferior) y el valor final (valor superior).

Error de regulación

Realizar un regulador de valor fijo o en cascada

Regulador de valor fijo o en cascada

En las estructuras de regulación 'Regulador de valor fijo o en cascada' el valor real efectivo es idéntico al valor real A.

Seleccionar el tipo de regulador

Para realizar un regulador de valor fijo o en cascada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°]".
3. En el área "Tipo de regulador" seleccione la opción "Regulador de valor fijo o en cascada".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para acondicionar la consigna proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Active los parámetros correspondientes en las diferentes áreas:
   - Origen de la consigna
     Seleccione el origen de la consigna.
   - Consigna de seguridad
     Ajuste aquí la consigna de seguridad. Defina asimismo la reacción a un fallo de la CPU y el comportamiento en arranque.
   - Rampa de consigna
     Defina el tiempo de subida de rampa.
   - Límite de consigna
     Seleccione aquí los límites superior e inferior de consigna.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar origen de la señal para el valor real A

Para seleccionar el origen de la señal para el valor real A proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione el origen correspondiente en el área "Variable manipulada". Se dispone de las siguientes opciones:
   - Cero
   - Entrada analógica acondicionada
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar un regulador de 3 componentes

Regulador de 3 componentes

En la estructura de regulación 'Regulador de 3 componentes', el valor real efectivo se define a través de la suma de los tres valores reales A, B y C, y mediante adición de un Offset ajustable. Los valores reales B y C se pueden evaluar adicionalmente mediante factores.

Seleccionar el tipo de regulador

Para materializar un regulador de tres componentes, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°]".
3. En el área "Tipo de regulador" seleccione la opción "Regulador de 3 componentes".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para ajustar cómo debe acondicionarse la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Origen de la consigna" el origen de referencia de la consigna.
4. Especifique un valor en el área "Consigna de seguridad", el cual será tomado como consigna de seguridad en caso de fallo de la CPU y durante el arranque.
5. Especifique el comportamiento del módulo de regulación válido como "Reacción a fallo de la CPU" y para el "Comportamiento en arranque". Las opciones posibles son:
   - Consigna = última consigna válida: al producirse el evento se asigna a la consigna la última consigna válida. Es decir, se mantiene la última consigna válida.
   - Consigna = consigna de seguridad: al producirse el evento se asigna a la consigna la consigna de seguridad. De esta forma se puede realizar un funcionamiento de emergencia seguro.
6. Especifique en el área "Rampa de consigna" el tiempo de rampa que se debe mantener al modificarse la consigna, para evitar una sobreoscilación.
7. Especifique en el área "Límite de consigna" un límite superior e inferior para la consigna.
8. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar el origen de la señal para los valores reales A, B y C

Para especificar los orígenes de la señal para los valores reales A, B y C proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".

3. Seleccione en el área "Variable maniuplada" el origen de la señal correspondiente para cada valor real. Puede seleccionar si no utiliza ninguna señal ("cero") como origen o si utiliza una de las entradas analógicas acondicionadas.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Registrar factores para los valores reales B y C, así como para el Offset

Para especificar los orígenes de la señal para los valores reales A, B y C proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Especifique en el área "Sumar" el factor de ponderación requerido para los valores actuales B y C.
4. Especifique el "Offset" necesario que se debe adicionar a los valores reales.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar un regulador de relación o mezcla

Seleccionar el tipo de regulador

Para materializar un regulador de relación o de mezcla, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°]".
3. En el área "Tipo de regulador" seleccione la opción "Regulador de relación o mezcla".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para ajustar cómo debe acondicionarse la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Origen de la consigna" el origen de referencia de la consigna.
4. Especifique un valor en el área "Consigna de seguridad", el cual será tomado como consigna de seguridad en caso de fallo de la CPU y durante el arranque.
5. Especifique el comportamiento del módulo de regulación válido como "Reacción a fallo de la CPU" y para el "Comportamiento en arranque". Las opciones posibles son:
   - Consigna = última consigna válida: al producirse el evento se asigna a la consigna la última consigna válida. Es decir, se mantiene la última consigna válida.
   - Consigna = consigna de seguridad: al producirse el evento se asigna a la consigna la consigna de seguridad. De esta forma se puede realizar un funcionamiento de emergencia seguro.

6. Especifique en el área "Rampa de consigna" el tiempo de rampa que se debe mantener al modificarse la consigna, para evitar una sobreoscilación.
7. Especifique en el área "Límite de consigna" un límite superior e inferior para la consigna.
8. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar origen de la señal para los valores reales A y D

Para especificar los orígenes de la señal para los valores reales A y D proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Variable manipulada" el origen de la señal correspondiente para cada valor real. Puede especificar no utilizar ninguna señal ("cero") como origen o utilizar una de las entradas analógicas acondicionadas, para el valor real D puede seleccionar también el valor manipulado de uno de los reguladores.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Registrar multiplicador

Para especificar los orígenes de la señal para los valores reales A y D proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Multiplicar" el Offset deseado.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

En el regulador de relación, el valor real A se utiliza como magnitud de regulación y el valor real D como magnitud de relación. La entrada de consigna sirve de factor de relación. Ésta se acondiciona para obtener una consigna efectiva mediante multiplicación por el valor real D y adición de un offset ajustable. Si se desactiva el valor real D, solamente se suma el Offset a la consigna.

Ajustar la generación de alarmas

Procedimiento

Para ajustar la generación de alarma, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".

3. En el área "Generar alarma" seleccione las opciones necesarias:
   - En el área "Avisador de valor límite de" seleccione la señal a la que se debe aplicar la alarma.
   - En "Alarma límite superior" especifique el valor a partir del cual se disparará la alarma al alcanzarse el límite superior.
   - En "Advertencia límite superior" especifique el valor a partir del cual se disparará una advertencia cuando se produzca una aproximación al límite superior.
   - En "Advertencia límite inferior" especifique el valor a partir del cual se disparará una advertencia cuando se produzca una aproximación al límite inferior.
   - En "Alarma límite inferior" especifique el valor a partir del cual se disparará la alarma al alcanzarse el límite inferior.
   - Para evitar que centelleen los indicadores de vigilancia se puede configurar una histéresis en "Histéresis".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Algoritmo de regulación

Realizar una regulación P

Procedimiento

Para realizar una regulación P proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "P", en "Modo de operación".
   Todas las opciones parametrizables para una regulación P se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación PI

Procedimiento

Para realizar una regulación PI proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".

4. En el área "General" seleccione la opción "PI", en "Modo de operación".
   Todas las opciones parametrizables para una regulación PI se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación I

Procedimiento

Para realizar una regulación I proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "I", en "Modo de operación".
   Todas las opciones parametrizables para una regulación I se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación PID

Procedimiento

Para realizar una regulación PID proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "PID", en "Modo de operación".
   Todas las opciones parametrizables para una regulación PID se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación PD

Procedimiento

Para realizar una regulación PD proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".

4. En el área "General" seleccione la opción "PD", en "Modo de operación".
   Todas las opciones parametrizables para una regulación PD se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar un regulador de temperatura con lógica difusa

Procedimiento

Para realizar una regulación P proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Módulo de regulación de temperatura por lógica difusa".
   Todas las opciones parametrizables para módulo de regulación de temperatura por lógica difusa se activan, el resto de opciones se desactivan.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Suprimir el nivel de ruido del error de regulación

Procedimiento

Para suprimir el nivel de ruido del error de regulación proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. Especifique en el área "General" el ajuste "Ancho de zona muerta".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Control anticipativo

Procedimiento

Para conectar una variable perturbadora para control anticipativo proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "General" seleccione de la lista "Origen de la perturbación" el origen de la variable perturbadora.
   Seleccione la entrada "Cero" para no conectar ninguna variable perturbadora para control anticipativo.

4. En "Factor de perturbación" especifique un valor de ponderación para la variable perturbadora.
   Un valor < 0 debilita la perturbación, un valor > 0 amplifica la perturbación. La perturbación permanece invariable si se especifica el valor 1.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Poner la acción P en rama de realimentación

Procedimiento

Para realizar un regulador de valor fijo o en cascada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Acción P" seleccione la opción "Por rama de realimentación".
   Si la opción está activada, la acción P queda configurada por rama de realimentación.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Debilitar la acción D cuando se producen modificaciones en la consigna

Seleccionar el tipo de regulador

Para realizar un regulador de valor fijo o en cascada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Acción D" seleccione con opción "Acción D" el origen de la acción D.
4. En el área "Factor derivativo" especifique el factor para la acción D.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Invertir el efecto del regulador

Principio

La inversión del regulador, cambiando la asignación de "error de regulación ascendente = magnitud manipulada ascendente" a "error de regulación ascendente = magnitud manipulada descendente", se realiza ajustando una ganancia proporcional (GAIN).

Procedimiento

Para invertir el efecto del regulador proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".

3. Especifique en el área "General" una ganancia proporcional con signo inverso.
   Si se encuentra especificada una ganancia proporcional positiva y se desea invertir el efecto del regulador, especifique una ganancia proporcional negativa.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Salida del regulador

Realizar un regulador continuo

Para realizar un regulador continuo se requiere un módulo analógico.

Procedimiento

1. Aplicar una variable manipulada externa (Página 10301)
2. Ajustar la variable manipulada de seguridad (Página 10302)
3. Activar el modo de seguimiento (Página 10302)
4. Limitar la variable manipulada (Página 10303)
5. Controlar dos actuadores - rango partido (Página 10303)
6. Seleccionar la señal de salida y el tipo de señal (Página 10305)

Regulador paso a paso sin realimentación de posición

Para realizar un regulador paso a paso sin realimentación de posición se requiere un módulo digital.

Procedimiento

1. Seleccionar la salida del regulador Regulador paso a paso sin realimentación de posición
2. Aplicar una variable manipulada externa (Página 10301)
3. Ajustar la variable manipulada de seguridad (Página 10302)
4. Formador de impulsos de la regulación paso a paso (Página 10304)

Realizar un regulador paso a paso con realimentación de posición

Para realizar un regulador paso a paso con realimentación de posición se requiere un módulo digital.

Procedimiento

1. Seleccionar la salida del regulador Regulador paso a paso con realimentación de posición
2. Aplicar una variable manipulada externa (Página 10301)
3. Activar el modo de seguimiento (Página 10302)

4. Ajustar la variable manipulada de seguridad (Página 10302)
5. Limitar la variable manipulada (Página 10303)
6. Ajustar la realimentación de posición (Página 10302)
7. Formador de impulsos de la regulación paso a paso (Página 10304)

Realizar un regulador de impulsos

Para realizar un regulador de impulsos se requiere un módulo digital.

Procedimiento

1. Seleccionar la salida del regulador Regulador de impulsos
2. Aplicar una variable manipulada externa (Página 10301)
3. Activar el modo de seguimiento (Página 10302)
4. Ajustar la variable manipulada de seguridad (Página 10302)
5. Limitar la variable manipulada (Página 10303)
6. Acondicionar la señal analógica para la conversión en señal digital
7. Convertir una señal analógica en una señal digital (Página 10304)

Parametrizar las funciones de la salida del regulador

Aplicar una variable manipulada externa

Procedimiento

Para aplicar la magnitud manipulada externa proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Seleccione en el área "Magnitud manipulada externa" la opción que debe activar la magnitud manipulada externa. Para ello puede seleccionar las siguientes opciones:
   – Activación del bloque de función.
   – Activación del bloque de función o de una de las entradas digitales.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Activar el modo de seguimiento

Procedimiento

Para activar el modo de seguimiento proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Seleccione en el área "Modo de seguimiento" la opción que debe activar la magnitud manipulada externa. Para ello puede seleccionar las siguientes opciones:
   - Activación del bloque de función.
   - Activación del bloque de función o de una de las entradas digitales.
4. Seleccione en el área "Modo de seguimiento" el origen de la señal a aplicar para el seguimiento. Para ello puede seleccionar las siguientes opciones:
   - Cero, con lo cual no se emplea ningún origen.
   - Una de las entradas analógicas.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Ajustar la realimentación de posición

Procedimiento

Para ajustar la realimentación de posición proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida digital > Salida [n°]".
3. Seleccione en el área "Entrada de realimentación de posición" la opción "Origen" para especificar la señal de realimentación de posición.
   - Cero: no se produce ninguna realimentación de posición.
   - Una de las entradas analógicas acondicionadas.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Ajustar la variable manipulada de seguridad

Procedimiento

Para ajustar la variable manipulada de seguridad proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Especifique en el área "Variable manipulada de seguridad" el valor a utilizar como variable manipulada de seguridad.

4. Seleccione en el área "Activación" la opción que debe activar la magnitud manipulada externa. Para ello puede seleccionar las siguientes opciones:
   – Activación del bloque de función.
   – Activación del bloque de función o de una de las entradas digitales.
5. Seleccione en "Reacción al arrancar el módulo" la opción a ejecutar al arrancar el módulo. Para ello puede seleccionar las siguientes opciones:
   – Lazo cerrado: el módulo entra inmediatamente en este modo al arrancar.
   – Variable manipulada de seguridad: primero se regula a la variable manipulada de seguridad.
6. Seleccione en el área "Reacción si falla el transductor de medida del valor real A" la opción a ejecutar si falla el transductor de medida del valor real A.
7. Seleccione en el área "Reacción si falla el transductor de medida de una entrada" la opción a ejecutar si falla el transductor de medida de una entrada.
8. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Limitar la variable manipulada

Procedimiento

Para limitar la variable manipulada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Especifique en el área "Límite de señal de salida" los siguientes valores:
   – Valor superior: representa el límite superior para la señal de salida.
   – Valor inferior: representa el límite interior para la señal de salida.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Controlar dos actuadores - rango partido

Procedimiento

Para controlar dos actuadores proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Seleccione en el área "Rango partido" la opción "Activar" para controlar dos actuadores.

4. Especifique los siguientes valores, para las señales de entrada y salida A y también para las entradas de entrada y salida B.
   - Señal de entrada inferior.
   - Señal de entrada superior.
   - Señal de salida inferior.
   - Señal de salida superior.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Formador de impulsos de la regulación paso a paso

Procedimiento

Para parametrizar el formador de impulsos proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida digital > Salida [n°]".
3. Especifique en el área "Formador de impulsos" los siguientes parámetros:
   - Tiempo de posicionamiento del motor.
   - Duración mínima de impulso.
   - Duración mínima de pausa.
4. Seleccione la opción adecuada para los valores superior e inferior, a partir de la cual se aplicará el valor:
   - del bloque de función.
   - Del bloque de función o de una de las entradas digitales.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Convertir una señal analógica en una señal digital

Procedimiento

Para convertir una señal analógica en una señal digital proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida digital > Canal [n°]".
3. Especifique en el área "Formador de impulsos" los siguientes parámetros:
   - Duración mínima de impulso.
   - Duración mínima de pausa.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar la señal de salida y el tipo de señal

Procedimiento

Para seleccionar la señal de salida y el tipo de señal proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida analógica > Salida [n°]".
3. Seleccione en la opción "Origen" la señal a utilizar como señal de salida. Se dispone de las siguientes opciones:
   – Cero: ningún origen para la señal de salida.
   – Una de las entradas analógicas.
   – Uno de los valores manipulados para el regulador A o B.
4. Seleccione en la opción "Tipo de señal" el tipo de señal en que se emitirá la señal de salida.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Puesta en servicio del FM 355-2

Crear y configurar un DB de instancia

Crear y configurar un DB de instancia

1. Cree un programa de usuario que efectúe cíclicamente una llamada a la instrucción FMT_PID.
2. En el programa del usuario, ponga READ_OUT cíclicamente a TRUE.
3. Genere para cada canal del regulador un DB de instancia de la instrucción FMT_PID.
4. Para cada DB de instancia, introduzca la dirección de módulo en el parámetro MOD_ADDR. Utilice como dirección de módulo del FM 355-2 la dirección inicial que ha especificado al configurar el hardware.
5. Introduzca en el parámetro FMT_PID.CHANNEL el número del canal del regulador (0, 1, 2 o 3).

1. Si utiliza las instrucciones FMT_PAR y FMT_TUN, introduzca en los parámetros FMT_PAR.CHANNEL y FMT_TUN.CHANNEL el número del canal del regulador.
2. Compruebe que esté activado el "Tipo de sensor Entrada analógica" del canal en cuestión. Si no se procesa la entrada analógica, todos los parámetros PID se ajustan a cero, y no se calcula ninguna variable manipulada.

Nota

DBs multiinstancia

Si utiliza DBs multiinstancia en su proyecto, los FM se deben poner en servicio a través de una tabla de observación.

Consulte también

FMT_PID (Página 7528)
FMT_TUN (Página 7557)
FMT_PAR (Página 7549)

Puesta en servicio del regulador

Introducción

Los parámetros PID se pueden optimizar del modo siguiente:

- Optimización inicial
- Optimización fina de la calefacción
- Optimización fina de la refrigeración
- Optimización fina manual

Procedimiento

Para optimizar un regulador, proceda del siguiente modo:

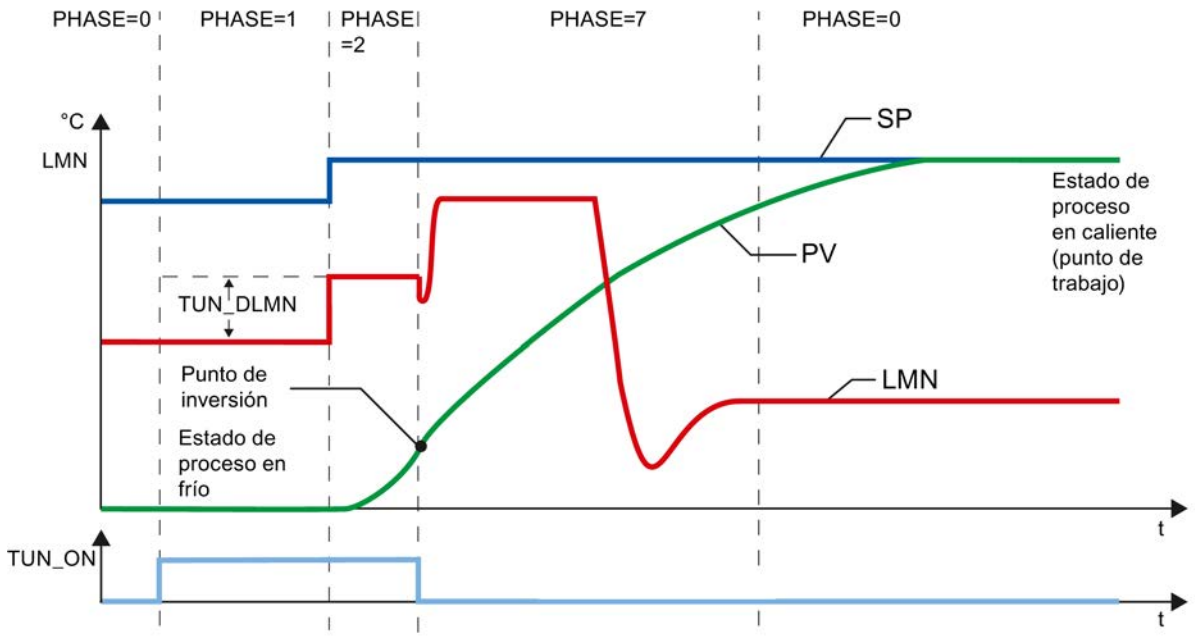
1. Coloque la CPU en el estado RUN.
2. Al utilizar un regulador S se mide el tiempo de posicionamiento del motor.
3. Realice una optimización del regulador.
4. Observe el lazo de regulación con el registrador de curvas.

Posibilidades de optimización del regulador

Optimización inicial

Durante esta optimización, el punto de operación del estado frío se pone en marcha a través de un escalón de consigna.

Con TUN_ON = TRUE, cree la disposición de optimización. El regulador cambia de PHASE = 0 a PHASE = 1.



La variable manipulada de la optimización (LMN0 + TUN_DLMN) se aplica mediante modificación de la consigna (transición PHASE 1 -> 2). La consigna no es efectiva hasta que se alcanza el punto de inversión (sólo entonces se conmutará al modo Automático).

El delta de la excitación manipulada (TUN_DLMN) se define de acuerdo con la modificación admisible del valor real bajo responsabilidad propia. El signo de TUN_DLMN debe estar en concordancia con la modificación prevista del valor real (observar el sentido de regulación).

El escalón de consigna y TUN_DLMN deben estar coordinados. Un TUN_DLMN demasiado grande conlleva el peligro de que el punto de inversión no esté dentro del 75% del escalón de consigna.

Sin embargo, TUN_DLMN debe ser lo suficientemente grande como para que el valor real alcance como mínimo el 22% del escalón de consigna. En caso contrario, el procedimiento permanece en modo Optimización (fase 2).

Solución: reduzca la consigna durante la búsqueda del punto de inversión.

Nota

En procesos con un fuerte retardo, es aconsejable definir para la optimización una consigna de destino un poco más baja que el punto de operación deseado y observar con detenimiento los bits de estado y el PV (peligro de sobreoscilaciones).

Optimización únicamente en el rango lineal:

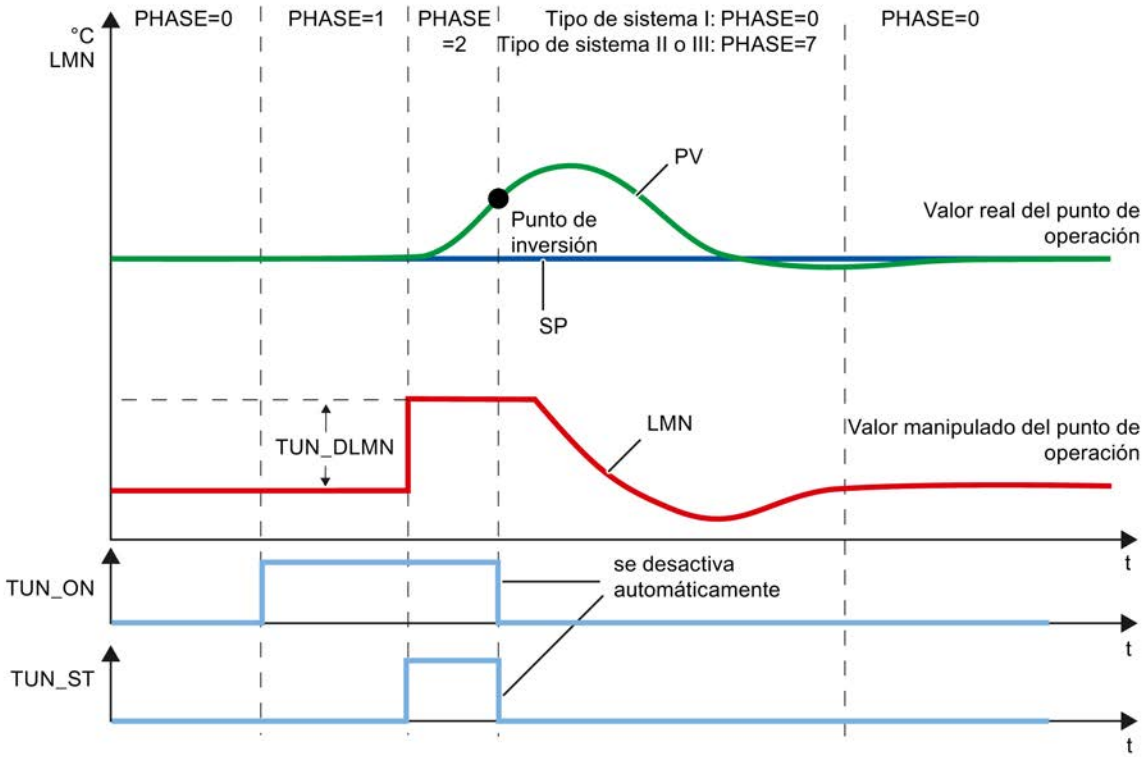
Determinados sistemas regulados (p. ej., calderas de fusión de zinc o magnesio) pasan por un rango no lineal poco antes del punto de operación (modificación del estado de agregación).

Si se selecciona con habilidad el escalón de consigna es posible limitar la optimización al rango lineal. Si el valor real ha pasado del 75% del escalón de consigna (SP_INT-PV0), la optimización concluye.

Paralelamente, TUN_DLMN debería reducirse hasta que el punto de inversión se encuentre con seguridad antes de alcanzar el 75% del escalón de consigna.

Optimización fina de la calefacción

Durante esta optimización, el proceso se excita, en caso de una consigna constante, mediante un salto de la variable manipulada.



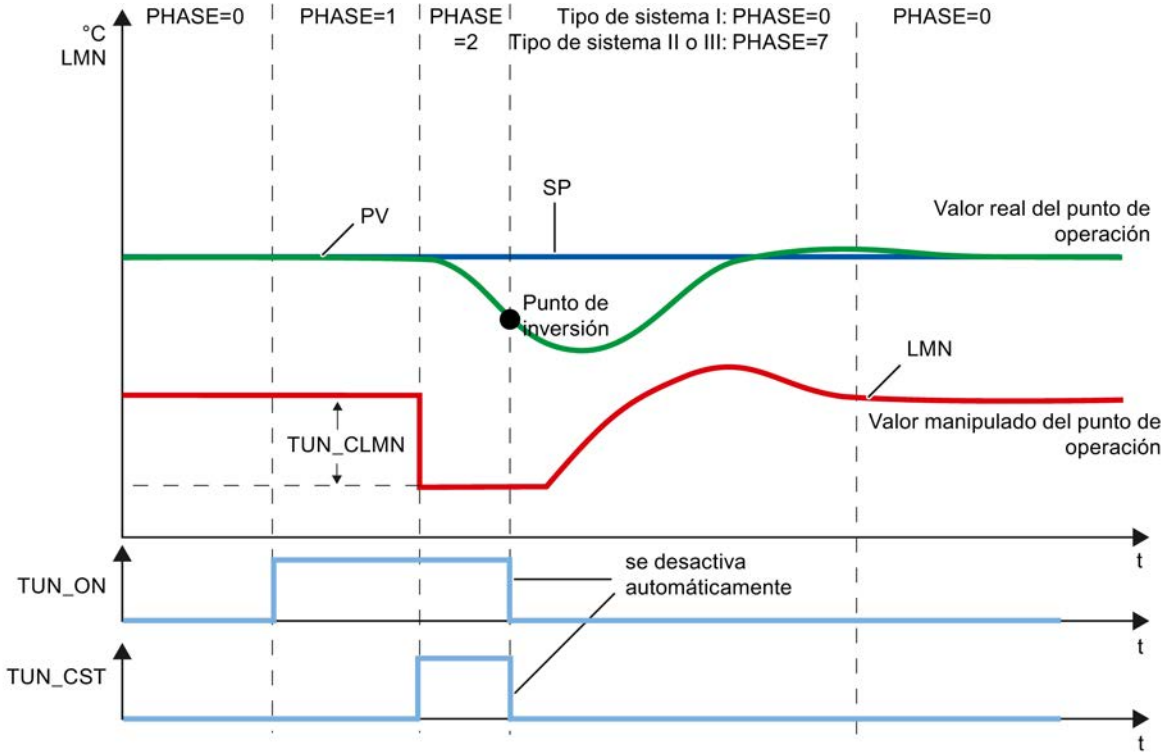
La variable manipulada de la optimización (LMN0 + TUN_DLMN) se aplica mediante activación del bit de inicio TUN_ST (transición fase 1 -> 2). Si se modifica la consigna, la nueva consigna no es efectiva hasta que no se alcanza el punto de inversión (sólo entonces se conmuta al modo Automático).

El delta de la excitación manipulada (TUN_DLMN) se define de acuerdo con la modificación admisible del valor real bajo responsabilidad propia. El signo de TUN_DLMN debe estar en concordancia con la modificación prevista del valor real (observar el sentido de regulación).

ATENCIÓN
Si la excitación se realiza mediante TUN_ST no se produce ninguna desconexión de seguridad al 75%. La optimización finaliza cuando se alcanza el punto de inversión. Sin embargo, en procesos con ruidos es posible que se sobrepase claramente el punto de inversión.

Optimización fina de la refrigeración

Durante esta optimización, el proceso se excita, en caso de una consigna constante, mediante un salto de la variable manipulada.



La variable manipulada de la optimización (LMN0 + TUN_CLMN) se aplica mediante la activación del bit de inicio TUN_CST (transición fase 1 -> 2). Entretanto, la variable manipulada determinada anteriormente LMN0 se mantiene constante y TUN_CLMN se aplica directamente en la función de rango dividido, por lo que sólo se modifica la potencia de refrigeración. TUN_CST se vuelve a desactivar de inmediato.

Si modifica la consigna durante la fase 2, la nueva consigna no es efectiva hasta que se alcance el punto de inversión (sólo entonces se pasa al modo automático).

El delta de la excitación manipulada (TUN_CLMN) se define de acuerdo con la modificación admisible del valor real bajo responsabilidad propia. El signo de TUN_CLMN debe estar en concordancia con la modificación prevista del valor real (observar el sentido de regulación). Tenga en cuenta que un TUN_CLMN negativo provoca un aumento de la potencia de refrigeración. No obstante, con TUN_CLMN > 0.0 también se puede aplicar un escalón de la variable manipulada reduciendo la potencia de refrigeración.

ATENCIÓN
Si la excitación se realiza mediante TUN_CST no se produce ninguna desconexión de seguridad al 75%. La optimización finaliza cuando se alcanza el punto de inversión. Sin embargo, en procesos con ruidos es posible que se sobrepase claramente el punto de inversión.

Función de rango dividido

En la optimización, la pendiente de la función de rango dividido se atribuye al proceso. Si modifica la pendiente de la función de rango dividido después de la optimización, también debe adaptar en consecuencia el parámetro del regulador GAIN o RATIOFAC.

Optimización fina manual en modo de regulación

Para conseguir un comportamiento de consigna sin sobreoscilaciones, pueden tomarse las medidas siguientes:

- Adaptar la zona de regulación
- Optimizar el comportamiento de referencia
- Amortiguar los parámetros de regulación
- Modificar los parámetros de regulación

Optimización automática

Requisitos del proceso

Régimen transitorio

El proceso debe presentar un régimen transitorio estable, con retardo y asintótico.

Después de un salto de la magnitud manipulada, el valor real debe pasar a un estado estacionario. Con ello se excluyen procesos que presentan un comportamiento oscilante incluso sin regulación, así como procesos regulados sin compensación (integrador en el proceso regulado).

Consolidación de un estado inicial cuasi-estacionario (fase 0)

Si el valor real presenta oscilaciones de baja frecuencia, p. ej., debido a unos parámetros del regulador incorrectos, antes de iniciar la optimización se debe situar el regulador en modo manual y esperar a que desaparezcan las oscilaciones. Alternativamente, también puede conmutar a un regulador PI con ganancia proporcional pequeña y periodo de integración grande.

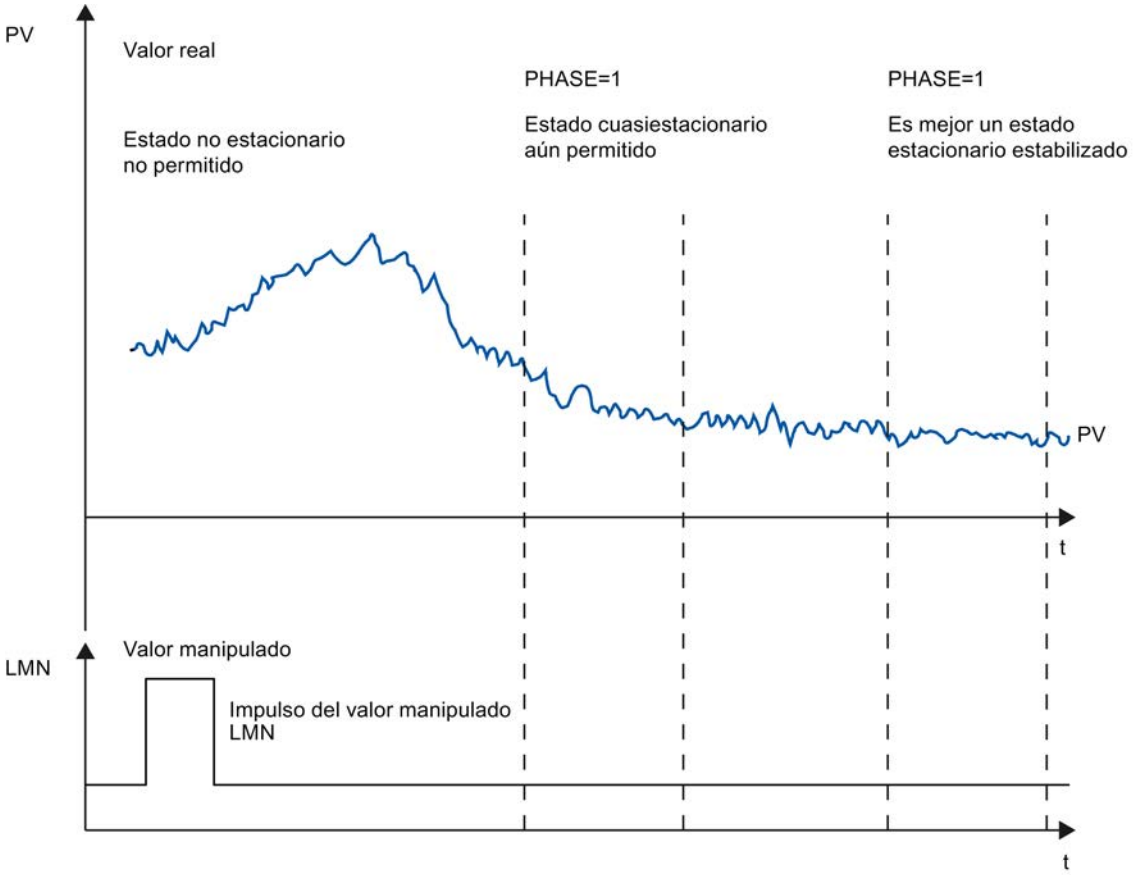
Ahora deberá esperar hasta que se alcance el estado estacionario, es decir, hasta que el valor real y la variable manipulada sean estacionarios. También está permitido un régimen transitorio asintótico o un desplazamiento lento del valor real (estado cuasi-estacionario,

consulte la figura siguiente). La magnitud manipulada debe ser constante u oscilar alrededor de un valor medio constante.

Nota

Evite modificar la magnitud manipulada justo antes del inicio de la optimización. La variable manipulada también se puede modificar de manera involuntaria al alterar las condiciones de ensayo (p. ej., cerrar una puerta abierta). Si este fuera el caso, deberá esperar como mínimo a que el valor real experimente de nuevo un régimen transitorio de forma asintótica a un estado estacionario No obstante, conseguirá mejores parámetros de regulación, si espera hasta que el proceso de régimen transitorio haya finalizado completamente.

En la siguiente figura, se muestra el régimen transitorio al estado estacionario:



Linealidad y zona de trabajo

El proceso debe presentar un comportamiento lineal en la zona de trabajo. Un comportamiento no lineal se produce, p. ej., con un cambio de un estado de agregación. La optimización debe tener lugar en una parte lineal de la zona de trabajo.

Esto significa que, tanto para la optimización como para el modo de regulación normal, los efectos no lineales dentro de dicha zona de trabajo deben ser despreciablemente pequeños. Sin embargo, en caso de cambio del punto de operación, el proceso se puede optimizar de

nuevo ejecutando la optimización en un margen reducido en torno al nuevo punto de operación y si no se atraviesa ninguna zona de no linealidad.

Si la presencia de ciertas zonas estáticas de no linealidad (p. ej., características de válvulas) es conocida, resulta siempre aconsejable compensarlas previamente con una línea poligonal para linealizar el comportamiento del proceso.

Perturbaciones en procesos de temperatura

Las perturbaciones, como la transferencia de calor a zonas colindantes, no deben influir excesivamente en el proceso global de temperatura. Así, p. ej., al optimizar las zonas de una extrusionadora, todas las zonas deben calentarse simultáneamente.

Desarrollo de la optimización del regulador

Parámetros de regulador PI/PID

En el FM 355-2, con la optimización del regulador se ajustan automáticamente los parámetros del regulador PI/PID. Se pueden optimizar tanto procesos de sólo calefacción o sólo refrigeración, como procesos con dos actuadores de efectos opuestos (p. ej., actuador calefactor y actuador refrigerador).

Fases de la optimización del regulador

Durante la optimización del regulador se ejecutan fases individuales, que se indican en el parámetro PHASE .

PHASE = 0

No se efectúa ninguna optimización. El FM 355-2 funciona en modo automático o en modo manual.

Mientras PHASE = 0, el sistema regulado debe cumplir con los requisitos para una optimización.

Al final de la optimización, FM 355-2 vuelve a cambiar a PHASE = 0.

PHASE = 1

FM 355-2 se encuentra en disposición de optimización. La PHASE = 1 sólo puede iniciarse si se cumplen las condiciones para una optimización.

Mientras PHASE = 1 se calculan los siguientes valores:

- Ruido del valor real NOISE_PV
- Subida inicial PVDT0
- Valor medio de la variable manipulada

PHASE = 2

En la fase 2 se busca el punto de inversión del valor real con una variable manipulada constante. El procedimiento impide que el punto de inversión se detecte demasiado pronto debido al ruido de PV:

Al principio, este cálculo del valor medio permanece inactivo, es decir, sólo se calcula el valor medio a lo largo de un ciclo. En cuanto el ruido excede una medida determinada, se duplica el número de ciclos.

Se determinan el período y la amplitud del ruido. La búsqueda del punto de inversión no se cancela y se sale de la fase 2 hasta que el gradiente es siempre menor que el ascenso máximo durante el período estimado. Sin embargo, TU y T_P_INF se calculan en el punto de inversión real.

De todas formas, la optimización no finaliza hasta que se cumplen las dos condiciones siguientes:

1. El valor real está más alejado del punto de inversión que 2*NOISE_PV.
2. El valor real ha superado el punto de inversión en un 20%.

Nota

En caso de excitación mediante escalón de consigna, la optimización finaliza a más tardar cuando el valor real ha superado el 75% del escalón de consigna (SP_INT-PV0) (véase abajo).

PHASE = 3, 4, 5, 6

Cada una de las fases 3, 4, 5 y 6 dura 1 ciclo.

En la fase 3, se guardan los parámetros PI/PID válidos antes de la optimización y se calculan los parámetros del proceso.

En la fase 4, se calculan los nuevos parámetros PI/PID.

En las fases 5 y 6 se calcula la nueva variable manipulada y se envía al sistema regulado.

Si se ha identificado un sistema regulado del tipo I, el modo de optimización finaliza y la optimización se encuentra de nuevo en la fase 0. A continuación, el regulador arranca siempre con LMN = LMN0 + 0.75*TUN_DLMN en el modo automático (aunque lo hubiera regulado en el modo manual antes del inicio de la optimización).

Si se ha identificado un sistema regulado del tipo II o III, la optimización pasa a la fase 7.

PHASE = 7

En los sistemas típicos de temperatura (tipo de sistema I) se corre el riesgo de detectar el punto de inversión demasiado pronto a causa del ruido. En tal caso, el escaso tiempo necesario para alcanzar el punto de inversión T_P_INF puede provocar que se detecte el tipo de sistema II o III.

Por ello, en la fase 7 se comprueba si el tipo de sistema es correcto. Esta comprobación se efectúa en el modo automático con los parámetros del regulador recién calculados y finaliza, a más tardar, al cabo de 6*TA (tiempo de compensación) tras el punto de inversión. Si se

detecta el tipo de sistema I, los parámetros del regulador se calculan de nuevo (STATUS_D = 122); de lo contrario, los parámetros del regulador se mantienen sin cambios.

La comprobación del tipo de sistema se cancela si el valor real durante la optimización del punto de trabajo alcanza de nuevo el valor real previo a la optimización PV0.

Si la fase 7 se cancela con TUN_ON = FALSE, se conservan los parámetros del regulador ya calculados.

Nota

En principio, la optimización de la calefacción puede comenzar durante la refrigeración. En tal caso, la comprobación de orden en la fase 7 no se efectúa.

El caso contrario (inicio de la optimización de la refrigeración durante la calefacción) no resulta crítico ya que, en principio, la fase 7 de la optimización de refrigeración no se efectúa.

Cancelación anticipada de la optimización

En las fases 1, 2 ó 3, la optimización se puede cancelar sin calcular parámetros nuevos mediante el restablecimiento de TUN_ON = FALSE. El regulador empieza en modo automático con LMN = LMN0. Si el regulador se encontraba en modo manual antes de la optimización, se emitirá la variable manipulada manual antigua.

Si la optimización se cancela en la fase 4, 5, 6 o 7 con TUN_ON = FALSE, los parámetros del regulador calculados hasta ese momento se conservan.

Resultado de la optimización

Estado de optimización

La cifra izquierda de STATUS_H y STATUS_C muestra el estado de optimización.

STATUS_H STATUS_C	Resultado
0	Se han encontrado parámetros del regulador predeterminados, ninguno o ninguno nuevo.
10000	Encontrados parámetros de regulador apropiados
2xxxx	Se han encontrado parámetros del regulador superiores a los valores estimados; compruebe el comportamiento de regulación, lea el aviso de diagnóstico STATUS_H/C y repita la optimización del regulador.
3xxxx	Se han producido errores de operador; lea el aviso de diagnóstico STATUS_H/C y repita la optimización del regulador.

Los siguientes parámetros del regulador se actualizan tras detectar el punto de inversión en la fase 6 en el FM 355-2 y en el DB de instancia online de la instrucción FMT_PID:

- Factor de atenuación del componente proporcional PFAC_SP = 0,8
- Ganancia del regulador GAIN
- Periodo de integración TI (limitado a ≥0,5 s)

- Periodo de derivación TD (limitado a ≥1,0 s)
- Factor diferencial D_F = 5,0
- Zona de regulación on/off CONZ_ON=TRUE/FALSE
- Ancho de zona de regulación CON_ZONE=250/GAIN
- P_SEL = TRUE (aunque previamente el regulador integral puro)

Si RATIOFAC<>0,0, CON_ZONE se multiplica por 1,5.

Si el TD calculado por la optimización es <1,0 s, sólo se detecta un regulador PI y tanto PID_ON como los parámetros PID se ponen a 0.

La zona de regulación sólo se activa si el tipo de sistema es adecuado (tipo de sistema I y II) y si el regulador PID está activado (CONZ_ON = TRUE).

En función de PID_ON se regula con el regulador PI o PID. Los parámetros antiguos del regulador se guardan y pueden volver a activarse mediante UNDO_PAR. Además, también se guarda un juego de parámetros PI y un juego de parámetros PID. Mediante LOAD_PID y la activación correspondiente de PID_ON, también es posible cambiar más adelante entre los parámetros optimizados PI o PID.

Si previamente existía una descomposición de la estructura (componente diferencial en la realimentación), ésta se mantiene.

Nota

Nada más terminar la optimización del regulador, compruebe el funcionamiento correcto de los parámetros del regulador.

Función de rango dividido

En la optimización, la pendiente de la función de rango dividido se atribuye al proceso. Si modifica la pendiente de la función de rango dividido después de la optimización, también debe adaptar en consecuencia el parámetro del regulador GAIN.

Síntomas de error y soluciones posibles

Interceptar errores de operador

Todos los errores recogidos en esta tabla provocan que la optimización quede cancelada.

Error	STATUS y medidas	Comentario
Activación simultánea de TUN_ON y TUN_ST o TUN_CST	STATUS_H/C = 30001 • Transición a la fase 0 • TUN_ON = FALSE	No puede aparecer con los asistentes, ya que en la primera pantalla TUN_ST y TUN_CST se ponen a FALSE.
El signo de la modificación de la consigna no concuerda con el cambio del valor real.	Finalización de la optimización en el punto de inversión Desconexión de seguridad al 75% sin efecto	

Error	STATUS y medidas	Comentario
|TUN_DLMN| o |TUN_DLMN efectivo| < 5% (fin de la fase 1) Caso especial STATUS_H=30004: Diferencia efectiva de la variable manipulada limitada por los límites de rango dividido y no por los límites de la variable manipulada	STATUS_H/C = 30002/30004 • Transición a fase 99 (fase 0 en optimización paralela) • Se emite LMN = LMN0	Puede que haya parametrizado un valor demasiado bajo para TUN_DLMN o que la variable manipulada estuviera próxima a un límite de posicionamiento antes del inicio de la optimización. Si STATUS_H=30004: Recuerde además que, p. ej., con LMN_A < 5% no es posible una optimización de calentamiento con TUN_DLMN negativo (razón: La potencia de refrigeración no puede modificarse). En ese caso hay que impedir que el regulador se ajuste a la nueva consigna y abandone innecesariamente el punto de operación estacionario (no es posible en la optimización paralela). Proceda del siguiente modo: • Reducir consigna • TUN_ON=FALSE • Corregir TUN_DLMN • Iniciar de nuevo la optimización
Sólo en el regulador paso a paso sin realimentación de posición analógica: |TUN_DLMN efectivo| < 5% (inicio de la fase 2)	STATUS_H = 30002 • Transición a la fase 0 • TUN_ON = FALSE	El error aparece si |TUN_DLMN| >= 5% pero la variable manipulada estaba próxima a un límite de posicionamiento antes del inicio de la optimización.
Si TUN_CST: |TUN_CLMN| o |TUN_CLMN efectivo| < 5% (fin de la fase 1) Caso especial STATUS_H=30004: Diferencia efectiva de la variable manipulada limitada por los límites de rango dividido y no por los límites de la variable manipulada.	STATUS_C = 30002/30004 • Transición a la fase 0 • TUN_ON = FALSE	Puede que haya parametrizado un valor demasiado bajo para TUN_CLMN o que la variable manipulada estuviera próxima a un límite de posicionamiento antes del inicio de la optimización.
Si TUN_CST: Caso especial de "|TUN_CLMN efectivo| < 5% (fin de la fase 1)": TUN_CLMN <= -5%, pero LMN_LLM > -5%.	STATUS_C = 30003 • Transición a la fase 0 • TUN_ON = FALSE	El límite inferior LMN_LLM es demasiado alto (p. ej., 0%) e impide por ello la indicación de una potencia de refrigeración.
TUN_CST pero ninguna optimización anterior de calentamiento	STATUS_C = 30008 • Transición a la fase 0 • TUN_ON = FALSE	
Modo de seguridad	STATUS_C = 30009 • Transición a la fase 0 • TUN_ON = FALSE	Si el FM 355-2 pasa en PHASE > 2 al modo de seguridad, esta circunstancia se muestra con STATUS_C=30009.

Punto de inversión no alcanzado (sólo en caso de excitación mediante escalón de consigna)

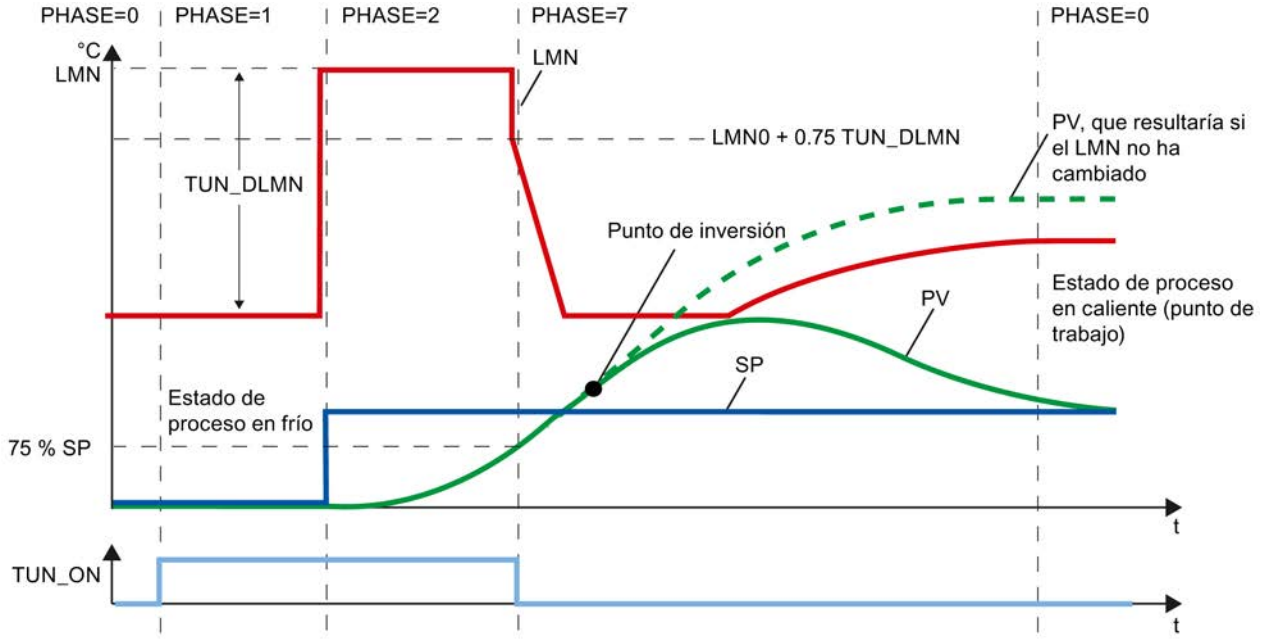
La optimización finaliza como muy tarde cuando el valor real ha pasado el 75 % del escalón de consigna (SP_INT-PV0). Esta circunstancia se notifica mediante "Punto de inversión no alcanzado" en STATUS_H/STATUS_C (2xx2x).

Siempre rige la consigna que está ajustada en cada momento. Así pues, si se reduce la consigna, es posible que la optimización finalice anticipadamente.

En casos de procesos de temperatura típicos, suele bastar con cancelar la optimización al 75 % del escalón de consigna para impedir sobreoscilaciones. Sin embargo, hay que tomar precauciones, especialmente en procesos con un retardo considerable (TU/TA > 0.1, tipo III). Si la excitación manipulada es demasiado fuerte en comparación con el escalón de consigna, el valor real tendrá muchas sobreoscilaciones (hasta el factor 3).

Si, en procesos de orden superior, el punto de inversión sigue muy alejado una vez alcanzado el 75 % del escalón de consigna, hay una sobreoscilación clara. Además, los parámetros del regulados son demasiado rigurosos. En este caso, atenúe los parámetros del regulador o repita el ensayo.

La figura siguiente representa la sobreoscilación del valor real en caso de excitación demasiado fuerte (tipo de sistema III):



En procesos de temperatura típicos, una cancelación justo ante de alcanzar el punto de inversión no es crítica respecto de los parámetros del regulador.

Si repite el ensayo, reduzca TUN_DLMN o TUN_CLMN o bien aumente el escalón de consigna.

Principio: la variable manipulada de optimización debe concordar con el escalón de consigna.

Error de estimación con tiempo de retardo u orden

El tiempo de retardo (STATUS_H/STATUS_C = 2x1xx, 2x2xx o 2x3xx) o el orden (STATUS_H/STATUS_C = 21xxx o 22xxx) no se han podido registrar correctamente. Se sigue trabajando con un valor estimado que puede provocar parámetros no óptimos del regulador.

Repita la optimización y asegúrese de que no se produzcan perturbaciones en el valor real.

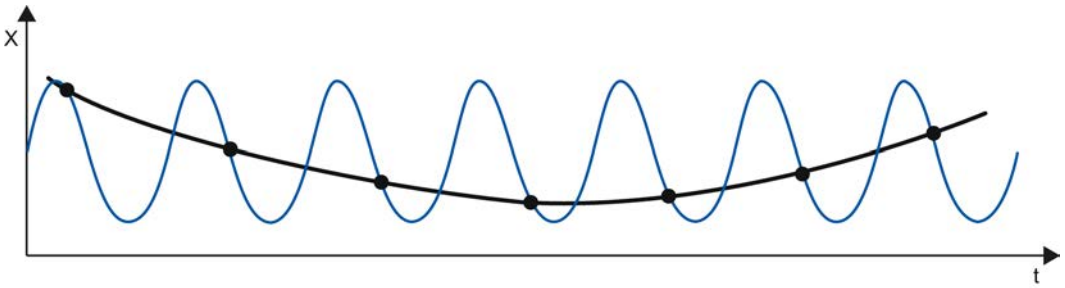
Nota

El caso especial de un proceso PT1 puro también se notifica con STATUS_H/STATUS_C = 2x2xx (TU < CYCLE). En este caso no es necesario repetir el ensayo. Atenúe los parámetros del regulador si la regulación oscila.

Calidad de las señales medidas (ruido medido, perturbaciones de baja frecuencia)

El resultado de la optimización puede verse entorpecido por ruidos medidos o perturbaciones de baja frecuencia. En tal caso, considere lo siguiente:

- Dentro de un período de ruido, el valor real debe muestrearse como mínimo dos veces. La medida de ruido no debe superar un 5% del cambio de señal útil.
- Las perturbaciones radioeléctricas ya no se pueden filtrar con un módulo software. Dichas perturbaciones ya deberían haberse filtrado en el sensor de medición para impedir el efecto aliasing.
  La figura siguiente representa el efecto aliasing con un tiempo de muestreo demasiado grande:



- Si las perturbaciones son de baja frecuencia, se puede asumir que el período de muestreo del FM 355-2 es suficientemente pequeño. Por otro lado, en tal caso el FM 355-2 debe generar una señal medida homogénea debido al amplio intervalo de filtrado por valor medio. Una filtración de valores medios debe abarcar como mínimo dos períodos de ruido. De esta forma, dentro del bloque se generan rápidamente tiempos de muestreo más grandes, de modo que se merma la precisión de la optimización. Una precisión suficiente se garantiza con un mínimo de 40 períodos de ruido hasta el punto de inversión. Medidas posibles al repetir el ensayo: aumentar TUN_DLMN/TUN_CLMN.

Sobreoscilación

En las situaciones siguientes pueden producirse sobreoscilaciones:

Situación	Causa	Solución
Fin de la optimización	• Excitación mediante un cambio demasiado brusco de la variable manipulada en comparación con el escalón de consigna (véase arriba). • Regulador PI activado mediante PID_ON = FALSE.	• Aumentar el escalón de consigna o reducir el escalón de la variable manipulada • Si el proceso admite un regulador PID, inicie la optimización con PID_ON = TRUE.
Optimización en fase 7	Primero se han determinado parámetros más suaves para el regulador (tipo de sistema III) que pueden provocar sobreoscilaciones en la fase 7.	-
Modo de regulación	Regulador PI y con PFAC_SP = 1.0 para el tipo de sistema I.	Si el proceso admite un regulador PID, inicie la optimización con PID_ON = TRUE.

Optimización con regulador paso a paso

Introducción

En general son válidas las explicaciones de la optimización del regulador.

Particularidades para el regulador paso a paso

- Los reguladores paso a paso del FM 355-2 funcionan sin zona de regulación.
- La fase 7 no se lleva a cabo.
- Sin optimización de refrigeración.

Nota

El tiempo de posicionamiento del motor no es determinado por la optimización, por lo que se debe medir o especificar antes del inicio de la optimización.

Diseño del regulador

El tiempo de posicionamiento del motor MTR_TM debe ser lo más pequeño posible en comparación con el tiempo necesario para alcanzar el punto de inversión T_P_INF y el tiempo de retardo TU.

Para valores más grandes del tiempo de posicionamiento del motor, automáticamente los reguladores diseñados resultan más suaves.

Cuanto mayor sea la excitación manipulada TUN_DLMN, tanto más afecta el tiempo de posicionamiento del motor en el diseño del regulador.

Parámetros PI o PID

En comparación con el diseño para reguladores continuos y reguladores de impulsos, los parámetros PI se reducen aprox. un 25%. Los parámetros PID también se calculan (pero sin el 25% de seguridad). Los parámetros PID únicamente se deben utilizar si el tiempo de posicionamiento del motor no es demasiado grande en comparación con los parámetros del proceso y si la solicitación del actuador por el componente diferencial no resulta excesiva.

Regulador paso a paso sin realimentación analógica de posición

La emisión de órdenes Abrir/Cerrar se detiene ya al principio de la fase 1 (cuasi-manual).

En la fase 1 no se calcula el valor medio de la variable manipulada (LMN0).

Al principio de la fase 2, durante el tiempo MTR_TM * TUN_DLMN / 100, se emite una orden Abrir (o bien, si TUN_DLMN es negativo, una orden Cerrar).

Si una señal de tope se activa durante la pulsación en la fase 2, TUN_DLMN efectivo se calcula como: 100 * tiempo / MTR_TM.

Si el valor de TUN_DLMN efectivo es < 5%, se emite el mensaje de error STATUS_H=30002 y la optimización se cancela. Este error sólo puede ocurrir si, contra toda previsión, el actuador se encontraba cerca de un límite.

Fin o cancelación de la optimización

El regulador se inicia con LMN = LMN0 + TUN_DLMN.

Realizar una optimización inicial

Requisitos

- Las exigencias planteadas al proceso se cumplen.
- Se ha creado y configurado un DB de instancia para el canal.
- Como origen del valor real está parametrizado "Instrucción".
- El FM 355-2 no se encuentra en el modo de seguridad (SAFE_ON = FALSE). Si el modo de seguridad se activa, la optimización en ejecución queda cancelada.

Procedimiento

⚠ ADVERTENCIA

Pueden producirse daños materiales considerables, lesiones físicas graves o incluso la muerte.

Durante una optimización, el parámetro LMN_REON no tiene efecto alguno. Tampoco intervienen los circuitos de seguimiento, p. ej., derivados de los límites de alarma. Por ello, la variable manipulada o el valor real pueden tomar valores no deseados, e incluso extremos.

La variable manipulada se especifica mediante optimización. Para cancelar la optimización se debe activar TUN_ON = FALSE. De este modo, LMN_REON vuelve a ser efectivo.

Para determinar los parámetros PID óptimos en la primera puesta en servicio, proceda del siguiente modo:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Medición activada".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. En la lista "Modo" seleccione la entrada "Optimización inicial".
   El FM 355-2 se encuentra en disposición de optimización (PHASE = 1).
4. Espere hasta que alcance el estado estacionario.
5. En el campo "Escalón de la variable manipulada" indique el valor en el que debe aumentarse la variable manipulada.
6. En el campo "Consigna", indique una consigna. El escalón de la variable manipulada se hace efectivo sólo con una nueva consigna.
7. Haga clic en el símbolo "Iniciar optimización".
   Se inicia la optimización inicial. Se muestra el estado de la optimización.

Nota

En la fase 1, una modificación de consigna muy pequeña (p. ej., ruido de medición de una entrada analógica) puede activar una optimización. En ese caso, la optimización concluye muy rápidamente (parámetros de regulador erróneos; peligro de inestabilidad).

Realizar optimización fina de calefacción

Requisitos

- Las exigencias planteadas al proceso se cumplen.
- Se ha creado y configurado un DB de instancia para el canal.
- Como origen del valor real está parametrizado "Instrucción".
- El FM 355-2 no se encuentra en el modo de seguridad (SAFE_ON = FALSE). Si el modo de seguridad se activa, la optimización en ejecución queda cancelada.

Procedimiento

Para determinar los parámetros PID óptimos en el punto de operación, proceda del siguiente modo:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Medición activada".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. En la lista "Modo" seleccione la entrada "Optimización fina de calefacción".
   El FM 355-2 se encuentra en disposición de optimización (PHASE = 1).
4. En el campo "Escalón de la variable manipulada" indique el valor en el que debe aumentarse la variable manipulada.
5. Haga clic en el símbolo "Iniciar optimización".
   Se inicia la optimización fina. Se muestra el estado de la optimización.

Realizar optimización fina de refrigeración

Introducción

En las regulaciones que funcionan con dos actuadores de efectos opuestos (actuador calefactor y actuador refrigerador), el FM355-2 determina con un escalón de la variable manipulada del actuador refrigerador la relación entre las ganancias de proceso RATIOFAC (ganancia de calefacción/ganancia de refrigeración). El ancho de la zona de regulación CON_ZONE también se calcula de nuevo. Los demás parámetros del regulador se mantienen sin cambios.

Requisitos

- Como origen del valor real está parametrizado "Instrucción".
- El FM 355-2 no se encuentra en el modo de seguridad (SAFE_ON = FALSE). Si el modo de seguridad se activa, la optimización en ejecución queda cancelada.
- La optimización fina de calefacción para el canal ha concluido correctamente. Tras una falta de corriente del FM 355-2, la optimización de calefacción se debe repetir.
- En una instalación con varias zonas de temperatura acopladas (p. ej., máquinas de plástico), la optimización fina ha concluido correctamente para todas las zonas y se ha alcanzado el estado estabilizado.
- La variable manipulada A se debe usar para la calefacción y la variable manipulada B para la refrigeración.
- RATIOFAC es efectivo si LMN<0,0, por lo que la función de rango dividido debe estar definida en consecuencia: A para LMN>=0,0 y B para LMN<0,0.
- Para que la optimización fina de la refrigeración pueda comenzar durante la calefacción, las señales de calefacción y refrigeración deben poder ser efectivas en el proceso simultáneamente.

Procedimiento

Para determinar los parámetros PID óptimos en el punto de operación, proceda del siguiente modo:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Medición activada".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. En la lista "Modo" seleccione la entrada "Optimización fina de la refrigeración".
   El FM 355-2 se encuentra en disposición de optimización (PHASE = 1).
4. En el campo "Escalón de la variable manipulada" indique el valor en el que debe aumentarse o disminuirse la variable manipulada.
5. Haga clic en el símbolo "Iniciar optimización".
   Se inicia la optimización fina. Se muestra el estado de la optimización.

Resultado

En cuanto se encuentra un punto de inversión en la evolución del valor real, el FM 355-2 retorna de nuevo al modo de regulación (FASE 0).

A diferencia de lo que ocurre en la optimización de la calefacción, los parámetros antiguos del regulador no se guardan (el RATIOFAC antiguo ya se había guardado durante la optimización de la calefacción) y la fase 4 no se lleva a cabo.

STATUS_D permanece inalterado y sigue siempre refiriéndose a la optimización de la calefacción. También se conservan los juegos de parámetros PI y PID de la última optimización de la calefacción, de tal modo que se pueden cargar con LOAD_PID después de una optimización de la refrigeración.

Optimización paralela de los canales del regulador

Zonas vecinas con fuerte acoplamiento térmico

Si dos o más reguladores de un módulo regulan la temperatura p. ej. de una placa (p. ej. dos calefacciones y dos valores reales medidos con fuerte acoplamiento de calor), existe la posibilidad de realizar una optimización paralela para estos reguladores.

Procedimiento

Para optimizar simultáneamente varios canales, proceda del modo siguiente:

1. Parametrice los canales correspondientes en la zona A o en la zona B.
2. Inicie la optimización en uno de los canales. La optimización de los canales conexos se inicia entonces automáticamente.

Nota

En caso de excitación por escalón de consigna, el usuario debe garantizar la activación simultánea de los escalones de consigna. Los parámetros PID_ON y TUN_DLMN/TUN_CLMN sólo se pueden especificar para un canal. Los parámetros para los demás canales se deben predefinir a través de los correspondientes DBs de instancia de la instrucción FMT_PID antes de iniciar la optimización.

Cancelación de la optimización

Si aparece un error en uno de los canales (fase 0 o final de la fase 1), se cancela la optimización de todos los canales.

El usuario puede restablecer la optimización poniendo TUN_ON=FALSE en alguno de los canales implicados.

Si un canal pasa al modo de seguridad, también se desactiva la optimización de los canales implicados.

Sólo se muestra la información de estado del canal seleccionado. Encontrará las causas de error de los canales restantes en los correspondientes DBs de instancia.

Ventaja

Los reguladores implicados emiten su correspondiente magnitud manipulada de optimización hasta que todos los reguladores hayan abandonado la fase 2. De esta forma se evita que el regulador que primero finalice la optimización falsifique el resultado de optimización del resto de reguladores al modificar su magnitud manipulada.

⚠ PRECAUCIÓN

El hecho de alcanzar el 75% del escalón de consigna en uno de los canales implicados no finaliza la optimización (peligro de sobreoscilación). El modo automático se inicia cuando todos los reguladores implicados han finalizado su optimización.

Zonas vecinas con escaso acoplamiento térmico

En general se aplica la norma de que habría que optimizar tal y como se vaya a regular después. Si en el modo de producción las zonas se recorren juntas en paralelo, de forma que las diferencias de temperatura entre las zonas se mantienen iguales, durante la optimización también debería aumentarse correspondientemente el nivel de temperatura de las zonas vecinas.

Las diferencias de temperatura al principio del ensayo no son relevantes, porque se compensan mediante una calefacción inicial adecuada (→pendiente inicial = 0).

Optimización fina manual en modo de regulación

Introducción

Para conseguir una respuesta de regulación sin sobreoscilaciones se pueden tomar las medidas que se describen a continuación:

Adaptar la zona de regulación

En la optimización, el FM 355-2 determina un ancho de zona de regulación CON_ZONE y, si el tipo de sistema es correcto (tipos de sistema I y II), el regulador PID se activa: CONZ_ON = TRUE. En el modo de regulación es posible modificar la zona de regulación o desactivarla completamente (con CONZ_ON = FALSE).

Ninguna zona de regulación para el tipo de sistema III ni para los reguladores PI y paso a paso.

Activar la zona de regulación en procesos de orden superior (tipo de sistema III) no suele suponer ninguna ventaja, porque en ese caso la zona de regulación es mayor que el rango de regulación asequible con el 100% de la variable manipulada. Tampoco es ventajoso activar la zona de regulación para regulador PI.

Nota

Antes de activar a mano la zona de regulación, asegúrese de no ajustar un ancho de zona de regulación demasiado pequeño. Si el ancho ajustado para la zona de regulación fuera insuficiente, se producirán oscilaciones en las curvas de la variable manipulada y del valor real.

Atenuar continuamente la respuesta de regulación con PFAC_SP

La zona de regulación es la mejor posibilidad dinámica para conseguir una respuesta de regulación sin sobreoscilaciones. Utilícela siempre que resulte posible, es decir, con los reguladores PID para los tipos de sistema I y II.

Con los reguladores PI, tipo de sistema III, o en caso de escalón de consigna en la zona de regulación, la respuesta de regulación se puede atenuar con el parámetro PFAC_SP. Dicho parámetro determina qué proporción de la acción P será efectiva en casos de escalones de consigna.

PFAC_SP es preajustado por la optimización a 0,8, independientemente del tipo de sistema; el valor se puede modificar a continuación. Para limitar las sobreoscilaciones provocadas por escalones de consigna a aprox. un 2% (si los demás parámetros del regulador son correctos), resultan suficientes los valores siguientes para PFAC_SP:

	Tipo de sistema I	Tipo de sistema II	Tipo de sistema III
	Proceso de temperatura típico	Rango de transición	Proceso de temperatura de orden superior
PI	0.8	0.82	0.8
PID	0.6	0.75	0.96

Ajuste el factor predeterminado (0.8), especialmente en los casos siguientes:

- Tipo de sistema I con PID (0,8 → 0,6): Con PFAC_SP = 0,8, los escalones de consigna dentro de la zona de regulación siguen provocando una sobreoscilación del 18%.
- Tipo de sistema III con PID (0,8 → 0,96): Con PFAC_SP = 0,8 los escalones de consigna se amortiguan demasiado. Se regala un tiempo de ajuste claro.

Amortiguar los parámetros de regulación

Si en el lazo cerrado de regulación se producen oscilaciones, o si hay sobreoscilaciones después de escalones de consigna, se puede reducir la ganancia del regulador GAIN (p. ej., al 80% del valor original) y aumentar el periodo de integración TI (p. ej., al 150% del valor original). Si la variable manipulada analógica del regulador continuo se convierte en señales manipuladas binarias con un generador de impulsos, es posible que aparezcan pequeñas oscilaciones continuas debido a los efectos de cuantificación. Éstas se pueden eliminar ampliando la zona muerta DEADB_W.

Optimizar los parámetros PID manualmente

Requisitos

- Las exigencias planteadas al proceso se cumplen.
- Se ha creado y configurado un DB de instancia para el canal.
- Como origen del valor real está parametrizado "Instrucción".

Procedimiento

Para ajustar manualmente el regulador, proceda del modo siguiente:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Medición activada".
   Si aún no existe ninguna conexión online, ésta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. Introduzca los valores correspondientes para los parámetros siguientes:
   - Acción P
   - Ponderación del componente proporcional para escalones de consigna
   - Acción I
   - Acción D
   - Coeficiente DT1
   - Coeficiente calefacción/refrigeración
4. En el grupo "Optimización", haga clic en el símbolo "Enviar parámetros PID al FM".
5. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna" .
6. Introduzca una nueva consigna.

7. Desactive en caso necesario la casilla de verificación "Modo manual" y, en el grupo "Valores actuales", haga clic en el símbolo .
   El regulador trabaja con los nuevos parámetros y regula según la nueva consigna.
8. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
9. Repita los pasos del 3 hasta el 8, hasta que esté satisfecho/a con el resultado de la regulación.

Nota

FMT_PID.OP.UNDO_PAR = TRUE permite ajustar de nuevo los parámetros antiguos del regulador.

Cargar los parámetros optimizados en el dispositivo

Introducción

Para impedir que los parámetros del FM se sobrescriban con los datos del sistema en el siguiente arranque de la CPU, se puede proceder del modo siguiente:

- Tras cada arranque del FM 355-2, cargue en este con LOAD_PAR = TRUE los parámetros PID optimizados del DB de instancia del FMT_PID.
- Cargue los parámetros PID optimizados en el SDB.

Nota

Los juegos de parámetros PI, PID y SAVE no se pueden guardar en el SDB.

Procedimiento

Para cargar los parámetros en el SDB, proceda del siguiente modo:

1. Haga clic en el símbolo "Transfiera datos online al proyecto offline".
   Los parámetros PID se guardan en el ajuste de parámetros del FM.
2. Pare la CPU.
3. Haga clic en el símbolo "Cargar en dispositivo".

Guardar y volver a cargar los parámetros del regulador

Juego de parámetros guardado

En el FM 355-2 se dispone de un juego de parámetros guardado además del juego de parámetros efectivo. Para guardarlo y volverlo a cargar se emplea SAVE_PAR y UNDO_PAR en la estructura OP de la instrucción FMT_PID.

Así se puede, p. ej., guardar los parámetros actuales con SAVE_PAR = TRUE antes de introducir un cambio manual. Por medio de UNDO_PAR = TRUE se pueden volver a activar

para el regulador los parámetros de éste guardados en último lugar. Una vez finalizada la acción, FMT_PID desactiva SAVE_PAR o UNDO_PAR.

Si desea saber qué valores se encuentran en el juego de parámetros actual y cuáles hay en el juego de parámetros guardado, puede leerlos por medio de FMT_PID.READ_PAR = TRUE o FMT_TUN.READ_OUT = TRUE:

Parámetros actuales en la estructura del PAR del FMT_PID:	Parámetros guardados en los parámetros de salida del FMT_TUN:
PFAC_SP	SAV_PFAC
GAIN	SAV_GAIN
TI	SAV_TI
TD	SAV_TD
D_F	SAV_D_F
CON_ZONE	SAV_CONZ
RATIOFAC	SAV_RATI
CONZ_ON	SAV_CZON
P_SEL	SAV_PSEL

Nota

Al final de una optimización del regulador, los parámetros guardados se sobrescriben con los valores válidos antes de la optimización (excepción: en la optimización de la refrigeración se suprime este almacenamiento intermedio).

Cambio entre parámetros PI y PID

Después de una optimización, tanto los parámetros PI como los PID se guardan en el FM355-2. La carga de estos juegos de parámetros se lleva a cabo activando FMT_PID.OP.LOAD_PID. Si PID_ON es en la estructura PAR = TRUE, el juego de parámetros PID se copia en los parámetros efectivos del regulador; de lo contrario, se copia el juego de parámetros PI. Una vez finalizada la acción, FMT_PID desactiva LOAD_PID. El valor de PID_ON se mantiene, lo que resulta lógico pues no se trata de un parámetro de operación.

Si desea saber qué valores hay en el juego de parámetros PI o PID, puede averiguarlo a través de FMT_TUN. Leer READ_OUT = TRUE:

	Juego de parámetros PI	Juego de parámetros PID
Ganancia del regulador	PI_GAIN	PID_GAIN
Período de integración	PI_TI	PID_TI
Tiempo de acción derivativa	0.0	PID_TD

Nota

Tenga en cuenta lo siguiente:

- Los parámetros del regulador sólo se rescriben con UNDO_PAR o LOAD_PID si la ganancia del regulador (SAV_GAIN, PID_GAIN o PI_GAIN) es distinta de cero:
- Con ello se tiene en cuenta el caso de que aún no se haya llevado a cabo ninguna optimización o de que falte algún parámetro PID. Si eran, p. ej., PID_ON = TRUE y PID_GAIN = 0.0, PID_ON se pone a FALSE y los parámetros PI se copian.
- D_F, PFAC_SP se preajustan con la optimización. Seguidamente pueden ser modificados por el usuario. LOAD_PID no modifica este parámetro.
- La zona de regulación siempre se calcula de nuevo con LOAD_PID (CON_ZONE = 250/GAIN o 375/GAIN si RATIOFAC <> 0.0), incluso si CONZ_ON = FALSE.

Conmutación sin discontinuidad con LOAD_PID y UNDO_PAR

La conmutación se efectúa de manera que la suma del componente proporcional y el componente integral permanece constante. Al conmutar de PID a PI se produce un salto en la magnitud de ajuste, ya que el componente diferencial desaparece. El paso de PI a PID no provoca discontinuidad alguna en un principio. No obstante, al cabo de algunos ciclos, el componente diferencial provoca un cambio rápido.

Realizar medición del tiempo de posicionamiento del motor

Introducción

Para conseguir un buen resultado de la regulación, el tiempo de posicionamiento del motor debe ser lo más preciso posible. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento.

Si utiliza actuadores con realimentación de posición o con señales de tope, es posible medir el tiempo de posicionamiento del motor durante la puesta en servicio. Si no se dispone de una realimentación de posición ni de señales de tope, no es posible medir el tiempo de posicionamiento del motor.

Requisitos

- El valor parametrizado del tiempo de posicionamiento del motor debe equivaler como mínimo a la mitad del tiempo de posicionamiento real. El actuador se desplaza en un sentido durante el doble del tiempo de posicionamiento como máximo.
- En los actuadores con realimentación de posición analógica, la realimentación de posición debe estar interconectada.
- En los actuadores con señales de tope, las señales de tope deben estar interconectadas.

Actuadores con realimentación de posición analógica

Para medir el tiempo de posicionamiento del motor con realimentación de posición, proceda del siguiente modo:

1. Elija el DB instancia para el que se debe medir el tiempo de posicionamiento del motor.
2. Active la casilla de verificación "Utilizar realimentación de posición".
3. En el campo de entrada "Posición de destino" indique hacia dónde desea mover el elemento final de control.
   La realimentación de posición actual (posición de inicio) se muestra. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se desplaza hasta la posición de destino desde la posición de inicio. La medición del tiempo se inicia de inmediato y finaliza en cuanto el elemento final de control ha alcanzado la posición de destino. El tiempo de posicionamiento del motor se calcula según la fórmula:

Tiempo de posicionamiento del motor = (Límite superior del valor de salida - Límite inferior del valor de salida) x Tiempo de medición /VALOR ABSOLUTO(Posición de destino - Posición de inicio).

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se muestra en el campo "Tiempo de actuación medido". Una vez finalizada la medición del tiempo de posicionamiento se conecta el modo manual de las señales de posicionamiento (OP.LMNS_ON = TRUE). El elemento final de control deja de moverse.

Actuadores con señales de tope

Para medir el tiempo de posicionamiento del motor con señales de tope, proceda del siguiente modo:

1. Elija el DB instancia para el que se debe medir el tiempo de posicionamiento del motor.
2. Active la casilla de verificación "Utilizar señales de tope del actuador".
3. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se mueve primero hasta el tope inferior, después, hasta el tope superior y, de nuevo, hasta el tope inferior. La medición del tiempo se inicia cuando el actuador ha alcanzado el tope inferior y finaliza cuando el actuador alcanza por segunda vez el tope inferior. El tiempo medido dividido entre dos da lugar al tiempo de posicionamiento del motor.

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se muestra en el campo "Tiempo de actuación medido". Una vez finalizada la medición del tiempo de posicionamiento se conecta el modo manual de las señales de posicionamiento (OP.LMNS_ON = TRUE). El elemento final de control deja de moverse.

Cargar el tiempo de ajuste medido

1. Para guardar en el proyecto el tiempo de posicionamiento del motor medido, haga clic en el símbolo "Transfiere datos online al proyecto offline".
   El tiempo de ajuste medido se guarda en el ajuste de parámetros del FM.
2. Para que los cambios surtan efecto en el arranque siguiente de la CPU, haga clic en el icono "Cargar en dispositivo".

Cancelar medición del tiempo de posicionamiento

Si cancela la medición del tiempo de posicionamiento, el modo manual de las señales de posicionamiento se conecta (OP.LMNS_ON = TRUE) y el modo manual con especificación de la variable manipulada se desconecta (OP.LMN_REON = FALSE). El elemento final de control deja de moverse.

Trabajar con el registrador de curvas

Observar

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Puesta en servicio".
2. Navegue en la vista de configuración hasta "Registrador de curvas".
3. Seleccione el DB de instancia que desee observar. Tras seleccionarlo se muestra el canal al que está asignado el DB de instancia.
4. Defina el tiempo de actualización con el que debe registrar el registrador de curvas.
5. Haga clic en "Medición activada" para iniciar la medición.
6. Haga clic en "Medición desactivada" para finalizar la medición.

Procedimiento

Para adaptar la escala, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Puesta en servicio".
2. Navegue en la vista de configuración hasta "Registrador de curvas".

3. Para modificar el eje de tiempo:
   - Haga clic en el botón de flecha junto a la fecha, para abrir un calendario. Seleccione una fecha en el calendario. Para bloquear la selección, haga clic en el candado azul. Esta posibilidad existe tanto para la fecha de inicio como para la fecha de fin.
   - Haga clic con el puntero del ratón en la escala de tiempo. El puntero del ratón se representa como flecha a la izquierda/derecha. Mantenga el puntero del ratón presionado y arrástrelo hacia la izquierda o derecha.
     Si el tiempo de inicio o fin está bloqueado, se modifica el escalado del tiempo.
     Si no está bloqueado ni el tiempo de inicio ni el de fin, se modifica el período indicado; el escalado se mantiene igual.
4. Para modificar el eje de tiempo:
   - Haga clic en el campo de entrada para el valor superior o el valor inferior mostrado. Introduzca ahí el valor deseado. Para bloquear la selección, haga clic en el candado azul. Esta posibilidad existe tanto para el valor superior como también para el inferior.
   - Haga clic con el puntero del ratón en la escala de valores. El puntero del ratón se representa como flecha arriba/abajo. Mantenga el puntero del ratón presionado y arrástrelo hacia arriba o abajo.
     Si el valor superior o inferior está bloqueado, se modifica la escala del eje de valores.
     Si no está bloqueado el valor superior ni el valor inferior, se modifica el rango indicado; la escala se mantiene igual.

FM 355-2: Diagnóstico a través de indicadores LED

Indicación de errores mediante el LED de error de grupo

Introducción

Si el LED rojo de error de grupo se enciende, significa que se ha producido un error en el módulo (error interno) o en las conexiones (error externo).

Si el LED amarillo parpadea, el firmware se borra. Este estado sólo puede aparecer si existe un defecto en el hardware o si el proceso de carga del firmware se cancela.

¿Qué errores se muestran?

Cuando se ilumina el LED de error de grupo se señalan los errores siguientes:

Tipo de error	Aviso de diagnóstico	Causa posible	Solución
Errores internos	Módulo defectuoso	Error de hardware	Cambie el módulo
	Vigilancia de tiempo (watch dog) activado	Error de hardware	Cambie el módulo
	Contenido EEPROM no válido	Fallo de la tensión de alimentación al parametrizar	Vuelva a parametrizar el módulo
Errores externos	Parámetro erróneo en el módulo	Se han transferido parámetros incorrectos al módulo	Vuelva a parametrizar el módulo

Tipo de error	Aviso de diagnóstico	Causa posible	Solución
	Error en las entradas o las salidas analógicas	Error de hardware en la entrada analógica	Cambie el módulo
		Rotura de hilo en la entrada analógica	Solucione la rotura de hilo
		Rebase por defecto del rango de medición en la entrada analógica	Revise la señal de medición
		Rebase por exceso del rango de medición en la entrada analógica	Revise la señal de medición
		Rotura de hilo en la salida analógica	Solucione la rotura de hilo
		Cortocircuito en la salida analógica	Solucione el cortocircuito
	Falta tensión auxiliar externa	Fallo en la alimentación 24 V	Restablezca la alimentación de 24 V

Alarma de diagnóstico en caso de error

Todos los errores pueden disparar una alarma de diagnóstico si ésta se ha habilitado en la pantalla de parametrización correspondiente. En los registros de diagnóstico DS0 y DS1 puede ver qué error ha provocado que se encienda el LED. La asignación de los registros de diagnóstico DS0 y DS1 está descrita en el siguiente apartado.

Disparo de alarmas de diagnóstico

¿Qué es una alarma de diagnóstico?

Si el programa de usuario reacciona a un error interno o externo, es posible parametrizar una alarma de diagnóstico. El programa cíclico de la CPU se interrumpe y se llama la alarma de diagnóstico OB, OB 82.

¿Qué eventos pueden disparar una alarma de diagnóstico?

La siguiente lista muestra qué eventos pueden disparar una alarma de diagnóstico:

- Módulo no parametrizado o errores en la parametrización
- Módulo defectuoso
- Rotura de hilo en entradas analógicas (sólo 4 a 20 mA)
- Rebase por exceso y por defectos en entradas analógicas
- Rotura de carga y cortocircuito en salidas analógicas

Habilitación de la alarma de diagnóstico

La ficha Parámetros básicos permite bloquear o habilitar la alarma de diagnóstico para el módulo.

La alarma de diagnóstico está bloqueada de forma predeterminada.

Respuestas a un evento que dispara una alarma

Cuando se produce un evento que puede disparar una alarma de diagnóstico ocurre lo siguiente:

- La información de diagnóstico se registra en los registros de diagnóstico DS0 y DS1 del módulo.
- El LED de error de grupo se enciende.
- Se llama al OB de alarma de diagnóstico (OB 82).
- El registro de diagnóstico DS0 se graba en la información de arranque del OB de alarma de diagnóstico.
- Si no hay ningún defecto de hardware, el módulo sigue regulando.

La CPU pasa a modo STOP si no se ha programado el OB 82.

Registro de diagnóstico DS0 y DS1

La información que indica qué evento ha provocado una alarma de diagnóstico se guarda en los registros de diagnóstico DS0 y DS1. El registro de diagnóstico DS0 consta de cuatro bytes; DS1 consta de 16 bytes, siendo los cuatro primeros idénticos a DS0.

Registros de diagnóstico DS0 y DS1

Leer el registro desde el módulo

El registro de diagnóstico DS0 se transfiere automáticamente a la información de arranque en la llamada del OB de diagnóstico. Allí, estos cuatro bytes se guardan en los datos locales (bytes 8-11) del OB 82.

El registro de datos DS1 (y, con ello, también el contenido del DS0) puede leerse con el FB 55 "FMT_DS1" del módulo. Esta lectura sólo tiene sentido si en el DS0 se notifica un error en un canal.

El FB 55 "FMT_DS1" debe llamarse en el mismo OB que un posible FB FMT_PID existente (por ejemplo, OB 35). Esto se consigue con la medida siguiente: Ajuste el bit READ_DS1 mientras se procesa el OB 82. El FMT_DS1 del OB 35 lee a continuación el registro de diagnóstico DS1.

¿Cómo aparecen los textos de diagnóstico en el búfer de diagnóstico?

Si se desea registrar el aviso de diagnóstico en el búfer de diagnóstico, se debe llamar la instrucción "WR_REC" "Introducir avisos específicos de usuario en el búfer de diagnóstico" dentro del programa de usuario. En cada caso, el número de evento del aviso de diagnóstico se especifica en el parámetro de entrada EVENTN. La alarma se introduce en el búfer de diagnóstico con x=1 como entrante y con x=0 como saliente. El búfer de diagnóstico contiene los textos de diagnóstico correspondientes en la columna de "Significado", así como el momento en que se produjo la entrada.

Asignación del registro de diagnóstico DS0 y de la información de arranque

La tabla siguiente muestra la asignación del registro de diagnóstico DS0 en la información de arranque. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Tabla 11-9 Asignación del registro de diagnóstico DS0

Byte	Bit	Significado	Nota	Nº de evento
0	0	Avería en módulo	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Error interno	Se activa en todos los errores internos: • Vigilancia de tiempo activada • Error EPROM • Error ADC/DAC • Error de hardware en la entrada analógica	8:x:01
	2	Error externo	Se activa en todos los errores externos: • Falta tensión auxiliar externa • Error de parametrización • Rotura de hilo en la entrada analógica (sólo en el margen 4 a 20 mA) • Rebase por defecto del rango de medición en la entrada analógica • Rebase del rango de medición en la entrada analógica • Rotura de carga de la salida analógica • Cortocircuito en la salida analógica	8:x:02
	3	Fallo en un canal	Consultar el DS1, a partir del byte 7, para más detalles	8:x:03
	4	Falta tensión auxiliar externa	Fallo en la alimentación 24 V del FM 355-2	8:x:04
	6	sin usar	-	-
	7	Error de parametrización	El módulo no puede utilizar un parámetro. Motivo: Parámetro desconocido o combinación de parámetros no permitida. Consulte el menú Sistema de destino > Mostrar error de parametrización.	8:x:07
1	0 ... 3	Clase de módulo	Siempre tiene asignado un 8.	-
	4	Diagnóstico específico de canal	Se activa cuando el módulo puede proporcionar información del canal adicional y hay un error de canal (consulte DS1 bytes 7 a 12)	-
2	3	Vigilancia de tiempo (watch dog) activado	Error de hardware	8:x:33
3	2	Error EPROM	Módulo defectuoso	8:x:42
	4	Error ADC/DAC	Módulo defectuoso	8:x:44

Registro de diagnóstico DS1

El registro de diagnóstico DS1 se compone de 16 bytes. Los 4 primeros bytes son idénticos a los del registro de diagnóstico DS0. La siguiente tabla muestra la asignación de los bytes restantes. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Registro de diagnóstico DS1 del FM 355-2

Tabla 11-10 Asignación de los bytes del 4 a 12 del registro de diagnóstico DS1 del FM 355-2

Byte	Bit	Significado	Nota		Nº de evento
4	0 ... 7	Tipo de canal	Siempre tiene asignado 75H		-
5	0 ... 7	Longitud de la información de diagnóstico	Siempre tiene asignado un 8.		-
6	0 ... 7	Cantidad de canales	Siempre tiene asignado un 5 (4 reguladores + 1 canal de referencia)		-
7	0 ... 4	Vector de error de canal	Cada bit tiene asignado un canal (0 a 3, 4 para el canal de referencia).		-
8	0	Error de hardware en la entrada analógica	Diagnóstico específico del canal 0		8:x:B0
	1	sin usar			8:x:B1
	2	Rotura de hilo en la entrada analógica (sólo en el margen 4 a 20 mA)			8:x:B2
	3	sin usar			8:x:B3
	4	Rebase por defecto del rango de medición en la entrada analógica			8:x:B4
	5	Rebase del rango de medición en la entrada analógica			8:x:B5
	6	Rotura de hilo en la salida analógica		Sólo en la salida de intensidad del regulador K	8:x:B6
	7	Cortocircuito en la salida analógica		Sólo en la salida de tensión del regulador K	8:x:B7
9	0 ... 7	véase byte 8	Diagnóstico específico del canal 1		véase arriba
10	0 ... 7	véase byte 8	Diagnóstico específico del canal 2		véase arriba
11	0 ... 7	véase byte 8	Diagnóstico específico del canal 3		véase arriba
12	0 ... 5	véase byte 8	Diagnóstico para el canal de referencia		véase arriba

Información adicional

Para obtener más información sobre el parámetro "RET_REALU", consulte Lista de los avisos RET_REALU (Página 10339) y el área de instrucciones avanzadas del sistema de ayuda.

Fallo del transductor de medida

Averías en los transductores de medida

El FM 355-2 puede detectar los siguientes fallos en el transductor de medida:

- Rebase por defecto del rango de medición
- Rebase del margen de medición
- Rotura de hilo (no se muestra en todos los márgenes de medición)

Si se produce una de estas averías en el registro de diagnóstico DS0 se define el bit de error de grupo "Error externo" y, en el registro de diagnóstico DS1, bits de error específicos del canal (consulte las tablas anteriores). Cuando desaparecen estas averías los bits correspondientes se desactivan.

La siguiente tabla muestra los límites de los diferentes márgenes de medición en los que los bits de error se activan o desactivan:

Rango de medida	Bit de error Rebase por defecto del rango de medición en ...	Bit de error Rebase del rango de medición en ...	Bit de error Mostrar rotura de hilo
	DS1: Byte 10 a 26, Bit 4	DS1: Byte 10 a 26, Bit 5	DS1: Byte 10 a 26, Bit 2
0 a 20 mA	< - 3,5 mA	> 23,5 mA	-
4 a 20 mA	Bit de error = 1 a < 3,6 mA Bit de error = 0 a > 3,8 mA	> 22,8 mA	Bit de error = 1 a < 3,6 mA Bit de error = 0 a > 3,8 mA
0 a 10 V	< - 1,175 V	> 11,75 V	-
Pt 100 (-200 a 850 °C) (-328 a 1562 °F)	< 30,82 mV	> 650,46 mV	-
Pt 100 (-200 a 556 °C) (-328 a 1032 °F)	< 30,82 mV	> 499,06 mV	-
Pt 100 (-200 a 130 °C) (-328 a 264 °F)	< 30,82 mV	> 254,12 mV	-
Termoelemento tipo B	< 0 mV	> 13,81 mV	-
Termoelemento tipo E (-270,0 a 1000,03 °C)	< -9,84 mV	> 76,36 mV	-
Termoelemento tipo J	< - 8,1 mV	> 69,54 mV	-
Termoelemento tipo K	< - 6,45 mV	> 54,88 mV	-
Termoelemento tipo R	< - 0,23 mV	> 21,11 mV	-
Termoelemento tipo S	< - 0,24 mV	> 18,7 mV	-
Termopar libre	< Valor de entrada inferior de la línea poligonal	> Valor de entrada superior de la línea poligonal	-

Lista de los avisos RET_REALU

Avisos RET_REALU

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
7000	28672	-32624	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.
7001	28673	-32624	Primera llamada con REQ=1: Transferencia de datos lanzada; BUSY tiene el valor 1.
7002	28674	-32624	Llamada intermedia (REQ irrelevante). Transferencia de datos ya activa; BUSY tiene el valor 1.
8090	32912	-32624	La dirección base lógica indicada no es válida: No existe ninguna asignación en el SDB1/SDB2x, o no se trata de una dirección base.
80A0	32928	-32608	Acuse negativo al leer el módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2	32930	-32606	Error de protocolo DP en capa 2
80A3	32931	-32605	Error de protocolo de DP con User-Interface/User
80A4	32932	-32604	Fallo de comunicación en el bus K
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo teórico.
80C0	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1	32961	-32575	Los datos de una petición de escritura semejante todavía no se han procesado en el módulo.
80C2	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados en estos momentos.
80C4	32964	-32572	Error de comunicación
80C5	32965	-32571	La periferia descentralizada no está disponible.
80C6	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (petición de escritura).
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (petición de escritura).
853A	34106	-31430	DB de parámetros no presente. (petición de escritura).
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras aparecer un error (petición de escritura).
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (petición de lectura).

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU. No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (petición de lectura).
873A	34618	-30918	DB de parámetros no presente. (petición de lectura).
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras aparecer un error. (petición de lectura).
80ff	33023	-32513	Indicación de índice incorrecta en el bloque FMT_PAR
Los errores 80A2..80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario. Los avisos con la estructura 7xxx indican situaciones temporales de la comunicación.			

11.3.5.7 Utilizar el FM 455

Configuración del FM 455

Configuración hardware

Procedimiento

Para poder configurar el hardware debe haber creado previamente un proyecto donde pueda guardar la parametrización. A continuación se explican únicamente los pasos más importantes.

1. Inicie el TIA portal y abra la configuración de dispositivos.
2. Seleccione un rack.
3. Seleccione un módulo de regulación del catálogo de módulos.
4. Arrastre el módulo de regulación hasta la posición correspondiente del rack.
5. Anote la dirección de entrada del módulo a partir de la tabla de configuración, p. ej. 512. El valor leído se muestra en formato decimal.

Parametrizar el FM 455

Parámetros del módulo

Procedimiento

A través de la parametrización se configuran los parámetros del módulo.

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo. Seleccione en la ventana de inspección la pestaña "Propiedades".
   Resultado: Las propiedades se muestran en la ventana de inspección.
2. Configure los parámetros básicos del módulo y el resto de parámetros.

3. Configure los demás parámetros del bloque en la vista de configuración.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".
5. Compile el proyecto con "Editar > Compilar".
6. Cargue los datos de parametrización en el estado de STOP de la CPU con "Online > Cargar en dispositivo..."
   Resultado: Los datos se encuentran en la memoria de la CPU y se transfieren desde ésta directamente al módulo.
7. Arranque la CPU.

Parámetros de la entrada analógica

Evaluación de la entrada analógica

Procedimiento

Para seleccionar un tipo de sensor proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Evaluación de la entrada analógica".
3. Defina los siguientes parámetros:
   - Resolución entrada analógica
   - Tipo de sensor entrada analógica
     La opción "Vigilancia de rotura de hilo" se habilita para diversos tipos de sensor. Se puede activar la opción según requerimiento.
   - Temperatura de referencia
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Funciones de filtro

Procedimiento

Para definir las funciones de filtro proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Funciones de filtro".

3. Defina los siguientes parámetros:
   - Active la opción "Filtro de señal" para filtrar por una constante de tiempo.
     Si la opción "Filtro de señal" se encuentra seleccionada se puede activar la constante de tiempo del filtro.
   - Constante de tiempo del filtro
     Se puede activar solamente si la opción "Filtro de señal" se encuentra activada.
   - Active la opción "Filtro de raíz cuadrada" para linealizar la señal del sensor.
   - Active la opción "Filtro de polígono" para linealizar la característica de la señal del sensor.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Filtro de señal

Filtro de primer orden cuya respuesta temporal viene determinada por la constante de tiempo.

Filtro de raíz cuadrada

Su finalidad es linealizar señales de sensores que están en relación cuadrada con la magnitud de proceso medida, en las cuales el valor real obedece a una magnitud física.

Filtro de polígono

Su finalidad es linealizar características de sensores.

Los 13 nodos seleccionables se pueden seleccionar en:

- mA en entrada de corriente
- mV en entrada de tensión

Normalización

Procedimiento

Para convertir la señal de entrada en una unidad física proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Entrada analógica > Entrada [n°] > Normalización".
3. Defina los siguientes parámetros:
   - Valor superior
   - Valor inferior
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Normalización

Para convertir la señal de entrada en otra unidad física mediante interpolación lineal entre el valor inicial (valor inferior) y el valor final (valor superior).

Error de regulación

Realizar un regulador de valor fijo o en cascada

Regulador de valor fijo o en cascada

En las estructuras de regulación 'Regulador de valor fijo o en cascada' el valor real efectivo es idéntico al valor real A.

Seleccionar el tipo de regulador

Para realizar un regulador de valor fijo o en cascada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°]".
3. En el área "Tipo de regulador" seleccione la opción "Regulador de valor fijo o en cascada".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para acondicionar la consigna proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Active los parámetros correspondientes en las diferentes áreas:
   - Origen de la consigna
     Seleccione el origen de la consigna.
   - Consigna de seguridad
     Ajuste aquí la consigna de seguridad. Defina asimismo la reacción a un fallo de la CPU y el comportamiento en arranque.
   - Rampa de consigna
     Defina el tiempo de subida de rampa.
   - Límite de consigna
     Seleccione aquí los límites superior e inferior de consigna.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar origen de la señal para el valor real A

Para seleccionar el origen de la señal para el valor real A proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione el origen correspondiente en el área "Variable manipulada". Se proporcionan las siguientes posibilidades de selección:
   - Cero
   - Entrada analógica acondicionada
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar un regulador de 3 componentes

Regulador de 3 componentes

En la estructura de regulación 'Regulador de 3 componentes', el valor real efectivo se define a través de la suma de los tres valores reales A, B y C, y mediante adición de un Offset ajustable. Los valores reales B y C se pueden evaluar adicionalmente mediante factores.

Seleccionar el tipo de regulador

Para materializar un regulador de tres componentes, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°]".
3. En el área "Tipo de regulador" seleccione la opción "Regulador de 3 componentes".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para ajustar cómo debe acondicionarse la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Origen de la consigna" el origen de referencia de la consigna.
4. Especifique un valor en el área "Consigna de seguridad", el cual será tomado como consigna de seguridad en caso de fallo de la CPU y durante el arranque.
5. Especifique el comportamiento del módulo de regulación válido como "Reacción a fallo de la CPU" y para el "Comportamiento en arranque". Las posibilidades de selección son:
   - Consigna = última consigna válida: al producirse el evento se asigna a la consigna la última consigna válida. Es decir, se mantiene la última consigna válida.
   - Consigna = consigna de seguridad: al producirse el evento se asigna a la consigna la consigna de seguridad. De esta forma se puede realizar un funcionamiento de emergencia seguro.

6. Especifique en el área "Rampa de consigna" el tiempo de rampa que se debe mantener al modificarse la consigna, para evitar una sobreoscilación.
7. Especifique en el área "Límite de consigna" un límite superior e inferior para la consigna.
8. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar el origen de la señal para los valores reales A, B y C

Para especificar los orígenes de la señal para los valores reales A, B y C proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Variable manipulada" el origen de la señal correspondiente para cada valor real. Puede seleccionar si no utiliza ninguna señal ("cero") como origen o si utiliza una de las entradas analógicas acondicionadas.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Registrar factores para los valores reales B y C, así como para el Offset

Para especificar los orígenes de la señal para los valores reales A, B y C proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Especifique en el área "Sumar" el factor de ponderación requerido para los valores actuales B y C.
4. Especifique el "Offset" necesario que se debe adicionar a los valores reales.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar un regulador de relación o mezcla

Seleccionar el tipo de regulador

Para materializar un regulador de relación o de mezcla, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°]".
3. En el área "Tipo de regulador" seleccione la opción "Regulador de relación o mezcla".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Acondicionamiento de la consigna

Para ajustar cómo debe acondicionarse la consigna, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Origen de la consigna" el origen de referencia de la consigna.
4. Especifique un valor en el área "Consigna de seguridad", el cual será tomado como consigna de seguridad en caso de fallo de la CPU y durante el arranque.
5. Especifique el comportamiento del módulo de regulación válido como "Reacción a fallo de la CPU" y para el "Comportamiento en arranque". Las posibilidades de selección son:
   - Consigna = última consigna válida: al producirse el evento se asigna a la consigna la última consigna válida. Es decir, se mantiene la última consigna válida.
   - Consigna = consigna de seguridad: al producirse el evento se asigna a la consigna la consigna de seguridad. De esta forma se puede realizar un funcionamiento de emergencia seguro.
6. Especifique en el área "Rampa de consigna" el tiempo de rampa que se debe mantener al modificarse la consigna, para evitar una sobreoscilación.
7. Especifique en el área "Límite de consigna" un límite superior e inferior para la consigna.
8. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar origen de la señal para los valores reales A y D

Para especificar los orígenes de la señal para los valores reales A y D proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Variable manipulada" el origen de la señal correspondiente para cada valor real. Puede especificar no utilizar ninguna señal ("cero") como origen o utilizar una de las entradas analógicas acondicionadas, para el valor real D puede seleccionar también el valor manipulado de uno de los reguladores.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Registrar multiplicador

Para especificar los orígenes de la señal para los valores reales A y D proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. Seleccione en el área "Multiplicar" el Offset deseado.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

En el regulador de relación, el valor real A se utiliza como magnitud de regulación y el valor real D como magnitud de relación. La entrada de consigna sirve de factor de relación. Ésta

se acondiciona para obtener una consigna efectiva mediante multiplicación por el valor real D y adición de un offset ajustable. Si se desactiva el valor real D, solamente se suma el Offset a la consigna.

Ajustar la generación de alarmas

Procedimiento

Para ajustar la generación de alarma, proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Error de regulación".
3. En el área "Generar alarma" seleccione las opciones necesarias:
   - En el área "Avisador de valor límite de" seleccione la señal a la que se debe aplicar la alarma.
   - En "Alarma límite superior" especifique el valor a partir del cual se disparará la alarma al alcanzarse el límite superior.
   - En "Advertencia límite superior" especifique el valor a partir del cual se disparará una advertencia cuando se produzca una aproximación al límite superior.
   - En "Advertencia límite inferior" especifique el valor a partir del cual se disparará una advertencia cuando se produzca una aproximación al límite inferior.
   - En "Alarma límite inferior" especifique el valor a partir del cual se disparará la alarma al alcanzarse el límite inferior.
   - Para evitar que centelleen los indicadores de vigilancia se puede configurar una histéresis en "Histéresis".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Algoritmo de regulación

Realizar una regulación P

Procedimiento

Para realizar una regulación P proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "P", en "Modo de operación".
   Todas las opciones parametrizables para una regulación P se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación PI

Procedimiento

Para realizar una regulación PI proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "PI", en "Modo de operación".
   Todas las opciones parametrizables para una regulación PI se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación I

Procedimiento

Para realizar una regulación I proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "I", en "Modo de operación".
   Todas las opciones parametrizables para una regulación I se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación PID

Procedimiento

Para realizar una regulación PID proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "PID", en "Modo de operación".
   Todas las opciones parametrizables para una regulación PID se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar una regulación PD

Procedimiento

Para realizar una regulación PD proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Regulador PID".
4. En el área "General" seleccione la opción "PD", en "Modo de operación".
   Todas las opciones parametrizables para una regulación PD se activan, el resto de opciones se desactivan.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Realizar un regulador de temperatura con lógica difusa

Procedimiento

Para llevar a cabo una regulación de temperatura por lógica difusa, proceda del modo siguiente:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Tipo" seleccione la opción "Módulo de regulación de temperatura por lógica difusa".
   Todas las opciones parametrizables para módulo de regulación de temperatura por lógica difusa se activan, el resto de opciones se desactivan.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Suprimir el nivel de ruido del error de regulación

Procedimiento

Para suprimir el nivel de ruido del error de regulación proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. Especifique en el área "General" el ajuste "Ancho de zona muerta".
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Control anticipativo

Procedimiento

Para conectar una variable perturbadora para control anticipativo proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "General" seleccione de la lista "Origen de la perturbación" el origen de la variable perturbadora.
   Seleccione la entrada "Cero" para no conectar ninguna variable perturbadora para control anticipativo.
4. En "Factor de perturbación" especifique un valor de ponderación para la variable perturbadora.
   Un valor < 0 debilita la perturbación, un valor > 0 amplifica la perturbación. La perturbación permanece invariable si se especifica el valor 1.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Configurar la acción P por rama de realimentación

Procedimiento

Para realizar un regulador de valor fijo o en cascada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Acción P" seleccione la opción "Por rama de realimentación".
   Si la opción está activada, la acción P queda configurada por rama de realimentación.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Debilitar la acción D cuando se producen modificaciones en la consigna

Seleccionar el tipo de regulador

Para realizar un regulador de valor fijo o en cascada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. En el área "Acción D" seleccione con opción "Acción D" el origen de la acción D.
4. En el área "Factor derivativo" especifique el factor para la acción D.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Invertir el efecto del regulador

Principio

La inversión del regulador, cambiando la asignación de "error de regulación ascendente = magnitud manipulada ascendente" a "error de regulación ascendente = magnitud manipulada descendente", se realiza ajustando una ganancia proporcional (GAIN).

Procedimiento

Para invertir el efecto del regulador proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Algoritmo de regulación".
3. Especifique en el área "General" una ganancia proporcional con signo inverso.
   Si se encuentra especificada una ganancia proporcional positiva y se desea invertir el efecto del regulador, especifique una ganancia proporcional negativa.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Salida del regulador

Realizar un regulador continuo

Para realizar un regulador continuo se requiere un módulo analógico.

Procedimiento

1. Aplicar la variable manipulada externa (Página 10353)
2. Ajustar la variable manipulada de seguridad (Página 10354)
3. Activar el modo de seguimiento (Página 10353)
4. Limitar la variable manipulada (Página 10354)
5. Controlar dos actuadores - rango partido (Página 10355)
6. Seleccionar la señal de salida y el tipo de señal (Página 10356)

Regulador paso a paso sin realimentación de posición

Para realizar un regulador paso a paso sin realimentación de posición se requiere un módulo digital.

Procedimiento

1. Seleccionar la salida del regulador Regulador paso a paso sin realimentación de posición
2. Aplicar la variable manipulada externa (Página 10353)

3. Ajustar la variable manipulada de seguridad (Página 10354)
4. Formador de impulsos de la regulación paso a paso (Página 10355)

Realizar un regulador paso a paso con realimentación de posición

Para realizar un regulador paso a paso con realimentación de posición se requiere un módulo digital.

Procedimiento

1. Seleccionar la salida del regulador Regulador paso a paso con realimentación de posición
2. Aplicar la variable manipulada externa (Página 10353)
3. Activar el modo de seguimiento (Página 10353)
4. Ajustar la variable manipulada de seguridad (Página 10354)
5. Limitar la variable manipulada (Página 10354)
6. Ajustar la realimentación de posición (Página 10353)
7. Formador de impulsos de la regulación paso a paso (Página 10355)

Realizar un regulador de impulsos

Para realizar un regulador de impulsos se requiere un módulo digital.

Procedimiento

1. Seleccionar la salida del regulador Regulador de impulsos
2. Aplicar la variable manipulada externa (Página 10353)
3. Activar el modo de seguimiento (Página 10353)
4. Ajustar la variable manipulada de seguridad (Página 10354)
5. Limitar la variable manipulada (Página 10354)
6. Acondicionar la señal analógica para la conversión en señal digital
7. Convertir una señal analógica en una señal digital (Página 10356)

Parametrizar las funciones de la salida del regulador

Aplicar la variable manipulada externa

Procedimiento

Para aplicar la variable manipulada externa proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Seleccione en el área "Magnitud manipulada externa" la opción con la debe se debe activar la magnitud manipulada externa. Para ello puede seleccionar las siguientes opciones:
   – Activación del bloque de función.
   – Activación por el bloque de función o por una de las entradas digitales.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Activar el modo de seguimiento

Procedimiento

Para activar el modo de seguimiento proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Seleccione en el área "Modo de seguimiento" la opción que debe activar la magnitud manipulada externa. Para ello puede seleccionar las siguientes opciones:
   – Activación por el bloque de función.
   – Activación por el bloque de función o una de las entradas digitales.
4. Seleccione en el área "Modo de seguimiento" el origen de la señal a aplicar para el seguimiento. Para ello puede seleccionar las siguientes opciones:
   – Cero, con lo cual no se emplea ningún origen.
   – Una de las entradas analógicas.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Ajustar la realimentación de posición

Procedimiento

Para ajustar la realimentación de posición proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida digital > Salida [n°]".

3. Seleccione en el área "Entrada de realimentación de posición" la opción "Origen" para especificar la señal de realimentación de posición.
   - Cero: no se produce ninguna realimentación de posición.
   - Una de las entradas analógicas acondicionadas.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Ajustar la variable manipulada de seguridad

Procedimiento

Para ajustar la variable manipulada de seguridad proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Especifique en el área "Variable manipulada de seguridad" el valor a utilizar como variable manipulada de seguridad.
4. Seleccione en el área "Activación" la opción que debe activar la magnitud manipulada externa. Para ello puede seleccionar las siguientes opciones:
   - Activación del bloque de función.
   - Activación del bloque de función o de una de las entradas digitales.
5. Seleccione en "Reacción al arrancar el módulo" la opción a ejecutar al arrancar el módulo. Para ello puede seleccionar las siguientes opciones:
   - Lazo cerrado: el módulo entra inmediatamente en este modo al arrancar.
   - Variable manipulada de seguridad: primero se regula a la variable manipulada de seguridad.
6. Seleccione en el área "Reacción si falla el transductor de medida del valor real A" la opción a ejecutar si falla el transductor de medida del valor real A.
7. Seleccione en el área "Reacción si falla el transductor de medida de una entrada" la opción a ejecutar si falla el transductor de medida de una entrada.
8. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Limitar la variable manipulada

Procedimiento

Para limitar la variable manipulada proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".

3. Especifique en el área "Límite de señal de salida" los siguientes valores:
   – Valor superior: representa el límite superior para la señal de salida.
   – Valor inferior: representa el límite interior para la señal de salida.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Controlar dos actuadores - rango partido

Procedimiento

Para controlar dos actuadores proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Regulador > Canal [n°] > Salida del regulador".
3. Seleccione en el área "Rango partido" la opción "Activar" para controlar dos actuadores.
4. Especifique los siguientes valores, para las señales de entrada y salida A y también para las entradas de entrada y salida B.
   – Señal de entrada inferior.
   – Señal de entrada superior.
   – Señal de salida inferior.
   – Señal de salida superior.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Formador de impulsos de la regulación paso a paso

Procedimiento

Para parametrizar el formador de impulsos proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida digital > Salida [n°]".
3. Especifique en el área "Formador de impulsos" los siguientes parámetros:
   – Tiempo de posicionamiento del motor.
   – Duración mínima de impulso.
   – Duración mínima de pausa.
4. Seleccione la opción adecuada para los valores superior e inferior, a partir de la cual se aplicará el valor:
   – del bloque de función.
   – Del bloque de función o de una de las entradas digitales.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Convertir una señal analógica en una señal digital

Procedimiento

Para convertir una señal analógica en una señal digital proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida digital > Canal [n°]".
3. Especifique en el área "Formador de impulsos" los siguientes parámetros:
   - Duración mínima de impulso.
   - Duración mínima de pausa.
4. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Seleccionar la señal de salida y el tipo de señal

Procedimiento

Para seleccionar la señal de salida y el tipo de señal proceda del siguiente modo:

1. En la vista de dispositivos, haga doble clic en la "Configuración de parámetros" del módulo.
2. En la vista de configuración seleccione "Salida analógica > Salida [n°]".
3. Seleccione en la opción "Origen" la señal a utilizar como señal de salida. Se proporcionan las siguientes opciones:
   - Cero: ningún origen para la señal de salida.
   - Una de las entradas analógicas.
   - Una de las variables manipuladas para el regulador A o B.
4. Seleccione en la opción "Tipo de señal" el tipo de señal en que se emitirá la señal de salida.
5. Guarde los parámetros configurados mediante "Proyecto > Guardar".

Programación del FM 455

Puesta en servicio del FM 455

Crear y configurar DBs de instancia

Procedimiento

Para satisfacer los requisitos para la puesta en marcha, proceda de la manera siguiente:

1. Cree un programa de usuario que efectúe cíclicamente una llamada a la instrucción PID_FM_455 en un OB de alarma cíclica.
2. En el programa del usuario, ponga READ_VAR cíclicamente a TRUE.

3. Genere para cada canal del regulador un DB de instancia individual de la instrucciónPID_FM_455.
4. Para cada DB de instancia, introduzca la dirección de módulo en el parámetro PID_FM_455.MOD_ADDR.
   Utilice como dirección de módulo del FM 455 la dirección inicial especificada al configurar el hardware.
5. Introduzca en el parámetro PID_FM_455.CHANNEL el número del canal de regulación (1 ... 16).
6. Si utiliza la instrucción PID_PAR_455, introduzca en el parámetro PID_PAR_455.CHANNEL el número del canal del regulador.
7. Compruebe que esté activado el "Tipo de sensor Entrada analógica" del canal en cuestión. Si no se procesa la entrada analógica, todos los parámetros PID se ajustan a cero, y no se calcula ninguna variable manipulada.

Nota

DBs multiinstancia

Si utiliza DBs multiinstancia en su proyecto, los FM se deben poner en servicio a través de una tabla de observación.

Consulte también

Descripción de PID_FM_455 (Página 7570)

Descripción de PID_PAR_455 (Página 7597)

Puesta en servicio del regulador

Introducción

Los parámetros PID se pueden optimizar del modo siguiente:

- Regulador PID/PI (regulador paso a paso, regulador continuo, regulador de impulsos): Manual
- Regulador de temperatura por lógica difusa: Ciclo continuo
- Regulador paso a paso: Automáticamente con la instrucción TUN_ES
- Regulador continuo: Automáticamente con la instrucción TUN_EC
- Regulador de impulsos: Automáticamente con la instrucción TUN_EC

Procedimiento

Para optimizar un regulador, proceda del siguiente modo:

1. Coloque la CPU en el estado RUN.
2. Si utiliza un regulador paso a paso, mida el tiempo de posicionamiento del motor.

3. Optimice los parámetros PID con alguna de las posibilidades citadas anteriormente.
4. Observe el lazo de regulación con el registrador de curvas.

Consulte también

Descripción de TUN_ES (Página 7149)
Descripción de TUN_EC (Página 7129)
Medir el tiempo de posicionamiento del motor (Página 10365)

Optimizar los parámetros PID manualmente

Requisitos

En la parametrización del algoritmo de regulación ha elegido el "Regulador PID".

Procedimiento

Para ajustar manualmente un regulador PI/PID, proceda del modo siguiente:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. Introduzca los valores correspondientes para la acción P, la acción I, la acción D y el retardo.
4. En el grupo "Optimización", haga clic en el símbolo "Enviar parámetros PID al FM".
5. En el grupo "Valores actuales", active la casilla de verificación "Especificar consigna".
6. Introduzca una nueva consigna.
7. Desactive en caso necesario la casilla de verificación "Modo manual" y, en el grupo "Valores actuales", haga clic en el símbolo .
   El regulador trabaja con los nuevos parámetros y regula según la nueva consigna.
8. Compruebe la calidad de los parámetros PID mediante los recorridos de la curva.
9. Repita los pasos del 3 hasta el 8, hasta que esté satisfecho/a con el resultado de la regulación.

Identificación de recorridos por lógica difusa

Requisitos del proceso para un regulador de temperatura por lógica difusa

Requisitos

Para conseguir una regulación óptima con la ayuda del regulador de temperatura por lógica difusa, el proceso debe satisfacer los requisitos siguientes:

- En caso de calefacciones de baños, el líquido calefactor debe estar bien mezclado.
- En caso de regulaciones tipo baño en baño, ambos líquidos deben estar bien agitados. Simultáneamente, se debe proporcionar una buena transferencia de calor entre todos los medios caloportadores. Si se emplean materiales que son malos caloportadores, para conseguir un buen transporte del calor se deben utilizar superficies de transferencia de gran tamaño.
- En caso de regulación de la temperatura del aire ambiental, se debe proporcionar una buena mezcla (p. ej., mediante ventiladores).
- La ganancia del sistema regulado no debe superar un factor 3.
- El tiempo de retardo debe equivaler como máximo al 3% del tiempo de compensación.
- Durante el periodo de muestreo del regulador, y con la salida de la variable manipulada en el valor máximo, la temperatura que se desea regular debe variar como máximo un 1‰ de la temperatura máxima preajustada.

Clasificación de los sistemas regulados

Un sistema regulado, o bien un proceso que se desea regular, se caracteriza por parámetros como, p. ej., la potencia de calefacción, la masa que se debe calentar o la capacidad térmica del medio que se desea calentar. En lo que se refiere al regulador por lógica difusa, se efectúa una distinción entre sistemas de temperatura regulada "críticos" y "no críticos" del modo siguiente: La regulación es tanto más crítica

- cuanto mayor es la potencia de calefacción,
- cuanto mayor es la capacidad térmica de la calefacción,
- cuanto menor es la capacidad térmica del medio que se desea calentar,
- cuanto mayor es la resistencia a la transmisión de calor,
- cuanto menor es la superficie de transferencia de calor.

Tras el seguimiento automático de un salto en la variable manipulada por parte del sistema regulado, éste reacciona con una respuesta a un escalón. La clasificación de los sistemas regulados también se puede llevar a cabo en función de esta respuesta a un escalón: La regulación es tanto más crítica cuanto mayor es el cociente t_u / t_g y cuanto mayor es la ganancia del sistema regulado. Si t_u / t_g < 1/10, el sistema regulado no es crítico.

Desarrollo de la identificación del sistema por lógica difusa

Desarrollo de la optimización de parámetros con un regulador de temperatura por lógica difusa

La optimización de parámetros se basa en un regulador por lógica difusa autooptimizante.

La identificación del sistema regulado comienza con una fase de observación, durante la cual la potencia de calefacción emitida es nula. Esta fase de observación dura lo siguiente:

- Fase de observación de un regulador continuo: aprox. 1 minuto
- Fase de observación de un regulador paso a paso: aprox. 1 minuto + 1/2 x tiempo de desplazamiento del actuador

Este tiempo se emplea para poder detectar tendencias de temperatura en el medio calefactor. A continuación se emite el 100% de la potencia máxima de calefacción; esto se aprecia en el parámetro de salida LMN del DB de instancia de la instrucción PID_FM.

El área del primer 4% de incremento de la temperatura del rango de consigna se utiliza para la identificación, ya que tanto en el 1% como en el 4% del incremento de temperatura se obtiene información sobre el comportamiento del proceso a partir del tiempo transcurrido.

La identificación concluye cuando la potencia de calefacción pasa a ser inferior al 100%. Este comportamiento se puede utilizar, p. ej., para desconectar la identificación a través del programa de usuario. Tras la fase de identificación, el regulador sigue funcionando con los parámetros obtenidos.

Si la divergencia provocada por el salto de consigna no se compensa y la potencia de calefacción permanece a cero de manera continua, significa que la identificación ha resultado fallida, es decir, el regulador no puede regular el sistema regulado al que está conectado.

Comportamiento del regulador para distintos sistemas regulados

Si el sistema regulado es "no crítico", no hay problemas en la identificación ni en la regulación.

La identificación de un sistema regulado "demasiado crítico" queda cancelada. Si un sistema regulado se identifica como "crítico", su regulación se efectúa con mucha "precaución", es decir, despacio.

Efectuar la identificación de sistema por lógica difusa.

Requisitos

En la parametrización del algoritmo de regulación ha elegido el "Regulador por lógica difusa".

Si es posible, asegúrese de que antes de la identificación el sistema regulado se haya estabilizado (ningún proceso activo de calefacción o refrigeración) o varíe con lentitud y de forma monótona. Se puede utilizar como criterio al respecto que el comportamiento de la variación de temperatura durante un minuto se pueda aproximar por una recta. En procesos rápidos, este requisito resulta especialmente actual.

Dado que el valor emitido para la variable manipulada es nulo durante aprox. un minuto al principio de la identificación del regulador, la temperatura que se desea regular debe ser cercana a la temperatura ambiente.

Iniciar la identificación del sistema por lógica difusa

Para optimizar un regulador por lógica difusa, proceda del modo siguiente:

1. Seleccione el DB de instancia del canal que desee optimizar.
2. Haga clic en el símbolo "Start".
   Si aún no existe ninguna conexión online, esta se crea. Se graban los valores actuales de consigna, valor real y variable manipulada.
3. En el grupo "Optimización" introduzca una nueva consigna (SP_OP).
   La nueva consigna debe cumplir las condiciones siguientes:
   – Salto de consigna > 5%
   – Nueva consigna > valor real + límite superior de la consigna x 0,12
4. En el grupo "Optimización", haga clic en el símbolo "Identificación del sistema por lógica difusa".

Resultado

La nueva consigna (SP_OP), SP_OP_ON = TRUE y FUZID_ON = TRUE se envían al FM. La identificación del sistema por lógica difusa comienza. El parámetro IDSTATUS de la instrucción CH_DIAG proporciona información sobre el estado de la identificación. En el cuadro de visualización de estado puede ver si hay una identificación del sistema por lógica difusa en estado activo o en ejecución.

Desactivar la identificación del sistema por lógica difusa

Mientras el bit FUZID_ON = TRUE, cada vez que se produzca un salto de consigna suficientemente grande se iniciará la identificación siguiente.

Para desconectar el modo de optimización, proceda del modo siguiente:

1. Inserte FUZID_ON en una tabla de observación.
2. Defina FUZID_ON = FALSE.

Nota

La desconexión del modo de optimización FUZID_ON = FALSE antes de terminar la identificación no provoca que esta se detenga. Una vez iniciada la identificación, esta sigue adelante pase lo que pase, excepto si se produce un salto de consigna negativo.

Repetir la identificación del sistema por lógica difusa

Cada vez que se produce un salto de consigna ≥12% se efectúa una nueva identificación, a no ser que esta se desconecte de nuevo con FUZID_ON = FALSE.

También existe la posibilidad de iniciar de nuevo la identificación disminuyendo y, a continuación, aumentando de nuevo la consigna en un valor >12%.

Cancelar la identificación del sistema por lógica difusa

La identificación se puede cancelar en los casos siguientes:

- Por parte del regulador, si se identifica un sistema regulado "crítico". Tras la cancelación, el regulador se encuentra en estado de error. Este estado se manifiesta en que el regulador desactiva la variable manipulada de forma permanente. Este estado no varía aunque se desconecte la optimización. El inicio de una nueva identificación anula el estado de error. Se puede obtener información sobre el estado de la identificación a través del parámetro IDSTATUS del FB CH_DIAG. Más detalles al respecto al final de este capítulo, en la sección "Información sobre el estado del regulador".
- Por parte del operador, generando un salto de consigna negativo, siempre y cuando la consigna rebase por defecto la condición 2 (véase más arriba, "¿Cómo iniciar la identificación?").

Para cancelar la optimización, proceda del modo siguiente:

1. Introduzca una nueva consigna que sea inferior al valor real actual y que cumpla las mismas condiciones que al principio de la identificación del sistema por lógica difusa.
2. En el grupo "Optimización", haga clic en el símbolo "Identificación del sistema por lógica difusa".

Consulte también

Parámetro IDSTATUS (Página 7595)

Cargar los parámetros optimizados en el dispositivo

Introducción

Para impedir que los parámetros del FM se sobrescriban con los datos del sistema en el siguiente arranque de la CPU, se puede proceder del modo siguiente:

- Tras cada arranque del FM , cargue en este con LOAD_PAR=TRUE los parámetros PID optimizados del DB de instancia del PID_FM.
- Cargue los parámetros PID optimizados en el SDB.

Nota

Los juegos de parámetros PI, PID y SAVE no se pueden guardar en el SDB.

Procedimiento

Para cargar los parámetros en el SDB, proceda del siguiente modo:

1. Haga clic en el símbolo "Transfiera datos online al proyecto offline".
   Los parámetros PID se guardan en el ajuste de parámetros del FM.
2. Pare la CPU.
3. Haga clic en el símbolo "Cargar en dispositivo".

Trabajar con el registrador de curvas

Observar

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Puesta en servicio".
2. Navegue en la vista de configuración hasta "Registrador de curvas".
3. Seleccione el DB de instancia que desee observar. Tras la selección se muestra el canal al que está asignado el DB de instancia.
4. Defina el tiempo de actualización con el que debe registrarse el registrador de curvas.
5. Haga clic en "Medición activada" para iniciar la medición.
6. Haga clic en "Medición desactivada" para finalizar la medición.

Procedimiento

Para adaptar la escala, proceda del siguiente modo:

1. En la vista de dispositivos, bajo el módulo, haga doble clic en "Puesta en servicio".
2. Navegue en la vista de configuración hasta "Registrador de curvas".
3. Para modificar el eje de tiempo:
   - Haga clic en el botón de flecha junto a la fecha, para abrir un calendario. Seleccione una fecha en el calendario. Para bloquear la selección, haga clic en el candado azul. Esta posibilidad existe tanto para la fecha inicial como para la fecha final.
   - Haga clic con el puntero del ratón en la escala de tiempo. El puntero del ratón se representa como flecha a la izquierda/derecha. Mantenga el puntero del ratón presionado y arrástrelo hacia la izquierda o derecha.
     Si el tiempo inicial o final está bloqueado, se modifica la escala del tiempo.
     Si no está bloqueado el tiempo inicial ni el final, se modifica el período indicado; la escala se mantiene igual.
4. Para modificar el eje de tiempo:
   - Haga clic en el campo de entrada para el valor superior o el valor inferior mostrado. Introduzca ahí el valor deseado. Para bloquear la selección, haga clic en el candado azul. Esta posibilidad existe tanto para el valor superior como también para el inferior.
   - Haga clic con el puntero del ratón en la escala de valores. El puntero del ratón se representa como flecha arriba/abajo. Mantenga el puntero del ratón presionado y arrástrelo hacia arriba o abajo.
     Si el valor superior o inferior está bloqueado, se modifica la escala del eje de valores.
     Si no está bloqueado el valor superior ni el valor inferior, se modifica el rango indicado; la escala se mantiene igual.

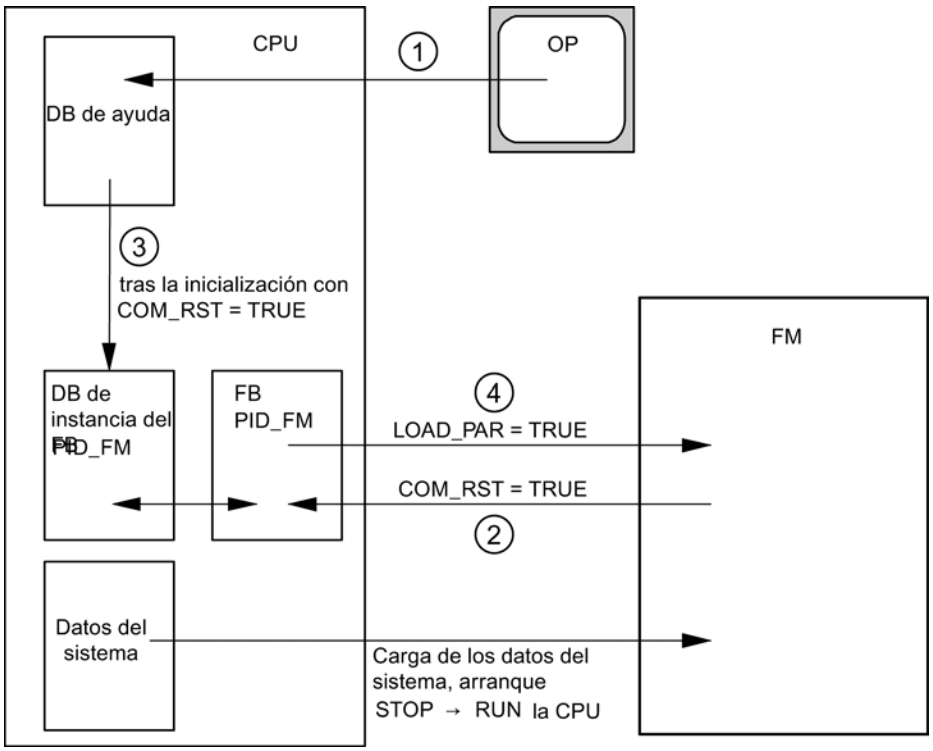
Modificar los parámetros a través de la HMI

Procedimiento

Si desea modificar los parámetros del regulador de la instrucción PID_FM mediante el manejo a través de la HMI, proceda del modo siguiente:

1. Desde la HMI, escriba en la CPU (véase ①) en un DB auxiliar los parámetros que se deben modificar.
2. Ponga PID_FM.COM_RST a TRUE.
   La instrucción ejecuta una inicialización. Los parámetros se sincronizan entre el FM y la CPU (véase ②).
3. Transfiera los parámetros que se deben modificar desde el DB auxiliar hasta el DB de instancia de la instrucción PID_FM (véase ③).
4. Ponga PID_FM.LOAD_PAR a TRUE.
   Los parámetros modificados se transfieren al FM (véase ④).

Resulta necesario depositar los parámetros en un DB auxiliar porque, tras el arranque de la CPU, PID_FM lee con COM_RST = TRUE aquellos parámetros del módulo que la CPU ha transferido previamente desde los datos del sistema al FM.



Consulte también

Funcionamiento de PID_FM_455 (Página 7571)

Medir el tiempo de posicionamiento del motor

Introducción

Para conseguir un buen resultado de la regulación, el tiempo de posicionamiento del motor debe ser lo más preciso posible. Los datos de la documentación del elemento final de control son valores medios para este tipo de elemento final de control. El valor puede variar para el elemento final de control que se use en cada momento.

Si utiliza actuadores con realimentación de posición o con señales de tope, es posible medir el tiempo de posicionamiento del motor durante la puesta en servicio. Si no se dispone de una realimentación de posición ni de señales de tope, no es posible medir el tiempo de posicionamiento del motor.

Requisitos

- El valor parametrizado del tiempo de posicionamiento del motor debe equivaler como mínimo a la mitad del tiempo de posicionamiento real. El actuador se desplaza en un sentido durante el doble del tiempo de posicionamiento como máximo.
- En los actuadores con realimentación de posición analógica, la realimentación de posición debe estar interconectada.
- En los actuadores con señales de tope, las señales de tope deben estar interconectadas.

Actuadores con realimentación de posición analógica

Para medir el tiempo de posicionamiento del motor con realimentación de posición, proceda del siguiente modo:

1. Elija el DB instancia para el que se debe medir el tiempo de posicionamiento del motor.
2. Active la casilla de verificación "Utilizar realimentación de posición".
3. En el campo de entrada "Posición de destino" indique hacia dónde desea mover el elemento final de control.
   La realimentación de posición actual (posición de inicio) se muestra. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se desplaza hasta la posición de destino desde la posición de inicio. La medición del tiempo se inicia de inmediato y finaliza en cuanto el elemento final de control ha alcanzado la posición de destino. El tiempo de posicionamiento del motor se calcula según la fórmula:

Tiempo de posicionamiento del motor = (Límite superior del valor de salida - Límite inferior del valor de salida) x Tiempo de medición /VALOR ABSOLUTO(Posición de destino - Posición de inicio).

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se muestra en el campo "Tiempo de actuación medido". Una vez finalizada la medición del tiempo de posicionamiento se conecta el modo manual de las

señales de posicionamiento (LMNSOPON = TRUE). El elemento final de control deja de moverse.

Actuadores con señales de tope

Para medir el tiempo de posicionamiento del motor con señales de tope, proceda del siguiente modo:

1. Elija el DB instancia para el que se debe medir el tiempo de posicionamiento del motor.
2. Active la casilla de verificación "Utilizar señales de tope del actuador".
3. Haga clic en el botón "Iniciar medición tiempo de posicionamiento".

Resultado

El elemento final de control se mueve primero hasta el tope inferior, después, hasta el tope superior y, de nuevo, hasta el tope inferior. La medición del tiempo se inicia cuando el actuador ha alcanzado el tope inferior y finaliza cuando el actuador alcanza por segunda vez el tope inferior. El tiempo medido dividido entre dos da lugar al tiempo de posicionamiento del motor.

Se muestran el progreso y el estado de la medición del tiempo de posicionamiento. El tiempo de posicionamiento medido se muestra en el campo "Tiempo de actuación medido". Una vez finalizada la medición del tiempo de posicionamiento se conecta el modo manual de las señales de posicionamiento (LMNSOPON = TRUE). El elemento final de control deja de moverse.

Cargar el tiempo de ajuste medido

1. Para guardar en el proyecto el tiempo de posicionamiento del motor medido, haga clic en el símbolo "Transfiere datos online al proyecto offline".
   El tiempo de ajuste medido se guarda en el ajuste de parámetros del FM.
2. Para que los cambios surtan efecto en el arranque siguiente de la CPU, haga clic en el icono "Cargar en dispositivo".

Cancelar medición del tiempo de posicionamiento

Si cancela la medición del tiempo de posicionamiento, el modo manual de las señales de posicionamiento se conecta (LMNSOPON = TRUE) y el modo manual con especificación de la variable manipulada se desconecta (LMNOP_ON = FALSE). El elemento final de control deja de moverse.

Consulte también

Parámetros de entrada/salida de PID_FM_455 (Página 7577)

FM 455: Diagnóstico a través de indicadores LED

Indicación de errores mediante el LED de error

¿Dónde se indican los errores?

El LED rojo INTF indica que hay un error interno en el módulo.

El LED EXTF indica que hay un error externo, por ejemplo, un error en las conexiones.

Si el LED amarillo parpadea, el firmware se borra. Este estado sólo puede aparecer si existe un defecto en el hardware o si el proceso de carga del firmware se cancela.

¿Qué errores se indican?

Los siguientes errores se indican con la iluminación de los LEDs de error:

Tipo de error	Aviso de diagnóstico	Causa posible	Solución
Errores internos	Módulo defectuoso	Error de hardware	Cambie el módulo
	Vigilancia de tiempo (watch dog) activado	Error de hardware	Cambie el módulo
	Contenido EEPROM no válido	Fallo de la tensión de alimentación al parametrizar	Vuelva a parametrizar el módulo
Errores externos	Parámetro erróneo en el módulo	Se han transferido parámetros incorrectos al módulo	Vuelva a parametrizar el módulo
	Error en las entradas o las salidas analógicas	Error de hardware en la entrada analógica	Cambie el módulo
		Rotura de hilo en la entrada analógica	Solucione la rotura de hilo
		Rebase por defecto del rango de medición en la entrada analógica	Revise la señal de medición
		Rebase del rango de medición en la entrada analógica	Revise la señal de medición
		Rotura de hilo en la salida analógica	Solucione la rotura de hilo
		Cortocircuito en la salida analógica	Solucione el cortocircuito
	Falta tensión auxiliar externa	El conector frontal izquierdo no está insertado Fallo en la alimentación 24 V	Inserte el conector frontal izquierdo Restablezca la alimentación de 24 V

Alarma de diagnóstico en caso de error

Todos los errores pueden disparar una alarma de diagnóstico si ésta se ha habilitado en la pantalla correspondiente de parametrización. En los registros de diagnóstico DS0 y DS1 puede ver qué error ha provocado que se encienda el LED. La asignación de los registros de diagnóstico DS0 y DS1 está descrita en el capítulo "Disparo de alarmas de diagnóstico (Página 10368)".

Disparo de alarmas de diagnóstico

¿Qué es una alarma de diagnóstico?

Si se quiere que un programa de usuario responda a un error interno o externo, se puede parametrizar una alarma de diagnóstico que interrumpa el programa cíclico de la CPU y llame al OB de alarma de diagnóstico (OB 82).

¿Qué eventos pueden disparar una alarma de diagnóstico?

La siguiente lista muestra qué eventos pueden disparar una alarma de diagnóstico:

- Módulo no parametrizado o errores en la parametrización
- Módulo defectuoso
- Rotura de hilo en entradas analógicas (4 a 20 mA)
- Rebase por exceso y por defectos en entradas analógicas
- Rotura de carga y cortocircuito en salidas analógicas
- Rotura de hilo en termoelementos y Pt 100

Ajuste por defecto

La alarma de diagnóstico está bloqueada de forma predeterminada.

Habilitación de la alarma de diagnóstico

La pantalla Parámetros básicos permite bloquear o habilitar la alarma de diagnóstico para el módulo.

Respuestas a un evento que dispara una alarma

Cuando se produce un evento que puede disparar una alarma de diagnóstico ocurre lo siguiente:

- La información de diagnóstico se registra en los registros de diagnóstico DS0 y DS1 del módulo.
- Uno o los LEDs de error se encienden.
- Se llama al OB de alarma de diagnóstico (OB 82).
- El registro de diagnóstico DS0 se graba en la información de arranque del OB de alarma de diagnóstico.
- Si no hay ningún defecto de hardware, el módulo sigue regulando.

La CPU pasa a modo STOP si no se ha programado el OB 82.

Registro de diagnóstico DS0 y DS1

Introducción

La información que indica qué evento ha provocado una alarma de diagnóstico se guarda en los registros de diagnóstico DS0 y DS1. El registro de diagnóstico DS0 consta de cuatro bytes; DS1 consta de 27 bytes, siendo los cuatro primeros idénticos a DS0.

Leer el registro desde el módulo

El registro de diagnóstico DS0 se transfiere automáticamente a la información de arranque en la llamada del OB de diagnóstico. Allí, estos cuatro bytes se guardan en los datos locales (bytes 8-11) del OB 82.

El registro de diagnóstico DS1 se puede leer (y en consecuencia también el contenido de DS0) desde el módulo con la instrucción "CH_DIAG_455". Esta lectura sólo tiene sentido si en el DS0 se notifica un error en un canal.

¿Cómo aparecen los textos de diagnóstico en el búfer de diagnóstico?

Si se desea registrar el aviso de diagnóstico en el búfer de diagnóstico, se debe llamar la instrucción "WR_REC" "Introducir avisos específicos de usuario en el búfer de diagnóstico" dentro del programa de usuario. En cada caso, el número de evento del aviso de diagnóstico se especifica en el parámetro de entrada EVENTN. La alarma se introduce en el búfer de diagnóstico con x=1 como entrante y con x=0 como saliente. El búfer de diagnóstico contiene los textos de diagnóstico correspondientes en la columna de "Significado", así como el momento en que se produjo la entrada.

Asignación del registro de diagnóstico DS0 en la información de arranque

La tabla siguiente muestra la asignación del registro de diagnóstico DS0 en la información de arranque. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Tabla 11-11 Asignación del registro de diagnóstico DS0

Byte	Bit	Significado	Nota	Nº de evento
0	0	Avería en módulo	Se activa en todos los eventos de diagnóstico	8:x:00
	1	Error interno	Se activa en todos los errores internos: • Vigilancia de tiempo activada • Contenido de la EEPROM no válido; el módulo arranca sin regular y espera a que la CPU realice una nueva parametrización • Error EPROM • Error ADC/DAC • Error de hardware en la entrada analógica	8:x:01

Byte	Bit	Significado	Nota	Nº de evento
	2	Error externo	Se activa en todos los errores externos: • Falta tensión auxiliar externa • Error de parametrización • Rotura de hilo en la entrada analógica (sólo en el margen 4 a 20 mA) • Rebase por defecto del rango de medición en la entrada analógica • Rebase del rango de medición en la entrada analógica • Rotura de hilo en la salida analógica • Cortocircuito en la salida analógica	8:x:02
	3	Fallo en un canal	Consultar el DS1, a partir del byte 7, para más detalles	8:x:03
	4	Falta tensión auxiliar externa	Fallo en la alimentación 24 V del FM 455	8:x:04
	6	Contenido EEPROM no válido	Fallo de la tensión de alimentación durante un proceso de escritura en la EEPROM. El módulo arranca con los parámetros predeterminados.	8:x:03
	7	Error de parametrización	El módulo no puede utilizar un parámetro. Motivo: Parámetro desconocido o combinación de parámetros no permitida. Consulte el menú **Sistema de destino > Mostrar errores de parametrización**	8:x:07
1	0 ... 3	Clase de módulo	Siempre tiene asignado un 8	–
	4	Diagnóstico específico de canal	Se activa cuando el módulo puede proporcionar información del canal adicional y hay un error de canal (consulte DS1 bytes 7 a 12)	–
2	0	Falta módulo de usuario o es incorrecto	La posición (A, B, C, D) del módulo del rango de medición no concuerda con la parametrización del FM 455.	8:x:30
	3	Vigilancia de tiempo (watch dog) activado	Error de hardware	8:x:33
3	2	Error EPROM	Módulo defectuoso	8:x:42
	4	Error ADC/DAC	Módulo defectuoso	8:x:44

Asignación del registro de diagnóstico DS1 del FM 455

El registro de diagnóstico DS1 se compone de 27 bytes. Los 4 primeros bytes son idénticos a los del registro de diagnóstico DS0. La siguiente tabla muestra la asignación de los bytes restantes. Todos los bits que no se incluyen en la tabla son irrelevantes y tienen el valor cero.

Tabla 11-12 Asignación de los bytes del 4 a 12 del registro de diagnóstico DS1 del FM 455

Byte	Bit	Significado	Nota		Nº de evento
4	0 ... 7	Tipo de canal	Siempre tiene asignado 75H		–
5	0 ... 7	Longitud de la información de diagnóstico	Siempre tiene asignado un 8		–
6	0 ... 7	Cantidad de canales	Siempre tiene asignado un 17 (16 reguladores + 1 canal de referencia)		–
7	0 … 7	Vector de error de canal	A los canales 1 a 8 se les asignan sendos bits		-
8	0 … 7	Vector de error de canal	A los canales 9 a 16 se les asignan sendos bits		-
9	0	Error en el canal de referencia			-
10	0	Error de hardware en la entrada analógica	Diagnóstico específico del canal 1	–	8:x:B0
	1	Error de parametrización		La posición (A, B, C, D) del módulo del rango de medición no concuerda con la parametrización del FM 455.	8:x:B1
	2	Rotura de hilo en la entrada analógica (sólo en el margen 4 a 20 mA)			8:x:B2
	3	sin usar			8:x:B3
	4	Rebase por defecto del rango de medición en la entrada analógica[1]			8:x:B4
	5	Rebase por defecto del rango de medición en la entrada analógica[1]			8:x:B5
	6	Rotura de hilo en la salida analógica		Sólo en la salida de intensidad del regulador K	8:x:B6
	7	Cortocircuito en la salida analógica		Sólo en la salida de tensión del regulador K	8:x:B7
11	0 ... 7	véase byte 10	Diagnóstico específico del canal 2		véase arriba

Byte	Bit	Significado	Nota	Nº de evento
12	0 ... 7	véase byte 10	Diagnóstico específico del canal 3	véase arriba
13	0 ... 7	véase byte 10	Diagnóstico específico del canal 4	véase arriba
14	0 ... 7	véase byte 10	Diagnóstico específico del canal 5	véase arriba
15	0 ... 7	véase byte 10	Diagnóstico específico del canal 6	véase arriba
16	0 ... 7	véase byte 10	Diagnóstico específico del canal 7	véase arriba
17	0 ... 7	véase byte 10	Diagnóstico específico del canal 8	véase arriba
18	0 ... 7	véase byte 10	Diagnóstico específico del canal 9	véase arriba
19	0 ... 7	véase byte 10	Diagnóstico específico del canal 10	véase arriba
20	0 ... 7	véase byte 10	Diagnóstico específico del canal 11	véase arriba
21	0 ... 7	véase byte 10	Diagnóstico específico del canal 12	véase arriba
22	0 ... 7	véase byte 10	Diagnóstico específico del canal 13	véase arriba
23	0 ... 7	véase byte 10	Diagnóstico específico del canal 14	véase arriba
24	0 ... 7	véase byte 10	Diagnóstico específico del canal 15	véase arriba
25	0 ... 7	véase byte 10	Diagnóstico específico del canal 16	véase arriba
26	0 ... 7	véase byte 10	Diagnóstico para el canal de referencia	véase arriba
[1] consulte capítulo "Fallo del transductor de medida (Página 10373)"				

Factores importantes que deben tenerse en cuenta

El FM 455 recibe alimentación de tensión exclusivamente a través del conector frontal izquierdo.

Así pues, la CPU detecta "Módulo desenchufado/no responde" cuando se da uno de los siguientes casos:

- cuando el conector frontal izquierdo del FM 455 no se ha insertado
- cuando falta la alimentación de tensión de 24 V del conector frontal izquierdo

Nota

En el caso de que haya registrado la entrada "Modulo desenchufado/no responde" en el búfer de diagnóstico de la CPU, compruebe si se ha insertado el conector frontal izquierdo y si el FM 455 dispone de una alimentación de tensión de 24 V.

Información adicional

Para obtener más información sobre el parámetro "RET_REALU", consulte Lista de los avisos RET_REALU (Página 10374) y el área de instrucciones avanzadas del sistema de ayuda.

Fallo del transductor de medida

Fallo del transductor de medida

El FM 455 puede detectar los siguientes fallos en el transductor de medida:

- Rebase por defecto del rango de medición
- Rebase del margen de medición
- Rotura de hilo (no se muestra en todos los márgenes de medición)

Si se produce una de estas averías en el registro de diagnóstico DS0 se define el bit de error de grupo "Error externo" y, en el registro de diagnóstico DS1, bits de error específicos del canal (consulte las tablas en la sección anterior). Cuando desaparecen estas averías los bits correspondientes se desactivan.

La siguiente tabla muestra los límites de los diferentes márgenes de medición en los que los bits de error se activan o desactivan:

Rango de medida	Bit de error Rebase por defecto del rango de medición en ...	Bit de error Rebase del rango de medición en ...	Bit de error Mostrar rotura de hilo
	DS1: Byte 10 a 26, Bit 4	DS1: Byte 10 a 26, Bit 5	DS1: Byte 10 a 26, Bit 2
0 a 20 mA	< - 3,5 mA	> 23,5 mA	–
4 a 20 mA	Bit de error = 1 a < 3,6 mA Bit de error = 0 a > 3,8 mA	> 22,8 mA	Bit de error = 1 a < 3,6 mA Bit de error = 0 a > 3,8 mA
0 a 10 V	< - 1,175 V	> 11,75 V	–
Pt 100 (-200 a 850 °C) (-328 a 1562 °F)	< 30,82 mV	> 650,46 mV	sí, parametrizable, Visualización: 650,46 mV
Pt 100 (-200 a 556 °C) (-328 a 1032 °F)	< 30,82 mV	> 499,06 mV	sí, parametrizable, Visualización: 499,06 mV
Pt 100 (-200 a 130 °C) (-328 a 264 °F)	< 30,82 mV	> 254,12 mV	sí, parametrizable, Visualización: 254,12 mV
Termoelemento tipo B	< 0 mV	> 13,81 mV	sí, parametrizable, se mantiene el último valor
Termoelemento tipo J	< - 8,1 mV	> 69,54 mV	sí, parametrizable, se mantiene el último valor
Termoelemento tipo K	< - 6,45 mV	> 54,88 mV	sí, parametrizable, se mantiene el último valor

Rango de medida	Bit de error Rebase por defecto del rango de medición en ...	Bit de error Rebase del rango de medición en ...	Bit de error Mostrar rotura de hilo
	DS1: Byte 10 a 26, Bit 4	DS1: Byte 10 a 26, Bit 5	DS1: Byte 10 a 26, Bit 2
Termoelemento tipo R	< - 0,23 mV	> 21,11 mV	sí, parametrizable, se mantiene el último valor
Termoelemento tipo S	< - 0,24 mV	> 18,7 mV	sí, parametrizable, se mantiene el último valor
Termopar libre	< Valor de entrada inferior de la línea poligonal	> Valor de entrada superior de la línea poligonal	sí, parametrizable, se mantiene el último valor

Lista de los avisos RET_REALU

Avisos RET_REALU

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
7000	28672	-32624	Primera llamada con REQ=0: No hay ninguna transferencia de datos activa; BUSY tiene el valor 0.
7001	28673	-32624	Primera llamada con REQ=1: Transferencia de datos lanzada; BUSY tiene el valor 1.
7002	28674	-32624	Llamada intermedia (REQ irrelevante). Transferencia de datos ya activa; BUSY tiene el valor 1.
8090	32912	-32624	La dirección base lógica indicada no es válida: No existe ninguna asignación en el SDB1/SDB2x, o no se trata de una dirección base.
80A0	32928	-32608	Acuse negativo al leer el módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2	32930	-32606	Error de protocolo para capa 2
80A3	32931	-32605	Error de protocolo en User-Interface / User
80A4	32932	-32604	Fallo de comunicación en el bus K
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo teórico.
80C0	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1	32961	-32575	Los datos de una petición de escritura semejante todavía no se han procesado en el módulo.
80C2	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados en estos momentos.
80C4	32964	-32572	Error de comunicación
80C5	32965	-32571	La periferia descentralizada no está disponible.

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80C6	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (petición de escritura).
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (petición de escritura).
853A	34106	-31430	DB de parámetros no presente. (petición de escritura).
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras aparecer un error. (petición de escritura).
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (petición de lectura).
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU. No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (petición de lectura).
873A	34618	-30918	DB de parámetros no presente. (petición de lectura).
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras aparecer un error. (petición de lectura).
80ff	33023	-32513	Indicación de índice incorrecta en el bloque FMT_PAR
Los errores 80A2..80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario. Los avisos con la estructura 7xxx indican situaciones temporales de la comunicación.			

Proyectos de ejemplo para el FM 455

11.3.5.8 Utilizar ET 200S 2PULSE

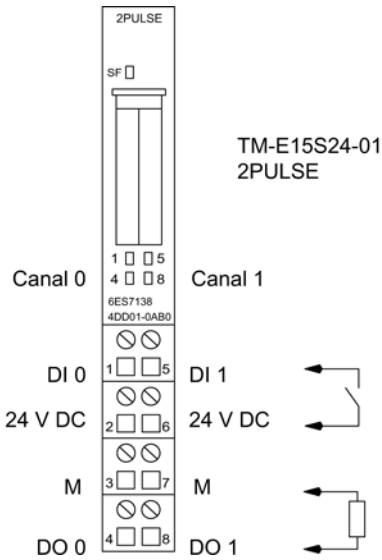
Asignación de conectores del 2PULSE

Reglas de cableado

Los cables (terminales 1 y 2 y terminales 5 y 6) deben estar apantallados. La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contacto (ver el anexo de las instrucciones de servicio Sistema de periferia descentralizada ET 200S (http://support.automation.siemens.com/WW/view/es/1144348)).

Asignación de conexiones del 2PULSE

La tabla siguiente muestra la asignación de conexiones del 2PULSE.

Vista	Asignación de conexiones	Significado
	TM-E15S24-01 y 2PULSE	Canal 0: Borne 1...4 Canal 1: Borne 5...8 24VDC: Alimentación del encóder M Masa DI: Señal de entrada DO: Señal de salida (máx. 2 A por canal)

Modos de operación y funciones

Sinopsis

Principio

El 2PULSE dispone de dos canales. Se puede seleccionar un modo de operación diferente para cada canal. El modo de operación se define al parametrizar. No es posible cambiar el modo de operación parametrizado con el programa de control.

Para cada canal se puede optar entre distintos modos:

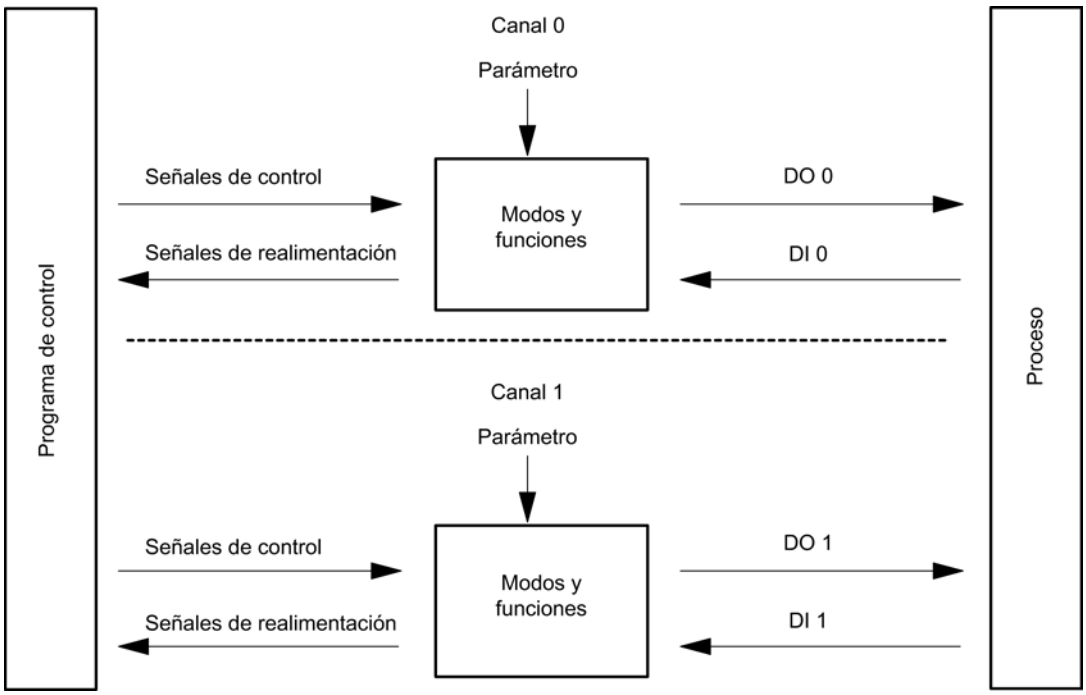
- Salida de impulsos (Página 10378)
- Modulación de ancho de impulsos (Página 10383)
- Tren de impulsos (Página 10393)
- Retardo a la conexión/desconexión (Página 10400)
- Salida de frecuencia (Página 10408) (a partir de 6ES7138-4DD01-0AB0)

Aparte del modo de operación ajustado, el 2PULSE puede ejecutar las siguientes funciones:

- Medición de intensidad (Página 10414) en los modos de operación "Modulación de ancho de impulsos" y "Tren de impulsos" (a partir de 6ES7138-4DD01-0AB0).
- Control directo de la salida digital DO (Página 10415) por medio del programa de control; controlable de forma independiente para cada canal.

- Detección de errores/diagnóstico (Página 10416); el 2PULSE detecta los errores de cada canal de forma independiente.
- Conexión en paralelo de ambos canales (Página 10418) para alcanzar una intensidad de salida más alta (a partir de 6ES7138-4DD01-0AB0).
- Reacción a STOP de la CPU/del maestro (Página 10420); el 2PULSE detecta el paso a STOP de la CPU/del maestro para ambos canales y reacciona de acuerdo con la parametrización efectuada.

La figura siguiente muestra el funcionamiento del 2PULSE.



Interfaces con el programa de control y el proceso

Para ejecutar los modos de operación y las funciones, el 2PULSE dispone como interfaz con el proceso de una entrada digital y una salida digital para cada canal (DI 0, DO 0 para el canal 0 y DI 1, DO1 para el canal 1).

Los modos de operación y las funciones se pueden modificar y observar con el programa de control mediante señales de control (Página 10425) y señales de realimentación (Página 10428).

Los distintos modos de operación tienen asignados parámetros.

En las descripciones correspondientes a los modos de operación y las funciones encontrará:

- los parámetros relevantes y
- las señales de control y realimentación.

La descripción de los modos de operación y las funciones se aplica a ambos canales, por lo que en esta descripción los canales ya no se marcan de forma independiente.

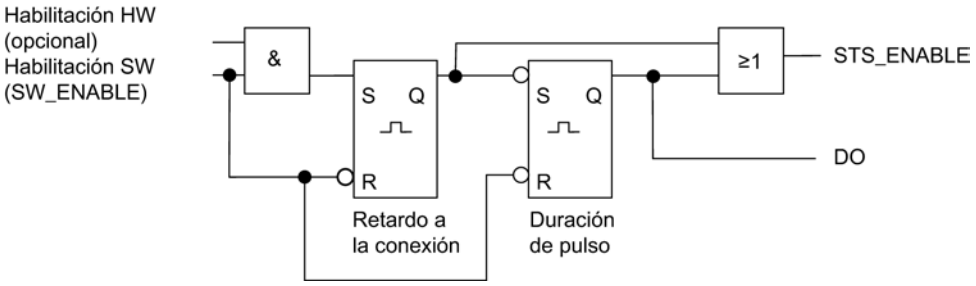
Modo "Salida de impulsos"

Modo "Salida de impulsos"

Salida de impulsos

Para la duración del impulso que se haya fijado, el 2PULSE emite un impulso en la salida digital DO (secuencia de salida) una vez transcurrido el retardo a la conexión (Página 10423) parametrizado.

La figura siguiente muestra el esquema de principio del modo de operación "Salida de impulsos".



Iniciar la secuencia de salida

La habilitación de la secuencia de salida debe realizarse siempre en el programa de control por habilitación software (Página 10425) (SW_ENABLE 0 → 1; MANUAL_DO = 0).

El bit de realimentación (Página 10428)ACK_SW_ENABLE indica la habilitación software pendiente en el 2PULSE.

Además es posible ajustar con el parámetro "Función DI (Página 10423)" la entrada digital DI del 2PULSE como habilitación hardware (habilitación HW).

Para trabajar simultáneamente con la habilitación software y la habilitación hardware, la secuencia de salida comienza con el primer flanco ascendente de la habilitación hardware una vez obtenida la habilitación software. El 2PULSE ignora los posteriores flancos ascendentes de la habilitación hardware durante la secuencia de salida actual. Una vez obtenida la habilitación software, un flanco ascendente de la habilitación hardware es suficiente para iniciar la siguiente secuencia de salida.

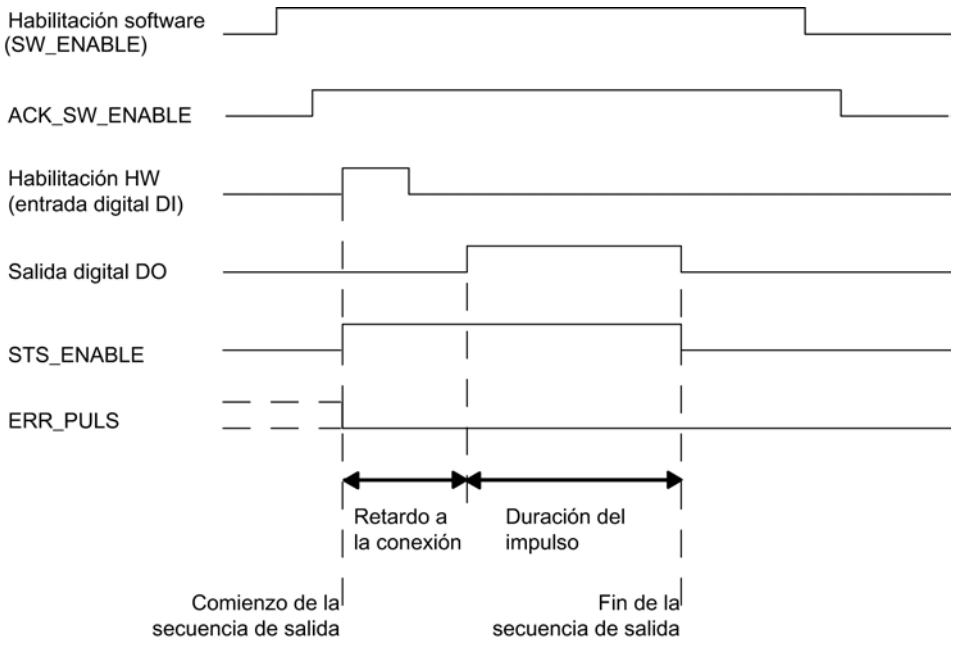
Cuando se produce la habilitación (flanco ascendente) comienza a contar el retardo a la conexión y se activa el bit STS_ENABLE. Una vez transcurrido el retardo a la conexión, el impulso se emite con la duración del impulso fijada. La secuencia de salida termina (Página 10413) con el fin del impulso; STS_ENABLE se borra.

En caso de modificar la duración del impulso durante el funcionamiento de forma no admisible, la señal (Página 10428)ERR_PULS indica un error en la salida de impulsos. Entonces deberá reiniciar la secuencia de salida.

La próxima vez que inicie la secuencia de salida, el 2PULSE borrará el bit de realimentación ERR_PULS.

Diagrama de impulsos

La figura siguiente muestra una secuencia de salida en el modo de operación "Salida de impulsos".



Cancelar la secuencia de salida

Borrando la habilitación software (SW_ENABLE = 0) durante el retardo a la conexión o durante la duración del impulso se cancela la secuencia de salida y el bit STS_ENABLE y la salida digital DO se borran.

Entonces deberá reiniciar la secuencia de salida.

Tabla de verdad

La tabla siguiente muestra ante qué señales y de qué forma reacciona el 2PULSE.

Habilitación software SW_ENABLE	Habilitación hardware (entrada digital DI)	Salida digital DO	STS_ENABLE	Secuencia de salida
1	0 → 1	0, si el retardo a la conexión >0 1, si el retardo a la conexión =0	0 → 1	iniciar
0 → 1	no utilizado	0 si el retardo a la conexión > 0 1 si el retardo a la conexión = 0	0 → 1	iniciar
0	cualquier estado	0	0	cancelar
1	cualquier estado	el estado previo se mantiene		-
0 → 1	0	0	0	-

Ajustar tiempos mediante la base de tiempo

Mediante la base de tiempo (Página 10423) parametrizable se seleccionan la resolución y el rango de la duración del impulso y del retardo a la conexión.

Base de tiempo = 0,1 ms:	Se pueden fijar tiempos desde 0,2 ms hasta 6,5535 s con una resolución de 0,1 ms.
Base de tiempo = 1 ms:	Se pueden fijar tiempos desde 1 ms hasta 65,535 s con una resolución de 1 ms.

Ajustar y modificar la duración del impulso

La duración del impulso (Página 10424) se ajusta directamente en el programa de control como valor numérico comprendido entre 0 y 65535.

Duración del impulso = base de tiempo x valor numérico fijado

En caso de modificar la duración del impulso mientras se ejecuta una secuencia de salida, el tiempo ya transcurrido en la salida se restará de la nueva duración del impulso y el impulso se continuará emitiendo.

Reducir la duración del impulso

En caso de reducir la duración del impulso a un tiempo más corto que el tiempo ya transcurrido, la secuencia de salida se cancelará, STS_ENABLE y la salida digital DO se borran y el bit de estado ERR_PULS se activa. En la siguiente secuencia de salida se borra el bit de estado ERR_PULS.

Ajustar y modificar el retardo a la conexión en caso de utilizar la interfaz de control corta

El retardo a la conexión (Página 10423) se define en los parámetros como valor numérico entre 0 y 65535.

Retardo a la conexión parametrizado = base de tiempo x valor numérico ajustado

Con el factor para el retardo a la conexión se puede adaptar el tiempo parametrizado en el programa de control. El factor se ajusta entre 0 y 255 con una ponderación de 0,1.

Retardo a la conexión = factor x 0,1 x retardo a la conexión parametrizado

En caso de modificar el factor de retardo a la conexión durante la secuencia de salida, el nuevo retardo a la conexión tendrá efecto en la siguiente secuencia de salida.

Modo isócrono (a partir de 6ES7138-4DD01-0AB0)

En el modo de operación "Salida de impulsos" el modo isócrono no tiene ningún efecto sobre la funcionalidad.

Parámetros del modo de operación "Salida de impulsos"

Parámetros del modo de operación "Salida de impulsos"

La tabla siguiente muestra los parámetros del modo de operación "Salida de impulsos".

Parámetro	Significado	Rango de valores	Ajuste predeterminado
Modo de operación (Página 10422)	Ajuste el modo de operación "Salida de impulsos".	• Salida de impulsos • Modulación de ancho de impulsos • Tren de impulsos • Retardo a la conexión/ desconexión • Salida de frecuencia	Modulación de ancho de impulsos
Base de tiempo (Página 10423)	Con la base de tiempo se seleccionan la resolución y el rango de la duración del impulso y del retardo a la conexión.	• 0,1 ms • 1 ms	0,1 ms
Función DI (Página 10423)	La entrada digital DI se puede utilizar como entrada o como habilitación hardware.	• Entrada • Habilitación HW	Entrada
Retardo a la conexión (Página 10423) [1]	Tiempo que transcurre desde el inicio de la secuencia de salida hasta la salida del impulso. El retardo a la conexión se puede modificar en el programa de control con el "Factor de retardo a la conexión".	de 0 a 65535	0
[1] Sólo si se utiliza la interfaz de control (Página 10425) corta. Si se utiliza la interfaz de control larga, ajuste este valor directamente en la interfaz de control.			

Señales de control y realimentación del modo de operación "Salida de impulsos"

Señales de control y realimentación del modo de operación "Salida de impulsos"

La tabla siguiente muestra las señales de control y realimentación del modo de operación "Salida de impulsos".

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Señales de control en caso de utilizar la interfaz de control corta (8 bytes)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación hardware	Byte 2: Bit 0	Byte 6: Bit 0

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Duración del impulso	Tiempo en que la salida digital DO está activada una vez transcurrido el retardo a la conexión.	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535 Si se excede el límite inferior del rango, el 2PULSE no emitirá ningún impulso.	Palabra 0	Palabra 4
Factor de retardo a la conexión	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor x 0,1 x retardo a la conexión parametrizado	de 0 a 255 Si resulta un retardo a la conexión < 0,2 ms o con un factor = 0, el retardo a la conexión efectivo = 0. Si resulta un retardo a la conexión > 65,535 s, el retardo a la conexión se limita a 65,535 s.	Byte 3	Byte 7
Señales de control cuando se usa la interfaz de control larga (12 bytes) (a partir de 6ES7138-4DD01-0AB0)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación hardware	Byte 4: Bit 0	Byte 10: Bit 0
Duración del impulso	Tiempo en que la salida digital DO está activada una vez transcurrido el retardo a la conexión.	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535 Si se excede el límite inferior del rango, el 2PULSE no emitirá ningún impulso.	Palabra 0	Palabra 6
Retardo a la conexión	El retardo a la conexión se puede modificar antes de que comience la secuencia de salida.	de 0 a 65535 Retardo a la conexión = base de tiempo x valor numérico fijado	Palabra 2	Palabra 8
Señales de realimentación				
STS_ENABLE	Indica una secuencia de salida en curso.	0 = salida de impulsos bloqueada 1 = salida de impulsos en curso	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Indica el nivel de señal en la salida digital DO. Observe la velocidad de actualización.	0 = señal 0 en la salida digital DO 1 = señal 1 en la salida digital DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Indica el nivel de señal en la entrada digital DI.	0 = señal 0 en la entrada digital DI 1 = señal 1 en la entrada digital DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Indica el estado de SW_ENABLE.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 0: Bit 3	Byte 4: Bit 3
ERR_PULS	Indica un error de salida de impulsos.	0 = no hay error de salida de impulsos 1 = error de salida de impulsos	Byte 0: Bit 4	Byte 4: Bit 4
SEQ_CNT (a partir de 6ES7138-4DD01-0AB0)	Se incrementa una vez finalizada una secuencia de salida.	Sin habilitación HW: 0 … 1 Con habilitación HW: 0 … 15	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Reservado	–	0	Palabra 2	Palabra 6

Señales de entrada y de salida del modo de operación "Salida de impulsos"

Señales de entrada y de salida del modo de operación "Salida de impulsos"

La tabla siguiente muestra las señales de entrada y de salida del modo de operación "Salida de impulsos".

Señal de entrada y salida	Significado	Rango	Terminal del canal 0	Terminal del canal 1
Señal de entrada				
Habilitación HW	La habilitación HW se puede seleccionar con el parámetro "Función DI". La señal de la entrada digital DI será evaluada por el 2PULSE al iniciar la secuencia de salida.	0 = habilitación HW borrada 1 = habilitación HW obtenida 0→1 = inicio de la secuencia de salida; depende de la habilitación SW (SW_ENABLE)	1	5
Señal de salida				
Impulso en la salida digital DO	Se emite un impulso en la salida digital DO por la duración de impulso fijada.	0 = no hay impulso 1 = impulso	4	8

Modo de operación "Modulación de ancho de impulsos" (PWM)

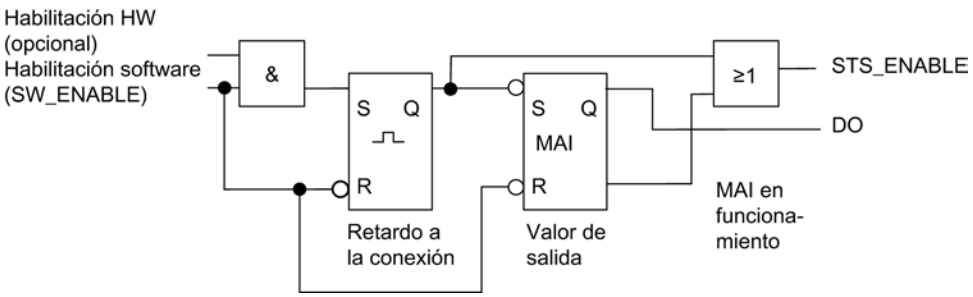
Modo de operación "Modulación de ancho de impulsos" (PWM)

Modulación de ancho de impulsos

Para el 2PULSE se especifica un valor de salida. El 2PULSE genera impulsos continuos sobre esta base. El valor de salida determina la relación impulso/pausa dentro de un período (modulación de ancho de impulsos). La duración del período es ajustable.

La secuencia de impulsos se emite una vez transcurrido el retardo a la conexión (Página 10423) parametrizado en la salida digital DO del 2PULSE (secuencia de salida).

La figura siguiente muestra el esquema de principio del modo de operación "Modulación de ancho de impulsos".



Inicio de la secuencia de salida

La habilitación de la secuencia de salida debe realizarse siempre en el programa de control por habilitación software (Página 10425) (SW_ENABLE 0 → 1; MANUAL_DO = 0).

El bit de realimentación (Página 10428)ACK_SW_ENABLE indica la habilitación software pendiente en el 2PULSE.

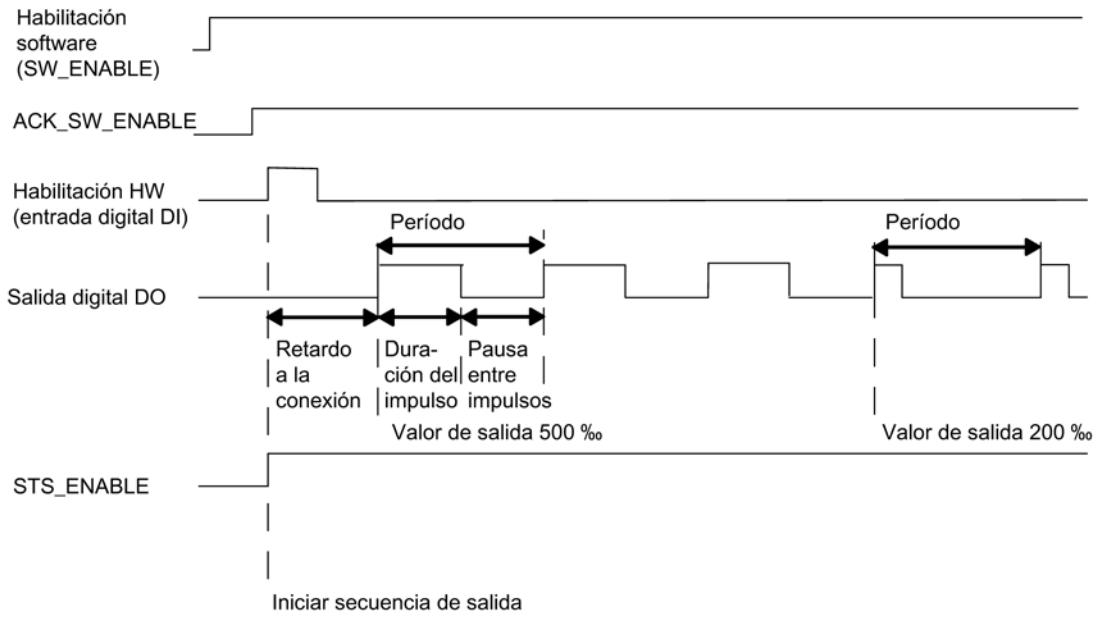
Además, es posible ajustar con el parámetro "Función DI (Página 10423)" la entrada digital DI del 2PULSE como habilitación hardware. En el modo isócrono (a partir de 6ES7138-4DD01-0AB0) la configuración de una habilitación de HW no tiene efecto.

Para trabajar simultáneamente con la habilitación software y la habilitación hardware, la secuencia de salida comienza con el primer flanco ascendente de la habilitación hardware una vez obtenida la habilitación software. El 2PULSE ignora los posteriores flancos ascendentes de la habilitación hardware durante la secuencia de salida actual.

Cuando se produce la habilitación (flanco ascendente) comienza a contar el retardo a la conexión y se activa el bit STS_ENABLE. La secuencia de impulsos se emite una vez transcurrido el retardo a la conexión. Mientras el bit SW_ENABLE está activado, la secuencia de salida se ejecuta de modo continuo.

Diagrama de impulsos

La figura siguiente muestra una secuencia de salida en el modo de operación "Modulación de ancho de impulsos".



Cancelar la secuencia de salida

Borrando la habilitación software (SW_ENABLE = 0) durante el retardo a la conexión o durante la salida de impulsos se cancela la secuencia de salida y el bit STS_ENABLE y la salida digital DO se borran.

A continuación es necesario reiniciar la secuencia de salida.

Tabla de verdad

La tabla siguiente muestra ante qué señales y de qué forma reacciona el 2PULSE.

Habilitación software SW_ENABLE	Habilitación hardware (entrada digital DI)	Salida digital DO	STS_ENABLE	Secuencia de salida
1	0 → 1	0, si el retardo a la conexión >0 1, si el retardo a la conexión =0	0 → 1	iniciar
0 → 1	no utilizado	0, si el retardo a la conexión >0 1, si el retardo a la conexión =0	0 → 1	iniciar
0	cualquier estado	0	0	cancelar
1	cualquier estado	el estado previo se mantiene		-
0 → 1	0	0	0	-

Duración mínima del impulso y pausa mínima entre impulsos

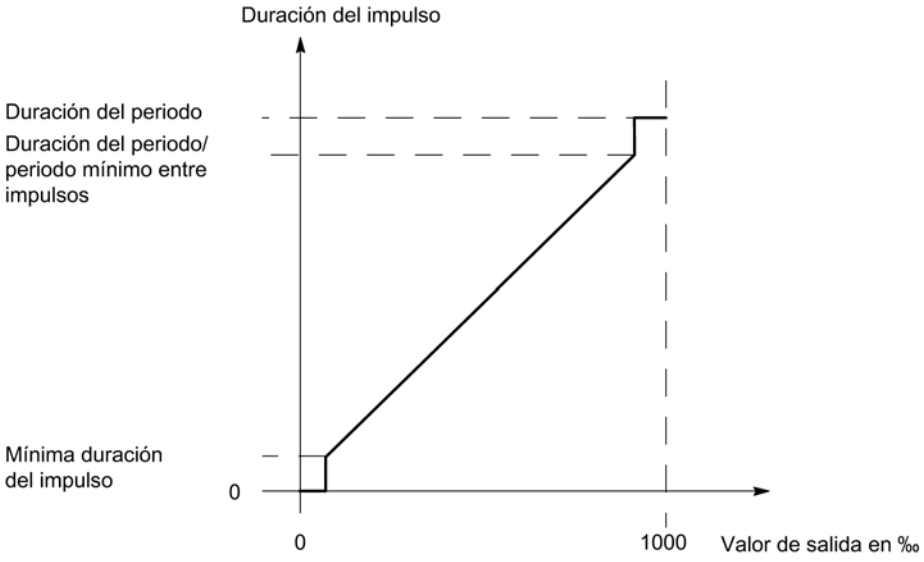
La duración mínima de impulso y la pausa mínima entre impulsos se superponen a la característica de salida proporcional.

La duración mínima de impulso y la pausa mínima entre impulsos se parametrizan con el parámetro "Duración mínima de impulso (Página 10424)"; ambos tienen siempre el mismo valor.

Una duración de impulso calculada por el 2PULSE que sea más corta que la duración mínima de impulso se inhibe.

Una duración de impulso calculada por el 2PULSE que sea más larga que el período menos la pausa mínima entre impulsos se ajusta al 1000 ‰.

La figura siguiente muestra la modulación de ancho de impulsos.



Ajustar tiempos mediante la base de tiempo

Mediante la base de tiempo (Página 10423) parametrizable se seleccionan la resolución y el rango de la duración del período, la duración mínima del impulso y del retardo a la conexión.

Base de tiempo = 0,1 ms:	Se pueden fijar tiempos desde 0,2 ms a 6,5535 s con una resolución de 0,1 ms.
Base de tiempo = 1 ms:	Se pueden fijar tiempos desde 1 ms a 65,535 s con una resolución de 1 ms.

Ajustar y modificar el valor de salida

Con el parámetro "Formato de salida (Página 10423)" se selecciona el rango del valor de salida.

- Formato de salida "Por mil": rango de valores entre 0 y 1000
  El 2PULSE calcula la duración de impulso con el valor de salida ajustado:
  Duración de impulso = (valor de salida / 1000) x período
- Formato de salida "Salida analógica S7": rango de valores entre 0 y 27648
  El 2PULSE calcula la duración de impulso con el valor de salida ajustado:
  Duración de impulso = (valor de salida / 27648) x período

El valor de salida se especifica directamente con el programa de control. El nuevo valor de salida se adopta con el siguiente flanco ascendente de la salida.

Ajuste del período

Dependiendo de la configuración, el período se define como valor de parámetro o a través de la interfaz de control. La tabla siguiente muestra cuándo se utiliza qué valor.

Para el modo isócrono (a partir de 6ES7138-4DD01-0AB0) es necesario definir el período en el mismo momento de efectuar la parametrización. A diferencia del modo no isócrono, aquí no es posible modificar posteriormente el período durante el funcionamiento.

	Configuración con …	
	Interfaz de control corta	Interfaz de control larga
Modo isócrono	Período = valor del parámetro "Período" El valor de la interfaz de control se ignora.	
Modo no isócrono	Período = valor del parámetro "Período" × "Factor de período" de la interfaz de control	Período = consigna "Período" de la interfaz de control El valor del parámetro "Período" se ignora.

Período al parametrizar con una interfaz de control corta

La duración del período (Página 10424) se define en los parámetros como valor numérico en función de la base de tiempo:

- con base de tiempo 1 ms: de 1 a 65535
- con base de tiempo 0,1 ms: de 2 a 65535

Período parametrizado = base de tiempo x valor numérico ajustado

Con el factor para el período, es posible adaptar el tiempo parametrizado en el programa de control. El factor se ajusta entre 0 y 255 con una ponderación de 0,1.

Período = factor x 0,1 x período parametrizado

Las nuevas consignas para el período se toman con el siguiente flanco ascendente de la salida.

Nota

Si se modifica el período durante el funcionamiento, éste pueda experimentar una prolongación momentánea leve (aprox. 10 µs).

Nota

Una modificación del período a través de la interfaz de control no tiene efecto en el modo isócrono (a partir de 6ES7138-4DD01-0AB0).

Ajustar la duración mínima del impulso y la pausa mínima entre impulsos

La duración mínima de impulso y la pausa mínima entre impulsos se definen como un valor numérico entre 0 y 65535 con el parámetro "Duración mínima de impulso (Página 10424)".

Duración mínima de impulso / pausa mínima entre impulsos parametrizada = base de tiempo x valor numérico ajustado

Ajustar el retardo a la conexión

El retardo a la conexión (Página 10423) se define en los parámetros como valor numérico entre 0 y 65535.

Retardo a la conexión parametrizado = base de tiempo x valor numérico ajustado

Si utiliza la interfaz de control larga, con el factor para el retardo a la conexión se puede adaptar el tiempo parametrizado en el programa de control. El factor se ajusta entre 0 y 255 con una ponderación de 0,1.

Retardo a la conexión = factor x 0,1 x retardo a la conexión parametrizado

Medición de intensidad (a partir de 6ES7138-4DD01-0AB0)

Un campo de aplicación importante de la medición de intensidad en el modo de operación "Modulación de ancho de impulsos" es la regulación de intensidad de una válvula proporcional. Esto permite, por ejemplo, compensar los efectos térmicos que actúan sobre la válvula.

Encontrará más información en "Función: Medición de intensidad (a partir de 6ES7138-4DD01-0AB0) (Página 10414)".

Modo isócrono en el modo de operación "Modulación de ancho de impulsos"

Modo isócrono

Nota

Con el 2PULSE el modo isócrono es posible a partir del número de pedido 6ES7138-4DD01-0AB0.

En el modo de operación "Modulación de ancho de impulsos" el modo isócrono tiene las siguientes particularidades:

- La salida comienza siempre sin retardo a la conexión, un retardo a la conexión parametrizado se ignora.
- La salida comienza siempre tras la obtención de la habilitación software, una habilitación hardware parametrizada se ignora.
- No es posible cambiar el período durante el funcionamiento, un ajuste a través de la interfaz de control se ignora.

En el modo isócrono la secuencia de salida está sincronizada con el instante T_o. El período se adapta al ciclo DP. En este caso es posible que el período ajustado por el usuario no se pueda ejecutar con exactitud.

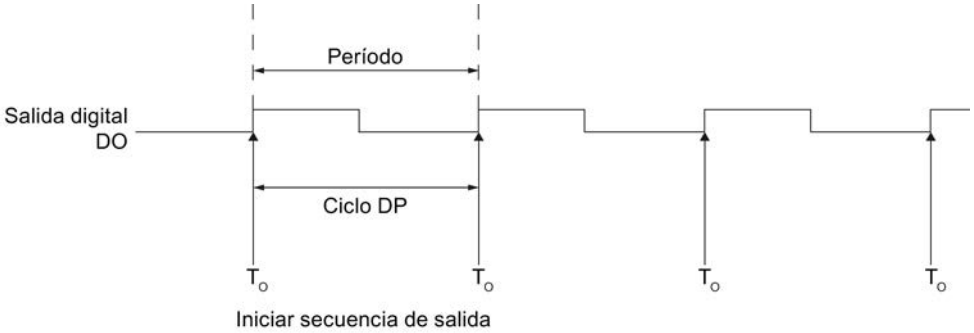
El 2PULSE adapta el valor parametrizado del período al ciclo DP con base en un algoritmo de cálculo. El cálculo se ejecuta de tal manera que se minimiza la diferencia entre el período ajustado y el calculado. En el caso menos favorable la desviación equivale a la mitad del ciclo DP. La tabla siguiente muestra ejemplos de ello.

Tiempo de ciclo DP T_{DP} [1]	Período ajustado T_{nom}	Período calculado T_{real} [1]
10 ms	5 ms	5 ms
10 ms	2 ms	2 ms
10 ms	6 ms	5 ms
10 ms	16 ms	20 ms

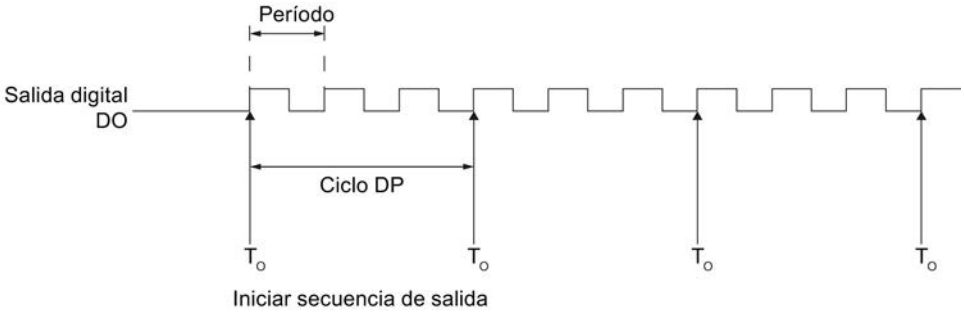
[1] La relación existente entre el período calculado y el tiempo de ciclo DP corresponde siempre a un número entero.

A continuación se representa gráficamente la relación entre el período y el ciclo DP. En los ejemplos, la relación impulso/pausa aparece representada con un 50 % respectivamente. Encontrará más información en el manual de funciones Isochrone Mode (http://support.automation.siemens.com/WW/view/en/15218045).

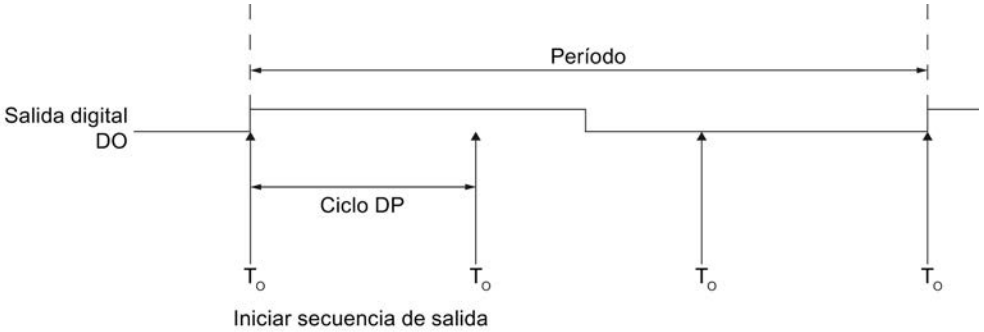
- El período es igual al ciclo DP



- El período es más corto que el ciclo DP



- El período es más largo que el ciclo DP



Parámetros del modo de operación "Modulación de ancho de impulsos"

Parámetros del modo de operación "Modulación de ancho de impulsos"

La tabla siguiente muestra los parámetros del modo de operación "Modulación de ancho de impulsos".

Parámetro	Significado	Rango	Ajuste predeterminado
Modo de operación (Página 10422)	Ajuste el modo de operación "Modulación de ancho de impulsos".	• Salida de impulsos • Modulación de ancho de impulsos • Tren de impulsos • Retardo a la conexión/ desconexión • Salida de frecuencia	Modulación de ancho de impulsos
Base de tiempo (Página 10423)	Con la base de tiempo se elige la resolución y el rango de valores del período, la duración mínima de impulso y del retardo a la conexión.	• 0,1 ms • 1 ms	0,1 ms
Formato de salida (Página 10423)	Dependiendo de la resolución requerida para el valor de salida, elija entre los formatos de salida Por mil o Salida analógica S7.	• Por mil • Salida analógica S7	Por mil
Función DI (Página 10423)	La entrada digital DI se puede utilizar como entrada o como habilitación HW.	• Entrada • Habilitación HW	Entrada
Retardo a la conexión (Página 10423)	Tiempo que transcurre desde el comienzo de la secuencia de salida hasta la salida de la secuencia de impulsos.	de 0 a 65535	0
Duración mínima de impulso (Página 10424)	Duración mínima del impulso y pausa mínima entre impulsos: Introduzca aquí el tiempo de respuesta del elemento final de control conectado a la salida digital DO.	de 0 a 65535	0
Duración del período (Página 10424) [1]	Consigna del período Al especificar el período, tenga en cuenta la duración mínima de impulso ajustada y el tiempo de respuesta del elemento final de control conectado a la salida digital DO. Si utiliza la interfaz de control corta, en el modo no isócrono se puede modificar el período en el programa de control con el "Factor de período".	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535	20000 → 2 s
[1] Observe la información referente al "Ajuste del período (Página 10380)" y al "Modo isócrono (Página 10385)".			

Señales de control y realimentación del modo de operación "Modulación de ancho de impulsos"

Señales de control y realimentación del modo de operación "Modulación de ancho de impulsos"

La tabla siguiente muestra las señales de control y realimentación del modo de operación "Modulación de ancho de impulsos".

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Señales de control en caso de utilizar la interfaz de control corta (8 bytes)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación hardware	Byte 2: Bit 0	Byte 6: Bit 0
Valor de salida	El valor que se aplica a la salida digital DO con modulación de ancho de impulsos.	Dependiendo del formato de salida: • Por mil: 0 ... 1000 • Salida analógica S7: 0 ... 27648 Si se introduce un valor de salida > 1000 ó 27648, el 2PULSE lo limitará a 1000 ó 27648, respectivamente.	Palabra 0	Palabra 4
Factor de período	La duración parametrizada del período se puede modificar: Período = factor x 0,1 x período parametrizado	Factor: 0 a 255 Período: 2 × duración mínima de impulso a 65,635 s. Si resulta un período < 2 × duración mínima de impulso o < 200 µs o si el factor = 0, el período se limita a 2 × duración mínima de impulso (mínimo 200 µs). Si resulta una duración del período > 65.535 s, se limitará a 65.535 s.	Byte 3	Byte 7
Señales de control cuando se usa la interfaz de control larga (12 bytes) (a partir de 6ES7138-4DD01-0AB0)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación hardware	Byte 4: Bit 0	Byte 10: Bit 0
Valor de salida	Valor que se emite en la salida digital DO con modulación de ancho de impulsos.	Dependiendo del formato de salida: • Por mil: 0 ... 1000 • Salida analógica S7: 0 ... 27648 Si se introduce un valor de salida > 1000 ó 27648, el 2PULSE lo limitará a 1000 ó 27648 respectivamente.	Palabra 0	Palabra 6

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Período [1]	Consigna del período en modo no isócrono	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535	Palabra 2	Palabra 8
Factor de retardo a la conexión	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor x 0,1 x retardo a la conexión parametrizado	de 0 a 255 Si resulta un retardo a la conexión < 0,2 ms o con un factor = 0, el retardo a la conexión efectivo = 0. Si resulta un retardo a la conexión > 65,535 s, el retardo a la conexión se limita a 65,535 s.	Byte 5	Byte 11
Señales de realimentación				
STS_ENABLE	Indica una secuencia de salida en curso.	0 = salida de impulsos bloqueada 1 = salida de impulsos en curso	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Indica el nivel de señal en la salida digital DO. Observe la velocidad de actualización.	0 = señal 0 en la salida digital DO 1 = señal 1 en la salida digital DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Indica el nivel de señal en la entrada digital DI.	0 = señal 0 en la entrada digital DI 1 = señal 1 en la entrada digital DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Indica el estado de SW_ENABLE.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 0: Bit 3	Byte 4: Bit 3
SEQ_CNT (a partir de 6ES7138-4DD01-0AB 0)	Función no disponible en este modo de operación.	–	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Valor de medición de intensidad (a partir de 6ES7138-4DD01-0AB 0)	Intensidad en la DO como valor analógico SIMATIC S7.	0 … 27648 (margen de saturación hasta 32767)	Palabra 2	Palabra 6
[1] Observe la información referente al "Ajuste del período (Página 10380)".				

Señales de entrada y de salida del modo de operación "Modulación de ancho de impulsos"

Señales de entrada y de salida del modo de operación "Modulación de ancho de impulsos"

La tabla siguiente muestra las señales de entrada y de salida del modo de operación "Modulación de ancho de impulsos".

Señal de entrada y salida	Significado	Rango	Terminal del canal 0	Terminal del canal 1
Señal de entrada				
Habilitación HW	La habilitación HW se puede seleccionar con el parámetro "Función DI". La señal de la entrada digital DI será evaluada por el 2PULSE al iniciar la secuencia de salida.	0 = habilitación HW borrada 1 = habilitación HW obtenida 0 → 1 = inicio de la secuencia de salida; depende de la habilitación SW	1	5
Señal de salida				
Secuencia de impulsos en la salida digital DO	La secuencia de impulsos se emite en la salida digital DO.	0 = no hay impulso 1 = impulso	4	8

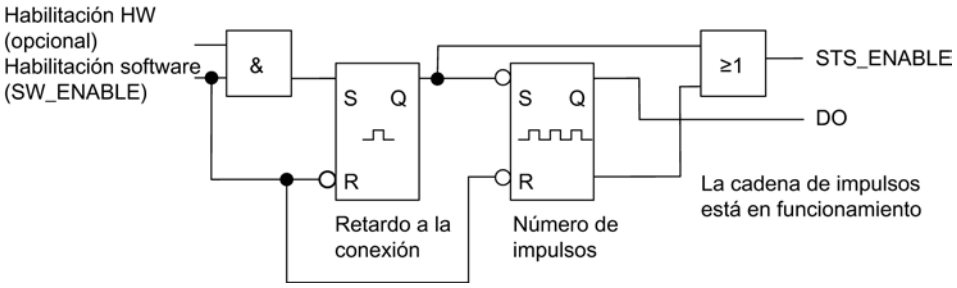
Modo "Tren de impulsos"

Modo "Tren de impulsos"

Definición

El 2PULSE emite el número de impulsos predeterminado por el usuario como tren de impulsos en la salida digital DO (secuencia salida) una vez transcurrido el retardo a la conexión (Página 10423) ajustado. El período y la duración del impulso se pueden ajustar.

La figura siguiente muestra el esquema de principio del modo de operación "Tren de impulsos".



Iniciar la secuencia de salida

La habilitación de la secuencia de salida debe realizarse siempre en el programa de control por habilitación software (Página 10425) (SW_ENABLE 0 → 1; MANUAL_DO = 0). El bit de

realimentación (Página 10428)ACK_SW_ENABLE indica la habilitación software pendiente en el 2PULSE.

Además, es posible ajustar con el parámetro "Función DI (Página 10423)" la entrada digital DI del 2PULSE como habilitación hardware.

Para trabajar simultáneamente con la habilitación software y la habilitación hardware, la secuencia de salida comienza con el primer flanco ascendente de la habilitación hardware una vez obtenida la habilitación software. El 2PULSE ignora los posteriores flancos ascendentes de la habilitación hardware durante la secuencia de salida actual. Una vez obtenida la habilitación software, un flanco ascendente de la habilitación hardware es suficiente para iniciar la siguiente secuencia de salida.

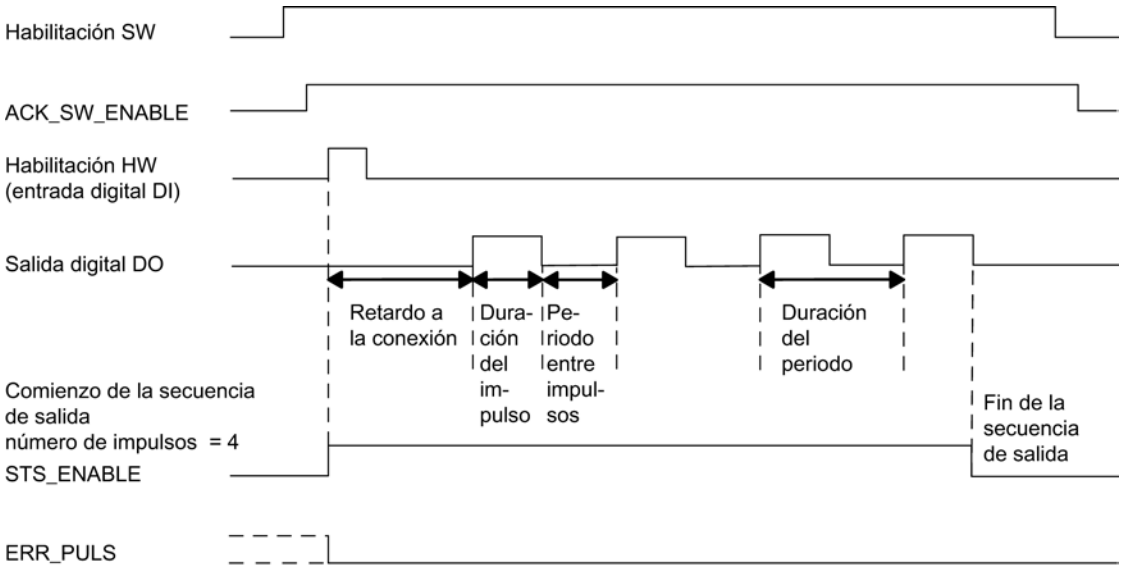
Cuando se produce la habilitación (flanco ascendente) comienza a contar el retardo a la conexión y se activa el bit STS_ENABLE. Una vez transcurrido el retardo a la conexión, el tren de impulsos se emite con el número de impulsos fijado. La secuencia de salida termina (Página 10413) tan pronto como se emite el último impulso; STS_ENABLE se borra.

En caso de modificar el número de impulsos durante el funcionamiento de forma no admisible, la señal (Página 10428)ERR_PULS indica un error en la salida de impulsos.

En la siguiente secuencia de salida, el 2PULSE borrará el bit de realimentación ERR_PULS.

Diagrama de impulsos

La figura siguiente muestra una secuencia de salida en el modo de operación "Tren de impulsos".



Cancelar la secuencia de salida

Borrando la habilitación software durante el retardo a la conexión o durante el tren de impulsos, se cancela la secuencia de salida y el bit STS_ENABLE y la salida digital DO se borran.

Entonces deberá reiniciar la secuencia de salida.

Tabla de verdad

La tabla siguiente muestra ante qué señales y de qué forma reacciona el 2PULSE.

Habilitación software SW_ENABLE	Habilitación hardware (entrada digital DI)	Salida digital DO	STS_ENABLE	Secuencia de salida
1	0 → 1	0, si el retardo a la conexión >0 1, si el retardo a la conexión =0	0 → 1	iniciar
0 → 1	no utilizado	0 si el retardo a la conexión > 0 1 si el retardo a la conexión = 0	0 → 1	iniciar
0	cualquier estado	0	0	cancelar
1	cualquier estado	el estado previo se mantiene		-
0 → 1	0	0	0	-

Ajustar tiempos mediante la base de tiempo

Mediante la base de tiempo (Página 10423) parametrizable se seleccionan la resolución y el rango de valores de la duración del período, de la duración del impulso y del retardo a la conexión.

Base de tiempo = 0,1 ms:	Se pueden fijar tiempos desde 0,2 ms a 6,5535 s con una resolución de 0,1 ms.
Base de tiempo = 1 ms:	Se pueden fijar tiempos desde 1 ms a 65,535 s con una resolución de 1 ms.

Ajustar y modificar el número de impulsos

El número de impulsos (Página 10425) se ajusta directamente con el programa de control como valor numérico entre 0 y 65535.

Si se modifica el número de impulsos una vez transcurrido el retardo a la conexión, el nuevo valor será efectivo de inmediato:

- Si se aumenta el número de impulsos, se emitirá el nuevo número de impulsos más grande
- Si se reduce el número de impulsos y ya ha sido emitido este número menor de impulsos, se interrumpe la secuencia de salida, se borran el bit STS_ENABLE y la salida digital DO y se activa ERR_PULS. En la siguiente secuencia de salida se borra ERR_PULS .

Ajustar y modificar el período en caso de utilizar la interfaz de control corta

La duración del período (Página 10424) se define en los parámetros como un valor numérico entre 2 y 65535:

Período parametrizado = base de tiempo x valor numérico ajustado

Con el factor para el período, es posible adaptar el tiempo parametrizado en el programa de control. El factor se ajusta entre 0 y 255 con un peso de 0,1.

Período = factor x 0,1 x período parametrizado

En caso de modificar el factor durante la secuencia de salida, el nuevo período tendrá efecto al comenzar la siguiente secuencia de salida.

Ajustar la duración de impulso

La duración del impulso (Página 10424) se define en el parámetro "Duración del impulso" como un valor numérico entre 1 y 65535:

Duración de impulso parametrizada = base de tiempo x valor numérico fijado

Si utiliza la interfaz de control larga es posible ajustar la relación entre la duración de impulso y la pausa entre impulsos:

Relación impulso/pausa = (consigna / 255) × período

Ajustar el retardo a la conexión

El retardo a la conexión (Página 10423) se define en los parámetros como valor numérico entre 0 y 65535.

Retardo a la conexión parametrizado = base de tiempo x valor numérico ajustado

Modo isócrono (a partir de 6ES7138-4DD01-0AB0)

En el modo de operación "Tren de impulsos" el modo isócrono no tiene ningún efecto en la funcionalidad.

Medición de intensidad (a partir de 6ES7138-4DD01-0AB0)

Para más información, ver "Función: Medición de intensidad (a partir de 6ES7138-4DD01-0AB0) (Página 10414)".

Parámetros del modo de operación "Tren de impulsos"

Parámetros del modo de operación "Tren de impulsos"

La tabla siguiente muestra los parámetros del modo de operación "Tren de impulsos".

Parámetro	Significado	Rango	Ajuste predeterminado
Modo de operación (Página 10422)	Ajuste el modo de operación "Tren de impulsos".	• Salida de impulsos • Modulación de ancho de impulsos • Tren de impulsos • Retardo a la conexión/ desconexión • Salida de frecuencia	Modulación de ancho de impulsos
Base de tiempo (Página 10423)	Con la base de tiempo se seleccionan la resolución y el rango de valores del período, de la duración de impulso y del retardo a la conexión.	• 0,1 ms • 1 ms	0,1 ms
Función DI (Página 10423)	La entrada digital DI se puede utilizar como entrada o como habilitación HW.	• Entrada • Habilitación HW	Entrada
Retardo a la conexión (Página 10423)	Tiempo que transcurre desde el comienzo de la secuencia de salida hasta la salida del tren de impulsos.	de 0 a 65535	0
Duración del impulso (Página 10424) [1]	Duración de impulso: Introduzca aquí el tiempo de realimentación del elemento final de control conectado a la salida digital DO.	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535 Si se sobrepasa el límite inferior del rango, el 2PULSE fija la duración de impulso a 0,2 ms o 1 ms.	10000 → 1 s
Duración del período (Página 10424) [1]	La duración del período debería ser siempre un múltiplo del tiempo de respuesta del elemento final de control conectado a la salida digital DO. Defina el período de acuerdo con la tasa de repetición necesaria de los impulsos. El período se puede modificar en el programa de control con el "Factor de período".	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535	20000 → 2 s

[1] Sólo si se utiliza la interfaz de control (Página 10425) corta.
Si se utiliza la interfaz de control larga, ajuste este valor directamente en la interfaz de control.

Señales de control y realimentación del modo de operación "Tren de impulsos"

Señales de control y realimentación del modo de operación "Tren de impulsos"

La tabla siguiente muestra las señales de control y realimentación del modo de operación "Tren de impulsos".

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Señales de control en caso de utilizar la interfaz de control corta (8 bytes)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación HW	Byte 2: Bit 0	Byte 6: Bit 0
Número de impulsos	Número de impulsos que se emiten en la salida digital DO una vez transcurrido el retardo a la conexión.	de 0 a 65535 Si el número de impulsos es 0, el 2PULSE no emitirá ningún impulso. La secuencia de salida termina con ERR_PULS = 1.	Palabra 0	Palabra 4
Factor de período	El período parametrizado se puede modificar antes de que comience la secuencia de salida: Período = factor x 0,1 x período parametrizado	Factor: de 0 a 255 Duración del período: > duración del impulso hasta 65,535 s Si resulta un período > 65,535 s, éste se ajusta a 65,535 s. Si resulta un período ≤ duración del impulso, ésta se ajusta a la duración del impulso + 0,2 ms (o bien duración del impulso + 1 ms).	Byte 3	Byte 7
Señales de control cuando se usa la interfaz de control larga (12 bytes) (a partir de 6ES7138-4DD01-0AB0)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación HW	Byte 4: Bit 0	Byte 10: Bit 0
Número de impulsos	Número de impulsos que se emiten en la salida digital DO una vez transcurrido el retardo a la conexión.	de 0 a 65535 Si el número de impulsos es 0, el 2PULSE no emitirá ningún impulso. La secuencia de salida termina con ERR_PULS = 1.	Palabra 0	Palabra 6
Duración del período	Consigna del período	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535	Palabra 2	Palabra 8
Relación impulso-pausa	Es posible ajustar la relación entre la duración del impulso y la pausa entre impulsos.	de 0 a 255 P. ej. el valor 128 corresponde a una relación de 50 : 50.	Byte 5	Byte 11
Señales de respuesta				

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
STS_ENABLE	Indica una secuencia de salida en curso.	0 = salida de impulsos bloqueada 1 = salida de impulsos en curso	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Indica el nivel de señal en la salida digital DO. Observe la velocidad de actualización.	0 = señal 0 en salida digital DO 1 = señal 1 en salida digital DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Indica el nivel de señal en la entrada digital DI.	0 = señal 0 en la entrada digital DI 1 = señal 1 en la entrada digital DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Indica el estado de SW_ENABLE.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 0: Bit 3	Byte 4: Bit 3
ERR_PULS	Indica un error de salida de impulsos.	0 = no hay error de salida de impulsos 1 = error de salida de impulsos	Byte 0: Bit 4	Byte 4: Bit 4
SEQ_CNT (a partir de 6ES7138-4DD01-0AB 0)	Se incrementa una vez finalizada una secuencia de salida.	Sin habilitación HW: 0 … 1 Con habilitación HW: 0 … 15	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Valor de medición de intensidad (a partir de 6ES7138-4DD01-0AB 0)	Intensidad en la DO como valor analógico SIMATIC S7.	0 … 27648 (margen de saturación hasta 32767)	Palabra 2	Palabra 6

Señales de entrada y de salida del modo de operación "Tren de impulsos"

Señales de entrada y de salida del modo de operación "Tren de impulsos"

La tabla siguiente muestra las señales de entrada y de salida del modo de operación "Tren de impulsos".

Señal de entrada y salida	Significado	Rango	Terminal del canal 0	Terminal del canal 1
Señal de entrada				
Habilitación HW	La habilitación HW se puede seleccionar con el parámetro "Función DI". La señal de la entrada digital DI será evaluada por el 2PULSE al inicio.	0 = habilitación HW borrada 1 = habilitación HW obtenida 0 → 1 = inicio de la secuencia de salida; depende de la habilitación SW (SW_ENABLE).	1	5
Señal de salida				
Tren de impulsos en la salida digital DO	El número de impulsos definido se emite en la salida digital DO.	0 = no hay impulso 1 = impulso	4	8

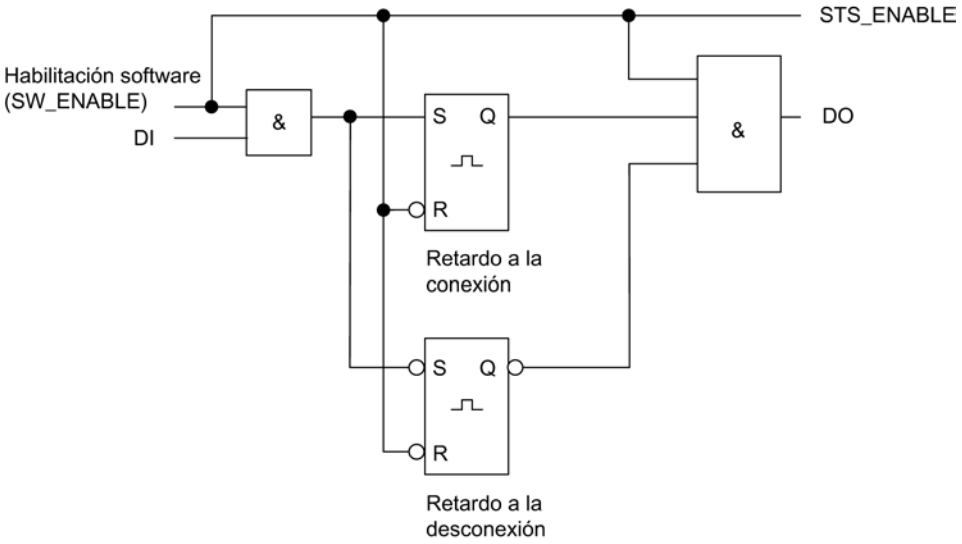
Modo "Retardo a la conexión/desconexión"

Modo "Retardo a la conexión/desconexión"

Definición

La señal pendiente en la entrada digital DI es aplicada por el 2PULSE a la salida digital DO con un retardo a la conexión y desconexión.

La figura siguiente muestra el esquema de principio del modo de operación "Retardo a la conexión/desconexión".



Habilitación de la secuencia de salida

La habilitación para la secuencia de salida debe realizarse siempre en el programa de control por habilitación software (Página 10425) (SW_ENABLE 0 → 1; MANUAL_DO = 0), con lo que STS_ENABLE se activa. El bit de realimentación (Página 10428)ACK_SW_ENABLE indica la habilitación software pendiente en el 2PULSE.

El flanco ascendente en la entrada digital DI (0 → 1) inicia el retardo a la conexión y, una vez transcurrido el retardo a la conexión, se activa la salida digital DO (Página 10413).

El flanco ascendente en la entrada digital DI (1 → 0) inicia el retardo a la desconexión y, una vez transcurrido el retardo a la desconexión, se borra la salida digital DO (Página 10413).

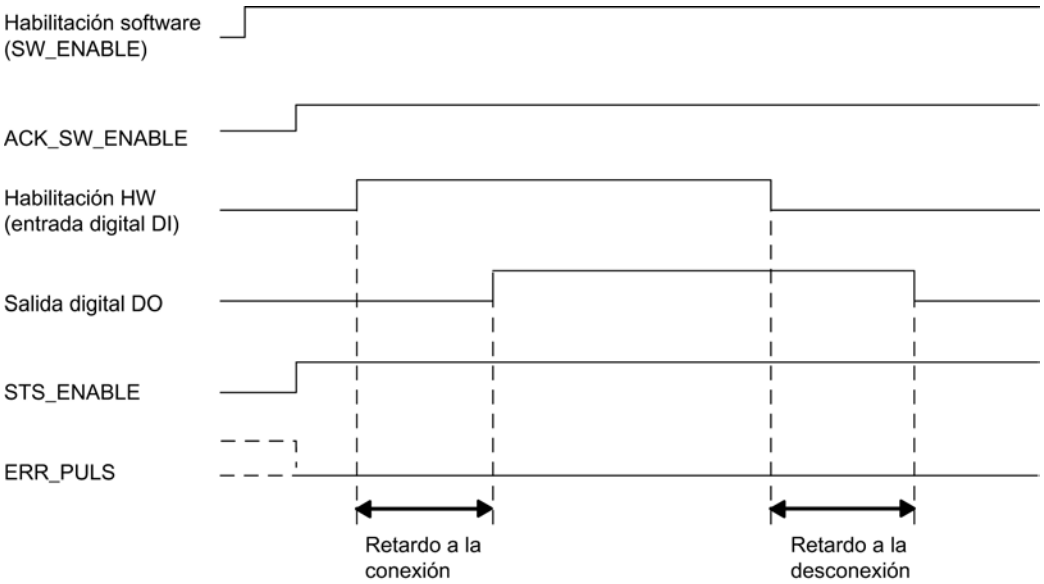
Si el 2PULSE detecta una duración del impulso o una pausa entre impulsos demasiado corta, indica el error de salida de impulsos (Página 10428)ERR_PULS.

Con el siguiente flanco en la entrada digital DI, el 2PULSE borra el bit de realimentación ERR_PULS.

Diagrama de impulsos

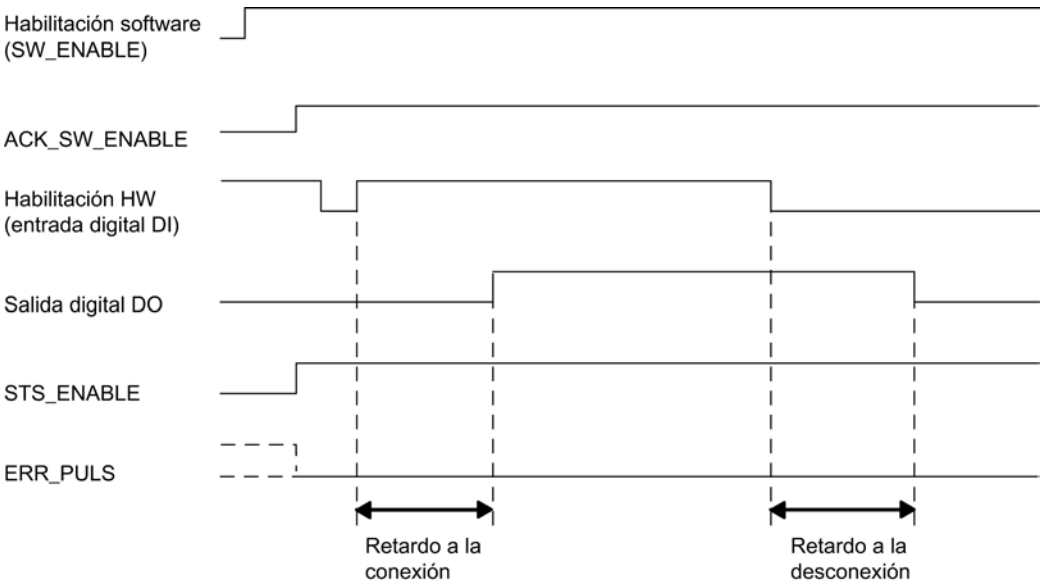
SW_ENABLE se activa mientras DI = 0:

La figura siguiente muestra una secuencia de salida en el modo de operación "Retardo a la conexión/desconexión" cuando se inicia con DI = 0.



SW_ENABLE se activa mientras DI = 1:

Si se activa SW_ENABLE mientras DI = 1, el primer flanco en DI (flanco descendente) se ignora. La figura siguiente muestra la secuencia de salida correspondiente.



Cancelación de la secuencia de salida

Borrando la habilitación software (SW_ENABLE = 0) durante la secuencia de salida, ésta se cancela y se borran el bit STS_ENABLE y la salida digital.

Tabla de verdad

La tabla siguiente muestra ante qué señales y de qué forma reacciona el 2PULSE.

Habilitación software SW_ENABLE	Entrada digital DI	Salida digital DO	STS_ENABLE	Secuencia de salida
1	0 → 1	0 si el retardo a la conexión > 0 1 si el retardo a la conexión = 0	1	iniciar
1	1 → 0	1, si el retardo a la desconexión > 0 0, si el retardo a la desconexión = 0	1	iniciar
0	cualquier estado	0	0	cancelar
1	cualquier estado	el estado previo se mantiene	1	-
0 → 1	0	0	1	-

Duración/pausa mínima del impulso de la salida digital DO

La duración/pausa mínima de impulso de la salida digital DO es de 100 µs.

Esto deberá tenerse en cuenta al ajustar el retardo a la conexión/desconexión y la duración/pausa del impulso de la entrada digital DI;de lo contrario, la reacción en la salida digital DO no está definida.

La duración de impulso de la entrada digital DI es demasiado corta

El 2PULSE detecta un impulso demasiado corto en el flanco descendente de la entrada digital DI cuando:

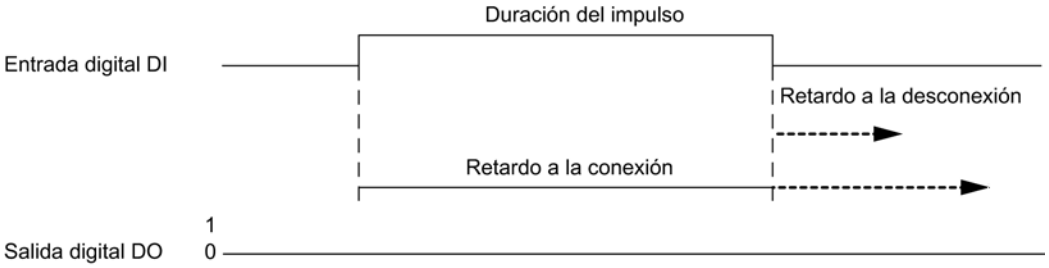
Duración del impulso + retardo a la desconexión ≤ retardo a la conexión.

Reacción del 2PULSE a una duración de impulso demasiado corta:

- ERR_PULS se activa
- el actual retardo a la conexión se borra.
- el retardo a la desconexión no se inicia.
- el nivel de señal en la salida digital DO permanece a 0.

ERR_PULS se borra con el siguiente flanco ascendente en la entrada digital DI.

La figura siguiente muestra el comportamiento cuando la duración del impulso es demasiado breve.



Pausa entre impulsos de la entrada digital DI demasiado corta

El 2PULSE detecta en el flanco ascendente de la entrada digital DI una pausa demasiado corta entre impulsos, cuando:

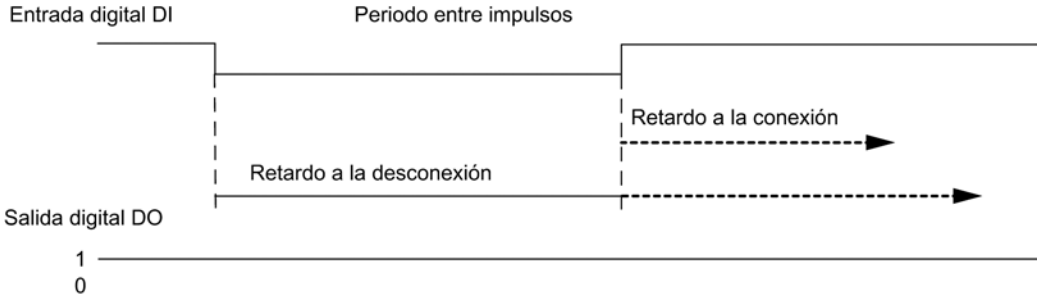
Pausa entre impulsos + retardo a la conexión ≤ retardo a la desconexión.

Reacción del 2PULSE a una pausa demasiado corta entre impulsos:

- ERR_PULS se activa
- el actual retardo a la desconexión se borra.
- el retardo a la conexión no se inicia.
- el nivel de señal en la salida digital DO permanece a 1.

ERR_PULS se borra con el siguiente flanco descendente en la entrada digital DI.

La figura siguiente muestra el comportamiento cuando la pausa entre impulsos es demasiado breve.



Redisparo del actual retardo a la conexión

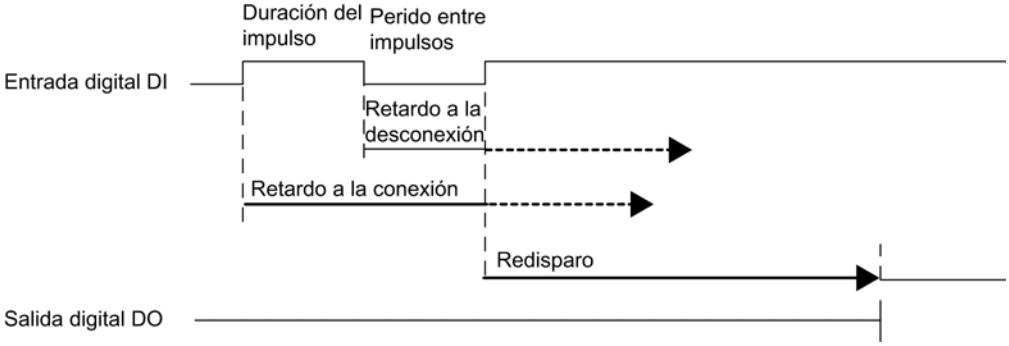
El 2PULSE reinicia el retardo a la conexión con el flanco ascendente en la entrada digital DI cuando:

Retardo a la conexión > duración del impulso + pausa entre impulsos.

Esto desactiva el actual retardo a la desconexión.

La salida digital DO sólo se activa si hay un nivel de señal 1 en la entrada digital DI durante un tiempo superior al del retardo a la conexión. Esto permite filtrar secuencias de impulsos rápidas.

La figura siguiente muestra el redisparo del actual retardo a la conexión.



Redisparo del actual retardo a la desconexión

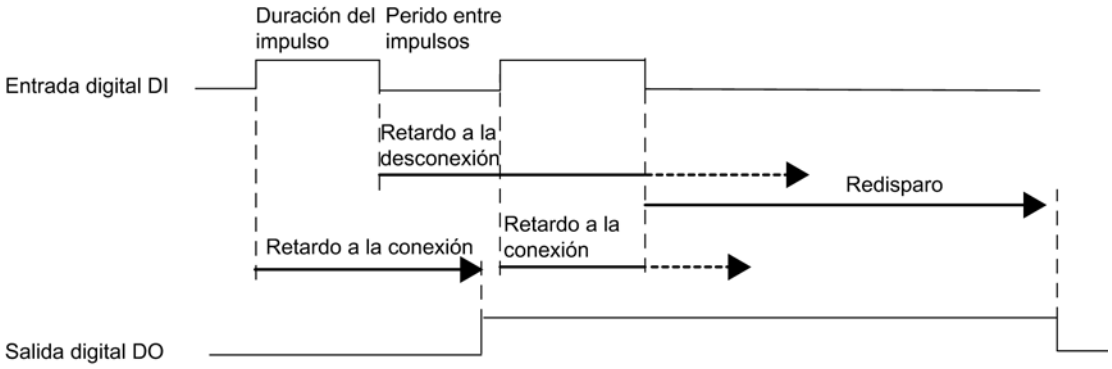
El 2PULSE reinicia el retardo a la desconexión con el flanco descendente de la entrada digital DI si:

Retardo a la desconexión > duración del impulso + pausa entre impulsos.

Esto desactiva el actual retardo a la conexión.

La salida digital DO se borra sólo si hay un nivel de señal 0 en la entrada digital DI durante un tiempo superior al del retardo a la desconexión.

La figura siguiente muestra el redisparo del actual retardo a la desconexión.



Ajustar tiempos mediante la base de tiempo

Mediante la base de tiempo (Página 10423) parametrizable se seleccionan la resolución y el rango del retardo a la conexión y del retardo a la desconexión.

Base de tiempo = 0,1 ms:	Se pueden fijar tiempos desde 0,2 ms a 6,5535 s con una resolución de 0,1 ms.
Base de tiempo = 1 ms:	Se pueden fijar tiempos desde 1 ms a 65,535 s con una resolución de 1 ms.

Ajustar y modificar el retardo a la conexión en caso de utilizar la interfaz de control corta

El retardo a la conexión (Página 10423) se define en los parámetros como valor numérico entre 0 y 65535.

Retardo a la conexión parametrizado = base de tiempo × valor numérico ajustado

Con el factor para el retardo a la conexión se puede adaptar el tiempo parametrizado en el programa de control. El factor se ajusta entre 0 y 255 con una ponderación de 0,1:

Retardo a la conexión = factor x 0,1 x retardo a la conexión parametrizado

Si se cambia el factor de retardo a la conexión, el nuevo retardo a la conexión se activa con el siguiente flanco ascendente en la entrada digital DI.

Ajustar y modificar el retardo a la desconexión

El retardo a la desconexión se ajusta directamente como valor numérico entre 0 y 65535 en el programa de control (Página 10425):

Retardo a la desconexión = base de tiempo x valor numérico fijado

Si se cambia el factor de retardo a la desconexión, el nuevo retardo a la desconexión se activa con el siguiente flanco descendente en la entrada digital DI.

Modo isócrono (a partir de 6ES7138-4DD01-0AB0)

En el modo de operación "Retardo a la conexión/desconexión" el modo isócrono no tiene ningún efecto en la funcionalidad.

Parámetros del modo de operación "Retardo a la conexión/desconexión"

Parámetros del modo de operación "Retardo a la conexión/desconexión"

La tabla siguiente muestra los parámetros del modo de operación "Retardo a la conexión/desconexión".

Parámetro	Significado	Rango	Ajuste predeterminado
Modo de operación (Página 10422)	Ajuste el modo de operación "Retardo a la conexión/desconexión".	• Salida de impulsos • Modulación de ancho de impulsos • Tren de impulsos • Retardo a la conexión/desconexión • Salida de frecuencia	Modulación de ancho de impulsos
Base de tiempo (Página 10423)	Con la base de tiempo se seleccionan la resolución y el rango del retardo a la conexión y del retardo a la desconexión.	• 0,1 ms • 1 ms	0,1 ms
Retardo a la conexión (Página 10423) [1]	Tiempo que transcurre entre el flanco ascendente de la entrada digital DI y su salida a través de la salida digital DO. El retardo a la conexión se puede modificar en el programa de control con el "Factor de retardo a la conexión".	de 0 a 65535	0
[1] Sólo si se utiliza la interfaz de control (Página 10425) corta. Si se utiliza la interfaz de control larga, ajuste este valor directamente en la interfaz de control.			

Señales de control y realimentación del modo de operación "Retardo a la conexión/desconexión"

Señales de control y realimentación del modo de operación "Retardo a la conexión/desconexión"

La tabla siguiente muestra las señales de control y realimentación del modo de operación "Retardo a la conexión/desconexión".

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Señales de control en caso de utilizar la interfaz de control corta (8 bytes)				
Habilitación software (SW_ENABLE)	La habilitación software debe activarse siempre en el programa de control. Si se desactiva la habilitación software, la actual secuencia de salida se cancela.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 2: Bit 0	Byte 6: Bit 0
Retardo a la desconexión	Tiempo que transcurre entre el flanco descendente de la entrada digital DI y su salida a través de la salida digital DO.	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535 Si se excede el límite inferior del rango, el retardo a la desconexión no tiene efecto.	Palabra 0	Palabra 4
Factor de retardo a la conexión	El retardo a la conexión parametrizado se puede modificar: Retardo a la conexión = factor x 0,1 x retardo a la conexión parametrizado	Factor: de 0 a 255 Retardo a la conexión: 0,2 ms a 65,535 s Si resulta un retardo a la conexión < 0,2 ms o con un factor = 0, el retardo a la conexión efectivo = 0. Si resulta un retardo a la conexión > 65,535 s, el retardo a la conexión se limita a 65,535 s.	Byte 3	Byte 7
Señales de control cuando se usa la interfaz de control larga (12 bytes) (a partir de 6ES7138-4DD01-0AB0)				
Habilitación software (SW_ENABLE)	La habilitación software debe activarse siempre en el programa de control. Si se desactiva la habilitación software, la actual secuencia de salida se cancela.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 4: Bit 0	Byte 10: Bit 0
Retardo a la desconexión	Tiempo que transcurre entre el flanco descendente de la entrada digital DI y su salida a través de la salida digital DO.	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535 Si se excede el límite inferior del rango, el retardo a la desconexión no tiene efecto.	Palabra 0	Palabra 6

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Retardo a la conexión	Tiempo que transcurre entre el flanco ascendente de la entrada digital DI y su salida a través de la salida digital DO.	con base de tiempo 0,1 ms: de 2 a 65535 con base de tiempo 1 ms: de 1 a 65535 Si se excede el límite inferior del rango, el retardo a la conexión no tiene efecto.	Palabra 2	Palabra 8
Señales de respuesta				
STS_ENABLE	Indica el estado de la habilitación software (SW_ENABLE).	0 = habilitación software bloqueada 1 = habilitación SW obtenida	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Indica el nivel de señal en la salida digital DO. Observe la velocidad de actualización.	0 = señal 0 en salida digital DO 1 = señal 1 en salida digital DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Indica el nivel de señal en la entrada digital DI.	0 = señal 0 en la entrada digital DI 1 = señal 1 en la entrada digital DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Indica el estado de SW_ENABLE.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 0: Bit 3	Byte 4: Bit 3
ERR_PULS	Indica un error en la salida de impulsos si la duración del impulso o la pausa entre impulsos es demasiado corta.	0 = no hay error de salida de impulsos 1 = error de salida de impulsos	Byte 0: Bit 4	Byte 4: Bit 4
SEQ_CNT (a partir de 6ES7138-4DD01-0AB 0)	Se incrementa en la DO con cada flanco (ascendente y descendente).	0 … 15	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Reservado	–	0	Palabra 2	Palabra 6

Señales de entrada y de salida del modo de operación "Retardo a la conexión/desconexión"

Señales de entrada y de salida del modo de operación "Retardo a la conexión/desconexión"

La tabla siguiente muestra las señales de entrada y de salida del modo de operación "Retardo a la conexión/desconexión".

Señal de entrada y salida	Significado	Rango	Terminal del canal 0	Terminal del canal 1
Señal de entrada				
Entrada digital DI	La señal de la entrada digital DI es emitida por el 2PULSE con un retardo a la conexión/desconexión en la salida digital DO.	0 = no hay impulso 1 = impulso	1	5

Señal de entrada y salida	Significado	Rango	Terminal del canal 0	Terminal del canal 1
Señal de salida				
Impulso en la salida digital DO	La señal de la entrada digital DI es emitida por el 2PULSE con un retardo a la conexión/ desconexión en la salida digital DO.	0 = no hay señal 1 = señal	4	8

Modo de operación "Salida de frecuencia" (a partir de 6ES7138-4DD01-0AB0)

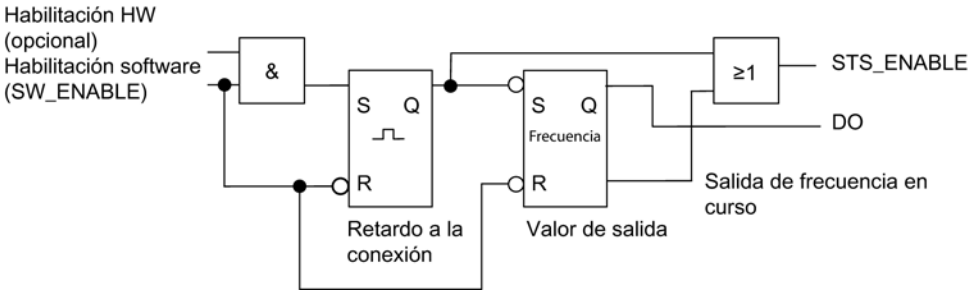
Modo "Salida de frecuencia"

Definición

Salida de una señal de onda cuadrada en la salida digital del 2PULSE con una frecuencia definida.

La secuencia de salida se inicia una vez transcurrido el retardo a la conexión (Página 10423) parametrizado en la salida digital DO del 2PULSE.

La figura siguiente muestra el esquema de principio para el modo de operación "Salida de frecuencia".



Inicio de la secuencia de salida

La habilitación para la secuencia de salida debe realizarse siempre en el programa de control por habilitación software (Página 10425) (SW_ENABLE 0 → 1; MANUAL_DO = 0).

El bit de respuesta (Página 10428)ACK_SW_ENABLE indica la habilitación software pendiente en el 2PULSE.

Además, es posible ajustar con el parámetro "Función DI (Página 10423)" la entrada digital DI del 2PULSE como habilitación hardware.

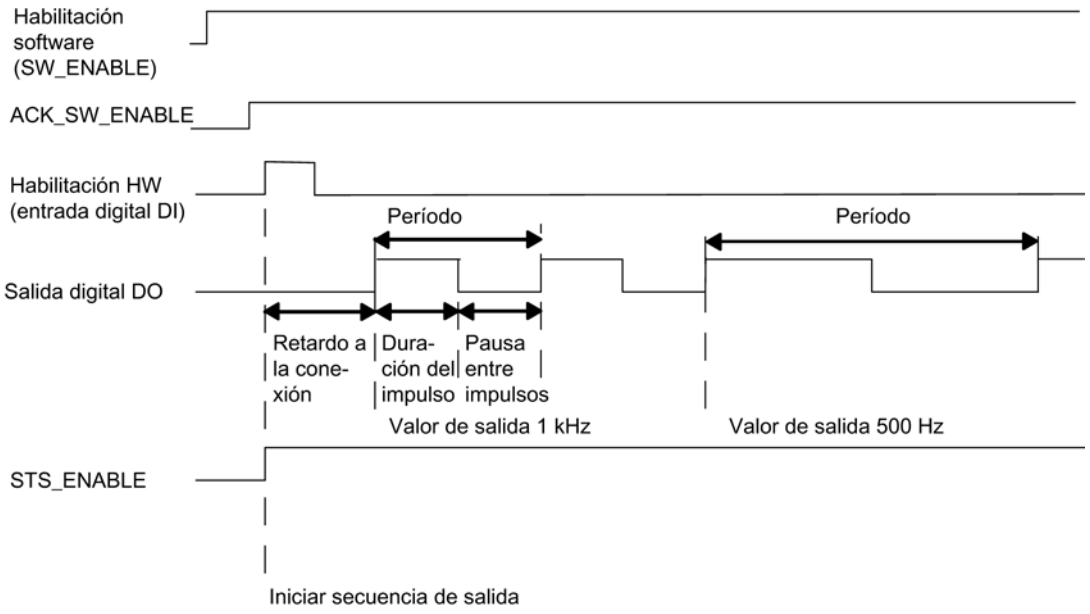
Para trabajar simultáneamente con la habilitación software y la habilitación hardware, la secuencia de salida se comienza con el primer flanco ascendente de la habilitación hardware una vez obtenida la habilitación software. El 2PULSE ignora los posteriores flancos ascendentes de la habilitación hardware durante la secuencia de salida actual.

Cuando se produce la habilitación (flanco ascendente) comienza a contar el retardo a la conexión y se activa el bit STS_ENABLE. La salida de frecuencia se inicia una vez transcurrido

el retardo a la conexión. Mientras el bit SW_ENABLE está activado, la secuencia de salida se ejecuta de modo continuo.

Diagrama de impulsos

La figura siguiente muestra una secuencia de salida en el modo de operación "Salida de frecuencia".



Cancelación de la secuencia de salida

Borrando la habilitación software (SW_ENABLE = 0) durante el retardo a la conexión o durante la salida de frecuencia se cancela la secuencia de salida y el bit STS_ENABLE y la salida digital DO se borran.

A continuación es necesario reiniciar la secuencia de salida.

Tabla de verdad

La tabla siguiente muestra ante qué señales y de qué forma reacciona el 2PULSE.

Habilitación software SW_ENABLE	Habilitación hardware (entrada digital DI)	Salida digital DO	STS_ENABLE	Secuencia de salida
1	0 → 1	0 si el retardo a la conexión > 0 1 si el retardo a la conexión = 0	0 → 1	iniciar
0 → 1	no utilizado	0 si el retardo a la conexión > 0 1 si el retardo a la conexión = 0	0 → 1	iniciar
0	cualquier estado	0	0	cancelar

Habilitación software SW_ENABLE	Habilitación hardware (entrada digital DI)	Salida digital DO	STS_ENABLE	Secuencia de salida
1	cualquier estado	el estado previo se mantiene		-
0 → 1	0	0	0	-

Ajustar tiempos mediante la base de tiempo

Mediante la base de tiempo (Página 10423) parametrizable se seleccionan la resolución y el rango del retardo a la conexión.

Base de tiempo = 0,1 ms:	Se pueden fijar tiempos desde 0,2 ms a 6,5535 s con una resolución de 0,1 ms.
Base de tiempo = 1 ms:	Se pueden fijar tiempos desde 1 ms a 65,535 s con una resolución de 1 ms.

Ajustar y modificar el valor de salida

Con el parámetro "Formato de salida (Página 10423)" se selecciona el rango del valor de salida.

- Si el valor de salida está comprendido entre 1 Hz y 5000 Hz, seleccione el formato de salida "1 Hz".
- Si el valor de salida es un valor analógico SIMATIC S7 (entre 0 y 27648), seleccione el formato de salida "Salida analógica S7". Con "Salida analógica S7" se puede alcanzar una mayor precisión en la especificación de consigna.
  El 2PULSE calcula la frecuencia conforme a la fórmula siguiente:
  Frecuencia = (consigna / 27648) × 5 kHz

El valor de salida se especifica directamente con el programa de control. El nuevo valor de frecuencia se emite con el siguiente flanco ascendente de la salida.

Precisión de la frecuencia de salida

La frecuencia de salida parametrizada se emite en la salida digital DO con la siguiente precisión:

- 0,01 Hz con frecuencia de salida de 0,1 Hz a 800 Hz
- 0,1 Hz con frecuencia de salida de > 800 Hz a 2000 Hz
- 0,5 Hz con frecuencia de salida de > 2000 Hz a 5000 Hz

Ajustar el retardo a la conexión en caso de utilizar la interfaz de control corta

El retardo a la conexión (Página 10423) se define en los parámetros como valor numérico entre 0 y 65535.

Retardo a la conexión parametrizado = base de tiempo × valor numérico ajustado

Si utiliza la interfaz de control corta, con el factor para el retardo a la conexión se puede adaptar el tiempo parametrizado en el programa de control. El factor se ajusta entre 0 y 255 con una ponderación de 0,1:

Retardo a la conexión = factor × 0,1 × retardo a la conexión parametrizado

Modo isócrono

En el modo de operación "Salida de frecuencia" el modo isócrono no tiene ningún efecto en la funcionalidad.

Parámetros del modo de operación "Salida de frecuencia"

Parámetros del modo de operación "Salida de frecuencia"

La tabla siguiente muestra los parámetros del modo de operación "Salida de frecuencia".

Parámetro	Significado	Rango	Ajuste predeterminado
Modo de operación (Página 10422)	Ajuste el modo de operación "Salida de frecuencia".	• Salida de impulsos • Modulación de ancho de impulsos • Tren de impulsos • Retardo a la conexión/ desconexión • Salida de frecuencia	Modulación de ancho de impulsos
Base de tiempo (Página 10423)	Mediante la base de tiempo se seleccionan la resolución y el rango del retardo a la conexión.	• 0,1 ms • 1 ms	0,1 ms
Formato de salida (Página 10423)	Dependiendo de la resolución requerida para la salida de frecuencia, elija entre 1 Hz o Salida analógica S7.	• 1 Hz • Salida analógica S7	1 Hz
Función DI (Página 10423)	La entrada digital DI se puede utilizar como entrada o como habilitación HW.	• Entrada • Habilitar HW	Entrada
Retardo a la conexión (Página 10423) [1]	Tiempo que transcurre desde el comienzo de la secuencia de salida hasta la salida de la secuencia de impulsos. El retardo a la conexión se puede modificar en el programa de control con el "Factor de retardo a la conexión".	de 0 a 65535	0

[1] Sólo si se utiliza la interfaz de control (Página 10425) corta.
Si se utiliza la interfaz de control larga, ajuste este valor directamente en la interfaz de control.

Señales de control y realimentación del modo de operación "Salida de frecuencia"

Señales de control y realimentación del modo de operación "Salida de frecuencia"

La tabla siguiente muestra las señales de control y realimentación del modo de operación "Salida de frecuencia".

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
Señales de control en caso de utilizar la interfaz de control corta (8 bytes)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación hardware	Byte 2: Bit 0	Byte 6: Bit 0
Valor de salida	El valor que se aplica a la salida digital DO como frecuencia.	Dependiendo del formato de salida: • 1 Hz: 0 ... 5000 • Salida analógica S7: 0 ... 27648 Si se introduce un valor de salida > 5000 ó 27648, el 2PULSE lo limitará a 5000 ó 27648, respectivamente.	Palabra 0	Palabra 4
Factor de retardo a la conexión	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor × 0,1 × retardo a la conexión parametrizado	de 0 a 255 Si resulta un retardo a la conexión < 0,2 ms o con un factor = 0, el retardo a la conexión efectivo = 0. Si resulta un retardo a la conexión > 65,535 s, el retardo a la conexión se limita a 65,535 s.	Byte 3	Byte 7
Señales de control en caso de utilizar la interfaz de control larga (12 bytes)				
Habilitación software (SW_ENABLE)	Iniciar y cancelar la secuencia de salida.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado 0 → 1 = inicio de la secuencia de salida; dado el caso, depende de la habilitación hardware	Byte 4: Bit 0	Byte 10: Bit 0
Valor de salida	Valor que se emite en la salida digital DO como frecuencia.	Dependiendo del formato de salida: • 1 Hz: 0 ... 5000 • Salida analógica S7: 0 ... 27648 Si se introduce un valor de salida > 5000 ó 27648, el 2PULSE lo limitará a 5000 ó 27648 respectivamente.	Palabra 0	Palabra 6
Retardo a la conexión	El retardo a la conexión se puede modificar antes de que comience la secuencia de salida.	de 0 a 65535 Retardo a la conexión = base de tiempo x valor numérico fijado	Palabra 2	Palabra 8
Señales de respuesta				

Señales de control y realimentación	Significado	Rango	Dirección del canal 0	Dirección del canal 1
STS_ENABLE	Indica una secuencia de salida en curso.	0 = salida de impulsos bloqueada 1 = salida de impulsos en curso	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Indica el nivel de señal en la salida digital DO. Observe la velocidad de actualización.	0 = señal 0 en salida digital DO 1 = señal 1 en salida digital DO	Byte 0: Bit 1	Byte 4: Bit 1
STS_DI	Indica el nivel de señal en la entrada digital DI.	0 = señal 0 en la entrada digital DI 1 = señal 1 en la entrada digital DI	Byte 0: Bit 2	Byte 4: Bit 2
ACK_SW_ENABLE	Indica el estado de SW_ENABLE.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 0: Bit 3	Byte 4: Bit 3
SEQ_CNT	no tiene función en este modo de operación.	–	Byte 1: Bit 0 … 3	Byte 5: Bit 0 … 3
Reservado	–	0	Palabra 2	Palabra 6

Señales de entrada y de salida del modo de operación "Salida de frecuencia"

Señales de entrada y de salida del modo de operación "Salida de frecuencia"

La tabla siguiente muestra las señales de entrada y de salida del modo de operación "Salida de frecuencia".

Señal de entrada y salida	Significado	Rango	Terminal del canal 0	Terminal del canal 1
Señal de entrada				
Habilitación HW	La habilitación HW se puede seleccionar con el parámetro "Función DI". La señal de la entrada digital DI se evaluará con el 2PULSE al inicio de la secuencia de salida.	0 = habilitación HW borrada 1 = habilitación HW obtenida 0 → 1 = inicio de la secuencia de salida; depende de la habilitación SW	1	5
Señal de salida				
Frecuencia en la salida digital DO	La frecuencia se emite en la salida digital DO.	0 = no hay impulso 1 = impulso	4	8

Contador de secuencias (a partir de 6ES7138-4DD01-0AB0)

El 2PULSE dispone de un contador de secuencias para cada canal que permite contar las secuencias de salida finalizadas.
Es posible vigilar el fin de una secuencia de salida en la interfaz de respuesta utilizando el contador de secuencias SEC_CNT.

El contador tiene una anchura de 4 bits. Tras un desbordamiento, el contador salta a 0 nuevamente.

Funcionamiento del contador de secuencias

Dependiendo del modo de operación, el contador de secuencias tiene las siguientes funciones:

- Salida de impulsos (Página 10375) y tren de impulsos (Página 10390)
  - Sin habilitación hardware, el contador se pone a 1 una vez finalizada la secuencia de salida.
  - Con habilitación hardware, el contador se incrementa cada vez que finaliza una secuencia de salida.
- Retardo a la conexión/desconexión (Página 10397)
  El contador se incrementa en la DO con cada flanco (ascendente y descendente).
- Modulación de ancho de impulsos y salida de frecuencia
  El contador no tiene ninguna función.

Borrando la habilitación software (SW_ENABLE = 0) el contador se pone a 0 nuevamente.

Posibilidades de uso

El contador de secuencias se puede utilizar especialmente para:

- Medir (contar) impulsos demasiado cortos.
- Contar las secuencias de salida en combinación con la habilitación hardware.

Función: Medición de intensidad (a partir de 6ES7138-4DD01-0AB0)

La medición de intensidad se puede utilizar en toda una serie de aplicaciones para llevar a cabo funciones de regulación y diagnóstico. En la interfaz de respuesta puede leerse el valor medido de intensidad.

Funcionamiento

La medición de intensidad sólo tiene lugar durante la emisión de impulsos en los modos de operación "Modulación de ancho de impulsos (Página 10380)" y "Tren de impulsos (Página 10390)". Si no hay salida de impulsos (con STS_ENABLE = 0) se reporta 0 como valor medido. Con STS_ENABLE = 1 el valor medido es válido.

En los modos de operación "Salida de impulsos", "Retardo a la conexión/desconexión" y "Salida de frecuencia" siempre se reporta 0 como valor medido de intensidad.

El valor medido de intensidad que se devuelve es un valor medio formado por valores medidos que se registran dentro de la duración de un período. En períodos < 5 ms, entre dos puntos de medición individuales existe una separación temporal de 40 µs como máximo. En períodos aún más largos se registran siempre 128 puntos de medición por período. De esta manera se crea una separación temporal más larga entre los puntos de medición.

En el modo isócrono, en cada instante T_i se suministra un nuevo valor medido de intensidad a través de la interfaz de respuesta. En el modo no isócrono se suministra un nuevo valor medido de intensidad aprox. cada 500 µs.

Nota

Para que la medición de intensidad funcione correctamente no se debe integrar ningún diodo de libre circulación en la carga.

Rango de medida y valor medido

El rango de medida está fijado en una corriente nominal de 2 A.

El valor medido de intensidad se suministra en la interfaz de respuesta (Página 10428) como valor analógico SIMATIC S7.

- Una intensidad de 2 A corresponde al valor analógico SIMATIC S7 27648.
- En caso de una conexión en paralelo, una intensidad de 4 A corresponde al valor analógico SIMATIC S7 27648.

Las mediciones de intensidad se pueden realizar hasta el valor analógico SIMATIC S7 32767. Esto corresponde a una intensidad de 2,4 A. Intensidades por encima de 2 A sólo pueden presentarse momentáneamente.

Precisión

La precisión de la medición de intensidad asciende al 3 % del rango de medida.

Nota

Esta precisión en la medición de intensidad sólo se puede alcanzar si durante la medición no se cambia el período.

Función: Control directo de la salida digital DO

Definición

Para probar el actuador conectado, es posible controlar directamente la salida digital DO del 2PULSE. Para ello hay que seleccionar la función por medio del programa de control (Página 10425) con el bit de control MANUAL_DO activado y el bit de control SW_ENABLE borrado. La función

- sirve de apoyo para poner en marcha una instalación
- permite aprovechar un canal no utilizado del 2PULSE como DO

Una vez seleccionada la función, el 2PULSE borra los bits de respuesta (Página 10428)STS_ENABLE y ERR_PULS y se cancela la secuencia de salida en curso. La función es prioritaria frente al modo de operación seleccionado actualmente.

El estado de la salida digital DO se especifica con el bit de control (Página 10425)SET_DO.

Borrando el bit de control MANUAL_DO, se deselecciona la función "Control directo de la salida digital DO". Con ello se borra la salida digital DO. A continuación se puede reiniciar una secuencia de salida con el flanco ascendente de SW_ENABLE.

Señales de control y realimentación/señal de salida

Señales	Significado	Rango de valores	En caso de utilizar la interfaz de control corta (8 bytes)		En caso de utilizar la interfaz de control larga (12 bytes)	
			Dirección del canal 0	Dirección del canal 1	Dirección del canal 0	Dirección del canal 1
Señales de control						
SW_ENABLE	Para poder seleccionar la función tiene que haberse borrado el bit de control.	0 = SW_ENABLE borrado 1 = SW_ENABLE activado	Byte 2: Bit 0	Byte 6: Bit 0	Byte 4: Bit 0	Byte 10: Bit 0
MANUAL_DO	Con el bit de control se selecciona y deselecciona la función.	0 = control directo de la salida DO no seleccionado. 1 = control directo de la salida DO seleccionado.	Byte 2: Bit 1	Byte 6: Bit 1	Byte 4: Bit 1	Byte 10: Bit 1
SET_DO	Con el bit de control se especifica el estado de la salida digital DO.	0 = señal 0 en la salida digital DO 1 = señal 1 en la salida digital DO	Byte 2: Bit 2	Byte 6: Bit 2	Byte 4: Bit 2	Byte 10: Bit 2
Señales de respuesta						
STS_ENABLE	Se borra tras seleccionarse la función.	0 = salida de impulsos bloqueada 1 = salida de impulsos en curso	Byte 0: Bit 0	Byte 4: Bit 0	Byte 0: Bit 0	Byte 4: Bit 0
STS_DO	Indica el nivel de señal en la salida digital DO. Observe la velocidad de actualización.	0 = señal 0 en la salida digital DO 1 = señal 1 en la salida digital DO	Byte 0: Bit 1	Byte 4: Bit 1	Byte 0: Bit 1	Byte 4: Bit 1
STS_ DI	Indica el nivel de señal en la entrada digital DI.	0 = señal 0 en la entrada digital DI 1 = señal 1 en la entrada digital DI	Byte 0: Bit 2	Byte 4: Bit 2	Byte 0: Bit 2	Byte 4: Bit 2
Señal de salida						
Salida digital DO	El estado preajustado con el bit de control SET_DO se emite por la salida digital DO.	0 = no hay señal 1 = señal	4	8	4	8

Función: Detección de fallos/Diagnóstico

Error de parametrización ERR_PARA

Si el 2PULSE no puede identificar los parámetros como propios, genera un error de parametrización (Página 10428). Entonces los dos canales no están parametrizados.

El slot configurado del 2PULSE debe coincidir con la configuración real.

Error de salida de impulsos ERR_PULS

El 2PULSE detecta un error de salida de impulsos específico de un canal en los modos de operación "Salida de impulsos (Página 10375)", "Retardo a la conexión/desconexión (Página 10397)" y "Tren de impulsos (Página 10390)".

Las causas y reacciones se encuentran en la descripción del modo de operación correspondiente y en "Interfaz de respuesta del 2PULSE (Página 10428)".

El error de salida de impulsos detectado se muestra para el canal afectado con el bit de respuesta (Página 10428)ERR_PULS.

Cortocircuito en la alimentación del sensor ERR_24V

El 2PULSE detecta un cortocircuito de la alimentación del sensor que pone a disposición en los terminales 2 y 6.

El cortocircuito detectado se indica para ambos canales con el bit de respuesta (Página 10428)ERR_24V.

Una caída de la tensión de alimentación de 24 V también puede ocasionar una señalización de cortocircuito.

Cortocircuito en la salida digital ERR_DO

El 2PULSE detecta un cortocircuito en la salida digital del canal.

El cortocircuito detectado se muestra para el canal afectado con el bit de respuesta (Página 10428)ERR_DO.

Nota

En caso de cortocircuito, la salida digital está en capacidad de suministrar momentáneamente una intensidad mucho más alta que el valor asignado.

Aviso de diagnóstico

En caso de "cortocircuito en la alimentación del encóder" o "cortocircuito de la salida digital", el 2PULSE genera un aviso de diagnóstico para la CPU/el maestro conectado. Para ello se debe habilitar el parámetro "Diagnóstico de grupo (Página 10421)".

Nota

Mediante avisos de diagnóstico se notifica que en un canal o en ambos canales del 2PULSE ha ocurrido un error. El diagnóstico se puede evaluar con STEP 7.

El registro de diagnóstico DS1 sólo contiene la información correspondiente al canal 0. No obstante, también se genera una alarma de diagnóstico si ocurren errores en el canal 1. Las señales de respuesta proporcionan información detallada sobre los errores ocurridos en ambos canales. Evalúe la interfaz de respuesta (Página 10428) en programas de usuario nuevos (ver abajo).

Parámetros

Parámetros	Significado	Rango	Por defecto
Diagnóstico colectivo (Página 10421)	Estando habilitado el diagnóstico colectivo, el 2PULSE genera un aviso de diagnóstico para la CPU/el maestro.	inhibir/habilitar	inhibir
Diagnóstico DO (Página 10422)	El 2PULSE detecta un cortocircuito en las salidas digitales si el "Diagnóstico DO" = ON.	Off/On	ON

Señales de respuesta

Señales de respuesta	Significado	Rango	Dirección del canal 0	Dirección del canal 1
ERR_PARA	Indica un error de parametrización.	0 = no hay error de parametrización 1 = error de parametrización	Byte 0: Bit 5	Byte 4: Bit 5
ERR_PULS	Indica un error de salida de impulsos.	0 = no hay error de salida de impulsos 1 = error de salida de impulsos	Byte 0: Bit 4	Byte 4: Bit 4
ERR_24V	Indica un cortocircuito de la alimentación del sensor.	0 = no hay cortocircuito de la alimentación del sensor 1 = cortocircuito de la alimentación del sensor	Byte 0: Bit 7	Byte 4: Bit 7
ERR_DO	Indica un cortocircuito de la salida digital DO.	0 = no hay cortocircuito de la salida digital 1 = cortocircuito de la salida digital	Byte 0: Bit 6	Byte 4: Bit 6

Conmutación paralela de ambos canales (a partir de 6ES7138-4DD01-0AB0)

Definición

Para alcanzar una intensidad de salida más alta, se pueden conectar en paralelo ambos canales del 2PULSE. En ese caso se obtiene una intensidad asignada de 4 A. Esto permite operar directamente en el 2PULSE actuadores con un consumo de corriente más elevado.

Parametrización y manejo

La conexión en paralelo de los dos canales se activa al realizar la parametrización con las casillas de verificación "Conexión en paralelo canal 0 y 1 (Página 10422)".

Tras efectuar esta activación, el 2PULSE funciona como un módulo monocanal. Cualquier parametrización o manejo que se lleve a cabo mediante la interfaz de control (Página 10425) debe efectuarse a través del canal 0. El canal 1 no se puede parametrizar ni manejar.

Asimismo, las señales de respuesta (Página 10428) sólo se suministran para el canal 0.

Medición de intensidad

En la medición de intensidad se vuelve a calcular un valor medio con base en los valores medidos en ambos canales. De esta manera resulta un rango de medida de hasta 4 A.

El valor medido de intensidad se suministra en la interfaz de respuesta como valor analógico SIMATIC S7. En este caso, una intensidad de 4 A corresponde al valor analógico SIMATIC S7 27648.

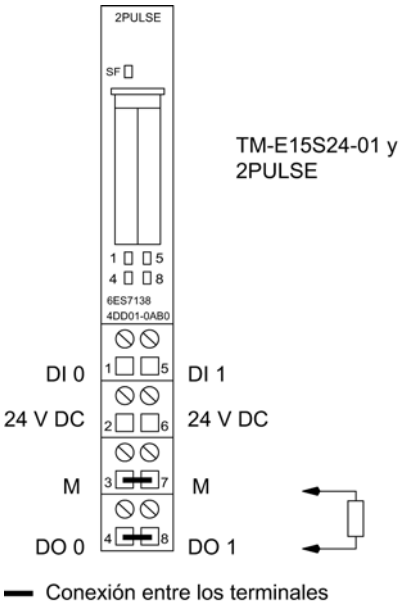
En la conexión en paralelo, la precisión de la medición de intensidad asciende también al 3 % del rango de medida.

Asignación de terminales para la conexión en paralelo

Para conectar en paralelo ambos canales del 2PULSE deben interconectarse los siguientes terminales en el módulo de pines:

- Terminal 3 y terminal 7
- Terminal 4 y terminal 8

La carga (p. ej. un actuador) se conecta luego directamente a los terminales 3 y 4 o bien a los terminales 7 y 8. La figura siguiente muestra un esquema de conexiones.



Reacción a STOP de la CPU/del maestro

Definición

Puede parametrizar la reacción (Página 10421) del 2PULSE en caso de fallo del controlador principal conjuntamente para ambos canales.

Reacción a STOP de la CPU/del maestro	Reacción específica del canal y estado del 2PULSE
Desconectar DO	borrar la salida digital DO, borrar STS_ENABLE y cancelar la actual secuencia de salida.
Continuar el modo de operación	la salida digital DO permanece inalterada, STS_ENABLE permanece inalterado y la actual secuencia de salida sigue funcionando.
Aplicar valor sustitutivo DO	emitir el valor sustitutivo parametrizado de la salida digital DO específico del canal, borrar STS_ENABLE y cancelar la actual secuencia de salida.
Mantener el último valor DO	la salida digital DO permanece inalterada, borrar STS_ENABLE y cancelar la actual secuencia de salida.

Arranque

Para iniciar una nueva secuencia de salida después de un STOP de la CPU/del maestro y con el bit ACK_SW_ENABLE activado, hay que borrar primero el bit SW_ENABLE. Repita este borrado hasta que el bit ACK_SW_ENABLE también se borre.

Si el modo de operación seleccionado debe continuar al cambiar de STOP de la CPU/del maestro a RUN (arranque), la CPU/el maestro no debe borrar las salidas. **Solución posible:** active el bit de control "Habilitación software" (SW_ENABLE = 1) en aquella parte del programa de usuario que se procesa durante el arranque.

Parametrización modificada

El estado adoptado por el 2PULSE en STOP de la CPU/del maestro se mantiene incluso en caso de parametrización o configuración de la estación ET 200S. Esto ocurre, por ejemplo, al conectarse la alimentación de la CPU/del maestro o del IM 151 o al retornar la transmisión DP.

Pero en el estado "Continuar el modo de operación" y tras cargar una parametrización o configuración modificada de la estación ET 200S en la CPU/el maestro, el 2PULSE cancela el proceso. Como resultado, el 2PULSE hace lo siguiente:

- borra la salida digital DO
- borra STS_ENABLE y
- cancela la actual secuencia de salida.

Configuración y parametrización del 2PULSE

Parámetros

Seleccionar modo de operación

Modo de operación

Seleccione el modo de operación del 2PULSE en función de sus necesidades:

- 2 Pulse: 8 bytes de datos de salida: compatible con el módulo de versión anterior
  La interfaz de control ocupa menos direcciones en la memoria imagen de proceso.
- 2 Pulse interfaz de control ampl.: 12 bytes de datos de salida – posibilidades de control mejoradas
  Las consignas se pueden introducir más fácilmente.

Activar diagnóstico de grupo

Diagnóstico de grupo

El diagnóstico de grupo se activa en la casilla de verificación correspondiente.

Si está habilitado el diagnóstico de grupo, el 2PULSE genera un aviso de diagnóstico para la CPU o el maestro.

Seleccionar la reacción a STOP de la CPU/del maestro

Reacción a STOP de la CPU/del maestro

En la lista desplegable, seleccione el comportamiento deseado en caso de STOP de la CPU o del maestro.

- Desconectar DO
- Modo de operación Continuar
- Aplicar valor sustitutivo DO
- Mantener el último valor DO

Encontrará más información sobre las reacciones y el estado del 2PULSE en ""Reacción a STOP de la CPU o del maestro (Página 10417)".

Activar conexión en paralelo del canal 0 y 1

Conexión en paralelo canal 0 y 1 (a partir de 6ES7138-4DD01-0AB0)

Si necesita una intensidad nominal mayor de 2 A en las salidas digitales, puede conmutar en paralelo los dos canales del 2PULSE. Para ello active la casilla de verificación "Conexión en paralelo del canal 0 y 1".

Encontrará más información en "Conexión en paralelo de los dos canales (a partir de 6ES7138-4DD01-0AB0) (Página 10415)".

Canal n

Activar diagnóstico DO

Diagnóstico DO

Si activa la casilla de verificación Diagnóstico DO, el 2PULSE detecta un cortocircuito en las salidas digitales.

Seleccionar valor sustitutivo de DO

Valor sustitutivo DO

En la lista desplegable seleccione el valor sustitutivo deseado para la salida digital:

- 0
- 1

El parámetro "Valor sustitutivo DO" sólo puede tener un valor distinto de cero si el parámetro "Reacción a STOP de la CPU/maestro (Página 10418)" está ajustado a "Aplicar valor sustitutivo DO".

Seleccionar el modo de operación

Modo de operación

En la lista desplegable seleccione el modo de operación deseado:

- Salida de impulsos (Página 10375)
- Modulación de ancho de impulsos (PWM) (Página 10380)
- Tren de impulsos (Página 10390)
- Retardo a la conexión/desconexión (Página 10397)
- Salida de frecuencia (Página 10405) (a partir de 6ES7138-4DD01-0AB0)

Seleccionar formato de salida

Formato de salida

El formato de salida es relevante para los modos de operación "Modulación de ancho de impulsos (PWM)" y "Salida de frecuencia" (a partir de 6ES7138-4DD01-0AB0).

- En el modo de operación "Modulación de ancho de impulsos (PWM) (Página 10380)" es posible seleccionar el formato de salida en la lista desplegable:
  - Por mil
  - Salida analógica S7
- En el modo de operación "Salida de frecuencia (Página 10405)" es posible seleccionar en la lista desplegable el formato de salida:
  - 1 Hz
  - Salida analógica S7

Seleccionar base de tiempo

Base de tiempo

Seleccione a través de la base de tiempo la resolución y el margen de valores de todos los tiempos predefinidos:

- 0,1 ms
  Se pueden fijar tiempos desde 0,2 ms a 6,5535 s con una resolución de 0,1 ms.
- 1 ms
  Se pueden fijar tiempos desde 1 ms a 65,535 s con una resolución de 1 ms.

Seleccionar función DI

Función DI

La entrada digital DI se puede utilizar como entrada o como habilitación hardware.

En la lista desplegable seleccione la función deseada para la entrada digital:

- Entrada
- Habilitación HW

Definir el retardo a la conexión

Retardo a la conexión

El retardo a la conexión es el tiempo que transcurre desde el inicio de la secuencia de salida hasta la salida del impulso.

Registre el retardo a la conexión deseado en el campo de entrada.

Rango de valores admisible: 0 … 65535

Definir duración mínima del impulso

Duración mínima del impulso

La duración mínima del impulso es relevante para los modos de operación "Modulación de ancho de impulsos" (PWM)" (Página 10380) y "Salida de frecuencia (Página 10405)" (a partir de 6ES7138-4DD01-0AB0).

La duración mínima de impulso y la pausa mínima entre impulsos se parametrizan con el parámetro "Duración mínima de impulso"; ambos tienen siempre el mismo valor.

Por ejemplo, introduzca aquí el tiempo de respuesta del elemento final de control conectado a la salida digital DO.

Rango de valores admisible: 0 … 65535

Definir duración del impulso

Duración de impulso

La duración de impulso es relevante para el modo de operación "Tren de impulsos (Página 10390)".

Por ejemplo, introduzca aquí el tiempo de respuesta del actuador conectado a la salida digital DO.

Rango de valores admisible:

- con base de tiempo 1 ms: de 1 a 65535
- con base de tiempo 0,1 ms: de 2 a 65535

Definir duración del período

Período

La duración del período es relevante para los modos de operación "Modulación de ancho de impulsos (PWM) (Página 10380)" y "Tren de impulsos (Página 10390)".

Al ajustar la duración del período, observe la duración mínima del impulso ajustada / la duración del impulso y el tiempo de respuesta del elemento final de control conectado a la salida digital DO. La duración del período debería ser siempre un múltiplo del tiempo de respuesta del elemento final de control conectado a la salida digital DO.

La duración del período se define como valor numérico en función de la base de tiempo:

- con base de tiempo 1 ms: de 1 a 65535
- con base de tiempo 0,1 ms: de 2 a 65535

Duración de período parametrizada = Base de tiempo x valor numérico predefinido

Interfaz de control y realimentación del 2PULSE

Interfaz de control del 2PULSE

Nota

Para el 2PULSE, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 a 3
- Bytes 4 a 7

Asignación de la interfaz de control corta (8 bytes)

La siguiente tabla muestra la asignación de la interfaz de control corta (salidas) del 2PULSE.

Dirección		Asignación
Canal 0	Canal 1	
Palabra 0	Palabra 4	según el modo de operación • Salida de impulsos: duración del impulso • Modulación de ancho de impulsos: valor de salida • Tren de impulsos: número de impulsos • Retardo a la conexión/desconexión: retardo a la desconexión • Salida de frecuencia: Frecuencia de salida (a partir de 6ES7138-4DD01-0AB0)
Byte 2	Byte 6	Bit 7: reserva = 0 Bit 6: reserva = 0 Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: reserva = 0 Bit 2: SET_DO Bit 1: MANUAL_DO Bit 0: SW_ENABLE
Byte 3	Byte 7	según el modo de operación • Salida de impulsos: factor de retardo a la conexión • Modulación de ancho de impulsos: factor de período • Tren de impulsos: factor de período • Retardo a la conexión/desconexión: factor de retardo a la conexión • Salida de frecuencia: Factor de retardo a la conexión (a partir de 6ES7138-4DD01-0AB0)

Asignación de la interfaz de control larga (12 bytes) (a partir de 6ES7138-4DD01-0AB0)

La siguiente tabla muestra la asignación de la interfaz de control larga (salidas) del 2PULSE.

Dirección		Asignación
Canal 0	Canal 1	
Palabra 0	Palabra 6	según el modo de operación • Salida de impulsos: duración del impulso • Modulación de ancho de impulsos: valor de salida • Tren de impulsos: número de impulsos • Retardo a la conexión/desconexión: retardo a la desconexión • Salida de frecuencia: frecuencia de salida
Palabra 2	Palabra 8	según el modo de operación • Salida de impulsos: Retardo a la conexión • Modulación de ancho de impulsos: Período • Tren de impulsos: Período • Retardo a la conexión/desconexión: Retardo a la conexión • Salida de frecuencia: Retardo a la conexión
Byte 4	Byte 10	Bit 7: reserva = 0 Bit 6: reserva = 0 Bit 5: reserva = 0 Bit 4: reserva = 0 Bit 3: reserva = 0 Bit 2: SET_DO Bit 1: MANUAL_DO Bit 0: SW_ENABLE
Byte 5	Byte 11	según el modo de operación • Salida de impulsos: no utilizado • Modulación de ancho de impulsos: factor de retardo a la conexión • Tren de impulsos: relación impulso-pausa • Retardo a la conexión/desconexión: no utilizado • Salida de frecuencia: no utilizado

Significado de las señales de control

Señal de control	Significado
Modo de operación "Salida de impulsos":	
• Duración del impulso [1) 2)]	Tiempo en que la salida digital DO está activada una vez transcurrido el retardo a la conexión.
• Factor de retardo a la conexión [1)]	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor × 0,1 × retardo a la conexión parametrizado
• Retardo a la conexión [2)]	Tiempo que transcurre desde el comienzo de la secuencia de salida hasta la salida de la secuencia de impulsos.
Modo de operación "Modulación de ancho de impulsos":	

Señal de control	Significado
• Valor de salida [1] [2]	Valor emitido con modulación de ancho de impulsos en la salida digital DO una vez transcurrido el retardo a la conexión.
• Factor de período [1]	El período parametrizado se puede modificar: Período = factor × 0,1 × período parametrizado
• Período [2]	Al ajustar el período observe la duración mínima del impulso ajustada y el tiempo de respuesta del elemento final de control conectado a la salida digital DO.
• Factor de retardo a la conexión [2]	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor × 0,1 × retardo a la conexión parametrizado
Modo de operación "Tren de impulsos":	
• Número de impulsos [1] [2]	Número de impulsos que se emiten en la salida digital DO una vez transcurrido el retardo a la conexión.
• Factor de período [1]	El período parametrizado se puede modificar antes de que comience la secuencia de salida: Período = factor × 0,1 × período parametrizado
• Período [2]	Al ajustar el período observe la duración mínima del impulso ajustada y el tiempo de respuesta del elemento final de control conectado a la salida digital DO.
• Relación impulso-pausa [2]	Es posible ajustar la relación entre la duración del impulso y la pausa entre impulsos: Relación impulso-pausa = (consigna / 255) × período
Modo de operación "Retardo a la conexión/desconexión":	
• Retardo a la desconexión [1] [2]	Tiempo que transcurre entre el flanco descendente de la entrada digital DI y su salida a través de la salida digital DO.
• Factor de retardo a la conexión [1]	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor × 0,1 × retardo a la conexión parametrizado
• Retardo a la conexión [2]	Tiempo que transcurre entre el flanco ascendente de la entrada digital DI y su salida a través de la salida digital DO.
Modo de operación "Salida de frecuencia" (a partir de 6ES7138-4DD01-0AB0)**:**	
• Frecuencia de salida [1] [2]	Frecuencia que se emite en la salida digital DO una vez transcurrido el retardo a la conexión.
• Factor de retardo a la conexión [1]	El retardo a la conexión parametrizado se puede modificar antes de que comience la secuencia de salida: Retardo a la conexión = factor × 0,1 × retardo a la conexión parametrizado
• Retardo a la conexión [2]	Tiempo que transcurre desde el comienzo de la secuencia de salida hasta la salida de la frecuencia.
Control directo de la salida digital:	
• MANUAL_DO	El bit de control permite seleccionar y deseleccionar la función "Control directo de la salida digital DO".
• SET_DO	El bit de control permite ajustar el estado de la salida digital DO.
Habilitación de software	
• SW_ENABLE	La habilitación software debe activarse siempre en el programa de control. Si no se utiliza una habilitación HW, la secuencia de salida será iniciada por el flanco ascendente de la habilitación de software. Si se desactiva la habilitación de software, la secuencia de salida actual se cancela.

[1] en caso de utilizar la interfaz de control corta

[2] en caso de utilizar la interfaz de control larga

Interfaz de realimentación del 2PULSE

Nota

Para el 2PULSE, los datos siguientes de la interfaz de realimentación son datos relacionados, es decir, coherentes:

- Bytes 0 a 3
- Bytes 4 a 7

Asignación de la Interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 2PULSE.

Dirección		Asignación
Canal 0	Canal 1	
Byte 0	Byte 4	Bit 7: ERR_24V Bit 6: ERR_DO Bit 5: ERR_PARA Bit 4: ERR_PULS Bit 3: ACK_SW_ENABLE Bit 2: STS_DI Bit 1: STS_DO Bit 0: STS_ENABLE
Byte 1	Byte 5	Bit 0 … Bit 3: Contador de secuencias SEQ_CNT (a partir de 6ES7138-4DD01-0AB0) Bit 4 … Bit 7: reserva = 0
Palabra 2	Palabra 6	Valor de medición de intensidad (a partir de 6ES7138-4DD01-0AB0)

Nota

Dependiendo del módulo de interfaz utilizado IM 151 y de la configuración, en el caso de que se produzca un fallo en el módulo, se devuelve 16#7FFF7FFF o 16#00000000 como valor de realimentación (valor sustitutivo).

Significado de las señales de realimentación

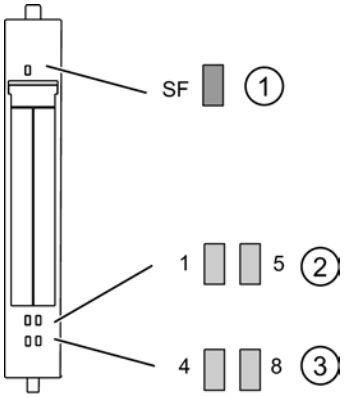
Señal de realimentación	Significado
ACK_SW_ENABLE	Indica el estado pendiente de la habilitación software en el módulo 2PULSE.
ERR_24V	Indica un cortocircuito o bien una caída de la alimentación del sensor.
ERR_DO	Indica un cortocircuito en la salida digital.

Señal de realimentación	Significado
ERR_PARA	Indica un error de parametrización.
ERR_PULS	Modo de operación "Salida de impulsos": Indica un error de salida de impulsos. Si una vez transcurrido el retardo a la conexión se acorta la duración de impulso de manera que el tiempo sea más corto que el tiempo ya transcurrido, el 2PULSE lo detectará. Cuando comience la siguiente secuencia de salida el 2PULSE borrará el bit de realimentación ERR_PULS. Modo de operación "Tren de impulsos": Indica un error de salida de impulsos. Si una vez transcurrido el retardo a la conexión se reduce el número de impulsos y si el número de impulsos reducido ya se ha emitido, el 2PULSE lo detectará. Cuando comience la siguiente secuencia de salida, el 2PULSE borrará el bit de realimentación ERR_PULS. Modo de operación "Retardo a la conexión/desconexión": Indica un error en la salida de impulsos cuando la duración del impulso o la pausa entre impulsos es demasiado corta. El 2PULSE borra el bit de realimentación ERR_PULS con el siguiente flanco ascendente de la habilitación software o con el siguiente flanco en la entrada digital DI.
STS_ DI	Indica el nivel de señal en la entrada digital DI.
STS_DO	Indica el nivel de señal en la salida digital DO.
STS_ENABLE	Modo de operación "Salida de impulsos": Se activa al comenzar la secuencia de salida hasta que termine la duración de impulsos. Si se desactiva la habilitación software (SW_ENABLE) o si el 2PULSE detecta un error de salida de impulsos ERR_PULS, entonces se borra STS_ENABLE. Modo de operación "Modulación de ancho de impulsos" (PWM): Se activa al comenzar la secuencia de salida. Si se borra la habilitación software (SW_ENABLE), STS_ENABLE también se borra. Modo de operación "Tren de impulsos": Se activa al comenzar la secuencia de salida hasta la salida del último impulso. Si se desactiva la habilitación software (SW_ENABLE) o si el 2PULSE detecta un error de salida de impulsos ERR_PULS, entonces se borra STS_ENABLE. Modo de operación "Retardo a la conexión/desconexión": Señaliza que el 2PULSE ha detectado la habilitación software (SW_ENABLE) e indica el estado de la misma. Modo de operación "Salida de frecuencia" (a partir de 6ES7138-4DD01-0AB0): se activa al comenzar la secuencia de salida. Si se borra la habilitación software (SW_ENABLE), STS_ENABLE también se borra.
SEQ_CNT (a partir de 6ES7138-4DD01-0AB0)	Cuenta las secuencias de salida finalizadas en los modos de operación "Salida de impulsos", "Tren de impulsos" y "Retardo a la conexión/desconexión". Sin función en los modos "Modulación de ancho de impulsos" y "Salida de frecuencia".
Valor de medición de intensidad (a partir de 6ES7138-4DD01-0AB0)	Intensidad en la DO como valor analógico SIMATIC S7

Diagnóstico del 2PULSE

Diagnóstico con indicadores LED

Indicador LED del 2PULSE



① Error de grupo (rojo)
② Indicador de estado de la entrada digital (verde)
③ Indicador de estado de la salida digital (verde)

Indicadores de estado y error mediante los LEDs del 2PULSE

La siguiente tabla muestra los indicadores de estado y error del 2PULSE.

Evento (LED)					Causa	Solución
SF	1	5	4	8		
ON					No hay parametrización. Hay un aviso de diagnóstico.	Compruebe los bits de error en la interfaz de realimentación. Evalúe el aviso de diagnóstico.
	ON				Entrada DI0 activada en el canal 0.	
		ON			Entrada DI1 activada en el canal 1.	
			ON		Salida DO0 activada en el canal 0.	
				ON	Salida DO1 activada en el canal 1.	

Tipos de error

En el manual de producto del módulo de interfaz utilizado en la estación ET 200S encontrará información sobre la estructura del diagnóstico de canal.

Tipos de error del 2PULSE

La tabla siguiente muestra los tipos de error del 2PULSE.

Tipo de error		Significado	Solución
1_D	00001: cortocircuito	Cortocircuito de la alimentación del encóder o actuador. Una caída de la tensión de alimentación de 24 V también puede ocasionar este aviso de error.	Compruebe el cableado de los interruptores y actuadores. Corregir el cableado del proceso.
9_D	01001: Error	Ha aparecido un error interno en el módulo.	Sustituir el módulo.
16_D	10000: Error de parametrización	Módulo no parametrizado.	Corregir la parametrización.

También es posible obtener una evaluación de errores simplificada en la interfaz de realimentación (Página 10425) a través de los bits correspondientes.

11.4 Control de movimiento

11.4.1 Control de movimiento (S7-1200, S7-1500)

11.4.1.1 Utilizar S7-1200 Motion Control

Introducción

Funcionalidad Motion de la CPU S7-1200

El portal TIA asiste al usuario junto con la funcionalidad "Motion Control" de la CPU S7-1200 al controlar motores paso a paso y servomotores con interfaz de impulsos:

- En el TIA Portal se configuran los objetos tecnológicos Eje de posicionamiento y Tabla de peticiones. Con ayuda de estos objetos tecnológicos la CPU S7-1200 controla las salidas del generador de impulsos para controlar los accionamientos.
- El programa de usuario le permite controlar el eje con las instrucciones Motion Control e iniciar con ello las peticiones de movimiento de su accionamiento.

Encontrará una introducción a Multimedia en Internet (http://www.automation.siemens.com/mcms/topics/en/simatic/simatic-technology/integrated-functions/simatic-s7-1200/Pages/Default.aspx).

Consulte también

Componentes de hardware para Motion Control (Página 10432)

Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

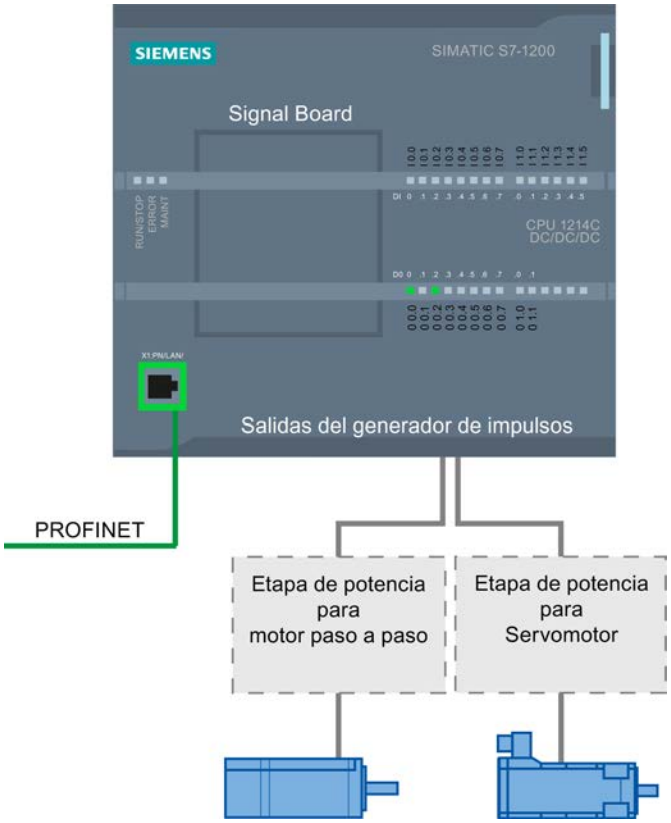
Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Utilización del objeto tecnológico Tabla de peticiones (Página 10482)

Herramientas del objeto tecnológico Tabla de peticiones (Página 10483)

Componentes de hardware para Motion Control

La siguiente representación muestra la estructura básica de hardware para una aplicación Motion Control con la CPU S7-1200.



CPU S7-1200

La CPU S7-1200 unifica la funcionalidad de un autómata programable con la funcionalidad Motion Control para utilizar motores paso a paso y servomotores con interfaz de impulsos. La funcionalidad Motion Control asume el control y la vigilancia de los accionamientos.

Las variantes DC/DC/DC de la CPU S7-1200 disponen de salidas integradas para el control directo de los accionamientos. Las variantes de relé de la CPU requieren una de las Signal Boards descritas a continuación para controlar un accionamiento.

Signal Board

La Signal Board le permite ampliar la CPU con entradas y salidas adicionales. En caso necesario, las salidas digitales pueden utilizarse como salidas del generador de impulsos para controlar accionamientos.

En CPUs con salidas de relé, la señal de impulso no se puede emitir a través salidas integradas, ya que los relés no soportan las frecuencias de conmutación necesarias. Para poder utilizar el PTO en esas CPUs (Pulse Train Output), es necesario utilizar una Signal Board con salidas digitales.

PROFINET

La interfaz PROFINET le permite establecer una conexión online entre la CPU S7-1200 y la programadora. Junto con las funciones online de la CPU dispone además de funciones adicionales de puesta en servicio y diagnóstico para Motion Control.

Información para pedidos de firmware V4 de la CPU

Las siguientes informaciones de pedido listadas a continuación son válidas para la fase de suministro actualmente instalada (sin Hardware Support Packages eventualmente instalados) del TIA Portal.

Designación	Referencia MLFB
CPU 1211C DC/DC/DC	6ES7211-1AE40-0XB0
CPU 1211C AC/DC/RLY	6ES7211-1BE40-0XB0
CPU 1211C DC/DC/RLY	6ES7211-1HE40-0XB0
CPU 1212C DC/DC/DC	6ES7212-1AE40-0XB0
CPU 1212C AC/DC/RLY	6ES7212-1BE40-0XB0
CPU 1212C DC/DC/RLY	6ES7212-1HE40-0XB0
CPU 1214C DC/DC/DC	6ES7214-1AG40-0XB0
CPU 1214C AC/DC/RLY	6ES7214-1BG40-0XB0
CPU 1214C DC/DC/RLY	6ES7214-1HG40-0XB0
CPU 1214FC DC/DC/DC	6ES7214-1AF40-0XB0
CPU 1214FC DC/DC/RLY	6ES7214-1HF40-0XB0
CPU 1215C DC/DC/DC	6ES7215-1AG40-0XB0
CPU 1215C AC/DC/RLY	6ES7215-1BG40-0XB0
CPU 1215C DC/DC/RLY	6ES7215-1HG40-0XB0
CPU 1215FC DC/DC/DC	6ES7215-1AF40-0XB0
CPU 1215FC DC/DC/RLY	6ES7215-1HF40-0XB0
CPU 1217C DC/DC/DC	6ES7217-1AG40-0XB0
Signal Board DI2/DO2 x DC24V 20kHz	6ES7223-0BD30-0XB0
Signal Board DI2/DO2 x DC24V 200kHz	6ES7223-3BD30-0XB0
Signal Board DO4 x DC24V 200kHz	6ES7222-1BD30-0XB0
Signal Board DI2/DO2 x DC5V 200kHz	6ES7223-3AD30-0XB0
Signal Board DO4 x DC5V 200kHz	6ES7222-1AD30-0XB0

Puede instalar nuevos componentes de hardware con un Hardware Support Package (HSP). El componente de hardware queda disponible a continuación en el catálogo de hardware.

Consulte también

Funcionalidad Motion de la CPU S7-1200 (Página 10428)

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10434)

Bases para trabajar con S7-1200 Motion Control

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4)

El número de accionamientos utilizables depende del número de PTO (Pulse Train Outputs) y del número de salidas disponibles del generador de impulsos.

Las siguientes representaciones proporcionan información sobre las correspondientes dependencias:

Número máximo de PTO

Con la versión tecnológica V4 se dispone de 4 PTO por cada CPU. De este modo se pueden controlar 4 accionamientos como máximo.

Tipo de señal del PTO

Dependiendo del tipo de señal seleccionado del PTO, se necesitan 1-2 salidas del generador de impulsos por PTO (accionamiento).

Tipo de señal	Número de salidas del generador de impulsos
Impulso A y sentido B (salida de sentido desactivada) *)	1
Impulso A y sentido B *)	2
Contaje ascendente A, contaje descendente B	2
A/B desfasado	2
A/B desfasado - cuádruple	2

*) La salida de sentido debe encontrarse integrada o en una Signal Board.

Salidas utilizables del generador de impulsos y frecuencias límite

Las variantes de relé de las CPU solo pueden acceder a las salidas del generador de impulsos de una Signal Board.

Dependiendo de la CPU y de la Signal Board, se pueden utilizar las siguientes salidas del generador de impulsos con las frecuencias límite indicadas a continuación:

Integrada	Q0.0	Q0.1	Q0.2	Q0.3	Q0.4	Q0.5	Q0.6	Q0.7	Q1.0	Q1.1
CPU 1211 (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	-	-	-	-	-	-
CPU 1212 (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	30 kHz	30 kHz	-	-	-	-
CPU 1214(F) (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	30 kHz	30 kHz	30 kHz	30 kHz	30 kHz	30 kHz
CPU 1215(F) (DC/DC/DC)	100 kHz	100 kHz	100 kHz	100 kHz	30 kHz	30 kHz	30 kHz	30 kHz	30 kHz	30 kHz
CPU 1217 (DC/DC/DC)	1 MHz	1 MHz	1 MHz	1 MHz	100 kHz	100 kHz	100 kHz	100 kHz	100 kHz	100 kHz
Signal Board	**Qx.0**	**Qx.1**	**Qx.2**	**Qx.3**	-	-	-	-	-	-
Signal Board DI2/DO2 x DC24V 20kHz	20 kHz	20 kHz	-	-	-	-	-	-	-	-
Signal Board DI2/DO2 x DC24V 200kHz	200 kHz	200 kHz	-	-	-	-	-	-	-	-
Signal Board DO4 x DC24V 200kHz	200 kHz	200 kHz	200 kHz	200 kHz	-	-	-	-	-	-
Signal Board DI2/DO2 x DC5V 200kHz	200 kHz	200 kHz	-	-	-	-	-	-	-	-
Signal Board DO4 x DC5V 200kHz	200 kHz	200 kHz	200 kHz	200 kHz	-	-	-	-	-	-

La frecuencia límite inferior es de 1 Hz en cada caso.

Las salidas del generador de impulsos se pueden asignar libremente a los PTO.

Nota

Si se utilizan salidas del generador de impulsos de diferentes frecuencias límite como corresponda al tipo de señal, se utiliza en cada caso la frecuencia límite más baja.

El tipo de señal "Impulso A y sentido B" constituye una excepción; con este tipo de señal se utiliza siempre la frecuencia límite de la salida del generador de impulsos.

Nota

Acceso a las salidas del generador de impulsos a través de la memoria imagen de proceso

Si se ha activado el PTO (Pulse Train Output) y se ha asignado a un eje, el firmware asume el control sobre las salidas del generador de impulsos y de sentido correspondientes.

Al asumir el control se interrumpe también la conexión entre la memoria imagen de proceso y la salida de periferia. Si bien el usuario tiene la posibilidad de escribir en la memoria imagen de proceso de las salidas del generador de impulsos y de sentido con el programa de usuario o la tabla de observación, no se transmite a la salida de periferia. Por consiguiente, tampoco es posible observar la salida de periferia a través del programa de usuario o la tabla de observación. Las informaciones leídas reflejan el valor de la memoria imagen de proceso, las cuales no coinciden con el estado verdadero de la salida de periferia.

En las demás salidas de la CPU no asignadas fijamente por el firmware de la CPU, el estado de la salida de periferia puede ser controlado u monitorizado de la forma habitual a través de la memoria imagen de proceso.

Salidas para señales del accionamiento

Para Motion Control puede parametrizar opcionalmente una interfaz de accionamiento para "Accionamiento habilitado" y "Accionamiento listo".

Si se utiliza la interfaz de accionamiento es posible seleccionar libremente la salida digital para el accionamiento habilitado y la entrada digital para "Accionamiento listo".

Límites de aceleración/deceleración

Para la aceleración y deceleración se aplican los siguientes límites:

Aceleración/deceleración	Valor
Aceleración/deceleración mínimas	5,0E-3 pulsos/s^2
Aceleración/deceleración máximas	9,5E+9 pulsos/s^2

Límites de tirón

Para el tirón son válidos los siguientes límites:

Tirón	Valor
Tirón mínimo	4,0E-3 pulsos/s^3
Tirón máximo	1,0E+10 pulsos/s^3

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V1...3) (Página 10593)

Principio de la interfaz de impulsos (Página 10436)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10437)

Finales de carrera por hardware y por software (Página 10441)

Limitación de tirones (Página 10442)

Referenciar (Página 10443)

Componentes de hardware para Motion Control (Página 10429)

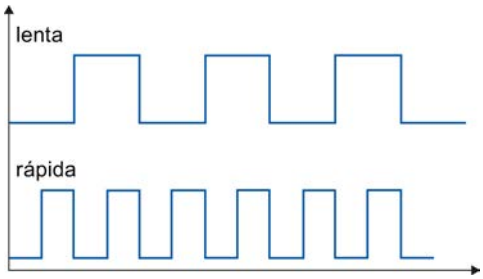
Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Principio de la interfaz de impulsos

En función de los ajustes del motor paso a paso, cada impulso hace que el motor paso a paso se desplace en un ángulo definido. Si el motor paso a paso está ajustado, p. ej., a 1000 impulsos por revolución, el motor paso a paso rotará 0,36° por impulso.

La velocidad del motor paso a paso viene determinada por el número de impulsos por unidad de tiempo.



(Lo aquí enunciado también rige para los servomotores con interfaz de impulsos)

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10437)

Finales de carrera por hardware y por software (Página 10441)

Limitación de tirones (Página 10442)

Referenciar (Página 10443)

Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Relación entre el tipo de señal y el sentido de desplazamiento

La CPU emite la velocidad y el sentido de desplazamiento a través de dos salidas.

Las relaciones entre la configuración y el sentido de desplazamiento difieren en función del tipo de señal seleccionado. En la configuración del eje, en "Parámetros básicos > General", puede configurar los siguientes tipos de señal:

- "PTO (Impulso A y sentido B)"
- "PTO (Contaje ascendente A, contaje descendente B)" (a partir de V4)
- "PTO (A/B desfasado)" (a partir de V4)
- "PTO (A/B desfasado - cuádruple)" (a partir de V4)

El sentido de dirección se ajusta en la configuración del eje, en el área "Parámetros avanzados > Mecánica". Si activa la opción "Invertir sentido", se invierte la lógica de sentido descrita a continuación para el correspondiente tipo de señal.

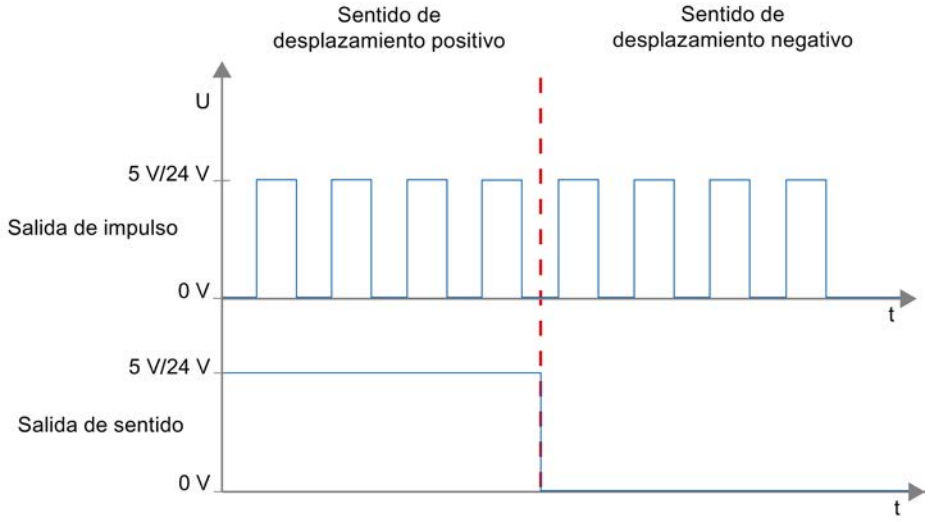
PTO (Impulso A y sentido B)

En este tipo de señal se evalúan los impulsos de la salida de impulsos y el nivel de la salida de sentido.

Los impulsos se emiten a través de la salida de impulso de la CPU. La salida de sentido de la CPU predetermina el sentido de giro del accionamiento:

- 5 V/24 V en la salida de sentido ⇒ sentido de giro positivo
- 0 V en la salida de sentido ⇒ sentido de giro negativo

La tensión indicada depende del hardware empleado. Los valores mencionados no son válidos para las salidas diferenciales de la CPU 1217.

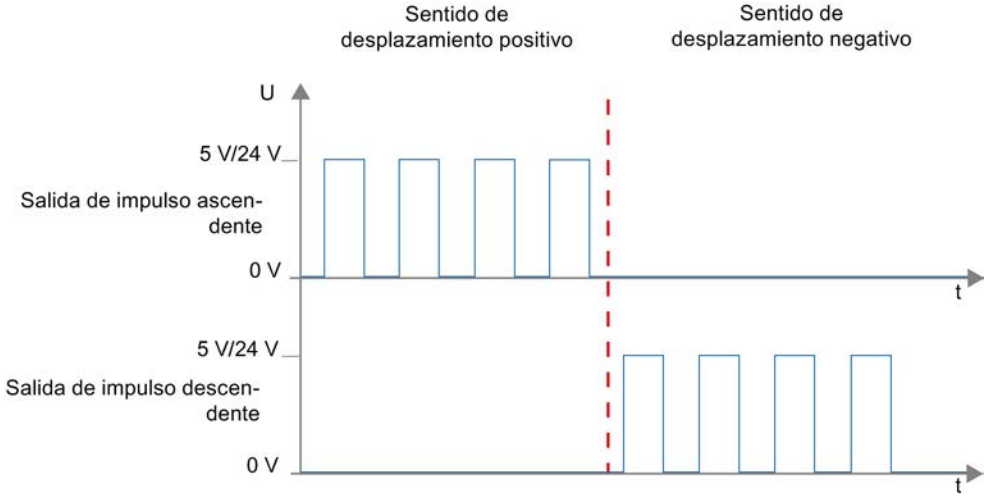


PTO (Contaje ascendente A, contaje descendente B) (a partir de V4)

En este tipo de señal se evalúan los impulsos de una salida.

El impulso para el sentido positivo se emite a través de la "Salida de impulso ascendente". El impulso para el sentido negativo se emite a través de la "Salida de impulso descendente".

La tensión indicada depende del hardware empleado. Los valores mencionados no son válidos para las salidas diferenciales de la CPU 1217.



PTO (A/B desfasado) (a partir de V4)

Para este tipo de señal se evalúan los flancos ascendentes de una salida en cada caso.

El impulso se emite a través de la salida "Fase A" y, desfasado, a través de la salida "Fase B". El desfase entre las salidas define el sentido de giro:

- Fase A adelantada 90° con respecto a fase B ⇒ sentido de giro positivo
- Fase B adelantada 90° con respecto a fase A ⇒ sentido de giro negativo

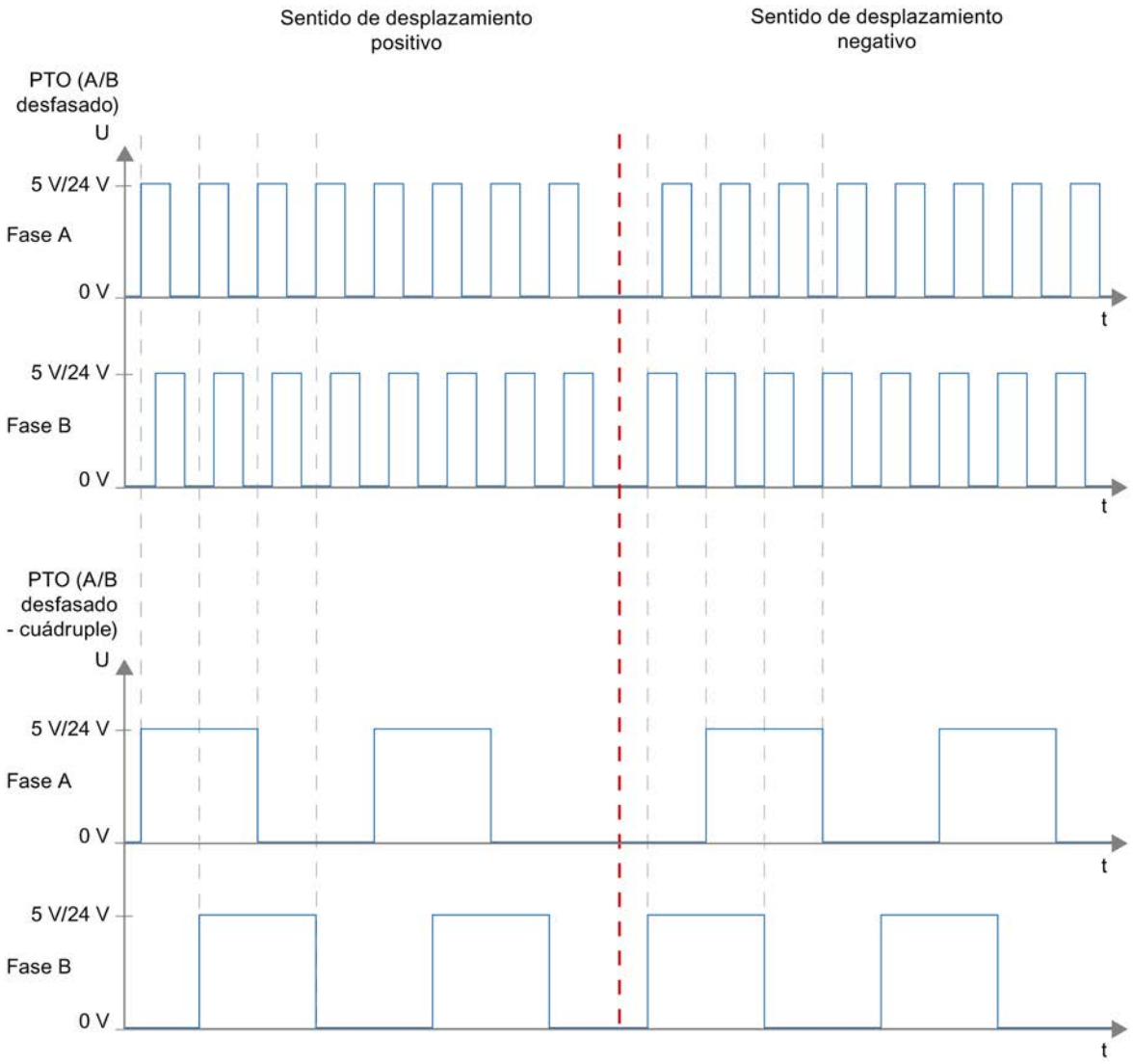
PTO (A/B desfasado - cuádruple) (a partir de V4)

Para este tipo de señal se evalúan los flancos ascendentes y descendentes de las dos salidas. Un período de impulso tiene cuatro flancos en dos fases (A y B). Por ello, la frecuencia de impulsos en la salida se reduce a una cuarta parte.

El impulso se emite a través de la salida "Fase A" y, desfasado, a través de la salida "Fase B". El desfase entre las salidas define el sentido de giro:

- Fase A adelantada 90° con respecto a fase B ⇒ sentido de giro positivo
- Fase B adelantada 90° con respecto a fase A ⇒ sentido de giro negativo

La tensión indicada depende del hardware empleado. Los valores mencionados no son válidos para las salidas diferenciales de la CPU 1217.



Invertir sentido de dirección

Si se activa la opción "Invertir sentido", se invierte la lógica de sentido:

- PTO (Impulso A y sentido B)
  - 0 V en la salida de sentido (nivel bajo) ⇒ sentido de giro positivo
  - 5 V/24 V en la salida de sentido (nivel alto) ⇒ sentido de giro negativo

  La tensión indicada depende del hardware empleado. Las tensiones indicadas no son válidas para las salidas diferenciales de la CPU 1217.
- PTO (Contaje ascendente A, contaje descendente B)
  Las salidas "Salida de impulso descendente" y "Salida de impulso ascendente" se intercambian.

- **PTO (A/B desfasado)**
  Las salidas "Fase A" y "Fase B" se intercambian.
- **"PTO (A/B desfasado - cuádruple)"**
  Las salidas "Fase A" y "Fase B" se intercambian.

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Principio de la interfaz de impulsos (Página 10433)

Finales de carrera por hardware y por software (Página 10441)

Limitación de tirones (Página 10442)

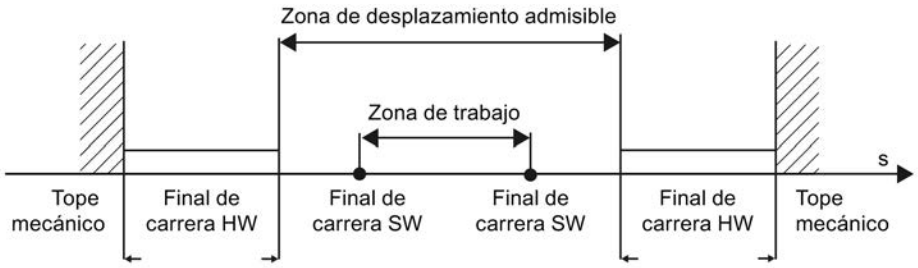
Referenciar (Página 10443)

Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Finales de carrera por hardware y por software

Limite la "zona de desplazamiento admisible" y la "zona de trabajo" de su objeto tecnológico Eje de posicionamiento con finales de carrera por hardware y por software. La siguiente representación recoge las relaciones:



Los finales de carrera de hardware son finales de carrera que limitan la "zona de desplazamiento admisible" máxima del eje. Los finales de carrera por hardware son elementos físicos de conexión que se deben conectar a entradas de la CPU aptas para notificación.

Con los finales de carrera por software se limita la "zona de trabajo" del eje. Éstos deben encontrarse, en relación a la zona de desplazamiento, dentro de los finales de carrera por hardware. Dado que las posiciones de los finales de carrera por software se pueden ajustar con flexibilidad, la zona de trabajo del eje se puede adaptar individualmente según el perfil de desplazamiento actual. Al contrario de lo que ocurre con los finales de carrera por hardware, los finales de carrera por software únicamente pueden ser realizados con el software y no precisan elementos de conexión propios.

Los finales de carrera de hardware y por software se deben activar antes de su utilización en la configuración o en el programa de usuario. Los finales de carrera por software se hacen efectivos después de referenciar el eje.

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Principio de la interfaz de impulsos (Página 10433)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

Limitación de tirones (Página 10442)

Referenciar (Página 10443)

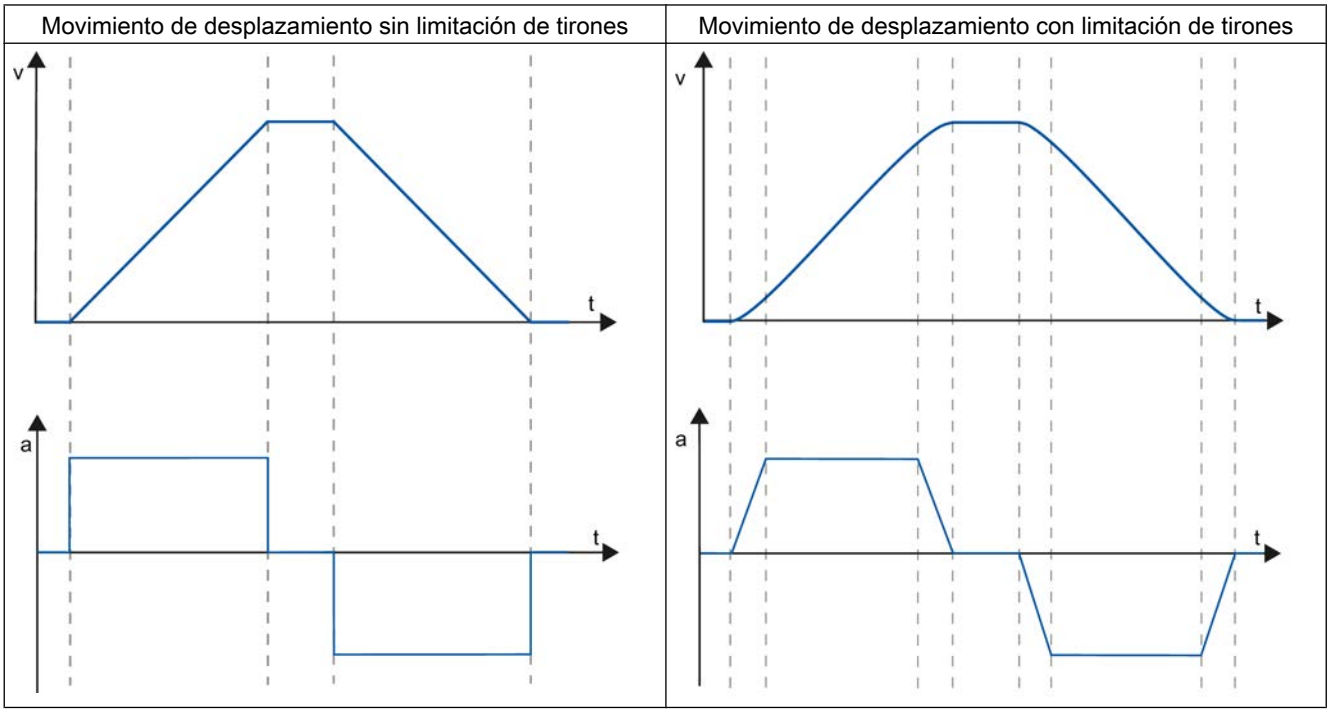
Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Monitorización de posición (Página 10463)

Limitación de tirones

La limitación de tirones le permite reducir los esfuerzos mecánicos durante una rampa de aceleración y deceleración. Si la limitación de tirones está activa, el valor de aceleración y deceleración no se modifica abruptamente, sino que va aumentando o disminuyendo suavemente. La siguiente figura muestra la evolución de la velocidad y la aceleración sin y con limitación de tirones:



La limitación de tirones produce un perfil de velocidad " redondeado" del movimiento del eje. Así se garantiza, p. ej., un arranque y frenado suaves de una cinta transportadora.

Consulte también

Comportamiento del eje al utilizar la limitación de tirones (Página 10472)

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Principio de la interfaz de impulsos (Página 10433)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

Finales de carrera por hardware y por software (Página 10438)

Referenciar (Página 10443)

Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Referenciar

Por referenciar se entiende la sincronización de las coordenadas del objeto tecnológico a la posición física real del accionamiento. Dado que en los ejes con control de posición las entradas e indicaciones de posición se refieren exactamente a esta coordenada del eje, resulta extremadamente relevante que haya una concordancia con la situación real. Solamente así es posible garantizar que la posición absoluta de destino del eje sea alcanzada exactamente en el accionamiento.

En la CPU S7-1200 la referenciación del eje se efectúa con la instrucción de Motion Control "MC_Home". Se distingue entre los siguientes modos de referenciación

Modos de referenciación

- Referenciación activa
  En la referenciación activa, la instrucción de Motion Control "MC_Home" realiza la marcha necesaria al punto de referencia. El eje se referencia, conforme a la configuración, al detectar el sensor del punto de referencia. Los desplazamientos en curso se cancelan.
- Referenciación pasiva
  En la referenciación pasiva, la instrucción de Motion Control "MC_Home" no realiza ningún movimiento de referencia. El desplazamiento necesario para ello debe ser realizado por el usuario con otras instrucciones de Motion Control. El eje se referencia, conforme a la configuración, al detectar el sensor del punto de referencia. Los movimientos en curso no se cancelan cuando se inicia la referenciación pasiva.
- Referenciación directa absoluta
  La posición del eje se fija sin considerar el sensor del punto de referencia. Los procesos de movimiento en curso no se cancelan. El valor del parámetro de entrada "Position" de la instrucción Motion-Control "MC_Home" se aplica inmediatamente como punto de referencia del eje.
- Referenciación directa relativa
  La posición del eje se fija sin considerar el sensor del punto de referencia. Los procesos de movimiento en curso no se cancelan. Para la posición del eje después de la referenciación es válido lo siguiente:
  Nueva posición de eje = posición actual del eje + valor del parámetro "Position" de la instrucción "MC_Home".

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Principio de la interfaz de impulsos (Página 10433)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

Finales de carrera por hardware y por software (Página 10438)

Limitación de tirones (Página 10439)

Integración del objeto tecnológico Eje de posicionamiento (Página 10452)

Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Referenciar (a partir del objeto tecnológico Eje de posicionamiento V2) (Página 10475)

Guía para utilizar Motion Control

La guía aquí descrita muestra el procedimiento básico para utilizar Motion Control con la CPU S7-1200.

Requisitos

Para utilizar el objeto tecnológico Eje de posicionamiento es necesario haber creado un proyecto con una CPU S7-1200.

Procedimiento

Proceda según el orden recomendado a continuación para utilizar Motion Control con la CPU S7-1200. Para ello consulte los enlaces listados:

1. Agregar objeto tecnológico Eje de posicionamiento (Página 10457)
2. Trabajando con el diálogo de configuración (Página 10457)
3. Cargar en la CPU (Página 10502)
4. Test de funcionamiento del eje en la ventana de puesta en marcha (Página 10504)
5. Programar (Página 10526)
6. Diagnosticar el control del eje (Página 10544)

Utilizar versiones

Visión global de la versión

La representación siguiente recoge las relaciones existentes entre las versiones relevantes del S7-1200 Motion Control:

Versión de la tecnología

Se puede comprobar la versión actualmente seleccionada de esta tecnología en la Task Card "Instrucciones", en la carpeta "Tecnología > Motion Control > S7-1200 Motion Control" y en el cuadro de diálogo "Agregar objeto".

La versión de la tecnología se selecciona en la Task Card "Instrucciones", en la carpeta "Tecnología > Motion Control > S7-1200 Motion Control".

Si en el cuadro de diálogo "Agregar objeto" se agrega un TO con una versión alternativa, también se conmuta la versión de la tecnología.

Nota

La selección de una versión alternativa de la tecnología también afecta a la versión de las instrucciones de Motion Control (Task Card).

Los TO y las instrucciones de Motion Control únicamente se convierten conforme a la versión elegida de la tecnología en la compilación o al "Cargar en el dispositivo".

Versión del objeto tecnológico

La versión de un TO se puede comprobar en la ficha "Propiedades > General > Información", en el campo "Versión".

Versión de la instrucción Motion Control

La versión de la instrucción de Motion Control se puede comprobar en la ficha "Propiedades > General > Información", en el campo "Versión".

Si la versión utilizada de la instrucción de Motion Control no responde a la siguiente lista de compatibilidad, las correspondientes instrucciones de Motion Control aparecerán seleccionadas en el editor de programas.

Lista de compatibilidad

Tecnología		CPU	Objeto tecnológico	Instrucción de Motion Control
V1.0		V1.0, V2.0, V2.1, V2.2, V3.0	Eje V1.0	MC_Power V1.0 MC_Reset V1.0 MC_Home V1.0 MC_Halt V1.0 MC_MoveAbsolute V1.0 MC_MoveRelative V1.0 MC_MoveVelocity V1.0 MC_MoveJog V1.0
V2.0	Novedades: • Limitación de tirones • Tabla de peticiones • MC_ChangeDynamic	V2.1, V2.2, V3.0	Eje V2.0, tabla de peticiones V2.0	MC_Power V2.0 MC_Reset V2.0 MC_Home V2.0 MC_Halt V2.0 MC_MoveAbsolute V2.0 MC_MoveRelative V2.0 MC_MoveVelocity V2.0 MC_MoveJog V2.0 MC_CommandTable V2.0 MC_ChangeDynamic V2.0
V3.0	Novedad: Carga en estado operativo RUN	V2.2, V3.0, V4.0	Eje V3.0, tabla de peticiones V3.0	MC_Power V3.0 MC_Reset V3.0 MC_Home V3.0 MC_Halt V3.0 MC_MoveAbsolute V3.0 MC_MoveRelative V3.0 MC_MoveVelocity V3.0 MC_MoveJog V3.0 MC_CommandTable V3.0 MC_ChangeDynamic V3.0
V4.0	Novedades: • MC_ReadParam • MC_WriteParam • Unificación de los bloques de datos tecnológicos de los S7-1200 y S7-1500 Motion Control.	V4.0	Eje de posicionamiento V4.0, tabla de peticiones V4.0	MC_Power V4.0 MC_Reset V4.0 MC_Home V4.0 MC_Halt V4.0 MC_MoveAbsolute V4.0 MC_MoveRelative V4.0 MC_MoveVelocity V4.0 MC_MoveJog V4.0 MC_CommandTable V4.0 MC_ChangeDynamic V4.0 MC_ReadParam V4.0 MC_WriteParam V4.0

Consulte también

Cambio de versión de la tecnología (Página 10447)

Lista de compatibilidad de las variables (Página 10448)

Estado del final de carrera (Página 10451)

Cambio de versión de la tecnología

Para disfrutar de las ventajas de la nueva versión tecnológica, es preciso configurar o, en su caso, cambiar la versión de la tecnología en los proyectos existentes.

Nota

Compatibilidad de las variables del objeto tecnológico

Al cambiar entre V1...3 y V4, tenga en cuenta la lista de compatibilidad (Página 10448) en caso de utilizar variables del objeto tecnológico en el programa de usuario, en las tablas de observación, etc.

Ajuste/cambio de versión tecnológica

Para ajustar/cambiar la versión tecnológica, proceda del siguiente modo:

1. Abra el editor de programas (p. ej. abriendo el OB1).
2. Abra la versión tecnológica deseada en la carpeta "Tecnología > Motion Control > S7-1200 Motion Control" en la Task Card "Instrucciones".
3. Guarde y compile el proyecto. Tenga en cuenta las posibles indicaciones de error al compilar. Elimine las causas de los errores mostrados.
4. Compruebe la configuración de los objetos tecnológicos.
5. Dado el caso, ajuste los nombres de variables según la lista de compatibilidad en los objetos siguientes.

- Programa de usuario
- Tablas de observación
- Tablas de forzado permanente
- Configuraciones HMI
- Configuraciones de Trace

Consulte también

Visión global de la versión (Página 10441)

Estado del final de carrera (Página 10451)

Lista de compatibilidad de las variables

En el marco de la tecnología V4 se han unificado los bloques de datos tecnológicos para S7-1200 Motion Control y S7-1500 Motion Control. De ello resultan nuevas variables y nombres de variables para el objeto tecnológico Eje de posicionamiento V4 y la tabla de peticiones V4.

Tenga en cuenta las siguientes tablas si ha utilizado variables de los objetos tecnológicos en el programa de usuario y desea convertir el proyecto de V1...3 a V4 (o viceversa).

Las variables marcadas en la columna "Conversión automática V1... 3 a V4" se convierten automáticamente al compilar el proyecto. No se convierten los nombres de variables en las tablas de observación y de forzado permanente así como en HMI y configuraciones Trace.

Las siguientes variables son nuevas o han sido adaptadas y tal vez deban ser corregidas en el programa de usuario, en las tablas de observación, etc.

Variables Config (eje de posicionamiento)

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Nombre de eje>.Config.DynamicDefaults.Acceleration	<Nombre de eje>.DynamicDefaults.Acceleration	Sí
<Nombre de eje>.Config.DynamicDefaults.Deceleration	<Nombre de eje>.DynamicDefaults.Deceleration	Sí
<Nombre de eje>.Config.DynamicDefaults.EmergencyDeceleration	<Nombre de eje>.DynamicDefaults.EmergencyDeceleration	Sí
<Nombre de eje>.Config.DynamicDefaults.Jerk	<Nombre de eje>.DynamicDefaults.Jerk	Sí
<Nombre de eje>.Config.DynamicDefaults.JerkActive	No disponible El tirón está activado si el tirón configurado > 0.004 impulsos/s^3.	No
<Nombre de eje>.Config.DynamicLimits.MaxVelocity	<Nombre de eje>.DynamicLimits.MaxVelocity	Sí
<Nombre de eje>.Config.DynamicLimits.MinVelocity	<Nombre de eje>.DynamicLimits.MinVelocity	Sí
<Nombre de eje>.Config.General.LengthUnit	<Nombre de eje>.Units.LengthUnit	Sí
<Nombre de eje>.Config.Homing.AutoReversal	<Nombre de eje>.Homing.AutoReversal	Sí
<Nombre de eje>.Config.Homing.Direction	<Nombre de eje>.Homing.ApproachDirection	Sí
<Nombre de eje>.Config.Homing.FastVelocity	<Nombre de eje>.Homing.ApproachVelocity	Sí
<Nombre de eje>.Config.Homing.Offset	<Nombre de eje>.Sensor[1].ActiveHoming.HomePositionOffset	Sí
<Nombre de eje>.Config.Homing.SideActiveHoming	<Nombre de eje>.Sensor[1].ActiveHoming.SideInput	Sí
<Nombre de eje>.Config.Homing.SidePassiveHoming	<Nombre de eje>.Sensor[1].PassiveHoming.SideInput	Sí

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Nombre de eje>.Config.Homing.SlowVelocity	<Nombre de eje>.Homing.ReferencingVelocity	Sí
<Nombre de eje>.Config.Homing.SwitchedLevel	<Nombre de eje>.Sensor[1].ActiveHoming.SwitchLevel <Nombre de eje>.Sensor[1].PassiveHoming.SwitchLevel	No
<Nombre de eje>.Config.Mechanics.InverseDirection	<Nombre de eje>.Actor.InverseDirection	Sí
<Nombre de eje>.Config.Mechanics.LeadScrew	<Nombre de eje>.Mechanics.LeadScrew	Sí
<Nombre de eje>.Config.Mechanics.PulsesPerDriveRevolution	<Nombre de eje>.Actor.DriveParameter.PulsesPerDriveRevolution	Sí
<Nombre de eje>.Config.PositionLimits_HW.Active	<Nombre de eje>.PositionLimitsHW.Active	Sí
<Nombre de eje>.Config.PositionLimits_HW.MaxSwitchedLevel	<Nombre de eje>.PositionLimitsHW.MaxSwitchLevel	Sí
<Nombre de eje>.Config.PositionLimits_HW.MinSwitchedLevel	<Nombre de eje>.PositionLimitsHW.MinSwitchLevel	Sí
<Nombre de eje>.Config.PositionLimits_SW.Active	<Nombre de eje>.PositionLimitsSW.Active	Sí
<Nombre de eje>.Config.PositionLimits_SW.MaxPosition	<Nombre de eje>.PositionLimitsSW.MaxPosition	Sí
<Nombre de eje>.Config.PositionLimits_SW.MinPosition	<Nombre de eje>.PositionLimitsSW.MinPosition	Sí
No disponible	<Nombre de eje>.Actor.DirectionMode	No
No disponible	<Nombre de eje>.Actor.Type	No
No disponible	<Nombre de eje>.Sensor[1].ActiveHoming.Mode	No
No disponible	<Nombre de eje>.Sensor[1].PassiveHoming.Mode	No

Variables ErrorBits (eje de posicionamiento)

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Nombre de eje>.ErrorBits.HwLimitMax	<Nombre de eje>.ErrorBits.HWLimit (Tenga en cuenta también los nuevos bits de estado y el capítulo Estado del final de carrera (Página 10451).)	No
<Nombre de eje>.ErrorBits.HwLimitMin		
<Nombre de eje>.ErrorBits.SwLimitMaxExceeded	<Nombre de eje>.ErrorBits.SWLimit (Tenga en cuenta también los nuevos bits de estado y el capítulo Estado del final de carrera (Página 10451).)	No
<Nombre de eje>.ErrorBits.SwLimitMaxReached		
<Nombre de eje>.ErrorBits.SwLimitMinExceeded		

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Nombre de eje>.ErrorBits.SwLimitMinReached		
No disponible	<Nombre de eje>.ErrorBits.DirectionFault	No

Variables MotionStatus (eje de posicionamiento)

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Nombre de eje>.MotionStatus.Distance	<Nombre de eje>.StatusPositioning.Distance	Sí
<Nombre de eje>.MotionStatus.Position	<Nombre de eje>.Position	Sí
<Nombre de eje>.MotionStatus.TargetPosition	<Nombre de eje>.StatusPositioning.TargetPosition	Sí
<Nombre de eje>.MotionStatus.Velocity	<Nombre de eje>.Velocity	Sí

Variables StatusBits (eje de posicionamiento)

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Nombre de eje>.StatusBits.Homing	<Nombre de eje>.StatusBits.HomingCommand	Sí
<Nombre de eje>.StatusBits.SpeedCommand	<Nombre de eje>.StatusBits.VelocityCommand	Sí
No disponible	<Nombre de eje>.StatusBits.HWLimitMaxActive	No
No disponible	<Nombre de eje>.StatusBits.HWLimitMinActive	No
No disponible	<Nombre de eje>.StatusBits.SWLimitMaxActive	No
No disponible	<Nombre de eje>.StatusBits.SWLimitMinActive	No

Variables (tabla de peticiones)

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Tabla de peticiones>.Config.Command[n].Position	<Tabla de peticiones>.Command[n].Position	Sí
<Tabla de peticiones>.Config.Command[n].Velocity	<Tabla de peticiones>.Command[n].Velocity	Sí
<Tabla de peticiones>Config.Command[n].Duration	<Tabla de peticiones>.Command[n].Duration	Sí

Nombre de la variable V1.0 a V3.0	Nombre de la variable V4.0	Conversión automática V1..3 a V4
<Tabla de peticiones>Config.Command[n].NextStep	<Tabla de peticiones>.Command[n].NextStep	Sí
<Tabla de peticiones>.Config.Command[n].StepCode	<Tabla de peticiones>.Command[n].StepCode	Sí

Consulte también

Visión global de la versión (Página 10441)

Cambio de versión de la tecnología (Página 10444)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

Estado del final de carrera

Los bits de estado y de error para la indicación de final de carrera alcanzado se han adaptado en la versión V4.

Para emular el comportamiento de los bits de error de las versiones V1...3, utilice las siguientes combinaciones lógicas:

V1...3	V4
<Nombre de eje>.ErrorBits.HwLimitMin	<Nombre de eje>.ErrorBits.HWLimit AND <Nombre de eje>.StatusBits.HWLimitMinActive
<Nombre de eje>.ErrorBits.HwLimitMax	<Nombre de eje>.ErrorBits.HWLimit AND <Nombre de eje>.StatusBits.HWLimitMaxActive
<Nombre de eje>.ErrorBits.SwLimitMinReached	<Nombre de eje>.ErrorBits.SWLimit AND (<Nombre de eje>.Position = <Nombre de eje>.PositioningLimits_SW.MinPosition)
<Nombre de eje>.ErrorBits.SwLimitMinExceeded	<Nombre de eje>.ErrorBits.SWLimit AND (<Nombre de eje>.Position < <Nombre de eje>.PositioningLimits_SW.MinPosition)
<Nombre de eje>.ErrorBits.SwLimitMaxReached	<Nombre de eje>.ErrorBits.SWLimit AND (<Nombre de eje>.Position = <Nombre de eje>.PositioningLimits_SW.MaxPosition)
<Nombre de eje>.ErrorBits.SwLimitMaxExceeded	<Nombre de eje>.ErrorBits.SWLimit AND (<Nombre de eje>.Position > <Nombre de eje>.PositioningLimits_SW.MaxPosition)

Consulte también

Visión global de la versión (Página 10441)

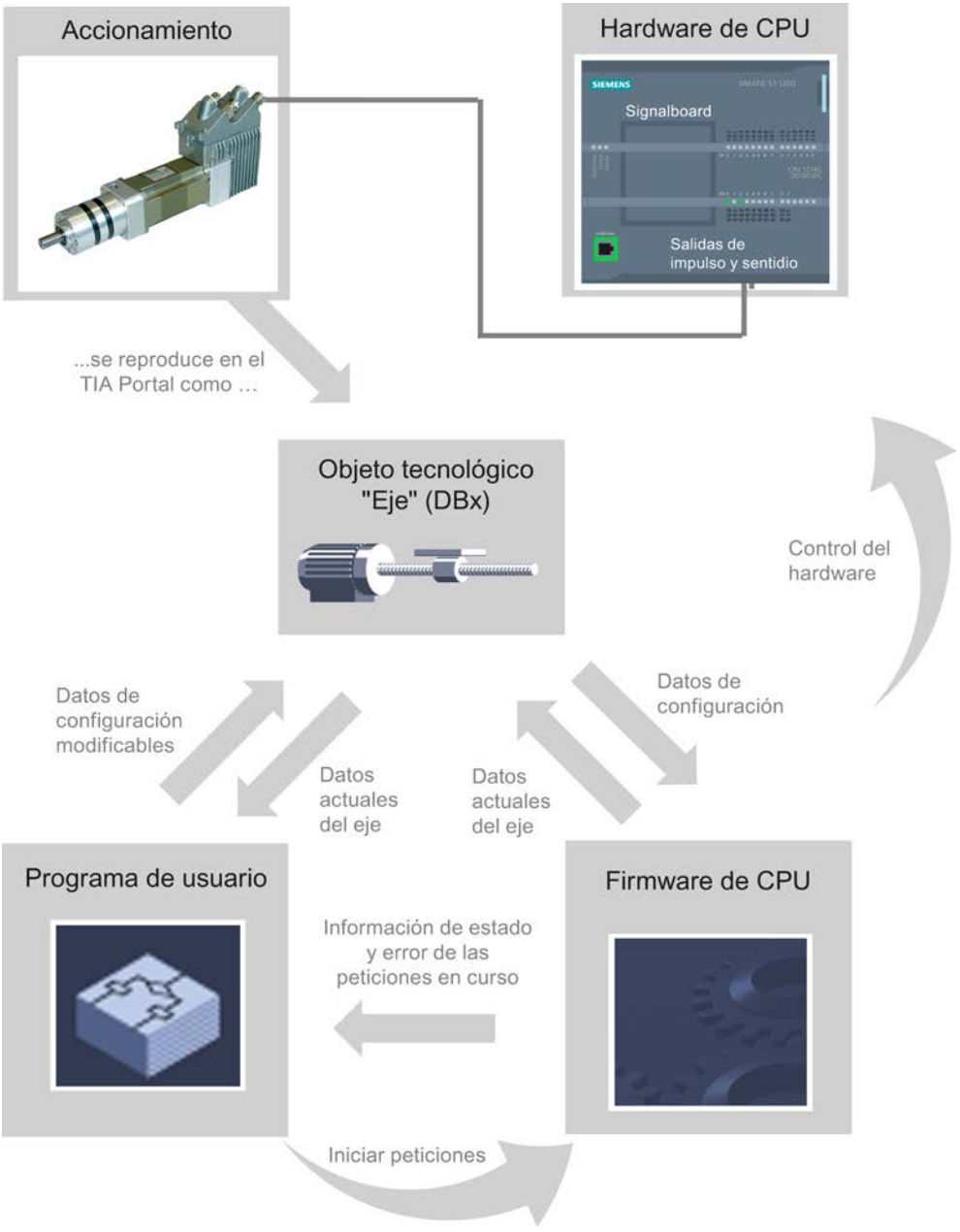
Cambio de versión de la tecnología (Página 10444)

Lista de compatibilidad de las variables (Página 10445)

Objeto tecnológico Eje de posicionamiento

Integración del objeto tecnológico Eje de posicionamiento

En la siguiente representación se muestran las relaciones que se producen entre los componentes de hardware y software cuando se utiliza el objeto tecnológico Eje de posicionamiento:



Hardware de la CPU

El hardware de la CPU permite controlar y supervisar el accionamiento físico.

Accionamiento

El accionamiento representa la unidad formada por la etapa de potencia y el motor. Se pueden utilizar motores paso a paso o servomotores con interfaz de impulsos.

Objeto tecnológico Eje de posicionamiento

El accionamiento físico, mecanismo incluido, se representa en el TIA Portal como TO Eje de posicionamiento. Configure el objeto tecnológico Eje de posicionamiento con los siguientes parámetros:

- Selección del PTO que va a utilizarse (Pulse Train Output) y configuración de la interfaz del accionamiento
- Parámetros mecánicos y de multiplicación del engranaje del accionamiento (o de la máquina o instalación)
- Parámetros para la monitorización de posición, para parámetros de dinámica y para referenciación

La configuración del objeto tecnológico Eje de posicionamiento se guarda en el objeto tecnológico (bloque de datos). Este bloque de datos conforma simultáneamente la interfaz entre el programa de usuario y el firmware de la CPU. Durante el tiempo de ejecución del programa de usuario los datos actuales del eje se guardan en el bloque de datos del objeto tecnológico.

Programa de usuario

El programa de usuario le permite iniciar peticiones en el firmware de la CPU mediante instrucciones de Motion Control. Se pueden realizar las siguientes peticiones para controlar el eje:

- Habilitar y bloquear el eje
- Posicionar eje de forma absoluta
- Posicionar eje de forma relativa
- Mover eje con preajuste de velocidad
- Ejecutar peticiones de eje como secuencia de movimientos (a partir de la tecnología V2)
- Mover eje en modo Jog
- Parar eje
- Referenciar eje; fijar punto de referencia
- Modificar ajustes dinámicos del eje
- Leer continuamente datos de movimiento del eje
- Escribir variable del eje
- Acusar errores

Los parámetros de entrada de las instrucciones de Motion Control y la configuración del eje le permiten determinar los parámetros de la petición. Los parámetros de salida de la instrucción le proporcionan información actual sobre el estado y eventuales errores de la petición.

Antes de iniciar una petición para el eje debe habilitarlo con la instrucción de Motion Control "MC_Power".

Las variables del objeto tecnológico le permiten leer en el programa de usuario los datos de configuración y los datos actuales del eje. Puede modificar variables individuales modificables del objeto tecnológico (p. ej. la aceleración actual) desde el programa de usuario.

Además, con la instrucción de Motion Control "MC_ChangeDynamic" se pueden modificar los ajustes de dinámica del eje y con "MC_WriteParam" se pueden escribir otros datos de configuración. Con la instrucción de Motion Control "MC_ReadParam" se puede leer el estado de movimiento actual del eje.

Firmware de la CPU

Las peticiones de Motion Control lanzadas desde el programa de usuario se procesan en el firmware de la CPU. Al utilizar el panel de mando del eje, el lanzamiento de las peticiones de Motion Control se realiza a través del mando del eje. El firmware de la CPU cumple las siguientes funciones conforme a la configuración del eje:

- Cálculo del perfil de movimiento exacto para peticiones de movimiento y situaciones de parada de emergencia
- Control de la habilitación del accionamiento, así como de la señal de impulsos y sentido
- Vigilancia del accionamiento y de los finales de carrera por hardware y por software
- Realimentación actual de informaciones de estado y error de las peticiones a las instrucciones de Motion Control en el programa de usuario
- Escritura de datos actuales del eje en el bloque de datos del objeto tecnológico

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

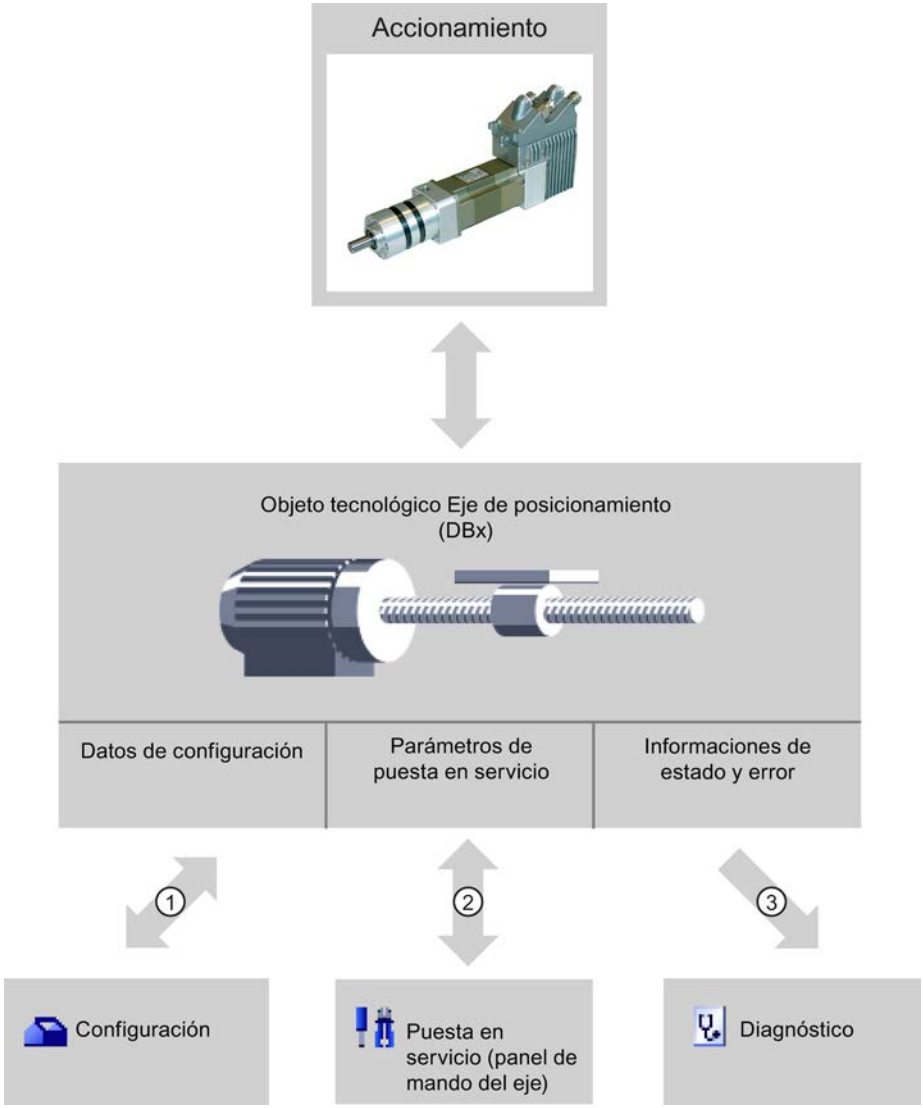
Herramientas del objeto tecnológico Eje de posicionamiento (Página 10455)

Finales de carrera por hardware y por software (Página 10438)

Referenciar (Página 10440)

Herramientas del objeto tecnológico Eje de posicionamiento

El TIA Portal dispone de las herramientas "Configuración", "Puesta en servicio" y "Diagnóstico" para el objeto tecnológico Eje de posicionamiento. La siguiente representación muestra la interacción de las tres herramientas con el objeto tecnológico y el accionamiento:



①	Escritura y lectura de los datos de configuración del objeto tecnológico
②	Control del accionamiento a través del objeto tecnológico. Lectura del estado del eje para su visualización en el panel de mando del eje
③	Lectura de las informaciones actuales de estado y error del objeto tecnológico

Configuración

Configure las siguientes propiedades del objeto tecnológico Eje de posicionamiento con la herramienta "Configuración":

- Selección del PTO empleado y configuración de la interfaz del accionamiento
- Propiedades de la mecánica y multiplicación del engranaje, del accionamiento (o de la máquina o instalación)
- Propiedades para la monitorización de posición, para parámetros de dinámica y para la referenciación

La configuración se guarda en el bloque de datos del objeto tecnológico.

Puesta en servicio

Realice un test de funcionamiento de su eje con la herramienta "Puesta en servicio", sin necesidad de crear un programa de usuario. Al iniciar la herramienta se abre el panel de mando del eje. El panel de mando del eje incorpora los siguientes comandos:

- Habilitación y bloqueo del eje
- Desplazamiento del eje en modo Jog
- Posicionamiento absoluto y relativo del eje
- Referenciación del eje
- Acuse de errores

Los valores dinámicos se pueden adaptar en consecuencia para los comandos de movimiento. El panel de mando del eje muestra también el estado actual del eje.

Diagnóstico

Compruebe con la herramienta "Diagnóstico" las informaciones actuales de estado y error del eje y del accionamiento.

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

Integración del objeto tecnológico Eje de posicionamiento (Página 10449)

Finales de carrera por hardware y por software (Página 10438)

Referenciar (Página 10440)

Configurar el objeto tecnológico Eje de posicionamiento (Página 10457)

Puesta en servicio del eje - Panel de control del eje (Página 10504)

Diagnóstico del eje (Página 10544)

Agregar objeto tecnológico Eje de posicionamiento

Requisitos

Hay un proyecto creado con una CPU S7-1200.

Procedimiento

Para agregar un objeto tecnológico Eje de posicionamiento en el árbol del proyecto, proceda del siguiente modo:

1. Abra la carpeta "CPU > Objetos tecnológicos" en el árbol del proyecto.
2. Haga doble clic en el comando "Agregar objeto".
   Se abre el cuadro de diálogo "Agregar objeto".
3. Seleccione la tecnología "Motion Control".
4. Abra la carpeta "Motion Control > S7-1200 Motion Control".
5. Seleccione la versión deseada de la tecnología en la columna "Versión".
6. Seleccione el objeto "TO_PositioningAxis".
7. En el campo de entrada "Nombre" introduzca el nombre del eje.
8. Para modificar los números asignados automáticamente al bloque de datos, seleccione la opción "manual".
9. Para visualizar y añadir más información sobre el objeto tecnológico, haga clic en "Más información".
10. Confirme la entrada realizada con "Aceptar".

Resultado

Se crea el nuevo objeto tecnológico y se guarda en la capeta "Objetos tecnológicos" del árbol del proyecto.

Consulte también

Guía para utilizar Motion Control (Página 10441)

Configurar el objeto tecnológico Eje de posicionamiento

Trabajando con el diálogo de configuración

Las propiedades del objeto tecnológico se configuran en la ventana de configuración. Para abrir la ventana de configuración del objeto tecnológico, proceda del siguiente modo:

1. En el árbol del proyecto, abra el grupo correspondiente al objeto tecnológico que desea utilizar.
2. Haga doble clic en el objeto "Configuración".

La configuración se divide en las siguientes categorías:

- **Parámetros básicos**
  Los parámetros básicos contienen todos los parámetros que deben configurarse para un eje listo para funcionar.
- **Parámetros avanzados**
  Los parámetros avanzados contienen parámetros que se pueden adaptar al propio accionamiento o bien a la instalación.

Símbolos de la ventana de configuración

Los símbolos que aparecen en la navegación local de la configuración muestran más detalles sobre el estado de la configuración:

	La configuración contiene valores predeterminados y está completa. La configuración contiene exclusivamente valores predeterminados. Con ellos es posible utilizar el objeto tecnológico sin más modificaciones.
	La configuración contiene valores definidos por el usuario y está completa Todos los campos de entrada de la configuración contienen valores válidos y al menos un valor predeterminado ha sido modificado.
	La configuración es errónea o está incompleta Al menos un campo de entrada o una lista desplegable contiene un valor no válido. El campo o la lista desplegable en cuestión se marca en rojo. Al hacer clic, el roll out con el aviso de error indica la causa del error.
	La configuración es válida, pero contiene advertencias P. ej., solo se ha configurado un final de carrera por hardware. Dependiendo de la instalación, si un final de carrera por hardware no está configurado ello puede representar un peligro. El campo o la lista desplegable en cuestión se marca en amarillo.

Consulte también

Guía para utilizar Motion Control (Página 10441)

Parámetros básicos (Página 10459)

Parámetros avanzados (Página 10462)

Comparar valores

La función "Comparar valores" ofrece las siguientes opciones:

- Comparación de los valores de arranque configurados del proyecto con los valores de arranque de la CPU y los valores actuales
- Procesamiento directo de los valores actuales y de los valores de arranque del proyecto
- Detección y visualización inmediatas de errores de introducción con propuestas de corrección
- Copia de seguridad de los valores actuales en el proyecto
- Transferencia de los valores de arranque del proyecto a la CPU como valores actuales

Símbolos y elementos de control

Están disponibles los siguientes símbolos y elementos de control:

Símbolo	Función
	El valor de arranque en la CPU es igual al valor de arranque configurado en el proyecto
	El valor de arranque en la CPU es diferente al valor de arranque configurado en el proyecto
	No se puede realizar la comparación entre valor de arranque en la CPU y valor de arranque configurado en el proyecto
	Se guarda una copia de seguridad de los valores actuales en el proyecto
	Los valores de arranque actualizados en el proyecto se transfieren a la CPU (inicializar valores de ajuste)
	Se abre el cuadro de diálogo "Comparar valores"

Parámetros básicos

Configuración - General (objeto tecnológico Eje de posicionamiento a partir de V4)

Configure las propiedades básicas del objeto tecnológico Eje de posicionamiento en la ventana de configuración "General".

Nombre del eje

Defina en este campo el nombre del eje o del objeto tecnológico Eje de posicionamiento. El objeto tecnológico aparece con ese nombre en el árbol del proyecto.

Hardware - Interfaz

Los impulsos se transmiten a la etapa de potencia del accionamiento a través de salidas digitales de asignación fija.

En CPUs con salidas de relé la señal de impulso no puede emitirse en ellas, ya que los relés no soportan las frecuencias de conmutación necesarias. Para poder utilizar el PTO en esas CPUs (Pulse Train Output), es necesario utilizar una Signal Board con salidas digitales.

Nota

El PTO requiere la funcionalidad de un contador rápido (HSC). Para ello se utiliza un HSC interno cuyo estado de contador no se puede evaluar.

Selección del generador de impulsos

En la lista desplegable elija el PTO (Pulse Train Output) a través del cual deben proporcionarse los impulsos para controlar los motores paso a paso o los servomotores con la interfaz de impulsos. Si en la configuración de dispositivos no se han utilizado los generadores de impulso

ni los contadores rápidos para otro fin, la interfaz por hardware podrá configurarse automáticamente. En este caso, el PTO elegido aparece marcado en blanco en la lista desplegable.

Botón "Configuración de dispositivos"

Este botón permite acceder a la parametrización de las opciones de impulso en la configuración de dispositivos de la CPU.

Tipo de señal

Seleccione el tipo de señal deseado en la lista desplegable. Están disponibles los tipos de señal siguientes:

- PTO (Impulso A y sentido B)
  Para controlar el motor paso a paso se utilizan una salida de impulso y una salida de sentido.
- PTO (Contaje ascendente A, contaje descendente B)
  Para controlar el motor paso a paso se utilizan una salida de impulso para el movimiento en sentido positivo y otra para el movimiento en sentido negativo.
- PTO (A/B desfasado)
  Los ciclos de las dos salidas de impulsos para la fase A y para la fase B tienen la misma frecuencia.
  En el lado del accionamiento se evalúa el periodo de las salidas de impulso como paso.
  El desfase entre las fases A y B determina el sentido de movimiento.
- PTO (A/B desfasado - cuádruple)
  Las salidas de impulsos para la fase A y para la fase B tienen la misma frecuencia.
  En el lado del accionamiento se evalúan todos los flancos ascendentes y descendentes de la fase A y de la fase B como pasos.
  El desfase entre las fases A y B determina el sentido de movimiento.

Salida de impulso (tipo de señal "PTO (Impulso A y sentido B)")

En este campo seleccione la salida deseada como salida de impulsos.

Puede elegirse la salida a través de una dirección simbólica o asignarse a una dirección absoluta.

Activar salida de sentido (tipo de señal "PTO (Impulso A y sentido B)")

En el modo "pulse and direction" puede desactivar o activar la salida de sentido. Con esta opción puede limitar el sentido.

Salida de sentido (tipo de señal "PTO (Impulso A y sentido B)")

En este campo seleccione la salida deseada como salida de sentido.

Puede elegirse la salida a través de una dirección simbólica o asignarse a una dirección absoluta.

Salida de impulso ascendente (tipo de señal "PTO (Contaje ascendente A, contaje descendente B)")

En este campo seleccione la salida de impulso deseada para movimientos en sentido positivo.

Puede elegirse la salida a través de una dirección simbólica o asignarse a una dirección absoluta.

Salida de impulso descendente (tipo de señal "PTO (Contaje ascendente A, contaje descendente B)")

En este campo seleccione la salida de impulso deseada para movimientos en sentido negativo.

Puede elegirse la salida a través de una dirección simbólica o asignarse a una dirección absoluta.

Fase A (tipos de señal "PTO (A/B desfasado)" y "PTO (A/B desfasado - cuádruple)")

En este campo seleccione la salida de impulso deseada para las señales de fase A.

Puede elegirse la salida a través de una dirección simbólica o asignarse a una dirección absoluta.

Fase B (tipos de señal "PTO (A/B desfasado)" y "PTO (A/B desfasado - cuádruple)")

En este campo seleccione la salida de impulso deseada para las señales de fase B.

Puede elegirse la salida a través de una dirección simbólica o asignarse a una dirección absoluta.

Unidad personalizada

En la lista desplegable, seleccione la unidad deseada para el sistema de medida del eje. La unidad seleccionada se utiliza para una nueva configuración del objeto tecnológico Eje de posicionamiento y la visualización de los datos actuales del eje.

Los valores en los parámetros de entrada (Position, Distance, Velocity, ...) de las instrucciones de Motion Control también se refieren a esta unidad.

Nota

En determinadas circunstancias, un cambio posterior del sistema de medida no podrá convertirse correctamente en todas las ventanas de configuración del objeto tecnológico. En ese caso, compruebe la configuración de todos los parámetros del eje.

Si fuera necesario, en el programa de usuario se deberán adaptar los valores de los parámetros de entrada de las instrucciones de Motion Control a la nueva unidad de medida.

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

Configuración - General (objeto tecnológico "Eje" V1...3) (Página 10597)

Parámetros avanzados

Configuración - Señales del accionamiento

En la ventana de configuración "Señal de accionamiento" configure la salida para la habilitación del accionamiento y la entrada para la realimentación "Accionamiento listo" del accionamiento.

La habilitación del accionamiento es controlada por la instrucción de Motion Control "MC_Power" y otorga al accionamiento la habilitación de potencia. La señal se pone a disposición del accionamiento a través de la salida que se va a configurar.

Si el accionamiento está listo para ejecutar movimientos una vez recibida la habilitación del accionamiento, lo notificará a la CPU con la señal "Accionamiento listo". La señal "Accionamiento listo" se reenvía a la CPU a través de la entrada que se va a configurar.

Si el accionamiento no dispone de ninguna interfaz de este tipo, no se deberán configurar los parámetros. En este caso, elija para la entrada de disponibilidad el valor TRUE.

Consulte también

Configuración - Mecánica (Página 10462)

Monitorización de posición (Página 10463)

Dinámica (Página 10468)

Referenciar (a partir del objeto tecnológico Eje de posicionamiento V2) (Página 10475)

Configuración - Mecánica

Configure las propiedades mecánicas del accionamiento en la ventana de configuración "Mecánica".

Impulsos por vuelta del motor

En este campo, configure cuántos impulsos necesita el motor para una vuelta del motor.

Límites (independientes de la unidad de medida seleccionada):

- 0 < impulsos por vuelta del motor ≤ 2147483647

Recorrido por vuelta del motor

En este campo, configure qué distancia debe recorrer la mecánica de la instalación por cada vuelta del motor.

Límites (independientes de la unidad de medida seleccionada):

- 0.0 < recorrido por vuelta del motor: ≤ 1.0e12

Sentido de giro permitido (versión de la tecnología V4 o superior)

En este campo configure si la mecánica de su instalación debe moverse en ambos sentidos o solo en sentido positivo o negativo.

Si no ha activado la salida de sentido en el modo "PTO (Impulso A y sentido B)" del generador de impulsos, la selección está limitada al sentido positivo o negativo.

Invertir sentido de dirección

Con la casilla de verificación "Invertir sentido" puede adaptar el controlador a la lógica de sentido del accionamiento.

La lógica de sentido se invierte de acuerdo con el modo seleccionado para el generador de impulsos:

- **PTO (Impulso A y sentido B)**
  - 0 V en la salida de sentido ⇒ sentido de giro positivo
  - 5 V/24 V en la salida de sentido ⇒ sentido de giro negativo

  La tensión indicada depende del hardware empleado. Los valores mencionados no son válidos para las salidas diferenciales de la CPU 1217.
- **PTO (Contaje ascendente A, contaje descendente B)**
  Las salidas "Salida de impulso descendente" y "Salida de impulso ascendente" se intercambian.
- **PTO (A/B desfasado)**
  Las salidas "Fase A" y "Fase B" se intercambian.
- **"PTO (A/B desfasado - cuádruple)"**
  Las salidas "Fase A" y "Fase B" se intercambian.

Consulte también

Configuración - Señales del accionamiento (Página 10459)

Monitorización de posición (Página 10463)

Dinámica (Página 10468)

Referenciar (a partir del objeto tecnológico Eje de posicionamiento V2) (Página 10475)

Relación entre el tipo de señal y el sentido de desplazamiento (Página 10434)

Monitorización de posición

Requisitos del final de carrera de hardware

Utilice exclusivamente finales de carrera de hardware que permanezcan conectados permanentemente durante la puesta en movimiento. Este estado de conexión no se podrá cambiar hasta regresar al área de desplazamiento admisible.

Consulte también

Configuración - Monitorización de posición (Página 10464)

Comportamiento del eje al reaccionar las monitorizaciones de posición (Página 10465)

Modificar la configuración de la monitorización de posición en el programa de usuario (Página 10467)

Configuración - Monitorización de posición

En la ventana de configuración "Monitorización de posición" configure los finales de carrera por hardware y por software del eje.

Activar finales de carrera HW

Active con esta casilla de verificación la función de los finales de carrera por hardware inferior y superior. Durante una aproximación al punto de referencia activa se pueden utilizar los finales de carrera por hardware para invertir el sentido. Encontrará más detalles en la descripción de la configuración para la referenciación.

Entrada final de carrera HW inferior/superior

Seleccione la entrada digital para el final de carrera por hardware inferior o superior en la lista desplegable. La entrada debe ser apta para alarmas. Como entradas para los finales de carrera por hardware están disponibles las entradas digitales de CPU On-board y las entradas digitales de una Signal Board insertada.

Nota

Las entradas digitales están ajustadas de forma estándar con un tiempo de filtración de 6,4 ms. Al utilizar como final de carrera por hardware pueden producirse deceleraciones no deseadas. En dicho caso reduzca el tiempo de filtración para las salidas digitales correspondientes.

El tiempo de filtración se puede ajustar en la configuración de dispositivos de las entradas digitales, en "Filtros de entrada".

Selección de nivel

Elija en la lista desplegable el nivel de señal activo en la CPU con el final de carrera por hardware alcanzado.

- Selección "Nivel inferior"
  0 V (FALSE) en la entrada de CPU representa final de carrera por hardware alcanzado
- Selección "Nivel superior"
  5 V / 24 V (TRUE) en la entrada de la CPU equivale a aproximación al final de carrera por hardware (la tensión indicada depende del hardware empleado)

Activar finales de carrera por software

Active con esta casilla de verificación la función de los finales de carrera por software inferior y superior.

Nota

Los finales de carrera por software activados sólo son efectivos con el eje referenciado.

Posición final carrera SW inferior/superior

En estos campos se especifica el valor de posición de los finales de carrera por software inferior y superior.

Límites (independientes de la unidad de medida seleccionada):

- -1.0e12 ≤ Posición final carrera SW inferior ≤ 1.0e12
- -1.0e12 ≤ Posición final carrera SW superior ≤ 1.0e12

El valor del final de carrera por software superior debe ser mayor o igual al valor del final de carrera por software inferior.

Consulte también

Requisitos del final de carrera de hardware (Página 10460)

Comportamiento del eje al reaccionar las monitorizaciones de posición (Página 10465)

Modificar la configuración de la monitorización de posición en el programa de usuario (Página 10467)

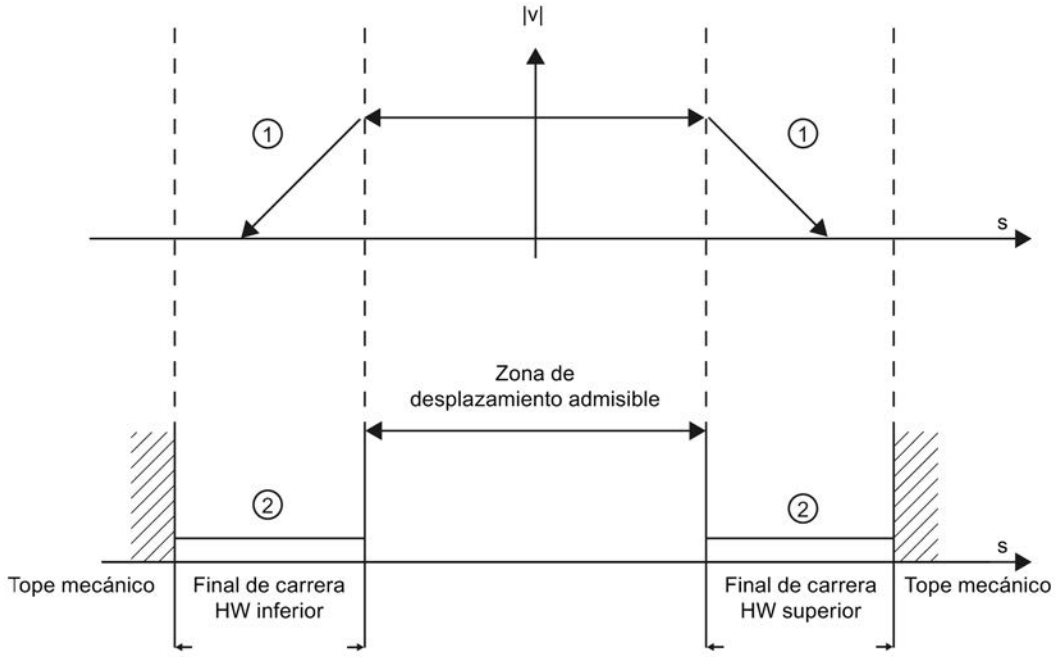
Configuración - Referenciar - Activa (Página 10476)

Comportamiento del eje al reaccionar las monitorizaciones de posición

Comportamiento del eje al alcanzar el final de carrera por hardware

Al alcanzar el final de carrera por hardware el eje se detiene con la deceleración de parada de emergencia configurada hasta pararse por completo. Se debe elegir una deceleración de parada de emergencia lo suficientemente grande, para que el eje se pare de forma segura

hasta el tope mecánico. La siguiente representación muestra el comportamiento del eje después de alcanzar el final de carrera por hardware:



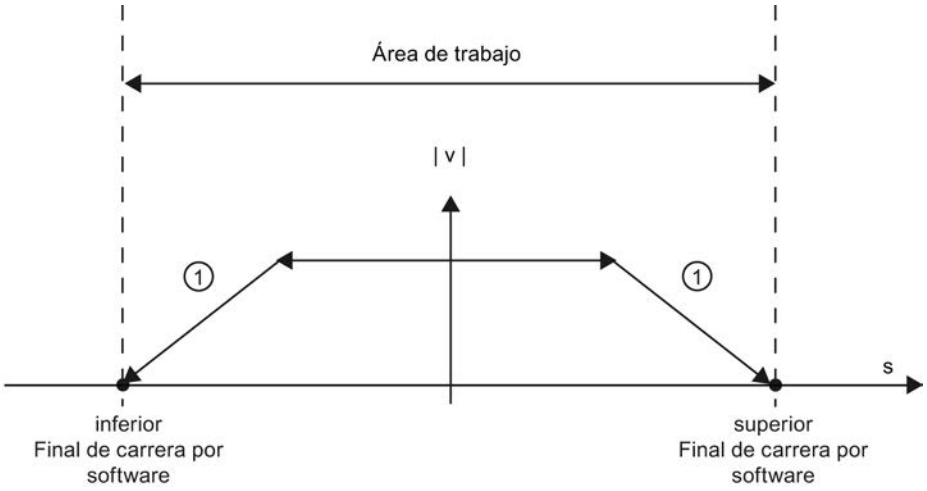
①	El eje frena con la deceleración de parada de emergencia configurada hasta pararse.
②	Área en el que los finales de carrera por hardware notifican el estado "alcanzado".

En la instrucción de Motion Control causante, en "MC_Power" y en las variables del objeto tecnológico se muestra el error "final de carrera por hardware alcanzado". En el anexo encontrará una guía para eliminar el error, en el apartado "Lista de ErrorIDs y ErrorInfos"

Comportamiento del eje al alcanzar el final de carrera por hardware

Con los finales de carrera por software activos, cualquier movimiento en curso se detiene en la posición del final de carrera por software. El eje es frenado con la deceleración de parada de emergencia configurada.

La siguiente representación muestra el comportamiento del eje hasta alcanzar el final de carrera por software:



①	El eje frena con la deceleración configurada hasta pararse.

En la instrucción de Motion Control causante, en "MC_Power" y en las variables del objeto tecnológico se muestra el error "Final de carrera por software alcanzado". En el anexo encontrará una guía para eliminar el error, en el apartado "Lista de ErrorIDs y ErrorInfos".

Las circunstancias que pueden dar lugar al rebase del final de carrera por software se pueden consultar en los capítulos "Finales de carrera por software y el proceso de referenciación (Página 10555)" y "Final de carrera por software y modificaciones en la dinámica (Página 10560)".

Utilice adicionalmente finales de carrera por hardware cuando detrás de los finales de carrera por software se encuentre un tope mecánico con riesgo de que se pueda producir un daño mecánico.

Consulte también

Requisitos del final de carrera de hardware (Página 10460)

Configuración - Monitorización de posición (Página 10461)

Modificar la configuración de la monitorización de posición en el programa de usuario (Página 10467)

Modificar la configuración de la monitorización de posición en el programa de usuario

Los siguientes parámetros de configuración pueden modificarse en la CPU mientras se ejecuta el programa de usuario:

Fin de carrera de hardware

También puede activar y desactivar los finales de carrera de hardware durante el tiempo de ejecución del programa de usuario. Para ello utilice la siguiente variable del objeto tecnológico:

- <Nombre de eje>.PositionLimitsHW.Active

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico (Página 10573).

Fin de carrera por software

También puede activar y desactivar los finales de carrera por software y modificar sus valores de posición durante el tiempo de ejecución del programa de usuario. Para ello utilice las siguientes variables del objeto tecnológico:

- <Nombre de eje>.PositionLimitsSW.Active
  para activar y desactivar los finales de carrera por software
- <Nombre de eje>.PositionLimitsSW.MinPosition
  para modificar la posición del final de carrera por software inferior
- <Nombre de eje>.PositionLimitsSW.MaxPosition
  para modificar la posición del final de carrera por software superior

Para saber cuándo tienen efecto las modificaciones de los parámetros de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico.

Consulte también

Lista de compatibilidad de las variables (Página 10445)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

Requisitos del final de carrera de hardware (Página 10460)

Configuración - Monitorización de posición (Página 10461)

Comportamiento del eje al reaccionar las monitorizaciones de posición (Página 10462)

Dinámica

Configuración - Dinámica General

En la ventana de configuración "Dinámica General" configure la velocidad máxima, la velocidad de arranque/parada, la aceleración y deceleración, así como la limitación de tirones del eje (a partir del objeto tecnológico Eje de posicionamiento V2).

Unidad de los límites de velocidad

En la lista desplegable, seleccione la unidad física con la que desea fijar los límites de velocidad. La unidad aquí ajustada es independiente de la unidad de medida ajustada en " Configuración - General" y su función es exclusivamente la de simplificar la entrada.

Velocidad máxima / Velocidad de arranque/parada

Defina en estos campos la velocidad máxima admisible y la velocidad de arranque/parada del eje. La velocidad de arranque/parada es la velocidad mínima admisible del eje.

Límites:

Los límites mencionados a continuación se refieren a la unidad de medida "Impulso/s":

- Objeto tecnológico Eje de posicionamiento V4
  - 1 ≤ velocidad de arranque y parada ≤ 20000 (Signal Board 20 kHz)
    1 ≤ velocidad de arranque y parada ≤ 200000 (Signal Board 200 kHz)
    1 ≤ velocidad de arranque y parada ≤ 100000 (salidas integradas de la CPU 100 kHz)
    1 ≤ velocidad de arranque y parada ≤ 30000 (salidas integradas de la CPU 30 kHz)
    1 ≤ velocidad de arranque y parada ≤ 1000000 (salidas integradas de la CPU 1217 1 MHz)
  - 1 ≤ velocidad máxima ≤ 20000 (Signal Board 20 kHz)
    1 ≤ velocidad máxima ≤ 200000 (Signal Board 200 kHz)
    1 ≤ velocidad máxima ≤ 100000 (salidas integradas de la CPU 100 kHz)
    1 ≤ velocidad máxima ≤ 30000 (salidas integradas de la CPU 30 kHz)
    1 ≤ velocidad máxima ≤ 1000000 (salidas integradas de la CPU 1217 1 MHz)

Los límites para el objeto tecnológico Eje de posicionamiento < V4 se pueden consultar en el anexo Salidas de la CPU relevantes para Motion Control (versión tecnológica V1...3) (Página 10593)

El valor de la velocidad máxima debe ser mayor o igual al valor de la velocidad de arranque/parada.

Los valores límite para otras unidades de medida deben ser convertidos por el usuario conforme a la mecánica dada.

Aceleración / deceleración - Tiempo de aceleración/tiempo de deceleración

Ajuste la aceleración deseada en los campos "Tiempo de aceleración" o "Aceleración". La deceleración deseada se puede ajustar en los campos "Tiempo deceleración" o "Deceleración".

La relación entre el tiempo de aceleración y la aceleración, así como entre el tiempo de deceleración y la deceleración se expresa a través de las siguientes ecuaciones:

$$\text{Tiempo de posicionamiento} = \frac{\text{Velocidad máxima} - \text{Velocidad de arranque/parada}}{\text{Aceleración}}$$

$$\text{Tiempo de deceleración} = \frac{\text{Velocidad máxima} - \text{Velocidad de arranque/parada}}{\text{Retardo}}$$

Las peticiones de movimiento lanzadas a través del programa de usuario se ejecutan con la aceleración / deceleración seleccionada.

Puede consultar los límites para la aceleración y deceleración en el capítulo Componentes de hardware para Motion Control (Página 10429).

Nota

Las modificaciones realizadas en los límites de velocidad ("Velocidad de arranque/parada", así como "Velocidad máxima") afectan a los valores de aceleración y deceleración del eje. Los tiempos de aceleración y deceleración se conservan.

Activar limitación de tirones (a partir del objeto tecnológico Eje de posicionamiento V2)

Active la limitación de tirones con esta casilla de verificación.

Nota

En caso de error, el eje decelera con la deceleración de parada de emergencia configurada. Una limitación de tirones activada no se tiene en cuenta en dicho caso.

Tiempo de redondeo / tirones (a partir del objeto tecnológico Eje de posicionamiento V2)

Los parámetros de limitación de tirones pueden introducirse en el campo "Tiempo de redondeo" u opcionalmente en el campo "Tirón":

- Ajuste el tirón deseado para la rampa de aceleración y deceleración en el campo "Tirón".
- Ajuste el tiempo de redondeo deseado para la rampa de aceleración en el campo "Tiempo de redondeo".

Nota

Tiempo de redondeo V2...3

El tiempo de redondeo ajustado y visible en la configuración sólo es válido para la rampa de aceleración.

En el caso de que los valores de aceleración y deceleración sean distintos, el tiempo de redondeo de la rampa de deceleración se calcula y se utiliza conforme al tirón de la rampa de aceleración. (véase también Comportamiento del eje al utilizar la limitación de tirones (Página 10472))

El tiempo de redondeo de la deceleración se adapta del siguiente modo:

- **aceleración > deceleración**
  En la rampa de deceleración se aplica un tiempo de redondeo menor que en la rampa de aceleración.
- **aceleración < deceleración**
  En la rampa de deceleración se aplica un tiempo de redondeo mayor que en la rampa de aceleración.
- **aceleración = deceleración**
  Los tiempos de redondeo de la rampa aceleración y de la rampa de deceleración son iguales.

La relación entre los tiempos de redondeo y el tirón se expresa a través de las siguientes ecuaciones:

$$\text{Tiempo de redondeo (rampa de aceleración)} = \frac{\text{Aceleración}}{\text{Tirón}}$$

$$\text{Tiempo de redondeo (rampa de deceleración)} = \frac{\text{Deceleración}}{\text{Tirón}}$$

Las peticiones de movimiento lanzadas a través del programa de usuario se ejecutan con el tirón seleccionado.

Puede consultar los límites para el tirón en el capítulo Componentes de hardware para Motion Control (Página 10429).

Consulte también

Comportamiento del eje al utilizar la limitación de tirones (Página 10472)

Componentes de hardware para Motion Control (Página 10429)

Configuración - Dinámica Parada de emergencia (Página 10471)

Modificar la configuración de los valores de dinámica en el programa de usuario (Página 10474)

Configuración - Dinámica Parada de emergencia

Configure la deceleración de parada de emergencia del eje en la ventana de configuración "Dinámica Parada de emergencia". En caso de error y al producirse un bloqueo del eje con la instrucción de Motion Control "MC_Power" (parámetro de entrada StopMode = 0 o 2), el eje se para con esta deceleración.

Límites de velocidad

Para una mejor perspectiva, en esta área se vuelven a mostrar los valores de velocidad parametrizados en la ventana de configuración "Dinámica General".

Deceleración

Ajuste el valor de deceleración para la parada de emergencia en los campos "Deceleración de parada de emergencia" o "Tiempo de deceleración de parada de emergencia".

La relación entre el tiempo de deceleración de parada de emergencia y la deceleración de parada de emergencia se expresa a través de la siguiente ecuación:

$$\text{Tiempo de deceleración de parada de emergencia} = \frac{\text{Velocidad máxima - velocidad de arranque y parada}}{\text{Deceleración de parada de emergencia}}$$

La deceleración de parada de emergencia debe elegirse lo suficientemente grande para que el eje pueda pararse a tiempo en caso de emergencia (p. ej. al alcanzar el final de carrera por hardware, antes de alcanzar el tope mecánico).

Al seleccionar la deceleración de parada de emergencia ésta se debe basar en la velocidad máxima configurada del eje.

Límites:

Los límites mencionados a continuación se refieren a la unidad de medida "Impulso/s^2".

- a partir del firmware V3 de la CPU
  0.005 ≤ deceleración de parada de emergencia ≤ 9.5E9
- Firmware V1...2 de la CPU
  0.28 ≤ deceleración de parada de emergencia ≤ 9.5E9

Los límites para otras unidades de medida se deben convertir conforme a la mecánica dada.

Consulte también

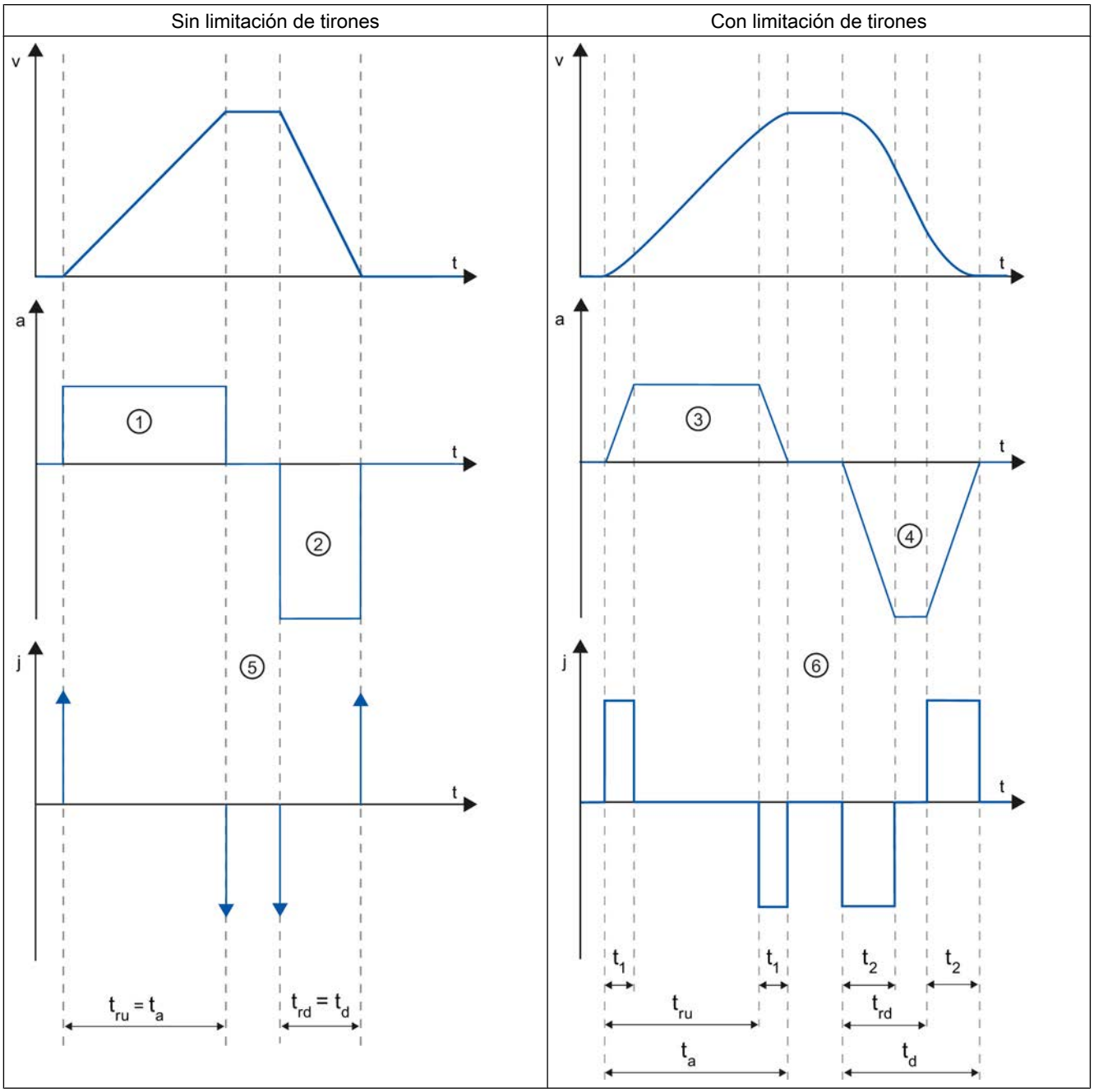
Configuración - Dinámica General (Página 10465)

Modificar la configuración de los valores de dinámica en el programa de usuario (Página 10474)

Comportamiento del eje al utilizar la limitación de tirones

Al activar la limitación de tirones, la aceleración y deceleración del eje no se modifican de forma abrupta, sino que se adaptan suavemente en función del tirón o los tiempos de redondeo

ajustados. La siguiente representación muestra detalladamente el comportamiento del eje con y sin limitación de tirones activada:



t	Eje temporal
v	Velocidad
a	Aceleración
j	Tirón
t_{ru}	Tiempo aceleración

t_a	Tiempo para acelerar el eje
t_{rd}	Tiempo deceleración
t_d	Tiempo para decelerar el eje
t_1	Tiempo de redondeo de la rampa de aceleración
t_2	Tiempo de redondeo de la rampa de deceleración

En el ejemplo se representa un movimiento de desplazamiento en el que el valor de la deceleración ② es dos veces más grande que el de la aceleración ①. De aquí se deduce un tiempo de deceleración t_{rd} que sólo representa la mitad del tiempo de aceleración t_{ru}.

Sin limitación de tirones la aceleración ① y el retraso ② se modifican de forma abrupta. Con la limitación de tirones activada, la aceleración ③ y la deceleración ④ se modifican de forma suave. Dado que el tirón es válido para todo el movimiento, la pendiente es idéntica para el aumento y para la disminución de la aceleración y de la deceleración.

Sin limitación de tirones los valores para el tirón j son infinitamente grandes ⑤ en el momento del cambio. Con limitación de tirones el tirón se limita al valor configurado ⑥.

El tiempo de redondeo t_1 indicado en la configuración se aplica para la rampa de aceleración. El tiempo de redondeo de la rampa de deceleración t_2 se calcula a partir del valor configurado para el tirón y de la deceleración configurada.

Consulte también

Configuración - Dinámica General (Página 10465)

Modificar la configuración de los valores de dinámica en el programa de usuario

Los siguientes parámetros de configuración pueden modificarse en la CPU mientras se ejecuta el programa de usuario:

Aceleración y deceleración

Puede modificar los valores de la aceleración y deceleración incluso durante el tiempo de ejecución del programa de usuario. Para ello utilice las siguientes variables del objeto tecnológico:

- <Nombre de eje>.DynamicDefaults.Acceleration
  para modificar la aceleración
- <Nombre de eje>.DynamicDefaults.Deceleration
  para modificar la deceleración

Para saber cuándo tienen efecto las modificaciones de los parámetros de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico (Página 10573).

Deceleración de parada de emergencia

Puede modificar el valor de la deceleración de parada de emergencia incluso durante el tiempo de ejecución del programa de usuario. Para ello utilice la siguiente variable del objeto tecnológico:

- <Nombre de eje>.DynamicDefaults.EmergencyDeceleration

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico.

Nota

Una vez modificado este parámetro puede resultar necesario tener que adaptar las posiciones de los finales de carrera de hardware, así como otras configuraciones relevantes para la seguridad.

Limitación de tirones

También puede activar y desactivar la limitación de tirones durante el tiempo de ejecución del programa de usuario y modificar el valor del tirón. Para ello utilice la variable del objeto tecnológico <Nombre de eje>.DynamicDefaults.Jerk.

Si introduce para el tirón un valor > 0.004 impulsos/s^3, la limitación de tirón estará activada con el valor introducido.

Si introduce un valor = 0.0 para el tirón, la limitación de tirón estará desactivada.

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico.

Consulte también

Cambio de configuración de los valores dinámicos en el programa de usuario (objeto tecnológico "Eje" V1...3) (Página 10599)

Lista de compatibilidad de las variables (Página 10445)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

Configuración - Dinámica General (Página 10465)

Configuración - Dinámica Parada de emergencia (Página 10468)

Referenciar (a partir del objeto tecnológico Eje de posicionamiento V2)

Configuración - Referenciar - Pasiva

En la ventana de configuración "Referenciar - Pasiva", configure los parámetros que se necesitan para la referenciación pasiva.

En la referenciación pasiva, el movimiento debe ser activado por parte del usuario (p. ej., mediante una petición de desplazamiento del eje). La referenciación pasiva se inicia a través

de la instrucción de Motion Control "MC_Home" con el parámetro de entrada utilizado "Mode" = 2.

Entrada del sensor del punto de referencia

Seleccione la entrada digital para el sensor del punto de referencia en la lista desplegable. La entrada debe ser apta para alarmas. Como entradas para el sensor del punto de referencia están disponibles las entradas de CPU On-board y las entradas de una Signal Board insertada.

Nota

Las entradas digitales están ajustadas de forma estándar con un tiempo de filtrado de 6,4 ms.

La utilización como sensor del punto de referencia puede provocar deceleraciones no deseadas y, con ello, imprecisiones. En determinadas circunstancias, y dependiendo de la velocidad de referenciación y la dimensión del sensor del punto de referencia, puede que no se reconozca el punto de referencia. El tiempo de filtración se puede ajustar en la configuración de dispositivos de las entradas digitales, en "Filtros de entrada".

Se debe seleccionar un tiempo de filtración menor que la duración de la señal de entrada en el sensor del punto de referencia.

Selección de nivel

Seleccione en la lista despegable el nivel del sensor del punto de referencia con el que se debe referenciar.

Lado del sensor del punto de referencia

Indique aquí si el eje debe ser referenciado por el lado inferior o superior al sensor del punto de referencia.

Posición del punto de referencia

Como posición del punto de referencia se utiliza la posición parametrizada en la instrucción de Motion Control "MC_Home".

Nota

Si la referenciación pasiva se efectúa sin una petición de desplazamiento del eje (eje en reposo), la referenciación se lleva a cabo en el siguiente flanco ascendente o descendente del sensor del punto de referencia.

Configuración - Referenciar - Activa

En la ventana de configuración "Referenciar - Activa", configure los parámetros que se necesitan para la referenciación activa. La referenciación activa se inicia a través de la instrucción de Motion Control "MC_Home" con el parámetro de entrada utilizado "Mode" = 3.

Entrada del sensor del punto de referencia

Seleccione la entrada digital para el sensor del punto de referencia en la lista desplegable. La entrada debe ser apta para alarmas. Como entradas para el sensor del punto de referencia están disponibles las entradas de CPU On-board y las entradas de una Signal Board insertada.

Nota

Las entradas digitales están ajustadas de forma estándar con un tiempo de filtrado de 6,4 ms.

La utilización como sensor del punto de referencia puede provocar deceleraciones no deseadas y, con ello, imprecisiones. En determinadas circunstancias, y dependiendo de la velocidad de referenciación y la dimensión del sensor del punto de referencia, puede que no se reconozca el punto de referencia. El tiempo de filtración se puede ajustar en la configuración de dispositivos de las entradas digitales, en "Filtros de entrada".

Se debe seleccionar un tiempo de filtración menor que la duración de la señal de entrada en el sensor del punto de referencia.

Selección de nivel

Seleccione en la lista despegable el nivel del sensor del punto de referencia con el que se debe referenciar.

Permitir inversión de sentido en el final de carrera por hardware

Active esta casilla de verificación si desea utilizar los finales de carrera por hardware como levas de inversión para la aproximación al punto de referencia. Los finales de carrera por hardware deben estar activados para poder realizar la inversión del sentido (al menos debe estar configurado el final de carrera por hardware en el sentido de aproximación).

Si se llega al final de carrera por hardware durante la referenciación activa, el eje frena con la deceleración configurada (no con la deceleración de parada de emergencia) y realiza una inversión del sentido. Entonces el sensor del punto de referencia se busca en la dirección contraria.

Si esta inversión del sentido no está activada y el eje alcanza el final de carrera por hardware durante la referenciación activa, la aproximación al punto de referencia se cancela con un error y el eje se frena con la deceleración de parada de emergencia.

Nota

Dentro de lo posible, asegúrese de que la máquina no golpee contra un tope mecánico durante una inversión del sentido, adoptando para ello una de las siguientes medidas:

- Mantenga una velocidad de aproximación reducida.
- Aumente la aceleración / deceleración configuradas.
- Aumente la distancia entre el final de carrera por hardware y el tope mecánico.

Sentido de aproximación / referenciación

Con el sentido seleccionado, determine el sentido de aproximación durante la referenciación activa a fin de buscar el sensor del punto de referencia, así como el sentido de referenciación. El sentido de referenciación determina el sentido con el cual el eje se aproxima al lado del sensor del punto de referencia configurado para realizar la operación de referenciación.

Lado del sensor del punto de referencia

Indique aquí si el eje debe ser referenciado por el lado inferior o superior al sensor del punto de referencia.

Velocidad de aproximación

Especifique en este campo la velocidad con la que se buscará el sensor del punto de referencia durante la marcha correspondiente.

Límites (independientes de la unidad de medida seleccionada):

- Velocidad de arranque/parada ≤ velocidad de aproximación ≤ velocidad máxima

Velocidad de referenciación

Especifique en este campo la velocidad con la que debe realizarse el posicionamiento hacia el sensor del punto de referencia para la referenciación.

Límites (independientes de la unidad de medida seleccionada):

- Velocidad de arranque/parada ≤ velocidad de referenciación ≤ velocidad máxima

Offset del punto de referencia

En este campo se puede indicar el offset del punto de referencia si la posición de referencia deseada difiere de la posición del sensor del punto de referencia.

Si el valor es diferente de 0, el eje ejecuta las siguientes acciones después de la referenciación en el sensor del punto de referencia:

1. Movimiento del eje a la velocidad de referenciación por el valor del offset del punto de referencia
2. Después de extraer el offset del punto de referencia, el eje se encuentra en la posición del punto de referencia que se ha indicado en el parámetro de entrada "Position" de la instrucción de Motion Control "MC_Home".

Límites (independientes de la unidad de medida seleccionada):

- -1.0e12 ≤ offset del punto de referencia: ≤ 1.0e12

Posición del punto de referencia

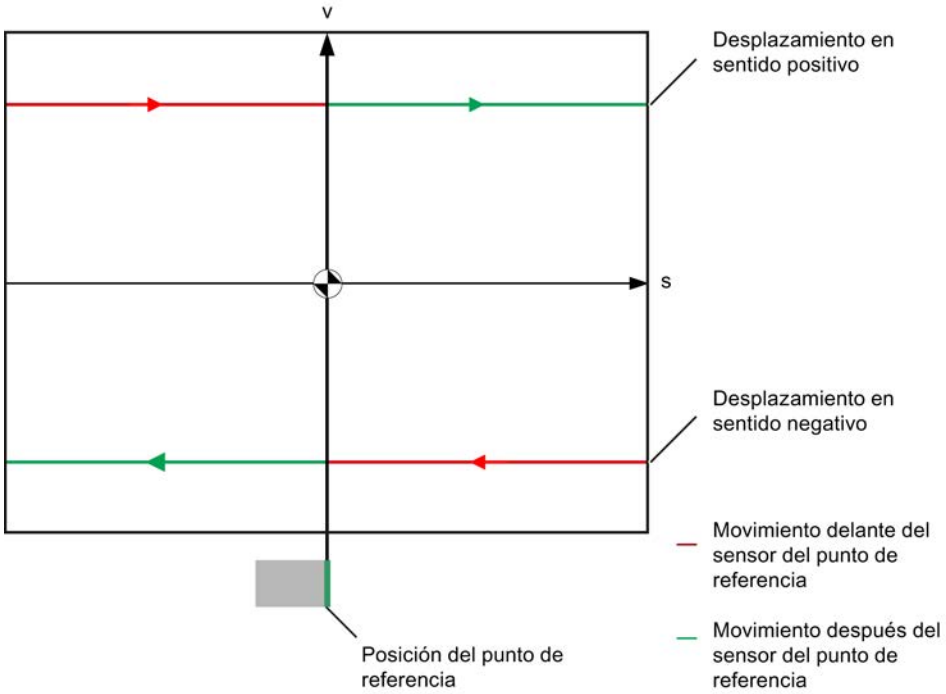
Como posición del punto de referencia se utiliza la posición parametrizada en la instrucción de Motion Control "MC_Home".

Marcha - Referenciación pasiva

La referenciación pasiva se inicia con la instrucción de Motion Control "MC_Home" (parámetro de entrada Mode = 2). El parámetro de entrada "Position" indica aquí la coordenada absoluta del punto de referencia.

La siguiente representación muestra un ejemplo del desarrollo de un proceso de referenciación pasiva con los siguientes parámetros de configuración:

- "Lado del sensor del punto de referencia" = Lado superior"



Movimiento antes del sensor del punto de referencia (tramo de curva rojo)

Con el inicio de la referenciación pasiva, la instrucción de Motion Control "MC_Home" no realiza ningún movimiento de referencia. El movimiento de desplazamiento necesario para alcanzar el sensor del punto de referencia deber realizarse por parte del usuario a través de otras instrucciones Motion Control, como por ejemplo MC_MoveRelative. Si el eje ya había sido referenciado anteriormente, la variable <Nombre de eje>.StatusBits.HomingDone permanece en TRUE durante el referenciado pasivo.

Referenciación del eje (transición del tramo de curva rojo al verde)

El eje se referencia al alcanzar el lado configurado del sensor del punto de referencia. La posición actual del eje se establece en la posición del punto de referencia. Ésta se indica en el parámetro "Position" de la instrucción de Motion Control "MC_Home". Si aún no se había referenciado el eje, la variable <Nombre de eje>.StatusBits.HomingDone se establece en TRUE. El movimiento de desplazamiento iniciado antes no se cancela.

Movimiento después del sensor del punto de referencia (tramo de curva verde)

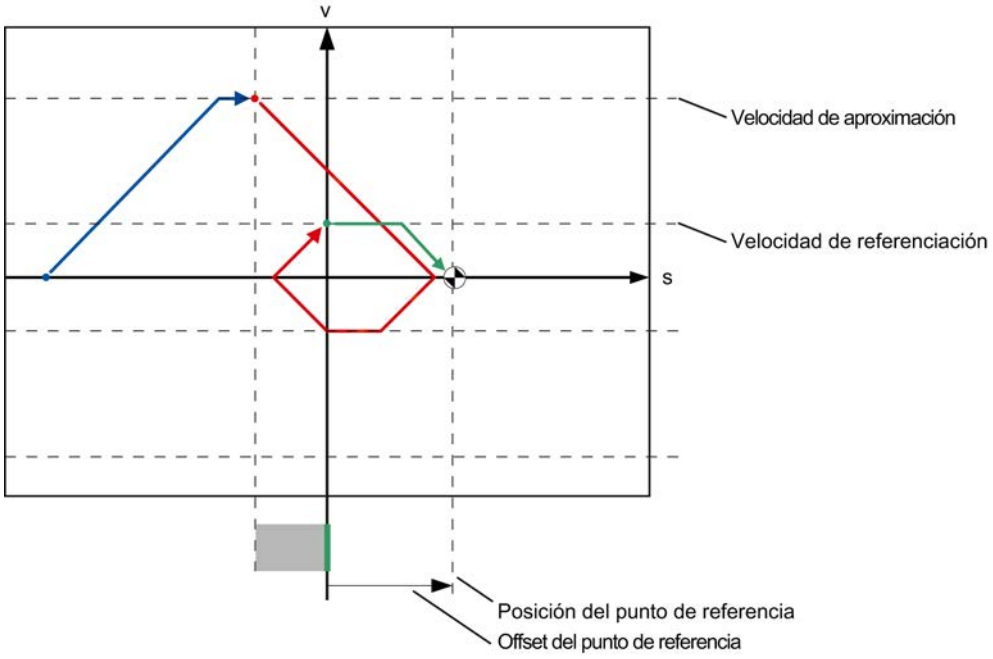
Después de la referenciación en el sensor del punto de referencia el eje continúa el movimiento de desplazamiento iniciado antes con la posición del eje corregida hasta el final.

Marcha - referenciación activa

La referenciación activa se inicia con la instrucción de Motion Control "MC_Home" (parámetro de entrada Mode = 3). El parámetro de entrada "Position" indica aquí la coordenada absoluta del punto de referencia. La referenciación activa también puede iniciarse a través del panel de mando del eje para fines de prueba.

La siguiente representación muestra un ejemplo del desarrollo de una búsqueda activa del punto de referencia con los siguientes parámetros de configuración:

- "Sentido de aproximación / referenciación" = "sentido positivo"
- "Lado del sensor del punto de referencia" = Lado superior"
- Valor del "offset del punto de referencia" > 0



Búsqueda del sensor del punto de referencia (tramo de curva azul)

Al iniciar la referenciación activa, el eje acelera a la "velocidad de aproximación" configurada, buscando el sensor del punto de referencia a dicha velocidad. La variable <Nombre de eje>.StatusBits.HomingDone se establece en FALSE.

Aproximación al punto de referencia (tramo de curva rojo)

En este ejemplo, el eje frena en cuanto detecta el sensor del punto de referencia e invierte para referenciar a la "velocidad de posicionamiento" configurada en el lado configurado del

sensor del punto de referencia. Con la referenciación la variable <Nombre de eje>.StatusBits.HomingDone cambia a TRUE.

Extracción del offset del punto de referencia (tramo de curva verde)

Una vez finalizada la referenciación, el eje recorre la distancia del offset del punto de referencia a la velocidad de referenciación. Una vez allí, el eje se queda en la posición del punto de referencia que se ha indicado en el parámetro de entrada "Position" de la instrucción de Motion Control "MC_Home".

Consulte también

Configuración - Referenciar - General (objeto tecnológico Eje V2...3) (Página 10603)

Modificar la configuración para la referenciación en el programa de usuario

A partir del objeto tecnológico Eje de posicionamiento V2 los siguientes parámetros de configuración pueden modificarse en la CPU mientras se ejecuta el programa de usuario:

Referenciación pasiva

El lado del sensor del punto de referencia para la referenciación pasiva puede modificarse mientras se ejecuta el programa de usuario. Para ello utilice la siguiente variable del objeto tecnológico:

- <Nombre de eje>.Sensor[1].PassiveHoming.SideInput
  para modificar el lado del sensor del punto de referencia

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico (Página 10573).

Referenciación activa

Es posible modificar el sentido de aproximación, el lado del sensor del punto de referencia, la velocidad de aproximación, la velocidad de referenciación y el offset del punto de referencia para la referenciación activa mientras se ejecuta el programa de usuario. Para ello utilice las siguientes variables del objeto tecnológico:

- <Nombre de eje>.Homing.AutoReversal
  para modificar la inversión del sentido en el final de carrera por hardware
- <Nombre de eje>.Homing.ApproachDirection
  para modificar el sentido de aproximación o de referenciación
- <Nombre de eje>.Sensor[1].ActiveHoming.SideInput
  para modificar el lado del sensor del punto de referencia
- <Nombre de eje>.Homing.ApproachVelocity
  para modificar la velocidad de aproximación

- <Nombre de eje>.Homing.ReferencingVelocity
  para modificar la velocidad de referenciación
- <Nombre de eje>.Sensor[1].ActiveHoming.HomePositionOffset
  para modificar el offset del punto de referencia

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico.

Consulte también

Lista de compatibilidad de las variables (Página 10445)

MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)

Objeto tecnológico Tabla de peticiones

Utilización del objeto tecnológico Tabla de peticiones

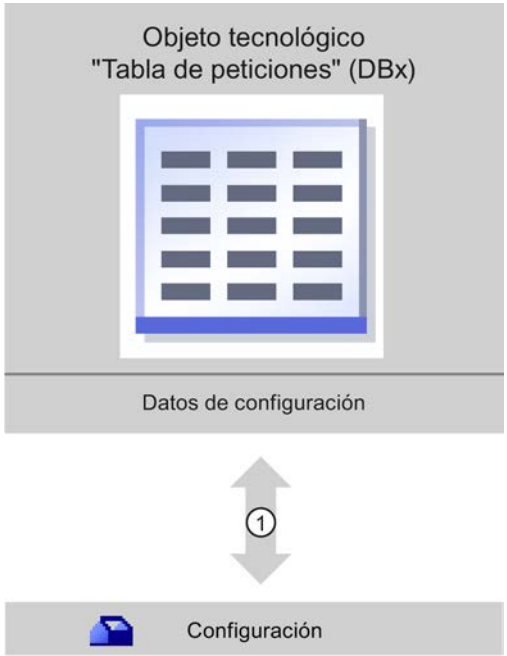
Utilice el objeto tecnológico "Tabla de peticiones" para reunir varias peticiones individuales en secuencias de movimiento a fin de controlar un eje. El TO se puede utilizar a partir de la versión V2 de la tecnología.

La secuencia de movimientos se configura mediante una tabla en un cuadro de dialogo de configuración.

El perfil de movimiento de la secuencia de movimientos puede comprobarse gráficamente incluso antes de cargar el proyecto en la CPU. A continuación, en el programa de usuario se pueden efectuar llamadas a las tablas de peticiones creadas a través de la instrucción de Motion Control "MC_CommandTable", relacionarlas con un eje y utilizarlas. La tabla de peticiones se puede procesar por completo o sólo parcialmente.

Herramientas del objeto tecnológico Tabla de peticiones

El portal TIA incorpora la herramienta "Configuración" para el objeto tecnológico "Tabla de peticiones". La siguiente representación muestra la interacción de la herramienta con el objeto tecnológico:



①	Escritura y lectura de la configuración del objeto tecnológico

Configuración

Configure las siguientes propiedades del objeto tecnológico "Tabla de peticiones" con la herramienta "Configuración":

- Cree una o varias secuencias de movimiento configurando peticiones individuales.
- Configure la representación gráfica para comprobar la secuencia de movimientos basándose en un eje ya configurado o en un eje de ejemplo configurable.

Los datos de la secuencia de movimientos se almacenan en el bloque de datos del objeto tecnológico.

Agregar el objeto tecnológico Tabla de peticiones

Requisitos

- Hay un proyecto creado con una CPU S7-1200.
- La versión de firmware de la CPU es la V2.1 o superior.

Procedimiento

Para agregar un objeto tecnológico "Tabla de peticiones" en el árbol del proyecto, proceda del siguiente modo:

1. Abra la carpeta "CPU > Objetos tecnológicos" en el árbol del proyecto.
2. Haga doble clic en el comando "Agregar objeto".
   Se abre el cuadro de diálogo "Agregar objeto".
3. Seleccione la tecnología "Motion Control".
4. Abra la carpeta "Motion Control > S7-1200 Motion Control".
5. Seleccione la versión deseada de la tecnología en la columna "Versión".
6. Seleccione el objeto "TO_CommandTable".
7. En el campo de entrada "Nombre" introduzca el nombre de la tabla de peticiones.
8. Para modificar los números asignados automáticamente al bloque de datos, seleccione la opción "manual".
9. Para visualizar y añadir más información sobre el objeto tecnológico, haga clic en "Más información".
10. Confirme la entrada realizada con "Aceptar".

Resultado

Se crea el nuevo objeto tecnológico y se guarda en la capeta "Objetos tecnológicos" del árbol del proyecto.

Configuración del objeto tecnológico Tabla de peticiones

Trabajando con el diálogo de configuración

Las propiedades del objeto tecnológico se configuran en la ventana de configuración. Para abrir la ventana de configuración del objeto tecnológico, proceda del siguiente modo:

1. Abra el grupo correspondiente al objeto tecnológico deseado en el árbol del proyecto.
2. Haga doble clic en el objeto "Configuración".

La configuración se divide en las siguientes categorías:

- **Parámetros básicos**
  Los parámetros básicos contienen todos los parámetros que deben configurarse para una tabla de peticiones con disponibilidad de funcionamiento.
- **Parámetros avanzados**
  Los parámetros avanzados contienen los parámetros del eje de ejemplo, o bien muestran los valores de los parámetros del eje elegido.

Símbolos de la ventana de configuración

Los símbolos que aparecen en la navegación local de la configuración muestran más detalles sobre el estado de la configuración:

	La configuración contiene valores predeterminados y está completa. La configuración contiene exclusivamente valores predeterminados. Con ellos es posible utilizar el objeto tecnológico sin más modificaciones.
	La configuración contiene valores definidos por el usuario y está completa Todos los campos de entrada de la configuración contienen valores válidos y al menos un valor predeterminado ha sido modificado.
	La configuración es errónea o está incompleta Al menos un campo de entrada o una lista desplegable contiene un valor no válido. El campo o la lista desplegable en cuestión se marca en rojo. Al hacer clic, el roll out con el aviso de error indica la causa del error.
	La configuración contiene valores de parámetros no compatibles entre sí La configuración contiene valores de parámetros contradictorios en lo que respecta a su magnitud o bien a su lógica. El campo o la lista desplegable en cuestión se marca en amarillo.

Consulte también

Guía para utilizar Motion Control (Página 10441)

Parámetros básicos (Página 10486)

Parámetros avanzados (Página 10500)

Comparar valores

La función "Comparar valores" ofrece las siguientes opciones:

- Comparación de los valores de arranque configurados del proyecto con los valores de arranque de la CPU y los valores actuales
- Procesamiento directo de los valores actuales y de los valores de arranque del proyecto
- Detección y visualización inmediatas de errores de introducción con propuestas de corrección
- Copia de seguridad de los valores actuales en el proyecto
- Transferencia de los valores de arranque del proyecto a la CPU como valores actuales

Símbolos y elementos de control

Están disponibles los siguientes símbolos y elementos de control:

Símbolo	Función
	El valor de arranque en la CPU es igual al valor de arranque configurado en el proyecto
	El valor de arranque en la CPU es diferente al valor de arranque configurado en el proyecto

Símbolo	Función
	No se puede realizar la comparación entre valor de arranque en la CPU y valor de arranque configurado en el proyecto
	Se guarda una copia de seguridad de los valores actuales en el proyecto
	Los valores de arranque actualizados en el proyecto se transfieren a la CPU (inicializar valores de ajuste)
	Se abre el cuadro de diálogo "Comparar valores"

Parámetros básicos

Configuración - General

Configure el nombre del objeto tecnológico en la ventana de configuración "General".

Nombre

Defina en este campo el nombre de la tabla de peticiones o el nombre del objeto tecnológico "Tabla de peticiones". El objeto tecnológico aparece con ese nombre en el árbol del proyecto.

Consulte también

Configuración - Tabla de peticiones (Página 10486)

Comandos del menú contextual - Tabla de peticiones (Página 10490)

Trabajando con el diagrama de curvas (Página 10491)

Comandos del menú contextual - Diagrama de curvas (Página 10495)

Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10496)

Modificar la configuración de la tabla de peticiones en el programa de usuario (Página 10498)

Configuración - Tabla de peticiones

Configure en la ventana de configuración "Tabla de peticiones" la secuencia de movimientos deseada y compruebe el resultado a través de la representación gráfica, en el diagrama de curvas.

Nota

Pueden producirse ligeras divergencias entre la representación del comportamiento temporal y la posición en las curvas en comparación con el movimiento real del eje. No se representan los procesos de movimiento que son reacción a la llegada a los finales de carrera de software.

Activar advertencias

Active con esta casilla de verificación la visualización de advertencias en la tabla de peticiones.

Utilizar parámetros del eje de

Seleccione en la lista desplegable los parámetros del eje que deben servir como base para la representación gráfica y para la comprobación de la secuencia de movimientos. Seleccione "Eje de ejemplo" si aún no ha insertado ningún eje en la carpeta "Objetos tecnológicos" o si desea utilizar valores que no se han configurado en ningún eje existente. Las propiedades del eje de ejemplo se configuran en el área "Parámetros avanzados".

Para procesar la tabla de peticiones en el programa de usuario se utilizan los parámetros del eje seleccionado en el parámetro "Axis".

Columna: Paso

Muestra el número de pasos de la petición.

Columna: Tipo de petición

Seleccione en esta columna los tipos de petición que se deben ejecutar al procesar la tabla de peticiones. Se pueden registrar hasta 32 peticiones. Las peticiones se procesan de forma secuencial. Efectúe su selección entre las entradas y tipos de peticiones siguientes:

- Empty
  La entrada hace las veces de comodín para hipotéticas peticiones que se deban insertar. La entrada vacía se ignora al procesar la tabla de peticiones.
- Halt
  Detener eje
  (la petición solo actúa tras una petición "Velocity set point")
- Positioning Relative
  Posicionar eje de forma relativa
- Positioning Absolute
  Posicionar eje de forma absoluta
- Velocity set point
  Mover eje con preajuste de velocidad
- Wait
  Espera hasta que el tiempo indicado haya transcurrido. Wait no detiene los desplazamientos que se están efectuando.
- Separator
  Inserta una fila Separator por encima de la fila seleccionada. La fila Separator actúa como límite de zona para la representación gráfica del diagrama de curvas.
  Utilice filas Separator si desea procesar partes de la tabla de peticiones.

Columna: Posición

Indique en esta columna la posición o el recorrido de desplazamiento para la petición seleccionada:

- **Petición "Positioning Relative"**
  La petición desplaza el eje en el recorrido indicado.
- **Petición "Positioning Absolute"**
  La petición desplaza el eje a la posición indicada.
- **Separator**
  El valor indicado corresponde a la posición inicial de la representación gráfica.

Límites (independientes de la unidad de usuario seleccionada):

- -1.0e12 ≤ posición / recorrido de desplazamiento ≤ -1.0e-12
- 1.0e-12 ≤ posición / recorrido de desplazamiento ≤ 1.0e12
- Posición/recorrido de desplazamiento = 0.0

Columna: Velocidad

Indique en esta columna la velocidad para la petición seleccionada:

- **Petición "Positioning Relative"**
  La petición desplaza el eje con la velocidad indicada.
  Si el recorrido de desplazamiento elegido no es suficientemente grande, no se alcanza la velocidad indicada.
- **Petición "Positioning Absolute"**
  La petición desplaza el eje con la velocidad indicada.
  Si la posición de destino está demasiado cerca de la posición de inicio, no se alcanza la velocidad indicada.
- **Petición " Velocity set point"**
  La petición desplaza el eje con la velocidad indicada.
  Si se elige un tiempo de ejecución demasiado corto, la velocidad indicada no se puede alcanzar a lo largo de la duración de la petición.

Límites (independientes de la unidad de usuario seleccionada):

- Para las peticiones: "Positioning Relative" y "Positioning Absolute"
  - 1.0e-12 ≤ velocidad ≤ 1.0e12
- Para la petición: "Velocity set point"
  - -1.0e12 ≤ velocidad ≤ -1.0e-12
  - 1.0e-12 ≤ velocidad ≤ 1.0e12
  - Velocidad = 0.0

Columna: Duración

Indique en esta columna la duración de la petición elegida:

- **Petición " Velocity set point"**
  La petición desplaza el eje durante la duración indicada. La duración incluye tanto la fase de aceleración como la fase de marcha constante. Una vez transcurrida la duración se pasa a la petición siguiente.
- **Petición "Wait"**
  Espera hasta que haya transcurrido la duración indicada.

Límites (independientes de la unidad de usuario seleccionada):

- 0.001s ≤ duración ≤ 64800 s

Columna: Siguiente paso

Elija en la lista desplegable el modo de transición al paso siguiente:

- **Finalizar petición**
  La petición finaliza. Se pasa a la petición siguiente sin ningún hueco temporal.
- **Suavizar transición del movimiento**
  El movimiento de la petición actual se transforma suavemente en el movimiento de la petición siguiente. El modo de transición "Suavizar transición del movimiento" se puede seleccionar en los tipos de petición "Positioning Relative" y "Positioning Absolute".
  El movimiento se transforma suavemente en el movimiento de los siguientes tipos de petición:
  - Positioning Relative
  - Positioning Absolute
  - Velocity set point

Con otros tipos de petición no se transforma suavemente.

Para conocer el comportamiento exacto del eje al añadir o suavizar la transición de una petición consulte el capítulo: Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10496)

Columna: Código de paso

Indique en esta columna el valor numérico/patrón de bits que se debe emitir por el parámetro de salida "StepCode" de la instrucción de Motion Control "MC_CommandTable" durante el procesamiento de la petición.

Límites:

- 0 ≤ clave ≤ 65535

Consulte también

Configuración - General (Página 10483)

Comandos del menú contextual - Tabla de peticiones (Página 10490)

Trabajando con el diagrama de curvas (Página 10491)

Comandos del menú contextual - Diagrama de curvas (Página 10495)

Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10496)

Modificar la configuración de la tabla de peticiones en el programa de usuario (Página 10498)

Comandos del menú contextual - Tabla de peticiones

La tabla de peticiones ofrece los comandos de menú contextual siguientes:

Insertar fila Empty

Inserta una fila vacía encima de la fila seleccionada.

El comando del menú contextual únicamente se puede ejecutar si al final de la tabla de peticiones se dispone de suficientes filas vacías.

Agregar fila Empty

Inserta una fila vacía debajo de la fila seleccionada.

El comando del menú contextual únicamente se puede ejecutar si al final de la tabla de peticiones se dispone de suficientes filas vacías.

Insertar fila de separación

Inserta una fila de separación encima de la fila seleccionada.

No es posible insertar dos filas de separación consecutivas.

Agregar fila de separación

Inserta una fila de separación debajo de la fila seleccionada.

No es posible insertar dos filas de separación consecutivas; tampoco se puede insertar una fila de separación al final de la tabla de peticiones.

Cortar

Elimina las filas seleccionadas o el contenido de las celdas seleccionadas y las coloca en el portapapeles.

Las filas seleccionadas se eliminan; las filas siguientes de la tabla de peticiones se desplazan hacia arriba.

Copiar

Copia las filas seleccionadas o el contenido de las celdas seleccionadas y las coloca en el portapapeles.

Pegar

- Filas seleccionadas:
  Inserta las filas del portapapeles encima de la fila seleccionada.
- Celda seleccionada:
  Inserta el contenido del portapapeles en la fila seleccionada.

El comando del menú contextual únicamente se puede ejecutar si al final de la tabla de peticiones se dispone de suficientes filas vacías.

Reemplazar

Sustituye las filas seleccionadas con las filas del portapapeles.

Borrar

Elimina las filas seleccionadas. Las filas siguientes de la tabla de peticiones se desplazan hacia arriba.

Consulte también

Configuración - General (Página 10483)

Configuración - Tabla de peticiones (Página 10483)

Trabajando con el diagrama de curvas (Página 10491)

Comandos del menú contextual - Diagrama de curvas (Página 10495)

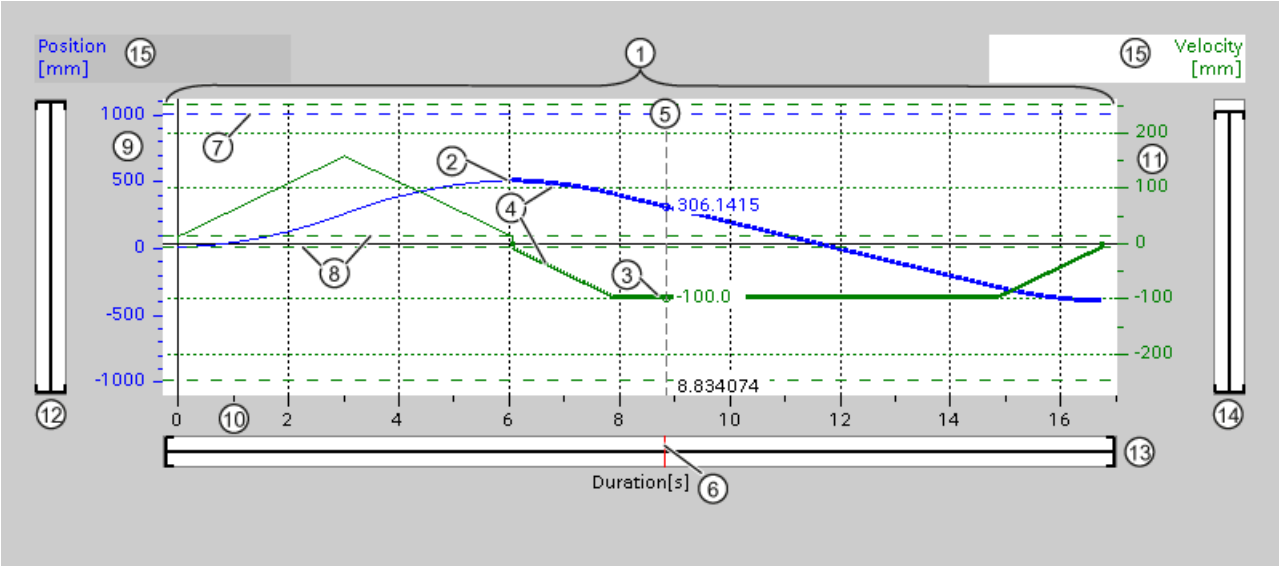
Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10496)

Modificar la configuración de la tabla de peticiones en el programa de usuario (Página 10498)

Trabajando con el diagrama de curvas

El diagrama de curvas incorpora las herramientas e informaciones siguientes:

Vista y componentes del diagrama de curvas



①	Ventana de curvas
②	Curva de posición
③	Curva de velocidad
④	Sección de curva de una petición seleccionada
⑤	Regla de medición
⑥	Marca de posición de la regla de medición
⑦	Posición del final de carrera por SW
⑧	Velocidad de arranque/parada
⑨	Área de escala del eje de posición
⑩	Área de escala del eje de tiempo
⑪	Área de escala del eje de velocidad
⑫	Área de la barra de desplazamiento del eje de posición
⑬	Área de la barra de desplazamiento del eje de tiempo
⑭	Área de la barra de desplazamiento del eje de velocidad
⑮	Elección de la cuadrícula

Selección de las secciones de separación

Si la tabla de peticiones está compuesta por varias secciones divididas por separadores, las secciones del diagrama de curvas también se pueden seleccionar mediante selección de una petición perteneciente a la sección correspondiente.

Seleccionar peticiones

Las peticiones se pueden seleccionar en el diagrama de curvas y en la tabla de peticiones:

- En el diagrama de curvas, haga clic en un punto de la curva de velocidad o de la curva de posición. La petición correspondiente queda seleccionada en la tabla de peticiones.
- Seleccione una petición en la tabla de peticiones.
  Las zonas correspondientes de la curva se destacan en el área de curva.

Seleccionar el área visible del diagrama de curvas

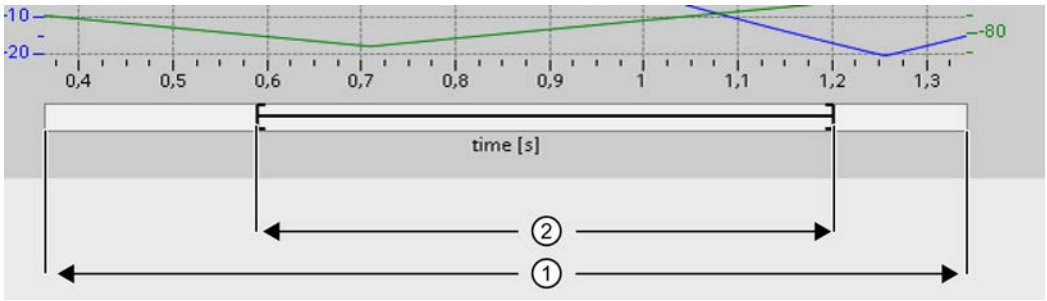
Para adaptar la sección del diagrama de curvas que se desea representar, proceda como se describe a continuación:

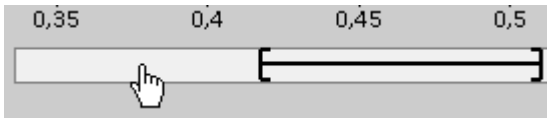
Selección de la escala en el menú contextual:

- Escala de curvas:
  Escala los ejes de manera que la curva de posición y la curva de velocidad sean visibles.
- Escala de curvas y límites:
  Escala los ejes de manera que tanto la curva de posición y la curva de velocidad, como las posiciones finales de carrera por software activadas y los límites inferior y superior de velocidad sean visibles.

La vista actualmente seleccionada se identifica en el menú contextual con una marca de verificación.

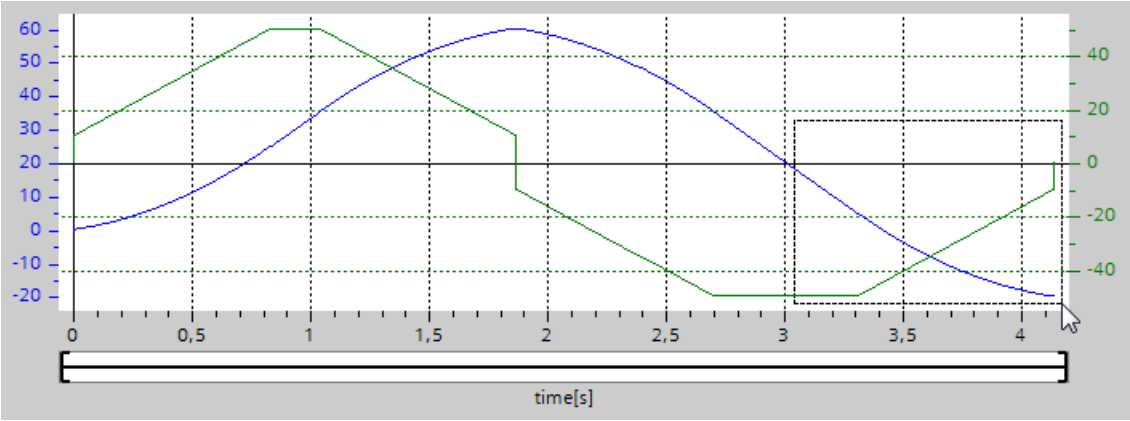
Seleccionar en el área de escala la sección que se desea representar:



①	Área en la que se encuentran los valores de la curva y/o los valores límite. (véase la selección en el menú contextual)
②	Área seleccionada que se representa en la ventana de curva. El área seleccionada se ajusta mediante el borde derecho y el borde izquierdo con el cursor redondo. La posición dentro del área ① se ajusta con el cursor de desplazamiento. La posición también se puede determinar haciendo clic en el área ①. 

Elegir con el ratón la sección que se desea representar:

Haga clic en una sección con el ratón y arrástrela hasta el diagrama de curvas. Tras soltar el ratón, la representación del área de curva se amplía a la selección.



Deshacer el último cambio de la sección:

Para deshacer el último cambio de la sección elija el comando del menú contextual "Deshacer zoom".

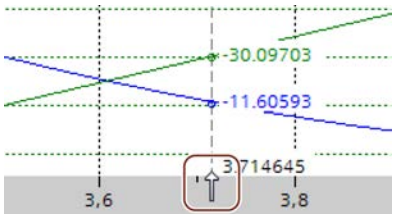
Sincronizar la cuadrícula

Haciendo clic en las escalas de eje, elija si la cuadrícula se debe sincronizar con el eje de posición o con el eje de velocidad.

Leer los valores de la curva en la regla de medición

Active la regla de medición con el comando de menú contextual "Insertar regla de medición".

Con el cursor de la regla de medición, desplace ésta hasta los puntos que desee de las curvas.



Consulte también

Configuración - General (Página 10483)

Configuración - Tabla de peticiones (Página 10483)

Comandos del menú contextual - Tabla de peticiones (Página 10487)

Comandos del menú contextual - Diagrama de curvas (Página 10495)

Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10496)

Modificar la configuración de la tabla de peticiones en el programa de usuario (Página 10498)

Comandos del menú contextual - Diagrama de curvas

El área de la ventana de curvas incorpora los comandos de menú contextual siguientes:

Zoom 100%

Se elige un factor de zoom con el que se puede ver el 100% de los valores de las curvas y/o los valores límite.

Deshacer el zoom

Anula la última modificación de zoom.

Escala de curvas

Escala los ejes de manera que la curva de posición y la curva de velocidad sean visibles.

Escala de curvas y límites

Escala los ejes de manera que tanto la curva de posición y la curva de velocidad, como las posiciones finales de carrera por software activadas y los límites inferior y superior de velocidad sean visibles.

Insertar límites de velocidad

Inserta las líneas de los límites de velocidad.

Insertar fines de carrera por software

Inserta las líneas de los fines de carrera por software.

Insertar regla de medición

Muestra u oculta la regla de medición.

Utilice la regla de medición si desea observar valores individuales de las curvas.

Consulte también

Configuración - General (Página 10483)

Configuración - Tabla de peticiones (Página 10483)

Comandos del menú contextual - Tabla de peticiones (Página 10487)

Trabajando con el diagrama de curvas (Página 10488)

Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10496)

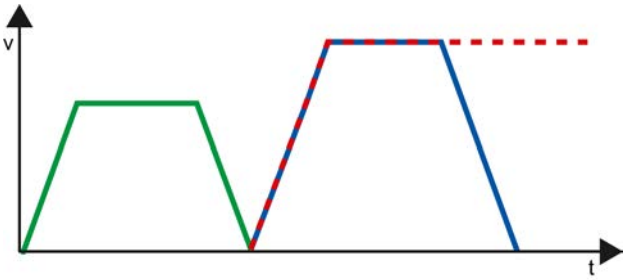
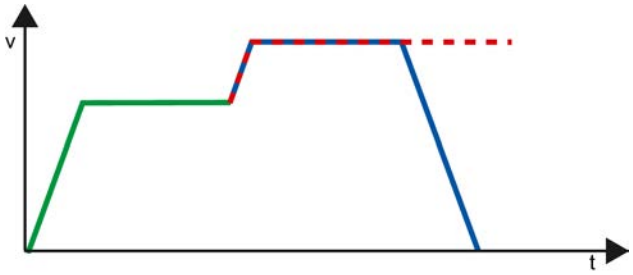
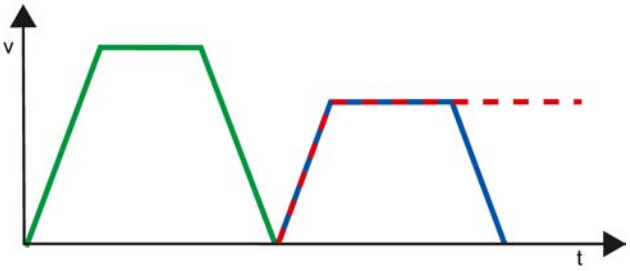
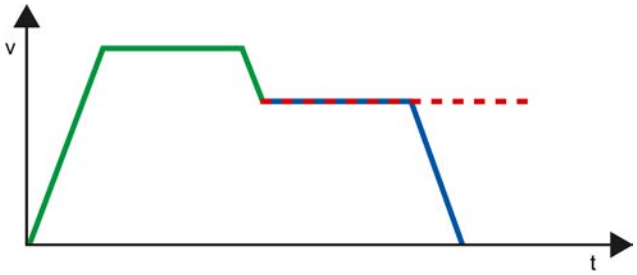
Modificar la configuración de la tabla de peticiones en el programa de usuario (Página 10498)

Transición de petición "Finalizar petición"/"Suavizar transición del movimiento"

Los diagramas siguientes muestran la transición del movimiento para distintas selecciones del modo de transición en la columna "Siguiente paso":

Transición del movimiento si la petición precedente es una petición de posicionamiento

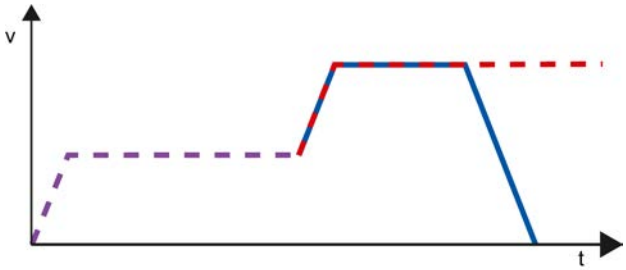
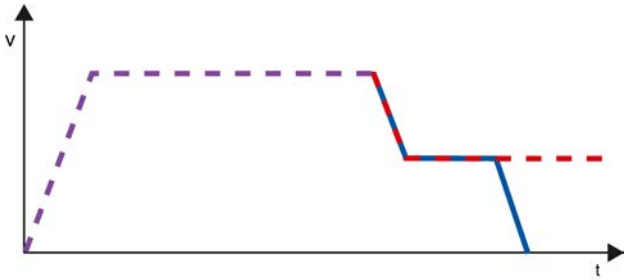
Los siguientes diagramas representan una secuencia de peticiones con dos peticiones de movimiento. La primera es una petición de posicionamiento (verde). La segunda es una petición de velocidad (roja) o de posicionamiento (azul):

Finalizar petición	Suavizar transición del movimiento
Transición de velocidad inferior a velocidad superior  Una petición con una velocidad mayor se añade a una petición de posicionamiento anterior. La primera petición de posicionamiento finaliza en su posición de destino con la velocidad "0". La segunda petición comienza a partir de la posición de parada.	**Transición de velocidad inferior a velocidad superior**  Una petición con una velocidad mayor se suaviza con una petición de posicionamiento anterior. La primera petición de posicionamiento finaliza sin parada en la posición de destino. La segunda petición se suaviza a la nueva velocidad.
Transición de velocidad superior a velocidad inferior  Una petición con una velocidad menor se añade a una petición de posicionamiento anterior. La primera petición de posicionamiento finaliza en su posición de destino con la velocidad "0". La segunda petición comienza a partir de la posición de parada.	**Transición de velocidad superior a velocidad inferior**  Una petición con una velocidad inferior se suaviza con una petición de posicionamiento anterior. La primera petición de posicionamiento finaliza sin parada en la posición de destino. La primera petición se suaviza a la nueva velocidad.

———	1. Petición "Positioning Relative" o "Positioning Absolute"
- - -	2. Petición "Velocity set point"
———	2. Petición "Positioning Relative" o "Positioning Absolute"

Transición del movimiento si la petición precedente es una petición de velocidad

Los siguientes diagramas representan una secuencia de peticiones con dos peticiones de movimiento. La primera es una petición de velocidad (violeta). La segunda es una petición de velocidad (roja) o de posicionamiento (azul):

Transición de velocidad inferior a velocidad superior	Transición de velocidad superior a velocidad inferior
	
Una petición con una velocidad mayor se añade a una petición de velocidad anterior. La primera petición de velocidad finaliza después del tiempo indicado. La segunda petición se suaviza a la nueva velocidad.	Una petición con una velocidad inferior se suaviza con una petición de velocidad anterior. La segunda petición se suaviza a la nueva velocidad.

	1. Petición "Velocity set point"
	2. Petición "Velocity set point"
	2. Petición "Positioning Relative" o "Positioning Absolute"

Consulte también

Configuración - General (Página 10483)

Configuración - Tabla de peticiones (Página 10483)

Comandos del menú contextual - Tabla de peticiones (Página 10487)

Trabajando con el diagrama de curvas (Página 10488)

Comandos del menú contextual - Diagrama de curvas (Página 10492)

Modificar la configuración de la tabla de peticiones en el programa de usuario (Página 10498)

Modificar la configuración de la tabla de peticiones en el programa de usuario

Los siguientes parámetros de configuración pueden modificarse en la CPU mientras se ejecuta el programa de usuario:

Peticiones y valores correspondientes

Los parámetros de la tabla de peticiones se pueden modificar incluso durante el tiempo de ejecución del programa de usuario. Para ello utilice las siguientes variables del objeto tecnológico:

- <Nombre de la tabla>.Command[1..32].Type
  para modificar el tipo de petición correspondiente
- <Nombre de la tabla>.Command[1..32].Position
  para modificar la posición o el recorrido
- <Nombre de la tabla>.Command[1..32].Velocity
  para modificar la velocidad
- <Nombre de la tabla>.Command[1..32].Duration
  para modificar la duración
- <Nombre de la tabla>.Command[1..32].NextStep
  para modificar el parámetro "Siguiente paso"
- <Nombre de la tabla>.Command[1..32].StepCode
  para modificar el código de paso

Para saber cuándo tienen efecto las modificaciones de los parámetros de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico (Página 10592).

Consulte también

Lista de compatibilidad de las variables (Página 10445)

Configuración - General (Página 10483)

Configuración - Tabla de peticiones (Página 10483)

Comandos del menú contextual - Tabla de peticiones (Página 10487)

Trabajando con el diagrama de curvas (Página 10488)

Comandos del menú contextual - Diagrama de curvas (Página 10492)

Transición de petición "Finalizar petición"/"Suavizar transición del movimiento" (Página 10493)

Parámetros avanzados

Parámetros de diagrama

Configuración - General

Configure las propiedades básicas de la representación de diagrama del objeto tecnológico "Tabla de peticiones" en la ventana de configuración "General".

Nota

Si en el área "Utilizar parámetros de eje de" se ha seleccionado un eje de ejemplo, la unidad de medida se puede editar. Si se ha elegido un eje configurado, se muestra el valor del eje.

Utilizar parámetros del eje de

Seleccione en la lista desplegable los parámetros del eje que deben servir como base para la representación gráfica y para la comprobación de la secuencia de movimientos. Seleccione "Eje de ejemplo" si aún no ha insertado ningún eje en la carpeta "Objetos tecnológicos" o si desea utilizar valores que no se han configurado en ningún eje existente.

Los parámetros de eje del eje seleccionado en el parámetro del mismo nombre se utilizan para el procesamiento de la tabla de peticiones en el programa de usuario.

Unidad de medida

Introduzca en este campo la unidad de medida para el eje de ejemplo. Si en el área "Utilizar parámetros de eje de" se ha seleccionado un eje ya configurado, se muestra la unidad de medida configurada allí.

Configuración - Dinámica

En la ventana de configuración "Dinámica" configure la aceleración y la deceleración, así como la limitación de tirones del eje de ejemplo.

Nota

Si en el área "Utilizar parámetros de eje de" se ha seleccionado un eje de ejemplo, se pueden editar los campos descritos a continuación. Si se ha elegido un eje configurado, se muestran los valores del eje.

Aceleración/deceleración

Ajuste la aceleración deseada para el eje de ejemplo en el campo "Aceleración". La deceleración deseada se puede ajustar en el campo "Deceleración".

Las peticiones de movimiento configuradas en la tabla de peticiones se calculan con la aceleración / deceleración seleccionada.

Límites:

- 1,0e-12 ≤ aceleración ≤ 1,0e12
- 1,0e-12 ≤ deceleración ≤ 1,0e12

Activar limitación de tirones

Active la limitación de tirones con esta casilla de verificación.

Tirón

Ajuste el tirón deseado para la rampa de aceleración y deceleración en el campo "Tirón".

Las peticiones de movimiento configuradas en la tabla de peticiones se calculan con el tirón seleccionado.

Límites:

- 1,0e-12 ≤ tirón ≤ 1,0e12

Configuración - Valores límite

En la ventana de configuración "Valores límite", configure la velocidad máxima, la velocidad de arranque/parada y los finales de carrera por SW del eje de ejemplo.

Nota

Si en el área "Utilizar parámetros de eje de" se ha seleccionado un eje de ejemplo, se pueden editar los campos descritos a continuación. Si se ha elegido un eje configurado, se muestran los valores del eje.

Velocidad máxima / Velocidad de arranque/parada

Defina en estos campos la velocidad máxima admisible y la velocidad de arranque/parada del eje de ejemplo. La velocidad de arranque/parada es la velocidad mínima admisible del eje de ejemplo.

Límites:

- 1,0e-12 ≤ velocidad de arranque/parada ≤ 1,0e12
  Velocidad de arranque/parada = 0.0
- 1,0e-12 ≤ velocidad máxima ≤ 1,0e12
  Velocidad máxima = 0.0

El valor de la velocidad máxima debe ser mayor o igual al valor de la velocidad de arranque/parada.

Activar final de carrera de software

Active con esta casilla de verificación la función de los finales de carrera por software inferior y superior. En el diagrama de curvas no se representan los procesos de movimiento que son reacción a la llegada a los finales de carrera de software.

Final de carrera por software inferior/superior

En estos campos se especifica el valor de posición de los finales de carrera por software inferior y superior.

Límites:

- -1,0e12 ≤ final de carrera por software inferior ≤ -1,0e12
  1,0e12 ≤ final de carrera por software inferior ≤ 1,0e12
  Final de carrera por software inferior = 0,0
- -1,0e12 ≤ final de carrera por software superior ≤ -1,0e12
  1,0e12 ≤ final de carrera por software superior ≤ 1,0e12
  Final de carrera por software superior = 0,0

El valor del final de carrera por software superior debe ser mayor o igual al valor del final de carrera por software inferior.

Cargar en la CPU

Con la carga en la CPU S7-1200 se asegura siempre que los datos de proyecto sean coherentes tanto online como offline. No es posible cargar módulos individuales. Al seleccionar bloques individuales, se cargan siempre todos los módulos nuevos y modificados.

Los siguientes grupos de objetos pueden cargarse en la CPU:

Comando del menú contextual "Cargar en dispositivo"	Descripción
Todo	Cargar todos los bloques nuevos y modificados y una configuración hardware nueva o modificada
Configuración hardware	Cargar una configuración hardware nueva o modificada
Software	Cargar todos los bloques nuevos y modificados
Software (todos los bloques)	Cargar todos los bloques

Los datos de los objetos tecnológicos Motion Control se guardan en bloques de datos. Para cargar un objeto tecnológico nuevo o modificado rigen por lo tanto las condiciones para la carga de "Bloques"

⚠ PRECAUCIÓN

Posibles fallos en el funcionamiento del eje al cargar sin configuración hardware

La configuración hardware cambia cuando se realizan los siguientes cambios en la configuración del eje:

- Modificación del generador de impulsos (PTO)
- Modificación de la dirección del final de carrera por HW
- Modificación de la dirección del sensor del punto de referencia

Si la configuración modificada del eje se carga mediante los comandos del menú contextual "Software" o "Software (todos los bloques)" sin cargar la configuración hardware, puede producirse un mal funcionamiento del eje.

Asegúrese de cargar la configuración hardware actual en la CPU bajo las condiciones mencionadas.

Carga en el estado operativo RUN de la CPU S7-1200 (a partir de la versión de firmware V2.2)

A partir de la versión de firmware V2.2 de la CPU S7-1200, al realizar una carga en el estado operativo RUN se comprueba si es posible cargar sin conmutar la CPU a STOP.

Al cargar los bloques de datos en el estado operativo RUN rigen las condiciones siguientes:

	Carga en la memoria de carga	Carga en la memoria de trabajo
Valores modificados del bloque de datos	SÍ	No
Estructura modificada del bloque de datos	Sí (a partir de la versión de firmware V4)	Sí (a partir de la versión de firmware V4) • Durante la carga con reinicialización • Para variables en la reserva al cargar sin reinicialización
	No (firmware V2.2...3)	No (firmware V2.2...3)
Bloque de datos nuevo	SÍ	SÍ
Bloque de datos borrado	SÍ	SÍ

Tenga en cuenta también lo siguiente al borrar y cargar con reinicialización de bloques de datos:

- Al cargar un objeto tecnológico Eje de posicionamiento, el eje debe estar bloqueado.
- Al cargar un objeto tecnológico Tabla de peticiones no debe haber ninguna petición MC_CommandTable activa con esta tabla de peticiones (parámetro "Busy" = FALSE).
- Al cargar un bloque de datos de instancia MC_Power, no debe estar activa la correspondiente instrucción MC_Power (parámetro "Busy" = FALSE).

A partir de la versión tecnológica V3.0, los objetos tecnológicos Motion Control (bloques de datos) también pueden cargarse con la CPU en el estado operativo RUN.

Los objetos tecnológicos de versión inferior a V3.0 no pueden cargarse con la CPU en el estado operativo RUN.

Seleccione una de las acciones descritas a continuación para cargar en la memoria de trabajo la configuración modificada de un objeto tecnológico Motion Control (a partir de la versión V3.0):

- Objeto tecnológico Eje y tabla de peticiones
  Cambie al estado operativo de la CPU de STOP a RUN.
- Objeto tecnológico Eje
  Bloquee el eje y ejecute con la instrucción Motion Control "MC_Reset" un "Restart".
- Objeto tecnológico Tabla de peticiones
  Asegúrese de que la tabla de peticiones no se utiliza. Cargue el bloque de datos de la tabla de peticiones en la memoria de trabajo con la instrucción avanzada "READ_DBL".

Consulte también

Guía para utilizar Motion Control (Página 10441)

MC_Reset: Acusar errores V4 (Página 5023)

Puesta en servicio del eje - Panel de control del eje

Utilice el panel de mando del eje para mover el eje en modo manual, optimizar los ajustes del eje y realizar un test de su instalación.

El panel de mando del eje sólo puede manejarse si hay establecida una conexión online con la CPU.

Nota

Tiempos de reacción del panel de mando del eje

El tiempo de reacción durante el manejo del panel de mando del eje depende de la carga de comunicación de la CPU. Cierre las demás ventanas online del portal TIA para disminuir el tiempo de reacción.

Botón "Control manual"

Haga clic en el botón "Control manual" para mover el eje en control manual. Retire antes la habilitación del eje a través de la instrucción de Motion Control "MC_Power" en el programa de usuario. En el modo de operación "Control manual" el panel de mando tiene la máxima

autoridad de control de las funciones del eje. El programa de usuario no tiene ninguna influencia en las funciones del eje hasta que no finalice el control manual.

⚠ PRECAUCIÓN
Ejes adicionales en el modo automático El control manual sólo está activo para un eje. Si se encuentran ejes adicionales en el modo automático, pueden producirse situaciones peligrosas. En ese caso, bloquee todos los ejes adicionales.

Botón "Modo automático"

Haga clic en el botón "Modo automático" para finalizar el modo de operación "Control manual". El panel de mando del eje devuelve la autoridad de control y el eje puede ser controlado por el programa de usuario. En el programa de usuario se debe habilitar nuevamente el eje y, si fuera necesario, volver a referenciar.

Antes de pasar al modo automático finalice todos los movimientos en curso, de lo contrario el eje se frenará con la deceleración de parada de emergencia.

Botón "Habilitar"

Haga clic en el botón "Habilitar" para habilitar el eje en el modo de operación "Control manual". Una vez habilitado se pueden utilizar las funciones del panel de mando del eje.

Si el eje no puede ser habilitado por no cumplir los requisitos, consulte el aviso de error emitido en el campo "Aviso de error". En el anexo encontrará informaciones que le permitirán eliminar el error, en el apartado "Lista de ErrorIDs y ErrorInfos" Habilite nuevamente el eje una vez subsanado el error.

Botón "Bloquear"

Haga clic en el botón "Bloquear" para bloquear el eje temporalmente en el modo de operación "Control manual".

Área "Comando"

El manejo en el área "Comando" sólo es posible con el eje habilitado. Seleccione uno de los siguientes comandos:

- JOG
  El comando se corresponde con la petición de Motion Control "MC_MoveJog" del programa de usuario.
- Posicionamiento
  El comando se corresponde con las peticiones de Motion Control "MC_MoveAbsolute" y "MC_MoveRelative" del programa de usuario. Para realizar un posicionamiento absoluto es necesario que el eje esté referenciado.
- Referenciación
  El comando se corresponde con la petición de Motion Control "MC_Home" del programa de usuario.
  - El botón "Fijar punto de referencia" representa Mode = 0 (referenciación directa absoluta)
  - El botón "Referenciar - Activa" representa Mode = 3 (referenciación activa)

  Para la referenciación activa se debe haber configurado el sensor del punto de referencia en la configuración del eje.
  Los valores para las velocidades de aproximación y referenciación y para el offset del punto de referencia se toman, sin modificaciones, de la configuración del eje.

Dependiendo de la selección aparecen campos relevantes para la entrada de los valores de consigna y botones para iniciar el comando.

Con el botón "Activar limitación de tirones" se puede activar y desactivar la limitación de tirones. De forma predeterminada, se adopta el tirón con el 10 % del valor configurado. Este valor se puede modificar según sea necesario.

Área "Estado del eje"

Si el modo de operación "Control manual" se encuentra activado, en el área "Estado del eje" se muestra el estado actual del eje y del accionamiento. La posición y velocidad actuales del eje se indican en "Valores actuales".

Una vez solucionados, acuse los errores pendientes haciendo clic en el botón "Acusar".

El campo "Mensaje de información" muestra información detallada sobre el estado del eje.

Área "Valores actuales"

En esta área se muestra la posición y la velocidad actuales del eje.

Aviso de error

El campo "Mensaje de error" muestra el error actual. Una vez solucionada la causa del error, en el modo de operación "Control manual" se puede borrar la entrada correspondiente haciendo clic en el botón "Acusar".

Nota

Valores iniciales de velocidad, aceleración / deceleración y tirón

Por razones de seguridad, al activar el panel de mando del eje, los parámetros "Velocidad" y "Aceleración / Deceleración" y "Tirón" se inicializan únicamente con el 10% de los valores ajustados en la configuración. El parámetro "Tirón" se utiliza a partir del objeto tecnológico "Eje" V2.0.

Para la inicialización se toman como base los valores de la vista de configuración "Parámetros avanzados > Dinámica > General".

El parámetro "Velocidad" del panel de control del eje se deriva de la "Velocidad máxima" de la configuración, y el parámetro "Aceleración/deceleración" se deriva de la "Aceleración" de la configuración.

Los parámetros "Velocidad", "Aceleración / Deceleración" y "Tirón" pueden modificarse en el panel de mando; los valores de la configuración no se ven afectados por ello.

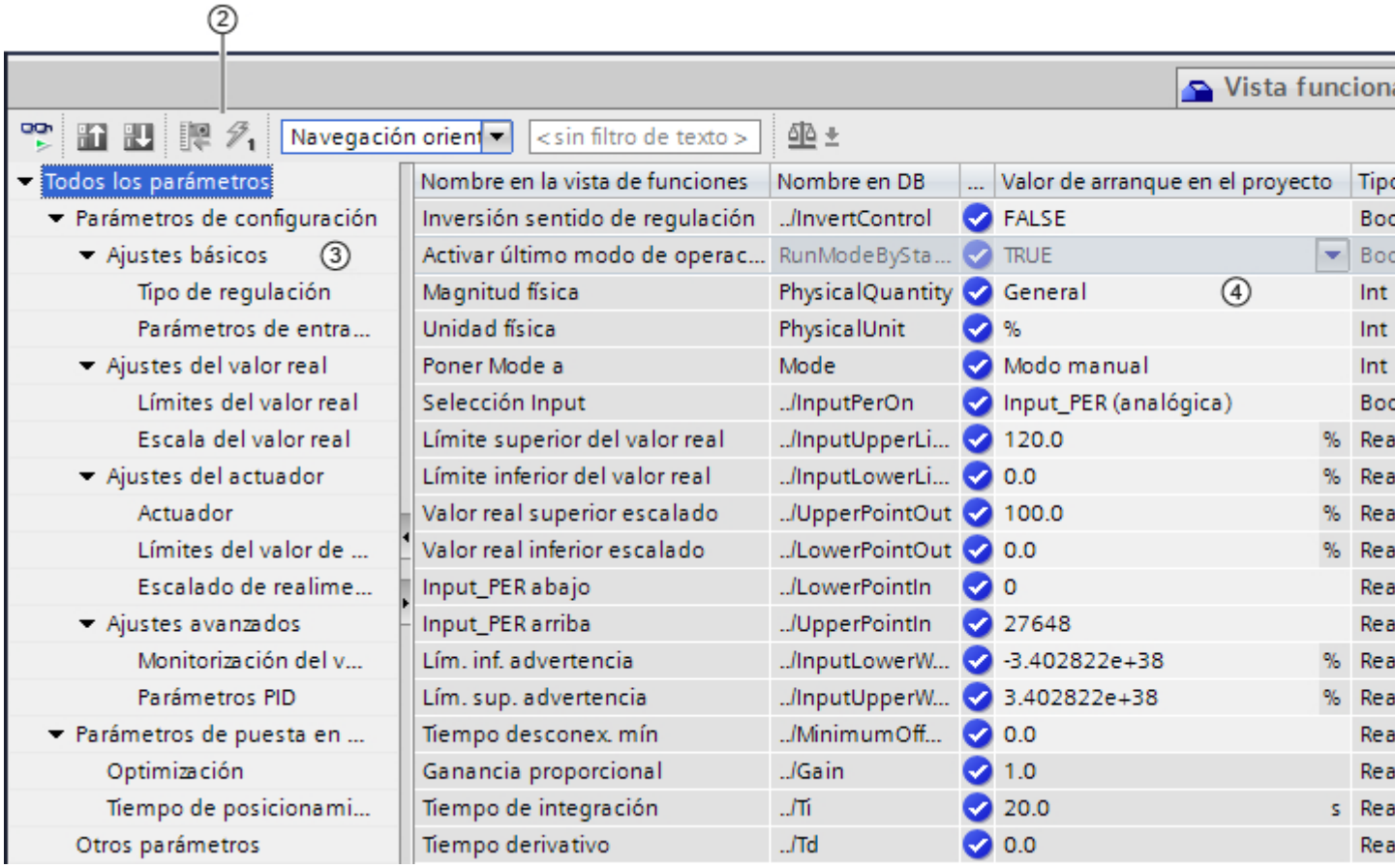
Consulte también

Guía para utilizar Motion Control (Página 10441)

Vista de parámetros

Introducción a la vista de parámetros

La vista de parámetros le ofrece una visión de conjunto de todos los parámetros relevantes de un objeto tecnológico. Proporciona una vista general de los ajustes de los parámetros, de manera que usted puede modificarlos cómodamente en el modo offline y online.



① Ficha "Vista de parámetros"

② Barra de herramientas (Página 9945)

③ Navegación (Página 9946)

④ Tabla de parámetros (Página 9946)

Funcionalidad

Para analizar los parámetros de los objetos tecnológicos, y para poderlos observar y forzar de manera selectiva, hay disponibles las siguientes funciones.

Funciones de indicación:

- Indicación de los valores de parámetros en modo offline y online
- Indicación de la información de estado de los parámetros

- Indicación de desviaciones de los valores y posibilidad de corregirlas directamente
- Indicación de errores de configuración
- Indicación de variaciones de valores como consecuencia de dependencias de parámetros
- Indicación de todos los valores almacenados de un parámetro: valor de arranque en la CPU, valor de arranque en el proyecto, valor de observación
- Indicación de la comparación de parámetros de los valores almacenados de un parámetro

Funciones de manejo:

- Navegación, para cambiar rápidamente entre los parámetros y las estructuras de parámetros.
- Filtro de texto, para encontrar más rápidamente determinados parámetros.
- Función de ordenación, para adaptar el orden de parámetros y grupos de parámetros en función de las necesidades.
- Función de almacenamiento, para guardar ajustes estructurales de la vista de parámetros.
- Observar y forzar online valores de parámetro.
- Guardar instantánea de valores de parámetros de la CPU para reproducir situaciones transitorias y reaccionar a ellas.
- Aplicar instantánea de valores de parámetros como valores de arranque.
- Cargar en la CPU valores de arranque modificados.
- Funciones de comparación, para comparar entre sí valores de parámetros.

Validez

La vista de parámetros está disponible para los siguientes objetos tecnológicos:

- PID_Compact
- PID_3Step
- CONT_C (solo S7-1500)
- CONT_S (solo S7-1500)
- TCONT_CP (solo S7-1500)
- TCONT_S (solo S7-1500)
- TO_Axis_PTO (S7-1200 Motion Control)
- TO_Positioning_Axis (S7-1200 Motion Control)
- TO_CommandTable_PTO (S7-1200 Motion Control)
- TO_CommandTable (S7-1200 Motion Control)

Estructura de la vista de parámetros

Barra de herramientas

En la barra de herramientas de la vista de parámetros pueden seleccionarse las siguientes funciones:

Símbolo	Función	Explicación
	Observar todos	Inicia la observación de los parámetros visibles en la vista de parámetros activa (modo online).
	Crear instantánea de los valores de observación y aplicar valores de ajuste de esa instantánea como valores de arranque	Aplica los valores de observación actuales a la columna "Instantánea" y actualiza los valores de arranque en el proyecto. Solo en modo online con PID_Compact y PID_3Step.
	Inicializar valores de ajuste	Transfiere a la CPU los valores de arranque actualizados en el proyecto. Solo en modo online con PID_Compact y PID_3Step.
	Crear instantánea de los valores de observación	Aplica los valores de observación actuales a la columna "Instantánea". Solo en modo online.
	Forzar una vez e inmediatamente todos los parámetros seleccionados	Este comando se ejecuta una sola vez y lo antes posible, sin referencia a un punto determinado del programa de usuario. Solo en modo online.
Navegación orient	Seleccionar la estructura de navegación	Cambia entre la navegación orientada a funciones y la navegación orientada a datos.
Navegación orient	Filtro de texto...	Tras introducir una cadena de caracteres: indicación de todos los parámetros que contienen la cadena de caracteres introducida en una de las columnas visibles actualmente.
	Seleccionar valores de comparación	Selección de los valores de parámetros que deben compararse entre sí en modo online (valor de arranque en el proyecto, valor de arranque en la CPU, instantánea) Solo en modo online.
	Memorizar disposición	Guarda los ajustes de visualización de la vista de parámetros realizados (p. ej., estructura de navegación seleccionada, columnas activadas de la tabla, etc.).

Navegación

Dentro de la ficha "Vista de parámetros" pueden seleccionarse como alternativa las siguientes estructuras de navegación:

Navegación		Explicación
Navegación orientada a funciones	Todos los parámetros Parámetros de configuración Parámetros de puesta en servicio Otros parámetros	En la navegación orientada a funciones, la estructura de los parámetros se basa en la estructura de los cuadros de diálogo de configuración (ficha "Vista de funciones"), puesta en marcha y diagnóstico. El último grupo "Otros parámetros" contiene los restantes parámetros del objeto tecnológico.
Navegación orientada a datos	Todos los parámetros Input Output InOut Static Otros parámetros	En la navegación orientada a datos, la estructura de los parámetros se basa en la estructura del DB de instancia. El último grupo "Otros parámetros" contiene los parámetros que no están incluidos en el DB de instancia.

Con la lista desplegable "Seleccionar la estructura de navegación" puede cambiar la estructura de navegación.

Tabla de parámetros

La tabla siguiente muestra el significado de las distintas columnas de la tabla de parámetros. Las columnas pueden mostrarse u ocultarse según convenga.

- Columna "Offline" = X: la columna es visible en modo offline.
- Columna "Online" = X: la columna es visible en modo online (conexión online con la CPU).

Columna	Explicación	Offline	Online
Nombre en la vista de funciones	Nombre del parámetro en la vista de funciones. Este campo de visualización está vacío en el caso de parámetros que no se configuran por medio del objeto tecnológico.	X	X
Nombre completo en DB	Ruta completa del parámetro en el DB de instancia. Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Nombre en DB	Nombre del parámetro en el DB de instancia. Si el parámetro es parte de una estructura o UDT, se añade el prefijo ". ./". Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Estado de la configuración	Indicación de la integridad de la configuración mediante símbolos de estado ver Estado de la configuración (offline) (Página 9954)	X	
Resultado de la comparación	Resultado de la función "Comparar valores". Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. ver Comparar valores (Página 9940)		X

Columna	Explicación	Offline	Online
Valor de arranque en el proyecto	Valor de arranque configurado en el proyecto. Indicación de errores en caso de valores mal introducidos desde el punto de vista sintáctico o tecnológico.	X	X
Valor predeterminado	Valor predeterminado del parámetro. Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Instantánea	Instantánea de los valores actuales en la CPU (valores de observación). Indicación de errores en caso de valores incorrectos desde el punto de vista tecnológico.	X	X
Valor de arranque en la CPU	Valor de arranque en la CPU. Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. Indicación de errores en caso de valores incorrectos desde el punto de vista tecnológico.		X
Valor de observación	Valor actual en la CPU. Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. Indicación de errores en caso de valores incorrectos desde el punto de vista tecnológico.		X
Valor de forzado	Valor con que debe modificarse el valor de observación. Esta columna aparece si existe una conexión online y el botón "Observar todos" está seleccionado. Indicación de errores en caso de valores mal introducidos desde el punto de vista sintáctico o tecnológico.		X
Selección del valor de forzado	Selección de los valores de forzado que deben transferirse por medio del botón "Forzar una vez e inmediatamente todos los parámetros seleccionados". Esta columna aparece junto con la columna "Valor de forzado".		X
Valor mínimo	Valor tecnológico más bajo del parámetro. Si el valor mínimo depende de otros parámetros, entonces viene determinado: • Offline: por los valores de arranque en el proyecto. • Online: por los valores de observación.	X	X
Valor máximo	Valor tecnológico más alto del parámetro. Si el valor máximo depende de otros parámetros, entonces viene determinado: • Offline: por los valores de arranque en el proyecto. • Online: por los valores de observación.	X	X
Valor de ajuste	Identifica el parámetro como valor de ajuste. Estos parámetros pueden inicializarse online.	X	X
Tipo de datos	Tipo de datos del parámetro. Este campo de visualización está vacío en el caso de parámetros que no están incluidos en el DB de instancia.	X	X
Remanencia	Identifica el valor como remanente. Los valores de los parámetros remanentes se conservan incluso después de desconectar la tensión de alimentación.	X	X
Accesible desde HMI	Indica si HMI puede acceder a este parámetro durante el tiempo de ejecución.	X	X
Visible en HMI	Indica si el parámetro es visible por defecto en la lista de selección de HMI.	X	X
Comentario	Descripción abreviada del parámetro.	X	X

Abrir la vista de parámetros

Requisitos

El objeto tecnológico se ha agregado al navegador de proyectos, lo que significa que se ha generado el correspondiente DB de instancia de la instrucción.

Procedimiento

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. Abra el objeto tecnológico en el árbol del proyecto.
3. Haga doble clic en el objeto "Configuración".
4. Seleccione en la esquina superior derecha la ficha "Vista de parámetros".

Resultado

Se abre la vista de parámetros. En la tabla de parámetros cada parámetro mostrado viene está representado por una fila de la tabla.

Las propiedades visualizables de los parámetros (columnas de la tabla) dependen de si la vista de parámetros está en modo offline u online.

Además, es posible mostrar y ocultar las distintas columnas de la tabla de manera selectiva.

Consulte también

Ajuste predeterminado de la vista de parámetros (Página 9948)

Ajuste predeterminado de la vista de parámetros

Ajustes predeterminados

Para trabajar de forma efectiva con la vista de parámetros, puede adaptar la representación de los parámetros y guardar los ajustes realizados.

Pueden efectuarse y guardarse los siguientes ajustes:

- Mostrar y ocultar columnas
- Modificar el ancho de las columnas
- Modificar el orden de las columnas
- Cambiar la navegación
- Seleccionar un grupo de parámetros en la navegación
- Seleccionar valores de comparación

Mostrar y ocultar columnas

Para mostrar y ocultar columnas en la tabla de parámetros, proceda del siguiente modo:

1. Sitúe el puntero del ratón en el encabezado de la tabla de parámetros.
2. En el menú contextual, seleccione el comando "Mostrar/ocultar".
   Se muestra la selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación de la columna.
4. Para ocultar una columna, desactive la casilla de verificación de la columna.

o bien

1. Sitúe el puntero del ratón en el encabezado de la tabla de parámetros.
2. En el menú contextual, seleccione el comando "Mostrar todas las columnas" si desea mostrar todas las columnas del modo offline u online.

Algunas columnas solo pueden mostrarse en modo online: ver Tabla de parámetros (Página 9946).

Modificar el ancho de las columnas

Para adaptar el ancho de una columna al contenido de manera que pueda leerse todo el texto de las filas, proceda del siguiente modo:

1. Sitúe el puntero del ratón en el encabezado de la tabla de parámetros, a la derecha junto a la columna que se desea optimizar, hasta que el puntero del ratón adopte la forma de una cruz.
2. Haga doble clic en este punto.

o bien

1. Abra el menú contextual del encabezado de la tabla de parámetros.
2. Haga clic en
   - "Optimizar ancho de columna" o
   - "Optimizar ancho de todas las columnas".

Situando brevemente el puntero del ratón sobre el campo en cuestión, aparece el contenido completo de los distintos campos en las columnas de ancho insuficiente.

Modificar el orden de las columnas

Las columnas de la tabla de parámetros pueden situarse donde se desee.

Para modificar el orden de las columnas, proceda del siguiente modo:

1. Haga clic sobre el encabezado de la columna, arrástrelo y suéltelo en el lugar deseado. Cuando suelte el botón del ratón, la columna se anclará en la nueva posición.

Cambiar la navegación

Para cambiar la estructura de visualización de los parámetros, proceda del siguiente modo:

1. Seleccione la navegación que desee en la lista desplegable "Seleccionar la estructura de navegación":
   - Navegación orientada a datos
   - Navegación orientada a funciones

Ver también Navegación (Página 9946).

Seleccionar un grupo de parámetros en la navegación

Dentro de la navegación elegida puede optar entre la visualización "Todos los parámetros" o la visualización de un grupo de parámetros subordinado deseado.

1. Haga clic sobre el grupo de parámetros deseado en la navegación.
   En la tabla de parámetros se muestran solo los parámetros del grupo de parámetros.

Seleccionar valores de comparación (online)

Para ajustar los valores de comparación para la función "Comparar valores", proceda del siguiente modo:

1. Seleccione en la lista desplegable "Seleccionar valores de comparación" los valores de comparación que desee:
   - Valor de arranque en el proyecto / valor de arranque en la CPU
   - Valor de arranque en el proyecto / instantánea
   - Valor de arranque en la CPU / instantánea

Por defecto está ajustada la opción "Valor de arranque en el proyecto / Valor de arranque en la CPU".

Guardar el ajuste predeterminado de la vista de parámetros

Para guardar las modificaciones anteriores de la vista de parámetros, proceda del siguiente modo:

1. Adapte la vista de parámetros en función de sus necesidades.
2. Haga clic en el botón "Memorizar disposición" situado en la parte superior derecha de la vista de parámetros.

Trabajar con la vista de parámetros

Sinopsis

La siguiente tabla ofrece una vista general de las funciones de la vista de parámetros que se describen a continuación en los modos online y offline.

- Columna "Offline" = X: esta función está disponible en modo offline.
- Columna "Online" = X: esta función está disponible en modo online.

Función/acción	Offline	Online
Filtrar la tabla de parámetros (Página 9951)	X	X
Ordenar la tabla de parámetros (Página 9952)	X	X
Aplicar datos de parámetros en otros editores (Página 9952)	X	X
Mostrar errores (Página 9953)	X	X
Editar valores de arranque en el proyecto (Página 9953)	X	X
Estado de la configuración (offline) (Página 9954)	X	
Observar online los valores en la vista de parámetros (Página 9955)		X
Crear instantánea de los valores de observación (Página 9956)		X
Forzar valores (Página 9957)		X
Comparar valores (Página 9958)		X
Aplicar valores como valores de arranque desde el programa online (Página 9959)		X
Inicializar los valores de ajuste en el programa online (Página 9960)		X

Filtrar la tabla de parámetros

Los parámetros de la tabla de parámetros pueden filtrarse de las siguientes maneras:

- Con el filtro de texto
- Con los subgrupos de la navegación

Los dos métodos de filtrado pueden utilizarse simultáneamente.

Con el filtro de texto

Puede filtrarse por los textos que son visibles en la tabla de parámetros. Esto significa que solo se puede filtrar por los textos de las líneas de parámetros y columnas mostradas.

1. Introduzca en el campo de entrada "Filtro de texto..." la cadena de caracteres por la que desee filtrar.
   La tabla de parámetros solo muestra aquellos parámetros que contienen la cadena de caracteres.

El filtro de texto se restablece:

- Seleccionando otro grupo de parámetros en la navegación.
- Cambiando entre la navegación orientada a datos y a funciones.

Con los subgrupos de la navegación

1. Haga clic sobre el grupo de parámetros deseado en la navegación, p. ej., "Static".
   En la tabla de parámetros se muestran solo los parámetros "Static". En algunos grupos de la navegación puede seleccionar otros subgrupos.
2. Haga clic sobre "Todos los parámetros" en la navegación si desea volver a mostrar todos los parámetros.

Ordenar la tabla de parámetros

Los valores de los parámetros están dispuestos por filas. La tabla de parámetros puede ordenarse según cualquiera de las columnas mostradas.

- En las columnas con valores numéricos se ordena según la altura del valor numérico.
- En las columnas con texto se ordena alfabéticamente.

Ordenar por columnas

1. Sitúe el puntero del ratón en el encabezado de la columna deseada.
   El fondo de esta celda se marca en azul.
2. Haga clic sobre el encabezado de la columna.

Resultado

Toda la tabla de parámetros se ordena según la columna marcada. En el encabezado de la columna aparece un triángulo con la punta hacia arriba.

Haciendo clic de nuevo sobre el encabezado de la columna, el orden cambia de la siguiente manera:

- Símbolo "▲": la tabla de parámetros se ordena de manera ascendente.
- Símbolo "▼": la tabla de parámetros se ordena de manera descendente.
- Ningún símbolo: se anula la ordenación. La tabla de parámetros adopta la vista predeterminada.

En la ordenación se ignora el prefijo "../" de la columna "Nombre en DB".

Aplicar datos de parámetros en otros editores

Una vez seleccionada toda una fila de parámetros de la tabla de parámetros, con

- arrastrar y soltar
- <Ctrl+C>/<Ctrl+V>
- copiar/pegar por medio del menú contextual

pueden aplicarse parámetros en los siguientes editores del TIA Portal:

- en el editor de programas;
- en la tabla de observación;
- en la tabla de señales para Trace.

El parámetro se añade con el nombre completo: ver indicación en la columna "Nombre completo en DB".

Mostrar errores

Señalización de errores

Los errores de parametrización que conllevan errores de compilación (p. ej., rebase de límite), se indican en la vista de parámetros.

Cada vez que se introduce un valor en la vista de parámetros, se comprueba y muestra de inmediato la corrección tecnológica y sintáctica.

Los valores erróneos se indican con:

- Un símbolo de error rojo en las columnas "Estado de la configuración" (modo offline) o "Resultado de la comparación" (modo online, en función del tipo de comparación elegido)

y/o

- El campo de la tabla con el fondo rojo
  Al hacer clic sobre el campo erróneo, se muestra el mensaje de error desplegable que indica el rango de valores admisible o la sintaxis requerida (formato)

Errores de compilación

Desde el mensaje de error del compilador puede abrirse directamente la vista de parámetros (navegación orientada a funciones) con el parámetro causante del error para aquellos parámetros que no se muestren en el cuadro de diálogo de configuración.

Editar valores de arranque en el proyecto

Con la vista de parámetros pueden editarse los valores de arranque en el proyecto en los modos offline y online:

- Los valores se modifican en la columna "Valor de arranque en el proyecto" de la tabla de parámetros.
- En la columna "Estado de la configuración" de la tabla de parámetros se muestra el progreso de la configuración mediante los símbolos de estado conocidos del cuadro de diálogo de configuración del objeto tecnológico.

Condiciones

- Si hay otros parámetros que dependen del parámetro cuyo valor de arranque se ha modificado, el valor de arranque del parámetro dependiente se adapta igualmente.
- Si un parámetro de un objeto tecnológico no es editable, tampoco lo es en la vista de parámetros. La posibilidad de edición de un parámetro puede depender también de los valores de otros parámetros.

Definir nuevos valores de arranque

Para especificar valores de arranque para parámetros en la vista de parámetros, proceda del siguiente modo:

1. Abra la vista de parámetros del objeto tecnológico.
2. Introduzca en la columna "Valor de arranque en el proyecto" los valores de arranque deseados. El valor debe ser conforme al tipo de datos del parámetro y no debe rebasar el rango de valores del parámetro.
   En las columnas "Valor máximo" y "Valor mínimo" se muestran los valores límite del rango de valores.

En la columna "Estado de la configuración" se muestra el "progreso" de la configuración mediante símbolos de colores.

Ver también Estado de la configuración (offline) (Página 9954)

Una vez adaptados los valores de arranque y cargado el objeto tecnológico en la CPU, los parámetros adoptan durante el arranque el valor definido, siempre y cuando estos parámetros estén declarados como remanentes (columna "Remanencia").

Señalización de errores

Cuando se introduce un valor de arranque, se comprueba y muestra de inmediato la corrección tecnológica y sintáctica:

Los valores de arranque erróneos se indican con

- Un símbolo de error rojo en las columnas "Estado de la configuración" (modo offline) o "Resultado de la comparación" (modo online, en función del tipo de comparación elegido)

y/o

- Un fondo rojo en el campo "Valor de arranque en el proyecto"
  Al hacer clic sobre el campo erróneo, se muestra el mensaje de error desplegable que indica el rango de valores admisible o la sintaxis requerida (formato)

Corregir valores de arranque erróneos

1. Corrija los valores de arranque erróneos con ayuda de la información del mensaje de error desplegable.
   El símbolo de error rojo, el fondo rojo del campo y el mensaje de error desplegable dejarán de mostrarse.

El proyecto solo puede compilarse correctamente con valores de arranque correctos.

Estado de la configuración (offline)

El estado de la configuración se indica mediante símbolos:

- En la columna "Estado de la configuración" en la tabla de parámetros
- En la estructura de navegación de la navegación orientada a funciones o la navegación orientada a datos

Símbolo en la columna "Estado de la configuración"

Símbolo	Significado
	El valor de arranque del parámetro coincide con el valor predeterminado y es válido. El usuario todavía no ha definido ningún valor de arranque.
	El valor de arranque del parámetro contiene un valor definido por el usuario. El valor de arranque difiere del valor predeterminado. El valor de arranque no contiene errores y es válido.
	El valor de arranque del parámetro no es válido (error sintáctico o tecnológico). El campo de entrada tiene fondo rojo. Al hacer clic, el roll out con el aviso de error indica la causa del error.
	Solo con S7-1200 Motion Control: El valor de arranque del parámetro es válido, pero contiene advertencias. El campo de entrada tiene fondo amarillo.

Símbolo en la navegación

Los símbolos en la navegación muestran el "progreso" de la configuración de la misma manera que en el cuadro de diálogo de configuración del objeto tecnológico.

Ver Configurar objetos tecnológicos (Página 9935)

Observar online los valores en la vista de parámetros

Es posible observar directamente en la vista de parámetros los valores que los parámetros del objeto tecnológico van adoptando actualmente en la CPU (valores de observación).

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.

Procedimiento

1. Inicie la observación haciendo clic en el símbolo .
   En cuanto la vista de parámetros esté online, se mostrarán también las siguientes columnas:
   - Resultado de la comparación
   - Valor de arranque en la CPU
   - Valor de observación
   - Valor de forzado
   - Selección del valor de forzado

   La columna "Valor de observación" muestra los valores actuales de los parámetros en la CPU.
   Significado de las demás columnas: ver Tabla de parámetros (Página 9946)
2. La observación se finaliza haciendo clic de nuevo en el símbolo .

Indicación

Todas las columnas disponibles únicamente online tienen fondo naranja:

- Los valores de las celdas naranja claro se pueden modificar.
- Los valores de las celdas con fondo naranja oscuro no admiten modificaciones.

Crear instantánea de los valores de observación

Es posible guardar los valores actuales del objeto tecnológico en la CPU (valores de observación) y mostrarlos en la vista de parámetros.

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.

Procedimiento

Para mostrar los valores actuales de los parámetros, proceda del siguiente modo:

1. Haga clic en la vista de parámetros sobre el símbolo "Crear instantánea de los valores de observación".

Resultado

Los valores de observación actuales se aplican una vez en la columna "Instantánea" de la tabla de parámetros.

Los valores "congelados" así pueden analizarse mientras que los valores de observación de la columna "Valores de observación" se siguen actualizando.

Forzar valores

Con la vista de parámetros puede forzar los valores del objeto tecnológico en la CPU.

Puede asignar valores al parámetro (valor de forzado) una sola vez y forzarlos de inmediato. Al ejecutar esta función la orden de forzado se lleva a cabo lo antes posible, sin referencia a un punto determinado del programa de usuario.

PELIGRO

Peligro en el forzado:

Una modificación de los valores de los parámetros con la instalación en marcha puede causar importantes daños materiales y personales en caso de fallos de funcionamiento o errores de programa.

Asegúrese de que no puedan darse situaciones de peligro antes de ejecutar la función "Forzar".

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.
- El parámetro puede forzarse (el campo correspondiente en la columna "Valor de forzado" tiene fondo naranja claro).

Procedimiento

Para forzar parámetros inmediatamente, proceda del siguiente modo:

1. Introduzca los valores de forzado deseados en la columna "Valores de forzado" de la tabla de parámetros.
2. Compruebe si se ha activado la casilla de verificación para el forzado en la columna "Selección del valor de forzado".
   Los valores de forzado y las casillas de verificación correspondientes de los parámetros dependientes se adaptan también automáticamente.
3. Haga clic en el símbolo "Forzar una vez e inmediatamente todos los parámetros seleccionados".

Los parámetros seleccionados se fuerzan una vez e inmediatamente con los valores especificados y pueden observarse en la columna "Valores de observación". Las casillas de verificación para el forzado en la columna "Selección del valor de forzado" se desactivan automáticamente una vez ejecutada la orden de forzado.

Señalización de errores

Cuando se introduce un valor de forzado, se comprueba y muestra de inmediato la corrección tecnológica y sintáctica:

Los valores de forzado erróneos se indican con

- Un fondo rojo en el campo "Valor de forzado"

y

- Al hacer clic sobre el campo erróneo, se muestra el mensaje de error desplegable que indica el rango de valores admisible o la sintaxis requerida (formato)

Valores de forzado erróneos

- Los valores de forzado erróneos desde el punto de vista tecnológico pueden transferirse.
- Los valores de forzado erróneos desde el punto de vista sintáctico **no** pueden transferirse.

Comparar valores

Las funciones de comparación permiten comparar los siguientes valores almacenados de un parámetro:

- Valor de arranque en el proyecto
- Valor de arranque en la CPU
- Instantánea

Requisitos

- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").

- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.

Procedimiento

Para comparar los valores de arranque de los diversos sistemas de destino, proceda del siguiente modo:

1. Haga clic en el símbolo "Seleccionar valores de comparación".
   Se abre una lista de selección con las opciones de comparación:
   – Valor de arranque en el proyecto - valor de arranque en la CPU (ajuste predeterminado)
   – Valor de arranque en el proyecto - instantánea
   – Valor de arranque en la CPU - instantánea
2. Elija la opción de comparación que desee.
   La opción de comparación seleccionada se ejecuta de la siguiente manera:
   – En las celdas del encabezado de las dos columnas seleccionadas para la comparación aparece un símbolo con una balanza.
   – En la columna "Resultado de la comparación" se muestra mediante símbolos el resultado de la comparación de las columnas seleccionadas.

Símbolo en la columna "Resultado de la comparación"

Símbolo	Significado
	Los valores de comparación son iguales y no contienen errores.
	Los valores de comparación son distintos y no contienen errores.
	Por lo menos uno de los dos valores de comparación es incorrecto desde el punto de vista tecnológico o sintáctico.
	No se ha podido llevar a cabo la comparación. Por lo menos uno de los dos valores de comparación no está disponible (p. ej., instantánea).

Símbolo en la navegación

Los símbolos se muestran de la misma manera en la navegación cuando el resultado de la comparación se cumple como mínimo para uno de los parámetros de debajo de la estructura de navegación mostrada.

Aplicar valores como valores de arranque desde el programa online

Para aplicar al proyecto en un paso valores optimizados de la CPU como valores de arranque, genere una instantánea de los valores de observación. A continuación, los valores de la instantánea marcados como "Valor de ajuste" se aplicarán como valores de arranque en el proyecto.

Requisitos

- El objeto tecnológico es de tipo "PID_Compact" o "PID_3Step".
- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.

Procedimiento

Para aplicar valores optimizados de la CPU, proceda del siguiente modo:

1. Haga clic en el símbolo "Crear instantánea de los valores de observación y aplicar valores de ajuste de esa instantánea como valores de arranque".

Resultado

Los valores de observación actuales se aplican a la columna "Instantánea" y sus valores de ajuste se copian como valores de arranque nuevos en la columna "Valor de arranque en el proyecto".

Nota

Aplicar valores de parámetros individuales

También es posible aplicar los valores de determinados parámetros que no estén marcados como valor de ajuste de la columna "Instantánea" a la columna "Valores de arranque en el proyecto". Para ello copie los valores con ayuda de los comandos "Copiar"·y "Pegar" del menú contextual y péguelos en la columna "Valor de arranque en el proyecto".

Inicializar los valores de ajuste en el programa online

Todos los parámetros marcados en la vista de parámetros como "Valor de ajuste" se pueden inicializar en la CPU con nuevos valores en un solo paso. Para ello se deben cargar los valores de arranque del proyecto en la CPU. La CPU se mantiene en el estado operativo "RUN".

Para evitar una pérdida de datos en la CPU al arrancar en frío o al reiniciar (arranque en caliente), es necesario cargar también el objeto tecnológico en la CPU.

> ⚠ PELIGRO
>
> **Peligro al modificar valores de parámetros**
>
> Una modificación de los valores de los parámetros con la instalación en marcha puede causar importantes daños materiales y personales en caso de fallos de funcionamiento o errores de programa.
>
> Asegúrese de que no puedan darse situaciones de peligro antes de reinicializar los valores de ajuste.

Requisitos

- El objeto tecnológico es de tipo "PID_Compact" o "PID_3Step".
- Existe una conexión online.
- El objeto tecnológico está cargado en la CPU.
- La ejecución del programa está activa (CPU en "RUN").
- La vista de parámetros del objeto tecnológico está abierta.
- El botón "Observar todos" está activado.
- Los parámetros marcados como "Valor de ajuste" disponen de un "Valor de arranque en el proyecto" sin errores tecnológicos o sintácticos.

Procedimiento

Para inicializar todos los valores de ajuste, proceda del siguiente modo:

1. Introduzca los valores deseados en la columna "Valor de arranque en el proyecto". Compruebe que los valores de arranque no contengan errores tecnológicos o sintácticos.
2. Haga clic en el símbolo "Inicializar valores de ajuste".

Resultado

Los valores de ajuste de la CPU se inicializan con los valores de arranque del proyecto.

Programar

Sinopsis de las instrucciones de Motion Control

Las instrucciones de Motion Control le permiten controlar el eje desde el programa de usuario. Las instrucciones inician peticiones de Motion Control que ejecutan las funciones deseadas.

Puede consultar en los parámetros de salida de las instrucciones de Motion Control el estado de las peticiones de Motion Control, así como los eventuales errores producidos durante su procesamiento. Están disponibles las siguientes instrucciones de Motion Control:

- MC_Power: Habilitar, bloquear ejes V4 (Página 5019)
- MC_Reset: Acusar errores V4 (Página 5023)
- MC_Home: Referenciar ejes, fijar punto de referencia V4 (Página 5025)
- MC_Halt: Parar ejes V4 (Página 5028)
- MC_MoveAbsolute: Posicionar ejes de forma absoluta V4 (Página 5032)
- MC_MoveRelative: Posicionar ejes de forma relativa V4 (Página 5035)
- MC_MoveVelocity: Mover ejes con preajuste de velocidad V4 (Página 5039)
- MC_MoveJog: Mover ejes en el modo de operación Jog V4 (Página 5044)
- MC_CommandTable: Ejecutar peticiones de eje como secuencia de movimientos V4 (Página 5047)
- MC_ChangeDynamic: Modificar ajustes dinámicos del eje V4 (Página 5049)
- MC_ReadParam: Leer de forma continuada datos de movimiento de un eje de posicionamiento V4 (Página 5052)
- MC_WriteParam: Escribir variable del eje de posicionamiento V4 (Página 5053)

Consulte también

Crear un programa de usuario (Página 10527)

Notas de programación (Página 10530)

Comportamiento de peticiones de Motion Control tras desconexión y rearranque completo (Página 10532)

Seguimiento de las peticiones activas (Página 10532)

Indicaciones de error de las instrucciones de Motion Control (Página 10543)

Crear un programa de usuario

A continuación se crea en el programa de usuario la estructura básica para controlar el eje. Las instrucciones de Motion Control utilizadas permiten controlar todas las funciones del eje disponibles.

Requisitos

- El objeto tecnológico está creado y configurado sin errores.

Antes de crear y probar el programa de usuario se recomienda probar el funcionamiento del eje y de los elementos correspondientes de la instalación con el panel de mando del eje.

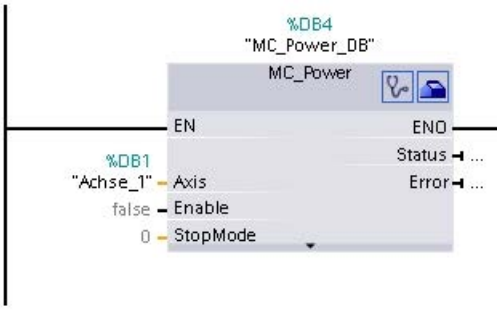
Procedimiento

Para crear el programa de usuario proceda conforme al principio descrito a continuación:

1. En el árbol del proyecto haga doble clic en su bloque lógico (el bloque lógico debe llamarse en el programa cíclico).
   El bloque lógico se abre en el editor de programación y se muestran las instrucciones disponibles.
2. Abra la categoría "Tecnología" y las carpetas "Motion Control" y "S7-1200 Motion Control".
3. Arrastre con Drag & Drop la instrucción "MC_Power" al segmento que desee del bloque lógico.
   Se abre el cuadro de diálogo para definir el DB de instancia.
4. Elija en el diálogo entre las siguientes alternativas:
   Instancia individual
   Haga clic en el botón "Instancia individual" y seleccione si desea determinar el nombre y el número del DB de instancia automática o manualmente.
   Multiinstancia
   Haga clic en el botón "Multiinstancia" y seleccione si desea determinar el número de la multiinstancia automática o manualmente.
5. Haga clic en el botón "Aceptar".
   La instrucción de Motion Control "MC_Power" se inserta en el segmento.

Los parámetros marcados con "<???>" se deben asignar; el resto de parámetros se encuentran asignados con valores predeterminados.
Los parámetros representados en negro son necesarios para usar la instrucción de Motion Control.

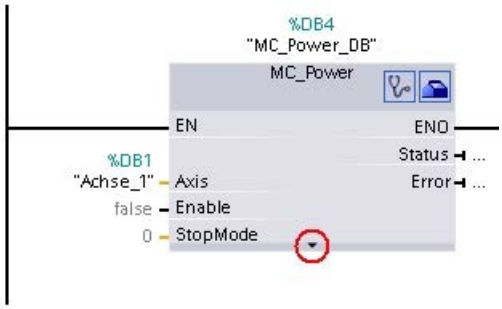
6. Seleccione el objeto tecnológico en el árbol del proyecto y arrástrelo con Drag & Drop hasta <???>.



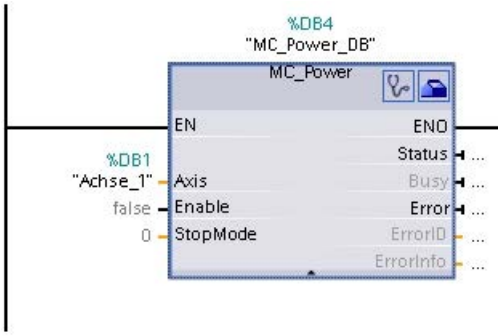
Una vez seleccionado el bloque de datos del objeto tecnológico están disponibles los siguientes botones:

Haga clic en el símbolo del estetoscopio para abrir el diagnóstico del objeto tecnológico.

Haga clic en el símbolo de la caja de herramientas si desea abrir la vista de configuración del objeto tecnológico.



Con un clic en la flecha hacia abajo se abre la vista de otros parámetros de la instrucción Motion Control.



Los parámetros visibles representados ahora en gris pueden ser utilizados de manera opcional.

7. Siga los pasos 3 a 6 para introducir otras instrucciones de Motion Control deseadas.

Resultado

Se ha creado la estructura básica para controlar el eje en el programa de usuario.

Introduzca en otras partes del programa de usuario los parámetros de entrada de las instrucciones de Motion Control para lanzar las peticiones deseadas en el objeto tecnológico "Eje".

Evalúe los parámetros de salida de las instrucciones de Motion Control y las variables del bloque de datos del eje para hacer un seguimiento de las peticiones lanzadas y del estado del eje.

Para más detalles acerca de los parámetros de las instrucciones de Motion Control consulte las descripciones detalladas de los mismos.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10523)

Notas de programación (Página 10530)

Comportamiento de peticiones de Motion Control tras desconexión y rearranque completo (Página 10532)

Seguimiento de las peticiones activas (Página 10532)

Indicaciones de error de las instrucciones de Motion Control (Página 10543)

Notas de programación

Tenga en cuenta las siguientes indicaciones al crear su programa de usuario:

- **Llamada cíclica de las instrucciones de Motion Control empleadas**
  El estado actual de procesamiento de las peticiones se puede consultar a través de los parámetros de salida de la instrucción de Motion Control. El estado se actualiza cada vez que se llama la instrucción Motion Control. Por ello, asegúrese de llamar las instrucciones de Motion Control de forma cíclica.
- **Aceptación de los valores de los parámetros de una instrucción de Motion Control**
  Los valores de los parámetros en los parámetros de entrada se aceptan con un flanco positivo en el parámetro de entrada "Execute" al llamar el bloque.
  La petición de Motion Control se inicia con estos valores de parámetros. Los valores de parámetros modificados a posteriori en la instrucción de Motion Control no se aceptan hasta el siguiente inicio de la petición de Motion Control.
  Una excepción es el parámetro de entrada "StopMode" de la instrucción de Motion Control "MC_Power" y "Velocity" de la instrucción de Motion Control "MC_MoveJog". Una modificación del parámetro de entrada también se acepta si "Enable" = TRUE, o bien "JogForward" y "JogBackward". .

- **Programación considerando las informaciones del estado**
  En un procesamiento paso a paso de peticiones de Motion Control asegúrese de que la petición en curso haya finalizado antes de iniciar una nueva petición. Compruebe la finalización de la petición en curso a través de los avisos de estado de la instrucción de Motion Control y de las variables "StatusBits" del objeto tecnológico.
  En los siguientes ejemplos se debe tener en cuenta la secuencia descrita. Si no se tiene en cuenta se muestra un error del eje o de la petición.
  - **Habilitación del eje con la instrucción de Motion Control "MC_Power"**
    El eje debe ser habilitado antes de que éste pueda aceptar peticiones de movimiento. Compruebe la habilitación del eje mediante una operación lógica Y de la variable <Nombre del eje>.StatusBits.Enable = TRUE con el parámetro de salida Status = TRUE de la instrucción de Motion Control "MC_Power".
  - **Acusar errores con la instrucción de Motion Control "MC_Reset"**
    Antes de iniciar una petición de Motion Control se deben acusar los errores activos con "MC_Reset". Corrija la causa del error y acúselo con la instrucción de Motion Control "MC_Reset". Compruebe que el error haya sido acusado correctamente antes de lanzar una nueva petición. Emplee para ello una operación lógica Y de la variable <Nombre del eje>.StatusBits.Error = FALSE con el parámetro de salida Done = TRUE de la instrucción de Motion Control "MC_Reset".
  - **Referenciación del eje con la instrucción de Motion Control "MC_Home"**
    Antes de poder iniciar una petición MC_MoveAbsolute se debe referenciar el eje. Compruebe tras la referenciación del eje su correcta finalización con una operación lógica Y de la variable <Nombre del eje>.StatusBits.HomingDone = TRUE con el parámetro de salida Done = TRUE de la instrucción de Motion Control "MC_Home".
- **Procesamiento de relevo de peticiones de Motion Control**
  Las peticiones de Motion Control para mover el eje también se pueden ejecutar con carácter de relevo.
  Si se inicia una nueva petición de Motion Control del eje durante una petición de Motion Control en curso, la petición en curso queda relevada por la nueva petición, sin que la petición en curso deba ser ejecutada completamente. La petición relevada notifica esta circunstancia a la instrucción de Motion Control mediante CommandAborted = TRUE. De esta forma, una petición MC_MoveRelative en curso puede ser relevada, p.ej., por una petición MC_MoveAbsolute.
- **Evitar la reutilización de la misma instancia**
  Todas las informaciones relevantes de una petición de Motion Control se guardan en su instancia.
  Cuando utilice esta instancia no inicie ninguna otra petición si desea hacer un seguimiento del estado de la petición actual. Utilice distintas instancias si desea realizar un seguimiento por separado de las peticiones. Si se utiliza la misma instancia para varias peticiones de Motion Control, las informaciones de estado y error de las distintas peticiones se sobrescriben unas a otras.
- **Llamada de instrucciones de Motion Control en distintas clases de prioridad (niveles de ejecución)**
  Las instrucciones de Motion Control que tienen la misma instancia no pueden llamarse en distintas clases de prioridad sin enclavamiento. Para más detalles sobre cómo llamar las instrucciones de Motion Control enclavadas para efectuar un seguimiento, consulte el apartado "Seguimiento de peticiones de clases de prioridad superiores (niveles de ejecución) (Página 10553)".

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10523)

Crear un programa de usuario (Página 10524)

Comportamiento de peticiones de Motion Control tras desconexión y rearranque completo (Página 10532)

Seguimiento de las peticiones activas (Página 10532)

Indicaciones de error de las instrucciones de Motion Control (Página 10543)

Realizar un seguimiento de las peticiones de clases de prioridad superiores (niveles de procesamiento) (Página 10553)

Comportamiento de peticiones de Motion Control tras desconexión y rearranque completo

Con un POWER OFF o STOP de la CPU se cancelan todas las peticiones de Motion Control activas. Todas las salidas de la CPU se inicializan, incluidas las salidas de impulsos y sentido.

Con el siguiente POWER ON o rearranque completo de la CPU (CPU en RUN) se inicializan nuevamente los objetos tecnológicos y las peticiones de Motion Control.

Todos los datos actuales de los objetos tecnológicos, así como todas las informaciones de estado y error de las peticiones de Motion Control activas anteriormente se ponen a los valores iniciales.

Antes de poder utilizar el eje nuevamente, es necesario habilitarlo con la instrucción de Motion Control "MC_Power". Si desea una referenciación, también deberá referenciar el eje nuevamente con la instrucción de Motion Control "MC_Home".

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10523)

Crear un programa de usuario (Página 10524)

Notas de programación (Página 10527)

Seguimiento de las peticiones activas (Página 10532)

Indicaciones de error de las instrucciones de Motion Control (Página 10543)

Seguimiento de las peticiones activas

Seguimiento de las peticiones activas

Al realizar un seguimiento de las peticiones activas de Motion Control se pueden diferenciar tres grupos típicos:

- Instrucciones Motion Control con el parámetro de salida "Done"
- La instrucción de Motion Control "MC_MoveVelocity"
- La instrucción de Motion Control "MC_MoveJog"

Instrucciones Motion Control con parámetro de salida Done

Las instrucciones de Motion Control con el parámetro de salida "Done" se inician a través del parámetro de entrada "Execute" y tienen un final definido (p. ej. con la instrucción de Motion Control "MC_Home": Referenciación realizada con éxito). Con ello, la petición concluye y el eje se para.

Las peticiones de la siguiente instrucción de Motion Control tienen un final predeterminado:

- MC_Reset
- MC_Home
- MC_Halt
- MC_MoveAbsolute
- MC_MoveRelative
- MC_CommandTable (a partir del objeto tecnológico V2)
- MC_ChangeDynamic (a partir del objeto tecnológico V2)
- MC_WriteParam (a partir del objeto tecnológico V4)
- MC_ReadParam (a partir del objeto tecnológico V4)

El parámetro de salida "Done" muestra el valor TRUE si la petición se ejecuta con éxito.

Los parámetros de salida "Busy", "CommandAborted" y "Error" indican que la petición aún está siendo procesada, ha sido cancelada o que se ha producido un error. La instrucción de Motion Control "MC_Reset" no puede ser cancelada, por lo que no dispone de ningún parámetro de salida "CommandAborted". Dado que la instrucción de Motion Control "MC_ChangeDynamic" finaliza de forma inmediata, no dispone de ningún parámetro de salida "Busy" ni "CommandAborted".

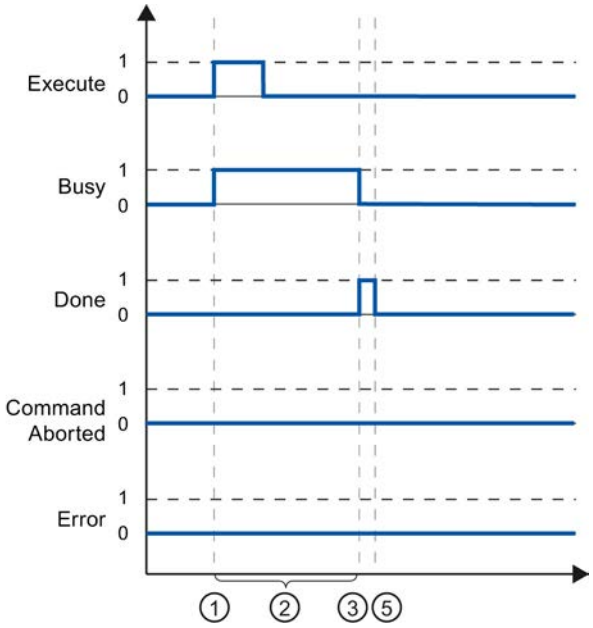
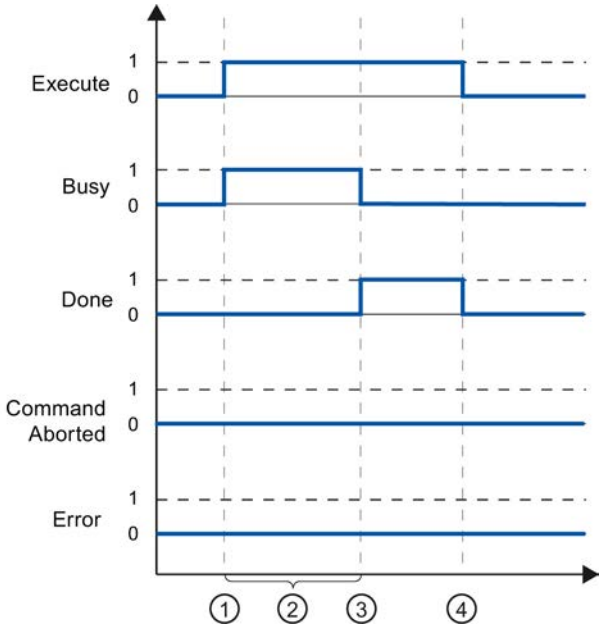
Mientras se procesa la petición de Motion Control el parámetro de salida "Busy" muestra el valor TRUE. Si la petición ha concluido, ha sido cancelada o detenida por un error, el parámetro de salida "Busy" cambia su valor a FALSE. Ello ocurre independientemente de la señal en el parámetro de entrada "Execute".

Los parámetros de salida "Done", "CommandAborted" y "Error" muestran el valor TRUE durante al menos un ciclo. Mientras el parámetro de entrada Execute sea TRUE, los avisos de estado se mostrarán con memoria.

A continuación se muestra, a modo de ejemplo, el comportamiento del bit de estado en diversas situaciones:

Procesamiento íntegro de la petición

Si la petición de Motion Control se procesa íntegramente hasta su finalización, ello se indica a través del parámetro de salida "Done" con el valor TRUE. El estado de la señal del parámetro de entrada "Execute" afecta a la duración de la visualización en el parámetro de salida "Done":

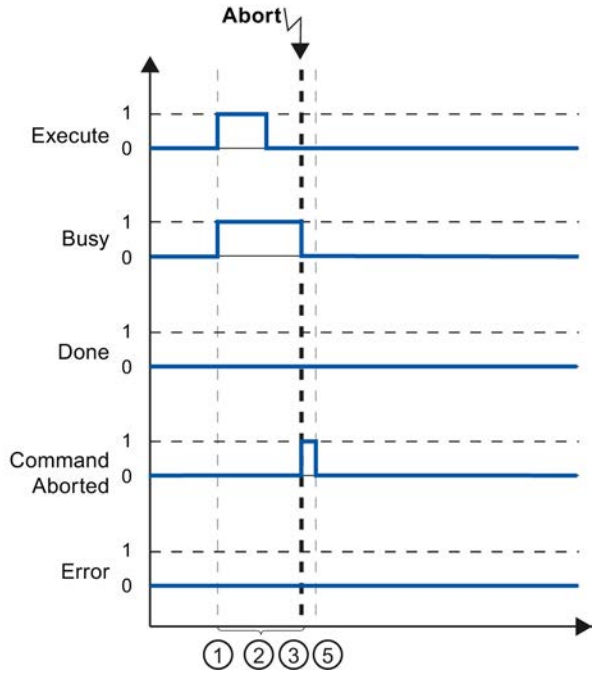
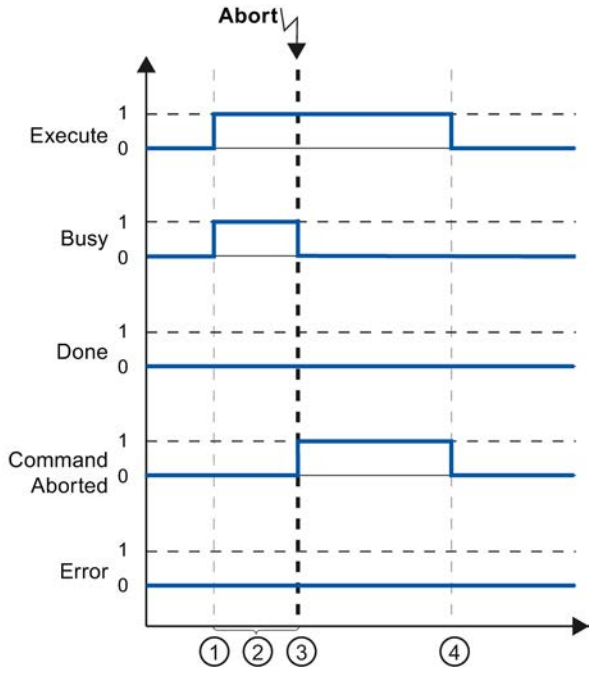
"Execute" cambia su valor **durante el procesamiento** de la petición a FALSE	"Execute" cambia su valor **tras finalizar** la petición a FALSE
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada Execute. Según la programación, es posible desactivar "Execute" a FALSE durante la petición o mantener el valor TRUE hasta la conclusión de la petición.
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	Una vez finalizada la petición (p. ej. con la instrucción de Motion Control "MC_Home": Referenciación realizada con éxito) el parámetro de salida "Busy" cambia a FALSE y "Done" a TRUE.
④	Si "Execute" mantiene el valor TRUE hasta el final de la petición, "Done" también permanece en TRUE y cambia su valor junto con "Execute" a FALSE.
⑤	Si "Execute" cambia a FALSE antes de que la petición finalice, "Done" muestra el valor TRUE únicamente durante un ciclo de procesamiento.

Cancelación de la petición

Si la petición de Motion Control es cancelada durante su procesamiento, ello se indica a través del parámetro de salida "CommandAborted" con el valor TRUE. El estado de la señal del

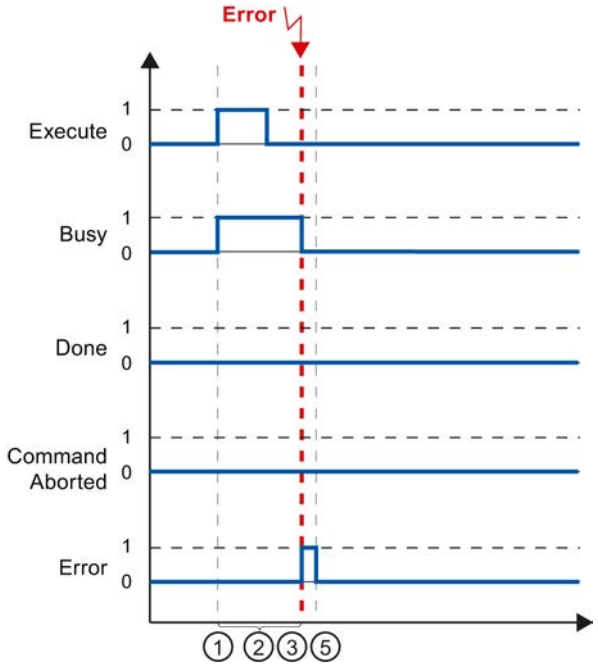
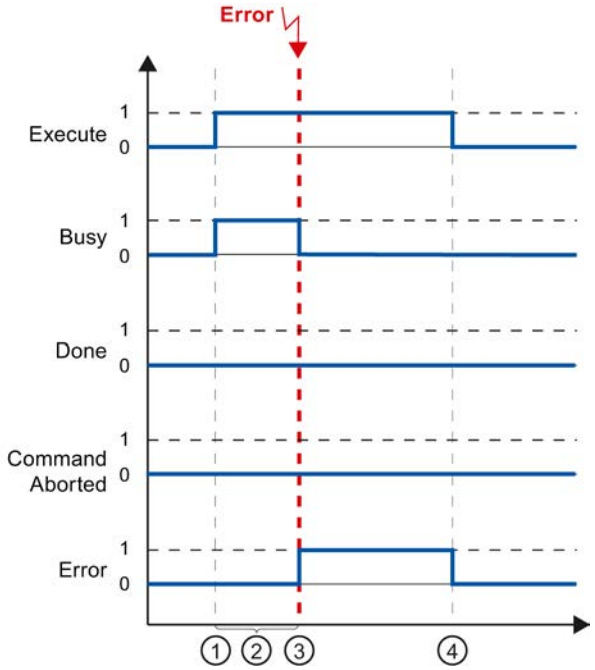
parámetro de entrada "Execute" afecta a la duración de la visualización en el parámetro de salida "CommandAborted":

"Execute" cambia su valor a FALSE **antes** de cancelar la petición.	"Execute" cambia su valor a FALSE **después** de cancelar la petición.
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada Execute. Según la programación, es posible desactivar "Execute" a FALSE durante la petición o mantener el valor TRUE hasta la conclusión de la petición.
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	La petición es cancelada por otra petición de Motion Control durante su procesamiento. Con la cancelación de la petición el parámetro de salida "Busy" cambia a FALSE y "CommandAborted" a TRUE.
④	Si "Execute" mantiene el valor TRUE hasta después de la cancelación de la petición, "CommandAborted" también permanece en TRUE y cambia su valor junto con "Execute" a FALSE.
⑤	Si "Execute" cambia a FALSE ya antes de la cancelación de la petición, "CommandAborted" muestra el valor TRUE únicamente durante un ciclo de procesamiento.

Error durante el procesamiento de la petición

Si se produce un error al procesar la petición de Motion Control, ello se muestra en el parámetro de salida "Error" con el valor TRUE. El estado de la señal del parámetro de entrada "Execute" afecta a la duración de la visualización en el parámetro de salida "Error":

"Execute" cambia su valor a FALSE **antes** de producirse el error.	"Execute" cambia su valor a FALSE **después** de producirse el error.
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada Execute. Según la programación, es posible desactivar "Execute" a FALSE durante la petición o mantener el valor TRUE hasta la conclusión de la petición.
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	Se produce un error al procesar la petición. Al producirse el error el parámetro de salida "Busy" cambia a FALSE y "Error" a TRUE.
④	Si "Execute" mantiene el valor TRUE hasta después de producirse el error, "Error" también permanece en TRUE y cambia su valor junto con "Execute" a FALSE.
⑤	Si "Execute" cambia a FALSE antes de producirse el error, "Error" muestra el valor TRUE únicamente durante un ciclo de procesamiento.

Instrucción Motion Control MC_MoveVelocity

Una petición "MC_MoveVelocity" se inicia con un flanco ascendente en el parámetro "Execute". El objetivo de la petición se cumple en cuanto se alcanza la velocidad parametrizada y el eje se mueve a velocidad constante. En cuanto se alcanza y mantiene la velocidad parametrizada, en el parámetro "InVelocity" se muestra el valor TRUE.

El movimiento del eje se puede detener, p. ej. con una petición "MC_Halt".

Los parámetros de salida "Busy", "CommandAborted" y "Error" indican que la petición aún está siendo procesada, ha sido cancelada o que se ha producido un error.

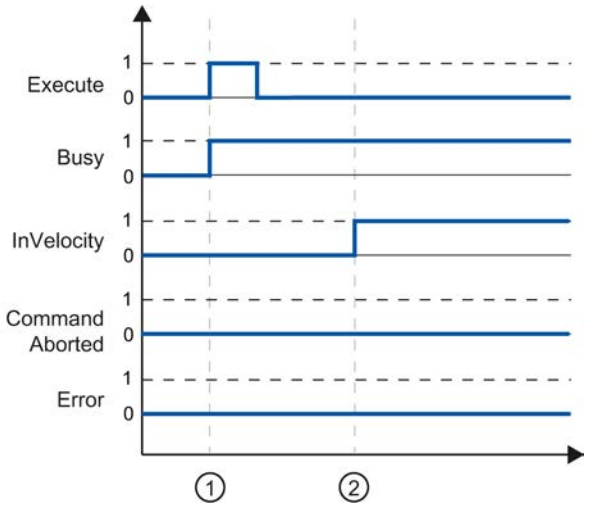
Mientras se procesa la petición de Motion Control el parámetro de salida "Busy" muestra el valor TRUE. Si la petición se cancela por otra petición o por un error, el parámetro de salida "Busy" cambia su valor a FALSE. Ello ocurre independientemente de la señal en el parámetro de entrada "Execute".

Los parámetros de salida "CommandAborted" y "Error" muestran el valor TRUE durante al menos un ciclo. Mientras el parámetro de entrada Execute sea TRUE, los avisos de estado se mostrarán con memoria.

A continuación se muestra, a modo de ejemplo, el comportamiento del bit de estado en diversas situaciones:

Se alcanza la velocidad parametrizada

Si la petición de Motion Control se procesa hasta alcanzar la velocidad parametrizada, ello se indica a través del parámetro de salida "InVelocity" con el valor TRUE. El parámetro "Execute" no influye en la duración de visualización del parámetro "InVelocity".

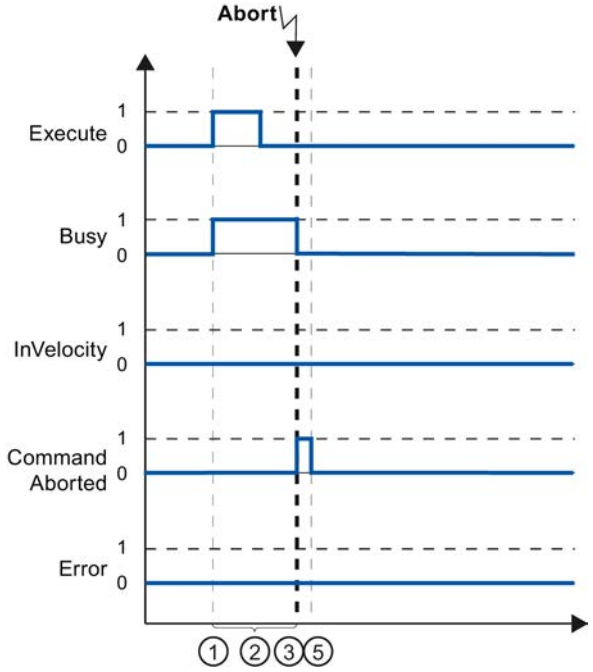
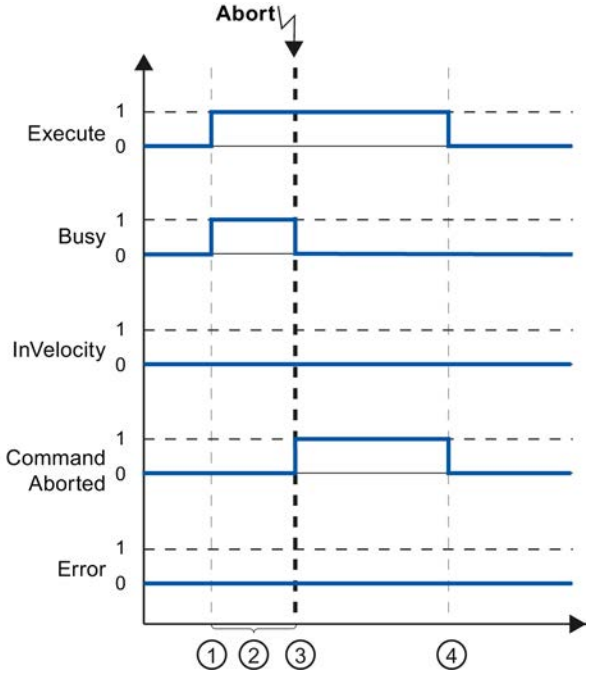


①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, "Execute" se puede volver a poner a FALSE antes o después de alcanzar la velocidad parametrizada. Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
②	Al alcanzar la velocidad parametrizada, el parámetro "InVelocity" cambia a TRUE. Los parámetros "Busy" y "InVelocity" se mantienen en el valor TRUE hasta que la petición "MC_MoveVelocity" es relevada por otra petición de Motion Control o es cancelada con un error.

La petición se cancela antes de alcanzar la velocidad parametrizada

Si la petición de Motion Control se cancela antes de alcanzar la velocidad parametrizada, ello se indica en el parámetro de salida "CommandAborted" con el valor TRUE. El estado de la

señal del parámetro de entrada "Execute" afecta a la duración de la visualización en el parámetro de salida "CommandAborted":

"Execute" cambia su valor a FALSE **antes** de cancelar la petición.	"Execute" cambia su valor a FALSE **después** de cancelar la petición.
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada Execute. Según la programación, "Execute" puede cambiar a FALSE durante la petición o mantener el valor TRUE hasta después de cancelar la petición.
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	La petición es cancelada por otra petición de Motion Control durante su procesamiento. Con la cancelación de la petición el parámetro de salida "Busy" cambia a FALSE y "CommandAborted" a TRUE.
④	Si "Execute" mantiene el valor TRUE hasta después de la cancelación de la petición, "CommandAborted" también permanece TRUE y cambia su estado junto con "Execute" a FALSE.
⑤	Si "Execute" vuelve a ser FALSE antes de la cancelación de la petición, "CommandAborted" muestra el valor TRUE únicamente durante un ciclo de procesamiento.

Nota

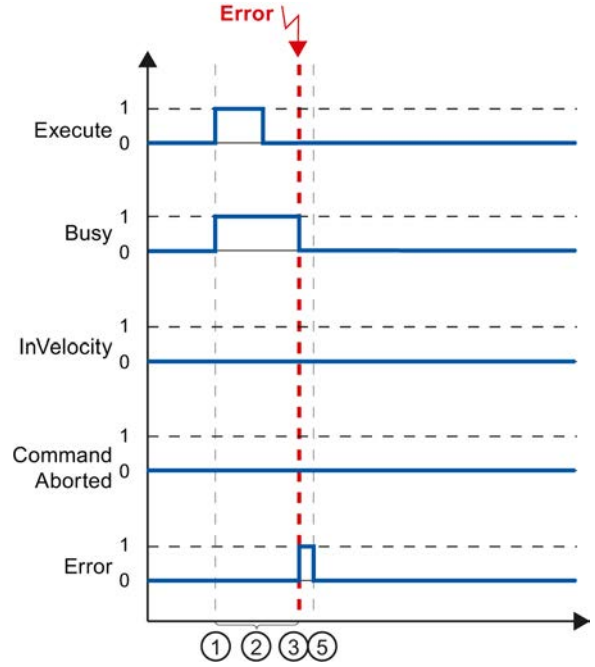
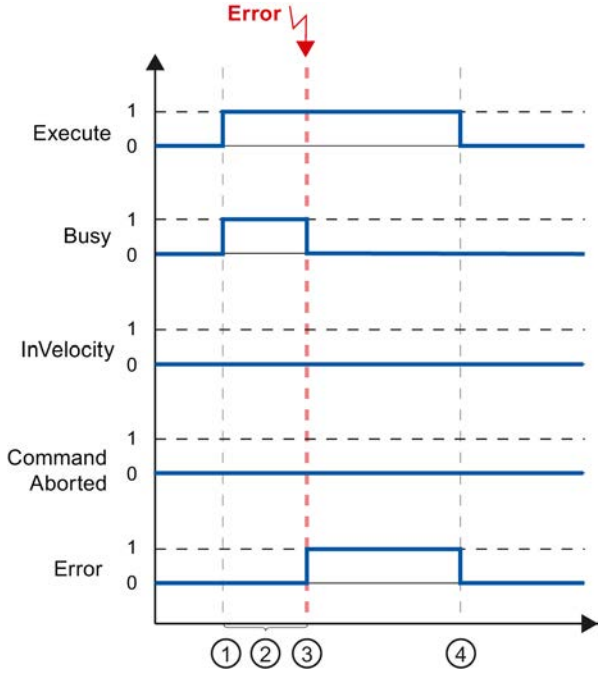
Bajo las siguientes condiciones no se muestra ninguna cancelación en el parámetro de salida "CommandAborted":

Se ha alcanzado la velocidad parametrizada, el parámetro de entrada "Execute" es FALSE y se lanza una nueva petición de Motion Control.

Antes de alcanzarse la velocidad parametrizada se produce un error

Si se produce un error mientras se procesa la petición de Motion Control antes de alcanzarse la velocidad parametrizada, ello se muestra en el parámetro de salida "Error" con el valor

TRUE. El estado de la señal del parámetro de entrada "Execute" afecta a la duración de la visualización en el parámetro de salida "Error":

"Execute" cambia su valor a FALSE **antes** de producirse el error.	"Execute" cambia su valor a FALSE **después** de producirse el error.
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada Execute. Según la programación, "Execute" puede cambiar a FALSE durante la petición o mantener el valor TRUE hasta después de producirse el error.
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	Se produce un error al procesar la petición. Al producirse el error el parámetro de salida "Busy" cambia a FALSE y "Error" a TRUE.
④	Si "Execute" mantiene el valor TRUE hasta después de producirse el error, "Error" también permanece en TRUE y cambia su estado a FALSE con "Execute".
⑤	Si "Execute" cambia a FALSE antes de producirse el error, "Error" muestra el valor TRUE únicamente durante un ciclo de procesamiento.

Nota

Bajo las siguientes condiciones no se muestra ningún error en el parámetro de salida "Error":

Se ha alcanzado la velocidad parametrizada, el parámetro de entrada "Execute" tiene el valor FALSE y se produce un error en el eje (p. ej. se alcanza el final de carrera por software).

El error del eje se indica únicamente en la instrucción de Motion Control "MC_Power".

Instrucción Motion Control MC_MoveJog

Las peticiones de la instrucción de Motion Control "MC_MoveJog" realizan un modo Jog.

Las peticiones de las instrucciones de Motion Control "MC_MoveJog" no tienen un final definido. El objetivo de la petición se cumple en cuanto se alcanza por primera vez la velocidad parametrizada y el eje se mueve a velocidad constante. En cuanto se alcanza la velocidad parametrizada, en el parámetro de salida "InVelocity" se muestra el valor TRUE.

La petición finaliza en cuanto al parámetro de entrada "JogForward" o "JogBackward" es FALSE y el eje se para por completo.

Los parámetros de salida "Busy", "CommandAborted" y "Error" indican que la petición aún está siendo procesada, ha sido cancelada o que se ha producido un error.

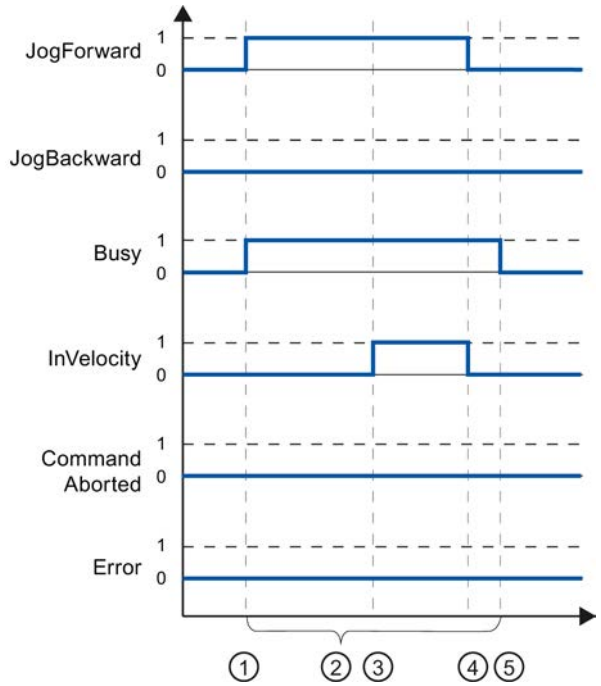
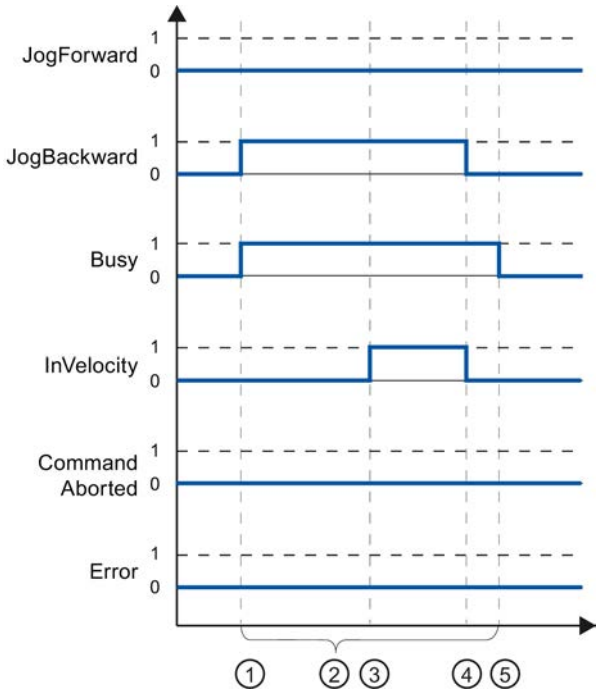
Mientras se procesa la petición de Motion Control el parámetro de salida "Busy" muestra el valor TRUE. Si la petición ha concluido, ha sido cancelada o detenida por un error, el parámetro de salida "Busy" cambia su valor a FALSE.

El parámetro de salida "InVelocity" muestra el estado TRUE mientras que el eje se mueva a la velocidad parametrizada. Los parámetros de salida "CommandAborted" y "Error" muestran el estado durante al menos un ciclo. Mientras uno de los parámetros de entrada "JogForward" o "JogBackward" es TRUE, los avisos de estado se muestran con copia.

A continuación se muestra, a modo de ejemplo, el comportamiento del bit de estado en diversas situaciones:

Se alcanza y mantiene la velocidad parametrizada

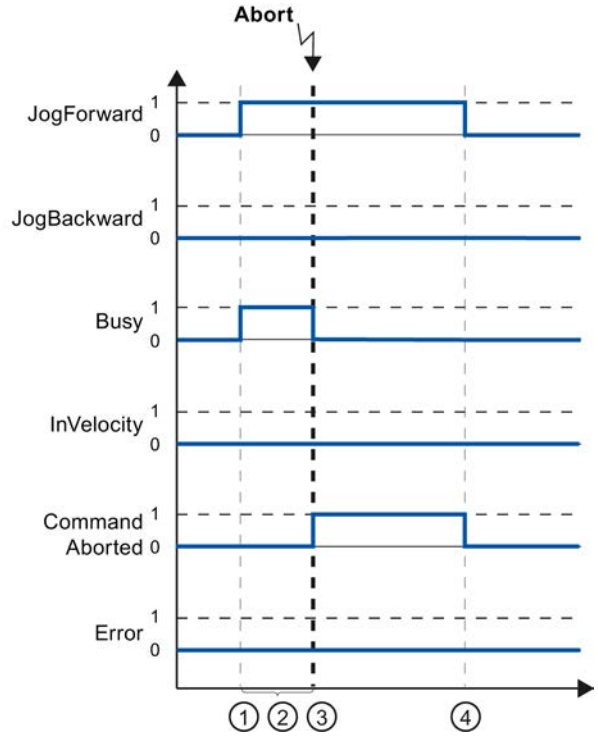
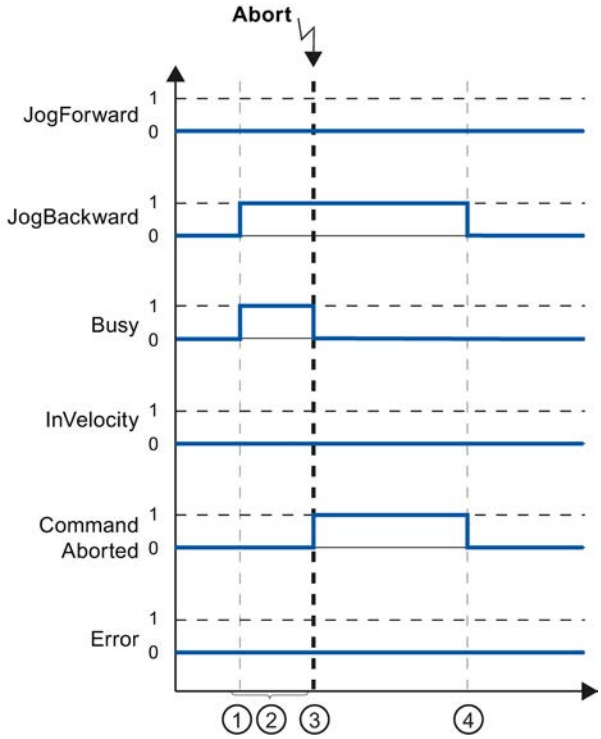
Si la petición de Motion Control se procesa hasta alcanzar la velocidad parametrizada, ello se indica a través del parámetro de salida "InVelocity" con el valor TRUE.

El modo Jog se controla a través del parámetro de entrada "JogForward"	El modo Jog se controla a través del parámetro de entrada "JogBackward".
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada "JogForward" o "JogBackward".
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	Al alcanzar la velocidad parametrizada el parámetro de salida "InVelocity" cambia a TRUE.
④	El movimiento del eje finaliza en cuanto se reajusta el parámetro de entrada "JogForward" o "JogBackward" al valor FALSE. El eje comienza a decelerar. De esta forma, el eje deja de moverse a velocidad constante y el parámetro de salida "InVelocity" cambia su estado a FALSE.
⑤	En cuanto el eje se para, la petición de Motion Control finaliza y el parámetro de salida "Busy" cambia su valor a FALSE.

La petición se cancela durante el procesamiento

Si la petición de Motion Control es cancelada durante su procesamiento, ello se indica a través del parámetro de salida "CommandAborted" con el valor TRUE. El comportamiento depende de si se ha alcanzado o no la velocidad parametrizada.

El modo Jog se controla a través del parámetro de entrada "JogForward".	El modo Jog se controla a través del parámetro de entrada "JogBackward".
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada "JogForward" o "JogBackward".
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.
③	La petición es cancelada por otra petición de Motion Control durante su procesamiento. Con la cancelación de la petición el parámetro de salida "Busy" cambia a FALSE y "CommandAborted" a TRUE.
④	Al inicializar el parámetro de entrada "JogForward" o "JogBackward" al valor FALSE, el valor del parámetro de salida"CommandAborted" cambia su valor a FALSE.

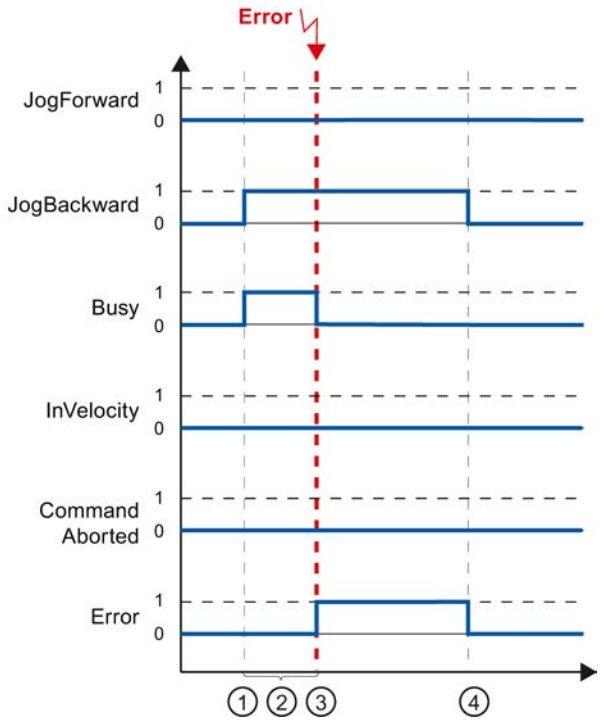
Nota

La cancelación de la petición se muestra en el parámetro de salida "CommandAborted" únicamente durante un ciclo de procesamiento, cuando se cumplen todas las condiciones siguientes:

Los parámetros de entrada "JogForward" y "JogBackward" tienen el valor FALSE (el eje continúa no obstante decelerando) y se lanza una nueva petición de Motion Control.

Se produce un error mientras se procesa la petición

Si se produce un error mientras se procesa la petición de Motion Control, ello se muestra en el parámetro de salida "Error" con el valor TRUE. El comportamiento depende de si se ha alcanzado o no la velocidad parametrizada.

El modo Jog se controla a través del parámetro de entrada "JogForward".	El modo Jog se controla a través del parámetro de entrada "JogBackward".
	

①	La petición se inicia con un flanco ascendente en el parámetro de entrada "JogForward" o "JogBackward".
②	Mientras la petición se encuentra activa, el parámetro de salida "Busy" muestra el valor TRUE.

③	Se produce un error al procesar la petición. Al producirse el error el parámetro de salida "Busy" cambia a FALSE y "Error" a TRUE.
④	Al reajustar el parámetro de entrada "JogForward" o "JogBackward" a FALSE, el valor del parámetro de salida"Error" cambia su valor a FALSE.

Nota

Si se produce un error, éste se muestra en el parámetro de salida "Error" únicamente durante un ciclo de procesamiento, si se cumplen todas las condiciones siguientes:

Los parámetros de entrada "JogForward" y "JogBackward" tienen el valor FALSE (el eje continúa no obstante decelerando) y se produce un error (p.ej. se alcanza un final de carrera por software).

Indicaciones de error de las instrucciones de Motion Control

Las instrucciones de Motion Control muestran posibles errores de las peticiones Motion Control y del objeto tecnológico en los parámetros de salida "Error", "ErrorID" y "ErroInfo" de las instrucciones de Motion Control.

Indicación de errores en los parámetros de salida "Error", "ErrorID" y "ErrorInfo"

El parámetro de salida "Error" muestra el valor TRUE para indicar que la petición no ha podido ser ejecutada o que ésta no ha sido ejecutada completamente. La causa del error se desprende del valor del parámetro de salida "ErrorID" . El valor del parámetro de salida "ErrorInfo" provee información detallada sobre la causa del error. En la indicación de errores se distingue entre las siguientes clases de error:

- **Error de funcionamiento con parada del eje (p.ej. "final de carrera por hardware alcanzado")**
  Los errores de funcionamiento con parada del eje son errores que surgen durante el tiempo de ejecución del programa de usuario. Si el eje se encuentra en movimiento, en función del error se detiene con la deceleración configurada o con la de parada de emergencia. Los errores se muestran en la instrucción de Motion Control causante del error y en la instrucción de Motion Control "MC_Power".
- **Error de funcionamiento sin parada del eje (p.ej. "eje no referenciado")**
  Los errores de funcionamiento sin parada del eje son errores que surgen durante el tiempo de ejecución del programa de usuario. Si el eje se encuentra en movimiento, el movimiento continúa. Los errores sólo se muestran en la instrucción de Motion Control causante del error.
- **Error de parametrización de la instrucción de Motion Control (p. ej. "Valor incorrecto en el parámetro "Velocity")**
  Los errores de parametrización se producen cuando existen indicaciones incorrectas en los parámetros de entrada de las instrucciones de Motion Control. Si el eje se encuentra en movimiento, el movimiento continúa. Los errores sólo se muestran en la instrucción de Motion Control causante del error.

- **Error de configuración en el objeto tecnológico "Eje" (p. ej. el valor de "aceleración" no es válido)**
  Se produce un error de configuración cuando en la configuración del eje se configuran incorrectamente uno o varios parámetros o cuando se modifican incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa. Si el eje se encuentra en movimiento, éste para con la deceleración de parada de emergencia configurada. El error se muestra en la instrucción de Motion Control causante del error y en la instrucción de Motion Control "MC_Power".
- **Error de configuración en el TO "Tabla de peticiones" (p. ej., "El valor de "Velocidad" no es válido")**
  Existe un error de configuración cuando en la configuración de la tabla de peticiones se configuran incorrectamente uno o varios parámetros o cuando se modifican incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa. Si el eje se encuentra en movimiento, el movimiento continúa. Los errores se indican únicamente en la instrucción de Motion Control "MC_CommandTable".
- **Error interno**
  El eje se para en cuanto se produce un error interno. Los errores se muestran en la instrucción de Motion Control causante del error y, en parte, en la instrucción de Motion Control "MC_Power".

En el anexo (Página 10563) encontrará una descripción detallada de los ErrorID y ErrorInfo, así como ayudas para ellos.

Consulte también

Sinopsis de las instrucciones de Motion Control (Página 10523)

Crear un programa de usuario (Página 10524)

Notas de programación (Página 10527)

Comportamiento de peticiones de Motion Control tras desconexión y rearranque completo (Página 10529)

Seguimiento de las peticiones activas (Página 10529)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

Diagnóstico del eje

Bits de estado y error (objetos tecnológicos a partir de V4)

Con la función de diagnóstico "Bits de estado y error" se vigilan en el portal TIA los principales avisos de estado y error del eje. Cuando el eje se encuentra activo la indicación de la función de diagnóstico está disponible en el modo online, en los modos de operación "Control manual" y "Modo automático". Los avisos de estado y error mostrados tienen el siguiente significado:

Avisos de estado

Aviso de estado - Eje	Descripción
Habilitado	El eje está habilitado y listo para ser controlado mediante peticiones de Motion Control. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Enable)
Referenciado	El eje está referenciado y puede ejecutar peticiones de posicionamiento absoluto de la instrucción de Motion Control "MC_MoveAbsolute". Para realizar un posicionamiento relativo no es necesario que el eje esté referenciado. Casos especiales: • Durante la referenciación activa el estado es FALSE. • Si se referencia un eje de forma pasiva, el estado durante la referenciación pasiva es TRUE. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.HomingDone)
Error en eje	Se ha producido un error en el objeto tecnológico "Eje". En el modo automático, la información detallada sobre el error se puede consultar en los parámetros ErrorID y ErrorInfo de las instrucciones de Motion Control. En el modo manual, la causa detallada del error se muestra en el campo "Aviso de error" del panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Error)
Panel de mando activo	Se ha activado el modo de operación "Control manual" en el panel de mando del eje. El panel de mando asume la autoridad sobre el control del objeto tecnológico "Eje". El eje no puede ser controlado por el programa de usuario. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.ControlPanelActive)
Reinicio requerido	Se ha cargado en la memoria de carga una configuración modificada del eje en el estado operativo RUN de la CPU. Para cargar la configuración modificada en la memoria de trabajo, es necesario reiniciar el eje. Utilice para ello la instrucción de Motion Control MC_Reset. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.RestartRequired)

Aviso de estado - Accionamiento	Descripción
Listo	El accionamiento está listo. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.DriveReady)
Error en accionamiento	El accionamiento ha indicado un error debido a un fallo en su señal de disponibilidad. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.DriveFault)

Aviso de estado - Movimiento	Descripción
Parada	El eje está parado. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.StandStill)
Aceleración	El eje está acelerando. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Accelerating)
Velocidad constante	El eje se mueve a una velocidad constante. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.ConstantVelocity)
Deceleración	El eje está decelerando (frenando). (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Decelerating)

Aviso de estado - Tipo de movimiento	Descripción
Posicionamiento	El eje ejecuta una petición de posicionamiento de la instrucción de Motion Control "MC_MoveAbsolute", "MC_MoveRelative" o del panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.PositioningCommand)
Movimiento con velocidad predeterminada	El eje ejecuta una petición a la velocidad especificada en la instrucción de Motion Control "MC_MoveVelocity", "MC_MoveJog" o en el panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.VelocityCommand)
Referenciación	El eje ejecuta una petición de referenciación de la instrucción de Motion Control "MC_Home" o del panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.HomingCommand)
Tabla de peticiones activa	El eje se controla con la instrucción de Motion Control "MC_CommandTable". (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.CommandTableActive)

Avisos de estado de final de carrera

Aviso de estado de final de carrera	Descripción
Final de carrera por SW inferior aproximado	Se ha aproximado o sobrepasado un final de carrera por software. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.SWLimitMinActive)
Final de carrera por SW superior aproximado	Se ha aproximado o sobrepasado un final de carrera por hardware. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.SWLimitMaxActive)
Final de carrera por HW inferior aproximado	Se ha aproximado o sobrepasado el final de carrera por hardware inferior. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.HWLimitMinActive)
Final de carrera por HW superior aproximado	Se ha aproximado o sobrepasado el final de carrera por hardware superior. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.HWLimitMaxActive)

Avisos de error

Aviso de error	Descripción
Final de carrera por SW aproximado	Se ha aproximado o sobrepasado un final de carrera por software. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SWLimit)
Final de carrera por HW aproximado	Se ha aproximado o sobrepasado un final de carrera por hardware. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.HWLimit)
Sentido de movimiento inadmisible	El sentido de la petición no se corresponde con el sentido configurado. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.DirectionFault)
PTO en uso	Un segundo eje utiliza el mismo PTO (Pulse Train Output) y HSC (High Speed Counter) y se encuentra habilitado con "MC_Power". (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.HWUsed)

Aviso de error	Descripción
Error de configuración	El objeto tecnológico "eje" ha sido configurado incorrectamente o se han modificado incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa de usuario. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.ConfigFault)
Error interno	Se ha producido un error interno. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SystemFault)

En la ventana de salida que se encuentra debajo se muestra el primer error notificado y aún no acusado.

Consulte también

Variables StatusBits a partir de V4 (Página 10586)

Variables ErrorBits a partir de V4 (Página 10590)

Bits de diagnóstico, estado y error (objeto tecnológico "Eje" V1...3) (Página 10606)

Lista de compatibilidad de las variables (Página 10445)

Estado de movimiento (Página 10547)

Estado de movimiento

Con la función de diagnóstico "Estado de movimiento" se vigila el estado de movimiento del eje en el portal TIA. Cuando el eje se encuentra activo la indicación de la función de diagnóstico está disponible en el modo online, en los modos de operación "Control manual" y "Modo automático". Las informaciones de estado mostradas tienen el siguiente significado:

Estado	Descripción
Posición actual	El campo "Posición actual" muestra la posición actual del eje. Si el eje no está referenciado, el valor muestra el valor de posicionamiento relativo a la posición de habilitación del eje. (Variable del objeto tecnológico: <Nombre de eje>.Position)
Velocidad actual	El campo "Velocidad actual" muestra la velocidad actual del eje. (Variable del objeto tecnológico: <Nombre de eje>.Velocity)
Posición de destino	El campo "Posición de destino" indica la posición de destino actual de una petición de posicionamiento activa o del panel de mando del eje. El valor de la "Posición de destino" solamente es válido mientras se ejecuta una petición de posicionamiento. (Variable del objeto tecnológico: <Nombre de eje>.StatusPositioning.TargetPosition)
Recorrido residual	El campo "Recorrido restante" indica el recorrido que queda en la actualidad de una petición de posicionamiento activa o del panel de mando del eje. El valor de "Recorrido restante" solamente es válido mientras se ejecuta una petición de posicionamiento. (Variable del objeto tecnológico: <Nombre de eje>.StatusPositioning.Distance)

Consulte también

Variable Position a partir de V4 (Página 10573)

Variable Velocity a partir de V4 (Página 10573)

Variables StatusPositioning a partir de V4 (Página 10585)

Lista de compatibilidad de las variables (Página 10445)

Bits de estado y error (objetos tecnológicos a partir de V4) (Página 10541)

Ajustes dinámicos

Con la función de diagnóstico "Ajustes dinámicos" se vigilan en el portal TIA los valores límite dinámicos configurados para el eje. Cuando el eje se encuentra activo la indicación de la función de diagnóstico está disponible en el modo online, en los modos de operación "Control manual" y "Modo automático". Las informaciones de estado mostradas tienen el siguiente significado:

Límite dinámico	Descripción
Aceleración	El campo "Aceleración" muestra la aceleración actual configurada para el eje. (Variable del objeto tecnológico: <Nombre de eje>.DynamicDefaults.Acceleration)
Deceleración	El campo "Deceleración" muestra la deceleración actual configurada para el eje. (Variable del objeto tecnológico: <Nombre de eje>.DynamicDefaults.Deceleration)
Deceleración de parada de emergencia	El campo "Deceleración de parada de emergencia" muestra la deceleración de parada de emergencia actual configurada para el eje. (Variable del objeto tecnológico: <Nombre de eje>.DynamicDefaults.EmergencyDeceleration)
Tirón (a partir del objeto tecnológico V2)	El campo muestra el tirón configurado en la actualidad para el eje. (Variable del objeto tecnológico: <Nombre de eje>.DynamicDefaults.Jerk)

Consulte también

Variables DynamicDefaults a partir de V4 (Página 10580)

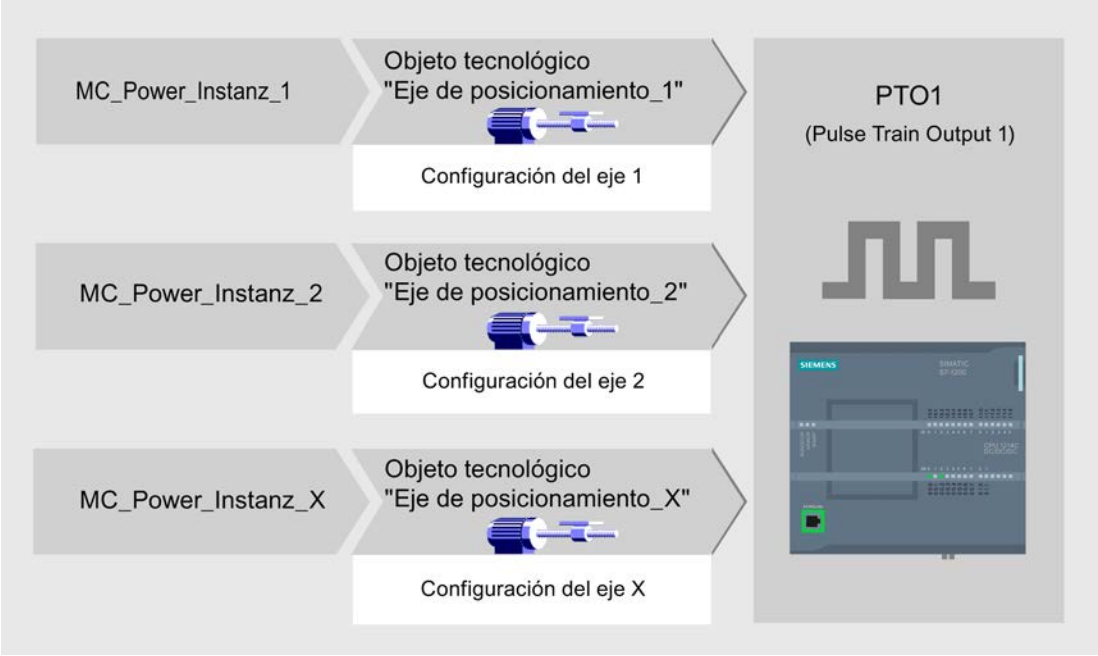
Lista de compatibilidad de las variables (Página 10445)

Anexo

Utilizando varios ejes con el mismo PTO

Utilice la funcionalidad de Motion Control de la CPU S7-1200 para utilizar varios objetos tecnológicos Eje de posicionamiento con el mismo PTO (Pulse Train Output) y, de esta forma, con las mismas salidas de la CPU. Ello resulta adecuado, p.ej., para utilizar a través de un PTO diversas configuraciones de eje para diferentes procesos de producción. Es posible

alternar a voluntad entre estas configuraciones de eje, tal y como se describe a continuación. La siguiente representación muestra las principales relaciones funcionales:

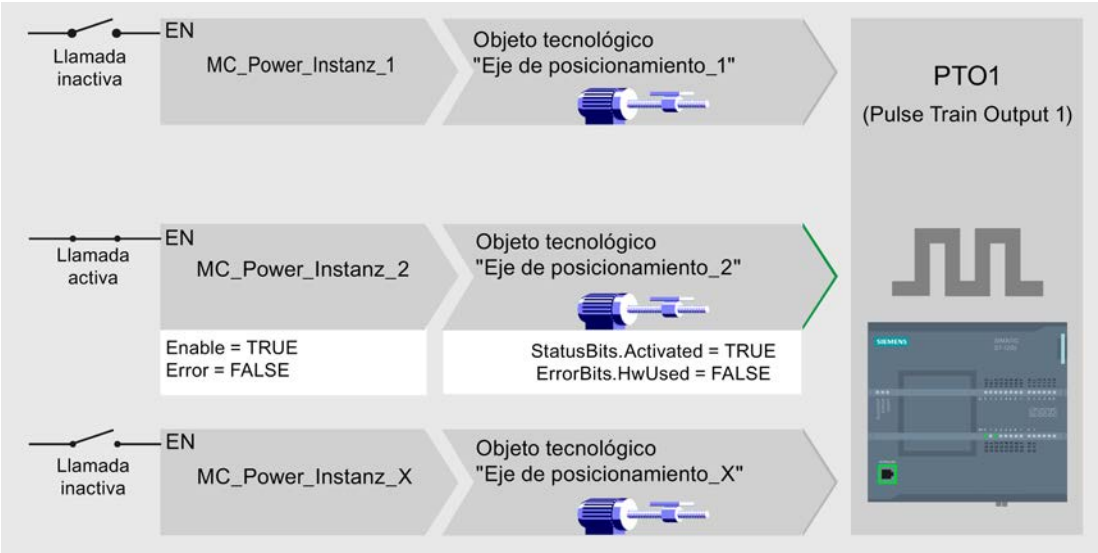


En el ejemplo ilustrado, varios objetos tecnológicos Eje de posicionamiento, cada uno con una configuración propia de eje, utilizan el mismo PTO. En el programa de usuario se debe llamar cada eje con una llamada propia de la instrucción de Motion Control "MC_Power" con un bloque de datos de instancia propio. En cada momento el PTO puede ser utilizado cada vez por un solo eje. El eje que utiliza el PTO en el instante actual lo indica con la variable <Nombre de eje>.StatusBits.Activated = TRUE.

Cambio del objeto tecnológico Eje de posicionamiento

El siguiente esquema de programa muestra cómo cambiar entre diversos objetos tecnológicos y, con ello, entre diversas configuraciones de eje. Para poder utilizar el mismo PTO con diferentes ejes sin indicaciones de error, solamente se deben llamar instrucciones de Motion Control del eje a utilizar actualmente.

La siguiente representación muestra lo anterior utilizando la instrucción de Motion Control "MC_Power" como ejemplo:



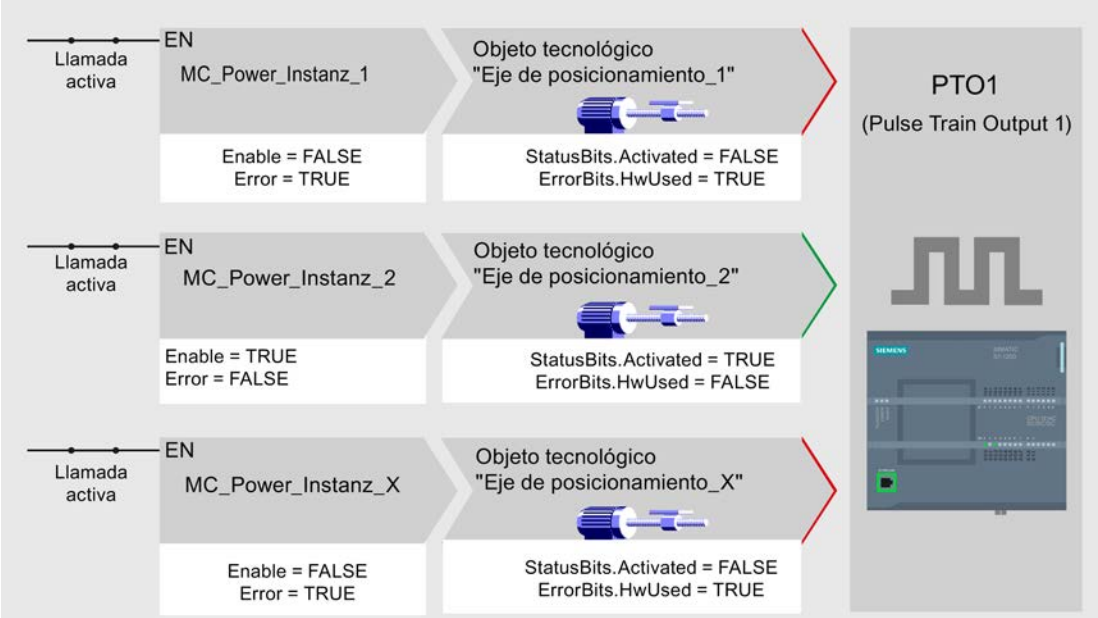
Las variables del eje activado (aquí "eje de posicionamiento_2") muestran las siguientes visualizaciones típicas en el programa de usuario:

- <Nombre de eje>.StatusBits.Activated = TRUE
- <Nombre de eje>.ErrorBits.HWUsed = FALSE

Proceda como se describe a continuación para cambiar el objeto tecnológico Eje de posicionamiento. El ejemplo muestra el cambio del "Eje de posicionamiento_2" al "Eje de posicionamiento_1":

1. Finalice los eventuales movimientos en curso del "Eje de posicionamiento_2" activado
2. Bloquee el "Eje de posicionamiento_2" con la pertinente instrucción de Motion Control "MC_Power" mediante el parámetro de entrada Enable = FALSE
3. Compruebe si se ha bloqueado el "Eje de posicionamiento_2" con una operación lógica Y del parámetro de salida Status = FALSE de la instrucción de Motion Control "MC_Power" y la variable del objeto tecnológico <nombre del eje>.StatusBits.Enable = FALSE.
4. Desactive la llamada condicionada de las instrucciones de Motion Control para el "Eje de posicionamiento_2"
5. Active la llamada condicionada de las instrucciones de Motion Control para el "Eje de posicionamiento_1". Con la primera llamada de la correspondiente instrucción de Motion Control "MC_Power" se desactiva el "Eje de posicionamiento_2" y se activa el "Eje de posicionamiento_1".
6. Desbloquee el "Eje de posicionamiento_1" con la instrucción de Motion Control "MC_Power" mediante el parámetro de entrada Enable = TRUE.
7. Compruebe la habilitación del "Eje de posicionamiento_1" realizada con una operación lógica Y del parámetro de salida Status = TRUE de la instrucción de Motion Control "MC_Power" y la variable del objeto tecnológico <Nombre de eje>.StatusBits.Enable = TRUE.

Básicamente se pueden llamar también cíclicamente todas las instrucciones de Motion Control de todos los ejes que operan con un PTO.



Un eje (aquí "Eje de posicionamiento_2") queda activado al habilitarlo.

Al contrario de lo que ocurre con la llamada condicionada, las instrucciones de Motion Control de los ejes desactivados (aquí "Eje de posicionamiento_1" y "Eje de posicionamiento_x") muestran errores. Las variables de estos ejes muestran el estado <Nombre de eje>.StatusBits.Activated = FALSE y <Nombre de eje>.ErrorBits.HWUsed = TRUE.

Utilice la llamada condicionada de las instrucciones de Motion Control si desea configurar el programa de usuario sin indicaciones de error.

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

Utilizar varios accionamientos con el mismo PTO (Página 10552)

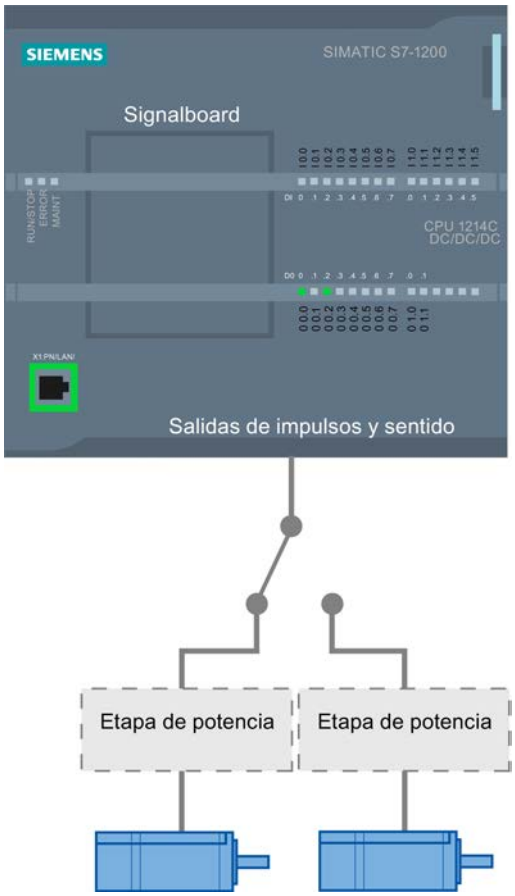
Realizar un seguimiento de las peticiones de clases de prioridad superiores (niveles de procesamiento) (Página 10553)

Casos especiales al utilizar finales de carrera por software (Página 10555)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

Utilizar varios accionamientos con el mismo PTO

Si desea utilizar varios accionamientos de forma alternativa, puede utilizarlos con un PTO conjunto mediante una conmutación (Pulse Train Output). La siguiente representación muestra la estructura básica de conexión:



La conmutación de los accionamientos puede ser controlada, si fuera necesario, por el programa de usuario, a través de una salida digital. Si se precisan diferentes configuraciones de eje para los diferentes accionamientos, éstas se deben conmutar para el PTO. Más información al respecto en " Utilizando varios ejes con el mismo PTO (Página 10545)".

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

Utilizando varios ejes con el mismo PTO (Página 10545)

Realizar un seguimiento de las peticiones de clases de prioridad superiores (niveles de procesamiento) (Página 10553)

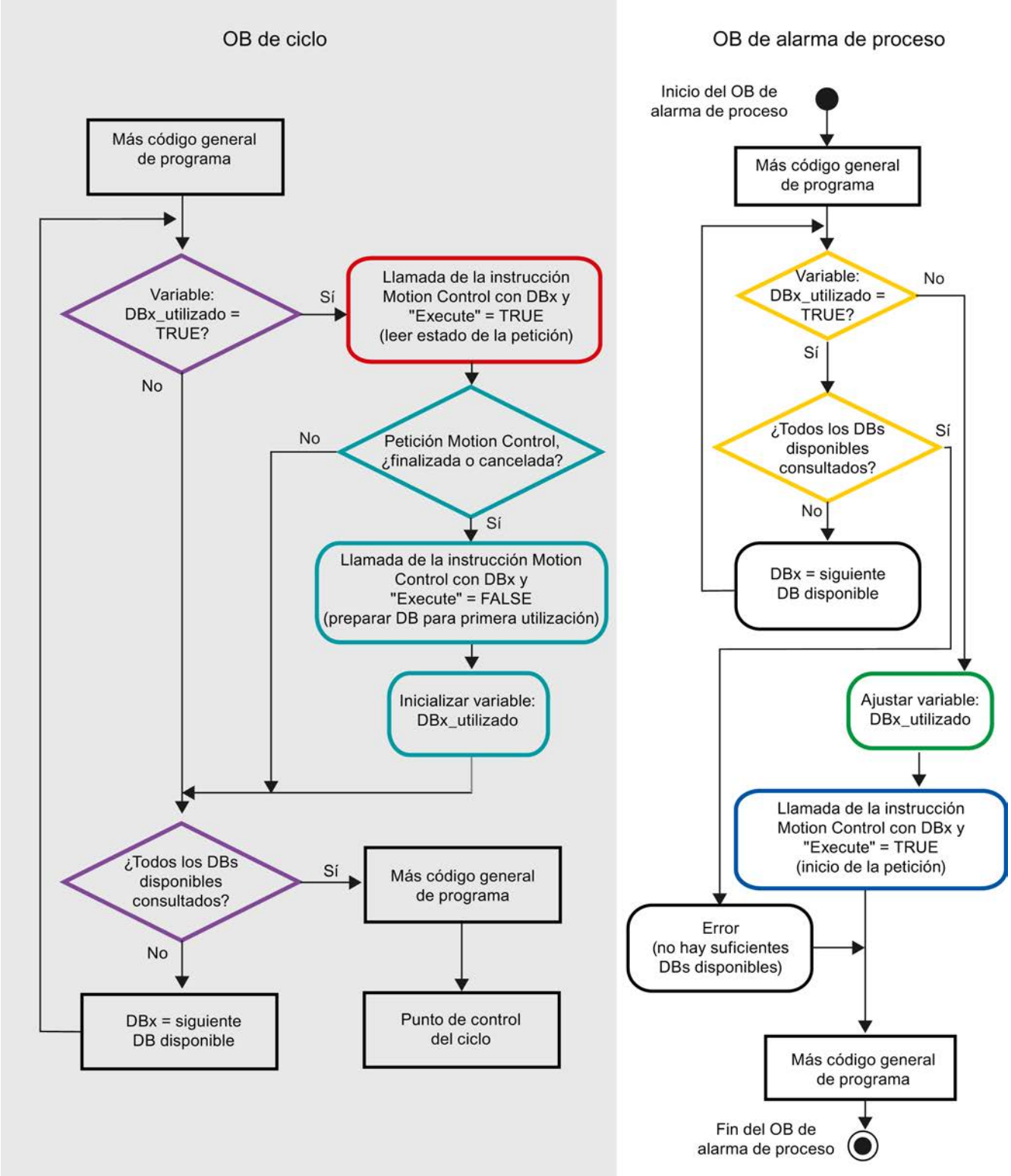
Casos especiales al utilizar finales de carrera por software (Página 10555)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

Realizar un seguimiento de las peticiones de clases de prioridad superiores (niveles de procesamiento)

Según la aplicación puede resultar necesario tener que iniciar peticiones de Motion Control (p.ej. con control de alarma) en una clase de prioridad superior.

Para el seguimiento de estado, las llamadas a las instrucciones de Motion Control se deben efectuar con poca distancia temporal. Si las instrucciones de Motion Control de la clase superior de prioridad se llaman una sola vez o a intervalos de tiempo muy dilatados, las peticiones de Motion Control no se pueden seguir suficientemente. En este caso existe la posibilidad de efectuar un seguimiento a posteriori en el OB de ciclo. Para cada inicio de una petición de Motion Control de la clase superior de prioridad debe existir un bloque de datos de instancia no utilizado actualmente. El siguiente diagrama de flujo muestra cómo iniciar peticiones de Motion Control en una clase de prioridad superior (p.ej. OB de alarma del proceso) y realizar un seguimiento a posteriori en el OB de ciclo.



En función de la frecuencia de las peticiones de Motion Control a iniciar se debe haber generado una cantidad suficiente de bloques de datos de instancia. En las variables

DBx_utilizado el usuario debe especificar el bloque de datos de instancia empleado actualmente.

Inicio de la petición de Motion Control en el OB de alarma de proceso

Mediante consultas binarias de las variables DBx_utilizado (naranja) se busca un bloque de datos de instancia no utilizado en ese instante. Si se encuentra, el bloque de datos de instancia utilizado se identifica como "utilizado" (verde) y la petición de Motion Control se inicia con este bloque de datos de instancia (azul).

A continuación se procesan eventualmente otras partes del programa del OB de alarma del proceso y posteriormente se regresa al OB de ciclo.

Seguimiento de las peticiones de Motion Control iniciadas en el OB de ciclo

En el OB de ciclo se comprueba con la variable DBx_utilizado (violeta) la utilización de todos los bloques de instancia disponibles.

Si un bloque de datos de instancia está siendo utilizado (petición de Motion Control en proceso), se procede a llamar la instrucción de Motion Control y el parámetro de entrada Execute = TRUE, para leer los avisos de estado (rojo).

Si la petición ha sido procesada o cancelada se continúa con las siguientes medidas (azul verdoso):

- Llamada de la instrucción de Motion Control con el parámetro de entrada Execute = FALSE
- Reajuste de la variable DBx_utilizado

Con ello finaliza el seguimiento de la petición y el bloque de datos de instancia vuelve a estar disponible para ser utilizado nuevamente.

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

Utilizando varios ejes con el mismo PTO (Página 10545)

Utilizar varios accionamientos con el mismo PTO (Página 10549)

Casos especiales al utilizar finales de carrera por software (Página 10555)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10563)

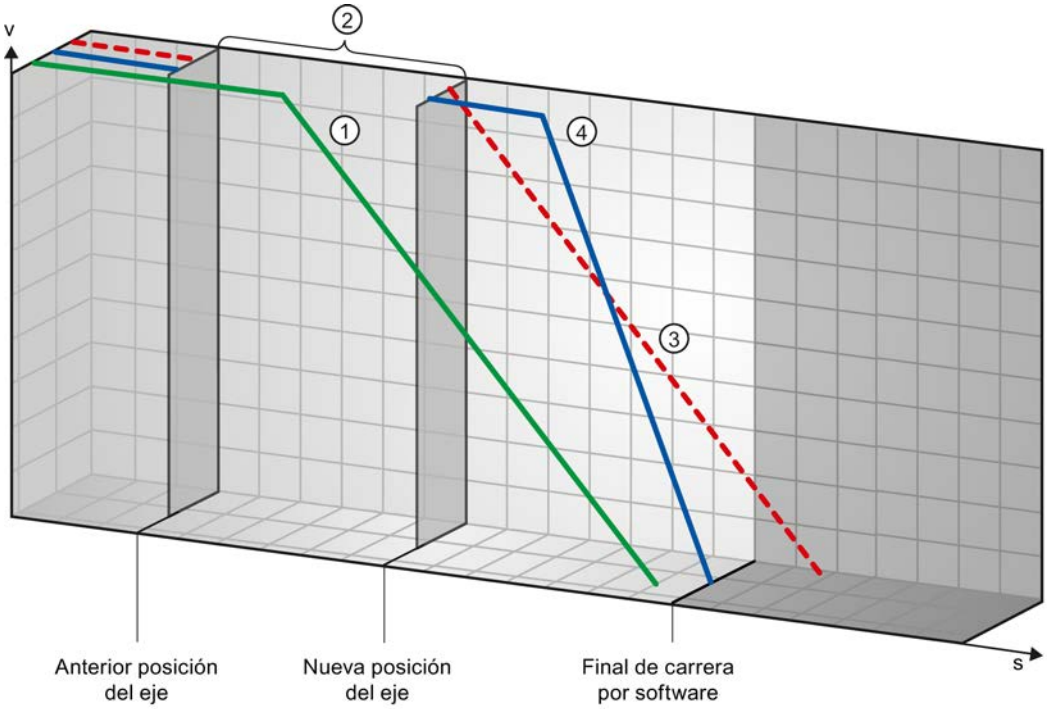
Casos especiales al utilizar finales de carrera por software

Finales de carrera por software y el proceso de referenciación

La parametrización desfavorable de peticiones de referenciación puede influir en la deceleración del eje en el final de carrera por software. Tenga en cuenta los siguientes ejemplos a la hora de realizar su programación.

Ejemplo 1:

Mientras se produce un comando de movimiento, la posición actual del eje se desplaza mediante una petición de referenciación (p.ej. fijar puntos de referencia) en el sentido del final de carrera por software. Aún es posible parar el eje hasta alcanzar el final de carrera por software:

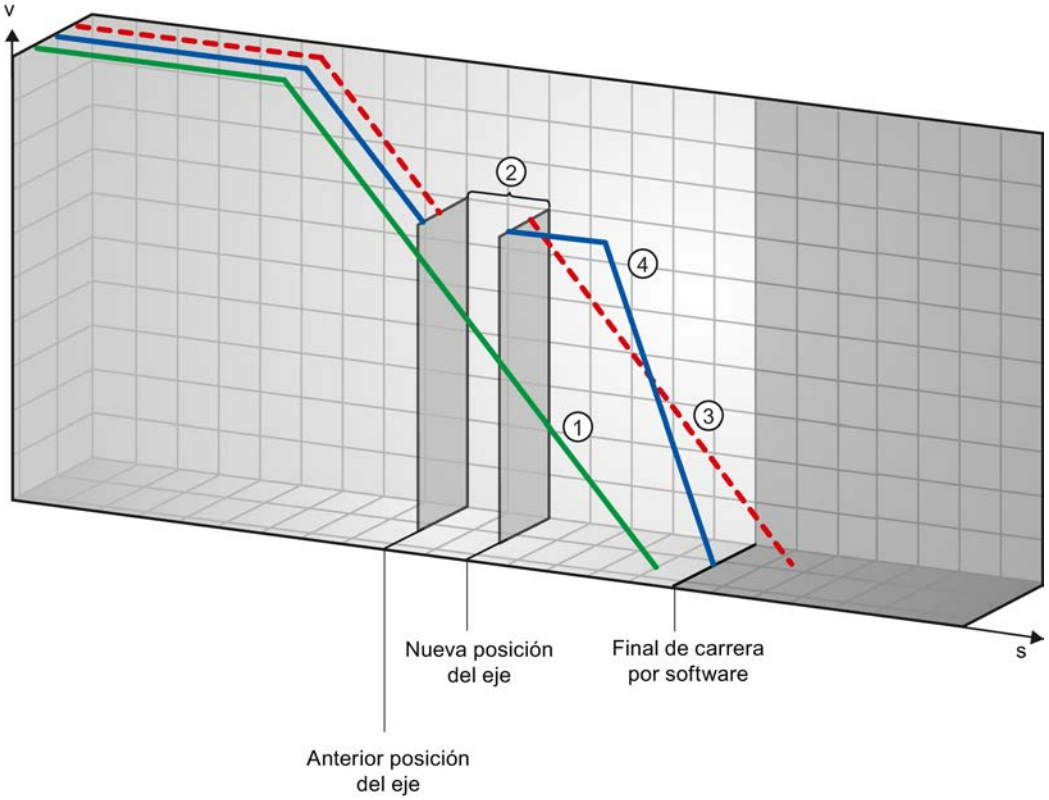


①	La curva verde muestra el movimiento **sin** petición de referenciación. El eje frena con la deceleración configurada, y se para antes de la posición del final de carrera por software.
②	Con la petición de referenciación se fija una nueva posición del eje. De esta forma se omite el área entre la antigua y la nueva posición del eje.
③	Condicionado por la nueva posición del eje, teóricamente el eje se pararía con la deceleración configurada detrás de la posición del final de carrera por software (curva roja).
④	Dado que ya no es posible frenar con la deceleración configurada, el eje se mueve realmente conforme a la curva azul. Después de moverse a una marcha constante, el eje frena con la deceleración de parada de emergencia configurada, y se para en la posición del final de carrera por software.

Ejemplo 2:

Mientras se produce un comando de movimiento, la posición actual del eje se desplaza mediante una petición de referenciación (p.ej. fijar puntos de referencia) en el sentido del final de carrera por software. Al contrario de lo que ocurre en el ejemplo 1, ya no es posible parar

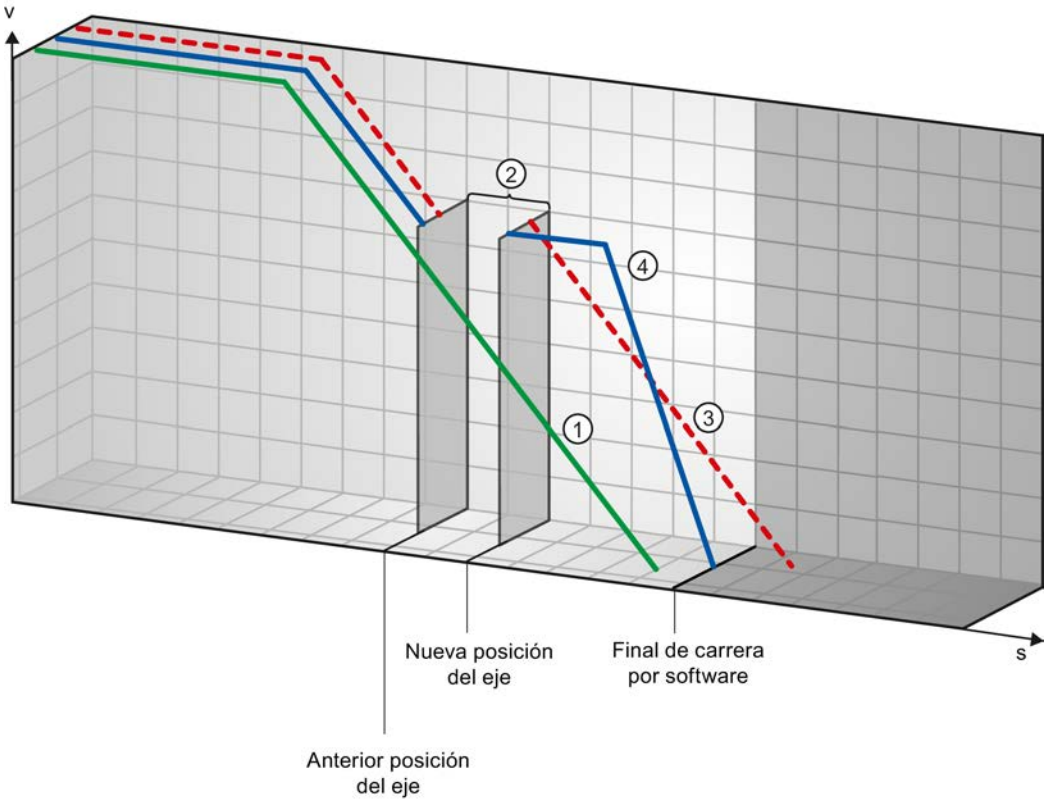
el eje hasta alcanzar el final de carrera por software. El eje traspasa la posición del final de carrera por software.



①	La curva verde muestra el movimiento **sin** petición de referenciación. El eje frena con la deceleración configurada, y se para antes de la posición del final de carrera por software.
②	Con la petición de referenciación se fija una nueva posición del eje. De esta forma se omite el área entre la antigua y la nueva posición del eje.
③	Condicionado por la nueva posición del eje, teóricamente el eje se pararía con la deceleración configurada muy por detrás de la posición del final de carrera por software (curva roja).
④	Dado que ya no es posible frenar con la deceleración configurada, el eje se mueve realmente conforme a la curva azul. El eje frena con la deceleración de parada de emergencia configurada. No obstante, la deceleración de parada de emergencia no es suficiente para parar el eje en la posición del final de carrera por software. El eje sobrepasa la posición del final de carrera por software.

Ejemplo 3:

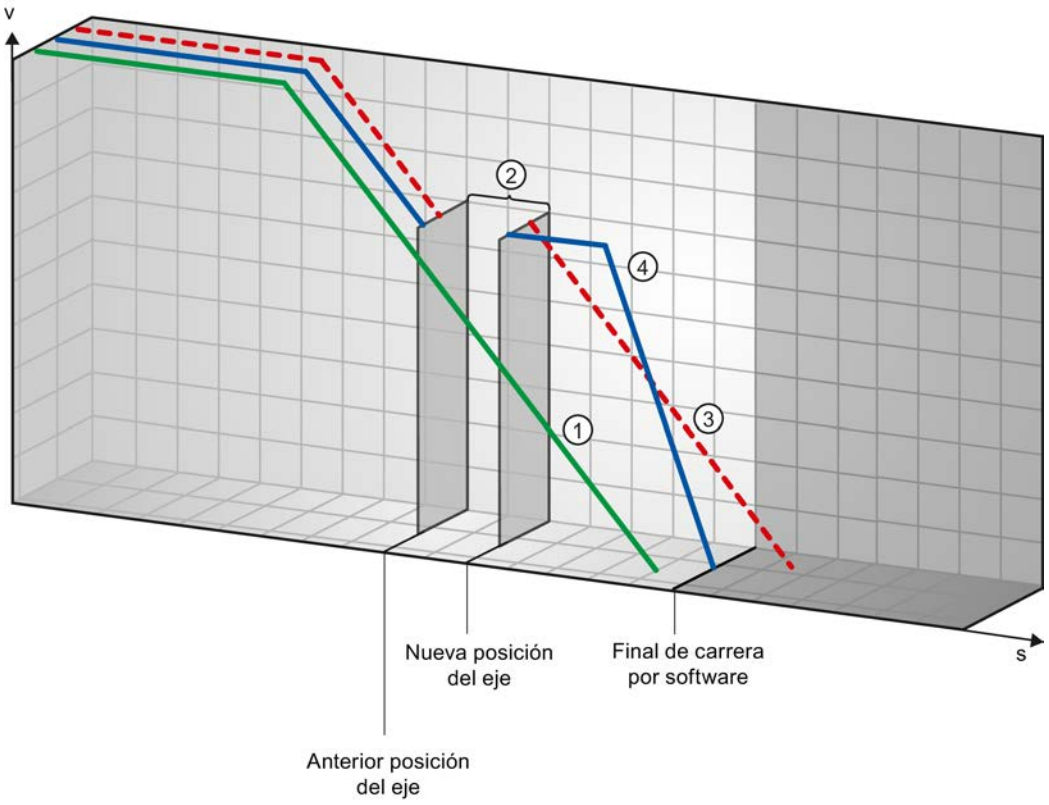
Mientras se produce la operación de frenado, la posición actual del eje se desplaza mediante una petición de referenciación (p.ej. fijar puntos de referencia) en el sentido del final de carrera por software. Aún es posible parar el eje hasta alcanzar el final de carrera por software:



①	La curva verde muestra el movimiento **sin** petición de referenciación. El eje frena con la deceleración configurada, y se para antes de la posición del final de carrera por software.
②	Con la petición de referenciación se fija una nueva posición del eje. De esta forma se omite el área entre la antigua y la nueva posición del eje.
③	Condicionado por la nueva posición del eje, teóricamente el eje se pararía con la deceleración configurada detrás de la posición del final de carrera por software (curva roja).
④	Dado que ya no es posible frenar con la deceleración configurada, el eje se mueve realmente conforme a la curva azul. Después de moverse a una marcha constante, el eje frena con la deceleración de parada de emergencia configurada, y se para en la posición del final de carrera por software.

Ejemplo 4:

Mientras se produce la operación de frenado, la posición actual del eje se desplaza mediante una petición de referenciación (p.ej. fijar puntos de referencia) en el sentido del final de carrera por software. Al contrario de lo que ocurre en el ejemplo 3, ya no es posible parar el eje hasta alcanzar el final de carrera por software. El eje traspasa la posición del final de carrera por software.



①	La curva verde muestra el movimiento **sin** petición de referenciación. El eje frena con la deceleración configurada, y se para antes de la posición del final de carrera por software.
②	Con la petición de referenciación se fija una nueva posición del eje. De esta forma se omite el área entre la antigua y la nueva posición del eje.
③	Condicionado por la nueva posición del eje, teóricamente el eje se pararía con la deceleración configurada muy por detrás de la posición del final de carrera por software (curva roja).
④	Dado que ya no es posible frenar con la deceleración configurada, el eje se mueve realmente conforme a la curva azul. El eje frena con la deceleración de parada de emergencia configurada. No obstante, la deceleración de parada de emergencia no es suficiente para parar el eje en la posición del final de carrera por software. El eje sobrepasa la posición del final de carrera por software.

Consulte también

Finales de carrera por software en relación con los cambios de posicionamiento del final de carrera por software. (Página 10560)

Final de carrera por software y modificaciones en la dinámica (Página 10560)

Comportamiento del eje al reaccionar las monitorizaciones de posición (Página 10462)

Finales de carrera por software en relación con los cambios de posicionamiento del final de carrera por software.

Con una modificación poco favorable de la posición del final de carrera por software mientras se ejecuta el programa de usuario, es posible acortar de forma abrupta la distancia entre la posición actual del eje y la posición del final de carrera por software.

La reacción del eje que se produce es similar a la descrita en el capítulo Finales de carrera por software y el proceso de referenciación (Página 10552).

Consulte también

Finales de carrera por software y el proceso de referenciación (Página 10552)

Final de carrera por software y modificaciones en la dinámica (Página 10560)

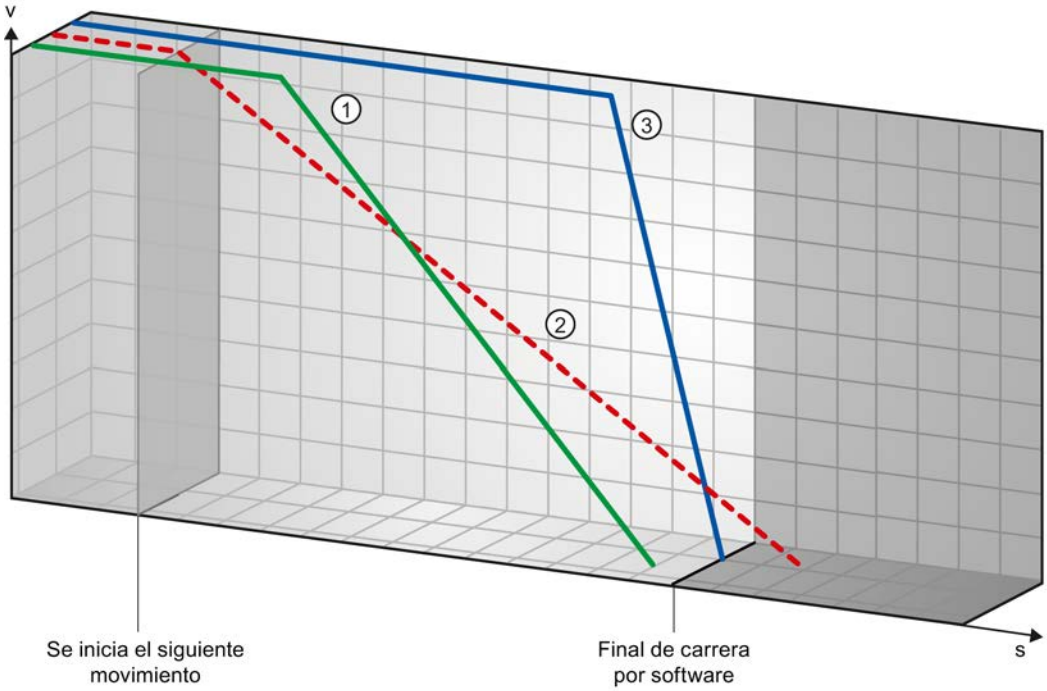
Comportamiento del eje al reaccionar las monitorizaciones de posición (Página 10462)

Final de carrera por software y modificaciones en la dinámica

En combinación con peticiones de movimiento reemplazantes se puede influir en la deceleración del eje en el área del final de carrera por software. Esto es posible cuando la petición de movimiento reemplazante se inicia con una deceleración inferior (Variable <Nombre de eje>.DynamicDefaults.Deceleration). Tenga en cuenta los siguientes ejemplos a la hora de realizar su programación.

Ejemplo 1:

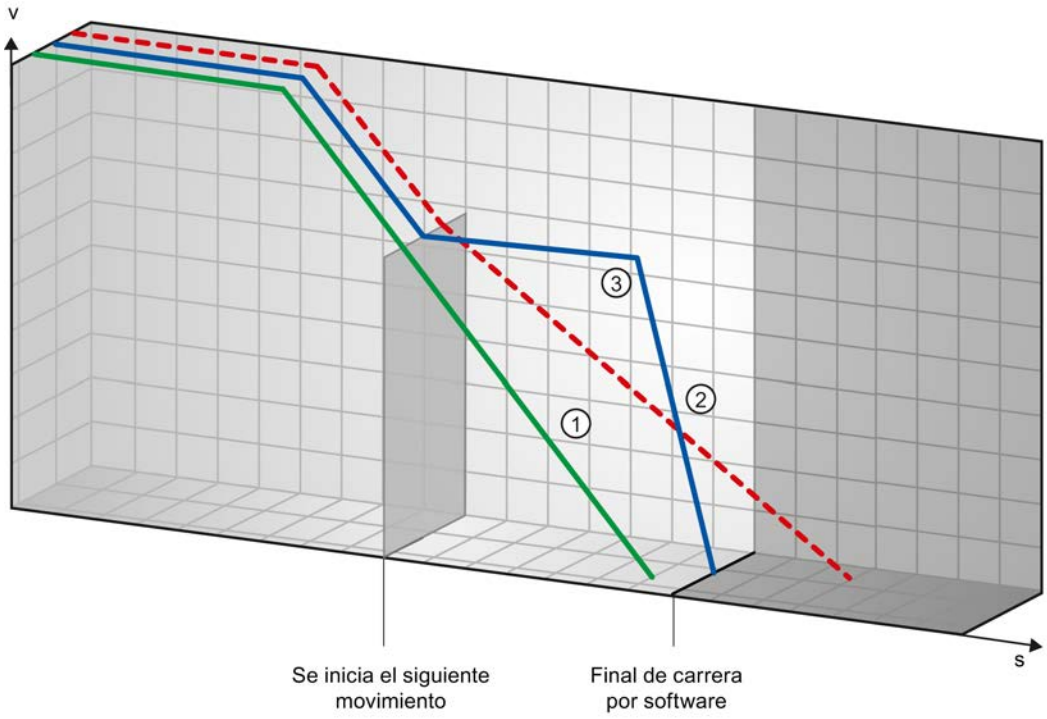
Mientras el eje se mueve se reemplaza una petición de marcha en curso por otra petición de movimiento con una deceleración inferior:



①	La curva verde muestra el movimiento de una petición en curso **sin** reemplazamiento. El eje frena con la deceleración configurada, parándose antes de la posición del final de carrera por software.
②	Condicionado por la petición de movimiento de reemplazo con una deceleración inferior, el eje se pararía teóricamente detrás de la posición del final de carrera por software (curva roja).
③	Dado que ya no es posible frenar con la deceleración configurada de la petición de movimiento de reemplazo, el eje se mueve realmente conforme a la curva azul. Después de moverse a una marcha constante, el eje frena con la deceleración de parada de emergencia, parándose en la posición del final de carrera por software.

Ejemplo 2:

Mientras el eje frena se reemplaza una petición de marcha en curso por otra petición de movimiento con una deceleración inferior:



①	La curva verde muestra el movimiento de una petición en curso **sin** reemplazamiento. El eje frena con la deceleración configurada, parándose antes de la posición del final de carrera por software.
②	Condicionado por la petición de movimiento de reemplazo con una deceleración inferior, el eje se pararía teóricamente muy por detrás de la posición del final de carrera por software (curva roja).
③	Dado que ya no es posible frenar con la deceleración configurada de la petición de movimiento de reemplazo, el eje se mueve realmente conforme a la curva azul. Después de moverse a una marcha constante, el eje frena con la deceleración de parada de emergencia, parándose en la posición del final de carrera por software.

Consulte también

Finales de carrera por software y el proceso de referenciación (Página 10552)

Finales de carrera por software en relación con los cambios de posicionamiento del final de carrera por software. (Página 10557)

Comportamiento del eje al reaccionar las monitorizaciones de posición (Página 10462)

Reducción de la velocidad para una duración breve de posicionamiento

Si la duración planificada del posicionamiento es < 2 ms, la velocidad de la petición de posicionamiento puede ser reducida por la CPU.

La petición se ejecuta con una velocidad reducida en todo el recorrido. La velocidad reducida (impulsos/s) se calcula con la fórmula siguiente:

- Velocidad reducida = Número de impulsos que deben emitirse * 500 Hz

Si la duración planificada del posicionamiento es >= 2 ms, no se efectúa ninguna reducción de la velocidad.

Adaptación dinámica de la velocidad de arranque/parada

Debido a la configuración de sus velocidades límite (velocidad de arranque/parada, velocidad máxima), de los valores dinámicos (aceleración, deceleración, tirón), así como de la velocidad de destino de la petición de desplazamiento, en determinadas circunstancias puede ocurrir que la CPU adapte la velocidad de arranque/parada dinámicamente.

Esto sucede cuando, p. ej., debido a que la velocidad de arranque/parada configurada es pequeña, el tiempo necesario para el primer impulso sería superior al permitido para toda la aceleración. En estos casos el primer impulso se emite a una velocidad superior a la velocidad de arranque/parada configurada. Los impulsos siguientes también se adaptan dinámicamente, de tal modo que el proceso de aceleración pueda finalizar en el tiempo especificado.

En caso de una posible pérdida de impulso, asegúrese de que el hardware (accionamiento) utilizado esté adaptado a esta situación, o bien cambie la configuración de la dinámica del eje para evitar que se produzca esta adaptación dinámica de la velocidad de arranque/parada.

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4)

Las siguientes tablas recogen una relación de todos los ErrorID y ErrorInfos que pueden ser mostrados en las instrucciones de Motion Control. Junto a la causa del error se muestran también ayudas para eliminar los errores:

Error de funcionamiento con parada del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8000		Error de accionamiento, fallo de "Accionamiento listo"	
	16#0001	-	Acusar error con la instrucción "MC_Reset"; proporcionar la señal del accionamiento; eventualmente iniciar de nuevo la petición
16#8001		Final de carrera por SW inferior activado	
	16#000E	Se ha alcanzado la posición del final de carrera por software inferior con la deceleración actual configurada	Acusar el error con la instrucción "MC_Reset"; salir del final de carrera por software con una petición de marcha en sentido positivo
	16#000F	Se ha alcanzado la posición del final de carrera por software inferior con la deceleración de parada de emergencia	
	16#0010	Se ha rebasado la posición del final de carrera por software inferior con la deceleración de parada de emergencia	

ErrorID	ErrorInfo	Descripción	Solución
16#8002		Final de carrera por SW superior activado	
	16#000E	Se ha alcanzado la posición del final de carrera por software superior con la deceleración actual configurada	Acusar el error con la instrucción "MC_Reset"; salir del final de carrera por software con una petición de marcha en sentido negativo
	16#000F	Se ha alcanzado la posición del final de carrera por software superior con la deceleración de parada de emergencia	
	16#0010	Se ha rebasado la posición del final de carrera por software superior con la deceleración de parada de emergencia	
16#8003		Final de carrera por HW inferior aproximado	
	16#000E	El final de carrera por hardware inferior ha sido aproximado. Se ha frenado el eje con la deceleración de parada de emergencia. (no se ha encontrado el sensor del punto de referencia durante una búsqueda activa del mismo)	Acusar el error en eje habilitado con la instrucción "MC_Reset"; salir del final de carrera por hardware con una petición de marcha en sentido positivo.
16#8004		Final de carrera por HW superior aproximado	
	16#000E	El final de carrera por hardware superior ha sido aproximado. Se ha frenado el eje con la deceleración de parada de emergencia. (no se ha encontrado el sensor del punto de referencia durante una búsqueda activa del mismo)	Acusar el error en eje habilitado con la instrucción "MC_Reset"; salir del final de carrera por hardware con una petición de marcha en sentido negativo.
16#8005		PTO y HSC ya están siendo utilizados por otro eje	
	16#0001	-	El eje ha sido configurado de forma incorrecta: Corregir la configuración del PTO (Pulse Train Output) / HSC (High Speed Counter) y cargarla en el controlador
			Varios ejes deben trabajar con un PTO: Otro eje utiliza el PTO / HSC. Si el eje actual debe asumir el control, se debe bloquear el otro eje con "MC_Power" Enable = FALSE. (Ver también Utilizando varios ejes con el mismo PTO (Página 10545))
16#8006		Se ha producido un error de comunicación en el panel de mando del eje	
	16#0012	Se ha producido un error de desbordamiento de tiempo.	Revise la conexión de cables y vuelva a hacer clic en el botón "Control manual".
16#8007		Imposible habilitar el eje	
	16#0025	Ejecutando restart.	Espere hasta que haya concluido el restart del eje.
	16#0026	Ejecutando carga en el estado operativo RUN	Espere hasta que haya concluido la operación de carga.
16#8008		Sentido de movimiento inadmisible	
	16#002E	El sentido de movimiento seleccionado no está permitido.	• Adapte el sentido de movimiento y reinicie la petición. • Adapte el sentido de giro permitido en la configuración del objeto tecnológico en "Parámetros avanzados > Mecánica". Reinicie la petición.
	16#002F	No es posible un movimiento invertido con el sentido seleccionado.	

Error de funcionamiento sin parada del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8200		El eje no está habilitado	
	16#0001	-	Habilitar el eje; volver a iniciar la petición
16#8201		El eje ya ha sido habilitado por otra instancia de "MC_Power"	
	16#0001	-	Habilitar el eje sólo con una instancia "MC_Power"
16#8202		Se ha superado el número máximo de peticiones de Motion Control simultáneamente activas (máx. 200 peticiones para todos los objetos tecnológicos de Motion Control)	
	16#0001	-	Reducir el número de peticiones activas simultáneamente; volver a iniciar la petición En el parámetro "Busy" = TRUE de la instrucción de Motion Control se reconoce una petición activa.
16#8203		El eje está funcionando en este momento en "Control manual" (panel de mando)	
	16#0001	-	Finalizar el "Control manual"; volver a iniciar la petición
16#8204		El eje no está referenciado	
	16#0001	-	Referenciar el eje con la instrucción "MC_Home"; volver a iniciar la petición
16#8205		El eje está siendo controlado por el programa de usuario (el error sólo se muestra en el panel de mando del eje)	
	16#0013	El eje está habilitado en el programa de usuario.	Bloquear eje con la instrucción "MC_Power" y seleccionar nuevamente "Control manual" en el panel de mando del eje
16#8206		Objeto tecnológico pendiente de activación	
	16#0001	-	Active el eje con la instrucción "MC_Power" Enable = TRUE, o habilite el eje en el panel de mando del eje.
16#8207		Petición rechazada	
	16#0016	Referenciación activa en curso; no es posible iniciar otro tipo de referenciación.	Espere a que la referenciación activa finalice o cancele la referenciación activa con una petición de movimiento, p. ej., "MC_Halt".
	16#0018	Mientras el eje se está referenciando de forma directa o pasiva, no se puede desplazar con una tabla de peticiones.	Espere a que la referenciación directa o pasiva termine.
	16#0019	Durante el procesamiento de una tabla de peticiones el eje no se puede referenciar de forma directa o pasiva.	Espere a que se cierre la tabla de peticiones o interrumpa dicha tabla con una petición de movimiento p. ej. "MC_Halt".
16#8208		La diferencia entre la velocidad máxima y la velocidad de arranque/parada no es válida	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#8209		El valor de aceleración del TO "Eje" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#820A		No es posible el restart del eje	
	16#0013	El eje está habilitado en el programa de usuario.	Bloquear eje con la instrucción "MC_Power"; ejecutar Restart nuevamente

ErrorID	ErrorInfo	Descripción	Solución
	16#0027	El eje está funcionando en este momento en "Control manual" (panel de mando)	Finalizar el "Control manual"; repetir Restart
16#820B		Imposible ejecutar la tabla de peticiones	
	16#0026	Ejecutando carga en el estado operativo RUN	Espere hasta que haya finalizado la operación de carga.

Error de parámetros de bloque

ErrorID	ErrorInfo	Descripción	Solución
16#8400		El valor del parámetro "Position" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8401		El valor del parámetro "Distance" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8402		El valor del parámetro "Velocity" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#0008	El valor es mayor que la velocidad máxima configurada	
	16#0009	El valor es inferior a la velocidad de arranque/ parada configurada	
	16#0024	El valor es menor que 0	
16#8403		El valor del parámetro "Direction" de la instrucción de Motion Control no es válido	
	16#0011	El valor de selección no es válido	Corregir el valor de selección; volver a iniciar la petición
16#8404		El valor del parámetro "Mode" de la instrucción de Motion Control no es válido	
	16#0011	El valor de selección no es válido	Corregir el valor de selección; volver a iniciar la petición
	16#0015	Referenciación activa / pasiva no configurada	Corregir la configuración y cargar en el controlador; habilitar el eje y reiniciar la petición
	16#0017	La inversión del sentido en el final de carrera por hardware está activada aunque los finales de carrera por hardware están desactivados.	• Activar los finales de carrera por hardware a través de la variable <Nombre de eje>.PositionLimitsHW.Active = TRUE, reiniciar petición • Corregir la configuración y cargar en el controlador; habilitar el eje y reiniciar la petición
16#8405		El valor del parámetro "StopMode" de la instrucción de Motion Control no es válido	
	16#0011	El valor de selección no es válido	Corregir el valor de selección; habilitar nuevamente el eje

ErrorID	ErrorInfo	Descripción	Solución
16#8406		Pulsación simultánea de avance y retroceso no permitida	
	16#0001	-	Evite la simultaneidad del estado de señal TRUE de los parámetros "JogForward" y "JogBackward"; inicie nuevamente la petición.
16#8407		Cambio del eje en la instrucción "MC_Power" permitido únicamente con eje bloqueado.	
	16#0001	-	Bloquear eje activo; a continuación se puede cambiar y habilitar el eje.
16#8408		El valor del parámetro "Axis" de la instrucción de Motion Control no es válido	
	16#001A	El valor indicado no se corresponde con la versión exigida del TO	Corregir el valor; volver a iniciar la petición
	16#001B	El valor indicado no se corresponde con el tipo exigido del TO	
	16#001C	El valor indicado no es un bloque de datos tecnológicos de Motion Control	
16#8409		El valor del parámetro "CommandTable" de la instrucción de Motion Control no es válido	
	16#001A	El valor indicado no se corresponde con la versión exigida del TO	Corregir el valor; volver a iniciar la petición
	16#001B	El valor indicado no se corresponde con el tipo exigido del TO	
	16#001C	El valor indicado no es un bloque de datos tecnológicos de Motion Control	
16#840A		El valor del parámetro "StartStep" de la instrucción de Motion Control no es válido	
	16#000A	El valor es menor o igual que cero.	Corregir el valor; volver a iniciar la petición
	16#001D	El paso de inicio es mayor que el paso final	
	16#001E	El valor es superior a 32	
16#840B		El valor del parámetro "EndStep" de la instrucción de Motion Control no es válido	
	16#000A	El valor es menor o igual que cero.	Corregir el valor; volver a iniciar la petición
	16#001E	El valor es superior a 32	
16#840C		El valor del parámetro "RampUpTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#840D		El valor del parámetro "RampDownTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#840E		El valor del parámetro "EmergencyRampTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#840F		El valor del parámetro "JerkTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#8410		El valor del parámetro "Parameter" de la instrucción de Motion Control no es válido	
	16#0028	El tipo de datos del puntero VARIANT "Parámetro" y "Valor" no son compatibles.	Utilizar el tipo de datos apropiado; volver a iniciar la petición
	16#0029	El puntero VARIANT "Parámetro" no señala un bloque de datos del objeto tecnológico.	Corregir el puntero VARIANT; volver a iniciar la petición

ErrorID	ErrorInfo	Descripción	Solución
	16#002A	No se puede leer el valor del puntero VARIANT "Parámetro".	Corregir el puntero VARIANT; volver a iniciar la petición
	16#002B	No se puede escribir el valor del puntero VARIANT "Parámetro".	Corregir el puntero VARIANT o el valor; volver a iniciar la petición
	16#002C	El eje no está bloqueado.	Bloquear el eje; volver a iniciar la petición
16#8411		El valor del parámetro "Value" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición

Error de configuración del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8600		Parametrización del generador de impulsos (PTO) no válida	
	16#000B	La dirección no es válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0014	El hardware seleccionado está en uso en otra aplicación	
16#8601		La parametrización del contador rápido (HSC) no es válida	
	16#000B	La dirección no es válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0014	El hardware seleccionado está en uso en otra aplicación	
16#8602		La parametrización de la "salida de habilitación" no es válida	
	16#000B	La dirección no es válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
16#8603		La parametrización de la "entrada de disponibilidad" no es válida	
	16#000B	La dirección no es válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
16#8604		El valor de "Impulsos por vuelta del motor" no es válido	
	16#000A	El valor es inferior o igual a cero	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
16#8605		El valor de "Recorrido por vuelta del motor" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#000A	El valor es inferior o igual a cero	
16#8606		El valor de "Velocidad de arranque/parada" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
	16#0007	La velocidad de arranque/parada es mayor que la velocidad máxima	
16#8607		El valor de "Velocidad máxima" no es válido	

ErrorID	ErrorInfo	Descripción	Solución
	16#0002	El valor no tiene un formato numérico válido	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#8608		El valor de "Aceleración" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#8609		El valor de "Deceleración" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#860A		El valor de "Deceleración de parada de emergencia" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#860B		El valor de posición del final de carrera por software inferior no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
	16#0030	El valor de posición del final de carrera por software inferior es mayor que el del superior	
16#860C		El valor de posición del final de carrera por software superior no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#860D		La dirección del final de carrera por hardware inferior no es válida	
	16#000B	Dirección no válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#000C	La dirección del flanco descendente no es válida	
	16#000D	La dirección del flanco ascendente no es válida	
16#860E		La dirección del final de carrera por hardware superior no es válida	

ErrorID	ErrorInfo	Descripción	Solución
	16#000B	Dirección no válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#000C	La dirección del flanco descendente no es válida	
	16#000D	La dirección del flanco ascendente no es válida	
16#860F		El valor de "Offset del punto de referencia" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8610		El valor de "Velocidad de aproximación" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8611		El valor de "Velocidad de referenciación" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8612		La dirección del sensor del punto de referencia no es válida	
	16#000B	Dirección no válida	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#000C	La dirección del flanco descendente no es válida	
	16#000D	La dirección del flanco ascendente no es válida	
16#8613		En la referenciación activa, la inversión del sentido en el final de carrera por hardware se encuentra activada, a pesar de que los finales de carrera por hardware no han sido configurados	
	16#0001	-	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
16#8614		El valor de "Tirón" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#001F	El valor es mayor que el tirón máximo admisible	
	16#0020	El valor es menor que el tirón mínimo admisible	
16#8615		El valor de "Unidad de medida" no es válido	
	16#0011	El valor de selección no es válido	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"

ErrorID	ErrorInfo	Descripción	Solución
16#8616		La dirección del sensor del punto de referencia no es válida (referenciación pasiva a partir de V4)	
	16#0011	El valor de selección no es válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
16#8617		El valor de la variable <Nombre de eje>.Sensor.Sensor[1].ActiveHoming.Mode no es válido	
	16#0011	El valor de selección no es válido (Valor válido: 2 = referenciación mediante entrada digital)	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
16#8618		El valor de la variable <Nombre de eje>.Sensor.Sensor[1].PassiveHoming.Mode no es válido	
	16#0011	El valor de selección no es válido (Valor válido: 2 = referenciación mediante entrada digital)	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
16#8619		El valor de la variable <Nombre de eje>.Actor.Type no es válido	
	16#0011	El valor de selección no es válido (Valor válido: 2 = conexión mediante interfaz de impulsos)	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
16#861A		El valor para "Sentido de giro permitido" no es válido	
	16#0011	El valor de selección no es válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#002D	"Ambos sentidos" no está permitido con salida de sentido desconectada	

Error de configuración de la tabla de peticiones

ErrorID	ErrorInfo	Descripción	Solución
16#8700		El valor de "Tipo de petición" en la tabla de peticiones no es válido	
	16#0001	-	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
16#8701		El valor de "Posición / recorrido " en la tabla de peticiones no es válido	

ErrorID	ErrorInfo	Descripción	Solución
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8702		**El valor de "Velocidad" en la tabla de peticiones no es válido**	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0008	El valor es mayor que la velocidad máxima configurada	
	16#0009	El valor es inferior a la velocidad de arranque/ parada configurada	
16#8703		**El valor de "Duración" en la tabla de peticiones no es válido**	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0021	El valor es superior a 64800 s	
	16#0022	El valor es inferior a 0.001 s	
16#8704		**El valor de "Siguiente paso" en la tabla de peticiones no es válido**	
	16#0011	El valor de selección no es válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0023	La transición de peticiones no está permitida para esta petición	

Errores internos

ErrorID	ErrorInfo	Descripción	Solución
16#8FFF		**Error interno**	
	16#F0**	-	POWER OFF y POWER ON de la CPU Si esto no produce el resultado deseado, póngase en contacto con el Servicio de Atención al Cliente. Tenga al alcance la siguiente información: • ErrorID • ErrorInfo • Registros del búfer de diagnóstico

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10573)

ErrorID y ErrorInfos (Página 10609)

Indicaciones de error de las instrucciones de Motion Control (Página 10540)

Utilizando varios ejes con el mismo PTO (Página 10545)

Utilizar varios accionamientos con el mismo PTO (Página 10549)

Realizar un seguimiento de las peticiones de clases de prioridad superiores (niveles de procesamiento) (Página 10550)

Casos especiales al utilizar finales de carrera por software (Página 10552)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4

Variable Position a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RCCP	La variable se puede leer en el programa de usuario y se actualiza en el punto de control del ciclo.
	RP	La variable se puede leer con la instrucción de Motion Control "MC_ReadParam". El valor actual de las correspondientes variables se determina al iniciarse la petición.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Position				
Posición actual del eje (Indicación en la unidad de medida configurada) Si el eje no está referenciado, la variable muestra el valor de posicionamiento relativo a la posición de habilitación del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RCCP, RP	-	X

Consulte también

Estado de movimiento (Página 10544)

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variable Velocity a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RCCP	La variable se puede leer en el programa de usuario y se actualiza en el punto de control del ciclo.
	RP	La variable se puede leer con la instrucción de Motion Control "MC_ReadParam". El valor actual de las correspondientes variables se determina al iniciarse la petición.

efectivo	Indica el momento en que una modificación de la variable resulta efectiva.
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).

<Nombre de eje>.Velocity				
Velocidad actual del eje (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RCCP, RP	-	X

Consulte también

Estado de movimiento (Página 10544)

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables Actor a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	2	Con la habilitación del eje
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Actor.Type				
Conexión del accionamiento El accionamiento se conecta mediante una interfaz de impulsos.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DInt	2	R	-	X

<Nombre de eje>.Actor.InverseDirection				
Invertir sentido de dirección FALSE = El sentido de dirección no se invierte. TRUE = El sentido de dirección se invierte.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X
		WP	2	-

<Nombre de eje>.Actor.DirectionMode				
Sentido de giro permitido 0 = Ambos sentidos 1 = Sentido positivo 2 = Sentido negativo				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	0	R	-	X
		WP	2	-

<Nombre de eje>.Actor.Interface.EnableDriveOutput				
Salida de habilitación (parámetro interno)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
VREF	-	-	-	-

<Nombre de eje>.Actor.Interface.DriveReadyInput				
Entrada de disponibilidad (parámetro interno)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
VREF	-	-	-	-

<Nombre de eje>.Actor.Interface.PTO				
Salida de impulso (parámetro interno)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	0	-	-	-

<Nombre de eje>.Actor.DriveParameter.PulsesPerDriveRevolution				
Pulsos por vuelta del motor				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DInt	1000	R	-	X
		WP	2	-

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables Sensor[1] a partir de V4

Variables ActiveHoming a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario. La variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	2	Con la habilitación del eje
	8	Al iniciar una petición de referenciación activa
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Sensor[1].ActiveHoming.Mode				
Modo referenciación activa 2 = referenciación mediante entrada digital				
Tipo de datos	**Valor inicial**	**Acceso**	**efectivo**	**HMI**
DInt	2	R	-	X
		WP	2	-

<Nombre de eje>.Sensor[1].ActiveHoming.SideInput				
Lado del sensor del punto de referencia que se utiliza con la referenciación activa FALSE = Lado inferior TRUE = Lado superior				
Tipo de datos	**Valor inicial**	**Acceso**	**efectivo**	**HMI**
Bool	FALSE	RW	1, 8	X

<Nombre de eje>.Sensor[1].ActiveHoming.DigitalInputAddress				
Dirección de entrada simbólica del sensor del punto de referencia (parámetro interno)				
Tipo de datos	**Valor inicial**	**Acceso**	**efectivo**	**HMI**
VREF	-	-	-	-

<Nombre de eje>.Sensor[1].ActiveHoming.HomePositionOffset				
Offset del punto de referencia (referenciación activa) (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RW	1, 8	X

<Nombre de eje>.Sensor[1].ActiveHoming.SwitchLevel				
Selección del nivel de señal existente en la entrada de la CPU con el sensor del punto de referencia aproximado FALSE = Nivel inferior (Low activo) TRUE = Nivel superior (High activo)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	TRUE	RW	1, 8	X

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables PassiveHoming a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario. La variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	2	Con la habilitación del eje
	7	Al iniciar una petición de referenciación pasiva
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Sensor[1].PassiveHoming.Mode				
Modo de referenciación pasiva 2 = referenciación mediante entrada digital				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DInt	2	R	-	X
		WP	2	-

<Nombre de eje>.Sensor[1].PassiveHoming.SideInput				
Lado del sensor del punto de referencia que se utiliza con la referenciación pasiva FALSE = Lado inferior TRUE = Lado superior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RW	1, 7	X

<Nombre de eje>.Sensor[1].PassiveHoming.DigitalInputAddress				
Dirección de entrada simbólica del sensor del punto de referencia (parámetro interno)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
VREF	-	-	-	-

<Nombre de eje>.Sensor[1].PassiveHoming.SwitchLevel				
Selección del nivel existente en la entrada de la CPU con el sensor del punto de referencia aproximado FALSE = Nivel inferior (Low activo) TRUE = Nivel superior (High activo)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	TRUE	RW	1, 7	X

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variable Units a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	2	Con la habilitación del eje
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Units.LengthUnit				
Unidad de medida configurada de los parámetros -1 = "Impulsos" 1005 = "°" (grados) 1013 = "mm" 1010 =: "m" 1018 = "ft" 1019 = "in"				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	1013	R	-	X
		WP	2	-

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variable Mechanics a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	2	Con la habilitación del eje
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Mechanics.LeadScrew				
Recorrido por vuelta del motor (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	10.0	R	-	X
		WP	2	-

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables DynamicLimits a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	2	Con la habilitación del eje
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.DynamicLimits.MaxVelocity				
Velocidad máxima del eje (Indicación en la unidad de medida configurada)				
Tipo de datos	**Valor inicial**	**Acceso**	**efectivo**	**HMI**
Real	250.0	R	-	X
		WP	2	-

<Nombre de eje>.DynamicLimits.MinVelocity				
Velocidad de arranque/parada del eje (Indicación en la unidad de medida configurada)				
Tipo de datos	**Valor inicial**	**Acceso**	**efectivo**	**HMI**
Real	10.0	R	-	X
		WP	2	-

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables DynamicDefaults a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario. La variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	

	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	5	Con el siguiente inicio de una petición MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt o MC_CommandTable o de una petición activa MC_Home (Mode = 3).
	6	Con la parada de una petición MC_MoveJog
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.DynamicDefaults.Acceleration				
Aceleración del eje (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	48.0	RW	1, 5, 6	X

<Nombre de eje>.DynamicDefaults.Deceleration				
Deceleración del eje (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	48.0	RW	1, 5, 6	X

<Nombre de eje>.DynamicDefaults.EmergencyDeceleration				
Deceleración de parada de emergencia del eje (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	120.0	RW	1, 5, 6	X

<Nombre de eje>.DynamicDefaults.Jerk				
Tirón durante la rampa de aceleración y deceleración del eje (Indicación en la unidad de medida configurada) El tirón está activado si el tirón configurado es superior a 0.00004 mm/s².				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	192.0	RW	1, 5	X

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variable PositionLimitsSW a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.

Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario. La variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	5	Con el siguiente inicio de una petición MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt o MC_CommandTable o de una petición activa MC_Home (Mode = 3).
	6	Con la parada de una petición MC_MoveJog
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.PositionLimitsSW.Active				
Activación de los finales de carrera por software FALSE = Los finales de carrera por software están desactivados. TRUE = Los finales de carrera por software están activados.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RW	1, 5, 6	X

<Nombre de eje>.PositionLimitsSW.MinPosition				
Posición del final de carrera por software inferior (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	-10000.0	RW	1, 5, 6	X

<Nombre de eje>.PositionLimitsSW.MaxPosition				
Posición del final de carrera por software superior (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	10000.0	RW	1, 5, 6	X

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variable PositionLimitsHW a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario. La variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".

	R	Se puede leer la variable en el programa de usuario.
	WP	Si el eje está bloqueado (MC_Power.Status = FALSE), la variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	2	Con la habilitación del eje
	5	Con el siguiente inicio de una petición MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt o MC_CommandTable o de una petición activa MC_Home (Mode = 3).
	6	Con la parada de una petición MC_MoveJog
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.PositionLimitsHW.Active				
Activación de los finales de carrera por hardware FALSE = Los finales de carrera por hardware están desactivados. TRUE = Los finales de carrera por hardware están activados.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RW	1, 5, 6	X

<Nombre de eje>.PositionLimitsHW.MinSwitchLevel				
Selección del nivel de señal existente en la entrada de la CPU con el final de carrera por hardware inferior aproximado FALSE = Nivel inferior (Low activo) TRUE = Nivel superior (High activo)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X
		WP	2	-

<Nombre de eje>.PositionLimitsHW.MinSwitchAddress				
Dirección de entrada simbólica del final de carrera por hardware inferior (parámetro interno)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
VREF	-	-	-	-

<Nombre de eje>.PositionLimitsHW.MaxSwitchLevel				
Selección del nivel de señal existente en la entrada de la CPU con el final de carrera por hardware superior aproximado FALSE = Nivel inferior (Low activo) TRUE = Nivel superior (High activo)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X
		WP	2	-

<Nombre de eje>.PositionLimitsHW.MaxSwitchAddress				
Dirección de entrada del final de carrera por hardware superior (parámetro interno)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
VREF	-	-	-	-

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables Homing a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario. La variable se puede escribir con la instrucción de Motion Control "MC_WriteParam".
efectivo	Indica el momento en que una modificación de la variable resulta efectiva:	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	8	Al iniciar una petición de referenciación activa
HMI	La variable se puede utilizar en un sistema HMI (conforme al acceso a la variable en el programa de usuario).	

<Nombre de eje>.Homing.AutoReversal				
Activación de la inversión de sentido en el final de carrera por hardware con la referenciación activa FALSE = La inversión de sentido en el final de carrera por hardware está desactivada. TRUE= La inversión de sentido en el final de carrera por hardware está activada.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RW	1, 8	X

<Nombre de eje>.Homing.ApproachDirection				
Sentido de aproximación y de referenciación del eje con la referenciación activa FALSE = sentido de aproximación negativo para la búsqueda del sensor del punto de referencia y sentido de referenciación negativo TRUE = sentido de aproximación positivo para la búsqueda del sensor del punto de referencia y sentido de referenciación positivo				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	TRUE	RW	1, 8	X

<Nombre de eje>.Homing.ApproachVelocity				
Velocidad de aproximación del eje con la referenciación activa (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	200.0	RW	1, 8	X

<Nombre de eje>.Homing.ReferencingVelocity				
Velocidad de referenciación del eje con la referenciación activa (Indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	40.0	RW	1, 8	X

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables StatusPositioning a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RCCP	La variable se puede leer en el programa de usuario y se actualiza en el punto de control del ciclo.
	RP	La variable se puede leer con la instrucción de Motion Control "MC_ReadParam". El valor actual de las correspondientes variables se determina al iniciarse la petición.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.StatusPositioning.Distance				
Distancia actual del eje hasta la posición de destino (Indicación en la unidad de medida configurada) El valor de la variable solo es válido durante la ejecución de una petición de posicionamiento con "MC_MoveAbsolute", "MC_MoveRelative" o del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RCCP, RP	-	X

<Nombre de eje>.StatusPositioning.TargetPosition				
Posición de destino del eje (Indicación en la unidad de medida configurada) El valor de la variable solo es válido durante la ejecución de una petición de posicionamiento con "MC_MoveAbsolute", "MC_MoveRelative" o del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RCCP, RP	-	X

Consulte también

Estado de movimiento (Página 10544)

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables StatusBits a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RCCP	La variable se puede leer en el programa de usuario y se actualiza en el punto de control del ciclo.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.StatusBits.Activated				
Activación del eje FALSE = El eje no está activado. TRUE = El eje está activado. Se encuentra conectado con el PTO asignado (Pulse Train Output). Los datos del bloque de datos tecnológico se actualizan cíclicamente.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.Enable				
Estado de habilitación del eje FALSE = el eje no está habilitado. TRUE = el eje está habilitado y listo para aceptar peticiones de Motion Control.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.HomingDone				
Estado de referenciación del eje FALSE = el eje no está referenciado. TRUE = el eje está referenciado y puede ejecutar peticiones de posicionamiento absolutas. Para realizar un posicionamiento relativo no es necesario que el eje esté referenciado. Durante la referenciación activa el estado es FALSE. Durante la referenciación pasiva, el estado TRUE se mantiene si el eje ya estaba referenciado previamente.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.Done				
Procesamiento de peticiones del eje FALSE = en el eje hay una petición de Motion Control activa. TRUE = en el eje no hay ninguna petición de Motion Control activa.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.Error				
Estado de error del eje FALSE = en el eje no hay ningún error. TRUE = se ha producido un error en el eje. Si requiere información más detallada sobre el error, consulte los parámetros "ErrorID" y "ErrorInfo" de las instrucciones Motion Control en el modo automático. En el modo manual, la causa detallada del error se muestra en el campo "Aviso de error" del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.StandStill				
Parada del eje FALSE = el eje está en movimiento. TRUE = el eje está parado.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.PositioningCommand				
Procesamiento de una petición de posicionamiento FALSE = no hay ninguna petición de posicionamiento activa en el eje. TRUE = el eje ejecuta una petición de posicionamiento de las instrucciones de Motion Control "MC_MoveRelative" o "MC_MoveAbsolute".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.VelocityCommand				
Procesamiento de una petición con velocidad predeterminada FALSE = en el eje no hay ninguna petición con velocidad predeterminada activa. TRUE = el eje ejecuta una petición de movimiento con velocidad predeterminada de las instrucciones de Motion Control "MC_MoveVelocity" o "MC_MoveJog".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.HomingCommand				
Procesamiento de una petición de referenciación FALSE = no hay ninguna petición de referenciación activa en el eje. TRUE = el eje ejecuta una petición de referenciación de la instrucción de Motion Control "MC_Home".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.CommandTableActive				
Procesamiento de una tabla de peticiones FALSE = no hay ninguna tabla de peticiones activa en el eje. TRUE = el eje se controla con la instrucción de Motion Control "MC_CommandTable".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.ConstantVelocity				
Velocidad constante FALSE = el eje se acelera, decelera o está parado. TRUE = se ha alcanzado la velocidad de consigna. El eje se mueve a velocidad constante.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.Accelerating				
Proceso de aceleración FALSE = el eje se decelera, avanza a velocidad constante o está parado. TRUE = el eje se acelera.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.Decelerating				
Proceso de deceleración FALSE = el eje se acelera, avanza a velocidad constante o está parado. TRUE = el eje se frena.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.ControlPanelActive				
Estado de activación del panel de mando del eje FALSE = El modo de operación "Modo automático" está activado. El programa de usuario tiene el control sobre el eje. TRUE = se ha activado el modo de operación "Control manual" en el panel de mando del eje. El panel de mando del eje tiene el control sobre el eje. El eje no puede ser controlado por el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.DriveReady				
Estado de operación del accionamiento FALSE = el accionamiento no está listo. Las consignas no se ejecutan. TRUE = el accionamiento está listo. Las consignas pueden ejecutarse.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.RestartRequired				
Es necesario reiniciar el eje FALSE = no es necesario reiniciar el eje. TRUE = los valores han sido modificados en la memoria de carga. Para cargar los valores en la memoria de trabajo con la CPU en el estado operativo RUN, es necesario reiniciar el eje. Utilice para ello la instrucción de Motion Control "MC_Reset".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.SWLimitMinActive				
Estado del final de carrera por software inferior FALSE = se cumple el rango de trabajo configurado del eje. TRUE = se ha aproximado o sobrepasado el final de carrera por software inferior.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.SWLimitMaxActive				
Estado del final de carrera por software superior FALSE = se cumple la zona de trabajo configurada. TRUE = se ha aproximado o sobrepasado el final de carrera por software superior.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.HWLimitMinActive				
Estado del final de carrera por hardware inferior FALSE = se cumple la zona de desplazamiento admisible configurada. TRUE = se ha aproximado o sobrepasado el final de carrera por hardware inferior.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.StatusBits.HWLimitMaxActive				
Estado del final de carrera por hardware superior FALSE = se cumple la zona de desplazamiento admisible configurada. TRUE = se ha aproximado o sobrepasado el final de carrera por hardware superior.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

Consulte también

Bits de estado y error (objetos tecnológicos a partir de V4) (Página 10541)

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables ErrorBits a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RCCP	La variable se puede leer en el programa de usuario y se actualiza en el punto de control del ciclo.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.ErrorBits.SystemFault				
Error interno de sistema				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.ErrorBits.ConfigFault				
Configuración incorrecta del eje				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.ErrorBits.DriveFault				
Error en el accionamiento. Fallo de la señal "Accionamiento listo".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.ErrorBits.SWLimit				
Final de carrera por software aproximado o sobrepasado				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.ErrorBits.HWLimit				
Final de carrera por hardware aproximado o sobrepasado				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.ErrorBits.DirectionFault				
Sentido de movimiento inadmisible				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

<Nombre de eje>.ErrorBits.HWUsed				
Un segundo eje utiliza el mismo PTO (Pulse Train Output) y está habilitado con "MC_Power".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RCCP	-	X

Consulte también

Bits de estado y error (objetos tecnológicos a partir de V4) (Página 10541)

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables ControlPanel a partir de V4

Las variables "ControlPanel" no contienen datos relevantes para el usuario. En el programa de usuario no se puede acceder a dichas variables.

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Variables Internal a partir de V4

Las variables "Internal" no contienen datos relevantes para el usuario. En el programa de usuario no se puede acceder a dichas variables.

Consulte también

Variables del objeto tecnológico Eje V1...3 (Página 10623)

Actualización de las variables del objeto tecnológico

Las informaciones de estado y error del eje mostradas en las variables del objeto tecnológico se actualizan respectivamente en el punto de control del ciclo.

Las modificaciones de valores en las variables de configuración no son efectivas de forma inmediata. Consulte en la descripción detallada de la variable correspondiente las condiciones bajo las cuales una modificación resulta efectiva.

Variables del objeto tecnológico Tabla de peticiones a partir de V4

Variables Command[1...32] a partir de V4

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración de la tabla de peticiones.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Tabla de peticiones>.Command[n].Type				
Tipo de petición • 0 = Petición "Vacía" • 2 = Petición "Parada" • 5 = Petición "Positioning Relative" • 6 = Petición "Positioning Absolute" • 7 = Petición "Velocity setpoint" • 151 = Petición "Wait"				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	0	RW	-	X

<Tabla de peticiones>.Command[n].Position				
Posición de destino/recorrido de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RW	-	X

<Nombre de eje>.Command[n].Velocity				
Velocidad de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RW	-	X

<Tabla de peticiones>.Command[n].Duration				
Duración de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RW	-	X

<Tabla de peticiones>.Command[n].NextStep				
Modo para la transición a la siguiente petición • 0 = "Finalizar petición" • 1 = "Suavizar transición del movimiento"				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	0	RW	-	X

<Tabla de peticiones>.Command[n].StepCode				
Código de paso de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Word	0	RW	-	X

Consulte también

Variables del objeto tecnológico Tabla de peticiones V1...3 (Página 10638)

Versiones V1...3

Salidas de la CPU relevantes para Motion Control (versión tecnológica V1...3)

El número de accionamientos utilizables depende de la CPU, del número de PTO (Pulse Train Outputs) y del número de salidas disponibles del generador de impulsos.

Las siguientes representaciones proporcionan información sobre las correspondientes dependencias:

Número máximo de PTO

El número máximo de PTO (accionamientos) controlables depende de la referencia de la CPU:

Referencia CPU	Número de PTO
xxxxxxx-1xx**30**-xxxx	2
xxxxxxx-1xx**31**-xxxx	4

El número máximo de PTO (accionamientos) controlables es válido con independencia del uso de una Signal Board.

Salidas utilizables del generador de impulsos

La CPU dispone de una salida de impulsos o una salida de sentido para controlar un motor paso a paso o un servomotor con interfaz de impulsos. El accionamiento recibe a través de la salida de impulsos los impulsos necesarios para mover el motor. La salida de sentido controla el sentido de desplazamiento del accionamiento.

La salida de impulsos y la salida de sentido se encuentran firmemente asignadas entre sí y conforman una salida del generador de impulsos. Como salidas del generador de impulsos se pueden utilizar salidas integradas de la CPU o bien salidas de una Signal Board. Puede

elegir entre las salidas integradas de la CPU y las salidas de la Signal Board en la configuración de dispositivos, bajo Generadores de impulsos (PTO/PWM) en la rúbrica "Propiedades".

La tabla siguiente muestra el número de accionamientos utilizables por CPU o Signal Board:

CPU		Integrada	Signal Board				
			DI2/DO2 x DC24V 20kHz	DI2/DO2 x DC24V 200kHz	DO4 x DC24V 200kHz	DI2/DO2 x DC5V 200kHz	DO4 x DC5V 200kHz
CPU 1211C, CPU 1212C, CPU 1214C (MLFB - referencia xxxxxxx-1xx**30**-xxxx)	DC/DC/DC	2	2	2	2	2	2
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1211C (MLFB - referencia xxxxxxx-1xx**31**-xxxx)	DC/DC/DC	2	3	3	4	3	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1212C (MLFB - referencia xxxxxxx-1xx**31**-xxxx)	DC/DC/DC	3	4	4	4	4	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1214C (MLFB - referencia xxxxxxx-1xx**31**-xxxx)	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2
CPU 1215C	DC/DC/DC	4	4	4	4	4	4
	AC/DC/RLY	-	1	1	2	1	2
	DC/DC/RLY	-	1	1	2	1	2

La asignación posible de direcciones de las salidas de impulsos y sentido se recoge en la siguiente tabla:

CPU S7-1200		Salidas PTO1 *)		Salidas PTO2 **)		Salidas PTO3 *)		Salidas PTO4 **)	
		Imp.	Sentido	Imp.	Sentido	Imp.	Sentido	Imp.	Sentido
CPU 1211C, CPU 1212C, CPU 1214C, CPU1215C (DC/DC/DC)	CPU	Ax.0	Ax.1	Ax.2	Ax.3	Ax.4	Ax.5	Ax.6	Ax.7
	Signal Board	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C (AC/DC/ RLY)	CPU	-	-	-	-	-	-	-	-
	Signal Board	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3
CPU 1211C, CPU 1212C, CPU 1214C, CPU 1215C (DC/DC/ RLY)	CPU	-	-	-	-	-	-	-	-
	Signal Board	Ay.0	Ay.1	Ay.2	Ay.3	Ay.0	Ay.1	Ay.2	Ay.3

x = dirección inicial de byte de las salidas On-board de la CPU (valor estándar = 0)

y = dirección inicial de byte de las salidas Signal Board (valor estándar = 4)

* Si se utiliza una variante de CPU DC/DC/DC con una Signal Board DI2/DO2, las señales del PTO1/3 pueden ser emitidas a través de las salidas integradas de la CPU o a través de la Signal Board.

** Si se utiliza una variante de CPU DC/DC/DC con una Signal Board DO4, tanto las señales del PTO1/3 como las del PTO2/4 pueden ser emitidas a través de las salidas integradas de la CPU o a través de la Signal Board.

PTO3 y PTO4 solamente están disponibles para CPU con la referencia xxxxxxx-1xx**31**-xxxx.

Nota

Acceso a las salidas del generador de impulsos a través de la memoria imagen de proceso

Si se ha activado el PTO (Pulse Train Output) y se ha asignado a un eje, el firmware asume el control sobre las correspondientes salidas del generador de impulsos.

Al asumir el control se interrumpe también la conexión entre la memoria imagen de proceso y la salida de periferia. Si bien el usuario tiene la posibilidad de escribir en la memoria imagen de proceso de las salidas del generador de impulsos con el programa de usuario o la tabla de observación, esta no se transmite a la salida de periferia. Por consiguiente, tampoco es posible observar la salida de periferia a través del programa de usuario o la tabla de observación. Las informaciones leídas reflejan el valor de la memoria imagen de proceso, las cuales no coinciden con el estado verdadero de la salida de periferia.

En las demás salidas de la CPU no asignadas fijamente por el firmware de la CPU, el estado de la salida de periferia puede ser controlado u monitorizado de la forma habitual a través de la memoria imagen de proceso.

Frecuencias límite de las salidas de impulsos

Para las salidas de impulsos son válidas las siguientes frecuencias límite:

Salida de impulso	Frecuencias límite para objeto tecnológico Eje de posicionamiento V1	Frecuencias límite del objeto tecnológico Eje de posicionamiento V2/V3 para la CPU < V3.0	Frecuencias límite del objeto tecnológico Eje de posicionamiento V2/V3 para la CPU V3.0
On-board (MLFB - referencia xxxxxxx-1xx**30**-xxxx)	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 100 kHz	1 Hz ≤ f ≤ 100 kHz
On-board (MLFB - referencia xxxxxxx-1xx**31**xxxx)	2 Hz ≤ f ≤ 100 kHz (PTO 1+2) 2 Hz ≤ f ≤ 20 kHz (PTO 3+4)	2 Hz ≤ f ≤ 100 kHz (PTO 1+2) 2 Hz ≤ f ≤ 20 kHz (PTO 3+4)	1 Hz ≤ f ≤ 100 kHz (PTO 1+2) 1 Hz ≤ f ≤ 20 kHz (PTO 3+4)
Signal Board DI2/DO2 x DC24V 20kHz	2 Hz ≤ f ≤ 20 kHz	2 Hz ≤ f ≤ 20 kHz	1 Hz ≤ f ≤ 20 kHz
Signal Board DI2/DO2 x DC24V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
Signal Board DO4 x DC24V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
Signal Board DI2/DO2 x DC5V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz
Signal Board DO4 x DC5V 200kHz	2 Hz ≤ f ≤ 100 kHz	2 Hz ≤ f ≤ 200 kHz	1 Hz ≤ f ≤ 200 kHz

Señales del accionamiento

Para Motion Control puede parametrizar opcionalmente una interfaz de accionamiento para "Accionamiento habilitado" y "Accionamiento listo". Si se utiliza la interfaz de accionamiento es posible seleccionar libremente la salida digital para el accionamiento habilitado y la entrada digital para "Accionamiento listo".

Límites de aceleración/deceleración

Para la aceleración y deceleración se aplican los siguientes límites:

Aceleración/deceleración	Valor (CPU < V3.0)	Valor (CPU V3.0)
Aceleración/deceleración mínimas	2,8E-1 pulsos/s^2	5,0E-3 pulsos/s^2
Aceleración/deceleración máximas	9,5E+9 pulsos/s^2	9,5E+9 pulsos/s^2

Límites de tirón

Para el tirón son válidos los siguientes límites:

Tirón	Valor (CPU < V3.0)	Valor (CPU V3.0)
Tirón mínimo	4,0E-2 pulsos/s^3	4,0E-3 pulsos/s^3
Tirón máximo	1,5E+8 pulsos/s^3	1,0E+10 pulsos/s^3

Consulte también

Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)

Diálogos de configuración

Configuración - General (objeto tecnológico "Eje" V1...3)

Configure las propiedades básicas del objeto tecnológico "Eje" en la ventana de configuración "General".

Nombre del eje:

Defina en este campo el nombre del eje o del objeto tecnológico "Eje". El objeto tecnológico aparece con ese nombre en el árbol del proyecto.

Hardware - Interfaz

Los impulsos se transmiten a la etapa de potencia del accionamiento a través de salidas digitales de asignación fija.

En CPUs con salidas de relé la señal de impulso no puede emitirse en ellas, ya que los relés no soportan las frecuencias de conmutación necesarias. Para poder utilizar el PTO en esas CPUs (Pulse Train Output), es necesario utilizar una Signal Board con salidas digitales.

Nota

El PTO requiere la funcionalidad de un contador rápido (HSC). Para la CPU versión < V3.0 se utiliza para ello un HSC que seguidamente dejará de estar a disposición del usuario. Para la CPU versión ≥ V3.0 se utiliza para ello un HSC interno.

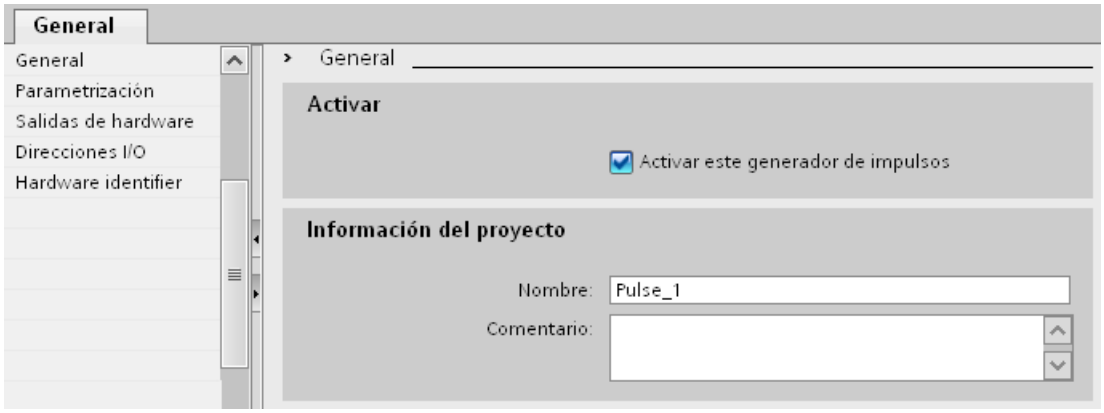
El estado del contador no puede evaluarse por medio de su dirección de entrada.

La asignación entre PTO y HSC es fija. Si el usuario activa el PTO1, éste queda conectado al HSC1. Al activar el PTO2, éste queda conectado al HSC2.

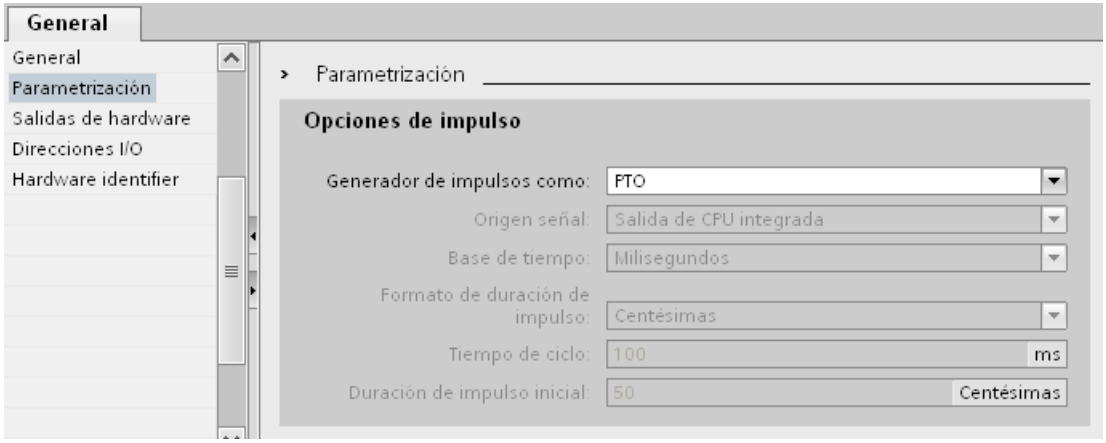
En la lista desplegable "Selección del generador de impulsos" elija el PTO (Pulse Train Output) a través del cual deben proporcionarse los impulsos para controlar los motores paso a paso o los servomotores con interfaz de impulsos. Si en la configuración de dispositivos no se han utilizado los generadores de impulso ni los contadores rápidos para otro fin, la interfaz por hardware podrá configurarse automáticamente. En este caso, el PTO elegido aparece marcado en blanco en la lista desplegable. En los campos de salida "Fuente de salidas", "Salida de impulso", "Salida de sentido" y "Contador rápido asignado" se enumeran las interfaces utilizadas.

Proceda tal como se indica a continuación si desea modificar las interfaces o si el PTO no se ha podido configurar automáticamente (la entrada de la lista desplegable "Selección del generador de impulsos" aparece con fondo rojo):

1. Haga clic en el botón "Configuración de dispositivos".
   La configuración de dispositivos del generador de impulsos se abre.
   Amplíe el área de propiedades de la configuración de dispositivos si la configuración del generador de impulsos no resulta visible.



2. Active la casilla de verificación "Activar este generador de impulsos".
3. Seleccione la entrada "Parametrización" en el navegador local.
   La "Parametrización" se abre.



4. En la lista desplegable elija "Generador de impulsos como:" la entrada "PTO".
5. En la lista desplegable elija "Origen señal:" la entrada "Salida de CPU integrada" o "Salida de la Signal Board". La entrada "Salida de la Signal Board" será seleccionable únicamente para PTO1, o bien para PTO1 y PTO2 según la Signal Board empleada. Para más información consulte el capítulo: Salidas de la CPU relevantes para Motion Control (versión tecnológica V4) (Página 10431)
6. Regrese a la configuración del eje.
   Si el contador rápido correspondiente no se ha utilizado ya de otro modo, los campos de PTO de la configuración del eje "General" no están resaltados sobre un fondo rojo. Si no es el caso, corrija la configuración con ayuda de los avisos de error.

Unidad personalizada

En la lista desplegable, seleccione la unidad deseada para el sistema de medida del eje. La unidad seleccionada se utiliza para una nueva configuración del objeto tecnológico "Eje" y para la visualización de los datos actuales del eje.

Los valores en los parámetros de entrada (Position, Distance, Velocity, ...) de las instrucciones de Motion Control también se refieren a esta unidad.

Nota

En determinadas circunstancias, un cambio posterior del sistema de medida no podrá convertirse correctamente en todas las ventanas de configuración del objeto tecnológico. En ese caso, compruebe la configuración de todos los parámetros del eje.

Si fuera necesario, en el programa de usuario se deberán adaptar los valores de los parámetros de entrada de las instrucciones de Motion Control a la nueva unidad de medida.

Consulte también

Configuración - General (objeto tecnológico Eje de posicionamiento a partir de V4) (Página 10456)

Cambio de configuración de los valores dinámicos en el programa de usuario (objeto tecnológico "Eje" V1...3)

Los siguientes parámetros de configuración pueden modificarse en la CPU mientras se ejecuta el programa de usuario:

Aceleración y deceleración

Puede modificar los valores de la aceleración y deceleración incluso durante el tiempo de ejecución del programa de usuario. Para ello utilice las siguientes variables del objeto tecnológico:

- <Nombre de eje>.Config.DynamicDefaults.Acceleration
  para modificar la aceleración
- <Nombre de eje>.Config.DynamicDefaults.Deceleration
  para modificar la deceleración

Para saber cuándo tienen efecto las modificaciones de los parámetros de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico (Página 10623).

Deceleración de parada de emergencia

Puede modificar el valor de la deceleración de parada de emergencia incluso durante el tiempo de ejecución del programa de usuario. Para ello utilice la siguiente variable del objeto tecnológico:

- <Nombre de eje>.Config.DynamicDefaults.EmergencyDeceleration

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico.

ATENCIÓN
Una vez modificado este parámetro puede resultar necesario tener que adaptar las posiciones de los finales de carrera de hardware, así como otras configuraciones relevantes para la seguridad.

Limitación de tirones (a partir del objeto tecnológico Eje V2.0)

También puede activar y desactivar la limitación de tirones durante el tiempo de ejecución del programa de usuario y modificar el valor del tirón. Para ello utilice las siguientes variables del objeto tecnológico:

- <Nombre de eje>.Config.DynamicDefaults.JerkActive
  para activar y desactivar la limitación de tirones
- <Nombre de eje>.Config.DynamicDefaults.Jerk
  para modificar el tirón

Para saber cuándo tienen efecto las modificaciones del parámetro de configuración, consulte en el anexo la descripción de las variables del objeto tecnológico.

Consulte también

Modificar la configuración de los valores de dinámica en el programa de usuario (Página 10471)

Configuración - Referenciar (objeto tecnológico "Eje" V1)

En la ventana de configuración "Referenciación", configure los parámetros para la referenciación activa y pasiva. El tipo de referenciación se ajusta a través del parámetro de entrada "Mode" de la instrucción de Motion Control. Aquí el valor Mode = 2 se corresponde con la referenciación pasiva y Mode = 3 con la referenciación activa.

Entrada del sensor del punto de referencia

Seleccione la entrada digital para el sensor del punto de referencia en la lista desplegable. La entrada debe ser apta para alarmas. Como entradas para el sensor del punto de referencia están disponibles las entradas de CPU On-board y las entradas de una Signal Board insertada.

Nota

Las entradas digitales están ajustadas de forma estándar con un tiempo de filtración de 6,4 ms.

La utilización como sensor del punto de referencia puede provocar deceleraciones no deseadas y, con ello, imprecisiones. En determinadas circunstancias, y dependiendo de la velocidad de referenciación y la dimensión del sensor del punto de referencia, puede que no se reconozca el punto de referencia. El tiempo de filtración se puede ajustar en la configuración de dispositivos de las entradas digitales, en "Filtros de entrada".

Se debe seleccionar un tiempo de filtración menor que la duración de la señal de entrada en el sensor del punto de referencia.

Permitir inversión del sentido en el final de carrera por hardware (sólo referenciación activa)

Active esta casilla de verificación si desea utilizar los finales de carrera por hardware como levas de inversión para la aproximación al punto de referencia. Los finales de carrera por hardware deben estar activados para poder realizar la inversión del sentido. Si se utiliza el firmware V1.0 de la CPU, ambos finales de carrera por hardware deberán estar configurados. Si se utiliza el firmware V2.0 de la CPU, sólo deberá estar configurado en el sentido de aproximación el final de carrera por hardware.

Si se llega al final de carrera por hardware durante la referenciación activa, el eje frena con la deceleración ajustada (no con la deceleración de parada de emergencia) y realiza una inversión del sentido. Entonces el sensor del punto de referencia se busca en la dirección contraria.

Si esta inversión del sentido no está activada y el eje alcanza el final de carrera por hardware durante la referenciación activa, la aproximación al punto de referencia se cancela con un error y el eje se frena con la deceleración de parada de emergencia.

Nota

Asegúrese de que la máquina no golpee contra un tope mecánico durante una inversión del sentido, adoptando para ello una de las siguientes medidas:

- Mantenga una velocidad baja de aproximación
- Incremente la aceleración / deceleración configurada
- Incremente la distancia entre el final de carrera por hardware y el tope de hardware

Sentido de aproximación / referenciación (referenciación activa y pasiva)

Con el sentido seleccionado, determine el "sentido de aproximación" durante la referenciación activa para buscar el sensor del punto de referencia, así como el sentido de referenciación.

El sentido de referenciación determina el sentido con el cual el eje se aproxima al lado del sensor del punto de referencia configurado para realizar la operación de referenciación.

Consulte el efecto del sentido de aproximación ajustado para la referenciación pasiva en la tabla que encontrará en "Sensor del punto de referencia".

Lado del sensor del punto de referencia (referenciación activa y pasiva)

- **Referenciación activa**
  Indique aquí si el eje debe ser referenciado por el lado superior o inferior del sensor del punto de referencia.

  Nota

  Dependiendo de la posición inicial del eje y de la configuración de los parámetros de referenciación, la aproximación al punto de referencia puede diferir del gráfico mostrado en la ventana de configuración.

- **Referenciación pasiva**
  En la referenciación pasiva, los movimientos para la referenciación deben ser realizados por el usuario mediante peticiones de marcha. El lado del sensor del punto de referencia por el cual se realiza la referenciación depende de los siguientes factores:
  - Configuración "sentido de aproximación"
  - Configuración "sensor del punto de referencia"
  - Sentido actual de marcha durante la referenciación pasiva

  La siguiente tabla muestra detalles del efecto de los factores:

Factores de influencia:			Resultado:
Configuración - sentido de aproximación	**Configuración - sensor del punto de referencia**	**Sentido actual de marcha**	**Referenciación en el sensor del punto de referencia**
positivo	"Lado inferior"	sentido positivo	**Lado superior**
		sentido negativo	**Lado inferior**
positivo	"Lado superior"	sentido positivo	**Lado inferior**
		sentido negativo	**Lado superior**
negativo	"Lado inferior"	sentido positivo	**Lado inferior**
		sentido negativo	**Lado superior**
Negativo	"Lado superior"	sentido positivo	**Lado superior**
		sentido negativo	**Lado inferior**

Velocidad (sólo referenciación activa)

Especifique en este campo la velocidad con la que se buscará el sensor del punto de referencia durante la marcha correspondiente.

Límites (independientes de la unidad de medida seleccionada):

- Velocidad de arranque/parada ≤ velocidad de aproximación ≤ velocidad máxima

Velocidad de referenciación (sólo referenciación activa)

Especifique en este campo la velocidad con la que el eje se posicionará en el sensor del punto de referencia para la referenciación.

Límites (independientes de la unidad de medida seleccionada):

- Velocidad de arranque/parada ≤ velocidad de referenciación ≤ velocidad máxima

Offset del punto de referencia (sólo referenciación activa)

En este campo se puede indicar el offset del punto de referencia si la posición de referencia deseada difiere de la posición del sensor del punto de referencia.

Si el valor es diferente de 0, el eje ejecuta las siguientes acciones después de la referenciación en el sensor del punto de referencia:

1. Movimiento del eje a la velocidad de referenciación por el valor del offset del punto de referencia
2. Después de extraer el offset del punto de referencia, el eje se encuentra en la posición del punto de referencia que se ha indicado en el parámetro de entrada "Position" de la instrucción de Motion Control "MC_Home".

Límites (independientes de la unidad de medida seleccionada):

- -1.0e12 ≤ offset del punto de referencia: ≤ 1.0e12

Posición del punto de referencia

Como posición del punto de referencia se utiliza la posición parametrizada en la instrucción de Motion Control "MC_Home".

Configuración - Referenciar (objeto tecnológico "Eje" V2...3)

Configuración - Referenciar - General (objeto tecnológico Eje V2...3)

En la ventana de configuración "Referenciar - General", configure la entrada del sensor del punto de referencia para la referenciación activa y pasiva.

Entrada del sensor del punto de referencia

Seleccione la entrada digital para el sensor del punto de referencia en la lista desplegable. La entrada debe ser apta para alarmas. Como entradas para el sensor del punto de referencia están disponibles las entradas de CPU On-board y las entradas de una Signal Board insertada.

Nota

Las entradas digitales están ajustadas de forma estándar con un tiempo de filtrado de 6,4 ms.

La utilización como sensor del punto de referencia puede provocar deceleraciones no deseadas y, con ello, imprecisiones. En determinadas circunstancias, y dependiendo de la velocidad de referenciación y la dimensión del sensor del punto de referencia, puede que no se reconozca el punto de referencia. El tiempo de filtración se puede ajustar en la configuración de dispositivos de las entradas digitales, en "Filtros de entrada".

Se debe seleccionar un tiempo de filtración menor que la duración de la señal de entrada en el sensor del punto de referencia.

Selección de nivel

Seleccione en la lista despegable el nivel del sensor del punto de referencia con el que se debe referenciar.

Consulte también

Marcha - referenciación activa (Página 10477)

Configuración - Referenciar - Pasiva (objeto tecnológico Eje V2...3)

En la ventana de configuración "Referenciar - Pasiva", configure los parámetros que se necesitan para la referenciación pasiva.

En la referenciación pasiva, el movimiento debe ser activado por parte del usuario (p. ej., mediante una petición de desplazamiento del eje). La referenciación pasiva se inicia a través de la instrucción de Motion Control "MC_Home" con el parámetro de entrada utilizado "Mode" = 2.

Lado del sensor del punto de referencia

Indique aquí si el eje debe ser referenciado por el lado inferior o superior al sensor del punto de referencia.

Posición del punto de referencia

Como posición del punto de referencia se utiliza la posición parametrizada en la instrucción de Motion Control "MC_Home".

Nota

Si la referenciación pasiva se efectúa sin una petición de desplazamiento del eje (eje en reposo), la referenciación se lleva a cabo en el siguiente flanco ascendente o descendente del sensor del punto de referencia.

Configuración - Referenciar - Activa (objeto tecnológico Eje V2...3)

En la ventana de configuración "Referenciar - Activa", configure los parámetros que se necesitan para la referenciación activa. La referenciación activa se inicia a través de la instrucción de Motion Control "MC_Home" con el parámetro de entrada utilizado "Mode" = 3.

Permitir inversión de sentido en final de carrera por hardware

Active esta casilla de verificación si desea utilizar los finales de carrera por hardware como levas de inversión para la aproximación al punto de referencia. Los finales de carrera por hardware deben estar activados para poder realizar la inversión del sentido (al menos debe estar configurado el final de carrera por hardware en el sentido de aproximación).

Si se llega al final de carrera por hardware durante la referenciación activa, el eje frena con la deceleración configurada (no con la deceleración de parada de emergencia) y realiza una inversión del sentido. Entonces el sensor del punto de referencia se busca en la dirección contraria.

Si esta inversión del sentido no está activada y el eje alcanza el final de carrera por hardware durante la referenciación activa, la aproximación al punto de referencia se cancela con un error y el eje se frena con la deceleración de parada de emergencia.

Nota

Dentro de lo posible, asegúrese de que la máquina no golpee contra un tope mecánico durante una inversión del sentido, adoptando para ello una de las siguientes medidas:

- Mantenga una velocidad de aproximación reducida.
- Aumente la aceleración / deceleración configuradas.
- Aumente la distancia entre el final de carrera por hardware y el tope mecánico.

Sentido de aproximación / referenciación

Con el sentido seleccionado, determine el sentido de aproximación durante la referenciación activa a fin de buscar el sensor del punto de referencia, así como el sentido de referenciación. El sentido de referenciación determina el sentido con el cual el eje se aproxima al lado del sensor del punto de referencia configurado para realizar la operación de referenciación.

Lado del sensor del punto de referencia

Indique aquí si el eje debe ser referenciado por el lado inferior o superior al sensor del punto de referencia.

Velocidad

Especifique en este campo la velocidad con la que se buscará el sensor del punto de referencia durante la marcha correspondiente.

Límites (independientes de la unidad de medida seleccionada):

- Velocidad de arranque/parada ≤ velocidad de aproximación ≤ velocidad máxima

Velocidad de referenciación

Especifique en este campo la velocidad con la que debe realizarse el posicionamiento hacia el sensor del punto de referencia para la referenciación.

Límites (independientes de la unidad de medida seleccionada):

- Velocidad de arranque/parada ≤ velocidad de referenciación ≤ velocidad máxima

Offset del punto de referencia

En este campo se puede indicar el offset del punto de referencia si la posición de referencia deseada difiere de la posición del sensor del punto de referencia.

Si el valor es diferente de 0, el eje ejecuta las siguientes acciones después de la referenciación en el sensor del punto de referencia:

1. Movimiento del eje a la velocidad de referenciación por el valor del offset del punto de referencia
2. Después de extraer el offset del punto de referencia, el eje se encuentra en la posición del punto de referencia que se ha indicado en el parámetro de entrada "Position" de la instrucción de Motion Control "MC_Home".

Límites (independientes de la unidad de medida seleccionada):

- -1.0e12 ≤ offset del punto de referencia: ≤ 1.0e12

Posición del punto de referencia

Como posición del punto de referencia se utiliza la posición parametrizada en la instrucción de Motion Control "MC_Home".

Bits de diagnóstico, estado y error (objeto tecnológico "Eje" V1...3)

Con la función de diagnóstico "Bits de estado y error" se vigilan en el portal TIA los principales avisos de estado y error del eje. Cuando el eje se encuentra activo la indicación de la función de diagnóstico está disponible en el modo online, en los modos de operación "Control manual" y "Modo automático". Los avisos de estado y error mostrados tienen el siguiente significado:

Estado del eje

Estado	Descripción
Habilitado	El eje está habilitado y listo para ser controlado mediante peticiones de Motion Control. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Enable)
Referenciado	El eje está referenciado y puede ejecutar peticiones de posicionamiento absoluto de la instrucción de Motion Control "MC_MoveAbsolute". Para realizar un posicionamiento relativo no es necesario que el eje esté referenciado. Casos especiales: • Durante la referenciación activa el estado es FALSE. • Si se referencia un eje de forma pasiva, el estado durante la referenciación pasiva es TRUE. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.HomingDone)
Error en eje	Se ha producido un error en el objeto tecnológico "Eje". En el modo automático, la información detallada sobre el error se puede consultar en los parámetros ErrorID y ErrorInfo de las instrucciones de Motion Control. En el modo manual, la causa detallada del error se muestra en el campo "Aviso de error" del panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Error)
Panel de mando del eje activo	Se ha activado el modo de operación "Control manual" en el panel de mando del eje. El panel de mando asume la autoridad sobre el control del objeto tecnológico "Eje". El eje no puede ser controlado por el programa de usuario. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.ControlPanelActive)
Reinicio requerido	Se ha cargado en la memoria de carga una configuración modificada del eje en el estado operativo RUN de la CPU. Para cargar la configuración modificada en la memoria de trabajo, es necesario reiniciar el eje. Utilice para ello la instrucción de Motion Control MC_Reset.

Estado del accionamiento

Estado	Descripción
Listo	El accionamiento está listo. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.DriveReady)
Error en accionamiento	El accionamiento ha indicado un error debido a un fallo en su señal de disponibilidad. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.DriveFault)

Estado del movimiento del eje

Estado	Descripción
Parada	El eje está parado. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.StandStill)
Aceleración	El eje está acelerando. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Acceleration)
Velocidad constante	El eje se mueve a una velocidad constante. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.ConstantVelocity)
Deceleración	El eje está decelerando (frenando). (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Deceleration)

Estado del tipo de movimiento

Estado	Descripción
Posicionamiento	El eje ejecuta una petición de posicionamiento de la instrucción de Motion Control "MC_MoveAbsolute", "MC_MoveRelative" o del panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.PositioningCommand)
Desplazar a la velocidad especificada	El eje ejecuta una petición a la velocidad especificada en la instrucción de Motion Control "MC_MoveVelocity", "MC_MoveJog" o en el panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.SpeedCommand)
Referenciación	El eje ejecuta una petición de referenciación de la instrucción de Motion Control "MC_Home" o del panel de mando del eje. (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.Homing)
Tabla de peticiones activa (a partir del TO Eje V2.0)	El eje se controla con la instrucción de Motion Control "MC_CommandTable". (Variable del objeto tecnológico: <Nombre de eje>.StatusBits.CommandTableActive)

Avisos de error

Error	Descripción
Final de carrera por SW inferior alcanzado	El final de carrera por software inferior ha sido alcanzado. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SwLimitMinReached)
Final de carrera por SW inferior rebasado	Se ha rebasado por defecto el final de carrera por software inferior. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SwLimitMinExceeded)
Final de carrera por SW superior alcanzado	Se ha alcanzado el final de carrera por software superior. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SwLimitMaxReached)
Final de carrera por SW superior rebasado	El final de carrera por software superior ha sido rebasado. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SwLimitMaxExceeded)
Final de carrera por HW inferior aproximado	Se ha aproximado el final de carrera por hardware inferior. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.HwLimitMin)
Final de carrera por HW superior aproximado	Se ha aproximado el final de carrera por hardware superior. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.HwLimitMax)
PTO y HSC ya se utilizan	Un segundo eje utiliza el mismo PTO (Pulse Train Output) y HSC (High Speed Counter) y se encuentra habilitado con "MC_Power". (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.HwUsed)
Error de configuración	El objeto tecnológico "eje" ha sido configurado incorrectamente o se han modificado incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa de usuario. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.ConfigFault)
Error interno	Se ha producido un error interno. (Variable del objeto tecnológico: <Nombre de eje>.ErrorBits.SystemFault)

Consulte también

Bits de estado y error (objetos tecnológicos a partir de V4) (Página 10541)

ErrorID y ErrorInfos

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos V2...3)

Las siguientes tablas recogen una relación de todos los ErrorID y ErrorInfos que pueden ser mostrados en las instrucciones de Motion Control. Junto a la causa del error se muestran también ayudas para eliminar los errores:

Error de funcionamiento con parada del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8000		Error de accionamiento, fallo de "Accionamiento listo"	
	16#0001	-	Acusar error con la instrucción "MC_Reset"; proporcionar la señal del accionamiento; eventualmente iniciar de nuevo la petición
16#8001		Final de carrera por SW inferior activado	
	16#000E	Se ha alcanzado la posición del final de carrera por software inferior con la deceleración actual configurada	Acusar el error con la instrucción "MC_Reset"; salir del final de carrera por software con una petición de marcha en sentido positivo
	16#000F	Se ha alcanzado la posición del final de carrera por software inferior con la deceleración de parada de emergencia	
	16#0010	Se ha rebasado la posición del final de carrera por software inferior con la deceleración de parada de emergencia	
16#8002		Final de carrera por SW superior activado	
	16#000E	Se ha alcanzado la posición del final de carrera por software superior con la deceleración actual configurada	Acusar el error con la instrucción "MC_Reset"; salir del final de carrera por software con una petición de marcha en sentido negativo
	16#000F	Se ha alcanzado la posición del final de carrera por software superior con la deceleración de parada de emergencia	
	16#0010	Se ha rebasado la posición del final de carrera por software superior con la deceleración de parada de emergencia	
16#8003		Final de carrera por HW inferior aproximado	
	16#000E	El final de carrera por hardware inferior ha sido aproximado. Se ha frenado el eje con la deceleración de parada de emergencia. (no se ha encontrado el sensor del punto de referencia durante una búsqueda activa del mismo)	Acusar el error en eje habilitado con la instrucción "MC_Reset"; salir del final de carrera por hardware con una petición de marcha en sentido positivo.
16#8004		Final de carrera por HW superior aproximado	
	16#000E	El final de carrera por hardware superior ha sido aproximado. Se ha frenado el eje con la deceleración de parada de emergencia. (no se ha encontrado el sensor del punto de referencia durante una búsqueda activa del mismo)	Acusar el error en eje habilitado con la instrucción "MC_Reset"; salir del final de carrera por hardware con una petición de marcha en sentido negativo.

ErrorID	ErrorInfo	Descripción	Solución
16#8005		**PTO y HSC ya están siendo utilizados por otro eje**	
	16#0001	-	**El eje ha sido configurado de forma incorrecta:** Corregir la configuración del PTO (Pulse Train Output) / HSC (High Speed Counter) y cargarla en el controlador
			Varios ejes deben trabajar con un PTO: Otro eje utiliza el PTO / HSC. Si el eje actual debe asumir el control, se debe bloquear el otro eje con "MC_Power" Enable = FALSE. (Ver también Utilizando varios ejes con el mismo PTO (Página 10545))
16#8006		**Se ha producido un error de comunicación en el panel de mando del eje**	
	16#0012	Se ha producido un error de desbordamiento de tiempo.	Revise la conexión de cables y vuelva a hacer clic en el botón "Control manual".
16#8007		**Imposible habilitar el eje**	
	16#0025	Ejecutando restart.	Espere hasta que haya concluido el restart del eje.
	16#0026	Ejecutando carga en el estado operativo RUN	Espere hasta que haya finalizado la operación de carga.

Error de funcionamiento sin parada del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8200		**El eje no está habilitado**	
	16#0001	-	Habilitar el eje; volver a iniciar la petición
16#8201		**El eje ya ha sido habilitado por otra instancia de "MC_Power"**	
	16#0001	-	Habilitar el eje sólo con una instancia "MC_Power"
16#8202		**Se ha superado el número máximo de peticiones de Motion Control simultáneamente activas (máx. 200 peticiones para todos los objetos tecnológicos de Motion Control)**	
	16#0001	-	Reducir el número de peticiones activas simultáneamente; volver a iniciar la petición En el parámetro "Busy" = TRUE de la instrucción de Motion Control se reconoce una petición activa.
16#8203		**El eje está funcionando en este momento en "Control manual" (panel de mando)**	
	16#0001	-	Finalizar el "Control manual"; volver a iniciar la petición
16#8204		**El eje no está referenciado**	
	16#0001	-	Referenciar el eje con la instrucción "MC_Home"; volver a iniciar la petición
16#8205		**El eje está siendo controlado por el programa de usuario (el error sólo se muestra en el panel de mando del eje)**	
	16#0013	El eje está habilitado en el programa de usuario.	Bloquear eje con la instrucción "MC_Power" y seleccionar nuevamente "Control manual" en el panel de mando del eje
16#8206		**Objeto tecnológico pendiente de activación**	

ErrorID	ErrorInfo	Descripción	Solución
	16#0001	-	Active el eje con la instrucción "MC_Power" Enable = TRUE, o habilite el eje en el panel de mando del eje.
16#8207		Petición rechazada	
	16#0016	Referenciación activa en curso; no es posible iniciar otro tipo de referenciación.	Espere a que la referenciación activa finalice o cancele la referenciación activa con una petición de movimiento, p. ej., "MC_Halt".
	16#0018	Mientras el eje se está referenciando de forma directa o pasiva, no se puede desplazar con una tabla de peticiones.	Espere a que la referenciación directa o pasiva termine.
	16#0019	Durante el procesamiento de una tabla de peticiones el eje no se puede referenciar de forma directa o pasiva.	Espere a que se cierre la tabla de peticiones o interrumpa dicha tabla con una petición de movimiento p. ej. "MC_Halt".
16#8208		La diferencia entre la velocidad máxima y la velocidad de arranque/parada no es válida	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#8209		El valor de aceleración del TO "Eje" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#820A		No es posible el restart del eje	
	16#0013	El eje está habilitado en el programa de usuario.	Bloquear eje con la instrucción "MC_Power"; ejecutar Restart nuevamente
	16#0027	El eje está funcionando en este momento en "Control manual" (panel de mando)	Finalizar el "Control manual"; repetir Restart
16#820B		Imposible ejecutar la tabla de peticiones	
	16#0026	Ejecutando carga en el estado operativo RUN	Espere hasta que haya finalizado la operación de carga.

Error de parámetros de bloque

ErrorID	ErrorInfo	Descripción	Solución
16#8400		El valor del parámetro "Position" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8401		El valor del parámetro "Distance" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8402		El valor del parámetro "Velocity" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición

ErrorID	ErrorInfo	Descripción	Solución
	16#0008	El valor es mayor que la velocidad máxima configurada	
	16#0009	El valor es inferior a la velocidad de arranque/ parada configurada	
	16#0024	El valor es menor que 0	
16#8403		El valor del parámetro "Direction" de la instrucción de Motion Control no es válido	
	16#0011	El valor de selección no es válido	Corregir el valor de selección; volver a iniciar la petición
16#8404		El valor del parámetro "Mode" de la instrucción de Motion Control no es válido	
	16#0011	El valor de selección no es válido	Corregir el valor de selección; volver a iniciar la petición
	16#0015	Referenciación activa / pasiva no configurada	Corregir la configuración y cargar en el controlador; habilitar el eje y reiniciar la petición
	16#0017	La inversión del sentido en el final de carrera por hardware está activada aunque los finales de carrera por hardware están desactivados.	• Activar los finales de carrera por hardware a través de la variable <Eje>.Config.PositionLimits_HW.Active = TRUE, reiniciar petición • Corregir la configuración y cargar en el controlador; habilitar el eje y reiniciar la petición
16#8405		El valor del parámetro "StopMode" de la instrucción de Motion Control no es válido	
	16#0011	El valor de selección no es válido	Corregir el valor de selección; habilitar nuevamente el eje
16#8406		Pulsación simultánea de avance y retroceso no permitida	
	16#0001	-	Evite la simultaneidad del estado de señal TRUE de los parámetros "JogForward" y "JogBackward"; inicie nuevamente la petición.
16#8407		Cambio del eje en la instrucción "MC_Power" permitido únicamente con eje bloqueado.	
	16#0001	-	Bloquear eje activo; a continuación se puede cambiar y habilitar el eje.
16#8408		El valor del parámetro "Axis" de la instrucción de Motion Control no es válido	
	16#001A	El valor indicado no se corresponde con la versión exigida del TO	Corregir el valor; volver a iniciar la petición
	16#001B	El valor indicado no se corresponde con el tipo exigido del TO	
	16#001C	El valor indicado no es un bloque de datos tecnológicos de Motion Control	
16#8409		El valor del parámetro "CommandTable" de la instrucción de Motion Control no es válido	
	16#001A	El valor indicado no se corresponde con la versión exigida del TO	Corregir el valor; volver a iniciar la petición
	16#001B	El valor indicado no se corresponde con el tipo exigido del TO	
	16#001C	El valor indicado no es un bloque de datos tecnológicos de Motion Control	
16#840A		El valor del parámetro "StartStep" de la instrucción de Motion Control no es válido	
	16#000A	El valor es menor o igual que cero.	Corregir el valor; volver a iniciar la petición
	16#001D	El paso de inicio es mayor que el paso final	
	16#001E	El valor es superior a 32	

ErrorID	ErrorInfo	Descripción	Solución
16#840B		El valor del parámetro "EndStep" de la instrucción de Motion Control no es válido	
	16#000A	El valor es menor o igual que cero.	Corregir el valor; volver a iniciar la petición
	16#001E	El valor es superior a 32	
16#840C		El valor del parámetro "RampUpTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#840D		El valor del parámetro "RampDownTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#840E		El valor del parámetro "EmergencyRampTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	
16#840F		El valor del parámetro "JerkTime" de la instrucción de Motion Control no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor; volver a iniciar la petición
	16#000A	El valor es menor o igual que cero.	

Error de configuración del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8600		Parametrización del generador de impulsos (PTO) no válida	
	16#000B	La dirección no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0014	El hardware seleccionado está en uso en otra aplicación	
16#8601		La parametrización del contador rápido (HSC) no es válida	
	16#000B	La dirección no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0014	El hardware seleccionado está en uso en otra aplicación	
16#8602		La parametrización de la "salida de habilitación" no es válida	
	16#000B	La dirección no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
16#8603		La parametrización de la "entrada de disponibilidad" no es válida	
	16#000B	La dirección no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
16#8604		El valor de "Impulsos por vuelta del motor" no es válido	
	16#000A	El valor es inferior o igual a cero	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
16#8605		El valor de "Recorrido por vuelta del motor" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#000A	El valor es inferior o igual a cero	

ErrorID	ErrorInfo	Descripción	Solución
16#8606		El valor de "Velocidad de arranque/parada" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
	16#0007	La velocidad de arranque/parada es mayor que la velocidad máxima	
16#8607		El valor de "Velocidad máxima" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#8608		El valor de "Aceleración" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#8609		El valor de "Deceleración" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#860A		El valor de "Deceleración de parada de emergencia" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite superior de hardware	
	16#0004	El valor es menor que el límite inferior de hardware	
16#860B		El valor de posición del final de carrera por software inferior no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
	16#0007	El valor de posición del final de carrera por software inferior es mayor que el del superior	
16#860C		El valor de posición del final de carrera por software superior no es válido	

ErrorID	ErrorInfo	Descripción	Solución
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#860D		La dirección del final de carrera por hardware inferior no es válida	
	16#000C	La dirección del flanco descendente no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#000D	La dirección del flanco ascendente no es válida	
16#860E		La dirección del final de carrera por hardware superior no es válida	
	16#000C	La dirección del flanco descendente no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#000D	La dirección del flanco ascendente no es válida	
16#860F		El valor de "Offset del punto de referencia" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8610		El valor de "Velocidad de aproximación" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8611		El valor de "Velocidad de referenciación" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8612		La dirección del sensor del punto de referencia no es válida	
	16#000C	La dirección del flanco descendente no es válida	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"
	16#000D	La dirección del flanco ascendente no es válida	
16#8613		En la referenciación activa, la inversión del sentido en el final de carrera por hardware se encuentra activada, a pesar de que los finales de carrera por hardware no han sido configurados	
	16#0001	-	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
16#8614		El valor de "Tirón" no es válido	

ErrorID	ErrorInfo	Descripción	Solución
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#001F	El valor es mayor que el tirón máximo admisible	
	16#0020	El valor es menor que el tirón mínimo admisible	
16#8615		El valor de "Unidad de medida" no es válido	
	16#0011	El valor de selección no es válido	Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power"

Error de configuración de la tabla de peticiones

ErrorID	ErrorInfo	Descripción	Solución
16#8700		El valor de "Tipo de petición" en la tabla de peticiones no es válido	
	16#0001	-	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
16#8701		El valor de "Posición / recorrido " en la tabla de peticiones no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango numérico (menor que $-1e^{12}$)	
16#8702		El valor de "Velocidad" en la tabla de peticiones no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0008	El valor es mayor que la velocidad máxima configurada	
	16#0009	El valor es inferior a la velocidad de arranque/ parada configurada	
16#8703		El valor de "Duración" en la tabla de peticiones no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0021	El valor es superior a 64800 s	
	16#0022	El valor es inferior a 0.001 s	
16#8704		El valor de "Siguiente paso" en la tabla de peticiones no es válido	
	16#0011	El valor de selección no es válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir online el valor erróneo y, en caso necesario, iniciar de nuevo la petición
	16#0023	La transición de peticiones no está permitida para esta petición	

Errores internos

ErrorID	ErrorInfo	Descripción	Solución
16#8FFF		Error interno	
	16#F0**	-	POWER OFF y POWER ON de la CPU Si esto no produce el resultado deseado, póngase en contacto con el Servicio de Atención al Cliente. Tenga al alcance la siguiente información: • ErrorID • ErrorInfo • Registros del búfer de diagnóstico

Consulte también

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10560)

Utilizando varios ejes con el mismo PTO (Página 10545)

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos V1)

Las siguientes tablas recogen una relación de todos los ErrorID y ErrorInfos que pueden ser mostrados en las instrucciones de Motion Control. Junto a la causa del error se muestran también ayudas para eliminar los errores:

Error de funcionamiento con parada del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8000		Error de accionamiento, fallo de "Accionamiento listo"	
	16#0001	-	Acusar error con la instrucción "MC_Reset"; proporcionar la señal del accionamiento; eventualmente iniciar de nuevo la petición
16#8001		Final de carrera por SW inferior activado	
	16#000E	Se ha alcanzado la posición del final de carrera por software inferior con la deceleración actual configurada	Acusar el error con la instrucción "MC_Reset"; salir del final de carrera por software con una petición de marcha en sentido positivo
	16#000F	Se ha alcanzado la posición del final de carrera por software inferior con la deceleración de parada de emergencia	
	16#0010	Se ha rebasado la posición del final de carrera por software inferior con la deceleración de parada de emergencia	
16#8002		Final de carrera por SW superior activado	
	16#000E	Se ha alcanzado la posición del final de carrera por software superior con la deceleración actual configurada	Acusar el error con la instrucción "MC_Reset"; salir del final de carrera por software con una petición de marcha en sentido negativo
	16#000F	Se ha alcanzado la posición del final de carrera por software superior con la deceleración de parada de emergencia	

ErrorID	ErrorInfo	Descripción	Solución
	16#0010	Se ha rebasado la posición del final de carrera por software superior con la deceleración de parada de emergencia	
16#8003		Final de carrera por HW inferior aproximado	
	16#000E	El final de carrera por hardware inferior ha sido aproximado. Se ha frenado el eje con la deceleración de parada de emergencia. (no se ha encontrado el sensor del punto de referencia durante una búsqueda activa del mismo)	Acusar el error en eje habilitado con la instrucción "MC_Reset"; salir del final de carrera por hardware con una petición de marcha en sentido positivo.
16#8004		Final de carrera por HW superior aproximado	
	16#000E	El final de carrera por hardware superior ha sido aproximado. Se ha frenado el eje con la deceleración de parada de emergencia. (no se ha encontrado el sensor del punto de referencia durante una búsqueda activa del mismo)	Acusar el error en eje habilitado con la instrucción "MC_Reset"; salir del final de carrera por hardware con una petición de marcha en sentido negativo.
16#8005		PTO y HSC ya están siendo utilizados por otro eje	
	16#0001	-	El eje ha sido configurado de forma incorrecta: Corregir la configuración del PTO (Pulse Train Output) / HSC (High Speed Counter) y cargarla en el controlador
			Varios ejes deben trabajar con un PTO: Otro eje utiliza el PTO / HSC. Si el eje actual debe asumir el control, se debe bloquear el otro eje con "MC_Power" Enable = FALSE. (Ver también Utilizando varios ejes con el mismo PTO (Página 10545))

Error de funcionamiento sin parada del eje

ErrorID	ErrorInfo	Descripción	Solución
16#8200		El eje no está habilitado	
	16#0001	-	Habilitar el eje; volver a iniciar la petición
16#8201		El eje ya ha sido habilitado por otra instancia de "MC_Power"	
	16#0001	-	Habilitar el eje sólo con una instrucción "MC_Power"
16#8202		Se ha superado el número máximo de peticiones de Motion Control activas (máx. 200 peticiones para todos los objetos tecnológicos de Motion Control)	
	16#0001	-	Reducir el número de peticiones activas simultáneamente; volver a iniciar la petición En el parámetro "Busy" = TRUE de la instrucción de Motion Control se reconoce una petición activa.
16#8203		El eje está funcionando en este momento en "Control manual" (panel de mando)	
	16#0001	-	Finalizar el "Control manual"; volver a iniciar la petición
16#8204		El eje no está referenciado	

ErrorID	ErrorInfo	Descripción	Solución
	16#0001	-	Referenciar el eje con la instrucción "MC_Home"; volver a iniciar la petición
16#8205		**El eje está siendo controlado por el programa de usuario (el error sólo se muestra en el panel de mando del eje)**	
	16#0001	-	Bloquear eje con la instrucción "MC_Power" y seleccionar nuevamente "Control manual" en el panel de mando del eje
16#8206		**El objeto tecnológico Eje aún no está activado**	
	16#0001	-	Active el eje con la instrucción "MC_Power" Enable = TRUE, o habilite el eje en el panel de mando del eje.
16#8207		**Petición rechazada**	
	16#0016	Referenciación activa en curso; no es posible iniciar otro tipo de referenciación.	Espere a que la referenciación activa finalice o cancele la referenciación activa con una petición de movimiento, p. ej., "MC_Halt". A continuación se puede iniciar el otro tipo de referenciación.

Error de parámetros de bloque

ErrorID	ErrorInfo	Descripción	Solución
16#8400		**El valor del parámetro "Position" de la instrucción de Motion Control no es válido**	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor "Position"; volver a iniciar la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8401		**El valor del parámetro "Distance" de la instrucción de Motion Control no es válido**	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor "Distance"; volver a iniciar la petición
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8402		**El valor del parámetro "Velocity" de la instrucción de Motion Control no es válido**	
	16#0002	El valor no tiene un formato numérico válido	Corregir el valor "Velocity"; volver a iniciar la petición
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8403		**El valor del parámetro "Direction" de la instrucción de Motion Control no es válido**	
	16#0011	Valor de selección no válido	Corregir el valor de selección; volver a iniciar la petición
16#8404		**El valor del parámetro "Mode" de la instrucción de Motion Control no es válido**	
	16#0011	Valor de selección no válido	Corregir el valor de selección; volver a iniciar la petición
	16#0015	Referenciación activa / pasiva no configurada	Corregir la configuración y cargar en el controlador; habilitar el eje y reiniciar la petición

ErrorID	ErrorInfo	Descripción	Solución
	16#0017	La inversión del sentido en el final de carrera por hardware está activada aunque los finales de carrera por hardware están desactivados	• Activar los finales de carrera por hardware a través de la variable <Eje>.Config.PositionLimits_HW.Active = TRUE, reiniciar petición • Corregir la configuración y cargar en el controlador; habilitar el eje y reiniciar la petición
16#8405		El valor del parámetro "StopMode" de la instrucción de Motion Control no es válido	
	16#0011	Valor de selección no válido	Corregir el valor de selección; habilitar nuevamente el eje
16#8406		Pulsación simultánea de avance y retroceso no permitida	
	16#0001	-	Evite la simultaneidad del estado de señal TRUE de los parámetros "JogForward" y "JogBackward"; inicie nuevamente la petición.
16#8407		Cambio del eje en la instrucción de Motion Control "MC_Power" permitido únicamente con eje bloqueado	
	16#0001	-	Bloquear eje activo; a continuación se puede cambiar y habilitar el eje.

Error de configuración

ErrorID	ErrorInfo	Descripción	Solución
16#8600		Parametrización del generador de impulsos (PTO) no válida	
	16#000B	La dirección no es válida	Corregir la configuración de PTO / Pulse Train Output y cargarla en el controlador
16#8601		La parametrización del contador rápido (HSC) no es válida	
	16#000B	La dirección no es válida	Corregir la configuración de HSC / High Speed Counter y cargarla en el controlador
16#8602		La parametrización de la "salida de habilitación" no es válida	
	16#000D	La dirección no es válida	Corregir la configuración y cargarla en el controlador
16#8603		La parametrización de la "entrada de disponibilidad" no es válida	
	16#000D	La dirección no es válida	Corregir la configuración y cargarla en el controlador
16#8604		El valor de "Impulsos por vuelta del motor" no es válido	
	16#000A	El valor es inferior o igual a cero	Corregir la configuración y cargarla en el controlador
16#8605		El valor de "Recorrido por vuelta del motor" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir la configuración y cargarla en el controlador
	16#000A	El valor es inferior o igual a cero	
16#8606		El valor de "Velocidad de arranque/parada" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir la configuración y cargarla en el controlador
	16#0003	El valor es mayor que el límite de hardware	
	16#0004	El valor es menor que el límite de hardware	
	16#0007	La velocidad de arranque/parada es mayor que la velocidad máxima	

ErrorID	ErrorInfo	Descripción	Solución
16#8607		El valor de "Velocidad máxima" no es válido	
	16#0002	El valor no tiene un formato numérico válido	Corregir la configuración y cargarla en el controlador
	16#0003	El valor es mayor que el límite de hardware	
	16#0004	El valor es menor que el límite de hardware	
16#8608		El valor de "Aceleración" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite de hardware	
	16#0004	El valor es menor que el límite de hardware	
16#8609		El valor de "Deceleración" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite de hardware	
	16#0004	El valor es menor que el límite de hardware	
16#860A		El valor de "Deceleración de parada de emergencia" no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0003	El valor es mayor que el límite de hardware	
	16#0004	El valor es menor que el límite de hardware	
16#860B		El valor de posición del final de carrera por software inferior no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
	16#0007	El valor de posición del final de carrera por software inferior es mayor que el del superior	
16#860C		El valor de posición del final de carrera por software superior no es válido	
	16#0002	El valor no tiene un formato numérico válido	• Cargar la configuración sin errores en el controlador; volver a habilitar el eje con la instrucción "MC_Power" • Corregir el valor erróneo online, acusar el error con la instrucción "MC_Reset" y volver a iniciar la petición si fuera necesario
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#860D		La dirección del final de carrera por hardware inferior no es válida	
	16#000C	La dirección del flanco descendente no es válida	Corregir la configuración y cargarla en el controlador
	16#000D	La dirección del flanco ascendente no es válida	
16#860E		La dirección del final de carrera por hardware superior no es válida	
	16#000C	La dirección del flanco descendente no es válida	Corregir la configuración y cargarla en el controlador
	16#000D	La dirección del flanco ascendente no es válida	

ErrorID	ErrorInfo	Descripción	Solución
16#860F		**El valor de "Offset del punto de referencia" no es válido**	
	16#0002	El valor no tiene un formato numérico válido	Corregir la configuración y cargarla en el controlador
	16#0005	El valor se encuentra fuera del rango de números (mayor que $1e^{12}$)	
	16#0006	El valor se encuentra fuera del rango de números (menor que $-1e^{12}$)	
16#8610		**El valor de "Velocidad de aproximación" no es válido**	
	16#0002	El valor no tiene un formato numérico válido	Corregir la configuración y cargarla en el controlador
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8611		**El valor de "Velocidad de referenciación" no es válido**	
	16#0002	El valor no tiene un formato numérico válido	Corregir la configuración y cargarla en el controlador
	16#0008	La velocidad es mayor que la velocidad máxima	
	16#0009	La velocidad es inferior a la velocidad de arranque/parada	
16#8612		**La dirección del sensor del punto de referencia no es válida**	
	16#000C	La dirección del flanco descendente no es válida	Corregir la configuración y cargarla en el controlador
	16#000D	La dirección del flanco ascendente no es válida	
16#8613		**En la referenciación activa, la inversión del sentido en el final de carrera por hardware se encuentra activada, a pesar de que los finales de carrera por hardware no han sido configurados**	
	16#0001	-	Corregir la configuración y cargarla en el controlador

Errores internos

ErrorID	ErrorInfo	Descripción	Solución
16#8FFF		**Error interno**	
	16#F0**	-	POWER OFF y POWER ON de la CPU Si esto no produce el resultado deseado, póngase en contacto con el Servicio de Atención al Cliente. Tenga al alcance la siguiente información: • ErrorID • ErrorInfo • Registros del búfer de diagnóstico

Consulte también

Lista de los ErrorID y las ErrorInfos (objetos tecnológicos a partir de V4) (Página 10560)

Utilizando varios ejes con el mismo PTO (Página 10545)

Variables del objeto tecnológico Eje V1...3

Variables Config V1...3

Variables Config.General V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.General.PTO				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWORD	DW#16#00000000	-	-	-

<Nombre de eje>.Config.General.HSC				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWORD	DW#16#00000000	-	-	-

<Nombre del eje>.Config.General.LengthUnit (objeto tecnológico "Eje" a partir de V2.0)				
La unidad de medida seleccionada en la configuración de los parámetros: • 1013 = "mm" • 1010 =: "m" • 1019 = "in" • 1018 = "ft" • 1005 = "°" (grados) • -1 = "Impulsos"				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	1013	R	-	X

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.DriveInterface V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.DriveInterface.EnableOutput				
Las variables no pueden ser evaluadas en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
-	-	-	-	-

<Nombre de eje>.Config.DriveInterface.ReadyInput				
Las variables no pueden ser evaluadas en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
-	-	-	-	-

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.Mechanics V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.Mechanics.PulsesPerDriveRevolution				
Impulsos por vuelta del motor				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DInt	L#1000	R	-	X

<Nombre de eje>.Config.Mechanics.LeadScrew				
Recorrido por vuelta del motor (indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	1.0E+001	R	-	X

<Nombre de eje>.Config.Mechanics.InverseDirection				
Invertir sentido de dirección				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.DynamicLimits V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.DynamicLimits.MinVelocity				
Velocidad de arranque/parada del eje (indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	1.0E+001	R	-	X

<Nombre de eje>.Config.DynamicLimits.MaxVelocity				
Velocidad máxima del eje (indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	2.5E+002	R	-	X

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.DynamicDefaults V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	2	Con la habilitación del eje
	5	Con el próximo inicio de una petición activa MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt-, MC_CommandTable o MC_Home (Mode = 3).
	6	Con la parada de una petición MC_MoveJog
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.DynamicDefaults.Acceleration					
Aceleración del eje (indicación en la unidad de medida configurada)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	4.8E+001	RW	5	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

<Nombre de eje>.Config.DynamicDefaults.Deceleration					
Deceleración del eje (indicación en la unidad de medida configurada)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	4.8E+001	RW	5, 6	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

<Nombre de eje>.Config.DynamicDefaults.EmergencyDeceleration				
Deceleración de parada de emergencia del eje (indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI

<Nombre de eje>.Config.DynamicDefaults.EmergencyDeceleration					
Real	1.2E+002	RW	2, 5, 6	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

<Nombre del eje>.Config.DynamicDefaults.JerkActive (objeto tecnológico "Eje" a partir de V2.0)				
TRUE = La limitación de tirones está activada				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	RW	1, 5	X

<Nombre del eje>.Config.DynamicDefaults.Jerk (objeto tecnológico "Eje" a partir de V2.0)				
Tirón durante la rampa de aceleración y deceleración del eje (indicación en la unidad de medida configurada)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	1.92E+002	RW	1, 5	X

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.PositionLimits_SW V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	4	Tras una parada del eje, con el siguiente inicio de una petición de Motion Control. La parada del eje se puede comprobar con la variable <Nombre de eje>. StatusBits.Standstill.
	5	Con el próximo inicio de una petición activa MC_MoveAbsolute,MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt-, MC_CommandTable o MC_Home (Mode = 3).
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.PositionLimits_SW.Active					
TRUE = Los finales de carrera por software están activados					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Bool	FALSE	RW	4	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

<Nombre de eje>.Config.PositionLimits_SW.MinPosition					
Posición del final de carrera por software inferior (indicación en la unidad de medida configurada)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	-1.0E+004	RW	4	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

<Nombre de eje>.Config.PositionLimits_SW.MaxPosition					
Posición del final de carrera por software superior (indicación en la unidad de medida configurada)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	1.0E+004	RW	4	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.PositionLimits_HW V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje

	3	Después de habilitar el eje (el eje debe encontrarse previamente bloqueado y parado). La parada del eje se puede comprobar con la variable <Nombre de eje>. StatusBits.Standstill.
	4	Tras una parada del eje, con el siguiente inicio de una petición de Motion Control. La parada del eje se puede comprobar con la variable <Nombre de eje>. StatusBits.Standstill.
	5	Con el próximo inicio de una petición activa MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_MoveJog, MC_Halt, MC_CommandTable o MC_Home (Mode = 3).
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.PositionLimits_HW.Active					
TRUE = Los finales de carrera por hardware están activos.					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Bool	FALSE	RW	3, 4	Firmware V1.0 de la CPU	X
			1, 5, 6	Firmware de la CPU a partir de la versión V2.0	

<Nombre de eje>.Config.PositionLimits_HW.MinSwitchedLevel				
TRUE = 24 V en la entrada de CPU representa que se ha alcanzado el final de carrera por hardware inferior FALSE = 0 V en la entrada de la CPU representa que se ha alcanzado el final de carrera por hardware inferior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.Config.PositionLimits_HW.MinFallingEvent				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	DW#16#00000000	-	-	-

<Nombre de eje>.Config.PositionLimits_HW.MinRisingEvent				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	DW#16#00000000	-	-	-

<Nombre de eje>.Config.PositionLimits_HW.MaxSwitchedLevel				
TRUE = 24 V en la entrada de CPU representa que se ha alcanzado el final de carrera por hardware superior FALSE = 0 V en la entrada de CPU representa que se ha alcanzado el final de carrera por hardware superior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.Config.PositionLimits_HW.MaxFallingEvent				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	DW#16#00000000	-	-	-

<Nombre de eje>.Config.PositionLimits_HW.MaxRisingEvent				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	DW#16#00000000	-	-	-

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Config.Homing V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración del eje.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
	1	Con la activación (la variable <Nombre de eje>.StatusBits.Activated cambia de FALSE a TRUE), el bloqueo o la habilitación del eje
	7	Al iniciar una petición de referenciación pasiva
	8	Al iniciar una petición de referenciación activa
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.Config.Homing.AutoReversal					
TRUE = Se ha activado la inversión del sentido en el final de carrera por hardware (referenciación activa) FALSE = Se ha desactivado la inversión del sentido en el final de carrera por hardware (referenciación activa)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Bool	TRUE	R	-	Objeto tecnológico "Eje" V1.0	X
		RW	1, 8	Objeto tecnológico "Eje" V2.0	

<Nombre de eje>.Config.Homing.Direction				
TRUE = sentido positivo de aproximación para la búsqueda del sensor del punto de referencia y sentido positivo de referenciación (referenciación activa) FALSE = sentido negativo de aproximación para la búsqueda del sensor del punto de referencia y sentido positivo de referenciación (referenciación activa)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI

<Nombre de eje>.Config.Homing.Direction					
Bool	TRUE	R	-	Objeto tecnológico "Eje" V1.0	X
		RW	1, 8	Objeto tecnológico "Eje" V2.0	

<Nombre del eje>.Config.Homing.SideActiveHoming (objeto tecnológico "Eje" a partir de V2.0)				
TRUE = referenciación en el lado superior al sensor del punto de referencia (referenciación activa) FALSE = referenciación en el lado inferior del sensor del punto de referencia (referenciación activa)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	TRUE	RW	1, 8	X

<Nombre de eje>.Config.Homing.SidePassiveHoming (objeto tecnológico "Eje" a partir de V2.0)				
TRUE = referenciación en el lado superior al sensor del punto de referencia (referenciación pasiva) FALSE = referenciación en el lado inferior del sensor del punto de referencia (referenciación pasiva)				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	TRUE	RW	1, 7	X

<Nombre de eje>.Config.Homing.RisingEdge (objeto tecnológico "Eje" V1.0)				
TRUE = referenciación con el flanco de señal descendente del sensor del punto de referencia (referenciación activa) FALSE = referenciación con el flanco de señal ascendente del sensor del punto de referencia (referenciación activa) Para ver cómo afecta la variable a la referenciación pasiva consulte la descripción, en el apartado " Configuración - referenciación".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.Config.Homing.Offset					
Offset del punto de referencia / Indicación en la unidad de medida configurada (referenciación activa)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	0.0	R RW	-	Objeto tecnológico "Eje" V1.0	X
			1, 8	Objeto tecnológico "Eje" V2.0	

<Nombre de eje>.Config.Homing.FastVelocity					
Velocidad de aproximación / Indicación en la unidad de medida configurada (referenciación activa)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	2.0E+002	R	-	Objeto tecnológico "Eje" V1.0	X
		RW	1, 8	Objeto tecnológico "Eje" V2.0	

<Nombre de eje>.Config.Homing.SlowVelocity					
Velocidad de referenciación / Indicación en la unidad de medida configurada (referenciación activa)					
Tipo de datos	Valor inicial	Acceso	efectivo		HMI
Real	4.0E+001	R	-	Objeto tecnológico "Eje" V1.0	X
		RW	1, 8	Objeto tecnológico "Eje" V2.0	

<Nombre de eje>.Config.Homing.FallingEvent				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	DW#16#00000000	-	-	-

<Nombre de eje>.Config.Homing.RisingEvent				
La variable no puede ser evaluada en el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
DWord	DW#16#00000000	-	-	

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables MotionStatus V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.MotionStatus.Position				
Posición actual del eje (indicación en la unidad de medida configurada). Si el eje no está referenciado, la variable muestra el valor de posicionamiento relativo a la posición de habilitación del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	R	-	X

<Nombre de eje>.MotionStatus.Velocity				
Velocidad actual del eje (indicación en la unidad de medida configurada).				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	R	-	X

<Nombre de eje>.MotionStatus.Distance				
Distancia actual con respecto a la posición de destino del eje (indicación en la unidad de medida configurada). El valor de la variable sólo es válido durante la ejecución de una petición de posicionamiento con "MC_MoveAbsolute", "MC_MoveRelative" o del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	R	-	X

<Nombre de eje>.MotionStatus.TargetPosition				
Posición de destino del eje (indicación en la unidad de medida configurada). El valor de la variable sólo es válido durante la ejecución de una petición de posicionamiento con "MC_MoveAbsolute", "MC_MoveRelative" o del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	R	-	X

Consulte también

Estado de movimiento (Página 10544)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables StatusBits V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.StatusBits.Activated				
TRUE = El eje está activado. Se encuentra conectado con el PTO asignado (Pulse Train Output). Los datos del bloque de datos tecnológico se actualizan cíclicamente.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.Enable				
TRUE = el eje está habilitado y listo para aceptar peticiones de Motion Control.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.HomingDone				
TRUE = el eje está referenciado y puede ejecutar peticiones de posicionamiento absolutas. Para realizar un posicionamiento relativo no es necesario que el eje esté referenciado. Durante la referenciación activa el estado es FALSE. Durante la referenciación pasiva, el estado TRUE se mantiene si el eje ya estaba referenciado previamente.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.Done				
TRUE = en el eje no hay ninguna petición de Motion Control activa.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.Error				
TRUE = se ha producido un error en el objeto tecnológico Eje. Si requiere información más detallada sobre el error, consulte los parámetros "ErrorID" y "ErrorInfo" de las instrucciones Motion Control en el modo automático. En el modo manual, la causa detallada del error se muestra en el campo "Aviso de error" del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.StandStill				
TRUE = el eje está parado.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.PositioningCommand				
TRUE = el eje ejecuta una petición de posicionamiento.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.SpeedCommand				
TRUE = el eje ejecuta una petición de posicionamiento con el preajuste de velocidad.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.Homing				
TRUE = el eje ejecuta una petición de referenciación de la instrucción de Motion Control "MC_Home" o del panel de mando del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.CommandTableActive				
TRUE = El eje se controla con la instrucción de Motion Control "MC_CommandTable".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.ConstantVelocity				
TRUE = el eje se mueve a una velocidad constante.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.Acceleration				
TRUE = el eje acelera.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.Deceleration				
TRUE = el eje decelera (frena).				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.ControlPanelActive				
TRUE = se ha activado el modo de operación "Control manual" en el panel de mando del eje. El panel de mando del eje asume la autoridad sobre el control del objeto tecnológico "Eje". El eje no puede ser controlado por el programa de usuario.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.DriveReady				
TRUE = el accionamiento está listo.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.StatusBits.RestartRequired				
TRUE = los valores han sido modificados en la memoria de carga. Para cargar los valores en la memoria de trabajo con la CPU en el estado operativo RUN, es necesario reiniciar el eje. Utilice para ello la instrucción de Motion Control MC_Reset.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

Consulte también

Bits de estado y error (objetos tecnológicos a partir de V4) (Página 10541)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables ErrorBits V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Nombre de eje>.ErrorBits.SystemFault				
TRUE = Error interno de sistema.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.ConfigFault				
TRUE = Configuración incorrecta del eje.				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.DriveFault				
TRUE = El accionamiento ha indicado un error debido a un fallo en su señal de disponibilidad				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.SwLimitMinReached				
TRUE = Se ha alcanzado el final de carrera por software inferior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.SwLimitMinExceeded				
TRUE = Se ha rebasado el final de carrera por software inferior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.SwLimitMaxReached				
TRUE = Se ha alcanzado el final de carrera por software superior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.SwLimitMaxExceeded				
TRUE = Se ha rebasado el final de carrera por software superior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.HwLimitMin				
TRUE = Se ha aproximado el final de carrera por hardware inferior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.HwLimitMax				
TRUE = Se ha aproximado el final de carrera por hardware superior				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

<Nombre de eje>.ErrorBits.HwUsed				
TRUE = Un segundo eje utiliza el mismo PTO (Pulse Train Output) y está habilitado con "MC_Power".				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Bool	FALSE	R	-	X

Consulte también

Bits de estado y error (objetos tecnológicos a partir de V4) (Página 10541)

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables Internal V1...3

Las variables "Internal" no contienen datos relevantes del usuario; no se puede acceder a estas variables desde el programa de usuario.

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Variables ControlPanel V1...3

Las variables "ControlPanel" no contienen datos relevantes del usuario; no se puede acceder a estas variables desde el programa de usuario.

Consulte también

Variables del objeto tecnológico Eje de posicionamiento a partir de V4 (Página 10570)

Actualización de las variables del objeto tecnológico

Las informaciones de estado y error del eje mostradas en las variables del objeto tecnológico se actualizan respectivamente en el punto de control del ciclo.

La modificación de valores realizada en variables de configuración modificables no resulta eficaz de forma inmediata. Consulte en la descripción detallada de la variable correspondiente las condiciones bajo las cuales una modificación resulta efectiva.

Variables del objeto tecnológico Tabla de peticiones V1...3

Variables Config.Command.Command[1 ... 32] V1...3

Leyenda

Tipo de datos	Tipo de datos de la variable	
Valor inicial	Valor inicial de la variable El valor inicial puede ser sobrescrito por la configuración de la tabla de peticiones.	
Acceso	Acceso a la variable en el programa de usuario:	
	RW	Se puede leer y escribir la variable en el programa de usuario.
	R	Se puede leer la variable en el programa de usuario.
	-	No se puede utilizar la variable en el programa de usuario.
efectivo	Indica el momento en que una modificación de la variable resulta efectiva.	
HMI	La variable puede ser utilizada en un sistema HMI.	

<Tabla de peticiones>.Config.Command.Command[x].Type				
Tipo de petición de la petición • 0 = Petición "Vacía" • 2 = Petición "Parada" • 5 = Petición "Positioning Relative" • 6 = Petición "Positioning Absolute" • 7 = Petición "Velocity setpoint" • 151 = Petición "Wait"				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	0	RW	-	X

<Tabla de peticiones>. Config.Command.Command[x].Position				
Posición de destino/recorrido de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real		RW	-	X

<Tabla de peticiones>. Config.Command.Command[x].Velocity				
Velocidad de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RW	-	X

<Tabla de peticiones>. Config.Command.Command[x].Duration				
Duración de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Real	0.0	RW	-	X

<Tabla de peticiones>. Config.Command.Command[x].BufferMode				
Valor de "Siguiente paso" de la petición • 0 = "Finalizar petición" • 1 = "Suavizar transición del movimiento"				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Int	0	RW	-	X

<Tabla de peticiones>. Config.Command.Command[x].StepCode				
Código de paso de la petición				
Tipo de datos	Valor inicial	Acceso	efectivo	HMI
Word	0	RW	-	X

Consulte también

Variables del objeto tecnológico Tabla de peticiones a partir de V4 (Página 10589)

11.4.1.2 Utilizar S7-1500 Motion Control

Introducción

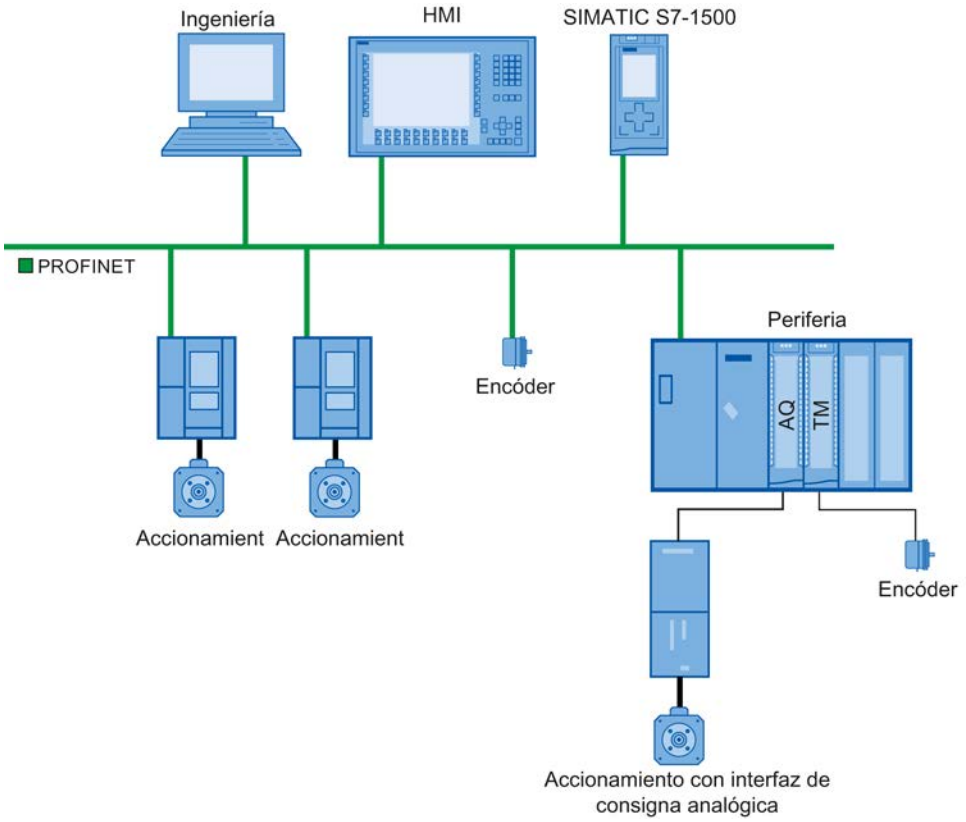
Funcionalidad Motion Control de la CPU S7-1500

La funcionalidad Motion Control soporta el posicionamiento y desplazamiento regulados de ejes y forma parte de todas las CPUs S7-1500.

Posibilidades de configuración

Los accionamientos y encóders se conectan por medio de PROFIBUS DP y PROFINET IO. Los accionamientos con interfaz de consigna analógica se conectan utilizando una salida analógica (AQ). Adicionalmente, los encóders pueden leerse mediante un módulo tecnológico (TM).

La figura siguiente muestra un ejemplo de configuración:

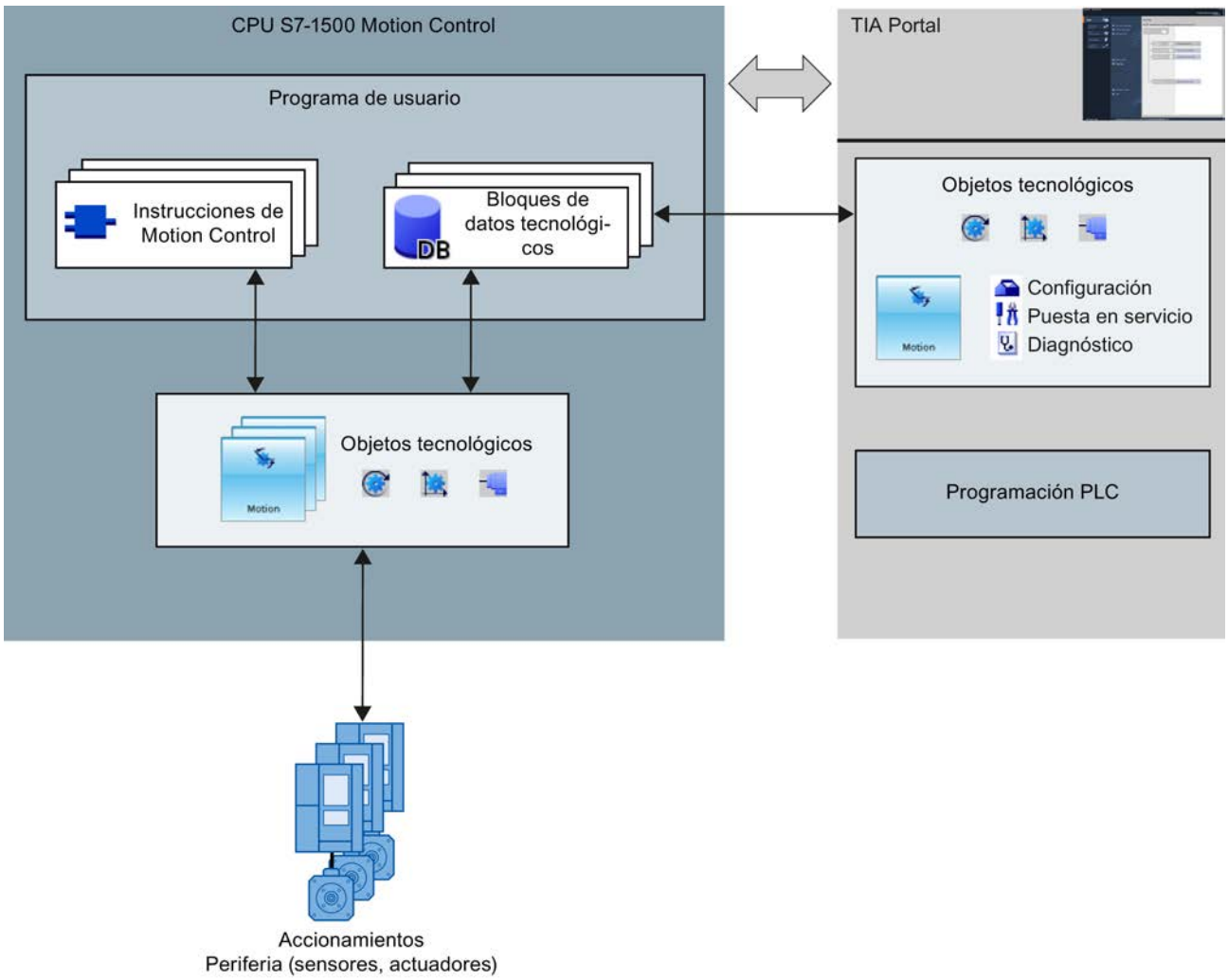


Funcionamiento de Motion Control

Sinopsis

El TIA Portal permite crear un proyecto, configurar objetos tecnológicos y cargar la configuración en la CPU. La funcionalidad Motion Control se procesa en la CPU.
Con las instrucciones de Motion Control se controlan los objetos tecnológicos desde el programa de usuario.
El TIA Portal dispone de funciones adicionales para la puesta en servicio, la optimización (Página 10773) y el diagnóstico (Página 10784).

La figura siguiente muestra en forma de esquema las interfaces de usuario y la integración de Motion Control en la CPU S7-1500. Los conceptos se aclaran brevemente a continuación:



TIA Portal

El TIA Portal le ayuda a configurar y poner en servicio la funcionalidad Motion Control:

- Integración y configuración del hardware
- Creación y configuración de objetos tecnológicos
- Creación del programa de usuario
- Diagnóstico

Instrucciones de Motion Control

Con las instrucciones de Motion Control se ejecutan las funciones deseadas en los objetos tecnológicos. Las instrucciones de Motion Control están disponibles en el TIA Portal en "Instrucciones > Tecnología > Motion Control > S7-1500 Motion Control".

Las instrucciones de Motion Control cumplen la norma PLCopen (versión 2.0).

Objetos tecnológicos

Los objetos tecnológicos representan los objetos reales correspondientes (p. ej. un accionamiento) en el controlador. Las funciones de los objetos tecnológicos se llaman desde las instrucciones de Motion Control en el programa de usuario. Los objetos tecnológicos controlan o regulan el movimiento de objetos reales y notifican información de estado (p. ej. la posición actual).

La configuración de los objetos tecnológicos reproduce las propiedades de los objetos reales. Los datos de configuración se guardan en un bloque de datos tecnológico.

Para Motion Control están disponibles los siguientes objetos tecnológicos:

- **Objeto tecnológico Eje de velocidad de giro**
  El objeto tecnológico Eje de velocidad de giro ("SpeedAxis") sirve para especificar la velocidad de giro de un accionamiento. El movimiento del eje se programa con instrucciones de Motion Control.
- **Objeto tecnológico Eje de posicionamiento**
  El objeto tecnológico Eje de posicionamiento ("PosAxis") sirve para posicionar un accionamiento con regulación de posición. Las instrucciones de Motion Control permiten enviar peticiones de posicionamiento al eje desde el programa de usuario.
- **Objeto tecnológico Encóder externo**
  El objeto tecnológico Encóder externo ("ExternalEncoder") registra una posición y la pone a disposición del controlador. La posición determinada puede evaluarse en el programa de usuario.

Capacidad

El número de objetos tecnológicos utilizables se indica en los datos técnicos de la CPU empleada.

Bloque de datos tecnológico

El bloque de datos tecnológico representa el objeto tecnológico y contiene todos los datos de configuración, consignas y valores reales, además de información de estado del objeto tecnológico. El bloque de datos tecnológico se genera automáticamente al crear el objeto tecnológico. Desde el programa de usuario se accede a los datos del bloque de datos tecnológico.

Programa de usuario

Las instrucciones de Motion Control y el bloque de datos tecnológico representan las interfaces de programación de los objetos tecnológicos. Las instrucciones de Motion Control permiten lanzar y seguir peticiones de Motion Control para objetos tecnológicos desde el programa de usuario. En este caso, el bloque de datos tecnológico representa el objeto tecnológico.

Accionamientos y encóders

Los accionamientos hacen que el eje se mueva. Se integran en la configuración hardware como esclavos.

Al ejecutar una petición de Motion Control en el programa de usuario, el objeto tecnológico se encarga de controlar el accionamiento y leer los valores de los captadores de posición.

Los accionamientos y encóders se conectan mediante telegramas PROFIdrive. Son posibles las siguientes interfaces:

- PROFIBUS DP
- PROFINET IO
- Módulo tecnológico (TM)

Para una conexión analógica del accionamiento existe la posibilidad de especificar la consigna mediante una salida analógica con señal de habilitación. Por medio de módulos E/S se ponen a disposición entradas y salidas analógicas.

Un accionamiento también se denomina actuador y un encóder, sensor.

Principios básicos

Funciones

Las funciones de Motion Control se ejecutan en el programa de usuario o el TIA Portal (en la puesta en servicio) mediante instrucciones de Motion Control.

La tabla siguiente muestra las funciones soportadas por los objetos tecnológicos:

Función	Eje de velocidad de giro	Eje de posicionamiento	Encóder externo
Instrucciones de Motion Control (programa de usuario)			
"MC_Power (Página 5088)" Habilitar y bloquear objetos tecnológicos	X	X	X
"MC_Home (Página 5093)" Referenciar objetos tecnológicos, ajustar el punto de referencia	-	X	X
"MC_MoveJog (Página 5098)" Mover ejes en modo Jog	X	X	-
"MC_MoveVelocity (Página 5102)" Mover ejes con especificación de velocidad	X	X	-
"MC_MoveRelative (Página 5107)" Posicionar ejes de forma relativa	-	X	-
"MC_MoveAbsolute (Página 5111)" Posicionar ejes de forma absoluta	-	X	-
"MC_Halt (Página 5115)" Parar ejes	X	X	-
"MC_Reset (Página 5119)" Acusar alarmas, reinicializar objetos tecnológicos	X	X	X
TIA Portal			

Función	Eje de velocidad de giro	Eje de posicionamiento	Encóder externo
"Panel de mando de eje (Página 10775)" Desplazamiento y referenciado de ejes mediante el TIA Portal	X	X	-
"Optimización (Página 10779)" Optimización de la regulación de posición (factor "Kv")	-	X	-

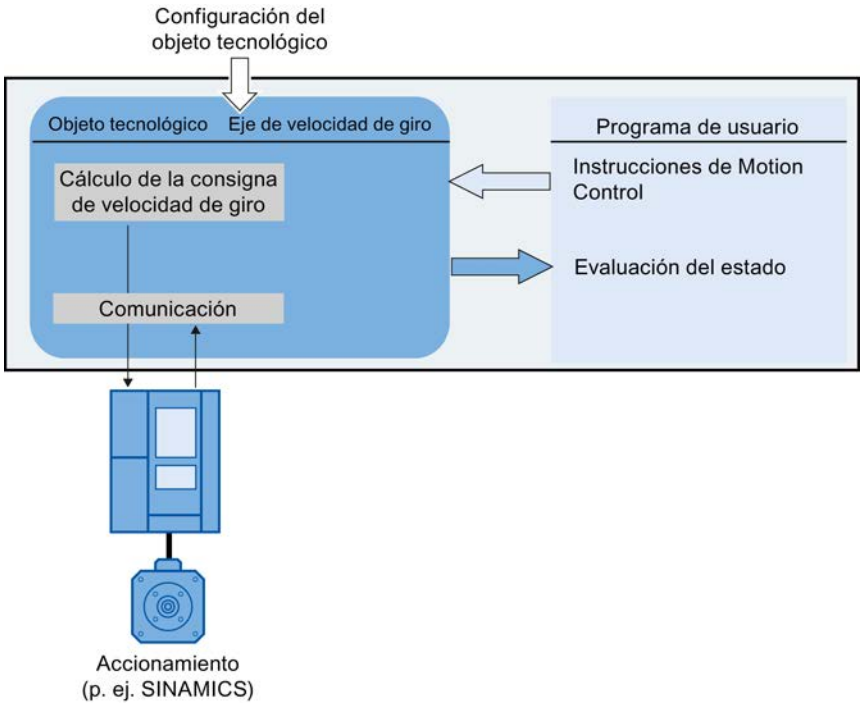
Objeto tecnológico Eje de velocidad de giro

El objeto tecnológico Eje de velocidad de giro calcula consignas de velocidad de giro teniendo en cuenta las especificaciones dinámicas y las transfiere al accionamiento. Todos los movimientos del eje de velocidad de giro están controlados por la velocidad de giro. El sistema tiene en cuenta la existencia de un reductor de carga.

A cada eje de velocidad de giro se le asigna un accionamiento mediante una trama PROFIdrive o mediante una interfaz de consigna analógica.

La velocidad de giro se expresa en vueltas por unidad de tiempo.

La figura siguiente muestra el principio de funcionamiento del objeto tecnológico Eje de velocidad de giro:



Objeto tecnológico Eje de posicionamiento

El objeto tecnológico Eje de posicionamiento calcula consignas de posición teniendo en cuenta las especificaciones dinámicas y transfiere al accionamiento las consignas de

velocidad de giro correspondientes. Todos los movimientos del eje de posicionamiento están regulados por la posición. Para el posicionamiento absoluto, el objeto tecnológico Eje de posicionamiento debe conocer la posición física.

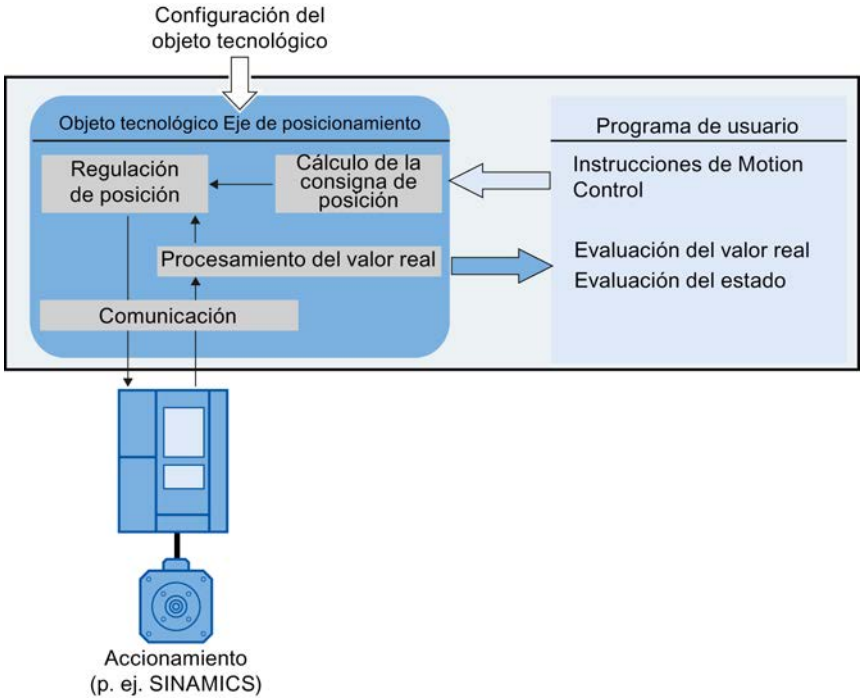
A cada eje de posicionamiento se le asignan un accionamiento mediante una trama PROFIdrive o mediante una interfaz de consigna analógica y un encóder mediante una trama PROFIdrive.

La relación de los valores del encóder con una posición definida se establece por parametrización de las propiedades mecánicas y los ajustes del encóder, así como por un proceso de referenciado. El objeto tecnológico ejecuta también movimientos sin relación de posición y movimientos de posicionamiento relativos en estado no referenciado.

Nota

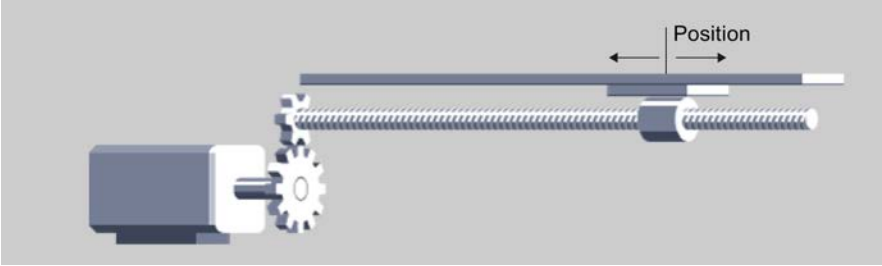
Los objetos tecnológicos Eje de posicionamiento y Encóder externo son independientes y no pueden acoplarse entre sí. La posición real de un encóder externo no puede influir en la regulación de posición de un eje de posicionamiento.

La figura siguiente muestra el principio de funcionamiento del objeto tecnológico Eje de posicionamiento:



Según sea la mecánica, un eje de posicionamiento está diseñado como eje lineal o rotativo:

- Eje lineal



En los ejes lineales, la posición del eje se indica como medida de longitud, p. ej. milímetros (mm).

- Eje rotativo

En los ejes rotativos, la posición del eje se indica como medida angular, p. ej. grados (°).

Consulte también

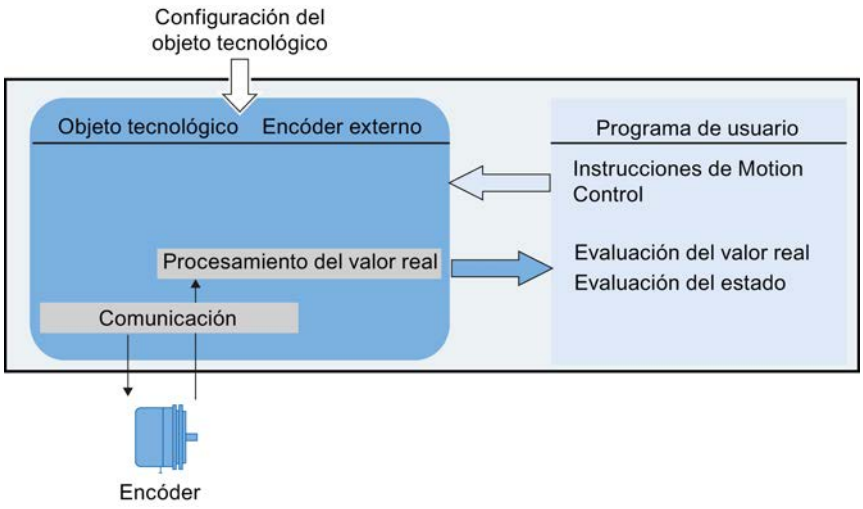
Ajustes de módulo (Página 10648)

Objeto tecnológico Encóder externo

El objeto tecnológico Encóder externo registra una posición y la pone a disposición del controlador.

La relación de los valores del encóder con una posición definida se establece por parametrización de las propiedades mecánicas y los ajustes del encóder, así como por un proceso de referenciado.

La figura siguiente muestra el principio de funcionamiento del objeto tecnológico Encóder externo:



La posición se indica de acuerdo con el sistema de unidades seleccionado:

- Sistema de unidades lineal
  La posición se indica como medida de longitud, p. ej. milímetros (mm).
- Sistema de unidades rotativo
  La posición se indica como medida angular, p. ej. grados (°).

Ajustes de módulo

Los objetos tecnológicos Eje de posicionamiento y Encóder externo pueden parametrizarse con el ajuste "Módulo".

Si "Módulo" está activado, el valor de posición del objeto tecnológico se reproduce en un rango de módulo repetitivo.

El rango de módulo está definido por el valor inicial y la longitud de módulo.

Así, por ejemplo, el valor de posición de un eje rotativo con valor inicial = 0° y longitud = 360° puede reproducirse en el rango de módulo de 0° a 359,9°. Para la resolución del encóder se toma en este ejemplo el valor 0,1°/incremento del encóder.

Unidades de medida

La tabla siguiente muestra las unidades de medida soportadas para recorrido y velocidad:

Recorrido	Velocidad
nm, µm, mm, m, km	mm/s, mm/min, mm/h, m/s, m/min, m/h, km/min, km/h
in, ft, mi	in/s, in/min, ft/s, ft/min, mi/h
°, rad	°/s, °/min, rad/s, rad/min

- La aceleración se ajusta en correspondencia como recorrido/s².
- El tirón se ajusta en correspondencia como recorrido/s³.
- La velocidad de giro se ajusta como vuelta por unidad de tiempo: 1/s, 1/min, 1/h

Nota

Al ajustar o modificar unidades de medida, tenga en cuenta su repercusión en la representación y en el programa de usuario:

- representación en el bloque de datos tecnológico
- asignación de parámetros en el programa de usuario
- entrada e indicación de la posición en el TIA Portal

Todos los datos y todas las indicaciones se muestran conforme a la unidad de medida seleccionada.

Conexión de accionamientos y encóders

Descripción breve

A un eje de velocidad de giro se le asigna un accionamiento.
A un eje de posicionamiento se le asignan un accionamiento y un encóder.
A un encóder externo se le asigna un valor de encóder.

La consigna para el accionamiento se especifica mediante telegramas PROFIdrive o mediante una salida analógica.

Para un encóder existen las posibilidades de conexión siguientes:

- encóder con accionamiento
- encóder con módulo tecnológico
- encóder PROFIdrive directamente con PROFIBUS DP/PROFINET IO

El valor del encóder se transfiere exclusivamente vía telegramas PROFIdrive.

PROFIdrive

PROFIdrive es el perfil estándar normalizado para tecnología de accionamiento a la hora de conectar accionamientos y encóders vía PROFIBUS DP y PROFINET IO.

Los accionamientos que soportan el perfil PROFIdrive se conectan conforme a la norma PROFIdrive.

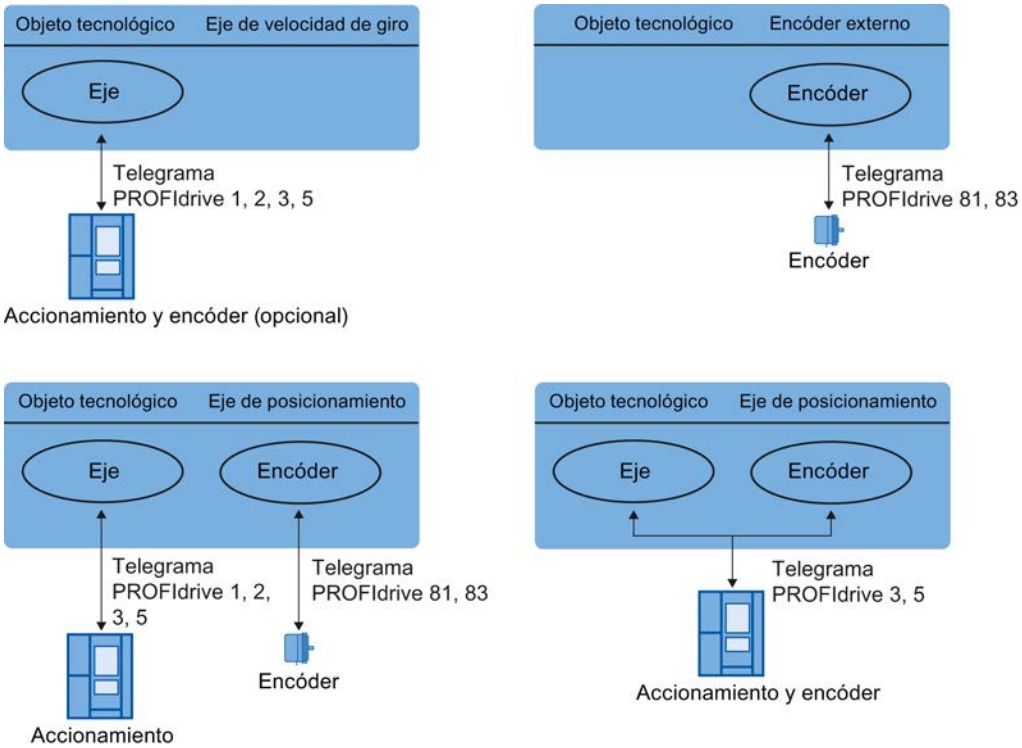
La comunicación entre el controlador y el accionamiento o el encóder se realiza a través de diferentes tramas PROFIdrive. Cada telegrama tiene una estructura normalizada. En cada caso se selecciona el telegrama adecuado a cada aplicación. En los telegramas PROFIdrive se transfieren palabras de control y estado así como consignas y valores reales.

El perfil PROFIdrive soporta también el concepto de regulación "Dynamic Servo Control" (DSC). DSC utiliza la regulación de posición rápida del accionamiento. De este modo es posible solventar tareas de posicionamiento altamente dinámicas.

Telegramas

El valor del encóder se transfiere bien en un telegrama junto con la consigna (telegrama 3 o telegrama 5) bien en un telegrama de encóder separado (telegrama 81 o telegrama 83).

La figura siguiente representa la relación entre los objetos tecnológicos y los accionamientos y encóders:



Explicación de la figura:

- La consigna de un eje de velocidad de giro se transfiere a un accionamiento por medio del telegrama PROFIdrive 1, 2, 3 o 5.
- El valor de un encóder externo se transfiere por medio del telegrama PROFIdrive 81 o 83.
- La consigna de un eje de posicionamiento se transfiere a un accionamiento por medio del telegrama PROFIdrive 1, 2, 3 o 5.
- Valor de encóder del eje de posicionamiento
  El valor del encóder puede transferirse en los siguientes telegramas PROFIdrive:
  - Transferencia en el mismo telegrama PROFIdrive en el que se transfiere la consigna. P. ej. con el telegrama PROFIdrive 3 o 5
  - Transferencia en el telegrama PROFIdrive 81 o 83.

Tipos de telegrama

La tabla siguiente muestra los tipos de telegramas PROFIdrive soportados para la asignación de accionamientos y encóders:

Telegrama	Descripción breve
Telegramas estándar	
1	• Consigna de velocidad de giro 16 bits (NSOLL) • Valor real de velocidad de giro 16 bits (NIST)
2	• Consigna de velocidad de giro 32 bits (NSOLL) • Valor real de velocidad de giro 32 bits (NIST) • Señal de vida
3	• Consigna de velocidad de giro 32 bits (NSOLL) • Valor real de velocidad de giro 32 bits (NIST) • Valor real del encóder • Señal de vida
5	• Consigna de velocidad de giro 32 bits (NSOLL) • Valor real de velocidad de giro 32 bits (NIST) • Dynamic Servo Control (DSC) • Valor real del encóder • Señal de vida
Telegramas estándar del encóder	
81	• Valor real del encóder • Señal de vida
83	• Valor real de velocidad de giro 32 bits (NIST) • Valor real del encóder • Señal de vida

En caso de conexión a través de una trama PROFIdrive, los accionamientos y encóders se manejan y conectan de acuerdo con el perfil PROFIdrive.

Ajustar magnitudes de referencia

Las magnitudes de referencia para la conexión de accionamientos y encóders deben estar ajustadas de forma idéntica tanto en el controlador como en el accionamiento o encóder.

La consigna de velocidad de giro NSOLL y el valor real de velocidad de giro NIST se transfieren en el telegrama PROFIdrive como valor porcentual en relación a la velocidad de giro de referencia. El valor de referencia para la velocidad de giro debe estar ajustada de forma idéntica tanto en el controlador como en el accionamiento.

La resolución del valor real en el telegrama PROFIdrive también debe estar ajustada de forma idéntica tanto en el controlador como en el accionamiento o módulo encóder.

Ajuste de telegramas

Los ajustes del controlador se realizan en el TIA Portal en "Objeto tecnológico > Configuración > Interfaz de hardware > Transferencia de datos".

Los ajustes del accionamiento y el encóder se realizan en la configuración del hardware correspondiente.

La tabla siguiente compara los ajustes del controlador con los correspondientes parámetros PROFIdrive (<TO> significa objeto tecnológico):

Ajuste en el TIA Portal	Controlador Variable del bloque de datos tecnológico	Parámetro PROFIdrive
Telegrama	<TO>.Actor.Interface.Telegram	P922
Velocidad de giro de referencia en [rpm]	<TO>.Actor.DriveParameter.ReferenceSpeed	(Accionamientos SINAMICS: P2000)
Velocidad de giro máxima del motor en [1/min]	<TO>.Actor.DriveParameter.MaxSpeed	(Accionamientos SINAMICS: P1082)
Telegrama del encóder	<TO>.Sensor[n].Interface.Telegram	P922
Sistema del encóder (encóder lineal o rotativo)	<TO>.Sensor[n].System	P979[1] Bit0
Tipo de encóder 0: incremental 1: absoluto 2: cíclico absoluto	<TO>.Sensor[n].Type	P979[5]
Resolución (encóder lineal) La división de la retícula está indicada en la placa de características del encóder como distancia entre rayas en el sistema de medición lineal.	<TO>.Sensor[n].Parameter.Resolution	P979[2]
Incrementos por vuelta (encóder rotativo)	<TO>.Sensor[n].Parameter.StepsPerRevolution	P979[2]
Número de bits para la resolución fina XIST1 (valor real cíclico del encóder, encóder lineal o rotativo)	<TO>.Sensor[n].Parameter.FineResolutionXist1	P979[3]
Número de bits para la resolución fina XIST2 (valor absoluto del encóder, encóder lineal o rotativo)	<TO>.Sensor[n].Parameter.FineResolutionXist2	P979[4]
Vueltas distinguibles del encóder (encóder absoluto rotativo)	<TO>.Sensor[n].Parameter.DeterminableRevolutions	P979[5]

Variables

Las variables siguientes del objeto tecnológico son relevantes para la conexión de accionamientos y encóders:

Telegrama del accionamiento	
<TO>.Actor.Interface.Telegram	Número de telegrama
<TO>.Actor.DriveParameter.ReferenceSpeed	Velocidad de referencia / velocidad de giro de referencia para la velocidad / velocidad de giro NSOLL transferida como valor porcentual

<TO>.Actor.DriveParameter.MaxSpeed	Valor máximo de la consigna de velocidad de giro del accionamiento (N-soll)(PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Consigna analógica: MaxSpeed ≤ 1.17 × ReferenceSpeed)
Telegrama del encóder	
<TO>.Sensor[n].Interface.Telegram	Número de telegrama
<TO>.Sensor[n].System	Sistema de encóder lineal o rotativo
<TO>.Sensor[n].Type	Tipo de encóder incremental, absoluto o cíclico absoluto
<TO>.Sensor[n].Parameter.StepsPerRevolution	Incrementos por vuelta para encóders rotativos
<TO>.Sensor[n].Parameter.DeterminableRevolutions	Número de vueltas distinguibles del encóder en un encóder absoluto multivuelta
<TO>.Sensor[n].Parameter.Resolution	Resolución para encóders lineales La división de la retícula equivale a la distancia entre dos rayas.
Resolución fina	
<TO>.Sensor[n].Parameter.FineResolutionXist1	Número de bits para la resolución fina XIST1 (valor real cíclico del encóder)
<TO>.Sensor[n].Parameter.FineResolutionXist2	Número de bits para la resolución fina XIST2 (valor absoluto del encóder)

Valores reales

Descripción breve

Para el procedimiento y posicionamiento con regulación de posición, es necesario que el controlador conozca el valor real de posición.

El valor real de posición se transfiere en un telegrama PROFIdrive. Tras una única transición del estado operativo de STOP a RUN se actualiza el valor real.

Los valores reales se representan de forma incremental o absoluta en el telegrama PROFIdrive y se normalizan a la unidad tecnológica en el controlador. Mediante referenciado, el valor real se convierte a la posición física del eje o del encóder externo.

El controlador soporta los siguientes tipos de valores reales:

- Valor real incremental
- Valor real absoluto con el ajuste absoluto (rango de medición > rango de desplazamiento del eje)
- Valor real absoluto con el ajuste cíclico absoluto (rango de medición < rango de desplazamiento del eje)

Valor real incremental

El valor real del telegrama PROFIdrive se basa en un valor incremental.

Tras un POWER ON se muestra la posición cero. Tras una única transición del estado operativo de STOP a RUN se actualiza el valor real. La relación entre el objeto tecnológico y la posición mecánica debe volver a establecerse mediante referenciado (Página 10656).

Valor real absoluto

El valor real del telegrama PROFIdrive se basa en un valor absoluto.

Tras un POWER ON se muestra la posición cero. Tras una única transición del estado operativo de STOP a RUN se actualiza el valor real. Mediante el reajuste del encóder absoluto (Página 10671) se asigna el valor absoluto proporcionado a la posición mecánica correspondiente del eje. El reajuste del encóder absoluto debe realizarse una sola vez. El offset del valor absoluto se guarda de forma remanente sin tener en cuenta la conexión o desconexión del controlador.

Distinción de los valores absolutos:

- El rango de medición del encóder es mayor que el rango de desplazamiento del eje: valor absoluto con ajuste absoluto
- El rango de medición del encóder es menor que el rango de desplazamiento del eje: valor absoluto con ajuste cíclico absoluto

Valor real absoluto con el ajuste absoluto (rango de medición > rango de desplazamiento)

La posición del eje resulta directamente del valor actual del encóder. El rango de desplazamiento debe estar dentro de un rango de medición del encóder. Esto significa que el paso por cero del encóder no debe estar dentro del rango de desplazamiento.

Al conectar el controlador se determina la posición del eje a partir del valor absoluto del encóder.

Valor real absoluto con el ajuste cíclico absoluto (rango de medición < rango de desplazamiento)

El encóder proporciona un valor absoluto dentro de su rango de medición. El controlador cuenta los rangos de medición pasados y determina de ese modo la posición correcta del eje sin verse influido por el rango de medición.

Al desconectar el controlador, los rangos de medición pasados se guardan en el área de memoria remanente del controlador.
Al volver a conectarlo, los desbordamientos guardados se tienen en cuenta para calcular el valor real de posición.

ATENCIÓN
Los movimientos del eje con el controlador desconectado pueden alterar el valor real
Si se mueve el eje o el encóder más de la mitad de un rango de medición del encóder cuando el controlador está desconectado, el valor real del controlador ya no coincide con la posición mecánica del eje.

Variables

Para el ajuste de los valores reales son relevantes las variables citadas en el capítulo Referenciado (Página 10672).

Mecánica

Descripción breve

Para la distinción de la posición del objeto tecnológico por parte del usuario es determinante si la posición representa una unidad de longitud (eje lineal) o una magnitud angular (eje rotativo).

Ejemplos de unidades de longitud: mm, m, km

Ejemplos de magnitudes angulares: °, rad

Para determinar la posición física a partir de un valor real del encóder, el sistema debe conocer las diferentes propiedades y disposiciones del sistema mecánico.

Eje de posicionamiento

Se soportan las siguientes posibilidades de ajuste del sistema mecánico:

- Reductor de carga
- Husillo (solo ejes lineales)
- Disposiciones del encóder:
  - En el lado del motor (delante del reductor de carga)
  - En el lado de la carga (después del reductor de carga y, dado el caso, del husillo)
  - Externo (p. ej. rueda de medición)
- Inversión de la consigna
- Inversión del valor real

Encóder externo

Se soportan las siguientes posibilidades de ajuste del sistema mecánico:

- Reductor de medida (con encóders rotativos)
- Husillo (solo con sistema de unidades lineal y encóders rotativos)
- Inversión del valor real

Eje de velocidad de giro

Se soportan las siguientes posibilidades de ajuste del sistema mecánico:

- Reductor de carga
- Inversión de la consigna

Variables

Las variables siguientes del objeto tecnológico son relevantes para el ajuste del sistema mecánico:

Tipo de movimiento	
<TO>.Properties.MotionType	Visualización de movimiento lineal o rotativo 0: movimiento lineal 1: movimiento rotativo
Reductor de carga	
<TO>.LoadGear.Numerator	Reductor de carga del contador
<TO>.LoadGear.Denominator	Reductor de carga del denominador
Paso del husillo	
<TO>.Mechanics.LeadScrew	Paso del husillo
Modo de montaje del encóder	
<TO>.Sensor[n].MountingMode	Modo de montaje del encóder
<TO>.Sensor[n].Parameter.Distance PerRevolution	Recorrido de la carga por vuelta del encóder con encóders montados externamente
Inversión	
<TO>.Actor.InverseDirection	Inversión de la consigna
<TO>.Sensor[n].InverseDirection	Inversión del valor real
Módulo	
<TO>.Modulo.Enable	Activar módulo
<TO>.Modulo.Length	Longitud de módulo
<TO>.Modulo.StartValue	Valor inicial de módulo

Referenciado

Descripción breve

El referenciado establece la referencia entre la posición del objeto tecnológico y la posición mecánica. El valor de posición del objeto tecnológico se asigna para ello a una marca de referencia. Dicha marca representa una posición mecánica conocida.

Con valores reales incrementales, este proceso se denomina referenciado, mientras que con valores reales absolutos se denomina reajuste del encóder absoluto.

El referenciado es un requisito imprescindible para visualizar la posición correcta en el objeto tecnológico y para el posicionamiento absoluto.

El referenciado se activa con la instrucción "MC_Home" de Motion Control y está activo en cada caso para un solo proceso de referenciado.

Estado de referenciado

La variable del objeto tecnológico <TO>.StatusWord.HomingDone indica si el objeto tecnológico Eje o Encóder externo está referenciado.

Tipo de referenciado

El referenciado puede llevarse a cabo mediante un movimiento independiente de referenciado (referenciado activo), mediante el registro de una marca de referencia durante un movimiento iniciado por el usuario (referenciado pasivo) o mediante una asignación de posición directa.

Se distingue entre los siguientes tipos de referenciado:

- Referenciado activo
  El referenciado activo inicia un movimiento de referenciado y ejecuta la marcha necesaria hasta la marca de referencia. Al reconocer la marca de referencia, la posición real se pone al valor configurado. Es posible indicar un offset para el decalaje del punto de referencia. Cuando se inicia el referenciado activo se cancelan los movimientos de desplazamiento en curso. El offset se extrae automáticamente después de la marcha al punto de referencia.
- Referenciado pasivo
  La petición de referenciado no ejecuta ningún movimiento de referenciado propio. Al reconocer la marca de referencia durante un movimiento iniciado por el usuario, la posición real se pone al valor configurado.
  El referenciado pasivo también se denomina referenciado volante.
- Referenciado directo
  Con la petición de referenciado, la posición real se pone directamente al valor configurado o se mueve lo correspondiente a dicho valor.
- Reajuste del encóder absoluto
  El reajuste del encóder absoluto sincroniza la posición del objeto tecnológico con el valor real absoluto existente.

Modo de referenciado

En función del tipo de marca de referencia y de la búsqueda de marca de referencia se distingue entre los modos de referenciado (Página 10658) siguientes:

- Referenciado con marca cero a través de telegrama PROFIdrive y detector de proximidad
- Referenciado con marca cero a través de telegrama PROFIdrive
- Referenciado con entrada digital

Términos

Marca de referencia

Una marca de referencia es una señal de entrada cuya aparición permite asignar una posición mecánica conocida a los valores reales.

Una marca de referencia puede ser:

- Una marca cero
  Se utiliza como marca de referencia la marca cero de un encóder incremental o una marca cero externa.
  La marca cero se registra en el módulo de accionamiento o encóder y se transfiere con el telegrama PROFIdrive. El ajuste y la evaluación de marca cero del encóder o marca cero externa se realiza en el módulo de accionamiento o encóder.
- Un flanco en la entrada digital
  El flanco descendente o ascendente en una entrada digital se utiliza como marca de referencia.

Detector de proximidad

Si en el rango de desplazamiento hay varias marcas cero, el detector de proximidad sirve para seleccionar una marca cero específica delante o detrás del mismo.

Posición de la marca de referencia

Es la posición asignada a la marca de referencia.

La posición de la marca de referencia equivale a la posición del punto de referencia menos el decalaje del punto de referencia.

Punto de referencia

Al final del movimiento de referenciado activo, el eje se desplaza hasta el punto de referencia.

Decalaje del punto de referencia

La diferencia entre la posición de la marca de referencia y la posición del punto de referencia es el decalaje del punto de referencia.

Un decalaje entre la posición de la marca de referencia y el punto de referencia solo es efectivo si el referenciado está activo. El decalaje se extrae tras la sincronización del eje mediante la instrucción "MC_Home" de Motion Control. En ejes con ajuste de módulo, el decalaje del punto de referencia se extrae siempre con el ajuste de sentido para el recorrido más corto.

Inversión de sentido en final de carrera por hardware (leva de inversión)

Los finales de carrera por hardware pueden utilizarse como levas de inversión cuando el referenciado está activo. Si no se detecta la marca de referencia o si se realiza la aproximación desde el lado erróneo, la marcha continúa tras la leva de inversión en sentido contrario.

Modo de referenciado

Para los objetos tecnológicos Eje de posicionamiento y Encóder externo están disponibles varios modos de referenciado con encóders incrementales. El modo de referenciado se ajusta en la configuración.

Referenciado con marca cero a través de telegrama PROFIdrive y detector de proximidad

El sistema comprueba que se llegue al detector de proximidad. Una vez que se ha llegado al detector de proximidad y que este se abandona en el sentido de referenciado parametrizado, se activa el registro de la marca cero mediante el telegrama PROFIdrive.

Una vez que se ha alcanzado la marca cero en el sentido preseleccionado, la posición real del objeto tecnológico se pone a la posición de la marca de referencia.

Referenciado con marca cero a través de telegrama PROFIdrive

El sistema activa el registro de la marca cero en cuanto el valor real del objeto tecnológico se mueve en el sentido de referenciado parametrizado.

Una vez que se ha alcanzado la marca cero en el sentido de referenciado preseleccionado, la posición real del objeto tecnológico se pone a la posición de la marca de referencia.

Referenciado con entrada digital

El sistema comprueba el estado de la entrada digital en cuanto el valor real del eje o encóder se mueve en el sentido de referenciado parametrizado.

Una vez que se ha alcanzado la marca de referencia (activación de la entrada digital) en el sentido de referencia preseleccionado, la posición real del objeto tecnológico se pone a la posición de la marca de referencia.

Nota

Las entradas digitales deben colocarse en la memoria imagen parcial de proceso "IPP OB Servo".

El tiempo de filtrado de las entradas digitales debe ser menor que la duración de la señal de entrada en el interruptor de referenciado.

Consulte también

Referenciado activo en accionamientos SINAMICS con marca cero externa (Página 10854)

Referenciado activo con marca cero y detector de proximidad

Los ejemplos siguientes muestran movimientos de referenciado en sentido positivo y negativo.

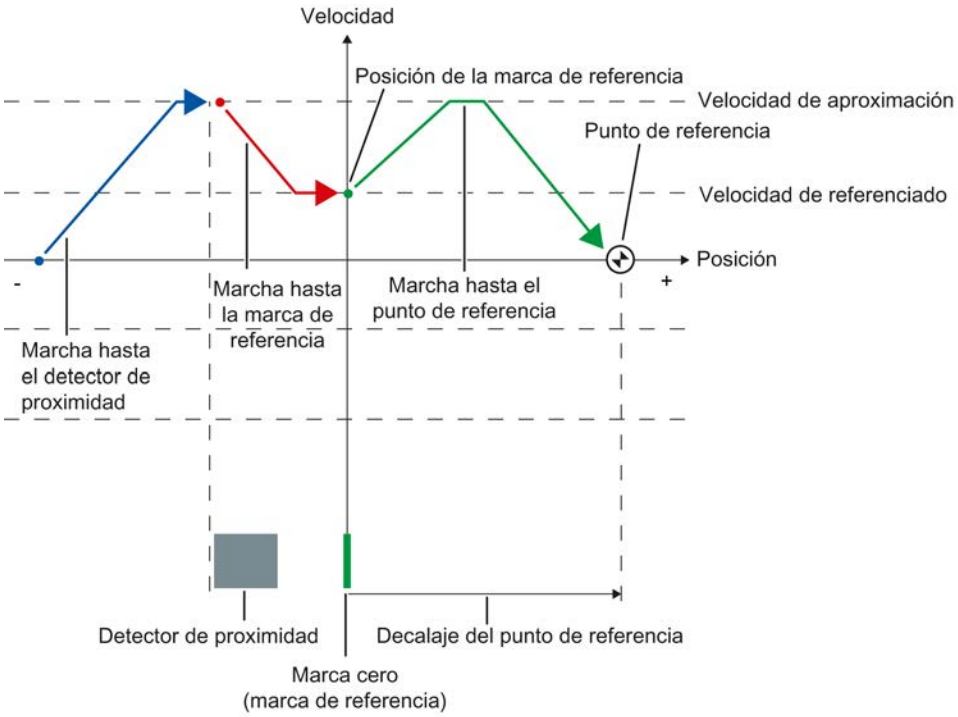
Ejemplo de referenciado en sentido positivo

La marcha hasta la marca de referencia y el punto de referencia se realiza en sentido positivo.

La figura siguiente muestra el movimiento de referenciado con los ajustes siguientes:

- Referenciado activo con marca cero y detector de proximidad
- Aproximación en sentido positivo

- Referenciado en sentido positivo
- Decalaje positivo del punto de referencia

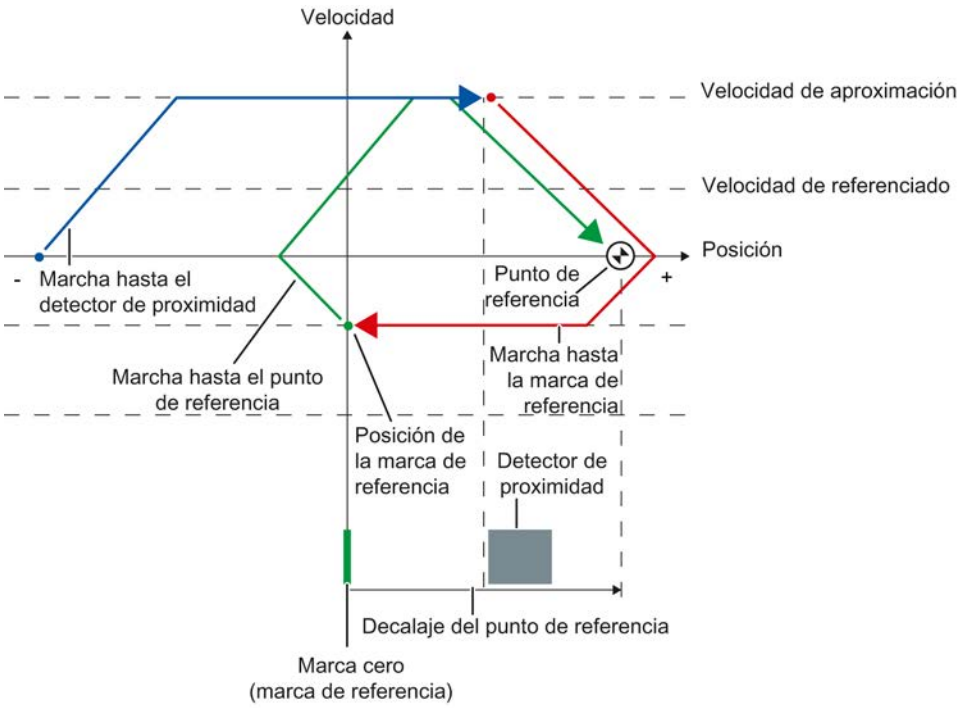


Ejemplo de referenciado en sentido negativo

La marcha hasta la marca de referencia se realiza en sentido negativo debido a una inversión de sentido durante el proceso de referenciado. La marcha hasta el punto de referencia implica otra inversión de sentido y se realiza en sentido positivo.

La figura siguiente muestra el movimiento de referenciado con los ajustes siguientes:

- Referenciado activo con marca cero y detector de proximidad
- Aproximación en sentido positivo
- Referenciado en sentido negativo
- Decalaje positivo del punto de referencia



Desarrollo del movimiento

El movimiento se desarrolla del siguiente modo:

1. **Inicio del referenciado activo mediante la instrucción "MC_Home" de Motion Control**
2. **Marcha hasta el detector de proximidad**
3. **Reconocimiento del detector de proximidad en el sentido de referenciado y marcha hasta la velocidad de referenciado**
4. **Abandono del detector de proximidad y marcha hasta la marca de referencia**
   Al abandonar el detector de proximidad se activa el registro de la marca de referencia.

5. **Reconocimiento de la marca de referencia**
   La posición del objeto tecnológico se fija al reconocer la marca de referencia, en función del Mode ajustado:
   - Parámetro "Mode" en "MC_Home" = 4
     Posición = valor del parámetro "Position" menos
     <TO>.Sensor[n].ActiveHoming.HomePositionOffset
   - Parámetro "Mode" en "MC_Home" = 5
     Posición = valor de la variable <TO>.Homing.HomePosition menos
     <TO>.Sensor[n].ActiveHoming.HomePositionOffset
6. **Marcha hasta el punto de referencia**
   - Parámetro "Mode" en "MC_Home" = 4
     El eje se mueve hasta la posición indicada en el parámetro "Position".
   - Parámetro "Mode" en "MC_Home" = 5
     El eje se mueve hasta la posición indicada en la variable <TO>.Homing.HomePosition .

Nota

Si en el trayecto que va del reconocimiento del detector de proximidad hasta la marca cero no es posible reducir a velocidad de referenciado, se referencia con la velocidad existente al pasar por la marca cero.

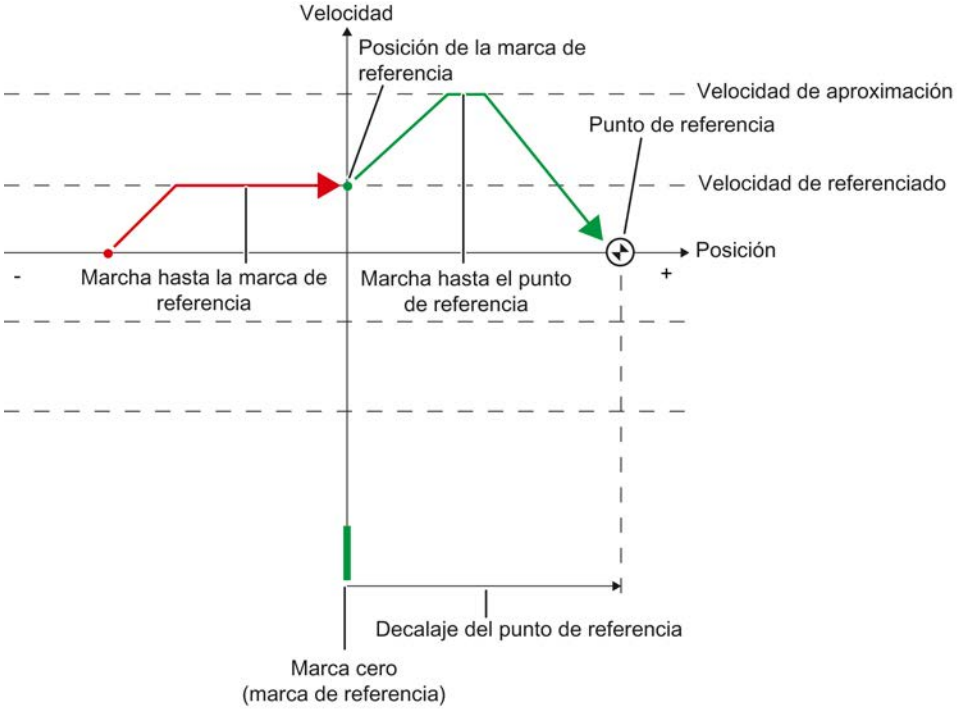
Consulte también

Referenciado activo en accionamientos SINAMICS con marca cero externa (Página 10854)

Referenciado activo con marca cero

La figura siguiente muestra a modo de ejemplo el movimiento de referenciado con los ajustes siguientes:

- Referenciado activo con marca cero
- Referenciado en sentido positivo
- Decalaje positivo del punto de referencia



Desarrollo del movimiento

El movimiento se desarrolla del siguiente modo:

1. **Inicio del referenciado activo mediante la instrucción "MC_Home" de Motion Control**
2. **Marcha hasta la marca de referencia en el sentido de referenciado a velocidad de referenciado**

3. **Reconocimiento de la marca de referencia**
   La posición del eje o encóder se fija al reconocer la marca de referencia, en función del Mode ajustado:
   - Parámetro "Mode" en "MC_Home" = 4
     Posición = valor del parámetro "Position" menos
     <TO>.Sensor[n].ActiveHoming.HomePositionOffset
   - Parámetro "Mode" en "MC_Home" = 5
     Posición = valor de la variable <TO>.Homing.HomePosition menos
     <TO>.Sensor[n].ActiveHoming.HomePositionOffset
4. **Marcha hasta el punto de referencia**
   - Parámetro "Mode" en "MC_Home" = 4
     El eje se mueve hasta la posición indicada en el parámetro "Position".
   - Parámetro "Mode" en "MC_Home" = 5
     El eje se mueve hasta la posición indicada en la variable <TO>.Homing.HomePosition.

Consulte también

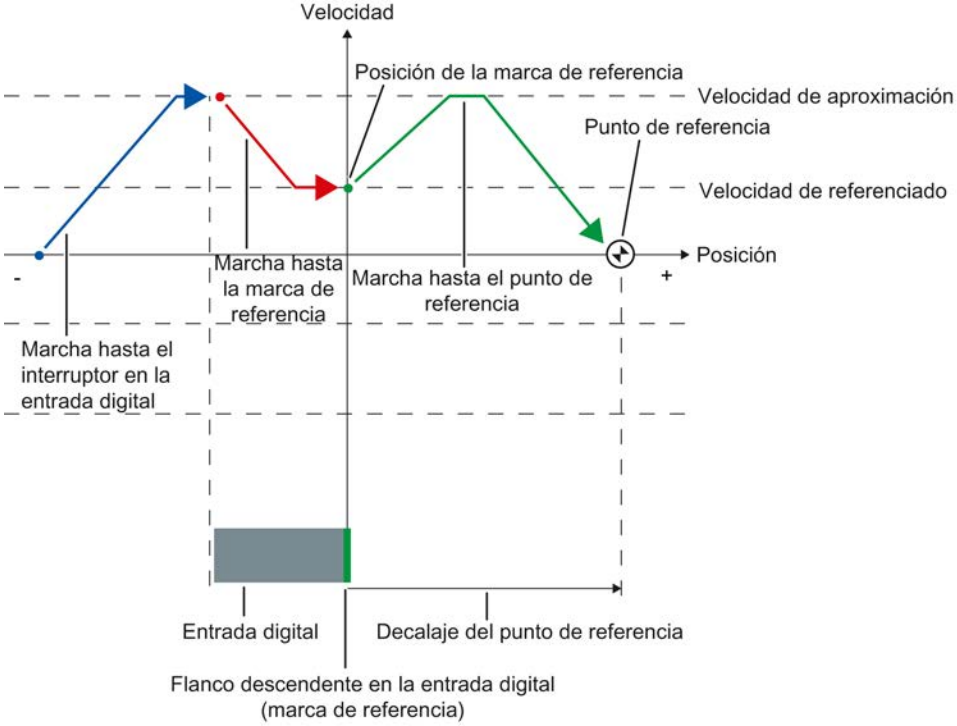
Referenciado activo en accionamientos SINAMICS con marca cero externa (Página 10854)

Referenciado activo con entrada digital

La figura siguiente muestra a modo de ejemplo el movimiento de referenciado con los ajustes siguientes:

- Referenciado activo con entrada digital
- Aproximación en sentido positivo

- Marca de referencia en el lado positivo de la entrada digital
- Decalaje positivo del punto de referencia



Desarrollo del movimiento

El movimiento se desarrolla del siguiente modo:

1. Inicio del referenciado activo mediante la instrucción "MC_Home" de Motion Control
2. Reconocimiento del flanco ascendente en la entrada digital y marcha hasta la velocidad de referenciado
3. Marcha hasta la marca de referencia

4. **Reconocimiento de la marca de referencia**
   En el ejemplo, el flanco descendente del interruptor en la entrada digital representa la marca de referencia.
   La posición del eje o encóder se fija al reconocer la marca de referencia, en función del Mode ajustado.
   - Parámetro "Mode" en "MC_Home" = 4
     Posición = valor del parámetro "Position" menos
     <TO>.Sensor[n].ActiveHoming.HomePositionOffset
   - Parámetro "Mode" en "MC_Home" = 5
     Posición = valor de la variable <TO>.Homing.HomePosition menos
     <TO>.Sensor[n].ActiveHoming.HomePositionOffset
5. **Marcha hasta el punto de referencia**
   - Parámetro "Mode" en "MC_Home" = 4
     El eje se mueve hasta la posición indicada en el parámetro "Position".
   - Parámetro "Mode" en "MC_Home" = 5
     El eje se mueve hasta la posición indicada en la variable <TO>.Homing.HomePosition.

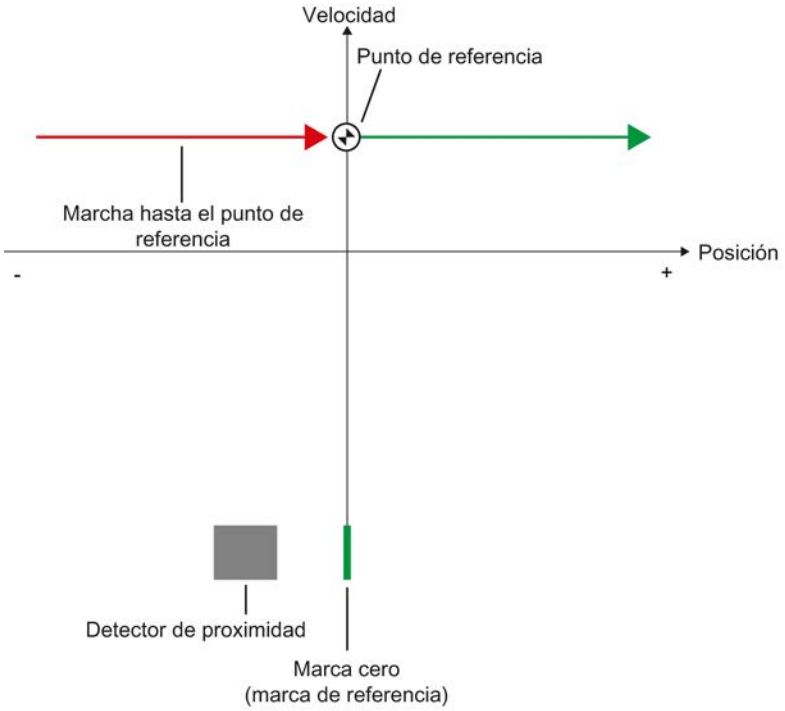
Nota

Si en el trayecto que va del reconocimiento del flanco ascendente hasta el flanco descendente no es posible reducir a velocidad de referenciado, se referencia con la velocidad existente al pasar por la marca de referencia.

Referenciado pasivo con marca cero y detector de proximidad

La figura siguiente muestra a modo de ejemplo el movimiento de referenciado con los ajustes siguientes:

- Referenciado pasivo con marca cero y detector de proximidad
- Referenciado en sentido positivo



Desarrollo del movimiento

El movimiento se desarrolla del siguiente modo:

1. **Activación del referenciado pasivo mediante la instrucción "MC_Home" de Motion Control**
2. **Marcha causada por una petición de movimiento del usuario**
   El registro del detector de proximidad y de la marca de referencia se activa cuando el valor real de posición del eje o encóder se mueve en el sentido de referenciado parametrizado.
3. **Reconocimiento del detector de proximidad**

4. **Abandono del detector de proximidad**
   Al abandonar el detector de proximidad se activa el registro de la marca de referencia.
5. **Reconocimiento de la marca de referencia**
   La posición del eje o encóder se fija al reconocer la marca de referencia, en función del Mode ajustado:
   – Parámetro "Mode" en "MC_Home" = 2
     Posición = valor del parámetro "Position"
   – Parámetro "Mode" en "MC_Home" = 3
     Posición = valor de la variable <TO>.Homing.HomePosition

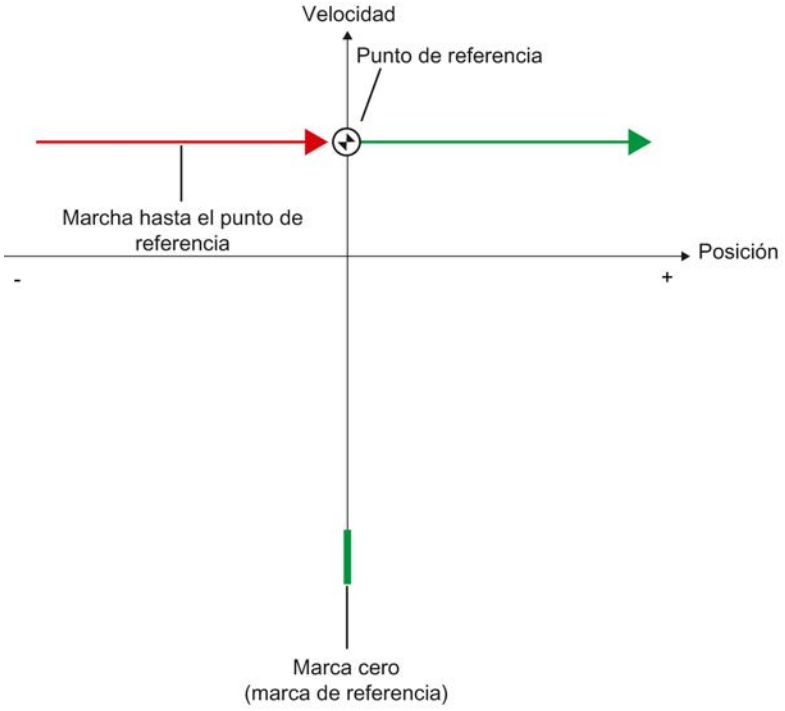
Nota

Si el sentido del movimiento cambia después de abandonar el detector de proximidad y antes de reconocer la marca de referencia, es necesario volver a reconocer el detector de proximidad. La instrucción "MC_Home" de Motion Control permanece activada.

Referenciado pasivo con marca cero

La figura siguiente muestra a modo de ejemplo el movimiento de referenciado con los ajustes siguientes:

- Referenciado pasivo con marca cero
- Referenciado en sentido positivo



Desarrollo del movimiento

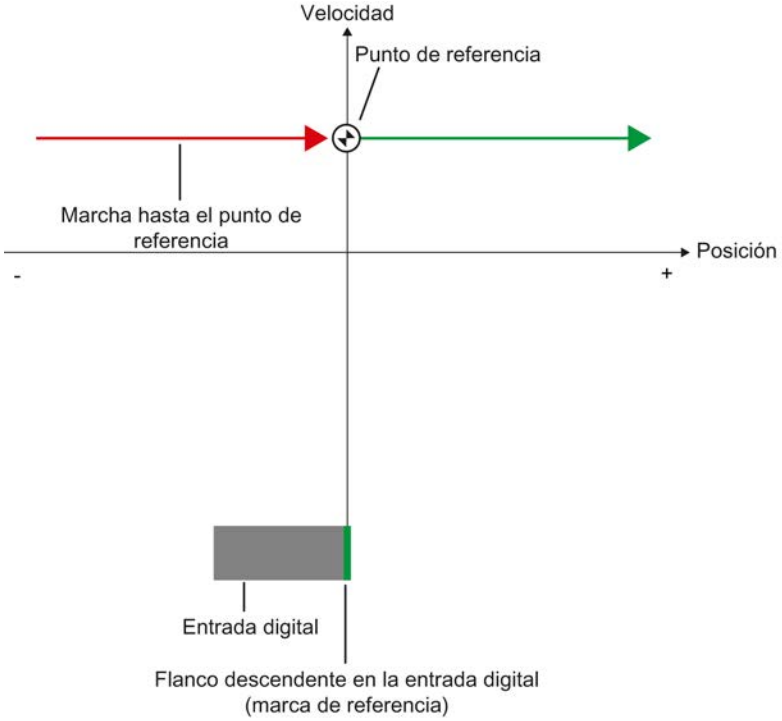
El movimiento se desarrolla del siguiente modo:

1. **Activación del referenciado pasivo mediante la instrucción "MC_Home" de Motion Control**
2. **Marcha causada por una petición de movimiento del usuario**
   El registro de la marca de referencia se activa cuando el valor real de posición del eje o encóder se mueve en el sentido de referenciado parametrizado.
3. **Reconocimiento de la marca de referencia**
   La posición del eje o encóder se fija al reconocer la marca de referencia, en función del Mode ajustado:
   – Parámetro "Mode" en "MC_Home" = 2
     Posición = valor del parámetro "Position"
   – Parámetro "Mode" en "MC_Home" = 3
     Posición = valor de la variable <TO>.Homing.HomePosition

Referenciado pasivo con entrada digital

La figura siguiente muestra a modo de ejemplo el movimiento de referenciado con los ajustes siguientes:

- Referenciado pasivo con entrada digital
- Referenciado en sentido positivo
- Marca de referencia en el lado positivo de la entrada digital



Desarrollo del movimiento

El movimiento se desarrolla del siguiente modo:

1. **Activación del referenciado pasivo mediante la instrucción "MC_Home" de Motion Control**
2. **Marcha causada por una petición de movimiento del usuario**
   El registro de la marca de referencia en la entrada digital se activa cuando el valor real de posición del eje o encóder se mueve en el sentido de referenciado parametrizado.
3. **Reconocimiento de la marca de referencia**
   En el ejemplo, el flanco descendente del interruptor en la entrada digital representa la marca de referencia.
   La posición del eje o encóder se fija al reconocer la marca de referencia, en función del Mode ajustado:
   – Parámetro "Mode" en "MC_Home" = 2
     Posición = valor del parámetro "Position"
   – Parámetro "Mode" en "MC_Home" = 3
     Posición = valor de la variable <TO>.Homing.HomePosition

Inversión de sentido en final de carrera por hardware (leva de inversión)

Con el referenciado activo es posible utilizar opcionalmente los finales de carrera por hardware como levas de inversión. Si no se detecta la marca de referencia o si la aproximación no se realiza en el sentido de referenciado, la marcha continúa tras la leva de inversión en sentido contrario con la velocidad de aproximación.

Cuando se alcanza el final de carrera por hardware, los ajustes predeterminados de dinámica se vuelven efectivos. No se frena con la deceleración de parada de emergencia.

ATENCIÓN
Evitar la marcha hasta un tope mecánico Aplicando una de las medidas siguientes, asegúrese de que la máquina no marcha hasta un tope mecánico en una inversión de sentido: • Mantenga una velocidad de aproximación reducida. • Aumente la aceleración / deceleración configurada. • Aumente la distancia entre el final de carrera por hardware y el tope mecánico.

Referenciado directo

La posición de los objetos tecnológicos Eje de posicionamiento y Encóder externo puede fijarse de forma absoluta o relativa en función del Mode ajustado en "MC_Home".

Fijar una posición absoluta

Para fijar una posición absoluta, proceda del siguiente modo:

1. Introduzca la posición absoluta en el parámetro "Position" de la instrucción "MC_Home" de Motion Control.
2. Llame la instrucción "MC_Home" de Motion Control con el parámetro "Mode" = 0.

La posición se pone al valor especificado en el parámetro "Position".

Fijar una posición relativa

Para fijar una posición relativa, proceda del siguiente modo:

1. Introduzca la posición relativa en el parámetro "Position" de la instrucción "MC_Home" de Motion Control.
2. Llame la instrucción "MC_Home" de Motion Control con el parámetro "Mode" = 1.

La posición se pone a la posición actual más el valor especificado en el parámetro "Position".

Reajuste del encóder absoluto

Con el reajuste del encóder absoluto, Motion Control calcula un offset del valor absoluto que se guarda de forma remanente en la CPU.

La posición del eje o encóder se fija de forma absoluta o relativa en función del Mode ajustado en la instrucción "MC_Home" de Motion Control:

- Parámetro "Mode" = 7 (especificación de posición absoluta)
  Posición = valor del parámetro "Position"
- Parámetro "Mode" = 6 (especificación de posición relativa)
  Posición = posición actual + valor del parámetro "Position"

Desactivación del estado "referenciado"

Encóder incremental

En los casos siguientes, el estado "referenciado" se desactiva y es necesario volver a referenciar el objeto tecnológico:

- Error en el sistema del sensor/fallo del encóder
- Lanzamiento del referenciado activo con la instrucción "MC_Home" de Motion Control con "Mode" = 4, 5 (el estado "referenciado" se activa de nuevo una vez concluido correctamente el proceso de referenciado)
- Lanzamiento del referenciado pasivo con la instrucción "MC_Home" de Motion Control con "Mode" = 2, 3 (el estado "referenciado" se activa de nuevo una vez concluido correctamente el proceso de referenciado)
- Sustitución de la CPU

- Sustitución de la SIMATIC Memory Card
- POWER OFF
- Borrado total
- Cambio de la configuración del encóder
- Reinicialización del objeto tecnológico
- Restablecimiento de la configuración de fábrica de la CPU
- Transferencia de otro proyecto al controlador

Encóder absoluto

En los casos siguientes, el estado "referenciado" se desactiva y es necesario volver a referenciar el objeto tecnológico:

- Error en el sistema del sensor/fallo del encóder
- Sustitución de la CPU
- Cambio de la configuración del encóder
- Restablecimiento de la configuración de fábrica de la CPU
- Transferencia de otro proyecto al controlador

El borrado total de la CPU o la actualización de un proyecto no requieren un reajuste del encóder absoluto.

Variables

Las variables siguientes del objeto tecnológico son relevantes para el referenciado:

Indicadores de estado	
<TO>StatusWord.HomingCommand	Comando de referenciado activo
<TO>StatusWord.HomingDone	Objeto tecnológico referenciado
<TO>ErrorWord.HomingFault	Error al referenciar
Marcha hasta el detector de proximidad	
<TO>.Homing.ApproachDirection	Sentido inicial o de aproximación en la marcha hasta el detector de proximidad
<TO>.Homing.ApproachVelocity	Velocidad para marchar hasta el detector de proximidad
Marcha hasta la marca de referencia	
<TO>.Sensor[n].ActiveHoming.Direction	Sentido de referenciado
<TO>.Homing.ReferencingVelocity	Velocidad para aproximarse a la marca de referencia
Marcha hasta el punto de referencia	
<TO>.Homing.ApproachVelocity	Velocidad para marchar hasta el punto de referencia
Posiciones	
<TO>.Homing.AutoReversal	Inversión en los finales de carrera por hardware

<TO>.Homing.HomePosition	Punto de referencia
<TO>.StatusSensor[n].AbsEncoderOffset	Offset calculado tras el reajuste del encóder absoluto
Parámetros de referenciado activo	
<TO>.Sensor[n].ActiveHoming.Mode	Modo de referenciado
<TO>.Sensor[n].ActiveHoming.SideInput	Lado de la entrada digital
<TO>.Sensor[n].ActiveHoming.Direction	Sentido de referenciado o aproximación
<TO>.Sensor[n].ActiveHoming.DigitalInputAddress	Número de byte de la dirección de periferia de la entrada digital
<TO>.Sensor[n].ActiveHoming.DigitalInputBitNumber	Número de bit de la dirección de periferia de la entrada digital
<TO>.Sensor[n].ActiveHoming.HomePositionOffset	Offset de la marca de referencia hasta el punto de referencia
Parámetros de referenciado pasivo	
<TO>.Sensor[n].PassiveHoming.Mode	Modo de referenciado
<TO>.Sensor[n].PassiveHoming.SideInput	Lado de la entrada digital
<TO>.Sensor[n].PassiveHoming.Direction	Sentido de referenciado o aproximación
<TO>.Sensor[n].PassiveHoming.DigitalInputAddress	Número de byte de la dirección de periferia de la entrada digital
<TO>.Sensor[n].PassiveHoming.DigitalInputBitNumber	Número de bit de la dirección de periferia de la entrada digital

Nota

Evaluación de los bits en StatusWord, ErrorWord y WarningWord

Tenga en cuenta las indicaciones del capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10752).

Regulación

Descripción breve

El regulador de posición del eje de posicionamiento es un regulador P con control anticipativo de velocidad.

Si el accionamiento soporta Dynamic Servo Control (DSC), es posible utilizar opcionalmente el regulador de posición en el accionamiento.

Dynamic Servo Control (DSC)

En accionamientos que soportan Dynamic Servo Control (DSC), es posible utilizar opcionalmente el regulador de posición en el accionamiento. El regulador de posición en el accionamiento suele ejecutarse en el ciclo de regulación de velocidad de giro rápido. De este modo se mejora la calidad de regulación en los accionamientos con acoplamiento digital. Los tiempos de comunicación entre el controlador y el accionamiento se tienen en cuenta automáticamente.

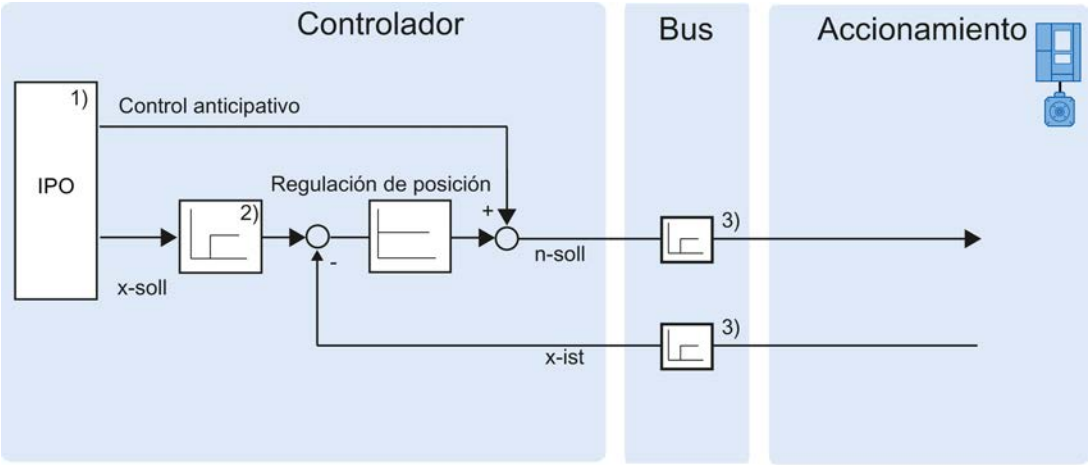
Para DSC debe utilizarse el encóder del accionamiento y este último debe soportar el telegrama estándar 5.

Consulte también

Telegramas (Página 10647)

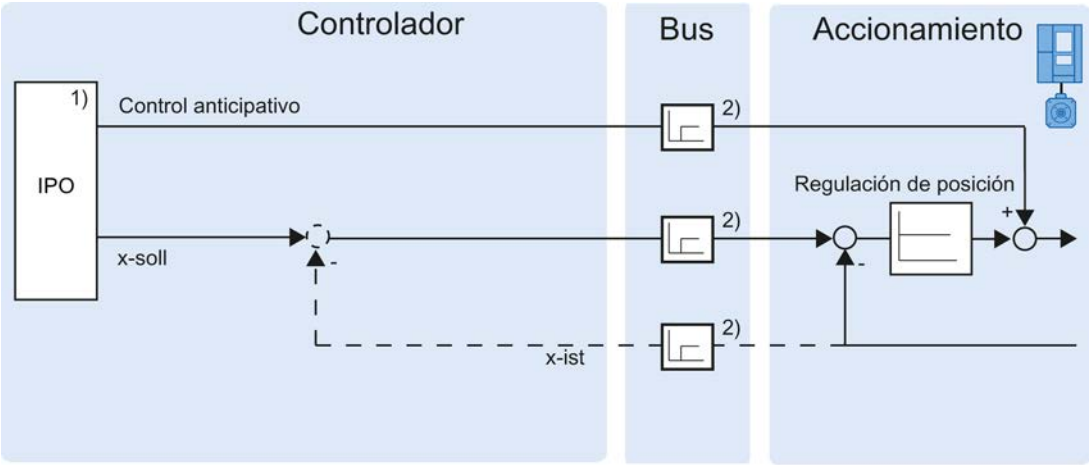
Estructura de regulación

La figura siguiente muestra la estructura de regulación efectiva sin DSC:



1) Interpolador con control de movimiento
2) Consideración interna de los ciclos de señal
3) Comunicación controlador-accionamiento

La figura siguiente muestra la estructura de regulación efectiva con DSC:



1) Interpolador con control de movimiento
2) Comunicación controlador-accionamiento

Variables

Las variables siguientes del objeto tecnológico son relevantes para la regulación:

Parámetro	
<TO>.PositionControl.Kv	Ganancia P de la regulación de posición
<TO>.PositionControl.Kpc	Control anticipativo de velocidad de la regulación de posición
<TO>.PositionControl.EnableDSC	Activar DSC

Vigilancias relacionadas con la posición

Descripción breve

Para vigilar el posicionamiento y movimiento están disponibles las funciones siguientes en el objeto tecnológico Eje de posicionamiento:

- Vigilancia de posicionamiento (Página 10675)
  El valor real de posición debe alcanzar una ventana de posicionamiento en un tiempo determinado y permanecer en dicha ventana durante un tiempo mínimo de permanencia.
- Vigilancia de errores de seguimiento (Página 10676)
  El error de seguimiento se vigila basándose en un límite dependiente de la velocidad. El error de seguimiento máximo permitido depende de la consigna de velocidad.

En caso de violarse una vigilancia se emiten alarmas tecnológicas. El objeto tecnológico reacciona de acuerdo con la reacción a alarma.

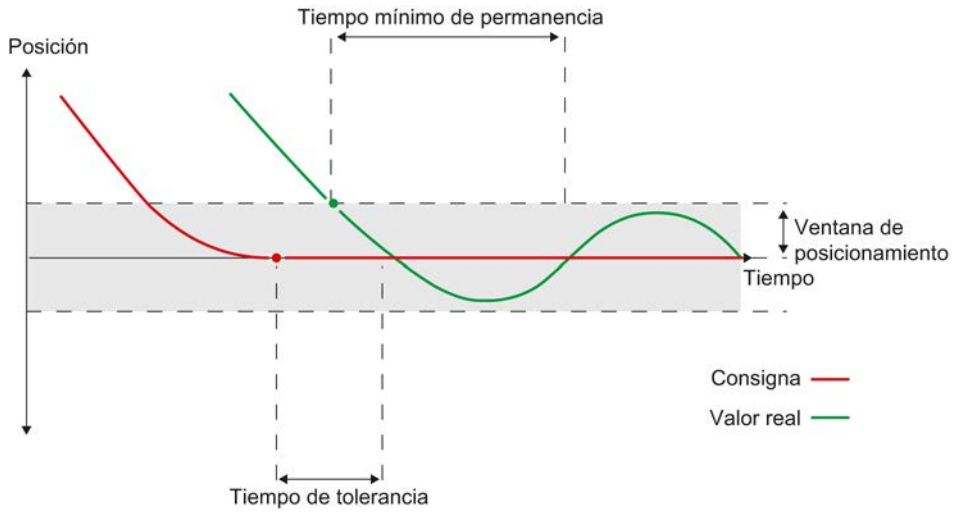
Vigilancia de posicionamiento

La vigilancia de posicionamiento vigila el comportamiento de la posición real al final del cálculo de consigna.

En cuanto la consigna de velocidad alcanza el valor cero, el valor real de posición debe estar en la ventana de posicionamiento dentro de un tiempo de tolerancia. El valor real no debe salir de la ventana de posicionamiento mientras dura el tiempo mínimo de permanencia.

Si, al final de un movimiento de posicionamiento, el valor real de posición alcanza la ventana de posicionamiento dentro del tiempo de tolerancia y permanece en dicha ventana durante el tiempo mínimo de permanencia, se activa <TO>.StatusWord.Done en el bloque de datos tecnológico. Con ello ha concluido una petición de movimiento.

La figura siguiente muestra el desarrollo temporal y la ventana de posicionamiento:



La vigilancia de posicionamiento no distingue cómo finaliza la interpolación de consigna. El final de la interpolación de consigna puede alcanzarse del siguiente modo, por ejemplo:

- cuando la consigna alcanza la posición de destino
- cuando el interpolador se detiene con regulación de posición durante el movimiento a causa de la instrucción "MC_Halt" de Motion Control

Violación de la vigilancia de posicionamiento

En los casos siguientes, la vigilancia de posicionamiento lanza la alarma tecnológica 541 y el objeto tecnológico se bloquea (reacción a alarma: retirar habilitación):

- El valor real no alcanza la ventana de posicionamiento dentro del tiempo de tolerancia.
- El valor real sale de la ventana de posicionamiento mientras dura el tiempo mínimo de permanencia.

Vigilancia de errores de seguimiento

El error de seguimiento en el objeto tecnológico Eje de posicionamiento se vigila teniendo en cuenta un límite dependiente de la velocidad. El error de seguimiento permitido depende de la consigna de velocidad.

Con velocidades inferiores a una velocidad mínima ajustable es posible especificar un error de seguimiento constante admisible.

Por encima de esta velocidad mínima, el error de seguimiento admisible aumenta proporcionalmente a la consigna de velocidad. El error de seguimiento máximo admisible puede alcanzarse con la velocidad máxima.

Cálculo del error de seguimiento

El error de seguimiento es la diferencia entre la consigna de posición y el valor real de posición. Los tiempos de transferencia de la consigna al accionamiento y del valor real de posición al controlador se tienen en cuenta a la hora de calcular el error de seguimiento.

Límite de advertencia

Para el error de seguimiento es posible especificar un límite de advertencia. Dicho límite se ajusta como valor porcentual y es efectivo en relación al error de seguimiento admisible actualmente. Cuando se alcanza el límite de advertencia del error de seguimiento se lanza la alarma tecnológica 522. Es una advertencia y no incluye ninguna reacción a alarma.

Rebase por exceso del error de seguimiento admisible

En caso de rebase por exceso del error de seguimiento admisible se emite la alarma tecnológica 521 y el objeto tecnológico se bloquea (reacción a alarma: retirar habilitación).

Variables

Las variables siguientes del objeto tecnológico son relevantes para la vigilancia de posicionamiento:

Indicadores de estado	
<TO>.StatusWord.Standstill	Se pone al valor TRUE cuando el valor real de posición rebasa por defecto el umbral de velocidad y no lo abandona durante el tiempo mínimo de permanencia. La señal de velocidad cero solo existe en el eje de posicionamiento.
<TO>.StatusWord.Done	**Eje de posicionamiento** Se pone al valor TRUE cuando el valor real de velocidad alcanza la ventana de posicionamiento dentro del tiempo de tolerancia y permanece allí durante el tiempo mínimo de permanencia. **Eje de velocidad de giro** Se pone a TRUE cuando el movimiento ha concluido y, por consiguiente, la consigna de velocidad de giro es cero.
<TO>.ErrorWord.PositioningFault	Ha ocurrido un error de posicionamiento.
Posiciones y tiempos	
<TO>.PositioningMonitoring.ToleranceTime	Tiempo máximo admisible hasta alcanzar la ventana de posicionamiento El tiempo se inicia con el final de la interpolación de consigna.
<TO>.PositioningMonitoring.MinDwellTime	Tiempo mínimo de permanencia en la ventana de posición
<TO>.PositioningMonitoring.Window	Ventana de posicionamiento
Señal de velocidad cero	
<TO>.StandstillSignal.VelocityThreshold	Umbral de velocidad para la señal de velocidad cero
<TO>.StandstillSignal.MinDwellTime	Tiempo mínimo de permanencia por debajo del umbral de velocidad

Las variables siguientes del objeto tecnológico son relevantes para la vigilancia de errores de seguimiento:

Indicadores de estado	
<TO>.StatusPositioning.FollowingError	Error de seguimiento actual
<TO>.ErrorWord.FollowingErrorFault	Indicador de estado que indica que el error de seguimiento es demasiado grande
<TO>.WarningWord.FollowingErrorWarning	Indicador de estado que indica que se ha alcanzado el límite de advertencia del error de seguimiento
Bits de control	
<TO>.FollowingError.EnableMonitoring	Activar y desactivar vigilancia de errores de seguimiento
Valores límite	
<TO>.FollowingError.MinVelocity	Consigna de velocidad inferior para la curva característica del error de seguimiento máximo
<TO>.FollowingError.MinValue	Error de seguimiento admisible por debajo de <TO>.FollowingError.MinVelocity
<TO>.FollowingError.MaxValue	Error de seguimiento máximo admisible a la velocidad máxima del eje
<TO>.FollowingError.WarningLevel	Límite de advertencia como valor porcentual en relación al error de seguimiento máximo admisible (en función de la velocidad según la curva característica)

Limitación del rango de desplazamiento

Descripción breve

Los finales de carrera por hardware y software limitan el rango de desplazamiento y la zona de trabajo admisibles del eje de posicionamiento. Deben activarse en la configuración o en el programa de usuario antes de utilizarse.

La relación entre zona de trabajo, rango de desplazamiento máximo y finales de carrera se representa en la figura siguiente:

Final de carrera por hardware

Los finales de carrera por hardware son finales de carrera que limitan el rango de desplazamiento máximo admisible del eje.

Las posiciones de los finales de carrera por hardware deben elegirse de modo que, en caso necesario, el eje disponga de suficiente carrera de deceleración. El eje debería pararse antes de llegar a un tope mecánico.

Aproximación a los finales de carrera por hardware

A la hora de vigilar la limitación de rango no se distingue si se alcanzan o sobrepasan los interruptores.

Cuando se realiza la aproximación a un final de carrera por hardware se emite la alarma tecnológica 531 y el objeto tecnológico se bloquea (reacción a alarma: retirar habilitación).

Excepción

Si en el referenciado se utilizan los finales de carrera por hardware como levas de inversión o levas de referencia, la vigilancia de los finales de carrera por hardware es inactiva.

Marcha libre

La posición del eje cuando se detecta el final de carrera por hardware se guarda internamente en la CPU. El estado del final de carrera por hardware aproximado no se restablece hasta que no se abandona el final de carrera por hardware y el eje se encuentra de nuevo en el rango de desplazamiento máximo.

Para que el eje marche libremente después de alcanzar el final de carrera por hardware y restablecer su estado, proceda del siguiente modo:

1. Acuse la alarma tecnológica para permitir movimientos en el sentido de la marcha libre.
2. Desplace el eje en el sentido de la marcha libre hasta abandonar el final de carrera por hardware.
   Posteriormente, el eje debe estar en el rango de desplazamiento máximo.
   Si se produce un desplazamiento en contra del sentido de la marcha libre antes de abandonar el final de carrera por hardware, se dispara de nuevo la vigilancia.

El diagrama siguiente muestra el comportamiento de la palabra de estado durante la aproximación al final de carrera por hardware y la marcha libre del eje:

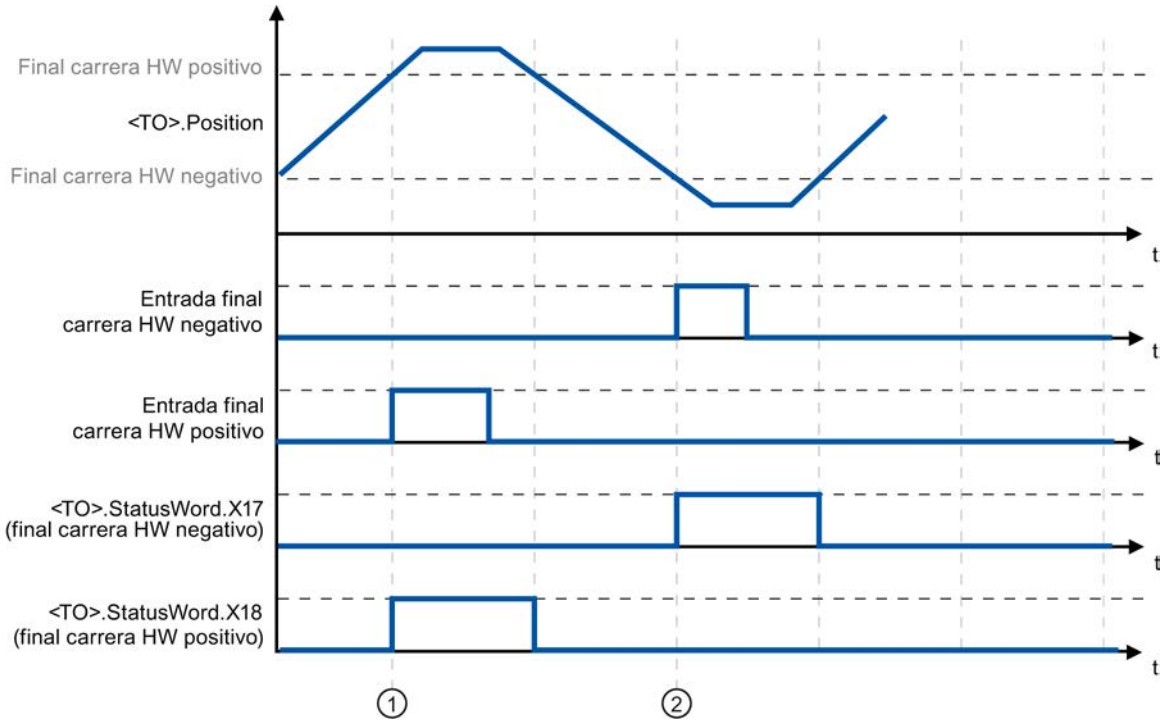


Figura 11-10 Aviso de estado_HW-ES

En el tiempo ① se guarda en la CPU la posición del eje al detectarse el final de carrera por hardware positivo. Para restablecer el estado del final de carrera por hardware es necesario que se rebase por defecto esta posición.

En el tiempo ② se guarda en la CPU la posición del eje al detectarse el final de carrera por hardware negativo. Para restablecer el estado del final de carrera por hardware es necesario que se rebase por exceso esta posición.

Final de carrera por software

Los finales de carrera por software limitan la zona de trabajo del eje. Los finales de carrera por software deben posicionarse siempre dentro de los finales de carrera por hardware, en lo que se refiere al rango de desplazamiento. Puesto que las posiciones de los finales de carrera por software pueden ajustarse de forma flexible, es posible adaptar la zona de trabajo del eje individualmente según el perfil de velocidad actual.

Los finales de carrera por software no son efectivos hasta que hay un valor real válido tras referenciar el objeto tecnológico. La vigilancia de los finales de carrera por software se aplica a la consigna.

Módulo activado

Cuando módulo está activado se vigila su posición.

Los finales de carrera por software se activan o desactivan por medio de las variables del bloque de datos tecnológico. Si las posiciones de ambos finales de carrera por software están fuera del rango de módulo, la vigilancia no es efectiva. No se comprueba si las posiciones de los finales de carrera por software están dentro del rango de módulo.

Aproximación a los finales de carrera por software

Cuando se inicia un movimiento de posicionamiento no se comprueba si se pasa por el final de carrera por software al aproximarse a la posición de destino especificada.

Si se pasa por el final de carrera por software se emite la alarma tecnológica 533 y el eje se para con los valores dinámicos máximos (reacción a alarma: parada con valores dinámicos máximos). El objeto tecnológico permanece habilitado.

Superación de los finales de carrera por software

Cuando se supera el final de carrera por software se emite la alarma tecnológica 534 y el objeto tecnológico se bloquea (reacción a alarma: retirar habilitación).

Marcha libre

Para que el eje pueda marchar libremente después de violar el final de carrera por software, proceda del siguiente modo:

1. Acuse la alarma tecnológica.
2. Mueva el eje en el sentido de la marcha libre hasta abandonar el final de carrera por software.
   Si antes de abandonar el final de carrera por software se produce un movimiento en contra del sentido de la marcha libre, vuelve a dispararse la vigilancia.

Variables

Las variables siguientes del objeto tecnológico son relevantes para los finales de carrera por software:

Indicadores de estado	
<TO>.StatusWord.SWLimitMinActive	El final de carrera por software negativo está activo
<TO>.StatusWord.SWLimitMaxActive	El final de carrera por software positivo está activo
<TO>.ErrorWord.SWLimit	Hay una alarma pendiente que indica que se ha violado un final de carrera por software
Bits de control	
<TO>.PositonLimits_SW.Active	Activa / desactiva la vigilancia de los finales de carrera por software
Valores de posición	
<TO>.PositonLimits_SW.MinPosition	Posición del final de carrera por software negativo
<TO>.PositonLimits_SW.MaxPosition	Posición del final de carrera por software positivo

Las variables siguientes del objeto tecnológico son relevantes para los finales de carrera por hardware:

Indicadores de estado	
<TO>.StatusWord.HWLimitMinActive	El final de carrera por hardware negativo está activo
<TO>.StatusWord.HWLimitMaxActive	El final de carrera por hardware positivo está activo

<TO>.ErrorWord.HWLimit	Hay una alarma pendiente; se ha violado un final de carrera por hardware
Bits de control	
<TO>.PositonLimits_HW.Active	Activa / desactiva la vigilancia de los finales de carrera por hardware
Parámetro	
<TO>.PositonLimits_HW.MinSwitchLevel	Selección de nivel para activar el final de carrera por hardware inferior: FALSE: con el nivel Low la señal está activa TRUE: con el nivel High la señal está activa
<TO>.PositonLimits_HW.MinSwitchAddress	Número de byte de la dirección de periferia del final de carrera por hardware para la posición inferior o mínima
<TO>.PositonLimits_HW.MinSwitchBitNumber	Número de bit de la dirección de periferia del final de carrera por hardware para la posición inferior o mínima
<TO>.PositonLimits_HW.MaxSwitchLevel	Selección de nivel para activar el final de carrera por hardware superior: FALSE: con el nivel Low la señal está activa TRUE: con el nivel High la señal está activa
<TO>.PositonLimits_HW.MaxSwitchAddress	Número de byte de la dirección de periferia del final de carrera por hardware para la posición superior o máxima
<TO>.PositonLimits_HW.MaxSwitchBitNumber	Número de bit de la dirección de periferia del final de carrera por hardware para la posición superior o máxima

Control de movimiento y límites dinámicos

Descripción breve

El movimiento del eje se controla con perfiles de velocidad (Página 10682). Los perfiles de velocidad se calculan de acuerdo con las especificaciones dinámicas. Un perfil de velocidad define el comportamiento del eje al arrancar, frenar y cambiar de velocidad. Durante el posicionamiento se calcula un perfil de velocidad que mueve el eje hasta el punto de destino.

Los valores máximos de velocidad, aceleración y tirón se calculan a partir de las propiedades del accionamiento y del sistema mecánico. Estos valores máximos pueden ajustarse en los límites dinámicos. Los límites dinámicos son efectivos en todos los movimientos generados mediante el objeto tecnológico.

La deceleración de parada de emergencia (Página 10684) ajustable se dispara mediante la instrucción MC_Power de Motion Control o una alarma tecnológica.

La limitación de tirones reduce el esfuerzo del sistema mecánico durante una rampa de aceleración o deceleración. De ahí resulta un perfil de velocidad "redondeado".

Perfil de velocidad

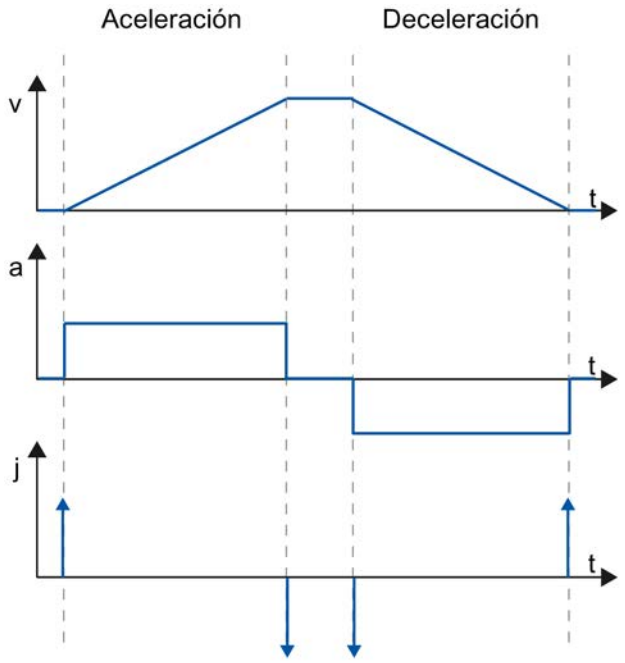
Para controlar el movimiento del eje se soportan un perfil de velocidad sin limitación de tirones y otro con limitación de tirones.

Los valores dinámicos para el movimiento se especifican en la petición de movimiento. También existe la posibilidad de utilizar los valores del preajuste de dinámica. Los preajustes y límites de velocidad, aceleración, deceleración y tirón se ajustan en la configuración.

Para influir en la velocidad es posible sobreponer una corrección de velocidad a la velocidad de desplazamiento actual.

Perfil de velocidad sin limitación de tirones

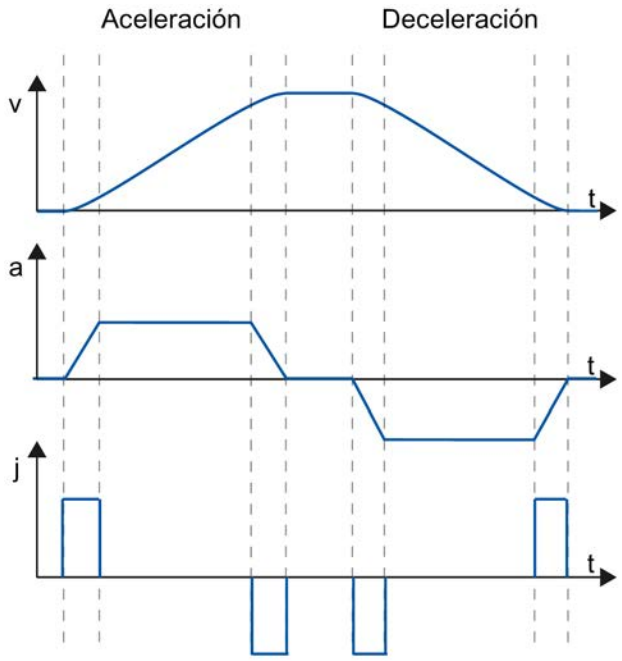
La figura siguiente muestra la velocidad, la aceleración y el tirón:



v - velocity (velocidad)
a - acceleration (aceleración)
j - jerk (tirón)
t - time (tiempo)

Perfil de velocidad con limitación de tirones

La figura siguiente muestra la velocidad, la aceleración y el tirón:



v - velocity (velocidad)
a - acceleration (aceleración)
j - jerk (tirón)
t - time (tiempo)

Un perfil de velocidad con limitación de tirones se utiliza para un recorrido de aceleración y deceleración constantes. El tirón puede especificarse previamente.

Deceleración de parada de emergencia

En caso de parada con la rampa de parada de emergencia, el eje se frena hasta pararse con la deceleración de parada de emergencia ajustada sin limitación de tirones.

En los casos siguientes es efectiva la deceleración de parada de emergencia ajustada:

- En caso de rampa de parada de emergencia activada mediante la instrucción "MC_Power" de Motion Control con el parámetro "StopMode" = 0.
- En caso de una alarma tecnológica con la reacción local a alarma "Parada con rampa de parada de emergencia".

Esta deceleración de parada de emergencia puede ser mayor que la deceleración máxima. Si se ajusta una deceleración de parada de emergencia menor, en caso de una "parada en final de carrera por software" y de aparecer de alarma tecnológica con la reacción local a alarma puede suceder que el eje no se detenga hasta después del final de carrera.

Variables

Las variables siguientes del objeto tecnológico son relevantes para el control de movimiento:

Estado	
<TO>.StatusWord	Indicador de estado para un movimiento activo
<TO>.Position	Consigna de posición
<TO>.Velocity	Consigna de velocidad / de velocidad de giro
<TO>.ActualPosition	Posición real
<TO>.ActualVelocity	Velocidad real
<TO>.ActualSpeed	Velocidad de giro real del motor (si la consigna analógica = 0.0)
<TO>.Acceleration	Aceleración de consigna
<TO>.ActualAcceleration	Aceleración real
Solapamiento	
<TO>.Override.Velocity	Corrección de velocidad
Valores límite de dinámica	
<TO>.DynamicLimits.MaxVelocity	Limitación dinámica para velocidad máxima
<TO>.DynamicLimits.MaxAcceleration	Limitación dinámica para aceleración máxima
<TO>.DynamicLimits.MaxDeceleration	Limitación dinámica para deceleración máxima
<TO>.DynamicLimits.MaxJerk	Limitación dinámica para tirón máximo
Preajustes de dinámica	
<TO>.DynamicDefaults.Velocity	Preajuste de velocidad
<TO>.DynamicDefaults.Acceleration	Preajuste de aceleración
<TO>.DynamicDefaults. Deceleration	Preajuste de deceleración
<TO>.DynamicDefaults.Jerk	Preajuste de tirón
<TO>.DynamicDefaults. EmergencyDeceleration	Deceleración de parada de emergencia

Comportamiento de ejecución

Bloques de organización para Motion Control

Descripción

Cuando se crea un objeto tecnológico se generan también automáticamente bloques de organización para procesar objetos tecnológicos. La funcionalidad Motion Control de los objetos tecnológicos genera un nivel de ejecución propio y se llama de acuerdo con el ciclo de la aplicación.

Se crean los siguientes bloques de organización:

- MC-Servo [OB91]
  Cálculo del regulador de posición
- MC-Interpolator [OB92]
  Evaluación de las instrucciones de Motion Control, la generación de consignas y función de vigilancia

Los bloques de organización están protegidos (protección de know-how). No es posible ver o modificar el código del programa.

La relación de ciclo entre los dos bloques de organización es siempre 1:1. MC-Servo [OB91] se ejecuta siempre antes de MC-Interpolator [OB92].

El ciclo de aplicación y la prioridad de los bloques de organización se pueden ajustar conforme a las exigencias de calidad de regulación y carga del sistema.

Ciclo de aplicación

El ciclo de aplicación en el que se llama al MC-Servo [OB91] puede ajustarse en las propiedades del bloque de organización, en "General > Tiempo de ciclo":

- Sincronía con el bus
  El MC-Servo [OB91] se llama de forma síncrona o reducida con respecto a un sistema en bus. El tiempo de ciclo de emisión se ajusta en las propiedades del sistema en bus seleccionado. En la lista desplegable "Periferia descentralizada" pueden seleccionarse los siguientes sistemas en bus:
  - PROFIBUS DP isócrono
  - PROFINET IO isócrono

  El MC-Servo [OB91] no puede llamarse de forma síncrona con respecto a un sistema en bus conectado a la CPU mediante un procesador de comunicaciones / módulo de comunicaciones (CP/CM).
- Cíclico
  El MC-Servo [OB91] se llama de forma cíclica con el ciclo de aplicación indicado.

El ciclo de aplicación debe elegirse lo suficientemente grande como para poder procesar todos los objetos tecnológicos en un solo ciclo. Si el tiempo de procesamiento de los objetos tecnológicos es mayor que el ciclo de aplicación se producen desbordamientos (Página 10687).

El tiempo de ejecución de MC-Servo [OB91] y MC-Interpolator [OB92] se comprueba con la instrucción avanzada "RT_INFO".

Reducción de ciclo (a partir de CPU V1.5)

El ciclo de aplicación del MC-Servo [OB91] puede reducirse al tiempo de ciclo de emisión de un sistema PROFINET IO isócrono. Como factor puede ajustarse un múltiplo entero del tiempo de ciclo de emisión. Para el ciclo de aplicación son posibles valores de hasta 14 veces el tiempo de ciclo de emisión (como máximo 32 ms).

Si se llaman un OB de alarma de sincronismo y el MC-Servo [OB91] de forma síncrona al mismo sistema PROFINET IO, debe ajustarse el mismo ciclo de aplicación en ambos bloques de organización.

Prioridad

En caso necesario, la prioridad de los bloques de organización se ajusta en sus propiedades, en "General > Atributos > Prioridad".

- **MC-Servo [OB91]**
  Prioridad 17 a 31 (valor predeterminado 25)
- **MC-Interpolator [OB92]**
  Prioridad 16 a 30 (valor predeterminado 24)

La prioridad del MC-Servo [OB91] debe ser como mínimo un punto más alta que la prioridad del MC-Interpolator [OB92].

Memoria imagen parcial de proceso "IPP OB Servo"

La memoria imagen parcial de proceso "IPP OB Servo" está disponible para Motion Control de forma isócrona. Todos los accionamientos y encóders utilizados por Motion Control se asignan a esta memoria imagen parcial de proceso.

También deberían asignarse a la misma todos los módulos E/S utilizados por Motion Control (p. ej. finales de carrera por hardware). Por medio de la asignación se realiza un procesamiento síncrono en el tiempo con el objeto tecnológico.

La memoria imagen parcial de proceso de las entradas también se actualiza en STOP.

Memoria imagen parcial de proceso en el programa de usuario

A partir de la versión V1.5 de la CPU es posible acceder a la memoria imagen parcial de proceso "IPP OB Servo" desde el programa de usuario mediante direccionamiento simbólico. Esto permite también evaluar la memoria imagen parcial de proceso utilizando la función Trace y analizador lógico.

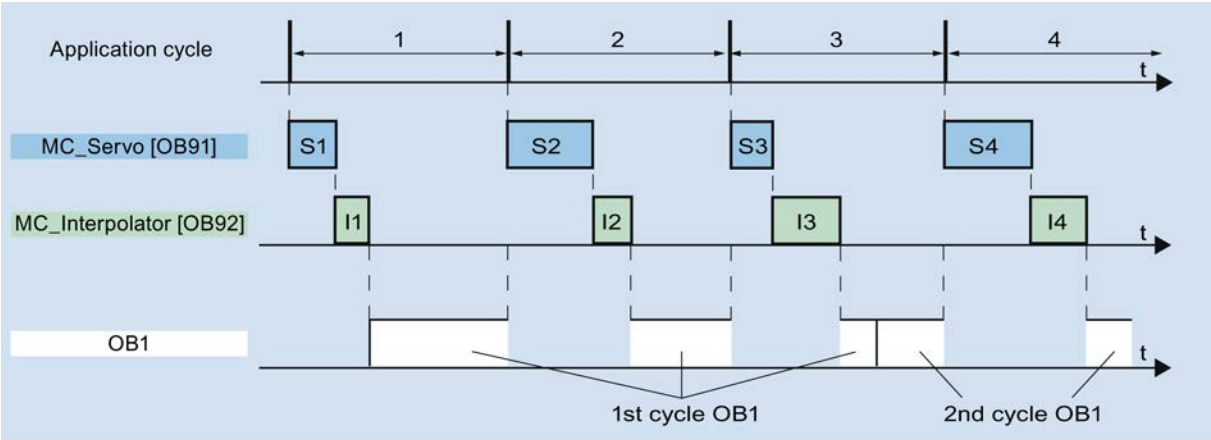
Comportamiento de ejecución y desbordamientos

Durante el procesamiento de la funcionalidad Motion Control se llaman y procesan los bloques de organización MC-Servo [OB91] y MC-Interpolator [OB92] en cada ciclo de aplicación. El tiempo de ciclo restante está disponible para procesar el programa de usuario.

Para que el programa se ejecute correctamente se aplican las siguientes reglas:

- En cada ciclo de aplicación (application cycle) debe iniciarse el MC-Servo [OB91] y procesarse hasta el final.
- En cada ciclo de aplicación debe iniciarse por lo menos el MC-Interpolator [OB92] correspondiente.

La figura siguiente muestra a modo de ejemplo el comportamiento de ejecución correcto para el procesamiento del bloque de organización OB1:

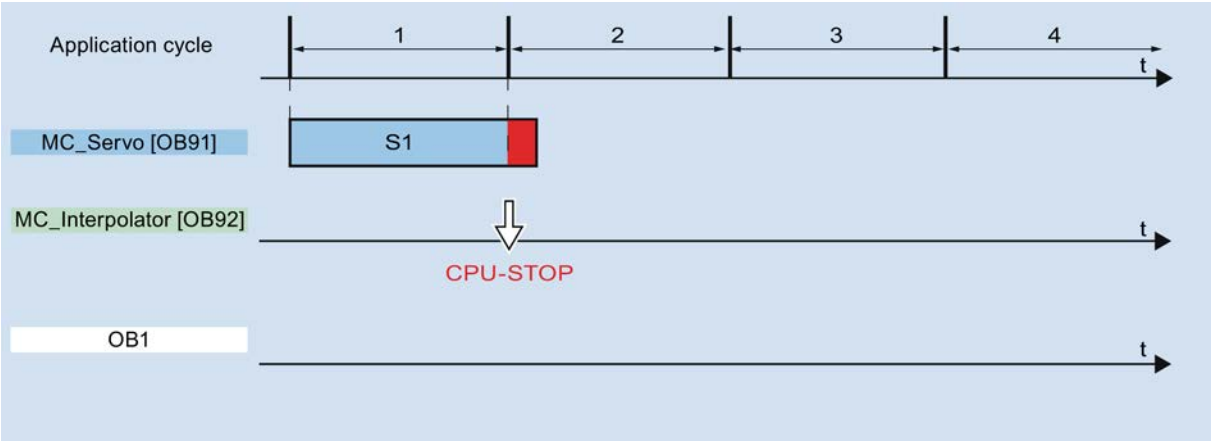


Desbordamientos

Si no se mantiene el ciclo de aplicación ajustado, p. ej. porque es demasiado corto, es posible que se produzcan desbordamientos.

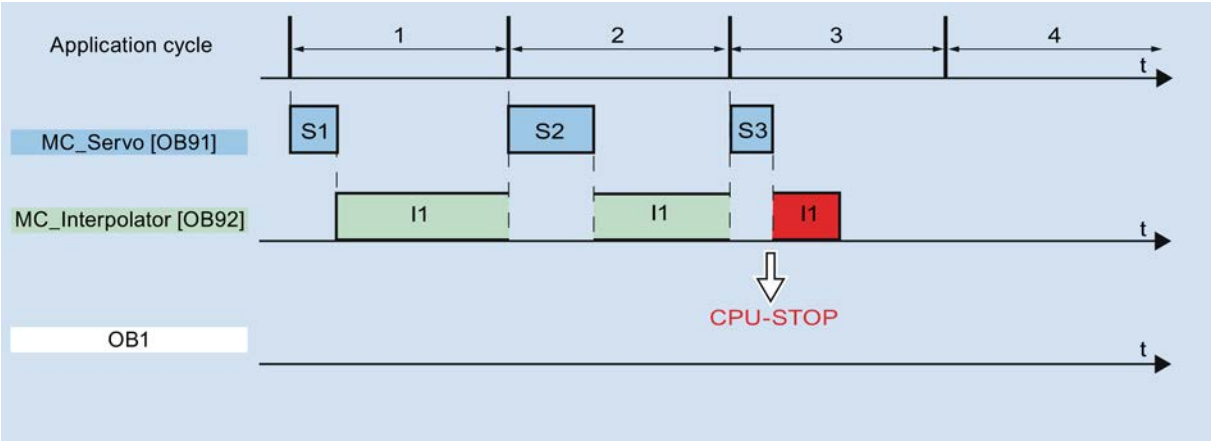
La CPU no tolera desbordamientos del MC-Servo [OB91]. En caso de desbordamiento, la CPU pasa al estado operativo STOP.

La figura siguiente muestra el comportamiento de ejecución en caso de desbordamiento del MC-Servo [OB91]:



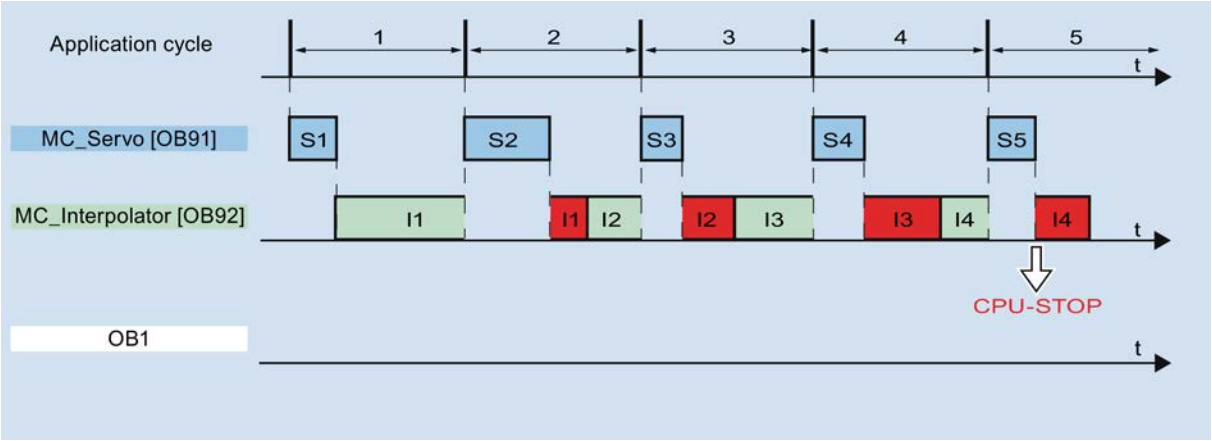
El procesamiento de un MC-Interpolator [OB92] puede interrumpirse como mucho por una llamada del MC-Servo [OB91]. Si se producen más interrupciones, la CPU pasará al estado operativo STOP.

La figura siguiente muestra el comportamiento de ejecución en caso de interrupción de un MC-Interpolator [OB92] en dos segmentos de tiempo:



La CPU tolera como máximo tres desbordamientos consecutivos del MC-Interpolator [OB92]. Si se producen más desbordamientos, la CPU pasará al estado operativo STOP.

La figura siguiente muestra el comportamiento de ejecución en caso de cuatro desbordamientos individuales consecutivos del MC-Interpolator [OB92]:

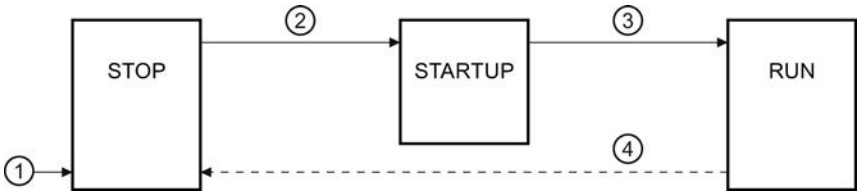


Estados operativos

En este apartado se explica el comportamiento de Motion Control en los diferentes estados operativos y en las transiciones entre los mismos. Encontrará una descripción general de los estados operativos en el manual de sistema del S7-1500.

Estados operativos y transiciones

La CPU tiene tres estados operativos: STOP, ARRANQUE (STARTUP) y RUN. La figura siguiente muestra los estados operativos y las transiciones entre los mismos:



Estado operativo STOP

En estado operativo STOP no se procesa el programa de usuario y todas las salidas del proceso se desactivan. Por consiguiente, no se ejecutan peticiones de Motion Control.

Los bloques de datos tecnológicos se actualizan.

Estado operativo ARRANQUE (STARTUP)

Antes de que la CPU empiece a procesar el programa de usuario cíclico, se ejecutan una vez los OBs de arranque.

Durante el ARRANQUE, las salidas del proceso están bloqueadas. Las peticiones de Motion Control se rechazan.

Los bloques de datos tecnológicos se actualizan.

Estado operativo RUN

En estado operativo RUN se procesa el programa de usuario.

En RUN se llaman y procesan cíclicamente las peticiones programadas de Motion Control.

Los bloques de datos tecnológicos se actualizan.

Transiciones de estados operativos

La tabla siguiente muestra el comportamiento de Motion Control en las transiciones entre los estados operativos:

N.º	Transición de estado operativo	Comportamiento
①	POWER ON → STOP	La CPU ejecuta un rearranque completo de los objetos tecnológicos. Los objetos tecnológicos se reinicializan con los valores de la memoria de carga.
②	STOP → ARRANQUE	Irrelevante para Motion Control.
③	ARRANQUE → RUN	Las salidas de proceso se habilitan.
④	RUN → STOP	Cuando la CPU pasa del estado operativo RUN a STOP, todos los objetos tecnológicos se bloquean conforme a la reacción a alarma "retirar habilitación". Las peticiones de Motion Control en curso se cancelan.

Guía

Guía para utilizar Motion Control

La guía aquí descrita muestra el procedimiento básico para utilizar Motion Control con la CPU S7-1500. Dicha guía sirve de referencia.

Requisitos

- Se ha creado un proyecto con una CPU S7-1500.

Procedimiento

Para utilizar Motion Control con la CPU S7-1500, proceda del siguiente modo:

1. Agregar objeto tecnológico (Página 10698)
2. Trabajar con el editor de configuración (Página 10699)
3. Programar (Página 10750)
4. Cargar en CPU (Página 10772)
5. Test de funcionamiento en la ventana de puesta en servicio (Página 10773)
6. Diagnóstico (Página 10784)

Configurar

Agregar y configurar accionamientos en la configuración de dispositivos

Agregar y configurar accionamientos PROFINET IO

A continuación se describe la forma de agregar y configurar un accionamiento PROFINET IO con ayuda de un **accionamiento SINAMICS S120**. La forma de agregar y configurar otros accionamientos PROFINET IO puede diferir en puntos concretos de la descripción mostrada.

Requisitos

- El dispositivo SIMATIC S7-1500 está creado en el proyecto.
- El accionamiento deseado está disponible en el catálogo de hardware.

Si el accionamiento no se encuentra en el catálogo de hardware, deberá instalarse como archivo de descripción del dispositivo (GSD) en el menú "Opciones".

Agregar accionamiento y trama en la configuración de dispositivos

1. Abra la configuración del dispositivo y cambie a la vista de redes.
2. Abra la carpeta "Otros dispositivos de campo > PROFINET IO > Drives > Siemens AG > SINAMICS" del catálogo de hardware.

3. Seleccione el accionamiento de la versión deseada y arrástrelo a la vista de redes por medio de Drag&Drop.
4. Asigne el accionamiento a la interfaz PROFINET del PLC.
5. Abra el accionamiento en la vista de dispositivos.
6. Mediante Drag&Drop arrastre un Drive Object (DO) y una trama estándar desde el catálogo de hardware a un slot de la vista general de dispositivos del accionamiento.

Dependiendo de la versión del accionamiento SINAMICS S120 será necesario seleccionar la trama "DO con trama estándar X" o bien "DO Servo" y una "trama estándar X".

Encontrará más información sobre las tramas adecuadas en el capítulo Telegramas (Página 10647).

Repita el paso 6 si desea agregar otro accionamiento y otra trama estándar.

Activar el modo isócrono del accionamiento en la configuración de dispositivos

Básicamente los accionamientos PROFINET pueden funcionar en modo isócrono o en modo asíncrono. El modo isócrono aumenta la calidad de la regulación de posición del accionamiento.

Si desea activar el accionamiento en modo isócrono, proceda del modo siguiente:

1. Seleccione la vista de dispositivos del accionamiento.
2. En el cuadro de diálogo de propiedades seleccione la ficha "Interfaz PROFINET [X1] > Opciones avanzadas > Modo isócrono".
3. En esa ficha active la casilla de verificación "Modo isócrono".

En la vista de detalle también debe estar seleccionada la entrada de la trama para el modo isócrono.

Conectar el puerto del PLC con el puerto del accionamiento

1. Abra la vista topológica de la configuración de dispositivos.
2. Conecte el puerto del PLC con el puerto del accionamiento.

Configurar el PLC como maestro Sync y ajustar un ciclo isócrono

1. Seleccione la vista de dispositivos del PLC.
2. En el cuadro de diálogo de propiedades seleccione la ficha "Interfaz PROFINET [X1] > Opciones avanzadas > Configuración en tiempo real > Sincronización".
3. Seleccione la entrada "Maestro Sync" en la lista desplegable "Función de sincronización".
4. Confirme haciendo clic en el botón "Ajustes del dominio".
5. Seleccione la ficha "Administración de dominios > Dominios Sync" y ajuste el "Tiempo de ciclo de emisión" deseado (ciclo isócrono).

Seleccionar un accionamiento en la configuración del objeto tecnológico

1. Agregue un objeto tecnológico Eje nuevo, o abra la configuración de un eje ya existente.
2. Abra la configuración " Interfaz de hardware > Accionamiento".
3. Seleccione la entrada "PROFIdrive" en la lista desplegable "Tipo de accionamiento".
4. En la lista "Accionamiento", seleccione Drive Object del accionamiento PROFINET.

Si quiere saber cómo agregar un objeto tecnológico, consulte el capítulo Agregar objeto tecnológico (Página 10698).

Resultado

El objeto tecnológico está asociado al accionamiento y puede revisarse o configurarse el bloque de organización "MC-Servo".

La trama del accionamiento configurado se asigna a la memoria imagen del proceso "IPP OB Servo".

Comprobar y configurar las propiedades de MC-Servo

1. Abra la carpeta "Bloques de programa" en el árbol del proyecto.
2. Seleccione el bloque de organización "MC-Servo".
3. Seleccione el comando del menú contextual "Propiedades".
4. Seleccione la entrada "Tiempo de ciclo" en la navegación local.
5. En el cuadro de diálogo siguiente debe estar seleccionada la opción "Sincronía con el bus".
6. En la lista desplegable "Periferia descentralizada" debe estar seleccionado un "Sistema PROFINET IO".
7. El ciclo de aplicación del "MC-Servo" debe ser equivalente al tiempo de ciclo de emisión del bus o bien estar reducido en un número entero al tiempo de ciclo de emisión del bus.

Resultado

El accionamiento PROFINET IO está configurado de modo que puede ser activado en modo isócrono en la red PROFINET IO.

Las propiedades del accionamiento SINAMICS deben configurarse conforme a la configuración del eje con un programa STARTER independiente.

Comprobar la sincronización de ciclo en el accionamiento

Si en la configuración del eje no se ha respetado la secuencia anterior y al compilar el proyecto se producen errores específicos del accionamiento, puede revisarse el ajuste del modo isócrono en el accionamiento.

1. Seleccione la vista de dispositivos del accionamiento.
2. Seleccione la entrada de la trama estándar en la vista general de dispositivos.

3. Seleccione el cuadro de diálogo de propiedades "General > Direcciones E/S".
4. Para las direcciones de entrada y salida se aplican los ajustes siguientes:
   – Está activado "Modo isócrono".
   – Para "Bloque de organización" debe estar seleccionado "MC-Servo".
   – Para "Memoria imagen de proceso" debe estar seleccionado "IPP OB Servo".

Consulte también

Telegramas (Página 10647)

Agregar objeto tecnológico (Página 10698)

Agregar y configurar accionamientos PROFIBUS DP

A continuación se describe la forma de agregar y configurar un accionamiento PROFIBUS con ayuda de un **SINAMICS S120**. La forma de agregar y configurar otros accionamientos PROFIBUS puede diferir en puntos concretos de la descripción mostrada.

Requisitos

- El dispositivo SIMATIC S7-1500 está creado en el proyecto.
- Se dispone de conocimientos básicos sobre la configuración de redes PROFIBUS DP.
- El accionamiento deseado está disponible en el catálogo de hardware.

Si el accionamiento no se encuentra en el catálogo de hardware, deberá instalarse como archivo de descripción del dispositivo (GSD) en el menú "Opciones".

Agregar accionamiento y trama en la configuración de dispositivos

1. Abra la configuración del dispositivo y cambie a la vista de redes.
2. Abra la carpeta "Otros dispositivos de campo > PROFIBUS DP > Accionamientos > Siemens AG > SINAMICS" del catálogo de hardware.
3. Seleccione la carpeta del accionamiento deseado en la versión deseada y arrastre el objeto del accionamiento hasta la vista de redes mediante Drag&Drop.
4. Asigne el accionamiento a la interfaz PROFIBUS del PLC.
5. Abra el accionamiento en la vista de dispositivos.
6. Mediante Drag&Drop, arrastre una trama estándar desde el catálogo de hardware a un slot de la vista general de dispositivos del accionamiento.

Encontrará más información sobre las tramas adecuadas en el capítulo Telegramas (Página 10647).

Utilice el "separador de ejes" del catálogo de hardware si desea agregar otro accionamiento y otra trama estándar a la vista general de dispositivos.

Activar el modo isócrono del accionamiento en la configuración de dispositivos

Básicamente los accionamientos PROFIBUS pueden funcionar en modo cíclico o isócrono. El modo isócrono aumenta la calidad de la regulación de posición del accionamiento.

Si desea activar el accionamiento en modo isócrono, proceda del modo siguiente:

1. Seleccione la vista de dispositivos del accionamiento.
2. Seleccione la ficha "General > Modo isócrono" en el cuadro de diálogo de propiedades.
3. Active la casilla de verificación "Sincronizar el esclavo DP con ciclo DP equidistante".

Ajustar el ciclo isócrono

1. Seleccione la vista de redes.
2. Seleccione el sistema maestro DP.
3. Seleccione la ficha "General > Equidistancia" en el cuadro de diálogo de propiedades.
4. Seleccione el modo "Ciclo DP equidistante" deseado.

Seleccionar un accionamiento en la configuración del objeto tecnológico

1. Agregue un objeto tecnológico Eje nuevo, o abra la configuración de un eje ya existente.
2. Abra la configuración " Interfaz de hardware > Accionamiento".
3. Seleccione la entrada "PROFIdrive" en la lista desplegable "Tipo de accionamiento".
4. En la lista "Accionamiento", seleccione la trama estándar del accionamiento PROFIBUS.

Si quiere saber cómo agregar un objeto tecnológico, consulte el capítulo Agregar objeto tecnológico (Página 10698).

Resultado

El objeto tecnológico está asociado al accionamiento, y puede revisarse o configurarse el bloque de organización "MC-Servo".

La trama del accionamiento configurado se asigna a la memoria imagen del proceso "IPP OB Servo".

Comprobar y configurar las propiedades de MC-Servo

1. Abra la carpeta "Bloques de programa" en el árbol del proyecto.
2. Seleccione el bloque de organización "MC-Servo".
3. Seleccione el comando del menú contextual "Propiedades".
4. Seleccione la entrada "Tiempo de ciclo" en la navegación local.
5. En el cuadro de diálogo siguiente debe estar seleccionada la opción "Sincronía con el bus".

6. En la lista desplegable "Periferia descentralizada" debe estar seleccionado un "Sistema PROFINET IO".
7. El ciclo de aplicación del "MC-Servo" debe ser equivalente al tiempo de ciclo de emisión del bus o bien estar reducido en un número entero al tiempo de ciclo de emisión del bus.

En la configuración del objeto tecnológico es posible seleccionar un accionamiento conectado a la CPU por medio de un procesador de comunicaciones / módulo de comunicaciones (CP/CM). El sistema maestro DP del CP/CM no puede seleccionarse como origen del ciclo del MC-Servo [OB91].

Resultado

El accionamiento PROFIBUS DP está configurado de modo que puede ser activado en modo isócrono en la red PROFIBUS.

Las propiedades del accionamiento SINAMICS deben configurarse conforme a la configuración del eje con un programa STARTER independiente.

Comprobar la sincronización de ciclo en el accionamiento

Si en la configuración del eje no se ha respetado la secuencia anterior y al compilar el proyecto se producen errores específicos del accionamiento, puede revisarse el modo isócrono en el accionamiento.

1. Seleccione la vista de dispositivos del accionamiento.
2. Seleccione la entrada de la trama estándar en la vista general de dispositivos.
3. Seleccione el cuadro de diálogo de propiedades "General > Direcciones E/S".
4. Para las direcciones de entrada y salida se aplican los ajustes siguientes:
   - Para "Bloque de organización" debe estar seleccionado "MC-Servo".
   - Para "Memoria imagen de proceso" debe estar seleccionado "IPP OB Servo".

Consulte también

Agregar objeto tecnológico (Página 10698)

Telegramas (Página 10647)

Agregar y configurar accionamientos con conexión analógica del accionamiento

A continuación se describe cómo agregar y configurar un accionamiento con conexión analógica. La conexión de un encóder incremental se produce, por ejemplo, a través de un módulo tecnológico en el rack del PLC.

Requisitos

El dispositivo SIMATIC S7-1500 está creado en el proyecto.

Agregar y configurar un módulo de salidas analógicas en la configuración de dispositivos

1. Abra la configuración de dispositivos del PLC.
2. Seleccione un módulo de salidas analógicas del catálogo de hardware y arrástrelo mediante Drag&Drop hasta el rack del PLC.
3. Seleccione el módulo de salidas analógicas en la vista general de dispositivos.
4. En el cuadro de diálogo de propiedades seleccione la ficha "General", y dentro de ella "Nombre del módulo de salidas analógicas > Direcciones E/S".
5. Introduzca la dirección de inicio deseada.
6. En el cuadro de diálogo de propiedades seleccione la ficha "Variables I/O".
7. Asigne un nombre de variable a la salida analógica deseada.

Agregar y configurar un módulo tecnológico

1. Cambie a la vista de dispositivos del PLC.
2. En el catálogo de hardware abra la carpeta "TM > Contaje > TM Count 2X24V".
3. Arrastre el módulo de contador hasta un slot libre del rack mediante Drag&Drop.
4. Seleccione el módulo tecnológico en la vista de dispositivos.
5. En el cuadro de diálogo de propiedades abra la ficha "General", y dentro de la misma "Count 2x24V > Parámetros básicos > Canal X > Modo de operación" del canal que va a emplearse.
6. En "Selección del modo de operación del canal" seleccione la opción "Registro de posición para Motion Control".
7. En "Parámetros del módulo" adapte los parámetros del encóder incremental (incrementos por vuelta).
8. En "Reacción a STOP de la CPU" elija la entrada "Continuar".

Seleccionar accionamiento y encóder en la configuración del objeto tecnológico

1. Agregue un objeto tecnológico "Eje de posicionamiento" o abra la configuración de un eje de posicionamiento ya existente.
2. Abra la configuración " Interfaz de hardware > Accionamiento".
3. Seleccione la entrada "Conexión analógica del accionamiento" en la lista desplegable "Tipo de accionamiento".
4. En la lista "Salida", seleccione el nombre de variable definido previamente para la salida analógica.
5. Abra la configuración " Interfaz de hardware > Encóder".
6. En "Seleccionar acoplamiento del encóder" elija la opción "Conexión vía módulo tecnológico (TM)".
7. En "local modules" de la lista "Módulo tecnológico", elija el canal del encóder incremental.

Puede consultar en el capítulo Agregar objeto tecnológico (Página 10698) cómo agregar un objeto tecnológico.

Resultado

La conexión analógica del accionamiento y la conexión del encóder están configuradas.

Las direcciones analógicas y las direcciones del módulo TM se asignan a la memoria imagen de proceso "IPP OB Servo".

Comprobar la conexión del encóder o del accionamiento

Los datos del encóder se aplican en el ciclo de la regulación de posición. En caso de duda compruebe los ajustes siguientes:

1. Cambie a la vista de dispositivos del PLC.
2. Seleccione el módulo tecnológico.
3. Abra el cuadro de diálogo de propiedades "Parámetros básicos > Direcciones E/S".
4. Para las direcciones de entrada y salida se aplican los ajustes siguientes:
   – Para "Bloque de organización" debe estar seleccionado "MC-Servo".
   – Para "Memoria imagen de proceso" debe estar seleccionado "IPP OB Servo".
5. Seleccione el módulo analógico.
6. Abra el cuadro de diálogo de propiedades "Nombre del módulo analógico > Direcciones E/S".
7. Para las direcciones de entrada y salida se aplican los ajustes del paso 4.

Consulte también

Agregar objeto tecnológico (Página 10698)

Agregar objeto tecnológico

Para agregar un objeto tecnológico en el árbol del proyecto, proceda del siguiente modo:

Requisitos

Se ha creado un proyecto con una CPU S7-1500.

Procedimiento

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta Objetos tecnológicos.
3. Haga doble clic en "Agregar objeto".
   Se abre el cuadro de diálogo "Agregar objeto".
4. Seleccione la tecnología "Motion Control".
5. Abra la carpeta "Motion Control" > "S7-1500 Motion Control".
6. Haga clic en la entrada de la versión y seleccione una versión alternativa de la tecnología por si deseara agregar un eje de una versión anterior.

7. Elija el objeto "TO_SpeedAxis" para un eje de velocidad de giro, "TO_PositioningAxis" para un eje de posicionamiento o "TO_ExternalEncoder" para un encóder externo.
8. Adapte el nombre del eje a sus necesidades en el campo de entrada "Nombre".
9. Seleccione la opción "Manual" si desea modificar el número de bloque de datos propuesto.
10. Haga clic en "Más información" si desea complementar el objeto tecnológico con información propia.
11. Haga clic en el botón "Aceptar" para agregar el objeto tecnológico.
    Haga clic en el botón "Cancelar" para rechazar las entradas.
    Active la casilla de verificación "Agregar y abrir" si desea abrir la configuración después de agregar el objeto tecnológico.

Resultado

Se ha generado el nuevo objeto tecnológico y se ha guardado en la carpeta "Objetos tecnológicos" del árbol del proyecto.

Si todavía no se habían agregado los bloques de organización "MC-Servo" y "MC-Interpolator", ahora ya se han agregado.

Trabajar con el editor de configuración

Las propiedades de un objeto tecnológico se configuran en la ventana de configuración. Para abrir la ventana de configuración del objeto tecnológico en la vista del proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, abra el grupo "Objetos tecnológicos" del dispositivo.
2. Seleccione el objeto tecnológico y haga doble clic en "Configuración".

La configuración está dividida en categorías según el tipo de objeto, p. ej.: parámetros básicos, interfaz de hardware o parámetros avanzados.

Símbolos del editor de configuración

Los símbolos que aparecen en la navegación local de la configuración muestran más detalles sobre el estado de la configuración:

	La configuración contiene valores predeterminados y está completa. La configuración contiene exclusivamente valores predeterminados. Con ellos es posible utilizar el objeto tecnológico sin más modificaciones.
	La configuración contiene valores definidos por el usuario y está completa Todos los campos de entrada de la configuración contienen valores válidos y al menos un valor predeterminado ha sido modificado.
	La configuración es errónea o está incompleta Al menos un campo de entrada o una lista desplegable contiene un valor no válido. El campo o la lista desplegable en cuestión se marca en rojo. Al hacer clic en este campo, el roll out con el aviso de error indica la causa del error.

Comparar valores

La función "Comparar valores" ofrece las siguientes opciones:

- Comparación de los valores de arranque configurados del proyecto con los valores de arranque de la CPU y los valores actuales
- Procesamiento directo de los valores actuales y de los valores de arranque del proyecto
- Detección y visualización inmediatas de errores de introducción con propuestas de corrección
- Copia de seguridad de los valores actuales en el proyecto
- Transferencia de los valores de arranque del proyecto a la CPU como valores actuales

Símbolos y elementos de control

Están disponibles los siguientes símbolos y elementos de control:

Símbolo	Función
	El valor de arranque en la CPU es igual al valor de arranque configurado en el proyecto
	El valor de arranque en la CPU es diferente al valor de arranque configurado en el proyecto
	No se puede realizar la comparación entre valor de arranque en la CPU y valor de arranque configurado en el proyecto
	Se guarda una copia de seguridad de los valores actuales en el proyecto
	Los valores de arranque actualizados en el proyecto se transfieren a la CPU (inicializar valores de ajuste)
	Se abre el cuadro de diálogo "Comparar valores"

Configurar el objeto tecnológico Eje de posicionamiento

Configuración - parámetros básicos

Nombre del eje

Defina en este campo el nombre del eje de posicionamiento. El objeto tecnológico aparece con ese nombre en el árbol del proyecto. Las variables del eje de posicionamiento pueden utilizarse en el programa de usuario con este nombre.

Tipo de eje

Configure en esta selección si el eje debe realizar movimientos lineales o rotativos.

Unidad de usuario

Seleccione en las listas desplegables los sistemas de medida deseados para la posición y velocidad del eje.

Módulo

Active la casilla de verificación "Activar módulo" si desea utilizar un sistema de medida repetitivo para el eje (p. ej. 0-360° con un eje del tipo "rotativo").

- Valor inicial
  En este campo, defina la posición en la que debe empezar el rango del módulo (p. ej. 360° con un eje del tipo "rotativo").
- Longitud
  En este campo, defina la longitud del rango del módulo (p. ej. 360° con un eje del tipo "rotativo").

Interfaz de hardware

Configuración - accionamiento

En la ventana de configuración "Accionamiento", configure el tipo y el accionamiento que desea utilizar.

Tipo de accionamiento

Seleccione en la lista desplegable si desea utilizar un accionamiento PROFIdrive o un accionamiento con conexión analógica.

Los accionamientos PROFIdrive se conectan al controlador por medio de un sistema de comunicación digital (PROFINET o PROFIBUS). La comunicación se realiza utilizando telegramas PROFIdrive.

Los accionamientos con conexión analógica obtienen la consigna de velocidad de giro a través de una señal de salida analógica (p. ej. de -10 V a +10 V) del PLC.

Accionamiento (tipo de accionamiento: PROFIdrive)

Seleccione un accionamiento PROFIdrive / slot previamente configurado en el campo "Accionamiento". En caso de seleccionar un accionamiento PROFIdrive, puede configurarse mediante el botón "Configuración del dispositivo".

Cambie a la configuración del dispositivo y agregue un accionamiento PROFIdrive en la vista de redes si no hubiera ninguno disponible.

Salida (tipo de accionamiento: conexión analógica del accionamiento)

En el campo "Salida", seleccione la variable PLC de la salida analógica con la que debe controlarse el accionamiento. En caso de seleccionar una salida, puede configurarse mediante el botón "Configuración del dispositivo".

Para poder seleccionar una salida es necesario que en la configuración del dispositivo se haya agregado un módulo de salidas analógicas y se haya definido el nombre de variable PLC para la salida analógica.

Activar salida de habilitación (tipo de accionamiento: conexión analógica del accionamiento)

En el campo "Salida de habilitación" seleccione la variable PLC de la salida digital para habilitar el accionamiento. La salida de habilitación habilita o bloquea el regulador de velocidad de giro del accionamiento.

Para poder seleccionar una salida de habilitación es necesario que en la configuración del dispositivo se haya agregado un módulo de salidas digitales y se haya definido el nombre de variable PLC para la salida digital.

Nota

Si no se utiliza ninguna salida de habilitación, el sistema no puede bloquear el accionamiento directamente como consecuencia de reacciones a errores o funciones de vigilancia. No se garantiza una parada controlada del accionamiento.

Activar entrada de disponibilidad (tipo de accionamiento: conexión analógica del accionamiento)

En el campo "Entrada de disponibilidad", seleccione la variable PLC de la entrada digital a través de la cual el accionamiento notifica su disponibilidad al objeto tecnológico. La etapa de potencia está activada y la entrada analógica de la consigna de velocidad de giro está activa.

Para poder seleccionar una entrada de disponibilidad es necesario que en la configuración del dispositivo se haya agregado un módulo de entradas digitales y se haya definido el nombre de variable PLC para la entrada digital.

Nota

La salida de habilitación y la entrada de disponibilidad pueden activarse de forma independiente entre sí.

Para la entrada de disponibilidad activada rigen las condiciones marco siguientes:

- El eje no se habilita (MC_Power Status=TRUE) hasta que hay una señal en la entrada de disponibilidad.
- Si en un eje habilitado desaparece la señal de la entrada de disponibilidad, el eje se bloquea con un error.
- Si el eje se bloquea mediante la instrucción MC_Power (Enable= FALSE), se bloquea también con una señal pendiente en la entrada de disponibilidad.

Configuración - encóder

Para la regulación de posición, los ejes de posicionamiento requieren un valor real de posición en forma de posición del encóder. La posición del encóder se transfiere al controlador mediante un telegrama PROFIdrive. En la ventana de configuración "Encóder", configure el acoplamiento para conectar el encóder.

Seleccionar acoplamiento del encóder:

Seleccione en esta área cómo debe conectarse el encóder de acuerdo con la representación gráfica.

- **Conexión con accionamiento (no en el caso de conexión analógica del accionamiento)**
  Seleccione esta opción si el encóder debe conectarse al accionamiento. El encóder se configura en la configuración del accionamiento PROFIdrive. El accionamiento evalúa las señales del encóder y las envía al controlador en el telegrama PROFIdrive.
- **Conexión vía módulo tecnológico (TM)**
  Seleccione esta opción si el encóder debe conectarse a un módulo tecnológico (TM).
  - **Módulo tecnológico:**
    En el campo "Módulo tecnológico" seleccione un módulo tecnológico previamente configurado y el canal que debe utilizarse. Para la selección solo se muestran módulos tecnológicos ajustados al modo de operación "Lectura de posición para Motion Control". Si no hubiera ningún módulo tecnológico disponible, cambie a la configuración del dispositivo y agregue uno. Una vez se ha seleccionado un módulo tecnológico se accede a la configuración del mismo mediante el botón "Configuración del dispositivo". El módulo tecnológico puede utilizarse de forma centralizada en un PLC S7-1500 o de forma descentralizada en una periferia descentralizada. Para conectar un encóder incremental utilice el módulo tecnológico TM Count2x24V o TM PosInput2. Para conectar un encóder absoluto utilice el módulo tecnológico TM PosInput2.
- **Conexión vía PROFINET/PROFIBUS (PROFIdrive)**
  Seleccione esta opción si se utiliza un encóder compatible con PROFIdrive.
  - **Selección del encóder:**
    Seleccione un encóder previamente configurado en PROFINET/PROFIBUS en el campo "Selección del encóder". En caso de seleccionar un encóder, puede configurarse mediante el botón "Configuración del dispositivo".
    Cambie a la vista de redes en la configuración del dispositivo y agregue un encóder si no hubiera ninguno disponible.

Configuración - Intercambio de datos

Configuración - Intercambio de datos

Configure el intercambio de datos con el accionamiento y el encóder en la ventana de configuración "Intercambio de datos". La configuración es diferente según el tipo de accionamiento y el acoplamiento del encóder:

Conexión analógica del accionamiento - Conexión vía módulo tecnológico (TM) (Página 10704)

Conexión analógica del accionamiento - Conexión vía PROFINET/PROFIBUS (Página 10706)

Accionamiento PROFIdrive - encóder con accionamiento (Página 10708)

Accionamiento PROFIdrive - Conexión vía módulo tecnológico (TM) (Página 10710)

Accionamiento PROFIdrive - Conexión vía PROFINET/PROFIBUS (Página 10712)

Conexión analógica del accionamiento - Conexión vía módulo tecnológico (TM)

Intercambio de datos con el accionamiento

Configure en esta área el intercambio de datos con el accionamiento.

- Telegrama:
  Puesto que el accionamiento analógico se controla mediante una señal analógica, no es necesario seleccionar un tipo de telegrama.
- Velocidad de giro de referencia:
  La velocidad de giro de referencia del accionamiento es la velocidad con la que gira el accionamiento en la salida analógica con una salida del 100%. La velocidad de giro de referencia debe configurarse en el accionamiento y aplicarse en la configuración del objeto tecnológico.
  El valor analógico emitido con el 100% depende del tipo de salida analógica. Así, por ejemplo, en una salida analógica de +/- 10 V al 100% se emite el valor 10 V.
  Las salidas analógicas pueden saturarse en un 17% aproximadamente. Esto significa que una salida analógica puede utilizarse en un rango de -117% a 117%, siempre que el accionamiento lo permita.
- Velocidad de giro máxima
  Introduzca en este campo la velocidad de giro máxima del accionamiento.
- Invertir sentido
  Active esta casilla de verificación si debe invertirse el sentido de giro del accionamiento.

Intercambio de datos con el encóder

Configure en esta área cómo deben evaluarse los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

- Telegrama:
  Los datos del encóder se transfieren con el telegrama del accionamiento. Por este motivo no es posible seleccionar un telegrama.
- Tipo de encóder:
  Configure los parámetros descritos a continuación según sea el tipo de encóder seleccionado:

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Conexión analógica del accionamiento - Conexión vía PROFINET/PROFIBUS

Intercambio de datos con el accionamiento

Configure en esta área el intercambio de datos con el accionamiento.

- **Telegrama:**
  Puesto que el accionamiento analógico se controla mediante una señal analógica, no es necesario seleccionar un tipo de telegrama.
- **Velocidad de giro de referencia:**
  La velocidad de giro de referencia del accionamiento es la velocidad con la que gira el accionamiento en la salida analógica con una salida del 100%. La velocidad de giro de referencia debe configurarse en el accionamiento y aplicarse en la configuración del objeto tecnológico.
  El valor analógico emitido con el 100% depende del tipo de salida analógica. Así, por ejemplo, en una salida analógica de +/- 10 V al 100% se emite el valor 10 V.
  Las salidas analógicas pueden saturarse en un 17% aproximadamente. Esto significa que una salida analógica puede utilizarse en un rango de -117% a 117%, siempre que el accionamiento lo permita.
- **Velocidad de giro máxima**
  Introduzca en este campo la velocidad de giro máxima del accionamiento.
- **Invertir sentido**
  Active esta casilla de verificación si debe invertirse el sentido de giro del accionamiento.

Intercambio de datos con el encóder

Configure en esta área el telegrama del encóder y los criterios para evaluar los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

- **Telegrama:**
  Seleccione el telegrama del encóder en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.
- **Tipo de encóder:**
  Configure los parámetros descritos a continuación según sea el tipo de encóder seleccionado:

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para resolución fina en valor real absoluto (Gn_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para resolución fina en valor real absoluto (Gn_XIST2)	Configure en este campo el número de bits para la resolución fina dentro del valor real absoluto (Gn_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para resolución fina en valor real absoluto (Gn_XIST2)	Configure en este campo el número de bits para la resolución fina dentro del valor real absoluto (Gn_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Accionamiento PROFIdrive - encóder con accionamiento

Intercambio de datos con el accionamiento

Configure en esta área el intercambio de datos con el accionamiento.

- **Telegrama:**
  Seleccione el telegrama hacia el accionamiento en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.
- **Velocidad de giro de referencia:**
  Configure en este campo la velocidad de giro de referencia del accionamiento de acuerdo con las indicaciones del fabricante. La velocidad de giro del accionamiento se especifica porcentualmente a la velocidad de giro de referencia en un rango de -200% a 200%.
- **Velocidad de giro máxima**
  Introduzca en este campo la velocidad de giro máxima del accionamiento.
- **Invertir sentido**
  Active esta casilla de verificación si debe invertirse el sentido de giro del accionamiento.

Intercambio de datos con el encóder

Configure en esta área cómo deben evaluarse los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

- **Telegrama:**
  Los datos del encóder se transfieren con el telegrama del accionamiento. Por este motivo no es posible seleccionar un telegrama.
- **Tipo de encóder:**
  Configure los parámetros descritos a continuación según sea el tipo de encóder seleccionado:

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.

Tipo de encóder	Rotativo absoluto
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Accionamiento PROFIdrive - Conexión vía módulo tecnológico (TM)

Intercambio de datos con el accionamiento

Configure en esta área el intercambio de datos con el accionamiento.

- Telegrama:
  Seleccione el telegrama hacia el accionamiento en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.
- Velocidad de giro de referencia:
  Configure en este campo la velocidad de giro de referencia del accionamiento de acuerdo con las indicaciones del fabricante. La velocidad de giro del accionamiento se especifica porcentualmente a la velocidad de giro de referencia en un rango de -200% a 200%.
- Velocidad de giro máxima
  Introduzca en este campo la velocidad de giro máxima del accionamiento.
- Invertir sentido
  Active esta casilla de verificación si debe invertirse el sentido de giro del accionamiento.

Intercambio de datos con el encóder

Configure en esta área cómo deben evaluarse los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

- Telegrama:
  Los datos del encóder se transfieren con el telegrama del accionamiento. Por este motivo no es posible seleccionar un telegrama.
- Tipo de encóder:
  Configure los parámetros descritos a continuación según sea el tipo de encóder seleccionado:

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.

Tipo de encóder	Rotativo absoluto
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Accionamiento PROFIdrive - Conexión vía PROFINET/PROFIBUS

Intercambio de datos con el accionamiento

Configure en esta área el intercambio de datos con el accionamiento.

- **Telegrama:**
  Seleccione el telegrama hacia el accionamiento en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.
- **Velocidad de giro de referencia:**
  Configure en este campo la velocidad de giro de referencia del accionamiento de acuerdo con las indicaciones del fabricante. La velocidad de giro del accionamiento se especifica porcentualmente a la velocidad de giro de referencia en un rango de -200% a 200%.
- **Velocidad de giro máxima**
  Introduzca en este campo la velocidad de giro máxima del accionamiento.
- **Invertir sentido**
  Active esta casilla de verificación si debe invertirse el sentido de giro del accionamiento.

Intercambio de datos con el encóder

Configure en esta área el telegrama del encóder y los criterios para evaluar los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

- **Telegrama:**
  Seleccione el telegrama del encóder en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.
- **Tipo de encóder:**
  Configure los parámetros descritos a continuación según sea el tipo de encóder seleccionado:

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.

Tipo de encóder	Rotativo absoluto
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para resolución fina en valor real absoluto (GN_XIST2)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real absoluto (GN_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para resolución fina en valor real absoluto (GN_XIST2)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real absoluto (GN_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para resolución fina en valor real absoluto (GN_XIST2)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real absoluto (GN_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Parámetros avanzados

Configuración - Mecánica

Configuración - Mecánica

Configure el modo de montaje del encóder y la adaptación del valor del encóder a las particularidades mecánicas en la ventana de configuración "Mecánica".

Modo de montaje del encóder

En la lista desplegable, seleccione cómo está montado el encóder en el sistema mecánico. La configuración difiere según el tipo de eje y el modo de montaje del encóder.

Tipo de eje: lineal

Lineal - en el eje del motor (Página 10714)

Lineal - en el lado de la carga (Página 10715)

Lineal - sistema de medida externo (Página 10715)

Tipo de eje: rotativo

Rotativo - en el eje del motor (Página 10716)

Rotativo - en el lado de la carga (Página 10716)

Rotativo - sistema de medida externo (Página 10717)

Lineal - en el eje del motor

El encóder está unido mecánicamente al eje del motor de forma fija. El motor y el encóder forman una unidad.

Reductor de carga

- **Considerar reductor de carga/paso del husillo**
  Active esta casilla de verificación si desea realizar la configuración de acuerdo con el índice de reducción y el paso del husillo.
  Desactive esta casilla de verificación si desea realizar la configuración de acuerdo con las vueltas del motor y el recorrido resultante.
- **Número de vueltas del motor / Número de vueltas de la carga**
  El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.

Parámetros de posición

- **Paso del husillo ("Considerar reductor de carga/paso del husillo" activado)**
  Configure en este campo el trayecto que recorrerá la carga cuando el husillo gire una vuelta.
- **Movimiento de la carga por vuelta del motor**
  Configure en este campo el recorrido de la carga para una vuelta del motor ("Considerar reductor de carga/paso del husillo" desactivado).

Lineal - en el lado de la carga

El encóder está unido mecánicamente al lado de la carga del reductor de forma fija.

Reductor de carga

- **Número de vueltas del motor / Número de vueltas de la carga**
  El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.
  Elija los mismos valores para el número de vueltas del motor y de la carga si no hay ningún reductor de carga.

Parámetros de posición

- **Paso del husillo**
  Configure en este campo el trayecto que recorrerá la carga cuando el husillo gire una vuelta.

Lineal - sistema de medida externo

Un sistema de medida externo proporciona los valores de posición del movimiento lineal de la carga.

Reductor de carga

- **Número de vueltas del motor / Número de vueltas de la carga**
  El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.
  Elija los mismos valores para el número de vueltas del motor y de la carga si no hay ningún reductor de carga.

Parámetros de posición

- **Movimiento de la carga por vuelta del motor**
  Configure en este campo el trayecto que recorrerá la carga cuando el motor gire una vuelta.
- **Recorrido por vuelta del encóder**
  Configure en este campo el recorrido por vuelta del encóder capturado por el sistema de medida externo.

Rotativo - en el eje del motor

El encóder está unido mecánicamente al eje del motor de forma fija. El motor y el encóder forman una unidad.

Reductor de carga

- **Número de vueltas del motor / Número de vueltas de la carga**
  El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.
  Elija los mismos valores para el número de vueltas del motor y de la carga si no hay ningún reductor de carga.

Rotativo - en el lado de la carga

El encóder está unido mecánicamente al lado de la carga del reductor de forma fija.

Reductor de carga

- **Número de vueltas del motor / Número de vueltas de la carga**
  El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.
  Elija los mismos valores para el número de vueltas del motor y de la carga si no hay ningún reductor de carga.

Rotativo - sistema de medida externo

Un sistema de medida externo proporciona los valores de posición del movimiento rotativo de la carga.

Reductor de carga

- **Número de vueltas del motor / Número de vueltas de la carga**
  El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.
  Elija los mismos valores para el número de vueltas del motor y de la carga si no hay ningún reductor de carga.

Parámetros de posición

- **Sistema de medida externo**
  Configure en este campo el valor por vuelta de la carga que indica el sistema de medida externo.

Configuración - Límites de posición

En la ventana de configuración "Límites de posición" configure los finales de carrera por hardware y software del eje.

Activar final de carrera por hardware

Esta casilla de verificación activa la función de los finales de carrera por hardware negativo y positivo. El final de carrera por hardware negativo está en el lado en sentido de marcha negativo y el final de carrera por hardware positivo, en el lado en sentido de marcha positivo.

Con los finales de carrera por hardware activados, el accionamiento se bloquea al aproximarse y sobrepasarlos. El accionamiento frena con la rampa de deceleración configurada en el mismo.

Excepción: si se sobrepasa un final de carrera por hardware durante una marcha activa al punto de referencia con inversión de sentido activada en el final de carrera por hardware, el

eje se para con la deceleración máxima configurada y continúa la marcha al punto de referencia en sentido inverso.

Nota

Utilice solo finales de carrera por hardware que se mantienen activados permanentemente después de la aproximación. Este estado de conmutación no debe retirarse hasta después de regresar al rango de desplazamiento admisible.

Las entradas digitales de los finales de carrera por hardwarwe se evalúan de forma estándar durante el intercambio cíclico de datos. En la configuración del módulo de entradas, seleccione bajo "Direcciones E/S" la entrada "MC-Servo" en "Bloque de organización" e "IPP OB Servo" en "Imagen de proceso", si desea que los finales de carrera por hardware se evalúen con el ciclo del regulador de posición del accionamiento.

Entrada del final de carrera por hardware negativo / positivo

Seleccione en los campos la variable PLC de la entrada digital para el final de carrera por hardware negativo y positivo.

Para poder seleccionar una entrada es necesario que en la configuración del dispositivo se haya agregado un módulo de entradas digitales y se haya definido el nombre de variable PLC para la entrada digital.

⚠ PRECAUCIÓN

Al colocar los finales de carrera por hardware deben tenerse en cuenta los tiempos de filtrado de las entradas digitales.

Hay que considerar los tiempos de deceleración que resultan del tiempo para un servociclo y el tiempo de filtrado de las entradas digitales.

El tiempo de filtrado puede ajustarse en la configuración del dispositivo de los diferentes módulos de entradas digitales.

Las entradas digitales están ajustadas de forma estándar con un tiempo de filtrado de 6,4 ms. Si se utilizan como final de carrera por hardware pueden producirse deceleraciones no deseadas. En ese caso, reduzca el tiempo de filtrado para las salidas digitales correspondientes.

El tiempo de filtrado se puede ajustar en la configuración del dispositivo de las entradas digitales, en "Filtro de entradas".

Selección de nivel

Seleccione el nivel de señal del final de carrera por hardware ("Nivel inferior"/"Nivel superior") causante. Para "Nivel inferior" la señal de entrada es FALSE cuando se alcanza o supera el final de carrera por hardware. Para "Nivel superior" la señal de entrada es TRUE cuando se alcanza o supera el final de carrera por hardware.

Activar final de carrera por software

Esta casilla de verificación activa la función de los finales de carrera por software inferior y superior. Con los finales de carrera por software activados, cualquier movimiento en curso se detiene en la posición del final de carrera por software. El objeto tecnológico notifica un error. Una vez acusado el error es posible volver a mover el eje en la dirección de la zona de trabajo.

Nota

Los finales de carrera por software activados solo son efectivos con el eje referenciado.

Final de carrera por software inferior/superior

Configure la zona de trabajo del eje con las posiciones del final de carrera por software inferior y superior.

Configuración - límites de dinámica

En la ventana de configuración "Límites dinámicos" configure los valores máximos de velocidad, aceleración, deceleración y tirón del eje.

Velocidad máxima

Defina en este campo la velocidad máxima permitida del eje.

Aceleración máxima / deceleración máxima - tiempo de aceleración / tiempo de deceleración

Ajuste la aceleración deseada en los campos "Tiempo de aceleración" o "Aceleración". La deceleración deseada puede ajustarse en los campos "Tiempo de deceleración" o "Deceleración".

La relación entre el tiempo de aceleración y la aceleración o entre el tiempo de deceleración y la deceleración se explica en las siguientes ecuaciones:

$$\text{Tiempo de posicionamiento} = \frac{\text{Velocidad máxima}}{\text{Aceleración}}$$

$$\text{Tiempo de deceleración} = \frac{\text{Velocidad máxima}}{\text{Deceleración}}$$

Nota

Si se modifica la velocidad máxima se influye en los valores de aceleración y deceleración del eje. Los tiempos de aceleración y deceleración se mantienen.

La "deceleración máxima" para el referenciado activo con inversión de sentido en el final de carrera por hardware debe seleccionarse lo suficientemente grande como para que el eje pueda frenar antes de llegar al tope mecánico.

Tiempo de redondeo mínimo / tirón máximo

Los parámetros para la limitación de tirones se introducen en el campo "Tiempo de redondeo mínimo" o "Tirón máximo":

- Ajuste el tirón deseado para la rampa de aceleración y deceleración en el campo "Tirón máximo". El valor 0 significa que el tirón no está limitado.
- Ajuste el tiempo de redondeo deseado para la rampa de aceleración en el campo "Tiempo de redondeo mínimo".

Nota

El tiempo de redondeo ajustado y visualizado en la configuración solo es válido para la rampa de aceleración.

En caso de que los valores de aceleración y deceleración sean distintos, el tiempo de redondeo de la rampa de deceleración se calcula y utiliza de acuerdo con el tirón de la rampa de aceleración.

El tiempo de redondeo de la deceleración se adapta del siguiente modo:

- **aceleración > deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo menor que en la rampa de aceleración.
- **aceleración < deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo mayor que en la rampa de aceleración.
- **aceleración = deceleración**
  Los tiempos de redondeo son iguales en la rampa de aceleración y deceleración.

En caso de error, el eje reduce la velocidad con la deceleración de parada de emergencia configurada. En este caso no se tiene en cuenta una limitación de tirones configurada.

La relación entre los tiempos de redondeo y el tirón se explica en las ecuaciones siguientes:

$$\text{Tiempo de redondeo (rampa de aceleración)} = \frac{\text{Aceleración}}{\text{Tirón}}$$

$$\text{Tiempo de redondeo (rampa de deceleración)} = \frac{\text{Deceleración}}{\text{Tirón}}$$

Las peticiones de desplazamiento lanzadas en el programa de usuario se ejecutan con el tirón seleccionado.

Configuración - preajustes de dinámica

En la ventana de configuración "Preajustes de dinámica" configure los valores predeterminados de velocidad, aceleración, deceleración y tirón del eje.

Los valores predeterminados son efectivos cuando se han indicado valores < 0 en las instrucciones de Motion Control para los parámetros "Velocity", "Acceleration", "Deceleration" o "Jerk". Los valores predeterminados pueden aplicarse individualmente a cada parámetro citado.

Los valores predeterminados de aceleración y deceleración son efectivos también en los movimientos de desplazamiento del referenciado activo.

Velocidad

Defina en este campo el valor predeterminado para la velocidad del eje.

Aceleración / deceleración - tiempo de aceleración / tiempo de deceleración

Ajuste el valor predeterminado deseado para la aceleración en los campos "Tiempo de aceleración" o "Aceleración". La deceleración deseada puede ajustarse en los campos "Tiempo de deceleración" o "Deceleración".

La relación entre el tiempo de aceleración y la aceleración o entre el tiempo de deceleración y la deceleración se explica en las siguientes ecuaciones:

$$\text{Tiempo de arranque} = \frac{\text{Velocidad}}{\text{Aceleración}}$$

$$\text{Tiempo de deceleración} = \frac{\text{Velocidad}}{\text{Deceleración}}$$

Nota

Si se modifica la velocidad se influye en los valores de aceleración y deceleración del eje. Los tiempos de aceleración y deceleración se mantienen.

Tiempo de redondeo / tirón

Los parámetros para la limitación de tirones se introducen en el campo "Tiempo de redondeo" o "Tirón":

- Ajuste el tirón deseado para la rampa de aceleración y deceleración en el campo "Tirón". El valor 0 significa que el tirón está desactivado.
- Ajuste el tiempo de redondeo deseado para la rampa de aceleración en el campo "Tiempo de redondeo".

Nota

El tiempo de redondeo ajustado y visualizado en la configuración solo es válido para la rampa de aceleración.

En caso de que los valores de aceleración y deceleración sean distintos, el tiempo de redondeo de la rampa de deceleración se calcula y utiliza de acuerdo con el tirón de la rampa de aceleración.

El tiempo de redondeo de la deceleración se adapta del siguiente modo:

- **aceleración > deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo menor que en la rampa de aceleración.
- **aceleración < deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo mayor que en la rampa de aceleración.
- **aceleración = deceleración**
  Los tiempos de redondeo son iguales en la rampa de aceleración y deceleración.

En caso de error, el eje reduce la velocidad con la deceleración de parada de emergencia configurada. En este caso no se tiene en cuenta una limitación de tirones configurada.

La relación entre los tiempos de redondeo y el tirón se explica en las ecuaciones siguientes:

$$\text{Tiempo de redondeo (rampa de aceleración)} = \frac{\text{Aceleración}}{\text{Tirón}}$$

$$\text{Tiempo de redondeo (rampa de deceleración)} = \frac{\text{Deceleración}}{\text{Tirón}}$$

Las peticiones de desplazamiento lanzadas en el programa de usuario se ejecutan con el tirón seleccionado.

Configuración - parada de emergencia

En la ventana de configuración "Parada de emergencia" configure la deceleración de parada de emergencia del eje. En caso de error y al bloquear el eje con la instrucción "MC_Power" de Motion Control (parámetro de entrada StopMode = 0), el eje se para con la deceleración ajustada.

Deceleración de parada de emergencia

Ajuste el valor de deceleración para parada de emergencia en los campos "Deceleración de parada de emergencia" o "Tiempo de deceleración de parada de emergencia".

La relación entre el tiempo de deceleración de parada de emergencia y la deceleración de parada de emergencia se explica en la siguiente ecuación:

$$\text{Tiempo de deceleración de parada de emergencia} = \frac{\text{Velocidad máxima}}{\text{Deceleración de parada de emergencia}}$$

La configuración de la deceleración de parada de emergencia hace referencia a la velocidad máxima configurada del eje. Si se modifica la velocidad máxima del eje, también cambia el valor de la deceleración de parada de emergencia (el tiempo de deceleración de parada de emergencia no cambia).

Referenciación

Por referenciar se entiende la sincronización del valor de posición de un objeto tecnológico a la posición física real del accionamiento. Solo es posible aproximarse a las posiciones de destino absolutas del eje con un eje referenciado.

En S7-1500 Motion Control, el referenciado del eje se efectúa con la instrucción "MC_Home" de Motion Control. Se distingue entre los siguientes modos de operación de la instrucción de Motion Control:

Modos de operación de la instrucción "MC_Home" de Motion Control

- Referenciado activo
  En el referenciado activo, la instrucción "MC_Home" de Motion Control realiza la marcha configurada al punto de referencia. Los desplazamientos en curso se cancelan. La posición del eje se establece de acuerdo con la configuración al detectar la marca de referencia.
- Referenciado pasivo
  En el referenciado pasivo, la instrucción de Motion Control "MC_Home" no realiza ningún movimiento de referencia. El desplazamiento necesario para ello debe ser realizado por el usuario con otras instrucciones de Motion Control. Los movimientos en curso no se cancelan cuando se inicia el referenciado pasivo. El eje se establece de acuerdo con la configuración al detectar la marca de referencia.
- Referenciado directo absoluto
  La posición del eje se fija sin considerar el sensor del punto de referencia. Los procesos de movimiento en curso no se cancelan. El valor del parámetro de entrada "Position" de la instrucción Motion-Control "MC_Home" se aplica inmediatamente como punto de referencia del eje.
- Referenciado directo relativo
  La posición del eje se fija sin considerar el sensor del punto de referencia. Los procesos de movimiento en curso no se cancelan. Para la posición del eje después del referenciado es válido lo siguiente:
  Nueva posición de eje = posición actual del eje + valor del parámetro "Position" de la instrucción "MC_Home".

Referenciado activo

Configuración: referenciar activo

En la ventana de configuración "referenciar activo" se configuran los parámetros para el referenciado activo. El referenciado activo se ejecuta con la instrucción "MC_Home" de Motion Control Mode = 4 y 5.

Selección del modo de referenciado

Elija entre los modos de referenciado siguientes:

Marca cero a través de telegrama PROFIdrive y detector de proximidad (Página 10724)

Marca cero mediante telegrama PROFIdrive (Página 10725)

Marca de referencia a través de entrada digital (Página 10726)

Marca cero a través de telegrama PROFIdrive y detector de proximidad

Activar la inversión de sentido en el final de carrera por hardware

Active esta casilla de verificación si desea utilizar los finales de carrera por hardware como levas de inversión para la marcha al punto de referencia. Si se llega al final de carrera por hardware durante el referenciado activo, el eje frena con la deceleración máxima configurada y realiza una inversión de sentido. Seguidamente, el detector de proximidad se busca en sentido contrario. Si esta función no está activada y el eje alcanza el final de carrera por hardware durante el referenciado activo, el accionamiento se bloquea y se frena con la rampa configurada en el accionamiento.

Sentido de aproximación

Seleccione el sentido de aproximación para buscar el detector de proximidad.

"Positivo" es el sentido de aproximación en el sentido de los valores de posición positivos; "Negativo", en el sentido de los valores de posición negativos.

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la marca cero para el referenciado.

Velocidad de aproximación

Especifique en este campo la velocidad con la que se buscará el detector de proximidad durante la marcha al punto de referencia. Si se ha ajustado un decalaje del punto de referencia, este se extrae con la misma velocidad.

Velocidad de referenciado

Especifique en este campo la velocidad con la que el eje debe llegar a la marca cero para el referenciado. Para detectar la marca cero debe haberse abandonado el detector de proximidad.

Decalaje del punto de referencia

Si la posición del sensor de la marca cero difiere de la posición del punto de referencia, indique el decalaje correspondiente en este campo. El eje se aproxima a la posición de referencia con la velocidad de aproximación.

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 5.

Marca cero mediante telegrama PROFIdrive

Activar la inversión de sentido en el final de carrera por hardware

Active esta casilla de verificación si desea utilizar los finales de carrera por hardware como levas de inversión para la marcha al punto de referencia. Si se llega al final de carrera por hardware durante el referenciado activo, el eje frena con la deceleración máxima configurada y realiza una inversión de sentido. Seguidamente, la marca cero se busca en sentido contrario. Si esta función no está activada y el eje alcanza el final de carrera por hardware durante el referenciado activo, el accionamiento se bloquea y se frena con la rampa configurada en el accionamiento.

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la próxima marca cero para el referenciado.

"Positivo" es el sentido de referenciado en el sentido de los valores de posición positivos; "Negativo", en el sentido de los valores de posición negativos.

Velocidad de aproximación

En el modo de referenciado "Marca cero a través de telegrama PROFIdrive" se utiliza la velocidad de aproximación para extraer el decalaje del punto de referencia.

Velocidad de referenciado

Especifique en este campo la velocidad con la que el eje debe llegar a la marca cero para el referenciado.

Decalaje del punto de referencia

Si la posición del sensor de la marca cero difiere de la posición del punto de referencia, indique el decalaje correspondiente en este campo. El eje se aproxima a la posición de referencia con la velocidad de aproximación.

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 5.

Marca de referencia a través de entrada digital

Si se utiliza una entrada digital como marca de referencia, la precisión del proceso de referenciado no es tan alta como con el referenciado por hardware a través de marcas cero.

Existe la posibilidad de mejorar la precisión utilizando una velocidad de referenciado reducida y módulos de entrada con soporte por temporizador.

Procure también ajustar tiempos de filtrado cortos en la entrada digital.

Entrada digital

Seleccione en este campo la variable PLC de la entrada digital que debe actuar de marca de referencia (leva de referencia).

Para poder seleccionar una entrada es necesario que en la configuración del dispositivo se haya agregado un módulo de entradas digitales y se haya definido el nombre de variable PLC para la entrada digital.

Activar la inversión de sentido en el final de carrera por hardware

Active esta casilla de verificación si desea utilizar los finales de carrera por hardware como levas de inversión para la marcha al punto de referencia. Si se llega al final de carrera por hardware durante el referenciado activo, el eje frena con la deceleración máxima configurada y realiza una inversión de sentido. Seguidamente, la marca de referencia se busca en sentido contrario. Si esta función no está activada y el eje alcanza el final de carrera por hardware durante el referenciado activo, el accionamiento se bloquea y se frena con la rampa configurada en el accionamiento.

Sentido de aproximación

Seleccione el sentido de aproximación para buscar la marca de referencia.

"Positivo" es el sentido de aproximación en el sentido de los valores de posición positivos; "Negativo", en el sentido de los valores de posición negativos.

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la marca de referencia para el referenciado.

Marca de referencia

Elija la posición de conmutación de la "entrada digital" que debe utilizarse como marca de referencia.

Si se sobrepasa una "entrada digital" se generan dos flancos de conmutación separados espacialmente entre sí. Al seleccionar el lado positivo o negativo se garantiza que la marca de referencia se evalúe siempre en la misma posición mecánica.

El lado positivo es la posición de conmutación con un valor de posición mayor, mientras que el lado negativo es la posición de conmutación con el valor de posición menor.

La selección del lado es independiente del sentido de marcha y también de si provoca un flanco ascendente o descendente.

Velocidad de aproximación

Especifique en este campo la velocidad con la que debe buscarse la "entrada digital" durante la marcha al punto de referencia. Si se ha ajustado un decalaje del punto de referencia, este se extrae con la misma velocidad.

Velocidad de referenciado

Especifique en este campo la velocidad con la que el eje debe llegar al punto de referencia para el referenciado.

Decalaje del punto de referencia

Si la posición del punto de referencia difiere de la posición del punto de referencia, indique el decalaje correspondiente en este campo. El eje se aproxima a la posición de referencia con la velocidad de aproximación.

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 5.

Referenciado pasivo

Configuración - referenciación pasiva

En la ventana de configuración "Referenciado pasivo" (referenciado volante) se configuran los parámetros para el referenciado pasivo. La función "referenciado pasivo" se ejecuta con la instrucción "MC_Home" de Motion Control Mode = 2 y 3.

Selección del modo de referenciado

Elija entre los modos de referenciado siguientes:

Marca cero a través de telegrama PROFIdrive y detector de proximidad (Página 10728)

Marca cero mediante telegrama PROFIdrive (Página 10728)

Marca de referencia a través de entrada digital (Página 10729)

Marca cero a través de telegrama PROFIdrive y detector de proximidad

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la marca cero para el referenciado. Se utiliza la próxima marca cero después de abandonar el detector de proximidad.

Existen las opciones siguientes:

- Positivo
  El eje se mueve en el sentido de los valores de posición más altos.
- Negativo
  El eje se mueve en el sentido de los valores de posición más bajos.
- Actual
  Para el referenciado se utiliza el sentido de marcha efectivo en estos momentos

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 3.

Marca cero mediante telegrama PROFIdrive

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la próxima marca cero para el referenciado. Existen las opciones siguientes:

- Positivo
  El eje se mueve en el sentido de los valores de posición más altos.
- Negativo
  El eje se mueve en el sentido de los valores de posición más bajos.
- Actual
  Para el referenciado se utiliza el sentido de marcha efectivo en estos momentos

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 3.

Marca de referencia a través de entrada digital

Entrada digital

Seleccione en este cuadro de diálogo una entrada digital que debe actuar de marca de referencia (leva de referencia).

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la marca de referencia para el referenciado.

Existen las opciones siguientes:

- Positivo
  El eje se mueve en el sentido de los valores de posición más altos.
- Negativo
  El eje se mueve en el sentido de los valores de posición más bajos.
- Actual
  Para el referenciado se utiliza el sentido de marcha efectivo en estos momentos

Marca de referencia

Elija la posición de conmutación de la "entrada digital" que debe utilizarse como marca de referencia.

Si se sobrepasa una "entrada digital" se generan dos flancos de conmutación separados espacialmente entre sí. Al seleccionar el lado positivo o negativo se garantiza que la marca de referencia se evalúe siempre en la misma posición mecánica.

El lado positivo es la posición de conmutación con un valor de posición mayor, mientras que el lado negativo es la posición de conmutación con el valor de posición menor.

La selección del lado es independiente del sentido de marcha y también de si provoca un flanco ascendente o descendente.

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 3.

Monitorización de posición

Configuración - vigilancia de posicionamiento

En la ventana de configuración "Vigilancia de posicionamiento" configure los criterios para vigilar la posición de destino.

Ventana de posicionamiento:

Configure en este campo el tamaño de la ventana de posicionamiento. Si el eje está dentro de esta ventana, se considera que la posición ha sido "alcanzada".

Tiempo de tolerancia de posicionamiento:

Configure en este campo el tiempo de tolerancia dentro del cual el valor de posición debe alcanzar la ventana de posicionamiento.

Tiempo mínimo de permanencia en la ventana de posicionamiento:

Configure en este campo el tiempo mínimo de permanencia. El valor de posición actual debe estar en la ventana de posicionamiento como mínimo durante el "tiempo de permanencia mínimo".

Si se viola uno de los criterios, el eje se detiene y se muestra una alarma de posicionamiento.

Configuración - error de seguimiento

En la ventana de configuración "Error de seguimiento" configure la divergencia admisible entre la posición real del eje y la posición de consigna. El error de seguimiento puede adaptarse dinámicamente a la velocidad actual del eje.

Activar vigilancia de errores de seguimiento

Active esta casilla de verificación si desea activar la vigilancia de errores de seguimiento. Si la vigilancia de errores de seguimiento está activada, el eje se detiene en el rango de error (naranja); en el rango de advertencia se visualiza una alarma.

Si la vigilancia de errores de seguimiento está desactivada, los límites ajustados no tienen efecto alguno.

Error de seguimiento máximo:

Configure en este campo el error de seguimiento admisible a velocidad máxima.

Nivel de advertencia:

Configure en este campo un valor porcentual del límite actual del error de seguimiento a partir del cual debe emitirse una advertencia.

Ejemplo: el error de seguimiento máximo actual es de 100 mm; el nivel de advertencia está configurado al 90%. Si el error de seguimiento actual supera un valor de 90 mm, se emite una advertencia.

Error de seguimiento:

Configure en este campo el error de seguimiento admisible para velocidades pequeñas (sin adaptación dinámica).

Inicio de la adaptación dinámica:

Configure en este campo la velocidad a partir de la cual debe adaptarse dinámicamente el error de seguimiento. A partir de dicha velocidad, el error de seguimiento se adapta al error de seguimiento máximo hasta la velocidad máxima.

Configuración - señal de velocidad cero

En la ventana de configuración "Señal de velocidad cero" configure los criterios para detectar la velocidad cero.

Ventana de velocidad cero:

Configure en este campo el tamaño de la ventana de velocidad cero. Para visualizar la velocidad cero es necesario que la velocidad del eje se mueva dentro de esta ventana.

Tiempo mínimo de permanencia en la ventana de velocidad cero:

Configure en este campo el tiempo mínimo de permanencia en la ventana de velocidad cero. La velocidad del eje debe estar dentro de la ventana de velocidad cero al menos durante el tiempo indicado.

Si se cumplen ambos criterios se visualiza la velocidad cero del eje.

Configuración - lazo de regulación

En la ventana de configuración "Lazo de regulación" configure el control anticipativo y la ganancia Kv del lazo de regulación de posición.

El factor Kv repercute en los siguientes valores característicos:

- Precisión de posicionamiento y regulación de parada
- Uniformidad del movimiento
- Tiempo de posicionamiento

Cuanto mejores sean los requisitos constructivos del eje (gran rigidez) más alto puede ajustarse el factor Kv. Con ello se reduce el error de seguimiento y se consigue una mayor dinámica.

Encontrará información básica en el capítulo Auto-Hotspot.

Control anticipativo

Configure en este campo el control anticipativo de velocidad porcentual para el lazo de regulación de posición.

Ganancia

Introduzca en el campo de entrada la ganancia Kv del lazo de regulación de posición.

Dynamic Servo Control (DSC):

En ejes con regulación de posición (ejes de posicionamiento), la regulación de posición puede realizarse bien en la CPU bien en el accionamiento, siempre que este soporte el procedimiento de regulación Dynamic Servo Control (DSC). Elija el procedimiento de regulación que prefiera:

- **Regulación de posición y velocidad de giro en el accionamiento (DSC activado)**
- **Regulación de posición y velocidad de giro en el PLC**

Nota

El Dynamic Servo Control (DSC) solo es posible con trama estándar 5.

Consulte también

Función y estructura de la optimización (Página 10779)

Configurar el objeto tecnológico Eje de velocidad

Configuración - parámetros básicos

Configure las propiedades básicas del objeto tecnológico en la ventana de configuración "Parámetros básicos".

Nombre del eje

Defina en este campo el nombre del eje de velocidad de giro. El objeto tecnológico aparece con ese nombre en el árbol del proyecto. Las variables del eje de velocidad de giro pueden utilizarse en el programa de usuario con este nombre.

Unidad de usuario

En la lista desplegable, seleccione el sistema de medida deseado para la velocidad de giro.

Interfaz de hardware

Configuración - accionamiento

En la ventana de configuración "Accionamiento", configure el tipo y el accionamiento que desea utilizar.

Tipo de accionamiento

Seleccione en la lista desplegable si desea utilizar un accionamiento PROFIdrive o un accionamiento con conexión analógica.

Los accionamientos PROFIdrive se conectan al controlador por medio de un sistema de comunicación digital (PROFINET o PROFIBUS). La comunicación se realiza utilizando telegramas PROFIdrive.

Los accionamientos con conexión analógica obtienen la consigna de velocidad de giro a través de una señal de salida analógica (p. ej. de -10 V a +10 V) del PLC.

Accionamiento (tipo de accionamiento: PROFIdrive)

Seleccione un accionamiento PROFIdrive / slot previamente configurado en el campo "Accionamiento". En caso de seleccionar un accionamiento PROFIdrive, puede configurarse mediante el botón "Configuración del dispositivo".

Cambie a la configuración del dispositivo y agregue un accionamiento PROFIdrive en la vista de redes si no hubiera ninguno disponible.

Salida (tipo de accionamiento: conexión analógica del accionamiento)

En el campo "Salida", seleccione la variable PLC de la salida analógica con la que debe controlarse el accionamiento. En caso de seleccionar una salida, puede configurarse mediante el botón "Configuración del dispositivo".

Para poder seleccionar una salida es necesario que en la configuración del dispositivo se haya agregado un módulo de salidas analógicas y se haya definido el nombre de variable PLC para la salida analógica.

Activar salida de habilitación (tipo de accionamiento: conexión analógica del accionamiento)

En el campo "Salida de habilitación" seleccione la variable PLC de la salida digital para habilitar el accionamiento. La salida de habilitación habilita o bloquea el regulador de velocidad de giro del accionamiento.

Para poder seleccionar una salida de habilitación es necesario que en la configuración del dispositivo se haya agregado un módulo de salidas digitales y se haya definido el nombre de variable PLC para la salida digital.

Nota

Si no se utiliza ninguna salida de habilitación, el sistema no puede bloquear el accionamiento directamente como consecuencia de reacciones a errores o funciones de vigilancia. No se garantiza una parada controlada del accionamiento.

Activar entrada de disponibilidad (tipo de accionamiento: conexión analógica del accionamiento)

En el campo "Entrada de disponibilidad", seleccione la variable PLC de la entrada digital a través de la cual el accionamiento notifica su disponibilidad al objeto tecnológico. La etapa de potencia está activada y la entrada analógica de la consigna de velocidad de giro está activa.

Para poder seleccionar una entrada de disponibilidad es necesario que en la configuración del dispositivo se haya agregado un módulo de entradas digitales y se haya definido el nombre de variable PLC para la entrada digital.

Nota

La salida de habilitación y la entrada de disponibilidad pueden activarse de forma independiente entre sí.

Para la entrada de disponibilidad activada rigen las condiciones marco siguientes:

- El eje no se habilita (MC_Power Status=TRUE) hasta que hay una señal en la entrada de disponibilidad.
- Si en un eje habilitado desaparece la señal de la entrada de disponibilidad, el eje se bloquea con un error.
- Si el eje se bloquea mediante la instrucción MC_Power (Enable= FALSE), se bloquea también con una señal pendiente en la entrada de disponibilidad.

Consulte también

Configuración - Intercambio de datos (Página 10734)

Configuración - Intercambio de datos

Transferencia de datos hacia el accionamiento

Configure en esta área la transferencia de datos hacia el accionamiento.

Tipo de telegrama (tipo de accionamiento PROFIdrive)

Seleccione el telegrama hacia el accionamiento en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.

Velocidad de giro de referencia (tipo de accionamiento PROFIdrive)

Configure en este campo la velocidad de giro de referencia del accionamiento de acuerdo con las indicaciones del fabricante. La velocidad de giro del accionamiento se especifica porcentualmente a la velocidad de giro de referencia en un rango de -200% a 200%.

Velocidad de giro de referencia (tipo de accionamiento Conexión analógica del accionamiento)

La velocidad de giro de referencia del accionamiento es la velocidad con la que gira el accionamiento en la salida analógica con una salida del 100%. La velocidad de giro de referencia debe configurarse en el accionamiento y aplicarse en la configuración del objeto tecnológico.

El valor analógico emitido con el 100% depende del tipo de salida analógica. Así, por ejemplo, en una salida analógica de +/- 10 V al 100% se emite el valor 10 V.

Las salidas analógicas pueden saturarse en un 17% aproximadamente. Esto significa que una salida analógica puede utilizarse en un rango de -117% a 117%, siempre que el accionamiento lo permita.

Velocidad de giro máxima

Introduzca en este campo la velocidad de giro máxima del accionamiento.

Invertir sentido

Active esta casilla de verificación si debe invertirse el sentido de giro del accionamiento.

Parámetros avanzados

Configuración - Mecánica

Configure la conexión de la carga al accionamiento en la ventana de configuración "Mecánica".

Utilizar parámetros de reductor para cálculo

Active esta casilla de verificación si desea incluir un reductor en el cálculo de la velocidad de giro de la carga.

Número de vueltas del motor / Número de vueltas de la carga

El índice de reducción del reductor de carga se indica como relación entre vueltas del motor y de la carga. Indique aquí un número entero para las vueltas del motor y el número resultante de vueltas de la carga.

Configuración - límites de dinámica

En la ventana de configuración "Límites dinámicos" configure los valores máximos de velocidad de giro, aceleración, deceleración y tirón del eje.

Velocidad de giro máxima

Defina en este campo la velocidad de giro máxima permitida del eje.

Aceleración máxima / deceleración máxima - tiempo de aceleración / tiempo de deceleración

Ajuste la aceleración deseada en los campos "Tiempo de aceleración" o "Aceleración". La deceleración deseada puede ajustarse en los campos "Tiempo de deceleración" o "Deceleración".

La relación entre el tiempo de aceleración y la aceleración o entre el tiempo de deceleración y la deceleración se explica en las siguientes ecuaciones:

$$\text{Tiempo de posicionamiento} = \frac{\text{Velocidad máxima}}{\text{Aceleración}}$$

$$\text{Tiempo de deceleración} = \frac{\text{Velocidad de giro máxima}}{\text{Deceleración}}$$

Nota

Si se modifica la velocidad de giro máxima se influye en los valores de aceleración y deceleración del eje. Los tiempos de aceleración y deceleración se mantienen.

Tiempo de redondeo mínimo / tirón máximo

Los parámetros para la limitación de tirones se introducen en el campo "Tiempo de redondeo mínimo" o "Tirón máximo":

- Ajuste el tirón deseado para la rampa de aceleración y deceleración en el campo "Tirón máximo". El valor 0 significa que el tirón no está limitado.
- Ajuste el tiempo de redondeo deseado para la rampa de aceleración en el campo "Tiempo de redondeo mínimo".

Nota

El tiempo de redondeo ajustado y visualizado en la configuración solo es válido para la rampa de aceleración.

En caso de que los valores de aceleración y deceleración sean distintos, el tiempo de redondeo de la rampa de deceleración se calcula y utiliza de acuerdo con el tirón de la rampa de aceleración.

El tiempo de redondeo de la deceleración se adapta del siguiente modo:

- **aceleración > deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo menor que en la rampa de aceleración.
- **aceleración < deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo mayor que en la rampa de aceleración.
- **aceleración = deceleración**
  Los tiempos de redondeo son iguales en la rampa de aceleración y deceleración.

En caso de error, el eje reduce la velocidad con la deceleración de parada de emergencia configurada. En este caso no se tiene en cuenta una limitación de tirones configurada.

La relación entre los tiempos de redondeo y el tirón se explica en las ecuaciones siguientes:

$$\text{Tiempo de redondeo (rampa de aceleración)} = \frac{\text{Aceleración}}{\text{Tirón}}$$

$$\text{Tiempo de redondeo (rampa de deceleración)} = \frac{\text{Deceleración}}{\text{Tirón}}$$

Las peticiones de desplazamiento lanzadas en el programa de usuario se ejecutan con el tirón seleccionado.

Configuración - preajustes de dinámica

En la ventana de configuración "Preajustes de dinámica" configure los valores predeterminados de velocidad de giro, aceleración, deceleración y tirón del eje.

Los valores predeterminados son efectivos cuando se han indicado valores < 0 en las instrucciones de Motion Control para los parámetros "Velocity", "Acceleration", "Deceleration" o "Jerk". Los valores predeterminados pueden aplicarse individualmente a cada parámetro citado.

Velocidad

Defina en este campo el valor predeterminado para la velocidad de giro del eje.

Aceleración / deceleración - tiempo de aceleración / tiempo de deceleración

Ajuste el valor predeterminado deseado para la aceleración en los campos "Tiempo de aceleración" o "Aceleración". La deceleración deseada puede ajustarse en los campos "Tiempo de deceleración" o "Deceleración".

La relación entre el tiempo de aceleración y la aceleración o entre el tiempo de deceleración y la deceleración se explica en las siguientes ecuaciones:

$$\text{Tiempo de aceleración} = \frac{\text{Velocidad de giro}}{\text{Aceleración}}$$

$$\text{Tiempo de deceleración} = \frac{\text{Velocidad de giro}}{\text{Deceleración}}$$

Nota

Si se modifica la velocidad se influye en los valores de aceleración y deceleración del eje. Los tiempos de aceleración y deceleración se mantienen.

Tiempo de redondeo / tirón

Los parámetros para la limitación de tirones se introducen en el campo "Tiempo de redondeo" o "Tirón":

- Ajuste el tirón deseado para la rampa de aceleración y deceleración en el campo "Tirón". El valor 0 significa que el tirón está desactivado.
- Ajuste el tiempo de redondeo deseado para la rampa de aceleración en el campo "Tiempo de redondeo".

Nota

El tiempo de redondeo ajustado y visualizado en la configuración solo es válido para la rampa de aceleración.

En caso de que los valores de aceleración y deceleración sean distintos, el tiempo de redondeo de la rampa de deceleración se calcula y utiliza de acuerdo con el tirón de la rampa de aceleración.

El tiempo de redondeo de la deceleración se adapta del siguiente modo:

- **aceleración > deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo menor que en la rampa de aceleración.
- **aceleración < deceleración**
  En la rampa de deceleración se utiliza un tiempo de redondeo mayor que en la rampa de aceleración.
- **aceleración = deceleración**
  Los tiempos de redondeo son iguales en la rampa de aceleración y deceleración.

En caso de error, el eje reduce la velocidad con la deceleración de parada de emergencia configurada. En este caso no se tiene en cuenta una limitación de tirones configurada.

La relación entre los tiempos de redondeo y el tirón se explica en las ecuaciones siguientes:

$$\text{Tiempo de redondeo (rampa de aceleración)} = \frac{\text{Aceleración}}{\text{Tirón}}$$

$$\text{Tiempo de redondeo (rampa de deceleración)} = \frac{\text{Deceleración}}{\text{Tirón}}$$

Las peticiones de desplazamiento lanzadas en el programa de usuario se ejecutan con el tirón seleccionado.

Configuración - parada de emergencia

En la ventana de configuración "Rampa de parada de emergencia" configure la deceleración de parada de emergencia del eje. En caso de error y al bloquear el eje con la instrucción "MC_Power" de Motion Control (parámetro de entrada StopMode = 0), el eje se para con la deceleración ajustada.

Deceleración de parada de emergencia

Ajuste el valor de deceleración para parada de emergencia en los campos "Deceleración de parada de emergencia" o "Tiempo de deceleración de parada de emergencia".

La relación entre el tiempo de deceleración de parada de emergencia y la deceleración de parada de emergencia se explica en la siguiente ecuación:

$$\text{Tiempo de deceleración de parada de emergencia} = \frac{\text{Velocidad de giro máxima}}{\text{Deceleración de parada de emergencia}}$$

La configuración de la deceleración de parada de emergencia hace referencia a la velocidad de giro máxima configurada del eje. Si se modifica la velocidad de giro máxima del eje, también cambia el valor de la deceleración de parada de emergencia (el tiempo de deceleración de parada de emergencia no cambia).

Configurar el objeto tecnológico Encoder externo

Configuración - parámetros básicos

Configure las propiedades básicas del objeto tecnológico en la ventana de configuración "Parámetros básicos".

Nombre del encóder externo

Defina en este campo el nombre del encóder externo. El objeto tecnológico aparece con ese nombre en el árbol del proyecto. Las variables del encóder externo pueden utilizarse en el programa de usuario con este nombre.

Tipo de encóder externo

Configure en esta selección si el encóder externo debe captar movimientos lineales o rotativos.

Unidad de usuario

Seleccione en las listas desplegables los sistemas de medida deseados para la posición y velocidad del encóder externo.

Módulo

Active la casilla de verificación "Activar módulo" si desea utilizar un sistema de medida repetitivo para el encóder externo (p. ej. 0-360° con un encóder externo del tipo "rotativo").

- Valor inicial
  En este campo, defina la posición en la que debe empezar el rango del módulo (p. ej. 360° con un encóder externo del tipo "rotativo").
- Longitud
  En este campo, defina la longitud del rango del módulo (p. ej. 360° con un encóder externo del tipo "rotativo").

Interfaz de hardware

Configuración - encoder

El encóder externo adopta la posición de un accionamiento controlado externamente. El encóder necesario para ello transmite la posición del mismo al controlador por medio de un telegrama PROFIdrive. En la ventana de configuración "Encóder", configure el acoplamiento para conectar el encóder.

Seleccionar acoplamiento del encóder:

Seleccione en esta área cómo debe conectarse el encóder de acuerdo con la representación gráfica.

- **Conexión vía módulo tecnológico (TM)**
  Seleccione esta opción si el encóder debe conectarse a un módulo tecnológico (TM).
  - **Módulo tecnológico:**
    En el campo "Módulo tecnológico" seleccione un módulo tecnológico previamente configurado y el canal que debe utilizarse. Para la selección solo se muestran módulos tecnológicos ajustados al modo de operación "Lectura de posición para Motion Control". Si no hubiera ningún módulo tecnológico disponible, cambie a la configuración del dispositivo y agregue uno. Una vez se ha seleccionado un módulo tecnológico se accede a la configuración del mismo mediante el botón "Configuración del dispositivo". El módulo tecnológico puede utilizarse de forma centralizada en un PLC S7-1500 o de forma descentralizada en una periferia descentralizada. Para conectar un encóder incremental utilice el módulo tecnológico TM Count2x24V o TM PosInput2. Para conectar un encóder absoluto utilice el módulo tecnológico TM PosInput2.
- **Conexión vía PROFINET/PROFIBUS (PROFIdrive)**
  Seleccione esta opción si se utiliza un encóder compatible con PROFIdrive.
  - **Selección del encóder:**
    Seleccione un encóder previamente configurado en PROFINET/PROFIBUS en el campo "Selección del encóder". En caso de seleccionar un encóder, puede configurarse mediante el botón "Configuración del dispositivo".
    Cambie a la vista de redes en la configuración del dispositivo y agregue un encóder si no hubiera ninguno disponible.

Configuración - Intercambio de datos

Configuración - Intercambio de datos

Configure la transferencia de datos hacia el encóder en la ventana de configuración "Transferencia de datos". La configuración es diferente según el acoplamiento del encóder:

Encóder con módulo tecnológico (Página 10742)

Encóder con PROFINET/PROFIBUS (Página 10743)

Encóder con módulo tecnológico

Telegrama del encóder

Configure en esta área el telegrama del encóder y los criterios para evaluar los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

Trama

En la lista desplegable del módulo tecnológico seleccione el telegrama que ha configurado en el módulo tecnológico.

Tipo de encóder

Configure en esta área cómo deben evaluarse los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.

Tipo de encóder	Rotativo cíclico absoluto
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (Gn_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (Gn_XIST1).
Bits para factor de multiplicación del valor absoluto de la resolución fina (GN_XIST2)	Configure en este campo el número de bits reservados para el factor de multiplicación del valor absoluto de la resolución fina (Gn_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Consulte también

Configuración - Intercambio de datos (Página 10738)

Encóder con PROFINET/PROFIBUS (Página 10743)

Encóder con PROFINET/PROFIBUS

Telegrama del encóder

Configure en esta área el telegrama del encóder y los criterios para evaluar los datos del encóder. Los datos deben coincidir con los de la configuración del dispositivo.

Transferencia de datos

Seleccione el telegrama del encóder en la lista desplegable. Los datos deben coincidir con el ajuste de la configuración del dispositivo.

Tipo de encóder

Configure los parámetros descritos a continuación según sea el tipo de encóder seleccionado:

Tipo de encóder	Rotativo incremental
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Bits para resolución fina en valor real incremental (GN_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (GN_XIST1).

Tipo de encóder	Rotativo absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (GN_XIST1)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real incremental (GN_XIST1).
Bits para resolución fina en valor real absoluto (GN_XIST2)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real absoluto (GN_XIST2).

Tipo de encóder	Rotativo cíclico absoluto
Incrementos por vuelta:	Configure en este campo el número de incrementos que desata el encóder por vuelta.
Número de vueltas:	Configure en este campo el número de vueltas que puede registrar el encóder absoluto.
Bits para resolución fina en valor real incremental (GN_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (GN_XIST1).
Bits para resolución fina en valor real absoluto (GN_XIST2)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real absoluto (GN_XIST2).

Tipo de encóder	Lineal incremental
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (GN_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (GN_XIST1).

Tipo de encóder	Lineal absoluto
Distancia entre dos incrementos:	Configure en este campo el recorrido entre dos incrementos del encóder.
Bits para resolución fina en valor real incremental (GN_XIST1)	Configure en este campo el número de bits para la resolución fina dentro del valor real incremental (GN_XIST1).
Bits para resolución fina en valor real absoluto (GN_XIST2)	Configure en este campo el número de bits reservados para la resolución fina dentro del valor real absoluto (GN_XIST2).

Invertir sentido

Active esta casilla de verificación si desea invertir el valor real del encóder.

Consulte también

Configuración - Intercambio de datos (Página 10738)

Encóder con módulo tecnológico (Página 10739)

Parámetros avanzados

Configuración - Mecánica

Configuración - Mecánica

En la ventana de configuración "Mecánica" configure los parámetros del encóder para captar la posición del accionamiento controlador externamente.

La configuración es diferente según el tipo de encóder:

Lineal (Página 10746)

Rotativo (Página 10746)

Lineal

Distancia entre dos incrementos

Configure en este campo la distancia entre dos incrementos del encóder incremental o absoluto.

Consulte también

Configuración - Mecánica (Página 10742)

Rotativo (Página 10746)

Rotativo

Utilizar parámetros de reductor para cálculo

Active esta casilla de verificación si desea incluir un reductor en el cálculo de la posición real. Desactive la casilla de verificación si desea configurar el recorrido por vuelta del encóder.

Reductor de medida

- **Número de vueltas del encóder / Número de vueltas de la carga**
  El índice de reducción del reductor de medida se indica como relación entre vueltas del encóder y de la carga. Indique aquí un número entero para las vueltas de la carga y el número resultante de vueltas del encóder.
  Elija los mismos valores para el número de vueltas del motor y de la carga si no hay ningún reductor de carga.

Parámetros de posición

- **Recorrido por vuelta del encóder**
  Configure en este campo el valor por vuelta del encóder que indica el sistema de medida.

Consulte también

Configuración - Mecánica (Página 10742)

Lineal (Página 10743)

Referenciado

Configuración - Referenciación

En la ventana de configuración "Referenciado" se configuran los parámetros para el referenciado del encóder externo. El referenciado se ejecuta con la instrucción "MC_Home" de Motion Control Mode = 2 y 3.

Selección del modo de referenciado

Elija entre los modos de referenciado siguientes:

Marca cero a través de telegrama PROFIdrive y detector de proximidad (Página 10747)

Marca cero mediante telegrama PROFIdrive (Página 10748)

Marca de referencia a través de entrada digital (Página 10748)

Marca cero a través de telegrama PROFIdrive y detector de proximidad

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la marca cero para el referenciado. Se utiliza la próxima marca cero después de abandonar el detector de proximidad.

Existen las opciones siguientes:

- Positivo
  El eje se mueve en el sentido de los valores de posición más altos.
- Negativo
  El eje se mueve en el sentido de los valores de posición más bajos.
- Actual
  Para el referenciado se utiliza el sentido de marcha efectivo en estos momentos

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 3.

Consulte también

Configuración - Referenciación (Página 10744)

Marca cero mediante telegrama PROFIdrive (Página 10748)

Marca de referencia a través de entrada digital (Página 10748)

Marca cero mediante telegrama PROFIdrive

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la próxima marca cero para el referenciado. Existen las opciones siguientes:

- Positivo
  El eje se mueve en el sentido de los valores de posición más altos.
- Negativo
  El eje se mueve en el sentido de los valores de posición más bajos.
- Actual
  Para el referenciado se utiliza el sentido de marcha efectivo en estos momentos

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 3.

Consulte también

Configuración - Referenciación (Página 10744)

Marca cero a través de telegrama PROFIdrive y detector de proximidad (Página 10744)

Marca de referencia a través de entrada digital (Página 10748)

Marca de referencia a través de entrada digital

Entrada digital

Seleccione en este cuadro de diálogo una entrada digital que debe actuar de marca de referencia (leva de referencia).

Sentido de referenciado

Elija el sentido en el que debe realizarse la aproximación a la marca de referencia para el referenciado.

Existen las opciones siguientes:

- Positivo
  El eje se mueve en el sentido de los valores de posición más altos.
- Negativo
  El eje se mueve en el sentido de los valores de posición más bajos.
- Actual
  Para el referenciado se utiliza el sentido de marcha efectivo en estos momentos

Marca de referencia

Elija la posición de conmutación de la "entrada digital" que debe utilizarse como marca de referencia.

Si se sobrepasa una "entrada digital" se generan dos flancos de conmutación separados espacialmente entre sí. Al seleccionar el lado positivo o negativo se garantiza que la marca de referencia se evalúe siempre en la misma posición mecánica.

El lado positivo es la posición de conmutación con un valor de posición mayor, mientras que el lado negativo es la posición de conmutación con el valor de posición menor.

La selección del lado es independiente del sentido de marcha y también de si provoca un flanco ascendente o descendente.

Posición del punto de referencia

Configure en este campo las coordenadas absolutas de la posición del punto de referencia. La posición configurada aquí para el punto de referencia es efectiva cuando se ejecuta la instrucción "MC_Home" de Motion Control con Mode = 3.

Consulte también

Configuración - Referenciación (Página 10744)

Marca cero mediante telegrama PROFIdrive (Página 10745)

Marca cero a través de telegrama PROFIdrive y detector de proximidad (Página 10744)

Configuración - señal de velocidad cero

En la ventana de configuración "Señal de velocidad cero" configure los criterios para detectar la velocidad cero.

Ventana de velocidad cero:

Configure en este campo el tamaño de la ventana de velocidad cero. Para visualizar la velocidad cero es necesario que la velocidad del encóder externo se mueva dentro de esta ventana.

Tiempo mínimo de permanencia en la ventana de velocidad cero:

Configure en este campo el tiempo mínimo de permanencia en la ventana de velocidad cero. La velocidad del encóder externo debe estar dentro de la ventana de velocidad cero al menos durante el tiempo indicado.

Si se cumplen ambos criterios se visualiza la velocidad cero del encóder externo.

Programar

Introducción

El capítulo de programación contiene información general sobre el suministro y la evaluación de las instrucciones de Motion Control, así como sobre el bloque de datos tecnológico.

Encontrará una sinopsis de las instrucciones de Motion Control en el capítulo Funciones (Página 10641).

Las instrucciones de Motion Control permiten enviar peticiones al objeto tecnológico desde el programa de usuario. La petición se define utilizando los parámetros de entrada de la instrucción de Motion Control. El estado actual de la petición se muestra en los parámetros de salida.

Otra interfaz adicional hacia el objeto tecnológico es el bloque de datos tecnológico. Desde el programa de usuario se accede a los datos del bloque de datos tecnológico.

Bloque de datos tecnológico

Introducción

El bloque de datos tecnológico representa el objeto tecnológico y contiene todos los datos de configuración, consignas y valores reales, además de información de estado del objeto tecnológico. El bloque de datos tecnológico se genera automáticamente al crear el objeto tecnológico. Desde el programa de usuario se accede a los datos del bloque de datos tecnológico.

Encontrará una relación y una descripción de las variables en el anexo (Página 10800).

Evaluación del bloque de datos tecnológico

Descripción

El acceso a los datos del bloque de datos tecnológico se realiza de forma análoga al acceso a los bloques de datos estándar.

Lectura de valores desde el bloque de datos tecnológico

En el programa de usuario es posible leer valores reales (p. ej. posición actual) e información de estado o también detectar mensajes de error en el objeto tecnológico. Si se programa una consulta (p. ej. velocidad actual) en el programa de usuario, el valor se lee directamente del objeto tecnológico.

La lectura de valores desde el bloque de datos tecnológico requiere más tiempo que en otros bloques de datos. Si se utilizan reiteradamente variables en un ciclo del programa de usuario, es recomendable copiar los valores de las variables en variables locales y utilizar estas últimas en el programa.

Escritura de valores en el bloque de datos tecnológico

Mediante la configuración del objeto tecnológico en el TIA Portal se escriben los datos correspondientes en el bloque de datos tecnológico. Después de cargarlos en la CPU, dichos datos están guardados en la SIMATIC Memory Card (memoria de carga) de la CPU.

- **Necesidad de cambios en el bloque de datos tecnológico**
  En el programa de usuario puede ser necesario escribir valores en el bloque de datos tecnológico en los casos siguientes, por ejemplo:
  - Adaptación de la configuración del objeto tecnológico (p. ej. límites dinámicos, finales de carrera por software)
  - Uso de correcciones
  - Adaptación de la regulación de posición (p. ej. parámetro "Kv")
- **Efectividad de los cambios en el bloque de datos tecnológico**
  Los cambios realizados en valores del bloque de datos tecnológico desde el programa de usuario pueden hacerse efectivos en instantes distintos. Encontrará la propiedad correspondiente a cada variable en su descripción en el anexo (Página 10800):
  - **Efectivos directamente:**
    Tipo de datos LREAL
    (p. ej. <TO>.Override.Velocity)
    Los cambios de valores son efectivos directamente al escribir las variables. El objeto tecnológico realiza una comprobación de rango del valor escrito y trabaja inmediatamente con el nuevo valor. Si se violan límites de rango al escribir, el objeto tecnológico corrige los valores automáticamente. En caso de rebase por defecto del rango, el valor se pone al límite inferior del rango, en caso de rebase por exceso, al límite superior. Los cambios se escriben mediante asignaciones directas. Los cambios se mantienen hasta el próximo POWER OFF de la CPU o hasta una reinicialización del objeto tecnológico.
    Tipos de datos DINT/BOOL
    (p. ej. <TO>.Senso[n].ActiveHoming.Direction)
    Solo se permite realizar cambios en el rango de valores definido. Los cambios de valores que queden fuera del rango de valores no se aplican. Si se introducen valores no permitidos, se inicia el OB de error de programación (OB 121). Los cambios se escriben mediante asignaciones directas. Los cambios se mantienen hasta el próximo POWER OFF de la CPU o hasta una reinicialización del objeto tecnológico.
  - **Efectivos tras reinicialización:**
    (p. ej. <TO>.Homing.AutoReversal)
    Puesto que las variables relevantes para la reinicialización están ligadas a otras variables, los cambios de valores no pueden aplicarse en cualquier momento. Los cambios solo se aplican tras una reinicialización (restart) del objeto tecnológico. El objeto tecnológico se reinicializa con los datos de la memoria de carga. Por este motivo, los cambios se escriben en el valor inicial de la memoria de carga con la instrucción avanzada "WRIT_DBL" (escribir en bloque de datos de memoria de carga).
    La reinicialización se lanza desde el programa de usuario mediante la instrucción "MC_Reset" de Motion Control con el parámetro "Restart" = TRUE. Encontrará más información sobre la reinicialización en el capítulo Reinicialización de objetos tecnológicos (Página 10771).

Nota

Guardar modificaciones con "WRIT_DBL"

Los cambios en las variables efectivas directamente se pierden con un POWER OFF de la CPU o con la reinicialización del objeto tecnológico.

Si desea guardar los cambios en el bloque de datos tecnológico incluso tras un POWER OFF de la CPU o una reinicialización del objeto tecnológico, es necesario escribir los cambios en el valor inicial de la memoria de carga con la instrucción avanzada "WRIT_DBL".

Nota

Empleo de las funciones de bloques de datos "READ_DBL" y "WRIT_DBL"

Las funciones de los bloques de datos "READ_DBL" y "WRIT_DBL" solo pueden aplicarse en relación con las variables del objeto tecnológico a variables concretas. Las funciones de bloques de datos "READ_DBL" y "WRIT_DBL" no pueden aplicarse a estructuras de datos del objeto tecnológico.

Evaluación de datos en el OB de alarma de sincronismo

Si desea procesar datos del bloque de datos tecnológico de forma isócrona a partir de un ciclo de Motion Control, existe la posibilidad de evaluarlos en el OB de alarma de sincronismo.

Para ello, asigne al OB de alarma de sincronismo el mismo origen del ciclo (Página 10682) que al MC_Servo [OB91]. De este modo, el OB de alarma de sincronismo se llama en cada ciclo de Motion Control.

Lea los datos necesarios en variables globales del OB de alarma de sincronismo. Utilice dichas variables para la posterior programación.

Evaluar StatusWord, ErrorWord y WarningWord

Con el fin de utilizar simbólicamente determinadas informaciones de estado y error de las palabras dobles de datos "StatusWord", "ErrorWord" y "WarningWord", es posible evaluarlas tal como se describe a continuación. Para una evaluación coherente deberían evitarse los direccionamientos de bits a estas palabras dobles de datos en el bloque de datos tecnológico. El acceso a un solo bit del bloque de datos tecnológico dura tanto como el acceso a toda la palabra de datos.

En caso necesario, copie la palabra doble de datos requerida en la variable de una estructura de datos y consulte los diferentes bits de la variable.

Encontrará una descripción de las palabras dobles de datos y la asignación de los diferentes bits en el anexo:

- StatusWord (Página 10815)
- ErrorWord (Página 10817)
- WarningWord (Página 10819)

Requisitos

El objeto tecnológico está creado.

Procedimiento

Para evaluar los diferentes bits en la palabra de datos "StatusWord", proceda del siguiente modo:

1. Cree una estructura de datos global. Ponga un nombre a la estructura de datos, p. ej. "Status".
2. Cree una palabra doble (DWORD) en la estructura de datos "Status". Ponga un nombre a la palabra doble, p. ej. "Temp".
3. Cree 32 variables booleanas en la estructura de datos "Status". Para mantener una visión de conjunto clara, ponga a las diferentes variables booleanas el mismo nombre que a los bits del bloque de datos tecnológico (p. ej. ponga el nombre "HomingDone" a la quinta variable booleana).
4. Dado el caso, copie la variable <TO>.StatusWord del bloque de datos tecnológico en la palabra doble "Temp" de su estructura de datos.
5. Copie los diferentes bits de la palabra doble "Temp" con accesos a bits en las variables booleanas correspondientes.
6. Consulte los diferentes bits de estado por medio de las variables booleanas.

Evalúe las palabras de datos "ErrorWord" y "WarningWord" conforme a los pasos 1 a 6.

Ejemplo

El ejemplo siguiente explica cómo leer y guardar el quinto bit "HomingDone" de la palabra de datos "StatusWord":

SCL

```
Status.Temp        :=    <TO>.StatusWord;                    Copiar palabra
                                                             de estado
Status.HomingDone  :=    Status.Temp.X5;                     Copiar el bit
                                                             concreto para
                                                             acceso por bits
```

AWL

```
L    <TO>.StatusWord                                         Copiar palabra
                                                             de estado
T    Status.Temp
U    Status.Temp.X5                                          Copiar el bit
                                                             concreto para
                                                             acceso por bits
=    Status.HomingDone
```

Cambiar datos relevantes para la reinicialización

Para cambiar datos relevantes para la reinicialización en el bloque de datos tecnológico, escriba en el valor inicial de la variable en la memoria de carga con la instrucción avanzada "WRIT_DBL". Para que se apliquen los cambios es necesario reinicializar el objeto tecnológico.

En la descripción de las variables que figura en el anexo (Página 10800) encontrará información sobre si los cambios de valor de una variable son relevantes para la reinicialización.

Requisitos

El objeto tecnológico está creado.

Procedimiento

Para cambiar datos relevantes para la reinicialización, proceda del siguiente modo:

1. Cree un bloque de datos e introdúzcale los valores relevantes para la reinicialización que desee modificar en el bloque de datos tecnológico. Los tipos de datos deben corresponderse con las variables que deben modificarse.
2. Con la instrucción avanzada "WRIT_DBL", escriba las variables de su bloque de datos en el valor inicial de las variables del bloque de datos tecnológico en la memoria de carga. Si se han modificado datos relevantes para la reinicialización, se mostrará en la variable del objeto tecnológico <TO>.StatusWord.OnlineStartValuesChanged.
3. Reinicialice el objeto tecnológico con la instrucción "MC_Reset" de Motion Control con el parámetro "Restart" = TRUE.

Tras reinicializar el objeto tecnológico, el valor nuevo está aplicado en la memoria de trabajo del bloque de datos tecnológico y es efectivo.

Instrucciones de Motion Control

Parámetros de las instrucciones de Motion Control

Descripción

Al crear el programa de usuario, tenga en cuenta las explicaciones siguientes referentes a los parámetros de las instrucciones de Motion Control.

Referencia al objeto tecnológico

El objeto tecnológico se indica en la instrucción de Motion Control del siguiente modo:

- **Parámetro "Axis"**
  En el parámetro de entrada "Axis" de una instrucción de Motion Control se indica una referencia al objeto tecnológico que debe ejecutar la petición correspondiente.

Tipo de petición y aplicación de los parámetros de entrada de una instrucción de Motion Control

Cuando se inician peticiones y se aplican parámetros modificados se distingue entre las siguientes instrucciones de Motion Control:

- **Instrucciones de Motion Control con el parámetro "Execute"**
  Con un flanco ascendente en el parámetro "Execute" se inicia la petición y se aplican los valores pendientes en los parámetros de entrada.
  Los valores de parámetros modificados con posterioridad no se aplican hasta el inicio de la próxima petición.
  La desactivación del parámetro "Execute" no finaliza la petición pero influye en la duración de visualización del estado de la petición. Mientras "Execute" esté puesto en TRUE se actualizarán los parámetros de salida. Si "Execute" se desactiva antes de concluir una petición, los parámetros "Done", "Error" y "CommandAborted" se activarán solo durante un ciclo de llamada.
- **Instrucciones de Motion Control con el parámetro "Enable"**
  Al activarse el parámetro "Enable" se inicia la petición.
  Mientras "Enable" = TRUE, la petición permanecerá activa y los valores de parámetros modificados se aplicarán directamente.
  Al desactivarse el parámetro "Enable" se finaliza la petición.
  Los parámetros de entrada "JogForward" y "JogBackward" de la instrucción "MC_MoveJog" de Motion Control equivalen en su comportamiento al parámetro "Enable".

Estado de la petición

Los siguientes parámetros de salida indican el estado de procesamiento de la petición:

- **Instrucciones de Motion Control con el parámetro "Done"**
  Si el parámetro "Done" = TRUE, se indica que una petición ha concluido correctamente.
- **Instrucciones de Motion Control sin el parámetro "Done"**
  La consecución del objetivo de la petición se indica con otros parámetros (p. ej. "Status", "InVelocity"). Encontrará más información al respecto en el capítulo Seguimiento de peticiones en curso (Página 10761).
- **Parámetro "Busy"**
  Mientras se procesa una petición, el parámetro "Busy" presenta el valor TRUE. Cuando se finaliza o cancela una petición, "Busy" presenta el valor FALSE.
- **Parámetro "CommandAborted"**
  Cuando una petición cancela otra petición, el parámetro "CommandAborted" presenta el valor TRUE.
- **Parámetro "Error"**
  Cuando se produce un error en la instrucción de Motion Control, el parámetro "Error" presenta el valor TRUE. En el parámetro "ErrorID" se muestra el identificador de error correspondiente.

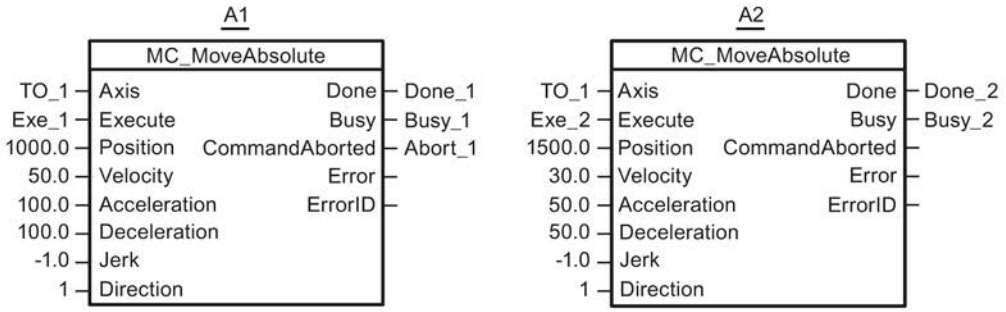
Mientras el parámetro "Execute" o "Enable" esté puesto en TRUE se actualizarán los parámetros de salida. En caso contrario, los parámetros "Done", "Error" y "CommandAborted" se activarán solo durante un ciclo.

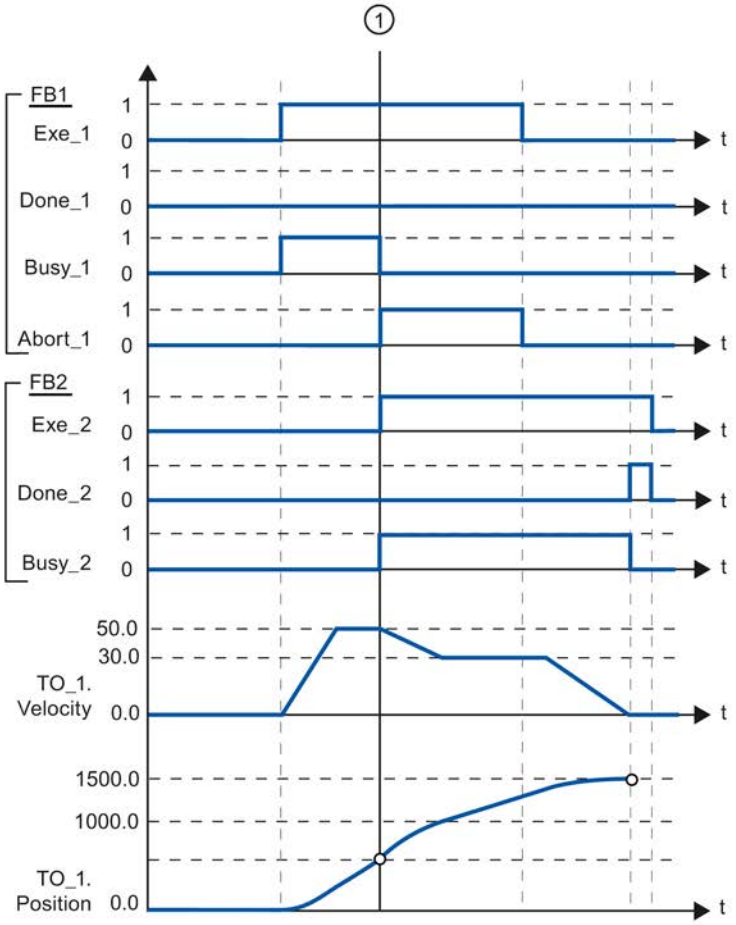
Cancelación de peticiones en curso

Una petición de movimiento en curso se cancela lanzando otra petición de movimiento. En este caso, las consignas dinámicas actuales (aceleración, deceleración, tirón y velocidad) se pondrán a los valores de la petición que realiza el relevo.

Ejemplo de respuesta de los parámetros

La respuesta de los parámetros de instrucciones de Motion Control se representa en el diagrama siguiente en el ejemplo de dos peticiones "MC_MoveAbsolute":





Mediante "Exe_1" se lanza una petición "MC_MoveAbsolute" (A1) con posición de destino 1000.0. "Busy_1" se pone a TRUE. El eje se acelera hasta la velocidad indicada y se mueve hasta la posición de destino (véase TO_1.Velocity y TO_1.Position). Antes de alcanzar la posición de destino, la petición es relevada por otra petición "MC_MoveAbsolute" (A2) en el instante ①. La cancelación se notifica mediante "Abort_1" y "Busy_1" se pone a FALSE. El

eje se frena hasta la velocidad indicada en A2 y se mueve hasta la nueva posición de destino 1500.0. En cuanto se alcanza la posición de destino el sistema lo notifica mediante "Done_2".

Inserción de instrucciones de Motion Control

Las instrucciones de Motion Control se insertan en un bloque de programa igual que otras instrucciones. Las instrucciones de Motion Control permiten controlar todas las funciones disponibles del objeto tecnológico.

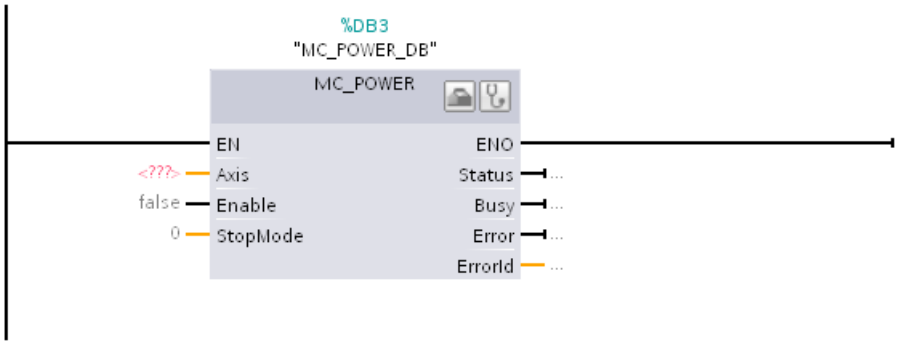
Requisitos

El objeto tecnológico está creado.

Procedimiento

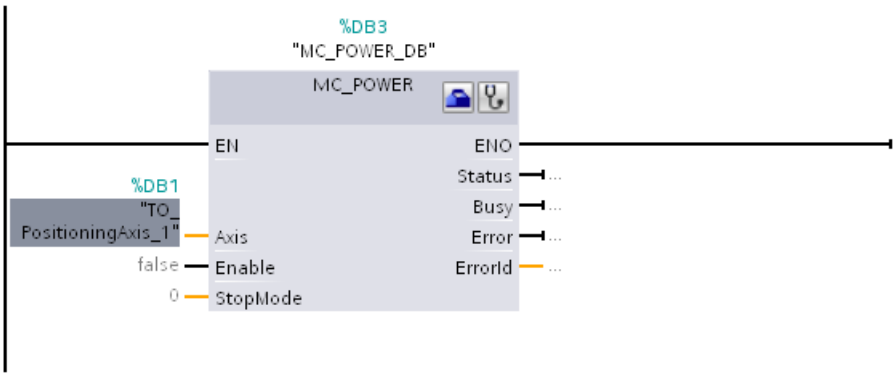
Para insertar las instrucciones de Motion Control en el programa de usuario, proceda del siguiente modo:

1. Haga doble clic en el bloque de programa dentro del árbol del proyecto (el bloque de programa debe llamarse en el programa cíclico).
   El bloque de programa se abre en el editor de programación y se muestran las instrucciones disponibles.
2. Abra la carpeta "Tecnología > Motion Control > S7-1500 Motion Control" en la Task Card "Instrucciones".
3. Utilizando la función de Drag&Drop, arrastre la instrucción de Motion Control, p. ej. "MC_Power", hasta el segmento deseado del bloque de programa.
   Se abre el cuadro de diálogo "Opciones de llamada".
4. Dentro del cuadro de diálogo, asigne un nombre y un número al bloque de datos de instancia de la instrucción de Motion Control.
5. Haga clic en el botón "Aceptar".
   La instrucción "MC_Power" de Motion Control se inserta en el segmento.



El bloque de datos de instancia se crea automáticamente en "Bloques de programa > Bloques de sistema > Recursos de programa".

6. Es necesario suministrar valores a los parámetros de entrada sin valor predeterminado (p. ej. "Axis"). Seleccione el objeto tecnológico en el árbol del proyecto y arrástrelo hasta <...> en el parámetro"Axis" utilizando la función Drag&Drop.



Después de indicar el objeto tecnológico en el parámetro "Axis" estarán disponibles los botones siguientes:
Para abrir la configuración del objeto tecnológico, haga clic en el icono de la caja de herramientas.
Para abrir el diagnóstico del objeto tecnológico, haga clic en el icono del estetoscopio.

7. Inserte otras instrucciones de Motion Control siguiendo los pasos 3 a 6.

Consulte también

Seguimiento de peticiones en curso (Página 10761)

Variables del objeto tecnológico Eje de posicionamiento (Página 10800)

Inicio de peticiones de Motion Control

Descripción

Las peticiones de Motion Control se inician activando el parámetro "Execute" o "Enable" de la instrucción de Motion Control. Las instrucciones de Motion Control deberían llamarse en un nivel de ejecución para un objeto tecnológico.

Al ejecutar peticiones de Motion Control debe considerarse también el estado del objeto tecnológico.

El inicio de las peticiones de Motion Control debe ejecutarse siguiendo los pasos siguientes:

1. Consultar el estado del objeto tecnológico.
2. Lanzar una petición nueva para el objeto tecnológico.
3. Comprobar el estado de la petición.

Estos pasos se explican tomando como ejemplo una petición de posicionamiento absoluto.

1. Consultar el estado del objeto tecnológico

Asegúrese de que el objeto tecnológico se encuentra en el estado adecuado para ejecutar la petición deseada:

- **¿Está habilitado el objeto tecnológico?**
  Para ejecutar peticiones de movimiento, el objeto tecnológico debe estar habilitado.
  La habilitación se lleva a cabo mediante la instrucción "MC_Power" de Motion Control.
  El parámetro "MC_Power.Status" debe presentar el valor TRUE.
- **¿Hay pendiente una alarma tecnológica?**
  Para ejecutar peticiones de movimiento no debe haber alarmas tecnológicas pendientes.
  Tras solucionar el error, acuse las alarmas pendientes, si las hubiera, con la instrucción "MC_Reset" de Motion Control.
  Encontrará una lista de las alarmas tecnológicas y las reacciones a alarma en el anexo Alarmas tecnológicas (Página 10841).
- **¿Está referenciado el objeto tecnológico?**
  Para ejecutar una petición de posicionamiento absoluto, el objeto tecnológico Eje de posicionamiento debe estar referenciado. El referenciado se lleva a cabo mediante la instrucción "MC_Home" de Motion Control. La variable del objeto tecnológico <TO>.StatusWord.HomingDone debe presentar el valor TRUE .

2. Lanzar una petición nueva para el objeto tecnológico

Lance una petición nueva.

En el parámetro "Position" de la instrucción "MC_MoveAbsolute", indique la posición hasta la que debe moverse el eje. Inicie la petición con un flanco ascendente en el parámetro "Execute".

3. Comprobar el estado de la petición

La conclusión correcta de una petición (en este caso, cuando se alcanza la posición de destino), se indica con el parámetro "Done" de la instrucción de Motion Control.

Si se detecta un error, el parámetro "Error" de la instrucción de Motion Control se pone a TRUE y la petición se rechaza.

Existe la posibilidad de programar una rutina de errores para la petición de Motion Control. Para ello, evalúe un error visualizado en el parámetro "Error". La causa del error se indica en el parámetro "ErrorID". Reinicie la petición tras solucionar la causa del error.

Si mientras se procesa la petición "Error" = TRUE y "ErrorID" = 16#8001, significa que se ha producido una alarma tecnológica.

Encontrará una lista de las ErrorIDs en el anexo Identificador de error (Página 10844).

Información adicional

Encontrará una posibilidad de evaluar los diferentes bits de estado, error y advertencia en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Seguimiento de peticiones en curso

Introducción

El estado actual de procesamiento de la petición está disponible en los parámetros de salida de la instrucción de Motion Control. Dichos parámetros se actualizan con cada llamada de la instrucción de Motion Control.

En el seguimiento de peticiones se distinguen tres grupos:

- Instrucciones de Motion Control con el parámetro "Done" (Página 10761)
- Instrucción "MC_MoveVelocity" de Motion Control (Página 10764)
- Instrucción "MC_MoveJog" de Motion Control (Página 10767)

Instrucciones de Motion Control con el parámetro "Done"

Descripción

Las peticiones de instrucciones de Motion Control con el parámetro "Done" se inician con un flanco ascendente en el parámetro "Execute". Si la petición ha concluido sin errores y sin interrupciones provocadas por otra petición (p. ej. "MC_ MoveAbsolute": posición de destino alcanzada), el parámetro "Done" presenta el valor TRUE.

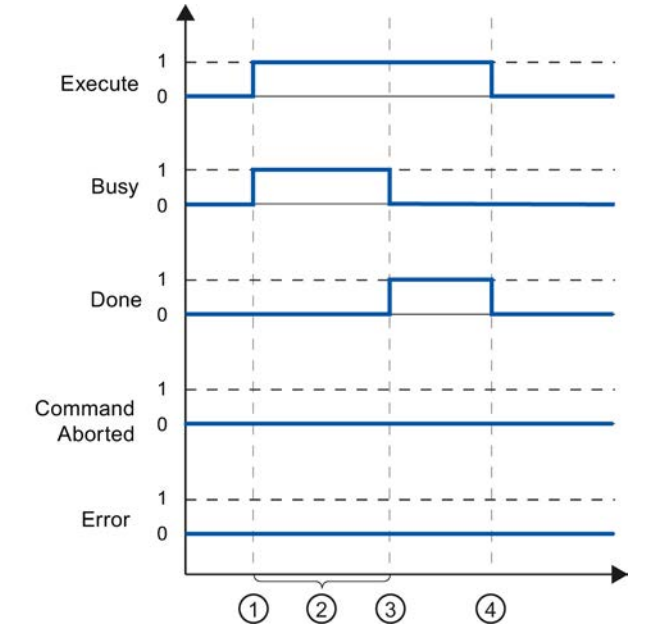
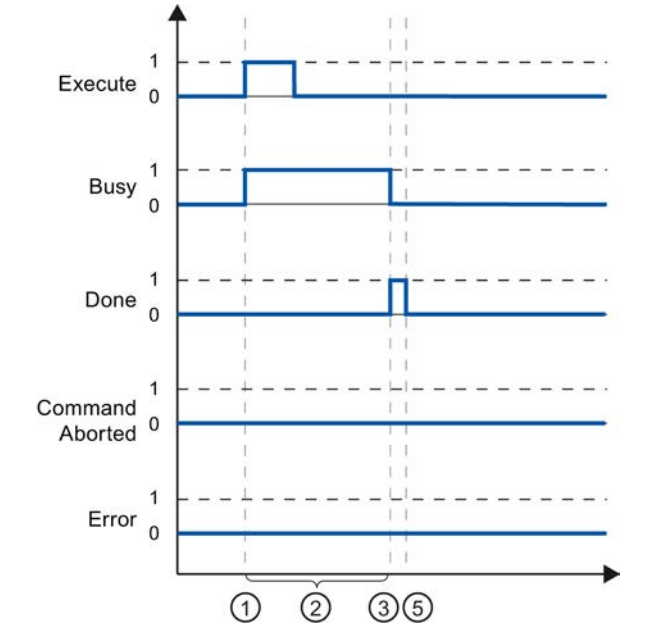
Las siguientes instrucciones de Motion Control tienen un parámetro "Done":

- MC_Home
- MC_MoveRelative
- MC_MoveAbsolute
- MC_Halt
- MC_Reset

A continuación se muestra el comportamiento de los parámetros tomando como ejemplo diversas situaciones:

Procesamiento íntegro de la petición

Si la petición de Motion Control se procesa íntegramente hasta su finalización, ello se indica en el parámetro "Done" con el valor TRUE. El estado lógico del parámetro "Execute" afecta a la duración de la visualización en el parámetro "Done":

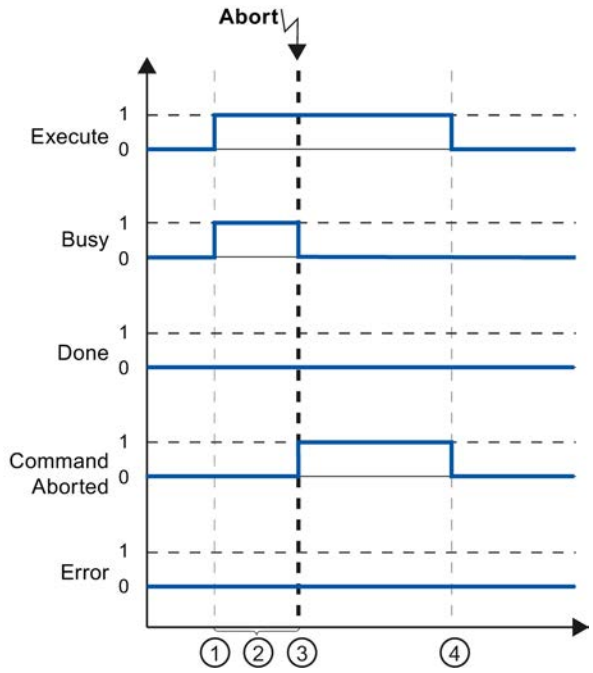
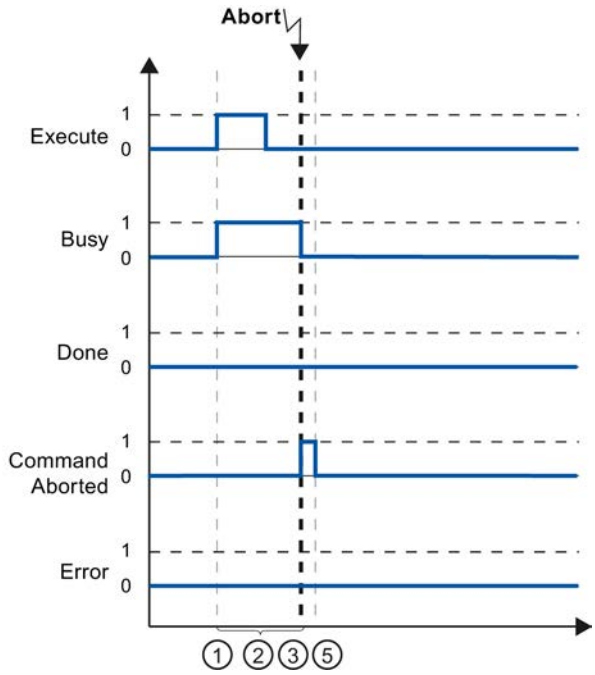
"Execute" se pone a FALSE **tras finalizar** la petición.	"Execute" se pone a FALSE **durante el procesamiento** de la petición.
	

①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, es posible poner "Execute" al valor FALSE durante la petición o mantener el valor TRUE hasta después de concluirse la petición.
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	Una vez finalizada la petición (p. ej. con la instrucción "MC_MoveAbsolute" de Motion Control: posición de destino alcanzada), el parámetro "Busy" cambia a FALSE y el parámetro "Done" a TRUE.
④	Mientras el parámetro "Execute" conserve el valor TRUE una vez finalizada la petición, el parámetro "Done" también permanecerá en el valor TRUE.
⑤	Si el parámetro "Execute" cambia a FALSE antes de que la petición finalice, el parámetro "Done" presenta el valor TRUE únicamente durante un ciclo de procesamiento.

Cancelación de la petición

Si la petición de Motion Control es cancelada por otra petición durante el procesamiento, ello se indica en el parámetro "CommandAborted" con el valor TRUE. El estado lógico del

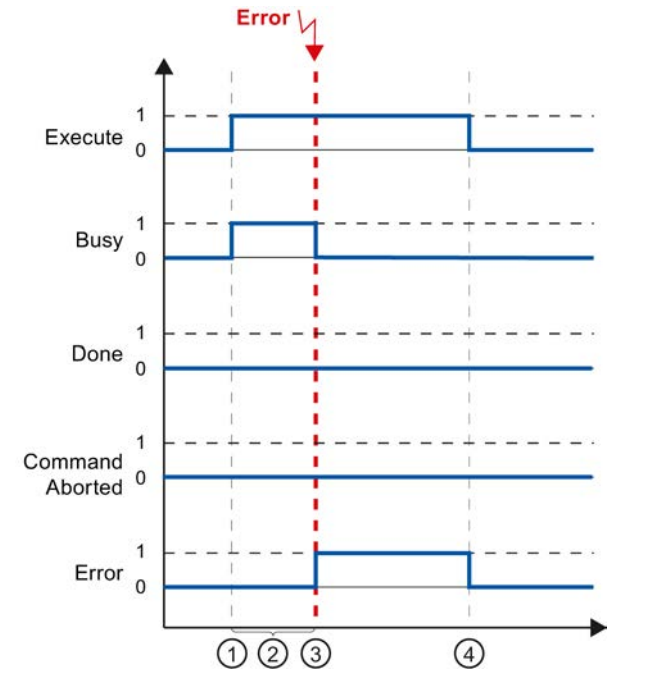
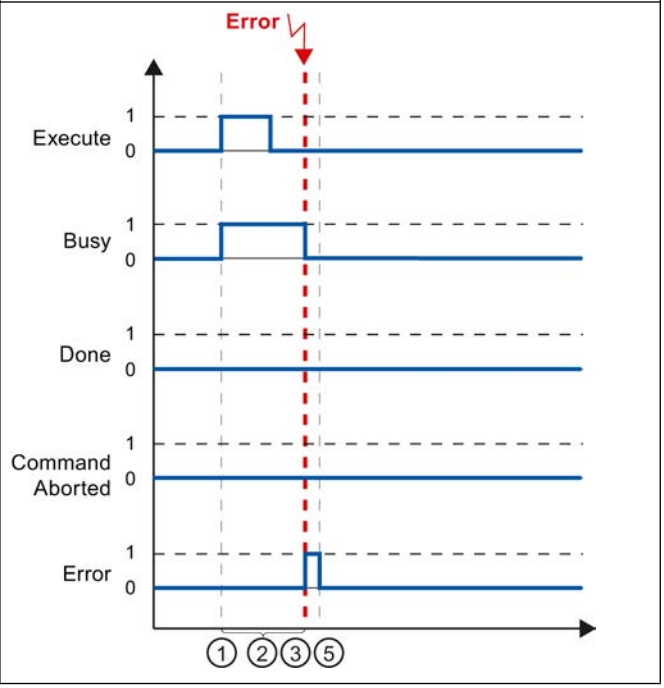
parámetro "Execute" afecta a la duración de la visualización en el parámetro "CommandAborted":

"Execute" se pone a FALSE **tras cancelarse** la petición.	"Execute" se pone a FALSE **antes** de cancelarse la petición.
	

①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, es posible poner "Execute" al valor FALSE durante la petición o mantener el valor TRUE hasta después de concluirse la petición.
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	La petición es cancelada por otra petición de Motion Control durante su procesamiento. Con la cancelación de la petición, el parámetro "Busy" cambia a FALSE y "CommandAborted" a TRUE.
④	Mientras el parámetro "Execute" conserve el valor TRUE una vez finalizada la petición, el parámetro "CommandAborted" también permanecerá en el valor TRUE.
⑤	Si el parámetro "Execute" cambia a FALSE antes de que la petición se cancele, el parámetro "CommandAborted" presenta el valor TRUE únicamente durante un ciclo de procesamiento.

Error durante el procesamiento de la petición

Si se produce un error mientras se procesa la petición de Motion Control, ello se muestra en el parámetro "Error" con el valor TRUE. El estado lógico del parámetro "Execute" afecta a la duración de la visualización en el parámetro "Error":

"Execute" se pone a FALSE **tras** producirse el error.	"Execute" se pone a FALSE **antes** de producirse el error.
	

①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, es posible poner "Execute" al valor FALSE durante la petición o mantener el valor TRUE hasta después de concluirse la petición.
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	Se produce un error al procesar la petición. Al producirse el error, el parámetro "Busy" cambia a FALSE y el parámetro "Error" a TRUE.
④	Mientras el parámetro "Execute" conserve el valor TRUE tras producirse el error, el parámetro "Error" también permanecerá en el valor TRUE.
⑤	Si el parámetro "Execute" cambia a FALSE antes de que se produzca el error, el parámetro "Error" presenta el valor TRUE únicamente durante un ciclo de procesamiento.

Instrucción "MC_MoveVelocity" de Motion Control

Descripción

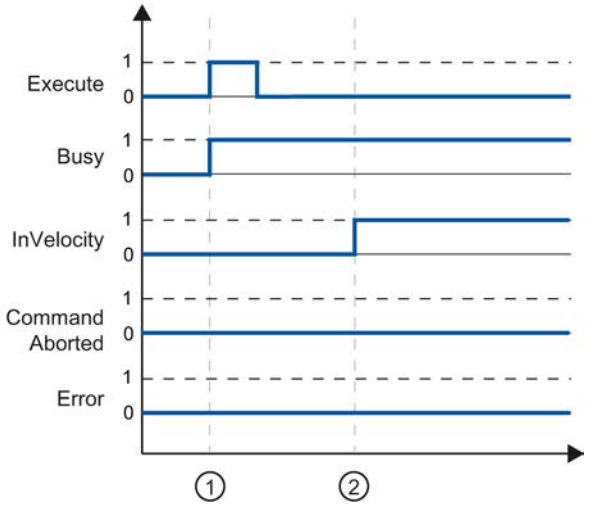
Una petición "MC_MoveVelocity" se inicia con un flanco ascendente en el parámetro "Execute". El objetivo de la petición se cumple en cuanto se alcanza la velocidad parametrizada y el eje se mueve a velocidad constante. En cuanto se alcanza y mantiene la velocidad parametrizada, en el parámetro "InVelocity" se muestra el valor TRUE.

El movimiento del eje se puede detener, p. ej. con una petición "MC_Halt".

A continuación se muestra el comportamiento de los parámetros tomando como ejemplo diversas situaciones:

Se alcanza y mantiene la velocidad parametrizada

En cuanto se alcanza la velocidad parametrizada, en el parámetro "InVelocity" se muestra el valor TRUE. El parámetro "Execute" no influye en la duración de visualización del parámetro "InVelocity".

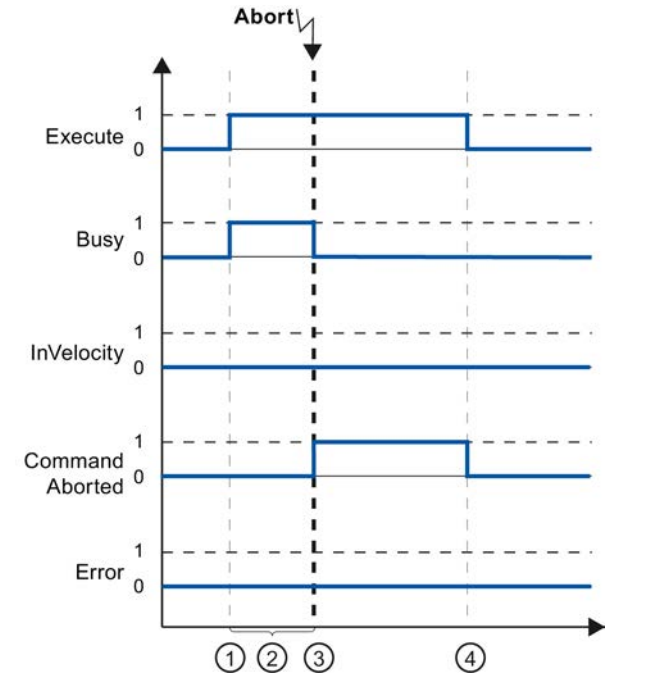
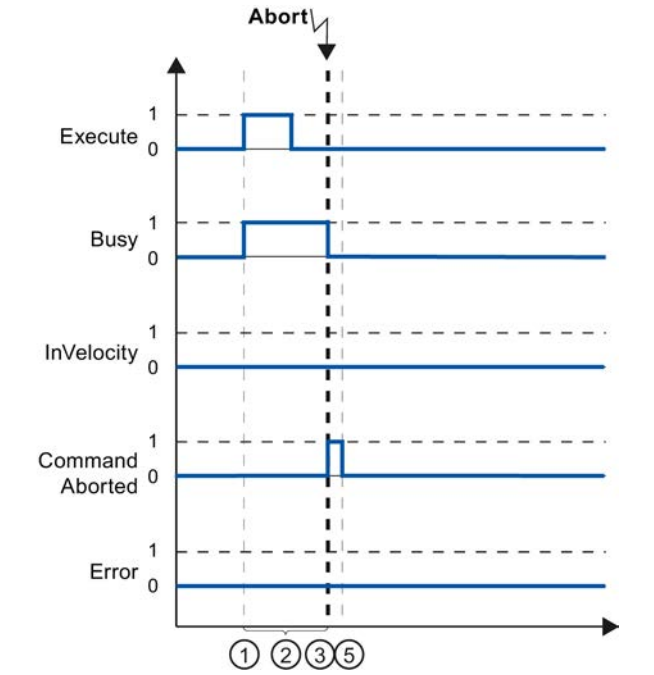


①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, "Execute" se puede volver a poner a FALSE antes o después de alcanzar la velocidad parametrizada. Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
②	Al alcanzar la velocidad parametrizada, el parámetro "InVelocity" cambia a TRUE. Los parámetros "Busy" y "InVelocity" se mantienen en el valor TRUE hasta que la petición "MC_MoveVelocity" es relevada por otra petición de Motion Control.

La petición se cancela antes de alcanzar la velocidad parametrizada

Si la petición de Motion Control es cancelada por otra petición antes de alcanzar la velocidad parametrizada, ello se indica en el parámetro "CommandAborted" con el valor TRUE. El

estado lógico del parámetro "Execute" afecta a la duración de la visualización en el parámetro "CommandAborted".

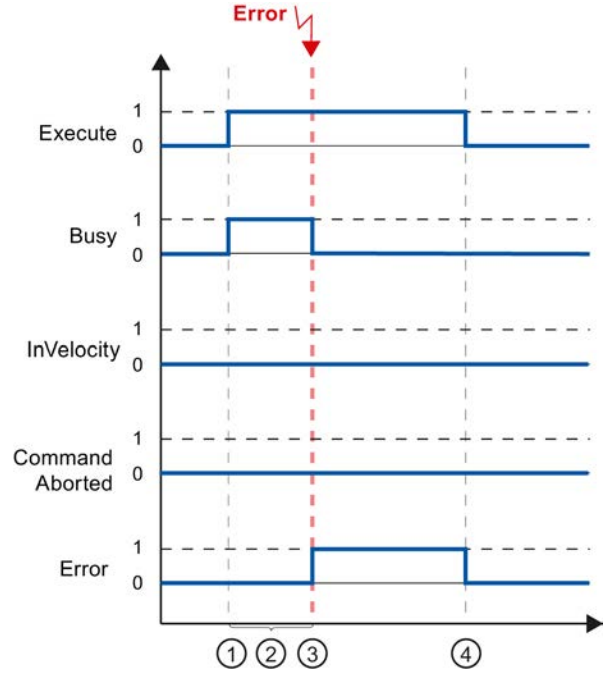
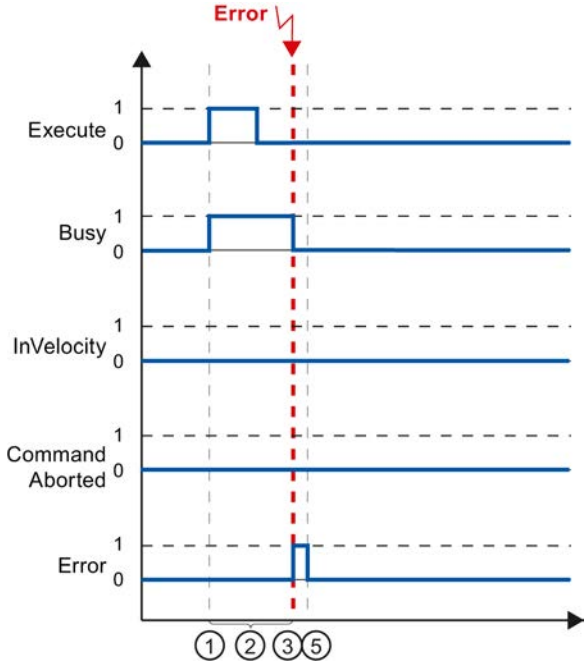
"Execute" se pone a FALSE **tras cancelarse** la petición.	"Execute" se pone a FALSE **antes** de cancelarse la petición.
	

①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, es posible poner "Execute" al valor FALSE durante la petición o mantener el valor TRUE hasta después de concluirse la petición.
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	La petición es cancelada por otra petición de Motion Control durante su procesamiento. Con la cancelación de la petición, el parámetro "Busy" cambia a FALSE y "CommandAborted" a TRUE.
④	Mientras el parámetro "Execute" conserve el valor TRUE una vez finalizada la petición, el parámetro "CommandAborted" también permanecerá en el valor TRUE.
⑤	Si el parámetro "Execute" cambia a FALSE antes de que la petición se cancele, el parámetro "CommandAborted" presenta el valor TRUE únicamente durante un ciclo de procesamiento.

Antes de alcanzarse la velocidad parametrizada se produce un error

Si se produce un error mientras se procesa la petición de Motion Control antes de alcanzarse la velocidad parametrizada, ello se muestra en el parámetro "Error" con el valor TRUE. El

estado lógico del parámetro "Execute" afecta a la duración de la visualización en el parámetro "Error".

"Execute" se pone a FALSE **tras** producirse el error.	"Execute" se pone a FALSE **antes** de producirse el error.
	

①	La petición se inicia con un flanco ascendente en el parámetro "Execute". Según la programación, es posible poner "Execute" al valor FALSE durante la petición o mantener el valor TRUE hasta después de que se produzca el error.
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	Se produce un error al procesar la petición. Al producirse el error, el parámetro "Busy" cambia a FALSE y el parámetro "Error" a TRUE.
④	Mientras el parámetro "Execute" conserve el valor TRUE una vez finalizada la petición, el parámetro "Error" también permanecerá en el valor TRUE.
⑤	Si el parámetro "Execute" cambia a FALSE antes de que la petición se cancele, el parámetro "Error" presenta el valor TRUE únicamente durante un ciclo de procesamiento.

Instrucción "MC_MoveJog" de Motion Control

Descripción

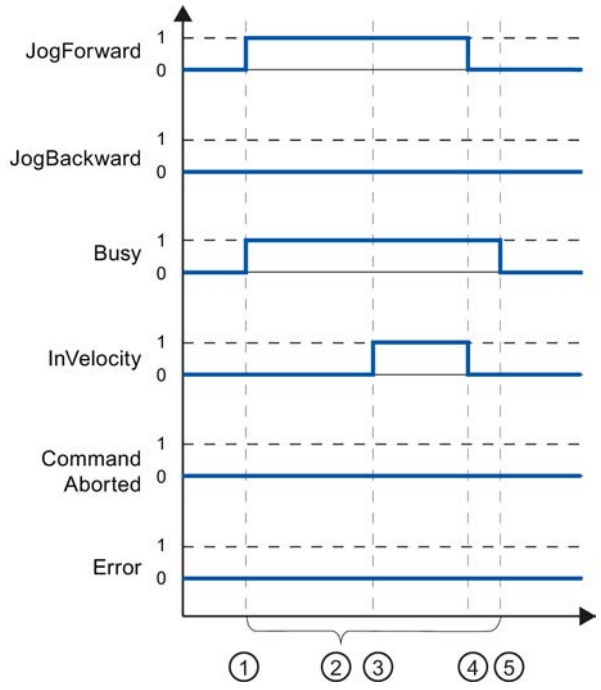
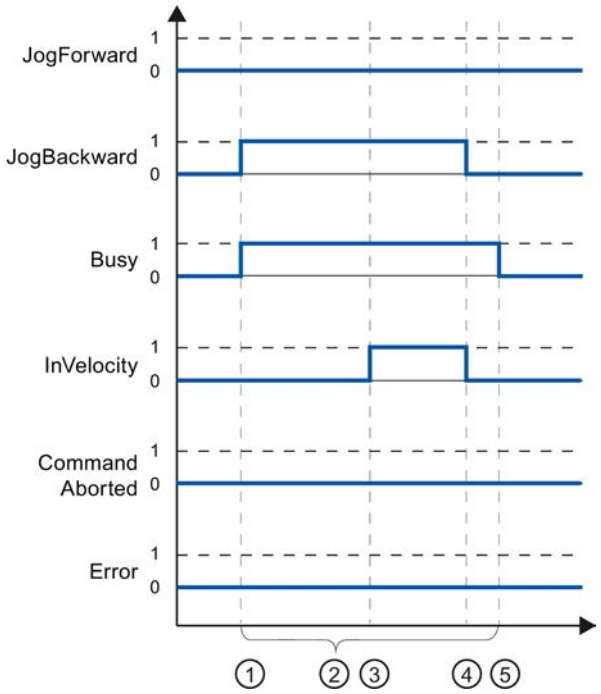
Una petición "MC_MoveJog" se inicia al activarse el parámetro "JogForward" o "JogBackward". El objetivo de la petición se cumple en cuanto se alcanza la velocidad parametrizada y el eje se mueve a velocidad constante. En cuanto se alcanza y mantiene la velocidad parametrizada, en el parámetro "InVelocity" se muestra el valor TRUE.

La petición finaliza en cuanto al parámetro "JogForward" o "JogBackward" se pone al valor FALSE y el eje se para por completo.

A continuación se muestra el comportamiento de los parámetros tomando como ejemplo diversas situaciones:

Se alcanza y mantiene la velocidad parametrizada

Si la petición de Motion Control se ejecuta hasta alcanzar la velocidad parametrizada, ello se indica en el parámetro "InVelocity" con el valor TRUE.

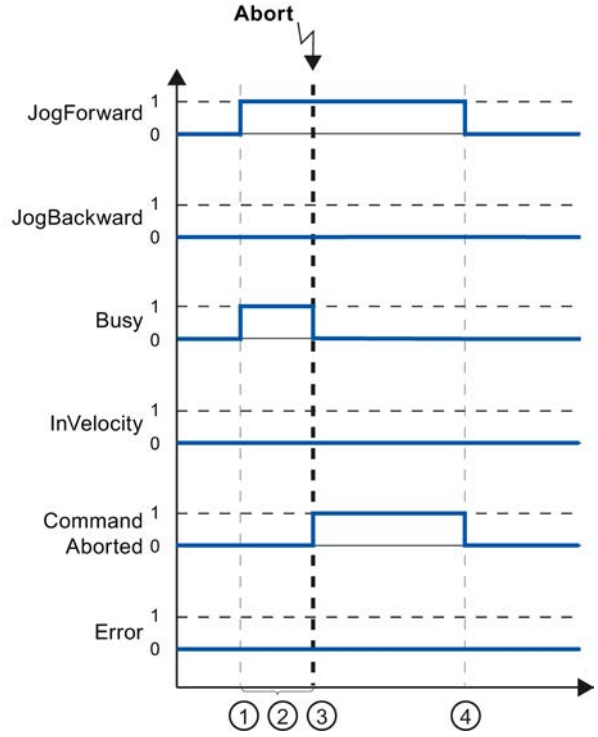
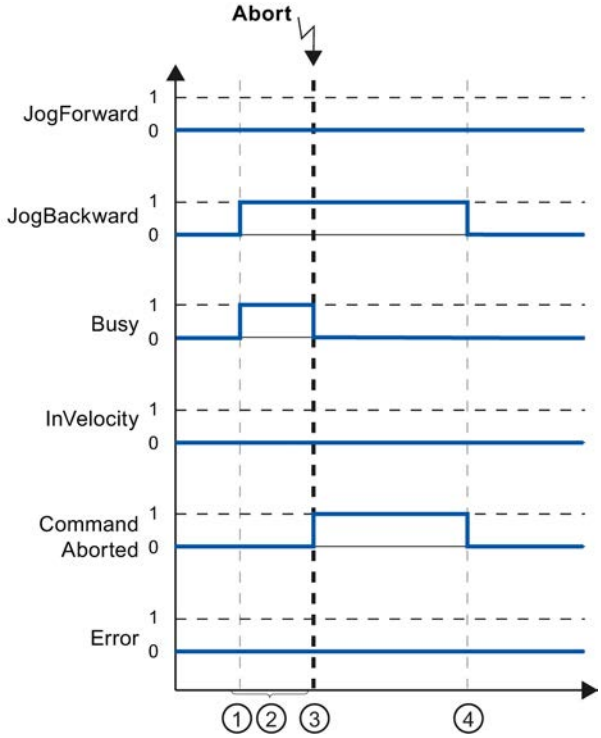
El modo Jog se controla a través del parámetro "JogForward"	El modo Jog se controla a través del parámetro "JogBackward".
	

①	La petición se inicia al activarse el parámetro "JogForward" o "JogBackward".
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	Al alcanzar la velocidad parametrizada, el parámetro "InVelocity" cambia a TRUE.
④	El movimiento del eje finaliza en cuanto se desactiva el parámetro "JogForward" o "JogBackward". El eje se frena. El parámetro "InVelocity" cambia a FALSE.
⑤	En cuanto el eje se para, la petición de Motion Control finaliza y el parámetro "Busy" cambia a FALSE.

La petición se cancela durante el procesamiento

Si la petición de Motion Control es cancelada por otra petición durante el procesamiento, ello se indica en el parámetro "CommandAborted" con el valor TRUE. El comportamiento del

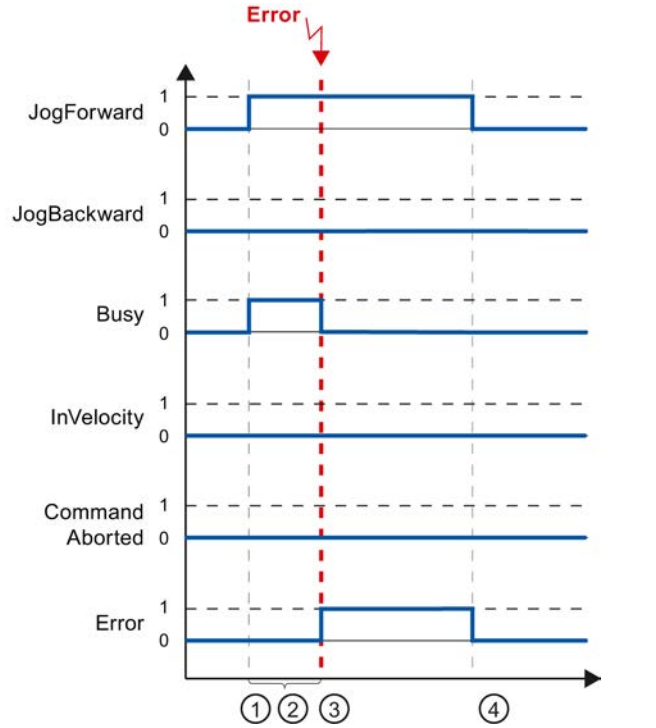
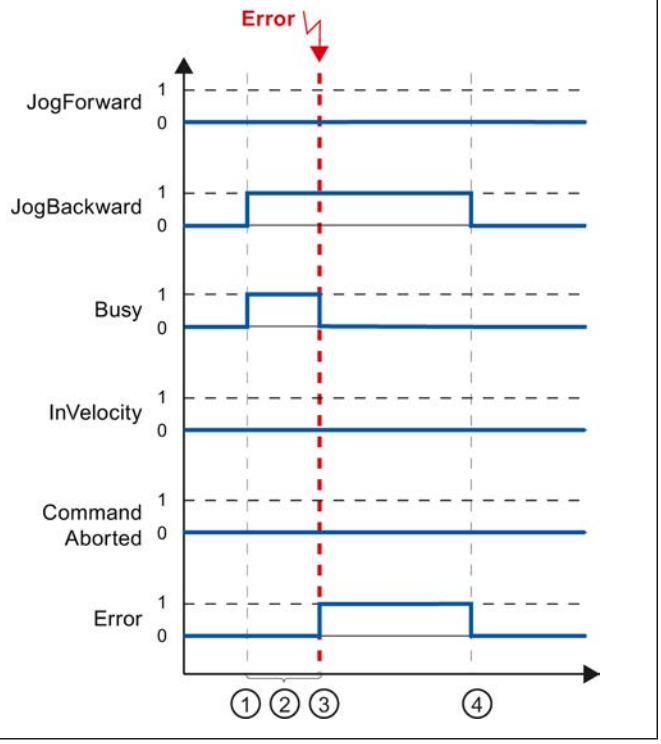
parámetro "CommandAborted" es independiente del hecho de alcanzar o no la velocidad parametrizada.

El modo Jog se controla a través del parámetro "JogForward".	El modo Jog se controla a través del parámetro "JogBackward".
	

①	La petición se inicia al activarse el parámetro "JogForward" o "JogBackward".
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	La petición es cancelada por otra petición de Motion Control durante su procesamiento. Con la cancelación de la petición, el parámetro "Busy" cambia a FALSE y "CommandAborted" a TRUE.
④	Al desactivarse el parámetro "JogForward" o "JogBackward", el parámetro "CommandAborted" cambia también a FALSE.

Se produce un error mientras se procesa la petición

Si se produce un error mientras se procesa la petición de Motion Control, ello se muestra en el parámetro "Error" con el valor TRUE. El comportamiento del parámetro "Error" es independiente del hecho de alcanzar o no la velocidad parametrizada.

El modo Jog se controla a través del parámetro "JogForward".	El modo Jog se controla a través del parámetro "JogBackward".
	

①	La petición se inicia al activarse el parámetro "JogForward" o "JogBackward".
②	Mientras se procesa la petición, el parámetro "Busy" presenta el valor TRUE.
③	Se produce un error al procesar la petición. Con la aparición del error, el parámetro "Busy" cambia a FALSE y "Error" a TRUE.
④	Cuando el parámetro "JogForward" o "JogBackward" se pone al valor FALSE, el parámetro "Error" cambia también a FALSE.

Finalización de peticiones de Motion Control

Al finalizar una petición, se distingue entre la conclusión correcta de la petición y la cancelación de un movimiento.

Conclusión de la petición

La conclusión de una petición de Motion Control se indica tal como se describe en el capítulo Seguimiento de peticiones en curso (Página 10758).

Cancelación de un movimiento

Cuando es necesario cancelar un movimiento pueden tomarse las medidas siguientes:

- Ejecutar "MC_Halt (Página 5115)"
  Para cancelar un movimiento y detener el eje puede utilizarse la instrucción "MC_Halt".
- Desactivar "MC_Power (Página 5088)"
  En caso de emergencia es posible detener el eje mediante una rampa de parada rápida. Ponga para ello el parámetro "Enable" de la instrucción "MC_Power" a FALSE. El eje se frena de acuerdo con el "StopMode" seleccionado y todas las peticiones del objeto tecnológico se cancelan.

Reinicialización de objetos tecnológicos

Descripción

El sistema inicializa automáticamente los objetos tecnológicos después de conectar la CPU y después de cargar los valores iniciales del bloque de datos tecnológico en la CPU. Si al volver a realizar la carga en la CPU se detectan cambios relevantes para la reinicialización, el objeto tecnológico se reinicializa automáticamente.

Si con el programa de usuario se han modificado datos relevantes para la reinicialización en RUN, el usuario debe reinicializar el objeto tecnológico para que se apliquen los cambios.

Si desea guardar los cambios en el bloque de datos tecnológico incluso tras la reinicialización del objeto tecnológico, es necesario escribir los cambios en el valor inicial de la memoria de carga con la instrucción avanzada "WRIT_DBL".

Reinicialización necesaria

Si es necesario reinicializar el objeto tecnológico, se indica en "Objeto tecnológico > Diagnóstico > Bits de estado y de error > Estado eje o Estado encóder > Valor inicial online modificado" y en la variable del objeto tecnológico <TO>.StatusWord.OnlineStartValuesChanged.

Reinicialización de un objeto tecnológico

El usuario reinicializa el objeto tecnológico con la instrucción "MC_Reset" de Motion Control con el parámetro "Restart" = TRUE.

Al realizar la reinicialización, todos los datos de configuración del objeto tecnológico se carga de la memoria de carga a la memoria de trabajo. Al hacerlo, se sobrescriben los valores actuales del bloque de datos tecnológico.

Tenga en cuenta las indicaciones siguientes al reinicializar el objeto tecnológico:

- Con una reinicialización se desactiva el estado "referenciado" de un objeto tecnológico con valores reales incrementales (<TO>.StatusWord.HominDone).
- Mientras se ejecuta una reinicialización, el objeto tecnológico no puede ejecutar peticiones. Una reinicialización activa se indica en "Objeto tecnológico > Diagnóstico > Bits de estado y de error > Estado eje o Estado encóder > Reinicio activo" y en la variable del objeto tecnológico <TO>.StatusWord.RestartActive.

- Durante una reinicialización, las peticiones de Motion Control se rechazan con los parámetros "Error" = TRUE y "ErrorID" = 16#800D (petición no ejecutable porque hay una reinicialización activa).
- Mientras se ejecuta una reinicialización no es posible acceder al bloque de datos tecnológico.

Consulte también

Cambiar datos relevantes para la reinicialización (Página 10751)

Cargar en CPU

Descripción

Con la carga en la CPU S7-1500 se asegura siempre que los datos de proyecto sean coherentes tras la carga tanto online como offline.

Los datos de los objetos tecnológicos se guardan en bloques de datos tecnológicos. Para cargar objetos tecnológicos nuevos o modificados rigen por lo tanto las condiciones para la carga de bloques.

Es posible cargar los siguientes grupos de objetos en la CPU:

Comando del menú contextual "Cargar en dispositivo"	Descripción
Hardware y software	Cargar todos los bloques nuevos y modificados y una configuración hardware nueva o modificada.
Configuración hardware	Cargar una configuración hardware nueva o modificada.
Software (solo cambios)	Cargar todos los bloques nuevos y modificados.
Software (todos los bloques)	Cargar todos los bloques.

Cargar en estado operativo RUN

Al cargar en estado operativo RUN de la CPU se comprueba si es posible cargar sin reinicializar el objeto tecnológico.

Si se han modificado valores de configuración relevantes para la reinicialización, tras la carga en la CPU el objeto tecnológico se reinicializa automáticamente.

Solo es posible cargar un objeto tecnológico si está bloqueado.

Los cambios siguientes no pueden cargarse en la CPU en estado operativo RUN:

- cambios de los ciclos de reloj del MC-Servo
- cambios en la interfaz de hardware del objeto tecnológico, en "Objeto tecnológico > Configuración > Interfaz de hardware"

Puesta en servicio

Introducción

La guía siguiente describe los pasos que deben tenerse en cuenta al poner en servicio las unidades de proceso específicas de Motion Control.

La puesta en servicio de otras unidades del sistema de automatización depende de la configuración de la instalación en cuestión. La puesta en servicio (sin Motion Control) se describe en el manual de sistema Sistema de automatización S7-1500 (http://support.automation.siemens.com/WW/view/es/59191792).

Guía de puesta en servicio

La presente guía sirve de referencia para la puesta en servicio de una instalación con Motion Control. El procedimiento se describe tomando como ejemplo un objeto tecnológico Eje de posicionamiento.

Requisitos

- La configuración de los componentes siguientes se ha terminado:
  - CPU
  - Comunicación BUS
  - Accionamientos
  - Objetos tecnológicos
- El programa de usuario está creado.
- El cableado de la CPU y la periferia correspondiente está terminado.
- La puesta en servicio y optimización del accionamiento está terminada.

Procedimiento

Para poner en servicio las unidades de proceso específicas de Motion Control, proceda del siguiente modo:

Paso	Acción	Soportada por el TIA Portal
Conectar CPU	Conecte la alimentación y la CPU.	-
"Desactivar" regulador de posición	Ponga a cero la ganancia del lazo de regulación de posición (factor Kv). (Sirve para evitar movimientos involuntarios del accionamiento a causa de posibles errores de parametrización en el lazo de regulación de posición.)	"Objeto tecnológico > Configuración > Parámetros avanzados > Lazo de regulación"
Cargar proyecto en la CPU	Ponga la CPU al estado operativo STOP. Cargue el proyecto en la CPU (cargar hardware y software).	• "Barra de herramientas > Parar CPU" • "Barra de herramientas > Cargar en dispositivo"

Paso	Acción	Soportada por el TIA Portal
Establecer una conexión online con la CPU	En "Online y Diagnóstico > Accesos online" active la casilla de verificación "Recibir avisos". Ajuste la interfaz del TIA Portal y establezca una conexión online con la CPU.	• Configuración del dispositivo • "Online y diagnóstico > Accesos online"
Desactivar un programa de usuario específico de Motion Control	Para evitar conflictos con el panel de mando del eje, bloquee la habilitación de los objetos tecnológicos en el programa de usuario (MC_Power.Enable = FALSE).	• Programación PLC • Instrucciones de Motion Control
Evaluar avisos pendientes	Evalúe el visor de avisos en la ventana de inspección. Solucione las causas de las alarmas tecnológicas pendientes. Acuse las alarmas tecnológicas (Página 10785).	"Ventana de inspección > Diagnóstico > Visor de avisos"
Comprobar finales de carrera por hardware	Pulse los finales de carrera por hardware. Compruebe que el visor de avisos sea correcto (alarma tecnológica 531). Acuse la alarma tecnológica.	"Ventana de inspección > Diagnóstico > Visor de avisos"
Comprobar la conexión y configuración del accionamiento (consigna)	Ponga la CPU al estado operativo RUN. Abra el Panel de mando del eje (Página 10775) y tome el control. Realice los pasos siguientes: • Habilite el objeto tecnológico. ⇒ El accionamiento tiene que conectarse y, dado el caso, soltar el freno. La posición se mantiene. • Mueva el eje en sentido positivo en modo Jog a velocidad reducida. ⇒ El accionamiento tiene que moverse. El valor real de posición debe aumentar (sentido positivo). • Bloquee el objeto tecnológico. ⇒ El accionamiento tiene que desconectarse y, dado el caso, cerrar el freno.	"Objeto tecnológico > Puesta en servicio > Panel de mando del eje"
Comprobar la conexión y configuración del encóder (valor real)	Compruebe la escala de los valores reales (sentido de giro, valoración del recorrido y resolución del encóder). ⇒ El cambio mecánico real de la posición debe coincidir con el cambio de los valores reales.	• "Objeto tecnológico > Diagnóstico > Telegrama PROFIdrive" • "Objeto tecnológico > Puesta en servicio > Panel de mando del eje"
Comprobar la velocidad de giro de referencia	Mueva el eje en sentido positivo en modo Jog a velocidad reducida. ⇒ La velocidad actual visualizada debe coincidir con la consigna de velocidad. Si la velocidad actual indicada difiere considerablemente de la consigna de velocidad, adapte la velocidad de giro de referencia.	• "Objeto tecnológico > Interfaz de hardware > Intercambio de datos" • "Objeto tecnológico > Puesta en servicio > Panel de mando del eje"
Optimizar el regulador de posición	Determine la ganancia óptima Kv del lazo de regulación de posición con la función de puesta en servicio Optimización. Adapte para ello los límites de error de seguimiento si fuera necesario.	"Objeto tecnológico > Puesta en servicio > Optimización"
Aplicar la ganancia Kv en el proyecto.	Introduzca en la configuración la ganancia Kv determinada con la función de optimización. Cargue el proyecto en la CPU.	"Objeto tecnológico > Configuración > Parámetros avanzados > Lazo de regulación"
Activar un programa de usuario específico de Motion Control	Elimine el bloqueo de habilitación de los objetos tecnológicos en el programa de usuario (MC_Power.Enable = TRUE).	• Programación PLC • Instrucciones de Motion Control

Paso	Acción	Soportada por el TIA Portal
Comprobar el funcionamiento del programa de usuario	Compruebe las funciones programadas en el programa de usuario.	• Tablas de observación y forzado permanente • Funciones online y de diagnóstico
Fin de la puesta en servicio para un objeto tecnológico Eje de posicionamiento	Vuelva a realizar los pasos descritos para poner en servicio otros objetos tecnológicos.	Véase más arriba.

Consulte también

Función y estructura de la optimización (Página 10779)

Panel de mando del eje

Función y estructura del panel de mando del eje

Descripción

Con el panel de mando del eje se desplazan diferentes ejes.

Para la operación del panel de mando del eje no se requiere ningún programa de usuario. El panel de mando del eje permite asumir el control de un objeto tecnológico y controlar los movimientos del eje.

⚠ ADVERTENCIA

Movimientos descontrolados del eje

Cuando se opera con el panel de mando del eje, es posible que el este ejecute movimientos descontrolados (p. ej. debido a una configuración errónea del accionamiento o del objeto tecnológico).

Por consiguiente, tome las medidas de protección siguientes antes de hacer funcionar el eje con el panel de mando:

- Asegúrese de que el interruptor de parada de emergencia está al alcance del operador.
- Active los finales de carrera por hardware.
- Active los finales de carrera por software.
- Asegúrese de que esté activada la vigilancia de errores de seguimiento.

En el árbol del proyecto encontrará los paneles de mando de los objetos tecnológicos Eje de velocidad de giro y Eje de posicionamiento en "Objeto tecnológico > Puesta en servicio".

El panel de mando del eje está dividido en las siguientes áreas:

- Control maestro
- Eje
- Modo de operación

- Forzar
- Estado del eje
- Valores actuales

La tabla siguiente muestra una relación de los elementos del panel de mando del eje:

Área	Elemento	Descripción
Control maestro		El área "Control maestro" permite asumir el control del objeto tecnológico o devolverlo al programa de usuario.
	Botón "Activar"	Con en el botón "Activar" se establece una conexión online con la CPU y se asume el control del objeto tecnológico seleccionado. • Para que sea posible asumir el control, el objeto tecnológico tiene que estar bloqueado en el programa de usuario. • Si la conexión online con la CPU falla mientras se opera el eje con el panel de mando, el eje se detendrá con la deceleración máxima una vez transcurrida la vigilancia de señal de vida. En este caso se muestra un mensaje de error ("ErrorID" = 16#8013) y el control se devuelve al programa de usuario. Si el control del panel de mando del eje se pierde repetidamente sin un mensaje de error directo, es posible que la conexión online con la CPU falle debido a una sobrecarga en la comunicación. En este caso, en el fichero del visor de avisos se muestra el aviso "Error durante la puesta en marcha. Fallo de la señal de vida entre el controlador y el TIA Portal". Para solucionar este error debe adaptarse la vigilancia de señal de vida en la variable del objeto tecnológico <TO>.ControlPanel.Input.TimeOut. El valor predeterminado de la variable es de 2000 ms. Dicho valor puede cambiarse directamente. El cambio será efectivo cuando vuelva a asumirse el control. • Hasta que no se devuelve el control, el programa de usuario no puede influir en las funciones del objeto tecnológico. Las peticiones de Motion Control del programa de usuario para el objeto tecnológico se rechazarán con un error ("ErrorID" = 16#8012: panel de mando del eje activado). • Al tomar el control, se aplica la configuración del objeto tecnológico. Los cambios en la configuración del objeto tecnológico no son efectivos hasta que no se devuelve el control. Por lo tanto, realice los cambios que sean necesarios antes de asumir el control. • Una vez se ha asumido el control del objeto tecnológico, el panel de mando del eje está bloqueado para el acceso desde otra instalación del TIA Portal (Team Engineering a partir de CPU V1.5).
	Botón "Desactivar"	Con el botón "Desactivar" se devuelve el control al programa de usuario.
Eje		En el área "Eje" se habilita o bloquea el objeto tecnológico.
	Botón "Habilitar"	Con el botón "Habilitar" se habilita el objeto tecnológico seleccionado.
	Botón "Bloquear"	Con el botón "Bloquear" se bloquea el objeto tecnológico seleccionado.
Modo de operación		Seleccione el modo de operación deseado para el panel de mando del eje en la lista desplegable "Modo de operación".
Forzar		En el área "Forzar" se muestran los parámetros para el desplazamiento con el panel de mando del eje de acuerdo con el modo de operación seleccionado.
	Posición	Posición a la que se referencia el eje. (Solo en los modos de operación Referenciar, Establecer punto de referencia)

Área	Elemento	Descripción
	Recorrido	Recorrido del eje. (Solo en el modo de operación Posicionamiento relativo)
	Posición de destino	Posición a la que se mueve el eje. (Solo en el modo de operación Posicionamiento absoluto)
	Velocidad/ Consigna de velocidad	Velocidad o velocidad de giro con la que se mueve el eje. Ajuste predeterminado: 10 % del valor predeterminado (Solo en los modos de operación Velocidad consignada/Especificación de velocidad de giro, Jog, Posicionamiento)
	Aceleración	Aceleración con la que se mueve el eje. Ajuste predeterminado: 10 % del valor predeterminado
	Deceleración	Deceleración con la que se mueve el eje. Ajuste predeterminado: 10 % del valor predeterminado
	Tirón	Tirón con el que se mueve el eje. Ajuste predeterminado: 100 % del valor predeterminado
	Botón "Inicio"	Con el botón "Inicio" se lanza una petición conforme al modo de operación seleccionado.
	Botón "Adelante"	Con el botón "Adelante" se inicia un movimiento en sentido positivo conforme al modo de operación seleccionado.
	Botón "Atrás"	Con el botón "Atrás" se inicia un movimiento en sentido negativo conforme al modo de operación seleccionado.
	Botón "Parada"	Con el botón "Parada" se cancela una petición o se detiene el eje.
	Botón "Ajustar punto de referencia"	Con el botón "Establecer punto de referencia" se pone la posición real al valor indicado en "Posición". (Solo en el modo de operación Establecer punto de referencia)
Estado del eje		En el área "Estado eje" se muestra el estado del eje y del accionamiento.
	Accionamiento listo	El accionamiento está preparado para ejecutar consignas.
	Error	Ha ocurrido un error en el objeto tecnológico.
	Habilitado	El objeto tecnológico está habilitado. El eje puede desplazarse con peticiones de movimiento.
	Referenciado	El objeto tecnológico está referenciado.
	Más	Mediante el enlace "Más" se accede al cuadro de diálogo "Objeto tecnológico > Diagnóstico > Bits de estado y de error".
	Error pendiente	En el campo de texto "Error pendiente" se muestra el último error ocurrido.
	Botón "Acusar"	Con el botón "Acusar" se acusan errores pendientes.
Valores actuales		En el área "Valores actuales" se muestran los valores reales del eje.
	Posición	Posición real del eje
	Velocidad	Velocidad real del eje

Nota

Los parámetros no se aplican

Los valores de parámetros ajustados se rechazan al devolver el control. En caso necesario, transfiera los valores a la configuración.

Si se modifican valores de configuración durante el funcionamiento con el panel de mando del eje, estos cambios no afectarán al funcionamiento del panel de mando del eje.

Modo de operación

La tabla siguiente explica los modos de operación del panel de mando del eje:

Modo de operación	Descripción
Referenciado	Esta función equivale al referenciado activo. Los parámetros de referenciado (Página 10653) deben estar configurados. El referenciado no es posible con un encóder absoluto. En caso de utilizar este modo de operación con un encóder absoluto, el objeto tecnológico no se referenciará.
Establecer punto de referencia	Esta función equivale al referenciado directo (absoluto). Con el botón "Establecer punto de referencia" se pone la posición real al valor indicado en "Posición" y se establece el estado "Referenciado".
Jog	Los comandos de movimiento se llevan a cabo por Jog. Con el botón "Adelante" o "Atrás" se inicia un movimiento en sentido positivo o negativo, respectivamente. El movimiento se ejecuta mientras se mantiene pulsado el botón del ratón.
Velocidad consignada/ Especificación de velocidad de giro	El eje se mueve a la velocidad o a la velocidad de giro indicada hasta que se detiene el movimiento. Los comandos de movimiento se ejecutan de acuerdo con las consignas parametrizadas en "Control".
Posicionamiento relativo	El posicionamiento se lleva a cabo como movimiento de desplazamiento relativo y regulado conforme a las especificaciones parametrizadas en "Control".
Posicionamiento absoluto	El posicionamiento se lleva a cabo como movimiento de desplazamiento absoluto y regulado conforme a las especificaciones parametrizadas en "Control". Si se ha activado el ajuste "Módulo" del objeto tecnológico, en el área "Control" se muestran los botones "Adelante" y "Atrás". El eje se posiciona dentro del área del módulo. Las especificaciones de posición fuera del área del módulo se convierten a esta última en correspondencia. Si no se ha activado el ajuste "Módulo" del objeto tecnológico, en el área "Control" solo se muestra el botón "Inicio". Se puede acceder directamente a la posición introducida.

Utilizar el panel de mando del eje

Requisitos

- La CPU está en el estado operativo RUN.
- El proyecto está creado y cargado en la CPU.

- El objeto tecnológico está bloqueado desde el programa de usuario (MC_Power.Enable = FALSE).
- El panel de mando del eje para el objeto tecnológico no es utilizado por ninguna otra instalación del TIA Portal (Team Engineering a partir de CPU V1.5).

Procedimiento

Para controlar el eje con el panel de mando, proceda del siguiente modo:

1. Haga clic en el botón "Recoger" del área "Control" para tomar el control del objeto tecnológico y establecer una conexión online con la CPU.
   Se muestra una advertencia. Confirme con "Aceptar".
2. Haga clic en el botón "Habilitar" del área "Eje" para habilitar el objeto tecnológico.
3. Seleccione la función deseada del panel de mando del eje en la lista desplegable del área "Modo de operación".
4. Indique los valores de parámetros correspondientes a la petición en el área "Forzar".
5. Haga clic en el botón "Inicio", "Adelante" o "Hacia atrás", en función del modo de operación ajustado, para iniciar la petición.
6. Para detener la petición, haga clic en el botón "Parada".
7. Repita los pasos 3 a 6 para otras peticiones.
8. Haga clic en el botón "Bloquear" del área "Eje" para bloquear el objeto tecnológico.
9. Haga clic en el botón "Devolver" del área "Control" para devolver el control al programa de usuario.

Optimización

Función y estructura de la optimización

Descripción

La función "Optimización" le ayuda a determinar la ganancia óptima (factor Kv) para la regulación de posición del eje. Para ello, con la función Trace se registra la curva de velocidad del eje durante un movimiento de posicionamiento especificable. Seguidamente, existe la posibilidad de evaluar el registro y adaptar la ganancia en correspondencia.

En el árbol del proyecto encontrará la función "Optimización" del objeto tecnológico Eje de posicionamiento en "Objeto tecnológico > Puesta en servicio".

El cuadro de diálogo "Optimización" está dividido en las siguientes áreas:

- Control maestro
- Eje
- Optimizar ganancia
- Trace

La tabla siguiente explica los elementos de la optimización:

Área	Elemento	Descripción
Control maestro		El área "Control maestro" permite asumir el control del objeto tecnológico o devolverlo al programa de usuario.
	Botón "Activar"	Con en el botón "Activar" se establece una conexión online con la CPU y se asume el control del objeto tecnológico seleccionado. • Para que sea posible asumir el control, el objeto tecnológico tiene que estar bloqueado en el programa de usuario. • Si la conexión online con la CPU falla mientras se opera el eje con el panel de mando, el eje se detendrá con la deceleración máxima una vez transcurrida la vigilancia de señal de vida. En este caso se muestra un mensaje de error ("ErrorID" = 16#8013) y el control se devuelve al programa de usuario. Si el control del panel de mando del eje se pierde repetidamente sin un mensaje de error directo, es posible que la conexión online con la CPU falle debido a una sobrecarga en la comunicación. En este caso, en el fichero del visor de avisos se muestra el aviso "Error durante la puesta en marcha. Fallo de la señal de vida entre el controlador y el TIA Portal". Para solucionar este error debe adaptarse la vigilancia de señal de vida en la variable del objeto tecnológico <TO>.ControlPanel.Input.TimeOut. El valor predeterminado de la variable es de 2000 ms. Dicho valor puede cambiarse directamente. El cambio será efectivo cuando vuelva a asumirse el control. • Hasta que no se devuelve el control, el programa de usuario no puede influir en las funciones del objeto tecnológico. Las peticiones de Motion Control del programa de usuario para el objeto tecnológico se rechazarán con un error ("ErrorID" = 16#8012: panel de mando del eje activado). • Al tomar el control, se aplica la configuración del objeto tecnológico. Los cambios en la configuración del objeto tecnológico no son efectivos hasta que no se devuelve el control. Por lo tanto, realice los cambios que sean necesarios antes de asumir el control. • Una vez se ha asumido el control del objeto tecnológico, el panel de mando del eje está bloqueado para el acceso desde otra instalación del TIA Portal (Team Engineering a partir de CPU V1.5).
	Botón "Desactivar"	Con el botón "Desactivar" se devuelve el control al programa de usuario.
Eje		En el área "Eje" se habilita o bloquea el objeto tecnológico.
	Botón "Habilitar"	Con el botón "Habilitar" se habilita el objeto tecnológico seleccionado.
	Botón "Bloquear"	Con el botón "Bloquear" se bloquea el objeto tecnológico seleccionado.
Optimizar ganancia		En el área "Optimizar ganancia" se llevan a cabo los ajustes para optimizar la ganancia.
	Ganancia	Ganancia actual del regulador de posición (Kv) Con el símbolo situado junto al campo se abre una lista de valores. La lista de valores contiene los valores siguientes de la ganancia: • Valor actual online • Valor inicial online • Valor inicial de proyecto Introduzca el valor nuevo de la ganancia en el campo de entrada del valor real. El valor nuevo se aplica haciendo clic en el botón "Adelante" o "Atrás".
	Recorrido	Recorrido de un incremento de prueba
	Tiempo de medición	Duración de un incremento de prueba

Área	Elemento	Descripción
	Casilla de verificación "Adaptar valores dinámicos"	La casilla de verificación "Adaptar valores dinámicos" permite adaptar la aceleración y la velocidad máxima.
	Aceleración	Ajuste predeterminado de la aceleración para un incremento de prueba
	Velocidad máxima	Ajuste predeterminado de la velocidad máxima para un incremento de prueba
	Botón "Adelante"	Con el botón "Adelante" se inicia un incremento de prueba en sentido positivo para la optimización.
	Botón "Atrás"	Con el botón "Atrás" se inicia un incremento de prueba en sentido negativo para la optimización.
Trace		En la zona inferior del cuadro de diálogo "Optimización" se muestra la función Trace. Con cada incremento de prueba se inicia automáticamente un registro Trace de los parámetros necesarios, el cual se visualiza tras finalizar el incremento de prueba. Una vez que se ha devuelto el control se borra el registro Trace. Encontrará una descripción completa de la función Trace en el capítulo Función Trace y analizador lógico.

Nota

Los parámetros no se aplican

Los valores de parámetros ajustados se rechazan tras devolver el control.

En caso necesario, transfiera los valores a la configuración. El valor de ganancia puede aplicarse en la configuración a través de "Valor inicial de proyecto".

Consulte también

Regulación (Página 10670)

Optimizar el regulador de posición

Requisitos

- La CPU está en el estado operativo RUN.
- El proyecto está creado y cargado en la CPU.
- El objeto tecnológico está bloqueado desde el programa de usuario (MC_Power.Enable = FALSE).
- El panel de mando del eje para el objeto tecnológico no es utilizado por ninguna otra instalación del TIA Portal (Team Engineering a partir de CPU V1.5).

Optimizar la ganancia del regulador de posición (Kv)

Para optimizar la ganancia (Kv), proceda del siguiente modo:

1. Haga clic en el botón "Recoger" del área "Control" para tomar el control del objeto tecnológico y establecer una conexión online con la CPU.
   Se muestra una advertencia. Confirme con "Aceptar".
2. Haga clic en el botón "Habilitar" del área "Eje" para habilitar el objeto tecnológico.
3. En caso necesario, configure los valores de recorrido, duración y dinámica de un incremento de prueba.
4. Introduzca un valor inicial para la ganancia. Empiece con un valor bajo.
5. Haga clic en el botón "Adelante" o "Atrás" para iniciar un incremento de prueba para la optimización en sentido positivo o negativo, respectivamente.
   Durante el tiempo indicado se emitirá una consigna de acuerdo con el recorrido indicado. El eje recorre el trayecto indicado. Automáticamente se crea un registro Trace del movimiento (consignas y valores reales).
   Encontrará una descripción completa de la función Trace en el capítulo Función Trace y analizador lógico.

   Nota

   Adaptar límites de error de seguimiento

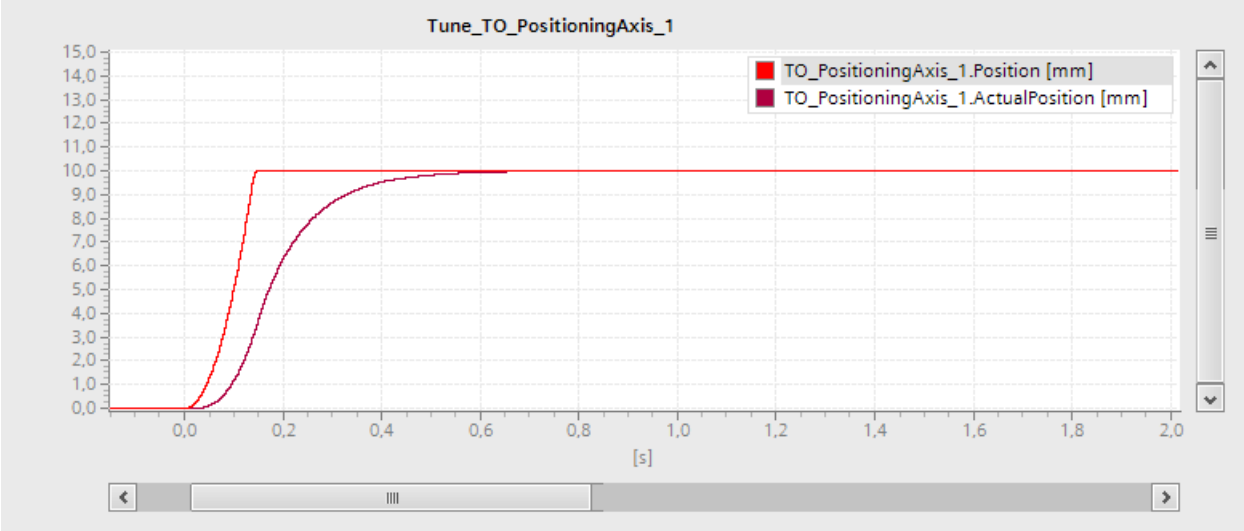
   Si durante la optimización se muestran repetidamente mensajes de error por medio de la vigilancia de errores de seguimiento, adapte por tiempos los límites de error de seguimiento.

6. Evalúe la curva del registro Trace. Adapte la ganancia por incrementos. Cada vez que introduzca un valor, haga clic en el botón "Adelante" o "Atrás". Aplique el valor e inicie un movimiento nuevo y el registro Trace.

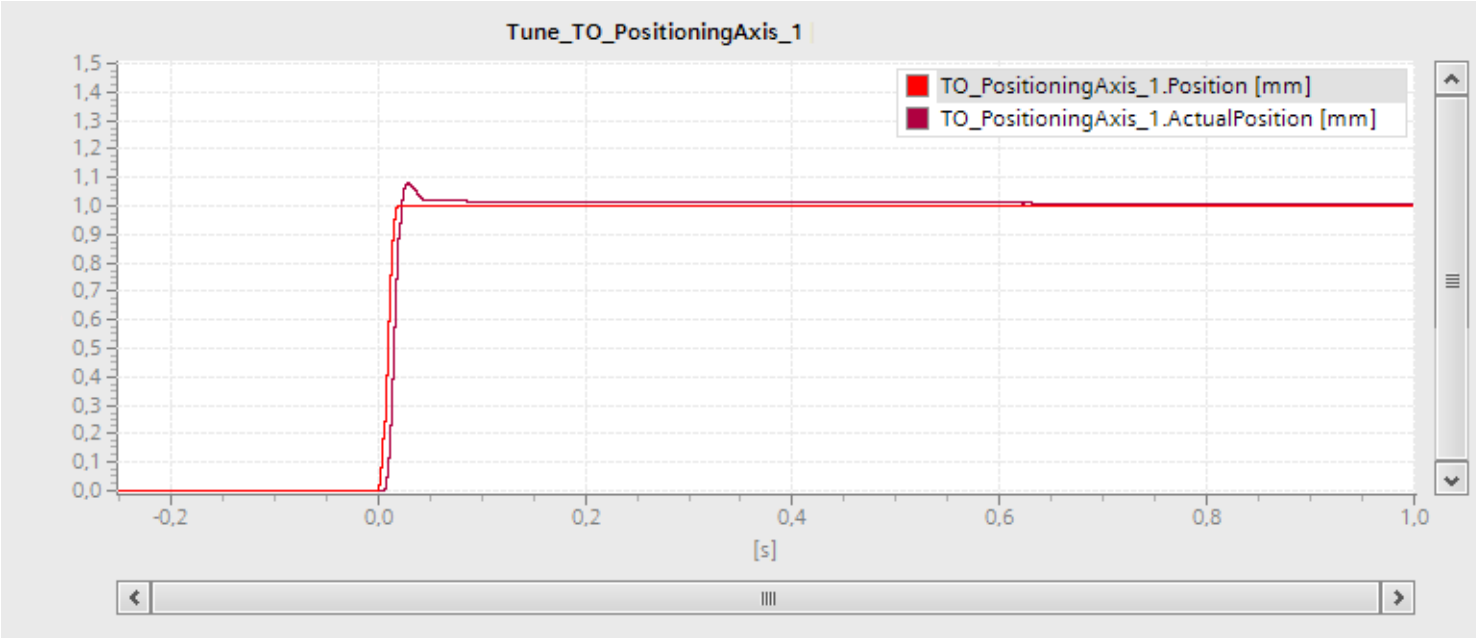
Al adaptar la ganancia, tenga en cuenta las propiedades siguientes de la curva:

- La curva presenta un tiempo de estabilización corto.
- La curva no muestra ninguna inversión de movimiento de la posición real.
- Al aproximarse a la posición consignada no se produce ningún rebase transitorio.
- La curva presenta un comportamiento global estable (sin oscilaciones).

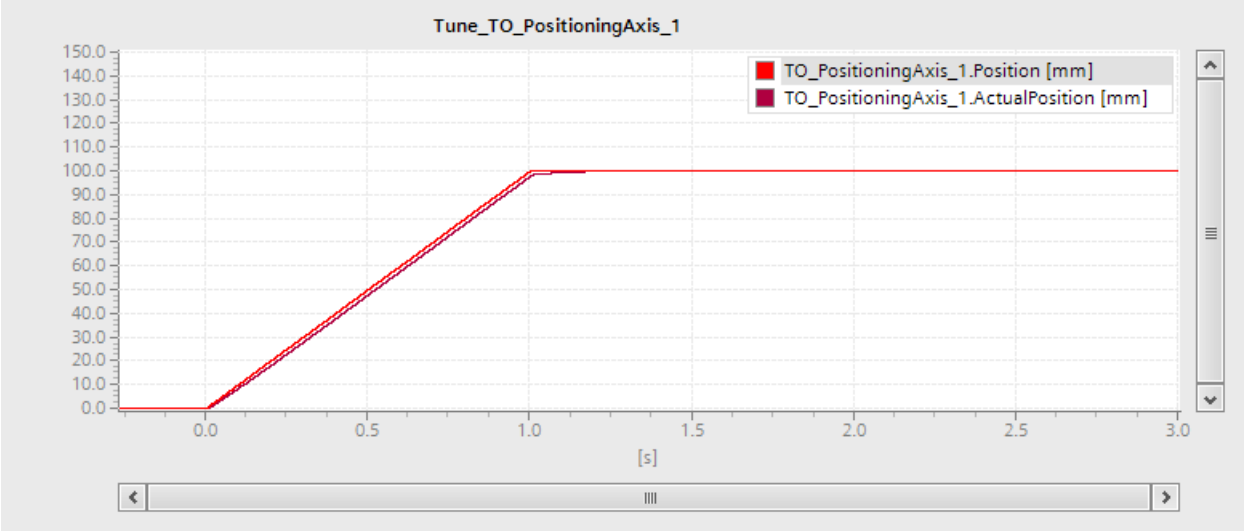
El registro Trace siguiente muestra una curva con un tiempo de estabilización largo:



El registro Trace siguiente muestra una curva con un rebase transitorio al aproximarse a la consigna:



El registro Trace siguiente muestra una curva con una ganancia optimizada y un comportamiento general estable:



Aplicar la ganancia del regulador de posición (Kv) al proyecto

Para aplicar la ganancia (Kv) determinada al proyecto, proceda del siguiente modo:

1. Haga clic en el símbolo situado junto al campo "Ganancia".
   Se abre una lista de valores.
2. Introduzca el valor determinado de la ganancia en el campo "Valor de arranque proyecto" de la lista de valores.
3. Haga clic en el botón "Bloquear" del área "Eje" para bloquear el objeto tecnológico.
4. Haga clic en el botón "Devolver" del área "Control" para devolver el control al programa de usuario.
5. Cargue el proyecto en la CPU.

Diagnóstico

Introducción

El capítulo Diagnóstico se limita a la descripción del concepto de diagnóstico para Motion Control y a la descripción de la vista de diagnóstico de los diferentes objetos tecnológicos en el TIA Portal.

Encontrará una descripción completa del diagnóstico de sistema de la CPU S7-1500 en el manual de funciones "Diagnóstico del sistema (http://support.automation.siemens.com/WW/view/es/59192926)".

Concepto de diagnóstico

El concepto de diagnóstico abarca alarmas y los avisos correspondientes, así como mensajes de error para las instrucciones de Motion Control. Además, el TIA Portal dispone de comprobaciones de coherencia que le ayudan a configurar los objetos tecnológicos y a crear el programa de usuario.

Todas las alarmas que se producen durante el funcionamiento (de la CPU, tecnología, hardware, etc.) se muestran en la ventana de inspección del TIA Portal. La información de diagnóstico relacionada con los objetos tecnológicos (alarmas tecnológicas, información de estado) se muestran además en la ventana de diagnóstico del objeto tecnológico en cuestión.

Si se produce un error en un objeto tecnológico al realizar un movimiento (p. ej. aproximación a un final de carrera por hardware), se lanza una alarma tecnológica (Página 10785) y se visualiza un aviso al respecto tanto en el TIA Portal como en los dispositivos HMI).

En el programa de usuario, las alarmas tecnológicas se notifican básicamente mediante bits de error en el bloque de datos tecnológico. Siempre se muestra la alarma tecnológica de mayor prioridad. Con el fin de simplificar la evaluación de errores, mediante los parámetros "Error" y "ErrorID" de las instrucciones de Motion Control también se indica que hay una alarma tecnológica pendiente.

Durante la parametrización o en la secuencia de procesamiento de las instrucciones de Motion Control pueden producirse errores de programa (Página 10788) (p. ej. indicación de parámetros no admisible en la llamada de la instrucción o lanzamiento de una petición sin habilitación con MC_Power). Los errores en las instrucciones de Motion Control se muestran al llamar las instrucciones con los parámetros "Error" y "ErrorID".

Alarmas tecnológicas

Descripción

Si se produce un error en el objeto tecnológico (p. ej. aproximación a un final de carrera por hardware), se lanza y visualiza una alarma tecnológica. Las repercusiones de una alarma tecnológica sobre el objeto tecnológico están definidas por la reacción a alarma.

Clases de alarma

Las alarmas tecnológicas están divididas en tres clases:

- Advertencia (acusable)
  Se continúa con el procesamiento de las peticiones de Motion Control. El movimiento en curso del eje puede verse influido, p. ej. por una limitación de los valores dinámicos actuales a los valores límite configurados.
- Alarma (acuse obligatorio)
  Las peticiones de movimiento se cancelan de acuerdo con la reacción a alarma. Para continuar procesando la petición tras solucionar la causa del error, es necesario acusar las alarmas.
- Error grave
  Las peticiones de movimiento se cancelan de acuerdo con la reacción a alarma.
  Para poder utilizar de nuevo el objeto tecnológico una vez solucionada la causa del error hay que reinicializar el objeto tecnológico (Página 10768).

Visualización de alarmas tecnológicas

Una alarma tecnológica se visualiza en los lugares siguientes:

- TIA Portal
  - "Objeto tecnológico > Diagnóstico > Bits de estado y de error"
    Visualización de alarmas tecnológicas pendientes por objeto tecnológico.
  - "Objeto tecnológico > Puesta en servicio > Panel de mando del eje"
    Visualización de la última alarma tecnológica pendiente por objeto tecnológico.
  - "Ventana de inspección > Diagnóstico > Visor de avisos"
    En "Online y Diagnóstico > Accesos online" active la casilla de verificación "Recibir avisos" para visualizar alarmas tecnológicas en el visor de avisos.
    En caso de conexión online con la CPU se muestran las alarmas tecnológicas pendientes de todos los objetos tecnológicos. Adicionalmente, está disponible la vista de ficheros.
    También es posible activar y visualizar el visor de avisos en un HMI conectado.
  - "CPU > Online y diagnóstico"
    Visualización de las alarmas tecnológicas introducidas en el búfer de diagnóstico de la CPU.
- Programa de usuario
  - Variables <TO>.ErrorDetail.Number y <TO>.ErrorDetail.Reaction
    Visualización del número y la reacción de la última alarma tecnológica aparecida.
  - Variable <TO>.StatusWord
    Mediante el bit "Error" se indica que hay una alarma tecnológica pendiente.
  - Variable <TO>.ErrorWord
    Visualización de alarmas y errores graves.
  - Variable <TO>.WarningWord
    Visualización de advertencias.
  - Parámetros "Error" y "ErrorID"
    En una instrucción de Motion Control, los parámetros "Error" = TRUE y "ErrorID" = 16#8001 indican que hay una alarma tecnológica pendiente.
- Pantalla de la CPU
  Para que las alarmas tecnológicas se muestren en la pantalla de la CPU, realice el siguiente ajuste al cargar en la CPU:
  En el cuadro de diálogo "Vista preliminar Carga", seleccione la acción "Carga coherente" para la entrada "Librerías de textos".

Reacción a alarma

Una alarma tecnológica implica siempre una reacción a alarma que describe la repercusión sobre el objeto tecnológico. La reacción a alarma está predefinida por el sistema.

La enumeración siguiente muestra posibles reacciones a alarmas con prioridad ascendente:

- Ninguna reacción (solo advertencias)
  <TO>.ErrorDetail.Reaction = 0
  Se continúa con el procesamiento de las peticiones de Motion Control. El movimiento en curso del eje puede verse influido, p. ej. por una limitación de los valores dinámicos actuales a los valores límite configurados.
- Parada con valores dinámicos actuales
  <TO>.ErrorDetail.Reaction = 1
  Los comandos de movimiento en curso se cancelan. El eje se frena con los valores dinámicos pendientes en la instrucción de Motion Control y se para.
- Parada con valores dinámicos máximos
  <TO>.ErrorDetail.Reaction = 2
  Los comandos de movimiento en curso se cancelan. El eje se frena con los valores dinámicos máximos configurados en "Objeto tecnológico > Parámetros avanzados > Valores dinámicos" y se para. En este caso se tiene en cuenta el tirón máximo configurado.
  Parada con rampa de parada de emergencia
  <TO>.ErrorDetail.Reaction = 3
  Los comandos de movimiento en curso se cancelan. El eje se frena sin limitación de tirones con la deceleración de parada de emergencia configurada en "Objeto tecnológico > Parámetros avanzados > Rampa de parada de emergencia" y se para.
- Retirar habilitación
  <TO>.ErrorDetail.Reaction = 4
  Se emite la consigna cero y se retira la habilitación. En función de la configuración, el eje se frena en el accionamiento y se para.

Acusar alarmas tecnológicas

Las alarmas tecnológicas se acusan del modo siguiente:

- TIA Portal
  - "Objeto tecnológico > Diagnóstico > Bits de estado y de error"
    Haga clic en el botón "Acusar" para acusar todas las alarmas y advertencias pendientes del objeto tecnológico seleccionado.
  - "Objeto tecnológico > Puesta en servicio > Panel de mando del eje"
    Haga clic en el botón "Acusar" para acusar todas las alarmas y advertencias pendientes del objeto tecnológico seleccionado.
  - "Ventana de inspección > Diagnóstico > Visor de avisos"
    Las alarmas y advertencias de todos los objetos tecnológicos pueden acusarse individualmente o en grupo.
    En un HMI también es posible acusar las alarmas y advertencias de todos los objetos tecnológicos individualmente o en grupo si el visor de avisos está activado.
- Programa de usuario
  Acuse las alarmas tecnológicas pendientes de un objeto tecnológico con la instrucción "MC_Reset" de Motion Control.

Información adicional

Encontrará una lista de las alarmas tecnológicas y las reacciones a alarma en el anexo Alarmas tecnológicas (Página 10841).

Errores en las instrucciones de Motion Control

Descripción

Los errores en las instrucciones de Motion Control (p. ej. indicación de un valor de parámetro no válido) se muestran con los parámetros de salida "Error" y "ErrorID".

En las condiciones siguientes, la instrucción "Error" de Motion Control presenta TRUE y "ErrorID" presenta 16#8xxx:

- Estado inadmisible del objeto tecnológico que impide la ejecución de la petición.
- Parametrización inadmisible de la instrucción de Motion Control que impide la ejecución de la petición.
- Como consecuencia de la reacción a alarma de un error en el objeto tecnológico.

Indicador de errores

Cuando hay un error pendiente en una instrucción de Motion Control, el parámetro "Error" presenta el valor TRUE. La causa del error se puede consultar en el valor del parámetro "ErrorID" ".

Si "Error" = TRUE, las peticiones para el objeto tecnológico se rechazan. Las peticiones en curso no se ven influidas por peticiones rechazadas.

Si mientras se procesa la petición "Error" = TRUE y "ErrorID" = 16#8001, significa que se ha producido una alarma tecnológica. Evalúe en este caso la visualización de la alarma tecnológica.

Si durante el procesamiento de una petición "MC_MoveJog" se muestra "Error" = TRUE , el eje se frena hasta detenerse. En este caso actúa la deceleración que se había parametrizado en la instrucción "MC_MoveJog".

Acusar errores

No es necesario acusar errores en las instrucciones de Motion Control.

Reinicie la petición tras solucionar el error.

Información adicional

Encontrará una lista de las ErrorIDs en el anexo Identificador de error (Página 10844).

Objeto tecnológico Eje de velocidad de giro

Bits de estado y de error

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Bits de estado y de error" permite vigilar los principales avisos de estado y mensajes de error del objeto tecnológico en el TIA Portal. La función de diagnóstico está disponible en modo online.

En las tablas siguientes se explica el significado de los avisos de estado y mensajes de error. Entre paréntesis se indica la variable correspondiente del objeto tecnológico.

Estado del eje

La tabla siguiente muestra los posibles estados del eje:

Estado	Descripción
Habilitado	El objeto tecnológico está habilitado. El eje puede desplazarse con peticiones de movimiento. (<TO>.StatusWord.Enable)
Error	Ha ocurrido un error en el objeto tecnológico. Encontrará información detallada sobre el error en el área "Errores" y en las variables del objeto tecnológico <TO>.ErrorDetail.Number y <TO>.ErrorDetail.Reaction. (<TO>.StatusWord.Error)
Reinicialización activa	El objeto tecnológico se reinicializa. (<TO>.StatusWord.RestartActive)
Panel de mando del eje activo	El panel de mando del eje está activado. El panel de mando del eje tiene el control del objeto tecnológico. El eje no puede ser controlado por el programa de usuario. (<TO>.StatusWord.ControlPanelActive)
Accionamiento conectado	El accionamiento está preparado para ejecutar consignas. (<TO>.StatusDrive.InOperation)
Valor inicial online modificado	Se han modificado datos relevantes para la reinicialización. Los cambios no se aplican hasta que no se reinicializa el objeto tecnológico (<TO>.StatusWord.OnlineStartValuesChanged).

Estado del movimiento

La tabla siguiente muestra los posibles estados del movimiento del eje:

Estado	Descripción
Done (ninguna petición activa)	No hay ninguna petición de movimiento en el objeto tecnológico. (<TO>.StatusWord.Done)
Jog	El eje se mueve con una petición de modo Jog de la instrucción "MC_MoveJog" de Motion Control o del panel de mando del eje. (<TO>.StatusWord.JogCommand)

Estado	Descripción
Especificación de velocidad de giro	El eje se mueve con una petición con especificación de velocidad de giro de la instrucción "MC_MoveVelocity" de Motion Control o del panel de mando del eje. (<TO>.StatusWord.VelocityCommand)
Velocidad de giro constante	El eje avanza a velocidad de giro constante o está parado. (<TO>.StatusWord.ConstantVelocity)
Acelerar	El eje se acelera. (<TO>.StatusWord.Accelerating)
Decelerar	El eje se frena. (<TO>.StatusWord.Decelerating)

Advertencias

La tabla siguiente muestra las posibles advertencias:

Advertencia	Descripción
Configuración adaptada	Uno o varios parámetros de configuración se adaptan internamente de modo temporal. (<TO>.WarningWord.ConfigurationFault)
Petición rechazada	Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios. (<TO>.WarningWord.CommandNotAccepted)
Limitación dinámica	Los valores dinámicos se limitan a los límites dinámicos. (<TO>.WarningWord.DynamicError)

Errores

La tabla siguiente muestra los posibles errores:

Error	Descripción
Sistema	Ha ocurrido un error interno del sistema. (<TO>.ErrorWord.SystemFault)
Configuración	Error de configuración Hay uno o varios parámetros de configuración incoherentes o inadmisibles. El objeto tecnológico se ha configurado erróneamente o bien se han modificado incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa de usuario. (<TO>.ErrorWord.ConfigurationFault)
Programa de usuario	Error en una instrucción de Motion Control o en su utilización en el programa de usuario. (<TO>.ErrorWord.UserFault)
Accionamiento	Error en el accionamiento. (<TO>.ErrorWord.DriveFault)
Transferencia de datos	Hay interferencias en la comunicación con un dispositivo conectado. (<TO>.ErrorWord.CommunicationFault)

Error	Descripción
Periferia	Error al acceder a una dirección lógica. (<TO>.ErrorWord.PeripheralError)
Petición rechazada	Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios (p. ej. no se ha hecho referencia a un objeto tecnológico). (<TO>.ErrorWord.CommandNotAccepted)
Limitación dinámica	Los valores dinámicos se limitan a los límites dinámicos. (<TO>.ErrorWord.DynamicError)

Acusar

Haga clic en el botón "Acusar" para acusar las alarmas y advertencias pendientes.

Información adicional

Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Estado del movimiento

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Estado movimiento" permite vigilar el estado de movimiento del eje en el TIA Portal. La función de diagnóstico está disponible en modo online.

Área "Valores actuales"

La tabla siguiente describe el significado de la información de estado:

Estado	Descripción
Consigna de velocidad de giro	Consigna de velocidad de giro del eje (<TO>.Velocity)
Corrección de la velocidad de giro	Corrección porcentual de la especificación de velocidad de giro La consigna de velocidad de giro especificada por instrucciones de Motion Control o por el panel de mando del eje se sustituye y se adapta porcentualmente mediante una corrección. Como corrección de velocidad de giro se admiten valores desde 0.0% hasta 200.0%. (<TO>.Override.Velocity)

Área "Límites dinámicos"

En esta área se muestran los valores límite configurados para los parámetros dinámicos.

La tabla siguiente describe el significado de la información de estado:

Estado	Descripción
Velocidad de giro	Velocidad de giro máxima configurada (<TO>.DynamicLimits.MaxVelocity)
Aceleración	Aceleración máxima configurada (<TO>.DynamicLimits.MaxAcceleration)
Deceleración	Deceleración máxima configurada (<TO>.DynamicLimits.MaxDeceleration)

Telegrama PROFIdrive

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Telegrama PROFIdrive" permite vigilar el telegrama PROFIdrive del accionamiento al controlador en el TIA Portal. La función de diagnóstico está disponible en modo online.

Área "Accionamiento"

En esta área se muestran los parámetros siguientes del telegrama PROFIdrive del accionamiento al controlador:

- Las palabras de estado "ZSW1" y "ZSW2"
- La consigna de velocidad de giro dada al accionamiento (NSOLL)
- La velocidad de giro real notificada por el accionamiento (NIST)

Objeto tecnológico Eje de posicionamiento

Bits de estado y de error

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Bits de estado y de error" permite vigilar los principales avisos de estado y mensajes de error del objeto tecnológico en el TIA Portal. La función de diagnóstico está disponible en modo online.

En las tablas siguientes se explica el significado de los avisos de estado y mensajes de error. Entre paréntesis se indica la variable correspondiente del objeto tecnológico.

Estado del eje

La tabla siguiente muestra los posibles estados del eje:

Estado	Descripción
Habilitado	El objeto tecnológico está habilitado. El eje puede desplazarse con peticiones de movimiento. (<TO>.StatusWord.Enable)
Referenciado	El objeto tecnológico está referenciado. La relación entre la posición en el objeto tecnológico y la posición mecánica se ha establecido correctamente. (<TO>.StatusWord.HomingDone)
Error	Ha ocurrido un error en el objeto tecnológico. Encontrará información detallada sobre el error en el área "Errores" y en las variables del objeto tecnológico <TO>.ErrorDetail.Number y <TO>.ErrorDetail.Reaction. (<TO>.StatusWord.Error)
Reinicialización activa	El objeto tecnológico se reinicializa. (<TO>.StatusWord.RestartActive)
Panel de mando del eje activo	El panel de mando del eje está activado. El panel de mando del eje tiene el control del objeto tecnológico. El eje no puede ser controlado por el programa de usuario. (<TO>.StatusWord.ControlPanelActive)
Accionamiento conectado	El accionamiento está preparado para ejecutar consignas. (<TO>.StatusDrive.InOperation)
Encóder válido	Los valores del encóder son válidos. (<TO>.StatusSensor[n].State)
Valor inicial online modificado	Se han modificado datos relevantes para la reinicialización. Los cambios solo se aplican si se reinicializa el objeto tecnológico. (<TO>.StatusWord.OnlineStartValuesChanged)

Estado Final de carrera

La tabla siguiente muestra las posibles activaciones de los finales de carrera por software y hardware:

Final de carrera	Descripción
Final de carrera por software neg. alcanzado	Se ha alcanzado el final de carrera por software negativo. (<TO>.StatusWord.SWLimitMinActive)
Final de carrera por software neg. alcanzado	Se ha alcanzado el final de carrera por software positivo. (<TO>.StatusWord.SWLimitMaxActive)
Final de carrera por hardware neg. alcanzado	Se ha alcanzado o sobrepasado el final de carrera por hardware negativo. (<TO>.StatusWord.HWLimitMinActive)
Final de carrera por software pos. alcanzado	Se ha alcanzado o sobrepasado el final de carrera por hardware positivo. (<TO>.StatusWord.HWLimitMaxActive)

Estado del movimiento

La tabla siguiente muestra los posibles estados del movimiento del eje:

Estado	Descripción
Done (ninguna petición activa)	No hay ninguna petición de movimiento en el objeto tecnológico. (<TO>.StatusWord.Done)
Petición de referenciado	El objeto tecnológico ejecuta una petición de referenciado de la instrucción "MC_Home" de Motion Control o del panel de mando del eje. (<TO>.StatusWord.HomingCommand)
Jog	El eje se mueve con una petición de modo Jog de la instrucción "MC_MoveJog" de Motion Control o del panel de mando del eje. (<TO>.StatusWord.JogCommand)
Especificación de velocidad	El eje se mueve con una petición con especificación de velocidad de la instrucción "MC_MoveVelocity" de Motion Control o del panel de mando del eje. (<TO>.StatusWord.VelocityCommand)
Petición de posicionamiento	El eje se mueve con una petición de posicionamiento de la instrucción "MC_MoveAbsolute" o "MC_MoveRelative" de Motion Control o del panel de mando del eje. (<TO>.StatusWord.PositioningCommand)
Velocidad constante	El eje avanza a velocidad constante o está parado. (<TO>.StatusWord.ConstantVelocity)
Velocidad cero	El eje está parado. (<TO>.StatusWord.StandStill)
Acelerar	El eje se acelera. (<TO>.StatusWord.Accelerating)
Decelerar	El eje se frena. (<TO>.StatusWord.Decelerating)

Advertencias

La tabla siguiente muestra las posibles advertencias:

Advertencia	Descripción
Configuración adaptada	Uno o varios parámetros de configuración se adaptan internamente de modo temporal. (<TO>.WarningWord.ConfigurationFault)
Petición rechazada	Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios. (<TO>.WarningWord.CommandNotAccepted)
Limitación dinámica	Los valores dinámicos se limitan a los límites dinámicos. (<TO>.WarningWord.DynamicError)

Errores

La tabla siguiente muestra los posibles errores:

Error	Descripción
Sistema	Ha ocurrido un error interno del sistema. (<TO>.ErrorWord.SystemFault)
Configuración	Error de configuración Hay uno o varios parámetros de configuración incoherentes o inadmisibles. El objeto tecnológico se ha configurado erróneamente o bien se han modificado incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa de usuario. (<TO>.ErrorWord.ConfigurationFault)
Programa de usuario	Error en una instrucción de Motion Control o en su utilización en el programa de usuario. (<TO>.ErrorWord.UserFault)
Accionamiento	Error en el accionamiento. (<TO>.ErrorWord.DriveFault)
Encóder	Error en el sistema del encóder. (<TO>.ErrorWord.SensorFault)
Transferencia de datos	Hay interferencias en la comunicación con un dispositivo conectado. (<TO>.ErrorWord.CommunicationFault)
Periferia	Error al acceder a una dirección lógica. (<TO>.ErrorWord.PeripheralError)
Petición rechazada	Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios (p. ej. no se ha hecho referencia a un objeto tecnológico). (<TO>.ErrorWord.CommandNotAccepted)
Referenciado	Error en un proceso de referenciado. (<TO>.ErrorWord.HomingFault)
Posicionar	El eje no se ha posicionado correctamente al final de un movimiento de posicionamiento. (<TO>.ErrorWord.PositioningFault)
Limitación dinámica	Los valores dinámicos se limitan a los límites dinámicos. (<TO>.ErrorWord.DynamicError)
Error de seguimiento	Se ha rebasado por exceso el error de seguimiento máximo admisible. (<TO>.ErrorWord.FollowingErrorFault)
Final de carrera por software	Se ha alcanzado un final de carrera por software. (<TO>.ErrorWord.SwLimit)
Final de carrera de hardware	Se ha alcanzado o sobrepasado un final de carrera por hardware. (<TO>.ErrorWord.HWLimit)

Acusar

Haga clic en el botón "Acusar" para acusar las alarmas y advertencias pendientes.

Información adicional

Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Estado del movimiento

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Estado movimiento" permite vigilar el estado de movimiento del eje en el TIA Portal. La función de diagnóstico está disponible en modo online.

Área "Valores actuales"

La tabla siguiente describe el significado de la información de estado:

Estado	Descripción
Posición de destino	Posición de destino actual de una petición de posicionamiento activa El valor de la posición de destino solo es válido durante la ejecución de una petición de posicionamiento. (<TO>.StatusPositioning.TargetPosition)
Posición real	Posición real del eje Si el objeto tecnológico no está referenciado, el valor se indica en relación a la posición que había en el momento de habilitar el objeto tecnológico. (<TO>.ActualPosition)
Velocidad real	Velocidad real del eje (<TO>.ActualVelocity)
Corrección de la velocidad	Corrección porcentual de la especificación de velocidad La consigna de velocidad especificada por instrucciones de Motion Control o por el panel de mando del eje se sustituye y se adapta porcentualmente mediante una corrección. Como corrección de velocidad se admiten valores desde 0.0% hasta 200.0%. (<TO>.Override.Velocity)

Área "Límites dinámicos"

En esta área se muestran los valores límite configurados para los parámetros dinámicos.

La tabla siguiente describe el significado de la información de estado:

Estado	Descripción
Velocidad	Velocidad máxima configurada (<TO>.DynamicLimits.MaxVelocity)
Aceleración	Aceleración máxima configurada (<TO>.DynamicLimits.MaxAcceleration)
Deceleración	Deceleración máxima configurada (<TO>.DynamicLimits.MaxDeceleration)

Telegrama PROFIdrive

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Telegrama PROFIdrive" permite vigilar los telegramas PROFIdrive del accionamiento y el encóder en el TIA Portal. La visualización de la función de diagnóstico está disponible en modo online.

Área "Accionamiento"

En esta área se muestran los parámetros siguientes del telegrama PROFIdrive del accionamiento al controlador:

- Las palabras de estado "ZSW1" y "ZSW2"
- La consigna de velocidad de giro dada al accionamiento (NSOLL)
- La velocidad de giro real notificada por el accionamiento (NIST)

Área "Encóder"

En esta área se muestran los parámetros siguientes del telegrama PROFIdrive del encóder al controlador:

- La palabra de estado "G1_ZSW"
- El valor real de posición "G1_XIST1" (valor real cíclico del encóder)
- El valor real de posición "G1_XIST2" (valor absoluto del encóder)

Objeto tecnológico Encóder externo

Bits de estado y de error

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Bits de estado y de error" permite vigilar los principales avisos de estado y mensajes de error del objeto tecnológico en el TIA Portal. La función de diagnóstico está disponible en modo online.

En las tablas siguientes se explica el significado de los avisos de estado y mensajes de error. Entre paréntesis se indica la variable correspondiente del objeto tecnológico.

Estado del encóder

La tabla siguiente muestra los posibles estados del encóder externo:

Estado	Descripción
Habilitado	El objeto tecnológico está habilitado. (<TO>.StatusWord.Enable)
Referenciado	El objeto tecnológico está referenciado. La relación entre la posición en el objeto tecnológico y la posición mecánica se ha establecido correctamente. (<TO>.StatusWord.HomingDone)
Error	Ha ocurrido un error en el objeto tecnológico. Encontrará información detallada sobre el error en el área "Errores" y en las variables del objeto tecnológico <TO>.ErrorDetail.Number y <TO>.ErrorDetail.Reaction. (<TO>.StatusWord.Error)
Reinicialización activa	El objeto tecnológico se reinicializa. (<TO>.StatusWord.RestartActive)
Encóder válido	Los valores del encóder son válidos. (<TO>.StatusSensor[n].State)
Valor inicial online modificado	Se han modificado datos relevantes para la reinicialización. Los cambios no se aplican hasta que no se reinicializa el objeto tecnológico (<TO>.StatusWord.OnlineStartValuesChanged).

Estado del movimiento

La tabla siguiente muestra los posibles estados del movimiento del eje:

Estado	Descripción
Done (ninguna petición activa)	En el objeto tecnológico no hay ninguna petición de Motion Control activa. (excepto habilitación por petición "MC_Power") (<TO>.StatusWord.Done)
Petición de referenciado	El objeto tecnológico ejecuta una petición de referenciado de la instrucción "MC_Home" de Motion Control. (<TO>.StatusWord.HomingCommand)

Errores

La tabla siguiente muestra los posibles errores:

Error	Descripción
Sistema	Ha ocurrido un error interno del sistema. (<TO>.ErrorWord.SystemFault)
Configuración	Error de configuración Hay uno o varios parámetros de configuración incoherentes o inadmisibles. El objeto tecnológico se ha configurado erróneamente o bien se han modificado incorrectamente datos de configuración modificables durante el tiempo de ejecución del programa de usuario. (<TO>.ErrorWord.ConfigurationFault)
Programa de usuario	Error en una instrucción de Motion Control o en su utilización en el programa de usuario. (<TO>.ErrorWord.UserFault)
Encóder	Error en el sistema del encóder. (<TO>.ErrorWord.SensorFault)
Transferencia de datos	Falta la comunicación o es errónea. (<TO>.ErrorWord.CommunicationFault)

Acusar

Haga clic en el botón "Acusar" para acusar las alarmas y advertencias pendientes.

Información adicional

Encontrará una posibilidad de evaluar los diferentes bits de estado en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Estado del movimiento

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Estado movimiento" permite vigilar los valores del encóder en el TIA Portal. La función de diagnóstico está disponible en modo online.

Área "Valores actuales"

La tabla siguiente describe el significado de la información de estado:

Estado	Descripción
Posición real	Posición real del eje Si el objeto tecnológico no está referenciado, el valor se indica en relación a la posición que había en el momento de habilitar el objeto tecnológico. (<TO>.ActualPosition)
Velocidad real	Velocidad real del eje (<TO>.ActualVelocity)

Telegrama PROFIdrive

Descripción

La función de diagnóstico "Objeto tecnológico > Diagnóstico > Telegrama PROFIdrive" permite vigilar el telegrama PROFIdrive del encóder en el TIA Portal. La visualización de la función de diagnóstico está disponible en el modo online Objeto tecnológico.

Área "Encóder"

En esta área se muestran los parámetros siguientes del telegrama PROFIdrive del encóder al controlador:

- La palabra de estado "G1_ZSW"
- El valor real de posición "G1_XIST1" (valor real cíclico del encóder)
- El valor real de posición "G1_XIST2" (valor absoluto del encóder)

Anexo

Variables del objeto tecnológico Eje de posicionamiento

Leyenda

Variable	Nombre de la variable	
Tipo de datos	Tipo de datos de la variable	
Valores	Rango de valores de la variable- valores mínimo a máximo Sin una indicación de valores específica rigen los límites del rango de valores del tipo de datos correspondiente o la indicación bajo "Descripción".	
W	Efectividad de los cambios en el bloque de datos tecnológico	
	DIR	Directo: el valor cambia por asignación directa y es efectivo inmediatamente.

	RES	Restart: el valor inicial de la memoria de carga cambia debido a la instrucción avanzada "WRIT_DBL" (escribir en la memoria de carga del DB). Los cambios no son efectivos hasta que se reinicializa el objeto tecnológico.
	RON	Read only: no se puede o no se debe modificar la variable.
Descripción	Descripción de la variable	

L - indicación lineal R - indicación rotativa

El acceso a las variables se realiza por medio de "<TO>.<nombre de variable>". El comodín <TO> representa el nombre del objeto tecnológico.

Valores reales y consignas (eje de posicionamiento)

Las variables siguientes muestran las consignas y los valores reales del objeto tecnológico.

Variables

Leyenda (Página 10797)

Variable	Tipo de datos	Valores	W	Descripción
Position	LREAL	-	RON	Consigna de posición
Velocity	LREAL	-	RON	Consigna de velocidad / de velocidad de giro
ActualPosition	LREAL	-	RON	Posición real
ActualVelocity	LREAL	-	RON	Velocidad real
ActualSpeed	LREAL	-	RON	Velocidad de giro real del motor (si la consigna analógica = 0.0)
Acceleration	LREAL	-	RON	Aceleración de consigna
ActualAcceleration	LREAL	-	RON	Aceleración real

Variables Actor (eje de posicionamiento)

La estructura de variable <TO>.Actor.<nombre de variable> incluye la configuración del accionamiento dada por el controlador.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
Actor.		STRUCT			
	Type	DINT	-	RON	Conexión del accionamiento 1: Telegrama PROFIdrive 0: Salida analógica

Variable			Tipo de datos	Valores	W	Descripción
	InverseDirection		BOOL	-	RES	Inversión de la consigna FALSE: no TRUE: sí
	Interface.		STRUCT			
		AddressIn	UDINT	de 0 a 65535	RON	Dirección de entrada para el telegrama PROFIdrive
		AdressOut	UDINT	de 0 a 65535	RON	Dirección de salida para el telegrama PROFIdrive o la consigna analógica
		Telegram	UDINT	-	RON	Telegrama PROFIdrive del accionamiento 1: NSOLL (16 bits), NIST (16 bits) 2: NSOLL (32 bits), NIST (32 bits), señal de vida 3: NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida 5: NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida, DSC (NSOLL - consigna de velocidad de giro, NIST - valor real de velocidad de giro)
		EnableDriveOutput	BOOL	-	RON	"Salida de habilitación" para accionamientos analógicos FALSE: desactivada TRUE: activada
		EnableDriveOutputAddress	UDINT	de 0 a 65535	RON	Número de byte para la "salida de habilitación"
		EnableDriveOutputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la "salida de habilitación"
		DriveReadyInput	BOOL	-	RON	"Entrada de disponibilidad" para accionamientos analógicos El accionamiento analógico notifica su disponibilidad para recibir consignas de velocidad de giro. FALSE: desactivada TRUE: activada
		DriveReadyInputAddress	UDINT	de 0 a 65535	RON	Número de byte para la "entrada de disponibilidad"
		DriveReadyInputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la "entrada de disponibilidad"
	DriveParameter.					

Variable			Tipo de datos	Valores	W	Descripción
		ReferenceSpeed	LREAL	de 0.0 a 1.0E12	RES	Valor de referencia (100%) para la consigna de velocidad de giro del accionamiento (N-soll) La consigna de velocidad de giro se transfiere en el telegrama PROFIdrive como valor normalizado de -200 % a 200 % de "ReferenceSpeed". En la especificación de consigna mediante un valor analógico, la salida analógica puede utilizarse en un rango de -117 % a 117 %, siempre que el accionamiento lo permita.
		MaxSpeed	LREAL	de 0.0 a 1.0E12	RES	Valor máximo de la consigna de velocidad de giro del accionamiento (N-soll) (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Consigna analógica: MaxSpeed ≤ 1.17 × ReferenceSpeed)

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables Sensor[n] (eje de posicionamiento)

La estructura de variable <TO>.Sensor[n].<nombre de variable> incluye la configuración del encóder dada por el controlador y la configuración del referenciado activo y pasivo.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
Sensor[n].		ARRAY [1..1] OF STRUCT			
	Type	DINT	-	RES	Tipo de encóder 0: incremental 1: absoluto 2: cíclico absoluto
	InverseDirection	BOOL	-	RES	Inversión del valor real FALSE: no TRUE: sí
	System	DINT	-	RES	Sistema de encóder 0: encóder lineal 1: encóder rotativo

Variable		Tipo de datos	Valores	W	Descripción
MountingMode		DINT	-	RES	Modo de montaje del encóder 0: en el lado de accionamiento 1: en el lado de carga 2: externo
Interface.					
	AddressIn	UDINT	de 0 a 65535	RON	Dirección de entrada para el telegrama PROFIdrive
	AddressOut	UDINT	de 0 a 65535	RON	Dirección de salida para el telegrama PROFIdrive
	Telegram	UDINT	-	RON	Telegrama PROFIdrive del encóder 3: valores reales vía telegrama de accionamiento (NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida) 5: valores reales vía telegrama de accionamiento (NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida, DSC) 81: valores reales vía telegrama propio del encóder (valor real del encóder, señal de vida) 83: valores reales vía telegrama propio del encóder (NIST (32 bits), valor real del encóder, señal de vida) Para el objeto tecnológico Encóder Externo solo se admiten los telegramas 81 y 83. (NSOLL - consigna de velocidad de giro, NIST - valor real de velocidad de giro)
Parameter.					
	Resolution	LREAL	de 1.0E-12 a 1.0E12	RES	Resolución de un encóder lineal (distancia entre dos rayas del encóder)
	StepsPerRevolution	UDINT	de 1 a 8388608	RES	Incrementos por vuelta en un encóder rotativo
	FineResolutionXist1	UDINT	de 0 a 31	RES	Número de bits para la resolución fina Gn_XIST1 (valor real cíclico del encóder)
	FineResolutionXist2	UDINT	de 0 a 31	RES	Número de bits para la resolución fina Gn_XIST2 (valor absoluto del encóder)
	DeterminableRevolutions	UDINT	de 0 a 8388608	RES	Número de vueltas distinguibles del encóder en un encóder absoluto multivuelta (en encóders absolutos monovuelta = 1; en encóders incrementales = 0)
	DistancePerRevolution	LREAL	de 0.0 a 1.0E12	RES	Recorrido de la carga por vuelta en un encóder montado externamente
ActiveHoming.		STRUCT			
	Mode	DINT	-	RES	Modo de referenciado 0: solo marca de referencia 1: marca de referencia y detector de proximidad 2: entrada digital
	SideInput	BOOL	-	DIR	Lado de la entrada digital en el referenciado activo: FALSE: lado negativo TRUE: lado positivo

Variable			Tipo de datos	Valores	W	Descripción
		Direction	DINT	-	DIR	Sentido de referenciado / sentido de aproximación hasta la marca de referencia 1: sentido de referenciado positivo 2: sentido de referenciado negativo
		HomePositionOffset	LREAL	de -1.0E12 a 1.0E12	DIR	Decalaje del punto de referencia
		DigitalInputAddress	UDINT	de 0 a 65535	RON	Número de byte de la entrada digital
		DigitalInputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la entrada digital
	PassiveHoming.		STRUCT			
		Mode	DINT	-	RES	Modo de referenciado 0: solo marca de referencia 1: marca de referencia y detector de proximidad 2: entrada digital
		SideInput	BOOL	-	DIR	Lado de la entrada digital en el referenciado pasivo: FALSE: lado negativo TRUE: lado positivo
		Direction	DINT	-	DIR	Sentido de referenciado / sentido de aproximación hasta la marca de referencia 0: sentido de referenciado actual 1: sentido de referenciado positivo 2: sentido de referenciado negativo
		DigitalInputAddress	UDINT	de 0 a 65535	RON	Número de byte de la entrada digital
		DigitalInputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la entrada digital

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables LoadGear (eje de posicionamiento)

La estructura de variable <TO>.LoadGear.<nombre de variable> incluye la configuración del reductor de carga.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Rango de valores	W	Descripción
LoadGear.		STRUCT			
	Numerator	UDINT	de 1 a 4294967295	RES	Reductor de carga del contador
	Denominator	UDINT	de 1 a 4294967295	RES	Reductor de carga del denominador

Variables Mechanics (eje de posicionamiento)

La estructura de variable <TO>.Mechanics.<nombre de variable> incluye la configuración del sistema mecánico.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Rango de valores	W	Descripción
Mechanics.					
	LeadScrew	LREAL	de 0.0 a 1.0E12	RES	Paso del husillo

Variables Properties (eje de posicionamiento)

La estructura de variable <TO>.Properties.<nombre de variable> incluye la configuración del tipo de eje y movimiento.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Rango de valores	W	Descripción
Properties.					
	MotionType	DINT	-	RON	Visualización del tipo de eje o movimiento: 0: eje o movimiento lineal 1: eje o movimiento rotativo

Variables Modulo (eje de posicionamiento)

La estructura de variable <TO>.Modulo.<nombre de variable> incluye la configuración del módulo.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
Modulo.		STRUCT			
	Enable	BOOL	-	RES	FALSE: conversión de módulo desactivada TRUE: conversión de módulo activada Si la conversión de módulo está activada, se comprueba que la longitud del módulo > 0.0.
	Length	LREAL	de 0.001 a 1.0E12	RES	Longitud de módulo
	StartValue	LREAL	de -1.0E12 a 1.0E12	RES	Valor inicial de módulo

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables DynamicLimits (eje de posicionamiento)

La estructura de variable <TO>.DynamicLimits.<nombre de variable> incluye la configuración de los límites dinámicos. En el control de movimiento no se admiten valores dinámicos mayores que los límites dinámicos. Si en una instrucción de Motion Control se indican valores mayores, el desplazamiento se realiza con los límites dinámicos y se muestra una advertencia (alarma de 501 a 503 - los valores dinámicos se limitan).

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
DynamicLimits.		STRUCT			
	MaxVelocity	LREAL	de 0.0 a 1.0E12	RES	Velocidad máxima admisible del eje
	MaxAcceleration	LREAL	de 0.0 a 1.0E12	DIR	Aceleración máxima admisible del eje
	MaxDeceleration	LREAL	de 0.0 a 1.0E12	DIR	Deceleración máxima admisible del eje
	MaxJerk	LREAL	de 0.0 a 1.0E12	DIR	Tirón máximo admisible del eje

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables DynamicDefaults (eje de posicionamiento)

La estructura de variable <TO>.DynamicDefaults.<nombre de variable> incluye la configuración de los ajustes predeterminados de dinámica. Estos ajustes se utilizan cuando se indica un valor dinámico inferior a 0.0 en una instrucción de Motion Control (excepciones: MC_MoveJog.Velocity, MC_MoveVelocity.Velocity). Los cambios en los ajustes predeterminados de dinámica se aplican con el próximo flanco ascendente en el parámetro "Execute" de una instrucción de Motion Control.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
DynamicDefaults.		STRUCT			
	Velocity	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de velocidad
	Acceleration	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de aceleración
	Deceleration	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de deceleración
	Jerk	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de tirón
	EmergencyDeceleration	LREAL	de 0.0 a 1.0E12	DIR	Deceleración de parada de emergencia

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables PositionLimits_SW (eje de posicionamiento)

La estructura de variable <TO>.PositionLimits_SW.<nombre de variable> incluye la configuración de la vigilancia de posición con finales de carrera por software. Con finales de carrera por software se limita el área de trabajo de un eje de posicionamiento.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
PositionLimits_SW.		STRUCT			
	Active	BOOL	-	DIR	FALSE: vigilancia desactivada TRUE: vigilancia activada
	MinPosition	LREAL	de -1.0E12 a 1.0E12	DIR	Posición del final de carrera por software negativo
	MaxPosition	LREAL	de -1.0E12 a 1.0E12	DIR	Posición del final de carrera por software positivo ("MaxPosition" > "MinPosition")

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables PositionLimits_HW (eje de posicionamiento)

La estructura de variable <TO>.PositionLimits_HW.<nombre de variable> incluye la configuración de la vigilancia de posición con finales de carrera por hardware. Con finales de carrera por hardware se limita el rango de desplazamiento de un eje de posicionamiento.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
PositionLimits_HW.		STRUCT			
	Active	BOOL	-	RES	FALSE: vigilancia desactivada TRUE: vigilancia activada Con "Active" se activan o desactivan los dos finales de carrera por hardware (negativo y positivo).
	MinSwitchLevel	BOOL	-	RES	Selección de nivel para activar el final de carrera por hardware negativo: FALSE: nivel inferior (Low activo) TRUE: nivel superior (High activo)
	MinSwitchAddress	UDINT	de 0 a 65535	RON	Número de byte para el final de carrera por hardware negativo
	MinSwitchBitNumber	UDINT	de 0 a 7	RON	Número de bit para el final de carrera por hardware negativo
	MaxSwitchLevel	BOOL	-	RES	Selección de nivel para activar el final de carrera por hardware positivo: FALSE: nivel inferior (Low activo) TRUE: nivel superior (High activo)
	MaxSwitchAddress	UDINT	de 0 a 65535	RON	Número de byte para el final de carrera por hardware positivo
	MaxSwitchBitNumber	UDINT	de 0 a 7	RON	Número de bit para el final de carrera por hardware positivo

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables Homing (eje de posicionamiento)

La estructura de variable <TO>.Homing.<nombre de variable> incluye la configuración para el referenciado del objeto tecnológico.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
Homing.		STRUCT			
	AutoReversal	BOOL	-	RES	Inversión en los finales de carrera por hardware: FALSE: no TRUE: sí
	ApproachDirection	BOOL	-	DIR	Sentido de aproximación hasta el sensor del punto de referencia FALSE: sentido negativo TRUE: sentido positivo
	ApproachVelocity	LREAL	L: de 0.0 a 10000.0 mm/s R: de 0.0 a 360000.0°/s	DIR	Velocidad de aproximación Velocidad con referenciado activo a la que se dirige hacia el detector de proximidad y el punto de referencia.
	ReferencingVelocity	LREAL	L: de 0.0 a 1000.0 mm/s R: de 0.0 a 36000.0°/s	DIR	Velocidad de referenciado Velocidad con referenciado activo a la que se desplaza hasta la posición del punto de referencia.
	HomePosition	LREAL	de -1.0E12 a 1.0E12	DIR	Posición del punto de referencia

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables Override (eje de posicionamiento)

La estructura de variable <TO>.Override.<nombre de variable> incluye la configuración de los parámetros de corrección. Con parámetros de corrección se realiza una corrección porcentual de los valores predefinidos. Un cambio de corrección es efectivo inmediatamente y se extrae con los ajustes de dinámica efectivos en la instrucción de Motion Control.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
Override.		STRUCT			
	Velocity	LREAL	de 0.0 a 200.0 %	DIR	Corrección de la velocidad o de la velocidad de giro Corrección porcentual de la velocidad o de la velocidad de giro

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables PositionControl (eje de posicionamiento)

La estructura de variable <TO>.PositionControl.<nombre de variable> incluye los ajustes de la regulación de posición.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
PositionControl.					
	Kv	LREAL	de 0.0 a 2147480.0	DIR	Ganancia P de la regulación de posición ("Kv" > 0.0)
	Kpc	LREAL	de 0.0 a 150.0 %	DIR	Control anticipativo de velocidad de la regulación de posición Ajuste recomendado: • Conexión de accionamiento isócrona con PROFIdrive: 100.0% • Conexión de accionamiento no isócrona con PROFIdrive: de 0.0 a 100.0 % • Conexión de accionamiento analógica: de 0.0 a 100.0 %
	EnableDSC	BOOL	-	RES	Dynamic Servo Control (DSC) FALSE: DSC desactivado TRUE: DSC activado DSC solo posible si se usa el telegrama 5.

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables FollowingError (eje de posicionamiento)

La estructura de variable <TO>.FollowingError.<nombre de variable> incluye la configuración de la vigilancia dinámica de errores de seguimiento.

En caso de rebase por exceso del error de seguimiento admisible se emite la alarma tecnológica 521 y el objeto tecnológico se bloquea (reacción a alarma: retirar habilitación).

Cuando se alcanza el nivel de advertencia se emite una advertencia (alarma tecnológica 522).

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
FollowingError.		STRUCT			
	EnableMonitoring	BOOL	-	RES	FALSE: vigilancia de errores de seguimiento desactivada TRUE: vigilancia de errores de seguimiento activada
	MinValue	LREAL	de 0.0 a 1.0E12	DIR	Error de seguimiento admisible en velocidades inferiores al valor de "MinVelocity".
	MaxValue	LREAL	de 0.0 a 1.0E12	DIR	Error de seguimiento máximo admisible que puede alcanzarse con la velocidad máxima.
	MinVelocity	LREAL	de 0.0 a 1.0E12	DIR	"MinValue" está por debajo de esta velocidad y se mantiene constante.
	WarningLevel	LREAL	de 0.0 a 100.0	DIR	Nivel de advertencia Valor porcentual en relación con el error de seguimiento máximo admisible.

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables PositioningMonitoring (eje de posicionamiento)

La estructura de variable <TO>.PositioningMonitoring.<nombre de variable> incluye la configuración de la vigilancia de posicionamiento al final de un movimiento de posicionamiento.

Si, al final de un movimiento de posicionamiento, el valor real de posición alcanza la ventana de posicionamiento dentro del tiempo de tolerancia y permanece en dicha ventana durante el tiempo mínimo de permanencia, se activa <TO>.StatusWord.Done en el bloque de datos tecnológico. Con ello ha concluido una petición de movimiento.

Si se rebasa por exceso el tiempo de tolerancia se muestra la alarma tecnológica 541 "Vigilancia de posicionamiento" con el valor adicional 1: "Rango de destino no alcanzado dentro del tiempo de tolerancia de posicionamiento".

Si se rebasa por defecto el tiempo mínimo de permanencia se muestra la alarma tecnológica 541 "Vigilancia de posicionamiento" con el valor adicional 2: "Rango de destino abandonado dentro del tiempo mínimo de permanencia".

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
PositioningMonitoring.		STRUCT			
	ToleranceTime	LREAL	de 0.0 a 1.0E12	DIR	Tiempo de tolerancia Duración máxima permitida desde que se alcanza la consigna de velocidad cero hasta que se entra en la ventana de posicionamiento.
	MinDwellTime	LREAL	de 0.0 a 1.0E12	DIR	Tiempo mínimo de permanencia en la ventana de posicionamiento
	Window	LREAL	de 0.0 a 1.0E12	DIR	Ventana de posicionamiento

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables StandstillSignal (eje de posicionamiento)

La estructura de variable <TO>.StandstillSignal.<nombre de variable> incluye la configuración de la señal de velocidad cero.

Si el valor real de velocidad rebasa por defecto el umbral de velocidad y no lo rebasa por exceso durante el tiempo mínimo de permanencia, se activa la señal de velocidad cero <TO>.StatusWord.Standstill.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
StandstillSignal.		STRUCT			Configuración de la señal de velocidad cero
	VelocityThreshold	LREAL	de 0.0 a 1.0E12	DIR	Umbral de velocidad Si se rebasa por defecto empieza el tiempo mínimo de permanencia.
	MinDwellTime	LREAL	de 0.0 a 1.0E12	DIR	Tiempo mínimo de permanencia

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables StatusPositioning (eje de posicionamiento)

La estructura de variable <TO>.StatusPositioning.<nombre de variable> muestra el estado de un movimiento de posicionamiento.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
StatusPositioning.		STRUCT			
	Distance	LREAL	-	RON	Distancia respecto de la posición de destino
	TargetPosition	LREAL	-	RON	Posición de destino
	FollowingError	LREAL	-	RON	Error de seguimiento actual

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables StatusDrive (eje de posicionamiento)

La estructura de variable <TO>.StatusDrive.<nombre de variable> muestra el estado del accionamiento.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
StatusDrive.		STRUCT			
	InOperation	BOOL	-	RON	Estado de operación del accionamiento FALSE: accionamiento no preparado. Las consignas no se ejecutan. TRUE: accionamiento listo. Las consignas pueden ejecutarse.
	CommunicationOK	BOOL	-	RON	Comunicación BUS cíclica entre controlador y accionamiento FALSE: no establecida TRUE: establecida
	Error	BOOL	-	RON	FALSE: no hay errores en el accionamiento TRUE: error en el accionamiento

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables StatusSensor[n] (eje de posicionamiento)

La estructura de variable <TO>.StatusSensor[n].<nombre de variable> muestra el sistema de medida.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
StatusSensor[n].		ARRAY [1..1] OF STRUCT			
	State	DINT	-	RON	Estado del valor real del encóder: 0: "NOT_VALID" (no válido) 1: "Waiting_FOR_VALID" (esperando estado válido) 2: "VALID" (válido)
	CommunicationOK	BOOL	-	RON	Comunicación BUS cíclica entre controlador y encóder FALSE: no establecida TRUE: establecida
	Error	BOOL	-	RON	FALSE: no hay errores en el sistema de medida TRUE: error en el sistema de medida
	AbsEncoderOffset	LREAL	-	RON	Decalaje del punto de referencia hasta el valor de un encóder absoluto. El valor se guarda de forma remanente en la CPU.

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variable StatusWord (eje de posicionamiento)

La variable <TO>.StatusWord incluye la información de estado del objeto tecnológico.

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 5 "HomingDone") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10797)

Variable	Tipo de datos	Valores	W	Descripción
Status Word	DWORD	-	RON	Información de estado del objeto tecnológico
Bit 0	-	-	-	"Enable" Estado de habilitación 0: objeto tecnológico bloqueado 1: objeto tecnológico habilitado
Bit 1	-	-	-	"Error" 0: no hay errores 1: hay errores
Bit 2	-	-	-	"RestartActive" 0: ningún "Restart" activo 1: "Restart" activo. El objeto tecnológico se reinicializa.
Bit 3	-	-	-	"OnlineStartValuesChanged" 0: "variable "Restart" sin cambios 1: cambios en la variable "Restart". Para aplicar los cambios es necesario reinicializar el objeto tecnológico.
Bit 4	-	-	-	"ControlPanelActive" Panel de mando del eje 0: desactivado 1: activado
Bit 5	-	-	-	"HomingDone" Estado de referenciado 0: objeto tecnológico no referenciado 1: objeto tecnológico referenciado
Bit 6	-	-	-	"Done" 0: petición de movimiento en proceso o panel de mando del eje activado 1: no hay petición de movimiento en proceso y el panel de mando del eje está desactivado
Bit 7	-	-	-	"Standstill" Señal de velocidad cero 0: eje en movimiento 1: eje parado
Bit 8	-	-	-	"PositioningCommand" 0: ninguna petición de posicionamiento activa 1: petición de posicionamiento activa ("MC_MoveReleative", "MC_MoveAbsolute")
Bit 9	-	-	-	"JogCommand" 0: ninguna petición "MC_MoveJog" activa 1: petición "MC_MoveJog" activa
Bit 10	-	-	-	"VelocityCommand" 0: ninguna petición "MC_MoveVelocity" activa 1: "petición "MC_MoveVelocity" activa

Variable	Tipo de datos	Valores	W	Descripción
Bit 11	-	-	-	"HomingCommand" 0: ninguna petición "MC_Home" en procesamiento 1: "petición "MC_Home" en procesamiento
Bit 12	-	-	-	"ConstantVelocity" 0: el eje se acelera o decelera. 1: consigna de velocidad alcanzada. El eje avanza a esta velocidad constante o está parado.
Bit 13	-	-	-	"Accelerating" 0: ningún proceso de aceleración activo 1: proceso de aceleración activo
Bit 14	-	-	-	"Decelerating" 0: ningún proceso de deceleración activo 1: proceso de deceleración activo
Bit 15	-	-	-	"SWLimitMinActive" 0: final de carrera por software negativo no aproximado 1: final de carrera por software negativo aproximado o sobrepasado
Bit 16	-	-	-	"SWLimitMaxActive" 0: final de carrera por software positivo no aproximado 1: final de carrera por software positivo aproximado o sobrepasado
Bit 17	-	-	-	"HWLimitMinActive" 0: final de carrera por hardware negativo no aproximado 1: final de carrera por hardware negativo aproximado o sobrepasado
Bit 18	-	-	-	"HWLimitMaxActive" 0: final de carrera por hardware positivo no aproximado 1: final de carrera por hardware positivo aproximado o sobrepasado
Bit 19	-	-	-	"ErrorStopActive" Parada como consecuencia de una reacción a alarma 0: inactiva 1: activa
Bit 20	-	-	-	"MC_PowerStopActive" Parada como consecuencia de un bloqueo mediante "MC_Power" 0: inactiva 1: activa
Bit 21... Bit 31	-	-	-	Reservado

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variable ErrorWord (eje de posicionamiento)

La variable <TO>.ErrorWord señala errores en el objeto tecnológico (alarmas tecnológicas).

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 3 "CommandNotAccepted") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10797)

Variable	Tipo de datos	Valores	W	Descripción
Error Word	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Error de sistema
Bit 1	-	-	-	"ConfigurationFault" Error de configuración Hay uno o varios parámetros de configuración incoherentes o inadmisibles.
Bit 2	-	-	-	"UserFault" Error en una instrucción de Motion Control o en su utilización en el programa de usuario
Bit 3	-	-	-	"CommandNotAccepted" Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios.
Bit 4	-	-	-	"DriveFault" Error en el accionamiento
Bit 5	-	-	-	"SensorFault" Error en el sistema del encóder
Bit 6	-	-	-	"DynamicError" Las especificaciones de valores dinámicos se limitan a valores fiables.
Bit 7	-	-	-	"CommunicationFault" Error de comunicación Falta la comunicación o es errónea.
Bit 8	-	-	-	"SWLimit" Final de carrera por software alcanzado o sobrepasado.
Bit 9	-	-	-	"HWLimit" Final de carrera por hardware alcanzado o sobrepasado.
Bit 10	-	-	-	"HomingFault" Error en el proceso de referenciado No es posible finalizar el referenciado.
Bit 11	-	-	-	"FollowingErrorFault" Límites de error de seguimiento rebasados por exceso
Bit 12	-	-	-	"PositioningFault" Error de posicionamiento

Variable	Tipo de datos	Valores	W	Descripción
Bit 13	-	-	-	"PeripheralError" Error al acceder a una dirección lógica
Bit 14... Bit 31	-	-	-	Reservado

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables ErrorDetail (eje de posicionamiento)

La estructura de variable <TO>.ErrorDetail.<nombre de variable> incluye el número de alarma y la reacción local efectiva a la alarma tecnológica actualmente pendiente en el objeto tecnológico.

Encontrará una lista de las alarmas tecnológicas y las reacciones a alarma en el anexo Alarmas tecnológicas (Página 10841).

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
ErrorDetail.		STRUCT			
	Number	UDINT	-	RON	Número de alarma
	Reaction	DINT	de 0 a 4	RON	Reacción efectiva a alarma 0: sin reacción 1: parada con valores dinámicos actuales 2: parada con valores dinámicos máximos 3: parada con rampa de parada de emergencia 4: retirar habilitación En el objeto tecnológico Encóder externo solo se admiten las reacciones a alarma 0 y 4.

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Alarmas tecnológicas (Página 10782)

Variable WarningWord (eje de posicionamiento)

La variable <TO>.WarningWord indica las advertencias pendientes en el objeto tecnológico.

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 11 "FollowingErrorWarning") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10797)

Variable	Tipo de datos	Valores	W	Descripción
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	Reservado
Bit 1	-	-	-	"ConfigurationFault" Error de configuración Uno o varios parámetros de configuración se adaptan internamente.
Bit 2	-	-	-	Reservado
Bit 3	-	-	-	"CommandNotAccepted" Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios.
Bit 4	-	-	-	Reservado
Bit 5	-	-	-	Reservado
Bit 6	-	-	-	"DynamicError" Las especificaciones de valores dinámicos se limitan a valores fiables.
Bit 7	-	-	-	Reservado
Bit 8	-	-	-	Reservado
Bit 9	-	-	-	Reservado
Bit 10	-	-	-	Reservado
Bit 11	-	-	-	"FollowingErrorWarning" Nivel de advertencia alcanzado / rebasado por exceso para la vigilancia de errores de seguimiento
Bit 12	-	-	-	Reservado
Bit 13	-	-	-	Reservado
Bit 14... Bit 31	-	-	-	Reservado

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables ControlPanel (eje de posicionamiento)

La estructura de variable <TO>.ControlPanel.<nombre de variable> no incluye datos relevantes para el usuario. Esta estructura de variable se utiliza internamente.

Una excepción la constituye la variable <TO>.ControlPanel.Input.TimeOut para adaptar la vigilancia de señal de vida.

Variables

Leyenda (Página 10797)

Variable				Tipo de datos	Valores	W	Descripción
ControlPanel.				STRUCT			
	Input.			STRUCT			
		Command.		ARRAY [1..2] OF STRUCT			
			ReqCounter	UDINT	-	DIR	-
			Type	UDINT	-	DIR	-
			Position	LREAL	-	DIR	-
			Velocity	LREAL	-	DIR	-
			Acceleration	LREAL	-	DIR	-
			Deceleration	LREAL	-	DIR	-
			Jerk	LREAL	-	DIR	-
			Param	LREAL	-	DIR	-
		TimeOut		LREAL	-	DIR	Vigilancia de señal de vida Valor predeterminado: 2000 ms
		EsLifeSign		UDINT	-	DIR	-
	Output.			STRUCT			
		Command.		ARRAY [1..2] OF STRUCT			
			AckCounter	UDINT	-	RON	-
			Error	BOOL	-	RON	-
			ErrorID	UDINT	-	RON	-
			Done	BOOL	-	RON	-
			Aborted	BOOL	-	RON	-
		RTLifeSign		UDINT	-	RON	-

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables InternalTOTrace (eje de posicionamiento)

La estructura de variable <TO>.InternalTOTrace.<nombre de variable> no incluye datos relevantes para el usuario. Esta estructura de variable se utiliza internamente.

Variables

Leyenda (Página 10797)

Variable		Tipo de datos	Valores	W	Descripción
InternalTOTrace.		ARRAY [1..4] OF STRUCT			
	Id	DINT	-	DIR	-
	Value	LREAL	-	RON	-

Consulte también

Evaluación del bloque de datos tecnológico (Página 10747)

Variables del objeto tecnológico Eje de velocidad de giro

Leyenda

Variable	Nombre de la variable	
Tipo de datos	Tipo de datos de la variable	
Valores	Rango de valores de la variable- valores mínimo a máximo Sin una indicación de valores específica rigen los límites del rango de valores del tipo de datos correspondiente o la indicación bajo "Descripción".	
W	Efectividad de los cambios en el bloque de datos tecnológico	
	DIR	Directo: el valor cambia por asignación directa y es efectivo inmediatamente.
	RES	Restart: el valor inicial de la memoria de carga cambia debido a la instrucción avanzada "WRIT_DBL" (escribir en la memoria de carga del DB). Los cambios no son efectivos hasta que se reinicializa el objeto tecnológico.
	RON	Read only: no se puede o no se debe modificar la variable.
Descripción	Descripción de la variable	

L - indicación lineal R - indicación rotativa

El acceso a las variables se realiza por medio de "<TO>.<nombre de variable>". El comodín <TO> representa el nombre del objeto tecnológico.

Valores reales y consignas (eje de velocidad de giro)

Las variables siguientes muestran las consignas y los valores reales del objeto tecnológico.

Variables

Leyenda (Página 10819)

Variable	Tipo de datos	Valores	W	Descripción
Velocity	LREAL	-	RON	Consigna de velocidad / de velocidad de giro
ActualSpeed	LREAL	-	RON	Velocidad de giro real del motor (si la consigna analógica = 0.0)
Acceleration	LREAL	-	RON	Aceleración de consigna

Variables Actor (eje de velocidad de giro)

La estructura de variable <TO>.Actor.<nombre de variable> incluye la configuración del accionamiento dada por el controlador.

Variables

Leyenda (Página 10819)

Variable			Tipo de datos	Valores	W	Descripción
Actor.			STRUCT			
	Type		DINT	-	RON	Conexión del accionamiento 1: Telegrama PROFIdrive 0: Salida analógica
	InverseDirection		BOOL	-	RES	Inversión de la consigna FALSE: no TRUE: sí
	Interface.		STRUCT			
		AddressIn	UDINT	de 0 a 65535	RON	Dirección de entrada para el telegrama PROFIdrive
		AdressOut	UDINT	de 0 a 65535	RON	Dirección de salida para el telegrama PROFIdrive o la consigna analógica
		Telegram	UDINT	-	RON	Telegrama PROFIdrive del accionamiento 1: NSOLL (16 bits), NIST (16 bits) 2: NSOLL (32 bits), NIST (32 bits), señal de vida 3: NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida 5: NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida, DSC (NSOLL - consigna de velocidad de giro, NIST - valor real de velocidad de giro)
		EnableDriveOutput	BOOL	-	RON	"Salida de habilitación" para accionamientos analógicos FALSE: desactivada TRUE: activada
		EnableDriveOutput Address	UDINT	de 0 a 65535	RON	Número de byte para la "salida de habilitación"

Variable			Tipo de datos	Valores	W	Descripción
		EnableDriveOutputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la "salida de habilitación"
		DriveReadyInput	BOOL	-	RON	"Entrada de disponibilidad" para accionamientos analógicos El accionamiento analógico notifica su disponibilidad para recibir consignas de velocidad de giro. FALSE: desactivada TRUE: activada
		DriveReadyInputAddress	UDINT	de 0 a 65535	RON	Número de byte para la "entrada de disponibilidad"
		DriveReadyInputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la "entrada de disponibilidad"
	DriveParameter.					
		ReferenceSpeed	LREAL	de 0.0 a 1.0E12	RES	Valor de referencia (100%) para la consigna de velocidad de giro del accionamiento (N-soll) La consigna de velocidad de giro se transfiere en el telegrama PROFIdrive como valor normalizado de -200 % a 200 % de "ReferenceSpeed". En la especificación de consigna mediante un valor analógico, la salida analógica puede utilizarse en un rango de -117 % a 117 %, siempre que el accionamiento lo permita.
		MaxSpeed	LREAL	de 0.0 a 1.0E12	RES	Valor máximo de la consigna de velocidad de giro del accionamiento (N-soll) (PROFIdrive: MaxSpeed ≤ 2 × ReferenceSpeed Consigna analógica: MaxSpeed ≤ 1.17 × ReferenceSpeed)

Variables LoadGear (eje de velocidad de giro)

La estructura de variable <TO>.LoadGear.<nombre de variable> incluye la configuración del reductor de carga.

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Rango de valores	W	Descripción
LoadGear.		STRUCT			
	Numerator	UDINT	de 1 a 4294967295	RES	Reductor de carga del contador
	Denominator	UDINT	de 1 a 4294967295	RES	Reductor de carga del denominador

Variables DynamicLimits (eje de velocidad de giro)

La estructura de variable <TO>.DynamicLimits.<nombre de variable> incluye la configuración de los límites dinámicos. En el control de movimiento no se admiten valores dinámicos mayores que los límites dinámicos. Si en una instrucción de Motion Control se indican valores mayores, el desplazamiento se realiza con los límites dinámicos y se muestra una advertencia (alarma de 501 a 503 - los valores dinámicos se limitan).

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Valores	W	Descripción
DynamicLimits.		STRUC			
	MaxVelocity	LREAL	de 0.0 a 1.0E12	RES	Velocidad máxima admisible del eje
	MaxAcceleration	LREAL	de 0.0 a 1.0E12	DIR	Aceleración máxima admisible del eje
	MaxDeceleration	LREAL	de 0.0 a 1.0E12	DIR	Deceleración máxima admisible del eje
	MaxJerk	LREAL	de 0.0 a 1.0E12	DIR	Tirón máximo admisible del eje

Variables DynamicDefaults (eje de velocidad de giro)

La estructura de variable <TO>.DynamicDefaults.<nombre de variable> incluye la configuración de los ajustes predeterminados de dinámica. Estos ajustes se utilizan cuando se indica un valor dinámico inferior a 0.0 en una instrucción de Motion Control (excepciones: MC_MoveJog.Velocity, MC_MoveVelocity.Velocity). Los cambios en los ajustes predeterminados de dinámica se aplican con el próximo flanco ascendente en el parámetro "Execute" de una instrucción de Motion Control.

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Valores	W	Descripción
DynamicDefaults.		STRUCT			
	Velocity	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de velocidad
	Acceleration	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de aceleración
	Deceleration	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de deceleración
	Jerk	LREAL	de 0.0 a 1.0E12	DIR	Preajuste de tirón
	EmergencyDeceleration	LREAL	de 0.0 a 1.0E12	DIR	Deceleración de parada de emergencia

Variables Override (eje de velocidad de giro)

La estructura de variable <TO>.Override.<nombre de variable> incluye la configuración de los parámetros de corrección. Con parámetros de corrección se realiza una corrección porcentual

de los valores predefinidos. Un cambio de corrección es efectivo inmediatamente y se extrae con los ajustes de dinámica efectivos en la instrucción de Motion Control.

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Valores	W	Descripción
Override.		STRUCT			
	Velocity	LREAL	de 0.0 a 200.0 %	DIR	Corrección de la velocidad o de la velocidad de giro Corrección porcentual de la velocidad o de la velocidad de giro

Variables StatusDrive (eje de velocidad de giro)

La estructura de variable <TO>.StatusDrive.<nombre de variable> muestra el estado del accionamiento.

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Valores	W	Descripción
StatusDrive.		STRUCT			
	InOperation	BOOL	-	RON	Estado de operación del accionamiento FALSE: accionamiento no preparado. Las consignas no se ejecutan. TRUE: accionamiento listo. Las consignas pueden ejecutarse.
	CommunicationOK	BOOL	-	RON	Comunicación BUS cíclica entre controlador y accionamiento FALSE: no establecida TRUE: establecida
	Error	BOOL	-	RON	FALSE: no hay errores en el accionamiento TRUE: error en el accionamiento

Variable StatusWord (eje de velocidad de giro)

La variable <TO>.StatusWord incluye la información de estado del objeto tecnológico.

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 5 "HomingDone") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10819)

Variable	Tipo de datos	Valores	W	Descripción
Status Word	DWORD	-	RON	Información de estado del objeto tecnológico
Bit 0	-	-	-	"Enable" Estado de habilitación 0: objeto tecnológico bloqueado 1: objeto tecnológico habilitado
Bit 1	-	-	-	"Error" 0: no hay errores 1: hay errores
Bit 2	-	-	-	"RestartActive" 0: ningún "Restart" activo 1: "Restart" activo. El objeto tecnológico se reinicializa.
Bit 3	-	-	-	"OnlineStartValuesChanged" 0: "variable "Restart" sin cambios 1: cambios en la variable "Restart". Para aplicar los cambios es necesario reinicializar el objeto tecnológico.
Bit 4	-	-	-	"ControlPanelActive" Panel de mando del eje 0: desactivado 1: activado
Bit 5	-	-	-	Reservado
Bit 6	-	-	-	"Done" 0: petición de movimiento en proceso o panel de mando del eje activado 1: no hay petición de movimiento en proceso y el panel de mando del eje está desactivado
Bit 7	-	-	-	Reservado
Bit 8	-	-	-	Reservado
Bit 9	-	-	-	"JogCommand" 0: ninguna petición "MC_MoveJog" activa 1: petición "MC_MoveJog" activa
Bit 10	-	-	-	"VelocityCommand" 0: ninguna petición "MC_MoveVelocity" activa 1: "petición "MC_MoveVelocity" activa
Bit 11	-	-	-	Reservado
Bit 12	-	-	-	"ConstantVelocity" 0: el eje se acelera o decelera. 1: consigna de velocidad alcanzada. El eje avanza a esta velocidad constante o está parado.
Bit 13	-	-	-	"Accelerating" 0: ningún proceso de aceleración activo 1: proceso de aceleración activo

Variable	Tipo de datos	Valores	W	Descripción
Bit 14	-	-	-	"Decelerating" 0: ningún proceso de deceleración activo 1: proceso de deceleración activo
Bit 15	-	-	-	Reservado
Bit 16	-	-	-	Reservado
Bit 17	-	-	-	Reservado
Bit 18	-	-	-	Reservado
Bit 19	-	-	-	"ErrorStopActive" Parada como consecuencia de una reacción a alarma 0: inactiva 1: activa
Bit 20	-	-	-	"MC_PowerStopActive" Parada como consecuencia de un bloqueo mediante "MC_Power" 0: inactiva 1: activa
Bit 21... Bit 31	-	-	-	Reservado

Variable ErrorWord (eje de velocidad de giro)

La variable <TO>.ErrorWord señala errores en el objeto tecnológico (alarmas tecnológicas).

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 3 "CommandNotAccepted") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10819)

Variable	Tipo de datos	Valores	W	Descripción
Error Word	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Error de sistema
Bit 1	-	-	-	"ConfigurationFault" Error de configuración Hay uno o varios parámetros de configuración incoherentes o inadmisibles.
Bit 2	-	-	-	"UserFault" Error en una instrucción de Motion Control o en su utilización en el programa de usuario

Variable	Tipo de datos	Valores	W	Descripción
Bit 3	-	-	-	"CommandNotAccepted" Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios.
Bit 4	-	-	-	"DriveFault" Error en el accionamiento
Bit 5	-	-	-	Reservado
Bit 6	-	-	-	"DynamicError" Las especificaciones de valores dinámicos se limitan a valores fiables.
Bit 7	-	-	-	"CommunicationFault" Error de comunicación Falta la comunicación o es errónea.
Bit 8	-	-	-	Reservado
Bit 9	-	-	-	Reservado
Bit 10	-	-	-	Reservado
Bit 11	-	-	-	Reservado
Bit 12	-	-	-	Reservado
Bit 13	-	-	-	"PeripheralError" Error al acceder a una dirección lógica
Bit 14... Bit 31	-	-	-	Reservado

Variables ErrorDetail (eje de velocidad de giro)

La estructura de variable <TO>.ErrorDetail.<nombre de variable> incluye el número de alarma y la reacción local efectiva a la alarma tecnológica actualmente pendiente en el objeto tecnológico.

Encontrará una lista de las alarmas tecnológicas y las reacciones a alarma en el anexo Alarmas tecnológicas (Página 10841).

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Valores	W	Descripción
ErrorDetail.		STRUCT			
	Number	UDINT	-	RON	Número de alarma
	Reaction	DINT	de 0 a 4	RON	Reacción efectiva a alarma 0: sin reacción 1: parada con valores dinámicos actuales 2: parada con valores dinámicos máximos 3: parada con rampa de parada de emergencia 4: retirar habilitación En el objeto tecnológico Encóder externo solo se admiten las reacciones a alarma 0 y 4.

Variable WarningWord (eje de velocidad de giro)

La variable <TO>.WarningWord indica las advertencias pendientes en el objeto tecnológico.

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 11 "FollowingErrorWarning") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10819)

Variable	Tipo de datos	Valores	W	Descripción
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	Reservado
Bit 1	-	-	-	"ConfigurationFault" Error de configuración Uno o varios parámetros de configuración se adaptan internamente.
Bit 2	-	-	-	Reservado
Bit 3	-	-	-	"CommandNotAccepted" Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios.
Bit 4	-	-	-	Reservado
Bit 5	-	-	-	Reservado
Bit 6	-	-	-	"DynamicError" Las especificaciones de valores dinámicos se limitan a valores fiables.
Bit 7	-	-	-	Reservado

Variable	Tipo de datos	Valores	W	Descripción
Bit 8	-	-	-	Reservado
Bit 9	-	-	-	Reservado
Bit 10	-	-	-	Reservado
Bit 11	-	-	-	Reservado
Bit 12	-	-	-	Reservado
Bit 13	-	-	-	Reservado
Bit 14... Bit 31	-	-	-	Reservado

Variables ControlPanel (eje de velocidad de giro)

La estructura de variable <TO>.ControlPanel.<nombre de variable> no incluye datos relevantes para el usuario. Esta estructura de variable se utiliza internamente.

Una excepción la constituye la variable <TO>.ControlPanel.Input.TimeOut para adaptar la vigilancia de señal de vida.

Variables

Leyenda (Página 10819)

Variable	Tipo de datos	Valores	W	Descripción
ControlPanel.	STRUCT			
Input.	STRUCT			
Command.	ARRAY [1..2] OF STRUCT			
ReqCounter	UDINT	-	DIR	-
Type	UDINT	-	DIR	-
Position	LREAL	-	DIR	-
Velocity	LREAL	-	DIR	-
Acceleration	LREAL	-	DIR	-
Deceleration	LREAL	-	DIR	-
Jerk	LREAL	-	DIR	-
Param	LREAL	-	DIR	-
TimeOut	LREAL	-	DIR	Vigilancia de señal de vida Valor predeterminado: 2000 ms
EsLifeSign	UDINT	-	DIR	-
Output.	STRUCT			
Command.	ARRAY [1..2] OF STRUCT			
AckCounter	UDINT	-	RON	-

Variable				Tipo de datos	Valores	W	Descripción
			Error	BOOL	-	RON	-
			ErrorID	UDINT	-	RON	-
			Done	BOOL	-	RON	-
			Aborted	BOOL	-	RON	-
		RTLifeSign		UDINT	-	RON	-

Variables InternalTOTrace (eje de velocidad de giro)

La estructura de variable <TO>.InternalTOTrace.<nombre de variable> no incluye datos relevantes para el usuario. Esta estructura de variable se utiliza internamente.

Variables

Leyenda (Página 10819)

Variable		Tipo de datos	Valores	W	Descripción
InternalTOTrace.		ARRAY [1..4] OF STRUCT			
	Id	DINT	-	DIR	-
	Value	LREAL	-	RON	-

Variables del objeto tecnológico Encóder externo

Leyenda

Variable	Nombre de la variable	
Tipo de datos	Tipo de datos de la variable	
Valores	Rango de valores de la variable- valores mínimo a máximo Sin una indicación de valores específica rigen los límites del rango de valores del tipo de datos correspondiente o la indicación bajo "Descripción".	
W	Efectividad de los cambios en el bloque de datos tecnológico	
	DIR	Directo: el valor cambia por asignación directa y es efectivo inmediatamente.
	RES	Restart: el valor inicial de la memoria de carga cambia debido a la instrucción avanzada "WRIT_DBL" (escribir en la memoria de carga del DB). Los cambios no son efectivos hasta que se reinicializa el objeto tecnológico.
	RON	Read only: no se puede o no se debe modificar la variable.
Descripción	Descripción de la variable	

L - indicación lineal R - indicación rotativa

El acceso a las variables se realiza por medio de "<TO>.<nombre de variable>". El comodín <TO> representa el nombre del objeto tecnológico.

Valores reales y consignas (encóder externo)

Las variables siguientes muestran las consignas y los valores reales del objeto tecnológico.

Variables

Leyenda (Página 10829)

Variable	Tipo de datos	Valores	W	Descripción
ActualPosition	LREAL	-	RON	Posición real
ActualVelocity	LREAL	-	RON	Velocidad real
ActualAcceleration	LREAL	-	RON	Aceleración real

Variables Sensor (encóder externo)

La estructura de variable <TO>.Sensor.<nombre de variable> incluye la configuración del encóder dada por el controlador y la configuración del referenciado activo y pasivo.

Variables

Leyenda (Página 10829)

Variable			Tipo de datos	Valores	W	Descripción
Sensor.			STRUCT			
	Type		DINT	-	RES	Tipo de encóder 0: incremental 1: absoluto 2: cíclico absoluto
	InverseDirection		BOOL	-	RES	Inversión del valor real FALSE: no TRUE: sí
	System		DINT	-	RES	Sistema de encóder 0: encóder lineal 1: encóder rotativo
	MountingMode		DINT	-	RES	Modo de montaje del encóder 0: en el lado de accionamiento 1: en el lado de carga 2: externo
	Interface.					
		AddressIn	UDINT	de 0 a 65535	RON	Dirección de entrada para el telegrama PROFIdrive
		AddressOut	UDINT	de 0 a 65535	RON	Dirección de salida para el telegrama PROFIdrive

Variable			Tipo de datos	Valores	W	Descripción
		Telegram	UDINT	-	RON	Telegrama PROFIdrive del encóder 3: valores reales vía telegrama de accionamiento (NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida) 5: valores reales vía telegrama de accionamiento (NSOLL (32 bits), NIST (32 bits), valor real del encóder, señal de vida, DSC) 81: valores reales vía telegrama propio del encóder (valor real del encóder, señal de vida) 83: valores reales vía telegrama propio del encóder (NIST (32 bits), valor real del encóder, señal de vida) Para el objeto tecnológico Encóder Externo solo se admiten los telegramas 81 y 83. (NSOLL - consigna de velocidad de giro, NIST - valor real de velocidad de giro)
	Parameter.					
		Resolution	LREAL	de 1.0E-12 a 1.0E12	RES	Resolución de un encóder lineal (distancia entre dos rayas del encóder)
		StepsPerRevolution	UDINT	de 1 a 8388608	RES	Incrementos por vuelta en un encóder rotativo
		FineResolutionXist1	UDINT	de 0 a 31	RES	Número de bits para la resolución fina Gn_XIST1 (valor real cíclico del encóder)
		FineResolutionXist2	UDINT	de 0 a 31	RES	Número de bits para la resolución fina Gn_XIST2 (valor absoluto del encóder)
		DeterminableRevolutions	UDINT	de 0 a 8388608	RES	Número de vueltas distinguibles del encóder en un encóder absoluto multivuelta (en encóders absolutos monovuelta = 1; en encóders incrementales = 0)
		DistancePerRevolution	LREAL	de 0.0 a 1.0E12	RES	Recorrido de la carga por vuelta en un encóder montado externamente
	PassiveHoming.		STRUCT			
		Mode	DINT	-	RES	Modo de referenciado 0: solo marca de referencia 1: marca de referencia y detector de proximidad 2: entrada digital
		SideInput	BOOL	-	DIR	Lado de la entrada digital en el referenciado pasivo: FALSE: lado negativo TRUE: lado positivo
		Direction	DINT	-	DIR	Sentido de referenciado / sentido de aproximación hasta la marca de referencia 0: sentido de referenciado actual 1: sentido de referenciado positivo 2: sentido de referenciado negativo
		DigitalInputAddress	UDINT	de 0 a 65535	RON	Número de byte de la entrada digital
		DigitalInputBitNumber	UDINT	de 0 a 7	RON	Número de bit de la entrada digital

Variables Mechanics (encóder externo)

La estructura de variable <TO>.Mechanics.<nombre de variable> incluye la configuración del sistema mecánico.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Rango de valores	W	Descripción
Mechanics.					
	LeadScrew	LREAL	de 0.0 a 1.0E12	RES	Paso del husillo

Variables LoadGear (encóder externo)

La estructura de variable <TO>.LoadGear.<nombre de variable> incluye la configuración del reductor de carga.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Rango de valores	W	Descripción
LoadGear.		STRUCT			
	Numerator	UDINT	de 1 a 4294967295	RES	Reductor de carga del contador
	Denominator	UDINT	de 1 a 4294967295	RES	Reductor de carga del denominador

Variables Properties (encóder externo)

La estructura de variable <TO>.Properties.<nombre de variable> incluye la configuración del tipo de eje y movimiento.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Rango de valores	W	Descripción
Properties.					
	MotionType	DINT	-	RON	Visualización del tipo de eje o movimiento: 0: eje o movimiento lineal 1: eje o movimiento rotativo

Variables Modulo (encóder externo)

La estructura de variable <TO>.Modulo.<nombre de variable> incluye la configuración del módulo.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Valores	W	Descripción
Modulo.		STRUCT			
	Enable	BOOL	-	RES	FALSE: conversión de módulo desactivada TRUE: conversión de módulo activada Si la conversión de módulo está activada, se comprueba que la longitud del módulo > 0.0.
	Length	LREAL	de 0.001 a 1.0E12	RES	Longitud de módulo
	StartValue	LREAL	de -1.0E12 a 1.0E12	RES	Valor inicial de módulo

Variables Homing (encóder externo)

La estructura de variable <TO>.Homing.<nombre de variable> incluye la configuración para el referenciado del objeto tecnológico.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Valores	W	Descripción
Homing.		STRUCT			
	HomePosition	LREAL	de -1.0E12 a 1.0E12	DIR	Posición del punto de referencia

Variables StatusSensor (encóder externo)

La estructura de variable <TO>.StatusSensor.<nombre de variable> muestra el estado del sistema de medición.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Valores	W	Descripción
StatusSensor.		STRUCT			
	State	DINT	-	RON	Estado del valor real del encóder: 0: "NOT_VALID" (no válido) 1: "Waiting_FOR_VALID" (esperando estado válido) 2: "VALID" (válido)
	CommunicationOK	BOOL	-	RON	Comunicación BUS cíclica entre controlador y encóder FALSE: no establecida TRUE: establecida
	Error	BOOL	-	RON	FALSE: no hay errores en el sistema de medida TRUE: error en el sistema de medida
	AbsEncoderOffset	LREAL	-	RON	Decalaje del punto de referencia hasta el valor de un encóder absoluto. El valor se guarda de forma remanente en la CPU.

Variable StatusWord (encóder externo)

La variable <TO>.StatusWord incluye la información de estado del objeto tecnológico.

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 5 "HomingDone") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10829)

Variable	Tipo de datos	Valores	W	Descripción
Status Word	DWORD	-	RON	Información de estado del objeto tecnológico
Bit 0	-	-	-	"Enable" Estado de habilitación 0: objeto tecnológico bloqueado 1: objeto tecnológico habilitado
Bit 1	-	-	-	"Error" 0: no hay errores 1: hay errores
Bit 2	-	-	-	"RestartActive" 0: ningún "Restart" activo 1: "Restart" activo. El objeto tecnológico se reinicializa.

Variable	Tipo de datos	Valores	W	Descripción
Bit 3	-	-	-	"OnlineStartValuesChanged" 0: variable "Restart" sin cambios 1: cambios en la variable "Restart". Para aplicar los cambios es necesario reinicializar el objeto tecnológico.
Bit 4	-	-	-	Reservado
Bit 5	-	-	-	"HomingDone" Estado de referenciado 0: objeto tecnológico no referenciado 1: objeto tecnológico referenciado
Bit 6	-	-	-	"Done" 0: petición de movimiento en proceso o panel de mando del eje activado 1: no hay petición de movimiento en proceso y el panel de mando del eje está desactivado
Bit 7	-	-	-	Reservado
Bit 8	-		-	Reservado
Bit 9	-	-	-	Reservado
Bit 10	-	-	-	Reservado
Bit 11	-	-	-	"HomingCommand" 0: ninguna petición "MC_Home" en procesamiento 1: "petición "MC_Home" en procesamiento
Bit 12	-	-	-	Reservado
Bit 13	-	-	-	Reservado
Bit 14	-	-	-	Reservado
Bit 15	-	-	-	Reservado
Bit 16	-	-	-	Reservado
Bit 17	-	-	-	Reservado
Bit 18	-	-	-	Reservado
Bit 19	-	-	-	Reservado
Bit 20	-	-	-	Reservado
Bit 21... Bit 31	-	-	-	Reservado

Variable ErrorWord (encóder externo)

La variable <TO>.ErrorWord señala errores en el objeto tecnológico (alarmas tecnológicas).

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 3 "CommandNotAccepted") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10829)

Variable	Tipo de datos	Valores	W	Descripción
Error Word	DWORD	-	RON	
Bit 0	-	-	-	"SystemFault" Error de sistema
Bit 1	-	-	-	"ConfigurationFault" Error de configuración Hay uno o varios parámetros de configuración incoherentes o inadmisibles.
Bit 2	-	-	-	"UserFault" Error en una instrucción de Motion Control o en su utilización en el programa de usuario
Bit 3	-	-	-	"CommandNotAccepted" Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios.
Bit 4	-	-	-	Reservado
Bit 5	-	-	-	"SensorFault" Error en el sistema del encóder
Bit 6	-	-	-	Reservado
Bit 7	-	-	-	"CommunicationFault" Error de comunicación Falta la comunicación o es errónea.
Bit 8	-	-	-	Reservado
Bit 9	-	-	-	Reservado
Bit 10	-	-	-	"HomingFault" Error en el proceso de referenciado No es posible finalizar el referenciado.
Bit 11	-	-	-	Reservado
Bit 12	-	-	-	Reservado
Bit 13	-	-	-	"PeripheralError" Error al acceder a una dirección lógica
Bit 14... Bit 31	-	-	-	Reservado

Variables ErrorDetail (encóder externo)

La estructura de variable <TO>.ErrorDetail.<nombre de variable> incluye el número de alarma y la reacción local efectiva a la alarma tecnológica actualmente pendiente en el objeto tecnológico.

Encontrará una lista de las alarmas tecnológicas y las reacciones a alarma en el anexo Alarmas tecnológicas (Página 10841).

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Valores	W	Descripción
ErrorDetail.		STRUCT			
	Number	UDINT	-	RON	Número de alarma
	Reaction	DINT	de 0 a 4	RON	Reacción efectiva a alarma 0: sin reacción 1: parada con valores dinámicos actuales 2: parada con valores dinámicos máximos 3: parada con rampa de parada de emergencia 4: retirar habilitación En el objeto tecnológico Encóder externo solo se admiten las reacciones a alarma 0 y 4.

Variable WarningWord (encóder externo)

La variable <TO>.WarningWord indica las advertencias pendientes en el objeto tecnológico.

Encontrará indicaciones para evaluar los diferentes bits (p. ej. bit 11 "FollowingErrorWarning") en el capítulo Evaluar StatusWord, ErrorWord y WarningWord (Página 10749).

Variable

Leyenda (Página 10829)

Variable	Tipo de datos	Valores	W	Descripción
WarningWord	DWORD	-	RON	
Bit 0	-	-	-	Reservado
Bit 1	-	-	-	"ConfigurationFault" Error de configuración Uno o varios parámetros de configuración se adaptan internamente.
Bit 2	-	-	-	Reservado
Bit 3	-	-	-	"CommandNotAccepted" Comando no ejecutable. No es posible ejecutar una instrucción de Motion Control porque no se cumplen los requisitos necesarios.
Bit 4	-	-	-	Reservado
Bit 5	-	-	-	Reservado

Variable	Tipo de datos	Valores	W	Descripción
Bit 6	-	-	-	Reservado
Bit 7	-	-	-	Reservado
Bit 8	-	-	-	Reservado
Bit 9	-	-	-	Reservado
Bit 10	-	-	-	Reservado
Bit 11	-	-	-	Reservado
Bit 12	-	-	-	Reservado
Bit 13	-	-	-	Reservado
Bit 14... Bit 31	-	-	-	Reservado

Variables InternalTOTrace (encóder externo)

La estructura de variable <TO>.InternalTOTrace.<nombre de variable> no incluye datos relevantes para el usuario. Esta estructura de variable se utiliza internamente.

Variables

Leyenda (Página 10829)

Variable		Tipo de datos	Valores	W	Descripción
InternalTOTrace.		ARRAY [1..4] OF STRUCT			
	Id	DINT	-	DIR	-
	Value	LREAL	-	RON	-

Alarmas tecnológicas

La tabla siguiente muestra una sinopsis de las alarmas tecnológicas y las correspondientes reacciones. Cuando se produzca una alarma tecnológica, evalúe todo el texto de alarma visualizado para encontrar la causa exacta.

Encontrará preguntas frecuentes y soluciones sobre las alarmas tecnológicas en Internet (http://support.automation.siemens.com/WW/view/es/66958052).

Leyenda

N.º	Número de la alarma tecnológica (equivale a <TO>.ErrorDetail.Number)
Reacción	Reacción efectiva a alarma (equivale a <TO>.ErrorDetail.Reaction)

Bit de error	Bit activado en <TO>.ErrorWord cuando se produce la alarma tecnológica
Bit de advertencia	Bit activado en <TO>.WarningWord cuando se produce la alarma tecnológica
Reinicialización	Para acusar la alarma tecnológica debe reinicializarse el objeto tecnológico (restart).
Texto de alarma	Texto de alarma visualizado (limitado)

Lista de alarmas tecnológicas

N.º	Reacción	Bit de error	Bit de advertencia	Reinicialización	Texto de alarma
101	Retirar habilitación	ConfigurationFault	-	X	Error en la configuración.
104	Parada con valores dinámicos máximos	ConfigurationFault	-	-	Error en la indicación de finales de carrera por software.
105	Retirar habilitación	ConfigurationFault	-	X	Error de configuración (accionamiento/actuador).
106	Retirar habilitación	ConfigurationFault	-	X	Error de configuración en el driver de accionamiento/ actuador
107	Retirar habilitación	ConfigurationFault	-	X	Error de configuración del encóder.
108	Retirar habilitación	ConfigurationFault	-	X	Error de configuración del driver del encóder.
109	Retirar habilitación	ConfigurationFault	-	-	Error de configuración.
110	No hay reacción	-	ConfigurationFault	-	La configuración se adapta internamente.
201	Retirar habilitación	SystemFault	-	-	Error interno.
202	No hay reacción	SystemFault	-	-	Error de configuración interno.
203	Retirar habilitación	SystemFault	-	-	Error interno. Diríjase al servicio de atención al cliente.
204	Retirar habilitación	SystemFault	-	-	Error en la puesta en servicio.
304	Parada con rampa de parada de emergencia	UserFault	-	-	El valor límite de velocidad máxima es cero.
305	Parada con rampa de parada de emergencia	UserFault	-	-	El valor límite de aceleración/deceleración es cero.
306	Parada con rampa de parada de emergencia	UserFault	-	-	El valor límite de tirón es cero.
307	Parada con valores dinámicos máximos	UserFault	-	-	Se ha alcanzado el rango de valores numérico de la posición.
308	Retirar habilitación	UserFault	-	-	Se ha rebasado por exceso el rango de valores numérico de la posición.

N.º	Reacción	Bit de error	Bit de advertencia	Reinicialización	Texto de alarma
321	Parada con rampa de parada de emergencia	CommandNot Accepted	-	-	El eje no está referenciado.
322	No hay reacción	-	CommandNot Accepted	-	Restart no ejecutado.
323	No hay reacción	-	CommandNot Accepted	-	No se ha ejecutado MC_Home porque se produciría un rebase por exceso del rango de valores numérico.
341	Retirar habilitación	HomingFault	-	-	Datos de referenciado incorrectos.
342	Parada con rampa de parada de emergencia	HomingFault	-	-	No se ha encontrado la leva de referencia/marca cero del encóder.
401	Retirar habilitación	PeripheralError	-	-	Error al acceder a una dirección lógica.
411	Retirar habilitación	SensorFault	-	-	Encóder averiado en dirección lógica.
412	Retirar habilitación	SensorFault	-	-	Rebase por exceso del límite de rango del valor real incremental.
421	Retirar habilitación	DriveFault	-	-	Accionamiento/actuador averiado en dirección lógica.
431	Retirar habilitación	Communication Fault	-	-	Interferencias en la comunicación con el dispositivo.
501	No hay reacción	-	DynamicError	-	La velocidad programada se limita.
502	No hay reacción	-	DynamicError	-	La aceleración/deceleración programada se limita.
503	No hay reacción	-	DynamicError	-	El tirón programado se limita.
504	No hay reacción	-	DynamicError	-	Vigilancia activa de la consigna de velocidad de giro.
521	Retirar habilitación	FollowingError Fault	-	-	Rebase por exceso de la ventana para la vigilancia de errores de seguimiento.
522	No hay reacción		FollowingError Fault		Nivel de advertencia alcanzado para la vigilancia de errores de seguimiento.
531	Retirar habilitación	HWLimit	-	-	• Final de carrera por hardware positivo aproximado. • Final de carrera por hardware negativo aproximado. • Sentido de retirada no permitido con final de carrera por HW activo
531	Retirar habilitación	HWLimit	-	X	• Final de carrera por HW con los polos invertidos, retirada imposible • Los dos finales de carrera por HW están activos, retirada imposible

N.º	Reacción	Bit de error	Bit de advertencia	Reinicialización	Texto de alarma
533	Parada con valores dinámicos máximos	SWLimit	-	-	Final de carrera por software aproximado.
534	Retirar habilitación	SWLimit	-	-	Final de carrera por software sobrepasado.
541	Retirar habilitación	PostioningFault	-	-	Error en la vigilancia de posicionamiento.

Consulte también

Alarmas tecnológicas (Página 10782)

Identificación de error

Los errores en las instrucciones de Motion Control se avisan mediante los parámetros "Error" y "ErrorID".

En las condiciones siguientes, la instrucción "Error" de Motion Control presenta TRUE y "ErrorID" presenta 16#8xxx:

- Estado inadmisible del objeto tecnológico que impide la ejecución de la petición.
- Parametrización inadmisible de la instrucción de Motion Control, que impide la ejecución de la petición.
- Como consecuencia de la reacción a alarma de un error en el objeto tecnológico.

ErrorIDs y solución

La tabla siguiente recoge una relación de todas las "ErrorIDs" que pueden mostrarse en las instrucciones de Motion Control. Junto a la causa del error se muestran también ayudas para eliminar los errores:

ErrorID	Descripción	Solución
16#0000	Ningún error	-
16#8001	Durante el procesamiento de la instrucción de Motion Control se ha producido una alarma tecnológica (error en el objeto tecnológico).	En el bloque de datos tecnológico se emite un mensaje de error en la variable "ErrorDetail.Number". Encontrará una lista de las alarmas tecnológicas y las reacciones a alarma en el anexo Alarmas tecnológicas (Página 10838).
16#8002	Valor no admisible en el parámetro "Axis".	Compruebe el valor del parámetro "Axis".
16#8003	Indicación de velocidad inadmisible	Indique un valor válido para la velocidad en el parámetro "Velocity".
16#8004	Indicación de aceleración inadmisible	Indique un valor válido para la aceleración en el parámetro "Acceleration".
16#8005	Indicación de deceleración inadmisible	Indique un valor válido para la deceleración en el parámetro "Deceleration".
16#8006	Indicación de tirón inadmisible	Indique un valor válido para el tirón en el parámetro "Jerk".

ErrorID	Descripción	Solución
16#8007	Indicación de sentido inadmisible	Indique un valor válido para el sentido de giro en el parámetro "Direction".
	Si el eje está parado, "Direction" = 4 (sentido actual) no está permitido.	
	Entrada inadmisible Los dos parámetros "JogForward" y "JogBackward" son TRUE al mismo tiempo. El eje se frena con la última deceleración válida.	Desactive los dos parámetros "JogForward" y "JogBackward".
16#8008	Indicación de distancia inadmisible	Indique un valor válido para la distancia en el parámetro "Distance".
16#8009	Indicación de posición inadmisible	Indique un valor válido para la posición en el parámetro "Position".
16#800A	Modo de operación no admisible	Indique un modo de operación válido en el parámetro "Mode".
16#800B	Modo de parada inadmisible	Indique un modo de parada válido en el parámetro "StopMode".
16#800C	Solo se admite una instancia de la instrucción "MC_Power" por objeto tecnológico.	Se ha indicado un objeto tecnológico en varias instancias de la instrucción "MC_Power" en el parámetro "Axis". Asegúrese de que el objeto tecnológico solo está indicado en una instrucción "MC_Power".
16#800D	Petición no permitida en el estado actual. "Restart" se ejecuta.	Espere hasta que haya concluido el "Restart" del objeto tecnológico.
16#800E	Si el objeto tecnológico está habilitado no es posible un "Restart".	Antes de un "Restart" bloquee el eje tecnológico con "MC_Power.Enable" = FALSE.
16#800F	El objeto tecnológico está bloqueado. La petición no puede ejecutarse.	Habilite el objeto tecnológico con "MC_Power.Enable" = TRUE. Reinicie la petición.
16#8010	Función de referenciado inadmisible con encóder incremental.	Con un encóder incremental no es posible un ajuste del encóder absoluto ("Mode" = 6, 7). Inicie un proceso de referenciado para un encóder incremental con el parámetro "Mode" = 0, 1, 2, 3, 4, 5, 8.
16#8011	Función de referenciado inadmisible con encóder absoluto.	Con un encóder absoluto no es posible un referenciado pasivo y activo ("Mode" = 2, 3, 4, 5). Inicie un proceso de referenciado para un encóder absoluto con el parámetro "Mode" = 0, 1, 6, 7.
16#8012	El comando no puede ejecutarse porque el panel de mando del eje está activo.	Devuelva el control al programa de usuario. Reinicie la petición.
16#8013	Ha fallado la conexión online entra la CPU y el TIA Portal.	Revise la conexión online con la CPU.
16#8014	No hay disponible memoria de peticiones interna.	Se ha alcanzado el número máximo posible de peticiones de Motion Control. Reduzca el número de peticiones que procesar (parámetro "Execute" = FALSE).
16#8015	No es posible acusar el error con "MC_Reset". Error en la configuración del objeto tecnológico.	Revise la configuración del objeto tecnológico.

ErrorID	Descripción	Solución
16#8016	Los valores reales no son válidos.	Para ejecutar una petición MC_Home, los valores reales deben ser válidos. Revise el estado de los valores reales. La variable del objeto tecnológico <TO>.StatusSensor[n].State debe presentar el valor 2 (válido).
16#8FFF	Error no especificado	Póngase en contacto con su representante de Siemens en las agencias y oficinas de ventas de su zona. Encontrará a su persona de contacto en: Personas de contacto en Industry Automation and Drive Technologies (http://www.siemens.com/automation/partner)

Consulte también

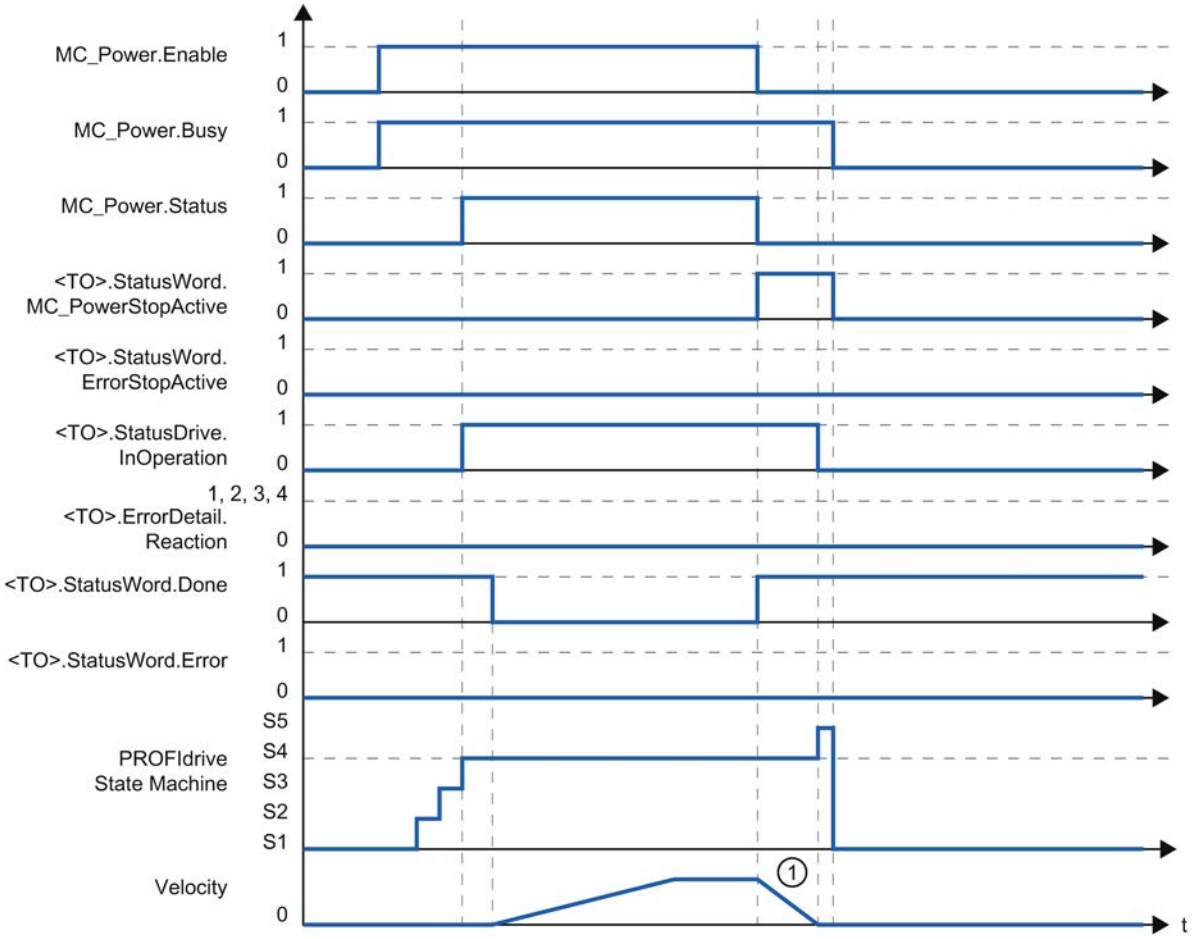
Errores en las instrucciones de Motion Control (Página 10785)

Diagramas de funcionamiento MC_Power

Conexión del accionamiento con PROFIdrive

StopMode 0

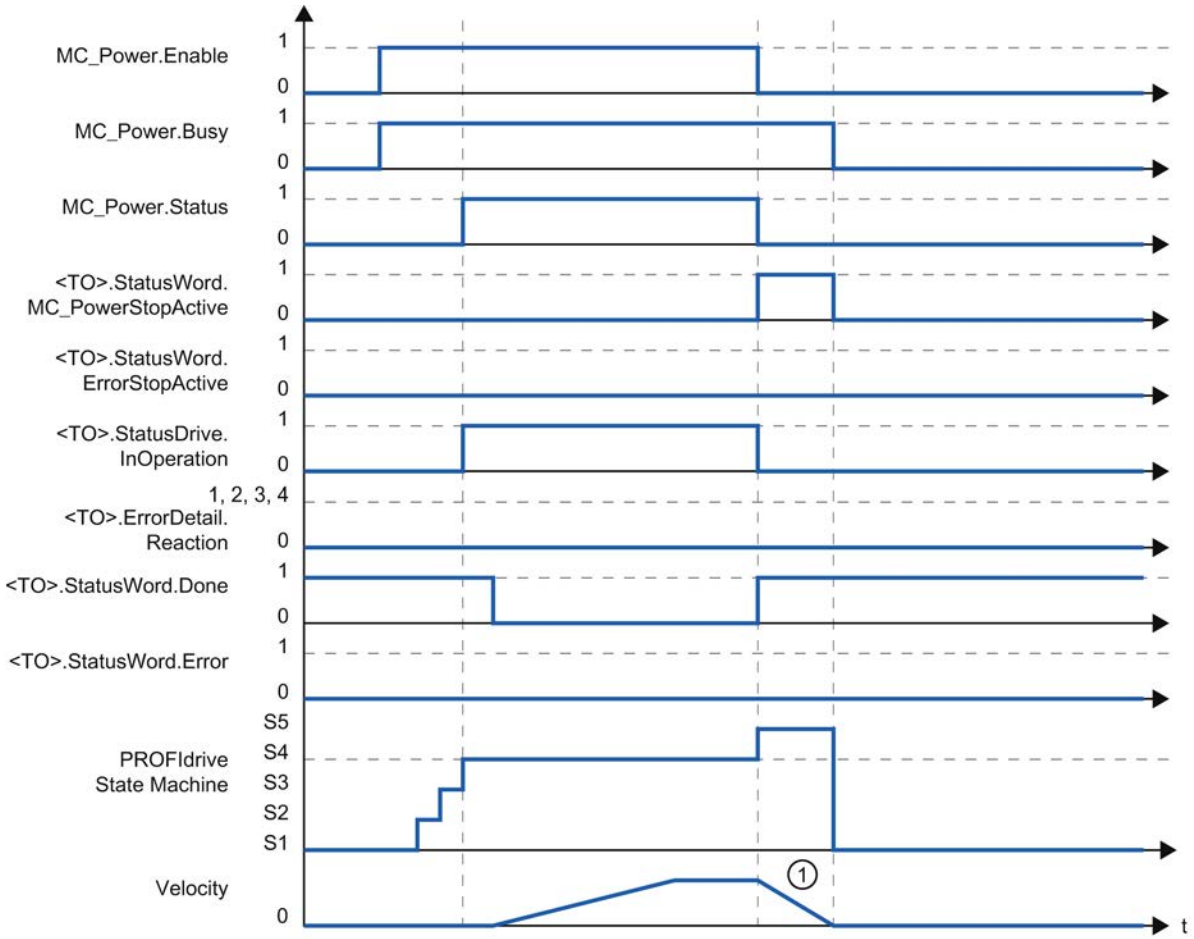
Diagrama de funcionamiento: habilitar un objeto tecnológico y bloquearlo con "StopMode" = 0



① El eje se frena con la deceleración de parada de emergencia configurada.

StopMode 1

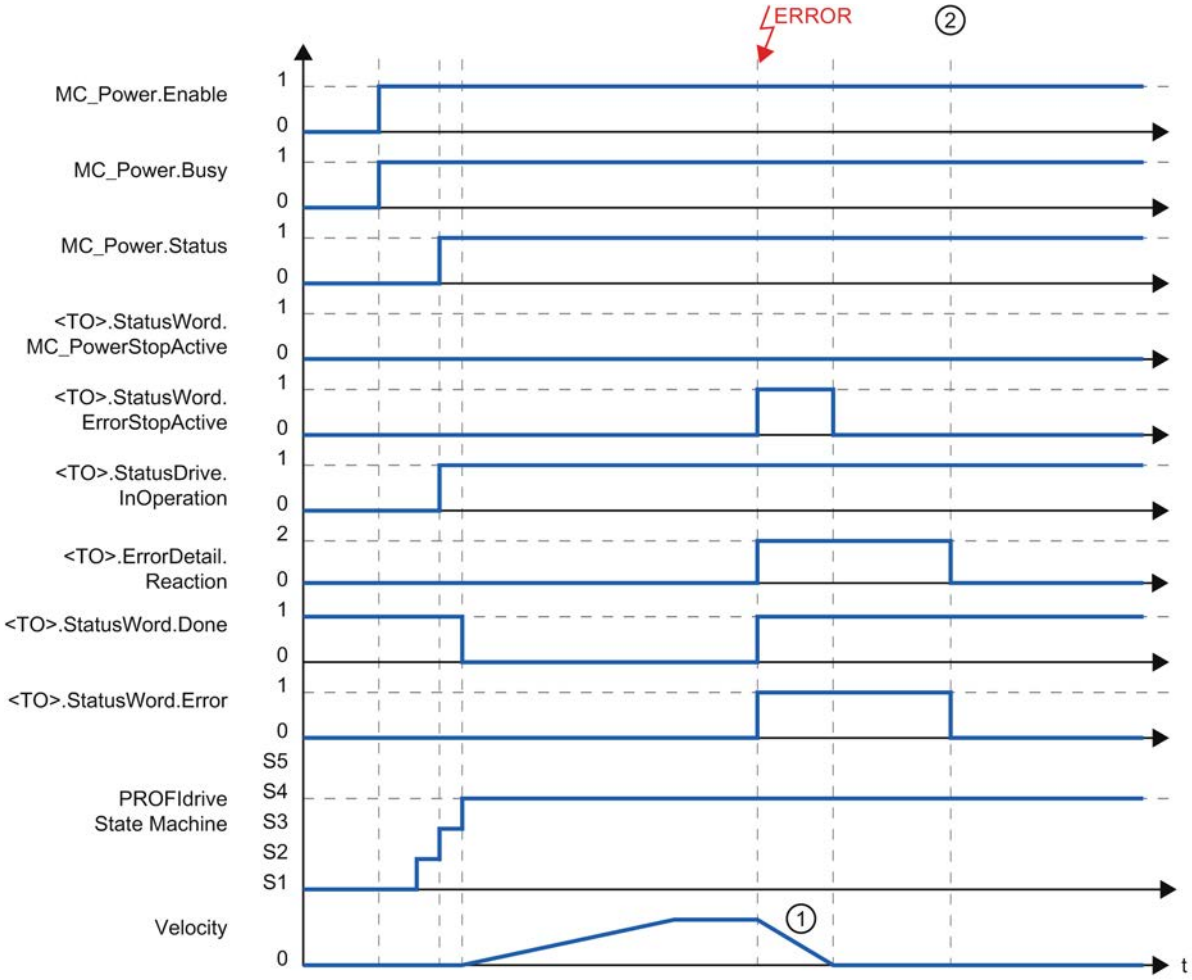
Diagrama de funcionamiento: habilitar un objeto tecnológico y bloquearlo con "StopMode" = 1



① La rampa de deceleración depende de la configuración en el accionamiento.

Reacción a alarma "Parada con valores dinámicos máximos"

Diagrama de funcionamiento: habilitar un objeto tecnológico y aparición de una alarma tecnológica con reacción a alarma "Parada con valores dinámicos máximos"

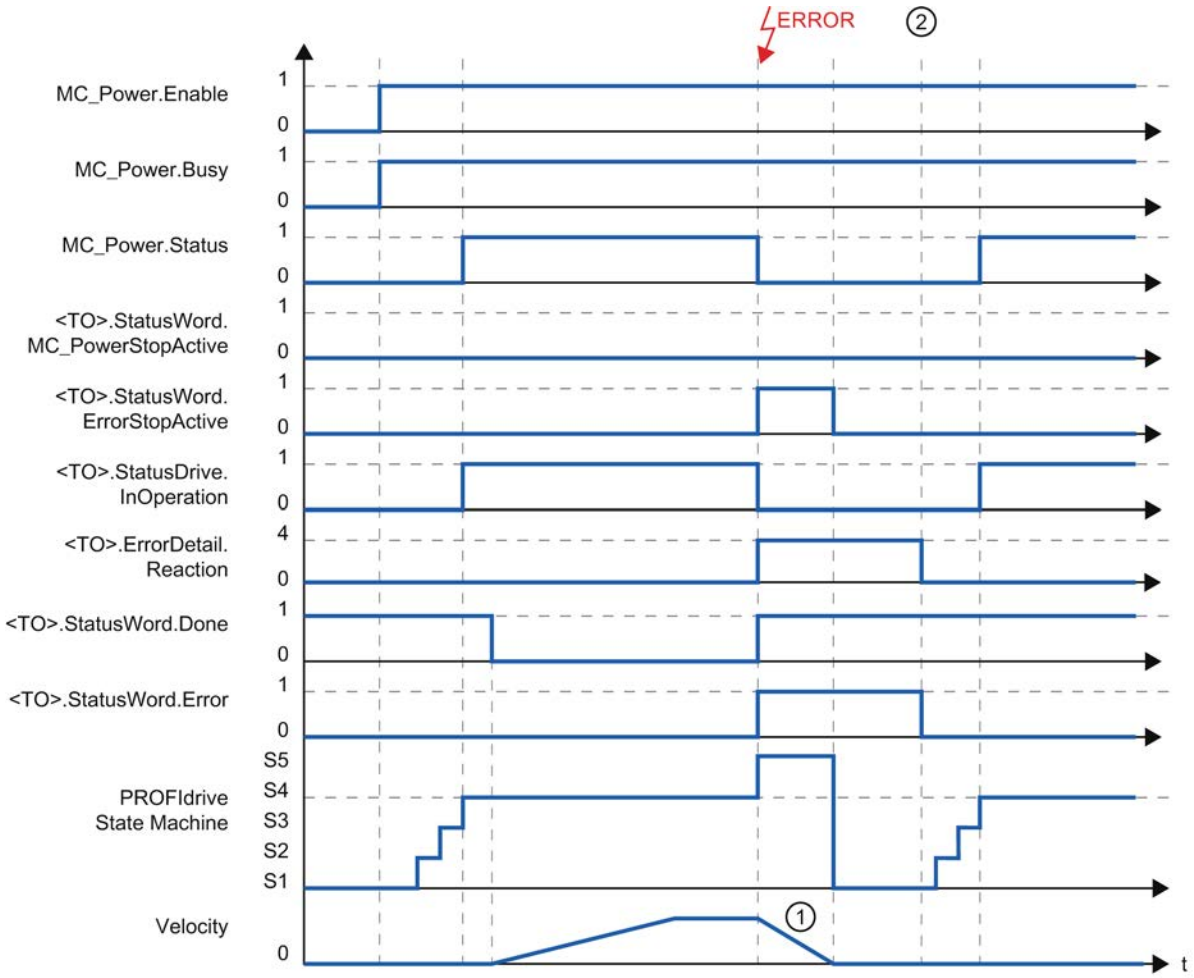


① El eje se frena con la deceleración máxima configurada.

② La alarma tecnológica se acusa.

Reacción a alarma "Retirar habilitación"

Diagrama de funcionamiento: habilitar un objeto tecnológico y aparición de una alarma tecnológica con reacción a alarma "Retirar habilitación"



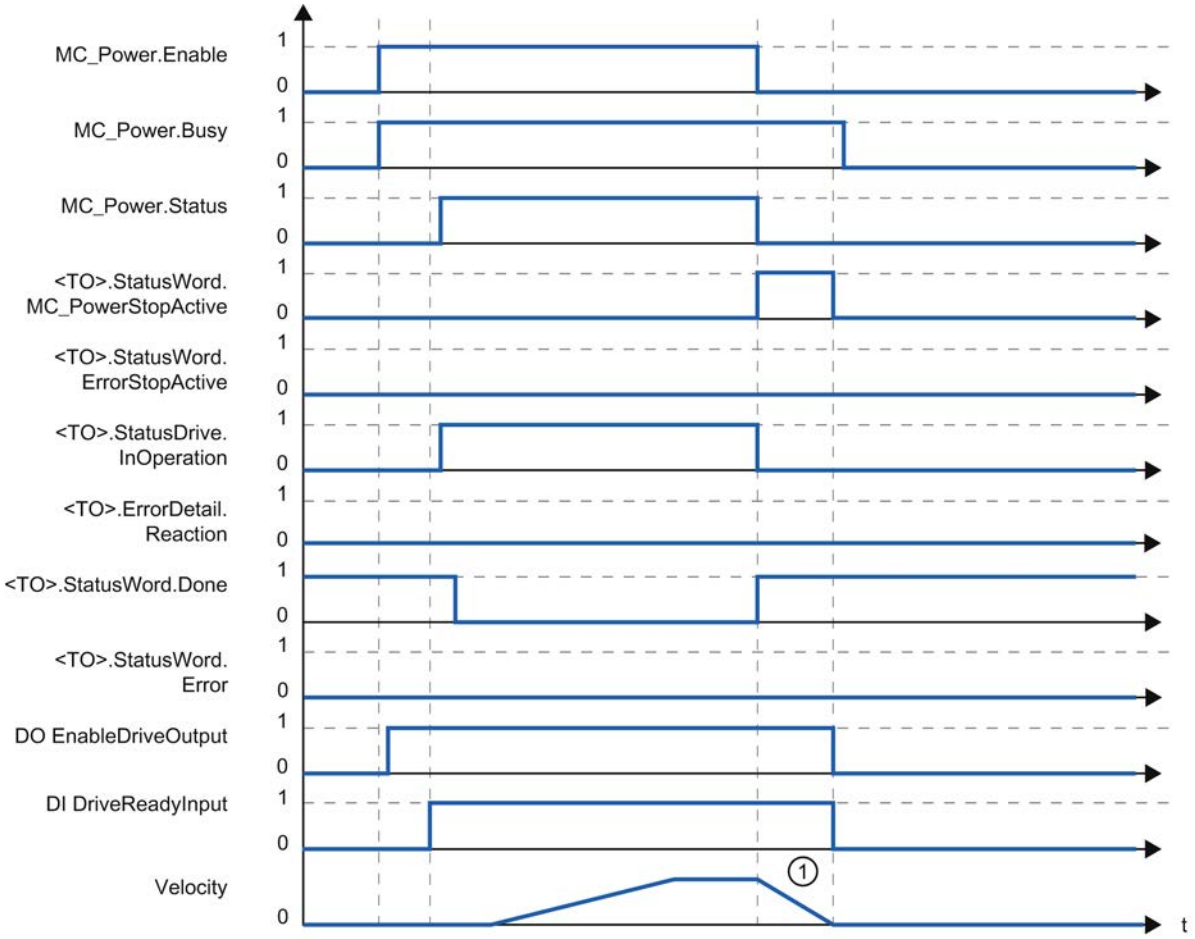
① La rampa de deceleración depende de la configuración en el accionamiento.

② La alarma tecnológica se acusa en el instante ②.

Conexión analógica del accionamiento

StopMode 0

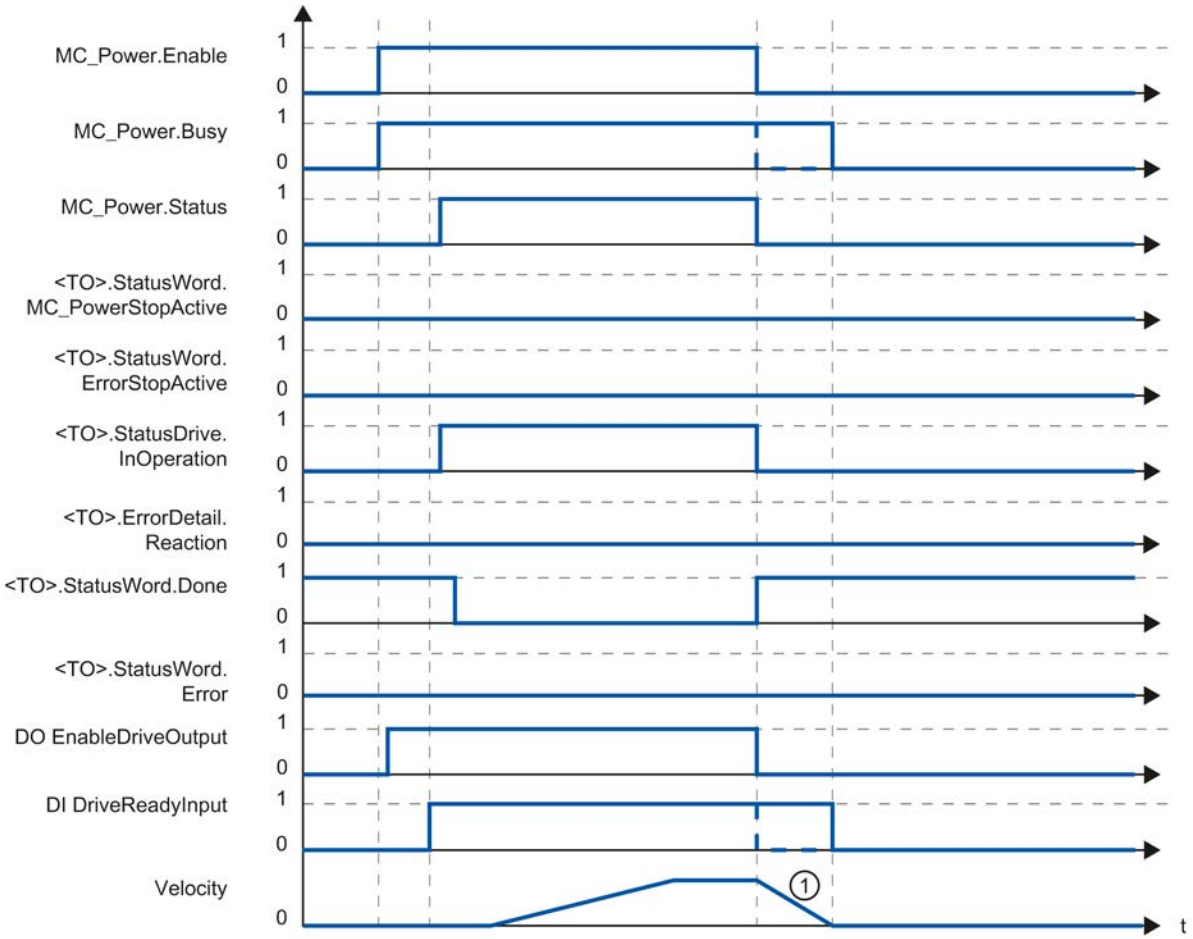
Diagrama de funcionamiento: habilitar un objeto tecnológico y bloquearlo con "StopMode" = 0



① El eje se frena con la deceleración de parada de emergencia configurada.

StopMode 1

Diagrama de funcionamiento: habilitar un objeto tecnológico y bloquearlo con "StopMode" = 1

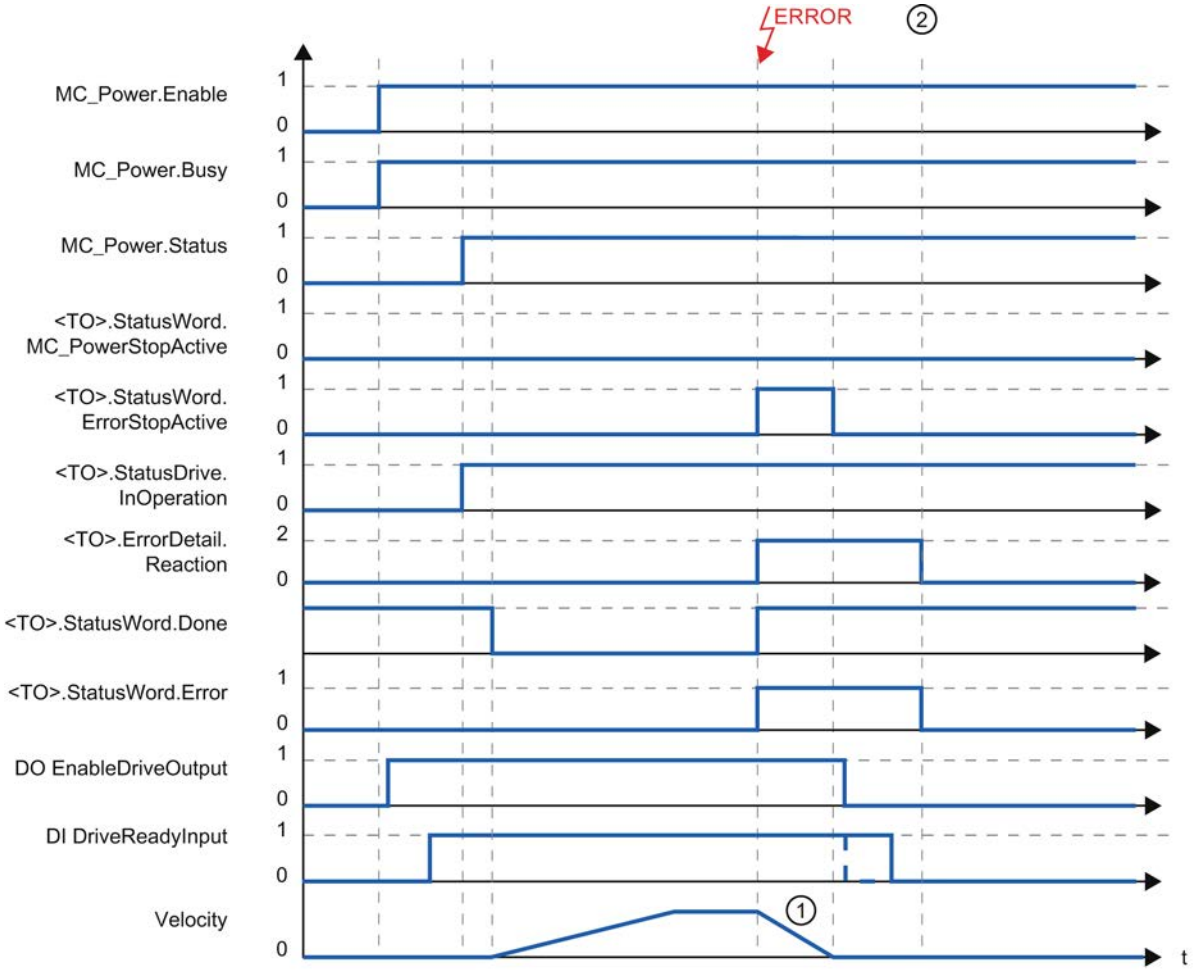


① La rampa de deceleración depende de la configuración en el accionamiento.

- - - El comportamiento de la señal de disponibilidad del accionamiento "DI DriveReadyInput" es específico del fabricante.

Reacción a alarma "Parada con valores dinámicos máximos"

Diagrama de funcionamiento: habilitar un objeto tecnológico y aparición de una alarma tecnológica con reacción a alarma "Parada con valores dinámicos máximos"



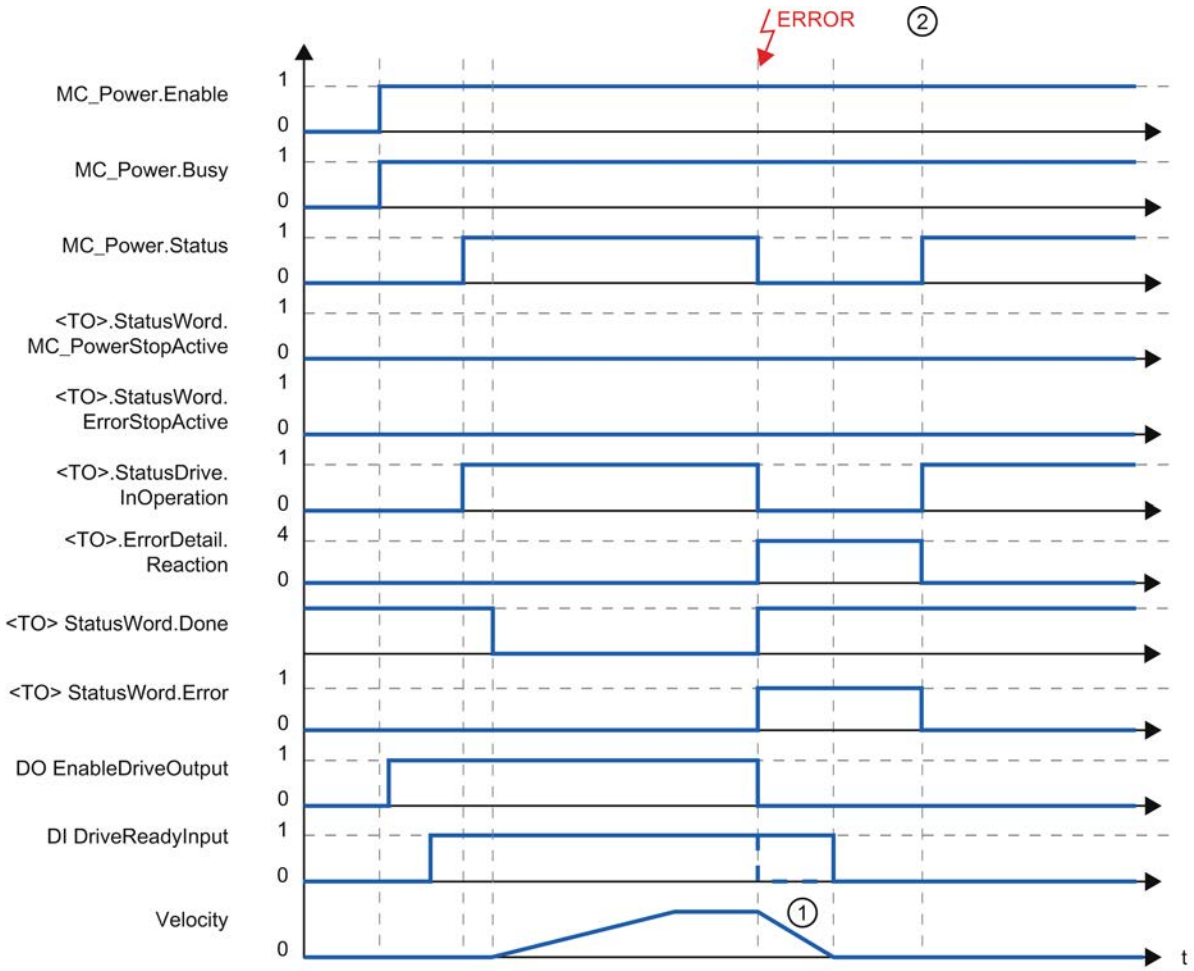
① El eje se frena con la deceleración máxima configurada.

② La alarma tecnológica se acusa en el instante ②.

El comportamiento de la señal de disponibilidad del accionamiento "DI DriveReadyInput" es específico del fabricante.

Reacción a alarma "Retirar habilitación"

Diagrama de funcionamiento: habilitar un objeto tecnológico y aparición de una alarma tecnológica con reacción a alarma "Retirar habilitación"



① La rampa de deceleración depende de la configuración en el accionamiento.

② La alarma tecnológica se acusa en el instante ②.

- - - El comportamiento de la señal de disponibilidad del accionamiento "DI DriveReadyInput" es específico del fabricante.

Accionamientos SINAMICS

Referenciado activo en accionamientos SINAMICS con marca cero externa

En los accionamientos SINAMICS con marca cero externa se sincroniza siempre hacia el lado izquierdo de la señal de la marca cero externa cuando el referenciado está activo. Esto

significa que en sentido de marcha positivo se sincroniza hacia un flanco positivo y, en sentido de marcha negativo, hacia un flanco negativo.

Si se invierte la señal también es posible sincronizar en el lado derecho de la señal de la marca cero externa. La inversión puede ajustarse en el accionamiento con el parámetro P490 de SINAMICS.

El referenciado hacia una marca cero del encóder o una marca cero externa se ajusta en el parámetro P495 de SINAMICS.

11.4.2 Control de movimiento (S7-300, S7-400)

11.4.2.1 Principios básicos para la lectura del recorrido

¿Qué es la lectura del recorrido?

Lectura del recorrido

La lectura del recorrido es la detección de los valores reales que están normalizados en unidades de longitud.

En función del encóder utilizado existen las siguientes diferencias:

- Con encóder incremental
  El módulo cuenta las señales del encóder incremental. Para crear un punto de referencia para la posición del encóder, antes de la lectura del recorrido es preciso realizar una búsqueda del punto de referencia. Por ejemplo, el contador puede ponerse en una marca de referencia a "0".
  Después de la búsqueda del punto de referencia (con eje sincronizado) el estado del contador es una medida para la posición actual.
- Con encóders absolutos
  No es preciso realizar una búsqueda del punto de referencia. La señal emitida corresponde a un posición fija.

Los valores calculados pueden procesarse de inmediato en el programa de usuario. De este modo es posible la respuesta directa a los valores del sensor desde sistemas en movimiento.

Lectura de los valores del sensor

Descripción

El encóder absoluto transfiere sus valores de encóder en telegramas al módulo. La transferencia de los telegramas es iniciada por el módulo.

- En el modo no isócrono la lectura de los valores del sensor se desarrolla de forma libre.
- En el modo isócrono la lectura de los valores del sensor se efectúa de forma síncrona respecto al ciclo de bus en cada T_i.

Lectura libre de los valores del sensor:

El módulo hace que se transfiera un telegrama cada vez que termine el temporizador monoestable parametrizado.

El módulo procesa de forma asíncrona con respecto a estos telegramas libres el valor leído del encóder en el ciclo de su tasa de actualización.

De ahí resulta que si elegimos una lectura asíncrona de los valores del sensor obtendremos valores de distinta antigüedad. La diferencia entre la antigüedad máxima y mínima es el jitter.

Lectura de los valores del sensor en modo isócrono

La lectura isócrona de los valores del sensor se ajusta automáticamente cuando el sistema de bus y el módulo periférico funcionan de forma sincronizada.

El módulo inicia la transmisión de un telegrama en cada ciclo de bus en el instante T_i.

El módulo procesa el valor transferido del encóder de forma síncrona con el ciclo de bus.

Convertidor Gray/Dual

Convertidor Gray/Dual

El parámetro "Convertidor Gray/Dual" permite determinar el tratamiento del código suministrado por el encóder:

- En el ajuste "Gray", el valor suministrado por el encóder absoluto en código gray se convierte a código dual.
- En el ajuste "Dual" no se altera el valor procedente del sensor.

Nota

Si se ha seleccionado el ajuste "Gray", el módulo convierte siempre todo el valor del encóder (13 a 25 bits). Debido a ello, los bits especiales antepuestos influyen en el valor del sensor y podrían falsearse eventualmente los bits agregados.

Valor del sensor y normalización

Valor del sensor y normalización

El valor transmitido por el encóder incluye la posición del encóder absoluto. Según el encóder utilizado, además de su posición se transmiten otros bits, situados delante y detrás de la posición del encóder.

Para que el módulo pueda determinar la posición del encóder, se debe especificar lo siguiente:

- Tipo de encóder
- Número de bits a la derecha
- Incrementos/Vuelta del encóder

Con la normalización se especifica la representación de los valores del sensor en la interfaz de realimentación.

- Con la entrada "Número de bits a la derecha = 1, 2 … 12" (con SM 338) o la "normalización activada" (con ET 200S 1SSI) se especifica que los bits irrelevantes del valor del encóder sean desplazados. El valor del encóder se alinea a la derecha en el área de direcciones (véase ejemplo siguiente).
- Con la entrada "Número de bits a la derecha = 0" (con SM 338) o la "normalización desactivada" (con ET 200S 1SSI) se especifica que los bits pospuestos se mantengan y se utilicen para la evaluación.
  Esto puede resultar útil si utiliza un encóder absoluto que, en los bits pospuestos, transfiere informaciones (consulte las indicaciones del fabricante) y desea evaluar dichas informaciones. Tenga en cuenta a tal fin las informaciones sobre el convertidor Gray/ Dual (Página 10853).

Ejemplo para la normalización de un valor del encóder

Preajustes:

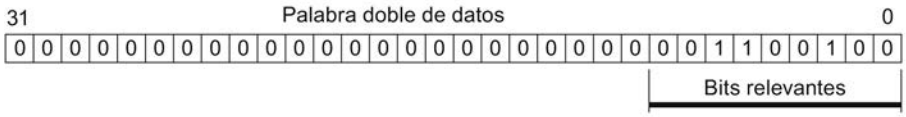
Se utiliza un encóder monovuelta con 2^9 (equivale a 9 bits) = 512 incrementos/vuelta de encóder (resolución / 360°) con la siguiente parametrización:

- Tipo de encóder: SSI-13 Bit
- Número de bits a la derecha: 4 dígitos
- Incrementos/vuelta del encóder: 512

La figura siguiente muestra la ubicación de los bits relevantes antes y después de la normalización.

Antes de la normalización: valor del captador registrado cíclicamente 100

Después de la normalización: valor del encoder 100



Resultado: se han suprimido los bits de 0 a 3 (4 dígitos identificados mediante"x").

11.4.2.2 Utilizar SM 338

Asignación de terminales del conector frontal del SM 338

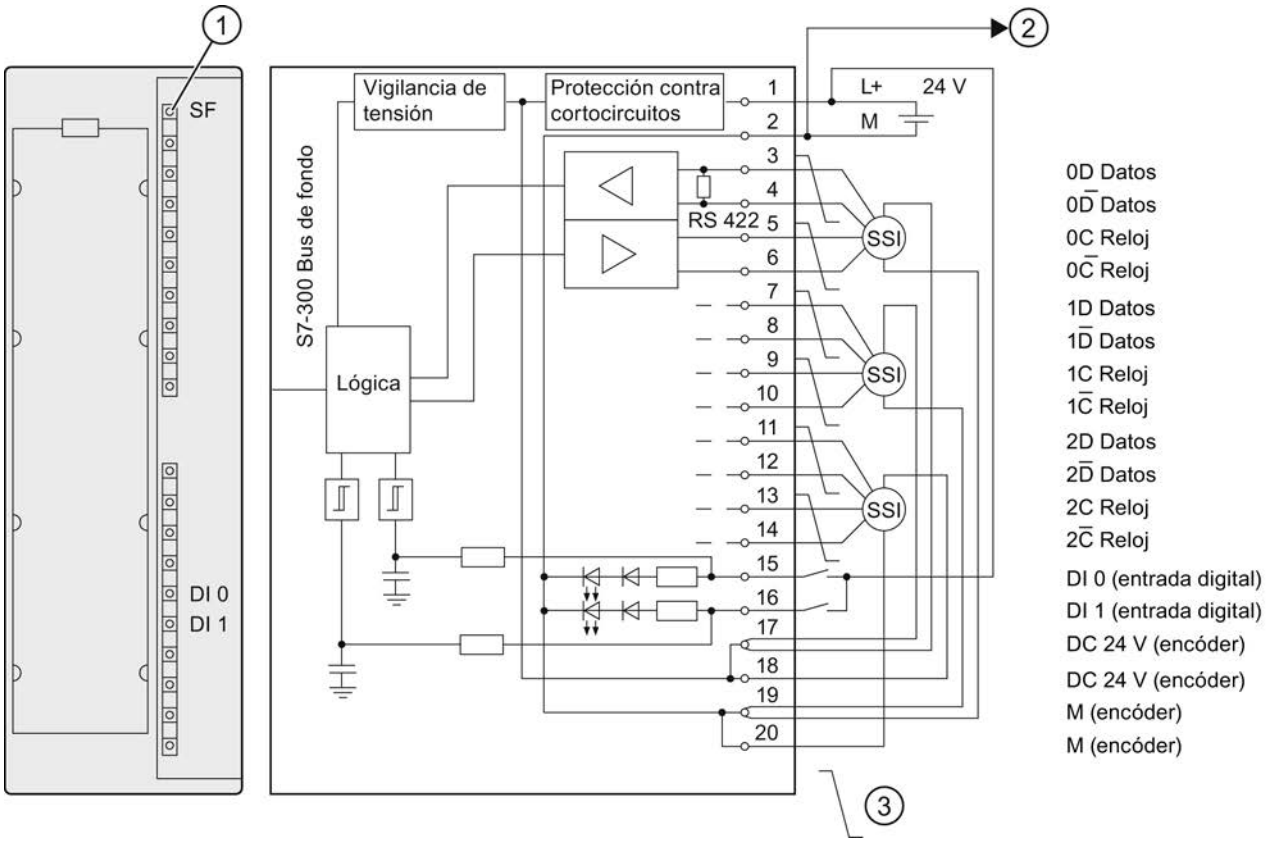
Reglas de cableado

Al realizar el cableado del módulo, observe las siguientes reglas:

- La masa de la alimentación del sensor no está separada galvánicamente respecto a la masa de la CPU. Por consiguiente, hay que conectar el pin 2 del SM 338 (M) con baja impedancia a la masa de la CPU.
- Los conductores de sensor (pins de 3 a 14) deben estar apantallados y trenzados por pares. Aplicar el blindaje en ambos extremos.
  Para el apantallamiento en el SM 338, utilice el estribo de contactado de pantallas (referencia: 6ES7390-5AA00-0AA0).
- Si se rebasa la máxima intensidad de salida (900 mA) de la alimentación del encóder, es necesario conectar una tensión de alimentación externa.

Asignación de terminales del conector frontal y esquema de principio

La figura siguiente muestra la asignación de terminales del conector frontal del SM 338 y el esquema de principio.



① Indicador de error - rojo
② Enlace a la masa de CPU
③ Conductores trenzados por pares

Configurar SM 338

Parametrizar SM 338

Modo de operación

Activación del Fast Mode

Fast Mode

El Fast Mode se activa en las correspondientes casillas de verificación.

En el Fast Mode el módulo trabaja con una captación de valores de encóder más rápida y una interfaz de respuesta comprimida. La función FREEZE no es posible.

Operación sincronizada

Requisitos de hardware

Para el modo isócrono del SM 338 necesita lo siguiente:

- la CPU debe soportar el modo isócrono
- el maestro DP debe soportar el ciclo de bus equidistante
- el módulo de interfaz IM 153-x debe soportar el modo isócrono

Propiedades del SM 338 en el modo isócrono

Según la parametrización del sistema, el SM 338 funciona o bien en modo no isócrono o bien en modo isócrono.

En modo isócrono se intercambian los datos entre el maestro DP y el SM 338 con sincronismo de reloj respecto al ciclo PROFIBUS DP.

En modo isócrono son coherentes los 16 bytes de la interfaz de respuesta.

Si se pierde el modo isócrono debido a una anomalía o bien por fallar o retrasarse el Global Control (GC), el SM 338 reanuda en el próximo ciclo el modo isócrono sin ninguna reacción por este error.

En caso de perderse el sincronismo de cadencias no es actualizada la interfaz de respuesta.

Nota

Los principios básicos del modo isócrono se describen en el manual de funciones Isochrone Mode (http://support.automation.siemens.com/WW/view/en/15218045).

Habilitación

Activación de la alarma de diagnóstico

Alarma de diagnóstico

Las alarmas de diagnóstico no están preajustadas, es decir que están bloqueadas sin la activación correspondiente.

Las alarmas de diagnóstico se activan en la correspondiente casilla de verificación.

En caso de haber habilitado alarmas de diagnóstico, los eventos de error entrantes (primera aparición del error) y los eventos de error salientes (aviso tras eliminarse todos los errores) se notifican a través de alarmas (Página 10866).

La CPU interrumpe la ejecución del programa de usuario y procesa el OB de alarma diagnóstico.

La información de diagnóstico es coherente mientras actúa el OB de alarma de diagnóstico. Tras abandonarse el OB de alarma de diagnóstico, la alarma de diagnóstico se confirma en el módulo.

Entrada de encóder n

Seleccionar tipo de encóder

Tipo de encóder

El SM 338 admite los siguientes tipos de encóder:

- Encóder absoluto (SSI) con una longitud de telegramas de 13 bits
- Encóder absoluto (SSI) con una longitud de telegramas de 21 bits
- Encóder absoluto (SSI) con una longitud de telegramas de 25 bits

Encontrará el tipo de encóder en los datos técnicos del encóder absoluto que utilice.

Con la selección "Sin encóder" la entrada de encóder está desactivada.

Selección del tipo de código

Tipo de código

Con el parámetro "Tipo de código" se determina el tratamiento del código que proporciona el encóder.

- Con el ajuste "Gray", el valor suministrado por el encóder absoluto en código gray se convierte a código dual.
- Con el ajuste "Dual", el valor proporcionado por el encóder permanece invariable.

Seleccionar velocidad de transferencia

Velocidad de transferencia

Con la "Velocidad de transferencia" se determina la velocidad de la transmisión de datos del encóder SSI al SM 338.

Seleccione la velocidad de transferencia deseada en la lista desplegable.

Tenga en cuenta la relación entre la longitud del cable y la velocidad de transferencia:

- hasta máx. 320 m → 125 kHz
- hasta máx. 160 m → 250 kHz

- hasta máx. 60 m → 500 kHz
- hasta máx. 20 m → 1 MHz

Seleccionar tiempo monoestable

Tiempo monoestable

El tiempo monoestable es el tiempo de pausa entre 2 telegramas SSI.

El tiempo monoestable parametrizado debe ser mayor que el tiempo monoestable del encóder absoluto utilizado (ver los datos técnicos del fabricante).

Al valor parametrizado se suma además el tiempo 2 × (1/velocidad de transferencia). Con una velocidad de transferencia de 125 kHz, con un tiempo monoestable parametrizado de 16 µs en realidad es efectivo un tiempo monoestable de 32 µs.

Para el tiempo monoestable del encóder absoluto se aplica la siguiente limitación:

(1/velocidad de transferencia) < tiempo monoestable del encóder absoluto < 64 µs + 2 × (1/velocidad de transferencia)

Seleccione el tiempo monoestable correspondiente en la lista desplegable.

La indicación del tiempo monoestable es relevante para la captación de valores de encóder asíncrona.

Introducir número de bits a la derecha

Número de bits a la derecha [0..12]

Para determinar la posición del encóder y detectar correctamente el sentido del movimiento del encóder, el SM 338 necesita el número de bits a la derecha.

Encontrará este valor en los datos técnicos del encóder absoluto que utilice.

Selección de incrementos/vuelta

Incrementos/vuelta

Como máximo están disponibles 13 bits para los incrementos/vuelta. De acuerdo con la indicación "Número de bits a la derecha (Página 10859)" se calcula automáticamente el número resultante de incrementos/vueltas.

Selección de la entrada digital para la función FREEZE

Función FREEZE

La función FREEZE sirve para "congelar" los valores de encóder actuales del SM 338. Esta función FREEZE está acoplada a las entradas digitales DI 0 y DI 1 del SM 338.

La congelación es iniciada por un cambio de flanco (flanco ascendente) en la entrada digital DI 0 o DI 1. Un valor de encóder congelado se identifica por estar activado el bit 31 (dirección de salida). A través de una entrada digital es posible congelar uno, dos o tres valores del encóder.

Los valores de captador se conservan hasta que concluye la función FREEZE, siendo así evaluables en función de los eventos.

Activación de la función FREEZE

La función FREEZE se debe activar al efectuar la parametrización.

Para ello seleccione la entrada digital en la que un flanco ascendente congela el valor de encóder.

- desactivar (el valor de encóder no se congela)
- DI 0
- DI 1

Conclusión de la función FREEZE

La función FREEZE se debe concluir tras cada entrada de captador. Puede confirmar la función en el programa de usuario seteando el bit 0, 1 o el 2 con la operación T PAB "xyz" de acuerdo con la entrada de encóder (ver ejemplo de programa (Página 10863)).

Tras la cancelación se borra el bit 31 del respectivo valor de encóder y vuelven a actualizarse los valores de encóder. Para congelar nuevamente los valores de captador, basta con borrar el bit de cancelación en la dirección de salida del módulo.

En modo isócrono se procesa la cancelación en el instante T_o. A partir de ese instante es posible congelar nuevamente los valores de captador a través de las entradas digitales.

Nota

La función FREEZE se confirma automáticamente si se reparametriza la correspondiente entrada de encóder con valores diferentes.
En caso de parámetros idénticos, la función FREEZE permanece inalterada.

Direccionamiento del SM 338

Áreas de datos para los valores de captador

Tanto las entradas como las salidas del SM 338 son direccionadas a partir de la dirección inicial del módulo. La dirección de entrada y la de salida se determinan al configurar el SM 338.

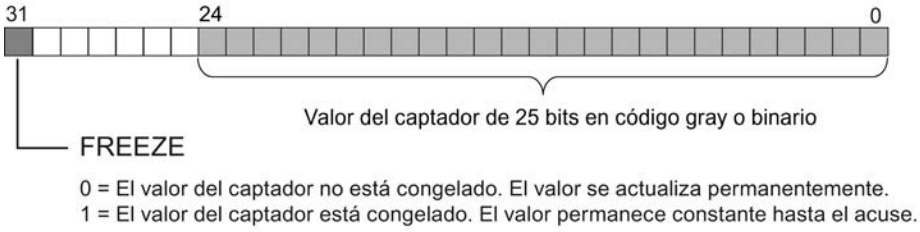
Direcciones de entrada

La tabla siguiente muestra las direcciones de entrada del módulo SM 338.

Entrada de captador	Dirección de entrada (de la configuración) + offset de dirección
0	"Dirección inicial del módulo"
1	"Dirección inicial del módulo" + offset de dirección de 4 bytes
2	"Dirección inicial del módulo" + offset de dirección de 8 bytes

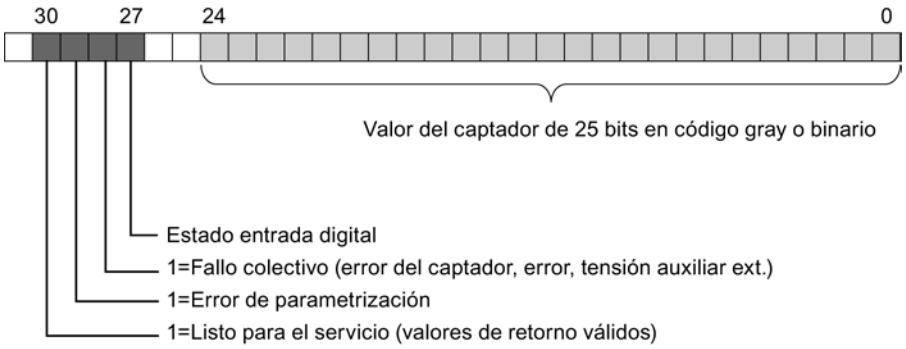
Estructura de la palabra doble de datos en Standard Mode

La tabla siguiente muestra, por entrada de encóder, la estructura de la palabra doble de datos en Standard Mode.



Estructura de la palabra doble de datos en Fast Mode

La tabla siguiente muestra, por entrada de encóder, la estructura de la palabra doble de datos en Fast Mode.

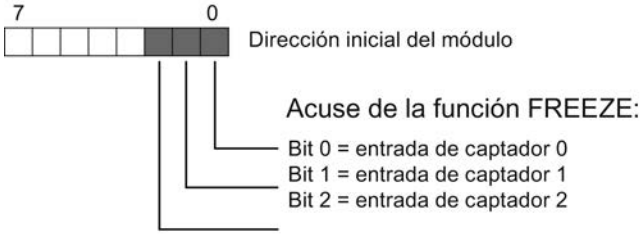


En el bit 27 (estado de la entrada digital) de la palabra doble de datos de la entrada de encóder 0 se indica el estado de la entrada digital DI 0, en tanto que en la palabra doble de datos de la entrada de encóder 1 se indica el estado de la entrada digital DI 1.

En la palabra doble de datos de la entrada de encóder 2 el bit 27 siempre es = 0.

Datos de salida en Standard Mode

La figura siguiente muestra la estructura de los datos de salida en Standard Mode.



En el Fast Mode no se soportan datos de salida.

Lectura de áreas de datos

Puede leer las áreas de datos en su programa de usuario con la operación L PED "xyz".

Ejemplo de programa para el acceso a valores de encóder y utilización de la función FREEZE

Ud. desea leer el valor del encóder en sus entradas y evaluarlo. La dirección inicial del módulo es 256.

AWL	Explicación
L PED256	// Leer valor del encóder en el área de direcciones para la entrada de encóder 0
T MD 100	// Guardar el valor del encóder en la palabra doble de marcas
AD W#16#1FFFFFF	// conectar lógicamente con "Y" los 25 bits inferiores para inhibir por completo los 7 bits superiores
T MD 120	// El valor del encóder sin FREEZE u otros bits de estado también pueden utilizarse en Fast Mode, debido a que sólo se visualizan 25 bits
A M 100.7	// Estado FREEZE para su posterior acuse
= M 99.0	// Determinar y guardar
L PID 260	// Leer valor del encóder en el área de direcciones para la entrada de encóder 1
T MD 104	// Guardar el valor del encóder en la palabra doble de marcas
AD W#16#1FFFFFF	// conectar lógicamente con "Y" los 25 bits inferiores para inhibir por completo los 7 bits superiores
T MD 124	// El valor del encóder sin FREEZE u otros bits de estado también pueden utilizarse en Fast Mode, debido a que sólo se visualizan 25 bits
A M 104.7	// Estado FREEZE para su posterior acuse
= M 99.1	// Determinar y guardar

AWL	Explicación
L PID 264	// Leer valor del encóder en el área de direcciones para la entrada de encóder 2
T MD 108	// Guardar el valor del encóder en la palabra doble de marcas
AD W#16#1FFFFFF	// conectar lógicamente con "Y" los 25 bits inferiores para inhibir por completo los 7 bits superiores
T MD 128	// El valor del encóder sin FREEZE u otros bits de estado también pueden utilizarse en Fast Mode, debido a que sólo se visualizan 25 bits
A M 108.7	// Estado FREEZE para su posterior acuse
= M 99.2	// Determinar y guardar
L MB 99	// Cargar el estado FREEZE y
T PQB 256	// acusarlo (SM 338: dirección de salida 256)

Después pueden seguir procesándose los valores de encóder del área de marcas MD 100, MD 104 y MD 108. El valor del encóder aparece en los bits de 0 a 30 de la palabra doble de marcas.

SM 338: Diagnóstico online

Diagnóstico del SM 338

El SM 338 ofrece avisos de diagnóstico. Es decir, el SM 338 pone siempre a disposición todos los avisos de diagnóstico sin intervención del usuario.

Acciones después del aviso de diagnóstico

Cada aviso de diagnóstico provoca las acciones siguientes:

- El aviso es registrado en el búfer de diagnóstico del módulo y retransmitido a la CPU.
- Se enciende el LED SF del módulo.
- Si ha parametrizado "Habilitar alarmas de diagnóstico (Página 10857)", se activa una alarma de diagnóstico y se llama el OB de alarma de diagnóstico.

Lectura de avisos de diagnóstico

Los avisos de diagnóstico detallados se pueden leer mediante instrucciones en el programa de usuario.

La causa del error se puede mostrar en el diagnóstico del módulo.

Aviso de diagnóstico mediante el LED SF

El SM 338 señaliza errores a través de su LED SF (LED de error agrupado). El LED SF se enciende tan pronto como el SM 338 activa un aviso de diagnóstico. y se apaga tras haberse eliminado todos los errores.

El LED SF se enciende también en caso de fallos externos (cortocircuito de la alimentación del encóder), independientemente del estado de la CPU (con RED CON).

El LED SF se enciende brevemente en el arranque durante el autodiagnóstico del SM 338.

Avisos de diagnóstico del SM 338

La tabla siguiente muestra los avisos de diagnóstico del SM 338.

Aviso de diagnóstico	LED	Validez del diagnóstico
Fallo del módulo	SF	Módulo
Error interno	SF	Módulo
Error externo	SF	Módulo
Error de canal existente	SF	Módulo
Falta tensión auxiliar externa	SF	Módulo
Módulo no parametrizado	SF	Módulo
Parámetros erróneos	SF	Módulo
Información de canal existente	SF	Módulo
Temporizador de vigilancia (watch dog) activado	SF	Módulo
Error de canal existente	SF	Canal (entrada del captador)
Error de configuración/parametrización	SF	Canal (entrada del captador)
Error de canal externo (error del captador)	SF	Canal (entrada del captador)

causas de anomalía y remedios

La tabla siguiente muestra causas de error y remedios en caso de que aparezcan avisos de diagnóstico del SM 338.

Aviso de diagnóstico	Causa posible	Remedio
Fallo del módulo	Se presentó una anomalía cualquiera detectada por el módulo.	
Error interno	El módulo detectó un error dentro del sistema de automatización.	
Error externo	El módulo detectó un error fuera del sistema de automatización.	
Error de canal existente	Indica que sólo están defectuosos determinados canales.	
Falta tensión auxiliar externa	Falta la tensión de alimentación L+ del módulo.	Llevar la alimentación L+

Aviso de diagnóstico	Causa posible	Remedio
Módulo no parametrizado	El módulo tiene que saber si debe operar mediante parámetros predeterminados por parte del sistema o con sus propios parámetros.	Este aviso permanece desde RED CON hasta concluir la transmisión de los parámetros desde la CPU; en caso dado, parametrizar el módulo.
Parámetros erróneos	Un parámetro o combinación de parámetros no es plausible.	Reparametrizar el módulo
Información de canal existente	Se presentó un error de canal; el módulo puede proporcionar más información de canal.	
Temporizador de vigilancia (watch dog) activado	Temporalmente altas perturbaciones electromagnéticas	Eliminar las perturbaciones
Error de canal existente	En una entrada de captador se presentó una anomalía cualquiera detectada por el módulo.	
Error de configuración/ parametrización	Parámetro erróneo transferido al módulo.	Reparametrizar el módulo
Error de canal externo (error del captador)	Rotura de hilo en el cable del captador, cable del captador no conectado o captador defectuoso.	Revisar los captadores conectados

11.4.2.3 Utilizar ET 200S 1SSI

Asignación de terminales del 1SSI

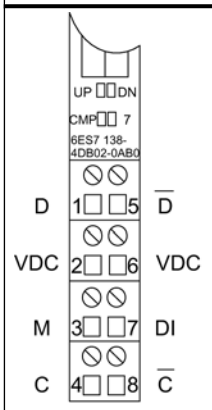
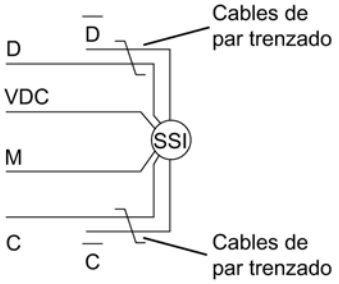
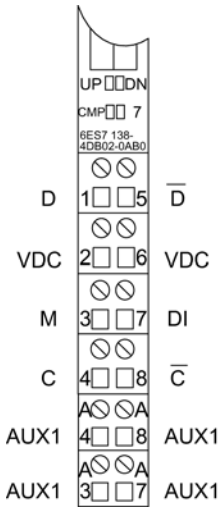
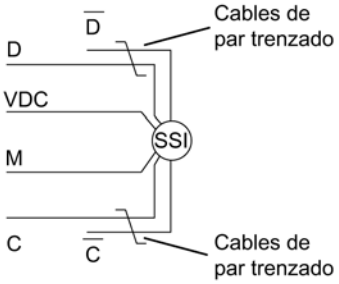
Reglas de cableado

Los cables (bornes 1 y 5 y bornes 4 y 8) deben estar apantallados y deben ser de par trenzado. La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contactado de pantallas (ver el anexo de las instrucciones de servicio Sistema de periferia descentralizada ET 200S (http://support.automation.siemens.com/WW/view/es/1144348)).

Asignación de terminales del 1SSI

La tabla siguiente muestra la asignación de terminales para el 1SSI.

Vista	Asignación de terminales	Observaciones	
		Bornes 1 a 8:	
		1/5:	Datos del encóder SSI [1)]
		2/6:	Alimentación para encóder absoluto e interruptor [2)]
		3:	Tierra del bastidor
		7:	Entrada digital Función de congelación

Vista	Asignación de terminales	Observaciones	
	TM-E15S24-01 y EM 1SSI 	4/8:	Reloj SSI (línea reloj) [1]
	TM-E15S26-A1 y EM 1SSI 		

[1] Asegúrese durante el cableado de que la polaridad sea correcta. De lo contrario se notificará un error del encóder absoluto. Señales según RS 422

[2] A prueba de cortocircuitos, máx. 0,5 A.

Configurar y parametrizar 1SSI

En servicio

Seleccionar modo de operación

Modo de servicio

Al preseleccionar un modo de operación se elige la funcionalidad con la que se utilizará el 1SSI.

Modo de servicio	Descripción
Modo estándar (Página 10870)	Modo de operación estándar del 1SSI con funcionalidad completa: • Normalización del valor del encóder • Inversión del sentido de giro para adaptar el sentido de giro del encóder absoluto al eje. • Modo de lectura de los valores del sensor seleccionable: – libre – sincronizado con la tasa de actualización • Consideración de la velocidad máxima de muestreo del encóder (p. ej. en caso de encóders ultrasónicos) en modo isócrono • Señal de vida en modo isócrono • Posibilidad de comprobar la paridad del valor del encóder • Convertidor Gray/Dual • Función de congelación para congelar el valor del encóder actual • Función de comparación entre el valor del encóder actual y los valores de comparación que pueden cargarse Este modo de operación resulta adecuado para vigilar o reconocer puntos de recorrido, así como para medir longitudes, captar bordes y sincronizar piezas de trabajo.
Modo Fast (Página 10882)	En este modo de operación el 1SSI tiene una funcionalidad comprimida. No están disponibles las funciones de congelación y comparación. Este modo de operación resulta adecuado para leer rápidamente los valores del sensor, así como para aplicaciones de regulación, como las regulaciones de posición con recorrido como valor real.

Parámetros para el modo estándar

Activar diagnóstico de grupo

Diagnóstico de grupo

Si activa el diagnóstico de grupo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar lectura de los valores del sensor

Descripción

Para la lectura de los valores del sensor existen las siguientes alternativas:

- Lectura libre de los valores del sensor
- Lectura síncrona de los valores del sensor
- Lectura de los valores del sensor en modo isócrono

La lectura libre o síncrona de los valores del sensor se selecciona en la parametrización con el parámetro "Detección". Este parámetro actúa únicamente en el modo no isócrono.

Obtendrá una lectura de los valores del sensor en modo isócrono cuando el 1SSI funcione en modo isócrono. El parámetro "Detección" no se evaluará en este caso.

La tabla siguiente muestra estas relaciones de la lectura de los valores del sensor.

En servicio	Parámetro "Detección"	Lectura de los valores del sensor
Modo no isócrono	libre	Lectura libre de los valores del sensor
	isócrono	Lectura síncrona de los valores del sensor
Modo isócrono	- (irrelevante)	Lectura de los valores del sensor en modo isócrono

Lectura libre de los valores del sensor:

En la lectura libre de los valores del sensor se obtiene la máxima precisión con la función de congelación del valor del encóder (Página 10876).

El 1SSI hace que se transfiera un telegrama cada vez que termine el tiempo monoestable (Página 10873) parametrizado.

El 1SSI procesa de forma asíncrona con respecto a estos telegramas libres el valor leído del encóder en el ciclo de su tasa de actualización.

De ahí resulta que si elegimos una lectura asíncrona de los valores del sensor obtendremos valores de distinta antigüedad. La diferencia entre la antigüedad máxima y mínima es el jitter.

Lectura síncrona de los valores del sensor

En la lectura síncrona de los valores del sensor se obtiene la máxima precisión con la lectura del valor del encóder.

El 1SSI inicia la transmisión de un telegrama en el ciclo de su tasa de actualización.

El 1SSI procesa el valor transferido del encóder de forma síncrona con su tasa de actualización.

Seleccionar el tipo de encóder

Tipo de encóder

El 1SSI soporta los siguientes tipos de encóder:

- Encóder absoluto (SSI) de 13 bits
  a
- Encóder absoluto (SSI) de 25 bits

Este tipo de encóder figura en los datos técnicos del encóder absoluto utilizado.

La selección "Ningún encóder" desactiva la entrada del encóder.

Seleccionar convertidor Gray/Dual

Convertidor Gray/Dual

El parámetro Convertidor Gray/Dual permite determinar el tratamiento del código suministrado por el encóder:

- En el ajuste "Gray", el valor suministrado por el encóder absoluto en código gray se convierte a código dual.
- En el ajuste "Dual" no se altera el valor procedente del encóder.

Seleccionar la velocidad de transferencia

Velocidad de transferencia

Con la "Velocidad de transferencia" se define la velocidad de la transferencia de datos del encóder SSI al 1SSI.

Seleccione la velocidad de transferencia deseada en la lista desplegable.

Observe que la velocidad de transferencia influye en la precisión y la vigencia de los valores del sensor.

La velocidad de transferencia máxima depende de la velocidad de transferencia:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 60 m → 500 kHz
- 20 m → 1 MHz
- 10 m → 1,5 MHz
- 8 m → 2 MHz

Consulte también

Seleccionar el temporizador monoestable (Página 10873)

Seleccionar la paridad

Paridad

Con la "Paridad" se define si para el encóder absoluto no debe transferirse ninguna paridad o si debe transferirse una paridad par o impar.

Un bit de paridad parametrizado también cuenta en el caso del parámetro "Tipo de encóder (Página 10869)". Si un encóder de 25 bits está parametrizado con paridad, se leen 26 bits del encóder.

El bit del encóder que sigue al LSB (Least significant Bit) se evalúa como un bit de paridad. Un error de paridad se notifica en la interfaz de realimentación (Página 10881) a través del bit ERR_SSI.

Seleccionar el temporizador monoestable

Tiempo monoestable

El tiempo monoestable es la pausa entre 2 telegramas SSI.

El tiempo monoestable parametrizado debe ser mayor que el tiempo monoestable del encóder absoluto utilizado (ver los datos técnicos del fabricante).

Para el tiempo monoestable del encóder absoluto se aplica la siguiente limitación:

(1 / velocidad de transferencia) < tiempo monoestable del encóder absoluto < 64 µs

Seleccione el tiempo monoestable correspondiente en la lista desplegable.

La especificación del tiempo monoestable es relevante para la lectura libre de los valores del sensor (Página 10868).

Consulte también

Seleccionar la velocidad de transferencia (Página 10869)

Activar la normalización

Valor del sensor y normalización

El valor transmitido por el encóder incluye la posición del encóder absoluto. Según el encóder utilizado, además de su posición se transmiten otros bits, situados delante y detrás de la posición del encóder.

Para que el 1SSI pueda determinar la posición del encóder, se debe especificar lo siguiente:

- Tipo de encóder (Página 10869)
- Número de bits a la derecha (Página 10874)
- Incrementos/Vuelta del encóder (Página 10875)

Con la normalización se especifica la representación del valor del encóder en la interfaz de realimentación (Página 10881).

- Con la normalización **activada** se especifica que los bits irrelevantes que siguen al valor del encóder sean desplazados.
- Con la normalización **desactivada**, se especifica que los bits a la derecha se conserven y estén disponibles para la evaluación.

Introducir número de bits a la derecha

Número de bits a la derecha

Para determinar la posición del encóder y para detectar correctamente el sentido de movimiento del encóder, el 1SSI necesita el número de bits a la derecha.

Este valor figura en los datos técnicos del encóder absoluto utilizado.

Activar inversión del sentido de giro

Detección del sentido e inversión del sentido de giro

Para detectar correctamente el sentido en el que se mueve el encóder, el 1SSI necesita la siguiente información:

- Tipo de encóder (Página 10869)
- Número de bits a la derecha (Página 10871)
- Incrementos/Vuelta del encóder (Página 10875)

El sentido de movimiento calculado se muestra en la interfaz de realimentación (Página 10881) y en los LEDs.

- LED UP: Cambio de posición del encóder del valor menor al valor mayor
- LED DN: Cambio de posición del encóder del valor mayor al valor menor

Con la **inversión del sentido de giro** se adapta el sentido del movimiento del encóder al del eje.

Se pueden realizar dos ajustes:

- **Desactivada:** El sentido de la posición transferida por el encóder se conserva.
- **Activada:** El sentido de la posición transferida del encóder se invierte. Es decir, aunque el encóder suministre valores ascendentes, se muestran valores descendentes.
  Esta inversión se refiere a los incrementos o a la vuelta del encóder que se han ajustado en los parámetros.

Ejemplo de inversión del sentido de giro

Preajustes:

Se utiliza un encóder monovuelta con 2^{10} (equivale a 10 bits) = 1024 incrementos / vuelta de encóder (resolución / 360°) con la siguiente parametrización:

- Tipo de encóder: SSI-13 Bit
- Número de bits a la derecha: 3 dígitos
- Inversión del sentido de giro: on
- Incrementos/vuelta del encóder: 1024

Valor del encóder antes de la inversión del sentido de giro: Posición del encóder registrada cíclicamente 1023.

Valor del encóder después de la inversión del sentido de giro: Posición visualizada del encóder 0

Introducir incrementos por vuelta del encóder

Incrementos por vuelta

El parámetro "Incrementos por vuelta del encóder" indica el número de incrementos que emite un encóder por cada vuelta.

Este valor figura en los datos técnicos del encóder absoluto utilizado.

Rangos de valores admisibles

Para los límites debe diferenciarse entre los distintos modelos de encóder. Como entrada se permiten sólo valores en potencias de 2.

- Tipo de encóder de 13 bits: 16 a 8192
- Tipo de encóder de 14 bits: 16 a 16384
- Tipo de encóder de 15 bits: 16 a 32768
- Tipo de encóder de 16 bits: 16 a 65536
- Tipo de encóder de 17 bits: 16 a 131072
- Tipo de encóder de 18 bits: 16 a 262144
- Tipo de encóder de 19 bits: 16 a 524288
- Tipo de encóder de 20 bits: 16 a 1048576
- Tipo de encóder de 21 bits: 16 a 2097152
- Tipo de encóder de 22 bits: 16 a 4194304
- Tipo de encóder de 23 bits: 16 a 8388608
- Tipo de encóder de 24 bits: 16 a 16777216
- Tipo de encóder de 25 bits: 16 a 33554432

Seleccionar congelación del valor del encóder

Congelación del valor del encóder

Con la función de congelación se congela en un flanco de la entrada digital (DI) el valor de encóder actual del 1SSI.

El valor del encóder puede evaluarse así en función de los eventos.

Un valor de encóder congelado está identificado mediante el bit 31 activado y se mantiene hasta que finaliza la función de congelación.

El valor de encóder congelado se introduce en la interfaz de realimentación (Página 10881) en el punto del valor que normalmente de registra de forma cíclica y se dota de la identificación "Bit 31 activado".

Nota

Con el valor del encóder congelado también se realiza una detección del sentido y una función de comparación, así como una monitorización de errores.

Requisitos para utilizar la función de congelación

A través de la parametrización se selecciona el flanco que congela el valor del encóder en la entrada digital.

- Flanco descendente en la entrada DI
- Flanco ascendente en la entrada DI
- Ambos flancos en la entrada DI

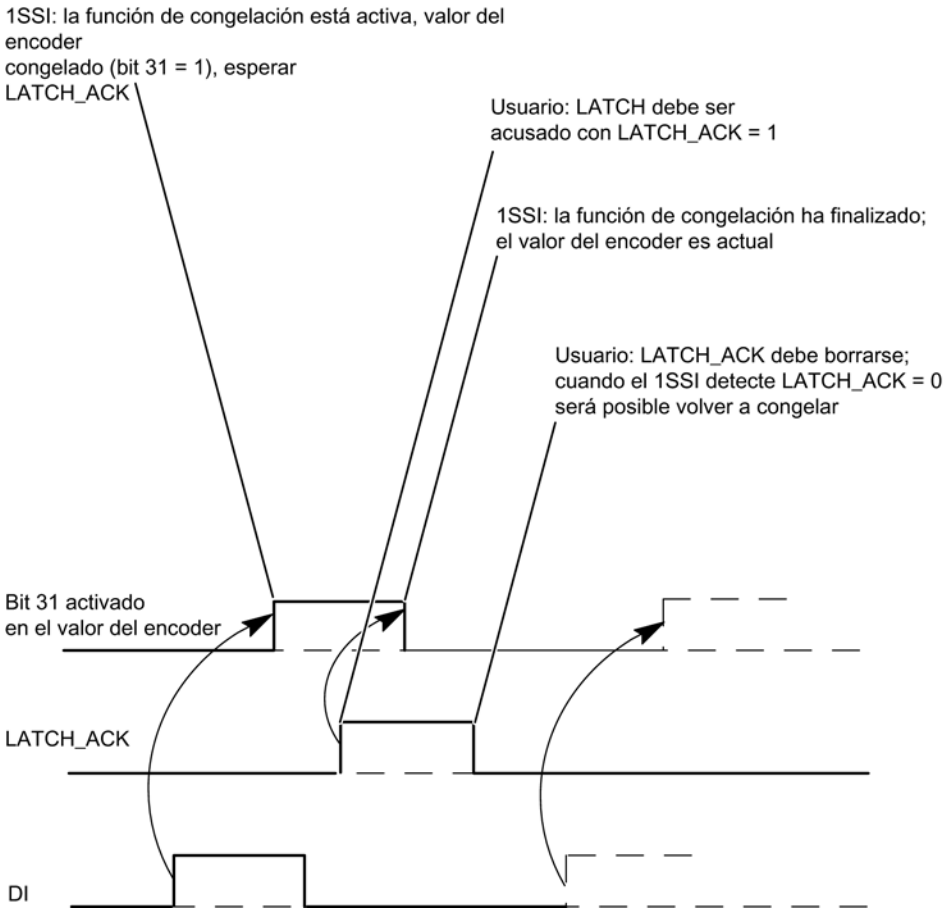
Con la selección de flancos se activa la función de congelación que está acoplada en la entrada digital.

Con la selección "Inactivo" no es posible realizar la congelación del valor del sensor.

Conclusión de la función de congelación

La función de congelación debe confirmarse. Cuando el programa de control confirma la adopción del valor del encóder, el bit 31 se elimina y el valor del encóder se actualiza de nuevo. A continuación puede volver a realizar una congelación.

La figura siguiente muestra el desarrollo de la función de congelación.



Seleccionar comportamiento de los comparadores

Ajuste del comparador

La posición registrada del encóder puede compararse hasta con dos valores que pueden cargarse. Los dos resultados de la comparación se almacenan en la interfaz de realimentación (Página 10881). Sólo después de cargar el valor de comparación, se activa el comparador correspondiente.

Los dos comparadores se ajustan en los parámetros "Comparador 1" y "Comparador 2".

Ajuste	Efecto sobre el resultado de la comparación (CMPx)
Inactivo	El valor del encóder no se compara. El bit de realimentación CMPx = 0.
Adelante	El valor del encóder se compara en sentido ascendente (UP). • Si el valor del encóder ≥ al valor de comparación, el bit de realimentación CMPx = 1. • Si el valor del encóder ≥ al valor de comparación, el bit de realimentación CMPx = 0. • Si el sentido es descendente, el bit de realimentación CMPx no cambia. • Si no se detecta ningún cambio en el valor del encóder, el bit de realimentación CMPx no cambia.
Atrás	El valor del encóder se compara en sentido descendente (DN). • Si el valor del encóder ≤ al valor de comparación, el bit de realimentación CMPx = 1. • Si el valor del encóder > al valor de comparación, el bit de realimentación CMPx = 0. • Si el sentido es ascendente, el bit de realimentación CMPx no cambia. • Si no se detecta ningún cambio en el valor del encóder, el bit de realimentación CMPx no cambia.
En ambos sentidos	El valor del encóder se compara en ambos sentidos. Si el sentido es ascendente, se aplican las siguientes condiciones: • Si el valor del encóder ≥ al valor de comparación, el bit de realimentación CMPx = 1. • Si el valor del encóder ≥ al valor de comparación, el bit de realimentación CMPx = 0. • Si no se detecta ningún cambio en el valor del encóder, el bit de realimentación CMPx no cambia. Si el sentido es descendente, se aplican las siguientes condiciones: • Si el valor del encóder ≤ al valor de comparación, el bit de realimentación CMPx = 1. • Si el valor del encóder > que el valor de comparación, el bit de realimentación CMPx = 0. • Si no se detecta ningún cambio en el valor del encóder, el bit de realimentación CMPx no cambia.

En cuanto se carga un valor de comparación, se borra el resultado de comparación y se registra el nuevo resultado de la comparación según el ajuste del sentido.

Nota

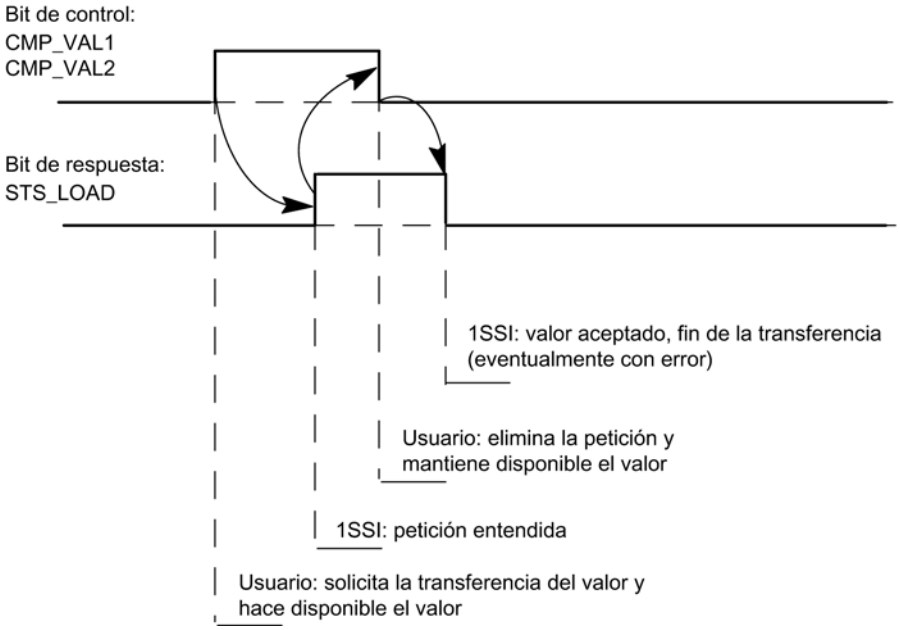
No puede haber nunca más de un bit de control activado al mismo tiempo:

CMP_VAL1 o bien CMP_VAL2.

De no ser así, se notificará el error ERR_LOAD hasta que se borren ambos bits de control.

Cargar valor de comparación

En la figura siguiente se muestra el proceso de una transferencia de valores.



Comparador en el modo isócrono

En el modo isócrono, los valores de comparación se cargan en el instante T_o y tienen efecto a partir del instante T_i en el mismo ciclo de bus.

Activar señal de vida (modo estándar)

Señal de vida

El parámetro sólo es activo en modo isócrono.

Si la señal de vida está activada, en el modo isócrono se invierte el bit de la señal de vida cada vez que se lee un valor del encóder, es decir, el último valor enviado se invierte. Si se ha parametrizado un factor de ciclo mediante el parámetro "Tasa de muestreo encóder (Página 10879)", el valor sólo se invertirá si realmente se ha leído un valor del encóder.

El bit de señal de vida se encuentra en la interfaz de realimentación (Página 10881) en el byte 4 / bit 7.

Ajustar la tasa de muestreo del encóder

Tasa de muestreo del encóder

Este parámetro sólo es efectivo en modo isócrono.

Aquí se ajusta una tasa de muestreo del encóder que, dado el caso, debe tenerse en cuenta. Con ello es posible utilizar encóders más lentos (p. ej. encóders ultrasónicos) en un ciclo de procesamiento rápido.

A partir de la frecuencia ajustada se calcula un factor de ciclo interno n. En este caso, el encóder sólo se leerá cada n ciclos de reloj.

Ejemplo

- Ciclo de procesamiento de 500 µs
- Tasa de muestreo del encóder: 1,2 kHz (a 833 µs aprox.)

Esto da lugar a un factor de ciclo n = 2.

Así pues, el encóder sólo se lee de nuevo cada dos ciclos de procesamiento, es decir, cada milisegundo.

Interfaz de control y respuesta para el modo estándar

Interfaz para el modo estándar

Nota

Para el 1SSI, los datos siguientes de la interfaz de control son datos relacionados o, lo que es lo mismo, coherentes:

- Bytes 0 … 3
- Bytes 4 … 7

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (las salidas) del 1SSI en el modo estándar.

Dirección	Ocupación		Nombre
Bytes 0 a 3	Valor de comparación 1 ó 2 (palabra doble)		
Byte 4	Bit 7	Acuse del error	EXTF_ACK
	Bit 6	Confirmación de la función de congelación	LATCH_ACK
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	reservado = 0	
	Bit 2	reservado = 0	
	Bit 1	Carga del valor de comparación 2	CMP_VAL2
	Bit 0	Carga del valor de comparación 1	CMP_VAL1
Byte 5	reservado = 0		
Bytes 6 a 7	reservado = 0		

Significado de los bits de control

Bit	Significado
CMP_VAL1	Carga del valor de comparación 1
CMP_VAL2	Carga del valor de comparación 2
EXTF_ACK	Acuse de error para el "Error encóder absoluto" ERR_SSI y "Cortocircuito en la alimentación del encóder" ERR_24V.
LATCH_ACK	Acuse para la función de congelación

Interfaz de realimentación para el modo estándar

Nota

Para el 1SSI, los datos siguientes de la interfaz de realimentación son datos relacionados, es decir, coherentes:

- Byte 0...3
- Byte 4...7

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 1SSI en el modo estándar.

Dirección	Ocupación		Nombre
Bytes 0 a 3	Valor del encóder (palabra doble) (bit 31 activado, valor del encóder congelado)		
Byte 4	Bit 7	reservado = 0 o señal de vida	LZ
	Bit 6	Listo para servicio	RDY
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	Error encóder absoluto	ERR_SSI
	Bit 3	Cortocircuito en la alimentación del encóder	ERR_24V
	Bit 2	Estado de DI	STS_DI
	Bit 1	Estado DN	STS_DN
	Bit 0	Estado UP	STS_UP
Byte 5	Bit 7	reservado = 0	
	Bit 6	reservado = 0	
	Bit 5	reservado = 0	
	Bit 4	reservado = 0	
	Bit 3	Valor de comparación 2 alcanzado	CMP2
	Bit 2	Valor de comparación 1 alcanzado	CMP
	Bit 1	Error en función de carga	ERR_LOAD
	Bit 0	Función de carga en curso	STS_LOAD
Bytes 6 a 7	reservado = 0		

Nota sobre los bits de realimentación

Bit	Significado
CMP	Resultado de comparación del comparador 1
CMP2	Resultado de comparación del comparador 2
ERR_24V	La alimentación del encóder está cortocircuitada. El ERR_24V se desactiva cuando se elimina el cortocircuito y se acusa con el bit de control EXTF_ACK.
ERR_LOAD	Error al cargar los valores de comparación porque ambos bits de control CMP_VAL1 y CMP_VAL2 están activados.
ERR_PARA	Hay parámetros erróneos en la parametrización de la estación ET 200S. Causa: Los incrementos o la vuelta del encóder no se encuentran en el rango de valores permitido para el tipo de encóder. El bit de parámetro se borra cuando se transfiere una parametrización correcta.
ERR_SSI	El 1SSI detecta el "Error encóder absoluto" cuando fallan los telegramas en la interfaz SSI. Causas: no hay ningún encóder conectado; rotura de hilo del encóder; subtensión en la alimentación de encóders; el tipo de encóder, la velocidad de transferencia, el error de paridad y el temporizador monoestable no se corresponden con los del encóder conectado; los encóders programables no se corresponden con los ajustes del 1SSI; el encóder está defectuoso o existen fallos de paridad. El ERR_SSI se desactiva cuando se elimina la causa del error y se acusa con el bit de control EXTF_ACK.
LZ	El parámetro sólo es activo en modo isócrono. Si la señal de vida está activada, en el modo isócrono se invierte el bit de la señal de vida cada vez que se lee un valor del encóder, es decir, el último valor enviado se invierte. Si se ha parametrizado un factor de ciclo mediante el parámetro "Tasa de muestreo encóder", el valor sólo se invertirá si realmente se ha leído un valor del encóder.
STS_DI	El bit muestra el estado de la entrada digital DI.
STS_DN	Estado de sentido descendente; cuando cambia el valor del encóder de posiciones más altas a posiciones más bajas (inc. paso por cero)
STS_LOAD	Bit de respuesta de CMP_VAL1 o CMP_VAL2. El 1SSI muestra con este bit que se carga un valor de comparación.
STS_UP	Estado de sentido ascendente; cuando cambia el valor del encóder de posiciones más bajas a posiciones más altas (incl. paso por cero)
RDY	El 1SSI está parametrizado correctamente y ejecuta sus funciones. Las respuestas mostradas son válidas. Con el error el encóder absoluto también está activado ERR_SSI.

Parametrizar para el modo Fast

Activar diagnóstico de grupo

Diagnóstico de grupo

Si activa el diagnóstico de grupo en su parametrización, se realiza un diagnóstico relativo al canal.

Seleccionar lectura de los valores del sensor

Descripción

El encóder absoluto transfiere sus valores de encóder en telegramas al 1SSI. La transferencia de los telegramas es iniciada por el 1SSI. Para la lectura de los valores del sensor existen las siguientes alternativas:

- Lectura libre de los valores del sensor
- Lectura síncrona de los valores del sensor
- Lectura de los valores del sensor en modo isócrono

La lectura libre o síncrona de los valores del sensor se selecciona en la parametrización con el parámetro "Detección". Este parámetro actúa únicamente en el modo no isócrono.

Obtendrá una lectura de los valores del sensor en modo isócrono cuando el 1SSI funcione en modo isócrono. El parámetro "Detección" no se evalúa en este caso.

La tabla siguiente muestra estas relaciones de la lectura de los valores del sensor.

En servicio	Parámetro "Detección"	Lectura de los valores del sensor
Modo no isócrono	libre	Lectura libre de los valores del sensor
	isócrono	Lectura síncrona de los valores del sensor
Modo isócrono	- (irrelevante)	Lectura de los valores del sensor en modo isócrono

Lectura libre de los valores del sensor:

El 1SSI hace que se transfiera un telegrama cada vez que termine el tiempo monoestable (Página 10885) parametrizado.

El 1SSI procesa de forma asíncrona con respecto a estos telegramas libres el valor leído del encóder en el ciclo de su tasa de actualización.

De ahí resulta que si elegimos una lectura asíncrona de los valores del sensor obtendremos valores de distinta antigüedad. La diferencia entre la antigüedad máxima y mínima es el jitter.

Lectura síncrona de los valores del sensor

En la lectura síncrona de los valores del sensor se obtiene la máxima precisión con la lectura del valor del encóder.

El 1SSI inicia la transmisión de un telegrama en el ciclo de su tasa de actualización.

El 1SSI procesa el valor transferido del encóder de forma síncrona con su tasa de actualización.

Lectura de los valores del sensor en modo isócrono

La lectura isócrona de los valores del sensor se ajusta automáticamente cuando el sistema de bus y el módulo periférico funcionan de forma sincronizada.

El 1SSI dispone la transferencia de un telegrama en cada ciclo de bus en el instante T_i siempre que la tasa máxima de muestreo del encóder no provoque un factor de ciclo.

El 1SSI procesa el valor transferido del encóder de forma síncrona con el ciclo de bus.

Seleccionar el tipo de encóder

Tipo de encóder

El 1SSI soporta los siguientes tipos de encóder:

- Encóder absoluto (SSI) de 13 bits
  a
- Encóder absoluto (SSI) de 25 bits

Este tipo de encóder figura en los datos técnicos del encóder absoluto utilizado.

La selección "Ningún encóder" desactiva la entrada del encóder.

Seleccionar convertidor Gray/Dual

Convertidor Gray/Dual

El parámetro Convertidor Gray/Dual permite determinar el tratamiento del código suministrado por el encóder:

- En el ajuste "Gray", el valor suministrado por el encóder absoluto en código gray se convierte a código dual.
- En el ajuste "Dual" no se altera el valor procedente del encóder.

Seleccionar la velocidad de transferencia

Velocidad de transferencia

Con la "Velocidad de transferencia" se define la velocidad de la transferencia de datos del encóder SSI al 1SSI.

Seleccione la velocidad de transferencia deseada en la lista desplegable.

Observe que la velocidad de transferencia influye en la precisión y la vigencia de los valores del sensor.

La velocidad de transferencia máxima depende de la velocidad de transferencia:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 60 m → 500 kHz
- 20 m → 1 MHz

- 10 m → 1,5 MHz
- 8 m → 2 MHz

Consulte también

Seleccionar el temporizador monoestable (Página 10885)

Seleccionar paridad

Paridad

Con la "Paridad" se define si para el encóder absoluto no debe transferirse ninguna paridad o si debe transferirse una paridad par o impar.

Un bit de paridad parametrizado también cuenta en el caso del parámetro "Tipo de encoder (Página 10881)". Si un encóder de 25 bits está parametrizado con paridad, se leen 26 bits del encóder.

El bit del encóder que sigue al LSB (Least significant Bit) se evalúa como un bit de paridad. Un error de paridad se notifica en la interfaz de realimentación (Página 10889) a través del bit EXTF.

Seleccionar el temporizador monoestable

Tiempo monoestable

El tiempo monoestable es la pausa entre 2 telegramas SSI.

El tiempo monoestable parametrizado debe ser mayor que el tiempo monoestable del encóder absoluto utilizado (ver los datos técnicos del fabricante).

Para el tiempo monoestable del encóder absoluto se aplica la siguiente limitación:

(1 / velocidad de transferencia) < tiempo monoestable del encóder absoluto < 64 µs

Seleccione el tiempo monoestable correspondiente en la lista desplegable.

La especificación del tiempo monoestable es relevante para la lectura libre de los valores del sensor (Página 10880).

Consulte también

Seleccionar la velocidad de transferencia (Página 10881)

Activar la normalización

Valor del sensor y normalización

El valor transmitido por el encóder incluye la posición del encóder absoluto. Según el encóder utilizado, además de su posición se transmiten otros bits, situados delante y detrás de la posición del encóder.

Para que el 1SSI pueda determinar la posición del encóder, se debe especificar lo siguiente:

- Tipo de encóder (Página 10881)
- Número de bits a la derecha (Página 10886)
- Incrementos/Vuelta del encóder (Página 10887)

Con la normalización se especifica la representación del valor del encóder en la interfaz de realimentación (Página 10889).

- Con la normalización **activada** se especifica que los bits irrelevantes que siguen al valor del encóder sean desplazados.
- Con la normalización **desactivada**, se especifica que los bits a la derecha se conserven y estén disponibles para la evaluación.

Introducir número de bits a la derecha

Número de bits a la derecha

Para determinar la posición del encóder y para detectar correctamente el sentido de movimiento del encóder, el 1SSI necesita el número de bits a la derecha.

Este valor figura en los datos técnicos del encóder absoluto utilizado.

Activar inversión del sentido de giro

Detección del sentido e inversión del sentido de giro

Para detectar correctamente el sentido en el que se mueve el encóder, el 1SSI necesita la siguiente información:

- Tipo de encóder (Página 10881)
- Número de bits a la derecha (Página 10883)
- Incrementos/Vuelta del encóder (Página 10887)

El sentido de movimiento calculado se muestra en la interfaz de realimentación (Página 10889) y en los LEDs.

- LED UP: Cambio de posición del encóder del valor menor al valor mayor
- LED DN: Cambio de posición del encóder del valor mayor al valor menor

Con la **inversión del sentido de giro** se adapta el sentido del movimiento del encóder al del eje.

Se pueden realizar dos ajustes:

- **Desactivada:** El sentido de la posición transferida por el encóder se conserva.
- **Activada:** El sentido de la posición transferida del encóder se invierte. Es decir, aunque el encóder suministre valores ascendentes, se muestran valores descendentes.
  Esta inversión se refiere a los incrementos o a la vuelta del encóder que se han ajustado en los parámetros.

Ejemplo de inversión del sentido de giro

Preajustes:

Se utiliza un encóder monovuelta con 2^{10} (equivale a 10 bits) = 1024 incrementos / vuelta de encóder (resolución / 360°) con la siguiente parametrización:

- Tipo de encóder: SSI-13 Bit
- Número de bits a la derecha: 3 dígitos
- Inversión del sentido de giro: on
- Incrementos/vuelta del encóder: 1024

Valor del encóder antes de la inversión del sentido de giro: Posición del encóder registrada cíclicamente 1023.

Valor del encóder después de la inversión del sentido de giro: Posición visualizada del encóder 0

Introducir incrementos por vuelta del encóder

Incrementos por vuelta

El parámetro "Incrementos por vuelta del encóder" indica el número de incrementos que emite un encóder por cada vuelta.

Este valor figura en los datos técnicos del encóder absoluto utilizado.

Rangos de valores admisibles

Para los límites debe diferenciarse entre los distintos modelos de encóder. Como entrada se permiten sólo valores en potencias de 2.

- Tipo de encóder de 13 bits: 16 a 8192
- Tipo de encóder de 14 bits: 16 a 16384
- Tipo de encóder de 15 bits: 16 a 32768
- Tipo de encóder de 16 bits: 16 a 65536
- Tipo de encóder de 17 bits: 16 a 131072
- Tipo de encóder de 18 bits: 16 a 262144
- Tipo de encóder de 19 bits: 16 a 524288
- Tipo de encóder de 20 bits: 16 a 1048576

- Tipo de encóder de 21 bits: 16 a 2097152
- Tipo de encóder de 22 bits: 16 a 4194304
- Tipo de encóder de 23 bits: 16 a 8388608
- Tipo de encóder de 24 bits: 16 a 16777216
- Tipo de encóder de 25 bits: 16 a 33554432

Activar señal de vida (modo Fast)

Señal de vida

El parámetro sólo es activo en modo isócrono.

Si la señal de vida está activada, en el modo isócrono se invierte el bit de la señal de vida cada vez que se lee un valor del encóder, es decir, el último valor enviado se invierte. Si se ha parametrizado un factor de ciclo mediante el parámetro "Tasa de muestreo encóder (Página 10888)", el valor sólo se invertirá si realmente se ha leído un valor del encóder.

El bit de señal de vida se encuentra en la interfaz de realimentación (Página 10889) en el byte 0 a 3 / bit 31.

Ajustar la tasa de muestreo del encóder

Tasa de muestreo del encóder

Este parámetro sólo es efectivo en modo isócrono.

Aquí se ajusta una tasa de muestreo del encóder que, dado el caso, debe tenerse en cuenta. Con ello es posible utilizar encóders más lentos (p. ej. encóders ultrasónicos) en un ciclo de procesamiento rápido.

A partir de la frecuencia ajustada se calcula un factor de ciclo interno n. En este caso, el encóder sólo se leerá cada n ciclos de reloj.

Ejemplo

- Ciclo de procesamiento de 500 µs
- Tasa de muestreo del encóder: 1,2 kHz (a 833 µs aprox.)

Esto da lugar a un factor de ciclo n = 2.

Así pues, el encóder sólo se lee de nuevo cada dos ciclos de procesamiento, es decir, cada milisegundo.

Interfaz de realimentación para el modo Fast

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del 1SSI en el modo Fast.

Dirección	Ocupación		Nombre
Bytes 0 a 3	Bit 31	reservado = 0 o señal de vida	LZ
	Bit 30	listo para funcionar (las respuestas son válidas)	RDY
	Bit 29	Error de parametrización	ERR_PARA
	Bit 28	Error de grupo Encóder absoluto o cortocircuito de la alimentación del encóder	EXTF
	Bit 27	Estado de DI	STS_DI
	Bit 26	Estado DN	STS_DN
	Bit 25	Estado UP	STS_UP
	Bits 0 a 24	Valor del encóder	

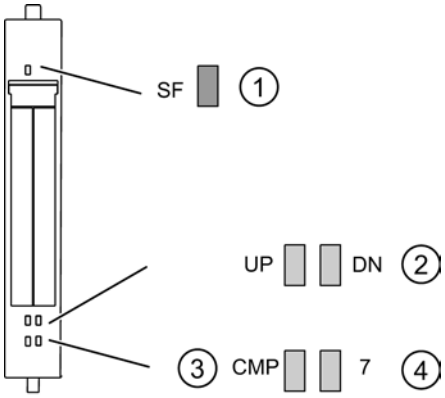
Nota sobre los bits de realimentación

Bit	Significado
ERR_PARA	Hay parámetros erróneos en la parametrización de la estación ET 200S. Causa: Los incrementos o la vuelta del encóder no se encuentran en el rango de valores permitido para el tipo de encóder. El bit de parámetro se borra cuando se transfiere una parametrización correcta.
EXTF	Error de grupo Encóder absoluto o cortocircuito de la alimentación del encóder Causas: • La alimentación del encóder está cortocircuitada. • no hay ningún encóder conectado; rotura de hilo del encóder; el tipo de encóder, la velocidad de transferencia y el tiempo monoestable no se corresponden con los del encóder conectado; los encóders programables no se corresponden con los ajustes del 1SSI; el encóder está defectuoso o existen fallos o errores de paridad. EXTF se desactiva cuando se eliminan las causas del error.
LZ	El parámetro sólo es activo en modo isócrono. Si la señal de vida está activada, en el modo isócrono se invierte el bit de la señal de vida cada vez que se lee un valor del encóder, es decir, el último valor enviado se invierte. Si se ha parametrizado un factor de ciclo mediante el parámetro "Tasa de muestreo encóder", el valor sólo se invertirá si realmente se ha leído un valor del encóder.
STS_DI	El bit muestra el estado de la entrada digital DI.
STS_DN	Estado de sentido descendente; cuando cambia el valor del encóder de posiciones más altas a posiciones más bajas (incl. paso por cero)
STS_UP	Estado de sentido ascendente; cuando cambia el valor del encóder de posiciones más bajas a posiciones más altas (incl. paso por cero)
RDY	El 1SSI está parametrizado correctamente y ejecuta sus funciones. Las respuestas mostradas son válidas. Con el error Encóder absoluto también está activado EXTF.

Diagnóstico del 1SSI

Diagnóstico con indicadores LED

Indicador LED del 1SSI



① Error de grupo (rojo)
② Indicador de estado de cambio de valor del sensor (verde)
③ Indicador de estado del resultado de la comparación
④ Indicador de estado de la entrada digital (verde)

Indicadores de estado y error mediante los LEDs del 1SSI

La tabla muestra los indicadores de estado y error del 1SSI.

Evento (LED)					Causa	Solución
SF	UP	DN	CMP	7		
ON					No hay parametrización. Hay un aviso de diagnóstico.	Compruebe la parametrización. Evalúe el aviso de diagnóstico.
	ON				Cuando cambia el valor del sensor de un valor menor a uno mayor (incl. paso por cero)	
		ON			Cuando cambia el valor del sensor de un valor mayor a uno menor (incl. paso por cero)	
			ON		con resultado de comparación CMP 1 activado	
				ON	DI (latch) activada	

Tipos de error

En el manual de producto del módulo de interfaz utilizado en la estación ET 200S encontrará información sobre la estructura del diagnóstico de canal.

Tipos de error del 1SSI

La tabla muestra los tipos de error del 1SSI.

Tipo de error		Significado	Solución
1_D	00001: Cortocircuito	Cortocircuito en la alimentación del encoder absoluto.	Corregir el cableado del proceso.
9_D	01001: Error	Ha aparecido un error interno en el módulo.	Sustituir el módulo.
		Tensión de carga insuficiente del módulo de potencia.	Corregir el cableado del proceso. Compruebe la tensión de carga.
16_D	10000: Error de parametrización	Módulo no parametrizado.	Corregir la parametrización.
26_D	11010: Error externo	Error de bit de arranque/paro (error de encóder absoluto): Rotura de hilo en el cable del encóder o cable no conectado. El tipo de encóder, la velocidad de transferencia y el tiempo monoestable no se corresponden con el encóder conectado. Los encóders programables no se corresponden con los ajustes en el 1SSI El sensor está defectuoso o hay interferencias.	Sustituir el encóder. Corregir el cableado del proceso Corregir la parametrización

11.4.2.4 Principios básicos del posicionamiento con módulos de función

¿Qué es posicionar?

Posicionamiento

Posicionar consiste en adoptar una posición espacial concreta, en particular, definida mediante coordenadas.

En la tecnología SIMATIC, entendemos por posicionar el ajuste y el posicionamiento de ejes mecánicos. Para ello se utilizan diversos procedimientos:

- Posicionamiento controlado con marcha rápida y lenta
- Posicionamiento con regulación de posición
- Posicionamiento con motor paso a paso

Pueden controlarse los ejes lineales con una zona de desplazamiento limitada y ejes rotativos de giro sin fin.

Posicionamiento controlado con marcha rápida y lenta

Posicionamiento controlado

Todos los procesos de posicionamiento están caracterizados por:

- una posición inicial
- el destino en el que se realizará el posicionamiento
- los parámetros que determinan el transcurso del posicionamiento

En primer lugar se realiza una aproximación al punto de destino a una velocidad mayor, la marcha rápida. Al llegar a una distancia predeterminada de la posición de destino, se cambia a una velocidad inferior, la marcha lenta. Poco antes de que el eje llegue al punto de destino, también a una distancia predefinida con respecto al punto de destino, el accionamiento se desconecta. En este proceso el módulo vigila la aproximación al destino.

El accionamiento se controla mediante salidas digitales con marcha rápida o marcha lenta y en el sentido correspondiente.

Zonas y puntos de conexión del módulo de posicionamiento

Objetivo

Define la posición absoluta o relativa del eje a la que se llega en el posicionamiento.

Definición de los puntos y las zonas de conexión

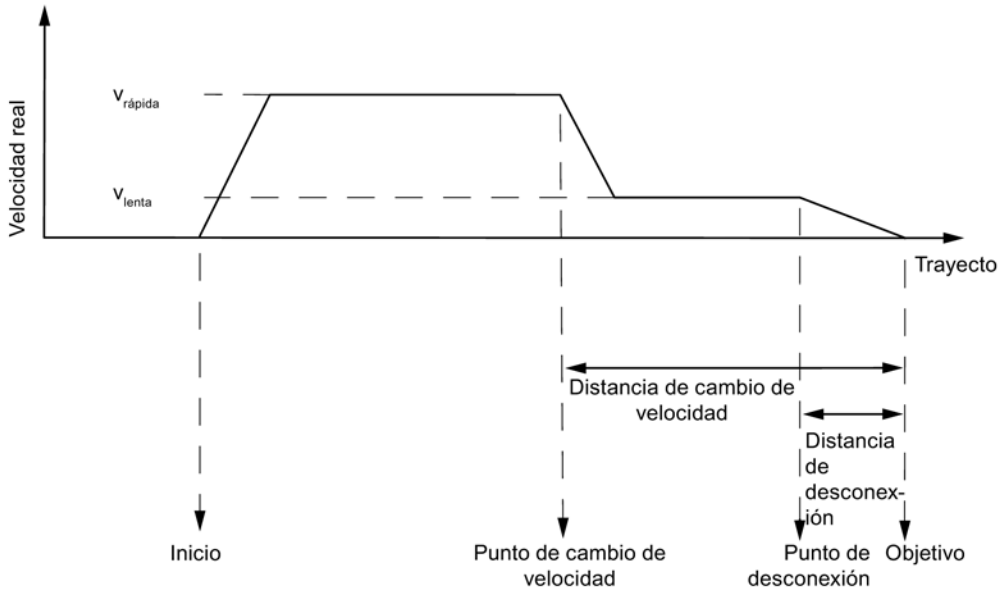
Para el posicionamiento controlado pueden parametrizarse las zonas y las posiciones siguientes:

Zona / Posición	Explicación
Zona de trabajo	Define el área que se determina para la tarea en cuestión con ayuda de los finales de carrera por software o con el final del eje rotativo.
Distancia de cambio de velocidad	Define la distancia con respecto al punto de destino en el que el accionamiento cambia la velocidad de marcha rápida a marcha lenta.
Punto de cambio de velocidad	Define la posición en la que el accionamiento cambia la velocidad de marcha rápida a marcha lenta.
Distancia de desconexión	define la distancia hasta el punto de destino en el que se desconecta el accionamiento.
Punto de desconexión	Define la posición en la que el accionamiento se desconecta. El módulo de posicionamiento retoma las funciones de vigilancia a partir de este punto.

Zona / Posición	Explicación
Zona de destino	Define la precisión de posicionamiento de su aplicación y se sitúa de forma simétrica en torno al destino.
Zona de parada	Define una zona simétrica en torno al destino que es vigilada por el módulo de posicionamiento.

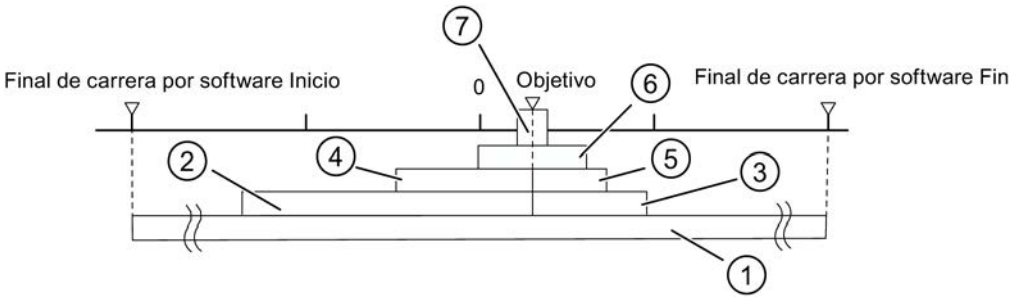
Puntos y distancias de conmutación

La figura siguiente muestra la forma en la que pueden disponerse los puntos de conexión y las diferencias de conexión para un proceso de posicionamiento. Para simplificar, se presupone que la velocidad real cambia de forma lineal a lo largo del trayecto recorrido. Las rampas surgidas se explican por la inercia mecánica o por las posibilidades de parametrización de la etapa de potencia.



Áreas de conexión

La figura siguiente muestra la forma en la que pueden disponerse las zonas de conexión en torno al destino.



① Zona de trabajo
② Distancia de cambio de velocidad en el sentido de marcha positivo
③ Distancia de cambio de velocidad en el sentido de marcha negativo
④ Distancia de desconexión en el sentido de marcha positivo
⑤ Distancia de desconexión en el sentido de marcha negativo
⑥ Zona de parada
⑦ Zona de destino

Señales del encóder

Encóders compatibles

Encóders incrementales conectables

Se soportan encóders incrementales (Página 10896) de dos impulsos con un desfase de 90º con o sin marca cero:

- Encóder con señales de salida asimétricas con nivel de 24 V
  - Frecuencia límite = 50 kHz
  - Longitud máx. 100 m de cable apantallado
- Encóder con señales de salida simétricas con un puerto diferencial de 5 V según RS 422.
  - Frecuencia límite = 400 kHz
  - Con una alimentación de tensión de 5 V: Longitud del cable: máx. 32 m, apantallada
  - Con una alimentación de tensión de 24 V: Longitud del cable: máx. 100 m, apantallada

Nota

Si el encóder de 5 V no emite ninguna señal de marca cero y se ha activado la comprobación de rotura de hilo, es preciso interconectar externamente las entradas de marca cero N y /N para que estas entradas presenten un nivel distinto (p. ej. N a 5 V, /N a la masa).

Encóders absolutos conectables

Se soportan encóders absolutos (Página 10897) con interfaz serie. La transferencia de la información de recorrido se realiza de forma síncrona según el protocolo SSI (**i**nterfaz **s**erial**s**íncrona). El FM x51 soporta sólo el código GRAY. Según la disposición de los bits de datos en el telegrama resultan los formatos de datos de 25 bits (abeto) y 13 bits (medio abeto).

Entre los encóders absolutos se distingue entre

- Encóder monovuelta
  Los encóders monovuelta reflejan el rango completo de medida en una vuelta del encóder.
- Encóder multivuelta
  Los encóders multivuelta reflejan el rango completo de medida en varias vueltas del encóder.

Tipo de encóder	Longitud de telegrama
Encóder monovuelta	13 bits
Encóder monovuelta	25 bits
Encóder multivuelta	13 bits … 25 bits

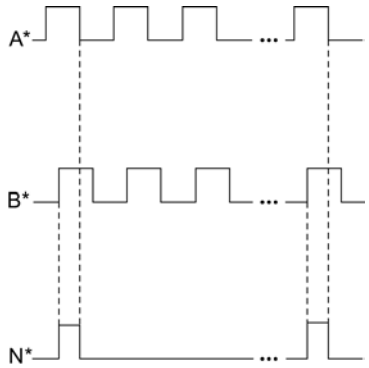
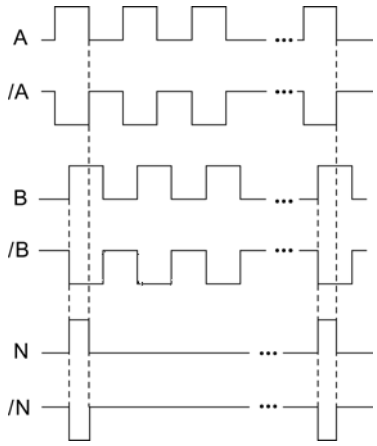
La velocidad de la transferencia de datos depende de la longitud del cable:

- Máx. 188 kHz con 200 m longitud de cable apantallado
- Máx. 375 kHz con 100 m longitud de cable apantallado
- Máx. 750 kHz con 40 m longitud de cable apantallado
- Máx. 1,5 MHz con 12 m longitud de cable apantallado

Encóder incremental

Formas de señal

En la table que sigue a continuación se muestran las formas de señal de los encóders con señales de salida asimétricas y simétricas.

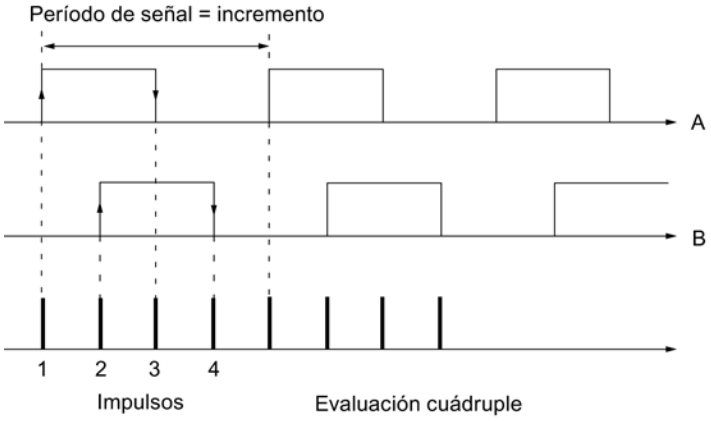
Asimétrica	Simétrica
	

Interpretación de la señal

Incrementos

Un incremento caracteriza un período de señal de las dos señales A y B de un encóder. Este valor se indica en los datos técnicos del encóder y/o en su placa de características.

La figura siguiente muestra la relación de incrementos e impulsos.



Impulsos

El módulo de posicionamiento evalúa los 4 flancos de las señales A y B de cada incremento (evaluación cuádruple).

Impulsos
1 incremento (dado por el encóder) = 4 impulsos (evaluación en el módulo)

Tiempos de respuesta

El módulo de posicionamiento tiene los siguientes tiempos de respuesta para encóders incrementales conectados:

Tiempos de respuesta
Tiempo de respuesta = tiempo de conexión de los elementos conectados

Nota

El tiempo mínimo de respuesta se puede compensar mediante una parametrización adecuada de la distancia de cambio de velocidad y la distancia de desconexión.

Imprecisión

La imprecisión afecta a la exactitud del posicionamiento. En los encóders incrementales la imprecisión puede ignorarse.

Encóder absoluto

Evaluación de impulsos del encóder absoluto

Evaluación de impulsos del encóder absoluto
1 incremento (dado por el encóder) = 1 impulso (evaluación FM)

Tiempos de respuesta

El FM x51 tiene los tiempos de respuesta siguientes para encóders absolutos:

Tiempos de respuesta
Tiempo mínimo de respuesta = tiempo de ejecución de telegramas + tiempo de conexión de los elementos conectados
Tiempo máximo de respuesta = 2 × tiempo de ejecución de telegramas + tiempo monoestable + tiempo de conexión de los elementos conectados
En el caso de encóders absolutos programables: Tiempo máximo de respuesta = tiempo de ejecución de telegramas + tiempo monoestable + tiempo de conexión de los elementos conectados + 1/frecuencia de pasos máx.

Tiempo monoestable

El tiempo monoestable es de 64 µs.

No se permiten encóders con valores superiores a los límites que aquí se indican.

Tiempos de ejecución de telegramas

Los tiempos de ejecución de telegramas dependen de la velocidad de transferencia:

Velocidad de transferencia	Tiempo de ejecución de telegramas a 13 bits	Tiempo de ejecución de telegramas a 25 bits
0,188 MHz	75 µs	139 µs
0,375 MHz	38 µs	70 µs
0,750 MHz	19 µs	35 µs
1,500 MHz	10 µs	18 µs

Ejemplo de tiempos de respuesta

Este ejemplo muestra cómo calcular el tiempo de respuesta máximo y mínimo. En el ejemplo no se emplea ningún encóder programable.

- Tiempo de conexión del hardware: aprox. 150 µs
- Tiempo de ejecución de telegramas: 18 µs a una velocidad de transferencia de 1,5 MHz (telegramas de 25 bits)
- Tiempo monoestable: 64 µs

Tiempo de respuesta mínimo = 18 µs + 150 µs = 168 µs

Tiempo de respuesta máximo = 2 × 18 µs + 64 µs + 150 µs = 250 µs

Nota

El tiempo mínimo de respuesta se puede compensar mediante una parametrización adecuada de la distancia de cambio de velocidad y la distancia de desconexión.

Imprecisión

La imprecisión es la diferencia entre el tiempo de respuesta máximo y mínimo. En el caso de un encóder absoluto es:

Imprecisión
Imprecisión = tiempo ejecución telegrama + tiempo monoestable
En el caso de encóders absolutos programables: Imprecisión = tiempo de ejecución de telegrama + tiempo monoestable + 1/frecuencia de pasos máx.

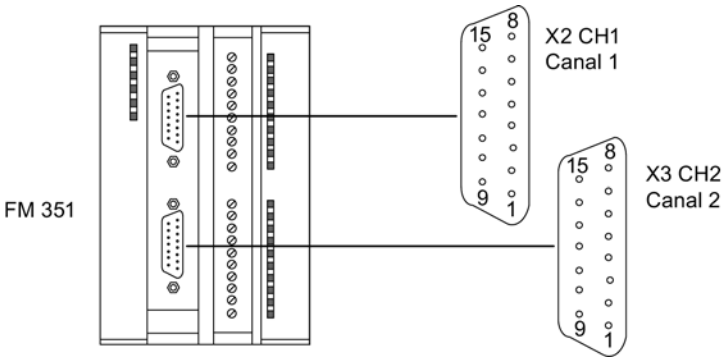
11.4.2.5 Uso del FM 351

Asignación de terminales de los conectores frontales del FM 351

Descripción de la interfaz del encóder

Posición de los conectores hembra sub-D

En la figura siguiente se muestra la posición de montaje y el nombre de los conectores hembra del módulo. En los dos conectores hembra sub-D se pueden conectar encóders incrementales o encóders absolutos (SSI).



Asignación de las interfaces de encóder (X2 y X3)

Pin	Nombre	Encóder incremental (24 V)	Encóder incremental (5 V)	Encóder absoluto
1	A*	Señal de encóder A	---	---
2	CLS	---	---	Pulso de desplazamiento SSI
3	/CLS	---	---	Pulso de desplazamiento SSI inverso
4	B*	Señal de encóder B	---	---
5	DC 24 V	Alimentación del encoder	Alimentación del encóder	Alimentación del encóder
6	DC 5.2 V	---	Alimentación del encóder	Alimentación del encóder
7	M	Tierra del bastidor	Tierra del bastidor	Tierra del bastidor
8	N*	Señal de marca cero	---	---
9	RE	Salida tipo PNP/NPN [1)]	---	---
10	N	---	Señal de marca cero	---
11	/N	---	Señal de marca cero invertida	---

Pin	Nombre	Encóder incremental (24 V)	Encóder incremental (5 V)	Encóder absoluto
12	/B	---	Señal de encóder B invertida	---
13	B	---	Señal de encóder B	---
14	/A / /DAT	---	Señal de encóder A invertida	Datos SSI invertidos
15	A / DAT	---	Señal de encóder A	Datos SSI
[1]**Nota:** si se conecta un encóder incremental de otro fabricante en contrafase (salida tipo PNP/NPN), debe tenerse en cuenta lo siguiente: • Salida tipo PNP: conecte RE (9) con la masa (7). • Salida tipo NPN: conecte RE (9) con +24 V (5).				

Asignación del conector frontal

Conector frontal

En el conector frontal de 20 polos se conectan:

- las entradas y salidas digitales asignadas a los canales;
- las tensiones de alimentación de los encóders y de las salidas digitales.

Asignación de terminales del conector frontal (X1)

La tabla siguiente contiene la asignación de terminales del conector frontal del FM 351.

Terminal	Nombre	Función	Encóder incremental	Encóder absoluto
1	1L+	Tensión auxiliar de DC 24 V para la alimentación del encóder		
2	1M	Masa de la alimentación del encóder		
3	1I0	Canal 1: entrada digital 0	Sensor del punto de referencia	no utilizado
4	1I1	Canal 1: entrada digital 1	Contacto inversor	no utilizado
5	1I2	Canal 1: entrada digital 2	Entrada de habilitación	
6	1I3	Canal 1: entrada digital 3	no utilizado	
7	2I0	Canal 2: entrada digital 0	Sensor del punto de referencia	no utilizado
8	2I1	Canal 2: entrada digital 1	Contacto inversor	no utilizado
9	2I2	Canal 2: entrada digital 2	Entrada de habilitación	
10	2I3	Canal 2: entrada digital 3	no utilizado	
11	1Q0	Canal 1: salida digital 0		
12	1Q1	Canal 1: salida digital 1		
13	1Q2	Canal 1: salida digital 2		
14	1Q3	Canal 1: salida digital 3		
15	2Q0	Canal 2: salida digital 0		

Terminal	Nombre	Función	Encóder incremental	Encóder absoluto
16	2Q1	Canal 2: salida digital 1		
17	2Q2	Canal 2: salida digital 2		
18	2Q3	Canal 2: salida digital 3		
19	2L+	Tensión auxiliar de DC 24 V para la alimentación de la carga		
20	2M	Masa de la alimentación de la carga		

Encontrará más información sobre el diseño y el cableado en las instrucciones de servicio (http://support.automation.siemens.com/WW/view/es/2103043) del módulo de posicionamiento FM 351.

Tensión auxiliar para la alimentación del encóder (1L+, 1M)

Aquí puede conectar una tensión auxiliar de DC 24 V para la alimentación del encóder. En el FM 351, el potencial de referencia de esta alimentación (1M) no está conectado con la masa de la alimentación de la carga (2M).

Se vigila si la tensión auxiliar de DC 24 V para la alimentación del encóder presenta subtensión y rotura de hilo de masa.

En la interfaz del encóder (conectores hembra sub-D X2 y X3) se proporcionan DC 24 V y DC 5,2 V para los diferentes tipos de encóder.

⚠ PRECAUCIÓN

Preste atención a la correcta polaridad de la tensión auxiliar de DC 24 V para la alimentación del encóder (1L+, 1M).

Si conecta la tensión auxiliar de DC 24 V para la alimentación del encóder con los polos invertidos, el módulo sufrirá daños y deberá sustituirse.

Tensión auxiliar para la alimentación de la carga (2L+, 2M)

En los bornes 2L+ y 2M debe conectar una tensión auxiliar de DC 24 V para la alimentación de la carga de las salidas digitales.

⚠ PRECAUCIÓN

Preste atención a la correcta polaridad de la tensión auxiliar de DC 24 V para la alimentación de la carga (2L+, 2M).

Si conecta la tensión auxiliar de DC 24 V para la alimentación de la carga con los polos invertidos, el módulo sufrirá daños y deberá sustituirse.

Indicación sobre el cableado para DC 24 V

Al efectuar el cableado, tenga en cuenta que los bornes 1L+, 1M y 2L+, 2M deben estar interconectados para que el módulo funcione sin fallos.

Si conecta 1L+, 1M y 2L+, 2M a alimentaciones separadas, en caso de producirse una pérdida de tensión se mantendrá la sincronización de los ejes gracias a la tensión auxiliar para la alimentación de la carga (2L+, 2M).

Alimentaciones de carga

La alimentación de la carga DC debe satisfacer los siguientes requisitos:

Como alimentación de la carga sólo debe utilizarse una pequeña tensión DC ≤ 60 V separada con seguridad de la red. La separación segura puede estar efectuada según los requisitos que figuran, entre otros, en:

- VDE 0100 Parte 410/HD 384-4-41/IEC 364-4-41
  (como pequeña tensión funcional con separación segura),
- VDE 0805/EN 60950/IEC 950
  (como pequeña tensión de seguridad SELV) o bien
- VDE 0106 Parte 101.

8 entradas digitales (de 1I0 a 2I3)

El FM 351 dispone de 4 entradas digitales por canal.

En las 8 entradas digitales es posible conectar conectores libres de rebotes (de 24 V tipo P) o sensores sin contacto (detectores de proximidad de 2 ó 3 hilos).

No se vigila si las entradas digitales presentan cortocircuitos o roturas de hilo; dichas entradas están aisladas galvánicamente de la masa de la alimentación del encóder y de la masa de la CPU.

El estado de cada entrada se indica con el LED correspondiente.

8 salidas digitales (de 1Q0 a 2Q3)

El FM 351 dispone de 4 salidas digitales por canal.

Con las salidas digitales se controla la etapa de potencia. El funcionamiento de las salidas digitales depende del modo de control (Página 10941).

No se vigila si las salidas digitales presentan cortocircuitos o roturas de hilo; dichas salidas están aisladas galvánicamente de la masa de la alimentación del encóder y de la masa de la CPU.

El estado de cada salida se indica en el LED correspondiente.

Funciones de FM 351

Funciones

Al especificar un modo de operación, una petición o un selector de función, se elige la funcionalidad con la que se utilizará el módulo. La siguiente tabla ofrece una visión general de la funcionalidad del módulo.

Funcionalidad	Descripción
Escribir y leer los datos máquina (Página 10904)	Con los datos máquina el módulo de posicionamiento se adapta al eje y al encóder.
Escribir y leer tablas de incrementos (Página 10906)	Con los incrementos se especifican destinos que pueden ser controlados por el módulo de posicionamiento en el modo de operación "Modo incremental relativo/absoluto".
Final de una operación de posicionamiento (Página 10908)	El final de una operación de posicionamiento se logra de tres formas distintas: • Aproximación al destino • Desactivación • Cancelar
Modo de operación "Jog" (Página 10915)	En el modo de operación "Jog", el accionamiento se mueve en un sentido al pulsar una tecla.
Modo de operación "Búsqueda del punto de referencia" (Página 10918)	El modo de operación "Búsqueda del punto de referencia" permite sincronizar el eje en función de un evento externo repetitivo.
Modo de operación "Modo incremental" (Página 10924)	En el modo de operación "Modo incremental" el módulo de posicionamiento puede • mover el accionamiento a destinos absolutos, • mover el accionamiento de forma relativa alrededor de una parte del trayecto en un sentido predeterminado.
Petición "Definir valor real" / "Cancelar preselección del valor real" (Página 10930)	Con la petición "Ajustar valor real" se asigna una nueva coordinada al estado actual del sensor. Con la petición "Cancelar valor real ajustado" se restablecen todos los offsets de la zona de trabajo que han sido generados con "Ajustar valor real".
Petición "Definir punto de referencia" (Página 10932)	Con la petición "Definir punto de referencia" se sincroniza el eje.
Selector de función "Aproximación en bucle" (Página 10933)	La "Aproximación en bucle" define el sentido en el que se aproxima al destino con cierre de fuerza.
Selector de función "Entrada de habilitación" (Página 10936)	La entrada de habilitación es una entrada externa con la que puede habilitarse un posicionamiento a través de un evento externo.
Petición "Leer datos de posición" (Página 10937)	Con la petición "Leer datos de posición" es posible leer el incremento, el trayecto residual y la velocidad en el instante actual.
Petición "Leer datos del encóder" (Página 10937)	Con la función "Leer datos de encóder" se leen los datos actuales del encóder, así como el valor del reajuste del encóder absoluto.
Señales de realimentación del posicionamiento (Página 10938)	Con la función "Señales de respuesta para el posicionamiento" se informa del estado actual del posicionamiento.
Señales de realimentación del diagnóstico (Página 10938)	Con las "Señales de realimentación del diagnóstico" se recibe información sobre los eventos de diagnóstico que se han producido.

Escribir y leer los datos máquina

Escribir, activar y leer los datos máquina

Con los datos máquina el módulo de posicionamiento se adapta al eje y al encóder.

El programa de usuario permite modificar y leer los parámetros durante el funcionamiento.

Todos los parámetros se depositan en el DB de parámetros:

- Los datos máquina se encuentran en el DB de parámetros, en las direcciones que van del 4.0 al 116.0.

El número del DB de parámetros se tiene que registrar en el DB de canal correspondiente en cada caso.

Los parámetros pueden introducirse con el editor DB, o bien introducirse cómodamente en los cuadros de diálogo "Accionamiento", "Eje" y "Encóder" y se escriben con la función "Exportar" en el DB de parámetros.

Los parámetros de un DB de parámetros existente pueden importarse a los cuadros de diálogo con la función "Importar".

Primera parametrización

Si el canal todavía no contiene ningún dato máquina, proceda de la siguiente manera para una primera parametrización sin el cuadro de diálogo de parametrización:

1. Introduzca los nuevos valores en el DB de parámetros y almacénelos.
2. Cargue el DB de parámetros en la CPU.
3. Active el siguiente bit de inicio en el DB de canal para la petición:
   - Escribir datos máquina (MDWR_EN = 1)
4. En el programa de usuario cíclico llame la instrucción ABS_CTRL (Página 7420).

Modificar dato máquina

Para modificar datos máquina mediante el programa de usuario, proceda del siguiente modo:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Active los bits de inicio en el DB de canal para las peticiones:
   - Escribir datos máquina (MDWR_EN = 1)
   - Activar datos máquina (MD_EN = 1)
3. En el programa de usuario cíclico llame la instrucción ABS_CTRL (Página 7420).

Si define los bits de inicio de estas peticiones al mismo tiempo, la instrucción ABS_CTRL se encargará de que las peticiones se procesen en la secuencia correcta.

De lo contrario, cambie siempre los datos máquina en la siguiente secuencia:

- Escribir datos máquina
- Activar datos máquina

Leer datos máquina

Para leer los datos máquina actuales desde un canal, proceda del siguiente modo:

1. Active el siguiente bit de inicio en el DB de canal:
   – Leer datos máquina (MDRD_EN = 1)
2. En el programa de usuario cíclico llame la instrucción ABS_CTRL (Página 7420).

De este modo se depositan en la CPU los datos máquina del DB de parámetros.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.0	MDWR_EN	BOOL	FALSE	1 = Escribir datos máquina
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina
36.5	MDRD_EN	BOOL	FALSE	1 = Leer datos máquina

Nota

Si se modifican parámetros relevantes para la sincronización, al activar los datos máquina el módulo realiza las siguientes acciones para el canal afectado:

- La sincronización se borrará.
- Los selectores de función y el decalaje de señal cero se inicializan.
- Los datos máquina y las tablas de incrementos existentes hasta ahora quedan invalidados.

Son parámetros relevantes para la sincronización:

- Tipo de eje
- Final del eje rotativo
- Tipo de señal
- Recorrido por vuelta
- Incrementos por vuelta
- Número de vueltas
- Coordenada del punto de referencia
- Reajuste del encóder absoluto
- Tipo de búsqueda del punto de referencia
- Sentido de contaje

Escribir y leer tablas de incrementos

Escribir y leer tablas de incrementos

Con los incrementos se especifican destinos que pueden ser controlados por el módulo de posicionamiento en el modo de operación "Modo incremental relativo/absoluto".

El programa de usuario permite modificar y leer los parámetros durante el funcionamiento.

Todos los parámetros se depositan en el DB de parámetros:

- Las tablas de incrementos se encuentran en el DB de parámetros en las direcciones 120.0 a 516.0.

El número del DB de parámetros se tiene que registrar en el DB de canal correspondiente en cada caso.

Los parámetros pueden introducirse con el editor DB, o bien introducirse cómodamente en el cuadro de diálogo "Incrementos" y se escriben con la función "Exportar" en el DB de parámetros.

Los parámetros de un DB de parámetros existente pueden importarse al cuadros de diálogo con la función "Importar".

Primera parametrización

Si el canal todavía no contiene ninguna tabla de incrementos, proceda del siguiente modo para una primera parametrización sin el cuadro de diálogo de parametrización:

1. Introduzca los nuevos valores en el DB de parámetros y almacénelos.
2. Cargue el DB de parámetros en la CPU.
3. Active los bits de inicio en el DB de canal para las peticiones:
   – Escribir tabla de incrementos 1 (TRGL1WR_EN = 1) y/o escribir tabla de incrementos 2 (TRGL2WR_EN = 1)
4. En el programa de usuario cíclico llame la instrucción ABS_CTRL (Página 7420).

Modificar tablas de incrementos

Para modificar las tablas de incrementos mediante el programa de usuario, proceda del siguiente modo:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Active los bits de inicio en el DB de canal para las peticiones:
   – Escribir tabla de incrementos 1 (TRGL1WR_EN = 1) y/o escribir tabla de incrementos 2 (TRGL2WR_EN = 1)
3. En el programa de usuario cíclico llame la instrucción ABS_CTRL (Página 7420).

Leer tablas de incrementos

Para leer las tablas de incrementos a partir de un canal, proceda del siguiente modo:

1. Active los bits de inicio en el DB de canal para las peticiones:
   – Leer tabla de incrementos 1 (TRGL1RD_EN = 1) y/o leer tabla de incrementos 2 (TRGL2RD_EN = 1)
2. En el programa de usuario cíclico llame la instrucción ABS_CTRL (Página 7420).

De este modo se depositan en la CPU las tablas de incrementos del DB de parámetros.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (1 ... 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (51 ... 100)
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 1 (1 ... 50)
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 2 (51 ... 100)

Nota

Si se modifican parámetros relevantes para la sincronización, al activar los datos máquina el módulo realiza las siguientes acciones para el canal afectado:

- La sincronización se borrará.
- Los selectores de función y el decalaje de señal cero se inicializan.
- Los datos máquina y las tablas de incrementos existentes hasta ahora quedan invalidados.

Son parámetros relevantes para la sincronización:

- Tipo de eje
- Final del eje rotativo
- Tipo de señal
- Recorrido por vuelta
- Incrementos por vuelta
- Número de vueltas
- Coordenada del punto de referencia
- Reajuste del encóder absoluto
- Tipo de búsqueda del punto de referencia
- Sentido de contaje

Final de una operación de posicionamiento

Definición

El final de una operación de posicionamiento se logra a través de la señal de realimentación WORKING = 0. Esto se logra de tres formas distintas:

- Aproximación al destino
- Desactivación
- Cancelar

Vigilancias

Al final de una operación de posicionamiento se encuentran activas las siguientes vigilancias:

- Tiempo de vigilancia
  El tiempo de vigilancia se redispara en el punto de desconexión por última vez y pierde su validez al finalizar la operación de posicionamiento.
  En el transcurso de este tiempo es necesario alcanzar el final de la operación de posicionamiento. De lo contrario, las salidas se desconectan y se notifica el fallo operativo "Error en la aproximación al destino" (número de error 5).
- Vigilancia de la zona de destino
  El FM x51 crea un área simétrica alrededor de cada destino y define con ello la precisión de posicionamiento de su aplicación. Dentro de esta área es preciso que el eje se pare durante una aproximación al destino. Una especificación de valor 0 desconecta la tolerancia en la aproximación al destino.
- Vigilancia de la velocidad de parada
  La velocidad de parada permite determinar si el accionamiento se para dentro de la zona de destino. Una vez alcanzado el punto de desconexión, el FM x51 vigila si la velocidad real del accionamiento es inferior a la velocidad de parada.
  La velocidad de parada debe rebasarse por defecto dentro de la zona de trabajo. De lo contrario, el FM x51 notifica el fallo operativo "Rebasar zona de destino" (número de error 10).
  El desbordamiento por defecto de la velocidad de parada sólo se vigila una vez por aproximación al destino.
  Tenga en cuenta en este caso que la velocidad de parada para la determinación de la velocidad del módulo puede rebasarse temporalmente por defecto si el eje se mueve con una velocidad de posicionamiento muy pequeña (menos de 2 impulsos por 8 ms).
- Vigilancia de la zona de parada
  Tras finalizar una operación de posicionamiento se vigila si el accionamiento se detiene en una posición de destino alcanzada o si se aleja de ella.
  La zona de parada se vigila
  - después de que el FM x51 haya notificado la señal de realimentación "PEH",
  - si el tiempo de vigilancia se rebasa por exceso,
  - si la velocidad de parada se rebasa por defecto.

  Si la zona de parada se abandona sin una petición de marcha válida, el FM x51 notifica el fallo operativo "Abandonar zona de parada" (número de error 6).

Aproximación al destino

La aproximación al destino en los modos de operación "Modo incremental absoluto" y "Modo incremental relativo" comienza cuando se alcanza el punto de desconexión. A partir de este punto se desconecta el accionamiento y el FM x51 se hace cargo de las funciones de vigilancia.

En función de las vigilancias que se hayan parametrizado, surgen diferentes casos para la creación de la señal de repuesta "PEH (POS_RCD)". La operación de posicionamiento se cancela cuando se crea la señal "PEH (POS_RCD)".

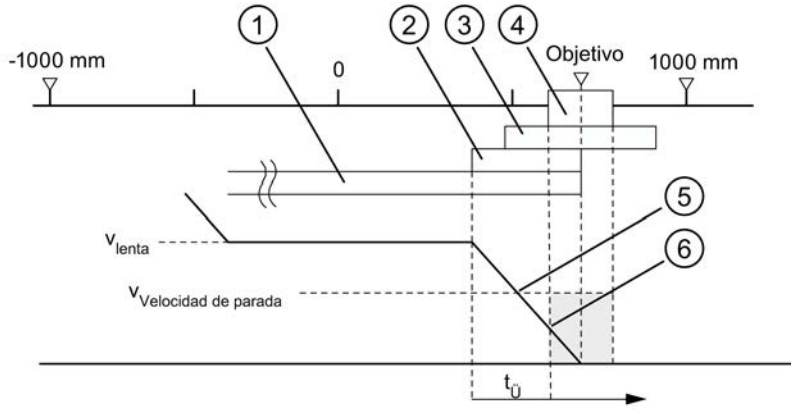
Caso 1: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) > 0
- Velocidad de parada (ZSPEED_L) > 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si la velocidad de parada se rebasa por defecto y se alcanza la zona de destino. Es irrelevante la condición que se cumpla en primer lugar.

No se genera PEH si el valor real no alcanza la zona de destino dentro del tiempo de vigilancia o si la zona de destino se sobrepasa sin rebasar por defecto la velocidad de parada.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino > 0 y velocidad de parada > 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Zona de destino
⑤ Velocidad de parada alcanzada
⑥ Zona de destino con V_{parada} alcanzada: PEH se activa
$t_Ü$ Tiempo de vigilancia

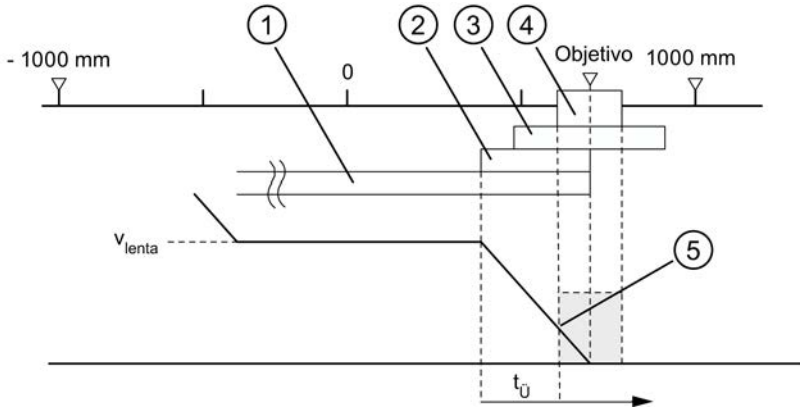
Caso 2: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) > 0
- Velocidad de parada (ZSPEED_L) = 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si se alcanza la zona de destino.

No se genera una PEH si el valor real no alcanza la zona de destino dentro del tiempo de vigilancia.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino > 0 y velocidad de parada = 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Zona de destino
⑤ Zona de destino con V_{parada} alcanzada: PEH se activa
$t_Ü$ Tiempo de vigilancia

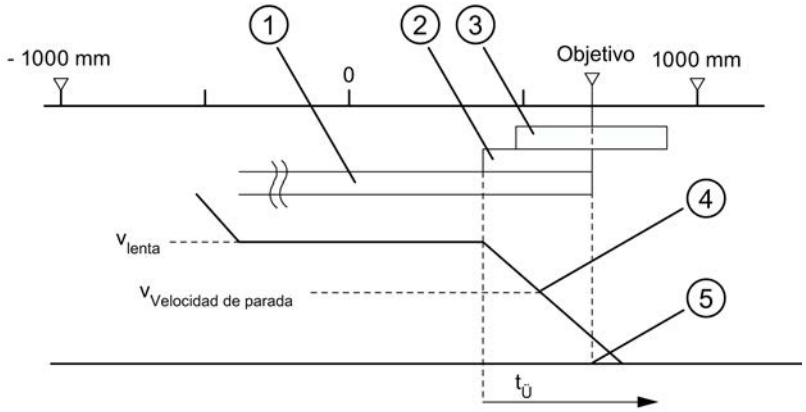
Caso 3: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) = 0
- Velocidad de parada (ZSPEED_L) > 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si la velocidad de parada se rebasa por defecto y se alcanza el destino.

No se genera una PEH si el valor real no alcanza el destino dentro del tiempo de vigilancia o si la zona de destino se sobrepasa sin rebasar por defecto la velocidad de parada.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino = 0 y velocidad de parada > 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Velocidad de parada alcanzada
⑤ Destino alcanzado: PEH se activa
$t_Ü$ Tiempo de vigilancia

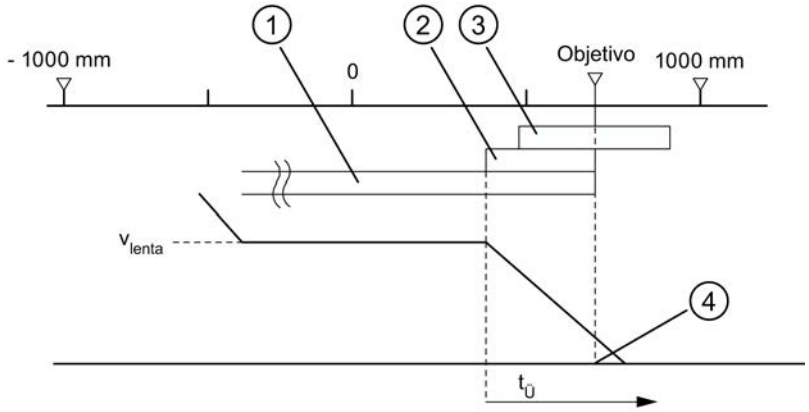
Caso 4: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) = 0
- Velocidad de parada (ZSPEED_L) = 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si se alcanza el destino.

No se genera una PEH si el valor real no alcanza el destino dentro del tiempo de vigilancia.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino = 0 y velocidad de parada = 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Destino alcanzado: PEH se activa
$t_Ü$ Tiempo de vigilancia

Caso 5: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) ≥ 0
- Velocidad de parada (ZSPEED_L) ≥ 0
- Tiempo de vigilancia (MON_TIME) = 0

Si en este caso el accionamiento se para delante de la zona de destino, el final de la operación de posicionamiento no se detecta. No se genera una PEH y la señal de realimentación WORKING permanece activada. La operación de posicionamiento sólo puede cancelarse eliminando la habilitación del accionamiento (DRV_EN = 0).

Desactivación sin destino predeterminado

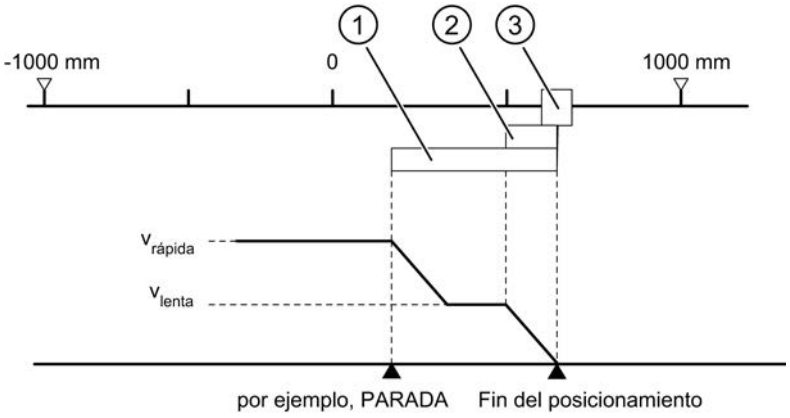
Desactivación significa: El proceso de posicionamiento se cierra de forma precisa manteniendo las diferencias desde la marcha rápida a la marcha lenta.

La operación de posicionamiento se desactiva si:

- el FM x51 recibe una señal de STOP (STOP = 1);
- los modos de operación "Jog" y "Búsqueda del punto de referencia" finalizan;
- aparecen errores de operador o el fallo operativo "Rebasar destino" " (número de error 9).

La señal de realimentación "PEH (POS_RCD)" no se activa. Los procesos son análogos a los de la aproximación al destino.

La siguiente figura muestra la desactivación de una operación de posicionamiento.



① Distancia de cambio de velocidad positiva

② Distancia de desconexión positiva

③ Zona de destino

Cancelar

Cancelar significa: La operación de posicionamiento finaliza inmediatamente sin que se utilicen la distancia de cambio de velocidad ni la distancia de desconexión de la marcha rápida o la marcha lenta después de la parada. Para ello, todas las salidas relevantes del modo de control pertinente se desconectan inmediatamente y, además:

- Incremento = valor real
- Recorrido residual = Cero

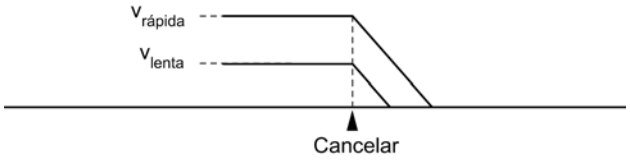
La operación de posicionamiento se cancela si:

- la señal de habilitación del accionamiento se borra (DRV_EN = 0);
- la CPU pasa a STOP;
- aparecen errores de diagnóstico u otros fallos operativos aparte del error "Rebasar destino" " (número de error 9).

La señal de realimentación "PEH (POS_RCD)" no se activa en el modo de operación "Modo incremental".

Cuando la velocidad de parada está parametrizada, la vigilancia de parada se activa en cuanto se rebasa por defecto. Cuando la velocidad de parada está parametrizada, la vigilancia de parada se activa con la desactivación de las salidas.

La siguiente figura muestra la cancelación de una operación de posicionamiento.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
76.0	TRG_RANGE	DINT	L#1000	Zona de destino
80.0	MON_TIME	DINT	L#2000	Tiempo de vigilancia
84.0	ZSPEED_R	DINT	L#1000	Zona de parada
88.0	ZSPEED_L	DINT	L#30000	Velocidad de parada

Señales de respuesta en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
23.1	WORKING	BOOL	FALSE	1 = Operación de posicionamiento en curso
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada

Modo de operación "Jog"

Definición

En el modo de operación "Jog", el accionamiento se mueve en un sentido al pulsar una tecla. Para los dos sentidos (positivo y negativo) es preciso instalar sendos botones. El modo de operación "Jog" se puede utilizar tanto para el eje sincronizado como para el eje no sincronizado.

Si el eje no está sincronizado el modo "Jog" permite realizar un posicionamiento en el sentido preestablecido.

Si el eje está sincronizado (eje lineal), el modo "Jog" permite realizar un posicionamiento en el final de carrera por software.

Requisitos

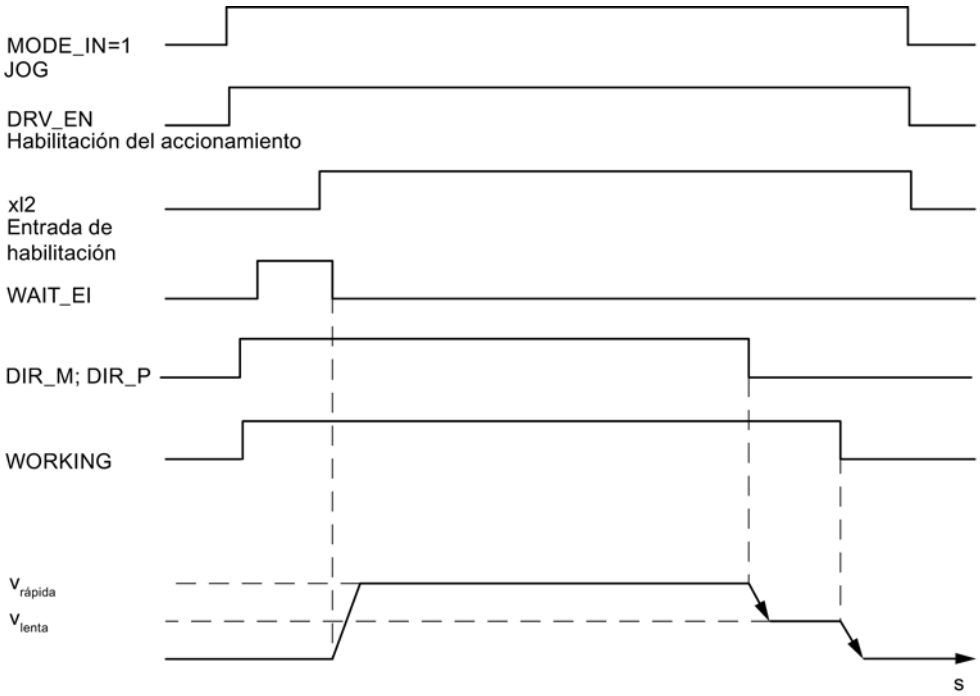
El eje debe estar parametrizado.

Desarrollo del modo de operación "Jog"

1. Active la señal de control para el modo de operación "Jog" (MODE_IN = 1).
2. Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).
3. Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.
4. Introduzca la velocidad de inicio.
   - Marcha rápida (MODE_TYPE = 1)
   - Marcha lenta (MODE_TYPE = 0)

5. Conecte la señal de control para el sentido de marcha positivo o negativo (DIR_P = 1 o DIR_M = 1).
6. Llame la instrucción ABS_CTRL (Página 7420).

La figura siguiente muestra un ejemplo del modo de operación "Jog".



Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	1 = Jog
17.0	MODE_TYPE	BYTE	B#16#0	1 = Marcha rápida 0 = Marcha lenta
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Operación de posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Desactivación del modo Jog

El modo de operación "Jog" se desactiva cuando

- se vuelve a abrir el botón con el que realiza el modo "Jog" (DIR_M = 0 o DIR_P = 0);
- el FM 351 recibe una señal de STOP (STOP = 1),
- el valor real de un eje lineal sincronizado alcanza el límite de la zona de trabajo. La continuación de la marcha sólo es posible en el sentido contrario.

Tras desactivar la marcha puede continuarse la marcha en el sentido que se desee.

Cancelación del modo Jog

El modo de operación "Jog" se cancela cuando

- la señal de habilitación del accionamiento se borra (DRV_EN = 0),
- se ha rebasado un límite de la zona de desplazamiento en un eje lineal.

Vigilancias

En el modo de operación "Jog", las siguientes vigilancias no están activas al final del posicionamiento:

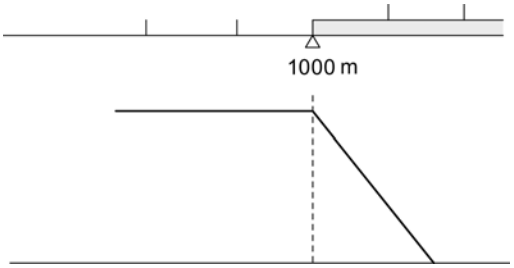
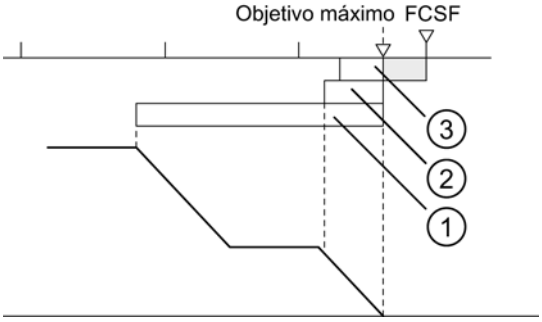
- Vigilancia de la zona de parada
- Vigilancia de la zona de destino
- Vigilancia de la velocidad de parada

El modo "Jog" no produce un posicionamiento correcto en el que al final se activa la señal "Posición alcanzada".

Límite de la zona de trabajo en un eje lineal

Los límites para el modo "Jog" son distintas en función de si el eje está sincronizado o no sincronizado.

La tabla siguiente muestra el modo "Jog" con un eje sincronizado y no sincronizado.

El eje no está sincronizado.	El eje está sincronizado.
Si en el modo "Jog" se rebasa el límite de la zona de desplazamiento. • el indicador del valor real ya no es válido. • se cancela la operación de posicionamiento.  El indicador del valor real ya no es válido	El modo "Jog" representa un posicionamiento en destinos que están separados de los finales de carrera por software en toda la zona de destino. Los límites de la zona de trabajo se calculan a partir de • **SEE - ½ zona de destino** para el final del eje lineal en sentido positivo • **SEE + ½ zona de destino** para el final del eje lineal en sentido negativo Si no abre antes el botón, el FM 351 se desconecta en un punto que se encuentra la mitad de la zona de destino delante del final de carrera por software correspondiente. El FM 351 activa todas las áreas que se necesitan para la desactivación correcta alrededor de este punto de destino.  = Área de la zona de trabajo en la que no puede haber ninguna posición de destino ① = Distancia de cambio de velocidad positiva ② = Distancia de desconexión positiva ③ = ½ zona de destino

Modo de operación "Búsqueda del punto de referencia"

Definición

El modo de operación "Búsqueda del punto de referencia" permite sincronizar el eje en función de un evento externo repetitivo.

Requisitos

- Un encóder incremental con señal cero.
- El eje está parametrizado:

Conexiones	Canal 1	Canal 2
Sensor del punto de referencia	Entrada digital 1I0	Entrada digital 2I0
	El sensor del punto de referencia debe estar estructurado de tal modo que el accionamiento puede frenar de forma segura de la marcha rápida a la marcha lenta en el área del interruptor.	
Contacto inversor	Entrada digital 1I1	Entrada digital 2I1
	Durante la parametrización asegúrese de que el inicio de la búsqueda del punto de referencia esté parametrizada en el sentido del contacto inversor. Sólo así puede garantizarse que el sensor del punto de referencia se encontrará siempre sin problemas.	
Entrada de habilitación	Entrada digital 1I2	Entrada digital 2I2

Desarrollo del modo de operación "Búsqueda del punto de referencia"

El botón del sentido de desplazamiento positivo y negativo debe instalarse para cada canal.

1. Introduzca el valor de la coordenada del punto de referencia en el DB de parámetros (REFPT).
2. Introduzca el tipo de "Búsqueda del punto de referencia" en el DB de parámetros.
   Para ello, existen las posibilidades siguientes:

Inicio en sentido ...	Produce la sincronización ...	
plus	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **positivo**	REFPT_TYPE = 0
plus	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **negativo**	REFPT_TYPE = 1
menos	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **positivo**	REFPT_TYPE = 2
menos	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **negativo**	REFPT_TYPE = 3

3. Introduzca la velocidad de inicio;
   - Marcha rápida (REFPT_SPD = 0)
   - Marcha lenta (REFPT_SPD = 1)
4. Escriba y active los datos máquina.
5. Active la señal de control para el modo de operación "Búsqueda del punto de referencia" (MODE_IN = 3).
6. Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).
7. Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.
8. Conecte la señal de control para el sentido de marcha positivo o negativo o la señal de inicio (DIR_P = 1, DIR_M = 1 o START = 1).
9. Llame la instrucción ABS_CTRL (Página 7420).

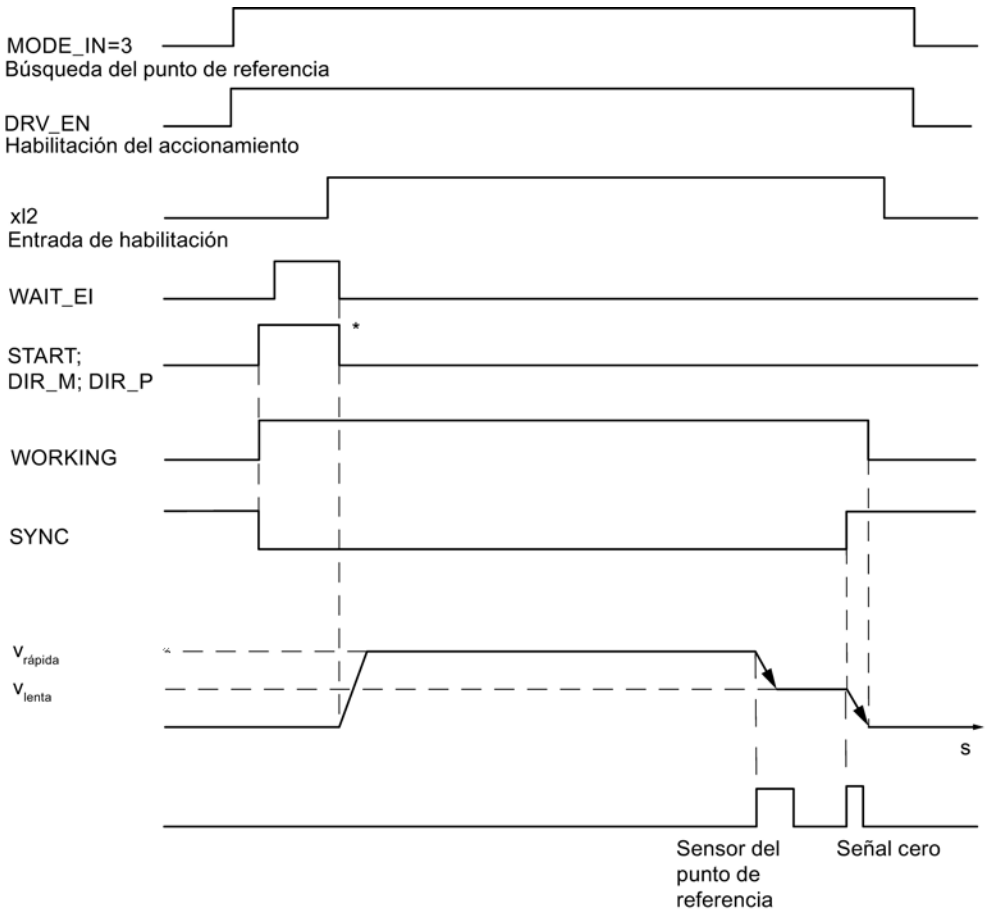
La tabla siguiente muestra los comandos de inicio para una búsqueda del punto de referencia.

Comando de inicio	Tarea	Comentario
DIR_P	El accionamiento se inicia en el sentido de los valores positivos. Se mueve, pues, en el sentido del final de la zona de desplazamiento.	Si en los datos máquina se introduce un sentido negativo, el FM 351 notifica un error de operador. No se realiza ninguna búsqueda del punto de referencia.
DIR_M	El accionamiento se inicia en el sentido de los valores negativos. Se mueve, pues, en el sentido del principio de la zona de desplazamiento.	Si en los datos máquina se introduce un sentido positivo , el FM 351 notifica un error de operador. No se realiza ninguna búsqueda del punto de referencia.
START	El accionamiento se inicia en el sentido que se ha introducido en los datos máquina.	

Nota

Para el eje rotativo rige lo siguiente: La reproducibilidad del punto de referencia sólo está garantizado si entre el valor **Final del eje rotativo** y el valor **Recorrido por vuelta** existe una relación de número entero.

La figura siguiente muestra un ejemplo del modo de operación "Búsqueda del punto de referencia".



* Las señales de inicio se desactivan con la instrucción ABS_CTRL (Página 7420).

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.0	START	BOOL	FALSE	1 = Iniciar operación de posicionamiento
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	3 = Búsqueda del punto de referencia
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Operación de posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	Coordenada del punto de referencia
52.0	REFPT_TYPE	DINT	L#0	Tipo de búsqueda del punto de referencia
99.0	REFPT_SPD	BOOL	TRUE	Velocidad de inicio para la búsqueda del punto de referencia 0 = Marcha rápida 1 = Marcha lenta

Efectos del modo de operación

- Con el inicio de la marcha se elimina la sincronización.
- La posición real se ajusta al valor de la coordenada del punto de referencia si se activa la señal de realimentación SYNC.
- La zona de trabajo se fija en el eje.
- Los distintos puntos dentro de la zona de trabajo conservan sus valores originales, pero se encuentran en posiciones físicas nuevas.

Cancelación de la búsqueda del punto de referencia

El modo de operación "Búsqueda del punto de referencia" se cancela cuando

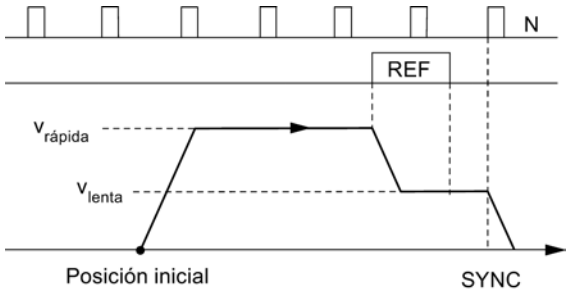
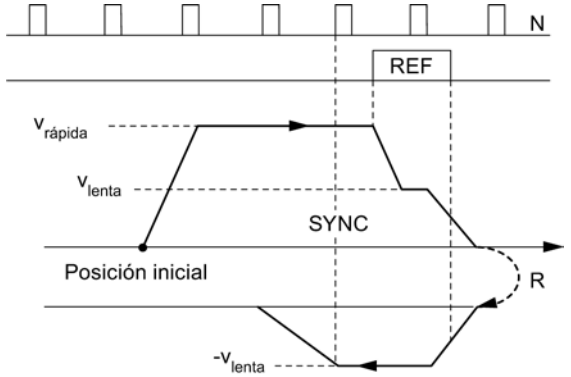
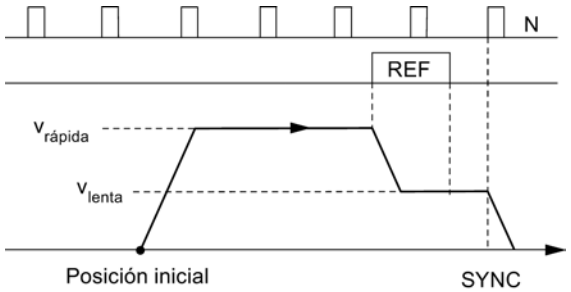
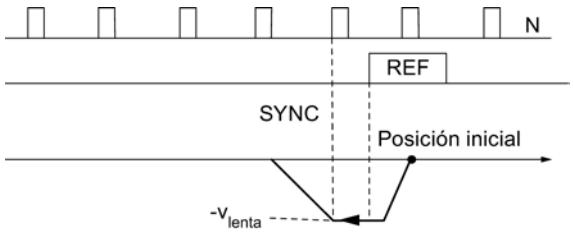
- la señal de habilitación del accionamiento se borra (DRV_EN = 0),
- se ha rebasado un límite de la zona de desplazamiento en un eje lineal.

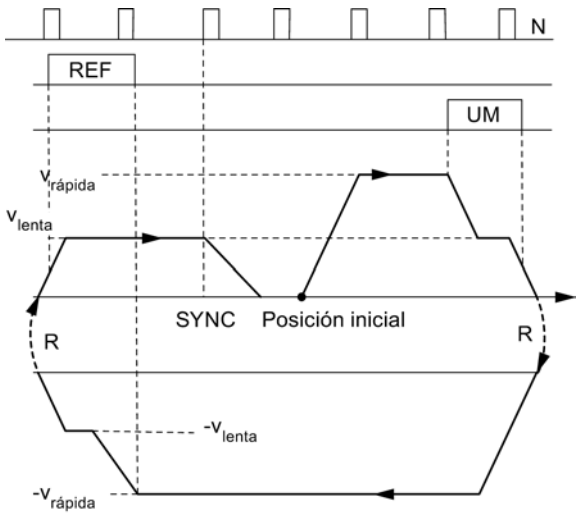
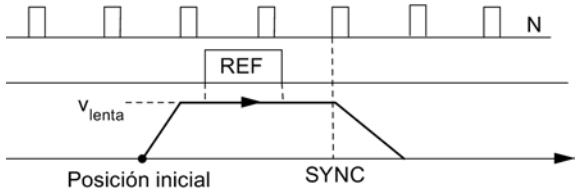
Búsqueda del punto de referencia dependiente de la posición de inicio

En una búsqueda del punto de referencia es preciso distinguir diversos casos que dependen

- de la posición del accionamiento al iniciar una búsqueda del punto de referencia,
- del sentido de arranque parametrizado,
- de la posición parametrizada de la señal cero respecto al sensor del punto de referencia.

La tabla siguiente muestra los casos de REFPT_TYPE = 0 y REFPT_TYPE = 1. Para REFPT_TYPE = 2 y REFPT_TYPE = 3 rigen las figuras según corresponda.

Condiciones para la búsqueda del punto de referencia	Desarrollo de la búsqueda del punto de referencia
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 0): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizada en sentido positivo.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 1): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido negativo.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 0): • El sentido de inicio debe estar parametrizado en sentido positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizada en sentido positivo. • El contacto inversor es más positivo que el sensor del punto de referencia.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 1): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido negativo. • La posición de inicio de la búsqueda del punto de referencia se encuentra en el sensor del punto de referencia.	

Condiciones para la búsqueda del punto de referencia	Desarrollo de la búsqueda del punto de referencia
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 0): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizada en sentido positivo. • El contacto inversor es más positivo que el sensor del punto de referencia.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE=0): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizada en sentido positivo. • Velocidad de inicio = velocidad lenta	
R = Inversión del sentido REF = Sensor del punto de referencia UM = Contacto inversor N = señal cero del encóder SYNC = Se ha alcanzado la sincronización	

Modo de operación "Modo incremental"

Definición

En el modo de operación "Modo incremental" el FM 351 puede mover el accionamiento

- a destinos **absolutos**,
- mover el accionamiento **de forma relativa** alrededor de una parte del trayecto en un sentido predeterminado.

El objetivo o los recorridos relativos se predeterminan como incrementos del FM 351. Tiene la posibilidad de insertar un máximo de 100 incrementos en una tabla, que son válidas tanto el modo de operación "Modo incremental relativo" como para el "Modo incremental absoluto". Independientemente de la tabla de incrementos es posible especificar el recorrido con el número de incremento 254 (Página 10970) y el número de incremento 255 (Página 10970).

Requisitos

- El eje debe estar parametrizado.
- El eje debe estar sincronizado.
- Los incrementos deben estar presentes en el módulo.

Interpretación de los incrementos

En función del modo incremental que se seleccione, el FM 351 interpreta los datos especificados de una manera u otra.

- Modo incremental absoluto: Los incrementos se interpretan como posición de destino absoluta.
- Modo incremental relativo: Los incrementos se interpretan como diferencia de recorrido desde la posición de inicio.

Nota

Para el modo de operación "Modo incremental relativo" sólo se permiten incrementos positivos.

Los incrementos obtienen su signo mediante las especificaciones de sentido DIR_P o DIR_M.

Desarrollo del modo de operación "Modo incremental" con número de incremento 1 a 100

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 1 a 100	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Anote los incrementos en las tablas (TRGL1; TRGL2).	
3	Escriba las tablas de incrementos (TRGL1/2WR_EN = 1).	
4	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
5	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
6	Introduzca el número de incremento (MODE_TYPE = 1 … 100).	
7	Active la señal de control: • Eje lineal: – START; el sentido se determina de forma unívoca a través del destino y el valor real actual. • Eje rotativo: – START; se llega al punto de destino por el trayecto más corto. – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	Active la señal de control: • Eje lineal: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo • Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 1 a 100	
8	Llame la instrucción ABS_CTRL (Página 7420).	
Los pasos 2 y 3 se necesitan ahora si no hay aún incrementos o si es preciso modificar las medidas existentes.		

Desarrollo del modo de operación "Modo incremental" con número de incremento 254

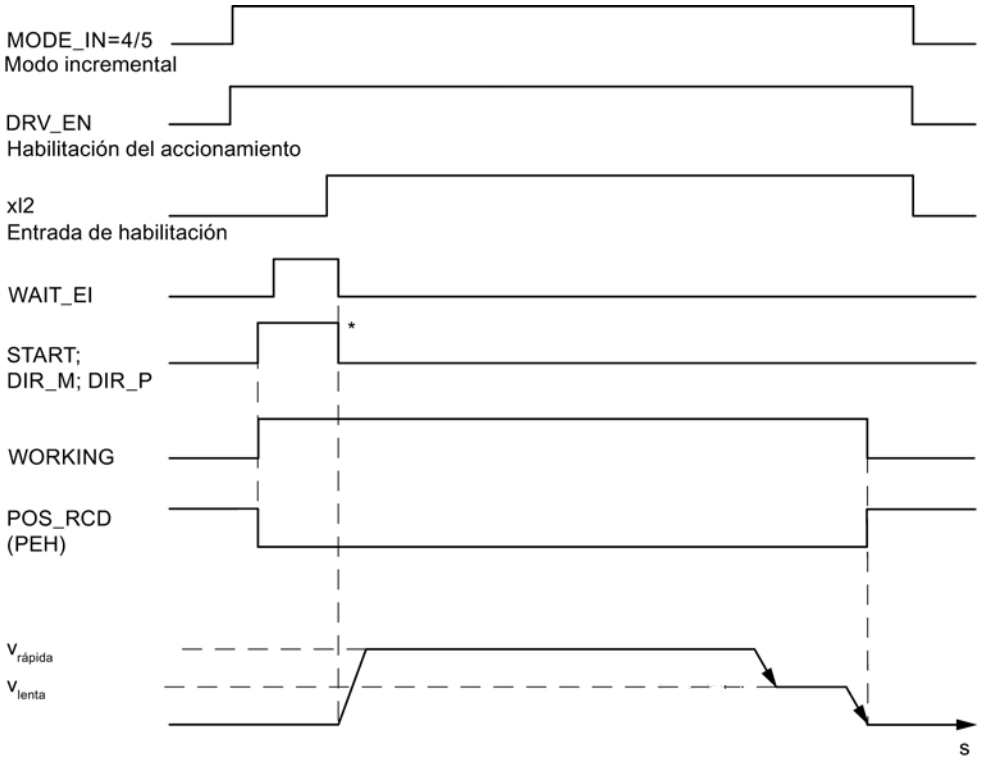
Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 254	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
3	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
4	Introduzca el número de incremento (MODE_TYPE = 254).	
5	Introduzca el incremento para el número de incremento 254 (TRG252_254).	
6	Active el bit de inicio para escribir el incremento (TRG252_254_EN = 1).	
7	Active la señal de control: • Eje lineal: – START; el sentido se determina de forma unívoca a través del destino y el valor real actual. • Eje rotativo: – START; se llega al punto de destino por el trayecto más corto. – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	Active la señal de control: • Eje lineal: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo • Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo
8	Llame la instrucción ABS_CTRL (Página 7420).	

Desarrollo del modo de operación "Modo incremental" con número de incremento 255

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 255	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
3	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
4	Introduzca el número de incremento (MODE_TYPE = 255).	
5	Introduzca el incremento para el número de incremento 255 (TRG255).	
6	Introduzca el valor para la distancia de cambio de velocidad del número de incremento 255 (CHGDIF255).	

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 255	
7	Introduzca el valor para la distancia de desconexión del número de incremento 255 (CUTDIF255).	
8	Active el bit de inicio para escribir el incremento y la distancia de desconexión y de cambio de velocidad (TRG255_EN = 1).	
9	Active la señal de control: • Eje lineal: – START; el sentido se determina de forma unívoca a través del destino y el valor real actual. • Eje rotativo: – START; se llega al punto de destino por el trayecto más corto. – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	Active la señal de control: • Eje lineal: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo • Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo
10	Llame la instrucción ABS_CTRL (Página 7420).	

La figura siguiente muestra un ejemplo del modo de operación "Modo incremental".



* Las señales de inicio se resetean con la instrucción ABS_CTRL (Página 7420).

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.0	START	BOOL	FALSE	1 = Iniciar posicionamiento
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	4 = Modo incremental relativo 5 = Modo incremental absoluto
17.0	MODE_TYPE	BYTE	B#16#0	Número de incremento 1 … 100, 254 o 255
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación
36.2	TRG252_254_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 254
36.3	TRG255_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 255
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (número de incremento 1 ... 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (número de incremento 51 ... 100)
96.0	TRG252_254	DINT	L#0	Incremento para número de incremento 254
100.0	TRG255	DINT	L#0	Incremento para número de incremento 255
104.0	CHGDIF_255	DINT	L#0	Distancia de cambio de velocidad para número de incremento 255
108.0	CUTDIF_255	DINT	L#0	Distancia de desconexión para número de incremento 255

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios	
100.0	CHGDIF_P	DINT	L#5000	Distancia de cambio de velocidad positiva	
104.0	CHGDIF_M	DINT	L#5000	Distancia de cambio de velocidad negativa	
108.0	CUTDIF_P	DINT	L#2000	Distancia de desconexión positiva	
112.0	CUTDIF_M	DINT	L#2000	Distancia de desconexión negativa	
120.0	TRGL1.TRG[1]	DINT	L#0	Número de incremento 1	Tabla de incrementos 1
. . .	. . .	. . .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Número de incremento 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Número de incremento 51	Tabla de incrementos 2

Dirección	Nombre	Tipo	valor inicial	Comentarios	
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Número de incremento 100	

Distancia residual

La distancia residual es la diferencia entre el destino (incremento) y el tiempo real. La distancia residual es

- positivo si aún no ha alcanzado el destino.
- negativo si ya se ha rebasado el destino.

En un eje rotativo no puede utilizarse la distancia residual mostrada.

Desactivación del modo incremental

El modo de operación "Modo incremental" se desactiva si el FM 351 recibe una señal de STOP (STOP = 1).

Tras la desactivación de la marcha se mantiene la distancia residual.

La distancia residual del "Modo incremental relativo" puede realizarse hasta el final si

- el modo de operación no ha cambiado y
- el número de incremento no ha cambiado y
- el sentido no ha cambiado y
- la distancia residual es mayor que la distancia de desconexión parametrizada.

La distancia residual se realiza ejecutando de nuevo el "Modo incremental relativo" sin cambios.

Cancelación del modo incremental

El modo de operación "Modo incremental" se cancela si la señal "Habilitación del accionamiento" se borra (DRV_EN = 0).

Borrar distancia residual

La petición "Borrar distancia residual" borra una distancia residual pendiente.

Al llamar otro modo de operación o iniciar el modo de operación en el otro sentido también se borra la distancia residual pendiente.

Dato utilizado en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.2	DELDIST_EN	BOOL	FALSE	1 = Borrar distancia residual

Petición "Definir valor real" / "Cancelar preselección del valor real"

Definición

Con la petición "Ajustar valor real" se asigna una nueva coordenada al estado actual del encóder. La zona de trabajo se proyecta sobre otra zona del eje.

El desplazamiento de la zona de trabajo se calcula con ($REAL_{nuevo}$ - $REAL_{actual}$).

- $REAL_{nuevo}$ es el valor predeterminado
- $REAL_{actual}$ es el valor real en el instante de la ejecución

Requisitos

- El eje debe estar parametrizado.
- El eje debe estar sincronizado.

Desarrollo de la petición

1. Introduzca las coordenadas AVAL para el valor real ($REAL_{nuevo}$).
   - Eje lineal:
     El valor real indicado se debe seleccionar de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admitido tras la llamada de la petición.
     El valor de decalaje que resulta de ($REAL_{nuevo}$ - $REAL_{actual}$) tiene que ser menor o igual al valor de la zona de desplazamiento admisible (máximo 100 m o 1000 m, respectivamente).
   - Eje rotativo:
     Para el valor real indicado rige:
     0 ≤ Valor real < Final del eje rotativo
2. Active el bit de inicio correspondiente (AVAL_EN = 1).

Si se emite la petición "Ajustar valor real" durante un posicionamiento, se retiene hasta el final del posicionamiento y no se realiza hasta que se llama el bloque.

Datos utilizados en el DB de canal

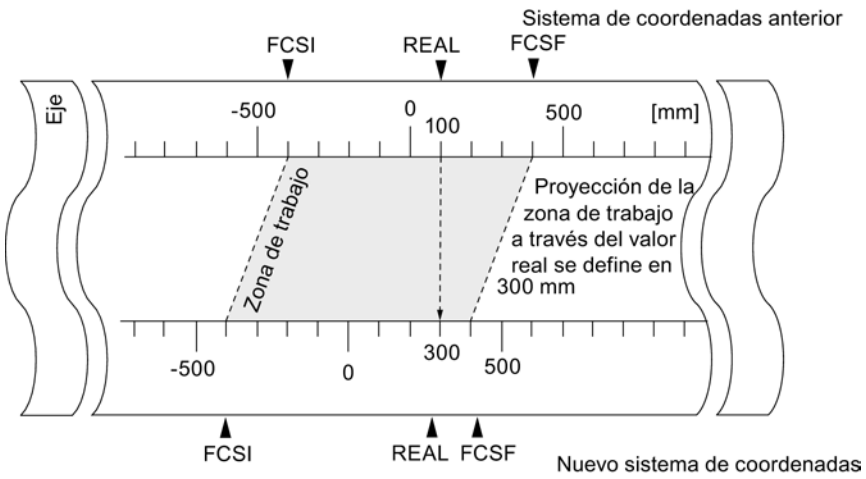
Dirección	Nombre	Tipo	valor inicial	Comentarios
35.7	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
84.0	AVAL	DINT	L#0	Coordenada de "Ajustar valor real"

Efectos de la petición

En el ejemplo "Ajustar valor real" en 300 mm se ve cómo esta petición proyecta la zona de trabajo a una posición concreta del eje. De ello se derivan los siguientes efectos:

- La posición real se fija en el valor de la coordenada del valor real.
- La zona de trabajo se desplaza hacia el eje.
- Los distintos puntos (por ejemplo, fin del final de carrera por software) dentro de la zona de trabajo conservan su valor original, pero se encuentran en posiciones físicas nuevas.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Ajustar valor real".

Ajustar valor real	FCSI	REAL	FCSF
	-400	100	400
	-400	300	400

Deshacer una petición ("Cancelar preselección del valor real")

Con la petición "Cancelar preselección del valor real" se deshacen todos los desplazamientos de la zona de trabajo que se han creado con "Ajustar valor real".

La suma de todos los desplazamientos de la zona de trabajo no puede ser superior a la zona de desplazamiento para que esta petición pueda ejecutarse correctamente.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.3	AVALREM_EN	BOOL	FALSE	1 = Cancelar preselección del valor real

Petición "Definir punto de referencia"

Definición

Con la petición "Definir punto de referencia" se sincroniza el eje. Esta petición desplaza la zona de trabajo. Todos los decalajes generados por la función "Ajustar valor real" se conservan.

Este ajuste proyecta la zona de trabajo sobre el eje. Así, con distintas entradas de valores, la zona de trabajo puede encontrarse en posiciones diferentes dentro del rango físico del eje.

Requisitos

- El posicionamiento debe haber finalizado.
- El eje debe estar parametrizado.

Desarrollo de la petición

1. Introduzca el valor REFPT para las coordenadas del punto de referencia.
   - Eje lineal:

     La coordenada del punto de referencia no puede encontrarse fuera de los finales de carrera por software. Esto rige también para la coordenada del punto de referencia de un sistema de coordenadas desplazado.
   - Eje rotativo:
     Para la coordenada del punto de referencia rige:
     0 ≤ Coordenada del punto de referencia < Final del eje rotativo
2. Active el bit de inicio correspondiente (REFPT_EN = 1).

Datos utilizados en el DB de canal

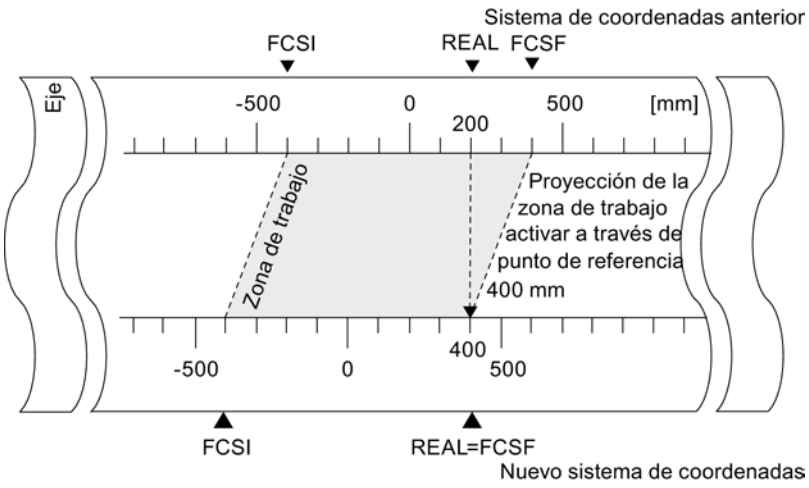
Dirección	Nombre	Tipo	valor inicial	Comentarios
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado
35.6	REFPT_EN	BOOL	FALSE	1 = Definir punto de referencia
92.0	REFPT	DINT	L#0	Coordenada del punto de referencia

Efectos de la petición

En el ejemplo "Definir punto de referencia" en 400 mm se ve cómo esta petición proyecta la zona de trabajo a una posición física concreta del eje. De ello se derivan los siguientes efectos:

- La posición real se fija en el valor de la coordenada del punto de referencia.
- La zona de trabajo se desplaza hacia el eje.
- Los distintos puntos (por ejemplo, fin del final de carrera por software) conservan su valor original pero se encuentran en posiciones físicas nuevas.
- El bit SYNC de las señales de respuesta se activa.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Definir punto de referencia".

Definir punto de referencia	FCSI	REAL	FCSF
	-400	200	400
	-400	400	400

Particularidades del encóder absoluto

Esta petición es necesaria para un reajuste del encóder absoluto (Página 10956).

Selector de función "Aproximación en bucle"

Definición

Con la "Aproximación en bucle" define el sentido en el que se avanza hacia el punto de destino con cierre de fuerza. La aproximación en bucle puede utilizarse cuando sólo se puede garantizar el cierre de fuerza en un sentido entre el motor y el eje.

El destino al que se realiza la aproximación en contra del sentido predefinido se rebasa al principio. A continuación, el módulo realiza una inversión del sentido y realiza la aproximación al destino en el sentido predeterminado.

Requisitos

- El eje debe estar parametrizado.
- El eje debe estar sincronizado.

- En una aproximación en bucle en contra del sentido del desplazamiento hacia el destino, la posición de destino se encuentra:
  - en el sentido de desplazamiento positivo
    Destino < SEE - ½ zona de destino - distancia de desconexión positiva - distancia de cambio de velocidad negativa
  - en el sentido de desplazamiento negativo
    Destino < SEA + ½ zona de destino + distancia de desconexión negativa + distancia de cambio de velocidad positiva
- Una aproximación en bucle no se lleva a cabo, si la aproximación al destino se realiza en el sentido de la aproximación en bucle. En este caso se realiza un posicionamiento en modo incremental sin inversión del sentido.
- Hay que conocer el transcurso del modo de operación "Modo incremental" (Página 10920).

Desarrollo de la aproximación en bucle

1. Active la señal de control para el modo de operación "Modo incremental absoluto/relativo" (MODE_IN = 5/4).
2. Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).
3. Active el selector de función "Entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.
4. Introduzca el número de incremento (MODE_TYPE = 1 … 100, 254, 255).
5. Active el selector de función "Aproximación en bucle" (PLOOP_ON / MLOOP_ON = 1).
6. Inicie el modo incremental.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.0	START	BOOL	FALSE	1 = Iniciar posicionamiento
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	4 / 5 = Modo incremental relativo/absoluto
17.0	MODE_TYPE	BYTE	B#16#0	Número de incremento 1 … 100, 254 ó 255
34.0	PLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido positivo
34.1	MLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido negativo
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

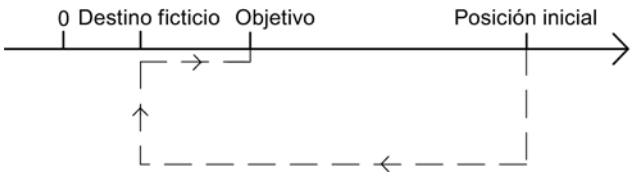
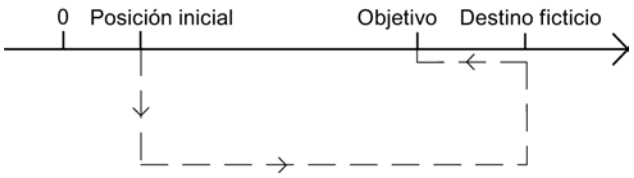
Destino ficticio

Si inicia un posicionamiento en un destino que se encuentra en el sentido opuesto a la aproximación en bucle parametrizada, el módulo determina para este destino un destino

ficticio en el que éste ejecuta una inversión del sentido y, después, se aproxima al destino en el sentido correcto.

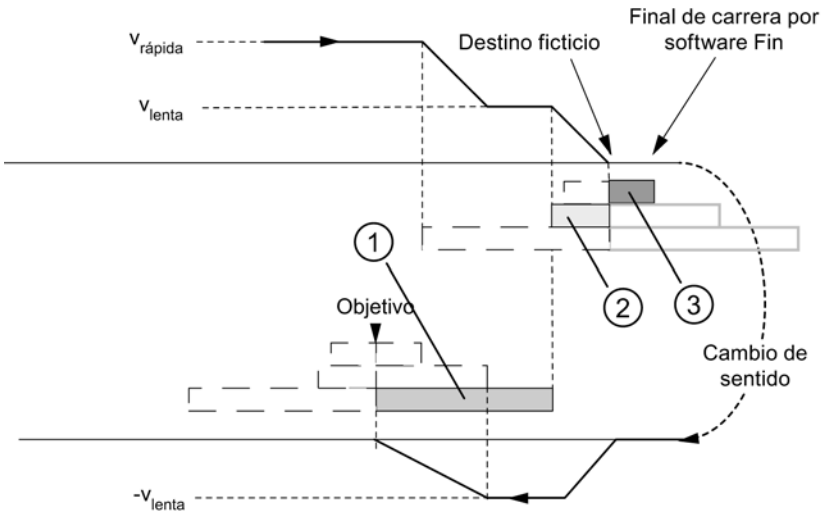
Este destino ficticio **tiene que** encontrarse al menos a media zona de destino delante del final de carrera por software correspondiente.

La eliminación del destino ficticio del destino parametrizado se determina en función del sentido. La tabla siguiente muestra el cálculo de la situación del destino ficticio en la aproximación en bucle.

Preajustes	Posición del destino ficticio
Parametrización: Bucle + (cierre de fuerza positivo) y marcha en sentido negativo	El destino ficticio (**Destino**$_f$) tiene el valor: **Destino**$_f$ **= Destino - Distancia de desconexión negativa - distancia de cambio de velocidad positiva** 
Parametrización: Bucle - (cierre de fuerza negativo) y marcha en el sentido positivo	El destino ficticio (**Destino**$_f$) tiene el valor: **Destino**$_f$ **= Destino + Distancia de desconexión positiva + distancia de cambio de velocidad negativa** 

Ejemplo

A continuación se muestra la posición del destino ficticio tomando como base un posicionamiento con aproximación en bucle negativa hacia un destino máximo.



① Distancia de cambio de velocidad negativa

② Distancia de desconexión positiva

③ ½ zona de destino

Selector de función "Entrada de habilitación"

Definición

La entrada de habilitación es una entrada externa con la que puede habilitarse un posicionamiento a través de un evento externo.

Evaluar entrada de habilitación (EI_OFF = 0)

La entrada de habilitación correspondiente (xI2) para el canal debe estar cableada.

Con ello tiene la posibilidad de preparar el inicio de un posicionamiento. El posicionamiento se inicia independientemente del desarrollo del programa de usuario creando una señal "1" en la entrada de habilitación.

La marcha comienza cuando se crea una señal "1" en la entrada de habilitación y se desactiva cuando se crea una señal "0" en la entrada de habilitación.

No evaluar entrada de habilitación (EI_OFF = 1)

Si desconecta la evaluación de la entrada de habilitación, un modo de operación comienza inmediatamente después de detectar la señal de inicio. A continuación no es posible preparar un modo de operación ni iniciarlo en un instante posterior concreto.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Petición "Leer datos de posición"

Definición

Con la petición "Leer datos de posición" es posible leer el incremento, la distancia residual y la velocidad en el instante actual.

Desarrollo de la petición

1. Active el bit de inicio en el DB de canal (ACTSPD_EN = 1).
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
37.1	ACTSPD_EN	BOOL	FALSE	1 = Leer datos de posición
112.0	ACTSPD	DINT	L#0	Velocidad actual
116.0	DIST_TO_GO	DINT	L#0	Distancia residual
120.0	ACT_TRG	DINT	L#0	Incremento actual

Petición "Leer datos del encóder"

Definición

Con la petición "Leer datos de encóder" se leen los datos actuales del encóder, así como el valor del reajuste del encóder absoluto.

Requisitos

Es posible leer el valor para el reajuste del encóder absoluto (Página 10956) después de ejecutar la petición "Definir punto de referencia" (Página 10928).

Desarrollo de la petición

1. Active el bit de inicio en el DB de canal (ENCVAL_EN = 1).
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
37.2	ENCVAL_EN	BOOL	FALSE	1 = Leer valores del encóder
124.0	ENCVAL	DINT	L#0	Valor real del encóder (representación interna)
128.0	ZEROVAL	DINT	L#0	Último valor de la marca cero (representación interna)
132.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto

Consultar señales de respuesta del posicionamiento

Definición

Con la función "Señales de respuesta para el posicionamiento" se informa del estado actual del posicionamiento.

Desarrollo

1. Con cada llamada de la instrucción ABS_CTRL (Página 7420) los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
23.4	SPEED_OUT	BOOL	FALSE	0 = Marcha lenta 1 = Marcha rápida
23.5	ZSPEED	BOOL	FALSE	1 = El eje se encuentra en la zona de parada
23.6	CUTOFF	BOOL	FALSE	1 = El eje se encuentra en el área de desconexión
23.7	CHGOVER	BOOL	FALSE	1 = El eje se encuentra en la zona de cambio de velocidad
24.0	MODE_OUT	BYTE	B#16#0	Modo de operación activo
25.2	GO_M	BOOL	FALSE	1 = El eje se desplaza en sentido negativo
25.3	GO_P	BOOL	FALSE	1 = El eje se desplaza en sentido positivo
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada
26.0	ACT_POS	DINT	L#0	Valor real actual (posición actual del eje)

Consultar las señales de realimentación del diagnóstico

Definición

Con las "Señales de respuesta del diagnóstico" se recibe información sobre los eventos de diagnóstico que se han producido.

Desarrollo

1. Cuando el módulo introduce un nuevo evento en el búfer de diagnóstico, activa el bit DIAG en la interfaz de respuesta. Cada aparición de un error (Página 10984) provoca una entrada en el búfer de diagnóstico. Al borrar el búfer de diagnóstico se activa el bit DIAG.
2. Si no es posible llamar un modo de operación o controlar un modo de operación activo, o si esta operación se ha realizado de forma errónea, el módulo activa un error de operador OT_ERR en la interfaz de respuesta. La causa del error se registra en el búfer de diagnóstico. Mientras el error de operador está activo, no es posible iniciar un modo de operación ni continuar el modo de operación detenido. Un error de operador existente se confirma con OT_ERR_A = 1.
3. Cuando el módulo detecta una petición de escritura con datos erróneos, activa el bit DATA_ERR en la interfaz de respuesta. La causa del error se registra en el búfer de diagnóstico.
4. Las señales de respuesta se depositan en el DB de canal.
5. Si se ha leído el búfer de diagnóstico, el módulo restablece el bit DIAG a 0 en todos los canales.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
22.2	DIAG	BOOL	FALSE	1 = Búfer de diagnóstico modificado
22.3	OT_ERR	BOOL	FALSE	1 = Error de operador
22.4	DATA_ERR	BOOL	FALSE	1 = Error de datos

Configurar FM 351

Parametrizar el FM 351

Especificar canales

Seleccionar sistema de medidas

Seleccionar sistema de medidas

Para la entrada y salida de datos es posible elegir entre los siguientes sistemas de medida:

- Milímetros (ajuste predeterminado)
- Inch
- Grados

Nota

Si se modifica el sistema de medida en el cuadro de diálogo "Especificar canales", los valores se convierten al nuevo sistema. Aquí se pueden producir errores de redondeo.

Si se modifica el sistema de medida mediante el dato máquina, los valores no se convierten automáticamente.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentario
8.0	UNITS	DINT	L#1	**Sistema de medida** Valores permitidos: • 1 = 10^{-3} mm • 2 = 10^{-4} pulgadas • 3 = 10^{-4} grados • 4 = 10^{-2} grados • 6 = 10^{-3} grados

Sistema de medida estándar en la ayuda en pantalla

Para la indicación de valores mínimos y máximos se utiliza el sistema de medida mm. Por ello, para determinar los valores límite en los otros sistemas de medida, realice la conversión siguiente:

Para convertir de		calcule
mm → pulgadas		Valor límite (inch) = valor límite (mm) × 0,1 [1]
mm → grados	10^{-4} (4 decimales)	Valor límite (grados) = valor límite (mm) × 0,1
	10^{-3} (3 decimales)	Valor límite (grados) = valor límite (mm) × 1
	10^{-2} (2 decimales)	Valor límite (grados) = valor límite (mm) × 10
pulgadas → grados		Valor límite (grados) = valor límite (pulgadas)

[1] En el valor máximo el número de decimales repercute en el número de dígitos que preceden a la coma. En el sistema de medida Inch se utilizan 4 decimales, por lo tanto se pueden introducir como máximo 100 000,0000 inch. El sistema de medida Milímetros utiliza 3 decimales y por lo tanto se pueden introducir como máximo 1 000 000,000 mm.

Relación entre los incrementos y el sistema de medida

Las señales del encóder de un dispositivo conectado son evaluadas por el módulo y se convierten al sistema actual. Para la conversión se utiliza la resolución (Página 10963).

Si el módulo de posicionamiento

- ha contado 10 incrementos y
- ha ajustado una resolución de 100 µm por incremento a través de los datos de encóder parametrizados,

esto significa que el eje se ha movido un trayecto de 1 mm.

Copiar canales

Copiar canales

Este parámetro permite copiar un canal en otro canal.

1. Seleccione el canal cuyos parámetros desee copiar (origen del canal).
2. Active el canal en el que desee copiar los parámetros (destino del canal).
3. Pulse el botón "Copiar".

Accionamiento del canal n

Seleccionar el modo de control

Visión global de los tipos de control

Sinopsis

El modo de control define el modo en que se utiliza un motor en las 4 salidas digitales (Q) de un canal a través de la etapa de potencia. Para el módulo de posicionamiento existen 4 modos de control por canal:

- Modo de control 1 (Página 10942)
- Modo de control 2 (Página 10943)
- Modo de control 3 (Página 10944)
- Modo de control 4 (Página 10945)

La siguiente tabla muestra los estados de las 4 salidas digitales para cada modo de control, donde x representa los canales 1 y 2.

	Marcha rápida		Marcha lenta		PEH La posición alcanza la parada
	Sentido +	Sentido -	Sentido +	Sentido -	
Modo de control 1					
xQ0	1	1	0	0	-
xQ1	0	0	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-

	Marcha rápida		Marcha lenta		PEH La posición alcanza la parada
	Sentido +	Sentido -	Sentido +	Sentido -	
Modo de control 2					
xQ0	1	1	0	0	0
xQ1	0	0	0	0	1
xQ2	1	0	1	0	0
xQ3	0	1	0	1	0
Modo de control 3					
xQ0	1	1	0	0	-
xQ1	1	1	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-
Modo de control 4					
xQ0	1	0	0	0	-
xQ1	1	0	1	0	-
xQ2	0	1	0	0	-
xQ3	0	1	0	1	-

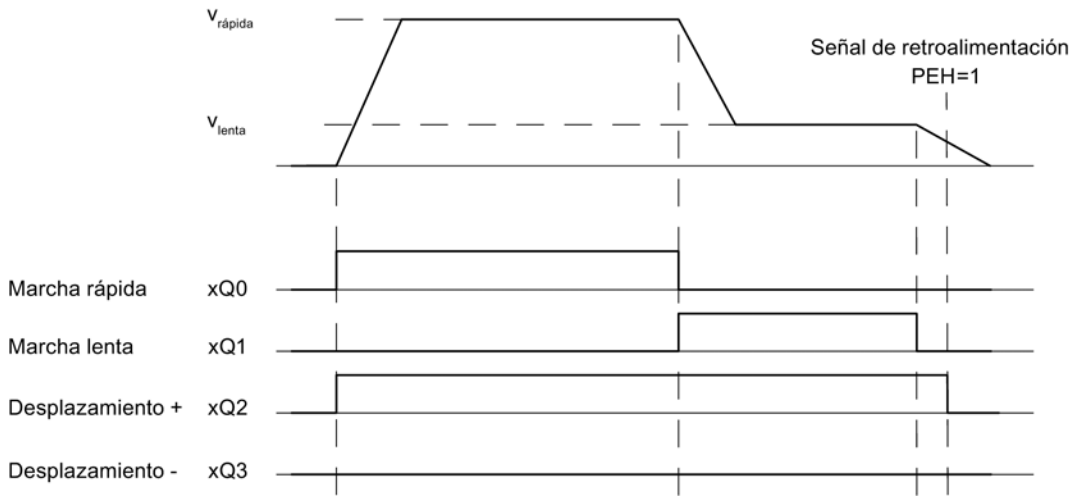
Modo de control 1

Modo de control 1

En el modo de control 1 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida
- xQ1: Marcha lenta
- xQ2: Desplazamiento +
- xQ3: Desplazamiento -

La figura siguiente muestra la función de las salidas digitales en el modo de control 1.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	1 = Modo de control 1

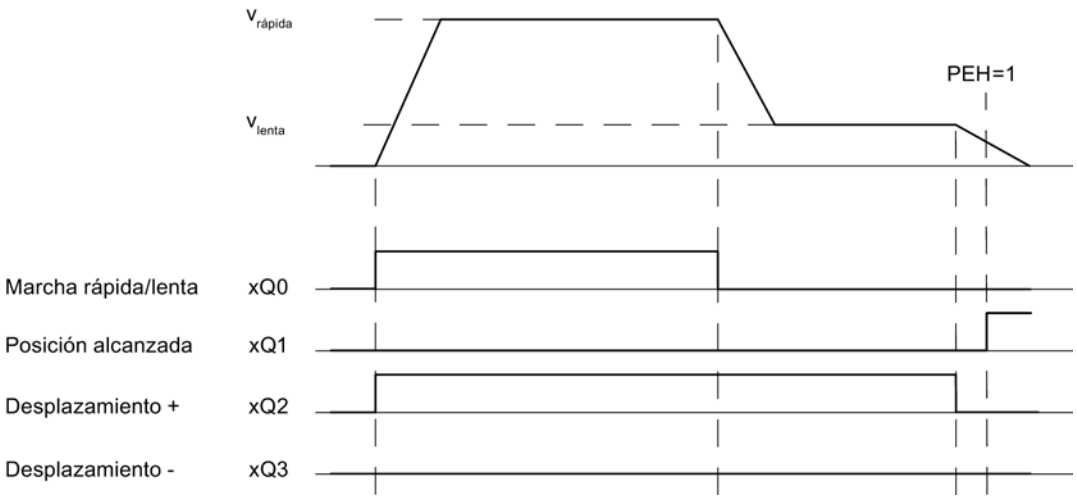
Modo de control 2

Modo de control 2

En el modo de control 2 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida/lenta
- xQ1: Posición alcanzada
- xQ2: Desplazamiento +
- xQ3: Desplazamiento -

La figura siguiente muestra la función de las salidas digitales en el modo de control 2.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	2 = Modo de control 2

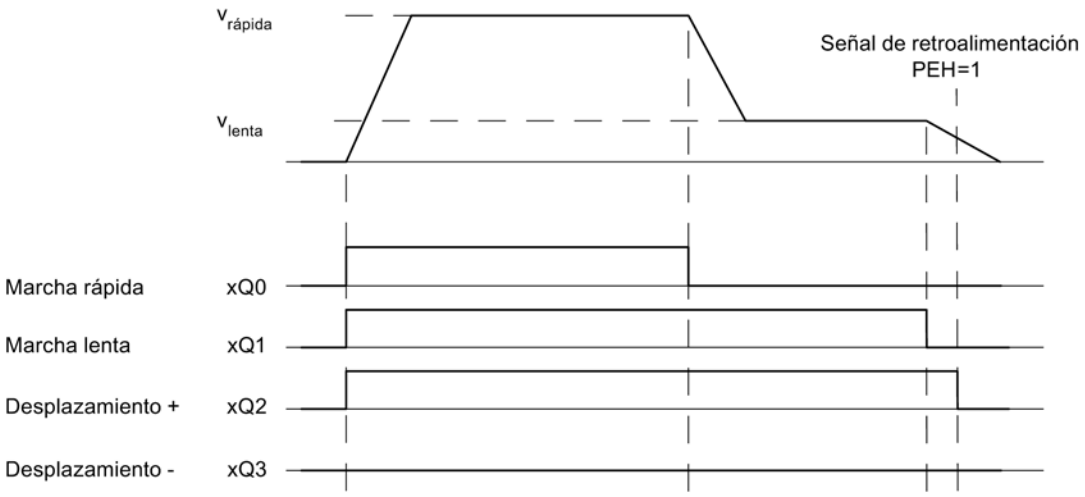
Modo de control 3

Modo de control 3

En el modo de control 3 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida
- xQ1: Marcha lenta
- xQ2: Desplazamiento +
- xQ3: Desplazamiento -

La figura siguiente muestra la función de las salidas digitales en el modo de control 3.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	3 = Modo de control 3

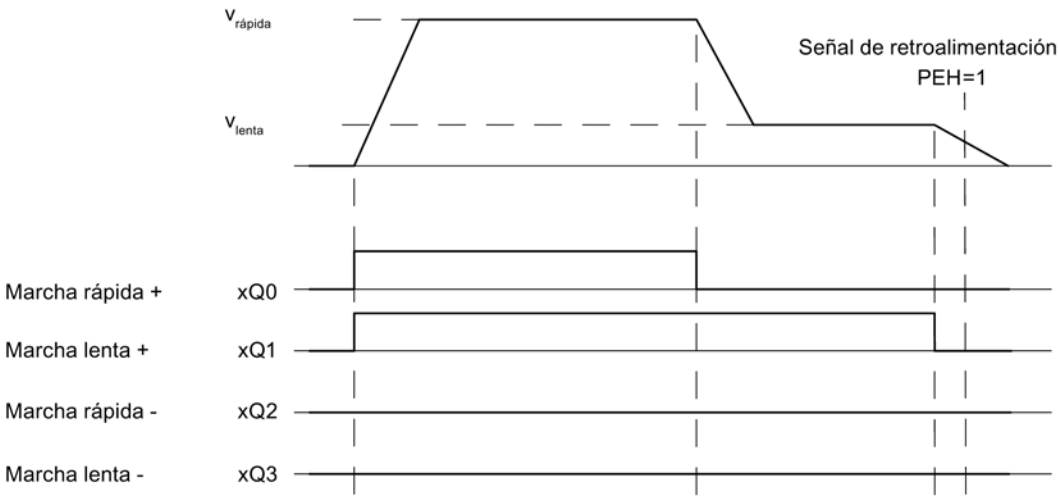
Modo de control 4

Modo de control 4

En el modo de control 4 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida +
- xQ1: Marcha lenta +
- xQ2: Marcha rápida -
- xQ3: Marcha lenta -

La figura siguiente muestra la función de las salidas digitales en el modo de control 4.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	4 = Modo de control 4

Definir distancia de cambio de velocidad

Distancia de cambio de velocidad

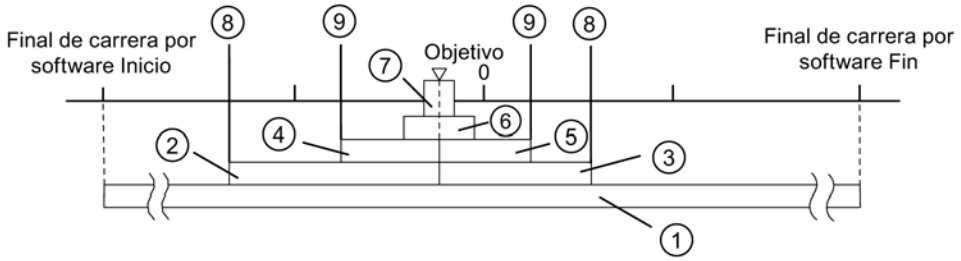
La "Distancia de cambio de velocidad" define la distancia con respecto al punto de destino en el que el accionamiento cambia la velocidad, de velocidad rápida a velocidad lenta.

Los valores predefinidos se aplican a todos los objetivos a los que se aproxima el módulo de posicionamiento. Una excepción está constituida por el modo incremental con el incremento 255, para la que se define una distancia de cambio de velocidad especial.

Reglas para la distancia de cambio de velocidad

- La distancia de cambio de velocidad positiva y la distancia de cambio de velocidad negativa no tienen por qué estar dispuestas de forma simétrica en torno al destino. Los valores pueden ser muy distintos.
- La distancia de cambio de velocidad debe ser mayor que la distancia de desconexión.
- La distancia de cambio de velocidad se encuentra dentro de la zona de trabajo.
- La distancia de cambio de velocidad debe ser menor que el final del eje rotativo.
- Entre el punto de cambio de velocidad y el punto de desconexión se debe seleccionar una distancia tan grande que permita cambiar de marcha rápida a marcha lenta.
- El punto de cambio de velocidad y el punto de desconexión se encuentran a una distancia temporal de al menos 8 ms entre sí.

La figura siguiente muestra la ubicación de las áreas de cambio de velocidad en torno al destino.



①	Zona de trabajo	⑦	Zona de destino
② ③	Distancia de cambio de velocidad positiva/negativa	⑧	Punto de cambio de velocidad
④ ⑤	Distancia de desconexión positiva/negativa	⑨	Punto de desconexión
⑥	Zona de parada		

Si desea obtener más información sobre la disposición de las áreas, consulte "Zonas y puntos de cambio de velocidad del módulo de posicionamiento".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CHGDIF_P	DINT	L#5000	**Distancia de cambio de velocidad positiva**
104.0	CHGDIF_M	DINT	L#5000	**Distancia de cambio de velocidad negativa** Rango de valores admisible: • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir distancia de desconexión

Distancia de desconexión

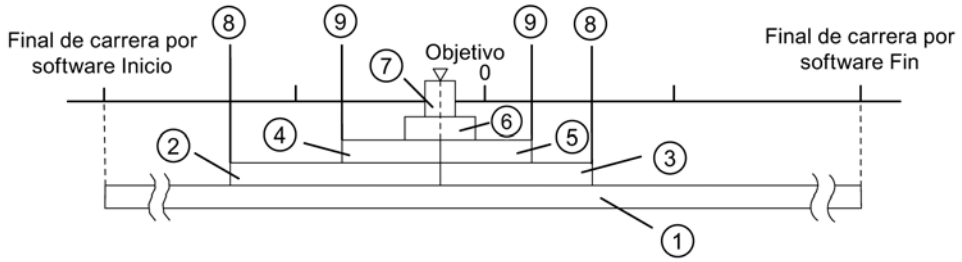
La "Distancia de desconexión" define la distancia hasta el punto de destino en el que se desconecta el accionamiento (la marcha lenta). Las funciones de vigilancia retoma el módulo de posicionamiento a partir de este punto.

Los valores predefinidos se aplican a todos los objetivos a los que se aproxima el módulo de posicionamiento. Una excepción está constituida por el modo incremental con el incremento 255, para la que se define una distancia de desconexión especial.

Reglas para la distancia de desconexión

- La distancia de desconexión positiva y la distancia de desconexión negativa no tienen por qué estar dispuestas de forma simétrica en torno al destino. Los valores pueden ser muy distintos.
- La distancia de desconexión debe ser mayor que la mitad de la zona de destino.
- Entre el punto de desconexión y el destino se debe seleccionar una distancia que permita que el accionamiento se detenga dentro de la zona de destino.
- El punto de desconexión y el inicio de la zona de destino se encuentran a una distancia temporal de al menos 8 ms entre sí.

La figura siguiente muestra la ubicación de las zonas de cambio de velocidad en torno al destino.



①	Zona de trabajo	⑦	Zona de destino
② ③	Distancia de cambio de velocidad positiva/negativa	⑧	Punto de cambio de velocidad
④ ⑤	Distancia de desconexión positiva/negativa	⑨	Punto de desconexión
⑥	Zona de parada		

Si desea obtener más información sobre la disposición de las áreas, consulte "Zonas y puntos de conexión del módulo de posicionamiento".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
108.0	CUTDIF_P	DINT	L#2000	**Distancia de desconexión positiva**
112.0	CUTDIF_M	DINT	L#2000	**Distancia de desconexión negativa** Rango de valores admisible: • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir aproximación al destino

Definir zona de destino

Zona de destino

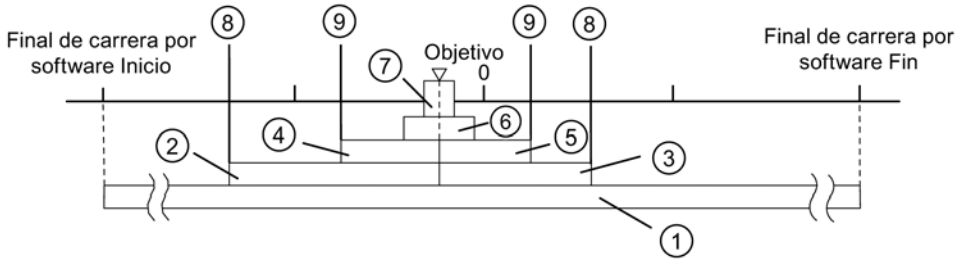
El módulo de posicionamiento crea un área simétrica alrededor de cada destino: la zona de destino. Dentro de esta área es preciso que el valor real se encuentre en una aproximación al destino.

Una especificación de valor "0 "desconecta la tolerancia en la aproximación al destino. La velocidad al alcanzar el destino debe ser entonces inferior a la velocidad de parada (Página 10951) parametrizada.

Reglas para la zona de destino

- La zona de destino está situada de forma simétrica en torno al destino.
- Una especificación de valor "0 "desconecta la vigilancia de la zona de destino.

La figura siguiente muestra la ubicación de las zonas en torno al destino.



①	Zona de trabajo	⑦	Zona de destino
② ③	Distancia de cambio de velocidad positiva/negativa	⑧	Punto de cambio de velocidad
④ ⑤	Distancia de desconexión positiva/negativa	⑨	Punto de desconexión
⑥	Zona de parada		

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Zonas y puntos de conexión del módulo de posicionamiento", así como "Fin de un proceso de posicionamiento (Página 10904)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
76.0	TRG_RANGE	DINT	L#1000	**Zona de destino** Rango de valores admisible: • 0 = sin vigilancia • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir tiempo de vigilancia

Tiempo de vigilancia

Durante el tiempo de vigilancia el módulo vigila:

- el movimiento del eje hasta el punto de desconexión.
  El tiempo de vigilancia comienza con el inicio de un posicionamiento y se redispara cada vez que se modifica el valor real en el sentido de la marcha.
  Dentro del tiempo de vigilancia, el valor real debe haberse modificado al menos un incremento (la parte del trayecto de la resolución).
- la aproximación al destino.
  El tiempo de vigilancia se redispara por última vez cuando se alcanza el punto de desconexión.
  Dentro del tiempo de vigilancia debe haberse alcanzado la zona de destino.

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Final de un proceso de posicionamiento (Página 10904)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
80.0	MON_TIME	DINT	L#2000	**Tiempo de vigilancia** Rango de valores admisible: • 0 = sin vigilancia • De 1 a 100.000 ms
Tiempo de vigilancia real Para el tiempo vigilancia es posible predefinir todos los valores a partir del área de valores. El módulo redondea el tiempo predefinido a un múltiplo de 8 ms (ciclo de módulo).				

Definir zona de parada

Zona de parada

La zona de parada está situada de forma simétrica en torno al destino correspondiente. En esta área se vigila si el accionamiento se detiene en una posición de destino alcanzada o si se aleja de ella. La vigilancia comienza después de que el módulo de posicionamiento ha notificado "Posición alcanzada" (PEH).

Si la zona de parada se abandona sin una petición de marcha válida, el módulo de posicionamiento notifica un error.

Reglas para la zona de parada

- La zona de parada está situada de forma simétrica en torno al destino correspondiente: (Destino – ½ × zona de parada) ≤ destino ≤ (destino + ½ × zona de parada)
- La zona de parada debe ser mayor o igual que la zona de destino.
- Los valores de coordenadas para la zona de parada se encuentran en la zona de trabajo.
- Con el valor "0" se desconecta la vigilancia de parada.

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Zonas y puntos de conexión del módulo de posicionamiento", así como "Fin de un proceso de posicionamiento (Página 10904)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
84.0	ZSPEED_R	DINT	L#1000	**Zona de parada** Rango de valores admisible: • 0 = sin vigilancia • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir velocidad de parada

Velocidad de parada

La velocidad de parada sirve como velocidad de referencia. Si la velocidad real alcanza la velocidad de parada durante un posicionamiento y el accionamiento se encuentra en la zona de destino, se notifica el aviso "Posición alcanzada" (PEH). El proceso de posicionamiento queda correctamente finalizado.

La precisión de posicionamiento aumenta cuando el valor de la velocidad de parada se reduce. Con el valor "0" se desconecta la vigilancia de parada.

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Final de un proceso de posicionamiento (Página 10904)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
88.0	ZSPEED_L	DINT	L#30000	**Velocidad de parada** Rango de valores admisible: • 0 = sin vigilancia • 1 µm/min a 100 000 µm/min

Eje del canal n

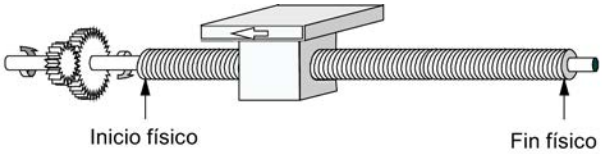
Definir tipo de eje

Seleccionar tipo de eje

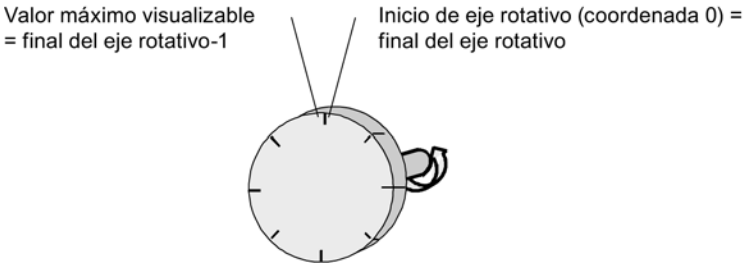
Tipos de eje

El módulo de posicionamiento puede controlar los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento físico limitado.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
12.0	AXIS_TYPE	DINT	L#0	**Tipo de eje** Valores permitidos: • 0 = Eje lineal • 1 = Eje rotativo

Definir final de carrera por software

Final de carrera por software

Los finales de carrera por software existen sólo para el eje lineal.

El área limitada por el final de carrera por software es la zona de trabajo.

El inicio del final de carrera por software (FCSF) debe ser siempre menor que el fin del final de carrera por software (SEE).

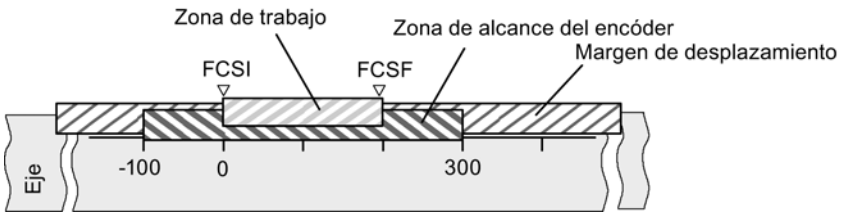
Los finales de carrera por software se vigilan después de que el eje está sincronizado.

- Si se utiliza un encóder incremental:
  La sincronización del eje se alcanza mediante la búsqueda del punto de referencia o la definición del punto de referencia.
- Si se utiliza un encóder absoluto (SSI):
  El eje está sincronizado después de que el módulo ha recibido un telegrama completo y sin errores para el canal correspondiente.
  El encóder absoluto tiene que abarcar por lo menos la zona de trabajo, incluido el final de carrera por software.

Relación entre la zona de trabajo, el área del encóder y la zona de desplazamiento

- La zona de trabajo es el área que el usuario define para sus tareas mediante los finales de carrera por software.
- El área del encóder es el área que el encóder cubre claramente. En el caso de un eje lineal del módulo, el área del encóder se define de forma simétrica por encima de la zona de trabajo. Es decir, el módulo desplaza el área del encóder de tal modo que las distancias entre los finales de carrera por software y los finales del área del encóder son iguales en cada caso (véase la figura inferior).
- La zona de desplazamiento es el rango de valores que puede procesar el módulo de posicionamiento. La zona de desplazamiento depende de la resolución.

La figura siguiente muestra la relación entre los finales de carrera por software, la zona de trabajo, el área del encóder y la zona de desplazamiento.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
64.0	SSW_STRT	DINT	L#-100000000	**Inicio del final de carrera por software**
68.0	SSW_END	DINT	L#100000000	**Fin del final de carrera por software** Rango de valores admisible: • -1 000 000 000 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • -100.000.000 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir final del eje rotativo

Final del eje rotativo

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar. Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo (0).

El valor más grande que se muestra, en el caso de un eje rotativo, tiene el valor:

Final del eje rotativo [µm] - Resolución [µm / impulso] × 1 [impulso]

Ejemplo: Final del eje rotativo 1000 mm, resolución 1000 µm/impulso

La visualización pasa:

- con un sentido de giro positivo, de 999 mm a 0 mm
- con un sentido de giro negativo, de 0 mm a 999 mm

Eje rotativo con encóders absolutos

En el caso de un eje rotativo con un encóder absoluto, el área del eje rotativo (0 hasta el final del eje rotativo) tiene que coincidir exactamente con el área de sensor del encóder absoluto:
Fin del eje rotativo [µm] = Número de vuelltas del encóder × Recorrido/vuelta del encóder [µm]

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
16.0	ENDROTAX	DINT	L#100000	**Final del eje rotativo** Rango de valores admisible: • 1 µm a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir búsqueda del punto de referencia

Seleccionar tipo de búsqueda del punto de referencia

Tipo de búsqueda del punto de referencia

Con la función "Tipo de búsqueda del punto de referencia" se determinan las condiciones para la sincronización del eje:

- El primer dato define el sentido de arranque en el que comienza la búsqueda del punto de referencia.
- El segundo dato define la posición de la marca cero que conduce a la sincronización, en referencia con el conmutador de punto de referencia.

Si desea obtener información detallada al respecto consulte Modo de operación "Búsqueda del punto de referencia" (Página 10914).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
52.0	REFPT_TYPE	DINT	L#0	**Tipo de búsqueda del punto de referencia** Valores permitidos: • 0 = positivo, conmutador del punto de referencia en el sentido + • 1 = positivo, conmutador del punto de referencia en el sentido - • 2 = negativo, conmutador del punto de referencia en el sentido + • 3 = negativo, conmutador del punto de referencia en el sentido -

Definir coordenada del punto de referencia

Coordenada del punto de referencia

- **Encóders inc.:**
  La coordenada del punto de referencia es necesaria para el modo de operación "Búsqueda del punto de referencia". Si el eje no está sincronizado después de escribir y activar los datos máquina, el valor real se establece al valor de la coordenada del punto de referencia.
- **Encóder absoluto (SSI)**
  La coordenada del punto de referencia es necesaria para el ajuste mecánico del encóder. Consulte al respecto también la descripción del reajuste del encóder absoluto (Página 10956), en la que se explica la interacción de dicho ajuste con el resto de datos.

El valor de la coordenada del punto de referencia debe encontrarse dentro de la zona de trabajo:

- **Eje lineal**
  Incluido el final de carrera por software.
- **Eje rotativo**
  Mayor o igual que 0 y menor que el valor "Final del eje rotativo" (0 ≤ Coordenada del punto de referencia < Final del eje rotativo).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1 000.000.000 µm a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • -100.000.000 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir reajuste del encóder absoluto

Calcular reajuste del encóder absoluto

Reajuste del encóder absoluto

El ajuste del encóder absoluto sólo es necesario para un encóder absoluto (SSI).

Con el reajuste del encóder absoluto y las coordenadas del punto de referencia el rango de valores del sensor se reproduce claramente en el sistema de coordenadas del eje.

Encontrará más información al respecto en "Procedimiento de reajuste del encóder absoluto (Página 10957)", así como en un ejemplo (Página 10958).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1.000.000.000 µm a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • -100.000.000 µm a 100.000.000 µm a una resolución < 1 µm/impulso
48.0	ENC_ADJ	DINT	L#0	**Reajuste del encóder absoluto** Rango de valores admisible: • 0 a (2^{25} - 1)

Procedimiento para un reajuste del encóder absoluto

Determinar el reajuste correcto del encóder absoluto

Tras la primera parametrización hay que llevar a cabo otros pasos para que se establezca una relación correcta entre el encóder y el sistema de coordenadas. El procedimiento se representa utilizando el cuadro de diálogo de parametrización.

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Fin del final de carrera por software".
2. Llame la petición "Definir punto de referencia (Página 10928)" con la coordenada del punto definido en 1.
   El módulo de posicionamiento determina entonces un valor de encóder para la coordenada del punto de referencia registrada en el eje en el DB de canal (REFPT): el reajuste del encóder absoluto. Este valor se puede leer con la petición "Leer datos de encóder (Página 10933)".
3. Registre el valor leído en el campo "Reajuste del encóder absoluto".
4. Guarde la parametrización en el DB de parámetros correspondiente con la función de exportación.
5. Cargue los datos de HWCN en la CPU.
6. Para activar los datos, rearranque la CPU.

Nota

Esta sincronización se realiza una vez en la puesta en marcha. Tras una parametrización, el módulo de posicionamiento queda sincronizado durante el arranque en cuanto se recibe un telegrama sin errores del encóder.

Alternativa: Reajuste mecánico de un encóder

También se puede obtener una relación correcta entre el sistema de coordenadas y el encóder del modo siguiente:

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Inicio del final de carrera por software".
2. Introduzca este valor de coordenadas en los datos máquina a modo de "coordenada del punto de referencia".
3. Lea el valor del encóder que se muestra en esta posición con la petición "Leer datos del encóder (Página 10933)".
4. Introduzca este valor como reajuste del encóder absoluto en los datos máquina.

Tras la parametrización se mostrará siempre un valor real correcto.

En lugar del paso 3 y 4 también es posible resetear a cero el codificador a través de la opción "Reset", si está disponible. A continuación, introduzca el valor "0" como reajuste del encóder absoluto en los datos máquina.

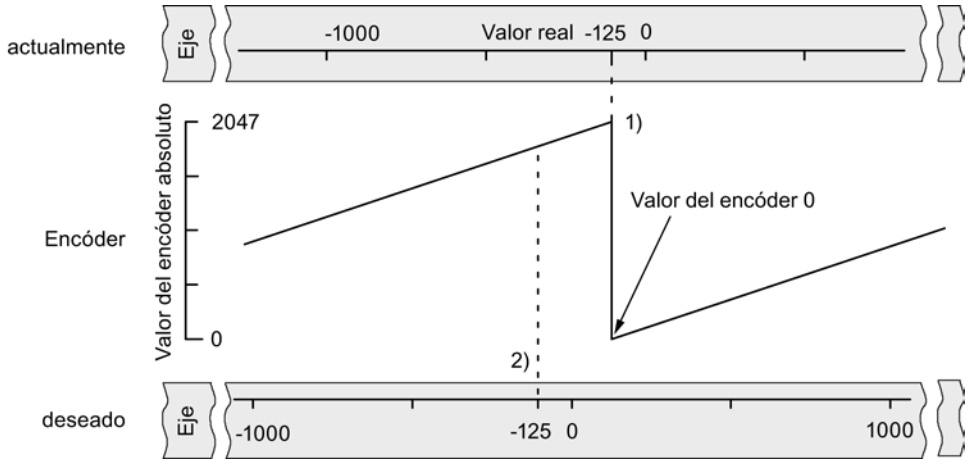
Ejemplo para un reajuste del encóder absoluto

Ejemplo para un reajuste del encóder absoluto

Para este ejemplo se parte de las premisas siguientes:

- Coordenada del punto de referencia = -125 mm
- Área de trabajo de SSW_STRT = -1.000 mm a SSW_END = 1.000 mm
- Reajuste del encóder absoluto = 0
- Área del encóder = 2.048 incrementos con una resolución de 1 mm/impulso
- El encóder absoluto utilizado no permite un ajuste mecánico exacto y tampoco ofrece la posibilidad de definir el valor del encóder.

La figura siguiente muestra un ejemplo de reajuste del encóder absoluto.



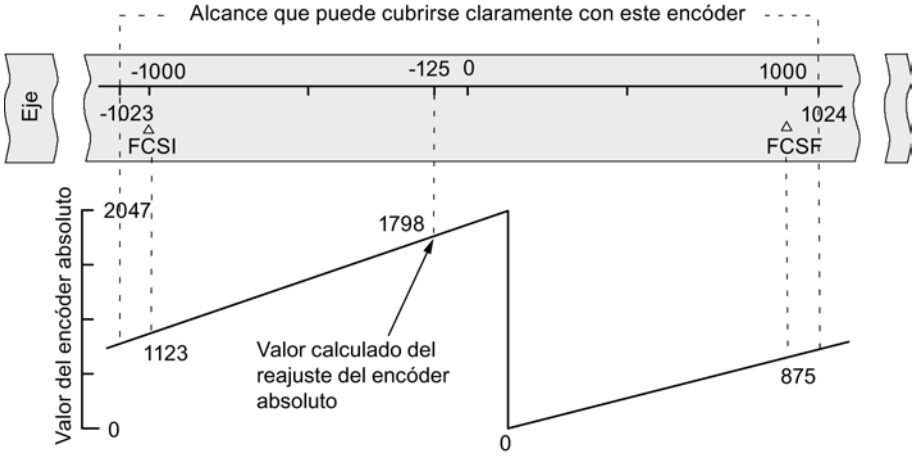
1) Asignación del sistema de coordenadas a los valores del sensor con el reajuste de encóder absoluto definido. El valor del encóder 0 equivale al valor real -125.

2) Asignación deseada del sistema de coordenadas con respecto al encóder. En esta posición, la coordenada tiene que ser -125.

Resultado tras "Definir el punto de referencia"

Tras "Definir el punto de referencia", la relación es la siguiente:

La coordenada del punto de referencia en el eje (-125) se asigna al valor del encóder calculado (1798) a partir del reajuste del encóder absoluto. La figura siguiente muestra esta asignación.



El encóder devuelve 2.048 valores unívocos. La zona de trabajo está definida por el final de carrera por software. Sin embargo, debido a la resolución seleccionada de 1 mm por impulso, el encóder puede abarcar una zona de trabajo mayor que la prevista por el final de carrera por software.

Con la resolución definida, la zona de trabajo está ya cubierta con 2.001 valores. Por ello en el ejemplo quedan 47 impulsos "sobrantes" dispuestos simétricamente alrededor de la zona de trabajo.

Seleccionar velocidad de inicio

Velocidad de inicio en la búsqueda del punto de referencia

Con este dado se selecciona la velocidad para el inicio de una búsqueda del punto de referencia.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
99.0	REFPT_SPD	BOOL	TRUE	**Velocidad de inicio en la búsqueda del punto de referencia** Valores permitidos: • 0 = Marcha rápida • 1 = Marcha lenta

Encóders del canal n

Seleccionar tipo de señal

Tipo de señal

El encóder da información de recorrido al módulo, el cual la evalúa y la convierte en un valor real con la resolución.

Es posible seleccionar un tipo de señal de uno de los encóders siguientes:

- Encóder incremental de 5 V
  Seleccione el tipo de señal "5 V Señal diferencial" si utiliza un encóder incremental de 5 V con señales diferenciales A, /A, B, /B y N, /N.
- Encóder incremental de 24 V
  Seleccione el tipo de señal "24 V pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A*, B* y N*.
- Encóder absoluto (SSI)
  Seleccione el tipo de señal "SSI absoluto" si desea utilizar un encóder absoluto con una interfaz en serie.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
20.0	ENC_TYPE	DINT	L#1	**Tipo de señal** Valores permitidos: • 1 = 5 V Señal diferencial • 2 = 24 V pista A+B desfasado • 3 = SSI longitud del telegrama 13 bits [1)] • 4 = SSI longitud del telegrama 25 bits [1)] • 5 = SSI longitud del telegrama 14 bits [1)] • 6 = SSI longitud del telegrama 15 bits [1)] • 7 = SSI longitud del telegrama 16 bits [1)] • : • : • 14 = SSI longitud del telegrama 23 bits [1)] • 15 = SSI longitud del telegrama 24 bits [1)]

[1)] La selección de la longitud del telegrama se realiza en el cuadro de diálogo "SSI".

Seleccionar sentido de contaje

Sentido de contaje

Con el dato máquina "Sentido de contaje" se adapta el sentido de la captura del recorrido al sentido del movimiento del eje. Para ello, tenga en cuenta todos los sentidos de giro de los elementos de transmisión (como p. ej., acoplamientos y engranajes).

- normal = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real ascendentes
- invertida = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real descendentes

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
59.0	CNT_DIR	BOOL	FALSE	**Sentido de contaje** Valores permitidos: • 0 = normal • 1 = invertida

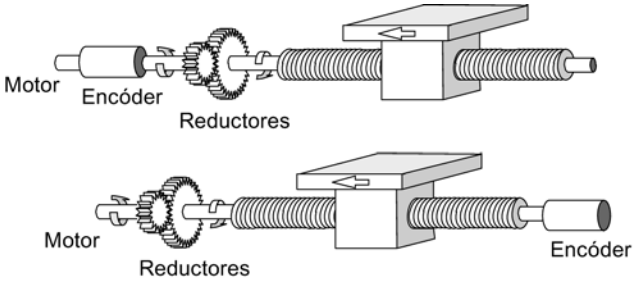
Definir resolución

Definir recorrido por vuelta del encóder

Recorrido por vuelta del encóder

Con el dato máquina "Recorrido por vuelta del encóder" se indica al módulo de posicionamiento qué recorrido realiza el sistema de accionamiento en cada vuelta.

El valor "Recorrido por vuelta del encóder" depende de la estructura del eje y del modo en que está colocado el encóder. Debe tener en cuenta todos los elementos de transmisión, como acoplamientos o engranajes. La figura siguiente muestra dos disposiciones de encóder distintas a modo de ejemplo.



La relación entre los datos máquina "Recorrido por vuelta del encóder" e "Incrementos por vuelta del encóder" se describe en el apartado Resolución (Página 10963).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
24.0	DISP_REV	DINT	L#80000	**Recorrido por vuelta del encóder** Rango de valores admisible: • 1 µm a 1.000.000.000 µm

Definir incrementos por vuelta del encóder

Incrementos por vuelta

El dato máquina "Incremento por vuelta del encóder" indica la cantidad de incrementos que proporciona un encóder en cada vuelta. A partir de este valor y del dato máquina "Recorrido por vuelta del encóder", el módulo de posicionamiento calcula la resolución (Página 10963).

Encóder incremental

Como entrada es posible indicar cualquier valor dentro del rango de valores. El módulo evalúa los incrementos en múltiplos de 4.

Encóder absoluto

Para los límites debe diferenciarse entre los distintos modelos de encóder. Como entrada se permiten sólo valores en potencias de 2.

- Encóder monovuelta (número de vueltas = 1) con longitud del telegrama de 13 bits:
  - valor mínimo = 4
  - valor máximo = 8192
- Encóder multivuelta (número de vueltas > 1) con longitud del telegrama de 25 bits:
  - valor mínimo = 4
  - valor máximo = 8192
- Encóder monovuelta (número de vueltas = 1) con longitud del telegrama de 25 bits:
  - valor mínimo = 4
  - valor máximo = 2^{25}

Las medidas lineales se parametrizan como encóder multivuelta tal como se indica a continuación:

- Incrementos por vuelta del encóder = 8192
- Número de vueltas x 8192 ≥ Número de pasos de la medida lineal

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
32.0	INC_REV	DINT	L#500	**Incrementos por vuelta del encóder** Rango de valores admisible: • 1 a 2^{25}

Resolución

Resolución

La resolución indica el desplazamiento que corresponde s un impulso. Es una medida de la exactitud del posicionamiento y determina también la zona de desplazamiento máximo posible del módulo de posicionamiento.

Para calcular la resolución (RES) se procede tal como se indica en la tabla siguiente:

	Encóder incremental	Encóder absoluto
Valores de entrada	• Recorrido por vuelta del encóder • Incrementos por vuelta del encóder: – Evaluación cuádruple – 1 Incremento = 4 Impulsos	• Recorrido por vuelta del encóder • Incrementos por vuelta del encóder: – 1 Incremento = 1 Impulso
Cálculo	$RES = \frac{\frac{\text{Trayecto}}{\text{Vuelta del encóder}}}{\frac{\text{Impulsos}}{\text{Vuelta del encóder}}}$	

Nota

Todos los datos de posición se redondean al múltiplo entero de la resolución. De este modo es posible diferenciar entre los valores introducidos y los utilizados.

Rango de valores de la resolución

El sistema de medida seleccionado determina el margen de valores de la resolución.

Sistema de medición	Datos en ...	Rango de valores de la resolución
mm	10^{-3} mm	$0{,}1 \times 10^{-3}$ … 1000×10^{-3} mm/impulso
pulgadas	10^{-4} pulgadas	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} pulgada/impulso
grados	10^{-4} grados 10^{-3} grados 10^{-2} grados	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} grados/impulso $0{,}1 \times 10^{-3}$ … 1000×10^{-3} grados/impulso $0{,}1 \times 10^{-2}$ … 1000×10^{-2} grados/impulso

Ejemplo

- Un encóder incremental tiene los datos siguientes:
  - Incrementos por vuelta del encóder: 5000
  - Recorrido por vuelta del encóder: 1.000 mm
  - 1 Incremento = 4 Impulsos

  De ello resulta la resolución (evaluación cuádruple):

$$\text{Resolución} = \frac{1000\ \text{mm}}{5.000\ \text{incrementos}} = 0{,}2000\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{impulsos}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impulso}}$$

- Un encóder SSI tiene los datos siguientes:
  - Incrementos por vuelta del encóder: 4096
  - Recorrido por vuelta del encóder: 1.000 mm
  - 1 Incremento = 1 Impulso

  De ello resulta la resolución:

$$\text{Resolución} = \frac{1000\ \text{mm}}{4096\ \text{incrementos}} = 0{,}2441\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2441\ \frac{\text{mm}}{\text{Impulso}}$$

Dependencia entre la zona de desplazamiento y la resolución

La zona de desplazamiento está limitada por la representación numérica en el módulo de posicionamiento. Esta representación numérica varía en función de la resolución. Por ello, al definir valores hay que procurar mantenerse siempre dentro de los límites admitidos.

La zona de desplazamiento máxima se muestra en la tabla siguiente:

Resolución (RES) dentro del margen	Zona de desplazamiento máxima
0,1 µm/impulso ≤ RES < 1 µm/impulso	-10^8 µm a 10^8 µm (-100 m a +100 m)
1 µm/impulso ≤ RES ≤ 1000 µm/impulso	-10^9 µm a 10^9 µm (-1000 m a +1000 m)

Seleccionar datos para el encóder absoluto (SSI)

Número de vueltas del encóder

Número de vueltas del encóder

El dato máquina "Número de vueltas del encóder" sólo existe en encóders absolutos. Con él se indica el número de vueltas posibles con este encóder.

El **total de pasos del encóder** no es un dato máquina. Se calcula del modo siguiente:

Total de pasos = Incremento por vuelta x Número de vueltas

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.0	NO_REV	DINT	L#1	**Número de vueltas del encóder** Rango de valores admisible: • 1 (encóder monovuelta) • 2 a 4096 en potencias de 2 (encóder multivuelta)

Velocidad de transferencia

Velocidad de transferencia

Con el dato máquina "Velocidad de transferencia" se define la velocidad de la transferencia de datos del encóder SSI al módulo de posicionamiento.

La velocidad de transferencia máxima depende de la velocidad de transferencia:

- 200 m → 188 kHz
- 100 m → 375 kHz
- 40 m → 750 kHz
- 12 m → 1.500 kHz

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
40.0	BAUDRATE	DINT	L#0	**Velocidad de transferencia** Valores permitidos: • 0 = 188 kHz • 1 = 375 kHz • 2 = 750 kHz • 3 = 1.500 kHz

Longitud de telegrama

Longitud de telegrama

Con la "Longitud de telegrama" se define el marco de impulsos que emite el módulo de posicionamiento.

Se permite una longitud de telegrama de 13 bits a 25 bits.

Datos utilizados en el DB de parámetros

La longitud de telegrama seleccionada aquí se dispone en el DB de parámetro con el dato ENC_TYPE (Página 10956).

Paridad

Paridad

Con la "Paridad" se define si para el encóder absoluto no debe transferirse ninguna paridad o si debe transferirse una paridad par o impar.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
28.0	PARITY	DINT	L#0	**Paridad** Valores permitidos: • 0 = ninguno • 1 = paridad par (even) • 2 = paridad impar (odd)

Seleccionar vigilancia

Vigilancia

Rotura de hilo

Con la activación de la vigilancia, el módulo de posicionamiento vigila todos los conductos en un encóder incremental de 5 V y un encóder absoluto. La vigilancia detecta:

- Rotura de hilo
- Cortocircuito de los distintos cables
- Distancia entre flancos de los impulsos de contaje (también en un encóder incremental de 24 V)

Para la monitorización en un encóder incremental de 24 V es preciso ajustar un tiempo de monitorización MON_TIME (Página 10946) > 0.

En encóders incrementales de 5 V sin marca cero es preciso desactivar la monitorización de rotura de hilo o bien interconectar las señales N y /N de forma externa.

Error de telegrama

El módulo monitoriza el telegrama de un encóder absoluto (SSI) para:

- ver si hay errores del bit de inicio y del bit de parada

La monitorización de errores de telegrama en los encóders absolutos (SSI) no puede desactivarse.

Impulsos erróneos (encóder incremental)

Entre dos marcas cero consecutivas, un encóder incremental siempre tiene que proporcionar el mismo número de incrementos.

El módulo de posicionamiento comprueba si la marca cero de un encóder incremental aparece en el estado adecuado del encóder.

En el caso de encóders sin marca cero es preciso desactivar la monitorización de los impulsos de error.

Igualmente es preciso desactivar la monitorización de rotura de hilo o bien interconectar las entradas de marca cero N y /N de forma externa.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
				Vigilancias:
63.0	MON_WIRE	BOOL	TRUE	1 = Rotura de hilo
63.1	MON_FRAME	BOOL	TRUE	1 = Error de telegrama (tiene que ser siempre 1)
63.2	MON_PULSE	BOOL	TRUE	1 = Impulsos de error

Incrementos del canal n

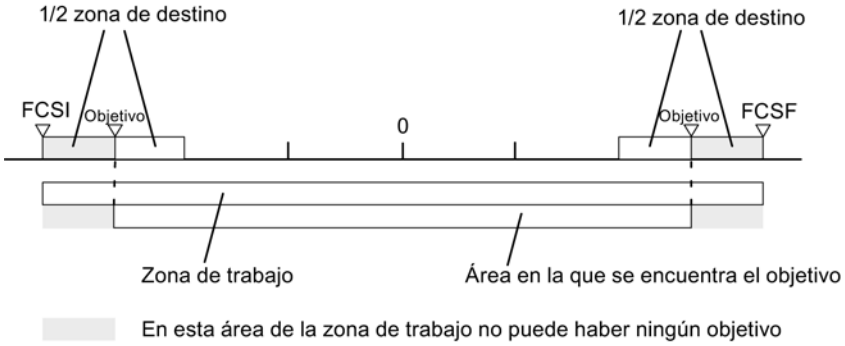
Definir incrementos

Definición

Los incrementos son destinos especificados que pueden ser controlados por el módulo de posicionamiento en el modo de operación "Modo incremental relativo" o "Modo incremental absoluto".

Área posible para los incrementos

El destino al que tiene que realizarse una aproximación tiene que encontrarse al menos a media zona de destino delante del final de carrera por software correspondiente. La figura siguiente muestra los límites de una especificación del incremento.



Número de incremento 1 a 100

Número de incremento 1 a 100

Tiene la posibilidad de insertar un máximo de 100 incrementos en una tabla, que son válidas tanto el modo de operación "Modo incremental relativo" como para el "Modo incremental absoluto".

Tenga en cuenta que el módulo de posicionamiento para el "Modo incremental relativo" no soporta valores negativos. Los valores son interpretados por el módulo de posicionamiento en función del sentido de movimiento como diferencia positiva o negativa.

Nota

La entrada se realiza en la unidad del sistema de medida ajustado. Tenga en cuenta en este caso sobre todo las posiciones decimales.

Ejemplo de números:

- Incremento: 800 mm
- Sistema de medida: 10^{-3} mm
- Entrada en el DB de parámetros: 800000

Recomendación: En la tabla de incrementos defina áreas separadas para incrementos relativos y absolutos.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (incremento 1 a 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (incremento 1 a 100)
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 1 (incremento 1 a 50)
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 2 (incremento 1 a 100)

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios	
120.0	TRGL1.TRG[1]	DINT	L#0	Número de incremento 1	Tabla de incrementos 1
. . .	. . .	. .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Número de incremento 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Número de incremento 51	Tabla de incrementos 2

Dirección	Nombre	Tipo	valor inicial	Comentarios	
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Número de incremento 100	

Número de incremento 254

Número de incremento 254

Dependiendo de la tabla de incrementos es posible utilizar el número de incremento 254 como otra especificación del recorrido. Para las distancias de cambio de velocidad y de desconexión, las entradas del DB de parámetros tienen validez para este incremento.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.2	TRG252_254_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 254
96.0	TRG252_254	DINT	L#0	Incremento para número de incremento 254

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CHGDIF_P	DINT	L#5000	Distancia de cambio de velocidad positiva
104.0	CHGDIF_M	DINT	L#5000	Distancia de cambio de velocidad negativa
108.0	CUTDIF_P	DINT	L#2000	Distancia de desconexión positiva
112.0	CUTDIF_M	DINT	L#2000	Distancia de desconexión negativa

Número de incremento 255

Número de incremento 255

Con el número de incremento 255 dispone de otra especificación del recorrido.

Las distancias de desconexión y las distancias de cambio de velocidad se transfieren junto con el incremento. Al contrario de lo que sucede en otros incrementos, el incremento 255 utiliza los valores fijados en el DB del canal para la distancia de desconexión y de la distancia de cambio de velocidad. Las entradas de los datos máquina no tienen validez para este incremento.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.3	TRG255_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 255
100.0	TRG255	DINT	L#0	Incremento para número de incremento 255
104.0	CHGDIF255	DINT	L#0	Distancias de cambio de velocidad para número de incremento 255
108.0	CUTDIF255	DINT	L#0	Distancias de desconexión para número de incremento 255

Interfaz de control y realimentación del FM 351

Aplicación

En aplicaciones especiales o en un nivel de alarma puede que se necesite un acceso especialmente rápido a señales de realimentación y de control. A estos datos se accede directamente a través de las áreas de entrada y salida del módulo.

Para coordinar el arranque cada vez que arranca el módulo (p. ej. tras insertar el módulo o después de una transición STOP-RUN de la CPU), la instrucción ABS_CTRL (Página 7420) debe llamarse hasta que se muestre el final del arranque a través de RETVAL = 0. A continuación la instrucción ABS_CTRL ya no puede utilizarse.

Nota

No es posible utilizar la instrucción ABS_CTRL en combinación con un acceso en escritura.

Leer señales de realimentación mediante acceso directo

Las direcciones de byte se indican de forma relativa respecto a la dirección de inicio de las salidas del canal correspondiente. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7428).

Dirección inicial del canal 1 = Dirección inicial del módulo

Dirección inicial del canal 2 = Dirección inicial del módulo + 8

En AWL se accede a los datos con los comandos PEB (leer 1 byte), PEW (leer 2 bytes) y PED (leer 4 bytes).

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del FM 351.

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	PARA	interno	interno	DATA_ERR	OT_ERR	DIAG	interno	interno
Byte 1	CHGOVER	CUTOFF	ZSPEED	SPEED_OUT	0	WAIT_EI	WORKING	ST_ENBLD
Byte 2	MODE_OUT							
Byte 3	POS_RCD	0	0	0	GO_P	GO_M	0	SYNC

Dirección	Número de bit
Byte 4	ACT_POS
Byte 5	
Byte 6	
Byte 7	

Ejemplo: Valor real de posición ACT_POS

La dirección inicial del módulo es 512.

```
AWL                 Descripción
L PED 516           //Leer valor real de posición actual (ACT_POS) del canal 1:
                    //con acceso directo:
                    //Dirección inicial del canal + 4
```

Escribir las señales de control mediante acceso directo

Las direcciones de byte se indican de forma relativa respecto a la dirección inicial de las entradas del canal correspondiente. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7428).

Dirección inicial del canal 1 = Dirección inicial del módulo

Dirección inicial del canal 2 = Dirección inicial del módulo + 8

En AWL se accede a los datos con los comandos PAB (escribir 1 byte), PAW (escribir 2 bytes) y PAD (escribir 4 bytes).

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del FM 351.

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	OT_ERR_A	0	0	0
Byte 1	DRV_EN	0	0	0	DIR_P	DIR_M	STOP	START
Byte 2	MODE_IN							
Byte 3	MODE_TYPE							
Byte 4	Reservado							
Byte 5								
Byte 6								
Byte 7								

Ejemplo: Señales de inicio del canal 2

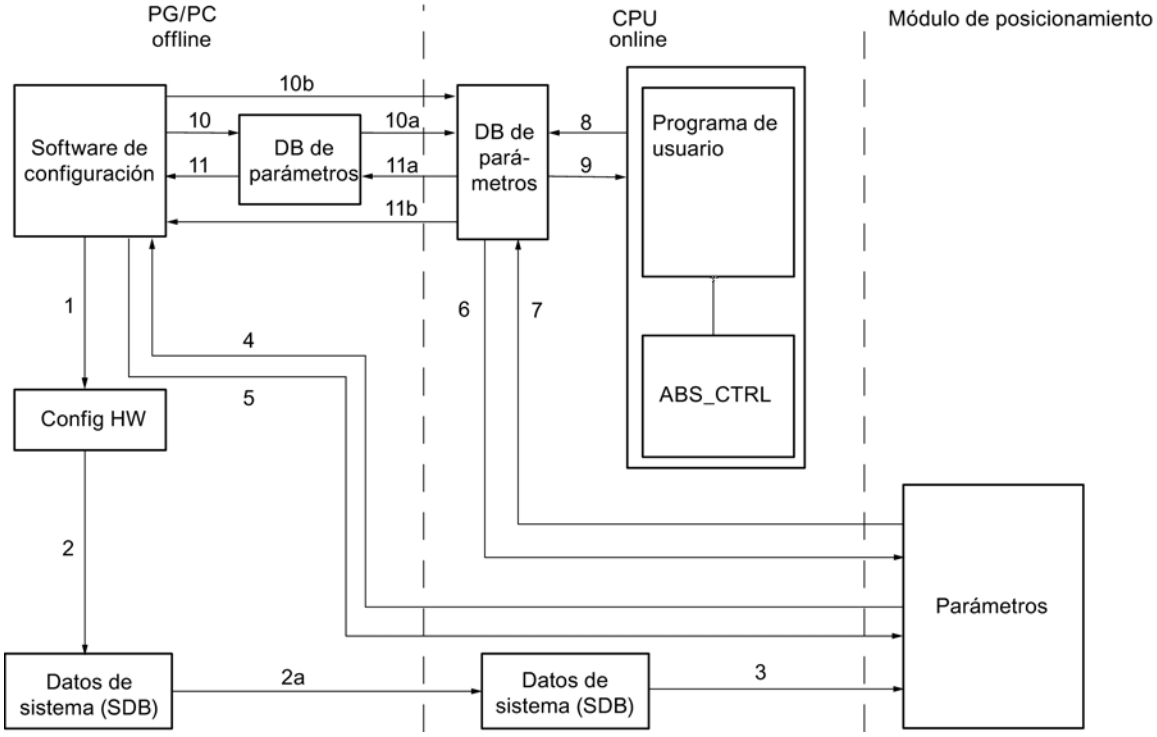
La dirección inicial del módulo es 512.

AWL	Descripción
L 2#10001000	//Poner DRV_EN y DIR_P a 1
T PAB 521	//Escribir señales con acceso directo para canal 2:
	//Dirección inicial del módulo + 8 + 1

Vías de transferencia de parámetros

Se entiende por parámetros los datos máquina y los incrementos que se indican a continuación.

La figura siguiente muestra las vías de transferencia de parámetros al FM x51.



1 Guardar los parámetros en el software de configuración.

2 Guardar y compilar la configuración hardware.

2a Cargar configuración hardware en la CPU. La CPU ejecuta automáticamente el paso 3.

3 La CPU escribe los parámetros en el módulo en parametrizaciones de sistema.

4 Cargar los parámetros de un canal del módulo en la programadora con el comando "Cargar sistema de destino en programadora".

5 Cargar parámetros del software de configuración con el comando "Cargar sistema de destino" en un canal del módulo.

6 Escribir parámetros en un canal del módulo con peticiones del programa de usuario.

7 Leer parámetros desde un canal del módulo a través de peticiones del programa de usuario.

8 Almacenar parámetros del programa de usuario en el DB Online.

9 Leer parámetros del DB Online en el programa de usuario.

10 Exportar parámetros del software de configuración al DB Offline.

10a Cargar DB offline en la CPU.

10b Exportar parámetros del software de configuración al DB Online.

11 Importar parámetros del DB Offline en el software de configuración.

11a Cargar parámetros del DB Online en la programadora.

11b Importar parámetros del DB Online en el software de configuración.

Casos de aplicación de la transferencia de parámetros

Caso de aplicación	Pasos
Los parámetros se editan con el software de configuración. A continuación, los canales del módulo sólo pueden parametrizarse de forma automática en el momento del arranque.	Ejecute los pasos 1, 2 y 2a.
Los parámetros se modifican durante la puesta en servicio en el modo de prueba en el software de configuración.	Ejecute los pasos 4 y 5.
A continuación, los parámetros modificados durante la puesta en servicio se tienen que cargar automáticamente durante el arranque.	Ejecute los pasos 1, 2 y 2a.
Los parámetros se crean con el software de configuración. Los canales del módulo sólo pueden parametrizarse con el programa de usuario a través de bloques de datos.	Ejecute los pasos 10, 10a y 6 y o 10b y 6.
Desea crear datos de reserva para recetas de forma cómoda.	Ejecute los pasos 10 y 10a.
Los parámetros se crean con el software de configuración. Éstos tienen que estar a disposición del programa de usuario para realizar modificaciones provisionales.	Ejecute los pasos 1, 2 y 2a para la parametrización automática.
	Ejecute los pasos 10, 10a, 9, 8 y 6 para el acceso por medio del programa de usuario.
Los parámetros existentes se modifican exclusivamente con el programa de usuario.	Ejecute los pasos 7, 9, 8 y 6.
Desea ver los datos modificados por el programa de usuario con el software de configuración.	Ejecute los pasos 11a y 11 o sólo 11b.
Los parámetros modificados por el programa de usuario se tienen que cargar automáticamente también en el arranque.	Ejecute los pasos 11b u 11a, 11 y, después, 1, 2 y 2a.

Puesta en servicio del FM 351

Uso del FM 351 desde el programa de usuario

Inclusión del FM 351 en un programa de usuario

Para integrar el FM 351 en un programa de usuario pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

Las instrucciones ABS_INIT y ABS_CTRL se activan en el programa de usuario de la CPU. Éstas realizan la comunicación entre la CPU y el FM 351. El DB de canal (Página 7428) se crea automáticamente como DB de instancia en la instrucción ABS_CTRL, si se integra la instrucción ABS_CTRL en el proyecto.

Todos los datos de parámetros pertenecientes a un canal del módulo (datos máquina, tablas de incrementos) se encuentran en el DB de parámetros (Página 7434) del módulo. Inserte un bloque de datos del tipo ABS_PARATYPE en el proyecto mediante "Bloques de programa > Agregar bloque".

Además, para el FM 351 se puede utilizar también la instrucción ABS_DIAG, que permite transferir informaciones de diagnóstico al DB de diagnóstico. El DB de diagnóstico

(Página 7436) se crea automáticamente como DB de instancia en la instrucción ABS_DIAG, si se integra la instrucción ABS_DIAG en el proyecto.

Puesta en servicio del módulo

Incluso sin programa de usuario es posible realizar un primer test del módulo con el cuadro de diálogo "Puesta en servicio". Tenga en cuenta particularmente la advertencia siguiente.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

Para evitar daños personales y materiales observe los puntos siguientes:

- Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Sólo así puede garantizarse que en caso de un fallo del software o del equipo sea posible desconectar la instalación de modo seguro.
- Instale finales de carrera por hardware que actúen directamente en las etapas de potencia de todos los accionamientos.
- Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.
- Si controla y vigila paralelamente el FM 351 desde el programa y desde el cuadro de diálogo "Puesta en servicio" pueden producirse conflictos cuyos efectos no están definidos. Por lo tanto, conmute siempre la CPU al estado STOP cuando trabaje con el cuadro de diálogo "Puesta en servicio" o bien desactive su programa.

Consulte también

ABS_INIT (Página 7419)

ABS_CTRL (Página 7420)

ABS_DIAG (Página 7422)

FM 351: Diagnóstico online

Información actual sobre el módulo

Diagnóstico estándar

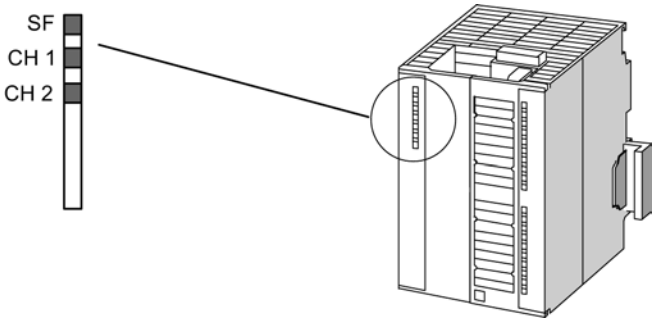
Indicación de errores mediante los LEDs de error del FM 351

¿Dónde se indica el error?

Si el LED rojo de error general se enciende, significa que se ha producido un error o bien en el módulo (error interno), o bien en las conexiones de cable (error externo).

Los LEDs rojos CH 1 y CH 2 indican un error de canal.

En la figura siguiente se muestra la posición de los LEDs de error en el FM 351.



¿Qué errores se indican?

La tabla siguiente contiene los tipos de error que se indican mediante los LEDs de error.

LED	Significado	Explicación
SF	Fallo agrupado	Este LED indica un estado de error del FM 351. Alarma de diagnóstico (error, de canal, interno o externo) Para eliminar el error, ver Clases de error (Página 10984).
CH 1 CH 2	Error de canal 1 Error de canal 2	Estos LEDs indican un error en el canal 1 o en el canal 2. • Rotura de hilo en el encóder • Error del encóder absoluto • Errores en señal del encóder incremental • Fallo operativo • Error de parametrización

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

Procesamiento de las alarmas

El módulo de posicionamiento puede activar alarmas de diagnóstico. Estas alarmas se procesan en un OB de alarma. Cuando se dispara una alarma sin que se encuentre cargado el OB correspondiente, la CPU pasa a STOP.

Para habilitar las alarmas de diagnóstico se procede del modo siguiente:

1. Seleccione el módulo en HWCN.
2. Habilite la alarma de diagnóstico en Edición > Propiedades del objeto > Parámetros básicos
3. Guardar y compilar la configuración.
4. Cargue la configuración en la CPU

Visión de conjunto de las alarmas de diagnóstico

Los eventos y errores siguientes disparan una alarma de diagnóstico:

- Fallo operativo
- Datos máquina erróneos
- Error de diagnóstico

Estos errores se describen con detalle en el apartado "Clases de error (Página 10984)".

Reacción del módulo de posicionamiento en caso de error con alarma de diagnóstico

- El posicionamiento se cancela.
- Con las alarmas de diagnóstico siguientes se borra la sincronización:
  - Falta conector frontal, falta tensión auxiliar externa para la alimentación del encóder
  - se ha detectado un error de señal cero, error de línea (señales de encóder de 5 V)
  - se ha rebasado ^la zona de desplazamiento (se notifica con un error de operador)
  - Ajustar valor real no se puede ejecutar (se notifica con un error de operador)
- Las señales de control START, DIR_P y DIR_M dejan de procesarse excepto en un caso.
  Excepción:
  Si se produce un error de operador sigue siendo posible un modo Jog en el sentido de la zona de trabajo.
- El selector de función y las peticiones se continúan procesando.

El módulo de posicionamiento detecta un fallo ("entrante")

Una alarma de diagnóstica es "entrante" cuando por lo menos hay un error en cola de espera. Si no se han solventado todos los errores los demás que queden en cola de espera, se notificarán de nuevo como "entrantes".

Secuencia:

1. El módulo de posicionamiento detecta uno o varios errores y dispara una alarma de diagnóstico. El LED "SF" y, en función del error, los LEDs "CH1" / "CH2" se encienden. El error se registra en el búfer de diagnóstico.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Es posible evaluar la información de inicio del OB 82.
4. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
5. Encontrará más información si llama la instrucción ABS_DIAG (Página 7422).

El módulo de posicionamiento detecta el paso al estado libre de fallos ("saliente")

Una alarma sólo es "saliente" cuando se ha eliminado el último error del módulo.

Secuencia:

1. El módulo de posicionamiento detecta que todos los fallos han sido eliminados y dispara una alarma de diagnóstico. El LED "SF" está apagado. El búfer de diagnóstico no se modifica.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
4. Evalúe el bit OB82_MDL_DEFECT.
   Si este bit es "0", no hay errores en el módulo. Su evaluación puede finalizar aquí.

Alarmas de diagnóstico según el estado de la CPU

- En estado STOP de la CPU las alarmas de diagnóstico están bloqueadas por el módulo de posicionamiento.
- Si con la CPU en estado STOP no se eliminan todos los errores pendientes, el módulo vuelve a notificar como "entrantes" los errores todavía no eliminados tras la transición al estado RUN.
- Si se eliminan todos los errores pendientes con la CPU en estado STOP, el estado libre de errores del módulo **no** se notificará con una alarma de diagnóstico tras la transición al estado RUN.

Evaluación de errores en el programa de usuario

Reacción de errores en el programa de usuario

En el programa de usuario es posible reaccionar de forma precisa a los errores que aparecen. Para ello se dispone de los siguientes recursos:

- Los valores de retorno RETVAL del FB estándar integrado:
  Este valor se vuelve a calcular cada vez que se llama el bloque.
  RETVAL = -1 es un indicador de grupo para un error isócrono en una petición o en la comunicación con el módulo.
- A cada petición le corresponde un bit de error _ERR como indicador de grupo de un error de la petición o de uno de sus predecesores en la cadena de peticiones:
  El bit de errores se activa para una petición de escritura y sus peticiones de continuación si el módulo notifica un error de datos o si se produce un error de comunicación.
  En el caso de peticiones de lectura el bit de error se establece para la petición afectada cuando se ha producido un error de comunicación.
  La instrucción ABS_CTRL (Página 7420) activa de nuevo los bits de error después del procesamiento de una petición. No obstante, estos deben ser retomados por el programa de usuario para una evaluación de errores.
- La señal de realimentación DATA_ERR como indicador de grupo de un error que el módulo ha detectado durante una petición de escritura. La señal se vuelve a calcular en la siguiente petición de escritura.
- La señal de realimentación OT_ERR (error de operador) como indicador de grupo de un error que el módulo ha detectado al iniciar un recorrido. El bit de error tiene que ser acusado mediante el OT_ERR_A = 1 una vez solucionada la causa.

- La señal de realimentación DIAG se activa cuando el contenido del búfer de diagnóstico se ha modificado. Esta señal puede aparecer más tarde que las señales DATA_ERR y OT_ERR.
- El error de comunicación JOB_ERR contiene el código de error (Página 10983) si existe un problema de comunicación entre la instrucción y el módulo. El valor se recalcula después de procesar una petición y se crea en el DB del canal con la instrucción ABS_CTRL y en el DB de diagnóstico con la instrucción ABS_DIAG.
- La instrucción ABS_DIAG (Página 7422) para leer el búfer de diagnóstico del módulo. Aquí es posible averiguar las causas de error para eventos isócronos y asíncronos.
- Las alarmas de diagnóstico para la respuesta rápida a los eventos en el OB de la alarma de diagnóstico (por ejemplo, OB 82).

Evaluación de una alarma de diagnóstico en el programa de usuario

Las siguientes entradas en los datos locales del OB de la alarma de diagnóstico (por ejemplo, OB 82) con activadas por el módulo de posicionamiento. Los errores se introducen en el búfer de diagnóstico (clase de error 128 (Página 10991)).

Dirección	Nombre	Tipo	Comentarios
0.0	OB82_EV_CLASS	BYTE	Clase de evento e indentificadores: B#16#38: evento saliente B#16#39: evento entrante
1.0	OB82_FLT_ID	BYTE	Código de error (B#16#42)
2.0	OB82_PRIORITY	BYTE	Clase de prioridad: B#16#1A en estado operativo RUN B#16#1C en estado operativo ARRANQUE
3.0	OB82_OB_NUMBR	BYTE	Número de OB (82)
4.0	OB82_RESERVED_1	BYTE	Reservado
5.0	OB82_IO_FLAG	BYTE	Módulo de entrada: B#16#54
6.0	OB82_MDL_ADDR	INT	Dirección básica lógica de la tarjeta en la que se ha producido el error.
8.0	OB82_MDL_DEFECT	BOOL	Fallo del módulo
8.1	OB82_INT_FAULT	BOOL	Error interno
8.2	OB82_EXT_FAULT	BOOL	Error externo
8.3	OB82_PNT_INFO	BOOL	Error de canal existente
8.4	OB82_EXT_VOLTAGE	BOOL	Falta tensión auxiliar externa
...	no utilizado		
10.3	OB82_WTCH_DOG_FLT	BOOL	Activación de la vigilancia de tiempo
...	no utilizado		
12.0	OB82_DATE_TIME	DATE_AND_TIME	Fecha y hora en la que se ha solicitado el OB

Búfer de diagnóstico del módulo

Eventos de diagnóstico

El búfer de diagnóstico del módulo contiene como máximo 9 eventos de diagnóstico y está organizado como búfer anular.

Un evento de diagnóstico se escribe en el búfer cuando se detecta un aviso (de error) entrante. Esto puede ser un aviso, un error isócrono (error de datos, error de operador) o incluso un error asíncrono (error de servicio y error de diagnóstico). De una causa de un error pueden surgir también varias entradas como errores consiguientes. Los avisos salientes no generan entradas en el búfer de diagnóstico.

Para cada evento de diagnóstico se indican:

- Estado (siempre entrante)
- Error interno
- Error externo
- Clase de error
- Número de error
- Número de canal
- Número de incremento (en errores de la tabla de incrementos)

Cuando se escribe un evento de diagnóstico en el búfer de diagnóstico, la señal de realimentación DIAG = 1 se activa en todos los canales parametrizados.

El búfer de diagnóstico puede transferirse como un todo con la instrucción ABS_DIAG (Página 7422) a un bloque de datos (DB de diagnóstico) o a través de un cuadro de diálogo de evaluación de errores del software de configuración. Si se lee el búfer de diagnóstico, el módulo activa la señal de realimentación DIAG = 0.

Nota

Si el búfer de diagnóstico se lee al mismo tiempo con la instrucción ABS_DIAG y el cuadro de diálogo de evaluación de errores, puede ocurrir que uno de los eventos de diagnóstico que se ha producido no sea detectado por el programa.

Datos y estructura del DB de diagnóstico

Datos y estructura del DB de diagnóstico

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

La tabla siguiente muestra la estructura del DB de diagnóstico.

Dirección	Nombre	Tipo	valor inicial	Comentarios
0.0	MOD_ADDR	INT	0	Dirección del módulo
256.0	JOB_ERR	INT	0	Número de error del error de comunicación
258.0	JOBBUSY	BOOL	FALSE	1 = Petición activa
258.1	DIAGRD_EN	BOOL	FALSE	1 = Leer el búfer de diagnóstico necesariamente
260.0	DIAG_CNT	INT	0	Número de entradas válidas de la lista
262.0	DIAG[1]	STRUCT		Entrada más reciente de datos de diagnóstico
272.0	DIAG[2]	STRUCT		Segunda entrada de datos de diagnóstico
282.0	DIAG[3]	STRUCT		Tercera entrada de datos de diagnóstico
292.0	DIAG[4]	STRUCT		Cuarta entrada de datos de diagnóstico
302.0	DIAG[5]	STRUCT		Quinta entrada de datos de diagnóstico
312.0	DIAG[6]	STRUCT		Sexta entrada de datos de diagnóstico
322.0	DIAG[7]	STRUCT		Séptima entrada de datos de diagnóstico
332.0	DIAG[8]	STRUCT		Octava entrada de datos de diagnóstico
342.0	DIAG[9]	STRUCT		Novena entrada de datos de diagnóstico

La tabla siguiente muestra la estructura de una entrada de diagnóstico DIAG[n].

Dirección	Nombre	Tipo	valor inicial	Comentarios
+0.0	STATE	BOOL	FALSE	0 = Evento saliente 1 = Evento entrante
+0.1	INTF	BOOL	FALSE	1 = Error interno
+0.2	EXTF	BOOL	FALSE	1 = Error externo
+2.0	FCL	INT	0	Clase de error: 1: Fallo operativo 2: Error de operador 4: Error de datos 5: Error de datos máquina 6: Error de las tablas de incrementos 15: Avisos 128: Error de diagnóstico
+4.0	FNO	INT	0	Número de error
+6.0	CH_NO	INT	0	Número de canal
+8.0	TRG_NO	INT	0	Número de incremento

Lista de los avisos JOB_ERR

Avisos JOB_ERR

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A0	32928	-32608	Acuse negativo al leer del módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2 [1]	32930	-32606	Error de protocolo para capa 2
80A3 [1]	32931	-32605	Error de protocolo en User-Interface / User
80A4 [1]	32932	-32604	Fallo en la comunicación del bus K
80B0	32944	-32592	Registro de datos/Petición desconocida.
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo previsto.
80C0 [1]	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1 [1]	32961	-32575	Los datos de una petición de escritura semejante todavía no se han procesado en el módulo.
80C2 [1]	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3 [1]	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados de momento.
80C4 [1]	32964	-32572	Error de comunicación
80C5 [1]	32965	-32571	La periferia descentralizada no está disponible.
80C6 [1]	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (petición de escritura).
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (petición de escritura).
853A	34106	-31430	DB de parámetros no presente. (petición de escritura).
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras aparecer un error (petición de escritura).
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (petición de lectura).
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (petición de lectura).
873A	34618	-30918	DB de parámetros no presente. (petición de lectura).
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras aparecer un error (petición de lectura).

[1] Los errores 80A2 a 80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario.

Clases de errores

Fallo operativo

Clase de error 1: Fallo operativo

Los fallos operativos se detectan de forma asíncrona respecto a un control. Los fallos operativos hacen que se cancele el posicionamiento a excepción del número de error 9. El número de error 9 hace que se desactive el posicionamiento.

La tabla siguiente muestra los fallos operativos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	Rebase del final de carrera por software Inicio		Sí
	Causa	El valor real se encuentra fuera de la zona de trabajo.	
2	Rebase del final de carrera por software Fin		Sí
	Causa	El valor real se encuentra fuera de la zona de de trabajo.	
3	Rebase del inicio de la zona de desplazamiento		Sí
	Causa	Rebasar el límite de la zona de desplazamiento (las coordenadas de los límites de la zona de desplazamiento pertenecen a la zona de desplazamiento).	
4	Rebase del fin de la zona de desplazamiento		Sí
	Causa	Rebasar el límite de la zona de desplazamiento (las coordenadas de los límites de la zona de desplazamiento pertenecen a la zona de desplazamiento).	
5	Error en la aproximación al destino		Sí
	Causa	La zona de destino no se ha alcanzado dentro del tiempo de vigilancia.	
6	Abandonar zona de parada		Sí
	Causa	El valor real se encuentra fuera de la zona de parada.	
7	Acoplamiento conjunto		Sí
	Causa	Modificación del valor real > ½ zona de parada en el sentido erróneo.	
8	Falta modificación del valor real o es demasiado reducida		Sí
	Causa	No hay modificaciones del valor real o modificación del valor real en contra del sentido nominal dentro del tiempo de vigilancia.	
9	Rebasar destino (FM 451)		Sí
	Causa	En "Ajustar valor real a vuelo" se ha rebasado el destino.	
10	Rebasar zona de destino		Sí
	Causa	La zona de destino se ha rebasado después de la aproximación al destino.	
11	Punto de cambio de velocidad conectado de forma errónea		Sí
	Causa	El eje oscila en el punto de cambio de velocidad.	
12	Punto de desconexión conectado de forma errónea		Sí
	Causa	El eje oscila en el punto de desconexión/punto de inversión	
13	Inicio de la zona de destino conectado de forma errónea		Sí
	Causa	El eje oscila en la zona de destino.	
14	Modificación mayor que la mitad de la zona del eje rotativo		Sí
	Causa	La velocidad/frecuencia es muy alta o existen saltos erróneos del valor real.	

Nº	Significado		Alarma de diagnóstico
15	Modificación mayor que la zona del eje rotativo		Sí
	Causa	La velocidad/frecuencia es muy alta o existen saltos erróneos del valor real.	
16	Incremento para número de incremento 252 no transferido (FM 451)		Sí
	Causa	No se ha transferido el incremento.	
17	Incremento para número de incremento 252 no aproximable (FM 451)		Sí
	Causa	La distancia entre la posición real actual y el incremento especificado es inferior a la distancia de cambio de velocidad o la distancia de desconexión.	
18	Incremento erróneo para el número de incremento 252 (FM 451)		Sí
	Causa	El incremento se encuentra fuera de la zona de trabajo.	

Error de operador

Clase de error 2: Error de operador

Los errores de operador se detectan al cambiar las señales de control en el área de datos de usuario. Los fallos de operador producen una desactivación del posicionamiento.

La tabla siguiente muestra los errores de operador posibles y sus causas.

N.º	Significado		Alarma de diagnóstico
1	Modo de operación no admisible		no
	Causa	El modo de operación seleccionado no está permitido.	
3	Petición de interfaz no permitida		no
	Causa	La señal seleccionada no está permitida en este modo de operación.	
4	Parámetro erróneo del modo de operación		no
	Causa	• En el modo de operación "Jog" la velocidad especificada es distinta de la velocidad rápida o la velocidad lenta. • En el modo de operación "Modo incremental" el incremento es distinto de 1 a 100 o distinto de 252, 254 y 255.	
5	Habilitación de inicio no existente		no
	Causa	Habilitación al iniciar no existente	
7	El destino/La zona de destino se encuentra fuera de la zona de trabajo.		no
	Causa	El destino especificado o calculado se encuentra fuera de los finales de carrera por software.	
8	Eje no parametrizado		no
	Causa	No se han parametrizado datos máquina para el eje o se han parametrizado datos erróneos.	
9	El eje no está sincronizado.		no
	Causa	El modo de operación "Modo incremental" sólo es posible con un eje ya sincronizado.	
10	Destino/Parte del trayecto no posicionable		no
	Causa	La distancia entre la posición real actual y el destino especificado es inferior a la distancia de desconexión.	

N.º	Significado		Alarma de diagnóstico
17	**La búsqueda del punto de referencia no es posible**		no
	Causa	Se ha conectado un encóder SSI.	
18	**El modo incremental relativo o absoluto no es posible**		no
	Causa	El incremento no es válido.	
19	**La distancia de desconexión no es mayor que ½ de la zona de destino en el número de incremento 255**		no
	Causa	La distancia de desconexión para el incremento 255 es inferior a la mitad de la zona de destino.	
20	**Desplazamiento no permitido en el sentido especificado**		no
	Causa	La distancia del final de carrera por software es insuficiente.	

Error de datos

Clase de error 4: Error de datos

Los errores de datos se detectan de forma síncrona respecto a un control/mando. Los datos de errores no producen ninguna respuesta al error.

La tabla siguiente muestra los errores de datos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
6	**Incremento especificado demasiado grande**		no
	Causa	El valor está fuera de ±100 m ó ±1000 m. El recorrido / destino no puede ser superior a la zona de desplazamiento. En un eje rotativo la coordenada debe ser ≥ 0 e inferior al final del eje rotativo.	
10	**Decalaje de señal cero erróneo (FM 451)**		no
	Causa	El decalaje de señal cero es mayor que ± 100 m o ± 1.000 m. Los finales de carrera por software, tras ajustar el decalaje de señal cero, se encuentran fuera de la zona de desplazamiento (100 m...+100 m o -1.000 m... +1.000 m). Eje rotativo: El valor del decalaje de señal cero es mayor que el final del eje rotativo.	
11	**Indicación del valor real errónea**		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
12	**Punto de referencia erróneo**		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
20	**Activar datos máquina no admisible**		no
	Causa	En el módulo no hay datos máquina nuevos sin errores.	
27	**Ajuste codificado en bits no permitido**		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	

Nº	Significado		Alarma de diagnóstico
29	Codificación de bits no permitida		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
34	No es posible cancelar la activación del valor real		no
	Causa	Con un encóder SSI y un eje lineal, tras la ejecución del ajuste el valor real de posición se encontraría fuera de la zona de trabajo.	
36	Distancia de cambio de velocidad errónea en número de incremento 255		no
	Causa	El valor se encuentra fuera del margen numérico permitido ±100 m o ±1000 m. En un eje rotativo la coordenada debe ser ≥ 0 e inferior al final del eje rotativo.	
37	Distancia de desconexión errónea en número de incremento 255		no
	Causa	El valor se encuentra fuera del margen numérico permitido ±100 m o ±1000 m. La distancia de desconexión es menor que la distancia de cambio de velocidad.	
107	Eje no parametrizado		no
	Causa	En el eje no hay ningún dato máquina o los datos no están activados.	
108	El eje no está sincronizado.		no
	Causa	Uno de las peticiones "Ajustar valor real" y "Cancelar preselección del valor real" se ha activado a pesar de que el eje no está sincronizado.	

Error de datos máquina

Clase de error 5: Error de datos máquina

La alarma de diagnóstico sólo se disparará con un bloque de datos del sistema (SDB) con errores. Los errores de datos máquina no producen ninguna respuesta al error.

La tabla siguiente muestra errores de datos máquina posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
5	Error en el ajuste de la alarma de proceso		Sí
	Causa	Ha intentado seleccionar una alarma de proceso que el módulo no soporta.	
6	Distancia mínima entre bordes errónea (FM 451)		Sí
	Causa	Ha indicado como distancia mínima entre bordes un valor < 0 ó > 10^9 µm.	
7	Sistema de medida erróneo		Sí
	Causa	El valor del sistema de medida se encuentra fuera del rango admisible de 1 a 4 y 6.	
8	Tipo de eje erróneo		Sí
	Causa	Como tipo de eje no ha indicado ni 0 ni 1.	
9	Final erróneo del eje rotativo		Sí
	Causa	El valor del final del eje rotativo se encuentra fuera del rango admitido de 1 a 10^9 µm o respectivamente de 1 a 10^8 µm (según sea la resolución).	
10	Tipo de encóder erróneo		Sí
	Causa	El valor del tipo de encóder se encuentra fuera del rango admisible de 1 a 4.	
11	Recorrido erróneo por vuelta de encóder		Sí

Nº	Significado		Alarma de diagnóstico
	Causa	El valor del recorrido/vuelta del encóder se encuentra fuera del rango admisible de 1 a 10^9 µm (independientemente de la resolución).	
13	**Número erróneo de incrementos por vuelta del encóder**(véase el apartado "Incrementos por vuelta del encóder (Página 10958)")		Sí
14	**Número erróneo de vueltas** (véase el apartado ""Número de vueltas de encóder (Página 10961)")		Sí
15	**Velocidad de transferencia errónea**		Sí
	Causa	Para la velocidad de transferencia se ha indicado un valor fuera del rango admisible de 0 a 3.	
16	**Coordenada del punto de referencia errónea**		Sí
	Causa	La coordenada se encuentra fuera del margen de -100 m a +100 m o, respectivamente, de -1000 m a +1000 m (según la resolución). Eje lineal: La coordenada se encuentra fuera de la zona de trabajo. Eje rotativo: La coordenada es mayor que el final del eje rotativo o < 0.	
17	**Calibración errónea del encóder absoluto**		Sí
	Causa	Encóder de recorrido SSI: El valor del reajuste del encóder absoluto no se encuentra dentro del rango del encóder (incrementos por vuelta del encóder × cantidad de vueltas - 1).	
18	**Tipo erróneo de búsqueda del punto de referencia**		Sí
	Causa	Se ha indicado un valor por fuera de los valores admisibles 0, 1, 2 y 3.	
19	**Sentido de contaje erróneo**		Sí
	Causa	Se ha indicado un valor por fuera de los valores admisibles 0 y 1.	
20	**Vigilancia de hardware no posible**		Sí
	Causa	Se ha puesto en "FALSE" la monitorización de errores de telegrama en el DB de parámetros. En el encóder utilizado, la vigilancia de impulsos erróneos no es posible. Desactive el parámetro MON_PULSE.	
21	**Inicio erróneo del final de carrera por software**		Sí
	Causa	Eje lineal: El inicio del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m o 1000 m...+1000 m, dependiendo de la resolución). Eje lineal: El inicio del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es menor que -100 m o -1000 m (según la resolución).	
22	**Fin erróneo del final de carrera por software**		Sí
	Causa	Eje lineal: El fin del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó 1000 m...+1000 m, dependiendo de la resolución) o es menor que el inicio del final de carrera por software. Eje lineal: El fin del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es mayor que +100 m ó +1000 m (según la resolución).	
23	**Velocidad máxima errónea**		Sí
	Causa	Los datos no mostrados en el DB de parámetros tienen que ser 0.	
24	**Zona de destino errónea**		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m ó 1000 m, según la resolución. Eje rotativo: Zona superior al final del eje rotativo.	
25	**Tiempo de vigilancia erróneo**		Sí

Nº	Significado		Alarma de diagnóstico
	Causa	El valor del tiempo de vigilancia se encuentra fuera del rango admisible de 0 a 100 000 ms.	
26	**Zona de parada errónea**		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m o 1000 m, según la resolución. Eje rotativo: Zona superior al final del eje rotativo.	
127	**Velocidad de parada errónea**		Sí
	Causa	El valor de la velocidad de parada se encuentra por fuera del rango admisible de 0 a 100 000 µm/min.	
128	**Modo de control erróneo**		Sí
	Causa	Para el modo de control se ha indicado un valor que está fuera del rango admisible de 1 a 4.	
129	**Velocidad de inicio errónea para la búsqueda del punto de referencia**		Sí
	Causa	Como velocidad de inicio no ha indicado ni 0 ni 1.	
130	**Distancia de cambio de velocidad errónea en el sentido +**		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m o 1000 m (según la resolución). Eje rotativo: Zona superior al final del eje rotativo e inferior a la mitad de la zona de destino.	
131	**Distancia de cambio de velocidad errónea en el sentido -**		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m ó 1000 m (según la resolución). Eje rotativo: Zona superior al final del eje rotativo e inferior a la mitad de la zona de destino.	
132	**Distancia de desconexión errónea en el sentido +**		Sí
	Causa	La distancia de desconexión es mayor que la distancia de cambio de velocidad positiva, inferior a la mitad de la zona de destino, o se encuentra fuera del margen admisible entre 0 y 100 m o 1000 m (según resolución).	
133	**Distancia de desconexión errónea en el sentido -**		Sí
	Causa	La distancia de desconexión es mayor que la distancia de cambio de velocidad negativa, inferior a la mitad de la zona de destino, o se encuentra fuera del margen admisible entre 0 y 100 m o 1000 m (según resolución).	
200	**Resolución errónea**		Sí
	Causa	Ha indicado una resolución < 0,1 µm/impulso o >1000 µm/impulso. Ha indicado un recorrido/vuelta y una cantidad de impulsos/vuelta que dan una resolución < 0,1 ó > 1000.	
201	**El encóder no se adapta a la zona de trabajo / área del eje rotativo**		Sí
	Causa	Encóder SSI y eje rotativo: El encóder no cubre exactamente el área del eje rotativo. Eje lineal: El encóder no cubre ni siquiera la zona de trabajo (finales de carrera por software incluidos).	

Error de las tablas de incrementos

Clase de error 6: Error de las tablas de incrementos

Los errores de tablas de incrementos no producen ninguna respuesta al error.

La tabla siguiente muestra los posibles errores de tablas de incrementos y sus causas.

Nº	Significado		Alarma de diagnóstico
6	Incremento especificado demasiado grande en la tabla de incrementos		no
	Causa	El valor está fuera de ±100 m ó ±1000 m. El recorrido / destino no puede ser superior a la zona de desplazamiento. En un eje rotativo, la coordenada debe ser ≥ 0 e inferior al final del eje rotativo.	

Avisos

Clase de error 15: Avisos

Los avisos no producen ninguna respuesta al error.

La tabla siguiente muestra los avisos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	Inicio de parametrización		no
	Causa	El módulo ha detectado una parametrización mediante un bloque de datos del sistema.	
2	Final de la parametrización		no
	Causa	El módulo ha procesado sin problemas la parametrización mediante un bloque de datos del sistema.	
11	Distancia insuficiente del punto de cambio de velocidad		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	
12	Distancia insuficiente del punto de inversión		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	
14	Distancia insuficiente del punto de desconexión		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	
15	Distancia insuficiente del principio de la zona de destino		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	

Error de diagnóstico

Clase de error 128: Error de diagnóstico

La tabla siguiente muestra los errores de diagnóstico posibles y sus causas, así como los efectos y las soluciones posibles.

Nº	Significado		Alarma de diagnóstico
4	**Falta tensión auxiliar externa**		Sí
	Causa	• La tensión auxiliar externa de 24 V no se ha conectado o ha fallado • El fusible del módulo está defectuoso. • Tensión insuficiente • Rotura de hilo de masa • Cortocircuito (p. ej. en el encóder conectado)	
	Efecto	• El posicionamiento se cancela en todos los canales • Desactivar las salidas • Eliminación de la sincronización en encóders incrementales cuando falta la tensión auxiliar para la alimentación del encóder • El módulo de posicionamiento no está parametrizado • Borrado de la habilitación de inicio	
	Solución	Comprobar que la conexión de 24 V sea correcta (si la conexión de 24 V es correcta, el módulo está averiado).	
5	**Falta conector frontal (FM 451)**		Sí
	Causa	El conector frontal del módulo de posicionamiento no está insertado	
	Efecto	• Falta tensión auxiliar externa de 24 V • El módulo no está listo para funcionar	
	Solución	Enchufar el conector frontal en el módulo de posicionamiento	
51	**La vigilancia de tiempo (watchdog) ha reaccionado**		Sí
	Causa	• Interferencias fuertes en el módulo de posicionamiento • Error en el módulo de posicionamiento	
	Efecto	• El módulo se resetea • Desactivar todas las salidas • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, este vuelve a estar listo para funcionar. • El módulo notifica el transcurso de la vigilancia del tiempo (watchdog) con "entrante" y "saliente"	
	Solución	• Eliminación de interferencias • Consulte al departamento comercial correspondiente cuando las circunstancias exactas que han dado lugar al error son de mayor entidad. • Cambio del módulo de posicionamiento	
144	**Rotura de hilo Encóder**		Sí

Nº	Significado		Alarma de diagnóstico
	Causa	• El cable del encóder no está enchufado o está cortado • Encóder sin señales de cuadratura • Asignación de pines incorrecta • Longitud de cable excesiva • Cortocircuito de las señales del encóder • Errores de flancos de las señales de encóder • Máxima frecuencia de entrada de la entrada del encóder sobrepasada • Corte de la alimentación del encóder	
	Efecto	• El posicionamiento se interrumpe • Desactivar las salidas • Borrado de la sincronización, con encóders incrementales • Borrado de la habilitación de inicio	
	Solución	• Comprobar el cable del encóder • Guardar los datos técnicos del encóder • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador. • Guardar las especificaciones técnicas del módulo	
145	**Error de telegrama Encóder absoluto**		Sí
	Causa	El tráfico de telegramas entre el módulo de posicionamiento y el encóder absoluto (SSI) tiene errores o está interrumpido: • El cable del encóder no está enchufado o está cortado • Tipo de encóder erróneo • Encóder mal configurado (encóders programables) • Longitud de telegrama mal indicada • El encóder proporciona valores erróneos (encóder defectuoso) • Interferencia en el cable del sistema de medición • Velocidad de transferencia escogida demasiado alta • Tiempo monoestable del encóder superior a 64 µs	
	Efecto	• El posicionamiento se interrumpe • Desactivar las salidas • Borrado de la habilitación de inicio	
	Solución	• Comprobar el cable del encóder • Comprobar el encóder • Comprobar el tráfico de telegramas entre el encóder y el módulo de posicionamiento	
146	**Pulsos erróneos (encóder incremental)**		Sí
	Causa	• Vigilancia del encóder ha detectado pulsos erróneos • El número de incrementos por vuelta está mal introducido • Encóder defectuoso: no proporciona el número de impulsos indicado • Marca cero errónea o no presente • Interferencias en el cable del encóder	

Nº	Significado		Alarma de diagnóstico
	Efecto	• El posicionamiento se interrumpe • Desactivar las salidas • Borrado de la habilitación de inicio	
	Solución	• Indicar correctamente el número de incrementos/vuelta del encóder (cuadro de diálogo de parametrización) • Comprobar el encóder y su cable • Cumplir las normas de apantallamiento y de conexión a tierra • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador.	

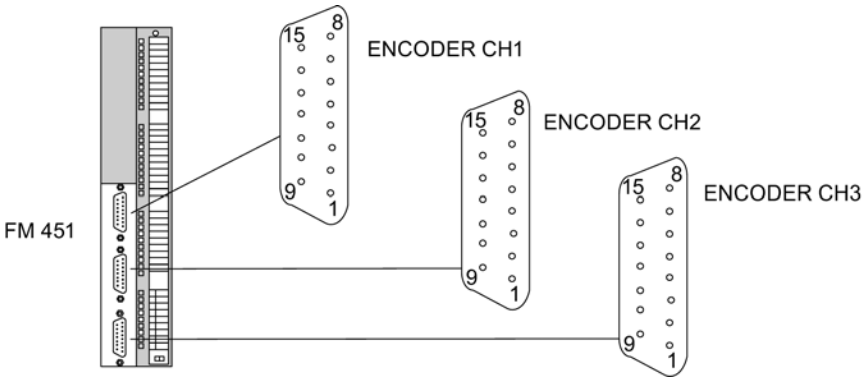
11.4.2.6 Uso del FM 451

Asignación de terminales de los conectores frontales del FM 451

Descripción de la interfaz del encóder

Posición de los conectores hembra sub-D

En la figura siguiente se muestra la posición de montaje y el nombre de los conectores hembra del módulo. En los conectores hembra sub-D se pueden conectar encóders incrementales o encóders absolutos (SSI).



CH = canal

Asignación de las interfaces de encóder (encóder CH1, CH2 y CH3)

Pin	Nombre	Encóder incremental (24 V)	Encóder incremental (5 V)	Encóder absoluto
1	A*	Señal de encóder A	---	---
2	CLS	---	---	Pulso de desplazamiento SSI

Pin	Nombre	Encóder incremental (24 V)	Encóder incremental (5 V)	Encóder absoluto
3	/CLS	---	---	Pulso de desplazamiento SSI inverso
4	B*	Señal de encóder B	---	---
5	DC 24 V	Alimentación del encóder	Alimentación del encóder	Alimentación del encóder
6	DC 5.2 V	---	Alimentación del encóder	Alimentación del encóder
7	M	Tierra del bastidor	Tierra del bastidor	Tierra del bastidor
8	N*	Señal de marca cero	---	---
9	RE	Salida tipo PNP/NPN [1)]	---	---
10	N	---	Señal de marca cero	---
11	/N	---	Señal de marca cero invertida	---
12	/B	---	Señal de encóder B invertida	---
13	B	---	Señal de encóder B	---
14	/A / /DAT	---	Señal de encóder A invertida	Datos SSI invertidos
15	A / DAT	---	Señal de encóder A	Datos SSI

[1)]**Nota:** si se conecta un encóder incremental de otro fabricante en contrafase (salida tipo PNP/NPN), debe tenerse en cuenta lo siguiente:

- Salida tipo PNP: conecte RE (9) con la masa (7).
- Salida tipo NPN: conecte RE (9) con +24 V (5).

Asignación del conector frontal

Conector frontal

En el conector frontal de 48 polos se conectan:

- las entradas y salidas digitales asignadas a los canales;
- las tensiones de alimentación de los encóders y de las salidas digitales.

Asignación de terminales del conector frontal (X1)

La tabla siguiente contiene la asignación de terminales del conector frontal del FM 451.

Terminal	Nombre	Función	Encóder incremental	Encóder absoluto
1		Asignado; contiene un puente de cable para detectar el conector frontal enchufado.		
2				
3	1L+	Tensión auxiliar de DC 24 V para la alimentación del encóder		
de 4 a 7		No utilizado		
8	1I0	Canal 1: entrada digital 0	Sensor del punto de referencia	No utilizado
9	1I1	Canal 1: entrada digital 1	Contacto inversor	No utilizado

Terminal	Nombre	Función	Encóder incremental	Encóder absoluto
10	1I2	Canal 1: entrada digital 2	Entrada de habilitación	
11	1I3	Canal 1: entrada digital 3	Entrada para "Ajustar valor real al vuelo"/"Medición de longitud"/"Detección de bordes"/ "Evento externo para número de incremento 252"	
12		No utilizado		
13	2L+	Tensión auxiliar de DC 24 V para la alimentación de la carga		
14	2L+	Los dos bornes están conectados internamente en el módulo.		
15	2I0	Canal 2: entrada digital 0	Sensor del punto de referencia	No utilizado
16	2I1	Canal 2: entrada digital 1	Contacto inversor	No utilizado
17	2I2	Canal 2: entrada digital 2	Entrada de habilitación	
18	2I3	Canal 2: entrada digital 3	Entrada para "Ajustar valor real al vuelo"/"Medición de longitud"/"Detección de bordes"/ "Evento externo para número de incremento 252"	
19	3I0	Canal 3: entrada digital 0	Sensor del punto de referencia	No utilizado
20	3I1	Canal 3: entrada digital 1	Contacto inversor	No utilizado
21	3I2	Canal 3: entrada digital 2	Entrada de habilitación	
22	3I3	Canal 3: entrada digital 3	Entrada para "Ajustar valor real al vuelo"/"Medición de longitud"/"Detección de bordes"/ "Evento externo para número de incremento 252"	
de 23 a 24		No utilizado		
25	3L+	Tensión auxiliar de DC 24 V para la alimentación de la carga		
26	3L+	Los dos bornes están conectados internamente en el módulo.		
27	1Q0	Canal 1: salida digital 0		
28	1Q1	Canal 1: salida digital 1		
29	1Q2	Canal 1: salida digital 2		
30	1Q3	Canal 1: salida digital 3		
31	2Q0	Canal 2: salida digital 0		
32	2Q1	Canal 2: salida digital 1		
33	2Q2	Canal 2: salida digital 2		
34	2Q3	Canal 2: salida digital 3		
de 35 a 36		No utilizado		
37	4L+	Tensión auxiliar de DC 24 V para la alimentación de la carga		
38	4L+	Los dos bornes están conectados internamente en el módulo.		
39	3Q0	Canal 3: salida digital 0		
40	3Q1	Canal 3: salida digital 1		
41	3Q2	Canal 3: salida digital 2		
42	3Q3	Canal 3: salida digital 3		
de 43 a 47		No utilizado		
48	M	Masa de las tensiones auxiliares 1L+, 2L+, 3L+ y 4L+		

Encontrará más información sobre el diseño y el cableado en las instrucciones de servicio (http://support.automation.siemens.com/WW/view/es/1118070) del módulo de posicionamiento FM 451.

Tensión auxiliar para la alimentación del encóder (1L+, M)

Aquí puede conectar una tensión auxiliar de DC 24 V para la alimentación del encóder. En el FM 451, el potencial de referencia de esta alimentación (M) está conectado a la masa de la alimentación de la carga (M).

Se vigila si la tensión auxiliar de DC 24 V para la alimentación del encóder presenta subtensión y rotura de hilo.

En la interfaz del encóder (conectores hembra sub-D encóder CH1, CH2 y CH3) se proporcionan DC 24 V y DC 5,2 V para los diferentes tipos de encóder.

⚠ PRECAUCIÓN
Preste atención a la correcta polaridad de la tensión auxiliar de DC 24 V para la alimentación del encóder (1L+, M). Si conecta la tensión auxiliar de DC 24 V para la alimentación del encóder con los polos invertidos, el módulo sufrirá daños y deberá sustituirse.

Tensión auxiliar para la alimentación de la carga (2L+, 3L+, 4L+, M)

En los bornes 2L+, 3L+, 4L+ y M debe conectar las tensiones auxiliares de DC 24 V para la alimentación de la carga de las salidas digitales.

⚠ PRECAUCIÓN
Preste atención a la correcta polaridad de la tensión auxiliar de DC 24 V para la alimentación de la carga (2L+, 3L+, 4L+, M). Si conecta la tensión auxiliar de DC 24 V para la alimentación de la carga con los polos invertidos, el módulo sufrirá daños y deberá sustituirse.

Indicación sobre el cableado para DC 24 V

Al efectuar el cableado, tenga en cuenta que todos los bornes del 1L+ al 4L+ deben estar interconectados para que el módulo funcione sin fallos.

Comience con el cableado de DC 24 V al borne 38 y conecte la tensión auxiliar del borne 37 al borne 26. Proceda de forma análoga con los demás bornes.

Como alternativa también puede conectar alimentaciones separadas a los terminales 1L+ (encóder), 2L+ (canal 1), 3L+ (canal 2) y 4L+ (canal 3). No obstante, asegúrese de que todas las alimentaciones tengan un potencial de masa común. En este caso, después de haberse producido una pérdida de tensión se mantiene la sincronización de los ejes gracias a las tensiones auxiliares para la alimentación de la carga.

Alimentaciones de carga

La alimentación de la carga DC debe satisfacer los siguientes requisitos:

Como alimentación de la carga sólo debe utilizarse una pequeña tensión DC ≤ 60 V separada con seguridad de la red. La separación segura puede estar efectuada según los requisitos que figuran, entre otros, en:

- VDE 0100 Parte 410/HD 384-4-41/IEC 364-4-41
  (como pequeña tensión funcional con separación segura),
- VDE 0805/EN 60950/IEC 950
  (como pequeña tensión de seguridad SELV) o bien
- VDE 0106 Parte 101.

12 entradas digitales (de 1I0 a 3I3)

El FM 451 dispone de 4 entradas digitales por canal.

En las 12 entradas digitales es posible conectar conectores libres de rebotes (de 24 V tipo P) o sensores sin contacto (detectores de proximidad de 2 ó 3 hilos).

No se vigila si las entradas digitales presentan cortocircuitos o roturas de hilo; dichas entradas carecen de aislamiento galvánico respecto a la masa de la alimentación del encóder.

El estado de cada entrada se indica con el LED correspondiente.

12 salidas digitales (de 1Q0 a 3Q3)

El FM 451 dispone de 4 salidas digitales por canal.

Con las salidas digitales se controla la etapa de potencia. El funcionamiento de las salidas digitales depende del modo de control (Página 11042).

No se vigila si las salidas digitales presentan cortocircuitos o roturas de hilo; dichas salidas carecen de aislamiento galvánico respecto a la masa de la alimentación del encóder.

El estado de cada salida se indica en el LED correspondiente.

Funciones de FM 451

Funciones

Al preseleccionar un modo de operación, una petición o un selector de función, se selecciona la funcionalidad con la que se utilizará el módulo. La siguiente tabla ofrece una visión de conjunto de la funcionalidad del módulo.

Funcionalidad	Descripción
Escribir y leer los datos máquina (Página 10998)	Con los datos máquina el módulo de posicionamiento se adapta al eje y al encóder.
Escribir y leer tablas de incrementos (Página 11001)	Con los incrementos se especifican destinos que pueden ser controlados por el módulo de posicionamiento en el modo de operación "Modo incremental relativo/absoluto".

Funcionalidad	Descripción
Final de una operación de posicionamiento (Página 11003)	El final de una operación de posicionamiento se logra de tres formas distintas: • Aproximación al destino • Desactivación • Cancelar
Modo de operación "Jog" (Página 11009)	En el modo de operación "Jog", el accionamiento se mueve en un sentido al pulsar una tecla.
Modo de operación "Búsqueda del punto de referencia" (Página 11012)	El modo de operación "Búsqueda del punto de referencia" permite sincronizar el eje en función de un evento externo repetitivo.
Modo de operación "Modo incremental" (Página 11018)	En el modo de operación "Modo incremental" el módulo de posicionamiento puede • mover el accionamiento a destinos absolutos, • mover el accionamiento de forma relativa alrededor de una parte del trayecto en un sentido predeterminado.
Petición "Definir valor real" / "Ajustar valor real al vuelo" / "Cancelar preselección del valor real" (Página 11025)	Con la petición "Ajustar valor real" o "Ajustar valor real al vuelo" se asigna una nueva coordenada al estado actual del encóder. Con la petición "Cancelar valor real ajustado" se restablecen todos los offsets de la zona de trabajo que han sido generados con "Ajustar valor real".
Petición "Decalaje de señal cero" (Página 11027)	Con la petición "Decalaje de señal cero" se desplaza en el valor introducido la señal cero del sistema de coordenadas.
Petición "Definir punto de referencia" (Página 11029)	Con la petición "Definir punto de referencia" se sincroniza el eje.
Selector de función "Medición de longitud" / "Captura de bordes" (Página 11031)	Con las funciones "Medición de longitud" y "Captura de bordes" es posible calcular la longitud de una pieza de trabajo.
Selector de función "Aproximación en bucle" (Página 11034)	Con la "Aproximación en bucle" define el sentido en el que se avanza hacia el punto de destino con cierre de fuerza.
Selector de función "Entrada de habilitación" (Página 11037)	La entrada de habilitación es una entrada externa con la que puede habilitarse un posicionamiento a través de un evento externo.
Petición "Leer datos de posición" (Página 11038)	Con la petición "Leer datos de posición" es posible leer el incremento, la distancia residual y la velocidad en el instante actual.
Petición "Leer datos del encóder" (Página 11038)	Con la petición "Leer datos de encóder" se leen los datos actuales del encóder, así como el valor del reajuste del encóder absoluto.
Señales de respuesta del posicionamiento (Página 11039)	Con la función "Señales de respuesta para el posicionamiento" se informa del estado actual del posicionamiento.
Señales de realimentación del diagnóstico (Página 11039)	Con las "Señales de respuesta del diagnóstico" se recibe información sobre los eventos de diagnóstico que se han producido.

Escribir y leer los datos máquina

Escribir, activar y leer los datos máquina

Con los datos máquina se adapta el módulo de posicionamiento al eje y al encóder.

El programa de usuario permite modificar y leer los parámetros durante el funcionamiento.

Todos los parámetros se encuentran en el DB de parámetros:

- Los datos máquina se encuentran en el DB de parámetros, en las direcciones que van del 4.0 al 116.0.

El número del DB de parámetros se tiene que registrar en el DB de canal respectivo.

Los parámetros se pueden introducir con el editor DB, o bien cómodamente en los cuadros de diálogo "Accionamiento", "Eje" y "Encóder" y escribirse en el DB de parámetros con la función "Exportar".

Los parámetros de un DB de parámetros existente pueden importarse a los cuadros de diálogo con la función "Importar".

Primera parametrización

Si el canal todavía no contiene ningún dato máquina, proceda de la siguiente manera para una primera parametrización sin el cuadro de diálogo de parametrización:

1. Introduzca los nuevos valores en el DB de parámetros y almacénelos.
2. Cargue el DB de parámetros en la CPU.
3. Active el bit de inicio siguiente en el DB de canal para la petición:
   - Escribir datos máquina (MDWR_EN = 1)
4. Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) en el programa de usuario cíclico.

Modificar datos máquina

Para modificar datos máquina desde el programa de usuario, proceda del siguiente modo:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Active los bits de inicio en el DB de canal para las peticiones:
   - Escribir datos máquina (MDWR_EN = 1)
   - Activar datos máquina (MD_EN = 1)
3. Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) en el programa de usuario cíclico.

Si define los bits de inicio de estas peticiones al mismo tiempo, la instrucción ABS_CTRL (o ABS_CTRL_451) se encargará de que las peticiones se procesen en la secuencia correcta.

De lo contrario, cambie siempre los datos máquina en la siguiente secuencia:

- Escribir datos máquina
- Activar datos máquina

Leer datos máquina

Para leer los datos máquina actuales desde un canal, proceda del modo siguiente:

1. Active el bit de inicio siguiente en el DB de canal:
   – Leer datos máquina (MDRD_EN = 1)
2. Llame las instrucciones ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) en el programa de usuario cíclico.

De este modo se depositan en la CPU los datos máquina del DB de parámetros.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.0	MDWR_EN	BOOL	FALSE	1 = Escribir datos máquina
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina
36.5	MDRD_EN	BOOL	FALSE	1 = Leer datos máquina

Nota

Si se modifican parámetros relevantes para la sincronización, al activar los datos máquina el módulo realiza las siguientes acciones para el canal afectado:

- La sincronización se borra.
- El sensor de funciones y el decalaje de señal cero se desactivan.
- Los datos máquina y las tablas de incrementos existentes hasta ahora quedan invalidados.

Son parámetros relevantes para la sincronización:

- Tipo de eje
- Final del eje rotativo
- Tipo de señal
- Recorrido por vuelta del encóder
- Incrementos por vuelta del encóder
- Número de vueltas
- Coordenada del punto de referencia
- Reajuste del encóder absoluto
- Tipo de búsqueda del punto de referencia
- Sentido de contaje

Escribir y leer tablas de incrementos

Escribir y leer tablas de incrementos

Con los incrementos se especifican destinos que pueden ser controlados por el módulo de posicionamiento en el modo de operación "Modo incremental relativo/absoluto".

El programa de usuario permite modificar y leer los parámetros durante el funcionamiento.

Todos los parámetros se depositan en el DB de parámetros:

- Las tablas de incrementos se encuentran en el DB de parámetros en las direcciones 120,0 a 516,0.

El número del DB de parámetros se tiene que registrar en el DB de canal correspondiente en cada caso.

Los parámetros pueden introducirse con el editor DB, o bien introducirse cómodamente en el cuadro de diálogo "Medidas incrementales" y se escriben con la función "Exportar" en el DB de parámetros.

Los parámetros de un DB de parámetros existente pueden importarse al cuadro de diálogo con la función "Importar".

Primera parametrización

Si el canal todavía no contiene ninguna tabla de incrementos, proceda del siguiente modo para una primera parametrización sin el cuadro de diálogo de parametrización:

1. Introduzca los nuevos valores en el DB de parámetros y almacénelos.
2. Cargue el DB de parámetros en la CPU.
3. Active los bits de inicio en el DB de canal para las peticiones:
   – Escribir tabla de incrementos 1 (TRGL1WR_EN = 1) y/o escribir tabla de incrementos 2 (TRGL2WR_EN = 1)
4. Llame las instrucciones ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) en el programa de usuario cíclico.

Modificar tablas de incrementos

Para modificar las tablas de incrementos mediante el programa de usuario, proceda del siguiente modo:

1. Introduzca los nuevos valores en el DB de parámetros.
2. Active los bits de inicio en el DB de canal para las peticiones:
   – Escribir tabla de incrementos 1 (TRGL1WR_EN = 1) y/o escribir tabla de incrementos 2 (TRGL2WR_EN = 1)
3. Llame las instrucciones ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) en el programa de usuario cíclico.

Leer tablas de incrementos

Para leer las tablas de incrementos a partir de un canal, proceda del siguiente modo:

1. Active los bits de inicio en el DB de canal para las peticiones:
   – Leer tabla de incrementos 1 (TRGL1RD_EN = 1) y/o leer tabla de incrementos 2 (TRGL2RD_EN = 1)
2. Llame las instrucciones ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) en el programa de usuario cíclico.

De este modo se depositan en la CPU las tablas de incrementos del DB de parámetros.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (1 ... 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (51 ... 100)
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 1 (1 ... 50)
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 2 (51 ... 100)

Nota

Si se modifican parámetros relevantes para la sincronización, al activar los datos máquina el módulo realiza las siguientes acciones para el canal afectado:

- La sincronización se borra.
- El sensor de funciones y el decalaje de señal cero se desactivan.
- Los datos máquina y las tablas de incrementos existentes hasta ahora quedan invalidados.

Son parámetros relevantes para la sincronización:

- Tipo de eje
- Final del eje rotativo
- Tipo de señal
- Recorrido por vuelta del encóder
- Incrementos por vuelta del encóder
- Número de vueltas
- Coordenada del punto de referencia
- Reajuste del encóder absoluto
- Tipo de búsqueda del punto de referencia
- Sentido de contaje

Final de una operación de posicionamiento

Definición

El final de una operación de posicionamiento se logra a través de la señal de realimentación WORKING = 0. Esto se logra de tres formas distintas:

- Aproximación al destino
- Desactivación
- Cancelar

Vigilancias

Al final de una operación de posicionamiento se encuentran activas las siguientes vigilancias:

- Tiempo de vigilancia
  El tiempo de vigilancia se redispara en el punto de desconexión por última vez y pierde su validez al finalizar la operación de posicionamiento.
  En el transcurso de este tiempo es necesario alcanzar el final de la operación de posicionamiento. De lo contrario, las salidas se desconectan y se notifica el fallo operativo "Error en la aproximación al destino" (número de error 5).
- Vigilancia de la zona de destino
  El FM x51 crea un área simétrica alrededor de cada destino y define con ello la precisión de posicionamiento de su aplicación. Dentro de esta área es preciso que el eje se pare durante una aproximación al destino. Una especificación de valor 0 desconecta la tolerancia en la aproximación al destino.
- Vigilancia de la velocidad de parada
  La velocidad de parada permite determinar si el accionamiento se para dentro de la zona de destino. Una vez alcanzado el punto de desconexión, el FM x51 vigila si la velocidad real del accionamiento es inferior a la velocidad de parada.
  La velocidad de parada debe rebasarse por defecto dentro de la zona de trabajo. De lo contrario, el FM x51 notifica el fallo operativo "Rebasar zona de destino" (número de error 10).
  El desbordamiento por defecto de la velocidad de parada sólo se vigila una vez por aproximación al destino.
  Tenga en cuenta en este caso que la velocidad de parada para la determinación de la velocidad del módulo puede rebasarse temporalmente por defecto si el eje se mueve con una velocidad de posicionamiento muy pequeña (menos de 2 impulsos por 8 ms).
- Vigilancia de la zona de parada
  Tras finalizar una operación de posicionamiento se vigila si el accionamiento se detiene en una posición de destino alcanzada o si se aleja de ella.
  La zona de parada se vigila
  - después de que el FM x51 haya notificado la señal de realimentación "PEH",
  - si el tiempo de vigilancia se rebasa por exceso,
  - si la velocidad de parada se rebasa por defecto.

  Si la zona de parada se abandona sin una petición de marcha válida, el FM x51 notifica el fallo operativo "Abandonar zona de parada" (número de error 6).

Aproximación al destino

La aproximación al destino en los modos de operación "Modo incremental absoluto" y "Modo incremental relativo" comienza cuando se alcanza el punto de desconexión. A partir de este punto se desconecta el accionamiento y el FM x51 se hace cargo de las funciones de vigilancia.

En función de las vigilancias que se hayan parametrizado, surgen diferentes casos para la creación de la señal de repuesta "PEH (POS_RCD)". La operación de posicionamiento se cancela cuando se crea la señal "PEH (POS_RCD)".

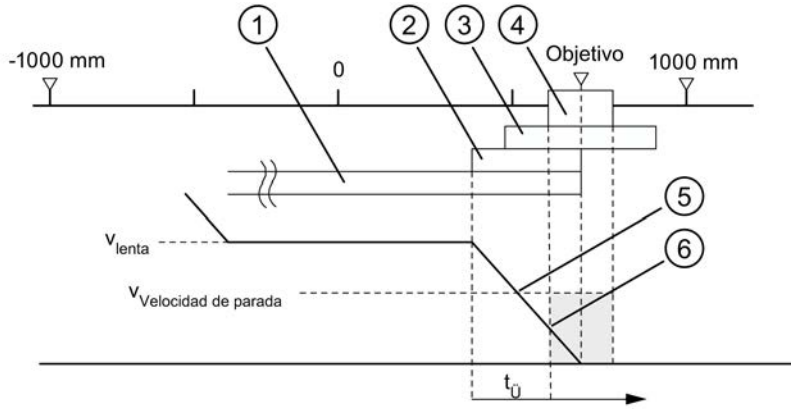
Caso 1: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) > 0
- Velocidad de parada (ZSPEED_L) > 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si la velocidad de parada se rebasa por defecto y se alcanza la zona de destino. Es irrelevante la condición que se cumpla en primer lugar.

No se genera PEH si el valor real no alcanza la zona de destino dentro del tiempo de vigilancia o si la zona de destino se sobrepasa sin rebasar por defecto la velocidad de parada.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino > 0 y velocidad de parada > 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Zona de destino
⑤ Velocidad de parada alcanzada
⑥ Zona de destino con V_{parada} alcanzada: PEH se activa
$t_Ü$ Tiempo de vigilancia

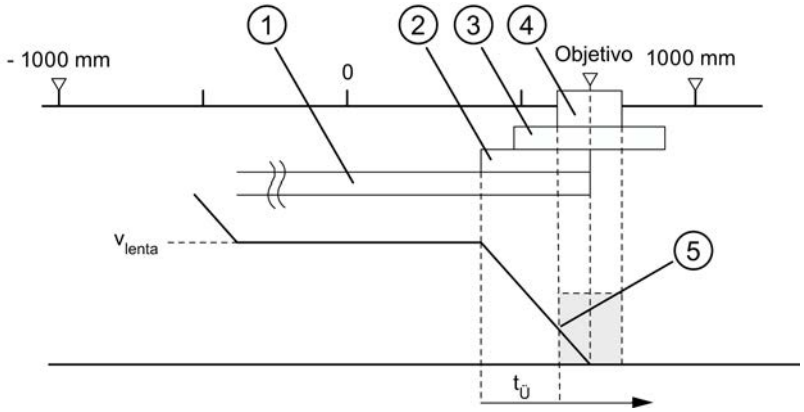
Caso 2: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) > 0
- Velocidad de parada (ZSPEED_L) = 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si se alcanza la zona de destino.

No se genera una PEH si el valor real no alcanza la zona de destino dentro del tiempo de vigilancia.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino > 0 y velocidad de parada = 0.



① Distancia de cambio de velocidad positiva

② Distancia de desconexión positiva

③ Zona de parada

④ Zona de destino

⑤ Zona de destino con V_{parada} alcanzada: PEH se activa

$t_Ü$ Tiempo de vigilancia

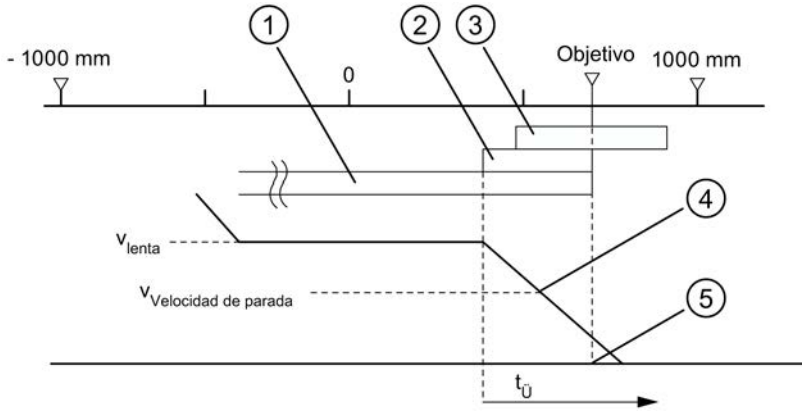
Caso 3: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) = 0
- Velocidad de parada (ZSPEED_L) > 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si la velocidad de parada se rebasa por defecto y se alcanza el destino.

No se genera una PEH si el valor real no alcanza el destino dentro del tiempo de vigilancia o si la zona de destino se sobrepasa sin rebasar por defecto la velocidad de parada.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino = 0 y velocidad de parada > 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Velocidad de parada alcanzada
⑤ Destino alcanzado: PEH se activa
$t_Ü$ Tiempo de vigilancia

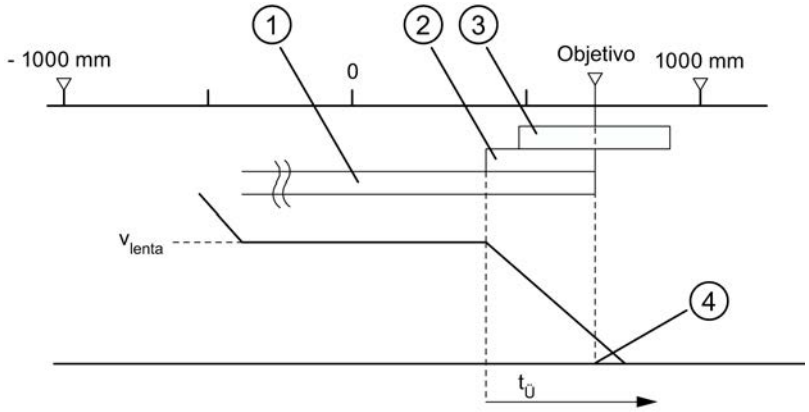
Caso 4: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) = 0
- Velocidad de parada (ZSPEED_L) = 0
- Tiempo de vigilancia (MON_TIME) > 0

Se genera una PEH si se alcanza el destino.

No se genera una PEH si el valor real no alcanza el destino dentro del tiempo de vigilancia.

La siguiente figura muestra la aproximación al destino en modo incremental con zona de destino = 0 y velocidad de parada = 0.



① Distancia de cambio de velocidad positiva
② Distancia de desconexión positiva
③ Zona de parada
④ Destino alcanzado: PEH se activa
$t_Ü$ Tiempo de vigilancia

Caso 5: Se ha parametrizado lo siguiente:

- Zona de destino (TRG_RANGE) ≥ 0
- Velocidad de parada (ZSPEED_L) ≥ 0
- Tiempo de vigilancia (MON_TIME) = 0

Si en este caso el accionamiento se para delante de la zona de destino, el final de la operación de posicionamiento no se detecta. No se genera una PEH y la señal de realimentación WORKING permanece activada. La operación de posicionamiento sólo puede cancelarse eliminando la habilitación del accionamiento (DRV_EN = 0).

Desactivación sin destino predeterminado

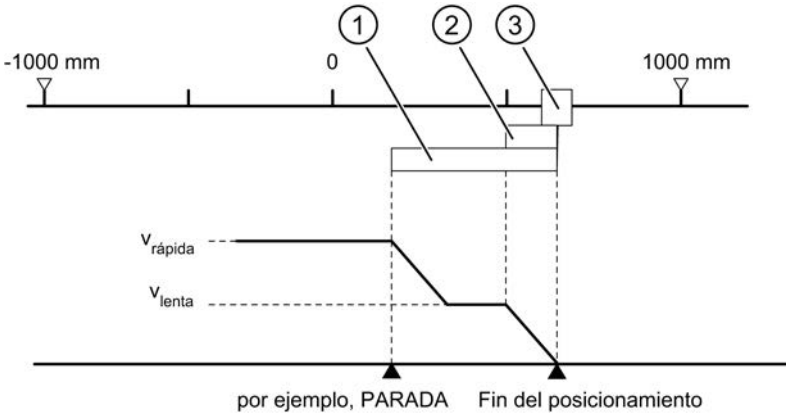
Desactivación significa: El proceso de posicionamiento se cierra de forma precisa manteniendo las diferencias desde la marcha rápida a la marcha lenta.

La operación de posicionamiento se desactiva si:

- el FM x51 recibe una señal de STOP (STOP = 1);
- los modos de operación "Jog" y "Búsqueda del punto de referencia" finalizan;
- aparecen errores de operador o el fallo operativo "Rebasar destino" " (número de error 9).

La señal de realimentación "PEH (POS_RCD)" no se activa. Los procesos son análogos a los de la aproximación al destino.

La siguiente figura muestra la desactivación de una operación de posicionamiento.



① Distancia de cambio de velocidad positiva

② Distancia de desconexión positiva

③ Zona de destino

Cancelar

Cancelar significa: La operación de posicionamiento finaliza inmediatamente sin que se utilicen la distancia de cambio de velocidad ni la distancia de desconexión de la marcha rápida o la marcha lenta después de la parada. Para ello, todas las salidas relevantes del modo de control pertinente se desconectan inmediatamente y, además:

- Incremento = valor real
- Recorrido residual = Cero

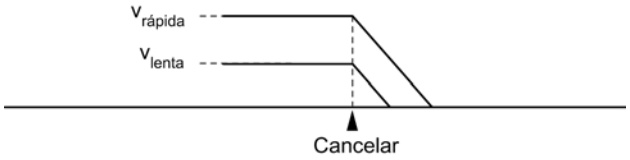
La operación de posicionamiento se cancela si:

- la señal de habilitación del accionamiento se borra (DRV_EN = 0);
- la CPU pasa a STOP;
- aparecen errores de diagnóstico u otros fallos operativos aparte del error "Rebasar destino" " (número de error 9).

La señal de realimentación "PEH (POS_RCD)" no se activa en el modo de operación "Modo incremental".

Cuando la velocidad de parada está parametrizada, la vigilancia de parada se activa en cuanto se rebasa por defecto. Cuando la velocidad de parada está parametrizada, la vigilancia de parada se activa con la desactivación de las salidas.

La siguiente figura muestra la cancelación de una operación de posicionamiento.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
76.0	TRG_RANGE	DINT	L#1000	Zona de destino
80.0	MON_TIME	DINT	L#2000	Tiempo de vigilancia
84.0	ZSPEED_R	DINT	L#1000	Zona de parada
88.0	ZSPEED_L	DINT	L#30000	Velocidad de parada

Señales de respuesta en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
23.1	WORKING	BOOL	FALSE	1 = Operación de posicionamiento en curso
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada

Modo de operación "Jog"

Definición

En el modo de operación "Jog", el accionamiento se mueve en un sentido al pulsar una tecla. Para los dos sentidos (positivo y negativo) es preciso instalar sendos botones. El modo de operación "Jog" se puede utilizar tanto para el eje sincronizado como para el eje no sincronizado.

Si el eje no está sincronizado, el modo "Jog" permite realizar un posicionamiento en el sentido preestablecido.

Si el eje está sincronizado (eje lineal), el modo "Jog" permite realizar un posicionamiento en el final de carrera por software.

Requisitos

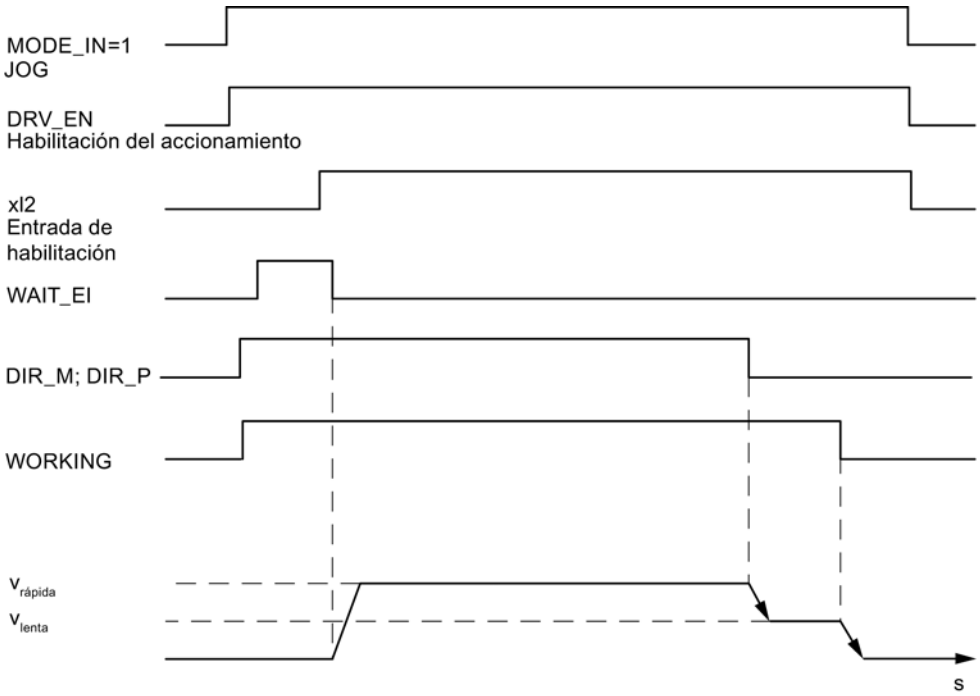
El eje debe estar parametrizado.

Desarrollo del modo de operación "Jog"

1. Active la señal de control para el modo de operación "Jog" (MODE_IN = 1).
2. Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).
3. Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.
4. Introduzca la velocidad de inicio.
   - Marcha rápida (MODE_TYPE = 1)
   - Marcha lenta (MODE_TYPE = 0)

5. Conecte la señal de control para el sentido de marcha positivo o negativo (DIR_P = 1 o DIR_M = 1).
6. Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).

La figura siguiente muestra un ejemplo del modo de operación "Jog".



Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	1 = Jog
17.0	MODE_TYPE	BYTE	B#16#0	1 = Marcha rápida 0 = Marcha lenta
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Operación de posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Desactivación del modo Jog

El modo de operación "Jog" se desactiva cuando

- se vuelve a abrir el botón con el que realiza el modo "Jog" (DIR_M = 0 o DIR_P = 0);
- el FM 451 recibe una señal de STOP (STOP = 1),
- el valor real de un eje lineal sincronizado alcanza el límite de la zona de trabajo. La continuación de la marcha sólo es posible en el sentido contrario.

Tras desactivar la marcha puede continuarse la marcha en el sentido que se desee.

Cancelación del modo Jog

El modo de operación "Jog" se cancela cuando

- la señal de habilitación del accionamiento se borra (DRV_EN = 0),
- se ha rebasado un límite de la zona de desplazamiento en un eje lineal.

Vigilancias

En el modo de operación "Jog" no están activas, al final del posicionamiento, las siguientes vigilancias:

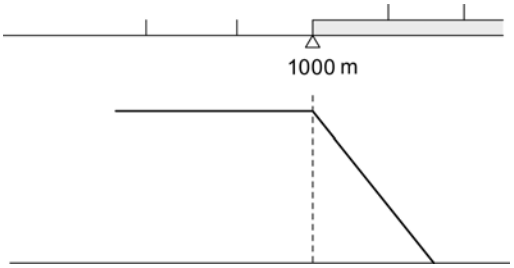
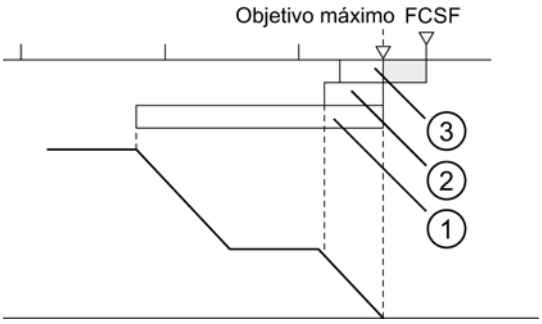
- Vigilancia de la zona de parada
- Vigilancia de la zona de destino
- Vigilancia de la velocidad de parada

El modo "Jog" no produce un posicionamiento correcto en el que al final se activa la señal "Posición alcanzada".

Límite de la zona de trabajo en un eje lineal

Los límites para el modo "Jog" son distintos en función de si el eje está sincronizado o no sincronizado.

La tabla siguiente muestra el modo "Jog" con un eje sincronizado y no sincronizado.

El eje no está sincronizado.	El eje está sincronizado.
Si en el modo "Jog" se rebasa el límite de la zona de desplazamiento. • el indicador del valor real ya no es válido. • se cancela la operación de posicionamiento.  El indicador del valor real ya no es válido	El modo "Jog" representa un posicionamiento en destinos que están separados de los finales de carrera por software en toda el área de destino. Los límites de la zona de trabajo se calculan a partir de • **SEE - ½ zona de destino** para el final del eje lineal en el sentido positivo • **SEE + ½ zona de destino** para el final del eje lineal en el sentido negativo Si no abre antes el botón, el FM 451 se desconecta en un punto que se encuentra la mitad del área de destino delante del final de carrera por software correspondiente. El FM 451 activa todas las áreas que se necesitan para la desactivación correcta alrededor de este punto de destino.  = Área de la zona de trabajo en la que no puede haber ninguna posición de destino ① = Distancia de cambio de velocidad positiva ② = Distancia de desconexión positiva ③ = ½ zona de destino

Modo de operación "Búsqueda del punto de referencia"

Definición

El modo de operación "Búsqueda del punto de referencia" permite sincronizar el eje en función de un evento externo repetitivo.

Requisitos

- Un encóder incremental con señal cero.
- El eje debe estar parametrizado:

Conexiones	Canal 1	Canal 2	Canal 3
Sensor del punto de referencia	Entrada digital 1I0	Entrada digital 2I0	Entrada digital 3I0
	El sensor del punto de referencia debe estar estructurado de tal modo que, el accionamiento, puede cambiar de forma segura de la marcha rápida a la marcha lenta en el área del interruptor.		
Contacto inversor	Entrada digital 1I1	Entrada digital 2I1	Entrada digital 3I1
	Durante la parametrización asegúrese de que el inicio de la búsqueda del punto de referencia esté parametrizada en el sentido del contacto inversor. Sólo así puede garantizarse que el sensor del punto de referencia se encontrará siempre sin problemas.		
Entrada de habilitación	Entrada digital 1I2	Entrada digital 2I2	Entrada digital 3I2

Desarrollo del modo de operación "Búsqueda del punto de referencia"

El botón del sentido de desplazamiento positivo y negativo debe instalarse para cada canal.

1. Introduzca el valor de la coordenada del punto de referencia en el DB de parámetros (REFPT).
2. Introduzca el tipo de "Búsqueda del punto de referencia" en el DB de parámetros.
   Para ello, existen las posibilidades siguientes:

Inicio en sentido ...	Produce la sincronización ...	
plus	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **positivo**	REFPT_TYPE = 0
plus	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **negativo**	REFPT_TYPE = 1
menos	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **positivo**	REFPT_TYPE = 2
menos	la primera señal cero después de abandonar el sensor del punto de referencia en el sentido **negativo**	REFPT_TYPE = 3

3. Introduzca la velocidad de inicio;
   - Marcha rápida (REFPT_SPD = 0)
   - Marcha lenta (REFPT_SPD = 1)
4. Escriba y active los datos máquina.
5. Active la señal de control para el modo de operación "Búsqueda del punto de referencia" (MODE_IN = 3).
6. Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).
7. Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.
8. Conecte la señal de control para el sentido de marcha positivo o negativo o la señal de inicio (DIR_P = 1, DIR_M = 1 o START = 1).
9. Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).

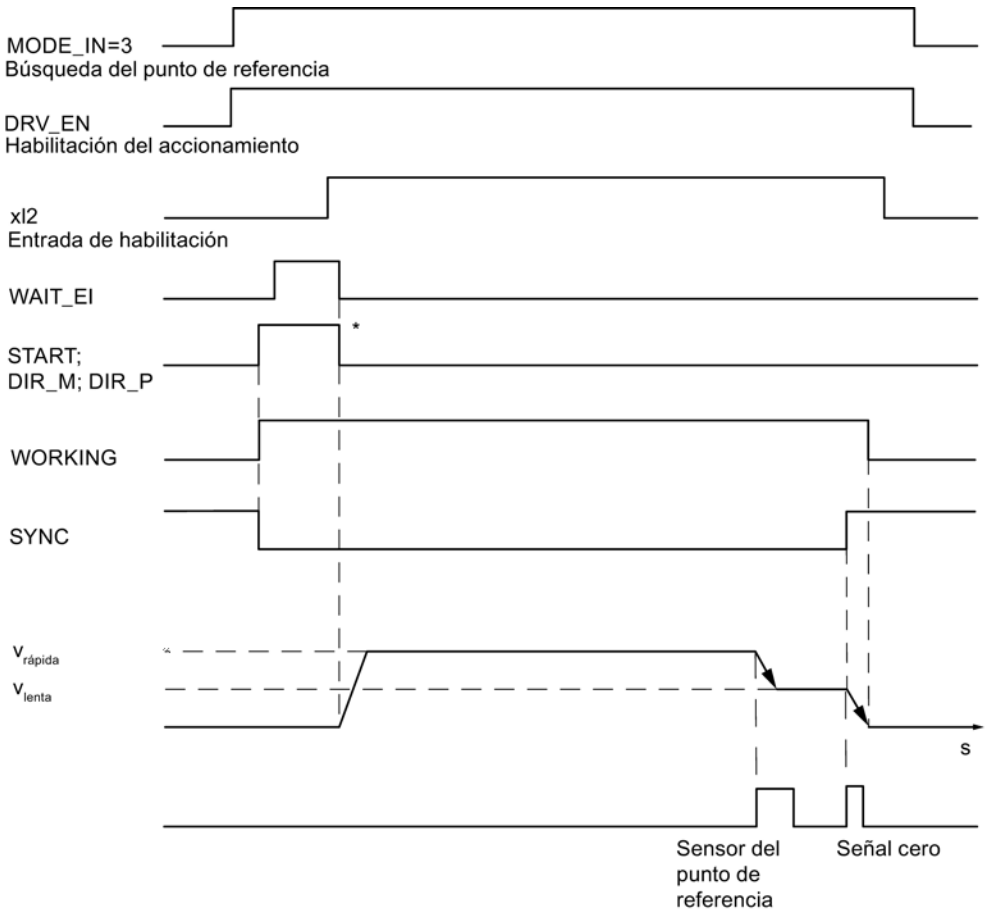
La tabla siguiente muestra los comandos de inicio para una búsqueda del punto de referencia.

Comando de inicio	Tarea	Comentario
DIR_P	El accionamiento se inicia en el sentido de los valores positivos. Se mueve, pues, en el sentido del final de la zona de desplazamiento.	Si en los datos máquina se introduce un sentido negativo, el FM 451 notifica un error de operador. No se realiza ninguna búsqueda del punto de referencia.
DIR_M	El accionamiento se inicia en el sentido de los valores negativos. Se mueve, pues, en el sentido del principio de la zona de desplazamiento.	Si en los datos máquina se introduce un sentido positivo , el FM 451 notifica un error de operador. No se realiza ninguna búsqueda del punto de referencia.
START	El accionamiento se inicia en el sentido que se ha introducido en los datos máquina.	

Nota

Para el eje rotativo rige lo siguiente: La reproducibilidad del punto de referencia sólo está garantizado si entre el valor **Final del eje rotativo** y el valor **Recorrido por vuelta** existe una relación de número entero.

La figura siguiente muestra un ejemplo del modo de operación "Búsqueda del punto de referencia".



* Las señales de inicio se desactivan con la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.0	START	BOOL	FALSE	1 = Iniciar operación de posicionamiento
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	3 = Búsqueda del punto de referencia
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Operación de posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	Coordenada del punto de referencia
52.0	REFPT_TYPE	DINT	L#0	Tipo de búsqueda del punto de referencia
99.0	REFPT_SPD	BOOL	TRUE	Velocidad de inicio para la búsqueda del punto de referencia 0 = Marcha rápida 1 = Marcha lenta

Efectos del modo de operación

- Con el inicio de la marcha se elimina la sincronización.
- La posición real se ajusta al valor de la coordenada del punto de referencia si se activa la señal de respuesta SYNC.
- La zona de trabajo se fija en el eje.
- Los distintos puntos dentro de la zona de trabajo conservan sus valores originales, pero se encuentran en posiciones nuevas.

Cancelación de la búsqueda del punto de referencia

El modo de operación "Búsqueda del punto de referencia" se cancela cuando

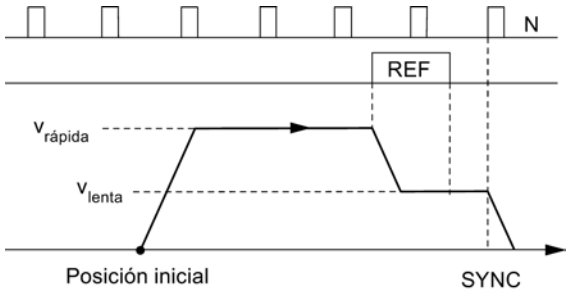
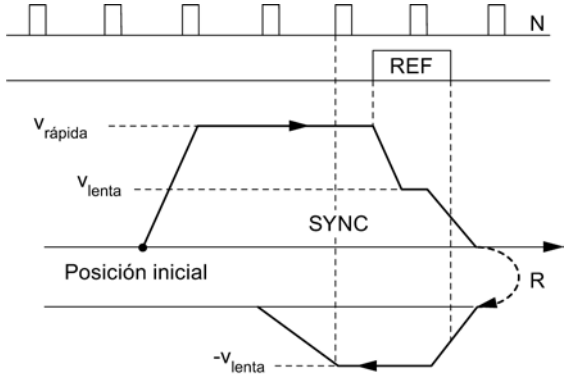
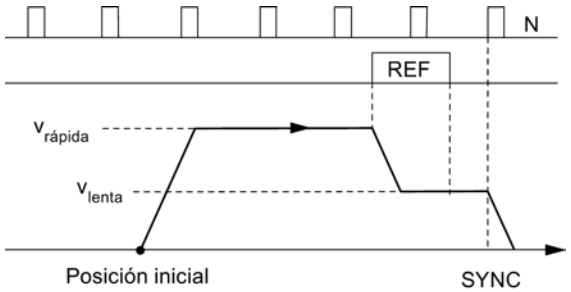
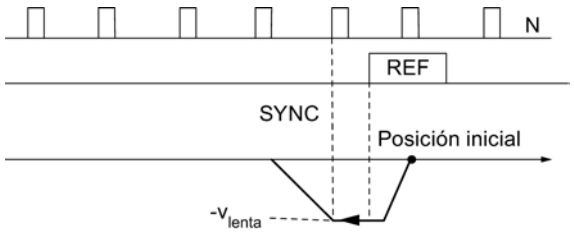
- la señal de habilitación del accionamiento se borra (DRV_EN = 0),
- se ha rebasado un límite de la zona de desplazamiento en un eje lineal.

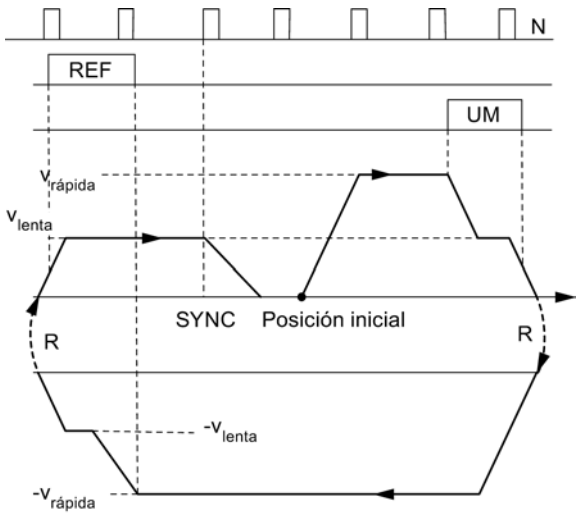
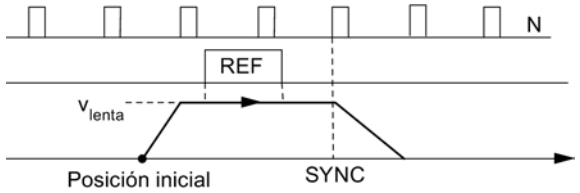
Búsqueda del punto de referencia dependiente de la posición de inicio

En una búsqueda del punto de referencia es preciso distinguir diversos casos que dependen

- de la posición del accionamiento al iniciar una búsqueda del punto de referencia,
- del sentido de arranque parametrizado,
- de la posición parametrizada de la señal cero respecto al sensor del punto de referencia.

La tabla siguiente muestra los casos de REFPT_TYPE = 0 y REFPT_TYPE = 1. Para REFPT_TYPE = 2 y REFPT_TYPE = 3 rigen las figuras según corresponda.

Condiciones para la búsqueda del punto de referencia	Desarrollo de la búsqueda del punto de referencia
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 0): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido positivo.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 1): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido negativo.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 0): • El sentido de inicio debe estar parametrizado en el sentido positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido positivo. • El contacto inversor es más positivo que el sensor del punto de referencia.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 1): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido negativo. • La posición de inicio de la búsqueda del punto de referencia se encuentra en el sensor del punto de referencia.	

Condiciones para la búsqueda del punto de referencia	Desarrollo de la búsqueda del punto de referencia
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE = 0): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido positivo. • El contacto inversor es más positivo que el sensor del punto de referencia.	
Ejemplo de búsqueda del punto de referencia (REFPT_TYPE=0): • El sentido de arranque es positivo. • La posición de la señal cero del sensor del punto de referencia está parametrizado en el sentido positivo. • Velocidad de inicio = velocidad lenta	
R = Inversión del sentido REF = Sensor del punto de referencia UM = Contacto inversor N = señal cero del encóder SYNC = Se ha alcanzado la sincronización	

Modo de operación "Modo incremental"

Definición

En el modo de operación "Modo incremental" el FM 451 puede mover el accionamiento

- a destinos **absolutos**,
- mover el accionamiento **de forma relativa** alrededor de una parte del trayecto en un sentido predeterminado.

El objetivo o los recorridos relativos se predeterminan como incrementos del FM 451. Tiene la posibilidad de insertar un máximo de 100 incrementos en una tabla, que son válidos tanto para el modo de operación "Modo incremental relativo" como para el "Modo incremental absoluto". Independientemente de la tabla de incrementos, es posible especificar el recorrido con el número de incremento 254 (Página 11071) y elnúmero de incremento 255 (Página 11072).

Con el incremento 252 (Página 11071) es posible mover el accionamiento, en primer lugar sin destino, en el sentido predeterminado (marcha sin fin). Si se produce un evento externo (xI3) se realiza el posicionamiento en el destino parametrizado.

Requisitos

- El eje debe estar parametrizado.
- El eje debe estar sincronizado.
- Los incrementos deben estar presentes en el módulo.

Interpretación de los incrementos

En función del modo incremental que se seleccione, el FM 451 interpreta los datos especificados de una manera u otra.

- Modo incremental absoluto: Los incrementos se interpretan como posición de destino absoluta.
  En el caso del eje rotativo no puede sobrepasarse el final del eje.
- Modo incremental relativo: Los incrementos se interpretan como distancia de la posición de inicio.
  En el caso del eje rotativo no puede sobrepasarse el final del eje (se admiten vueltas múltiples).

Nota

Para el modo de operación "Modo incremental relativo" sólo se permiten incrementos positivos.

Los incrementos obtienen su signo de las especificaciones de sentido DIR_P o DIR_M.

Desarrollo del modo de operación "Modo incremental" con número de incremento 1 a 100

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 1 a 100	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Anote los incrementos en las tablas (TRGL1; TRGL2).	
3	Escriba las tablas de incrementos (TRGL1/2WR_EN = 1).	
4	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
5	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
6	Introduzca el número de incremento (MODE_TYPE = 1 … 100).	

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 1 a 100	
7	Active la señal de control: • Eje lineal: – START; el sentido se determina de forma unívoca a través del destino y el valor real actual. • Eje rotativo: – START; se llega al punto de destino por el trayecto más corto. – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	Active la señal de control: • Eje lineal: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo • Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo
8	Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).	
Los pasos 2 y 3 se necesitan únicamente si no hay aún incrementos o si es preciso modificar las medidas existentes.		

Desarrollo del modo de operación "Modo incremental" con número de incremento 252

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 252	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
3	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
4	Introduzca el número de incremento (MODE_TYPE = 252).	
5	Introduzca la velocidad de inicio; • Marcha rápida (SPEED252 = 1) • Marcha lenta (SPEED252 = 0)	
6	Introduzca el incremento para el número de incremento 252 (TRG252_254).	
7	Active el bit de inicio para escribir el incremento (TRG252_254_EN = 1).	
8	Active la señal de control: • Eje lineal / Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	
9	Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).	
10	Si se produce un evento externo (entrada digital xI3) se realiza el posicionamiento en el destino parametrizado (incremento).	
Las acciones descritas en los pasos 6 y 7 también pueden realizarse durante la marcha. En este caso, el módulo FM 451 controla, hasta la indicación del incremento, un destino que se encuentra a la mitad de la zona de destino delante del final de carrera por software.		

Desarrollo del modo de operación "Modo incremental" con número de incremento 254

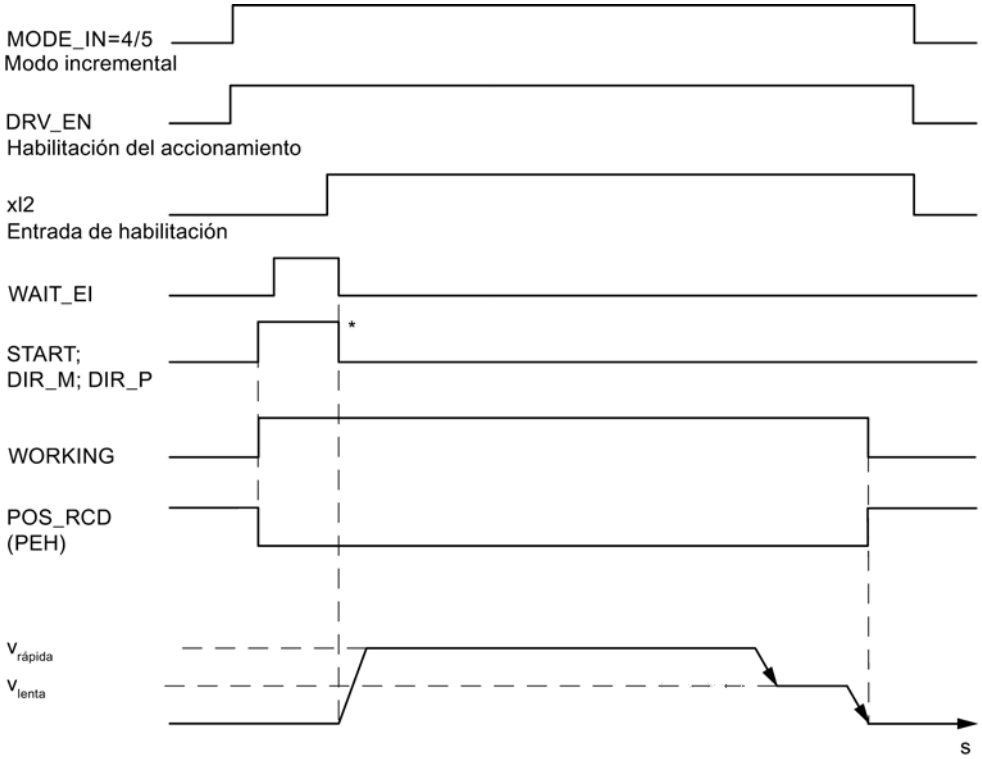
Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 254	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
3	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
4	Introduzca el número de incremento (MODE_TYPE = 254).	
5	Introduzca el incremento para el número de incremento 254 (TRG252_254).	
6	Active el bit de inicio para escribir el incremento (TRG252_254_EN = 1).	
7	Active la señal de control: • Eje lineal: – START; el sentido se determina de forma unívoca a través del destino y el valor real actual. • Eje rotativo: – START; se llega al punto de destino por el trayecto más corto. – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	Active la señal de control: • Eje lineal: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo • Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo
8	Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).	

Desarrollo del modo de operación "Modo incremental" con número de incremento 255

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 255	
1	Active la señal de control para el modo de operación "Modo incremental absoluto" (MODE_IN = 5).	Active la señal de control para el modo de operación "Modo incremental relativo" (MODE_IN = 4).
2	Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).	
3	Active el selector de función para "No evaluar la entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.	
4	Introduzca el número de incremento (MODE_TYPE = 255).	
5	Introduzca el incremento para el número de incremento 255 (TRG255).	
6	Introduzca el valor para la distancia de cambio de velocidad del número de incremento 255 (CHGDIF255).	
7	Introduzca el valor para la distancia de desconexión del número de incremento 255 (CUTDIF255).	
8	Active el bit de inicio para escribir el incremento y la distancia de desconexión y de cambio de velocidad (TRG255_EN = 1).	

Paso	Modo incremental absoluto	Modo incremental relativa
	Número de incremento 255	
9	Active la señal de control: • Eje lineal: – START; el sentido se determina de forma unívoca a través del destino y el valor real actual. • Eje rotativo: – START; se llega al punto de destino por el trayecto más corto. – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo	Active la señal de control: • Eje lineal: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo • Eje rotativo: – DIR_P; Inicio en sentido positivo – DIR_M; Inicio en sentido negativo
10	Llame la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).	

La figura siguiente muestra un ejemplo del modo de operación "Modo incremental".



* Las señales de inicio se desactivan con la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424).

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
15.0	START	BOOL	FALSE	1 = Iniciar posicionamiento
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.6	SPEED252	BOOL	FALSE	Velocidad de inicio para el modo incremental con número de incremento 252 • 0 = Marcha lenta • 1 = Marcha rápida
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	4 = Modo incremental relativo 5 = Modo incremental absoluto
17.0	MODE_TYPE	BYTE	B#16#0	Número de incremento 1 … 100, 252, 254 ó 255
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación
36.2	TRG252_254_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 252 ó 254
36.3	TRG255_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 255
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (número de incremento 1 ... 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (número de incremento 51 ... 100)
96.0	TRG252_254	DINT	L#0	Incremento para número de incremento 252 ó 254
100.0	TRG255	DINT	L#0	Incremento para número de incremento 255
104.0	CHGDIF_255	DINT	L#0	Distancia de cambio de velocidad para número de incremento 255
108.0	CUTDIF_255	DINT	L#0	Distancia de desconexión para número de incremento 255

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios	
100.0	CHGDIF_P	DINT	L#5000	Distancia de cambio de velocidad positiva	
104.0	CHGDIF_M	DINT	L#5000	Distancia de cambio de velocidad negativa	
108.0	CUTDIF_P	DINT	L#2000	Distancia de desconexión positiva	
112.0	CUTDIF_M	DINT	L#2000	Distancia de desconexión negativa	
120.0	TRGL1.TRG[1]	DINT	L#0	Número de incremento 1	Tabla de incrementos 1
. . .	. . .	. . .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Número de incremento 50	

Dirección	Nombre	Tipo	valor inicial	Comentarios	
320.0	TRGL2.TRG[51]	DINT	L#0	Número de incremento 51	Tabla de incrementos 2
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Número de incremento 100	

Distancia residual

La distancia residual es la diferencia entre el destino (incremento) y el valor real. La distancia residual es

- positivo si aún no ha alcanzado el destino.
- negativo si ya se ha rebasado el destino.

En un eje rotativo no puede utilizarse la distancia residual mostrada.

Desactivación del modo incremental

El modo de operación "Modo incremental" se desactiva si el FM 451 recibe una señal de parada (STOP = 1).

Tras la desactivación de la marcha, se mantiene la distancia residual.

La distancia residual en el "Modo incremental relativo" y el "Modo incremental absoluto con el número de incremento 252" puede realizarse hasta el final si

- el modo de operación no ha cambiado y
- el número de incremento no ha cambiado y
- el sentido no ha cambiado y
- la distancia residual es mayor que la distancia de desconexión parametrizada.

La distancia residual se realiza ejecutando de nuevo el "Modo incremental relativo" o el "Modo incremental absoluto" con el número de incremento 252 sin cambios.

Cancelación del modo incremental

El modo de operación "Modo incremental" se cancela si la señal "Habilitación del accionamiento" se borra (DRV_EN = 0).

Borrar distancia residual

La petición "Borrar distancia residual" borra una distancia residual pendiente.

En el "modo incremental con número de incremento 252" la marcha comienza de nuevo con la marcha sin fin después de la función "Borrar distancia residual".

Al llamar otro modo de operación o iniciar el modo de operación en el otro sentido, también se borra la distancia residual pendiente.

Dato utilizado en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.2	DELDIST_EN	BOOL	FALSE	1 = Borrar distancia residual

Petición "Ajustar valor real" / "Ajustar valor real al vuelo" / "Cancelar preselección del valor real"

Definición

La petición "Ajustar valor real" o "Ajustar valor real al vuelo" permite asignar una nueva coordenada al estado actual del encóder. La zona de trabajo se proyecta sobre otra zona del eje.

El desplazamiento de la zona de trabajo se calcula con ($REAL_{nuevo}$ - $REAL_{actual}$).

- $REAL_{nuevo}$ es el valor predeterminado
- $REAL_{actual}$ es el valor real en el instante de la ejecución

Requisitos

- El eje debe estar parametrizado.
- El eje debe estar sincronizado.

Para la función "Ajustar valor real" es preciso finalizar el proceso de posicionamiento.

Desarrollo de la petición

1. Introduzca las coordenadas AVAL o FVAL para el valor real ($REAL_{nuevo}$).
   - Eje lineal:
     El valor real indicado se debe seleccionar de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admitido tras la llamada de la petición.
     El valor de decalaje que resulta de ($REAL_{nuevo}$ - $REAL_{actual}$) tiene que ser menor o igual al valor de la zona de desplazamiento admisible (máximo 100 m o 1000 m, respectivamente).
   - Eje rotativo:
     Para el valor real indicado rige:
     0 ≤ Valor real < Final del eje rotativo
2. Active el bit de inicio correspondiente (AVAL_EN = 1 o FVAL_EN = 1).

Se ejecuta de inmediato "Ajustar valor real".

"Ajustar valor real al vuelo" se ejecuta con el flanco ascendente siguiente en la entrada digital xI3.

Datos utilizados en el DB de canal

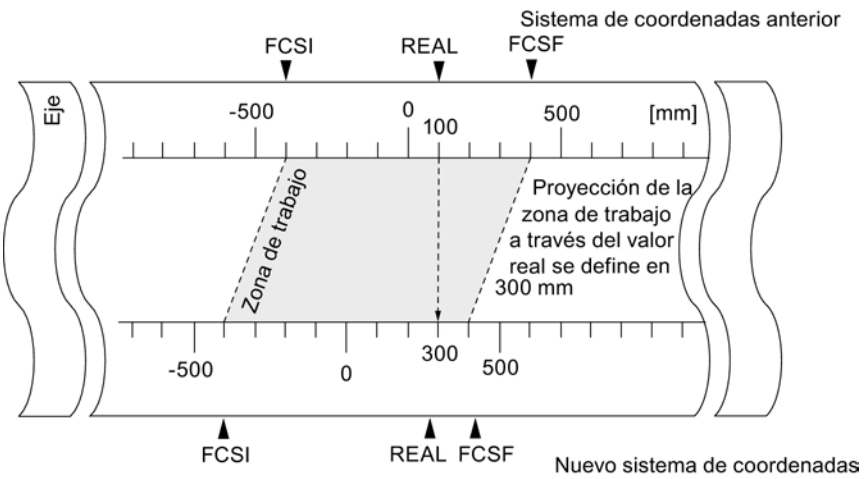
Dirección	Nombre	Tipo	valor inicial	Comentarios
35.7	AVAL_EN	BOOL	FALSE	1 = Ajustar valor real
36.0	FVAL_EN	BOOL	FALSE	1 = Ajustar valor real al vuelo
84.0	AVAL	DINT	L#0	Coordenada de "Ajustar valor real"
88.0	FVAL	DINT	L#0	Coordenada de "Ajustar valor real al vuelo"

Efectos de la petición

En el ejemplo "Ajustar valor real" en 300 mm se ve cómo esta petición proyecta la zona de trabajo a una posición concreta del eje. De ello se derivan los siguientes efectos:

- La posición real se fija en el valor de la coordenada del valor real.
- La zona de trabajo se desplaza hacia el eje.
- Los distintos puntos (p. ej., fin del final de carrera por software) dentro de la zona de trabajo conservan su valor original, pero se encuentran en posiciones nuevas.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Ajustar valor real".

Ajustar valor real	FCSI	REAL	FCSF
	-400	100	400
	-400	300	400

Deshacer una petición ("Cancelar preselección del valor real")

Con la petición "Cancelar preselección del valor real" se deshace el último desplazamiento de la zona de trabajo que se ha creado con "Ajustar valor real".

Un "Ajustar valor real al vuelo" iniciado no se puede borrar antes de su ejecución mediante un flanco ascendente en la entrada digital xI3. Pero sí puede sobrescribirse con un nuevo "Ajustar valor real al vuelo". Con un arranque del módulo estas peticiones se inicializan.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.3	AVALREM_EN	BOOL	FALSE	1 = Cancelar preselección del valor real

Petición "Decalaje de señal cero"

Definición

Con la petición "Decalaje de señal cero" se desplaza en el valor introducido la señal cero del sistema de coordenadas.

El signo define el sentido del decalaje en el sistema de coordenadas.

Calcular nueva coordenada

Todos los valores del sistema desplazado de coordenadas se calculan con la fórmula siguiente:

$\mathbf{Coordenada_{nueva} = coordenada_{antigua} + (DSC_{nueva} - DSC_{antigua})}$

$\mathbf{DSC_{antigua}}$ indica un decalaje anterior de señal cero.

Si antes de la llamada no ha habido activo ningún decalaje de señal cero, introduzca para $\mathbf{DSC_{antiguo}}$ el valor 0.

Con esto es posible calcular qué valores de coordenadas adoptan, por ejemplo, los finales de carrera por software.

Requisitos

- El posicionamiento debe haber finalizado.
- El eje debe estar parametrizado.

Desarrollo de la petición

1. Introduzca el valor ZOFF para el decalaje de señal cero.
   - Eje lineal:
     El decalaje de señal cero se tiene que escoger de tal modo que los finales de carrera por software se encuentren todavía dentro de la zona de desplazamiento admisible tras la llamada de la petición.
   - Eje rotativo:
     Para el decalaje de señal cero rige:
     Valor del decalaje de señal cero ≤ Fin del eje rotativo
2. Active el bit de inicio correspondiente (ZOFF_EN = 1).

Datos utilizados en el DB de canal

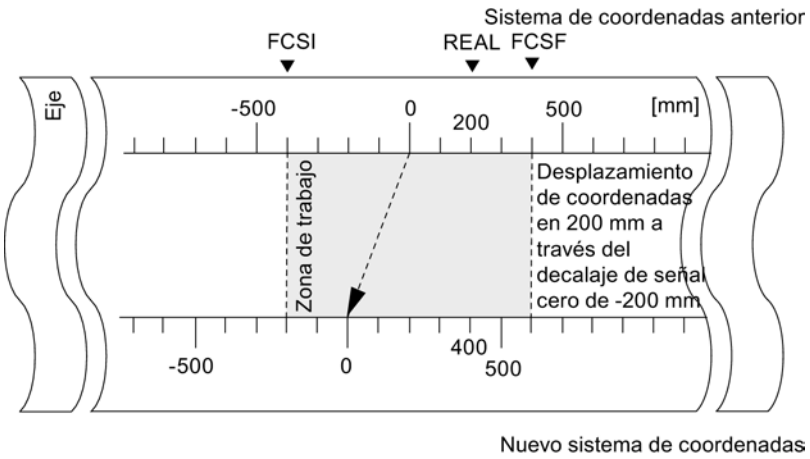
Dirección	Nombre	Tipo	valor inicial	Comentarios
36.1	ZOFF_EN	BOOL	FALSE	1 = Activar decalaje de señal cero
80.0	ZOFF	DINT	L#0	Decalaje de señal cero

Efectos de la petición en un eje lineal

En el ejemplo de un decalaje de señal cero de -200 mm se ve cómo esta petición desplaza el sistema de coordenadas en sentido positivo. De ello se derivan los siguientes efectos:

- La zona de trabajo **no** se desplaza físicamente.
- A cada uno de los puntos (como, p.ej. a los finales de carrera por software) se les asignan nuevos valores de coordenadas.

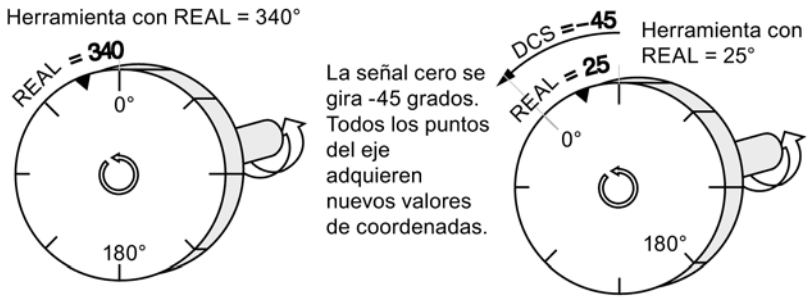
La tabla siguiente muestra el desplazamiento del sistema de coordenadas mediante el "Decalaje de señal cero".

Decalaje de señal cero	FCSI	REAL	FCSF
	-400	200	400
	-200	400	600

Efectos de la petición en un eje rotativo

En el ejemplo de un decalaje de señal cero en -45° se ve cómo esta petición **gira** el sistema de coordenadas.

La figura siguiente muestra el giro del sistema de coordenadas mediante decalaje de la señal cero.



Considerando un $DSC_{antiguo}$ = 0 resulta un nuevo valor de 385°

Como al final del eje rotativo en un sentido de giro positivo el valor real empieza de nuevo en 0 el valor real de 25º se calcula a partir de:

$Coordenada_{nueva} = coordenada_{antigua} - (DSC_{nueva} - DSC_{antigua})$ -final de eje rotativo

Pérdida de la sincronización

Si se pierde la sincronización a causa de un error o se reactiva mediante "Búsqueda del punto de referencia" **se mantiene** activo un decalaje de señal cero.

Deshacer petición

Al definir un decalaje de señal cero de 0 mm se recupera el decalaje de señal cero existente.

Petición "Definir punto de referencia"

Definición

Con la petición "Definir punto de referencia" se sincroniza el eje. Esta petición desplaza la zona de trabajo. Todos los decalajes generados por la función "Ajustar valor real" se conservan.

Este ajuste proyecta la zona de trabajo sobre el eje. Así, con distintas entradas de valores, la zona de trabajo puede encontrarse en posiciones diferentes dentro del rango físico del eje.

Requisitos

- El posicionamiento debe haber finalizado.
- El eje debe estar parametrizado.

Desarrollo de la petición

1. Introduzca el valor REFPT para las coordenadas del punto de referencia.
   – Eje lineal:

   La coordenada del punto de referencia no puede encontrarse fuera de los finales de carrera por software. Esto rige también para la coordenada del punto de referencia de un sistema de coordenadas desplazado.

   – Eje rotativo:
     Para la coordenada del punto de referencia rige:
     0 ≤ Coordenada del punto de referencia < Final del eje rotativo
2. Active el bit de inicio correspondiente (REFPT_EN = 1).

Datos utilizados en el DB de canal

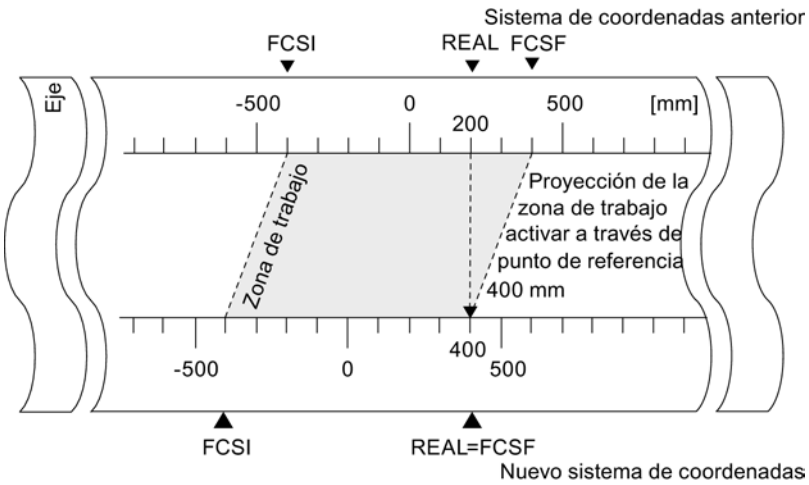
Dirección	Nombre	Tipo	valor inicial	Comentarios
25.0	SYNC	BOOL	FALSE	1 = Eje sincronizado
35.6	REFPT_EN	BOOL	FALSE	1 = Definir punto de referencia
92.0	REFPT	DINT	L#0	Coordenada del punto de referencia

Efectos de la petición

En el ejemplo "Definir punto de referencia" en 400 mm se ve cómo esta petición proyecta la zona de trabajo a una posición física concreta del eje. De ello se derivan los siguientes efectos:

- La posición real se fija en el valor de la coordenada del punto de referencia.
- La zona de trabajo se desplaza hacia el eje.
- Los distintos puntos (por ejemplo, fin del final de carrera por software) conservan su valor original pero se encuentran en posiciones físicas nuevas.
- El bit SYNC de las señales de respuesta se activa.

La tabla siguiente muestra el desplazamiento de la zona de trabajo en el eje a través de la función "Definir punto de referencia".

Definir punto de referencia	FCSI	REAL	FCSF
	-400	200	400
	-400	400	400

Particularidades del encóder absoluto

Esta petición es necesaria para un reajuste del encóder absoluto (Página 11057).

Selector de función "Medición de longitud" / "Captura de bordes"

Definición

Con las funciones "Medición de longitud" y "Captura de bordes" es posible calcular la longitud de una pieza de trabajo.

El selector de función "Medición de longitud" o "Captura de bordes" se activa y se mantiene así hasta que se desconecta o se selecciona otro método de medición. Si se seleccionan ambos métodos de medición a la vez, la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) activa la medición de longitud.

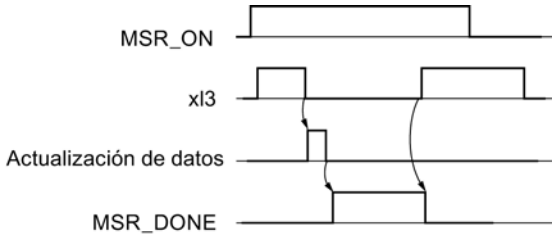
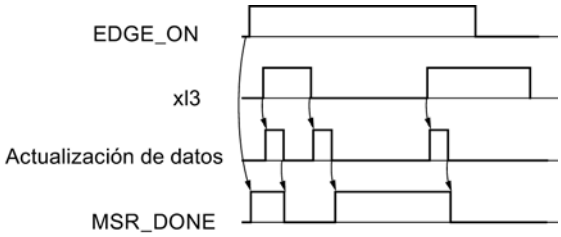
Requisitos

- El posicionamiento debe haber finalizado.
- El eje debe estar parametrizado.
- El eje debe estar sincronizado.
- En la entrada xI3 hay conectado un conector **libre de rebotes**.
- Si estos selectores de función se seleccionan durante un posicionamiento, la "Medición de longitud" y la "Captura de bordes" pueden no realizarse hasta el posicionamiento siguiente.

Desarrollo de las mediciones

Según el tipo de medición, el FM 451 actualiza los datos del módulo en momentos distintos. El FM 451 notifica cada actualización en un parámetro en la interfaz de realimentación.

La tabla siguiente muestra el desarrollo de las mediciones.

Desarrollo	Medición de longitud	Captura de bordes
		
1	Active el selector de función para la "Medición de longitud" (MSR_ON = 1).	Dado el caso, introduzca un valor EDGEDIST para la separación mínima de bordes en el DB de parámetros. Escriba y active los datos máquina.
2	Inicie un posicionamiento.	Active el selector de función de "Captura de bordes" (EDGE_ON = 1). Se activa el parámetro MSR_DONE.
3	El flanco ascendente de la entrada xI3 inicia la medición de longitud.	Inicie un posicionamiento.
4	El flanco descendente de la entrada xI3 pone fin a una medición en curso. El FM 451 actualiza los datos Valor inicial, Valor final y Longitud.	El flanco ascendente de la entrada xI3 inicia la medición. Tras el procedimiento de la parte del trayecto "Distancia mínima entre bordes" la medición es válida. Esto se notifica mediante MSR_DONE = 0. Los resultados de la medición se actualizan y se pueden leer, el valor inicial de la medición se registra; el valor final y la longitud pasan a ser -1.
5	Con el parámetro activado MSR_DONE el FM 451 comunica la actualización de los datos. El parámetro indica que la medición ha finalizado. Los resultados de la medición se pueden leer.	Tras la actualización, el FM 451 notifica el cambio desactivando el parámetro MSR_DONE.
6	El inicio de otra medición con el flanco ascendente de xI3 desactiva el parámetro MSR_DONE.	El flanco descendente de la entrada xI3 pone fin a una medición en curso. El FM 451 actualiza los datos del valor final de la medición y la longitud.
7	-	Tras la actualización, el FM 451 notifica el cambio activando el parámetro MSR_DONE. Los resultados de la medición se pueden leer.
8	-	El inicio de otra medición con el flanco ascendente de xI3 desactiva el parámetro MSR_DONE.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
25.1	MSR_DONE	BOOL	FALSE	1 = Medición de longitud cancelada/Captura de bordes finalizada
34.3	EDGE_ON	BOOL	FALSE	1 = Captura de bordes activada
34.4	MSR_ON	BOOL	FALSE	1 = Medición de longitud activada
35.0	MDWR_EN	BOOL	FALSE	1 = Escribir datos máquina
35.1	MD_EN	BOOL	FALSE	1 = Activar datos máquina
37.0	MSRRD_EN	BOOL	FALSE	1 = Leer valores medidos
136.0	BEG_VAL	DINT	L#0	Valor inicial de la medición de longitud/Captura de bordes
140.0	END_VAL	DINT	L#0	Valor final de la medición de longitud/Captura de bordes
144.0	LEN_VAL	DINT	L#0	Longitud

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
4.0	EDGEDIST	DINT	L#0	Distancia mínima entre bordes en la captura de bordes
Si el final del proceso de medición se encuentra dentro de este rango, la medición se desestima (BEG_VAL, END_VAL y LEN_VAL = -1).				

Condiciones marco para una medición de longitud

- La distancia entre el flanco de activación y el flanco de desactivación en la entrada xI3 tiene que ser tal que su programa en la CPU pueda evaluar correctamente el resultado de la medición antes de iniciar una nueva medición.
- La distancia mínima entre el flanco ascendente y descendente en la entrada xI3, al igual que entre el flanco descendente y el siguiente ascendente en la entrada xI3, tiene que ser mayor que 8 ms.

Medición con errores

En caso de una medición de longitud/Captura de bordes con errores, el FM 451 devuelve el valor -1 para la longitud. Una medición es errónea cuando

- en un eje rotativo la longitud medida es superior a 2^{31},
- al mismo tiempo el FM 451 detecta el flanco de activación y el de desactivación, por ejemplo, por rebotes del conector.

Selector de función "Aproximación en bucle"

Definición

Con la "Aproximación en bucle" define el sentido en el que se avanza hacia el punto de destino con cierre de fuerza. La aproximación en bucle puede utilizarse cuando sólo se puede garantizar el cierre de fuerza en un sentido entre el motor y el eje.

El destino al que se realiza la aproximación en contra del sentido predefinido se rebasa al principio. A continuación, el módulo realiza una inversión del sentido y realiza la aproximación al destino en el sentido predeterminado.

Requisitos

- El eje debe estar parametrizado.
- El eje debe estar sincronizado.
- En una aproximación en bucle en contra del sentido del desplazamiento hacia el destino, la posición de destino se encuentra:
  - en el sentido de desplazamiento positivo
    Destino < SEE - ½ zona de destino - distancia de desconexión positiva - distancia de cambio de velocidad negativa
  - en el sentido de desplazamiento negativo
    Destino < SEA + ½ zona de destino + distancia de desconexión negativa + distancia de cambio de velocidad positiva
- Una aproximación en bucle no se lleva a cabo, si la aproximación al destino se realiza en el sentido de la aproximación en bucle. En este caso se realiza un posicionamiento en modo incremental sin inversión del sentido.
- Hay que conocer el transcurso del modo de operación "Modo incremental" (Página 11014).

Desarrollo de la aproximación en bucle

1. Active la señal de control para el modo de operación "Modo incremental absoluto/relativo" (MODE_IN = 5/4).
2. Active la señal de control para la habilitación del accionamiento (DRV_EN = 1).
3. Active el selector de función "Entrada de habilitación" (EI_OFF = 1) o cablee la entrada de habilitación para el canal correspondiente.
4. Introduzca el número de incremento (MODE_TYPE = 1 … 100, 254, 255).
5. Active el selector de función "Aproximación en bucle" (PLOOP_ON / MLOOP_ON = 1).
6. Inicie el modo incremental.

Datos utilizados en el DB de canal

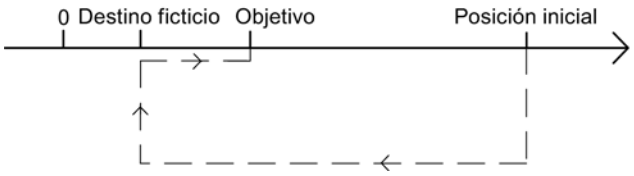
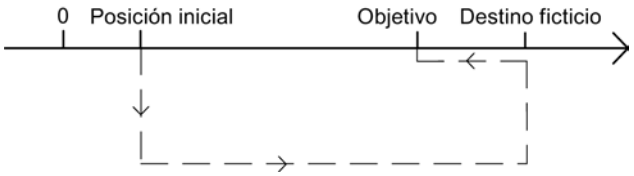
Dirección	Nombre	Tipo	valor inicial	Comentarios
15.0	START	BOOL	FALSE	1 = Iniciar posicionamiento
15.2	DIR_M	BOOL	FALSE	1 = Sentido negativo
15.3	DIR_P	BOOL	FALSE	1 = Sentido positivo
15.7	DRV_EN	BOOL	FALSE	1 = Conectar habilitación del accionamiento
16.0	MODE_IN	BYTE	B#16#0	4 / 5 = Modo incremental relativo/absoluto
17.0	MODE_TYPE	BYTE	B#16#0	Número de incremento 1 … 100, 254 ó 255
34.0	PLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido positivo
34.1	MLOOP_ON	BOOL	FALSE	1 = Aproximación en bucle en sentido negativo
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Destino ficticio

Si inicia un posicionamiento en un destino que se encuentra en el sentido opuesto a la aproximación en bucle parametrizada, el módulo determina para este destino un destino ficticio en el que éste ejecuta una inversión del sentido y, después, se aproxima al destino en el sentido correcto.

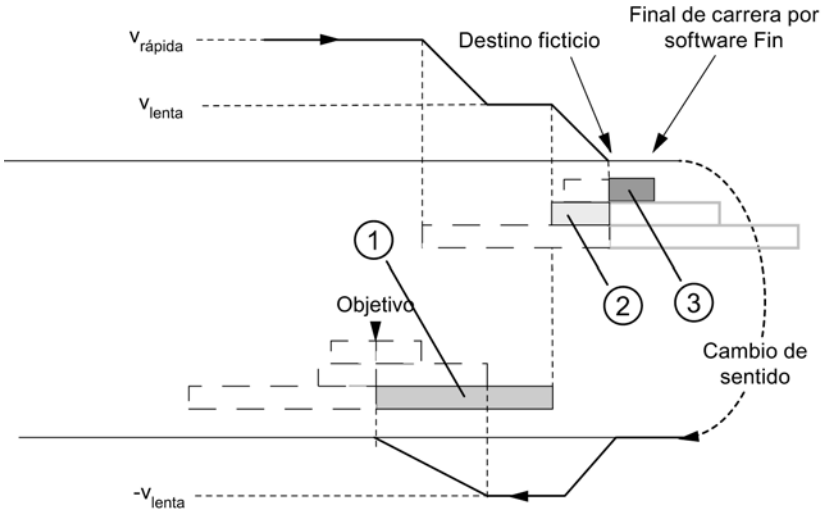
Este destino ficticio **tiene que** encontrarse al menos a media zona de destino delante del final de carrera por software correspondiente.

La eliminación del destino ficticio del destino parametrizado se determina en función del sentido. La tabla siguiente muestra el cálculo de la situación del destino ficticio en la aproximación en bucle.

Preajustes	Posición del destino ficticio
Parametrización: Bucle + (cierre de fuerza positivo) y marcha en sentido negativo	El destino ficticio (**Destino$_f$**) tiene el valor: **Destino$_f$ = Destino - Distancia de desconexión negativa - distancia de cambio de velocidad positiva** 
Parametrización: Bucle - (cierre de fuerza negativo) y marcha en el sentido positivo	El destino ficticio (**Destino$_f$**) tiene el valor: **Destino$_f$ = Destino + Distancia de desconexión positiva + distancia de cambio de velocidad negativa** 

Ejemplo

A continuación se muestra la posición del destino ficticio tomando como base un posicionamiento con aproximación en bucle negativa hacia un destino máximo.



① Distancia de cambio de velocidad negativa
② Distancia de desconexión positiva
③ ½ zona de destino

Selector de función "Entrada de habilitación"

Definición

La entrada de habilitación es una entrada externa con la que puede habilitarse un posicionamiento a través de un evento externo.

Evaluar entrada de habilitación (EI_OFF = 0)

La entrada de habilitación correspondiente (xI2) para el canal debe estar cableada.

Con ello tiene la posibilidad de preparar el inicio de un posicionamiento. El posicionamiento se inicia independientemente del desarrollo del programa de usuario creando una señal "1" en la entrada de habilitación.

Tiene las siguientes posibilidades para el tipo de evaluación de la entrada de habilitación:

- controlado por nivel (EI_TYPE = 0)
  La marcha comienza cuando se crea una señal "1" en la entrada de habilitación y se desactiva cuando se crea una señal "0" en la entrada de habilitación.
- controlado por flanco (EI_TYPE = 1)
  La marcha comienza con la detección de un flanco ascendente en la entrada de habilitación. La siguiente curvav de señales en esta entrada ya no afecta al desarrollo del desplazamiento iniciado.

No evaluar entrada de habilitación (EI_OFF = 1)

Si desconecta la evaluación de la entrada de habilitación, un modo de operación comienza inmediatamente después de detectar la señal de inicio. A continuación no es posible preparar un modo de operación ni iniciarlo en un instante posterior concreto.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
34.2	EI_OFF	BOOL	FALSE	1 = No evaluar entrada de habilitación

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
99.1	EI_TYPE	BOOL	FALSE	0 = Controlado por nivel 1 = Controlado por flanco

Petición "Leer datos de posición"

Definición

Con la petición "Leer datos de posición" es posible leer el incremento, la distancia residual y la velocidad en el instante actual.

Desarrollo de la petición

1. Active el bit de inicio en el DB de canal (ACTSPD_EN = 1).
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
37.1	ACTSPD_EN	BOOL	FALSE	1 = Leer datos de posición
112.0	ACTSPD	DINT	L#0	Velocidad actual
116.0	DIST_TO_GO	DINT	L#0	Distancia residual
120.0	ACT_TRG	DINT	L#0	Incremento actual

Petición "Leer datos del encóder"

Definición

Con la petición "Leer datos de encóder" se leen los datos actuales del encóder, así como el valor del reajuste del encóder absoluto.

Requisitos

Es posible leer el valor para el reajuste del encóder absoluto (Página 11057) después de ejecutar la petición "Definir punto de referencia" (Página 11025).

Desarrollo de la petición

1. Active el bit de inicio en el DB de canal (ENCVAL_EN = 1).
2. Los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
37.2	ENCVAL_EN	BOOL	FALSE	1 = Leer valores del encóder
124.0	ENCVAL	DINT	L#0	Valor real del encóder (representación interna)

Dirección	Nombre	Tipo	valor inicial	Comentarios
128.0	ZEROVAL	DINT	L#0	Último valor de la marca cero (representación interna)
132.0	ENC_ADJ	DINT	L#0	Reajuste del encóder absoluto

Señales de realimentación del posicionamiento

Definición

Con la función "Señales de realimentación para el posicionamiento" se informa del estado actual del posicionamiento.

Desarrollo

Con cada llamada de la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) los datos se depositan en el DB de canal.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
23.0	ST_ENBLD	BOOL	FALSE	1 = Habilitar inicio
23.1	WORKING	BOOL	FALSE	1 = Posicionamiento en curso
23.2	WAIT_EI	BOOL	FALSE	1 = El eje espera una habilitación externa
23.4	SPEED_OUT	BOOL	FALSE	0 = Marcha lenta 1 = Marcha rápida
23.5	ZSPEED	BOOL	FALSE	1 = El eje se encuentra en la zona de parada
23.6	CUTOFF	BOOL	FALSE	1 = El eje se encuentra en la zona de desconexión
23.7	CHGOVER	BOOL	FALSE	1 = El eje se encuentra en la zona de cambio de de velocidad
24.0	MODE_OUT	BYTE	B#16#0	Modo de operación activo
25.2	GO_M	BOOL	FALSE	1 = El eje se desplaza en sentido negativo
25.3	GO_P	BOOL	FALSE	1 = El eje se desplaza en sentido positivo
25.5	FVAL_DONE	BOOL	FALSE	1 = Ajuste del valor real al vuelo finalizado
25.7	POS_RCD	BOOL	FALSE	1 = Posición alcanzada
26.0	ACT_POS	DINT	L#0	Valor real actual (posición actual del eje)

Consultar las señales de realimentación del diagnóstico

Definición

Con las "Señales de respuesta del diagnóstico" se recibe información sobre los eventos de diagnóstico que se han producido.

Desarrollo

1. Cuando el módulo introduce un nuevo evento en el búfer de diagnóstico, activa el bit DIAG en la interfaz de respuesta. Cada aparición de un error (Página 11085) provoca una entrada en el búfer de diagnóstico. Al borrar el búfer de diagnóstico se activa el bit DIAG.
2. Si no es posible llamar un modo de operación o controlar un modo de operación activo, o si esta operación se ha realizado de forma errónea, el módulo activa un error de operador OT_ERR en la interfaz de respuesta. La causa del error se registra en el búfer de diagnóstico. Mientras el error de operador está activo, no es posible iniciar un modo de operación ni continuar el modo de operación detenido. Un error de operador existente se confirma con OT_ERR_A = 1.
3. Cuando el módulo detecta una petición de escritura con datos erróneos, activa el bit DATA_ERR en la interfaz de respuesta. La causa del error se registra en el búfer de diagnóstico.
4. Las señales de respuesta se depositan en el DB de canal.
5. Si se ha leído el búfer de diagnóstico, el módulo restablece el bit DIAG a 0 en todos los canales.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
22.2	DIAG	BOOL	FALSE	1 = Búfer de diagnóstico modificado
22.3	OT_ERR	BOOL	FALSE	1 = Error de operador
22.4	DATA_ERR	BOOL	FALSE	1 = Error de datos

Configurar FM 451

Parametrización del FM 451

Especificar canales

Seleccionar sistema de medidas

Seleccionar sistema de medidas

Para la entrada y salida de datos es posible elegir entre los siguientes sistemas de medida:

- Milímetros (ajuste predeterminado)
- Inch
- Grados

Nota

Si se modifica el sistema de medida en el cuadro de diálogo "Especificar canales", los valores se convierten al nuevo sistema. Aquí se pueden producir errores de redondeo.

Si se modifica el sistema de medida mediante el dato máquina, los valores no se convierten automáticamente.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentario
8.0	UNITS	DINT	L#1	**Sistema de medida** Valores permitidos: • 1 = 10^{-3} mm • 2 = 10^{-4} pulgadas • 3 = 10^{-4} grados • 4 = 10^{-2} grados • 6 = 10^{-3} grados

Sistema de medida estándar en la ayuda en pantalla

Para la indicación de valores mínimos y máximos se utiliza el sistema de medida mm. Por ello, para determinar los valores límite en los otros sistemas de medida, realice la conversión siguiente:

Para convertir de		calcule
mm → pulgadas		Valor límite (inch) = valor límite (mm) × 0,1 [1]
mm → grados	10^{-4} (4 decimales)	Valor límite (grados) = valor límite (mm) × 0,1
	10^{-3} (3 decimales)	Valor límite (grados) = valor límite (mm) × 1
	10^{-2} (2 decimales)	Valor límite (grados) = valor límite (mm) × 10
pulgadas → grados		Valor límite (grados) = valor límite (pulgadas)

[1] En el valor máximo el número de decimales repercute en el número de dígitos que preceden a la coma. En el sistema de medida Inch se utilizan 4 decimales, por lo tanto se pueden introducir como máximo 100 000,0000 inch. El sistema de medida Milímetros utiliza 3 decimales y por lo tanto se pueden introducir como máximo 1 000 000,000 mm.

Relación entre los incrementos y el sistema de medida

Las señales del encóder de un dispositivo conectado son evaluadas por el módulo y se convierten al sistema actual. Para la conversión se utiliza la resolución (Página 11065).

Si el módulo de posicionamiento

- ha contado 10 incrementos y
- ha ajustado una resolución de 100 µm por incremento a través de los datos de encóder parametrizados,

esto significa que el eje se ha movido un trayecto de 1 mm.

Copiar canales

Copiar canales

Este parámetro permite copiar un canal en otro canal.

1. Seleccione el canal cuyos parámetros desee copiar (origen del canal).
2. Active el canal en el que desee copiar los parámetros (destino del canal).
3. Pulse el botón "Copiar".

Accionamiento canal n

Seleccionar el modo de control

Visión global de los tipos de control

Sinopsis

El modo de control define el modo en que se utiliza un motor en las 4 salidas digitales (Q) de un canal a través de la etapa de potencia. Para el módulo de posicionamiento existen 4 modos de control por canal:

- Modo de control 1 (Página 11043)
- Modo de control 2 (Página 11044)
- Modo de control 3 (Página 11045)
- Modo de control 4 (Página 11046)

La siguiente tabla muestra los estados de las 4 salidas digitales para cada modo de control, donde x representa los canales 1, 2 y 3.

	Marcha rápida		Marcha lenta		PEH La posición alcanza la parada
	Sentido +	Sentido -	Sentido +	Sentido -	
Modo de control 1					
xQ0	1	1	0	0	-
xQ1	0	0	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-

	Marcha rápida		Marcha lenta		PEH La posición alcanza la parada
	Sentido +	Sentido -	Sentido +	Sentido -	
Modo de control 2					
xQ0	1	1	0	0	0
xQ1	0	0	0	0	1
xQ2	1	0	1	0	0
xQ3	0	1	0	1	0
Modo de control 3					
xQ0	1	1	0	0	-
xQ1	1	1	1	1	-
xQ2	1	0	1	0	-
xQ3	0	1	0	1	-
Modo de control 4					
xQ0	1	0	0	0	-
xQ1	1	0	1	0	-
xQ2	0	1	0	0	-
xQ3	0	1	0	1	-

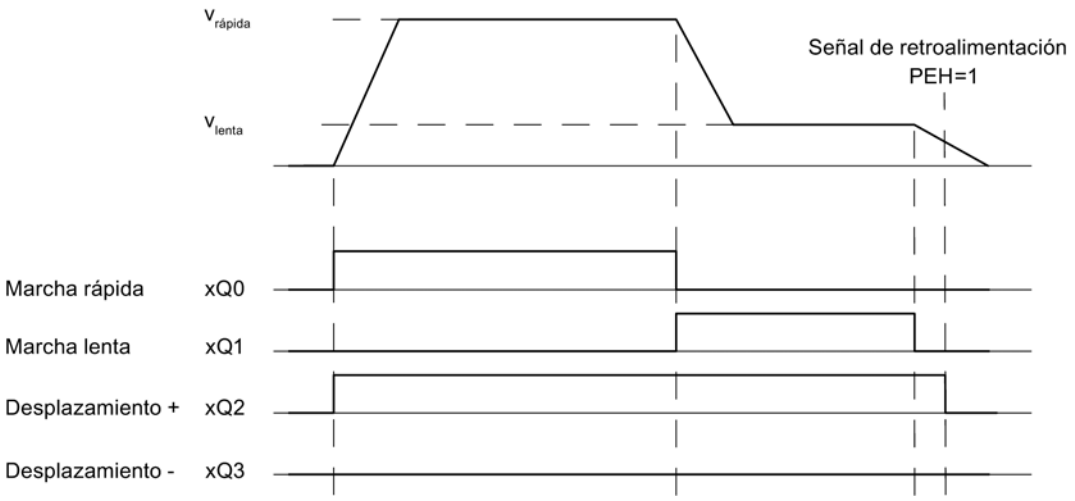
Modo de control 1

Modo de control 1

En el modo de control 1 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida
- xQ1: Marcha lenta
- xQ2: Desplazamiento +
- xQ3: Desplazamiento -

La figura siguiente muestra la función de las salidas digitales en el modo de control 1.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	1 = Modo de control 1

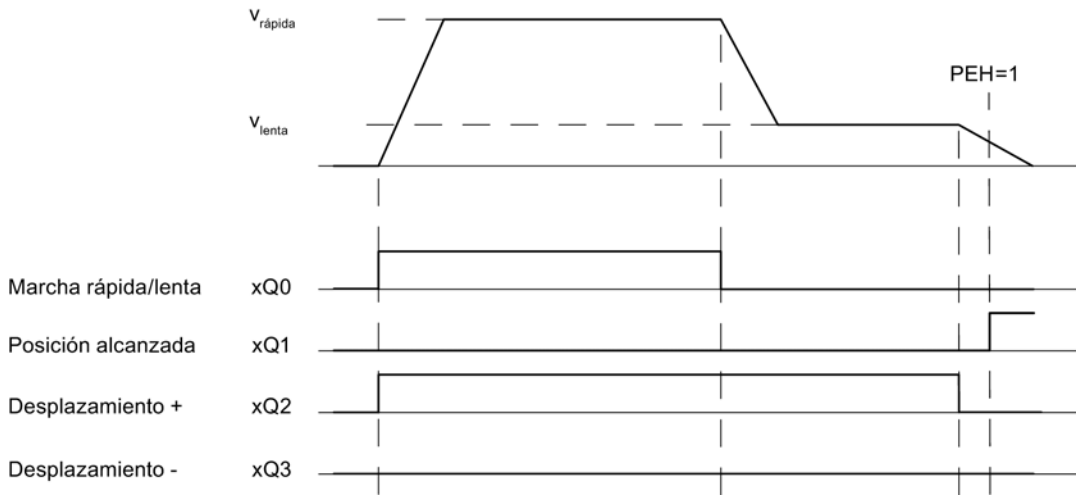
Modo de control 2

Modo de control 2

En el modo de control 2 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida/lenta
- xQ1: Posición alcanzada
- xQ2: Desplazamiento +
- xQ3: Desplazamiento -

La figura siguiente muestra la función de las salidas digitales en el modo de control 2.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	2 = Modo de control 2

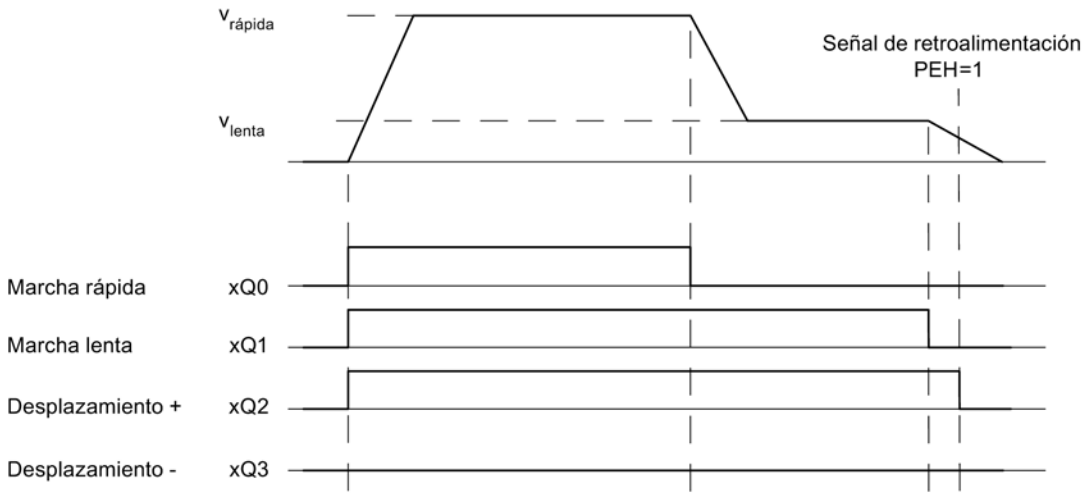
Modo de control 3

Modo de control 3

En el modo de control 3 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida
- xQ1: Marcha lenta
- xQ2: Desplazamiento +
- xQ3: Desplazamiento -

La figura siguiente muestra la función de las salidas digitales en el modo de control 3.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	3 = Modo de control 3

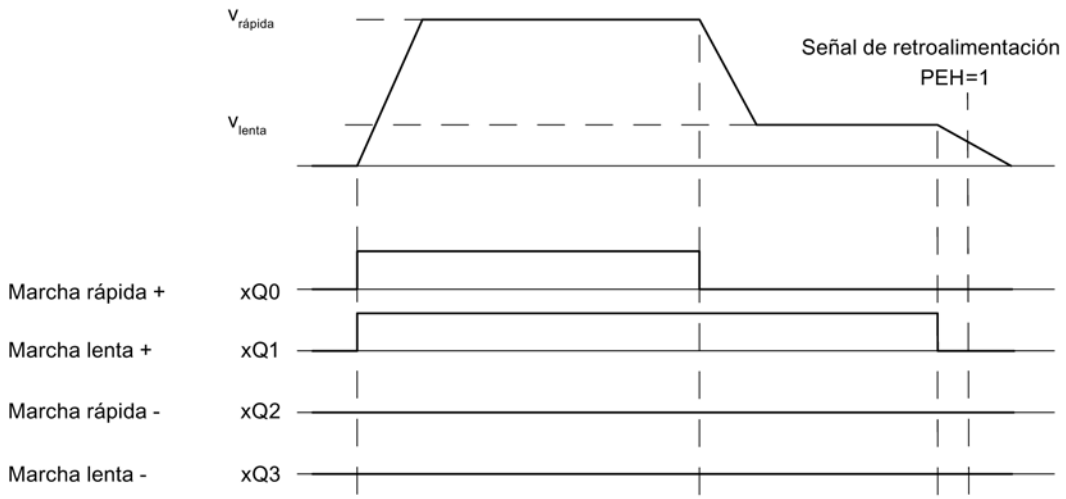
Modo de control 4

Modo de control 4

En el modo de control 4 se aplican los ajustes siguientes (x representa el canal n):

- xQ0: Marcha rápida +
- xQ1: Marcha lenta +
- xQ2: Marcha rápida -
- xQ3: Marcha lenta -

La figura siguiente muestra la función de las salidas digitales en el modo de control 4.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
92.0	CTRL_TYPE	DINT	L#1	4 = Modo de control 4

Definir distancia de cambio de velocidad

Distancia de cambio de velocidad

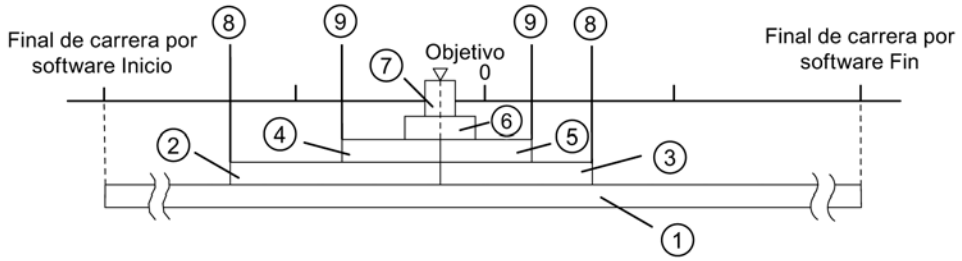
La "Distancia de cambio de velocidad" define la distancia con respecto al punto de destino en el que el accionamiento cambia la velocidad, de velocidad rápida a velocidad lenta.

Los valores predefinidos se aplican a todos los objetivos a los que se aproxima el módulo de posicionamiento. Una excepción está constituida por el modo incremental con el incremento 255, para la que se define una distancia de cambio de velocidad especial.

Reglas para la distancia de cambio de velocidad

- La distancia de cambio de velocidad positiva y la distancia de cambio de velocidad negativa no tienen por qué estar dispuestas de forma simétrica en torno al destino. Los valores pueden ser muy distintos.
- La distancia de cambio de velocidad debe ser mayor que la distancia de desconexión.
- La distancia de cambio de velocidad se encuentra dentro de la zona de trabajo.
- La distancia de cambio de velocidad debe ser menor que el final del eje rotativo.
- Entre el punto de cambio de velocidad y el punto de desconexión se debe seleccionar una distancia tan grande que permita cambiar de marcha rápida a marcha lenta.
- El punto de cambio de velocidad y el punto de desconexión se encuentran a una distancia temporal de al menos 8 ms entre sí.

La figura siguiente muestra la ubicación de las áreas de cambio de velocidad en torno al destino.



① Zona de trabajo
② Distancia de cambio de velocidad positiva/negativa
③
④ Distancia de desconexión positiva/negativa
⑤
⑥ Zona de parada
⑦ Zona de destino
⑧ Punto de cambio de velocidad
⑨ Punto de desconexión

Si desea obtener más información sobre la disposición de las áreas, consulte "Zonas y puntos de cambio de velocidad del módulo de posicionamiento".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CHGDIF_P	DINT	L#5000	**Distancia de cambio de velocidad positiva**
104.0	CHGDIF_M	DINT	L#5000	**Distancia de cambio de velocidad negativa** Rango de valores admisible: • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir distancia de desconexión

Distancia de desconexión

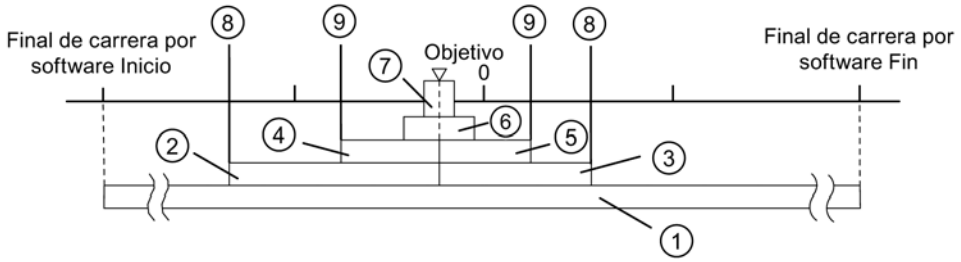
La "Distancia de desconexión" define la distancia hasta el punto de destino en el que se desconecta el accionamiento (la marcha lenta). Las funciones de vigilancia retoma el módulo de posicionamiento a partir de este punto.

Los valores predefinidos se aplican a todos los objetivos a los que se aproxima el módulo de posicionamiento. Una excepción está constituida por el modo incremental con el incremento 255, para la que se define una distancia de desconexión especial.

Reglas para la distancia de desconexión

- La distancia de desconexión positiva y la distancia de desconexión negativa no tienen por qué estar dispuestas de forma simétrica en torno al destino. Los valores pueden ser muy distintos.
- La distancia de desconexión debe ser mayor que la mitad de la zona de destino.
- Entre el punto de desconexión y el destino se debe seleccionar una distancia que permita que el accionamiento se detenga dentro de la zona de destino.
- El punto de desconexión y el inicio de la zona de destino se encuentran a una distancia temporal de al menos 8 ms entre sí.

La figura siguiente muestra la ubicación de las zonas de cambio de velocidad en torno al destino.



①	Zona de trabajo	⑦	Zona de destino
② ③	Distancia de cambio de velocidad positiva/negativa	⑧	Punto de cambio de velocidad
④ ⑤	Distancia de desconexión positiva/negativa	⑨	Punto de desconexión
⑥	Zona de parada		

Si desea obtener más información sobre la disposición de las áreas, consulte "Zonas y puntos de conexión del módulo de posicionamiento".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
108.0	CUTDIF_P	DINT	L#2000	**Distancia de desconexión positiva**
112.0	CUTDIF_M	DINT	L#2000	**Distancia de desconexión negativa** Rango de valores admisible: • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir la aproximación al destino

Definir zona de destino

Zona de destino

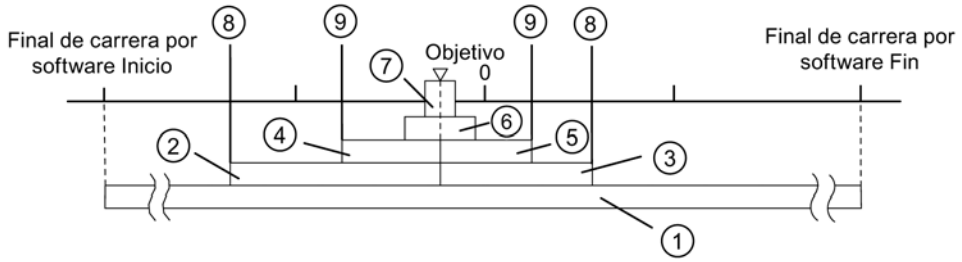
El módulo de posicionamiento crea un área simétrica alrededor de cada destino: la zona de destino. Dentro de esta área es preciso que el valor real se encuentre en una aproximación al destino.

Una especificación de valor "0 "desconecta la tolerancia en la aproximación al destino. La velocidad al alcanzar el destino debe ser entonces inferior a la velocidad de parada (Página 11052) parametrizada.

Reglas para la zona de destino

- La zona de destino está situada de forma simétrica en torno al destino.
- Una especificación de valor "0 "desconecta la vigilancia de la zona de destino.

La figura siguiente muestra la ubicación de las zonas en torno al destino.



①	Zona de trabajo	⑦	Zona de destino
② ③	Distancia de cambio de velocidad positiva/negativa	⑧	Punto de cambio de velocidad
④ ⑤	Distancia de desconexión positiva/negativa	⑨	Punto de desconexión
⑥	Zona de parada		

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Zonas y puntos de conexión del módulo de posicionamiento", así como "Fin de un proceso de posicionamiento (Página 10999)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
76.0	TRG_RANGE	DINT	L#1000	**Zona de destino** Rango de valores admisible: • 0 = sin vigilancia • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir tiempo de vigilancia

Tiempo de vigilancia

Durante el tiempo de vigilancia el módulo vigila:

- el movimiento del eje hasta el punto de desconexión.
  El tiempo de vigilancia comienza con el inicio de un posicionamiento y se redispara cada vez que se modifica el valor real en el sentido de la marcha.
  Dentro del tiempo de vigilancia, el valor real debe haberse modificado al menos un incremento (la parte del trayecto de la resolución).
- la aproximación al destino.
  El tiempo de vigilancia se redispara por última vez cuando se alcanza el punto de desconexión.
  Dentro del tiempo de vigilancia debe haberse alcanzado la zona de destino.

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Final de un proceso de posicionamiento (Página 10999)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
80.0	MON_TIME	DINT	L#2000	**Tiempo de vigilancia** Rango de valores admisible: • 0 = sin vigilancia • De 1 a 100.000 ms
Tiempo de vigilancia real Para el tiempo vigilancia es posible predefinir todos los valores a partir del área de valores. El módulo redondea el tiempo predefinido a un múltiplo de 8 ms (ciclo de módulo).				

Definir zona de parada

Zona de parada

La zona de parada está situada de forma simétrica en torno al destino correspondiente. En esta área se vigila si el accionamiento se detiene en una posición de destino alcanzada o si se aleja de ella. La vigilancia comienza después de que el módulo de posicionamiento ha notificado "Posición alcanzada" (PEH).

Si la zona de parada se abandona sin una petición de marcha válida, el módulo de posicionamiento notifica un error.

Reglas para la zona de parada

- La zona de parada está situada de forma simétrica en torno al destino correspondiente: (Destino – ½ × zona de parada) ≤ destino ≤ (destino + ½ × zona de parada)
- La zona de parada debe ser mayor o igual que la zona de destino.
- Los valores de coordenadas para la zona de parada se encuentran en la zona de trabajo.
- Con el valor "0" se desconecta la vigilancia de parada.

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Zonas y puntos de conexión del módulo de posicionamiento", así como "Fin de un proceso de posicionamiento (Página 10999)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
84.0	ZSPEED_R	DINT	L#1000	**Zona de parada** Rango de valores admisible: • 0 = sin vigilancia • 1 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir velocidad de parada

Velocidad de parada

La velocidad de parada sirve como velocidad de referencia. Si la velocidad real alcanza la velocidad de parada durante un posicionamiento y el accionamiento se encuentra en la zona de destino, se notifica el aviso "Posición alcanzada" (PEH). El proceso de posicionamiento queda correctamente finalizado.

La precisión de posicionamiento aumenta cuando el valor de la velocidad de parada se reduce. Con el valor "0" se desconecta la vigilancia de parada.

Si desea obtener más información sobre el tema de la aproximación al destino, consulte el apartado "Final de un proceso de posicionamiento (Página 10999)".

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
88.0	ZSPEED_L	DINT	L#30000	**Velocidad de parada** Rango de valores admisible: • 0 = sin vigilancia • 1 µm/min a 100 000 µm/min

Eje canal n

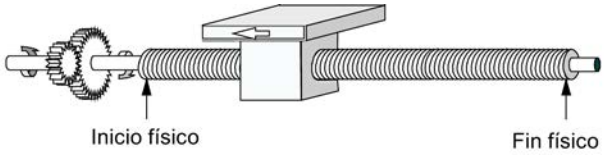
Definir el tipo de eje

Seleccionar tipo de eje

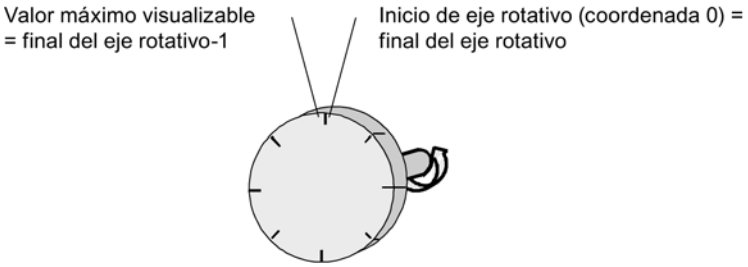
Tipos de eje

El módulo de posicionamiento puede controlar los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento físico limitado.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
12.0	AXIS_TYPE	DINT	L#0	**Tipo de eje** Valores permitidos: • 0 = Eje lineal • 1 = Eje rotativo

Definir final de carrera por software

Final de carrera por software

Los finales de carrera por software existen sólo para el eje lineal.

El área limitada por el final de carrera por software es la zona de trabajo.

El inicio del final de carrera por software (FCSF) debe ser siempre menor que el fin del final de carrera por software (SEE).

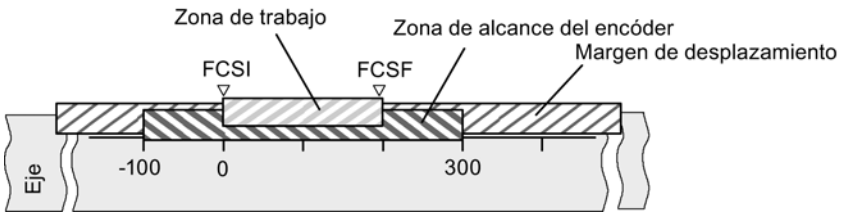
Los finales de carrera por software se vigilan después de que el eje está sincronizado.

- Si se utiliza un encóder incremental:
  La sincronización del eje se alcanza mediante la búsqueda del punto de referencia o la definición del punto de referencia.
- Si se utiliza un encóder absoluto (SSI):
  El eje está sincronizado después de que el módulo ha recibido un telegrama completo y sin errores para el canal correspondiente.
  El encóder absoluto tiene que abarcar por lo menos la zona de trabajo, incluido el final de carrera por software.

Relación entre la zona de trabajo, el área del encóder y la zona de desplazamiento

- La zona de trabajo es el área que el usuario define para sus tareas mediante los finales de carrera por software.
- El área del encóder es el área que el encóder cubre claramente. En el caso de un eje lineal del módulo, el área del encóder se define de forma simétrica por encima de la zona de trabajo. Es decir, el módulo desplaza el área del encóder de tal modo que las distancias entre los finales de carrera por software y los finales del área del encóder son iguales en cada caso (véase la figura inferior).
- La zona de desplazamiento es el rango de valores que puede procesar el módulo de posicionamiento. La zona de desplazamiento depende de la resolución.

La figura siguiente muestra la relación entre los finales de carrera por software, la zona de trabajo, el área del encóder y la zona de desplazamiento.



Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
64.0	SSW_STRT	DINT	L#-100000000	**Inicio del final de carrera por software**
68.0	SSW_END	DINT	L#100000000	**Fin del final de carrera por software** Rango de valores admisible: • -1 000 000 000 µm a 1 000 000 000 µm a una resolución ≥ 1 µm/impulso • -100.000.000 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir final del eje rotativo

Final del eje rotativo

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar. Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo (0).

El valor más grande que se muestra, en el caso de un eje rotativo, tiene el valor:

Final del eje rotativo [µm] - Resolución [µm / impulso] × 1 [impulso]

Ejemplo: Final del eje rotativo 1000 mm, resolución 1000 µm/impulso

La visualización pasa:

- con un sentido de giro positivo, de 999 mm a 0 mm
- con un sentido de giro negativo, de 0 mm a 999 mm

Eje rotativo con encóders absolutos

En el caso de un eje rotativo con un encóder absoluto, el área del eje rotativo (0 hasta el final del eje rotativo) tiene que coincidir exactamente con el área de sensor del encóder absoluto:
Fin del eje rotativo [µm] = Número de vuelltas del encóder × Recorrido/vuelta del encóder [µm]

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
16.0	ENDROTAX	DINT	L#100000	**Final del eje rotativo** Rango de valores admisible: • 1 µm a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • 1 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir la búsqueda del punto de referencia

Seleccionar tipo de búsqueda del punto de referencia

Tipo de búsqueda del punto de referencia

Con la función "Tipo de búsqueda del punto de referencia" se determinan las condiciones para la sincronización del eje:

- El primer dato define el sentido de arranque en el que comienza la búsqueda del punto de referencia.
- El segundo dato define la posición de la marca cero que conduce a la sincronización, en referencia con el conmutador de punto de referencia.

Si desea obtener información detallada al respecto consulte Modo de operación "Búsqueda del punto de referencia" (Página 11008).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
52.0	REFPT_TYPE	DINT	L#0	**Tipo de búsqueda del punto de referencia** Valores permitidos: • 0 = positivo, conmutador del punto de referencia en el sentido + • 1 = positivo, conmutador del punto de referencia en el sentido - • 2 = negativo, conmutador del punto de referencia en el sentido + • 3 = negativo, conmutador del punto de referencia en el sentido -

Definir coordenada del punto de referencia

Coordenada del punto de referencia

- **Encóders inc.:**
  La coordenada del punto de referencia es necesaria para el modo de operación "Búsqueda del punto de referencia". Si el eje no está sincronizado después de escribir y activar los datos máquina, el valor real se establece al valor de la coordenada del punto de referencia.
- **Encóder absoluto (SSI)**
  La coordenada del punto de referencia es necesaria para el ajuste mecánico del encóder. Consulte al respecto también la descripción del reajuste del encóder absoluto (Página 11057), en la que se explica la interacción de dicho ajuste con el resto de datos.

El valor de la coordenada del punto de referencia debe encontrarse dentro de la zona de trabajo:

- **Eje lineal**
  Incluido el final de carrera por software.
- **Eje rotativo**
  Mayor o igual que 0 y menor que el valor "Final del eje rotativo" (0 ≤ Coordenada del punto de referencia < Final del eje rotativo).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1 000.000.000 µm a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • -100.000.000 µm a 100.000.000 µm a una resolución < 1 µm/impulso

Definir el reajuste del encóder absoluto

Calcular reajuste del encóder absoluto

Reajuste del encóder absoluto

El ajuste del encóder absoluto sólo es necesario para un encóder absoluto (SSI).

Con el reajuste del encóder absoluto y las coordenadas del punto de referencia el rango de valores del sensor se reproduce claramente en el sistema de coordenadas del eje.

Encontrará más información al respecto en "Procedimiento de reajuste del encóder absoluto (Página 11058)", así como en un ejemplo (Página 11059).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
44.0	REFPT	DINT	L#0	**Coordenada del punto de referencia** Rango de valores admisible: • -1.000.000.000 µm a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • -100.000.000 µm a 100.000.000 µm a una resolución < 1 µm/impulso
48.0	ENC_ADJ	DINT	L#0	**Reajuste del encóder absoluto** Rango de valores admisible: • 0 a (2^{25} - 1)

Procedimiento para un reajuste del encóder absoluto

Determinar el reajuste correcto del encóder absoluto

Tras la primera parametrización hay que llevar a cabo otros pasos para que se establezca una relación correcta entre el encóder y el sistema de coordenadas. El procedimiento se representa utilizando el cuadro de diálogo de parametrización.

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Fin del final de carrera por software".
2. Llame la petición "Definir punto de referencia (Página 11025)" con la coordenada del punto definido en 1.
   El módulo de posicionamiento determina entonces un valor de encóder para la coordenada del punto de referencia registrada en el eje en el DB de canal (REFPT): el reajuste del encóder absoluto. Este valor se puede leer con la petición "Leer datos de encóder (Página 11034)".
3. Registre el valor leído en el campo "Reajuste del encóder absoluto".
4. Guarde la parametrización en el DB de parámetros correspondiente con la función de exportación.
5. Cargue los datos de HWCN en la CPU.
6. Para activar los datos, rearranque la CPU.

Nota

Esta sincronización se realiza una vez en la puesta en marcha. Tras una parametrización, el módulo de posicionamiento queda sincronizado durante el arranque en cuanto se recibe un telegrama sin errores del encóder.

Alternativa: Reajuste mecánico de un encóder

También se puede obtener una relación correcta entre el sistema de coordenadas y el encóder del modo siguiente:

1. Desplace el eje a un punto definido y reproducible que físicamente sea único.
   Puede ser, por ejemplo, el "Inicio del final de carrera por software".
2. Introduzca este valor de coordenadas en los datos máquina a modo de "coordenada del punto de referencia".
3. Lea el valor del encóder que se muestra en esta posición con la petición "Leer datos del encóder (Página 11034)".
4. Introduzca este valor como reajuste del encóder absoluto en los datos máquina.

Tras la parametrización se mostrará siempre un valor real correcto.

En lugar del paso 3 y 4 también es posible resetear a cero el codificador a través de la opción "Reset", si está disponible. A continuación, introduzca el valor "0" como reajuste del encóder absoluto en los datos máquina.

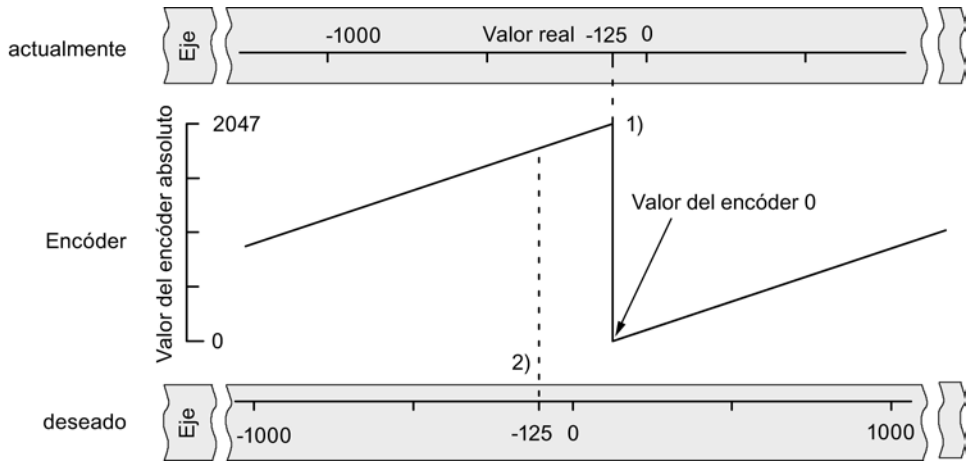
Ejemplo para un reajuste del encóder absoluto

Ejemplo para un reajuste del encóder absoluto

Para este ejemplo se parte de las premisas siguientes:

- Coordenada del punto de referencia = -125 mm
- Área de trabajo de SSW_STRT = -1.000 mm a SSW_END = 1.000 mm
- Reajuste del encóder absoluto = 0
- Área del encóder = 2.048 incrementos con una resolución de 1 mm/impulso
- El encóder absoluto utilizado no permite un ajuste mecánico exacto y tampoco ofrece la posibilidad de definir el valor del encóder.

La figura siguiente muestra un ejemplo de reajuste del encóder absoluto.



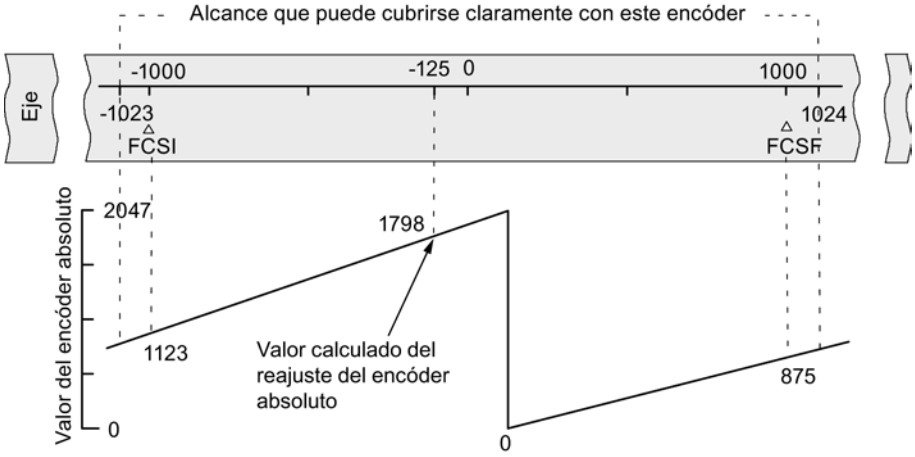
1) Asignación del sistema de coordenadas a los valores del sensor con el reajuste de encóder absoluto definido. El valor del encóder 0 equivale al valor real -125.

2) Asignación deseada del sistema de coordenadas con respecto al encóder. En esta posición, la coordenada tiene que ser -125.

Resultado tras "Definir el punto de referencia"

Tras "Definir el punto de referencia", la relación es la siguiente:

La coordenada del punto de referencia en el eje (-125) se asigna al valor del encóder calculado (1798) a partir del reajuste del encóder absoluto. La figura siguiente muestra esta asignación.



El encóder devuelve 2.048 valores unívocos. La zona de trabajo está definida por el final de carrera por software. Sin embargo, debido a la resolución seleccionada de 1 mm por impulso, el encóder puede abarcar una zona de trabajo mayor que la prevista por el final de carrera por software.

Con la resolución definida, la zona de trabajo está ya cubierta con 2.001 valores. Por ello en el ejemplo quedan 47 impulsos "sobrantes" dispuestos simétricamente alrededor de la zona de trabajo.

Seleccionar velocidad de inicio

Velocidad de inicio en la búsqueda del punto de referencia

Con este dado se selecciona la velocidad para el inicio de una búsqueda del punto de referencia.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
99.0	REFPT_SPD	BOOL	TRUE	**Velocidad de inicio en la búsqueda del punto de referencia** Valores permitidos: • 0 = Marcha rápida • 1 = Marcha lenta

Definir la distancia mínima entre bordes

Distancia mínima entre bordes

Con el dato máquina "Distancia mínima entre bordes" se define un rango tras la detección del inicio de la medición en una captura de bordes. Si el final del proceso de medición se encuentra dentro de este rango, la medición se desestima.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
4.0	EDGEDIST	DINT	L#0	**Distancia mínima entre bordes** Rango de valores admisible: • 0 a 1.000.000.000 µm a una resolución ≥ 1 µm/impulso • 0 a 100.000.000 µm a una resolución < 1 µm/impulso

Encóder del canal n

Seleccionar tipo de señal

Tipo de señal

El encóder da información de recorrido al módulo, el cual la evalúa y la convierte en un valor real con la resolución.

Es posible seleccionar un tipo de señal de uno de los encóders siguientes:

- Encóder incremental de 5 V
  Seleccione el tipo de señal "5 V Señal diferencial" si utiliza un encóder incremental de 5 V con señales diferenciales A, /A, B, /B y N, /N.
- Encóder incremental de 24 V
  Seleccione el tipo de señal "24 V pista A+B desfasado" si desea utilizar un encóder incremental de 24 V con las señales A*, B* y N*.
- Encóder absoluto (SSI)
  Seleccione el tipo de señal "SSI absoluto" si desea utilizar un encóder absoluto con una interfaz en serie.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
20.0	ENC_TYPE	DINT	L#1	**Tipo de señal** Valores permitidos: • 1 = 5 V Señal diferencial • 2 = 24 V pista A+B desfasado • 3 = SSI longitud del telegrama 13 bits [1)] • 4 = SSI longitud del telegrama 25 bits [1)] • 5 = SSI longitud del telegrama 14 bits [1)] • 6 = SSI longitud del telegrama 15 bits [1)] • 7 = SSI longitud del telegrama 16 bits [1)] • : • : • 14 = SSI longitud del telegrama 23 bits [1)] • 15 = SSI longitud del telegrama 24 bits [1)]

[1)] La selección de la longitud del telegrama se realiza en el cuadro de diálogo "SSI".

Seleccionar sentido de contaje

Sentido de contaje

Con el dato máquina "Sentido de contaje" se adapta el sentido de la captura del recorrido al sentido del movimiento del eje. Para ello, tenga en cuenta todos los sentidos de giro de los elementos de transmisión (como p. ej., acoplamientos y engranajes).

- normal = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real ascendentes
- invertida = los impulsos de contaje adelantes (encóder incremental) o valores del sensor (encóder absoluto) equivalen a los valores de posición real descendentes

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
59.0	CNT_DIR	BOOL	FALSE	**Sentido de contaje** Valores permitidos: • 0 = normal • 1 = invertida

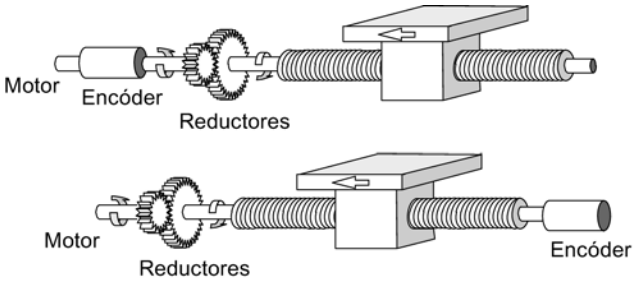
Definir resolución

Definir recorrido por vuelta del encóder

Recorrido por vuelta del encóder

Con el dato máquina "Recorrido por vuelta del encóder" se indica al módulo de posicionamiento qué recorrido realiza el sistema de accionamiento en cada vuelta.

El valor "Recorrido por vuelta del encóder" depende de la estructura del eje y del modo en que está colocado el encóder. Debe tener en cuenta todos los elementos de transmisión, como acoplamientos o engranajes. La figura siguiente muestra dos disposiciones de encóder distintas a modo de ejemplo.



La relación entre los datos máquina "Recorrido por vuelta del encóder" e "Incrementos por vuelta del encóder" se describe en el apartado Resolución (Página 11065).

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
24.0	DISP_REV	DINT	L#80000	**Recorrido por vuelta del encóder** Rango de valores admisible: • 1 µm a 1.000.000.000 µm

Definir incrementos por vuelta del encóder

Incrementos por vuelta

El dato máquina "Incremento por vuelta del encóder" indica la cantidad de incrementos que proporciona un encóder en cada vuelta. A partir de este valor y del dato máquina "Recorrido por vuelta del encóder", el módulo de posicionamiento calcula la resolución (Página 11065).

Encóder incremental

Como entrada es posible indicar cualquier valor dentro del rango de valores. El módulo evalúa los incrementos en múltiplos de 4.

Encóder absoluto

Para los límites debe diferenciarse entre los distintos modelos de encóder. Como entrada se permiten sólo valores en potencias de 2.

- Encóder monovuelta (número de vueltas = 1) con longitud del telegrama de 13 bits:
  - valor mínimo = 4
  - valor máximo = 8192
- Encóder multivuelta (número de vueltas > 1) con longitud del telegrama de 25 bits:
  - valor mínimo = 4
  - valor máximo = 8192
- Encóder monovuelta (número de vueltas = 1) con longitud del telegrama de 25 bits:
  - valor mínimo = 4
  - valor máximo = 2^{25}

Las medidas lineales se parametrizan como encóder multivuelta tal como se indica a continuación:

- Incrementos por vuelta del encóder = 8192
- Número de vueltas x 8192 ≥ Número de pasos de la medida lineal

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
32.0	INC_REV	DINT	L#500	**Incrementos por vuelta del encóder** Rango de valores admisible: • 1 a 2^{25}

Resolución

Resolución

La resolución indica el desplazamiento que corresponde s un impulso. Es una medida de la exactitud del posicionamiento y determina también la zona de desplazamiento máximo posible del módulo de posicionamiento.

Para calcular la resolución (RES) se procede tal como se indica en la tabla siguiente:

	Encóder incremental	Encóder absoluto
Valores de entrada	• Recorrido por vuelta del encóder • Incrementos por vuelta del encóder: – Evaluación cuádruple – 1 Incremento = 4 Impulsos	• Recorrido por vuelta del encóder • Incrementos por vuelta del encóder: – 1 Incremento = 1 Impulso
Cálculo	$RES = \frac{\frac{\text{Trayecto}}{\text{Vuelta del encóder}}}{\frac{\text{Impulsos}}{\text{Vuelta del encóder}}}$	

Nota

Todos los datos de posición se redondean al múltiplo entero de la resolución. De este modo es posible diferenciar entre los valores introducidos y los utilizados.

Rango de valores de la resolución

El sistema de medida seleccionado determina el margen de valores de la resolución.

Sistema de medición	Datos en ...	Rango de valores de la resolución
mm	10^{-3} mm	$0{,}1 \times 10^{-3}$ … 1000×10^{-3} mm/impulso
pulgadas	10^{-4} pulgadas	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} pulgada/impulso
grados	10^{-4} grados 10^{-3} grados 10^{-2} grados	$0{,}1 \times 10^{-4}$ … 1000×10^{-4} grados/impulso $0{,}1 \times 10^{-3}$ … 1000×10^{-3} grados/impulso $0{,}1 \times 10^{-2}$ … 1000×10^{-2} grados/impulso

Ejemplo

- Un encóder incremental tiene los datos siguientes:
  - Incrementos por vuelta del encóder: 5000
  - Recorrido por vuelta del encóder: 1.000 mm
  - 1 Incremento = 4 Impulsos

  De ello resulta la resolución (evaluación cuádruple):

$$\text{Resolución} = \frac{1000\ \text{mm}}{5.000\ \text{incrementos}} = 0{,}2000\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2000\ \frac{\text{mm}}{4\ \text{impulsos}}$$

$$= 0{,}0500\ \frac{\text{mm}}{\text{Impulso}}$$

- Un encóder SSI tiene los datos siguientes:
  - Incrementos por vuelta del encóder: 4096
  - Recorrido por vuelta del encóder: 1.000 mm
  - 1 Incremento = 1 Impulso

  De ello resulta la resolución:

$$\text{Resolución} = \frac{1000\ \text{mm}}{4096\ \text{incrementos}} = 0{,}2441\ \frac{\text{mm}}{\text{Incremento}} = 0{,}2441\ \frac{\text{mm}}{\text{Impulso}}$$

Dependencia entre la zona de desplazamiento y la resolución

La zona de desplazamiento está limitada por la representación numérica en el módulo de posicionamiento. Esta representación numérica varía en función de la resolución. Por ello, al definir valores hay que procurar mantenerse siempre dentro de los límites admitidos.

La zona de desplazamiento máxima se muestra en la tabla siguiente:

Resolución (RES) dentro del margen	Zona de desplazamiento máxima
0,1 µm/impulso ≤ RES < 1 µm/impulso	-10^8 µm a 10^8 µm (-100 m a +100 m)
1 µm/impulso ≤ RES ≤ 1000 µm/impulso	-10^9 µm a 10^9 µm (-1000 m a +1000 m)

Datos para el encóder absoluto

Número de vueltas del encóder

Número de vueltas del encóder

El dato máquina "Número de vueltas del encóder" sólo existe en encóders absolutos. Con él se indica el número de vueltas posibles con este encóder.

El **total de pasos del encóder** no es un dato máquina. Se calcula del modo siguiente:

Total de pasos = Incremento por vuelta x Número de vueltas

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.0	NO_REV	DINT	L#1	**Número de vueltas del encóder** Rango de valores admisible: • 1 (encóder monovuelta) • 2 a 4096 en potencias de 2 (encóder multivuelta)

Velocidad de transferencia

Velocidad de transferencia

Con el dato máquina "Velocidad de transferencia" se define la velocidad de transferencia de datos del encóder SSI al módulo de posicionamiento.

La velocidad de transferencia máxima depende de la longitud del cable:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 63 m → 500 kHz
- 20 m → 1000 kHz

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
40.0	BAUDRATE	DINT	L#0	**Velocidad de transferencia** Valores permitidos: • 0 = 125 kHz • 1 = 250 kHz • 2 = 500 kHz • 3 = 1000 kHz

Longitud de telegrama

Longitud de telegrama

Con la "Longitud de telegrama" se define el marco de impulsos que emite el módulo de posicionamiento.

Se permite una longitud de telegrama de 13 bits a 25 bits.

Datos utilizados en el DB de parámetros

La longitud de telegrama seleccionada aquí se dispone en el DB de parámetro con el dato ENC_TYPE (Página 11057).

Paridad

Paridad

Con la "Paridad" se define si para el encóder absoluto no debe transferirse ninguna paridad o si debe transferirse una paridad par o impar.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
28.0	PARITY	DINT	L#0	**Paridad** Valores permitidos: • 0 = ninguno • 1 = paridad par (even) • 2 = paridad impar (odd)

Seleccionar vigilancia

Vigilancia

Rotura de hilo

Con la activación de la vigilancia, el módulo de posicionamiento vigila todos los conductos en un encóder incremental de 5 V y un encóder absoluto. La vigilancia detecta:

- Rotura de hilo
- Cortocircuito de los distintos cables
- Distancia entre flancos de los impulsos de contaje (también en un encóder incremental de 24 V)

Para la monitorización en un encóder incremental de 24 V es preciso ajustar un tiempo de monitorización MON_TIME (Página 11047) > 0.

En encóders incrementales de 5 V sin marca cero es preciso desactivar la monitorización de rotura de hilo o bien interconectar las señales N y /N de forma externa.

Error de telegrama

El módulo monitoriza el telegrama de un encóder absoluto (SSI) para:

- ver si hay errores del bit de inicio y del bit de parada

La monitorización de errores de telegrama en los encóders absolutos (SSI) no puede desactivarse.

Impulsos erróneos (encóder incremental)

Entre dos marcas cero consecutivas, un encóder incremental siempre tiene que proporcionar el mismo número de incrementos.

El módulo de posicionamiento comprueba si la marca cero de un encóder incremental aparece en el estado adecuado del encóder.

En el caso de encóders sin marca cero es preciso desactivar la monitorización de los impulsos de error.

Igualmente es preciso desactivar la monitorización de rotura de hilo o bien interconectar las entradas de marca cero N y /N de forma externa.

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
				Vigilancias:
63.0	MON_WIRE	BOOL	TRUE	1 = Rotura de hilo
63.1	MON_FRAME	BOOL	TRUE	1 = Error de telegrama (tiene que ser siempre 1)
63.2	MON_PULSE	BOOL	TRUE	1 = Impulsos de error

Incremento del canal n

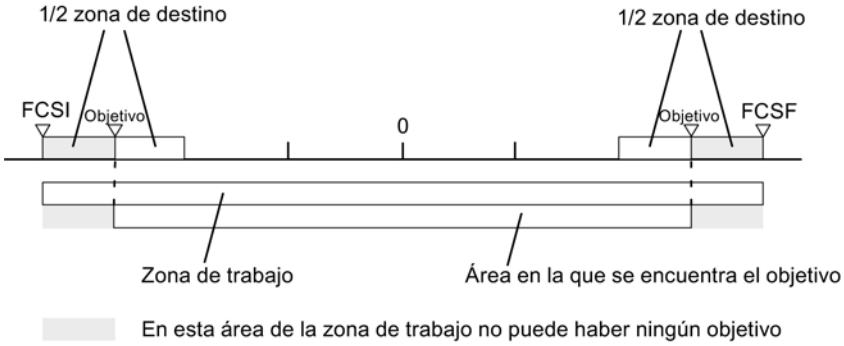
Definir incrementos

Definición

Los incrementos son destinos especificados que pueden ser controlados por el módulo de posicionamiento en el modo de operación "Modo incremental relativo" o "Modo incremental absoluto".

Área posible para los incrementos

El destino al que tiene que realizarse una aproximación tiene que encontrarse al menos a media zona de destino delante del final de carrera por software correspondiente. La figura siguiente muestra los límites de una especificación del incremento.



Número de incremento 1 a 100

Número de incremento 1 a 100

Tiene la posibilidad de insertar un máximo de 100 incrementos en una tabla, que son válidas tanto el modo de operación "Modo incremental relativo" como para el "Modo incremental absoluto".

Tenga en cuenta que el módulo de posicionamiento para el "Modo incremental relativo" no soporta valores negativos. Los valores son interpretados por el módulo de posicionamiento en función del sentido de movimiento como diferencia positiva o negativa.

Nota

La entrada se realiza en la unidad del sistema de medida ajustado. Tenga en cuenta en este caso sobre todo las posiciones decimales.

Ejemplo de números:

- Incremento: 800 mm
- Sistema de medida: 10^{-3} mm
- Entrada en el DB de parámetros: 800000

Recomendación: En la tabla de incrementos defina áreas separadas para incrementos relativos y absolutos.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
35.4	TRGL1WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 1 (incremento 1 a 50)
35.5	TRGL2WR_EN	BOOL	FALSE	1 = Escribir tabla de incrementos 2 (incremento 1 a 100)
36.6	TRGL1RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 1 (incremento 1 a 50)
36.7	TRGL2RD_EN	BOOL	FALSE	1 = Leer tabla de incrementos 2 (incremento 1 a 100)

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios	
120.0	TRGL1.TRG[1]	DINT	L#0	Número de incremento 1	Tabla de incrementos 1
. . .	. . .	. .	. . .	. . .	
316.0	TRGL1.TRG[50]	DINT	L#0	Número de incremento 50	
320.0	TRGL2.TRG[51]	DINT	L#0	Número de incremento 51	Tabla de incrementos 2

Dirección	Nombre	Tipo	valor inicial	Comentarios	
. . .	. . .	. . .	. . .	. . .	
516.0	TRGL2.TRG[100]	DINT	L#0	Número de incremento 100	

Número de incremento 252

Número de incremento 252

El número de incremento 252 ofrece la posibilidad de mover primero el accionamiento sin destino (marcha sin fin) y, si se produce un evento externo en la entrada digital xI3, permite realizar un posicionamiento en el destino parametrizado (incremento).

Para las distancias de cambio de velocidad y de desconexión, las entradas del DB de parámetros tienen validez para este incremento.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.2	TRG252_254_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 252 ó 254
96.0	TRG252_254	DINT	L#0	Incremento para número de incremento 252 ó 254

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CHGDIF_P	DINT	L#5000	Distancia de cambio de velocidad positiva
104.0	CHGDIF_M	DINT	L#5000	Distancia de cambio de velocidad negativa
108.0	CUTDIF_P	DINT	L#2000	Distancia de desconexión positiva
112.0	CUTDIF_M	DINT	L#2000	Distancia de desconexión negativa

Número de incremento 254

Número de incremento 254

Independientemente de la tabla de incrementos, es posible utilizar el número de incremento 254 como otra especificación del recorrido. Para las distancias de cambio de velocidad y de desconexión, las entradas del DB de parámetros tienen validez para este incremento.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.2	TRG252_254_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 252 ó 254
96.0	TRG252_254	DINT	L#0	Incremento para número de incremento 252 ó 254

Datos utilizados en el DB de parámetros

Dirección	Nombre	Tipo	valor inicial	Comentarios
100.0	CHGDIF_P	DINT	L#5000	Distancia de cambio de velocidad positiva
104.0	CHGDIF_M	DINT	L#5000	Distancia de cambio de velocidad negativa
108.0	CUTDIF_P	DINT	L#2000	Distancia de desconexión positiva
112.0	CUTDIF_M	DINT	L#2000	Distancia de desconexión negativa

Número de incremento 255

Número de incremento 255

Con el número de incremento 255 dispone de otra especificación del recorrido.

Las distancias de desconexión y las distancias de cambio de velocidad se transfieren junto con el incremento. Al contrario de lo que sucede en otros incrementos, el incremento 255 utiliza los valores fijados en el DB del canal para la distancia de desconexión y de la distancia de cambio de velocidad. Las entradas de los datos máquina no tienen validez para este incremento.

Datos utilizados en el DB de canal

Dirección	Nombre	Tipo	valor inicial	Comentarios
36.3	TRG255_EN	BOOL	FALSE	1 = Escribir incremento para número de incremento 255
100.0	TRG255	DINT	L#0	Incremento para número de incremento 255
104.0	CHGDIF255	DINT	L#0	Distancias de cambio de velocidad para número de incremento 255
108.0	CUTDIF255	DINT	L#0	Distancias de desconexión para número de incremento 255

Interfaz de control y realimentación del FM 451

Aplicación

En aplicaciones especiales o en un nivel de alarma puede que se necesite un acceso especialmente rápido a señales de realimentación y de control. A estos datos se accede directamente a través de las áreas de entradas y salidas del módulo.

Para coordinar el arranque cada vez que arranca el módulo (p. ej. tras insertar el módulo o después de una transición STOP-RUN de la CPU), la instrucción ABS_CTRL (Página 7420) debe llamarse hasta que se muestre el final del arranque a través de RETVAL = 0. A continuación la instrucción ABS_CTRL ya no puede utilizarse.

Nota

No es posible utilizar la instrucción ABS_CTRL en combinación con un acceso de escritura.

Leer señales de realimentación mediante acceso directo

Las direcciones de byte se indican de forma relativa respecto a la dirección de inicio de las salidas del canal correspondiente. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7428).

Dirección inicial del canal 1 = Dirección inicial del módulo

Dirección inicial del canal 2 = Dirección inicial del módulo + 8

Dirección inicial del canal 3 = Dirección inicial del módulo + 16

En AWL se accede a los datos con los comandos PEB (leer 1 byte), PEW (leer 2 bytes) y PED (leer 4 bytes).

La siguiente tabla muestra la asignación de la interfaz de realimentación (entradas) del FM 451.

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	PARA	interno	interno	DATA_ERR	OT_ERR	DIAG	interno	interno
Byte 1	CHGOVER	CUTOFF	ZSPEED	SPEED_OUT	0	WAIT_EI	WORKING	ST_ENBLD
Byte 2	MODE_OUT							
Byte 3	POS_RCD	0	FVAL_DONE	0	GO_P	GO_M	MSR_DONE	SYNC
Byte 4	ACT_POS							
Byte 5								
Byte 6								
Byte 7								

Ejemplo: Valor real de posición ACT_POS

La dirección inicial del módulo es 512.

AWL	Descripción
L PED 516	//Leer valor real de posición actual (ACT_POS) del canal 1
	//con acceso directo:
	//Dirección inicial del canal + 4

Escribir las señales de control mediante acceso directo

Las direcciones de byte se indican de forma relativa respecto a la dirección inicial de las entradas del canal correspondiente. Los nombres de los parámetros corresponden a los nombres del DB de canal (Página 7428).

Dirección inicial del canal 1 = Dirección inicial del módulo

Dirección inicial del canal 2 = Dirección inicial del módulo + 8

Dirección inicial del canal 3 = Dirección inicial del módulo + 16

En AWL se accede a los datos con los comandos PAB (escribir 1 byte), PAW (escribir 2 bytes) y PAD (escribir 4 bytes).

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del FM 451.

Dirección	Número de bit							
	7	6	5	4	3	2	1	0
Byte 0	0	0	0	0	OT_ERR_A	0	0	0
Byte 1	DRV_EN	SPEED252	0	0	DIR_P	DIR_M	STOP	START
Byte 2	MODE_IN							
Byte 3	MODE_TYPE							
Byte 4	Reservado							
Byte 5								
Byte 6								
Byte 7								

Ejemplo: Señales de inicio del canal 2

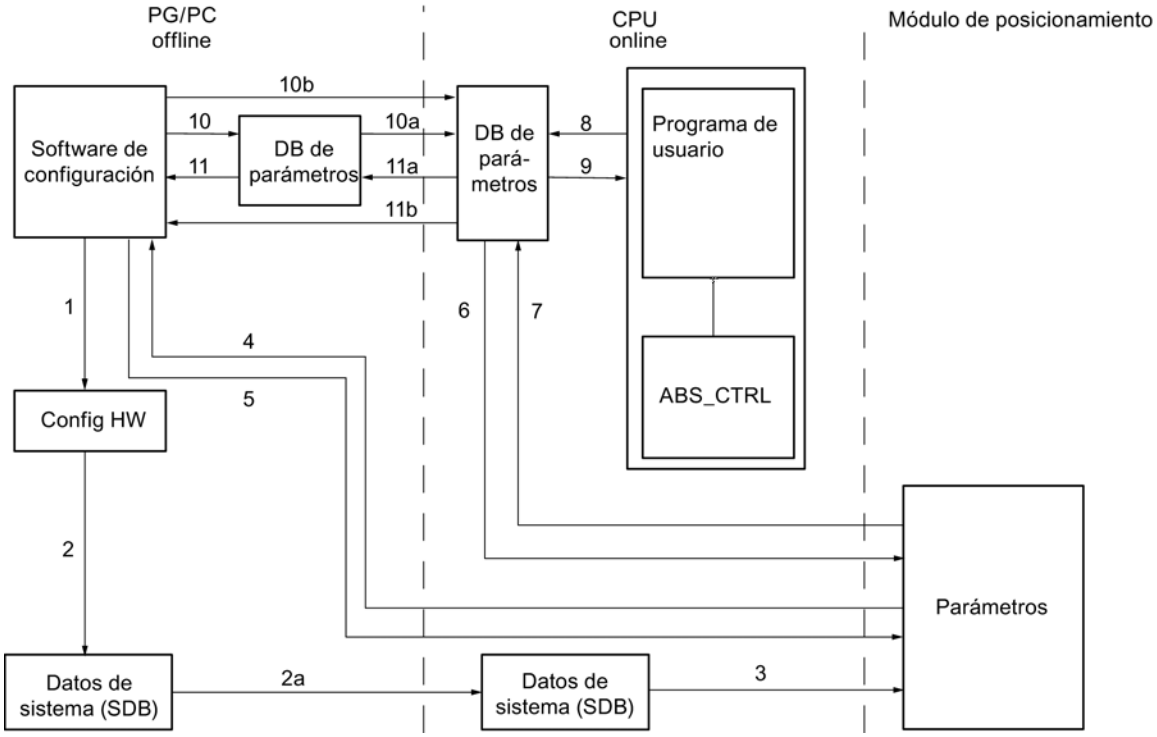
La dirección inicial del módulo es 512.

AWL	Descripción
L 2#10001000	//Poner DRV_EN y DIR_P a 1
T PAB 521	//Escribir señales con acceso directo para canal 2:
	//Dirección inicial del módulo + 8 + 1

Vías de transferencia de parámetros

Se entiende por parámetros los datos máquina y los incrementos que se indican a continuación.

La figura siguiente muestra las vías de transferencia de parámetros al FM x51.



1 Guardar los parámetros en el software de configuración.

2 Guardar y compilar la configuración hardware.

2a Cargar configuración hardware en la CPU. La CPU ejecuta automáticamente el paso 3.

3 La CPU escribe los parámetros en el módulo en parametrizaciones de sistema.

4 Cargar los parámetros de un canal del módulo en la programadora con el comando "Cargar sistema de destino en programadora".

5 Cargar parámetros del software de configuración con el comando "Cargar sistema de destino" en un canal del módulo.

6 Escribir parámetros en un canal del módulo con peticiones del programa de usuario.

7 Leer parámetros desde un canal del módulo a través de peticiones del programa de usuario.

8 Almacenar parámetros del programa de usuario en el DB Online.

9 Leer parámetros del DB Online en el programa de usuario.

10 Exportar parámetros del software de configuración al DB Offline.

10a Cargar DB offline en la CPU.

10b Exportar parámetros del software de configuración al DB Online.

11 Importar parámetros del DB Offline en el software de configuración.

11a Cargar parámetros del DB Online en la programadora.

11b Importar parámetros del DB Online en el software de configuración.

Casos de aplicación de la transferencia de parámetros

Caso de aplicación	Pasos
Los parámetros se editan con el software de configuración. A continuación, los canales del módulo sólo pueden parametrizarse de forma automática en el momento del arranque.	Ejecute los pasos 1, 2 y 2a.
Los parámetros se modifican durante la puesta en servicio en el modo de prueba en el software de configuración.	Ejecute los pasos 4 y 5.
A continuación, los parámetros modificados durante la puesta en servicio se tienen que cargar automáticamente durante el arranque.	Ejecute los pasos 1, 2 y 2a.
Los parámetros se crean con el software de configuración. Los canales del módulo sólo pueden parametrizarse con el programa de usuario a través de bloques de datos.	Ejecute los pasos 10, 10a y 6 y o 10b y 6.
Desea crear datos de reserva para recetas de forma cómoda.	Ejecute los pasos 10 y 10a.
Los parámetros se crean con el software de configuración. Éstos tienen que estar a disposición del programa de usuario para realizar modificaciones provisionales.	Ejecute los pasos 1, 2 y 2a para la parametrización automática.
	Ejecute los pasos 10, 10a, 9, 8 y 6 para el acceso por medio del programa de usuario.
Los parámetros existentes se modifican exclusivamente con el programa de usuario.	Ejecute los pasos 7, 9, 8 y 6.
Desea ver los datos modificados por el programa de usuario con el software de configuración.	Ejecute los pasos 11a y 11 o sólo 11b.
Los parámetros modificados por el programa de usuario se tienen que cargar automáticamente también en el arranque.	Ejecute los pasos 11b u 11a, 11 y, después, 1, 2 y 2a.

Puesta en servicio del FM 451

Uso del FM 451 desde el programa de usuario

Inclusión del FM 451 en un programa de usuario

Para integrar el FM 451 en un programa de usuario, pueden utilizarse instrucciones que le facilitarán en lo posible el manejo de las funciones deseadas:

Las instrucciones ABS_INIT y ABS_CTRL o ABS_CTRL_451 se activan en el programa de usuario de la CPU. Éstas realizan la comunicación entre la CPU y el FM 451.

Si integra la instrucción ABS_CTRL en el proyecto, el DB de canal (Página 7428) se creará automáticamente como DB de instancia en la instrucción ABS_CTRL.
Si integra la instrucción ABS_CTRL_451 en el proyecto, entonces deberá insertar en su programa como DB de canal un bloque de datos del tipo ABS_CHANTYPE mediante "Bloques de programa > Agregar bloque".

Todos los datos de parámetros pertenecientes a un canal del módulo (datos máquina, tablas de incrementos) se encuentran en el DB de parámetros (Página 7434) del módulo. Inserte un

bloque de datos del tipo ABS_PARATYPE en el proyecto mediante "Bloques de programa > Agregar bloque".

Además, para el FM 451 se puede utilizar también la instrucción ABS_DIAG o ABS_DIAG_451, que permite transferir informaciones de diagnóstico al DB de diagnóstico.

Si integra la instrucción ABS_DIAG en el proyecto, el DB de diagnóstico (Página 7436) se creará automáticamente como DB de instancia en la instrucción ABS_DIAG.
Si integra la instrucción ABS_DIAG_451 en el proyecto, entonces deberá insertar en el programa como DB de diagnóstico un bloque de datos del tipo ABS_DIAGTYPE mediante "Bloques de programa > Agregar bloque".

Puesta en servicio del módulo

Incluso sin programa de usuario es posible realizar un primer test del módulo con el cuadro de diálogo "Puesta en servicio". Tenga en cuenta particularmente la advertencia siguiente.

⚠ ADVERTENCIA

Se pueden producir daños personales y materiales.

Para evitar daños personales y materiales observe los puntos siguientes:

- Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Sólo así puede garantizarse que en caso de un fallo del software o del equipo sea posible desconectar la instalación de modo seguro.
- Instale finales de carrera por hardware que actúen directamente en las etapas de potencia de todos los accionamientos.
- Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.
- Si controla y vigila paralelamente el FM 451 desde el programa y desde el cuadro de diálogo "Puesta en servicio" pueden producirse conflictos cuyos efectos no están definidos. Por lo tanto, conmute siempre la CPU al estado STOP cuando trabaje con el cuadro de diálogo "Puesta en servicio" o bien desactive su programa.

Consulte también

ABS_INIT (Página 7419)

ABS_CTRL (Página 7420)

ABS_DIAG (Página 7422)

ABS_CTRL_451 (Página 7424)

ABS_DIAG_451 (Página 7426)

FM 451: Diagnóstico online

Información actual sobre el módulo

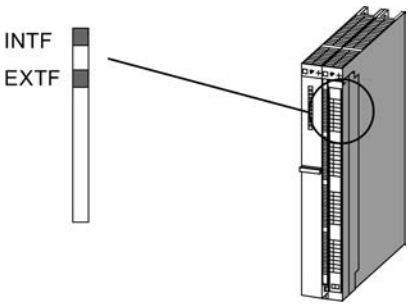
Diagnóstico estándar

Indicación de errores mediante los diodos de error del FM 451

¿Dónde se indica el error?

Los diodos rojos de error del módulo indican diferentes estados de error. Los LED se encienden también en caso de errores que aparecen brevemente, por lo menos, durante 3 s.

En la figura siguiente se muestra la posición de los diodos de error en el FM 451.



¿Qué errores se indican?

La tabla siguiente contiene los tipos de error que se indican mediante los diodos de error.

LED	Significado	Explicación
INTF	Error de grupo para errores internos	Este diodo indica los siguientes estados de error del FM 451: • una avería de módulo; • un error (de canal) interno; • El FM 451 no está configurado. Configure el módulo. Para eliminar el error, ver Clases de error (Página 11085).
EXTF	Error de grupo para errores externos	Este diodo indica un error (de canal) externo.

Diagnóstico específico de canal

Disparo de alarmas de diagnóstico

Procesamiento de las alarmas

El módulo de posicionamiento puede activar alarmas de diagnóstico. Estas alarmas se procesan en un OB de alarma. Cuando se dispara una alarma sin que se encuentre cargado el OB correspondiente, la CPU pasa a STOP.

Para habilitar las alarmas de diagnóstico se procede del modo siguiente:

1. Seleccione el módulo en HWCN.
2. Habilite la alarma de diagnóstico en "Edición > Propiedades del objeto > Parámetros básicos".
3. Guardar y compilar la configuración.
4. Cargue la configuración en la CPU

Visión de conjunto de las alarmas de diagnóstico

Los eventos y errores siguientes disparan una alarma de diagnóstico:

- Fallo operativo
- Datos máquina erróneos
- Error de diagnóstico

Estos errores se describen con detalle en el apartado "Clases de error (Página 11085)".

Reacción del módulo de posicionamiento en caso de error con alarma de diagnóstico

- El posicionamiento se interrumpe.
- Con las alarmas de diagnóstico siguientes se borra la sincronización:
  - Falta conector frontal, falta tensión auxiliar externa para la alimentación del encóder
  - se ha detectado un error de señal cero, error de línea (señales de encóder de 5 V)
  - se ha rebasado la zona de desplazamiento (se notifica con un error de operador)
  - Ajustar valor real no se puede ejecutar (se notifica con un error de operador)
- Las señales de control START, DIR_P y DIR_M dejan de procesarse excepto en un caso.
  Excepción:
  Si se produce un error de operador sigue siendo posible un modo Jog en el sentido de la zona de trabajo.
- El selector de función y las peticiones se siguen procesando.

El módulo de posicionamiento detecta un fallo ("entrante")

Una alarma de diagnóstica es "entrante" cuando por lo menos hay un error en cola de espera. Si no se han solventado todos los errores, los demás que queden en cola de espera, se notificarán de nuevo como "entrantes".

Secuencia:

1. El módulo de posicionamiento detecta uno o varios errores y dispara una alarma de diagnóstico. Uno de los LEDs "INTF" o "EXTF" se enciende. El error se registra en el búfer de diagnóstico.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Es posible evaluar la información de inicio del OB 82.
4. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
5. Encontrará más información si llama la instrucción ABS_DIAG (Página 7422) o ABS_DIAG_451 (Página 7426).

El módulo de posicionamiento detecta el paso al estado libre de fallos ("saliente")

Una alarma sólo es "saliente" cuando se ha eliminado el último error del módulo.

Secuencia:

1. El módulo de posicionamiento detecta que todos los fallos han sido eliminados y dispara una alarma de diagnóstico. El LED "INTF" o "EXTF" está apagado. El búfer de diagnóstico no se modifica.
2. El sistema operativo de la CPU solicita el OB de alarma de diagnóstico, por ejemplo, OB 82.
3. Con el parámetro OB82_MOD_ADDR se lee qué módulo ha disparado la alarma.
4. Evalúe el bit OB82_MDL_DEFECT.
   Si este bit es "0", no hay errores en el módulo. Su evaluación puede finalizar aquí.

Alarmas de diagnóstico según el estado de la CPU

- En estado STOP de la CPU las alarmas de diagnóstico están bloqueadas por el módulo de posicionamiento.
- Si con la CPU en estado STOP no se eliminan todos los errores pendientes, el módulo vuelve a notificar como "entrantes" los errores todavía no eliminados tras la transición al estado RUN.
- Si se eliminan todos los errores pendientes con la CPU en estado STOP, el estado libre de errores del módulo **no** se notificará con una alarma de diagnóstico tras la transición al estado RUN.

Evaluación de errores en el programa de usuario

Reacción a errores en el programa de usuario

En el programa de usuario es posible reaccionar de forma precisa a los errores que aparecen. Para ello se dispone de los siguientes recursos:

- Los valores de retorno RET_VAL de las instrucciones integradas:
  Este valor se vuelve a calcular cada vez que se llama la instrucción.
  RET_VAL = -1 es un indicador de grupo para un error síncrono en una petición o en la comunicación con el módulo.
- A cada petición le corresponde un bit de error _ERR como indicador de grupo de un error de la petición o de uno de sus predecesores en la cadena de peticiones:
  El bit de errores se activa para una petición de escritura y sus peticiones de continuación si el módulo notifica un error de datos o si se produce un error de comunicación.
  En el caso de peticiones de lectura, el bit de error se establece para la petición afectada cuando se ha producido un error de comunicación.
  La instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) activa de nuevo los bits de error después del procesamiento de una petición. No obstante, estos deben ser retomados por el programa de usuario para una evaluación de errores.
- La señal de realimentación DATA_ERR como indicador de grupo de un error que el módulo ha detectado durante una petición de escritura. La señal se vuelve a calcular en la siguiente petición de escritura.
- La señal de realimentación OT_ERR (error de operador) como indicador de grupo de un error que el módulo ha detectado al iniciar un recorrido. El bit de error tiene que ser acusado mediante el OT_ERR_A = 1, una vez solucionada la causa.
- La señal de realimentación DIAG se activa cuando el contenido del búfer de diagnóstico se ha modificado. Esta señal puede aparecer más tarde que las señales DATA_ERR y OT_ERR.
- El error de comunicación JOB_ERR contiene el código de error (Página 11084) si existe un problema de comunicación entre la instrucción y el módulo. El valor se recalcula después de procesar una petición y se crea en el DB del canal con la instrucción ABS_CTRL (Página 7420) o ABS_CTRL_451 (Página 7424) o en el DB de diagnóstico con la instrucción ABS_DIAG (Página 7422) o ABS_DIAG_451 (Página 7426).
- La instrucción ABS_DIAG (Página 7422) o ABS_DIAG_451 (Página 7426) para leer el búfer de diagnóstico del módulo. Aquí es posible averiguar las causas de error para eventos síncrono y asíncronos.
- Las alarmas de diagnóstico para la respuesta rápida a los eventos en el OB de la alarma de diagnóstico (por ejemplo, OB 82).

Evaluación de una alarma de diagnóstico en el programa de usuario

Las siguientes entradas en los datos locales del OB de la alarma de diagnóstico (por ejemplo, OB 82) son activadas por el módulo de posicionamiento. Los errores se introducen en el búfer de diagnóstico (clase de error 128 (Página 11092)).

Dirección	Nombre	Tipo	Comentarios
0.0	OB82_EV_CLASS	BYTE	Clase de evento e identificadores: B#16#38: evento saliente B#16#39: evento entrante
1.0	OB82_FLT_ID	BYTE	Código de error (B#16#42)
2.0	OB82_PRIORITY	BYTE	Clase de prioridad: B#16#1A en estado operativo RUN B#16#1C en estado operativo ARRANQUE
3.0	OB82_OB_NUMBR	BYTE	Número de OB (82)
4.0	OB82_RESERVED_1	BYTE	Reservado
5.0	OB82_IO_FLAG	BYTE	Módulo de entrada: B#16#54
6.0	OB82_MDL_ADDR	INT	Dirección básica lógica de la tarjeta en la que se ha producido el error.
8.0	OB82_MDL_DEFECT	BOOL	Fallo del módulo
8.1	OB82_INT_FAULT	BOOL	Error interno
8.2	OB82_EXT_FAULT	BOOL	Error externo
8.3	OB82_PNT_INFO	BOOL	Error de canal existente
8.4	OB82_EXT_VOLTAGE	BOOL	Falta tensión auxiliar externa
...	no utilizado		
10.3	OB82_WTCH_DOG_FLT	BOOL	Activación de la vigilancia de tiempo
...	no utilizado		
12.0	OB82_DATE_TIME	DATE_AND_TIME	Fecha y hora en la que se ha solicitado el OB

Búfer de diagnóstico del módulo

Eventos de diagnóstico

El búfer de diagnóstico del módulo contiene como máximo 9 eventos de diagnóstico y está organizado como búfer anular.

Un evento de diagnóstico se escribe en el búfer cuando se detecta un aviso (de error) "entrante". Esto puede ser un aviso, un error síncrono (error de datos, fallo de operador) o incluso un error asíncrono (error de servicio y error de diagnóstico). De una causa de un error pueden surgir también varias entradas como errores consiguientes. Los avisos salientes no generan entradas en el búfer de diagnóstico.

Para cada evento de diagnóstico se indican:

- Estado (siempre entrante)
- Error interno
- Error externo

- Clase de error
- Número de error
- Número de canal
- Número de incremento (en errores de la tabla de incrementos)

Cuando se escribe un evento de diagnóstico en el búfer de diagnóstico, la señal de realimentación DIAG = 1 se activa en todos los canales parametrizados.

El búfer de diagnóstico puede transferirse como un todo con la instrucción ABS_DIAG (Página 7422) o ABS_DIAG_451 (Página 7426) a un bloque de datos (DB de diagnóstico) o visualizarse a través de un cuadro de diálogo de evaluación de errores del software de configuración. Si se lee el búfer de diagnóstico, el módulo activa la señal de realimentación DIAG = 0.

Nota

Si el búfer de diagnóstico se lee al mismo tiempo con la instrucción ABS_DIAG o ABS_DIAG_451 y el cuadro de diálogo de evaluación de errores, puede ocurrir que uno de los eventos de diagnóstico que se ha producido no sea detectado por el programa.

Datos y estructura del DB de diagnóstico

Datos y estructura del DB de diagnóstico

Nota

Los datos que no se indican en esta tabla no deben ser modificados.

La tabla siguiente muestra la estructura del DB de diagnóstico.

Dirección	Nombre	Tipo	valor inicial	Comentarios
0.0	MOD_ADDR	INT	0	Dirección del módulo
256.0	JOB_ERR	INT	0	Número de error del error de comunicación
258.0	JOBBUSY	BOOL	FALSE	1 = Petición activa
258.1	DIAGRD_EN	BOOL	FALSE	1 = Leer el búfer de diagnóstico necesariamente
260.0	DIAG_CNT	INT	0	Número de entradas válidas de la lista
262.0	DIAG[1]	STRUCT		Entrada más reciente de datos de diagnóstico
272.0	DIAG[2]	STRUCT		Segunda entrada de datos de diagnóstico
282.0	DIAG[3]	STRUCT		Tercera entrada de datos de diagnóstico
292.0	DIAG[4]	STRUCT		Cuarta entrada de datos de diagnóstico
302.0	DIAG[5]	STRUCT		Quinta entrada de datos de diagnóstico
312.0	DIAG[6]	STRUCT		Sexta entrada de datos de diagnóstico
322.0	DIAG[7]	STRUCT		Séptima entrada de datos de diagnóstico
332.0	DIAG[8]	STRUCT		Octava entrada de datos de diagnóstico
342.0	DIAG[9]	STRUCT		Novena entrada de datos de diagnóstico

La tabla siguiente muestra la estructura de una entrada de diagnóstico DIAG[n].

Dirección	Nombre	Tipo	valor inicial	Comentarios
+0.0	STATE	BOOL	FALSE	0 = Evento saliente 1 = Evento entrante
+0.1	INTF	BOOL	FALSE	1 = Error interno
+0.2	EXTF	BOOL	FALSE	1 = Error externo
+2.0	FCL	INT	0	Clase de error: 1: Fallo operativo 2: Error de operador 4: Error de datos 5: Error de datos máquina 6: Error de las tablas de incrementos 15: Avisos 128: Error de diagnóstico
+4.0	FNO	INT	0	Número de error
+6.0	CH_NO	INT	0	Número de canal
+8.0	TRG_NO	INT	0	Número de incremento

Lista de los avisos JOB_ERR

Avisos JOB_ERR

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A0	32928	-32608	Acuse negativo al leer del módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.
80A2 [1)]	32930	-32606	Error de protocolo para capa 2
80A3 [1)]	32931	-32605	Error de protocolo en User-Interface / User
80A4 [1)]	32932	-32604	Fallo en la comunicación del bus K
80B0	32944	-32592	Registro de datos/Petición desconocida.
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo previsto.
80C0 [1)]	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1 [1)]	32961	-32575	Los datos de una petición de escritura semejante todavía no se han procesado en el módulo.
80C2 [1)]	32962	-32574	El módulo procesa de instante el número máximo posible de peticiones.
80C3 [1)]	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados de momento.
80C4 [1)]	32964	-32572	Error de comunicación

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80C5 [1]	32965	-32571	La periferia descentralizada no está disponible.
80C6 [1]	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB. (petición de escritura).
8532	34098	-31438	El número de DB del DB de parámetros es demasiado grande. (petición de escritura).
853A	34106	-31430	DB de parámetros no presente. (petición de escritura).
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras aparecer un error (petición de escritura).
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB. (petición de lectura).
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU No se pueden escribir los datos en el DB (petición de lectura)
8732	34610	-30926	El número de DB del DB de parámetros es demasiado grande. (petición de lectura).
873A	34618	-30918	DB de parámetros no presente. (petición de lectura).
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras aparecer un error (petición de lectura).

[1] Los errores 80A2 a 80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario.

Clases de fallos

Fallo operativo

Clase de error 1: Fallo operativo

Los fallos operativos se detectan de forma asíncrona respecto a un control. Los fallos operativos hacen que se cancele el posicionamiento a excepción del número de error 9. El número de error 9 hace que se desactive el posicionamiento.

La tabla siguiente muestra los fallos operativos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Rebase del final de carrera por software Inicio**		Sí
	Causa	El valor real se encuentra fuera de la zona de trabajo.	
2	**Rebase del final de carrera por software Fin**		Sí
	Causa	El valor real se encuentra fuera de la zona de de trabajo.	
3	**Rebase del inicio de la zona de desplazamiento**		Sí
	Causa	Rebasar el límite de la zona de desplazamiento (las coordenadas de los límites de la zona de desplazamiento pertenecen a la zona de desplazamiento).	
4	**Rebase del fin de la zona de desplazamiento**		Sí

Nº	Significado		Alarma de diagnóstico
	Causa	Rebasar el límite de la zona de desplazamiento (las coordenadas de los límites de la zona de desplazamiento pertenecen a la zona de desplazamiento).	
5	**Error en la aproximación al destino**		Sí
	Causa	La zona de destino no se ha alcanzado dentro del tiempo de vigilancia.	
6	**Abandonar zona de parada**		Sí
	Causa	El valor real se encuentra fuera de la zona de parada.	
7	**Acoplamiento conjunto**		Sí
	Causa	Modificación del valor real > ½ zona de parada en el sentido erróneo.	
8	**Falta modificación del valor real o es demasiado reducida**		Sí
	Causa	No hay modificaciones del valor real o modificación del valor real en contra del sentido nominal dentro del tiempo de vigilancia.	
9	**Rebasar destino (FM 451)**		Sí
	Causa	En "Ajustar valor real a vuelo" se ha rebasado el destino.	
10	**Rebasar zona de destino**		Sí
	Causa	La zona de destino se ha rebasado después de la aproximación al destino.	
11	**Punto de cambio de velocidad conectado de forma errónea**		Sí
	Causa	El eje oscila en el punto de cambio de velocidad.	
12	**Punto de desconexión conectado de forma errónea**		Sí
	Causa	El eje oscila en el punto de desconexión/punto de inversión	
13	**Inicio de la zona de destino conectado de forma errónea**		Sí
	Causa	El eje oscila en la zona de destino.	
14	**Modificación mayor que la mitad de la zona del eje rotativo**		Sí
	Causa	La velocidad/frecuencia es muy alta o existen saltos erróneos del valor real.	
15	**Modificación mayor que la zona del eje rotativo**		Sí
	Causa	La velocidad/frecuencia es muy alta o existen saltos erróneos del valor real.	
16	**Incremento para número de incremento 252 no transferido (FM 451)**		Sí
	Causa	No se ha transferido el incremento.	
17	**Incremento para número de incremento 252 no aproximable (FM 451)**		Sí
	Causa	La distancia entre la posición real actual y el incremento especificado es inferior a la distancia de cambio de velocidad o la distancia de desconexión.	
18	**Incremento erróneo para el número de incremento 252 (FM 451)**		Sí
	Causa	El incremento se encuentra fuera de la zona de trabajo.	

Error de operador

Clase de error 2: Error de operador

Los errores de operador se detectan al cambiar las señales de control en el área de datos de usuario. Los fallos de operador producen una desactivación del posicionamiento.

La tabla siguiente muestra los errores de operador posibles y sus causas.

N.º	Significado		Alarma de diagnóstico
1	**Modo de operación no admisible**		no
	Causa	El modo de operación seleccionado no está permitido.	
3	**Petición de interfaz no permitida**		no
	Causa	La señal seleccionada no está permitida en este modo de operación.	
4	**Parámetro erróneo del modo de operación**		no
	Causa	• En el modo de operación "Jog" la velocidad especificada es distinta de la velocidad rápida o la velocidad lenta. • En el modo de operación "Modo incremental" el incremento es distinto de 1 a 100 o distinto de 252, 254 y 255.	
5	**Habilitación de inicio no existente**		no
	Causa	Habilitación al iniciar no existente	
7	**El destino/La zona de destino se encuentra fuera de la zona de trabajo.**		no
	Causa	El destino especificado o calculado se encuentra fuera de los finales de carrera por software.	
8	**Eje no parametrizado**		no
	Causa	No se han parametrizado datos máquina para el eje o se han parametrizado datos erróneos.	
9	**El eje no está sincronizado.**		no
	Causa	El modo de operación "Modo incremental" sólo es posible con un eje ya sincronizado.	
10	**Destino/Parte del trayecto no posicionable**		no
	Causa	La distancia entre la posición real actual y el destino especificado es inferior a la distancia de desconexión.	
17	**La búsqueda del punto de referencia no es posible**		no
	Causa	Se ha conectado un encóder SSI.	
18	**El modo incremental relativo o absoluto no es posible**		no
	Causa	El incremento no es válido.	
19	**La distancia de desconexión no es mayor que ½ de la zona de destino en el número de incremento 255**		no
	Causa	La distancia de desconexión para el incremento 255 es inferior a la mitad de la zona de destino.	
20	**Desplazamiento no permitido en el sentido especificado**		no
	Causa	La distancia del final de carrera por software es insuficiente.	

Error de datos

Clase de error 4: Error de datos

Los errores de datos se detectan de forma síncrona respecto a un control/mando. Los datos de errores no producen ninguna respuesta al error.

La tabla siguiente muestra los errores de datos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
6	**Incremento especificado demasiado grande**		no
	Causa	El valor está fuera de ±100 m ó ±1000 m. El recorrido / destino no puede ser superior a la zona de desplazamiento. En un eje rotativo la coordenada debe ser ≥ 0 e inferior al final del eje rotativo.	
10	**Decalaje de señal cero erróneo (FM 451)**		no
	Causa	El decalaje de señal cero es mayor que ± 100 m o ± 1.000 m. Los finales de carrera por software, tras ajustar el decalaje de señal cero, se encuentran fuera de la zona de desplazamiento (100 m...+100 m o -1.000 m... +1.000 m). Eje rotativo: El valor del decalaje de señal cero es mayor que el final del eje rotativo.	
11	**Indicación del valor real errónea**		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
12	**Punto de referencia erróneo**		no
	Causa	Eje lineal: La coordenada se encuentra fuera del final de carrera por software actual (que puede estar desplazado). Eje rotativo: La coordenada es < 0 o mayor que el final del eje rotativo.	
20	**Activar datos máquina no admisible**		no
	Causa	En el módulo no hay datos máquina nuevos sin errores.	
27	**Ajuste codificado en bits no permitido**		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
29	**Codificación de bits no permitida**		no
	Causa	Los bits no utilizados y no descritos aquí son distintos a 0.	
34	**No es posible cancelar la activación del valor real**		no
	Causa	Con un encóder SSI y un eje lineal, tras la ejecución del ajuste el valor real de posición se encontraría fuera de la zona de trabajo.	
36	**Distancia de cambio de velocidad errónea en número de incremento 255**		no
	Causa	El valor se encuentra fuera del margen numérico permitido ±100 m o ±1000 m. En un eje rotativo la coordenada debe ser ≥ 0 e inferior al final del eje rotativo.	
37	**Distancia de desconexión errónea en número de incremento 255**		no
	Causa	El valor se encuentra fuera del margen numérico permitido ±100 m o ±1000 m. La distancia de desconexión es menor que la distancia de cambio de velocidad.	
107	**Eje no parametrizado**		no
	Causa	En el eje no hay ningún dato máquina o los datos no están activados.	
108	**El eje no está sincronizado.**		no
	Causa	Uno de las peticiones "Ajustar valor real" y "Cancelar preselección del valor real" se ha activado a pesar de que el eje no está sincronizado.	

Error de datos máquina

Clase de error 5: Error de datos máquina

La alarma de diagnóstico sólo se disparará con un bloque de datos del sistema (SDB) con errores. Los errores de datos máquina no producen ninguna respuesta al error.

La tabla siguiente muestra errores de datos máquina posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
5	**Error en el ajuste de la alarma de proceso**		Sí
	Causa	Ha intentado seleccionar una alarma de proceso que el módulo no soporta.	
6	**Distancia mínima entre bordes errónea (FM 451)**		Sí
	Causa	Ha indicado como distancia mínima entre bordes un valor < 0 ó > 10^9 µm.	
7	**Sistema de medida erróneo**		Sí
	Causa	El valor del sistema de medida se encuentra fuera del rango admisible de 1 a 4 y 6.	
8	**Tipo de eje erróneo**		Sí
	Causa	Como tipo de eje no ha indicado ni 0 ni 1.	
9	**Final erróneo del eje rotativo**		Sí
	Causa	El valor del final del eje rotativo se encuentra fuera del rango admitido de 1 a 10^9 µm o respectivamente de 1 a 10^8 µm (según sea la resolución).	
10	**Tipo de encóder erróneo**		Sí
	Causa	El valor del tipo de encóder se encuentra fuera del rango admisible de 1 a 4.	
11	**Recorrido erróneo por vuelta de encóder**		Sí
	Causa	El valor del recorrido/vuelta del encóder se encuentra fuera del rango admisible de 1 a 10^9 µm (independientemente de la resolución).	
13	**Número erróneo de incrementos por vuelta del encóder**(véase el apartado "Incrementos por vuelta del encóder (Página 11060)")		Sí
14	**Número erróneo de vueltas** (véase el apartado ""Número de vueltas de encóder (Página 11062)")		Sí
15	**Velocidad de transferencia errónea**		Sí
	Causa	Para la velocidad de transferencia se ha indicado un valor fuera del rango admisible de 0 a 3.	
16	**Coordenada del punto de referencia errónea**		Sí
	Causa	La coordenada se encuentra fuera del margen de -100 m a +100 m o, respectivamente, de -1000 m a +1000 m (según la resolución). Eje lineal: La coordenada se encuentra fuera de la zona de trabajo. Eje rotativo: La coordenada es mayor que el final del eje rotativo o < 0.	
17	**Calibración errónea del encóder absoluto**		Sí
	Causa	Encóder de recorrido SSI: El valor del reajuste del encóder absoluto no se encuentra dentro del rango del encóder (incrementos por vuelta del encóder × cantidad de vueltas - 1).	
18	**Tipo erróneo de búsqueda del punto de referencia**		Sí
	Causa	Se ha indicado un valor por fuera de los valores admisibles 0, 1, 2 y 3.	
19	**Sentido de contaje erróneo**		Sí
	Causa	Se ha indicado un valor por fuera de los valores admisibles 0 y 1.	

Nº	Significado		Alarma de diagnóstico
20	Vigilancia de hardware no posible		Sí
	Causa	Se ha puesto en "FALSE" la monitorización de errores de telegrama en el DB de parámetros. En el encóder utilizado, la vigilancia de impulsos erróneos no es posible. Desactive el parámetro MON_PULSE.	
21	Inicio erróneo del final de carrera por software		Sí
	Causa	Eje lineal: El inicio del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m o 1000 m...+1000 m, dependiendo de la resolución). Eje lineal: El inicio del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es menor que -100 m o -1000 m (según la resolución).	
22	Fin erróneo del final de carrera por software		Sí
	Causa	Eje lineal: El fin del final de carrera por software se encuentra fuera de la zona de desplazamiento (-100 m...+100 m ó 1000 m...+1000 m, dependiendo de la resolución) o es menor que el inicio del final de carrera por software. Eje lineal: El fin del final de carrera por software (dado el caso, incluido un decalaje de señal cero presente) es mayor que +100 m ó +1000 m (según la resolución).	
23	Velocidad máxima errónea		Sí
	Causa	Los datos no mostrados en el DB de parámetros tienen que ser 0.	
24	Zona de destino errónea		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m ó 1000 m, según la resolución. Eje rotativo: Zona superior al final del eje rotativo.	
25	Tiempo de vigilancia erróneo		Sí
	Causa	El valor del tiempo de vigilancia se encuentra fuera del rango admisible de 0 a 100 000 ms.	
26	Zona de parada errónea		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m o 1000 m, según la resolución. Eje rotativo: Zona superior al final del eje rotativo.	
127	Velocidad de parada errónea		Sí
	Causa	El valor de la velocidad de parada se encuentra por fuera del rango admisible de 0 a 100 000 µm/min.	
128	Modo de control erróneo		Sí
	Causa	Para el modo de control se ha indicado un valor que está fuera del rango admisible de 1 a 4.	
129	Velocidad de inicio errónea para la búsqueda del punto de referencia		Sí
	Causa	Como velocidad de inicio no ha indicado ni 0 ni 1.	
130	Distancia de cambio de velocidad errónea en el sentido +		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m o 1000 m (según la resolución). Eje rotativo: Zona superior al final del eje rotativo e inferior a la mitad de la zona de destino.	
131	Distancia de cambio de velocidad errónea en el sentido -		Sí
	Causa	Eje lineal: Zona entre 0 y 100 m ó 1000 m (según la resolución). Eje rotativo: Zona superior al final del eje rotativo e inferior a la mitad de la zona de destino.	

Nº	Significado		Alarma de diagnóstico
132	**Distancia de desconexión errónea en el sentido +**		Sí
	Causa	La distancia de desconexión es mayor que la distancia de cambio de velocidad positiva, inferior a la mitad de la zona de destino, o se encuentra fuera del margen admisible entre 0 y 100 m o 1000 m (según resolución).	
133	**Distancia de desconexión errónea en el sentido -**		Sí
	Causa	La distancia de desconexión es mayor que la distancia de cambio de velocidad negativa, inferior a la mitad de la zona de destino, o se encuentra fuera del margen admisible entre 0 y 100 m o 1000 m (según resolución).	
200	**Resolución errónea**		Sí
	Causa	Ha indicado una resolución < 0,1 µm/impulso o >1000 µm/impulso. Ha indicado un recorrido/vuelta y una cantidad de impulsos/vuelta que dan una resolución < 0,1 ó > 1000.	
201	**El encóder no se adapta a la zona de trabajo / área del eje rotativo**		Sí
	Causa	Encóder SSI y eje rotativo: El encóder no cubre exactamente el área del eje rotativo. Eje lineal: El encóder no cubre ni siquiera la zona de trabajo (finales de carrera por software incluidos).	

Error de las tablas de incrementos

Clase de error 6: Error de las tablas de incrementos

Los errores de tablas de incrementos no producen ninguna respuesta al error.

La tabla siguiente muestra los posibles errores de tablas de incrementos y sus causas.

Nº	Significado		Alarma de diagnóstico
6	**Incremento especificado demasiado grande en la tabla de incrementos**		no
	Causa	El valor está fuera de ±100 m ó ±1000 m. El recorrido / destino no puede ser superior a la zona de desplazamiento. En un eje rotativo, la coordenada debe ser ≥ 0 e inferior al final del eje rotativo.	

Avisos

Clase de error 15: Avisos

Los avisos no producen ninguna respuesta al error.

La tabla siguiente muestra los avisos posibles y sus causas.

Nº	Significado		Alarma de diagnóstico
1	**Inicio de parametrización**		no
	Causa	El módulo ha detectado una parametrización mediante un bloque de datos del sistema.	
2	**Final de la parametrización**		no
	Causa	El módulo ha procesado sin problemas la parametrización mediante un bloque de datos del sistema.	
11	**Distancia insuficiente del punto de cambio de velocidad**		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	
12	**Distancia insuficiente del punto de inversión**		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	
14	**Distancia insuficiente del punto de desconexión**		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	
15	**Distancia insuficiente del principio de la zona de destino**		no
	Causa	Los tiempos de respuesta del hardware no pueden mantenerse, pues la distancia entre los puntos de conmutación es demasiado reducida.	

Error de diagnóstico

Clase de error 128: Error de diagnóstico

La tabla siguiente muestra los errores de diagnóstico posibles y sus causas, así como los efectos y las soluciones posibles.

Nº	Significado		Alarma de diagnóstico
4	**Falta tensión auxiliar externa**		Sí
	Causa	• La tensión auxiliar externa de 24 V no se ha conectado o ha fallado • El fusible del módulo está defectuoso. • Tensión insuficiente • Rotura de hilo de masa • Cortocircuito (p. ej. en el encóder conectado)	
	Efecto	• El posicionamiento se cancela en todos los canales • Desactivar las salidas • Eliminación de la sincronización en encóders incrementales cuando falta la tensión auxiliar para la alimentación del encóder • El módulo de posicionamiento no está parametrizado • Borrado de la habilitación de inicio	
	Solución	Comprobar que la conexión de 24 V sea correcta (si la conexión de 24 V es correcta, el módulo está averiado).	
5	**Falta conector frontal (FM 451)**		Sí
	Causa	El conector frontal del módulo de posicionamiento no está insertado	

Nº	Significado		Alarma de diagnóstico
	Efecto	• Falta tensión auxiliar externa de 24 V • El módulo no está listo para funcionar	
	Solución	Enchufar el conector frontal en el módulo de posicionamiento	
51	La vigilancia de tiempo (watchdog) ha reaccionado		Sí
	Causa	• Interferencias fuertes en el módulo de posicionamiento • Error en el módulo de posicionamiento	
	Efecto	• El módulo se resetea • Desactivar todas las salidas • Siempre que tras el reseteado del módulo no se detecte ningún defecto en el módulo, este vuelve a estar listo para funcionar. • El módulo notifica el transcurso de la vigilancia del tiempo (watchdog) con "entrante" y "saliente"	
	Solución	• Eliminación de interferencias • Consulte al departamento comercial correspondiente cuando las circunstancias exactas que han dado lugar al error son de mayor entidad. • Cambio del módulo de posicionamiento	
144	Rotura de hilo Encóder		Sí
	Causa	• El cable del encóder no está enchufado o está cortado • Encóder sin señales de cuadratura • Asignación de pines incorrecta • Longitud de cable excesiva • Cortocircuito de las señales del encóder • Errores de flancos de las señales de encóder • Máxima frecuencia de entrada de la entrada del encóder sobrepasada • Corte de la alimentación del encóder	
	Efecto	• El posicionamiento se interrumpe • Desactivar las salidas • Borrado de la sincronización, con encóders incrementales • Borrado de la habilitación de inicio	
	Solución	• Comprobar el cable del encóder • Guardar los datos técnicos del encóder • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador. • Guardar las especificaciones técnicas del módulo	
145	Error de telegrama Encóder absoluto		Sí

Nº	Significado		Alarma de diagnóstico
	Causa	El tráfico de telegramas entre el módulo de posicionamiento y el encóder absoluto (SSI) tiene errores o está interrumpido: • El cable del encóder no está enchufado o está cortado • Tipo de encóder erróneo • Encóder mal configurado (encóders programables) • Longitud de telegrama mal indicada • El encóder proporciona valores erróneos (encóder defectuoso) • Interferencia en el cable del sistema de medición • Velocidad de transferencia escogida demasiado alta • Tiempo monoestable del encóder superior a 64 µs	
	Efecto	• El posicionamiento se interrumpe • Desactivar las salidas • Borrado de la habilitación de inicio	
	Solución	• Comprobar el cable del encóder • Comprobar el encóder • Comprobar el tráfico de telegramas entre el encóder y el módulo de posicionamiento	
146	**Pulsos erróneos (encóder incremental)**		Sí
	Causa	• Vigilancia del encóder ha detectado pulsos erróneos • El número de incrementos por vuelta está mal introducido • Encóder defectuoso: no proporciona el número de impulsos indicado • Marca cero errónea o no presente • Interferencias en el cable del encóder	
	Efecto	• El posicionamiento se interrumpe • Desactivar las salidas • Borrado de la habilitación de inicio	
	Solución	• Indicar correctamente el número de incrementos/vuelta del encóder (cuadro de diálogo de parametrización) • Comprobar el encóder y su cable • Cumplir las normas de apantallamiento y de conexión a tierra • La vigilancia puede desactivarse de forma provisional con el cuadro de diálogo de parametrización bajo la responsabilidad del operador.	

11.4.2.7 Uso del ET 200S 1PosU

Asignación de terminales del 1PosU

Reglas de cableado

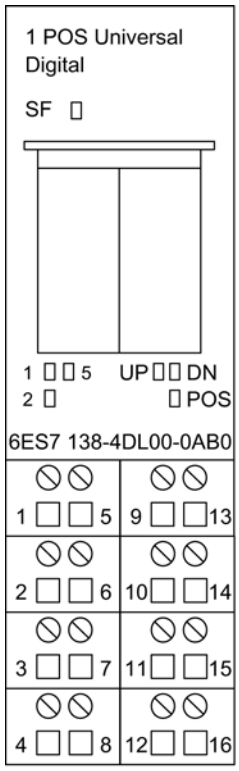
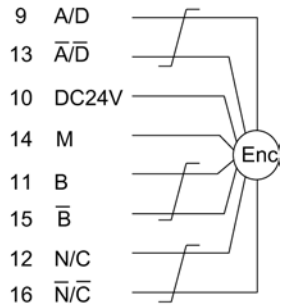
Al utilizar un encóder de posición con señales diferenciales de 5 V, los cables a los bornes 9 y 13, los bornes 12 y 16, y, en caso de un encóder incremental, los cables a los bornes 11 y 15 deben estar apantallados y trenzados por pares.

Al utilizar un encóder incremental con señales de 24 V, los cables a los bornes 9, 11 y 12 deben estar apantallados.

La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contactado de pantallas (ver el anexo de las instrucciones de servicio Sistema de periferia descentralizada ET 200S (http://support.automation.siemens.com/WW/view/es/1144348)).

Asignación de terminales

La tabla siguiente muestra la asignación de terminales para el 1PosU.

Vista	Asignación de terminales	Observaciones			
		Conexión de los interruptores y del accionamiento: Bornes 1 … 8		**Conexión del encóder de posición, señales diferenciales de 5 V o señales de 24 V: Bornes 9 … 16**	
		1: IN0	Final de carrera -	9: A/D	Pista A/datos del encóder SSI
		5: IN1	Final de carrera +	13: /A / /D	
		2: IN2	Leva reductora; señal de congelación	10: DC 24 V	Alimentación para el encóder de posición
		6: DC 24 V	Alimentación para los interruptores	14: M	
		3: OUT0	Desplazamiento negativo o bien marcha rápida	11: B	Pista B
		7: 2L+	Alimentación de la tensión de carga para OUT0, OUT1 y OUT2	15: /B	
		4: OUT1	Desplazamiento positivo o bien marcha lenta	12: N/C	Pista N/reloj SSI (línea reloj)
		8: OUT2	Marcha rápida/ lenta o bien desplazamiento positivo/negativo	16: /N / /C	

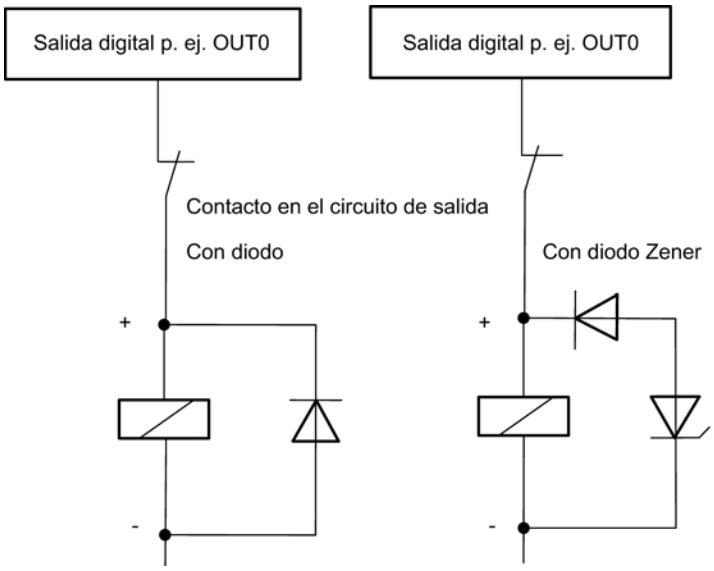
Conexión de relés y contactores a las salidas digitales

Nota

La conexión directa de inductancias (p. ej. de relés y contactores) es posible sin cableado externo. Si los circuitos de salida SIMATIC pueden desactivarse mediante contactos montados adicionalmente (p. ej. contactos de relé), en caso de inductancias deberá prever unos dispositivos de protección contra sobretensión adicionales (ver el siguiente ejemplo de la protección contra sobretensión).

Ejemplo de protección contra sobretensión

La figura siguiente muestra un circuito de salida que hace necesarios dispositivos adicionales de protección de sobretensión. Las bobinas con mando por corriente continua se conectan con diodos normales o diodos Zener.



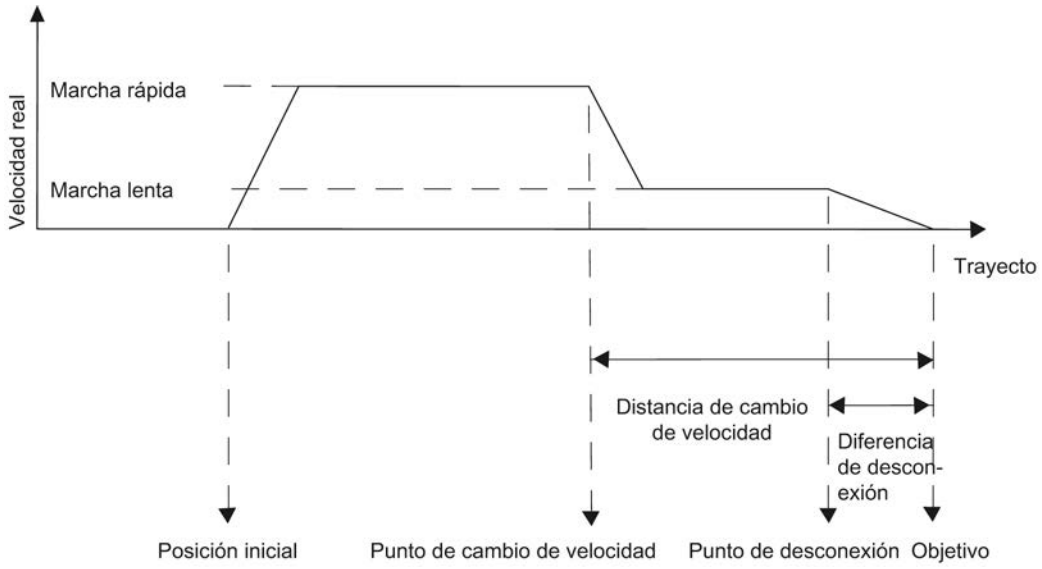
Principios básicos del posicionamiento controlado a través de una marcha rápida/lenta

Proceso de posicionamiento

A partir de la posición de inicio primero se realiza una aproximación al punto de destino a una velocidad mayor (velocidad rápida). Al llegar a una distancia predeterminada de la posición de destino (punto de cambio de velocidad), se cambia a una velocidad inferior (velocidad lenta). Poco antes de que el eje llegue al punto de destino, también a una distancia predefinida con respecto al punto de destino, el accionamiento se desconecta (punto de desconexión).

El accionamiento se controla mediante salidas digitales a marcha rápida o lenta y en el sentido correspondiente.

La figura siguiente muestra los puntos de de conmutación y las distancias de cambio de velocidad. Para simplificar el proceso, el cambio de velocidad se representa a lo largo del trayecto recorrido.



Definiciones

La tabla siguiente contiene los términos principales para el posicionamiento controlado a través de la marcha rápida y la marcha lenta.

Término	Explicación
Zona de trabajo	define la zona que se determina para su tarea, con ayuda de los finales de carrera por hardware. En un encóder SSI, la zona de trabajo se ve limitada además por la zona cubierta por el encóder SSI: La zona de alcance del encóder debe especificarse en los parámetros • Impulsos por vuelta del encóder y • número de vueltas. Impulsos por vuelta del encóder = impulsos por vuelta × vueltas Zona máxima de trabajo: • Eje lineal máx. 0 hasta (área del encóder – 1) • Eje rotativo de 0 a (área del encóder – 1) En los encóders incrementales, la zona de trabajo está limitada a: • máx. 0 hasta 16 777 215 pasos para eje lineal • 0 hasta fin parametrizado del eje rotativo configurado para eje rotativo
Distancia de cambio de velocidad	Define la distancia con respecto al punto de destino a la que el accionamiento cambia la velocidad de velocidad rápida a velocidad lenta.
Punto de cambio de velocidad	Define la posición en la que el accionamiento cambia la velocidad de velocidad rápida a velocidad lenta.
Distancia de desconexión	define la distancia hasta el punto de destino a la que se desconecta el accionamiento. Si la distancia de desconexión ≥ distancia de cambio de velocidad, no existe punto de cambio de velocidad. No se cambia la velocidad de velocidad rápida a velocidad lenta.
Punto de desconexión	Define la posición en la que el accionamiento se desconecta. El 1PosU indica en este punto el final de la marcha.

Término	Explicación
Posición inicial	Define la posición del accionamiento dentro de la zona de trabajo desde la que se inicia la marcha. Si la posición inicial se encuentra dentro de la distancia de desconexión, el accionamiento no se controla. El 1PosU indica en este punto el final de la marcha. Si la posición de inicio se encuentra dentro de la distancia de cambio de velocidad, la marcha sólo se produce en velocidad lenta.
Objetivo	Define la posición absoluta o relativa del eje a la que se llega en el posicionamiento. El punto de destino es la posición de un eje que debe alcanzarse con una marcha. En una marcha absoluta se define el punto de destino directamente desde el programa de control. En una marcha relativa el punto de destino se calcula a partir de la posición inicial y del trayecto predeterminado en el programa de control. Para determinar la precisión con la que ha alcanzado el punto de destino debe comparar el valor real con los datos de la posición.
Eje lineal	Define el tipo de eje con zona de trabajo limitada. Está limitado por: • la zona del encóder • el margen numérico representable (0 hasta 16 777 215 pasos) • Fin de carrera por hardware
Eje rotativo	Define el tipo de eje con zona de trabajo ilimitada. La posición del eje se pone de nuevo a 0 tras un giro (final parametrizado del eje rotativo para encóder incremental o zona del encóder parametrizada para encóder SSI).
Sentido -	Si el accionamiento se mueve en sentido negativo, el valor real indicado se reduce.
Sentido +	Si el accionamiento se mueve en sentido positivo, el valor real indicado aumenta.

Funciones del 1PosU

Sinopsis de las funciones

Resumen

El 1PosU proporciona las siguientes funciones para mover los ejes:

- Detener (Página 11104)
- Jog (Página 11105)
- Búsqueda del punto de referencia (Página 11107)
- Posicionamiento relativo (Página 11113)
- Posicionamiento absoluto (Página 11115)

Además de los tipos de movimiento, el 1PosU dispone de las siguientes funciones:

- Cancelar procesamiento de JOB (Página 11118)
- Ajustar valor real (Página 11119)
- Mover zona de alcance del encóder (Página 11120)

- Modificar distancia de desconexión (Página 11122)
- Modificar distancia de cambio de velocidad (Página 11123)
- Evaluar señal de búsqueda de referencia (Página 11124)
- Función congelar (latch) (Página 11126)
- Ajustar vigilancia de sentido de giro (Página 11128)
- Mostrar valores actuales (Página 11129)
- Detección de fallos/Diagnóstico (Página 11131)
- Comportamiento en caso de STOP de la CPU/del maestro (Página 11138)

Parámetro

En los parámetros se definen una única vez los valores relacionados con accionamiento, eje y encóder.

Modo de dosificación

Al usar encóders incrementales es posible utilizar 1PosU para dosificar. El modo de dosificación se ajusta en los parámetros. En el modo de dosificación, el 1PosU sólo evalúa la señal de encóder A (/A). El valor real se incrementa con cada flanco ascendente.

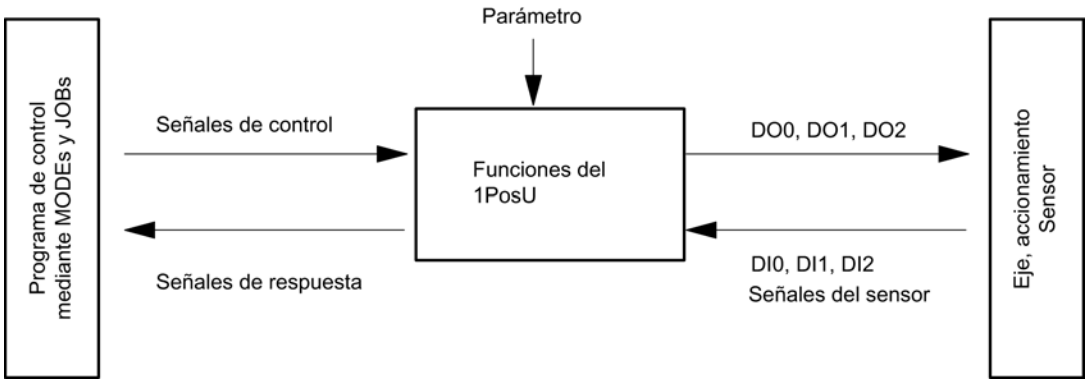
En el modo de dosificación sólo están disponibles las funciones "Jog" y "Posicionamiento relativo" para el control de las salidas digitales.

La propia dosificación se lanza mediante la función "Posicionado relativo". La cantidad de dosificado se predetermina durante el inicio a través de la interfaz de control (parte del trayecto).

Cada vez que se inicia, el valor real se ajusta a 0 y las salidas digitales se controlan en función de la distancia de cambio de velocidad y de desconexión.

Funcionamiento del 1PosU

La figura siguiente muestra el funcionamiento del 1PosU.



Interfaces con el programa de control y con el eje

Para ejecutar la función, el 1PosU, como interfaz con el eje, dispone de entradas digitales, señales de encóder para la conexión de un encóder y salidas digitales para controlar el accionamiento.

Inicio de tipos de movimiento (MODEs)

Los tipos de movimiento (MODEs) se controlan y observan desde el programa de control mediante señales de control y señales de realimentación.

Inicio de MODEs

La tabla siguiente muestra la forma en la que se inician los MODE, así como la respuesta que puede tener el 1PosU.

Su acción	Respuesta del 1PosU
Alimente la interfaz de control (Página 11158) según el MODE con los datos necesarios. Compruebe si el bit de realimentación (Página 11159) es POS_ACK = 0.	
Ponga el bit de control START de 0 a 1.	El 1PosU activa el bit de realimentación POS_ACK = 1 y POS_DONE = 0. A continuación se detecta que el inicio del 1PosU se ha reconocido y que en el caso de POS_ERR = 0 se ejecuta el MODE. En el caso de POS_ERR = 1 el MODE no se ejecuta.
Active el bit de control START de 1 a 0.	El 1PosU activa el bit de realimentación POS_ACK = 0.
	En las funciones de detención, búsqueda del punto de referencia, posicionamiento absoluto y posicionamiento relativo, el 1PosU activa el bit de realimentación POS_DONE = 1 cuando el MODE ha finalizado sin ningún fallo. En el caso de POS_ERR = 1 el MODE se finaliza con un error.
Sólo si POS_ACK = 0 puede iniciar un nuevo MODE. Si se produce un arranque durante un MODEs en marcha, el 1PosU se encarga del nuevo movimiento y realiza un cambio de sentido en caso necesario.	

La siguiente figura muestra la interacción de las señales de control y realimentación al iniciar el MODEs.

Activación de otras funciones (JOBs)

El resto de funciones (JOBs) se controla y observa desde el programa de control mediante señales de control y realimentación.

Activación de JOBs

La tabla siguiente muestra la forma en que se inician los JOBs, así como las reacciones que puede tener el 1PosU.

Su actividad	Reacción del 1PosU
Suministre los datos necesarios a la interfaz de control (Página 11158) según el JOB. Compruebe si el bit de realimentación (Página 11159) es JOB_ACK = 0.	
Ajuste el bit de control JOB_REQ de 0 a 1.	El 1PosU activa el bit de realimentación JOB_ACK = 1. A continuación se detecta que el inicio del 1PosU ha sido detectado y que en el caso de JOB_ERR = 0 se ejecuta el JOB. • En el caso de "Evaluar señal de referenciación", el 1PosU activa simultáneamente el bit de realimentación SYNC = 0. • Para la función de congelación el 1PosU activa simultáneamente el bit de realimentación LATCH_DONE = 0. • Todos los demás JOBs quedan ejecutados. En el caso de JOB_ERR = 1, el JOB no se ejecuta.
Ajuste el bit de control JOB_REQ de 1 a 0.	El 1PosU activa el bit de realimentación JOB_ACK = 0.
	En el caso de "Evaluar señal de referenciación", el 1PosU activa el bit de realimentación SYNC = 1 tras ejecutarse la función. En el caso de congelación el 1PosU activa el bit de realimentación LATCH_DONE = 1 tras ejecutarse la función.
Sólo si JOB_ACK = 0 podrá iniciarse otro JOB.	

La siguiente figura muestra la interacción de las señales de control y realimentación al activar JOBs.

Eje, accionamiento y encóder

Evaluación de las señales del encóder

El 1PosU evalúa las señales suministradas por el encóder en función del tipo de encóder;

- Encóder SSI:
  El 1PosU evalúa el valor suministrado por el encóder SSI directamente por pasos y seguidamente determina el valor real en pasos (valor real = valor del encóder).
  El valor real se encuentra dentro del rango del encóder de 0 (número de vueltas × número de incrementos) – 1. En los límites de la zona de alcance, el 1PosU genera un rebase por exceso o por defecto del valor real.
- Encóders incrementales:
  El 1PosU evalúa cuatro veces los impulsos del encóder y los suma al valor real de forma específica al sentido. Para todos los recorridos especificados en los parámetros y la interfaz de control o realimentación deberá tenerse en cuenta esta evaluación cuádruple:
  1 impulso del encóder incremental equivale a 4 incrementos del 1PosU.
  El valor real se encuentra dentro de la zona de trabajo de 0 … 16 777 215 incrementos.
  En los límites de la zona de trabajo el 1PosU genera un rebase por exceso o por defecto del valor real en la zona de trabajo.
- Encóders incrementales en modo de dosificación:
  El 1PosU sólo evalúa los flancos ascendentes de la señal de encóder A y los suma al valor real.
  El valor real se encuentra dentro de la zona de trabajo entre 0 … 16 777 215 incrementos.
  En los límites de la zona de trabajo superior, el 1PosU genera un desbordamiento del valor real.

Control del accionamiento

Las 3 salidas digitales del 1PosU controlan el accionamiento.

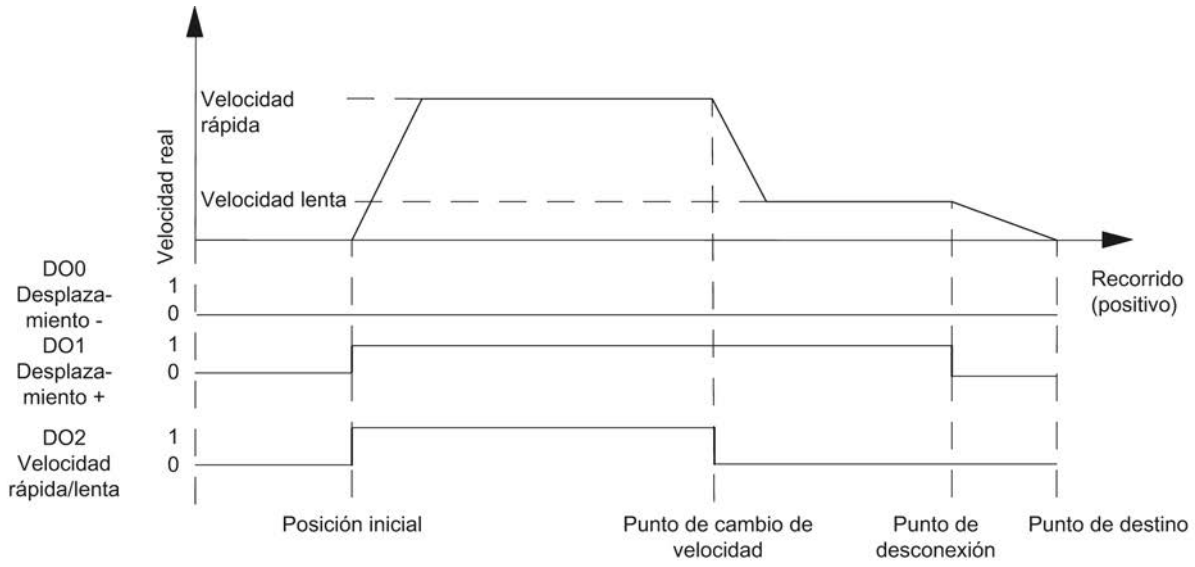
La velocidad se selecciona mediante bit de control SPEED (SPEED = 0 es la marcha lenta; SPEED = 1 es la marcha rápida). La velocidad también puede modificarse durante la marcha.

El cambio de sentido (Página 11146) se controla con el parámetro "T_{min} Cambio de sentido".

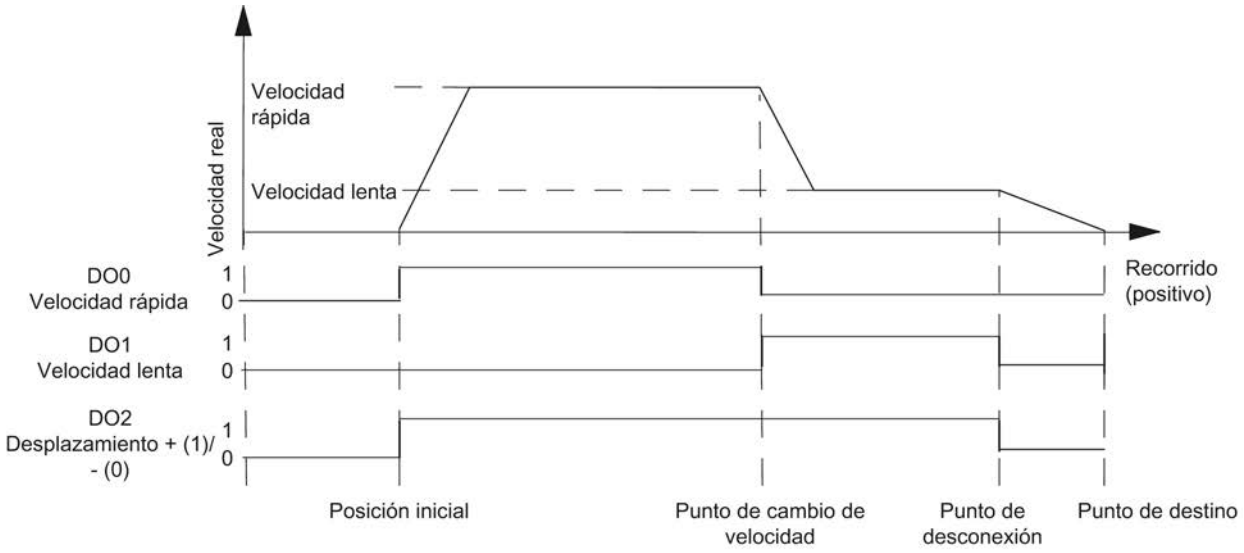
El estado de cada una de las salidas puede leerse en la interfaz de realimentación (Página 11159) (DO 0, DO 1 y DO 2).

La función de las salidas digitales depende del modo de control:

La figura siguiente muestra los estados de las salidas digitales con el **modo de control 0**.



La figura siguiente muestra los estados de las salidas digitales con el **modo de control 1**.



Efecto de las habilitaciones en función del sentido

Descripción

Los bits de control DIR_M y DIR_P habilitan las salidas digitales en función del sentido.

- Con DIR_M = 1 se desplaza en el sentido negativo.
- Con DIR_P = 1 se desplaza en el sentido positivo.

Interrumpir y continuar la marcha

Si, durante la marcha se resetea la habilitación relevante dependiente del sentido, el movimiento del eje se detiene, las 3 salidas digitales se ponen a "0" y la marcha se interrumpe.

Si vuelve a activar la habilitación relevante dependiente del sentido, se continúa la marcha.

Detener (MODE 0)

Detener

Si inicia MODE 0, el 1PosU detiene el desplazamiento en curso, las tres salidas digitales se ponen a "0" y la marcha finaliza (POS_ERR = 0, POS_DONE = 1).

Un marcha finalizada con MODE 0 ya no puede continuarse. Para volver a mover el eje, inicie un nuevo MODE.

Señales de control

La siguiente tabla muestra los datos relevantes de la función "detener" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	0	0	0	MODE 0 = Detener
	Bit 0: START					

Señales de realimentación

La siguiente tabla muestra los datos relevantes de la función "detener" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK

Jog (MODE 1)

JOG

Con "Jog" se controla el accionamiento directamente a través de los bits de control (Página 11100)DIR_M o DIR_P en un sentido.

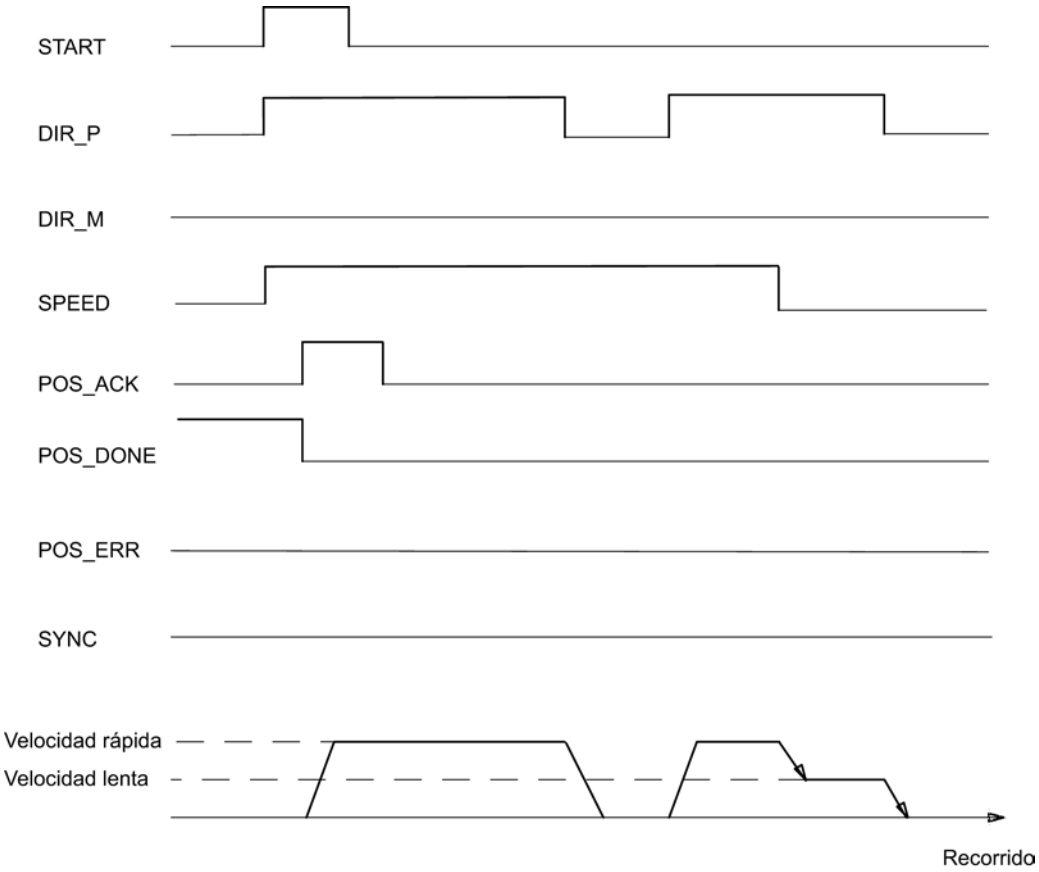
Si inicia MODE 1, el 1PosU mueve el accionamiento con la velocidad predeterminada (bit de control (Página 11158)SPEED) en el sentido predeterminado (bits de control (Página 11158)DIR_M o DIR_P).

El accionamiento se detiene activando los bits de control DIR_M = 0 y DIR_P = 0.

Se realiza un cambio de sentido (Página 11146) después de transcurrido el tiempo T_{min}.

El modo Jog también puede realizarse en un eje no sincronizado (bit de respuesta SYNC = 0), si hay un error de encóder existente (bit de respuesta ERR_ENCODER = 1) o si el encóder no está conectado.

La siguiente figura muestra el desarrollo del modo de movimiento "Jog".



Señales de control

La siguiente tabla muestra los datos relevantes para la función "Jog" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	0	0	1	MODE 1 = Jog
	Bit 3: SPEED (SPEED = 0 es marcha lenta; SPEED = 1 es marcha rápida) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					

Señales de realimentación

La siguiente tabla muestra los datos relevantes de la función "Jog" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Bytes 1 a 3	Valor real: • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1

Causas del error para POS_ERR

Las causas de error deben determinarse con el JOB 15 (Mostrar valores actuales (Página 11129)).

La tabla siguiente muestra los números relevantes en la función "Jog", con la causa de error y las posibles soluciones.

Número de error.	Causa del error	Solución
2	Se indica ERR_2L+	Compruebe la tensión de carga (2L+) en la terminal 7.
5	El final de carrera en cuyo sentido se controla el accionamiento está activo	Revise los encóders y el cableado, así como los parámetros "DI0 Final de carrera - (Página 11142)" y "DI1 Final de carrera + (Página 11142)".
7	Jog: DIR_P = 1 del DIR_M = 1	

Número de error.	Causa del error	Solución
13	El sentido de giro del accionamiento y del encóder es diferente	Revise el cableado del accionamiento y el encóder, así como el parámetro "Inversión del sentido de giro (Página 11141)".
15	En el modo de dosificación: DIR_M = 1	

Búsqueda del punto de referencia (MODE 3)

Búsqueda del punto de referencia (MODE 3)

Búsqueda del punto de referencia

La búsqueda del punto de referencia sólo puede iniciarse en el encóder incremental y con un accionamiento de dosificación no activado.

La búsqueda del punto de referencia permite sincronizar el eje a partir de una señal de referencia externa. Como señal de búsqueda de referencia puede escoger entre 3 entradas digitales y la señal cero (Página 11110).

Las entradas digitales DI0 (final de carrera -), DI1 (final de carrera +) y DI2 (leva reductora) se pueden parametrizar como contacto normalmente cerrado o contacto normalmente abierto.

Alimente la interfaz de control (Página 11158) con la coordenada del punto de referencia e inicie MODE 3. El 1PosU activa la señal de respuesta SYNC = 0 y mueve el accionamiento con la velocidad predeterminada (bit de control SPEED) hacia el sentido de arranque parametrizado y busca la señal de referencia. A este respecto, el 1PosU realiza automáticamente el cambio de sentido necesario en los finales de carrera o en la leva de reducción.

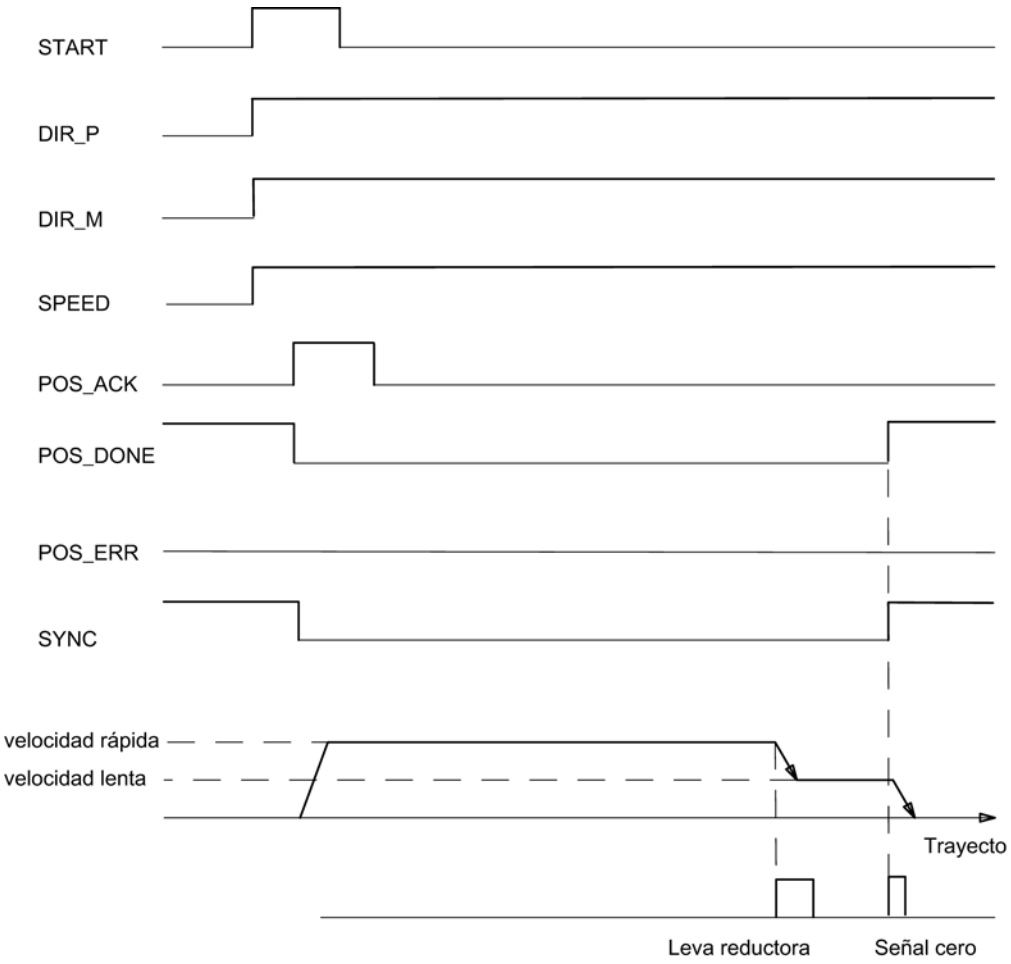
Realice las habilitaciones dependientes del sentido (Página 11100) que sean necesarias (DIR_M, DIR_P) para que se controle el accionamiento.

Si el 1PosU detecta la señal de búsqueda de referencia parametrizada, controla el accionamiento con marcha lenta en el sentido de la búsqueda de referencia. Éste se determina a partir de los parámetros "Señal de búsqueda de referencia" y "Sensor punto de referencia".

	Sensor de punto de referencia: Leva de reducción en sentido negativo	Sensor de punto de referencia: Leva de reducción en sentido positivo	Sensor de punto de referencia: Final de carrera negativo	Sensor de punto de referencia: Final de carrera positivo
Señal de búsqueda de referencia: Sensor punto de referencia y señal cero	Sentido de búsqueda de referencia -	Sentido de búsqueda de referencia +	Sentido de búsqueda de referencia +	Sentido de búsqueda de referencia -
Señal de búsqueda de referencia: Sensor de referencia				
Señal de búsqueda de referencia: señal cero	El sentido de búsqueda de referencia no está definido. Con la siguiente señal cero se sincroniza el eje.			

Tras rebasar la señal de búsqueda de referencia el eje está sincronizado. El 1PosU activa la señal de respuesta SYNC = 1 y asigna la coordenada del punto de referencia al valor real.

La figura siguiente muestra el desarrollo del modo de movimiento "Búsqueda del punto de referencia".



Señales de control

La siguiente tabla muestra los datos relevantes para la función "Búsqueda del punto de referencia" en la interfaz de control (Página 11158).

Dirección	Asignación					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	0	1	1	MODE 3 = Búsqueda del punto de referencia
	Bit 3: SPEED (SPEED = 0 es marcha lenta; SPEED = 1 es marcha rápida) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Bytes 1 a 3	Coordenada del punto de referencia: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … Final del eje rotativo – 1					

Señales de realimentación

La siguiente tabla muestra los datos relevantes de la "Búsqueda del punto de referencia" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Bytes 1 a 3	Valor real: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … Final del eje rotativo – 1

Causas del error para POS_ERR

Las causas del error deben calcularse con el JOB 15 (Mostrar valores actuales (Página 11129)).

La tabla siguiente muestra los números relevantes en la función "Búsqueda del punto de referencia", con la causa del error y las posibles soluciones.

Número de error	Causa del error	Solución
1	MODO no permitido en el modo de dosificación	
2	Se indica ERR_2L+	Compruebe la tensión de carga (2L+) en la terminal 7.
3	Se indica ERR_ENCODER	Revise el cableado del encóder.
10	Búsqueda del punto de referencia: Coordenada del punto de referencia ≥ Final del eje rotativo	
11	No se ha encontrado ninguna señal de búsqueda de referencia hasta el final de carrera o entre los finales de carrera	Revise los interruptores, el encóder y el cableado.
13	El sentido de giro del accionamiento y del encóder es diferente	Revise el cableado del accionamiento y el encóder, así como el parámetro "Inversión del sentido de giro (Página 11141)".

Desarrollo de una búsqueda del punto de referencia

Proceso de una búsqueda del punto de referencia dependiendo de la parametrización y la posición de inicio

Proceso de una búsqueda del punto de referencia dependiendo de la parametrización y la posición de inicio

En una búsqueda del punto de referencia es preciso distinguir diversos casos que dependen

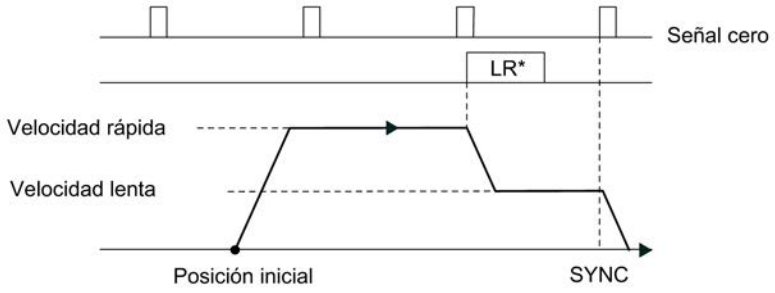
- de la posición del accionamiento al iniciar una búsqueda del punto de referencia,
- del sentido de arranque parametrizado,
- de la señal de búsqueda de referencia parametrizada y
- del sensor del punto de referencia parametrizado.

Ejemplo 1: Búsqueda del punto de referencia con leva de reducción y marca cero

Posición de inicio y parametrización:

- Posición de inicio: entre final de carrera negativo y leva de reducción
- Sentido de arranque: Más +
- Señal de búsqueda de referencia: Sensor punto de referencia y señal cero
- Sensor punto de referencia: leva reductora hacia +

La siguiente figura muestra una búsqueda del punto de referencia con leva de referencia y marca cero.



*LR = Leva reductora

La sincronización también puede realizarse en la leva de reducción sin marca cero.

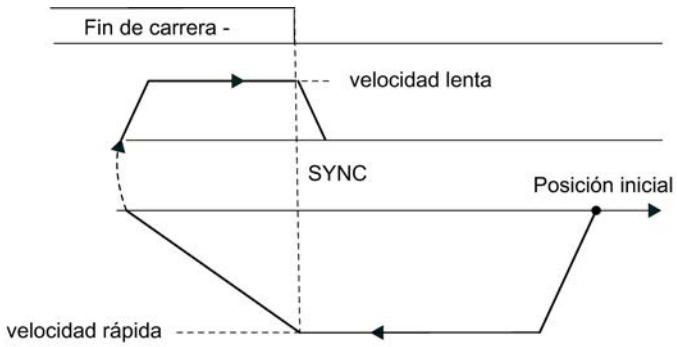
Si la posición de inicio se encuentra en la leva de reducción, el 1PosU controla el accionamiento con marcha lenta directamente en el sentido de la búsqueda de referencia.

Ejemplo 2: Búsqueda del punto de referencia con final de carrera negativo

Posición de inicio y parametrización:

- Posición de inicio: entre el final de carrera negativo y el final de carrera positivo
- Sentido de arranque: Negativo
- Señal de búsqueda de referencia: Sensor punto de referencia
- Sensor punto de referencia: Final de carrera -

La figura siguiente muestra una búsqueda del punto de referencia con final de carrera negativo.



La sincronización también puede realizarse en el final de carrera con la siguiente marca cero.

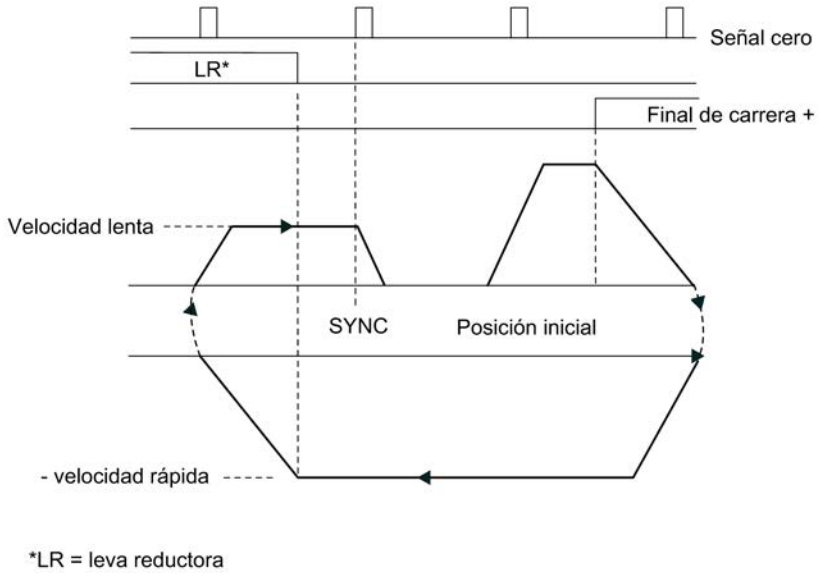
Si la posición de inicio se encuentra en el final de carrera, el 1PosU controla el accionamiento con marcha lenta directamente en el sentido de la búsqueda de referencia.

Ejemplo 3: Búsqueda del punto de referencia con inversión del sentido en el final de carrera positivo

Posición de inicio y parametrización:

- Posición de inicio: entre final de carrera negativo y leva de reducción
- Sentido de arranque: Más +
- Señal de búsqueda de referencia: Sensor punto de referencia y señal cero
- Sensor punto de referencia: leva reductora hacia +

La siguiente figura muestra una búsqueda del punto de referencia con inversión de sentido en el final de carrera positivo.



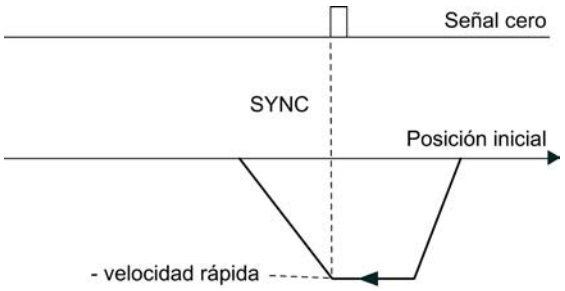
Si la posición de inicio se encuentra en el final de carrera positivo, el 1PosU controla el accionamiento con marcha rápida directamente en contra del sentido de arranque parametrizado.

Ejemplo 4: Búsqueda del punto de referencia sólo con marca cero

Posición de inicio y parametrización:

- Posición de inicio: entre el final de carrera negativo y el final de carrera positivo
- Sentido de arranque: Negativo
- Señal de búsqueda de referencia: Señal cero
- Sensor punto de referencia: irrelevante

La figura siguiente muestra una búsqueda del punto de referencia sólo con marca cero.



Posicionamiento relativo (MODE 4)

Posicionamiento relativo

Con el "Posicionamiento relativo", el 1PosU controla el accionamiento desde la posición de inicio una parte del trayecto predeterminado en un sentido predeterminado.

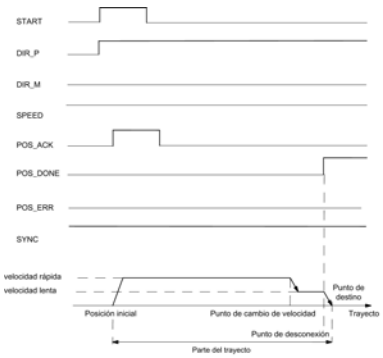
Alimente la interfaz de control (Página 11158) con la parte del trayecto e inicie el MODE 4 con la especificación de sentido (DIR_M o DIR_P). El 1PosU mueve el accionamiento con la velocidad predeterminada (bit de control SPEED) en torno a la parte del trayecto. En el punto de conmutación el 1PosU pasa de marcha rápida a marcha lenta y, en el punto de desconexión, el 1PosU finaliza la marcha.

Si realiza el arranque durante una marcha en curso, el 1PosU ejecuta un cambio de sentido (Página 11146) necesario una vez transcurrido el tiempo T_{min}.

El 1PosU no comprueba la parte del recorrido predeterminada. De esta manera, en el caso de ejes rotativos puede moverse más de una vuelta.

En el modo de dosificación (Página 11094) el "Posicionamiento relativo" sólo es posible en el sentido positivo. A este respecto, en cada inicio el valor real se establece a 0.

La figura siguiente muestra el desarrollo del modo de movimiento "Posicionamiento relativo".



Señales de control

La siguiente tabla muestra los datos relevantes para el "Posicionamiento relativo" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	1	0	0	MODE 4 = Posicionamiento relativo
	Bit 3: SPEED (SPEED = 0 es marcha lenta; SPEED = 1 es marcha rápida) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Bytes 1 a 3	Parte del trayecto: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … 16 777 215					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para el "Posicionamiento relativo" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Bytes 1 a 3	Valor real: • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1

Causas del error para POS_ERR

Las causas del error deben calcularse con el dem JOB 15 (Mostrar valores actuales (Página 11129)).

La tabla siguiente muestra los números relevantes en la función "Posicionamiento relativo", con la causa del error y las posibles soluciones.

Número de error.	Causa del error	Solución
2	Se indica ERR_2L+	Compruebe la tensión de carga (2L+) en la terminal 7.
3	Se indica ERR_ENCODER	Revise el cableado del encóder.
5	El final de carrera en cuyo sentido se controla el accionamiento está activo	Revise los encóders y el cableado, así como los parámetros "DI0 Final de carrera - (Página 11142)" y "DI1 Final de carrera + (Página 11142)".
7	Posicionamiento relativo: Inicio con DIR_P = 0 y DIR_M = 0 o DIR_P = 1 y DIR_M = 1	
13	El sentido de giro del accionamiento y del encóder es diferente	Revise el cableado del accionamiento y el encóder, así como el parámetro "Inversión del sentido de giro (Página 11141)".
15	En el modo de dosificación: DIR_M = 1	

Posicionamiento absoluto (MODE 5)

Posicionamiento absoluto

Con el "Posicionamiento absoluto" el 1PosU mueve el accionamiento hacia destinos absolutos. Para ello el eje debe estar sincronizado.

El "Posicionamiento absoluto" no es posible en el servicio de dosificación activado.

Alimente la interfaz de control (Página 11158) con el destino e inicie el MODE 5 con lahabilitación dependiente del sentido (Página 11100) que sea necesaria (DIR_M, DIR_P). El 1PosU mueve el accionamiento con la velocidad predeterminada (bit de control SPEED) hacia el destino. En el punto de conmutación el 1PosU pasa de marcha rápida a marcha lenta y, en el punto de desconexión, el 1PosU finaliza la marcha.

Si realiza el arranque durante una marcha en curso, el 1PosU ejecuta un cambio de sentido (Página 11146) necesario una vez transcurrido el tiempo T_{min}.

Eje lineal

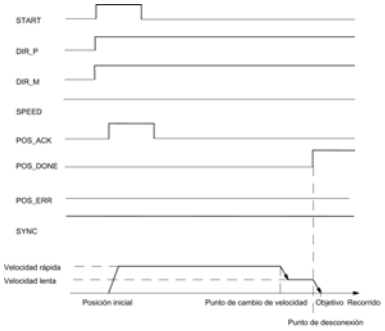
El 1PosU determina el sentido en el que se avanza hacia el punto de destino. En el inicio debe activar la habilitación necesaria en función del sentido (DIR_M, DIR_P). También puede activar ambas habilitaciones.

Eje rotativo

El sentido en el que se avanza hacia el destino se determina mediante la selección de la habilitación en función del sentido (DIR_M, DIR_P):

Bits de control DIR_P y DIR_M	Sentido
DIR_P = 1 DIR_M = 0	Se avanza al punto de destino en sentido +.
DIR_P = 0 DIR_M = 1	Se avanza al punto de destino en sentido -.
DIR_P = 1 DIR_M = 1	Se llega al punto de destino por el trayecto más corto. El 1PosU determina el sentido en el que se avanza hacia el punto de destino. Si el trayecto que debe recorrerse es más pequeño que la distancia de desconexión, no se inicia ninguna marcha (POS_DONE = 1).

La siguiente figura muestra el desarrollo del modo de movimiento "Posicionamiento absoluto".



Señales de control

La siguiente tabla muestra los datos relevantes para el "Posicionamiento absoluto" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 0	Bit 0.7 … 0.4:					
	Bit	7	6	5	4	
		0	1	0	1	MODE 5 = Posicionamiento absoluto
	Bit 3: SPEED (SPEED = 0 es marcha lenta; SPEED = 1 es marcha rápida) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Bytes 1 a 3	Objetivo: • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para el "Posicionamiento absoluto" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Bytes 1 a 3	Valor real: • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1

Causas del error para POS_ERR

Las causas del error deben calcularse con el JOB 15 (Mostrar valores actuales (Página 11129)).

La tabla siguiente muestra los números relevantes en la función "Posicionamiento absoluto", con la causa del error y las posibles soluciones.

Número de error.	Causa del error	Solución
1	MODO no permitido en el modo de dosificación	
2	Se indica ERR_2L+	Compruebe la tensión de carga (2L +) en la terminal 7.
3	Se indica ERR_ENCODER	Revise el cableado del encóder.
4	El eje no está sincronizado (SYNC = 0).	Puede sincronizar el eje: • Con búsqueda del punto de referencia • Evaluar señal de búsqueda de referencia • Activar valor real
5	El final de carrera en cuyo sentido se controla el accionamiento está activo	Revise los encóders y su cableado, así como los parámetros "DI0 Final de carrera - (Página 11142)" y "DI1 Final de carrera + (Página 11142)".
7	Inicio con DIR_P = 0 y DIR_M = 0 o bit de control relevante DIR_P = 0 o DIR_M = 0	
8	Destino ≥ Final del eje rotativo (en encóder incremental) o Destino ≥ Zona de alcance del encóder (en el caso de encóder SSI)	

Número de error.	Causa del error	Solución
9	Posicionamiento absoluto cancelado porque se ha iniciado JOB 9 (sólo en encóders incrementales)	
13	El sentido de giro del accionamiento y del encóder es diferente	Revise el cableado del accionamiento y el encóder, así como el parámetro "Inversión del sentido de giro (Página 11141)".

Cancelar procesamiento de JOB (JOB 0)

Cancelar procesamiento JOB

Si activa JOB 0, el 1PosU reacciona de la siguiente manera:

- se cancela cualquier proceso JOB 9 "Evaluar señal de búsqueda de referencia" que esté en marcha,
- se cancela cualquier proceso JOB 10 "Función de congelación" que esté en marcha,
- activa un JOB_ERR = 0 pendiente.

El JOB 0 puede activarse en cualquier estado del eje.

Efecto sobre los MODEs

Los MODEs no dependen del JOB 0.

Señales de control

La siguiente tabla muestra los datos relevantes para "Cancelar procesamiento JOB" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		0	0	0	0	JOB 0 = Cancelar procesamiento JOB
	Bit 0: JOB_REQ					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para "Cancelar procesamiento JOB" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Ajustar valor real (JOB 1)

Ajustar valor real

La función "Ajustar valor real" asigna una nueva coordenada al valor real mostrado. De esta manera la zona de trabajo se desplaza a otra zona del eje.

El eje se sincroniza en el caso del encóder incremental y si el modo de dosificación no está activado.

Alimente la interfaz de control (Página 11158) con la nueva coordenada de valor real y active JOB 1.

El 1PosU activa la coordenada de valor real predeterminada en el valor real mostrado en lainterfaz de respuesta (Página 11159) y activa el bit de respuesta SYNC = 1.

Efecto sobre los MODEs

La tabla siguiente muestra los efectos del JOB 1 sobre los diferentes MODEs.

MODE	¿Qué ocurre?
Búsqueda del punto de referencia	Tenga en cuenta que, en el caso del encóder incremental y el modo de dosificación no está activado durante la evaluación de la búsqueda del punto de referencia, el bit de respuesta SYNC = 1 se activa de inmediato. La búsqueda del punto de referencia sigue transcurriendo.
Jog	-
Posicionamiento absoluto	Son posibles las reacciones siguientes: • Distancia hasta el destino ≤ Distancia de desconexión El punto de desconexión se ha alcanzado o sobrepasado; se desconecta directamente y la marcha finaliza con POS_DONE = 1. En este caso es posible que se haya sobrepasado el punto de destino. • Distancia hasta el destino ≤ Distancia de cambio de velocidad El punto de conmutación se ha alcanzado o sobrepasado, se reduce directamente de velocidad rápida a velocidad lenta. En este caso el trayecto recorrido en velocidad lenta es inferior que (distancia de cambio de velocidad - distancia de desconexión). • Distancia hasta el destino > Distancia de cambio de velocidad El accionamiento se controla a marcha rápida aunque antes hubiera cambiado a marcha lenta.
Posicionamiento relativo	La parte especificada del trayecto se sigue recorriendo.

Señales de control

La siguiente tabla muestra los datos relevantes para "Ajustar el valor real" en la interfaz de control (Página 11158).

Dirección	Ocupación				
Byte 4	Bit 4.7 … 4.4:				
	Bit	7	6	5	4

Dirección	Ocupación					
		0	0	0	1	JOB 1 = Ajustar valor real
	Bit 0: JOB_REQ					
Bytes 5 a 7	Coordenadas del valor real: • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para "Ajustar el valor real" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 3: SYNC
Bytes 1 a 3	Valor real: • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Causas del error para JOB_ERR

La tabla siguiente muestra los números relevantes en la función "Ajustar valor real", con la causa del error y las posibles soluciones.

Número de error.	Significado	Solución
23	Se indica ERR_ENCODER	Revise el cableado del encóder.
34	Coordenada del valor real ≥ Final del eje rotativo (en encóders incrementales) o bien Coordenada de valor real ≥ Zona del encóder (en encóders SSI)	

Mover zona de alcance del encóder (JOB 2)

Mover zona de alcance del encóder

La función "Mover zona de alcance del encóder" sólo se puede ejecutar con encóders SSI.

La función "Mover zona de alcance del encóder" ajusta el valor del encóder de tal modo que el valor real mostrado corresponde al valor real actual. Para ello la marcha en curso debe haber finalizado.

Alimente la interfaz de control (Página 11158) con el offset y active el JOB 2.

El offset se determina tal como se indica a continuación:

- Offset = valor real mostrado – valor real actual

Si aparece un offset negativo, calcule lo siguiente:

- Offset = valor real mostrado – valor real actual +
  (número de vueltas × número de pasos)

El 1PosU realiza el offset predeterminado y muestra la coordenada actual del valor real en la interfaz de respuesta (Página 11159).

Efecto sobre los MODEs

Los MODEs no dependen del JOB 2.

Señales de control

La siguiente tabla muestra los datos relevantes para el "Desplazamiento de la zona del encóder" en la interfaz de control (Página 11158).

<table>
<tr><th>Dirección</th><th colspan="6">Ocupación</th></tr>
<tr><td rowspan="3">Byte 4</td><td colspan="6">Bit 4.7 … 4.4:</td></tr>
<tr><td rowspan="2">Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td rowspan="2">JOB 2 = Desplazar zona del encóder</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td colspan="6">Bit 0: JOB_REQ</td></tr>
<tr><td>Bytes 5 a 7</td><td colspan="6">Offset
(0...zona del encóder)</td></tr>
</table>

Señales de realimentación

La siguiente tabla muestra los datos relevantes para el "Desplazamiento de la zona del encóder" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Bytes 1 a 3	Valor real: • 0 … Zona del encóder – 1
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Causas del error para JOB_ERR

La tabla siguiente muestra los números relevantes en la función "Desplazar zona del encóder", con la causa del error y las posibles soluciones.

Número de error.	Significado	Solución
21	JOB no permitido en el caso de encóders incrementales	
23	Se indica ERR_ENCODER	Revise el cableado del encóder.
26	JOB 2 "Desplazar zona del encóder" no puede iniciarse porque hay una marcha en curso	
33	en el caso de JOB 2: Offset no en la zona de alcance del encóder	

Modificar la distancia de desconexión (JOB 3)

Modificar distancia de desconexión

El JOB "Modificar la distancia de desconexión" adapta el control del accionamiento a diferentes condiciones de carga y mecánicas.

Alimente la interfaz de control (Página 11158) con la nueva distancia de desconexión y active JOB 3.

El 1PosU aplica la nueva distancia de desconexión.

La distancia de desconexión continúa siendo válida hasta una nueva parametrización (Página 11138) del 1PosU con parámetros modificados.

Efecto sobre los MODEs

La tabla siguiente muestra los efectos del JOB 3 sobre los diferentes MODEs.

MODE	¿Qué ocurre?
Búsqueda del punto de referencia	-
JOG	
Posicionamiento absoluto	Distancia hasta el destino ≤ Distancia de desconexión El punto de desconexión se ha alcanzado o sobrepasado; se desconecta directamente y la marcha finaliza con POS_DONE = 1. En este caso es posible que se haya sobrepasado el punto de destino.
Posicionamiento relativo	

Señales de control

La siguiente tabla muestra los datos relevantes para "Modificar la distancia de desconexión" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	JOB 3 = Modificar distancia de desconexión
		0	0	1	1	
	Bit 0: JOB_REQ					
Bytes 5 a 7	Distancia de desconexión: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … 16 777 215					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para "Modificar la distancia de desconexión" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 4	Bit 0: JOB_ACK

Modificar la distancia de cambio de velocidad (JOB 4)

Modificar distancia de cambio de velocidad

La función "Modificar la distancia de cambio de velocidad" adapta el control del accionamiento a diferentes condiciones de carga y mecánicas.

Alimente la interfaz de control (Página 11158) con la nueva distancia de cambio de velocidad y active JOB 4.

El 1PosU aplica la nueva distancia de cambio de velocidad.

La distancia de cambio de velocidad continúa siendo válida hasta una nueva parametrización (Página 11138) del 1PosU con parámetros modificados.

Efecto sobre los MODEs

La tabla siguiente muestra los efectos del JOB 4 sobre los diferentes MODEs.

MODE	¿Qué ocurre?
Búsqueda del punto de referencia	-
JOG	

MODE	¿Qué ocurre?
Posicionamiento absoluto	Son posibles las reacciones siguientes: • Distancia hasta el punto de destino ≤Distancia de cambio de velocidad Se ha alcanzado o sobrepasado el punto de cambio de velocidad, se reduce directamente de velocidad rápida a velocidad lenta. En este caso el trayecto recorrido en velocidad lenta es inferior que (distancia de cambio de velocidad - distancia de desconexión). • Distancia hasta el destino >Distancia de cambio de velocidad El accionamiento se controla a marcha rápida aunque antes hubiera pasado a velocidad lenta.
Posicionamiento relativo	

Señales de control

La siguiente tabla muestra los datos relevantes para "Modificar la distancia de cambio de velocidad" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	JOB 4 = Modificar distancia de cambio de velocidad
		0	1	0	0	
	Bit 0: JOB_REQ					
Bytes 5 a 7	Distancia de cambio de velocidad: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … 16 777 215					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para "Modificar la distancia de cambio de velocidad" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 4	Bit 0: JOB_ACK

Evaluar señal de búsqueda de referencia (JOB 9)

Evaluar señal de búsqueda de referencia

La función de "Evaluar señal de búsqueda de referencia" sólo está disponible con encóders incrementales y si el modo de dosificación no está activado.

"Evaluar señal de búsqueda de referencia" sincroniza el eje a partir de una señal externa de búsqueda de referencia durante una marcha en curso en los MODEs "Jog" y "Posicionamiento relativo". Como señal de búsqueda de referencia puede escoger entre 3 entradas digitales y la señal cero (Página 11106).

Las entradas digitales DI0 (final de carrera -), DI1 (final de carrera +) y DI2 (leva reductora) se pueden parametrizar como contacto normalmente cerrado o contacto normalmente abierto.

Alimente la interfaz de control (Página 11158) con la coordenada del punto de referencia y active JOB 9. El 1PosU pone la señal de realimentación SYNC = 0.

Cuando el 1PosU detecta que se sobrepasa la señal parametrizada de búsqueda de referencia en el sentido de búsqueda, el eje está sincronizado. El 1PosU pone la señal de realimentación SYNC = 1 y asigna la coordenada del punto de referencia al valor real.

Éste se determina a partir de los parámetros "Señal de búsqueda de referencia" y "Sensor de punto de referencia".

	Sensor de punto de referencia: Leva de reducción en el sentido negativo	**Sensor de punto de referencia: Leva de reducción en el sentido positivo**	**Sensor de punto de referencia: Final de carrera negativo**	**Sensor de punto de referencia: Final de carrera positivo**
Señal de búsqueda de referencia: Sensor punto de referencia y señal cero	Sentido de búsqueda de referencia -	Sentido de búsqueda de referencia +	Sentido de búsqueda de referencia +	Sentido de búsqueda de referencia -
Señal de búsqueda de referencia: sensor de referencia				
Señal de búsqueda de referencia: señal cero	El sentido de búsqueda de referencia no está definido. Con la siguiente señal cero se sincroniza el eje.			

Efecto sobre los MODEs

La tabla siguiente muestra los efectos del JOB 9 sobre los diferentes MODEs.

MODE	**¿Qué ocurre?**
Búsqueda del punto de referencia	La coordenada de búsqueda de referencia indicada con el JOB 9 es válida
JOG	-
Posicionamiento absoluto	Cancelación de la marcha con POS_ERR = 1, pues SYNC se ha borrado
Posicionamiento relativo	-

Señales de control

La siguiente tabla muestra los datos relevantes para la "evaluación de la señal de búsqueda de referencia" en la interfaz de control (Página 11158).

Dirección	**Ocupación**				
Byte 4	Bit 4.7 … 4.4:				
	Bit	7	6	5	4

Dirección	Ocupación					
		1	0	0	1	JOB 9 = Evaluar señal de búsqueda de referencia
	Bit 0: JOB_REQ					
Bytes 5 a 7	Coordenada del punto de referencia: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … Final del eje rotativo – 1					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para "Evaluar la señal de búsqueda de referencia" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 0	Bit 3: SYNC
Bytes 1 a 3	Valor real: • Eje lineal: 0 … 16 777 215 • Eje rotativo: 0 … Final del eje rotativo – 1
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Causas del error para JOB_ERR

La tabla siguiente muestra los números relevantes en la función "Evaluar señal de búsqueda de referencia", con la causa del error y las posibles soluciones.

Número de error.	Significado	Solución
21	JOB no permitido en encóder SSI o en modo de dosificación	
23	Se indica ERR_ENCODER	Revise el cableado del encóder.
30	Coordenada del punto de referencia ≥ Final del eje rotativo	

Función de congelación (JOB 10)

Función de congelación

Con la función de congelación, el valor real se almacena una vez en un flanco de la entrada digital DI2. Esta función puede utilizarse, por ejemplo, para la captura de bordes o la medición longitudinal.

Alimente la interfaz de control (Página 11158) con el flanco deseado y active el JOB 10.

Si el 1PosU detecta el flanco predeterminado en la entrada digital DI2, almacena el valor real correspondiente, lo muestra como valor de respuesta (Página 11159) y activa el bit de respuesta LATCH_DONE = 1.

A continuación es posible volver a activar la "Función de congelación".

Función de congelación y búsqueda del punto de referencia o "Evaluar señal de búsqueda de referencia"

Si el 1PosU realiza la sincronización en el mismo flanco, almacena el valor real antes de asignar la coordenada del punto de referencia.

Efecto sobre los MODEs

Los MODEs no dependen del JOB 10.

Señales de control

La siguiente tabla muestra los datos relevantes para la "Función de congelación" en la interfaz de control (Página 11158).

<table>
<tr><th>Dirección</th><th colspan="6">Ocupación</th></tr>
<tr><td rowspan="3">Byte 4</td><td colspan="6">Bit 4.7 … 4.4:</td></tr>
<tr><td rowspan="2">Bit</td><td>7</td><td>6</td><td>5</td><td>4</td><td rowspan="2">JOB 10 = Función de congelación</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td></tr>
<tr><td></td><td colspan="6">Bit 0: JOB_REQ</td></tr>
<tr><td>Byte 5</td><td colspan="6">Bit 1: Congelación con flanco descendente en DI2
Bit 0: Congelación con flanco ascendente en DI2</td></tr>
</table>

Señales de realimentación

La siguiente tabla muestra los datos relevantes para la "Función de congelación" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 4	Bit 2: LATCH_DONE Bit 1: JOB_ERR Bit 0: JOB_ACK
Bytes 5 a 7	Valor de respuesta: Valor real en el flanco de DI2 • en el caso de encóders incrementales: – Eje lineal: 0 … 16 777 215 – Eje rotativo: 0 … Final del eje rotativo – 1 • con encóders SSI: – 0 … Zona del encóder – 1

Causas del error para JOB_ERR

La tabla siguiente muestra los números relevantes en la "Función de congelación" con la causa del error y las posibles soluciones.

Número de error.	Significado	Solución
23	Se indica ERR_ENCODER	Revise el cableado del encóder.
36	Selección de flanco desconocida	

Ajustar vigilancia de sentido de giro (JOB 11)

Ajustar vigilancia de sentido de giro

La función de "Ajustar vigilancia de sentido de giro" no está disponible en el modo de dosificación.

"Ajustar vigilancia de sentido de giro" adapta la vigilancia de sentido de giro del 1PosU a sus condiciones de carga y mecánicas.

La vigilancia de sentido de giro siempre está activada. El 1PosU detecta si el sentido de giro del accionamiento y del encóder es el mismo. Si no lo es, se aplica una tolerancia hasta la distancia de trayecto predefinida para la vigilancia de sentido de giro. Si la distancia de trayecto predefinida se sobrepasa, el 1PosU indica POS_ERR = 1.

Hasta que no se activa el JOB 11, se utiliza como distancia de trayecto para al vigilancia de sentido de giro la distancia de desconexión doble resultante de los parámetros. JOB 3 "Modificar distancia de desconexión" afecta a la distancia de trayecto para la vigilancia del sentido de giro.

Alimente la interfaz de control (Página 11158) con la distancia de trayecto y active JOB 11.

El 1PosU aplica la distancia de trayecto predefinida para la vigilancia de sentido de giro.

La distancia de trayecto para el sentido de giro continúa siendo válida hasta una nueva parametrización (Página 11138) del 1PosU con parámetros modificados.

Desconectar la vigilancia de sentido de giro

Con una distancia de trayecto 0, la vigilancia de sentido de giro está desactivada.

Efecto sobre los MODEs

Los MODEs no dependen del JOB 11.

Señales de control

La siguiente tabla muestra los datos relevantes para el "ajuste de la vigilancia del sentido de giro" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		1	0	1	1	JOB 11 = Ajustar vigilancia de sentido de giro
	Bit 0: JOB_REQ					
Byte 5	0					
Byte 6, 7	Distancia de trayecto para vigilancia de sentido de giro: • 0 … 65 535					

Señales de realimentación

La siguiente tabla muestra los datos relevantes para el "Ajuste de la vigilancia del sentido de giro" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK

Causas del error para JOB_ERR

La tabla siguiente muestra los números relevantes en la función de "Ajuste de la vigilancia del sentido de giro", con la causa del error y las posibles soluciones.

Número de error.	Significado	Solución
21	JOB no admisible en el modo de dosificación	
38	Supervisión de sentido de giro distancia de trayecto > 65 535	

Mostrar valores actuales (JOB 15)

Mostrar valores actuales

Puede visualizar los siguientes valores en la interfaz de realimentación en forma de valores de respuesta:

- Distancia residual
- Velocidad real
- Causas del error para POS_ERR y JOB_ERR

El 1PosU tiene predefinido como valor de respuesta la distancia residual.

Independientemente del valor de respuesta seleccionado, el 1PosU muestra el valor real correspondiente de forma continua en la interfaz de respuesta.

Alimente la interfaz de control (Página 11158) con el valor de respuesta deseado y active el JOB 15.

El valor de respuesta seleccionado continúa siendo válida hasta una nueva parametrización (Página 11138) del 1PosU con parámetros modificados.

"Mostrar valores actuales" y función congelar

Al activar la función congelar, el 1PosU activa el valor de realimentación = 0 y muestra el valor real correspondiente al flanco almacenado en la entrada digital DI2.

El JOB 15 no se puede volver a activar hasta que no haya finalizado la función de congelación.

Distancia residual

El 1PosU calcula la distancia hasta el punto de destino como distancia residual en el MODE "Posicionamiento absoluto" y "Posicionamiento relativo". El trayecto residual es positivo siempre que el valor real se encuentre antes del punto de destino. Pasa a ser negativo cuando se sobrepasa el punto de destino. En los MODEs restantes, la distancia residual es = 0.

El 1PosU indica la distancia residual con signo entre -8 388 608 y 8 388 607 pasos. Los valores negativos se representan en complemento doble. Si el trayecto residual se encuentra fuera de estos límites, se indica el valor límite correspondiente.

Velocidad real

El 1PosU calcula la velocidad real como modificación del valor del encóder en pasos de 10 ms. Indica la velocidad entre 0 y 16 777 215.

Causas del error para POS_ERR y JOB_ERR

El 1PosU muestra las causas del error de POS_ERR y JOB_ERR, así como el MODE y el JOB introducidos en lainterfaz de control (Página 11158) .

Efecto sobre los MODEs

Los MODEs no dependen del JOB 15.

Señales de control

La siguiente tabla muestra los datos relevantes para "Mostrar valores actuales" en la interfaz de control (Página 11158).

Dirección	Ocupación					
Byte 4	Bit 4.7 … 4.4:					
	Bit	7	6	5	4	
		1	1	1	1	JOB 15 = Mostrar valores actuales

Dirección	Ocupación
	Bit 0: JOB_REQ
Byte 5	0: Distancia residual 1: Velocidad real 2: Causas del error para POS_ERR y JOB_ERR

Señales de realimentación

La siguiente tabla muestra los datos relevantes para la "presentación de valores actuales" en la interfaz de respuesta (Página 11159).

Dirección	Ocupación
Byte 4	Bit 1: JOB_ERR Bit 0: JOB_ACK
Bytes 5 a 7	En función del valor de respuesta seleccionado: • Para trayecto residual: –8 388 608 … 8 388 607 • Para velocidad real: 0 … 16 777 215 • en el caso de las causas del error para POS_ERR y JOB_ERR – Byte 5: Causas del error (Página 11136) de POS_ERR – Byte 6: Causas del error (Página 11137) de JOB_ERR – Bit 7.3 … 7.0: MODE (= Bit 0.7 … 0.4 de las señales de control) – Bit 7.7 … 7.4: JOB (= Bit 4.7 … 4.4 de las señales de control)

Causas del error para JOB_ERR

La tabla siguiente muestra los números relevantes en la función "Mostrar valores actuales", con la causa del error y las posibles soluciones.

Número de error.	Significado	Solución
35	Mostrar valores actuales: Selección desconocida	
37	Mostrar valores actuales: El JOB 15 no puede activarse con la función congelar en marcha.	

Detección de errores / Diagnóstico

Vista general de los errores con 1PosU

Al utilizar el 1PosU pueden producirse diversos errores. En este caso se distinguen los siguientes tipos de error:

- Errores de parametrización (Página 11132)
- Fallos externos (Página 11133)
- Errores en el control de MODEs y JOBs (Página 11135)

Los errores de "falta de tensión de carga 2L+", "cortocircuito alimentación del encóder" y "rotura de hilo/cortocircuito de las señales del encóder" deben acusarse (Página 11135).

Si surgen errores al iniciar MODEs o activar JOBs, aquí se muestra un resumen:

- Causas de error de POS_ERR (Página 11136)
- Causas de error de JOB_ERR (Página 11137)

Errores de parametrización

Errores de parametrización

La tabla siguiente indica los errores de parametrización con el 1PosU.

Errores de parametrización	Reacción del 1PosU
Causas: • El 1PosU no puede identificar como propios los siguientes parámetros. • El slot del 1PosU que se ha configurado no coincide con la configuración real. Sólo con encóders SSI: • Valor no permitido en el parámetro "Impulsos por vuelta del encóder". • Valor no permitido en el parámetro "Vueltas". • Impulsos por vuelta del encóder × vueltas es mayor que 4096 × 4096. Sólo en modo de dosificación: • "Diagnóstico señales de encóder" activado • "Inversión del sentido de giro" activada Solución: • Revise la configuración y la configuración real.	• El 1PosU no está parametrizado y no puede ejecutar sus funciones • Generar un diagnóstico de canal

Errores externos

Fallos externos

La tabla siguiente indica los fallos externos con el 1PosU.

Tensión de carga 2L+ no existe	Reacción del 1PosU
Causas: • Falta tensión de carga 2L+ o es insuficiente en el borne 7 Solución: • Compruebe el cableado. Elimine el cortocircuito que pueda existir. • Acuse (Página 11135) el error con el bit de control EXTF_ACK.	• El desplazamiento en curso se detiene; no es posible iniciar otro desplazamiento: – Las 3 salidas digitales pasan a 0. – Bit de respuesta POS_ERR = 1 – Bit de respuesta POS_DONE = 0 • Bit de respuesta ERR_2L+ = 1 • Generar un diagnóstico de canal • espera a acuse de error EXTF_ACK
Cortocircuito en alimentación del encóder	**Reacción del 1PosU**
Causas: • Cortocircuito en la alimentación del encóder dispone en los bornes 6 y 10 Solución: • Revise el cableado y elimine el cortocircuito. • Acuse (Página 11135) el error con el bit de control EXTF_ACK.	• Los MODEs en ejecución "Búsqueda del punto de referencia", "Posicionamiento relativo" y "Posicionamiento absoluto" se detienen. No es posible volver a iniciar un desplazamiento en estos MODEs: – Las 3 salidas digitales pasan a 0. – Bit de respuesta POS_ERR = 1 – Bit de respuesta POS_DONE = 0 • Bit de respuesta ERR_ENCODER = 1 • Bit de respuesta SYNC = 0 • Generar un diagnóstico de canal • espera a acuse de error EXTF_ACK • El MODE "Jog" no se ve afectado por este error. • El JOB activo "Evaluar señal de búsqueda de referencia" se interrumpe.

Rotura de hilo/cortocircuito señales de encóder	Reacción del 1PosU
En caso de utilizar un encóder SSI, rige lo siguiente: Requisitos: • Para la detección de errores en las señales del encóder es preciso habilitar el parámetro "Diagnóstico de señales de encóder". Causas: • Rotura de hilo o cortocircuito de las señales del encóder en los bornes 9 y 13 o 12 y 16. • Los parámetros correspondientes al encóder SSI no concuerdan con el encóder conectado. Solución: • Revise el cableado y elimine el cortocircuito. • Compare la parametrización con las especificaciones técnicas del encóder. • Acuse (Página 11135) el error con el bit de control EXTF_ACK. **Si se utiliza un encóder incremental rige lo siguiente:** Requisitos: • Para la detección de fallos de las señales A, /A y B, /B con señales diferenciales 5V o de las señales A* y B* con señales de 24 V, es preciso habilitar el parámetro "Diagnóstico de señales de encóder". • Para la detección de fallos de las señales N, /N con señales diferenciales 5 V es preciso habilitar el parámetro "Diagnóstico de señales de cero". Si se utiliza un encóder sin marca cero, la detección de fallos se desconecta. Si está activado el modo de dosificación no será posible el "Diagnóstico de señales de cero". Causas: • Sólo para señales diferenciales 5 V: rotura de hilo o cortocircuito de las señales de encóder en los bornes 9 y 13 u 11 y 15 ó 12 y 16. • Detectado error de flanco de la señal de encóder de forma que el 1PosU no puede detectar el sentido con claridad. Solución: • Revise el cableado y elimine el cortocircuito. • Acuse (Página 11135) el error con el bit de control EXTF_ACK.	• Los MODEs en ejecución "Posicionamiento relativo" y "Posicionamiento absoluto" se detienen. No es posible volver a iniciar una marcha en estos MODEs. – Las 3 salidas digitales pasan a 0. – Bit de respuesta POS_ERR = 1 – Bit de respuesta POS_DONE = 0 • Bit de respuesta ERR_ENCODER = 1 • Bit de respuesta SYNC = 0 • Generar un diagnóstico de canal • espera a acuse de error EXTF_ACK • El MODE "Jog" no se ve afectado por este error. • El JOB en marcha "Evaluar señal de búsqueda de referencia" se interrumpe.

Error en el control de MODEs y JOBs

Error en el control de MODEs y JOBs

La tabla siguiente muestra los errores en el control de los MODEs y JOBs con el 1PosU.

POS_ERR	Reacción del 1PosU
Causas: • Al iniciar un MODEs hay determinados requisitos o condiciones que no se cumplen.	• El MODE iniciado no se ejecuta. • La marcha en curso se detiene: – Las 3 salidas digitales pasan a 0. – Bit de respuesta POS_ERR = 1 – Bit de respuesta POS_DONE = 0
JOB_ERR	**Reacción del 1PosU**
Causas: • Al activar un JOBs hay determinados requisitos o condiciones que no se cumplen.	• El JOB activado no se ejecuta. – Bit de respuesta JOB_ERR = 1

Acuse del error EXTF_ACK

Acuse del error EXTF_ACK

Los errores externos (Página 11129) eliminados "Falta de tensión de carga 2L+", "Cortocircuito alimentación de encóders" y "Rotura de hilo/cortocircuito de las señales de encóder" deben acusarse. La siguiente tabla muestra el proceso necesario para ello

Su acción	Reacción del 1PosU
	• Bit de respuesta ERR_2L+ = 1 y/o • Bit de respuesta ERR_ENCODER = 1
El programa de control detecta el bit de respuesta activado ERR_2L+ o ERR_ENCODER. Ejecute la respuesta al error específica de la aplicación. Elimine la causa del error.	
Ponga el bit de control (Página 11158)EXTF_ACK de 0 a 1.	El 1PosU activa los bits de respuesta (Página 11159)ERR_2L + = 0 y ERR_ENCODER = 0. Eso le permite saber que el error ha sido eliminado y acusado. Si ERR_2L+ = 1 y ERR_ENCODER = 1, la causa del error todavía no se ha eliminado.
Ponga el bit de control (Página 11158)EXTF_ACK de 1 a 0.	
Con un acuse de error constante (EXTF_ACK = 1) o una parada de CPU/maestro, el 1PosU indica los errores en cuanto los detecta y los borra en cuanto se eliminan.	

Causas de error para POS_ERR

Causas del error para POS_ERR

La tabla siguiente muestra las causas de error y las soluciones posibles para POS_ERR.

Número de error	Causa del error	Solución
1	MODE desconocido o no permitido	MODEs permitidos son: • MODE 0 • MODE 1 • MODE 3 (no es posible con encóder SSI o en modo de dosificación) • MODE 4 • MODE 5 (no es posible en modo de dosificación)
2	Se indica ERR_2L+	Compruebe la tensión de carga (2L+) en terminal 7.
3	Se indica ERR_ENCODER	Revise el cableado del encóder.
4	El eje no está sincronizado (SYNC = 0).	Encóder SSI: • Elimine el error de encóder. Con encóder incremental y modo de dosificación no activado el eje puede sincronizarse con: • Búsqueda del punto de referencia • Evaluar señal de búsqueda de referencia • Ajustar valor real
5	El final de carrera en cuyo sentido se controla el accionamiento está activo	Revise los encóders y el cableado, así como los parámetros "DI0 Final de carrera - (Página 11142)" y "DI1 Final de carrera + (Página 11142)".
7	Jog: DIR_P = 1 del DIR_M = 1 Posicionamiento absoluto: Iniciar con DIR_P = 0 y DIR_M = 0 o bit de control relevante DIR_P = 0 o DIR_M = 0 Posicionamiento relativo: Inicio con DIR_P = 0 y DIR_M = 0 o DIR_P = 1 y DIR_M = 1	
8	Posicionamiento absoluto: Punto de destino ≥ fin del eje rotativo con encóder incremental o destino ≥ zona de alcance del encóder con encóder SSI	
9	Posicionamiento absoluto cancelado porque se ha iniciado JOB 9	
10	Búsqueda del punto de referencia: Coordenada del punto de referencia ≥ Final del eje rotativo	
11	Búsqueda del punto de referencia: No se ha encontrado ninguna señal de búsqueda de referencia hasta el final de carrera o entre los finales de carrera	Revise los interruptores, el encóder y el cableado.

Número de error	Causa del error	Solución
13	El sentido de giro del accionamiento y del encóder es diferente	Revise el cableado del accionamiento y el encóder, así como el parámetro "Inversión del sentido de giro (Página 11141)".
15	Jog, posicionamiento relativo: En el modo de dosificación DIR_M = 1	

Causas de error para JOB_ERR

Causas del error para JOB_ERR

La tabla siguiente muestra las causas de error y las soluciones posibles para POS_ERR.

Número de error.	Significado	Solución
21	JOB desconocido o no permitido	JOBs permitidos son: • JOB 0 • JOB 1 • JOB 2 (sólo es posible con encóder SSI) • JOB 3 • JOB 4 • JOB 9 (no es posible con encóder SSI o en modo de dosificación (Página 11094)) • JOB 10 • JOB 11 (no es posible en modo de dosificación) • JOB 15
23	Se indica ERR_ENCODER	Revise el cableado del encóder.
26	JOB 2 "Desplazar zona del encóder" no puede iniciarse porque hay una marcha en curso	
29	Evaluar señal de búsqueda de referencia: Coordenada del punto de referencia ≥ Final del eje rotativo	
34	Ajustar valor real: Coordenadas de valor real ≥ Zona de alcance del encóder	
35	Mostrar valores actuales: Selección desconocida	
36	Función congelar (latch): Selección de flanco desconocida	
37	Mostrar valores actuales: El JOB 15 no puede activarse con la función congelar en marcha.	
38	Supervisión de sentido de giro distancia de trayecto > 65 535	

STOP de CPU/maestro y estado RESET

Comportamiento en caso de STOP de la CPU/del maestro

La siguiente tabla muestra el comportamiento del 1PosU en STOP de la CPU/del maestro.

Reacción a STOP de la CPU/del maestro	Reacción del 1PosU
• por desconexión (POWER OFF) de la CPU/del maestro del bus o bien • Por desconexión (POWER OFF) IM 151 / IM 151 FO o bien • Por fallo en la transmisión de bus o bien • Por cambio de estado operativo de RUN a STOP	• La marcha en curso se detiene. • Las 3 salidas digitales pasan a 0. • Bit de respuesta (Página 11159)POS_ERR = 0 • Bit de respuesta (Página 11159)POS_DONE = 1

Abandono del estado STOP de la CPU/del maestro

La siguiente tabla muestra el comportamiento del 1PosU después de abandonar el estado STOP de la CPU/del maestro.

Abandono del estado STOP de la CPU/del maestro	Reacción del 1PosU
• Por conexión (POWER ON) de la CPU/del maestro del bus o bien • Por conexión (POWER ON) IM 151 / IM 151 FO o bien • Después de un fallo en la transmisión de bus o bien • Tras cambio de estado operativo de STOP a RUN	• La interfaz de respuesta (Página 11159) del 1PosU permanece actual. • El eje permanece sincronizado y el valor real está actualizado. • La zona de alcance del encóder que ha sido desplazada permanece vigente. • La distancia de desconexión y de cambio de velocidad modificada así como la distancia de trayecto para la vigilancia de sentido de giro permanecen vigentes. • Un JOB 9 iniciado "Evaluar señal de búsqueda de referencia (Página 11120)" y JOB 10 "Función de congelación (Página 11122)" permanecen activos. • El JOB 15Valor de respuesta (Página 11125) seleccionado está actualizado.

Estado RESET del 1PosU

La siguiente tabla muestra el comportamiento del 1PosU en el estado RESET.

Estado RESET del 1PosU y cambio de parámetros del 1PosU	Reacción del 1PosU
• Por la modificación de los parámetros del 1PosU y por la carga de la parametrización o configuración de la estación ET 200S en la CPU/el maestro del bus o bien • Por conexión (POWER ON) en el módulo de potencia del 1PosU o bien • Inserción del 1PosU bajo tensión	• El eje está sincronizado y el valor real equivale al valor de encóder actual. • La zona de alcance del encóder no ha sido desplazada. • Las distancias de desconexión y cambio de velocidad se adoptan de los parámetros. • A la distancia de trayecto para la vigilancia de sentido de giro se le asigna el valor correspondiente al doble de la distancia de desconexión. • Un JOB 9 iniciado "Evaluar señal de búsqueda de referencia (Página 11120)" y JOB 10 "Función de congelación (Página 11122)" permanecen activos. • La distancia residual se indica como valor de respuesta (Página 11125).

Configurar y parametrizar 1PosU

En servicio

Seleccionar interfaz del encóder

Interfaz del encóder

Al preseleccionar la interfaz del encóder se elige el tipo de señal con el que se desea controlar el 1PosU.

Interfaz del encóder	Descripción
Incremental 5 V (Página 11140)	El 1PosU recibe las señales de los encóders incrementales con señales diferenciales de 5 V.
Incremental 24 V (Página 11146)	El 1PosU recibe las señales de los encóders incrementales con señales de 24 V.
SSI absoluto (Página 11152)	El 1PosU recibe las señales de los encóders absolutos (SSI).

Parámetros en encóders incrementales con señales diferenciales de 5 V.

Habilitar

Activar el diagnóstico de grupo

Diagnóstico de grupo

El diagnóstico de grupo se activa en la casilla de verificación correspondiente.

Un error de encóder ERR_ENCODER, la falta de tensión de carga ERR_2L+ o un error de parametrización provocan un diagnóstico específico de canal con diagnóstico de grupo habilitado.

Activación del diagnóstico de las señales de encóder

Diagnóstico de las señales del encóder

El diagnóstico de las señales del encóder se activa en la casilla de verificación correspondiente.

Si se ha habilitado el "Diagnóstico de las señales del encóder" las señales en del encóder A, /A y B, /B se someten a una vigilancia de cortocircuito y rotura de hilo. Si no está activado el modo de dosificación se vigila además la secuencia de señales. Un cambio de flanco simultáneo de las señales A y B provoca un error.

Activar diagnóstico de señales de marca cero

Diagnóstico de señales de marca cero

El diagnóstico de las señales de marca cero se activa en la casilla de verificación correspondiente.

Si se ha habilitado el "Diagnóstico de señales de marca cero", las señales de marca cero N, /N se vigilan para ver si presentan cortocircuito o rotura de hilo.

Encóders y ejes

Activar modo de dosificación

Modo de dosificación

El modo de dosificación se activa en la casilla de verificación correspondiente.

En el modo de dosificación (Página 11094) el 1PosU sólo evalúa las señales del encóder A y /A. De esta forma sólo es posible realizar una modificación del desplazamiento en sentido positivo. El valor real se incrementa con cada flanco ascendente.

Activar inversión del sentido de giro

Inversión del sentido de giro

La inversión del sentido de giro activa en la casilla de verificación correspondiente.

Con el parámetro "Inversión del sentido de giro" es posible adaptar el sentido de giro del encóder al del accionamiento y al del eje.

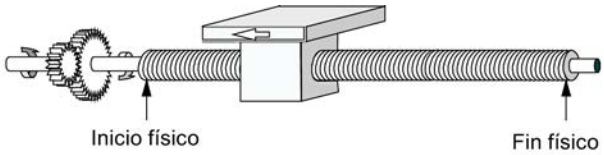
En el modo de dosificación (Página 11094) no es posible realizar una inversión del sentido de giro.

Seleccionar tipo de eje

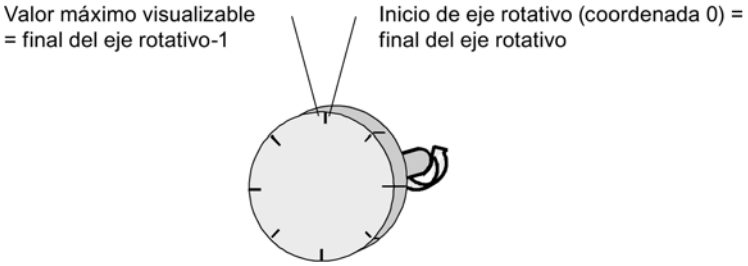
Tipos de eje

El módulo de posicionamiento 1PosU puede controlar los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento física limitada.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Definir final del eje rotativo

Final del eje rotativo

Este parámetro sólo es relevante para el tipo de eje "Eje rotativo".

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar. Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo (0).

La indicación salta

- si se rebasa (el fin del eje rotativo – 1) a 0
- si se rebasa por 0 al (final del eje rotativo – 1)

Rango de valores permitido para el final del eje rotativo: 1 … 16 777 215
La entrada "0" provoca un error de parametrización.

Entradas digitales

Seleccionar final de carrera negativo y final de carrera positivo

Efecto del final de carrera por hardware

Las dos entradas digitales DI0 y DI1 son evaluadas por el 1PosU como final de carrera por hardware:

- DI0 es el de final de carrera - y limita la zona de trabajo en sentido -
- DI1 es el de final de carrera + y limita la zona de trabajo en sentido +

Puede parametrizar cada uno de los finales de carrera por hardware individualmente **como contacto normalmente cerrado o contacto normalmente abierto**.

Los finales de carrera por hardware se evalúan para ejes lineales y para ejes rotativos.

Sólo se evalúa el final de carrera por hardware correspondiente al sentido en el que se controla el accionamiento.

Así, al alcanzar o sobrepasar un final de carrera por hardware, es posible abandonarlo sin ninguna indicación de fallo simplemente iniciando la marcha en el sentido contrario.

El nivel actual de las entradas digitales, retardado por la velocidad de actualización, está indicado en la interfaz de realimentación.

La tabla siguiente muestra la forma en la que los finales de carrera por hardware tienen efecto sobre los distintos MODEs.

MODE	Efecto del final de carrera por hardware
Búsqueda del punto de referencia	En el final de carrera por hardware el 1PosU ejecuta una inversión de sentido automática.
Jog	En el final de carrera por hardware se detiene el movimiento del eje, las 3 salidas digitales pasan a 0 y se notifica el bit de respuesta POS_ERR.
Posicionamiento absoluto	
Posicionamiento relativo	

Arrancar en el final de carrera por hardware

La tabla siguiente muestra la reacción del 1PosU al iniciar un MODE en el final de carrera por hardware.

Sentido	Reacción del 1PosU
Arrancar dentro de la zona de trabajo	El 1PosU inicia el MODE predeterminado.
Arrancar fuera de la zona de trabajo	El bit de respuesta POS_ERR = 1 se activa.

Seleccionar leva de reducción

Efecto de la leva de reducción

La entrada digital DI2 es evaluada por el 1PosU como leva de reducción o entrada de congelación:

- Evaluar como leva de reducción en MODE 3 "Búsqueda del punto de referencia (Página 11103)" y JOB 9 "Señal de búsqueda de referencia (Página 11120)"
- como entrada de congelación en JOB 10 "función de congelación (Página 11122)"

La leva de reducción puede parametrizarse **como contacto normalmente cerrado o contacto normalmente abierto**.

Evaluar búsqueda del punto de referencia y señal de búsqueda de referencia

Seleccionar señal de búsqueda de referencia

Señal de búsqueda de referencia

El parámetro "Señal de búsqueda de referencia" determina para MODE 3 "Búsqueda del punto de referencia (Página 11103)" y JOB 9 "Evaluar señal de búsqueda de referencia (Página 11120)" el encóder relevante o la combinación de encóder y la marca cero. Opciones posibles:

- Sensor del punto de referencia y señal cero
- Sensor del punto de referencia
- Señal cero

El tipo de encóder de referencia se selecciona con el parámetro "Sensor del punto de referencia (Página 11144)".

Seleccionar el sensor del punto de referencia

Sensor del punto de referencia

El parámetro "Sensor punto de referencia" determina el interruptor relevante y el sentido de búsqueda de referencia en el que debe sobrepasarse el encóder. Posibilidades de selección:

- Leva reductora a -
- Leva reductora +
- Final de carrera -
- Final de carrera +

Este parámetro sólo es relevante si en "Señal de búsqueda de referencia (Página 11139)" ha seleccionado:

- Sensor del punto de referencia y señal cero
  o bien
- Sensor del punto de referencia

Seleccionar sentido de arranque

Sentido de arranque:

El parámetro "Sentido de arranque" determina el sentido de inicio para la MODE 3 "búsqueda del punto de referencia (Página 11103)". Posibilidades de selección:

- Más (+)
- Menos (-)

Accionamiento

Seleccionar el modo de control

Modo de control

El parámetro "Modo de control" define la función de las salidas digitales para la activación del accionamiento (Página 11098). Posibilidades de selección:

- Modo de control 0:
  - DO0 Desplazamiento negativo
  - DO1 Desplazamiento positivo
  - DO2 Marcha rápida/lenta
- Modo de control 1:
  - DO0 Marcha rápida
  - DO1 Marcha lenta (en tal caso la marcha rápida es 0)
  - DO2 Desplazamiento positivo (1) / Desplazamiento negativo (0)

Introducir distancia de desconexión

Distancia de desconexión

La distancia de desconexión (Página 11092) define la distancia con respecto al destino a la que el accionamiento decelera la marcha lenta a 0.

Si la distancia de desconexión ≥ distancia de cambio de velocidad, no existe punto de cambio de velocidad. No se reduce la marcha rápida a marcha lenta, sino que se ejecuta directamente la reacción en el punto de desconexión.

Rango de valores admisible:

- 0 a 65 535

El JOB 3 (Página 11118) permite modificar la distancia de desconexión.

Introducir la distancia de cambio de velocidad

Distancia de cambio de velocidad

La distancia de cambio de velocidad (Página 11092) define la distancia con respecto al destino a la que el accionamiento reduce de marcha rápida a marcha lenta.

Rango de valores admisible:

- 0 a 65 535

El JOB 4 (Página 11119) permite modificar la distancia de cambio de velocidad.

Introducir el tiempo de deceleración para el cambio de sentido

T_{min} Cambio de sentido

El parámetro "T_{min} Cambio de sentido" determina el tiempo de deceleración para el cambio de sentido. Las salidas digitales se desconectan y a continuación se ejecuta el cambio de sentido con una deceleración de T_{min}. T_{min} se aplica en cada cambio de sentido durante un desplazamiento.

T_{min} no es efectivo al iniciar tras POS_DONE = 1 o POS_ERR = 1.

Rango de valores admisible:

- 0 a 255

El valor que introduzca se multiplica por 10. Introduzca pues T_{min} en incrementos de 10 ms (p. ej: 0 ms, 10 ms o 2550 ms).

Parámetros en encóders incrementales con señales de 24 V

Habilitaciones

Activar el diagnóstico de grupo

Diagnóstico de grupo

El diagnóstico de grupo se activa en la casilla de verificación correspondiente.

Un error de encóder ERR_ENCODER, la falta de tensión de carga ERR_2L+ o un error de parametrización provocan un diagnóstico específico de canal con diagnóstico de grupo habilitado.

Activación del diagnóstico de las señales del encóder

Diagnóstico de las señales de encóder

El diagnóstico de las señales del encóder se activa en la casilla de verificación correspondiente.

Si se habilita el "Diagnóstico de las señales del encóder", un cambio de flanco simultáneo de las señales A* y B* provoca un error.

Encóder y ejes

Activar el modo de dosificación

Modo de dosificación

El modo de dosificación se activa en la casilla de verificación correspondiente.

En el modo de dosificación (Página 11094) el 1PosU evalúa sólo la señal de encóder A*. De esta forma sólo es posible realizar una modificación del desplazamiento en sentido positivo. El valor real se incrementa con cada flanco ascendente.

Seleccionar entradas de encóder

Entradas de encóder

En las entradas del encóder es posible conectar diferentes encóders:

- Salida tipo PNP/simétrica
- Salida tipo NPN

Nota

Si se ha seleccionado el ajuste "Salida tipo NPN" en el parámetro "Entradas del encóder", es preciso utilizar también sensores tipo NPN.

En la lista desplegable seleccione la salida del encóder utilizado para las señales A*, B* y N*.

Activar inversión del sentido de giro

Inversión del sentido de giro

La inversión del sentido de giro activa en la casilla de verificación correspondiente.

Con el parámetro "Inversión del sentido de giro" es posible adaptar el sentido de giro del encóder al del accionamiento y al del eje.

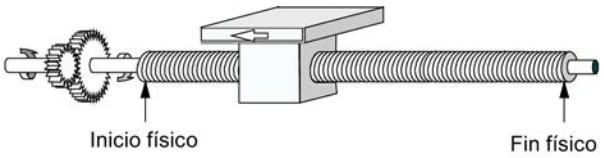
En el modo de dosificación (Página 11094) no es posible realizar una inversión del sentido de giro.

Seleccionar tipo de eje

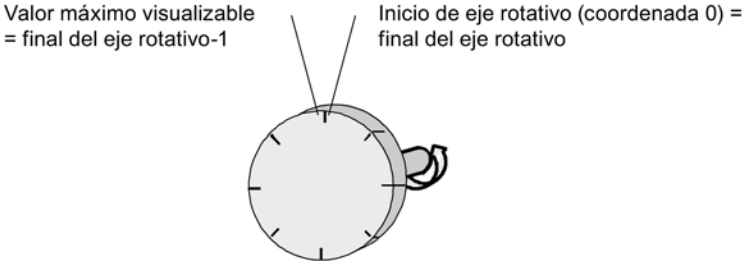
Tipos de eje

El módulo de posicionamiento 1PosU puede controlar los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento física limitada.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Definir final del eje rotativo

Final del eje rotativo

Este parámetro sólo es relevante para el tipo de eje "Eje rotativo".

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar. Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo (0).

La indicación salta

- si se rebasa (el fin del eje rotativo – 1) a 0
- si se rebasa por 0 al (final del eje rotativo – 1)

Rango de valores permitido para el final del eje rotativo: 1 … 16 777 215
La entrada "0" provoca un error de parametrización.

Entradas digitales

Seleccionar final de carrera negativo y final de carrera positivo

Efecto del final de carrera por hardware

Las dos entradas digitales DI0 y DI1 son evaluadas por el 1PosU como final de carrera por hardware:

- DI0 es el de final de carrera - y limita la zona de trabajo en sentido -
- DI1 es el de final de carrera + y limita la zona de trabajo en sentido +

Puede parametrizar cada uno de los finales de carrera por hardware individualmente **como contacto normalmente cerrado o contacto normalmente abierto**.

Los finales de carrera por hardware se evalúan para ejes lineales y para ejes rotativos.

Sólo se evalúa el final de carrera por hardware correspondiente al sentido en el que se controla el accionamiento.

Así, al alcanzar o sobrepasar un final de carrera por hardware, es posible abandonarlo sin ninguna indicación de fallo simplemente iniciando la marcha en el sentido contrario.

El nivel actual de las entradas digitales, retardado por la velocidad de actualización, está indicado en la interfaz de realimentación.

La tabla siguiente muestra la forma en la que los finales de carrera por hardware tienen efecto sobre los distintos MODEs.

MODE	Efecto del final de carrera por hardware
Búsqueda del punto de referencia	En el final de carrera por hardware el 1PosU ejecuta una inversión de sentido automática.
Jog	En el final de carrera por hardware se detiene el movimiento del eje, las 3 salidas digitales pasan a 0 y se notifica el bit de respuesta POS_ERR.
Posicionamiento absoluto	
Posicionamiento relativo	

Arrancar en el final de carrera por hardware

La tabla siguiente muestra la reacción del 1PosU al iniciar un MODE en el final de carrera por hardware.

Sentido	Reacción del 1PosU
Arrancar dentro de la zona de trabajo	El 1PosU inicia el MODE predeterminado.
Arrancar fuera de la zona de trabajo	El bit de respuesta POS_ERR = 1 se activa.

Seleccionar leva de reducción

Efecto de la leva de reducción

La entrada digital DI2 es evaluada por el 1PosU como leva de reducción o entrada de congelación:

- Evaluar como leva de reducción en MODE 3 "Búsqueda del punto de referencia (Página 11103)" y JOB 9 "Señal de búsqueda de referencia (Página 11120)"
- como entrada de congelación en JOB 10 "función de congelación (Página 11122)"

La leva de reducción puede parametrizarse **como contacto normalmente cerrado o contacto normalmente abierto**.

Evaluar búsqueda del punto de referencia y señal de búsqueda de referencia

Seleccionar señal de búsqueda de referencia

Señal de búsqueda de referencia

El parámetro "Señal de búsqueda de referencia" determina para MODE 3 "Búsqueda del punto de referencia (Página 11103)" y JOB 9 "Evaluar señal de búsqueda de referencia (Página 11120)" el encóder relevante o la combinación de encóder y la marca cero. Opciones posibles:

- Sensor del punto de referencia y señal cero
- Sensor del punto de referencia
- Señal cero

El tipo de encóder de referencia se selecciona con el parámetro "Sensor del punto de referencia (Página 11150)".

Seleccionar el sensor del punto de referencia

Sensor del punto de referencia

El parámetro "Sensor punto de referencia" determina el interruptor relevante y el sentido de búsqueda de referencia en el que debe sobrepasarse el encóder. Posibilidades de selección:

- Leva reductora a -
- Leva reductora +
- Final de carrera -
- Final de carrera +

Este parámetro sólo es relevante si en "Señal de búsqueda de referencia (Página 11146)" ha seleccionado:

- Sensor del punto de referencia y señal cero
  o bien
- Sensor del punto de referencia

Seleccionar sentido de arranque

Sentido de arranque:

El parámetro "Sentido de arranque" determina el sentido de inicio para la MODE 3 "búsqueda del punto de referencia (Página 11103)". Posibilidades de selección:

- Más (+)
- Menos (-)

Accionamiento

Seleccionar el modo de control

Modo de control

El parámetro "Modo de control" define la función de las salidas digitales para la activación del accionamiento (Página 11098). Posibilidades de selección:

- Modo de control 0:
  - DO0 Desplazamiento negativo
  - DO1 Desplazamiento positivo
  - DO2 Marcha rápida/lenta
- Modo de control 1:
  - DO0 Marcha rápida
  - DO1 Marcha lenta (en tal caso la marcha rápida es 0)
  - DO2 Desplazamiento positivo (1) / Desplazamiento negativo (0)

Introducir distancia de desconexión

Distancia de desconexión

La distancia de desconexión (Página 11092) define la distancia con respecto al destino a la que el accionamiento decelera la marcha lenta a 0.

Si la distancia de desconexión ≥ distancia de cambio de velocidad, no existe punto de cambio de velocidad. No se reduce la marcha rápida a marcha lenta, sino que se ejecuta directamente la reacción en el punto de desconexión.

Rango de valores admisible:

- 0 a 65 535

El JOB 3 (Página 11118) permite modificar la distancia de desconexión.

Introducir la distancia de cambio de velocidad

Distancia de cambio de velocidad

La distancia de cambio de velocidad (Página 11092) define la distancia con respecto al destino a la que el accionamiento reduce de marcha rápida a marcha lenta.

Rango de valores admisible:

- 0 a 65 535

El JOB 4 (Página 11119) permite modificar la distancia de cambio de velocidad.

Introducir el tiempo de deceleración para el cambio de sentido

T_{min} Cambio de sentido

El parámetro "T_{min} Cambio de sentido" determina el tiempo de deceleración para el cambio de sentido. Las salidas digitales se desconectan y a continuación se ejecuta el cambio de sentido con una deceleración de T_{min}. T_{min} se aplica en cada cambio de sentido durante un desplazamiento.

T_{min} no es efectivo al iniciar tras POS_DONE = 1 o POS_ERR = 1.

Rango de valores admisible:

- 0 a 255

El valor que introduzca se multiplica por 10. Introduzca pues T_{min} en incrementos de 10 ms (p. ej: 0 ms, 10 ms o 2550 ms).

Parámetros en encóders SSI

Habilitaciones

Activar el diagnóstico de grupo

Diagnóstico de grupo

El diagnóstico de grupo se activa en la casilla de verificación correspondiente.

Un error de encóder ERR_ENCODER, la falta de tensión de carga ERR_2L+ o un error de parametrización provocan un diagnóstico específico de canal con diagnóstico de grupo habilitado.

Activación del diagnóstico de las señales del encóder

Diagnóstico de las señales de encóder

El diagnóstico de las señales del encóder se activa en la casilla de verificación correspondiente.

Si se ha habilitado el "Diagnóstico de señales" de marca cero, las señales de marca cero D, /D se vigilan para ver si presentan cortocircuito o rotura de hilo. Se controla además el telegrama (bit de inicio y bit de parada)

Encóder y ejes

Seleccionar tipo de encóder

Tipo de encóder

Seleccione el encóder absoluto utilizado en la lista desplegable:

- Encóder monovuelta (SSI-13 bits)
- Encóder multivuelta (SSI-25 bits)

Seleccionar velocidad de transferencia

Velocidad de transferencia

Con el parámetro "Velocidad de transferencia" se define la velocidad de la transferencia de datos del encóder SSI al módulo de posicionamiento.

La velocidad de transferencia máxima depende de la velocidad de transferencia:

- 320 m → 125 kHz
- 160 m → 250 kHz
- 60 m → 500 kHz
- 20 m → 1 MHz

Seleccionar incrementos por vuelta del encóder

Incrementos por vuelta del encóder

El parámetro "Incrementos por vuelta del encóder" indica el número de incrementos que emite un encóder absoluto por cada vuelta.

Para los límites debe diferenciarse entre los distintos modelos de encóder. Sólo pueden seleccionarse valores en potencias de 2.

- Encóder monovuelta (número de vueltas = 1) con longitud del telegrama de 13 bits:
  - valor mínimo = 4
  - valor máximo = 8192
- Encóder multivuelta (número de vueltas > 1) con longitud del telegrama de 25 bits:
  - valor mínimo = 4
  - valor máximo = 8192

Seleccionar número de vueltas del encóder

Número de vueltas del encóder

Con este parámetro se indica el número de vueltas posibles con este encóder.

La indicación sólo es relevante para el encóder multivuelta. En los encóders monovuelta, el 1PosU establece el número de vueltas en 1.

Rango de valores admisible:

- 2 a 4096 en potencias de 2

Activar inversión del sentido de giro

Inversión del sentido de giro

La inversión del sentido de giro activa en la casilla de verificación correspondiente.

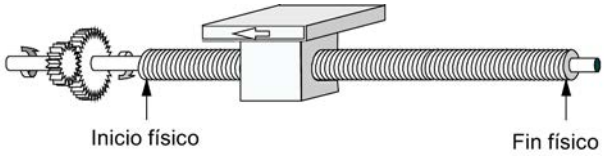
Con el parámetro "Inversión del sentido de giro" es posible adaptar el sentido de giro del encóder al del accionamiento y al del eje.

Seleccionar tipo de eje

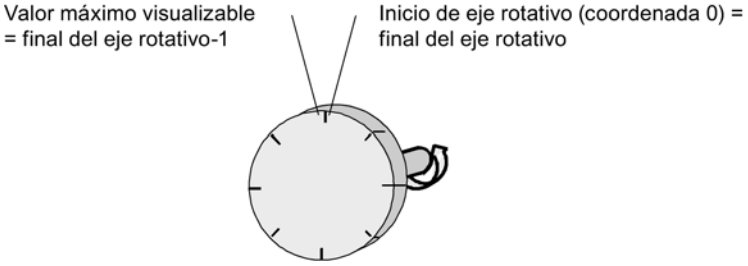
Tipos de eje

El módulo de posicionamiento 1PosU puede controlar los siguientes tipos de ejes:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento física limitada.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Entradas digitales

Seleccionar final de carrera negativo y final de carrera positivo

Efecto del final de carrera por hardware

Las dos entradas digitales DI0 y DI1 son evaluadas por el 1PosU como final de carrera por hardware:

- DI0 es el de final de carrera - y limita la zona de trabajo en sentido -
- DI1 es el de final de carrera + y limita la zona de trabajo en sentido +

Puede parametrizar cada uno de los finales de carrera por hardware individualmente **como contacto normalmente cerrado o contacto normalmente abierto**.

Los finales de carrera por hardware se evalúan para ejes lineales y para ejes rotativos.

Sólo se evalúa el final de carrera por hardware correspondiente al sentido en el que se controla el accionamiento.

Así, al alcanzar o sobrepasar un final de carrera por hardware, es posible abandonarlo sin ninguna indicación de fallo simplemente iniciando la marcha en el sentido contrario.

El nivel actual de las entradas digitales, retardado por la velocidad de actualización, está indicado en la interfaz de realimentación.

La tabla siguiente muestra la forma en la que los finales de carrera por hardware tienen efecto sobre los distintos MODEs.

MODE	Efecto del final de carrera por hardware
Búsqueda del punto de referencia	En el final de carrera por hardware el 1PosU ejecuta una inversión de sentido automática.
Jog	En el final de carrera por hardware se detiene el movimiento del eje, las 3 salidas digitales pasan a 0 y se notifica el bit de respuesta POS_ERR.
Posicionamiento absoluto	
Posicionamiento relativo	

Arrancar en el final de carrera por hardware

La tabla siguiente muestra la reacción del 1PosU al iniciar un MODE en el final de carrera por hardware.

Sentido	Reacción del 1PosU
Arrancar dentro de la zona de trabajo	El 1PosU inicia el MODE predeterminado.
Arrancar fuera de la zona de trabajo	El bit de respuesta POS_ERR = 1 se activa.

Seleccionar leva de reducción

Efecto de la leva de reducción

La entrada digital DI2 es evaluada por el 1PosU como leva de reducción o entrada de congelación:

- Evaluar como leva de reducción en MODE 3 "Búsqueda del punto de referencia (Página 11103)" y JOB 9 "Señal de búsqueda de referencia (Página 11120)"
- como entrada de congelación en JOB 10 "función de congelación (Página 11122)"

La leva de reducción puede parametrizarse **como contacto normalmente cerrado o contacto normalmente abierto**.

Accionamiento

Seleccionar el modo de control

Modo de control

El parámetro "Modo de control" define la función de las salidas digitales para la activación del accionamiento (Página 11098). Posibilidades de selección:

- Modo de control 0:
  - DO0 Desplazamiento negativo
  - DO1 Desplazamiento positivo
  - DO2 Marcha rápida/lenta
- Modo de control 1:
  - DO0 Marcha rápida
  - DO1 Marcha lenta (en tal caso la marcha rápida es 0)
  - DO2 Desplazamiento positivo (1) / Desplazamiento negativo (0)

Introducir distancia de desconexión

Distancia de desconexión

La distancia de desconexión (Página 11092) define la distancia con respecto al destino a la que el accionamiento decelera la marcha lenta a 0.

Si la distancia de desconexión ≥ distancia de cambio de velocidad, no existe punto de cambio de velocidad. No se reduce la marcha rápida a marcha lenta, sino que se ejecuta directamente la reacción en el punto de desconexión.

Rango de valores admisible:

- 0 a 65 535

El JOB 3 (Página 11118) permite modificar la distancia de desconexión.

Introducir la distancia de cambio de velocidad

Distancia de cambio de velocidad

La distancia de cambio de velocidad (Página 11092) define la distancia con respecto al destino a la que el accionamiento reduce de marcha rápida a marcha lenta.

Rango de valores admisible:

- 0 a 65 535

El JOB 4 (Página 11119) permite modificar la distancia de cambio de velocidad.

Introducir el tiempo de deceleración para el cambio de sentido

T_{min} Cambio de sentido

El parámetro "T_{min} Cambio de sentido" determina el tiempo de deceleración para el cambio de sentido. Las salidas digitales se desconectan y a continuación se ejecuta el cambio de sentido con una deceleración de T_{min}. T_{min} se aplica en cada cambio de sentido durante un desplazamiento.

T_{min} no es efectivo al iniciar tras POS_DONE = 1 o POS_ERR = 1.

Rango de valores admisible:

- 0 a 255

El valor que introduzca se multiplica por 10. Introduzca pues T_{min} en incrementos de 10 ms (p. ej: 0 ms, 10 ms o 2550 ms).

Interfaz de control y reacción del 1PosU

Interfaz de control del 1PosU

Asignación de la interfaz de control

La siguiente tabla muestra la asignación de la interfaz de control (salidas) del 1PosU.

Dirección	Asignación					
Byte 0	Bit 0.7 … 0.4 representan los MODEs					
	Bit	7	6	5	4	
		0	0	0	0	MODE 0 = Detener
		0	0	0	1	MODE 1 = Jog
		0	0	1	1	MODE 3 = Búsqueda del punto de referencia
		0	1	0	0	MODE 4 = Posicionamiento relativo
		0	1	0	1	MODE 5 = Posicionamiento absoluto
	Bit 3: SPEED (SPEED = 0 es marcha lenta; SPEED = 1 es marcha rápida) Bit 2: DIR_M Bit 1: DIR_P Bit 0: START					
Bytes 1 a 3	en MODE 3 = Búsqueda del punto de referencia: coordenada del punto de referencia					
	en MODE 4 = Posicionamiento relativo: parte del trayecto					
	en MODE 5 = Posicionamiento absoluto: punto de destino					
Byte 4	Bit 4.7 … 4.4 representan los JOBs					
	Bit	7	6	5	4	
		0	0	0	0	JOB 0 = Cancelar procesamiento JOB
		0	0	0	1	JOB 1 = Ajustar valor real
		0	0	1	0	JOB 2 = Desplazar zona del encóder (sólo con encóder SSI)
		0	0	1	1	JOB 3 = Modificar distancia de desconexión

Dirección	Asignación					
		0	1	0	0	JOB 4 = Modificar distancia de cambio de velocidad
		1	0	0	1	JOB 9 = Evaluar señal de búsqueda de referencia
		1	0	1	0	JOB 10 = Función de congelación
		1	0	1	1	JOB 11 = Ajustar vigilancia de sentido de giro
		1	1	1	1	JOB 15 = Mostrar valores actuales
	Bit 3: EXTF_ACK Bit 2: reservado = 0 Bit 1: reservado = 0 Bit 0: JOB_REQ					
Bytes 5 a 7	en función del JOB seleccionado: • en JOB 1 = Coordenada del valor real • en JOB 2 = Offset • en JOB 3 = Distancia de desconexión • en JOB 4 = Distancia de cambio de diferencia • en JOB 9 = Coordenada del punto de referencia • en JOB 10 – Byte 5: Bit 0 = Congelación con flanco ascendente en DI2 – Byte 5: Bit 1 = Congelación con flanco descendente en DI2 • en JOB 11 = Distancia de trayecto para vigilancia de sentido de giro • en JOB 15 – Byte 5: 0 = trayecto residual – Byte 5: 1 = velocidad real – Byte 5: 2 = Información de error					

Interfaz de realimentación del 1PosU

Asignación de la interfaz de realimentación

La siguiente tabla muestra la asignación de la interfaz de respuesta (entradas) del 1PosU.

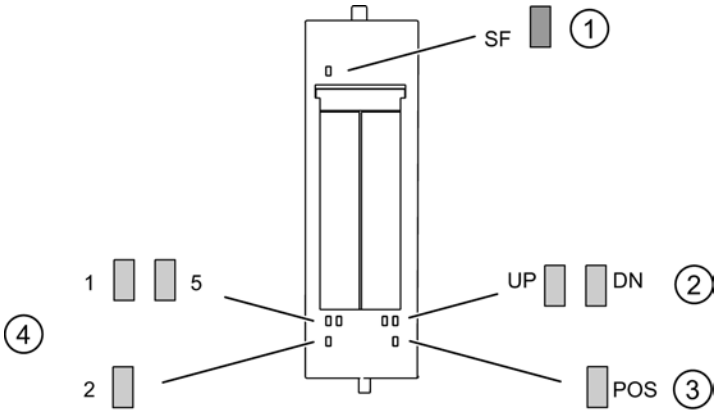
Dirección	Ocupación
Byte 0	Bit 7: ERR_ENCODER Bit 6: STATUS DO 2 Bit 5: STATUS DO 1 Bit 4: STATUS DO 0 Bit 3: SYNC Bit 2: POS_DONE Bit 1: POS_ERR Bit 0: POS_ACK
Bytes 1 a 3	Valor real

Dirección	Ocupación
Byte 4	Bit 7: ERR_2L+ Bit 6: STATUS DI 2 Leva de reducción Bit 5: STATUS DI 1 Final de carrera positivo Bit 4: STATUS DI 0 Final de carrera negativo Bit 3: reservado = 0 Bit 2: LATCH_DONE Bit 1: JOB_ERR Bit 0: JOB_ACK
Bytes 5 a 7	Valor de respuesta

Diagnóstico del 1PosU

Diagnóstico con indicadores LED

Indicación LED del 1PosU



① Error de grupo (rojo)
② Indicador de estado de cambio de valor real (verde)
③ Posicionamiento en curso (verde)
④ Indicadores de estado de las entradas digitales (verde)

Indicadores de estado y error mediante los LED del 1PosU

La tabla muestra los indicadores de estado y error del 1PosU.

Evento (LED)							Causa	Solución
SF	1	5	2	UP	DN	POS		
ON							No hay parametrización. Hay un aviso de diagnóstico.	Compruebe la parametrización. Evalúe el aviso de diagnóstico.
	ON						DI 0 activada	
		ON					DI 1 activada	
			ON				DI 2 activada	
				ON			cuando el valor real cambia de un valor menor a uno mayor	
					ON		cuando el valor real cambia de un valor mayor a uno menor	
						ON	Se está realizando el posicionamiento y una de las 3 salidas digitales está activada.	

Tipos de error

En el manual de producto del módulo de interfaz utilizado en la estación ET 200S encontrará información sobre la estructura del diagnóstico de canal.

Tipos de error del 1PosU

La tabla muestra los tipos de error del 1PosU.

Tipo de error		Significado	Solución
1_D	00001: Cortocircuito	Cortocircuito en la alimentación del encóder.	Compruebe el cableado del encóder. Corregir el cableado del proceso.
16_D	10000: Error de parametrización	Módulo no parametrizado.	Corregir la parametrización.

Tipo de error		Significado	Solución
17_D	10001: Falta la tensión de carga 2L+	No hay tensión de alimentación o bien es insuficiente.	Corregir el cableado del proceso. Compruebe la tensión de alimentación.
26_D	11010: Error externo	Rotura de hilo/cortocircuito de las señales del encóder. Rotura de hilo en el cable del encóder o cable no conectado. El encóder está defectuoso o hay interferencias. El tipo de encóder, la velocidad de transferencia o el tiempo monoflop no se corresponden con el encóder conectado; los encóders programables no coinciden con los ajustes del módulo.	Corregir el cableado del proceso Corregir la parametrización Sustituir el encóder.

11.4.2.8 Utilizar ET 200S 1STEP 5V

Asignación de terminales del 1STEP 5V

Reglas de cableado

Los cables (terminales 1 y 5 y terminales 4 y 8) conectados a la etapa de potencia deben estar apantallados y deben ser de par trenzado. La pantalla tiene que hacer contacto por ambos extremos. Para ello se debe usar el elemento de contactado de pantallas (ver el anexo de las instrucciones de servicio Sistema de periferia descentralizada ET 200S (http://support.automation.siemens.com/WW/view/es/1144348)).

Asignación de terminales

La tabla siguiente muestra la asignación de terminales para el 1STEP 5V.

Vista	Asignación de terminales	Observaciones
	TM-E15S24-01 y 1STEP 5V 1STEP 5V SF RDY POS 3 7 6ES7 138-4DC01-0AB0 P 1 5 $\overline{P}$ DC 24 V 2 6 DC 24 V DI0 3 7 DI1 D 4 8 $\overline{D}$ Impulsos Impulso_N Impulso DIR DIR_N Etapa de potencia Motor paso a paso Sentido	DC 24 V: tensión de carga DI0: habilitación de impulsos externa, STOP externo o final de carrera en sentido adelante/atrás DI1: sensor del punto de referencia o sensor del punto de referencia y final de carrera en sentido adelante/atrás P, /P y D, /D son señales según RS 422.
	TM-E15S26-A1 y 1STEP 5V 1STEP 5V SF RDY POS 3 7 6ES7 138-4DC01-0AB0 P 1 5 $\overline{P}$ DC 24 V 2 6 DC 24 V DI0 3 7 DI1 D 4 8 $\overline{D}$ AUX1 4 8 AUX1 AUX1 3 7 AUX1 Impulsos Impulso_N Impulso DIR DIR_N Etapa de potencia Motor paso a paso Sentido	DC 24 V: tensión de carga DI0: habilitación de impulsos externa, STOP externo o final de carrera en sentido adelante/atrás DI1: sensor del punto de referencia o sensor del punto de referencia y final de carrera en sentido adelante/atrás P, /P y D, /D son señales según RS 422.

Principios básicos del posicionamiento con motor paso a paso

Interacción de los componentes

A continuación se explica la interrelación entre los diferentes componentes para el posicionamiento con motor paso a paso.

1STEP 5V

El 1STEP 5V genera impulsos y una señal de sentido para etapas de potencia de motores paso a paso. En número de impulsos generados depende de la longitud del trayecto a recorrer. La frecuencia de impulsos es una medida para la velocidad. La variación de la frecuencia de impulsos por unidad de tiempo (segundo) es una medida para la aceleración o deceleración. El funcionamiento del 1STEP 5V se ajusta mediante sus parámetros y ajustes.

Etapa de potencia para motores paso a paso

La etapa de potencia es el elemento de unión entre el 1STEP 5V y el motor paso a paso. El 1STEP 5V envía señales diferenciales RS 422 para frecuencia y sentido. Una vez en la etapa de potencia, estas señales se convierten en corrientes de motor que controlan los movimientos del motor con máxima precisión.

Motores paso a paso

Los motores paso a paso se utilizan para el posicionamiento de ejes. Son la solución fácil y económica para trabajos de posicionamiento de alta precisión dentro de un amplio campo de aplicación.

El árbol de un motor paso a paso gira un ángulo determinado con cada impulso, de tal modo que, con secuencias de impulsos rápidas, el movimiento paso a paso se convierte en un movimiento de giro continuado.

Parámetros y ajustes

Informaciones necesarias

Para obtener una interrelación óptima entre los diferentes componentes debe proporcionar al 1STEP 5V diferentes datos:

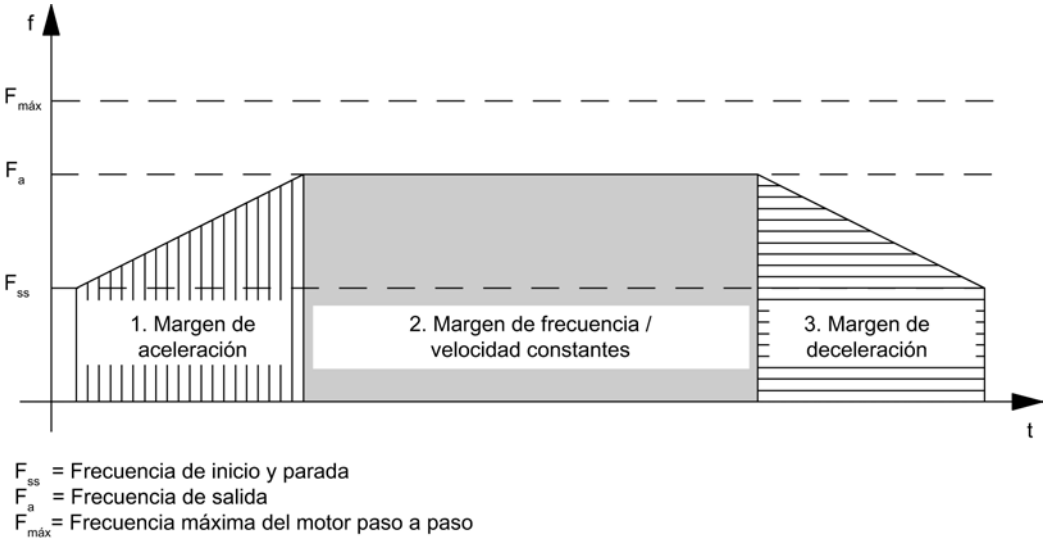
- **una única vez; en la configuración de los parámetros con su software de configuración**
  - Frecuencia base (Página 11168) F_b
  - Multiplicador n (Página 11165) para el ajuste de frecuencia de inicio y parada F_{ss}
  - Multiplicador i (Página 11165) para el ajuste de la aceleración/deceleración
  - Función y respuesta de las entradas digitales (Página 11185)
  - Zona de desplazamiento (Página 11181)
- **durante el funcionamiento; movimiento del motor por medio de una tarea de desplazamiento en su programa de usuario**
  - Multiplicador G (Página 11165) para la velocidad/frecuencia de salida F_a
  - Factor de reducción R (Página 11168)
  - Trayecto, posición o frecuencia
  - Modo de operación
  - Preajuste de sentido (tarea de desplazamiento) al inicio
- **durante el funcionamiento; para la adaptación a diferentes condiciones de carga como orden de parametrización en su programa de usuario**
  - Frecuencia base (Página 11168) F_b
  - Multiplicador n (Página 11165) para el ajuste de frecuencia de inicio y parada F_{ss}
  - Multiplicador i (Página 11165) para el ajuste de la aceleración/deceleración

Curva de desplazamiento del 1STEP 5V

Curva de desplazamiento

El 1STEP 5V ejecuta un modo incremental de acuerdo con la siguiente curva de desplazamiento. El 1STEP 5V genera los valores pico (frecuencia de inicio y parada,

frecuencia de salida y aceleración/deceleración) de la curva de desplazamiento con una frecuencia base (Página 11168) definida por el usuario.



Frecuencia de inicio y parada F_{ss}

La frecuencia de inicio y parada F_{ss} es la frecuencia a la que puede acelerarse el motor con carga en parado.

El valor de F_{ss} depende del momento de inercia de la carga. Se puede determinar mediante pruebas.

La frecuencia de inicio y parada F_{ss} es al mismo tiempo la frecuencia de salida mínima F_a, con la que puede mover el motor.

Ajuste de la frecuencia de inicio y parada F_{ss}

Mediante la parametrización, el 1STEP 5V permite el ajuste escalonado de la frecuencia de inicio y parada F_{ss}. Para ello, seleccione el multiplicador n entre 1 y 255 que va a multiplicarse con la frecuencia base F_b. Con el factor de reducción R (1 ó 0,1) en la tarea de desplazamiento es posible volver a reducir la frecuencia de inicio y parada F_{ss}.

La frecuencia de inicio y parada se calcula según la fórmula siguiente:

$F_{ss} = F_b \times n \times R$

Encontrará más información al respecto en la tabla (Página 11168) con los rangos para la frecuencia de inicio y parada, frecuencia de salida y aceleración.

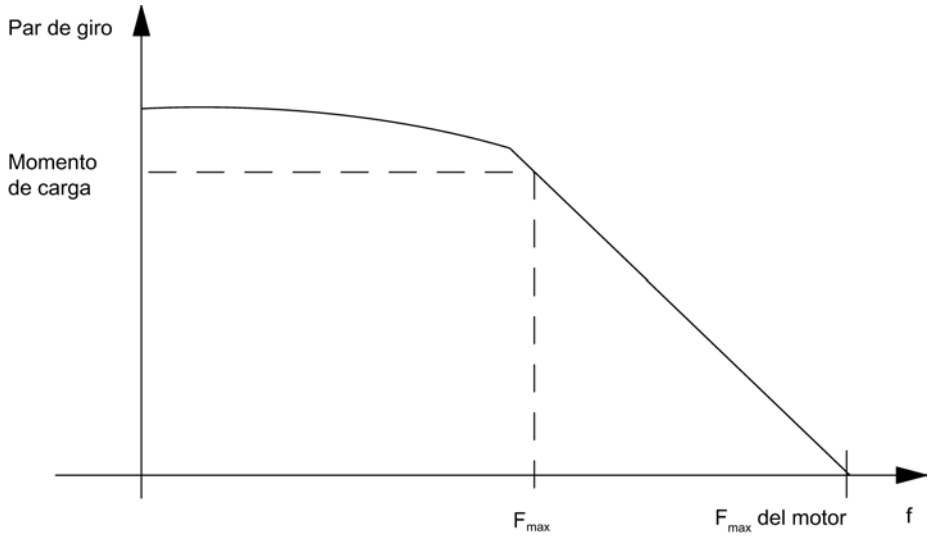
Frecuencia/velocidad máxima del eje $F_{máx}$

A la hora de elegir el motor paso a paso deben tenerse en cuenta los siguientes aspectos:

La frecuencia/velocidad máxima varía en función de la aplicación. Con esta frecuencia el motor debe alcanzar un par de giro suficientemente elevado para mover su carga.

Recuerde que aquí no se trata de la máxima frecuencia posible que puede soportar el motor o la etapa de potencia.

Con la curva característica correspondiente puede determinar la frecuencia máxima F_{max}. La figura siguiente muestra la curva característica de pares de giro de un motor paso a paso.



Frecuencia de salida/velocidad F_a

La frecuencia de salida puede regularse de forma diferente para cada marcha.

Al seleccionar la frecuencia de salida debe tener en cuenta también la duración mínima del impulso de la etapa de potencia utilizada (ver fórmula y tabla (Página 11168) con los rangos para la frecuencia de inicio y parada, la frecuencia de salida y la aceleración).

Si la frecuencia de salida seleccionada es menor que la frecuencia de inicio y parada F_{ss} ajustada, el 1STEP 5V iguala la frecuencia de salida con la frecuencia de inicio y parada F_{ss}.

F_a siempre debe ser inferior a F_{max}.

Ajuste de la frecuencia de salida/velocidad F_a

El 1STEP 5V permite el ajuste escalonado de la frecuencia de salida F_a. Para ello, seleccione el multiplicador G entre 1 y 255 que va a multiplicarse con la frecuencia base F_b. Con el factor de reducción R (1 ó 0,1) en la tarea de desplazamiento es posible volver a reducir la frecuencia de salida F_a.

La frecuencia de salida se calcula según la fórmula siguiente:

$F_a = F_b \times G \times R$

Encontrará más información al respecto en la tabla (Página 11168) con los rangos para la frecuencia de inicio y parada, frecuencia de salida y aceleración.

Aceleración/deceleración a

La aceleración/deceleración máxima permitida depende de la carga que se vaya a mover.

El motor debe alcanzar un par de giro lo suficientemente alto como para poder acelerar o decelerar la carga sin pérdida de pasos.

También deben considerarse otros criterios para determinar la aceleración/deceleración en función de la aplicación, como p.ej. un arranque y un frenado suaves.

Ajuste de la aceleración/deceleración a

El 1STEP 5V permite un ajuste escalonado de la aceleración/deceleración mediante parametrización aplicando el multiplicador i.

Durante la fase de aceleración, la frecuencia va aumentando continuamente desde la frecuencia de inicio y parada F_{ss} hasta alcanzar la frecuencia de salida F_a.

El intervalo de tiempo para el aumento continuo de la frecuencia se ajusta de forma escalonada. Para ello, seleccione un multiplicador i entre 1 y 255.

En la fase de deceleración, la frecuencia de salida se reduce de la misma forma.

Con el factor de reducción R (1 ó 0,1) en la tarea de desplazamiento es posible volver a reducir la aceleración/deceleración a.

La aceleración/deceleración se calcula según la fórmula siguiente:

$a = F_b \times R / (i \times 0{,}128\ ms)$

Encontrará más información al respecto en la tabla (Página 11168) con los rangos para la frecuencia de inicio y parada, frecuencia de salida y aceleración.

Ajuste de la frecuencia base

Introducción

El 1STEP 5V permite el ajuste escalonado de la frecuencia base mediante parametrización.

La frecuencia base determina el margen para la frecuencia de inicio y parada, la frecuencia de salida y la aceleración.

Procedimiento

1. Dependiendo de la prioridad de su solicitud, seleccione de la siguiente tabla un margen adecuado de la frecuencia de inicio y parada F_{ss} y de la frecuencia de salida F_a o bien de la aceleración a según los siguientes criterios:
   - Margen de la frecuencia de inicio y parada F_{ss},
     p. ej. para un arranque y un frenado tan rápidos como sea posible.
   - Margen de la frecuencia de salida F_a,
     p. ej. para un ajuste de velocidad lo más exacto posible.
   - Margen de aceleración a,
     p. ej. para procesos de posicionamiento tan rápidos como sea posible.
2. Determine la frecuencia base F_b con ayuda de la tabla.
   Para optimizar la frecuencia base F_b proceda del siguiente modo:

3. Compruebe si el resto de valores correspondientes cubren sus necesidades. Si es necesario, seleccione una frecuencia base F_b diferente, más adecuada a sus necesidades.
4. Determine los multiplicadores para el ajuste de la frecuencia de salida F_a, la aceleración/deceleración a y la frecuencia de inicio y parada F_{ss}.
5. Consulte en la tabla el factor de reducción R correspondiente.

La tabla siguiente muestra los rangos para la frecuencia de inicio y parada, la frecuencia de salida y la aceleración.

Frecuencia base F_b en Hz	Factor de reducción R	Margen de frecuencia de inicio y parada F_{ss} Frecuencia de salida F_a en Hz	Margen de aceleración a en Hz/ms
		Fórmula: $F_{ss} = F_b \times n \times R$ $F_a = F_b \times G \times R$	Fórmula: $a = F_b \times R / (i \times 0{,}128\ ms)$
4	0,1	0,4 … 102	0,01 … 3,13
8	0,1	0,8 … 204	0,02 … 6,25
20	0,1	2 … 510	0,06 … 15,6
4	1	4 … 1020	0,12 … 31,3
8	1	8 … 2040	0,25 … 62,5
20	1	20 … 5100	0,61 … 156
40	1	40 … 10200	1,23 … 313
80	1	80 … 20400	2,45 … 625
200	1	200 … 51000	6,13 … 1563
400	1	400 … 102000	12,25 … 3125
800	1	800 … 204000	24,51 … 6250
2000 [1]	1	2000 … 510000	61,27 … 15625

F_b = frecuencia base
F_{ss} = frecuencia de inicio y parada
F_a = frecuencia de salida
a = aceleración/deceleración
R = factor de reducción
n = multiplicador para el ajuste escalonado de la frecuencia de inicio y parada
G = multiplicador para el ajuste escalonado de la frecuencia de salida
i = multiplicador para el ajuste escalonado de la aceleración/deceleración

[1] a partir de 6ES7138-4DD01-0AB0

La duración mínima del impulso resulta de la frecuencia de salida F_a ajustada y se calcula según la fórmula $T_{Puls} = 1 / (2 \times F_a)$.

Funciones del 1STEP 5V

Vista general de las funciones

Vista general

La función del 1STEP 5V es el posicionamiento de un accionamiento en determinados puntos de destino predefinidos (modos incrementales) y el desplazamiento sin fin con frecuencias predefinibles (modo de velocidad).

Para realizar esta tarea dispone de las siguientes funciones:

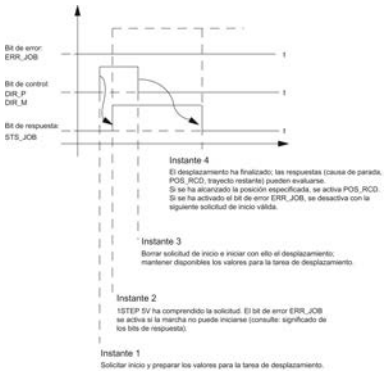
- Búsqueda del punto de referencia (Página 11171): el eje se sincroniza en función de un punto de referencia
- Definir punto de referencia (Página 11175) (a partir de 6ES7138-4DC01-0AB0): se asigna un valor a la posición actual.
- Modo incremental relativo (Página 11176) (posicionamiento relativo): el eje se desplaza un trayecto predeterminado
- Modo incremental absoluto (Página 11177) (posicionamiento absoluto) (a partir de 6ES7138-4DC01-0AB0): el eje se desplaza a una posición predeterminada.
- Modo de velocidad (Página 11178) (a partir de 6ES7138-4DC01-0AB0): el accionamiento se mueve con una velocidad flexible predefinible (frecuencia de impulsos).
- Detener la petición de desplazamiento (Página 11179)
- Modificar parámetros durante el funcionamiento (Página 11184)

Encontrará información sobre cómo controlar las funciones en la descripción de la interfaz de control (Página 11192).

Inicio de una petición de desplazamiento

Iniciar la petición de desplazamiento

La figura siguiente muestra el desarrollo de la solicitud de inicio para una petición de desplazamiento hasta finalizar el desplazamiento.



Evaluación del bit de error ERR_JOB

En cuanto el bit de realimentación (Página 11195) STS_JOB se haya borrado en el instante 4, deberá evaluar el bit de error ERR_JOB. Recuerde que el bit de realimentación STS_JOB no se borra hasta que los bits de control (Página 11192) DIR_P, DIR_M y C_PAR se han borrado.

Búsqueda del punto de referencia

Descripción del funcionamiento

El punto de referencia marca la referencia de su sistema de accionamiento (leva de referencia) para realizar las siguientes peticiones de desplazamiento. Puede determinar el punto de referencia p. ej. si monta un iniciador en la leva de referencia y lo conecta a la entrada digital DI1.

El 1STEP 5V garantiza una reproducción exacta del punto de referencia, ya que siempre parte en el mismo sentido. Este sentido debe definirse iniciando la búsqueda del punto de referencia siempre en el mismo sentido.

Petición de desplazamiento para búsqueda del punto de referencia

La petición de desplazamiento contiene la siguiente información:

- Multiplicador G para la velocidad/frecuencia de salida F_a
- Factor de reducción R para la frecuencia base F_b parametrizada
- Posición del punto de referencia
- Modo de operación = 1 para la búsqueda del punto de referencia
- Parada en la leva de referencia (Página 11179)
- Preajuste de sentido como inicio (Página 11192)

Nota

El 1STEP 5V comprueba los valores límite de la posición definida (mínimo 0 y máximo 16777215). El valor final se puede parametrizar.

Nota

Si ha parametrizado la reacción de la entrada digital (Página 11185) DI1 (7) como "Sensor punto de referencia y final de carrera", el 1STEP 5V elegirá automáticamente el sentido de arranque hacia el final de carrera, independientemente del preajuste de sentido de la petición de desplazamiento.

Bit de estado SYNC

El bit de estado (Página 11195)SYNC indica la sincronización del eje, es decir, este bit está activado tras la búsqueda correcta del punto de referencia y se ha borrado durante la marcha.

El bit de estado SYNC está borrado

- tras la parametrización de la estación ET 200S
- tras borrar la habilitación de impulsos
- tras un STOP de la CPU/del maestro

En estos casos resulta útil realizar una búsqueda del punto de referencia.

Bits de estado POS y POS_RCD

Durante la búsqueda del punto de referencia, su ejecución se indica con el bit de realimentación (Página 11195)POS activado.

Una vez finalizada correctamente la búsqueda del punto de referencia, el bit de realimentación POS_RCD activado indica que se ha llegado a la posición.

Si la búsqueda del punto de referencia se cancela, el bit de realimentación POS_RCD permanece desactivado.

Distancia residual, posición, frecuencia

El valor indicado como respuesta (Página 11195) es irrelevante durante la búsqueda del punto de referencia.

Nota

Para que el 1STEP 5V pueda llegar con exactitud al punto de referencia una y otra vez, el período de la frecuencia de inicio y parada debe ser mayor que el tiempo de operación de un paso individual, que va del 1STEP 5V al motor paso a paso y regresa al 1STEP 5V pasando por la leva de referencia (retardo de las entradas digitales: 4 ms).

Al detenerse en la leva de referencia o en un final de carrera durante la fase de aceleración, el 1STEP 5V sigue emitiendo impulsos durante 50 ms como máximo con la frecuencia ya alcanzada antes de iniciar el proceso de frenado. Con ello se evitan cambios de frecuencia abruptos que pueden provocar pérdidas de pasos.

Desarrollo de la búsqueda del punto de referencia

Fases de la búsqueda del punto de referencia

Una búsqueda del punto de referencia se compone como máximo de tres fases.

En la **primera fase** (1) y en la **segunda fase** (2) se garantiza que se encuentre la leva de referencia.

Ambas fases se recorren con la frecuencia de salida F_a predeterminada.

En la **tercera fase** (3) se avanza hasta la leva de referencia de forma reproducible con frecuencia de inicio y parada F_{ss} en el sentido seleccionado hasta llegar al punto de referencia ⊕.

Nota

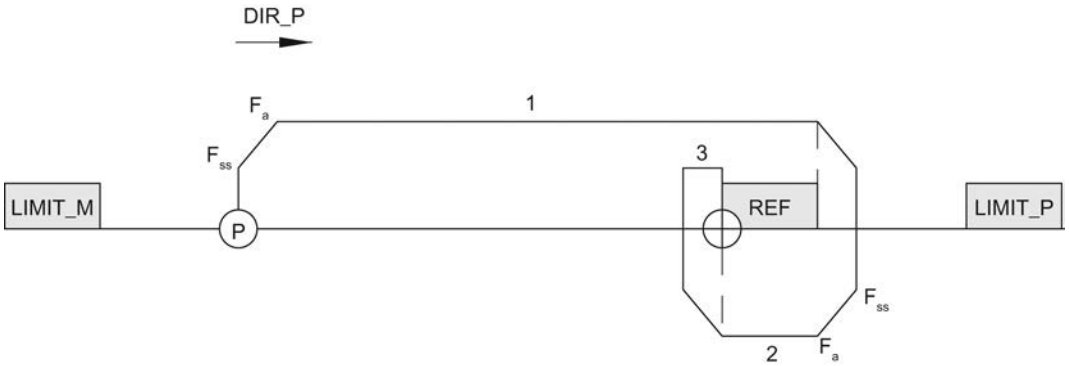
El número de impulsos generados en una fase equivale como máximo a la longitud parametrizada de la zona de desplazamiento menos 1.

Diferentes operaciones

Dependiendo de la ubicación (P) al iniciar la búsqueda del punto de referencia se obtienen diferentes modelos de la marcha (REF es la leva de referencia, cableada a la entrada digital DI1). La representación es aplicable al sentido de arranque Adelante (DIR_P).

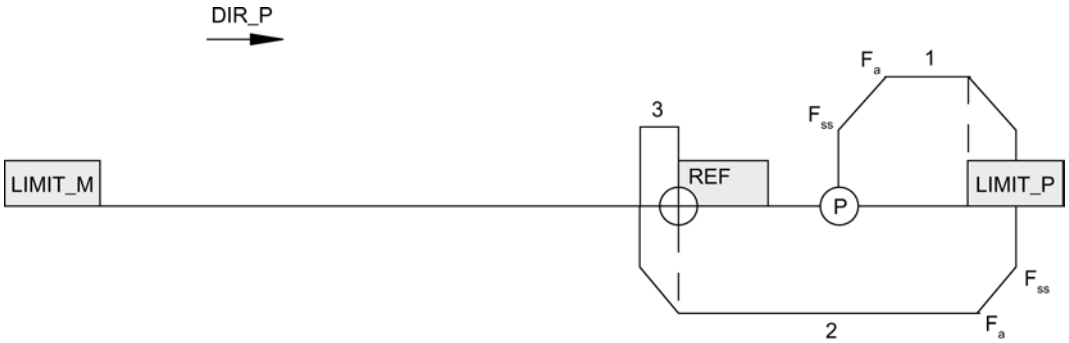
- **Inicio antes del REF o en el final de carrera LIMIT_M**

La figura siguiente muestra una búsqueda del punto de referencia con inicio antes de REF.



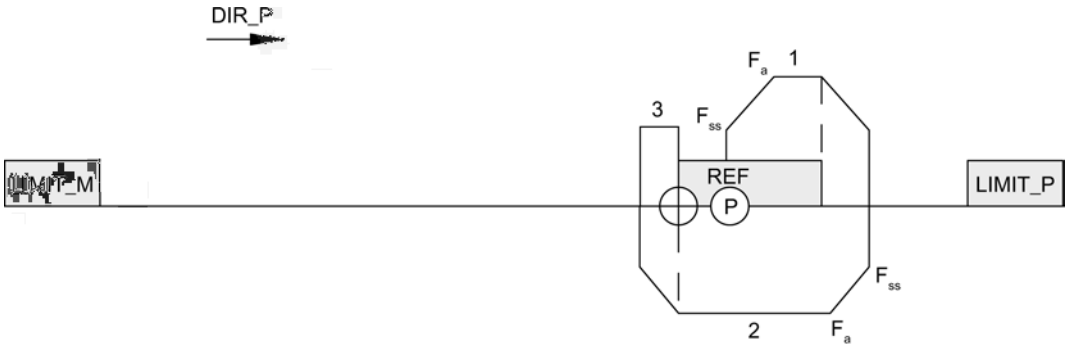
- **Inicio después de REF**

La figura siguiente muestra una búsqueda del punto de referencia con inicio después de REF.



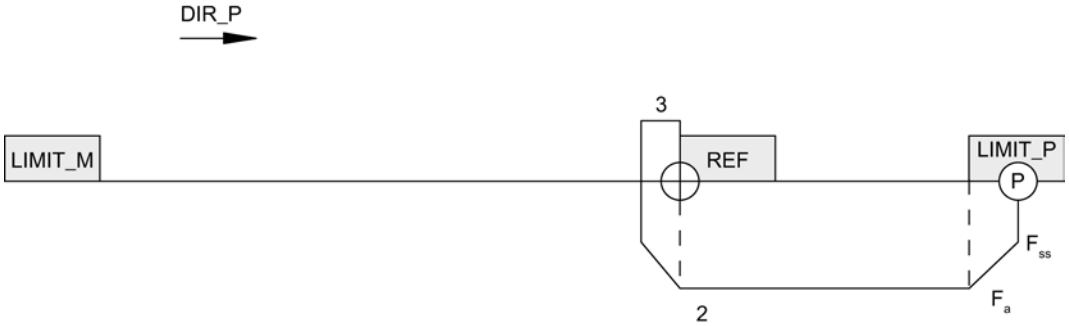
- **Inicio en el REF**

La figura siguiente muestra una búsqueda del punto de referencia con inicio sobre REF.



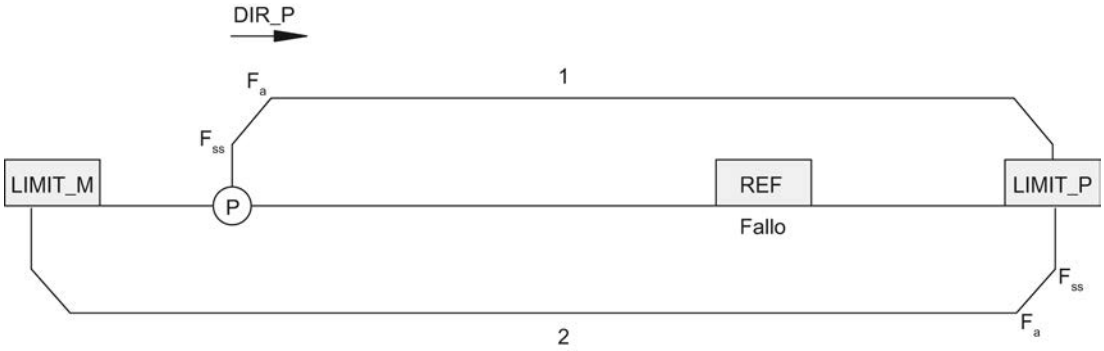
- Inicio sobre el final de carrera en sentido de arranque

La figura siguiente muestra una búsqueda del punto de referencia con inicio sobre el final de carrera en sentido de arranque.

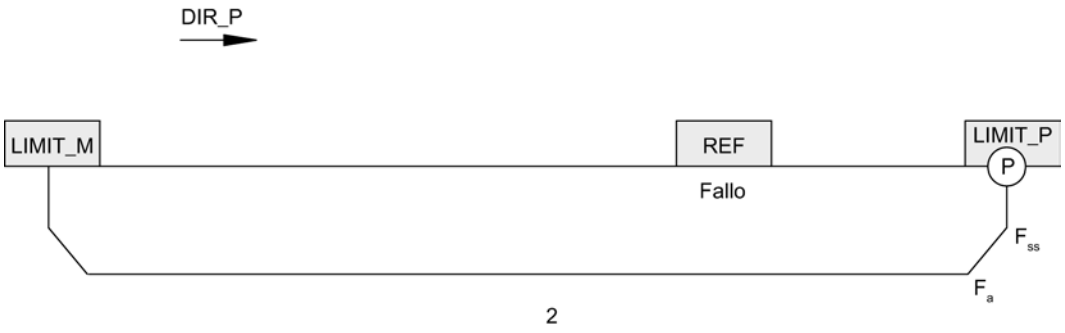


Reacción ante leva de referencia defectuosa con final de carrera (interrupción de la marcha)

La figura siguiente muestra una búsqueda del punto de referencia con la leva de referencia defectuosa e inicio antes de REF.



La figura siguiente muestra una búsqueda del punto de referencia con leva de referencia defectuosa e inicio sobre LIMIT_P.



Reacción en caso de leva de referencia siempre activada sin final de carrera

Al final de la primera fase, una vez emitidos 16777215 impulsos, la marcha finaliza con los bits de estado SYNC y POS_RCD borrados.

Reacción en caso de fallo de la leva de referencia sin final de carrera

Las tres fases de la marcha se ejecutan con la emisión de 16777215 impulsos en cada una. A continuación, la marcha se interrumpe con los bits de estado SYNC y POS_RCD borrados.

Definir punto de referencia (a partir de 6ES7138-4DC01-0AB0)

Descripción del funcionamiento

El punto de referencia marca la referencia de su sistema de accionamiento. Con base en esta referencia se orientarán tanto modos incrementales absolutos subsiguientes como el valor de posición en la interfaz de realimentación. Para definir el punto de referencia debe asignarse el valor de posición absoluto a la posición actual del motor paso a paso.

Petición de definición del punto de referencia

Una petición de definición del punto de referencia es una petición virtual sin desplazamiento. Contiene la siguiente información:

- Posición del punto de referencia
- Modo de operación = 4 para la definición del punto de referencia
- Libre preajuste de sentido como inicio (Página 11192)

Nota

El 1STEP 5V comprueba los valores límite de la posición definida (mínimo 0 y máximo 16777215). El valor final se puede parametrizar.

Respuestas

La correcta ejecución de la petición se señaliza mediante los bits de realimentación (Página 11195)SYNC y POS_RCD activados.

Modo incremental relativo (posicionamiento relativo)

Descripción del funcionamiento

Con el modo incremental relativo se puede seguir moviendo el motor paso a paso durante un trayecto predefinido y llegar de ese modo a una posición concreta.

El sentido y la velocidad de desplazamiento se definen con el arranque.

Petición de desplazamiento para modo incremental relativo

La petición de desplazamiento contiene la siguiente información:

- Trayecto (número de impulsos a ejecutar)
- Multiplicador G para la velocidad/frecuencia de salida F_a
- Factor de reducción R para la frecuencia base F_b parametrizada
- Modo de operación =0 para modo incremental relativo
- Parada en la leva de referencia (Página 11179)
- Preajuste de sentido como inicio (Página 11192)

Nota

El 1STEP 5V comprueba los valores límite del recorrido definido (mínimo 1 impulso y máximo 16777215 impulsos). El 1STEP 5V no comprueba la distancia hasta el final de carrera. La marcha se detiene como máximo al llegar al final de carrera.

Respuestas

Cuando se inicia el modo incremental se desactiva el bit de realimentación (Página 11195)POS_RCD.

Durante el modo incremental, su ejecución se indica con el bit de realimentación POS activado.

Una vez finalizado correctamente el modo incremental, el bit de realimentación POS_RCD activado indica que se ha llegado a la posición.

Si el modo incremental se cancela, el bit de realimentación POS_RCD permanece desactivado. Si el valor de realimentación (Página 11195) está ajustado en "Distancia residual", una vez se detiene el modo incremental se indica el trayecto que queda por recorrer.

Modo incremental absoluto (posicionamiento absoluto) (a partir de 6ES7138-4DC01-0AB0)

Descripción del funcionamiento

Con el modo incremental absoluto se puede mover el motor paso a paso hasta una posición predefinida y llegar de ese modo a una posición concreta.

La velocidad se define con el arranque. El 1STEP 5V define automáticamente el sentido y el trayecto a recorrer con base en la posición de salida (valor real de la posición). También es posible predefinir el sentido de un eje de valor módulo.

Nota

Si se activa simultáneamente inicio adelante e inicio atrás (DIR_P y DIR_M) para un eje de valor módulo, el 1STEP 5V elegirá automáticamente el trayecto más corto hasta alcanzar (Página 11181) la posición de destino.

Petición de desplazamiento para modo incremental absoluto

La petición de desplazamiento contiene la siguiente información:

- Posición de destino
- Multiplicador G para la velocidad/frecuencia de salida F_a
- Factor de reducción R para la frecuencia base F_b parametrizada
- Modo de operación = 2 para modo incremental absoluto
- Libre preajuste de sentido como inicio (Página 11192)

Nota

El 1STEP 5V comprueba los valores límite de la posición definida (mínimo 0 y máximo 16777215). El valor final se puede parametrizar.

La petición de desplazamiento sólo se ejecuta si previamente se ha calculado o definido la posición del punto de referencia (el bit SYNC debe estar activado, ver "Búsqueda del punto de referencia (Página 11167)" o bien "Definición del punto de referencia (Página 11171)").

La señal de control (Página 11192) "Parada en leva de referencia" no se tiene en cuenta.

Respuestas

Cuando se inicia el modo incremental se desactiva el bit de realimentación (Página 11195)POS_RCD.

Durante el modo incremental, su ejecución se indica con el bit de realimentación POS activado.

Una vez finalizado correctamente el modo incremental, el bit de realimentación POS_RCD activado indica que se ha llegado a la posición.

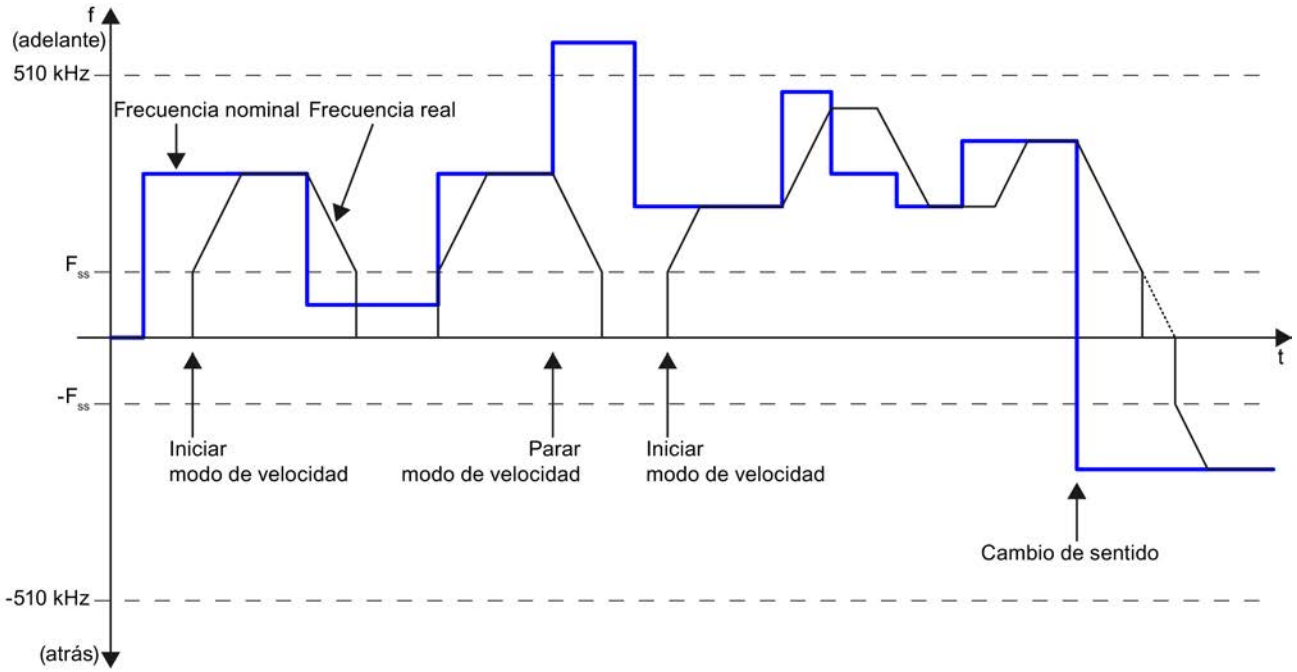
Si el modo incremental se cancela, el bit de realimentación POS_RCD permanece desactivado. Si el valor de realimentación (Página 11195) está ajustado en "Distancia residual", una vez se detiene el modo incremental se indica el trayecto que queda por recorrer.

Modo de velocidad (a partir de 6ES7138-4DC01-0AB0)

Descripción del funcionamiento

En este modo de operación se define la frecuencia con la que se deben generar los impulsos (pasos). Si se modifica la frecuencia, tras una fase de aceleración o deceleración se generan los impulsos con la nueva frecuencia. Los impulsos se generan de modo continuo hasta que se detenga la petición de desplazamiento o, en el caso de un eje lineal, hasta que se alcance un límite de la zona de desplazamiento.

La figura siguiente muestra el modo de velocidad con un eje de valor módulo.



Petición de desplazamiento para modo de velocidad

La petición de desplazamiento contiene la siguiente información:

- Frecuencia nominal como valor de 32 bits (tipo de datos STEP 7 REAL)
- Preajuste de sentido mediante el signo de la frecuencia nominal (positivo: adelante)
- Modo de operación = 3 para modo de velocidad
- Libre preajuste de sentido como inicio (Página 11192)

Nota

El 1STEP 5V comprueba los valores límite de la frecuencia definida (mínimo -510,0 kHz y máximo +510,0 kHz).

La frecuencia definida se alcanza con la aceleración a parametrizada teniendo en cuenta la frecuencia de inicio y parada F_{ss}. Si la frecuencia definida es menor que el valor F_{ss}, no se generarán impulsos.

La emisión continua de la frecuencia finaliza si surgen los siguientes eventos:

- Se alcanzan los límites de la zona de desplazamiento parametrizado (0 en sentido atrás), si no está parametrizado ningún eje de valor módulo
- Otras condiciones de interrupción para peticiones de desplazamiento (Página 11179)

Respuestas

Durante la tarea de desplazamiento, su ejecución se indica con el bit de realimentación (Página 11195)POS activado.

Si se define una nueva frecuencia, el bit de realimentación POS_RCD se borra. Una vez se alcanza la nueva frecuencia tras una fase de aceleración o deceleración, se vuelve a activar el bit de realimentación POS_RCD.

Si el valor de realimentación (Página 11195) está ajustado en "Frecuencia", la frecuencia actual se indica en la interfaz de realimentación como valor de 32 bits (tipo de datos STEP 7 REAL).

Detener la petición de desplazamiento

Parada selectiva de la petición de desplazamiento

- causada por	indicada por el bit de realimentación
STOP por bit de control	-
STOP externo en la entrada digital	STOP_EXT
Se ha alcanzado el final de carrera en sentido adelante (LIMIT_P o bien entrada digital)	STOP_LIMIT_P
Se ha alcanzado el final de carrera en sentido atrás (LIMIT_M o bien entrada digital)	STOP_LIMIT_M
Parada en leva de referencia	STOP_REF

Nota

Recuerde que los finales de carrera también se utilizan en el modo de operación Búsqueda del punto de referencia para la búsqueda de la leva de referencia.

Si una petición de desplazamiento se ha detenido en LIMIT_P, podrá retirar el eje del final de carrera con una nueva tarea de desplazamiento con DIR_M. Al detenerse en LIMIT_M podrá retirar el eje del final de carrera con DIR_P.

Parada en leva de referencia

Si al iniciar una marcha está seleccionada la función "Parada en leva de referencia" (el bit de control STOP_REF_EN está activado) y durante la marcha se reconoce la leva de referencia, se realiza una parada y la marcha finaliza.

Parada de la petición de desplazamiento en caso excepcional

En los siguientes casos se detiene la petición de desplazamiento con pérdida de sincronización:

- Control manual incorrecto en la interfaz de control con la tarea de desplazamiento en curso
- Fallo externo ERR_24V por sobrecarga en la alimentación del sensor (p. ej. cortocircuito)
- STOP de la CPU/del maestro
- Con eje lineal: se alcanzan los límites de la zona de desplazamiento

Efectos

Si se da una de las causas citadas que provocan la parada del posicionamiento en curso, éste finalizará con una rampa de deceleración.

Incluso en caso de parada excepcional de la tarea de desplazamiento, el valor de realimentación se sigue actualizando. Des este modo es posible recorrer la distancia residual en el momento de la parada mediante una nueva petición de desplazamiento en el modo incremental relativo.

Final de carrera y parada externa

Mediante la parametrización se especifica si para la parada externa y los finales de carrera se conectan contactos normalmente cerrados o normalmente abiertos.

- **Contacto normalmente cerrado significa:** la parada externa y el efecto de los finales de carrera se ejecutan con una señal 0. Al llegar a los finales de carrera se borra el bit de control correspondiente.
- **Contacto normalmente abierto significa:** la parada externa y el efecto de los finales de carrera se ejecutan con una señal 1. Al llegar a los finales de carrera se activa el bit de control correspondiente.

Nota

Al detenerse durante la fase de aceleración, el 1STEP 5V sigue emitiendo impulsos durante 50 ms como máximo con la frecuencia ya alcanzada antes de iniciar el proceso de frenado. Con ello se evitan cambios de frecuencia abruptos, que podrían provocar pérdidas de pasos.

Tipo de eje y área de desplazamiento (a partir de 6ES7138-4DC01-0AB0)

Resumen

Durante la parametrización se define el tipo de eje que va a controlar el motor paso a paso, el cual a su vez es controlado por el 1STEP 5V. Es posible elegir uno de los siguientes tipos de eje:

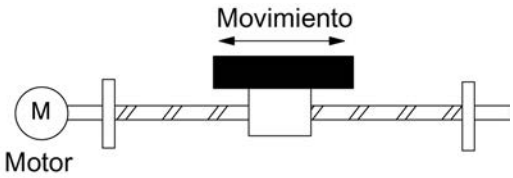
- Eje lineal
- Eje de valor módulo

Descripción del funcionamiento

Eje lineal

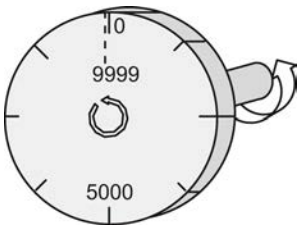
Es posible ajustar la zona de desplazamiento de un eje lineal. El límite inferior siempre es 0, el límite superior se parametriza y tiene un rango de valores entre 1 y 16777215.
La zona de desplazamiento se puede limitar aún más con finales de carrera (área de trabajo).

La figura siguiente muestra un eje lineal.



Eje de valor módulo

Un eje de valor módulo es una versión del eje rotativo. La figura siguiente muestra un eje de valor módulo.



Fin del eje de valor módulo

Con el parámetro "Zona de desplazamiento" se define el fin del eje de valor módulo.

El valor real de posición no puede alcanzar el valor de la zona de desplazamiento debido a que, desde el punto de vista físico, este valor máximo se encuentra en la misma posición que corresponde al inicio del eje de valor módulo (0).

Ejemplo:

Para la zona de desplazamiento se especifica el valor 10000, ver figura anterior.

Durante el movimiento hacia delante, el valor de posición salta de 9999 a 0 en la interfaz de realimentación y, durante el movimiento hacia atrás, de 0 a 9999.

Búsqueda del punto de referencia

Si durante la parametrización ha seleccionado el eje de valor módulo y el sistema de accionamiento está provisto de una leva de referencia, es posible llevar a cabo una búsqueda del punto de referencia (Página 11167).

Si después de generar un número de impulsos que corresponden a la zona de desplazamiento parametrizada no se encuentra la leva de referencia, se interrumpe la marcha. En ese caso permanecen borrados los bits de estado (Página 11195)SYNC y POS_RCD.

Definición del punto de referencia

La posición del punto de referencia sólo admite valores que van de 0 hasta el fin parametrizado de la zona de desplazamiento - 1.

Posicionamiento relativo

El fin de la zona de desplazamiento (fin del eje de valor módulo) puede sobrepasarse en ambos sentidos.

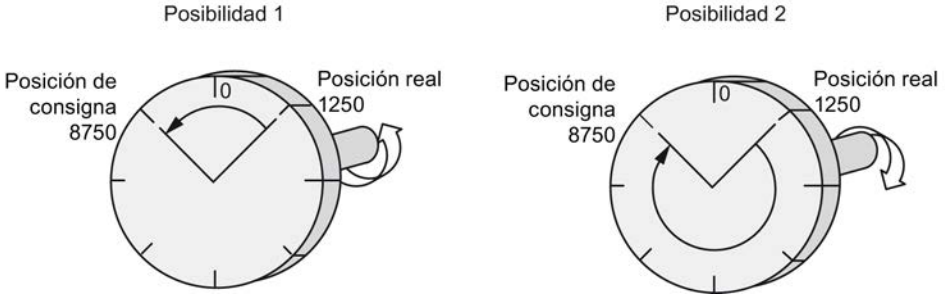
Posicionamiento absoluto

Si durante la parametrización se ha seleccionado el eje de valor módulo, para la posición de destino sólo pueden asignarse valores que van de 0 hasta el fin parametrizado de la zona de desplazamiento - 1.

A diferencia del eje lineal, indique al principio de la tarea de desplazamiento mediante la especificación de sentido (Página 11192) con qué recorrido debe alcanzarse la posición de destino:

- Inicio atrás (DIR_M): el 1STEP 5V llega a la posición de destino siguiendo la dirección en que los valores reales de la posición son más bajos (posibilidad 1 en la siguiente figura).
- Inicio adelante (DIR_P): el 1STEP 5V llega a la posición de destino siguiendo la dirección en que los valores reales de la posición son más altos (posibilidad 2 en la siguiente figura).
- Inicio adelante e inicio atrás simultáneamente (DIR_P y DIR_M): el 1STEP 5V elige automáticamente el trayecto más corto hasta alcanzar la posición de destino (posibilidad 1 en la siguiente figura).

La figura siguiente muestra el posicionamiento absoluto con un eje de valor módulo.



Habilitación de impulsos

Descripción del funcionamiento

La habilitación de impulsos habilita la emisión de impulsos desde el 1STEP 5V hasta la etapa de potencia. Sin la habilitación de impulsos no es posible realizar ninguna marcha.

Activar la habilitación de impulsos

La habilitación de impulsos se activa

- a través de la entrada digital DI0 con la "Función DI0" parametrizada como habilitación de impulsos externa (Página 11185);

o bien

- a través del bit de control DRV_EN con la "Función DI0" parametrizada como parada externa (Página 11185) o final de carrera adelante o atrás (Página 11185).

Se puede saber que la habilitación de impulsos está activada:

- porque el LED (Página 11198)RDY del 1STEP 5V luce cuando la parametrización es correcta;
- porque el bit de realimentación (Página 11195)STS_DRV_EN está activado.

Desactivar la habilitación de impulsos

Si la habilitación de impulsos se desactiva durante el desplazamiento, este se detiene inmediatamente, ya que no se emiten más impulsos a la etapa de potencia. La distancia residual y el valor real de la posición ya no son válidos. La sincronización del eje con el punto de referencia se pierde. El bit de realimentación SYNC y el LED RDY están borrados.

La desactivación de la habilitación de impulsos durante una parada provoca el borrado del bit de realimentación SYNC y la desactivación del LED RDY.

En este caso puede ser necesario llevar a cabo una búsqueda del punto de referencia (Página 11167).

Modificar parámetros durante el funcionamiento

Introducción

Algunos parámetros del 1STEP 5V se pueden modificar durante el funcionamiento sin que sea necesaria una reparametrización de toda la estación ET 200S.

Parámetros modificables

Los siguientes parámetros pueden modificarse:

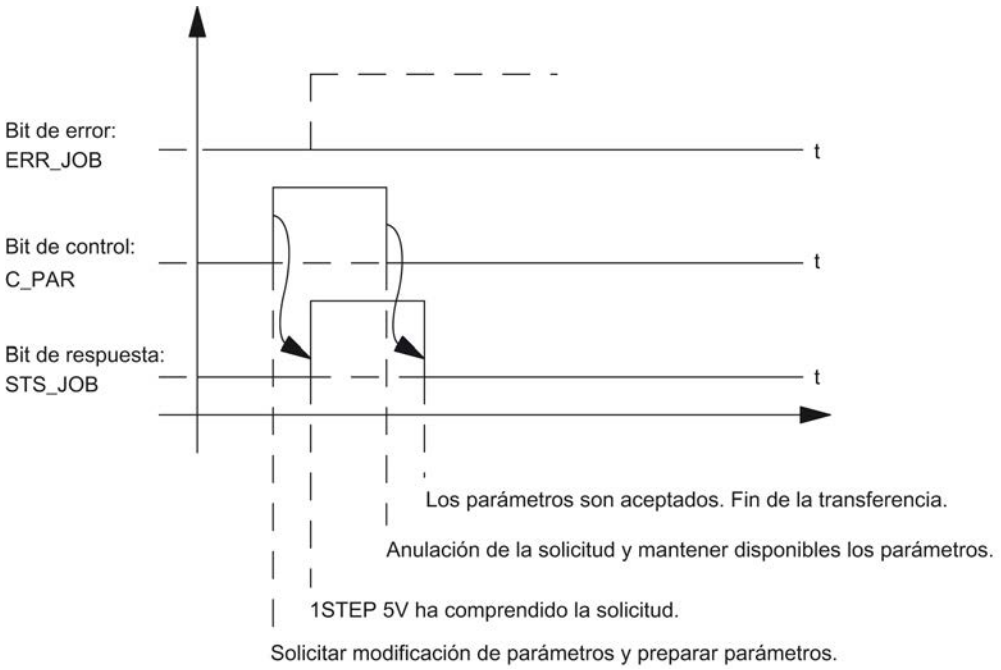
- Frecuencia base F_b:
- Multiplicador n para frecuencia de inicio y parada F_{ss}
- Multiplicador i para aceleración/deceleración
- Valor de realimentación en la interfaz de realimentación

Al iniciar la modificación de parámetros con el bit de control (Página 11192)C_PAR, se comprueban sus valores. Si se han aplicado valores no admisibles, se activa el bit de realimentación (Página 11195)ERR_JOB.

La orden de parametrización sólo afecta a los bits de realimentación correspondientes al desarrollo de la orden ERR_JOB y STS_JOB.

Modificación de parámetros

La figura siguiente muestra el proceso para la modificación de parámetros.



Nota

Sólo uno de los siguientes bits de control debe estar activado en cada momento:

DIR_P, DIR_M o C_PAR.

De lo contrario, se notifica el error ERR_JOB. El mensaje de error de la orden se borra cuando se inicia la siguiente orden.

Comportamiento de las entradas digitales

Introducción

Es posible parametrizar la función y la reacción (niveles activos) de las entradas digitales DI0 (3) y DI1 (7). Estos parámetros no pueden ser modificados desde el programa de usuario.

Entrada digital DI0 (3)

La función de la entrada digital DI0 (3) se puede parametrizar como:

- Habilitación de impulsos externa
- Parada externa
- Final de carrera +: en sentido ascendente
- Final de carrera -: en sentido descendente

Además, es posible parametrizar la reacción de la entrada digital DI0 (3) como:

- Contacto normalmente cerrado
- Contacto normalmente abierto

Entrada digital DI0 (3) como habilitación de impulsos externa

La entrada debe estar activada (Página 11179) durante el funcionamiento. Si la entrada está activada y no hay errores en la parametrización, el 1STEP 5V está preparado para el funcionamiento.

Entrada digital DI0 (3) como parada externa

Esta función de entrada permite detener (Página 11175) una tarea de desplazamiento en curso con una señal externa.

Entrada digital DI0 (3) como final de carrera positivo o negativo

Estas funciones de entrada le permiten limitar la zona de desplazamiento en sentido adelante o atrás con una señal externa. La señal tiene el mismo efecto que uno de los dos bits de control LIMIT_P o LIMIT_M en la interfaz de control (Página 11192).

Entrada digital DI1 (7)

La función de la entrada digital DI1 (7) se puede parametrizar como:

- Sensor punto de referencia: Leva de referencia
- Sensor de referencia y final de carrera +: en sentido ascendente
  Esta opción de parametrización sólo está disponible si la "Función DI0" no está parametrizada como "Final de carrera +".
- Sensor de referencia y final de carrera -: en sentido descendente
  Esta opción de parametrización sólo está disponible si la "Función DI0" no está parametrizada como "Final de carrera -".

Además, es posible parametrizar la reacción de la entrada digital DI1 (7) como:

- Contacto normalmente cerrado
- Contacto normalmente abierto

Entrada digital DI1 (7) como sensor punto de referencia

A esta entrada se conecta un sensor para la leva de referencia.

Se necesita una leva de referencia

- para una búsqueda del punto de referencia
- para un modo incremental con parada en la leva de referencia

Entrada digital DI1 (7) como sensor del punto de referencia y final de carrera positivo o negativo

Estas funciones de entrada permiten limitar adicionalmente la zona de desplazamiento en sentido adelante o atrás con la leva de referencia. La señal tiene además el mismo efecto que uno de los dos bits de control LIMIT_P o LIMIT_M en la interfaz de control (Página 11192).

Si se ha parametrizado la reacción de la entrada digital DI1 (7) como "Sensor del punto de referencia y final de carrera", el 1STEP 5V elige el sentido de arranque automáticamente hacia el final de carrera, independientemente del preajuste de sentido de la tarea de desplazamiento.

Reacción a STOP de la CPU/del maestro

Introducción

El 1STEP 5V detecta la parada de CPU/del maestro. Su reacción consiste en detener la tarea de desplazamiento (Página 11175) en curso.

Abandono del estado de parada de CPU/del maestro

Estación ET 200S	1STEP 5V
Sin reparametrización de la estación ET 200S	• La interfaz de realimentación del 1STEP 5V permanece actualizada. • Los valores modificados por orden de parametrización se mantienen. • Si en el momento de la parada de la CPU/del maestro había un bit de control activado (DIR_P, DIR_M, C_PAR), al abandonar el estado de parada de la CPU/del maestro los bits STS_JOB y ERR_JOB están activados. Borre el bit de control. La marcha / orden de parametrización no se ejecuta. A continuación puede iniciar una nueva marcha mediante el bit de control. • Tras la rampa de deceleración se desactivan la habilitación de impulsos, el LED RDY y el bit de estado SYNC.
Con reparametrización de la estación ET 200S	• Se reinician los datos sobre marchas y órdenes de parametrización anteriores. • Si en el momento de la parada de la CPU/del maestro la habilitación de impulsos estaba activada a través del bit de control DRV_EN, tras la rampa de deceleración se desactivan la habilitación de impulsos, el LED RDY y el bit de estado SYNC.

Reparametrización de la estación ET 200S

Una reparametrización de la estación ET 200S por medio de la CPU/del maestro DP tiene lugar en los siguientes casos:

- Conexión (POWER ON) de la CPU/el maestro DP
- Conexión (POWER ON) de IM 151 / IM 151 FO
- Tras fallo de la transmisión DP
- Tras cargar una parametrización o una configuración modificada de la estación ET 200S en la CPU/ el maestro DP
- Al insertar el 1STEP 5V
- Con conexión de alimentación o inserción del correspondiente módulo de potencia

Consulte también

Habilitación de impulsos (Página 11179)

Comportamiento de las entradas digitales (-4DC00-)

Entrada digital DI 3

La función de la entrada digital DI 3 se puede parametrizar como:

- Habilitación de impulsos externa
- Parada externa

Además, es posible parametrizar la reacción de la entrada digital DI 3 como:

- Contacto normalmente cerrado
- Contacto normalmente abierto

Entrada digital DI 3 como habilitación de impulsos externa

La entrada debe estar activada (Página 11188) durante el funcionamiento. Si se ha definido la entrada y no hay errores en la parametrización, el 1STEP 5V/204kHz está preparado para el funcionamiento.

Entrada digital DI 3 como parada externa

Esta función de entrada permite detener una tarea de desplazamiento en curso con una señal externa.

Entrada digital REF

A esta entrada se conecta un sensor para la leva de referencia.

Se necesita una leva de referencia

- para una búsqueda del punto de referencia
- para un modo incremental con parada en la leva de referencia

Habilitación de impulsos (-4DC00-)

Descripción del funcionamiento

La habilitación de impulsos habilita la emisión de impulsos desde el 1STEP 5V/204kHz hasta la etapa de potencia. Sin la habilitación de impulsos no es posible realizar ninguna marcha.

Activar la habilitación de impulsos

La habilitación de impulsos se activa

- a través de la entrada digital DI 3 con la "Función DI" parametrizada como habilitación de impulsos externa (Página 11184);

o bien

- a través del bit de control DRV_EN con la "Función DI" parametrizada como parada externa (Página 11184)

Se puede saber que la habilitación de impulsos está activada:

- porque el LED (Página 11198)RDY del 1STEP 5V/204kHz luce cuando la parametrización es correcta;
- porque está activado el bit de retroalimentación STS_DRV_EN.

Desactivar la habilitación de impulsos

Si la habilitación de impulsos se desactiva durante la marcha, ésta se detiene inmediatamente, ya que no se emiten más impulsos a la etapa de potencia. El motor paso a paso termina sin control de impulsos. De este modo, la distancia residual ya no es válida. La sincronización del eje con el punto de referencia se pierde. El bit de retroalimentación SYNC y el LED RDY están borrados.

La desactivación de la habilitación de impulsos durante una parada provoca el borrado del bit de retroalimentación SYNC y la desactivación del LED RDY.

En este caso puede que sea necesario realizar una búsqueda del punto de referencia.

Configurar y parametrizar 1STEP 5V

Parámetros

Activar diagnóstico de grupo

Diagnóstico de grupo

El diagnóstico de grupo se activa en la casilla de verificación correspondiente.

"Cortocircuito en la alimentación del sensor" ERR_24V o "Error de parametrización" ERR_PARA con diagnóstico de grupo habilitado provocan un diagnóstico de canal.

Seleccionar frecuencia base

Frecuencia base F_b

La frecuencia base F_b es el valor base para el ajuste de la frecuencia de inicio y parada, la frecuencia de salida y la aceleración/deceleración.

Seleccione la frecuencia base adecuada para su aplicación.

Encontrará información detallada al respecto en Ajuste de la frecuencia base (Página 11164).

Definir el multiplicador n

Multiplicador n

El multiplicador n permite ajustar la frecuencia de inicio y parada de forma escalonada:

$F_{ss} = F_b \times n$

Rango de valores permitidos: 1 … 255

Consulte también

Ajuste de la frecuencia base (Página 11164)

Definir el intervalo de tiempo i

Intervalo de tiempo i

El multiplicador i permite ajustar la aceleración/deceleración de forma escalonada:

$a = F_b / (i \times 0{,}128\ ms)$

Rango de valores permitidos: 1 … 255

Consulte también

Ajuste de la frecuencia base (Página 11164)

Seleccionar el valor de realimentación

Valor de realimentación

Seleccione qué valor de realimentación estará disponible en la interfaz de realimentación (Página 11195):

- Distancia residual
- Posición
- Frecuencia

Seleccionar función DI0

Función DI0

Seleccione qué función debe tener la entrada digital DI0 (3) (Página 11181):

- Habilitación de impulsos externa
- Parada externa

- Final de carrera en sentido adelante
- Final de carrera en sentido atrás

Seleccionar función DI1

Función DI1

Seleccione qué función debe tener la entrada digital DI1 (7) (Página 11181):

- Sensor punto de referencia (leva de referencia)
- Sensor punto de referencia y final de carrera en sentido adelante
  Esta opción sólo está disponible si la "Función DI0" no está parametrizada como "Final de carrera adelante".
- Sensor punto de referencia y final de carrera en sentido atrás
  Esta opción sólo está disponible si la "Función DI0" no está parametrizada como "Final de carrera atrás".

Seleccionar la reacción de las entradas DI0 y DI1

Entrada DI0, entrada DI1

Seleccione la reacción de las entradas digitales DI0 (3) y DI1 (7):

- Contacto normalmente cerrado
- Contacto normalmente abierto

Consulte también

Comportamiento de las entradas digitales (Página 11181)

Detener la petición de desplazamiento (Página 11175)

Seleccionar la reacción de final de carrera en la interfaz de control

Final de carrera en la interfaz de control

Seleccione la reacción del final de carrera en la interfaz de control (Página 11192):

- Contacto normalmente cerrado
- Contacto normalmente abierto

Consulte también

Detener la petición de desplazamiento (Página 11175)

Activar eje de valor módulo

Eje de valor módulo

En el 1STEP 5V está preajustado el tipo de eje "Eje lineal". Si desea controlar un "Eje de valor módulo", debe activar la casilla de verificación "Eje de valor módulo".

Consulte también

Tipo de eje y área de desplazamiento (a partir de 6ES7138-4DC01-0AB0) (Página 11177)

Definir la zona de desplazamiento

Zona de desplazamiento

El fin del eje de valor módulo se define con el parámetro "Zona de desplazamiento".

Rango de valores permitidos: 1 … 16777216

Consulte también

Tipo de eje y área de desplazamiento (a partir de 6ES7138-4DC01-0AB0) (Página 11177)

Interfaz de control y respuesta del 1STEP 5V

Interfaz de control del 1STEP 5V

Nota

Para el 1STEP 5V los datos siguientes de la interfaz de control están interrelacionados, es decir, son datos coherentes:

- Bytes 0 … 3
- Bytes 4 … 7

Asignación de la interfaz de control

La tabla siguiente muestra la asignación de la interfaz de control (salidas) del 1STEP 5V.

Dirección	Asignación	
Bytes 0 a 3	**Modo incremental relativo, modo incremental absoluto**	
	Byte 0	Multiplicador G: $F_a = F_b \times R \times G$ (rango de valores 1 … 255)
	Byte 1	Trayecto o posición bit 23 … bit 16
	Byte 2	Trayecto o posición bit 15 … bit 8
	Byte 3	Trayecto o posición bit 7 … bit 0

Dirección	Asignación		
	Búsqueda del punto de referencia		
	Byte 0	Multiplicador G: $F_a = F_b \times R \times G$ (rango de valores 1 … 255)	
	Byte 1	Posición bit 23 … bit 16	
	Byte 2	Posición bit 15 … bit 8	
	Byte 3	Posición bit 7 … bit 0	
	Definición del punto de referencia		
	Byte 0	Reservado = 0	
	Byte 1	Posición bit 23 … bit 16	
	Byte 2	Posición bit 15 … bit 8	
	Byte 3	Posición bit 7 … bit 0	
	Modo de velocidad		
	Bytes 0 a 3	Frecuencia como tipo de datos STEP 7 REAL	
	Orden de parametrización		
	Byte 0	Reservado = 0	
	Byte 1	Multiplicador i: $a = F_b \times R / (i \times 0{,}128\ ms)$ (rango de valores 1 … 255)	
	Byte 2	Multiplicador n: $F_{ss} = F_b \times n \times R$ (rango de valores 1 … 255)	
	Byte 3	Frecuencia base F_b: • 0 = 800 Hz • 1 = 400 Hz • 2 = 200 Hz • 3 = 80 Hz • 4 = 40 Hz • 5 = 20 Hz • 6 = 8 Hz • 7 = 4 Hz • 8 = 2000 Hz	
Byte 4	Bit 7	Factor de reducción • 0 = factor 1,0 (sin reducción) • 1 = factor 0,1	R
	Bit 6	Detener la tarea de desplazamiento	STOP
	Bit 5	Inicio atrás	DIR_M
	Bit 4	Inicio adelante	DIR_P
	Bit 3	Reservado = 0	
	De bit 2 a bit 0	Modo de operación • 0 = modo incremental relativo (posicionamiento relativo) • 1 = búsqueda del punto de referencia • 2 = modo incremental absoluto (posicionamiento absoluto) • 3 = modo de velocidad • 4 = definición del punto de referencia	MODE
Byte 5	Bit 7	Acuse de error de diagnóstico	EXTF_ACK
	Bit 6	Modificar parámetros	C_PAR

Dirección	Asignación		
	De bit 5 a bit 4	Valor de realimentación en la interfaz de realimentación • 00 = distancia residual • 01 = posición • 10 = frecuencia • 11 = reservado	FEEDBACK
	Bit 3	Parada en leva de referencia	STOP_REF_EN
	Bit 2	Habilitación de impulsos	DRV_EN
	Bit 1	Final de carrera en sentido adelante	LIMIT_P
	Bit 0	Final de carrera en sentido atrás	LIMIT_M
Byte 6	Reservado = 0		
Byte 7	Reservado = 0		

Aclaraciones sobre los bits de control

Bits	Aclaración
Frecuencia base F_b	Codificación para el ajuste escalonado de la frecuencia base: • 0 = 800 Hz • 1 = 400 Hz • 2 = 200 Hz • 3 = 80 Hz • 4 = 40 Hz • 5 = 20 Hz • 6 = 8 Hz • 7 = 4 Hz • 8 = 2000 Hz
Modo de operación	Codificación para modo de operación: • 0 = modo incremental relativo (posicionamiento relativo) • 1 = búsqueda del punto de referencia • 2 = modo incremental absoluto (posicionamiento absoluto) • 3 = modo de velocidad • 4 = definición del punto de referencia
C_PAR	Con este bit se solicita una modificación de parámetros.
DIR_M	Con este bit se solicita y se inicia una tarea de desplazamiento en sentido hacia atrás.
DIR_P	Con este bit se solicita y se inicia una tarea de desplazamiento en sentido hacia delante.
DRV_EN	Si se utiliza la entrada digital DI0 (3) como parada externa, este bit se valora como habilitación de impulsos.
Final de carrera LIMIT_M	Este final de carrera limita la zona de desplazamiento en sentido hacia atrás. Este bit se activa o se borra en el programa de usuario.
Final de carrera LIMIT_P	Este final de carrera limita la zona de desplazamiento en sentido hacia delante. Este bit se activa o se borra en el programa de usuario.
EXTF_ACK	Bit de acuse para aviso de diagnóstico

Bits	Aclaración
FEEDBACK	Codificación para el valor de realimentación en la interfaz de realimentación: • 00 = distancia residual • 01 = posición • 10 = frecuencia • 11 = reservado
Frecuencia	Valor de 32 bits (tipo de datos STEP 7 REAL) que contiene la frecuencia de impulsos que se va a generar.
Multiplicador G	Factor para el ajuste escalonado de la velocidad/frecuencia de salida
Multiplicador i	Factor para el ajuste escalonado de la aceleración/deceleración
Multiplicador n	Factor para el ajuste escalonado de la frecuencia de inicio y de parada
Posición	Valor de 24 bits que contiene la posición de destino que se debe alcanzar
Factor de reducción R	Con el bit activado la frecuencia base F_b se multiplica por 0,1. Con ello se reducen en igual medida la frecuencia de salida F_a, la frecuencia de inicio y parada F_{ss} y la aceleración/deceleración a.
STOP	Este bit permite detener en cualquier momento una tarea de desplazamiento con una rampa de deceleración (ver "Detener la tarea de desplazamiento (Página 11175)").
STOP_REF_EN	Con el bit activado la función "Parada en leva de referencia" está activa. Cuando se detecta la leva de referencia, la tarea de desplazamiento se detiene con una rampa de deceleración (ver "Detener la tarea de desplazamiento (Página 11175)").
Trayecto	Valor de 24 bits que contiene el número de impulsos que deben avanzarse (sin signo)

Interfaz de realimentación del 1STEP 5V

Nota

Para el 1STEP 5V los datos siguientes de la interfaz de realimentación están interrelacionados, es decir, son datos coherentes:

- Bytes 0 … 3
- Bytes 4 … 7

Asignación de la interfaz de realimentación

La tabla siguiente muestra la asignación de la interfaz de realimentación (entradas) del 1STEP 5V.

Dirección	Asignación		
Bytes 0 a 3	Distancia residual (bit 23 … bit 0 de 32 bits)/ Posición (bit 23 … bit 0 de 32 bits)/ Frecuencia (32 bits, tipo de datos STEP 7 REAL)		
Byte 4	Bit 7	Cortocircuito en la alimentación del sensor	ERR_24V
	Bit 6	Reservado = 0	
	Bit 5	Error de parametrización	ERR_PARA
	Bit 4	Punto de referencia calculado	SYNC
	Bit 3	Reservado = 0	
	Bit 2	Posición alcanzada	POS_RCD

Dirección	Asignación		
	Bit 1	Error en la transmisión de la orden	ERR_JOB
	Bit 0	Transmisión de la orden en curso	STS_JOB
Byte 5	Bit 7	Tarea de desplazamiento en curso	POS
	Bit 6	El final de carrera adelante es la causa de la parada	STOP_LIMIT_P
	Bit 5	El final de carrera atrás es la causa de la parada	STOP_LIMIT_M
	Bit 4	La parada externa es la causa de la parada	STOP_EXT
	Bit 3	La leva de referencia es la causa de la parada	STOP_REF
	Bit 2	Estado DI0	STS_DI0
	Bit 1	Estado DI1	STS_DI1
	Bit 0	Estado habilitación de impulso activada	STS_DRV_EN
Byte 6	Número de error en caso de error en la transmisión de la orden		
Byte 7	Reservado = 0		

Aclaraciones sobre los bits de realimentación

Tabla 11-13 Bits de realimentación

Bits	Aclaración
Frecuencia	Valor de 32 bits (tipo de datos STEP 7 REAL) que contiene la frecuencia de impulsos actual. La frecuencia de impulsos puede presentar valores positivos (sentido adelante) y negativos (sentido atrás).
ERR_JOB	Este bit se activa cuando la orden no es clara o no es posible. La causa del error se especifica mediante el número de error que se devuelve (ver la tabla "Número de error en la interfaz de realimentación" que aparece más abajo).
ERR_PARA	Hay parámetros incorrectos en la parametrización de la estación ET 200S. La causa del error se especifica mediante el número de error que se devuelve (ver la tabla "Número de error en la interfaz de realimentación" que aparece más abajo). El bit erróneo del parámetro se borra al transferir una parametrización correcta.
ERR_24V	La alimentación del encóder se ha sobrecargado (p. ej. por un cortocircuito) y actualmente está desconectada. ERR_24V se pone a cero si ha sido acusado con el bit de control EXTF_ACK. Una vez eliminada la sobrecarga, se vuelve a conectar la alimentación del encóder y ERR_24V permanece borrado.
Número de error	Indica la causa del error si ERR_JOB o ERR_PARA están activados (ver la tabla "Número de error en la interfaz de realimentación" que aparece más abajo).
POS	Desplazamiento: este bit permanece activado mientras la tarea de desplazamiento está en curso.
POS_RCD	Al iniciar un modo incremental o al especificar una nueva frecuencia nominal en el modo de velocidad, se borra POS_RCD. Una vez finalizado correctamente el modo incremental o alcanzada la frecuencia nominal en el modo de velocidad, se activa POS_RCD. Si la marcha se interrumpe (por parada de la tarea de desplazamiento (Página 11175) o por borrado de la habilitación de impulsos (Página 11179)), POS_RCD permanece borrado.
Posición	Valor de 24 bits que contiene la posición absoluta actual (sin signo). El byte 0 de la interfaz de realimentación es 0.
Distancia residual	Valor de 24 bits que contiene el número de impulsos que todavía deben avanzarse (sin signo). El byte 0 de la interfaz de realimentación es 0.

Bits	Aclaración
STOP_EXT	Causa de la parada: este bit permanece activado si la tarea de desplazamiento se ha detenido con una parada externa.
STOP_LIMIT_M	Causa de la parada: este bit permanece activado si la tarea de desplazamiento se ha detenido al alcanzar el final de carrera atrás.
STOP_LIMIT_P	Causa de la parada: este bit permanece activado si la tarea de desplazamiento se ha detenido al alcanzar el final de carrera adelante.
STOP_REF	Causa de la parada: este bit permanece activado si la tarea de desplazamiento se ha detenido al alcanzar la leva de referencia.
STS_DI0	El bit indica el estado de la entrada digital DI0 (3).
STS_DI1	El bit indica el estado de la entrada digital DI1 (7).
STS_DRV_EN	Este bit está activado si, dependiendo de la función parametrizada de la entrada digital DI0, • la habilitación de impulsos externa está activada o bien • el bit de control DRV_EN para la habilitación de impulsos está activado.
STS_JOB	Este bit se activa como respuesta al detectar una solicitud de tarea de desplazamiento o parametrización y después se reinicia una vez ejecutada la orden.
SYNC	Este bit se activa tras una búsqueda del punto de referencia finalizada correctamente o tras definir manualmente el punto de referencia. Tras la parametrización con nuevos parámetros de la estación ET 200S o tras el borrado de la habilitación de impulsos, el bit SYNC está borrado.

Número de error

Si en la interfaz de realimentación se indica un error en la transmisión de la orden (ERR_JOB activado) o un error en la parametrización básica (ERR_PARA activado), la causa de dicho error se especifica mediante un número de error.

La tabla siguiente muestra los números de error de la interfaz de realimentación.

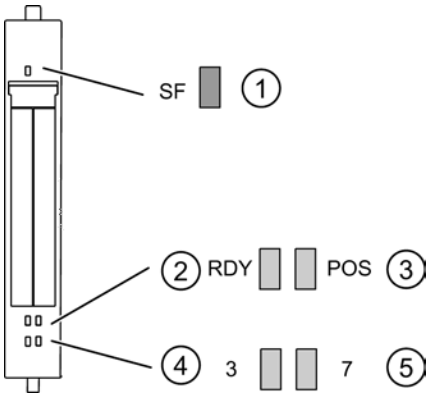
Número de error	Significado
Causas de error generales	
0	Ningún error (en este caso ERR_JOB o ERR_PARA no están activados)
1	La combinación de bits de control no es admisible (DIR_P, DIR_M, C_PAR)
2	Otra tarea ya está en curso
Causas de error en una tarea de desplazamiento	
16	Inicio adelante (DIR_P) con final de carrera adelante (LIMIT_P) activado
17	Inicio atrás (DIR_M) con final de carrera atrás (LIMIT_M) activado
18	Inicio con el bit de control de parada activado
19	Inicio con parada externa activada
20	Inicio aunque falte la habilitación de impulsos (interna o externa)
21	Inicio con STOP_REF_EN activado estando la leva de referencia activada
22	Inicio sin referencia (en modo incremental absoluto)
23	Inicio con error de diagnóstico pendiente
24	Inicio interrumpido por parada de CPU/maestro
25	Inicio con el modo de operación equivocado (difiere de la solicitud)

Número de error	Significado
26	Trayecto o datos de la posición no admisibles
27	Multiplicador G para la velocidad es cero
28	Frecuencia en modo de velocidad no admisible
Causas de error en una orden de parametrización o en la parametrización básica	
32	Los datos de la frecuencia base no son admisibles
33	El multiplicador n para la frecuencia de inicio y parada es cero
34	El multiplicador i para la aceleración/deceleración es cero
35	El valor de realimentación para la interfaz de realimentación no es admisible
36	Combinación de funciones de DI0 y DI1 no admisible (final de carrera)
37	Los datos correspondientes al fin de la zona de desplazamiento no son admisibles

Diagnóstico del 1STEP 5V

Diagnóstico con indicadores LED

Indicador LED del 1STEP 5V



① Error de grupo (rojo)
② Listo para tarea de desplazamiento (verde)
③ Posicionamiento en curso (verde)
④ Indicador de estado de la entrada digital 0 (verde)
⑤ Indicador de estado de la entrada digital 1 (verde)

Indicadores de estado y error mediante los LEDs del 1STEP 5V

La tabla siguiente muestra la indicación de estado y error del 1STEP 5V.

Evento (LED)					Causa	Solución
SF	RDY	POS	3	7		
ON					No hay parametrización o el módulo enchufado es incorrecto. Hay un aviso de diagnóstico.	Compruebe la parametrización. Evalúe el aviso de diagnóstico.
	ON				si el módulo está parametrizado correctamente y hay habilitación de pulsos	
		ON			si la tarea de desplazamiento está en curso	
			ON		DI 0 está activada	
				ON	DI 1 está activada	

Tipos de error

En el manual de producto del módulo de interfaz utilizado en la estación ET 200S encontrará información sobre la estructura del diagnóstico de canal.

Tipos de error del 1STEP 5V

La tabla siguiente muestra los tipos de error del 1STEP 5V.

Tipo de error		Significado	Solución
1_D	00001: Cortocircuito	Cortocircuito en la alimentación de sensor.	Compruebe el cableado de los interruptores. Corregir el cableado del proceso.
9_D	01001: Error	Ha aparecido un error interno en el módulo.	Sustituir el módulo.
16_D	10000: Error de parametrización	Módulo no parametrizado.	Corregir la parametrización.

11.4.2.9 Uso del IM 174

Vista general

Módulo de interfaz para accionamientos analógicos y motores paso a paso

Un módulo IM 174 (Interface Module) es un módulo de interfaz, capaz de accionar hasta 4 accionamientos con interfaz analógica de consigna con un encóder TTL o SSI por eje en el PROFIBUS isócrono.

La comunicación entre el controlador y el IM 174 se realiza exclusivamente a través de PROFIBUS, a través de un tipo de telegrama específico del IM 174, el cual contiene, junto a los datos digitales de entrada/salida por cada accionamiento un tipo de telegrama especificado según el perfil PROFIDrive (telegrama estándar 3 y 81). En el marco de la comunicación cíclica DP se transmiten al controlador los valores reales del accionamiento (valores de encóder) del módulo IM 174 a través del PROFIBUS isócrono y las consignas de velocidad calculadas por el controlador al módulo IM 174.

Las consignas de velocidad transmitidas son emitidas a continuación por el módulo IM 174 como valores analógicos o impulsos a los accionamientos.

Campo de aplicación

El IM 174 realiza las siguientes tareas de control:

- Controlar accionamientos analógicos
- Controlar motores paso a paso
- Leer datos de encóder

Configuración y programación del IM174

Configuración del IM174

Pasos necesarios - Descripción general

El IM 174 se configura en el área "Propiedades" de la ventana de inspección. Para ello es preciso ejecutar los siguientes pasos:

1. Agregar un módulo PROFIBUS a la configuración hardware
2. Parametrizar PROFIBUS isócrono
3. Definir parámetros del accionamiento y encóder para los ejes

Parametrizar PROFIBUS isócrono

1. En el área "Dirección PROFIBUS" seleccione la red PROFIBUS y asigne una dirección PROFIBUS.

   Nota

   Dirección PROFIBUS en el interruptor DIL

   En el módulo PROFIBUS IM 174 la dirección PROFIBUS se ajusta a través de interruptores DIL. Para que una dirección PROFIBUS ajustada en el interruptor DIL esté activa, es preciso realizar una transición de apagado y encendido del IM 174. La dirección PROFIBUS asignada debe coincidir con la dirección PROFIBUS ajustada en el interruptor DIL.

2. En el área "Modo isócrono" active la casilla de verificación "Sincronizar esclavo DP con ciclo DP equidistante" y ajuste los tiempos Ti/To.

3. En el área "Encóder y accionamientos" seleccione el tipo de telegrama (Página 11210).
4. En el área "Direcciones" defina las áreas de direcciones para los parámetros de entrada y salida de los ejes y asigne una imagen de proceso.

Definir parámetros de accionamiento y de encóder para los ejes

1. En el área "Encóder y accionamientos" seleccione para cada eje eltipo de accionamiento (Página 11201) y ajuste los parámetros de accionamiento (Página 11202).
2. En el área "Encóder y accionamientos" seleccione para cada eje el tipo de encóder (Página 11201) y, en caso necesario, ajuste los parámetros de encóder (Página 11206) necesarios.
3. En el área "Parámetros" puede ajustar los parámetros (Página 11209) para el comportamiento de desconexión.

Encóder y accionamientos

La propiedad "Encóder y accionamientos" de la navegación local permite parametrizar los accionamientos y encóders conectados al IM 174.

Tipos de accionamientos

Se pueden seleccionar los siguientes tipos de accionamientos:

- Servomotor (accionamiento analógico)
- Stepper (motor paso a paso)

Tipos de encóder para accionamientos analógicos

Para los servomotores se pueden seleccionar los siguientes tipos de encóder

- Tipo de encóder no disponible
- Encóder TTL
- Encóder SSI

Tipos de encóders para motores paso a paso

Para los motores paso a paso se pueden seleccionar los siguientes tipos de encóder

- Tipo de encóder no disponible
- Encóder TTL
- Encóder SSI
- Encóder Stepper

Parámetros de accionamiento

Tipos de accionamiento

Se pueden seleccionar los siguientes tipos de accionamiento:

- Servo (accionamiento analógico)
- Stepper (motor paso a paso)

Tipo de accionamiento Servo

Si ha seleccionado el tipo de accionamiento Servo, puede cambiar el rango de tensión de la tensión de salida analógica con el botón de opción "Unipolar".

Unipolar no seleccionado

Si la casilla "Unipolar" **no** está activada, se emite como consigna una señal analógica con una tensión comprendida en el rango de **-10V** a **+10V**. Esto permite desplazar el accionamiento en dos sentidos.

Unipolar seleccionado

Si la casilla "Unipolar" **no** está activada, se emite como consigna una señal analógica con una tensión comprendida en el rango de **0V** a **+10V**. El IM 174 define a continuación el sentido de giro en función de la consigna actual de velocidad, a través de una salida digital del IM 174:

- Señal de sentido de giro para el eje 1 -> salida digital X11, Pin 13
- Señal de sentido de giro para el eje 2 -> salida digital X11, Pin 15
- Señal de sentido de giro para el eje 3 -> salida digital X11, Pin 17
- Señal de sentido de giro para el eje 4 -> salida digital X11, Pin 19

Alt.DrvRdy

En función del accionamiento que se utilice, la señal actual Drive Ready emitirá distintos avisos de estado:

- El accionamiento notifica el estado "Listo para conexión" mediante la señal Drive Ready. Para conectar el accionamiento, es necesario habilitarlo por medio del controlador.
- El accionamiento notifica el estado "Disponibilidad" mediante la señal Drive Ready. En este estado, el accionamiento está listo para ser regulado y se guía directamente por una consigna.

No active la función "Alt.DrvRdy" si su accionamiento emite el estado "Listo para conexión" con la señal Drive Ready (por lo general accionamientos servo).

Active la función "Alt.DrvRdy" si su accionamiento emite el estado "Disponibilidad" con la señal Drive Ready (por lo general accionamientos Stepper).

En la información del producto del accionamiento se ofrecen datos sobre el aviso de estado que emitirá el accionamiento utilizado.

Tipo de accionamiento "Stepper"

Para accionamientos "Stepper" en los campos "Frecuencia máx. [Hz]" y "Frecuencia norm. [Hz]" es preciso introducir frecuencias [Hz] específicas del accionamiento o del proceso. Para poder operar un motor Stepper con un IM 174 se requiere la electrónica de potencia correspondiente (p. ej. FM Stepdrive), así como el motor Stepper.

En la siguiente figura se representa una configuración básica de un IM 174 con accionamientos Stepper con y sin encóder:

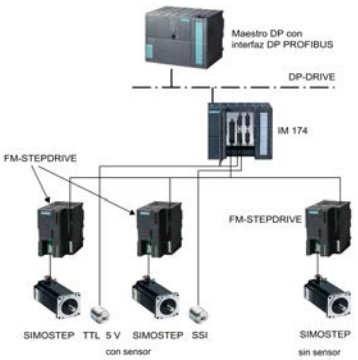


Figura 11-11 Estructura básica de un IM 174 con FM-STEPDRIVE y motor SIMOSTEP

Alt.DrvRdy

En función del accionamiento que utilice, la señal actual Drive Ready emitirá distintos avisos de estado:

- El accionamiento emite el estado "disponible" mediante la señal Drive Ready. Para conectar el accionamiento, es necesario habilitarlo por medio del controlador.
- El accionamiento emite el estado "Disponibilidad" mediante la señal Drive Ready. En este estado, el accionamiento está listo para ser regulado y se guía directamente por una consigna.

No active la función "Alt.DrvRdy" si su accionamiento emite el estado "Listo para conexión" con la señal Drive Ready (por lo general accionamientos servo).

Active la función "Alt.DrvRdy" si su accionamiento emite el estado "Disponibilidad" con la señal Drive Ready (por lo general accionamientos Stepper).

En la información del producto del accionamiento se ofrecen datos sobre el aviso de estado que emitirá el accionamiento utilizado.

Frecuencia máx. [Hz] y frecuencia norm. [Hz]

En los campos "Frecuencia máx. [Hz]" (frecuencia máxima) y "Frecuencia normal [Hz]" (frecuencia nominal) es preciso introducir frecuencias [Hz] específicas del accionamiento o del proceso.

Cálculo de la frecuencia normal

La frecuencia normal se calcula con la fórmula:

Frecuencia normal [Hz] = n [r/min] / 60 * Resolución en el motor paso a paso

n [r/min]:	Velocidad del motor Stepper (los valores característicos oscilan entre 500 y 1000 revoluciones por minuto)
Resolución en el motor paso a paso:	Número de incrementos en el motor Stepper (valores característicos: 500, 1000, 5000)

Cálculo de la frecuencia máx.

La frecuencia máx. se calcula con la fórmula:

Frecuencia mas [Hz] = $n_{máx}$ [r/min] / 60 * Resolución en el motor paso a paso

n_{max} [r/min]:	Velocidad máxima del motor Stepper (los valores característicos oscilan entre 500 y 1000 revoluciones por minuto)
Resolución en el motor paso a paso:	Número de incrementos en el motor Stepper (valores característicos: 500, 1000, 5000)

La velocidad máxima n_{max} puede determinarse en base a las curvas características de funcionamiento

de los motores Stepper. En este caso, con el momento M_t tecnológicamente necesario hay que

determinar una velocidad máxima n_{max} del motor que se corresponda con la velocidad máxima

del eje.

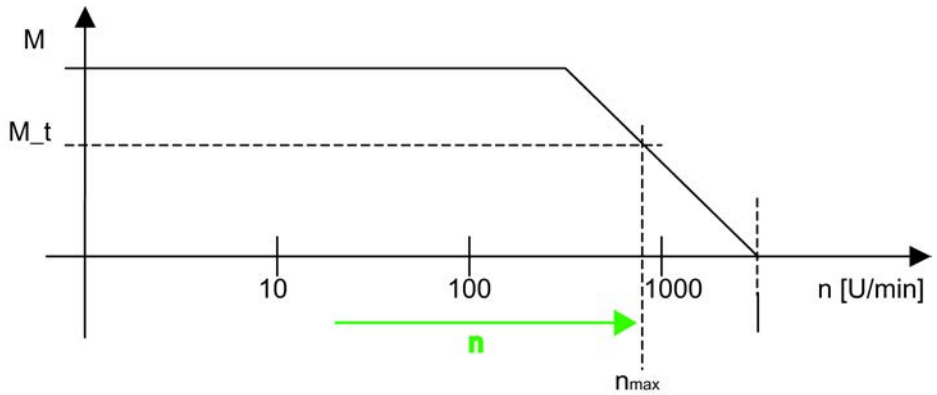


Figura 11-12 Curvas características de un motor paso a paso

Nota

Zona de sobrecarga

Para evitar un funcionamiento en la zona de sobrecarga utilice el accionamiento exclusivamente en la frecuencia norm.

Descripción del comportamiento de un motor Stepper

La precisión de la posición, la velocidad n y el par M desarrollado por el motor resultan decisivos para exigencias específicas de procesos. Se debe tener en cuenta el comportamiento del motor Stepper para poder determinar dichos valores óptimamente. A partir de una determinada velocidad del motor Stepper (aprox. a 500U/min), el par desarrollado por el motor decae de forma logarítmica, tendiendo a 0 a partir a velocidad máxima (aprox. a 3000U/min). Consulte los datos específicos en la hoja de datos del motor utilizado.

Seguimiento del modo de funcionamiento en la zona de sobrecarga

Si el motor Stepper no es capaz de desarrollar el momento requerido, se desincroniza con respecto a la frecuencia predeterminada y su velocidad se colapsa. Ello puede provocar incluso su parada. En este estado sólo se puede realizar un nuevo registro de movimiento después de una consigna temporal predeterminada de 0. En un eje de posicionamiento sin sensor adicional, la posición del recorrido se pierde y con ello la sincronización del eje.

Nota

Accionamiento paso a paso con un sensor tipo "Stepper" (modo PULSE REFEED)

Si se opera con un sensor del tipo "Stepper" (modo PULSE REFEED) en estado de parada puede oscilar el motor paso a paso.

Si el proyecto está parametrizado con los ajustes por defecto, esto puede ocurrir cuando el motor paso a paso, debido a un movimiento de desplazamiento previo, se ha llevado a una posición que queda exactamente entre dos pasos naturales del mismo.

Motor paso a paso con encóders TTL / SSI

Si se usan encóders TTL / SSI es necesario atender a la relación entre resolución del encóder (incl. resolución fina) y la resolución de pasos del motor.

Para evitar la oscilación del motor entre dos valores de posición cuando está parado, el incremento del motor paso a paso debe ser inferior, en distancia, a la representación de la realimentación del sensor (es decir, el incremento del motor paso a paso debe ser superior a la resolución del sensor). Dado el caso debe aplicarse un sensor con menor resolución o aumentar la resolución de pasos en el motor paso a paso.

Parámetros de sensores

Tipos de encóder para accionamientos analógicos

Para accionamientos analógicos es posible seleccionar los siguientes tipos de encóder:

- Tipo de sensor no existe
- Encóder TTL
- Sensor SSI

Tipos de encóder para accionamientos paso a paso

Para accionamientos paso a paso es posible seleccionar los siguientes tipos de encóder:

- Tipo de sensor no existe
- Encóder TTL
- Sensor SSI
- Encóder Stepper

Tipo de sensor "Ninguno"

Para el **tipo de accionamiento "Servo"** se aplica: El eje x no existe o no debe utilizarse. En el telegrama PROFIBUS sólo se transmiten datos de usuario vacíos para este eje.

Para el **tipo de accionamiento "Stepper"** se aplica: El Stepper se opera sin sensor.

Si el Stepper está parametrizado para el modo de funcionamiento sin sensor, sólo se autoriza la utilización como eje de velocidad.

Si no se utiliza un sensor (SSI/TTL), es posible seleccionar el tipo de encóder "Stepper" para permitir un posicionamiento, incluso si no se ha conectado ningún sensor.

Encóder TTL

En el encóder tipo "TTL" se pueden ajustar los siguientes parámetros:

- **Resolución**: Resolución del sensor expresada en impulsos por vuelta
- **Cálculo de la velocidad activado**: IM 174 calcula la velocidad si la casilla de verificación se encuentra activada.

- **Velocidad normalizada**: Este campo de entrada sólo es visible si se ha activado la opción "Cálculo de la velocidad activado". Indique la velocidad nominal del motor en r/min. Con el cálculo de la velocidad a través del IM 174, la velocidad se calcula con menos ruido.
- **Bits reservados para una resolución precisa**: Ajuste 0 - 15
  A través del parámetro "Bits reservados para resolución fina" se indica la multiplicación de impulsos deseada para los incrementos transferidos en la señal de valor real del sensor G1_XIST1 y G1_XIST2.
  El ajuste representa una multiplicación de impulso entre 2^0 = 1 hasta 2^{15} = 32768.

Nota

Resolución fina mínima para sensores TTL

En los sensores TTL se debe considerar y ajustar siempre una resolución fina mínima de 2^2(=4).

Transmitted actual encoder value G1_XIST1 or G1_XIST2

31		0
Overflows	Current encoder increments	Reserved bits for fine resolution

Sensor SSI

En el enc´der tipo "SSI" se pueden ajustar los siguientes parámetros:

- **Paridad**: Se debe seleccionar esta opción si los datos del encóder se transmiten al IM174 con bit de paridad.
- **Longitud del telegrama**: Ajuste 0 - 25
  Número de bits de datos de usuario enviados por el encóder
- **Encoding**: Se soportan los siguientes tipos de codificación del enc´der:
  - Binary
  - Gray

- **Velocidad de transferencia**: La velocidad ajustada debe ser idéntica en todos los encóders SSI. Si hay ajustadas diferentes velocidades, se adopta la velocidad del encóder SSI que tenga asignado el número de encóder más alto.
  Se soportan las siguientes velocidades:
  - 187,5 kbits/s
  - 375 kbits/s
  - 750 kbits/s
  - 1,5 Mbits/s
  - 3,0 Mbits/s

  Nota

  Ajuste como máximo 187,5 kbits/s. No es conveniente asignar a este módulo velocidades más altas.

- **Bits reservados para una resolución fina**: Ajuste 0 - 15
  A través del parámetro "Bits reservados para resolución fina" se indica la multiplicación de impulsos deseada para los incrementos transferidos en la señal de valor real del sensor G1_XIST1 y G1_XIST2.
  El ajuste representa una multiplicación de impulso entre 2^0 = 1 hasta 2^{15} = 32768.

Sensor Stepper

Accionamiento en modo PULSE REFEED

Un eje x con el tipo de accionamiento "Stepper" se puede operar con un sensor tipo "Stepper". En este modo de operación, el IM 174 retroalimenta las consignas al controlador en forma de valores reales.

En el encóder tipo "Stepper" se pueden ajustar los siguientes parámetros:

- **Vigilancia del motor**: Activando la casilla de verificación "Vigilancia motor" se supervisa continuamente la cantidad de pasos de un motor Stepper con base en la distancia al punto de referencia definido (ver "Distancia Bero"). Cuando se trabaja sin sensor, el accionamiento Stepper sirve también para realimentar el valor real (pulse refeed).
- **Distancia Bero**: En "Distancia Bero" se introduce la cantidad de pasos (para el accionamiento Stepper usado) entre dos señales Bero. El recorrido entre las dos señales Bero es la distancia al punto de referencia.
- **Tolerancia Bero**: Ajuste 0 - 65535
  En "Tolerancia Bero" se introduce la desviación de pasos permitida con respecto a la distancia Bero. El rango del número de pasos resultante se encuentra dentro de la siguiente tolerancia:
  Rango del número de pasos resultante = Distancia Bero ± 1/2 * Tolerancia Bero
- **Bits reservados para una resolución precisa**: Ajuste 0 - 15
  A través del parámetro "Bits reservados para resolución fina" se indica la multiplicación de impulsos deseada para los incrementos transferidos en la señal de valor real del sensor G1_XIST1 y G1_XIST2.
  El ajuste representa una multiplicación de impulso entre 2^0 = 1 hasta 2^{15} = 32768.

Parámetros

La propiedad "Parámetros" de la navegación local permite definir parámetros importantes para el IM 174.

Rampa de parada [ms]

El rango de valores oscila entre 0 y 65535 ms.

A través del parámetro "Rampa de parada" se puede definir una función lineal en el tiempo que, en caso de detectarse un fallo, haga que todos los accionamientos conectados al IM 174 se frenen hasta la consigna 0.

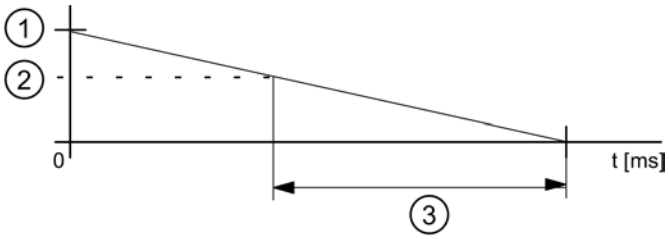
La rampa de parada actúa tras los fallos siguientes:

- Alarma de temperatura del módulo (aprox. 90°C CON, aprox. 85°C DES)
- Errores de señales de vida
- Error de sincronización entre maestro y esclavo

Nota

Valor de parámetro 0

En presencia de un fallo, un valor de parámetro 0 hace que el accionamiento se pare de forma natural.



1) Consigna máxima
2) Consigna actual
3) Valor del parámetro: Rampa de parada

Figura 11-13 Parámetros: Rampa de parada

Tiempo de retardo de apagado

El rango de valores oscila entre 0 y 65535 s.

El parámetro "Tiempo de retardo de apagado" permite especificar una temporización tras la que, en caso de que se produzca una **alarma de temperatura,** se activa la rampa de parada.

El tiempo de retardo de apagado inicia al superarse una temperatura de 90 °C en el módulo. Si mientras está activo el tiempo de retardo la temperatura baja a aprox. 85 °C, entonces no se activa la rampa de parada y se desactiva el tiempo de retardo de apagado.

Errores de señales de vida tolerables

El rango de valores oscila entre 0 y 15.

Con el parámetro "Errores de señales de vida tolerables" se especifica el número de errores de señales de vida tolerables del maestro DP. Si se sobrepasa el número parametrizado se activa la rampa de parada. El rango de valores se encuentra en

Modo conforme 611U

En el modo conforme 611U el origen de la señal para referenciar los ejes ya no se especifica mediante el telegrama estándar PROFIDrive (STD3, palabra de control del encóder G1_STW) sino a través de la palabra digital de salida adicional contenida en el telegrama PROFIBUS del IM 174 (ver tabla "Estructura de telegrama" en el apartado "Tipo de telegrama").

Modo conforme 611U:

- Desactivado
  La fuente de la señal para el referenciado se especifica a través de la palabra de control del encóder Gx_STW en el telegrama estándar PROFIDrive.
- Activado
  La fuente de la señal para el referenciado se especifica a través de la palabra digital de salida adicional contenida en el telegrama PROFIBUS.

Seleccionar tipo de telegrama

Tipo de telegrama

El Esclavo DP IM 174 opera con dos tipos de telegramas específicos: cuatro ejes con un sensor cada uno (telegrama estándar 3 y 81) y datos E/S.

Nota

Telegrama estándar 81

El telegrama estándar 81 sólo puede utilizarse para sensores.

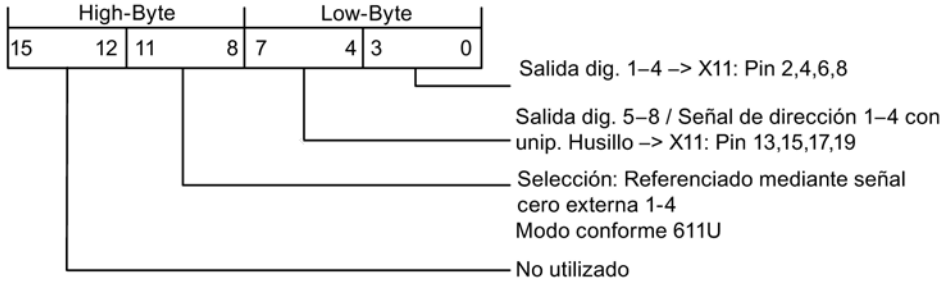
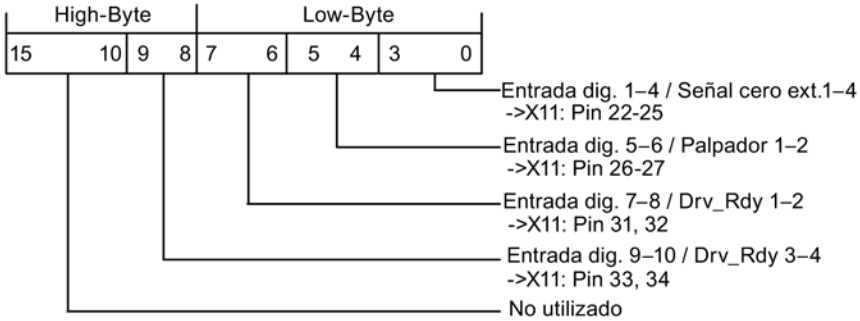
Estructura de telegrama

Estructura de telegrama

La siguiente visión global explica la estructura del telegrama.

Tabla 11-14

Tipo de telegrama	Descripción
4 ejes con un sensor cada uno, telegrama estándar 3 + EA, PZD-5/9 A/E 1/1 o telegrama 81 + EA, PZD-2/6 + A/E 1/1	4 telegramas estándar 3 ó 81 y por cada uno una palabra PZD para datos E/S digitales
PZD x/y Número de datos de proceso/palabras, x: consigna, y: valor real, p. ej. PZD-5/9: 5 datos de proceso/palabras para consignas 9 datos de proceso/palabras para valores reales	
Estructura de telegrama del IM174 Eje1: STD 3 ó STD 81; Eje2: STD 3; Eje3: STD 3; Eje4: STD 3; E/S: A-Word — Consignas (maestro -> esclavo) Low — High Eje1: STD 3 ó STD 81; Eje2: STD 3; Eje3: STD 3; Eje4: STD 3; E/S: E-Word — Valores reales (esclavo -> maestro) STD 3: Telegrama estándar 3 conforme la especificación PROFIDrive V3.0 A-Word: datos de salida digitales (16 bit) E-Word: datos de entrada digitales (16 bit)	
Telegrama estándar 3: Interfaz consigna de velocidad 32bit con 1 sensor PZD1: STW1; PZD1–PZD3: NSOLL_B; PZD4: STW2; PZD5: G1_STW — Consigna (maestro -> esclavo) Low — High PZD1: ZSW1; PZD2–PZD3: NIST_B; PZD4: ZSW2; PZD5: G1_ZSW; PZD6–PZD7: G1_XIST1 PZD8–PZD9: G1_XIST2 — Valor real (esclavo -> maestro)	
Telegrama estándar 81: Interfaz consigna de velocidad 32bit con 1 sensor PZD1: STW2; PZD2: G1_STW1 — Consigna (maestro -> esclavo) PZD1: ZSW2; PZD2: G1_ZSTW1; PZD3–PZD4: G1_XIST1; PZD5–PZD6: G1_XIST2 — Valor real (esclavo -> maestro)	

Tipo de telegrama	Descripción
A-Word (datos de salida dig. 16bit)	
	
E-Word (datos de entrada dig. 16bit)	
	

Nota

El tipo de telegrama del esclavo DP IM 174 ajustado debe coincidir con el tipo de telegrama ajustado en el controlador. No tiene lugar una sincronización automática.

Palabra de control del sensor Gx_STW

La siguiente visión global contiene una descripción de la palabra de control del encóder (extracto). La visión global se refiere a los siguientes requisitos:

- Búsqueda de la marca de referencia
- Medida al vuelo
- Fallo de sensor

Bit	Identifier		Signal state/description		
0	Find reference mark or On-the-fly measurement	Functions	Bit 7 = 0 => Request: Find reference mark		
			Bit	Meaning	Homing using:
			0	Function 1:	Encoder zero mark (except in "611 U conform mode")
			1	Function 2:	Not used
			2	Function 3:	Not used
			3	Function 4:	Not used
1			Bit 7 = 1 => Request: On-the-fly measurement		
			Bit	Meaning	
			0	Function 1:	Measuring input 1 rising edge
			1	Function 2:	Measuring input 1 falling edge
			2	Function 3:	Measuring input 2 rising edge
			3	Function 4:	Measuring input 2 falling edge
2			Note • Bit x = 1 Function requested Bit x = 0 Function not requested • If more than one function is enabled, the values for all functions cannot be read until all functions have ended and this has been signaled via the relevant status bit (G1_ZSW, Bit 0 - Bit 3 = 0). • On-the-fly measurement The rising and falling edges of the measuring input can be enabled simultaneously. The measuring input signal is detected according to the direction of the signal change. The measured values are read out consecutively.		
3					
4		Command	Bits 6, 5, 4	Meaning	
			000	--	
5			001	Enable function x	
6			010	Read value x	
			011	Cancel function x	
7		Mode	0	Find reference mark	
			1	On-the-fly measurement	
15	Acknowledge encoder error		0 1	-- Acknowledge encoder error	

Nota

Medición de flancos ascendentes o descendentes

El IM 174 soporta únicamente la medición de flancos ascendentes o descendentes.

Valor real adicional del sensor Gx_XIST2

La siguiente visión global describe los códigos de error en Gx_XIST2 con G1_ZSW, Bit15 == 1:

Tabla 11-15 Códigos de error en Gx_XIST2

G1_XIST2	Significado	Causas posibles/descripción
1_{Hex}	Error de suma del sensor	El nivel de señal de los sensores es demasiado bajo, presenta fallos (blindaje insuficiente) o la vigilancia de rotura de hilo se ha activado.
2_{Hex}	Vigilancia de señales cero	Entre dos señales cero del sensor ha surgido una fluctuación en la medición de la posición del rotor (posiblemente se han perdido 2 rayas del sensor).

Palabra de estado ZSW_1 - bit 11 a bit 14

La siguiente visión global describe la palabra de estado ZSW_1 en relación a los avisos específicos del módulo bit 11 a bit 14:

Tabla 11-16 Palabra de estado ZSW_1

Bit	Nombre	Descripción
11	Error de temperatura	Se ha excedido una temperatura determinada en la carcasa del módulo IM 174.
12	Error de sincronización PLL	El IM 174 no puede sincronizarse con la señal de control global.
13	Error Master Life Sign	El IM 174 no puede sincronizarse con la señal de vida maestra.
14	Error de accionamiento	Se ha producido un error de accionamiento, p. ej. falta la señal Drive Ready.

Vigilancia de la marca cero

Funcionamiento de la vigilancia de la marca cero del sensor

La vigilancia de la marca cero compara los incrementos detectados entre dos marcas cero con la resolución del sensor configurada. Si se detecta una diferencia se notifica un error del sensor.

Las pistas A y B, así como la pista Z del sensor, deben encontrarse simultáneamente en el nivel High, para que la vigilancia de la marca cero puede ejecutarse correctamente.

Limitaciones:

- El fallo de una marca cero en sí no provoca irremediablemente un error, ya que la comprobación solo se ejecuta al alcanzar la marca cero.
- El resultado de los incrementos MOD 16 contados debe ser distinto de cero para notificar un error.
- El resultado de los incrementos MOD 10 contados debe ser distinto de cero para notificar un error.

Ejemplo:

Configuración del sensor: rayas TTL 2048

incrementos del sensor contados 2049 -> error del sensor

incrementos del sensor contados 1024 -> ningún error del sensor

Programar IM 174

Para la programación del IM 174 se recomienda el paquete opcional "Easy Motion Control".

Instrucciones para IM 174

La lista de instrucciones Easy Motion Control incluye las siguientes instrucciones para IM 174:

- EncoderIM174 - Uso del IM 174 como módulo de lectura de recorrido para encóders incrementales y absolutos
- OutputIM174 - Uso de un canal del módulo IM 174 como módulo de salida

11.4.2.10 Uso de Easy Motion Control

Posicionamiento con Easy Motion Control

Descripción del producto

Descripción general

Easy Motion Control permite posicionar accionamientos de forma regulada. El movimiento se controla exclusivamente por software (instrucciones) en la CPU. Para la conexión con los accionamientos **no se requiere hardware especial**, pero para detectar los valores de posición real y para emitir las consignas de velocidad es obvio que debe estar presente la periferia correspondiente.

Las instrucciones de Easy Motion Control se han elaborado de conformidad con la "Technical Specification Function blocks for motion control V1.0" de **PLCopen**.

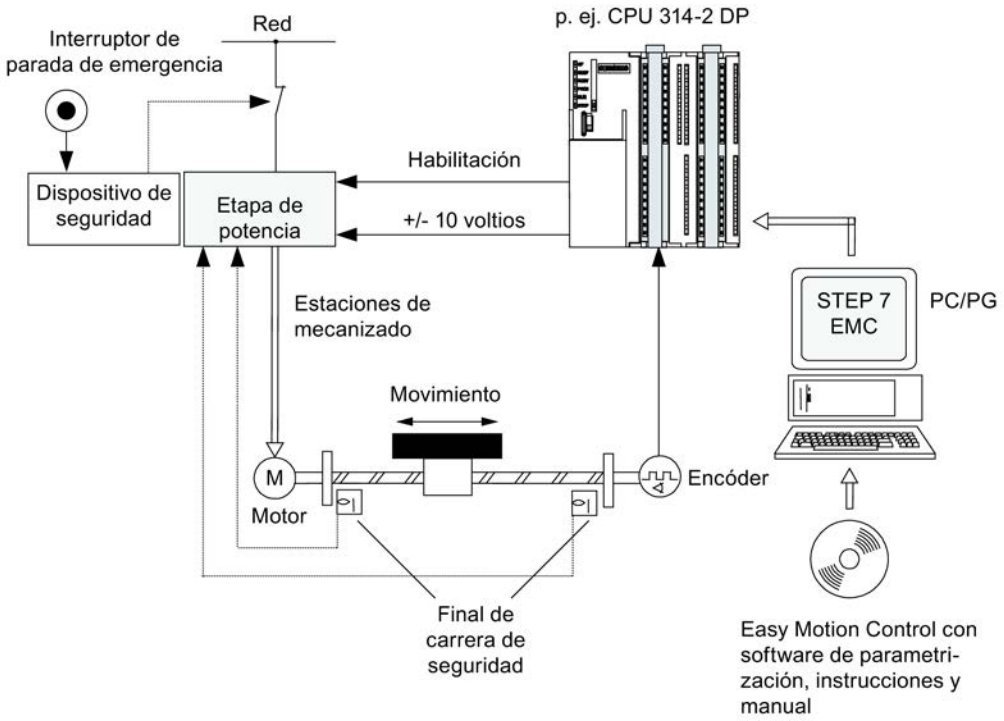
La parametrización se lleva a cabo mediante un software de configuración que también ofrece soporte en la puesta en marcha mediante funciones de prueba. Todos los parámetros de un eje se encuentran en un bloque de datos, el DB de eje.

Los accionamientos pueden conectarse a periferia centralizada o bien, a través de un maestro DP/controlador IO, a periferia descentralizada.

El desplazamiento se realiza interconectando las instrucciones correspondientes y ejecutándolas en el programa de usuario.

Componentes para el posicionamiento

La figura siguiente muestra una configuración de ejemplo con Easy Motion Control.



Dispositivo de seguridad

Cuando se dispara el dispositivo de seguridad, el interruptor de parada de emergencia desconecta la alimentación.

Etapa de potencia

La etapa de potencia controla el motor. Los finales de carrera de seguridad desconectan la etapa de potencia.

Motor

El motor es controlado por la etapa de potencia y acciona el eje.

Encóder

El encóder proporciona información de recorrido y sentido.

Easy Motion Control

Easy Motion Control ofrece las siguientes funciones:

- JOG
- Búsqueda del punto de referencia
- Posicionamiento absoluto/relativo
- Reductor electrónico
- Posicionamiento del punto de referencia
- Todas las funciones de vigilancia habituales para posicionamiento
- Corrección de velocidad
- Regulación de posición
- Simulación
- Drivers de entrada preconfeccionados para registro de valores del encóder o drivers de salida para módulos de salidas analógicas

Módulos para lectura de recorrido y salida de consigna

Los módulos constituyen la interfaz entre la CPU y la máquina. Leen las posiciones del eje y envían consignas a la etapa de potencia.

CPU

La CPU ejecuta el programa de usuario que incorpora las instrucciones de Easy Motion Control.

PC/PG

Un PC o programadora sirve para

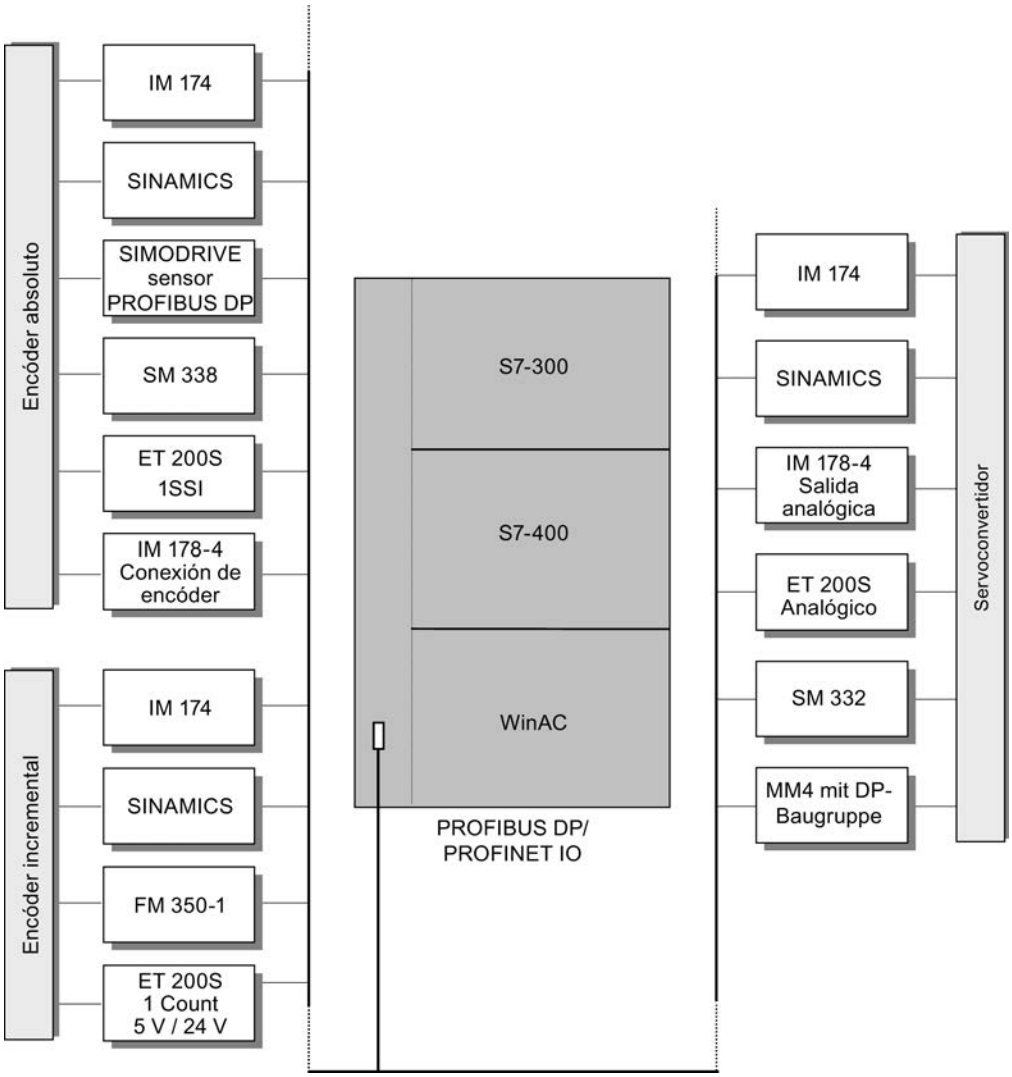
- Parametrizar: Easy Motion Control se parametriza con el software de configuración o directamente en el bloque de datos de eje (ver Configuración del objeto tecnológico Eje (Página 11228)).
- Programar: Las instrucciones de Easy Motion Control se incorporan directamente en el programa.

Configuraciones hardware admitidas

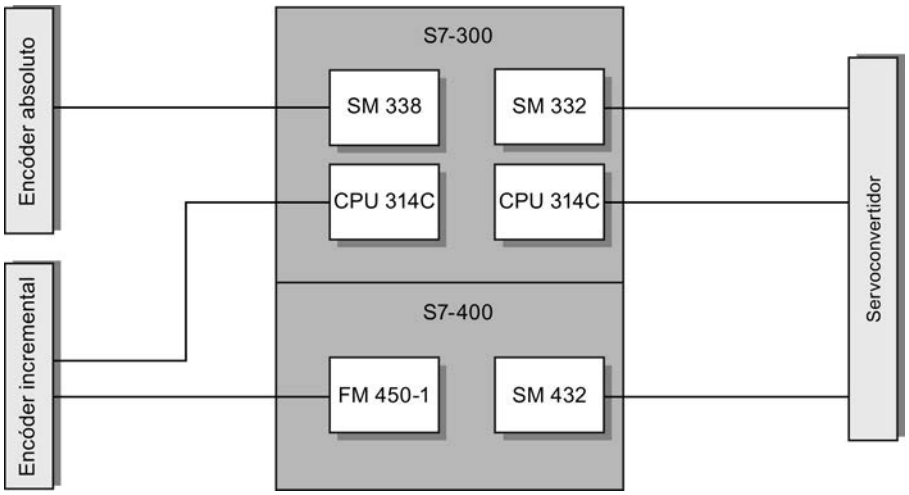
Las configuraciones hardware representadas a continuación son soportadas cómodamente por instrucciones de driver. Estas instrucciones se suministran con Easy Motion Control.

La adaptación a configuraciones que no son soportadas directamente es posible mediante instrucciones de driver libres que se suministran de igual forma.

La figura siguiente muestra configuraciones **descentralizadas** para Easy Motion Control.



La figura siguiente muestra configuraciones **centralizadas** para Easy Motion Control.

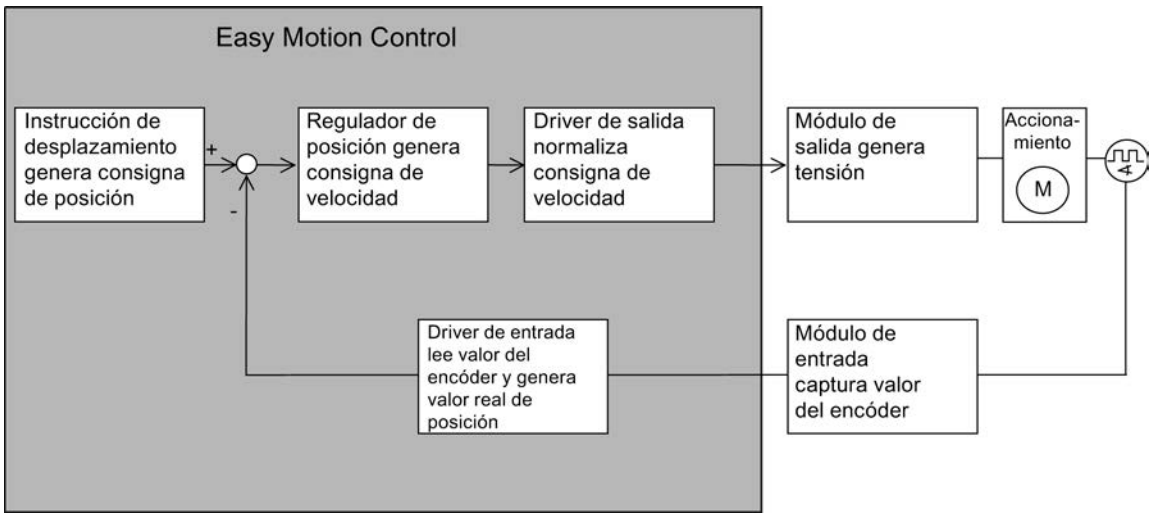


Principios básicos del posicionamiento con Easy Motion Control

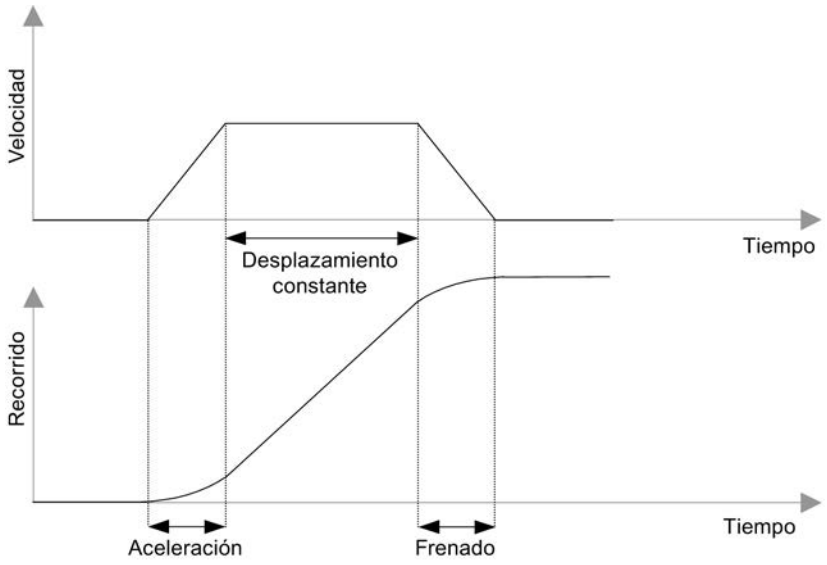
Diseño de la regulación de posición

En el posicionamiento regulado, la posición actual del eje se compara con la posición de consigna. Las desviaciones se corrigen por medio del regulador de posición durante el desplazamiento y también en parada.

La figura siguiente muestra el esquema de principio para el posicionamiento regulado con Easy Motion Control. Una instrucción de desplazamiento (Página 11252) especifica consignas de posición. El driver de entrada (Página 7348) detecta la posición real del eje. El regulador de posición (Página 11238) compara el valor de la posición real con la consigna y genera una consigna de velocidad a partir de la diferencia. El driver de salida (Página 7372) emite la consigna de velocidad normalizada.



La figura siguiente muestra la relación entre la velocidad y el trayecto recorrido por el eje durante un desplazamiento.



Definiciones de términos

Destino

Define la posición absoluta o relativa del eje que se alcanza en el posicionamiento.

Desplazamiento

Los desplazamientos son iniciados por la llamada de instrucciones, que en adelante llamaremos instrucciones de desplazamiento.

Distancia residual

La distancia residual es la diferencia entre la posición de destino y la consigna de posición actual.

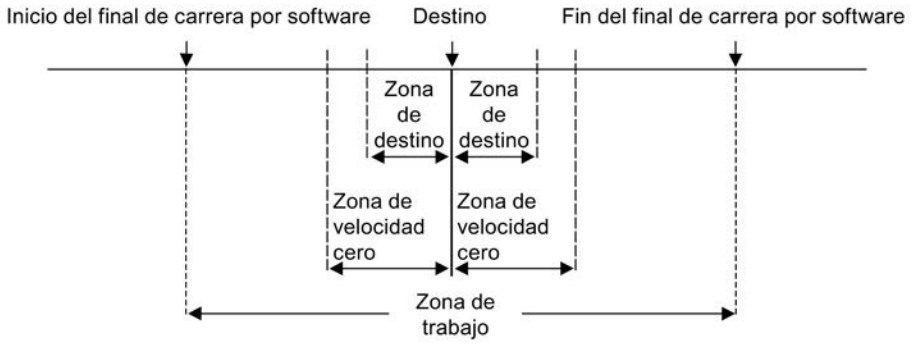
Error de seguimiento

Durante un desplazamiento, debido a la inercia del eje, se produce una diferencia entre la consigna de posición actual y la posición real. Esta diferencia se denomina error de seguimiento. Si la diferencia supera un valor parametrizable, se produce un error de seguimiento.

Zonas de vigilancia

- Zona de trabajo:
  Define la zona de desplazamiento permitida del eje, la cual se determina para la tarea en cuestión con ayuda de los finales de carrera por software (Página 11221) o con el final deleje rotativo (Página 11222).
- Zona de velocidad cero:
  Define una zona parametrizable que el eje no puede abandonar a velocidad cero. La zona de velocidad cero se encuentra a izquierda y derecha del punto de destino.
- Zona de destino:
  Cuando la consigna alcanza el destino al final de un desplazamiento, se vigila la entrada del eje en la zona de destino. La zona de destino sirve para vigilar la precisión de posicionamiento de la aplicación en cuestión, y se encuentra a izquierda y derecha del punto de destino.
  Nada más alcanzarse la zona de destino comienza la vigilancia de la zona de velocidad cero. Por consiguiente, la zona de velocidad cero debe ser mayor que la zona de destino.

La figura siguiente muestra la ubicación de las distintas zonas en torno al punto de destino.



Eje lineal

Descripción

La **zona de trabajo** va desde el inicio del final de carrera por software hasta el fin del final de carrera por software, y está limitada a 1×2^{24} unidades de longitud (en la zona de coordenadas entre -2^{24} y $+2^{24}$).

La **longitud física del eje** está limitada por los topes mecánicos del eje.

El **rango numérico** de Easy Motion Control tan solo es relevante si la vigilancia de la zona de trabajo está desactivada. Está limitado a 2×2^{24} unidades de longitud (de $-2^{24}+6$ a $+2^{24}-6$).

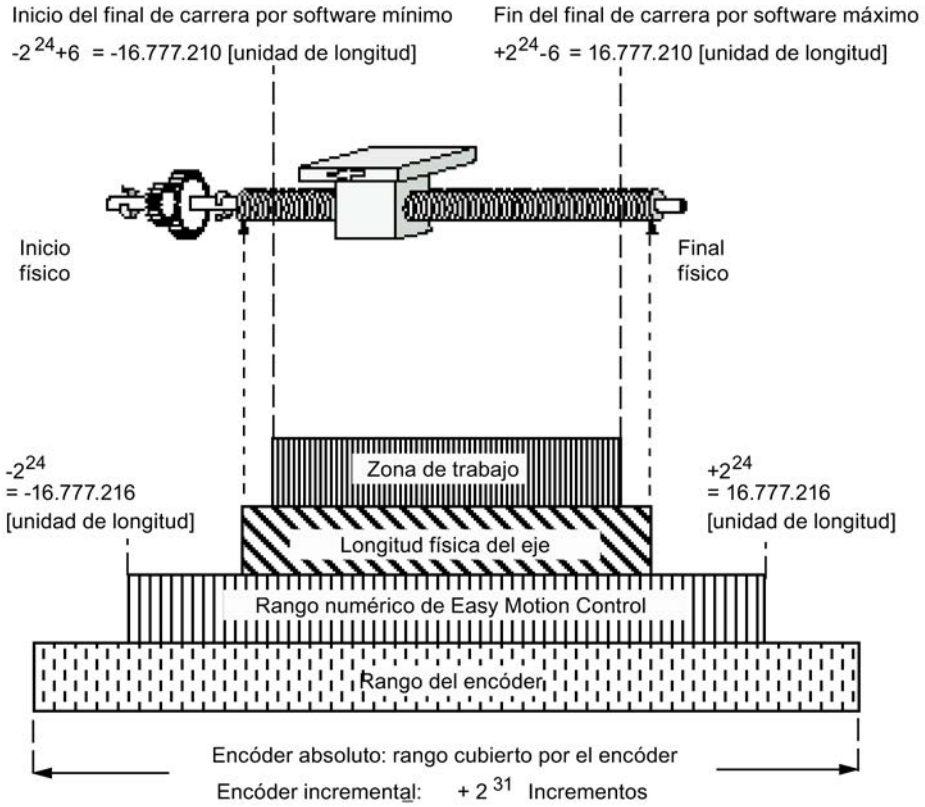
El **rango del encóder** es

- en el caso de un encóder absoluto, el rango abarcado por el encóder.
- en el caso de un encóder incremental, el trayecto desde -2^{31} hasta $+2^{31}$ incrementos.

En los ejes lineales, la **zona de trabajo** (= zona de desplazamiento) está limitada de la manera siguiente:

Zona de trabajo < Longitud física del eje < Rango numérico de Easy Motion Control < Rango del encóder

La figura siguiente muestra las diferentes zonas de un eje lineal.



Eje rotativo

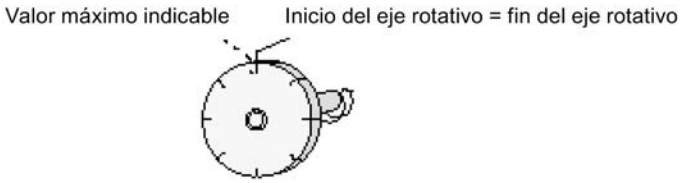
Descripción

La **zona de trabajo** va desde el principio del eje rotativo hasta el final del eje rotativo, y está limitada a 1 × 2^{24} unidades de longitud (en la zona de coordenadas entre -2^{24} y $+2^{24}$).

El **rango del encóder**

- en el caso de un encóder incremental, puede ser cualquiera.
- en el caso de un encóder absoluto, viene determinado por lo siguiente:
  Rango del encóder = zona de trabajo × n
  con:
  – n = 1 para encóders monovuelta
  – n > 1 y valor entero para encóders multivuelta

En el caso de un eje rotativo, el valor real retorna al comienzo del eje después de cada vuelta. Por lo tanto, los ejes rotativos pueden desplazarse de forma continua más allá del final o del principio del eje. La posición del eje se indica siempre como valor entre el principio y el final del eje rotativo.



Funciones con Easy Motion Control

Resumen

Easy Motion Control sirve para posicionar un eje en un punto de destino con la aceleración, velocidad y retardo que elija el usuario o para desplazarlo a velocidad constante. Los movimientos de desplazamiento necesarios para ello se inician mediante la llamada de instrucciones de desplazamiento.

Al lanzar una petición de desplazamiento, en la instrucción de desplazamiento se parametriza el punto de destino o el trayecto del movimiento de desplazamiento, y, dado el caso, también la aceleración, velocidad o retardo que el usuario desee. El sistema de medida (unidad de longitud) de estos valores puede elegirse a discreción (p. ej. mm).

JOG

Con la función "Jog" se mueve el accionamiento en sentido positivo o negativo. No se preselecciona ningún destino.

La función Jog se ejecuta con la instrucción MC_MoveJog (Página 7330).

Sincronizar

Para asignar un valor de encóder reproducible a la posición real debe establecerse una referencia (sincronización) entre la posición del eje y el valor del encóder.

La función Sincronizar se ejecuta con la instrucción MC_Home (Página 7333).

Sincronizar con encóder incremental

Al conectar la instalación, un encóder incremental no proporciona información sobre la posición actual del eje. En lugar de ello, empieza a contar desde la posición actual del eje.

Es necesario realizar una sincronización cada vez que se conecte la instalación, después de un fallo y después de sustituir el encóder.

La sincronización de un encóder incremental puede llevarse a cabo

- mediante búsqueda del punto de referencia
- mediante posicionamiento del punto de referencia

Sincronizar con encóder absoluto

Al poner en servicio un encóder absoluto, el valor indicado por el encóder debe sincronizarse con la posición real del eje.

No es necesario repetir la sincronización tras una nueva conexión de la instalación ni después de un fallo. Sin embargo, después de sustituir el encóder sí es necesario realizar de nuevo la sincronización.

La sincronización de un encóder absoluto solo puede llevarse a cabo mediante el posicionamiento del punto de referencia (también denominado reajuste del encóder absoluto).

Búsqueda del punto de referencia

El eje se mueve en dirección al punto de referencia. Al rebasar el punto de referencia, el eje queda sincronizado, es decir, se asigna un valor de posición al valor numérico del encóder incremental. En cada búsqueda del punto de referencia se puede asignar un nuevo valor de coordenadas al punto de referencia.

La búsqueda del punto de referencia tan solo es posible con encóders incrementales.

Posicionamiento del punto de referencia

En el posicionamiento del punto de referencia se asigna un nuevo valor de coordenadas a la posición actual.

El posicionamiento del punto de referencia es posible con encóders incrementales y absolutos.

Posicionamiento absoluto o relativo

Easy Motion Control puede

- mover el eje a puntos de destino **absolutos**,
- desplazar el eje **de forma relativa** por un tramo en un sentido predeterminado.

Las funciones Posicionamiento absoluto y Posicionamiento relativo se ejecutan con las instrucciones MC_MoveAbsolute (Página 7325) y MC_MoveRelative (Página 7328).

Reductor

Easy Motion Control implementa un reductor electrónico entre dos ejes con la instrucción MC_GearIn (Página 7342).

Detención de un desplazamiento

Un desplazamiento en curso puede ser detenido en cualquier momento.

La función Detener se ejecuta con la instrucción MC_StopMotion (Página 7337).

Anulación de un desplazamiento

Una instrucción de desplazamiento activa

- puede anularse en cualquier momento iniciando otra instrucción de desplazamiento.
- puede cancelarse en cualquier momento mediante la instrucción MC_StopMotion (ver Desarrollo de movimientos de desplazamiento (Página 11258)).

Seguimiento

La función "Seguimiento" consiste en corregir de manera continua la consigna de posición conforme a la posición real en el regulador de posición. Durante el seguimiento no se vigila si el eje abandona la zona de velocidad cero.

La función se activa retirando la señal "EnableDrive" en la instrucción MC_Control (Página 7339) o después de producirse un fallo.

Corrección de velocidad

La corrección de velocidad permite modificar de manera dinámica la velocidad parametrizada de una instrucción de desplazamiento. Los valores de aceleración y deceleración no se ven afectados.

Para más información sobre la corrección de la velocidad, ver Definir la corrección de la velocidad (Página 11244).

Simulación

El modo de simulación permite verificar el programa de desplazamiento sin necesidad de mover el eje.

La simulación se ejecuta con la instrucción MC_Simulation (Página 7341).

Reacciones a errores y diagnóstico

Todos los errores detectados por las instrucciones se indican en el DB de eje por medio de

- activación de un error agrupado y
- una señalización de error específica.

En los errores de acuse obligatorio, después de subsanar el error es necesario activar un acuse de error.

Si el error ha sido causado o detectado por una instrucción de desplazamiento (p. ej. MC_MoveJog o MC_MoveAbsolute), en esa instrucción se activa la salida de error "Error".

Sin embargo, si el error ha sido detectado por una instrucción de driver, la causa exacta del error se indicará en los datos estáticos de la instrucción (ver Driver de entrada (Página 7348)).

Encontrará información detallada sobre reacciones a errores, señalización de errores y acuse de errores en Señalización y acuse de errores en el DB de eje (Página 11273).

Configuración de Easy Motion Control

Guía para utilizar Easy Motion Control

La guía aquí descrita muestra el procedimiento básico para utilizar Easy Motion Control con una CPU S7-300/400.

Requisitos

Para utilizar el objeto tecnológico "Eje" es necesario haber creado un proyecto con una CPU.

Procedimiento

Proceda según el orden recomendado a continuación para utilizar Easy Motion Control con la CPU S7-300/400. Para ello consulte los enlaces listados:

1. Agregar el objeto tecnológico Eje (Página 11227)
2. Trabajando con el diálogo de configuración (Página 11228)
3. Cargar en la CPU (Página 11250)
4. Test de funcionamiento del eje en la ventana de puesta en marcha
5. Programaación de Easy Motion Control (Página 11252)
6. Diagnosticar el control del eje

Agregar el objeto tecnológico Eje

Agregar objeto tecnológico Eje en el árbol del proyecto

Al agregar un objeto tecnológico, se genera un DB de instancia para este objeto tecnológico. En él se guarda la configuración del eje.

Requisitos

Hay un proyecto creado con una CPU.

Procedimiento

Para agregar un objeto tecnológico Eje en el árbol del proyecto, proceda del siguiente modo:

1. Abra la carpeta de la CPU en el árbol del proyecto.
2. Abra la carpeta "Objetos tecnológicos".
3. Haga doble clic en "Agregar objeto".
   Se abre el cuadro de diálogo "Agregar objeto".
4. Seleccione la tecnología "Motion".
5. Abra la carpeta "Motion Control".
6. Abra la carpeta "Easy Motion Control".
7. Seleccione la instrucción para el objeto tecnológico: en este caso AXIS_REF.
8. En el campo de entrada "Nombre", introduzca un nombre personalizado para el eje.
9. Seleccione la opción "Manual" si desea modificar el número de bloque de datos propuesto para el DB de instancia.
10. Haga clic en "Más información" para añadir información propia sobre el objeto tecnológico.
11. Haga clic en el botón "Aceptar" para agregar el objeto tecnológico.
    Haga clic en el botón "Cancelar" para rechazar las entradas realizadas.

Resultado

El nuevo objeto tecnológico se genera y se guarda en la capeta "Objetos tecnológicos" del árbol del proyecto. El objeto tecnológico se utiliza cuando las instrucciones de este objeto tecnológico se llaman en un OB de alarma cíclica o de alarma de sincronismo.

Encontrará más información al respecto en "Diseño de un programa de usuario (Página 11252)".

Nota

Existe la posibilidad de activar la casilla de verificación "Agregar nuevo y abrir" situada en la parte inferior del diálogo. De ese modo, la configuración del objeto tecnológico se abrirá una vez agregado.

Configuración del objeto tecnológico Eje

Trabajando con el diálogo de configuración

Las propiedades del objeto tecnológico se configuran en la ventana de configuración. Para abrir la ventana de configuración del objeto tecnológico, proceda del siguiente modo:

1. Abra el grupo correspondiente al objeto tecnológico deseado en el árbol del proyecto.
2. Haga doble clic en el objeto "Configuración".

La configuración se divide en las siguientes categorías:

- **Parámetros básicos**
  Los parámetros básicos contienen todos los parámetros que deben configurarse para un eje listo para funcionar.
- **Parámetros avanzados**
  Los parámetros avanzados contienen parámetros que se pueden adaptar al propio accionamiento o bien a la instalación.

Símbolos de la ventana de configuración

Los símbolos que aparecen en la navegación local de la configuración muestran más detalles sobre el estado de la configuración:

	La configuración contiene valores predeterminados y está completa. La configuración contiene exclusivamente valores predeterminados. Con ellos es posible utilizar el objeto tecnológico sin más modificaciones.
	La configuración contiene valores definidos por el usuario y está completa Todos los campos de entrada de la configuración contienen valores válidos y al menos un valor predeterminado ha sido modificado.
	La configuración es errónea o está incompleta Al menos un campo de entrada o una lista desplegable contiene un valor no válido. El campo o la lista desplegable en cuestión se marca en rojo. Al hacer clic, el roll out con el aviso de error indica la causa del error.
	La configuración es válida, pero contiene advertencias Sólo se ha configurado un final de carrera por hardware. Dependiendo de la instalación, si un final de carrera por hardware no está configurado ello puede representar un peligro. El campo o la lista desplegable en cuestión se marca en amarillo.

Parámetros básicos

Definir el nombre del eje

Nombre del eje

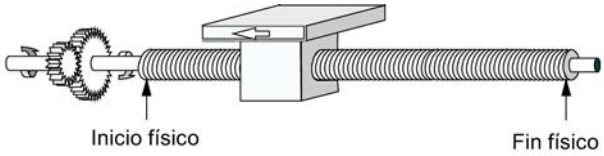
Defina en este campo el nombre del eje y del objeto tecnológico "Eje". El objeto tecnológico aparece con ese nombre en el árbol del proyecto.

Seleccionar tipo de eje

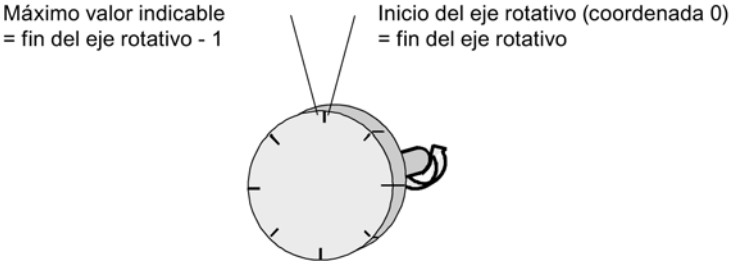
Tipos de eje

Easy Motion Control puede controlar los siguientes tipos de eje:

- Eje lineal
  El eje lineal es un eje con una zona de desplazamiento físico limitado.



- Eje rotativo
  El eje rotativo es un eje cuya zona de desplazamiento no está limitada por topes mecánicos.



Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
4.0	AxisType	BOOL	FALSE	**Tipo de eje** Valores permitidos: • FALSE = eje lineal • TRUE = eje rotativo	I

Seleccionar la unidad de longitud

Unidad de longitud

En la lista desplegable, seleccione la unidad de longitud deseada para el sistema de medida del eje. La unidad seleccionada se utiliza para una nueva configuración del objeto tecnológico "Eje" y la visualización de los datos actuales del eje.

Para la entrada y salida de datos es posible elegir entre las siguientes unidades de longitud:

- Milímetros (ajuste predeterminado)
- Grados
- Pulgadas

Los valores en los parámetros de entrada (Position, Distance, Velocity, ...) de las instrucciones de Easy Motion Control también se refieren a esta unidad.

Nota

El cambio de la unidad de longitud no influye en los valores de los parámetros. No tiene lugar una conversión.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
196.0	LengthUnit	STRING [4]	mm	**Unidad de longitud** para software de configuración Valores permitidos: • mm (ajuste predeterminado) • Grados • Pulgadas

Activar el modo de simulación

Modo de simulación

Active la casilla de verificación si desea probar el programa de desplazamiento sin necesidad de mover el eje.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
4.2	Sim	BOOL	FALSE	**Modo de simulación** Valores permitidos: • FALSE = desactivado • TRUE = activado

Parámetros avanzados

Interfaces de driver

Driver de entrada

Seleccionar el módulo de entradas

Módulo de entradas

Seleccione en la lista desplegable el módulo de entradas al que está conectado el encóder.

Conforme a su selección, en el campo "Instrucción" se muestra el nombre de la instrucción que debe llamar para la adquisición del valor real.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
202.0	InputModuleType	STRING [24]	IM 174	Driver de entrada **del módulo**. Este parámetro no se puede sobrescribir manualmente.

Seleccionar el tipo de encóder

Tipo de encóder

Active el botón de opción correspondiente al encóder utilizado.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
4.1	EncoderType	BOOL	FALSE	**Tipo de encóder** Valores permitidos: • FALSE = encóder incremental • TRUE = encóder absoluto	I

Módulo de entradas: Dirección del módulo Entradas

Dirección del módulo Entradas

Introduzca en el campo de entrada el valor asignado al módulo de entradas en la configuración hardware.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
58.0	InputModuleInAddr	INT	0	**Dirección inicial de las entradas del módulo de lectura de recorrido** Rango: • determinado por la configuración hardware	I

Módulo de entradas: Dirección del módulo Salidas

Dirección del módulo Salidas

Este parámetro tan solo se utiliza en aquellos módulos en que la "Dirección del módulo Salidas" se ajusta por separado de la "Dirección del módulo Entradas" en la configuración hardware.

Introduzca en el campo de entrada el valor asignado al módulo de entradas en la configuración hardware.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
60.0	InputModuleOutAddr	INT	0	**Dirección inicial de las salidas del módulo de lectura de recorrido** Rango: • determinado por la configuración hardware	I

Módulo de entradas: Número de canal

Número de canal

- Módulos monocanal: el campo de entrada está predeterminado con el valor "0" y no se puede editar.
- Módulos multicanal: introduzca en el campo de entrada el número del canal utilizado. Introduzca "0" para el primer canal, etc.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
62.0	InputChannelNo	INT	0	**Número de canal del módulo de lectura de recorrido** Rango: • depende del módulo	I

Driver de salida

Seleccionar el módulo de salidas

Módulo de salidas

Seleccione en la lista desplegable el módulo de salidas al que está conectado el accionamiento.

Conforme a su selección, en el campo "Instrucción" se muestra el nombre de la instrucción que debe llamar para la salida de la consigna de velocidad.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
228.0	OutputModuleType	STRING [24]	IM 174	Driver de salida **del módulo**. Este parámetro no se puede sobrescribir manualmente.

Módulo de salidas: Dirección del módulo Entradas

Dirección del módulo Entradas

El parámetro tan solo se utiliza en aquellos módulos en que la "Dirección del módulo Entradas" se ajusta por separado de la "Dirección del módulo Salidas" en la configuración hardware.

Introduzca en el campo de entrada el valor asignado al módulo de salidas en la configuración hardware.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
82.0	OutputModuleInAddr	INT	0	**Dirección inicial de las entradas del módulo de salidas** Rango: • determinado por la configuración hardware	I

Módulo de salidas: Dirección del módulo Salidas

Dirección del módulo Salidas

Introduzca en el campo de entrada el valor asignado al módulo de salidas en la configuración hardware.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
80.0	OutputModuleOutAddr	INT	0	**Dirección inicial de las salidas del módulo de salidas** Rango: • determinado por la configuración hardware	I

Módulo de salidas: Número de canal

Número de canal

- Módulos monocanal: el campo de entrada está predeterminado con el valor "0" y no se puede editar.
- Módulos multicanal: introduzca en el campo de entrada el número del canal utilizado. Introduzca "0" para el primer canal, etc.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
84.0	OutputChannelNo	INT	0	**Número de canal del módulo de salidas** Rango: • depende del módulo	I

Mecánica

Encóder

Definir incrementos por vuelta del encóder

Incrementos por vuelta del encóder

Introduzca en el campo de entrada el número de incrementos que emite el módulo de lectura de recorrido por cada vuelta del encóder.

Con este valor y el parámetro "Recorrido del eje por vuelta del encóder" (DB de eje Dirección 68.0 (Página 11236)) tiene lugar la conversión de incrementos a unidad de longitud.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
64.0	StepsPerRev	DINT	L#1	**Incrementos por vuelta del encóder** Rango admisible: • ≥ 1 [incrementos/vuelta del encóder]	I

Encóder incremental

Si el encóder incremental se lee con un módulo que realiza una evaluación doble o cuádruple de los impulsos, esto debe tenerse en cuenta al introducir los parámetros.

Ejemplo:

- Impulsos de encóder por vuelta: 500
- Evaluación cuádruple

En el campo de entrada "Incrementos por vuelta del encóder" se introduce: 2000

Encóder absoluto

Con encóders absolutos, el número de incrementos por vuelta del encóder generalmente es una potencia de dos.

Son valores habituales:

- En caso de encóder monovuelta: 4096 y 8192
- En caso de encóder multivuelta (12 × 12 bits en el telegrama de 25 bits): 4096

En caso de duda, el valor correcto se determina moviendo el encóder una vuelta y observando la variación del valor de encóder en el software de configuración.

Si hay una regla de medida, en el campo "Incrementos por vuelta del encóder" introduzca el número total de incrementos que corresponda a la longitud de la regla de medida.

Definir el número de vueltas del encóder

Número de vueltas del encóder

Si se trata de un encóder absoluto, introduzca en el campo de entrada el número de vueltas del encóder.

Para un encóder monovuelta, introduzca un "1".

Si hay una **regla de medida**, en el campo "Número de vueltas del encóder" introduzca un "1".

Si se trata de un encóder incremental, este parámetro carece de significado.

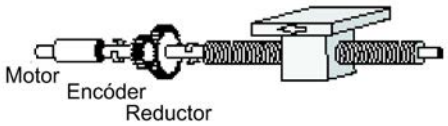
Datos utilizados en el DB de eje

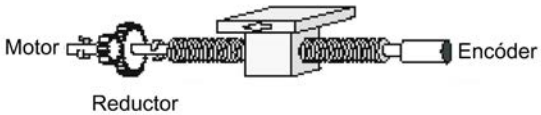
Dirección	Nombre	Tipo	Valor inicial	Comentario	
72.0	NumberRevs	INT	1	**Número de vueltas** de un encóder absoluto Rango admisible: • > 0	I

Definir el recorrido del eje por vuelta del encóder

Recorrido del eje por vuelta del encóder

Introduzca en el campo de entrada la distancia que recorre el eje en una vuelta del encóder. El valor depende del diseño del eje y del modo en que esté montado el encóder. Debe tener en cuenta todos los elementos de transmisión, como acoplamientos o engranajes.





Si hay una **regla de medida**, en el campo "Recorrido del eje por vuelta del encóder" debe introducirse la longitud de dicha regla.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
68.0	DisplacementPerRev	REAL	1.0	**Recorrido del eje por vuelta del encóder** Rango admisible: • > 0.0 [unidad de longitud/vuelta del encóder]	I

Particularidades de los ejes rotativos

Con cada vuelta del eje rotativo el encóder debe desplazarse un número entero de incrementos:

$$\frac{\text{(Fin del eje rotativo - inicio del eje rotativo)}}{\text{Recorrido del eje por vuelta del encóder}} \times \text{Incrementos por vuelta del encóder} = \text{número entero}$$

Particularidades de ejes rotativos con encóders absolutos

A fin de garantizar la reproducibilidad de cada posición, el recorrido del eje abarcado por el encóder debe ser un múltiplo entero de la distancia recorrida por el eje rotativo en una vuelta:

$$\frac{\text{(número de vueltas del encóder) x (recorrido del eje por vuelta del encóder)}}{\text{(Fin del eje rotativo - inicio del eje rotativo)}} = \text{número entero}$$

Ejemplo:

Encóder absoluto 24 bits

=> Número de vueltas del encóder = 4096

Principio del eje rotativo = 0.0°

Fin del eje rotativo = 360.0°

=> Fin del eje rotativo - Principio del eje rotativo = 360.0°

Recorrido del eje por vuelta del encóder = 45°

$$\frac{4096 \times 45°}{360°} = 512 \text{ (entero)}$$

Seleccionar la corrección de sentido del encóder

Corrección de sentido del encóder

Aquí se adapta el sentido de la lectura del recorrido al sentido de movimiento del eje:

- positivo = los impulsos de contaje ascendentes (encóder incremental) o valores del encóder (encóder absoluto) equivalen a los valores de posición real ascendentes.
- negativo = los impulsos de contaje ascendentes (encóder incremental) o los valores de encóder (encóder absoluto) equivalen a valores reales de posición descendentes.

Seleccione la opción correspondiente en la lista desplegable.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
74.0	PolarityEncoder	INT	1	**Corrección de sentido del encóder** Valores permitidos: • +1 ó -1	I

Regulador

Definir la ganancia del regulador

Ganancia del regulador

La ganancia óptima del regulador se puede determinar experimentalmente en el eje.

Aumente la ganancia del regulador en incrementos de 1.0 hasta que el eje comience a vibrar en marcha o en parada. Cuando esto ocurra, reduzca la ganancia del regulador hasta que deje de observarse una tendencia a la vibración.

Introduzca el valor correspondiente en el campo de entrada.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
44.0	FactorP	REAL	1.0	**Ganancia del regulador** Rango admisible: • > 0.0 [1/s]

Definir la velocidad de consigna en modo manual

Velocidad de consigna en el modo manual

Introduzca en el campo de entrada la velocidad consignada que desee para el modo manual.

Si el modo manual está activado, el regulador emite el valor de "velocidad consignada en modo manual" como magnitud manipulada y lo limita a "MaxVelocity". La consigna de posición se corrige conforme al valor de posición real.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
48.0	ManVelocity	REAL	0.0	**Velocidad de consigna en el modo manual** Rango admisible: • ≤ ± MaxVelocity (Página 11243) [unidad de longitud/s]
52.0	ManEnable	BOOL	FALSE	**Activar el modo manual** Valores permitidos: • FALSE = desactivado • TRUE = activado

Motor

Definir el valor de referencia para una velocidad del 100%

Valor de referencia para una velocidad del 100%

Si se utilizan etapas de potencia controladas con ±10 V, este parámetro es irrelevante.

Si se utiliza el driver de salida OutputMM4_DP, en este parámetro debe introducirse el valor con el que deberá controlarse el accionamiento para que alcance el 100% de su velocidad.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
96.0	DriveInputAt100	REAL	10.0	**Valor de referencia para una velocidad del 100%** Rango admisible: • -10.0 hasta +10.0 [V]	I

Ejemplo

- Velocidad nominal del motor (velocidad 100%): = 1380 r/min
- El motor está calibrado a: = 1380 r/min a 50 Hz

=> En el campo "Valor de referencia para una velocidad del 100%", introduzca: = 50 Hz

Definir el valor de referencia para la velocidad máxima del eje

Valor de referencia para velocidad máxima del eje

Si se utilizan etapas de potencia controladas con **±10 V**, en el campo de entrada deberá introducirse el valor de tensión en el que se alcanza la "velocidad máxima" del accionamiento (ver DB de eje Dirección 14.0 (Página 11243)).

Ejemplo:

- Velocidad nominal del motor: = 3000 r/min
- El motor está calibrado a: = 9 V a 3000 r/min
- Quiere alcanzar la velocidad máxima: = 1500 r/min

=> En el campo "Valor de referencia para la velocidad máxima del eje", introduzca: = 4.5 V

Si se utiliza el driver de salida **OutputMM4_DP**, en este campo de entrada se debe introducir el valor con el que debe controlarse el accionamiento para que alcance la "velocidad máxima" (ver DB de eje Dirección 14.0 (Página 11243)).

Ejemplo:

- Velocidad nominal del motor: = 1380 r/min
- El motor está calibrado a: = 1380 r/min a 50 Hz
- Quiere alcanzar la velocidad máxima: = 690 r/min

=> En el campo "Valor de referencia para la velocidad máxima del eje", introduzca: = 25 Hz

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
88.0	DriveInputAtMaxVel	REAL	9.0	**Valor de referencia para velocidad máxima del eje [V]** Rango admisible: • En función de la etapa de potencia	I

Definir la compensación del punto cero

Compensación del punto cero

Este parámetro permite realizar una compensación del punto cero si el accionamiento no ofrece tal posibilidad, El valor parametrizado aquí se suma al valor de salida.

El parámetro no se tiene en cuenta en el driver de salida OutputMM4_DP.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
92.0	OffsetCompensation	REAL	0.0	**Compensación del punto cero** Rango admisible: • -10.0 hasta +10.0 [V]

Seleccionar la corrección de sentido del accionamiento

Corrección de sentido del accionamiento

Aquí se selecciona el sentido de movimiento del accionamiento:

- positiva = las consignas de velocidad positivas provocan un movimiento en sentido positivo.
- negativa = las consignas de velocidad positivas provocan un movimiento en sentido negativo.

Seleccione la opción correspondiente en la lista desplegable.

Si se modifica la "Corrección de sentido del accionamiento", también es necesario modificar la "Corrección de sentido del encóder" (DB de eje Dirección 74.0 (Página 11234)).

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
86.0	PolarityDrive	INT	1	**Incrementos por vuelta del encóder** Valores permitidos: • +1 ó -1	I

Final de carrera

Activar finales de carrera por software

Fin de carrera por software

Los finales de carrera por software existen solo para los ejes lineales.

La vigilancia comienza en cuanto el eje se sincroniza y la vigilancia de los finales de carrera por software está activada.

Si la vigilancia de los finales de carrera por software está desactivada, los parámetros "Inicio Final de carrera por software" y "Fin Final de carrera por software" (DB de eje Dirección 6.0 y 10.0 (Página 11242)) carecen de significado.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
42.1	SWLimitEnable	BOOL	TRUE	**Vigilar finales de carrera por software** Valores permitidos: • TRUE = vigilancia activada • FALSE = vigilancia desactivada

Definir final de carrera por software

Fin de carrera por software

Los finales de carrera por software existen solo para los ejes lineales.

Con eje lineal: inicio/fin del final de carrera por software

La vigilancia comienza en cuanto el eje se sincroniza y la vigilancia de los finales de carrera por software está activada (ver "Activar finales de carrera por software (Página 11237)"). La zona de trabajo está limitada por los finales de carrera por software.

El "Inicio del final de carrera por software" debe ser siempre menor que el "Fin del final de carrera por software".

Encóder incremental

Después de cada arranque de la CPU el eje no está sincronizado. El eje no se sincroniza y la vigilancia de los finales de carrera por software no comienza hasta después de la búsqueda del punto de referencia o el posicionamiento del punto de referencia.

Encóder absoluto

Una vez se ha realizado el posicionamiento del punto de referencia, éste no se pierde con el siguiente arranque de la CPU. Los finales de carrera por software se pueden vigilar.
El rango del encóder absoluto debe abarcar al menos la zona de trabajo.

Con eje rotativo: inicio del eje rotativo/final del eje rotativo

Con los valores "Inicio del eje rotativo" y "Final del eje rotativo" se determina el rango de valores en que se indicarán las posiciones de eje durante una vuelta del eje.

El valor "Final del eje rotativo" es el valor máximo teórico que el valor real puede alcanzar:

$$\text{Máximo valor indicable} = \text{fin del eje rotativo} - \frac{\text{Recorrido por vuelta del encóder}}{\text{Incrementos por vuelta}}$$

Sin embargo, el máximo valor teórico nunca se indica porque físicamente identifica la misma posición que el inicio del eje rotativo.

Ejemplo:

- AxisLimitMin = 0.0°
- AxisLimitMax = 360.0°

En sentido positivo, el valor real adopta los valores ...359.8, 359.9, 0.0, 0.1, …

Si se utilizan encóders absolutos, debe tenerse en cuenta el parámetro "Recorrido del eje por vuelta del encóder (Página 11232)".

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
6.0	AxisLimitMax	REAL	$1.0e^{+6}$	Con eje lineal: **Final de carrera por software Fin** Con eje rotativo: **Final del eje rotativo** Rango admisible: • $-2^{24}+6 \leq$ AxisLimitMin < AxisLimitMax $\leq +2^{24}-6$ [unidad de longitud]	I
10.0	AxisLimitMin	REAL	$-1.0e^{+6}$	Con eje lineal: **Inicio del final de carrera por software** Con eje rotativo: **Inicio del eje rotativo** Rango admisible: • $-2^{24}+6 \leq$ AxisLimitMin < AxisLimitMax $\leq +2^{24}-6$ [unidad de longitud]	I

Dinámica

General

Definir la velocidad máxima del eje

Velocidad máxima del eje

Aquí se parametriza la velocidad que alcanza el eje cuando el motor se controla con el valor introducido en el parámetro "Valor de referencia para la velocidad máxima del eje" (DB de eje Dirección 88.0 (Página 11236)).

La "Velocidad máxima del eje" sirve como factor para el cálculo de la señal de salida al accionamiento y también para limitar la velocidad del eje.

Particularidades de ejes rotativos con encóders absolutos

Para poder registrar cada una de las vueltas del eje rotativo, la velocidad máxima del eje debe ser la mínima que permita que por cada tiempo de muestreo se recorra una distancia inferior a la mitad del rango del encóder absoluto.

Ejemplo de encóder monovuelta:

Requisito => una vuelta del eje equivale a una vuelta del encóder.

Recorrido/rango del encóder = 360.0°

Recorrido/mitad del rango del encóder = 180.0°

Tiempo de muestreo = 0.01 s

$$\Rightarrow \text{„Velocidad máxima“} \quad - \quad \frac{\text{Recorrido de desplazamiento/mitad del rango del encóder}}{\text{Tiempo de muestreo}}$$

=> "velocidad máxima" < 180.0° / 0.01 s = 18000/s corresponde a 50 vueltas de eje/s

Datos utilizados en el DB de eje

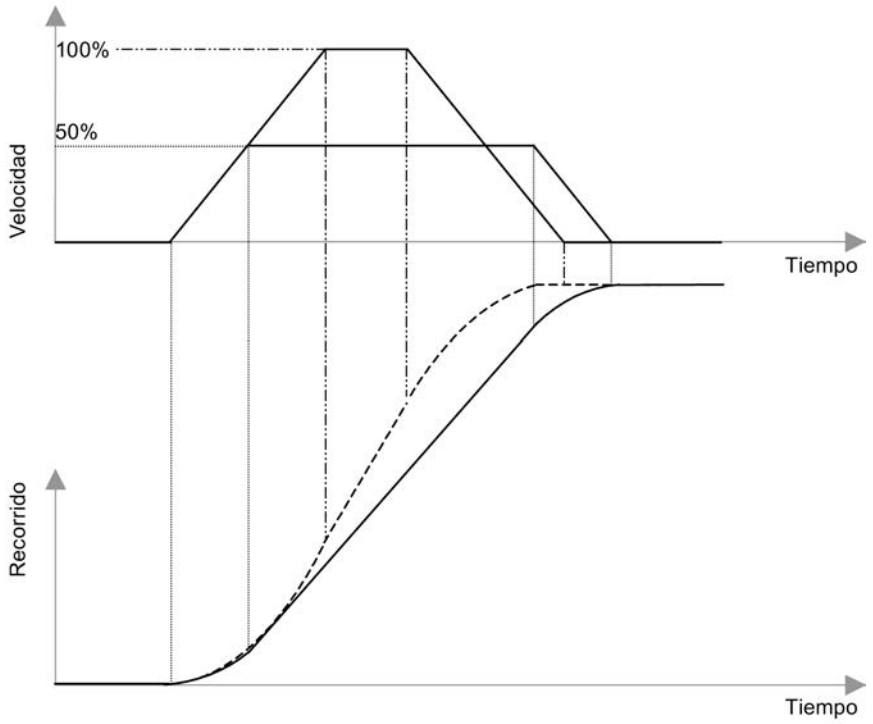
Dirección	Nombre	Tipo	Valor inicial	Comentario	
14.0	MaxVelocity	REAL	0.1	**Velocidad máxima del eje** Rango admisible: • > 0.0 [unidad de longitud/s]	I

Definir la corrección de la velocidad

Corrección de velocidad

La "Corrección de la velocidad" actúa de forma continua sobre la velocidad parametrizada de una instrucción de desplazamiento.
Los valores de aceleración y deceleración no se ven afectados.

Ejemplo: curvas de desplazamiento con corrección del 50% y del 100%



Al cambiar del 50% al 100%, la velocidad se dobla. Sin embargo, como la aceleración y la deceleración no se ven afectadas por el parámetro "Corrección de velocidad", el tiempo de posicionamiento **no** se reduce a la mitad.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
100.0	Override	REAL	100.0	**Corrección de velocidad** Rango admisible: • 0.0 hasta +100.0 [%]

Definir la aceleración máxima del eje

Aceleración máxima del eje

En modo regulado, este valor determina el límite de la pendiente de la rampa de aceleración del eje.

La aceleración máxima posible se puede determinar experimentalmente.

Debe ser la mínima que permita que al acelerar hasta la velocidad máxima (con carga) no se alcance por poco el límite de intensidad del accionamiento.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
18.0	MaxAcceleration	REAL	0.1	**Aceleración máxima del eje** Rango admisible: • > 0.0 [unidad de longitud/s^2]	I

Definir la deceleración máxima del eje

Deceleración máxima del eje

En modo regulado, este valor determina el límite de la pendiente de la rampa de frenado del eje.

La deceleración máxima posible se puede determinar experimentalmente.

Debe ser la mínima que permita que al frenar desde la velocidad máxima (con carga) no se alcance por poco el límite de intensidad del accionamiento.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
22.0	MaxDeceleration	REAL	0.1	**Deceleración máxima del eje** Rango admisible: • > 0.0 [unidad de longitud/s^2]	I

Definir el tiempo de muestreo

Tiempo de muestreo

Para el correcto funcionamiento de una regulación de posición es importante que todas las funciones de un eje se procesen en una base de tiempo fija. Esta base de tiempo se obtiene llamando todas las instrucciones de un eje en el mismo OB de alarma cíclica o de alarma de sincronismo. La base de tiempo de este OB se selecciona en la configuración hardware de la CPU.

Como "Tiempo de muestreo" se introduce la base de tiempo del OB en el que se llaman todas las instrucciones de un eje.

Ejemplo:

Se llama el programa con las instrucciones de Easy Motion Control en el OB 35. En la configuración hardware, la base de tiempo de la alarma cíclica del OB 35 está ajustada a 10 ms.

Como "Tiempo de muestreo" se introduce 0.010.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
0.0	Sample_T	REAL	0.01	**Tiempo de muestreo** de las instrucciones de eje Sugerencia: • En el S7-300 hasta CPU 316 ≥ 0.020 [s] • En el S7-400 y desde CPU 317 ≥ 0.004 [s]	I

Mostrar el tiempo de aceleración

Tiempo aceleración

El tiempo de aceleración es el tiempo durante el cual el eje acelera desde la velocidad cero hasta la velocidad máxima.

El tiempo de aceleración se calcula a partir de los siguientes parámetros:

Tiempo aceleración = velocidad máxima del eje × corrección de velocidad / aceleración máxima

En el campo de salida se muestra el tiempo de aceleración calculado.

Mostrar el tiempo de deceleración

Tiempo de deceleración

El tiempo de deceleración es el tiempo durante el cual el eje decelera sin limitación de tirones desde la velocidad máxima hasta la velocidad cero.

El tiempo de deceleración se calcula a partir de los siguientes parámetros:

Tiempo de deceleración = velocidad máxima del eje × corrección de velocidad / deceleración máxima

En el campo de salida se muestra el tiempo de deceleración calculado.

Parada de emergencia

Definir la deceleración para la parada de emergencia

Deceleración para parada de emergencia (parada brusca)

Aquí se parametriza la pendiente de la rampa de frenado controlada por tiempo con la que debe frenarse el accionamiento en caso de producirse un error con parada brusca.

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
54.0	EmergencyDec	REAL	1.0	**Deceleración para parada de emergencia** Rango admisible: • > 0.0 [unidad de longitud/s^2]	I

Mostrar el tiempo de deceleración

Tiempo de deceleración

El tiempo de deceleración es el tiempo durante el cual el eje decelera en modo de parada de emergencia desde la velocidad máxima hasta la velocidad cero.

El tiempo de deceleración se calcula a partir de los siguientes parámetros:

Tiempo de deceleración = velocidad máxima del eje/deceleración de parada de emergencia

En el campo de salida se muestra el tiempo de deceleración calculado.

Vigilancias

Definir la zona de destino

Zona de destino

La "zona de destino" se encuentra a izquierda y derecha del punto de destino (ver Definiciones de términos (Página 11216)).

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
30.0	TargetRange	REAL	1.0	**Zona de destino** Rango admisible: • > 0.0 [unidad de longitud]	I

Definir la zona de velocidad cero

Zona de velocidad cero

A la velocidad cero se vigila si el eje permanece fijo en una posición o se va alejando de ella. En cuanto el eje abandona la "Zona de velocidad cero" sin una petición de desplazamiento activa, en el DB de eje se activa el bit de error "Zona de velocidad cero abandonada" (DB de eje Dirección 130.3 (Página 11278)). La "Zona de velocidad cero" se encuentra a izquierda y derecha del punto de destino (ver Definiciones de términos (Página 11216)).

Recomendación: La "Zona de velocidad cero" debe ser mayor que la "zona de destino" (DB de eje Dirección 30.0 (Página 11244)).

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
34.0	StandstillRange	REAL	1.5	**Zona de velocidad cero** Rango admisible: • > 0.0 [unidad de longitud]	I

Vigilar la aproximación al destino

Vigilancia de la aproximación al destino

SI la vigilancia de la aproximación al destino está activada, un desplazamiento concluye en cuanto la posición real ha alcanzado la "zona de destino" (DB de eje Dirección 30.0) dentro del "Tiempo de vigilancia para aproximación a destino" (DB de eje Dirección 26.0).

Si la vigilancia de la aproximación a destino está desactivada, un desplazamiento concluye en cuanto la consigna de posición ha alcanzado el destino (ver Definiciones de términos (Página 11216)).

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario
42.0	MonitorTargetAppr	BOOL	TRUE	**Vigilar la aproximación al destino** Valores permitidos: • TRUE = vigilancia activada • FALSE = vigilancia desactivada

Definir el tiempo de vigilancia para la aproximación al destino

Tiempo de vigilancia para la aproximación al destino

Después de que en un desplazamiento la **consigna de posición** haya alcanzado el destino, comienza el "Tiempo de vigilancia para la aproximación a destino", si es que el bit "Vigilar aproximación a destino" (DB de eje Dirección 42.0 (Página 11245)) está activado. Si transcurrido este tiempo la **posición real** no ha alcanzado la zona definida en el parámetro "Zona de destino" (DB de eje Dirección 30.0 (Página 11244)), en el DB de eje se activa el bit de error "Error en la aproximación al destino" (DB de eje Dirección 130.4 (Página 11278)).

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
26.0	MonTimeTargetAppr	REAL	1.0	**Tiempo de vigilancia para la aproximación al destino** Rango admisible: • > 2 × Sample_T [s] (tiempo de muestreo (Página 11242))	I

Definir el error de seguimiento máximo permitido

Error de seguimiento máximo permitido

Con la vigilancia del error de seguimiento se comprueba si la posición real durante un desplazamiento se encuentra a una distancia excesivamente grande por detrás de la consigna de posición. Si se rebasa el error de seguimiento máximo, en el DB de eje se activa el bit de error "Error de seguimiento rebasado" (DB de eje Dirección 130.2 (Página 11278)).

Recomendación de ajuste: ("velocidad máxima" / "ganancia del regulador") × 1.1 (ver DB de eje Dirección 14.0 (Página 11239) y 44.0 (Página 11234))

Datos utilizados en el DB de eje

Dirección	Nombre	Tipo	Valor inicial	Comentario	
38.0	MaxFollowingDist	REAL	5.0	**Error de seguimiento máximo permitido** Rango admisible: • > 0.0 [unidad de longitud]	I

Cargar en la CPU

Los datos de proyecto nuevos o modificados se deben cargar en la CPU.

Carga de los bloques de programa y de la configuración del objeto tecnológico en la CPU

Para cargar proceda del siguiente modo:

1. En el árbol del proyecto seleccione el objeto "Bloques de programa" o el objeto "Objetos tecnológicos".
2. Seleccione el comando "Online > Cargar en dispositivo".

Nota

Al cargar bloques en la CPU, el TIA Portal se asegura siempre de que tras cada operación de carga los bloques offline en el proyecto y los bloques online en la CPU sean coherentes los unos con los otros.

En la CPU se cargan siempre todos los bloques offline modificados y nuevos y los objetos tecnológicos, independientemente de la marca en el árbol del proyecto.

En cada opreación de carga los valores actuales de los bloques de datos se resetean a los valores iniciales.

Carga de la configuración hardware en la CPU

Para cargar la configuración hardware proceda del siguiente modo:

1. Marque el objeto de la CPU en el árbol del proyecto.
2. Seleccione el comando "Cargar en dispositivo > Configuración hardware" en el menú contextual.

Carga de la configuración hardware, de los bloques de programa y de la configuración del objeto tecnológico en la CPU

Para cargar proceda del siguiente modo:

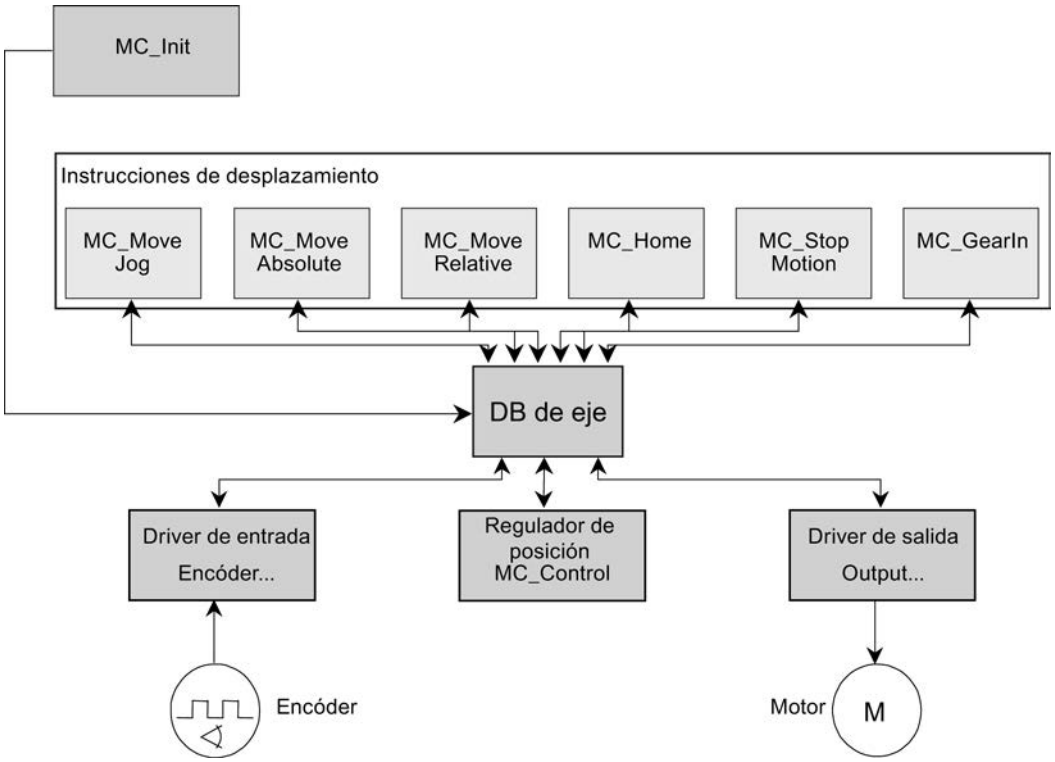
1. Marque el objeto de la CPU en el árbol del proyecto.
2. Seleccione el comando de menú "Online > Cargar en dispositivo".

Programaación de Easy Motion Control

Diseño de un programa de usuario

Interacción de las instrucciones

La figura siguiente muestra la interacción de las instrucciones para Easy Motion Control.



La instrucción MC_Init sirve para **inicializar** las instrucciones.

El **driver de entrada** (instrucción Encoder…) lee la posición real de un eje del módulo de periferia y la normaliza convirtiéndola a la unidad de longitud seleccionada por el usuario.

Una de las **instrucciones de desplazamiento** genera la consigna de posición.

El **regulador de posición** (posición MC_Control) compara el valor de posición real con la consigna y genera una consigna de velocidad a partir de la diferencia.

El **driver de salida** (instrucción Output…) normaliza la consigna de velocidad convirtiéndola al formato adecuado y la escribe en el módulo de periferia conectado.

El programa para un **lazo de regulación cerrado** se compone de un driver de entrada (Encoder…), al menos una instrucción de desplazamiento, el regulador (MC_Control) y un driver de salida (Output…).

Todas las instrucciones de un eje actúan junto con un **DB de eje**. El DB contiene todos los datos necesarios para la operación del eje, y hay uno para cada eje. A este respecto, tenga en cuenta las indicaciones relativas a la Carga del DB de eje (Página 11262).

Entorno de llamada

Para el correcto funcionamiento de una regulación de posición es importante que todas las funciones de un eje se procesen en una **base de tiempo** fija. Esta base de tiempo se determina llamando las instrucciones de Easy Motion Control en **un** nivel de ejecución. Para ello existen dos procedimientos, según la CPU utilizada y según el tipo de conexión de periferia:

OB de alarma cíclica (OB 30 a OB 38)

Se utiliza en caso de conexión centralizada de la periferia o en caso de conexión descentralizada a través de un bus de periferia no equidistante. El tiempo de alarma cíclica equivale al ciclo de regulación de posición.

OB de alarma de sincronismo (OB 61 a OB 64)

En caso de conexión de periferia mediante un bus de periferia siempre isócrono, el ciclo de regulación de posición equivale al valor que se haya seleccionado como "ciclo equidistante" al parametrizar el bus.

En ambos casos este valor debe introducirse en el DB de eje.

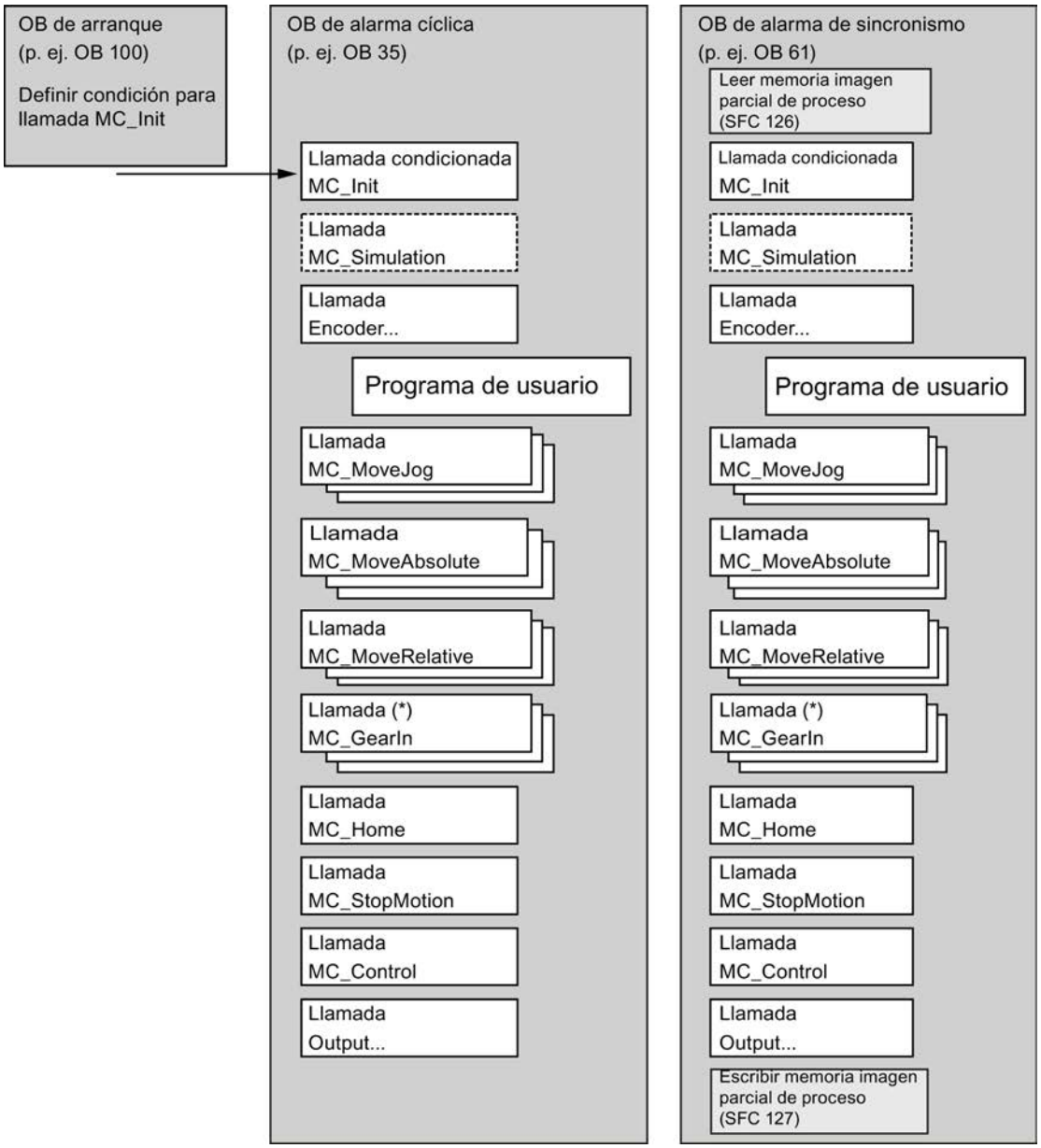
Hallará información general sobre el modo isócrono en la ayuda en pantalla del TIA portal y en el manual de funciones Isochrone Mode (http://support.automation.siemens.com/WW/view/en/15218045).

Desde el punto de vista de Easy Motion Control, deberían tenerse en cuenta las siguientes particularidades:

- Al configurar el bus de periferia y los esclavos DP/IO-Devices (y quizá también sus módulos), es preciso seleccionar y parametrizar la "equidistancia".
- Las direcciones de periferia de un eje utilizadas para la lectura de posición y para la salida de consigna deben estar en la misma memoria imagen parcial del (MIPP). En los ejes de reductor, ambos ejes deben estar en la misma MIPP.
- Al configurar la CPU, esta MIPP se asigna a la alarma de sincronismo utilizada.
- Esta MIPP debe actualizarse en el OB de alarma antes/después de llamar las instrucciones de Easy Motion Control con las instrucciones de sistema "SYNC_PI"/"SYNC_PO".
- Evalúe en su programa las respuestas de estas instrucciones de sistema y reaccione en consecuencia. (en caso de error, la MIPP podría no actualizarse, y entonces el valor real leído podría ser erróneo).

Orden de llamada recomendado

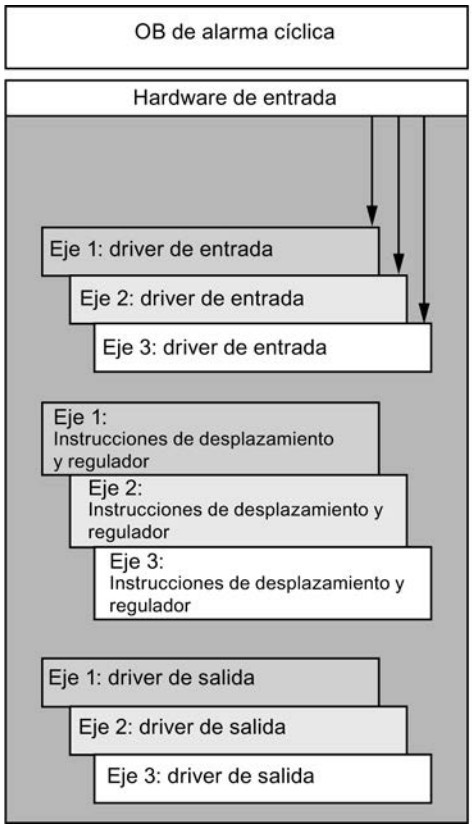
La figura siguiente muestra el orden de llamada recomendado para las instrucciones de Easy Motion Control.

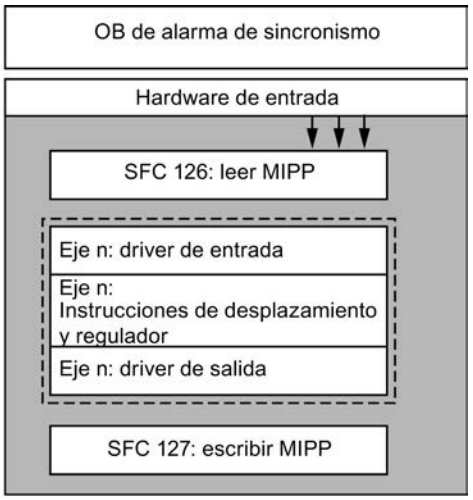


(*) La llamada de la instrucción de reductor MC_GearIn solo tiene sentido con ejes esclavos.

Varios ejes

Para el funcionamiento de varios ejes en una CPU se recomiendan las estructuras siguientes para el programa de usuario:





Recomendación para OB de alarma cíclica.	Recomendación para OB de alarma de sincronismo.
Ventaja: Llamar los drivers de entrada al inicio del OB se traduce en un mínimo desfase y una mínima fluctuación al leer la periferia.	**Ventaja:** es posible resumir *todas* las instrucciones de un eje en *una única* instrucción multiinstancia.
Desventaja: no es posible resumir *todas* las instrucciones de un eje en *una única* instrucción multiinstancia.	Gracias a la MIPP necesaria en los OB de alarma de sincronismo, los datos de periferia leídos son coherentes con independencia de la llamada del driver.

Todos los ejes que están relacionados entre sí deben llamarse en el mismo OB.

En una aplicación isócrona, **todos** los ejes implicados deben estar en la misma memoria imagen parcial del proceso.

Tratamiento de los datos de instancia

Los **datos de instancia** se pueden guardar en un DB propio para cada instrucción o en un DB multiinstancia para todas las instrucciones.

Después de cargar en el módulo los datos de instancia de una instrucción es necesario inicializar la instrucción. Para ello hay que activar la entrada "Init" de la instrucción. Ver también Inicialización y modificaciones de parámetros (Página 11257) yGuardar y cargar el bloque de datos de eje (Página 11262).

Interconexión de instrucciones

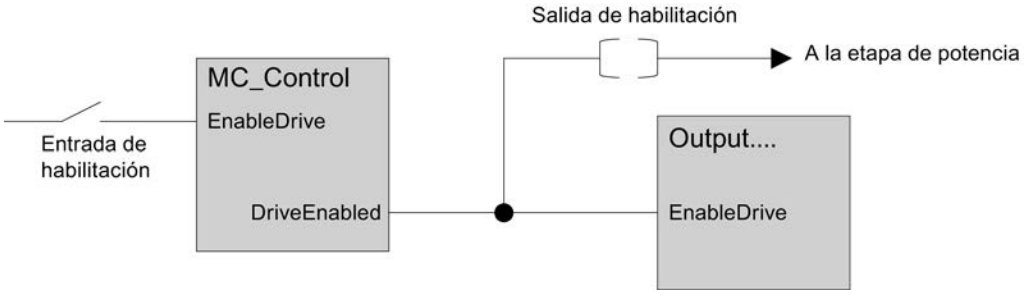
Las **conexiones** entre las instrucciones y el DB de eje representadas en la figura del apartado anterior sobre interacción de las instrucciones, se realizan por medio de estos parámetros:

- Axis
- Init

(ver Significado de los parámetros comunes (Página 11257))

Además, se requieren las interconexiones siguientes:

- La entrada "EnableDrive" de la instrucción MC_Control debe interconectarse con un interruptor de habilitación. La instrucción MC_Control emite una consigna de velocidad únicamente cuando "EnableDrive" = TRUE.
- La entrada "EnableDrive" de la instrucción Output… debe interconectarse con la salida "DriveEnabled" de la instrucción MC_Control. La instrucción Output… emite un valor distinto de cero únicamente cuando "EnableDrive" = TRUE.
- La salida "DriveEnabled" de la instrucción MC_Control debe interconectarse con una entrada de habilitación del accionamiento.



Factores que influyen en la regulación de posición en la CPU

Recomendaciones para optimizar la regulación de posición:

- El tiempo de muestreo debe respetarse con la mayor exactitud posible.
  - En la configuración hardware de la CPU debe parametrizarse la prioridad más alta posible para el OB de alarma cíclica llamando las instrucciones.
  - En la configuración hardware de la CPU debe parametrizarse la carga de ciclo por comunicación más baja posible.
  - Debe utilizarse un entorno isócrono; no se requieren otras medidas
- Al conectar la adquisición de valores de encóder mediante periferia descentralizada (PROFIBUS DP/PROFINET IO), debe ajustarse una velocidad de transferencia del bus de periferia lo más alta posible (p. ej. 12 Mbits/s).

Significado de los parámetros comunes

La tabla siguiente muestra los parámetros que se utilizan en todas las instrucciones.

Parámetro	Declaración	Tipo de datos	Descripción
EN	INPUT	BOOL	Entrada de habilitación del cuadro KOP/FUP
ENO	OUTPUT	BOOL	Salida de habilitación del cuadro KOP/FUP
Axis	IN-OUT	STRUCT	Este parámetro se interconecta con el DB de eje. Con un DB del tipo AXIS_REF, la interconexión es la siguiente: • Axis := DBname.Ax Con un DB que contiene una variable del tipo AXIS_REF, la interconexión es la siguiente: • Axis := DBname.Variablenname.Ax
Init	IN-OUT	BOOL	Este parámetro se interconecta con un bit de inicialización en el DB de eje que no es utilizado por ninguna otra instrucción ("Init.I0" … "Init.I31", verCampo de bits para la función de inicialización (Página 11265)). Con Init = TRUE, la instrucción ejecuta la inicialización y a continuación resetea el bit. Con un DB del tipo AXIS_REF, la interconexión es la siguiente: • Init := DBname.Ax.Init.Ix Con un DB que contiene una variable del tipo AXIS_REF, la interconexión es la siguiente: • Init := DBname.Variablenname.Ax.Init.Ix con Ix = I0 … I31

Inicialización y modificaciones de parámetros

La función de las instrucciones de Easy Motion Control está basada en los datos guardados en el DB de eje. Allí también se encuentran parámetros que describen el eje sustancialmente. Si estos parámetros se modifican, es necesario que **cada una** de las instrucciones de Easy Motion Control llamadas en el programa de usuario ejecute **una vez** su inicialización después de la modificación. En los apartados sobre configuración (Página 11224), estos parámetros se identifican con el siguiente símbolo:

Para la inicialización, en el DB de eje está definido el campo de bits "Init.Ix" y cada instrucción tiene el parámetro "Init". Las instrucciones ejecutan su rutina de inicialización si su entrada "Init" está activada y a continuación desactivan el bit de inicialización interconectado.

Los bits del campo de bits "Init.Ix" del DB de eje son activados:

- cada vez que se llama la instrucción MC_Init.
- por el software de configuración cuando se modifican los parámetros correspondientes del DB de eje.

Para cada instrucción de Easy Motion Control llamada en el programa de usuario, el parámetro "Init" debe interconectarse con un bit de inicialización del DB de eje que no sea utilizado por

ninguna otra instrucción. De este modo se garantiza que las instrucciones se inicializarán cuando sea necesario debido a una modificación de parámetros o a un arranque de la CPU ("Init.I0" … "Init.I31", ver Campo de bits para la función de inicialización (Página 11265)).

La instrucción MC_Init se llama una vez en los siguientes casos:

- cada vez que arranca la CPU
- si se ha modificado sin el software de configuración uno de los parámetros que en los apartados sobre configuración (Página 11224) están identificados con el siguiente símbolo:

Encontrará información complementaria sobre la instrucción de inicialización en MC_Init (Página 7324).

⚠ ADVERTENCIA
Los parámetros que en los apartados sobre configuración (Página 11224) están identificados con el símbolo siguiente únicamente pueden modificarse **con el eje parado**:
Si se incumple este requisito, no se garantiza un movimiento controlado del eje.

⚠ ADVERTENCIA
En la inicialización se calculan valores auxiliares que se guardan en los parámetros estáticos de las instrucciones y en el DB de eje. Por tanto, los bloques de datos de instancia y el DB de eje únicamente pueden sobrescribirse cuando el eje está parado.
Si se incumple este requisito, no se garantiza un movimiento controlado del eje.

Desarrollo de los movimientos de desplazamiento

Inicio de una petición de desplazamiento

Los parámetros de entrada de las instrucciones de desplazamiento se interpretan una vez al iniciar una petición de desplazamiento. La modificación de estos valores mientras se ejecuta la petición no influye en el desplazamiento en curso.

Recorrido máximo

Para cada desplazamiento, el recorrido está limitado a la distancia equivalente a

$$\text{Máximo recorrido} = 2^{24} \times \frac{\text{"Recorrido del eje por vuelta del encóder"}}{\text{"Incrementos por vuelta del encóder"}}$$

Ejemplo:

Recorrido del eje por vuelta del encóder	= 2,5 mm
Incrementos por vuelta del encóder	= 4096
recorrido máximo = 2^{24} x 2,5 mm / 4096	= 10240 mm

Excepción:

Las instrucciones MC_MoveJog y MC_GearIn no tienen limitación de recorrido en el caso de un eje rotativo.

Anulación de un desplazamiento

Una instrucción de desplazamiento activa

- puede anularse en cualquier momento iniciando otra instrucción de desplazamiento,
- puede cancelarse iniciando la instrucción MC_StopMotion,
- puede cancelarse activando el modo manual.

En caso de anulación mediante otra instrucción de desplazamiento, el eje se frena o se acelera hasta alcanzar la velocidad de la instrucción que causa la anulación y a continuación se desplaza hasta la nueva posición de destino sin realizar ninguna parada intermedia. La instrucción de desplazamiento anulada activa la salida "CommandAborted".

Si el nuevo desplazamiento requiere un cambio de sentido del eje, primero se para y a continuación se mueve en sentido contrario (inversión).

Por motivos de seguridad, las deceleraciones de las dos instrucciones de desplazamiento que se anulan deben coincidir. De este modo en caso de inversión se garantiza que no se sobrepase el destino **original** del movimiento.

Uso repetido de una instrucción de desplazamiento

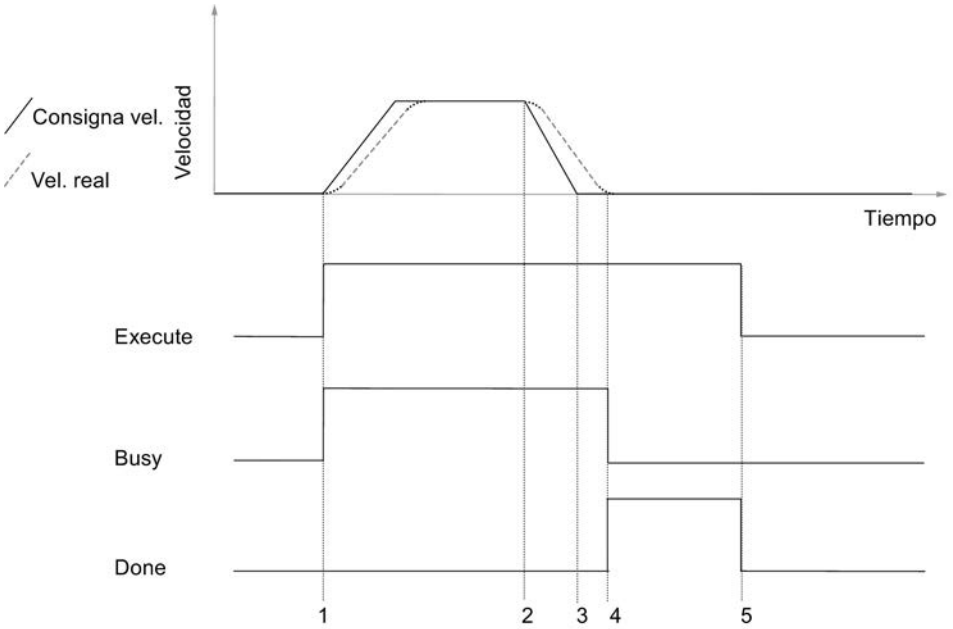
Una instrucción de desplazamiento puede llamarse varias veces para un eje si así lo requiere el programa.

Reacción a errores

Todos los errores que se producen se muestran en el DB de eje, donde el usuario debe acusarlos (ver Señalización y acuse de errores en el DB de eje (Página 11273)). Cada error causa la cancelación de una petición de desplazamiento activa. La condición para que pueda iniciarse una nueva petición de desplazamiento es que no haya ningún error presente.

Diagramas de evolución de señal

La figura siguiente muestra las señales en un desplazamiento sin error.



1 Inicio de la instrucción de desplazamiento

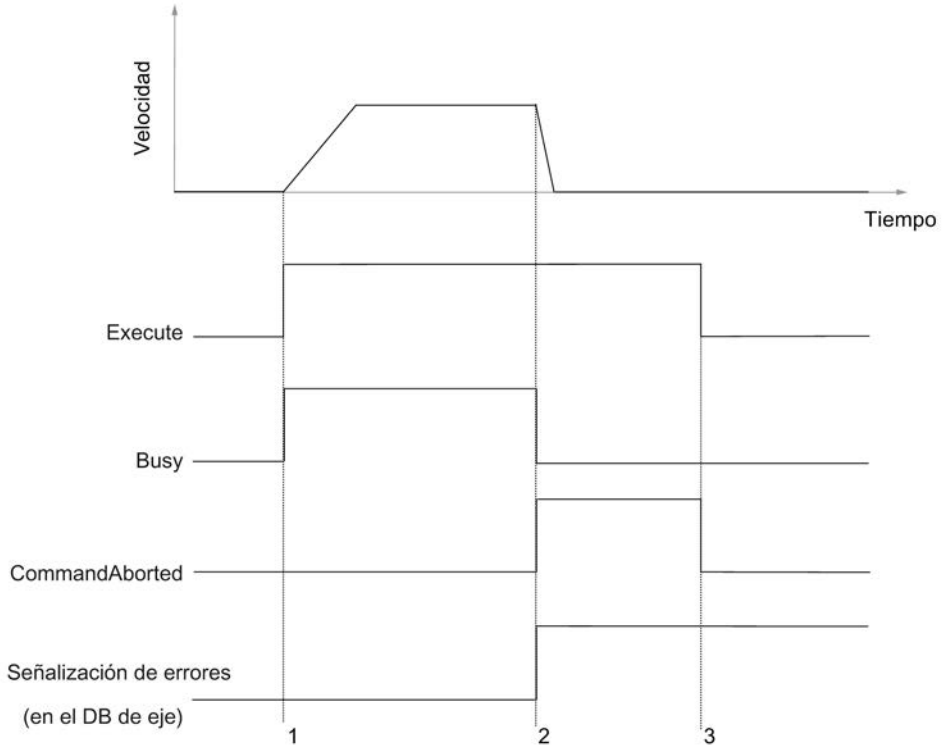
2 Punto de frenado

3 La consigna ha alcanzado la posición de destino

4 El eje ha alcanzado la zona de destino

5 Al resetear "Execute", se resetea "Done". El proceso ha finalizado. Cuando se resetea "Execute" antes de alcanzarse la posición de destino, se activa "Done" durante un ciclo al alcanzarse la posición de destino, y después se vuelve a resetear.

La figura siguiente muestra las señales en un desplazamiento con error.



1 Inicio de la instrucción de desplazamiento

2 Se ha producido un error

3 Al resetear "Execute", se resetea la respuesta "CommandAborted". Cuando se resetea "Execute" antes de producirse el error, se activa "MotionAborted" durante un ciclo al aparecer el error, y después se vuelve a resetear.

Bloque de datos de eje

Tarea

El DB de eje es el elemento central de la gestión de datos de Easy Motion Control. Contiene:

- Parámetros que describen el eje sustancialmente (p. ej. direcciones de periferia)
- Parámetros de la regulación de posición que se calculan a partir de estos parámetros básicos al inicializarse las instrucciones de Easy Motion Control
- Valores actuales de la regulación de posición (p. ej. posición, estados de fallo...)
- Bits de inicialización para coordinar el arranque de las instrucciones

La interacción de estos tipos de parámetros queda garantizada cuando se toma una de las medidas siguientes:

- Editar y transferir el DB de eje con el software de configuración
- Asegurarse de que se llama la instrucción MC_Init (Página 7324) después de modificar determinados parámetros

En ambos casos se comprueba en la medida de lo posible la plausibilidad de los parámetros introducidos y se calculan los valores que Easy Motion Control necesita internamente.

Asimismo, el software de configuración analiza el tipo de modificaciones y se encarga de controlar los procesos de carga.

Guardar y cargar el bloque de datos de eje

ADVERTENCIA

Únicamente está permitido sobrescribir todo el DB de eje cuando el eje está parado.

El DB de eje **NO** se debe cargar en RUN.

Si se carga el DB de eje en el módulo sin tener en cuenta las indicaciones de los apartados siguientes, no se garantiza el movimiento correcto del eje.

Offline

Después de modificar determinados parámetros es necesario que todas las instrucciones de Easy Motion Control ejecuten una inicialización.

En la descripción del DB de eje estos parámetros se identifican con el siguiente símbolo:

Modificar y cargar estos parámetros

- tan solo es posible cuando el eje está parado,
- debe incluir la activación de los bits de inicialización (ver Inicialización y modificaciones de parámetros (Página 11253)).

Online

No es posible que parámetros modificados online (p. ej. las coordenadas del punto de referencia en el caso de encóders absolutos) se apliquen a la gestión de datos offline.

Parámetros de eje

Todos los datos de longitud, velocidad o deceleración mencionados en el apartado "Configurar Easy Motion Control (Página 11222)" hacen referencia a una unidad de longitud cualquiera pero unificada. Todas las indicaciones de tiempo son en segundos.

En la columna "Nombre" de las tablas se indica el nombre de parámetro que hay en el DB de eje, mientras que en la columna "Comentario" figura el nombre de parámetro utilizado en el software de configuración.

Nota

Los parámetros no listados al configurar están reservados para uso interno por las instrucciones de Easy Motion Control y no deben modificarse.

Valores actuales

Resumen

Los siguientes parámetros del DB de eje muestran información actualizada sobre el estado del eje durante un desplazamiento. Está prohibido modificar el contenido de estos parámetros.

Valor real de posición

Aquí se muestra el valor real de posición actual.

Dirección	Nombre	Tipo	Valor inicial	Comentario
104.0	ActPosition	REAL	0.0	**Valor real de posición** Rango de valores: Número en coma flotante [unidad de longitud]

Error de seguimiento

Error de seguimiento = consigna de posición actual – valor real de posición actual

Dirección	Nombre	Tipo	Valor inicial	Comentario
108.0	FollowingDistance	INT	0	**Error de seguimiento** Rango de valores: Número en coma flotante [unidad de longitud]

Distancia residual

Aquí se indica durante un desplazamiento con indicación del destino (MC_MoveAbsolute, MC_MoveRelative) la distancia residual hasta la posición de destino.

Un desplazamiento sin destino (MC_MoveJog, MC_Home) o un error con parada brusca ajustan la distancia residual a 0.

Dirección	Nombre	Tipo	Valor inicial	Comentario
112.0	RemainingDistance	REAL	0.0	**Distancia residual** hasta el destino Rango de valores: Número en coma flotante [unidad de longitud]

Velocidad consignada

Durante un desplazamiento, aquí se muestra la velocidad consignada del eje calculada por la instrucción de desplazamiento.

Dirección	Nombre	Tipo	Valor inicial	Comentario
116.0	NomVelocity	REAL	0.0	**Velocidad consignada** Rango de valores: Número en coma flotante [unidad de longitud/s]

Velocidad real

Aquí se muestra la velocidad real del eje.

Dirección	Nombre	Tipo	Valor inicial	Comentario
120.0	ActVelocity	REAL	0.0	**Velocidad real** Rango de valores: Número en coma flotante [unidad de longitud/s]

Eje sincronizado

Si se utiliza un encóder incremental:

- FALSE: tras arrancar la CPU o detectarse un error de encóder
- TRUE: tras llevarse a cabo una búsqueda del punto de referencia o un posicionamiento del punto de referencia

Si se utiliza un encóder absoluto:

- FALSE: antes del primer posicionamiento del punto de referencia
- TRUE: después del posicionamiento del punto de referencia

Dirección	Nombre	Tipo	Valor inicial	Comentario
124.0	Sync	BOOL	FALSE	**Eje sincronizado** FALSE = no sincronizado TRUE = sincronizado

Valor actual del encóder

Aquí se muestra el valor actual del encóder.

Dirección	Nombre	Tipo	Valor inicial	Comentario
136.0	EncoderValue	DINT	L#0	Valor actual del encóder Rango de valores: Decimal [incrementos]

Campo de bits para la función de inicialización

Funcionamiento del campo de bits

Todas las instrucciones de Easy Motion Control deben inicializarse después de cada arranque de la CPU y después de modificarse determinados parámetros.

Para este fin, en el DB de eje está definido el campo de bits "Init.Ix" y cada instrucción tiene el parámetro "Init". Las instrucciones ejecutan su rutina de inicialización cuando su entrada "Init" está activada. A continuación desactivan el bit de inicialización interconectado.

Cada vez que se llama una instrucción, el parámetro "Init" debe interconectarse con un bit del campo de bits. Este bit no puede ser utilizado por ninguna otra instrucción (ver Inicialización y modificaciones de parámetros (Página 11253)).

Siempre que se llama la instrucción MC_Init o en el software de configuración se modifica un parámetro que no se activa hasta después de haberse inicializado las instrucciones, todos los bits de este campo se ajustan a TRUE simultáneamente. De este modo, todas las instrucciones ejecutan una inicialización la siguiente vez que se llaman.

Dirección	Nombre	Tipo	Valor inicial	Comentario
278.0	Init	STRUCT	TRUE	Campo de bits para inicialización de las instrucciones

Programa de usuario como FB multiinstancia

En caso de guardar todos los parámetros de las instrucciones de desplazamiento en un DB o si se quiere utilizar el mismo programa para varios ejes, el programa de usuario se crea en un FB multiinstancia.

Los datos de eje también deben crearse en este DB multiinstancia. Para ello, en el FB multiinstancia defina una variable estática del tipo AXIS_REF de la librería "Easy Motion Control".

Interconecte los parámetros Axis y Init de las instrucciones de Easy Motion Control con estas variables.

Llame la instrucción MC_Init en el bloque multiinstancia como primera instrucción de forma condicionada. La condición debe activarse en el OB de arranque (p. ej. OB 100) y desactivarse después de llamar la instrucción MC_Init.

Únicamente es posible crear nuevos DB multiinstancia con el TIA Portal o con el editor KOP/AWL/FUP.

Puesta en servicio de Easy Motion Control

Consignas de seguridad

Introducción

Antes de la puesta en marcha debe tenerse en cuenta lo siguiente:

> ⚠ ADVERTENCIA
>
> **Las funciones de puesta en marcha producen el movimiento del eje.**
>
> En la puesta en marcha y durante el funcionamiento puede haber movimientos de desplazamiento incontrolados del eje que causen daños personales y materiales graves.
>
> Para evitar daños personales y materiales es necesario tomar las medidas siguientes:
>
> - Instale un interruptor de PARADA DE EMERGENCIA en las inmediaciones del equipo. Solo así puede garantizarse que en caso de un fallo del hardware o del software sea posible desconectar la instalación de modo seguro.
> - Instale finales de carrera de seguridad que actúen directamente en las etapas de potencia de todos los accionamientos.
> - Asegúrese de que ninguna persona tenga acceso al área de la instalación donde hay piezas en movimiento.

Puesta en marcha del objeto tecnológico Eje

El editor de puesta en marcha le ayudará a poner en servicio el objeto tecnológico Eje y probar su funcionalidad. Se pueden realizar los siguientes tests y mediciones:

- Test de cableado (Página 11267)
- Medición de velocidad (Página 11268)
- Referenciación del eje (Página 11268)
- Medición de recorrido (Página 11269)

Procedimiento

Para abrir el editor de puesta en marcha del objeto tecnológico Eje, haga lo siguiente:

1. Abra la carpeta "Objetos tecnológicos" en el árbol del proyecto.
2. En el árbol del proyecto, abra la instancia correspondiente al objeto tecnológico AXIS_REF (p. ej. EMC_AXIS_1).
3. Haga doble clic en el objeto "Puesta en marcha".

Solución de errores

Si se producen errores durante la puesta en marcha, consulte la información sobre posibles soluciones que se ofrece en el apartado Síntomas de error y soluciones posibles durante la puesta en marcha del programa de usuario (Página 11270).

Información adicional

Encontrará información detallada sobre los parámetros y los botones de las interfaces de puesta en marcha en las respectivas etiquetas emergentes.

Test de cableado

El test de cableado sirve para determinar el sentido de giro del eje.

Procedimiento

Para realizar el test de cableado, haga lo siguiente:

1. En la vista de configuración, haga clic en "Test de cableado".
2. Haga clic en el botón "Activar".
   Se le preguntará si desea activar el control manual.
3. Haga clic en el botón "Aceptar".
   Se establecerá una conexión online entre STEP 7 (TIA Portal) y la CPU.
4. Introduzca una velocidad como porcentaje de la velocidad máxima en el campo de entrada "Velocidad especificada".
5. Introduzca un valor en el campo de entrada "Tiempo de aceleración".
6. Asegúrese de que la posición inicial permita el movimiento del eje en ambos sentidos.
7. Para mover el eje, haga clic en el botón "Jog hacia atrás" o "Jog hacia delante".
8. Indique en "Resultado" si el eje se ha desplazado en el sentido deseado.
9. Si selecciona "Sí" como resultado, haga clic en el botón "Aceptar" de la ventana que se mostrará.
10. Si selecciona "No" como resultado, haga clic en el botón "Sí" de la ventana que se mostrará.
    Los parámetros "Corrección de sentido del encóder" y "Corrección de sentido del accionamiento" cambiarán.

Medición de velocidad del eje

La medición de velocidad sirve para calcular la velocidad máxima del eje.

Procedimiento

Para medir la velocidad, haga lo siguiente:

1. En la vista de configuración, haga clic en "Medición de velocidad".
2. Haga clic en el botón "Activar".
   Se le preguntará si desea activar el control manual.
3. Haga clic en el botón "Aceptar".
   Se establecerá una conexión online entre STEP 7 (TIA Portal) y la CPU.
4. Introduzca una velocidad como porcentaje de la velocidad máxima en el campo de entrada "Velocidad especificada".
5. Introduzca un valor en el campo de entrada "Tiempo de aceleración".
6. Asegúrese de que la posición inicial permita el desplazamiento con una distancia suficiente en al menos un sentido.
7. Haga clic en el botón "Jog hacia atrás" o "Jog hacia delante" y manténgalo pulsado hasta que el valor del campo de salida "Salida v. real" alcance el valor de "Salida consigna".
   Easy Motion Control calcula la velocidad máxima del eje y la indica en el campo de salida "Valor determinado".
8. Si desea guardar en el proyecto la velocidad calculada del eje, haga clic en el botón "Aplicar".
   El valor calculado sobrescribirá la "Velocidad máxima" en la gestión de datos online/offline del objeto tecnológico.

Búsqueda del punto de referencia

La especificación de un punto de referencia está prevista para encóders absolutos y, en este caso, se denomina ajuste del encóder absoluto.

Procedimiento

Para especificar un punto de referencia para un encóder absoluto, proceda del siguiente modo:

1. En la vista de configuración, haga clic en "Referenciación del eje".
2. Haga clic en el botón "Activar".
   Se le preguntará si desea activar el control manual.
3. Haga clic en el botón "Aceptar".
   Se establecerá una conexión online entre STEP 7 (TIA Portal) y la CPU.
4. Introduzca una velocidad como porcentaje de la velocidad máxima en el campo de entrada "Velocidad especificada".
5. Introduzca un valor en el campo de entrada "Tiempo de aceleración".

6. Haga clic en el botón "Jog hacia atrás" o "Jog hacia delante" hasta que el eje se haya desplazado al punto de referencia.
   Alternativa:
   desplace el eje manualmente hasta el punto de referencia.
7. Introduzca un valor de posición como punto de referencia en el campo de entrada "Entrada de la posición de referencia".
   Se determina el offset entre la posición de referencia y la posición física del eje y se almacena en el DB del eje.
8. En la ventana que aparece a continuación, confirme si desea almacenar el offset para el ajuste del encóder absoluto en la gestión de datos offline del objeto tecnológico.

Nota

Si sustituye el encóder absoluto, deberá volver a especificar el punto de referencia.

Medición del recorrido del eje

La medición del recorrido sirve para calcular el recorrido del eje por vuelta del encóder.

Procedimiento

Para realizar la medición del recorrido, proceda del siguiente modo:

1. En la vista de configuración, haga clic en "Medición del recorrido".
2. Haga clic en el botón "Activar".
   Se le preguntará si desea activar el control manual.
3. Haga clic en el botón "Aceptar".
   Se establecerá una conexión online entre STEP 7 (TIA Portal) y la CPU.
4. Introduzca una velocidad como porcentaje de la velocidad máxima en el campo de entrada "Velocidad especificada".
5. Introduzca un valor en el campo de entrada "Tiempo de aceleración".
6. Haga clic en el botón "Jog hacia atrás" o "Jog hacia delante" hasta que el eje se haya desplazado a la posición inicial.
   Alternativa:
   desplace el eje manualmente a la posición inicial.
7. Introduzca un valor de posición como posición inicial en el campo de entrada "Entrada de la posición inicial".
8. Haga clic en el botón "Jog hacia atrás" o "Jog hacia delante" hasta que el eje se haya desplazado a la posición final.
   Alternativa:
   desplace el eje manualmente a la posición final.

9. Introduzca un valor de posición como posición final en el campo de entrada "Entrada de la posición final".
   Easy Motion Control calcula el recorrido del eje por cada vuelta del encóder y lo indica en el campo de salida "Valor calculado".
10. Si desea guardar en el proyecto el valor calculado de recorrido del eje por vuelta del encóder, haga clic en el botón "Aplicar".
    El valor calculado se aplica en la gestión de datos online/offline del objeto tecnológico.

Síntomas de error y soluciones posibles durante la puesta en marcha del programa de usuario

El eje está parado a pesar de una petición de desplazamiento

Si el eje está parado a pesar de una petición de desplazamiento, realice las comprobaciones siguientes:

Error	Soluciones
Cableado incorrecto del accionamiento	• Compruebe si la salida "DriveEnabled" de la instrucción MC_Control está interconectada sin discontinuidades con la entrada de habilitación de la etapa de potencia. • Compruebe si la salida "DriveEnabled" de la instrucción MC_Control está interconectada con la entrada "EnableDrive" del driver de salida.
Error en las direcciones de periferia o números de canal	Ajuste correctamente las direcciones de periferia y los números de canal: En el DB de eje se esperan las direcciones **iniciales** de los módulos. A partir de la dirección inicial y el número de canal (modo de contaje: a partir del canal 0), los bloques driver calculan las direcciones de periferia específicas de cada módulo para consigna y valor real.
Modo de simulación activado	Desactive el modo de simulación ("Sim" = FALSE). Si el modo de simulación está activado ("Sim" = TRUE), **no** se leen ni escriben valores de periferia.
Modo manual activado sin consigna de velocidad	Si está activado el modo manual ("ManEnable" = TRUE), especifique una "velocidad consignada para modo manual" ("ManVelocity" <> 0.0 [unidad de longitud/s]).
Habilitación de accionamiento no concedida	Ajuste la entrada "EnableDrive" de la instrucción MC_Control a TRUE.
Error de eje presente	Si en el DB de eje está activado el bit "Error agrupado" ("Error"), el tipo de error del eje puede verse en los campos de bits para la señalización de errores: • Después de llamar la instrucción MC_Init, el error "Estado de parada de acuse obligatorio" ("Err.StoppedMotion") está siempre activado. Confirme el error activando "Acuse agrupado" "ErrorAck" = TRUE. • En caso de error de parametrización, vuelva a ejecutar MC_Init después de corregir los valores erróneos. • En caso de error del encóder, compruebe si el módulo de lectura de recorrido se ha parametrizado por completo (en caso necesario, con ayuda del software de configuración de módulos). • Otros errores: elimine el error y acúselo activando "Acuse agrupado" "ErrorAck" = TRUE.
"Acuse agrupado" no se resetea	Si después de activar un "Acuse agrupado" ("ErrorAck"), Easy Motion Control no lo detecta ni resetea: • Evite que las instrucciones de Easy Motion Control permanezcan en su rama de inicialización debido a una llamada continua a la instrucción MC_Init. • Compruebe si el módulo de lectura de recorrido reacciona al acuse de error.

Error	Soluciones
La "corrección de la velocidad" es distinta a 0%	Parametrice "corrección de la velocidad" = 0% ("Override").
Petición de desplazamiento no procesada por el programa	En la instrucción de desplazamiento, ajuste el parámetro "Busy" = TRUE.
"Parámetro inadmisible de instrucción de desplazamiento" ("Err.DataErr") o "Destino fuera del rango de desplazamiento" ("Err.TargetErr")	Si carga el DB de eje en la CPU sin utilizar el software de configuración Easy Motion Control, asegúrese de que a continuación se llama la instrucción MC_Init . MC_Init guarda valores auxiliares en el DB de eje.

El eje se desplaza sin petición de desplazamiento

Si el eje se desplaza sin petición de desplazamiento, puede ser por las siguientes causas:

Error	Causas	Soluciones
El eje arranca muy rápido.	El sentido de acción de la regulación está mal ajustado.	Configure con ayuda del asistente "Test de cableado" los parámetros "Corrección de sentido del encóder" ("PolarityEncoder") y "Corrección de sentido del accionamiento" ("PolarityDrive"). De este modo, las instrucciones se adecuan a la instalación de hardware.
El eje se desplaza lentamente.	La etapa de potencia del accionamiento está habilitada y el eje no está regulado (la instrucción MC_Control emite "DriveEnabled" = FALSE).	—
	La salida analógica no está bien cableada con la etapa de potencia del accionamiento.	Corrija el cableado entre la salida analógica y la etapa de potencia del accionamiento.
	El driver de entrada o el driver de salida no accede a los módulos pertenecientes al accionamiento porque la dirección del módulo o el número de canal no están bien ajustados en el DB de eje.	Ajuste los parámetros con ayuda del asistente "Test de cableado".

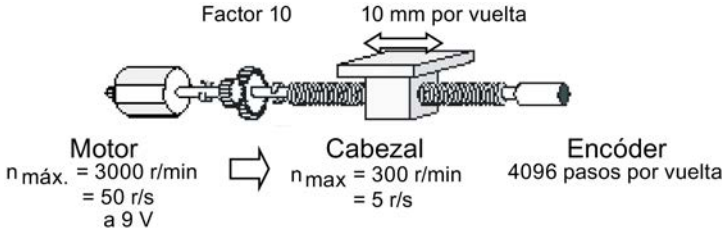
La velocidad del eje es distinta a la esperada o el eje vibra

¿Son correctos los parámetros que describen el eje?

Entre estos parámetros existen ciertas correlaciones que se aclararán por medio del siguiente ejemplo.

- "Ganancia del regulador" ("FactorP"):
  Si la "Ganancia del regulador" ("FactorP") seleccionada es demasiado grande, puede que el eje vibre.
- "Velocidad máxima del eje" ("MaxVelocity")

- "Valor de referencia para velocidad máxima del eje" ("DriveInputAtMaxVel"):
  Ejemplo para un control de ±10 V:



  Es aplicable la siguiente relación:
  Si el módulo de salidas emite 9 V, el accionamiento girará a 3000 r/min. En el cabezal son 300 r/min o 5 r/s. El carro se desplaza 10 mm por cada vuelta del cabezal. De esto resulta que la velocidad máxima del carro es de 10 mm/r • 300 r/min = 50 mm/s.
  Así pues, debe ajustarse lo siguiente:

  – "Velocidad máxima del eje" = 50 mm/s y "Valor de referencia para velocidad máxima del eje" = 9 V.

- "Incrementos por vuelta del encóder" ("StepsPerRev"):
  Para comprobar si el número de incrementos suministrado por el encóder coincide con el valor parametrizado, puede observar el parámetro "Valor del encóder" con el software de configuración mientras gira el **encóder una vez** con la mano.
  Ejemplo: El "valor del encóder" varía 4096 incrementos con cada vuelta del cabezal.
- "Recorrido del eje por vuelta del encóder" ("DisplacementPerRev"):
  Este parámetro asigna un determinado recorrido a una vuelta del encóder. Para comprobar si es correcto, puede observar el parámetro "Valor real de posición" con el software de configuración mientras gira el **encóder una vez** con la mano.
  Ejemplo: El "valor real de posición" varía 10 mm con cada vuelta del cabezal.
- "Tiempo de muestreo" ("Sample_T"):
  El tiempo de muestreo se utiliza, junto con la distancia recorrida desde el último muestreo, para calcular la velocidad del eje. Debe equivaler al tiempo de llamada del OB en el que se llaman las instrucciones de Easy Motion Control.
  En el DB de eje debe introducirse "Sample_T" en segundos.

El posicionamiento no es lo suficientemente exacto

Si el posicionamiento no se lleva a cabo con la suficiente exactitud:

- compruebe el valor ajustado en el parámetro "Incrementos por vuelta del encóder" ("StepsPerRev").
- determine el parámetro "Recorrido del eje por vuelta del encóder" ("DisplacementPerRev") con la mayor exactitud posible (asistente "Medición del recorrido").

Consulte también

Medición del recorrido del eje (Página 11265)

Error de seguimiento máximo rebasado

El error de seguimiento máximo puede rebasarse por varias razones:

- Para que se alcance la "Velocidad máxima del eje" ("MaxVelocity") debe cumplirse lo siguiente:

$$„MaxFollowingDistance“ \quad >= \frac{„MaxVelocity“}{„FactorP“}$$

- El eje real no puede alcanzar la aceleración parametrizada.
- Los parámetros "Velocidad máxima del eje" ("MaxVelocity") y "Valor de referencia para la velocidad máxima del eje" ("DriveInputAtMaxVel"), así como, dado el caso, "Valor de referencia para velocidad 100%" ("DriveInputAt100") no están correctamente ajustados (ver La velocidad del eje es distinta a la esperada o el eje vibra (Página 11267)).

La instrucción MC_GearIn no muestra ni "InGear" ni "Coupled"

La instrucción MC_GearIn no muestra ni "InGear" ni "Coupled" en los casos siguientes:

- El eje esclavo se acopla a un eje maestro que ya está desplazándose y (todavía) no ha alcanzado la velocidad de salida del reductor.
- El eje esclavo ha soltado el acoplamiento debido a un error.

La instrucción MC_GearIn muestra "InGear", pero no "Coupled"

La instrucción MC_GearIn muestra "InGear" pero no "Coupled" en los casos siguientes:

- El eje esclavo ha alcanzado la velocidad de salida del reductor (posiblemente durante un solo ciclo), pero a continuación la ha perdido de nuevo:
  - El eje maestro ha acelerado o decelerado más de lo que le es posible al eje esclavo de acuerdo con su parametrización y el factor de reducción.
  - El eje esclavo no puede alcanzar la velocidad que resulta de la velocidad consignada del eje maestro y del factor de reducción.

Diagnóstico

Señalización y acuse de errores en el DB de eje

Señalización y acuse de errores

Todos los errores detectados por las instrucciones se muestran en el DB de eje de la manera siguiente:

- Mediante activación del error agrupado "Error"
- Mediante una señalización de error específica "Err.xxxx"

En los errores de acuse obligatorio, después de subsanar el error es necesario activar un acuse de error (parámetro "ErrorAck" en el DB de eje, ver abajo).

Existen tres tipos de errores con distintas reacciones:

- Los errores con los que ya **no** se garantiza el correcto funcionamiento de la regulación de posición dan lugar a una **parada brusca**:
  La instrucción MC_Control hace que el eje se detenga con la rampa de frenado consignada en "EmergencyDec". A continuación, la instrucción MC_Control ajusta su salida "DriveEnabled" a "0".
  Después de eliminar el error hay que activar el acuse de error en el DB de eje. A continuación se activa de nuevo DriveEnabled.
- Los errores con los que la regulación de posición continúa funcionando dan lugar a una **parada suave**:
  La instrucción de desplazamiento activa frena el eje con la deceleración parametrizada. El eje continúa estando regulado, pero "DriveEnabled" **no** se resetea.
  En este estado el desplazamiento tan solo puede anularse mediante MC_StopMotion.
  Después de eliminar el error hay que activar el acuse de error en el DB de eje.
- Errores en los parámetros de eje (**errores de parametrización**):
  La instrucción MC_Init comprueba, dentro de lo posible, los parámetros introducidos en el DB de eje. En caso de error, activa el bit "Err.ConfigErr" e introduce la causa exacta del error en "Config.Err_xxx".
  Los errores de parametrización **no** se pueden eliminar mediante acuse. Se impide el arranque del eje. Después de corregir el parámetro mal introducido es necesario llamar de nuevo la instrucción MC_Init. De este modo se resetean las señalizaciones de errores de parametrización si ya no hay ningún error presente.

Nota

Para los errores con parada brusca y parada suave se aplica lo siguiente:

Mientras haya uno o más errores pendientes y no acusados, no es posible realizar ninguna petición de desplazamiento.

Excepción:

Si tan solo se rebasa un final de carrera por software, es posible una petición de desplazamiento en el sentido contrario. Después de salir del final de carrera por software, este error se acusa automáticamente.

Nota

Si el error ha sido detectado por una instrucción de desplazamiento (p. ej. MC_MoveJog o MC_MoveAbsolute) en esa instrucción se activa la salida de error "Error".

Si el error ha sido detectado por una instrucción de driver, en esa instrucción se muestra la causa exacta del error (ver Driver de entrada (Página 7348)).

Señalización de errores en el DB de eje

El error agrupado está en TRUE mientras haya pendiente al menos un error.

Dirección	Nombre	Tipo	Valor inicial	Comentario
124.1	Error	BOOL	FALSE	**Error de grupo** FALSE = no hay error TRUE = error agrupado

Acuse de errores en el DB de eje

Al activar el acuse agrupado se suprimen todas las señalizaciones de error acusables.

El acuse agrupado solo actúa cuando el eje está parado y después de la inicialización.

Tras evaluar el acuse de error, "ErrorAck" se resetea automáticamente.

Dirección	Nombre	Tipo	Valor inicial	Comentario
124.2	ErrorAck	BOOL	FALSE	**Acuse agrupado** para todos los errores. FALSE = no hay acuse TRUE = acusar errores

Error en eje

Error con parada suave

Descripción

Para todos los errores descritos a continuación se aplica lo siguiente: TRUE = el error está presente.

Cuando el error requiere un acuse, esto se indica con el siguiente símbolo:

La tabla siguiente muestra todos los errores con parada suave.

Dirección	Nombre: Err.xxx	Descripción		
128.0	SWLimitMinExceeded	**Inicio del final de carrera por software sobrepasado**		—
		Causa:	• El valor real de posición se encuentra fuera de la zona de trabajo. • Sustitución en caso de un encóder absoluto.	
		Solución:	• Desplace el eje fuera del final de carrera. • Desactive los finales de carrera por software ("SWLimitEnable" = FALSE), vuelva a sincronizar el eje y active de nuevo los finales de carrera por software.	

Dirección	Nombre: Err.xxx	Descripción		Q
128.1	SWLimitMaxExceeded	**Fin del final de carrera por software sobrepasado**		—
		Causa:	• El valor real de posición se encuentra fuera de la zona de trabajo. • Sustitución en caso de un encóder absoluto.	
		Solución:	• Desplace el eje fuera del final de carrera. • Desactive los finales de carrera por software ("SWLimitEnable" = FALSE), vuelva a sincronizar el eje y active de nuevo los finales de carrera por software.	
128.2	TargetErr	**Destino fuera de la zona de desplazamiento**		Q
		Causa:	• **MC_MoveAbsolute**: – El destino se encuentra en un final de carrera por software. – El destino es < "Inicio del eje rotativo" o ≥ "Fin del eje rotativo". – El recorrido de desplazamiento calculado es mayor que la distancia equivalente a 2^{24} incrementos del encóder. – Las magnitudes auxiliares en el DB de eje no se han calculado porque no se ha llamado la instrucción MC_Init. • **MC_MoveRelative**: – El recorrido de desplazamiento introducido lleva a un punto de destino en un final de carrera por software. – El recorrido de desplazamiento predeterminado es mayor que la distancia equivalente a 2^{24} incrementos del encóder. – Las magnitudes auxiliares en el DB de eje no se han calculado porque no se ha llamado la instrucción MC_Init. • **MC_Home**: – La coordenada del punto de referencia se encuentra en un final de carrera por software. – La coordenada del punto de referencia es < "Inicio del eje rotativo" o ≥ "Fin del eje rotativo". – Las magnitudes auxiliares en el DB de eje no se han calculado porque no se ha llamado la instrucción MC_Init.	
		Solución:	• Corrija los parámetros. • Llame la instrucción MC_Init.	
128.3	NoSync	**El eje no está sincronizado**		Q
		Causa:	La instrucción MC_MoveAbsolute se ha iniciado con el eje no sincronizado.	
		Solución:	Sincronice el eje (búsqueda del punto de referencia, posicionamiento del punto de referencia).	
128.4	DirectionErr	**Desplazamiento no permitido en el sentido especificado**		Q
		Causa:	El movimiento no puede continuar hacia un final de carrera por software.	
		Solución:	Inicie un petición de desplazamiento alejándose del final de carrera por software.	
128.5	DataErr	**Parámetro inadmisible de una instrucción de desplazamiento**		Q

Dirección	Nombre: Err.xxx	Descripción		Q
		Causa:	**MC_MoveHome**: • "Velocity" < 0 • "Velocity" > 0 y "Acceleration" ≤ 0 • "Velocity" > 0 y "Deceleration" ≤ 0 • "Velocity" > 0 con encóder absoluto • Para un eje lineal, "Position" está fuera del rango numérico. • Las magnitudes auxiliares en el DB de eje no se han calculado porque no se ha llamado la instrucción MC_Init. **MC_MoveAbsolute**, **MC_MoveRelative** y **MC_MoveJog**: • "Velocity" ≤ 0 • "Acceleration" ≤ 0 • "Deceleration" ≤ 0 • La coordenada de destino introducida o calculada está fuera del rango numérico (con MC_MoveAbsolute y MC_MoveRelative). • Denominador del factor de reducción (RatioDenominator") ≤ 0 (MC_GearIn) • Numerador del factor de reducción ("RatioNumerator") = 0 (MC_GearIn) • Una instrucción de desplazamiento se ha interrumpido y "Deceleration" de la instrucción que ha causado la interrupción no coincide con la deceleración parametrizada en la instrucción interrumpida (ver Anulación de un desplazamiento (Página 11221)). • Las magnitudes auxiliares en el DB de eje no se han calculado porque no se ha llamado la instrucción MC_Init.	
		Solución:	• Especifique los valores permitidos. • Inicialice las instrucciones. • Llame la instrucción MC_Init.	
128.6	StartErr	**Imposible iniciar en el estado actual del eje**		Q
		Causa:	• En una instrucción de desplazamiento o en MC_StopMotion, la entrada Execute se ha activado en un momento en que el eje se encontraba en un estado de error o en modo manual ("ManEnable" = TRUE) o mientras estaba activa una instrucción MC_StopMotion. • La instrucción MC_Home se ha iniciado con el eje no parado. • En la instrucción MC_MoveJog están activados ambos sentidos. • El eje maestro para la instrucción MC_GearIn se encuentra en un estado inadmisible: – Error con parada brusca – MC_Home está activo – Modo manual ("ManEnable" = TRUE)	
		Solución:	• Subsane los errores que pueda haber presentes. • Ponga el eje maestro en un estado admisible.	
128.7	DistanceErr	**Recorrido demasiado largo**		Q

Dirección	Nombre: Err.xxx	Descripción		Q
		Causa:	Un desplazamiento sin destino (MC_Home, MC_MoveJog, MC_GearIn) ha recorrido un trayecto que equivale a más de 2^{24} incrementos del encóder. El eje se frena aproximadamente en este punto. Para un eje lineal sincronizado, la instrucción MC_MoveJog o MC_GearIn ha frenado en un final de carrera por software. Además, en la mayoría de los casos está activado "SWLimitMinExceeded" o "SWLimitMaxExceeded".	
		Solución:	Procure que el trayecto recorrido no sobrepase el límite arriba indicado (ver Recorrido máximo (Página 11254)).	
129.0	MasterErr	**Eje maestro en estado inadmisible**		Q
		Causa:	Durante el acoplamiento del reductor, el eje maestro ha quedado en un estado inadmisible: • Error con parada brusca • MC_Home está activo • Modo manual ("ManEnable" = TRUE)	
		Solución:	Corrija el estado inadmisible del eje maestro.	

Error con parada brusca

Descripción

Para todos los errores descritos a continuación se aplica lo siguiente: TRUE = el error está presente.

Cuando el error requiere un acuse, esto se indica con el siguiente símbolo:

La tabla siguiente indica todos los errores con parada brusca.

Dirección	Nombre: Err.xxx	Descripción		Q
130.0	StoppedMotion	**Eje en estado de parada de acuse obligatorio**		Q
		Causa:	El eje se encuentra en el estado de parada de acuse obligatorio • tras la inicialización por llamada de la instrucción MC_Init. • debido a uno de los errores listados en esta tabla.	
		Solución:	Resuelva un posible error pendiente.	
130.1	EnableDriveErr	**Falta habilitación de accionamiento**		Q
		Causa:	En la instrucción MC_Control, "EnableDrive" no está activado.	
		Solución:	Active "EnableDrive".	

Dirección	Nombre: Err.xxx	Descripción		Q
130.2	FollowingDistErr	**Error de seguimiento rebasado**		Q
		Causa:	El eje no sigue las consignas, o las sigue demasiado lentamente.	
		Solución:	• Conecte el accionamiento. • Optimice la "ganancia del regulador" (DB de eje dirección 44.0 (Página 11234)). • Adapte la "corrección de sentido del encóder" (DB de eje dirección 74.0 (Página 11234)) de acuerdo con la "corrección de sentido del accionamiento" (DB de eje dirección 86.0 (Página 11237)). • Aumente el máximo error de seguimiento admisible (DB de eje dirección 38.0 (Página 11246)).	
130.3	StandstillErr	**Zona de velocidad cero abandonada**		Q
		Causa:	• El eje ha salido de la "zona de velocidad cero" sin petición de desplazamiento. • La "zona de velocidad cero" es menor que la "zona de destino". • La "corrección de sentido del encóder" o la "corrección de sentido del accionamiento" están mal parametrizadas.	
		Solución:	• Compruebe la conexión con la entrada de consignas de la etapa de potencia. • Corrija los parámetros "Zona de parada"/"Zona de destino" (DB de eje dirección 34.0 (Página 11244)/30.0 (Página 11244)) • Durante la primera puesta en marcha: Efectúe un test de cableado (DB de eje dirección 74.0 (Página 11234)/ 86.0 (Página 11237)).	
130.4	TargetApproachErr	**Error en la aproximación al destino**		Q
		Causa:	El valor real no ha alcanzado la "zona de destino" dentro del "tiempo de vigilancia para la aproximación a destino".	
		Solución:	• Optimice la "zona de destino" y el "tiempo de vigilancia para la aproximación a destino" (DB de eje dirección 30.0 (Página 11244)/26.0 (Página 11245)). • Compruebe el accionamiento y el eje.	
130.5	EncoderErr	**Error del encóder**		Q
		Causa:	• El driver de entrada ha detectado un error del encóder. La causa concreta del error se indica en el DB de instancia de la instrucción del driver. Véase Driver de entrada (Página 7348). • En el driver de entrada se ha activado la entrada "EncErr" (véase Driver de entrada (Página 7348), apartado Tratamiento de errores).	
		Solución:	elimine la causa del error.	
130.6	OutputErr	**Error en driver de salida**		Q

Dirección	Nombre: Err.xxx	Descripción		Q
		Causa:	En el driver de salida se ha activado la entrada "OutErr" (véase Driver de salida (Página 7372), apartado Tratamiento de errores).	
		Solución:	elimine la causa del error.	
130.7	ConfigErr	**Señalización agrupada: Datos del eje mal parametrizados**		**Acuse no posible**
		Causa:	Los parámetros del DB de eje son erróneos. Para la causa concreta del error, véase Error de parametrización (Página 11280).	
		Solución:	Modifique los parámetros erróneos y tome medidas para que se llame la instrucción MC_Init.	
131.0	DriveErr	**Error en accionamiento**		**Acuse no posible**
		Causa:	El driver de salida ha detectado un error en la etapa de potencia/el accionamiento. La causa concreta del error se indica en el DB de instancia de la instrucción del driver. Véase Driver de salida (Página 7372).	
		Solución:	elimine la causa del error.	

Error de parametrización

Error de parametrización

La tabla siguiente muestra los posibles errores de parametrización con Easy Motion Control.

La columna "Dirección" contiene la dirección de la señalización de errores en el DB de eje. La columna "Dirección del parámetro" remite a la dirección del parámetro erróneo en el DB de eje (véase Estructura del bloque de datos de eje (Página 7385)).

Dirección	Nombre Config.xxx	Error de parametrización	Dirección del parámetro
132.0	Err_AxisLimit	"Inicio del final de carrera por software" < -2^{24} [unidad de longitud] "Fin del final de carrera por software" > $+2^{24}$ [unidad de longitud] "Fin del final de carrera por software" ≤ "Inicio del final de carrera por software"	6.0 (Página 11238) , 10.0 (Página 11238)
132.1	Err_MaxVelocity	"Velocidad máxima del eje" ≤ 0.0 [unidad de longitud/s]	14.0 (Página 11239)
132.2	Err_MaxAcceleration	"Aceleración máxima" ≤ 0.0 [unidad de longitud/s^2]	18.0 (Página 11241)
132.3	Err_MaxDeceleration	"Deceleración máxima" ≤ 0.0 [unidad de longitud/s^2]	22.0 (Página 11241)
132.4	Err_MonTimeTargetAppr	"Tiempo de vigilancia para la aproximación a destino" ≤ 0.0 [s]	26.0 (Página 11245)
132.5	Err_TargetRange	"Zona de destino" ≤ 0.0 [unidad de longitud]	30.0 (Página 11244)
132.6	Err_StandstillRange	"Zona de velocidad cero" ≤ 0.0 [unidad de longitud]	34.0 (Página 11244)

Dirección	Nombre Config.xxx	Error de parametrización	Dirección del parámetro
132.7	Err_MaxFollowingDist	"Error de seguimiento máx. admisible" ≤ 0.0 [unidad de longitud]	38.0 (Página 11246)
133.0	Err_EmergencyDec	"Deceleración para parada brusca" ≤ 0.0 [unidad de longitud/ s^2]	54.0 (Página 11243)
133.1	Err_StepsPerRev	"Incrementos por vuelta del encóder" < 1	64.0 (Página 11231)
133.2	Err_DisplacementPerRev	"Recorrido del eje por vuelta del encóder" ≤ 0.0 [unidad de longitud]	68.0 (Página 11232)
133.3	Err_NumberRevs	"Número de vueltas del encóder" < 1 para encóder absoluto	72.0 (Página 11232)
133.4	Err_PolarityEncoder	"Corrección del sentido del encóder" distinto de ±1	74.0 (Página 11234)
133.5	Err_PolarityDrive	"Corrección de sentido del accionamiento" distinto de ±1	86.0 (Página 11237)
133.6	Err_DriveInputAtMaxVel	"Valor de referencia para máxima velocidad del eje" ≤ 0.0 o > "DriveInputAt100"	88.0 (Página 11236) , 96.0 (Página 11235)
133.7	Err_AxisLength	Longitud del eje > 2^{24} [unidad de longitud] o longitud del eje > (2^{31} -1) [incrementos]	6.0 (Página 11238) , 10.0 (Página 11238)
134.0	Err_EncoderRange	El rango del encóder no concuerda con la longitud del eje (véase "Recorrido del eje por vuelta del encóder", DB de eje dirección 68.0 (Página 11232))	6.0 (Página 11238) , 10.0 (Página 11238) , 68.0 (Página 11232) , 72.0 (Página 11232)
134.1	Err_MaxVelRotaryAxis	La velocidad máxima y el tiempo de muestreo no concuerdan con la longitud del eje rotativo (véase "Velocidad máxima del eje", DB de eje dirección 14.0 (Página 11239))	14.0 (Página 11239) , 0.0 (Página 11242) , 68.0 (Página 11232) , 72.0 (Página 11232)
134.2	Err_DriveInputAt100	"Valor de referencia para una velocidad del 100%" ≤ 0	96.0 (Página 11235)

Avisos JOB_ERR

Avisos JOB_ERR

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A0	32928	-32608	Acuse negativo al leer del módulo. Módulo extraído durante la lectura o módulo defectuoso.
80A1	32929	-32607	Acuse negativo al escribir en el módulo. Módulo extraído durante la escritura o módulo defectuoso.

JOB_ERR (Hex)	JOB_ERR (Dec)	JOB_ERR (Int)	Significado
80A2 1)	32930	-32606	Error de protocolo para capa 2
80A3 1)	32931	-32605	Error de protocolo en User-Interface/User
80A4 1)	32932	-32604	Fallo en la comunicación del bus K
80B0	32944	-32592	Registro / petición desconocida
80B1	32945	-32591	Indicación de longitud errónea. Parámetro FM_TYPE del DB de canal para el módulo utilizado no activado correctamente.
80B2	32946	-32590	El slot configurado no está ocupado.
80B3	32947	-32589	El tipo de módulo real no es igual al tipo de módulo previsto.
80C0 1)	32960	-32576	El módulo aún no tiene preparados los datos que hay que leer.
80C1 1)	32961	-32575	Los datos de una petición de escritura semejante todavía no se han procesado en el módulo.
80C2 1)	32962	-32574	El módulo está procesando en este momento el número máximo posible de peticiones.
80C3 1)	32963	-32573	Los recursos necesarios (memoria, etc.) están ocupados en este momento.
80C4 1)	32964	-32572	Error de comunicación
80C5 1)	32965	-32571	La periferia descentralizada no está disponible
80C6 1)	32966	-32570	Cancelación de las clases de prioridad (rearranque normal o en segundo plano)
8522	34082	-31454	DB del canal o DB de parámetros demasiado corto. No se pueden leer los datos del DB (petición de escritura).
8532	34098	-31438	El número del DB de parámetros es demasiado grande (petición de escritura).
853A	34106	-31430	DB de parámetros no presente (petición de escritura).
8544	34116	-31420	Error en el enésimo (n > 1) acceso de lectura a un DB tras aparecer un error (petición de escritura).
8723	34595	-30941	DB del canal o DB de parámetros demasiado corto. No se pueden escribir los datos en el DB (petición de lectura).
8730	34608	-30928	DB de parámetros protegido contra escritura en la CPU. No se pueden escribir los datos en el DB (petición de lectura).
8732	34610	-30926	El número del DB de parámetros es demasiado grande (petición de lectura).
873A	34618	-30918	DB de parámetros no presente (petición de lectura).
8745	34629	-30907	Error en el enésimo (n > 1) acceso de escritura a un DB tras aparecer un error (petición de lectura).

1) Los errores 80A2 ... 80A4 así como 80Cx son temporales, es decir, se pueden eliminar tras un tiempo de espera sin la intervención del usuario.

Uso del Team Engineering 12

12.1 Poner en marcha proyectos en equipo

12.1.1 Principios básicos de la puesta en marcha común

Introducción

En el marco del Team Engineering es posible poner en marcha proyectos conjuntamente. Así, varias personas pueden acceder paralela y conjuntamente a una CPU con un máximo de cinco sistemas de ingeniería (ES).

Una de las principales ventajas de esta opción es que en la fase de puesta en marcha es posible editar offline partes de un proyecto maestro de forma simultánea, paralela e independiente entre sí. Durante la carga se visualizan las modificaciones de los demás miembros del equipo en un diálogo para la "Sincronización del software" y, en lo posible, se sincronizan automáticamente.

Algunas funciones online también pueden ejecutarse simultánea y paralelamente en la CPU utilizada conjuntamente desde varios sistemas de ingeniería, por ejemplo:

- observar bloques en la CPU
- forzar bloques en la CPU
- funciones Trace

Las siguientes funciones online no pueden ejecutarse de forma simultánea:

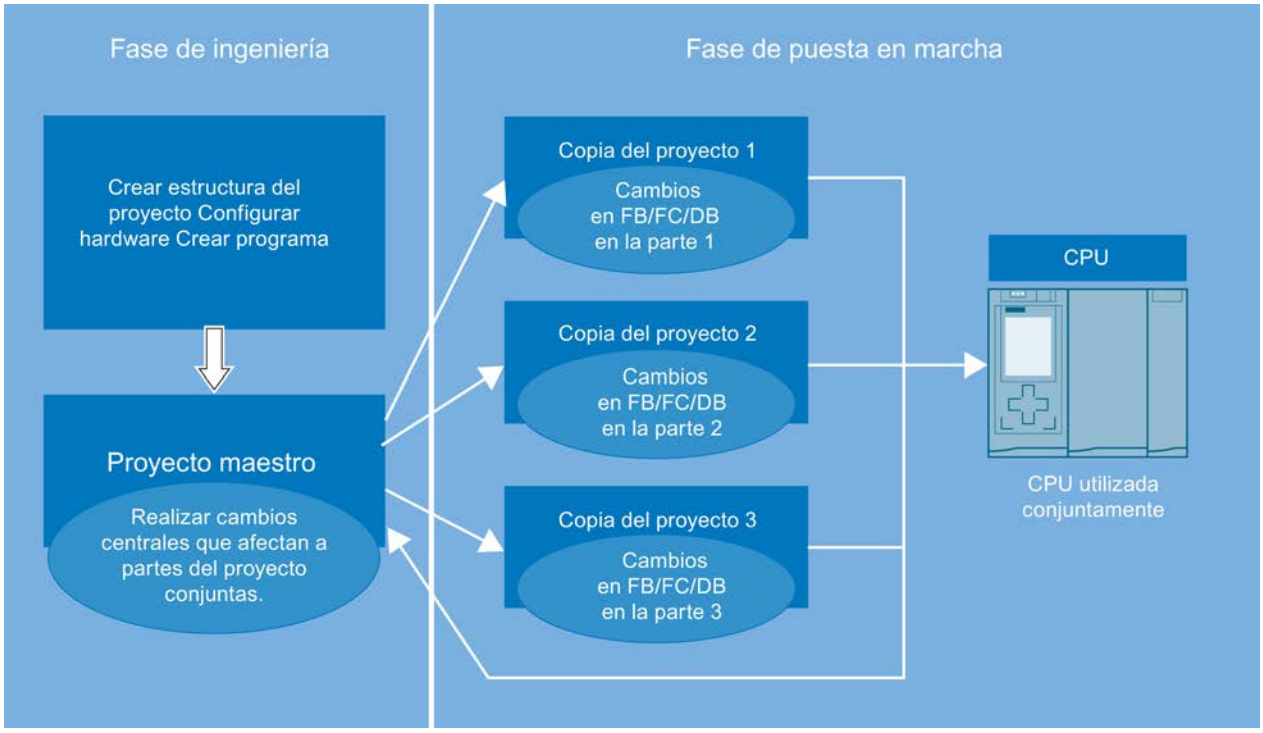
- Forzado permanente: solo un ES puede realizar forzados permanentes en la CPU en cada momento.
- Carga: solo un ES puede cargar en la CPU en cada momento.

Encontrará información adicional sobre la ingeniería de equipo en Servicio técnico y soporte en el Siemens Industry Online Support (http://support.automation.siemens.com/WW/view/es/82142829).

Crear un proyecto maestro

La base para la puesta en marcha conjunta es un "proyecto maestro" estructurado conforme a las reglas especificadas (Página 11290) que contiene la configuración hardware completamente configurada con todas las variables y bloques necesarios. Este proyecto se

carga en la CPU utilizada conjuntamente y, seguidamente, se distribuye como "proyecto maestro" mediante copias del proyecto a un máximo de cinco sistemas de ingeniería.



Trabajar con copias del proyecto

Cada sistema de ingeniería edita únicamente las partes que le han sido asignadas dentro de la correspondiente copia del proyecto. Esto es muy importante para evitar conflictos a la hora de cargar y para impedir que se sobrescriban los bloques modificados por otros ES.

Después de la edición, cada ES carga sus partes en la CPU utilizada conjuntamente. El TIA Portal le ayuda a la hora de realizar la carga en la CPU. En el diálogo "Sincronización del software antes de la carga en dispositivo" (denominado en adelante "diálogo de sincronización") se visualizan mediante una comparación online-offline las modificaciones realizadas desde la última carga. En él se visualizan propuestas para sincronizar los datos offline que ha editado el editor con los datos online existentes a la hora de cargarlos en la CPU.

Si hubiera cambios "rivalizantes" deberán sincronizarse manualmente.

Se producen cambios rivalizantes en los siguientes casos:

- Cuando varias personas editan paralelamente el mismo bloque en diferentes copias del proyecto.
- Cuando se agregan, modifican o borran variables PLC (I, Q, M, T, C) en las copias del proyecto.
- Cuando en las copias del proyecto se editan bloques que contienen referencias a la configuración hardware.
- Cuando se editan bloques F en las copias del proyecto.

Los editores no pueden cargar los cambios en la CPU hasta que se han eliminado todos los conflictos.

El procedimiento para editar y cargar las copias del proyecto desde los diferentes sistemas de ingeniería puede repetirse tantas veces como sea necesario hasta que la puesta en marcha conjunta haya concluido.

Volver a integrar copias del proyecto en el proyecto maestro

Para finalizar la puesta en marcha conjunta es necesario volver a integrar las diferentes copias del proyecto en el proyecto maestro. De este modo se guardan también los datos que existen únicamente en los proyectos offline, es decir, que no se cargan en la CPU. Aquí se incluyen, por ejemplo, las listas de textos y los idiomas del proyecto creados en las copias.

Para ello, abra el proyecto maestro y una copia del proyecto en cada caso como objeto de referencia. Por medio del editor de comparación es posible determinar las diferencias entre ambos proyectos realizando una comparación offline/offline. Copie los objetos editados de la copia correspondiente del proyecto al proyecto maestro. Realice esta acción para todas las copias del proyecto existentes.

De este modo, al finalizar la puesta en marcha en equipo está disponible un proyecto maestro ejecutable con todos los datos de proyecto generados.

La integración de las copias en el proyecto maestro también es necesaria cuando durante la fase de puesta en marcha se detecta la necesidad de realizar más ampliaciones o correcciones en los elementos centrales utilizados conjuntamente como, p. ej., una ampliación de la tabla de variables PLC o agregar un nuevo componente de hardware.

Indicaciones referentes a la compatibilidad

Para la puesta en marcha en equipo son válidas las siguientes reglas de compatibilidad:

- Si un sistema de ingeniería con un TIA Portal V13 o superior está online en la CPU y realiza una carga, ningún otro ES, sea cual sea la versión que tenga, podrá establecer una conexión online.
- Si un sistema de ingeniería con un TIA Portal < V13 está online en la CPU, ningún otro ES, sea cual sea la versión que tenga, podrá establecer una conexión online.

Consulte también

Requisitos para la puesta en marcha común (Página 11285)

Procedimiento para la puesta en marcha en equipo (Página 11287)

12.1.2 Requisitos para la puesta en marcha común

Requisitos de software y hardware

Para poner en marcha proyectos en equipo deben cumplirse los requisitos mínimos de software y hardware para la instalación del TIA Portal V13.

Adicionalmente, existen los requisitos siguientes:

Software:

- El TIA Portal V13 o superior y el paquete de software "SIMATIC STEP 7 Professional" están instalados en los sistemas de ingeniería correspondientes.
- Todos los sistemas de ingeniería deben tener instalada la misma versión de software.
- El proyecto maestro debe estar creado con V13 o superior.

Hardware:

- Se dispone de una CPU S7-1500 configurada y parametrizada a partir de la versión de firmware V1.5.
- Los sistemas de ingeniería (máx. 5) pueden establecer una conexión online con dicha CPU.

Requisitos para el trabajo paralelo en equipo

Para poner en servicio proyectos en equipo deben cumplirse los requisitos siguientes:

- Hay un proyecto creado con la configuración hardware completa y el programa de usuario completamente programado y listo para ser puesto en marcha.
- El proyecto está cargado en la CPU y definido como "proyecto maestro".
- Varios sistemas de ingeniería tienen acceso a esta CPU y pueden establecer una conexión online con ella.
- Los usuarios conocen las reglas y los procedimientos definidos para trabajar en equipo en una CPU.

Información sobre el modo de compatibilidad

La funcionalidad de la puesta en marcha en equipo no está disponible en modo de compatibilidad.

Consulte también

Requisitos de software y hardware de STEP 7 (Página 93)

Principios básicos de la puesta en marcha común (Página 11279)

Procedimiento para la puesta en marcha en equipo (Página 11287)

Reglas para la puesta en marcha común (Página 11290)

12.1.3 Procedimiento para la puesta en marcha en equipo

Introducción

Cuando se pone en marcha un proyecto en equipo en la ingeniería de equipo, es muy importante que todos los integrantes del proyecto se ajusten a un procedimiento definido.

Solo si se mantiene el procedimiento definido queda garantizado que todos los cambios y todas las correcciones realizados en el proyecto puedan sincronizarse y aplicarse automáticamente y que al cargar no se sobrescriban involuntariamente cambios realizados por determinados editores o incluso se pierdan por culpa de cambios rivalizantes.

Procedimiento para crear el proyecto maestro

El proyecto maestro se crea siguiendo los pasos descritos a continuación:

1. Cree un proyecto maestro que ya contenga su estructura completa.
2. Configure totalmente el hardware para el proyecto maestro.
3. Defina el idioma de proyecto que utilizarán de forma exclusiva todos los sistemas de ingeniería que intervienen.
4. Cree todas las variables y todos los bloques necesarios en el proyecto maestro.
5. Cree carpetas y grupos propios para los bloques que deben editar los diferentes sistemas de ingeniería.
6. Cree un programa de usuario totalmente programado y ejecutable.
7. Cargue el proyecto maestro en la CPU utilizada conjuntamente.
8. Guarde el proyecto maestro cada vez que efectúe una carga.

Observe también a este respecto las reglas para la puesta en marcha en equipo (Página 11290).

Procedimiento para la puesta en marcha en equipo

La puesta en marcha en equipo se lleva a cabo siguiendo los pasos descritos a continuación:

1. Cargue el proyecto maestro en la CPU utilizada conjuntamente.
2. Cree copias del proyecto maestro y distribúyalas entre los editores que trabajan en los sistemas de ingeniería correspondientes.
3. Informe a todos los editores de las partes que pueden editar de las copias del proyecto y que pueden cargar en la CPU.
4. Haga editar las copias del proyecto en los diferentes sistemas de ingeniería.
5. Tras la edición, cada uno de los sistemas de ingeniería cargará sucesivamente los cambios en la CPU.

6. Durante la carga, todos los cambios se determinan automáticamente por medio de una comparación online-offline. En el cuadro de diálogo de sincronización que aparece se muestran propuestas para sincronizar los datos modificados, siempre que sea posible. Posiblemente primero deba cargar en su ES bloques modificados por otros editores antes de poder cargar los cambios en la CPU. Esto es necesario para actualizar en su copia del proyecto las correcciones de otros editores que ya se han cargado en la CPU.
   Existen las siguientes posibilidades de sincronización:
   – Cargar bloque: la CPU contiene bloques modificados que deben actualizarse en la copia del proyecto del usuario.
   – Cargar bloque: la CPU contiene bloques nuevos que deben cargarse en la copia del proyecto.
   – Bloques con cambios rivalizantes: en este caso no es posible realizar una sincronización por parte del sistema; el conflicto debe solucionarse manualmente.
7. Solucione manualmente los conflictos que hayan podido surgir debido a cambios rivalizantes o a cambios en objetos centrales.
8. Cargue la copia del proyecto en la CPU cuando se hayan eliminado todos los conflictos.
9. Guarde su copia del proyecto cada vez que efectúe una carga.
10. Repita la edición de las copias del proyecto y la carga en la CPU tantas veces como sea necesario hasta que la puesta en marcha en equipo haya concluido.
11. Vuelva a integrar las copias terminadas en el proyecto maestro para que se guarden también los datos del proyecto que solo están disponibles offline.

Al hacerlo tenga también en cuenta Reglas para la puesta en marcha común (Página 11290).

Procedimiento para sincronizar manualmente cambios rivalizantes

La sincronización manual de cambios rivalizantes se lleva cabo siguiendo los pasos descritos a continuación:

1. Inicie el editor de comparación para solucionar manualmente los conflictos visualizados en el cuadro de diálogo de sincronización al cargar en la CPU. Seleccione para ello la CPU utilizada conjuntamente del árbol del proyecto y elija el comando "Comparar > offline/online" del menú contextual. Dependiendo del estado de los objetos se pueden definir determinadas acciones. No obstante, tenga en cuenta que en una sincronización solo es posible ejecutar acciones en un sentido.
2. Seleccione primero la acción "Cargar de dispositivo" para todos los bloques modificados por otros editores en la CPU que desea aplicar.
3. En caso necesario, realice una comparación detallada de los bloques para identificar las diferencias entre sus bloques offline y los bloques online cargados en la CPU.
4. Corrija manualmente los cambios rivalizantes en los bloques.
5. Seguidamente cargue los bloques afectados con el comando "Continuar sin sincronización" en la CPU.
6. Guarde su proyecto o la copia del proyecto cada vez que efectúe una carga.

Observe también a este respecto las reglas para la puesta en marcha en equipo (Página 11290).

Nota

Para evitar que se vuelvan a realizar cambios en modo online mientras se esté sincronizando manualmente un sistema de ingeniería, ninguno de los demás sistemas de ingeniería del equipo deberá realizar cargas en la CPU utilizada conjuntamente durante la sincronización manual.

Procedimiento para integrar copias del proyecto en el proyecto maestro

La integración de las copias en el proyecto maestro se lleva a cabo siguiendo los pasos descritos a continuación:

1. Abra el proyecto maestro y la copia que deba integrarse como proyecto de referencia.
2. Copie las partes del programa que haya editado de la copia en cuestión en proyecto maestro y confirme la sobrescritura de los objetos existentes. Otra posibilidad consiste en utilizar el editor de comparación para incorporar las partes del programa al proyecto maestro.
3. Guarde el proyecto maestro y cárguelo en la CPU utilizada conjuntamente con el comando "Continuar sin sincronización".
4. Guarde el proyecto maestro cada vez que efectúe una carga.

Observe también a este respecto las reglas para la puesta en marcha en equipo (Página 11290).

Procedimiento para modificar objetos centrales en un proyecto maestro

La modificación de objetos centrales que afectan a todas las partes del programa se lleva a cabo siguiendo los pasos descritos a continuación:

1. En primer lugar, deje de editar las copias del proyecto.
2. Integre sucesivamente todas las copias existentes en el proyecto maestro tal como se describe anteriormente.
3. Cargue el proyecto maestro en la CPU utilizada conjuntamente con el comando "Continuar sin sincronización".
4. Realice los cambios deseados en los objetos utilizados conjuntamente como, p. ej., en la configuración hardware o en la tabla de variables PLC. Para modificar otros objetos centrales, como objetos tecnológicos, bloques F, OBs, etc., el procedimiento es análogo.
5. Vuelva a cargar el proyecto maestro en la CPU cuando la modificación haya concluido.
6. Guarde el proyecto maestro cada vez que efectúe una carga.
7. Distribuya las copias actualizadas del proyecto a los sistemas de ingeniería en cuestión para que puedan seguir editándolas.

Observe también a este respecto las reglas para la puesta en marcha en equipo (Página 11290).

Consulte también

Principios básicos de la puesta en marcha común (Página 11279)

Requisitos para la puesta en marcha común (Página 11281)

Reglas para la puesta en marcha común (Página 11290)

12.1.4 Reglas para la puesta en marcha común

Introducción

Cuando se pone en marcha un proyecto en equipo en la ingeniería de equipo, para que el trabajo en equipo tenga éxito es muy importante que todos los integrantes del proyecto se ajusten a reglas predefinidas.

Reglas para el proyecto maestro

Deben observarse las reglas siguientes:

- Cree un proyecto maestro que sea adecuado para el trabajo en equipo.
- Divida el programa de usuario en partes independientes.
- Utilice "grupos" para separar también visualmente las partes del programa.
- Utilice un OB principal y por cada parte del programa un FC central que llame las funciones de la parte correspondiente.
- Siempre que sea posible, cree una tabla de variables PLC propia para cada parte.
- Defina en el proyecto maestro un idioma de proyecto que no podrá modificarse en las copias del proyecto.
- Si desea intercambiar datos entre las partes del programa, utilice los parámetros de interfaz de FC y FB (IN, OUT, INOUT) o bien bloques de datos globales.
- Utilice bloques de datos globales y no marcas, para guardar los datos de las diferentes partes del programa.
- No asigne nombres diferentes a bloques con direcciones idénticas.
- No asigne nombres idénticos a bloques con direcciones diferentes.

Ejemplo de la estructura del programa en el proyecto maestro

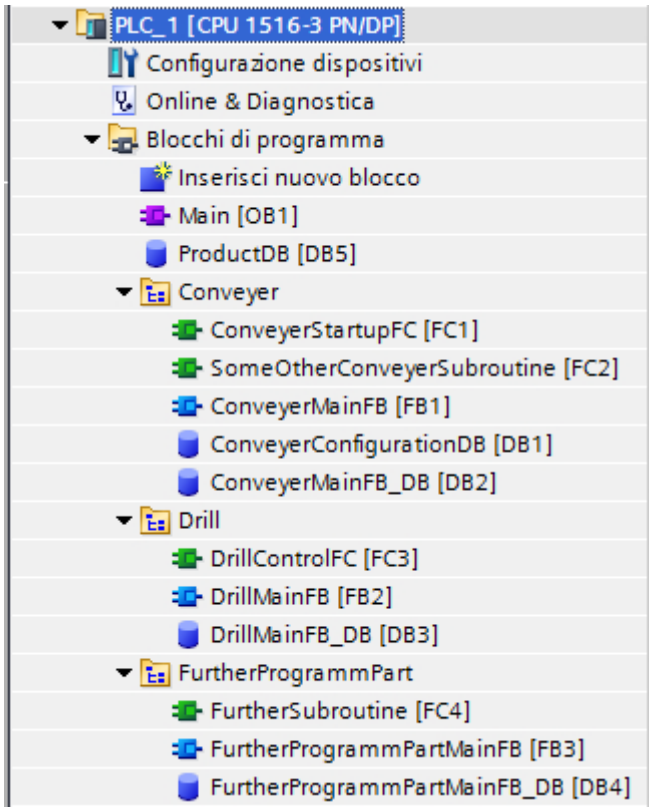
A continuación se muestra un ejemplo de un proyecto maestro con una estructura adecuada para trabajar en equipo y editarlo paralelamente durante la puesta en marcha conjunta.

Las partes del programa que deben editar los diferentes sistemas de ingeniería se dividen en las siguientes:

- Parte 1: "Conveyer"
- Parte 2: "Drill"
- Parte 3: "FurtherProgramPart"

En cada parte del programa existe un "Main-FB" que llama las funciones subordinadas para esta parte del programa.

Ejemplo: en la parte "Conveyer", "ConveyerMainFB" llama la función "ConveyerStartup".



Gracias a este tipo de subdivisión en la estructura del proyecto, en la ingeniería de equipo es posible editar paralelamente las diferentes partes del programa y sincronizar automáticamente los cambios durante la fase de puesta en marcha en equipo.

Reglas para trabajar en equipo en una CPU

Deben observarse las reglas siguientes para el trabajo en equipo:

- Cada editor edita en su copia del proyecto únicamente los bloques que se le han asignado en los grupos que se le han asignado.
- En las copias del proyecto solo pueden modificarse FB, DB, FC y UDT, es decir, los siguientes elementos del programa no pueden editarse en las copias del proyecto:
  - Configuraciones hardware
  - Tabla de variables PLC
  - Bloques de organización
  - Objetos tecnológicos
  - Listas de textos e idioma del proyecto
  - Bloques F
- En lugar de marcas, en la programación deberán utilizarse bloques de datos globales.
- En lugar de temporizadores y contadores SIMATIC, en la programación deberán utilizarse temporizadores y contadores IEC.
- El idioma definido en el proyecto maestro no se puede modificar en las copias del proyecto.

Reglas para editar objetos centrales utilizados conjuntamente

Deben observarse las reglas siguientes para objetos centrales utilizados conjuntamente:

- Los cambios realizados en objetos centrales utilizados conjuntamente solo pueden realizarse en el proyecto maestro.
  Algunos de estos objetos son:
  - Configuraciones hardware
  - Tabla de variables PLC
  - Bloques de organización
  - Objetos tecnológicos
  - Listas de textos e idioma del proyecto
  - Bloques F
- Antes de poder modificar objetos centrales utilizados conjuntamente en el proyecto maestro, es necesario integrar de nuevo en el proyecto maestro las diferentes versiones de trabajo de las copias del proyecto.
- Posteriormente es posible realizar los cambios en el proyecto maestro y volver a cargarlo en la CPU de uso común.
- Una vez finalizada la modificación es posible crear nuevas copias del proyecto y distribuirlas a los sistemas de ingeniería para seguir editándolas.

Reglas para cargar en la CPU

Deben observarse las reglas siguientes para la carga:

- Los editores solo pueden cargar las copias del proyecto en la CPU uno detrás de otro; en cada momento solo puede realizar la carga un sistema de ingeniería.
- A la hora de cargar, ningún otro sistema de ingeniería puede estar conectado online con la CPU.
- Ejecute la sincronización que se propone automáticamente en el diálogo de sincronización. De este modo se garantiza que no se sobrescriban involuntariamente cambios de otros editores.
- En el cuadro de diálogo de sincronización elimine los conflictos de nombres cambiando el nombre del bloque offline antes de cargarlo en la CPU. Se indicará un conflicto de nombres si desea cargar en la CPU un bloque que ya existe en la CPU con el mismo nombre.
- Las "modificaciones rivalizantes" deben eliminarse manualmente. Se producen cambios rivalizantes cuando dos editores editan simultáneamente el mismo bloque en dos sistemas de ingeniería. El primer editor puede cargar los cambios en la CPU sin problemas, pero cuando el segundo editor quiere cargar también en la CPU el mismo bloque con los cambios que ha realizado, en el cuadro de diálogo de sincronización se muestra un conflicto que no puede solucionarse automáticamente. Al realizar la carga, los cambios anteriores realizados en este bloque se perderían debido a la sobrescritura. En este caso, los editores deben decidir qué cambios deben aplicarse y cuáles se rechazarán. También puede agrupar manualmente los cambios por medio de la comparación detallada de los bloques en el editor de comparación.
  En todo caso, este tipo de cambios deberían evitarse sin falta con una buena estructuración del proyecto y una buena coordinación entre los integrantes del equipo.
- Los cambios realizados en objetos centrales se muestran en el cuadro de diálogo de sincronización, aunque no pueden sincronizarse automáticamente. Los cambios realizados en objetos centrales deben realizarse siempre desde el proyecto maestro.
- Guarde su proyecto o la copia del proyecto cada vez que efectúe una carga.

Nota

Creación y posterior borrado de objetos en las copias del proyecto

Tenga en cuenta que la sincronización del software sigue siendo necesaria antes de cargar si en una copia del proyecto se crea un objeto y se vuelve a borrar inmediatamente.

La creación de un objeto implica un cambio en la gestión interna de los datos que no se deshace al borrar el objeto seguidamente. Por lo tanto, siempre será necesario realizar una sincronización antes de la siguiente carga, aunque el objeto recién creado se haya borrado inmediatamente y no se vean cambios en los objetos existentes en la copia del proyecto.

Reglas para las funciones online

Para utilizar las funciones online, deben observarse las siguientes reglas:

Observar y forzar:

- Pueden observar y forzar bloques en la CPU hasta cinco sistemas de ingeniería a la vez.
- Un bloque lógico específico no puede ser observado y forzado por más de un sistema de ingeniería al mismo tiempo.
- Sin embargo otros sistemas de ingeniería pueden observar y forzar simultáneamente otros bloques lógicos.

Forzado permanente

- No se puede forzar permanentemente en la CPU desde más de un sistema de ingeniería a la vez.
- El sistema de ingeniería que inicia el forzado permanente ha lanzado la petición de forzado permanente en exclusiva. El resto de sistemas de ingeniería son informados de que se está ejecutando una petición de forzado permanente, pero no pueden acceder a dicha petición ni ver ningún resultado.
- En cuanto el sistema de ingeniería al que pertenece la petición de forzado permanente finaliza la conexión online, la petición de forzado permanente puede ser asumida por otro ES que establezca una conexión online con la CPU y modifique la petición de forzado permanente existente.

Funciones Trace:

- Pueden realizar funciones Trace hasta cuatro sistemas de ingeniería a la vez.
- El sistema de ingeniería que inicia el Trace ha lanzado la petición Trace en exclusiva. El resto de sistemas de ingeniería pueden ver esta petición en el editor de Trace, pero no pueden acceder a ella.
- En cuanto el sistema de ingeniería al que pertenece la petición Trace finaliza el editor de Trace, la petición Trace puede ser asumida por otro ES que vuelva a abrir el editor de Trace.

Consulte también

Principios básicos de la puesta en marcha común (Página 11279)

Requisitos para la puesta en marcha común (Página 11281)

Procedimiento para la puesta en marcha en equipo (Página 11283)

12.2 Intercambio de datos con ingeniería interproyecto (Inter Project Engineering) (IPE)

12.2.1 Principios básicos de la ingeniería interproyecto (IPE)

Introducción a la ingeniería interproyecto (IPE)

Con la funcionalidad de ingeniería interproyecto, en lo sucesivo IPE, se pueden intercambiar los datos de control de un proyecto de origen con otros proyectos mediante un proxy de dispositivo.

Estos datos se pueden incorporar a otros proyectos, p. ej. para la visualización en HMI, y ahí ser utilizados para otras configuraciones.

Así, la ingeniería interproyecto le permite, de un modo simple, disponer de datos de control interproyecto de una programación PLC para la configuración HMI.

Con el objeto "Datos del proxy del dispositivo" se efectúa un intercambio coherente y cómodo de los datos de control entre varios proyectos sin redundancia de labores de configuración.

De esta manera, el ingeniero de configuración HMI puede trabajar en un proyecto sin necesidad de que exista configuración hardware en su proyecto. Así, los trabajos los trabajos en el proyecto PLC y en el proyecto HMI pueden realizarse paralela y simultáneamente.

Con el proxy de dispositivo se pueden intercambiar los siguientes datos de control entre los proyectos:

- Bloques de programa
- Objetos tecnológicos
- Variables PLC
- Avisos PLC

En el intercambio se transfieren de forma automática los datos siguientes:

- Interfaces del controlador
- Procesadores y módulos de comunicación configurados

Con este intercambio automático de datos se asegura que, además de los datos seleccionados por usted, también se transfieran las interfaces y los procesadores y módulos de comunicación configurados. Estos datos son necesarios para un procesamiento ulterior correcto y coherente en su proyecto de destino.

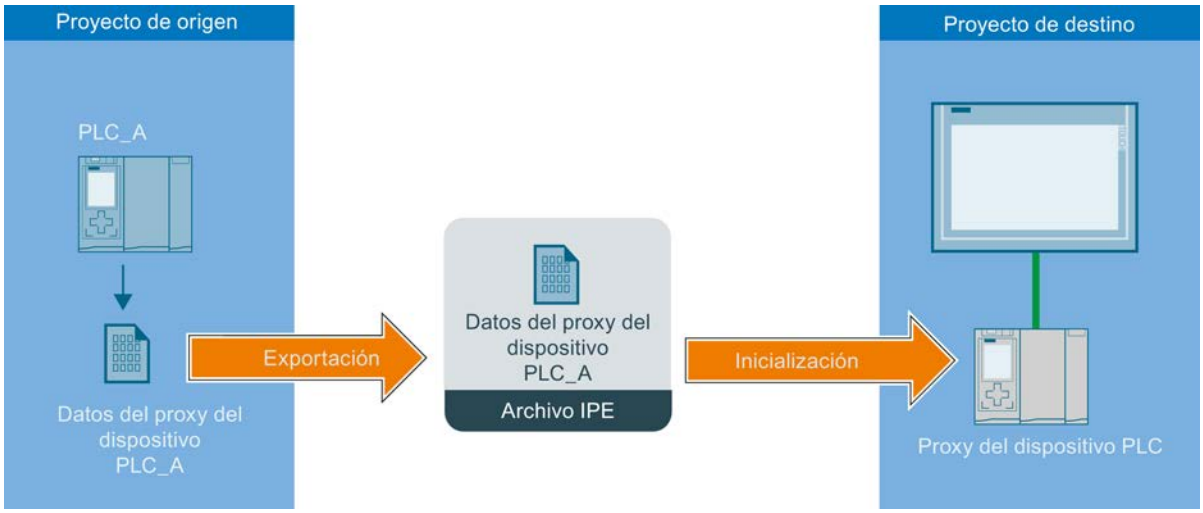
Intercambio de datos entre varios proyectos

Para el intercambio de datos entre varios proyectos con ingeniería interproyecto tiene las opciones siguientes:

- Intercambio de datos de control mediante un archivo IPE
- Intercambio de datos de control mediante un archivo de proyecto

Intercambio de datos de control mediante un archivo IPE

En la figura siguiente se muestra el intercambio de datos entre varios proyectos por medio de un archivo IPE.



Los datos de control del PLC procedentes del proyecto de origen se transfieren al PLC del proyecto de destino mediante el objeto "Datos del proxy del dispositivo".

Los datos de control que se intercambian con un archivo IPE tienen que proceder de proyectos de TIA Portal V13 o superior.

Intercambio de los datos de control mediante un archivo de proyecto

La figura siguiente muestra el intercambio de datos entre varios proyectos por medio de un archivo de proyecto.

Los datos de control de PLC procedentes del proyecto de origen se transfieren al PLC del proyecto de destino mediante un archivo de proyecto.

Es posible adoptar y reutilizar cómodamente datos de control tanto de antiguos proyectos del TIA Portal como de proyectos STEP 7 Classic que no están integrados en el TIA Portal en los proyectos TIA Portal actuales.

Consulte también

Requisitos para la ingeniería interproyecto (IPE) (Página 11297)

Procedimientos al trabajar con ingeniería interproyecto (IPE) (Página 11298)

12.2.2 Requisitos para la ingeniería interproyecto (IPE)

Requisitos de software y hardware

Para aprovechar la funcionalidad de la ingeniería interproyecto deben cumplirse los requisitos mínimos de software y hardware para la instalación del TIA Portal V13 o superior.

Adicionalmente, existen los requisitos siguientes:

- El TIA Portal V13 o superior y el paquete de software "SIMATIC STEP 7 Professional" están instalados.
- Utilice una CPU de la familia S7-1200/1500 o S7-300/400.
- Todos los controladores implicados deben tener instalada la misma versión de software.

Requisitos para IPE

Para poder importar datos de control procedentes de proyectos, existen los requisitos siguientes:

- Haber creado un proyecto con configuración hardware y datos PLC que puedan transferirse a otro proyecto.
- Haber comprobado si el proyecto es coherente antes de transferir los datos de control mediante los datos del proxy del dispositivo.
- Conocer los procedimientos definidos para la ingeniería de interproyecto.

Indicaciones sobre el modo de compatibilidad

La funcionalidad para la ingeniería interproyecto no está disponible en el modo de compatibilidad.

Consulte también

Principios básicos de la ingeniería interproyecto (IPE) (Página 11290)

Procedimientos al trabajar con ingeniería interproyecto (IPE) (Página 11298)

12.2.3 Procedimientos al trabajar con ingeniería interproyecto (IPE)

Intercambio de datos de control entre varios proyectos

Con la funcionalidad de ingeniería interproyecto (IPE) se pueden intercambiar los datos de control existentes con un proxy de dispositivo y utilizarlos a continuación en otro proyecto. El proxy de dispositivo le permite un intercambio de datos coherente y cómodo entre varios proyectos sin redundancia de labores de configuración.

Procedimiento para el intercambio de datos de control mediante un archivo IPE

Para el intercambio de datos mediante un archivo IPE siga estos pasos:

1. Con TIA Portal V13 o superior cree un proyecto con todos los datos de control necesarios y un programa de usuario ejecutable.
2. Configure el hardware necesario para el proyecto.
3. Compile el proyecto para asegurar la coherencia de su proyecto.
4. En su proyecto, debajo de la CPU deseada, cree el objeto "Datos del proxy del dispositivo" haciendo clic en el comando "Agregar datos del proxy del dispositivo" de la carpeta "Datos del proxy del dispositivo".
   Resultado: se ha creado el objeto "Datos del proxy del dispositivo".
5. Seleccione el objeto Datos del proxy del dispositivo que desee en el árbol del proyecto, y ábralo en el editor haciendo doble clic.
6. Introduzca en el editor los datos deseados para Datos del proxy del dispositivo bajo "General" o transfiera el ajuste predeterminado. Tiene la opción de modificar el nombre y de añadir un comentario para la exportación de los datos de control.
   Seleccione en el área "Definir contenido" del editor los datos de control que desee exportar como archivo IPE y haga clic en el comando "Exportar datos del proxy del dispositivo".
7. Indique en el cuadro de diálogo siguiente el nombre y la ubicación para el archivo IPE que se ha de crear y haga clic en "Guardar". En cuanto la exportación finalice correctamente, se le mostrará un aviso.
   Resultado: los datos de control seleccionados se han exportado como archivo IPE.
8. En el proyecto de destino, cree un proxy de dispositivo para poder importar los datos de control que hay en el archivo IPE. Para ello, haga clic en el comando "Agregar dispositivo" y seleccione el proxy de dispositivo que desee en el cuadro de diálogo siguiente.
9. Haga clic en el proxy recién creado y, en el menú contextual, seleccione el comando "Inicializar proxy del dispositivo".
10. Seleccione el archivo IPE que se creó antes y que va a utilizarse para la inicialización, y confirme la selección en el cuadro de diálogo siguiente con "Aceptar". A continuación se lleva a cabo la inicialización.
    Resultado: los datos de control del proyecto de origen contenidos en el archivo IPE se transfieren al proxy de dispositivo del proyecto de destino.

En cuanto se haya realizado correctamente el intercambio de datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

Procedimiento para el intercambio de datos de control mediante un archivo de proyecto

Al intercambiar datos de control mediante un archivo de proyecto, pueden haberse predefinido datos del proxy del dispositivo en el proyecto de origen o definirse y seleccionarse desde el proyecto de destino.

Para el intercambio de datos mediante un archivo de proyecto siga estos pasos:

1. Con TIA Portal V13 o superior cree un proyecto con todos los datos de control necesarios y un programa de usuario ejecutable.
2. Configure el hardware necesario para el proyecto.
3. Compile el proyecto para asegurar la coherencia de su proyecto.
4. Abra el proyecto de TIA Portal cuyos datos de control desee importar al proyecto de destino.
5. En su proyecto, debajo de la CPU deseada, cree el objeto "Datos del proxy del dispositivo" haciendo clic en el comando "Agregar datos del proxy del dispositivo" de la carpeta "Datos del proxy del dispositivo".
   Resultado: se ha creado el objeto "Datos del proxy del dispositivo".
6. Introduzca en el editor los datos deseados para Datos del proxy del dispositivo bajo "General" o transfiera el ajuste predeterminado. Tiene la opción de modificar el nombre y de añadir un comentario para los datos del proxy del dispositivo.
7. Seleccione en el editor, en el área "Definir contenido", qué datos de control desea poner a disposición a través del objeto Datos del proxy del dispositivo.
8. Haga clic en "Guardar" para guardar las modificaciones del proyecto.
   Resultado: ha guardado correctamente los datos de control seleccionados en sus datos del proxy del dispositivo.
9. En el proyecto de destino, cree un proxy de dispositivo para poder importar los datos de control que hay en el archivo de proyecto. Para ello, haga clic en el comando "Agregar dispositivo" y seleccione el proxy de dispositivo que desee en el cuadro de diálogo siguiente.
10. Haga clic en el proxy recién creado y, en el menú contextual, seleccione el comando "Inicializar proxy del dispositivo".
11. Seleccione el archivo de proyecto creado anteriormente con el objeto Datos del proxy del dispositivo preparado y que va a utilizarse para la inicialización, y confirme la selección en el cuadro de diálogo siguiente con "Aceptar". A continuación se lleva a cabo la inicialización.
    Resultado: los datos de control del proyecto de origen contenidos en el archivo de proyecto se transfieren al proxy del proyecto de destino.

En cuanto se haya realizado correctamente el intercambio de datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

Procedimiento para actualizar los datos de control ya transferidos

Requisitos para la actualización:

Para actualizar los datos de control ya transferidos mediante el objeto "Datos del proxy del dispositivo" ya creado deben cumplirse los siguientes requisitos:

- Para la actualización se utiliza el mismo proyecto que en la anterior transferencia de los datos de control.
- Para la transferencia de datos de control se utilizan el objeto "Datos del proxy del dispositivo" ya creado.
- No ha modificado la configuración hardware ni las interfaces de comunicación. Está permitido agregar nuevos componentes de hardware.

Procedimiento para la actualización:

1. Abra el proyecto del TIA Portal cuyos datos de control desee actualizar en el proyecto de destino.
2. Haga doble clic en el objeto "Datos del proxy del dispositivo" existente y seleccione en "Contenido" los datos de control que desee para la actualización.
3. Guarde el proyecto y exporte de nuevo el archivo IPE.
4. En el TIA Portal, abra el proyecto de destino que desee.
5. Haga clic en el proxy de dispositivos existente en el proyecto de destino y, en el menú contextual, seleccione el comando "Actualizar proxy del dispositivo".
6. Seleccione el archivo de origen que deba utilizarse para la actualización y confirme la selección.
7. En el cuadro de diálogo siguiente, seleccione en "Dispositivo" la CPU utilizada anteriormente como CPU de origen, desde la que se deben volver a importar los datos de control.
8. En "Datos definidos del proxy del dispositivo", seleccione el objeto que desee y cierre la ventana con "Aceptar".
9. A continuación comienza la actualización y los datos seleccionados del proxy del dispositivo se transfieren desde el proyecto de origen al proxy de dispositivo del proyecto de destino.

Una vez finalizada correctamente la actualización de los datos de control, se puede proseguir con la configuración en el proyecto de destino y, por ejemplo, conectar las variables PLC con las variables HMI. Si cambian los datos de control en el proyecto de origen, estos se pueden actualizar con la frecuencia que convenga.

Consulte también

Principios básicos de la ingeniería interproyecto (IPE) (Página 11290)

Requisitos para la ingeniería interproyecto (IPE) (Página 11293)

Creación de los datos del proxy del dispositivo (Página 3204)

Abrir datos del proxy del dispositivo (Página 3205)

Intercambio de datos de control mediante un archivo IPE (Página 3205)

Uso de las funciones Online y diagnóstico 13

13.1 Mostrar nodos accesibles

Dispositivios accesibles

Los dispositivos accesibles son todos aquellos dispositivos que están conectados con una interfaz de la PG/del PC y que están encendidos. Por tanto, también es posible mostrar dispositivos que, con los productos instalados actualmente, pueden configurarse de forma restringida o que no pueden configurarse en absoluto.

Mostrar los dispositivos accesibles desde una interfaz de la PG/del PC en el árbol del proyecto

Para mostrar los dispositivos accesibles desde una sola interfaz de la PG/del PC, proceda del siguiente modo:

1. Abra la carpeta "Accesos online" en el árbol del proyecto.
2. Haga clic en el símbolo de flecha que aparece a la izquierda de la interfaz para mostrar todos los objetos subordinados.
3. Haga doble clic en el comando "Actualizar dispositivos accesibles" debajo de la interfaz.
   En el árbol del proyecto se visualizan todos los dispositivos accesibles desde esa interfaz. Con un número elevado de dispositivos enlazados, el proceso de actualización puede requerir algo de tiempo. El progreso de la actualización puede observarse en la barra de estado. Si encuentra el dispositivo deseado antes de que finalice la actualización, puede cancelar la actualización de los dispositivos accesibles. Para ello, haga clic en la cruz situada a la derecha de la barra de progreso.

Mostrar los dispositivos accesibles en una lista

Para mostrar en una lista sinóptica los dispositivos accesibles desde todas las interfaces disponibles, proceda del siguiente modo:

1. Elija el comando "Dispositivos accesibles" del menú "Online".
   Se abre el cuadro de diálogo "Dispositivos accesibles".
2. Seleccione el tipo de interfaz en la lista desplegable "Tipo de interfaz PG/PC". En función de la selección, en la lista desplegable "Interfaz PG/PC" se mostrarán únicamente las interfaces de la PG/el PC que concuerdan con el tipo de interfaz seleccionado.
3. Elija la interfaz deseada de la PG/el PC en la lista desplegable "Interfaz PG/PC", p. ej. una tarjeta Industrial Ethernet.
   Si no se puede acceder a ningún dispositivo desde una interfaz, la línea de conexión entre la PG/el PC y el dispositivo se representará interrumpida. Si hay dispositivos accesibles, la línea de conexión se representará cerrada y todos los dispositivos accesibles desde la interfaz seleccionada de la PG/del PC se visualizarán en una lista.

4. Si, entretanto, se ha conectado un dispositivo nuevo, haga clic en el botón "Actualizar" para actualizar la lista de dispositivos accesibles.
5. Para saltar a un dispositivo en el árbol del proyecto, selecciónelo de la lista de dispositivos accesibles y haga clic en el botón "Mostrar".
   La interfaz a la que está conectado el dispositivo seleccionado se resalta en el árbol del proyecto.

Visualizar información adicional sobre los dispositivos accesibles en el árbol del proyecto

Para visualizar información adicional sobre los distintos dispositivos accesibles en el árbol del proyecto, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic en el símbolo de flecha que aparece a la izquierda de uno de los dispositivos accesibles.
   Para los dispositivos conocidos, se visualizan todos los datos disponibles online, p. ej. los bloques o datos de sistema. Los objetos que no se pueden editar directamente aquí se muestran en gris. Si existen otras posibilidades de edición para un dispositivo, como p. ej. la carga a través del menú contextual, el dispositivo se muestra en negro.

Consulte también

Modificación online de la configuración del dispositivo (Página 11302)

13.2 Modificación online de la configuración del dispositivo

Algunos dispositivos, preferentemente en configuraciones hardware reducidas, se pueden parametrizar directamente online. Para ello hay que crear un proyecto y disponer de datos offline. De ese modo es posible modificar la configuración del dispositivo rápida y fácilmente. No es necesario compilar la configuración hardware ni realizar una descarga. En función del dispositivo, todos los cambios serán efectivos de forma inmediata o se escribirán en el dispositivo tras una confirmación.

Requisitos

- El dispositivo debe soportar la parametrización online. Para saber si los dispositivos soportan esta función, consulte el manual de producto correspondiente.
- El dispositivo debe estar conectado a la programadora o el PC y estar disponible en la lista de dispositivos accesibles.

Procedimiento

Para modificar online la configuración del dispositivo, proceda del siguiente modo:

1. Visualice los dispositivos accesibles en la interfaz a través de la cual está conectado el dispositivo. Para saber cómo visualizar los dispositivos accesibles, consulte el capítulo anterior "Visualización de los dispositivos accesibles (Página 11297)".
2. Abra el dispositivo para visualizar los elementos subordinados.

3. Haga doble clic en la entrada "Parametrizar dispositivo".
   En el área de trabajo se abre una página de configuración para el dispositivo.
4. Realice todos los ajustes necesarios.
   En algunos dispositivos los cambios en los ajustes son efectivos de forma inmediata.
5. Opcionalmente y en función del dispositivo: Haga clic en el botón "Transferir al dispositivo".
   Los ajustes se transfieren al dispositivo.

13.3 Conectar dispositivos online

13.3.1 Información general sobre el modo online

Modo online

En modo online existe una conexión online entre la PG/el PC y uno o varios dispositivos.

En las siguientes actividades se necesita una conexión online entre la PG/el PC y el dispositivo:

- Comprobar programas de usuario
- Visualizar y cambiar el estado operativo de la CPU
- Visualizar y ajustar la fecha y hora de la CPU
- Visualizar la información del módulo
- Comparar bloques
- Diagnosticar el hardware

Para establecer una conexión online, la PG/el PC y el dispositivo deben estar interconectados físicamente o a través de una conexión remota. Alternativamente, algunos dispositivos soportan un modo de simulación. En este caso, se simula una conexión con el dispositivo a través de la interfaz virtual PLCSIM.

Una vez establecida la conexión, es posible acceder a los datos del dispositivo en la vista online y de diagnóstico o desde la Task Card "Herramientas online". Además, el estado online actual de un dispositivo se representa mediante un símbolo situado a la derecha de éste en el árbol del proyecto. El significado de los distintos símbolos de estado se explica en los respectivos tooltips.

Nota

Algunas funciones online dependen del volumen de software instalado, o bien de si un proyecto está abierto.

Modo de espera o estado de reposo de la PG/el PC

Si la PG/el PC pasa al modo de espera o al estado de reposo en una conexión online existente, se deshacen todas las conexiones online. Una vez la PG/el PC sale del modo de reposo, las conexiones online no se restablecen automáticamente.

En este contexto, tenga en cuenta que el corte repentino de la conexión online puede provocar la pérdida de datos o bien un dispositivo conectado puede interrumpir la ejecución del programa.

Realizar test de parpadeo LED

En muchos cuadros de diálogo online es posible realizar un test de parpadeo LED si el dispositivo conectado online soporta esta función. Si activa la casilla de verificación "Parpadear LED", parpadeará el LED del dispositivo seleccionado en ese momento. Esta función resulta útil por ejemplo cuando no se está seguro del dispositivo dentro de la estructura de hardware que corresponde al nodo que se ha seleccionado en el software en la actualidad.

Tenga en cuenta las indicaciones y las posibles limitaciones para el test de parpadeo LED que se indiquen en la documentación correspondiente del dispositivo.

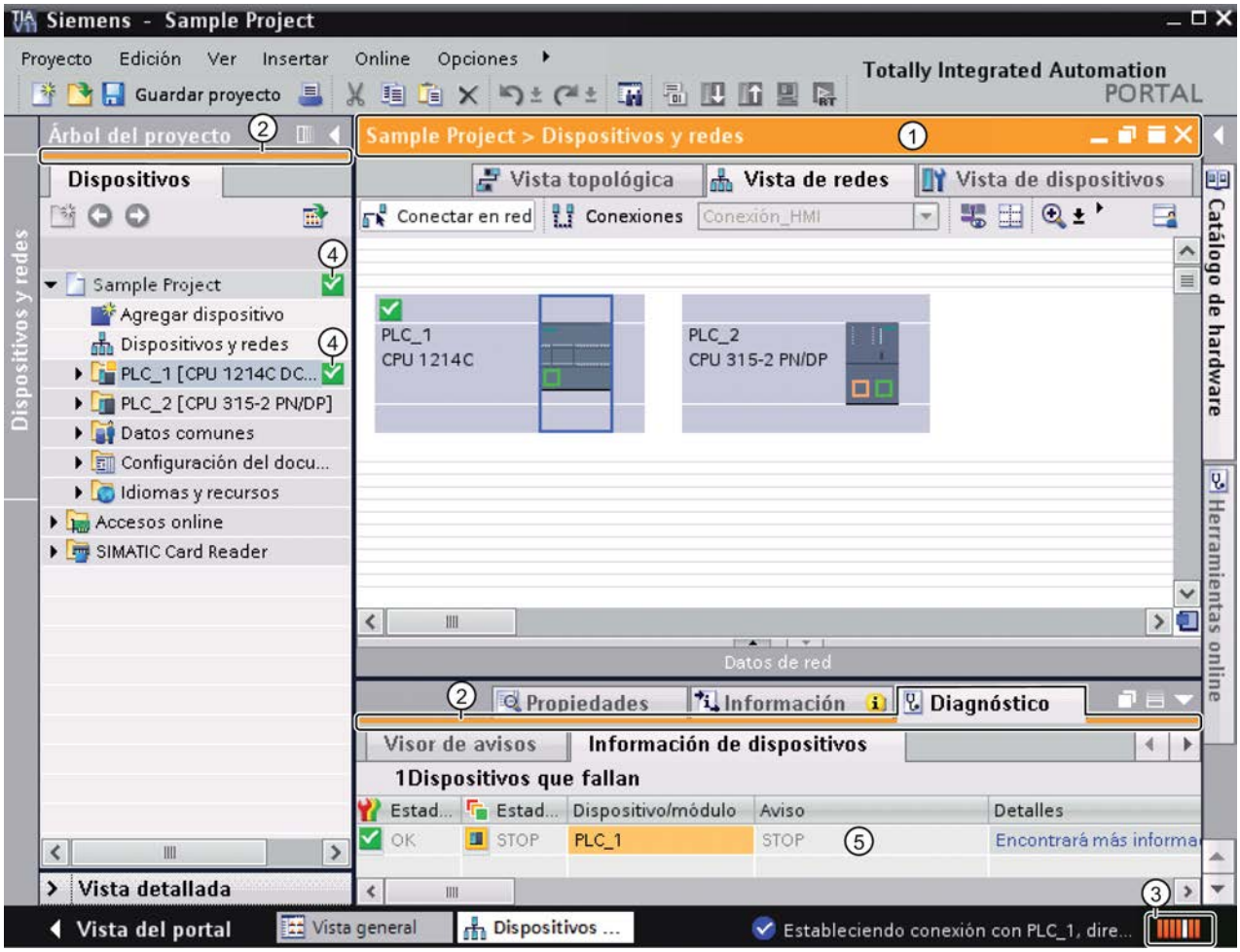
Consulte también

Vista en el modo online (Página 11305)

13.3.2 Vista en el modo online

Indicaciones online

Una vez establecida la conexión online, el aspecto de la interfaz de usuario cambia. Si un dispositivo no está accesible, se indica simbólicamente. La figura siguiente muestra un dispositivo conectado online y la interfaz de usuario correspondiente:



① La barra de título de la ventana activa se muestra entonces sobre un fondo naranja en cuanto como mínimo uno de los dispositivos visualizados en el editor se ha conectado online correctamente. Si uno o varios dispositivos no están accesibles, en la barra de título del editor se muestra un símbolo de conexión interrumpida.

② Las barras de título de las ventanas inactivas de la estación correspondiente aparecen con una raya naranja en su borde inferior.

③ En el margen derecho de la barra de estado aparece una barra naranja animada. Si la conexión se ha establecido pero es defectuosa, en lugar de la barra se muestra un símbolo que indica la interrupción de la conexión. En el apartado "Diagnóstico" de la ventana de inspección obtendrá más información relacionada el error.

④ En el árbol del proyecto se visualizan símbolos relativos al estado operativo y al diagnóstico junto a las estaciones conectadas online y sus objetos subordinados. Asimismo, se ejecuta una comparación automática entre el estado online y offline. Las diferencias entre los objetos online y offline también se marcan con símbolos.

⑤ El área "Diagnóstico > Información del dispositivo" aparece en primer plano en la ventana de inspección.

Corte de la conexión online

El modo online y su visualización se mantienen mientras haya como mínimo un dispositivo conectado online. Si la conexión online se corta en uno o varios dispositivos, el TIA Portal se mantiene en modo online. La representación del TIA Portal no cambia al modo offline hasta que ya no queda ninguna conexión online con los dispositivos.

Consulte también

Información general sobre el modo online (Página 11299)

Principios básicos de la comparación de datos de proyecto (Página 373)

13.3.3 Establecer y deshacer una conexión online

Requisitos

Hay instalada por lo menos una interfaz PG/PC que, a su vez, está conectada físicamente con un dispositivo (p. ej. con un cable Ethernet). Otra alternativa consiste en establecer una conexión online virtual mediante PLCSIM.

Establecer conexión online

Para establecer una conexión online, proceda del siguiente modo:

1. En el árbol del proyecto, seleccione uno o varios dispositivos con los que se deba establecer una conexión online.
2. Elija el comando "Establecer conexión online" del menú "Online".
   Si el dispositivo ya había estado conectado con una interfaz PG/PC en particular, la conexión online se establecerá automáticamente con esa misma interfaz PG/PC. En este caso no es necesario tener en cuenta los pasos siguientes. Si hasta ahora no se había establecido ninguna conexión online, aparecerá el cuadro de diálogo "Establecer conexión online".
3. Seleccione el tipo de interfaz en la lista desplegable "Tipo de interfaz PG/PC". En función de la selección, en la lista desplegable "Interfaz PG/PC" se mostrarán únicamente las interfaces de la PG/el PC que concuerdan con el tipo de interfaz seleccionado.
4. Elija la interfaz deseada de la PG/el PC en la lista desplegable "Interfaz PG/PC", p. ej. una tarjeta Industrial Ethernet.

5. En la lista desplegable "Conexión con subred", seleccione la interfaz a través de la cual se conecta el dispositivo con la PG o el PC. En este caso se establece una conexión directa con el dispositivo, sin nodos de red (por ejemplo, un switch) de por medio. Como alternativa, seleccione la subred adecuada para la conexión con la PG o el PC si es posible acceder al dispositivo a través de un nodo de red. Si desconoce cómo está conectado el dispositivo con la PG o el PC, seleccione la entrada "Probar todas las interfaces".
   Si se ha seleccionado una subred MPI o PROFIBUS, en este momento se aplicarán a la interfaz PG/PC los parámetros configurados para el bus.
6. Si el dispositivo deseado es accesible mediante un gateway, elija en la lista desplegable "Primer gateway" el gateway que interconecta las subredes en cuestión.
   Si no se puede acceder a ningún dispositivo desde la interfaz, la línea de conexión entre la PG/el PC y el dispositivo se representará interrumpida. Si hay dispositivos accesibles, la línea de conexión se representará cerrada y todos los dispositivos accesibles desde la interfaz seleccionada de la PG o del PC se visualizarán en la lista "Dispositivos compatibles en la subred de destino".
7. Opcionalmente: Haga clic en el botón "Actualizar" para actualizar la lista "Dispositivos compatibles en la subred de destino".
8. Opcionalmente: En el lado izquierdo del gráfico, active la casilla de verificación "Parpadear LED" para realizar un test de intermitencia de los LED. De esta manera se puede comprobar si se ha seleccionado el dispositivo correcto. No todos los dispositivos soportan el test de intermitencia de los LEDs.
9. Seleccione el dispositivo en la tabla "Dispositivos compatibles en la subred de destino" y confirme la selección con "Establecer conexión online".
   Se establece la conexión online con el dispositivo de destino seleccionado.

Resultado

Una vez establecida la conexión online, las barras de título de los editores aparecen en color naranja. Además, en la barra de título de un editor y en la barra de estado se visualizan sendas barras de progreso en color naranja. Los símbolos de estado muestran la diferencia entre los objetos disponibles online y offline en el árbol del proyecto.

La vía de conexión se guarda para intentos de conexión futuros. En este caso ya no es necesario abrir el cuadro de diálogo "Establecer conexión online", a menos que se desee determinar una nueva vía de conexión.

Nota

Si no se visualiza ningún dispositivo accesible, seleccione un acceso de red diferente para la interfaz PG/PC o compruebe la configuración de la interfaz.

Deshacer la conexión online

Para deshacer una conexión online existente, proceda del siguiente modo:

1. En el árbol del proyecto, seleccione el dispositivo del que desea deshacer la conexión online.
2. Elija el comando "Deshacer conexión online" del menú "Online".

Consulte también

Conexión online con varios dispositivos (Página 11308)

Vista en el modo online (Página 11301)

Asignar una dirección IP temporal (Página 11320)

Influencia de los derechos de usuario (Página 325)

13.3.4 Conexión online con varios dispositivos

Es posible establecer una conexión online con varios dispositivos a la vez sin tener que seleccionar previamente los diferentes dispositivos en el árbol del proyecto o en la vista de redes.

Requisitos

- No puede haber ningún dispositivo seleccionado.
- Hay instalada por lo menos una interfaz PG/PC que, a su vez, está conectada físicamente con un dispositivo (p. ej. con un cable Ethernet). Otra alternativa consiste en establecer una conexión online virtual mediante PLCSIM, o bien crear una conexión remota.

Procedimiento

Para establecer una conexión online con varios dispositivos a la vez, proceda del siguiente modo:

1. Seleccione el proyecto en el árbol del proyecto.
2. Elija el comando "Establecer conexión online" del menú "Online".
   Se abrirá el cuadro de diálogo "Seleccionar dispositivo para establecer la conexión online" con una tabla de todos los dispositivos disponibles.
3. En la columna "Conexión online" seleccione los dispositivos con los que desea establecer una conexión online.
4. Haga clic en el botón "Establecer conexión online".

Resultado

Se establece una conexión con todos los dispositivos seleccionados sin más consulta, siempre que previamente haya existido alguna conexión con los dispositivos seleccionados. Si hasta ahora no se había establecido ninguna conexión online, aparecerá el cuadro de diálogo "Establecer conexión online". En este caso, configure primero la conexión online tal como se describe en el capítulo "Establecer y deshacer una conexión online (Página 11302)".

Consulte también

Establecer y deshacer una conexión online (Página 11302)

Asignar una dirección IP temporal (Página 11320)

13.3.5 Deshacer conexiones online de varios dispositivos

Es posible deshacer a la vez las conexiones online existentes con varios dispositivos sin tener que seleccionar previamente los diferentes dispositivos en el árbol del proyecto o en la vista de redes.

Requisitos

- No hay ningún dispositivo seleccionado.
- Existe una conexión online como mínimo para un dispositivo.

Procedimiento

Para deshacer simultáneamente las conexiones online con varios dispositivos, proceda del siguiente modo:

1. Elija el comando "Deshacer conexión online" del menú "Online".
   Se abre el diálogo "Seleccionar dispositivos" con una tabla de todos los dispositivos disponibles.
2. En la columna "Deshacer conexión online", seleccione los dispositivos para los que desea deshacer la conexión online.
3. Haga clic en el botón "Deshacer conexión online".

Resultado

Se deshace la conexión online entre todos los dispositivos seleccionados.

13.4 Crear una copia de seguridad del software y la configuración de hardware de un dispositivo

13.4.1 Crear una copia de seguridad de un dispositivo

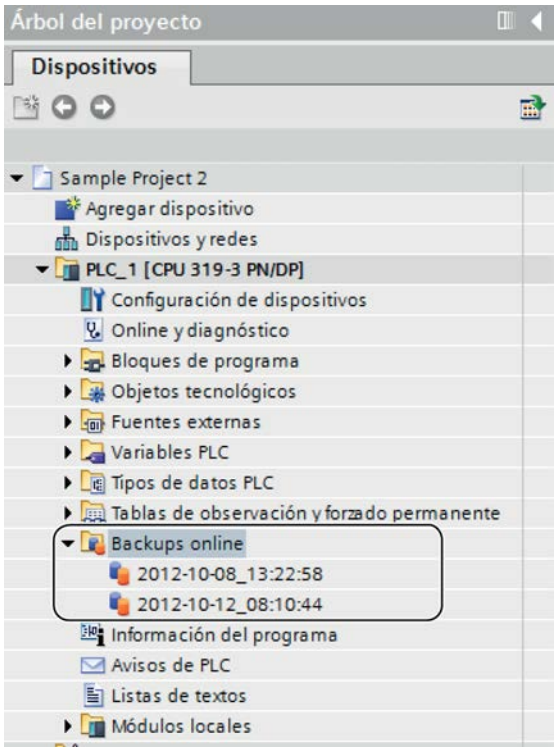
Copia de seguridad de la configuración de software y hardware de una CPU S7-300/400

Si ya ha cargado una configuración en una CPU S7-300/400, puede ser útil hacer una copia de seguridad de esta. Posiblemente haya modificado la configuración y desee probar la nueva configuración. Antes de cargar la nueva configuración en la CPU, cree una copia de seguridad del estado actual del dispositivo, que le permitirá restablecer la configuración actual más adelante. La copia de seguridad se realiza con los valores actuales de la CPU. En las CPU S7-400 con función de seguridad positiva se hacen copias de seguridad de los valores predeterminados.

Puede crear tantas copias de seguridad como desee para guardar diferentes configuraciones para una CPU. El nombre de las copias de seguridad incluye el nombre de la CPU y la hora

y fecha de la copia. La copia de seguridad está en la carpeta "Backups online" de la CPU en el árbol del proyecto.

La figura siguiente muestra una CPU S7-319 de la que se han realizado dos copias de seguridad:



Consulte también

Restablecer el software y la configuración de hardware de un dispositivo (Página 11311)

Copia de seguridad de la configuración de un dispositivo (Página 11310)

Información general acerca de la carga (Página 364)

13.4.2 Copia de seguridad de la configuración de un dispositivo

Puede hacer una copia de seguridad de la configuración de una CPU S7-300/400 en el TIA Portal. Esto permite cargar y probar sin riesgos una configuración nueva en un dispositivo. En caso necesario, es posible restablecer la configuración original de la CPU.

Requisitos

- La CPU ya debe estar creada en el proyecto.
- Debe tratarse de una CPU S7-300/400.
- Es necesario que la CPU esté conectada online. Si todavía no hay ninguna conexión online, al crear la copia de seguridad se establecerá una conexión online.

Procedimiento

Para crear una copia de seguridad de la configuración actual de una CPU, proceda de la siguiente manera:

1. Seleccione la CPU en el árbol del proyecto.
2. Elija el comando "Cargar backup del dispositivo online" del menú "Online".

Resultado:

Se crea una copia de seguridad de toda la configuración de hardware y del software. La copia de seguridad se almacena en la carpeta "Nombre de la CPU > Backups online" del árbol del proyecto. La copia de seguridad lleva el nombre de la CPU con la hora y la fecha de la copia de seguridad. Puede cambiar el nombre de la copia de seguridad; sin embargo, no se puede llevar a cabo ninguna modificación en los contenidos de dicha copia.

Consulte también

Establecer y deshacer una conexión online (Página 11302)

Restablecer el software y la configuración de hardware de un dispositivo (Página 11311)

Crear una copia de seguridad de un dispositivo (Página 11305)

13.4.3 Restablecer el software y la configuración de hardware de un dispositivo

Si previamente ha realizado una copia de seguridad de la configuración de un dispositivo, es posible volver a transferir dicha copia al dispositivo. En este caso, en el dispositivo se restablece la configuración guardada.

Requisitos

El dispositivo debe estar configurado y debe haberse guardado una copia de seguridad del dispositivo en el proyecto.

Procedimiento

Para restablecer en un dispositivo un estado anterior de software y hardware, proceda del siguiente modo:

1. Abra la carpeta del dispositivo en el árbol del proyecto para visualizar los objetos subordinados.
2. Abra la carpeta "Backups online".
3. Seleccione la copia de seguridad que desee restablecer.

4. En el menú "Online" seleccione el comando "Cargar en dispositivo".
   - Si ya ha establecido una conexión online, aparecerá el cuadro de diálogo "Cargar vista preliminar". En este cuadro de diálogo se visualizan avisos y se proponen las acciones necesarias para la carga.
   - Si hasta ahora no ha establecido ninguna conexión online, aparecerá el cuadro de diálogo "Carga avanzada" y primero deberá seleccionar las interfaces a través de las que va a establecer la conexión online con el dispositivo.
     Consulte también: Establecer y deshacer una conexión online (Página 11302)
5. Verifique los mensajes en el cuadro de diálogo "Cargar vista preliminar" y, dado el caso, active las acciones en la columna "Acción".

   Nota

   La realización de las acciones propuestas con la instalación en marcha puede causar graves daños materiales y personales en caso de fallos de funcionamiento o errores del programa.

6. El botón "Cargar" se activará tan pronto como sea posible la carga.
7. Haga clic en el botón "Cargar".
   La copia de seguridad se transfiere al dispositivo y se restablece el estado guardado. A continuación se abre el cuadro de diálogo "Cargar resultados". Este cuadro de diálogo permite comprobar si la carga se ha realizado correctamente y, en caso necesario, seleccionar acciones adicionales.
8. Haga clic en el botón "Finalizar".

Consulte también

Crear una copia de seguridad de un dispositivo (Página 11305)

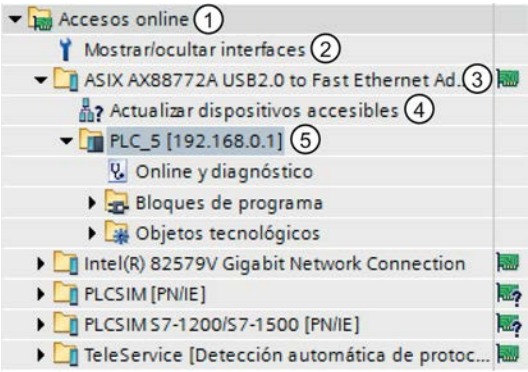
13.5 Configurar interfaces PG/PC

13.5.1 Accesos online

Acceso online del proyecto

En la carpeta "Accesos online" del árbol del proyecto están todas las interfaces activas de la PG/el PC. Todos los símbolos de interfaz proporcionan información sobre el estado correspondiente. Se pueden visualizar los dispositivos accesibles y, desde el menú contextual, las propiedades de una interfaz, que además pueden modificarse.

La siguiente imagen muestra la carpeta "Accesos online" en el árbol del proyecto:



① Carpeta "Accesos online" en el árbol del proyecto

En la carpeta "Accesos online" se pueden visualizar todas las interfaces instaladas en la PG/el PC.

② Mostrar/ocultar interfaces

La función "Mostrar/ocultar interfaces" permite mostrar y ocultar diferentes interfaces.

③ Indicador de estado de las interfaces

El estado actual de una interfaz se indica mediante un símbolo situado a la derecha del nombre. El significado de los símbolos se explica en los respectivos tooltips.

④ Actualización de la lista de los dispositivos accesibles.

Esta función está disponible para cualquier interfaz de hardware de la PG/del PC. Las interfaces de software, como por ejemplo, una conexión remota, no disponen de esta función.

⑤ Equipos que están conectados con la PG/el PC a través de la interfaz correspondiente

El tipo de equipo correspondiente y también su estado se muestran con ayuda del símbolo que aparece delante.

Mostrar o actualizar los dispositivos accesibles

Las siguientes opciones están disponibles para mostrar los dispositivos accesibles online en la PG/el PC:

- Mostrar los dispositivos accesibles desde una sola interfaz de la PG/del PC en el árbol del proyecto. Además, en el árbol del proyecto es posible visualizar información adicional sobre los distintos dispositivos accesibles.
- Mostrar en una lista los dispositivos accesibles de todas las interfaces.

Consulte también: Mostrar nodos accesibles

Vista general de los símbolos de los dispositivos accesibles

Los dispositivos accesibles se marcan con un símbolo conforme a su tipo y a su estado. A continuación se muestra una vista general de todos los símbolos y su significado.

	Símbolo para módulos no identificables Este símbolo se mostrará siempre que la identificación de un módulo aún no haya finalizado, o bien cuando la identificación de un módulo no se haya realizado correctamente, por ejemplo, porque no se han podido leer los datos online necesarios.
	Símbolo de los siguientes tipos de dispositivos: • PLC • Dispositivos SIMOCODE pro • Enlaces IE/PB • CP de sistemas PC • Módulos de cabecera SCALANCE • CP S7-300 y S7-400 • Dispositivos PROFINET IO y CP PROFINET • Módulos SCALANCE y gateways que no han podido ser identificados
	Dispositivos PROFINET IO, encóders, aparellaje, sensores y sistemas de identificación que han sido sustituidos por dispositivos similares porque dichos dispositivos no pudieron ser identificados
	Símbolo para los siguientes tipos de dispositivos: • Dispositivos HMI • Dispositivos PROFINET IO del tipo HMI, si estos no pudieron ser identificados y, por eso, fueron sustituidos por un dispositivo similar
	Dispositivos PROFINET IO del tipo accionamiento que no pudieron ser identificados y, por eso, fueron sustituidos por un dispositivo similar
	Dispositivos PROFINET IO del tipo Development Kit y componentes de red que no pudieron ser identificados y, por eso, fueron sustituidos por un dispositivo similar
	Dispositivos PROFINET IO del tipo TeleService Adapter que no pudieron ser identificados y, por eso, fueron sustituidos por un dispositivo similar

Consulte también

Mostrar y modificar las propiedades de una interfaz (Página 11316)

13.5.2 Principios básicos para parametrizar la interfaz PG/PC

Posibilidades de conexión con sistemas de destino

Si los dispositivos del proyecto están conectados a través de subredes distintas, hay que asignar a la interfaz PG/PC el acceso de red adecuado para poder establecer conexiones online con los sistemas de destino. Se admiten automáticamente las siguientes interfaces:

- MPI
- PROFIBUS
- Industrial Ethernet (ISO y TCP/IP)

Las interfaces ofrecen según el tipo distintas posibilidades de parametrización. En los capítulos siguientes verá las posibilidades de parametrización que existen.

Nota

Tenga en cuenta que los cambios efectuados en los parámetros de interfaz influyen directamente en el sistema operativo y en la PG/el PC. Por tanto, algunos parámetros sólo pueden modificarse si se dispone de suficientes derechos de usuario.

Consulte también

Parametrizar la interfaz Industrial Ethernet (Página 11317)

Parametrizar la interfaz MPI y PROFIBUS (Página 11321)

13.5.3 Mostrar u ocultar interfaces

Para tener una visión general más clara, existe la posibilidad de ocultar diferentes interfaces de la programadora o el PC en el árbol del proyecto y volver a mostrarlas en caso necesario.

Procedimiento

Para mostrar u ocultar interfaces individuales, proceda del siguiente modo:

1. Abra la carpeta "Accesos online" en el árbol del proyecto.
2. Haga doble clic en el icono "Mostrar/ocultar interfaces".
   Se abre el cuadro de diálogo "Mostrar/ocultar interfaces".
3. Active o desactive las interfaces deseadas en la columna "Mostrar en el árbol del proyecto".
4. Haga clic en el botón "Aplicar".
   Los cambios se aplicarán. La vista de las interfaces se actualiza en la carpeta "Accesos online".

13.5.4 Mostrar y modificar las propiedades de una interfaz

Introducción

Es posible visualizar las propiedades de toda interfaz y modificarlas en caso necesario (p. ej. tipo de red, dirección y estado).

Procedimiento

Para abrir las propiedades, proceda del siguiente modo:

1. En el árbol del proyecto, haga clic con el botón derecho del ratón en la interfaz deseada bajo "Accesos online".
2. Elija el comando "Propiedades" del menú contextual.
   Se abre el diálogo con las propiedades de la interfaz. La navegación local se encuentra en el lado izquierdo del diálogo. En las distintas entradas de la navegación local es posible visualizar los parámetros actuales y modificarlos en caso necesario.

13.5.5 Agregar interfaces

Después de la instalación del TIA Portal, tiene la posibilidad de instalar otras interfaces.

Procedimiento

Para instalar una interfaz a posteriori y agregarla al TIA Portal, proceda del siguiente modo:

1. Instale o actualice los drivers en el sistema operativo, después de haber instalado las interfaces en el lado del hardware.
2. Cierre el TIA Portal, en caso de que aún esté abierto.
3. Abra el Panel de control de Windows.
4. Abra la entrada "Ajustar interface PG/PC" en el Panel de control.
   Se abre el diálogo "Ajustar interface PG/PC".
5. Realice cambios, si es necesario, en la configuración de interfaces y confírmelos con "Aceptar". Haga clic en cualquier caso en el botón "Aceptar", aunque no haya realizado ningún cambio.
6. Reinicie el TIA Portal.

Resultado

La nueva interfaz instalada se muestra ahora en el árbol del proyecto, en la carpeta "Accesos online".

13.5.6 Parametrizar la interfaz Ethernet

13.5.6.1 Parametrizar la interfaz Industrial Ethernet

Posibilidades para parametrizar la interfaz Industrial Ethernet

Para parametrizar la interfaz Industrial Ethernet existen las siguientes posibilidades:

- Parámetros que predetermina el sistema operativo
  La interfaz Industrial Ethernet dispone de parámetros que se ajustan en el sistema operativo y que son válidos para todos los dispositivos conectados. Estos parámetros sólo se visualizan, pero pueden modificarse en la configuración de red del sistema operativo.
- Parámetros ajustables en el software

Nota

Tenga en cuenta que los cambios efectuados en los parámetros de interfaz influyen directamente en el sistema operativo y en la PG/el PC. Por tanto, algunos parámetros sólo pueden modificarse si se dispone de suficientes derechos de usuario.

Parámetros de la interfaz Industrial Ethernet

La tabla siguiente ofrece una vista general de los parámetros de la interfaz Industrial Ethernet predeterminados por el sistema operativo y de los que el usuario puede modificar.

Parámetros no modificables	Parámetros ajustables
Dirección MAC	Fast Acknowledge en el acceso IE-PG y para TCP/IP
Servidor DHCP activado/desactivado	Timeout en el acceso IE-PG y para TCP/IP
APIPA activado/desactivado	LLDP
Dirección IP	Direcciones IP temporales adicionales para la tarjeta de red
Máscara de subred	-
Direcciones DNS	-
Direcciones DHCP	-

Consulte también

Principios básicos para parametrizar la interfaz PG/PC (Página 11311)

Visualizar los parámetros del sistema operativo (Página 11318)

Conectar la interfaz PG/PC con una subred (Página 11318)

Parametrizar la interfaz Ethernet (Página 11319)

Asignar una dirección IP temporal (Página 11320)

Gestionar direcciones IP temporales (Página 11320)

Influencia de los derechos de usuario (Página 325)

13.5.6.2 Visualizar los parámetros del sistema operativo

La interfaz Ethernet forma parte del sistema operativo. Por tanto, todos los parámetros de la tarjeta de red pueden modificarse en la configuración de red del sistema operativo.

Es posible visualizar los siguientes parámetros en el software:

- Dirección física de la tarjeta de red
- Asignación de la dirección IP por un servidor DHCP activada o desactivada
- Asignación de una dirección IP privada por el sistema operativo activada o desactivada
- Dirección IP fija asignada actualmente
- Máscara de subred asignada
- Direcciones DNS
- Direcciones DHCP

Para obtener información sobre cómo modificar los parámetros, consulte la documentación del sistema operativo o de la tarjeta de red.

Visualizar los parámetros actuales de la interfaz Ethernet

Para visualizar los parámetros actuales de la interfaz Ethernet, proceda del siguiente modo:

1. Seleccione la interfaz Ethernet en "Accesos online" en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el diálogo de configuración de la interfaz.
3. Elija "Configuraciones > Industrial Ethernet" en la navegación local.

Consulte también

Parametrizar la interfaz Ethernet (Página 11319)

13.5.6.3 Conectar la interfaz PG/PC con una subred

Si ha creado varias subredes puede determinar con cuál de ellas debe conectarse la interfaz Ethernet.

Procedimiento

Para seleccionar la subred con la que está conectada la interfaz Ethernet, proceda del siguiente modo:

1. Seleccione la interfaz Ethernet en "Accesos online" en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el diálogo de configuración de la interfaz.

3. En "General > Asignación", seleccione la subred con la que desea conectar la interfaz Ethernet de la PG/del PC de la lista desplegable "Conexión con subred".
4. Cierre el diálogo con "Aceptar".

13.5.6.4 Parametrizar la interfaz Ethernet

Algunos de los parámetros relacionados con el protocolo de red pueden adaptarse directamente en el software.

Requisitos

Dispone de derechos de usuario suficientes.

Consulte también: Influencia de los derechos de usuario (Página 325).

Procedimiento

Para modificar los parámetros relacionados con el protocolo de red, proceda del siguiente modo:

1. Seleccione la interfaz Ethernet en "Accesos online" en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el diálogo de configuración de la interfaz.
3. Seleccione "Configuraciones > Acceso IE-PG" para adaptar los ajustes de protocolo relativos a la administración de red.
   - Active la casilla de verificación "Fast Acknowledge" para alcanzar tiempos de reacción más rápidos para paquetes de red pequeños.
   - En la lista desplegable "Timeout", seleccione el tiempo máximo que puede transcurrir hasta la detección de un nodo.
4. Para activar el protocolo LLDP para determinar con mayor precisión la topología de red, active la casilla de verificación "LLDP activo" en "Configuraciones > LLDP".
5. Seleccione "Configuraciones > TCP/IP" para adaptar el protocolo TCP/IP para el tráfico de red durante el servicio.
   - Active la casilla de verificación "Fast Acknowledge" para alcanzar tiempos de reacción más rápidos para paquetes de red pequeños.
   - En la lista desplegable "Timeout", seleccione el tiempo máximo que puede transcurrir en la comunicación con un nodo hasta que se produzca un timeout.

Consulte también

Influencia de los derechos de usuario (Página 325)

Visualizar los parámetros del sistema operativo (Página 11314)

13.5.6.5 Asignar una dirección IP temporal

Agregar una dirección IP temporal

Si la dirección IP de un dispositivo se encuentra en una subred diferente a la de la tarjeta de red, es preciso asignar a esta última una dirección IP adicional con la misma dirección de subred que la del dispositivo. Sólo así se podrá establecer la comunicación entre el dispositivo y la PG/el PC.

La asignación de una dirección IP temporal adicional se propone automáticamente si se desea realizar una acción online y la dirección IP actual de la PG/del PC no se encuentra aún en la subred correcta.

La dirección IP asignada temporalmente es válida hasta el siguiente rearranque de la PG/del PC o hasta que se borre manualmente.

Nota

Para poder asignar una dirección IP temporal es preciso disponer de derechos de usuario suficientes.

Consulte también: Influencia de los derechos de usuario (Página 325)

Consulte también

Gestionar direcciones IP temporales (Página 11320)

13.5.6.6 Gestionar direcciones IP temporales

Si la dirección IP de un dispositivo se encuentra en una subred diferente a la de la dirección IP fija asignada actualmente a la tarjeta de red, se asigna temporalmente a ésta última una dirección IP adecuada de la misma subred que el dispositivo.

Todas las direcciones asignadas temporalmente se pueden visualizar y borrar. Tenga en cuenta que las direcciones IP asignadas manualmente en el sistema operativo no se muestran en el TIA Portal.

Requisitos

Para borrar se requieren derechos de usuario suficientes.

Procedimiento

Para visualizar y borrar direcciones asignadas temporalmente, proceda del siguiente modo:

1. Seleccione la interfaz Ethernet en "Accesos online" en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el diálogo de configuración de la interfaz.

3. Elija "Configuraciones > Acceso IE-PG".
   Aparece una tabla con las direcciones IP asignadas.
4. Haga clic en el botón de comando "Borrar direcciones IP específicas del proyecto" para borrar todas las direcciones IP de una sola vez.

Consulte también

Influencia de los derechos de usuario (Página 325)

13.5.6.7 Restablecer la configuración TCP/IP

Si se han realizado cambios en los ajustes del protocolo TCP/IP, existe la posibilidad de restablecer los valores predeterminados.

Procedimiento

Para restablecer los valores predeterminados de la configuración TCP/IP, proceda del siguiente modo:

1. Seleccione la interfaz Ethernet en "Accesos online" en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el cuadro de diálogo de configuración de la interfaz.
3. Elija "Configuraciones > TCP/IP".
4. Haga clic en el botón "Estándar" para restablecer todos los ajustes.

13.5.7 Parametrizar la interfaz MPI y PROFIBUS

13.5.7.1 Parametrizar la interfaz MPI y PROFIBUS

Posibilidades para parametrizar la interfaz MPI y PROFIBUS

Para parametrizar la interfaz MPI y PROFIBUS existen las posibilidades siguientes:

- Configuración automática: es posible detectar automáticamente si un dispositivo está conectado con la interfaz PG/PC a través de PROFIBUS o MPI.
- Selección de una configuración predeterminada para PROFIBUS o MPI que puede adaptarse posteriormente.

Posibilidades de parametrización para MPI y PROFIBUS en relación al dispositivo y la red

En las interfaces MPI y PROFIBUS pueden ajustarse parámetros relacionados con el dispositivo y con la red. Los parámetros relacionados con el dispositivo son los ajustes locales de la interfaz. En cambio, los parámetros relacionados con la red deben coincidir en todos los dispositivos.

Parámetros modificables en la interfaz MPI

Es posible ajustar los siguientes parámetros predeterminados para la interfaz MPI:

Parámetros relacionados con el dispositivo	Parámetros relacionados con la red
Es el único maestro	Dirección más alta
Dirección propia	Velocidad de transferencia
Timeout	

Parámetros modificables en la interfaz PROFIBUS

Es posible ajustar los siguientes parámetros predeterminados para la interfaz PROFIBUS:

Parámetros relacionados con el dispositivo	Parámetros relacionados con la red
Es el único maestro	Dirección más alta
Dirección propia	Velocidad de transferencia
Timeout	Perfil
	Parámetros del bus
	Número de maestros en el bus
	Número de esclavos en el bus

Consulte también

Principios básicos para parametrizar la interfaz PG/PC (Página 11311)

13.5.7.2 Parametrizar la interfaz MPI o PROFIBUS automáticamente

Configurar la detección automática de los parámetros del bus

Si se selecciona una interfaz con detección automática de los parámetros del bus, p. ej. CP5611 (Auto), es posible conectar la PG/el PC a MPI o PROFIBUS sin tener que ajustar los parámetros del bus. Sin embargo, si la velocidad de transferencia es inferior a 187,5 Kbits/s, los tiempos de espera pueden prolongarse hasta un minuto.

Requisitos

- El bus tiene conectados maestros que distribuyen los parámetros del bus cíclicamente.
- En redes PROFIBUS, la distribución cíclica de los parámetros del bus debe estar activada.

Procedimiento

Para activar la detección automática de los parámetros del bus, proceda del siguiente modo:

1. Seleccione la interfaz en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el diálogo de configuración de la interfaz.
3. En "General > Configuraciones > Configuración activa", seleccione el ajuste "Detección automática de protocolo".
4. En "Configuraciones > Configuración automática > Configuración local", seleccione la dirección de la interfaz PG/PC en la lista desplegable "Dirección propia".
5. Si, a continuación, desea visualizar la configuración actual del bus, haga clic en el botón "Detección de red".

Consulte también

Parametrizar la interfaz MPI (Página 11323)

Parametrizar la interfaz PROFIBUS (Página 11325)

13.5.7.3 Parametrizar la interfaz MPI

Modificar los parámetros de la interfaz MPI

Para la interfaz MPI es posible ajustar parámetros de red y parámetros del bus. Primero debería elegir un ajuste predeterminado y adaptarlo después en caso necesario.

Parametrizar la interfaz MPI con ajustes predeterminados

Para ajustar los parámetros de la interfaz MPI, proceda del siguiente modo:

1. Seleccione la interfaz en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el cuadro de diálogo de configuración de la interfaz.

3. En "General > Asignación", seleccione la subred con la que debe conectarse la interfaz en la lista desplegable "Conexión con subred".
4. En "General > Configuraciones", seleccione un ajuste predeterminado para los parámetros relacionados con el dispositivo y la red. Los ajustes predeterminados son adecuados para la mayoría de configuraciones. Elija uno de los ajustes siguientes:
   - Detección automática del protocolo
     La PG puede conectarse al MPI o al PROFIBUS sin tener que ajustar los parámetros de bus. Sin embargo, si la velocidad de transferencia es inferior a 187,5 Kbits/s, los tiempos de espera pueden prolongarse hasta un minuto. La condición fundamental para la detección automática es que haya maestros conectados al bus que distribuyan los parámetros de bus de forma cíclica. En el caso de las subredes PROFIBUS, la distribución cíclica de los parámetros del bus no puede estar desactivada (ajuste predeterminado de red PROFIBUS).
   - MPI
     Se selecciona el protocolo de transferencia "MPI". Se ajustan parámetros estándar que resultan adecuados para la mayoría de configuraciones. No obstante, los parámetros pueden modificarse en caso necesario.
   - PROFIBUS
     Se selecciona el protocolo de transferencia "PROFIBUS". Se ajustan parámetros estándar que resultan adecuados para la mayoría de configuraciones. No obstante, los parámetros pueden modificarse en caso necesario.

Modificar los parámetros predeterminados

Para adaptar los ajustes predeterminados a sus necesidades, cambie los parámetros correspondientes en "Configuraciones > MPI".

Es posible ajustar los siguientes parámetros relacionados con el dispositivo:

- Es el único maestro
  Con ello se desactiva una función de seguridad adicional contra perturbaciones del bus al acoplar la PG/el PC a la red, porque la PG/el PC es el único maestro del bus.
  - Active esta opción sólo si únicamente se han conectado esclavos a la PG o al PC.
  - Si la casilla de verificación "Es el único maestro" está activada, no es posible determinar el dispositivo conectado directamente en la ventana "Dispositivos accesibles".
- Dirección propia
  Este dato hace relación a la PG/al PC en la/el que se ha llamado la parametrización de la interfaz. Ajuste aquí la dirección de dispositivo local de la PG/el PC.
  - La dirección debe ser unívoca en la red.
  - Es la dirección con la que se accede a la PG/el PC en la red MPI.

- Comprobar
  Con esto se activa una función de seguridad adicional contra fallos de bus al acoplar la PG/el PC a la red. El driver comprueba si la dirección propia ya está ocupada por otro nodo. En este caso no sólo se tienen en cuenta los nodos activos, sino también los pasivos. El driver escucha también en el PROFIBUS. En la verificación automática, el acoplamiento de la PG/del PC a la red dura un poco más de tiempo. Para poder utilizar la función de verificación, el driver debe soportar la función. Además, la opción "Es el único maestro" no puede estar activada.
- Timeout
  Aumente el valor del tiempo de vigilancia de conexión si p. ej. existen problemas debido a tiempos de respuesta elevados en la red.

Es posible ajustar los siguientes parámetros de red:

- Dirección más alta:
  Elija aquí la dirección de dispositivo más alta que se haya configurado. Asegúrese de que en todos los dispositivos de una red MPI está ajustada la misma dirección de dispositivo más alta.
- Velocidad de transferencia:
  Elija aquí la velocidad de transferencia que debe utilizarse en la red MPI.

Consulte también

Parametrizar la interfaz MPI o PROFIBUS automáticamente (Página 11318)

13.5.7.4 Parametrizar la interfaz PROFIBUS

Modificar los parámetros de la interfaz PROFIBUS

Para la interfaz PROFIBUS es posible ajustar con más precisión parámetros de red y parámetros del bus. Primero debería elegir un ajuste predeterminado y adaptarlo después en caso necesario.

Parametrizar la interfaz PROFIBUS con ajustes predeterminados

Para ajustar los parámetros de la interfaz PROFIBUS, proceda del siguiente modo:

1. Seleccione la interfaz en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el diálogo de configuración de la interfaz.

3. En "General > Asignación", seleccione la subred con la que debe conectarse la interfaz en la lista desplegable "Conexión con subred".
4. En "General > Configuración", seleccione un ajuste predeterminado para los parámetros relacionados con el dispositivo y la red. Los ajustes predeterminados son adecuados para la mayoría de configuraciones. Elija uno de los ajustes siguientes:
   - Detección automática del protocolo
     La PG puede conectarse al MPI o al PROFIBUS sin tener que ajustar los parámetros de bus. Sin embargo, si la velocidad de transferencia es inferior a 187,5 Kbits/s, los tiempos de espera pueden prolongarse hasta un minuto. La condición fundamental para la detección automática es que haya maestros conectados al bus que distribuyan los parámetros de bus de forma cíclica. En el caso de las subredes PROFIBUS, la distribución cíclica de los parámetros del bus no puede estar desactivada (ajuste predeterminado de red PROFIBUS).
   - MPI
     Se selecciona el protocolo de transferencia "MPI". Se ajustan parámetros estándar que resultan adecuados para la mayoría de configuraciones. No obstante, los parámetros pueden modificarse en caso necesario.
   - PROFIBUS
     Se selecciona el protocolo de transferencia "PROFIBUS". Se ajustan parámetros estándar que resultan adecuados para la mayoría de configuraciones. No obstante, los parámetros pueden modificarse en caso necesario.

Modificar los parámetros predeterminados

Para adaptar los ajustes predeterminados a sus necesidades, cambie los parámetros correspondientes en "Configuraciones > PROFIBUS".

Es posible ajustar los siguientes parámetros relacionados con el dispositivo:

- Es el único maestro
  Con ello se desactiva una función de seguridad adicional contra perturbaciones del bus al acoplar la PG/el PC a la red, porque la PG/el PC es el único maestro del bus.
  - Active esta opción sólo si únicamente se han conectado esclavos a la PG o al PC.
  - Si la casilla de verificación "Es el único maestro" está activada, no es posible determinar el dispositivo conectado directamente en la ventana "Dispositivos accesibles".
- Dirección propia
  Este dato hace relación a la PG/al PC en la/el que se ha llamado la parametrización de la interfaz. Ajuste aquí la dirección de dispositivo local de la PG/el PC.
  - La dirección debe ser unívoca en la red.
  - Es la dirección con la que se accede a la PG/el PC en la red PROFIBUS.

- Comprobar
  Con esto se activa una función de seguridad adicional contra fallos de bus al acoplar la PG/el PC a la red. El driver comprueba si la dirección propia ya está ocupada por otro nodo. En este caso no sólo se tienen en cuenta los nodos activos, sino también los pasivos. El driver escucha también en el PROFIBUS. En la verificación automática, el acoplamiento de la PG/del PC a la red dura un poco más de tiempo. Para poder utilizar la función de verificación, el driver debe soportar la función. Además, la opción "Es el único maestro" no puede estar activada.
- Timeout
  Aumente el valor del tiempo de vigilancia de conexión si p. ej. existen problemas debido a tiempos de respuesta elevados en la red.

Es posible ajustar los siguientes parámetros de red:

- Dirección más alta:
  Elija aquí la dirección de dispositivo más alta que se haya configurado. Asegúrese de que en todos los dispositivos de una red PROFIBUS está ajustada la misma dirección de dispositivo más alta.
- Velocidad de transferencia:
  Elija aquí la velocidad de transferencia que debe utilizarse en la red PROFIBUS.
- Perfil:
  Existen cuatro alternativas distintas para los ajustes del PROFIBUS. "DP", "Estándar" y "Universal (DP/FMS)" son ajustes predeterminados que no pueden modificarse. La opción "Personalizado" permite adaptar los parámetros del bus.
  - Si ha seleccionado "Personalizado", vaya a "Configuraciones > PROFIBUS > Parámetros del bus" en la navegación local.
  - Si ha seleccionado uno de los ajustes predeterminados (DP, estándar o universal (DP/FMS)), debería activar la casilla de verificación "Considerar" en "Configuraciones > PROFIBUS > Parámetros del bus > Parámetros adicionales". Seguidamente, podrá ajustar el número de maestros y esclavos en el bus. De este modo, los parámetros del bus se calculan con más exactitud y es posible excluir posibles fallos del bus. La opción no puede activarse en un perfil definido por el usuario.

Consulte también

Resumen de los parámetros del bus para PROFIBUS (Página 11327)

Parametrizar la interfaz MPI o PROFIBUS automáticamente (Página 11318)

13.5.7.5 Resumen de los parámetros del bus para PROFIBUS

Introducción

La subred PROFIBUS sólo funciona a la perfección si los parámetros del perfil de bus están sintonizados entre sí. Por esta razón, los valores predeterminados sólo deberían modificarse si se está familiarizado con la parametrización del perfil de bus para PROFIBUS.

En función del perfil de bus, los parámetros podrán ajustarse o no. Siempre se visualizan los valores offline de los parámetros del bus, incluso si están conectados online con el sistema de destino.

Los parámetros mostrados son válidos para toda la subred PROFIBUS.

Significado de los diferentes parámetros

- Tslot: tiempo de espera a recepción (slot time)
  El tiempo de espera a recepción (slot time) determina cuánto tiempo debe esperar como máximo el emisor hasta recibir una respuesta del interlocutor direccionado.
- Max. Tsdr: tiempo máximo de procesamiento del protocolo (max. station delay responder)
  El tiempo máximo de procesamiento del protocolo determina el tiempo máximo al cabo del cual el dispositivo que responde debe haber procesado el protocolo.
- Min. Tsdr: tiempo mínimo de procesamiento del protocolo (min. station delay responder)
  El tiempo mínimo de procesamiento del protocolo determina el tiempo mínimo que requiere el dispositivo que responde para procesar el protocolo.
- Tset: tiempo de activación (setup time)
  El tiempo de activación es el tiempo que puede transcurrir entre la recepción de un telegrama de datos y la reacción correspondiente.
- Tqui: tiempo de desactivación del modulador (quiet time for modulator)
  El tiempo de desactivación del modulador es el tiempo necesario para conmutar de Enviar a Recibir.
- Factor GAP: factor de actualización GAP (factor GAP)
  El factor de actualización GAP determina al cabo de cuántas rotaciones del testigo debe incorporarse un nuevo dispositivo al anillo lógico.
- Retry Limit: número máximo de repeticiones de llamada (retry limit)
  Este parámetro determina cuántos intentos pueden realizarse como máximo para acceder a un dispositivo.
- Trdy: tiempo de disponibilidad (ready time)
  El tiempo de disponibilidad es el tiempo de acuse o respuesta.
- Tid1: tiempo de estado de reposo 1 (idle time 1)
  El tiempo de estado de reposo 1 determina el tiempo de retardo tras recibir una respuesta.
- Tid2: tiempo de estado de reposo 2 (idle time 2)
  El tiempo de estado de reposo 2 determina el tiempo de retardo tras enviar una llamada sin respuesta.
- Ttr: tiempo consignado de rotación del token (target rotation time)
  El tiempo consignado de rotación del token es el tiempo máximo disponible para una rotación del token. Durante ese tiempo, todos los dispositivos activos (maestros) reciben una vez el testigo (token). La diferencia entre los tiempos de rotación teórico y real del token determina cuánto tiempo restante tienen los maestros para enviar telegramas de datos a los esclavos.
  Seleccione como Target Rotation Time (Ttr) mínimo = 5000 multiplicado por HSA (Highest Station Address).
- Supervisión de respuesta: tiempo de supervisión de respuesta (watchdog)
  La supervisión de respuesta determina el tiempo que debe pasar para direccionar un dispositivo.
  Elija un tiempo mínimo de supervisión de respuesta = 6250 multiplicado por HSA.

Nota

Si desea crear un perfil de bus personalizado, tenga en cuenta que el tiempo mínimo consignado de rotación del token (Ttr) debe ser 5000 multiplicado por HSA (dirección PROFIBUS más alta). Además, la supervisión de respuesta mínima (watchdog) debe ser 6250 multiplicado por HSA.

Consulte también

Parametrizar la interfaz PROFIBUS (Página 11321)

13.5.7.6 Restablecer la configuración MPI o PROFIBUS

Si se han realizado cambios en los ajustes del protocolo MPI o PROFIBUS, existe la posibilidad de restablecer los valores predeterminados.

Procedimiento

Para restablecer los valores predeterminados de la configuración MPI o PROFIBUS, proceda del siguiente modo:

1. Seleccione la interfaz MPI/PROFIBUS en "Accesos online" en el árbol del proyecto.
2. Elija el comando "Propiedades" del menú contextual de la interfaz.
   Se abre el cuadro de diálogo de configuración de la interfaz.
3. Elija "Configuraciones > MPI" o "Configuraciones > PROFIBUS" en función de las propiedades de interfaz que desee restablecer.
4. Haga clic en el botón "Estándar" para restablecer todos los ajustes.

13.6 Uso de la función Trace y de analizador lógico

Prólogo

Finalidad de la documentación

En la presente documentación se describen las posibilidades de diagnóstico con la función Trace y de analizador lógico. Las posibilidades de registro pueden variar en función del dispositivo utilizado.

Conocimientos básicos necesarios

Para una mejor comprensión de la presente documentación se precisan los siguientes conocimientos:

- Conocimientos generales de automatización
- Conocimientos sobre la utilización de equipos Windows
- S7-1200/1500 CPUs
  - Conocimientos en el manejo del sistema de automatización industrial SIMATIC
  - Conocimientos en el manejo de TIA Portal >= V12.0
- SINAMICS G120
  - Conocimientos en el manejo del accionamiento

Ámbito de validez de la documentación

Esta documentación es válida para todos los productos de la familia S7-1200, S7-1500 y SINAMICS G120.

Convenciones

La presente documentación contiene ilustraciones de los dispositivos descritos. Las ilustraciones pueden diferir del dispositivo suministrado en algunos detalles.

Preste atención también a las notas marcadas del modo siguiente:

Nota

Una nota contiene datos importantes acerca del producto descrito en la documentación, el manejo de dicho producto o la parte de la documentación a la que debe prestarse especial atención.

Soporte adicional

- Encontrará nuestra oferta en documentación técnica para los diferentes productos y sistemas SIMATIC en Internet (http://www.siemens.com/simatic-tech-doku-portal).
- Encontrará el catálogo online y el sistema de pedidos online en Internet (http://mall.automation.siemens.com).

13.6.1 Información de seguridad

Siemens suministra productos y soluciones con funciones de seguridad industrial que contribuyen al funcionamiento seguro de instalaciones, soluciones, máquinas, equipos y redes. Dichas funciones son un componente importante de un sistema global de seguridad industrial. En consideración de lo anterior, los productos y soluciones de Siemens son objeto de mejoras continuas. Por ello, le recomendamos que se informe periódicamente sobre las actualizaciones de nuestros productos

Para el funcionamiento seguro de los productos y soluciones de Siemens, es preciso tomar medidas de protección adecuadas (como el concepto de protección de células) e integrar cada componente en un sistema de seguridad industrial integral que incorpore los últimos avances tecnológicos. También deben tenerse en cuenta los productos de otros fabricantes que se estén utilizando. Encontrará más información sobre seguridad industrial en (http://www.siemens.com/industrialsecurity).

Si desea mantenerse al día de las actualizaciones de nuestros productos, regístrese para recibir un boletín de noticias específico del producto que desee. Encontrará más información en (http://support.automation.siemens.com).

13.6.1 Descripción

13.6.1.1 Hardware admitido

Los dispositivos (Página 11358) siguientes admiten la función Trace y de analizador lógico:

- CPU SIMATIC S7-1200 (a partir de versión de firmware V4.0)
- CPU SIMATIC S7-1500
- SINAMICS G120

13.6.1.2 Registro de medidas con la función Trace

Introducción

La función Trace y de analizador lógico se abre en el navegador de proyecto, en la carpeta del dispositivo con el nombre "Traces" (Página 11334).

La función Trace y de analizador lógico permite registrar las variables de un dispositivo y evaluar los registros. Las variables son, p. ej., parámetros de accionamiento o variables de sistema y de usuario de una CPU. El tamaño de la memoria limita la duración de registro máxima. El espacio disponible de la memoria para el registro depende del hardware que se emplee.

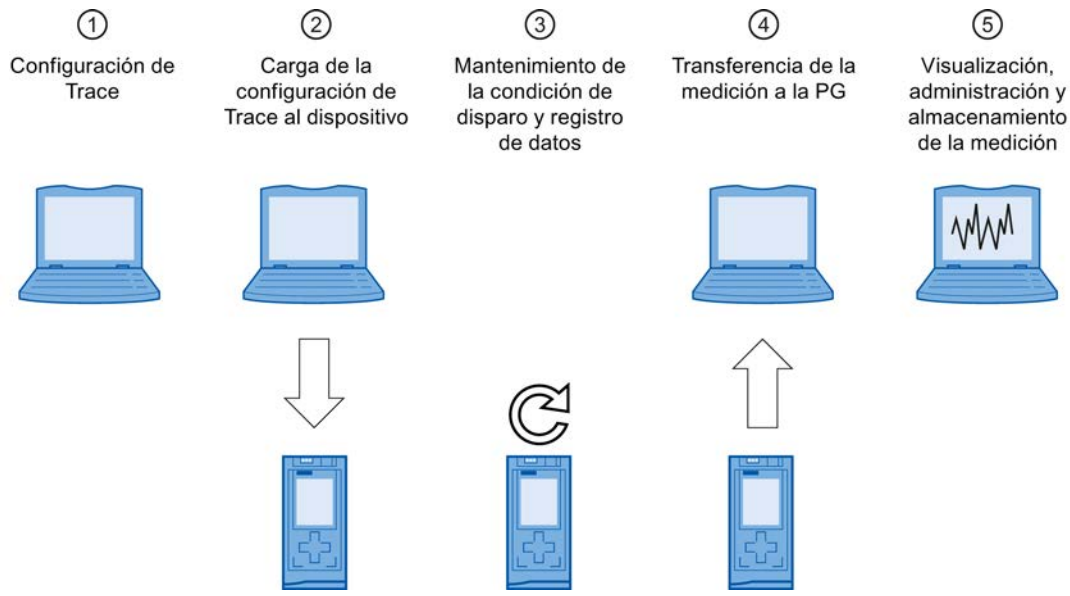
Los registros se guardan en el dispositivo, pueden leerse con el sistema de configuración (ES) en caso necesario y guardarse permanentemente. De tal forma, la función Trace y de analizador lógico es adecuada para visualizar procesos altamente dinámicos. Los valores registrados se sobrescriben al activar de nuevo "Traces en dispositivo".

Los registros activos del panel de mando de eje o del PID se muestran en "Traces en dispositivo". Estos registros se pueden añadir a las mediciones en el diagrama de curvas del panel de mando de eje o del PID mediante un comando del menú contextual.

Las posibilidades de registro pueden variar en función del dispositivo (Página 11358) utilizado.

En el capítulo Manejo encontrará una guía de inicio rápido (Página 11345) para trabajar con la función Trace y de analizador lógico.

La figura siguiente muestra el funcionamiento de "Traces":



① Configuración de Trace con la programadora (PG) en el TIA Portal

En la configuración de Trace se especifican las señales que deben registrarse, la duración del registro y la condición de disparo. La configuración de Trace depende del dispositivo y se describe en el correspondiente dispositivo (Página 11358).

② Transmisión de la configuración de Trace de la PG al dispositivo

Si existe una conexión online, se lleva a cabo la transmisión de la configuración de Trace (Página 11352) completa al dispositivo.

③ Esperar registro

Si la configuración de Trace está activada (Página 11352) en el dispositivo, el registro se realiza independientemente de la PG. Cuando se cumple la condición de disparo, se inicia el registro.

④ Transmisión de la medición del dispositivo a la PG

El almacenamiento de la medición en el proyecto (Página 11354) guarda la medición en el proyecto abierto del TIA Portal. El almacenamiento de la medición es independiente del momento de la medición y puede tener lugar en cualquier momento tras concluir el registro.

⑤ Evaluación, administración y almacenamiento de la medición

Para evaluar la medición, hay disponibles múltiples posibilidades en el diagrama de curvas y en la tabla de señales (Página 11353). Existen diferentes formatos de representación: p. ej., representación en bits para señales binarias.
También es posible exportar e importar mediciones como archivo.
Con el almacenamiento del proyecto (Página 11354) en el TIA Portal, también se guardan las mediciones transmitidas al proyecto.

13.6.1.3 Configuración de Trace, registro y medición

En este capítulo se explica el significado de los términos "configuración de Trace", "registro" y "medición".

configuración de Trace

En la configuración de Trace pueden realizarse los siguientes ajustes:

- Señales que deben registrarse
- Condiciones de registro
  - muestreo
  - disparo

registro

El registro tiene lugar en el dispositivo. Solo existe un registro para cada configuración de Trace del dispositivo. Si se reinicia un registro, el registro antiguo se sobrescribe.

Un registro del dispositivo no es remanente (se pierde tras conectar/desconectar el dispositivo) y puede guardarse permanentemente, como medición, en el proyecto.

Traces en dispositivo

Un Trace en el dispositivo se compone de una configuración de Trace y, opcionalmente, de un registro. El número máximo de Traces en el dispositivo depende del dispositivo.

El estado de un Trace en el dispositivo puede observarse online.

Medición

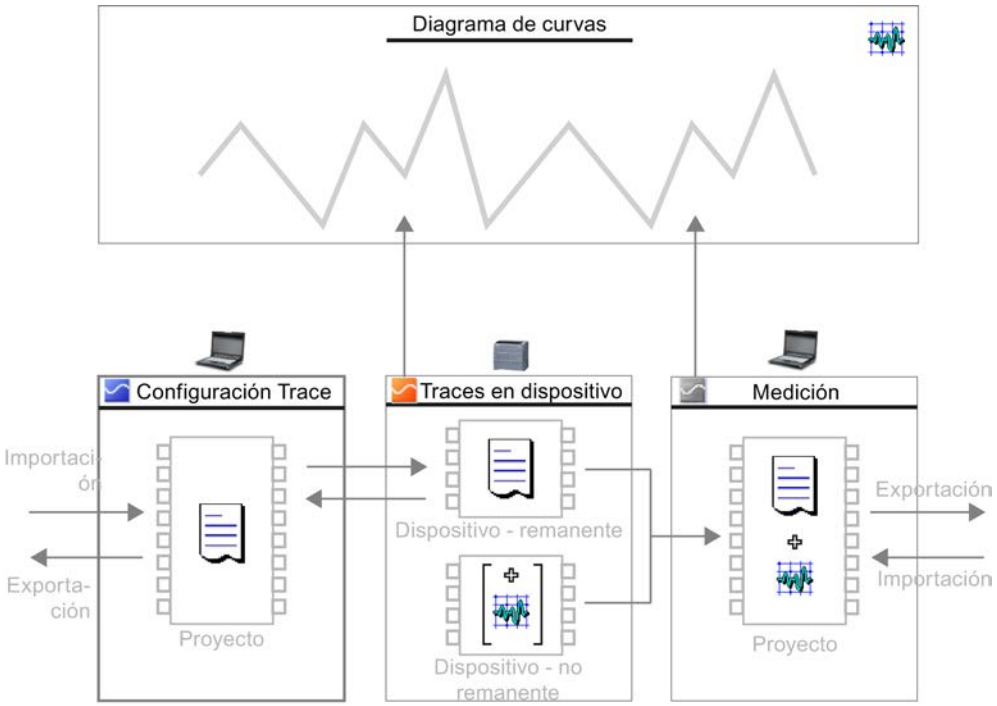
Una medición se compone siempre de una configuración de Trace con su registro correspondiente. Si un Trace en el dispositivo contiene un registro, puede guardarse en el proyecto como medición.

El registro de una medición puede observarse offline.

13.6.1.4 Almacenamiento de datos

La administración de Trace y el diagrama de curvas permiten, entre otros, la transmisión de la configuración de Trace y la observación del registro.

La figura siguiente muestra esquemáticamente el almacenamiento de datos:



Nota

Almacenamiento de configuración de Trace y medición

La configuración de Trace y la medición se guardan con el proyecto en el TIA Portal.

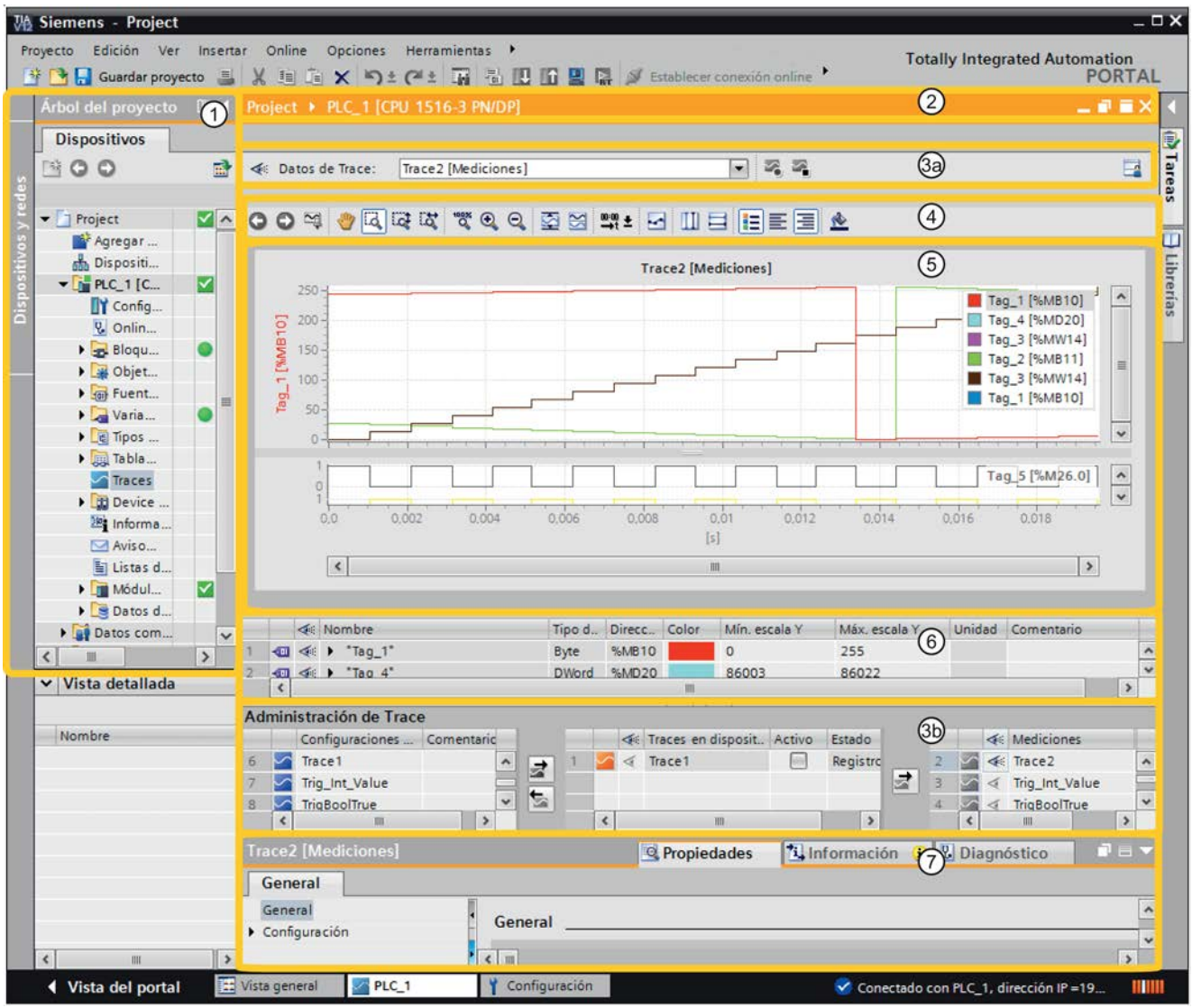
Si cierra el proyecto sin guardar, la configuración de Trace y las mediciones transmitidas al proyecto quedan descartadas. El editor de Trace puede cerrarse y volver a abrirse sin pérdida de datos hasta el cierre del proyecto.

13.6.2 Interfaz de software

Áreas de visualización

La interfaz de usuario de la función Trace y de analizador lógico se divide en varias áreas.

La figura siguiente muestra un ejemplo de distribución de la interfaz en el TIA Portal:



1	Navegador de proyecto	
	Área de trabajo	
2		Barra de título del área de trabajo Muestra a qué dispositivo corresponde la representación actual.
3a		Barra de herramientas de la administración de Trace (Página 11340) Selección de registros para la visualización en el diagrama de curvas y botones para activar/desactivar Traces en el dispositivo.
4		Barra de herramientas del diagrama de curvas (Página 11336) Herramientas para editar la medición y ajustar la representación en el diagrama de curvas
5		Diagrama de curvas (Página 11336) Visualización de los valores registrados
6		Tabla de señales (Página 11338) Señales de la medición visualizada

3b		Administración de Trace (Página 11340) Creación de nuevas configuraciones de Trace y administración de las mediciones.
	Área específica de dispositivo (ver Dispositivos (Página 11358))	
7		Área de propiedades en la ventana de inspección Diálogos de configuración para la duración del registro, la condición de disparo y la selección de señal.

13.6.2.1 Navegador de proyecto

Si un dispositivo soporta la función Trace y de analizador lógico, el árbol del proyecto ofrece la selección de "Traces" en el dispositivo .

Haciendo doble clic en "Traces", se abre el editor de Trace.

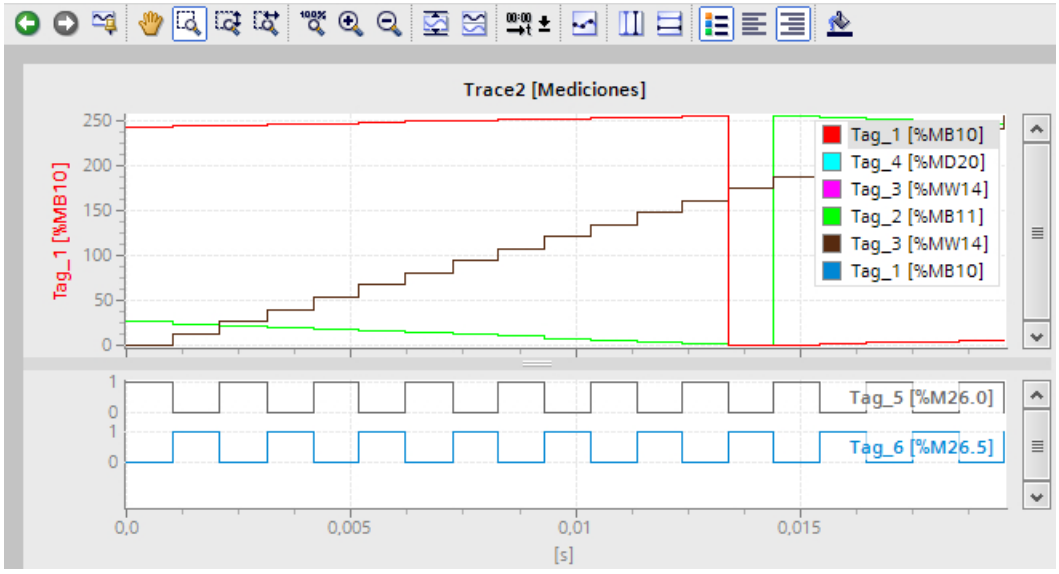
13.6.2.2 Área de trabajo

Interfaz Diagrama de curvas

El diagrama de curvas muestra las señales de un registro seleccionadas. Los bits se visualizan como traza de bits en el diagrama inferior. La representación de las señales se adapta en la tabla de señales (Página 11338) y con ayuda de la barra de herramientas del diagrama de curvas.

Posibilidades de ajuste y visualizaciones en el diagrama de curvas

La figura siguiente muestra un ejemplo de representación en el TIA Portal:



La escala del diagrama es válida para la señal seleccionada (sobre fondo gris) en la leyenda. La leyenda puede desplazarse a los laterales con el ratón.

Comandos del menú contextual

La tabla siguiente muestra los comandos del menú contextual en el diagrama de curvas:

Comando del menú contextual	Descripción
"Guardar diagrama como imagen en"	Exporta la visualización actual como mapa de bits.
"Copiar imagen en el portapapeles"	Copia la visualización actual en el portapapeles.
"Centrar cursor de medición"	Posiciona los cursores de medición activados en un punto central de la visualización actual.
"Agregar a las mediciones" (solo panel de mando de eje y PID)	Añade el registro visualizado a la tabla "Mediciones".

Barra de herramientas del diagrama de curvas

Mediante botones se ponen a disposición herramientas para ajustar la representación.

La tabla siguiente muestra las funciones de los botones:

Símbolo	Función	Descripción
	Deshacer zoom	Acción de deshacer la última función de zoom ejecutada. Si se han ejecutado varias funciones de zoom, estas pueden deshacerse paso a paso.
	Repetir zoom	Repetición de la última función de zoom deshecha. Si se han deshecho varias funciones de zoom, estas pueden repetirse paso a paso.
	Vista normal	Utilizar la vista actual como estándar para este registro. Si el registro de Trace se vuelve a mostrar con posterioridad, la vista normal se restablece.
	Desplazar la imagen	Desplazamiento de la visualización con el botón del ratón pulsado.
	Selección de zoom	Selección de un área cualquiera con el botón del ratón pulsado. La visualización se escala en función del área seleccionada.
	Selección de zoom vertical	Selección de un área vertical con el botón del ratón pulsado. La visualización se escala en función del área seleccionada.
	Selección de zoom horizontal	Selección de un área horizontal con el botón del ratón pulsado. La visualización se escala en función del área seleccionada.
	Visualizar todo	Escala la visualización de los datos existentes de modo que se muestren todo el rango de tiempo y todos los valores. **Nota** La barra de progreso dinámica de un Trace en curso se detiene al activar una función de zoom. Con este botón se activa de nuevo la barra de progreso.

Símbolo	Función	Descripción
	Zoom más	Ampliación de la visualización. Las áreas de eje de tiempo y eje de valor se reducen cada vez que se hace clic en el botón. Las curvas se representan más grandes.
	Zoom menos	Reducción de la visualización. Las áreas de eje de tiempo y eje de valor se amplían cada vez que se hace clic en el botón. Las curvas se representan más pequeñas.
	Escalado automático	Escalado de la visualización de modo que se muestren todos los valores para el rango de tiempo representado actualmente.
	Disponer en pistas	Las señales se disponen una bajo la otra sin solapamientos.
	Conmutación de la unidad del eje de tiempo	Conmutación de la unidad entre tiempo y ciclos.
	Visualizar puntos de medición	Los puntos de medición se muestran como pequeños círculos en las curvas.
	Visualizar cursores de medición verticales	Visualización de los cursores de medición verticales. La posición vertical de los dos cursores de medición puede desplazarse con el ratón. En la tabla de señales se muestran en función de la posición las medidas correspondientes y la diferencia de los cursores de medición.
	Visualizar cursores de medición horizontales	Visualización de los cursores de medición horizontal. La posición horizontal de los dos cursores de medición se puede modificar con el ratón.
	Mostrar leyenda	Mostrar u ocultar la leyenda en el diagrama de curvas.
	Mostrar leyenda justificada a la izquierda	Visualización de la leyenda a la izquierda del diagrama de curvas.
	Mostrar leyenda justificada a la derecha	Visualización de la leyenda a la derecha del diagrama de curvas.
	Cambiar color de fondo	Cambio entre diferentes colores de fondo.

Interfaz de la tabla de señales

La tabla de señales muestra una lista de las señales de la medición seleccionada y ofrece posibilidades de ajuste para algunas propiedades. Cuando se muestran los datos de registro de "Traces en dispositivo" y se modifican los ajustes de la tabla de señales, estos ajustes solo pueden conservarse hasta que se cambie al modo offline. Al añadir a las mediciones el "Trace en el dispositivo", se guardan los ajustes actuales de la tabla de señales en la medición.

Las señales se pueden ordenar mediante Drag & Drop. Es posible cambiar el orden de los bits dentro de una señal.

Posibilidades de ajuste y visualizaciones en la tabla de señales

La figura siguiente muestra un ejemplo de representación en el TIA Portal:

			Nombre	Tipo de datos	Dirección	Color	Mín. escala Y	Máx. escala Y	Y(t1)	Y(t2)	ΔY	Unidad	Comentario
1			"Tag_1"	Byte	%MB10		1	20	7	13	6		
2			"Tag_4"	DWord	%MD20		1	20	7	13	6		
3			"Tag_3"	Word	%MW14		3	60	21	39	18		

La tabla siguiente muestra los ajustes y las visualizaciones de las señales registradas:

Columna	Descripción
	Visualización estática del símbolo de señal
	Selección para la visualización en el diagrama de curvas El punto indica que para la señal en la selección de bits hay al menos un bit seleccionado para visualizar como traza de bits.
"Nombre"	Visualización del nombre de señal A hacer clic una vez sobre el nombre de una señal mostrada se actualiza la escala en el diagrama de curvas.
	Apertura de selección de bits Para los siguientes tipos de datos pueden seleccionarse también bits por separado para la visualización como traza de bits en el diagrama de curvas inferior: • Byte, Word, DWord, LWord • SInt, USInt, Int, UInt, DInt, UDInt, LInt, ULInt Ejemplo de una selección de bits abierta para el tipo de datos DWord: 1 "Tag_1" Byte %MB10 2 "Tag_4" DWord %MD20 3 Bool %M23.0 4 Bool %M23.1 5 Bool %M23.2 Seleccione o deseleccione el bit correspondiente la visualización haciendo clic en el símbolo .
"Tipo de datos"	Visualización del tipo de datos
"Dirección"	Visualización de la dirección (no con variables simbólicas)
"Color"	Visualización y posibilidad de ajuste del color de la señal
"Escala Y mín."	Visualización o introducción del valor mínimo para el escalado de la señal
"Escala Y máx."	Visualización o introducción del valor máximo para el escalado de la señal
"Grupo de señales"	Muestra o escribe el nombre de un grupo de señales Para todas las señales de un grupo de señales se hacen iguales las escalas en Y. Escriba un mismo nombre de grupo de señales para todas las señales que deban tener la misma escala. Es posible eliminar señales borrando el nombre de un determinado grupo de señales. Los grupos de señales pueden guardarse usando la función "Usar vista actual como estándar" (botón).
"Y(t1)"	Visualización del valor en la posición del primer cursor de medición
"Y(t2)"	Visualización del valor en la posición del segundo cursor de medición

Columna	Descripción
"ΔY"	Visualización de la diferencia de valor entre el primer y el segundo cursor de medición
"Unidad"	Visualización de la unidad (p. ej., con objetos tecnológicos)
"Comentario"	Visualización y posibilidad de introducción de un comentario relativo a la señal

Comandos del menú contextual

La tabla siguiente muestra los comandos del menú contextual de la tabla de señales:

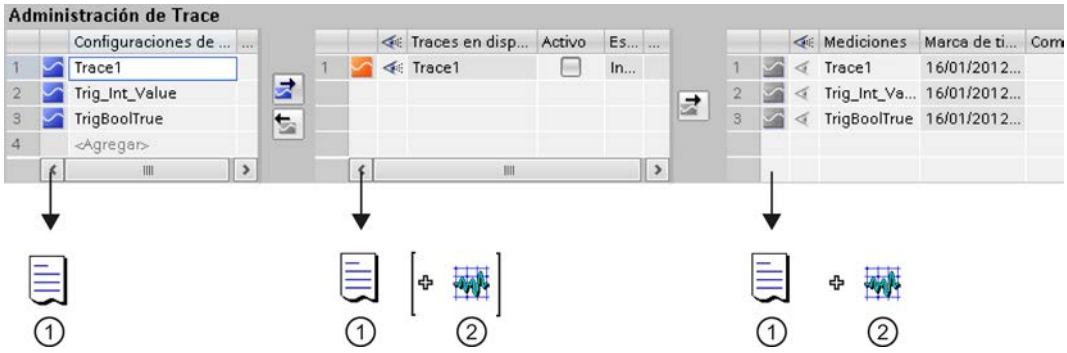
Comando del menú contextual	Descripción
"Cortar"	No seleccionable
"Copiar"	Copia el contenido de las filas seleccionadas en el portapapeles.
"Insertar"	No seleccionable
"Mostrar/ocultar señal"	Muestra u oculta la señal en el diagrama de curvas.
"En primer plano"	Muestra la señal seleccionada en primer plano. El resto de señales quedan en segundo plano.

Interfaz Administración de Trace

La administración de Trace permite configurar y administrar configuraciones de Trace.

Posibilidades de ajuste y visualizaciones en la administración de Trace

La figura siguiente muestra un ejemplo de representación en el TIA Portal:



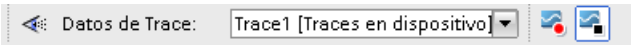
① Configuración de Trace

② Registro (valores registrados)

Barra de herramientas de la administración de Trace

A través de una lista desplegable y de los botones es posible activar, desactivar o seleccionar un registro para su visualización.

La figura siguiente muestra un ejemplo de la barra de herramientas de la administración de Trace



La tabla siguiente describe las funciones de los botones:

Símbolo	Descripción
	Símbolo estático: visualización de la selección de listas desplegables en el diagrama de curvas
	Selección de un "Trace en dispositivo" o de una "Medición" para su visualización en el diagrama de curvas
	Activación del registro El botón está disponible en "Traces en dispositivo".
	Desactivación del registro El botón está disponible en "Traces en dispositivo".

Tabla "Configuraciones de Trace"

La tabla contiene todas las configuraciones de Trace del proyecto que pueden transmitirse al dispositivo. Cada configuración de Trace contiene las variables que deben registrarse y los ajustes de registro necesarios. Los datos se muestran en modo offline y online.

La tabla siguiente muestra los ajustes y las visualizaciones:

Columna	Símbolo	Descripción
-		Visualización del símbolo para la configuración de Trace Las configuraciones de Trace incorrectas se representan con un marco rojo y no pueden transmitirse al dispositivo. Compruebe la configuración en la ventana de inspección. Al hacer doble clic en el símbolo, la ventana de inspección con las propiedades aparece en primer plano.
"Configuración de Trace"	-	Nombre de la configuración de Trace El nombre puede modificarse haciendo doble clic en el campo.
"Comentario"	-	Campo de entrada para un comentario

Comandos del menú contextual

La tabla siguiente muestra los comandos del menú contextual de la tabla "Configuraciones de Trace" con la fila seleccionada:

Comando del menú contextual	Descripción
"Cortar"	No seleccionable
"Copiar"	Copia el contenido de las filas seleccionadas en el portapapeles.
"Insertar"	Inserta el contenido del portapapeles.
"Borrar"	Borra las filas seleccionadas de la tabla o borra el nombre de la celda seleccionada.
"Cambiar nombre"	Pasa la celda seleccionada al modo de edición.

Comando del menú contextual	Descripción
"Transferir configuración de Trace al dispositivo"	Transmite la configuración de Trace seleccionada en la tabla "Configuraciones de Trace" al dispositivo. El botón está activo si se han efectuado correctamente todos los ajustes necesarios en la configuración específica del dispositivo, en la ventana de inspección, y se ha establecido una conexión online.
"Exportar configuración de Trace"	Exporta una configuración de Trace como archivo con la extensión "*.ttcfgx". La extensión de archivo "*.ttcfg" está soportada por motivos de compatibilidad con V12, pero no contiene información sobre la familia de dispositivos.
"Importar configuración de Trace"	Importa una configuración de Trace desde un archivo.
"Propiedades"	Trae la ventana de inspección con las propiedades específicas del dispositivo a un primer plano.

Tabla "Traces en dispositivo"

La tabla contiene las configuraciones de Trace ya transmitidas al dispositivo y su estado actual. Los datos se muestran en modo online.

La tabla siguiente muestra los ajustes y las visualizaciones:

Columna	Símbolo	Descripción
-		Visualización estática del símbolo de modo online Al hacer doble clic en el símbolo, la ventana de inspección con las propiedades específicas del dispositivo aparece en primer plano.
		Selección para la visualización en el diagrama de curvas Al hacer clic en el símbolo, se muestra u oculta el registro en el diagrama de curvas.
"Traces en dispositivo"	-	Nombre de la configuración de Trace
"Activo"		Activación del registro y notificación del estado La casilla de verificación se desactiva al final del registro. Si se repite el registro de un Trace en el dispositivo, se conservan también para el nuevo registro los ajustes relevantes para la visualización (diagrama de curvas y tabla de señales).
"Estado"	-	Visualización del estado de la configuración de Trace Existen los siguientes estados: • "Instalando" • "Esperar disparo" • "Registro" • "Registro finalizado" • "Cancelado" • "Estado de error"
"Comentario"	-	Visualización del comentario de la configuración de Trace

Comandos del menú contextual

La tabla siguiente muestra los comandos del menú contextual de la tabla "Traces en dispositivo" con la fila seleccionada:

Comando del menú contextual	Descripción
"Cortar"	No seleccionable
"Copiar"	Copia el contenido de las filas seleccionadas en el portapapeles.
"Insertar"	No seleccionable
"Borrar"	Borra las filas seleccionadas de la tabla.
"Cambiar nombre"	No seleccionable
"Agregar a las configuraciones de Trace"	Trasmite la configuración de Trace del dispositivo a la tabla Configuraciones de Trace.
"Agregar a las mediciones"	Agrega el "Trace en dispositivo" seleccionado a la tabla "Mediciones". **Nota** Los datos de "Trace en el dispositivo" se pueden guardar como medición una vez visualizados en el diagrama de curvas. Para la visualización se cargan los datos del registro desde el dispositivo.
"Activar"	Activa el registro y muestra el estado en la tabla
"Mostrar/ocultar en el diagrama de curvas"	Muestra u oculta el registro en el diagrama de curvas.
"Propiedades"	Trae la ventana de inspección con las propiedades específicas del dispositivo a un primer plano.

Tabla "Mediciones"

La tabla contiene las configuraciones de Trace con registros. Los datos se muestran en modo offline y online.

Las mediciones también pueden exportarse e importarse; ver Exportación e importación de mediciones (Página 11354).

La tabla siguiente muestra los ajustes y las visualizaciones:

Columna	Símbolo	Descripción
-		Visualización estática del símbolo para una medición Al hacer doble clic en el símbolo, la ventana de inspección con las propiedades específicas del dispositivo aparece en primer plano.
		Selección para la visualización en el diagrama de curvas Al hacer clic en el símbolo, se muestra u oculta el registro en el diagrama de curvas.
"Mediciones"	-	Nombre de la medición El nombre puede modificarse haciendo doble clic en el campo.

Columna	Símbolo	Descripción
"Marca de tiempo"	-	El significado de la marca de tiempo depende del dispositivo Ejemplos: • SIMATIC S7-1200/1500 CPUs Con el inicio del registro, se introduce la hora absoluta del control. • SINAMICS G120 Con la transferencia de la configuración de Trace al dispositivo se introduce el momento de transferencia.
"Comentario"	-	Campo de entrada para un comentario

Comandos del menú contextual

La tabla siguiente muestra los comandos del menú contextual de la tabla "Mediciones" con la fila seleccionada:

Comando del menú contextual	Descripción
"Cortar"	No seleccionable
"Copiar"	Copia el contenido de las filas seleccionadas en el portapapeles.
"Insertar"	Inserta el contenido del portapapeles.
"Borrar"	Borra las filas seleccionadas de la tabla o borra el nombre de la celda seleccionada.
"Cambiar nombre"	Pasa la celda seleccionada al modo de edición.
"Agregar a las configuraciones de Trace"	Trasmite la configuración de Trace de la medición a la tabla Configuraciones de Trace.
"Mostrar/ocultar en el diagrama de curvas"	Muestra u oculta el registro en el diagrama de curvas.
"Exportar medición"	Exporta una medición como archivo con la extensión "*.ttrecx" o "*.csv". La extensión de archivo "*.ttrec" está soportada por motivos de compatibilidad con V12, pero no contiene información sobre la familia de dispositivos.
"Importar medición"	Importa una medición desde un archivo con la extensión "*.ttrecx". La extensión de archivo "*.ttrec" está soportada por motivos de compatibilidad con V12, pero no contiene información sobre la familia de dispositivos.
"Propiedades"	Trae la ventana de inspección con las propiedades específicas del dispositivo a un primer plano.

Botones

La tabla siguiente muestra las funciones disponibles mediante botones:

Símbolo	Descripción
	Botón para la transmisión de una configuración de Trace seleccionada al dispositivo La configuración de Trace seleccionada en la tabla "Configuraciones de Trace" se transmite al dispositivo. El botón está activo si se han efectuado correctamente todos los ajustes necesarios en la configuración, en la ventana de inspección, y se ha establecido una conexión online.
	Botón para la transmisión de una configuración de Trace seleccionada desde el dispositivo La configuración de Trace seleccionada en la tabla "Traces en dispositivo" se transmite del dispositivo a la tabla "Configuraciones de Trace".
	Botón para la transmisión de una medición seleccionada del dispositivo al proyecto La selección de la tabla "Traces en dispositivo" se agrega a la tabla "Mediciones". **Nota para las CPU S7** Para guardar los datos de "Trace en el dispositivo" como medición, antes este debe mostrarse una vez en el diagrama de curvas. Con la visualización se cargan los datos del registro desde el dispositivo.

13.6.2.3 Área específica de dispositivo

El área específica de dispositivo permite realizar la configuración de Trace.

Aquí se especifican las señales que deben registrarse y la condición de disparo. Estas y otras posibilidades de configuración se describen en el área específica de dispositivo (Página 11358) de cada dispositivo.

13.6.3 Manejo

13.6.3.1 Guía de inicio rápido

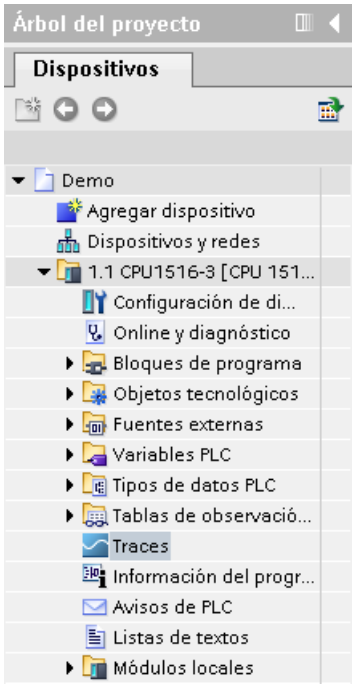
Esta descripción toma como ejemplo los pasos para efectuar un registro con la CPU S7-1500. Los ajustes mostrados pueden diferir en función del dispositivo.

Requisitos

Se ha configurado un dispositivo que admite la función Trace y de analizador lógico.

Llamada del editor de Trace

La figura siguiente muestra el navegador de proyecto con "Traces" en el dispositivo:

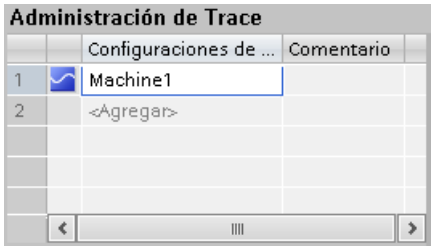


Procedimiento:

1. Abra el editor de Trace haciendo doble clic en la entrada "Traces".

Creación de una configuración de Trace

La figura siguiente muestra la tabla "Configuraciones de Trace":

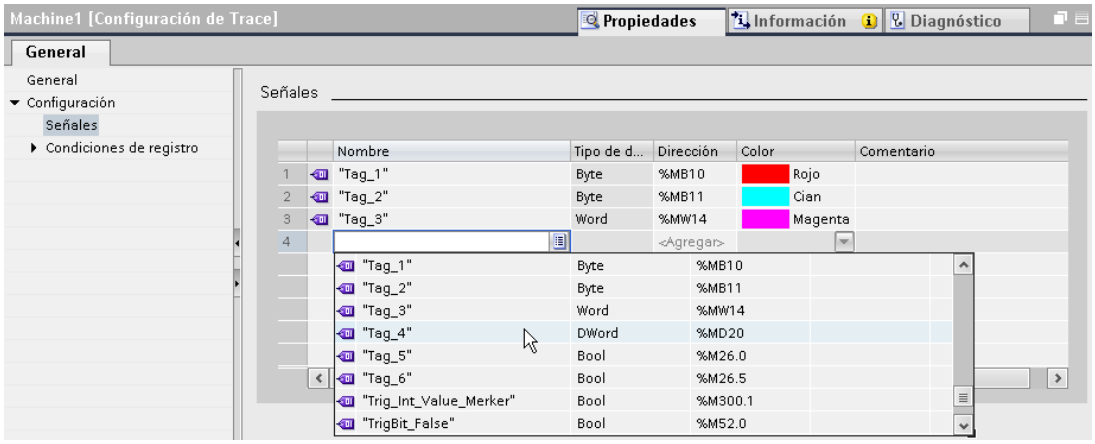


Procedimiento:

1. Introduzca el nombre de la configuración de Trace en la tabla "Configuraciones de Trace".

Selección de señales

La figura siguiente muestra la configuración de las señales:



Procedimiento:

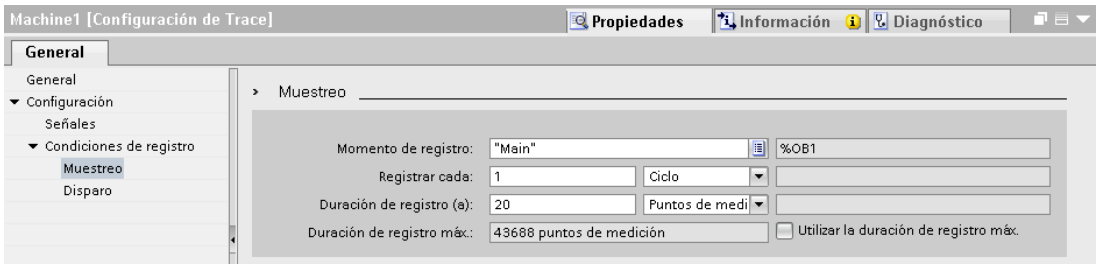
1. Seleccione las señales que deben registrarse en el área "Señales" de la ventana de inspección.

O bien:

1. Arrastre una señal hasta la tabla mediante Drag & Drop.

Configuración del ciclo de registro

La figura siguiente muestra la configuración del muestreo:

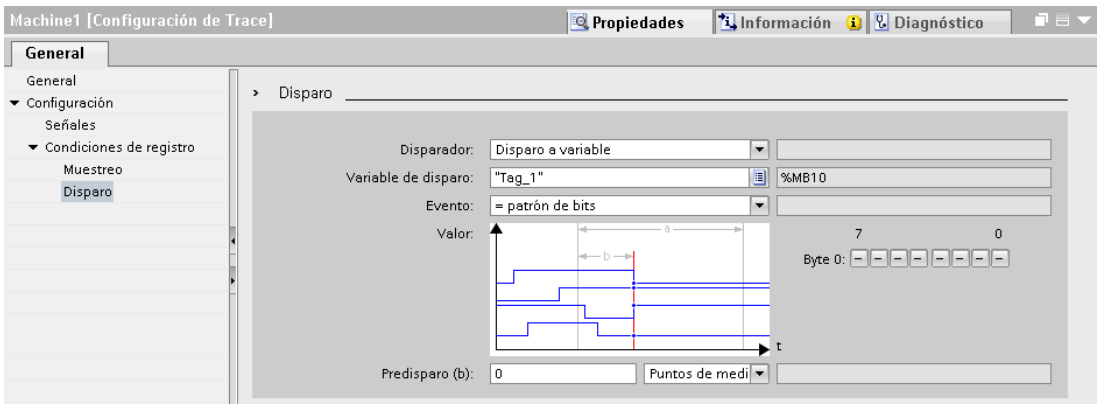


Procedimiento:

1. Configure el muestreo.

Configuración del disparador

La figura siguiente muestra la configuración del disparador:

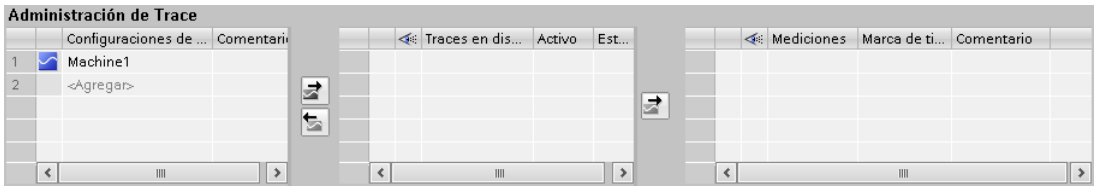


Procedimiento:

1. Configure el disparador y la condición de disparo para el disparador seleccionado.

Transmisión de la configuración de Trace al dispositivo

La figura siguiente muestra la administración de Trace en modo online:

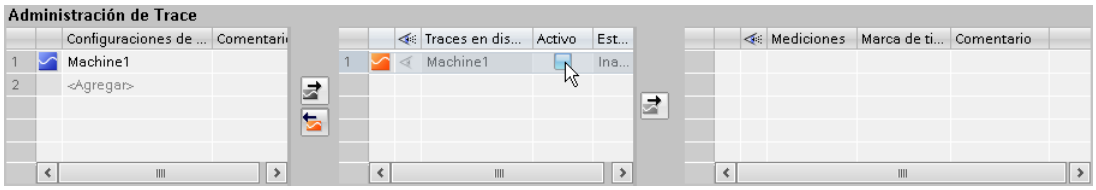


Procedimiento:

1. Compruebe, con el símbolo de la tabla "Configuraciones de Trace", si la configuración es correcta.
   El símbolo muestra una configuración de Trace correcta.
   El símbolo muestra una configuración de Trace incorrecta.
   Solo pueden transmitirse al dispositivo las configuraciones de Trace correctas.
2. Establezca una conexión online con el dispositivo.
3. Seleccione la configuración de Trace en la tabla "Configuraciones de Trace" y transmítala al dispositivo con el botón .

Activación de registro

La figura siguiente muestra la administración de Trace con una configuración de Trace en el dispositivo:

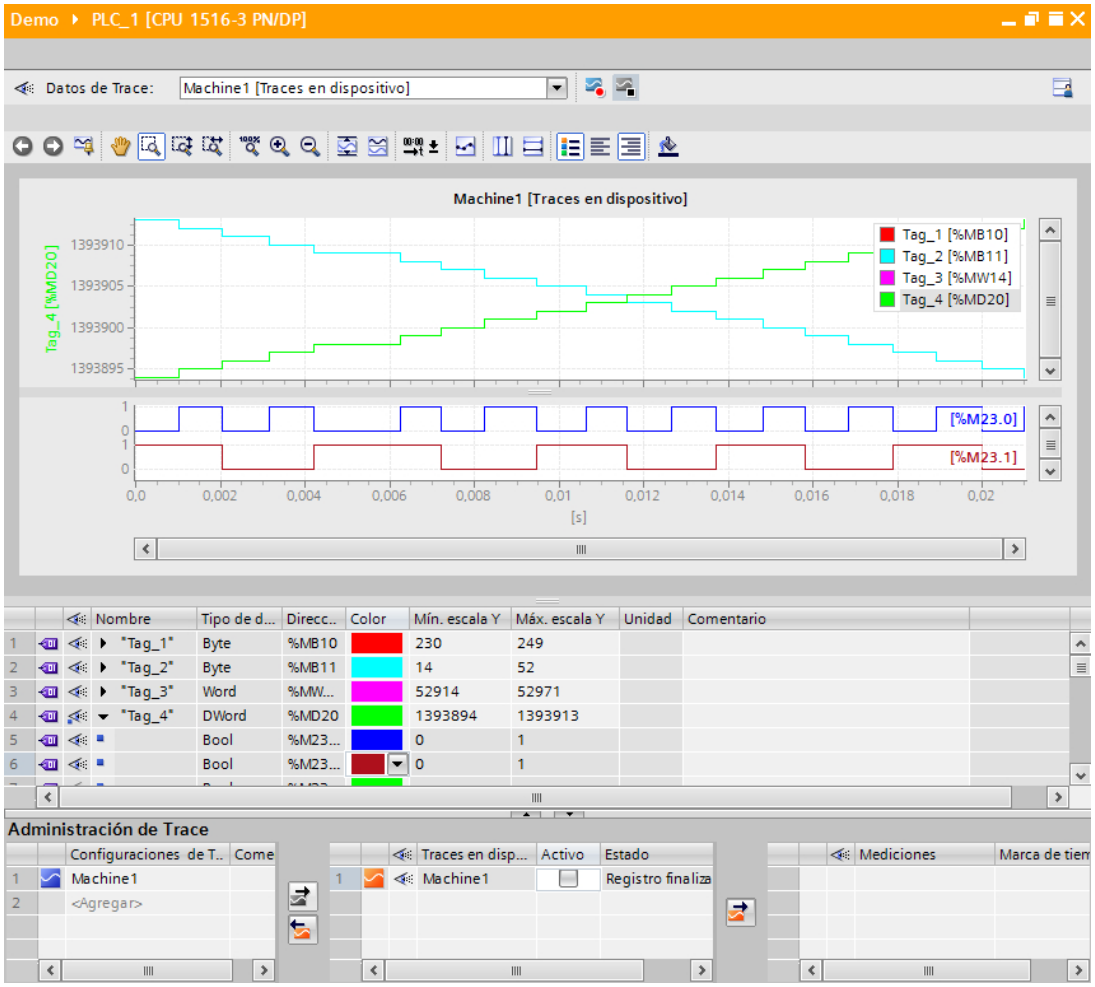


Procedimiento:

1. Haga clic en la casilla de verificación de la columna "Activo" en la tabla "Traces en dispositivo".

Visualización de registro

La figura siguiente muestra el diagrama de curvas con el registro de "Traces en dispositivo":

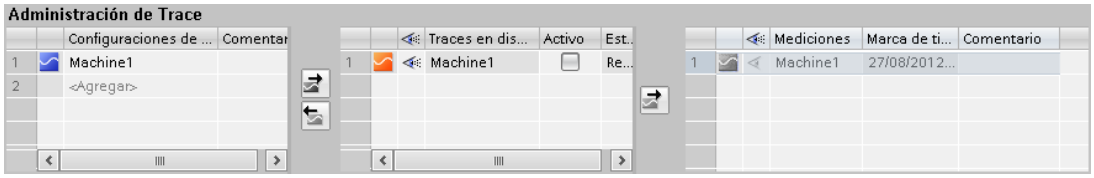


Procedimiento:

1. Espere hasta que se muestre el estado "Registro" o "Registro finalizado" en la tabla "Traces en dispositivo".
2. Haga clic en el símbolo de la tabla "Traces en dispositivo" para visualizar el registro en el diagrama de curvas.
3. Haga clic en el símbolo de una señal en la tabla de señales.
   Se ofrecen los bits de la señal por separado para su visualización como traza de bits.
4. Seleccione o deseleccione en la tabla de señales las señales individuales y los bits con el símbolo para su representación.

Almacenamiento de la medición en el proyecto

La figura siguiente muestra la medición guardada en el proyecto:



Procedimiento:

1. Seleccione la medición en la tabla "Traces en dispositivo" y transmítala al proyecto con el botón .

13.6.3.2 Uso de la función Trace: vista general

Requisitos

En el TIA Portal se ha configurado un dispositivo que admite la función Trace y de analizador lógico y con el que hay establecida una conexión online.

Procedimiento

La tabla siguiente muestra una vista general de las etapas que se siguen normalmente al trabajar con la función Trace y de analizador lógico.

Etapa	Descripción
1	Llamada del editor de Trace (Página 11351)
2	Configuración de Trace (Página 11351)
3	Transmisión de la configuración de Trace al dispositivo (Página 11352)
4	Activación/desactivación de Trace en el dispositivo (Página 11352)
5	Visualización de registro (Página 11353)
6	Almacenamiento de mediciones en el proyecto (Página 11354)
7	Visualización de registro (Página 11353)

13.6.3.3 Llamada del editor de Trace

Por dispositivo puede abrirse un editor de Trace en el navegador de proyecto.

A continuación se describe cómo llamar al editor de Trace en el área de trabajo del TIA Portal.

Requisitos

Se ha configurado un dispositivo que admite la función Trace y de analizador lógico.

Procedimiento

Para llamar al editor de Trace, proceda del siguiente modo:

1. En el navegador de proyecto, haga doble clic en el dispositivo.
   La carpeta del dispositivo se abre.
2. Haga doble clic en "Traces".

13.6.3.4 Administración de Trace

Configuración de Trace

Requisitos

El editor de Trace está abierto.

Creación y borrado de una configuración de Trace

Para crear una nueva configuración de Trace, proceda del siguiente modo:

1. En la administración de Trace (Página 11336), haga clic en la entrada "<Agregar>" de la tabla "Configuraciones de Trace".
2. Introduzca el nombre para la configuración de Trace.
   Se crea la nueva configuración de Trace.

Para borrar una configuración de Trace, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una configuración de Trace de la tabla "Configuraciones de Trace" y seleccione el comando "Borrar" del menú contextual.
   Se borra la configuración de Trace.

Configuración de Trace

Durante la configuración se especifican las condiciones de registro y disparo y se seleccionan las señales que deben registrarse.

Ver capítulo "Configuración" (en el dispositivo (Página 11358) correspondiente).

Nota

Almacenamiento de configuración de Trace

La configuración de Trace se guarda con el proyecto en el TIA Portal.

Si se cierra el proyecto sin guardar, la configuración queda descartada. El editor de Trace puede cerrarse y volver a abrirse sin pérdida de datos hasta el cierre del proyecto.

Transmisión de la configuración de Trace al dispositivo

Requisitos

- Hay una configuración de Trace válida en la tabla "Configuraciones de Trace".
- No se ha alcanzado todavía el número máximo de "Traces en dispositivo".
- Existe una conexión online con el dispositivo.

Procedimiento

Para transmitir una configuración de Trace al dispositivo, proceda del siguiente modo:

1. Seleccione una configuración de Trace en la tabla "Configuraciones de Trace".
2. Haga clic en el botón para transmitir la configuración de Trace al dispositivo.

Resultado

La configuración de Trace se transmite al dispositivo y se muestra en la tabla "Traces en dispositivo".

Activación/desactivación de Trace en el dispositivo

Requisitos

Hay una configuración de Trace en el dispositivo y se ha establecido una conexión online con este.

Activación de Trace en el dispositivo

Para activar el registro de un Trace en el dispositivo, proceda del siguiente modo:

1. Active la casilla de verificación de la columna "Activo" de la tabla "Traces en dispositivo". El Trace en el dispositivo se activa y se inicia el registro en función de la condición de disparo configurada. La condición de disparo es específica del dispositivo y se describe en el capítulo "Configuración", en el dispositivo (Página 11358) correspondiente.

Nota

Si se reinicia el registro, los valores registrados hasta el momento se pierden.

Para realizar una copia de seguridad de los valores registrados, guarde la medición en el proyecto (Página 11354) antes de volver a activar el registro.

Desactivación de Trace en el dispositivo

Para desactivar un Trace en el dispositivo activado, proceda del siguiente modo:

1. Desactive la casilla de verificación de la columna "Activo" de la tabla "Traces en dispositivo". El Trace en el dispositivo se desactiva.

Visualización de registro

Requisitos

Hay una configuración de Trace con registro en el dispositivo y se ha establecido una conexión online con este o hay una entrada en la tabla "Mediciones".

Procedimiento

Para visualizar el registro, proceda del siguiente modo:

1. Seleccione una entrada de la tabla "Traces en dispositivo".
2. Haga clic en la columna para la entrada seleccionada.

O:

1. Seleccione una entrada en la tabla "Mediciones".
2. Haga clic en la columna para la entrada seleccionada.

Resultado

El registro se muestra en el diagrama de curvas y en la tabla de señales.

Almacenamiento de mediciones en el proyecto

Requisitos

Hay una configuración de Trace con registro en el dispositivo y se ha establecido una conexión online con este.

Los datos de "Trace en el dispositivo" han de haberse mostrado en el diagrama de curvas al menos una vez. Para la visualización se cargan los datos del registro desde el dispositivo.

Procedimiento

Para guardar un registro en el proyecto, proceda del siguiente modo:

1. En la administración de Trace, haga clic en una entrada de la tabla "Traces en dispositivo".
2. Haga clic en el botón para transferir la entrada seleccionada.
   La entrada seleccionada de la tabla "Traces en dispositivo" se agrega a la tabla "Mediciones".
3. Guarde el proyecto en el TIA Portal.

Exportación e importación de mediciones

Requisitos

Hay al menos una medición en la tabla "Mediciones".

Exportación de mediciones

Para exportar una medición, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en una entrada de la tabla "Mediciones" y seleccione el comando "Exportar medición" del menú contextual.
2. Seleccione una carpeta, un nombre de archivo y un tipo de archivo para guardar la medición.
3. Haga clic en el botón "Guardar".

Importación de mediciones

Para importar una medición, proceda del siguiente modo:

1. Haga clic con el botón derecho del ratón en la tabla "Mediciones" y seleccione el comando "Importar medición" del menú contextual.
2. Seleccione el archivo tipo "*.ttrecx" con la medición que debe importarse.
3. Haga clic en el botón "Abrir".
   La medición importada se muestra en la tabla "Mediciones" con el nombre de archivo.

Transmisión de la configuración de Trace del dispositivo al proyecto

Requisitos

Hay una configuración de Trace en el dispositivo y se ha establecido una conexión online con este.

Procedimiento

Para transmitir una configuración de Trace al proyecto, proceda del siguiente modo:

1. Seleccione una entrada de la tabla "Traces en dispositivo".
2. Haga clic en el botón para transmitir la configuración de Trace desde el dispositivo.

Resultado

La configuración de "Trace en dispositivo" se incluye en la tabla "Configuraciones de Trace".

Borrado de Traces en el dispositivo

Requisitos

Hay un Trace en el dispositivo y se ha establecido una conexión online con este.

Borrado de Trace en el dispositivo

Para borrar un Trace del dispositivo, proceda del siguiente modo:

1. Seleccione una o varias filas que desee borrar en la tabla "Traces en dispositivo".
2. Pulse <Supr> para borrar los "Traces en dispositivo".

13.6.3.5 Tabla de señales

Utilización de la tabla de señales

La tabla de señales muestra las señales de un Trace en el dispositivo o de una medición. En la tabla es posible mostrar u ocultar cada una de las señales y ajustar propiedades para la visualización.
Para algunos tipos de datos se pueden seleccionar bits individuales y mostrarse como trazas de bits.

A continuación se describe el manejo de la tabla de señales.

Requisitos

- En las tablas "Traces en dispositivo" o "Mediciones" hay seleccionada una entrada en la columna .
- Para visualizar bits individuales como trazas de bits:
  Al menos una señal registrada soporta la visualización como traza de bit.

Selección de diferentes señales en la tabla de señales y modificación del formato

Para ajustar la visualización a sus necesidades, proceda del siguiente modo:

1. Haga clic en el símbolo de la señal correspondiente en la columna para seleccionarlo o deseleccionarlo en la visualización.
2. Para modificar el color predeterminado de la señal, haga clic en la columna "Color" para la señal correspondiente.
3. Haga clic en la entrada "En primer plano" en el menú contextual de una señal para visualizarla en primer plano.

Selección de bits individuales en la tabla de señales para visualizarlos como traza de bits

Para mostrar bits individuales como traza de bits en el diagrama de curvas inferior, proceda del siguiente modo:

1. Haga clic en el símbolo de una señal en la tabla de señales.
2. Haga clic en el símbolo de la selección de bits abierta de la señal para seleccionar o deseleccionar bits que desee visualizar.

13.6.3.6 Diagrama de curvas

Utilización del diagrama de curvas

El diagrama de curvas muestra las señales de un registro seleccionadas en la tabla de señales. El registro se selecciona en las tablas "Traces en dispositivo" o "Mediciones" de la administración de Trace.

Es posible aplicar zoom al área de representación libremente. Mediante cursores de medición pueden seleccionarse diferentes valores para visualizarlos en la tabla de señales.

A continuación se describe el uso de los cursores de medición.

Requisitos

Se ha seleccionado un registro para la visualización.

Evaluación de un momento concreto de un registro

Para visualizar los valores de un punto de medición concreto, proceda del siguiente modo:

1. Muestre los cursores de medición verticales con el botón .
2. Desplace con el ratón un cursor de medición a la posición deseada del registro.
   Los valores de las señales se muestran en la tabla de señales. El momento o el punto de medición hasta los cursores de medición se muestran en el área inferior del diagrama de curvas.

Evaluación de la diferencia entre dos puntos de medición

Para visualizar la diferencia, proceda del siguiente modo:

1. Muestre los cursores de medición verticales con el botón .
2. Desplace con el ratón los dos cursores de medición a los puntos de medición deseados del registro.
   Los valores de las señales y la diferencia se muestran en la tabla de señales. El momento o el punto de medición hasta los cursores de medición se muestran en el área inferior del diagrama de curvas.

Utilización de los cursores de medición horizontales

Para comprobar si se alcanza un determinado valor, proceda del siguiente modo:

1. Muestre los cursores de medición horizontales con el botón .
2. Desplace con el ratón un cursor de medición al valor deseado del registro.
   Los valores del cursor de medición para la señal seleccionada se muestran en el área inferior del diagrama de curvas.

Impresión de un registro

El diagrama de curvas permite guardar la visualización como mapa de bits y copiarla en el portapapeles. Utilice también estas funciones (Página 11332) para imprimir.

13.6.4 Dispositivos

13.6.4.1 CPU S7-1200/1500

Captación de variables

Captación de variables en función de dispositivo

A continuación se enumeran las áreas de operandos de las que pueden registrarse variables:

- Memoria imagen de proceso de las entradas
- Memoria imagen de proceso de las salidas
- Marca
- Bloques de datos

Tipos de datos

Es posible registrar los tipos de datos elementales. La disponibilidad de los distintos tipos de datos depende del dispositivo utilizado.

La tabla siguiente indica los tipos de datos elementales:

Tipos de datos	Nota
Números binarios	
BOOL	-
Secuencias de bits	
BYTE	-
WORD	-
DWORD	-
LWORD	Nombre simbólico necesario
Enteros	
SINT	-
USINT	-
INT	-
UINT	-
DINT	-
UDINT	-
LINT	Nombre simbólico necesario
ULINT	Nombre simbólico necesario
Números en coma flotante	
REAL	-
LREAL	Nombre simbólico necesario

Vida útil de la configuración de Trace y de los valores registrados en el dispositivo

Las configuraciones de Trace se conservan en caso de Power Off. Tras el rearranque de la CPU, el registro se activa de nuevo.

Los valores registrados se pierden con el rearranque.

Nota

Cargar una configuración en el dispositivo en el estado operativo "STOP"

Tras cargar una configuración en el estado operativo "STOP", tenga en cuenta que deberá comprobar los Traces en el dispositivo y que tal vez tenga que volverlos a activar o a transferir.

Nota

En caso de modificaciones de señales o de variables de disparo que repercuten en la dirección, es preciso volver a transferir también la configuración de Trace al dispositivo.

Este es el caso, p. ej. cuando se ha acortado o alargado un bloque de datos, o si se ha modificado el tipo de datos.

Niveles de registro

A continuación se enumeran los niveles de ejecución que pueden seleccionarse para el ciclo de registro:

- Ciclo de programa - OB 1
- Alarma horaria - OB 1x
- Alarma de retardo - OB 2x
- Alarma cíclica - OB 3x
- Ciclos de ejecución sincronizados - OB 6x, no OB 60
- Task de servo - OB 91 (S7-1500)
- IPO - OB 92 (S7-1500)

Nota

Las medidas se registran cada vez al final del OB tras la ejecución del programa del usuario.

Capacidad funcional

La tabla siguiente muestra la capacidad funcional máxima que puede registrarse con la función Trace y de analizador lógico.

Dispositivo	Número máximo de "Traces en dispositivo"	Número máximo de señales por configuración de Trace
S7-1200 (a partir de versión de firmware V4.0)	2	16
S7-1500	al menos 4 (en función del tipo de CPU)	16

Ejemplo S7-1516

- Máximo 3854 puntos de medición con 16 variables Long
- Máximo 21844 puntos de medición con 16 variables binarias
- Máximo 58250 puntos de medición con una variable binaria

Consulte más detalles en el manual de producto correspondiente.

Carga de CPU por registro de Trace

Un registro de Trace aumenta el tiempo de ejecución del nivel de registro correspondiente, lo que puede provocar un rebase del nivel de ejecución en caso de carga elevada de la CPU.

Solución

- Modificación de la configuración de Trace
  1) Configure menos variables y señales.
  2) Aumente a continuación el número de variables y señales gradualmente hasta alcanzar el número de señales máximo sin rebase del nivel de ejecución.
- Selección de un nivel de registro más lento

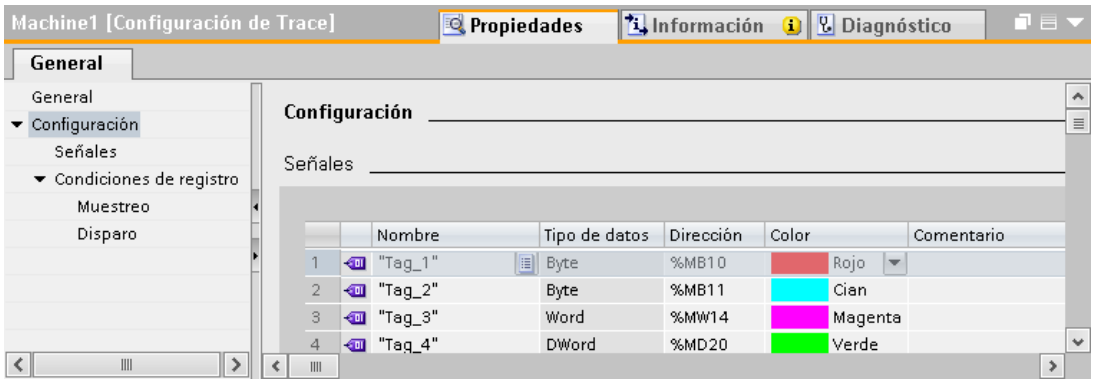
Interfaz de software del área específica de dispositivo

Estructura de la interfaz

Áreas de visualización de la ventana de propiedades de inspección

Las posibilidades de ajuste varían en función del dispositivo configurado.

La figura siguiente muestra un ejemplo de representación en el TIA Portal:



La navegación local ofrece las siguientes entradas para la selección:

- General
- Configuración
  - Señales (Página 11362)
  - Condiciones de registro (Página 11363)

Visualización de las propiedades de una configuración de Trace

La configuración de Trace se selecciona en la administración de Trace:

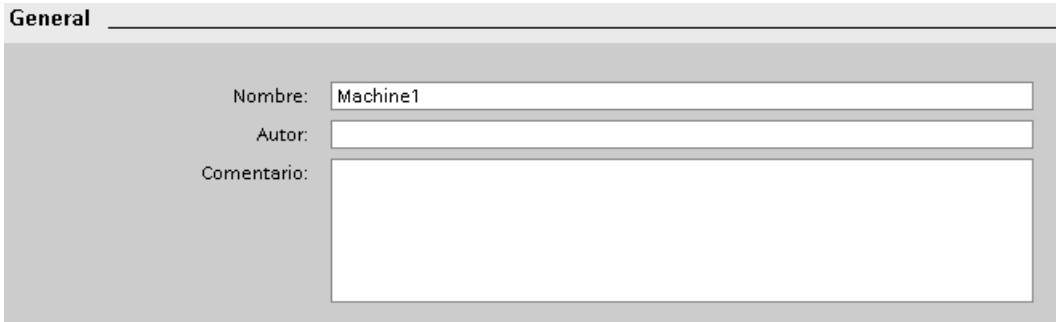
- Selección en la tabla "Configuraciones de Trace"
  Las propiedades se muestran en la ventana de inspección, donde también pueden modificarse.
- Selección en la tabla "Trace en dispositivo" o "Mediciones"
  Las propiedades se protegen contra escritura en la ventana de inspección.

Interfaz General

El área "General" muestra el nombre de la configuración de Trace y campos de entrada para el autor y un comentario.

Posibilidades de introducción de datos y visualizaciones en General

La figura siguiente muestra un ejemplo de representación en el TIA Portal:



La tabla siguiente muestra los campos de entrada y las visualizaciones:

Columna	Símbolo	Descripción
"Nombre"	-	Nombre de la configuración de Trace
"Autor"	-	Autor de la configuración de Trace
"Comentario"	-	Campo de entrada para un comentario

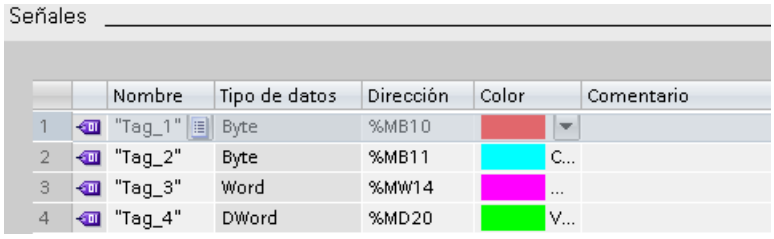
Interfaz Señales

El área "Señales" muestra una tabla en la que se configuran las señales que deben registrarse para la configuración de Trace seleccionada.

Las señales también se pueden insertar en la tabla mediante Drag & Drop.
Las señales se pueden ordenar mediante Drag & Drop.

Posibilidades de ajuste y visualizaciones en "Señales"

La figura siguiente muestra un ejemplo de representación en el TIA Portal:



La tabla siguiente muestra los ajustes y las visualizaciones:

Columna	Símbolo	Descripción
		Visualización del símbolo de señal para una señal seleccionada
"Nombre"	-	Campo de entrada para el nombre o la dirección de la señal **Ejemplos:** • M0.0 • DB1.DBW3 • "Bloque_datos_1".Presión
-		Botón para abrir la tabla de selección de señal El botón se muestra con la selección de las filas de la tabla. Haciendo clic en el símbolo se abre una tabla que ofrece señales que pueden seleccionarse. La señal seleccionada se muestra en el campo de entrada.
"Tipo de datos"	-	Campo de texto con la indicación del tipo de datos de la señal
"Dirección"	-	Campo de entrada para la dirección de la señal En las señales puramente simbólicas, el campo permanece vacío.
"Color"	-	Campo de texto para la indicación y selección del color Haga clic en el campo seleccionado para visualizar el color de la señal. Haga clic en el color de la señal para abrir el diálogo de selección de color.
"Comentario"		Campo de entrada para un comentario sobre la señal

Comandos del menú contextual

La tabla siguiente muestra los comandos del menú contextual de la tabla:

Comando del menú contextual	Descripción
"Cortar"	No seleccionable
"Copiar"	Copia el contenido de las filas seleccionadas en el portapapeles.
"Insertar"	Inserta el contenido del portapapeles a partir de la fila seleccionada. El contenido existente se sobrescribe.
"Borrar"	Borra las filas seleccionadas de la tabla o borra el nombre de la celda seleccionada.
"Cambiar nombre"	Pasa la celda seleccionada al modo de edición.

Condiciones de registro

Tipos de datos admitidos

La tabla siguiente muestra los tipos de datos admitidos para la variable de disparo:

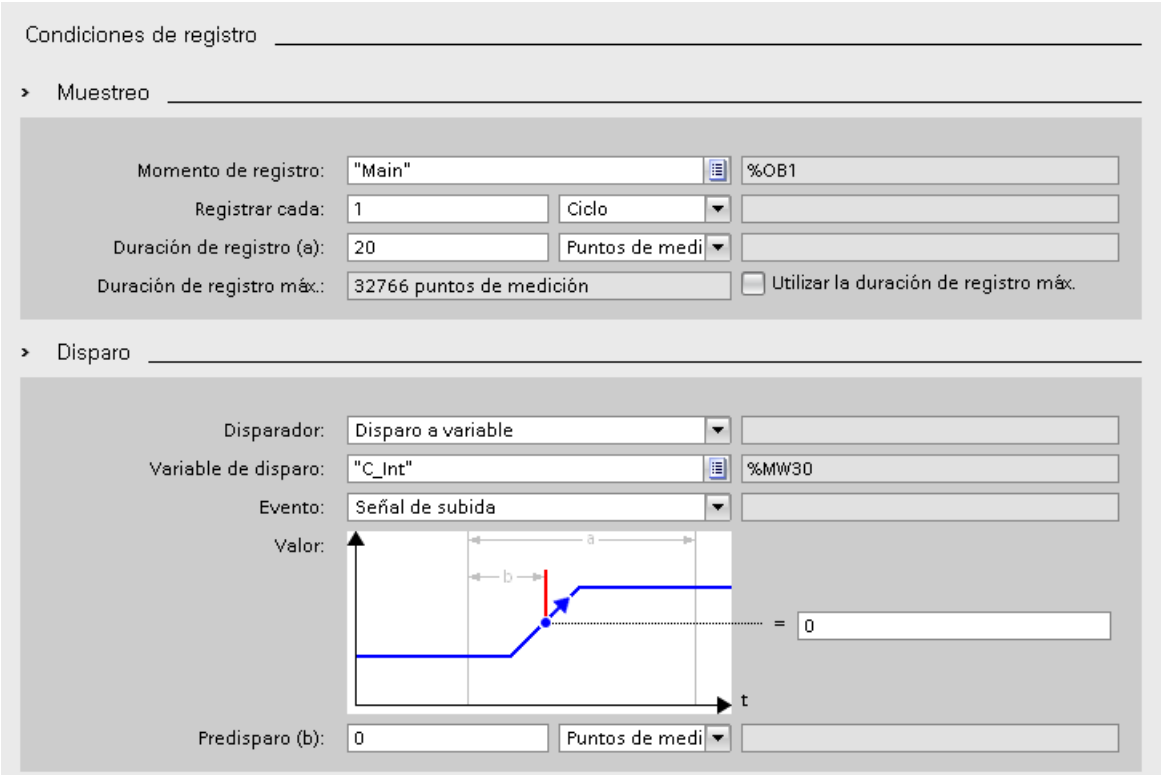
Memoria necesaria y formato del número	Tipo de datos
1 byte	BOOL
Enteros de 8 bits	SINT, USINT, BYTE
Enteros de 16 bits	INT, UINT, WORD
Enteros de 32 bits	DINT, UDINT, DWORD
Enteros de 64 bits	LINT, ULINT, LWORD (depende del dispositivo)
Números en coma flotante de 32 bits	REAL
Números en coma flotante de 64 bits	LREAL

Interfaz Condiciones de registro

El área "Condiciones de registro" muestra la condición de disparo para la configuración de Trace seleccionada y en qué ciclo, con qué rapidez y durante cuánto tiempo se registra. La configuración solo es posible si hay seleccionada una configuración de Trace en la tabla "Configuraciones de Trace" de la administración de Trace.

Posibilidades de ajuste y visualizaciones en "Condiciones de registro"

La figura siguiente muestra un ejemplo de representación en el TIA Portal:



Ajuste/visualización		Descripción
"Momento de registro"		
	Lista desplegable	Selección del momento de registro Ver Niveles de registro (Página 11355)
	Campo de texto	Información detallada sobre el momento de registro seleccionado
"Registrar cada"		
	Campo de entrada	Introducción de la reducción referida a los ciclos
	Lista desplegable	Selección del factor de reducción Pueden encontrarse los siguientes ajustes: • "Ciclo"
	Campo de texto	Visualización del intervalo de muestreo en función de la reducción configurada
"Duración de registro"		
	Campo de entrada	Introducción de la duración de registro referida a los puntos de medición Si la casilla de verificación de "Duración de registro = Duración de registro máx." está activada, el valor mostrado en "Duración de registro máx." sobrescribe las entradas.
	Lista desplegable	Selección de la unidad de tiempo para la duración de registro Pueden encontrarse los siguientes ajustes: • "Puntos de medición"

Ajuste/visualización		Descripción
	Campo de texto	Visualización de la duración de registro calculada
"Duración de registro máx."		
	Campo de texto	Visualización de la duración de registro máxima calculada La "Duración de registro máx." depende del número de señales que se registran y del tipo de datos que tienen estas señales.
	"Utilizar la duración de registro máx."	Ajuste de la duración de registro a la duración de registro máxima Al activar la casilla de verificación, la duración de registro se ajusta a la duración de registro máxima posible. Se tiene en cuenta el factor de reducción ajustado en el campo de entrada "Registrar cada". La duración de registro también se ajusta cuando se añaden otras señales.
"Disparador"		Selección del disparador
	Lista desplegable	Pueden encontrarse los siguientes ajustes: • "Registro inmediato" El registro se efectúa inmediatamente después de la activación en el dispositivo. • "Disparo en variable" El registro se efectúa en cuanto se activa el Trace en el dispositivo y se cumple la condición de disparo configurada.
	Campo de texto	Visualización del tipo de disparo
"Variable de disparo"		Con la "Variable de disparo" se especifica una señal que se utiliza para disparar el registro.
	Campo de entrada	Introducción de una señal **Ejemplos:** • M0.0 • DB1.DBW3 • "Bloque_datos_1".Temperatura
		Abrir la tabla de selección de señal Haciendo clic en el símbolo se abre una tabla que ofrece señales que pueden seleccionarse como variable de disparo. La señal seleccionada se muestra en el campo de entrada.
	Campo de texto	Visualización de la dirección de la variable de disparo
"Evento"		Conforme al tipo de datos de la variable de disparo, se ofrecen para su selección los eventos utilizables con esta variable de disparo. La configuración del evento solo es posible si se ha introducido una señal válida como variable de disparo.
	Lista desplegable	Selección de evento con el que se comprueba la variable de disparo Las entradas de la lista desplegable se describen en el capítulo Evento de disparo (Página 11366).
	Campo de texto	Visualización del tipo de comparación
"Valor"		Configuración del evento seleccionado Las posibilidades de configuración varían en función del formato de la variable de disparo y del evento seleccionado. Ver Evento de disparo (Página 11366).

Ajuste/visualización		Descripción
"Predisparo"		Con el "Predisparo" se define el intervalo de tiempo en el que se registran las señales antes de que se cumpla la condición de disparo propiamente dicha. Si el evento de disparo tiene lugar de inmediato o poco después de la activación del registro, la duración de registro puede ser menor. Ejemplo de "Duración de registro" = 20 puntos de medición y "Predisparo" = 5 puntos de medición: • El evento de disparo tiene lugar 50 puntos de medición tras la activación del registro. Duración de registro real = 20 puntos de medición • El evento de disparo tiene lugar 2 puntos de medición tras la activación del registro. Duración de registro real = 17 puntos de medición
	Campo de entrada	Introducción de la duración referida a la selección de la lista desplegable
	Lista desplegable	Selección de la unidad de tiempo Pueden encontrarse los siguientes ajustes: • "Puntos de medición"
	Campo de texto	Visualización del intervalo de tiempo "Predisparo" calculado

Evento de disparo

En función de la selección de la lista desplegable, se distinguen los siguientes ajustes para "Evento".

A continuación se describen los diferentes eventos.

"=TRUE"

Tipos de datos admitidos: Bit (Página 11359)

El registro se inicia con el estado TRUE del disparador.

"=FALSE"

Tipos de datos admitidos: Bit (Página 11359)

El registro se inicia con el estado FALSE del disparador.

"Flanco ascendente"

Tipos de datos admitidos: Bit (Página 11359)

El registro se inicia cuando el disparador cambia de estado FALSE a TRUE.
Tras activar el Trace en el dispositivo, son necesarios al menos dos ciclos para que se detecte el flanco.

"Señal de subida"

Tipos de datos admitidos: Números enteros y números en coma flotante (Página 11359)

El registro se inicia cuando el valor ascendente del disparador alcanza o excede el valor configurado para este evento.
Tras activar el Trace en el dispositivo, son necesarios al menos dos ciclos para que se detecte el flanco.

"Flanco descendente"

Tipos de datos admitidos: Bit (Página 11359)

El registro se inicia cuando el disparador cambia de estado TRUE a FALSE.
Tras activar el Trace en el dispositivo, son necesarios al menos dos ciclos para que se detecte el flanco.

"Señal de bajada"

Tipos de datos admitidos: Números enteros y números en coma flotante (Página 11359)

El registro se inicia cuando el valor descendente del disparador alcanza o está por debajo del valor configurado para este evento.
Tras activar el Trace en el dispositivo, son necesarios al menos dos ciclos para que se detecte el flanco.

"En rango"

Tipos de datos admitidos: Números enteros y números en coma flotante (Página 11359)

El registro se inicia en cuanto el valor del disparador se encuentra dentro del rango de valores configurado para este evento.

"Fuera del rango"

Tipos de datos admitidos: Números enteros y números en coma flotante (Página 11359)

El registro se inicia en cuanto el valor del disparador se encuentra fuera del rango de valores configurado para este evento.

"= valor"

Tipos de datos admitidos: Enteros (Página 11359)

El registro se inicia cuando el valor del disparador sea igual al valor configurado para este evento.

"<> valor"

Tipos de datos admitidos: Enteros (Página 11359)

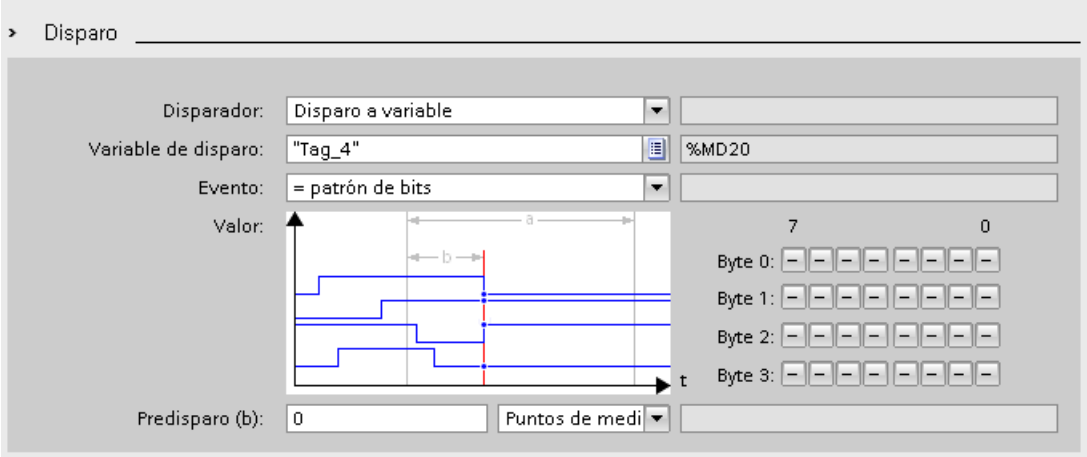
El registro se inicia cuando el valor del disparador sea diferente al valor configurado para este evento.

"= patrón de bits"

Tipos de datos admitidos: Enteros (Página 11359)

El registro se inicia cuando el valor del disparador coincida con el patrón de bits configurado para este evento.

La figura siguiente muestra las posibilidades de ajuste de un "patrón de bits":



Haciendo clic en el botón correspondiente, puede cambiarse entre los distintos símbolos.

La tabla siguiente muestra los símbolos:

Símbolo	Descripción
–	El bit no se evalúa.
0	El bit se comprueba buscando FALSE.
1	El bit se comprueba buscando TRUE.

"<> patrón de bits"

Tipos de datos admitidos: Enteros (Página 11359)

El registro se inicia cuando el valor del disparador no coincida con el patrón de bits configurado para este evento.

Consulte también

Configuración de las condiciones de disparo (Página 11370)

Configuración

Configuración de Trace: vista general

La configuración de las condiciones de registro y las señales que deben registrarse es específica de dispositivo.

Requisitos

Se ha creado una configuración de Trace y se ha seleccionado en la tabla "Configuraciones de Trace" de la administración de Trace.

Procedimiento

La tabla siguiente muestra el procedimiento de configuración.

Etapa	Descripción
1	Documentación de la configuración (opcional) Introduzca un comentario y un autor para la configuración.
2	Selección de señales (Página 11369) Seleccione las señales que deben registrarse en el área "Señales".
3	Configuración del ciclo y la duración de registro Seleccione un momento de registro, un ciclo y la duración en el área "Condiciones de registro".
4	Configuración de las condiciones de disparo (Página 11370) Especifique en el área "Condiciones de registro" si el registro se efectúa de forma inmediata o conforme a una condición de disparo.

Selección de señales

Requisitos

- Se ha creado una configuración de Trace y se ha seleccionado en la tabla "Configuraciones de Trace" de la administración de Trace.
- El área "Señales" está abierta en la ventana de inspección.

Procedimiento

Para configurar las señales que deben registrarse, proceda del siguiente modo:

1. Haga clic en la primera fila vacía de la tabla.
2. En la columna "Nombre", haga clic en la primera línea vacía.
3. Seleccione una señal. Existen las posibilidades siguientes:
   - En la columna "Nombre", haga clic en el botón y seleccione una variable.
   - En la columna "Nombre", introduzca el nombre simbólico de la variable en la celda.
   - Introduzca directamente la dirección en la columna "Dirección".
   - Arrastre una señal hasta la tabla mediante Drag & Drop.
4. Haga clic en la columna "Color de visualización" y seleccione un color para representar la señal.

5. Haga clic en la columna "Comentario" e introduzca un comentario para la señal.
6. Repita las operaciones a partir del paso 1 hasta que todas las señales que deben registrarse estén introducidas en la tabla.

Configuración del ciclo y la duración de registro

Requisitos

- Se ha creado una configuración de Trace y se ha seleccionado en la tabla "Configuraciones de Trace" de la administración de Trace.
- El área "Condiciones de registro" está abierta en la ventana de inspección.

Procedimiento

Para configurar el ciclo y la duración de un registro, proceda del siguiente modo:

1. Haga clic en el botón para el momento de registro.
2. Seleccione un OB para el momento de registro (Página 11355).
3. Seleccione una unidad para el factor de reducción en la lista desplegable de "Registrar cada".
4. Introduzca el factor de reducción en el campo de entrada de "Registrar cada".
5. Seleccione una unidad en la lista desplegable "Duración de registro".
6. Defina la duración de registro.
   Existen las posibilidades siguientes:
   – Introduzca un valor para la duración en el campo de entrada "Duración de registro".
   – Active la casilla de verificación "Utilizar la duración de registro máx."

Configuración de las condiciones de disparo

Requisitos

- Se ha creado una configuración de Trace y se ha seleccionado en la tabla "Configuraciones de Trace" de la administración de Trace.
- El área "Condiciones de registro" está abierta en la ventana de inspección.

Condición de disparo "Registro inmediato"

Para iniciar el registro de forma inmediata, proceda del siguiente modo:

1. Seleccione la entrada "Registro inmediato" en la lista desplegable "Disparador". Los campos de entrada para la variable de disparo se ocultan.

Condición de disparo "Disparo en variable"

Para iniciar el registro conforme a una condición, proceda del siguiente modo:

1. Seleccione la entrada "Disparo en variable" en la lista desplegable "Disparador".
2. Seleccione una variable de disparo. Existen las posibilidades siguientes:
   – Haga clic en el botón para la variable de disparo y seleccione una variable.
   – Introduzca directamente la dirección o el nombre simbólico de la variable en el campo de entrada de la variable de disparo.

   Se muestra una lista desplegable con eventos y campos de entrada. La visualización depende del tipo de datos de la variable.
3. Configure el evento.
4. Seleccione una unidad del predisparo en la lista desplegable de "Predisparo".
5. Para registrar un intervalo de tiempo antes del evento de disparo, introduzca un valor superior a 0 en el campo de entrada para el predisparo.

Nota

La condición de disparo se comprueba en cada ciclo independientemente del ajuste de "Registrar cada". Para la detección segura del disparo, la señal de disparo debe estar pendiente al menos un ciclo completo.

13.7 Establecer una conexión remota con TeleService

13.7.1 Principios básicos para el trabajo con TeleService

13.7.1.1 Introducción a TeleService

Introducción

Gracias a TeleService el controlador puede comunicarse a distancia. Ello permite gestionar, controlar y supervisar de forma centralizada instalaciones descentralizadas a través de conexiones remotas.

Funcionalidad

TeleService permite utilizar la funcionalidad del TIA Portal a través de una red telefónica o una conexión a Internet estableciendo una conexión remota con una instalación remota. A través de la conexión online es posible trabajar también con una instalación remota de la forma habitual con el TIA Portal.

Ventajas

El uso de TeleService proporciona las siguientes ventajas:

- Así también se puede acceder cómodamente a partes de instalaciones remotas e integrarlas en un sistema más amplio.
- En caso de fallos en una instalación remota, pueden ofrecerse ayuda y soporte con rapidez sin tener que estar presente.
- Uso eficaz de los recursos.
- Los costes se reducen considerablemente.
- Los tiempos improductivos de las instalaciones pueden reducirse sustancialmente.
- Aumenta la rentabilidad de su instalación.

Consulte también

Funcionalidad de TeleService (Página 11372)

13.7.1.2 Funcionalidad de TeleService

Posibles aplicaciones de TeleService

TeleService le ofrece las siguientes aplicaciones posibles:

- **Acceder a instalaciones remotas (asistencia técnica a distancia):**
  Las instalaciones descentralizadas se pueden gestionar, controlar y supervisar de forma centralizada a través de conexiones remotas.
  Esto es posible con una CPU S7-300/400, una CPU S7-1200 y una CPU S7-1500 y, en cada caso, con un TS Adapter MPI o un TS Adapter IE.
- **Establecer conexiones desde y hacia instalaciones remotas (acoplamiento remoto PG-PLC):**
  Con ayuda de PRODAVE MPI V5.0 y versiones superiores se puede establecer una conexión a distancia hacia una instalación remota y, con la instrucción de comunicación "PG_DIAL", se puede establecer una conexión a distancia desde una instalación remota.
  Esto es posible con una CPU S7-300/400 y un TS Adapter MPI.
- **Intercambiar datos entre instalaciones (acoplamiento remoto PLC-PLC):**
  Mediante la instrucción de comunicación "AS_DIAL", dos sistemas de automatización pueden intercambiar datos de proceso a través de la red telefónica.
  Esto es posible con una CPU S7-300/400 y un TS Adapter MPI.

- **Enviar un mensaje SMS desde una instalación:**
  Un sistema de automatización puede enviar un mensaje (SMS) a través de un módem inalámbrico GSM con la instrucción de comunicación "SMS_SEND".
  Esto es posible con una CPU S7-300/400 y un TS Adapter MPI.
- **Enviar un e-mail desde una instalación**
  Un sistema de automatización también puede enviar un correo electrónico utilizando las siguientes instrucciones de comunicación y un TS Adapter IE .
  - Las CPU S7-300/400 (CPU S7-31x-2PN/DP o CPU 41x-3PN/D) utilizan la instrucción "AS_MAIL"
  - Las CPU S7-1200 utilizan la instrucción "TM_MAIL"
  - Las CPU S7-1500 utilizan la instrucción "TMAIL_C"

13.7.1.3 Listín telefónico en TeleService

Introducción

Haciendo doble clic en la carpeta "Listín telefónico" en el árbol del proyecto, se abre el editor del listín telefónico, que muestra el Listín telefónico de TeleService.

Cada versión del TIA Portal dispone de un "listín telefónico global" propio. Si en una nueva versión del TIA Portal se encuentra un listín telefónico global procedente de una versión anterior del TIA Portal, se le consultará una única vez si desea importar este listín telefónico.

Esto tiene la ventaja de que en la nueva versión del TIA Portal también dispondrá de los datos de instalación de la versión anterior.

Propiedades del listín telefónico global

El listín telefónico global sirve en TeleService para gestionar los datos de instalación específicos que son necesarios para establecer una conexión remota.

Al abrir el listín telefónico por primera vez, se muestra un listín telefónico vacío con todas las columnas existentes; en caso contrario, se muestra el último listín telefónico editado.

En un listín telefónico puede introducirse el número de instalaciones que se desee. Las instalaciones contienen los datos necesarios para establecer una conexión remota, p. ej., el nombre y la ubicación del nodo, así como el número de teléfono que se debe marcar con los detalles específicos del país. En las conexiones VPN se puede introducir una dirección IP o un nombre DNS en vez de un número de teléfono.

Los TS Adapter utilizados para establecer la conexión se muestran en colores distintos en función de si se emplea un TS Adapter MPI o un TS Adapter IE para establecer la conexión.

Consulte también

Trabajar con el listín telefónico (Página 11374)

13.7.2 Trabajar con el listín telefónico

13.7.2.1 Principios básicos para trabajar con el listín telefónico

Trabajar con el listín telefónico

Se dispone de las siguientes posibilidades para trabajar con un listín telefónico:

- Abrir el listín telefónico
- Guardar el listín telefónico
- Importar datos del listín telefónico
- Exportar datos del listín telefónico
- Imprimir datos del listín telefónico
- Utilizar datos del listín telefónico para establecer una conexión remota

Puede implementar estas funciones cómodamente mediante los botones que se muestran en la barra de herramientas del listín telefónico.

Nota

Acceso a listines telefónicos

El listín telefónico se guarda en TeleService de forma específica para cada usuario. Sin embargo, no es posible acceder al listín telefónico global con más de una instancia del TIA Portal al mismo tiempo.

Consulte también

Abrir el listín telefónico (Página 11376)

Guardar el listín telefónico (Página 11377)

Exportar datos del listín telefónico (Página 11379)

Imprimir el listín telefónico (Página 11380)

Configuración del listín telefónico (Página 11374)

13.7.2.2 Configuración del listín telefónico

Introducción

En TeleService, un listín telefónico global sirve para gestionar los datos necesarios para establecer una conexión remota. Si ya se han creado una vez los datos de la conexión y se han almacenado en el listín telefónico, puede volver a acceder a ellos para establecer una conexión remota.

Configuración del listín telefónico

El listín telefónico global integrado en TeleService contiene las siguientes columnas:

Nombre de columna	Explicación
Nombre de la instalación	Introduzca aquí el nombre que desee para su instalación.
Tipo de adaptador	Seleccione de la lista desplegable el tipo de adaptador utilizado: TS Adapter MPI o TS Adapter IE.
Tipo de conexión	Seleccione el tipo de conexión deseado: conexión por línea telefónica o conexión VPN.
Código local	Introduzca aquí el código local deseado. Esta columna solo está activada en las conexiones por línea telefónica. En las conexiones VPN, esta columna no se puede editar.
Número de teléfono/Dirección remota	Introduzca aquí los datos de conexión que desee para establecer la conexión remota. Estos datos pueden ser un número de teléfono para las conexiones por línea telefónica y un nombre DNS o una dirección IP para las conexiones VPN.
Huella digital	Introduzca aquí la huella digital correspondiente para establecer una conexión VPN con el TS Adapter IE Advanced.
País	Introduzca el prefijo correspondiente al país. Esta columna solo está activada en las conexiones por línea telefónica. En las conexiones VPN, esta columna no se puede editar.
Nombre de usuario	Introduzca el nombre de usuario con el que se ha registrado.
Contraseña	Introduzca aquí la contraseña correspondiente al nombre de usuario.
Grupo	Introduzca aquí el grupo correspondiente en caso de que haya realizado una agrupación.
Empresa	Introduzca la empresa con la que hay que establecer contacto.
Departamento	Introduzca el departamento correspondiente.
Calle	Introduzca la calle.
Localidad	Introduzca la localidad con la que debe establecerse la conexión remota.
Comentario	Si lo desea, introduzca un comentario.

Mostrar u ocultar columnas

Las columnas se pueden mostrar u ocultar individualmente. Para ello, seleccione el título de la columna deseada y abra el menú contextual con el botón derecho del ratón.

13.7.2.3 Símbolos del listín telefónico

Significado de los símbolos en TeleService

La tabla siguiente muestra el significado de los símbolos en TeleService:

Símbolo	Significado
	Abre el listín telefónico global
	Importa un listín telefónico
	Exporta un listín telefónico

Símbolo	Significado
	Establece una conexión remota
	Desconecta la conexión remota activa
	Establece una conexión remota o la desconecta
	Muestra la conexión a un TS Adapter IE en el listín telefónico
	Muestra la conexión a un TS Adapter MPI en el listín telefónico
	Agrega una nueva fila al listín telefónico
	Inserta una nueva fila en el listín telefónico

13.7.2.4 Administrar el listín telefónico

Abrir el listín telefónico

Abrir listines telefónicos

Para abrir el listín telefónico, proceda del siguiente modo:

1. En el árbol del proyecto, haga doble clic en la carpeta "Listín telefónico" bajo "Accesos online" > "TeleService".
2. El listín telefónico se abre y pueden introducirse o editarse los datos de instalaciones deseados.

Insertar filas en el listín telefónico

Insertar filas en el listín telefónico

Para insertar una nueva fila en el listín telefónico, proceda del siguiente modo:

1. Seleccione la fila antes de la cual desee insertar nueva una fila.
2. Haga clic en el botón "Insertar fila" de la barra de herramientas.

Resultado

Se inserta una nueva fila en el listín telefónico antes de la seleccionada.

Mostrar y ocultar columnas en el listín telefónico

Mostrar y ocultar columnas

Para mostrar u ocultar columnas en el listín telefónico, proceda del siguiente modo:

1. Haga clic en el encabezado de una columna.
2. Elija el comando "Mostrar/ocultar columnas" del menú contextual.
   Aparece una selección de las columnas disponibles.
3. Para mostrar una columna, active la casilla de verificación correspondiente.
4. Para ocultar una columna, desactive la casilla de verificación correspondiente.

Resultado

Las columnas en cuestión se muestran o se ocultan en la vista del listín telefónico.

Guardar el listín telefónico

Guardar listines telefónicos

Al salir del editor de listines telefónicos o del TIA Portal, se le preguntará si desea guardar el listín telefónico global.

Para guardar el listín telefónico haga clic en "Sí".

Importar datos del listín telefónico

Introducción

Existe la posibilidad de importar los datos del listín telefónico de un archivo externo o de una versión anterior del TIA Portal.

Requisitos

Ha creado un archivo de listín telefónico importable.

Ha creado un listín telefónico con una versión anterior del TIA Portal.

Importar datos del listín telefónico del archivo de listín telefónico

Para importar datos del listín telefónico de un archivo de listín telefónico, proceda de siguiente modo:

1. Abra la carpeta "TeleService" en "Accesos online" en el árbol del proyecto.
2. Haga doble clic en la carpeta "Listín telefónico".

3. En el listín telefónico, haga clic en el botón "Importar" de la barra de herramientas.
4. En caso necesario, confirme con "Sí" la consulta sobre si debe guardarse el estado actual del listín telefónico de trabajo y seleccione en el cuadro de diálogo siguiente dónde debe guardarse el listín telefónico.
5. Si no desea guardar el estado actual del listín telefónico de trabajo, seleccione "No" en la consulta. Seleccione en el cuadro de diálogo siguiente qué archivo de listín telefónico debe guardarse en el listín telefónico actual.
6. Cierre el cuadro de diálogo haciendo clic en "Aceptar".

Resultado

Los datos del listín telefónico importados se muestran en el listín telefónico global.

Nota

Definir reglas de marcación

Para poder editar las columnas "Código local" y "País" en el listín telefónico importado para conexiones por línea telefónica, deben definirse unas reglas de marcación.

Para definir las reglas de marcación, siga el enlace que se muestra en el histórico.

Importar datos del listín telefónico de una versión anterior del TIA Portal

Para importar datos del listín telefónico de una versión anterior del TIA Portal, proceda del siguiente modo:

1. Abra la carpeta "TeleService" en "Accesos online" en el árbol del proyecto.
2. Haga doble clic en la carpeta "Listín telefónico".
3. En el listín telefónico, haga clic en el botón "Importar" de la barra de herramientas.
4. En caso necesario, confirme con "Sí" la consulta sobre si debe guardarse el estado actual del listín telefónico de trabajo y seleccione en el cuadro de diálogo siguiente dónde debe guardarse el listín telefónico.
5. Si no desea guardar el estado actual del listín telefónico de trabajo, seleccione "No" en la consulta.
6. En el siguiente cuadro de diálogo introduzca esta ruta: "%appdata%\siemens\automation \TeleService\GlobalTeleServicePhoneBook.tel".
7. Cierre el cuadro de diálogo haciendo clic en "Aceptar".

Resultado

Los datos del listín telefónico importados de la versión anterior del TIA Portal se muestran en el listín telefónico global.

Nota

Definir reglas de marcación

Para poder editar las columnas "Código local" y "País" en el listín telefónico importado para conexiones por línea telefónica, deben definirse unas reglas de marcación.

Para definir las reglas de marcación, siga el enlace que se muestra en el histórico.

Consulte también

Definir reglas de marcación (Página 11380)

Exportar datos del listín telefónico

Introducción

Existe la posibilidad de exportar los datos del listín telefónico a un archivo externo.

Requisitos

Ha creado un listín telefónico con los datos de la instalación correspondientes en TeleService.

Procedimiento

Para exportar los datos del listín telefónico, proceda del siguiente modo:

1. Abra la carpeta "TeleService" del árbol del proyecto.
2. Haga doble clic en la carpeta "Listín telefónico".
3. En el listín telefónico, haga clic en el botón "Exportar" de la barra de herramientas.
4. Seleccione en el cuadro de diálogo siguiente a dónde debe exportarse el listín telefónico actual.
5. Cierre el cuadro de diálogo haciendo clic en "Aceptar".

Resultado

Los datos del listín telefónico exportados se guardan en el archivo de exportación indicado.

Imprimir el listín telefónico

Imprimir listines telefónicos

Puede imprimir todos los datos de un listín telefónico o sólo una determinada parte de un listín telefónico.

Proceda del siguiente modo:

1. Abra el listín telefónico.
2. Seleccione el comando **Listín telefónico > Imprimir** o haga clic en el botón correspondiente de la barra de herramientas. Se abre el cuadro de diálogo "Imprimir".
3. En este cuadro de diálogo indique si desea imprimir todo el listín telefónico o sólo una parte del mismo y ajuste todas las demás opciones.
4. Arranque el proceso de impresión pulsando "Aceptar".

Resultado

Los datos del listín telefónico se imprimen en la impresora predeterminada. Si la impresión abarca varias páginas, en el borde inferior derecho de la página se imprime el número de página seguido de una identificación de las páginas que siguen. La última página no tiene esta identificación.

Definir reglas de marcación

Procedimiento

Para definir reglas de marcación específicas para una ubicación, proceda del siguiente modo:

1. Abra la carpeta "Accesos online" del TIA Portal y seleccione la carpeta "TeleService".
2. Abra las propiedades de "TeleService" a través del menú contextual.
3. En el cuadro de diálogo siguiente, abra la ficha "Configuración avanzada" y active la opción "Utilizar reglas de marcación".
4. Haga clic en el botón "Adaptar".
5. Introduzca en el cuadro de diálogo siguiente la información sobre la ubicación que desee para establecer la conexión correspondiente.
   - Para las nuevas ubicaciones, indique el país o la región y el código local correspondiente.
   - En caso necesario, ajuste el código de red correcto y el prefijo de marcado para línea urbana para llamadas locales/llamadas de larga distancia.
   - Seleccione el método de marcado deseado para la ubicación.
6. Cierre este cuadro de diálogo haciendo clic en "Aceptar".

7. Seleccione en el cuadro de diálogo siguiente la ubicación desde la que debe realizarse el marcado y haga clic en "Aceptar".
8. Compruebe en el siguiente cuadro de diálogo de TeleService si se ha introducido la ubicación deseada en "Ubicación".
9. Confirme las entradas realizadas haciendo clic en "Aceptar".

Nota

Utilización del módem en una línea urbana o ubicación secundaria

Si utiliza el módem en una línea urbana (centralita), no necesita indicar ningún prefijo de marcado para línea urbana. Los campos del prefijo de marcado para línea urbana deben estar vacíos para llamadas locales y llamadas de larga distancia.
Si opera el módem en una ubicación secundaria, debe introducir el prefijo de marcado que debe elegirse para obtener una línea urbana.

Ejemplo de utilización de reglas de marcación

En el siguiente ejemplo se muestra la utilización de reglas de marcación específicas para una ubicación para la localidad de Karlsruhe, Alemania.

- Introduzca en el listín telefónico de TeleService para una instalación el número de teléfono "1234567", sin prefijo y sin identificador de país.
  Resultado:
  Siempre se marcará el número de teléfono "1234567", independientemente de la ubicación de la persona que llama.
- Introduzca en el listín telefónico de TeleService, además del número de teléfono "1234567", los parámetros de marcado específicos de la ubicación, que son el prefijo "0721" de Karlsruhe y el identificador de país "+49" para Alemania.
  Resultado:
  Desde Karlsruhe se marca el número de teléfono "1234567".
  Desde otras localidades dentro de Alemania se marca el número de teléfono "07211234567".
  Desde otro país se marca el número de teléfono "00497211234567".

Nota

Visualización en el listín telefónico

Las columnas "Código local" y "País" del listín telefónico de TeleService solo se pueden editar si se han definido reglas de marcación tal como se ha descrito antes.

13.7.3 Conexión remota como conexión por línea telefónica

13.7.3.1 Principios básicos para el establecimiento de una conexión por línea telefónica

Uso de un TS Adapter para conexiones por línea telefónica

Para poder establecer una conexión remota como conexión por línea telefónica con TeleService, es necesario utilizar un TS Adapter.

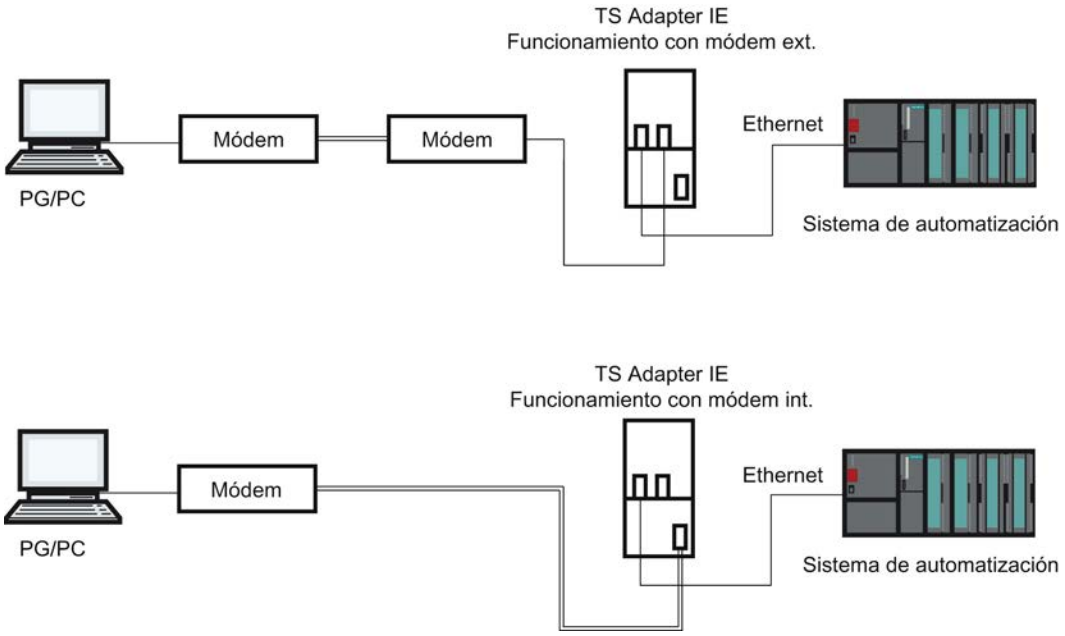
El TS Adapter se utiliza para conectar un sistema de automatización vía módem a la red telefónica y de este modo preparar el sistema para el uso de TeleService. Para ello, el TS Adapter dispone de una memoria de parámetros en la que se puede guardar un juego de parámetros para el uso de TeleService.

La función "Exportación de parámetros del adaptador" permite guardar distintos juegos de parámetros en archivos externos, mientras que con la función "Importación de parámetros del adaptador" pueden volver a cargarse en el TS Adapter .

Establecimiento de la conexión remota como conexión por línea telefónica

Puede elegir entre distintos TS Adapter que ofrecen en cada caso una funcionalidad diferente y distintas posibilidades de conexión.

La siguiente figura muestra dos configuraciones posibles para establecer una conexión por línea telefónica con una instalación mediante un TS Adapter IE.



Resumen de los TS Adapter que pueden utilizarse:

Existen los siguientes modelos de TS Adapter:

- TS Adapter II (también denominado "TS Adapter MPI")
- TS Adapter IE Standard (también denominado "TS Adapter IE")
- TS Adapter IE Basic (también denominado "TS Adapter IE")

Denominación "TS Adapter"

A continuación, se utiliza la denominación "TS Adapter" para todos los modelos. En los casos en los que solo se hace referencia a un modelo determinado, se especifica si se trata de "TS Adapter II" o "TS Adapter IE Standard".

Nota

Encontrará más información sobre el TS Adapter que está utilizando en la documentación que se le ha suministrado con su TS Adapter.

Consulte también

Breve descripción del TS Adapter MPI (Página 11391)

Breve descripción del TS Adapter IE (Página 11398)

Exportación de parámetros del adaptador (Página 11397)

Importación de parámetros del adaptador (Página 11398)

13.7.3.2 Redes telefónicas y módems

Redes telefónicas y módems permitidos

Redes telefónicas que pueden utilizarse

TeleService se puede utilizar con redes digitales (RDSI), redes analógicas y redes por radiofrecuencia (con tecnología GSM). La presente versión soporta una conexión remota a un TS Adapter.

Compatibilidad con módem

TeleService ha sido desarrollado independientemente del módem. Por ello, todos los módems de venta habitual en el mercado que se puedan instalar desde el Panel de control de Windows y que allí sean visibles como módem también pueden ser utilizados por TeleService.

El tipo de módem a elegir depende del hardware de la programadora / del PC y de la red telefónica utilizada.

Tipos de módems/medios compatibles:

- Módems (módems externos en el puerto COM, módems internos y tarjetas PCMCIA)
- Adaptador ISDN (RDSI) externo en el puerto COM o USB
- Adaptador ISDN (RDSI) interno con puerto COM virtual (p. ej., puerto AVM CAPI)
- Módems ISDN (RDSI) externos (adaptador ISDN (RDSI) con funcionalidad de módem analógico integrada) en el puerto COM o USB
- Módems inalámbricos con tecnología GSM, tarjeta adaptador PCMCIA o cable de datos y móvil

Routers

En principio es posible utilizar routers entre las distintas redes telefónicas. Las conexiones remotas de un adaptador ISDN (RDSI) a un módem analógico y a la inversa funcionan sólo con adaptadores telefónicos ISDN (RDSI) especiales.

Capacidad de las redes telefónicas

El flujo de datos de una conexión remota depende del módem utilizado, de la red telefónica utilizada y de la calidad de la línea telefónica.

Instalar el módem local

Introducción

Si el sistema operativo ya tiene instalado un módem para la transmisión de datos, puede utilizarse este módem también para TeleService.

Si el sistema operativo aún no reconoce ningún módem, debe instalarse un módem para establecer una conexión remota con TeleService.

Procedimiento

Proceda del siguiente modo:

1. Asegúrese de que tanto su programadora/PC como el módem estén desconectados.
2. Conecte físicamente el módem externo a una interfaz COM o USB de su programadora/PC. También puede instalar un módem interno o una tarjeta PCMCIA siguiendo las instrucciones del fabricante.
3. Ahora, conecte primero el módem externo y después la programadora o el PC.

Resultado

El sistema operativo reconoce e instala automáticamente los módems plug&play. Se le guiará por el proceso de instalación con ayuda de cuadros de diálogo.

Nota

Módems no aptos para plug&play

Si su módem no es detectado automáticamente al conectarlo, deberá instalarlo mediante el Panel de control.

Tenga en cuenta para ello las indicaciones incluidas en la documentación suministrada con su módem.

Conectar y parametrizar el módem remoto

Introducción

Una condición necesaria para el trabajo con TeleService es que a la instalación remota también esté conectado un módem. Este módem se denomina "módem remoto".

Parametrizar el módem remoto

El módem contiene todos los parámetros del TS Adapter conectado necesarios para el funcionamiento. Estos son, entre otros, los datos para la inicialización del módem y los ajustes para la transmisión en serie entre TS Adapter y el módem.

Los datos necesarios para el módem remoto se determinan durante la parametrización del TS Adapter.

En función del TS Adapter empleado, puede tratarse de un módem interno o externo.

Procedimiento para conectar un TS Adapter con un módem interno

1. Desconecte el TS Adapter.
2. Conecte el TS Adapter al sistema de automatización.
3. Conecte el TS Adapter a la conexión telefónica.
4. Encienda el TS Adapter.

Procedimiento para conectar un TS Adapter con un módem externo

1. Desconecte el módem.
2. Conecte el TS Adapter al sistema de automatización.
3. Conecte el cable de conexión del TS Adapter al módem.
4. Conecte el módem a la conexión telefónica.

5. Encienda el módem.
6. Encienda el TS Adapter.

Nota

Siga las siguientes indicaciones para parametrizar el módem remoto:

- Los parámetros para el módem y el puerto serie predeterminados en el TS Adapter deben garantizar un funcionamiento satisfactorio en la mayoría de los casos, de forma que sólo sea necesario realizar una reparametrización en casos excepcionales.
- No es necesario volver a parametrizar el TS Adapter a menos que se lleve a cabo la conexión al módem o que se desee adaptar u optimizar los ajustes de fábrica.
- El TS Adapter puede reparametrizarse mediante una conexión directa o una conexión remota.

13.7.3.3 Protección de acceso en las conexiones por línea telefónica

Información sobre los derechos de acceso

Introducción

Al parametrizar el TS Adapter pueden establecerse derechos de acceso tanto para la parametrización del propio TS Adapter como para las instalaciones remotas.

Validez de los derechos de acceso

Los derechos de acceso del TS Adapter MPI son válidos exclusivamente para conexiones remotas; en la conexión directa puede accederse en todo momento a la parametrización del TS Adapter.

En el TS Adapter IE existen derechos de acceso también en la conexión directa.

Información sobre los derechos de acceso

El TS Adapter MPI se suministra de fábrica con los derechos de acceso desactivados. En el TS Adapter IE existe una contraseña predeterminada.

El primer usuario que parametriza el adaptador puede activar los derechos de acceso definiendo la contraseña para un usuario o el número de rellamada.

Existen varios niveles de protección con varios usuarios, cada uno de ellos con o sin derechos de administrador. En el TS Adapter MPI sólo existen un administrador y como máximo dos usuarios.

Sólo el administrador puede crear otros usuarios, modificar los ajustes ya definidos y borrarlos cuando sea necesario. Al entrar como usuario sin derechos de administrador, sólo puede modificar su propia contraseña y su propio número de rellamada. Sin embargo, en la conexión directa del TS Adapter MPI se puede acceder a la parametrización del TS Adapter sin limitaciones.

Ventajas

Los derechos de acceso ofrecen las siguientes ventajas:

- Queda prácticamente excluida la posibilidad de que accedan personas no autorizadas ajenas a la empresa.
- El operador de la instalación asume la mayor parte de la tarifa telefónica.

Posibilidades de rellamada en TeleService

Variantes de rellamada

Por norma general, los costes de una conexión telefónica corren a cargo de la persona que llama, es decir, la persona que establece la conexión por línea telefónica.

No obstante, TeleService también puede utilizarse de forma que, tras una breve primera conexión, la conexión de módem vuelva a establecerse en sentido contrario, es decir, partiendo del TS Adapter (rellamada). En este caso, el usuario de la instalación corre con los gastos de la rellamada.

TeleService distingue entre dos variantes de rellamada:

1. Rellamada a un número indicado al establecer la conexión.
2. Rellamada a un número guardado en el TS Adapter.

Información sobre los niveles de protección

Introducción

Al configurar los derechos de acceso para el acceso a través de TeleService al TS Adapter existen dos niveles de protección que están vinculados a diferentes opciones.

Opciones para los derechos de acceso

Nivel de protección 1:

El TS Adapter está protegido mediante el nombre de usuario y la contraseña. Puede acceder al TS Adapter mediante las conexiones telefónicas que desee y, opcionalmente, indicar un número de rellamada cualquiera al establecer la conexión.

Nivel de protección 2:

El TS Adapter está protegido por el nombre de usuario, la contraseña y el número de rellamada. Sólo puede acceder al TS Adapter a través de una conexión telefónica por usuario.

La siguiente tabla muestra las condiciones antes indicadas para los diferentes niveles de protección:

Nivel de protección	Contraseña de administrador/usuario	Número de rellamada
1	Introducir	No introducir
2	Introducir	Introducir

Iniciar sesión en el TS Adapter

Al iniciar sesión en el TS Adapter, una vez que se hayan creado los derechos de acceso, se deberá introducir el nombre de usuario, la contraseña y, opcionalmente, un número de rellamada:

Nivel de protección	Contraseña de administrador/usuario	Número de rellamada
1	Introducir	No introducir o un número de rellamada cualquiera
2	Introducir	No introducir

Si al establecer la conexión se ha introducido un número de rellamada (nivel de protección 1) o se ha guardado un número de rellamada en el TS Adapter (nivel de protección 2), se deshace la conexión módem y el TS Adapter llama al teléfono indicado.

Crear derechos de acceso y número de rellamada para el TS Adapter

Introducción

En TeleService se pueden crear derechos de acceso y un número de rellamada para parametrizar el TS Adapter y conectarlo a la instalación remota. A continuación se describe la parametrización de un TS Adapter MPI. La parametrización de un TS Adapter IE se realiza de forma análoga. El procedimiento exacto se describe en la ayuda web del adaptador.

Requisitos

Un TS Adapter MPI está conectado a su equipo y se muestra en "Dispositivos accesibles" en el árbol del proyecto.

Procedimiento

Para crear los derechos de acceso para el TS Adapter, proceda del siguiente modo:

1. Haga clic en el comando "Parametrizar TS Adapter MPI" en el árbol del proyecto.
2. Abra la ficha "Protección de acceso".

3. Introduzca aquí una contraseña para su nombre de usuario o un número al que deba llamar el módem tras el inicio de sesión.
   - Si ha iniciado sesión como administrador, puede modificar todos los ajustes de administradores y usuarios, así como crear y borrar usuarios.
   - Si ha iniciado sesión como usuario, sólo puede modificar sus propios ajustes (contraseña y número de rellamada).
4. Confirme todas las entradas realizadas con "Aceptar" antes de salir del cuadro de diálogo.
5. Pulse el botón "Sí" para confirmar la siguiente consulta.

Resultado

La parametrización de los derechos de acceso y del número de rellamada se guarda en la memoria no volátil del TS Adapter MPI.

Nota

Al crear los derechos de acceso debe tenerse en cuenta obligatoriamente lo siguiente:

- Para garantizar la funcionalidad de rellamada debe asegurarse de que los ajustes de la ficha "Módem" correspondan a las características de la instalación.
- Si en el usuario "ADMIN" se introduce un número de rellamada erróneo, no se podrá volver a acceder al TS Adapter MPI a través de una conexión remota.
- Pruebe el número de rellamada antes de introducirlo en el usuario "ADMIN"; hágalo llamando a un número de rellamada indicado al establecer la conexión (nivel de protección 1).

Realizar la rellamada en TeleService

Posibilidades de rellamada

En TeleService existe la posibilidad de configurar dos variantes de rellamada diferentes.

Existen las siguientes posibilidades de rellamada:

- Rellamada a un número indicado al establecer la conexión.
- Rellamada a un número guardado en el TS Adapter.

Rellamada a un número indicado al establecer la conexión

1. Haga clic en la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Dentro de esta, haga clic en la carpeta "TeleService".
3. Haga doble clic en la entrada "Establecer/deshacer conexión a distancia". Se abre el cuadro de diálogo "Establecer conexión a distancia con la instalación remota".

4. Seleccione en la lista desplegable "Tipo de adaptador" el tipo de adaptador utilizado.
5. Seleccione en "Tipo de conexión" la "Conexión por línea telefónica", si es que no está ya seleccionada.
6. Seleccione en "Ajustes locales" el módem utilizado.
7. Introduzca el número de teléfono que se debe marcar en el campo de entrada correspondiente o bien abra el listín telefónico haciendo clic en el botón posterior y seleccione el número de teléfono deseado del listín telefónico.
8. Introduzca el nombre de usuario y la contraseña correspondiente del TS Adapter.
9. Si desea "Establecer conexión con rellamada", seleccione el botón de opción correspondiente.
10. Haga clic en el botón "Establecer" para establecer la conexión remota deseada. Este botón sólo se activa una vez introducidos todos los parámetros necesarios para establecer una conexión remota. En "Estado" se indica si ya existe o no una conexión remota.
11. En caso necesario, introduzca en el cuadro de diálogo siguiente el número de rellamada deseado.

Resultado

Se establece la conexión remota con la instalación deseada con rellamada.

En el árbol del proyecto se muestra la instalación conectada con el símbolo correspondiente.

Nota

Este procedimiento es útil cuando los costes de la conexión del módem deben correr a cargo de la instalación o cuando el número de rellamada es variable, es decir, cuando la rellamada no debe dirigirse siempre al mismo destinatario. Esta opción es interesante para los usuarios móviles.

Rellamada a un número guardado en el TS Adapter.

1. Parametrice el número de rellamada deseado en el TS Adapter.
2. Establezca una conexión con el TS Adapter del modo antes descrito teniendo en cuenta las siguientes particularidades:
   - Introduzca el nombre de usuario y la contraseña para los que se ha parametrizado el número de rellamada en el TS Adapter.
   - No es necesario seleccionar el botón de opción "Establecer conexión con rellamada", puesto que el TS Adapter ya conoce el número de rellamada.

Resultado

La rellamada a un número guardado en el TS Adapter está configurada. Si se establece una conexión remota, la rellamada se realiza desde la instalación remota.

Nota

Este procedimiento representa el nivel máximo de protección. De todas formas, existe el peligro de que, si se guarda un número de rellamada erróneo en el TS Adapter, el TS Adapter ya no sea accesible a través de una conexión de módem y sólo pueda volver a ponerse en funcionamiento mediante una reparametrización in situ.

13.7.3.4 TS Adapter MPI

Breve descripción del TS Adapter MPI

TS Adapter MPI:

La denominación "TS Adapter MPI" es un término genérico que agrupa todos los TS Adapter equipados con una interfaz MPI/DP.

El TS Adapter MPI está disponible en los siguientes modelos:

- como TS Adapter I (no puede parametrizarse desde el TIA Portal)
- como TS Adapter II

La siguiente tabla proporciona una breve descripción de la funcionalidad correspondiente. Encontrará más información sobre el TS Adapter que está utilizando en la documentación suministrada con su TS Adapter.

TS Adapter II:	
Conexión directa	La conexión se realiza a través del Universal Serial Bus (USB). El firmware puede intercambiarse. El módem está integrado o también puede conectarse como módem externo. El TS Adapter II conmuta automáticamente entre los módems. Si no hay ningún módem externo conectado y encendido, utiliza el módem interno.
Hay dos variantes	• Con módem analógico interno. Además puede conectarse un módem externo a la interfaz RS232. • Con adaptador ISDN (RDSI) interno. Además, en la interfaz RS232 puede conectarse un módem externo

Utilización de la denominación "TS Adapter"

En TeleService se utiliza la denominación "TS Adapter" como término genérico para todas los modelos. En los casos en los que sólo se hace referencia a un modelo determinado de un TS Adapter, se especifica si se trata de "TS Adapter II", "TS Adapter IE Standard" o "TS Adapter IE Basic".

Funcionamiento del TS Adapter MPI

Funcionamiento del TS Adapter MPI

En función de la configuración, el TS Adapter MPI se conecta al puerto serie o USB de la programadora/el PC (conexión directa) o al puerto serie de un módem (conexión módem) con la red MPI-/PROFIBUS del sistema de automatización.

El TS Adapter MPI dispone de una memoria no volátil. En ella están almacenados los parámetros para las funciones siguientes:

- la red MPI/PROFIBUS (parámetros de red)
- el funcionamiento del módem utilizado
- el puerto serie con el módem
- los derechos de acceso

la parametrización estándar

El TS Adapter se suministra con una parametrización estándar. Los parámetros se pueden ajustar durante una sesión de parametrización y se pueden guardar en la memoria no volátil del TS Adapter.

En la configuración "Conexión directa", el TS Adapter sólo utilizará los parámetros de la red para el acceso a la red MPI/PROFIBUS.

En la configuración "Conexión módem" se activarán todos los parámetros almacenados en el TS Adapter.

Nota

Encontrará más información sobre la parametrización del TS Adapter en la documentación suministrada con el TS Adapter.

Usar el TS Adapter MPI en la conexión directa

Conexión directa con TS Adapter MPI

La conexión directa sirve para parametrizar el TS Adapter MPI. En la misma configuración también se puede pasar a modo online en el TIA Portal y verificar si los parámetros MPI/PROFIBUS definidos son compatibles con el bus. Esto permite, al igual que con un PC Adapter, acceder a sistemas SIMATIC S7/C7 a través de la interfaz MPI/DP sin necesidad de ocupar ningún slot de programadora/PC con un módulo MPI/PROFIBUS.

En la conexión directa no está activo ningún derecho de acceso para el TS Adapter. Así es posible modificar sin más la parametrización del TS Adapter, p. ej., importando parámetros del adaptador.

Nota

Visualización del TS Adapter MPI en el TIA Portal

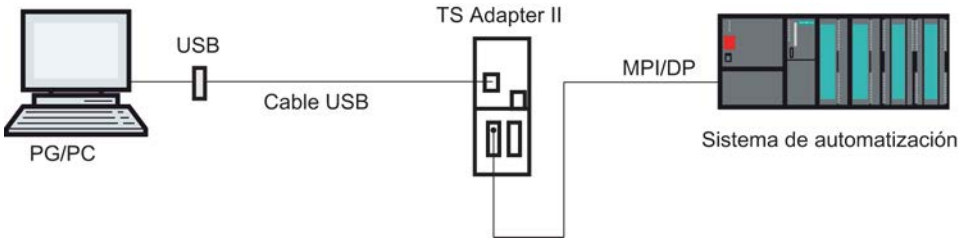
En cuanto se haya conectado un TS Adapter MPI a la PG/el PC a través del puerto USB, en el TIA Portal se mostrará la carpeta "TS Adapter" dentro del árbol del proyecto.

Si se abre la carpeta, se puede parametrizar el TS Adapter MPI conectado de la manera deseada a través de los cuadros de diálogo siguientes.

Configuración de la conexión directa para TS Adapter MPI

En la conexión directa, la programadora o el PC que tiene instalado TeleService, y el sistema de automatización están conectados directamente a través del TS Adapter MPI. No se requiere un módem.

La siguiente figura muestra la conexión directa del TS Adapter MPI.



Usar el TS Adapter MPI en la conexión módem

Introducción a la conexión módem con TS Adapter MPI

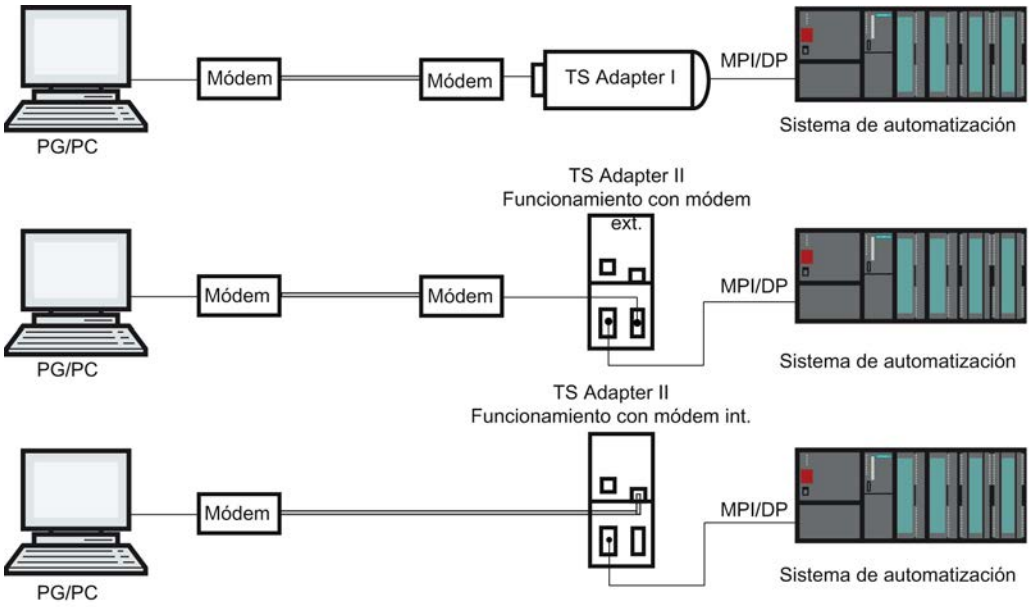
Esta configuración permite marcar una instalación remota. Para ello debe establecerse una conexión remota con una instalación remota con TeleService a través de una red telefónica. A través de la conexión módem configurada se puede trabajar con la instalación seleccionada de la forma habitual con el TIA Portal.

Configuración de la conexión módem con TS Adapter MPI

La conexión entre la PG o el PC en el que está instalado TeleService y el sistema de automatización en cuya interfaz MPI/DP se ha insertado el TS Adapter MPI se establece mediante un módem.

Por lo tanto, con esta configuración se conecta la PG o el PC a la interfaz MPI/DP del sistema de automatización a través de la red telefónica y el TS Adapter MPI.

La figura siguiente muestra la estructura de la conexión módem.



Nota

Funcionamiento paralelo de las conexiones directa y módem

El TS Adapter II dispone de dos conexiones para la comunicación con la PG/el PC, que pueden estar conectadas al mismo tiempo. Conectan el puerto USB con la PG/el PC y, a su vez, el puerto del módem con la red telefónica.

En esta configuración se puede utilizar la conexión directa o bien la conexión módem.

No es posible el funcionamiento paralelo.

Posibilidades de parametrización para el TS Adapter MPI

Información importante sobre la parametrización del TS Adapter MPI

El TS Adapter MPI puede parametrizarse tanto en la conexión directa como también mediante una conexión remota existente.

Existen las siguientes posibilidades de parametrización:

- Nueva parametrización (Página 11395)
- Restauración de la parametrización estándar (Página 11396)
- Importación de parámetros del adaptador (Página 11398)
- Exportación de parámetros del adaptador (Página 11397)
- Crear derechos de acceso (Página 11384)

Realizar parametrización

Parametrice su TS Adapter teniendo en cuenta la documentación suministrada con el TS Adapter. Allí encontrará la información necesaria en relación con el procedimiento exacto durante la parametrización.

Nota

Durante la parametrización del TS Adapter MPI tenga en cuenta lo siguiente:

- Si modifica los parámetros actuales en una conexión remota existente, puede presentarse el caso de que ya no pueda establecerse ninguna conexión módem con los parámetros modificados. En este caso, el TS Adapter MPI sólo puede parametrizarse en la conexión directa.
- Esto significa que, si utiliza una programadora o un PC, debe realizar la parametrización a pie de la instalación o bien que debe llevar el TS Adapter MPIal lugar donde se encuentre la programadora o el PC local para efectuar la parametrización.

Confirmación

Al hacer una parametrización, los datos se escriben en la EEPROM del TS Adapter MPI. La parametrización se acepta sólo después de que el adaptador haya comprobado que los cambios realizados son correctos para que en el caso de un corte de alimentación éstos no se pierdan.

Las modificaciones se hacen efectivas en el TS Adapter MPI del siguiente modo:

- Los parámetros serie, los parámetros de módem y los parámetros para derechos de acceso se activan después de deshacer la conexión remota.
- Los parámetros de red modificados se activan inmediatamente.

Parametrizar el TS Adapter MPI

Introducción

Puede parametrizar el TS Adapter MPI tanto en la conexión directa como también mediante una conexión remota configurada en la conexión módem.

A continuación se describe el procedimiento para efectuar la parametrización.

Requisitos

Un TS Adapter MPI está conectado al equipo y la carpeta "TS Adapter" se muestra en "Accesos online", en el árbol del proyecto.

Procedimiento

Para parametrizar el TS Adapter MPI im Direktanschluss , proceda del siguiente modo:

1. Haga doble clic en la carpeta "Accesos online" en el árbol del proyecto.
2. Abra la carpeta deseada:
   - En caso de una conexión directa, la carpeta "TS Adapter".
   - Si existe una conexión remota, la carpeta "TeleService" y, a continuación, la carpeta con el nombre de la instalación deseada.
3. Seleccione el comando "Parametrizar TS Adapter MPI". A continuación, se abrirá el cuadro de diálogo "Parametrizar TS Adapter MPI".
4. Ajuste los parámetros deseados en las distintas fichas del cuadro de diálogo.
5. Confirme las entradas realizadas con "Aceptar".

Resultado

Los parámetros ajustados se guardarán en la memoria no volátil del TS Adapter MPI. Con ello concluye la parametrización.

Restaurar la parametrización estándar para el TS Adapter MPI

Introducción

Es posible restaurar la parametrización estándar existente en el estado de suministro del TS Adapter MPI.

Requisitos

Un TS Adapter MPI está conectado al equipo y se muestra en la carpeta "TeleService" de "Accesos online", en el árbol del proyecto.

Procedimiento

Para restaurar la parametrización estándar del TS Adapter MPI proceda del siguiente modo:

1. Abra la carpeta "TeleService" del árbol del proyecto.
2. Haga doble clic en la carpeta "TS Adapter MPI".
3. Seleccione el comando "Parametrizar TS Adapter MPI". A continuación, se abrirá el cuadro de diálogo "Parametrizar TS Adapter MPI".
4. En "General", haga clic en el botón "Resetear".
5. Confirme las entradas realizadas con "Aceptar".

Resultado

Se restaurará la parametrización estándar con la que se suministró el TS Adapter MPI.

Consulte también

Posibilidades de parametrización para el TS Adapter MPI (Página 11390)

Exportación de parámetros del adaptador

Introducción

Existe la posibilidad de exportar la parametrización de un TS Adapter MPI a un archivo externo. La parametrización almacenada en dicho archivo se puede volver a importar a cualquier cantidad de TS Adapter MPI.

Esto es especialmente adecuado si desea parametrizar, por ejemplo, de forma idéntica varios TS Adapter MPI, o si desea guardar, documentar o distribuir el juego de parámetros.

Requisitos

Un TS Adapter MPI está conectado al equipo y se muestra en la carpeta "TeleService" de "Accesos online", en el árbol del proyecto.

Procedimiento

Para exportar los parámetros de adaptador de un TS Adapter MPI proceda del siguiente modo:

1. Abra la carpeta "TeleService" del árbol del proyecto.
2. Haga doble clic en la carpeta "TS Adapter MPI".
3. Seleccione el comando "Parametrizar TS Adapter MPI". A continuación, se abrirá el cuadro de diálogo "Parametrizar TS Adapter MPI".
4. Haga clic en el botón "Exportar".
5. En la siguiente ventana, seleccione el archivo al que desea exportar la parametrización del TS Adapter MPI.
6. Confirme haciendo clic en "Guardar".

Resultado

Los parámetros del TS Adapters MPI se guardarán en el archivo indicado (*.tap). Con ello finaliza la exportación de los parámetros del adaptador.

Importación de parámetros del adaptador

Introducción

Existe la posibilidad de importar la parametrización de un TS Adapter MPI de un archivo de exportación (*.tap) creado anteriormente.

La parametrización almacenada en dicho archivo se puede importar a cualquier cantidad de TS Adapter. Esto es especialmente adecuado si se desea parametrizar, por ejemplo, de forma idéntica varios TS Adapter MPI.

Puede realizar la importación de forma local en la conexión directa o mediante una conexión remota configurada en la conexión módem.

Requisitos

Un TS Adapter MPI está conectado al equipo y se muestra en la carpeta "TeleService" de "Accesos online", en el árbol del proyecto.

Procedimiento

Para importar los parámetros de adaptador de un TS Adapter MPI proceda del siguiente modo:

1. Abra la carpeta "TeleService" en el árbol del proyecto.
2. Haga doble clic en la carpeta "TS Adapter MPI".
3. Seleccione el comando "Parametrizar TS Adapter MPI". A continuación, se abrirá el cuadro de diálogo "Parametrizar TS Adapter MPI".
4. Haga clic en el botón "Importar".
5. Seleccione ahora el archivo en el que desea importar la parametrización del TS Adapter MPI.
6. Confirme el cuadro de diálogo posterior con "Sí".

Resultado

Los parámetros seleccionados se guardarán en la memoria no volátil del TS Adapter MPI. Con esto finaliza la importación de los parámetros del adaptador.

13.7.3.5 TS Adapter IE

Breve descripción del TS Adapter IE

TS Adapter IE

La denominación "TS Adapter IE" es un término genérico que agrupa todos los TS Adapter equipados con una interfaz Ethernet.

El TS Adapter IE está disponible en los siguientes modelos:

- como TS Adapter IE Standard
- como TS Adapter IE Basic

La siguiente tabla proporciona una breve descripción de la funcionalidad correspondiente. Encontrará más información sobre el TS Adapter que está utilizando en la documentación suministrada con su TS Adapter.

TS Adapter IE Standard:
Conexión directa a través de Industrial Ethernet (IE). Actualización del firmware posible. Módem integrado o también conectable como módem externo. No disponible conmutación automática de módem como en el caso del TS Adapter II. La parametrización se efectúa con una interfaz web.
Existen 2 variantes:
• Con módem analógico interno. Además puede conectarse un módem externo a la interfaz RS232.
• Con adaptador ISDN (RDSI) interno. Además puede conectarse un módem externo a la interfaz RS232.

TS Adapter IE Basic:
Conexión directa a través de Industrial Ethernet (IE). Actualización del firmware posible. Módulo enchufable. La parametrización se efectúa con una interfaz web.
Existen 4 variantes:
• TS Adapter IE Basic MÓDEM: equipo base TS Adapter IE Basic con módulo TS MÓDEM para el funcionamiento en la red telefónica analógica.
• TS Adapter IE Basic RDSI: equipo base TS Adapter IE Basic con módulo TS RDSI para el funcionamiento en la red telefónica RDSI.
• TS Adapter IE Basic GSM: equipo base TS Adapter IE Basic con módulo TS GSM para el funcionamiento en la red de radiofrecuencia GSM.
• TS Adapter IE Basic RS232: equipo base TS Adapter IE Basic con módulo TS RS232 para la conexión de un módem externo.

Utilización de la denominación "TS Adapter"

En la ayuda en pantalla sobre TeleService aparece la denominación "TS Adapter" como generalización para todos los modelos. En los casos en los que sólo se hace referencia a un modelo determinado de un TS Adapter, se explicita si se trata del "TS Adapter I", "TS Adapter II", "TS Adapter IE Standard" o "TS Adapter IE Basic".

Funcionamiento del TS Adapter IE

Funcionamiento del TS Adapter IE

El TS Adapter IE conecta la red telefónica o el puerto serie de un módem con la Industrial Ethernet del sistema de automatización.

El TS Adapter IE dispone de una memoria no volátil. En ella están almacenados los parámetros para las funciones siguientes:

- el funcionamiento del módem utilizado
- el puerto serie con el módem
- los derechos de acceso

la parametrización estándar

El TS Adapter IE se suministra con una parametrización estándar. Los parámetros se pueden ajustar durante una sesión de parametrización y se pueden guardar en la memoria no volátil del TS Adapter.

Nota

Encontrará más información sobre la parametrización del TS Adapter en la documentación suministrada con el TS Adapter.

Tipos de conexión

Tipos de conexión del TS Adapter IE Basic

Las siguientes figuras muestran los posibles tipos de conexión del TS Adapter IE Basic.

Conexión directa

En conexión directa en la PG/el PC puede parametrizar el TS Adapter IE Basic a través de Ethernet.

Nota

No está permitida la operación del TS Adapter IE Basic sin ningún TS Modul.

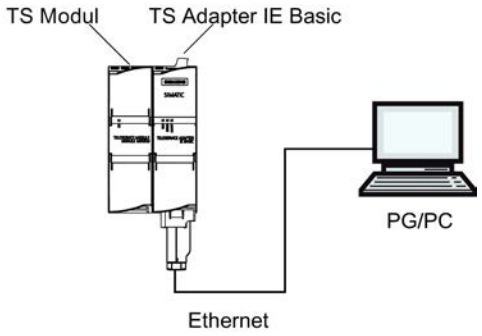


Figura 13-1 Conexión directa

Conexión a la red telefónica

Para la conexión directa a la red telefónica, debe operar el TS Adapter IE Basic en combinación con uno de los siguientes módulos TS:

- TS Module Modem
- TS Module ISDN

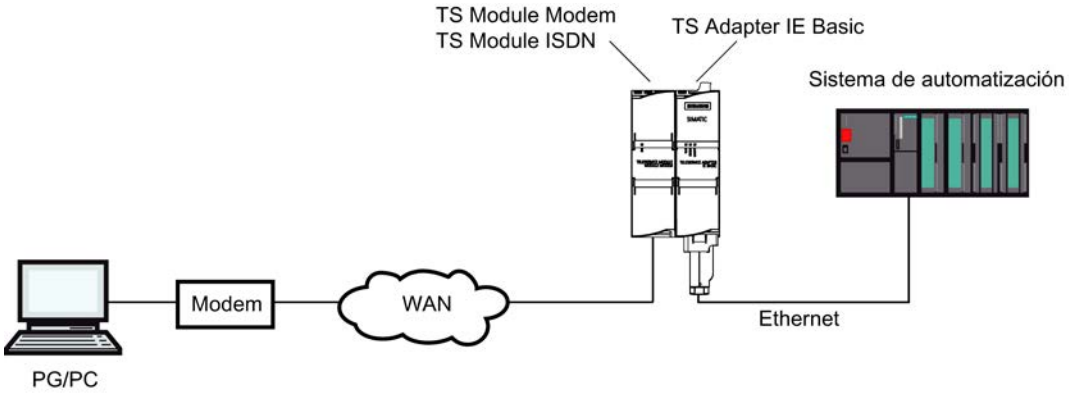


Figura 13-2 Conexión directa a la red telefónica

Más información acerca de los módulos TS encontrará en el manual *TS Adapter modular*.

Conexión a la red GSM

Para la conexión a la red GSM debe operar el TS Adapter IE Basic en combinación con este módulo TS:

- TS Module GSM

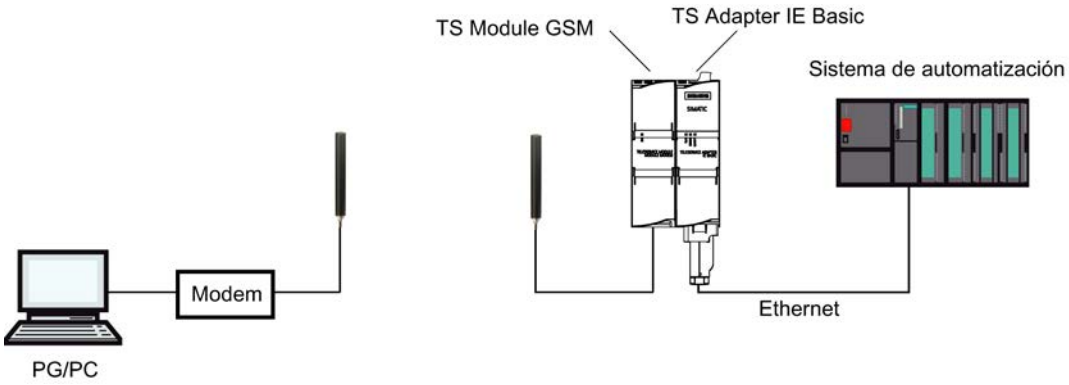


Figura 13-3 Conexión a la red GSM

Más información acerca de los módulos TS encontrará en el manual *TS Adapter modular*.

Conexión a la red telefónica a través de un módem externo

Para la conexión a un módem externo debe operar el TS Adapter IE Basic en combinación con este módulo:

- TS Module RS232

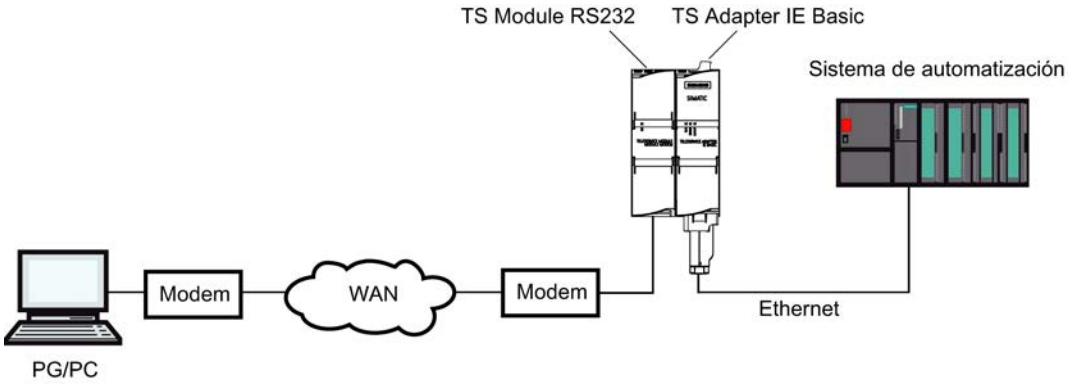


Figura 13-4 Conexión a un módem externo

Más información acerca de los módulos TS encontrará en el manual *TS Adapter modular*.

Posibilidades de parametrización del TS Adapter IE

Información importante para parametrizar el TS Adapter IE

El TS Adapter IE se parametriza con una interfaz web.

Para la parametrización del TS Adapter IE se dispone de una ayuda web asignada a la interfaz web.

Existen las siguientes posibilidades de parametrización:

- Nueva parametrización
- Restauración de la parametrización estándar
- Importación de parámetros del adaptador
- Exportación de parámetros del adaptador

Nota

Realizar la parametrización

Parametrice su TS Adapter teniendo en cuenta la documentación suministrada con su TS Adapter. Allí encontrará la información necesaria en relación con el procedimiento exacto durante la parametrización.

Parametrizar el TS Adapter IE

Introducción

Puede parametrizar el TS Adapter IE tanto en la conexión directa como también mediante una conexión remota configurada en la conexión módem.

A continuación se describen ambas posibilidades de parametrización.

Puede obtener detalles específicos sobre la parametrización del TS Adapter IE en la documentación del TS Adapter IE.

Parametrización del TS Adapter IE en la conexión directa

Requisitos

Existe una conexión LAN con su TS Adapter IE .

El TS Adapter IE Basic está conectado a la alimentación.

Procedimiento

Para parametrizar el TS Adapter IE , proceda del siguiente modo:

1. Abra la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Haga doble clic en la interfaz Ethernet del equipo.
3. Haga doble clic en el comando "Mostrar dispositivos accesibles". A continuación se muestra el TS Adapter IE.
4. Haga doble clic en la carpeta <TS Adapter IE> y después en "Online y diagnóstico", y asigne al TS Adapter IE la dirección IP deseada en los cuadros de diálogo siguientes. Para ello, tenga en cuenta que la dirección IP que asigne al TS Adapter IE debe encontrarse en la misma subred que la dirección IP de la tarjeta de interfaz de la PG/el PC.
5. Actualice la vista del árbol del proyecto de los "Dispositivos accesibles" para que se muestre el TS Adapter IE con la dirección IP recién asignada.
6. Abra la carpeta <TS Adapter IE> en la lista de dispositivos.
7. Haga doble clic en el comando "Parametrizar TS Adapter IE". A continuación se abre la interfaz web asignada para parametrizar el TS Adapter IE.
8. Inicie la sesión en la interfaz web.
9. Ajuste los parámetros deseados en las distintas fichas del cuadro de diálogo.
10. Confirme en cada caso las entradas realizadas con "Guardar ajustes".

Resultado

Los parámetros ajustados se guardarán en la memoria no volátil del TS Adapter IE. Con ello se finaliza la parametrización.

Parametrización del TS Adapter IE a través de la conexión remota

Requisitos

Hay una conexión remota establecida con un TS Adapter IE .

Procedimiento

1. Abra la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Abra la carpeta "TeleService" y, a continuación, la carpeta de la instalación deseada.
3. Haga doble clic en el comando "Parametrizar TS Adapter IE". A continuación se abre la interfaz web correspondiente para parametrizar el TS Adapter IE. El "inicio de sesión" en la interfaz web se efectúa automáticamente con los datos de inicio de sesión de la conexión remota.
4. Ajuste los parámetros deseados en las distintas fichas del cuadro de diálogo.
5. Confirme en cada caso las entradas realizadas con "Guardar ajustes".

Resultado

Los parámetros ajustados se guardarán en la memoria no volátil del TS Adapter IE. Con ello concluye la parametrización.

13.7.3.6 Establecimiento de una conexión por línea telefónica con una instalación remota

Establecer conexión por línea telefónica

Introducción al establecimiento de una conexión remota como conexión por línea telefónica

Se produce una conexión por línea telefónica cuando se utiliza TeleService para marcar una instalación remota a través de una red telefónica. Para ello, la programadora o el PC debe estar conectado a la red telefónica vía módem local utilizando TeleService. En el otro extremo, el sistema de automatización está conectado a la línea telefónica a través de un TS Adapter parametrizado y un módem.

Requisitos

Un módem local está instalado y parametrizado.

En la instalación remota existe un TS Adapter.

Un módem remoto está instalado y parametrizado.

Proceda del siguiente modo:

1. Haga clic en la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Dentro de esta, haga clic en la carpeta "TeleService".
3. Haga doble clic en la entrada "Establecer/deshacer conexión a distancia". Se abre el cuadro de diálogo "Establecer conexión a distancia con la instalación remota".
4. Seleccione en la lista desplegable "Tipo de adaptador" el tipo de adaptador utilizado.
5. Seleccione en "Tipo de conexión" la "Conexión por línea telefónica".
6. Seleccione en "Ajustes locales" el módem utilizado.
7. Introduzca el número de teléfono que se debe marcar en el campo de entrada correspondiente o bien abra el listín telefónico haciendo clic en el botón posterior y seleccione el número de teléfono deseado del listín telefónico.
8. Introduzca su nombre de usuario y la contraseña correspondiente.
9. Si desea "Establecer conexión con rellamada", seleccione el botón de opción correspondiente.
10. Haga clic en el botón "Establecer" para establecer la conexión remota deseada. Este botón solo se activa una vez introducidos todos los parámetros necesarios para establecer una conexión remota. En "Estado" se indica si ya existe o no una conexión remota.

Resultado

Se establece la conexión por línea telefónica con la instalación deseada.

En cuanto se establece la conexión por línea telefónica, se cierra el cuadro de diálogo. En la barra de estado del TIA Portal aparece el mensaje: "Conexión establecida". De esta manera se puede utilizar la conexión remota con el TIA Portal y comunicar con el sistema de automatización.

No es posible establecer la conexión

Si no puede establecer conexión, intente averiguar el motivo del fallo por medio de las "Indicaciones para la búsqueda de errores".

Deshacer la conexión

Si ha terminado de trabajar con la instalación remota, finalice la conexión remota en el árbol del proyecto haciendo doble clic en la entrada "Establecer/deshacer conexión a distancia".

Al salir del TIA Portal también finaliza la conexión remota.

Deshacer conexión por línea telefónica

Deshacer una conexión por línea telefónica activa

Nota

Deshacer la conexión por línea telefónica

Antes de deshacer la conexión remota debe cambiar al modo offline del TIA Portal.

Proceda del siguiente modo:

1. Haga doble clic en la entrada "Establecer/deshacer conexión a distancia" del TIA Portal.
2. Confirme la consulta del cuadro de diálogo posterior con "Sí".

Resultado

Se deshace la conexión.

13.7.4 Conexión remota como conexión VPN

13.7.4.1 Principios básicos para el establecimiento de una conexión VPN

Uso de un TS Adapter IE Advanced para conexiones VPN

Para poder establecer una conexión VPN con TeleService, es necesario utilizar un TS Adapter IE Advanced.

Definición VPN

VPN significa "Virtual Private Network". Es decir, una red informática privada cerrada basada en una infraestructura de red pública.

Una VPN se utiliza para conectar dispositivos separados de una red privada con un punto de conexión de otra red con la máxima seguridad posible.

Un dispositivo VPN obtiene acceso directo a un punto de conexión de otra red a través de una conexión codificada.

Conexiones VPN

Al establecer una conexión VPN con TeleService, para identificar de forma unívoca el TS Adapter IE Advanced se utiliza un certificado CA y una huella digital única generada a partir del certificado. La conexión VPN permite una comunicación a prueba de escuchas y manipulaciones entre el PC remoto y el TS Adapter.

Si desea establecer una conexión VPN a varias redes remotas, necesitará para cada red un TS Adapter IE Advanced propio como interfaz. Las redes conectadas a través del TS Adapter pueden utilizar la misma dirección IP interna. Solo deben ser diferentes las direcciones IP externas (WAN) de cada TS Adapter.

Solo se puede realizar una conexión VPN con un TS Adapter al mismo tiempo.

Nota

Autenticación

Tenga en cuenta que el certificado CA solo autentica el TS Adapter IE Advanced.

Un usuario se autentica a través del inicio de sesión (nombre de usuario y contraseña), igual que en las conexiones por línea telefónica.

Por lo tanto, utilice únicamente contraseñas seguras para el inicio de sesión.

Consulte al respecto también

- Establecer conexión VPN (Página 11412)
- Deshacer conexión VPN (Página 11414)

13.7.4.2 Principios básicos de los certificados CA

Introducción

Para poder establecer una conexión VPN segura con TeleService, es necesario generar un certificado CA durante la configuración de TS Adapter IE Advanced con una huella digital única.

Dicho certificado CA se puede instalar del siguiente modo en cualquier PC que deba tener acceso al TS Adapter IE Advanced:

- Mediante descarga automática en el cuadro de diálogo de conexión de TeleService, donde debe registrar la huella digital que corresponde al certificado CA
- Mediante instalación manual con Microsoft® Management Console.

Definición de certificado CA

Un certificado CA es un certificado digital emitido a través de un organismo de certificación autorizado ("certificate authority" o "certification authority", en adelante "CA"). En el caso de

TS Adapter IE Advanced, se utilizan certificados autofirmados; el organismo de certificación autorizado es aquí el propio TS Adapter IE Advanced.

Del certificado CA se derivan los certificados para "SSTP" (Secure Socket Tunneling Protocol) y "HTTPS" (Hypertext Transfer Protocol Secure).

Los certificados CA contienen una "clave" e información adicional que sirven para la autenticación, la codificación y la descodificación de datos confidenciales. Se incluye información adicional como, por ejemplo, el periodo de validez, remisiones a listas de revocación de certificados, etc., que se introduce en el certificado a través de CA.

Utilización de certificados CA en TeleService

Para identificar de forma inequívoca el TS Adapter IE Advanced como interlocutor respecto al PC remoto, desde el TS Adapter se genera un certificado CA con una huella digital única.

Para establecer una conexión VPN es imprescindible guardar este certificado CA en el almacén de certificados de Windows de su PG/PC. Si activa el servidor web en conexión directa y falta el certificado CA, se muestra una advertencia de seguridad que puede aceptarse.

Nota

Gestión de certificados CA

¡La gestión de certificados CA requiere conocimientos especiales sobre el sistema operativo y solo debe realizarla personal instruido!

Para gestionar los certificados CA necesitará tener derechos de administrador.

Definición de huella digital

La huella digital es una representación hexadecimal con una longitud de 20 bytes. Representa un valor unívoco para un certificado CA y se usa para identificar un certificado CA determinado.

La huella digital se calcula de forma dinámica con el algoritmo SHA1 y no existe físicamente en el certificado CA.

Utilización de la huella digital en TeleService

El certificado CA sirve para identificar de forma inequívoca el TS Adapter IE Advanced como interlocutor. Cada vez que el TS Adapter IE Advanced genera un certificado CA, se genera automáticamente una huella digital única de ese certificado CA con una longitud de 20 bytes. TS Adapter IE Advanced la calcula automáticamente cuando se genera un certificado. Cada certificado tiene una huella digital específica y única. Esta huella digital debe transferirse al equipo a través de un canal seguro, p. ej., por teléfono o mediante un correo codificado. Al establecer una conexión VPN con TeleService, deberá introducir la huella digital en el cuadro de diálogo de conexión, siempre que no haya guardado ya el certificado CA en el almacén de certificados de Windows de su PC.

Encontrará la huella digital correspondiente al TS Adapter IE Advanced en la interfaz web del TS Adapter IE Advanced. Para abrir la interfaz web, haga doble clic en el comando

"Parametrizar TS Adapter IE Advanced" en la lista de dispositivos del TIA Portal. Inicie sesión en la interfaz web para visualizar la huella digital en la ficha "Seguridad > Certificados".

Descarga del certificado CA al establecer la conexión

Al establecer la conexión, TeleService comprueba si se ha instalado un certificado CA adecuado en el almacén de certificados de Windows de su PG/PC. Si se encuentra un certificado CA adecuado, se establece la conexión VPN como conexión SSTP (Secure Socket Tunneling Protocol).

Si no se encuentra ningún certificado CA adecuado, se cargará primero el certificado CA del TS Adapter IE Advanced correspondiente. Para ello, se llamará al TS Adapter IE Advanced a través de la dirección remota introducida en el cuadro de diálogo de conexión. Si la descarga del certificado CA es correcta, se calcula la huella digital del certificado CA descargado y se compara con la huella digital introducida en el cuadro de diálogo de conexión. Si ambas huellas coinciden, se abre un cuadro de diálogo que le preguntará si desea guardar el certificado CA en el almacén de certificados de Windows de su PG/PC. Para guardar el certificado CA necesitará tener derechos de administrador.

A continuación, se establece la conexión VPN.

Consulte al respecto también

- Instalar certificado CA para conexión VPN (Página 11409)
- Borrar certificado CA para conexión VPN (Página 11412)

13.7.4.3 Instalar certificado CA para conexión VPN

Instalación de certificados CA

Para establecer una conexión VPN segura con TeleService entre su PG/PC y una instalación remota, necesitará un certificado CA válido generado por el TS Adapter IE Advanced. Dicho certificado debe estar guardado en el almacén de certificados de Windows de su PG/PC.

La instalación de un certificado CA se puede realizar mediante descarga automática o manualmente.

La gestión de certificados CA se realiza a través de Microsoft® Management Console.

Nota

Gestión de certificados CA

¡La gestión de certificados CA requiere conocimientos especiales sobre el sistema operativo y solo debe realizarla personal instruido!

¡Para gestionar los certificados CA necesitará tener derechos de administrador!

Requisitos

Su equipo no tiene instalado todavía ningún certificado CA en el almacén de certificados de Windows.

Instalación de certificados CA mediante descarga automática

Proceda del siguiente modo:

1. Inicie sesión en el sistema como administrador.
2. Transfiera la huella digital correspondiente al certificado CA a través de un "canal seguro", p. ej., por teléfono o mediante un correo codificado, del TS Adapter IE Advanced a su equipo.
   Encontrará la huella digital correspondiente al TS Adapter IE Advanced en la interfaz web del TS Adapter IE Advanced. Para abrir la interfaz web, haga doble clic en el comando "Parametrizar TS Adapter IE" en la lista de dispositivos del TIA Portal. Inicie sesión en la interfaz web para visualizar la huella digital en la ficha "Seguridad > Certificados".
3. Haga clic en la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
4. Dentro de esta, haga clic en la carpeta "TeleService".
5. Haga doble clic en la entrada "Establecer/deshacer conexión a distancia". Se abre el cuadro de diálogo "Establecer conexión a distancia con la instalación remota".
6. En la lista desplegable, seleccione "TS Adapter IE" como Tipo de adaptador.
7. Seleccione "VPN" como Tipo de conexión.
8. Introduzca la dirección IP o el nombre DNS para el TS Adapter IE Advanced de contacto en el campo de entrada correspondiente. También puede seleccionar los datos que haya disponibles en el listín telefónico haciendo clic en el botón posterior.
9. Introduzca su nombre de usuario y la contraseña correspondiente.
10. Copie la huella digital que se muestra en la interfaz web del TS Adapter IE Advanced en la columna "Huella digital".
11. Haga clic en el botón "Establecer" para establecer la conexión remota deseada. Este botón solo se activa una vez introducidos todos los parámetros necesarios para establecer una conexión remota.
12. Como no había ningún certificado CA instalado en su equipo, no se encuentra ningún certificado CA válido.
    Por tanto, se establece una conexión "normal" (no una conexión VPN) y el certificado necesario se descarga del TS Adapter IE Advanced a la memoria de trabajo de su equipo. A continuación, se calcula la huella digital (algoritmo SHA1) y se compara con la huella digital introducida en el cuadro de diálogo de conexión. Si ambas huellas coinciden, se abre un cuadro de diálogo que le preguntará si desea guardar el certificado CA en el almacén de certificados de Windows de su PG/PC.
13. Confirme que desea guardar el certificado CA.

Resultado

Se establece la conexión VPN con el TS Adapter IE Advanced deseado. En cuanto se establece la conexión, se cierra el cuadro de diálogo.

Instalación manual de certificados CA

Proceda del siguiente modo:

1. Inicie sesión en el sistema como administrador.
2. Abra el administrador de certificados de Windows de su PG/PC con Microsoft® Management Console.
   Para ello, haga clic en "Inicio", introduzca "mmc" en el campo de búsqueda y pulse la tecla INTRO.
   Se abre la consola.
3. En el menú "Archivo", haga clic en el comando "Agregar o quitar complemento...".
   Se abre el cuadro de diálogo "Agregar o quitar complemento".
4. En la lista "Complementos" haga doble clic en "Certificados" y seleccione "Cuenta de equipo" en el cuadro de diálogo que se abre a continuación.
5. En el cuadro de diálogo siguiente, seleccione "Equipo local" y haga clic en "Finalizar" y en "Aceptar".
   Se abre la Raíz de consola y se muestra la carpeta "Certificados (equipo local)".
6. Abra la carpeta "Certificados (equipo local)" mostrada y haga clic en "Entidades de certificación raíz de confianza".
7. Haga clic en la carpeta "Certificados" y, a través del menú contextual, abra el comando "Todas las tareas" > "Importar...".
8. Tenga en cuenta la información mostrada en el cuadro de diálogo "Asistente para importación de certificados" y haga clic en "Siguiente".
9. En el cuadro de diálogo siguiente, haga clic en "Examinar..." y seleccione el certificado CA deseado.
10. A continuación, haga clic dos veces en "Siguiente" y después en "Finalizar" para instalar el certificado CA.

Resultado

El certificado CA seleccionado se instala en la ubicación indicada del almacén de certificados de Windows.

Nota

Para más información...

... sobre la instalación de certificados CA, pulse la tecla "F1" de ayuda de su sistema operativo.

13.7.4.4 Borrar certificado CA para conexión VPN

Borrar certificados CA

Proceda del siguiente modo:

1. Inicie sesión en el sistema como administrador.
2. Abra el administrador de certificados de Windows de su PG/PC con Microsoft® Management Console.
   Para ello, haga clic en "Inicio", introduzca "mmc" en el campo de búsqueda y pulse la tecla INTRO.
   Se abre la consola.
3. En el menú "Archivo" haga clic en "Agregar o quitar complemento...".
   Se abre el cuadro de diálogo de selección de complementos.
4. En la lista "Complementos" haga doble clic en "Certificados" y seleccione "Cuenta de equipo" en el cuadro de diálogo que se abre a continuación.
5. En el cuadro de diálogo siguiente, seleccione "Equipo local" y haga clic en "Finalizar" y en "Aceptar".
   Se abre la Raíz de consola y se muestra la carpeta "Certificados (equipo local)".
6. Abra la carpeta "Certificados (equipo local)" mostrada y haga clic en "Entidades de certificación raíz de confianza".
7. Abra la carpeta "Certificados", seleccione el certificado CA deseado y haga clic en "Borrar" en el menú contextual.
8. Confirme la consulta posterior con "Sí".

Resultado

El certificado CA seleccionado se borra de la lista de certificados disponibles.

13.7.4.5 Establecimiento de una conexión VPN con una instalación remota

Establecer conexión VPN

Introducción al establecimiento de una conexión VPN

Se produce una conexión VPN cuando se utiliza TeleService para conectar con una instalación remota a través de Internet.

Para ello, por un lado, la PG/el PC con el TIA Portal instalado tiene conexión a Internet. Por otro lado, el sistema de automatización está conectado a Internet a través de la interfaz WAN (Wide Area Network) del TS Adapter IE Advanced parametrizado.

Requisitos

Su PG/PC está conectado a Internet.

En la instalación remota existe un TS Adapter IE Advanced.

El TS Adapter IE Advanced está parametrizado y conectado a Internet.

El certificado CA necesario para identificar el TS Adapter ha sido creado y se ha instalado en el almacén de certificados de Windows de su PG/PC.

Proceda del siguiente modo:

1. Haga clic en la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Dentro de esta, haga clic en la carpeta "TeleService".
3. Haga doble clic en la entrada "Establecer/deshacer conexión a distancia". Se abre el cuadro de diálogo "Establecer conexión a distancia con la instalación remota".
4. En la lista desplegable, seleccione "TS Adapter IE" como "Tipo de adaptador".
5. Como "Tipo de conexión", introduzca "VPN".
6. Introduzca la dirección IP o el nombre DNS para el TS Adapter IE Advanced de contacto en el campo de entrada correspondiente. También puede seleccionar los datos que haya disponibles en el listín telefónico haciendo clic en el botón posterior.
7. Introduzca su nombre de usuario y la contraseña correspondiente.
8. Haga clic en el botón "Establecer" para establecer la conexión VPN deseada. Este botón solo se activa una vez introducidos todos los parámetros necesarios para establecer la conexión remota.

Resultado

Se establece la conexión VPN con la instalación deseada. En "Estado" se muestra el progreso del establecimiento de la conexión. En cuanto se establece la conexión VPN, se cierra el cuadro de diálogo. En la barra de estado del TIA Portal aparece el mensaje: "Conexión establecida". De esta manera se puede utilizar la conexión remota con el TIA Portal y comunicar con el sistema de automatización.

No es posible establecer la conexión

Si no puede establecer conexión, intente averiguar el motivo del fallo por medio de las "Indicaciones para la búsqueda de errores".

Nota

Reglas para las direcciones IP

- Utilice únicamente direcciones IP que todavía no estén asignadas a la red de planta.
- Si la dirección IP parametrizada para el TS Adapter IE Advanced ya está asignada a la red de planta, solo podrá acceder al TS Adapter IE Advanced a través de su dirección MAC.

Consulte también

Instalar certificado CA para conexión VPN (Página 11405)

Deshacer conexión VPN

Deshacer una conexión VPN activa

Nota
Deshacer la conexión VPN
Antes de deshacer la conexión VPN debe cambiar al modo offline del TIA Portal.

Proceda del siguiente modo:

1. Haga doble clic en la entrada "Establecer/deshacer conexión a distancia" del TIA Portal.
2. Confirme la consulta del cuadro de diálogo posterior con "Sí".

Resultado

Se deshace la conexión VPN.

13.7.4.6 TS Adapter IE Advanced

Descripción abreviada del TS Adapter IE Advanced

TS Adapter IE Advanced

El TS Adapter IE Advanced tiene las siguientes propiedades:

- Conexión directa a través de Industrial Ethernet (IE), 2 puertos
- Interfaz WAN (Wide Area Network) para conexiones VPN
- Actualización del firmware posible
- Módulo enchufable
- La parametrización se efectúa con una interfaz web.

Nota

Más información sobre el TS Adapter IE Advanced

Encontrará más información sobre el TS Adapter que está utilizando en la documentación suministrada con su TS Adapter.

Tipos de conexión

Tipos de conexión del TS Adapter IE Advanced

Las siguientes figuras muestran los posibles tipos de conexión del TS Adapter IE Advanced.

Conexión directa

En conexión directa en la PG/el PC puede parametrizar el TS Adapter IE Advanced a través de Ethernet.

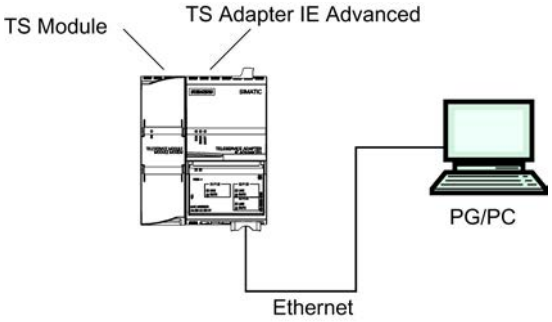


Figura 13-5 TS Adapter IE Advanced: conexión directa

Conexión a Internet (módem/router ADSL)

Para la conexión a Internet debe operar el módem/router ADSL en la conexión WAN del TS Adapter IE Advanced.

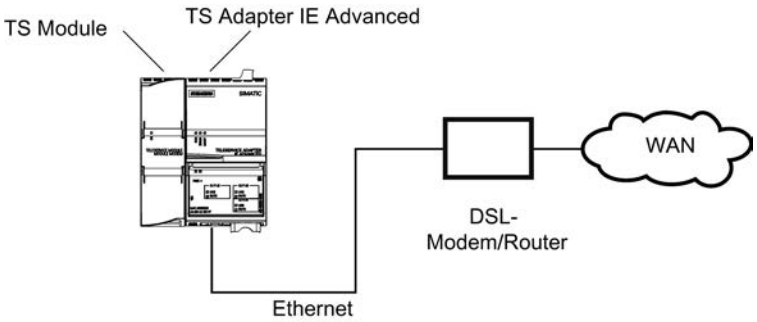


Figura 13-6 TS Adapter IE Advanced: conexión a Internet

Conexión a la red de empresa (Intranet)

Para la conexión a la Intranet debe operar la red de planta (Ethernet) en una de las dos conexiones LAN del TS Adapter IE Advanced.

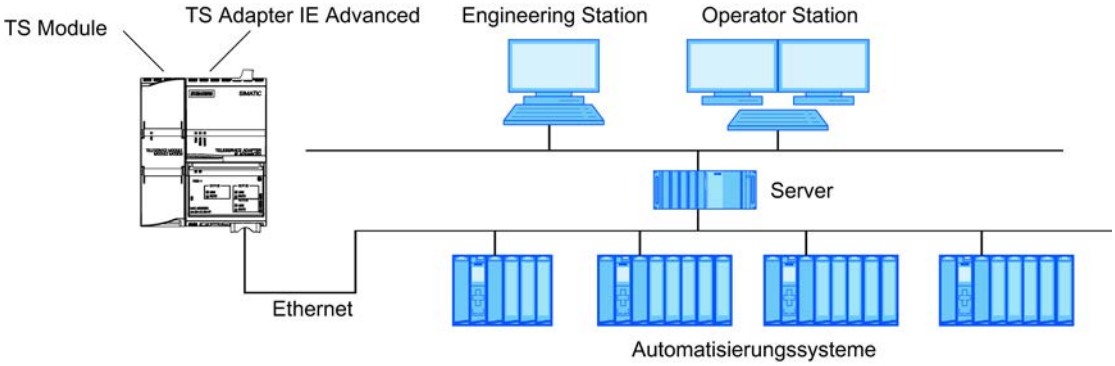


Figura 13-7 TS Adapter IE Advanced: conexión a la red de empresa (Intranet)

Más información

Encontrará más información sobre los módulos TS en la documentación que se le ha suministrado con su TS Adapter IE Advanced.

Posibilidades de parametrización del TS Adapter IE Advanced

Principios básicos para la parametrización del TS Adapter IE

El TS Adapter IE Advanced se parametriza con una interfaz web.

Para la parametrización del TS Adapter IE Advanced se dispone de una ayuda web asignada a la interfaz web.

Existen, p. ej., las siguientes posibilidades de parametrización:

- Nueva parametrización
- Restauración de la parametrización estándar
- Importación de parámetros del adaptador
- Exportación de parámetros del adaptador

Nota

Realizar la parametrización

Parametrice su TS Adapter IE Advanced teniendo en cuenta la documentación suministrada. Allí encontrará la información necesaria en relación con el procedimiento exacto durante la parametrización.

Parametrizar el TS Adapter IE Advanced

Introducción

Puede parametrizar el TS Adapter IE Advanced tanto en la conexión directa como también mediante una conexión remota ya configurada.

A continuación se describen ambas posibilidades de parametrización.

Puede obtener detalles específicos sobre la parametrización del TS Adapter IE Advanced en la documentación del TS Adapter IE Advanced.

Parametrización del TS Adapter IE Advanced en la conexión directa

Requisitos

Existe una conexión LAN con su TS Adapter IE Advanced.

Procedimiento

Para parametrizar el TS Adapter IE Advanced, proceda del siguiente modo:

1. Abra la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Haga doble clic en la interfaz Ethernet del equipo.
3. Haga doble clic en el comando "Mostrar dispositivos accesibles". A continuación se muestra el TS Adapter IE Advanced.
4. Haga doble clic en la carpeta "TS Adapter IE Advanced" y después en "Online y diagnóstico", y asigne al TS Adapter la dirección IP deseada en los cuadros de diálogo siguientes. Para ello, tenga en cuenta que la dirección IP que asigne al TS Adapter IE Advanced debe encontrarse en la misma subred que la dirección IP de la tarjeta de interfaz de la PG/el PC.
5. Actualice la vista del árbol del proyecto de los "Dispositivos accesibles" para que se muestre el TS Adapter IE Advanced con la dirección IP recién asignada.
6. Abra la carpeta "TS Adapter IE Advanced! en la lista de dispositivos.
7. Haga doble clic en el comando "Parametrizar TS Adapter IE". A continuación se abre la interfaz web asignada para parametrizar el TS Adapter.
8. Inicie la sesión en la interfaz web.
9. Ajuste los parámetros deseados en las distintas fichas del cuadro de diálogo.
10. Confirme en cada caso las entradas realizadas con "Guardar ajustes".

Resultado

Los parámetros ajustados se guardarán en la memoria no volátil del TS Adapter IE Advanced. Con ello finaliza la parametrización.

Parametrización del TS Adapter IE Advanced a través de la conexión remota

Requisitos

Hay una conexión remota establecida con un TS Adapter IE Advanced.

Procedimiento

1. Abra la carpeta "Accesos online" en el árbol del proyecto del TIA Portal.
2. Abra la carpeta "TeleService" y, a continuación, la carpeta de la instalación deseada.
3. Haga doble clic en el comando "Parametrizar TS Adapter IE". A continuación se abre la interfaz web correspondiente para parametrizar el TS Adapter IE. El "inicio de sesión" en la interfaz web se efectúa automáticamente con los datos de inicio de sesión de la conexión remota.
4. Ajuste los parámetros deseados en las distintas fichas del cuadro de diálogo.
5. Confirme en cada caso las entradas realizadas con "Guardar ajustes".

Resultado

Los parámetros ajustados se guardarán en la memoria no volátil del TS Adapter IE Advanced. Con ello finaliza la parametrización.

13.7.5 Conexiones remotas controladas por CPU en TeleService

13.7.5.1 Vista general de las conexiones remotas controladas por CPU

Introducción

TeleService ofrece, en función de la CPU utilizada en cada caso, diferentes posibilidades para establecer una conexión remota. La iniciativa para establecer la conexión parte siempre de la CPU. Para las distintas posibilidades de conexión se utilizan las instrucciones de comunicación que se indican a continuación.

Posibilidades para el establecimiento de conexiones con CPU S7-300/400

Existen las siguientes instrucciones de comunicación:

- Instrucción de comunicación "PG_DIAL": Establecer una conexión remota con una PG/un PC
- Instrucción de comunicación "SMS_SEND": Enviar un mensaje SMS

- Instrucción de comunicación "AS_DIAL": Establecer una conexión remota con el AS
- Instrucción de comunicación "AS_MAIL": Transmitir un mensaje de correo electrónico

Posibilidades para el establecimiento de conexiones con CPU S7-1200

Existe la siguiente instrucción de comunicación:

- Instrucción de comunicación "TM_MAIL": Transmitir un mensaje de correo electrónico

Posibilidades para establecer una conexión con CPU S7-1500

Existe la siguiente instrucción de comunicación:

- Instrucción de comunicación "TMAIL_C": Transmitir un mensaje de correo electrónico

---

Nota

Descripción de las distintas instrucciones de comunicación

Puede encontrar información detallada sobre las instrucciones de comunicación que pueden utilizarse en el sistema de información del TIA Portal en el directorio "Referencias > Comunicación > TeleService".

Consulte también

PG_DIAL: Establecer una conexión remota con una PG/un PC (Página 8278)

AS_DIAL: Establecer una conexión remota con el AS (Página 8283)

SMS_SEND: Enviar un mensaje SMS (Página 8289)

AS_MAIL: Transmitir un mensaje de correo electrónico (Página 8295)

TM_MAIL: Transmitir e-mail (Página 5588)

Posibilidades de parametrización del TS Adapter IE (Página 11398)

13.7.5.2 Establecer conexiones desde y hacia instalaciones remotas (acoplamiento remoto PG-AS)

Acceder desde una instalación remota a una programadora/un PC

Introducción

Con la aplicación TeleService y un TS Adapter MPI se puede establecer una conexión remota con una instalación remota y comunicarse con ella. La iniciativa para establecer la conexión remota parte de la programadora/del PC.

No obstante, en una instalación remota se producen a menudo eventos que exigen una intervención rápida. En tal caso, p. ej. cuando se produce un evento asíncrono, el sistema de

automatización puede iniciar el establecimiento de una conexión remota con una programadora/un PC.

La figura siguiente muestra gráficamente los componentes necesarios para establecer la conexión desde una instalación con una programadora/un PC.

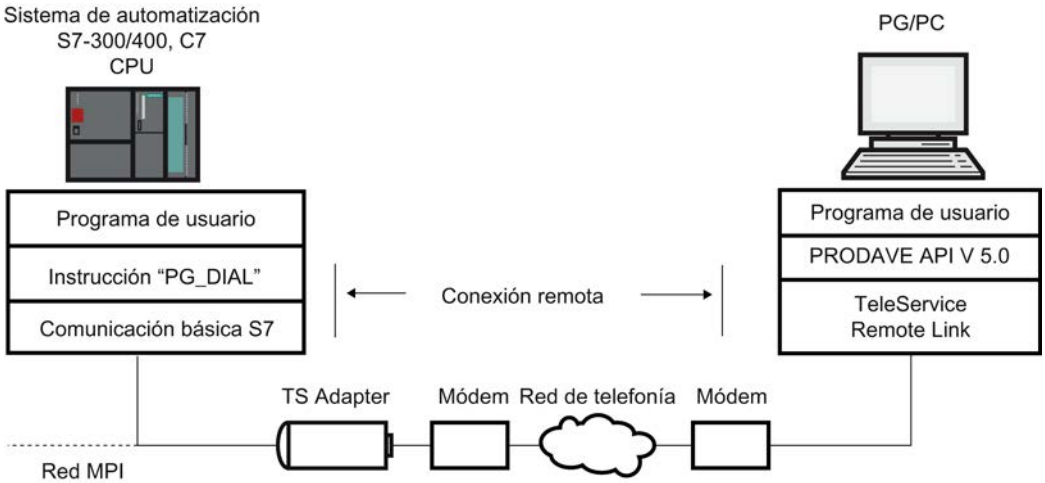


Figura 13-8 Funcionamiento de la instrucción de comunicación "PG_DIAL"

Requisitos para establecer una conexión

Introducción

Para que un sistema de automatización remoto pueda establecer una conexión remota con una programadora/un PC deben cumplirse ciertos requisitos de hardware y software. A continuación se describen estos requisitos.

Requisitos de hardware:

Para establecer una conexión remota desde una instalación remota a una programadora/un PC no se requiere más hardware que el necesario para acceder desde la programadora/el PC a la instalación remota.

Para establecer la conexión el programa de usuario llama la instrucción de comunicación "PG_DIAL" . Esta puede ejecutarse en una CPU S7-300 o S7-400 donde esté realizada la comunicación básica S7.

Debe utilizarse un TS Adapter I con una versión V5.0 o superior o un TS Adapter II .

Requisitos de software en la instalación:

En el volumen de suministro de TeleService está incluida la instrucción de comunicación "PG_DIAL" que se instala conjuntamente al instalar el TIA Portal. Puede encontrar la instrucción de comunicación instalada conjuntamente en la Task Card del editor de bloque, en la carpeta "Comunicación > TeleService".

Para que una instalación remota pueda establecer una conexión remota con una programadora o un PC, el programa de usuario de la instalación debe llamar la instrucción de comunicación "PG_DIAL".

Requisitos de software en la programadora o en el PC:

En la programadora o en el PC se necesita un producto de software que, en combinación con TeleService, espere la llamada de una instalación remota, la detecte e informe al programa de usuario.

13.7.5.3 Intercambio de datos entre instalaciones remotas (acoplamiento remoto AS-AS)

Principios básicos para el acoplamiento remoto AS-AS

Introducción

El acoplamiento remoto AS-AS permite que dos sistemas de automatización intercambien datos del proceso a través de la red telefónica.

Requisitos

La instrucción de comunicación "AS_DIAL" sólo está disponible si utiliza una CPU de la serie S7-300/400.

Definición: Sistema de automatización local y remoto

- El sistema de automatización del que parte la iniciativa de establecer la conexión remota se denomina **local**.
- El sistema de automatización con el que debe establecerse la conexión remota se denomina **remoto**.

Intercambio de datos a través del acoplamiento remoto AS-AS

El intercambio de datos tiene lugar a través de instrucciones de comunicación específicas para conexiones S7 no configuradas. Para el establecimiento de una conexión remota al sistema de automatización utilice la instrucción de comunicación "AS_DIAL".

Puede encontrar información detallada para el establecimiento de una conexión en el sistema de información en el directorio "Referencias > Comunicación > TeleService".

La figura siguiente muestra una representación gráfica de los componentes necesarios para establecer la conexión de un sistema de automatización local a otro remoto.

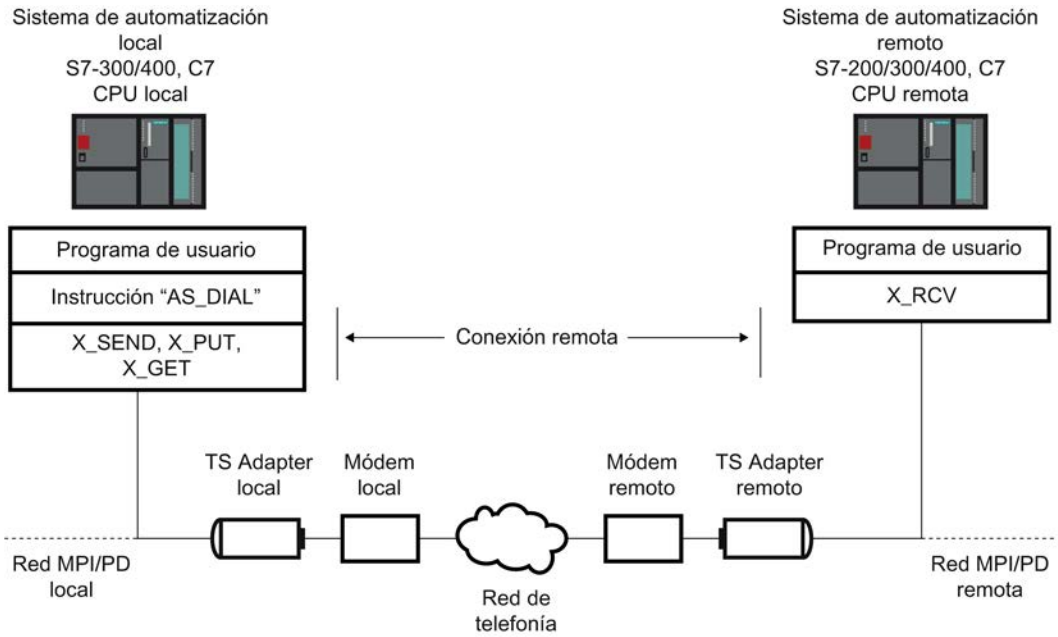


Figura 13-9 Intercambio de datos a través del acoplamiento remoto AS-AS

Requisitos de hardware y software para el acoplamiento remoto AS-AS

Introducción

Para que un sistema de automatización local pueda establecer una conexión a distancia con un sistema de automatización remoto deben cumplirse ciertos requisitos de hardware y software. A continuación se describen estos requisitos.

Requisitos de hardware

Para la transferencia de datos del proceso de un sistema de automatización local a otro remoto, sólo se necesita el hardware que requiere la programadora/el PC para acceder al sistema de automatización correspondiente.

Para establecer y deshacer la conexión remota, el programa de usuario TIA Portal de la CPU local llama una instrucción de comunicación. Esta instrucción de comunicación se puede ejecutar en una CPU S7-300/400 o en una CPU C7. La instrucción de comunicación requiere que en la CPU se haya establecido la comunicación básica S7. Asimismo, la CPU remota debe ser compatible con la comunicación básica S7.

Debe utilizarse un TS Adapter I con una versión V5.1 o superior o un TS Adapter II.

Requisitos de software

En el volumen de suministro de TeleService está incluida la instrucción de comunicación "AS_DIAL", que se incorpora en la librería del TIA Portal durante la instalación en la Task Card, en la carpeta de instrucciones de comunicación de TeleService. Para establecer y deshacer una conexión remota del sistema de automatización local al remoto hay que llamar

la instrucción de comunicación "AS_DIAL" en el programa de usuario TIA Portal de la CPU local.

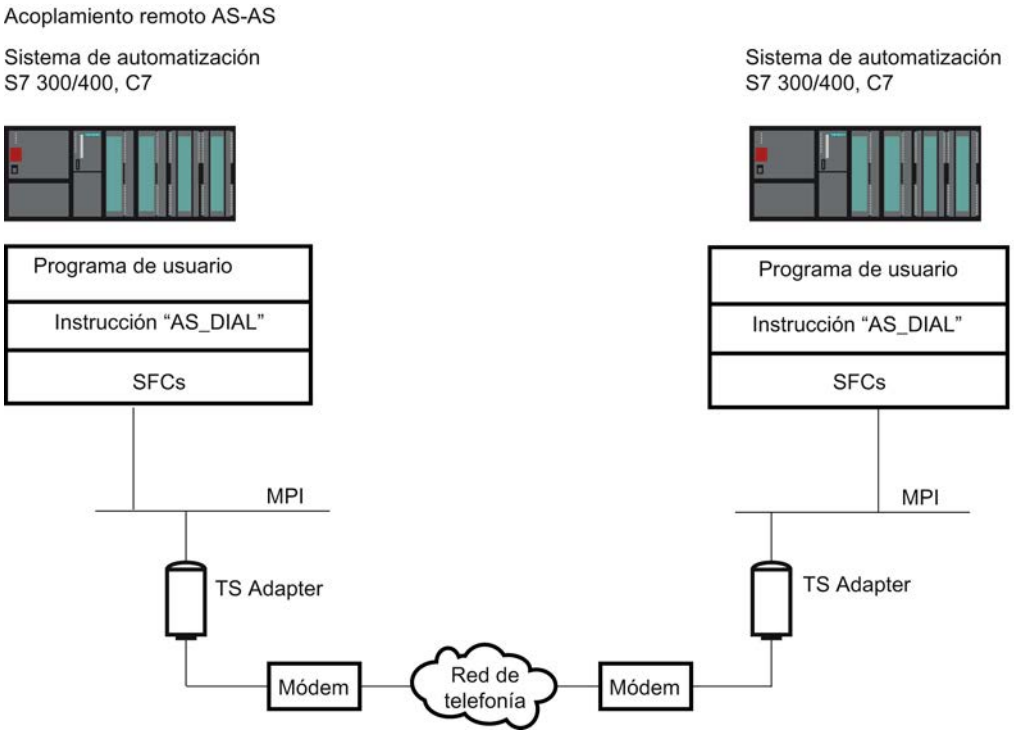


Figura 13-10 Requisitos de hardware y software para el acoplamiento remoto AS-AS

13.7.5.4 Enviar un SMS desde una instalación

Requisitos para enviar un mensaje SMS

Introducción

Para que la instalación pueda enviar un SMS se deben cumplir los requisitos de hardware y software siguientes. A continuación se describen estos requisitos.

Requisitos de hardware

Para enviar un mensaje SMS desde una instalación se requiere un módem GSM y un TS Adapter MPI.

Debe utilizarse un TS Adapter I con una versión V5.2 o superior o un TS Adapter II .

Requisitos de software en la instalación

En el volumen de suministro de TeleService está incluida la instrucción de comunicación "SMS_SEND", que se incorpora en la librería del TIA Portal durante la instalación en la Task Card, en la carpeta de instrucciones de comunicación de TeleService. Para que la instalación

pueda enviar el SMS es necesario que el programa de usuario que está cargado en la CPU de la instalación llame la instrucción de comunicación "SMS_SEND".

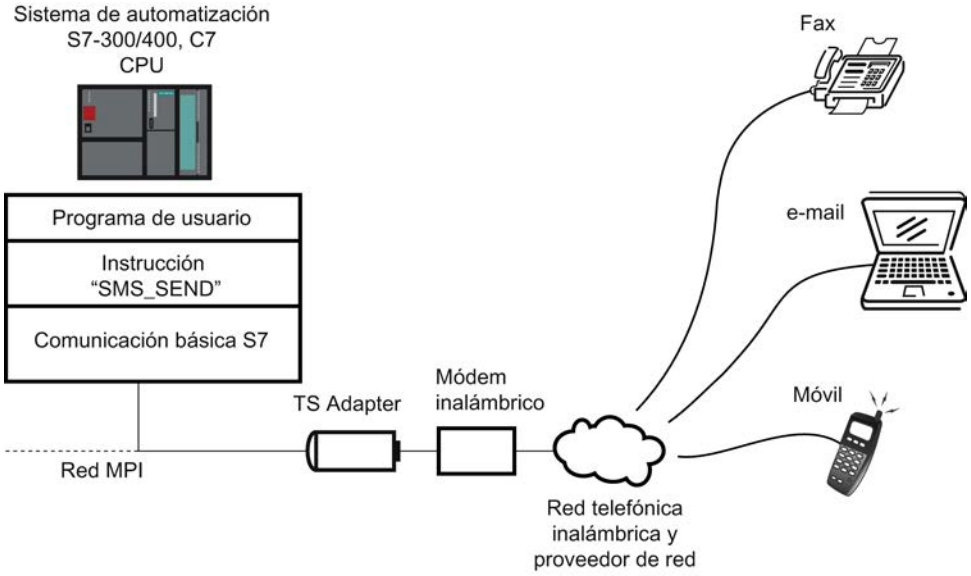


Figura 13-11 Funcionamiento de la instrucción de comunicación "SMS_SEND"

Nota

A través de los servicios adicionales de la compañía de telefonía móvil se puede enviar un SMS, si es necesario, no sólo a un móvil, sino también a una dirección de correo electrónico o a un fax.

13.7.5.5 Enviar un correo electrónico desde una instalación

Requisitos para enviar un correo electrónico

Introducción

Para que la instalación pueda enviar un correo electrónico se deben cumplir los requisitos de hardware y software siguientes:

Requisitos de hardware

Para enviar un correo electrónico desde una instalación se necesita un TS Adapter IE y una de las CPU indicadas a continuación:

- una CPU 31x-2 PN/DP a partir de la versión de firmware V2.5
- una CPU 41x-3 PN/DP

- una CPU de la familia S7-1200
- una CPU de la familia S7-1500

Requisitos de software en la instalación

En el volumen de suministro de TeleService se incluyen varias instrucciones de comunicación específicas de la CPU, que se incorporan durante la instalación en la Task Card, en la carpeta de instrucciones de comunicación de TeleService, en la librería del TIA Portal.

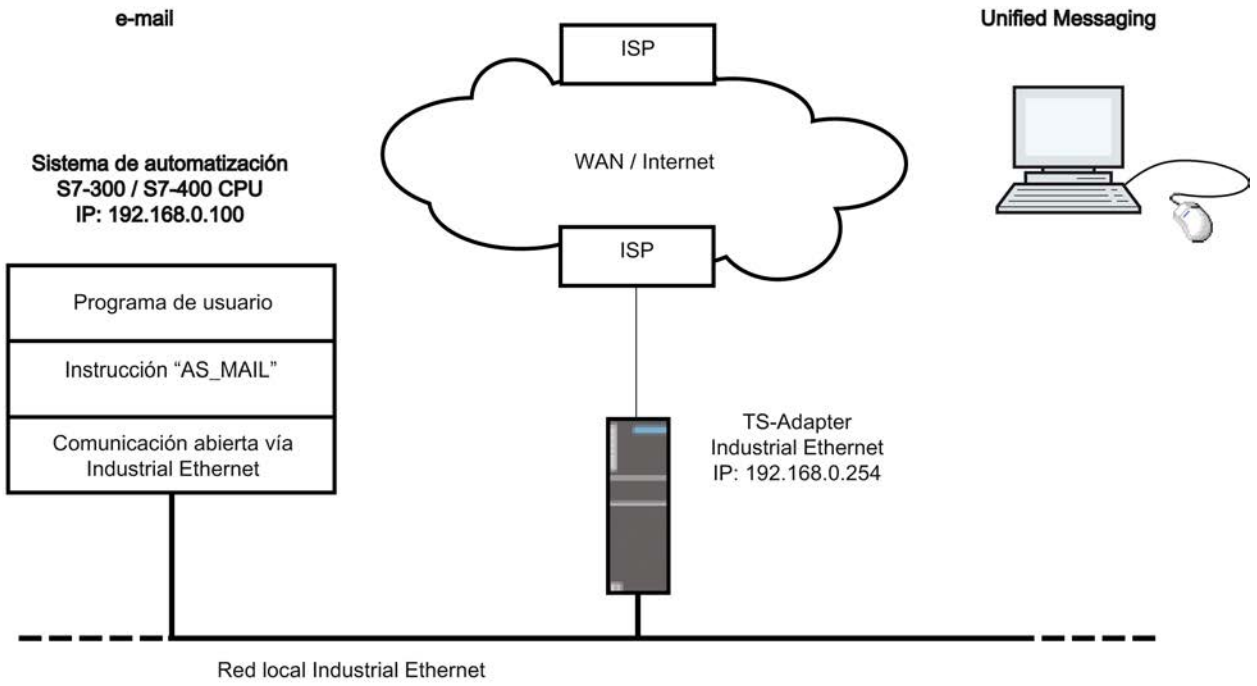
Para que la instalación pueda enviar un mensaje de correo electrónico es necesario que el programa de usuario de la instalación llame la instrucción de comunicación correspondiente específica de la CPU.

Existen las siguientes instrucciones de comunicación para enviar un correo electrónico:

- CPU S7-300/400: utiliza la instrucción de comunicación "AS_MAIL": Enviar e-mail
- CPU S7-1200 V 2.0 y V 3.0: utiliza la instrucción de comunicación "TM_MAIL": Enviar e-mail
- CPU S7-1200 >V3.0: utiliza la instrucción de comunicación "TMAIL_C": Enviar e-mail
- CPU S7-1500: utiliza la instrucción de comunicación "TMAIL_C": Enviar e-mail

Cada instrucción de comunicación transfiere un correo electrónico de una CPU a un servidor de correo electrónico a través del Simple Mail Transfer Protocol (SMTP) con el procedimiento de autenticación "LOGIN" . En este procedimiento SMTP se transfieren los datos sin cifrar.

La siguiente imagen muestra un ejemplo con la instrucción de comunicación "AS_MAIL":



Además, al configurar la CPU en la que se ejecuta la instrucción de comunicación "AS_MAIL" se debe ajustar la propiedad “Router / Utilizar router“ para la interfaz Ethernet. (Se encuentra en la configuración de dispositivos, en las direcciones Ethernet y, allí, en Protocolo IP) Como "Dirección" debe indicarse la dirección IP de la interfaz Ethernet del TS Adapter IE.

Nota

Encontrará más información al respecto en la Task Card de la carpeta "Instrucciones de comunicación" de TeleService.

13.7.6 Indicaciones para la búsqueda de errores

13.7.6.1 Indicaciones generales para la búsqueda de errores en caso de problemas con el módem

Introducción

Las siguientes indicaciones le facilitan la labor de encontrar y subsanar las causas de posibles problemas que se presenten con el módem:

1. Active la "grabación de un archivo de registro" para intercambiar datos entre la programadora/el PC y el módem. La grabación de este archivo puede proporcionar información valiosa para determinar las causas del error.
2. Conecte los altavoces de su módem local. Seleccione un volumen audible.
   De esta forma podrá oír si:
   – en la conexión existe una señal de línea libre,
   – el módem llamado está ocupado, o
   – el módem llamado recibe la llamada.

Problemas frecuentes del módem

Entre los problemas más frecuentes de los módems se incluye la conexión con el módem:

- No se establece la conexión con el módem.
- La conexión módem se interrumpe

En los siguientes temas encontrará tablas en las que se describen posibles causas y se proporcionan indicaciones para solucionar el error respectivo.

Consulte también

No se establece una conexión por línea telefónica con el TS Adapter (Página 11427)

No se establece una conexión por línea telefónica desde el TS Adapter (Página 11429)

La conexión módem se interrumpe (Página 11430)

Mensajes de error del módem (Página 11431)

Grabación de un archivo de informe para el módem (Página 11427)

13.7.6.2 Grabación de un archivo de informe para el módem

Introducción

Para poder localizar de forma más sencilla las causas de errores en un módem se recomienda permitir la grabación de un archivo de informe.

Procedimiento:

Proceda del siguiente modo:

1. Active a través del Panel de control, opción "Opciones de teléfono y módem", el diálogo de propiedades del módem utilizado.
2. Compruebe los ajustes de la opción "Informe" en la ficha "Diagnóstico" y modifique, si es necesario, los ajustes del archivo de informe para que éste se grabe.

Resultado:

Las actividades realizadas entre la programadora/el PC y el módem se introducen en el archivo de informe. Cuando se presenten problemas con el establecimiento de la conexión se puede analizar el archivo de informe para obtener indicaciones sobre la causa del error.

13.7.6.3 No se establece una conexión por línea telefónica con el TS Adapter

No se establece una conexión por línea telefónica con el TS Adapter

La siguiente tabla muestra posibles causas y la solución en caso de que no pueda establecerse ninguna conexión remota con el TS Adapter.

Causa posible	Comprobación/Solución
Cableado defectuoso	• ¿Están correctamente conectados todos los cables de conexión? • ¿Está flojo algún conector?
Ajuste erróneo de las propiedades de marcado para la centralita y el supletorio	• Las propiedades y los parámetros de marcado ajustados para su módem, ¿son adecuados para la línea de teléfono principal o secundaria? • Si utiliza el módem en una línea urbana (centralita), en el cuadro de diálogo "Propiedades de marcado" no debe indicar ningún prefijo para línea urbana. Los campos del prefijo de marcado para línea urbana deben estar vacíos para llamadas locales y llamadas de larga distancia.

Causa posible	Comprobación/Solución
Ajuste erróneo del método de marcado	• En el cuadro de diálogo de las propiedades de marcado de su módem, ¿está ajustado el método de marcado correcto (tonos/impulsos)? • Compruebe con un teléfono conectado, la toma en la que desee utilizar el módem. En caso de marcado por impulsos, durante el marcado pueden oírse en el teléfono ruidos de arrastre, y en caso de marcado por tonos, diferentes tonos. Ajuste en las propiedades de marcado del módem el procedimiento de marcado adecuado.
Bloqueo de marcado activado	• El bloqueo de marcado es una propiedad de módem prescrita según el país, y dependiendo del tipo de módem, se activa después de uno o varios intentos frustrados al establecer una conexión. • Si después de varios intentos de marcado el módem no reacciona, puede ser que esté activado el bloqueo de marcado. Después de la orden de marcado, al módem se le envían efectivamente señales, pero el módem no inicia el proceso de marcado. El driver recibe un mensaje de error general. • Infórmese sobre cómo se ha instalado el bloqueo de marcado en el módem. • Cree el archivo de informe (Página 11423) (modemlog.txt), donde se registrarán las acciones realizadas entre la programadora/el PC y el módem. Después compruebe si el archivo contiene una entrada (p. ej. DELAYED) causada por un bloqueo de marcado.
Conexión telefónica defectuosa u ocupada	• Conecte un teléfono y compruebe si en esa conexión puede oírse una señal de línea libre. • Si en la misma toma hay conectado un teléfono analógico, debe colgarse dicho teléfono. Si ya existe una conexión telefónica, puede ser que en esa toma no pueda establecerse una conexión módem adicional.
Ajuste erróneo de parámetros serie	• ¿Se han introducido los valores correctos en la ficha "Conexión" de las propiedades del módem (8 bits de datos, sin paridad, 1 bit de parada)? ¿Se ha ajustado la interfaz COM correcta en la ficha "General" de las propiedades del módem?
La cadena de inicialización del TS Adapter no se corresponde con el módem.	• Infórmese sobre los requisitos de la cadena de inicialización del módem y ajústela según los requisitos. Procedimiento para parametrizar el TS Adapter IE (Página 11395)
Las preferencias ajustadas para la corrección de errores entre el módem del TS Adapter y el módem de la programadora/del PC no son compatibles.	• Adapte las preferencias del módem. Información importante sobre la parametrización del TS Adapter MPI (Página 11390) Restablecer la parametrización estándar de un TS Adapter MPI (Página 11392) Procedimiento para parametrizar el TS Adapter IE (Página 11391)

13.7.6.4 No se establece una conexión por línea telefónica desde el TS Adapter

No se realiza la rellamada desde el TS Adapter

La siguiente tabla muestra posibles causas y su solución en caso de que no pueda realizarse ninguna rellamada desde el TS Adapter.

Causa posible	Comprobación/Solución
Ajustes erróneos de ubicación y llamada en el TS Adapter	Compruebe la parametrización del TS Adapter: • ¿Están correctamente ajustados el método de marcado y el prefijo de marcado para línea urbana su teléfono? • ¿El módem del TS Adapter soporta los caracteres parametrizados para el prefijo de marcado para línea urbana? • ¿Está desactivado "Esperar señal antes de marcar" en una estación secundaria?
Inicialización insuficiente del módem	Compruebe la cadena de inicialización del módem: • Tal vez el módem necesite una inicialización más amplia para poder establecer una conexión remota. • Propiedades de la cadena de inicialización del módem en el TS Adapter MPI
El número de rellamada no coincide	Compruebe la parametrización del número de rellamada que ha asignado.

No se realiza la llamada desde el TS Adapter MPI

La siguiente tabla muestra posibles causas y su solución en caso de que no pueda realizarse ninguna llamada desde el TS Adapter MPI.

Causa posible	Comprobación/Solución
El número de teléfono no coincide	¿Transfiere la instrucción de comunicación "PG_DIAL" el número de teléfono deseado?
La parametrización del TS Adapter MPI es incorrecta	Compruebe la parametrización del TS Adapter MPI: • ¿Están correctamente ajustados el método de marcado y el prefijo de marcado para línea urbana su teléfono? • ¿El módem del TS Adapter MPI soporta los caracteres parametrizados para el prefijo de marcado para línea urbana? • ¿Está desactivado "Esperar señal antes de marcar" en una estación secundaria?

13.7.6.5 La conexión módem se interrumpe

Se interrumpe la conexión módem

La siguiente tabla muestra posibles causas y su solución en caso de que se interrumpa la conexión módem.

Causa posible:	Comprobación/Solución:
Impulsos de tarificación en la línea	Si se ha solicitado a la compañía telefónica una frecuencia de tarificación, se generan impulsos de tarificación. Estos impulsos pueden impedir que el módem reconozca la señal portadora y se desconecte. • Ajuste un tiempo de espera o de desconexión mayor en su módem. • Solicite a la compañía telefónica que desactive la frecuencia de tarificación.
Blindaje	• ¿Están suficientemente blindados los cables de conexión utilizados? • Asegúrese de que las líneas de entrada al módem no pasen cerca de cables de red y que además guardan la mayor distancia posible con fuentes de alimentación y pantallas.
Timeout de informe	• Ajuste los tiempos de vigilancia fijos.
Desconexión automática	• Desactive la opción de autodesconexión de una conexión existente al transcurrir un tiempo determinado sin efectuar ninguna transferencia de datos ("Desconectar la llamada si está inactiva durante más de ...").
Utilizar control de flujo desactivado	• En la ficha "Conexión" de las propiedades del módem haga clic en el botón "Avanzada" y active en el cuadro de diálogo que aparece a continuación las siguientes opciones (si se requieren y aún no están activadas): – Control del flujo de datos – Hardware (RTS/CTS) – Compresión de datos – Controles de error
La cadena de inicialización del TS Adapter no se corresponde con el módem	• Ajuste la cadena de inicialización del módem según los requisitos: Consulte también: Posibilidades de parametrización del TS Adapter IE (Página 11398)

Consulte también

Posibilidades de parametrización para el TS Adapter MPI (Página 11390)

13.7.6.6 Lista de chequeo en caso de dificultades con el módem

Introducción

La siguiente lista le ayudará a encontrar las posibles causas en caso de tener problemas con el módem. En los temas de ayuda que figuran a continuación encontrará cómo y en qué cuadros de diálogo deberá definir los ajustes mencionados.

No se establece la conexión módem:

- Compruebe el cableado y los conectores.
- Compruebe si el método de marcado (tonos/pulsos) está ajustado correctamente.
- Si después de varios intentos de marcado fracasados el módem sigue sin reaccionar, puede ser que esté activado el bloqueo del marcado. Infórmese sobre cómo está implementado el bloqueo de marcado en el módem.
- ¿Se está utilizando el módem con una toma principal o con una estación secundaria? Ajuste de forma correspondiente las propiedades y las propiedades de marcado del módem.
- Active en las propiedades ampliadas la opción para registrar un archivo de protocolos. La siguiente conexión que se establezca se graba en el archivo de registros del directorio de Windows.
- Vigile que los TA ISDN empleados utilicen un mismo protocolo del canal B y D.

La conexión módem se interrumpe:

- Los pulsos de tarificación pueden afectar de forma negativa a una conexión. Solicite a su proveedor de servicios de telefonía que desconecte los pulsos de tarificación.
- Ajuste los tiempos de vigilancia fijos.
- Desactive la opción de autodesconexión de una conexión existente al transcurrir un tiempo determinado sin efectuar ninguna transferencia de datos (inactividad).
- Asegúrese de haber activado RTS/CTS para el control del flujo de datos.

13.7.6.7 Mensajes de error del módem

Información visualizada en el archivo de informe

Los mensajes del módem se guardan en un archivo de informe siempre que se haya activado su grabación.

El archivo de informe contiene la siguiente información:

Aviso:	Causa posible:	Solución:
NO DIALTONE	En este momento puede estar efectuándose una conversación telefónica en la misma línea.	• Repita el proceso cuando haya terminado la conversación telefónica en dicha línea.
NO CARRIER	La estación seleccionada no está preparada, no existe módem o no puede establecer una conexión con el modo operativo ajustado.	• Compruebe los números de teléfono y los ajustes.
BUSY	La estación seleccionada está ocupada.	• Vuelva a intentarlo de nuevo más tarde.
DELAYED:	Bloqueo de marcado	• Infórmese por medio de la documentación sobre cómo se ha instalado en su módem el bloqueo de marcado y elimínelo si así lo desea.

13.7.6.8 Posibles mensajes de error en las conexiones VPN

No se establece la conexión VPN con el TS Adapter IE Advanced

La siguiente tabla muestra posibles errores y la solución en caso de que no pueda establecerse ninguna conexión VPN con el TS Adapter IE Advanced.

Posibles errores	Comprobación/Solución
Al acceder a la interfaz web, en el explorador aparece el mensaje de error "No se puede mostrar la página web".	• Nota: Este problema se produce si en el almacén de certificados de Windows hay un certificado CA que tiene el mismo nombre que el certificado CA recién creado. • Si en el almacén de certificados de Windows ya hay instalado un certificado CA para este TS Adapter y se ha generado un nuevo certificado CA en el TS Adapter, quite el certificado CA antiguo del almacén de certificados de Windows e instale el certificado CA recién generado. • Consulte también: Instalar certificado CA para conexión VPN (Página 11405) • Consulte también: Borrar certificado CA para conexión VPN (Página 11408)
Mensaje de error: "Se ha producido un error interno en la aplicación".	• Compruebe si se han creado más usuarios en la base de datos de usuario del TS Adapter IE Advanced. • El usuario "Administrador" no puede establecer ninguna conexión remota.
Mensaje de error: "No se ha encontrado ningún certificado CA apropiado en el almacén de certificados de Windows".	• Utilice la descarga automática de certificados. Para ello, introduzca la huella digital en el campo correspondiente del cuadro de diálogo de conexión. • Consulte también: Establecer conexión VPN (Página 11408) O bien: • Exporte el certificado CA de la interfaz web del TS Adapter IE Advanced e instálelo manualmente en el almacén de certificados de Windows. • Consulte también: Instalar certificado CA para conexión VPN (Página 11405)
Mensaje de error: "La dirección remota indicada no concuerda con la dirección remota del TS Adapter".	• Para establecer la conexión debe utilizar la dirección remota que ha indicado en el TS Adapter IE Advanced. • No es posible utilizar la dirección IP estando registrado el nombre DNS (o viceversa).
Mensaje de error: "No se puede comprobar la firma del certificado".	• Si ha generado un nuevo certificado CA en el TS Adapter IE Advanced, debe borrar el certificado CA antiguo del almacén de certificados de Windows e instalar el nuevo. • Consulte también: Borrar certificado CA para conexión VPN (Página 11408) • Consulte también: Instalar certificado CA para conexión VPN (Página 11405)
Mensaje de error: "Error de informe"	• Compruebe si en la configuración del TS Adapter IE Advanced está definida la dirección IP del PC de mantenimiento en la red de la instalación (Parámetros > Red de la instalación).
Mensaje de error: "Se produjo un error al establecer la conexión".	• Compruebe si la dirección IP LAN del TS Adapter IE Advanced y la dirección IP de una de sus tarjetas de red se encuentran en la misma subred. En tal caso, desactive esta tarjeta de red antes de establecer la conexión remota.
No se pueden enviar correos electrónicos.	• Compruebe si existe una conexión a la red/a Internet en el puerto WAN. • Compruebe si el servidor de correo es accesible. • Compruebe si en el TS Adapter IE Advanced se admiten las conexiones salientes. • Compruebe si el puerto SMTP está abierto en el firewall para conexiones salientes.

Documentación del hardware 14

14.1 Información general sobre la documentación del hardware

Información adicional sobre el hardware disponible

Dependiendo de los productos instalados, el TIA Portal permite configurar una gran cantidad de hardware. El hardware disponible está recogido en el catálogo de hardware. Encontrará todos los manuales, instrucciones de servicio y FAQ actuales, así como las actualizaciones de los dispositivos, en el Siemens Industry Online Support (https://support.automation.siemens.com/).

Para facilitar la localización en el Siemens Industry Online Support de los documentos adecuados para el hardware utilizado, en los siguientes capítulos de ayuda encontrará todos los módulos y familias de módulos que pueden instalarse actualmente en el TIA Portal. Para cada módulo encontrará un enlace que le llevará directamente a los manuales e instrucciones de servicio correspondientes en el Siemens Industry Online Support.

14.2 PLC

14.2.1 SIMATIC S7-1200

14.2.1.1 CPU

CPU 1211C (6ES7 211-1xxx-0XB0)

Encontrará información sobre las CPU aquí:

- CPU 1211C AC/DC/Rly (6ES7 211-1BD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111BD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C AC/DC/Rly (6ES7 211-1BE31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111BE310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C AC/DC/Rly (6ES7 211-1BE40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111BE400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- CPU 1211C DC/DC/DC (6ES7 211-1AD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111AD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C DC/DC/DC (6ES7 211-1AE31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111AE310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C DC/DC/DC (6ES7 211-1AE40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111AE400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C DC/DC/Rly (6ES7 211-1HD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111HD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C DC/DC/Rly (6ES7 211-1HE31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111HE310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1211C DC/DC/Rly (6ES7 211-1HE40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72111HE400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

CPU 1212C (6ES7 212-1xxx-0XB0)

Encontrará información sobre las CPU aquí:

- CPU 1212C AC/DC/Rly (6ES7 212-1BD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121BD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C AC/DC/Rly (6ES7 212-1BE31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121BE310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C AC/DC/Rly (6ES7 212-1BE40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121BE400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- CPU 1212C DC/DC/DC (6ES7 212-1AD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121AD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C DC/DC/DC (6ES7 212-1AE31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121AE310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C DC/DC/DC (6ES7 212-1AE40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121AE400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C DC/DC/Rly (6ES7 212-1HD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121HD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C DC/DC/Rly (6ES7 212-1HE31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121HE310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1212C DC/DC/Rly (6ES7 212-1HE40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72121HE400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

CPU 1214C (6ES7 214-1xxx-0XB0)

Encontrará información sobre las CPU aquí:

- CPU 1214C AC/DC/Rly (6ES7 214-1BE30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&siteid=csius&aktprim=0&extranet=standard&viewreg=WW&objid=41886031&treeLang=de))
- CPU 1214C AC/DC/Rly (6ES7 214-1BG31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141BG310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1214C AC/DC/Rly (6ES7 214-1BG40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141BG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- CPU 1214C DC/DC/DC (6ES7 214-1AE30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141AE300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1214C DC/DC/DC (6ES7 214-1AG31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141AG310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1214C DC/DC/DC (6ES7 214-1AG40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141AG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1214C DC/DC/Rly (6ES7 214-1HE30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141HE300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1214C DC/DC/Rly (6ES7 214-1HG31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141HG310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1214C DC/DC/Rly (6ES7 214-1HG40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72141HG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

CPU 1215C (6ES7 215-1xxx-0XB0)

Encontrará información sobre las CPU aquí:

- CPU 1215C DC/DC/DC (6ES7 215-1BG31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72151BG310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1215C DC/DC/DC (6ES7 215-1BG40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72151BG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1215C AC/DC/Rly (6ES7 215-1AG31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72151AG310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- CPU 1215C AC/DC/Rly (6ES7 215-1AG40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72151AG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1215C DC/DC/Rly (6ES7 215-1HG31-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72151HG310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- CPU 1215C DC/DC/Rly (6ES7 215-1HG40-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72151HG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

CPU 1217C (6ES7 217-1xxx-0XB0)

Encontrará información sobre la CPU 1217C DC/DC/DC aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72171AG400XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.2.1.2 Signal Boards

Signal Boards (6ES7 2xx-xxx30-0XB0)

Encontrará información sobre las Signal Boards para el S7-1200 aquí:

- DI 4x24VDC (6ES7 221-3BD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72233BD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- DI 4x5VDC (6ES7 221-3AD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72213AD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 4x24VDC (6ES7 222-1BD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221BD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- DQ 4x5VDC (6ES7 222-1AD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221AD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))

- DI 2/DQ 2x24VDC (6ES7 223-0BD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72230BD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- DI 2/DQ 2x24VDC (6ES7 223-3BD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72213BD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 2/DQ 2x5VDC (6ES7 223-3AD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72233AD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- AI 1x12BIT (6ES7 231-4HA30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72314HA300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- AI 1xRTD (6ES7 231-5PA30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315PA300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- AI 1xTC (6ES7 231-5QA30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315QA300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))
- AQ 1x12BIT (6ES7 232-4HA30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72324HA300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=viewB))

14.2.1.3 Módulos de comunicación

Punto a punto

CB 1241 (6ES7 241-1CH30-1XB0)

Encontrará información sobre el módulo de comunicaciones CB 1241 aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72411CH301XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.2.1.4 Battery Boards

BB 1297 (6ES7 297-0AX30-0XA0)

Encontrará información sobre el módulo de comunicación BB 1297 aquí (http://support.automation.siemens.com/WW/llisapi.dll?aktprim=100&lang=es&referer=%2fWW%2f&func=cslib.cssearch&siteid=csius&extranet=standard&viewreg=WW&groupid=4000002&objaction=cssearch&content=adsearch%2Fadsearch%2Easpx).

14.2.1.5 Módulos de entradas digitales

Módulos de entradas digitales (6ES7 221-1Bx3x-0XB0)

Encontrará información sobre los módulos de entradas digitales aquí:

- DI8 x 24VDC (6ES7 221-1BF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72211BF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI8 x 24VDC (6ES7 221-1BF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72211BF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI16 x 24VDC (6ES7 221-1BH30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72211BH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI16 x 24VDC (6ES7 221-1BH32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72211BH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

14.2.1.6 Módulos de salidas digitales

Módulos de salidas digitales (6ES7 222-1xx3x-0XB0)

Encontrará información sobre los módulos de salidas digitales aquí:

- DQ 8x24VDC (6ES7 222-1BF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221BF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 8x24VDC (6ES7 222-1BF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221BF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 16x24VDC (6ES7 222-1BH30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221BH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 16x24VDC (6ES7 222-1BH32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221BH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 16xRelay (6ES7 222-1HH30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221HH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 16xRelay (6ES7 222-1HH32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221HH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 8xRelay (6ES7 222-1HF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221HF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 8xRelay (6ES7 222-1HF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221HF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- DQ 8xNO/NC Relay (6ES7 222-1XF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221XF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DQ 8xNO/NC Relay (6ES7 222-1XF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72221XF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

14.2.1.7 Módulos de entradas y salidas digitales

Módulos de entradas y salidas digitales (6ES7 223-1xx3x-0XB0)

Encontrará información sobre los módulos de entradas y salidas digitales aquí:

- DI 8/DQ 8x24VDC (6ES7 223-1BH30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231BH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 8/DQ 8x24VDC (6ES7 223-1BH32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231BH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 16/DQ 16x24VDC (6ES7 223-1BL30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231BL300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 16/DQ 16x24VDC (6ES7 223-1BL32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231BL320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 8x24VDC/DQ 8xRelay (6ES7 223-1PH30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231PH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 8x24VDC/DQ 8xRelay (6ES7 223-1PH32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231PH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI 16x24VDC/DQ 16xRelay (6ES7 223-1PL30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231PL300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- DI 16x24VDC/DQ 16xRelay (6ES7 223-1PL32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231PL320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI/DO 8x120VAC/DQ 8xRelay (6ES7 223-1QH30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231QH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- DI/DO 8x120VAC/DQ 8xRelay (6ES7 223-1QH32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72231QH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

14.2.1.8 Módulos de entradas analógicas

Módulos de entradas analógicas (6ES7 231-xxx3x-0XB0)

Encontrará información sobre los módulos de entradas analógicas aquí:

- AI 4x13BIT (6ES7 231-4HD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72314HD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4x13BIT (6ES7 231-4HD32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72314HD320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4x16BIT (6ES7 231-5ND30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315ND300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4x16BIT (6ES7 231-5ND32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315ND320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 8x13 BITS (6ES7 231-4HF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72314HF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 8x13BIT (6ES7 231-4HF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72314HF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

- AI 4xRTD (6ES7 231-5PD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315PD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4xRTD (6ES7 231-5PD32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315PD320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 8xRTD (6ES7 231-5PF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES723145PF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 8xRTD (6ES7 231-5PF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315PF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4xTC (6ES7 231-5QD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315QD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4xTC (6ES7 231-5QD32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315QD320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 8xTC (6ES7 231-5QF30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315QF300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 8xTC (6ES7 231-5QF32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72315QF320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

14.2.1.9 Módulos de salidas analógicas

Módulos de salidas analógicas (6ES7 234-4Hx3x-0XB0)

Encontrará información sobre los módulos de salidas analógicas aquí:

- AQ 2x14BIT (6ES7 232-4HB30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72344HB300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AQ 2x14BIT (6ES7 232-4HB32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72344HB320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AQ 4x14BIT (6ES7 232-4HD30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72344HD300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AQ 4x14BIT (6ES7 232-4HD32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72344HD320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

14.2.1.10 Módulos de entradas y salidas analógicas

Módulo de entradas y salidas analógicas (6ES7 234-4HE3x-0XB0)

Encontrará información sobre los módulos de entradas y salidas analógicas aquí:

- AI 4x13BIT/AQ 2x14BIT (6ES7 234-4HE30-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72324HB300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))
- AI 4x13BIT/AQ 2x14BIT (6ES7 234-4HE32-0XB0 (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72324HB320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view))

14.2.1.11 Módulos de comunicación

Industrial Remote Communication

CP 1242-7 (6GK7 242-7KX30-0XE00)

Encontrará información sobre el módulo de comunicaciones Telecontrol CP 1242-7 aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK72427KX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CP 1243-1 (6GK7 243-1JX30-0XE0)

Encontrará información sobre el módulo de comunicaciones Ethernet CP 1243-1 aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK72431JX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

PROFIBUS

CM 1242-5 (6GK7 242-5DX3x-0XE0)

Encontrará información sobre el módulo de comunicaciones PROFIBUS CM 1242-5 aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK72425DX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM 1243-5 (6GK7 243-5DX3x-0XE0)

Encontrará información sobre el módulo de comunicaciones PROFIBUS CM 1243-5 aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK72435DX300XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Punto a punto

CM 1241 RS232 (6ES7 241-1AH30-0XB0)

Encontrará información sobre los módulos de comunicaciones para conexiones punto a punto CM 1241 (RS232) aquí (http://support.automation.siemens.com/WW/llisapi.dll?

func=cslib.csinfo&lang=es&objid=6ES72411AH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM 1241 RS232 (6ES7 241-1AH32-0XB0)

Encontrará información sobre los módulos de comunicaciones para conexiones punto a punto CM 1241 (RS232) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72411AH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM 1241 RS485 (6ES7 241-1CH30-0XB0)

Encontrará información sobre los módulos de comunicaciones para conexiones punto a punto CM 1241 (RS485) aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=en&objid=6ES72411CH300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM 1241 RS422/485 (6ES7 241-1CH31-0XB0)

Encontrará información sobre los módulos de comunicaciones para conexiones punto a punto CM 1241 (RS422/485) aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72411CH310XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM 1241 RS422/485 (6ES7 241-1CH32-0XB0)

Encontrará información sobre los módulos de comunicaciones para conexiones punto a punto CM 1241 (RS422/485) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72411CH320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Sistemas de identificación

RF120C (6GT2 002-0LA00)

Encontrará información sobre el módulo de comunicaciones para sistemas de identificación RFC120C aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GT20020LA00&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AS-Interface

AS-Interface CM 1243-2 (3RK7 243-2AA30-0XB0)

Encontrará información sobre el módulo de comunicaciones AS-i CM 1243-2 aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=3RK72432AA300XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.2.1.12 Módulos tecnológicos

SIWAREX WP231 (7MH4 960-2AA01)

Encontrará información sobre el módulo de pesaje SIWAREX WP231 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=7MH4960-2AA01&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

SIWAREX WP241 (7MH4 960-4AA01)

Encontrará información sobre el módulo de pesaje SIWAREX WP241 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=7MH4960-4AA01&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

4SI IO-Link (6ES7 278-4BD32-0XB0)

Encontrará información sobre el módulo tecnológico IO-Link-Master 4SI IO-Link aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES72784BD320XB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.2.2 SIMATIC S7-1500

14.2.2.1 CPU

CPU 1511-1 PN (6xxx 511-1AK00-xAB0)

Encontrará información sobre la CPU 1511-1 PN (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?

func=cslib.csinfo&lang=es&objid=6ES75111AK000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1513-1 PN (6xxx 513-1AL00-xAB0)

Encontrará información sobre la CPU 1513-1 PN (también variante SIPLUS) aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75131AL000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1515-2 PN (6ES7 515-2AM00-0AB0)

Encontrará información sobre la CPU 1515-2 PN aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75152AM000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1516-3 PN/DP (6xxx 516-3AN00-xAB0)

Encontrará información sobre la CPU 1516-3 PN/DP (también variante SIPLUS) aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75163AN000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1518-3 PN/DP (6ES7 518-3AN00-0AB0)

Encontrará información sobre la CPU 1518-3 PN/DP aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75183AN000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1518-4 PN/DP (6ES7 518-4AP00-0AB0)

Encontrará información sobre la CPU 1518-4 PN/DP aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75184AP000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1511F-1 PN (6ES7 511-1FK00-0AB0)

Encontrará información sobre la CPU 1511F-1 PN aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75111FK000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1513F-1 PN (6ES7 513-1FL00-0AB0)

Encontrará información sobre la CPU 1513F-1 PN aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75131FL000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1515F-2 PN (6ES7 515-2FM00-0AB0)

Encontrará información sobre la CPU 1515F-2 PN aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75152FM000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1516F-3 PN/DP (6ES7 516-3FN00-0AB0)

Encontrará información sobre la CPU 1516F-1 PN/DP aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75163FN000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1518F-4 PN/DP (6ES7 518-4FP00-0AB0)

Encontrará información sobre la CPU 1518F-4 PN/DP aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75184FP000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.2.2.2 Otros módulos para S7-1500

Los módulos de S7-1500 también pueden utilizarse para el módulo de periferia descentralizada ET 200MP. Por lo tanto, encontrará la información sobre los módulos de ampliación en el capítulo "ET 200MP (Página 11450)".

Consulte también

ET 200MP (Página 11450)

14.3 Periferia descentralizada

14.3.1 ET 200MP

14.3.1.1 Módulos de interfaz

PROFINET

IM 155-5 PN ST (6ES7 155-5AA00-0AB0)

Encontrará información sobre el módulo de periferia descentralizada IM 155-5 PN ST aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71555AA000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

IM 155-5 PN HF (6ES7 155-5AA00-0AC0)

Encontrará información sobre el módulo de periferia descentralizada IM 155-5 PN HF aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71555AA000AC0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.2 Módulos de entradas digitales

DI 16x24VDC BA (6ES7 521-1BH10-0AA0)

Encontrará información sobre el módulo de entradas digitales DI 16x24VDC BA aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75211BH100AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 16x24VDC HF (6ES7 521-1BH00-0AB0)

Información sobre el módulo de entradas digitales DI 16x24VDC HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?

func=cslib.csinfo&lang=es&objid=6ES75211BH000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 16x24VDC SRC BA (6xxx 521-1BH50-xAA0)

Encontrará información sobre el módulo de entradas digitales DI 16x24VDC HF (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75211BH500AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 16x230VAC BA (6xxx 521-1FH00-xAA0)

Encontrará información sobre el módulo de entradas digitales DI 16x24VDC HF (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75211FH000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 32x24VDC HF (6ES7 521-1BL00-0AB0)

Información sobre el módulo de entradas digitales DI 16x24VDC HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75211BL000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 32x24VDC BA (6ES7 521-1BL10-0AA0)

Encontrará información sobre el módulo de entradas digitales DI 16x24VDC BA aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75211BL100AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.3 Módulos de salidas digitales

DQ 8x24VDC/2A HF (6xxx 522-1BF00-xAB0)

Encontrará información sobre el módulo de salidas digitales DQ 8x24VDC/2A HF (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75221BF000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 8x230VAC/2A ST (6xxx 522-5FF00-xAB0)

Encontrará información sobre el módulo de salidas digitales DQ 8x230VAC/2A ST (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75225FF000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 8x230VAC/5A ST (6xxx 522-5HF00-xAB0)

Encontrará información sobre el módulo de salidas digitales 8x230VAC/5A ST(también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75225HF000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 16x24VDC/0.5A BA (6ES7 522-1BH10-0AA0)

Encontrará información sobre el módulo de salidas digitales DQ 16x24VDC/0.5A BA aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75221BH100AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 16x24VDC/0.5A ST (6ES7 522-1BH00-0AB0)

Encontrará información sobre el módulo de salidas digitales DQ 16x24VDC/0.5A ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75221BH000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 32x24VDC/0.5A BA (6ES7 522-1BL10-0AA0)

Encontrará información sobre el módulo de salidas digitales DQ 32x 24VDC/0.5A BA aquí (http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75221BL100AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 32x24VDC/0.5A ST (6ES7 522-1BL00-0AB0)

Encontrará información sobre el módulo de salidas digitales DQ 32x 24VDC/0.5A aquí (https://support.automation.siemens.com/WW/llisapi.dll?

func=cslib.csinfo&lang=es&objid=6ES75221BL000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.4 Módulos de entradas y salidas digitales

DI16/DO 16x24VDC (6ES7 523-1BL00-0AA0)

Encontrará información sobre el módulo de entradas y salidas digitales DI 16/DO 16x24VDC aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75231BL000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.5 Módulos de entradas analógicas

AI 4xU/I/RTD/TC ST (6ES7 531-7QD00-0AB0)

Encontrará información sobre el módulo de entradas analógicas AI 4xU/I/RTD/TC ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75317QD00AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 8xU/I/RTD/TC ST (6ES7 531-7KF00-0AB0)

Encontrará información sobre el módulo de entradas analógicas AI 8xU/I/RTD/TC ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75317KF000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 8xU/I HS (6xxx 531-7NF10-xAB0)

Encontrará información sobre el módulo de entradas analógicas AI 8xU/I HS (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75317NF100AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.6 Módulos de salidas analógicas

AQ 2xU/I ST (6ES7 532-5NB00-0AB0)

Encontrará información sobre el módulo de salidas analógicas AQ 2xU/I ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75325NB000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AQ 4xU/I ST (6ES7 532-5HD00-0AB0)

Encontrará información sobre el módulo de salidas analógicas AQ 4xU/I ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75325HD000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AQ 8xU/I HS (6xxx 532-5HF00-xAB0)

Encontrará información sobre el módulo de salidas analógicas AQ 8xU/I HS (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75325HF000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.7 Módulos de entradas y salidas analógicas

AI/AQ 4xU/I/RTD/TC / 2xU/I ST (6ES7 534-7QE00-0AB0)

Encontrará información sobre el módulo de entradas y salidas analógicas AI/AQ 4xU/I/RTD/TC / 2xU/I ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75347QE000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.8 Módulos de comunicación

PROFINET/Ethernet

CM 1542-1 (6GK7 542-1AX00-0XE)

Encontrará información sobre el módulo de comunicaciones PROFINET/Ethernet CM 1542-1 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK75421AX000XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CP 1543-1 (6GK7 543-1AX00-0XE0)

Encontrará información sobre el procesador de comunicaciones PROFINET/Ethernet CP 1543-1 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK75431AX000XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

PROFIBUS

CM 1542-5 (6GK7 542-5DX00-0XE0)

Encontrará información sobre el módulo de comunicaciones PROFIBUS CM 1542-5 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6GK75425DX000XE0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Punto a punto

RS232

CM PtP RS232 BA (6xxx 540-1AD00-xAA0)

Encontrará información sobre el módulo de comunicación punto a punto CM PtP RS232 BA (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75401AD000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM PtP RS232 HF (6ES7 541-1AD00-0AB0)

Encontrará información sobre el módulo de comunicación punto a punto CM PtP RS232 HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75411AD000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

RS422/485

CM PtP RS422/485 BA (6xxx 540-1AB00-xAA0)

Encontrará información sobre el módulo de comunicación punto a punto CM PtP RS422/485 BA (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75401AB000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM PtP RS422/485 HF (6xxx 541-1AB00-xAB0)

Encontrará información sobre el módulo de comunicación punto a punto CM PtP RS422/485 HF (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75411AB000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.9 Módulos de alimentación

PS 25W 24VDC (6ES7 505-0KA00-0AB0)

Encontrará información sobre el módulo de alimentación PS 25W 24VDC aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75050KA000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

PS 60W 24/48/60VDC (6xxx 505-0RA00-xAB0)

Encontrará información sobre el módulo de alimentación PS 60W 24/48/60VDC (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75050RA000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

PS 60W 120/230VAC/DC (6xxx 507-0RA00-xAB0)

Encontrará información sobre el módulo de alimentación PS 60W 120/230VAC/DC (también variante SIPLUS) aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75070RA000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.1.10 Módulos tecnológicos

TM Count 2x24V (6ES7 550-1AA00-0AB0)

Encontrará información sobre el módulo contador TM Count 2x24V aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75501AA000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

TM PosInput 2 (6ES7 551-1AB00-0AB0)

Encontrará información sobre el módulo de posicionamiento TM PosInput 2 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75511AB000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CM AS-i Master (3RK7 137-6SA00-0BC1)

Encontrará información sobre el módulo CM AS-i Master aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=3RK71376SA000BC1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

F-CM AS-i (3RK7 136-6SC00-0BC1)

Encontrará información sobre el módulo F-CM AS-i aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=3RK71366SC000BC1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2 ET 200SP

14.3.2.1 CPU

CPU 1510SP-1 (6ES7 510-1DJ00-0AB0)

Encontrará información sobre la CPU 1510SP-1 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75101DJ000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

CPU 1512SP-1 (6ES7 512-1DK00-0AB0)

Encontrará información sobre la CPU 1512SP-1 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES75121DK000AB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.2 Módulos de interfaz

PROFINET

IM 155-6 PN ST (6ES7 155-6AU00-0BN0)

Información sobre el módulo de interfaz IM 155-6 PN ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71556AU000BN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

IM 155-6 PN HF (6ES7 155-6AU00-0CN0)

Información sobre el módulo de interfaz IM 155-6 PN HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71556AU000CN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

PROFIBUS

IM 155-6 DP HF (6ES7 155-6BU00-0CN0)

Información sobre el módulo de interfaz IM 155-6 DP HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71556BU000CN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Más buses de campo

IM 155-6 Receive (6ES7 155-6DU00-0BN0)

Información sobre el módulo de interfaz IM 155-6 Receive aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71556DU000BN0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.3 Módulos de entradas digitales

DI 4x120..230VAC ST (6ES7 131-6FD00-0BB1)

Encontrará información sobre el módulo de entradas digitales DI 4x120..230VAC ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71316FD000BB1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 8x24VDC ST (6ES7 131-6BF00-0BA0)

Encontrará información sobre el módulo de entradas digitales DI 8x24VDC ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71316BF000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 8x24VDC HF (6ES7 131-6BF00-0CA0)

Encontrará información sobre el módulo de entradas digitales DI 8x24VDC HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71316BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 8xNAMUR HF (6ES7 131-6TF00-0CA0)

Encontrará información sobre el módulo de entradas digitales DI 8xNAMUR HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71316TF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 8x24VDC SRC BA (6ES7 131-6BF60-0AA0)

Encontrará información sobre el módulo de entradas digitales DI 8x24VDC SRC BA aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71316BF600AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DI 16x24VDC ST (6ES7 131-6BH00-0BA0)

Encontrará información sobre el módulo de entradas digitales DI 16x24VDC ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71316BH000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.4 Módulos de salidas digitales

DQ 4x24VDC/2A ST (6ES7 132-6BD20-0BA0)

Encontrará información sobre el módulo de salidas digitales DQ 4x24VDC/2A ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326BD200BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 4x24..230VAC/2A ST (6ES7 132-6FD00-0BB1)

Encontrará información sobre el módulo de salidas digitales DQ 4x24..230VAC/2A ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326FD000BB1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 4x24VDC/2A HF (6ES7 132-6BD20-0CA0)

Encontrará información sobre el módulo de salidas digitales DQ 4x24VDC/2A HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?

func=cslib.csinfo&lang=es&objid=6ES71326BD200CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

RQ 4x120..230VUC/5A NO ST (6ES7 132-6HD00-0BB0)

Encontrará información sobre el módulo de salidas por relé RQ 4x120..230VUC/5A NO ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326HD000BB0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

RQ 4x24VUC/2A CO ST (6ES7 132-6GD50-0BA0)

Encontrará información sobre el módulo de salidas por relé RQ 4x24VUC/2A CO ST aquí.

DQ 8x24VDC/0.5A ST (6ES7 132-6BF00-0BA0)

Encontrará información sobre el módulo de salidas digitales DQ 8x24VDC/0.5A ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326BF000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 8x24VDC/0.5A SNK BA (6ES7 132-6BF60-0AA0)

Encontrará información sobre el módulo de salidas digitales DQ 8x24VDC/0.5A SNK BA aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326BF600AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 8x24VDC/0.5A HF (6ES7 132-6BF00-0CA0)

Encontrará información sobre el módulo de salidas digitales DQ 8x24VDC/0.5A HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326BF000CA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

DQ 16x24VDC/0.5A ST (6ES7 132-6BH00-0BA0)

Encontrará información sobre el módulo de salidas digitales DQ 16x24VDC/0.5A ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71326BH000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.5 Módulos de entradas analógicas

AI 2xU/I 2-/4-wire HF (6ES7 134-6HB00-0CA1)

Encontrará información sobre el módulo de entradas analógicas AI 2xU/I 2-/4-wire HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346HB000CA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 2xU/I 2-/4-wire HS (6ES7 134-6HB00-0DA1)

Encontrará información sobre el módulo de entradas analógicas AI 2xU/I 2-/4-wire HS aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346HB000DA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 4xI 2-/4-wire ST (6ES7 134-6GD00-0BA1)

Encontrará información sobre el módulo de entradas analógicas AI 4xI 2-/4-wire ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346GD000BA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 4xU/I 2-wire ST (6ES7 134-6HD00-0BA1)

Encontrará información sobre el módulo de entradas analógicas AI 4xU/I 2-wire ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346HD000BA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 4xRTD/TC 2-/3-/4-wire HF (6ES7 134-6JD00-0CA1)

Encontrará información sobre el módulo de entradas analógicas AI 4xRTD/TC 2-/3-/4-wire HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346JD000CA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI 8xRTD/TC 2-wire HF (6ES7 134-6JF00-0CA1)

Encontrará información sobre el módulo de entradas analógicas AI 8xRTD/TC 2-wire HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346JF000CA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AI EnergyMeter ST (6ES7 134-6PA00-0BD0)

Encontrará información sobre el módulo de salidas por relé AI EnergyMeter ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71346PA000BD0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.6 Módulos de salidas analógicas

AQ 2xU/I HF (6ES7 135-6HB00-0CA1)

Encontrará información sobre el módulo de salidas analógicas AQ 2xU/I HF aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71356HB000CA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AQ 2xU/I HS (6ES7 135-6HB00-0DA1)

Encontrará información sobre el módulo de salidas analógicas AQ 2xU/I HS aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71356HB000DA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

AQ 4xU/I ST (6ES7 135-6HD00-0BA1)

Encontrará información sobre el módulo de salidas analógicas AQ 4xU/I ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71356HD000BA1&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.7 Módulos de comunicación

CM PtP (6ES7 137-6AA00-0BA0)

Encontrará información sobre el módulo de comunicación CM PtP aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71376AA000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.8 Módulos de potencia

PM 120..230VAC/10A ST (6ES7 133-6AA00-0BC0)

Encontrará información sobre el módulo de potencia PM 120..230VAC/10A ST aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71336AA000BC0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.9 Módulos especiales

Módulo servidor (6ES7 193-6PA00-0AA0)

Encontrará información sobre el módulo servidor aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71936PA000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Adaptador de bus BA 2xSCRJ (6ES7 193-6AP00-0AA0)

Encontrará información sobre el adaptador de bus BA 2xSCRJ aquí.

Consulte también

Enlace al manual de producto (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71936AP000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view)

Adaptador de bus BA 2xRJ45 (6ES7 193-6AR00-0AA0)

Encontrará información sobre el adaptador de bus BA 2xRJ45 aquí (https://support.automation.siemens.com/WW/llisapi.dll?

func=cslib.csinfo&lang=es&objid=6ES71936AR000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Adaptador de bus FC (6ES7 193-6AF00-0AA0)

Encontrará información sobre el adaptador de bus FC aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71936AF000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Adaptador de bus Send (6ES7 193-6AS00-0AA0)

Encontrará información sobre el adaptador de bus Send aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71936AS000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Adaptador de bus Receive (6ES7 193-6AE00-0AA0)

Encontrará información sobre el adaptador de bus Receive aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71936AE000AA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

14.3.2.10 Módulos tecnológicos

TM Count 1x24V (6ES7 138-6AA00-0BA0)

Encontrará información sobre el módulo de salidas por relé TMCount 1x24V aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71386AA000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

TM PosInput 1 (6ES7 138-6BA00-0BA0)

Encontrará información sobre el módulo de posicionamiento TM PosInput 1 aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71386BA000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

IO-Link Master Module (6ES7 137-6BD00-0BA0)

Encontrará información sobre el módulo de salidas por relé IO-Link Master Module aquí (https://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&lang=es&objid=6ES71376BD000BA0&objaction=csviewmlfbbeitraege&subtype=133300&caller=view).

Índice alfabético

-

-, 2788
-D, 4132, 6226
-I, 4127, 6221
-R, 4141, 6235

6ES7 155-6AU00-0CN0, 11454
ADM_ZONE
  Parámetros de entrada/salida, 7609
  Parámetros de salida, 7609
  Variables estáticas, 7609
Archivo fuente externo
  Abrir, 2980
  Editar, 2980
ARRANQUE
  Rearranque (en caliente), 1624, 1903
  Rearranque completo (en caliente), 1704
Aumentar/reducir vista, (Véase Zoom)
Cadena de caracteres
  Convertir con S_CONV, 6716
Decalaje de señal cero
  Deshacer, 11025
Derechos de usuario , 2010
EstablecerColorRetroiluminación, 8694
Estación SIMATIC PC
  Cargar, 76
FMT_PID
  Parámetros de entrada/salida, 7533
Intervalo de medida, 9390, 9393
INVD, 4185, 6278
INVI, 4184, 6277
Llamada
  EncoderIM178, 7350
LP_ZONE
  Parámetros de entrada, 7606
  Parámetros de entrada/salida, 7606
  Parámetros de salida, 7606
  Variables estáticas, 7606
Master Control Relay (MCR)
  Funcionalidad, 2564
Módulo
  insertar, 523
Parametrización de la conexión
  General, 697
Parámetro
  EncoderIM178, 7351
PID_PAR
  Parámetros de salida, 7515
PULSEGEN
  Parámetros de entrada, 7097
  Parámetros de salida, 7098
Quality Code, 2141
READ_455
  Parámetros de salida, 7593
Rearranque completo (en caliente), 1704
Resto de división, 4139, 6233
Send_P2P (enviar datos punto a punto)
  Parámetros LENGTH y BUFFER, 5412, 7802
Valor absoluto, 3411, 3723, 4533, 5680, 5903, 6557

#

#, 2650

&

&, 2792, 2793

(

(), 2793

*

*, 2788
**, 2788
*.bmp, 8357
*.cer, 841
*.dat, 841
*.emf, 8357
*.gif, 8357
*.ico, 8357
*.jpeg, 8357
*.jpg, 8357
*.p12, 764, 841
*.tif, 8357
*.wmf, 8357
*D, 4133, 6227
*I, 4128, 6222
*R, 4142, 6237

/

/, 2788
/D, 4134, 6228
/FC, 2545
/I, 4129, 6223
/R, 4143, 6238

:

:=, 2793, 2794
:P, 1636, 1916

+

+, 2788
+AR1, 4277, 6370
+AR2, 4279, 6371
+D, 4131, 6225
+I, 4126, 6220
+R, 4140, 6234

<

<, 2790
<=, 2790
<>, 2790, 2793

=

=, 2790
==, 2793

>

>, 2790
>=, 2790

1

1Count24V, 9612
1Count5V, 9674
1PosU
  Activar JOBs, 11097
  Acuse de fallo EXTF_ACK, 11131
  Ajustar valor real (JOB 1), 11115
  Ajustar vigilancia de sentido de giro (JOB 11), 11124
  Búsqueda del punto de referencia (MODE 3), 11103
  Cancelar procesamiento de JOB (JOB 0), 11114
  Causas de error para JOB_ERR, 11133
  Causas de error para POS_ERR, 11132
  Control del accionamiento, 11098, 11141, 11147, 11153
  Detener (MODE 0), 11100
  Distancia de cambio de velocidad, 11141, 11148, 11153
  Distancia de desconexión, 11141, 11147, 11153
  Entradas del encóder, 11143
  Error en el control de MODEs y JOBs, 11131
  Errores de parametrización, 11128
  Errores externos, 11129
  Estado RESET, 11135
  Evaluación de las señales del encóder, 11098
  Evaluar señal de búsqueda de referencia (JOB 9), 11120
  Final de carrera por hardware, 11138, 11145, 11151
  Final del eje rotativo, 11137, 11144
  Funciones, 11094
  Habilitaciones en función del sentido, 11100
  Incrementos por vuelta del encóder, 11149
  Iniciar MODEs, 11096
  Interfaz de control, 11154
  Interfaz de realimentación, 11155
  Inversión del sentido de giro, 11137, 11143, 11150
  Jog (MODE 1), 11101
  Leva de reducción, 11139, 11146, 11152
  Modificar la distancia de cambio de velocidad (JOB 4), 11119
  Modificar la distancia de desconexión (JOB 3), 11118
  Modo de dosificación, 11095, 11136, 11143
  Mostrar valores actuales (JOB 15), 11125
  Mover zona de alcance del encóder (JOB 2), 11116
  Número de vueltas del encóder, 11150
  Posicionamiento absoluto (MODE 5), 11111
  Posicionamiento relativo (MODE 4), 11109
  Sensor del punto de referencia, 11140, 11146
  Sentido de arranque para la búsqueda del punto de referencia, 11140, 11147
  Señal de búsqueda de referencia, 11139, 11146
  STOP de CPU/maestro, 11134
  Tipo de eje, 11137, 11144, 11151
  Tipo de encóder, 11149
  Tmin Cambio de sentido, 11142, 11148, 11154
  Velocidad de transferencia, 11149

1SSI
Lectura de los valores del sensor, 10852, 10868, 10880
1STEP 5V
Aceleración/deceleración a, 11163
Búsqueda del punto de referencia, 11167
búsqueda del punto de referencia, desarrollo, 11168
curva de desplazamiento, 11161
Definición del punto de referencia, 11171
detener la petición de desplazamiento, 11175
eje de valor módulo, 11177
Entrada digital DI0 (3) , 11181
Entrada digital DI1 (7) , 11182
Frecuencia base, 11164
Frecuencia de inicio y parada Fss, 11162
Frecuencia de salida/velocidad Fa, 11163
Frecuencia/velocidad máxima del eje Fmax, 11162
habilitación de impulsos, 11179
inicio de una petición de desplazamiento, 11166
modificar parámetros durante el funcionamiento, 11180
Modo de velocidad, 11174
Modo incremental absoluto, 11173
Modo incremental relativo, 11172
reacción a STOP de la CPU/del maestro, 11182
Tipo de eje, 11177

2

2PULSE
Conexión en paralelo de ambos canales, 10415
contador de secuencias, 10411
Control directo de DO, 10412
Medición de intensidad, 10411
Modulación de ancho de impulsos, 10380
retardo a la conexión/desconexión, 10397
salida de frecuencia, 10405
salida de impulsos, 10375
tren de impulsos, 10390

3

3964(R)
Recibir datos, 1036
Transferir datos, 1035
3DES, 818
3RK7 136-6SC00-0BC1, 11454
3RK7 137-6SA00-0BC1, 11453
3RK7 243-2AA30-0XB0, 11443

6

6AG1 505-0RA00-0AB0, 11453
6AG1 507-0RA00-0AB0, 11453
6AG1 511-1AK00-2AB0, 11444
6AG1 513-1AL00-2AB0, 11444
6AG1 516-3AN00-2AB0, 11444
6AG1 521-1BH50-7AA0, 11447
6AG1 521-1FH00-7AA0, 11447
6AG1 522-1BF00-7AB0, 11448
6AG1 522-5FF00-7AB0, 11448
6AG1 522-5HF00-2AB0, 11448
6AG1 531-7NF10-7AB0, 11450
6AG1 532-5HF00-7AB0, 11450
6AG1 540-1AB00-7AA0, 11452
6AG1 540-1AD00-7AA0, 11452
6AG1 541-1AB00-7AB0, 11452
6ES7 131-6BF00-0BA0, 11455
6ES7 131-6BF00-0CA0, 11455
6ES7 131-6BF60-0AA0, 11456
6ES7 131-6BH00-0BA0, 11456
6ES7 131-6FD00-0BB1, 11455
6ES7 131-6TF00-0CA0, 11456
6ES7 132-6BD20-0BA0, 11456
6ES7 132-6BD20-0CA0, 11457
6ES7 132-6BF00-0BA0, 11457
6ES7 132-6BF00-0CA0, 11457
6ES7 132-6BF60-0AA0, 11457
6ES7 132-6BH00-0BA0, 11458
6ES7 132-6FD00-0BB1, 11456
6ES7 132-6GD50-0BA0, 11457
6ES7 132-6HD00-0BB0, 11457
6ES7 133-6AA00-0BC0, 11460
6ES7 134-6GD00-0BA1, 11458
6ES7 134-6HB00-0CA1, 11458
6ES7 134-6HB00-0DA1, 11458
6ES7 134-6HD00-0BA1, 11458
6ES7 134-6JD00-0CA1, 11458
6ES7 134-6JF00-0CA1, 11459
6ES7 134-6PA00-0BD0, 11459
6ES7 135-6HB00-0CA1, 11459
6ES7 135-6HB00-0DA1, 11459
6ES7 135-6HD00-0BA1, 11459
6ES7 137-6AA00-0BA0, 11460
6ES7 137-6BD00-0BA0, 11462
6ES7 138-6AA00-0BA0, 11461
6ES7 138-6BA00-0BA0, 11462
6ES7 155-5AA00-0AB0, 11446
6ES7 155-5AA00-0AC0, 11446
6ES7 155-6AU00-0BN0, 11454
6ES7 155-6BU00-0CN0, 11455

6ES7 155-6DU00-0BN0, 11455
6ES7 193-6AE00-0AA0, 11461
6ES7 193-6AF00-0AA0, 11461
6ES7 193-6AP00-0AA0, 11460
6ES7 193-6AR00-0AA0, 11461
6ES7 193-6AS00-0AA0, 11461
6ES7 193-6PA00-0AA0, 11460
6ES7 211-1AD30-0XB0, 11430
6ES7 211-1AE31-0XB0, 11430
6ES7 211-1AE40-0XB0, 11430
6ES7 211-1BD30-0XB0, 11430
6ES7 211-1BE31-0XB0, 11430
6ES7 211-1BE40-0XB0, 11430
6ES7 211-1HD30-0XB0, 11430
6ES7 211-1HE31-0XB0, 11430
6ES7 211-1HE40-0XB0, 11430
6ES7 212-1AD30-0XB0, 11431
6ES7 212-1AE31-0XB0, 11431
6ES7 212-1AE40-0XB0, 11431
6ES7 212-1BD30-0XB0, 11431
6ES7 212-1BE31-0XB0, 11431
6ES7 212-1BE40-0XB0, 11431
6ES7 212-1HD30-0XB0, 11431
6ES7 212-1HE31-0XB0, 11431
6ES7 212-1HE40-0XB0, 11431
6ES7 214-1AE30-0XB0, 11432
6ES7 214-1AG31-0XB0, 11432
6ES7 214-1AG40-0XB0, 11432
6ES7 214-1BE30-0XB0, 11432
6ES7 214-1BG31-0XB0, 11432
6ES7 214-1BG40-0XB0, 11432
6ES7 214-1HE30-0XB0, 11432
6ES7 214-1HG31-0XB0, 11432
6ES7 214-1HG40-0XB0, 11432
6ES7 215-1AG31-0XB0, 11433
6ES7 215-1AG40-0XB0, 11433
6ES7 215-1BG31-0XB0, 11433
6ES7 215-1BG40-0XB0, 11433
6ES7 215-1HG31-0XB0, 11433
6ES7 215-1HG40-0XB0, 11433
6ES7 217-1AG40-0XB0, 11433
6ES7 221-1BF30-0XB0, 11436
6ES7 221-1BF32-0XB0, 11436
6ES7 221-1BH30-0XB0, 11436
6ES7 221-1BH32-0XB0, 11436
6ES7 221-3AD30-0XB0, 11434
6ES7 221-3BD30-0XB0, 11434
6ES7 222-1AD30-0XB0, 11434
6ES7 222-1BD30-0XB0, 11434
6ES7 222-1BF30-0XB0, 11437
6ES7 222-1BF32-0XB0, 11437
6ES7 222-1BH30-0XB0, 11437
6ES7 222-1BH32-0XB0, 11437
6ES7 222-1HF30-0XB0, 11437
6ES7 222-1HF32-0XB0, 11437
6ES7 222-1HH30-0XB0, 11437
6ES7 222-1HH32-0XB0, 11437
6ES7 222-1XF30-0XB0, 11437
6ES7 222-1XF32-0XB0, 11437
6ES7 223-0BD30-0XB0, 11434
6ES7 223-1BH30-0XB0, 11438
6ES7 223-1BH32-0XB0, 11438
6ES7 223-1BL30-0XB0, 11438
6ES7 223-1BL32-0XB0, 11438
6ES7 223-1PH30-0XB0, 11438
6ES7 223-1PH32-0XB0, 11438
6ES7 223-1PL30-0XB0, 11438
6ES7 223-1PL32-0XB0, 11438
6ES7 223-1QH30-0XB0, 11438
6ES7 223-1QH32-0XB0, 11438
6ES7 223-3AD30-0XB0, 11434
6ES7 223-3BD30-0XB0, 11434
6ES7 231-4HA30-0XB0, 11434
6ES7 231-4HD30-0XB0, 11440
6ES7 231-4HD32-0XB0, 11440
6ES7 231-4HF30-0XB0, 11440
6ES7 231-4HF32-0XB0, 11440
6ES7 231-5ND30-0XB0, 11440
6ES7 231-5ND32-0XB0, 11440
6ES7 231-5PA30-0XB0, 11434
6ES7 231-5PD30-0XB0, 11440
6ES7 231-5PD32-0XB0, 11440
6ES7 231-5PF30-0XB0, 11440
6ES7 231-5PF32-0XB0, 11440
6ES7 231-5QA30-0XB0, 11434
6ES7 231-5QD30-0XB0, 11440
6ES7 231-5QD32-0XB0, 11440
6ES7 231-5QF30-0XB0, 11440
6ES7 231-5QF32-0XB0, 11440
6ES7 232-4HA30-0XB0, 11434
6ES7 232-4HB30-0XB0, 11440
6ES7 232-4HB32-0XB0, 11440
6ES7 232-4HD30-0XB0, 11440
6ES7 232-4HD32-0XB0, 11440
6ES7 234-4HE30-0XB0, 11441
6ES7 234-4HE32-0XB0, 11441
6ES7 241-1AH30-0XB0, 11442
6ES7 241-1AH32-0XB0, 11442
6ES7 241-1CH30-0XB0, 11442
6ES7 241-1CH30-1XB0, 11434, 11435
6ES7 241-1CH31-0XB0, 11442
6ES7 241-1CH32-0XB0, 11442
6ES7 278-4BD32-0XB0, 11443
6ES7 505-0KA00-0AB0, 11452

6ES7 505-0RA00-0AB0, 11452
6ES7 507-0RA00-0AB0, 11453
6ES7 510-1DJ00-0AB0, 11454
6ES7 511-1AK00-0AB0, 11444
6ES7 511-1FK00-0AB0, 11445
6ES7 512-1DK00-0AB0, 11454
6ES7 513-1AL00-0AB0, 11444
6ES7 513-1FL00-0AB0, 11445
6ES7 515-2AM00-0AB0, 11444
6ES7 515-2FM00-0AB0, 11445
6ES7 516-3AN00-0AB0, 11444
6ES7 516-3FN00-0AB0, 11445
6ES7 518-3AN00-0AB0, 11444
6ES7 518-4AP00-0AB0, 11444
6ES7 518-4FP00-0AB0, 11445
6ES7 521-1BH00-0AB0, 11447
6ES7 521-1BH10-0AA0, 11446
6ES7 521-1BL10-0AA0, 11447
6ES7 522-1BF00-0AB0, 11447
6ES7 522-1BH00-0AB0, 11448
6ES7 522-1BH10-0AA0, 11448
6ES7 522-1BL00-0AB0, 11449
6ES7 522-1BL10-0AA0, 11448
6ES7 522-5FF00-0AB0, 11448
6ES7 522-5HF00-0AB0, 11448
6ES7 523-1BL00-0AA0, 11449
6ES7 531-7KF00-0AB0, 11449
6ES7 531-7NF10-0AB0, 11449
6ES7 531-7QD00-0AB0, 11449
6ES7 532-5HD00-0AB0, 11450
6ES7 532-5HF00-0AB0, 11450
6ES7 532-5NB00-0AB0, 11450
6ES7 534-7QE00-0AB0, 11450
6ES7 540-1AB00-0AA0, 11452
6ES7 540-1AD00-0AA0, 11451
6ES7 541-1AB00-0AB0, 11452
6ES7 541-1AD00-0AB0, 11452
6ES7 550-1AA00-0AB0, 11453
6GK7 242-5DX30-0XE0, 11441
6GK7 242-5DX31-0XE0, 11441
6GK7 242-7KX30-0XE0, 11441
6GK7 243-1JX30-0XE0, 11441
6GK7 243-5DX30-0XE0, 11441
6GK7 243-5DX31-0XE0, 11441
6GK7 542-1AX00-0XE0, 11451
6GK7 542-5DX00-0XE0, 11451
6GK7 543-1AX00-0XE0, 11451
6GT2 002-0LA00, 11443

7

7MH4 960-2AA01, 11443
7MH4 960-4AA01, 11443

A

A, 4062, 6158
A(, 4072, 6168
A_DEAD_B
  Instrucción, 7234
  Parámetros de entrada, 7237
  Parámetros de salida, 7238
AB, 2571
Abreviatura idiomática, 968
Abreviaturas para los idiomas, 1006
Abrir
  Librería global, 9145
Abrir archivo de registro
  Eventos de filtrado de paquetes, 837, 839
  Eventos de seguridad, 836, 838
  Eventos de sistema, 835, 838
Abrir las preguntas frecuentes sobre un módulo, 506
Abrir listines telefónicos, 11372
Abrir los manuales de un módulo, 506
Abrir Siemens Industry Online Support, 506
Abrir un proyecto, 343
  Modo de compatibilidad V12 SP1, 341
Abrir una tabla de forzado permanente, 3148
Abrir una tabla de observación, 3120
ABS, 3411, 3723, 4144, 4335, 4533, 5680, 5903, 6239, 6420, 6557
ABS_CTRL, 7420
  Bloques de datos empleados, 7421
  Comportamiento en caso de error, 7422, 7426
  Descripción, 7420, 7424
  Estado de la petición, 7440
  Funcionamiento, 7420
  Llamada, 7420
  Peticiones, 7438
  Selectores de función, 7438
  Señales de control, 7421
  Valores de retorno, 7421
ABS_CTRL_451, 7424
  Bloques de datos empleados, 7425
  Funcionamiento, 7424
  Llamada, 7424
  Parámetros, 7425
  Peticiones, 7441
  Selectores de función, 7441
  Señales de control, 7425
  Valores de retorno, 7425
ABS_DIAG, 7422
  Arranque, 7423
  Bloques de datos empleados, 7422

Comportamiento en caso de error, 7423
Datos de diagnóstico, 7423
Descripción, 7422
Funcionamiento, 7422
Llamada, 7422
Peticiones, 7423
Valores de retorno, 7423
ABS_DIAG_451, 7426
Arranque, 7427
Bloques de datos empleados, 7427
Comportamiento en caso de error, 7427
Datos de diagnóstico, 7428
Descripción, 7426
Funcionamiento, 7426
Llamada, 7426
Parámetros, 7427
Peticiones, 7427
Valores de retorno, 7427
ABS_INIT, 7419
Bloques de datos empleados, 7419
Descripción, 7419
Llamada, 7419
Parámetros, 7420
Valores de retorno, 7420
Acceder a operandos, 2293, 2297, 2301
Acceder desde una instalación remota a una programadora/un PC, 11415
Acceso
~ directo a la periferia, 1810
Acceso a bloques
Bloque de datos, 2951
Acceso a la web, 941
Acceso a operandos, 2294, 2295, 2298, 2302
Acceso de red
Abrir propiedades, 11312
Acceso directo a la periferia, 1636, 1810, 1916
Acceso directo a una carpeta
Cambiar nombre, 8358
Editar, 8357
Eliminar, 8358
Acceso libre Layer 2, 683
Acceso online, 11308
Acceso optimizado, 2258, 2260
Acceso optimizado al bloque, 250
Acceso Slice, 2301
Acción
Acciones controladas por evento, 2839
Acciones estándar, 2837
Borrar, 2909
Contadores, 2844
Copiar, 2908
Cortar, 2908
Desplazar, 2909
Insertar, 2901
Introducción, 2833
Llamada de bloque, 2842
Modificar, 2907
Pegar, 2908
Principios básicos, 2836
Temporizadores, 2843
Accionamiento
Datos máquina, 11038
Accionamiento paso a paso, 11199
Accionamiento paso a paso con un sensor tipo Stepper, modo PULSE REFEED, 11201
Acciones
Deshacer, 416
Principios básicos para deshacer acciones, 413
Principios básicos para rehacer acciones, 413
Rehacer, 417
Acelerar el arranque (PROFINET), 1564
ACK, 9324
AcknowledgeAlarm, 8677
ACL, 1340
Acoplador DP/DP, 1389, 1464
Acoplador PN/PN
Acoplar subredes Ethernet, 1607
Agrupar, 1607
Acoplamiento de racks, 910, 912
Acoplamiento multipunto
Allen-Bradley DF1, 9022, 9023
Acoplamiento punto a punto, 1012
Activar señales de salida con SGN_SET, 5516
Allen-Bradley DF1, 9021
Borrar búfer de recepción con RCV_RST, 5514
Configurar parámetros de comunicación con PORT_CFG, 5499
Configurar parámetros de recepción con RCV_CFG, 5504
Configurar parámetros de transferencia serie con SEND_CFG, 5502
Consultar estado lógico con SGN_GET, 5515
Habilitar recepción con RCV_PTP, 5513
Lanzar la transferencia de datos con SEND_PTP, 5510
Acoplamiento punto a punto ((3964(R), CPU 31xC), 1798
Acoplamiento punto a punto ((ASCII), CPU 31xC), 1791
Acoplamiento punto a punto ((RK 512), CPU 31xC), 1803
Acoplamiento punto a punto (CPs, ET 200S 1SI), 1068

Acoplamientos
Allen-Bradley DF1, 9019, 9020
ACOS, 3430, 3743, 4153, 4351, 4549, 5699, 5922, 6247, 6434, 6573
ACT_TINT, 1745, 4895
Activación y desactivación de dispositivos IO, 1560
activar, 8715
Idioma del proyecto, 9203
Activar
Área de periferia, 5773, 6100, 6494, 6625
Área por bits, 5772, 5993, 6098, 6493, 6624
Área por bytes, 5775, 5997, 6102, 6495, 6628
Datos máquina, 9749, 10901, 10995
Estilo, 8353
Mapa de bits, 3290, 3599, 5772, 5993, 6098, 6493, 6624
Operando, 3289, 3292, 3598, 3602, 4083, 5607, 5609, 5828, 5829, 6179
RLO, 4085, 6181
Valor inicial del contador, 3678, 5883
Activar autonegotiation, 1511, 8803, 8854, 8907
Activar contador, 9593
con la entrada digital 1I2, 9594
Con la marca cero, 9596
Mediante una señal externa, 9594
Activar cortafuegos
SCALANCE S, 853, 855
Activar la comunicación tunelada
CP x43-1 Adv., 878
SCALANCE S, 854, 855
Activar operando con flanco de señal ascendente, 3298, 3607
Activar operando con flanco de señal descendente, 3299, 3608
Activar variable con flanco de señal ascendente, 3303, 3612, 3904, 4283
Activar variable con flanco de señal descendente, 3305, 3613, 3905, 4284
ActivarBit, 8691
ActivarBitDeVariable, 8692
ActivarBitMientrasTeclaPulsada, 8693
ActivarImagen, 8649
ActivarImagenAnterior, 8652
ActivarImagenConNúmero, 8650
ActivarPantallaDeLimpieza, 8651
ActivatePreviousScreen, 8652
ActivateScreen, 8649
ActivateScreenByNumber, 8650
Activo
Punteros de área, 8782
Actuador, 10641
Actualización, 1681
Componentes PROFINET CBA, 8263
Interconexión DP, 8263
variable, 8485
Variable, 8479
Actualización automática, 925
Actualización de la versión del dispositivo, 9272
Actualización del firmware, 1931, 2189
Actualización del firmware , 9272
Actualización del sistema operativo, 9272
Actualizar
Datos IPE, 9161, 9166
Librería global, 9249
Proxy de dispositivos, 9161, 9166
Proyecto, 9248
Sistema operativo, 9273
Versión de proyecto, 9248
Actualizar interconexiones DP con PN_DP, 8267
Actualizar librería, 482, 9147
Actualizar tipos a la versión más reciente, 482, 9147
Actualizar una librería global, 445
ActualizarVariable, 8653
Acumulador, 2753, 4274, 4275, 4276, 6363, 6364, 6366, 6367, 6368
Acumulador de tiempo, 3314, 3622, 3631, 3916, 4295, 4520, 5808, 6028, 6145, 6525, 6660
Acumuladores, 1808
Acusar, 8505, 8723
Alarma, 9326
Aviso, 9326
AcusarAviso, 8677
Acuse, 8507
Configurar, 8530, 8531, 8532
AD, 2571, 4249, 6339
Adaptación del tiempo de ciclo de emisión, 1521
Adaptador de red, 876
Adaptar proyecto
a otro panel de operador, 9277
ADD, 3400, 3711, 5672, 5894
ADM_ZONE
Instrucción, 7607
Admin, 9319
Administración de librerías
Abrir, 437, 9141
Filtrar, 9142
Filtrar por tipos no habilitados, 438
Sinopsis, 435, 9140
Visualizar instancias de versiones de tipo, 439
Visualizar referencias cruzadas de una instancia, 439
Visualizar relaciones entre objetos de librerías, 440
Administración de usuarios, 1753, 2010, 8599

Servidor web, 944
Configuración de runtime, 8316, 8604, 8621
estructurar, 8627
Migración, 162
Objeto con protección de acceso, 8624
Administrador de certificados, 760
Administrador del equilibrio de carga de red (NLB), 2060
Administrar
Gráfico, 8356
Grupo de usuarios, 8612
Usuarios, 8611
Usuarios en runtime, 8619
Administrator, 766
admisibles
Tipo de datos, 9092
Tipos de datos, 8835, 8887, 8944
Admisibles
Tipo de datos, 9004, 9050, 9060, 9078, 9108
Tipos de datos, 8985
Advanced Encryption Standard (AES), 818
AES, 818
AES-128, 812
AG_CNTEX, 8069
AG_CNTRL, 8081
AG_LOCK, 8064
AG_LRECV, 8202
AG_LSEND, 8195
AG_RECV / AG_LRECV / AG_SRECV, 8059
AG_SEND / AG_LSEND / AG_SSEND, 8055
AG_UNLOCK, 8066
Agente SNMPv1, 2022
Aggressive Mode, 817
Aging Time, 1344
AG-RECV, 8202
Agregar
Gráfico externo, 8358
Agregar fila de separación, 10487
Agregar fila Empty, 10487
Agrupar
Objeto, 8362
AG-SEND, 8195
Ajustar
Tiempo de retardo, 8518
Zona muerta, 8518
Ajustar el idioma, 9299
Ajustar la carga de comunicación, 550
Ajustar los nemónicos, 2630
Ajustar valor límite, 3417, 3729, 3938, 4340, 4538, 5685, 5908, 6064, 6423, 6562
Ajustar valor real, 9757, 9835, 10926, 11021
Datos en el DB de canal, 9758, 9836, 10926, 11022
Desarrollo, 9758, 9836, 10926, 11021
Efecto, 9759, 9836
Efectos, 10927, 11022
Requisitos, 9758, 9835, 10926, 11021
Ajustar valor real al vuelo, 9757, 9835, 11021
Datos en el DB de canal, 9758, 9836, 11022
Desarrollo, 9758, 9836, 11021
Requisitos, 9758, 9835, 11021
AjustarModoPLC, 8695, 8696
Ajuste
dinámicos, 9741
Ajuste de la hora, 1264
Ajuste del ancho de tabulación , 2798
Ajuste del canal
FM 350-2, 9556
Ajuste del tipo, tamaño y color de la fuente , 2798
Ajuste dinámico, 9741, 9810, 9888
Ajuste por defecto
Alarma de diagnóstico, 10332
Ajustes de avisos, 1931
Ajustes de escritura
Migrar, 154
Ajustes de fábrica
Restablecer , 9273
Ajustes de seguridad, 1279
Ajustes de test de GRAPH
Modificar, 3104
Resumen, 3103
Ajustes IKE, 816
Ajustes IPSec, 816
Al abrir un cuadro de diálogo, 8717
Al cerrar un cuadro de diálogo, 8718
Al rebasar por defecto, 8717
Al rebasar por exceso, 8717
Alarma
Acusar, 9326
Acuse desde el controlador, 8833, 8884, 8942, 8984, 8995, 9039, 9068, 9098, 9115
Acuse en el panel de operador, 8833, 8885, 8943, 8984, 8996, 9039, 9068, 9099, 9115
Configurar el acuse, 8832, 8884, 8942, 8984, 8995, 9039, 9068, 9098, 9114
enviar en el I-Slave, 1418
Alarma cíclica
Configuración, 1747
OB 30 a OB 38 (S7-300, S7-400), 1828
Alarma de actualización, 1681
Alarma de actualización (S7-1500), 1969
Alarma de diagnóstico, 9398, 9547, 9574, 9607

¿Qué es una alarma de diagnóstico?, 10331, 10365
Ajuste por defecto, 10280, 10332, 10365
Definición, 10280, 10331
Disparar, 10279, 10280, 10331, 10365
en casos de error, 10364
habilitación, 9430
habilitar, 10280, 10331, 10365
OB 82, 10280, 10332, 10365
Alarma de estado, 1680
Alarma de estado (S7-1500), 1968
Alarma de extracción/inserción (ET 200M), 2139
Alarma de multiprocesamiento
Disparar con MP_ALM, 6911
Alarma de proceso, 9398, 9532, 9569, 9604
Configuración, 1743
Disparar en el maestro DP con DP_PRAL, 6794
ejemplo, 1928
Funcionamiento, 1642
habilitación, 9436
habilitar, 9532, 9604
Habilitar, 9570
I-Slave, 1443
OB 40 a OB 47 (S7-300, S7-400), 1830
perdida, 9434
Alarma de proceso (S7-1500), 1973
Alarma de retardo
Anular con CAN_DINT, 4901, 6893
Configuración, 1744
Consultar con QRY_DINT, 4901, 6894
Iniciar con SRT_DINT, 4899, 6892
Instrucciones, 4898, 6891
OB 20 a OB 23 (S7-300, S7-400), 1827
Alarma de retardo (S7-1500), 1971
Alarma de sincronismo, 1748
Alarma de tiempo para control, 3568, 3571, 3883, 3886, 4034, 4038, 4488, 4491, 4651, 4654, 5792, 5795, 6013, 6016, 6123, 6127, 6510, 6513, 6644, 6648
Alarma horaria, 1745
Activar con ACT_TINT, 4895, 6889
Ajustar con SET_TINT, 4890, 6887
Ajustar con SET_TINTL, 4892
Ajustar y activar, 1677
Anular, 1677
Anular con CAN_TINT, 4894, 6888
Consultar con QRY_TINT, 4896, 6889
Consultar estado, 1677
Función, 1677
Instrucciones, 6884
OB 10 a OB 17 (S7-300, S7-400), 1823
Propiedades de SET_TINT, CAN_TINT, ACT_TINT, QRY_TINT, 6885
Reglas, 1677
Alarma horaria (S7-1500), 1964
Alarmas
Activar alarma horaria con ACT_TINT, 4895, 6889
Ajustar alarma horaria con SET_TINT, 4890, 6887
Ajustar alarma horaria con SET_TINTL, 4892
Alarma cíclica (S7-300, S7-400), 1828
Alarma de cambio de módulo (S7-300, S7-400), 1854
Alarma de diagnóstico (S7-300, S7-400), 1851
Alarma de proceso (S7-300, S7-400), 1830
Alarma horaria (S7-300, S7-400), 1823
Anular alarma de retardo con CAN_DINT, 4901, 6893
Anular alarma horaria con CAN_TINT, 4894, 6888
Con direcciones comprimidas, 1414, 1484
Consultar alarma de retardo con QRY_DINT, 4901, 6894
Consultar alarma horaria con QRY_TINT, 4896, 6889
De un esclavo I, enviar con SALRM, 6786
Disparar alarma de proceso con DP_PRAL, 6794
Iniciar alarma de retardo con SRT_DINT, 4899, 6892
Leer estado de alarma con lista de estado del sistema, 6987
OBs para DPV1, 1435
Recibir desde módulo de periferia con RALRM, 4753, 6745
Retardo (S7-300, S7-400), 1827
Alarmas de diagnóstico, 9818, 9898, 10973, 11075
Evaluación, 10976, 11078
Alarmas de proceso
habilitar, 9812, 9890
Alarmas tecnológicas, 10782
Lista de alarmas tecnológicas, 10838
Alcance de la impresión, 392
Algoritmo Lead y Lag, 3580, 3896, 4053, 4500, 4668, 5804, 6025, 6142, 6522, 6657
Algoritmo SHA, 1279
Alimentación de parámetros
Activar a través de registros, 2757
Alisamiento, 2115
Allen-Bradley, 8997, 9019, 9020
Aviso analógico, 9038
Basic Panel, 8788
DF1, 8998

Driver de comunicación Allen-Bradley DF1, 9019, 9020
Drivers de comunicación, 8998
EtherNet/IP, 8998
Mitsubishi, 9067
Tipo de datos, 9038
Allen-Bradley
Puntero de área, 9036
Allen-Bradley DF1
Conexión, 9014, 9018
Configurar una conexión, 9014
Migración de los tipos de datos, 170
Módulo KF2, 9019, 9020
Módulo KF3, 9019, 9020
Parámetros de conexión, 9016
Tipo de CPU, 9029
Tipo de datos admisible, 9027
Allen-Bradley Ethernet IP
Direccionamiento, 9006
Migración de los tipos de datos, 170
Multiplexar direcciones, 9010
Tipo de direccionamiento, 9009
Allen-Bradley EtherNet/IP
Conexión, 8999, 9003
Configurar una conexión, 8999
Parámetros de conexión, 9001
Tipo de datos, 9004
AN, 4063, 6159
AN(, 4073, 6169
ANALOG, 7629
Ancho de banda (comunicación IRT), 1530
Ancho de impulso mínimo, 9512, 9593
AND, 2792, 2793, 3539, 3589, 3852, 5740, 5821, 5963
Animación
Configurar, 8394
Flecha verde en vista general, 8394
Grupo de objetos, 8401
Movimiento diagonal, 8397
Movimiento directo, 8398
Movimiento horizontal, 8396
Movimiento vertical, 8397
Selección múltiple, 8401
Vista general, 8394
Anulación de un desplazamiento
Easy Motion Control, 11221, 11255
Anulación y aplicación de sangrías
Líneas, 2799
ANY, 2362
Añadir
Objetos al grupo, 8364
Aparecer, 8505
Aparecido, 8721
Apariencia
dinamizar un objeto, 8395
Aplicación web, 946, 983
Aplicaciones posibles del módulo de comunicación, 1121
Applet, 769
Aproximación al destino, 10905, 11000
Aproximación en bucle, 10929, 11030
Datos en el DB de canal, 10930, 11031
Desarrollo , 10930, 11030
Requisitos, 10929, 11030
Archivación de proyectos, 357, 358
Archivo CSV, 9169, 9171, 9180
Archivo de configuración, 1203
Archivo de informe
mensajes de error del módem , 11427
Archivo de proyecto
Inicialización, 9163
Inicializar, 9163
Archivo fuente externo
Enlazar tipos de datos con un editor, 2980
Exportar bloque, 2978
Importar bloque de organización, 49
Integrar, 2979
Principios básicos, 2976
Reglas de programación, 2977
Archivo gráfico
Agregar al juego de gráficos, 8358
agregar al juego de gráficos , 9215
Archivo gráfico externo
Administrar, 8329
Agregar al juego de gráficos, 8358
insertar en Gráficos, 9215
Archivo IPE, 9157
Inicialización, 9157
Inicializar, 9158
Archivo xlsx, 9174, 9181, 9186, 9187
Archivos de idioma
para visor de diagnóstico, 2021
Archivos GSD
configurar dispositivos (PROFIBUS), 1433, 1486
Instalar, 1432, 1485
revisiones GSD (PROFIBUS), 1431, 1484
Archivos GSD (PROFINET), 1515
Cambiar versión, 1516
Instalar, 1516
Arcocoseno, 3430, 3743, 4153, 4351, 4549, 5699, 5922, 6247, 6434, 6573
Arcoseno, 3429, 3741, 4151, 4350, 4548, 5697, 5920, 6245, 6433, 6572
Arcotangente, 4156, 4352, 6250, 6435

ARP, 874
Diagnóstico, 874
Tabla, 874
Arquitectura de consultas, 5396, 7786
Arquitectura de consultas del esclavo, 5397, 7787
Arquitectura de consultas del maestro, 5396, 7787
Arquitectura de red
PPI, 8764
Arrancar temporizador como impulso, 3625
Arrancar temporizador como retardo a la conexión, 3627
Arrancar temporizador como retardo a la desconexión, 3629
Arranque, 1913
ABS_DIAG, 7423
ABS_DIAG_451, 7427
Bloques de organización, 1675
CAM_DIAG, 7456
CAM_DIAG_452, 7459
CAM_MSRM_452, 7461
CPU, 7422, 7426, 7455, 7458
FM 351, 7422
FM 352, 7455, 7458
FM 451, 7426
FM 452, 7455, 7458
Vista general, 1710
ARRANQUE, 1703
Actividades de arranque, 1625, 1708, 1904
Cancelación, 1709
Comportamiento remanente, 1706
Función, 1623, 1902
Instrucciones para generar avisos de PLC, 6933
Rearranque completo (en caliente) automático, 1704
Rearranque completo (en caliente) automático sin respaldo, 1704
Rearranque completo (en caliente) manual, 1704
Arranque de dispositivos IO, 1560
Arranque en frío, 1704
Arranque en frío (S7-300, S7-400), 1875
Arranque priorizado, 1560
Arranque priorizado (PROFINET), 1564
Array, 260
ARRAY, 2307, 2317
Declaración en bloques de datos globales, 2945
Declaración en la interfaz del bloque, 2651
Declaración en tipos de datos PLC, 2970
Direccionar, 2298
Ejemplo, 2357
Formato, 2354, 2355
Asegurar la disposición de tabla, 310
Asegurar la interfaz de usuario, 310
Asegurar la disposición en editores, 310
Asiáticas, 31
Asignación, 2794, 3286, 3595, 4080, 5605, 5826, 6176
negada, 3287
negar, 3596
Registro de diagnóstico DS0, 10333
Registro de diagnóstico DS1 del FM 355-2, 10334
Asignación de bloques de datos globales, 50
Asignación de direcciones, 915
Datos de usuario con DPV1, 1441
Asignación de direcciones (información de calidad), 1919
Asignación de pines
6XV1440-2P para el protocolo PG de Mitsubishi, 9059
Allen-Bradley, 9023
Cable Allen-Bradley 1747-CP3, 9025
Cable Allen-Bradley 1761-CBL-PM02, 9026
Cable Allen-Bradley 1784-CP10, 9024
Cable multipunto 1:MP/TP/PC, 9106
Cable multipunto 2:RS422, MP/TP/PC, 9107
Cable punto a punto 1,
Cable punto a punto 2, 9091
Cable punto a punto PP1 para Omron, 9107
Cable punto a punto PP2 para Omron, 9108
Asignación de terminales
Conector frontal del FM 350-1, 9496
Conector frontal del FM 350-2, 9552
conector frontal del FM 351, 10897
conector frontal del FM 352, 9755
Conector frontal del FM 450-1, 9581
conector frontal del FM 451, 10990
conector frontal del FM 452, 9831
conector frontal del SM 338, 10856
interfaz del encóder del FM 351, 10896
interfaz del encóder del FM 352, 9754
interfaz del encóder del FM 451, 10989
Asignación de TSAP
Ejemplos, 720
Asignación de valor, 2794
Asignación global
de una tecla de función, 8404, 8406
Asignación local
de una tecla de función, 8404, 8408
Asignar
un gráfico a una tecla de función, 8412
Un objeto a un nivel, 8418
una función a una tecla de función, 8409
una tecla de función, 8404, 8406, 8408

Asignar derechos de acceso específicos de OPC Items, 2045
Asignar dirección IP
desde el contexto del proyecto, 2196, 2247
mediante "Dispositivos accesibles", 2195, 2246
Principios básicos, 2194, 2245
Asignar ENO automáticamente, 2795
Asignar la vía de conexión, 616
Asignar símbolo, 928, 974
Asignar variable, 928, 974
ASIN, 3429, 3741, 4151, 4350, 4548, 5697, 5920, 6245, 6433, 6572
Asíncrona
Transferir datos, 8824, 8877, 8934, 8980, 9126
AS-Interface, 1488
Asistente
Asistente de panel de operador, 9133
Asistente de panel de operador, 9133
Asistente de paneles de operador, 9133
AT , 2302
ATAN, 3432, 3744, 4156, 4352, 4551, 5700, 5923, 6250, 6435, 6574
ATH, 4724, 6719
ATTACH, 1642, 4884
ATTR_DB, 5001
Autenticación, 1280, 1376
Autonegotiation, 1512
Desactivado, 8804, 8855, 8908
Autorización
administrar, 8612
asignar, 8608, 8631
Borrar, 8614
Cambiar nombre, 8613
configurar, 8623
Crear, 8607, 8629
AvanzarPágina, 8689
Aviso
Acusar, 9326
Componentes, 3172, 8510
Configurar, 8516, 8522
Editar, 9327
En runtime, 8533, 9320
exportar, 9173
Función de sistema, 8538
importar, 9174
Mostrar, 8438
Parámetro formal, 3176
Tipo de datos de avisos, 3176
Transferencia con sello de tiempo con TIMESTMP, 6710
Ventana de inspección, 64
Aviso analógico, 8503
Allen-Bradley, 9038
Configurar, 8516, 8523
Mitsubishi, 9067
Omron, 9114
Rebase de valor límite, 8476
Aviso con acuse simple, 8509
Aviso de bit, 8503, 8504
Allen-Bradley, 9038
Configurar, 8514, 8522
Mitsubishi, 9067
Omron, 9114
Aviso de diagnóstico de usuario
Borrar, 3182
Crear, 3181
Edición, 3182
Aviso de programa
Borrar, 3180, 3188
Crear, 3178, 3186
Crear con Program_Alarm, 4916
Edición, 3179, 3186, 3187
Aviso de sistema, 8503
Parámetros, 8539
Significado, 8540, 8541, 8544, 8545, 8547, 8548, 8550, 8551, 8552, 8553, 8555, 8558, 8559
Aviso sin acuse, 8509
Avisos
Conexión no integrada, 9097
Configurar, 8883, 8941, 8983, 8994
Limitación, 9067
Modicon Modbus, 9097
Registrarse para ello en la CPU, 2242
Tipos de datos, 8883, 8941, 8983, 8994
Visualizar el valor de una variable, 8519
Avisos analógicos
Configurar, 8512
Avisos de bit
Configurar, 8512
Avisos de controlador
Configurar, 8512
Avisos de CPU - Servidor OPC, 2062
Avisos de PLC con instrucciones
Acusables, con ALARM_DQ, 6943
Bloquear con DIS_MSG, 6954
Comportamiento en arranque de las instrucciones, 6933
Comportamiento en fallo de las instrucciones, 6933
Con indicación de acuse con ALARM, 6930
Con valores asociados para ocho señales con ALARM_8P, 6925
Conocer el estado de acuse con ALARM_SC, 6945

Generar , 6917
Habilitar con EN_MSG, 6952
Leer recursos del sistema asignados dinámicamente, con READ_SI, 6956
Liberar recursos del sistema asignados dinámicamente, con DEL_SI, 6959
Recursos del sistema con ALARM_SQ y ALARM_S, 6936, 6939
Siempre acusados con ALARM_D, 6941
Sin indicador de acuse con NOTIFY, 6928
Sin indicador de acuse con NOTIFY_8P, 6920
Sin valores asociados para ocho señales con ALARM_8, 6923
Vista general, 6934
Avisos de sistema
Configurar, 8512
Avisos RET_REALU, 10284, 10285, 10336, 10337, 10371, 10372
AW, 2571, 4243, 6333
AWL, 2753, 3226, 3262
Alimentación de parámetros, 2757
Comparación detallada, 2993
Migración, 139
AWP_In_Variable, 990
AWP_Out_Variable, 951, 988
AX, 2571
Ayuda
Abrir, 330
Borrar temas de Favoritos, 333
Buscar palabras clave, 331
Búsqueda de texto completo, 331
examinar, 331
Expandir automáticamente los tooltips en cascada, 334
Guardar temas en Favoritos, 332
Identificación de temas de ayuda, 328
Imprimir temas de Ayuda, 333
Utilizar índice, 331
Utilizar temas de Favoritos, 332

Á

Ámbito de aplicación, 11196
Árbol del proyecto
Agregar dispositivo, 520
Contraer automáticamente, 294
Función, 288
Mostrar, 2628
Ocultar, 2628
Área
Activar en la periferia, 5773, 6100, 6494, 6625
Activar por bits, 5772, 5993, 6098, 6493, 6624
Activar por bytes, 5775, 5997, 6102, 6495, 6628
Área, 3797, 5927
Copiar, 3453, 3484, 3766, 3951, 3977, 4361, 4406, 4558, 4583, 5704, 6066, 6436, 6578
Copiar sin interrupciones, 3459, 3487, 3773, 3800, 3958, 3980, 4366, 4409, 4564, 4586, 5708, 5931, 6070, 6440, 6582
Desactivar en periferia, 5779, 6000, 6105, 6498, 6631
Desactivar por bits, 5777, 5999, 6104, 6497, 6630
Desactivar por bytes, 5781, 6002, 6107, 6500, 6633
Rellenar, 3461, 3490, 3775, 3803, 3960, 3983, 4368, 4411, 4566, 4589, 5710, 5933, 6073, 6442, 6585
Rellenar sin interrupciones, 3463, 3777, 3962, 4370, 4568
Área de datos de la periferia
Acceso, 1731
Datos de diagnóstico, 1732
Datos de parámetros, 1732
Datos de usuario, 1732
Registros del módulo, 1733
Área de datos del proceso
Dirección inicial de un módulo, 1635
Direccionamiento de la periferia, 1735
Direccionamiento de módulos, 1635, 1734
Área de direcciones
modificar, 916, 973
Área de memoria
Áreas de memoria remanentes, 1632
de las entradas y salidas, 1809
De un módulo, leer con lista de estado del sistema, 6978
Distribución, 1717
Memoria de carga, 1630
Memoria de trabajo, 1631
PI y PQ, 1810
Área de módulos no enchufados, 516
Área de operandos, 1631, 1723
Mostrar, 2234
Área de solicitud de curvas, 8830, 8993
Área de trabajo
Cambiar entre los elementos, 299
Desacoplar elementos, 296
Dividir, 295
Función, 291
Guardar la configuración de ventana de los editores y tablas, 313
Incrustar elementos desacoplados, 296
Maximizar, 293

Maximizar elementos, 298
Minimizar, 293
Minimizar elementos, 298
Uso de elementos agrupados, 297
Área de transferencia, 1440
Área de transferencia de curvas, 8830, 8993
Área de transferencia de curvas 1
Área de transferencia de curvas 2, 8830, 8993
Área del bloque de datos
enviar con SEND_PTP, 8237
enviar con SEND_RK, 8244
recoger con FETCH_RK, 8248
Áreas de memoria
Base de cálculo, 1722
Memoria de carga, 1718
Memoria de sistema, 1718
Memoria de trabajo, 1718
Memoria del sistema, leer con lista de estado del sistema, 6979

B

Bajo demanda, 8479
Barra, 8424, 9299
Cambio de color, 8424
Mostrar líneas de límite, 8424
Barra de herramientas
Orden, 8328, 8339
Barrido de matriz, 3578, 3893, 4047, 4497, 4663, 5802, 6022, 6137, 6519, 6654
Basic Panel
Drivers de comunicación, 8788
ETHERNET, 8790
IF1B, 8790
Inicio de runtime, 9261, 9284
Mitsubishi, 8788
Modicon Modbus, 8788
Objeto de indicación y manejo, 8423
Omron, 8788
OPC, 8788
Puerto, 8790
Punteros de área, 8791
BCC, 1034
BCD, 4179, 6272
BCDCPL, 3586, 3901, 4059, 4505, 4673, 5817, 6041, 6155, 6533, 6670
BE, 4234, 6325
BEC, 4235, 6326
BEU, 4237, 6328
Bit (0, 1)
Lista de gráficos, 8387
Lista de textos, 8379
Bit de datos (DBX), 1631
Bit de datos locales (L), 1631
Bit de estado, 2545, 2551, 2552, 2553, 2555, 2559, 2563
Consultar, 2556, 2560, 5982
Bit de marca (M), 1631
Bits
Contaje, 3587
Contar, 3903, 4060, 4506, 4674, 5818, 6043, 6156, 6534, 6671
Bits de estado
Consultar, 2557, 2562
BITSUM, 3587, 3903, 4060, 4506, 4674, 5818, 6043, 6156, 6534, 6671
BLD, 4280, 6380
BLKMOV, 51, 3484, 3797, 3977, 4406, 4583, 5704, 5927, 6066, 6436, 6578
BLOCK, 2366
BLOCK_DB_TO_WORD, 2539
Bloque
Abandonar, 4448, 6469
Abrir, 2614
Abrir bloques con protección de know-how, 3050
Acceso optimizado, 2258, 2260
Borrar offline, 2616
Borrar online, 2617
Buscar y abrir, 2614
Cambiar contraseña de bloques con protección de know-how, 3052
Cambiar nombre, 2616
Cargar de dispositivo, 3018, 3037
Cargar de un dispositivo, 3027
Cargar de una Memory Card, 3029, 3044
cargar desde un dispositivo, 3042
Cargar en dispositivo, 3018, 3023, 3037, 3039
Cargar en el dispositivo en el modo operativo RUN, 3038
Cargar en un dispositivo en estado operativo RUN, 3021
Cargar en una Memory Card, 3028, 3043
Cerrar, 2615
Comparación, 2982
Comparar, 2986
Comparar bloques de datos, 2985
Comparar bloques lógicos, 2984
Compilar, 3012
Compilar en el árbol del proyecto, 3015
Compilar en el editor de programación, 3016
Comprobación de coherencia, 3012, 3069, 3074, 3075
Comprobación de coherencia en la estructura de llamada, 3069

Configurar protección contra copia, 3048
Copiar, 2601, 2604
Editar propiedades, 2613
Exportar a un archivo fuente externo, 2978
Guardar, 2615
Imprimir bloques con protección de know-how, 3051
Insertar, 2601, 2605
Introducir comentario, 2606
Introducir título, 2606
Llamada condicional, 4240, 6330
Llamar, 5763, 5984
Mostrar las propiedades, 2613
Propiedades, 2608
Protección de know-how, 3045
Quitar protección contra copia, 3048
Sello de tiempo, 2611
Tipos, 2251
Utilizar una librería, 2599
bloque de datos, 1910
Bloque de datos, 1915
Abrir, 4227, 4228, 5729, 5730, 5952, 5953, 6319, 6320
Acceso optimizado, 2258, 2260
Actualizar, 2942
Adaptar los valores de datos durante la puesta en marcha, 2950, 2962, 2963, 2964, 2965
Basados en un tipo de datos PLC , 2947
Bloque de datos ARRAY, 2254, 2258, 2295, 2298, 2598, 2937, 2940, 2962, 3030
Bloque de datos de CPU, 2256
Bloque de datos de instancia, 2256, 2937
Bloque de datos globales, 2254
Bloque de datos globales , 2937
Borrar con DEL_DB, 7036
Borrar con DELETE_DB, 5002
Cargar cambios de bloque sin reinicializar, 2943
Comportamiento remanente, 2951, 2952
CONF_DB, 7769
Crear, 2598, 2940
Crear con CREAT_DB, 7038
DB de reglas WS_RULES, 6714
Declarar ARRAY, 2945
Declarar STRUCT, 2946
Escribir en la memoria de carga con WRIT_DBL, 4999, 7034
Generar con CREA_DB, 7039
Generar con CREA_DBL, 7041
Generar con CREATE_DB, 4994
Importar y exportar variables, 2960
Leer atributos con ATTR_DB, 5001
Leer de la memoria de carga con READ_DBL, 4997, 7032
Observar valores de datos online, 2961
Programar, 2937, 2944
Propiedades de las variables, 2953, 2955, 2957
Tabla de declaración, 2939
Testear con TEST_DB, 7044
Utilizar valores de ajuste, 2950, 2962, 2963, 2964, 2965
Valor de arranque, 2948, 2949
Valor predeterminado, 2948
Visualizar valores, 2961
Bloque de datos array, 4376, 4378, 4381
Bloque de datos ARRAY, 2254, 2258, 2295, 2298, 2598, 2937, 2940, 2962, 3030, 4374
Bloque de datos de array, 3468, 3470, 3473, 3476, 3781, 3784, 3786, 3789, 3964, 3967, 3969, 3973, 4572, 4574, 4576, 4580
Bloque de datos de canal, 7461
Bloque de datos de configuración, 8155
Bloque de datos de CPU
Borrar, 2618
Definición, 2256
Bloque de datos de eje
Cargar, 11258
Guardar, 11258
Bloque de datos de instancia
Crear, 2256
Definición, 2256
Modificar el tipo de datos con temporizadores y contadores CEI, 2809
Bloque de datos global
Configurar, 8784
Crear, 8784
Bloque de datos globales, 2254
Bloque de datos globales
Comportamiento remanente, 2952
Bloque de datos instancia
Comportamiento remanente, 2951
Bloque de datos tecnológico, 10640, 10747
Cambiar datos relevantes para la reinicialización, 10751
Evaluación del bloque de datos tecnológico, 10747
Evaluar StatusWord, ErrorWord y WarningWord, 10749
Variables del objeto tecnológico Eje de posicionamiento, 10798
Variables del objeto tecnológico Eje de velocidad de giro, 10819
Variables del objeto tecnológico Encóder externo, 10830

Bloque de función (FB)
Bloque de datos de instancia, 2253
Crear, 2597
Definición, 2253
Exportar a un archivo fuente externo, 2978
Bloque de organización (OB)
Activar OBs de alarma horaria desde el programa de usuario, 6889
Asignar evento con ATTACH, 4884
Crear, 2596
Deshacer asignación de eventos con DETACH, 4885
Determinar tiempo de ejecución del programa con OB_RT, 7023
Exportar a un archivo fuente externo, 2978
Función, 2252
Información de arranque, 2252
Leer información de arranque con RD_SINFO, 4923, 6961
Mostrar, 2234
OB de alarma cíclica, ajustar parámetros con SET_CINT, 4886
OB de alarma cíclica, consultar parámetros con QRY_CINT, 4888
OB de alarma de actualización (OB 56) (S7-300, S7-400), 1836
OB de alarma de diagnóstico (OB 82) (S7-300, S7-400), 1851
OB de alarma de estado (OB 55) (S7-300, S7-400), 1832
OB de alarma de sincronismo (de OB 61 a OB 64) (S7-300, S7-400), 1844
OB de alarmas específicas del fabricante (OB 57) (S7-300, S7-400), 1840
OB de arranque en frío (OB 102) (S7-300, S7-400), 1875
OB de cambio de módulo (OB83) (S7-300, S7-400), 1854
OB de error de acceso a la periferia (OB 122) (S7-300, S7-400), 1883
OB de error de comunicación (OB 87) (S7-300, S7-400), 1871
OB de error de ejecución del programa (OB 85) (S7-300, S7-400), 1859
OB de error de fuente de alimentación (OB 81) (S7-300, S7-400), 1848
OB de error de hardware en CPU (OB 84) (S7-300, S7-400), 1858
OB de error de programación (OB 121) (S7-300, S7-400), 1880
OB de error de tiempo (OB 80) (S7-300, S7-400), 1845
OB de fallo del rack (OB 86) (S7-300, S7-400), 1864
OB de interrupción del procesamiento (OB 88) (S7-300, S7-400), 1872
OB de rearranque (OB 101) (S7-300, S7-400), 1875
OB de rearranque completo (en caliente) (OB 100) (S7-300, S7-400), 1875
OB de tarea no prioritaria (OB 90) (S7-300, S7-400), 1874
OBs de alarma cíclica (de OB 30 a OB 38) (S7-300, S7-400), 1828
OBs de alarma de proceso (de OB 40 a OB 47) (S7-300, S7-400), 1830
OBs de alarma de retardo (de OB 20 a OB 23) (S7-300, S7-400), 1827
OBs de alarma horaria (de OB 10 a OB 17) (S7-300, S7-400), 1823
Programa cíclico (OB1) (S7-300, S7-400), 1822
Vista general de todos los OB, 1819
Bloque de sistema
Carpeta Bloques de sistema, 49, 2595
Mostrar, 2234
Bloque lógico, 1910
bloquear llamadas de lazos de regulación, 7322
Bloques de aviso, 3173
Bloques de datos empleados
ABS_CTRL, 7421
ABS_CTRL_451, 7425
ABS_DIAG, 7422
ABS_DIAG_451, 7427
CAM_CTRL, 7454
CAM_CTRL_452, 7457
CAM_DIAG, 7455
CAM_DIAG_452, 7459
CAM_MSRM_452, 7460
Bloques de datos utilizados
ABS_INIT, 7419
Bloques de datos web, 960, 998
Bloques de organización (OB)
Activar OB de alarma de retardo desde el programa de usuario, 6892
Anular OB de alarma de retardo desde el programa de usuario, 6893
Consultar OB de alarma de retardo desde el programa de usuario, 6894
Bobina de relé, 3286, 5605
negada, 3287
BOOL, 2325, 2376, 2411, 2464, 2481, 2511, 2522
BOOL_TO_, 2411, 2481, 2522
Borrada, 8715
Borrado total, 1628, 1713, 1907, 1913

ejecutar, 2188
Borrado total de la CPU, 1713
Borrar
Autorización, 8614
Componente de hardware, 525
Filtros en el plano de ocupación, 3060
Grupo de usuarios, 8613
Imagen, 8314
Objeto, 8331
Plantilla, 8320
Registro de receta, 9336
Usuarios, 8612, 9319
Variable, 8470
WinAC RTX, 39
Borrar búfer de recepción con RES_RCVB, 8242
Borrar certificado CA, 11408
Borrar distancia residual, 10925, 11020
Borrar una conexión, 184
Borrar WinAC RTX, 39
BorrarBúferDeAvisos, 8673
BorrarBúferDeAvisosProtool, 8674
Botón, 8448, 9309
Comportamiento, 9309
configurar, 8629
Configurar una protección de acceso, 8625
Definir hotkey, 8449
Gráfico, 8449
Manejo, 9309
Modo, 8449
Representación, 9309
Texto, 8449
Utilización, 9309
Botones e interruptores
Librería, 9138
BotónPulsar, 8685
BotónSoltar, 8686
BR, 2545, 2559, 4087, 5764, 5985, 6182
BRCV, 5261
Broadcast, 639, 684, 791, 799, 1041, 1129, 1346
Con UDP, 671
BSEND, 5258
BTD, 4179, 6272
BTI, 4176, 6270
Bucle, 4225, 4436, 4440, 4442, 4443, 4445, 4446, 6317, 6456, 6461, 6463, 6464, 6465, 6466
Bucle , 4225, 6317
Búfer alternativo, 8830, 8993
Búfer de avisos
En runtime, 8534, 9321
Búfer de diagnóstico, 936
Lista de estado del sistema, 7019
Organización, 1730, 2204
Principios básicos, 1635, 1730, 2204
Búfer de peticiones, 683, 684, 685, 8193
Búfer de recepción, 1033, 1089, 1127
Buscar y reemplazar, 417
Iniciar la búsqueda, 419
Opciones adicionales para la búsqueda, 418
Reemplazar términos de búsqueda, 419
Uso de la función de búsqueda, 418
Búsqueda
Catálogo de hardware, 513
Búsqueda de la marca de referencia, 11209
Búsqueda del punto de referencia, 10914, 11008
Cancelar, 10918, 11012
Datos en el DB de canal, 10917, 11011
Datos en el DB de parámetros, 10918, 11012
Efectos, 10918, 11012
Requisitos, 10915, 11009
Sincronización, 10914, 11008
Tipos, 10918, 11012
Velocidad de inicio, 10956, 11057
BUSY en instrucciones que funcionan asíncronamente, 5595
Buzón de datos
para recetas, 8573
BY, 4440
BYTE, 2326, 2377, 2412, 2465, 2482, 2512, 2523
Byte de datos (DBB), 1631
Byte de datos locales (LB), 1631
Byte de entrada (IB), 1631
Byte de marca (MB), 1631
Byte de salida (QB), 1631
BYTE_TO_, 2412, 2482, 2523
Bytes
Cambiar disposición, 4372
Cambiar disposición, 3466, 3779, 4570, 6575

C

Cabecera de petición, 8193
Cable de conexión
Cable Allen-Bradley 1747-CP3, 9025
Cable Allen-Bradley 1761-CBL-PM02, 9026
Cable Allen-Bradley 1784-CP10, 9024
Cable multipunto 1:MP/TP/PC, 9106
Cable multipunto 2:RS422, MP/TP/PC, 9107
Cable punto a punto 1,
Cable punto a punto 2, 9091
Cable punto a punto PP2 para Omron, 9107, 9108
Mitsubishi FX, 9059
Modicon, 9087
Omron Host Link, 9104

Cable de conexión de 9 polos
Sub D, RS 422, 9023
Cable multimaestro RS232/PPI, 1663
Cables de conexión
6XV1440-2P para el protocolo PG de Mitsubishi, 9059
CAD, 4191, 6284
Cadena de caracteres, 2350, 2351
Borrar caracteres con DELETE, 4732, 6726
Buscar caracteres con FIND, 4735, 6728
Comparar variables de string con S_COMP, 4707, 6715
Concatenar con CONCAT, 4730, 6723
Convertir a hexadecimal con ATH, 6719
Convertir con S_CONV, 4708
Convertir desde hexadecimal con HTA, 4727, 6721
Convertir en hexadecimal con ATH, 4724
Convertir número en cadena de caracteres con VAL_STRG, 4711
Convertir una cadena de caracteres numéricos con STRG_VAL, 4709
Copiar, 6716
Copiar los caracteres en una cadena de caracteres con Chars_TO_Strg, 4716
Determinar la longitud máxima con MAX_LEN, 4717
Determinar longitud con LEN, 4729, 6722
Insertar caracteres con INSERT, 4733, 6727
Leer los caracteres centrales con MID, 4732, 6725
Leer los caracteres derechos con RIGHT, 4731, 6724
Leer los caracteres izquierdos con LEFT, 4730, 6724
Reemplazar caracteres con REPLACE, 4734, 6727
Cadena secuencial
Borrar, 2887
Cambiar nombre, 2884
Comentar, 2884
Elementos, 2828
Finalizar, 2886
Inicializar, 3105
Insertar, 2883
Insertar elementos nuevos, 2885
Principios básicos, 2827
CALC, 3396, 3708
Calcular, 2690, 2734, 3396, 3708
Calcular coordenada
Decalaje de señal cero, 9760, 9838
Calcular cuadrado, 4540
Calcular raíz cuadrada, 4541
Calcular valor exponencial, 4345, 6428
CALCULATE, 2690, 2734
Calibración, 2201
Resumen, 2201
CalibrarPantallaTáctil, 8663
CalibrateTouchScreen, 8663
CALL, 4238, 5763, 5984, 6329
CAM_CTRL, 7453
Bloques de datos empleados, 7454
Comportamiento en caso de error, 7455
Descripción, 7453
Estado de la petición, 7474
Funcionamiento, 7454
Llamada, 7454
Peticiones, 7473
Selectores de función, 7474
Valores de retorno, 7454
CAM_CTRL_452, 7457
Bloques de datos empleados, 7457
Comportamiento en caso de error, 7458
Descripción, 7457
Estado de la petición, 7476
Funcionamiento, 7457
Llamada, 7457
Parámetros, 7457
Peticiones, 7474
Selectores de función, 7475
Valores de retorno, 7458
CAM_DIAG
Arranque, 7456
Bloques de datos empleados, 7455
Comportamiento en caso de error, 7456
Descripción, 7455
Funcionamiento, 7455
Llamada, 7455
Peticiones, 7456
Valores de retorno, 7456
CAM_DIAG_452
Arranque, 7459
Bloques de datos empleados, 7459
Comportamiento en caso de error, 7460
Descripción, 7458
Funcionamiento, 7459
Llamada, 7459
Parámetros, 7459
Peticiones, 7459
Valores de retorno, 7459
CAM_INIT
Descripción, 7453
Llamada, 7453
Parámetros, 7453

Valores de retorno, 7453
CAM_MSRM_452, 7460
Arranque, 7461
Bloques de datos empleados, 7460
Comportamiento en caso de error, 7461
Descripción, 7460
Llamada, 7460
Parámetros, 7460
Valores de retorno, 7461
Cambiador de herramientas, (Véase Docking Station)
Cambiar
Nombre de visualización de los grupos de usuarios, 8613
Cambiar el estado operativo, 2187
Cambiar entre el modo básico y el modo avanzado en la tabla de observación, 3117
Cambiar nombre
Autorización, 8613
Grupo de usuarios, 8613
Imagen, 8314
Plantilla, 8320
Usuarios, 8611
Variable, 8470
Cambiar y visualizar el modo de operación (ejemplo), 8641
CambiarConexión, 8705
Cambio de color
Barra, 8424
Cambio de dispositivo, 59
Cambio de DPV0 a DPV1, 1437
Cambio de idioma, 968, 1006, 9218, 9290
fichero, 9223
idioma de runtime, 9220
Cambio de imagen, 8718
Cambio de sentido, 10931, 11031
Cambio de usuario, 8718, 8732
Cambio en el valor real, 9757, 9834
Cambio rápido de levas
Comprobación de datos, 9767, 9845
Datos utilizados en el DB de canal, 9766, 9844
Definición, 9766, 9844
Desarrollo, 9766, 9844
Deshacer, 9768, 9846
Efectos, 9767, 9845
Requisitos, 9766, 9844
Campo, 2354, 2355
Véase ARRAY, 2651, 2945
Campo de aplicación, 1607, 11196
Campo de fecha y hora, 8426
Comportamiento, 9301
Formato, 8427
Manejo, 9301
Mostrar hora del sistema, 8427
Representación, 9300
Utilización, 9300
Utilizar variable, 8427
Campo de texto, 8453
Tamaño, 8454
Campo E/S, 8427, 9302
Cambiar entre campos de entrada, 9303
Campo E/S, 9303
Comportamiento, 9302
Entrada oculta, 8429
Formato de datos, 8428
Formato de representación, 66
Formato decimal, 66
Manejo, 9302
Manejo con pantalla táctil, 9303
Modo, 8428
Representación, 9302
Utilización, 9302
Campo E/S gráfico, 8432, 9305
Comportamiento, 9305
Manejo, 9305
Utilización, 9305
Visualizar lista de gráficos, 8391
Campo E/S simbólico, 8450, 9311
Comportamiento, 9312
Lista de textos, 8380, 8451
Manejo, 9311
Modo, 8450
Utilización, 9311
Campos de salida
Variable multiplexada, 70
CAN_DINT, 4901
CAN_TINT, 4894
Canal de referencia, 2125
Canal de referencia del módulo, 2136
Cancelación del proceso de calibración, 2203
Cancelar, 10910, 11004
Búsqueda del punto de referencia, 10918, 11012
JOG, 10913, 11007
Modo incremental, 10925, 11020
Cancelar impresión, 409
Cancelar preselección del valor real, 9757, 9835, 10927, 11022
AREALREM_EN, 9759, 9837
Datos en el DB de canal, 10927, 11023
Capa 2, 773
Capa 3, 773
Capacidad de memoria, 1722
Capacidad de memoria flexible, 1722
Capacidad de relectura de parámetros de conexión, 718

Capacidades del búfer de recepción, 1011
Captura (Capture), 9361, 9362, 9368, 9370, 9417, 9441, 9447
Captura de bordes, 11027
    Datos en el DB de canal , 11029
    Datos en el DB de parámetros, 11029
    Desarrollo, 11028
    Errónea, 11029
    Requisitos, 11027
CAR, 4170, 6264
Carácter, 2349
Carácter de comprobación de bloque, 1095
Carácter de control de bloque, 1034
Carácter de fin, 1119
Carácter de inicio, 1029
Caracteres asiáticos
    Interpretación, 9291
    Introducción en el panel de operador, 9291
    Memoria necesaria, 9291
Caracteres de control, 1095
    en textos de aviso, 1115
Caracteres especiales
    Texto de aviso, 70
Caracteres orientales
    Introducción en el panel de operador, 9291
Características
    Engineering System, 9339
    Panel de operador, 9341, 9344
Carga
    Mensaje de error, 9275
    Proyecto, 9260
Carga a través del routing S7, 9258
Carga avanzada, 76
Carga de ciclo, 1651, 1941
Carga de ciclo por comunicación, 1741
Carga de comunicación, 1646, 1651, 1931, 1941
Carga de CPU por Trace, 11356
Carga de firmware en CPs PROFIBUS S7-300/400 a través del CP 5612 / CP 5622, 2067
Carga de la memoria, 1911, 3076
    Datos de remanencia, 3076
    Introducción, 3076
    Memoria de carga, 3076
    Memoria de trabajo, 3076
    Memoria de trabajo para código, 3076
    Memoria de trabajo para datos, 3076
    Memoria remanente, 3076
Carga de la memoria de la CPU
    Mostrar, 3079
Carga de un proyecto
    con panel de operador conectado, 9258
Cargar, 4157, 4159, 4160, 4161, 4163, 4164, 4165, 4230, 4231, 4232, 4233, 6251, 6253, 6254, 6255, 6257, 6258, 6259, 6322, 6323, 6324
    Aplicaciones isócronas, 1611, 1617
    Bloque de datos de eje, 11258
    Bloques en un dispositivo, 3021, 3038
    Carga avanzada, 76
    Cargar bloques de una Memory Card, 3029
    Cargar bloques desde una Memory Card, 3044
    Cargar bloques en un dispositivo, 3023, 3024, 3039, 3040
    Cargar bloques en una Memory Card, 3028, 3043
    Cargar cambios de bloque sin reinicializar, 3030, 3032, 3033, 3034, 3035, 3036
    Cargar en una Memory Card, 368
    Cargar los datos de proyecto en un dispositivo, 367
    Comentarios de módulo, 41
    Comodín, 1618
    Conexión online, 1621
    Configuraciones con PROFIBUS, 1616
    Configuraciones con PROFINET, 1617
    Configuraciones con servidor web, 1616
    Configuraciones de dispositivos, 1613
    de un dispositivo, 39, 48, 370
    Dispositivos HMI, 1620
    en dispositivo, 39, 1611
    En el modo operativo RUN, 3038
    en estado operativo RUN, 3021
    en PG/PC, 1612
    Estación SIMATIC PC, 76
    GSDML, 1617
    I-Device, 1618
    IE/PB-Link, 1619
    Información general, 364
    Panel de operador, 9268
    Periferia descentralizada, 1614
    Proyecto, 9280
    Shared Devices, 1619
    Subredes, 1615
    Valor de contaje, 4113, 4114, 6207, 6209
    Valor del temporizador, 4093, 4095, 6189, 6191
Cargar datos en el PLC
    Mensaje de error, 75
Cargar diseño de ventana
    Carga mediante un acceso rápido, 311
    Cargar otros diseños de ventana, 311
Cargar la copia de seguridad de un dispositivo online, 11307
Carpeta de bloques, 49, 2595
CASE, 4438, 6458

Caso de repuesto, 8048, 8191
Catálogo, (véase Catálogo de hardware)
Catálogo de hardware
Acoplador DP/DP, 1389, 1464
Agregar dispositivo, 521
Esclavo DP, 1389, 1463
Esclavo I, 1389, 1464
examinar, 512
Selección de los componentes HW, 518
Selección de módulo central, 906
Selección de módulos de seguridad, 908
Task Card, 504, 8757
Categoría, 8510
Copiar, 3198
Crear, 3197
Edición, 3198
En runtime, 8534, 9320
Representación, 8535, 9323
Categorías, 8506
Cambio de nombre mediante la migración, 157
personalizadas, 8506
predefinidas, 8506
utilizar, 8506
Categorías personalizadas, 8506
Categorías predefinidas, 8506, 8507
Causa posible
Transferir datos, 8829, 8882, 8940, 8979, 9132
CAW, 4189, 6283
CC0, 2545, 2551, 2552, 2553, 2555, 2559, 2563
CC1, 2545, 2551, 2552, 2553, 2555, 2559, 2563
CD, 3369, 3681, 4119, 5662, 5886, 6213, (Véase Comunicación directa)
CDB, 4229, 6321
CEIL, 3496, 3809, 4418, 4595, 5716, 5939, 6448, 6591
Cerrar rama, 2830
Cerrar sesión
Usuarios, 9316
CerrarSesión, 8647
Certificado, 760, 816
Exportar, 760
firmado por el usuario, 762
firmado por una entidad emisora, 762
Importar, 760
reemplazar, 763
renovar, 762
sustituir, 763
Certificado CA, 759, 763, 11403
Certificado de grupo CA, 763
Certificado SSL, 763
Certificados FTPS, 760
Certificate Authority, 759
CH_DIAG
Diagrama de bloques, 7511
Instrucción, 7508
Parámetros de entrada, 7509
Parámetros de salida, 7509
CH_DIAG_455
Diagrama de bloques, 7596
Instrucción, 7594
Parámetros de entrada, 7594
Parámetros de salida, 7594
Change word order
Modicon MODBUS TCP/IP, 9074
ChangeConnection, 8705
Char
Variable de matriz, 69
CHAR, 2349, 2406, 2456, 2479, 2505, 2521, 2537
CHAR_TO_, 2456, 2505, 2537
Chars_TO_Strg, 4716
CHECKBACK_SIGNALS
CNT2_CTR, 7414
Cíclico
continuo, 8479
durante el funcionamiento, 8479
Ciclo de actualización, 8498
Ciclo de adquisición
Punteros de área, 8782
variable, 8485
Variable, 8479, 8498
Ciclo de aplicación, 1540
Ciclo de procesamiento, 1446
Ciclo DP
Tiempo para ciclo DP equidistante, 1445
Ciclos de bus, 1444
Ciclos de procesamiento sincronizados, 1446
Cifrado, 753
CiR, 5737, 5960, 6091, 6476, 6608
Circuito
Borrar, 2712
Insertar, 2711
Utilización, 2710
Circuito lógico
Borrar, 2752
Insertar, 2752
Utilización, 2751
Circuito secuencial sucesivo, 3564, 3879, 4026, 4485, 4643, 5784, 5788, 6005, 6009, 6111, 6119, 6503, 6506, 6636, 6640
Círculo, 8434
Radio, 8435
CJ_T_PAR
Instrucción, 7518
Parámetros de entrada, 7519

Parámetros de salida, 7519
CJ_T_PAR_455
Instrucción, 7602
Parámetros de entrada, 7603
Parámetros de salida, 7603
CJ1, 9109
CJ2, 9109
Clase de evento, 1885
Clase de evento 1 - Eventos de OB estándar, 1886
Clase de evento 2 - Eventos de error síncronos, 1886
Clase de evento 3 - Eventos de error asíncronos, 1887
Clase de evento 4 - Eventos de parada y otras transiciones de estado operativo, 1890
Clase de evento 5 - Eventos de procesamiento de estado operativo, 1894
Clase de evento 6 – Eventos de comunicación, 1895
Clase de evento 8 - Eventos de diagnóstico para módulos, 1896
Clase de evento 9 - Eventos de usuario estándar, 1898
Clase de prioridad, 1826, 1890
Clase de visualización de mensajes, 945
Clase RT, 1487
Clase RT PROFINET, 1487
Clases de error
Avisos, 9829, 9909
Error de datos de leva, 9828, 9908
Error de diagnóstico, 9829, 9909
Fallo operativo, 9823, 9903
Clases de error PtP, 5399, 7789
Clases de errores
Error de datos, 9824, 9904, 10982, 11083
Error de datos máquina, 10983, 11085
Error de las tablas de incrementos, 10985, 11087
Error de operador, 10981, 11082
Fallo operativo, 10980, 11081
Clases de evento A y B - Eventos libres de usuario, 1900
Clases de evento reservadas, 1900
Clases de fallos
Avisos, 10986, 11087
Error de datos máquina, 9826, 9906
Error de diagnóstico, 10987, 11088
Class of Service, 1309
Clave de licencia, 90, 103
Uso de las claves de licencia, 90, 103
ClearAlarmBuffer, 8673
ClearAlarmBufferProtoolLegacy, 8674
Cliente DNS, 1268
Cliente FTP
S7-1500, 2030
Cliente OPC UA, 2059
Cliente SMTP, 1276
Cliente Syslog, 1290
CLR, 4086, 6182
CMP <, 3381, 3693, 5670, 5893
CMP <=, 3377, 3689, 5668, 5891
CMP <>, 3373, 3685, 5665, 5889
CMP ==, 3371, 3682, 5664, 5887
CMP >, 3379, 3691, 4680, 4682, 4687, 4688, 5669, 5892, 6676, 6681, 6682
CMP >=, 3375, 3687, 5667, 5890
CNT_CTL1, 7390
Comportamiento en arranque, 7392
Comportamiento en caso de error, 7392
Descripción, 7390
Funcionamiento, 7390
Llamada, 7391
Parámetros, 7391
Peticiones, 7392
Transferencia de valores, 7400
CNT_CTL2, 7393
Descripción, 7393
Funcionamiento, 7393
Llamada, 7393
CNT_CTRL, 7388
Comportamiento en arranque, 7389
Comportamiento en caso de error, 7389
Descripción, 7388
Funcionamiento, 7388
Llamada, 7388
Parámetros, 7388
Peticiones, 7389
CNT2_CTR, 7405
CHECKBACK_SIGNALS, 7414
CONTROL_SIGNALS, 7414
Descripción, 7405
Funcionamiento, 7406
Llamada, 7406
Parámetros, 7406
CNT2RDPN, 7405
Descripción, 7405
Estado de la petición de lectura, 7418
Funcionamiento, 7405
Llamada, 7405
Parámetros, 7405
Petición de lectura, 7418
CNT2WRPN, 7404
Descripción, 7404
Estado de la petición de escritura, 7417
Funcionamiento, 7404

Llamada, 7404
Parámetros, 7404
Petición de escritura, 7415
Codificar, 3547, 3860, 4022, 4469, 4627, 5747, 5970, 6095, 6480, 6613
Código binario natural, 9398
Código Gray, 9398
Códigos de error
Gx_XIST2, 11210
Códigos de estado USS, 5530
Coeficiente de temperatura, 2114, 2134
Coherencia
Reglas para slots, 512
Coherencia de bloques
Comprobar, 3069
Comprobar en la estructura de dependencias, 3075
Coherencia de datos, 5244, 7668
Color
Diagnóstico de cables Ethernet, 2166
Diagnóstico de puertos, 2166
Color de fondo
modificar dinámicamente, 8395
Color de primer plano
modificar dinámicamente, 8395
Color transparente, 8431
Visualización en paneles de operador:, 9304
Visualización en Panels, 8431
Columna
Visor de avisos, 8440
Comando
visor de recetas simple, 8592
Visor de recetas simple, 9333
Comando AWP, 949, 950, 986, 987
Comando de control SYNC/FREEZE, 1403
Comando PING, 8069, 8070
Comandos de teclado, 507
Ajuste de editores, 319
AWL, 2623
Editar objetos, 321
Editar textos, 321, 2623
Editor de programación, 2623
Funciones básicas del TIA Portal, 315
Funciones online, 324
FUP, 2623
GRAPH, 2623
KOP, 2623
Navegación en el TIA Portal, 318
Procesamiento del proyecto, 316
SCL, 2623
Seleccionar objetos, 320
Tablas, 323
Ventana, 316
Combinar lógicamente
O, 3540, 3854, 5741, 5964
O-exclusiva, 3542, 3856, 5743, 5966
Y, 3539, 3852, 5740, 5963
Comentario
Punteros de área, 8782
Comentario de línea, 2781
Comentario del bloque
Mostrar, 2629
Ocultar, 2629
Comentario del segmento
Mostrar, 2629
Ocultar, 2629
Comentario libre
Borrar, 2693, 2737
Editar, 2692, 2736
Insertar, 2692, 2736
Introducción, 2691, 2735
Comentarios
insertar en el programa SCL, 2822
Comentarios de módulo, 41
Comfort Panel
S7 -1200, 76
Compact, 9079, 9093
CompactLogix, 9006
Comparación
Variables de tiempo, 4690, 6683
Comparación de bloques
Actualizar resultados de la comparación, 385, 3008
Comparación detallada, 2988, 2989, 2993, 2996, 2999
Comparar bloques de datos, 2985
Comparar bloques lógicos, 2984
Ejecutar acciones, 387
Modificar bloque, 3007
Navegación, 3006
Principios básicos, 2982
Sincronizar el desplazamiento, 3007
Comparación de objetos de librería, 428
Comparación detallada
Ejecutar, 384
Inicio, 2988
Representación del resultado de la comparación para AWL, 2993
Representación del resultado de la comparación para GRAPH, 2999
Representación del resultado de la comparación para KOP/FUP, 2989
Representación del resultado de la comparación para SCL, 2996

Comparar, 50
Cadenas de caracteres, con S_COMP, 4707, 6715
Diferente, 3373, 3685, 5665, 5889
Enteros, 4121, 6215
Enteros dobles, 4122, 6216
Igual, 3371, 3682, 5664, 5887
Máscara de bits, 3575, 3891, 4042, 4495, 4659, 5799, 6020, 6131, 6517, 6652
Mayor, 3379, 3691, 4680, 4682, 4687, 4688, 5669, 5892, 6676, 6681, 6682
Mayor o igual, 3375, 3687, 5667, 5890
Menor, 3381, 3693, 5670, 5893
Menor o igual, 3377, 3689, 5668, 5891
Números en coma flotante, 4124, 6218
Comparar offline/online
Asignación automática de dispositivos, 747
Compatibilidad, 39
Compatibilidad de proyectos con versiones anteriores, 341
Librerías globales de versiones anteriores del producto, 442
Proyectos con software adicional, 342
Proyectos de versiones anteriores del programa, 341
Versiones de firmware de CPU S7-1200, 1699
Versiones de WinCC, 9245
Compatibilidad hardware, 1931
Compatibilidad hardware soportada, 1931
Compensación de temperatura, 2101
Compilar
Bloque, 3015
Bloques, 3012
Comprobación de coherencia, 3012
Parámetros de dirección, 75
Proyecto, 9256
proyecto migrado, 150
Solucionar errores de compilación, 3017
Complemento a 1, 3544, 3858, 4184, 4185, 5744, 5967, 6277, 6278
Complemento a diez, 3586, 3901, 4059, 4505, 4673, 5817, 6041, 6155, 6533, 6670
Complemento a dos, 3407, 3719, 4186, 4187, 4188, 4532, 5679, 5902, 6279, 6280, 6282, 6556
Componente ARRAY, 47
Componente PROFINET CBA
Actualizar con instrucciones, 8263
Componentes
Aviso, 8510
puesta en marcha, 9034, 9053, 9063, 9081, 9094, 9111
Puesta en marcha, 9013
Componentes del panel de operador, 9250
Comportamiento
Botón, 9309
Campo de fecha y hora, 9301
Campo E/S, 9302
Campo E/S gráfico, 9305
Campo E/S simbólico, 9312
Interruptor, 9310
visor de recetas simple, 8592
Visor de recetas simple, 9333
Comportamiento cíclico, 1740
Comportamiento de arranque, 1196
Comportamiento de conmutación
de la leva de tiempo, 9757, 9834
Comportamiento de las salidas digitales
Activar al alcanzarse el valor de comparación por la duración del impulso hacia atrás, 9523, 9603
Activar al alcanzarse el valor de comparación por la duración del impulso hacia delante, 9523, 9602
Activar cuando se alcance el valor de comparación para la duración del impulso hacia delante/atrás, 9522, 9602
Activar si la lectura del contador <= valor de comparación, 9522, 9601
Activar si la lectura del contador >= valor de comparación, 9521, 9601
Ajuste predeterminado, 9520, 9566
Condiciones marco, 9525
Conmutar en los valores de comparación, 9524
en los modos de contaje, 9518, 9599
en los modos de medición, 9529
en modo isócrono, 9520
No se compara, 9521
Requisitos, 9603
Sin comparación, 9601
Comportamiento de remanencia tras un corte de la alimentación, 1706
Comportamiento de sobrecarga, 1926
Comportamiento de un motor Stepper, 11201
Comportamiento en arranque
CNT_CTL1, 7392
CNT_CTRL, 7389
Comportamiento en caso de cambio de idioma
Teclado de pantalla, 77
Comportamiento en caso de error
ABS_CTRL, 7422, 7426
ABS_DIAG, 7423
ABS_DIAG_451, 7427
CAM_CTRL, 7455
CAM_CTRL_452, 7458
CAM_DIAG, 7456

CAM_DIAG_452, 7460
CAM_MSRM_452, 7461
CNT_CTL1, 7392
CNT_CTRL, 7389
High_Speed_Counter, 5126
Comportamiento operativo
Configuración estándar, 1735
Configurar y cargar parámetros, 1735
Parámetros del módulo, 1735
Uso de marcas de ciclo, 1669, 1955
Uso de marcas de sistema, 1668, 1954
Utilizar funciones de reloj, 1653, 1749, 1952
Utilizar marcas de ciclo, 1751
Utilizar temporizadores y contadores, 1731
Comportamiento remanente, 1742
Marcas, temporizadores, contadores, 2578
Composición
de la lista de referencias cruzadas, 3081
Compresión de direcciones, 1407, 1477
COMPRESS, 5731, 5954, 6083, 6470, 6600
comprimidas
Alarma con direcciones, 1484
Comprimir, 5731, 5954, 6083, 6470, 6600
Memoria en la vista online y de diagnóstico, 2243
Comprobación de coherencia, 3069, 3074, 3075
Introducción, 3069, 3074
Comprobación de datos, 9767, 9845
por el módulo, 9767, 9845
Comprobación de la coherencia antes de la migración, 130
Comprobación del valor límite, 9296, 9297
Comprobar
Versión de panel, 9250
Comprobar conexión, 5331
Comprobar la capacidad de migración, 129
Comunicación, 1674, 1959, 8734
Arquitectura de consultas, 5396, 7786
Carga de ciclo, 1651, 1941
Datos de estado, 6993
Definición, 8734
Drivers de terceros, 8997
mediante punteros de área, 8741
Open User Communication, 615, 7725
Principios básicos, 8735
Recursos de conexión, 1754
S7 1200, 8841
S7 1500, 8792
S7 200, 8957
S7 300, 8896
S7 400, 8896
SIMATIC LOGO!, 8986
Comunicación a través de la instrucción PUT/GET
Borrar conexión, 729
Crear y parametrizar la conexión, 728
Iniciar la parametrización de la conexión, 727
Principios básicos de la instrucción PUT/GET, 723
Requisitos, 723
Resumen de la parametrización de la conexión, 724
Comunicación básica S7
I_ABORT, 8261
I_PUT, 8259
Parámetros comunes, 7661
X_ABORT, 8277
X_GET, 8273
X_PUT, 8275
X_RCV, 8269
X_SEND, 8268
Comunicación directa, (Véase Comunicación directa)
Configurar, 1425
Crear, 1425
Esclavo DP - esclavo I, 1391, 1393
Esclavo DP - maestro DP, 1391
Funcionamiento, 1423
PROFIBUS, 1424
Comunicación Modbus, 1040
Como esclavo con MB_SLAVE, 5543
Como maestro Modbus con MB_MASTER, 5535
Configurar puerto con MB_COMM_LOAD, 5532
Comunicación MPI
S7 300/400, 8918
Comunicación PROFIBUS
S7 1200, 8857
S7 1500, 8808
S7 300/400, 8910
Comunicación PtP
Programar, 5394, 7784
Comunicación punto a punto (PtP), 1659
Protocolo Freeport, 1659
Comunicación S7
Ejecutar rearranque completo o arranque en frío con START, 7714
Cambio del estado operativo, 7667
Comportamiento de arranque, 7673
Comportamiento en caso de fallo, 7675
Conmutar el dispositivo al estado operativo STOP con STOP, 7716
Consulta de una conexión, 7668
Consulta del estado operativo, 7667
Consultar el estado de conexión con C_CNTRL, 7694

Consultar el estado del dispositivo con STATUS, 7720
Datos a una instrucción remota del interlocutor con USEND, 5255, 7683
Determinar el estado de conexión con CONTROL, 7705
Ejecutar un rearranque con RESUME, 7718
Enviar datos a una impresora con PRINT, 7708
Enviar datos a una instrucción remota del interlocutor con BSEND, 5258, 7688
Enviar datos a una instrucción remota del interlocutor con USEND_S, 7700
Escribir datos en una CPU remota con PUT, 7680
Escribir datos en una CPU remota con PUT_S, 7698
Intercambio de datos , 7666
Leer datos de una CPU remota con GET_S, 7696
Leer los datos de una CPU remota con GET, 7677
Parámetro, 5246, 7669
Parámetros de transmisión y recepción, 5247, 7672
Recibir datos de una instrucción remota del interlocutor con BRCV, 5261, 7691
Recibir datos de una instrucción remota del interlocutor con URCV, 5256, 7685, 7703
Recibir el cambio del estado del dispositivo con USTATUS, 7722
Sinopsis de instrucciones, 7666
Volumen de datos de usuario, 5248, 7672
Comunicación USS, 1037
Controlar la transferencia al accionamiento con USS_PORT, 5523
Intercambio de datos con accionamientos a través de USS_DRIVE, 5524
Leer parámetros del accionamiento con USS_RPM, 5527
Modificar parámetros en el accionamiento con USS_WPM, 5529
Comunicar errores del sistema, (Véase Report System Errors)
CONCAT, 4730
Concepto de acuse, 8509
Aviso con acuse simple, 8509
Aviso sin acuse, 8509
Concepto de protección, 1671, 1674, 1956, 1959, 1960
Borrar derechos de acceso, 389
Introducción, 389
Concesión de licencias para el panel de operador
No basado en PC, 102
Condición
Insertar una instrucción permanente, 2920
Introducción, 2847
Programar interlock, 2910
Programar supervisión, 2911
Programar transición, 2912
Condiciones necesarias para establecer una conexión remota, 11400
Conectar
Controlador, 9003, 9045, 9059, 9075, 9104
Conectar dispositivos en red
Conectar varias interfaces en red al mismo tiempo, 536
Conexión en red en caso de existir una subred, 538
Conexión en red en caso de no existir ninguna subred, 536
Editar parámetros de interfaz, 541
Editar parámetros de red, 540
Principios básicos sobre la configuración de redes, 534
Redes dentro de un proyecto, 535
Requisitos, 540
Tipos de comunicación, 534
Conectar en red
de interlocutores, 8764
Dispositivos, 8764
Conectar en red en la vista de dispositivos
Posibilidades de la conexión en red, 539
Procedimiento, 539, 540
Conexión, 557, 569, 617, 8740
Allen-Bradley DF1, 9014, 9018
Allen-Bradley EtherNet/IP, 8999
borrar, 707
Con una estación concreta de un proyecto cualquiera, 612
Conexión integrada, 8461, 8776
Configurar, 579, 8765, 8769, 8999, 9014, 9041, 9055, 9070, 9082, 9100
Configurar cuando falta la asignación de red o ésta no es unívoca, 562, 624
Configurar cuando falta la interconexión, 629
Consultar el estado con C_CNTRL, 7694
Contraseña, 8801, 8813, 8852, 8865
Correo electrónico, 579, 583, 586, 588
crear, 704
Crear, 616, 8765, 8769
Detalles de dirección, 2217
Diagnosticar con C_DIAG, 7027
Drivers de comunicación Omron, 9112

Establecer con TCON, 7748
FDL, 579, 583, 586, 588
Influencia de la interconexión: comportamiento cuando falta la conexión en red, 617
Integrada, 8765
Interrumpir la conexión con el interlocutor dentro de la estación S7 propia con I_ABORT, 8261
Interrumpir la conexión con el interlocutor fuera de la estación S7 propia con X_ABORT, 8277
ISO-on-TCP, 579, 583, 586, 588
ISO-Transport, 579, 583, 586, 588
Mitsubishi MC TCP/IP, 9041
Modicon, 9087
Modicon Modbus RTU, 9082
Modicon Modbus TCP, 9070
No especificada, 617
offline, 8873
Offline, 8821, 8931, 8972
Omron Hostlink, 9100
Parámetro, 9001, 9056, 9084, 9102
Parámetros, 9016, 9043, 9072
Programada, 613, 615
Punteros de área, 8780, 8785
Resaltar, 8767
S7 200, 8957, 8986
Sin especificación del puerto, 612
Tabla, 8764
TCP, 579, 583, 586, 588
UDP, 579, 583, 586, 588
Ver ocupadas y reservadas, 2237
Conexión de accionamientos para Motion Control, 10646
Conexión de accionamientos y encóders, 10646
Conexión de comunicación, 579
Conexión de correo electrónico, 2005
Configurar, 687
Conexión de encóders para Motion Control, 10646
Conexión de un TS Adapter con un módem externo., 11381
Conexión de un TS Adapter con un módem interno., 11381
Conexión FDL
Acceso libre Layer 2, 683
Con Broadcast, 684
Con Multicast, 685
No especificada, 683
Conexión HMI, 181, 569, 1674, 1959, 8740
Contraseña, 8801, 8813, 8852, 8865
Crear, 8794, 8843, 8898
MPI, 8919
Parámetros MPI, 8921
Parámetros PROFIBUS, 8809, 8859, 8913
Parámetros PROFINET, 8796, 8845, 8900
PROFIBUS, 8806, 8911
PROFINET, 8793, 8794, 8842, 8843, 8897, 8898
S7 1200, 8843
S7 1500, 8794
S7 300/400, 8898
Conexión integrada, 8461, 8776
Conexión Modicon Modbus
Tipos de datos, 9097
Conexión NCM
Detalles de conexión, 2248
Conexión no especificada
Características, 640
Configuración gráfica, 637
Configurar a través de cuadros de diálogo, 638
Conexión online, 11299
Conectar varios dispositivos, 11304, 11305
Varias instancias del TIA Portal, 37
Conexión permanente, 1209
Conexión por línea telefónica
Desconectar, 11402
Establecer, 11400
Conexión remota
Como conexión por línea telefónica, 11378
Como conexión remota controlada por CPU, 11414
Como conexión VPN, 11402
Conexión S7, 557
TSAP, 568, 646
Conexión temporal, 1209
Conexión UDP/Conexión FDL
No especificada, 639
Conexión VPN
Borrar certificado CA, 11408
Desconectar, 11410
Establecer, 11408
Instalar certificado CA, 11405
Conexiones
Configuración gráfica, 621
Configurar, 8944, 8946, 8948, 8957, 8986
Editor, 8944, 8946, 8948
Conexiones HMI
Dispositivos y redes, 8765
Conexiones no especificadas
En Industrial Ethernet, 612
En PROFIBUS, 612
CONF_DB See Bloque de datos de configuración, 8155
Configuración
AWL, 2758
Cargar en dispositivo, 1611
Cargar en PG/PC, 1612

Esclavo DP, 1387, 1400, 1473
Esclavo DP, sencillo, 1390, 1464
Esclavo I, 1392
FUP, 2670, 2714
General, 279, 281, 282, 2640
GRAPH, 2864
Hardware, 509
Idiomas del sistema operativo, 9200
KOP, 2670, 2714
Modificar, 283, 2641, 2670, 2715, 2759, 2796, 2866
Módulos centrales, 898
Periferia descentralizada, 1387
SCL, 2795
variable, 8471
Variable HART, 2140
Configuración de avisos
Borrar valor asociado, 3185
Desinhibir atributo, 3194
Desinhibir texto, 3194
Inhibir atributo, 3194
Inhibir texto, 3193
Introducción, 3171
Introducir texto, 3193
Texto adicional, 3193
Texto informativo, 3193
Configuración de conexiones, 579
Gráfica, 616
Mediante cuadros de diálogo, 616
Configuración de conexiones con el servidor OPC, 611
configuración de fábrica
restablecer, 2192
Configuración de fábrica
restablecer, 2193
Configuración de FETCH/WRITE, 2032
Configuración de hardware
Agregar dispositivo, 520
Agregar módulo, 523
Configuración de interfaces
Instrucciones, 5394, 7785
Configuración de línea PROFIBUS
Anillo óptico, 550
Configuración de parámetros
LENGTH y BUFFER para Send_P2P, 5412, 7802
Configuración de puerto, 1293
Configuración de runtime
Administración de usuarios, 8316, 8604, 8621
Configuración de seguridad: Nombres simbólicos, 44
Configuración de submódulo, 1919
Configuración de tramas
Instrucciones, 5394, 7785
Configuración DNS, 2005
Configuración FTP, 2014
Tabla de asignación File, 2014
Configuración futura, 1409, 1412, 1479, 1482, (Véase Control de la configuración (ET 200SP)), (Véase Control de la configuración (ET 200MP))
Configuración futura (ver Control de la configuración), 930
Configuración hardware, 1910
Configuración IP
Modificar parámetros del programa de usuario, 5336, 7766
Configuración LDB, 2070
Configuración LSAP, 2067
Configuración máxima de una S7-1500, 903
Configuración OPC, 2068
Configuración XDB, 1203, 2075
Configurado
Loop-In-Alarm, 8520
configurar
Autorización, 8623
Protección de acceso, 8629
Configurar, 8516, 8525
Acuse de aviso, 8530, 8531, 8532
Aviso analógico, 8516, 8523
Aviso de bit, 8514, 8522
Avisos, 8883, 8983, 8994
Avisos analógicos, 8512
Avisos de bit, 8512
Avisos de controlador, 8512
Avisos de sistema, 8512
Comunicación directa, 1425
Conexiones, 8944, 8946, 8948, 8957, 8986
CP como esclavo I, 1421
CPU como esclavo I, 1422
Direcciones de diagnóstico en el I-Slave, 1418
Esclavo DP como esclavo I, 1423
Esclavo I, 1416, 1474
Estación PC, offline, 1203
Estación PC, online, 1201
Grupo de avisos, 8514
Idioma, 9299
Opciones de visualización de la estructura de dependencias, 3073
Opciones de visualización del plano de ocupación, 3058
Rectángulo, 8372
Representación del visor de avisos, 8527
Tareas controladas por eventos, 8518
varias variables, 8471

Visor de avisos, 8527
Visor de curvas para los valores del controlador, 8499
visor de recetas, 8588
Configurar avisos
Allen-Bradley, 9037
Conexión no integrada, 9066
Particularidades, 9067
Tipos de datos, 9067
Configurar conexiones
Mediante cuadros de diálogo, 625
Configurar el acuse
Alarma, 8832, 8884, 8942, 8984, 8995, 9039, 9068, 9098, 9114
Configurar la red vía Ethernet, 551, 8850, 8905
Ajustar la dirección IP, 551, 8850, 8905
Ajustar la máscara de subred, 552, 8850, 8905
Conectar redes, 553, 8851, 8906
Crear subredes privadas, 553, 8851, 8906
Relación entre la dirección IP y la máscara de subred, 552, 8850, 8905
Configurar manualmente nodos de red IP, 869
Configurar manualmente nodos de red MAC, 869
Configurar manualmente subredes internas, 869
Configurar una conexión
Allen-Bradley DF1, 9014
Allen-Bradley EtherNet/IP, 8999
Mitsubishi FX, 9055
Mitsubishi MC TCP/IP, 9041
Modicon Modbus RTU, 9082
Modicon Modbus TCP, 9070
Omron Hostlink, 9100
Configurar una subred PROFIBUS
Ajustar parámetros de bus, 545
Significado de los parámetros de bus, 546
Sintonizar parámetros entre sí, 545
Congelar, 9502
Congelar/redisparar, 9504
conjuntos de reglas IP, 774
conjuntos de reglas MAC, 774
Conmutar
Entre idiomas de runtime, 9218, 9290
Conmutar entre modo básico y modo avanzado en la tabla de forzado permanente, 3146
Conmutar OFF, 8725
Conmutar ON, 8725
Consigna de seguridad
Cambiar tooltip, 9298
modificar el registro de receta en segundo plano, 8578
Consignas de seguridad, 334
Constante
Constante con tipo, 2288
Constante global, 2289, 2290, 2292, 2583, 2584, 2587
Constante local, 2289, 2290, 2292, 2642, 2648, 2658
Constante simbólica, 2289
Constante sin tipo, 2288
Definición, 2288
Tipos de datos, 2290
Constante de usuario, 927
Constantes, 4135, 6229
Consultar flanco de señal ascendente de un operando, 3295, 3604, 5612, 5832
Consultar flanco de señal ascendente del RLO, 3300, 3610, 5614, 5835
Consultar flanco de señal descendente de un operando, 3296, 3606, 5613, 5834
Consultar flanco de señal descendente del RLO, 3302, 3611, 5616, 5836
CONT_C
Diagrama de bloques, 7081
Funcionamiento, 7079
Parámetros de entrada, 7081
Parámetros de salida, 7083
CONT_S
Diagrama de bloques, 7086
Funcionamiento, 7084
Instrucción, 7084
Parámetros de entrada, 7086
Parámetros de salida, 7088
Contacto inversor, 10915, 11009
Contacto normalmente abierto, 3283, 5602
Contacto normalmente cerrado, 3285, 5603
Contador
Ascendente, 3662, 3674, 3679, 3923, 4118, 4523, 5884, 6212, 6549
Ascendente y descendente, 3668, 3672, 3930, 4529, 6553
Descendente, 3665, 3676, 3681, 3926, 4119, 4526, 5886, 6213, 6551
Habilitar, 4111, 6206
Inicializar, 4117, 6211
Poner a 0, 6210
Poner a cero, 4116
Contador de horas de funcionamiento, 1751
Arrancar y parar con CTRL_RTM, 6696
Inicializar con SET_RTM, 6695
Inicializar, arrancar, parar y leer con RTM, 6694
Leer con READ_RTM, 6697
Manejar con RTM, 4705, 6694
Propiedades, 6693

Rango de valores, 6693
Ver información, 2239
Contador descendente de 32 bits, 2105
Contador rápido
Configurar, 1657
Funcionamiento, 1654
Generalidades, 1654
Contador, rápido, 1654
Contadores
Contadores rápidos, 5121
Controlar contadores rápidos, 5121
Contadores CEI, 255
Contaje
Ascendente, 3351, 3363, 3368, 4316, 4325, 5647, 5656, 5661, 5870, 5879, 6053, 6406, 6413
Ascendente y descendente, 3357, 3361, 4321, 4329, 5651, 5654, 5874, 5877, 6058, 6410, 6417
Continuo, 9468
Descendente, 3354, 3365, 3369, 4319, 4327, 5649, 5658, 5662, 5872, 5881, 6055, 6408, 6415
Periódico, 9470
Único, 9469
Valor de comparación, 9420, 9449
Contaje (8 DI NAMUR), 2103
Contaje (CPU 31xC), 1758
Contaje periódico, 9470
Contaje sin fin, 9468
Contaje único, 9469
Contenidos imprimibles, 391
CONTINUE, 4445, 6465
Contraseña, 1960, 9313
Caducidad de la contraseña, 8606
Complejidad de la contraseña, 8606
Conexión HMI, 8801, 8813, 8852, 8865
Guardar, 9313
Modificar, 8611
Nivel de jerarquía, 8605
para protección de acceso de la CPU, 1646, 1931
para teclas de función, 8411
Restablecer, 9313
Control de acceso, 1340
Aprendizaje automático, 1340
Control de la configuración, 919, 930, 935, 2120, 2132
Control de movimiento, 10679
Control de puerta, 9361, 9364, 9389, 9417, 9441
en modo isócrono, 9490
Control del flujo de datos, 1012, 1019, 1021, 1661
Hardware, 1022
Software, 1022
Control del flujo de datos de hardware, 1022
Control del flujo de datos de software, 1022
Control del flujo de datos por hardware, 1661
Control del flujo de datos por software, 1662
Control del programa, 3526, 3839, 4008, 4456, 4616, 5733, 5956, 6086, 6473, 6603
CONTROL_SIGNALS
CNT2_CTR, 7414
Controlador
Conectar, 9003, 9045, 9059, 9075, 9104
Leer registro de receta, 9337
Parámetros MPI, 8924
Parámetros PROFIBUS, 8812, 8862, 8916
Parámetros PROFINET, 8799, 8848, 8903
Transferir registro de receta, 9338
Variable, 8482
Controlador PROFIenergy, 8116
Controlar
Contadores rápidos, 5121
Controlar contador desde el programa de usuario con COUNT, 7650
ControlLogix, 9006
Convenio de codificación Diffie-Hellman, 817
Conversión, 2373, 2461, 2507
Convertir tiempos con T_CONV, 6683
explícita, 2447, 2490, 2491, 2493, 2495, 2497
Explícita, 2411, 2412, 2418, 2421, 2424, 2426, 2428, 2430, 2433, 2435, 2438, 2440, 2441, 2443, 2444, 2446, 2449, 2450, 2452, 2453, 2454, 2456, 2457, 2458, 2460, 2481, 2482, 2488, 2498, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2522, 2523, 2528, 2530, 2531, 2532, 2533, 2535, 2536, 2537, 2538, 2539
Explícita , 2414, 2416, 2483, 2485, 2524, 2526
implícita, 2376, 2392, 2408, 2511, 2512, 2514, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522
Implícita, 2377, 2379, 2380, 2382, 2383, 2384, 2385, 2386, 2388, 2389, 2390, 2393, 2394, 2395, 2397, 2398, 2399, 2401, 2402, 2403, 2404, 2406, 2407, 2409, 2464, 2465, 2466, 2467, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 2480
Conversión de tipos de datos, 47
Conversores
Convertir tiempos con T_CONV, 4691
CONVERT, 3493, 3805, 4415, 4592, 5713, 5936, 6446, 6588
Convertir, 3493, 3805, 4176, 4177, 4180, 4181, 4182, 4415, 4592, 5713, 6270, 6271, 6274, 6275, 6276, 6446, 6588
Cadena de caracteres con S_CONV, 4708, 6716
Cadena de caracteres en Array con Strg_TO_Chars, 4714

Convertir número en cadena de caracteres y números con VAL_STRG, 4711
Convertir tiempos con T_CONV, 4691, 6683
Convertir una cadena de caracteres numéricos en un número con STRG_VAL, 4709
Copiar los caracteres en una cadena de caracteres numéricos con Chars_TO_Strg, 4716
De hexadecimal a cadena de caracteres con HTA, 4727, 6721
Una cadena de caracteres a hexadecimal, con ATH, 4724, 6719
Convertir BCD en entero, 4179, 6272
convertir direcciones
Instrucciones para, 5003
Convertir una CPU no especificada, 180
Convertir valor, 5936
Cookie, 954, 991
Coordenada del punto de referencia, 9786, 9864, 10952, 11053
Coordenadas geográficas, 1266
Coordinación, 8741
Coordinación de accesos, 8063
Copia de seguridad, 104
Borrar, 9253
Cambiar nombre, 9253
Datos del panel de operador, 9271
Panel de operador, 9270, 9286
Copia de seguridad HMI
Borrar, 9253
Cambiar nombre, 9253
Mostrar, 9252
Copiar
Adaptar el tamaño de la imagen, 9239
Área, 3453, 3484, 3766, 3797, 3951, 3977, 4361, 4406, 4558, 4583, 5704, 5927, 6066, 6436, 6578
Área sin interrupciones, 3459, 3487, 3773, 3800, 3958, 3980, 4366, 4409, 4564, 4586, 5708, 5931, 6070, 6440, 6582
Cadena de caracteres en Array con Strg_TO_Chars, 4714
carpeta definida por el usuario, 9238
Color, 9239
Componente de hardware, 526
Formato Excel, 9169
Fuente, 9239
Imagen, 8314, 9239
Indicador de avisos, 64
Objeto no válido, 9238
Objetos enlazados, 9241
Panel de operador, 58
Plantilla, 8320
principio, 9237
Tecla de función, 9239
Valor, 3436, 3749, 3940, 5702, 5925
Variable, 8470
Ventana de avisos, 64
Visor de avisos, 64
Copiar área, 3455, 3768, 3953, 4363, 4560
Corrección de velocidad
Easy Motion Control, 11221
Correo electrónico
Enviar, 689
Transmitir un mensaje de correo electrónico con TM_MAIL, 5588
Transmitir un mensaje de correo electrónico con TMAIL_C, 5286
Cortafuegos, 939
Administrar grupos de servicios, 780, 784
Crear grupos de servicios, 779, 783
Definir servicios ICMP, 778
Reglas de cortafuegos, 773
Cortocircuito a L+, 2124
Cortocircuito a masa, 2123
CoS, 1309
Cola de espera, 1309
COS, 3426, 3738, 4152, 4348, 4546, 5694, 5917, 6246, 6431, 6569
CoS (Class of Service), 1225
Coseno, 3426, 3738, 4152, 4348, 4546, 5694, 5917, 6246, 6431, 6569
COUNT, 7650
COUNTER, 2366
CountOfElements, 3483, 3796, 4405
CP
Acoplamiento punto a punto, 1068
Como esclavo I, 1421
con una dirección MPI propia, 915
CP 1242-7, 43, 1207
CP 1613, 33
CP 1623, 33
CP 343-2, 42
CP PC, 751
CP S7, 751
CP1, 9109
C-PLUG
formatear, 1298
CPM, 9109
CPU
Ajustar hora con WR_SYS_T, 4695, 6687
Arranque, 7422, 7426, 7455, 7458
Cambiar el estado operativo, 2187
Como esclavo I, 1422
Conexiones ocupadas y reservadas, 2237
Identificar con lista de estado del sistema, 6981

Insertar Signal Board, 976
Leer características con lista de estado del sistema, 6976
Leer el búfer de diagnóstico, 2182
Leer fecha y hora con RD_SYS_T, 4696, 6688
Mostrar estado actual de los LEDs, 2177
Nivel de llenado de todos los tipos de memoria, 2178, 2179
Parámetros, 1737
Propiedades, 1645
Recursos de conexión no ocupados, 2237
Registrarse para acceder a avisos, 2242
Relación entre variante de protocolo y longitud de datos, 7738
Selección del catálogo de hardware, 518
CPU 1516-3 PN/DP, 1942
CPU 1518-4 PN/DP, 1942
CPU S7
Memoria de trabajo, 1631
Área de operandos, 1631, 1723
Áreas de memoria, 1717
Memoria de carga, 1630
CPU sin especificar, 522
CPUs S7-400
Tipos de memoria, 1720
CRC, 1041, 1130
Crear, 8581
Comunicación directa, 1425
Conexión, 8765, 8769, 8794, 8843, 8898
Grupo, 8362
Grupo de avisos, 8514
Grupo de usuarios, 8607, 8631
Imagen, 8313
Librería global, 9143
matriz, 8496
Plantilla, 8319
receta, 8581
Registro de receta, 9335
Registro de receta en el panel de operador, 8594, 8595
Texto de ayuda, 8516, 8518
Usuarios, 8609, 8632, 9315, 9317
Usuarios en runtime, 8618
variable externa, 8465
variable interna, 8467
Crear la vista preliminar, 404
Crear tipo
Imagen, 9154
Script, 9153
Crear tipos nuevos
Estilo, 9155
Crear tiras rotulables, 409
Crear una copia de seguridad, 9313
Datos del panel de operador, 9270, 9286
Crear una ruta, 862
Crear una tabla de observación, 3119
CREATE_DB, 4994
Criterio de fin, 1084
carácter de fin, 1085
longitud fija de telegrama, 1087
Tiempo de retardo de caracteres transcurrido, 1084
CRP_IN
Instrucción, 7238
Parámetros de entrada, 7239
Parámetros de salida, 7239
CRP_OUT
Instrucción, 7240
Parámetros de entrada, 7241
Parámetros de salida, 7241
Cruce
Borrar, 2709
Cambiar posición, 2709
Definición, 2708
Insertar, 2708
CS1, 9109
CTD, 3354, 3665, 3926, 4319, 4526, 5649, 5872, 6055, 6408, 6551
CTRL_HSC, 5121
CTRL_PWM, 4961
CTS, 1018
CTU, 3351, 3662, 3923, 4316, 4523, 5647, 5870, 6053, 6406, 6549
CTUD, 3357, 3668, 3930, 4321, 4529, 5651, 5874, 6058, 6410, 6553
CU, 3368, 3679, 4118, 4119, 5661, 5884, 6212, 6213
Cuadrado, 3419, 3731, 4145, 4342, 5688, 5910, 6240, 6425, 6564
Cuadro de diálogo ", 2073
Cuadro de diálogo "Configuración del equipo de destino", 2072
Cuadro de diálogo de inicio de sesión
Configurar una protección de acceso, 8625
Cuadro vacío
Insertar elemento KOP, 2682
Insertar un elemento FUP, 2725
Curvas disparadas por bit, 8831, 8992
Curvas disparadas por tiempo, 8831, 8992
Cyclic interrupt, 1683
Cyclic interrupt (S7-1500), 1972

D

D_ACT_DP, 919, 4780
D_ACT_DP (activación y desactivación de dispositivos IO), 1560
Data Encryption Standard (DES), 818
Data Log, 982
  Abrir con DataLogOpen, 4980
  Abrir con DataLogTypedOpen, 4981
  Borrado de la Memory Card en el S7-1500, 38
  Borrar con DataLogDelete, 4989
  Cerrar con DataLogClose, 4987
  Crear con DataLogCreate, 4973
  Crear con DataLogNewFile, 4991
  Crear con DataLogTypedNewFile, 4992
  Escribir con DataLogWrite, 4985
  Vaciar con DataLogClear, 4984
DataLogClear, 4984
DataLogClose, 4987
DataLogCreate, 4973
DataLogDelete, 4989
DataLogNewFile, 4991
DataLogOpen, 4980
DataLogTypedNewFile, 4992
DataLogTypedOpen, 4981
DataLogWrite, 4985
DATE, 2344, 2402, 2452, 2478, 2502, 2519, 2534
DATE_AND_LTIME, 2347
DATE_AND_TIME, 2346
DATE_TO_, 2452, 2502, 2534
Datos de control, 9157
  Inicializar, 9156, 9158
Datos de diagnóstico, 1732
  ABS_DIAG, 7423
  ABS_DIAG_451, 7428
  Leer con GET_DIAG, 4956
Datos de encóder, 10933, 11034
  Datos en el DB de canal, 10934, 11034
  Desarrollo durante la lectura, 10933, 11034
  Requisitos, 10933, 11034
Datos de leva
  Escribir, 9752
  Leer, 9753
Datos de llamada, 7234
Datos de parámetros, 1732
Datos de PLC, 9157
Datos de posición, 10933, 11034
  Datos en el DB de canal, 10933, 11034
  Desarrollo durante la lectura, 10933, 11034
Datos de proyecto
  Cargar, 367
  cargar de un dispositivo, 370
  Cargar en una Memory Card, 368
  Compilar, 364
  Exportar aviso, 9173
  Exportar lista de textos, 9186
  Exportar receta, 9169
  Exportar variable, 9180
  Importar aviso, 9174
  Información acerca de la carga, 364
  Información acerca de la compilación, 363
  Módulo maestro IO-Link, 39
Datos de recetas
  Migración, 162
Datos de rendimiento, 2235
Datos de servicio
  Guardar, 2186
Datos de sistema
  Bloque de parámetros para, 8156
Datos de usuario, 1732
  Área, 1636, 1916
  Crear una copia de seguridad, 9313
  Direcciones, 1440
  Restablecer, 9313
  Slots, 1440
Datos de variables
  Estructura para la importación, 9183, 9188
Datos en el DB de canal
  Ajustar valor real, 10926, 11022
  Ajustar valor real al vuelo, 11022
  Aproximación en bucle, 10930, 11031
  Búsqueda del punto de referencia, 10917, 11011
  Cancelar preselección del valor real, 10927, 11023
  Captura de bordes, 11029
  Datos de encóder, 10934, 11034
  Datos de posición, 10933, 11034
  Datos máquina, 10902, 10996
  Decalaje de señal cero, 11024
  Definir punto de referencia, 10928, 11026
  Entrada de habilitación, 10933, 11033
  JOG, 10912, 11006
  Medición de longitud, 11029
  Modo incremental, 10924, 10925, 11019, 11021
  Señales de realimentación del diagnóstico, 10935, 11036
  Señales de realimentación para el posicionamiento, 11035
  Señales de respuesta del posicionamiento, 10934
  tablas de incrementos, 10904
  Tablas de incrementos, 10998

Datos en el DB de parámetros
Búsqueda del punto de referencia, 10918, 11012
Captura de bordes, 11029
Entrada de habilitación, 11033
Final de una operación de posicionamiento, 10911, 11005
Modo incremental, 10924, 11019
Datos estadísticos, 2238
Datos I&M: Leer con GET_IM_Data, 4936
Datos IPE
Actualizar, 9161, 9166
Datos máquina, 10901, 10994
Accionamiento, 11038
Activar, 9749, 10901, 10995
Cambiar, 10901, 10995
Datos en el DB de canal, 10902, 10996
Entrada, 10937
Entrada de habilitación, 9803, 9881
Escribir, 9749, 10901, 10995
Leer, 9751, 10902, 10996
Datos máquina del encóder
Datos en el DB de parámetros, 9796, 9874
Datos runtime
Migración, 162
Datos útiles
Área, 1734
DB, 2598, 2940
DB 101 a 104
Parámetros de entrada, 7521
Parámetros de entrada/salida, 7527
Parámetros de salida, 7525
DB 101 a 116
Parámetros de entrada, 7623
Parámetros de entrada/salida, 7628
Parámetros de salida, 7626
DB de canal
Contenido, 7429, 7462
Estructura, 7428, 7462
Tarea, 7428, 7461
DB de contaje
Configuración, 7394
Contenido, 7394, 7407
Tarea, 7393, 7407
DB de diagnóstico
Configuración, 10977, 11079
Contenido, 7472
Datos, 9821, 9901, 10977, 11079
Estructura, 7472, 9821, 9901
Tarea, 7471
DB de diagnóstico FM x51
Contenido, 7437
Estructura, 7437
Tarea, 7436
DB de eje
Índice, 7385
Tarea, 7385, 11257
DB de instancia
Crear, 3180, 3188
DB de parámetros
Contenido, 7469
Estructura, 7469
Tarea, 7469
DB de parámetros FM x51
Contenido, 7435
Estructura, 7435
Tarea, 7434
DB de Web Control, 962, 1000
DB E-Mail, 690
DB_ANY_TO_VARIANT, 3994, 4429
DCAT, 3568, 3883, 4034, 4488, 4651, 5792, 6013, 6123, 6510, 6644
DCD, 1018
DCP (Primary Setup Tool), 783
DEAD_T
Bloque de datos global, 7243
Instrucción, 7241
Parámetros de entrada, 7243
Parámetros de salida, 7243
DEADBAND
Instrucción, 7243
Parámetros de entrada, 7245
Parámetros de salida, 7245
Dead-Peer-Detection (DPD), 825
DEC, 3410, 3722, 4138, 6232
Decalaje de señal cero, 11023
Datos en el DB de canal, 11024
Datos utilizados en el DB de canal, 9761, 9838
Definición, 9760, 9838
Desarrollo, 9760, 9838, 11023
Deshacer, 9762, 9840
Efectos, 11024
en un eje lineal, 9761, 9839
en un eje rotativo, 9762, 9839
Requisitos, 11023
ZOFF_EN, 9760, 9838
Decalaje del punto de referencia, 10655
Deceleración de parada de emergencia, 10681, 10719, 10736
Decimales, 34, 3433, 3746, 4353, 4552, 9296, 9297
Declaración de disponibilidad para la comunicación
- establecimiento de conexión pasivo, 612
Declaración de variables
Actualizar la interfaz del bloque , 2656
Basados en un tipo de datos PLC, 2947

Borrar variable, 2590, 2665, 2959, 2974
Declarar ARRAY, 2651
Declarar STRUCT, 2652
Declarar tipo de datos PLC, 2653
Declarar variable, 2648, 2650, 2654
Finalidad de la declaración de variables, 2642
Importar y exportar variables, 2666, 2960
Insertar una fila al final de la tabla, 2664, 2958, 2974
Insertar una fila en la tabla, 2588, 2664, 2958, 2974
Interfaz del bloque, 2642
Mostrar y ocultar columnas , 2591, 2666, 2960, 2976
Multiinstancia, 2655
Ordenar filas, 2590
Palabras clave reservadas, 2281
Propiedades de las variables, 2658, 2661, 2953, 2955, 2957
Rellenar celdas automáticamente, 2591, 2665, 2960, 2975
Remanencia, 2660
Superponer variables, 2654
Tipos de datos válidos, 2646, 2647
DECO, 3545, 3859, 4021, 4467, 4626, 5746, 5969, 6093, 6479, 6612
DecreaseFocusedValue, 8704
DecreaseTag, 8704
Decrementar, 3410, 3722, 4138, 6232
Definición
Alarma de diagnóstico, 10331
Definir
Filtros para el plano de ocupación, 3060
Objeto de referencia, 8336
Definir hotkey, 8449
Definir punto de referencia, 9762, 9840, 10928, 11025
Datos en el DB de canal, 10928, 11026
Datos utilizados en el DB de canal, 9763, 9841
Definición, 9762, 9840
Desarrollo, 9763, 9841, 10928, 11026
Efectos, 9763, 9841, 10928, 11026
Particularidades del encóder absoluto, 9764, 9842
Requisitos, 9763, 9840, 10928, 11025
Sincronización, 10928, 11025
DefinirVariable, 8697
DELETE, 4732
DELETE_DB, 5002
DEMUX, 3553, 3867, 4474, 4632
Denominación
TS Adapter, 11395
TS Adapter, 11379, 11387
Dependencia del dispositivo
Imagen, 8309
Dependencia del equipo
S7 1200 V1, 8886
S7 1200 V2, 8886
S7 1200 V3, 8886
S7 1200 V4, 8886
SIMATIC S7 1500 V1.0, 8834
Dependencias de derechos, 770
Depurar el proyecto para la migración, 131
Derechos de configuración, 768
Derechos de módulos, 768
DES, 812, 818
Desactivación, 10909, 11003
JOG, 10913, 11007
Modo incremental, 10925, 11020
Desactivar, 8718
Área de periferia, 5779, 6000, 6105, 6498, 6631
Área por bits, 5777, 5999, 6104, 6497, 6630
Área por bytes, 5781, 6002, 6107, 6500, 6633
Idioma del proyecto, 9203
Mapa de bits, 3291, 3600, 5777, 5999, 6104, 6497, 6630
Operando, 3288, 3294, 3597, 3603, 4082, 5606, 5610, 5827, 5831, 6177
RLO, 4086, 6182
DesactivarBit, 8683
DesactivarBitEnVariable, 8684
Desaparecer, 8505
Desaparecido, 8720
Desarchivación de proyectos, 359
Desarrollo
Ajustar valor real, 10926, 11021
Ajustar valor real al vuelo, 11021
Aproximación en bucle, 10930, 11030
Captura de bordes, 11028
Decalaje de señal cero , 11023
Definir punto de referencia, 10928, 11026
JOG, 10911, 11005
Leer datos de posición, 10933, 11034
Leer datos del encóder, 10933, 11034
Medición de longitud, 11028
Modo incremental con número de incremento 1-100, 10921, 11015
Modo incremental con número de incremento 252, 11016
Modo incremental con número de incremento 254, 10922, 11017
Modo incremental con número de incremento 255, 10922, 11017

Señales de realimentación del diagnóstico, 10935, 11036
Señales de realimentación para el posicionamiento, 11035
Señales de respuesta del posicionamiento, 10934
desbloquear
Usuarios, 8621
Desbordamiento, 2545, 2559, 8724
Desbordamiento del búfer de avisos, 8722
Descarga, (Véase Carga en dispositivo)
Descarga de firmware vía web, 2021
Descodificar, 3545, 3859, 4021, 4467, 4626, 5746, 5969, 6093, 6479, 6612
Descripción
ABS_CTRL, 7420
ABS_CTRL_451, 7424
ABS_DIAG, 7422
ABS_DIAG_451, 7426
ABS_INIT, 7419
CAM_CTRL, 7453
CAM_CTRL_452, 7457
CAM_DIAG, 7455
CAM_DIAG_452, 7458
CAM_INIT, 7453
CAM_MSRM_452, 7460
CNT_CTL1, 7390
CNT_CTL2, 7393
CNT_CTRL, 7388
CNT2_CTR, 7405
CNT2RDPN, 7405
CNT2WRPN, 7404
DIAG_INF, 7390
DIAG_RD, 7406
EncoderAbsSensorDP, 7356
EncoderCPU314C, 7363
EncoderET200S1Count, 7359
EncoderET200S1SSI, 7354
EncoderFM350, 7361
EncoderFM450, 7352
EncoderIM174, 7365
EncoderIM178, 7350
EncoderSINAMICS, 7368
EncoderSM338, 7358
EncoderUniversal, 7371
MC_Control, 7339
MC_GearIn, 7342
MC_Home, 7333
MC_Init, 7324
MC_MoveAbsolute, 7325
MC_MoveJog, 7330
MC_MoveRelative, 7328
MC_Simulation, 7341
MC_StopMotion, 7337
OutputCPU314C, 7376
OutputET200S2AO, 7375
OutputIM174, 7379
OutputIM178, 7373
OutputMM4_DP, 7378
OutputSINAMICS, 7381
OutputSM332, 7376
OutputSM432, 7374
OutputUniversal, 7377
Descripción de la conexión
Bloque de datos , 711, 713, 714
Estructura, 711, 714, 715
Modificar valores de parámetros, 717
Deserializar, 3442, 3755, 3943, 4355, 4553
Deserialize, 3442, 3755, 3943, 4355, 4553
Desescalar, 3508, 3821, 3999, 4433, 4607, 5723, 5946, 6080, 6454, 6597
Desfase (S7-1500), 1972
Deshacer
Decalaje de señal cero , 11025
Grupo, 8363
Deshacer acciones
Deshacer la última acción, 416
Dshacer varias acciones, 416
Principios básicos para deshacer acciones, 413
Deshacer la conexión online, 2209
Deshacer la conexión remota, 11401
Deshacer la función
Cancelar preselección del valor real, 9759, 9837
Desinstalación de claves de licencia, 90
Desmultiplexar, 3553, 3867, 4474, 4632
Desplazamiento
Derecha, 6346, 6348, 6352, 6356
Izquierda, 6350, 6354
Desplazar
Componentes de hardware, 528
Derecha, 3556, 3871, 4256, 4258, 4262, 4266, 4477, 4635, 5750, 5973, 6485, 6616
Imagen, 8314
Izquierda, 3558, 3873, 4260, 4264, 4479, 4637, 5752, 5975, 6487, 6618
Plantilla, 8320
Desplazar la imagen
Comandos de teclado, 508
Navegación general, 492, 495, 497
Desplazar la vista
Navegación general, 8745, 8752, 8754
Destino ficticio, 10930, 11031
Posición, 10931, 11031
Destrucción de claves de licencia, 90

DETACH, 4885
Detalles de conexión, 2217
Detalles sobre el establecimiento de la conexión, 649, 655, 662
Detección de fin de un telegrama de recepción, 1119
Detección de hardware, 522
Detección de topología, 1513, 8805, 8856, 8909
Detector de proximidad, 9744, 9747, 10655
Detener
    Easy Motion Control, 11221
Determinar automáticamente la vía de conexión, 622
Determinar el estado de diagnóstico, 2180
Determinar la hora, 2189
Device Tool, 1608, 1610
DeviceStates, 4948
DHCP
    Cliente, 1277
    Configuración de servidores, 863
DIAG_INF, 7390
    Descripción, 7390
    Funcionamiento, 7390
    Llamada, 7390
    Parámetros, 7390
DIAG_RD, 7406
    Descripción, 7406
    Funcionamiento, 7406
    Llamada, 7406
    Parámetros, 7407
Diagnostic error interrupt, 1687
Diagnostic error interrupt (S7-1500), 1979
Diagnóstico, 828, 1054, 1197, 2114
    Escribir un evento de diagnóstico en el búfer de diagnóstico con WR_USMSG, 6947
    Alarmas tecnológicas, 10782
    Componentes de PROFINET IO, 2241
    Concepto de diagnóstico Motion Control, 10781
    Conexión, 8748
    Conexiones con C_DIAG, 7027
    Datos de diagnóstico, 6912
    Datos de diagnóstico de módulo, 7022
    Datos de diagnóstico, específicos de canal, 6914
    Dispositivo, 8748
    Easy Motion Control, 11222, 11269
    Eje de posicionamiento, 10789
    Eje de velocidad de giro, 10786
    Encóder externo, 10794
    Errores en las instrucciones de Motion Control, 10785
    Esclavo DP, leer datos de diagnóstico con DPNRM_DG, 4806, 6803
    Específico de canal, 2239
    High_Speed_Counter, 9426
    Información de conexión, 8748
    Información del dispositivo, 8748
    Módulo tecnológico, 9455
    Módulo, lista de estado del sistema, 7020
    Motion Control, 10781
    PROFINET, 1525
    Registro de diagnóstico, 7021
Diagnóstico de canal, 2240
    Datos de diagnóstico, 6914
Diagnóstico de comunicación, 2240
Diagnóstico de conexiones
    Detalladamente, 2215
    Vista general, 2213
Diagnóstico de hardware, 2158
Diagnóstico de segmento
    Iconos, 2211
    Visualización gráfica, 2211
    Visualización mediante texto, 2211
Diagnóstico de sistema, 8563
    Visor de diagnóstico del sistema, 8563
    Vista de dispositivos, 8564
    Vista del búfer de diagnóstico, 8564
    Vista detallada, 8564
Diagnóstico del I-Device, 1554
Diagnóstico del puerto
    Diagnóstico SFP, 1304
Diagnóstico del sistema
    Bloques creados, 2221
    DB de estado de diagnóstico, 2224
    Introducción, 2218, 2230
Diagnóstico específico de canal, 2239
Diagnóstico online, 830
Diagnóstico online de seguridad de CPs S7, 45
Diagnóstico SFP, 1304
Diagnóstico web, 2021, 2022
Diagrama de funciones, 2712
DIF
    Parámetros de entrada, 7247
    Parámetros de salida, 7247
Diferenciador, 7245
Diferente, 3394, 3706
DIGITAL, 7640
Dinamizar
    Apariencia de un objeto, 8395
    Color de un objeto, 8395
    Flecha verde en vista general, 8394
    Movimiento de un objeto, 8396
    Movimiento directo, 8398
    Objeto, 8391
    Parpadeo, 8395

DINT, 2333, 2386, 2472, 2516
DINT , 2430, 2495, 2530
DINT_TO_, 2430, 2495, 2530
Dirección, (Véase Dirección inicial de un módulo)
    Mitsubishi, 9052
    Omron Host Link, 9110
    Punteros de área, 8782
Dirección de canal, (Véase Dirección inicial de un módulo)
Dirección de diagnóstico, 915
    Asignar, 1435
    Configurar, 1418
    Esclavo I, 1417
Dirección de dispositivo, 915, 971
Dirección de periferia, 915, 971, 1636, 1916
Dirección de subred IP, 1942
Dirección del CP, 8190
Dirección Gigabit, 763
Dirección I, 915, 971
Dirección inicial, 914, 915, 971, 1635, 1917
Dirección inicial de módulo, (Véase Dirección inicial de un módulo)
Dirección inicial de un módulo, 1917
Dirección IP, 33, 551, 553, 1491, (Véase Display)
Dirección IP del router, 862
Dirección IP WAN
    Definir, 826
Dirección lógica
    Determinar de un módulo con GADR_LGC, 5016, 7073
    Determinar de un módulo con RD_LGADR, 5014, 7072
    Determinar slot correspondiente con LGC_GADR, 5017, 7075
    Determinar slot correspondiente, con LOG_GEO, 5013, 7070
Dirección MPI
    S7 300, 8925
    S7 400, 8926
Dirección PROFINET, 763
Dirección Q, 915, 971
Direccionamiento, 1636, 1916
    Allen-Bradley, 9011, 9029
    Allen-Bradley Ethernet IP, 9006
    EtherNet/IP, 9011, 9029
    general, 914, 915, 971
    modificar, 916, 973
    MPI, 8925
    Multiplexar, 8483
Direccionamiento absoluto
    de una variable, 8461, 8776
Direccionamiento de la periferia, 1735
Direccionamiento de módulos, 1635, 1734
Direccionamiento de operandos, 2294, 2295, 2298, 2302, 2316
Direccionamiento indirecto, 252, 2305, 2306, 2307, 2309, 2311, 2314, 2315, 2316, 2317, 2318, 2319
Direccionamiento IP a través de DHCP, 612
Direccionamiento simbólico, 253
    de una variable, 8462, 8777
Direccionar
    Direccionar variable indirectamente, 8483
    Direccionar variables indirectamente, 8483
Direccionar indirectamente , 8483
Direccionar operandos, 2293, 2297, 2301, 2305, 2306, 2307, 2309, 2311, 2312, 2314, 2315, 2317, 2318, 2319
Direcciones, 2080
    Alarma con direcciones, 1414
    Alarmas en el esclavo I, 1419
    asignar, 928, 974
    Comprimir, 1407, 1477
    Datos de usuario con DPV1, 1440
    Descomprimir, 1408, 1478
    Determinar de un módulo con GEO_LOG, 5012, 7068
    Esclavo DP, 1442
    Informaciones de sistema, 1442
    Interfaces DP, 1440
    Leer dirección de estación con GetStationInfo, 4944
Direcciones IP
    PROFINET, 1517
Direcciones IPv4, 790
Direcciones IPv6, 791
DIS_AIRT, 4915
DIS_IRT, 4912
Discriminador, 1753
Diseñar borde, 8343
Diseñar fondo, 8343
Diseño de página
    en textos de aviso, 1115
Diseños de ventana
    Borrar diseño de ventana, 312
    Cambiar orden, 312
Disparador, 8730
    Disparo por eventos, 8730, 8733
Disparar
    Alarma de diagnóstico, 10331
Disparo por eventos, 8730, 8733
Display, 1939
Display (CPU), 1960
Display de 7 segmentos, 3584, 3899, 4503, 5815, 6040, 6153, 6531, 6668

Display de la CPU, (Véase Display)
Display de siete segmentos, 3584, 3899, 4503, 5815, 6040, 6153, 6531, 6668
Disponer
    Objeto en imagen, 8328, 8339
Disponibilidad
    Objeto para Basic Panel, 8423
Disposición de la secuencia de bytes, 3466, 3779, 4372, 4570, 6575
Disposición de los módulos, 512
Disposición del encóder, 10652
Dispositivo
    Agregar a una configuración de hardware, 520
    borrar, 525
    cambiar nombre, 746
    copiar, 58, 526
    desplazar, 528
    Dispositivos defectuosos, 2160
    Insertar, 8736
Dispositivo comodín, 1618
Dispositivo de campo, 1433, 1486, 1516
Dispositivo IO
    Conectar en red, 1505
    Tiempo de actualización, 1508
    Tiempo de supervisión de respuesta, 1508
Dispositivo IO inteligente, 1546
Dispositivo normalizado, 1515
Dispositivo PROFIenergy, 8116
Dispositivo PROFINET IO
    Asignación de nombre en el cuadro de diálogo, 2200
    Asignación de nombre en la vista Online y diagnóstico, 2198
    Asignar nombre, 2196
Dispositivos
    Conectar, 8740
    Conectar en red, 8764
    Conexión a red, 8739
Dispositivos accesibles, 1513, 8805, 8856, 8909
Dispositivos defectuosos, 2160
Dispositivos PROFINET IO
    Escribir datos coherentes con DPWR_DAT, 4793, 6778
    Escribir todas las salidas con SETIO, 4750, 6741
    Escribir una parte de las salidas con SETIO_PART, 4752, 6743
    Leer datos coherentes con DPRD_DAT, 4791, 6776
    Leer una parte de las entradas con GETIO_PART, 4751, 6742
Dispositivos y redes, 8739, 8764
    Conexiones, 8765
    Conexiones HMI, 8765
Distancia de cambio de velocidad, 10889, 10942, 11043
Distancia de cambio de velocidad negativa, 10943, 11044
Distancia de cambio de velocidad positiva, 10943, 11044
Distancia de desconexión, 10889, 10943, 11044
Distancia de desconexión negativa, 10944, 11045
Distancia de desconexión positiva, 10944, 11045
Distancia mínima entre bordes, 9793, 9871, 11057
Distancia residual, 10925, 11020
Distribuidor de saltos, 3516, 3829
Distribuir
    Objetos uniformemente, 8338
DIV, 3405, 3716, 5676, 5899
Dividir, 3405, 3716, 4129, 4134, 4143, 5676, 5899, 6223, 6228, 6238
DLE, 1034
DM 370 Dummy, 899
DMSK_FLT, 4909
DO, 4440, 4442, 6463
Docking Unit, 1560, 1562
Documentación del hardware, 11429
Dominio MRP, 1571, 1573, 2175
Dominio predeterminado, 1530
Dominio Sync, 1513, 1530, 2174, 8805, 8856, 8909
Dosificación, 9484
DP, 8864, 8917
DP_CTRL, 8226
DP_DIAG, 8217
DP_RECV, 8211
DP_SEND, 8208
DP_TOPOL, 4809
DPNRM_DG, 4806
DPRD_DAT, 919, 4791
DPSYC_FR, 919, 4800
DPV0/DPV1, 1433
DPV1, 1748
    Configurar ET 200S, 1413, 1483
    Evaluación de informaciones de alarma, 1438
    Evaluación de informaciones de diagnóstico, 1438
    Modelo de slot, 1440
    OBs para alarmas, 1435
DPWR_DAT, 919, 4793
Driver ASCII, 1081
    Búfer de recepción, 1089
    Enviar datos, 1082
    Recibir datos, 1083, 1118
    Señales cualificadoras RS232C, 1090, 1115
Driver de comunicación, 8741

Driver de entrada
Inicialización, 7349
Llamada, 7349
Modo de simulación, 7350
Tarea, 7348
Tratamiento de errores, 7349
Driver de impresora
Salida del texto de aviso, 1117
Textos de aviso, 1115
Driver de salida
Inicialización, 7372
Llamada, 7373
Modo de simulación, 7373
Tarea, 7372
Tratamiento de errores, 7373
Drivers de comunicación
Allen-Bradley, 8998
Basic Panel, 8788
Mitsubishi, 9040
Modicon Modbus, 9069
Omron, 9099
Drivers de impresora
Cadena de formato, 1116
Ejemplos, 1116
Handshake de software, 1117
Procedimiento Handshake, 1117
Señal BUSY, 1118
Variables, 1115
Drivers de terceros
Comunicación, 8997
Particularidades, 8998
DRUM, 3564, 3879, 4026, 4485, 4643, 5784, 6005, 6111, 6503, 6636
DRUM_X, 5788, 6009, 6119, 6506, 6640
DSC (Dynamic Servo Control), 10670
DSCP, 1309
DSR, 1018
DST
Horario de verano, 1282, 1283
DT, 2346, 2398, 2447, 2520, 2536
DT_TO_, 2447, 2536
DTB, 4181, 6275
DTL, 2348, 2401, 2450, 2477, 2504
DTL_TO_, 2450, 2504
DTR, 1018, 4182, 6276
Duración de certificados, 815
Duración del impulso, 9530, 9598
Ajuste predeterminado, 9530, 9599
Rango de valores, 9530, 9598
Duración del período (CPU 31xC), 1770
DWORD, 2327, 2379, 2416, 2467, 2485, 2514, 2526
DWORD_TO_, 2416, 2485, 2526
Dynamic Servo Control (DSC), 10670

E

Easy Motion Control
Anulación de un desplazamiento, 11221, 11255
Búsqueda del punto de referencia, 11220
Campo de bits para inicialización, 11261
Configuraciones hardware, 11213
Corrección de velocidad, 11221
DB de eje, 11257
Destino, 11216
Detener, 11221
Diagnóstico, 11222, 11269
Diseño del programa de usuario, 11248
Distancia residual, 11216
Eje lineal, 11217
Eje parado, 11266
Eje rotativo, 11218
El eje se desplaza sin petición de desplazamiento, 11267
El eje vibra, 11267
Entorno de llamada, 11249
Error con parada brusca, 11274
Error con parada suave, 11271
Error de parametrización, 11270, 11276
Error de seguimiento, 11216, 11269
FB multiinstancia, 11261
Inicio de una petición de desplazamiento, 11254
Interacción de las instrucciones, 11248
Interruptor de PARADA DE EMERGENCIA, 11262
JOG, 11220
Medición de recorrido, 11265
Medición de velocidad, 11264
Orden de llamada de las instrucciones, 11250
Parámetros comunes para todas las instrucciones, 11253
Parámetros de eje, 11259
Posicionamiento, 11215
Posicionamiento absoluto, 11221
Posicionamiento del punto de referencia, 11220
Posicionamiento relativo, 11221
Reacciones a errores, 11222
Reajuste del encóder absoluto, 11220
Reductor, 11221
Seguimiento, 11221
Simulación, 11222
Sincronizar, 11220
Test de cableado, 11263
Valores actuales, 11259
Velocidad del eje, 11267

Zona de desplazamiento, 11217
Zona de destino, 11217
Zona de trabajo, 11217
Zona de velocidad cero, 11217
Zonas de vigilancia, 11217
EB, 2571
EC31-HMI/RTX
Visualización, 1205
EC31-RTX, 42
ED, 2571
Edición
Lista de funciones, 8640
EditAlarm, 8654
Editar
Acceso directo a una carpeta, 8357
Aviso, 9327
Grupo, 8362
Objeto dentro de un grupo, 8365
Registro de receta, 9335
Editar conexión en red
Copiar subred, 542
Copiar subredes y dispositivos, 543
Editar conexiones en red
Eliminar conexión de red, 541
Editar la versión de un tipo en test, 467
EditarAviso, 8654
Editor
Conexión, 8769
Conexiones, 8944, 8946, 8948
Dispositivos y redes, 8767
Gráficos, 9213
Editor de avisos
Estructura, 3177
Editor de comparación
Definir acciones, 386
Filtrar una vista, 383
Modificar la vista, 388
Mostrar y ocultar columnas, 382
Sinopsis, 376
Editor de documentación
Estructura, 399
Editor de hardware
Catálogo de hardware, 504, 8757
Componentes, 490, 8742
Función, 489, 8742
Ventana de inspección, 502, 8755
Editor de hardware y redes
Vista de dispositivos, 494, 8751
Vista de redes, 491, 8744
Vista topológica, 496, 8753
Editor de programación
Ampliar la superficie de la ventana de programación, 2627
Barra de herramientas, 2621
Comandos de teclado, 2623
Configuración general, 279, 281, 282, 2640
Estructura, 2620
Favoritos, 2621
Función, 2619
Interfaz del bloque, 2621
Task Card Instrucciones, 2621
Task Card Test, 2621
Ventana de programación, 2621
Editor de redes
Catálogo de hardware, 504, 8757
Componentes, 490, 8742
Función, 489, 8742
Ventana de inspección, 502, 8755
Editor topológico, 553
Efecto de Ti
Modo isócrono, 1537
Efecto de To
Modo isócrono, 1538
Efectos
Ajustar valor real, 10927, 11022
Búsqueda del punto de referencia, 10918, 11012
Decalaje de señal cero , 11024
Definir punto de referencia, 10928, 11026
Eje
Coordenada del punto de referencia, 9786, 9864, 10952, 11053
Distancia mínima entre bordes, 9793, 9871, 11057
Final de carrera por software, 9784, 9862, 10949, 11050
Final del eje rotativo, 10950, 11051
Histéresis, 9790, 9868
Reajuste del encóder absoluto, 9787, 9865, 10952, 11053
Tipo de búsqueda del punto de referencia, 10951, 11052
Tipo de eje, 10948, 11049
Tipos de eje, 9783, 9861
Velocidad de inicio en la búsqueda del punto de referencia, 10956, 11057
Velocidad de simulación, 9792, 9870
Eje de posicionamiento
Configuración, 10697
Variables, 10798
Eje de velocidad de giro
Configuración, 10729
Variables, 10819

Eje lineal
Easy Motion Control, 11217
Eje lineal (CPU 31xC), 1781, 1788
Eje parado
Easy Motion Control, 11266
Eje rotativo
Easy Motion Control, 11218
Eje rotativo (CPU 31xC), 1781, 1789
Ejecución cíclica del programa
Posibilidades de interrupción, 1675
Programar, 1675
Ejecución del programa
Cíclica, 1675
Ejecutar, 8716
Ejemplo
Aplicación de categorías, 8506
Aviso de sistema, 8503
cálculo de una ecuación, 3247, 3257
Cálculo de una ecuación, 3230, 3266
Calentamiento de una caldera, 3231, 3267
Cambiar y visualizar el modo de operación, 8641
Control de una cinta transportadora, 3207, 3224, 3236, 3241, 3260, 3272
Controlar la temperatura ambiente, 3213, 3222, 3248, 3258
Detección del nivel de llenado de una zona de almacén, 3210, 3218, 3227, 3238, 3244, 3253, 3263, 3274
Detección del sentido de marcha de una cinta transportadora, 3209, 3217, 3226, 3237, 3243, 3252, 3262, 3273
Especificar un valor sustitutivo, 1884
para la entrada de valores de forzado permanente en la tabla de forzado permanente, 3151
para la vista de entradas y salidas en el plano de ocupación, 3055
para la vista de marcas en el plano de ocupación, 3055
Valores de forzado de la tabla de observación, 3124
Ejemplo/
Aviso de bit, 8504
Ejemplos
Control de una cinta transportadora, 3216, 3251
Ejes de módulo, 10645
El eje se desplaza sin petición de desplazamiento
Easy Motion Control, 11267
El TS Adapter no logra conectarse con la estación remota, 11423
Elaboración de manuales, 336
Elaboración individualizada de documentación, 336
Elemento de control
Visor de avisos, 8440
Visor de avisos simple, 8536, 9324
Visor de curvas, 9308
Elemento de manejo
Visor de avisos, 8442
Visor de curvas, 8436
Elemento de matriz
Nombre, 68
Ubicación de la variable HMI, 68
Elemento de receta
crear, 8581
Elemento FUP
Borrar, 2744
Copiar, 2738
Cortar, 2739
Insertar, 2724, 2725
Insertar desde el portapapeles, 2740
Insertar operando , 2745
Reemplazar, 2740
Reglas de inserción, 2722
Seleccionar, 2738
Elemento KOP
Borrar, 2700
Copiar, 2695
Cortar, 2695
Insertar, 2681
Insertar desde el portapapeles, 2696
Insertar operando, 2701
Reemplazar, 2697
Reglas de inserción, 2678
Seleccionar, 2694
Elementos de matriz, 8494
Elevar a potencia, 3435, 3747
Eliminar
Acceso directo a una carpeta, 8358
Formato en texto de aviso, 8520
Objeto del grupo, 8364
Elipse
Radio horizontal, 8430
Radio vertical, 8430
Elipse , 8429
ELSE, 4436, 4438, 6456, 6458
ELSIF, 4436, 6456
Embedded Controller, 1206
Cargar:descarga, 1206
EN_AIRT, 4915
EN_IRT, 4914
Encabezado de la petición, 683, 684, 685
Encabezado de petición, 8053, 8195
Encabezado de telegrama
Estructura del telegrama de orden RK512, 1104

ENCO, 3547, 3860, 4022, 4469, 4627, 5747, 5970, 6095, 6480, 6613
Encóder
conectable, 9492
incremental, 9492
incremental de posición, 9492
Incrementos por vuelta, 10958, 11060
Incrementos por vuelta del encóder, 9795, 9873
Longitud de telegrama, 9801, 9879, 10962, 11063
Número de vueltas del encóder, 9800, 9878, 10961, 11062
Paridad, 10963, 11064
recorrido por vuelta del encóder, 9794, 9872
Recorrido por vuelta del encóder, 10958, 11059
Seleccionar tipo de señal, 9508, 9561, 9589
Sentido de contaje, 9799, 9877, 10957, 11058
Sentido del detector de proximidad, 9802, 9880
Tipo de señal, 9793, 9871, 10956, 11057
Velocidad de transferencia, 9801, 9879, 10962, 11063
Vigilancia, 9799, 9877, 10963, 11064
Encóder absoluto, 9362, 9744, 10892
Escuchar, 9748
Evaluación de impulsos, 9747, 10894
Incrementos por vuelta del encóder, 10958, 11060
Tiempo monoestable, 9748, 10895
Tiempos de ejecución de telegramas, 9748, 10895
Tiempos de respuesta, 9748, 10894
Encóder absoluto SSI, 9444, 9445
Encóder externo
Configuración, 10737
Variables, 10830
Encóder incremental, 9744, 10891
Formas de señal, 9745, 10893
Impulso, 9746, 10893
Incremento, 9746, 10893
Incrementos por vuelta del encóder, 10958, 11060
Tiempos de respuesta, 9746, 10894
EncoderAbsSensorDP, 7356
Descripción, 7356
Llamada, 7357
Parámetro, 7357
Sincronización, 7357
Tratamiento de errores, 7357
EncoderCPU314C, 7363
Descripción, 7363
Llamada, 7364
Parámetro, 7364
Sincronización, 7364
Tratamiento de errores, 7365
EncoderET200S1Count, 7359
Descripción, 7359
Llamada, 7360
Parámetro, 7360
Sincronización, 7360
Tratamiento de errores, 7361
EncoderET200S1SSI, 7354
Descripción, 7354
Llamada, 7355
Parámetro, 7355
Sincronización, 7355
Tratamiento de errores, 7356
EncoderFM350, 7361
Descripción, 7361
Llamada, 7362
Parámetro, 7362
Sincronización, 7362
Tratamiento de errores, 7363
EncoderFM450, 7352
Descripción, 7352
Llamada, 7352
Parámetro, 7353
Sincronización, 7353
Tratamiento de errores, 7354
EncoderIM174, 7365
Acuse de errores, 7368
Descripción, 7365
Llamada, 7366
Parametrización del IM 174 (configuración hardware), 7365
Parámetro, 7367
Sincronización, 7367
Tratamiento de errores, 7368
EncoderIM178, 7350
Acuse de errores, 7352
Descripción, 7350
Llamada, 7350
Parametrización del IM 178-4 (configuración hardware), 7350
Parámetro, 7351
Sincronización, 7351
Tratamiento de errores, 7352
Encóders simétricos, 9492
EncoderSINAMICS, 7368
Acuse de errores, 7371
Descripción, 7368
Llamada, 7369
Parametrización del SINAMICS (configuración hardware), 7368
Parámetro, 7369

Sincronización, 7370
Tratamiento de errores, 7371
EncoderSM338, 7358
Descripción, 7358
Llamada, 7358
Parámetro, 7358
Sincronización, 7359
Tratamiento de errores, 7359
EncoderUniversal, 7371
Descripción, 7371
Funcionamiento, 7371
Llamada, 7372
Parámetro, 7372
Sincronización, 7372
Tratamiento de errores, 7372
END_CASE, 4438, 6458
END_FOR, 4440
END_IF, 4436, 6456
END_REPEAT, 4443, 6464
END_WHILE, 4442, 6463
Endian, 4394, 4396, 4398, 4400
ENDIS_PW, 3521, 3834, 4002, 4450, 4610
Engineering System
Características, 9339
Enlace de correo electrónico
Bloque de parámetros para, 8164
Enlace FTP
Bloque de parámetros para, 8165
Enlace ISO-on-TCP
Bloque de parámetros para, 8163
Enlace TCP
Bloque de parámetros para, 8161
Enlace UDP
Bloque de parámetros para, 8162
Enlazar carpeta
Gráfico externo, 8357
Enlazar variables HMI, 183
ENO, 256
Enrutamiento
Tabla de enrutamiento, 1257
ENSET_DN, 9514, 9594
ENSET_UP, 9514, 9594
ENT, 6367
Entero, 3499, 3812, 4193, 4421, 4598, 5719, 5942, 6286, 6450, 6593
16 bits, 2331, 2332
32 bits, 2333, 2334
64 bits, 2335, 2336
8 bits, 2330, 2331
Enteros, 2551
Entidad emisora, 760
Entidades emisoras de certificados de origen, 761
Entorno de llamada
Ajuste para bloques, 3088
Ajuste para puntos de parada, 3091
Principios básicos, 3087
Entorno TIA, 11196
Entrada
Datos máquina, 10937
Distancia de cambio de velocidad, 10942, 11043
Distancia de desconexión, 10943, 11044
Insertar, 2697, 3594, 5824
Modo de control 1, 10938, 11039
Modo de control 2, 10939, 11040
Modo de control 3, 10940, 11041
Modo de control 4, 10941, 11042
Quitar, 2698, 2742
Tiempo de vigilancia, 10946, 11047
Velocidad de parada, 10947, 11048
Zona de destino, 10945, 11046
Zona de parada, 10947, 11048
Entrada (I), 1631
Entrada de contador, 1655
Entrada de habilitación, 10915, 10932, 11009, 11033
Datos en el DB de canal, 10933, 11033
Datos en el DB de parámetros, 11033
evaluar, 10932, 11033
no evaluar, 10932, 11033
Entrada de periferia, 1631
Entrada digital
funciones, 9417
Funciones, 9441, 9447, 9448
Entrada finalizada, 8720
Entrada oculta, 8429
Entradas digitales
Filtro de entrada, 9512, 9593
Enumeraciones
Servidor web, 958, 995
Enviar datos
con TSEND, 7753
Driver ASCII, 1082, 1117
Fuera de la estación S7 propia con X_SEND, 8268
Procedimiento 3964(R), 1096
RK512, 1106
Enviar SMS, 5372
EQ_ElemType, 3390, 3702
EQ_Type, 3388, 3700
Equidistancia, 1446
Activación en la red PROFIBUS, 1455
Composición de los ciclos de bus, 1444
Definición, 1444
Requisitos, 1445

Tiempo para ciclo DP equidistante, 1445
Equilibrio de potencia, 2131
ERR_MON
Instrucción, 7247
Parámetros de entrada, 7249
Parámetros de salida, 7249
Error
externo, 10279, 10330, 10364
interno, 10279, 10330, 10364
Error asíncrono
Configuración, 1744
Error con parada brusca
Easy Motion Control, 11274
Error con parada suave
Easy Motion Control, 11271
Error de acceso a periferia, 1634, 1725
Error de acceso a periferia (PZF), 1726
Error de datos, 9550, 9611
Error de operador, 9612
Error de parametrización
Easy Motion Control, 11270, 11276
Error de seguimiento, 10673
Error de sintaxis
Buscar errores, 2930
Principios básicos, 2929
Error de telegrama, 9799, 9877, 10963, 11064
Error en canal de referencia, 2133
Errores de operador, 9552
Errores de parámetros PtP comunes, 5398, 7789
Errores de señales de vida, 11205
Errores de señales de vida tolerables, 11205
Errores en las instrucciones de Motion Control, 10785
Errores externos, 10279, 10330, 10364
Errores internos, 10279, 10330, 10364
Errores síncronos
OB 121 (S7-300, S7-400), 1880
OB 122 (S7-300, S7-400), 1883
ErrorID, 5130
Lista de ErrorIDs, 10841
ES ARRAY, 4334
EscalaLineal, 8672
Escalar, 3500, 3506, 3813, 3819, 3987, 3996, 4422, 4431, 4599, 4604, 5721, 5944, 6077, 6452, 6594
Escalar variable de forma lineal, 8480, 8481
esclavo, 1125
Esclavo DP, 897
Ajustar el modo isócrono, 1457
Asignar sistema maestro DP, 1400, 1472
Catálogo de hardware, 1389, 1463
Como esclavo I, 1423
Con preprocesamiento, (Véase Esclavo I), (Véase Esclavo I)
Conectar en red, 1401, 1466
Configurar, 1387, 1390, 1400, 1464, 1473
Desactivar y activar con D_ACT_DP, 4780, 6764
Desconectar del sistema maestro DP, 1400, 1472
Direcciones, 1442
Grupos SYNC/FREEZE, 1403
Insertar en el sistema maestro DP, 1399, 1472
inteligente, (Véase Esclavo I), (Véase Esclavo I)
Intercambio de datos, 1390, 1464
Leer datos de diagnóstico con DPNRM_DG, 4806, 6803
Reglas para slots, 1388, 1399
Sincronizar grupos con DP_SYC_FR, 4800, 6797
Tipos, 1399, 1471
Visualización del maestro DP, 1396, 1469
Esclavo DP inteligente, (Véase Esclavo I), (Véase Esclavo I)
Esclavo DP normalizado, 1432, 1485
Esclavo I, 1392, 1415, 1465, 1473
Acceso a datos, 1474
Catálogo de hardware, 1389, 1464
Comunicación directa, 1393
Configurar, 1392, 1416, 1474
Configurar CP, 1421
Configurar CPU, 1422
Configurar esclavo DP, 1423
Direcciones de diagnóstico, 1417
Disparar alarma de proceso con DP_PRAL, 6794
Dispositivos, 1389
enviar alarma con SALRM, 1418
Enviar alarma con SALRM, 6786
Intercambio de datos, 1392, 1465
Maestro DP, 1392
Esclavo normalizado, 1433, 1486
Esclavos DP normalizados
Escribir datos coherentes con DPWR_DAT, 4793, 6778
Escribir todas las salidas con SETIO, 4750, 6741
Escribir una parte de las salidas con SETIO_PART, 4752, 6743
Leer datos coherentes con DPRD_DAT, 4791, 6776
Leer todas las entradas con GETIO, 4749, 6740
Leer una parte de las entradas con GETIO_PART, 4751, 6742
Escribir
Datos de leva, 9752
Datos máquina, 9749
Señales de control, 7424

Un registro con WR_REC, 4790, 6774
Escribir área de memoria, 4392
Escribir bit de memoria, 4390
Escribir campo, 3451, 3764
Escribir datos
Dentro de la estación S7 propia con I_PUT, 8259
En CPU remota con PUT, 5252, 7680
En esclavos DP normalizados o dispositivos PROFINET IO con DPWR_DAT, 4793, 6778
Fuera de la estación S7 propia con X_PUT, 8275
Escribir dirección de memoria, 4388
Escribir variable, 990
Escritorio remoto, 32
Escritura
Datos máquina, 10901, 10995
Señales de control, 7420
tablas de incrementos, 10903
Tablas de incrementos, 10997
Esquema de contactos, 2667
Establecer conexión online, 2207
Establecer una conexión online, 1942
EstablecerColorRetroiluminación, 8694
EstablecerIdioma, 8696
EstablecerModoDeConexión, 8699
EstablecerModoDeOperaciónPanel, 8690
Establecimiento de una conexión desde una instalación remota ("PG_DIAL"), 11416
Establecimiento de una conexión remota, 11400
procedimiento, 11401
Estación
borrar, 525
cambiar nombre, 746
copiar, 526
desplazar, 528
Leer información con GetStationInfo, 4944
Estación PC
Conexiones, 611
Configuración OPC, 2062, 2067, 2068
Configuración XDB, 2075
configurar offline, 1203
configurar online, 1201
Insertar componentes, 1200
Insertar submódulos, 1200
Modo de operación, 1198
Reglas para slots, 1200
Términos, 1198
Estación permanente, 1209
Estación temporal, 1209
Estaciones activas, 822
Estaciones con dirección IP desconocida, 820
Estadísticas de Ethernet
Error de telegrama, 1253, 1254
Estadística de interfaz, 1251
Longitud de telegrama, 1252
Tipo de telegrama, 1253
Estado, 1680, 2545
Módulo averiado, 2181
Estado de aviso, 8511
Acusado, 8505
Aparecido, 8505
Desaparecido, 8505
Estado de conexión
Ver mediante iconos, 2214
Estado de diagnóstico
determinar y visualizar online, 2160
Estado de error, 1243
Estado de la conexión online, 2207
Estado de la hora
Modificar con SET_SW_S, 6707
Estado de la petición, 7443
ABS_CTRL, 7440
CAM_CTRL, 7474
CAM_CTRL_452, 7476
Estado de la petición de escritura
CNT2WRPN, 7417
Estado de la petición de lectura
CNT2RDPN, 7418
Estado de LED, 2236
Leer con LED, 4935
Leer mediante lista de estado del sistema, 6999
Estado del dispositivo
consultar de un interlocutor remoto con STATUS, 7720
Recibir el cambio con USTATUS, 7722
Estado del programa
Activar, 3098
AWL , 3110
Desactivar, 3098
Desactivar todas las etapas, 3105
Editar bloques, 3099
Entorno de llamada, 3087, 3088
Forzar variable, 3100
Función, 3086
FUP, 3109
GRAPH, 3102, 3105, 3106, 3112
Inicializar cadena secuencial, 3105
KOP, 3108
Modos de operación, 3102, 3106
SCL, 3111
Sincronización del sistema, 3103, 3107
Estado operativo, 876
ARRANQUE, 1623, 1902
Eventos de procesamiento, 1894
Introducción, 1621, 1900

PARADA, 1712
Prioridad, 1702
RUN, 1627, 1712, 1906
Sinopsis, 1701
STOP, 1627, 1703, 1906
Transiciones, 1622, 1702, 1890, 1901
Estados operativos de Motion Control, 10686
Estándar
PROFIBUS, 8864, 8917
Estilo
Activar, 8353
Borrar, 8323
Crear tipos nuevos, 9155
Editar, 8324
Modificar, 8324
Estilo estándar
Definir, 8324
Estilo HMI, 9155
Estilos predefinidos, 8352
Estructura, 2118, 2131
de dependencias, 3071
de la lista de referencias cruzadas, 9194
Estructura de llamadas, 3066
Ficha Carga de la memoria, 3078
STRUCT, 2358
Estructura de dependencias, 3070
Configurar las opciones de visualización, 3073
Estructura, 3071
Introducción, 3070
Mostrar, 3073
Significado de los símbolos, 3072
Estructura de hardware para Motion Control S7-1200, 10429
Estructura de la tabla de forzado permanente, 3145
Estructura de la trama, 1040
Estructura de llamadas, 1910, 1911, 3063
Estructura de llamadas
Significado de los símbolos, 3065
Configurar las opciones de visualización, 3068
Estructura, 3066
Introducción, 3063
Mostrar, 3067
Estructura de PDU con FETCH/WRITE, 2034
Estructura de regulación Motion Control, 10671
Estructura de telegrama , 11207
Estructura del telegrama, 1129
ET 200eco, 2096
ET 200eco PN, 2096, 2112
ET 200iSP, 2095
ET 200L, 2095
ET 200M, 1459, 2095, 2139
Definición, 2139
Reglas para slots, 1388
ET 200MP, 2130
ET 200pro, 2096
ET 200R, 2096
ET 200S, 1459, 2095
configuración futura, 1409, 1412, 1482
Configuración futura, 1479
Modo DPV1, 1413, 1483
Módulo de posicionamiento, 41
Reglas de slots, 1404, 1475
Uniones frías, 1405, 1475
ET 200S 1SI, 1068
ET 200S COMPACT, 2095
ET 200SP, 2118
Campo de aplicación, 2118
Utilización, 40
Etapa
Acciones, 2833
Borrar, 2895
Cambiar nombre, 2892
Comentar, 2893
Copiar y pegar, 2894
Desactivar todas las etapas, 3105
Elementos, 2832
Insertar, 2888
Interlock, 2832
Parámetros estáticos, 2857
Principios básicos, 2831
Renumerar, 2890
Saltar, 2895
Supervisión, 2833
Etapa inicial
Desmarcar, 2889
Establecer, 2889
Ethernet, 551, 553
Parámetros, 8838, 8893, 8952, 8962, 8990
ETHERNET
Basic Panel, 8790
Etiqueta, 3513, 3826, 4197, 5728, 5950, 6289
Etiqueta de salto, 4447, 6467
ETX, 1034
Evaluación
Alarmas de diagnóstico, 10976, 11078
Evaluación cuádruple, 9495
Evaluación de error con RET_VAL, 3279, 5598
Evaluación de errores
Programa de usuario, 9820
Evaluación de fallos
Programa de usuario, 9900, 10975, 11077
Evaluación de la señal, 9412, 9438
Evaluación doble, 9495
Evaluación simple, 9495

Evaluar
Informaciones de alarma con DPV1, 1438
Informaciones de diagnóstico con DPV1, 1438
Evaluar informaciones de alarma con DPV1, 1438
Evento, 1885
activar, 8715
Acusar, 8723
Al abrir un cuadro de diálogo, 8717
Al cerrar un cuadro de diálogo, 8718
Al rebasar por defecto, 8717
Al rebasar por exceso, 8717
Aparecido, 8721
Asignar bloque de organización (OB) con ATTACH, 4884
Borrada, 8715
Cambio de imagen, 8718
Cambio de usuario, 8718
Conmutar OFF, 8725
Conmutar ON, 8725
Desactivar, 8718
Desaparecido, 8720
Desbordamiento, 8724
Desbordamiento del búfer de avisos, 8722
Deshacer asignación a bloque de organización (OB) con DETACH, 4885
Ejecutar, 8716
Entrada finalizada, 8720
Hacer clic, 8721
Hacer clic cuando parpadee, 8721
Hacer doble clic, 8719
Loop-In-Alarm, 8722
Margen alcanzado, 8723
Modificación, 8715
Modificación de valor, 8726
Para listas de funciones, 8484
Parada de runtime, 8723
Poca memoria disponible, 8725
Poca memoria disponible, crítico, 8725
Pulsar, 8719
Pulsar ESC dos veces, 8720
Pulsar tecla, 8724
Selección modificada, 8716
soltar, 8722
Soltar tecla, 8724
Tiempo agotado, 8726
Variables, 8484
Ventana de inspección, 64
Evento de alarma
Bloquear con DIS_IRT, 4912, 6906
Habilitar con EN_AIRT, 4915, 6910
Habilitar con EN_IRT, 4914, 6908
Retardar con DIS_AIRT, 4915, 6909
Evento de arranque de OB, 1924
Evento de aviso
Acusar, 8505
Aparecer, 8505
Desaparecer, 8505
Evento de diagnóstico
Para módulos, 1896
Evento de error asíncrono
Bloquear con DIS_IRT, 4912, 6906
Habilitar con EN_AIRT, 4915, 6910
Habilitar con EN_IRT, 4914, 6908
ID de evento, 1887
Instrucciones, 6905
Retardar con DIS_AIRT, 4915, 6909
Eventos
Configuración, 1273
Filtro Severity, 1275
Tabla de registro, 1241
Eventos de comunicación, 1895
Eventos de diagnóstico , 9821, 9900, 10977, 11078
Eventos de error síncronos, 1886
Eventos de errores sincrónicos
Consultar desde el registro de estado de eventos con READ_ERR, 6903
Desenmascarar con DMSK_FLT, 6903
Enmascarar con MSK_FLT, 6902
Eventos de errores síncronos
Consultar desde el registro de estado de eventos con READ_ERR, 4910
Desenmascarar con DMSK_FLT, 4909
Enmascarar con MSK_FLT, 4908
Eventos de OB estándar, 1886
Eventos de parada, 1890
Eventos de seguridad, 836, 838
Eventos de sistema, 835, 838
Eventos de usuario estándar, 1898
Eventos e ID del evento, 1885
Eventos libres de usuario, 1900
EX, 2571
EXIT, 4446, 6466
EXP, 3423, 3736, 4148, 4345, 4544, 5692, 5915, 6242, 6428, 6567
Expandir y contraer secciones de código, 2801
Explorador de archivos, 936
Exportación
aviso, 9173
Formato Excel, 9169
Lista de textos, 9186
Receta, 9169
Textos del proyecto, 9190, 9210
Variable, 9180
Exportar servidores NTP, 809

Exportar tiras rotulables, 410
Expresión
    Expresión de comparación, 2790
    Expresión lógica, 2792
    Principios básicos, 2786
Expresión
    expresión aritmética, 2788
EXPT, 3435, 3747
externas, 47

F

F_TRIG, 3305, 3613, 3905, 4284
Facilidad, 797
Fallo de estación, (Error de rack)
Fallo del transductor de medida, 10283, 10335, 10370
FALSE, 4283, 4284
Falta la tensión de alimentación, 2133
Falta tensión de alimentación, 2125
Fase de configuración, 9279
Fase de control del proceso, 9279
Fast Startup (PROFINET IO), (véase Arranque priorizado)
Favoritos
    Agregar, 2685, 2729, 2774, 2810, 2881
    Eliminar, 2686, 2730, 2775, 2812, 2882
    Mostrar, 2629
    Ocultar, 2629
    Utilizar, 2685, 2729, 2775, 2811, 2882
FB, 2253
FB multiinstancia
    Easy Motion Control, 11261
FC, 2252
Fecha, 2344, 2346, 2347, 2348
Fecha/hora, 8741
    Puntero de área, 8867
Fecha/hora PLC, 8741
FETCH/WRITE, 1674, 1959
    Estructura de PDU, 2034
FETCH_RK, 1071, 8248
Ficha "Configuración IP", 1997
Ficha "Parámetros de puerto", 2003
Ficha "Propiedades OPC", 682
Ficha "Protección de acceso IP", 2006
Ficha Carga de la memoria
    Estructura, 3078
Fichero
    cambio de idioma, 9223
    Enviar datos con AR_SEND, 6950
    idioma de runtime, 9223
Fichero de avisos
    En runtime, 9322
Ficheros de proyecto, 357
FieldRead, 3449, 3762
FieldWrite, 3451, 3764
File Browser, 982
FILL, 51, 3490, 3803, 3983, 4411, 4589, 5710, 5933, 6073, 6442, 6585
FILL_BLK, 3461, 3775, 3960, 4368, 4566
Filtrado, 2134
Filtrar
    Plano de ocupación , 3061
Filtro
    Catálogo de hardware, 512
Filtro contextual, 512
Filtro de entrada, 9512, 9593
Filtro de escritura avanzado (EWF), 1206
Filtros
    Borrar, 3060
    definir para el plano de ocupación, 3060
    en el plano de ocupación, 3059
    Seleccionar, 3061
Fin
    Proceso de posicionamiento, 10904, 10999
Fin de bloque, 4234, 4235, 4237, 6325, 6326, 6328
Fin de cadena
    Finalizar cadenas secuenciales, 2886
    Introducción, 2830
Fin de la detección de topología, 1513, 8805, 8856, 8909
Fin de línea
    Línea, 8438
Fin del dominio Sync, 1513, 8805, 8856, 8909
Fin del registro de dispositivos accesibles, 1513, 8805, 8856, 8909
Final de carrera por hardware , 10676
Final de carrera por software, 9784, 9862, 10677, 10949, 11050
Final de una operación de posicionamiento
    Datos en el DB de parámetros, 10911, 11005
    Señales de respuesta en el DB de canal, 10911, 11005
Final del eje rotativo, 10950, 11051
Finales de carrera por hardware y por software
    Función, 10438
Finalizar el forzado permanente, 3169, 3170
FIND, 4735
Firmware, 1699
    Actualizar, 9272
Firmware Card, 1628
Firmware de la CPU, 1699

Flanco
Ascendente, 3295, 3298, 3300, 3604, 3607, 3610, 4090, 4283, 5612, 5614, 5832, 5835, 6185
Descendente, 3296, 3299, 3302, 3606, 3608, 3611, 4088, 4284, 5613, 5616, 5834, 5836, 6184
Flipflop
Activación/desactivación, 3292, 3602, 5609, 5829
Desactivación/activación, 3294, 3603, 5610, 5831
FLOOR, 3497, 3810, 4419, 4596, 5718, 5941, 6449, 6592
Flujo de datos, 9331
FM
con una dirección MPI propia, 915
FM 350-1, 9496
FM 350-2, 9552
Ajuste del canal, 9556
FM 351, 10896
Arranque, 7422
FM 352, 9754
Arranque, 7455, 7458
FM 355, 10252
FM 355-2, 10286
FM 450-1, 9581
FM 451, 10989
Arranque, 7426
FM 452, 9831
Arranque, 7455, 7458
FM 455, 10337
FMT_CJ_T
Instrucción, 7569
Parámetros de entrada , 7569
Parámetros de entrada/salida, 7570
Parámetros de salida, 7570
FMT_DS1
Instrucción, 7554
Parámetros de entrada , 7555
Parámetros de entrada/salida, 7555
Parámetros de salida, 7555
FMT_PAR
Instrucción, 7549
Parámetros de entrada , 7550
Parámetros de entrada/salida, 7551
Parámetros de salida, 7551
FMT_PID
Diagrama de bloques, 7540
Funcionamiento, 7530
Instrucción, 7528
Parámetros de entrada , 7532
Parámetros de salida, 7533
Variables estáticas, 7534, 7535, 7538
FMT_PV
Instrucción, 7563
Parámetros de entrada , 7567
Parámetros de entrada/salida, 7568
Parámetros de salida, 7568
FMT_TUN
Instrucción, 7557
Parámetros de entrada , 7558
Parámetros de entrada/salida, 7559
Parámetros de salida, 7558
Variables estáticas, 7560
FN, 4088, 6184
FOR, 4440, 6461
FORCE355
Instrucción, 7504
Parámetros de entrada, 7505
Parámetros de salida, 7506
FORCE455
Instrucción, 7589
Parámetros de entrada, 7591
Parámetros de salida, 7591
Formación de direcciones en el caso de las direcciones comprimidas, 1408, 1478
Formatear
Texto de aviso, 8520
Formato, 8427
Forzado permanente
Finalizar el forzado permanente, 3169, 3170
Forzar todos permanentemente, 3167
Forzado permanente de variables
Medidas de seguridad, 3144, 3162
FP, 4090, 6185
FR, 4091, 4111, 6187, 6206
FRAC, 3433, 3746, 4353, 4552
Fragmento, 958, 964, 996, 1002
Fragmento manual, 964, 1002
Frecuencia de filtrado, 9413, 9439
Frecuencia máx., 11200
Frecuencia normal, 11200
Frecuencímetro, 2109, 2112
Freeport
Búfer de recepción, 1033
Criterios de fin, 1029
Criterios de inicio, 1028
Fin del mensaje, 1027
Principio del mensaje, 1027
Recibir datos, 1047
Transferir datos, 1047
Transparencia del código, 1032
FREEZE, 1404
FREQUENC, 7654
FTP, 768, 770, 1674, 1959, 5348

FTP_CMD, 5348, 8167
FTPS, 5348
Fuente AWL
Generar bloques, 2981
Fuente de alimentación del sistema, 2131
Fuente predeterminada, 9222, 9289
Fuentes, 31
Fuentes externas, 47
Función
Ajustar valor real, 9835
Ajustar valor real al vuelo, 9757, 9835
asignar a una tecla de función, 8409
Cancelar preselección del valor real, 9757, 9835
Función (FC)
Crear, 2597
Definición, 2252
Exportar a un archivo fuente externo, 2978
Función de analizador lógico, 11327
Función de autocompletar
Función, 2637
Insertar instrucción, 2639
Insertar variable, 2638, 2639
Función de contaje en cascada, 2105
Función de documentación
Utilizar marcos y portadas prediseñados, 398
Función de paro de puerta, 9487
Función de puerta
de cancelación, 9489
de interrupción, 9490
Función de puerta de cancelación, 9489
Función de puerta de interrupción, 9490
Función de reloj, 1953, 1954, 2040
Ajustar la hora, 1653, 1749, 1952
Contador de horas de funcionamiento, 1751
Formato de hora, 1653, 1749, 1952
Leer la hora, 1653, 1749, 1952
Parámetros del reloj, 1653, 1750
Principios básicos, 1653, 1749, 1952
Función de sistema
en la lista de funciones, 8636
Lista de funciones, 8638
posibilidades de aplicación, 8636
utilización, 8635
Utilización, 8636
Función Trace, 11327
Funcionalidad
PROFIBUS DPV0/DPV1, 1434
ProSave, 9269
Funcionalidad de la tabla de forzado permanente, 3143
Funcionamiento
ABS_CTRL, 7420
ABS_CTRL_451, 7424
ABS_DIAG, 7422
ABS_DIAG_451, 7426
Ajustar valor real, 9757
CAM_CTRL, 7454
CAM_CTRL_452, 7457
CAM_DIAG, 7455
CAM_DIAG_452, 7459
CNT_CTL1, 7390
CNT_CTL2, 7393
CNT_CTRL, 7388
CNT2_CTR, 7406
CNT2RDPN, 7405
CNT2WRPN, 7404
Comunicación directa, 1423
Decalaje de señal cero, 9760, 9838
Definir punto de referencia, 9762, 9840
DIAG_INF, 7390
DIAG_RD, 7406
Generación de alarmas en el I-Slave, 1419
High_Speed_Counter, 5124
MC_Control, 7340
MC_GearIn, 7342
MC_Home, 7333
MC_Init, 7324
MC_MoveAbsolute, 7325
MC_MoveJog, 7330
MC_MoveRelative, 7328
MC_Simulation, 7341
MC_StopMotion, 7338
Modificar flancos de leva, 9764, 9842
Funcionamiento cíclico, 8956, 8959, 8991
Funcionamiento de los frecuencímetros, 2110
Funcionamiento online, 32, 37
Funcionamiento si la configuración teórica difiere de la real, 1405
Funciones
Actualizar valores de variables, 8461, 8776
Funciones de aprendizaje, 822
Funciones de contaje, 9360
Funciones de diagnóstico, 1054, 1197
Funciones de medición, 9389
Funciones de servidor OPC
Funciones de registro, 2069
para maestro DP clase 2, 2069
Funciones de sistema, 8634
ActivarBit, 8691
ActivarBitDeVariable, 8692
ActivarBitMientrasTeclaPulsada, 8693
ActivarImagen, 8649
ActivarImagenAnterior, 8652
ActivarImagenConNúmero, 8650

ActivarPantallaDeLimpieza, 8651
ActualizarVariable, 8653
AcusarAviso, 8677
AvanzarPágina, 8689
BorrarBúferDeAvisos , 8673
BorrarBúferDeAvisosProtool , 8674
BotónPulsar, 8685
BotónSoltar, 8686
CalibrarPantallaTáctil, 8663
CambiarConexión, 8705
CerrarSesión, 8647
DefinirVariable, 8697
DesactivarBit, 8683
DesactivarBitEnVariable, 8684
disponibles, 8644
EditarAviso, 8654
EscalaLineal, 8672
EstablecerIdioma, 8696
EstablecerModoDeConexión, 8699
EstablecerModoDeOperaciónPanel, 8690
IdentificarCambioDeUsuario, 8703
IncrementarValorEnfocado, 8657
IncrementarVariable, 8658
IniciarSesión, 8653
InvertirBit, 8660
InvertirBitEnVariable, 8661
InvertirEscalaLineal, 8662
IrAFin, 8659
IrAInicio, 8658
LeerContraseña, 8670
LeerNombreDeUsuario, 8669
LeerNúmeroDeGrupo, 8670
ModificarContraste, 8648
MostrarDiálogoInicioSesión, 8707
MostrarTextoDeAyuda, 8707
MostrarVentanaDeAvisos, 8708
ObjetoDeImagenAvanzarPágina, 8656
ObjetoDeImagenCursorAbajo, 8655
ObjetoDeImagenCursorArriba, 8655
ObjetoDeImagenRetrocederPágina, 8656
Parámetros en el programa del usuario, 910
PararRuntime, 8702
ReducirValorEnfocado, 8704
ReducirVariable, 8704
RetrocederPágina, 8689
SimularTeclaSistema, 8700
SimularVariable, 8701
VisorDeAvisosAcusarAviso, 8676
VisorDeAvisosEditarAviso, 8675
VisorDeAvisosMostrarTextoAyuda, 8676
VisorDeCurvasActivarModoRegla, 8667
VisorDeCurvasAmpliar, 8665
VisorDeCurvasAvancePágina, 8664
VisorDeCurvasComprimir, 8666
VisorDeCurvasIniciarParar, 8668
VisorDeCurvasIrAlPrincipio, 8668
VisorDeCurvasReglaDerecha, 8666
VisorDeCurvasReglaIzquierda, 8667
VisorDeCurvasRetrocesoPágina, 8665
VisorDeRecetasAbrir, 8679
VisorDeRecetasAtrás, 8682
VisorDeRecetasBorrarRegistro, 8678
VisorDeRecetasCambiarNombreDeRegistro, 8681
VisorDeRecetasEscribirRegistroEnControlador, 8680
VisorDeRecetasGuardarRegistro, 8681
VisorDeRecetasGuardarRegistroComo, 8681
VisorDeRecetasLeerRegistroDelControlador, 8678
VisorDeRecetasMenú, 8679
VisorDeRecetasMostrarTextoDeAyuda, 8682
VisorDeRecetasRegistroNuevo, 8678
Funciones matemáticas
CALCULATE, 2690, 2734
FUP, 2712, 3216, 3217, 3218, 3251, 3252, 3253
Comparación detallada, 2989
Migración, 139
FUZ_355
Instrucción, 7502
Parámetros de entrada, 7503
Parámetros de entrada/salida, 7504
Parámetros de salida, 7503
FUZ_455
Instrucción, 7587
Parámetros de entrada, 7588
Parámetros de entrada/salida, 7589
Parámetros de salida, 7588
FX3
Controlador, 9047

G

GADR_LGC, 5016
GEN_DIAG, 4953
Generador de impulsos
Activar/desactivar con CTRL_PWM, 4961
Generar información de diagnóstico, 4953
Generar patrón de bits para el display de 7 segmentos, 4057, 4671
GEO_LOG, 5012
GEO2LOG, 5004
Gestión de peticiones, 7420, 7424
GET, 5249

Get_AlarmState, 4919
GET_DIAG, 4956
Get_Features, 1014, 1048
Get_IM_Data, 4936
GET_IM_DATA, 51
GET_NAME, 31
GET_NAME:, 4940
GetBlockName, 4738
GetError, 2935, 2936, 3527, 3840, 4009, 4456, 4616
GetError , 2934
GetErrorID, 2934, 2935, 2936, 3531, 3844, 4013, 4460, 4620
GetGroupNumber, 8670
GetInstanceName, 4737
GETIO, 4749
GETIO_PART, 4751
GetPassword, 8670
GetPLCMode, 8671
GetStationInfo, 4944
GetSymbolName, 4736
GetUserName, 8669
Girar
  Objeto, 8328, 8340
GMRP, 1344
GOTO, 4447, 6467
GoToEnd, 8659
GoToHome, 8658
GPRS, 1208
Grabación de un archivo de informe para el módem, 11423
Gráfico
  Adaptar, 8431, 8433
  Adaptar tamaño, 8346
  Administrar, 8356
  Alineación, 8346
  asignar a una tecla de función, 8412
  Botón, 8449
  Con fondo transparente, 8356
  Distancia, 8347
  Editor, 9213
  Insertar, 8329
  insertar en la colección de gráficos, 9214
  Utilizar del juego de gráficos, 8356
  Visor de gráficos, 8431
Gráfico externo
  Borrar acceso directo a la carpeta, 8357
  Cambiar nombre a las carpetas, 8357
  Editar carpetas, 8357
  Enlazar carpeta, 8357
Gráficos, 8356, 8358
Gráficos , 9215
GRAPH, 47
  Acciones, 2836, 2837, 2839, 2842, 2843, 2844, 2901, 2907, 2908, 2909
  Acusar errores de supervisión, 3106
  Ajustes de test, 3103, 3104
  Borrar rama, 2899
  Cadenas secuenciales, 2827, 2828, 2883, 2884, 2885, 2886, 2887
  Cerrar rama, 2830, 2898
  Comparación detallada, 2999
  Condiciones, 2847, 2910, 2911, 2912, 2920
  Configuración, 2864
  Declarar parámetros internos remanentes, 2870
  Declarar parámetros internos visibles y accesibles desde HMI, 2870
  Definir el juego de parámetros de interfaz, 2868
  Definir el modelo de espacio de memoria, 2869
  Definir lenguaje de programación, 2867
  Desactivar todas las etapas, 3105
  Etapa, 2831, 2832, 2888, 2889, 2890, 2892, 2893, 2894, 2895
  Fin de cadena, 2830
  Inicializar cadena secuencial, 3105
  Instrucciones permanentes, 2846
  Instrucciones permanentes anteriores, 2873
  Instrucciones permanentes posteriores, 2878
  Interfaz del bloque, 2848, 2851, 2855, 2857, 2859, 2860
  Interlock, 2847
  Lenguaje de programación, 2825
  Llamadas de bloque, 2923, 2924, 2925
  Migración, 139
  Modificar la configuración, 2866
  Modos de operación, 3102, 3106
  Operandos específicos, 2834
  Principios básicos de los tipos de datos, 2905
  Rama alternativa, 2829, 2896
  Rama simultánea, 2828, 2897
  Resetear requisitos de registro, 49
  Salto a etapa, 2830, 2899, 2900, 2901
  Sincronización del sistema, 3103, 3107
  Supervisión, 2847
  Test con el estado del programa, 3102
  Transiciones, 2833, 2888, 2890, 2892, 2894, 2895
  Ventana de programación, 2871
  Vista de avisos, 2879
  Vista de cadena, 2874
  Vista de etapa individual, 2876
  Visualización del estado del programa, 3112
Grupo
  Añadir objetos, 8364
  Crear, 8362

Deshacer, 8363
Editar, 8362
Eliminar objeto, 8364
Grupo de avisos, 8509, 8511
Configurar, 8514
Crear, 8514
Migración, 156
Grupo de objetos
Animación, 8401
Editar objeto dentro de un grupo, 8365
Eliminar objeto, 8364
Grupo de redundancia, 2059
Grupo de usuarios, 9313
Administrar, 8612
Administrar autorización, 8612
Asignar, 8632
Asignar usuario, 8610
Borrar, 8613
Cambiar en runtime, 8620
Cambiar nombre, 8613
Cambiar nombre de visualización, 8613
Crear, 8631
Unauthorized, 8622
Grupo VPN, 814
Grupos de servicios, 780, 783
GSDML, (Ver archivos GSD (PROFINET))
Guardar
Bloque de datos de eje, 11258
Librería global, 9144
RLO en bit BR, 4087, 5764, 5985, 6182
Guardar diseño de ventana, 310
Guardar interfaz de usuario
Guardar diseño de ventana, 310
Guardar listines telefónicos, 11373
Guardar textos de aviso, 1115
Guardar una tabla de forzado permanente, 3149
Guardar una tabla de observación, 3121
Gx_STW, 11209, 11210
Gx_XIST2, 11210

H

Habilitación
Alarma de diagnóstico, 9398
Alarma de proceso, 9398
Habilitar
Alarma de diagnóstico, 10331
Avisos con EN_MSG, 6952
Hacer clic, 8721
Hacer clic cuando parpadee, 8721
Hacer doble clic, 8719
Handshake, 1662
Handshake de software, 1117
Hardware, 50
Comparar, 31
Configurar y parametrizar, 509
Detección, 42
Editar parámetros, 529
Editar propiedades, 529
Hardware interrupt, 1684
Hardware interrupt (S7-1500), 1973
Hardware Support Package
Instalar, 110
HART, 2100
Herramienta, (Véase Docking Unit), (Véase Docking Unit), (Véase Docking Unit)
Herramienta de migración, 148
Crear archivo de migración, 123
Desinstalación, 118
Incluir configuración hardware, 122
Requisitos del sistema, 117
Suministro y distribuidores, 117
Utilizar la herramienta de migración, 121
Herramientas, 8325
Herramientas online, 2159, 2168
High_Speed_Counter, 5123, 9406
Comportamiento en caso de error, 5126
configurar, 9409
descripción, 5123
diagnóstico, 9426
funcionamiento, 5124
llamada, 5123
Parámetros básicos, 9411
parámetros de entrada, 5127
parámetros de salida, 5129
programar, 9424
Puesta en servicio, 9425
variables estáticas, 5131
Histéresis, 9361, 9362, 9395, 9397, 9422, 9451, 9531, 9563, 9790, 9868
Leva de recorrido, 9790, 9868
Leva de tiempo, 9791, 9869
Hora, 1646, 1931, 2345, 2346, 2347, 2348
Ajustar e inicializar el estado de reloj con SET_CLKS, 6690
Ajustar la alarma horaria con S_LTINT, 6702
Ajustar la hora en la vista online y de diagnóstico, 2189
Calcular hora local con SET_TIMEZONE, 4700
Calcular la hora local con BT_LT, 6700
Calcular la hora patrón con LT_BT, 6701
De la CPU, ajustar con WR_SYS_T, 4695, 6687
Hora del sistema, 1281

Leer la hora del sistema de la CPU con TIME_TCK, 4704, 6689
Leer la hora y fecha de la CPU con RD_SYS_T, 4696, 6688
Precision Time Protocol, 1287
Sincronización, 2236
Sincronización horaria, 1284
Sincronizar entre esclavos y maestro, 6692
SNTP (Simple Network Time Protocol), 1284
Hora del sistema
Cliente NTP, 1286
Cliente PTP, 1287
Cliente SNTP, 1284
Hora local, 1646, 1931
Ajustar la alarma horaria con S_LTINT, 6702
Calcular a partir de la hora patrón con BT_LT, 6700
Calcular con SET_TIMEZONE, 4700
Escribir con WR_LOC_T, 4698
Leer con RD_LOC_T, 4697
Leer y calcular con LOC_TIME, 6699
Hora patrón
Calcular a partir de la hora local con LT_BT, 6701
Horario de verano/invierno
Alternar con SET_SW, 6704
Modificar el estado de la hora con SET_SW_S, 6707
HSC, 1654
HSP, (Véase Support Package)
HTA, 4727, 6721
HTTP, 778
Cargar/guardar, 1270
HTTPS, 854, 945, 981, 1752
Husillo, 10652
HW RTS siempre activado, 1025

I

I_ABORT, 8261
IB, 2571
ICMP, 785
Icono
Icono Overlay, 2166
para comparación, 2165
para diagnóstico de hardware, 2163
para diagnóstico de software, 2165
Para el estado de comparación, 2214
Para el estado de conexión, 2214
para estado operativo, 2166
Icono Overlay, 2166
ID, 2571
ID de enlace, 8160
Rango de valores, 8160
ID de evento, 1885
ID de la instalación, 1737
ID de red, 862
ID de situación, 1737
ID HW, (Véase ID de hardware)
véase ID de hardware, 971
Identificación de error, 10841
Identificación del fabricante, 1240
Identificación del proyecto, 8741
Identificador de hardware, 915, 917, 921, 971, 2120, 2132, (Véase ID de hardware), (véase ID de hardware)
Determinar con LOG2MOD, 5007
Identificador HW, 5367, 5370, 5373, 5377, 5383, 5384, 5385
Identificadores de elementos y bloques, 1812
IdentificarCambioDeUsuario, 8703
I-device, 1546
I-Device, 1501
Idioma
Activar el idioma del proyecto, 9203
Desactivar idioma del proyecto, 9203
fichero, 9223
Gráficos específicos del idioma, 9213
Idioma de edición, 9204
Idioma de referencia, 9204
Idiomas asiáticos, 9201
Localizar, 9201
Proyecto multilingüe , 9205
Representación de fechas, horas, importes y cifras en función del idioma, 9200
Representación en función del idioma, 9200
Sistema operativo asiático, 9201
Idioma (servidor web), 943, 944
Idioma de archivación, 9223
Idioma de edición, 349, 9199
Seleccionar, 9204
Idioma de la interfaz de usuario, 9199
seleccionar, 9202
Idioma de la interfaz para el servidor web, 944
Idioma de referencia, 349, 9199
Seleccionar, 9204
Idioma de runtime, 9199, 9217, 9288
fichero, 9223
Fuente, 9222, 9289
orden de cambio de idioma, 9220
Seleccionar, 9218, 9288
Idioma del proyecto, 349, 9199
activar, 9203
Desactivar, 9203

Idioma para el servidor web, 943, 944
Idiomas
Migrar, 159
Idiomas asiáticos
Configuración, 9291
Longitud de campos de texto, 9291
Tamaño de fuente, 9291
Idiomas del proyecto
Cambiar idioma de edición, 352
Definir idiomas del proyecto, 351
textos de usuario, 9206
textos del sistema, 9206
Utilización, 349
Idle Line, 1029
IE/AS-i Link PN IO, 1488
IE/PB Link PN IO, 2078
IEC V2.2, 1944
IEC V2.3, 1944
IEEE 802.3, 773
IF, 4436, 6456
IF1B
Basic Panel, 8790
IGMP, 1344
IGMP Querier, 1344
Igual, 3393, 3704
IGUAL A ARRAY, 3395, 3707
IM 155-6 DP HF, 40
IM 174
Configurar, 11196
Imagen, 9249, 9258, 9259, 9272
Borrar, 8314
Cambiar nombre, 8314
Copiar, 8314, 9239
Crear, 8313
Crear tipo, 9154
Dependencia del dispositivo, 8309
Desplazar, 8314
Fuente, 8310
Insertar, 8314
Librería, 9154
Panel de operador, 9268
Paso de trabajo para crear, 8313
Utilizar plantilla, 8322
Zoom, 8312
Imagen general, 8318
Plantilla, 8319
Tecla de función, 8318
Imagen inicial, 8315
IMC, 3575, 3891, 4042, 4495, 4659, 5799, 6020, 6131, 6517, 6652
Importación
Aviso, 9174
Estructura de los datos de receta, 9172
Estructura del aviso analógico, 9175
Estructura del aviso de bit, 9178
Lista de textos, 9187
Receta, 9171
Textos del proyecto, 9192, 9212
Variable, 9182
Importar aviso de bit
Estructura para la importación, 9178
Importar servidor NTP, 809
Importar un aviso analógico
Estructura para la importación, 9175
Importar una receta
Estructura para la importación, 9172
Importar variables
Estructura de los datos de las variables, 9183, 9188
Imposible establecer la conexión remota, 11401
Impresión de tiras rotulables, 411
Determinar el valor de corrección para desplazar la impresión, 413
Imprimir
Contenidos de impresión, 394
Crear marcos, 396
Crear portada, 397
Definir el área de impresión, 402
Definir el diseño, 394
Editar marco, 398
Editar portada, 398
Elementos de la librería, 396
Elementos del árbol del proyecto, 396
Elementos no imprimibles, 391
Estructura del impreso, 391
Función de documentación, 390
Imprimir vista de dispositivos, 498
Imprimir vista de redes, 498
Modificar los ajustes, 392
Objetos no imprimibles, 392
Utilizar marcos, 395
Utilizar marcos y portadas, 390
Utilizar plantillas de impresión, 394
Utilizar portada, 394
Imprimir un proyecto, 407
Impulso, 3306, 3316, 3328, 3343, 3615, 3637, 3652, 3907, 4099, 4286, 4301, 4512, 5617, 5624, 5639, 5838, 5845, 5860, 6044, 6194, 6382, 6391, 6540
Prolongado, 3331, 3344, 3640, 3654, 4101, 4304, 5627, 5640, 5848, 5862, 6196, 6394
Impulsos erróneos (encóder incremental), 9800, 9878, 10963, 11064
IN_RANGE, 3383, 3695
INC, 3409, 3720, 4137, 6231

IncreaseFocusedValue, 8657
IncreaseTag, 8658
Incrementar, 3409, 3720, 4137, 6231
IncrementarValorEnfocado, 8657
IncrementarVariable, 8658
Incrementos, 10964, 11065
Indicaciones generales para la búsqueda de errores, 11422
Indicaciones para parametrizar un módem remoto, 11382
Indicador de avisos, 64, 8443, 8526, 8537, 9325
    Categorías, 8444
    Configurar, 8528
    En runtime, 8534, 9322
    Eventos, 8444
    Manejo, 8537
    Representación, 8444, 8537
Indicador de ayuda, 8433
Indicadores de error, 10279, 10330, 10364
Industrial Ethernet, 8760
Información
    Error de telegrama, 1253, 1254
    Estadísticas de Ethernet, 1252
    Registro de evento, 1241
    Spanning Tree, 1244
    Tabla ARP, 1241
    Tipo de telegrama, 1253
    Versions, 1239
Información avanzada de estado, 2795
Información de arranque, 1961, 2252
    Como en las CPU S7-300 y S7-400, 1961, 1962, 1964, 1967, 1968, 1969, 1970, 1972, 1973, 1974, 1975, 1979, 1980, 1981, 1982, 1984, 1986
    Optimizado, 1961, 1962, 1964, 1967, 1968, 1969, 1970, 1972, 1973, 1974, 1975, 1979, 1980, 1981, 1982, 1984, 1986
Información de arranque del OB, 1961
Información de calidad, 1593, 1918, 1919
Información de conexión, 2215
    Diagnóstico, 8748
Información de error, 2934, 3279, 3527, 3531, 3840, 3844, 4009, 4013, 4456, 4460, 4616, 4620, 5598
Información del dispositivo, 2159
    Diagnóstico, 8748
Información del documento, 395
Información del programa, 1911
    en el plano de ocupación, 3054
    en la estructura de dependencias, 3070
    en la estructura de llamada, 3063
    en la ficha , 3076
    Mostrar, 3053
    Vistas, 3053
Información importante sobre la parametrización del TS Adapter MPI, 11390
Información relevante para la instalación
    Datos del fabricante, 2231
    Datos sobre el módulo, 2231
    Información del fabricante, 2169
    Información del módulo, 2169
Informaciones de diagnóstico
    Evaluar con DPV1, 1438
Informaciones de sistema
    Direcciones, 1442
Ingeniería de equipo
    Requisitos de software y hardware, 11281
    Requisitos para el trabajo paralelo en equipo, 11282
Ingeniería interproyecto
    Abrir "Datos del proxy del dispositivo", 3205
    Creación de "Datos del proxy del dispositivo" en el proyecto de origen, 3204
    Intercambio de datos de control entre varios proyectos, 3201, 11294
    Intercambio de datos de control mediante "Archivo IPE", 3205
    IPE, 9156
    Procedimiento de intercambio de datos de control mediante un archivo de proyecto, 3202, 11295
    Procedimiento de intercambio de datos de control mediante un archivo IPE, 3201, 11294
    Procedimiento para la actualización de los datos de control ya transferidos, 3203, 11295
    Requisitos, 3201, 11293
    Requisitos de software y hardware, 3200, 11293
Ingeniería interproyecto (IPE)
    Intercambio de datos de control mediante un archivo de proyecto, 3200, 11292
    Intercambio de datos de control mediante un archivo IPE, 3199, 11292
    Principios básicos, 3198, 11290
Inicialización, 1196
    Archivo de proyecto, 9163
    Archivo IPE, 9157
Inicialización (páginas web definidas por el usuario), 947, 984
Inicializar
    Archivo de proyecto, 9163
    Archivo IPE, 9158
    Contador, 4117, 6211
    Datos de control, 9156
    Temporizador, 4097, 6193
    Temporizador IEC, 3325, 3919, 4298
Inicializar contador, 9512
    Con la entrada digital I2, 9514

Con la señal de paso por cero, 9515
Mediante una señal externa, 9513
Inicializar el estado del reloj con SET_CLKS, 6690
Inicializar temporizador, 3634
Inicializar todos los datos remanentes, 3534, 3847, 4016, 4463, 4622
Iniciar el cargador de firmware, 2245
Iniciar el diagnóstico de NCM S7, 2241
Iniciar herramienta de migración, 122
Iniciar la desinstalación, 115
Iniciar la simulación, 488
Iniciar la vista online y de diagnóstico, 2167
Iniciar la vista topológica, 735
Iniciar sesión, 940
Inicio de sesión incorrecto, 8622
registrar en informes, 8626
Usuarios, 8621
Iniciar sesión para el servidor web, 979
IniciarSesión, 8653
Inicio de runtime
Basic Panel, 9261, 9284
Inicio de sesión en servidor web, 979
Inicio de una petición de desplazamiento
Easy Motion Control, 11254
Inicio desde la vista de recetas
Transferir datos, 8878, 8936, 9128
Inicio mediante una función configurada
Transferir datos, 8828, 8881, 8939, 8978, 9131
INIT_RD, 3534, 3847, 4016, 4463, 4622
Inserción de instrucciones de Motion Control, 10755
Inserción y extracción de CPs PC Ethernet, 2062
INSERT, 4733
insertar
Gráfico en la colección de gráficos, 9214
gráfico externo , 9215
Insertar
Gráfico, 8329
Objeto, 8328, 8329, 8353
Objeto de librería, 9152
Rectángulo, 8372
Insertar componentes
Estación PC, 1200
Insertar consecutivamente, 8353
Insertar entrada, 2741
Insertar fila de separación, 10487
Insertar fila Empty, 10487
Insertar sección de comentario, 4449, 6469
Insertar submódulos
Estación PC, 1200
Insertar/extraer una tarjeta de red USB, 2062
Instalación
Actualizaciones, 110
Actualizar productos, 112
Directorio de destino, 85
Ejecutar automáticamente, 85
Informe, 106
Iniciar, 107
Licencias, 86
Modificar productos, 112
ProSave, 75
Reparar productos, 114
Requisitos del sistema, 87
Startdrive, 60
Support Package, 110
Ver software, 111
Instalación de claves de licencia, 90
Instalar actualización, 110
Instalar certificado CA, 11405
Instalar el módem local, 11380
Instalar Support Package, 110
Instancia individual
Cambiar tipo de llamada, 2689, 2733, 2780
Definición, 2265
Ejemplo, 2265
Instrucción, 8190
AG_SEND / AG_LSEND / AG_SSEND, 8055
Borrar, 2825
Copiar, 2824
Cortar, 2824
Definir el tipo de datos, 2682, 2684, 2726, 2727, 2770, 2771
Insertar, 2824
Reglas, 2804
Versiones, 2631
Instrucción "WWW", 961, 999
Instrucción AWL
Borrar, 2785
Copiar, 2783
Cortar, 2784
Insertar, 2769
Insertar operando, 2772
Introducir , 2768
para insertar del portapapeles, 2784
Seleccionar , 2782
Instrucción de comunicación
"AS_DIAL", 11417, 11418
"AS_MAIL", 11420
"PG_DIAL", 11415, 11416
"SMS_SEND", 11419
Instrucción nula, 4280, 4281, 4282, 6380, 6381, 6382
Instrucción permanente
Insertar, 2920
Introducción, 2846

Instrucción PNIO_ALARM
Parámetros formales, 8113
Instrucción PNIO_RECV
Parámetros formales, 8100
Instrucción PNIO_RW_REC
Parámetros formales, 8109
Instrucción PNIO_SEND
Parámetros formales, 8093
Instrucción SCL
insertar, 2806
Insertar, 2805
Modificar el tipo de datos, 2808, 2809
Principios básicos de los tipos de datos, 2807
Reglas, 2804
Seleccionar, 2823
Instrucciones
AG_LOCK / AG_UNLOCK, 8063
AG_RECV / AG_LRECV / AG_SRECV, 8059
Indicaciones generales, 8190
Nombre simbólico, 8190
P3964_Config (configuración del protocolo), 5408, 7798
para conexiones configuradas, 8051
para el modo de operación DP, 8207
para enlace AG-AG, 8193
Port_Config (configuración del puerto), 5399, 7789
Receive_Config (configuración de la recepción), 5403, 7793
Receive_P2P (recibir datos punto a punto), 5413, 7803
Receive_Reset (resetear receptor), 5415, 7805
Send_Config (configuración de la transmisión), 5402, 7792
Send_P2P (enviar datos punto a punto), 5410, 7800
Signal_Get (consultar señales RS232), 5415, 7805
Signal_Set (establecer señales RS232), 5417, 7807
Sistema de automatización S7-400, 7966
USS_Drive_Control, 5440, 7831
USS_Port_Scan , 5438, 7828
USS_Read_Param, 5443, 7834
USS_Write_Param, 5445, 7835
Instrucciones de Motion Control
Programar, 10751
Instrucciones Modbus, 1052
Instrucciones para coordinación de accesos en caso de FETCH/WRITE, 8063
Instrucciones PtP, 1047
Instrucciones USS, 1050
INT, 2331, 2384, 2426, 2470, 2491, 2515, 2528
INT_TO_, 2426, 2491, 2528
INTEG
Instrucción, 7250
Parámetros de entrada, 7253
Parámetros de salida, 7253
Integrada
Conexión, 8765, 8769
Integrar en avisos los textos de listas de textos, 3196
Ejemplo, 3196
Intercambiar, 4170, 4229, 6264, 6321
Intercambio de datos, 2753, 8734
Curvas, 8831, 8992
Esclavo DP, 1390, 1464
Esclavo I - maestro DP, 1392, 1465
mediante punteros de área, 8741
Variables, 8740
Intercambio de datos a través del acoplamiento remoto AS-AS
instrucción de comunicación "AS_DIAL", 11417
Intercambio de datos unidireccional/bidireccional, 1074, 1091
Interconectar, 553
Interconectar puertos
Vista de tabla, 744, 746
vista gráfica, 745
Vista gráfica, 742
Interfaces, 31, 1010
Interfaces (PROFINET), 1942
Interfaces de comunicación
Programar, 5394, 7784
Interfaces Ethernet
EC31-HMI/RTX, 1205
EC31-RTX, 1205
EC31-RTX F, 1205
Interfaz, (Véase Interfaz del bloque)
agregar, 11312
cambiar nombre, 746
Mostrar, 2171, 2233
Interfaz de control, 9459
1Count24V para la lectura del recorrido, 9664
1Count24V para modos de contaje, 9634
1Count24V para modos de medición, 9652
1Count5V para la lectura del recorrido, 9724
1Count5V para modos de contaje, 9695
1Count5V para modos de medición, 9711
1SSI en el modo estándar, 10877, 10878
1STEP 5V, 11188
2PULSE (corto), 10422
2PULSE (largo), 10423
FM 350-1 para modos de contaje, 9534

FM 350-1 para modos de medición, 9541
FM 351, 10968
FM 352, 9814
FM 451, 11070
FM 452, 9892
Interfaz de impulsos
Principio, 10434
Interfaz de notificación
1Count5V para la lectura del recorrido, 9725
1Count5V para modos de contaje, 9697
1Count5V para modos de medición, 9713
Interfaz de realimentación, 9461
1Count24V para el modo Fast, 9672
1Count24V para la lectura del recorrido, 9665
1Count24V para modos de contaje, 9636
1Count24V para modos de medición, 9654
1Count5V para el modo Fast, 9732
1SSI en el modo Fast, 10886
1STEP 5V, 11191
2PULSE, 10425
FM 350-1 para modos de contaje, 9538
FM 350-1 para modos de medición, 9543
FM 351, 10967
FM 352, 9813
FM 451, 11069
FM 452, 9892
Interfaz de usuario
Árbol del proyecto, 288
Área de trabajo, 291
Maximizar el área de trabajo, 293
Minimizar el área de trabajo, 293
Paleta "Proyectos de referencia", 303
Task Card, 301
Ventana de inspección, 299
Ventana general, 306
Vista del portal, 284
Vista del proyecto, 285
Vista detallada, 305
Vistas, 283
Interfaz del bloque
Actualizar, 2656
Declarar ARRAY, 2651
Declarar STRUCT, 2652
Declarar tipo de datos PLC, 2653
Declarar variable, 2648, 2650, 2654
Estructura, 2642
Finalidad de la declaración de variables, 2642
GRAPH, 2848
Importar y exportar variables, 2666
Mostrar, 2628
Multiinstancia, 2655
Ocultar, 2628
Propiedades de las variables, 2658, 2661
Remanencia, 2660
Tipos de datos válidos, 2646, 2647
Interfaz DP, 1394, 1468, 2234
Cambiar, 1398
Direcciones, 1440
Interfaz Ethernet
Visualizar parámetros, 11314
Interfaz MPI, 2233
Interfaz MPI/DP, 2233
Interfaz PG/PC
Establecer conexión online, 11303
Parámetros ajustables en la interfaz MPI, 11318
Parámetros ajustables en la interfaz PROFIBUS, 11318
Interfaz PROFINET, 1501, 1646, 1931, 1942, 2171, 2234
Interfaz X27 (RS422/485), 1123
Interlock
Introducción, 2832, 2847
Programar, 2910
Interlocutores
Conectar en red, 8764
Controlador, 8734
Panel de operador, 8734
SIMATIC S7, 8841, 8896, 8954, 8964, 8967
Internal Ethernet Port
Open settings, 9050
Internet Key Exchange (IKE), 817
Interno
Puerto Ethernet, 9050
Interpretación de la señal, 9494
Interruptor, 8447, 9310
Comportamiento, 9310
Manejo, 9310
Representación, 9310
Tipo, 8447
Utilización, 9310
Interruptor de PARADA DE EMERGENCIA
Easy Motion Control, 11262
Interruptor libre de rebotes, 11027
Introducir
Tipo de telegrama, 11206
Intrusion Detection System (IDS), 95, 99
INV, 3544, 3858, 4625, 5744, 5967, 6610
InverseLinearScaling, 8662
Inversión de sentido en final de carrera por hardware, 10667
Inversor, 2142
InvertBit, 8660
InvertBitInTag, 8661

Invertir, 3286, 3544, 3595, 3858, 4084, 4184, 4185, 4189, 4191, 5604, 5744, 5825, 5967, 6180, 6277, 6278, 6283, 6284
InvertirBit, 8660
InvertirBitEnVariable, 8661
InvertirEscalaLineal, 8662
IO access error (S7-1500), 1986
IO2MOD, 5008
IO-Link, 1609
IP_CONFIG, 8150
IP-ACL (lista IP Access Control), 2006
iPad, 971
IPE, 9156
    Archivo, 9157, 9158
IPP, (Véase Memoria imagen parcial de proceso), (Véase Memoria imagen parcial de proceso)
IPv6
    Cliente FTP, 2152
    E-mail, 2153
    FETCH/WRITE, 2152
    Notación, 2152
    Servidor FTP, 2152
    SNMP, 2153
    Uso en el CP 1543-1, 2152
IQ-Sense, 2144
IrAFin, 8659
IrAInicio, 8658
IRT, 1530
IRT (Isochronous Realtime), 1526, 1530, 1533
IS_ARRAY, 3395, 3707, 4334
IS_NULL, 3393, 3704
ISAKMP, 826
I-Slave
    Activar alarma de proceso, 1443
    Modelo de slot de los esclavos DP, 1440
ISO-on-TCP
    Características, 709
    TSAP, 719
ITB, 4177, 6271
ITD, 4180, 6274
IW, 2571
IX, 235, 2571

Í

Índice de una estación PC, 1198
Índice variable, 260

J

JBI, 4205, 6297
JC, 4199, 6292
JCB, 4202, 6295
JCN, 4201, 6293
Jerarquía de llamada, 2264
JL, 4223, 6316
JM, 4216, 6309
JMP, 3511, 3824, 5726, 5948
JMP_LIST, 3514, 3827
JMPN, 3512, 3825, 5727, 5949
JMZ, 4220, 6312
JN, 4212, 6305
JNB, 4203, 6296
JNBI, 4206, 6299
JO, 4207, 6300
JOG, 10911, 11005
    Cancelar, 10913, 11007
    Datos en el DB de canal, 10912, 11006
    Desactivación, 10913, 11007
    Desarrollo, 10911, 11005
    Easy Motion Control, 11220
    Límite de la zona de trabajo, 10913, 11007
    Requisitos, 10911, 11005
    Vigilancias, 10913, 11007
JOS, 4208, 6301
JP, 4214, 6307
JPZ, 4218, 4220, 6310, 6312
jQuery, 971
JU, 4198, 6291
Juego de caracteres
    en textos de aviso, 1115
Juego de parámetros de interfaz, 2848, 2851, 2855, 2868
JUO, 4222, 6314
Justificar
    Objeto justificado, 8337
JZ, 4210, 6303

K

KEY-PLUG, 1297, 1299
    formatear, 1298
KOP, 2667, 3209, 3243
    Comparación detallada, 2989
    Migración, 139

L

L, 4093, 4113, 4157, 6189, 6207, 6251
L DBLG, 4230, 6322
L DBNO, 4231, 6323
L DILG, 4232, 6323

L DINO, 4233, 6324
L STW, 4159, 6253
La conexión módem se interrumpe, 11427
La conexión VPN no se establece
Comprobación y solución, 11428
La estación remota no logra conectarse con el TS Adapter, 11425
LABEL, 3513, 3826, 5728, 5950
LACP, 1335
LAG1ST
Instrucción, 7253
Parámetros de entrada, 7257
Parámetros de salida, 7257
LAG2ND
Parámetros de entrada, 7260
Parámetros de salida, 7260
LAR1, 4160, 6254
LAR1 <D>, 4161, 6255
LAR1 AR2, 4163, 6257
LAR2, 4164, 6258
LAR2 <D>, 4165, 6259
Layer 3, 1299
LC, 4095, 4114, 6191, 6209
LDT, 2347, 2399, 2449
LDT_TO_, 2449
LEAD_LAG, 3580, 3896, 4053, 4500, 4668, 5804, 6025, 6142, 6522, 6657
LEAVE, 6368
Lector de tarjetas USB, 34
Lectura
Registro de receta, 9337
Señales de respuesta, 7420
Lectura continua
Variable, 8479
Lectura de datos de servicio, 939
Lectura de los valores del sensor
isócrono, 10868, 10880
libre, 10853, 10868, 10880
modo isócrono, 10853, 10880
Lectura de posición
con encóder absoluto SSI, 9362
Lectura de recorrido
con encóder absoluto SSI, 9362
con Motion Control, 9398
Lectura de recorrido (CPU 31xC), 1776, 1784
LED, 4935
LED ACK, 9324
LED de error agrupado
SM 338, 10864
LED de error de grupo, 10279, 10330
FM 350-1, 9546
LED de error general
FM 351, 10972
FM 352, 9818
LEDs de error, 10364
Leer
Contador de horas de funcionamiento con READ_RTM, 6697
Datos de leva, 9753
Datos máquina, 9751, 10902, 10996
De un bloque de datos de la memoria de carga con READ_DBL, 4997, 7032
Estado de LED con LED, 4935
Hora local con LOC_TIME, 6699
Hora local con RD_LOC_T, 4697
Lista parcial SZL con RDSYSST, 6963
Recursos del sistema asignados dinámicamente, con READ_SI, 6956
Registro de un módulo con RD_DPAR, 4875, 6874
Registro de un módulo de forma asíncrona, con RD_DPARA, 4878, 6876
Señales de respuesta, 7424
Tablas de incrementos, 10903, 10998
Un registro con RD_REC, 4785, 6769
Leer bit de memoria, 4386
Leer campo, 3449, 3762
Leer datos
De esclavos DP normalizados o dispositivos PROFINET IO con DPRD_DAT, 4791, 6776
de una CPU remota con GET, 5249
Dentro de la estación S7 propia con I_GET, 8257
Fuera de la estación S7 propia con X_GET, 8273
Leer dirección de memoria, 4384
Leer el búfer de diagnóstico, 2182
Leer la hora del sistema con TIME_TCK, 4704, 6689
Leer variable, 951, 988
LeerContraseña, 8670
LeerModoPLC, 8671
LeerNombreDeUsuario, 8669
LeerNúmeroDeGrupo, 8670
LEFT, 4730
Legitimación de contraseñas, 3521, 3834, 4002, 4450, 4610
LEN, 4729
Lenguaje de programación
Cambiar, 2867, 2931
FUP, 2712
GRAPH, 2825
KOP, 2667
Reglas para cambiar, 2931
Leva
Ajuste dinámico, 9810, 9888

Alarma de proceso, 9812, 9890
Alcance, 9804, 9882
Cambio de sentido, 9741
Comportamiento de conmutación, 9811, 9889
Datos de leva, 9804, 9882
Final de leva en levas de recorrido, 9806, 9884
Inicio de leva en levas de recorrido, 9806, 9884
Inicio de leva en levas de tiempo, 9807, 9885
Inicio de levas en levas de tiempo, 9807, 9885
Leva "válida", 9805, 9883
Número de pista, 9805, 9883
Sentido de actuación, 9810, 9888
Tiempo de anticipación, 9808, 9886
Tiempo de conexión, 9807, 9885
Tipo de leva, 9806, 9884
Leva de inversión, 10655, 10667
Leva de recorrido, 9735, 9736
Leva de tiempo, 9735, 9736
Activar, 9757, 9834
Comportamiento de conmutación, 9757, 9834
Influencia de las funciones, 9757, 9834
LGC_GADR, 5017
Librería, 9134
Abrir, 9145
Abrir una versión de tipo habilitada, 462
Agregar plantilla maestra, 456
Agregar y utilizar un bloque, 2599
Armonizar nombres y estructura de rutas, 484
Borrar instancias, 481
Borrar tipos y versiones, 481
Botones e interruptores, 9138
Comparación de elementos de librerías, 486
Copiar tipos en el portapapeles, 480
Copiar un objeto de librería, 9137
Cortar elementos de librería, 480
Cortar plantillas maestras, 480
Cortar tipos, 480
Crear carpetas, 454
Depositar objeto, 9151
Desplazar elementos de librería, 481
Estilo, 9155
Guardar, 9144
Limpiar, 484
Mostrar propiedades de un tipo, 463
Pegar plantillas maestras del portapapeles, 480
Pegar tipos del portapapeles, 480
Plantilla maestra, 9137
Plantillas maestras, 427
Principios básicos, 427
Task Card, 429
Tipo, 9137
Tipos, 427
Tipos de datos PLC generados por el sistema, 50
Utilización de tipos, 458
Utilizar la vista de elementos, 431
Utilizar la vista de filtro, 457
Utilizar plantillas maestras, 455, 458
Versión de tipo en proceso, 460
Versión de tipo en test, 460
Versión de tipo habilitada, 461
Versionamiento de tipos, 459
Librería del protocolo USS
Información general sobre el dispositivo de accionamiento, 5446, 7837
Requisitos para el uso, 5435, 7825
USS_Drive_Control, 5440, 7831
USS_Port_Scan , 5438, 7828
USS_Read_Param, 5443, 7834
USS_Write_Param, 5445, 7835
Vista general, 5434, 7824
Librería del proyecto, 59, 427, 9134
Actualizar proyecto, 472
Agregar tipos, 463
Asignar una versión común a los tipos, 478
Comprobar la coherencia de una versión de tipo, 468
Crear carpetas, 454
Habilitar todas las versiones, 471
Habilitar versiones de tipos, 470
Limpiar, 484
Migrar, 161
Rechazar todas las versiones, 469
Rechazar versión de tipo, 469
Uso de tipos, 464
Utilizar la vista de elementos, 431
Utilizar la vista de filtro, 457
Librería global, 9134
Abrir, 443, 9145
Actualizar, 9249
Actualizar proyecto, 477
Agregar tipos, 474
Archivar, 451
Borrar, 449
Cerrar, 448
Crear, 441, 9143
Crear carpetas, 454
Guardar, 447, 9144
Limpiar, 485
Migrar, 161
Mostrar las propiedades, 446
Sinopsis del protocolo USS, 5434, 7824
Uso de tipos, 475
Utilizar la vista de elementos, 431
Utilizar la vista de filtro, 457

Visualizar informes, 447, 9145
Librerías globales, 428
Archivar, 450
Compatibilidad con versiones anteriores del producto, 442
Desarchivar, 452
Modo de compatibilidad V12 SP1, 442
Licencia
Defectuosa, 92, 105
Inicio sin licencia válida, 89, 101
Licencia defectuosa, 92, 105
Licencia defectuosa, 92, 105
Licencia no válida
Al cambiar la zona horaria, 91, 105
Licencias, 86
License Manager Panel Plugin, 102
LIMALARM
Instrucción, 7260
Parámetros de entrada, 7262
Parámetros de salida, 7262
LIMIT, 3417, 3729, 3938, 4340, 4538, 5685, 5908, 6064, 6423, 6562
Limitación
Modicon Modbus RTU, 9096
Modicon Modbus TCP/IP, 9096
Limitación de tirones, 10679, 10716, 10718, 10733, 10735
Función, 10439
Limitación del rango de desplazamiento, 10675
Limitaciones, 1513, 8805, 8856, 8909
Limitar y habilitar legitimación de la contraseña, 3521, 3834, 4002, 4450, 4610
Límite
Variable, 8475
Límite de la zona de trabajo
JOG, 10913, 11007
Límite superior de contaje, 9501, 9560, 9633, 9694
LIMITER
Instrucción, 7263
Parámetros de entrada, 7264
Parámetros de salida, 7264
Límites de ARRAY, 2795
Límites de contaje, 9360, 9363, 9416
Límites del sistema
Panel de operador, 9341, 9344
Límites dinámicos, 10679, 10716, 10733
Line Break, 1029
Línea, 8437
Diseñar, 8343
Fin de línea, 8343, 8438
Principio de línea, 8438
LinearScaling, 8672
Link Layer Discovery Protocol (LLDP), 1944
LINT, 2335, 2389, 2435
LINT_TO_, 2435
Lista de chequeo en caso de dificultades con el módem, 11426
Lista de contraseñas, 9313
Lista de control de acceso IP, 769
Lista de elementos, 8577
Lista de estado del sistema
SZL-ID W#16#xy90: información del sistema maestro DP, 7001
SZL-ID W#16#xy9C: Información del intercambiador de herramientas (PROFINET IO), 7016
Definición, 6971
Estructura de una lista parcial, 6972
Extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#0005, 6994
Extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#0008, 6994
Extracto de lista parcial con la ID de SZL W#16#0132 Índice W#16#000B, 6996
Extracto de lista parcial con la SZL-ID W#16#0132 Índice W#16#000C, 6997
Extracto de lista parcial con la SZL-ID W#16#0232 Índice W#16#0004, 6998
ID de SZL W#16#xy11: ID de módulo, 6975
ID de SZL W#16#xy12: características de la CPU, 6976
ID de SZL W#16#xy13: áreas de memoria del usuario, 6978
ID de SZL W#16#xy14: áreas del sistema, 6979
ID de SZL W#16#xy15: Tipos de bloque, 6981
ID de SZL W#16#xy1C: identificación de un componente, 6981
ID de SZL W#16#xy22: estado de alarma, 6987
ID de SZL W#16#xy25: correspondencia entre memorias imagen parcial de proceso y OBs, 6990
ID de SZL W#16#xy32: Datos de estado de comunicación, 6993
ID de SZL, estructura, 6973
Leer extracto de lista parcial SZL con RDSYSST, 6963
Listas parciales, sinopsis, 6974
SZL-ID W#16#00B1: información de diagnóstico de rack, 7020
SZL-ID W#16#00B2: registro de diagnóstico 1 a través de dirección física, 7021
SZL-ID W#16#00B3: datos de diagnóstico de módulo a través de la dirección base lógica, 7022

SZL-ID W#16#00B4: datos de diagnóstico de un esclavo DP, 7022
SZL-ID W#16#0x94: información de estado de rack o estación, 7010
SZL-ID W#16#xy74: estado de los LEDs de módulo, 6999
SZL-ID W#16#xy91: información de estado de módulo, 7002
SZL-ID W#16#xy92: Información de estado de rack o estación, 7007
SZL-ID W#16#xy95: información avanzada del sistema maestro DP, 7012
SZL-ID W#16#xy96: información de estado de módulo con PROFINET IO y PROFIBUS DP, 7013
SZL-ID W#16#xyA0: búfer de diagnóstico, 7019
Lista de funciones, 8484, 8636, 8729
Función de sistema, 8638
Procesamiento, 8638
Procesamiento asíncrono, 8641
Procesamiento en runtime, 8641
Procesamiento síncrono, 8641
Scripts de usuario, 8638
Lista de gráficos
Bit (0, 1), 8387
Campo E/S gráfico, 8391
Crear, 8383
Emitir datos de configuración, 8390
Número de bit (0 - 31), 8390
Rango (... - ...), 8386
Rango (0 - 31), 8384
Utilización, 8382
Lista de objetos
variable, 8466
Lista de recetas, 8577
Lista de referencias cruzadas, (Véase Lista de referencias cruzadas), (Véase referencias cruzadas)
Clasificar columnas, 3081
Configuración, 3081, 9194
Estructura, 3081, 9194
mostrar, 3083
Mostrar, 9194
Ordenar columnas, 9194
Visión de conjunto, 9194
Vista general, 3083
Vistas, 3081, 9194
Lista de registros, 8577
Lista de saltos, 3514, 3827, 4223, 6316
Lista de textos
Bit (0, 1), 8379
Campo E/S simbólico, 8380, 8451
Crear, 8375
exportar, 9186
importar, 9187
Número de bit (0 - 31), 8380
Utilización, 8374
Valor/rango, 8378
Lista de variables
Direccionar indirectamente, 8483
Listas de funciones
Edición, 8640
Listas de textos
Del sistema, 421
Introducción, 420
Personalizado, 421
uso en registros de receta, 8576
Listas de textos de sistema
Editar, 423
Modificar textos, 424
Listas de textos de usuario
Crear, 422
Editar, 422
Editar rangos de valores y textos, 423
Listas de textos para avisos, 3195
Listín telefónico en TeleService, 11369
Llamada
ABS_CTRL, 7420
ABS_CTRL_451, 7424
ABS_DIAG, 7422
ABS_DIAG_451, 7426
ABS_INIT, 7419
CAM_CTRL, 7454
CAM_CTRL_452, 7457
CAM_DIAG, 7455
CAM_DIAG_452, 7459
CAM_INIT, 7453
CAM_MSRM_452, 7460
CNT_CTL1, 7391
CNT_CTL2, 7393
CNT_CTRL, 7388
CNT2_CTR, 7406
CNT2RDPN, 7405
CNT2WRPN, 7404
DIAG_INF, 7390
DIAG_RD, 7406
EncoderAbsSensorDP, 7357
EncoderCPU314C, 7364
EncoderET200S1Count, 7360
EncoderET200S1SSI, 7355
EncoderFM350, 7362
EncoderFM450, 7352
EncoderIM174, 7366
EncoderSINAMICS, 7369
EncoderSM338, 7358

EncoderUniversal, 7372
High_Speed_Counter, 5123
MC_Control, 7340
MC_GearIn, 7347
MC_Home, 7336
MC_Init, 7325
MC_MoveAbsolute, 7327
MC_MoveJog, 7332
MC_MoveRelative, 7329
MC_Simulation, 7342
MC_StopMotion, 7339
OutputCPU314C, 7376
OutputET200S2AO, 7375
OutputIM174, 7380
OutputIM178, 7373
OutputMM4_DP, 7379
OutputSINAMICS, 7384
OutputSM332, 7377
OutputSM432, 7374
OutputUniversal, 7378
Llamada de alarma, 1209
Llamada de bloque
actualizar, 2821
Actualizar, 2688, 2732, 2779, 2925
Cambiar tipo de llamada, 2689, 2733, 2780
GRAPH, 2842
insertar, 2818, 2819
Insertar, 2687, 2731, 2776, 2778, 2813, 2814, 2816, 2817, 2923, 2924
Instancia individual, 2265
Llamada como instancia individual o multiinstancia, 2264
Multiinstancia, 2266
Principios básicos, 2263
Profundidad de anidamiento, 2264
Llamar, 4238, 4240, 4241, 5763, 5984, 6329, 6330, 6332
LLDP, 769, 770, 1513
LLDP (Link Layer Discovery Protocol), 1513, 8805, 8856, 8909
LLDP (ver Link Layer Discovery Protocol), 1944
LLDP conforme a IEEE, 1944
LMNGEN_C
Instrucción, 7264
Parámetros de entrada, 7268
Parámetros de entrada/salida, 7269
Parámetros de salida, 7268
Variables estáticas, 7269
LMNGEN_S
Instrucción, 7269
Parámetros de entrada, 7275
Parámetros de entrada/salida, 7276
Parámetros de salida, 7275
Variables estáticas, 7276
LN, 3422, 3734, 4149, 4344, 4542, 5690, 5913, 6243, 6427, 6566
LOG_GEO, 5013
LOG2GEO, 5006
LOG2MOD, 5007
Logaritmo, 3422, 3734, 4149, 4344, 4542, 5690, 5913, 6243, 6427, 6566
Lógica de bits
O, 4065, 4066, 4071, 4074
O-exclusiva, 4068, 4069, 4076, 4078
Y, 4062, 4063, 4072, 4073
Logoff, 8647
Logon, 8653
Long word, 2328
Longitud
Puntero de área, 8784
Punteros de área, 8782
Longitud de campos de texto
Idiomas asiáticos , 9291
Longitud del mensaje en el mensaje, 1030
Longitud fija de la trama, 1030
Longitud máxima
Variable, 77
LOOP, 4225, 6317
Loop-In-Alarm, 8722
Configurado, 8520
LP_SCHED
Parámetros de entrada, 7233
LP_ZONE
Instrucción, 7604
LREAL, 2339, 2393, 2441, 2475, 2500
LREAL_TO_, 2441, 2500
LTIME, 2343, 2397, 2446
LTIME_TO_, 2446
LTOD, 2345, 2404, 2454
LTOD_TO_, 2454
LWORD, 2328, 2380, 2418
LWORD_TO_, 2418

M

maestro, 1125
Maestro DP, 897, 2099
Agregar sistema maestro DP, 1398, 1470
Desconexión del sistema maestro DP, 1397, 1470
Dispositivos y módulos, 1394, 1468
Funcionalidad DPV1, 1434
Interfaz DP, 1394, 1468
Visualización en el esclavo DP, 1396, 1469

Maestro Modbus, 7984
Maestro Sync, 1532
Maestro Sync redundante, 1532
maestro Sync secundario, (Véase Maestro Sync redundante)
Maestro USS
Orden de las llamadas de funciones, 1194
Protocolo USS, 1037, 1192
Protocolo USS: Bloque de datos de red, 1193
Protocolo USS: Campo de datos, 1038
Protocolo USS:Cifrado de datos, 1038
Protocolo USS:Codificación de datos, 1192
Protocolo USS:Estructura de la trama, 1038
Protocolo USS:Estructura de telegrama, 1192
Protocolo USS:Procedimiento de transmisión de datos, 1038, 1193
Relación de funciones, 1039
Sinopsis de funciones, 1193
Magnitudes de referencia para Motion Control, 10648
Main, 1675
Main (S7-1500), 1963
Main Mode, 817
Manejar una receta
Crear un registro de receta, 8594, 8595
leer un registro, 8595
transferir un registro, 8596
Manejo
Botón, 9309
Campo de fecha y hora, 9301
Campo E/S, 9302
Campo E/S gráfico, 9305
Campo E/S simbólico, 9311
Indicador de avisos, 8537
Interruptor, 9310
Visor de avisos simple, 8536, 9323
Visor de curvas, 9307
Visor de gráficos, 9304
visor de recetas simple, 8591
Visor de recetas simple, 9332
Visor de usuarios simple, 9315
Manejo automático de las señales cualificadoras, 1025
Manejo con pantalla táctil
Campo E/S, 9303
Manejo con teclado
Campo de fecha y hora, 9301
Campo E/S, 9303
Manejo simultáneo de varias teclas, 80
Manuales, 8759
Mapa de bits
Activar, 3290, 3599, 5772, 5993, 6098, 6493, 6624
Activar en el área de periferia, 5773, 6100, 6494, 6625
Activar en el área de periferia , 5995
Desactivar, 3291, 3600, 5777, 5999, 6104, 6497, 6630
Desactivar en el área de periferia, 5779, 6000, 6105, 6498, 6631
Mapeo del módulo de transferencia, 1524
Máquina virtual (VM)
plataformas de virtualización compatibles, 98
Plataformas de virtualización compatibles, 94
Marca cero, 10655
Marca de referencia, 10654
Marcadores
Activar, 2802
Borrar, 2803
Función, 2801
Navegar, 2802
Marcas, 1912
Activar la indicación de las marcas remanentes, 3062
Marcas de ciclo, 1646, 1669, 1751, 1931, 1955
Marcas de sistema, 1646, 1668, 1931, 1954
Marcas remanentes
Activar indicación, 3062
Marco
Comodines para la información del documento, 404
Definir el área de impresión, 402
Margen alcanzado, 8723
Máscara de bits, 3575, 3891, 4042, 4495, 4659, 5799, 6020, 6131, 6517, 6652
Máscara de subred, 551, 1493
Master Control Relay, 5765, 5767, 5769, 5770, 5985, 5988, 5990, 5992, 6373, 6375, 6377, 6378
Master Control Relay (MCR)
Accesos a parámetros formales, 2566
Indicaciones de manejo, 2566
Transferencia de parámetros, 2566
Matriz, 69, 8494, 8496
crear, 8496
Direccionar indirectamente, 8483, 8494
MAX, 3415, 3727, 3936, 4338, 4536, 5684, 5906, 6063, 6422, 6560
MAX_LEN, 4717
Máximo, 3415, 3727, 3936, 4338, 4536, 5684, 5906, 6063, 6422, 6560
MB, 2571
MB_CLIENT, 5549, 5568

MB_COMM_LOAD, 5532
MB_MASTER, 5535
MB_SERVER, 5557, 5580
MB_SLAVE, 5543
MC_ChangeDynamic
Instrucción, 5049
Parámetro, 5050
Parámetro, 5052
MC_CommandTable
Instrucción, 5047
Parámetro, 5048
MC_Control, 7339
Descripción, 7339
Funcionamiento, 7340
Llamada, 7340
Modo de regulación, 7340
Modo de seguimiento, 7340
Modo manual, 7340
Parámetro, 7341
MC_GearIn, 7342
Descripción, 7342
Funcionamiento, 7342
Llamada, 7347
Parámetro, 7347
MC_Halt, 5115, 5118
Diagrama de funcionamiento, 5031
Instrucción, 5028
Parámetro, 5029, 5066
MC_Home, 5093, 7333
Búsqueda del punto de referencia, 7333
Descripción, 7333
Funcionamiento, 7333
Instrucción, 5025
Llamada, 7336
Parámetro, 5026, 7337
Posicionamiento del punto de referencia, 7334
MC_Init, 7324
Descripción, 7324
Funcionamiento, 7324
Llamada, 7325
MC_MoveAbsolute, 5111, 5114, 7325
Parámetro, 5033
Descripción, 7325
Diagrama de funcionamiento, 5034
Funcionamiento, 7325
Instrucción, 5032
Llamada, 7327
Parámetro, 7327
MC_MoveJog, 5098, 5101, 7330
Descripción, 7330
Diagrama de funcionamiento, 5046
Funcionamiento, 7330
Instrucción, 5044
Llamada, 7332
Parámetro, 5045, 7332
MC_MoveRelative, 5107, 5110, 7328
Descripción, 7328
Diagrama de funcionamiento, 5038
Funcionamiento, 7328
Instrucción, 5035
Llamada, 7329
Parámetro, 5036, 7329
MC_MoveVelocity, 5102, 5106
Diagrama de funcionamiento, 5043
Instrucción, 5039
MC_Power, 5088, 5092
Diagrama de funcionamiento, 5022
Instrucción, 5019
Parámetro, 5019
MC_Reset, 5119
MC_Simulation, 7341
Descripción, 7341
Funcionamiento, 7341
Llamada, 7342
MC_StopMotion, 7337
Cancelación de un desplazamiento, 7338
Descripción, 7337
Funcionamiento, 7338
Llamada, 7339
Parámetro, 7339
MCAT, 3571, 3886, 4038, 4491, 4654, 5795, 6016, 6127, 6513, 6648
MC-Interpolator, 1977, 10682
MCR, 5765, 5767, 5769, 5770, 5985, 5988, 5990, 5992, 6373, 6375, 6377, 6378
MCRA, 5769, 5990, 6377
MCRD, 5770, 5992, 6378
MC-Servo, 1976, 10682
Parametrizar, 1993
MD, 2571
MD5, 812, 818
MDM, 336
Mecanismo EN/ENO
Ejemplo en AWL, 2544
Ejemplo en FUP, 2543
Ejemplo en KOP, 2542
Ejemplo en SCL, 2543
Principios básicos, 2539
Mecanismos de conexión, 1674, 1959
Medición
Frecuencia, 9478
Período, 9481
valor de comparación, 9443
Velocidad, 9480

Medición de frecuencia, 9389, 9478
Medición de frecuencia (CPU 31xC), 1765
Medición de frecuencia con FREQUENC, 7654
Medición de longitud, 11027
Datos en el DB de canal , 11029
Desarrollo, 11028
Distancia de flanco, 11029
Errónea, 11029
Requisitos, 11027
Medición de recorrido
Easy Motion Control, 11265
Medición de velocidad, 9389, 9480
Easy Motion Control, 11264
Medición del período, 9389, 9481
Medida al vuelo, 11209
Medidas de seguridad para el forzado permanente de variables, 3144, 3162
Medio de almacenamiento extraíble, 1513
Medio de transferencia/dúplex, 1510
Medir tiempo de ejecución, 3536, 3850, 4019, 4465
Medir tiempos, 9517
entre dos flancos, 9517
Memoria, 1910, 1913
Área de comandos, 2243
Comprimir, 2243, 5731, 5954, 6083, 6470, 6600
Copiar de RAM a ROM, 2243
Memoria de carga, 1630, 1718, 1722, 1910, 3076
Mostrar, 3080
Memoria de carga máxima disponible
Mostrar, 3080
Memoria de sistema, 1718
Acceso a través de FETCH/WRITE, 676, 2031
Áreas de operandos, 1631, 1723
Búfer de diagnóstico, 1635, 1730, 2204
Memoria imagen de proceso de las entradas y salidas, 1633, 1725
Pila de bloques, 1729
Pila de datos locales, 1726
Pila de interrupciones, 1728
Memoria de trabajo, 1631, 1718, 1722, 1910, 3076
Memoria imagen de proceso, 924, 925, 927, 1636, 1916
Actualizar, 1634, 1725
Actualizar entradas con UPDAT_PI, 4738, 6729
de las entradas y salidas, 1809
De las entradas, sincronizar con SYNC_PI, 4742, 6732
De las salidas, actualizar con UPDAT_PO, 4740, 6731
De las salidas, sincronizar con SYNC_PO, 4744, 6734
Error de acceso a periferia, 1634, 1725
Escribir salidas con SETIO, 4750, 6741
Leer entradas con GETIO, 4749, 6740
Leer un área de la memoria imagen de proceso con GETIO_PART, 4751, 6742
Principios básicos, 1633, 1725
Transferir un área de la memoria imagen de proceso con SETIO_PART, 4752, 6743
Memoria imagen parcial de proceso, 925, 927, 928, 1454
Memoria imagen parcial de proceso "IPP OB Servo", 10684
Memoria necesaria
Receta, 9349
Memoria remanente, 1719, 1910, 3076
Memoria sombra, 8263
Memory Card, 33, 1628, 1716, 1907, 1910
Acceder, 425
Agregar lector de tarjetas, 425
Borrar Data Log en el S7-1500, 38
Desenchufar/enchufar, 38
Formatear, 2194
Introducción, 424
Mostrar las propiedades, 426
Memory Card de actualización de firmware, 1629, 1908
Memory Card SIMATIC, 1910
Formatear, 2194
Memory Cards, 34
Mensaje
Definir el fin, 1667
Definir principio, 1666
enviar, 1665
Mensaje de error
Carga, 9275
Cargar datos en el PLC, 75
Mensajes de error del módem
archivo de informe, 11427
Métodos de autenticación, 815, 816
MIB, 769
Micro, 9079
Micro Memory Card, 1714
MicroLogix, 9006, 9029
MID, 4732
Migración, 33, 136, 139, 141, 195, 197, 203, 205, 211, 216, 235, 237
Adaptaciones previas, 152
Ajustes de escritura, 154
de datos runtime, 162
de grupos de avisos, 156
de idiomas, 159
de la administración de usuarios, 162
de los datos de recetas, 162

de textos, 159
de textos del proyecto, 159
de variables, 155
de variables externas, 169
Desarrollo de la migración, 120
Inclusión de la configuración hardware, 120
Introducción, 146
Introducción a la migración, 119
Librería del proyecto, 161
Librería global, 161
Migración de programas PLC a S7-1500, 185, 186, 188, 189, 208, 209, 210, 218, 219, 221, 222, 223, 224, 226, 227, 229, 230, 232, 240, 241, 243
Migrar proyecto integrado, 165, 177
Migrar proyectos, 149, 167
Mostrar archivo de registro, 126
Mostrar histórico, 125
paneles de operador soportados, 151
Principios básicos, 147
productos soportados, 119
Proyecto integrado, 177
Tipos de datos Allen-Bradley Ethernet IP, 170
Tipos de datos de Allen-Bradley DF1, 170
Tipos de datos de Mitsubishi FX, 171
Tipos de datos Modicon Modbus, 171
Tipos de datos Modicon Modbus TCP/IP, 172
Tipos de datos Omron Hostlink/Multilink, 172
Tipos de datos SIMATIC S7 200, 173
Tipos de datos SIMATIC S7 300/400, 173
WinCC V7.0 SP3, 62
Migración de la lista de protección de acceso IP al activar la seguridad, 44
Migración de PLC
Perfil de instrucción, 38
Migración de programas PLC a S7-1500, 195, 197, 203, 205, 211, 216, 235, 237
Migración de proyectos con CP Ethernet y seguridad informática, 44
Migración de proyectos STEP 7
Áreas de transferencia, 132
Avisos, 135
Avisos de error de sistema, 135
CBA con controladores IO, 132
Dispositivos GSD y GSDML, 133
Grupos de actualización PROFINET IO, 133
Introducción, 126
I-Slaves e I-Devices, 133
Modo isócrono con PROFINET IO, 132
Modo isócrono PROFIBUS, 133
Multiproyecto, 134
Nombres de los dispositivos PROFINET IO, 131
Nombres de módulos y racks, 131
Redundancia de medios, 132
Requisitos, 127
Requisitos que debe cumplir el proyecto, 128
Shared Devices, 132
Tipos de conexiones que se pueden migrar, 134
Volumen de migración, 126
Migrar proyectos
Procedimiento, 124
Requisitos, 124
MIN, 3413, 3724, 3934, 4336, 4535, 5682, 5904, 6061, 6421, 6558
Mínimo, 3413, 3724, 3934, 4336, 4535, 5682, 5904, 6061, 6421, 6558
MIP OB 1 (memoria imagen de proceso del OB 1), 1809
MIPP, (Véase Memoria imagen parcial de proceso)
Mirroring, 1316, 1318
Mitsubishi, 8997
Aviso analógico, 9067
Basic Panel, 8788
Dirección, 9052
Drivers de comunicación, 9040
FX, 9040
Interrupción de la conexión, 72
TCP/IP, 9040
Mitsubishi FX
Conexión, 9055, 9059
Configurar una conexión, 9055
Migración de los tipos de datos, 171
Mitsubishi MC TCP/IP, 9046, 9058
Parámetros de conexión, 9056
Tipo de CPU, 9061
Tipo de datos, 9060
Mitsubishi MC TCP/IP
Conexión, 9041, 9045
Configurar una conexión, 9041
Parámetros de conexión, 9043
Tipo de CPU, 9051
Tipo de datos, 9050
MOD, 2788, 3406, 3717, 4139, 5678, 5900, 6233
MODB_341, 1072
MODB_441, 1072
Modbus
Exception Code, 1042
Fin de la trama, 1041
Modbus_Comm_Load, 5465, 7856
Modbus_Slave, 5468, 5474, 7859, 7865
Señales RS232, 1025
Modbus_Comm_Load, 1014, 1052, 5465, 7856
Modbus_Master, 1014, 1052
Modbus_Slave, 1014, 1052, 5468, 5474, 7859, 7865
Modelo de espacio de memoria, 2848, 2869

Modelo de slot en DPV1, 1440
Módem
local, 11380
problemas, 11422
remoto, 11381
Módems no aptos para plug&play, 11381
Modicon
Cable de conexión, 9087
Conexión, 9087
Limitaciones con Modbus RTU, 9088
tipos de comunicación autorizados con Modbus RTU, 9087
Modicon M340, 9079
Modicon Modbus, 8997
Basic Panel, 8788
Drivers de comunicación, 9069
Migración de los tipos de datos, 171
RTU, 9069
TCP, 9069
Modicon Modbus RTU, 9093
Conexión, 9082
Configurar una conexión, 9082
Parámetros de conexión, 9084
Tipo de datos, 9092
Modicon Modbus TCP, 9079
Conexión, 9070, 9075
Configurar una conexión, 9070
Parámetros de conexión, 9072
Tipo de datos, 9078
Modicon Modbus TCP/IP
Migración de los tipos de datos, 172
Modicon MODBUS TCP/IP
Change word order, 9074
Modificación, 8715
Datos máquina, 10901, 10995
Tablas de incrementos, 10903, 10997
Modificación de software, 1915
Modificación de una interconexión de puertos
Vista gráfica, 743
Modificación de valor, 8726
Modificación online de la configuración, 11298
Modificación online de la configuración del dispositivo, 11298
Modificar
Contraseña, 8611
Grupo de usuarios en runtime, 8620
Nombre del usuario, 8620
Tamaño del objeto, 8334
Tiempo de cierre de sesión en runtime, 8620
Usuarios, 9318
Modificar conexiones a través de cuadros de diálogo, 635
Modificar el interlocutor, 634
Modificar flancos de leva, 9764, 9842
Datos utilizados en el DB de canal, 9765, 9843
Definición, 9764, 9842
Desarrollo, 9764, 9842
Deshacer, 9766, 9844
Efectos, 9765, 9843
Requisitos, 9764, 9842
Modificar la vía de conexión en la ventana de inspección, 636
Modificar la vía de una conexión configurada, 635
ModificarContraste, 8648
Modo, 8449
Campo E/S, 8428
Campo E/S gráfico, 8433
Modo automático del controlador PROFINET IO, 2063
Modo avanzado, 3116
Reglas de cortafuegos globales, 774
Modo básico, 3116
Modo combinado, 819
Modo conforme 611U, 11206
Modo de adquisición
Variable, 8479
Modo de aprendizaje, 867
Modo de configuración, 1264
Modo de control 1, 10938, 11039
Modo de control 2, 10939, 11040
Modo de control 3, 10940, 11041
Modo de control 4, 10941, 11042
Modo de demostración
Engineering System, 101
Paneles de operador, 103
Sistema de ingeniería, 89
Modo de direccionamiento S5, 2033
Modo de direccionamiento S7, 2032
Modo de enrutamiento, 797
Modo de etapa individual
Activar puntos de parada, 3092
Ajustar el entorno de llamada para puntos de parada, 3091
Borrar puntos de parada, 3096
Continuar la ejecución del programa, 3094
Continuar la ejecución del programa hasta la marca, 3094
Desactivar puntos de parada, 3092
Ejecutar llamada, 3094
Ejecutar siguiente instrucción, 3094
Finalizar el test, 3096
Navegar hasta los puntos de parada, 3091
Posicionar puntos de parada, 3090
Principios básicos, 3089

Modo de forzado, 3131
Modo de observación, 3131
Modo de observación y forzado, 3131
Modo de operación, 9431
Búsqueda del punto de referencia, 10914, 11008
Contaje periódico, 9470
Contaje sin fin, 9468
Contaje único, 9469
Dosificación, 9484
JOG, 10911, 11005
Medición, 9437
Medición de frecuencia, 9478
Medición de velocidad, 9480
Medición del período, 9481
Modo incremental, 10920, 11014
Para el test de programas GRAPH, 3102
Test en modo automático, 3106
Test en modo manual, 3106
Test en modo semiautomático, 3106
Modo de paleta, 303
Modo de proceso, 1757
Modo de prueba, 1757
Modo de referenciado, 10655
Modo dúplex, 1015, 1075
Modo incremental, 10920, 11014
absoluta, 11014
absoluto, 10920
Cancelar, 10925, 11020
Datos en el DB de canal, 10924, 10925, 11019, 11021
Datos en el DB de parámetros, 10924, 11019
Desactivación, 10925, 11020
Número de incremento 1-100:, 10921, 11015
Número de incremento 252 , 11016
Número de incremento 254 , 10922, 11017
Número de incremento 255, 10922, 11017
relativo, 10920, 11014
Requisitos, 10921, 11015
Modo incremental con número de incremento 252
Desarrollo, 11016
Modo incremental con número de incremento 254
Desarrollo, 10922, 11017
Modo incremental con número de incremento 255
Desarrollo, 10922, 11017
Modo isócrono, 1540, 1611, 1617, 9405
¿Por qué?, 1534
Ajustes en el esclavo DP, 1457
configuración en la subred, 1543
configurar, 1541, 1542
Configurar, 1453
Efecto de Ti, 1537
Efecto de To, 1538
Memorias imagen parciales de proceso, 1454
Mismos tiempos Ti/To para todos los esclavos, 1458
no seleccionable, 1456
Proceso, 1535
Propiedades de la CPU, 1454
Propiedades en PROFIBUS, 1456
Reglas, requisitos y condiciones marco, 1451, 1544
Reservas, 1462
Solapamiento de Ti y To, 1461
Tiempo de ciclo DP y valores Ti/To, 1456
Tiempo de retardo, 1455
Modo isócrono , 1446
Modo multiplex/síncrono, 2148
Modo online, 11299
Deshacer la conexión online, 11303
Establecer conexión online, 11302
Modo de espera o estado de reposo de la PG/el PC, 11300
Mostrar los dispositivos accesibles desde una interfaz, 11297
Mostrar todos los dispositivos accesibles en una lista, 11297
Visualización del modo online, 11301
Modo PROFIdrive, 2069
Modo RS232, 1018
Modo RS422, 1019
Modo RS485, 1020
Modo semidúplex, 1015, 1075
Modo VLAN, 816
Modos de conexión, 1209
Modulación del ancho de impulso (CPU 31xC), 1769
Modulación por ancho de impulso con PULSE, 7658
ModuleStates, 4951
Módulo, 10645
Información de estado de módulo, mediante lista de estado del sistema, 7002
borrar, 525
Clase de módulo, 6973
copiar, 526
Datos de diagnóstico, 6912
Desenchufar, 39
desplazar, 528
Determinar dirección lógica con GADR_LGC, 5016, 7073
Determinar direcciones lógicas con RD_LGADR, 5014, 7072
Determinar el estado de diagnóstico, 2180
Determinar el slot de una dirección lógica con LGC_GADR, 5017, 7075

Determinar la dirección inicial con GEO_LOG, 5012, 7068
Determinar la dirección lógica del slot con LOG_GEO, 5013, 7070
direccionar, 914, 915, 971
Enchufar, 39
Final de carrera por software, 10677
Hora de un módulo, 2189
Identificación a través de la lista de estado del sistema, 6975
Información de diagnóstico, lista de estado del sistema, 7020
insertar, 524
Leer registro con RD_DPAR, 4875, 6874
Leer registro de forma asíncrona con RD_DPARA, 4878, 6876
Parametrizar con PARM_MOD, 6877
Parametrizar y configurar, 38
Propiedades de comunicación, 2232
seleccionar, 515
sustituir, 529
Módulo analógico
Restablecimiento de la configuración de fábrica, 2204
Módulo comodín, 899
Módulo de bus activo (ET 200M), 2139
Módulo de comunicación (CM), 5520
Módulo de comunicaciones (CM)
Programar, 5394, 7784
Recepción de datos, 5413, 7803
Módulo de interfaz, 910
Módulo de reserva
Valor sustitutivo, 1410, 1480
Módulo de seguridad, 751
Módulo emisor, 910
Módulo no enchufado, 516
Módulo receptor, 910
Módulo tecnológico
aplicaciones posibles, 9358
Características, 9357
configuración de hardware, 9428
Interfaz de control, 9459
Interfaz de realimentación, 9461
Online y diagnóstico, 9455
Parametrizar, 9436
parámetros básicos, 9429
Módulos
Estado de la comunicación, 834
Módulos centrales
Parámetros en el programa del usuario, 910
Propiedades de las CPUs, 909
Selección del catálogo de hardware, 906
Sinopsis de la configuración, 898
Módulos de comunicación, 1659
Propiedades, 1659
Módulos de fuente de alimentación redundantes, 900
Módulos de fuente de alimentación, redundantes, 900
Módulos de seguridad
Selección del catálogo de hardware, 908
Módulos del S7-1200, 40
Módulos electrónicos y de pines, 2098
Momentum, 9079, 9093
Mostrar
Estructura de llamadas , 3067
Carga de la memoria de la CPU, 3079
Copia de seguridad HMI, 9252
de referencias cruzadas, 3083, 9194
Estructura de dependencias, 3073
Información del programa, 3053
Instrucciones de comunicación para conexiones S7 no configuradas, 7662
Líneas de valores límite en la barra, 8424
Memoria de carga, 3080
Memoria de carga máxima disponible, 3080
Plano de ocupación , 3057
Referencia cruzada, 9196
Tooltip, 9298
Mostrar archivos PDF, 87
Mostrar avisos, 2153
Mostrar el soporte para un módulo, 506
Mostrar estado online, 11303
Mostrar hora del sistema, 8427
Mostrar la Welcome Tour, 87
Mostrar los tipos de memoria de una CPU, 2178, 2179
Mostrar u ocultar informaciones de variables, 2703, 2747, 2773, 2809, 2921
Mostrar y ocultar columnas, 382, 2591, 2666, 2960, 2976
Mostrar y ocultar operandos absolutos, 2799
Mostrar/ocultar interfaces, 11311
MostrarDiálogoInicioSesión, 8707
MostrarTextoDeAyuda, 8707
MostrarVentanaDeAvisos, 8708
Motion Control
Eje de posicionamiento, 10643
Eje de velocidad de giro, 10642
Eje lineal, 10644
Eje rotativo, 10644
Encóder externo, 10644
Funcionamiento, 10638
Sistema de ejecución, 10682

Motion Control CPU S7-1200
Manual, 10441
Motor paso a paso, 10429
MOVE, 258, 3436, 3749, 3940, 5702, 5925
MOVE_BLK, 3453, 3766, 3951, 4361, 4558
MOVE_BLK_VARIANT, 3455, 3768, 3953, 4363, 4560
MoveVelocity
Parámetro, 5040
Movimiento diagonal
Animación, 8397
Movimiento vertical
Animación, 8397
MPI, 8760
Arquitectura de red, 8763
Asignación de direcciones, 915
Conexión, 8948
Conexión HMI, 8919
Direccionamiento, 8925
Parámetros, 8921, 8923, 8924
Red, 8762
S7 200, 8955, 8965
S7 300/400, 8918, 8919
MRP (Media Redundancy Protocol), 1567, 1569
MSI, 1918
MSI (Shared Input interna del módulo), 1919
MSI/MSO, 1593
MSK_FLT, 4908
MSO, 1918
MSTP, 1330
MUL, 3403, 3714, 5675, 5897
Multicast, 639, 685, 799, 1342
Con UDP, 671, 672
Multiinstancia, 2297
Cambiar tipo de llamada, 2689, 2733, 2780
Declarar, 2655
Definición, 2266
Multiple Spanning Tree, 1330
Multiplexado
con direcciones absolutas, 8486
con direcciones simbólicas, 8487
Multiplexado de direcciones, 8486
Multiplexado de direcciones
con direcciones absolutas, 8486
con direcciones simbólicas, 8487
Multiplexar, 3550, 3864, 4471, 4630, 6482, 8483
Multiplexar direcciones con Allen-Bradley Ethernet IP, 9010
Multiplicar, 3403, 3714, 4128, 4133, 4142, 5675, 5897, 6222, 6227, 6237
Multipunto, 1125
MUX, 3550, 3864, 4471, 4630, 6482
MW, 2571
MX, 2571
My Documentation Manager, 336

N

N, 3296, 3299, 3606, 3608, 5613, 5834
N_TRIG, 3302, 3611, 5616, 5836
NAK, 1034
NAT/NAPT
Routing, 798
Navegación general, 492, 495, 497, 8745, 8752, 8754
NE_ElemType, 3392, 3703
NE_Type, 3389, 3701
NEG, 3407, 3719, 4532, 5679, 5902, 6556
Negar, 3407, 3719, 4186, 4187, 4188, 4532, 5679, 5902, 6279, 6280, 6282, 6556
NEGD, 4187, 6280
NEGI, 4186, 6279
Negotiation, 1290
NEGR, 4188, 6282
Nivel
Asignar un objeto a un nivel, 8418
Nivel de protección, 1670, 1671, 1756, 1955
Borrar derechos de acceso, 389
Modificar, 5735, 5957, 6087, 6474, 6604
No integrada
Conexión, 8769
No se establece la conexión módem, 11427
Nodos de red internos
Configurar, 822
Diagnóstico, 834
Nombre
Elemento de matriz, 68
Nombre convertido (PROFINET), 1498
Nombre de dispositivo, 1491, 1498, 1606
Nombre de dispositivo PROFINET, 1498, 1606, (Véase Display)
Nombre de dispositivo, puesta en servicio automática (PROFINET), 1494
Nombre de estación, (Véase Nombre de dispositivo)
Nombre de evento, 1693
Nombre de usuario, 765
Nombre de visualización
Punteros de área, 8782
Nombre del dispositivo
PROFINET, 1518
Nombre del grupo, 778, 781
Nombre del rol, 767
Nombres de categorías
Cambio mediante la migración, 157

Nombres de objeto
Univocidad, 147
NONLIN
Instrucción, 7277
Parámetros de entrada, 7279
Parámetros de salida, 7279
NOP 0, 4281, 6381
NOP 1, 4282, 6382
NORM
Instrucción, 7280
Parámetros de entrada, 7281
Parámetros de salida, 7281
NORM_X, 3503, 3816, 3989, 4424, 4601
Normalizar, 3503, 3816, 3989, 4424, 4601
NOT, 2792, 2793, 3286, 4084, 4625, 5604, 6180, 6610
NOT_NULL, 3394, 3706
NOT_OK, 3387, 3699
Notificar errores de sistema, (Véase Diagnóstico del sistema)
NTP, 1342
NTP (seguro), 809
Nuevos tipos de datos, 251
NULL, 2359
Número de aviso, 8510
Asignar, 3172
Número de bit (0 - 31)
Lista de gráficos , 8390
Lista de textos, 8380
Número de dispositivo, 1493
PROFINET, 1518
Número de imagen, 8741
Número de incremento 1 a 100, 10965, 11066
Número de incremento 1-100, 10921, 11015
Número de incremento 252, 11016, 11067
Número de incremento 254, 10922, 10966, 11017, 11067
Número de incremento 255, 10922, 10966, 11017, 11068
Número de serie, 1240
Número en coma flotante, 2338, 2339
Comprobar invalidez, 3387, 3699
Comprobar validez, 3386, 3698
Número máximo de caracteres, 1030
Números de puerto, 716
Números en coma flotante, 2340, 2552
no válidos, 2340

O

O, 3590, 3591, 4065, 4071, 5821, 5822, 6161, 6167, 11300
O(, 4074, 6170
OB
Eventos y OB, 1636, 1924
fuente de eventos, 1925
Resumen, 1636, 1924
OB 1, 1675, 1822
OB 10 a OB 17 (S7-300, S7-400), 1823
OB 100
OB 101 y OB 102 (S7-300, S7-400), 1875
OB 121 (S7-300, S7-400), 1880
OB 122 (S7-300, S7-400), 1883
OB 20 a OB 23 (S7-300, S7-400), 1827
OB 30 a OB 38 (S7-300, S7-400), 1828
OB 40 a OB 47 (S7-300, S7-400), 1830
OB 61 (S7-300, S7-400), 1844
OB 61...64, 1449
OB 62 (S7-300, S7-400), 1844
OB 63 (S7-300, S7-400), 1844
OB 64 (S7-300, S7-400), 1844
OB 80, 1686
OB 80 (S7-300, S7-400), 1845
OB 81 (S7-300, S7-400), 1848
OB 82, 1687
Alarma de diagnóstico, 10280, 10332, 10365
OB 82 (S7-300, S7-400), 1851
OB 83, 1688
OB 83 (S7-300, S7-400), 1854
OB 84 (S7-300, S7-400), 1858
OB 85, 1726
OB 85 (S7-300, S7-400), 1859
OB 86 (S7-300, S7-400), 1864
OB 87 (S7-300, S7-400), 1871
OB 88 (S7-300, S7-400), 1872
OB de alarma cíclica
Ajustar parámetros con SET_CINT, 4886
Consultar parámetros con QRY_CINT, 4888
Descripción, 1683
parametrizar, 1989
Parametrizar, 1692
OB de alarma cíclica (S7-1500), 1972
OB de alarma de actualización, 1681
OB de alarma de actualización (OB 56) (S7-300, S7-400), 1836
OB de alarma de actualización (S7-1500), 1969
OB de alarma de diagnóstico, 1687
OB de alarma de diagnóstico (OB 82) (S7-300, S7-400), 1851
OB de alarma de diagnóstico (S7-1500), 1979
OB de alarma de estado, 1680
OB de alarma de estado (OB 55) (S7-300, S7-400), 1832
OB de alarma de estado (S7-1500), 1968

OB de alarma de fabricante, 1681
OB de alarma de perfil, 1681
OB de alarma de proceso
Descripción, 1684
parametrizar, 1990
Parametrizar, 1693
OB de alarma de proceso (S7-1500), 1973
OB de alarma de retardo, 1682
OB de alarma de retardo (S7-1500), 1971
OB de alarma de sincronismo, 1449
OB 61 a OB 64 (S7-300, S7-400), 1844
parametrizar, 1991
OB de alarma de sincronismo (S7-1500), 1975
OB de alarma específica del fabricante (S7-1500), 1970
OB de alarma específica del perfil (S7-1500), 1970
OB de alarma horaria
parametrizar, 1988
Parametrizar, 1691
OB de alarma horaria (S7-1500), 1964
OB de alarmas específicas del fabricante (OB 57) (S7-300, S7-400), 1840
OB de arranque
Descripción, 1675
OB de arranque (S7-1500), 1962
OB de cambio de módulo, 1688
OB de cambio de módulo (S7-1500), 1981
OB de cambio de módulo (S7-300, S7-400), 1854
OB de ciclo
Descripción, 1675
OB de ciclo (S7-1500), 1963
OB de error de acceso a la periferia, 48
OB de error de acceso a la periferia (S7-1500), 1986
OB de error de acceso a la periferia (S7-300, S7-400), 1883
OB de error de comunicación (OB 87) (S7-300, S7-400), 1871
OB de error de ejecución del programa (OB 85) (S7-300, S7-400), 1859
OB de error de fuente de alimentación (OB 81) (S7-300, S7-400), 1848
OB de error de hardware en CPU (OB 84) (S7-300, S7-400), 1858
OB de error de programación (S7-1500), 1983
OB de error de programación (S7-300, S7-400), 1880
OB de error de tiempo, 1686
mecanismo de valores umbrales, 1927
OB de error de tiempo (OB 80) (S7-300, S7-400), 1845
OB de error de tiempo (S7-1500), 1978
OB de fallo de rack, 1689
OB de fallo de rack (S7-1500), 1982
OB de fallo del rack (OB 86) (S7-300, S7-400), 1864
OB de interpolador MC, 1977
OB de interrupción del procesamiento (OB 88) (S7-300, S7-400), 1872
OB de tarea no prioritaria (OB 90) (S7-300, S7-400), 1874
OB MC-Servo, 1976
Objetivo, 10889
Objeto
Agrupar, 8362
Alinear uniformemente, 8338
Asignar a un nivel, 8418
Borrar, 8331
Crear un objeto OLE, 8329
Crear un objeto OLE de un archivo, 8329
Del mismo tipo, 8328
Depositar en la librería, 9151
Dinamizar, 8391
Dinamizar la apariencia, 8395
Dinamizar movimiento directo, 8398
Diseñar, 8343
Diseñar color de fondo, 8344
Diseñar patrón de relleno, 8345
Disponer, 8328, 8339
Disponibilidad para Basic Panel, 8423
Editar dentro de un grupo, 8365
Fuera del área, 8340
Girar, 8328, 8340
Insertar, 8328, 8329
Insertar varios, 8353
Justificar, 8337
Modificar la posición y el tamaño de varios objetos, 8329, 8355
Modificar posición, 8332
Modificar tamaño, 8334
Orden de tabulación, 8328
Reproducir, 8353
Seleccionar varios, 8328, 8335
Transparencia, 8345
Voltear, 8328
Objeto de imagen
Dinamizar el movimiento, 8396
Objeto de indicación
Disponibilidad para Basic Panel, 8423
Objeto de librería, 9134, 9135
Insertar, 9152
Objeto de manejo
Disponibilidad para Basic Panel, 8423
Objeto de referencia
Definir, 8336

Objeto OLE
Agregar a los gráficos, 8358
agregar al juego de gráficos , 9215
Crear, 8329
Crear de un archivo, 8329
Objeto tecnológico
Eje de posicionamiento, 10640, 10697
Eje de velocidad de giro, 10640, 10642, 10729
Encóder externo, 10640, 10737
High_Speed_Counter, 9406
Objeto tecnológico Eje de posicionamiento, 10643
Configuración de accionamiento listo, 10459
Configuración de la habilitación del accionamiento, 10459
Configuración de la referenciación pasiva, 10472
Configuración de la unidad de usuario, 10458
Configuración de la velocidad de referenciación, 10475
Configuración de la velocidad máxima/velocidad de arranque y parada, 10466
Configuración de PTO y HSC, 10456
Configuración del lado del sensor del punto de referencia, 10473
Configuración del recorrido por vuelta del motor, 10459
Configuración del sentido de aproximación/ referenciación, 10475
Actualización de las variables, 10588
Agregar objeto, 10454
Componentes de hardware y software, 10449
Comportamiento con la limitación de tirones activada, 10470
Configuración de la aceleración, 10466
Configuración de la deceleración, 10466
Configuración de la deceleración de parada de emergencia, 10468
Configuración de la dinámica general, 10465
Configuración de la entrada del sensor del punto de referencia, 10473, 10474
Configuración de la interfaz del hardware, 10456
Configuración de la inversión de sentido, 10460
Configuración de la limitación de tirones, 10467
Configuración de la referenciación activa, 10473
Configuración de la unidad del límite de velocidad, 10465
Configuración de las señales del accionamiento, 10459
Configuración de los impulsos por vuelta del motor, 10459
Configuración del lado del sensor del punto de referencia, 10475
Configuración del mecanismo, 10459
Configuración del nombre del eje, 10456
Configuración del offset del punto de referencia, 10475
Configuración del permiso de inversión del sentido en el final de carrera por hardware, 10474
Configuración del sensor del punto de referencia, 10473, 10475
Configuración del tiempo de aceleración, 10466
Configuración del tiempo de deceleración, 10466
Configuración del tiempo de redondeo, 10467
Herramientas, 10452
Modificar los parámetros de configuración de la dinámica en el programa de usuario, 10471
Modificar los parámetros de configuración del referenciado en el programa de usuario, 10478
Parámetros avanzados, 10455
Parámetros básicos, 10455
Procedimiento de referenciado activo, 10477
Procedimiento de referenciado pasivo, 10476
Resumen de la configuración, 10453
Resumen de la puesta en servicio, 10453
Resumen del diagnóstico, 10453
Símbolos de la ventana de configuración, 10455
Variable Mechanics, 10576
Variable Position, 10570
Variable Units, 10576
Variable Velocity, 10571
Variables ActiveHoming, 10573
Variables Actor, 10571
Variables DynamicDefaults, 10578
Variables DynamicLimits, 10577
Variables ErrorBits, 10587
Variables Homing, 10581
Variables PassiveHoming, 10574
Variables PositionLimitsHW, 10580
Variables PositionLimitsSW, 10579
Variables StatusBits, 10583
Variables StatusPositioning, 10582
Objeto tecnológico Eje y tabla de peticiones
Lista de los ErrorIDs y las ErrorInfos, 10560
Objeto tecnológico Encóder externo, 10644
Objeto tecnológico Tabla de peticiones
Herramientas, 10480
Agregar objeto, 10481
Configuración de la activación de advertencias, 10484
Configuración de la duración, 10486
Configuración de la tabla de peticiones, 10483
Configuración de la velocidad, 10485
Configuración de posición/recorrido de desplazamiento, 10485

Configuración de Utilizar parámetros del eje de, 10484
Configuración del código de paso, 10486
Configuración del paso siguiente, 10486
Configuración del tipo de petición, 10484
Configuración general, 10483
Parámetros avanzados, 10481
Parámetros básicos, 10481
Símbolos de la ventana de configuración, 10482
Uso, 10479
Variables Command.Command[1...32], 10589
ObjetoDeImagenAvanzarPágina, 8656
ObjetoDeImagenCursorAbajo, 8655
ObjetoDeImagenCursorArriba, 8655
ObjetoDeImagenRetrocederPágina, 8656
Objetos enlazados
Copiar, 9241
Objetos tecnológicos
CONT_C, 10029
CONT_S, 10034
PID_3Step, 9994
PID_Compact, 9962
PID_CP, 10062
PID_ES, 10113
TCONT_CP, 10037
TCONT_S, 10058
OBs de arranque OB 100, OB 101 y OB 102 (S7-300, S7-400), 1875
Observar
Comando "Observar inmediatamente", 3134, 3159
Comando "Observar todos", 3133, 3159
Obtener derechos de usuario
Iniciar sesión con derechos de administrador en el sistema operativo, 326
Mediante el Control de cuentas de usuario de Windows, 326
Ocupación estándar, 1125
Ocupación estándar de la línea de recepción, 1020, 1123
OD, 4251, 6341
O-exclusiva, 3591, 3593, 5823, 5824
Operaciones lógicas con bits, 3591, 5823
OF, 4438, 6458
Offline, 752
OK, 3386, 3698
Omron, 8997
Aviso analógico, 9114
Basic Panel, 8788
Drivers de comunicación, 9099
Hostlink, 9099
Omron Host Link, 9109
Cable de conexión, 9104
Conexión, 9104
Dirección, 9110
Parámetros de conexión, 9102
Tipo de datos, 9108
Omron Hostlink
Conexión, 9100
Configurar una conexión, 9100
Omron Hostlink/Multilink
Migración de los tipos de datos, 172
ON, 4066, 6162
ON(, 4075, 6171
Online, 31, 752
Detección de hardware, 522
OP 73
Velocidad de transferencia en PROFIBUS, 8839, 8894, 8953, 8955, 8963, 8965, 8966
OP 77A
Velocidad de transferencia en PROFIBUS, 8839, 8894, 8953, 8955, 8963, 8965, 8966
OP73
Cargar un proyecto, 76
OP77A
Cargar un proyecto, 76
OPC
Basic Panel, 8788
Opciones de puerto, 1512, 8804, 8855, 8908
Activar autonegotiation, 1511, 8803, 8854, 8907
medio de transferencia/dúplex, 1510
Velocidad de transferencia / dúplex, 8803, 8854, 8907
Vigilar, 1511, 8803, 8854, 8907
Opciones de visualización
configurar para la estructura de dependencias, 3073
configurar para la estructura de llamada, 3068
para el plano de ocupación, 3058
Open User Communication, 51, 615, 7725
Enviar datos con TSEND, 7753
Modificar valores de parámetros, 717
Borrar conexión, 707
Capacidad de relectura, 718
Crear conexión, 704
Descripción de la conexión, 711, 713, 714
deshacer el punto de acceso local de comunicación con TDISCON, 7751
Deshacer una conexión con TDISCON, 5305
Enviar datos a través de UDP con TUSEND, 5323, 7760
Enviar datos con TSEND, 5307, 5310
Establecer conexión con TCON, 5298, 5301

Establecer conexión y enviar datos con TSEND_C, 5263, 5267
Establecer conexión y leer datos con TRCV_C, 5274, 5279
Establecimiento de la conexión, 697
Estructura de la información de dirección del interlocutor remoto en UDP, 7737
Funcionamiento de las instrucciones, 7726
FW_IOT, 7776
FW_TCP, 7772
General, 696
Iniciar la parametrización de la conexión, 703
Instrucciones, 697
Modificar parámetros de configuración IP con IP_CONF, 7766
Modificar parámetros de configuración IP con T_CONFIG, 5336
Números de puerto, 716
Parametrización de la conexión, 697, 699
Parametrización de las conexiones en TCP e ISO on TCP, 7730
Parametrización del punto de acceso local en UDP, 7735
Parámetros de conexión, 700
Protocolos utilizados, 708
Recibir datos a través de UDP con TURCV, 5326, 7763
Recibir datos con TRCV, 5314, 5318, 7756
TCON, 7748
TCON_IP_RFC, 715
TCON_IP_v4, 714
TCON_Param, 711
TSAP, 718
Vía Industrial Ethernet, 615, 7725
Operación de Restore
Clave de licencia, 104
Operación en el visor de recetas
Transferir datos, 8976
Operación en runtime
Manejo simultáneo de varias teclas, 80
Operación lógica
AD, 4249, 6339
AW, 4243, 6333
OD, 4251, 6341
OW, 4245, 6335
XOD , 4253, 6344
XOW, 4247, 6337
Operación lógica con bits
O, 4075, 6171
Operaciones lógicas con bits
Insertar entrada, 3594, 5824
O, 3590, 3591, 5821, 5822, 6161, 6162, 6167, 6170
O-exclusiva, 3593, 5824, 6164, 6165, 6172, 6173
Y, 3588, 3589, 5819, 5821, 6158, 6159, 6168, 6169
Operaciones lógicas con palabras, 2563
Operando, 2280, 2281, 2286, 2288, 2289, 2290, 2292, 2293, 2294, 2295, 2297, 2298, 2301, 2302
Insertar, 2701, 2745, 2772
Operandos autorizados para la tabla de observación, 3123
Operandos permitidos para la tabla de forzado permanente, 3150
OPN, 4227, 5729, 5952, 6319
OPNDI, 4228, 6320
OPNI, 5730, 5953
Optimizar la configuración
Allen-Bradley DF1, 9013, 9035, 9053, 9064, 9081, 9094, 9111
OR, 2545, 2793, 3540, 3591, 3854, 5741, 5822, 5964
Orden
Procesamiento de peticiones, 7439, 7442
Orden de control
Transferir datos, 8879, 9129
OS, 2545, 2551, 2552, 2553, 2559
OSPF
Area Range, 1366
Areas, 1365
Áreas, 1231
Configuration, 1363
Estado del router, 1231
Interfaces, 1367
Link State Advertisment, 1231
OSPFv2 Interfaces, 1257
OSPFv2 LSDB (información), 1261
OSPFv2 Virtual Neighbors, 1260
Router, 1231
Vecinos OSPFv2, 1258
Vecinos virtuales OSPFv2, 1260
Virtual Links, 1370
Otros controladores
Direccionamiento, 8998
Particularidades, 8998
Tipos de datos, 8998
Otros dispositivos de campo (PROFIBUS y PROFINET), 1433, 1486, 1516
OUT_RANGE, 3384, 3696
OutputCPU314C, 7376
Descripción, 7376
Llamada, 7376
Parámetro, 7376

OutputET200S2AO, 7375
Descripción, 7375
Llamada, 7375
Parámetro, 7375
OutputIM174, 7379
Descripción, 7379
Llamada, 7380
Parametrización del IM 174 (configuración hardware), 7379
Parámetro, 7380
OutputIM178, 7373
Descripción, 7373
Llamada, 7373
Parámetro, 7374
Salidas digitales, 7374
OutputMM4_DP, 7378
Descripción, 7378
Llamada, 7379
Parámetro, 7379
OutputSINAMICS, 7381
Descripción, 7381
Llamada, 7384
Parametrización del SINAMICS en HWCN, 7381
Parámetro, 7384
OutputSM332, 7376
Descripción, 7376
Llamada, 7377
Parámetro, 7377
OutputSM432, 7374
Descripción, 7374
Llamada, 7374
Parámetro, 7375
OutputUniversal, 7377
Descripción, 7377
Funcionamiento, 7377
Llamada, 7378
Parámetro, 7378
OV, 2545, 2551, 2552, 2553, 2555, 2559, 2563
OVERRIDE
Instrucción, 7281
Parámetros de entrada, 7283
Parámetros de salida, 7283
OW, 4245, 6335

P

P, 3295, 3298, 3604, 3607, 5612, 5832
P_PRINT, 1070
P_PRINT341, 1070
P_RCV, 1070
P_RCV_RK, 1070, 7984
P_RESET, 1070
P_SEND, 1070
P_SND_RK, 1070, 7984
P_TRIG, 3300, 3610, 5614, 5835
P3964_Config, 1014, 1047
P3964_Config (configuración del protocolo), 5408, 7798
PageDown, 8689
PageUp, 8689
Página web definida por el usuario, 971
Páginas web
Intervalo de actualización, 2021
Páginas web definidas por el usuario, 946, 949, 958, 960, 961, 983, 986, 995, 996, 998, 999
Páginas web en el navegador, 967, 1005
Palabra de control del sensor, 11209
Palabra de datos (DBW), 1631
Palabra de datos locales (LW), 1631
Palabra de entrada (IW), 1631
Palabra de estado, 2545, 2551, 2552, 2553, 2555, 2556, 2559, 2560, 2563
Palabra de marca (MW), 1631
Palabra de salida (QW), 1631
Palabra doble, 2327
Palabra doble de datos (DBD), 1631
Palabra doble de datos locales (LD), 1631
Palabra doble de marca (MD), 1631
Palabra doble de salida (QD), 1631
Palabras clave, 2281
Panel de control de la CPU
Área de visualización, 2178, 2236
panel de operador
Estructura, 9250
Panel de operador, 102
Características, 9341, 9344
Cargar, 9268
Copia de seguridad, 9268, 9270, 9286
Crear una copia de seguridad de datos, 9271
Imagen, 9268
Límites del sistema, 9341, 9344
Modificar tipo de panel, 9277
Parámetros MPI, 8923
Parámetros PROFIBUS, 8811, 8861, 8914
Parámetros PROFINET, 8797, 8846, 8901
Poner nuevamente en marcha, 9280
Primera puesta en marcha, 9280
Restablecer la configuración de fábrica, 9274
Restaurar los datos, 9270, 9272, 9286
Software, 9268
Volver a cargar el sistema operativo, 9273
Panel de operador en un controlador
puesta en marcha, 9013, 9034, 9053, 9063, 9081, 9094, 9111

Panning, (Véase Desplazar la imagen)
PARA_CLT
Parámetros de entrada, 7284
Parámetros de salida, 7285
PARA_CTL
Instrucción, 7283
PARADA, 1712
Parada de runtime, 8723
Parametrización
Hardware, 509
Módulo tecnológico, 9359, 9436
Parametrización de la conexión
iniciar, 703
Parámetros de conexión, 700
Resumen, 699
Parametrización de la conexión de la instrucción PUT/GET, 727
Parametrizar
De un módulo con PARM_MOD, 6877
Parametrizar el módem remoto, 11381
Parametrizar la configuración de red AS-i, 557
Parametrizar la interfaz Ethernet, 11315
Agregar una dirección IP temporal, 11316
Borrar direcciones IP temporales, 11316
Conectar con subred, 11314
Parámetros modificables, 11313
Posibilidades de parametrización, 11313
Parametrizar la interfaz PG/PC, 11311
Detección automática de los parámetros del bus, 11318
Parametrizar la interfaz MPI, 11319
Parametrizar la interfaz PROFIBUS, 11321
Parámetro, 7661
Comunicación básica S7, 7661
Conexión, 9001, 9016, 9056, 9084, 9102
EncoderAbsSensorDP: , 7357
EncoderCPU314C, 7364
EncoderET200S1Count, 7360
EncoderET200S1SSI, 7355
EncoderFM350, 7362
EncoderFM450, 7353
EncoderIM174, 7367
EncoderSINAMICS, 7369
EncoderSM338, 7358
EncoderUniversal, 7372
ErrorID, 5130
MC_Control, 7341
MC_GearIn, 7347
MC_Home, 7337
MC_MoveAbsolute, 7327
MC_MoveJog, 7332
MC_MoveRelative, 7329
MC_StopMotion, 7339
Modificar ajustes internos, 2870
OutputCPU314C, 7376
OutputET200S2AO, 7375
OutputIM174, 7380
OutputIM178, 7374
OutputMM4_DP, 7379
OutputSINAMICS, 7384
OutputSM332, 7377
OutputSM432, 7375
OutputUniversal, 7378
para CPU, 1646
S7 200, 8955, 8960, 8965, 8966
Parámetro BUFFER, Send_P2P, 5412, 7802
Parámetro de entrada PID_ES, 7212
Parámetro LENGTH, Send_P2P, 5412, 7802
Parámetros
ABS_CTRL_451, 7425
ABS_DIAG_451, 7427
ABS_INIT, 7420
CAM_CTRL_452, 7457
CAM_DIAG_452, 7459
CAM_INIT, 7453
CAM_MSRM_452, 7460
CNT_CTL1, 7391
CNT_CTRL, 7388
CNT2_CTR, 7406
CNT2RDPN, 7405
CNT2WRPN , 7404
Conexión, 9043, 9072
DIAG_INF, 7390
DIAG_RD, 7407
Dinámicos, transferir con WR_PARM, 6882
MPI, 8921, 8923, 8924
Parámetros ocultos, 2703, 2747
Predefinidos, escribir con WR_DPARM, 4882, 6883
PROFIBUS, 8809, 8811, 8812, 8859, 8861, 8862, 8913, 8914, 8916
S7 1200, 8891
S7 1500, 8836
S7 200, 8950
SIMATIC LOGO!, 8988
Parámetros (CPU), 1931
parámetros básicos
High_Speed_Counter, 9411
Parámetros básicos
módulo tecnológico, 9429
Parámetros de arranque, 1626, 1737, 1905
Parámetros de bloque, 2268, 2270, 2272, 2274, 2276, 2277, 2278, 2279
Interfaz del bloque, 2642

Principios básicos, 1690
Parámetros de conexión
Allen-Bradley DF1, 9016
Allen-Bradley EtherNet/IP, 9001
Mitsubishi FX, 9056
Mitsubishi MC TCP/IP, 9043
Modicon Modbus RTU, 9084
Modicon Modbus TCP, 9072
Omron Host Link, 9102
Parámetros de conexión de la instrucción PUT/GET, 726
Parámetros de dirección
Broadcast, 672
Parámetros de dirección IP, 1493, 1501
Parámetros de entrada
High_Speed_Counter, 5127
LP_SCHED_S, 7323
Parámetros de entrada del bloque de función GRAPH, 2851
Parámetros de interfaz PROFINET
Restablecer, 2244
Parámetros de protocolo, 1119
Parámetros de salida
High_Speed_Counter, 5129
Parámetros de salida del bloque de función GRAPH, 2855
Parámetros del reloj, 1953, 1954, 2040
Parámetros estáticos del bloque de función GRAPH, 2857, 2859, 2860
Parámetros internos, 47
declarar remanentes, 2870
Declarar visibles y accesibles desde HMI, 2870
Parámetros IP, 2171
Parámetros MPI
S7 300/400, 8921, 8923, 8924
Parámetros ocultos, 2703, 2747
Parámetros PROFIBUS
S7 1200, 8859, 8861, 8862
S7 1500, 8809, 8811, 8812
S7 300/400, 8913, 8914, 8916
Parámetros PROFINET
Conexión HMI, 8796, 8845, 8900
Controlador, 8799, 8848, 8903
Panel de operador, 8797, 8846, 8901
S7 1200, 8845, 8848
S7 1500, 8796, 8797, 8799
S7 300/400, 8900, 8901, 8903
PararRuntime, 8702
Paréntesis, 4079, 6175
Parpadeo, 8395
Particularidades
Omron Host Link, 9113
Paso de trabajo
Para crear imágenes, 8313
Paso por cero, 9506, 9589
PE_CMD, 4826, 6826
PE_DS3_Write_ET200S, 4831, 6831
PE_End_RSP, 4863, 6861
PE_Error_RSP, 4860, 6858
PE_Get_Mode_RSP, 4865, 6864
PE_I_DEV, 4856, 6855
PE_List_Modes_RSP, 4864, 6862
PE_Measurement_List_RSP, 4871, 6869
PE_Measurement_Value_RSP, 4872, 6870
PE_Mode_ID, 8117, 8142
PE_PEM_Status_RSP, 4867, 6866
PE_Start_End, 4822, 6822
PE_Start_RSP, 4861, 6860
PE_WOL, 50, 4833, 6832
PEEK, 4384
Leer dirección de memoria, 4384
PEEK_BOOL, 4386
Leer bit de memoria, 4386
Pegar
Adaptar el tamaño de la imagen, 9239
Color, 9239
Fuente, 9239
Objeto no válido, 9238
principio, 9237
Tecla de función, 9239
Pérdida
Sincronización, 11025
Pérdida de la sincronización, 9762, 9840
Perfect Forward Secrecy, 818
Perfil, (Véase perfil de bus)
Perfil de instrucción
Abrir, 2634
Activar y desactivar, 2636
Borrar, 2636
Crear, 2633
Editar, 2634
Principios básicos, 2632
Perfil de velocidad, 10679
Perfiles, 1681
Perfiles (S7-1500), 1970
Perfiles PROFIBUS, 547, 8864, 8917
Diferentes perfiles en la misma subred, 8864, 8917
Distintos perfiles en la misma subred, 547
Repercusiones en la velocidad de transferencia, 548
Repercusiones sobre la velocidad de transferencia, 8864, 8917
Significado de los perfiles, 548, 8864, 8918

Periferia, 40, 52, 53, 2294
acceso directo, 1636, 1916
acceso directo a la ~, 1810
Periferia descentralizada, 1387, 1463, 2112
Configurar, 1387
Números de slots, 1388
Período
Resolución, 9507, 9649, 9709
Permiso, 9313
Permitir acceso vía comunicación PUT/GET del interlocutor remoto, 1674, 1959
Permitir sobrescribir el nombre de dispositivo, 1513
personalizada, 31
Petición
Ajustar valor real, 10926, 11021
Ajustar valor real al vuelo, 11021
Cancelar preselección del valor real, 10927, 11022
Decalaje de señal cero, 11023
Definir punto de referencia, 10928, 11025
Petición de control, 8741
Petición de escritura
CNT2WRPN, 7415
Petición de forzado
en la tarjeta SD, 33
Petición de forzado permanente, 1907
Petición de lectura
CNT2RDPN, 7418
Petición del controlador
Transferir datos, 8826, 8937, 8977
Peticiones
ABS_CTRL, 7438
ABS_CTRL_451, 7441
ABS_DIAG, 7423
ABS_DIAG_451, 7427
CAM_CTRL, 7473
CAM_CTRL_452, 7474
CAM_DIAG, 7456
CAM_DIAG_452, 7459
CNT_CTL1, 7392
CNT_CTRL, 7389
Peticiones de escritura, 7439, 7442
Peticiones de lectura, 7439, 7442
Peticiones de Motion Control
Finalización de peticiones de Motion Control, 10767
Inicio de peticiones de Motion Control, 10756
Seguimiento de peticiones en curso, 10758
PID
Instrucción, 7285
Modo de funcionamiento, 7286
Parámetros de entrada, 7288
Parámetros de salida, 7289
PID_3Step
Instrucción, 5180, 5212
Parámetros de entrada, 5187, 5218
Parámetros de entrada/salida, 5192
Parámetros de salida, 5190, 5220
Variables estáticas, 5221
PID_Compact
Parámetros de entrada, 5140, 5164
Instrucción, 5160
Parámetros de entrada/salida, 5142
Parámetros de salida, 5141, 5164
Variables estáticas, 5143, 5165
PID_CP
Parámetros de salida, 7185
Descripción, 7168
Parámetros de entrada, 7183
Parámetros de entrada/salida, 7186
PID_ES
Parámetros de salida, 7214
PID_FM
Diagramas de bloques, 7495
Funcionamiento, 7485
Instrucción, 7484
Parámetros de entrada, 7488
Parámetros de entrada/salida, 7491
Parámetros de salida, 7488
PID_FM_455
Diagramas de bloques, 7581
Funcionamiento, 7571
Instrucción, 7570
Parámetros de entrada, 7573
Parámetros de entrada/salida, 7577
PID_PAR
Instrucción, 7512
Parámetros de entrada, 7514
Parámetros modificables, 7515
PID_PAR_455
Instrucción, 7597
Parámetros de entrada, 7599
Parámetros de salida, 7599
Parámetros modificables, 7600
Pila de bloques, 1721, 1729
Pila de datos locales, 1726
Pila de interrupciones, 1728
Ping, 1301
PIP (tipo de datos), 927
Pista de leva de contaje, 9739
Pista de leva de freno, 9740
Pistas
Activar señal de pista, 9738
Control de las salidas de pista, 9802, 9880

Desbloquear, 9738
Habilitación externa, 9738
Pista de leva de contaje, 9803, 9881
Pista de leva de freno, 9803, 9881
Pistas especiales, 9803, 9881
Resultado de pista, 9737
Valor de contaje superior pista de leva de contaje, 9803, 9881
Pistas de levas, 9737
Pistas especiales, 9739
Planificador de tareas, 8726
Área de trabajo, 8728
Disparador, 8730
Disparo por eventos, 8730, 8733
Lista de funciones, 8729
Tarea desactivada, 8730
Plano de ocupación, 1910, 1911
Plano de ocupación
Significado de los símbolos, 3056
Activar la indicación de las marcas remanentes, 3062
Borrar filtros, 3060
Configurar las opciones de visualización, 3058
Definir filtros, 3060
Ejemplo de la vista de entradas y salidas, 3055
Ejemplo de la vista de marcas, 3055
Estructura, 3055
Filtrar, 3061
Introducción, 3054
Mostrar, 3057
Posibilidades de filtrado, 3059
Plantilla, 3174, 8406
Borrar, 8320
Cambiar nombre, 8320
Copiar, 8320
Crear, 8319
Desplazar, 8320
Edición, 3180, 3188
Imagen general, 8319
Insertar, 8320
utilizar en la imagen, 8322
Plantilla maestra
Librería, 9137
PLC5, 9029
PLCSIM, 38, 47
PLUG, 1299
C-PLUG, (C-PLUG)
KEY-PLUG, (KEY-PLUG)
PNIO_RECV, 8099
PNIO_SEND, 8092
Poca memoria disponible, 8725
Poca memoria disponible, crítico, 8725
POINTER, 2359
POKE, 4388
Escribir dirección de memoria, 4388
POKE_BLK, 4392
Escribir área de memoria, 4392
POKE_BOOL, 4390
Escribir bit de memoria, 4390
Poner
Valor inicial del contador, 5660
Poner a 0
Contador, 6210
Poner a cero
Contador, 4116
Poner al valor inicial
Valor inicial del contador, 3367
Poner nuevamente en marcha
Panel de operador, 9280
POP, 4276, 6366
PORT_CFG, 5499
Port_Config, 1014, 1047
Port_Config (configuración del puerto), 5399, 7789
Portada
Comodines para la información del documento, 404
Posibilidades de rellamada en TeleService, 11385
Posibilidades de test en la Tabla de forzado permanente, 3143
Posibles aplicaciones, 1119
Posición
De objetos, 8332
Modificar varios objetos, 8355
Posición de la marca de referencia, 10655
Posicionamiento (CPU 31xC), 1774
Posicionamiento absoluto
Easy Motion Control, 11221
Posicionamiento controlado
analógicamente con ANALOG, 7629
digital con DIGITAL, 7640
Posicionamiento relativo
Easy Motion Control, 11221
PPI, 8760
Arquitectura de red, 8764
Red, 8763
S7 200, 8966
Preajustes de dinámica, 10718, 10735
Preguntas frecuentes, 8759
Premium, 9079
Preparar la migración, 131
Preshared Keys, 816
Primera consulta, 2545
Primera puesta en marcha
Panel de operador, 9280

Principio de línea
Línea, 8438
Principios básicos
Migración, 147
Prioridad de OB, 1925
Prioridades
Tratamiento de errores local, 2935
Procedimiento 3964(R), 1033, 1094
Carácter de comprobación de bloque, 1095
Caracteres de control, 1034, 1094
Enviar datos, 1096
Prioridad, 1034, 1095
Recibir datos, 1099
Procedimiento 3964R
Carácter de control de bloque, 1034
Procedimiento de aviso analógico, 8503
Procedimiento de aviso de bit, 8503
Procedimiento handshake, 1021
Procedimientos de aviso, 8503
Procesador de comunicaciones
Software SIMATIC NET PC, 2042
STEP 7 V5 + STEP 7 Professional, 2066
Procesamiento de peticiones
Orden, 7439, 7442
Procesamiento de valores analógicos, 3500, 3503, 3813, 3816, 3987, 3989, 4422, 4424, 4599, 4601
Proceso CiR
Controlar, 5960, 6091, 6476
Proceso de posicionamiento
Fin, 10904, 10999
PROFIBUS, 40, 547, 1446, 8760, 8864, 8917
Acoplamiento con PROFINET, 1488
Cambiar interfaz DP, 1398
Comunicación directa, 1424
Conexión HMI, 8806, 8858, 8911
Crear un sistema maestro DP, 1427
DPV0/DPV1, 1433
Estándar, 8864, 8917
ET 200S en modo DPV1, 1413, 1483
Funcionalidad distinta, 1435
OP 73, 8839, 8894, 8953, 8955, 8963, 8965, 8966
OP 77A, 8839, 8894, 8953, 8955, 8963, 8965, 8966
Parámetros, 8809, 8811, 8812, 8839, 8859, 8861, 8862, 8894, 8913, 8914, 8916, 8953, 8963
Red, 8762
S7 1200, 8857, 8858
S7 1500, 8808
S7 300/400, 8806, 8910, 8911
Universal, 8864, 8917
PROFIBUS DP, 1401, 1466, 8762
Conexión, 8946
PROFIBUS DPV1, 1748
PROFIBUS isócrono, (Véase modo isócrono)
PROFIdrive, 10646
PROFIenergy, 1576
Ajustar el modo de activación de los módulos de potencia, 4831, 6831
Descripción, 4820, 6820
Forzar comandos PROFIenergy en el iDevice, 4856, 6855
Generación de respuesta negativa a comando, 4860, 6858
Generar respuesta a comando para final de pausa, 4863, 6861
Generar respuesta a comando para iniciar pausa, 4861, 6860
Generar respuesta con comandos PROFIenergy soportados, 4869, 6867
Generar respuesta con datos de energía consultados, 4865, 6864
Generar respuesta con estado PEM, 4867, 6866
Generar respuesta con lista de valores medidos soportados, 4871, 6869
Generar respuesta con modos de ahorro de energía consultados, 4864, 6862
Generar respuesta con valores medidos solicitados, 4872, 6870
Iniciar y finalizar modo de ahorro de energía, 4822, 4826, 6822, 6826
Leer información de estado, 4826, 6826
PE_CMD, 4826, 6826
PE_DS3_Write_ET200S, 4831, 6831
PE_End_RSP, 4863, 6861
PE_Error_RSP, 4860, 6858
PE_Get_Mode_RSP, 4865, 6864
PE_I_DEV, 4856, 6855
PE_Identify_RSP, 4869, 6867
PE_List_Modes_RSP, 4864, 6862
PE_Measurement_List_RSP, 4871, 6869
PE_Measurement_Value_RSP, 4872, 6870
PE_PEM_Status_RSP, 4867, 6866
PE_Start_End, 4822, 6822
PE_Start_RSP, 4861, 6860
Wake On LAN, 4833, 6832
PROFINET, 553, 867, 1521, 1526, 1530, 1554, 1571, 1573, 1944, 8760
Acoplamiento con PROFIBUS, 1488
Conexión, 8944
Conexión HMI, 8793, 8794, 8842, 8843, 8897, 8898
Diagnóstico, 1525
Direcciones IP, 1517

Nombre del dispositivo, 1518
Número de dispositivo, 1518
Permitir sustitución de dispositivo sin medio de almacenamiento extraíble, 1513
S7 1200, 8842
S7 1500, 8793
S7 300/400, 8897
PROFINET CBA
Actualizar entradas de la interfaz del programa con PN_IN, 8265
Actualizar interconexiones con PN_DP, 8267
Actualizar salidas de la interfaz PROFINET con PN_OUT, 8266
PROFINET IO, 1486, 1491, 1493, 1503, 1505, 1546
Opciones de puerto, 1510, 8803, 8854, 8907
PROFINET IRT, 1533
Profundidad de anidamiento, 2264
Program_Alarm, 4916
Programa
Cíclico (S7-300, S7-400), 1822
Determinar tiempo de ejecución con OB_RT, 7023
Programa antivirus, 85, 95, 99
Programa AWL
Comentario de línea, 2781
Insertar instrucción, 2769
Insertar llamada de bloque, 2776, 2778
Introducir instrucción, 2768
Reglas, 2767
Sección de comentario, 2781
Seleccionar instrucción, 2782
Visualización del estado del programa, 3110
Programa cíclico (OB 1) (S7-300, S7-400), 1822
Programa cíclico (S7-1500), 1963
Programa de arranque, 1675
Programa de arranque (S7-1500), 1962
Programa de ejemplo
OB 122, 1884
OB 81 (S7-300, S7-400), 1851
Programa de seguridad, 95, 99
Programa de usuario
Buscar errores, 2930
Evaluación de fallos, 9820, 9900, 10975, 11077
Función, 2249
Probar, 3085
Programa SCL
Insertar comentarios, 2822
Insertar instrucción, 2805, 2806
Insertar llamada de bloque, 2813, 2816
Insertar una llamada de bloque, 2814, 2817, 2818, 2819
Mostrar y ocultar una lista de parámetros, 2821
Selección de una instrucción, 2823
Visualización del estado del programa, 3111
Programación
Estructurada, 2250
Instrucciones PtP, 5394, 7784
Lineal, 2250
Modbus, 1051
PtP, 1044
USS, 1049
Programación estructurada, 2250
Programación lineal, 2250
Programación punto a punto, 5394, 7784
Programación simbólica
Mostrar direcciones absolutas, 2630
Programar, 10747
Instrucciones de Motion Control, 10751
Programming error (S7-1500), 1983
Prolongación del impulso, 2124, 2143
Propiedad de bloque
Editar, 2613
Función, 2607
Mostrar, 2613
Resumen, 2608
Propiedades (CPU, 1646, 1931
Propiedades de comunicación de un módulo, 2232
Propiedades de grupo, 816
Propiedades de interfaces, 2171, 2233
Propiedades de la CPU, 1646, 1931
Propiedades del dominio MRP, 2174
Propiedades del dominio Sync, 2173
Propiedades del grupo VPN, 816
Propiedades OPC", 653, 660, 666
ProSave, 9269
Instalación, 75
Protección contra copia, 3048
Protección contra escritura, 1670
Protección de acceso, 1960, 8625
Administración de usuarios, 8624
configurar, 8629
de fábrica, 11382
nivel de protección 1, 11383
nivel de protección 2, 11383
validez, 11382
ventajas, 11383
Protección de acceso IP, 614, 2005, 2008
Lista IP Access Control, 2007
Protección de acceso local, 1960
Protección de know-how
Abrir bloques, 3050
Cambiar la contraseña, 3052
Configurar, 3049
Imprimir bloque, 3051

Introducción, 3045
Quitar, 3052
Protección por contraseña, 979, 1646, 1670, 1671, 1756, 1931, 1955
Borrar derechos de acceso, 389
Concepto de protección, 389
PROTECT, 5735, 5957, 6087, 6474, 6604
Proteger
tecla de función con contraseña, 8411
Protocolo, 778
Protocolo 3964(R) (CPU 31xC), 1798
Protocolo ASCII, 1027
Protocolo ASCII (CPU 31xC), 1791
Protocolo de comunicación
definir, 1663
Resumen, 1662
Protocolo de interfaz serie universal, (Véase protocolo USS)
Protocolo ESP, 818
Protocolo Freeport, 1027
Protocolo IP, 8848
Protocolo ISO, 867
Protocolo para mainframe RK512, 1103
Enviar datos, 1106
Recoger datos, 1109
Telegrama de orden, 1103, 1104
telegrama de reacción, 1103
Protocolo RK 512 (CPU 31xC), 1803
Protocolo USS
Estructura general del bloque de datos de red:Área de datos de proceso (PZD), 1193
Estructura general del bloque de datos de red:Área de parámetros (PKW), 1193
Estructura general del bloque de datos:Área de datos de proceso (PZD), 1039
Estructura general del bloque de datos:Área de parámetros (PKW), 1039
Instrucciones, 5518
Protocolos de los módulos de comunicaciones, 1012
Proxy ARP, 874
Proxy de dispositivos
Actualizar, 9161, 9166
Proxy de PROFINET IO, 2078
Proxy del dispositivo
Inicializar, 9163
Proyecto
Actualizar, 9248
Armonizar, 484
Borrar, 348
Carga, 9260
Cargar, 9280
Cargar (resumen), 9254
Cerrar, 347
Comparación detallada, 384
Comparar datos, 39, 373
Compilar , 9256
Compilar (resumen), 9254
Compilar datos de proyecto, 364
Crear, 340
Guardar, 347
migrar, 167
Migrar, 149
Mostrar las propiedades, 346
multilingüe, 9205
Principios básicos, 339
Realización de una comparación offline/offline, 375
Realización de una comparación online/offline, 374
Simular a través del simulador de variables, 9264
Varias conexiones, 8821, 8873, 8931, 8972
Proyecto de maquinaria de serie, 1597, 1600, 1603, 1606
Proyecto de referencia
Abrir, 361
Cerrar, 361
Comparar, 362
Principios básicos, 360
Proyecto de WinCC flexible
Migrar, 149, 167
Proyecto integrado
Borrar una conexión no especificada, 184
Convertir una CPU no especificada, 180
Crear una conexión HMI integrada, 181
Enlazar variables HMI, 183
Migrar, 165, 177
Reajustar, 179
Proyecto migrado
Compilar, 150
Prueba de coherencia ("Check Consistency"), 865
a nivel de proyecto, 759
local, 759
PRVREC, 4797
PT, 3635
PTO, 49
Puerta
hardware, 9486
interna, 9487
software, 9486
Puerta hardware, 9361, 9365, 9417, 9441, 9486
Puerta HW
Abrir y cerrar de forma controlada por flanco, 9489
Abrir y cerrar de forma controlada por nivel, 9488

Puerta interna, 9487
Puerta software, 9365, 9486
Puerta SW, 9361
Abrir y cerrar, 9488
Puerto, 553
102 (protocolo S7 - TCP), 778
123 (NTP), 800, 807
161 (SNMP), 800
20/21 (FTP), 778
443 (HTTPS), 800
4500 (IPsec), 800
500 (IPsec), 800
500 (ISAKMP - UDP), 826
514 (Syslog), 800
80 (HTTP), 778
8448 (diagnóstico de seguridad), 890
Basic Panel, 8790
cambiar nombre, 746
Configuración, 1291
Configuración de puerto, 1290, 1293
Interconexión, 1507
Puerto de comunicación
configurar, 1660
Puerto de redundancia, 2060
Puerto en anillo, 1571, 1573
Puerto interlocutor cambiante, 1560
Puerto TCP, 2032
Puertos interlocutores que cambian durante el funcionamiento, 1560
Puesta en servicio
High_Speed_Counter, 9425
módulo tecnológico, 9455
Puesta en servicio automática, 1513
Pull or plug of modules (S7-1500), 1981
Pulsar, 8719
Pulsar ESC dos veces, 8720
Pulsar tecla, 8724
PULSE, 7658
PULSE REFEED, 11201
PULSEGEN
Funcionamiento, 7089
Instrucción, 7088
PULSEGEN_M
Instrucción, 7289
Modo de funcionamiento, 7290
Parámetros de entrada, 7296
Parámetros de salida, 7297
Puntero
ANY, 2362
POINTER, 2359
VARIANT, 2364
Puntero de área, 8815, 8866, 8926, 8967
Activo, 8782
Allen-Bradley, 9036
Conexión, 8785
Configurar, 8783
coordinación, 8870, 8971
Coordinación, 8819, 8929, 9121
Curvas, 8831, 8992
Fecha/hora, 8816, 8867, 8868, 8927, 8968, 9119
Fecha/hora del controlador, 8818, 8928, 8969, 9120
Identificador del proyecto, 8820, 8873, 8930, 8972, 9122
Longitud, 8782, 8784
Mitsubishi, 9065
Modicon Modbus, 9095
Número de imagen, 8816, 8872, 8926, 8968, 9118
Omron, 9112
Petición de control, 8874, 8973, 9123
Petición del controlador, 8821, 8931
registro, 8573
Registro, 8824, 8876, 8934, 8975, 9126
Punteros de área, 8779, 9116
Basic Panel, 8791
Ciclo de adquisición, 8782
Comentario, 8782
Conexión, 8780
Crear, 8783
Dirección, 8782
Editor Conexiones, 8781
Ficha, 8782
Nombre de visualización, 8782
Variable de controlador, 8782
Punto a punto, 1125, 1229
Punto de cambio de velocidad, 10889
Punto de desconexión, 10889
Punto de parada
Activar, 3092
Ajustar el entorno de llamada, 3091
Borrar, 3096
Continuar la ejecución del programa, 3094
Continuar la ejecución del programa hasta la marca, 3094
Desactivar, 3092
Ejecutar llamada, 3094
Ejecutar siguiente instrucción, 3094
Navegar hasta los puntos de parada, 3091
Posicionar, 3090
Test en modo de etapa individual, 3089
Punto de referencia, 10655
Puntos de conmutación, 10889
PUSH, 4275, 6364

PUT, 5252
PUT/GET, 1674, 1959
PWM, 49
PZF (error de acceso a periferia), 1726

Q

Q
  Controlador, 9048
Q0xUDEH CPU
  Controlador, 9050
QB, 2571
QD, 2571
QI (Quality Information), (Véase Información de calidad)
QRY_CINT, 4888
QRY_DINT, 4901
QRY_TINT, 4896
Quantum, 9079
QW, 2571
QX, 2571

R

R, 547, 3288, 3597, 4082, 4097, 4116, 5606, 5827, 6177, 6193, 6210
R_TRIG, 3303, 3612, 3904, 4283
Rack, 514, 910
  Insertar módulo, 523
  Número máximo, 899, 900
  Reglas de equipamiento del S7-300, 899
  Reglas de equipamiento S7-400, 900
Rack ampliable, 911
Rack de ampliación
  ER1 (S7-400), 912
  IM 46x, 912
  Reglas, 913
Rack or station failure, 1689
Rack or station failure (S7-1500), 1982
Radio, 8435
Radio horizontal, 8430
Radio vertical, 8430
RADIUS, 1375
Raíz cuadrada, 3421, 3733, 4147, 4343, 5689, 5912, 6241, 6426, 6565
RALRM, 4753
Rama
  Borrar, 2707, 2750, 2899
  Cerrar, 2706, 2898
  Definición, 2705, 2749
  Insertar, 2706, 2750
  Reglas, 2705
  Reglas , 2749
Rama alternativa
  Crear, 2896
  Introducción, 2829
Rama simultánea
  Crear, 2897
  Introducción, 2828
Ramificación, 4436, 4438, 6456, 6458
Rampa de parada de emergencia, 10681
Rango
  Valor dentro del rango, 3383, 3695
  Valor fuera del rango, 3384, 3696
Rango (... - ...)
  Lista de gráficos, 8386
Rango de contaje, 9505, 9588
Rango de direcciones, 790
Rango de histéresis, 9740
Rango de medición, 9390, 9393
Rango de valores de la dirección IP, máscara de subred y dirección de la transición de red, 2001
Rangos de datos
  Escribir, 8783, 9118
  Leer, 8783, 9118
  Punteros de área, 8783, 9118
RCV_CFG, 5504
RCV_PTP, 1071, 5513, 8239
RCV_RST, 5514
RCVREC, 4795
RD_ADDR , 5009
RD_DPAR, 4875
RD_DPARA, 4878
RD_DPARM, 4880
RD_LGADR, 5014
RD_LOC_T, 4697
RD_REC, 4785
RD_SINFO, 4923
RD_SYS_T, 4696
RDREC, 919, 4746
RE_TRIGR, 3526, 3839, 4007, 4455, 4615, 5733, 5955, 6085, 6471, 6602
Reacción a STOP de la CPU, 9429
Reacción a STOP de la CPU/del maestro
  1STEP 5V, 11182
Reacciones a errores
  Easy Motion Control, 11222
READ_355
  Instrucción, 7506
  Parámetros de entrada, 7507
  Parámetros de salida, 7508
READ_455
  Instrucción, 7592

Parámetros de entrada, 7593
READ_BIG, 4398
READ_DBL, 51, 4997
READ_ERR, 4910
READ_LITTLE, 4394
ReadFromArrayDB, 3468, 3781, 3964, 4374, 4572
ReadFromArrayDBL, 3473, 3786, 3969, 4378, 4576
Reajuste del encóder absoluto, 9787, 9865, 10952, 11053
Calcular, 9788, 9866, 10953, 11054
Ejemplo, 9789, 9867, 10954, 11055
Reajuste mecánico, 9788, 9866, 10954, 11055
Reajuste del valor absoluto, 10668
REAL, 2338, 2392, 2440, 2474, 2498, 2516, 2531
REAL_TO_, 2440, 2498, 2531
Rearranque, 1739
Ejecutar con RESUME, 7718
Rearranque (en caliente), 1624, 1903
Rearranque (S7-300, S7-400), 1875
Rearranque completo, 1269
Rearranque completo (en caliente) (S7-300, S7-400), 1875
Rearranque completo de la CPU, 1196
Rearranque de la CPU, 1196
Rearranque normal, 1705, 1739
Rearranque normal manual, 1739
Reasignar, 9196
Rebase de valor límite
Aviso analógico, 8476
Rebase del margen de medición, 10283, 10335, 10370
Rebase del tiempo de respuesta, 1029
Rebase por defecto, 2116, 2125
Rebase por defecto del rango de medición, 10283, 10335, 10370
Rebase por exceso, 2116, 2125
Receive_Config, 1014, 1047
Receive_Config (configuración de la recepción), 5403, 7793
Receive_P2P, 1014, 1047
Receive_P2P (recibir datos punto a punto) , 5413, 7803
Receive_Reset, 1014, 1047
Receive_Reset (resetear receptor), 5415, 7805
Receta, 8567, 8569, 8570, 9172, 9328
crear, 8581
Ejemplo de aplicación: Parametrización de máquinas, 8569
ejemplo de aplicación: producción por lotes, 8569
Elemento, 8570, 9329
exportar, 9169
Exportar a un archivo CSV, 4963
Flujo de datos, 9331
importar, 9171
Importar a bloque de datos, 4965
Memoria necesaria, 9349
Principios básicos, 8567, 8569
Registro, 8570, 9328
uso de listas de textos, 8576
utilización, 8569
Recibir datos
De fuera de la estación S7 propia con X_RCV, 8269
Driver ASCII, 1083, 1118
En I-Device con RCVREC, 4795, 6781
Procedimiento 3964(R), 1099
y guardar con RCV_PTP, 8239
y poner a disposición con SERVE_RK, 8253
Recibir SMS, 5376
Recipe, 982
RecipeExport, 4963
RecipeImport, 4965
Recoger datos
RK512, 1109
Recomendaciones de programación, 252
Reconectar
Variable, 8474
Reconectar una variable PLC, 8474
Reconfiguración mediante programa de usuario, 930
Recorrido de anticipación, 9742
Rectángulo
Configurar, 8372
Insertar, 8372
Insertar y configurar, 8371
Radio de las esquinas X, 8445
Radio de las esquinas Y, 8445
Recurso de conexión, 558, 570
Recursos de conexión, 595, 2237
online, 2215, 2216
Recursos de programa, 49, 2595
Recursos del sistema asignados dinámicamente
Liberar con DEL_SI, 6959
RECV_440, 1071
Red, 46, 8734
Ethernet, 8761
MPI, 8762
PPI, 8763
PROFIBUS, 8762
PROFINET, 8761
Red de comunicación, 8760
PROFINET, 8761
Red GSM, 1208
Redisparar el punto de referencia, 9786, 9864

Redondear, 3494, 3496, 3497, 3499, 3807, 3809, 3810, 3812, 4192, 4193, 4194, 4195, 4417, 4418, 4419, 4594, 4595, 4596, 4598, 5715, 5716, 5718, 5719, 5938, 5939, 5941, 5942, 6285, 6286, 6287, 6288, 6447, 6448, 6449, 6589, 6591, 6592, 6593
Reducir el ciclo del sistema, 1459
Reducir el tiempo de reacción del proceso, 1459
ReducirValorEnfocado, 8704
ReducirVariable, 8704
Reductor
    Easy Motion Control, 11221
Reductor de carga, 10652
Redundancia de anillo, 1323
Redundancia de medios (MRP), 1567, 1569, 1571, 1573
Redundancia OPC UA, 2059
Reemplazar
    Referencias cruzadas, 9196
Referencia, 504, 1240, (véase Referencia)
Referencia cruzada
    Ventana de inspección, 64
Referenciación
    Modos de referenciado, 10440
Referenciado, 10653
    Estado de referenciado, 10653
    Modo de referenciado, 10654
    Reajuste del valor absoluto, 10654
    Referenciado activo, 10654
    Referenciado directo, 10654
    Referenciado pasivo, 10654
    Referenciado volante, 10654
Referenciado activo
    Con entrada digital, 10661
    Con marca cero, 10660
    Con marca cero y detector de proximidad, 10656
Referenciado directo, 10667
Referenciado pasivo
    Con entrada digital, 10666
    Con marca cero, 10665
    Con marca cero y detector de proximidad, 10664
Referencias, 252, 1010
Referencias cruzadas
    Introducción, 3080, 9193
    mostrar, 3083
    Mostrar, 9194, 9196
    Reasignar, 9196
    Reemplazar, 9196
    Uso, 9193
    Utilidad, 3080
Refresco de imagen, 4280, 6380
Registro, 828, 8741
    CP x43-1 Adv., 878
    De un módulo, leer con RD_DPAR, 4875, 6874
    De un módulo, leer de forma asíncrona con RD_DPARA, 4878, 6876
    Escribir con WR_REC, 4790, 6774
    Escribir parámetros predefinidos con WR_DPARM, 4882, 6883
    Escribir y leer registros, 4874, 6872
    lectura, 8595
    Leer con RD_REC, 4785, 6769
    Leer con RDREC, 4746, 6735
    Leer de datos del sistema configurados, con RD_DPARM, 4880, 6881
    Parámetros dinámicos, transferir con WR_PARM, 6882
    Proveer en I-Device con PRVREC, 4797, 6784
    Recibir en I-Device con RCVREC, 4795, 6781
    SCALANCE S, 854, 855
    transferir, 8596
    Transferir con WRREC, 4748, 6738
Registro de control, 930
Registro de desplazamiento, 5809, 5813, 6030, 6036, 6147, 6151, 6526, 6529, 6662, 6665
Registro de diagnóstico, 10280, 10281, 10332, 10366
Registro de diagnóstico DS0
    Asignación, 10281, 10333, 10366
Registro de diagnóstico DS1
    Asignación, 10282, 10334, 10368
Registro de diagnóstico DS1 del FM 355-2
    Asignación, 10334
Registro de direcciones, 2311, 2312, 2318, 2319, 4277, 4279, 6370, 6371
Registro de receta
    Borrar, 9336
    crear, 8581
    Crear, 9335
    Crear en el panel de operador, 8594, 8595
    Editar, 9335
    Leer del controlador, 9337
    Sincronizar con el controlador, 9335
    Transferir al controlador, 9338
    Transferir el proyecto, 81
    uso de listas de textos, 8576
Registro del bloque de datos, 4229, 6321
Registro del sistema en red, 828
Registro local, 828
Registros de auditoría, 836, 838
Registros de filtrado de paquetes, 837, 839
Registros de la CPU, 1808
Registros del sistema, 835, 838
Reglas
    Segmento AWL, 2767

Reglas de cableado (PROFINET), 1512, 8804, 8855, 8908
Reglas de conexión, 885, 894
Reglas de cortafuegos globales, 774
    Asignar, 776
Reglas de cortafuegos predefinidas
    CP 1543-1, 890
    CP 1628, 881
    CP x43-1 Adv., 878
    SCALANCE S, 853, 855
Reglas de filtrado de paquetes IP, 785
Reglas de marcación en listines telefónicos, 11376
Reglas de slots
    DPV1, 1440
    ET 200S, 1404, 1475
Reglas generales, 2269
Reglas globales de filtrado de paquetes, 776
Reglas para filtrado de paquetes MAC, 788
Reglas para la asignación de direcciones, 915
Reglas para la configuración de redes MPI
    Reglas para la asignación de direcciones MPI, 543
Reglas para la configuración de redes PROFIBUS
    Asignar direcciones de dispositivos, 544
    Asignar direcciones de nodos, 543
Reglas para slots, 511
    Esclavo DP, 1388, 1399
    Estación PC, 1200
    ET 200M, 1388
    S7-1500, 902
    S7-300, 899
    S7-400, 900
    Servidor OPC, 1200
Reguladores de software
    Configuración, 9933
Rehacer acciones
    Principios básicos para rehacer acciones, 413
    Rehacer acciones, 417
Rehacer diseño de ventana, 312
Reinicialización de objetos tecnológicos, 10768
Reinicializar objetos tecnológicos, 10768
Reiniciar conexión, 5329
Rellamada a un número indicado al establecer la conexión, 11385
Rellenar
    Área, 3461, 3490, 3775, 3803, 3960, 3983, 4368, 4411, 4566, 4589, 5710, 5933, 6073, 6442, 6585
    Área sin interrupciones, 3463, 3777, 3962, 4370, 4568
Reloj del sistema, 1535
Reloj en tiempo real, 2235
Remanencia, 1632, 1913, 2258, 2260
    Bloque de datos, 2951
    Bloque de datos , 2952
    Interfaz del bloque, 2660
    Marcas, temporizadores, contadores, 2578
    Memoria de carga, memoria de trabajo, memoria de sistema, 1719
    Variable PLC, 2578
Remanencia de parámetros de dirección IP, 1501
Remanentes, 47
Rendimiento, 250
Renovar un certificado de grupo CA, 822
REPEAT, 4443, 6464
Repetidor de diagnóstico, 2210
REPL_VAL, 5783, 6004, 6110, 6501, 6635
REPLACE, 4734
Report System Errors, 942, (Véase Diagnóstico del sistema)
    OB de error creados, 2222
Representación
    Barra, 8424
    Botón, 8448, 9309
    Campo de fecha y hora, 8426, 9300
    Campo de texto, 8453
    Campo E/S, 8428, 9302
    Campo E/S gráfico, 8432
    Campo E/S simbólico, 8450
    Círculo, 8435
    Elipse, 8430
    Indicador de avisos, 8444, 8537
    Interruptor, 8447, 9310
    Línea, 8438
    Rectángulo, 8445
    Visor de avisos, 8438, 8439, 8442, 8535, 9323
    Visor de curvas, 8436, 9307
    Visor de gráficos, 8431
    visor de recetas simple, 8591
    Visor de usuarios, 8426
    Visor de usuarios simple, 9314
Representar valores
    como curva, 8499
REQ en instrucciones que funcionan asíncronamente, 5595
Requisitos
    Ajustar valor real, 10926, 11021
    Ajustar valor real al vuelo, 11021
    Aproximación en bucle, 10929, 11030
    Búsqueda del punto de referencia, 10915, 11009
    Captura de bordes, 11027
    Datos de encóder, 10933, 11034
    Decalaje de señal cero , 11023
    Definir punto de referencia, 10928, 11025
    JOG, 10911, 11005

Medición de longitud, 11027
Modo incremental, 10921, 11015
Requisitos de hardware, 94
instrucción de comunicación "AS_MAIL", 11420
instrucción de comunicación "PG_DIAL", 11416
instrucción de comunicación "SMS_SEND", 11419
Requisitos de hardware para el acoplamiento remoto AS-AS
instrucción de comunicación "AS_DIAL", 11418
Requisitos de software en la instalación
instrucción de comunicación "AS_MAIL", 11421
instrucción de comunicación "PG_DIAL", 11416
instrucción de comunicación "SMS_SEND", 11420
Requisitos de software en la programadora o en el PC
instrucción de comunicación "PG_DIAL", 11417
Requisitos de software para el acoplamiento remoto AS-AS
instrucción de comunicación "AS_DIAL", 11419
Requisitos del sistema en WinCC Basic, 96
Requisitos del sistema STEP 7 Professional, 93
Requisitos para enviar un correo electrónico, 11420
Requisitos para enviar un mensaje SMS, 11419
Requisitos para establecer una conexión VPN, 11408
RES_RCVB, 1071, 8242
RES_RECV, 1071
Resaltar
Conexión, 8767
Resaltar palabras clave, 2795
Reserva de memoria, 50, 1916
RESET, 5777, 5999, 6104, 6497, 6630
RESET_BF, 3291, 3600
ResetBit, 8683
ResetBitInTag, 8684
RESETI, 5781, 6002, 6107, 6500, 6633
RESETP, 5779, 6000, 6105, 6498, 6631
Resistencia PTC, 2128
Resolución, 9796, 9874, 10959, 11061
Valores admitidos, 9797, 9875
Valores posibles, 10960, 11061
Resolución del período, 9507, 9649, 9709
Respaldo, 1722
Restablecer, 1269, 9313
Configuración de fábrica, 2192, 2193, 9273
Configuración de fábrica , 9273
Parámetros de interfaz PROFINET, 2244
Restablecer copia de seguridad, 11307
Restablecer diseño de interfaz de usuario, 313
Restablecer dispositivo, 11307
Restablecer los ajustes MPI o PROFIBUS, 11325
Restablecer los ajustes TCP/IP, 11317
Restablecimiento de la configuración de fábrica, 1913
Restar, 3402, 3713, 4127, 4132, 4141, 5673, 5896, 6221, 6226, 6235
Restar tiempos con T_SUB, 4693, 6685
Restaurar
Datos del panel de operador, 9270, 9272, 9286
Restaurar los datos
Panel de operador, 9270, 9272, 9286
Resto de división, 3406, 3717, 5678, 5900
Restricciones debidas a derechos de usuario, 325, 326
Reconocer, 325
Resultado binario, 2545, 2559
Resultado lógico, 2545, 2559
Resumen de los identificadores de bits, bytes, palabras y palabras dobles, 1811
Resumen de los TS Adapter que pueden utilizarse, 11379
Resumen de versiones, 1699
RET, 3519, 3832, 5729, 5951
RET_VAL en instrucciones que funcionan asíncronamente, 5595
Retardo, 4018, 4624, 6086
Retardo al conectar, 3309, 3319, 3334, 3346, 3617, 3643, 3656, 3910, 4104, 4289, 4307, 4515, 5619, 5630, 5642, 5840, 5851, 5864, 6047, 6199, 6385, 6397, 6543
Con memoria, 3337, 3348, 3646, 3658, 4106, 4310, 5633, 5644, 5854, 5866, 6201, 6400
Retardo al desconectar, 3311, 3321, 3340, 3350, 3620, 3649, 3660, 3913, 4109, 4292, 4313, 4517, 5621, 5636, 5646, 5842, 5857, 5868, 6050, 6203, 6388, 6403, 6546
Retardo de tiempo, 3535, 3849, 4464, 5734, 5957, 6472, 6603
RetrocederPágina, 8689
RETURN, 4448, 6469
RI, 1018
RIGHT, 4731
RIP
Estadística RIPv2, 1263
RIPv2
Configuración, 1372
Interfaces, 1373
RLD, 4268, 6358
RLDA, 4272, 6361
RLO, 2545, 2559
Invertir, 3286, 3595, 4084, 5604, 5825, 6180
Poner a 0, 4086, 6182
Poner a 1, 4085, 6181

RMON
Estadística, 1348
Historial, 1349
RMP_SOAK
Instrucción, 7297
Parámetros de entrada, 7303
Parámetros de salida, 7303
RND, 4192, 6285
RND-, 4195, 6288
RND+, 4194, 6287
ROC_LIM
Instrucción, 7304
Parámetros de entrada, 7310
Parámetros de salida, 7310
ROL, 3562, 3877, 4482, 4641, 5756, 5979, 6491, 6622
Rol definido por el sistema
administrator, 766
diagnostics, 766
remote access, 766
standard, 766
Roles, 766
Definido por el usuario, 766
Definidos por el sistema, 766
Roles definidos por el usuario, 766
ROR, 3560, 3875, 4481, 4639, 5754, 5977, 6488, 6620
Rotar
Derecha, 3560, 3875, 4273, 4481, 4639, 5754, 5977, 6362, 6488, 6620
Izquierda, 3562, 3877, 4268, 4270, 4272, 4482, 4641, 5756, 5979, 6358, 6359, 6361, 6491, 6622
Rotulación del módulo, 514
Rotura de hilo, 2115, 2133, 9430, 9799, 9877, 10283, 10335, 10370, 10963, 11064
ROUND, 3494, 3807, 4417, 4594, 5715, 5938, 6447, 6589
Router, 551
Router estándar, 862
Routing, 1230
Rutas estáticas, 1230
VRRP, 1230
Routing IO, 1524
Routing S7
Carga, 9258
RRD, 4270, 6359
RRDA, 4273, 6362
RS, 3294, 3603, 5610, 5831
RT, 3325, 3634, 3919, 4298
RT_INFO, 4932
RTM, 2371, 4705
RTS, 1018
RTS de HW siempre ON, 1024
RTS de HW siempre ON, ignorar DTR/DSR, 1024
RUN, 1627, 1710, 1712, 1906
RUN de la CPU, 1053, 1196
Runtime
Imagen inicial, 8315
Procesar la lista de funciones, 8641
Rueda del ratón, 79
Simular, 9262
RUNTIME, 3536, 3850, 4019, 4465
Medir tiempo de ejecución, 3536, 3850, 4019, 4465
RW_REC, 8108

S

S, 3289, 3598, 4083, 4117, 5607, 5828, 6179, 6211
S_CD, 3365, 3676, 5658, 5881
S_COMP, 4707
S_CONV, 4708
S_CU, 4325, 6413
S_CUD, 3361, 3672, 4329, 5654, 5877, 6417
S_MODB, 1072
S_MOVE, 4706
S_ODT, 3334, 3643, 4307, 5630, 5851, 6397
S_ODTS, 3337, 3646, 4310, 5633, 5854, 6400
S_OFFDT, 3340, 3649, 4313, 5636, 5857, 6403
S_PEXT, 3331, 3640, 4304, 5627, 5848, 6394
S_PULSE, 3328, 3637, 4301, 5624, 5845, 6391
S_RCV, 1072
S_RTS, 1072
S_SEND, 1072, 1140
S_USSI, 1072
S_USSR, 1072
S_USST, 1072
S_V24, 1072
S_VSET, 1072
S_VSTAT, 1072
S_XON, 1072
S5TIME, 2341, 2394, 2443, 2518, 2533
S5TIME_TO_, 2443, 2533
S7 1200
Comunicación, 8841
Conexión HMI, 8843
Parámetros, 8891
Parámetros de conexión, 8891
Parámetros PROFIBUS, 8859, 8861, 8862
PROFIBUS, 8857, 8858
PROFINET, 8842
S7 1200 V2
Tipos de datos, 8887

S7 1500
Comunicación, 8792
Conexión HMI, 8794
Parámetros, 8836
Parámetros de conexión, 8836
Parámetros PROFIBUS, 8809, 8811, 8812
Parámetros PROFINET, 8796, 8797, 8799
PROFIBUS, 8806, 8808
PROFINET, 8793
Tipos de datos, 8835
S7 200
Comunicación, 8957
Conexión, 8957, 8986
MPI, 8955, 8965
Parámetro, 8955, 8960, 8965, 8966
Parámetros de conexión, 8960
PPI, 8966
Tipos de datos, 8985
S7 300
Comunicación, 8896
Dirección MPI, 8925
S7 300/400
Conexión HMI, 8898
MPI, 8918, 8919
Parámetros, 8950
Parámetros de conexión, 8950
Parámetros MPI, 8921, 8923, 8924
Parámetros PROFIBUS, 8913, 8914, 8916
Parámetros PROFINET, 8900, 8901, 8903
PROFIBUS, 8910, 8911
PROFINET, 8897
Tipos de datos, 8944
S7 400
Comunicación, 8896
Dirección MPI, 8926
S7-1200, 40
S7-1500
Campo de aplicación, 901
Definición, 901
Propiedades, 901
segmento de potencia, 902
S7-300
Reglas para slots, 899
S7-400
Reglas para slots, 900
S7-modular Embedded Controller, 1204
Configuración, 1205
S7-PCT, 1610
S7-Port Configuration Tool, 40
Salida
Insertar, 2697, 2741
Quitar, 2698, 2742
Salida (Q), 1631
Salida de avisos
Configurar:salida de avisos, 8525
Salida de habilitación ENO, 256
Salida de periferia, 1631
Salida de sentido y sentido de la marcha
Relación, 10434
Salida digital
Funciones, 9420, 9443, 9449
Valor de comparación, 9378, 9420, 9443, 9449
Salidas digitales
Activar y desactivar, 9519
comportamiento en los modos de contaje, 9518
comportamiento en los modos de medición, 9529
Comportamiento en los modos de montaje, 9599
habilitar, 9519, 9529, 9600
salir
Runtime, 9286
Salir
Programa de usuario, 3526, 3839, 4008, 4456, 4616, 5733, 5956, 6086, 6473, 6603
SALRM, 1418
Saltar, 3511, 3512, 3513, 3514, 3516, 3519, 3824, 3825, 3826, 3827, 3829, 3832, 4197, 4223, 5726, 5727, 5728, 5729, 5948, 5949, 5950, 5951, 6289, 6316
incondicional, 4198, 6291
Si BR = 0, 4206, 6299
Si BR = 1, 4205, 6297
Si el resultado es cero, 4210, 6303
Si el resultado es mayor o igual a cero, 4218, 6310
Si el resultado es mayor que cero, 4214, 6307
Si el resultado es menor o igual a cero, 4220, 6312
Si el resultado es menor que cero, 4216, 6309
Si el resultado no es cero, 4212, 6305
Si el resultado no es válido, 4222, 6314
Si OS = 1, 4208, 6301
Si OV = 1, 4207, 6300
Si RLO = 0, 4201, 4203, 6293, 6296
Si RLO = 1, 4199, 4202, 6292, 6295
Salto
Finalizar cadenas secuenciales, 2886
Salto a etapa
Borrar, 2901
Cambiar destino de salto, 2900
Insertar, 2899
Introducción, 2830
Navegar hasta el destino de salto, 2900
SAVE, 4087, 5764, 5985, 6182
SC, 3367, 3678, 5660

SCALANCE S, 751
SCALE, 3506, 3819, 3996, 4431, 4604, 5721, 5944, 6077, 6452, 6594
SCALE_M
Instrucción, 7311
Parámetros de entrada, 7312
Parámetros de salida, 7312
SCALE_X, 3500, 3813, 3987, 4422, 4599
SCL, 47, 2786, 3236, 3237, 3272, 3273
Comparación detallada, 2996
Insertar comentario, 4449, 6469
Migración, 141
Ventana de programación, 2797
Script
Actualizar valores de variables, 8461, 8776
Crear tipo, 9153
Librería, 9153
Scripts de usuario
en listas de funciones, 8638
SD, 3344, 3346, 3656, 4101, 4104, 5640, 5642, 5864, 6196, 6199
SE, 3344, 3654, 4101, 5640, 5862, 6196
Se interrumpe la conexión módem, 11426
causas y solución, 11426
Sección de comentario, 2781
Secuencia de bits, 2326, 2327
64 bits, 2328
Secuencia de fin, 1030
Secuencias de inicio, 1029
Security Configuration Tool
Vistas de configuración, 752
SEG, 3584, 3899, 4057, 4503, 4671, 5815, 6040, 6153, 6531, 6668
Segmento
Borrar, 2673, 2718, 2762, 2916
Contraer, 2674, 2718, 2763, 2917
Copiar, 2673, 2717, 2761, 2916
Expandir, 2674, 2718, 2763, 2917
Insertar, 2671, 2673, 2716, 2717, 2760, 2761, 2914, 2916
Insertar el título, 2675, 2720, 2764, 2918
Introducir comentario, 2676, 2721, 2765
Navegar en los bloques, 2677, 2722, 2766, 2919
Seleccionar, 2672, 2716, 2760, 2915
Utilizar, 2671, 2715, 2759, 2914
Segmento de barra
Definir, 8425
Segmento de potencia, 902
Segmento FUP
Borrar rama, 2750
Insertar rama, 2750
Insertar una llamada de bloque, 2687, 2731
Rama, 2749
Reglas relativas a ramas, 2749
Visualización del estado del programa, 3109
Segmento KOP
Borrar rama, 2707
Borrar un cruce, 2709
Cambiar la posición de un cruce, 2709
Cerrar rama, 2706
Cruce, 2708
Insertar cruce, 2708
Insertar rama, 2706
Insertar una llamada de bloque, 2687, 2731
Interconexiones no permitidas, 2680
Rama, 2705
Reglas para ramas paralelas, 2705
Visualización del estado del programa, 3108
Seguimiento
Easy Motion Control, 11221
Seguridad, 46
Seguridad (servidor web), 939
Seguridad de la transmisión, 1016
con 3964(R), 1017
con Freeport, 1016
con Modbus y USS, 1017
Seguridad de transmisión, 1078
con el driver ASCII, 1080
Con RK512, 1081
SEL, 3548, 3862, 4024, 4470, 4628, 5748, 5971, 6096, 6481, 6614
Selección de la función de puerta, 9469
Selección modificada, 8716
Selección múltiple, 515, 8335
Seleccionar, 3548, 3862, 4024, 4470, 4628, 5748, 5971, 6096, 6481, 6614
Varios objetos, 8328, 8335
Seleccionar el control de puerta, 9470, 9471, 9479, 9480, 9482, 9484
Seleccionar la vía de conexión de forma específica, 624
Selector de modo, 2236
Selectores de función
ABS_CTRL, 7438
ABS_CTRL_451, 7441
CAM_CTRL, 7474
CAM_CTRL_452, 7475
Sellado de tiempo, 2102
Sello de tiempo, 8510
Con transferencia de aviso con TIMESTMP, 6710
SEND_440, 1071
SEND_CFG, 5502
Send_Config, 1014, 1047

Send_Config (configuración de la transmisión), 5402, 7792
Send_P2P, 1014, 1047
Send_P2P (enviar datos punto a punto), 5410, 7800
SEND_PTP, 1071, 5510, 8237
SEND_RK, 1071, 8244
Seno, 3425, 3737, 4150, 4347, 4545, 5693, 5916, 6244, 6430, 6568
Sensor, 10641
Sensor Cu10, 2128
Sensor del punto de referencia, 10915, 11009
Sensor tipo NPN (en sumidero), 9413, 9439
Sensor tipo PNP (en fuente), 9413, 9439
Sensores NAMUR, 9556
Sentido principal de contaje, 9471, 9506, 9561, 9628, 9688
Señal BUSY, 1118
Señal de sentido de giro, 11198
Señal N, 9413, 9439
Señales cualificadoras, 1012
Señales cualificadoras RS232
    Manejo automático, 1025
Señales cualificadoras RS232C, 1090
    Manejo automático, 1091
Señales de control
    ABS_CTRL, 7421
    ABS_CTRL_451, 7425
    Escribir, 7420, 7424
Señales de realimentación del diagnóstico, 10934, 11035
    Datos en el DB de canal, 10935, 11036
    Desarrollo, 10935, 11036
Señales de realimentación del posicionamiento, 11035
    Datos en el DB de canal, 11035
Señales de realimentación para el posicionamiento
    Desarrollo, 11035
Señales de respuesta
    Leer, 7420, 7424
Señales de respuesta del posicionamiento, 10934
    Datos en el DB de canal, 10934
    Desarrollo, 10934
Señales de respuesta en el DB de canal
    Final de una operación de posicionamiento, 10911, 11005
Señales del encóder de 24 V, 9493
    Filtro de entrada, 9511, 9512, 9592, 9593
Señales del encóder de 5 V, 9492
    Vigilancia, 9493
Señales RS232, 1018
Serializar, 3445, 3759, 3947, 4358, 4555
Serialize, 3445, 3759, 3947, 4358, 4555
Serie FX1, 9061
Serie FX2, 9061
Serie FX3, 9051
Serie iQ, 9051
Serie Q, 9051
SERVE_RK, 1071, 8253
Service & Support, 8759
Service Packs .Net, 32
Service_Request_ID, 8126
Servicios FETCH/WRITE
    Configuración para transporte ISO, 676, 2031
Servicios IP, 778
Servicios MAC, 781
Servidor DCP, 1264
Servidor de correo electrónico, 688
Servidor de redundancia UA, 2057
Servidor DHCP, 864
Servidor FTP
    S7-1500, 2023
Servidor OPC
    Proveedor central de de servicios de comunicación, 1199
    Reglas para slots, 1200
    Uso como servidor SMTP, 687
Servidor OPC - Avisos de CPU, 2062
Servidor OPC UA, 2059
Servidor SSH, 1263
Servidor Syslog, 828
Servidor Telecontrol, 1208
Servidor web, 896, 936, 943, 944, 946, 949, 960, 976, 978, 983, 986, 998, 1752
    Activar, 941, 942, 980
    Administración de usuarios, 944
    administrar acceso, 941
    Clases de visualización de mensajes, 945
    Enumeraciones, 958, 995
    Fragmento, 958, 996
    HTTPS, 945, 981
Servidores NTP, 809
Servomotor, 10429
SET, 4085, 5772, 5993, 6098, 6181, 6493, 6624, 9512
SET_BF, 3290, 3599
SET_CINT, 4886
SET_CLKS, 6690
Set_Features, 1014, 1048
SET_TIMEZONE, 4700
SET_TINT, 4890
SET_TINTL, 4892
SetBit, 8691
SetBitInTag, 8692
SetConnectionMode, 8699

SetDeviceMode, 8690
SETI, 5775, 5997, 6102, 6495, 6628
SETIO, 4750
SETIO_PART, 4752, 6743
SetLanguage, 8696
SETP, 5773, 5995, 6100, 6494, 6625
SetPLCMode, 8695, 8696
Severidad, 797
SF, 3350, 3660, 4109, 5646, 5868, 6203
SGN_GET, 5515
SGN_SET, 5516
SHA1, 812, 818
Shared Device, 1577, 1593, 1919
    Controlador IO correspondiente, 2173
Shared Input interna del módulo (MSI), 1593
Shared Output interna del módulo (MSO), 1593
SHL, 3558, 3873, 4479, 4637, 5752, 5975, 6487, 6618
ShowAlarmWindow, 8708
SHR, 3556, 3871, 4477, 4635, 5750, 5973, 6485, 6616
SHRB, 5813, 6036, 6151, 6529, 6665
SiClock, 783
Signal Board, 975
    insertar, 976
Signal_Get, 1014, 1048
Signal_Get (consultar señales RS232), 5415, 7805
Signal_Set, 1014, 1048
Signal_Set (establecer señales RS232), 5417, 7807
Significado de las columnas utilizadas en la tabla de forzado permanente, 3145
Signo, 3411, 3723, 4144, 4533, 5680, 5903, 6239, 6557
SIMATIC LOGO!
    Comunicación, 8986
    Parámetros, 8988
    Parámetros de conexión, 8988
SIMATIC S7
    Interlocutores, 8841, 8896, 8954, 8964, 8967
SIMATIC S7 200
    Migración de los tipos de datos, 173
SIMATIC S7 300/400
    Migración de los tipos de datos, 173
SIMATIC S7 Embedded Controller, 42
Símbolo
    Categorías, 8526
    Para comparación de valores, 9940, 10455, 10482
Simbología, 250
Simbología homogénea, 250
Símbolos
    en el plano de ocupación, 3056
    en la estructura de dependencias, 3072
    en la estructura de llamada, 3065
    en TeleService, 11371
Símbolos de estado, 2074
Símbolos de la tabla de observación, 3118
Símbolos utilizados en la tabla de forzado permanente, 3147
Simulación, 81
    Conexión al controlador, 81
    Easy Motion Control, 11222
    Runtime, 9262
Simulador de variables, 9266
Simular
    Proyecto a través del simulador de variables, 9264
Simular dispositivos, 487
Simular hardware, 487
Simular módulos, 487
Simular software, 487
SimularTeclaSistema, 8700
SimularVariable, 8701
SIN, 3425, 3737, 4150, 4347, 4545, 5693, 5916, 6244, 6430, 6568
Síncrona
    Transferir datos, 8825, 8935, 9127
Síncronamente
    Transferir datos, 8981
Sincronismo
    Combinación de DP con y sin sincronismo, 1452
    Crear un programa de usuario, 1458
    Ejemplo: registro de varios puntos de medición, 1450
Sincronización, 9361, 9371, 9417, 9441
    Búsqueda del punto de referencia, 10914, 11008
    con señal N, 9376, 9417
    Con señal N, 9441
    De esclavos DP con DP_SYC_FR, 4800, 6797
    Definir punto de referencia, 10928, 11025
    mediante entrada digital, 9374
    Pérdida, 9762, 9840, 11025
    Relojes esclavos con SNC_RTCB, 4703, 6692
Sincronización (páginas web definidas por el usuario), 947, 984
Sincronización de la hora, 1750, 1953
Sincronización del sistema
    Durante el test de programas GRAPH, 3103
    Ejecutar, 3107
Sincronización horaria, 809, 1646, 1931, 1998, 2036
    CP PROFIBUS, 2036
Sincronizar, 8460, 8775
    Easy Motion Control, 11220
Sincronizar la hora, 1750, 1953

Sincronizar las páginas web definidas por el usuario, 5346
Sincronizar variable PLC, 8474
Sinopsis
acerca de los formatos de visualización en la tabla de forzado permanente, 3153
de los formatos de visualización en la tabla de visualización, 3126
Sinóptico de direcciones, 531, 923
SINT, 2330, 2382, 2421, 2468, 2488
SINT_TO_, 2421, 2488
Sintaxis
Direccionamiento, 9008
Sintaxis de los comandos AWP, 950, 987
Sintaxis para constantes, 1811
Sistema
Configuración del sistema, 1263
Información general, 1265
Sistema de automatización, 8735, 8760
Configurar, 8735
local, 11417
remoto, 11417
Sistema de avisos, 8502
Sistema de información
Componentes del sistema de información, 327
Roll-out, 328
Tooltip, 329
Sistema de medida
Seleccionar, 9782, 9860, 10935, 11036
Sistema de memoria de S7-400, 1720
Sistema de periferia descentralizada ET 200M, 2139
Sistema de seguridad, 9312
Sistema IO, 1504, 1597, 1600, 1603
Crear, 1505
Sistema IO de múltiples aplicaciones, 1597, 1600, 1603, 1606
Sistema maestro DP
Crear, 1387, 1394, 1401, 1427, 1463, 1466, 1467
Desconectar esclavo DP, 1400, 1472
Desconexión de la subred, 1397, 1470
Desconexión de los nodos, 1397, 1470
Determinar topología con DP_TOPOL, 4809, 6806
Editar propiedades, 1398, 1471
Información, lista de estado del sistema, 7001
Insertar esclavo DP, 1399, 1472
Resaltar, 1396, 1469
varios, 1391
Sistema monomaestro, 1390, 1465
Sistema operativo
Actualizar, 9273
Configuración (occidental), 9201
Configuración de idiomas asiáticos, 9201
Sistema operativo , 2249
Sistema operativo asiático, 9201
Sistema PROFINET IO, 1504
Sistemas multimaestro, 1391
SLC, 9029
SLD, 4264, 6354
Slicing, 252
Slot
Asignación de slots en DPV1, 1440
Datos de usuario con DPV1, 1440
Determinar dirección lógica con LGC_GADR, 5017, 7075
Determinar dirección lógica, con LOG_GEO, 5013, 7070
Modelo de slot, 1440
Rack, 514
seleccionar, 515
SLW, 4260, 6350
SM 338
alarma de diagnóstico, 10857
Avisos de diagnóstico, 10863
direccionarlo, 10860
modo isócrono, 10857
velocidad de transferencia, 10858
SMC, 3578, 3893, 4047, 4497, 4663, 5802, 6022, 6137, 6519, 6654
SMS despertador, 1209
SNC_RTCB, 4703
SNMP, 769, 1226, 1264, 1279, 2022
Activar, 2022
Grupos, 1279
SNMP Trap, 1278
SNMPv1, 1226
SNMPv2c, 1226
SNMPv3, 1226
Usuario, 1280
SNMPv1, 812
SNMPv3, 812
Sobrescritura del nombre de dispositivo PROFINET, 1606
Softkey, 8404, 8408
global, 8406
Softkey global, 8406
SOFTNET Security Client
Base de datos, 841
Configurar en el proyecto, 840
Crear un archivo de configuración, 840
Uso del, 840
SOFTNET Security Client, 751
Software
Panel de operador, 9268

Software de encriptación, 95, 99
Software SIMATIC NET PC para procesador de comunicaciones, 2042
Solapamiento de Ti y To
    Ajuste, 1461
    Principio de funcionamiento, 1459
    Reglas, 1462
    Requisitos, 1459
Solicitud de curvas
    Transferencia de curvas, 8830, 8993
Solo servidor HTTPS, 1263
soltar, 8722
Soltar tecla, 8724
Soporte del producto, 8759
SP, 3343, 3652, 4099, 5639, 5860, 6194
SP_GEN
    Instrucción, 7312
    Parámetros de entrada, 7314
    Parámetros de salida, 7315
Spanning Tree
    Información, 1244
    Rapid Spanning Tree, 1229
Speedy Splitter, 491, 494, 496, 507
SPLT_RAN
    Instrucción, 7315
    Parámetros de entrada, 7317
    Parámetros de salida, 7317
SQR, 4145, 4342, 6240, 6425
SQR , 3419, 3731, 4540, 5688, 5910, 6564
SQRT, 3421, 3733, 4147, 4343, 4541, 5689, 5912, 6241, 6426, 6565
SR, 3292, 3602, 5609, 5829
SRD, 4266, 6356
SRT_DINT, 4899
SRW, 4262, 6352
SS, 3348, 3658, 4106, 5644, 5866, 6201
SSD, 4258, 6348
SSI, 4256, 6346
STA, 2545
Startdrive
    Instalación, 60
Startup, 1675
Startup (S7-1500), 1962
Stateful Packet Inspection, 773
Station_Info, 919
STOP, 1627, 1703, 1710, 1906
STOP de la CPU, 1053, 1196, 9429
StopRuntime, 8702
STP, 3526, 3839, 4008, 4456, 4616, 5733, 5956, 6086, 6473, 6603
Strg_TO_Chars, 4714
STRG_VAL, 4709
String
    Comparar variables con S_COMP, 4707, 6715
STRING, 2309, 2350, 2408, 2458, 2480, 2506, 2522, 2538
    Direccionar, 2300
STRING_TO_, 2458, 2506, 2538
STRUCT
    Declaración en bloques de datos globales, 2946
    Declarar en la interfaz del bloque, 2652
    Declarar en un tipo de datos PLC, 2971
    Direccionar, 2300
    Estructura, 2358
STX, 1034
SUB, 3402, 3713, 5673, 5896
Submódulo, 1593
Submódulo base, 1919
Submódulo MSI, 1919
Subred, 553
Subred IP, 553
Sumar, 3400, 3711, 4126, 4131, 4135, 4140, 5894, 6220, 6225, 6229, 6234
    Sumar tiempos con T_ADD, 4691, 6685
Superior al tiempo de activación de la etapa, 4680, 4687, 6676, 6681
Superior al tiempo de activación ininterrumpido de la etapa, 4682, 4688, 6676, 6682
Superponer
    Variable, 2302
Superponer variables, 2654
Supervisión
    Acusar errores de supervisión, 3106
    Introducción, 2833, 2847
    Programar, 2911
Supresión de frecuencias perturbadoras, 1405, 2134
Sustitución, 9157
Sustitución de dispositivos, 529
Sustitución de módulos, 180
Sustitución de módulos en marcha, 2139
Sustituir
    Módulo, 529
Sustituir dispositivo, 529
    Limitación de conexiones, 9227
    Principio, 9227
SW_GATE, 9365
SWAP, 3466, 3779, 4372, 4570, 6575
SWITCH, 3516, 3829
    Instrucción, 7317
    Parámetros de entrada, 7318
    Parámetros de salida, 7318
Switchport, 1512, 8804, 8855, 8908
SYNC, 1403
SYNC_PI, 1748, 4742

SYNC_PO, 1748, 4744
Synchronous cycle (S7-1500), 1975
SZ, 5883

T

T, 4167, 6260
T STW, 4169, 6263
T_ADD, 4691
T_COMBINE, 4694
T_COMP, 4690
T_CONFIG, 5336
T_CONV, 4691
T_DIAG, 5331
T_DIFF, 4694
T_RESET, 5329
T_SUB, 4693
Tabla
    Buscar valor con TBL_FIND, 7048
    Calcular la desviación estándar con DEV, 7059
    Combinar un valor lógicamente y guardarlo con WRD_TBL, 7056
    Comparar valores de tabla con CDT, 7061
    Conexión, 8764
    Copiar valor con TBL_WRD, 7054
    Distribuir/recoger datos con PACK, 7065
    Ejecutar operaciones con TBL, 7052
    Emitir el primer valor con FIFO, 7046
    Emitir el último valor con LIFO, 7050
    Enlazar con TBL_TBL, 7063
    Introducir valor con ATT, 7045
Tabla de asignación File, 2014, 2024
Tabla de códigos ASCII, 720
Tabla de conexiones, 620, 633
Tabla de forzado permanente
    Significado de los símbolos, 3147
    Abrir, 3148
    Comprobación de sintaxis, 3149
    Conmutar entre el modo básico y el modo avanzado, 3146
    Display, 3148
    Estructura, 3145
    Funcionalidad, 3143
    Guardar, 3149
    Modo avanzado, 3146
    Modo básico, 3146
    Modo de observación y forzado, 3131
    Observar inmediatamente variables, 3159
    Operandos permitidos, 3150
    Operandos permitidos para valores de forzado permanente, 3151
    Posibilidades de test, 3143
    Significado de las columnas, 3145
    Sinopsis de los formatos de visualización, 3153
    Vista general de las posibilidades de test, 3143
Tabla de observación
    Significado de los símbolos, 3118
    Abrir, 3120
    Accesos múltiples a la misma CPU, 52
    Cargar bloques de datos mientras se procesa una petición de forzado, 52
    Comprobación de sintaxis, 3122
    Conmutar entre el modo básico y el modo avanzado, 3117
    Copiar, 3120
    Crear, 3119
    Ejemplo de cómo rellenar una tabla de observación, 3122
    Estructura, 3116
    Guardar, 3121
    Modo avanzado, 3116
    Modo básico, 3116
    Modo de observación y forzado, 3131
    Operandos autorizados, 3123
    Operandos permitidos para valores de forzado, 3124
    Posibles utilizaciones, 3115
    Servidor web, 940
    Significado de las columnas, 3116
    Sinopsis de las posibilidades de test, 3115
    Sinopsis de los formatos de visualización, 3126
    Test del cableado, 3115
Tabla de peticiones de TO: Comandos del menú contextual, 10487
Tabla de registro
    Registro de evento, 1241
Tabla de variables
    Definida por el usuario, 8458
    Estándar, 8458, 8773
    para dispositivos HMI, 8458, 8773
    personalizadas, 8773
    Servidor web, 940
Tabla:Diseñar borde, 8349
tablas de incrementos, 10902
    Datos en el DB de canal, 10904
Tablas de incrementos, 10997
    Cambiar, 10903, 10997
    Datos en el DB de canal, 10998
    Escribir, 10903, 10997
    Leer, 10903, 10998
TAK, 4274, 6363
Tamaño
    De varios objetos, modificar, 8355

Tamaño de fuente
Idiomas asiáticos , 9291
Tamaño de los datos de usuario en el dispositivo IO, 2004
Tamaño de telegrama en el dispositivo IO, 2004
Tamaño del objeto
Unificar, 8334
Tan, 4349, 6432
TAN, 3427, 3740, 4155, 4547, 5696, 5919, 6248, 6570
Tangente, 3427, 3740, 4155, 4349, 4547, 5696, 5919, 6248, 6432, 6570
TAR1, 4171, 6265
TAR1 <D>, 4172, 6266
TAR1 AR2, 4173, 6267
TAR2, 4174, 6268
TAR2 <D>, 4175, 6269
Tarea, 8727
Desactivar, 8730
Tareas controladas por eventos
Configurar:tareas controladas por eventos, 8516
Tareas de control, 11196
Tarjeta de memoria, (Véase Memory Card), (Véase Memory Card)
Tarjeta de programa, 1628
Tarjeta de transferencia, 1628, (Véase Memory Card)
Tarjeta o módulo Ethernet
Desenchufar/enchufar, 39
Tarjeta SD
Petición de forzado abierta, 33
Task Card, 8310
Cambiar modo de paleta, 303
Catálogo de hardware, 504, 8757
Contraer automáticamente, 294
Función, 301
Herramientas, 8325
Herramientas online, 2168
TCI (Tool Calling Interface), 1608
TCON, 5298, 5301
TCON_IP_RFC, 715
TCON_IP_v4, 714
TCON_Param, 711
TCONT_CP
Funcionamiento, 7099
Instrucción, 7098
Parámetros de entrada, 7111
Parámetros de entrada/salida, 7113
Parámetros de salida, 7112
Variables estáticas, 7113
TCONT_S
Funcionamiento, 7121
Instrucción, 7120
Parámetros de entrada, 7125
Parámetros de entrada/salida, 7126
Parámetros de salida, 7126
Variables estáticas, 7127
TCP, 778
Características, 709
Números de puerto, 716
TDISCON, 5305
TDP (reloj del sistema), 1447
Team Engineering
Crear un proyecto maestro, 11279
Ejemplo de la estructura del programa en el proyecto maestro, 11286
Indicaciones referentes a la compatibilidad, 11281
Introducción a la puesta en marcha en equipo, 11279
Procedimiento para crear el proyecto maestro, 11283
Procedimiento para integrar copias del proyecto en el proyecto maestro, 11285
Procedimiento para la puesta en marcha en equipo, 11283
Procedimiento para modificar objetos centrales en un proyecto maestro, 11285
Procedimiento para sincronizar manualmente cambios rivalizantes, 11284
Reglas para cargar en la CPU, 11289
Reglas para editar objetos centrales utilizados conjuntamente, 11288
Reglas para el proyecto maestro, 11286
Reglas para las funciones online, 11289
Reglas para trabajar en equipo en una CPU, 11288
Trabajar con copias del proyecto, 11280
Volver a integrar copias del proyecto en el proyecto maestro, 11281
Tecla de función, 8404
asignar globalmente, 8406
asignar localmente, 8318, 8408
Asignar un gráfico, 8412
Asignar una función, 8409
Imagen general, 8318
proteger con contraseña, 8411
Utilizar asignación general, 8318
utilizar para navegar por imágenes, 8416
Tecla directa
Configurar, 8414
Teclado de pantalla
Alfanumérico, 9295, 9297
Asignación de teclas, 9295, 9296

Comportamiento en caso de cambio de idioma, 77
KTP400 Basic, 9294
KTP600, KTP1000, TP1500 Basic, 9296
numérico, 9295, 9297
TELECONTROL SERVER BASIC, 1208
Telegrama de continuación, 1103
Telegrama de orden, 1103
telegrama de reacción, 1103
Estructura y contenido, 1105
Telegrama FETCH, 1103
Telegrama FETCH de continuación, 1111
Telegrama PUT, 1103
Telegrama SEND, 1103, 1107
Telegrama SEND de continuación, 1107
Telegrama SEND/PUT, 1103, 1104, 1107
Telegramas Ethernet-Non-IP, 773
Telegramas para Motion Control, 10647
TeleService
Propiedades del listín telefónico, 11369
significado de los símbolos, 11371
abrir el listín telefónico, 11372
Acceso a listines telefónicos, 11370
AS_DIAL, 8283, 11417
AS_MAIL, 8295
Capacidad de las redes telefónicas, 11380
Compatibilidad con módem, 11379
Configuración del listín telefónico, 11371
Definición de reglas de marcación, 11376
Definir reglas de marcación, 11376
Enviar un mensaje de correo electrónico, 8295
Enviar un mensaje SMS, 8289
Establecer una conexión con el AS, 8283, 11417
Establecer una conexión con una programadora o un PC, 8278
Exportar el listín telefónico, 11375
Funciones, 11368
Guardar el listín telefónico, 11373
Importar listín telefónico, 11373
Imprimir el listín telefónico, 11376
Insertar filas en el listín telefónico, 11372
Instrucción de comunicación: "AS_DIAL", 11368
Instrucción de comunicación: "PG_DIAL", 11368
Instrucción de comunicación: "SMS_SEND", 11369
Instrucción de comunicación:"AS_MAIL", 11369
Instrucción de comunicación:"TM_MAIL", 11369
Instrucción de comunicación:"TMAIL_C", 11369
Listín telefónico, 11369
Mostrar u ocultar columnas en el listín telefónico, 11373
PG_DIAL, 8278
Posibilidades para establecer una conexión:CPU S7-1200, 11415
Posibilidades para establecer una conexión:CPU S7-1500, 11415
Posibilidades para establecer una conexión:CPU S7-300/400, 11414
Routers, 11380
SMS_SEND, 8289
Tipos de módems/medios, 11380
Trabajar con el listín telefónico, 11370
Transmitir un mensaje de correo electrónico con TM_MAIL, 5588
variantes de rellamada, 11383
TeleService a través de GPRS, 1212
Establecer conexión, 1213
Temperatura de referencia, 2125, 2135, 2136
Temperatura de referencia dinámica, 2136
Temperatura de referencia fija, 2136
Template, 8404
temporal, 1209
Temporizador, 2341, 2343, 3328, 3637, 5624, 5845
Habilitar, 4091, 6187
Inicializar, 4097, 6193
Temporizador CEI, 2343
Temporizadores
Combinar fecha y hora con T_COMBINE, 4694
Comparar variables de tiempo con T_COMP, 4690
convertir con T_CONV, 4691
Determinar una diferencia con T_DIFF, 4694
Restar con T_SUB, 4693
Sumar con la instrucción T_ADD, 4691
Temporizadores CEI, 255
Tensión de alimentación, 2125
Test de cableado
Easy Motion Control, 11263
Test de parpadeo, 2209
Test del cableado, 3115
Texto
Botón, 8449
Campo de texto, 8453
Texto de aviso, 8511
Campos de salida, 70
Caracteres especiales, 70
Eliminar formato, 8520
Formatear, 8520
Texto de ayuda, 8511
Crear, 8516, 8518
Texto de diagnóstico, 10281, 10332, 10366
Texto del proyecto
exportar, 9190, 9210
Texto SMS, 5374, 5378, 5384

Textos
Migrar, 159
Textos de aviso, 1115
Caracteres de control, 1115
Diseño de página, 1115
Tabla de caracteres de control, 1115
Tabla de conversión de caracteres, 1115
Textos de usuario, 349
Textos del proyecto
Ejemplo de aplicación, 356
Exportar textos individuales, 355
Exportar todos los textos, 355
importar, 356, 9192, 9212
Migrar, 159
Modificación de textos de objetos seleccionados, 353
Mostrar el texto de referencia, 353, 9209
traducción a los idiomas del proyecto, 9206
Traducción a los idiomas del proyecto, 348
Traducir textos del proyecto, 352
traducir textos individuales, 9207
Textos del sistema, 349
TFTP
Cargar/guardar, 1272
THEN, 4436, 6456
THIS, 2295
Ti (momento en que se leen los datos de entrada), 1447, 1459, 1543
TIA Portal
Cerrar, 279
Iniciar, 279
Tiempo, 2341, 3326, 3635, 3921, 4299
Alternar horario de verano/invierno con SET_SW, 6704
Registrar, 3323
Tiempo agotado, 8726
Tiempo de aceleración, 10716, 10718, 10733, 10735
Tiempo de actualización, 1508, 9389, 9474
Tiempo de anticipación, 9741
Tiempo de ciclo, 1646, 1650, 1931, 1940
mostrar medido, 2176
mostrar parametrizado, 2170
Tiempo de ciclo de emisión, 1508, 1521, 1530
Tiempo de ciclo de emisión impar, 1521
Tiempo de ciclo de la leva, 9804, 9882
Tiempo de ciclo máximo, 1650, 1940
Tiempo de ciclo mínimo, 1627, 1650, 1906, 1940
Tiempo de cierre de sesión, 9313
Cambiar en runtime, 8620
Modificar, 8612
Tiempo de conmutación, 1665
Tiempo de deceleración, 10716, 10718, 10733, 10735
Tiempo de integración, 9474
Tiempo de medición, 9474
Tiempo de mensaje excedido, 1029
Tiempo de preprocesamiento Ti, 1448
Tiempo de retardo
ajustar, 8518
Modo isócrono, 1455
Tiempo de retardo de apagado, 11205
Tiempo de retardo de caracteres, 1030
Tiempo de retardo de caracteres TRC, 1041, 1130
Tiempo de supervisión de respuesta, 1508
Tiempo de vigilancia, 10905, 10906, 10946, 10999, 11000, 11047
Tiempo de vigilancia del ciclo, 1627, 1651, 1906, 1940, 3526, 3839, 4007, 4455, 4615, 5733, 5955, 6085, 6471, 6602
Tiempos
Combinar con T_COMBINE, 6687
Comparar variables de tiempo con T_COMP, 6683
Convertir con T_CONV, 6683
Determinar una diferencia con T_DIFF, 6686
Restar con T_SUB, 6685
Sumar con la instrucción T_ADD, 6685
TIME, 2343, 2395, 2444, 2476, 2501, 2517, 2532
Time delay interrupt, 1682
Time delay interrupt (S7-1500), 1971
Time error interrupt, 1686
Time error interrupt (S7-1500), 1978
Time of day (S7-1500), 1964
TIME_TCK, 4704
TIME_TO_, 2444, 2501, 2532
TIMER, 2366
Tipo
Librería, 9137
Tipo de accionamiento, 11198, 11199, 11205
Tipo de accionamiento Servo, 11198
Tipo de accionamiento Stepper, 11199
Tipo de arranque, 1646, 1931
Tipo de búsqueda del punto de referencia, 10951, 11052
Tipo de contador, 1655
Tipo de CPU
Allen-Bradley DF1, 9029
Allen-Bradley EtherNet/IP, 9006
Mitsubishi FX, 9061
Mitsubishi MC TCP/IP, 9051
Modicon Modbus RTU, 9093
Modicon Modbus TCP, 9079
Omron Host Link, 9109

Serie FX3, 9051
Tipo de datos
Convertir explícitamente, 3493, 3805, 4415, 4592, 6446
admisibles, 9004, 9092
Admisibles, 9050, 9060, 9078, 9108
Allen-Bradley, 9038
Allen-Bradley EtherNet/IP, 9004
ANY, 2362
ARRAY, 2354, 2355
Aviso de bit, 9038, 9067, 9114
BOOL, 2325, 2376, 2411, 2464, 2481, 2511, 2522
BYTE, 2326, 2377, 2412, 2465, 2482, 2512, 2523
CHAR, 2349, 2406, 2456, 2479, 2505, 2521, 2537
conversión, 2517
Conversión, 2373, 2379, 2461, 2467, 2476, 2477, 2478, 2507, 2514, 2519, 2520
Conversión explícita, 2411, 2412, 2414, 2416, 2418, 2421, 2424, 2426, 2428, 2430, 2433, 2435, 2438, 2440, 2441, 2443, 2444, 2446, 2447, 2449, 2450, 2452, 2453, 2454, 2456, 2457, 2458, 2460, 2481, 2482, 2483, 2485, 2488, 2490, 2491, 2493, 2495, 2497, 2498, 2500, 2501, 2502, 2503, 2504, 2505, 2506, 2522, 2523, 2524, 2526, 2528, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 2537, 2538
conversión implícita, 2511, 2513, 2515, 2516, 2518, 2521, 2522
Conversión implícita, 2376, 2377, 2378, 2380, 2382, 2383, 2384, 2385, 2386, 2388, 2389, 2390, 2392, 2393, 2394, 2395, 2397, 2398, 2399, 2401, 2402, 2403, 2404, 2406, 2407, 2408, 2409, 2464, 2465, 2466, 2468, 2469, 2470, 2471, 2472, 2473, 2474, 2475, 2477, 2479, 2480, 2512, 2516, 2520
DATE, 2344, 2402, 2452, 2478, 2502, 2519, 2534
DINT, 2333, 2386, 2430, 2472, 2495, 2516, 2530
DT, 2346, 2398, 2447, 2520, 2536
DTL, 2348, 2401, 2450, 2477, 2504
DWORD, 2327, 2379, 2416, 2467, 2485, 2514, 2526
Instrucción GRAPH, 2905
Instrucción SCL, 2807, 2808, 2809
INT, 2331, 2384, 2426, 2470, 2491, 2515, 2528
LDT, 2347, 2399, 2449
LINT, 2335, 2389, 2435
LREAL, 2339, 2393, 2441, 2475, 2500
LTIME, 2343, 2397, 2446
LTIME_OF_DAY, 2345
LTOD, 2404, 2454
LWORD, 2328, 2380, 2418
Mitsubishi, 9067
Mitsubishi FX, 9060
Mitsubishi MC TCP/IP, 9050
Modicon Modbus RTU, 9092
Modicon Modbus TCP, 9078
Omron, 9114
Omron Host Llink, 9108
PIP, 927
POINTER, 2359
REAL, 2338, 2392, 2440, 2474, 2498, 2516, 2531
S5TIME, 2341, 2394, 2443, 2518, 2533
SINT, 2330, 2382, 2421, 2468, 2488
STRING, 2350, 2408, 2458, 2480, 2506, 2522, 2538
STRUCT, 2358
TIME, 2343, 2395, 2444, 2476, 2501, 2517, 2532
TIME_OF_DAY, 2345
Tipo de datos PLC, 2367, 2967
TOD, 2403, 2453, 2477, 2503, 2520, 2535
UDINT, 2334, 2388, 2433, 2473, 2497
UINT, 2332, 2385, 2428, 2471, 2493
ULINT, 2336, 2390, 2438
USINT, 2331, 2383, 2424, 2469, 2490
Validez, 2321
Variable interna, 8464, 8779
VARIANT, 2364
WCHAR, 2349, 2407, 2457
WORD, 2327, 2378, 2414, 2466, 2483, 2513, 2524
WSTRING, 2351, 2409, 2460
Tipo de datos admisible
Allen-Bradley DF1, 9027
Tipo de datos PLC
Borrar, 2969
Comparación offline/offline, 2985
Comparar, 3011
Crear, 2968
Declaración en bloques de datos, 2947
Declarar, 2972
Declarar ARRAY, 2970
Declarar en la interfaz del bloque, 2653
Declarar STRUCT, 2971
Definición, 2367, 2967
Direccionar, 2300
Programar estructura, 2969
Propiedades de las variables, 2972, 2973
Tabla de declaración para los tipos de datos PLC, 2967
Tipos de datos PLC generados por el sistema en librerías, 50

Tipo de eje, 10948, 11049
Tipo de encóder, 11197
    TTL, 11202
Tipo de encoder Inversor, 2142
Tipo de encóder no disponible, 11197
Tipo de panel de operador
    Modificar, 9277
Tipo de sensor, 9413, 9439, 11202
    SSI, 11203
Tipo de sensor no existe, 11202
Tipo de señal, 9412, 9438, 9445
Tipo de telegrama
    Introducir, 11206
Tipos
    Búsqueda del punto de referencia, 10918, 11012
Tipos de comunicación
    Allen-Bradley, 9003
    Controlador acoplable, 9089
    Controladores acoplables, 9077
    EtherNet/IP, 9003
    habilitado, 9046, 9058, 9076, 9087, 9104
    Limitación, 9076, 9088
Tipos de datos, 251
    admisibles, 8835, 8887, 8944
    Admisibles, 8985
    Curvas, 8831, 8994
    Migración, 169
    S7 1200 V1, 8887
    S7 1500, 8835
    S7 200, 8985
    S7 300/400, 8944
Tipos de datos admisibles
    Curvas, 8831, 8994
Tipos de datos de hardware, 2371
Tipos de datos PLC, 46, 251
Tipos de eje, 9783, 9861
Tipos de enlace
    Bloques de parámetros para, 8160
Tipos de enumeraciones, 956, 994
Tipos de esclavos DP, 1399, 1471
Tipos de parámetros, 2366
Tipos de subbloque, 8157
TM, 1524
TM_MAIL, 5588
TMAIL_C, 51, 5286
TO, 4440
To (momento en que se emiten los datos de salida), 1459, 1543
To (tiempo en que se emiten los datos de salida), 1447
TO_PositioningAxis, 10454
TOD, 2345, 2403, 2453, 2477, 2503, 2520, 2535
TOD_TO_, 2453, 2503, 2535
TOF, 3311, 3321, 3620, 3629, 3913, 4292, 4517, 5621, 5842, 6050, 6388, 6546
TON, 3309, 3319, 3617, 3627, 3910, 4289, 4515, 5619, 5840, 6047, 6385, 6543
TONR, 3314, 3323, 3622, 3631, 3916, 4295, 4520, 5808, 6028, 6525, 6660
TONR_X, 6145
Tool Calling Interface (TCI), 1608
Tooltip
    mostrar, 9298
Topologías, 1124
TP, 3306, 3316, 3615, 3625, 3907, 4286, 4512, 5617, 5838, 6044, 6382, 6540
TP177A
    Cargar un proyecto, 76
Trace, 11327
    administración de Trace, 11336
    almacenamiento de configuración de Trace, 11348
    almacenamiento de datos, 11329
    capacidad funcional, 11356
    captación de variables, 11354
    carga de CPU, 11356
    ciclo de registro, 11343, 11366
    condiciones de registro, 11359, 11362
    configuración de Trace, 11329, 11337, 11347, 11348, 11351, 11365
    cursores de medición, 11352
    diagrama de curvas, 11332, 11352
    Disparador, 11361
    disparo, 11367
    dispositivos admitidos, 11327
    duración de registro, 11366
    editor de Trace, 11347
    guía de inicio rápido, 11341
    imprimir, 11353
    interfaz, 11330, 11356, 11357
    marca de tiempo, 11340
    medición, 11329, 11339, 11350
    muestreo, 11343
    navegador de proyecto, 11332
    niveles de registro, 11355
    predisparo, 11362
    Reducción, 11360
    registro, 11329, 11349
    señales, 11365
    Señales, 11358
    tabla de señales, 11334, 11352
    Traces en dispositivo, 11338
    traza de bits, 11332
    variable de disparo, 11361

Variable de disparo, 11359
vida útil de los valores, 11355
Trace S7-1200/1500, 11354
Traducir
Editor, 9205
Traducir textos, 348
Tráfico de datos bidireccional, 1015
Tráfico de datos unidireccional/bidireccional, 1019
Transfer Card; véase Tarjeta de transferencia, 1628
Transferencia, 6260, 6263, 6266, 6267, 6268, 6269
Transferencia coordinada, 8573
con controlador, 8573
Transferencia de datos asíncrona, 1015
Transferencia de datos con el maestro Modbus ET 200S, 1140
Transferencia de datos serie, 1015
Transferencia de datos UART, 1658
Transferencia de datos, lanzar, 5410, 7800
Transferencia de parámetros, 2268, 2270, 2272
Transferencia de valores
CNT_CTL1, 7400
Parámetros del DB para modos de contaje, 7401
Parámetros del DB para modos de medición, 7403
Transferencia de valores de parámetros, 2269
Transferir, 4167, 4169, 4171, 4172, 4173, 4174, 4175, 6265
Registro de receta, 9338
Transferir datos
Causa posible, 8829, 8882, 8940, 8979, 9132
Con sincronización, 8825, 8935, 8981, 9127
Inicio desde la vista de recetas, 8878, 9128
Inicio mediante una función configurada, 8828, 8881, 8939, 8978, 9131
Operación en el visor de recetas, 8936, 8976
Orden de control, 8826, 8879, 8937, 9129
Petición del controlador, 8977
Puntero de área, 8779, 8815, 8866, 8926, 8967
puntero de área "Coordinación", 8870, 8971
Puntero de área "Coordinación", 8819, 8929, 9121
puntero de área "Fecha/hora del controlador", 8928, 8969
Puntero de área "Fecha/hora del controlador", 8818, 9120
puntero de área "Fecha/hora", 8867, 8968
Puntero de área "Fecha/hora", 8816, 8868, 8927, 9119
Puntero de área "Identificador del proyecto", 8820, 8873, 8930, 8972
Puntero de área "Petición de control", 8874, 8973, 9123
Puntero de área "Petición del controlador", 8821, 8931
Puntero de área "Registro", 8824, 8876, 8934, 8975, 9126
Puntero de área 'Identificación de proyecto', 9122
Puntero de área Número de imagen, 8816, 8872, 8926, 8968, 9118
Punteros de área, 9116
Sin sincronización, 8824, 8877, 8934, 8980, 9126
Transferir el proyecto
Panel de operador, 81
Registro de receta, 81
Transferir parámetros, 2274, 2276, 2277, 2278, 2279
Transición
Borrar, 2895
Cambiar nombre, 2892
Copiar y pegar, 2894
Insertar, 2888
Introducción, 2833
Parámetros estáticos, 2859
Programar, 2912
Renumerar, 2890
Transición de red, 551
Transiciones de estado operativo, 1053, 1196
Transparencia
En gráfico, 8356
Transparencia del código, 1032
Tratamiento de errores, 2936
GetError, 3527, 3840, 4009, 4456, 4616
GetErrorID, 3531, 3844, 4013, 4460, 4620
Principios básicos, 2932
Tratamiento de errores local, 2934
Tratamiento de errores local, 2934
Información de error, 2934
Instrucciones, 2934
Prioridades, 2935
TRCV, 5314, 5318
TRCV_C, 5274, 5279
TRUE, 4283, 4284
TRUNC, 3499, 3812, 4193, 4421, 4598, 5719, 5942, 6286, 6450, 6593
TS Adapter, 81
TS Adapter IE
Descripción, 11394
funcionamiento, 11395
parametrización estándar, 11396
parametrizar, 11399, 11413
TS Adapter IE Advanced
Descripción, 11410
Parametrizar, 11413
Posibilidades de parametrización, 11412

Tipos de conexión, 11411
TS Adapter MPI
conexión directa, 11388
conexión módem, 11389
configuración de la conexión directa, 11389
configuración de la conexión módem, 11389
Descripción, 11387
funcionamiento, 11388
parametrizar, 11391
posibilidades de parametrización, 11390
realizar parametrización , 11391
restaurar la parametrización estándar, 11392
TS Adapter MPI
parametrización estándar, 11388
TS Adapter IE Basic
conexión a la red GSM, 11397
conexión a la red telefónica, 11397
conexión a la red telefónica a través de un módem externo, 11398
Tipos de conexión, 11396
TS Adapter MPI
crear derechos de acceso, 11384
exportar parámetros del adaptador, 11393
importar parámetros del adaptador, 11394
TSAP, 2032
Estructura, 568, 646, 719
Tabla de códigos ASCII, 720
TSAP ASCII, 720
TSEND, 5307, 5310
TSEND_C, 5263, 5267
TUN_EC
Funcionamiento, 7133
Instrucción, 7129
Parámetros de entrada, 7138
Parámetros de entrada/salida, 7142
Parámetros de salida, 7140
TUN_ES
Funcionamiento, 7152
Instrucción, 7149
Parámetros de entrada, 7157
Parámetros de entrada/salida, 7161
Parámetros de salida, 7159
TURCV, 5326
TUSEND, 5323
TypeOf, 4331
TypeOfElements, 4332

U

UBLKMOV, 51, 3487, 3800, 3980, 4409, 4586, 5708, 5931, 6070, 6440, 6582
UC, 4241, 6332
UDINT, 2334, 2388, 2433, 2473, 2497
UDINT_TO_, 2433, 2497
UDP, 778, 785, 807
Características, 710
Números de puerto, 716
UDT, 46, 251, 262
UFILL_BLK, 3463, 3777, 3962, 4370, 4568
UINT, 2332, 2385, 2428, 2471, 2493
UINT_TO_, 2428, 2493
ULINT, 2336, 2390, 2438
ULINT_TO_, 2438
UMOVE_BLK, 3459, 3773, 3958, 4366, 4564
Unidad de red, 33
Unidad periférica descentralizada ET 200iSP
Definición, 2098
Unidades de medida, 10645
Unificar
Tamaño del objeto, 8334
Unión fría, 1406, 1476, 2101, 2116, 2125, 2135, 2136
Unión fría interna, 2136
Universal
PROFIBUS, 8864, 8917
Univocidad
de nombres de objeto, 147
Unknown peers, 820
UNSCALE, 3508, 3821, 3999, 4433, 4607, 5723, 5946, 6080, 6454, 6597
UNTIL, 4443, 6464
UPDAT_PI, 4738
UPDAT_PO, 4740
Upload, 1612
URCV, 5256
USEND, 5255
USINT, 2331, 2383, 2424, 2469, 2490
USINT_TO_, 2424, 2490
Uso de UDTs, 262
Uso de un TS Adapter para TeleService, 11378, 11402
Uso del teclado de pantalla, 324
USS_DRIVE, 5524
USS_Drive_Control, 1014, 1050, 5435, 5440, 7825, 7831
USS_PORT, 5523
USS_Port_Scan, 1014, 1050, 5435, 5438, 7825, 7828
USS_Read_Param, 1014, 1050, 5435, 5443, 7825, 7834
USS_RPM, 5527
USS_WPM, 5529
USS_Write_Param, 1014, 1050, 5435, 5445, 7825, 7835

Usuario
Asignar roles, 768
Configurar, 765
Crear roles, 766
Usuario de diagnóstico, 766
Usuario estándar, 766
Usuario Remote Access, 766
Usuarios, 9313
actualizar al cambiar de usuario, 8732
Admin, 9319
Administrar, 8611, 8619
Asignar un grupo de usuarios, 8610
Borrar, 8612, 9319
Borrar en runtime, 8620
Cambiar nombre, 8611
Cerrar sesión, 9316
Crear, 8609, 8632, 9315
Crear en runtime, 8618
Crear, manejo con pantalla táctil, 9317
Crear, manejo táctil, 9315
desbloquear, 8621
Iniciar sesión, 8621
Modificar, 8620
Modificar, manejo con pantalla táctil, 9318
Registrar inicio de sesión, 8626
Utilización, 8569
Botón, 9309
Campo de fecha y hora, 9300
Campo E/S, 9302
Campo E/S gráfico, 9305
Campo E/S simbólico, 9311
de recetas, 8569
Función de sistema, 8636
Interruptor, 9310
Ventana de avisos simple, 9323
Visor de avisos simple, 9323
Visor de curvas, 9307
Visor de gráficos, 9304
Visor de recetas, 8446
Visor de usuarios, 8425
Visor de usuarios simple, 9314
Utilizar
Tecla de función para navegación de imágenes, 8416
Utilizar asignación general
Tecla de función, 8318
Utilizar marcos y portadas de la librería, 398
Utilizar variable, 8427

V

V24_SET, 1070
V24_SET_340, 1070
V24_SET_441, 1071
V24_STAT, 1070
V24_STAT_340, 1070
V24_STAT_441, 1071
VAL_STRG, 4711
Valor absoluto, 3411, 3723, 4144, 4335, 4533, 5680, 5903, 6239, 6420, 6557
Valor analógico, 3500, 3503, 3813, 3816, 3987, 3989, 4422, 4424, 4599, 4601
Valor asociado
Configuración, 3190
Ejemplo, 3191
Insertar, 3189
Valor de arcotangente, 3432, 3744, 4551, 5700, 5923, 6574
Valor de arranque, 49, 1910, 9361, 9416
Valor de carga, 9486, 9512, 9593
con anticipación, 9486
directo, 9486
Valor de comparación, 9378, 9420, 9443, 9449, 9519, 9600
Contaje, 9420, 9449
medición, 9443
Valor de forzado
Operandos permitidos, 3124
Valor de forzado permanente
Operandos permitidos, 3151
Valor de variable
Imprimir, 8499
Valor exponencial, 3423, 3736, 4148, 4544, 5692, 5915, 6242, 6567
Valor inicial
Variable, 8477, 8478
Valor límite, 2125, 3417, 3729, 4340, 6423, 8511
Variable, 8476
Valor límite inferior, 9528, 9651, 9710
Valor límite superior, 9528, 9651, 9710
Valor real absoluto, 10651
Valor real adicional del sensor, 11210
Valor real del sensor
adicional, 11210
Valor real incremental, 10650
Valor sustitutivo, 5783, 6004, 6110, 6501, 6635, 9521, 9569
Módulo de reserva, 1410, 1480
Valor/rango
Lista de textos, 8378
Valores
Comparar, 9940, 10455, 10482
Valores actuales
Easy Motion Control, 11259

Valores de retorno
ABS_CTRL, 7421
ABS_CTRL_451, 7425
ABS_DIAG, 7423
ABS_DIAG_451, 7427
ABS_INIT , 7420
CAM_CTRL, 7454
CAM_CTRL_452, 7458
CAM_DIAG, 7456
CAM_DIAG_452, 7459
CAM_INIT, 7453
CAM_MSRM_452, 7461
Instrucciones PtP, 5397, 7788
Valores de retorno de las instrucciones PtP, 5397, 7788
Valores de retorno del tiempo de ejecución de la recepción, 5413, 7803
Valores reales, 10650
Variable
actualización, 8485
Actualización, 8479
Ascendente, 3303, 3612, 3904
Borrar, 8470
Cambiar controlador, 8482
Cambiar nombre, 8470
ciclo de adquisición, 8485
Ciclo de adquisición, 8479, 8498
comentario, 8467
conexión con el controlador, 8465
configuración, 8471
configurar varias variables, 8471
Copiar, 8470
crear una variable externa, 8465
crear una variable interna, 8467
Descendente, 3305, 3613, 3905
direccionamiento, 8465
direccionar de forma absoluta, 8461, 8776
direccionar de forma simbólica, 8462, 8777
Direccionar indirectamente, 8483
en runtime, 8479
Escalar de forma lineal, 8480, 8481
Evento, 8484
exportar, 9180
importar, 9182
Lectura continua, 8479
Límite, 8475
lista de objetos, 8466
longitud, 8467
Longitud máxima, 77
Modo de adquisición, 8479
Mostrar u ocultar informaciones de variables, 2703, 2747, 2773, 2809, 2921
Multiplexar, 8483
Nombre, 8467
Reconectar, 8474
Representar valores, 8499
Resumen, 2286
Superponer, 2302
tipo de datos, 8467
valor inicial, 8477
Valor inicial, 8478
Valor límite, 8476
Variable externa, 8460, 8775
Variable interna, 8464, 8779
Variable PLC, 2566
Variables de índices, 8483
Variables indexadas, 8483
Variables PLC y variables DB, 2287
Visualizar en el aviso, 8519
Variable DB
Definición, 2287
Variable de disparo, 8511
Variable de matriz, 8494
Char, 69
Variable ERPC, 2013
Variable especial (servidor web), 954, 991
Variable global, (Véase Variable PLC)
Variable IEEE, 2099
Variable local, 2642
Variable PLC
Importar y exportar, 2591, 2592, 2593
Actualizar resultados de la comparación, 385
Comparación, 2982
Comparación offline/offline, 2985
Comparar, 3009
Comparar tabla de variables PLC, 2985
Comportamiento remanente, 2578
Copiar, 2589
Declarar, 2575, 2577, 2579, 2582
Definición, 2287
Direcciones y tipos de datos admisibles, 2573
Observar, 2588
Ordenar filas, 2590
Propiedades, 2585, 2586
Reconectar, 8474
Reglas, 2573
Rellenar celdas automáticamente, 2591
Tabla de variables PLC, 2566, 2567, 2573, 2574, 2575
Variables, 2012, 2013
Intercambio de datos, 8740
Migración, 169
Migrar, 155
Observar inmediatamente, 3134, 3159

Observar todos, 3133, 3159
Principios básicos, 8457, 8772
Variables de índices, 8483
Variables del objeto tecnológico Eje de posicionamiento, 10798
Variables del objeto tecnológico Eje de velocidad de giro, 10819
Variables del objeto tecnológico Encóder externo, 10830
Variables estáticas
High_Speed_Counter, 5131
Variables externas
Intercambio de datos, 8740
Variables HART
configurar, 2140
Estructura, 2141
Quality Code, 2141
Variables indexadas, 8483
Variables internas
Intercambio de datos, 8740
Variables PLC, 951, 953, 988, 990
Variables temporales (TEMP)
En el programa cíclico (S7-300, S7-400), 1822
Necesario para OB, 1883
VARIANT, 269, 2364, 3388, 3389, 3390, 3392, 3480, 3481, 3483, 3700, 3701, 3702, 3703, 3793, 3794, 3796, 3992, 3994, 4331, 4332, 4402, 4403, 4405, 4427, 4429
VARIANT_TO_DB_ANY, 3992, 4427
VariantGet, 3480, 3793, 4402
VariantPut, 3481, 3794, 4403
Velocidad de inicio en la búsqueda del punto de referencia, 10956, 11057
Velocidad de parada, 10947, 11048
Vigilancia, 10905, 10999
Velocidad de simulación, 9792, 9870
Velocidad de transferencia, 10869, 10881
Velocidad de transferencia / dúplex, 8803, 8854, 8907
Velocidades de transferencia de datos, 1011
Ventajas del uso de TeleService, 11368
Ventana de avisos, 64, 8525
Configurar, 8528
En runtime, 8534, 9322
ventana de avisos simple
Utilización, 9323
Ventana de avisos simple
Utilización, 8535
Ventana de inspección
Contraer automáticamente, 294
editor de hardware y redes, 502
Editor de hardware y redes, 8755
Estructura, 299
Evento, 64
Ficha Diagnóstico, 2159
Referencia cruzada, 64
Ventana de medición (PUERTA), 2112
Ventana de programación
Ajustar, 2798
GRAPH, 2871
Instrucciones permanentes anteriores, 2873
Instrucciones permanentes posteriores, 2878
SCL, 2797
Vista de avisos, 2879
Vista de cadena, 2874
Vista de etapa individual, 2876
Ventana de vigilancia, 2143
Ventana general, 59, 306
Clasificar la vista detallada, 309
Mostrar todos los bloques, 309
Mostrar y ocultar columnas, 309
Ventana sinóptica
Comparar objetos, 308
Verificación CEI, 2461
Ajustar, 2464
Verificación IEC
Ajustar, 2376, 2511
Versión de firmware V4, 31
Versión de hardware, 1240
Versión de módulo
Actualizar, 532
Versión de panel
cambiar, 9249
Comprobar, 9250
Versión de panel de operador, 9249, 9250, (Versión de panel), (Versión del dispositivo)
Versión de proyecto
Actualizar, 9248
Versión de Runtime, 9255
Versión de software, 1240
Versión del dispositivo, 9259
Actualizar, 9272
Versionamiento de tipos
Versión en proceso, 460
Versión en test, 460
Versión habilitada, 461
Versiones .Net, 32
Versiones de WinCC, 9245
Compatibilidad, 9245
Vía de conexión
Resaltar en la vista de redes, 630
Vida útil SA, 818
Vigilancia
Velocidad de parada, 10905, 10999

Zona de destino, 10905, 10999
Vigilancia de errores
Alimentación, 1294
modificación de estado de conexión, 1294
Redundancia, 1295
Vigilancia de errores de seguimiento, 10672, 10673
Vigilancia de inestabilidad de señal, 2143
Vigilancia de las señales del encóder de 5 V, 9493
Vigilancia de límites, 9476
Vigilancia de posicionamiento, 10672
Vigilancias, 10905, 10999
JOG, 10913, 11007
Vigilar, 1511, 8803, 8854, 8907
Visor de avisos, 64, 2153, 8438, 8441, 8525, 8536, 9324
Acusar un aviso, 2157
Columna, 8440
Configurar, 8527
Configurar representación, 8527
Elemento de control, 8440
Estructura de los avisos en la vista "Avisos activos", 2155
Estructura de los avisos en la vista de fichero, 2154
Exportar fichero, 2155
Ignorar avisos, 2157
Manejo, 8536, 9324
Manejo del teclado, 2158
Recibir avisos, 2154
Representación, 8439, 8535, 9323
Vaciar fichero, 2155
ventana de avisos, 9324
Ventana de avisos, 8536
Vista "Avisos activos", 2155
Vista de fichero, 2153
visor de avisos simple, 9323
Visor de avisos simple
Elemento de control, 8536, 9324
Manejo, 8536, 9323
Utilización, 8535, 9323
Visor de curvas, 8435, 9307
Botón, 8436
configurar para los valores del controlador, 8499
Elemento de control, 9308
Manejo, 9307
Representación, 9307
Utilización, 9307
Visor de diagnóstico del sistema
Configurar, 8566
Vista de dispositivos, 8452
Vista del búfer de diagnóstico, 8452
Vista detallada, 8452
Visor de gráficos, 8430, 9304
Color transparente, 8431
Manejo, 9304
Utilización, 9304
Visualización, 9304
Visor de recetas, 8577
actualización, 8578
Barra de herramientas, 8446
comportamiento al cambiar de imagen, 8578
configurar, 8588
manejar con teclas de función, 8578
registro de receta, 8579
Registro de receta, 8571
Representación, 8446
simple, 8577
Utilización, 8446
Visor de recetas simple, 8577
comando, 8592
Comando, 9332, 9333
comportamiento, 8592
Comportamiento, 9333
manejo, 8591
Manejo, 9332
representación, 8591
Visualización, 9332
Visor de topología, 2021
Visor de usuarios, 8614
configurar, 8617
Número de líneas, 8426
Representación, 8426
Utilización, 8425
Visor de usuarios complejo , 8616
Visor de usuarios simple, 8615
Visor de usuarios simple, 9314
Manejo, 9315
Representación, 9314
Utilización, 9314
VisorDeAvisosAcusarAviso, 8676
VisorDeAvisosEditarAviso, 8675
VisorDeAvisosMostrarTextoAyuda, 8676
VisorDeCurvasActivarModoRegla, 8667
VisorDeCurvasAmpliar, 8665
VisorDeCurvasAvancePágina, 8664
VisorDeCurvasComprimir, 8666
VisorDeCurvasIniciarParar, 8668
VisorDeCurvasIrAlPrincipio, 8668
VisorDeCurvasReglaDerecha, 8666
VisorDeCurvasReglaIzquierda, 8667
VisorDeCurvasRetrocesoPágina, 8665
VisorDeRecetasAbrir, 8679
VisorDeRecetasAtrás, 8682
VisorDeRecetasBorrarRegistro, 8678

VisorDeRecetasCambiarNombreDeRegistro, 8681
VisorDeRecetasEscribirRegistroEnControlador, 8680
VisorDeRecetasGuardarRegistro, 8681
VisorDeRecetasGuardarRegistroComo, 8681
VisorDeRecetasLeerRegistroDelControlador, 8678
VisorDeRecetasMenú, 8679
VisorDeRecetasMostrarTextoDeAyuda, 8682
VisorDeRecetasRegistroNuevo, 8678
Vista de avisos, 2879
Vista de cadena, 2874
Vista de dispositivos
    Área de módulos no enchufados, 516
    Editar parámetros, 529
    Editar propiedades, 529
    Editor de hardware y redes, 494, 8751
    Insertar módulo, 523
    Insertar Signal Board, 975
    Rack, 514, 910
Vista de etapa individual, 2876
Vista de librería
    Abrir, 435
    Resumen, 9135
    Salir, 435
    Sinopsis, 432
Vista de redes, 8764
    Agregar dispositivo, 521
    Editor de hardware y redes, 491, 8744
    Significado, 629
Vista del portal, 284
Vista del proyecto, 285
Vista detallada, 305
Vista general de dispositivos
    Área de direcciones, 914, 915, 971
Vista general de la lista de referencias cruzadas, 3083
Vista general de redes
    Funciones básicas, 538
    Funciones básicas para editar la tabla de la vista general de redes, 539
Vista online y de diagnóstico, 2158
vista topológica
    Aplicar interconexiones de puertos determinadas online, 749
Vista topológica
    Agregar dispositivo, 521
    Aplicar dispositivos determinados online, 750
    Comparar offline/online, 739
    Diferencias con respecto a la vista de redes, 734
    Editor de hardware y redes, 496, 8753
    Estado de diagnóstico en la vista de tabla, 739
    Estado de diagnóstico en la vista gráfica, 738
    Funciones, 733
    Interconectar puertos, 741
    Topología configurada, 736
Vistas de la lista de referencias cruzadas, 3081, 9194
Visualización, 1205
    Visor de gráficos, 9304
    Visor de recetas simple, 9332
Visualización de conexiones en la vista de redes, 620
Visualización de los números de línea, 2799
Visualización de una tabla de forzado permanente, 3148
Visualización online
    Color naranja, 2167
VLAN, 1223
    Prioridad, 1314
    Tag, 1314
    VID de puerto, 1314
    VLAN ID, 1226
    VLAN-tag, 1224
VLAN-Tagging, 816
VMware - CP PC, 44
VOID, 2366
Voltear
    Objeto, 8328
Volumen de datos bruto, 2004
Volumen de datos neto, 2004
VRRP
    Configuración, 1361
    Router, 1360
    Router de reserva, 1230
    Router maestro, 1230
    Router virtual, 1230
    Router VRRP, 1230
    Sinopsis de direcciones, 1362
    VRRP Statistics, 1246

W

WAIT, 3535, 3849, 4018, 4464, 4624, 5734, 5957, 6086, 6472, 6603
Wake On LAN, 4833, 6832
WCHAR, 2349, 2407, 2457
WCHAR_TO_, 2457
WHILE, 4442, 6463
WinCC V7.0 SP3, 62
WinCC flexible Runtime
    salir, 9286
Wireless-LAN/PB-Link, 1488
WORD, 2327, 2378, 2414, 2466, 2483, 2513, 2524
WORD_TO_, 2414, 2483, 2524

WORD_TO_BLOCK_DB, 2539
WR_DPARM, 4882
WR_LOC_T, 4698
WR_REC, 4790
WR_SYS_T, 4695
WRIT_DBL, 51, 4999
WRITE_BIG, 4400
WRITE_LITTLE, 4396
WriteToArrayDB, 3470, 3784, 3967, 4376, 4574
WriteToArrayDBL, 3476, 3789, 3973, 4381, 4580
WRREC, 919, 4748
WSR, 5809, 6030, 6147, 6526, 6662
WSTRING, 2309, 2351, 2409, 2460
    Direccionar, 2300
WSTRING_TO_, 2460
WWW (instrucción), 961, 999
WWW Sincronizar las páginas web definidas por el usuario, 5346

X

X, 4068, 6164
X(, 4076, 6172
XN, 4069, 6165
XN(, 4078, 6173
XOD, 4253, 6344
XON/XOFF, 1022
XOR, 2792, 2793, 3542, 3591, 3593, 3856, 5743, 5823, 5824, 5966
XOW, 4247, 6337

Y

Y, 3588, 3589, 5819, 5821

Z

Z_VORW, 3363, 3674, 5656, 5879
Zona de destino, 10890, 10945, 11046
    Vigilancia, 10905, 10999
Zona de parada, 10890, 10947, 11048
Zona de sobrecarga, 11201
Zona de trabajo, 10889
Zona muerta
    ajustar, 8518
Zonas de conexión, 10889
ZONE
    Instrucción, 7610
    Parámetros de entrada, 7612
    Parámetros de entrada/salida, 7612
    Variables estáticas, 7613, 7617
Zoom
    Ajuste del nivel de zoom, 492, 494, 497, 8745, 8752, 8754
    Comandos de teclado, 508
    Imagen, 8312